스키마간 연관성을 이용한 테이블 군집화 기법

조순이° , 이도헌
전남대학교 전산학과
{sicho, dhlee}@dbcore.chonnam.ac.kr

Table Clustering Using Inter-schema Association

Suni Cho° , Doheon Lee
Dept. of Computer Science, Chonnam National University

요 약

업무 데이터 분석을 통한 종합적인 의사결정을 지원할 수 있도록 데이터웨어하우스, OLAP, 데이터마이닝을 적용하려는 기업의 요구가 많아졌다. 그래서 기초 데이터의 이해, 선별, 수집, 가공, 정제가 매우 중요한 과정이나 테이블명 및 속성명이 표준화되어 있지 않고 코드나 시스템 카탈로그와 같은 기본 데이터는 부정확하고 부족하다. 본 논문에서는 거의 스키마 정보에만 의존하여 테이블의 의미적 연관성에 근거한 유사한 특성을 가진 집단끼리 분류하는 대략적인 군집분석 방법을 제안한다. 질의 수행시 사용자가 설정한 임계 거리에 따라 관련된 군집만 검색함으로써 신속한 응답시간을 보장하고, 분석시점에서 다양한 질의에 유연하게 대처할 수 있다는 장점이 있다. 또한 실제 데이터에 본 연구를 적용하여 산출한 군집결과와 사람이 매뉴얼하게 그룹핑한 군집결과와 비교한다.

1. 서론

과거 데이터베이스 시스템의 기능은 비교적 단순 업무를 반복적으로 처리하는 온라인 트랜잭션 처리(On-Line Transaction Processing:OLTP)가 대부분이었다. 하지만 최근 기업은 업무 데이터 분석을 통한 종합적인 의사결정 지원(decision support)을 요구하고 있다. 대규모 데이터를 축적하여 데이터웨어하우스를 구축하고 온라인 분석처리(On-Line Analytical Processing:OLAP)와 데이터마이닝을 적용하여 필요한 정보를 분석 가공하여 신속하게 제공할 수 있어야 한다. 그런데 데이터 웨어하우스의 원시 데이터에는 관계형 파일(테이블), 네크워크 파일, 계층형 파일, 일반 파일 등이 혼재하여 있어서 데이터에 대한 이해, 선별, 수집, 가공, 정제는 프로젝트 전체 공정의 80퍼센트 이상을 차지할 정도로 매우 어렵고도 중요한 일이다.

이러한 이질적인 데이터베이스에서 효율적인 검색의 문제를 데이터 역공학(Data reverse engineering)을 통해 해결해 보려는 연구들이 있어 왔다[1,2,5,6]. 또, LIEN은 테이블을 다치종속(multivalued dependency)에 의한 제 4정규형으로 무손실 분해를 한 후 계층적 뷰를 구성하는 방법을 제안했고[3] Goldstein은 테이블 사이의 연관성을 표현하는데 있어 시스템 카탈로그나 PL/SQL 같은 명시적 표현만을 사용해야 한다고 주장했다[4]. Chiang은 데이터 포함관계와 키에 의한 의미적 관계에 의해 개념적 스키마를 추론하였고[5], Jahnke는 속성간 연관정도를 부여하는 추론규칙을 제안하였다[6].

그러나 위와 같은 기존 연구들은 다음과 같은 이유로 실제 적용에는 부적합하다. 첫째, 실제 데이터에서는 테이블 및 속성 이름이 표준화되어 있지 않기 때문에, 문자열간의 비교만으로는 서로 동일한 테이블 및 속성을 식별하기 어렵고 둘째, 데이터 역시 코드화되어 있는 경우가 많기 때문에 데이터 값 자체로는 의미를 알기 어렵다. 마지막으로 외래키(foreign key), 주키(primary key), 유일성 제약조건(unique constraint) 등이 시스템 카탈로그에 정확히 빠짐없이 명기되어 있다고 가정하기 어렵다.

