Journal of KIIT. Vol. 18, No. 9, pp. 1-8, Sep. 30, 2020. pISSN 1598-8619, eISSN 2093-7571
http://dx.doi.org/10.14801/jkiit.2020.18.9.1

XGBoost와 Word2Vec을 이용한 온라인 쇼핑 패턴 기반 하이브리드 협업 필터링

박세준*, 성도현**, 변영철***

A Hybrid Collaborative Filtering based on Online Shopping Patterns using XGBoost and Word2Vec

Se-Joon Park*, Do-Hyun Soung**, and Yung-Cheol Byun***

본 논문은 2020년도 산업통상자원부의 재원으로 한국산업기술진흥원의 지원을 받아 수행된 연구임.
(N0002327, 2020년 산학융합지구 조성사업)

요 약

인터넷 서비스의 발전과 사용자의 증가로 방대하게 쌓여가는 정보 속에서 사용자는 자신이 원하는 정보를 빠르게 얻길 원한다. 사용자는 불필요한 정보에 허비되는 시간을 줄이고 싶어 하며 원하는 정보를 얻는 것에 만족감을 느낀다. 그래서 인터넷 서비스는 사용자에게 원하는 정보를 추천해주는 서비스인 협업 필터링(Collaborative filtering)을 제공했다. 본 논문에서는 사용자가 아이템을 구매하기 전 클릭 내역을 쇼핑 패턴으로 인지하고 Word2Vec에 적용시킨 후 머신러닝 모델인 XGBoost에 학습시켜 Word2Vec이 학습 결과의 미치는 영향을 비교한다. Word2Vec을 적용하지 않고 학습하였을 때 추천 정확도는 83.7%, 적용시켜 학습하였을 때 추천 정확도는 85.8%로 Word2Vec이 추천 정확도 증가에 영향을 주는 것을 알 수 있었다.

Abstract

From the huge amount of information accumulated with the development of Internet services and the increase of users, users want quickly the information that they need. Users want to reduce wasted time on unnecessary information and are satisfied with getting the information they need. Therefore, the Internet service provided collaborative filtering, a service that recommends desired information to users. In this paper, the user recognizes the click history before purchasing an item as a shopping pattern, applies it to Word2Vec, and then trains it in XGBoost, a machine learning model, to compare the effect of Word2Vec on the learning result. When learning without applying Word2Vec, the recommendation accuracy is 83.7%, and when learning by applying Word2Vec, the recommendation accuracy is 85.8%, indicating that Word2Vec affects the improvement of recommendation accuracy.

Keywords
machine learning, recommender system, collaborative filtering, pattern recognition, word2vec, xgboost

* 제주대학교 컴퓨터공학과 석사과정
- ORCID: http://orcid.org/0000-0003-2220-9318
** 제주대학교 컴퓨터공학과 학부생 연구원
- ORCID: http://orcid.org/0000-0001-5395-771X
*** 제주대학교 컴퓨터공학과 교수(교신저자)
- ORCID: http://orcid.org/0000-0002-1579-5323

· Received: Jul. 20, 2020, Revised: Sep. 08, 2020, Accepted: Sep. 11, 2020
· Corresponding Author: Yung-Cheol Byun
Dept. of Computer Engineering, Jeju National University, Jejudaehakro 102, Jeju, Jeju Special Self-Governing Province, Korea
Tel.: +82-64-754-3657, Email: ycb@jejunu.ac.kr

Ⅰ. 서 론

인터넷의 발전과 스마트폰 사용자의 증가로 인터넷 사용자에게 제공하는 서비스가 점점 다양해지고 있다. 기존의 웹으로 서비스가 제공되었던 온라인 쇼핑과 같은 전자상거래는 스마트폰의 보급과 발전으로 컴퓨터가 아닌 모바일 인터넷으로도 사용이 가능해졌을 뿐만 아니라 영화나 음악, 인터넷 방송 같은 여가 컨텐츠도 함께 발달되어 인터넷 서비스의 중요도는 날이 갈수록 높아지고 있다. 현재 인터넷 사용자의 수는 사용하는 사람보다 사용하지 않는 사람을 더 찾기가 힘들 정도로 많고, 그로 인해 인터넷 사용자가 다루는 정보의 양은 매우 방대해졌다. 방대하게 쏟아지는 정보들 속에서 사용자가 원하는 정보를 제공하는 것이 중요한데 이를 위해 인터넷 서비스에서 사용자에게 필수적으로 제공하는 것이 추천 시스템이다[1].

