희박한 고객 활동 데이터에서 최신성 기반 추천 성능 향상 연구

백상훈, 김주영, 안순홍
아시아나 IDT
ICT융합연구소

요 약

최근 AI를 산업 서비스에 적용하기 위해 많은 회사들이 활발히 연구를 하고 있다. 아마존과 넷플릭스 같은 거대 기업들은 이미 빅데이터와 AI 머신러닝을 이용한 추천 시스템을 구현하였고 아마존은 매출의 35%가 추천에 의해 발생하고 넷플릭스 75%의 사용자가 추천을 통해 영화를 선택한다고 보고되었다. 이러한 두 기업의 높은 추천 효율성의 이유는 협업 필터링(Collaborative filtering)과 같은 다양한 추천 알고리즘과 방대한 상품 및 고객 행동(구매, 시청 등) 데이터 등이 존재하고 있기 때문이다. 기계학습에서 알고리즘 학습을 위한 데이터의 양이 많지 않을 경우 알고리즘의 성능을 보장할 수 없다는 것이 일반적인 의견이다. 방대한 데이터를 가진 기업에서 추천 알고리즘을 적극적으로 활용 및 연구하고 있는 것도 이러한 이유 때문이다. 반면, 오프라인 및 여행사 기반에서 온라인 기반으로 영역을 차츰 확대하고 있는 항공 서비스 고객 데이터의 경우, 산업의 특성상 많은 회원에 비해 고객 1명당 온라인에서 활동하는 이력이 많지 않은 것이 특징이다. 이는, 추천 알고리즘을 통한 서비스 제공에서 큰 제약사항으로 작용한다. 본 연구에서는, 이러한 희박한 고객 활동 데이터에서 최신성 기반의 추천 시스템을 통하여 제약사항을 극복하고 추천 효율을 높이는 방법을 제안한다. 고객의 최근 접속 이력 로그를 시간 기준으로 데이터 셋을 분할하여 추천 알고리즘에 반영하였을 때, 추천된 노선에 대한 고객의 반응을 추천 성능 지표인 CTR(Click-Through Rate)로 측정하여 성능을 확인해 보았다.

1. 서론

최근 빅데이터를 이용한 AI 연구가 크게 늘고 있다. 넷플릭스와 아마존 등의 업체들을 중심으로 AI 머신러닝을 이용한 추천 시스템을 도입하고 있고 실제로 넷플릭스에서 대여되는 영화의 75%는 머신러닝에 의한 추천으로 발생하며 개인화된 비디오 추천은 전체 평점이 높은 비디오 추천보다 2~3배의 시청 효과가 있는 것으로 나타났다. [1]

일반적으로 추천 시스템을 적용한 산업은 아마존과 넷플릭스 같은 대량의 아이템(뉴스, 음악, 영화, 상품 등) 속에서 고객에게 적합한 아이템을 추천해 줌으로써 고객의 시간을 절약하여 주고 만족도를 높여 직접적인 매출로 이어지도록 하고 있다.

추천 기법은 크게 콘텐츠 기반, 협업 필터링 기반으로 분류할 수 있다. 콘텐츠 기반 추천은 사용자의 선호도와 아이템 특성 간의 유사도를 계산하고 사용자에게 유사도를 기반으로 추천하는 방식이고 협업 필터링 기반 추천은 사용자의 기록(구매, 평가, 로그 등)을 이용하여 사용자 간의 유사도를 행렬 형태로 모델링 하여 사용자 간 유사도 기반으로 추천하는 방식이다. 일반적으로 협업 필터링 기반의 추천은 성능을 높이기 위해 사용자의 기록(구매, 평가, 로그 등) 양이 많아야 하며 비교할 수 있는 사용자도 많아야 한다는 전제 조건이 있다. 따라서 아마존과 같은 E-Commerce 업종과 넷플릭스와 같은 OTT(Over The Top) 서비스에서 방대한 양의 아이템이 있고 수많은 사용자의 행위(구매, 시청 등)가 빈번히 일어날 경우 추천 시스템의 효과를 볼 수 있다. 실제 두 기업은 추천 시스템으로 상당한 이득을 얻고 있다.

반면, 오프라인 및 여행사 기반 산업에서는 아이템 수가 적고 사용자의 행위가 많지 않아 추천 시스템의 효과를 얻기가 힘들다.

