속성추출을 이용한 협동적 추천시스템의 개발

유상종*,권영식*

*동국대학교 산업공학과

Development of a Collaborative Recommendation System using feature selection

Yoo SangJong, Kwon, Young S.

Dongguk University, Dept. of Industrial Engineering

E-mail : sjyoo408@dongguk.ac.kr, yskwon@dongguk.edu

요 약

전자상거래의 급속한 발달로 인하여 많은 상품이 거래가 되로 있다. 기업은 상품들 가운데서 적절한 상품을 고객에게 추천하기 위해서 추천시스템을 개발을 하였다. 그러나 사용자와 고객의 수가 급증하면서 추천을 위해서 많은 시간과 비용이 들게 되었다. 본 논문에서는 이러한 확장성의 문제점을 해결하기 위해서 속성추출방법을 추천시스템에 적용하여 추천의 시간을 단축하여 확장성의 문제를 해결하고자 개선된 추천시스템을 개발했다. 개선된 추천시스템의 추천속도는 기존의 추천시스템에 비하여 빠른 추천이 가능하게 되었다. 이로 인해 확장성의 문제를 해결할 수 있게 되었다.

1. 서론

우리들은 급변하는 정보화 시대에 살고 있다. 얼마 전까지만 해도 컴퓨터 분야 전문가들만의 영역으로 여겨졌던 인터넷이 컴퓨터 통신망의 발달과 함께 우리 생활에 밀려 왔다. 이러한 인터넷의 발달과 대중화로 인해서 인터넷 사용자들은 국경과 시간의 제약을 넘어 전 세계적으로 자유로운 소통을 할 수 있게 되었고 전자상거래가 급성장하게 되었다.

전자 상거래의 시장 규모가 성장함으로 취급되는 상품의 수와 종류가 증가되었다. 전자상거래 운영자는 고객의 다양한 구매 정보로 인하여 사용자가 무엇을 좋아하고 싫어하는지에 관한 성향을 분석하는데 많은 어려움을 느끼게 되었다. 이런 성향 분석의 어려움을 해결하기 위해서 전자상거래 사이트나 포털 사이트의 운영자에 의해서 제시된 시스템이 추천 시스템(recommendation system)이다[1,3].

추천 시스템의 방법으로는 인구통계학적 자료를 이용한 추천, 내용기반 추천, 항목기반 추천, 협동적 추천 등의 방법을 이용하여 추천 시스템을 운영하고 있다. 이중에서 협동적 추천 방법을 이용한 시스템이 가장 일반화된 개인화 추천 시스템이다.

협동적 추천 방법은 사용자 사이의 유사도을 파악하여 추천에 이용하는 기법이다. 사용자가 구매한 상품의 정보를 기반으로 유사도가 높은 집단을 형성하고 집단에 속한 사용자의 정보를 이용하여 특정 사용자를 위한 추천에 사용하는 방법이다[4]. 협동적 추천 방법을 이용한 경우의 문제점으로 시스템의 초기에는 사용자와 사용자의 구매 정보가 매우 적기 때문에 발생하는 자료의 희소성에 대한 것을 들 수 있고, 시스템의 성장후의 문제점으로는 급속히 증가하는 사용자와 구매정보 때문에 빠른 추천을 할 수 없다는 확장성의 문제점을 들 수 있다.

본 논문에서는 협동적 추천 시스템의 확장성 문제를 해결하고자 하였다. 협동적 추천 시스템은 사용자간의 유사도를 비교하기 위해서 각각의 사용

자의 모든 구매정보를 모두 이용하여 유사도를 계산한다. 이러한 계산 방식은 시스템의 초기에는 데이터가 많지 않기 때문에 시스템의 성능에 지장을 주지 않지만 시스템이 성장한 후에는 추천을 위해서 많은 시간과 비용이 들게 된다. 이러한 문제를 해결하기 위해서 본 논문에서는 속성추출(feature selection) 방법을 이용해서 사용자의 구매 정보 중에서 유사도의 계산에 많은 영향을 주는 중요한 구매정보를 추출해서 사용함으로 추천시스템의 성능을 향상시키고자 하였다. 그리고 본 논문에서 제시한 속성추출기법을 이용한 추천방법의 경우와 기존의 협동적 추천방법의 성능을 비교 평가하는 연구를 했다.

