FCA 기반 계층적 구조 표현을 이용한 문서 통합 기법

김태환° 박제현 최중민
한양대학교 컴퓨터공학과
{kimth°, jhpark, jmchoi}@cse.hanyang.ac.kr

Methods for Integration of Documents using Hierarchical Structure Representation based on the Formal Concept Analysis

Taehwan Kim° Jaehyun Park Joongmin Choi
Department of Computer Science & Engineering
Hanyang University

요 약

가공해서 사용하는 정보량이 많아질수록 원하는 정보를 찾는 데 더 많은 노력이 필요하게 마련이다. 따라서 사람들은 대대로 정보를 구조화하는 방법들을 고안해왔으며, 여러 가지 계층적 구조화 방법들을 사용했었다. 이렇게 구현된 정보의 계층 구조는 키워드 검색을 바탕으로 수평적 계층 구조만을 가지는 구조였다. 자료가 전문화되고 정보를 검색하는 사용자 또한 검색된 정보와 관련된 정보를 더 원하는 현 시점에서 정보의 수평적 계층 구조만으로 사용자의 만족도를 충족할 수 없다.

이러한 문제점을 해결하기 위해 이 논문에서는 특정 도메인의 문서를 단락별 명사와 동사 및 목적어를 추출하여 해당 동사가 명사 및 목적어를 취할 수 있는 가능한 값을 체크하여 그 단락의 계층적 트리를 구성하고, 단락별 트리를 이용하여 문서의 내용을 트리로 재구성할 수 있게 된다. 이렇게 만들어진 문서의 트리들은 트리의 구조를 보고 특정 문서에 더 구체적인지 아니면 더 일반적인지 측정하여 문서와 문서간의 관계 또한 트리 형식으로 보여주어 사용자가 원하는 정보를 보다 쉽게 검색해 주는 자동화 문서 계층 구조를 제안한다.

1. 서 론

인터넷 페이지의 증가와 함께 검색엔진의 발전은 기존의 "검색"에 대한 개념을 바꾸어 놓고 있다. 연구자들은 더 이상 도서관에서 지면으로 발행된 논문지를 찾지 않고, 자신의 책상에서 인터넷으로 간단한 검색어를 입력하여 필요한 자료를 검색하여 찾는다. 여기에, 대학의 도서관들도 검색엔진과 도서 자료를 유기적으로 조합시켜 공개하고 있다. 도서관에서 직접 연구 자료를 찾는 것과 비교하면, 인터넷을 통한 검색은 많은 양의 정보를 편리하게 전달한다. 그러나 정보가 많아지면 많아질수록 실제로 관련 있는 연구 문서를 찾기 힘들어 지고, 연관성이 있는 연구 문서를 찾았다 하더라도 이 연구 문서와 관련된 문서를 다시 연계해서 찾기도 어렵다. 이러한 문제점을 해결하기 위해 CiteSeer에서는 ACI(Autonomous Citation Indexing) 기법을 제안하면서 이런 문제점을 해결하려 했다.[3] ACI 기법은 한 논문에서 인용된 논문들의 리스트를 페이퍼 원문에서 자동으로 인덱싱해서 인용된 페이퍼의 링크를 생성해 주는 기능이다.

CiteSeer의 문제점은 연구 문서간의 참고 문헌 정보를 참조하여 문서들을 연결하였기 때문에 저자가 참조하지 않은 다른 연구 문서와의 연계가 불가능하다는 것이다. 또한 연구 논문에만 한정해서 만들어 졌기 때문에 확장성이 용이하지 못하다. 이 논문에서는 이러한 단점들을 해결하기 위해 데이터 분석을 위해 사용되었던 FCA(Formal Concept Analysis)를 이용하여 문서를 하나의 계층적 개념 트리로 표현하고, 이 개념 트리를 통합하여 만든 문서 간의 계층적 구조를 이용하여 키워드 별 랭킹 시스템을 제안하려 한다.

