다중데이터베이스 마이닝에서 가중치 거리를 이용한 클러스터링

김진현°, 윤성대
부경대학교 전자계산학과
jhkim°@dol.pknu.ac.kr, sdyoun@pknu.ac.kr

A Weight Distance-based Clustering for MultiDatabase Mining

Jinhyun Kim°, Sungdae Youn
Dept. of Computer Science, Pukyong National University

요 약

다중데이터베이스 마이닝에서 하나의 데이터 집합을 형성하는 작업은 많은 부하가 따른다. 그러므로, 본 논문에서는, 가중치 거리를 이용한 클러스터링을 통해 관련성이 높은 데이터베이스를 식별하는 기법을 제안한다. 제안한 기법은 빈발한 항목으로 구성된 데이터 집합을 생성하여 데이터베이스 사이의 유사성과 거리를 측정하고, 데이터베이스간의 거리에 대한 식별성을 향상시키기 위하여 최다 빈발항목에 대한 비교 연산을 통해 가중치를 산출한다. 그리고 성능평가를 통하여 제안한 기법이 Ideal&Goodness 기법보다 다중데이터베이스의 트랜잭션 데이터베이스에 대한 식별 능력이 우수함을 알 수 있었다.

1. 서 론

대규모 데이터베이스로부터 유용한 정보나 의미 있는 사실을 추출하기 위한 연구분야를 데이터 마이닝(Data Mining)이라고 한다. 데이터 마이닝에서 사용되는 기법으로는 연관 규칙(Association Rule), 통계적 기법(Statistical Technique), 의사결정 트리(Decision Tree), 신경망(Neural Network), 유전자 알고리즘(Genetic Algorithm)등이 있다[1]. 이러한 연구 결과는 데이터 분석이나 요약을 위해 하나의 대형 데이터 집합을 필요로 한다. 이는 유용한 마이닝 알고리즘 실행 시간을 지연시킬 뿐 만 아니라, 알고리즘 동작에도 영향을 미치게 된다. 현실 세계의 대부분의 응용 부분에서는 지역적인 자치성을 고려하고, 복수 개의 데이터베이스를 사용하려는 경향에 따라 다중데이터베이스에 대한 관심이 커져가고 있다. 다중데이터베이스 마이닝을 위한 선처리(Preprocessing) 작업에 대한 연구로써는 질의어(Query)의 술어(Predicate)에 속한 테이블명과 연산자를 통한 관련성 측정[2], 트랜잭션 데이터베이스에 속한 모든 항목들에 대해서 비교 분석하는 유사성 측정 방법(Ideal&Goodness)[3]이 있다.

다중데이터베이스에 대한 질의어는 효율적인 시스템 성능을 달성하기 위해 포괄적인 질의 최적화를 필요로 한다. 질의어의 술어를 이용하는 방법은 다중데이터베이스가 갖는 특징 중 하나인 지역 자치성 때문에 응용분야에 독립적으로 수행되지 못하고, 일반화되어 있지 못하다는 단점이 있다. [3]에서는 클러스터링 방법에 대한 일반화는 고려되었으나, 항목집합 발생률이 유사한 경우에 대해서는 항목간의 식별이 불가능하다는 단점이 있다.

이러한 단점을 보완하고, 일반성을 최대한 고려하여 다중데이터베이스 마이닝을 위한 선처리 작업 단계로, 관련성이 높은 데이터베이스를 식별하기 위하여 거리에 대한 가중치를 이용한 클러스터링 기법을 제안한다.

2. 관련 연구

Agrawal *et al.*[4]에 의해 처음으로 소개되었고, 항목집합의 연관규칙 발견을 위해 사용되는 빈발 항목집합에 대해 살펴보면 다음과 같다.

