

# 자연어 요구사항으로부터 UML 시퀀스 모델을 경유한 3D 객체 추출 메커니즘

김현대<sup>1</sup>, 김장환<sup>2</sup>, 김영철<sup>3</sup>

<sup>1</sup>홍익대학교 일반대학원 소프트웨어공학 연구실 석사과정

<sup>2</sup>홍익대학교 일반대학원 소프트웨어공학 연구실 박사수료

<sup>3</sup>홍익대학교 소프트웨어융합학과 교수

<sup>1</sup>hyuntaekim@g.hongik.ac.kr, <sup>2</sup>lentoconstante, <sup>3</sup>bob}@hongik.ac.kr

## 3D Object Extraction Mechanism via UML Sequence Models from Natural Language Requirements

Hyuntae Kim<sup>1</sup>, Janghwan Kim<sup>2</sup>, R. Young Chul Kim<sup>3</sup>

<sup>1,2,3</sup>Dept. of Software and Communications Engineering, SELab, Hongik University

### 요 약

현재 다양한 분야에서 AI가 사용되고 있다. 최근에는 소프트웨어공학 관점에서 요구 사항 분석에 ChatGPT와 같은 LLM 모델을 적용하고 있다. 하지만 1) 대부분의 생성형 AI는 불투명한 공정을 통해 3D 이미지가 생성하고, 3D 이미지를 생성할 때마다 다른 이미지를 생성한다. 이에 따라 동일한 인물이나 사물을 사용하고 싶은 사용자들은 동일한 객체가 들어간 그림을 일관성 있게 생성할 수 없다. 2) 또한 LLM과 이미지 생성 AI와의 결합이 시도되고 있지만 문장 의미 분석 성능이 부족하다. 이를 해결하기 위해, 자연어 요구사항을 언어학적 기법을 통해 분석하고, 분석 결과를 기반으로 UML 시퀀스 다이어그램 및 3D 객체 생성 메커니즘을 제안한다. 즉 언어학적 분석 기법을 통해, 요구사항의 정확한 의미와 속성을 추출한다. 그런 다음 추출된 정보를 시퀀스 다이어그램과 매핑하여 3D 객체 이미지를 생성한다. 제안하는 방법을 통해 3D 객체 생성의 소프트웨어 개발 공정 사용으로 생산성을 높여 시간과 비용을 단축할 수 있을 것으로 기대한다.

### 1. 서론

최근, LLM을 통해 이미지를 생성하는 AI 기술 성능이 발전 중이다. Chat GPT와 Dall-E 같은 생성형 도구들은 프롬프트에 생성하고 싶은 그림의 특징을 자연어로 입력하면 그림이 생성된다. 하지만 Dall-E는 3D 이미지를 생성하지 못하고 오직 2D 이미지만 생성 가능하다. 3D 이미지를 생성하기 위해선 DreamFusion과 같은 AI를 이용해야 한다. DreamFusion을 이용하면 텍스트나 2D 이미지를 이용해 3D 이미지 생성이 가능하다[1].

하지만 이러한 AI는 소프트웨어 공학적 관점에서 볼 때 어떤 과정을 통해 이미지 생성이 진행되는지 알 수 없다는 문제점이 있다. 또한 프롬프트를 통해 입력 받은 자연어의 분석이 잘못되어 사용자의 의도와 다르게 해석되는 문제점이 존재한다. 이러한 문제점을 해결하기 위해 본 연구에서 제안하는 메커니즘

에서는 소프트웨어 공학의 요구공학과 언어학을 접목하여 자연어 분석을 한다. 또한 분석된 자연어를 제안하는 공정을 통해 시퀀스 다이어그램 및 3D 이미지를 생성한다.

2장 관련 연구에서는 본 연구에서 사용한 Chomsky와 Fillmore의 언어학적 관련 연구를 서술한다. 3장 UML 시퀀스 다이어그램 및 3D 오브젝트 생성에서는 자연어의 언어학적 분석 방법 및 다이어그램 생성 과정, 3D 오브젝트 이미지 생성에 관해 서술한다.

