

하이퍼 그래프 기반 Temporal RDF 모델링 기법

이태휘*, 임동혁**

*한국전자통신연구원

**호서대학교 컴퓨터공학과

e-mail : taewhi@etri.re.kr, dhim@hoseo.edu

A Hypergraph-based Modeling for Temporal RDF

Taewhi Lee*, Dong-Hyuk Im**

*Electronics and Telecommunications Research Institute

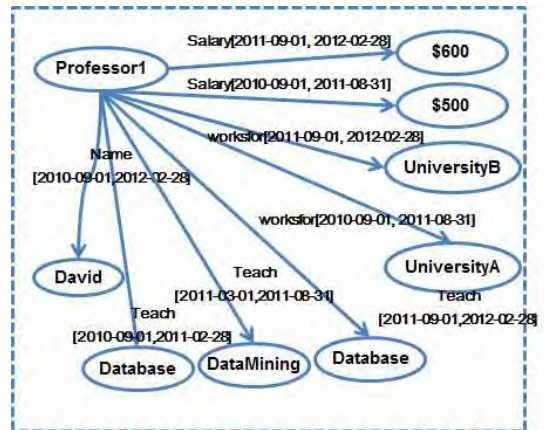
**Dept of Computer Engineering, Hoseo University

요 약

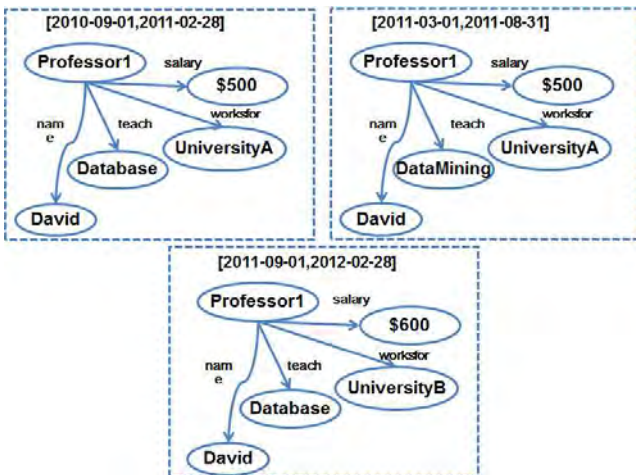
RDF 데이터에 대한 시간 속성에 대한 연구는 트리플의 속성에 시간을 부여하는 방법이 많이 사용되고 있다. 하지만 트리플마다 시간 속성을 부여하는 방법은 저장 및 관리 측면에서 비효율적이다. 본 논문에서는 하이퍼그래프 기반의 RDF 시간 속성 모델링 방법을 제안한다. 하나의 트리플마다 시간 속성을 부여하는 것이 아닌 여러 개의 트리플을 하나의 하이퍼 간선으로 연결하여 시간 속성을 부여하는 방법으로 기존 방법보다 RDF 데이터가 가지는 의미에 적합하며 직관적으로 이해하기가 쉽다. 또한 시간 속성 RDF에서 지원해야 하는 시간 관계를 하이퍼그래프의 여러 속성을 이용하여 처리할 수 있는 장점을 가지게 된다.

1. 서론

RDF와 링크드 데이터와 같은 트리플 데이터는 인간뿐만 아니라 기계가 이행할 수 있는 형태의 시맨틱 데이터를 구축함으로써 분산된 웹 환경에서 이질적인 데이터 간의 통합과 재사용의 목적을 가진 시맨틱 웹의 궁극적 목적을 이루기 위한 기술적 수단이며, 시맨틱 데이터 환경을 이루기 위한 기반 기술이다[1]. 하지만 RDF는 현실 세계를 모델링하기 때문에 현실 세계의 상태가 바뀌면 RDF 데이터도 그에 맞추어 변화를 반영해야 한다[2]. 일반적으로 시간 데이터베이스는 시간 스탬프에 초점을 두고 버전 관리는 특정 스냅샷에 초점을 두는 차이점을 가지며, 시간 속성이 변화(valid time)에 초점을 두는 반면 버전 관리는 특정 스냅샷을 유지하는데 초점을 두는 차이점을 가진다.