전자상거래 Coupang과 Amazon 같은 인터넷 서비스에서 추천 시스템이 중요한 이유는 사용자가 원하는 아이템을 추천해줌으로써 사용자의 불필요한 검색을 줄여주게 되고 자신이 원하는 아이템을 추천 받은 사용자는 만족감을 느낀다. 해당 사이트에서의 쇼핑을 만족한 사용자는 재방문과 재구매를 함으로써 회사는 매출증가라는 효과를 가진다. 이러한 추천 시스템이 가장 활발하게 일어나고 있는 곳 중 하나인 Netflix는 영화나 드라마 등의 여가 컨텐츠를 제공한다. Netflix는 추천도와 함께 연령별 인기 컨텐츠와 장르별 인기 컨텐츠 등의 여러 가지 알고리즘을 이용한 추천을 사용자에게 제공하여 세계 1등의 동영상 스트리밍 회사라는 타이틀을 얻으며 큰 성장을 이뤄냈다. 추천 시스템을 어떻게 잘 활용하는지에 따라 회사의 매출이 정해진다 해도 과언이 아니다.

추천 시스템은 크게 두가지로 나뉘어져 있다. 우선 대부분의 추천 시스템에서 사용하고 있는 기법인 협업 필터링이다. 협업 필터링은 같은 아이템을 좋아하는 사용자에게 성향이 비슷하다는 가정 하에 아이템을 추천하는 시스템이다. 또 다른 추천 시스템은 컨텐츠 기반 추천 시스템(Content based filtering)으로 아이템들 간의 유사도를 측정하여 비슷한 유사도의 아이템들을 추천하는 방식이다. 이 두 가지 추천 시스템의 특징을 더한 것이 하이브리드 추천 시스템(Hybrid recommender systems)이다. 하이브리드 추천 시스템을 사용하고 있는 대표적인 서비스 회사는 앞에서도 설명한 Netflix이다. Netflix는 사용자가 60~90초 내에 10~20개의 제목을 보고, 그중 3개 정도를 자세히 리뷰한다는 것을 파악했다. 단시간 내에 사용자들이 원하는 것을 추천하기 위해 Netflix는 여러가지 추천 시스템을 도입하고 지금 자리 잡은 것이 하이브리드 추천 시스템이다. Netflix는 사용자에게 유사한 성향의 다른 사용자가 좋아하는 영상을 추천하고 해당 영상과 유사도가 높은 다른 영상을 함께 추천하며 하이브리드 추천 시스템을 구축하고 있다.

본 논문에서도 하이브리드 추천 시스템을 이용한 온라인 쇼핑 아이템 추천 방법을 제시한다. 온라인 쇼핑 사용자의 아이템 클릭을 기반으로 쇼핑 패턴을 인식하고 최종적으로 구매할 아이템을 예측하여 추천하는 방식이다. 또한 Word2Vec을 이용하여 아이템들 간의 유사도를 구하고 추천해줄 아이템과 가까운 유사도의 아이템들을 함께 추천한다.

본 논문은 2장에서 관련연구에 대한 내용을 다룬다. 추천 시스템의 중요성과 협업 필터링에 대하여 자세히 다루고 컨텐츠 기반 추천 시스템에서 아이템들 간의 유사도를 출력시켜주는 연구들을 소개한다. 또한 추천 시스템의 발전이 어느 정도 이루어졌는지를 설명한다. 3장에서는 본 논문에서 제시하는 추천 방법과 Word2Vec을 이용하여 유사도를 구하는 과정을 소개하며 머신러닝 모델 중에 하나인 XGBoost로 쇼핑 패턴을 학습한다. 4장에서는 사용자가 구매할 아이템을 예측하고 실제 구매한 아이템과의 정확도와 Word2Vec이 추천 정확도에 미치는 영향을 보여준다. 마지막으로 5장에서는 본 논문의 결론을 보여주며 마무리 짓는다.

