향상된 예측을 위한 대표 속성을 이용한 협력적 여과 방법

류영석, 양성봉
연세대학교 컴퓨터과학과

Collaborative Filtering Method Using the Representative Attribute for Better Prediction Quality

Young-Suk Ryu, Sung-Bong Yang
Dept. of Computer Science, Yonsei Univ.

요 약

사회의 복잡화와 인터넷의 성장으로 인하여 매일 급속도로 증가하고 있는 정보들을 사용자가 모두 검토해 보고 자신의 기호에 맞는 정보들만 선택하여 사용하기는 어려운 일이다. 이를 보완하기 위해 자동화된 정보 여과 기술이 사용되는데 대표적인 방법으로 내용 기반 여과(Information Filtering) 기술과 협력적 여과(Collaborative Filtering) 기술이 있다. 이 중 협력적 여과 기술은 정보의 속성을 고려하지 않는다는 단점을 가지는데 본 논문에서는 이를 보완하여 정보의 대표 속성을 중심으로 선호도 예측을 수행하는 개선된 협력적 여과 방법을 제안한다. 그리고 기존 협력적 여과 기술과 예측의 정확성에 대하여 성능 비교 실험을 수행함으로써 제안한 방법의 타당성을 제시한다.

1. 서론

자동화된 정보 여과 기술은 많은 정보들 중에서 사용자에게 필요한 정보만을 추출하여 제공함으로써 사용자의 수고를 덜어 줄 수 있다. 현재까지 제안된 정보 여과 기술은 크게 내용 기반 여과 기술과 협력적 여과 기술로 나눌 수 있다.

내용 기반 여과는 기존의 내용 추출(Information Retrieval) 개념을 사용하는 방식으로, 정보의 다양한 속성(attribute)에 대한 사용자의 선호도들의 집합(profile)을 구성하여 정보의 내용과 사용자의 선호도를 비교함으로써 기호에 맞는 정보만을 여과해 낼 수 있다. 속성에 대한 사용자의 선호도는 사용자가 명시적으로 제시하거나 혹은 사용자의 행동을 통해 시스템이 내부적으로 반영시킬 수 있다. 이러한 내용 기반 여과는 특정한 단어나 속성값을 가지는 정보를 여과하기에 매우 효과적이다. 그러나 여과된 정보들의 질(quality)을 판별하기에는 적절하지 못하다[1]. 또한 사용자 프로파일에는 없는 속성항목을 가지는 정보에 대한 여과는 수행하기 어렵다.

협력적 여과는 정보 자체에 대한 사용자의 선호도들의 집합을 구성한 후 다른 사용자들의 선호도와 비교하게 되며 이를 통해 정보를 여과하게 된다. 즉 동일한 정보에 대한 선호도의 상관관계를 통하여 현 사용자와 기존 사용자들 사이에 기호의 유사성의 정도를 결정하게 되고, 기존 사용자들과의 유사성과 그들의 선호도에 기반하여 높은 선호도가 예상되는 정보만 현 사용자에게 여과하여 제공하게 되는 것이다.

이러한 협력적 여과는 자동화된 프로세스로는 쉽게 분석될 수 없는 정보의 질을 기존 사용자들의 선호도를 통해 어느 정도 반영한다는 장점이 있으나 정보의 속성에 대한 사용자의 선호도는 고려하지 않는다는 문제점을 가지고 있다.

본 논문에서는 정보의 대표 속성에 대한 사용자의 선호도를 협력적 여과에 반영함으로써 예측의 정확성을 향상시키는 개선된 협력적 여과 기술을 제안하고 그 성능을 평가한다.

본 논문의 구성은 다음과 같다. 2장에서는 협력적 여과 기술에 대한 관련 연구를 분석하고 3장에서는 본 논문에서 제시하는 개선된 협력적 여과 방법을 설명한다. 4장에서는 성능 평가를 위한 실험 환경 및 실험 결과 분석을 기술한다. 끝으로 5장에서는 결론을 맺는다.