$I = \{a_1, a_2, a_3, \ldots, a_t\}$를 항목(item)들의 전체 집합이라 한다. 전체 집합에 속한 항목들로 이루어진 집합을 항목집합(itemset)이라 하며, 항목으로 이루어진 전체 집합의 부분집합이 된다. 집합 I 에 속한 항목들로 구성된 집합 X가 트랜잭션 T에 포함되면, 즉 $X \subseteq T$ 관계이면 T는 X를 지지한다(support)라고 한다. X를 지지하는 D에 있는 모든 트랜잭션들의 개수를 지지도(support)라고 하며, *supp(X)*로 나타낸다. 만일 사용자가 지정한 최소지지도(*minsup*)에 대하여 $minsup \leq supp(X)$ 이면, 항목집합 X는 빈발하다라고 하고, X를 빈발 항목집합(frequent 혹은 large itemset)이라고 정의하며, *k*개의 항목으로 이루어진 빈발 항목집합을 빈발 *k*-항목집합(frequent *k*-itemset) 이라고 한다. 최소지지도는 사용자가 미리 지정한 값이며 최소지지도를 사용하는 이유는 항목에 대해 발생이 관심 수준 이상으로 빈발한지를 고려하기 위해서이다. 가령, 최소지지도가 0.5이고 하나의 데이터베이스에서 모든 트랜잭션의 발생횟수가 4번이라면, 동일한 항목이 2번 이상 발생한 경우에 대해 빈발하다라고 할 수 있다.

3. 알고리즘의 제안

3.1 빈발 1-항목집합 데이터베이스 유사성

다중데이터베이스 집합을 $MultiDB=\{D_1, D_2, D_3, \ldots, D_m\}$이라 하면, 데이터베이스 D_i ($i \in [1..m]$) 는 MultiDB의 부분집합이고, 트랜잭션 T는 D_i 의 부분집합이다.

D_1={(a, b, c, d);(a, b, c);(a, b, c, d, e);(a, e)}
D_2={(a, c, d);(a, b, c);(a, c);(b, c);(a, c, e)}
D_3={(a, b, d);(b, c, d);(b, c, d, e);(a, b, e);(a, d)}
D_4={(a, d);(d, e);(a, b);(a, b, d, e);(a, b, c, d)}
D_5={(d, g, h);(h, i);(d, h, j);(h, i, j)}
D_6={(g, h, i);(g, j);(g, h, i);(h, i)}

위와 같은 6개의 데이터베이스가 주어졌을 때, 각 데이터베이스는 여러 개의 트랜잭션으로 이루어져 있고, 세마콜론으로

나누어진 트랜잭션은 여러 개의 항목들을 포함한다.

표 1은 본 논문에 사용된 알고리즘 및 수식에 대한 파라미터들을 요약한 것이고, 본 논문에서 제안된 알고리즘 1은 D'_i를 생성하는 방법이다.

표 1. 다중데이터베이스 클러스터링 연산을 위한 파라미터

기호	기호 의미
L_1	빈발 1-항목집합
$L_1(D_i)$	L_1 으로 구성된 데이터베이스
D'_i	L_1 으로 구성되고, 빈발 수를 포함한 데이터베이스
a_i	항목 ($i \in [1..n]$)
card	카디널러티
freq_count	지지도
D'_{i_Max}	$L_1(D_i)$ 에서 항목 발생 빈도수가 가장 큰 항목 = 최다 빈발항목
$D'_i[a_i]$	$L_1(D_i)$ 에 속한 항목
C_k	콜러스터 C_k

```
입력 : Di ← 다중데이터베이스중의 하나를 선택
출력 : D'i ← 선택된 데이터베이스로부터 생성
1) procedure gen_freq_1_itemsets(Di)
2) begin
3)   tr_count = 0;  // 트랜잭션 카운트
4)   freq_count = 1; // 항목 발생 빈도수
5)   L1 = {∅};
6)   for each transaction tr ∈ Di do begin
7)     tr_count = tr_count + 1;
8)   forall item ai in tr do begin
     // ai는 tr에 속한 항목
9)     forall item f in L1 do begin
       // f는 L1에 속한 항목만을 나타냄
10)      if ai = f then
11)        f.freq_count = f.freq_count + 1;
12)      else
13)        ai.freq_count = 1;
14)    end
15)    L1 = L1∪{ai[freq_count]}; // 항목집합
16)  end
17)  end
18)  L1(Di)={ai∈L1|ai.freq_count/tr_count>=min_support}
19) L1(Di) ← 빈발 수에 대하여 내림차순으로 정렬
20) end
```

알고리즘 1. 빈발 1-항목집합 데이터베이스(D'_i) 생성

위에서 예시한 6개의 데이터베이스에 대해 빈발 1-항목집합과 항목에 대한 빈발 수를 갖는 빈발 1-항목집합 데이터베이스($L_1(D_i)$)를 생성하면 다음과 같다.

