

# 비정형 요구사항으로부터 생성된 원인-결과 그래프의 검증을 위한 C3Tree

## 모델 설계

장우성\*, 박보경\*, 김영철\*

\*홍익대학교 소프트웨어공학연구소실

\*{jang, park, bob}@selab.hongik.ac.kr

## Design of C3Tree Model for Verification of Cause-Effect Graph Generated from Natural Language Oriented Informal Requirement Specifications

Woo Sung Jang\*, Bo Kyung Park\*, R.Young Chul Kim\*

\*Software Engineering Lab., Hongik University

### 요 약

올바른 요구사항 기반 테스트케이스는 소프트웨어의 품질을 증가시키고, 유지보수 비용 절감이 가능하다. 대부분의 스타트업/중소기업은 여건상 올바르고 다양한 테스트케이스를 생성하는 것은 힘든 실정이다. 이를 해결하기 위해, 말뭉치 정규화와 의미 규칙 기반 요구사항 정제를 통한 테스트 케이스 생성이다. 이를 위한 중간 단계인 원인-결과 그래프 기반 테스트 케이스 자동 생성 방법이 있다. 요구사항으로부터 생성된 원인-결과 그래프의 정확성 검증이 중요하다. 이를 위해, 생성된 원인-결과 그래프의 검증을 위한 C3Tree 모델 설계 방법과 이 모델을 XMI 코드로 자동 생성하는 방법을 제안한다. 이는 원인-결과 그래프의 생성 과정을 단계 별로 자동 가시화함으로써 쉬운 검증이 가능하고, 최종적으로 XMI코드를 생성하여 원인-결과 그래프 기반의 XMI 코드로 자동 변환되기 위한 기반을 마련한다.

### 1. 서 론

소프트웨어 개발 시장에서 소프트웨어 테스트의 중요성은 증가하고 있다. 하지만 중소기업의 경우 테스트의 이해 부족, 개발 시간 부족 등의 이유로 열악한 테스트 환경을 가진다[1].

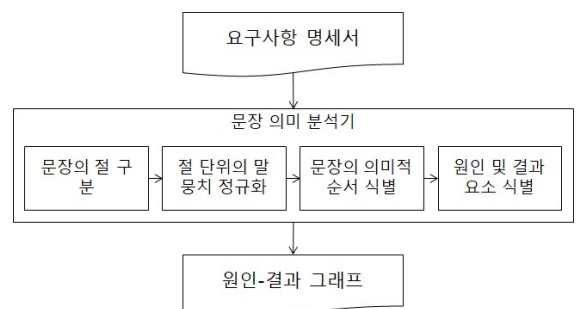
이러한 문제점을 해결하기 위한 방법 중 하나로써, 요구사항 명세서에서부터 테스트케이스를 자동 생성하는 방법이 있다. 하지만 추상적인 요구사항에 대한 테스트 자동화는 사람이 직접 하는 작업에 비해 정확도가 떨어진다. 이러한 문제점을 해결하기 위해 말뭉치 정규화와 의미 규칙 기반 요구사항 정제를 통한 원인-결과 그래프 자동 생성 방법[2]이 있다. 이 방법은 블랙박스 테스트케이스를 쉽게 생성하기 위해 원인-결과 그래프를 생성[3]하는 방법을 제안한다. 하지만 이 방법은 원인-결과 그래프 생성 과정을 검증할 수 있는 방법이 없다.

본 논문은 요구사항 정제를 통한 원인-결과 그래프 자동 생성 과정의 올바른 검증을 위한 C3Tree(Conditional and Conjunction Clause Tree) 모델 설계 방법을 제안한다. C3Tree 모델은 요구사항 문장의 절 관계를 명확하게 식별 및 분류하고, 가시화 한다. 요구사항 문장을 C3Tree 모델로 변환하고, C3Tree 모델을 원인-결과 그래프로 변환하면, 요구사항과 원인-결과 그래프 간의 추적성을 쉽게 확립하고, 표현할 수 있다.

본 논문의 구성은 다음과 같다. 2장은 관련 연구로써 말뭉치 정규화와 의미 규칙 기반 요구사항 정제를 통한 원인-결과 그래프 자동 생성 방법을 언급한다. 3장은 요구사항으로부터 생성된 원인-결과 그래프의 검증을 위한 C3Tree 모델 설계 방법을 언급한다. 4장은 적용 사례로써, 요구사항으로부터 C3Tree 자동 생성 환경을 언급한다. 마지막으로 결론 및 향후 연구를 언급한다.

### 2. 관련 연구

#### 2.1 말뭉치 정규화와 의미 규칙 기반 요구사항 정제를 통한 원인-결과 그래프 자동 생성 방법[2]



<그림 1> 요구사항으로부터 원인-결과 그래프 생성 과정

말뭉치 정규화와 의미 규칙 기반 요구사항 정제를 통한 원인-결과 그래프 자동 생성 방법은 <그림 1>과 같이 요구사항 문장을 정규화[4][5]하고, 문장의 접속문[6]과 조건문[4][7]을 식별하여 결과적으로 문장의 원인 및 결과 요소를 식별한다. 그리고 식별한 결과를 XMI 기반의 원인-결과 그래프로 자동 생성한다. 생성된 원인-결과 그래프는 블랙박스 기반 테스트케이스를 생성하기 위한 기초 모델이 된다[3].

