석사학위논문

Word2Vec을 이용한 하이브리드 머신러닝 모델 기반 상품 추천 시스템

제주대학교 대학원

컴퓨터공학과

박 세 준

2022년 2월

Word2Vec을 이용한 하이브리드 머신러닝 모델 기반 상품 추천 시스템

지도교수 변 영 철

박 세 준

이 논문을 컴퓨터공학 석사 논문으로 제출함

2021年 12月

박세준의 컴퓨터공학 석사학위 논문을 인준함

심사위원장 _____ ㊞

위 원 _____ ㊞

위 원 _____ ㊞

제주대학교 대학원

2021年 12月

Product recommendation system based on hybrid machine learning model using Word2Vec

Se-Joon Park
(Supervised by professor Yung-Cheol Byun)

A thesis submitted in partial fulfillment of the requirement for the degree of Master of Computer Engineering

2021. 12.

This thesis has been examined and approved.

Thesis Committee Chair, Ki-Joong Ahn	Prof. of Jeju National University
Thesis Committee Member, Sang-Joon Lee	Prof. of Jeju National University
Thesis Director, Yung-Cheol Byun	Prof. of Jeju National University

Date

Department of Computer Engineering
GRADUATE SCHOOL
JEJU NATIONAL UNIVERSITY

ABSTRACT

Now that non-face-to-face activities are widespread all over the world, more and more users are looking for online shopping malls that can avoid contact with people and carry out consumer activities. In addition, the importance of the recommendation system is growing as it attracts many users with its own recommendation systems such as Netflix and Amazon.

We propose a recommender system that uses Word2Vec to acquire similarities between items as a vector, recognizes shopping history using a machine learning model, and improves the accuracy of recommendations. The model used for training used a hybrid model that combined Random Forest, XGBoost, and Extra Tree to calculate the recommended accuracy. Existing data has the unique number of the item as data, it was only possible to use the item classification model. However, the similarity between items is expressed numerically using Word2Vec, so you can use a regression model. The availability of regression models is more than 10% better than the performance of classification models in Word2Vec's multi dimensional function. The learning time was about 60 minutes in the 5-dimensional vector of classification model, but the regression model showed a fast learning time in the about 1 minute.

The recommendation accuracy of the classification model before using Word2Vec and the recommendation accuracy of the classification model and the regression model after applying Word2Vec were compared. We also compared the increase in recommendation accuracy for each dimension of Word2Vec. The recommendation accuracy of the hybrid model before using Word2Vec was 84.23%. However, after applying Word2Vec, it showed that

the 5-dimensional vector recommendation accuracy of the classification model increased to 87.46%. Also, when comparing the accuracy of the classification model and the regression model after applying Word2Vec, the recommendation accuracy of the regression model in the Word2Vec 5-dimensional vector was 99.12%, which was superior to the classification model in performance and learning time.

목차

그림 목차

표 목차

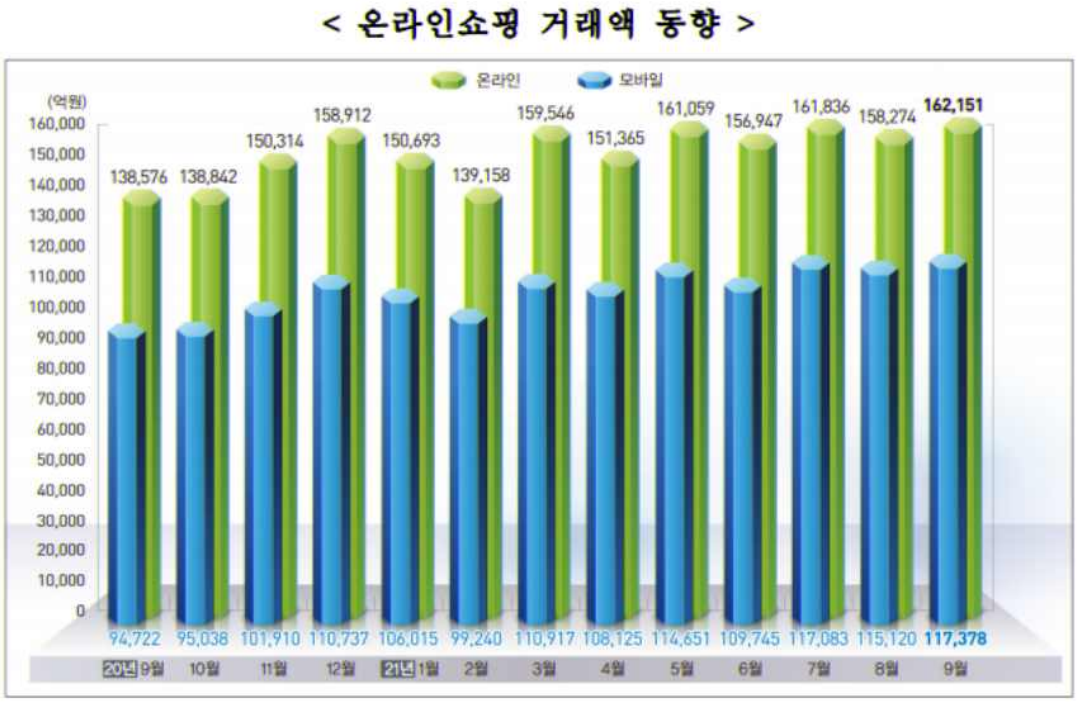
Ⅰ. 서 론

1.1 연구 배경 및 목적

1.1.1 연구 배경

COVID-19가 전 세계적으로 확산됨에 따라 각 정부에서는 최대한 접촉을 피하는 비대면 활동(untact)을 하도록 권하고 있다. 이로 인해 쇼핑몰 이용자들은 직접 오프라인 매장을 이용하기보다는 인터넷을 이용한 온라인 쇼핑을 선호하기 시작했으며 비대면 활동이 쇼핑에서도 활발히 일어나고 있다[1]. 한국 통계청 자료에 따르면 본격적으로 한국에 COVID-19가 확산된 2020년 2월부터 2020년 12월까지 온라인 쇼핑 총 거래액이 전년인 2019년과 동월대비 평균 15%이상 증가한 것으로 나타났다. 2021년 또한 계속되는 비대면 소비 활동으로 인하여 전년인 2020년보다도 온라인 쇼핑 총 거래액이 약 17% 증가한 것을 확인할 수 있었다[2]. 비대면 활동이 유행하여 온라인 쇼핑몰을 찾는 이용자들이 증가함에 따라 추천 시스템의 중요성이 높아졌다.

대부분의 소비자들이 스마트 폰을 소지하고 있는 현재, 스마트 폰을 활용한 모바일 쇼핑 또한 늘고 있다. 이를 그림 1-1에서 확인할 수 있다. 온라인 쇼핑 총 거래액 중 약 72%를 차지고 하고 있으며 모바일 쇼핑 동향 또한 전년 동월대비 평균 24% 증가하고 있다. 네이버 페이나 카카오 페이 등 간편해진 모바일 결제 시스템과 모바일 어플리케이션 개발자들에게 많은 투자를 하고 있는 모바일 쇼핑 플랫폼이 제공하는 쾌적한 쇼핑 환경 덕분에 온라인 쇼핑의 소비가 계속해서 증가했다.



[그림 1-1] 온라인 쇼핑 거래액 동향 (한국 통계청)

SNS나 동영상 스트리밍, 온라인 쇼핑몰과 같은 인터넷 서비스들이 다양해짐에 따라 인터넷 서비스를 이용하는 이용자들의 증가로 인터넷이 다루는 데이터의 규모 또한 커졌다. 그로 인해 인터넷 이용자들은 많은 인터넷 정보 속에서 자신이 원하는 정보를 얻는데 많은 시간을 보낸다. 인터넷 이용자들이 원하는 정보를 쉽게 찾을 수 있도록 돕기 위해서는 추천 시스템이 필요하다. 아마존, 넷플릭스 등의 세계적인 동영상 스트리밍 서비스는 자신들만의 독자적인 추천 시스템을 구축하여 서비스 이용자들의 니즈를 충족시켜 큰 성장을 거뒀다. 아마존은 인터넷상에서 책을 파는 것을 시작으로, 점차 사업을 확장하여 지금의 아마존이 되었다. 아마존은 전체 판매의 약 35% 정도가 추천 시스템을 통해 소비가 이루어지고 있으며, 이것은 소비자들이 구매 계획이 없던 상품을 추천 시스템을 통해 만족감을 가지며 추가 구매하는 것을 의미한다. 아마존의 추천 시스템 방식은 상품들 간의 유사성을 수치화하고, 사용자가 구매한 상품과 유사성이 가까운 것을

추천하는 방식이다. 넷플릭스는 DVD 대여와 온라인으로 영화나 드라마를 제공하는 온라인 스트리밍 회사였다. 2009년 Netflix Prize 라는 경진대회를 열어 100만 달러의 상금을 걸고 현재 넷플릭스 추천 시스템의 성능보다 10% 향상된 추천 알고리즘에 대한 대회를 열었다. 이 대회의 우승자는 당시 여러 개의 머신러닝 알고리즘을 결합한 앙상블 기법을 이용하였다. 넷플릭스는 자신들의 성공요인을 독자적인 추천 시스템이라고 주장하고 있으며 추천 시스템의 중요성을 강조했다[3].

