

클라우드 컴퓨팅에서 협업 필터링과 개인의 감정을 이용한 개인화 영화 추천 시스템

심대수*, 김민기*, 박두순*
 *순천향대학교 컴퓨터소프트웨어공학과
 e-mail : tlaetn123@naver.com

A Personalized Movie Recommendation System using Collaborative Filtering and Personal Sentiment in Cloud Computing Service

Dae-Soo Sim*, Min-Ki Kim*, Doo-Soon Park*
 *Dept. of Computer Software Engineering, SoonchungHyang University

요 약

정보화 시대에 들어오며 수많은 정보들의 폭발적인 증가로 인해 사용자들은 원하는 정보를 빠른 시간에 얻는 것이 어려워졌다. 그중 영화는 수없이 많은 정보를 누적해왔고 개인에 따라 선호하는 영화가 서로 다르기 때문에 각 개인에 맞는 영화를 찾는 것은 쉽지 않다. 본 논문에서는 협업 필터링과 개인의 감정을 이용하고 AWS(Amazon Web Service)를 통한 클라우드 컴퓨팅 시스템을 사용하여 각 개인에 더 적합한 영화 추천 시스템을 제안 한다.

1. 서론

정보의 홍수라 불리는 21세기 정보화 시대에 살아가며 수없이 많은 정보들이 생기는 것과는 반대로 수많은 정보 속에서 사람들이 개인에게 원하는 정보를 찾는 것이 오히려 어려워지고 있다. 그 중에서도 영화는 다른 콘텐츠들에 비해서 누적된 영화 콘텐츠뿐만 아니라 개봉되는 영화 콘텐츠도 무시할 수 없다. 또한 국제 영화의 분야와 장르, 종류가 날로 다양해지고 있으며 영화 시장의 규모도 매년 증가하는 추세이다. 2006년부터 2015년 까지 영화시장의 성장률은 (표 1)과 같다.

(표 1) 2006 - 2015 미국의 콘텐츠 시장 규모 변화[1]

구분	2006	2007	2008	2009	2010p	2011	2012	2013	2014	2015	2011-2015 CAGR(%)
영화	35,318	36,057	35,207	35,015	35,200	36,842	38,853	41,151	43,448	45,686	5.4
에이메이션	5,737	5,664	4,857	5,532	6,621	6,116	6,443	6,751	7,114	7,476	2.5
방송	155,008	158,842	160,865	154,329	164,450	170,255	183,985	189,794	203,081	210,848	4.8
게임	9,342	11,823	14,720	13,746	13,607	14,135	14,631	15,116	15,810	17,014	5.3
음악	11,728	10,615	8,667	7,524	6,599	6,432	6,342	6,323	6,373	6,476	-0.4
출판	113,274	112,061	102,385	88,808	87,238	87,269	88,144	89,660	91,440	93,614	1.4
만화	695	700	683	685	635	646	651	662	671	665	0.9
광고	199,195	198,957	187,878	160,848	169,537	173,083	185,131	189,892	201,676	207,851	4.2
지식정보	112,163	118,790	117,744	110,178	113,527	118,783	125,282	132,422	140,404	149,113	6.0
캐릭터	63,570	63,200	94,040	83,150	83,070	82,900	82,650	82,490	82,320	82,980	0.0
전체	455,520	469,713	461,737	429,912	443,144	458,842	484,835	503,901	531,718	555,277	4.6

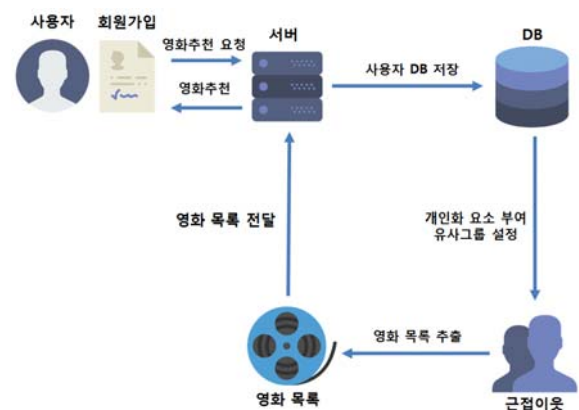
이와 같이 수많은 영화들의 개봉과 영화시장의 성장 속에서 각각의 사용자에게 알맞은 영화를 추천해 주는 시스

템은 왓챠(www.watcha.net)와 로튼 토마토(www.rottentomatoes.com) 등이 있으나 얼마나 사용자에게 잘 맞는지는 알기 어렵다. 또한, 사용자의 감정을 이용하여 영화를 추천해 주는 방법은 최근에 몇 개의 논문으로 제시되고 있다.

