R 에서 협업 필터링 을 이용한 드라마 추천 시스템

임성훈* , 백수빈* , 박두순*
*순천향대학교 컴퓨터소프트웨어공학과
e-mail : sasumpi123@naver.com

A drama recommendation system using collaborative filtering in R

Sung-Hun Lim*, Baek soo bin*, Doo-Soon Park*
*Dept. of Computer Software Engineering, SoonchungHyang University

요 약

종편 tv채널 증가와 모바일, 웹 시장의 발달로 엄청난 많은 양의 드라마가 양산되고 있다. 또한, 현대인들은 복잡한 현대의 생활 구조 때문에 드라마를 제 시간에 시청하거나 자신의 취향에 맞는 드라마를 골라서 시청하는 것은 매우 어렵다. 따라서, 본 논문에서는 R에서 협업 필터링 방법을 이용하여 사용자가 자신에게 가장 적합한 드라마를 추천하는 시스템을 제안한다.

1. 서론

최근 인터넷과 모바일 기기의 대중화에 더해 통신기술의 발달로 거대한 동영상 콘텐츠 시장이 형성되었다. 또한 시간과 장소에 구애 받지 않고 원하는 영상을 모바일 기기를 통해 볼 수 있다는 점이 사람들이 드라마를 시청하는 것에 큰 영향을 끼쳤다. 또한 tv드라마 뿐 만 아니라 현재 형식이나 내용에 제약이 없고 광고효과가 뚜렷하다는 장점을 지니고 있는 웹 드라마도 선풍적인 인기를 끌면서 드라마 종류는 급증하고 있다.

이와 같이 급증하는 드라마들로 인해 사용자들은 자신이 정말 원하는 드라마들을 찾기 힘들어졌다. 그래서 개인에 맞는 드라마를 추천해주는 추천시스템들이 등장하였다. 이러한 추천 시스템 중에서는 협업 필터링 방법이 가장 많이 사용하는 방법이다. 협업 필터링 방법은 초기에 사용자에 대한 특성을 파악할 수 없어서 추천에 대한 신뢰도가 떨어지는 Cold Start 문제가 발생한다.

본 논문에서는 Cold Start 문제를 개선하기 위하여 사용자들의 개인 성향들인 장르, 성별, 연령, 직업, 선호tv채널을 이용하여 각 개인에 맞는 드라마를 추천하는 시스템을 제안한다.

*corresponding author : doo-soon park
-이 논문은 한국연구재단의 지원을 받아 수행되었음 (NO.nrf-2017R1A2B1008421)

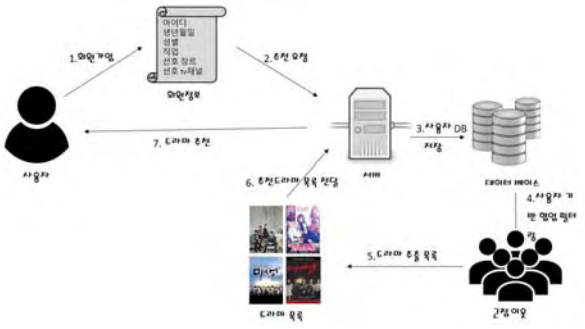
2. 드라마 추천 시스템의 구성

협업필터링(Collaborative Filtering)은 다수의 사용자들에게 얻은 기호정보(Taste Information)에 따라 사용자들의 관심사들을 예측하게 해주는 방법이다. 협업 필터링의 주요 기술로는 사용자 기반 방식과 아이템 기반 방식이 있으나, 본 논문에서는 사용자 기반 방식을 사용한다. 협업필터링을 활용하여 사용자간 유사도를 측정하였고 이를 이용하여 근접이웃을 구성하였다. 협업 필터링은 새로운 사용자가 들어왔을 때 고객들 간의 유사도를 분별할 수 없으므로 추천의 신뢰도가 떨어진다는 단점이 존재한다. 이를 Cold Start문제라 하는데 이 문제는 사용자의 평가 내역이 존재하지 않기 때문에 발생하는 문제이다. Cold Start 문제를 해결하기 위하여 처음에 사용자들의 성별, 연령, 직업, 선호 장르, 선호 tv채널 정보를 이용하여 Cold Start 문제를 해결한다. 성별, 연령, 직업, 선호 장르, 선호 tv채널 정보는 다음 <표1>과 같다.

