

기계학습 기반의 뉴스 추천 서비스 구조와 그 효과에 대한 고찰*: 카카오의 루빅스를 중심으로

박 승 태**

(신한카드 AI랩, 컴퓨터공학 박사)

성 인 재***

(카카오 루빅스 TF)

서 상 원****

(에이팀벤처스)

황 지 수*****

(카카오 루빅스 TF)

노 지 성*****

(카카오 루빅스 TF)

김 대 원*****

(카카오 정책지원팀, 언론학 박사)

* 본 논문의 필자들은 카카오에 속해 있을 때, 논문 작업에 참여했습니다.

** park.seungtaek@gmail.com

*** tevin.sung@kakaocorp.com

**** sangwon.s@gmail.com

***** mario.h@kakaocorp.com

***** justin.time@kakaocorp.com

***** egofree@gmail.com, 교신 저자

요 약

본 논문에서는 카카오가 2015년 6월부터 다음 뉴스에 적용한 루빅스(RUBICS)의 구조와 성과를 고찰한다. 루빅스는 다양한 추천 알고리즘으로 구성된 앙상블(Ensemble) 추천 시스템이며, 각 이용자의 성향을 반영한 개인화 추천과 이용자 집단 내의 트렌드를 반영한 추천을 통합적으로 제공한다. 본 논문에서는 루빅스의 초기(콜드 스타트(Cold-Start) 이용자가 다수인 상황)에 주요 알고리즘으로 사용된 멀티암드밴딧(Multi-Armed Bandit) 중심의 뉴스 추천 시스템을 설명한다. 이어 알고리즘 성능을 추적하고 개선하기 위해 사용한 성능평가 방법을 설명하고 이를 통한 점진적인 알고리즘 개선 프로세스에 대해 논의한다. 마지막으로 루빅스 도입 이전과 이후의 뉴스 서비스의 성과 지표 비교를 통해 루빅스의 효과를 설명한다. 루빅스가 다음 모바일 뉴스에 도입되기 전인 2015년 4월과 도입(2015년 6월) 후인 2015년 8월의 뉴스 소비를 비교한 결과, 루빅스가 적용된 후 일평균 클릭 수는 130% 증가했으며, 일평균 뉴스 이용자도 45% 늘었다. 또한 첫 화면에 노출되는 뉴스 콘텐츠의 양도 250% 증가하였다. 이는 루빅스가 이용자 경험의 개선을 통해 이용자의 증가와 뉴스 내의 다양성 확보에 효과가 있음을 보여준다.

키워드: 기계학습, 뉴스 추천 서비스, 카카오, 루빅스, 멀티암드밴딧 알고리즘

1. 서론

인터넷의 태동은 이용자들에게 상시적인 정보의 홍수(Information Overload) 상태¹⁾를 초래했다. 이를 해결하기 위해 인터넷의 무한한 정보들에서 이용자가 관심을 가질만한 정보를 추출해 내는

1) 정보의 과잉으로 인해 이용자가 모든 정보를 보고 가치의 유무를 판단할 수 없는 상태를 말한다.

기술(Information Filtering)로 개발된 것이 검색(Information Retrieval)과 추천(Recommendation)이다. 검색 기술은 이용자가 원하는 정보를 ‘검색어(Query)’를 통해 특정하여 찾아주는 방식이다. 추천 기술은 보다 능동적인 방식으로 ‘검색어’와 같은 이용자의 명시적인 요청이 없어도 이용자의 평상시 서비스 사용 패턴을 기반으로 하여 이용자가 선호할만한 콘텐츠를 제공해주는 구조다.²⁾

기계학습(Machine learning)은 기계가 일일이 코드로 명시하지 않은 동작을 데이터로부터 학습하여 실행할 수 있도록 하는 알고리즘을 개발하는 연구 분야(Samuel A. L., 1959)로, 검색과 추천 등의 다양한 분야에 오래전부터 활용되어 왔다. 최근 들어서는, 영미권을 중심으로 구글(Das, Datar, Garg, & Rajaram, 2007; Liu, Dolan, & Pedersen, 2010), 야후(Agarwal, Chen, & Elango, 2009; Chu et al., 2009; Li, Chu, Langford, & Schapire, 2010), 그리고 페이스북(Mosseri, 2016) 등에서 기계학습을 이용한 이용자 행동기반의 뉴스 추천 서비스³⁾가 발전되어 왔다. 국내 인터넷 업계에서도 기계학습 기반의 다양한 개인화 뉴스 추천 서비스가 시도되었지만 정교한 이용자 프로파일 생성의 한계⁴⁾와 중복 콘텐츠 처리의 어려움으로 내부 평가 단계를

2) 추천은 3R로 다음과 같이 정의되기도 한다: “Delivering the Right Contents to the Right user at the Right moment.”

3) 일반적으로 이용자들이 뉴스 서비스에서 본 기사목록기록, 그 목록의 기사들 중 어떤 기사를 클릭해서 그 기사를 얼마나 오래 열람했는지에 대한 기록 등의 데이터를 추천 시스템이 학습하여 이용자에게 적절한 뉴스를 추천해서 목록을 제공해주는 서비스를 의미한다.

4) 많은 이용자들이 뉴스 이외의 다른 서비스를 사용하기 위해 다음 또는 네이버를 방문한다. 또한 대부분의 뉴스 이용자는 로그인(log-in)을 하지 않는 이용자들이며, 하나의 고정형 개인용 컴퓨터(Personal computer, PC)를 적어도 2명 이상의 이용자가 함께 사용하는 경우(남편과 아내, 그리고 아이들이 공유)도 많다. 이러한 경우 여러 명의 이용자 사용 이력이 혼합되는 문제가 발생한다. 이에 따라, 일반적으로 고정형 PC체계 중심의 뉴스 서비스에서 이용자에 대한 정교한 프로파일링(Profiling)이 어렵다. 이러한 문제가 모바일 환경에서는 개인 장비(Personal Device)인 스마트폰의

넘어서지 못했다. 그러나 개인별 단말기를 사용하는 모바일 시대를 맞아 이러한 문제들을 일정 부분 극복한 맞춤형 콘텐츠 추천 서비스가 등장하기 시작했다.

카카오는 2015년 6월부터 국내 최초로 뉴스 서비스에 기계학습 기반의 추천 시스템인 루빅스(RUBICS)를 도입했다. 루빅스의 도입으로 이용자들은 동일한 뉴스 서비스 화면에 노출되는 것이 아니라 각 이용자의 관심사에 맞는 뉴스들로 이루어진 개인화된 뉴스 서비스 화면을 소비할 수 있게 됐다. 루빅스의 알고리즘은 모바일 다음 서비스부터 적용되면서 국내 미디어 시장 그리고 학계로부터 주목을 받고 있다.

구글과 야후 등 전 세계 인터넷 기업들은 자사의 알고리즘을 학술 논문의 방식으로 공개해 왔다. 그러나, 국내에서 인터넷 기업의 알고리즘이 공개되는 것은 이번이 처음이다. 그동안 기업 영업 비밀의 영역으로 간주되어 비공개로 유지되어 온 기계학습 기반의 뉴스 추천 알고리즘을 설명한 본 논문이 국내 언론학계에 인공지능 활용에 대한 건전한 논의의 장(場)이 형성되는 데 이바지하길 본 연구진은 기대한다.

본 논문은 우리 사회의 궁금증을 일정 부분 해소하고, 나아가 미디어 시장에서 점차 확산되고 있는 기계학습 기반의 알고리즘 구조 공개에 대한 학계의 요청에 부응하는 취지에서 작성됐다. 본 논문은 설정한 가설의 맞고 그릇을 좇기 보다는 알고리즘의 주요 내용을 상술하는 방식을 택했다. 루빅스의 도입 배경과 구체적인 작동 원리, 루빅스에 적용된 성능 평가 방식, 이후 성과 등이 본 논문에서 소개된다.

2. 논의 배경

1) 기계학습 기반의 추천 시스템

기계학습은 알고리즘이 데이터로부터 스스로 배워서 더 나은 성과를 낼 수 있는 시스템이다. 기계학습은 데이터로부터 학습을 기반으로 특정 대상의 행동 예측에 방점을 두고 있기에, 특정 정보의 추천 체계에 폭넓게 활용되어 왔다.

이를 위한 추천 알고리즘은 크게 두 가지로 나뉜다. 내용기반 필터링(Content-Based Filtering)과 협업 필터링(Collaborative Filtering) 알고리즘이다. 내용기반 필터링에서는 1) 각 콘텐츠의 메타 정보(Meta data)⁵⁾에서 특성(Features)을 추출하여 콘텐츠의 특성 프로파일(Item profile)을 생성하고, 2) 이용자가 선호했던 콘텐츠들에서 자주 나타나는 특성들을 추출하여 이용자의 선호도 프로파일을 만든 뒤, 3) 새로 유입되는 콘텐츠들 중 생성된 프로파일과 유사한 콘텐츠들만을 추출하여 이용자에게 제공한다(Pazzani, 1999).

협업 필터링 시스템은 이용자와 콘텐츠 소비성향이 비슷한 유사 이용자들을 찾아내고, 이용자가 소비하지 않았지만 유사 이용자군에서 많이 소비된 콘텐츠를 추천해 주는 체계이다. 협업 필터링 시스템은 1990년대 후반 이후 추천에서 가장 많이 사용되고 있다.

내용기반 필터링과 협업 필터링⁶⁾의 주요한 차이는 내용기반 필터링은

-
- 5) 콘텐츠의 특성을 보여주는 모든 정보를 포함한다. 영화를 예로 들면 감독, 배우, 대본가, 줄거리, 평론가 및 이용자 평점, 상영 시간, 장르, 평론가 및 이용자 리뷰 내용, 개봉일, 흥행 정보(박스 오피스)등 영화와 관련된 모든 사항이 정보에 포함된다.
 - 6) 다양한 협업 필터링 알고리즘이 1990년대 이후 연구되어 왔다. 해당 예시로는 베이시안 네트워크 모델(Bayesian network methods)(Breese, Heckerman, & Kadie, 1998), 분류기 모델(Classifier methods models)(Billsus & Pazzani

특정 개인의 행동 정보만을 사용하는 반면, 협업 필터링에서는 특정 개인뿐만 아니라 이용자와 비슷한 성향의 다른 이용자들의 정보(Community Information)도 함께 이용한다는 점이다. 협업 필터링 추천은 내용기반 필터링에 비해 신선한 추천(Serendipity Recommendation)⁷⁾이 가능하고, 다른 이용자들의 정보도 활용하기 때문에 내용기반의 추천에 비해 이용자의 '행동 데이터'(Behavioral Data)⁸⁾가 상대적으로 많지 않더라도 추천이 가능하다. 또한 콘텐츠 메타 데이터를 사용하지 않아 언어 종속적인 텍스트 분석⁹⁾이 필요하지 않고, 복잡한

1998; Miyahara & Pazzani, 2000), 군집화 모델(clustering methods)(Ungar & Foster 1998; Xue et al., 2005), K 최근접 이웃 모델 (Top-K nearest neighbor methods)(Breese et al., 1998; Deshpande & Karypis, 2004; Herlocker, Konstan, Borchers, & Riedl, 1999; Karypis, 2001; Resnick, Iacovou, Suchak, Bergstrom, & Riedl, 1994; Shardanand & Maes, 1995; Sarwar, Karypis, Konstan, & Riedl, 2001), 행렬분해 모델(Matrix factorization)(Billsus & Pazzani, 1998; DeCoste, 2006; Goldberg, Roeder, Gupta, & Perkins, 2001; Sarwar, Karypis, Konstan, & Riedl, 2000; Rennie & Srebro, 2005), 그래프 모델(Graph-based methods)(Aggarwal, Wolf, Wu, & Yu, 1999; Huang, Chen, & Zeng, 2004), 그리고 통계 모델(Probabilistic methods)(Hofmann & Puzicha, 1999; Pennock, Horvitz, Lawrence, & Giles, 2000; Popescul, Pennock, & Lawrence, 2001) 등을 들 수 있다.