D'_1 = {a[4], b[3], c[3], d[2], e[2]} → $L_1(D_1)$={a, b, c, d, e}
D'_2 = {c[5], a[4], b[2]} → $L_1(D_2)$ = {a, b, c}
D'_3 = {b[4], d[4], a[3], c[2], e[2]} → $L_1(D_3)$ = {a, b, c, d, e}
D'_4 = {a[4], d[4], b[3], e[2]} → $L_1(D_4)$ = {a, b, d, e}
D'_5 = {h[4], d[2], i[2], j[2]} → $L_1(D_5)$ = {d, h, i, j}
D'_6 = {g[3], h[3], i[3]} → $L_1(D_6)$ = {g, h, i}

[정의 1]

빈발 1-항목집합 데이터베이스 유사성(줄여서, '유사성'으로 한다.) :

$$\boldsymbol{sim}(L_1(D_i), L_1(D_j)) = \frac{card\, L_1(D_i) \cap L_1(D_j)}{card\, L_1(D_i) \cup L_1(D_j)} \quad (1 \le i \le n,\ 1 \le j \le n)$$

정의 1을 사용하여 유사성을 연산하고, 결과에 대한 예는 다음과 같다.

$$sim(L_1(D_1), L_1(D_1)) = \frac{card\, L_1(D_1) \cap L_1(D_2)}{card\, L_1(D_1) \cup L_1(D_2)} = \frac{\{a,b,c\}}{\{a,b,c,d,e\}} = \frac{3}{5} = 0.6$$

$$sim(L_1(D_1), L_1(D_1)) = \frac{card\, L_1(D_1) \cap L_1(D_1)}{card\, L_1(D_1) \cup L_1(D_1)} = \frac{\{a,b,c,d,e\}}{\{a,b,c,d,e\}} = \frac{5}{5} = 1$$

. . .

3.2 빈발 1-항목집합 데이터베이스간의 거리, 가중치 거리, 가중치 거리 평균, 측정값

[정의 2] 빈발 1-항목집합 데이터베이스간의 거리 :

$$\boldsymbol{dist}(L_1(D_i), L_1(D_j)) = 1 - sim(L1(D_i), L1(D_j)) \quad (1 \le i \le n,\ 1 \le j \le n)$$

빈발 1-항목집합 데이터베이스간의 거리를 연산하면 다음과 같다.

$dist(L_1(D_1), L_1(D_1))=1-1=0$, $dist(L_1(D_1), L_1(D_2))=1-3/5=2/5$
$dist(L_1(D_1), L_1(D_3))=1-1=0$, $dist(L_1(D_1), L_1(D_4))=1-4/5=1/5$
. . .

[정의 3]

빈발 1-항목집합 가중치 거리(줄여서, '가중치'라고 한다.) :

$$\boldsymbol{Weight_c}(D'_i[a_m], D'_j[a_m]) \quad (1 \le i,j \le n,\ 1 \le m \le \ell,\ c\text{:비교횟수})$$

본 논문에서 제안하는 가중치 거리 연산을 위한 수행 절차는 다음과 같다.

단계 1. D'_i과 D'_j를 선택한다. 비교횟수를 카운트하는 변수 c를 1로 초기화한다.

단계 2. D'_i에서 항목 빈발수가 가장 높은 항목(D'_{i_Max})과 빈발수(freq_count)를 선택한다.

단계 3. 단계 2에서 선택한 D'_{i_Max}과 동일한 항목, 그리고 그에 대한 빈발 수를 D'_j에서 선택한다. (만일 동일한 항목이 없으면, ∅로 치환하고 값은 0으로 한다.)

단계 4. 다음 식을 적용하여 연산한다.
$Weight_c(D'_i[a_m], D'_j[a_m])$ = |$D'_i[a_m]$의 freq_count − $D'_j[a_m]$의 freq_count|

단계 5. 단계 2에서 선택된 $D'_i[a_m]$와 동일한 freq_count를 가진 항목이 D'_i에 존재한다면, c=c+1하고, 단계 3, 4를 수행한다.