본 논문에서는 상대적으로 아이템의 수가 적고 사용자의 행위(구매)가 적은 항공 데이터를 이용하여 협업 필터링 기반의 추천 기법을 적용 시 사용자에게 적합한 추천이 되는지 증명한다.

2. 배경

2-1. Matrix Factorization 알고리즘

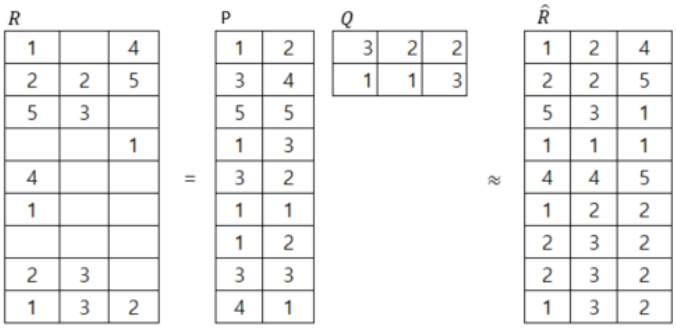
행렬 인수 분해(Matrix Factorization)는 협업 필터링 알고리즘(collaborative filtering algorithm) 중 하나의 추천 기법이다.

사용자와 상품이라는 두 가지의 요소를 가진 데이터에 대해, 사용자(User)가 상품(Item)에 대한 평점을 R_{ui}라고 할 때 해당 값을 아래와 같은 행렬(Matrix) 방식으로 표현할 수 있다.

	Item1	Item2	Item3	Item4	…	ItemN
User1	5	2	2	3	2	2
User2	1	1	0	4	3	1
User3	3	3	3		4	
User4	5			5		
User5	4	3	1	1	5	2
…	2	3		1	1	3
UserN	1	3	4	1	1	2

[표 1. User-Item Matrix]

하지만 위와 같이 일반적으로 *User* 당 모든 *Item*의 평가 값이 존재하기 힘들다. 행렬 인수 분해(Matrix Factorization) 추천 시스템에서는 이 존재하지 않는 값을 예측하는 것이다.



[그림 1. Matrix Factorization]

R은 *User* 수 n, *Item* 수 m을 가지는 n X m 매트릭스라고 볼 수 있다. R은 각각 n X k, k X m인 두 개의 행렬 P와 Q로 분해될 수 있다고 가정해 본다. 분해된 두 개의 행렬을 P X Q 계산을 하면, 원래의 matrix R과 가장 유사한 형태의 행렬이 되면서 기존의 행렬에서 없던 값들이 생성된다. P, Q는 각각 User-latent factor matrix, Item-latent factor matrix라는 각각의 내재적 의미를 나타내는 잠재 행렬로 나타난다. 이러한 특징은 기계가 해석하기 위한 블랙박스 모델이라고 볼 수 있다.

Latent 행렬을 각각 P, Q라고 했을 때 Matrix Factorization의 목적 함수(Objective Function)는 다음과 같다.

$$\sum_{(u,i)\in k} (r_{ui} - q_i^T p_u)^2 + \lambda(||q_i||^2 + ||p_u||^2)$$

[수식 1. Matrix Factorization의 목적 함수]

즉, Matrix Factorization에서는 목적 함수(Object Function)를 최소화하는 것이 목표이다.

2-2. 관련 연구

현재까지 OTA(Online Travel Agency) 기업들은 항공 티켓 추천을 효율적으로 하기 위해 다양한 기법을 사용하고 있다. 예를 들면 Google Flight의 경우 고객의 출발 시간과 도착 시간, 부가적인 여행 정보를 이용하여 가장 합리적인 티켓을 추천해준다. Ctrip Flight의 경우 가장 할인율이 높은 인기 노선을 Tag로 제공하고 있다. [2][3]

현재까지 개인화 항공 티켓 추천 알고리즘은 항공 데이터의 특성 때문에 cold-start problem[4]에 직면하게 된다. 때문에 부가적인 데이터 요소를 추가하여 cross-domain 추천 방식을 혼합하는 추천 알고리즘 방식을 사용하거나[5] 추가적인 사용자 정보(Social Relationship 등)를 모델에 적용하여 추천하는 방식[6]의 연구가 진행되었다.

하지만 이러한 방식들의 알고리즘은 양질의 데이터를 확보하기가 쉽지 않다. 또한 추가적인 요소를 적용할 때 최적의 방법을 찾기 까지 상당한 기간이 소요되고 계산의 복잡도가 증가하여 항공 산업에 바로 적용하기 어렵다.

