사후 필터링기법을 사용한 실시간 상황 인식 추천 시스템

최광훈, 유헌창
고려대학교 컴퓨터정보통신대학원
e-mail : hoon7773@korea.ac.kr

A Real-time Context Recognition Recommendation System Using Post-Filtering

Kwang-Hoon Choi, Heonchang Yu
Dept. of Computer and Information Technology, Korea University

요 약

추천 시스템은 다양한 분야에 적용되는 기술로서 활발한 연구가 진행되고 있고 기존 추천 시스템의 성능을 높이기 위해서 더욱 개인화된 차세대 추천 시스템의 필요성이 대두되고 있다. 본 논문은 하이퍼 개인화 범주에 속하는 사후 필터링기법을 사용한 실시간 상황 인식 추천 시스템을 제안한다. 실시간 상황 인식 추천 시스템은 사용자의 행동과 계속적인 동기화로 현재 상황에 가장 적합한 추천 목록을 생성하기 때문에 사용자 기반 협업 필터링(User-based Collaborative Filtering), 콘텐츠 기반 필터링(Content-based Filtering), 특이값 분해(Singular Value Decomposition) 보다 훨씬 미래 지향적인 추천 시스템이다.

1. 서론

최근에 영화나 도서 같은 콘텐츠 선택에 사용자들에게 도움이 되는 많은 추천 시스템이 제공되고 개발되고 있다. 추천 시스템은 머신 러닝 기술의 발전에 따라 상당한 진화를 해왔다.

1990 년대에 시작된 추천 시스템 연구는 정확성이 부족하고 조잡했다. 하지만 지금은 웹 기술의 발전으로 대량의 다양한 데이터가 축적되었고 이러한 데이터를 활용한 추천 시스템은 더욱 정확해 졌고 개인화된 맞춤형 서비스를 제공한다.

넷플릭스, 구글, 아마존, 네이버, 카카오 등의 기업들은 실제로 사용자의 행동이나 프로필로 개인 취향에 적합한 콘텐츠를 제공한다. 이러한 기업들의 추천 시스템 효과는 매우 크다. 넷플릭스의 사용자 80%가 추천으로 콘텐츠를 감상하고 아마존은 35%가 추천으로 구매한다고 밝혔다. 네이버도 추천 시스템으로 뉴스는 17%, 동영상은 18% 소비량이 상승되는 성과가 있었다고 한다.

본 논문에서는 기존의 일률적인 개인화 추천 방식을 좀 더 세분화하여 발전시킨 사용자가 사용하는 상황에서의 추천 기법을 제안한다.

제안한 추천 기법은 사용자 기반 협업필터링 기법으로 예측 값을 구한 후 시간대 별 장르 선호도를 분석해서 사용자의 시간적인 상황에 적합한 콘텐츠를 추천함으로써 추천의 다양성과 정확성을 높일 수 있다.

2. 관련 연구

2.1 협업 필터링 추천 시스템

추천 시스템의 가장 기본적인 기법으로 사용자나 아이템의 유사성을 그룹화 한 후 필터링하는 방법이다. 협업 필터링 시스템은 크게 두 가지 종류가 있다. 사용자 기반 협업 필터링은 과거에 비슷한 취향을 가진 사용자들이 미래에도 관심사가 동일할 것이라는 가정으로 추천을 한다. 아이템 기반 협업 필터링은 아이템들의 비슷한 점을 찾고 아이템의 유사도를 기반으로 사용자에게 추천해 주는 방식이다[1].

협업 필터링은 장점과 단점이 존재한다. 장점은 구현이 간단한 것에 비해 매우 정확하다. 단점은 콜드 스타트 문제가 있는데 사용자의 활동 데이터가 희박한 경우 적절한 추천이 어렵다는 점이다.