### 2. 관련 연구

#### 2.1. Chomsky's Syntactic Analysis

노엄 촘스키의 구조문법 이론은 언어학을 통한 텍스트 분석에 필수적인 이론이다. 이 이론은 모든 인간 언어가 공통적인 문법적 틀을 공유한다고 가정하며, 자연어 문장을 구조적으로 분석함으로써 다양한 단위(형태소, 품사, 구, 절 등)의 분석방법을 제안한다.

이러한 방법은 깊은 구조(Deep Structure)와 표면 구조(Surface Structure)로 이루어진 이중 구조를 통해, 문장의 의미적 기반과 발화의 형태 사이의 변환을 설명한다. 변형 규칙은 깊은 구조를 어떻게 표면 구조로 재배열하는지를 결정하며, 이는 문장 생성에 필수적인 요소로 작용한다. 본 논문에서는 촘스키의 이론을 통해 언어의 구조적 이해를 분류하고 요소들을 매핑하는데 사용한다. 촘스키의 이론을 적용하기 위해 Stanford Neural Parser 를 사용한다[2].

### 2.2. Fillmore's Case Grammar

찰스 필모어의 케이스 문법 이론은 언어학, 특히 구문론과 의미론 분야에 기여한 이론이다[3]. 이 방법은 문장 구성 요소 각각이 수행하는 역할을 케이스로 분류하여, 언어의 의미 구조를 체계적으로 해석한다. 주요 케이스로는 실행자(Agent), 수혜자(Patient), 도구(Instrument), 원인(Cause) 등이 있으며, 이들은 동사와 관련된 명사 또는 명사구의 기능을 명확하게 설명한다. 필모어는 문장의 의미가 개별 단어의 조합을 넘어서, 단어들이 수행하는 특정한 역할에 의해 형성된다고 강조한다. 이러한 분석 방법은 최근 대두되는 의미론적인(Semantic) 자연어 처리(Natural Language Processing) 기술 발전에 중요한 역할을 한다. 자연어 처리는 인공 지능 분야에서도 의미 파악과 문장 처리의 정확성을 높이는 데 활용된다. 본 연구는 이를 바탕으로 필모어의 케이스 문법을 활용하여 텍스트의 구조적 및 의미론적 분석을 수행한다.

## 3. UML 시퀀스 다이어그램 및 3D 오브젝트 생성

### 3.1. 자연어 문장 전처리

자연어 문장을 처리하기 위해서는 불필요한 정보를 제거하거나 필요한 정보를 추가하는 전처리 과정이 필요하다. 사용자가 자연어 요구사항 문장을 입력하면 문장 분석을 위해 전처리 해야 한다. 문장 전처리 과정은 4 단계로 구성되어 있다. 첫번째 단계는 주요 절 식별이다. 완전한 문장으로 단독으로 있을 수 있는 문장인 주절을 식별한다. 두번째 단계는 종속절 및 구 식별이다. 종속절과 구는 문장의 종속적인 부분으로, 독립적으로 있을 수 없고, 주절에 대한 추가적인 정보를 제공한다. 종속절(구)이 필요 없는 경우 제거한다. 세번째 단계에서는 절과 구를 분리한다. 각 절과 구를 독립적인 문장으로 바꾸어 절과 구가 독립적인 의미가 있는지 확인한다. 의미가 없는 절과 구는 제거한다. 마지막으로 네번째 단계는 문장 조정 단계이다. 각 문장이 완전하고 분리 후에도 의미가 있도록 일부 단어를 추가, 삭제 또는 변경한다.

