Code Generations from Natural Language Specification with Naming Traceability

HONGIK UNIVERSITY
SOFTWARE ENGINEERING LAB
JANGHWAN KIM

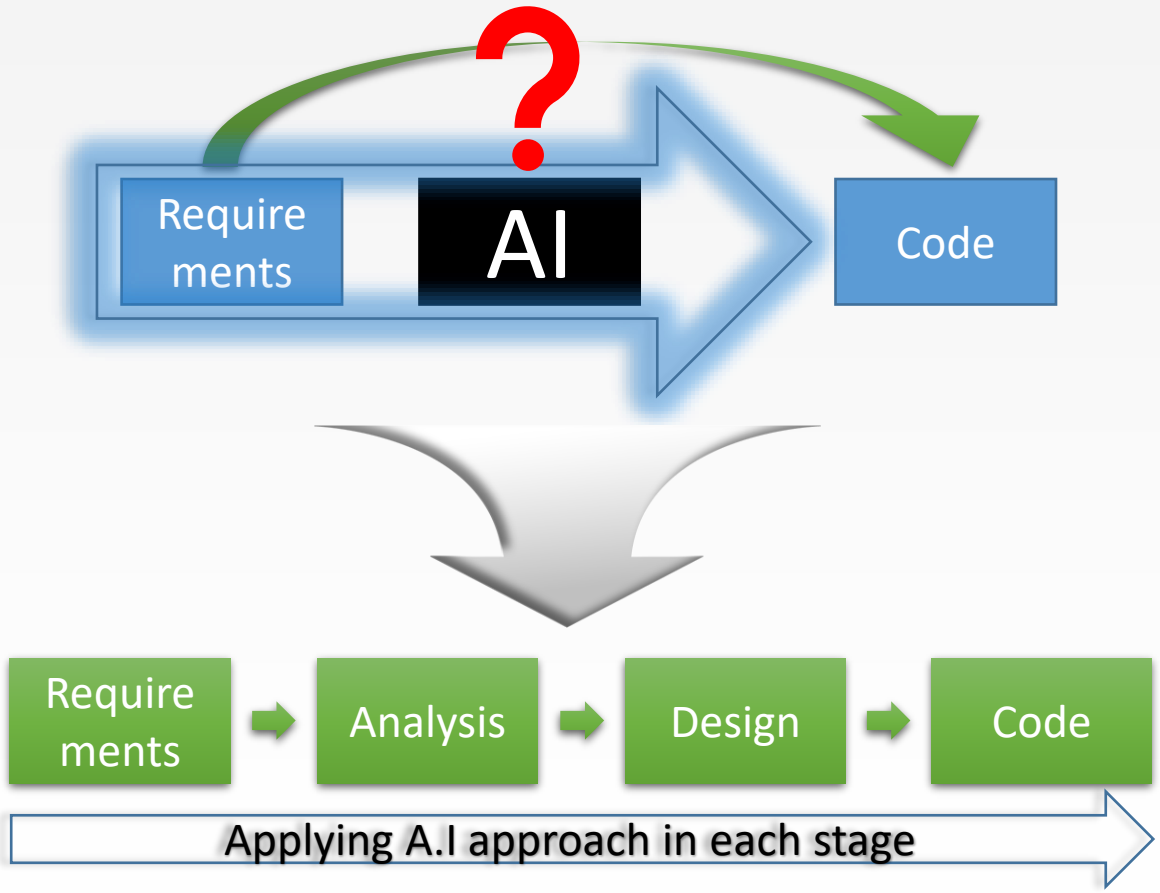
ADVISOR: Prof. ROBERT YOUNG CHUL KIM

CONTENTS

I. RESEARCH BACKGROUND

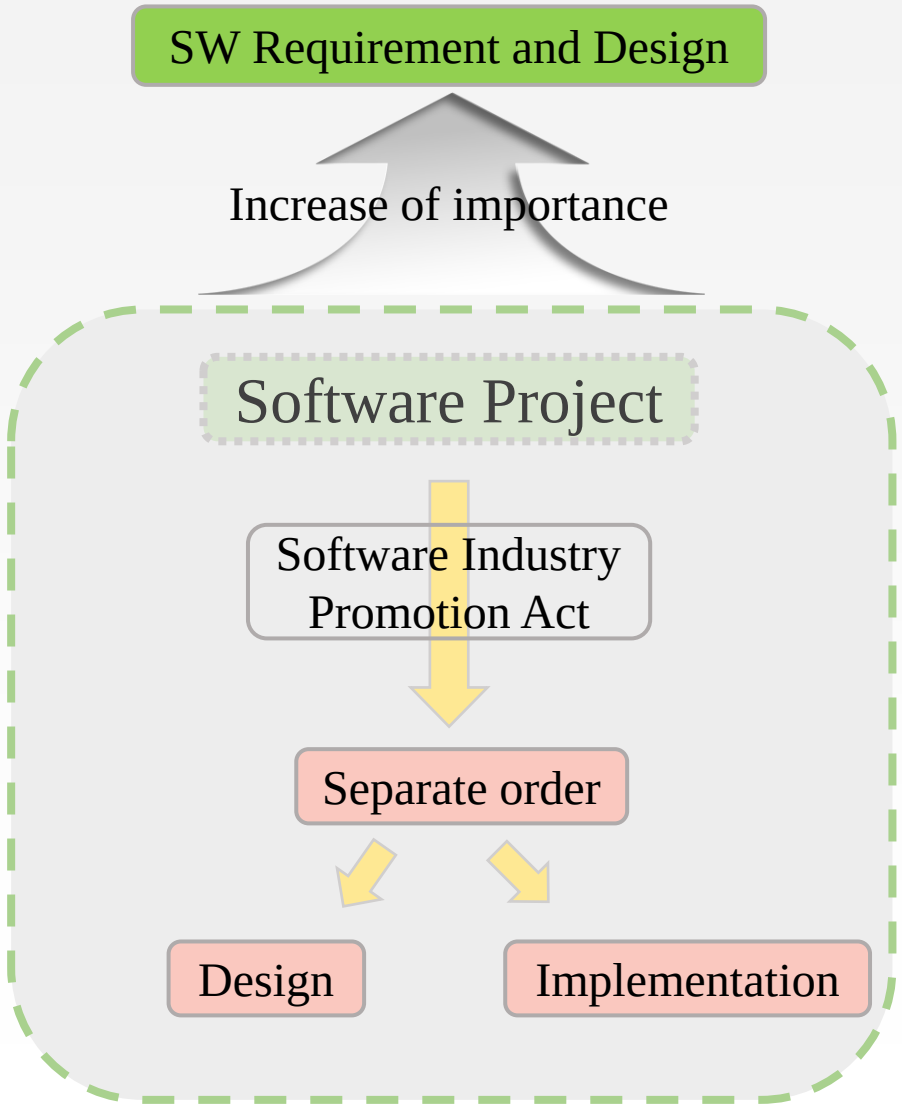
I. RESEARCH BACKGROUND – Issue 1

How to automatically generate code with AI?



- **SE for AI** vs **AI for SE**
- Recently, there are many researches to generate codes through AI techniques from requirements.
- Most researches try to generate code directly from requirements. However, these approaches can't explain how nor what is going on the intermediate steps between the requirements and generating code.
- So, we think that these types of approaches need a step-by-step process to generate code from requirements.
- This paper started from a research to identify each steps of development process to improve these problems.

I. RESEARCH BACKGROUND – Issue 2



- Previously, when a SW development project got contracted, one company takes full charge of the project through entire SW lifecycle, including SW requirements analysis, SW design, SW implementation and SW testing.
- Recently, Software Industry Promotion Act was passed.
- This is the law that makes the software project a separate order between SW design and Implementation.
- Therefore, the importance of the SW Requirement and Design increases immediately.

I. RESEARCH BACKGROUND – Issue 3

➢ Issues of SW Requirement

1. The software project **scale** continues to **grow**. → The complexity of SW requirement **increases.**
   → SW design complexity also **increases**. → SW Code complexity also **increases.**
2. Most of the requirement specification are using between **informal method** and **formal method**
3. Requirements **are constantly changed** according to **custom needs**
4. Natural language-based Requirements includes **vagueness.**

- Custom needs is still **unclear.**
- We focus on Requirements based on **Natural Language**.
- It requires to research about a process of **each steps** how to generate code from Requirements.