- 7) 협업 필터링의 추천은 내용기반 필터링과 달리 이용자가 평소 좋아하는 콘텐츠와의 내용적인 유사성이 전혀 없는 콘텐츠도 추천이 가능하다. 예를 들어, 협업 필터링 체계에서는 평상시 스포츠 뉴스만 보는 이용자 A에게도 경주 지진에 대한 뉴스 추천이 가능하다. 스포츠 뉴스를 많이 보는 다른 사용자들이 경주 지진에 대한 뉴스를 많이 본 행동 정보가 협업 필터링 추천에도 활용되기 때문이다. 그러나 내용기반 필터링은 이용자의 행동 정보만을 근거로 추천을 수행하기 때문에, 이용자 A의 평소 관심사가 아닌 경주 지진이 이용자 A에게 추천되는 것은 불가능하다.
- 8) 이용자의 행동 데이터는 서비스에서 이용자가 하는 모든 행동을 기록한 정보다. 뉴스 서비스의 경우, 이용자가 어떤 뉴스를 읽었으며, 해당 뉴스를 읽은 시간이 얼마인지, 그리고 이용자가 서비스를 주로 사용하는 시간 등이 중요한 행동 데이터로 취급된다.
- 9) 언어에 따라 각기 다른 형태소 분석 기술을 사용한다. 영어와 한국어는 언어 구조가 달라서 같은 기술을 적용할 수가 없기에, 두 언어의 분석을 위해서는 각기 다른 형태소 분석기를 사용해야 한다.

‘특성 추출 및 선택’(Feature Extraction & Selection) 기술¹⁰⁾이 필요하지 않다. 그러나 협업 필터링은 기존에 많이 축적된 이용자의 행동 정보에만 의존하여 새로운 콘텐츠 보다는 종래의 콘텐츠를 주로 추천하며, 데이터가 부족한 상황에서 추천 품질이 저하되는 ‘콜드 스타트’ 문제(Cold-Start Problem)를 내재하고 있다.

‘콜드 스타트 문제’는 이용자의 행동 데이터가 충분히 모이지 않아 추천이 어려운 상황을 지칭한다. 콜드 스타트에는 다음과 같은 3가지 상황이 존재한다. 첫 번째는 새로운 콘텐츠가 제공됐을 때 충분한 수의 이용자가 새로운 콘텐츠를 소비하기 전에는 해당 콘텐츠가 추천되지 않는 ‘아이템 콜드 스타트’(Item cold-start)로 새 콘텐츠가 추천 시스템에 신규 인입됐을 때 발생한다. 두 번째는 새로운 이용자의 행동 패턴을 분석할 수만큼 충분한 양의 행동 데이터가 모이기 전에 해당 이용자에게 추천을 제공할 수 없는 ‘이용자 콜드 스타트’(User cold-start)이다. 세 번째는 추천 서비스 자체가 출시된 지 오래되지 않아 이용자 행동 데이터가 전반적으로 부족해서 추천 품질이 저하되는 ‘시스템 콜드 스타트’(System cold-start) 이다(Park, Pennock, Madani, Good, & DeCoste, 2006).

콜드 스타트 문제를 해결하기 위하여 다양한 방법이 시도됐다. 시스템 콜드 스타트 문제를 해결하기 위해서 이용자-콘텐츠 행렬(User-item Matrix)의 차원을 줄이는 방법인 Latent Semantic Indexing(LSI)과 Principal Component Analysis(PCA)가 적용됐다(Billsus & Pazzani, 1998; Goldberg et al., 2001; Sarwar et al., 2000).

10) 메타 데이터에서 추출된 특성 중에 특히 중요한 특성을 뽑아내는 기술을 의미한다. 성능이 우수한 알고리즘을 사용할 경우 데이터 내의 군더더기(Noise)를 제거하는데 효과적이다. 데이터 내의 군더더기 제거는 분석 대상 특성의 수를 줄임으로써, 알고리즘의 학습 속도와 확장성(Scalability)의 향상에 기여한다.

내용기반 필터링은 새로운 콘텐츠를 추천할 때 특히 유용하다. 새로운 콘텐츠들에 대한 다른 이용자들의 행동 데이터 유무에 상관없이 과거에 이용자가 선호했던 특성들을 많이 가지고 있으면 이용자 대상의 추천이 가능하기 때문이다. 이러한 장점 때문에, 내용기반 필터링과 협업 필터링을 결합한 다양한 혼합(Hybrid) 알고리즘들이 고안됐다(Balabanović & Shoham, 1997; Basilico & Hofmann, 2004; Basu, Hirsh, & Cohen, 1998; Good et al., 1999; Melville, Mooney, & Nagarajan, 2002; Park et al., 2006; Popescul et al., 2001; Schein, Popescul, Ungar, & Pennock, 2002). 예컨대 이용자의 선호도에 따라 내용 기반 또는 협업 필터링의 가중치를 주는 방법과 데이터가 없는 초기 콜드 스타트 상황에서 내용기반의 추천에 가중치를 두고, 추후 데이터가 많아질 경우 자동적으로 협업 필터링 기반의 추천에 더 많은 가중치를 두는 방법 등이 제안됐다.

혼합 알고리즘도 행동 데이터와 이용자 정보가 전혀 없는 새로운 이용자에게 추천을 제공하는데 어려움을 겪는다(Park et al., 2006). 새로운 이용자를 위한 일반적인 추천 방법 중 하나는 해당 시점에 가장 인기 있는 콘텐츠를 추천하는 것이다. 예를 들어, 야후에서는 칼만 필터(Kalman Filter)를 이용하여 실시간으로 뉴스의 인기도를 측정했다(Agarwal et al., 2009). 이후 야후는 이용자의 선호도와 멀티아미밴드(Multi-Armed Bandit, MAB)을 결합한 컨텍스츄얼 밴드(Contextual MAB) 알고리즘을 고안했다(Li et al., 2010).

추천 시스템은 다양한 분야에 적용되어 왔다. 영화는 추천 시스템이 가장 먼저 광범위하게 적용된 분야이다. 초기의 몇몇 영화 추천 시스템들(예시, EachMovie¹¹⁾와 MovieLens¹²⁾)은 이용자 평점 데이터들을

11) URL: <http://www.cs.cmu.edu/~lebanon/IR-lab/data.html#intro>

12) URL: <http://grouplens.org/datasets/movielens>

연구자들에게 제공하여 추천 알고리즘 개발을 가속화 시켰다. DVD 렌탈 서비스인 넷플릭스(Netflix)가 주최한 넷플릭스 프라이즈(Netflix Prize)¹³⁾가 추천 알고리즘의 고도화에 불을 붙였다. 넷플릭스는 1억 건의 이용자 평점 데이터를 공개하고, 이를 바탕으로 자신들의 추천 알고리즘보다 나은 알고리즘을 개발하는 참가자를 시상하는 대회를 열었다. 넷플릭스 프라이즈는 가장 성능이 우수한 알고리즘을 개발한 팀(Bell & Koren, 2007a; 2007b; 2007c; Bell, Koren, & Volinsky, 2007)에게 총 100만 달러의 상금을 제공하는 대회로 언론·과학계의 주목을 받으며, 다양한 알고리즘의 개발을 독려하는 역할을 했다.

음악은 추천 시스템이 활발히 적용된 또 다른 영역이다. 링고(Ringo)(Shardan & Maes, 1995)는 초기 음악 추천 시스템들 중 하나로, 이 시스템을 위해 개발된 핵심 기술들은 이후에 반스앤드노블닷컴(barnesandnoble.com), 지디넷(ZDnet), 론치닷컴(launch.com)(2001년 6월 야후에 피인수됨) 그리고 마이 야후(MyYahoo!) 등에 이식됐다. 이후 스포티파이(Spotify), 애플(Apple)의 지니어스(Genius) 등 다양한 음악 추천 시스템이 개발됐다. 이 이외에도 다양한 영역, 이를테면 책(Linden, Smith, & York, 2003), 웹페이지(Balabanović & Shoham, 1997; Terveen, Hill, Amento, McDonald, & Creter, 1997), 농담(Joke) (Goldberg et al., 2001), 논문(Cosley, Lawrence, & Pennock, 2002; Middleton, Shadbolt, & De Roure, 2004), 광고¹⁴⁾등에 다양한 추천 알고리즘이 적용됐다.

13) URL: <http://www.netflixprize.com/>

14) 구글 특허, Automated collaborative filtering in world wide web advertising, US 5918014 A, URL: <https://www.google.com/patents/US5918014>

2) 뉴스 추천 알고리즘

뉴스 서비스도 추천 방식이 오래 전부터 연구되어 왔던 영역이다. 미국 미네소타대학 컴퓨터공학과에서 개발한 그룹렌즈(GroupLens)는 1994년부터 이용자 기반 협업 필터링(User-user Collaborative Filtering) 알고리즘을 사용하여 유즈넷(Usenet)에서 뉴스를 추천해왔다. 이후 그룹렌즈는 혼합 알고리즘을 사용한 탱고(Tango) 시스템을 개발했다(Claypool, Gokhale, Miranda, Murnikov, Netes, & Sart in, 1999; Konstan, Miller, Maltz, Herlocker, Gordon, & Riedl, 1997; Miller, Riedl, & Konstan, 1997; Resnick et al., 1994). 상업적인 뉴스 서비스에서 추천이 적용된 대표적인 예는 구글 뉴스(Google News)와 야후 첫 페이지(Yahoo Frontpage)다.

구글 뉴스의 초기 추천 알고리즘은 동시방문수(Covisitation), 민해쉬(MinHash), 그리고 PLSI(Probabilistic Latent Semantic Indexing)이란 세 가지 협업 필터링 방식을 결합한 ‘앙상블 알고리즘’(Ensemble algorithm)으로 구성됐다(Das et al., 2007). 구글 뉴스 체계에는 내용기반 필터링 방식까지 결합한 혼합 추천 알고리즘이 추가됐다(Liu et al., 2010). 구글 뉴스는 로그인 사용자 중 웹 로그(Web log) 기록 사용을 허가한 이용자들에게 개인화된 뉴스 추천을 제공한다. 이 이외에도 구글 뉴스는 이용자의 선호도를 기반으로 다양한 방법의 개인화된 뉴스 서비스를 진행하고 있다.

야후에서도 2008년부터 다양한 알고리즘을 사용하여 자사의 홈페이지에서 뉴스를 추천했다. 야후 뉴스 추천의 특징은 웹에서 생산되는 모든 뉴스가 아닌, 야후의 에디터가 품질 높은 뉴스들을 선정한 뒤, 이용자가 가장 좋아할만한 콘텐츠를 노출 영향도가 가장 높은 지역에 배치하는 것이다. 야후는 다양한 종류의 개인화 또는 비개인화 추천 알고리즘¹⁵⁾을

오프라인 시뮬레이션과 온라인 버킷 테스트(Bucket Test) 등을 통해 시험, 교체하면서 점진적인 알고리즘 개선 프로세스를 확립했다. 야후는 CORE의 약어로 불리는 콘텐츠 최적화 엔진(Content Optimization Relevance Engine) 기술을 통해 사용자에게 개인화 된 뉴스 추천 서비스를 제공하고 있다.