<표 1> 사용자 성향 정보

번호	연령	성별	직업(대분류)	장르	선호tv채널
1	± 5	남	경영사무	코미디	MBC
2		여	영업,고객상담	액션	SBS
3			IT,인터넷	멜론	KBS1
4			디자인	개그	KBS2
5			서비스	시트콤	OCN
6			전문직	판타지	tvN
7			의료	범죄 스릴러	JTBC
8			생산제조	공포	TV조선
9			건설	순정	넷플릭스
10			유통무역	스포츠	
11			미디어	사극	
12			교육	시대극	
13			특수계층	드라마	

사용자 성향 정보에서 연령은 자신을 기준으로 ±5년 이내인 사람과 비교한 후 연령이 비슷할 경우 더 큰 유사도를 나타낼수 있도록 분류하였다. 본 논문에서 구현할 드라마 추천 시스템의 시나리오는 다음(그림 1)과 같다.



(그림 1) 드라마 추천 시스템 시나리오

① 사용자가 재밌게 봤던 드라마, 좋아하는 장르 등 의 정보를 얻기 위해 일련의 정보들을 입력 후 회원가입을 진행한다.

② 회원가입 후 회원이 된 사용자가 드라마 추천을 요구한다. 회원가입에 사용된 요소들은 추천알고리즘에서 개인화 요소로 사용하므로 서버에 전송한다.

③ 서버는 사용자 정보를 DB에 저장하고 근접이웃을 구성하기 위하여 DB로부터 사용자 정보를 요청한다.

④ DB에 데이터가 충분히 누적되었을 때 협업 필터링을 이용하여 최 근접이웃을 구성한다. 이때 협업 필터링은 사용자들의 관심 표현 및 선호도를 바탕으로 관심도, 선호도가 비슷한 사용자들을 분류하는 방법이다. 과거에 사용자들이 즐겨봤던 드라마의 장르가 비슷하다면 사용자 서로 유사한 성향을 가지고 있다고 판단한 뒤 그 판단을 근거로 추천하는 방식이다[1]. 협업필터링을 통한 추천방식에는 크게 사용자 기반(User-based)과, 아이템 기반(Item-based) 방식이 있으나, 본 논문에서는 사용자 기반(User-based) 협업필터링을 사용하였다. 사용자간 유사도를 측정하는 방법으로는 코사인 유사도(Cosine Similarity)를 이용하여 두 사용자(A, B)간의 유사도를 측정하였다.

⑤ 최 근접이웃이 재밌게 본 드라마 목록을 추출하여 추천 드라마 목록을 생성한다. 근접 이웃을 찾을때는 먼저 그룹화 되어있는 사용자 집단에 대해 비교한 후 드라마를 추천하는 방식과 비교하여 효율적인 방법을 찾는다.

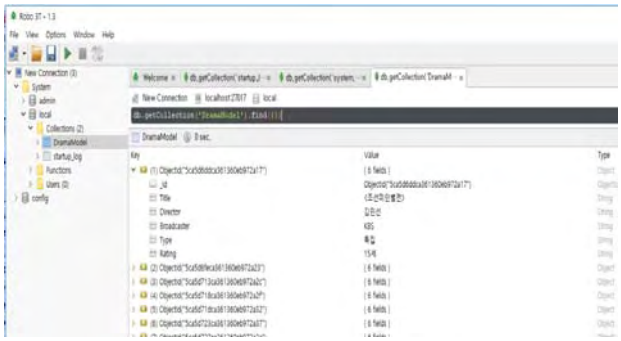
⑥ 추천 드라마 목록을 서버에 전달한다.