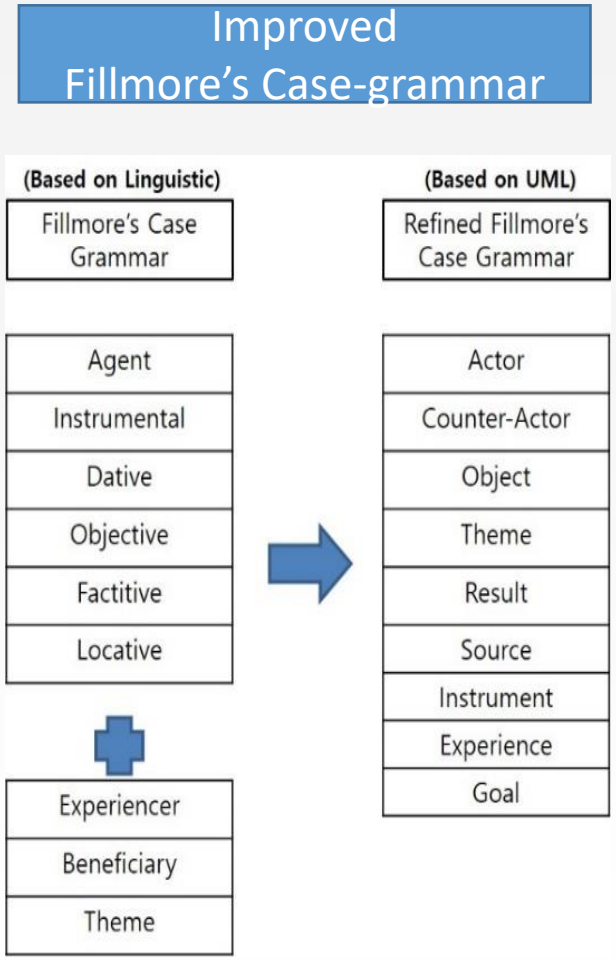
II. RELATED WORKS

Fillmore's Semantic Analysis

- Fillmore's **semantic analysis** starts from his case grammar approach.
- Fillmore defined the **roles of main verb-based nouns** in his paper.

THEME	MEANING
Agent	a subject that is perceived to cause an action represented by a verb.
Instrumental	an object that becomes the cause of an action or a state which a verb represents.
Dative	a person or an animal affected by a state, or an action represented by a verb.
Objective	an object that is affected by an action or a state which a verb expresses.
Factitive	a person or an animal that exists as a result of an action and a condition of a verb.
Locative	a state that a verb represents or where an action occurs.

II. RELATED WORKS



- In Park's Study, Fillmore's Case grammar was improved to fit the Requirement Engineering approach, especially for Unified Modeling Language(UML)
- In his paper, Agent, the subject of the sentence, is improved into Actor and Count-actor interacting with it.
- Also, in addition to the 6 cases that Fillmore initially claimed, more than 20 cases that were continuously developed were developed. He improved those cases into requirements engineering to facilitate the high-level design through UML from the requirements.

Abbott's Textual Analysis

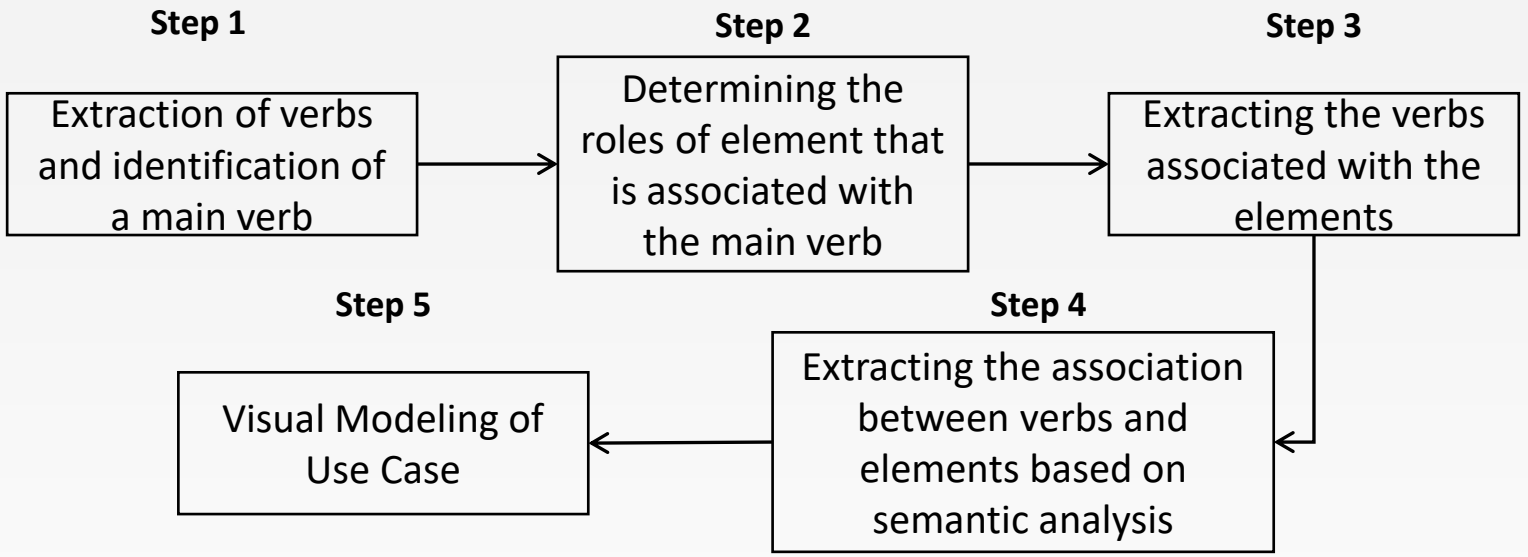
Part of speech	Model component
Proper noun	Object
Improper noun	Class
Doing verb	Operation
Being verb	Inheritance
Having verb	Aggregation
Modal verb	Constraints
Adjective	Attribute

- Abbott's textual analysis approach is suitable method for identifying key acting of the requirement.
- His approach works with modeling from language components such as part of speech to model components.
- Figure on the left shows the mapping Part of speech to component of the high-level design model for requirement analysis.

