전자상거래에서 연관규칙을 이용한 추천 시스템의 설계 및 구현

오재영° 전종훈
명지대학교 컴퓨터공학과
{ojy0217°, jchun}@mju.ac.kr

Design of recommendation system using association rule in e-Commerce

Jae Young Oh° Jonghoon Chun
Department of Computer Engineering, Myongji University

요 약

본 논문은 데이터 마이닝에서 사용되는 연관규칙(Association Rule)을 활용하여 고객에게 상품을 추천하는 방법을 제안한다. 일반적으로 한명의 고객에 대하여 적용할 수 있는 연관규칙의 개수가 한 개 이상이 될 수 있다는 가정하에, 고객과 연관규칙과의 적합성 여부를 값으로 나타내는 방안을 고안하고 이를 이용하여 고객에 대한 연관규칙의 순위를 부여하는 방식을 연구한다. 또한 상품 추천 시 범위 값을 가지는 속성을 고려하여 상품을 추천하도록 하는 방법을 제안하고 이 방법의 타당성과 타 방식과의 비교우위를 실험을 통하여 검증한다.

1. 서 론

인터넷 환경의 급속한 발달과 보급으로 전자상거래 시장이 빠르게 증가하고 전자상거래 시장의 증가에 따른 사용자들의 다양한 요구에 의해 개인화는 전자상거래의 중요한 핵심기술로 떠올랐다. 현재 인터넷의 많은 전자상거래 업체들의 관심사는 자신이 가지고 있는 많은 상품들을 사용자의 특성과 사용자의 상품에 대한 관심 정도를 고려하여 제공하여 상품 판매를 증가시켜 기업의 이익을 극대화하는 것이다.

현재까지 고객이 관심을 보일 만한 것들을 추천하는 많은 연구가 이루어져왔으며 그 중 대표적인 것은 내용기반 추천(content based)[1]과 협업(collaboration)을 통한 추천[2]이 있다. 내용기반 추천은 정보검색 분야에서 그 기원을 찾을 수 있는 것으로 사용자가 과거에 구매를 하였거나 관심을 보인 상품의 프로필과 유사한 상품간의 비교를 통하여 추천이 이루어진다. 예를들면 DVD를 많이 구매한 고객에게는 새로 나온 DVD를 보여주고 책을 많이 구매한 고객에게는 책을 추천하는 방식이다. 이 방식은 고객의 과거정보에만 의존하기 때문에 고객에게 제한된 추천만이 이루어지는 문제점이 있다. 협업을 통한 추천 방식은 고객이 과거에 구매한 상품들을 비교하여 다른 고객의 구매 양상과의 유사 정도를 측정하여 유사한 구매를 보인 다른 고객이 많이 구매한 상품을 추천하는 방식이다. 예를들어 고객 A와 고객 B가 있는데 고객 A는 DVD와 책을 구매했고 고객 B는 DVD와 CD만 구매했다고 하고 두 고객간의 유사도가 시스템에서 정한 임계치 이상이 된다면 고객 A에게는 CD를 추천하고 고객 B에게는 책을 추천하는 방식이다. 이 방식은 시스템 구축 초기에 일정량 이상의 고객 데이터가 축적되어야 추천을 시작할 수 있다는 단점과 고객이나 상품이 추가될 경우 다른 모든 고객과의 유사도를 다시 계산 해야하기 때문에 고객이나 상품이 많이 질 경우 성능에 문제가 생길 수 있다 는 단점이 존재한다. 본 논문에서 제안하는 연관규칙을 이용하여 상품을 추천할 경우 시스템 구축 초기의 일정량의 고객 데이터 없이 연관규칙의 정의만으로 추천 시스템을 구축할 수 있고 사용자가 속한 연관규칙을 찾아내는 연산 비용은 사용자와 상품의 수에 크게 영향을 받지 않고 추천을 할 수 있다.

