Generalized α chain rule에 기반한 Group Item Recommendation

염선희[0] 조동섭
이화여자대학교
{kingni, dscho}@ewha.ac.kr

Group Item Recommendation based on Generalized α Chain Rule

Sun-Hee Youm[0] Dong-Sub Cho
Dept. of Computer Science and Engineering, Ewha Womans University

요 약

데이터 마이닝을 통해 우리는 숨겨진 지식, 예상되지 않았던 경향 그리고 새로운 법칙들을 방대한 데이터에서 이끌어내고자 한다. 본 논문에서 우리는 사용자들의 구매 트랜잭션을 시간에 따라 분석하여 동시에 구매되는 상품을 미리 예측하는 알고리즘을 제안하고자 한다. 기존의 방법들에서는 구매된 상품간의 시간차를 고려하지 않은 방법만을 제안해 왔다. 따라서 서로 연관되지 않은 상품군이 예측 될 확률이 높았다. 본 논문에서 제안하고 있는 α chain rule에서는 일정 시간동안의 사용자들이 상품을 구매한 후 다음 상품을 구매할 때까지의 시간을 고려한다. 따라서 좀더 정확히 동시에 구매될 상품군을 예측할 수 있다. 본 논문은 제안하고 있는 α chain rule을 계산해 내는 알고리즘에 대해 주로 논의하겠다.

1. 서 론

많은 기업들은 방대한 양의 데이터를 수집하고 모인 데이터들로부터 숨겨진 지식, 예견되지 않았던 패턴 그리고 새로운 법칙들을 이끌어내고자 한다. 데이터마이닝은 이러한 기업들의 욕구에 부합하는 가장 적절한 수단이 되어 왔다.. 데이터마이닝이란 대량의 데이터로부터 쉽게 드러나지 않는 유용한 정보들을 추출하는 과정을 말한다. 여기서 정보는 묵시적이고 잘 알려져 있지 않지만 잠재적으로 활용가치가 있는 정보를 말한다.

특히 전자상거래에서 사용자들의 선호도, 관심, 구매경험과 같은 자료를 기초로 원하는 정보를 자동으로 제공하는 것은 중요한 문제이다. 이런 경우 사용자의 지속적인 이용을 이끌어 낼 수 있고 사용자 역시 유용한 정보를 쉽게 얻을 수 있다.

본 논문에서는 사용자의 아이템 구매패턴을 분석하여 관심이 있을 아이템을 예측하여 추천하는 시스템을 제안하고자 한다. 데이터마이닝 기법에는 연관규칙, 분류규칙, 클러스터링, 유사성 탐색, 순서 패턴, 신경망, 결정 트리 등등이 있다. 그 중에서 본 논문은 α chain rule이라는 새로운 방법을 제안한다. 기존의 연관 규칙과 같은 법칙에서는 동시에 판매될 상품을 예측할 때 상품이 판매된 시간차는 전혀 고려하지 않았다. 그러나 시간차를 고려하지 않는 것은 서로 관련 없는 상품들이 함께 예측될 수 있는 위험성을 가지고 있다. 그래서 본 논문에서는 동시에 판매되는 상품들을 예측할 때 상품이 판매되는 시간차가 0과 1사이의 α 값을 이용하여 반영되게 된다.

이 논문은 2001년도 두뇌한국 21사업에 의하여 지원되었음.

2. 본 론

2.1 관련연구

2.1.1 Associaton Rule

데이터 마이닝을 소개할 때 대표적으로 언급되는 기술로 백화점이나 슈퍼카멧에서 한 번에 함께 산 물건들에 관한 연관 규칙을 찾아내는 기술이다. 실제 데이터를 이용해 발견됐던 아주 유명한 연관 규칙 중 하나는 미국의 대형 편의저의 소비자 구매 데이터에 이 기술을 적용한 결과, 아기 일회용 기저귀를 사는 사람은 맥주도 같이 산다는 연관 규칙을 발견한 것이다. 이러한 패턴을 발견하고 소비자들에 관해 조사해 본 결과, 보통 아기 엄마가 남편에게 기저귀를 사오라고 하면 남편이 편의점에 들러 기저귀를 사면서 같이 맥주도 사간다는 것을 발견했다.

Association rule을 찾아내는 방법은 다음과 같다. I= $\{i_1, i_2, \cdots, i_k\}$을 항목이라 부르는 리터럴(literal)들의 집합이라 하자. D를 트랜잭션들의 집합이라 부르고, 각 트랜잭션 T 는 T ⊆ I인 항목들의 집합이다. X를 항목들의 집합이라 하자. ϕ이 아닌 항목 집합 X, Y에 대해 X ⊂I, Y⊂I 에 대한 연관 규칙 X→Y는 X∩Y= ϕ의 특성을 갖는다. X는 규칙의 가정, Y는 규칙의 결과라고 한다. 항목집합 I의 부분집합 X에 대해, X⊆T이면 T는 X를 만족한다고 정의한다. 최소 지지도(minimum support)를 만족하는 X⊆I를 빈발 항목집합이라 한다[5].