Ⅱ. 관련 연구

2.1 추천 시스템

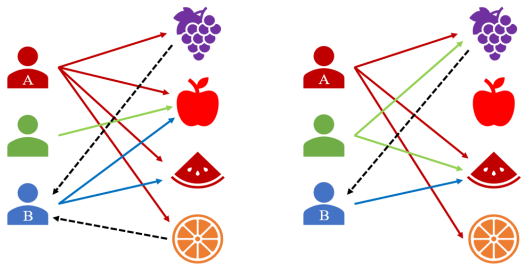
인터넷 서비스는 기본적으로 사용자에게 편의와 만족을 주기 위해 생긴 것이며 어떻게 인터넷 서비스를 발전시켜 사용자에게 제공하는 가가 사용자

수의 크게 영향을 미친다. 사용자의 수는 곧 해당 인터넷 서비스 회사의 매출과 규모를 나타내며 이러한 추천 서비스의 중요성을 느낀 인터넷 서비스 회사들은 추천 시스템을 계속해서 발전시켜왔다.

초반 추천 시스템의 방식은 사용자가 검색한 아이템과 비슷한 아이템을 추천하는 것이 전부였다. 사용자가 관심 있는 아이템과 비슷한 아이템을 추천해줌으로써 추천 서비스는 시작되었고, 점차 발전하게 되면서 단순히 비슷한 아이템을 추천하는 것을 넘어서 사용자가 선호할 만한 아이템을 추측하여 여러 가지 항목 중 사용자에게 적합한 특정 항목을 선택(Information filtering)을 하여 제공한다.

협업 필터링은 그림 1과 같이 사용자 기반 필터링과 아이템 기반 필터링 두 가지로 나뉜다. 우선 사용자 기반 필터링의 추천 방식은 A사용자가 포도, 사과, 수박, 오렌지를 좋아하고 B사용자가 사과, 수박을 좋아한다고 했을 때 A사용자와 B사용자의 성향이 비슷하다고 가정한다. 그리고 B사용자에게 A사용자가 좋아하는 다른 아이템인 포도와 오렌지를 추천하는 방식이다. 아이템의 유사성을 제외하고 단순히 사용자 간의 성향을 따져 아이템을 추천해준다. 아이템 기반 필터링은 B사용자가 수박을 좋아했을 때 다른 사용자들이 수박과 함께 포도를 선호하는 것을 확인하고 수박과 포도는 서로 비슷한 아이템이라 인지하고 포도를 B사용자에게 추천해준다[2].

정리하자면 사용자 기반 필터링은 사용자의 중점을 두고 비슷한 성향의 사용자의 선호 아이템을 추천한 방식이고 아이템 기반 필터링은 아이템에 중점을 두고 비슷한 성향의 아이템을 추천하는 방식이라 정리할 수 있다[3].



(a) 사용자 기반 필터링 (b) 아이템 기반 필터링

그림 1. 협업 필터링과 종류

Fig. 1. Collaborative and types, (a) User-based filtering, (b) Item-based filtering

최근 네이버 검색 엔진에서도 개개인의 사용자에게 맞춤 검색 엔진을 사용한다. 사용자는 연령, 관심있는 분야 등을 설정할 수 있고 다양한 알고리즘으로 정보를 제공받을 수 있다. 추천 시스템 이외에도 다른 인터넷 서비스에서 사용자 중심의 서비스가 끊임없이 발전하고 사용자에게 어떻게 서비스를 제공하며 만족을 주는 지가 현재 인터넷 서비스에서 매우 중요하다.

2.2 아이템 간의 유사도 출력에 대한 연구

아이템 간의 유사도를 나타내는 기법에는 TF-IDF(Term Frequency-Inverse Document Frequency)와 Word2Vec등 여러가지가 있지만 본 논문에서는 Word2Vec을 사용한다. 아이템 간의 유사도를 나타내는 대표적인 기법인 TF-IDF와 Word2Vec의 차이점에 대해 설명하자면 우선 TF-IDF는 문서 내에 특정단어의 개수에 따라 값을 나타내는 단어빈도(TF)와 어떤 단어가 문서 전체에서 많이 나오는지를 나타내는 역문서 빈도(IDF)를 곱하여 TF-IDF 값을 구할 수 있다. 정리하면 특정 문서에서의 단어 빈도와 문서 전체에서의 단어 빈도를 비교하여 특정 문서에서 해당 단어의 중요도를 나타낼 수 있다.