- We will describe more detail in the later presentation

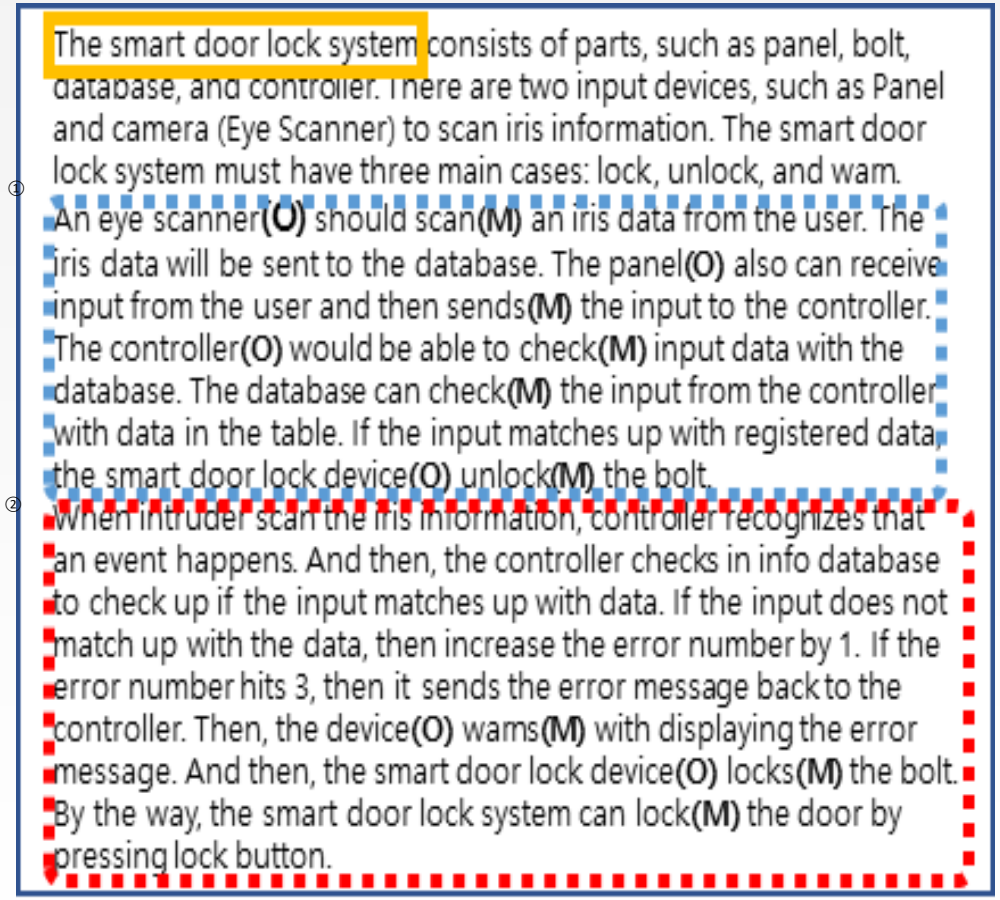
III. Code Generations from Natural Language Specification with Naming Traceability