2.2 콘텐츠 기반 추천 시스템

콘텐츠 기반 추천 시스템은 협업 필터링과 다르게 사용자나 아이템 선호도를 사용하는게 아니라 사용자 프로필이나 아이템 속성 정보의 사용자 선호도를

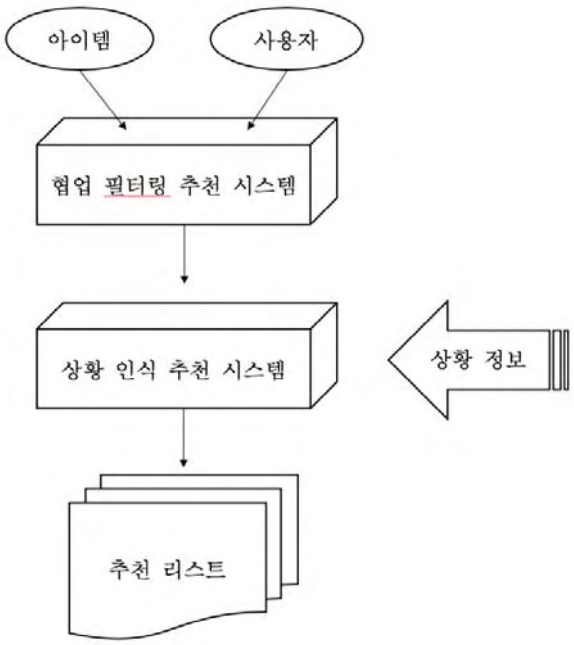
이용한다[2]. 예를 들면 유투브에서 호나우두가 나오는 동영상을 시청했다면 추천 동영상리스트에 축구나 호나우두가 노출된다. 이러한 콘텐츠 기반 추천 방식은 아이템의 특징을 찾아내는게 중요하기 때문에 아이템을 분석해서 속성을 찾아내는 과정을 필요로 한데 이러한 정보는 찾기 어려운 부분이 단점이기도 하다. 하지만 장점으로는 콜드 스타트 문제 해결이 가능하다. 사용자의 활동 데이터가 희박해도 아이템 속성을 이용한 적절한 추천이 가능하다.

2.3 특이값 분해(Singular Value Decomposition)

특이값 분해(Singular Value Decomposition)는 차원 축소 기법 추천 시스템 중 하나로 m × n 행렬을 직각행렬 2 개와 대각행렬 1 개로 분해한 후 차원축소로 예측 값을 생성한다. SVD 는 고차원 행렬을 저차원으로 축소시키는 기법으로 계산 속도가 빠르고 정확도가 높다[5]. 사용자의 데이터 증가로 평가 데이터 희소성이 늘어나고 성능이 감소되는 문제가 생기는데 SVD 는 이러한 문제를 효과적으로 처리하기 위한 기법이다. 평가나 관계 정보가 확실할 경우, 행렬 분해를 통해 잠재 요인(latent factor)을 잘 분류할 수 있다.

3. 사후 필터링 기반 실시간 상황 인식 추천 시스템

본 장에서는 기존 추천 시스템의 일률적인 개인화 문제점 해결을 위해 좀 더 정확하고 다양한 추천을 할 수 있는 기법인 사후 필터링을 사용한 실시간 상황 인식 추천 시스템을 설명한다[4].



(그림 1) 상황 인식 추천 시스템 구조

사후 필터링기법을 이용한 실시간 상황 인식 추천 시스템 구조는 (그림 1)과 같다. 아이템과 사용자 유사성을 사용한 협업 필터링 추천시스템으로 예측 값을 구한 후 상황 정보를 이용한 상황 인식 추천시스템으로 최종 예측 값을 구한다.