Word2Vec은 One-hot-vector에서 발전됐는데 One-hot-vector는 0과 1을 이용한 이진법으로 벡터를 나타낸다. 먼저 각 단어에 고유한 인덱스를 할당하고 해당 단어에게 할당된 인덱스 순서에 1의 값을 주고 나머지 벡터자리에는 0을 줌으로써 각 단어마다의 자리를 할당한다. One-hot-vector의 단점은 먼저 희소표현(Sparse representation)이다. 희소표현 이란 예를 들어, 단어가 100개가 있다 가정했을 때 벡터의 차원은 100이 된다. 만약 고양이라는 단어의 인덱스가 6으로 할당 되어있을 때 [0,0,0,0,0,1,0…0,0]의 100차원 벡터로 표현된다. 또한 이진법으로 표현되기 때문에 단어들 간의 유사도를 구할 수 없다는 단점을 가지고 있다.

이를 해결하기 위해 고안된 것이 Word2Vec이며 밀집표현(Dense representation)이다. 밀집 표현은 단어의 개수로 벡터의 차원이 정해지는 것이 아닌 사용자가 직접 설정한 값으로 모든 단어의 벡터의 차

원이 맞춰진다[4]. 사용자가 직접 설정이 가능한 이유는 Word2Vec에서는 벡터의 값이 아이템들 간의 유사도로 표현되기 때문이다. 이진법이 아닌 실수 형태의 유사도로 표현되기에 단어의 개수에 제약을 받지 않는 장점이 있다.

III. 제안하는 방법

본 논문이 제안하는 방법에 대한 연구 진행 순서는 그림 2와 같다. 추천 방법과 유사도를 출력하는 기법을 기존의 추천 시스템들과는 조금 다른 방법을 제시한다. 데이터 수집되는 과정은 먼저, 각 사용자가 원하는 아이템을 찾기 위해 클릭한 내역들을 수집한다. 한 명의 사용자가 원하는 아이템을 구매하기 위해 여러 아이템들을 클릭하는 동안 데이터는 쌓이며 최종적으로 아이템을 구매하기 위해 장바구니에 담는데 까지가 하나의 쇼핑 패턴으로 지정한다.

본 논문의 실험 데이터 셋 형태는 표 1과 같다. 데이터 셋은 50일 간에 '이제주몰'을 이용한 총 만명의 사용자들의 데이터를 사용한다. 데이터 셋의 세로축은 각각 '이제주몰'에서 쇼핑을 했던 만명의 사용자들이고 가로축은 해당 사용자가 아이템을 한 번 클릭할 때마다 쌓인 클릭 기록이다. 가장 많이 아이템을 클릭한 사용자의 클릭 횟수는 13번이고, 14번째 아이템은 최종적으로 사용자가 구매한 아이템이다. 또한 구매하기까지 아이템 클릭 횟수가 4번 미만인 사용자들은 모두 제외한다. 이유는 하나의 패턴으로 인식시키기에 쇼핑 패턴이 짧고 데이터가 불분명하기 때문이다.

예를 들어 온라인 쇼핑을 이용하기에 앞서 구매하기로 생각한 아이템을 바로 사버리는 경우와 금방 원하는 아이템을 발견한 경우 하나의 쇼핑 패턴으로 인식하기는 데이터가 부족하여 제외한다[5].

데이터 전처리한 이후에 Word2Vec을 이용하여 아이템 간의 유사도를 구한다. 본 연구에서 Word2Vec을 사용한 목적은 각 사용자들의 쇼핑 패턴 안에 있는 아이템들 간의 유사도가 높게 출력되기 때문에 패턴을 인식하는데 있어서 효과적이다.

Word2Vec을 이용하여 출력된 각 아이템의 유사도로 머신러닝 모델 XGBoost에 쇼핑 패턴을 학습시킨다. 구매하기 전 클릭했던 아이템들을 쇼핑 패턴으로 인지하고 학습 데이터로 학습 시킨 후, 최종적으로 구매할 아이템을 예측 데이터로 할당한다. 학습시킨 쇼핑 패턴을 기반으로 사용자가 구매할 아이템을 예측하고 예측한 아이템과 유사도가 높은 아이템들을 함께 추천한다.