2.1.2 Sequential Pattern Mining

바코드의 출현으로 소매 조직들은 방대한 판매 데이터들을 데이터베이스에 저장할 수 있게 되었고 이 데이터

들은 트랜잭션(Transaction) 날짜와 판매된 아이템으로 구성되어 있다. 소매 조직들은 바코드 등을 통해 받은 정보를 시간의 순서에 따라 저장하게 되었다. 여러 기업들에서는 이러한 데이터베이스를 분석하여 의사결정의 중요한 요소로 활용하고 있다.

Sequential pattern mining은 Agrawal[2]에 의해 처음 제기되었다. 이 문제는 데이터를 시간적으로 분석한다는 의미에서 연관 규칙(Association rule)과는 차이가 있다. 각각 연속 데이터는 항목들을 포함하는 트랜잭션들로 이루어지고 각 트랜잭션은 트랜잭션 시간들로 구성되어 있다. 결론적으로 사용자가 최소 지지도(Minimum support)를 만족하는 연속 패턴을 찾아내는 문제라고 볼 수 있다. 여기에서 최소 지지도는 패턴을 포함하는 연속 데이터의 백분율로 정의한다.

Sequential pattern mining은 소매산업과 우편을 이용한 마케팅, 부가 세일, 고객 만족 등에서 시작되었지만 다른 과학분야나 경영 분야에 적용되고 있다.

2.2 α Chain Rule

2.2.1 문제 기술 및 전처리

이 장에서는 α chain rule을 적용하는 방법에 대해서 설명하겠다. 사용자의 트랜잭션을 기본 데이터로 한다.

고객 아이템 구매 트랜잭션이 있다고 하자. 트랜잭션은 사용자 아이디, 트랜잭션 시간, 구매 된 아이템으로 구성되어 있다. 아이템셋(Itemset)은 각 사용자의 구매된 구매 된 아이템들의 집합이고 시간순서대로 되어 있다. 본 논문에서는 상품과 상품 사이의 구매 시간을 고려하여 각 차원의 아이템셋에 값(value)을 매기게 된다.

아이템셋을 만드는 과정을 보자.

표1이 다음과 같이 주어져 있다고 하자.

표1 사용자 아이디, 구매시간, 구매된 아이템으로 정렬된 데이터 베이스

사용자 아이디	구매 시간	구매된 아이템
1	March 3	111
1	March 15	112
2	March 1	114
2	April 4	112
2	May 15	116
3	February 27	119
3	April 13	120
3	April 24	111
4	March 5	123
4	March 19	121

사용자 아이디별로 구매시간의 순서에 맞추어 아이템 시퀀스를 만든다.
결과는 표 2와 같다.

표 2 그룹 1의 아이템시퀀스

사용자 ID	Itemsets
1	(111,112)
2	(114, 112, 116)
3	(119, 120, 111)
4	(123, 121)

표2와 같이 각 사용자별로 아이템 시퀀스가 만들어지면 이것을 입력으로 하여 동시에 구매되는 각각의 아이템셋의 값(value)을 정할 수 있다. 본 논문에서 제안하고 있는 α chain rule에 의해 이 값이 정해진다. 정해진 값(value)은 각각의 구매데이터에서 특정 아이템셋이 얼마나 자주 나타나는 가에 대한 지표로 사용 가능하다. 다음 장에서는 구체적으로 α chain rule을 계산하는 과정에 대해 설명하겠다.

2.2.2 실행 단계

앞에서도 설명했듯이 α chain rule은 여러 단계를 거친 후에 구매된 아이템의 연관성을 고려하여 각 아이템셋의 값(value)을 구하는 것이다. A라는 상품을 구매한 후에 B라는 상품의 구매가 다른 여러 상품을 구매한 후에 이루어졌다면 A라는 상품과 B라는 상품이 동시에 구매될 확률은 적다고 말할 수 있다. 따라서 가중치 $\alpha(0<a\leq 1)$를 주어서 그 연관성이 작음을 값(value)에 반영할 수 있다. 그러나 고려하고 있는 것은 상품 A와 상품 B가 선후가 아니다. 즉 상품 A를 산 후에 어떤 상품이 판매되었으냐를 예측하는 것이 아니고 상품 A와 동시에 구매될 상품이 어떤 것인가를 예측하는 것이다.

우선 2개짜리 아이템셋인 경우에 관해 보자.

알고리즘은 그림 1과 같다. 각 아이템시퀀스를 따라가면서 아이템 i를 산 후에 $k(1\leq k\leq n)$개의 물건을 산 후에 j를 산 경우 (i,j)의 값을 $1\times \alpha^{k-1}$ 만큼 증가시킨다. 이 과정을 모든 아이템시퀀스에 대해서 한다. 단 i가 j보다 큰 경우에는 (j,i)에 값을 증가시킨다. 왜냐하면 이 알고리즘에서는 구매된 순서는 고려하지 않기 때문이다.

예를 보면서 보자.

Ex) 아이템시퀀스가 각각 (111, 113, 115, 112), (111, 115, 112)이고 111,112,113,115의 일련번호가 각각 1,2,3,5라고 하자.

