제19회 한국정보처리학회 춘계학술발표대회 논문집 제10권 제1호 (2003.5)

항목집합의 거리를 이용한 다중데이터베이스 클러스터링

김진현*, 박성련**, 윤성대*
*부경대학교 전자계산학과
**부경대학교 전산교육학과

e-mail:jhkim@dol.pknu.ac.kr

A MultiDatabase Clustering using Distance of Itemsets

Jin-Hyun Kim*, Sung-Lyeon Park**, Sung-Dae Youn*
*Dept. of Computer Science, Pukyong National University
**Dept. of Computer Science Education, Pukyong National University

요 약

장바구니 데이터들로 구성된 다중데이터베이스를 마이닝 하기 위한 선처리 작업으로는 Ideal&Goodness 기법이 있으며, Ideal&Goodness기법은 유사한 항목이 존재하는 데이터베이스간의 식별이 불가능하다는 단점이 있다. 그러므로 본 논문에서 제안하는 기법은 항목으로만 구성된 집합을 생성하여 데이터베이스간의 거리를 측정하고 항목집합간의 식별능력을 향상시키기 위하여 항목과 지지도를 갖는 항목 데이터 집합을 생성하고 지지도에 대한 확률을 계산한 후, 이를 비교 연산하여 가중치를 계산한다. 본 논문에서는 장바구니 분석을 위한 선처리 단계로써 활용 가능한 클러스터링 기법을 제안하며 성능평가를 통하여 데이터베이스간의 우수한 식별 능력을 보인다.

1. 서론

최근 기업들은 유통 환경의 대형화와 전자 상거래의 영향으로 고객들이 어떤 제품을 구입하는지에 대한 많은 세일즈 정보들을 생성, 수집하여 데이터베이스에 저장하고, 이러한 정보들을 마케팅에 활용하기 위해 노력하고 있다. 세일즈 정보를 활용하기 위해 필요한 것이 숨겨진 지식 발견이며 이를 위한 많은 연구 중에서 특히, 고객이 구입하는 항목들 간에 존재하는 연관규칙을 찾아내는 연구를 '장바구니 분석(Market Basket Analysis)' 이라고 한다[1]. 장바구니 분석에 사용되는 장바구니 데이터(Basket Data)는 트랜잭션 단위로 구성되며, 트랜잭션내의 투플(Tuple)은 전형적으로 고객 ID, 트랜잭션 날짜, 고객이 구입한 항목으로 구성되어 있다. 고객이 구입한 항목들에 대한 발생 빈도를 분석하면 고객의 항목 구입 성향에 대한 가치 있는 정보를 판단 할 수 있다.

항목으로 구성된 트랜잭션을 포함한 데이터베이스는 하나의 DBMS에 의해 관리되지만, 기업은 대형화된 유통 환경의 효율적인 관리 및 업무처리를 위해 독립된 데이터베이스를 다수의 DBMS로 관리하는 다중데이터베이스를 이용하기도 한다[2].

다중데이터베이스 마이닝을 위한 선처리 작업에 대한 연구로써는 모든 항목들에 대해서 비교 분석하는 유사성 측정 방법(Ideal&Goodness)[3]이 있다. Ideal&Goodness기법은 다중데이터베이스의 항목집합에 대한 유사성만을 고려하였으므로 유사한 데이터베이스에 대해서는 식별이 불가능하다는 단점이 있다.

그러므로, 본 논문에서는 트랜잭션으로 이루어진 다중데이터베이스에서 항목 데이터 집합간의 거리를 측정하여 다중데이터베이스 마이닝을 위한 클러스터링 기법을 제안하고자 한다.

2. 관련 연구

장바구니 분석에 대한 연구 중에는 항목집합의 지지도가 최소 지지도를 만족하는 빈발 항목집합을 찾아내는 연관규칙(Association Rule)기법[4]이 있다.

앞서 연구된 연관규칙 기법에 사용되는 용어에 대해 살펴보면 다음과 같다.

$I=\{a_1, a_2, a_3, \ldots, a_n\}$를 항목(item)들의 전체 집합이라 한다. 전체 집합에 속한 항목들로 이루어진 집합을 항목집합(itemset)이라 하며, 항목으로 이루어진

전체 집합의 부분집합이 된다. 트랜잭션 T는 데이터베이스 D의 부분집합이고, 집합 I에 속한 항목들로 구성된 집합 X가 트랜잭션 T에 포함되는 X⊆T 관계이면 T는 X를 지지한다(support)라고 한다. X를 지지하는 D에 있는 모든 트랜잭션들의 개수를 지지도(support)라고 한다.