이 같은 과정에서 뉴스 추천 알고리즘의 성능을 평가하기 위해 다양한 지표가 개발됐다. Precision & Recall 곡선(Das et al., 2007), 클릭률(CTR)(Agarwal et al., 2009; Liu et al., 2010), 하루 평균 방문자 수(Liu et al., 2010), RMSE(Root Mean Squared Error)(Agarwal & Chen, 2009) 등 다양한 지표들이 사용됐다. 가장 많이 사용되는 지표는 ‘총 뉴스 클릭 수’다(Chu & Park, 2009; Chu et al., 2009; Das et al. 2007).

3. 루빅스 알고리즘에 대한 설명

1) 도입 배경과 루빅스의 도입 목표

모바일에서 상당수의 콘텐츠 서비스가 개인화되어 가는 방향으로 진화하고 있으며, 이용자들도 시간과 공간에 구애받지 않고 스스로 원하는 콘텐츠를 찾아 적극적으로 소비하는 방식으로 변하고 있다(김대원·윤영

15) 칼만 필터를 사용한 실시간 인기뉴스 추천 알고리즘(Agarwal et al. 2008), 이용자 클러스터 기반의 추천 알고리즘(Chu et al., 2009), 이용자와 뉴스의 특성들 사이의 선호도를 학습하는 Bilinear Regression 알고리즘(Chu & Park, 2009), Regression 기반의 행렬분할 모델(Agarwal & Chen 2009), 콘텍스트 정보를 활용한 콘텍스츠티일 멀티아드밴딧(Contextual Multi-Armed Bandit) 알고리즘(Li et al., 2010) 등이다.

민, 2015; 박현정, 2014). 또한 모바일 시대에서 페이스북과 같은 소셜 미디어 서비스들을 중심으로 한 개인화 된 뉴스 유통이 급속하게 증가하고 있다. 따라서 카카오는 이미 개인화 된 서비스에 익숙한 모바일 이용자들을 위해 뉴스 서비스도 개인화된 방식으로 변화를 시도하는 것이 타당하다고 판단했다.

개인화된 수요뿐만 아니라, 뉴스 서비스 제공자 입장에게는 넘쳐나는 정보의 효율적 관리도 중요한 과제가 됐다. 인터넷의 발달로 수많은 뉴스가 쏟아지지만 열람되지 못하는 뉴스가 무수히 많은 것이 현실이다. 카카오에게는 하나의 뉴스 서비스 화면에 20개 내외의 뉴스만 노출되는 공간 부족 문제를 해결할 수 있는 시스템이 필요했다. 카카오는 내부 논의를 거쳐 이용자의 뉴스 소비 패턴에 따라 뉴스의 노출과 배열이 효율적으로 결정되는 체계를 구축하기로 했다. 이러한 결정 하에서 개발된 뉴스 서비스용 추천 알고리즘이 루빅스다. 루빅스는 2015년 5월 시험 적용을 거쳐, 2015년 6월에 다음 모바일 서비스에 전면 도입됐다.

루빅스(RUBICS)는 'Real-time User Behavior Interactive Content recommender System'에서 주요한 단어의 각 머릿글자를 따서 만들어진 말이다. 루빅스는 "이용자의 행동에 대해 실시간으로 반응하여, 뉴스를 추천하는 체계"를 의미한다. 이용자 행동은 각 개인이 다음 모바일 뉴스를 이용하는 매개인 브라우저(Browser)를 근거로 식별된다. 여기서 행동이란 뉴스 소비 방식¹⁶⁾을 의미하며 이는 개인별, 또는 집단별로 분석될 수 있다. 예를 들어, 프로야구 뉴스를 주로 읽는 이용자라면 비슷한 행동 패턴(프로야구 뉴스를 자주 읽는)을 보이는 이용자들에게 인기있는 뉴스들을 추천받을 수 있다. 이는 이용자의 개인적인 특성에 따른 추천 결과이다. 행동 패턴 외에도 성별과 연령에 따라 이용자별로 추천받는 뉴스가 달라질 수 있다.

16) 소비 대상과 체류 시간 등이 포함된다.

실제 사례를 통해 확인해 보자. <그림 1>은 2016년 2월 26일 오후 1시 40분에 다음 모바일 뉴스를 통해 제공된 뉴스의 화면들이다. 동일한 시각임에도 이용자 A에게는 취업 뉴스, 이용자 B와 이용자 C에게는 각각 국제 뉴스와 아파트 시세 뉴스가 주요 뉴스로 제공됐다. 이용자 A, B, C는 임의로 선정된 표본이다. 이용자 별로 상이한 모바일 뉴스 화면 해당 사용자의 평소 뉴스 소비 패턴의 차이가 반영된 결과이다.



<그림 1> 루빅스에 의해 동일한 시간에 이용자마다 상이한 추천이 진행된 모습(자료: 카카오)

루빅스 개발에 앞서, 뉴스 서비스 기획자, 개발자, 그리고 데이터 사이언티스트(Data Scientist)들이 최우선적으로 집중한 과제는 서비스 주요 이용자들이 어떠한 특성을 갖고 있는지 파악하는 일이었다. 이용자

특성을 파악한 이후에야 최적화 된 알고리즘을 구축할 수 있기 때문이다. 루빅스 개발진은 이용자들이 누구이며, 어떠한 소비 패턴을 나타내는지 우선 분석했다. 그 결과, 다음의 모바일 서비스 이용자들 상당수가 뉴스가 아닌 다른 수요를 충족시키기 위해¹⁷⁾ 다음 앱 혹은 웹페이지를 방문하며, 뉴스를 소비하는 이용자 대부분은 로그인을 하지 않는 것으로 확인됐다. 즉, 추천 서비스를 제공할 대부분의 이용자들은 뉴스 서비스를 자주 사용하는 열성적인 이용자(Heavy User)가 아니라 이용자 행동 정보가 거의 없는 ‘콜드 스타트 이용자’인 것이다.

이러한 이용자에 대해서는 개인화 추천의 대표적인 방법인 협업 필터링 혹은 내용 기반 필터링으로 개인화 추천을 제공하기 어렵다. 이에 따라, 카카오는 전술한 두 방법 대신에 통계적 기계학습 기법인 ‘맞춤형 멀티암드밴딧’(Customized Multi-Armed Bandit) 알고리즘을 추천 시스템에 적용했다. 여기서 ‘맞춤형’이란 말은 카카오의 뉴스 서비스에 맞게 알고리즘이 수정됐다는 뜻이다.

2) 멀티암드밴딧 알고리즘

2016년 10월 현재 루빅스는 여러 가지 추천 알고리즘을 복합적으로 사용하고 있지만, 초창기의 주요 알고리즘은 멀티암드밴딧 방식이었다. 멀티암드밴딧의 어원은 ‘슬롯 머신’(Slot Machine)에서 비롯됐다. 슬롯 머신은 외팔이 강도(One-Armed Bandit)라는 속어로 불린다. 이 말은 베팅을 위해 위아래로 내리는 손잡이(Lever)가 기계에 붙어 있는 모습이 흡사 팔 하나를 가진 ‘강도’(強盜)를 연상하게 한다는 뜻에서 비롯됐다. 강도라는 말은 손님의 돈을 ‘뺏어가는’ 슬롯 머신의 속성 때문에 붙여

17) 다음 검색, 메일, 카페 등의 다른 다음서비스를 사용하기 위해 많은 이용자들이 다음 모바일 홈페이지를 방문한다.

진 것으로 추정된다. 멀티암드밴딧은 레버가 여러 개 달린 슬롯 머신 또는 한 줄로 늘어선 여러 개의 외팔이 슬롯 머신을 의미한다.

멀티암드밴딧의 개념은 미국 라스베이거스에서 도박을 하는 과정으로 설명될 수 있다. 여러 개의 슬롯 머신으로 도박을 하면서 이튿날 아침까지 가장 돈을 많이 따는 사람이 이기는 게임을 한다고 가정해보자. 각 슬롯 머신의 이길 확률은 사전에 정해져 있지만, 베팅을 하는 사람들은 그것을 알 도리가 없다.

이러한 상황에서 어떠한 전략으로 게임을 해야 가장 많은 돈을 딸 수 있을까? 이길 확률이 높은 슬롯 머신을 찾아 집중적으로 해당 머신에 베팅하는 것은 실효적인 전략이 될 수 있다. 이를 위해 각 슬롯 머신에서 승률을 확인하는 과정을 탐색(Exploration)이라고 한다면, 가장 승률 높은 슬롯 머신에서 집중적으로 돈을 따는 것을 수확(Exploitation)이라 볼 수 있다. 또 아침까지 딸 것으로 예상되는 돈을 기대보상(Expected Discounted Reward)이라고 할 수 있다.

관건은 ‘얼마나 빨리’, 즉 최소한의 테스트로 이길 확률이 가장 높은 슬롯 머신을 찾아내는 것이다. 탐색의 횟수를 줄이고, 수확에 집중하는 것이 수익을 극대화하는 방법이다. 멀티암드밴딧 방식 중 하나인 입실론-그리디(Epsilon-Greedy) 방식은 탐색과 수확 비율을 고정해 놓은 상태에서 승률을 반복해서 측정한다. 입실론-그리디 방식은 각 슬롯 머신 대상의 테스트를 충분히 수행한 이후에도 테스트를 게임이 완료될 때까지 지속하는 특성이 있는데, 이는 필요 이상으로 수확의 기회를 낭비하게 되어 기대보상이 감소되는 결과를 초래한다. 탐색 비율이 낮아 승률을 얻는 데 학습 시간이 길어지는 것도 문제지만, 탐색 비율이 높아지는 과잉 탐색도 기대보상을 감소시킨다.

입실론-그리디 알고리즘의 약점을 극복하기 위해 적용되는 것이 소프트맥스(Softmax)나 UCB(Upper Confidence Bound) 방식이다. 소

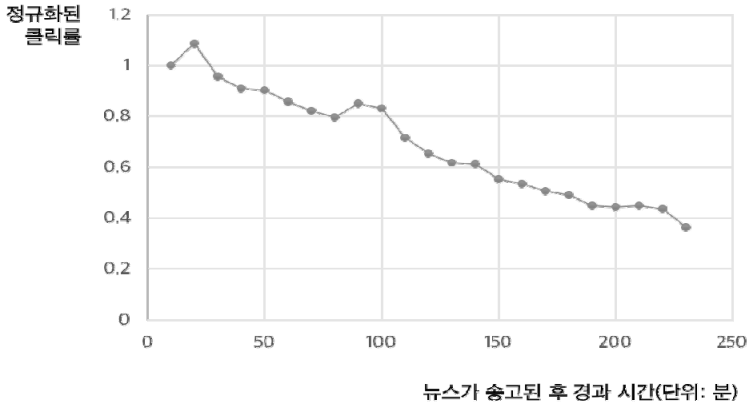
소프트맥스 방식은 기대보상을 극대화하기 위해 승률이 높은 슬롯 머신에 집중한다. 이 알고리즘은 승률이 낮은 것으로 확인된 슬롯 머신에는 더 이상의 탐색이 불필요하다는 전제 하에서 작동된다. 그러나 조사 대상표본 수가 충분치 않은 초기 단계에서는 승률 자체가 부정확하게 측정될 수 있다. 이러한 통계적 부정확성으로 인해 승률이 높은 슬롯 머신을 수확의 대상에서 배제하여 기대보상을 낮추는 결과가 초래될 수 있다. 이를 해결하기 위해 UCB가 고안됐다. UCB는 검증이 덜 된 슬롯 머신들에 더해 승률이 높은 슬롯 머신에 더 많은 검증 기회를 부여하는 구조다. UCB는 각 뉴스들이 선택될 확률을 뉴스의 클릭률에 통계적 불확실성을 더한 값으로 계산한다. 통계적 불확실성은 해당 뉴스를 테스트 한 숫자에 반비례한다. 이러한 구조를 통해, UCB는 소프트맥스처럼 클릭률이 높은 뉴스에 집중함으로써 테스트의 낭비를 줄이는 동시에, 통계적 불확실성으로 인해 탈락되는 뉴스에도 동시에 많은 테스트 기회를 부여하는 효과를 발휘한다. 요컨대, 입실론-그리디의 단점을 보완한 것이 소프트맥스이고, 소프트맥스의 단점을 보완한 알고리즘이 UCB 인 것이다.

