차애캐와 애니메이션 추천 시스템

CUAI 5기 RecSys 10팀
김병준(컴퓨터공학), 김정우(경영학), 김찬호(응용통계학), 박정현(응용통계학)

[요약] 본 논문은 콘텐츠 기반(Contents-based)과 협업 필터링(Collaborative filtering) 추천 시스템을 활용해 사용자가 선호할 확률이 높은 캐릭터와 애니메이션을 제안한다. 추천에는 'Anime-planet'과 'My Anime List'에서 제공된 데이터셋을 전처리 후 입력 데이터로 사용하였다. 추천 결과는 합리적이었지만 각 모델이 갖는 한계점과 그에 대한 개선 방향을 명시하였다.

1. 서 론

OTT 플랫폼 산업과 통신 기술 발달에 따라 언제 어디서든 문화 콘텐츠를 소비할 수 있는 환경이 구축되었다. 여러 콘텐츠 중에서도 애니메이션에 대한 수요는 지속적으로 증가하고 있으며 애니메이션 플랫폼 '라프텔'은 'YouTube'에 이어 국내 기준 사용자 체류 시간 2위를 기록했다. 이는 많은 사람들이 애니메이션과 캐릭터에 매력을 느끼고 있음을 의미한다.

따라서, 본 논문은 소비자의 애니메이션 시청에 긍정적 경험을 확장하기 위해 소비자의 최애캐릭터(이하 최애캐)와 유사한 차애캐릭터(이하 차애캐)를 추천하는 모델과 애니메이션을 추천하는 모델을 제안한다. 사용한 추천 방법론으로는 콘텐츠 기반 추천, 협업 필터링이 있다.

2. 본 론

본 논문은 'Anime-planet'에서 제공하는 애니메이션, 애니메이션 캐릭터, 애니메이션 평점 등을 담고 있는 데이터셋을 활용하여 콘텐츠 기반 추천 시스템과 협업 필터링 추천 시스템을 구축하였다. 콘텐츠 기반 추천 시스템을 구축할 때 word 수준의 속성(Feature)만을 사용하는 시스템과 sentence 수준의 속성(Description, Synopsis)를 사용하는 시스템으로 각각 구축하였다.

콘텐츠 기반 시스템의 Input은 애니메이션의 경우 'Naruto', 캐릭터의 경우 'Light YAGAMI'를 사용하였다.

가. 콘텐츠 기반 추천 시스템

1) 차애캐 추천

가) Feature 기반 추천

'Anime-planet'의 애니메이션 캐릭터 데이터셋의 'Gender', 'Hair Color', 'Tags' 속성을 사용하였다. 해당 속성은 모두 명목형 자료형이기 때문에 Sklearn의 CountVectorizer를 사용하여 각 속성에 등장하는 데이터에 고유한 정수 인덱스를 부여하여 단어 집합을 생성하고 각 인덱스의 위치에 해당되는 단어 토큰의 등장 횟수를 기록하여 one-hot vector형태로 변경하였다. 최종적으로 얻은 행렬은 13888 x 506 크기를 가지는 행렬이다. (13888은 캐릭터의 수, 506은 단어 집합의 수)

앞서 얻은 행렬을 자기 자신과의 행렬 유사도를 활용하여 기반 추천 시스템을 구축하였다. 행렬 간 유사도 측정 방법에는 코사인 유사도를 사용하였다.

$$similarity = \cos(\theta) = \frac{A \cdot B}{||A||\,||B||} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n}(A_i)^2} \times \sqrt{\sum_{i=1}^{n}(B_i)^2}}$$

얻어진 유사도 행렬은 13888 x 13888 크기를 가지는 행렬로 대각 요소는 모두 1을 가진다. (자기 자신과의 유사도는 1이기 때문이다.) 유사도 행렬 A에서 i 행의 데이터를 내림차순으로 정렬하여 top-k개를 선정하여 추천된 항목의 index를 얻을 수 있다.

```
Deishuu KAIKI
Gakuhou ASANO
Tsukasa SHISHIOU
Seiya KANIE
Johan LIEBERT
Shen Qing Qiu
Yu Wenzhou
Akushima
Fukusuke HIKYAKUYA
Ichiya SUZAKU
```

[그림 1] Feature 기반 시스템 캐릭터 추천 결과

나) Description 기반 추천

'Anime-planet'은 애니메이션 캐릭터의 성별, 특징 등 다양한 정보를 'description'과 함께 제공한다. Description은 캐릭터의 성격, 다른 캐릭터와의 관계 및 특이 사항을 포함한다. Description은 word level이 아닌 sentence level의 텍스트 데이터이므로 Feature 기반 추천과 분리된 추천 시스템을 구축하였다. 사용자의 최애캐를 기준으로 다른 모든 캐릭터와의 유사도를 계산해야 하므로 각 description에서 어떤 단어가 어느 정도의 중요도를 갖는 지 계산하는 TF-IDF 알고리즘을 사용하였다.