2. 관련연구

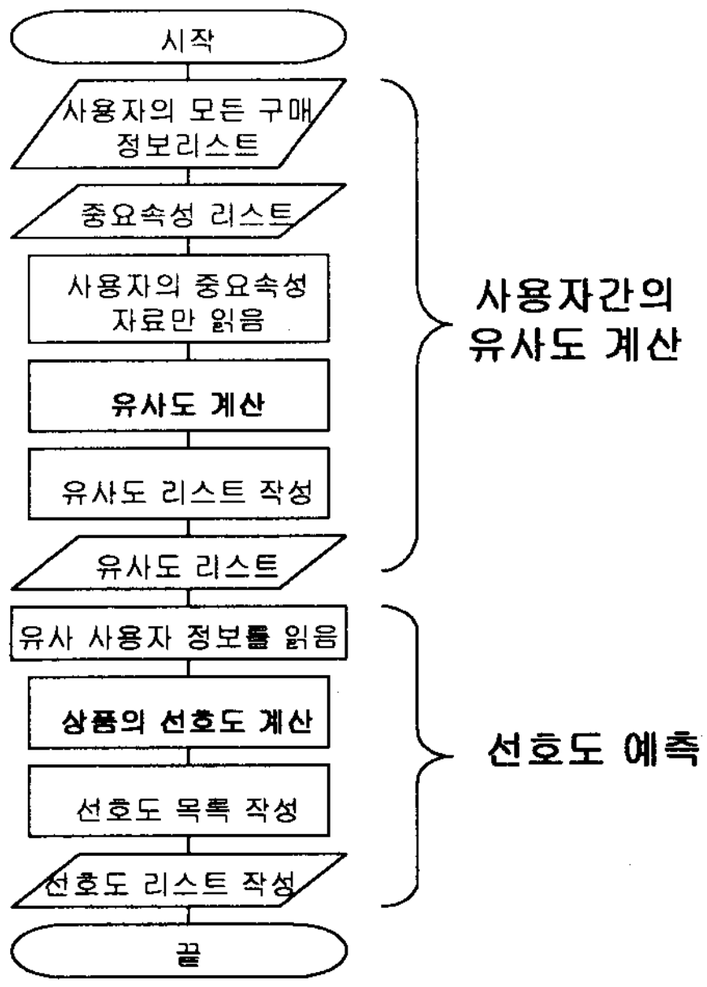
본 장에서 협동적 추천과 추천시스템에 대하여 소개를 한다.

2.1 협동적 추천

협동적 추천 방법은 사용자들 사이의 유사도를 구하여 유사도가 높은 사용자들이 선호한 상품을 추천하는 방법이다. 사용자들 사이의 유사도를 구할 때는 사용자가 상품에 대하여 이전에 평가한 선호도 정보를 이용하고, 추천하고자 하는 상품을 구할 때에는 현재 사용자가 평가하지 않은 상품에 대하여 유사도가 높은 사용자들이 선호한 상품을 추천하고자 하는 상품으로 지정하므로 상품에 대한 내용이 없어도 사용자에 대한 상품 추천이 가능하다.

협동적 추천은 [그림1]과 같고 특정 사용자에게 추천을 하는 과정은 유사도 계산단계와 사용자의 선호도를 계산하는 두 단계로 이루어진다.