단계 6. 동일한 freq_count가 없다면, 다음 식을 적용하여 가중치를 연산하고 종료한다.

$$Weight(D'_i[a_m], D'_i[a_m]) = \frac{1}{c}\sum_{k=1}^{c} Weight_k(D'_i[a_m], D'_j[a_m])$$

[정의 4] D'_i,D'_j 간의 가중치 거리 평균 :

$$\boldsymbol{Weight_{avg}}(D'_i, D'_j) = \frac{1}{2}(Weight(D'_i, D'_j) + Weight(D'_j, D'_i))$$

위에서 요약한 수행 절차와 정의 4를 이용하여 데이터베이스간의 거리에 대한 가중치와 가중치 거리 평균을 계산하면 다음과 같다.

$Weight_1(D'_1[a], D'_3[a])$=|4−3|=1, $Weight(D'_1[a], D'_3[a])$=1/1+1=1
$Weight_1(D'_3[b], D'_1[b])$=|4−3|=1, $Weight_2(D'_3[d], D'_1[d])$=|4−2|=2
$Weight(D'_3[d], D'_1[d])$=1/2(1+2)=1.5
$Weight_{avg}(D'_1, D'_3)$=1/2(1+1.5)=1.25
. . .

기존의 연구[3]에서는 $L_1(D_1)$과 $L_1(D_3)$에 대해 거리가 0으로 나타나지만, 제안한 기법을 적용하면 1.25로 데이터베이스간의 거리에 차이가 발생하였음을 알 수 있고, 이러한 결과는 데이

터베이스간의 세밀한 식별능력을 보여주는 것이다.

[정의 5] 빈발 1-항목집합 데이터베이스간의 가중치 거리 평균 : $dist_{weight}(L_1(D_i), L_1(D_j)) = dist(L_1(D_i), L_1(D_j)) + Weight_{avg}(D'_i, D'_j)$

[정의 6] 관련된 데이터베이스간의 식별을 위한 척도로써 다음 식을 이용한다.

$$Measure(C_k) = \left| \left(\sum_{L_1(D_i), L_1(D_j) \in C_k}^{1 \le i \le j \le n} dist_{weight}(L_1(D_i), L_1(D_j)) \right) - k \right|$$

(k : 클러스터의 개수, $1 \le i \le j \le n$)

정의 5, 6의 식을 연산하게 되면, 다음과 같은 측정값을 얻을 수 있다.

(1) $\{\{D_1\}, \{D_2\}, \{D_3\}, \{D_4\}, \{D_5\}, \{D_6\}\}$

$Measure(C_6) = |(0+0+0+0+0+0)-6| = 6$

(2) $\{\{D_1, D_2\}, \{D_3, D_4\}, \{D_5, D_6\}\}$

$Measure(C_3) = |(1.4+0.7+1.55)-3| = 0.65$

(3) $\{\{D_1, D_2, D_3\}, \{D_4, D_5, D_6\}\}$

$Measure(C_2) = |(1.4+1.25+3.4+4.36+5+1.55)-2| = 14.96$

(4) $\{\{D_1, D_2, D_3, D_4, D_5, D_6\}\}$

$Measure(C_1) = |(1.4+1.25+3.4+0.7+4.1+0.7+4.88+5.5+4.38+4.36+4.5+5+4.5+5+1.55)-1| = 50.22$

위의 결과를 살펴보면, 가장 낮은 측정값을 갖는 경우가 가장 관련성 있는 데이터베이스를 클러스터링 한 경우로써, 여기서는 0.65에 해당되는 $\{\{D_1, D_2\}, \{D_3, D_4\}, \{D_5, D_6\}\}$ 이 된다.

4. 실험 및 성능 평가

본 논문에서 클러스터링을 위한 가중치 거리 기법의 성능 평가를 위한 데이터 집합은 표 2와 같고, 다중데이터베이스 구성은 항목 값 범위와 트랜잭션 발생 순서를 고려하였다.

표 2. 데이터 집합

데이터베이스 개수	10	전체 트랜잭션 총량	30000
데이터베이스 당 트랜잭션 수	3000	전체 항목 수	1000
트랜잭션 당 평균 항목 수	5	패턴 수	100

그림 1은 트랜잭션 수 증가에 따른 실행시간 비교인데, 트랜잭션의 총량이 증가함에 따라 트랜잭션간의 비교연산이 증가하여, Ideal&Goodness 기법[3]과 제안된 기법 모두 실행시간이 선형적으로 변화하였다. 트랜잭션의 총량이 16500에서부터 실행시간에 대해 차이를 보이는 것은 빈발항목만을 고려한 제안된 기법의 비교연산 횟수가 항목집합 전체를 대상으로 하는 Ideal&Goodness 기법[3]보다 작기 때문이다. Ideal&Goodness 기법[3]은 데이터베이스에 포함된 모든 항목들에 대해서 유사성을 고려하고 α, |class|를 구하는 연산에서 부하가 많기 때문에 실행시간이 많이 걸리게 된다.

