Jeong-Jin Kang
Edward J. Rothwell
Yang Hao
Sang-Hyun Lee

Advanced and Applied Convergence Letters AACL 22

Advanced and Applied Convergence & Adavanced Culture Technology

11th International Symposium, ISAAC 2023 in Conjunction with ICACT 2023, ICKAI 2023

November 16 - 18, 2023, Jeju, Korea
Revised Selected Papers

IIBC The Institute of Internet, Broadcasting and Communication
IPACT The International Promotion Agency of Culture Technology
지식의 숲
FOREST OF KNOWLEDGE

Extraction Practices on UML Sequence Diagram through Natural Language based Requirement Specifications

Hyuntae Kim*, Janghwan Kim**, Jihoon Kong***, R. Young Chul Kim****

Department of Software and Communications Engineering, Hongik University,*
*Department of Software and Communications Engineering, Hongik University **,*
*Toonsquare***,*
*Department of Software and Communications Engineering, Hongik University *****
{hyuntaekim,janghwan.kim**}@g.hongik.ac.kr, john.tooning@toonsquare.com***,*
*bob@hongik.ac.kr*****

Abstract

The current software code is becoming increasingly complex. One of the methods for managing software quality is requirements analysis and management. However, when specifying natural language-based requirements, it can be challenging to precisely define and extract meaning. In this study, we aim to address this issue using natural language analysis (Fillmore's semantic analysis) within the requirements analysis mechanism. We apply the C3Tree model from previous research to transform complex sentences into simpler ones. We conduct syntactic analysis of natural language sentences using the Stanford Parser to identify the most basic morphemes. Subsequently, we perform semantic analysis of the parsed sentences based on Fillmore's Case Grammar to extract their meanings. We aim to enhance the quality of software design by generating UML sequence diagrams based on natural language requirements analysis. Additionally, we plan to use these sequence diagrams as a basis for extracting interactions between objects within the Cartoon system in the future.

Keywords: *Fillmore's Case Grammar, Stanford Parser, Chomsky Syntactic Structures, Requirements Analysis, Sequence Diagram*

1. Introduction

The current software code is becoming increasingly complex and extensive. For effective software quality management, it is crucial to analyze and manage requirements, design, and implementation. However, there is a problem with not properly formalizing requirement specifications, often documented informally during software development[1]. Furthermore, if implementation is done without going through the design process, the quality of the implementation may also suffer.

There is a growing trend in using natural language analysis techniques for requirements analysis. In this study, we aim to address these issues by analyzing natural language requirements using tools such as Stanford Parser based on Fillmore's Case Grammar and Chomsky's Syntactic Structure, as well as the C3Tree model [2,3]. Subsequently, we intend to utilize the analyzed results to create sequence diagrams and implement solutions to the problems at hand.

2. Related Works

2.1 Previous Natural Language Analysis Study

Natural language analysis techniques used in the mechanism of this study include Fillmore's Case Grammar, Stanford Parser based on Chomsky linguistics, and the C3Tree model. Fillmore's Case Grammar is used to analyze the meaning of words in sentences, while the Stanford Parser conducts syntactic analysis of sentences. The C3Tree model simplifies complex compound or compound-complex sentences into simpler ones.

2.1.1 Fillmore's Case Grammar

Fillmore's Case Grammar is a grammar framework developed by Fillmore. It analyzes the semantic relationships of words within sentences, primarily focusing on verbs and their roles, such as subject and object. Table 1 summarizes Fillmore's Case Grammar, redefined in previous research [1].

Table 1. Redefined Fillmore's Case Grammar

Case	Definition
Actor	The instigator of the event/action.
Counter Actor	The force or resistance against which an action is carried out
Object	The entity which moves or changes or whose existence is in consideration
Theme	The subjective entity of Object
Result	Entity that comes into existence as a result of the action
Source	Origin of object
Instrument	Facility used in carrying out an event
Experiencer	Entity that receives or accepts or experiences or undergoes the effect of an action
Goal	Destination of object

Based on the Cases presented in Table 1, this study proceeds with the semantic analysis of sentences.

