Louvain 클러스터링 알고리즘을 통한 요구 사항 상관성 추출

변은영O, 문소영, 서채연, 김영철
홍익대학교 소프트웨어공학연구실
{eybyun, msy, chyun, bob}@selab.hongik.ac.kr

Extracting Association Relationships among Requirements through Louvain Clustering Algorithm

Eun Young ByunO, So Young Moon, Chae Yun Seo, R. Young Chul Kim
SE Lab, Dept. of Computer Information Communication, Hongik University

요 약

소프트웨어 요구사항 분석 및 유스 케이스 모델링은 사용자의 목표를 정의하고 시스템을 파악하기 위해 필수적이다. GS, CMMI 등 소프트웨어 품질을 인증하기 위한 다양한 방식이 제안되고 있지만 실제 중소기업은 인력 및 비용의 한계가 있다. 소프트웨어 개발 초기 프로세스가 체계적으로 이루어지지 못하면 개발 효율이 감소할 뿐만 아니라 유지보수 시에 문제가 발생한다. 이 논문에서는 사용자의 요구사항을 기반으로 요구사항 간의 상관성을 분석하여 유스 케이스를 추출하기 위한 클러스터링 기법을 연구한다. 요구사항을 그래프화하여 Louvain Algorithm을 적용하고 이를 통해 식별되는 클러스터를 하나의 유스 케이스로 추출한다. 이를 자동화하기 위해 파이썬을 기반으로 자연어 처리와 그래프 가시화를 위한 API를 사용하여 시스템을 구축한다. 검증을 위해 '여행 상품' 시스템의 요구사항을 대상으로 적용하고 그 결과를 분석한다. 이 메커니즘을 통해 상관성이 있는 요구사항을 하나의 클러스터로 식별하여 소프트웨어 프로세스 산출물인 유스 케이스를 확보하고자 한다.

1. 서 론

소프트웨어 고품질을 위해 개발초기에 정확한 요구사항 분석 및 유스 케이스 모델 작성이 필수적이다. 유스 케이스는 사용자 관점에서 소프트웨어 행위자와 기능의 상호작용 관계를 모델링한다. 이를 통해, 사용자의 목표를 정확하게 정의하고 시스템의 간결한 파악이 가능하다[1-3].

GS, CMMI 등 소프트웨어의 품질을 인증하는 다양한 메커니즘에서 소프트웨어 프로세스에서의 산출물들의 중요성을 강조하고 있지만, 실제 소프트웨어 산업에서는 시간/비용의 문제로 유스 케이스의 문서화가 진행되지 않고 있다. 특히, 중소기업은 이를 따르기에는 한계가 있다. 이는 개발 및 유지보수를 진행함에 있어 주요한 문제점으로 작용한다.

이 논문에서는 요구사항 간의 상관성 분석을 통해 유스 케이스를 식별하기 위한 클러스터링 기법을 제안한다. 사용자의 요구사항을 분석하고 명사, 서술어를 식별하여 그래프화 한다. 그래프의 노드들을 대상으로 클러스터링 기법을 적용하여 요구사항 상관성 분석 및 유스 케이스 추출을 수행한다. 본 논문의 구성은 다음과 같다. 2장은 관련연구로 기존 클러스터링 알고리즘을 조사하고 요구사항 상관성 분석에 적합한지 분석한다. 3장은 요구사항 클러스터링 기법 및 자동화 시스템을 설명한다. 4장에서는 제안한 기법을 검증하기 위해 '여행 상품' 에 대한 요구사항을 자동화 시스템의 대상으로 적용하고, 그 결과를 기술한다. 5장에서는 결론 및 향후 연구를 언급한다.

2. 관련연구

그래프 클러스터링에 대한 다양한 연구가 진행되었다[4-6]. 클러스터링 알고리즘에 대해 조사하고, 요구사항 그래프의 클러스터링을 위해 적합한 알고리즘을 식별한다.

2.1. Strongly Connected Component

방향성 그래프에서 강하게 연결된 컴포넌트이다. 컴포넌트 내부의 모든 노드들이 자신을 제외한 다른 노드에 접근 가능하다면 이들을 강하게 연결된 하나의 컴포넌트로 식별한다[5]. 이 알고리즘은 순환 가능한 노드들을 하나의 컴포넌트로 인식한다. 따라서 요구사항 그래프에서는 주어와 목적어가 변경된 경우에만 순환 가능한 컴포넌트로 식별가능하다.