2. 관련 연구

2.1. GroupLens

초기에 유즈넷 뉴스를 대상으로 개발되었으며 식 1의 Pearson 상관관계 계수를 통하여 사용자 a와 사용자 k 사이의 유사성의 정도를 결정하고 이렇게 계산된 유사도를 이용하여 식 2와 같이 사용자 a의 정보 i에 대한 선호도의 예측을 하게 된다[2][3]. 여기서 j는 사용자 a와 사용자 k가 모두 선호도를 매긴 정보를 의미하고 $r_{x,y}$는 사용자 x의 정보 y에 대한 선호도를 나타내며 $\overline{r_x}$는 사용자 x의 전체 정보에 대한 평균 선호도이다.

$$w_{a,k}=\frac{\sum_j (r_{a,j}-\overline{r_a})(r_{k,j}-\overline{r_k})}{\sqrt{\sum_j (r_{a,j}-\overline{r_a})^2 * \sum_j (r_{k,j}-\overline{r_k})^2}} \quad (1)$$

$$p_{a,i} = \overline{r_a} + \frac{\sum_k w_{a,k} \times (r_{k,i} - \overline{r_k})}{\sum_k w_{a,k}} \quad (2)$$

2.2. Ringo

Ringo[4]는 음반과 서적등을 추천하기 위하여 개발되었으며 Pearson 알고리즘을 사용하여 현 사용자와 기존 사용자들간에 유사도를 결정한 후 유사도가 일정 임계치를 넘는 기존 사용자들만 현 사용자의 이웃(neighborhood)으로 인정하고 이러한 이웃들만 식 2에 적용하여 선호도 예측을 한다. 또한 식 3과 같이 Pearson 알고리즘에서 각 사용자의 전체 정보에 대한 평균 선호도 대신 선호도등급을 7단계로 구분하고 그 중간 값인 4를 일괄적으로 적용하는 Constrained Pearson 알고리즘을 제안하였다.

$$w_{a,k} = \frac{\sum_j (r_{a,j} - 4)(r_{k,j} - 4)}{\sqrt{\sum_j (r_{a,j} - 4)^2 * \sum_j (r_{k,j} - 4)^2}} \quad (3)$$

2.3. best n-neighbors

이 방법은 Pearson 알고리즘으로 현 사용자와 기존 사용자들 사이에 유사도를 결정한 후 유사도가 가장 큰 n명의 사용자들만 현 사용자의 이웃으로 인정하고 식 2를 통하여 선호도 예측을 한다[5]. 특히 기존 사용자들을 모두 이웃으로 인정하는 경우는 Pearson All but 1으로 지칭한다[6].

2.4. Filterbot

유즈넷 뉴스에 대하여 정보의 특정 속성 값에 대하여 항상 정해진 선호도를 보이는 가상의 유저들을 만들어 이들을 기존 사용자 그룹에 포함시켜 Pearson 알고리즘을 수행한다. 이러한 가상의 유저로는 SpellCheckerBot, IncludedMsgBot, LengthBot이 있으며 이들은 각각 맞춤법, 첨부메시지 양, 기사의 길이에 따라 미리 정해진 선호도를 보인다[1].

3. 제안하는 협력적 여과 방법

본 논문에서는 정보의 속성을 고려하지 않는 협력적 여과의 단점을 보완하여 좀더 효과적인 여과를 수행하기 위해 대표 속성(Representative Attribute)을 중심으로 선호도 예측을 수행하는 방법들을 제안한다. 대표 속성이란 정보의 선호도에 가장 크게 영향을 미치는 속성을 의미한다. 대표 속성 값은 대표 속성이 가질 수 있는 값을 지칭한다.