III. Steps of Code Generations from Natural Language Requirement



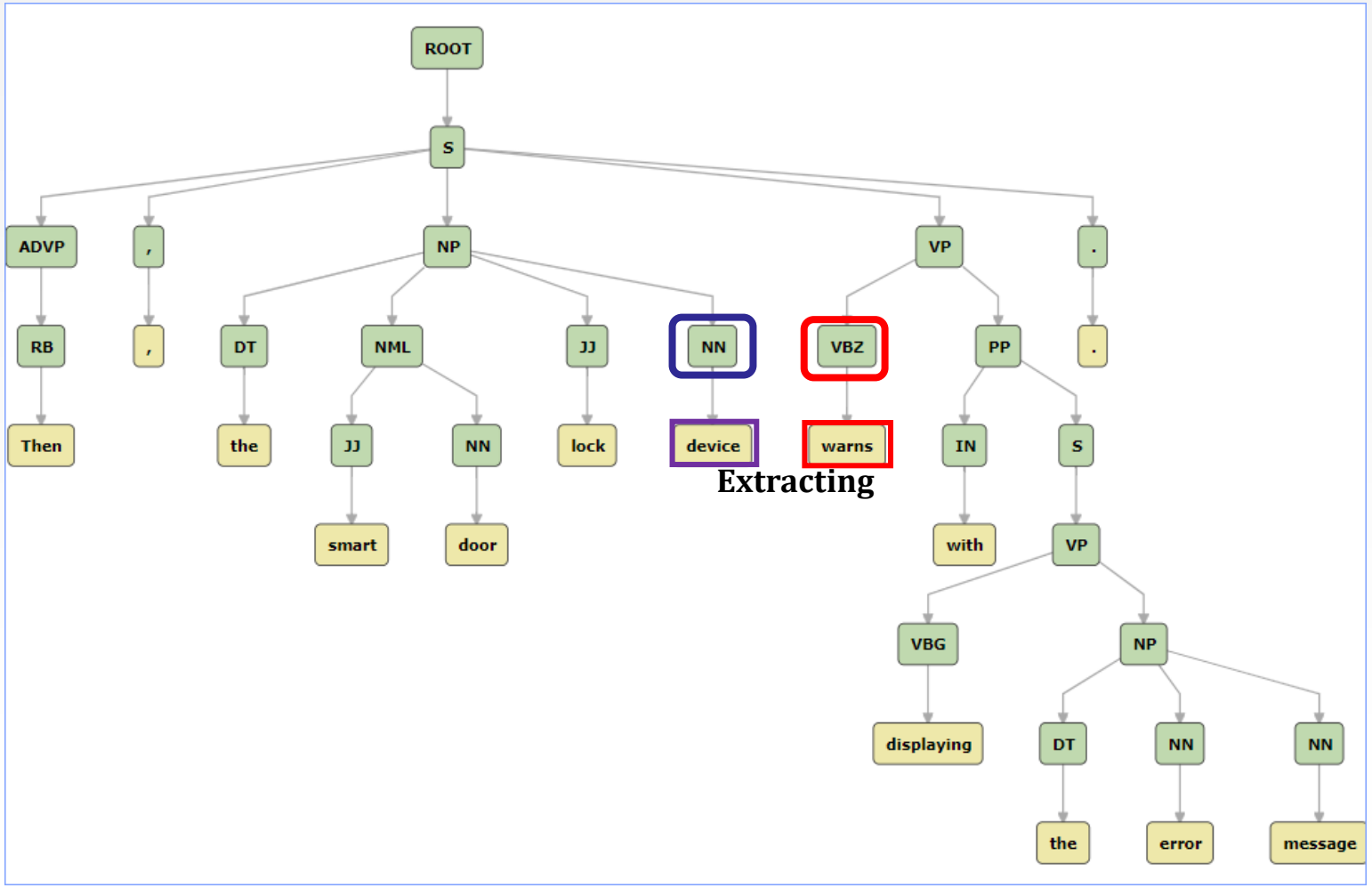
- Step 1 is to extract verb from the requirement and identify main verb
- Step 2 is to determine a role of elements that is associated with the main verb.
- Step 3 is to extract other verbs that are associated with elements.
- Step 4 is to mark up an association between those verbs and other elements
- Step 5 is to visualize use case

III. Code Generations from Natural Language Requirement



- We applied our approach to Smart Door lock System example.
- First, we mark up by functioning of the example.
- Then, we applied Abbott's textual analysis for the requirements by the sentences after dividing parts by the category.