2.2. Louvain Clustering Algorithm

Blondel이 고안한 Louvain Clustering Algorithm은 대규모 네트워크에서 커뮤니티를 추출하는 기법이다. 표 1은 Louvain Clustering Algorithm의 과정이다. 각 노드로부터 인접된 노드들

과 모듈성을 측정하고 그 수치가 최대가 되는 커뮤니티를 찾는다[6]. 이 알고리즘은 순환되는 노드와는 관계없이 노드들 간의 연결성을 기반으로 클러스터링을 수행한다. 요구사항 그래프에서는 동일한 단어가 다른 기능의 요구사항에서 포함될 가능성이 높으므로 이를 적용하여 유스 케이스를 추출하고자 한다.

표 1 Louvain Clustering Algorithm[6]

1. Put all vertices into distinct communities
2. Refine communities
▶ For each vertex i
• Compute $\triangle Q_{i \to c(j)}$ for each neighbor j
• Join the community c(j) that yields the largest gain in $\triangle Q$
▶ Repeat until no movement yields a gain
3. Reconstruct the graph
▶ The partitions become super vertices
▶ The weights of edges between communities are summed
4. Repeat steps 2 and 3 until convergence

3. 요구사항 클러스터링 기법 및 자동화 시스템

Louvain Clustering Algorithm을 적용하여 요구사항을 클러스터링 하는 자동화 시스템을 구축한다. 그림 1은 자동화 시스템 아키텍처이다. 이는 파이썬 언어로 개발되었고, 자연어로 작성된 요구사항을 분석하기 위한 Natural Language Parsing Tool(NLPT), 그래프 시각화를 위한 plotly를 API로 사용한다.

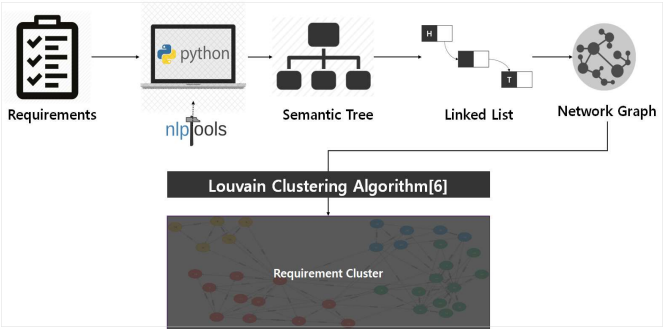

그림 1 자동화 시스템 아키텍처

시스템의 구조에 따른 각 단계를 설명하기 위해 표 2의 '여행 상품 요구사항'을 적용한다. 자연어로 작성된 요구사항을 분석하여 구문 트리로 구성하고 POS 태깅을 통해 주어, 동사, 목적어 등의 명사를 구분한다. 이를 기반으로 그림 2와 같이 주어->동사, 동사->명사들의 연결 리스트를 구성한다. 이를 기반으로 그래프의 변화 과정은 그림 3과 같다. R1은 동사 'be'를 중심으로 여러 명사('Hotels', 'city', 'selection', 'dates', 'room', 'type', 'bed')들과 하나의 그래프를 구성한다. 'be' 동사는 여러 명사들과 연결되므로 상관성을 추출할 때 하나의 유스 케이스 동사로 가능하다. 수동태 문장이 많다면 이런 경우로 인해 상관성 추출에 오류를 발생시킬 수 있으므로 전처리가 필요하다. R2 또한 동사 'be'를 중심으로 구성된 문장이므로 기존 'be' 노드에 새로운 명사들이 노드로 연결된다. Ticket Search 요구사항인 R5 또한 'be 동사를 중심으로 구성된 그래프이므로, 이 단계에서는 Hotel Search 와 Ticket Search의 구분이 어렵다. 하지만, R6는 새로운 동사 ' show'를 생성하며 새로운 중심 노드를 생성한다.

표 2 여행 상품 요구사항

여행 상품		
Hotel Search	R1	Hotels can be searched based on city selection, check-in out dates, room type and bed type
	R2	Hotels can be displayed along with the selected search conditions along with the photos, ratings, and information
	R3	Hotels can be sorted by name, rank, low-price, high-price
	R4	Each Hotel product can be booked, changed, and registered as a favorite product
Ticket Search	R5	The ticket can be searched based on the departure city, destination city, round-trip, one-way selection, departure date of Korea
	R6	The ticket show the price, TAX, date, airline, stay, destination, stopover, flight, availability
	R7	Each ticket can be booked, changed, or registered.
	R8	The joint purchase has the function of the customer purchasing the ticket presented by the travel agency
	R9	The joint purchase list shows the date of departure, the airline, the length of stay, the price, the destination, and the waypoint