### 3. 요구사항으로부터 생성된 원인-결과 그래프의 검증을 위한 C3Tree(Conditional and Conjunction Clause Tree) 모델 설계

#### 3.1 요구사항 명세서로부터 원인-결과 그래프 변환 시 필요한 요소 식별

C3Tree는 요구사항으로부터 원인-결과 그래프 생성의 중간 과정을 명확하게 분류하고, 가시화한다. 중간 과정에서 식별되는 요소들은 아래의 <표 1>과 같다.

<표 1> 요구사항으로부터 원인-결과 그래프의 생성 과정에서 식별되는 요소들

식별 요소	세부 식별 요소	속성
접속문	선행절	<ul style="list-style-type: none"> <li>뒷 절과의 관계(AND, OR, NOT, NORMAL)</li> <li>내부 절들의 조합 관계</li> </ul>
	후행절	<ul style="list-style-type: none"> <li>내부 절들의 조합 관계</li> </ul>
조건문	조건절	<ul style="list-style-type: none"> <li>조건들의 조합(AND, OR)</li> <li>내부 절들의 조합 관계</li> </ul>
	결과절	<ul style="list-style-type: none"> <li>결과들의 조합(AND, OR, NOT, NORMAL)</li> <li>내부 절들의 조합 관계</li> </ul>

#### 3.2 C3Tree를 구성하는 C3Node의 속성 정의

C3Tree는 C3Node들의 연결로 구성된다. C3Node는 요구사항으로부터 원인-결과 그래프의 생성 과정에서 파생되는 정보를 저장하기 위해 아래의 <표 2>와 같은 정보를 저장한다. Type은 절의 종류를 저장한다. 절의 정보는 조건절(Condition), 결과절(Result), 접속절(Conjunction), 단일 문장(Single), 최상위 절(Top)의 값을 가진다. Sentence는 절의 문장을 저장한다. 일반적인 문자열을 저장한다. Children은 절 내부에서 구성된 세부 절들의 정보를 저장한 C3Node의 링크이다. Combination은 같은 계층의 절들과의 조합 관계로써, AND, OR, NORMAL, SINGALE, LAST의 값을 가진다.

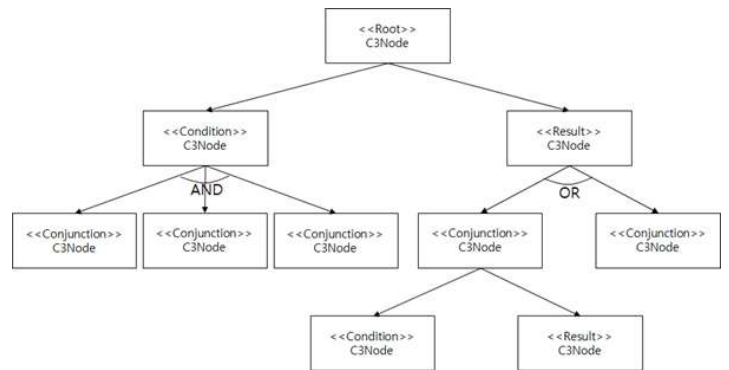
<표 2> C3Node의 속성

종류	설명	값
Type	절의 종류	Condition
		Result
		Conjunction
		Single
		Top
Sentence	절의 문장	String
Children	하위 노드들의 링크	C3Node Link
Combination	다음 노드와의 조합	AND
		OR
		LAST
		NORMAL
		SINGLE

#### 3.2 C3Tree 설계

C3Tree에서 <<Condition>> 노드는 원인-결과 그래프의 원인이 된다. <<Result>> 노드는 원인-결과 그래프의 결과가 된다. <<Conjunction>>은 원인의 조합 또는 결과의 조합이 된다.

C3Node들의 연결되어 구성된 C3Tree의 예는 아래의 <그림 2>와 같다. 첫 번째 계층의 <<Root>> 노드는 최상위 노드으로써, Sentence에 하나의 문장이 저장된다. 두 번째 계층의 <<Condition>> 노드는 <<Root>> 노드의 조건절 정보를 가진다. 두 번째 계층의 <<Result>> 노드는 <<Root>> 노드의 결과절 정보를 가진다. 세 번째 계층에서 AND 관계로 묶인 <<Conjunction>> 노드들은 상위의 <<Condition>> 노드 내에 속한 다수의 접속절들이다. OR 관계로 묶인 <<Conjunction>> 노드들은 상위의 <<Result>> 노드 내에 속한 다수의 접속절들이다. 네 번째 계층의 노드들은 상위의 <<Conjunction>> 노드 내에 속한 접속문 정보를 가진다.