본 논문에서는 항목들에 대한 지지도를 이용하여 데이터베이스간의 거리를 계산하고 유사한 항목으로 구성된 데이터베이스간의 식별 능력을 향상시키기 위해 거리 가중치 기법을 제안한다.

3. 데이터베이스간의 거리와 가중치 거리

3.1 항목 데이터 집합

다중데이터베이스 집합을 MultiDB={D_1, D_2, D_3, ... ,D_m}이라 하면, 데이터베이스 D_i(i∈[1..m])는 MultiDB의 부분집합이고, 트랜잭션 T는 D_i의 부분집합이다. 그리고 트랜잭션에 속한 항목과 각 항목에 대한 지지도로 구성된 집합을 본 논문에서는 항목 데이터 집합(IS_i, i∈[1..m])이라고 한다.

표 1은 6개의 데이터베이스로 구성된 다중데이터베이스를 예시한 것으로 각 데이터베이스는 항목집합과 트랜잭션 ID(TID)를 갖는 트랜잭션들로 구성된다.

<표 1> 6개의 데이터베이스로 구성된 다중데이터베이스

D_1

TID	항목집합
110	a, b, c
120	a, b, d
130	a, b, c, d, e
140	a, e

D_2

TID	항목집합
110	a, c
120	a, b, c
130	b, c
140	a, c, e

D_3

TID	항목집합
110	a, b, d, e
120	b, c
130	a, b, c, d
140	a, b, c, d, e
150	a, b, c

D_4

TID	항목집합
110	d, g, h, i
120	d, j
130	g, i, j
140	d, h, i
150	h, i

D_5

TID	항목집합
110	e, g, h, i
120	e, h, i
130	e, h, j
140	h, j
150	e, g

D_6

TID	항목집합
110	g, i
120	g, h, i, j
130	g, h
140	g, h, j

<표 2> 6개의 데이터베이스에 대한 항목과 지지도

D_1

TID	항목집합
110	a, b, c
120	a, b, d
130	a, b, c, d, e
140	a, e

⇨ IS_1

항목	지지도
a	4
b	3
c	2
d	2
e	2

D_2

TID	항목집합
110	a, c
120	a, b, c
130	b, c
140	a, c, e

⇨ IS_2

항목	지지도
c	4
a	3
b	2
e	1

표 2의 항목 데이터 집합(IS_i)은 앞서 소개한 연관규칙 기법[1]의 빈발 1-항목집합 구성과 유사한 방법으로 생성된다. 그러나 본 논문에서는 연관규칙 기법의 후보 항목집합에 대한 임계값으로 주어지는 최소 지지도는 고려하지 않는다.

다음은 IS_i에서 항목만을 선택하여 집합 $I(D_i)$를 구성한다.

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

거리를 연산하기에 앞서, 표 3은 본 논문에서 사용된 수식 및 수행절차에 대한 파라미터들을 요약한 것이다.

<표 3> 다중데이터베이스 클러스터링을 위한 파라미터

기호	기호 의미
$I(D_i)$	항목으로 구성된 데이터 집합
IS_i	항목 데이터 집합
Sup	지지도
$IS_i(a_n)$	IS_i 에 속한 항목(a_n)
a_n[Sup]	항목 a_n의 지지도
C_k	클러스터 C_k

3.2 항목으로 구성된 집합간의 거리

앞서 살펴 본 $I(D_i)$를 피연산자로 하는 정의 1은 데이터베이스간의 거리를 나타내는 것이며 다음과 같이 정의한다.

[정의 1]

$$dist(I(D_i),I(D_j)) = \frac{|(I(D_i)-I(D_j))\cup(I(D_j)-I(D_i))|}{|I(D_i)\cup I(D_j)|} \quad (1\le i, j\le m)$$

정의 1은 $I(D_i)$와 $I(D_j)$에 속한 항목간의 비유사성을 연산하기 위한 것으로 $I(D_i)-I(D_j)$는 집합의 차를 나타내며, 그 의미는 $I(D_i)$에 속한 항목 중 $I(D_j)$에 속하지 않는 항목을 찾아내는 것이다. 같은 방법으로 $I(D_j)$에는 속하지만 $I(D_i)$에는 속하지 않는 항목을 찾아낸다. 이렇게 얻어진 두 집합간의 차이를 더하고 원소의 개수를 구한 결과에 $I(D_i)$와 $I(D_j)$의 합집합 원소의 개수로 나누어 거리를 계산한다.