1.1.2 연구 목적

본 연구는 온라인 쇼핑 소비자들에게 보다 높은 만족도의 쇼핑 서비스를 제공하기 위해 소비자 관점에서 쇼핑 패턴 분석 및 추천 정확도 증가를 목적으로 연구를 수행한다.

- 첫째, 실제 온라인 쇼핑몰의 데이터를 이용하여 각 고객들이 상품을 구매하기 전까지의 살펴본 상품들을 순서대로 나열한 것을 한 개의 데이터로 인식하고 패턴 분석을 한다. 여러 소비자들에 데이터를 모아 성향이 비슷한 소비자들이 함께 살펴본 상품들 간의 연관성을 만들어준다. 이때, 상품들 간의 상관관계를 수치로 표현하며 연관성이 큰 상품들 간의 수치는 가깝게 나타난다. 이러한 과정을 통해 온라인 쇼핑에서 소비자 행동의 이해에 대한 연구에 도움이 되도록 하고자 한다.

- 둘째, 앞서 분석한 데이터를 토대로 여러 개의 머신러닝 알고리즘을 이용하여 학습하고 결과를 비교한다. 결과가 좋게 나온 머신러닝 알고리즘을 결합하여 앙상블 알고리즘인 하이브리드 모델을 이용한다. 단일 머신러닝 알고리즘보다 향상된 앙상블 알고리즘을 이용하여 추천 정확도를 증가시키기 위함이다.

- 셋째, 상품 추천 시스템에 이용되는 일반적인 학습 방식은 분류(classification) 모델을 이용하는 것이 아닌 회귀(regression)모델을 이용한다. 상품 간의 연관

성이 없이 인코딩된 데이터를 이용했을 때는 수치를 기반으로 예측하는 회귀 모델을 사용할 수 없다. 하지만 Word2Vec을 이용하여 상품 간의 연관성을 수치로 표현하고 회귀모델을 사용함으로써 학습 속도와 성능 모두 향상된 것을 확인할 수 있었다. 기존 Word2Vec의 사용 방식이 아닌 새로운 방식의 상품 추천 시스템을 제안함으로써 추천 알고리즘 연구에 기여되도록 하고자 한다.

1.2 연구 방법 및 논문 구성

1.2.1 연구 방법

본 연구는 온라인 쇼핑 이용고객의 쇼핑 내역을 기반으로 패턴 인식을 진행한다. 자연어 처리 기법인 Word2Vec을 기존 방식으로 이용하는 것이 아닌 상품 클릭 내역을 순서대로 나타낸 데이터에 적용하는 것으로 새로운 방식의 학습 방법을 제안한다. 또한, 여러 가지 머신러닝 알고리즘을 통한 결과 비교하고 최적의 알고리즘을 찾고 좋은 성능의 알고리즘을 결합한 앙상블 모델을 소개한다. 기존 인코딩되어 각 상품의 정보를 고유의 번호로 나타낸 데이터는 분류모델을 사용하지만, Word2Vec을 이용해 상품 간의 연관성을 수치로 표현함으로써 회귀모델이 사용 가능해진 이후의 추천 정확도 차이를 비교한다. 연구 방법은 다음과 같이 5단계로 나타냈다.

- 1단계 - 쇼핑몰 이용 고객의 클릭 내역을 순서대로 나열하고 데이터 전처리를 진행한다. 적은 클릭 횟수를 가진 사용자와 구매할 아이템을 선택하지 않은 사용자의 데이터는 제외한다.
- 2단계 - 전처리 데이터를 기반으로 Word2Vec를 이용하여 상품 간의 유사도 특징을 추출한다. 이때 Word2Vec의 벡터 차원을 1차원부터 5차원까지 각 차원별로 상품 간 유사도를 구한다.
- 3단계 - 클릭 내역으로 기반으로 타겟 데이터인 장바구니에 넣은 아이템을 예측하며, 단일 모델과 하이브리드 모델을 이용하여 상품 추천 정확도를 비교

하고 성능이 가장 좋은 모델을 이용한다.

- 4단계 - Word2Vec 적용 전과 후의 상품 추천 정확도 추이를 비교하며 모델 성능의 개선을 확인한다. 또한 추천 상품과 유사도가 가장 비슷한 상품 5개를 함께 추천하면서 그 안에 실제 장바구니의 넣은 상품이 있다면 적중한다고 가정하는 상품 추천 적중률을 평가지표 중 하나로 비교한다.
- 5단계 - Word2Vec를 이용하여 유사도를 추출한 데이터를 기반으로 회귀모델과 분류모델의 상품 추천 정확도를 비교하여 좋은 결과를 보인 방법을 제안한다.

1.2.2 논문 구성

본 논문은 서론을 포함하여 총 5장으로 구성되며 내용은 다음과 같다.

제1장에서는 본 논문 주제를 선정하게 된 배경과 필요성을 기술하고 연구 목적을 설명한다. 또한 연구 방법 및 구성에 대해 기술한다. 제2장에서는 연구에 사용되는 머신러닝 알고리즘과 협업 필터링, 상품 간의 연관성을 찾아주고 수치화하는 Word2Vec에 대하여 살펴본다. 제3장에서는 제2장에서 소개한 알고리즘들을 토대로 본 연구의 추천 시스템 방법을 제안하며, 각 알고리즘이 하는 역할에 대해 설명한다. 제4장에서는 실험 환경에 대한 언급과 각 알고리즘별 성능을 평가지표를 통해 비교하며 제시한 모델을 검증한다. 제5장에서는 앞서 제시한 연구 목적과 필요성에 따라 결론을 도출하며, 본 연구의 의미와 향후 과제에 대해 언급하며 글을 끝맺는다.

Ⅱ. 이론적 배경

2.1 추천 시스템과 협업 필터링

2.1.1 추천 시스템의 개념과 종류

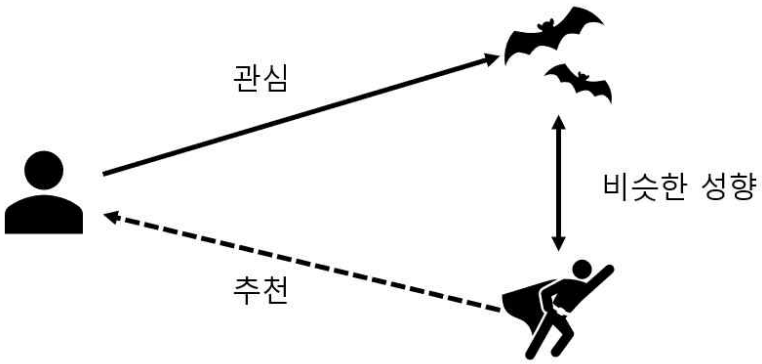
추천 시스템은 사용자가 원하는 정보를 스스로 추천해주는 시스템이다[4]. 인터넷 서비스는 사용자에게 최대한의 편의를 제공하려고 노력하며 발전함에 따라 인터넷 서비스 사용자들은 계속 발전하는 서비스를 이용하면서도 끊임없이 편의를 요구한다. 추천 시스템은 인터넷 서비스 사용자의 만족을 충족시켜주는 기본적이면서도 중요한 역할을 한다[5].

추천 시스템의 종류에는 여러 가지가 있지만 대표적인 추천 시스템 종류는 콘텐츠 기반 추천 시스템(contents-based recommender system), 협업 필터링(collaborative filtering), 추천 시스템 중 2개 이상의 알고리즘을 결합한 하이브리드 추천 시스템(hybrid recommender system) 등으로 나뉜다.

2.1.2 콘텐츠 기반 추천 시스템(Contents-based Recommender System)

콘텐츠 기반 추천 시스템은 비슷한 콘텐츠 성향을 가진 유사한 상품을 추천하는 방식이다. 예를 들어, 그림 2-1에서는 기존에 배트맨과 관련된 상품을 좋아했던 고객에게 비슷한 콘텐츠 성향을 가진 슈퍼맨과 관련된 상품을 추천해주는 방식이다[6]. 상품을 콘텐츠 기반으로 파트를 나누고 그 안에 있는 상품을 추천한다. 콘텐츠 기반 추천 시스템은 상품 간의 연관성을 상품 성향에 따라 나누며 새로 추가된 상품도 콘텐츠 기반으로 나뉘기 때문에 콜드 스타트(cold start) 문제를 해결할 수 있다. 콜드 스타트는 추천 시스템에서 새로운 상품이나 쇼핑몰 이용자로 인해 일어나는 문제로서 기존 데이터를 기반으로 추천하는 방식에서는

새로운 정보를 받아드리지 못하고 전혀 관계없는 상품을 추천하는 경우가 생길 수 있다[7]. 콘텐츠 기반 추천 시스템은 상품 간의 연관성을 다루지만 쇼핑몰 이용자들이 무엇을 선호하고 구매하는지에 대한 정보는 다루지 않는다. 즉, 구매 트렌드를 따라가지 못하기 때문에 유저의 성향을 기반으로 다루는 협업 필터링보다 추천 정확도 면에서 떨어질 수 있다.