2.1.2 Stanford CoreNLP Parser

Stanford CoreNLP Parser is a natural language processing software developed by Stanford University. The Stanford Parser analyzes the grammatical structure of sentences, determining which words serve as the subject or object of a verb and more. The following image illustrates the analysis results of the example sentence, "I love you."

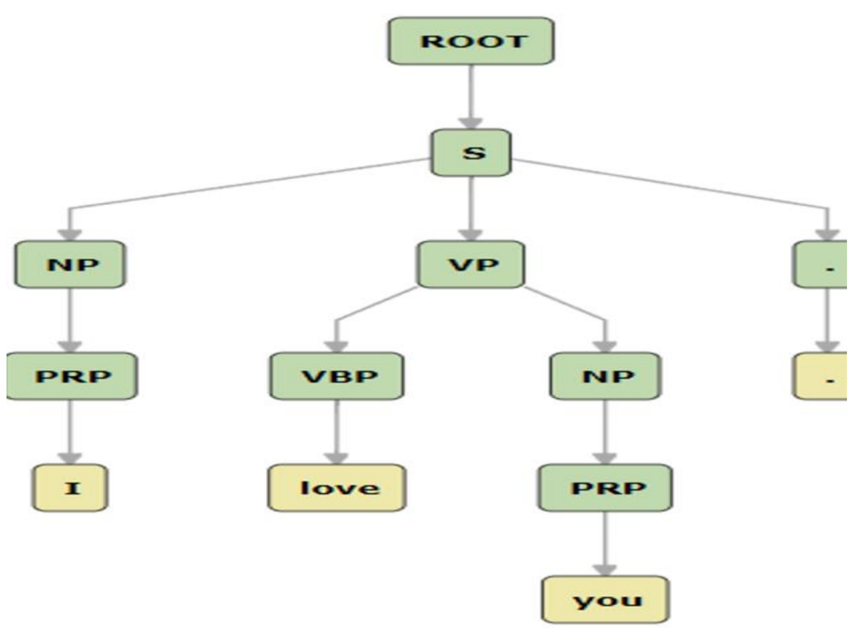

Figure 1. Analysis example using Stanford CoreNLP

"I" and "you" are classified as PRP, corresponding to nouns. In the given sentence, "love" is classified as VBP, indicating it's a verb and can be considered a "Message" in the sequence diagram. In this study, natural language analysis is conducted through syntax analysis, like the example described above.

3. Sequence Diagram Generation from Natural Language-based Requirements

3.1 Sequence Diagram Generation from Natural Language-based Requirements Process

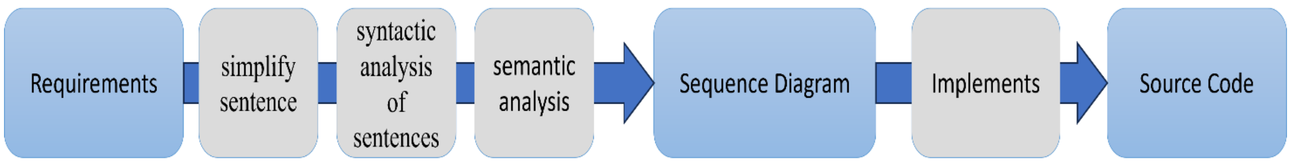

Figure 2. Mechanism Process Chart

Figure 2 is a diagram of the process configuration of the source code generation mechanism. It involves simplifying requirements into shorter sentences, conducting syntax analysis, and performing semantic analysis on the parsed sentence using Fillmore's Case Grammar. Subsequently, this process leads to the generation of sequence diagrams. Below, Figure 3 represents a crosswalk requirement.