표 1. 실험 데이터 셋 형태
Table 1. Experimental data set's shape

Click count / User	1	2	…	14
User A	Item	Item	…	Item
User B	Item	Item	…	Item
User C	Item	Item	…	Item
…	Item	Item	…	Item

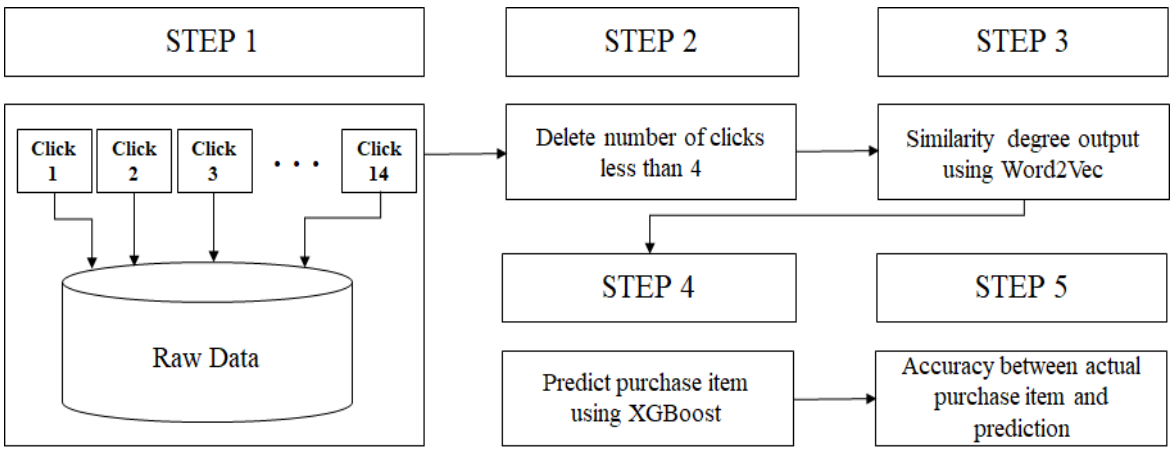


그림 2. 사용자의 쇼핑 패턴을 기반으로 한 추천 방법
Fig. 2. Recommendation method based on user's shopping pattern

Ⅳ. 연구 환경 및 성능 평가

4.1 연구 환경

본 연구는 사용자에게 더 나은 서비스를 제공하기 위해 고안되었다. 추천 시스템 자체는 사용자에게 편리함을 제공하기 위해 존재하며 사용자는 빠른 시간 안에 자신이 원하는 데이터를 얻길 바란다. 추천 시간은 구현 환경에 따라 걸리는 시간은 각기 다르고 머신러닝 특성상 CPU를 이용하여 학습을 시키기 때문에 CPU 사양에 대한 영향이 크다.

표 2는 본 연구의 개발 환경에 대한 테이블이다. Window 10 pro 64bit에서 진행하였고 Google chrome 브라우저를 이용하였다. 또한 데스크톱의 CPU는 Intel i5-9600k와 16GB의 RAM, 프로그래밍 언어 Python 버전 3.7.6의 Jupyter notebook에서 연구를 진행하였다.

표 2. 개발 환경
Table 2. Development environment

Programming language	Python 3.7.6
Operating system	Window 10 pro 64bit
Browser	Google chrome
Library and framework	Jupyter notebook
CPU	Intel(R) Core(TM) i5-9600k@3.70GHz
Memory	16GB

본 논문의 추천 시스템의 아이템 추천 계산 속도는 표 3과 같다. 총 10번 추천했을 때 아이템 하나에 대해 추천하는데 평균 0.0451초의 시간이 걸린다. 추천 계산 시간은 해당 추천 시스템을 서버에 적용시켰을 때 얼마나 빠른 시간 안에 사용자에게 아이템을 추천해주는 가에 대한 의미를 가진다[6].

표 3. 아이템 추천의 계산 시간
Table 3. Computation time of the recommended item

Recommendation time	Computation time	Recommendation time	Computation time
1	0.0448	6	0.0450
2	0.0453	7	0.0449
3	0.0450	8	0.0444
4	0.0448	9	0.0449
5	0.0468	10	0.0452
Average time		0.0451	

학습을 시키는 데에 있어서 학습이 제대로 이루어지고 있는지 확인할 수 있는 척도는 에러를 측정하는 것이다. 에러를 측정하는 방법에는 여러 가지가 있지만 본 논문에서는 MAE(Mean Absolute Error)를 사용했다[7]. MAE는 평균 절대 오차로 예측 값과 실제 값의 차이를 나타낸다. 오류는 예측하고자 하는 값이 제대로 예측하고 있지 못할 때 값이 크게 출력된다.

그림 3은 본 논문에서 사용한 학습데이터의 에러가 학습을 진행하며 감소하는 것을 보여준다. 값이 작을수록 예측이 제대로 진행되고 있는 것이고 학습이 계속 진행됨에 따라 오류는 계속해서 줄어든다[8].