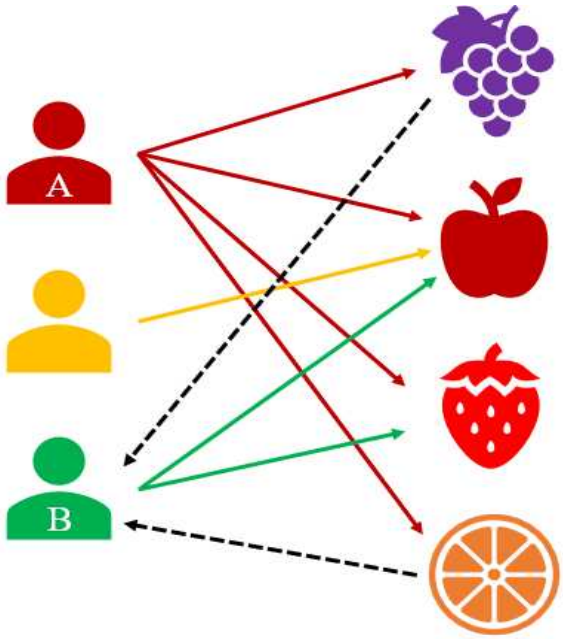
[그림 2-1] 콘텐츠 기반 필터링 예시

2.1.3 협업 필터링(Collaborative Filtering)

협업 필터링은 상업적으로 성공한 추천 시스템으로 알고리즘은 단순하지만 높은 성능을 보인다. 비슷한 성향이나 취향을 가진 고객들이 좋아한 상품을 이용 고객에게 추천해주는 방식이다. 협업 필터링의 같은 A상품과 B상품을 선호하는 고객들에게 그 중 한 고객이 선호한 또 다른 상품 C를 추천해주는 방식으로 비슷한 성향을 가진 고객은 선호하는 상품도 같을 것이다 라는 가정으로 이루어진 추천 시스템이다[8]. 이는 소비시장의 트렌드를 반영할 수 있으며 상업에 최적화된 추천 알고리즘이라고 할 수 있다. 하지만 2.1.2장에서 언급한 것과 같이 새로운 데이터가 들어오게 되면 해당 패턴을 인식하지 못하는 콜드 스타트 현상이 일어날 수 있다는 단점을 가지고 있다. 하이브리드 추천 시스템은 협업 필터링의

소비 트렌드를 반영하면서, 콜드 스타트를 해결할 수 있다.

그림 2-2은 협업 필터링의 예시이다. A는 포도, 사과, 딸기, 오렌지를 선호할 때, B는 사과, 딸기를 선호한다. 두 사람은 함께 사과와 딸기를 선호하기 때문에 두 사람의 소비 성향이 비슷하다는 가정을 한다. 그리고 B에게 A가 선호했던 다른 상품인 포도와 오렌지를 추천해주는 방식이 협업 필터링이다[9].



〔그림 2-2〕 협업 필터링 예시

2.1.4 하이브리드 추천 시스템(Hybrid Recommender System)

하이브리드 추천 시스템은 두 개 이상의 추천 시스템을 결합한 알고리즘으로 본 연구에서는 콘텐츠 기반 필터링과 협업 필터링이 결합된 알고리즘으로써 하이브리드 추천 시스템이라고 칭한다. 콘텐츠 기반 필터링의 단점인 소비 트렌드에 따른 추천이 이루어지지 못한다는 것과 협업 필터링의 문제점인 콜드 스타트 현상을 해결하는 방안으로 하이브리드 추천 시스템이 제안됐다. 기존 협업 필터

링에 각 상품 간의 연관성을 수치로 표현한 데이터를 사용함으로써 협업 필터링과 콘텐츠 기반 필터링의 단점을 모두 보완한 방법이다[10].

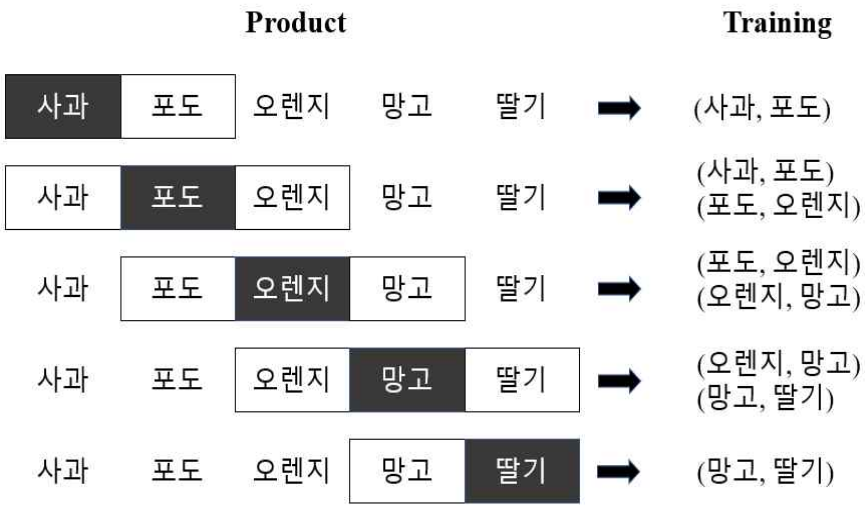
2.2 Word2Vec

Word2Vec은 본래 자연어 처리를 위해 쓰인 기법이다. Word2Vec은 One-hot 인코딩의 단점을 보완하기 위해 고안된 모델이다. One-hot 인코딩과 Word2Vec의 공통점은 각 단어를 벡터로 표현한다는 것이다. One-hot 인코딩은 각 단어들이 표현하고자 하는 단어의 인덱스의 값만 1이고 나머지 인덱스는 전부 0으로 표현하는 희소 표현(sparse representation)이다. 이러한 벡터 표현 방식은 단어 간의 유사성을 표현하지 못해 분산 표현(distributed representation) 방식을 고안했다[11]. 분산 표현 방식은 비슷한 위치에 등장하는 단어들은 비슷한 의미를 가진다 라는 가정으로 유사성을 찾는다. 각 차원은 단어들을 표현할 방법이고, 표현할 방법의 개수만큼 벡터의 차원이 존재한다. 즉 차원이 클수록 단어들을 벡터의 값으로 더욱 명확하게 나타낼 수 있다.

Word2Vec는 Continuous Bag of Words(CBOW)와 Skip-Gram, 두 가지 방식으로 표현이 가능하다. CBOW는 주변에 있는 단어들을 기반으로 중심에 있는 단어를, Skip-Gram은 중심단어를 기반으로 주변 단어들을 예측한다[12]. 예를 들어 Word2Vec이 각 단어들을 벡터 수치로 치환할 때 CBOW의 경우 주변 단어들의 벡터 값을 기반으로 가장 가까운 벡터 값을 중심 단어로 예측한다. 반대로 Skip-Gram은 중심 단어의 벡터 값을 기반으로 가까운 벡터 값을 주변 단어로 예측하는 방식이다[13].

그림 2-2는 Word2Vec이 단어 간의 유사도 특징을 추출하는 과정으로 본 연구에서는 각 단어가 상품이 된다. Word2Vec은 단어를 학습시키면서 현재 타겟이 되는 단어 주변의 단어들과 묶인다. 어떤 단어와 많이 묶였는지가 벡터 상의 유사도로 가깝게 나타난다. 또한 Word2Vec의 파라미터인 윈도우 사이즈를 설정하여 단어를 몇 개씩 묶어 유사도를 추출할 것인지 설정할 수 있다. 그림 2-3에

서는 윈도우 사이즈 1을 설정하여 중심단어와 주변 양쪽의 단어 각 1개씩을 묶어 유사도를 구한다. 본래 Word2Vec은 문장을 학습하면서 각 단어가 들어갈 자리를 학습하고 예측하는데 본 연구는 이러한 이론적 원리를 쇼핑 내역에 적용함으로써 유사도를 추출해낸다.



〔그림 2-3〕 Word2Vec 유사도 추출 과정

2.3 머신러닝 알고리즘

2.3.1 분류와 회귀

머신러닝의 분류모델과 회귀모델은 학습 데이터를 학습시킨 모델에 테스트할 데이터를 입력했을 때 예측한 값을 나타내는 공통점이 있다. 하지만 분류모델과 회귀모델의 쓰임에는 차이가 있다. 우선 분류모델은 주어진 피처에 따라 데이터를 클래스로 분류하는 방법이다. 예를 들면 스팸 메일을 구분할 때 분류모델의 이진 분류는 스팸 메일의 데이터를 피처로 입력 받고 스팸 메일 여부를 0과 1의 이진법으로 분류하여 예측한다. 다음으로 회귀모델은 주어진 피처에 따라 수치로 표현된 타겟 값을 추정하는 방법이다[14]. 예를 들어 주식의 가격을 예측할 때 이전 주식의 추세에 따른 시계열 데이터를 피처로 입력 받고 주식의 가격을 예측하는 방법이다. 머신러닝을 통한 데이터 학습시간은 대체적으로 회귀모델이 분류모델보다 빠르다는 장점이 있다.