일반적으로 앞의 세 가지 알고리즘과 함께 수확을 늘릴 수 있는 방법으로 담금질(Annealing)이 병행되기도 한다. 담금질은 이미 충분히 슬롯 머신을 검증했으면, 그에 반비례하여 탐색 횟수를 줄이는 것이다. 탐색 횟수가 줄어들면 수확의 비율이 늘어 기대보상이 증가되는 효과가 기대된다. 즉, 담금질은 불필요한 탐색을 줄임으로써 기대보상을 극대화하는 방법인 것이다.

3) 맞춤형 멀티암드밴딧 알고리즘

이번 장에서는 멀티암드밴딧 알고리즘이 카카오의 뉴스 제공 환경에 맞게 수정된 이유와 과정을 설명하고자 한다.

(1) 변화하는 클릭률



〈그림 2〉 시간의 경과에 따른 뉴스 클릭률의 변화 추이(자료: 카카오)

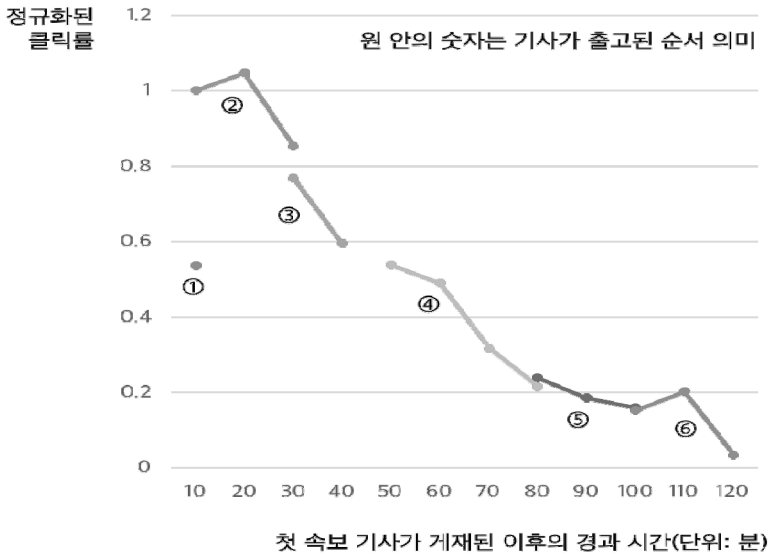
슬롯 머신 사례로 설명했던 멀티암드밴딧의 주요 요소를 뉴스 서비스에 적용시켜 보자. 각각의 슬롯 머신은 개별 뉴스를 의미하고, 슬롯 머신의 승률은 뉴스가 클릭 될 확률(Click Through Rate, CTR)에 해당된다. 기대보상은 총 뉴스 클릭수로 나타난다. 일반적으로 멀티암드밴딧 알고리즘에서 각 슬롯 머신의 승률은 고정값이다. 그러나 뉴스의 클릭률, 즉 CTR은 시간에 따라 변한다. 일반적으로 뉴스는 시간이 경과할수록 덜 소비되는 경향이 있다. 이러한 경향성은 〈그림 2〉를 통해 간명하게 확인된다. 〈그림 2〉는 임의로 선택된 뉴스의 시간의 경과에 따른 클릭률의 변동 추이를 보여준다. 가로 축은 해당 뉴스가 카카오 뉴스 시스템에 들어온 시점부터 경과된 시간(분)이며, 세로 축은 10분 간 해당 뉴스의 클릭률¹⁸⁾이다.

18) 본 논문에서 사용된 클릭률은 각 경우마다 정규화(Normalized) 과정을 거친 수치이

속보성 뉴스는 1시간만 지나도 소비 대상으로서의 매력이 현저히 떨어진다. 시간이 경과하는 동안에 소비자가 카카오 뉴스 서비스 외 TV나 신문, 온라인 신문 등 다른 매체에서 이미 동일하거나 비슷한 뉴스를 접했을 가능성이 높기 때문이다. <그림 3>은 속보성 뉴스의 이러한 특성을 잘 보여준다.¹⁹⁾ <그림 3>의 그래프는 2015년 2월 11일 서울 동작구 사당동 사당종합체육관 신축 공사장 사고 기사와 관련 속보 기사의 클릭률 변화 추이다. Y축의 클릭률은 2번째 기사의 첫 10분 간의 클릭률에 대한 비율로 정규화한 클릭률이며, X축은 사건 발생 이후 경과 시간이다. 이처럼 동일한 뉴스라 하더라도 해당 뉴스의 클릭률은 시간의 경과에 따라 끊임없이 변하기에, 적절한 추천을 위해서 알고리즘이 뉴스의 클릭률을 지속적으로 측정해야 한다. 이는 앞서 살펴 본 소프트맥스나 UCB 알고리즘 방식이 뉴스 서비스 추천에서는 효과적으로 작동하지 않을 수 있음을 의미한다.

다. 클릭률의 실제 수치는 기업의 영업 비밀에 해당하기에, 공중을 대상으로는 공개가 어렵다. 정규화된 값은 변화 양상을 보다 잘 드러내주기에 경향성을 파악하는 데 효과적인 장점을 내재하고 있기도 하다.

- 19) 첫 속보 기사의 클릭률이 2번째 기사보다 작은 것은 두 가지 이유 때문으로 해석된다. 첫째, 첫 속보 기사는 사건 발생 직후 기사의 제목만 있고 내용이 없는 속보성 기사였기 때문에 풀이된다. 여타 기사 대비 빨리 송고됐지만 제목으로만 구성된 기사가 후속 기사에 비해 클릭률이 낮은 것은 흔히 일어나는 현상이다. 첫 보도 이후 사건이 공중(公衆)에 인지되면서, 뉴스 소비자들이 보다 자세한 내용을 습득하기 위한 욕구를 충족시키기 위해 뉴스 서비스 제공업체에 방문하는 것도 2번째 속보 기사의 클릭이 급증한 이유로 해석될 수 있다.



〈그림 3〉 시간의 경과에 따른 속보성 기사의 클릭률 추이(자료: 카카오)

새로운 슬롯 머신이 끊임없이 추가되고, 동시에 기존의 슬롯 머신이 지속적으로 사라지는 특징 역시 뉴스 서비스가 전통적인 멀티암드밴딧 구조와 다른 점이다. 뉴스 서비스에서는 새로운 뉴스가 시시각각 유입되고 오래된 뉴스들은 서비스 대상에서 지속적으로 배제된다.

이러한 환경에서 담금질 전략은 끊임없이 초기화되기 때문에 효율성이 떨어진다. 담금질은 전체 탐색 횟수에 반비례해서 탐색 비율을 줄이는 전략이다. 즉, 탐색이 충분히 이뤄졌다면, 추가적인 탐색 횟수를 줄이고 수확(Exploitation)의 비율을 늘려서, 기대보상을 증대할 수 있다. 그러나 새로운 뉴스가 들어오면 전체 탐색횟수가 영(零)으로 초기화 되며, 다시 충분한 수준의 탐색이 추가로 필요하게 된다. 결과적으로 뉴스 서비스의 경우, 지속적으로 뉴스가 새로 인입되는 특성으로 인해, 담금질

\전략을 적용하기 어렵게 된다

실시간으로 변하는 뉴스의 클릭률(CTR)을 정확하게 측정하기 위해, 루빅스는 매 분 단위로 각 뉴스의 클릭률을 측정한다. 다만 1분이라는 단위 시간이 상대적으로 짧다보니, 표본의 부족으로 인해 클릭률이 부정확하게 측정될 수도 있다. 10번의 노출 중 1번 클릭되어 도출된 10%의 클릭률과 1만 번 노출하여 1,000번 클릭된 결과로 계산된 클릭률 10%가 동일하게 취급될 수 없다. 이 경우, 전자의 클릭률에 대한 신뢰도는 후자에 비해 낮게 평가될 수밖에 없다. 이러한 통계적 부정확성은 외부 요인에 의해 발생하기도 한다. 이용자들이 관심을 가지는 대형 속보가 이어지거나, 언론사에서 카카오 뉴스 서비스로 송고해주는 뉴스가 특정 시간대에 집중되는²⁰⁾ 구조로 인해, 측정 대상 뉴스 숫자가 급격히 늘면서 측정의 정확도가 떨어질 수 있다.

이와 같은 통계적 부정확성을 극복하고 정확한 실시간 클릭률을 측정하기 위해서, 루빅스에는 이동 평균(Moving Average) 알고리즘이 사용됐다. 이 방법은 분 단위 측정을 하면서 과거 데이터까지 모두 사용하는 동시에 최신 데이터일수록 더 많은 가중치를 줘, 이용자의 최신 뉴스 소비 패턴을 반영함으로써 클릭률 측정의 신뢰도를 높일 수 있다. 알고리즘에 활용된 주요 수식은 다음과 같다.

$$imp_{i,n} = v_{i,n} + \delta \times imp_{i,n-1} \quad \text{<수식 1>}$$

$$click_{i,n} = c_{i,n} + \delta \times click_{i,n-1} \quad \text{<수식 2>}$$

$$CTR_{i,n} = \frac{click_{i,n}}{imp_{i,n}} \quad \text{<수식 3>}$$

20) 정오를 중심으로는 석간 신문의 뉴스가 물리고, 새벽과 아침에는 조간 신문의 뉴스가 집중적으로 유입된다.

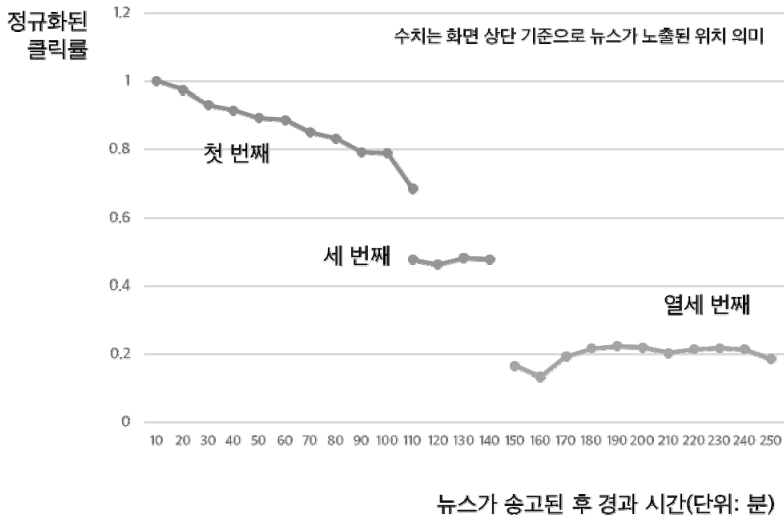
$v_{i,n}$ 는 뉴스 i 의 $n-1$ 분과 n 분 사이의 노출(Impression) 수이고, $c_{i,n}$ 는 뉴스 i 의 $n-1$ 분과 n 분 사이의 클릭수이다. δ 은 과거 데이터의 가중치를 의미하는데, 그 크기는 일반적으로 1보다 작다. $imp_{i,n}$ 는 뉴스 i 의 n 분까지의 노출 수의 가중합(Weighted summation)이며, $click_{i,n}$ 는 뉴스 i 의 n 분까지의 클릭 가중합이다. $CTR_{i,n}$ 은 뉴스 i 의 n 분에서의 시간 가중치가 반영된 클릭률로 그 시점까지의 클릭 가중합과 노출 수의 가중합의 비율로서 계산된다.