⑦ 서버에 전달된 추천 드라마 목록을 사용자에게 전달한다.

또한 추천이 완료되면 추천시스템의 추천 알고리즘에 대한 평가가 필요함으로 생성된 사용자데이터 3000개를 대상으로 MAE(Mean Absolute Error)를 이용한다.

3. 드라마 추천 시스템의 구현

본 시스템은 기본적으로 R Studio에서 구현하였다, 또한 Node.js로 서버를 구성하고 mongo DB로 데이터베이스를 구축하였다. 위 데이터베이스를 기반으로 R에서 추천 알고리즘, 개인화 요소 등의 데이터들을 다루게 된다. 아직까진 다양한 사용자들이 선호하는 드라마에 대한 데이터를 구할 방법이 없기에 사용자가 어떠한 연령, 성별, 직업을 가지고 있고 어떠한 장르, tv채널을 선호하는지에 대한 데이터는 임의로 생성한후 위 데이터들을 활용하여 추천시스템을 구현하였다.mongo DB 에서 생성한 데이터들을 가독성이 좋은 툴 ROBO3T 을 통하여 살펴보면 다음(그림 2)와 같다.



(그림 2) mongo DB에 저장된 드라마 데이터

드라마 데이터들을 저장후 신규 사용자가 추천을 요청하면 R을 통하여 개인요소들을 불러오며 결과는 다음 (그림 3)와 같다.

~/asdf - RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Doc_corpus

Filter

	User_number	genere	gender	age	job	channel
user1	1	3	1	25	1	1
user2	2	2	1	25	4	3
user3	3	3	1	22	5	2
user4	4	5	2	23	5	4
user5	5	8	2	31	6	5
user6	6	8	1	21	10	6
user7	7	9	2	40	8	7
user8	8	2	1	42	8	8
user9	9	1	2	15	7	1
user10	10	3	1	17	9	1
user11	11	4	2	34	3	1
user12	12	9	1	28	6	2
user13	13	7	1	24	5	5
user14	14	4	1	24	4	8
user15	15	2	2	27	2	3
user16	16	1	1	44	1	5
user17	17	3	2	32	11	4
user18	18	5	2	38	11	9
user19	19	4	2	27	12	9
user20	20	5	1	40	13	9

(그림 3) R에서 본 사용자 개인 요소

R Studio에서 "recommenderlab" 라이브러리를 이용하여 추천시스템을 구현하기 때문에 먼저 라이브러리를 설치해 준다. 관련 명령어를 입력하여 개발환경을 설정해주면 다음 (그림 4) 와 같다.

Console Terminal

```
package 'arules' successfully unpacked and MD5 sums checked
package 'proxy' successfully unpacked and MD5 sums checked
package 'registry' successfully unpacked and MD5 sums checked
package 'irlba' successfully unpacked and MD5 sums checked
package 'recommenderlab' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\hun\AppData\Local\Temp\Rtmp69mz82\downloaded_packages
> > install.packages("recommenderlab")
```

(그림 4) R Studio 개발환경

(그림 4)는 R Studio에 "recommenderlab" 라이브러리를 설치해준 그림이다. 이후 "recommenderlab" 에서 제공하는 함수들을 이용하여 근접이웃을 구성한다. 근접이웃을 구성할 때 코사인 유사도(Cosine similarity)를 이용하여 시스템 사용자들 사이의 유사도를 측정하고 이를 통해 근접이웃 리스트를 생성한다.코사인 유사도의 공식은 다음(그림 5)와 같다.

$$sim(A,B) = \frac{\sum_{i \in I_{AB}} R_{A,i} \; R_{B,i}}{\sqrt{\sum_{i \in I_{AB}} R_{A,i}^{\;2}} \sqrt{\sum_{i \in I_{AB}} R_{B,i}^{\;2}}}$$

(그림 5) 코사인 유사도(Cosine similarity)

이때 R은 n x m의 고객-상품 행렬이다. $R_{A,i}$, $R_{B,i}$는 사용자 A와 B가 공통으로 평가한 I 행렬의 각 아이템 I에 대한 평가치를 뜻한다. I_{AB} 는 A 사용자와 B 사용자가 공통으로 평가한 행렬이다. 따라서 A와 B가 공통으로 평가한 두 개의 아이템평가행렬 벡터의 사이 각을 구하여 최솟값을 가지는 최 근접이웃을 추출한다.