본 논문에 사용한 데이터 셋의 개수는 표 4와 같다. 실험에 진행된 총 사용자는 10000명이며 사용자 중에 가장 많이 아이템을 클릭한 횟수는 14번, 가장 적게 아이템을 클릭한 횟수는 4번이다. 학습데이터+에 사용된 사용자의 수는 80%로 8000명, 테스트데이터 즉 예측에 사용된 사용자의 수는 20%를 사용한 2000명이다.

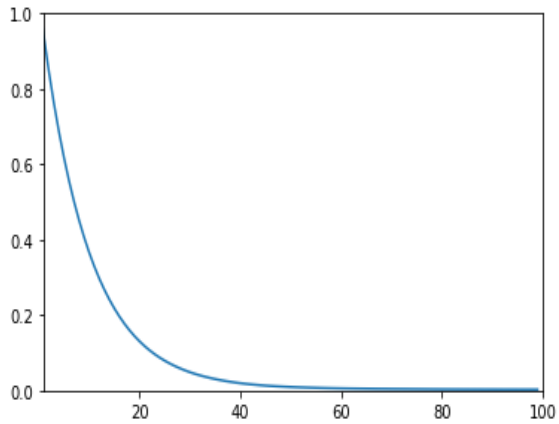


그림 3. 학습데이터 에러
Fig. 3. Train data error

표 4. 추천 데이터 셋
Table 4. Dataset for recommendation

Total number of records	10000
Maximum number of clicks	14
Minimum number of clicks	4
Training data	80%
Test data	20%

4.2 성능 평가

XGBoost는 트리형태의 부스팅(Boosting) 모델로 규모가 큰 데이터를 다루는 데에 안정성이 높고 다른 부스팅 모델에 비해 학습 속도가 빨라 본 연구에서는 XGBoost Classifier 모델을 채택했다. 머신러닝에는 크게 부스팅 방식과 배깅(Bagging) 방식 두 가지로 나뉜다. 우선 배깅 방식은 크기가 동일한 여러 개의 데이터를 각각 다른 모델에 학습시키는 방식으로 출력되어 나온 값의 평균으로 최종 값을 결정한다. 부스팅 방식은 한 모델에서 학습을 시켜 나온 데이터의 오류에 더 높은 가중치를 부여함으로써 계속해서 모델을 데이터에 알맞게 수정한다.

XGBoost는 순위와 회귀, 분류를 지원하고 하이퍼 파라미터들이 다양하기 때문에 활용성이 높아 사용자가 원하는 형태의 학습을 시킬 수 있다. 학습시킬 데이터에 알맞은 머신러닝 모델과 그 모델의 하이퍼 파라미터를 지정해줌에 따라 추천 정확도가 크게 달라진다. XGBoost의 하이퍼 파라미터 중 하나인 n_estimators는 학습 알고리즘의 개수로 기본 값은 100이며 쉽게는 학습 횟수를 의미한다.

표 5는 100부터 1000까지 100씩 학습했을 때의 추천 정확도이다. 설정 값 100에서 추천 정확도는 81.45%, 500을 설정했을 때 85.95%의 추천 정확도를 보였다. 100부터 500까지의 추천 정확도는 계속해서 증가한 반면 500부터 1000까지의 추천 정확도는 86%를 유지하였다.

표 6은 Word2Vec을 적용하기 전 아이템들의 데이터를 가지고 학습하여 예측했을 때 추천 정확도로 평균 83.7%이다. 표 7은 Word2Vec을 적용한 후 예측했을 때 추천 정확도로 Word2Vec의 적용 유무의 따라 추천 정확도의 차이를 비교한다. 최종적으로 10번을 예측했을 때 예측하여 나온 아이템과 실제 구매 아이템 간의 정확도는 평균 85.8%이다. Word2Vec을 적용했을 때의 추천 정확도가 적용하지 않았을 때의 추천 정확도보다 2.1% 증가한다. 위 결과로써 Word2Vec이 추천 정확도의 긍정적인 영향을 미치는 것을 확인할 수 있다[9].