(2) 뉴스 위치에 따른 클릭률 변화

전통적인 멀티암드밴딧 알고리즘에서는 수확을 위해 하나의 슬롯 머신만 선택한다. 하지만 뉴스 추천에서는 이용자에게 복수(複數)의 뉴스를 동시에 추천한다. 구체적으로 루빅스는 이용자에게 20개 이상의 뉴스를 한꺼번에 제공한다. 동시에 여러 개의 뉴스가 추천되다 보니, 뉴스는 화면 내 각기 다른 위치에 배치된다. 온라인에서 제공되는 뉴스는 서비스 화면 내의 뉴스 위치에 따라 클릭률이 변한다. 이를 위치 편향(Positional Bias)이라고 일컫는다(Agarwal et al., 2009).

예를 들어 같은 뉴스라도 서비스 화면 최상단에 배치되었을 때, 다른 영역에 배치되었을 때보다 클릭률이 높다. 위치변수로 인해 뉴스 간 클릭률을 단순 비교하기 어렵다. 만약 서비스 화면 최상단에 노출된 뉴스 A 클릭률이 10%라고 가정하고, 가장 아래 노출된 뉴스 B 클릭률이 5%로 나타났다고 할 때, A가 B보다 나은 뉴스라고 확신할 수 없다. <그림 4>는 동일한 뉴스가 다른 위치에 배치되었을 때 클릭률의 변화를 보여준다.²¹⁾

21) <그림 4>는 화면 노출 위치에 따른 뉴스 클릭률 변화를 보여주기 위해 선정된 임의의 사례이다. <그림 4>의 Y축은 하나의 기사가 첫 번째 위치에서, 세 번째, 그리고 열 세번째 위치로 옮겨 가면서 측정된 값이다. 뉴스의 게시 위치가 변동하면 기존의 위치



〈그림 4〉 모바일 화면의 노출 위치에 따른 뉴스 클릭률 변화(자료: 카카오)

이러한 위치 편향 현상을 해소하기 위해서 루빅스는 ‘이중의 탐색’을 실행한다. 이중의 탐색을 이용해서 클릭률을 계산하면 평균 위치에서의 기사 클릭률이 계산된다. N개의 위치가 있다면 모든 위치에서 $1/N \times$ 총 탐색 수만큼 동일하게 검증이 이루어지고 이들의 평균으로 클릭률로 계산되는 것이다. 이 과정을 거치면서 위치 편향은 소멸된다. 위치 편향의 해소로 뉴스 간 클릭률의 정확한 비교가 가능해진 것이다. 이는 이중의 탐색과 전체 데이터가 아닌 ‘이중의 탐색’ 데이터만으로 뉴스 클릭률의 측정한 데 따른 결과이다. 이 결과로 나온 ‘평균 위치에서의 기사 클릭률’은 기사 자체가 얼마나 선호되는 가를 의미하게 되며, 이를 기사의 ‘선호도’로 정의한다.

에서는 더 이상 노출이 이뤄지지 않기 때문에, 기존의 위치에서 클릭률 측정은 불가능하다. 대신, 게시 위치가 바뀐 뒤에는 새로운 위치에서의 클릭률이 계산된다.

당초 루빅스 개발진은 위치 편향 문제를 풀기 위해 뉴스 위치 가중치를 기반으로 뉴스 클릭률의 정확도를 높이는 방법을 선택했다. 최상단에 배치된 뉴스 A가 가장 아래 배치된 뉴스 B보다 클릭률이 평균 30% 증가하는지, 혹은 60% 증가하는지 여부를 놓고 위치 가중치 값을 결정하려는 시도였다. 그러나 위치와 관련한 클릭률 변화는 뉴스의 품질²²⁾ 그리고 함께 제공되는 다른 뉴스들의 클릭률 성과²³⁾에도 영향을 받는다. 뉴스 위치에 대해 가중치를 부여하는 방식은 다양하게 시도됐으나 유의미한 성과로 이어지지 않았다.

(3) 이용자 집단별 최적화

전체 이용자를 대상으로 뉴스의 클릭률을 계산하기보다, 이용자와 같은 성별 그리고 연령대의 이용자 집단에서 소비된 뉴스의 클릭률을 측정하는 것이 더욱 높은 정확성을 확보할 수 있는 방법이다. 루빅스의 맞춤형 멀티암드밴딩 방식은 각 이용자 집단(성별이나 연령대)의 뉴스 소비 패턴에 근거하여 해당 집단에 속한 개인에게 뉴스를 추천한다. 예를 들어, 20대 여성에게는 뷰티 및 취업, 30대 여성에게는 육아, 40대 여성에게는 교육, 50대 여성에게는 건강 관련 뉴스를 더욱 많이 소비하는 경향이 나타나며, 해당 집단에 속한 이용자의 추천에는 집단별 분석에서 확인된 결과가 반영된다. 스포츠 영역에서는 이용자들의 뉴스 소비 패턴에 기반을 두어 이용자를 따로 분류하는 ‘클러스터링(Clustering)’ 방식을 적용했다. 이 과정에서 최적의 ‘클러스터 개수’²⁴⁾를 얻기 위해 다양한 검

22) 뉴스 자체에 대한 선호도로 클릭률이 높아지면 뉴스 위치의 영향력은 둔화된다.

23) 동시에 추천되는 뉴스들 중 이용자에게 노출이 용이한 상단에 위치되는 뉴스들의 클릭률이 떨어지면, 상단 뉴스들의 클릭률이 높을 때보다 위치 편향으로 인한 클릭률 감소가 심화된다.

24) 클러스터의 개수는 가장 우수한 성능을 제공하는 수로 실험을 통하여 결정된다. 성능의 평가 지표는 총 클릭 수이다.

증을 거듭한 결과, 스포츠 뉴스 이용자들은 국내 프로야구, 미국 메이저 리그, 축구 등 3가지 클러스터로 분류됐다. 루빅스는 해당 클러스터에 속한 이용자에게 국내 프로야구, 미국 메이저리그, 또는 축구 뉴스를 상대적으로 많이 추천했다.

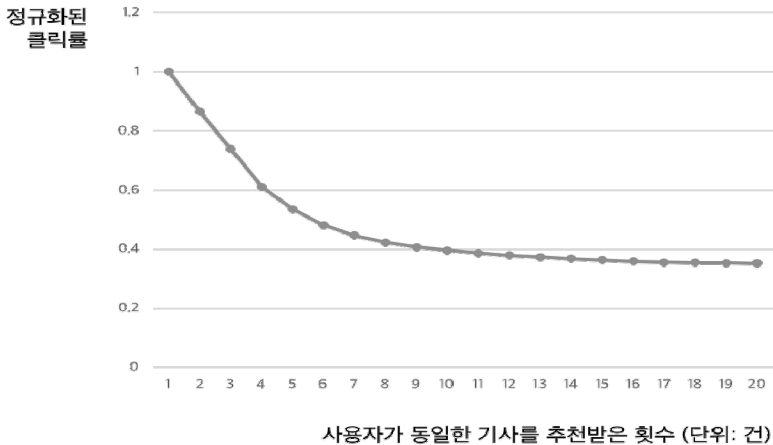
(4) 추천 결과 노출 방법

앞선 장에서는 각 이용자 집단에서 각 기사의 '선호도'를 보다 정확하게 측정하기 위한 방법을 설명했다. 이번 장에서는 측정된 선호도를 이용해 기사를 어떻게 노출할 것인가에 대해 알아 보도록 한다.

일반적으로 선호도가 높은 기사가 되도록 상위 위치에 배치되는 것이 뉴스 사용자의 효용을 극대화할 수 있다. 따라서 루빅스는 기본적으로, 이용자 집단을 알 수 있는 사용자의 경우, 해당 사용자가 속한 이용자집단에서 선호도가 높은 기사 순으로 배열한다. 이용자집단을 알 수 없는 사용자의 경우는, 전체 이용자의 선호도가 높은 기사 순으로 배열한다.

또한, 뉴스 서비스 이용자 만족도를 개선하기 위해서는 맞춤형으로 추천되는 뉴스의 다양성을 높이는 방안도 모색되어야 했다. 뉴스 서비스 화면에 계속 동일한 기사만 보일 경우, 서비스 만족도가 떨어질 수밖에 없기 때문이다. 동시에 추천되는 뉴스가 지나치게 자주 변경되면, 이용자 혼란을 가져올 수 있다. 루빅스 개발진은 이와 같은 문제를 함께 해결하면서 뉴스 다양성을 제고하는 방안으로 알고리즘을 개선해야 했다.

루빅스 적용 초기의 이용자 뉴스 소비 패턴을 분석한 결과, 이미 한번 읽은 뉴스를 다시 클릭하는 이용자는 극히 적었다. 이에 따라, 루빅스 개발진은 이미 읽은 뉴스를 추천 화면에서 제외하고 새로운 뉴스로 대체함으로써 이용자에게 더 다양한 뉴스들을 제공했다. 이와 함께 여러 가지 뉴스를 연속적으로 소비하는 이용자에게는 너무 잦은 뉴스 서비스 화면 변경에 혼란을 겪지 않도록 일정 시간 추천 뉴스 배치를 바꾸지 않도록 했다.



〈그림 5〉 동일한 뉴스의 반복적 추천에 따른 클릭률 추이(자료: 카카오)

루빅스 개발진은 지속적으로 뉴스를 소비하는 이용자에게는 이미 추천된 뉴스들이 반복적으로 추천되는 문제에도 주목했다. 반복적으로 추천해도 이용자가 반응하지 않는 뉴스는 해당 이용자가 관심이 없는 뉴스이거나, 이미 다른 곳에서 본 뉴스일 가능성이 높기 때문이다.

〈그림 5〉는 같은 뉴스를 반복적으로 이용자에게 노출했을 때, 뉴스의 클릭률이 떨어지는 현상을 보여준다. 〈그림 5〉는 다음 모바일 뉴스 서비스로 제공된 기사의 클릭률 평균치를 근거로 시각화 한 결과이다. 루빅스는 이용자가 반복된 추천에도 반응하지 않는 것을 이용자의 뉴스에 대한 부정적인 ‘피드백(Feedback)’으로 간주한다. 루빅스는 특정 이용자에게 여러 번 추천된 뉴스는 이용자의 클릭 여부와 상관없이 ‘패널티(Penalty)’를 부여하여, 해당 이용자에게 다른 뉴스를 볼 수 있는 기회를 제공한다. 이 알고리즘은 뉴스 서비스를 빈번하게 사용하는 이용자에게만 적용된다.

4) 루빅스 성능 평가 방법

루빅스에 활용되는 알고리즘은 지속적으로 테스트를 거치고 이용자 서비스 만족도, 즉 뉴스 추천 품질을 높이기 위해 끊임없이 교체된다.²⁵⁾ 루빅스 운영진은 두 가지의 테스트를 통해 새로운 알고리즘의 루빅스 적용 여부를 결정한다.