### 3.2. 형태소 분석 및 명사의 의미론적 역할 식별

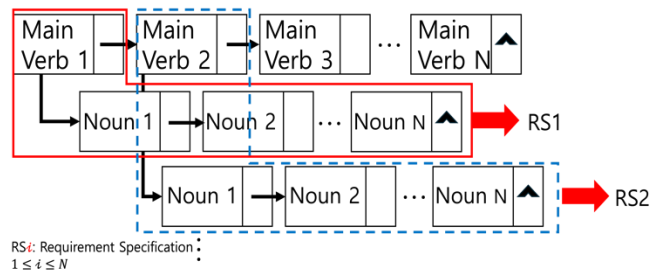
전처리가 완료된 문장은 Chomsky 의 언어학적 기법이 적용된 과서를 이용해 형태소 분석을 한다. 형태소 분석 결과에서 시퀀스 다이어그램의 객체와 파라미터로 사용할 명사와 메시지로 사용할 동사의 후보를 찾는다.

명사로 식별된 단어들을 필모어의 언어학적 기법을 적용해 의미론적 역할을 식별한다. 본 연구에서는 시퀀스 다이어그램과 이미지 생성에 적합하게 재정의한다. 아래 표 1 은 재정의한 Case Grammar 이다.

<표 1> 재정의한 Case Grammar

Case Grammar		정의
Actor		사건이나 행동을 촉발하는 사람이나 사물
Source		메시지를 보내는 객체
Target		메시지를 받는 객체
Dynamic Verb	Transitive Verb	하나 이상의 목적어를 갖는 동사
	Active Verb	문장의 주어가 특정한 행위를 하고 있는 주체
Instrument		동사의 동작에 관여하거나 이벤트를 실행하는 데 사용되는 객체
Lifeline		객체가 생성되고 소멸하는 시점

다이어그램을 생성하기 위해 필모어의 의미론적 역할이 적용된 정보를 추출하기 위해 다중 연결 리스트에 저장한다[4]. 아래 그림 1 은 의미론적 역할을 저장하는 다중 연결 리스트의 구조이다.



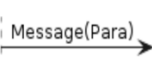



(그림 1) 다중 연결 리스트 구조

다중 연결 리스트의 구조는 각 문장의 메인 동사들의 연결을 우선적으로 한다. 동사의 리스트 연결 후 동사가 존재하는 문장 내 명사들의 리스트를 문장 단위로 연결한다.

3.3. 필모어 Case Grammar 와 시퀀스 모델 요소 매핑  
시퀀스 다이어그램을 생성하기 위해서는 필모어의 의미론적 역할이 식별된 명사들과 동사를 사용해야한다. 아래 표 2 는 필모어의 Case Grammar 의 역할과 다이어그램의 요소를 매핑한 결과다.

<표 2> 언어학적 요소와 UML 다이어그램 요소 매핑

Case Grammar	다이어그램 요소	Notation
Actor	액터	
Source	객체	
Target		
Dynamic Verb	메시지	
Transitive Verb / Active Verb	파라미터	
Instrument	라이프라인	

시퀀스 다이어그램에서의 액터는 외부 시스템과 상호작용하는 객체를 의미한다. Case Grammar 에서 액터는 사건이나 행동을 촉발하는 사람이나 사물이다. 따라서 다이어그램에서의 액터와 Case Grammar 에서의 액터를 매핑한다. Case Grammar 의 Source 는 메시지를 보내는 객체를 의미하고 Target 은 메시지를 받는 객체를 의미한다. 따라서 Source 와 Target 을 다이어그램의 객체와 매핑한다. 동사의 경우 다이어그램의 메시지에 해당하고 Instrument 는 동작에 관여하는 객체이기 때문에 메시지의 파라미터와 매핑된다.

PlantUML 을 사용하면 스크립트를 통한 다이어그램 생성이 가능하다[5]. 시퀀스 다이어그램을 생성하기 위한 기본 문법은 ‘메시지를 보내는 객체 -> 메시지를 받는 객체: 메시지(파라미터)’이다. 따라서 재정의한 Case Grammar 를 이용해 스크립트를 작성하면 ‘Source -> Target: Verb(Instrument)’가 된다.