본 논문에서는 Matrix Factorization 알고리즘에 시간 개념을 적용[7]하여 계산의 복잡도를 줄이고 추천 성능을 개선할 수 있는지 실험을 하였다.

3. 본 연구

3-1. 항공 데이터의 특징

항공 노선 구매 데이터는 일반적인 상품 구매 데이터와 다른 패턴을 보인다. 첫 번째로 판매하는 상품의 개수가 상대적으로 매우 한정적이다. 항공 데이터에서 상품의 종류는 항공사가 취항하는 전 세계의 공항 수이기 때문에 다른 판매 상품과 달리 고정적이고 변동이 거의 없다. 두 번째로 고객의 재 구매가 드물게 일어난다. 항공 티켓을 구매하기 위해 사이트에 접속하는 빈도도 상대적으로 다른 E-Commerce 사이트에 비해 매우 낮으며 접속 주기도 상당히 길다.

3-2. 최신성 기반 데이터 분할

본 논문에서 사용되는 데이터는 일반적으로 사용되는 추천 알고리즘의 데이터와는 다른 특성의 데이터이기 때문에 데이터 접근에 대한 두 가지 가설을 정의했다.

첫 번째, 특정 연휴 기간, 명절, 여름 휴가철 등 항공 수요가 많은 기간에 최근 접속한 고객들의 성향이 비슷할 것이라는 가설을 세웠다.

두 번째, 항공 티켓은 일반적인 제품보다 고가이고 여행 일정에 따라 영향을 많이 받는다. 때문에 실제로

구매로 이루어지기까지 수 시간에서부터 수 일의 시간이 소요된다는 점에서 가까운 과거에 접속한 고객은 가까운 미래에 다시 접속할 확률이 높다는 가설을 세웠다.

$$B_t(c_i) = \{b_k^{c_i}: \text{고객 } c_i \text{의 행동 이력}, c_i \in N_t(c)\}$$
$$N_t(c) = \{c_i: \text{t시점에 접속한 회원}\}$$
$$b_i: \text{i번째 행동이력}, \text{t: 날짜}, \text{c}_\text{i}: \text{회원 번호}, \text{i} = 1, \dots, \text{n}$$

[수식 2. 최신성 기반 데이터 분할]

3-3. 최신성 기반 추천 로직

최신성 기반의 데이터를 통한 알고리즘 추천 로직은 다음과 같은 수식으로 정리할 수 있다.

$$Rec_i = f_i^{MF}(B_t(c_i)), \qquad dt - i \le t < dt$$
$$dt: \text{기준일}, i: \text{기간}$$

[수식 3. 최신성 기반 추천]

추천 로직에 들어가는 고객 이력 데이터는 기준일 dt를 기준으로, $dt - i \le t < dt$ 사이에 접속한 고객들을 대상으로 정의한다. 예를 들어, dt가 2019년 01월 02일이고 i가 1인 경우, $N_t(c)$는 2019년 01월 01일 하루에 접속한 고객들을 의미하며, $B_t(c_i)$는 2019년 01월 01일 하루에 접속한 고객들의 행동 이력을 의미한다. Rec_i 는 2019년 01월 01일에 접속한 고객 $N_t(c)$ 을 대상으로 Matrix Factorization을 진행하여 나온 추천 결과를 의미한다.

4. 실험

본 실험은 2018년 7월 1일부터 2019년 6월 30일까지의 A 항공의 홈페이지 사용자 로그 데이터를 사용하여 진행되었다. 추천 상품은 A 항공이 취항하는 직항 도시 74(인천, 김포 제외)개로 선정하였다. 전체 데이터 중 접속 국가가 한국, 접속 언어가 한국어, 검색 기록은 3회 이상, 탑승 이력이 1회 이상 되는 고객의 로그를 선별하여 학습을 진행한다. 데이터의 개수는 총 *12,426,899*건, 선별 조건에 해당하는 고객 수는 *502,018*명이다. 추천 알고리즘은 Matrix Factorization 기법을 사용하였다.