첫 번째는 오프라인 시뮬레이션(Offline simulation)이다. 오프라인 시뮬레이션에서는 이용자들의 뉴스 소비 로그 데이터(Log data)를 특정 시간을 기준으로 훈련 데이터와 검증 데이터로 분리한다.²⁶⁾ 훈련데이터는 기계학습 알고리즘을 훈련시키기 위한 데이터이다. 이후 훈련 데이터와 검증 데이터에서 적어도 한 번씩 뉴스를 읽은 이용자를 테스트 이용자로 추출한다. 훈련 데이터에 사용자가 클릭한 뉴스가 없으면 사용자가 무엇을 좋아하는지 파악할 수 없고, 검증데이터에 사용자가 클릭한 뉴스가 없으면 추천이 제대로 되었는지 알 수 없다. 새로운 알고리즘에게 훈련 데이터를 기반으로 학습한 이후, 검증데이터에 포함된 뉴스를 추천하도록 한다. 이 결과와 테스트 이용자가 검증 데이터 내에서 실제 읽은 뉴스와 얼마나 겹쳐지는 지(추천 성공률, Hit ratio)를 파악한다.²⁷⁾ 평가 기준인 검증 데이터는 이용자가 뉴스를 소비한 시간에 제공

25) 보통 인터넷 서비스는 현재 버전(version)의 기능을 끊임없이 개선하기 보다는 새로운 기능을 추가한 뒤 일정 시간 이후에 발전된 다음 버전을 제공하는 방식으로 운용된다. 알고리즘의 교체 주기 역시 이와 같다. 그러나 루빅스 개발진은 새로운 알고리즘을 지속적으로 테스트하고 이용자 서비스 만족도, 즉 뉴스 추천 품질이 높은 알고리즘으로 루빅스를 끊임없이 대체하고 있다. 이는 품질의 지속적인 개선과 새로운 기능 제공을 통해 소비자의 편익을 높이려는 전략적 선택이 반영된 결과이다.

26) 정교한 추천 모델을 생성하기 위해서는 훈련데이터의 양이 충분해야 한다. 검증 데이터는 알고리즘에게 제공하지 않는 데이터로 알고리즘이 추천한 결과를 검증하기 위해 사용된다.

27) 예를 들어, 뉴스 사용자의 7월과 8월 로그 데이터를 가지고 뉴스 알고리즘을 훈련시켜 9월 뉴스를 대상으로 각 사용자에게 추천을 제공하고 추천된 뉴스 중에 각 사용자

된 뉴스로 한정된다. 새로운 알고리즘의 성능은 추천 성공률로 측정된다. 추천 성공률 측정을 위한 수식은 다음과 같다.

$$\text{추천 성공률} = \frac{\text{이용자가 클릭한 뉴스가 추천된 횟수}}{\text{추천 횟수}} \quad \text{<수식 4>}$$

오프라인 시뮬레이션은 새로운 알고리즘에 있을 수 있는 문제점들을 최소화하고 성능을 최적화하기 위한 알고리즘 튜닝, 엔지니어링 리소스(Engineering resource)의 효과적인 사용,²⁸⁾ 그리고 새로 시스템에 적용할 알고리즘이 현재 사용하는 알고리즘보다 우수한 성능을 보이는 것을 확인하기 위한 목적에서 수행된다. 오프라인 시뮬레이션에서는 현재 알고리즘보다 더 나은 성능을 보유한 것으로 평가된 알고리즘에 대해 온라인 버킷 테스트(Online bucket test)가 실시된다. 온라인 버킷 테스트는 임의적으로 선발한 일부의 실제 이용자에게 새로운 알고리즘을 기반으로 한 추천의 결과를 제공하여, 새 알고리즘을 시험해 보는 것이다.

버킷 테스트를 통해 검증되는²⁹⁾ 품질 평가 지표는 세 가지다. 첫 번째 품질 평가 지표는 야후의 뉴스 추천 품질 평가에 사용되는 지표인 ‘예상되는 하루 총 클릭 수’(Expected daily total clicks)(Chu and Park, 2009; Chu et al., 2009; Das et al. 2007)이다. 새로 테스트되는 알고리즘은 기존 알고리즘에 비해 적은 이용자에게만 노출되기에 동일한 이용자에게 노출되었을 때의 예상되는 하루 총 클릭수를 계산할

가 실제 클릭한 뉴스(검증 데이터)가 있는 지 여부로 추천 성공률이 측정되는 것이다.
 28) 구현할 알고리즘이 현재 알고리즘보다 성능이 우수한 지 아닌지를 미리 테스트 할 수 없다면 엔지니어들은 예상 가능한 모든 알고리즘을 구현해야 할 것이다. 오프라인 시뮬레이션을 통해 성능이 나쁜 알고리즘들을 걸러내고 성능이 좋은 알고리즘을 채택함으로써 엔지니어들의 시간을 효율적으로 사용할 수 있다.
 29) 버킷 테스트의 표본 규모는 일반적으로 모수의 5% 이하이다.

필요가 있다. 이는 실제 클릭 수에 총 이용자 수를 버킷 테스트 참여에 이용자 수로 나눈 값을 곱해서 구해진다.³⁰⁾

총 클릭수로만 알고리즘의 성능을 평가할 때 문제가 되는 점은 클릭 어뷰징(Click abusing)³¹⁾에 취약하다는 것이다.

이에 대한 대안으로 “예상되는 하루 총 체류 시간’(Expected daily total duration time)을 또 다른 품질 평가 지표로 도입했다. 클릭 어뷰징에 유도돼 뉴스를 소비한 이용자가 해당 뉴스에 체류하는 시간은 극히 적다. 이러한 특성 때문에 일정 시간 이상의 체류 시간이 발생한 클릭만을 사용하는 것은 클릭 어뷰징을 걸러낼 수 있는 효과적인 방법이 될 수 있다. 하루 총 체류 시간은 코리안 클릭(Korean Clicks) 등의 온라인 이용자 측정 기관에서 유용한 지표(Key performance indicator)로서 활용되고 있다.

세 번째의 품질 평가 지표는 ‘예상되는 하루 총 뉴스 이용자 수’(Expected daily unique visitor)이다. 예상되는 하루 총 뉴스 이용자 수는 뉴스 소비 이용자(하루에 적어도 하나의 뉴스를 읽은 이용자)의 규모를 의미하기에, 이 지표의 증가는 추천 서비스의 성패를 측정하는 데 유의미하다. 버킷 테스트에서 새로운 알고리즘이 현재의 알고리즘보다 성능이 좋을 경우, 점진적으로 루비스 시스템은 새로운 알고리즘으로 교체된다. 알고리즘에 의해 자동화 된 시스템에 대한 우려 중 하나는 시스템에 문제가 생겨도 그대로 운영되어 이용자의 편익을 훼손할 수 있다는 것이다(Introna, 2016). 알고리즘에 대한 지속적인 평가와 그에 따른 개선은 앞서의 우려와 같은 문제가 발생할 개연성을 원천적으로 제거하는 역할을 하는 셈이다.

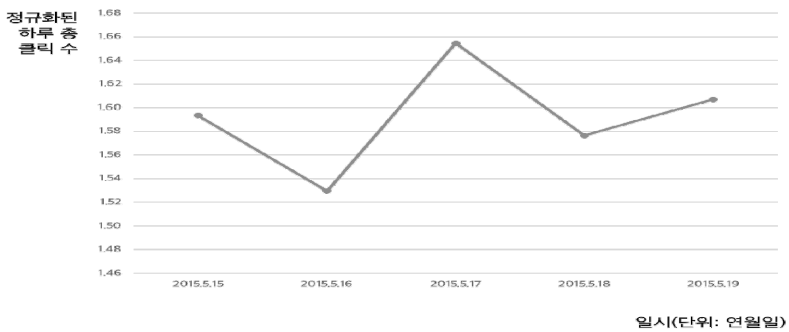
30) 버킷 테스트의 표본 규모가 모수 5%라면 예상 클릭 수는 버킷 클릭 수 $\times (1/0.05)$ 로 계산된다.

31) 충격, 경악 등 말초적인 단어를 뉴스 제목에 넣거나 선정적인 사진으로 사용자의 클릭을 유도하는 행위를 일컫는다.

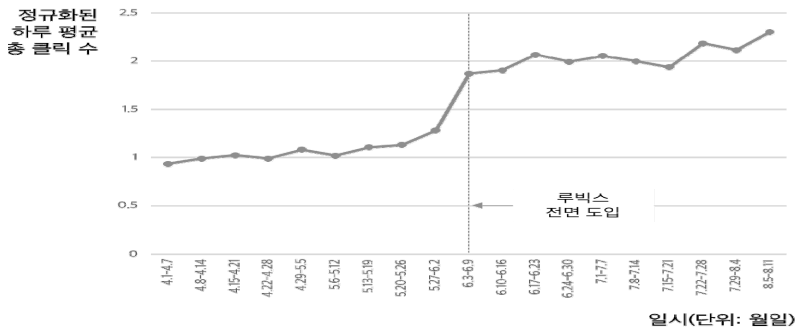
4. 루빅스 도입에 따른 효과

1) 하루 총 클릭 수

루빅스의 품질 평가 지표인 하루 총 클릭 수의 성과부터 확인해 보자.



〈그림 6〉 루빅스 도입 이전에 버킷 테스트 단계의 하루 총 클릭 수(자료: 카카오)



〈그림 7〉 루빅스 전면 도입 전의 하루 평균 총 클릭 수 추이(자료: 카카오)

〈그림 6〉은 루빅스가 도입되기 한 달 전인 2015년 5월 당시 뉴스 서비스 체계와 루빅스 알고리즘의 성능(루빅스를 대상으로 한 버킷 테스트 결과)을 일별로 비교한 결과이다. 〈그림 6〉에서 Y축의 값은 기존 뉴스 서비스 체계에 비한 루빅스의 성능 개선 정도를 나타낸다. Y값이 1.0보다 크면 루빅스 알고리즘이, 반대로 1.0보다 작으면 기존의 체계가 루빅스 알고리즘에 비해 우월함을 의미한다. 성능 지표로는 예상되는 하루 총 클릭수가 사용됐다. 〈그림 6〉을 통해 확인할 수 있듯이, 버킷 도입 테스트 단계임에도³²⁾ 루빅스 알고리즘 도입 이후의 하루 총 클릭 수는 이전의 서비스 시스템이 적용됐을 때에 비해 평균 60% 증가했다.

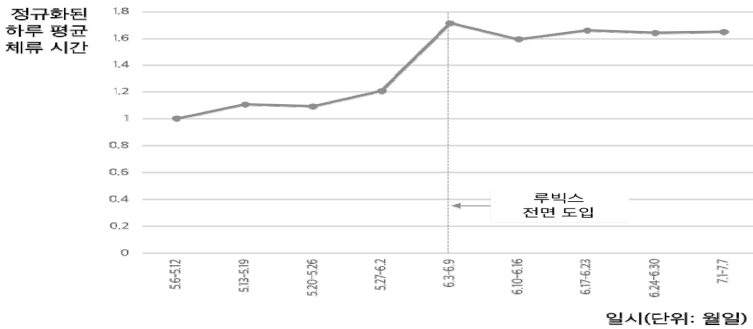
〈그림 7〉은 루빅스가 도입되기 전인 2015년 4월 첫 주부터 루빅스가 전면 도입(2015년 6월)되고 두 달이 흐른 2015년 8월 두 번째 주까지의 발생한 다음 모바일 뉴스 서비스의 “주간 단위 하루 평균 총 클릭 수”를 보여 준다. Y축은 2015년 4월 하루 평균 총 클릭 수 대비 증가 비율을 보여준다. 〈그림 7〉에서 볼 수 있듯이, 루빅스가 전면 도입된 2015년 6월은 다음 모바일 뉴스의 하루 총 클릭 수의 변곡점이 된다. 루빅스 도입으로 종전 체계에 비해 하루 평균 클릭 수가 2배 수준에 이른다. 2015년 8월 두 번째 주의 하루 평균 총 클릭 수는 루빅스 도입 전인 2015년 4월에 비해 130% 증가했다.

2) 이용자 총 체류 시간

클릭 수만이 아니라, 뉴스 소비의 질로 평가될 수 있는 체류 시간에서도 루빅스 효과가 확인된다. 〈그림 8〉은 2015년 5월 이후부터 7월 초까지의 다음 모바일 뉴스 서비스에서 이용자 총 체류 시간의 주간 단위 하

32) 2015년 5월 버킷 테스트 당시의 다음 모바일 뉴스 서비스에서 루빅스의 적용 폭은 5%였으며, 이 범위는 30%, 그리고 6월 100%(전면 적용)으로 점진적으로 확대됐다.