TF-IDF(단어 빈도-역문서 빈도)는 TF(Term Frequency)와 IDF(Inverse Document Frequency)를 곱한 값으로 식은 다음과 같다.

$$tf(t,d) = \frac{0.5 \times f(t,d)}{\max\{f(t,d) : w \in d\}}$$

$$idf(t,D) = \log\frac{|D|}{|\{d \in D : t \in d\}|}$$

$$tf \times idf$$

TF-IDF는 전체 문서에서 자주 등장하는 단어의 중요도는 낮게 평가하는 반면, 특정 문서에서만 여러 차례 사용되는 단어의 중요도는 높게 평가하는 특징이 있다. 따라서, description에 TF-IDF를 적용해 벡터화하고 벡터 행렬을 생성해 유사도를 비교해 최애캐릭터와 유사한 top-k개의 캐릭터를 도출하였다. 유사도 함수로 코사인 유사도를 사용하였다.

```
Sachiko YAGAMI
Heiji HATTORI
Kizaru
Cybersix
Kazunobu CHIBA
The Raven
Ritsu TAINAKA
Tsumugi KOTOBUKI
Dorry
Medusa GORGON
```

[그림 2] Description 기반 시스템 캐릭터 추천 결과

2) 애니메이션 추천

가) Feature 기반 추천

'Anime-planet'의 애니메이션 데이터를 사용하였으며 차애캐릭터 추천의 Feature 기반과 동일한 CountVectorizer를 사용한 행렬 유사도 방법을 적용하였다. 사용한 특성은 'Genre', 'Type', 'Source'이다.

```
Medaka Box
Shaman King
Boruto: Naruto Next Generations
Dragon Ball Z: Atsumare! Gokuu World
Rekka no Honoo
Dragon Ball Kai (2014)
Boruto: Jump Festa 2016 Special
Katekyo Hitman Reborn!
Naruto: Shippuuden
D.Gray-man
```

[그림 3] Feature 기반 시스템 애니메이션 추천 결과

나) Synopsis 기반 추천

'My Anime List'는 애니메이션의 장르, 방영일 등과 간략한 줄거리가 담긴 'synopsis' 데이터를 제공한다. 전술한 차애캐 추천과 같은 방법으로 TF-IDF 알고리즘을 사용했으며 마찬가지로 코사인 유사도를 사용해 top-k개의 애니메이션 추천 결과를 도출하였다.

```
Boruto: Naruto Next Generations
Naruto: Shippuuden
Boruto: Naruto the Movie
Naruto: Shippuuden Movie 6 - Road to Ninja
Naruto SD: Rock Lee no Seishun Full-Power Ninden
Naruto: Takigakure no Shitou - Ore ga Eiyuu Da...
The Last: Naruto the Movie
Naruto: Shippuuden Movie 4 - The Lost Tower
Naruto: Shippuuden Movie 2 - Kizuna
Naruto: Akaki Yotsuba no Clover wo Sagase
Naruto: Shippuuden Movie 1
```

[그림 3] Synopsis 기반 시스템 애니메이션 추천 결과

나. 협업 필터링 추천 시스템

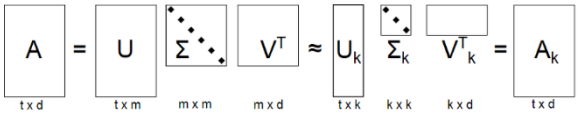
협업 필터링 추천 시스템은 유사한 콘텐츠를 추천하는 콘텐츠 기반 추천 시스템과 다르게 많은 사용자들로부터 얻은 평점 정보를 바탕으로 한 사용자가 좋은 평가를 할 것으로 예측되는 아이템을 추천한다.

본 연구진은 'My Anime List'에서 제공하는 애니메이션 평점 데이터셋을 사용하였으며 평점은 1부터 10까지 구성되어 있다. 또 Surprise 라이브러리를 사용하여 협업 필터링을 구현하였다. Surprise 라이브러리에서 제공하는 여러 알고리즘 중 가장 좋은 성능 즉, 가장 낮은 RMSE(Root Mean Squared Error)나 MAE(Mean Absolute Error)를 보이는 알고리즘을 선택하여 연구를 진행했다. Cross Validation을 통해 여러 알고리즘의 성능을 비교한 결과는 다음과 같다.

ALGORITHM	TEST_RMSE	TEST_MAE
SVD	1.0564	0.7801
SLOPEONE	1.2064	0.9022
NMF	1.8523	1.5859
KNNBASELINE	1.1582	0.8625
KNNBASIC	1.1920	0.8840
KNNWITHMEANS	1.1685	0.8729
KNNWITHZSCORE	1.1764	0.8708
BASELINEONLY	1.2085	0.9038
COCLUSTERING	1.1924	0.8917

[그림 4] Surprise 라이브러리를 이용한 협업 필터링 알고리즘 성능 교차 검증 결과

교차 검증 결과, SVD 알고리즘이 가장 좋은 성능을 보인다. SVD(Singular Value Decomposition, 특잇값 분해)는 행렬 분해 기법의 일종으로 사용자-아이템 평점 행렬을 3개의 행렬(사용자 행렬(V) X 특잇값 행렬(S) X 아이템 행렬(U))로 분해해서 이를 학습시키고 이 3개의 행렬로 원래의 행렬을 재현하여 해당 사용자가 평가하지 않은 아이템에 대한 평점을 예측하는 기법이다. 본 연구에서 사용자는 애니메이션에 평점을 매긴 사용자이며, 아이템은 애니메이션이다.