본 논문에서는 AWS(Amazon Web Service)의 클라우드 컴퓨팅 시스템에서 개인감정과 개인성향을 바탕으로 협업 필터링을 사용하여 사용자에게 보다 신뢰성 높은 영화를 추천하는 시스템을 제안한다.

2. 영화 추천 시스템의 구성

본 논문에서 구현하게 될 추천 시스템 시나리오는 (그림 1)과 같다.



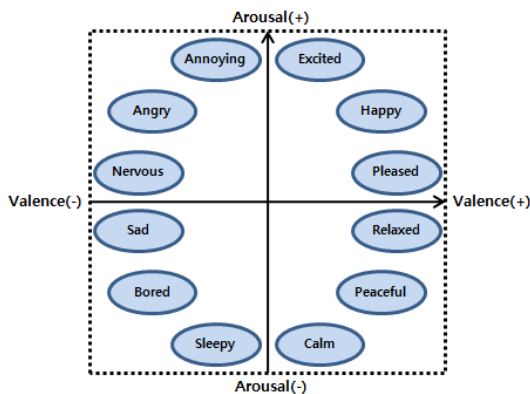
(그림 1) 추천 시스템 시나리오

(그림 1)의 추천 시나리오에서 첫 번째로 개인화 추천 시스템을 사용하기 위해 사용자가 회원가입 시 나이, 성별, 직업, 선호 카테고리 등의 개인화 요소를 입력 하여야 한다. 입력받은 사용자의 개인화 요소들과 웹 서버에서 받은 사용자의 감정 상태 DB를 클라우드 서버 데이터베이스에 저장한다.

두 번째로 회원가입 시 받은 개인화 요소와 웹 서버에서 받은 사용자의 감정상태에 협업 필터링을 이용하여 최근접 이웃을 구성한다. 여기서 협업 필터링이란, 사용자들의 선호도와 관심 표현을 바탕으로 선호도, 관심도가 비슷한 사용자들을 식별해 내는 방법으로 과거에 이용한 콘텐츠가 비슷하다면 사용자 간에 유사한 성향을 가지고 있다고 판단하고 그 근거를 토대로 추천하는 방식이다[2].

세 번째로 작성된 최근접 이웃의 영화 데이터를 추출한 뒤 사용자가 이미 본 영화는 필터링하여 사용자가 보지 않은 영화 데이터만을 추출한다. 네 번째로 서버에 영화 목록을 전달하여 사용자에게 영화 목록을 추천한다.

또한 본 논문에서 사용하는 개인화 요소 중 사용자가 영화추천 시 입력한 감정은 (그림 2) 과 같이 Thayer가 제안한 12개로 분류한 감정 모델을 사용한다.



(그림 2) 차원 접근법 감정 모델[3]

R.E. Thayer는 사람의 감정을 수치적으로 표현하기 위해 2차원 공간에 감정 상태를 나타내는 차원 접근법을 시도하였다. 차원접근법은 긍정적·부정성의 정도와 감정의 에너지 정도를 기준으로 감정을 분류한다[4].

따라서 본 논문에서는 Thayer의 감정 모델을 이용하여 ‘흥분되는’, ‘행복한’, ‘기쁜’, ‘느긋한’, ‘평화로운’, ‘차분한’, ‘졸린’, ‘지루한’, ‘슬픈’, ‘긴장된’, ‘화난’, ‘짜증나는’과 같은 12가지의 감정 상태를 이용하여 영화를 추천하고자 한다[4].

본 논문에서 사용되는 요소를 나이, 성별, 직업, 선호 장르, 감정으로 분류 하였는데 나이는 10살 단위로 연령을 분류하였다. 그리고 직업을 분류한 기준은 통계청에 있는 자료를 토대로 하여 대분류 항목을 기준으로 구성하였다. 개인화 요소의 분류 결과는 (표 2)와 같다.

