트리 구조 데이터의 의미 기반 라벨 리스트 표현 정보 추출

백주련O
OO평택대학교 데이터정보학과
e-mail: jrpaik@ptu.ac.krO

Semantic-Based Label Lists Represented Information Extraction from Tree Data

Juryon PaikO
ODept. of Digital Information & Statistics, Pyeongtaek University

● 요 약 ●

이형 데이터 간의 정보 전송과 교환을 가능하게 하는 유연한 트리 구조의 특성은 인터넷 및 IoT 환경에서의 대량 데이터 저장 · 전송 · 교환 등에 있어서 XML이나 JSON에서 주요하게 사용된다. 사용성에 있어서는 수월한 반면에, 감추어져 있는 가치있는 정보들을 트리 구조의 대량 데이터들로부터 찾아내는 것은 일반 단순 구조의 데이터에 비해서 훨씬 어려우며 복잡하고 난해한 문제들을 발생시킨다. 이는 트리가 갖는 계층 구조 때문이다. 본 논문에서는 계층 구조를 갖는 대량 트리 데이터들을 보다 단순한 리스트 구조로 변형한 후 해당 구조로부터 가장 자주 발생하는 유용한 정보들을 추출하는 방법을 제시한다.

키워드: 트리데이터(tree data), 리스트 표현(list representation), 시맨틱 트리마이닝(semantics tree mining)

I. Introduction

과거 웹 데이터 범람 시 웹 데이터 저장·전송·교환의 표준으로 사용되었던 XML처럼, 현재의 IoT 환경에서 센서 데이터들의 저장·전송·교환을 위해 이형 데이터 처리에 용이한 JSON이 주로 사용된다. XML과 JSON의 공통점은 데이터 구조로 모두 트리 구조를 갖는다는 것이다. 데이터 종류에 상관없이 저장, 전송 및 교환에 있어서 유연성을 제공하기 때문이다. 그러나, 사용의 용이성은 분석의 복잡성을 초래한다. 평면적인 구조의 데이터보다 복잡한 트리 구조는 특유의 계층 구조로 인해 유용하고 숨겨진 정보를 추출하는 데이터마이닝 분야에서는 어려운 대상으로 여겨진다. 본 논문은 대량의 트리 구조 데이터들을 시맨틱(의미)을 반영하면서도 트리보다는 단순한 리스트 구조로 변형한 후 드러나지 않지만 자주 발생하면서 유용한 정보들을 추출하는 방법을 제시하고자 한다.

II. Tree Mining

트리로 구조화된 다양한 종류의 비정형 데이터들로부터 효용성이 높지만 숨겨져 있는 정보들, 빈번하게 발생하는 서브트리로 간주되는 해당 부분들을 트리 데이터로부터 직접 추출하는 것은 난해하고 어려운 과정이다. 일반적으로, 과정과 연산이 수월하도록 트리 구조를 인접리스트(adjacency list)[1], 인접매트리스(adjacency matrix)[1], 또는 스트링(string)[2] 등을 사용하여 원정보의 손실 없이 변환 저장 후 조인을 포함한 다양한 연산을 수행하여 서브트리들을 유도하는 과정을 거친다. 본 연구는 기존 제시된 리스트 표현법[3]에 따라 트리 데이터들을 변환 저장 후 의미를 고려한 유사성을 계산하여 서브트리를 유도하는 마이닝 수행 방법을 제시한다.

III. The Proposed Scheme

트리의 노드들은 라벨 값을 갖는다. 라벨 값들은 실질적으로 트리 문서가 보유하고 있는 내용 즉, 데이터에 해당한다. 트리 DB에 저장될 때, 트리마다 id가 생성되며 각 트리의 노드들 또한 고유의 인덱스 값을 부여받는다. 본 연구에서는 트리 id, 노드 인덱스, 부모 노드 인덱스, 그리고 라벨을 이용하여 시맨틱 사전을 적용한 라벨 리스트를 생성 후 라벨 사전에 저장한다. Fig.1은 간단한 트리 데이터들과 이로부터 생성된 라벨 사전에 대한 도식화이다.