따라서 본 논문에서는 이러한 문제점을 극복하기 위해서 실제 현장에서 적용 가능한 다음과 같은 방법을 제시하고자 한다. 첫째, 데이터가 다르더라도 스키마가 유사하면 연관성이 높기 때문에 데이터상의 중복은 고려하지 않는 것이 좋다. 예를 들면 2000년 신입직원 명단과 2001년 신입직원 명단은 데이터상의 중복은 없지만 스키마는 유사하다. 둘째, 테이블 이름, 속성 이름, 데이터 등이 표준화되어 있지 않기 때문에 명확한 계층구조를 얻을 수는 없고, 불확실한 단서들을 가급적 수집하여 연관성을 평가해야 한다. 셋째, 그러나 시스템 카탈로그가 완전하지도 않고, 일반 파일인 경우에는 아예 시스템 카달로그도 없기 때문에 적용하기가 어렵다. 따라서 거의 스키마 정보에만 의존하여 테이블, 파일간의 의미적 연관성에 근거한 유사한 특성을 가진 집단끼리 분류하는 군집분석(clustering)을 하되 대략적인 분류를 하자는데 이 논문의 목적이 있다. 사용자 질의 수행시 관련 군집만 검색함으로써 검색효율을 크게 증대시킬 수 있고, 분석시점에서 다양한 질의에 유연하게 대처할 수 있다는 장점이 있다.

본 논문은 다음과 같이 구성되어 있다. 2절에서는 테이블 사이의 연관성을 결정짓는 각 요소들을 기술하고 예제를 제시하며 3절에서는 스키마 분석을 통한 계층적 군집화를 기술하고 예제를 제시한다. 4절에서는 가중치의 결정 방법을 기술하고 마지막으로 5절에서는 결론 및 추후 연구과제를 기술한다.

2. 테이블 사이의 연관성

테이블 사이의 연관성은 테이블 이름 사이의 연관성, 속성 집합 사이의 연관성과 키 공유 여부에 의한 연관성에 의해 결정된다고 본다. 다음에서 각각에 대한 정의 및 예제를 제시한 후 테이블 사이의 연관성을 정의하고 그 예제를 제시한다.

2.1 테이블 이름 사이의 연관성

[정의 1] 테이블 이름 사이의 연관성

두 개의 R,S 문자열간의 용어 유사도(Term Similarity)를 TS(R,S)라고 하고 테이블 이름 사이의 연관성을 Ψ_T(R,S)라고 했을 때, 테이블 이름 사이의 연관성은 다음과 같이 정의된다.

- Ψ_T(R,S) = TS(Table-Name(R), Table-Name(S))
  - TS(R,S)는 두 테이블 이름이 동일하거나 동의어(synonym)인 경우는 1.0, 부분 문자열을 공유한 경우나 합의관계(hypernym/hyponym)인 경우는 0.5로 한다.

[예제] A학교 업무 데이터베이스에 테이블 CSTUDENT, PSTUDENT가 있다. 두 테이블 이름 사이의 연관정도를 측정해 본다.

CSTUDENT, PSTUDENT는 STUDENT라는 연속적인 부분 문자

열을 공유하므로 위의 정의에 의해 테이블 이름 사이의 연관성은 다음과 같다.
Ψ_T(CSTUDENT, PSTUDENT) = 0.5

2.2 속성 집합 사이의 연관성
스키마 사이의 연관성은 두 테이블이 얼마나 많은 속성을 공유하는가를 측정한다. 정의 2에서 두 속성간의 속성 유사도를 먼저 정의한 후 정의 3에서 속성 집합 사이의 연관성을 정의한다.
[정의 2] 두 속성간의 속성 유사도
두 개의 속성 A, B에 나타난 속성값의 집합을 각각 V(A), V(B)라고 하고 속성 A, B 사이의 유사도를 AS(A,B)라고 했을 때, 두 속성간의 속성 유사도(Attribute Similarity)는 다음과 같이 정의된다.
· AS(A,B) = MAX(TS(A,B), (2*|V(A)∩V(B)|)/(|V(A)|+|V(B)|))
즉, 속성명에 대한 용어 유사도와 나타난 속성값의 공유정도 중 어느 한쪽이라도 충분히 높은 연관성을 보이면 두 속성은 연관성이 높다고 평가한다.
[예제] 테이블 CSTUDENT, PSTUDENT 스키마가 아래와 같을 때, 두 속성 CSTUDENT의 STUCODE와 PSTUDENT의 STUCODE간의 속성 유사도를 측정해 본다. 단, 각 테이블에서 속성 STUCODE는 A학교 학생의 학번으로써 주키이다.

