고객 요구사항으로부터 UCP 기반 소프트웨어 공수 산정

박보경[*1], 박영식[*2], 김영철[*3]
[*]홍익대학교 소프트웨어공학연구실
e-mail:{[*1]park, [*2]park12160422, [*3]bob}@selab.hongik.ac.kr

Software Effort Estimation Based on UCP from Customer Requirements

Bo Kyung Park[*1], Young Sik Park[*2], R. Young Chul Kim[*3]
[*1,2,3]SE Lab, Dept. of Software & Communications Engineering, Hongik University

요 약

현재의 소프트웨어는 매우 크고 복잡하다. 고품질 소프트웨어는 요구사항 단계부터 정확하게 분석해야 한다. 그러나 자연어 요구사항은 부정확하기 때문에 유스케이스 식별이 어렵다. 게다가, 기존 요구사항 분석 방법은 객체와 용어 식별에 대한 정확한 기준이 없다. 따라서 분석 결과는 분석가마다 다르다. 본 논문에서는 자연어 요구사항으로부터 유스케이스 추출 및 소프트웨어 공수 산정 방법을 제안한다. 제안한 방법에서는 Goal Modeling과 Fillmore의 Case 메커니즘을 개선했다. 이 방법은 자연어 요구사항을 단계별로 분석하여 유스케이스를 모델링한다. 또한 유스케이스 점수(Use Case Point)를 이용하여 소프트웨어의 공수를 산정(Effort Estimation)한다. 제안한 방법은 고품질 소프트웨어 개발을 위해 자연어 요구사항의 변경 없이 유스케이스 추출이 가능하다. 또한 추출된 유스케이스를 통해 UCP 기반의 공수 산정을 평가할 수 있다. 본 논문에서는 우체국 시스템의 사례에 적용하였다.

1. 서론

현재의 소프트웨어는 매우 크고 복잡하다. 고품질의 소프트웨어를 위해서, 소프트웨어 개발의 시작 단계부터 요구사항을 정확히 분석해야 한다. 그러나 대부분의 요구사항은 자연어로 작성된다. 자연어 요구사항은 부정확하기 때문에 객체와 용어 식별이 어렵다[1]. 이를 해결하기 위해, 많은 자연어 요구사항 분석 방법이 제안되었다. [2]의 연구에서는 필모어의 메커니즘을 통해 사용자 요구사항의 자연어 변환 방법을 제안했다. 변환된 요구사항은 시나리오 기반 분석 방법을 사용하여 요구사항을 식별한다. 그러나 이 방법은 최종 사용자의 요구만 분석하고, 고객의 니즈를 반영하지 않는다. 따라서 개발자는 고객과 최종 사용자의 요구를 동시에 분석하기 어렵다. [3]의 연구는 Goal&Scenario 기반의 자연어 및 요구사항 추출 방법을 이용하여 유스케이스를 분석했다. Goal Modeling은 효과적인 요구사항 식별 방법이다. 시나리오는 현실 세계를 묘사하고 있어 실제 요구사항을 수집하기 쉽다. 그러나 이 방식은 정의된 시나리오로 요구사항의 변경을 요구한다. 따라서 다른 기술보다 더 많은 시간을 필요로 한다.

기존의 요구사항은 모호한 표현으로 인해 요구사항 분석 기준이 명확하지 않다. 따라서 유스케이스 추출이 불가능하다. 또한 원래의 유스케이스 분석은 유스케이스들 간의 영향을 분석하기 어렵다[2]. 이러한 문제 해결을 위해, 본 논문에서는 자연어 요구사항에서 유스케이스를 추출하고 소프트웨어 공수 산정 방법을 제안한다. 제안된 방법은 자연어 문장의 주요 동사를 식별하고 논항을 추출한다. 그런 다음, 시각적 모델링을 통해 유스케이스를 추출한다. 보다 쉽고 정확하게 유스케이스를 추출하기 위해, 본 논문에서는 필모어의 격문법(Fillmore's Case Grammar) 개선 및 Goal Modeling 기술을 적용했다. 유스케이스 및 공수 산정 프로세스는 요구사항 분석, 문장 분석, 동사 리스트, 시각적 모델링을 통한 유스케이스 추출, 유스케이스 점수 기반의 공수 산정의 5단계로 구성된다.