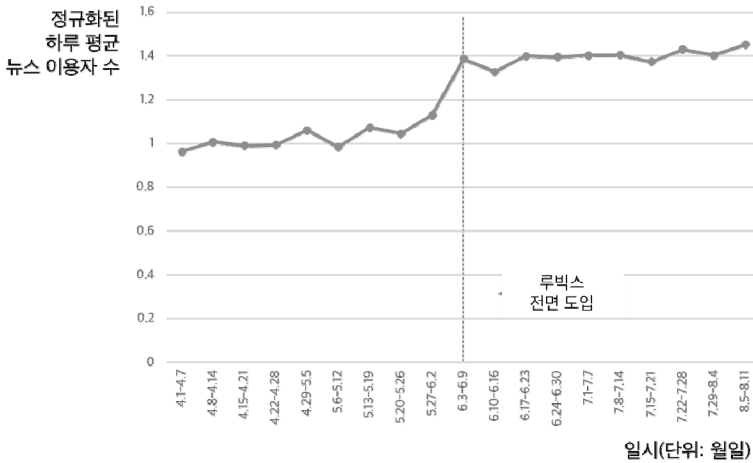
루 평균이 어떻게 변화하였는가를 보여준다. 체류 시간은 5월부터 측정을 시작해서 서비스 적용 전인 4월과의 직접적인 비교는 불가능하지만 큰 틀에서 루빅스의 성과를 보여주기에 부족함이 없고 판단된다.



〈그림 8〉 루빅스 전면 도입 전후의 하루 평균 총 체류 시간 추이(자료: 카카오)

〈그림 8〉에서 보듯이 다음 모바일 서비스 이용자들이 뉴스 서비스에서 체류하는 시간 역시 루빅스가 적용된 이후에 한 단계 상승하는 모습을 보였다. 7월 첫째 주 다음 모바일 뉴스 서비스에서의 이용자 체류 시간은 루빅스 도입 이전인 5월 둘째 주보다 65% 상승했다. 이 결과는 루빅스의 도입에 따른 클릭 수 상승이 단순한 클릭 어뷰징에 의한 것이 아닌 실제 이용자가 더 많은 뉴스를 보기 위한 소비 행동의 결과임을 의미한다.

3) 노출된 뉴스의 수와 하루 총 뉴스 이용자 수

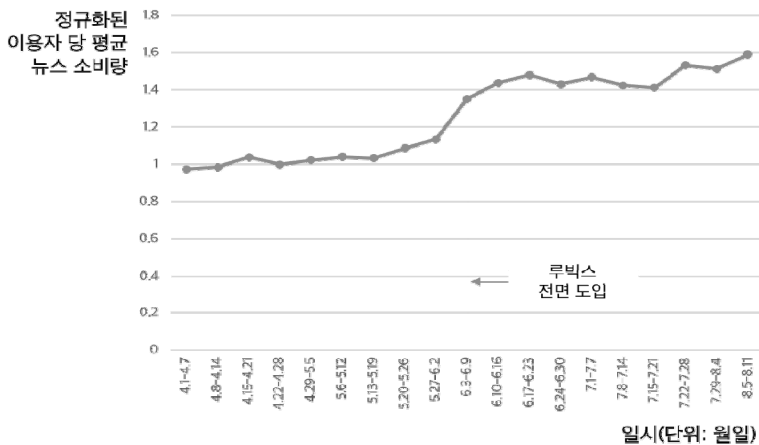


〈그림 9〉 루빅스 전면 도입 전후의 하루 평균 뉴스 이용자 수 추이(자료: 카카오)

루빅스 적용 후 다음 첫 화면에 노출되는 뉴스 콘텐츠도 다양해졌다. 다음 모바일 서비스를 통해 노출된 뉴스의 양은 250% 증가했다. 분야 별로 보면, 사회 126%, 경제 143%, 정치 231%, IT 261%, 문화생활 426%, 그리고 국제 뉴스가 557% 증가했다. 루빅스 도입 이전에는 20대 남성이나 40대 여성처럼 특정 연령대와 성별 선호도를 반영한 뉴스가 제공되긴 어려웠다. 이로 인해 상대적으로 서비스 될 수 있는 뉴스의 소재 폭이 적을 수밖에 없다. 이용자 별로 상이한 뉴스의 전달이 가능한 루빅스 체계에서는 이용자의 뉴스 소비 패턴에 부합하는 콘텐츠를 제공하기 위해, 다음 모바일 뉴스가 전반적으로 제공하는 뉴스가 다양해질 수 있게 됐다.

결과적으로 특정 이용자 군에만 인기 있는 뉴스들의 노출도 증가했다.

루빅스가 도입되고 나서 다음 모바일 뉴스 서비스는 여성향(向) 콘텐츠는 자동적으로 여성 이용자에게만 보여질 수 있는 구조로 변환되면서, 특정 이용자군만이 좋아하는 뉴스도 다음 뉴스 서비스에 게시될 가능성이 높아졌다. 루빅스 체계에서는 범용성이 떨어져서 기존의 뉴스 서비스에서 배제됐던 뉴스들도 소비자에게 적극적으로 제공될 여지가 커진 것이다.



〈그림 10〉 루빅스 전면 도입 전후의 이용자 당 평균 뉴스 소비량 추이(자료: 카카오)

〈그림 9〉는 2015년 4월부터 8월 둘째 주까지의 다음 모바일 뉴스 서비스에서 주별 하루 평균 뉴스 이용자 수를 보여준다. 루빅스 적용 이후 8월 두 번째 주를 기준으로 4월에 비해 뉴스 이용자 수가 45% 증가했다. 이는 루빅스 도입 이후에 다음 모바일 뉴스 서비스의 이용자 규모 자체가 확장됐음을 의미한다. 이는 뉴스 외의 다른 서비스를 사용하기 위해 다음 모바일 홈페이지를 방문한 이용자들이 점차적으로 뉴스를 소비하기 시작함을 보여준다. 기존보다 이용자가 45% 이상 증가하였지만 이용자 당 평균 뉴스 소비량은 줄지 않았다.〈그림 10〉은 루빅스가 적용

된 이후 이용자 당 평균 뉴스 소비량이 59% 증가하였음을 보여준다.

5. 결론: 기계학습 기반 뉴스 추천 구조의 적용 전망 및 함의

본 논문에서는 국내에서는 뉴스 서비스에 최초로 적용된 기계학습 체제인 카카오 루빅스의 도입 배경, 구조, 그리고 성과를 소개했다.

루빅스는 PC에서 스마트폰을 위시한 모바일 기기로 미디어 레퍼토리(Media Repertoire)가 변함에 따른 시장 변화에 적극적으로 대응하기 위한 카카오의 전략적 선택의 결과였다. 모바일 기기로의 미디어 콘텐츠의 소비가 수렴화되는 경향과 더불어 모바일 기기 사용의 일상화는 모바일 기기를 기반으로 개인을 특정하고 개인의 행동 패턴을 분석하여 소비자에게 개인화 된 서비스 제공을 이전에 비해 용이하게 만드는 토대가 됐다.

본 논문에서는 이러한 시대 변화 속에서 카카오가 뉴스 서비스 경쟁력을 제고하기 위한 목적에서 기계학습 기반의 자동화 된 뉴스 서비스 개발 과정을 고찰했다. 카카오는 멀티암드밴딩 알고리즘을 초기 체계의 주요 기술로 선택했다. 그러나, 뉴스의 특성 상 전통적인 멀티암드밴딩 알고리즘을 적용할 수는 없었다. 전통적인 멀티암드밴딩 알고리즘이 하나의 고정된 값을 근거로 클릭률을 평가하는데 반해, 뉴스 서비스는 끊임 없이 유입되고 소멸되는 뉴스들과 이들의 지속적으로 변화하는 클릭률이라는 뉴스 서비스의 특성 상 전통적인 멀티암드밴딩을 준용할 수는 없었다. 동일한 이유로 인해, 담금질 전략 역시도 뉴스 추천 서비스에서는 활용하기 어려웠다. 노출된 이후 시간의 경과에 따른 클릭률의 변화로 인해 소프트맥스와 UCB 역시 뉴스 추천에서 이용하기 어렵다는 사실이

루빅스 개발 과정에서 노정됐다.

이 같은 문제를 카카오는 이동 평균 알고리즘 등을 기반으로 한 맞춤형 멀티암드밴딧 알고리즘을 개발하는 방식으로 해결했다. 서비스 화면 배치에 따른 뉴스 클릭률 편향의 문제는 이중 탐색 방식을 적용하여 해당 문제를 최소화 하는 방향을 고안했다.

루빅스 도입 전후의 효과 지표를 분석한 결과, 다음 모바일 뉴스 서비스의 하루 총 클릭 수를 높였고, 동시에 체류 시간 증가를 통해 늘어난 클릭 수가 클릭 어뷰징이 아닌 실제 이용자의 소비 선택 증가에 따른 결과임이 확인됐다. 루빅스는 하루 총 뉴스 이용자 수를 늘려, 다음 모바일 뉴스 서비스의 소비자 규모 자체를 키운 것으로 나타났으며, 뉴스 서비스 화면을 통해 노출되는 뉴스의 수를 늘려 뉴스의 다양성 제고에도 기여한 것으로 확인됐다. 즉, 루빅스가 뉴스 콘텐츠의 다양화, 뉴스 이용자 규모, 그리고 뉴스 체류 시간 모두 증가하는 선순환 구조를 형성했음을 보여주며, 뉴스의 다양성은 루빅스로 인해 ‘신선한 추천(Serendipity Recommendation)’이 이뤄지고 있음을 설명하고 있다. 이 결과는 루빅스는 기계학습 기반의 자동화 된 개인화 뉴스 서비스로 다음 모바일 서비스에서 소비자의 편익을 높이는데 기여했음을 시사한다.

루빅스는 뉴스 이외에 현재 다양한 개인화 추천과 트렌드 추천 등을 제공하고 있다. 구체적으로 다음 모바일에서는 뉴스를 비롯한 연예, 스포츠 콘텐츠가 포함된 미디어탭, 그리고 홈, 쿠팡, 스타일, 자동차+, 여행 맛집, 쇼핑, 1boon 등의 탭에도 루빅스가 이미 적용되어 있다 루빅스를 통한 카카오의 목표는 이용자의 다양한 맥락에 부합하는 맞춤형 추천 서비스(Context-Aware Recommendation Service)를 제공하는 것이다. 루빅스로 대중적으로 검증된 기계학습 방식의 뉴스 추천 서비스는 점차 확대되는 양상이다. 기계학습 구조의 발전, 그리고 이에 대한 적용 사례의 확대는 기계학습 체계의 콘텐츠 서비스의 활성화에 촉매제가 될

것으로 전망된다.

본 논문에서는 루빅스라는 기계학습 기반의 뉴스 추천 서비스의 구조와 기본 메커니즘을 기술하고, 루빅스의 문제 설정과 그 문제를 풀기 위해 제시된 객관적인 성능 평가 기준, 그리고 루빅스가 실제 적용된 이후의 성과까지의 일련의 과정을 설명했다. 해외와 달리, 국내에서 인터넷 기업의 주요 서비스 알고리즘이 학술적인 공간에 공개되는 것은 이번이 처음이다.