(그림 2) 시간 속성 표현과 정규화



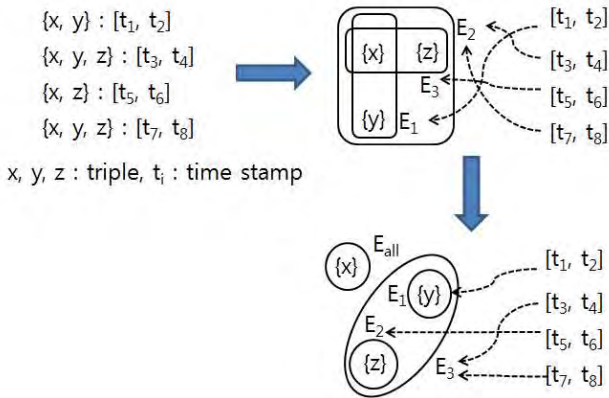
(그림 1) RDF 데이터의 시간적 변화

시간 변화를 고려하는 RDF 데이터를 설명하기 위해 교수1에 대한 RDF 데이터가 그림 1과 같이 변화한다고 하자. 이를 쉽게 모델링하는 방법은 버전 관리 시스템의 가장 일반적인 방법인 각 스냅샷마다 시간 기록을 하고 따로 저장하는 것이다. 이 방법은 특정 시간을 고려하는 질의 처리(하나의 스냅샷에 대한 질의)에 매우 좋은 성능을 보일 수 있으나 수많은 시간 속성을 가진 데이터들을 중복으로 가지게 된다(Professor1 name David 트리플은 3번 중복). 또한 하나의 스냅샷이 아닌 여러 스냅샷에 대한 질의(Temporal Query)를 수행할 때 성능이 저하된다. 따라서 [3,4]에서는 술어에 시간 정보(interval time)을 기입하고 시간 속성의 포함관계에 따른 시간 정규화(Time Normalization)를 제안하였다. 그림 2는 시간 정규화를 사용한 RDF 데이터를 보여준다. 시간 정규화란 시간 범위

에 따른 술어의 분리 혹은 합병을 의미한다. 이와 같이 시간 속성에 정규화를 사용하여 특정 트리플의 유효한 시간을 표현할 수 있다. 또한 시간 속성에 대해 합병을 하였기 때문에 여러 스냅샷에 걸쳐 질의(시간 속성 질의 예: Professor1의 2010-09-01부터 2011-08-31까지 근무한 학교[3])가 쉽게 처리된다.

2. 하이퍼그래프 기반 모델링 기법

[3,4]에서 제안한 모델은 저장 및 질의 처리에 있어 문제점을 가지고 있다. 예를 들면, 트리플(Professor1 teach Database)는 똑같은 내용을 가지지만 다른 시간 속성을 가지며 이와 반대로 몇몇 트리플들은 같은 시간 속성을 가지지만 트리플이 다르다. 본 논문에서는 이러한 문제점을 해결하기 위해 하이퍼그래프 기반 시간 속성 트리플 모델링 기법을 제안한다. 일반 그래프가 한 쌍의 노드와 한 개의 간선이 연결되어 있는데 반하여 하이퍼 그래프는 여러 개의 노드들이 하나의 간선에 연결될 수 있다. 따라서 본 논문에서는 기존의 시간 속성을 트리플 마다 부여하는 것이 아니고 하이퍼 에지에 부여하는 모델링을 제안한다. [5]에서는 하이퍼그래프 기반 버전 관리 시스템을 제안하였으나 본 연구는 시간 속성을 다룬다는 차이점이 있다. 그림 3은 하이퍼그래프 기반 시간 속성 RDF 데이터 모델링을 보여준다. 트리플들이 시간에 따라 변화하면 시간 속성 값을 하이퍼간선에 부여하게 된다.

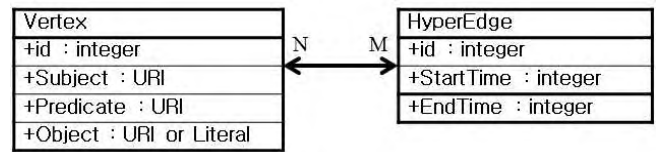


(그림 3) 하이퍼그래프 기반 모델링

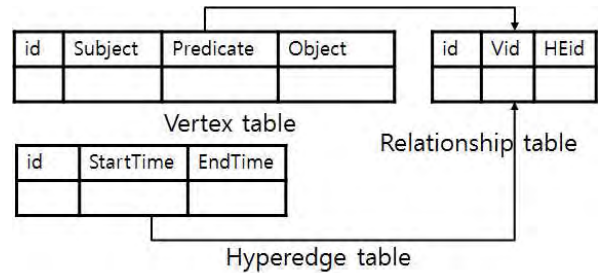
시간 RDF 데이터에서도 특정 시간에 포함되는 트리플들을 하나의 하이퍼 간선으로 그룹화하면 특정 시간 질의어를 빠르게 처리할 수 있다. 하이퍼그래프 기반 방법의 장점은 트리플이 속한 하이퍼간선 레이블(타임 속성)의 합집합을 통해 시간 속성 정규화를 표현할 수 있다는 것이다. 예를 들면, 그림 3에서 트리플 x는 모든 시간에 존재한다. 하이퍼그래프 모델에서 접근을 하면 x 노드는 모든 하이퍼간선에 존재하므로 그림3의 아래와 같이 새로운 하이퍼간선을 만들어 줄 수 있다.