본 논문은 다음과 같이 구성되어 있다. 2장은 관련연구를 설명한다. 3장은 개선된 필모어의 case 메커니즘을 설명한다. 4장은 자연어 요구사항에서 유스케이스 추출 및 공수 산정 절차를 설명한다. 5장은 결론과 향후 연구를 언급한다.

2. 관련 연구

Fillmore의 이론은 심층구조에서 구조격 대신에 의미역을 이용하여 문장을 기술한다[4,5]. 즉 이 이론은 한 문장 내에서 동사를 중심으로 의미적인 관계를 유지한다. 이 관계는 격(Case)이라는 범주로 논항(명사)을 분류한다. Fillmore(1968)는 Agent, Instrumental, Dative, Objective, Factitive, Locative의 6개 격을 제시했다. 1971년에 격은 9개 격으로 수정되었다.

[2]에서 제안한 방법은 Goal 지향 분석을 통해 사용자 행위 기반 요구사항을 추출한다. 이 방법은 5W1H를 기준

으로 사용자 정보(사용자 데이터)를 정리하고, 사용자 문맥 데이터를 생성한다. 이 데이터는 사용 목적, 사용자 및 객체 관계를 분석한다. 그런 다음, 분석가들은 사용자 행위 데이터를 구조화한다. 구조화된 데이터는 사용자 행위와 사용자 액션 사이의 연관성을 분석하여 요구사항을 추출한다. Goal 기반 분석은 Goal 지향 사용자 행위 데이터를 생성한다[9]. 활동(activity)은 각 하위 목표와 사용자 행동(action)을 연관시켜 도출한다. 그런 다음, 분석가들은 각각의 행위(behavior)를 정의한다.

3. 자연어 요구사항으로부터 유스케이스 추출을 위한 개선된 Case Mechanism

본 논문에서는 자연어로 작성된 요구사항으로부터 유스케이스를 추출하기 위해 개선된 격 메커니즘 모델링을 이용한다. 이 메커니즘은 유스케이스 추출을 위해서 UML을 기반으로 기존의 격 메커니즘을 개선하였다. 또한 Goal 기반의 자연어 요구사항 분석을 위해, 원래의 격 메커니즘을 구체화하였다. 이 방법은 동사를 바탕으로 단어 간의 의미적 관계를 분석한다. 따라서 자연어 처리가 용이하다. 또한 격 메커니즘은 더 큰 단위인 유스케이스 식별로부터 작은 단위의 객체와 기능을 식별할 수 있다. 개선된 격 메커니즘은 8가지 격으로 구성되어 있다. 표 1은 개선된 격이다.

<표 1> 개선된 Case

Case	Notation	Definition
Actor	A	이벤트 실시
Secondary -Agent	SA	행동이 수행되는 힘 또는 저항
Object	O	행동에 영향을 받거나 받아들이거나 경험하거나 겪는 엔티티
Theme	T	행동에 의해 변경되는 주체
Source	SA	위치 또는 공간 방향
Verb	V	행동의 결과로 존재하는 엔티티(즉, 동사)
Instrument	I	이벤트의 자극 또는 즉각적인 물리적 원인
Goal	G	사용자의 요구로부터 달성되는 시스템 목표

유스케이스 추출 모델링 방법은 개선된 격문법을 자연어 요구사항에 적용하고, 모델로 변환한다. 변환된 모델을 이용하여 연관관계에 따라 유스케이스를 추출한다.

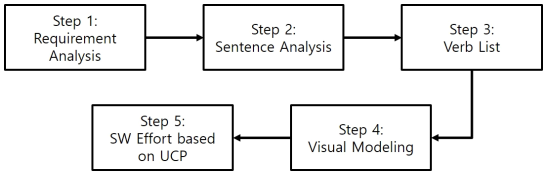
<표 2> Use Case Notation[5]

Notation	Comment	Notation	Comment
P	Predicate (main verb)	→	Semantic Correlation
A	Argument (noun)	⇔	Identical Proposition
I	Identification of the Same 'A' Type	- · - · →	Identification Line

표 2는 유스케이스 식별 표기법이다. 이 표기법은 요구사항 명세서를 기반으로 논항을 식별하며, 서술어, 논항, 식별자로 구성된다. 서술어는 동사, 명사, 형용사 등을 나타낸다. 논항은 서술어의 특성에 따라 취할 수 있는 명사이다. 마지막으로 식별자는 논항을 식별한다. 약어로 'P'는 문장의 서술어이다. 'Ai'는 서술어의 특성에 따라 취할 수 있는 논항들이다. 'I'는 식별자가 존재하는 경우이다.