데이터 마이닝에서는 고객의 구매 트랜잭션(transaction)을 분석하여 연관규칙(Association rule)을 추출한다. [3] 일반적으로 연관규칙은 "임의의 사건 A가 일어났을 때, 사건 B가 일어나는 것" 을 나타내며, 추천 시스템에서는 "임의의 고객이 조건 A를 만족할 경우, 조건 B를 만족한다." 를 나타낸다. 여기서 조건 A는 고객의 나이, 성별 등의 특징이 될 수 있고 고객이 구매한 상품들을 나타낼 수 있다. 그리고 조건 B는 고객에게 추천할 상품으로 정의된다. 연관규칙의 예를 들면 남성고객이 맥주와 기저귀를 구매한 고객이 전체 트랜잭션에서 일정회수 이상 존재할 경우 남성, 맥주, 기저귀가 관계가 있다고 보고 이를 특정 형식으로 나타내는 것을 의미한다. 이러한 연관 규칙은 일반적으로 기업에서 마케팅을 펼칠 때 사용되는데 예를 들어 자동차와 함께 구매한 상품들의 트랜잭션을 분석한 결과에 자동차와 함께 선글라스를 많이 구매하였다는 결과가 나올 경우 선글라스와 관련된 마케팅을 펼치면 관련 없는 상품에 관한 마케팅을 하는 것보다 더 좋은 효과를 가져올 수 있다.

연관규칙을 사용하면 고객의 구매 트랜잭션 정보 없이 연관규칙을 정의만으로 추천시스템을 구동할 수 있다. 또한 한명의 고객에 대해서는 여러 가지의 연관규칙이 적용될 수 있다. 이때에는 연관규칙의 순서를 정하는 알고리즘이 필요한데 본 논문에서는 연관규칙과 고객과의 유사정도를 수치로 나타내어 순위를 부여하는 세가지 방법을 제안하여 실험하였다. 또한 고객에게 적용되는 연관규칙을 결정하여 연관규칙의 상품을 추천하였을 경우 연관규칙과 고객과의 관계를 나타내는 수치를 조정하여 다음 번 추천에서는 보다 낮은 수치를 갖도록 하여 연관규칙의 순위를 낮추는 방법을 사용하였다.

일반적인 연관규칙은 나이나 몸무게와 같은 범위를 가지는 속성이 존재하는데 이러한 범위를 가지는 속성을 포함한 연관규칙은 속성의 구간이 구별되어있어 각 구간에 속한 고객들을 좀 더 세분화하여 고객과 연관규칙간의 관계를 계산하면 고객에게 더욱 적합한 연관규칙을 추천할 수 있다. 예를들어 "20대인 남성은 복권을 산다." 라는 연관규칙이 있을 경우 21세인 고객과 25세인 고객은 이 연관규칙과의 관계를 나타내는 수치가 같을 수도 있지만 일반적으로는 다른 수치를 갖는다. 이러한 경우 21세와 25세의 고객에게 다른 수치를 부여하면 좀 더 정확한 추천을 할 수 있게 된다.

본 논문은 다음과 같이 구성된다. 2장에서는 추천시스템에서 사용하는 연관규칙에 대하여 기술하고 3장과 4장에서는 연관규칙의 우선순위 부여 알고리즘과 추천시스템에 대하여 기술하고 5장에서는 실험 6장에서는 결론 및 향후 계획에 대하여 논한다.

2. 추천시스템에서의 연관규칙

2.1 연관규칙 정의

연관규칙을 추천 시스템에 사용하기 위해서는 먼저 추천시스템에 적합한 형식으로 표현되어야 한다.

추천 시스템에서 사용 가능한 연관규칙을 수식으로 나타내면 아래와 같다.

$Rule(R): \forall_{p,q} C => O$

R : 연관규칙

C: 연관규칙의속성을나타낸다.*Condition*이라부른다.

O: 연관규칙(R)에서*Condition*이만족할경우수행되는부분이다.

*Operation*이라부른다.

예를들어, 나이가 20대이고 컴퓨터 책을 구매한 사람은 DVD타이틀을 구매한다 는 연관규칙이 존재 할 경우 "나이 20대, 컴퓨터 책 구매" 는 연관규칙의 조건을 나타내고 본 논문에서는 Condition이라 하고 "DVD타이틀 구매" 는 연관규칙의 Condition이 만족할 경우 수행되는 부분이며 Operation이라 한다.

연관규칙을 좀 더 자세하게 나타내면 아래와 같다.

$\forall_{x,y,z} P_1(X,Y) \wedge P_2(X,Y) \cdots \wedge P_n(X,Y) \rightarrow R(X,Z = z_m)$

P: *Database*의*predicate*

X: 사용자집합

Y: $(v_i < V < v_j)$또는 $(V = v_k)$로나타낼수있다.