사용자간의 유사도를 계산하기 위한 방법으로는 본 논문에서는 벡터 유사도(Vector Similarity)를 사용해서 사용자사이의 유사도를 구하는데 사용했다. 피어슨 상관 계수는 여러 분야에서 많이 사용되고 있지만 사용자의 평가수가 적은 경우에 사용자의 평가가 모두 일치하거나 하나의 평가만 존재하게 됨으로 모든 사용자의 평가와 평가 평균의 차이가 0



[그림1] 협동적 추천 흐름도

이 되면 다른 사용자의 평가에 관계없이 유사도는 0이 된다. 이러한 문제를 피하고자 벡터 유사도를 본 논문에서는 사용했다. 벡터 유사도 공식은 다음의 식(1)와 같다.

$$Sim(Q,D) = \frac{Q \cdot D}{|Q| \cdot |D|} \qquad \text{식(1)}$$

Q, D는 사용자가 상품에 대하여 평가한 데이터 벡터를 말한다.

사용자의 선호도를 계산하는 단계에서는 유사도 계산단계에서 작성된 유사도 목록을 기초로 선호 상품을 예측하여 추천하게 된다. 선호도 예측단계에서 선호도 예측을 위해서 사용될 유사한 사용자의 수를 결정하는 방법으로 사용자간의 유사도가 일정값 이상인 사용자들을 사용하는 방법과 특정 사용자와 유사한 n명의 이웃을 사용하여 예측하는 방법이 있다. 본 논문에서는 특정 사용자와 유사한 n명의 이웃을 사용하는 예측방법을 사용하였다. 선호도 값을 예측하는데 식(2)을 사용한다.

$$P_{a,i} = \overline{r_a} + \frac{\sum_{r_{u,i}}^{n} (r_{u,i} - \overline{r_u})\, w_{a,u}}{\sum_{u=1}^{n} w_{a,u}} \quad \text{식(2)}$$

여기서, $w_{a,u}$ 는 사용자 a의 현재 사용자에 대한 유사도 값, $\overline{r_a}$는 사용자 a의 선호도 값의 평균, $r_{u,i}$는 사용자 u의 i번째 상품에 대한 평가값이고, $P_{a,i}$는 사용자 a의 i번째 상품에 대한 예측값을 말한다.

협동적 상품 추천에 대한 연구로 Breese[7]는 피어슨 관계 계수와 벡터 유사도(vector similarity) 방법을 사용하고 그들 각각에 기본 값, 역 사용자 빈도(inverse user frequency), 사례확대(case amplification)의 방법을 사용하여 정확도와 유효 범위를 향상시키는 것에 관한 연구와 기존의 확률적 방법인 베이시안(Bayesian) 방식의 모델 기반(model-based) 방법을 협동적 방법에 응용하여 연구를 수행하였다. Helocker[6]는 다양한 방식의 유사도 계산과 여러 가지 방식의 유사도 가중치 실험을 하였다. 유사도 계산에는 피어슨 상관 계수, 스피어만 관계 계수, 벡터 유사도를 이용하고, 선호도 값을 구하는 방법으로는 평균 가중치(average rating), 유사 사용자의 상품 선호도 가중치 합(deviation from mean), z 평균 점수(z score average)방법을 이용하여 실험을 했다. 실험 결과로써, 유사도를 구할 때 평가하는 선호도 값의 범위가 연속적인 경우에는 피어슨 관계 계수를 이용하는 것이 높은 정확도를 나타냈고, 선호도 값을 구할 때는 전체적으로 유사 사용자의 상품 선호도 가중치 합을 이용하는 것이 높은 정확도를 나타냈다. Billsus[11]은 상품 추천에 소요되는 시간을 단축하기 위하여 특이행렬분해(singular value decomposition)방법을 이용하여 상품의 차원을 줄임으로써 시간을 단축하고 유효 범위를 개선하고자 하였다.

2.2 속성 추출(feature selection)

2.2.1 문서 빈도 수

문서 빈도수(document frequency)는 하나의 속성이 얼마나 자주 학습 셋에 나타났는지를 말한다. 학습 셋(training set)에서 각각의 구별되는 속성에 대해서 학습 셋에서의 빈도 수를 계산한 후 정해 놓은 임계값보다 작은 값을 가지는 속성을 제거하게 된다. 문서 빈도수는 학습 셋에 희박하게 나타나는 속성이 학습 셋의 분류를 예측하는데 정보력을 가지고 있지 못하다는 기본적인 가정을 기반으로 이루어진다. 임계값을 기준으로 희박한 속성을 제거함으로서 속성 공간의 차수를 줄이게 된다.