3. 하이퍼그래프 표현을 위한 저장 기법

하이퍼그래프를 표현하는 방법은 인접 행렬을 이용하는 방법과 이분 그래프를 이용하는 방법이 있다. 인접 행렬을 이용하는 방법은 빈 공간을 많이 생성하는 문제점이 생기므로 본 연구에서는 이분 그래프 방법을 사용한다. 그림 4는 본 논문에서 제안하는 이분 그래프 방법의 하이퍼 그래프 클래스 다이어그램을 보여준다. 하이퍼그래프는 2개의 클래스, 노드와 하이퍼간선으로 구성되면 두 클래스 사이에 다대다 대응 관계가 성립하게 된다. 그림 5는 클래스 다이어그램을 바탕으로 관계형 데이터베이스 스키마 구조를 보여준다. 이분 그래프 방법을 사용하였기 때문에 매핑 테이블을 별도로 사용하게 된다.



(그림 4) 하이퍼그래프 표현의 클래스 다이어그램



(그림 5) 하이퍼그래프 관계형DB 스키마

시간 속성 RDF 데이터는 일반적으로 트리플들의 BEFORE, DURING, EQUAL, FINISHES, MEETS, OVERLAPS, STARTS 등의 시간적 관계를 계산할 수 있어야 한다. 본 논문에서는 하이퍼그래프의 속성을 이용하여 이를 계산할 수 있다. 예를 들면, 선후 관계(BEFORE)는 하이퍼 에지의 StartTime을 통한 선후 관계를 비교하여 알 수 있으며, 교차 관계(OVERLAPS)는 두 하이퍼 에지의 교차 여부로 알 수 있다. 또한 동등관계(EQUAL)은 같은 하이퍼간선을 사용하느냐로 판단할 수 있게 된다.

4. 결론 및 향후 과제

본 논문에서는 시간 속성을 가진 RDF 데이터를 하이퍼 그래프로 표현하는 방법에 대해 제안하였다. 트리플마다 시간 속성을 부여하는 것이 아닌 같은 시간대의 여러 개의 트리플을 하나의 하이퍼 간선으로 모델링할 수 있음을 보였다. 또한 하이퍼그래프가 가지는 속성을 이용하여 시간적 관계를 처리할 수 있음을 보였다.

향후 과제로는 하이퍼그래프를 이용하여 시간 속성을 지원하는 SPARQL 질의 처리[6,7]가 있으며 특히 대용량 데이터에 대한 시간 속성 처리에 대해 빅데이터 처리 기술을 활용하여 효율성을 높이고자 한다.

Acknowledgment

이 논문은 2014년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구(No. NRF-2014R1A1A1002236)이며, ETRI R&D 프로그램 ("듀얼모드 배치-쿼리 분석을 제공하는 빅데이터 플랫폼 핵심 기술 개발, 15ZS1400")의 일환으로 수행됨.

참고문헌

- [1] C. Bizer et al. "Linked Data - The Story So Far" International Journal on Semantic Web and Information Systems, 5(3), 2009
- [2] G. Flouris et al. "Ontology changes: classification and survey", The Knowledge Engineering Review, 23(2), 2008
- [3] C. Gutierrez et al. "Introducing Time into RDF" IEEE Transaction on Knowledge Engineering and Data Engineering, 19(2), 2007
- [4] A. Pugliese, O. Udrea, V.S. Subrahmanian, "Scaling RDF with Time", In Proc of WWW, 2008
- [5] Dong-Hyuk Im et al. "A Hypergraph-based Storage Policy for RDF Version Management System" In Proc of ICUIMC, 2012
- [6] Fabio Grandi, "T-SPARQL: a TSQL2-like Temporal Query Language for RDF", ADBIS, 2010
- [7] Jinhyun Ahn et al. "A design of Federated SPARQL Engine for Temporal Query Processing", In Proc of World IT 2015 Congress, 2015