3.4. 3D Object 를 위한 UML 핵심 속성 JSON 구조화  
 생성한 다이어그램의 핵심 속성을 추출하여 3D 오브젝트 생성에 필요한 JSON 파일을 생성한다. 다이어그램의 객체를 3D 객체로 정의하고 다이어그램의 행위 속성에 맞는 이미지 에셋을 선택한다. 아래 그림 2 는 객체의 속성을 정의한 JSON 파일의 일부이다.

```

"object": {
  "uuid": "000002",
  "src": "Richard.fbx",
  "type": " FBX ",
  "name": "Richard",
  "position": { "x": 3, "y": 3, "z": 0 },
  "rotation": { "x": 0, "y": Math.PI / 4, "z": 0 },
  "scale": { "x": 1, "y": 1, "z": 1 },
}
    
```

(그림 2) 다이어그램 핵심 요소로부터 JSON 파일 생성

3.5. 3D Object 이미지 생성

움직일 수 있는 3D 오브젝트 이미지를 생성하기 위한 에셋은 리깅된 데이터를 사용해야한다. 리깅된 데이터를 사용하면 오브젝트가 자세를 취하도록 할 수

있다. 3D 오브젝트 이미지 생성을 하기 위해서는 다이어그램의 속성을 이용해야 한다. 객체의 속성과 행동 정보가 담긴 JSON 파일을 작성한다. 작성한 JSON 파일을 Three.js 를 통하면 이미지를 생성할 수 있다[6].

4. 적용 사례

본 논문에서 제안하는 메커니즘을 적용해보기 위해 아래와 같은 자연어 요구사항을 사용한다.

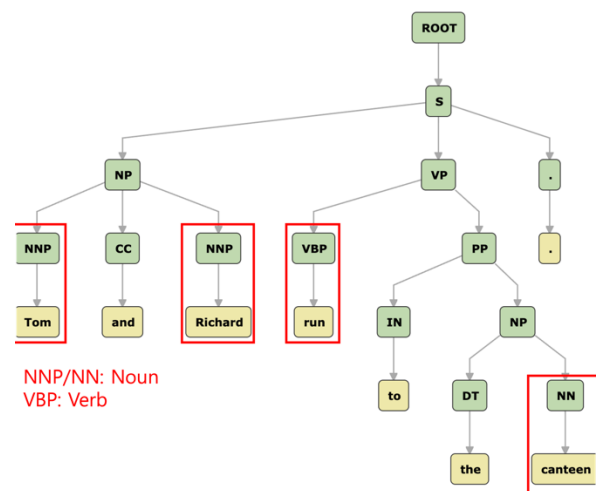
*During the break, Tom went to see Richard and told him to go to the canteen. Tom and Richard run to the canteen and look at the snacks on the shelves. Richard picks up the dumplings and pays the shopkeeper. Tom and Richard eat dumplings.*

문장을 분석하기 위해 먼저 아래 표 3 과 같이 전처리 과정을 거친다.

<표 3> 문장 전처리 결과

	전처리 전	전처리 후
R1	During the break, Tom went to see Richard and told him to go to the canteen.	Tom went to see Richard. Tom told Richard to go to the canteen.
R2	Tom and Richard run to the canteen and look at the snacks on the shelves.	Tom and Richard run to the canteen. Tome and Richard look at the snacks on the shelves.
R3	Richard picks up the dumplings and pays the shopkeeper.	Richard picks up the dumplings. Richard pays the shopkeeper.
R4	Tom and Richard eat dumplings.	Tom and Richard eat dumplings.

다음으로 Stanford Parser 를 사용해 전처리한 문장의 형태소 분석을 진행한다. 다음 그림 3 은 R2-1 문장의 형태소 분석 결과다.



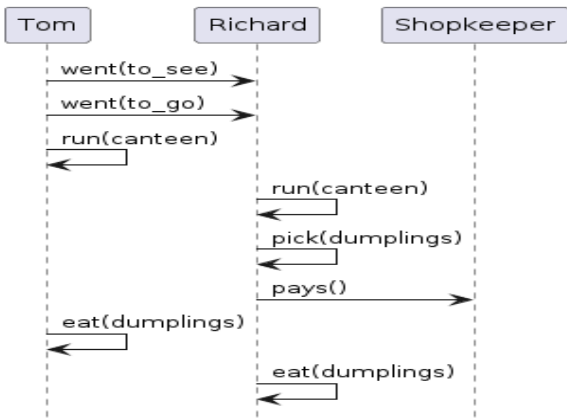
(그림 3) 형태소 분석 결과

분석 결과에서 객체로 분류하는 명사는 Tom, Richard, canteen 이고 동사는 run 이다. 형태소 분석 결과에 필모어의 의미론적 역할을 적용하면 Source 는 Tom, Richard 이고 Target 은 Source 와 동일하다. 동사의 run 은 active verb 이다. 아래 그림 4 는 필모어의 의미론적 역할을 기반으로 저장한 다중 연결 리스트이다.