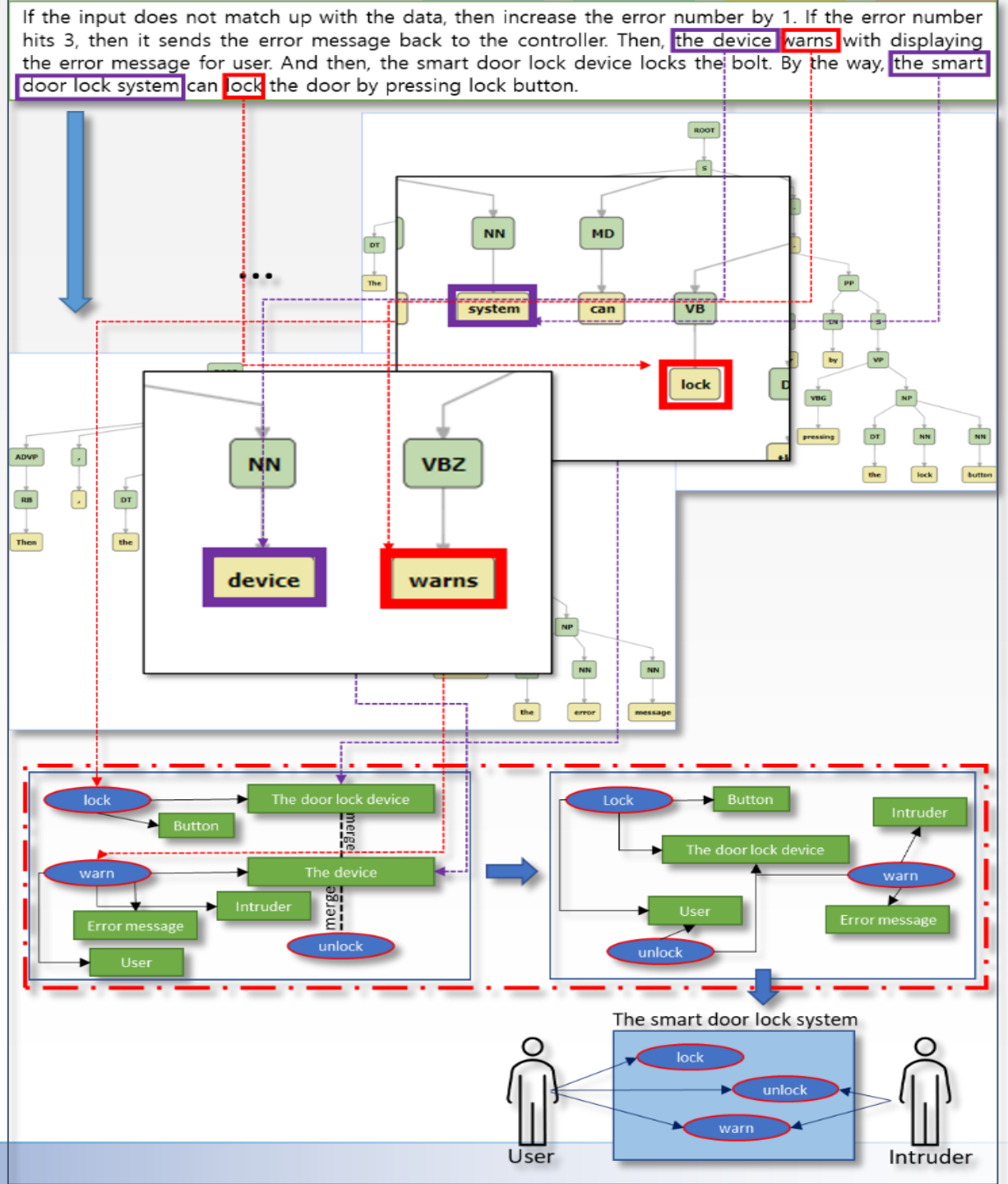
III. Structure Analysis of Natural Language Requirement



- We use Stanford parser to parser the requirements
- This parser structurally breaks the sentence into word units.
- VBZ is the main verb of the requirement. Then we find the ‘NN’ and extract the subject of the verb that actually performs the main verb.