3.1 사용자 기반 협업 필터링기법으로 평점 예측

사용자 기반 협업 필터링에 많이 사용하는 피어슨 상관계수를 사용한 유사도 계산 방법을 사용한다.

$$sim(a,b) = \frac{\sum_{p\in P}(r_{a,p}-\bar{r}_a)(r_{b,p}-\bar{r}_b)}{\sqrt{\sum_{p\in P}(r_{a,p}-\bar{r}_a)^2}\sqrt{\sum_{p\in P}(r_{b,p}-\bar{r}_b)^2}} \quad (1)$$

피어슨상관계수인 식 (1)는 2 개의 변수를 비교해서 선형적인 관계를 추론하는 방법으로 사용자간의 유사도를 계산할 수 있다.

$$pred(a,p) = \bar{r}_a + \frac{\sum_{b\in N} sim(a,b)*(r_{b,p}-\bar{r}_b)}{\sum_{b\in N} sim(a,b)} \quad (2)$$

식 (2)는 다른 사용자와 유사도를 비교해서 평점을 사용한 가중치를 적용하는 방법으로 최종적으로 <표 1>과 같이 평점을 예측한다.

<표 1> 아이템 평점 예측값 생성

	1	2	3	4	5	…	88	89	90	91	92
1				3.1		…				3	
2		4.1				…		5			4
3				3		…	5				
4	1					…		2.7			
5		4		3.3		…				4	
6			4.3			…				5	

∫

69	4				3	…	4.5	5		3.9	
70			4			…					
71		5		4.2		…	4				
73				3		…					3
74	1					…		1.5			

3.2 실시간 상황 인식 프로필 생성

실시간 상황 인식 프로필을 생성하는 계산방법은 식 (3)과 같다. $g_{t,u}$는 사용자가 시청한 시간대의 특정 장르의 합이다. $min_{g_{t,u}}$와 $max_{g_{t,u}}$ 는 사용자가 시청한 시간대의 특정 장르 합의 최소값과 최대값이다.

$$gs_{t,u} = \frac{g_{t,u} - min_{g_{t,u}}}{max_{g_{t,u}} - min_{g_{t,u}}} \qquad (3)$$

<표 2>은 식 (3)으로 계산된 프로필이다. 사용자의 시간대를 4 구간으로 나눠서 6 ~ 12 시는 오전, 12 ~ 18 시는 오후, 18 ~ 24 시는 저녁, 0 ~ 6 시는 심야로 구분했다.

<표 2> 실시간 상황 인식 프로필

User ID	Time	장르 선호도						
		Action	Adventure	Animation	Children	Comedy	Drama	…
112	오후	0.00	0.00	0.00	0.00	0.75	1.00	…
	저녁	0.00	0.00	0.00	0.00	0.50	1.00	…
450	오전	0.25	0.50	0.00	0.25	1.00	0.50	…
	오후	0.00	0.00	0.00	0.00	0.00	0.50	…
95	오전	0.00	0.00	0.66	1.00	0.33	0.33	…
	오후	0.25	0.00	0.25	0.25	0.50	0.00	…
376	오전	0.50	0.00	0.00	0.00	1.00	0.50	…
	오후	0.00	0.00	0.00	0.00	0.00	0.00	…
671	오전	1.00	1.00	0.00	0.00	1.00	0.00	…
	오후	1.00	0.57	0.00	0.00	0.00	0.28	…
	저녁	0.50	0.00	0.00	0.00	0.50	0.00	…
	심야	0.00	0.00	0.00	0.00	0.00	0.00	…

3.3 실시간 상황 인식 예측 값 생성

이 단계는 사용자가 실시간으로 접속하는 시간대에 상황 프로필 정보로 최종적인 예측 값을 계산한다.

$$R_{u,i} = \begin{cases} 1, & r_{u,i} < 3 \\ 0, & otherwise \end{cases} \qquad (4)$$

$$P_{u,i} = \sum_{l=1}^{n} (G_{i,l} \times C_{u,l}) \times R_{u,i} \qquad (5)$$

식 (4)에서 $r_{u,i}$는 콘텐츠 i 를 사용자 u 가 협업필터링 추천 시스템으로 예측한 평점이다. 이 평점을 이진 클래스로 변환한다. 식 (5)에서 $G_{i,l}$는 콘텐츠 i 의 장르 값 이고 $C_{u,l}$는 사용자 u 의 상황 인식 프로필 장르 선호도 값이다. 계산된 예측 값이 높을 수록 실시간 상황에서 사용자가 선호하는 콘텐츠이다.