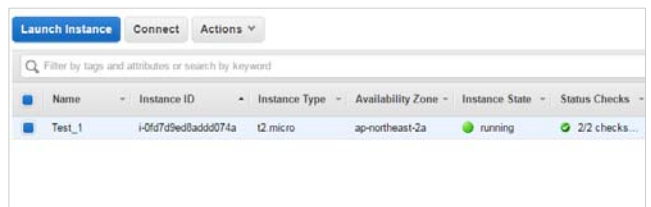
(표 2) 사용자 개인화 요소의 분류[5]

Index	Age	Gender	Job	Movie Genre
1	10대	남성	학생	드라마
2	20대	여성	관리자	액션
3	30대		전문가	스릴러
4	40대		사무직	공포
5	50대		서비스업	범죄
6	60대		판매직	판타지
7	70대		농림어업 종사자	애니메이션
8	80대		기능직	전쟁
9			기계 조작 및 조립 종사자	코미디
10			노무직	멜로
11			군인	미스터리
12				SF
13				사극
14				다큐
15				뮤지컬
16				성인물
17				기타

3. 영화 추천 시스템의 구현

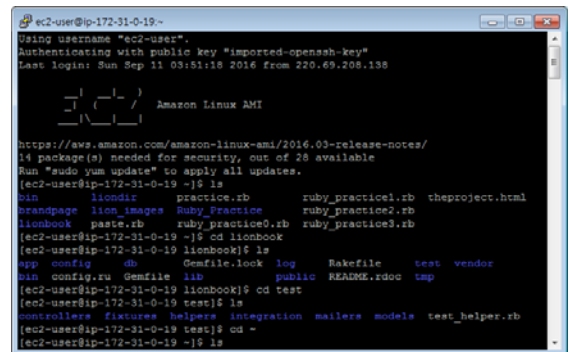
본 논문에서 구현한 영화 추천 시스템은 AWS의 대표적인 IaaS 서비스인 EC2 인스턴스를 사용하여 개발 환경을 구축하였다[6].

(그림 3)는 AWS EC2 인스턴스를 할당받아 서버를 구축한 모습이다.



(그림 3) AWS 콘솔 구축 환경

(그림 4)는 Putty 접속 프로그램을 이용하여 구축된 클라우드 환경에 접속한 모습이다.



(그림 4) 클라우드 환경 접속 화면

AWS EC2 인스턴스는 Linux 콘솔 환경으로 구성되어 있어, 클라우드 환경에 가장 빠르게 접속시킬 수 있는 Ruby on Rails 프레임 워크를 사용하여 개발하였다.

추천 시스템을 구현하기 위해서는 사용자에게 회원가입 시 필수적으로 개인화 요인들을 받아 DB화 하여야 한다. 회원가입 시 필요한 개인화 요인들로는 나이, 성별, 직업, 선호 장르 등이 있다. 반면에 영화를 찾을 시 입력되는 개인화 요소인 감정은 영화를 찾기 전 감정 선택 패널에서 선택되어 입력된다.

4개의 개인화 요소와 감정 정보를 AWS 클라우드 서버에

저장한다. (그림 5)는 AWS 클라우드 서버에 일부 회원들의 정보가 저장된 모습이며 저장된 순서는 ID, 이름, 성별, 나이, 직업, 선호 장르, 최근 감정 이다.

```
User Load (0.2ms) SELECT "users".* FROM "users"
=> #<ActiveRecord::Relation [#<User id: 1, user_id "claeoctn123", birth:"940617", sex: "Man", category_id: 2, job: "학생", movie_id: nil>, <User id: 2, user_id "donghyun94", birth:"940625", sex: "Man", category_id: 1, job: "학생", movie_id: nil>, <User id: 3, user_id "mingee95", birth:"950314", sex: "Woman", category_id: 2, job: "학생", movie_id: nil>, <User id: 4, user_id "sims000", birth:"911204", sex: "Man", category_id: 4, job: "학생", movie_id: nil>, <User id: 5, user_id "taktak123", birth:"930103", sex: "Man", category_id: 2, job: "학생", movie_id: nil>, <User id: 6, user_id "taehoon1994", birth:"940423", sex: "Man", category_id: 6, job: "학생", movie_id: nil>]>
```

(그림 5) 회원관리 User 데이터베이스

영화를 추천하기 위해서는 (그림 5)과 같이 회원들의 개인화 요소가 들어있는 데이터베이스뿐만이 아니라, 영화 데이터베이스가 필요하다. 본 논문에서는 Linux 환경인 AWS 클라우드 서버에서 wget 명령어를 사용하여 웹사이트의 정보를 가져왔으며, 웹사이트의 정보에서 영화제목, 영화장르들을 추출해 (그림 6)과 같이 데이터베이스를 구성한다.

```
Movie Load (0.2ms) SELECT "movies".* FROM "webtms"
=> #<ActiveRecord::Relation [#<Movie id: 1, name: "밀정", category_id: 2, rating: 8.54>, <Movie id: 2, name: "메카닉:리크루트", category_id: 2, rating: 7.5>, <Movie id: 3, name: "로빈슨 크루소", category_id: 7, rating: 8.4>, <Movie id: 4, name: "터널", category_id: 1, rating: 8.64>, <Movie id: 5, name: "라이엇", category_id: 4, rating: 7.7>]>
```