문서 빈도수 방법은 속성 수를 줄이는 가장 간단한 방법이다. 하지만 대개 효율성을 향상시키는데 있어 정보력 있는 속성을 선택하는 기준이 있는 것이 아니라 시행착오적인 방법으로서 속성의 선택기준을 얻게 된다. 또, 문서 빈도수는 전형적으로 빈도가 높은 속성을 제거하는데 사용되지 않는다. 그리고, 낮은 빈도를 가지는 속성들은 비교적 정보력이 크기 때문에 상당수를 제거하지 말아야 하지만 실제 실험 결과에서는 좋은 결과를 보여주고 있다[11]. 학습 셋에서 i번째 사용자 $User_i$에 속성 t가 존재하면 1, t가 존재하지 않으면 0이 되며 식(3)과 같다.

$$DF(t) = \sum_{i=1}^{n} User_i(t) \quad \text{식(3)}$$

2.2.2 엔트로피(Entropy)

엔트로피는 임의의 학습 셋에서 불순도를 측정하는 수단이다. 예를 들면 n개의 사건이 각각 발생할 확률이 같다면 하나의 사건이 발생할 확률(p)은 $p=1/n$이 된다. 이때 하나의 사건을 구별하기 위해 필요한 정보량은 $-\log(p)=\log(n)$이 된다. 가령 16개의 사건이 있으면 $\log(16)=4$, 즉 4비트가 있으면 16개의 사건을 모두 구별할 수 잇게 된다. 여기서 사건을 구별하는데 필요한 정보량을 엔트로피라 하는데 이것은 "Information Theory"에서 Claude Shannon에 의해서 소개된 개념으로서 엔트로피가 클수록 얻을 수 잇는 정보력이 적음을 의미하며 따라서 메시지를 구별하는데 있어서 더 많은 정보가 필요함을 의미한다.

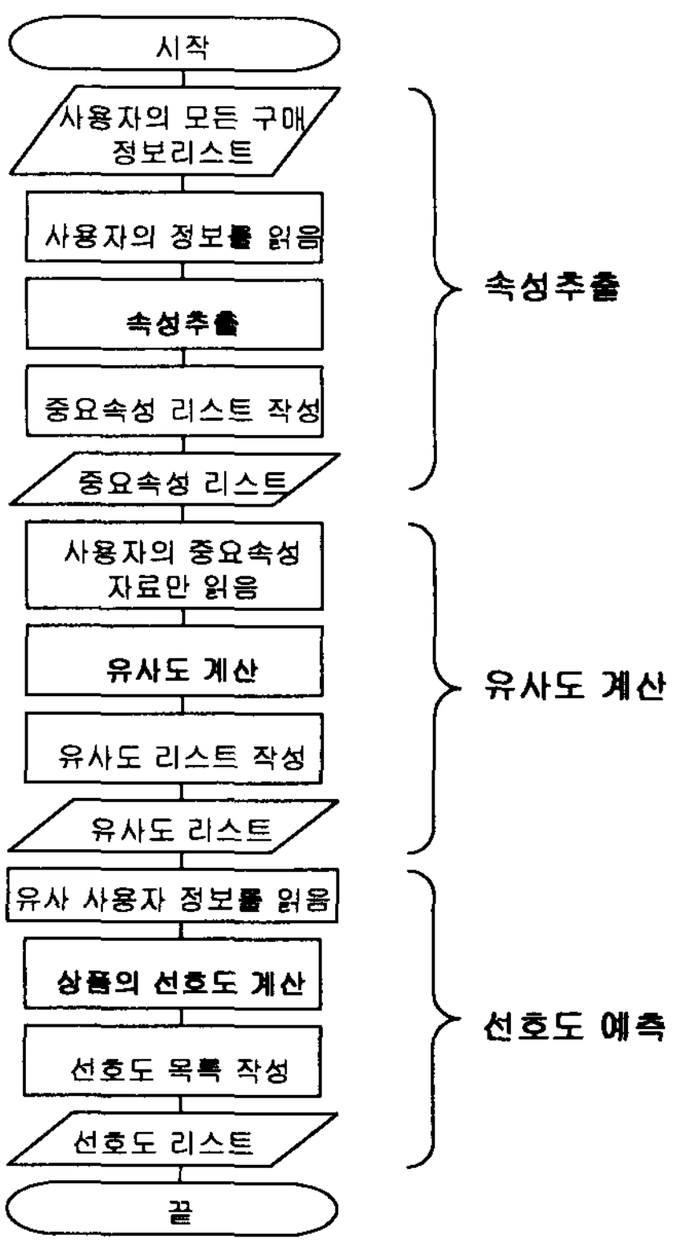
목적 속성 값이 불리언 값을 갖는 경우의 예를 들면, 신용평가에서 신용상태가 양호와 불량으로 분류하거나, 고객을 우수와 불량으로 나누는 경우와 같이 목적 속성 값이 2개의 값만 가지는 경우 엔트로피는 식(4) 같이 나타낼 수 있다.