2.3.2 데이터의 수와 머신러닝

머신러닝에서는 데이터의 수가 결과에 영향을 미친다. 데이터가 적다면 원하는 결과의 정확도가 나오기 힘들고 데이터가 많다면 이상적인 결과를 나타낼 수 있다. 하지만 데이터가 많다고 하더라도 꼭 좋은 결과가 나오는 것은 아니다. 데이터 서로 균형을 이뤄야 하고 명확하게 특징으로써 사용이 되어야 결과에 좋은 영향을 준다. 또한 데이터의 행(row)의 개수와 열(column)의 개수의 균형이 맞아야 좋은 결과를 얻을 수 있다. 데이터의 행의 개수가 열의 개수에 비해 적으면, 많은 열의 개수로 인해 과하게 학습이 되는 과적합(overfitting)이 일어날 수 있다. 반대로 데이터의 행의 개수가 열의 개수에 비해 많으면, 비교적 적은 열의 개수로 인해 학습이 덜 되는 과소적합(underfitting)이 일어날 수 있다[15].

2.3.3 머신러닝 학습 방법

머신러닝의 학습 단계에서는 입력 데이터와 타겟 데이터를 학습한다. 입력 데이터는 보통 X 데이터라 표현하고 본 연구에서는 상품을 구매하기 전 상품 클릭 내역을 의미한다. 타겟 데이터는 Y 데이터라 표현하고 구매에 관심 있어 장바구니에 넣은 상품으로 인식한다. 학습 단계에서는 입력 데이터인 각 사용자의 상품 클릭 내역과 해당 클릭 내역을 보였을 때 장바구니에 넣은 상품을 학습한다. 한 사용자의 클릭 내역은 하나의 쇼핑 패턴으로 인식한다[16].

머신러닝의 테스트 단계에서는 새로운 데이터가 입력으로 들어왔을 때 앞서 학습한 데이터를 기반으로 예측한 결과를 타겟 데이터로 출력한다. 본 연구에서의 테스트 단계는 새로운 클릭 내역이 입력으로 들어왔을 때 앞서 학습한 데이터를 기반으로 장바구니에 넣은 상품을 예측한다. 새로운 클릭 내역과 가장 유사한 클릭 내역의 장바구니에 넣은 상품을 추천해주는 형태이다. 즉 쇼핑하는 단계에서 클릭한 내역은 쇼핑 패턴이고 새롭게 입력으로 들어온 데이터와 가장 유사한 쇼핑 패턴의 장바구니에 넣은 상품을 추천한다.

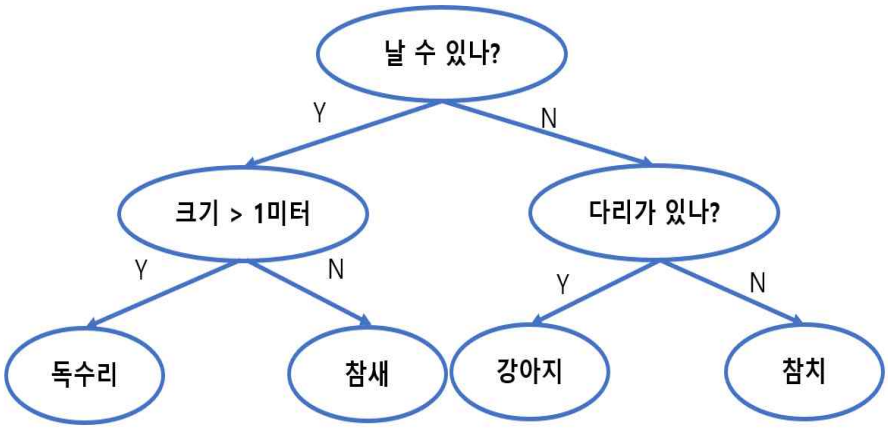
2.4 머신러닝 모델

본 연구는 Decision Tree, Random Forest, LGBM, XGBoost, Extra Trees 5개의 단일 머신러닝 알고리즘과 Random Forest와 XGBoost, Extra Trees를 결합한 하이브리드 모델(hybrid Model)을 사용하여 결과를 비교하였다.

2.4.1 결정트리(Decision Tree)

결정트리는 그림 2-4과 같이 각 노드에서 나타내는 질문에 따라 결과를 결정하는 방식이다. 예를 들어, 독수리, 참새, 강아지, 참치를 구분한다고 할 때 '날 수 있나?' 라는 질문을 통해 독수리와 참새 그룹과 강아지와 참치 그룹으로 나눌

수 있다. 이후 날 수 있다면 크기에 따라, 날 수 없다면 다리가 있는 지 여부에 따라 결정트리를 통해 결과를 도출해낼 수 있다[17].

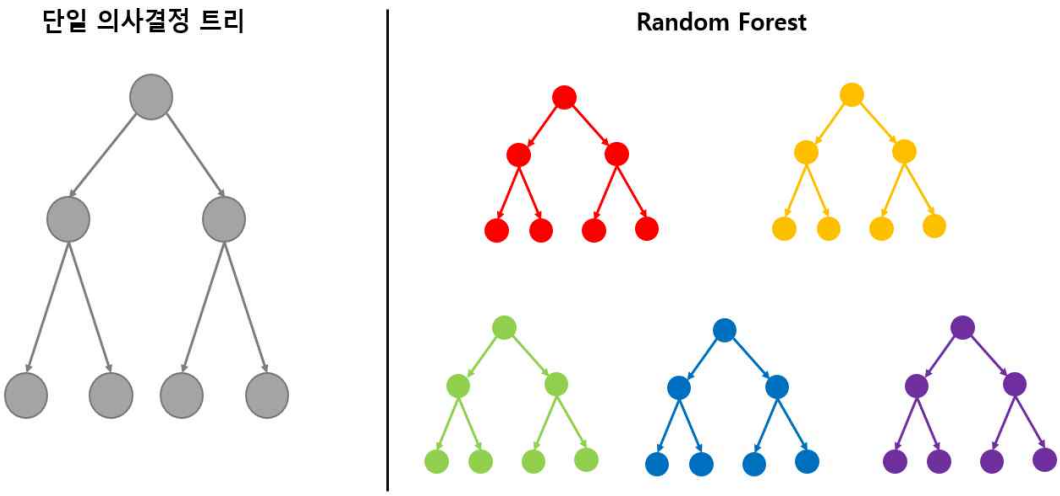


〔그림 2-4〕 결정트리 학습 방식

2.4.2 Random Forest

Random Forest는 여러 개의 결정트리를 통해 의사결정을 하는 알고리즘으로 1개의 똑똑한 알고리즘보다 100개의 평범한 알고리즘이 문제해결을 잘한다는 가정으로 만들어졌다. 학습 방법으로는 여러 개의 트리가 결정한 결과들을 모아 가장 많이 나온 것을 최종 예측 값을 정한다. Random Forest는 대표적인 Bagging 방법이면서도 Voting의 방식을 나타내고 있다. Bagging은 여러 번 샘플을 뽑아 그것을 바탕으로 결과를 집계하는 방법이며, 각 샘플은 독립적으로 결과를 예측한다. Voting은 여러 개의 샘플들이 낸 결과를 이용하여 가장 많이 나온 값을 최종적으로 예측하는 방법이다[18].

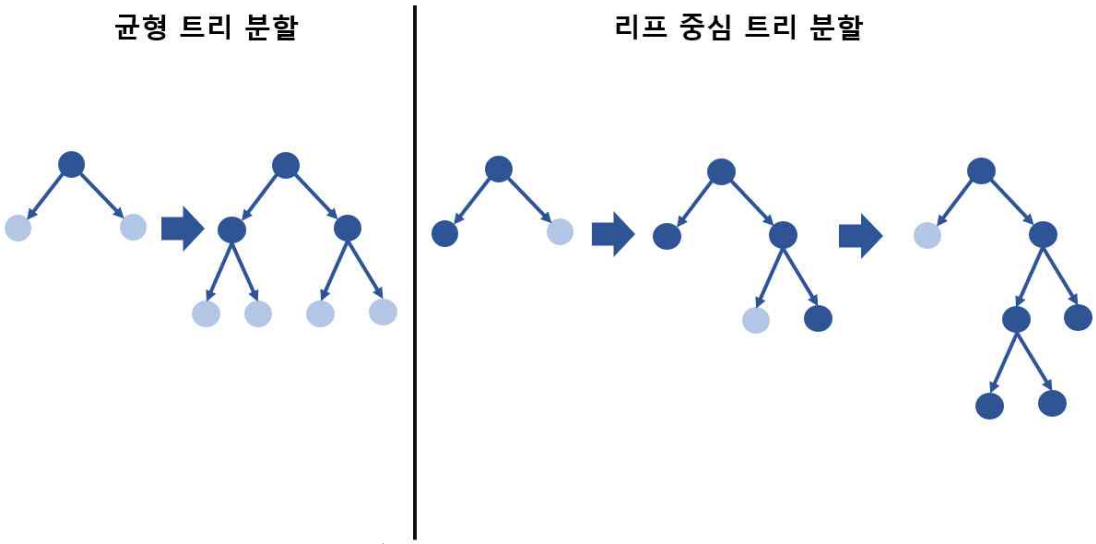
그림 2-5는 Random Forest의 학습 방식 예시로 하나의 단일 의사결정 트리를 이용하는 것이 아닌 프로그래머가 설정한 의사결정 트리의 개수만큼 학습을 진행하고 나온 결과들을 모아 가장 많이 언급된 결과를 최종 예측 값으로 출력한다.