	복문	중문
R1	During the break, Tom went to see Richard and told him to go to the canteen.	Tom went to see Richard. Tom told Richard to go to the canteen.
Main Verb	went → told → run ...	
Noun	Tom → Richard	RS1-1
Noun	RS1-2 ← Tom → Richard → canteen	
<b>Fillmore 분석 결과</b>		
RS1-1	Tom[Source] went[Main Verb] to see Richard[Target].	
RS1-1	Tom[Source] told[Main Verb] Richard[Target] to go to the canteen[Instrument].	

(그림 4) 다중 연결 리스트 저장 결과

다이어그램을 생성하기 위해 다중 연결리스트에 저장된 정보를 이용한다. 아래 그림 5는 다중 연결 리스트에 저장된 필모어 분석 결과를 바탕으로 생성한 다이어그램이다.



(그림 5) 시퀀스 다이어그램 생성 결과

생성한 다이어그램의 속성을 JSON 파일에 정의하고 Three.js 를 사용하여 3D 이미지를 생성한다. 그림 6은 R2-1 문장을 이미지로 생성한 결과다.



(그림 6) 생성된 3D 이미지

### 5. 결론

본 연구에서는 언어학적 분석 기법을 적용하여 Sequence Diagram 을 생성하고, Three.js 기술을 활용하여 3D 객체를 생성하는 메커니즘을 제안한다. 복잡한 소프트웨어 요구 사항을 구조적이고 의미론적으로 분석하여 문장에서 식별된 객체들의 행위와 메시지를 시각적으로 명확하게 표현한다. 이 메커니즘은 3D 모델 생성 과정을 효율적으로 만들어 개발 시간을 줄이고, 기존 AI 도구들과 비교할 때 언어학적(구조적, 의미적) 분석을 통해 보다 정확한 의미 추출로 시퀀스 다이어그램을 생성함으로써, 3D 객체를 통한 소프트웨어의 개발시간을 단축할 수 있기를 기대한다.

현재 메커니즘은 다양한 언어와 전문 용어 처리에 있어 제약이 있다. 향후 연구에서는 언어학적 분석방법을 다양화하고, 다양한 도메인의 자연어 처리를 통해 메커니즘의 적용 범위를 확대를 기대한다.

### ACKNOWLEDGMENT

본 연구는 2023/2024 년도 문화체육관광부의 재원으로 한국콘텐츠진흥원(과제명: 인공지능 기반 사용자 대화형 멀티모달인터랙티브스토리텔링 3D 장면 저작 기술 개발, 과제번호: RS-2023- 00227917,기여율:50%) 지원과 2023/2024 년도 정부(교육부)의 재원으로 한국연구재단기초연구사업(과제명: NLP BERT Model 기반 자동 리팩토링을 통한 무결점 코드화 연구, 과제번호: No.2021R1I1A3050407,기여율:50%)의 지원을 받아 수행된 연구임.

### 참고문헌

- [1] Poole, Ben, et al, "Dreamfusion: Text-to-3d using 2d diffusion," arXiv preprint arXiv:2209.14988, 2022.
- [2] Manning, Christopher D., Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky, The Stanford CoreNLP Natural Language Processing Toolkit In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, 2014, pp. 55-60.
- [3] Fillmore, Charles J, "The case for case," 1967
- [4] 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
- [5] PlantUML, [Internet], <https://plantuml.com/ko/>
- [6] Three.JS, [Internet], <https://threejs.org/>