2. 관련 연구

2.1 FCA(Formal Concept Analysis)

Formal Context는 문장 내에서 객체와 속성을 추출해낸 결과의 집합을 이야기한다. 정형적으로 Formal Context K는 $K=(G, M, I)$로 정의되며, 객체(주어)들의 집합 G와 속성(서술어)들의 집합M, 그리고 G와 M사이의 이항관계 $I \subseteq G \times M$ 으로 구성된다. 이항관계 I는, "G의 원소 g는 M

* 본 논문은 정통부 및 정보통신연구진흥원의 정보통신선도기반기술개발사업의 연구 결과로 수행되었습니다.

의 원소 m을 가지고 있다"라는 것을 나타낸다. 이 정의를 이용하여 두 집합 A', B'을 다음과 같이 정의하였다.

$$A' := \{ g \mid (g, m) \in I, m \in M, g \in A \}$$
$$B' := \{ m \mid (g, m) \in I, m \in M, g \in B \}$$

이 때 집합 A는 $A \subseteq G$이고, 집합 B는 $B \subseteq M$이다. 여기서 $(A=B') \wedge (B=A')$를 만족시킬 때, A는 extent라고 부르고 B는 intent라고 부른다.

위와 같은 정의를 바탕으로 문장으로부터 다양한 개념 (g, m)을 추출할 수 있다. 이러한 개념들 사이에는 일종의 상-하위 관계에 따른 순서가 존재한다. 즉, 임의의 개념 (A_1, B_1)과 (A_2, B_2)에 대하여, $A_1 \subseteq A_2$이거나 $B_2 \subseteq B_1$ 일 때, 개념 (A_1, B_1)은 개념 (A_2, B_2)의 하위 개념이라고 하며, 반대로 개념 (A_2, B_2)는 개념 (A_1, B_1)의 상위 개념이라고 한다.

이와 같이 주어진 문장으로부터 개념들을 추출하여 상위개념-하위개념 관계를 구성함으로써, 격자구조(Complete Lattice)를 구축할 수 있다. 또한 추출된 개념들은 자연스럽게 객체집합이나 속성집합에 의한 계층적 관계가 형성이 되며 이를 통해 개념격자(Concept Lattice)를 구축할 수 있다. 개념격자는 개념을 나타내는 정점들과 상-하위 개념 관계를 나타내는 변으로 구성되고 이것을 다시 표 1과 같은 형태로 만든다. 표 1.에서 가로는 속성을 나타내고 세로는 객체를 나타낸다. 객체가 가질 수 있는 속성의 수가 가장 많은 것이 상위 노드가 되고, 적은 것이 하위 노드가 된다. 이러한 방법으로 표 1.는 그림 1.과 같은 *문서 개념트리*로 변환될 수 있다. 이 트리의 각 노드에는 여러 개념이 갖는 extent들과 intent들에 대한 정보가 레이블로 표시된다. [2]

	Bookable	Rentable	Driveable	Rideable	joinable
hotel	X				
Apartment	X	X			
Car	X	X	X		
bike	X	X	X	X	
excursion	X				X
trip	X				X

표 3. Formal Context

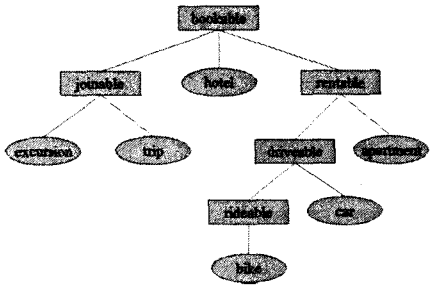

그림 1. Formal Concept Tree

2.2 CiteSeer

CiteSeer는 과학관련 논문들의 전자 도서관으로서 기본적인 논문 자료 열람과 검색기능 이외의 다양한 추가 기능을 제공한다.[6] CiteSeer의 대표적인 서비스인 ACI는 그림 2.와 같이 한 논문에서 인용된 다른 논문들의 목록을 논문 원문에서 자동으로 색인화하여 연결을 생성해주는 기능이다. 또한 CiteSeer는 인용된 논문들을 계속해서 갱신하여 연결하기 때문에, 한 논문과 관련된 최근의 연구 결과들을 찾아 볼 수 있다. 이외에도 검색 시 검색어와 관련된 부분만을 본문에서 요약해서 보여주는 기능, 논문의 인용빈도를 나타내는 그래프 작성 기능을 제공하고 있다. [4]