그러나, 본 논문에서는 연구 대상을 서비스 공급자에 한정함으로써, 실제 기계학습 체계가 적용된 뉴스 서비스가 소비자에게 체감되는 평가의 결과는 포괄하지 못했다. 본 논문이 가진 한계인 사용자 측면의 연구 결과 부재에 대해서는 추가적인 연구를 통해 보완하려 한다. 이용자 선택 행동에 기반하여 뉴스가 추천되어 ‘많이 읽힌 기사 중심으로’ 기사가 제공되는 결과가 발생하는 것은 기존의 뉴스 추천 시스템에 내재된 한계였다. 이에, 뉴스의 소비 방식까지 고려된 변인의 삽입이 기존 뉴스 추천 시스템에게는 과제였다. 이와 관련하여, 루빅스 개발진은 이용자들의 뉴스 체류 시간을 열독률이란 개념으로 만들어 신규 변인으로 삽입하는 방식으로 기존의 정량화된 뉴스 추천 체계의 문제를 개선해 나갈 계획이다.

인터넷 기업이 알고리즘을 감춰야 하는 우선적인 기업 비밀로 간주하여 폐쇄적으로 감싸고 있는 국내 현실을 감안하면, 이번 논문의 시도 자체가 학술적으로 매우 의미미한 결정이라고 생각된다. 이 논문을 통해 뉴스 추천 알고리즘의 발전을 위한 건전한 논의의 장이 확대되길 바란다.

■ 참 고 문 헌 ■

- 김대원·윤영민 (2015). SNS에서 형성된 신뢰가 위기 시 방어막이 될 수 있는가. <한국언론학보>, 59권 2호, 196-225.
- 김인중 (2014). Deep Learning: 기계학습의 새로운 트렌드. <한국통신학회지(정보와 통신)>, 31권 11호, 52-57.
- 박현정 (2014). 소비자의 라이프스타일 및 미디어 이용과 N스크린 서비스 수용 간의 관계. <한국콘텐츠학회논문지>, 14권 11호, 143-153.
- 지종민 (2014). ADHD 특성을 가진 영재 학생 지도 경험에 관한 자문화기술지. <교육인류학연구>, 17권 3호, 133-179.
- Agarwal, D., & Chen, B. C. (2009, June). Regression-based latent factor models. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 19-28). ACM.
- Agarwal, D., Chen, B. C., & Elango, P. (2009, April). Spatio-temporal models for estimating click-through rate. In *Proceedings of the 18th international conference on World wide web* (pp. 21-30). ACM.
- Agarwal, D., Chen, B. C., Elango, P., Motgi, N., Park, S. T., Ramakrishnan, R., Roy, S. & Zachariah, J. (2009). Online models for content optimization. In *Advances in Neural Information Processing Systems* (pp. 17-24).
- Aggarwal, C. C., Wolf, J. L., Wu, K. L., & Yu, P. S. (1999, August). Horting hatches an egg: A new graph-theoretic approach to collaborative filtering. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 201-212). ACM.
- Balabanović, M., & Shoham, Y. (1997). Fab: content-based, collaborative recommendation. *Communications of the ACM*, 40(3), 66-72.
- Basilico, J., & Hofmann, T. (2004, July). A joint framework for collaborative and content filtering. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 550-551). ACM.
- Basu, C., Hirsh, H., & Cohen, W. (1998, July). Recommendation as classification: Using social and content-based information in recommendation. In *Aaai/iaai*

(pp. 714-720).

- Billsus, D. & Pazzani, M. J. (1998, July). Learning Collaborative Information Filters. In, Vol. 98, 46-54.
- Bell, R. M., & Koren, Y. (2007a). Lessons from the Netflix prize challenge. *ACM SIGKDD Explorations Newsletter*, 9(2), 75-79.
- Bell, R. M., & Koren, Y. (2007b). Scalable collaborative filtering with jointly derived neighborhood interpolation weights. In *Seventh IEEE International Conference on Data Mining (ICDM 2007)* (pp. 43-52). IEEE.
- Bell, R. M., & Koren, Y. (2007c). Improved neighborhood-based collaborative filtering. In *KDD cup and workshop at the 13th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 7-14). sn.
- Bell, R., Koren, Y., & Volinsky, C. (2007, August). Modeling relationships at multiple scales to improve accuracy of large recommender systems. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 95-104). ACM.
- Breese, J. S., Heckerman, D., & Kadie, C. (1998, July). Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence* (pp. 43-52). Morgan Kaufmann Publishers Inc..
- Chu, W., & Park, S. T. (2009, April). Personalized recommendation on dynamic content using predictive bilinear models. In *Proceedings of the 18th international conference on World wide web* (pp. 691-700). ACM.
- Chu, W., Park, S. T., Beaupre, T., Motgi, N., Phadke, A., Chakraborty, S., & Zachariah, J. (2009, June). A case study of behavior-driven conjoint analysis on Yahoo!: Front Page Today Module. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 1097-1104). ACM.
- Claypool, M., Gokhale, A., Miranda, T., Murnikov, P., Netes, D., & Sartin, M. (1999, August). Combining content-based and collaborative filters in an online newspaper. In *Proceedings of ACM SIGIR workshop on recommender systems*, 60.

- Cosley, D., Lawrence, S., & Pennock, D. M. (2002, August). REFERENCE: An open framework for practical testing of recommender systems using ResearchIndex. In *Proceedings of the 28th international conference on Very Large Data Bases* (pp. 35-46). VLDB Endowment.
- Das, A. S., Datar, M., Garg, A., & Rajaram, S. (2007, May). Google news personalization: scalable online collaborative filtering. In *Proceedings of the 16th international conference on World Wide Web* (pp. 271-280). ACM.
- DeCoste, D. (2006, June). Collaborative prediction using ensembles of maximum margin matrix factorizations. In *Proceedings of the 23rd international conference on Machine learning* (pp. 249-256). ACM.
- Deshpande, M., & Karypis, G. (2004). Item-based top-n recommendation algorithms. *ACM Transactions on Information Systems*, 22(1), 143-177.
- Goldberg, K., Roeder, T., Gupta, D., & Perkins, C. (2001). Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval*, 4(2), 133-151.
- Good, N., Schafer, J. B., Konstan, J. A., Borchers, A., Sarwar, B., Herlocker, J., & Riedl, J. (1999, July). Combining collaborative filtering with personal agents for better recommendations. In *AAAI/IAAI* (pp. 439-446).
- Herlocker, J. L., Konstan, J. A., Borchers, A., & Riedl, J. (1999, August). An algorithmic framework for performing collaborative filtering. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 230-237). ACM.
- Hofmann, T., & Puzicha, J. (1999, July). Latent class models for collaborative filtering. In *IJCAI*, 99, 688-693.
- Huang, Z., Chen, H., & Zeng, D. (2004). Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering. *ACM Transactions on Information Systems*, 22(1), 116-142.
- Karypis, G. (2001, October). Evaluation of item-based top-n recommendation algorithms. In *Proceedings of the tenth international conference on Information and knowledge management* (pp. 247-254). ACM.
- Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R., & Riedl, J. (1997). GroupLens: applying collaborative filtering to Usenet news.

- Communications of the ACM*, 40(3), 77-87.
- Li, L., Chu, W., Langford, J., & Schapire, R. E. (2010, April). A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web* (pp. 661-670). ACM.
- Linden, G., Smith, B., & York, J. (2003). Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, 7(1), 76-80.
- Liu, J., Dolan, P., & Pedersen, E. R. (2010, February). Personalized news recommendation based on click behavior. In *Proceedings of the 15th international conference on Intelligent user interfaces* (pp. 31-40). ACM.
- Melville, P., Mooney, R. J., & Nagarajan, R. (2002, July). Content-boosted collaborative filtering for improved recommendations. In *Aaai/iaai* (pp. 187-192).
- Middleton, S. E., Shadbolt, N. R., & De Roure, D. C. (2004). Ontological user profiling in recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1), 54-88.
- Miller, B. N., Riedl, J., & Konstan, J. A. (1997, January). Experiences with grouplens: Making usenet useful again. In *Proceedings of the 1997 Usenix Winter Technical Conference*, 17.
- Miyahara, K., & Pazzani, M. J. (2000, August). Collaborative filtering with the simple Bayesian classifier. In *Pacific Rim International conference on artificial intelligence* (pp. 679-689). Springer Berlin Heidelberg.
- Mosseri, A. (2016, June). Building a Better News Feed for You. Retrieved from <http://newsroom.fb.com/news/2016/06/building-a-better-news-feed-for-you/>
- Park, S. T., Pennock, D., Madani, O., Good, N., & DeCoste, D. (2006, August). Naïve filterbots for robust cold-start recommendations. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 699-705). ACM.
- Pazzani, M. J. (1999). A framework for collaborative, content-based and demographic filtering. *Artificial Intelligence Review*, 13(5-6), 393-408.
- Pennock, D. M., Horvitz, E., Lawrence, S., & Giles, C. L. (2000, June). Collaborative filtering by personality diagnosis: A hybrid memory-and

- model-based approach. In *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence* (pp. 473-480). Morgan Kaufmann Publishers Inc..
- Popescul, A., Pennock, D. M., & Lawrence, S. (2001, August). Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence* (pp. 437-444). Morgan Kaufmann Publishers Inc..
- Rennie, J. D., & Srebro, N. (2005, August). Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the 22nd international conference on Machine learning* (pp. 713-719). ACM.
- Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. (1994, October). GroupLens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work* (pp. 175-186). ACM.
- Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of research and development*, 3(3), 210-229.
- Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2000). *Application of dimensionality reduction in recommender system-a case study* (No. TR-00-043). Minnesota Univ Minneapolis Dept of Computer Science.
- Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001, April). Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web* (pp. 285-295). ACM.
- Schein, A. I., Popescul, A., Ungar, L. H., & Pennock, D. M. (2002, August). Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 253-260). ACM.
- Shardanand, U., & Maes, P. (1995, May). Social information filtering: algorithms for automating “word of mouth”. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 210-217). ACM Press/Addison-Wesley Publishing Co..
- Ungar, L. H., & Foster, D. P. (1998, July). Clustering methods for collaborative

filtering. In *AAAI workshop on recommendation systems, 1*, 114-129.

Xue, G. R., Lin, C., Yang, Q., Xi, W., Zeng, H. J., Yu, Y., & Chen, Z. (2005, August). Scalable collaborative filtering using cluster-based smoothing. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 114-121). ACM.

논문 투고일 2017년 01월 15일

논문 수정일 2017년 02월 15일

게재 확정일 2017년 02월 20일

ABSTRACT

**News recommendation service using machine
learning: focusing on Kakao's RUBICS**

Seungtaek Park

ShinhanCard, Ph.D in Computer Science

Injae Sung

RUBICS TF in Kakao

Sangwon Seo

Ateam Ventures

Jisoo Hwang

RUBICS TF in Kakao

JiSung Noh

RUBICS TF in Kakao

Daewon Kim*

Policy Support Team in Kakao, Ph.D in Journalism and Mass communication

We study the structure of RUBICS (Real-time User Behavior Interactive Content recommender System) and its performance in the Daum mobile news service (the “News Service”). RUBICS has deployed in the News Service since June 2015. RUBICS is an ensemble recommender system consisting of various recommendation

* corresponding author

algorithms, which provides users with a combination of trend recommendation and personalized one. In this paper, we mainly explain a customized Multi-Armed Bandit (MAB) algorithm, which played a major role in the early version system where most users were in cold-start situations. Then, we explain our performance evaluation methods and test-procedures which were employed in order to improve the algorithms in our system. Finally, we discuss the impact of RUBICS on the News Service by comparing the KPIs of the service before and after deployment of RUBICS. RUBICS increased the average number of clicks per day by 130% and the average number of news clickers per day by 45% between April 2015 and August 2015. Also, the number of news articles shown in the Daum mobile homepage increased by 250% during the same period. These results show that RUBICS is efficient in improving user experiences and therefore, can increase the number of active users in the News Service, and enhance diversity of news articles. We hereby wish to share our knowledge obtained from the development of RUBICS and seek to help build an open discussion for the usage of machine learning in the news recommendation service.

Keywords: Machine Learning, News Recommendation Service, RUBICS, Multi-Armed Bandit