V가*continuous*한값을가지는속성집합일경우 $(v_i < V < v_j)$로나타내고

V가*distinct*한값을 가지는속성집합또는사용자가구매한상품의집합일경우

$(V = v_k)$로나타낸다.

Z: 전체상품의집합 $Z = z_1, z_2, z_3, \cdots, z_l, \cdots, z_m$

n : 규칙에서사용하는*predicate*의개수

추천시스템에서 사용되는 연관규칙은 상품 추천의 복잡도를 최소화 하기 위하여 Condition에는 여러 조건이 존재 할 수 있지만(남자이면서 CDP를 구매한 고객) Operation(DVD를 구매)에는 단 하나의 조건만 올 수 있다. 만약 Condition에 속하는 속성이 나이, 상품의 크기 같은 continuous한 값을 가지는 속성일 경우 범위를 나타낼 수 있도록 $(v_i < V < v_j)$ 와 같이 나타내고 성별 같은 distinct한 값을 가지거나 사용자가 구매한 상품을 나타낼 경우 $(V = v_k)$ 와 같이 나타낸다.

연관규칙은 각각 독립적이다. 그 이유는 아래와 같은 형식의 연관규칙들이 존재한다고 하면

$$X \rightarrow Y\,(1)$$
$$Y \rightarrow Z\,(2)$$
$$X \rightarrow Z\,(3)$$

연관규칙 (1)과 (2)가 유용한 연관규칙일 경우 연관규칙(3)도 유용한 연관규칙일 것이라는 추측이 가능하다 꼭 성립된다고 할 수 는 없다. 왜냐하면, 연관규칙이 유용한 연관규칙이 되기 위해서는 연관규칙의 조건에 맞는 트랜잭션의 발생 수치가 시스템에서 정한 임계치를 넘어야 하는데, 연관규칙 (1), (2)의 발생 수치가 임계치를 넘었다고 해서 규칙(3)의 발생수치가 임계치를 넘어서 유용한 연관규칙이 된다고 볼 수 없기 때문에 (1), (2)를 통해 (3)을 추측한다고 해도 (3)이 유용한 연관규칙이 된다고 볼 수 없다. 예를 들어 나이가 20대인 고객은 CDP를 구매한다는 규칙(ㄱ)와 CDP를 구매한 사람은 배터리는 구매한다

는 규칙(ㄴ)이 있다고 가정 할 경우 나이가 20대인 고객은 배터리를 구매한다는 규칙(ㄷ)을 추측을 통해 알 수 있지만 유용하게 사용할 수 있는 연관규칙이 될 수 없다.

본 논문에서는 연관규칙을 분석하여 연관규칙이 적용되는 고객을 선별하여 선별된 고객에게 연관규칙을 추천한다고 하는데 이는 연관규칙의 Operation에 해당되는 상품을 추천한다는 것이다. 예를 들면, 나이가 20대인 서울에 사는 남자는 CDP를 구매한다는 연관규칙이 존재하고 실제 고객중에 나이가 20대이며 서울에 사는 남자가 존재할 경우 이 고객은 위의 연관규칙에 적용된다고 하고 이 고객에게 이 연관규칙을 추천한다고 하면 CDP를 추천한다는 것을 의미한다.

2.2 연관규칙의 사용

일반적으로 데이터 마이닝에서 연관규칙의 유용성을 측정하는 기준은 지지도(Support)와 신뢰도(Confidence)이다. 지지도는 전체 거래에 대해서 두 개의 상품을 동시에 구매한 거래의 비율을 의미하며, 상품 A와 상품 B가 동시에 구매된 거래의 수를 전체 거래수로 나누어서 구한다. 이를 식으로 나타내면 아래와 같다.

$$\text{지지도}(Support) = P(A \cap B) = \frac{A\text{와 }B\text{를 동시에 포함하는 거래의 수}}{\text{전체 거래수}}$$

연관규칙의 유용성을 측정하는 또 하나의 기준인 신뢰도는 상품A가 포함된 거래비율 중 상품A와 B가 동시에 포함된 거래의 비율이다. 즉, 신뢰도는 조건 상품이 나타나면 결과 상품이 나타날 확률을 의미한다. 신뢰도를 식으로 나타내면 아래와 같다.

$$\text{신뢰도}(Confidence) \approx P(B \mid A) = \frac{A\text{와 }B\text{를 동시에 포함하는 거래의 비율}}{A\text{를 포함하는 거래의 비율}}$$