$$H(S) \equiv -p_+ \log_2 p_+ - p_- \log_2 p_- \qquad \text{식(4)}$$

여기서 S는 학습 셋이고 p_+는 학습 셋에 포함된 긍정적 예의 확률을 말하는 것이고 p_-는 학습 셋에 포함된 부정적 예제 확률을 말한다.

3. 속성추출을 이용한 추천 시스템

본 논문에서 개선한 추천시스템은 기존의 추천시스템에 속성추출방법을 적용한 것으로 흐름도는 [그림2]와 같다.



[그림2] 속성추출 적용 추천시스템

본 논문에서 속성추출단계에 사용한 방법으로는 문서 빈도수와 엔트로피를 이용하였다. 문서 빈도수는 식(3)로 구할 수 있고, 엔트로피는 식(4)로 구할 수 있다.

4. 실험

4.1 실험 자료 및 방법

실험에 사용된 자료는 미네소타 대학 (the University of Minnesota)의 GroupLens Research Project에 의해서 수집된 MovieLens의 자료다[17]. 사용자 943명이 1682개의 영화에 대하여 1부터 5까지 평가한 100,000만 건의 자료로 구성되어 있다. 이 자료는 1997년 9월 19일부터 1998년 4월 22일까지 약 7개월 동안 MovieLens의 사이트 (http://movielens.umn.edu)를 통해서 수집되었고, 각각의 사용자는 적어도 20개 이상의 영화에 응답을 했다.

본 논문에서는 943명 전체의 자료에서 사용자 200명을 선택하여 실험에 사용하였다. 선택한 자료의 80%는 추천시스템의 유사도와 제품의 선호도를 구하기 위해서 학습 셋으로 사용되었고, 20%의 자료는 추천시스템을 평가하기 위해서 테스트 셋 사용하였다. 자료는 1에서 5까지의 숫자로 평가가 되었는데 본 실험에서는 4와 5로 평가한 자료는 1로, 1부터 3까지로 평가한 자료는 0으로 바꾸었다. 변화된 자료에서 영화를 상품으로 생각하면 1은 상품의 구입을 의미하고 0은 구매하지 않은 것을 말한다.

실험은 제안된 추천시스템의 성능을 평가하기 위해서 속성추출 방법을 통해서 얻어진 중요속성의 수를 단계적으로 증가시키면서 성능을 평가했고, 추천시스템의 속도를 계산하기 위해서 기존의 추천시스템과 개선된 추천시스템 간의 연산횟수를 계산하여 추천의 속도를 비교했다.