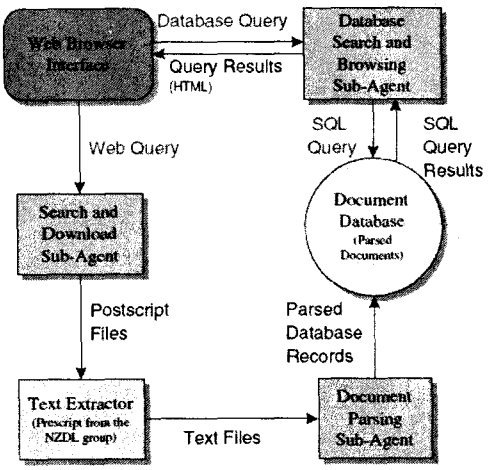

그림 2. CiteSeer Agent Architecture

그런데, CiteSeer는 논문에서 인용된 다른 논문들의 목록만을 보여주기 때문에 저자가 인용하지 않은 다른 논문들에 관한 참조 관계는 표현할 수 없다. 또한 논문간의 인용관계에 한정하였기 때문에 논문이 아닌 타 문서간의 연관관계를 확장하여 표현할 수 없다.

이 논문에서는 참조된 논문의 내용이 참조한 논문이 가지고 있는 특정 객체에 대하여 구체적이거나 일반적으로 설명하고 있을 때 참조관계가 성립한다고 가정하였다. 그래서 문서를 객체와 속성으로 이루어진 트리로 구성하고 그 트리를 병합함으로써 문서 간 일반화/상세화 관계를 표현하였다. 그래서 명시적인 참조관계가 존재하지 않아도 이 병합 트리를 이용하여 참조관계를 파악할 수 있다.

3. 문서의 통합 방법

FCA를 이용해서 만들어진 문서의 개념트리를 통합하기 위해서는 수직적 계층 트리의 통합과 수평적 계층 트리의 통합이 필요하다. 수평적 계층 트리는 문서간의 트리의 깊이가 같고, 해당 어휘가 비교적 상이한 상태에서의 통합이고, 수직적 계층 트리는 문서간의 트리의 깊이가 다르고, 해당 어휘가 유사한 상태에서의 통합이다.

3.1 문서 간 관계 정의

집합 D는 문서 집합을 나타내며 FCA에 적용하기 위하여 다음과 같이 정의한다.

$$D = (C, A, I_D)$$

이 때 C는 문서에서 나타내고 있는 객체, A는 속성을 말한다. I_D는 이항 관계로서 $I_D \subseteq I$이다.

정의 1. 불일치 관계

$$A_{ij} = \{\ a \mid (c_i, a) \in I_{dj},\ c_i \in C\ \}$$
$$C_{ij} = \{\ c \mid (c, a_i) \in I_{dj},\ a_i \in A\ \}$$

에서 다음의 조건이 만족되면 A_{ij}와 A_{ik}는 서로 불일치 관계에 있다고 한다.

$$A_{ij} \cap A_{ik} = \varnothing \ \wedge\ C_{ij} \cap C_{ik} = \varnothing$$

이 때 두 문서 d_j와 d_k는 유사하지 않다고 간주할 수 있다.

정의 2. 포함 관계

두 문서 d_j와 d_k에 대하여 다음의 조건이 만족되면

$$C_{ij} = C_{ik} \ \wedge\ A_{ij} \subset A_{ik}$$

문서 d_j는 d_k에 포함된다고 한다.