4-1. 실험 방법

웹 로그 내 고객의 노선 검색 데이터를 최신성 기준으로 분할하여 실험 데이터 셋을 구축하였다. 이때 기준일은 데이터의 마지막 시간인 2019년 6월 30일로 정하였다. 또한, 데이터 기간을 5가지(1일, 7일, 14일, 30일, 90일)로 정의하여 고객을 5개의 군집으로 분할하였다. 또한, 분할된 고객을 기반으로 데이터 셋을 분할하였다

$$B_t(c_i) = \begin{cases} B_{t_1}(c_i), c_i \in N_{t_1}(c), & dt - 1 \le t_1 < dt \\ B_{t_7}(c_i), c_i \in N_{t_7}(c), & dt - 7 \le t_7 < dt \\ B_{t_{14}}(c_i), c_i \in N_{t_{14}}(c), & dt - 14 \le t_{14} < dt \\ B_{t_{30}}(c_i), c_i \in N_{t_{30}}(c), & dt - 30 \le t_{30} < dt \\ B_{t_{90}}(c_i), c_i \in N_{t_{90}}(c), & dt - 90 \le t_{90} < dt \end{cases}$$

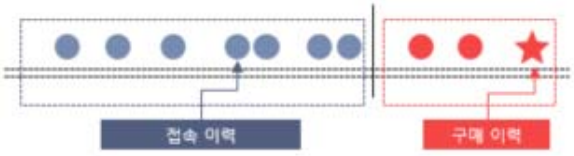
[수식 4. 실험 데이터 셋]

각 데이터 셋은 다음과 같은 정의를 통해 학습 데이터와 평가 데이터로 나눠진다.

$B_{t_1}(c_i)$	일별 데이터셋	훈련 데이터	어제 접속한 고객의 검색 이력
		평가 데이터	오늘 접속한 고객의 검색 이력
$B_{t_7}(c_i)$,	주별 데이터셋(1주)	훈련 데이터	1주 전 접속한 고객의 검색 이력
		평가 데이터	이번 주에 접속한 고객의 검색 이력
$B_{t_{14}}(c_i)$,	주별 데이터셋(2주)	훈련 데이터	2주 전에 접속한 고객의 검색 이력
		평가 데이터	1주 전부터 접속한 고객의 검색 이력
$B_{t_{30}}(c_i)$	월별 데이터셋	훈련 데이터	저번 달에 접속한 고객의 검색 이력
		평가 데이터	이번 달에 접속한 고객의 검색 이력
$B_{t_{90}}(c_i)$,	분기별 데이터 셋	훈련 데이터	저번 분기에 접속한 고객의 검색 이력
		평가 데이터	이번 분기에 접속한 고객의 검색 이력

[표 2. 데이터 셋 구분 기준]

실험 비교 대상 군은 전체 데이터 셋을 7:3(훈련 데이터:평가 데이터) 형태로 다음과 같이 분류하여 진행하였다.



[그림 2. 실험 비교 대상 군 훈련, 평가 데이터 구분]

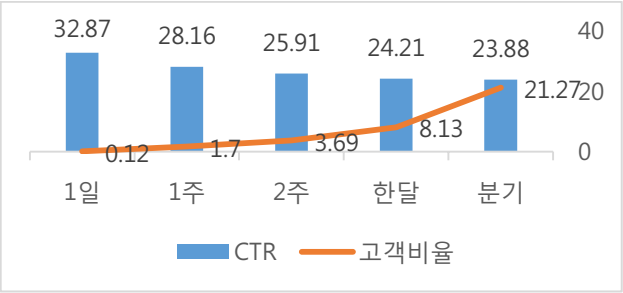
4-2. 실험 결과

추천 결과의 개수에 따른 추천 CTR의 결과 값은 다음과 같다.

데이터 셋	CTR (노선:3)	CTR (노선:5)	CTR (노선:10)	고객 수	고객비율
1일	29.24%	32.87%	39.62%	578	0.12%
1주	22.38%	28.16%	37.25%	8,558	1.70%
2주	20.02%	25.91%	35.60%	18,508	3.69%
한달	18.14%	24.21%	35.04%	40,808	8.13%
분기	16.87%	23.88%	35.77%	106,797	21.27%
일반 추천	5.08%	8.56%	17.35%	502,018	100%

[표 3. 추천 노선 수 별 CTR 결과]

[표 3]의 결과를 기반으로 추천 노선의 수가 많고 데이터 셋의 구성 시기가 최근인 경우 CTR이 증가하는 것을 확인할 수 있다. 본 연구에서는 웹 페이지에 표출하기 적합한 형태로 추천 노선 수를 5개로 정의했다. 내용을 도식화 해보면 [그림 3]으로 표현할 수 있다.