(1,2)의 경우 첫 번째 아이템시퀀스에서 두 번의 아이템을 거쳐서 구매되었으므로 a*a의 값이 더해지고, 두 번째 아이템시퀀스에서는 한 번의 아이템을 거쳐서 구매되었으므로 a의 값이 더해지게 된다. 다른 경우도 마찬가지이다. 따라서 각각 다음과 같은 값을 가지게 된다.

```
for(i=0; i(# of itemset in each sequence item; I++){
   item = 1;
   while(until all items are traversed in each
itemset){
      temp_pt1 = itemset[i][item];
      temp_pt2 = itemset[i][item+1];

if(temp_pt1 > temp_pt2)
{
temp = temp_pt1;
temp_pt1 = temp_pt2;
temp_pt2 = temp;
}
      set[temp_pt1][temp_pt2]
    =set[temp_pt1][temp_pt2]+1;
   }
   for(j=item-2; j>0; j--){
      value = value * a;
      temp_pt1 = itemset[i][j];
      temp_pt2 = itemset[i][item];
if(temp_pt1 > temp_pt2)
{
temp = temp_pt1;
temp_pt1 = temp_pt2;
temp_pt2 = temp;
}
      set[temp_pt1][temp_pt2]
=set[temp_pt1][temp_pt2]+value;
  }
item ++;
}
```

그림 1 αchain rule 알고리즘

표 4 αchain rule을 거친 후에 2개짜리 시퀀스의 값

시퀀스	값(value)
(1,2)	$\alpha \times \alpha + \alpha$
(1,3)	1
(1,4)	0
(1,5)	$\alpha + 1$
(2,3)	α
(2,4)	0
(2,5)	2
(3,4)	0
(3,5)	1
(4,5)	0

이렇게 구해진 값은 상품이 n개인 경우에도 확장가능하다. 그러나 αchain rule을 그대로 적용할 경우에 만들어진 2개짜리 아이템셋을 이용할 수 없고 또한 계산의 복잡함이 있기 때문에 좀더 간단한 방법을 제안하고자 한다.

2.2.3 n개의 아이템셋으로의 확장

3개 이상의 아이템셋의 값(value) 구하고자 할 때는 휴리스틱(heuristic)한 방법을 이용하고자 한다.

여기서는 우선 3개짜리 아이템셋에 관해서 설명하겠다.

예를 들면서 살펴보자. 아이템셋 (1,3,5)의 값(value)을 구하고자 할 때는 (1,3), (1,5), (3,5)의 값을 모두 더한 값이 최종값이 된다. 따라서 본 논문의 예제에서는 1+(1+α)+1이 (1,3,5)의 값이 되게 된다. 이렇게 할 경우 전체 아이템시퀀스에서 동시에 1,3,5가 나오는 경우를 찾기 위해 데이터베이스를 모두 훑어야 하는 번거로움을 피할 수 있어 계산량을 줄일 수 있다.

고차원의 경우도 3차원의 경우와 같은 방법으로 구할 수 있다.

제안한 알고리즘을 이용하여 웹사이트나 상점등에서 값(value)이 큰 시퀀스들을 동시에 진열하여 소비자의 구매를 촉진할 수 있다.

3. 결 론

본 논문에서는 구매된 아이템간의 관계를 가늠할 수 있는 값을 구하는 α chain rule을 소개했다. 제안하고 있는 알고리즘은 사용자들의 구매 트랜잭션을 시간에 따라 분석하여 동시에 구매되는 상품을 미리 예측할 수 있게 한다. 기존의 방법들에서는 구매된 상품간의 시간차를 고려하지 않은 방법만을 제안해 왔다. 따라서 서로 연관되지 않은 상품군이 예측 될 확률이 높았다. 본 논문에서 제안하고 있는 α chain rule에서는 일정 시간동안의 사용자들이 상품을 구매한 후 다음 상품을 구매할 때까지의 시간을 고려한다. 따라서 좀더 정확히 동시에 구매될 상품군을 예측할 수 있다.

앞으로 대량의 데이터에 대해서도 알고리즘을 적용해 보고 그 성능을 확인할 것이다. 또한 지속적인 성능평가를 통해 알고리즘을 발전시켜서 최적의 방법을 제시할 것이다. 또한 실제 데이터를 적용해서 결과를 확인하고 실제 웹사이트에 적용해 보고자 한다.

[참 고 문 헌]

[1] Tarun Khanna, "Foundations of Neural Networks," Addison-Wesley Publishing Company, 1990.

[2] Rakesh Agrawal, Ramakrishnan Srikant, "Mining Sequential Patterns," In Proc. of the 11th Int'l Conference on Data Engineering, Taipei, Taiwan, March 1995.

[3] Mehmet M. Dalkilic, Edward L. Roberston, "Information dependencies," Proceedings of the nineteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, pp. 245 ? 253 2000.

[4] Rosa Meo, "Theory of dependence values," ACM Trans. Database Syst. Pp. 380 ? 406, Sep. 2000.

[5] R. Agrawal and R. Srikant, "Fast algorithms for mining Association Rules in Large Databases," In Proceedings of the 20th VLDB Conference, Santiago, Chile, Sept., 1994.