(그림 6) 영화 정보를 추출한 데이터베이스

(그림 6)의 사용자 데이터베이스를 이용해 근접 이웃을 구성할 수 있다. 이때 개인화 요소에 따라서, 가중치에 따라서, 감정에 따라서 모두 근접 이웃의 구성이 달라진다. (표 3)은 저장된 사용자 데이터베이스의 일부이다.

(표 3) 저장된 사용자 데이터베이스 일부

ID	Age	Sex	Job	Genre	Movie
User_1	20대	남자	학생	액션	
User_2	20대	남자	학생	스릴러	타임 패러독스
User_3	30대	남자	사무직	범죄	신세계
User_4	10대	남자	학생	액션	밀정
User_5	10대	여자	학생	멜로	뷰티 인사이드
User_6	50대	여자	서비스업	사극	광해
User_7	50대	남자	사무직	액션	다이하드3
User_8	10대	여자	학생	애니메이션	겨울왕국
User_9	20대	여자	학생	액션	테이큰
User_10	30대	여자	판매직	공포	쏘우2

개인화 요소를 나이, 성별, 직업의 세 가지를 사용했을 때와 나이, 성별, 직업, 장르의 네 가지를 사용했을 때의 추천 영화는 (표 4), (표 5)와 같다. (표 4)와 (표 5)를 비교해보면 추천된 영화가 부분 집합관계로 주어진 데이터는 장르에 매우 민감함을 알 수 있다.

(표 4) User_1에 대한 유사도와 근접이웃 구성

ID	Union	Intersection	Similarity	Movie
User_1	X	X	X	X
User_2	3	3	1	타임 패러독스
User_28	3	3	1	반지의 제왕
User_18	3	3	1	트라이앵글
User_29	3	3	1	링
User_4	4	2	0.5	밀정
User_24	4	2	0.5	써니
User_16	4	2	0.5	라퐁젤
User_9	4	2	0.5	테이큰
User_14	4	2	0.5	어벤져스2

(표 5) User_1에 대한 유사도와 근접이웃 구성

ID	Union	Intersection	Similarity	Movie
User_1	X	X	X	X
User_2	5	3	0.6	타임 패러독스
User_4	5	3	0.6	밀정
User_9	5	3	0.6	테이큰
User_14	5	3	0.6	어벤져스2
User_18	5	3	0.6	트라이앵글
User_28	5	3	0.6	반지의 제왕
User_29	5	3	0.6	링
User_11	6	2	0.333333	해리포터 2
User_16	6	2	0.333333	라퐁젤

이번에는 나이, 성별, 직업, 장르에 가중치를 두어 추천한다. 나이(20%), 성별(20%), 직업(20%), 장르(40%)와 나이(40%), 성별(20%), 직업(20%), 장르(20%) 그리고 나이(20%), 성별(40%), 직업(20%), 장르(20%)의 가중치를 주었을 때 추천된 영화들은 (표 6), (표 7), (표 8)과 같다. (표 6)은 추천된 영화가 많이 줄어든 것을 볼 수 있고 이는 장르에 매우 민감함을 알 수 있다. (표 7)은 가중치를 주지 않았을 때와 가장 유사한 추천을 하고 있다. 이는 나이에는 민감하지 않다는 것이다. (표 8)은 약간의 차이는 보이지만 나이보다는 민감하지만 일반적인 경향임을 알 수 있다. 결국 이 데이터에서는 가중치를 부여하지 않고 네 가지의 요소를 가지고 추천하는 것이 좋은 것을 알 수 있다.