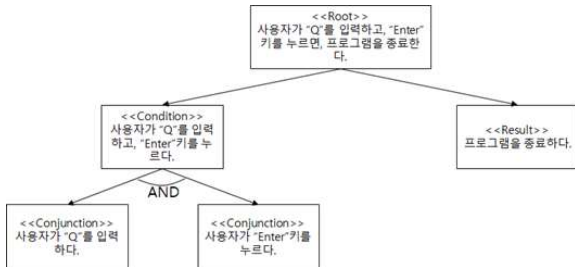
<그림 2> C3Tree의 예

### 4. 적용 사례

아래의 요구사항 문장을 C3Tree로 나타낸 결과는 아래의 <그림 3>과 같다.

사용자가 “Q”를 입력하고, “Enter” 키를 누르면, 프로그램을 종료한다.

첫 번째 계층의 <<Root>> 노드는 하나의 조건절과 하나의 결과절을 포함한다. 두 번째 계층의 <<Condition>> 노드는 두 개의 접속절을 포함한다.



<그림 3> 요구사항으로부터 만들어진 C3Tree의 예

요구사항 문장으로부터 C3Tree의 XMI 코드를 자동 생성한 예는 아래의 <표 3>과 같다.

<표 3> C3Tree의 XMI 코드

```
<?xml version = "1.0" encoding = "ISO-8859-1"?>
<ctm:c3tree xmi:version="2.0" xmlns:xmi = "http://www.omg.org/XMI"
xmlns:xsi = "http://www.w3.org/2001/XMLSchema-instance" xmlns:cem =
"http://cem/1.0">
<c3node xmi:id="cn1" type="root" children="cn2:cn3;" combination="single"
sentence="사용자가 Q를 입력하고, Enter 키를 누르면, 프로그램을 종료한다."/>
<c3node xmi:id="cn2" type="condition" children="cn4:cn5;"
combination="normal" sentence="사용자가 Q를 입력하고, Enter 키를 누른다."/>
<c3node xmi:id="cn3" type="result" children="" combination="normal"
sentence="프로그램을 종료하다."/>
<c3node xmi:id="cn4" type="conjunction" children="" combination="and"
sentence="사용자가 Q를 입력하다."/>
<c3node xmi:id="cn5" type="conjunction" children="" combination="last"
sentence="사용자가 Enter 키를 누른다."/>
</ctm:c3tree>
```

5. 결론 및 향후 연구

본 논문은 요구사항 명세서로부터 원인-결과 그래프 기반의 테스트 케이스 자동생성 할 때, 중간 단계의 원인-결과 그래프 자동 생성 과정을 C3Tree로 표현하는 방법을 제안한다. C3Tree는 요구사항 문장이 분할되고, 정규화되는 과정을 가시화할 수 있고, XMI 코드로 자동 생성된다. C3Tree를 통해 요구사항이 원인-결과 그래프로 바뀌는 과정을 확인할 수 있고, 만약 잘못된 원인-결과 그래프가 생성되었다면, 문장의 해석이 틀린 부분을 쉽게 추적 가능하다. 결과적으로 요구사항 분석 과정의 검증 및 추적이 가능하다.

향후 연구로써, C3Tree의 XMI 코드를 원인-결과 그래프의 XMI 코드로 자동 변형하고, 원인-결과 그래프의 XMI 코드를 이용하여 최종적으로 테스트케이스를 자동 생성하는 방법을 연구할 예정이다.

ACKNOWLEDGEMENT

본 논문은 2019년도 산업통상자원부의 ‘창의산업융합 특성화 인재양성사업’(과제번호 N0000717)과 2017년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(NRF-2017R1D1A3B03035421).

참고 문헌

[1] 권오승, 홍사능, “로그 기반 효과적 반복 테스트”, 한국경영정보학회 추계학술대회, pp.685-690, 2009.  
 [2] 장우성, 김영철, “말뭉치 정규화와 의미 규칙 기반 요구사항 정제를 통한 원인-결과 그래프 자동 생성”, 한국정보처리학회 2019 추계학술발표대회 논문집, Vol.26, No.2, pp.691-693, 2019.  
 [3] Amit Paradkar, K.C. Tai, M.A. Vouk, Specification-based testing .ng cause-effect graphs, Annals of Software Engineering, January 1997, pp.133-157, Volume 4, Issue 1,  
 [4] 조정미, 조영환, 김길창, “구문 관계 지식 추출을 위한 코퍼스 정규화에 대한 연구”, 한국정보과학회 언어공학연구회, pp.207-215, 1996.  
 [5] 조정미, 김길창, “구문 관계 정보 추출을 위한 말뭉치의 정규화에 대한 연구”, 한국인지과학회, Vol.7, No.2, pp.39-56, 1996.  
 [6] 김기성, “현대몽골어와 한국어의 접속문 비교연구”, 한국몽골학회, 몽골학 Vol.27, No.0, pp.151-186, 2009.  
 [7] 하정문, “한국어 조건 연결어미와 중국어 조건 관련사어의 대조 연구”, 경희대학교 대학원 국어국문학과, 2007.