〔그림 2-5〕 Random Forest 학습 방식

2.4.3 Light Gradient Boosting Model (LGBM)

Light Gradient Boosting Model은 XGBoost와 같이 Gradient Boosting을 기반으로 하고 있다. 다만 대부분의 Gradient Boosting 알고리즘은 트리를 그림 2-6의 균형 트리 분할과 같이 결정트리가 균형을 유지하며 리프 노드를 분할한다. 트리의 깊이는 최소화할 수 있지만 트리의 균형을 유지하기 위해 맞추는 시간이 필요하다. 즉, 학습에 소비되는 시간은 많지만 과적합이 상대적으로 덜 일어날 수 있다. 반면, LGBM에서 사용하는 알고리즘은 리프 중심 트리 분할로써 트리의 균형과 상관없이 최대 손실 값(max data loss)를 가지는 리프 노드를 계속해서 분할하며, 트리의 깊이 깊어지고 불균형한 트리가 생성된다. 이는 균형 트리 분할보다 상대적으로 학습 속도가 빠르고 더 많은 손실을 줄일 수 있으나 트리가 깊어지면서 과적합이 상대적으로 쉽게 일어난다[19].

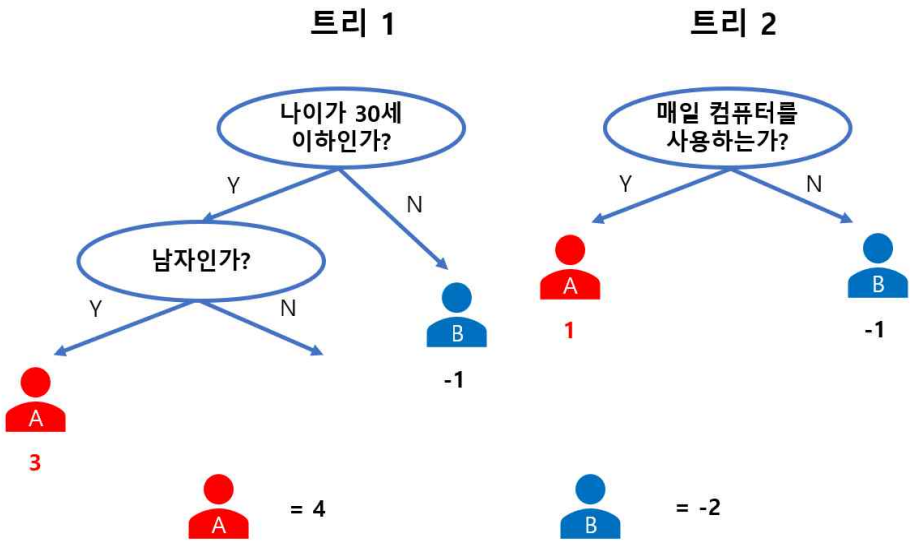


〔그림 2-6〕 LGBM 학습 방식

2.4.4 Extreme Gradient Boosting(XGBoost)

XGBoost는 Gradient Boosting 기법으로 각 샘플이 독립적으로 결과를 예측하는 Bagging 과는 반대로 첫 샘플에서 낸 결과의 가중치를 다음 샘플에서도 반영하는 방식이다. 이전 샘플의 결과 가중치를 계속해서 다음 샘플에도 영향을 주는 식으로 학습한다. XGBoost는 다른 Gradient Boosting을 기반으로 한 모델들보다 학습 속도가 빠르고 모델의 성능이 좋다. Gradient Boosting은 학습 데이터에 결과에만 집중하여 과적합이 쉽게 일어나는데 XGBoost는 자신들이 제공하는 하이퍼 파라미터 값들을 조절하는 것으로 프로그래머가 원하는 학습 방식을 설정할 수 있어 과적합을 방지할 수 있다.

그림 2-7은 XGBoost의 학습 방식 예시로 컴퓨터 게임을 좋아하는지에 대한 예측을 할 때 각 트리의 가중치를 계산하는 과정이다. 트리 1에서는 A가 30세보다 어리면서 남자이므로 가중치 값을 높게 줬으며, 트리 2에서도 A가 매일 컴퓨터를 사용하는 것에 가중치 값을 높게 줬다. B는 두 개의 트리가 결정하는 사항에 반대이므로 가중치 값을 낮게 준다. 두 개의 트리를 기반으로 컴퓨터 게임을 좋아하는 사람을 예측할 때 주어진 가중치 값을 기반으로 가장 높은 가중치 값을 가진 사람을 예측하는 원리이다[20].



〔그림 2-7〕 XGBoost 학습 방식

2.4.5 Extra Trees

Extra Trees는 Random Forest와 학습하는 방식이 비슷하다. Random Forest는 모든 피처(feature) 값을 사용하여 결과를 내는 반면, Extra Trees는 여러 개의 결정트리가 무작위로 피처를 선정하여 최적의 결과를 내는 방식을 선택한다[21]. 그렇기에 모든 피처 값을 사용하는 Random Forest보다 일부의 피처를 무작위로 선정하는 Extra Trees의 학습 속도가 더 빠르다.

2.4.6 하이브리드 머신러닝 모델(Hybrid Machine Learning Model)

하이브리드 머신러닝 모델은 학습 모델은 2개 이상 결합한 모델로 머신러닝을 이용한 경진대회에서 많이 쓰이는 앙상블 기법이다. 본 연구에서는 Random Forest, XGBoost, Extra Trees를 결합한 하이브리드 모델을 이용한다. 하이브리드 모델은 단일 학습 모델의 문제점을 보완하는 방식이다. 예를 들어, 앞서 설명한 하이브리드 추천 시스템은 협업 필터링의 콜드 스타트와 콘텐츠 기반 필터링

의 소비 트렌드를 따라가지 못한다는 문제점을 보완하는 방식으로 제안됐다. 하이브리드 모델 또한 Random Forest의 매 학습마다 결과 폭의 차이가 크다는 문제점과 Gradient Boost의 과적합 문제를 보완한다.

2.5 관련 연구

2.5.1 협업 필터링을 이용한 추천 시스템

해당 논문은 평점을 기반으로 사용자 기반 협업 필터링을 이용한 추천 시스템을 제안한다. 사용자 기반 협업 필터링 이전의 기술들은 사용자의 성향이나 기호를 기반으로 다른 사용자에게 적합한 상품을 추천해주는 것이 아닌 전체 사용자들에게 똑같이 구매율이 높은 상품만을 추천했다. 사용자 기반 협업 필터링을 이용함으로써 사용자 개개인에게 적합한 상품을 추천해주는 시스템을 해당 논문에서는 제안하고 있다. 본 논문도 협업 필터링을 기반으로 추천 시스템을 제안했다. 그러나 상품 간의 유사도를 이용하여 머신러닝을 통해 추천된 상품과 비슷한 성향의 상품을 5개 추천해줌으로써 새로운 쇼핑패턴의 데이터가 들어오면 인식하지 못하는 협업 필터링의 콜드 스타트를 보완했다[22].

2.5.2 머신러닝의 분류모델을 이용한 추천 시스템

해당 논문은 머신러닝을 이용하여 온라인 쇼핑 시장에서 고객의 구매패턴을 파악하고 고객이 관심 있는 상품을 추천하는 연구방법을 제안한다. 대부분의 추천 시스템에서 사용하는 머신러닝의 학습 방식은 분류모델이다. 구매패턴을 파악하기 위해서 전처리된 데이터는 각 상품의 고유번호를 나타내기 때문에 고유번호의 수치가 아닌 클래스를 예측하는 분류를 이용하는 것이 일반적이다. 그러나 본 논문은 Word2Vec을 이용하여 각 상품 간의 유사도를 수치로 표현했고, 회귀모델을 통해 수치를 예측하고 나온 벡터 값과 가까운 상품을 추천해주는 것으로

분류모델 뿐만 아니라 회귀모델까지 사용이 가능하다. Word2Vec의 3차원 미만으로는 분류모델의 성능이 회귀모델을 앞서지만 벡터 차원이 높아지면서 상품을 나타내는 벡터 값의 정보가 뚜렷해지면서 Word2Vec 3차원 이상에서 추천 정확도는 회귀모델이 더 높았다. 또한 학습에 걸리는 시간에서도 클래스를 하나하나 분석하며 분류하는 모델이 수치를 이용하는 회귀모델보다 오래 걸렸으며, 회귀모델을 추천 시스템에 이용하는 것으로 추천 정확도와 학습 시간 모두 보완했다[23].