연관규칙은 지지도나 신뢰도의 값이 시스템에서 지정한 임계치를 넘을 때 유용하다고 본다. 일반적으로 지지도는 얼마나 많은 고객이 지지하는 가를 나타내는 것으로 1000명의 사용자가 있을 경우 그 중 10명이 연관규칙에 적용된다고 하면 지지도는 1%가 되는 것이고 100명이 연관규칙에 적용된다고 하면 10%가 되는 것이다. 이 규칙이 실제 추천에서 사용된다고 할 때 1%일 때는 추천 효과가 미미할 것이지만 10%일 때는 더 많은 고객에게 적용되어 많은 효과를 볼 수 있다. 추천 시스템에서 사용하기 위한 연관규칙은 지지도의 임계치를 정하여 유용한 연관규칙을 선별한다. 신뢰도는 연관규칙이 얼마나 유용한 규칙이 될 수 있는 지를 나타낸다. 위의 예에서 1000명의 사용자 중 100명이 연관규칙에 적용되고 이 연관규칙의 Condition만 만족하는 사용자가 150명일 경우에는 신뢰도가 67%이고 500명일 경우에는 신뢰도가 20%가 된다. 이 때 신뢰도가 높을 경우 연관규칙에서 Condition을 만족하는 사용자에게 Operation의 상품을 추천하면 67% 또는 20%의 고객이 구매를 한다는 결론이 나온다. 지지도와 신뢰도의 정의에서 지지도는 연관규칙이 추천에 사용될 경우 어느 정도의 효과를 낼 수 있는 가를 측정하고 신뢰도는 연관규칙이 얼마나 유용한가를 측정한다.

연관규칙의 Condition에 해당되는 조건을 만족하는 고객을 연관규칙에 적용되는 고객이라 한다. 임계치를 넘은 연관규칙, 즉 유용한 연관규칙은 Condition을 만족하는 모든 고객에게 적용 될 수 있다. 그러므로 하나의 연관규칙을 다수의 고객에게 적용될 수 있다. 이와는 반대로 시스템 내에 유용한 연관규칙이 다수 존재할 경우 하나의 고객에게 여러 개의 연관규칙이 적용될 수 있다. 이렇게 하나의 고객에게 여러 개의 연관규칙이 적용될 경우 고객에게 추천할 연관규칙의 순서를 정하는 작업이 필요하다. 다시 말해 고객에게 추천할 연관규칙들에게 우선순위를 부여하여 우선순위가 높은 연관규칙을 고객에게 추천하는 등의 작업이 필요하다. 일반적으로 연관규칙의 우선순위는 연관규칙의 신뢰도에 영향을 받는다. 그 이유는 앞에서 말한 바와 같이 신뢰도는 연관규칙이 얼마나 유용한 가를 나타내는 척도로서 같은 수의 고객에게 높은 신뢰도를 갖는 연관규칙과 낮은 신뢰도를 갖는 연관규칙을 추천할 경우 높은 연관규칙을 갖는 연관규칙을 추천한 고객이 추천된 연관규칙의 Operation에 정의된 상품을 더 많이 구매하는 결과를 볼 수 있다.

고객이 연관규칙에 대한 우선순위를 나타내는 값을 고객의 연관규칙에 대한 적합도라고 하며 이 적합도를 행렬로 나타낼 수 있다. 아래 행렬은 고객과 연관규칙과의 적합도를 행렬 $W(m \times n)$으로 표현한 것이다.

$$W = \begin{bmatrix} w11 & w12 & \Lambda & w1(n-1) & w1n \\ w21 & w22 & \Lambda & w2(n-1) & w2n \\ M & M & cij & M & M \\ w(m-1)1 & w(m-1)1 & \Lambda & w(m-1)(n-1) & w(m-1)n \\ wm1 & wm2 & \Lambda & wm(n-1) & wmn \end{bmatrix}$$