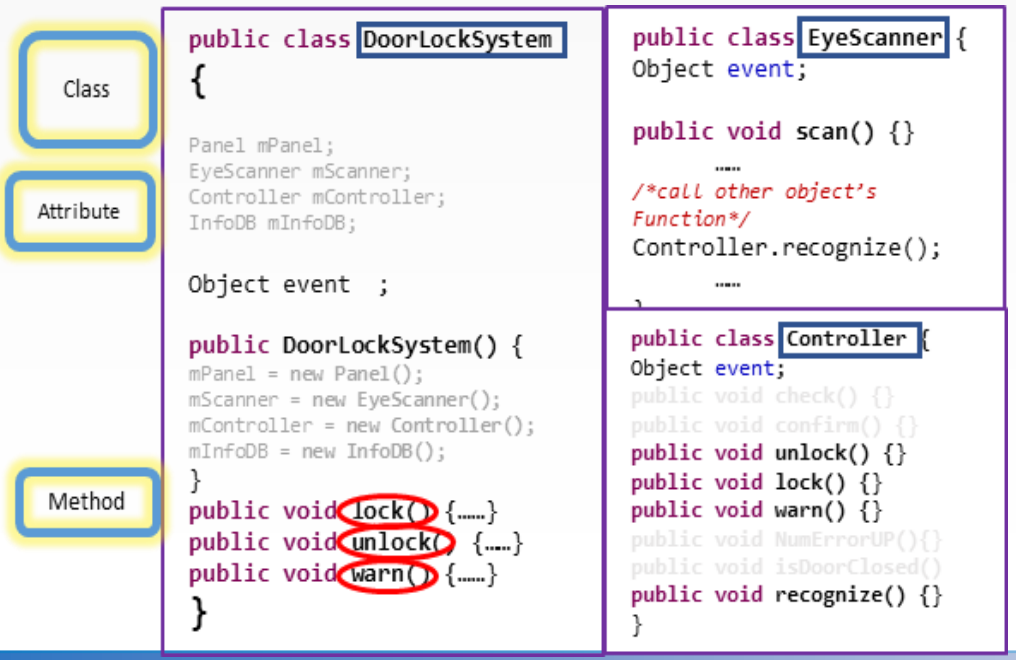
III. Generating Use case

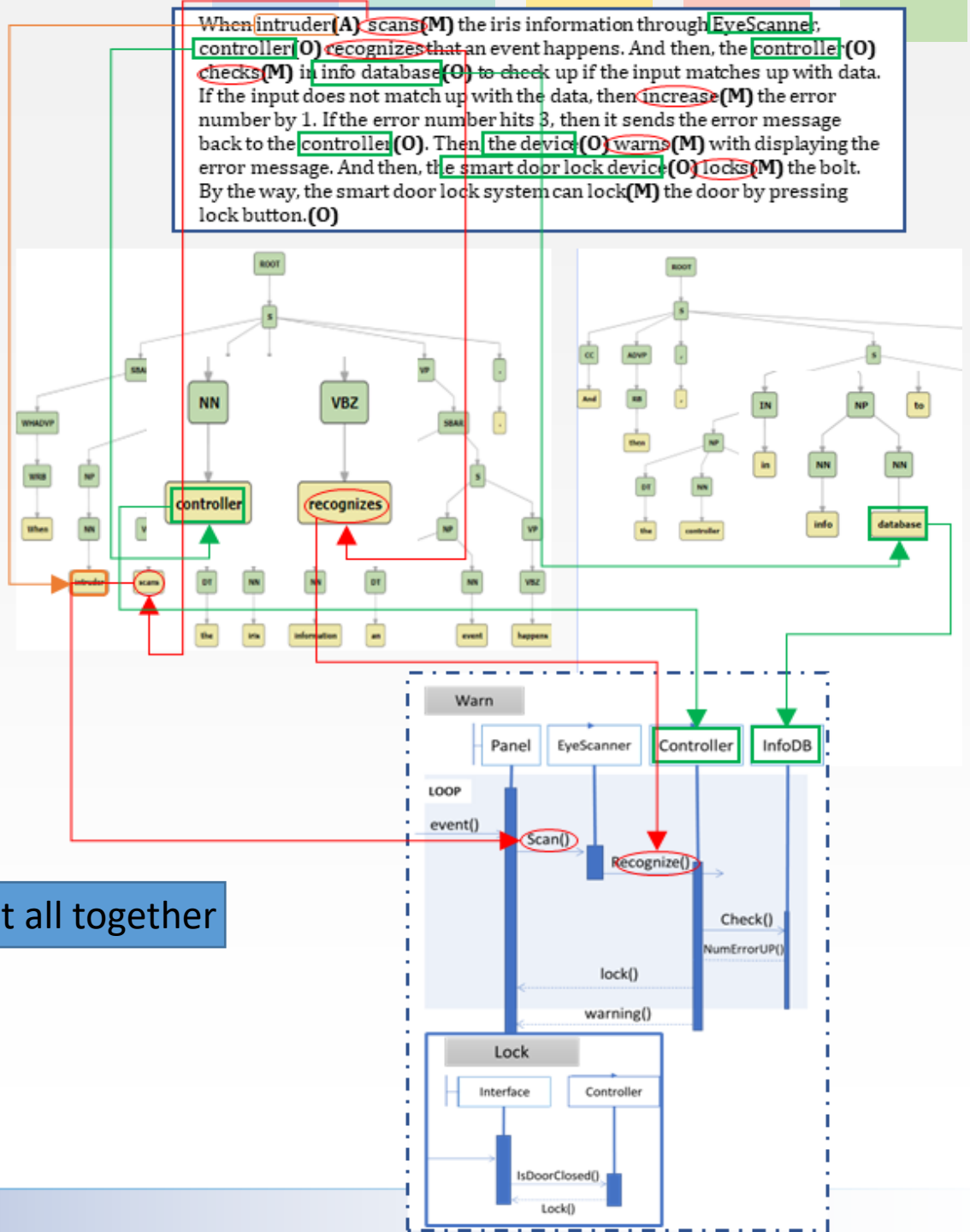
- We analyzed the structure of each requirement sentence by parsing it as we mentioned above.
- Based on the analyzed results, NN, VBZ, and other elements were found in each requirement sentence and presented in diagram form as shown at the bottom of the figure on the right.
- Looking at these diagrams, we can find that the same names exist in the form of nouns in the diagrams.
- The result of mapping the same nouns together is shown in the form of a use case diagram.