3.1. 대표 속성 값들에 대한 평균 선호도를 이용

식 1과 식 2를 살펴보면 기존의 Pearson 알고리즘은 사용자의 각 정보에 대한 선호도의 정도를 반영하여 예측하기 위해 해당 사용자의 전체 정보들에 대한 평균 선호도를 감하거나 더한다. 그러나 전체 정보들의 평균 선호도를 사용하는 것은, 정보의 대표 속성 값들에 대해 사용자가 차별적인 선호도를 가지는 경우 이를 제대로 반영하지 못하는 단점을 가진다. 그러므로 본 논문에서는 이를 보완하기 위해 각 대표 속성 값에 대한 평균 선호도를 Pearson 알고리즘에 이용하는 방법을 제안하였고 이를 Pearson UMRAV(Using the Mean of Representative Attribute value)로 명하였다. 결과적으로 기존 Pearson 알고리즘의 식들은 아래의 식 4와 식 5와 같이 변형되게 된다. 여기서 $\overline{r_{x,y}'}$는 정보 y와 같은 대표 속성 값을 가지는 정보들에 대한 사용자 x의 평균 선호도를 의미한다.

$$w'_{a,k} = \frac{\sum_j (r_{a,j} - \overline{r_{a,j}'})(r_{k,j} - \overline{r_{k,j}'})}{\sqrt{\sum_j (r_{a,j} - \overline{r_{a,j}'})^2 * \sum_j (r_{k,j} - \overline{r_{k,j}'})^2}} \quad (4)$$

$$p'_{a,i} = \overline{r_{a,i}'} + \frac{\sum_k w'_{a,k} \times (r_{k,i} - \overline{r_{k,i}'})}{\sum_k w'_{a,k}} \quad (5)$$

식 4에서 만약 사용자 a와 사용자 k가 j와 동일한 대표 속성 값을 가지는 정보들에 대해 동시에 임계치 α회 이상 선호도 입력을 하지 않았을 경우는 대표 속성 값에 대한 선호도 평균 대신에 전체 정보들에 대한 선호도 평균을 이용하게 된다. 이는 근거 데이터의 수가 적을 경우 부정확한 예측이 나오는 것을 방지하기 위해서이다.

식 5에서 사용자 a와 사용자 k가 각각 임계치 α회 이상 선호도 입력을 하지 않았을 경우는 해당 사용자의 대표 속성 값에 대한 선호도 평균 대신에 전체 정보들에 대한 선호도 평균을 이용하게 된다.

3.2. 성별·연령별 그룹 이용

일반적으로 성별, 연령별로 사고 방식과 선호도의 차이가 있는 점에 착안하여 서로 동일한 성별과 유사한 연령을 가지는 사용자들은 공통적인 선호도를 가진다고 가정하였다.

협력적 여과에 이러한 성별, 연령별 선호도의 차를 반영하기 위하여 기존 사용자를 이용해 성별, 나이별로 그룹을 만들어 그룹별로 정보의 대표 속성 값들에 대한 선호도의 평균을 프로파일로 가지는 가상의 사용자들을 생성한다. 다음으로 현 사용자와 그가 속하는 그룹간에, 대표 속성의 각 값들에 대한 선호도를 기준으로 상관관계를 계산하여 임계치 β 이상이면 협력적 여과에 포함시킨다.

이러한 과정을 통하여 현 사용자의 선호도가 자신이 속한 성별과 나이의 경향과 일정 수준 이상 유사하다면 그러한 성별과 나이를 가지는 그룹들의 선호도를 예측에 반영하게 된다.

4. 실험 및 성능 평가

4.1. 실험 환경

실험에는 EachMovie[7] 데이터를 사용하였다. 이 데이터에는 영화에 관한 사용자의 선호도가 0, 0.2, 0.4, 0.6, 0.8, 1.0 의 6개 단계의 수치로 표현되어 있다. 본 실험에서는 최소 100회 이상 선호도 입력을 한 사용자 4788명을 추출하여 이 가운데 1000명을 기존 사용자군으로 두고 나머지 사용자들 중에 무작위로 테스트 사용자 100명을 선택하여 총 1628개의 영화 중 임의의 5개의 영화에 대하여 선호도를 예측하고 실제 선호도와 비교하였다.

성별·연령별 특징을 이용하기 위하여 나눈 그룹과 해당 사용자 수는 표 1과 같다.

영화의 대표 속성은 장르로 가정하고 실험을 수행하였고 각 영화의 장르는 EachMovie 데이터에 구분되어 있는 것을 따랐다. 즉 장르는 액션, 애니메이션, 외국예술, 고전, 코미디, 드라마, 가족, 공포, 로맨스, 스릴러의 10가지로 구분된다.