실제 온라인 쇼핑몰을 이용할 때 하나의 아이템만 추천해주는 것이 아닌 여러가지 비슷한 아이템을 추천해주는 것처럼 Word2Vec으로 출력된 아이템 간의 유사도를 이용하여 가장 가까운 아이템 5개를 함께 추천해주는 방식으로 연구를 진행하여 단순히 하나의 아이템만 추천했을 때와 정확도를 비교한다. 표 8은 유사도가 가까운 아이템 5개를 추가하여 추천 정확도를 나타낸 결과이다.

표 5. 하이퍼 파라미터 n_estimators에 따른 정확도
Table 5. Recommendation accuracy by n_estimators

n_estimators	Recommendation accuracy (%)	n_estimators	Recommendation accuracy (%)
100	81.5	600	86.0
200	84.0	700	85.9
300	84.5	800	86.0
400	85.5	900	86.1
500	86.0	1000	86.0

표 6. Word2Vec 적용 전 추천 정확도
Table 6. Recommendation accuracy before using Word2Vec

Recommendation time	Recommendation accuracy (%)	Recommendation time	Recommendation accuracy (%)
1	85.1	6	83.6
2	82.6	7	83.8
3	84.3	8	82.6
4	84.1	9	84.0
5	83.9	10	83.6
Average accuracy			83.7

표 7. Word2Vec 적용 후 추천 정확도
Table 7. Recommendation accuracy after using Word2Vec

Recommendation time	Recommendation accuracy (%)	Recommendation time	Recommendation accuracy (%)
1	86.2	6	85.4
2	84.7	7	86.3
3	86.5	8	84.5
4	86.4	9	85.9
5	86.1	10	86.4
Average accuracy			85.8

표 8. 유사도가 가까운 아이템 5개 추가했을 때의 추천 정확도
Table 8. Recommendation accuracy with 5 close similarity items

Recommendation time	Recommendation accuracy (%)	Recommendation time	Recommendation accuracy (%)
1	87.6	6	86.0
2	85.1	7	86.7
3	87.2	8	85.2
4	86.9	9	86.6
5	87.1	10	87.0
Average accuracy			86.5

유사도가 가까운 아이템 5개를 추가하여 10차례 예측했을 때 추천 정확도는 86.5%이다. 구매하기 위해 장바구니에 아이템을 넣기 전 클릭했던 아이템들의 데이터, 즉 구매 전 쇼핑 패턴들을 가지고 하나의 아이템을 예측한 결과와 예측한 아이템과 가까운 유사도의 아이템 5개의 정확도를 비교했을 때 평균 0.7% 정확도의 증가를 확인할 수 있다.

Ⅴ. 결 론

본 논문에서는 사용자의 온라인 쇼핑 패턴을 기반으로 데이터를 수집하고 수집한 데이터를 토대로 사용자에게 맞춤 추천 시스템을 제공한다. 또한 구매할 것이라 예측하고 추천한 아이템과 Word2Vec을 이용하여 출력된 각 아이템 간의 관계를 수치로 알아볼 수 있고 가까운 유사도의 아이템을 함께 추천해줌으로써 사용자를 중심으로 추천하는 협업 필터링, 아이템 중심으로 추천하는 컨텐츠 기반 필터링을 모두 사용한 하이브리드 추천 시스템 방식을 제안한다.

Word2Vec의 적용 유무에 대한 추천 정확도를 비교한 결과 Word2Vec을 적용했을 때가 적용하지 않았을 때에 비해서 2.1%의 정확도 차이를 보였다[10]. 이로써 Word2Vec은 아이템 간의 유사도를 나타내는 것뿐만 아니라 학습 결과에도 긍정적인 영향을 미치는 것을 확인할 수 있었다.

인터넷 서비스와 스마트폰의 개발로 사용자들은 폭발적으로 증가하고 지금 이 시간에도 계속해서 쏟아지고 있는 인터넷의 방대한 정보 속에서 사용자가 원하는 정보를 얻기란 쉽지 않다. 방대한 정보 속에는 불필요한 정보들이 즐비하고 이를 해결하고 사용자에게 편리를 주기 위해 인터넷 서비스는 이제까지 계속 발전해왔고 앞으로도 끊임없이 발전해 나갈 것이다. 본 논문 또한 인터넷 서비스 사용자가 원하는 정보를 보다 효과적으로 전달하기 위해 고안되었다. 본 연구는 규모가 큰 데이터를 다루며 실제 사용자가 구매한 아이템과도 정확도가 높다.