정의 3. 동등 관계

만일 문서 d_j가 d_k에 포함되고 동시에 d_k가 d_j에 포함될 때, 두 문서는 서로 동등 관계에 있다고 한다.

정의 4. 부분 관계

두 문서 사이에 포함 관계가 성립하지 않을 경우, 다음의 조건을 만족한다면 두 문서는 부분 관계에 있다고 한다.

$$A_{ij} \cap A_{ik} \neq \varnothing \ \wedge\ C_{ij} \cap C_{ik} \neq \varnothing$$

3.2 수평적 계층의 통합

객체와 속성이 두개의 문서에 같이 존재하고 트리의 구조가 다를 때 그림 3.과 같이 문서의 통합이 이루어진다.

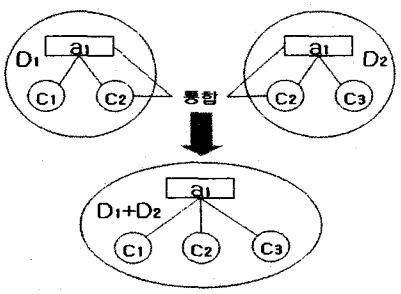

그림 3. 수평적 계층의 통합 예

그런데, 하나의 문서에서 속성으로 사용된 객체가 하나인데 반해 다른 문서에는 같은 속성에 사용된 객체가 두 개인 경우, 어느 노드로 통합할 것인지 결정해야 한다. 이 경우에는 해당 객체간의 유사도를 측정하여 그 값에 따라 트리를 구성한다. 유사도 측정 방법은 다음과 같다.[7]

$$H(c) = -\log P(c)$$
$$sim(c_i, c_j) = \max_{c \in S(c_i, c_j)} (H(c))$$

이 때 $P(c)$는 객체 c가 나타날 확률이고, 집합 $S(c_i, c_j)$는 c_i와 c_j를 포함하는 객체들의 집합이며, $H(c)$는 두 개의 객체 간 유사도를 비교할 때 WordNet 상에서 두 객체가 동시에 부모로 갖는 최하위 노드를 나타낸다. 예를 들어, 그림 4.에서 car와 bicycle의 유사도를 측정하기 위해서는 $H(c)$의 값이 필요한데 여기서 car와 bycycle의 상위 분기 노드인 vehicle이 $H(c)$가 된다. 이것을 위의 식으로 표현

하면 car와 bicycle의 유사도는 *sim*(car, bicycle)이고, 상위 분기 노드인 vehicle은 H(bicycle)이다.

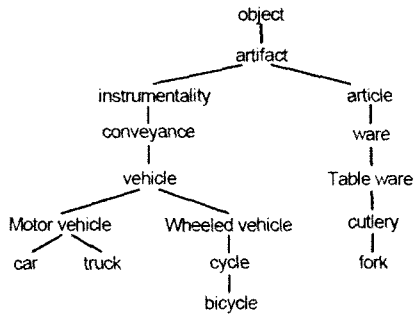

그림 4. WordNet의 개념에 대한 정보의 예

이런 방법으로 유사도를 측정하며, 그림 5.와 그림 6과 같이 어느 객체의 유사도에 따라서 트리의 구성이 달라진다.