다음 555번 사용자는 경영사무 업종에 종사하며 남자이고 25살, 코미디 장르를 선호하고 MBC채널을 선호하는 사용자이며 다른 사용자들 간의 유사도를 비교한 후 제일 높은 사용자부터 차례로 나열한 데이터는 다음 (그림 6)과 같다.

corpus User_Sim

Filter

User_number	genere	gender	age	job	channel	similarity
555	4	1	25	1	1	1.000
342	3	1	17	1	2	0.882
87	3	1	25	10	1	0.855
910	7	1	19	5	4	0.752
844	5	2	22	1	1	0.726
311	4	1	30	2	1	0.718
520	7	2	30	2	5	0.512
667	9	2	18	8	6	0.442
3	3	1	22	5	2	0.420

(그림 6) 555번 사용자 유사도 리스트

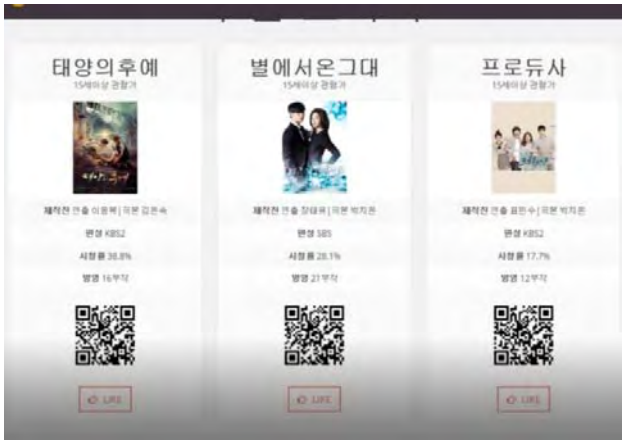
(그림 6)처럼 유사도 리스트가 생성되면 그중 유사도가 가장 높은 3명에게 드라마를 하나씩 총 3개의 드라마를 추천받는다. 위와 같은 방법은 사용자들 간의 유사도를 측정하여 유사한 성향을 지닌 사용자들이 추천 해주는 드라마를 사용자에게 추천 하는 방식이다. 또 하나의 방법으로 기존 사용자들 데이터 중 유사한 성향을 가지는 사용자들끼리 먼저 그룹화를 진행한다. 이후 새로운 사용자가 드라마 추천을 요청하면 유사한 그룹에 사용자를 추가하고 그룹에 속해있는 사용자들이 추천 해주는 드라마를 추천 해주는 방법이다. 다음 (그림 7)은 먼저 555번 사용자를 그룹화 한 후 그룹에 소속된 집단과 유사도 리스트를 생성한 화면이다.

	User_number	genere	gender	age	job	channel	similarity
user555	555	4	1	25	1	1	1.000
user2043	2043	4	1	20	1	1	0.921
user2272	2272	4	1	19	1	1	0.911
user224	224	4	1	22	2	1	0.907
user571	571	3	2	25	1	1	0.887
user342	345	3	1	17	1	2	0.882
user70	70	3	1	25	2	1	0.852
user667	779	7	1	25	2	1	0.810
user927	927	4	2	25	1	4	0.799

(그림 7) 555번 사용자 그룹화 유사도 리스트

이때 유사도가 가장 높은 그룹에 새로운 사용자를 추가하고 그룹에 있는 사용자들이 추천하는 드라마중 가장 중복이 많이되는 드라마순 으로 추천을 해준다. 중복되는 드라

마가 한 개도 없다면 가장 유사도가 높은 사람이 추천해준 드라마 순으로 드라마를 추천해준다. 555번 사용자가 추천받은 드라마는 다음
(그림 8)와 같다.