m: 고객

n: 연관규칙

wij: 고객i의 연관규칙j에 대한 적합도

추천시스템에서 고객 i에게 연관규칙을 추천할 때 다시 말해 연관규칙의 Operation에 정의된 상품을 추천할 때 고객 i를 기준으로 적합도가 높은 연관규칙을 내림차순으로 정렬하여 상위 N개의 연관규칙을 추천한다. 고객에게 좋은 연관규칙이라 함은 신뢰도가 높은 규칙을 의미하지만 연관규칙의 Condition에 사용되는 속성의 특징을 고려하면 좀 더 좋은 결과를 얻을 수 있다. 속성의 특징을 고려한다 함은 속성 중에 연속적인 값을 가지는 것들은 규칙이 일반적으로 나이일 경우 20대, 30대 등으로 나타내어 지는데 이러한 속성을 포함하는 연관규칙이 적용되는 고객일 경우 이 연관규칙이 21세의 고객과 25세의 고객에게 미치는 영향은 각기 다르다고 할 수 있다. 그림 (1)에서 연관규칙을 하나의 클러스터로 볼 경우 user i와 user j는 이 연관규칙에 대해 다른 신뢰도 값을 가진다고 볼 수 있다.

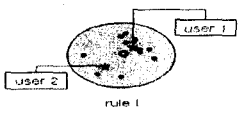

그림 1 연관규칙 클러스터

이렇게 범위를 갖는 속성들을 고려하여 고객의 연관규칙에 대한 적합도를 구하기 위해서는 연관규칙의 신뢰도 뿐만 아니라 다른 값을 적용하여 연관규칙의 우선순위를 정해야 한다.

3. 추천 알고리즘 소개

본 논문에서는 연관규칙의 우선순위를 정하기 위한 적합도 값을 계산하는 세 가지 방법을 제안한다. 첫 번째 방식은 연관규칙 내에 고객의 위치에 따라 다른 적합도를 부여하는 방식이다. 식(1)은 그림 (1)에서 보는 바와 같이 연관규칙 클러스터내에 고객의 위치에 따른 적합도를 계산하는 식을 나타낸다.

$$W_{ij} = \frac{\left(\frac{\vec{u_i} \cdot \vec{r_j}}{|\vec{u_i}| \times |\vec{r_j}|}\right) \times \frac{n_j}{N_j}}{1 + k_{ij}} \quad \text{---------식(1)}$$

$rule$: 연관규칙, $if\ condition\ then\ operation$

W_{ij}: $user\ i(ui)$에 대한 $rule\ j(rj)$의 가중치

u_i: i번째 $user$

r_j: j번째 연관규칙

k_{ij}: $rule\ i$와 $user\ j$에 대한 이전에 추천된 횟수

n_j: $rule\ j$에서 $condition$과 $operation$을 만족하는 사용자의 수

N_j: $rule\ j$에서 $condition$을 만족하는 사용자의 수

적합도 계산 식(1)에서 $\left(\frac{\vec{u_i} \cdot \vec{r_j}}{|\vec{u_i}| \times |\vec{r_j}|}\right)$은 연관규칙과 고객의 유사 정도를 Cosine계산을 이용하여 나타내었다.[4] $\frac{n_j}{N_j}$은 연관규칙 j의 신뢰도를 나타낸다. 위의 식에서 고객의 범위를 가지는 속성이 연관규칙 클러스터의 중앙에 위치할 수록 다시 말해 나이 속성일 경우 범위의 중간 값을 가질 경우 더 높은 적합도 값을 가지고 신뢰도가 높을 수록 높은 적합도 값을 가진다고 볼 수 있다. 그리고 한번 추천된 연관규칙은 $1+k_{ij}$에 의하여 선호도가 낮아진다. 예를들면 나이가 20대이며 몸무게가 70-80키로그램인 남성은 맥주를 산다. 라는 연관규칙이 존재할 경우 나이가 20대이며 몸무게가 70-80키로그램인 고객 중 나이는 25세이고 몸무게가 75인 고객에게 다른 고객보다 높은 적합도를 준다. 또한 이 고객에게 위의 연관규칙이 높은 우선순위를 가져서 계속 추천되는 것을 방지하기 위하여 한번 추천된 연관규칙의 적합도 값을 낮춰서 연관규칙의 우선순위를 조정하여 계속적으로 추천되는 것을 방지한다.

적합도를 계산하는 두 번째 방식은 연관규칙 클러스터 내에서 일정 범위 내에 존재하는 고객들의 밀집도에 따른 적합도를 계산하는 방식이다. 그림(2)와 같이 연관규칙 클러스터 내에서 일정범위 내에 다른 고객이 많이 존재하는 user i가 그렇지 않은 user j보다 집합도가 높다고 볼 수 있다.