4. 자연어 요구사항으로부터 유스케이스 추출 및 공수 산정 방법

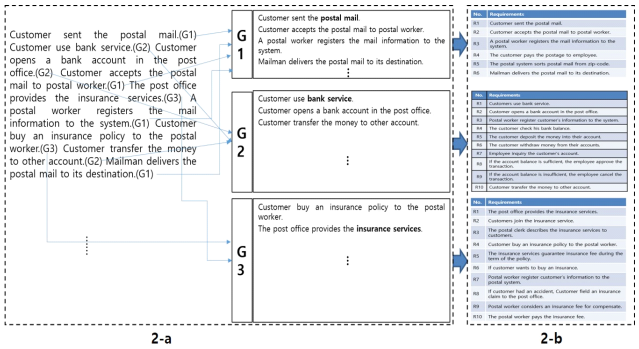
본 논문에서는 격 메커니즘을 이용하여 자연어 요구사항에서 유스케이스 다이어그램을 추출한다. 또한 추출된 유스케이스에서 UCP 기반의 공수 산정을 평가한다. 제안된 방법의 장점은 다음과 같다. 1) 개선된 격문법과 Goal Modeling을 통해 Goal에 맞는 요구사항을 분석할 수 있다. 2) 자연어 요구 사항에 개선된 격 메커니즘을 적용했다. 결과적으로 이 방법은 유스케이스를 쉽게 생성 할 수 있다. 3) 이 방법은 UCP 기반의 SW Effort를 추출할 수 있다. 유스케이스 추출 및 공수 산정 절차는 총 5단계로 구성된다. 본 논문에서는 제안한 방법을 우체국 시스템에 적용하였다. 그림 1은 유스케이스 추출 및 공수 산정 프로세스이다.



(그림 1) 유스케이스 추출 및 공수 산정 프로세스

Step 1: 요구사항 분석

이 단계는 자연어로 작성된 요구사항을 Goal 중심으로 분석한다. 이 단계는 자연어 요구사항을 분석하여 하위 목표를 추출한다. 각 요구사항은 하위 목표에 따라 분류된다. 본 논문에서는 개선된 목표 계층 구조 방법을 적용하였다. 그림 2는 자연어 요구사항 분석을 위해 개선된 Goal 계층 구조 방법을 적용한 것이다. 그림 2-a는 자연어 요구 사항을 분석하여 하위 목표를 추출한다. 각 요구사항은 하위 목표에 따라 분류된다. 그림 2-b에서처럼, 이 요구사항들은 문장 별로 번호를 부여하고, 리스트화한다.

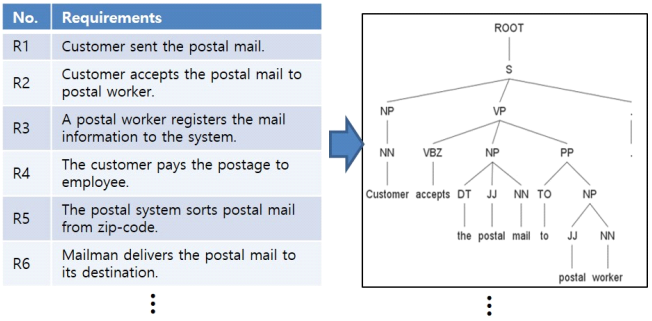


(그림 2) 개선된 Goal 계층 구조 기반의 자연어 요구사항 분석

Step 2: 요구사항 문장 분석

이 단계는 Parser를 이용하여 요구사항의 문장 구조를 분석한다. 여기서 파서는 스탠포드 파서(Stanford Parser)를 적용하였다. 스탠포드 파서는 신경망을 기반으로 하며, 문장의 문법 구조를 결정한다. 종속성의 특성을 분석함으로써, 파서는 문장의 텍스트 관계를 검출한다. 이 의존성은 그래프에서 문장의 문장과 관계를 나타낸다. 문장의 종

속성 특성 분석을 통해 문장의 텍스트 관계를 추출하였다.



(그림 3) 문장 분석

Step 3: 동사 리스트

Step 4: Visual Modeling

시각적 모델링은 6단계로 나뉜다.

4-1단계: 요구사항 분석 및 핵심 술어를 검토한다.

4-2단계: 핵심 동사의 성질을 조사하여 그 주요 역할을 식별한다.