(표 6) User_1에 대한 유사도와 근접이웃 구성

ID	Union	Intersection	Similarity	Movie
User_1	X	X	X	X
User_4	1.2	0.8	0.666667	밀정
User_9	1.2	0.8	0.666667	테이큰
User_14	1.2	0.8	0.666667	어벤져스2
User_2	1.4	0.6	0.428571	타임 패러독스
User_7	1.4	0.6	0.428571	다이하드3
User_18	1.4	0.6	0.428571	트라이앵글
User_28	1.4	0.6	0.428571	반지의 제왕
User_29	1.4	0.6	0.428571	링
User_24	1.6	0.4	0.25	써니

(표 7) User_1에 대한 유사도와 근접이웃 구성

ID	Union	Intersection	Similarity	Movie
User_1	X	X	X	X
User_2	1.2	0.8	0.666667	타임 패러독스
User_14	1.2	0.8	0.666667	어벤져스2
User_9	1.2	0.8	0.666667	테이큰
User_18	1.2	0.8	0.666667	트라이앵글
User_28	1.2	0.8	0.666667	반지의 제왕
User_29	1.2	0.8	0.666667	링
User_4	1.4	0.6	0.428571	밀정
User_7	1.6	0.4	0.25	다이하드3
User_16	1.6	0.4	0.25	라퐁젤

(표 8) User_1에 대한 유사도와 근접이웃 구성

ID	Union	Intersection	Similarity	Movie
User_1	X	X	X	X
User_2	1.2	0.8	0.666667	타임 패러독스
User_4	1.2	0.8	0.666667	밀정
User_18	1.2	0.8	0.666667	트라이앵글
User_28	1.2	0.8	0.666667	반지의 제왕
User_29	1.2	0.8	0.666667	링
User_9	1.4	0.6	0.428571	테이큰
User_11	1.4	0.6	0.428571	해리포터 2
User_14	1.4	0.6	0.428571	어벤져스2
User_16	1.4	0.6	0.428571	라퐁젤

또한 위의 결과에서 감정을 추가할 경우 근접 이웃의 구성이 또 바뀌게 된다. 다른 개인화 요소와 다르게 감정의 경우 2차원 모델을 사용하기 때문에 감정 간의 유사도를 사용해야 한다. (표 9)는 2차원 모델링된 12개의 감정 간의 유사도를 나타내는 표이며, 감정의 유사도는 Thayer가 제안한 2차원으로 구성된 12개로 분류한 감정 모델의 거리를 사용하여 측정하였다.

(표 9) 12개의 감정과 사이의 유사도표

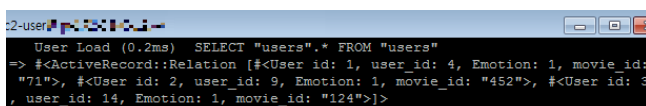
Emotion Similarity												
Emotion	흥분되는	행복한	기쁜	느긋한	평화로운	차분한	졸린	지루한	슬픈	긴장된	화난	짜증나는
흥분되는	1	0.8333	0.6666	0.5	0.3333	0.1666	0	0.1666	0.3333	0.5	0.6666	0.8333
행복한	0.8333	1	0.8333	0.6666	0.5	0.3333	0.1666	0	0.1666	0.3333	0.5	0.6666
기쁜	0.6666	0.8333	1	0.8333	0.6666	0.5	0.3333	0.1666	0	0.1666	0.3333	0.5
느긋한	0.5	0.6666	0.8333	1	0.8333	0.6666	0.5	0.3333	0.1666	0	0.1666	0.3333
평화로운	0.3333	0.5	0.6666	0.8333	1	0.8333	0.6666	0.5	0.3333	0.1666	0	0.1666
차분한	0.1666	0.3333	0.5	0.6666	0.8333	1	0.8333	0.6666	0.5	0.3333	0.1666	0
졸린	0	0.1666	0.3333	0.5	0.6666	0.8333	1	0.8333	0.6666	0.5	0.3333	0.1666
지루한	0.1666	0	0.1666	0.3333	0.5	0.6666	0.8333	1	0.8333	0.6666	0.5	0.3333
슬픈	0.3333	0.1666	0	0.1666	0.3333	0.5	0.6666	0.8333	1	0.8333	0.6666	0.5
긴장된	0.5	0.3333	0.1666	0	0.1666	0.3333	0.5	0.6666	0.8333	1	0.8333	0.6666
화난	0.6666	0.5	0.3333	0.1666	0	0.1666	0.3333	0.5	0.6666	0.8333	1	0.8333
짜증나는	0.8333	0.6666	0.5	0.3333	0.1666	0	0.1666	0.3333	0.5	0.6666	0.8333	1

(표 9)의 감정을 개인화 요소로 추가한 뒤 최종적으로 가중치를 나이(10%), 성별(10%), 직업(15%), 장르(35%), 감정(30%) 로 두었을 경우의 User_1에 대한 사용자들의 유사도와 근접 이웃은 (표 10)과 같다.