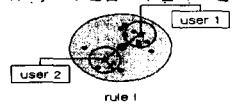

그림 2 연관규칙 클러스터

식(2)는 고객의 위치와 밀집도에 따른 적합도를 계산하는 식이다.

$$W_{ij} = \frac{\left(\frac{\alpha \cdot N(l_{ij})}{N(r_i)} \times \frac{\vec{u_i} \cdot \vec{r_j}}{|\vec{u_i}| \times |\vec{r_j}|}\right) \times \frac{n_j}{N_j}}{1 + k_{ij}} \quad \text{---------식(2)}$$

l=연관규칙클러스터의정규화된거리

$$l = N(ri) \times \frac{\beta}{100}$$

$N(l_{ij})$: $rule\ i$에 속하는 $user$중 $user\ j$에서 거리 l안에있는 $user$의 수

$N(ri)$: $rule\ i$에속하는 $user\ j$의수

α: 선호도 최적화 변수

β: l을 구하기위한 최적화변수

식(2)에서 $\left(\frac{\alpha \cdot N(l_{ij})}{N(r_i)} \times \frac{\vec{u_i} \cdot \vec{r_j}}{|\vec{u_i}| \times |\vec{r_j}|}\right)$는 user i의 주위에 고객이 많을 경우 $\frac{\vec{u_i} \cdot \vec{r_j}}{|\vec{u_i}| \times |\vec{r_i}|}$와 $N(r_i)$는 일정하지만 $N(l_{ij})$가 높아지기 때문에 높은 적합도 값을 가지게 된다. 예를들어 두명의 고객 i, j에서 거리 30이내의 고객의 수가 i의 주위에는 5명이 존재하고 j의 주위에는 2명이 존재할 경우 고객 i가 현재의 연관규칙과 관련하여 좀 더 높은 적합도를 가진다고 볼 수 있다.

적합도를 계산하는 세 번째 방식은 연관규칙 클러스터 내에서 고객과 가까운

위치에 존재하는 n명의 다른 고객들의 거리를 이용하여 적합도를 계산한다. 그림 (3)에서 n이 4일 때 user i와 가까운 4명의 고객의 거리의 평균은 user j와 가까운 4명의 고객의 거리의 평균보다 작기 때문에 user i의 적합도가 고객 j의 적합도보다 높다고 할 수 있다.

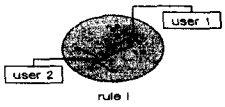

그림 3 연관규칙 클러스터

식(3)은 고객의 위치와 주위 고객과의 거리에 따른 선호도를 계산하는 식이다.

$$W_{ij} = \frac{\left(\alpha \frac{m}{\sum_{x=1}^{m} Dis(u_i, u_x)} \times \frac{\vec{u_i} \cdot \vec{r_j}}{|\vec{u_i}| \times |\vec{r_j}|} \right) \times \frac{n_j}{N_j}}{1 + k_{ij}} \quad \text{식(3)}$$

m : 클러스터에 정규화된 user 의 수

$m = N(r_i) \times \frac{\beta}{100}$

$N(r_i)$: $rule\ i$에 속하는 $user$ 의 수

$Dis\ (u_i, u_x)$: u_i와 u_x의 거리

α : 선호도 최적화 변수

β : m를 구하기 위한 최적화 변수

식(3)에서 $\left(\alpha \frac{m}{\sum_{x=1}^{m} Dis(u_i, u_x)} \times \frac{\vec{u_i} \cdot \vec{r_j}}{|\vec{u_i}| \times |\vec{r_j}|} \right)$에서 $\alpha \frac{m}{\sum_{x=1}^{m} Dis(u_i, u_x)}$는 user i와 이 user와 가까운 m명의 고객과의 거리의 평균의 역수를 나타내는 것으로 m명의 고객이 가까울수록 값이 커지게 되어 결과적으로 전체 적합도의 증가를 가져온다. 식(3)에서 고객이 연관규칙의 중심에 가까울수록 주위에 고객이 가까이 있을 수록 신뢰도가 높을수록 높은 적합도를 가지게 되어 고객에게 추천될 확률이 높아진다. 예를들면 나이가 20대인 남자는 ODP를 산다 라는 연관규칙이 존재할 경우 나이가 25이면서 나이가 22~28이며 남자인 고객이 많을 경우 적합도가 높아져서 이 연관규칙은 이 고객에 관한 높은 우선순위를 갖게 된다.