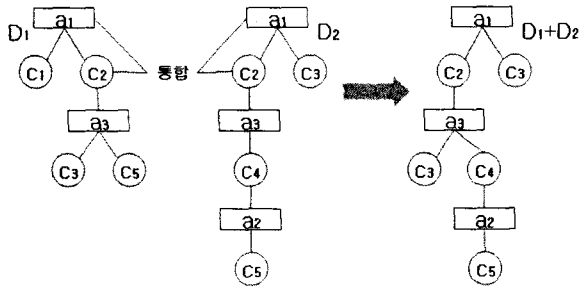

그림 6. C_3가 C_5보다 C_4와 더 유사한 경우

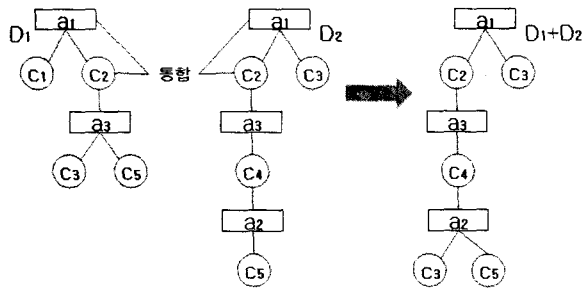

그림 6. C_3가 C_4보다 C_5와 더 유사한 경우

3.2 수직적 계층의 통합

문서 간 상위 속성이 같고 그 문서 간 사용된 객체가 존재한다면 그 객체가 사용된 속성이 다른 것에 상관없이 그림 7과 같이 통합 될 수 있다. 만일 수직적 계층 구조에서 수평적 구조와 마찬가지 문제가 발생하면 수평적 통합의 경우와 마찬가지로 유사도를 이용하여 해결한다.

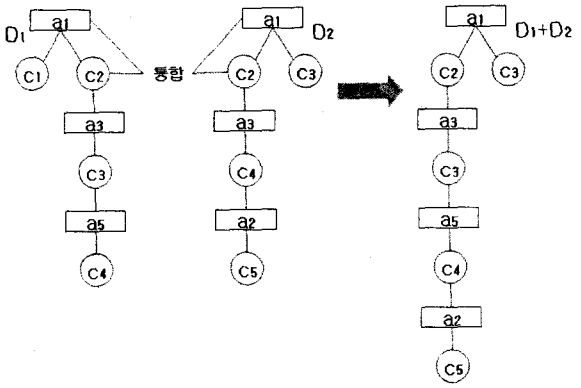

그림 7. 수직적 계층의 통합의 예

4. 서비스

통합된 트리를 이용한 서비스는 크게 두 가지가 있다. 첫 번째는 임의의 키워드에 대한 문서의 순위를 표현해 주는 서비스이며, 두 번째는 선택된 문서에 대해서 더 구체적(Specific)으로 설명한 문서와 더 일반적(General)으로 설명한 문서는 어떤 것들이 있는지 계층적으로 표현해주는 서비스이다.

4.1 키워드를 이용한 문서의 순위

문서 d_1에서 나타난 객체가 사용하고 있는 속성의 수가 문서 d_2의 같은 객체가 사용하고 있는 속성의 수보다 적을 때 해당 객체에 대하여 d_1보다 d_2가 더 비중이 있다고 한다. 이를 이용하여 그림 8.과 같이 문서 순위를 표현할 수 있다. C_5로 문서를 검색할 때 D_1과 D_2에서, C_5가 가질 수 있는 속성의 수로써 검색 순위를 $D_1 < D_2$ 로 표현할 수 있다.

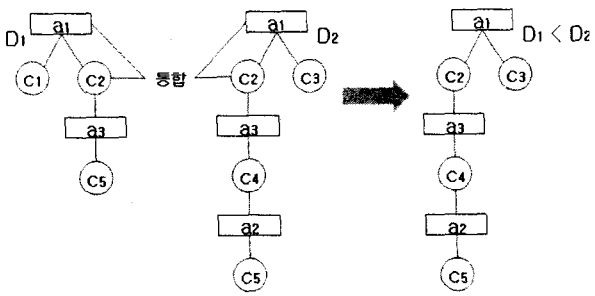

그림 8. 검색어 C_5로 검색할 때 문서의 랭킹 순위 예

4.2 선택된 문서를 이용한 계층적 문서 구조

계층적 문서 구조를 표현하기 위해서는 문서 간의 상위, 하위, 및 동등 관계를 이용한다. 현재 문서의 root 노드와 leaf 노드의 객체를 타 문서와 비교하여 현재 문서의 leaf 노드의 객체의 깊이가 타 문서보다 더 깊고 같은 트리 상에 위치해 있다면 현재 문서는 타 문서 보다 더 일반적이다.