그룹		사용자수(명)
성별 그룹	남	3831
	여	957
연령별 그룹	1-14세	115
	15-19세	551
	20-24세	1023
	25-29세	1055
	30-39세	1066
	40-49세	734
	50-59세	208
	60-99세	36

표 1. 성별 · 연령별 그룹과 사용자수

4.2. 평가 기준

예측의 정확성을 평가하기 위하여 MAE(Mean Absolute Error)를 사용하였다. 아래 식은 MAE를 나타낸 것이며 여기서 N은 총 예측 횟수, ε는 예측된 선호도와 실제 선호도간의 오차를 나타낸다.

$$|\overline{E}| = \frac{\sum_{i=0}^{N} |\varepsilon_i|}{N} \qquad (6)$$

4.3. 실험 결과 및 분석

실험을 통하여, 제안한 방법들과 기존의 Pearson 알고리즘을 사용한 방법간에 예측의 정확성에 대한 비교를 수행하였다. UMRAV에는 임계치 α=60이 사용되었고 성별 · 연령별 그룹 이용 시에는 임계치 β=0.5가 사용되었다.

실험 결과는 표 2와 같다. 표 2를 살펴보면 제안한 UMRAV가 기존의 Pearson Max50보다 예측의 오차가 적음을 알 수 있다. 또한 성별 · 연령별 그룹을 Pearson Max50에 기존 사용자군에 추가한 경우에도 기존 Pearson Max50 보다 정확한 예측을 수행하였음을 볼 수 있으며 UMRAV에 성별 · 연령별 그룹을 추가한 방법의 경우 가장 좋은 성능을 보임을 확인할 수 있다.

Method	MAE
Pearson All but 1	0.253087
Pearson Max50	0.235725
Pearson Max50 with 성별 · 연령별 그룹	0.235282
Pearson UMRAV Max50	0.234201
Pearson UMRAV Max50 with 성별 · 연령별 그룹	0.232534

표2. 실험 결과 - 예측의 정확성 비교

5. 결 론

본 논문에서는 정보의 속성을 고려하지 않는 협력적 여과를 개선하기 위해 대표 속성을 이용하여 협력적 여과를 수행하는 방법들, 즉 각 대표 속성 값에 대한 평균 선호도를 Pearson 알고리즘에 이용하는 UMRAV 방법과 성별 · 연령별 그룹을 기존 사용자군에 포함시키는 방법을 제안하였다.

제안한 여과 방법들에 대해 예측의 정확성을 확인하기 위하여 영화 정보에 대하여 장르를 대표 속성으로 두고 실험을 수행하였으며 그 결과 기존 여과 방법들에 비해 성능의 향상이 있음을 확인할 수 있었다.

참고 문헌

[1] Badrul M. Sarwar, Joseph A. Konstan, Al Borchers, Jon Herlocker, Brad Miller, and John Riedl, "Using filtering agents to improve prediction quality in the grouplens research collaborative filtering system," *Proceedings of 1998 Conference on Computer Supported Collaborative Work*, 1998.

[2] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, and John Riedl, "GroupLens: An open architecture for collaborative filtering of netnews," *Proceedings of ACM CSCW'94 Conference on Computer Supported Cooperative Work*, pp. 175-186, 1994.

[3] Joseph A. Konstan, Bradley N. Miller, David Maltz, Jonathan L. Herlocker, Lee R. Gordon, and John Riedl, "GroupLens: Applying collaborative filtering to Usenet news," *Communications of the ACM*, 40(3), pp. 63-65, 1997.

[4] Upendra Shardanand and Patti Maes, "Social information filtering: Algorithms for automating "word of mouth"," *Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems*, pp. 210-217, 1995.

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

[6] John S. Breese, David Heckerman, and Carl Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, pp. 43-52, 1998.

[7] P. McJones, EachMovie collaborative filtering data set, URL: http://www.research.digital.com/SRC/eachmovie/, 1997.