4.2 평가 방법

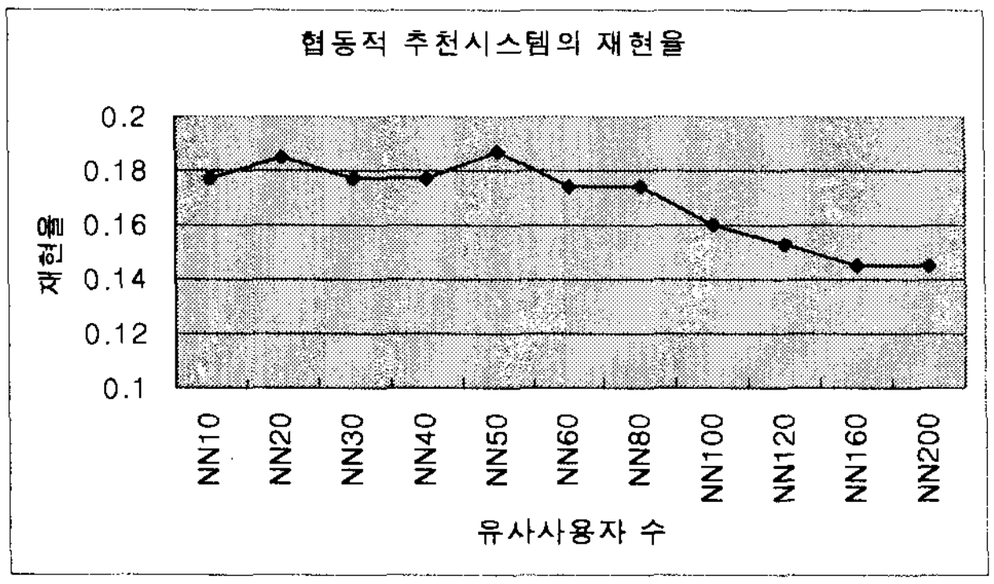
추천의 정확성을 평가하기 위해서 재현율(recall)을 사용하여 평가했다[8]. 여기서 추천의 수를 선호도가 높은 10개를 선택하는 Top-10추천을 했고, 10개의 추천 목록에 대해서 재현율을 구하여 비교를 했다.

재현율을 식(5)로 구하여 진다.

$$\text{재현율(recall)} = \frac{\text{size of hit set}}{\text{size of test set}} \qquad \text{식(6)}$$

4.3 실험 결과

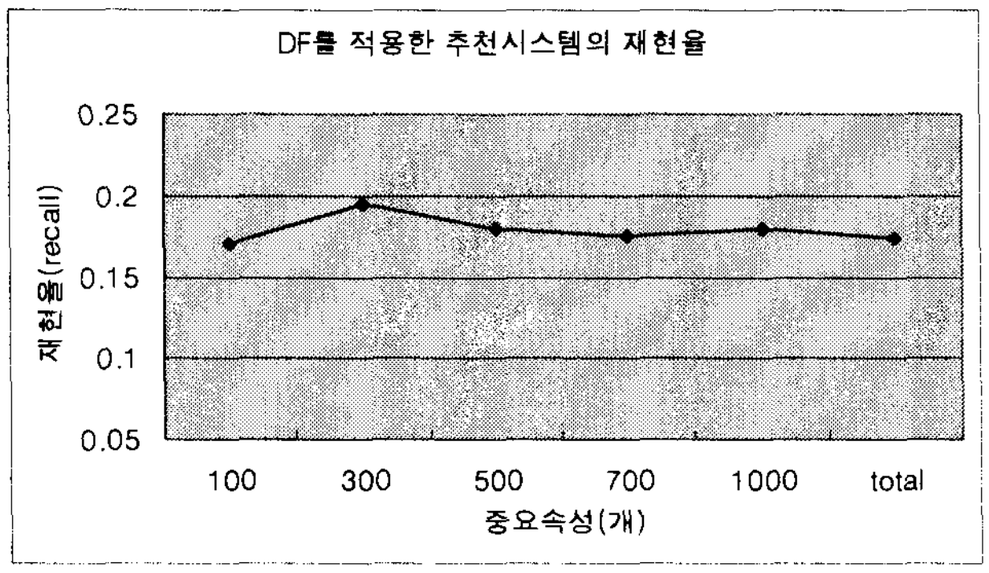
기존의 추천시스템에 유사사용자를 상위의 10명, 20명, 30명, 40명으로 단계별로 증가 시켜 모든 유사사용자를 모두 사용할 때까지 증가시키면서 재현율을 구한 결과는 [그림3]과 같다.