[그림 3. 데이터 셋의 구성 시간과 CTR, 대상 수 비교]

위 결과를 통해 학습 데이터 세트를 전체 데이터로 한 번에 적용했을 때 보다 주기 별로 짧게 적용할수록 CTR이 높게 나오는 것을 알 수 있었다.

본 연구에서 앞서 제시한 두 가지 가설, *'최근 접속한 고객들의 성향이 비슷하다.'*, *'가까운 과거에 접속한 고객은 가까운 미래에 다시 접속할 확률이 높다.'* 을 실험을 통해 검증했다. 그 결과 두 가설 모두 의미가 있음을 확인하였다.

5. 결론

본 연구에서는 상품 수가 적고 고객의 재 구매 횟수가 상대적으로 적은 항공 데이터로 고객에게 적합한 추천이 가능함을 검증하였다. 데이터 셋을 고객의 최근 접속 시점 별로 구분하여 학습한 결과가 구분 없이 전체를 사용한 데이터보다 더 좋은 결과를 보였다. 이를 통해 최근 접속 시기가 짧은 데이터 셋일수록 더 좋은 추천 성능을 보임을 알 수 있었다. 그리고 앞서 제시한 두 가지 가설에 대해 의미 있는 증명을 보였다. 또한 산업의 비즈니스 특성이 반영된 학습 데이터 셋을 기반으로 한 군집화를 통해 보다 높은 추천 성능이 나타남을 증명하였다.

즉 항공 개인화 노선 추천의 경우 고객의 최근 이용 시점에 따라 데이터 셋을 분류 구성하여 Matrix Factorization 추천 시스템이 적용가능 하다는 것을 증명하였다.

하지만 최신성 데이터를 사용할 경우 추천 대상 고객 수가 줄고 장기 미 접속 고객에 대한 추천 방법이 없다는 단점이 있다.

향후 고객의 추가적인 특성을 반영한 데이터 셋의 군집화와 외부 영향이 반영된 요소를 적용시켜 단점을 보완하고 추천 성능을 향상시킬 수 있는 방법에 대해서 연구할 계획이다.

참고문헌

[1]https://www.mckinsey.com/industries/retail/our-insights/how-retailers-can-keep-up-with-consumsers

[2] Google Flight. Available at: https://www.google.com/flights.

[3] Ctrip Flight. Available at: http://flights.ctrip.com.

[4] Mirbakhsh, Nima, and Charles X. Ling. "Improving top-n recommendation for cold-start users via cross-domain information." ACM Transactions on Knowledge Discovery from Data (TKDD) 9.4 (2015): 33.

[5] Cao, Jian, et al. "PFS: a personalized flight recommendation service via cross-domain triadic factorization." 2018 IEEE International Conference on Web Services (ICWS). IEEE, 2018.

[6] Qi Gu ; Jian Cao ; Yafeng Zhao ; Yudong Tan IEEE, "Addressing the Cold-Start Problem in Personalized Flight Ticket Recommendation"

[7] Grabocka, Josif, Alexandros Nanopoulos, and Lars Schmidt-Thieme. "Classification of sparse time series via supervised matrix factorization." Twenty-Sixth AAAI Conference on Artificial Intelligence. 2012.

[8] Leskovec, Jure, Anand Rajaraman, and Jeffrey David Ullman. Mining of massive datasets. Cambridge university press, 2014.

[9] Koren, Yehuda, Robert Bell, and Chris Volinsky. "Matrix factorization techniques for recommender systems." Computer 8 (2009): 30-37

[10] Candès, Emmanuel J., and Benjamin Recht. "Exact matrix completion via convex optimization.", 2009

[11] Kennedy, Ryan. "Low-rank matrix completion.", 2013

[12] Paterek, Arkadiusz (2007). "Improving regularized singular value decomposition for collaborative filtering" (PDF). Proceedings of KDD Cup and Workshop.

[13] Fang, Yi, and Luo Si. "Matrix co-factorization for recommendation with rich side information and implicit feedback." Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems. ACM, 2011.

[14] Zhao, C., et al. "HYBRID MATRIX FACTORIZATION FOR RECOMMENDER SYSTEMS IN SOCIAL NETWORKS." Neural Network World 26.6 (2016): 559.