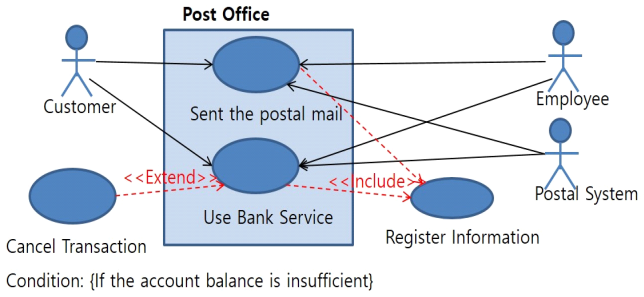
4-3단계: 식별된 논항과 관련 동사를 추출한다.

4-4 단계: 동사와 논항의 관계를 식별한다.

4-5 단계: 관련 유스케이스 다이어그램을 추출한다.

4-6 단계: '포함과 확장'사이의 관계를 식별한다.

위 단계를 통해서, 그림 4는 'Use-Bank Service' 유스케이스와 'Cancel Transaction'유스케이스의 확장 관계를 보여준다.



(그림 4) 우체국 Use Case 다이어그램

Step 5: 공수 산정 방법

이 단계는 SW Effort Estimation이다. 우리는 추출된 유스케이스에서 개발하고자 하는 소프트웨어의 Total Man-Hours를 계산한다[7]. 계산 방법은 Use Case Point를 사용한다. Man-hours를 구하기 위해서는 개발 노력에 대한 추정 값이 필요하다. Karner는 이 값을 20 시간/UCP로 정의하였다. 본 연구는 Karner가 제시한 20시간/UCP를 적용하였다. 식 1은 Total Man-hours의 계산 방법이다.

Total Man-Hours = ER(Effort Rate)*UCP --- (식1)

위의 사례에서 추출된 유스케이스는 4개이다. 표 3은 Total Man-Hours이다.

<표 3> 최종 공수 산정 결과

UseCase	Actor	UC	UUCP	TCF	EF
Send postal mail	3	15	18	30	11
Use Bank Service	3	15	18	32	11.5
Cancel Transaction	2	10	12	28.5	9.5
Register Information	1	5	6	16.5	9
Total	9	45	54	107	41
Final TCF				1.67	
Final EF				0.17	
UCP				15.3306	
Total Man-Hours(ER: 20)				306.612	

5. 결론

고품질의 소프트웨어를 개발하려면 소프트웨어 개발의 초기 단계에서 요구사항을 정확하게 분석해야 한다. 그러나 자연어 요구사항은 분석하기가 어렵다. 우리는 자연어 처리 기술과 Goal 모델링을 통한 유스케이스 추출 방법 및 공수 산정 방법을 제안하였다. 자연어 요구 사항을 파악하기 위해 필모어의 격문법과 Goal 모델링을 개선했다.

제안한 방법은 자연어 요구사항의 변경 없이 유스케이스 추출이 가능하며, 이는 사람들이 이해하기 쉽게 자연어 요구사항을 처리할 수 있다. 또한 UCP 기법을 기반으로 공수 산정이 가능하다.

ACKNOWLEDGE

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

참고문헌

[1] Karl E. Wiegers, "Software Requirements," Microsoft Press, 2003

[2] 안성빈, 김동호, 서채연, 김영철, "Fillmore의 Case Grammar를 통한 사용자 요구사항으로부터 객체 추출 및 모델링 방법", 한국정보과학회, Vol.37, No.1, pp.98-99, 2010

[3] Jintae. Kim, Sooyong. Park, and Vijayan. Sugumaran, "A Linguistics-Based Approach for Use Case Driven Analysis Using Goal and Scenario Authoring," Natural Language Processing and Information Systems Lecture Notes in Computer Science, Vol. 3136, pp. 159-170, 2004.

[4] 박보경, 권하은, 강건희, 양효석, 황준순, 이근상, 김영철, "요구사항으로부터 유스케이스 추출 및 우선순위화 방안", 소프트웨어공학회, Vol.17, No.1, pp.394-395, 2015

[5] 김보연, "개선된 Fillmore의 Case 메카니즘 기반 고객 요구사항에서 유스케이스 추출 방법", 석사학위논문, 홍익대학교, 2014.

[6] Karner, Gustav, "Resource Estimation for Objectory Projects," Objective Systems SF AB, 1993.