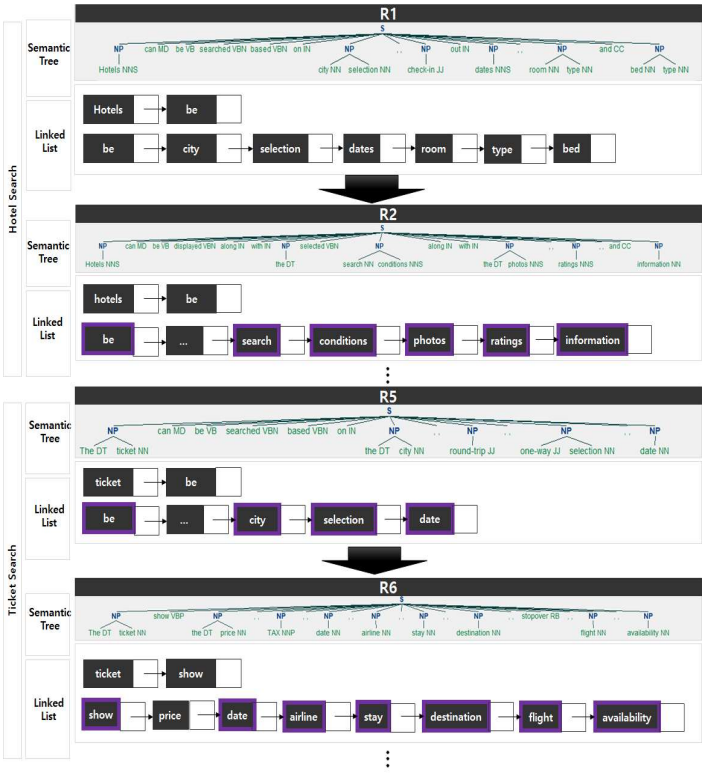

그림 2 Semantic Tree 기반의 Linked List 생성

그 후, Louvain Clustering Algorithm을 적용하여 노드들을 클러스터링하며 그 결과는 다음 장에서 사례 연구로 설명한다.

4. 사례 적용 및 결과 분석

'여행 상품'에 대한 요구사항을 대상으로 적용하였다. 표 2에 따르면 여행상품의 요구사항은 기능적으로 호텔 검색, 항

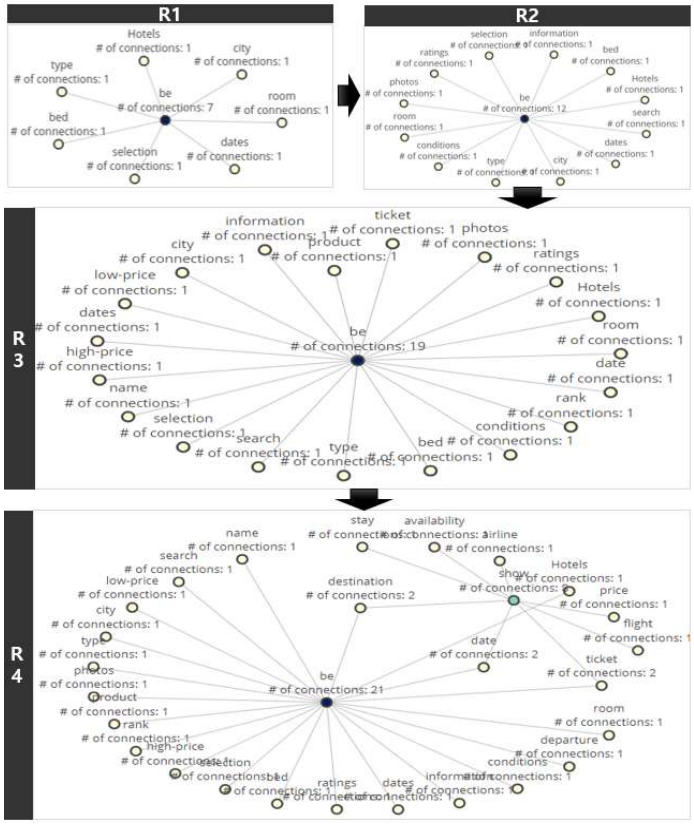

그림 3 Linked List 기반의 그래프 생성

공권 검색 두 개의 항목으로 구분된다. 해당 요구사항을 클러스터링 자동화 시스템에 입력한 결과 명사, 동사를 기반으로 그래프를 구성하고 이에 Louvain 클러스터링 알고리즘을 적용하면 노드 간의 연결을 기반으로 연관성을 측정하여 커뮤니티를 생성한다. 그림 4는 여행 상품 요구사항 클러스터링 가시화이다. 생성된 커뮤니티는 빨간색 노드와 파란색 노드, 총 2가지이다. 이 그래프와 '여행 상품' 요구사항을 비교해본 결과 기능적으로 호텔 검색, 항공권 검색을 분류 가능하다. 두 항목에 동일하게 사용된 'destination', 'date', 'departure', 'ticket'과 같은 단어들은 두 컴포넌트에 모두 연결되어 있지만, 검색 항목에 더 많이 사용되었으므로 검색 항목 컴포넌트로 포함되어 파란색으로 분류된다.