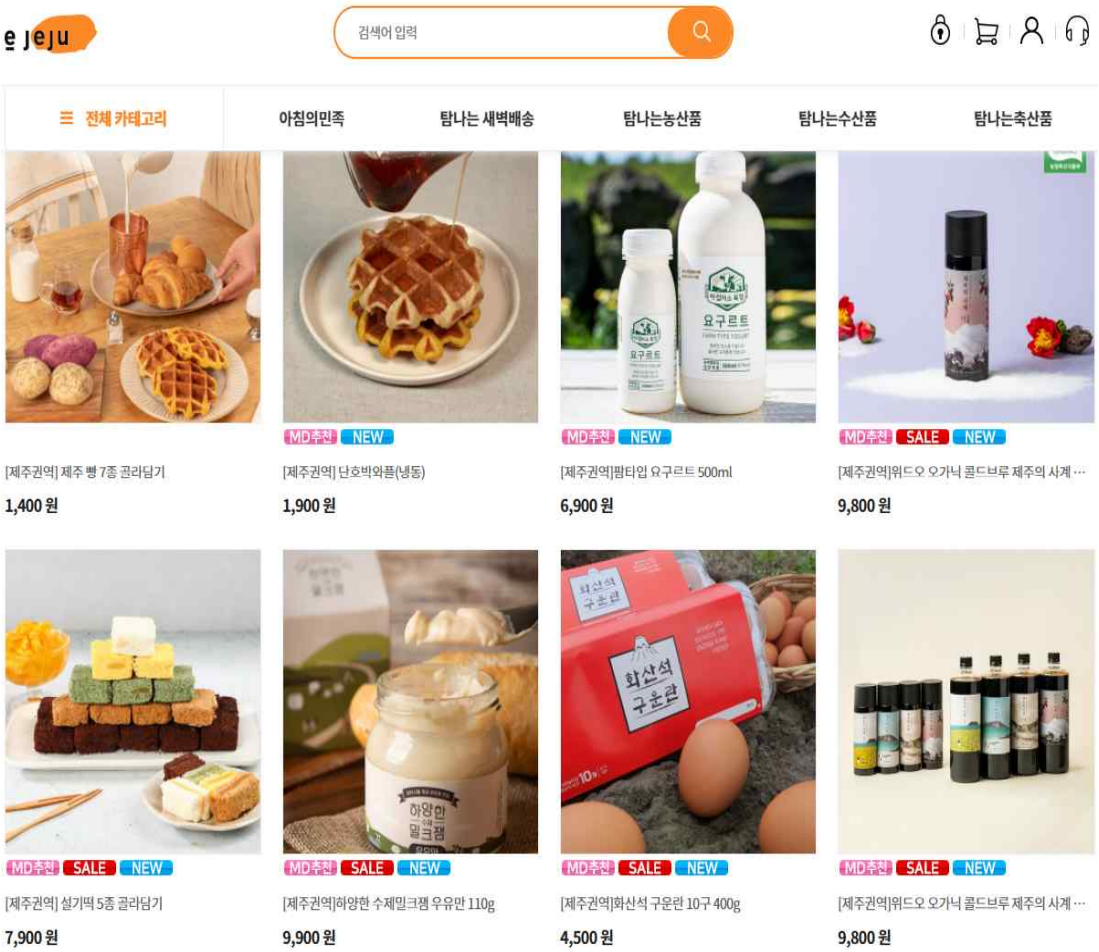
2.5.3 상품 간의 유사도를 이용안 추천 시스템

해당 논문은 Word2Vec를 기반으로 만들어진 Item2Vec을 이용한 하이브리드 협업 필터링에 대한 논문이다. Word2Vec의 단어를 하나의 상품으로 인지하고 상품 간의 유사도를 구한다. 이후 사용자 기반 협업 필터링과 아이템 기반 협업 필터링을 통해 유사도가 가까운 상품을 추천한다. 그러나 본 논문은 단순히 상품 간의 유사도만으로 추천하는 것이 아닌 Word2Vec을 이용하여 상품 간 유사도를 추출한 데이터에 머신러닝을 사용했다. 사용자들의 쇼핑패턴을 학습하고 그것을 기반으로 예측한 상품과 유사도가 비슷한 상품 5개를 함께 추천하는 것으로 상품 추천 정확도를 증가시켰다[24].

Ⅲ. 제안하는 방법

3.1 실험 데이터

표 3-1은 본 연구에서 사용한 데이터에 대한 표이다. 50일간 제주 A몰에서 온라인 쇼핑몰을 이용한 10000명의 사용자의 데이터이다. 그림 3-1은 실제 제주 A몰의 홈페이지이다. 각 유저가 클릭한 상품의 내역을 시간의 순서대로 가로로 나열했다. 즉 한 개의 행은 한 사용자의 쇼핑 내역이다.



〔그림 3-1〕 A몰 홈페이지

데이터 출처	A몰
데이터 수집 기간	50일
총 데이터 개수	10,000개
학습 데이터	80%
테스트 데이터	20%

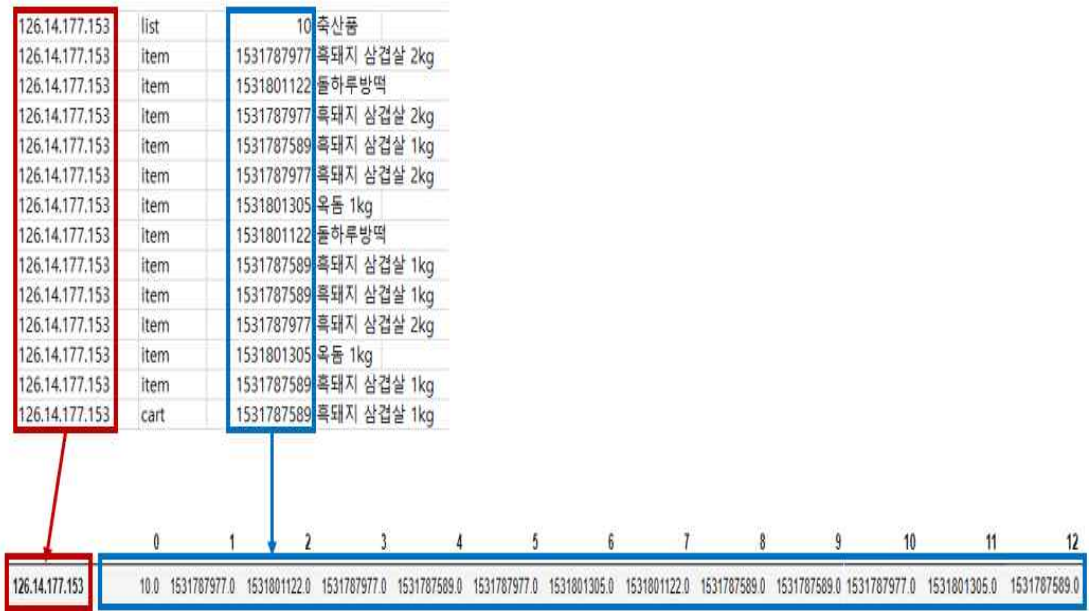
〈표 3-1〉 실험 데이터

실험에 이용한 데이터는 총 10,000개이며 이 중 80%는 학습 데이터로, 20%는 테스트 데이터로 이용했다.

그림 3-2는 데이터 수집 단계에서 쌓인 실제 쇼핑 이용객들의 기록이며 각각의 사용자들이 클릭한 상품의 고유 번호를 수집한다. 데이터 타입 list는 상품을 클릭하기 전 비슷한 상품들을 모아둔 카테고리 페이지이며, item은 상품을 클릭했을 때 나타나는 데이터 타입이다. 마지막으로 cart는 본 연구에서 타겟 데이터를 의미하며, 구매에 관심이 있어 장바구니에 추가한 상품이다. 이렇게 수집한 데이터를 그림 3-3과 같이 데이터 하나당 한 사용자가 클릭한 상품들을 순서대로 나열한다. 각각의 행은 한 사용자의 쇼핑 내역을 클릭 순서에 따라 나열했다. 가장 많이 클릭한 사용자의 클릭 횟수는 13번이고, 13번의 클릭보다 많아지면 최근 클릭한 13개의 상품의 목록을 나열하고 그전 클릭한 상품은 삭제한다. 또한 가장 마지막에 클릭한 상품인 13번째 상품은 해당 사용자의 장바구니에 넣은 상품이다. 상품 클릭 횟수를 13회로 제한한 이유는 본 데이터에서 사용자의 95%가 13번 클릭 이전에 쇼핑을 마쳤다. 상품 클릭 횟수 13번 이상으로 데이터 전처리를 하게 되면 열의 수는 많아진다. 하지만 13번 이상 상품을 클릭한 사용자가 적기 때문에 나머지 95%의 사용자들의 데이터프레임은 빈값이 되므로 결과에 악영향을 끼칠 수 있다. 표 3-2는 전처리된 데이터의 샘플이다.