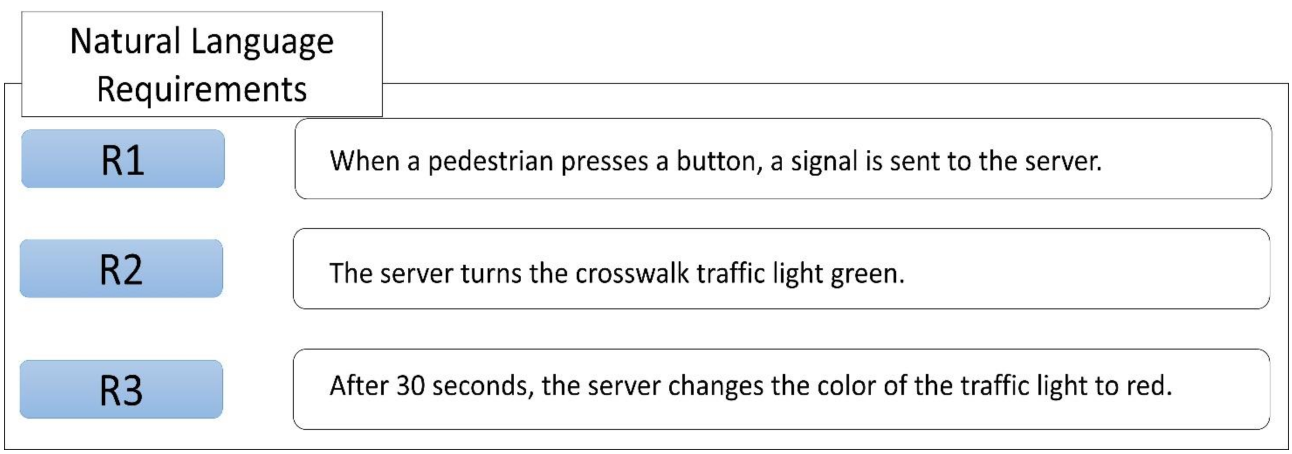

Figure 3. Natural Language Requirements

This requirement is analyzed using the process proposed in this study's mechanism, and based on the results, a sequence diagram is generated.

3.1.1 Syntax analysis using Stanford CoreNLP Parser

To create a sequence diagram, you need objects and messages. To extract these elements from natural language requirements, a natural language analysis process is required. Sentences consisting of double or complex sentences have the problem of making it difficult to analyze requirements. Therefore, the C3Tree model is used to simplify sentences into simpler forms.

The simplified sentences are subjected to syntax analysis using the Stanford Parser following Chomsky's approach. This analysis involves identifying messages in the sequence diagram and locating nouns. The following image provides an example of syntax analysis using the Stanford Parser.