4 추천 방법 정의

3.2에서 제시한 방식으로 범위를 가지는 속성을 고려하여 고객과 연관규칙과의 적합도를 연산하여 고객에게 적절한 상품을 추천할 수 있게 된다. 아래는 추천하는 과정을 나타내는 알고리즘이다.

사용자 (U)와 연관규칙 (R)과의 선호도를 행렬 $W(m \times n)$으로 나타내면

$$W = \begin{bmatrix} w_{11} & w_{12} & w_{13} & \Lambda & w_{1n} \\ w_{21} & w_{22} & w_{23} & \Lambda & w_{2n} \\ M & M & M & w_{ij} & M \\ w_{m1} & w_{m2} & w_{m3} & \Lambda & w_{mn} \end{bmatrix}$$

이 된다.

Wij :사용자 (U) i에 대한 연관규칙 (R) j의 선호도

```
WList  :사용자와  연관규칙과의      선호도 집합
WList  = (W11,  W12,  Λ  , W1m,
          W21,  W22,  Λ  , W2m,
                 M
          Wn1,  Wn2,  Λ  , Wnm)
KList  :사용자와  연관규칙과의    이전 추천 횟수
KList  = (k11,  k12,  Λ  , k1m,
          k21,  k22,  Λ  , k2m,
                 M
          kn1,  kn2,  Λ  , knm)
Default  :시스템  기본 추천 아이템
Default  :(d1, d2, Λ , dN)

사용자 (U) x가 로그인 하였을 때
WList x = Desc_Sort( Wx1, Wx2, Wx3, Λ , Wxm)
WLis t'x = Select  TOP  N  (WList x)
For p = 0 to Count(WLis  t'x)
        Recommend(  WLis t'x[p])
        kx = kx + 1

        Value (WLis t'x[p]) = sim(Ux,  WLis t'x[p]) × n/N
                              ─────────────────────
                                    1 + kx
End For
For q = Count(WLis  t'x) - 1 to N
        Recommend(  Default[q] )
End For
```

위의 추천 알고리즘은 고객 x가 로그인 하면 행렬 $W(m \times n)$에서 모든 연관규칙에 대한 고객 x의 적합도를 추출하여 x에게 적합도가 가장 높은 N개의 연관규칙을 선별한다. 이때 연관규칙이 N개가 안될 경우 시스템에서 정해놓은 기본규칙(Default)을 사용자에게 추천한다. 추천된 연관규칙에 대해서는 다음 추천 시 다시 추천되는 것을 방지하기 위해 적합도를 줄여서 고객 x에게 상품을 추천할 경우 같은 연관규칙이 추천되는 것을 방지한다.

5. 실험

5.1 실험 환경 및 데이터집합

실험은 펜티엄4-1.6Ghz에서 이루어 졌으며 사용언어는 Java이고 사용한 데이터베이스는 MSSQL2000 이다. 데이터 마이닝 툴로는 공개 툴인 Weka라는 툴을 이용하였으며 실험에 사용한 데이터집합은 [5]에서 사용된 구매 데이터이다. [5]의 데이터는 3,465개의 instance와 220개의 속성을 가지고 43개의 범위를 가지는 속성이 있다. 나이, 연봉, 상품 가격 등이 범위를 가지는 속성으로 사용되었고 자동차소유여부, 대출금여부 등이 범위를 가지지 않는 속성으로 사용되었다. 본 논문에서는 instance를 고객 한명의 특징(속성)과 구매정보를 포함하는 데이터로 간주하여 실험하였다.

5.2 실험

실험은 10-fold cross-validation 방법을 사용하여 3,465개의 instance를 90%의 Traning Set과 10%의 Test Set으로 데이터를 분류하여 Training Set으로 Weka라는 데이터 마이닝 툴을 이용하여 일반적으로 전자상거래에서 연관규칙을 추출할 때 사용되는 수치인 지지도 0.2, 신뢰도 0.8을 기준으로 연관규칙 200개를 추출 한 뒤 Test Set의 instance를 이용하여 적합도를 이용한 행렬(고객 × 연관규칙)을 구성한다. 구성된 행렬을 사용하여 개별 고객에게 상위 N개의 연관규칙을 추천한 뒤 고객이 실제로 추천된 아이템을 구매했는지를 평가하였다. 평가에는 범위를 가진 속성을 고려하지 않은 경우와 범위를 가진 속성을 고려한 경우를 구분하여 추천한 상품 수에 대한 실제로 구매한 상품 수의 비율을 추측값의 평균절대오차(MAE)를 사용하여 비교하였다.