access_ip	access_date	access_time	item_type	item_id	item_name
188.244.165.181	2/11/2020	17:45:46	list	12	VERSACE
210.54.66.88	2/11/2020	17:45:46	list	30	GUCCI 남성
210.54.66.88	2/11/2020	17:45:57	item	580807562	Web 디테일의 소프트 GG 수프림 더플 백
210.54.66.88	2/11/2020	17:45:58	item	557537434	오버사이즈 인터로킹 G 코튼 티셔츠
210.54.66.88	2/11/2020	17:46:02	item	368671191	가죽 레이스업
210.54.66.88	2/11/2020	17:46:04	item	441575980	별 모양의 Web 프린트 울 스톨
210.54.66.88	2/11/2020	17:46:05	item	307998406	돌하르방 가죽 벨트
210.54.66.88	2/11/2020	17:46:09	item	594381494	심볼 모티브 실크 타이
210.54.66.88	2/11/2020	17:46:10	item	587473611	[에이스] GG 울 스니커즈
210.54.66.88	2/11/2020	17:46:11	item	445940038	[락이톤] 구찌 로고 락이톤 가죽 스니커즈
210.54.66.88	2/11/2020	17:46:14	item	445785117	[락이톤] 구찌 락이톤 가죽 스니커즈
210.54.66.88	2/11/2020	17:46:16	item	384274798	돔베고기
210.54.66.88	2/11/2020	17:46:20	item	222584378	[르 마르셰 드 메르베이(Le Marche des Merveilles)] 38mm 시계
210.54.66.88	2/11/2020	17:46:24	item	445938213	구찌 로고 가죽 카드 케이스
62.47.31.96	2/11/2020	17:47:00	list	12	VERSACE
188.244.165.181	2/11/2020	17:45:46	list	12	VERSACE
88.189.204.99	2/19/2020	16:59:23	item	1543286117	CU상품권
95.90.201.200	2/11/2020	17:47:00	list	30	GUCCI 남성
95.90.201.200	2/11/2020	17:47:09	item	584188627	스몰 사이즈 십자가목걸이
95.90.201.200	2/11/2020	17:47:11	item	557537662	뉴욕 양키스™(New York Yankees™) 패치가 장식된 남성 코튼 조깅 팬츠
95.90.201.200	2/11/2020	17:47:16	item	557532603	[플래시트렉] 스니커즈
95.90.201.200	2/11/2020	17:47:18	item	445941449	오버사이즈 구찌 스탬프 프린트 티셔츠
95.90.201.200	2/11/2020	17:47:20	item	470981561	모나코 엠티 도트 트윌 수트
95.90.201.200	2/11/2020	17:47:23	item	454770862	GG 슈프림 Mystic Cat 백팩
95.90.201.200	2/11/2020	17:47:26	item	126728362	돌하르방 가죽 벨트
95.90.201.200	2/11/2020	17:47:30	item	400846333	웹(Web) 디테일의 퀼트 나일론 자켓
95.90.201.200	2/11/2020	17:47:33	item	504807317	GG 스트라이프 니트 스웨터
95.90.201.200	2/11/2020	17:47:34	item	307998406	돌하르방 가죽 벨트
95.90.201.200	2/11/2020	17:47:38	item	557537256	뉴욕 양키스™(New York Yankees™) 패치가 장식된 남성 가디건
95.90.201.200	2/11/2020	17:47:40	item	384274798	돔베고기
95.90.201.200	2/11/2020	17:47:41	item	557537073	흑돼지 오겹살 1kg
95.90.201.200	2/11/2020	17:47:43	item	445941434	구찌 스트라이프 코튼 조깅 팬츠
95.90.201.200	2/11/2020	17:47:44	item	384274798	돔베고기
95.90.201.200	2/11/2020	17:47:47	item	510308132	킹스네이크 프린트 GG 슈프림 카드 케이스
93.14.188.153	2/11/2020	17:47:00	list	30	GUCCI 남성
93.14.188.153	2/11/2020	17:51:24	item	481389628	벅스 버니 코튼 스웨트셔츠
93.14.188.153	2/11/2020	17:51:26	item	469978540	자수 코튼 치노
93.14.188.153	2/11/2020	17:51:29	item	445941287	구찌 스트라이프 리버서블 아세테이트 봄버
93.14.188.153	2/11/2020	17:51:31	item	348489805	벌 패턴의 실크 타이
93.14.188.153	2/11/2020	17:51:34	item	481900003	구찌 엠튼 존 바이닐 코튼 스웨트셔츠
93.14.188.153	2/11/2020	17:51:36	item	481385109	구찌 로고 탑 핸들 토트백
93.14.188.153	2/11/2020	17:51:38	item	397457141	자수 실크 캐시미어 스카프
93.14.188.153	2/11/2020	17:51:40	item	504903707	[구찌 프린트] 반달 모양의 호보백
93.14.188.153	2/11/2020	17:51:41	item	188864408	웹(Web) GG 수프림 장지갑
93.14.188.153	2/11/2020	17:51:44	item	493855114	구찌 스트라이프 러버 솔라이드 샌들
93.14.188.153	2/11/2020	17:51:45	item	397457112	자수 실크 캐시미어 스카프
93.14.188.153	2/11/2020	17:51:47	item	365098808	[구찌 다이브] 45mm 시계
93.14.188.153	2/11/2020	17:51:48	cart	365098808	[구찌 다이브] 45mm 시계

〔그림 3-2〕 원시 데이터



〔그림 3-3〕 데이터 전처리 과정

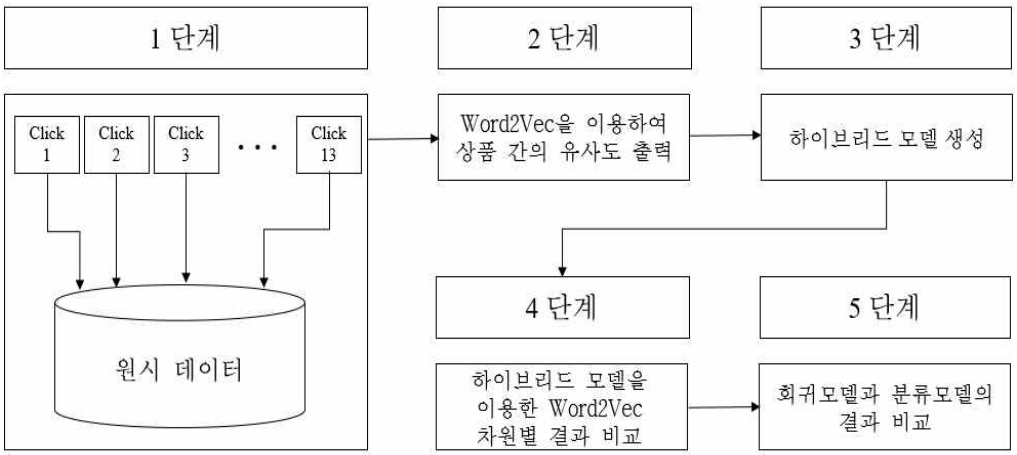
User	1번째 상품	2번째 상품	...	장바구니에 넣은 상품
95.90.201.200	481389628	469978540	...	365098808
210.54.66.88	1531787977	1531801122	...	1531787589
157.168.67.74	1547428994	1547427506	...	1547429909
197.198.238.56	397457110	587617958	...	445940920

〈표 3-2〉 전처리 데이터 결과의 예

3.2 시스템 구성도

그림 3-4는 본 연구의 시스템 구성도이다. 우선 데이터 수집 과정에서는 사용자가 상품을 클릭하면 클릭 내역 데이터를 수집한다. 수집한 클릭 내역 데이터를 각 사용자를 각 행으로 배치한다. 이후 행들의 순서들은 해당 사용자가 클릭한 상품을 순서대로 배치한 것이다. 다음으로 Word2Vec을 이용하여 각 상품들 간

의 연관성을 찾는다. 찾은 연관성을 기반으로 Word2Vec은 각 상품들을 벡터상의 수치로 표현한다. 마지막으로 Extra Trees, XGBoost, Random Forest, 하이브리드 모델의 추천 정확도를 비교하고 가장 좋은 결과를 보인 모델을 이용하여 이후 해당 모델만을 통해 분류모델과 회귀모델의 상품 추천 정확도를 구한다. 구매 상품의 벡터 값을 예측하는 과정까지는 같다. 하지만 분류모델과 회귀모델의 예측 값은 다르다. 분류모델은 데이터에 존재하는 상품의 고유 값을 가지고 예측한다. 회귀모델은 데이터를 기반으로 벡터 값의 수치를 예측한다는 점이 분류모델과 다르다. 분류모델은 예측한 값을 실제 값과 비교하여 정확도를 구한다. 회귀모델은 예측한 벡터 값과 가장 가까운 상품의 벡터 값을 구한 다음, 실제 값과 비교하여 정확도를 구한다. 본 논문에서는 벡터 상에서 가까운 상품을 구할 때 유클리드 거리 공식을 이용한다.



〔그림 3-4〕 시스템 구성도

3.3 Word2Vec을 이용한 상품 간의 유사도 특징 추출

본 연구에서 Word2Vec이 하는 역할은 각 데이터 프레임 즉 각 상품 간의 연관성을 찾고 벡터 상의 수치로 표현한다. 보통 Word2Vec은 자연어 처리에 사용하지만 본 연구는 자연어 처리가 주가 아니다. 데이터 전처리 과정에서 Word2Vec의 Skip-Gram을 이용한다. Skip-Gram은 시간의 흐름에 따라 상품을 클릭한 데이터에 중심 상품과 주변 상품을 학습한다. 클릭 순서가 가까운 상품들 간의 연관성을 찾고 벡터 상의 수치로 표현한다. 현재 내가 보고 있는 상품과 다음에 볼 상품은 서로 비슷한 선호도를 가지고 있다고 생각하는 방식이다. 보통 사용자들이 쇼핑하는 과정에서 비슷한 선호도의 상품을 클릭하며 자신이 원하는 상품을 구매하는 경우가 많다. 물론 전혀 상관없는 상품을 클릭하는 경우도 있지만 그러한 경우는 적기 때문에 Word2Vec은 자주 중복되는 상품일수록 비슷한 선호도의 상품이라고 인식한다. Word2Vec을 이용하여 나온 벡터상의 수치가 서로 가까운 Word들은 관련성이 크다. Word2Vec을 사용하여 구매할 것이라 예측한 상품의 추천 정확도가 증가하는 것을 확인했다. 사용자가 클릭한 상품을 순서대로 배치한 것이다. 다음으로 Word2Vec을 이용하여 각 상품들 간의 연관성을 찾는다. 찾은 연관성을 기반으로 Word2Vec은 각 상품들을 벡터상의 수치로 그림 3-5와 같이 표현한다.