CSTUDENT = (STUCODE CLASSIFYSCHOOL SCHOOLYEAR SCHOOLNAME BRANCHNAME CLASSNAME TASNUMBER TEACHERNAME SPECIAL STATE)
PSTUDENT = (STUCODE NAME SEX REGISTERNUM BLOODABO BLOODRH PROTECTOR)

두 속성 CSTUDENT의 STUCODE와 PSTUDENT의 STUCODE의 문자열을 비교하면 완전 일치하므로 TS(STUCODE, STUCODE)는 1이다. 또한 속성 STUCODE는 A학교 학생의 학번으로써 속성값이 완전 중복되므로 (2*|V(STUCODE)∩V(STUCODE)|)/(|V(STUCODE)|+|V(STUCODE)|)도 1, 따라서 두 속성간의 속성 유사도는 정의 2에 적용하면 AS(STUCODE,STUCODE) = MAX(1, 1) = 1이다.

[정의 3] 속성 집합 사이의 연관성
속성 A에 대하여, 최고 속성 유사도를 가진 상대편 테이블의 속성을 MX(A)라고 하고 속성 집합 사이의 연관성을 Ψ_S(R,S)라고 했을 때, 두 테이블 R과 S의 스키마로부터 평가되는 연관성은 다음과 같이 정의된다. 단, r + s 는 두 테이블 R과 S의 속성 개수의 합이다.
· $\Psi_S(R,S) = (\Sigma_i\, AS(A_i, MX(A_i)) + \Sigma_j\, AS(MX(B_j), B_j)) / (r + s)$
즉, 두 테이블이 많은 속성을 공유할수록 높은 값을 갖게 되며, 완전히 동일한 스키마를 가지고 있으면 1, 전혀 공유하는 속성이 없으면 0을 갖게 된다.
[예제] 테이블 CSTUDENT, PSTUDENT 스키마를 이용해 속성 집합 사이의 연관성을 측정해 본다.
테이블 CSTUDENT, PSTUDENT에서 최고 속성 유사도를 가진 속성은 STUCODE이다. 또 CSTUDENT는 10개 PSTUDENT는 7개의 속성을 가지고 있으므로 위의 정의를 적용하면 속성 집합 사이의 연관성은 다음과 같다.
Ψ_S(CSTUDENT,PSTUDENT) = (1+1)/(10+7) = 0.12

2.3 키 공유 여부에 의한 연관성
[정의 4] 키 공유 여부에 의한 연관성
IsKey(A)는 속성 A가 키인 경우 1을, 아닌 경우 0을 갖는다고 하고 키 공유 여부에 의한 연관성을 Ψ_K(R,S)라고 했을 때, 키 공유 여부에 의한 연관성은 다음과 같이 정의된다.
· $\Psi_K(R,S) = \Sigma_{ij}$ MIN(AS(A_i,B_j), (IsKey(A_i)+IsKey(B_j))/2)
즉, 키 공유 여부에 의한 연관성은 두 스키마의 속성 유사도와 키 공유 여부에 의한 유사도 중 최소값을 취한다.
[예제] 테이블 CSTUDENT, PSTUDENT에서 키 공유 여부에 의한 연관성을 측정해 본다.
AS(STUCODE, STUCODE)가 1이고 STUCODE가 각 테이블의 키이므로 (IsKey(STUCODE)+IsKey(STUCODE))/2)도 1, 따라서 키 공유 여부에 의한 연관성은 위 정의를 적용하면 다음과 같다.
Ψ_K(CSTUDENT,PSTUDENT) = MIN(1,1) = 1