그림 2는 항목 수 증가에 따른 클러스터 개수를 비교한 것으로 데이터베이스가 포함하는 트랜잭션 수가 고정되어 있고, 항목 수가 증가함에 따라 데이터베이스간의 항목들에 대한 유사성은 감소하게 된다. 따라서, 항목 수가 적은 경우 즉, 유사성이 높은 경우에 대한 Ideal&Goodness 기법[3]의 결과는 데이터베이스간의 식별이 제대로 이루어지지 않아 클러스터 수가 작게 나타난다. 제안한 기법은 유사한 항목이 발생하더라도 빈발 수에 따라 다중데이터베이스를 클러스터링하여 보다 관련성 높은 데이터베이스들로 클러스터를 형성한다.

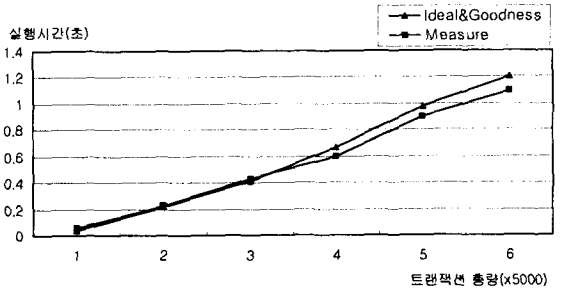

그림 1. 트랜잭션 수 증가에 따른 실행시간 비교

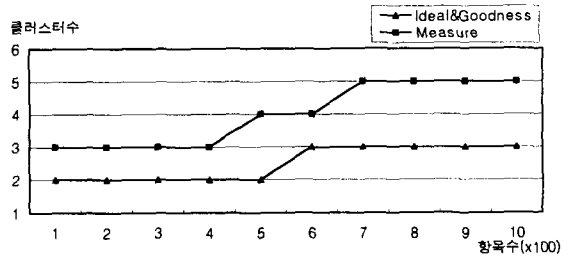

그림 2. 항목 수 증가에 따른 클러스터 개수 비교

5. 결 론

다중데이터베이스 마이닝을 위한 기존의 연구는 하나의 데이터 집합을 구축한 후, 여기에 적합한 알고리즘을 설계하는 것이었다. 이처럼, 필요한 정보를 발견하기까지는 많은 구축 비용이 필요하다.

제안한 기법은 항목집합을 포함한 다중데이터베이스에 대한 유사성을 식별하여 클러스터링 한다. 식별되어진 클러스터에 포함된 데이터베이스를 하나의 데이터 집합으로 구성한 후, 적합한 마이닝 알고리즘을 선택하여 적용하면 효율적인 마이닝 작업과 정확한 결과를 얻을 수 있다. 제안한 기법은 유사한 항목으로 구성된 데이터베이스간의 식별이 가능하고 최다 빈발항목만을 고려한 비교연산의 수행이므로 실행시간이 단축된다는 특징이 있다.

향후 연구방향으로는 빈발 1-항목집합에서 최다 빈발항목만을 고려하여 생성한 데이터 집합을 실수 값 범위에 있는 시간 데이터에 접목시켜 시간관계 유사성을 고려한 클러스터링 기법에 관한 연구로 확장하는 것이다.

참 고 문 헌

[1]P. Adriaans and D. Zantinge, "Data Mining",Addison-Wesley, JUN, 1996.

[2]H. Liu, H. Lu, and J. Yao, "Identifying Relevant Databases for Multidatabase Mining", Proc. of PAKDD, pp.210-221, 1998.

[3]C. Zhang and S. Zhang, "Database Clustering for Mining Multi-Databases", Proc. of the 2002 IEEE Int. Conf. Volume: 2, pp.974 -979, 2002.

[4]R. Agrawal, T. Imielinski, and A. Swami. "Mining association rules between sets of items in large databases", Proc. of the 20th 1993 ACM SIGMOD Int. Conf. on Management of Data, pp.207-216, 1993.