3.4 실시간 상황 인식 추천 목록 생성

최종적으로 계산된 실시간 상황 인식 예측 값을 이용해서 Top-N 기법으로 나열하면 추천 목록이 생성된다. <표 3>은 실시간 상황 인식 추천 목록의 일부이다. 사용자가 사용하는 시간 대에 따라서 추천 목록이 바뀌는 것을 확인할 수 있다.

<표 3> 실시간 상황 인식 추천 목록의 일부

User ID	Time	추천 목록의 Movie ID							
112	오후	921	948	1589	3	79	119	156	…
	저녁	921	1589	3	79	119	156	234	…
450	오전	1196	765	766	1001	115	131	474	…
	오후	44	138	153	173	229	311	320	…
95	오전	1176	1628	1676	36	104	1038	1054	…
	오후	391	874	948	1523	31	156	548	…
376	오전	1523	1623	807	1305	1576	391	762	…
	오후	324	1184	587	920	1523	79	234	…
671	오전	778	14	114	126	713	765	766	…
	오후	765	18	21	120	191	202	352	…
	저녁	762	157	777	807	877	915	1312	…
	심야	14	61	65	177	223	273	415	…

4. 실험 및 평가

4.1 실험데이터

본 논문의 실험에 사용한 MovieLens 데이터는 미네소타 대학 GroupLens Research Project 에서 조사한 영화 정보와 평점이 포함된 데이터이다. 총 1,682 편의 영화와 942 명의 사용자 평가 정보로 구성되어 있다. 데이터를 수집한 기간은 1997 년 9 월 부터 1998 년 4 월 까지 9 개월 동안이다. MovieLens 평가 데이터의 구성은 사용자 ID 와 영화 ID, 사용자들이 평가한 1 점에서 5 점 범위의 평점, 평가한 시간이다. 사용자들은 10 만개의 평가치를 작성하였고 사용자 × 영화 매트릭스의 총 칸은 1,586,126 개로 6.3% 평점이 작성되었다. 따라서 MovieLens 데이터의 희박성은 93.7% 이다. 또한 MovieLens 데이터는 영화의 장르 데이터를 포함하고 있고 영화 데이터베이스 IMDB(Internet Movie Database) 기준에 충족된 18 개의 장르로 분류되어 있다. 영화는 1 개에서 5 개의 장르에 포함되어 있다. <표 4>는 장르별 영화 편수이다.

<표 4> 장르별 영화 편수

장르	편수	장르	편수
Unknown	2	Film-Noir	24
Action	251	Horror	92
Adventure	135	Musical	56
Animation	42	Mystery	61
Children	122	Romance	247
Comedy	505	Sci-Fi	101
Crime	109	Thriller	251
Documentary	50	War	71
Drama	725	Westerm	27
Fantasy	22		

4.2 평가 방법

추천시스템은 사용자가 얼마나 만족했는지 유효성 측정이 필요하다. 추천 목록의 유효성 평가는 분류 모델 중 이진클래스로 다룰 수 있다. 본 논문에서는 Top-N 기법을 평가하기 위한 방법으로 분류 모델의 유명한 평가방법인 정확도(precision)와 재현률(recall)을 사용한다.