(표 10) User_1에 대한 유사도와 근접이웃 구성

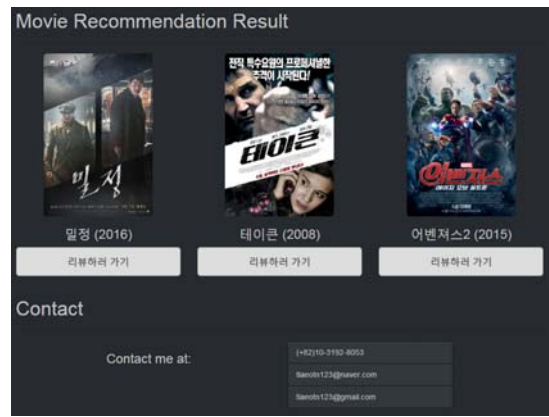
ID	Union	Intersection	Similarity	Movie
User_1	X	X	X	X
User_14	1.1	0.85	0.772727	어벤져스2
User_4	1.1	0.75	0.681818	밀정
User_9	1.2	0.8	0.666667	테이큰
User_7	1.25	0.65	0.52	다이하드3
User_29	1.35	0.6	0.444444	링
User_28	1.35	0.5	0.37037	반지의 제왕
User_2	1.35	0.5	0.37037	타임 패러독스
User_24	1.45	0.5	0.344828	써니
User_16	1.45	0.5	0.344828	라퐁젤

(표 10)과 같이 감정을 넣었을 경우 최 근접 이웃의 위치가 변화 하는 것을 볼 수 있다. 최종적으로 본 논문에서는 개인화 요인을 나이, 성별, 직업, 선호 장르, 감정 5가지의 개인화 요소를 사용하였으며, 각각의 가중치는 나이(10%), 성별(10%), 직업(15%), 선호 장르(35%), 개인감정(30%) 으로 가중치를 두어 근접 사용자를 추출한다. (그림 7)은 추출된 근접 사용자와 그 사용자들이 추천한 영화이다.



(그림 7) 추출된 유사 사용자와 영화목록

마지막으로 작성된 근접 그룹이 추천한 영화를 추천한다. (그림 8)은 20대, 남자, 학생, 액션, '흥분된' 이라는 개인화 요소를 두었을 때 추천되는 영화를 나타낸 것이다.



(그림 8) 사용자에게 추천된 영화 결과

(그림 8)에서 영화 리뷰남기기 버튼을 클릭하면 평점과 리뷰를 남길 수 있다.

4. 결론

본 논문에서는 수많은 영화 정보들 사이에서 사용자 개인에게 적합한 영화를 추천하기 위하여 개인화 요소와 개인감정을 이용한 협업필터링 시스템을 사용하며, AWS에서 지원하는 클라우드 컴퓨팅 기술을 이용해 구현 하였다. 구현된 추천 시스템을 이용하여 여러 가지 추천들을 해서 결과를 비교분석 하였다.

개인화 요소를 나이, 성별, 직업의 세 가지를 사용했을 EO와 나이, 성별, 직업, 장르의 네 가지를 사용했을 때를 비교해 보면 추천된 영화가 부분 집합관계로 주어진 데이터는 장르에 매우 민감함을 알 수 있다.

다음으로는 개인화 요소에 가중치를 준 경우와 가중치를 부여하지 않은 경우에 오히려 가중치를 부여하지 않은 방법으로 추천하는 것이 좋다. 또한, 감정을 넣었을 경우 추천이 달라짐을 볼 수 있다.

참고문헌

- [1] 해외 콘텐츠시장조사(미국), 한국콘텐츠진흥원, 2012.
- [2] 김영아, 박두순, “협업 필터링 기반 드라마 추천 시스템”, 한국정보처리학회 추계학술대회 발표 논문집, 제주 라대학교, pp. 1137-1138, 2013.11
- [3] Robert E. Thayer, The Biopsychology of Mood and Arousal, Oxford University Press, 1989.
- [4] 김선호, 개인의 감정과 협업필터링을 이용한 개인화 영화 추천 시스템, 순천향대학교 컴퓨터소프트웨어공학과 졸업논문, 2014.05
- [5] “한국표준 직업분류 표_대분류”, 통계청, 2015.
- [6] 이건호, 박두순, “클라우드 컴퓨팅에서 구축한 협업필터링 기반 웹툰 추천 시스템”, 한국정보처리학회 춘계 학술대회 발표 논문집, 동국대학교, pp. 451-454, 2016.05