2.4 테이블 사이의 연관성
[정의 5] 테이블 사이의 연관성
임의의 두 테이블 R, S 스키마가 각각 R = (A_1, ..., A_r), S = (B_1, ..., B_s)로 구성되어 있고 테이블 사이의 연관성을 Ψ(R,S)라고 했을 때, R과 S의 연관성은 다음과 같이 정의된다. 단, A_1, ..., A_r B_1, ..., B_s은 속성 집합이다.
· $\Psi(R,S) = \omega_T\Psi_T(R,S) + \omega_S\Psi_S(R,S) + \omega_K\Psi_K(R,S)$
이 때, ω_T, ω_S, ω_K는 각 요소의 가중치를 표시하며 가중치의 결정 방법에 대해서는 4장에서 기술한다. 또한 테이블 이름 사이의 연관성, 속성 집합 사이의 연관성, 키공유 여부에 의한 연관성은 각각을 별도로 취급하여 계층 군집화를 할 수도 있다.
[예제] 테이블 CSTUDENT, PSTUDENT 사이의 연관정도를 측정해 본다. 단, 가중치 ω_T, ω_S, ω_K는 각각 1로 한다.
2.1 2.2 2.3의 예제에서 산출된 각각의 연관성을 위의 정의에 적용하면 테이블 CSTUDENT와 PSTUDENT 사이의 연관성은 다음과 같다.
Ψ(CSTUDENT, PSTUDENT)= 0.5 + 0.12 + 1 = 1.62

3. 스키마 분석을 통한 계층적 군집화
군집화란 물리적 혹은 추상적 객체를 비슷한 객체군으로 그룹핑하는 과정이다. 군집분석의 목적은 각 객체가 군집의 갯수, 내용, 구조 등이 사전에 정의되지 않은 상황하에서 객체 사이의 유사성(또는 거리)에 근거하여 식별함으로써 전체 다변량 자료의 구조를 파악하고, 군집의 형성과정과 그 특성, 그리고 식별된 군집간의 관계 등을 체계적으로 연구, 분석하는 것이다. 군집유형은 유사성이나 비유사성의 정의, 군집형태에 따라 상호배타적(disjoint) 군집, 계층적(hierarchical) 군집, 중복(overlapping) 군집, 퍼지(fuzzy) 군집 등이 있는데 본 논문에서는 계층적 군집방법을 사용한다. 계층적 군집이라 함은 한 군집이 다른 군집의 내부에 포함되나 군집간의 중복이 허용되지 않고 가계보와 같이 군집들이 매단계 계층적인 구조를 이룬다. 계층적 군집 방법에는 가까운 개체들끼리 묶어나가는 병합적(agglomerative) 방법과 먼 개체들을 나누어 가는 분할적(divisive) 방법이 있다. 일반적으로 분할하는 것보다 병합하는 쪽이 계산 비용이 적게 들기 때문에 병합적 방법으로 대부분의 알고리즘들이 개발되어 있다. 본 논문도 병합적 방법을 사용하여 군집들을 계층화 한다.

다음에서 본 논문에서 제안한 테이블 사이의 연관성에 대한 정의를 적용한 후 산출된 군집들이 사람이 매끄럽하게 그룹핑한 군집들과 얼마나 일치하는지를 예제를 통해서 보인다. 군집간 거리는 두 군집 사이의 거리를 각 군집에 속하는 임의의 두 개체들 사이의 거리 중 최단거리를 사용하여 계산한다. 이외에도 최장거리, 평균거리 등을 사용하여 군집간 거리를 계산하는 방법도 있다.
[예제] A학교 업무 데이터베이스에는 다음과 같은 테이블들이 있다. 편의상 테이블 CSTUDENT는 1, PSTUDENT는 2, CBODY는 3, CINFECTION은 4, CSTRENGTH는 5, GRSEND는 6으로 표현한다.