정의 1을 사용하여 데이터베이스간의 거리를 연산하면 다음과 같다.

$$dist(I(D_1),I(D_1)) = \frac{|\{\}|}{|\{a,b,c,d,e\}|} = \frac{0}{5} = 0$$

$$dist(I(D_1),I(D_2)) = \frac{|\{d\}\cup\{\}|}{|\{a,b,c,d,e\}|} = \frac{1}{5}$$

. . .

$$dist(I(D_5),I(D_6)) = \frac{|\{e\}\cup\{\}|}{|\{e,g,h,i,j\}|} = \frac{1}{5}$$

$$dist(I(D_6),I(D_6)) = \frac{|\{\}|}{|\{g,h,i,j\}|} = \frac{0}{5} = 0$$

위의 결과에서 $dist(I(D_i),I(D_j))$는 0에서 1사이의 범위를 가진다. 값이 1이면 2개의 데이터베이스간에는 전혀 다른 항목으로만 구성되고, 0이면 동일한 항목으로

구성되며 0과 1사이의 값은 유사한 항목들로 구성된다.

정의 1에서 연산한 값은 항목 발생만을 고려한 데이터베이스간의 거리를 나타냄으로 데이터베이스의 트랜잭션들에서 여러 번 발생한 항목들과 한번 발생한 항목이 존재하는 경우에 대해서는 거리 차이가 없는 것으로 나타난다. 이것은 유사한 항목들로 구성된 데이터베이스간의 세밀한 거리 식별 능력을 갖지 못함을 의미한다. 이러한 단점을 보완하여 유사한 데이터베이스간의 거리를 명확하게 구분하는 방법으로써 아래와 같은 기법을 제안한다.

3.3 항목 데이터 집합의 가중치 거리

항목 데이터 집합의 가중치 거리는 데이터베이스간의 가중치 거리를 나타내는 것이며 이러한 거리를 연산하기 위해서 사용되는 각 항목의 지지도에 대한 확률은 다음과 같이 정의한다.

[정의 2]

$$P[a_i]=\frac{a_i[Sup]}{\sum_{i=1}^{n} a_i[Sup]} \quad (a_i \in IS_j,\ 1 \le j \le m)$$

정의 2는 전체 항목 지지도에 대한 각 항목의 지지도의 비율을 의미하며, 위의 예시에서 확률을 연산한 결과는 표 4와 같다.

<표 4> 항목 지지도에 대한 확률

IS_1

항목	지지도	P[ai]
a	4	0.31
b	3	0.23
c	2	0.15
d	2	0.15
e	2	0.15

IS_2

항목	지지도	P[ai]
c	4	0.4
a	3	0.3
b	2	0.2
e	1	0.1

IS_3

항목	지지도	P[ai]
b	5	0.28
a	4	0.22
c	4	0.22
d	3	0.17
e	2	0.11

IS_4

항목	지지도	P[ai]
i	4	0.29
d	3	0.21
h	3	0.21
g	2	0.14
j	2	0.14

IS_5

항목	지지도	P[ai]
h	4	0.29
e	4	0.29
g	2	0.14
i	2	0.14
j	2	0.14

IS_6

항목	지지도	P[ai]
g	4	0.36
h	3	0.27
i	2	0.18
j	2	0.18

항목 데이터 집합의 가중치 거리는 $dist_{wgt}(D_i, D_j)$, $(1 \le i, j \le m)$이며 그 값을 구하기 위한 수행 절차는 다음과 같다.

단계 1. 각 항목 데이터 집합에서 최대 지지도를 갖는 항목을 포함한 항목 데이터 집합만을 선택하여 클래스를 구성한다.

단계 2. 클래스에 속한 항목 데이터 집합 IS_i과 IS_j를 선택한다.

단계 3. 정의 2에 따라 IS_i의 각 항목 지지도의 확률을 구하고 확률이 가장 높은 것과 그에 대한 항목을 선택한다. 만일, 최대 지지도를 갖는 항목이 하나의 항목 데이터 집합에서 2개 이상 존재한다면, 다음 단계를 수행한다.