[그림3] 협동적 추천시스템의 재현율

[그림3]에서 유사사용자를 증가시킬수록 재현율은 증가하다 감소하는 것을 알 수 있고, 유사사용자를 50명으로 했을 때 재현율이 가장 좋은 것을 알 수 있다.

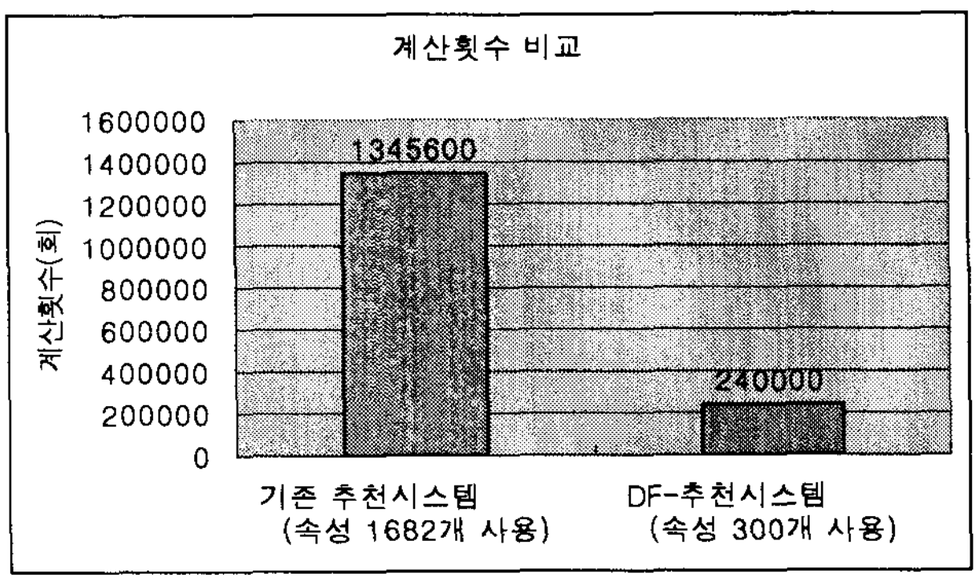
[그림4]는 속성추출을 이용해서 단계적으로 중요속성의 수를 증가시켰을 때의 결과를 보여주고 있다. 속성의 수는 100개, 300개, 500개로 단계적으로 증가시켰고, 전체의 속성을 사용한 경우는 기존의 추천시스템이다. 유사사용자는 50명을 선택하여 상품의 선호도를 구했다.



[그림4] DF를 적용한 추천시스템의 재현율

[그림4]에서 속성추출을 적용한 경우의 재현율은 기존의 추천방법에 비하여 재현율이 조금 향상된 것을 알 수 있다. 엔트로피를 적용한 경우의 결과도 [그림4]와 매우 유사하였다.

[그림5]는 기존의 추천시스템과 개선된 시스템 간의 속도를 비교하기 위해서 유사도 계산단계의 연산 횟수를 비교한 결과를 보여주고 있다.



[그림5] 협동적 추천시스템과 DF를 적용한 추천시스템의 연산횟수

[그림5]의 결과를 보면 계산횟수가 현저히 줄어든 것을 알 수 있다. 이 결과로 추천시스템의 속도가 빨라진 것을 알 수 있다. 연산횟수는 약 5배정도 줄어들었다.