표 1 : R, S 테이블 사이의 연관정도 행렬

Ψ(R,S)	1	2	3	4	5	6
1	3.00	1.62	1.65	1.65	1.77	1.06
2	1.62	3.00	1.05	1.06	1.10	1.07
3	1.65	1.05	3.00	1.55	1.64	1.04
4	1.65	1.06	1.55	3.00	1.59	1.06
5	1.77	1.10	1.64	1.59	3.00	1.00
6	1.06	1.07	1.04	1.06	1.00	3.00

표 1은 정의 1을 적용하여 계산한 각 테이블 사이의 연관정도를 나타낸 행렬(similarity matrix)이고 표 2는 두 테이블 사이의 거리행렬(distance matrix)이다. R, S 테이블 사이의 거리값 Δ(R,S)는 3에서 테이블 사이의 연관정도 Ψ(R,S)를 빼서 산출한 값이다. 숫자 3은 이 예제에서 테이블 사이의

연관성 중 최대값이다. 예를 들면 CSTUDENT와 PSTUDENT의 거리는 두 테이블의 연관정도인 1.62(2.4 예제에서 산출)를 3에서 뺀 값인 1.38이라고 할 수 있다.

표 2 : R, S 테이블 사이의 거리행렬

Δ(R,S)	1	2	3	4	5	6
1	0.00	1.38	1.35	1.35	1.23	1.94
2	1.38	0.00	1.95	1.94	1.90	1.93
3	1.35	1.95	0.00	1.45	1.36	1.96
4	1.35	1.94	1.45	0.00	1.41	1.94
5	1.23	1.90	1.36	1.41	0.00	2.00
6	1.94	1.93	1.96	1.94	2.00	0.00

표 3 : 1차 병합 후 R, S 거리행렬

Δ(R,S)	1,5	2	3	4	6
1,5	0.00	1.38	1.35	1.35	1.94
2	1.38	0.00	1.95	1.94	1.93
3	1.35	1.95	0.00	1.45	1.96
4	1.35	1.94	1.45	0.00	1.94
6	1.94	1.93	1.96	1.94	0.00

표 2에서 보면 테이블 1과 5는 거리가 1.23으로 가장 가까우므로 첫 번째 병합 대상이 되고 그 결과 표 3으로 갱신된다. 2차 병합은 다음 식에서 알 수 있듯이 각각의 거리를 계산해보면 d{(3)(1, 5)}와 d{(4)(1, 5)}가 1.35로써 군집 1,5와 가장 가까우므로 1,5,3,4가 병합되어 새로운 군집이 되는 것이다. 그다음의 과정도 마찬가지 방식으로 이루어지면 마지막에서 모든 테이블이 하나의 군집으로 통합된다.

d{(2)(1, 5)} = min{d12, d25} = d12 = 1.38
d{(3)(1, 5)} = min{d13, d35} = d13 = 1.35
d{(4)(1, 5)} = min{d14, d45} = d14 = 1.35
d{(6)(1, 5)} = min{d16, d56} = d16 = 1.94

표 4 : 2차 병합 후 R, S 거리행렬

Δ(R,S)	1,5,3,4	2	6
1,5,3,4	0.00	1.38	1.94
2	1.38	0.00	1.93
6	1.94	1.93	0.00

표 5 : 3차 병합 후 R, S 거리행렬

Δ(R,S)	1,5,3,4,2	6
1,5,3,4,2	0.00	1.94
6	1.94	0.00

그림 1은 가장 유사한 두 개체 군집을 병합함으로써 군집수를 하나씩 줄여 나가는 과정을 나무그림으로 표현한 것이다. 군집간 거리의 차이에 의해 군집들의 계층이 이루어진다. CSTUDENT CSTRENGTH CBODY CINFECTION PSTUDENT GRSEND 순위로 테이블의 유사성을 보여주고 있다. 실제 매뉴얼하게 A학교 업무 데이터베이스를 보면 CSTUDENT CSTRENGTH CBODY CINFECTION은 학생의 건강에 관한 테이블이고 PSTUDENT는 학생 신상에 관한 기초자료 테이블이고 GRSEND는 성적에 관한 테이블이다. 결과적으로 본 논문에서 제안한 테이블 사이의 연관정도의 계산식에 의한 군집이 직관에 의한 군집과 일치한다고 할 수 있다. 사용자는 필요에 따라 임계 거리를 변화시켜 가면서 여러 가지 형태의 그룹핑을 할 수 있다.