정확도(precision)는 예측한 추천 목록들 중 결과가 참인 경우의 비율로 식 (6)과 같다. 참 긍정(true positive)은 결과가 참이고 예측도 참인 모든 경우의 수, 거짓 긍정(false positive)은 결과가 거짓이고 예측은 참인 모든 경우의 수를 의미한다.

$$precision = \frac{true\ positive}{true\ positive + false\ positive} \quad (6)$$

재현률(recall)은 결과가 참인 목록 중 추천 목록에 포함되는 비율로 식 (7)과 같다. 거짓 부정(false negative)은 결과가 참이지만 예측이 거짓인 모든 경우의 수를 의미한다.

$$recall = \frac{true\ positive}{true\ positive + false\ negative} \quad (7)$$

4.3 실험과 분석

본 논문은 성능을 검증하기 위한 방법으로 기존에 많이 사용하는 추천 기법 중 세 가지와 비교해 보았다. 기존의 세 가지 추천 기법은 사용자 기반 협업필터링(User-based collaborative Filtering) 과 콘텐츠 기반 협업필터링(Content-based collaborative Filtering) 과 특이값 분해(Singular Value Decomposition) 이다.

실험에 사용한 데이터를 80:20 으로 분류하고 80 은 학습에 사용하는 데이터이고 20 은 테스트에 사용된 데이터이다. 실험 방법은 평점 점수가 가장 높은 순서로 상위 30 개의 추천 리스트를 사용자의 실시간 평점을 적용한 테스트 데이터와 비교해서 분석하였다.

<표 5> 기존 추천 기법과 비교한 성능 측정

추천 기법	Precision	Recall
사용자 기반 협업 필터링	0.02663139	0.03036318
콘텐츠 기반 협업 필터링	0.02328042	0.02569977
특이값 분해 (SVD)	0.02304965	0.02669019
제안 방법	0.03862434	0.04174868

추천의 정확도와 재현률에 대한 두 가지 성능 측정이 이루어졌고 <표 5>와 같은 결과를 얻었다. 측정 결과는 기존의 세가지 추천 기법에 비해 제안된 기법이 정확도는 약 44 ~ 67%, 재현률은 37 ~ 44%의 성능 향상이 있다는 것을 확인할 수 있었다.

5. 결론

사용자들은 수 많은 콘텐츠 중에서 자신의 취향에 맞는 것을 찾기 위해 많은 노력을 하고 있다. 추천 시스템은 이러한 사용자들에게 원하는 콘텐츠를 빠르고 다양하게 제공한다. 기업들은 사용자 개개인의 취향을 정확하게 파악할 수록 사용자가 늘어날 것이고 소비량도 늘어날 것이다. 널리 사용하는 협업필터링 기법이나 SVD 기법으로는 사용자의 상황이 없는 개인의 일률적인 추천만 할 수 있었다. 사용자의 상황은 계속 변하고 있고 이러한 상황에 맞는 추천을 해준다면 추천을 좀 더 세분화하고 다양화할 수 있다.

본 논문은 사후 필터링기법을 사용한 실시간 상황인식 추천 시스템을 기존 추천 시스템과 비교해 보았다. 실험 평가를 통해서 기존 기법에 비해 정확도는 44 ~ 67% , 재현률은 37 ~ 44%의 높은 성능 향상을 확인했다.

향후 연구과제로는 사용자의 다양한 상황(장소, 계절, 동행인)을 활용한 추천 시스템의 정확도 향상 연구를 필요로 한다.

참고문헌

[1] J. S. Breese, D. Heckerman, and C. Kadie, “Empirical analysis of predictive algorithms for collaborative filtering”, in *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pp. 43-52, 1998.

[2] R. J. Mooney and L. Roy, “Content-based book recommendation using learning for text categorization”, in *Proceedings of the ACM Conference on Digital Libraries*, pp. 195-204, 2000.

[3] Jae Sik Lee, Seog Du Park, “Performance Improvement of a Movie Recommendation System using Genre-wise Collaborative Filtering”, in *Korea Intelligent Information Systems Society*, pp. 65-78, 2007.

[4] Suresh Kumar Gorakala, “Building Recommendation Engines”, *Packt Publishing*, 169~207, 2016.

[5] Y. Koren, R. Bell, and C. Volinssky, “Matrix factorization technique for recommender filtering”, *IEEE Computer*, Vol. 42, No. 8, pp. 30-37, 2009.