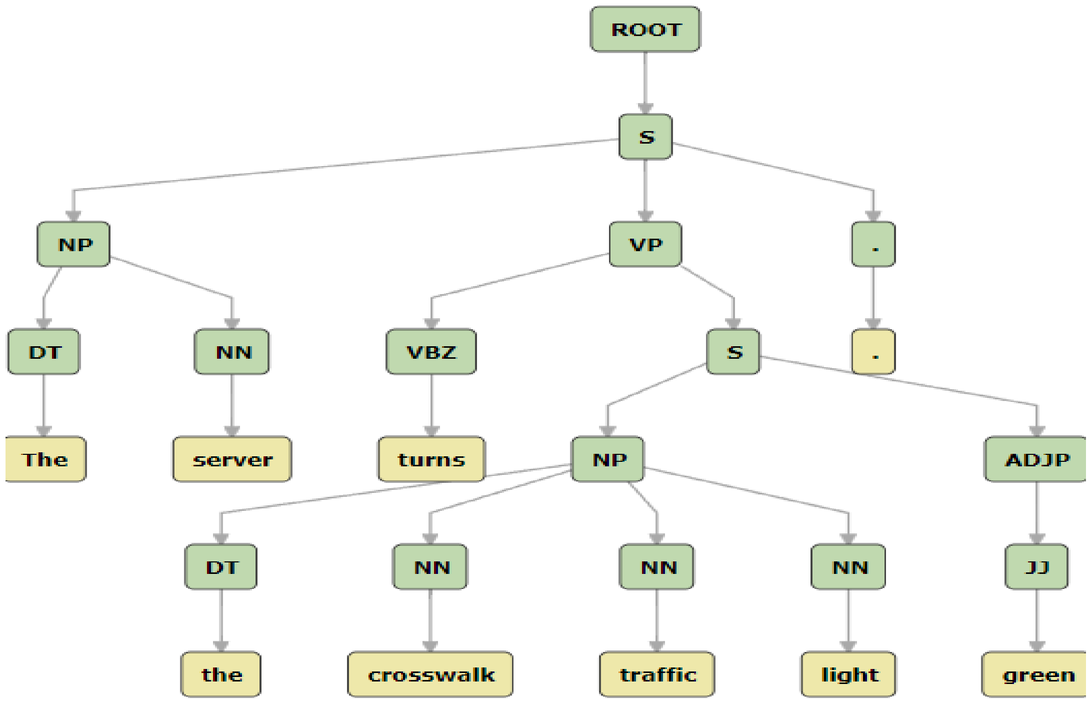

Figure 4. R2 Requirements Parsing Results

In Figure 4, the nouns corresponding to objects are "server" and "traffic light". The verb corresponding to the message is "turns". Therefore, in this natural language requirement, the words corresponding to the Message and objects are "turns" and "server," "traffic light," respectively.

Next, like the previous steps, we perform semantic analysis of the sentence that conducted syntax analysis using Fillmore's Case Grammar. Figure 5 is an example of the analysis of the R2 requirement sentence.

R2	The server(Actor) turns(Verb) the crosswalk traffic light(Theme) green(Result).

Figure 5. Results of Applying Case Grammar to R2 Requirements

In the R2 requirement, the entities used as objects are "server," serving as an Actor, and "traffic light," serving as a Theme. Similarly, using Fillmore's Case Grammar, we identify the necessary objects within the sentence for the sequence diagram. Therefore, in the analysis of the natural language requirement using Stanford Parser and Fillmore's Case Grammar, the objects are "server" and "traffic light," and the Message is "turn".

The image below represents the results of conducting the same analysis process as the R2 requirement, including implementing a sequence diagram and source code.