5. 결론 및 향후 연구과제

본 논문은 기존의 추천시스템이 가지는 확장성의 문제점을 해결하고자 시도가 되었다. 문제점을 해결하기 위해서 유사도를 계산하기 전에 속성추출방법을 이용하여 중요속성을 선택하고 선택한 속성을 유사도 계산 단계에 적용하여 개선된 추천시스템을 제안하게 되었다. 개선된 추천시스템에서는 유사도를 계산하는 단계에서 많은 시간을 단축할 수 있었다. 그리고, 성능에서도 모든 속성을 전부 사용했을 경우와 유사하거나 좋은 결과를 보였다. 속성추출방법으로 사용된 단어 빈도수와 엔트로피를 적용한 추천시스템은 모두 기존의 추천시스템에 비하여 성능이 향상되었다. 기존의 협동적 추천시스템의 경우에 유사사용자의 수를 50으로 했을 때, 재현율이 가장 좋았다. 속성추출방법을 적용한 개선된 추천시스템의 경우는 중요속성을 300개 선택한 경우에 재현율이 가장 좋았다.

본 논문의 실험 결과를 실제 추천시스템의 구축에 접목을 시킨다면 기존의 추천시스템에서 추천

을 위해서 사용한 시간과 비용을 감소시킬 수 있다. 또한 사용자의 갑작스런 증가와 상품의 증가로 인한 확장성의 문제를 고민하고 확장성 문제를 해결하고자 사용될 시간과 비용 또한 줄일 수 있을 것이다. 즉 본 논문에서 제시한 방법을 사용함으로 추천시스템의 성능을 향상시키고 안정적인 상태를 유지할 수 있다.

본 논문은 추천시스템의 유사도를 계산하는 단계의 계산횟수를 줄이는 방법을 통해서 성능향상을 시도했지만, 앞으로 선호도 계산단계를 개선한다면 더욱 좋은 성능의 추천시스템을 구현 할 수 있을 것이다. 그리고, 속성추출의 여러 가지 다른 방법을 이용한 실험이 필요하다.

[참고문헌]

[1] 김중섭, 연관 규칙과 협동적 추천의 통합에 의한 추천 시스템의 성능 향상, 동국대학교 대학원, 2000.

[2] 이은령,이황규,허계범, 성공하는 E-Business를 위한 전자상거래 시스템 구축과 응용, 이한출판사, 2001.

[3] Balabanovi, M., Shoham, Y., "Content-Based Collaborative Recommendation", Communications of the ACM , Vol.40, 1997.

[4] Daniel Billsus, Michael J. Pazzani, "Learning Collaborative Information Filters", In Proceedings of Recommender Systems Workshop. Tech. Report WS-98-08, AAAI Press 1998.

[5] J. Ben Schafer, Joseph Konstant, John Reidl, "Recommender Systems in E-Commerce", In Proceedings of ACM E-Commerce 1999 conference, 1999.

[6] J. L. Herlocker, J. A. Konstan, A. Borchers, John. Riedl, "An Algorithmic Framework for Performing Collaborative Filtering", Proceedings of the Conference on Research and Development in Information Retrieval, 1999.

[7] John S. Breese, David Heckerman, Carl Kadie, "Empirical Analysis of Predictive for collaborative Filtering", Proceedings of the 14th Conference of Uncertainty in Artificial Intelligence, 1998.

[8] MovieLens Dataset, http://www.grouplens.org/data/

[9] P. Resnick, N. Iacovou, M. Suchak, P. Pergstom and J, Riedl, "GroupLens: An Open Architecture for Collaborative Filtering of Netnews", Proceedings of the ACM 1994 Conference on Computer Supported Cooperative Work, 1994.

[10] Tom M. Mitchell, "MACHINE LEARNING", The McGraw-Hill Company, 1997.

[11] Y. Yang & J. Pedersen, "A comparative study on feature selection in text categorization", ICML, 1997.