① 항목에 있어서 최대 지지도가 같은 경우는 확률도 같으므로 하나의 확률만을 선택하고, 항목들에 대해서는 모두 선택한다.

② 선택한 항목들과 동일한 항목을 IS_j에서 선택하되, 확률 차이가 가장 큰 항목 하나만을 선택한다.

단계 4. 단계 3에서 선택한 항목과 동일한 항목, 그리고 그에 대한 확률을 IS_j에서 선택한다.

단계 5. 다음 식을 적용하여 연산한다. 여기서 Weight는 가중치를 나타낸다.

Weight($IS_i(a_n)$, $IS_j(a_n)$) = |$IS_i(a_n)$의 $P[a_n]$ - $IS_j(a_n)$의 $P[a_n]$|

단계 6. Weight($IS_j(a_n)$, $IS_i(a_n)$)은 단계 2에서 6까지의 동일한 수행절차를 거쳐 연산한다.

단계 7. 다음 식을 적용하여 가중치 거리($dist_{wgt}(D_i, D_j)$)를 연산하고 종료한다.

Weight($IS_i(a_n)$, $IS_j(a_n)$)+Weight($IS_j(a_n)$, $IS_i(a_n)$)

위에서 요약한 수행 절차와 정의 1의 결과를 더하여 데이터베이스간의 거리를 계산하고, 결과값 중에서 가장 작은 것을 선택하여 다중데이터베이스를 클러스터링한다. 아래의 행렬은 위에서 예시한 6개의 데이터베이스간의 거리에 대해 두 개의 클래스($Class_1$, $Class_2$)로 분류하고 각 클래스에 속한 항목 데이터 집합간의 거리를 연산한 결과이다.

$Class_1 = \{D_1, D_2, D_3\}$, $Class_2 = \{D_4, D_5, D_6\}$

$$M_1=\begin{array}{c|ccc} & D_1 & D_2 & D_3 \\ \hline D_1 & 0 & - & - \\ D_2 & 0.26 & 0 & - \\ D_3 & 0.14 & 0.26 & 0 \end{array} \qquad M_2=\begin{array}{c|ccc} & D_4 & D_5 & D_6 \\ \hline D_4 & 0 & - & - \\ D_5 & 0.23 & 0 & - \\ D_6 & 0.33 & 0.24 & 0 \end{array}$$

단계 1에서 다중데이터베이스를 클래스로 구성한 이유는 최대 지지도를 포함하지 않은 데이터베이스를 일차적으로 구분하여 데이터베이스간의 비교 연산 횟수를 줄이기 위해서이다. 위의 예시된 연산과정에서는 9번의 비교 연산 횟수가 줄어들었다.

앞에서 소개한 정의 2와 수행 절차를 적용하여 데이터베이스간의 거리에 대한 측정값을 구하기 위한 식은 다음과 같다.

$$Measure(C_k)=|k-\sum_{D_i,D_j \in C_x}^{1 \le x \le n} dist(D_i,D_j)+dist_{wgt}(D_i,D_j)| \quad (1)$$

식 (1)에서는 클러스터 개수 증가에 따른 중심점간의 거리 합이 증가하게 되므로 클러스터 개수 k를 적용한다. 클러스터 개수 k에서 항목으로 구성된 집

합간의 거리와 가중치 거리의 합의 차를 구하여 데이터베이스간의 거리에 대한 측정값으로 사용한다. 각 클래스에 대해 측정값을 연산하면 다음과 같다.

$M(Class_1)$ =

	D_1	D_2	D_3
D_1	0	-	-
D_2	0.46	0	-
D_3	0.14	0.46	0

$M(Class_2)$ =

	D_4	D_5	D_6
D_4	0	-	-
D_5	0.56	0	-
D_6	0.53	0.44	0

데이터베이스간의 거리에 대한 측정값에서 가장 작은 결과를 갖는 클러스터링을 선택하여 최종적으로 클러스터를 얻는다. 앞서 예시된 다중데이터베이스에 대한 클러스터링은 다음과 같다.