III. Generating Sequence Diagram

- A sequence diagram is created through Abbott's textual analysis results and semantic analysis results.
- Green box will be the objects and then, verbs becomes sequences.
- We generate code guideline from Sequence diagram and Use case diagram.

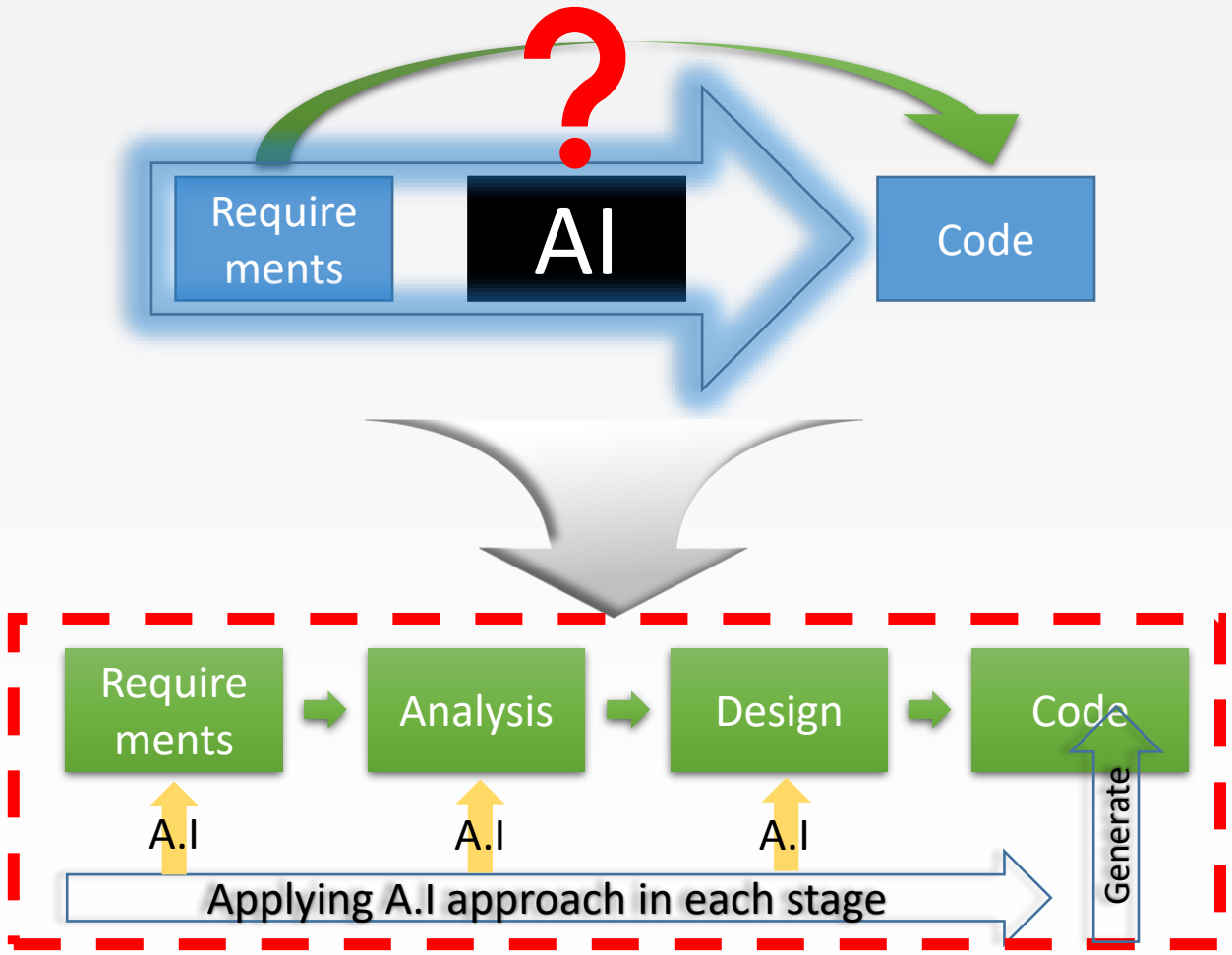




Put all together

VI. CLUSION

IV. Conclusion



- This paper describes each steps starting from natural language requirements and provides **guidelines** for generating codes.
- This method go through the process from natural language-based requirements so that it can be source that can be applied by AI techniques.
- In the future, we plan to apply AI techniques to the results of each steps in the process.