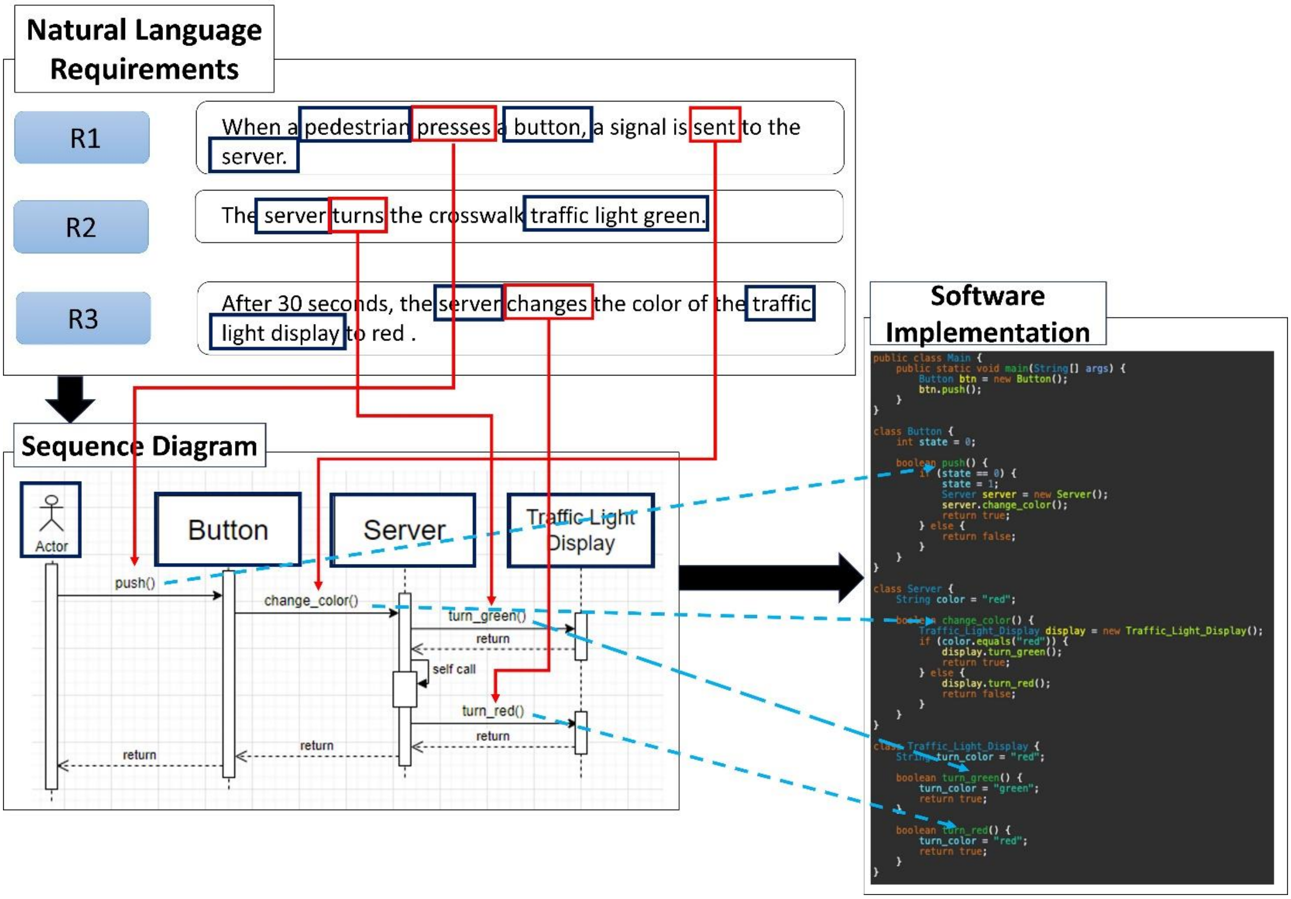

Figure 6. Sequence Diagram and Implementation results based on Requirements Analysis

4. Conclusion

The current problem lies in the informality and lack of standardization in requirements, and this paper addresses this issue by proposing a mechanism for generating sequence diagrams through requirements analysis. By using the mechanism proposed in this study, it first enables the analysis of unstructured requirements. Second, it facilitates the generation of precise sequence diagrams and code implementation based on the analyzed results, grounded in accurate requirements.

Through this, it is anticipated that in project development environments where requirements are not properly organized, high-quality code can be produced through accurate requirements analysis. In the future, we aim to implement the mechanism proposed in this study, aiming to contribute to the software market.

Acknowledgement

This research was supported by Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency(KOCCA) grant funded by the Ministry of Culture, Sports and Tourism(MCST) in 2023(Project Name: Development of AI-based user interactive multi-modal interactive storytelling 3D scene authoring technology, Project Number: RS-2023-00227917, Contribution Rate: 100%)

References

[1] B. K. Park, and R. Y. C. Kim, “Effort estimation approach through extracting use cases via informal requirement specifications,” Applied Sciences, vol.10, no.9, 2020
[2] Manning, Christopher D., Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 55-60.
[3] W. S. Jang and S. Y. Moon and R. Y. C. Kim, “Metamodeling Construction for Generating Test Case via Decision Table Based on Korean Requirement Specifications," KTSDE, vol.12, no. 9, pp.381-386, 2023.