(1) $Class_1$에 대한 측정값
1) {{D_1}, {D_2}, {D_3}} : $Measure(C_3)=|3-(0+0+0)|=3$
2) {D_1, D_2, D_3} : $Measure(C_1)=|1-(0.26+0.14+0.26)|=0.34$
3) {D_1, {D_2, D_3}} : $Measure(C_2)=|2-(0+0.26)|=1.74$

(2) $Class_2$에 대한 측정값
1) {{D_4}, {D_5}, {D_6}} : $Measure(C_3)=|3-(0+0+0)|=3$
2) {D_4, D_5, D_6} : $Measure(C_1)=|1-(0.23+0.33+0.24)|=0.2$
3) {{D_4, D_6}, D_5} : $Measure(C_2)=|2-(0.33+0)|=1.67$

위의 결과에서 가장 낮은 측정값을 갖는 경우가 가장 관련성 높은 데이터베이스를 클러스터링 한 경우로써 $Class_1$은 0.34에 해당되는 {D_1, D_2, D_3}으로 클러스터링하고, $Class_2$는 0.2에 해당되는 {D_4, D_5, D_6}으로 클러스터링 한다.

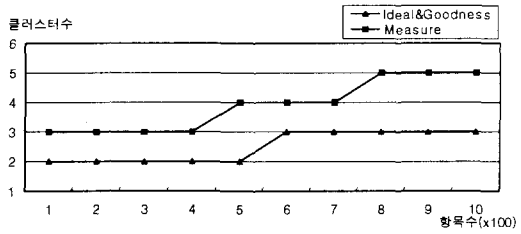
4. 실험 및 성능 평가

본 논문에서는 성능 평가를 위해 표 5와 같은 Synthetic Data Set[4]을 사용하였다. 다중데이터베이스는 항목 값 범위와 트랜잭션 발생 순서를 고려하여 구성하였다.

<표 5> 데이터 집합

데이터베이스 개수	15	전체 항목 수	1000	패턴 수	5
데이터베이스 당 트랜잭션 수	2000	전체 트랜잭션 총량	30000	트랜잭션 당 평균 항목 수	100

그림 1은 제안한 기법과 Ideal&Goodness기법[3]의 항목 수 증가에 따른 클러스터 개수를 비교한 것이다. 트랜잭션 수가 일정하고, 항목 수가 증가하게 되면 데이터베이스를 구성하는 항목의 종류는 많아지고, 최대 지지도는 감소한다. 따라서, 항목 수가 작은 경우 즉, 데이터베이스간의 유사성이 높은 경우에 대한 Ideal&Goodness기법[3]의 결과는 항목 종류만을 고려하였기에 식별이 제대로 이루어지지 않아 클러스터 수가 작게 나타난다. 그러나, 제안한 기법은 동일한 항목이 존재하더라도 지지도에 따른 확률의 차이가 발생하여 유사성 높은 데이터베이스들을 세밀하게 식별한다.



(그림 1) 항목 수 증가에 따른 클러스터 개수 비교

5. 결론

기존의 연구는 장바구니 데이터로 구성된 트랜잭션 데이터베이스들을 식별하는 방법으로 모든 항목에 대한 유사성 측정 방법이며, 항목 발생이 유사한 경우에 대해서는 데이터베이스간의 식별이 불가능하다는 단점이 있다. 이러한 단점을 보완하기 위해 본 논문은 항목 지지도에 대한 최고 확률을 고려한 클러스터링 기법을 제안한다. 제안한 기법은 항목집합의 거리를 구하고, 데이터베이스간의 세밀한 식별 능력을 갖는 가중치 거리 비교 연산을 위해 항목 지지도에 대한 확률을 이용한다. 제안한 기법의 타당성을 입증하기 위한 실험은 유사한 항목 발생 수 증가에 따른 클러스터 개수를 비교하고, 제안한 기법이 성능평가를 통하여 기존의 연구보다 데이터베이스간의 식별 능력이 우수함을 알 수 있었다.

향후 연구과제는 장바구니 데이터 발생이 특정 시간과 밀접한 관계에 있으므로 사건 발생 시간을 고려한 다중데이터베이스 클러스터링 기법에 관한 연구로 확장하는 것이다.

참고문헌

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

[2] D. Calvanese, G. De Giacomo, M. Lenzerini, D. Nardi, R. Rosati "Conceptual Modeling and Reasoning Support" Conference on Cooperative Information Systems. 1998.

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

[4] R. Agrawal and R. Srikant. "Fast algorithms mining association rules in large databases" Research Report RJ 9839, IBM Almaden Research Center, San Jose, California, June 1994.