5. 결론 및 향후 연구

요구사항 그래프에 Louvain 알고리즘을 적용한 요구사항 클러스터링 기법을 제안한다. 이를 통해 기능적 요구사항의 상관성을 측정하여 유스 케이스를 추출하고자 한다. 두 가지의 카테고리로 분류되어 있는 '여행 상품' 요구사항을 대상으로 적용하여, 제안하는 기법 및 구축한 시스템의 검증을 시도했다. 이 메커니즘을 통해 사용자의 요구사항을 기능별로 클러스터링하여, 소프트웨어 프로세스 산출물인 유스 케이스를 확보하기 위한 기반을 마련하고자 한다.

하지만, 명확한 요구사항을 정의되었다는 전제 하에 이 기법은 적용될 수 있다. 향후에는 대소문자, 단/복수 단어들을 하나의 단어로 식별하고, 동사와 명사의 의미로 모두 쓰이는 단어들을 처리하고자 한다. 또한, 수동태 문장들은 능동태로 변환하여 클러스터링 기법의 정확성을 향상시키고자 한다.

Acknowledge

이 논문은 2017년도 정부(교육부) 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업(NRF-2017R1D1A3B03035421)과 2018년도 정보통신산업진흥원의 정보통신, 방송 연구개발사업(개방형OS 환경개발 및 보급, 확산 사업)의 지원을 받아 수행된 연구임(S1113-18-1001).

참고 문헌

[1] http://dthomas-software.co.uk/resources/frequently-asked-questions/why-implement-use-cases/
[2] 권하은, 박보경, 김영수, 김영철, "역공학을 통한 소스코드로부터 유스케이스 설계 추출", 한국스마트미디어학회, 제 5권, 제 1호, pp. 289-291, 2016
[3] 박보경, 권하은, 강건희, 양효석, 황준순, 이근상, 김영철, "요구사항으로부터 유스케이스 추출 및 우선순위화 방안", 소프트웨어공학회, 제 17권, 제 1호, pp. 394-395, 2015
[4] Scott Emmons, Stephen Kobourov, Mike Gallant, Katy Borner, "Analysis of Network Clustering Algorithms and Cluster Quality Metrics at Scale", 2016.
[5] Strongly Connected, https://en.wikipedia.org/wiki/Strongly_connected_component
[6] Louvain, https://en.wikipedia.org/wiki/Louvain_Modularity

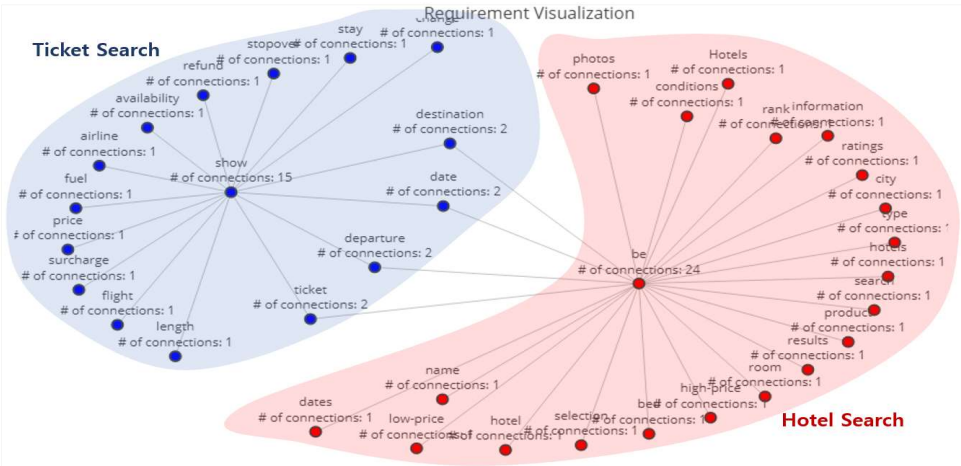

그림 4 여행 상품 요구사항 클러스터링 가시화