(그림 8) 555번 사용자가 추천받은 드라마

추천받는 두가지 방법을 통해서 추천받은 드라마 평점 데이터를 이용하여 추천 알고리즘의 평가를 위하여
MAE(Mean Abolute Error)를 이용한다. MAE의 공식은 다음 (그림 9)과 같다.

$$MAE = \frac{\sum_{i=1}^{q} |\text{실제고객평가치 } i - \text{예측된 평가치 } i|}{q}$$

(그림 9) MAE (Mean Abolute Error)

위 MAE공식에서 q는 사용자가 평가한 드라마의 개수이며, “실제 고객평가치 i”는 테스트 데이터에서 실제로 사용자가 평가한 드라마의 평가 데이터를 의미한다. 같은 의미로 “예측된 평가치 i“는 사용자를 제외한 근접 이웃인 평가한 평가 데이터의 평균으로서 근접 이웃의 평가 데이터 평균으로 사용자의 평가 데이터를 예측한 것 이다.

따라서 MAE는 실제 사용자가 평가한 데이터 값과, 근접 이웃이 예측한 사용자의 예측 데이터 값의 오차의 합을 q로 나눈 예측 데이터의 평균으로 실제 평가 데이터의 오차를 나타내는 지표이다.[2] 사용자 데이터3000개중 1800개의 데이터는 학습데이터, 1200개의 데이터는 테스트 데이터로 구성하여 테스트 데이터를 사용하여 MAE를 평가하면 다음 (그림 10)과 같은 결과를 얻을 수 있다.

Person MAE	Group MAE
1.342	1.255

(그림 10) 두 방법의 MAE결과 편차

Person MAE 는 전체 사용자 간의 유사도를 모든 사람과 비교한 후 예측 평가치와 실제 고객의 평가 차의 평균이며 Group MAE 는 기존 사용자들을 그룹핑 한 후 새로운 사용자를 그룹에 넣어 추천하는 드라마 평점을 받아 예측 평가치와 실제 사용자의 평가치 차이의 평균이다. MAE결과를 보면 Group MAE가 0.087만큼 낮은걸 확인 할 수 있다. 위 결과에 따르면 전체 사용자들의 유사도를 비교하여 리스트를 생성하여 추천하는 방식보다 그룹핑후 새로운 사용자를 그룹에 추가하여 유사한 사용자의 드라마를 추천해주는 방식이 신뢰도가 더 높다고 볼 수 있다.

4.결론

본 논문에서 빅 데이터화 되고 있는 수많은 드라마들을 사용자 기호에 맞게 추천해주는 시스템들 구현하였고, 협업필터링의 문제점 중 Cold Start의 희소성을 해결하기 위해 개인화 요소를 추가하여, 근접 이웃을 추출하여 드라마를 추천 하였을 때 기존의 방식보다 조금 더 신뢰성 있는 추천결과를 도출해내는 추천시스템이다. 향후 조금 더 정확한 추천을 위하여 각 드라마에 대한 사용자들의 정보 데이터를 수집하고 보다 많은 개인화 요소들을 검증해보아야 하며 더욱 적합한 개인화 요소와, Cold Start를 보안할 방법을 찾아야 할 것이다.

참고문헌

[1] 김영아, 박두순, “협업 필터링 기반 드라마 추천 시스템”, 한국정보처리학회 추계학술대회 발표 논문집, 제주 한라대학교, pp. 1137-1138, 2013.11
[2] 심대수, 김철환, 박진수, 박두순 “R에서 협업 필터링과 개인화 요인을 이용한 개인화 영화 추천 시스템”, 순천향대학교 컴퓨터소프트웨어공학과 2017.11