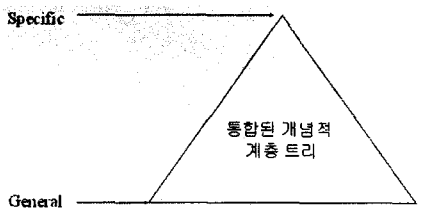

그림 9. 통합된 트리 상의 General and Specific

만일 현재 문서의 깊이가 타 문서의 깊이와 유사하고 문서 간의 같이 사용된 객체와 속성이 존재한다면 두 문서는 서로 동등 관계에 있다. 이러한 관계를 바탕으로 선택된 문서의 계층 구조로 문서를 표현할 수 있게 된다.

5. 성능평가

현재 존재하는 검색 엔진 중 선택된 문서를 이용하여 계층적 관계를 표현해 주는 엔진은 CiteSeer뿐이다. 이런 점이 본 논문과 관련이 있어 CiteSeer와 비교 평가해 보았다. 평가 방법은 시스템 지향적 방법을 이용한 평균 재현율과 평균 정확률의 비교이다.

$$\text{평균재현율} = \frac{\sum_{i=1}^{n}(\text{검색된 상위 적합 문서수} + \text{검색된 하위 적합 문서수})}{\sum_{i=1}^{n}(\text{상위 적합 문서수} + \text{하위 적합 문서수})}$$

$$\text{평균정확율} = \frac{\sum_{i=1}^{n}(\text{검색된 상위 적합 문서수} + \text{검색된 하위 적합 문서수})}{\sum_{i=1}^{n}(\text{검색된 상위 문서수} + \text{검색된 하위 문서수})}$$

6. 결론

이 논문에서는 계층적 개념 트리를 이용하여 키워드에 대한 순위 및 선택된 문서에 대한 계층적 문서 표현에 대한 방법을 제시하였다. 이 방법을 통해 다음과 같은 이점을 얻을 수 있다.

- 연구 논문에 관련된 자료 검색 시 연구자가 참고 논문에 적지 않았던 논문 까지 문서의 계층 구조로 검색 가능하다.
- 새로운 방법의 문서의 순위 시스템을 제안했다.
- 계층적 개념 트리로 표현하였기 때문에 연구 논문을 바탕으로 요약된 타 문서의 검색도 용이해 졌다.

이런 문서의 통합된 계층적 개념 트리의 표현은 검색을 이용하는 사용자들에게 연구 문서와 연계된 타 문서를 구조적으로 보여 줌으로써 연구 문서와 관련된 문서를 검색하는 시간을 줄여주며, 해당 연구 문서와 관련된 자료를 많이 보여 줌으로 써 좀 더 빨리 이해할 수 있게 도움을 준다.

참고 문헌

[1] http://www.google.co.kr

[2] Jan van Eijck, Joost Zwarts, Formal Concept Analysis and Prototypes, Setember 23, 2004

[3] Giles, C. L. Bollacker, K, D. Lawrence, S. CiteSeer : An Automatic Citation Indexing System. ACM Conference, Jun 1998

[4] Giles, C. L. CiteSeer : Past, Present, and Future, Computer Science Adances in Web Intelligence 2004

[5] 김희수, 조용석, 최익규, 김민구, "문서로부터 계층적 개념 트리 자동 구축", 한국정보과학회 가을 학술발표논문집, Vol.32, No.2, pp.103-105, 2004

[6] http://citeseer.ist.psu.edu CiteSeer

[7] Miyoung Cho, Hanil Kim, Pankoo Kim, A New Method for Ontology Merging based on Concept using WordNet, Proceedings in ICACT, pp.1573-1576, 2006