기존의 추천 시스템과는 다른 사용자 맞춤 추천 서비스를 제공함으로써 Amazon과 Netflix 등의 인터넷 서비스 회사들은 큰 성장을 이뤄냈다. 이처럼 추천 서비스가 현재 인터넷 서비스에서 많은 영향을 끼치고 있다.

References

[1] Haesung Lee and Joonhee Kwon, "A New Distributed Graph Data Storage System for Large-Scale Recommender Engines", The Journal of Korean Institute of Information Technology, Vol. 11, No. 7, pp. 139-149, Jul. 2013.

[2] Z Zhang, M Dong, K Ota, and Y Kudo, "Alleviating new user cold-start in user-based collaborative filtering via bipartite network", IEEE Transactions on Computational Social Systems, Vol. 7 No. 3, pp. 672-685, Mar. 2020.

[3] Yunju Lee, Haram Won, Jaeseung Shim, and Hyunchul Ahn, "A Hybrid Collaborative Filtering-based Product Recommender System using Search Keywords", Journal of Intelligent Information Systems Society, Vol. 26 No. 1, pp. 151-166, Mar. 2020.

[4] Hyungsuc Kang and Janghoon Yang, "Analyzing Semantic Relations of Word Vectors trained by The Word2vec Model", Journal of KIISE, Vol. 46 No. 10, pp. 1088-1093, Oct. 2019.

[5] Siwoon Son, Myeong-Seon Gil, and Yang-Sae Moon, "Anomaly Detection Technique of Log Data Using Hadoop Ecosystem", KIISE Transactions on Computing Practices, Vol. 23 No. 2, pp. 128-133, Feb. 2017.

[6] Zeinab Shahbazi and Yung-Cheol Byun, "Product Recommendation Based on Content-based Filtering Using XGBoost Classifier", International Journal of Advanced Science and Technology, Vol. 29 No. 4, pp. 6979-6988, Jul. 2020.

[7] Geunsik Jeon, Sungeon Kong, and Yongsuk Choi, "Word2Vec based collaborative filtering for movie rating prediction", The Korean Institute of Information Scientists and Engineers, Vol. 2017 No. 12, pp. 844-846, Dec. 2017.

[8] Jian Wei, Jianhua He, Kai Chen, Yi Zhou, and Zuoyin Tang, "Collaborative filtering and deep learning based recommendation system for cold start items", Expert Systems with Applications, Vol. 69, pp. 29-39, Mar. 2017.

[9] Boo Sik Kang, "Improving Predictive Accuracy of User-based Collaborative Filtering Using Word2Vec", Journal of Knowledge Information Technology and Systems, Vol. 13 No. 1, pp. 169-176, Feb. 2018.

[10] Yunhwan Keon, Hyuna Kim, Jin Young Choi, Dongho Kim, Su Young Kim, and Seonho Kim, "Call Center Call Count Prediction Model by Machine Learning", Journal of JAITC, Vol. 8, No. 1, pp. 31-42, Jul. 2018.

저자소개

박 세 준 (Se-Joon Park)

2020년 2월 : 제주대학교 컴퓨터공학과(공학사)
2020년 3월 ~ 현재 : 제주대학교 컴퓨터 공학과 석사과정
관심분야 : 인공지능(머신러닝, 딥러닝), 자연어 처리, 인지과학

성 도 현 (Do-Hyun Soung)

2014년 3월 ~ 현재 : 제주대학교 컴퓨터 공학과 학사과정
2019년 6월 ~ 2020년 1월 : UCI 학부생 방문 연구원
2019년 3월 ~ 현재 : 제주대학교 머신러닝 Lab 학부생 연구원
관심분야 : 자율주행, 백엔드 개발, 클라우드 컴퓨팅

변 영 철 (Yung-Cheol Byun)

1993년 2월 : 제주대학교 컴퓨터공학
1995년 3월 ~ 2001년 2월 : 연세대학교 컴퓨터과학 석박사
2000년 ~ 2001년 : 삼성전자 IT 전문 인스트럭터
2001년 ~ 2003년 : 한국전자통신연구원(ETRI) 선임 연구원
2012년 ~ 2014년 : 플로리다대학교 연구 교수
2003년 ~ 현재 : 제주대학교 컴퓨터공학과 교수
관심분야 : AI와 머신러닝, 패턴인식, 지능형 블록체인 시스템, 빅데이터, 지식발견, 시계열 데이터 분석, 지능형 에이전트, 의료영상처리