5.3 결과 분석

본 논문에서는 사용자의 연관규칙에 대한 선호도를 계산하여 추천 할 때 범위를 가지는 속성을 고려하지 않은 것과 범위를 가지는 속성을 고려하였을 때 추천 결과 각각 사용자에게 추천 되었을 때 추천이 적합하지 않을 확률(에러율이라 함)의 차이를 측정하였다. 그림(4)에서 범위를 가진 속성을 고려하지 않은 방법은 추천되는 연관규칙의 수에 관계 없이 전체적으로 비슷한 에러율을 가지는데 반해 범위를 가진 속성을 고려한 방법에서는 그렇지 않은 경우보다 최소 0%에서 최대 9%까지 에러율이 줄어든 것을 볼 수 있다. 또한 연관규칙의 수가 적으면 적을 수록 에러율이 최대 9%까지 차이가 난다. 이는 현재 추천 시스템을 사용하는 곳이나 개인화 웹페이지를 제공해주는 곳에서 많은 수의 추천보다는 적은 수의 상품 등을 추천하는데 사용이 적합하다고 할 수 있다.

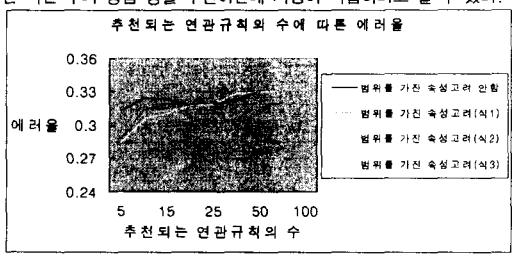

그림 4 추천되는 연관규칙 수에 따른 에러율

6. 결론 및 향후 계획

기존의 추천시스템에서는 협업을 통한 추천과 내용 기반의 추천으로 추천을 하였으나 이러한 방법을 사용하기 위해서는 시스템 구축 후 고객의 취향과 구매정보가 일정량 이상 축적 되어야 추천을 할 수 있다. 하지만 연관규칙을 사용하는 추천에서는 사용하려는 시스템에서 연관규칙을 추출해 낼 수도 있지만 기존에 존재하는 신뢰할 만한 쇼핑몰 등에서 연관규칙을 추출해내어 새로 구축하려는 쇼핑몰이나 추천시스템을 사용하려는 곳에 곧바로 적용가능 하다는 장점이 있다.

본 논문에서는 연관규칙을 사용하여 사용자에게 상품을 추천하는 방식을 사용하였는데 연관규칙의 속성 중 범위를 갖는 속성과 범위를 갖지 않는 속성을 구별하여 사용자의 상품에 대한 적합도를 계산하여 더욱 정확한 추천이 이루어지도록 하였다. 실험에서는 적합도를 계산하는데 세가지 방법을 사용하였는데 실험 데이터를 다양한 방식으로 추출하여 실험하여 보다 정확한 추천이 이루어지도록 하는 부분을 향후 과제로 남겨두겠다.

참고문헌

[1] Wu, Y. H. and Chen, A. L. P. Index Structures of User Profiles for Efficient Web Page Filtering Services. In Proceedings of IEEE Conference on Distributed Computing Systems, pp. 644-651, 2000

[2] Goldberg, D., Nichols, D., Oki, B. M. and Terry, D., Using Collaborative Filtering to Weave an Information Tapestry, Communications of the ACM, 35(12): 61-70, 1992

[3] Agrawal, R., Imielinski, T., and Swami A. Mining association rules between sets of items in large databases. In Proceedings of the ACM SIGMOD Conference on Management of Data 207-216, 1993

[4] Salton, G., Wong, A., and Yang, C. S. A vector space model for automatic indexing. Communications of the ACM 18(11), ACM Press, 613-620, 1975

[5] Zheng, Z., Kohavi, R. and Mason, L. Real World Performance of Association Rule Algorithms. In Proceedings of the Seventh ACM-SIGKDD International Conference on Knowledge Discovery and Data Mining,, 2001