[그림 5] SVD 행렬 분해 작동 방식

실제 행렬(A)과 예측 행렬(A_K)의 RMSE와 MAE 계산 시 오류가 가장 적게 나온 파라미터를 찾기 위해 Grid Search를 진행한 결과, 각 파라미터 값 SGD의 반복 횟수(n_epochs)는 17, SVD의 잠재 요인(K)의 크기는 2000일 때 가장 오류가 적게 도출되었다.

위에서 결정된 파라미터를 이용하여 모델을 훈련시킨 후 한 사용자가 평점을 매기지 않은 애니메이션에 한해 평점을 예측한다. 예측된 평점 중 가장 높은 평점을 매길 것으로 예측된 상위 10개의 애니메이션을 추천해주는 방식이다. 사용자 아이디(user_id)가 478인 사용자에게 추천된 애니메이션 10개는 다음과 같다.

```
Katekyo Hitman Reborn! : 9.083834449003131
Mononoke : 8.968436672746423
Major S5 : 8.896757860526067
Hunter x Hunter (2011) : 8.856850018649558
Mahou Shoujo Madoka★Magica Movie 3: Hangyaku no Monogatari : 8.789397204110024
Gintama° : 8.768553224869489
Kimi no Na wa. : 8.759830850041032
Gintama. : 8.715619723113676
Re:Creators : 8.70757873411669
Tsurezure Children : 8.691663300444697
```

[그림 6] 협업 필터링 추천 시스템 추천 결과

추천 결과를 살펴보면, 애니메이션 이름과 함께 예상 평점을 볼 수 있다. 'Kateyo Hitman Reborn!'부터 'Tsurezure Children'까지 예상 평점이 내림차순으로 정렬되어 있다. 판타지, 스포츠, 액션 등 장르적으로 다양한 애니메이션들이 추천되었음을 알 수 있다.

3. 결 론

1) 콘텐츠 기반 추천

Feature 기반 추천은 여러 후보를 선정하여 결과를 확인했을 때 실제 추천된 상위 k개의 후보 중 실제 연관성이 높으면서 참신한 후보도 확인하였지만 전혀 연관성이 없을 것으로 생각되는 후보가 여럿 등장하는 것을 확인하였다.

CountVectorizer 방법을 사용하였는데 모든 속성에 동일한 가중치를 적용하였기 때문에 다음과 같은 두 가지 문제점을 가지는 것을 확인하였다.

1. 모든 속성이 동일한 가중치를 가진다.
   추천 시스템을 구축할 때 추천 대상마다 중요도를 알맞게 설정하는 과정이 필요하지만 이를 간과하여 비교적 덜 중요한 속성이 잡음처럼 작용할 수 있음을 확인하였다.
2. 데이터셋이 가지는 여러 feature에 등장하는 단어의 수가 일정하지 않다.
   'Tag' 속성의 경우 다른 속성에 비해 등장하는 단어의 가짓수가 다른 속성에 비해 높기 때문에 추천 결과에 영향을 미치는 정도가 다른 속성에 비해 크고 애니메이션의 경우 같은 애니메이션에 등장하는 캐릭터들은 대부분 같은 'Tag'를 공유하고 있을 것이기 때문에 Feature 기반 추천 시스템의 추천 결과에서 확인할 수 있듯 같은 애니메이션에 등장하는 캐릭터가 여럿 확인된다.

이러한 문제점은 속성을 Vectorize할 때 속성에 따라 가중치를 다르게 설정하거나 데이터셋에 양질의 feature를 추가하여 해소할 수 있을 것으로 예상한다.

캐릭터의 description과 애니메이션의 synopsis 데이터에 기반한 추천은 TF-IDF 알고리즘을 사용한다. 추천 결과를 보면 애니메이션으로는 비슷한 시리즈의 애니메이션을 위주로, 차애캐로는 같은 애니메이션에 등장한 캐릭터 위주로 추천한다는 점을 확인할 수 있다. 이와 같은 문제점은 고유 명사에 부여하는 가중치가 크기 때문에 발생한 문제로 같은 애니메이션에 등장한 캐릭터나 같은 시리즈의 애니메이션을 추천 대상에서 배제하는 방법으로 개선할 수 있다.

참고 문헌

1) Charu C. Recommender systems. Vol. 1. Cham: Springer International Publishing, 2016.
2) Zhang, Yin, Rong Jin, and Zhi-Hua Zhou. "Understanding bag-of-words model: a statistical framework." International journal of machine learning and cybernetics 1.1 (2010): 43-52.
3) Ramos, Juan. "Using tf-idf to determine word relevance in document queries." Proceedings of the first instructional conference on machine learning. Vol. 242. No. 1. 2003.
4) Schafer, J. Ben, et al. "Collaborative filtering recommender systems." The adaptive web. Springer, Berlin, Heidelberg, 2007. 291-324.