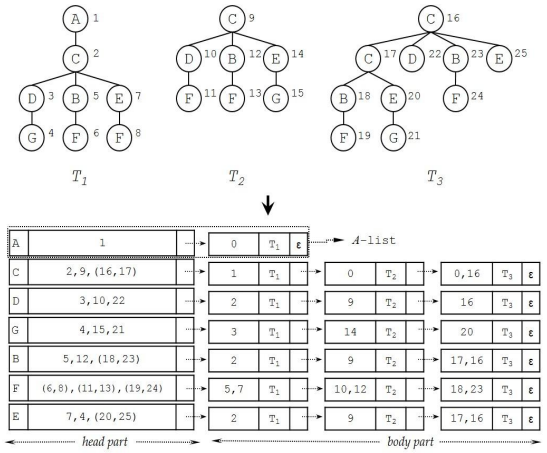

Fig. 1. 트리 데이터들과 라벨 사전

라벨 사전을 사용하여 마이닝 작업을 수행한다. 자주 발생하는 라벨들을 찾아야 하며 이 때 최소지지도임계값을 적용한다. 라벨 리스트를 구성하고 있는 리스트의 멤버 수가 최소지지도 이상이면 해당 라벨은 트리에서 빈번하게 발생하는 빈발 라벨리스트로 분류되지만 그렇지 못한 라벨은 후보 해쉬테이블로 분류된다. 1차로 구해진 빈발 라벨리스트들은 노드 한 개로 이루어진 빈발 서브트리와 동일하다. Apriori 방식에 기반한 트리 마이닝에서는 노드 1개로 이루어진 서브트리들을 하나씩 조인하여 서브트리들을 확장해가면서 해당 서브트리가 최소지지도를 만족하는지 여부를 판단한다. 본 논문에서 제안하는 방식은 조인 방식을 사용하지 않는다. 대신 후보 해쉬테이블을 사용한다. 해쉬테이블에 저장된 라벨리스트는 빈발 라벨리스트를 구성하고 있는 실질적인 노드들의 부모 노드들 또한 마찬가지로 최소지지도 임계값을 만족하는지 여부를 확인하기 위해 사용된다. 부모의 노드 인덱스가 후보 해쉬테이블에 존재한다면 조상-후손 관계에 기인한 서브트리는 절대 빈발서브트리가 될 수 없기 때문이다. 후보 해쉬테이블을 사용하여 빈발 라벨리스트들은 부모에 해당하는 노드 인덱스가 없을 때($\lambda_{노드인덱스}$로 표시)까지 계속해서 라벨 즉 노드를 확장해가며 최소지지도 만족 여부를 확인한다. Fig2.는 그 작업을 수행하는 과정 중 일부를 보인다.

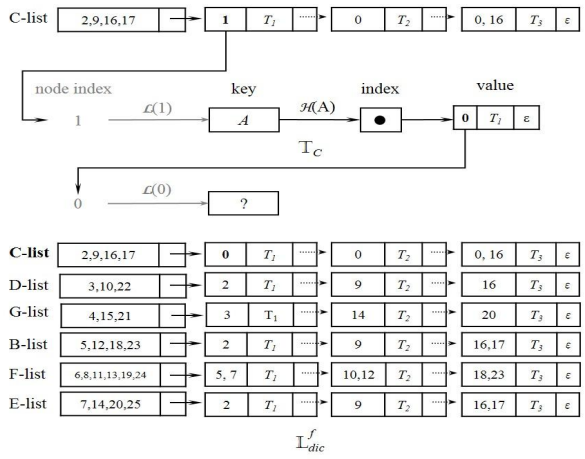

Fig. 2. 루트 노드 (λ)의 발견 과정

빈발 라벨리스트들에 대해서 후보 해쉬테이블을 사용하여 조상 라벨들의 최소지지도 만족 여부 확인이 모두 종료되었다면 빈발 라벨리스트들은 효용성 높은 서브트리들에 대한 정보만 보유하고 있게 된다. 각 라벨 리스트들의 노드정보들을 이용해서 트리 구조로 복구하면 최종적으로 최대 빈발서브트리가 유도된다. Fig.3은 이에 대한 결과이다.

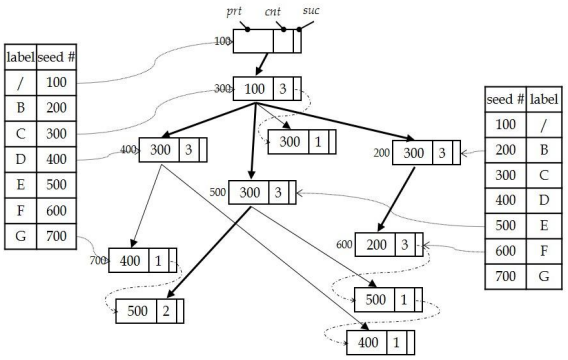

Fig. 3. 루트 노드 (λ)의 발견 과정

논문에서 제시한 방법에 대한 증명과 구체적인 구현 그리고 평가를 현재 수행하고 있다.

REFERENCES

[1] Harmanjit Singh and Richa Sharma, “Role of Adjacency Matrix & Adjacency List in Graph Theory,” International Journal of Computers & Technology , Vol. 3, No. 1, pp. 1021-1035, August 2012, DOI: 10.24297/ijct.v3i1c.2775

[2] Kunihiko Sadakane and Gonzalo Navarro, “Fully-Functional Succinct Trees,” Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2010, pp. 134-149, Austin, Texas, USA, January 2010, DOI: 10.1137/1.9781611973075.13

[3] J. Paik, J. Nam, U. M. Kim, and D. Won, “Fast Extraction of Maximal Frequent Subtrees Using Bits Representation,” J. of Information Science and Engineering, Vol. 25, No. 2, pp. 435-464, March 2009, DOI: 10.1.1.423.292