4. 가중치의 결정

테이블 사이의 연관정도를 결정하는 가중치는 각 요소의 상대적인 중요도를 반영한다. 가중치 부여 기준은 어떤 요소에 의한 정보를 보다 정확하게 얻을 수 있는가에 달려있으므로, 일반적인 부여기준은 없고, 다음과 같은 단계를 거쳐 결정된다. 첫째, 수작업으로 처리 가능한 개수의 테이블만을 샘플링 하여 직관에 의해 테이블 그룹핑을 한다. 둘째, 모든 가능한 가중치의 조합을 고려한다. 예를 들어, 가중치에 0.0 0.2 0.4 0.6 0.8 1.0의 6가지가 있다고 할 때 ω_T, ω_S, ω_K 각각에 대해서 조합하면 200여 경우를 고려해 볼 수 있다. 셋째, 모든 조합 가능한 가중치를 본 논문의 알고리즘에 적용하여 테이블 그룹핑을 한다. 마지막 단계에서는 수작업에 의한 그룹핑과 알고리즘에 의한 그룹핑을 비교하여 가장 근사한 결과를 가져오는 가중치 조합을 선택한다. 이 가중치의 조합이 특정의 데이터베이스 상황에 가장 적합하다고 볼 수 있다.

5. 결론

이질적인 테이블, 파일들의 기본 데이터가 부족한 상황에서 데이터의 이해, 선별, 수집, 가공, 정제과정은 매우 어렵고도 중요하다. 따라서 본 연구는 거의 스키마 정보에만 의존하여 테이블의 의미적 연관성에 근거한 유사한 특성을 가진 집단끼리 대략적인 군집분석을 하는 방법을 제안하였다. 테이블, 파일간의 연관정도를 측정하기 위해서 테이블 이름 사이의 연관성, 속성 집합 사이의 연관성과 키 공유 여부에 의한 연관성을 정의하였고 그 예제를 제시하였다. 실제 데이터에의 적용에 의해 계층적 군집화 과정을 보임으로써 사람이 매뉴얼하게 그룹핑한 군집 결과와 일치함을 입증하였다. 사용자는 질의를 할 때 임계 거리를 설정하여 해당되는 군집만을 검색함으로써 검색효율을 증대시킬 수 있고, 분석시점에서 다양하게 질의를 변경하여 그 결과를 비교해 볼 수 있다.

추후 테이블 이름과 속성 이름 사이의 함의 관계, 부분문자열 포함관계를 계측할 수 있는 시소로스를 활용하여 연관정도를 좀 더 정량적으로 측정할 필요가 있고 가중치의 결정 방법에 대한 연구도 보강되어야 할 것이다.

6. 참고 문헌

[1]. C. Fahrner and G. Vossen, "A survey of database design transformations based on the Entity-Relationship model, Data Relationship Approach" (1985) 280-286.

[2]. J-L, Hainaut, M. Chandelon, C. Tonneau and M. Joris, "Contribution to a theory of database reverse engineering", Proc. IEEE Working Conf. on Reverse Engineering (1993) 161-170.

[3]. E. Lien, "Hierarchical schemata for relational databases", ACM Trans. Database Syst. 6 (1981) 48-69.

[4]. Robert C. Goldstein, Veda C. Storey, "Data Abstractions: Why and how?", Data & Knowledge Engineering 29 (1999) 293-311

[5]. Roger H.L. Chiang, Terence M. Barron, Veda C. Storey, "A framework for the design and evaluation of reverse engineering methods for relational databases", Data & Knowledge Engineering 21 (1997) 57-77

[6]. J. H. Jahnke, W, Schäfer, and A. Zündorf. "Generic fuzzy reasoning nets as a basis for reverse engineering relational database applications". European Software Engineering Conference, LNCS 1302, Springer Verlag, (1997).

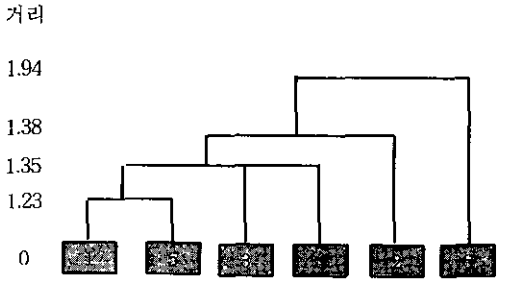

그림 1 : 덴드로그램