앞서 설명했듯이 Word2Vec은 각 데이터 프레임 간의 연관성을 찾고 벡터 상의 수치로 표현한다. 해당 벡터 차원은 프로그래머가 직접 설정할 수 있으며, 본 연구에서는 Word2Vec의 벡터 차원을 1차원부터 5차원까지 한 차원씩 증가시켜 차원별 추천 정확도를 구했다. 벡터 차원이 증가할수록 각 상품들 간의 연관성이 뚜렷해지며, 그로 인해 생기는 정확도의 변화를 나타냈다.

0	1	2	3	4	5	6	7	8	9	10	11	12
-0.699768	-0.761709	-0.791931	-0.761709	-0.783514	-0.761709	-0.778654	-0.791931	-0.783514	-0.783514	-0.761709	-0.778654	-0.783514
-0.454276	-0.530428	-0.530428	-0.541286	-0.530428	-0.512369	-0.541286	-0.541286	-0.512369	-0.512369	-0.512369	-0.512369	-0.530428
-3.167422	-1.492999	-1.364949	-1.757407	-2.768543	-1.323182	-2.154332	-2.047143	-2.211164	-1.640382	-1.822567	-1.685993	-1.903447
-3.167422	-1.686950	-1.819322	-1.703870	-1.216967	-1.424027	-2.852783	-2.852783	-1.626338	-1.449358	-2.922199	-1.622840	-1.877541
-3.460542	-4.419822	-4.419822	-4.564969	-4.395570	-4.504427	-4.504427	-4.466349	-4.555246	-4.497240	-4.395570	-4.466349	-4.419822
...	...	...	...	...	...	...	...	...	...	...	...	...
-1.253573	-1.569481	-1.417415	-1.417415	-1.395381	-1.413267	-1.377896	-1.377896	-1.417415	-1.413267	-1.391864	-1.417415	-1.413267
-3.460542	-4.466349	-4.504427	-4.504427	-4.419822	-4.564969	-4.419822	-4.555246	-4.564969	-4.497240	-4.419822	-4.497240	-4.419822
-0.454276	-0.512369	-0.530428	-0.541286	-0.530428	-0.530428	-0.530428	-0.530428	-0.541286	-0.541286	-0.541286	-0.530428	-0.530428
-1.253573	-1.417415	-1.410077	-1.413267	-1.410077	-1.569481	-1.569481	-1.377896	-1.413267	-1.395381	-1.417415	-1.569481	-1.413267
-1.012874	-1.111053	-1.072266	-1.072266	-1.110131	-1.111053	-1.072266	-1.083733	-1.116041	-1.115622	-1.116041	-1.111053	-1.111053

〔그림 3-5〕 Word2Vec을 이용한 유사도 추출 결과의 예

3.4 머신러닝을 이용한 상품 추천 시스템

머신러닝의 분류모델과 회귀모델을 이용하여 상품 추천 정확도를 구한다. 본 논문의 데이터는 모두 다중 클래스, 다중 타겟 데이터이므로 Sklearn에서 제공하는 Multi output 라이브러리를 사용한다[25]. 이후 구매 상품의 벡터 값을 예측하는 과정까지는 같다. 하지만 분류모델과 회귀모델의 예측 값은 다르다. 분류모델은 데이터에 존재하는 상품의 고유 값을 가지고 예측한다. 회귀모델은 데이터를 기반으로 벡터 값의 수치를 예측한다는 점이 분류모델과 다르다. 분류모델은 예측한 값을 실제 값과 비교하여 정확도를 구한다. 회귀모델은 예측한 벡터 값과 가장 가까운 상품의 벡터 값을 구한 다음, 실제 값과 비교하여 정확도를 구한다. 본 논문에서는 벡터 상에서 가까운 상품을 구할 때 유클리드 거리 공식을 이용한다.

본 연구의 학습 과정은 세 단계로 나뉜다. 우선 Word2Vec를 이용하기 전 상품 간의 연관성이 없는 인코딩된 데이터에 분류모델을 이용하여 Random Forest, XGBoost, Extra Trees, 하이브리드 모델의 결과를 비교한다. 이후 가장 좋은 결

과를 보인 머신러닝 알고리즘을 이용하여 Word2Vec을 이용한 연관성 수치화 데이터에 Word2Vec 차원별 결과를 비교한다. 이때 분류모델과 회귀모델을 모두 이용하여 모델 성능을 비교하고 가장 결과가 좋게 나왔던 Word2Vec의 벡터 차원을 도출한다.

Ⅳ.실험 환경 및 평가지표

4.1 실험 환경

Word2Vec의 벡터 차원 증가에 따른 학습시간을 비교한다. 벡터 차원이 증가할수록 데이터의 양은 그만큼 증가하기 때문에 추천 정확도와 학습시간을 고려하여 좋은 결과의 차원을 구한다[26]. 실험 환경에 따라 학습에 걸리는 시간은 차이가 있기에 본 연구를 진행한 실험 환경에 대한 정보는 표 4-1과 같다.

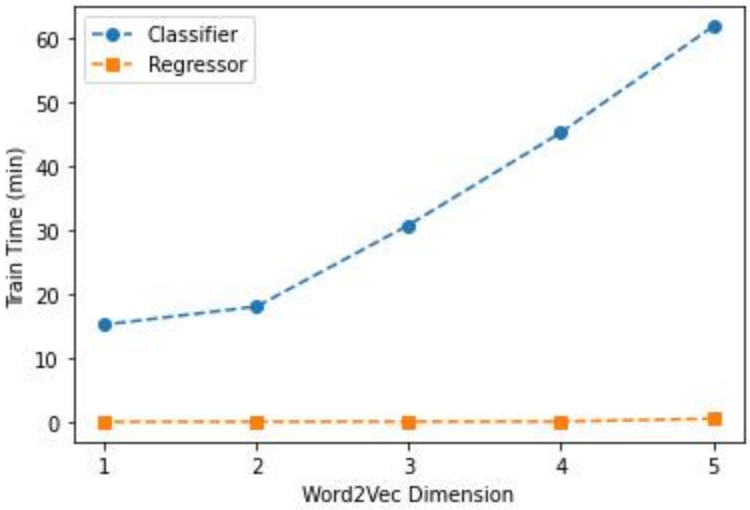
Programming language	Python 3.7.6
Operating system	Window 10 pro 64bit
Browser	Google chrome
Library and framework	Jupyter notebook
CPU	Intel(R) Core(TM) i5-9600k@3.70GHz
Memory	16GB

〈표 4-1〉 실험 환경

4.2 학습 시간

분류모델밖에 사용하지 못하는 데이터에 회귀모델을 쓸 수 있게 됐을 때의 가장 큰 장점은 학습 속도가 빠르다는 것이다. 분류모델과 회귀모델 모두 Word2Vec 벡터 차원을 1차원부터 5차원까지 증가시켜 학습시간을 비교했다. Word2Vec의 벡터 차원이 증가할수록 데이터의 수도 증가하기 때문에 학습속도 또한 점차 증가한다. 회귀모델의 학습속도는 1차원부터 5차원까지 증가하지만 5차원에서의 학습속도가 1분정도로 분류모델에 비하면 빠르다. 분류모델의 Word2Vec 벡터 차원별 학습속도는 회귀모델에 비해 크게 증가한다. 결과적으로

Word2Vec 벡터 차원이 증가할수록 학습속도가 증가하고, 회귀모델이 분류모델보다 학습속도가 빠르다는 것을 확인할 수 있다[27].



〔그림 4-1〕 회귀모델과 분류모델의 학습 시간 비교 그래프

4.3 평가지표

각 상품의 고유 벡터 값에 대한 데이터를 학습시켜 구매할 상품을 예측한다. 예측한 상품과 실제 구매상품을 비교하며 정확도를 구한다. 평가지표는 정확도(Accuracy), 정밀도(Precision), 재현율(Recall)등이 있으며 본 연구에서는 정확도를 사용하여 모델의 성능을 측정했다.

4.3.1 정확도(Accuracy)

정확도는 진실인 것을 True로, 거짓인 것을 False로 표기하고 옳게 예측한 경우를 전체 경우에서 나눈 것이다. TP는 실제 진실인 값을 True로 옳게 예측하는 경우이고 FP는 실제 거짓인 값을 True로 잘못 예측을 하는 경우이다. FN은 실제 진실인 값을 False로 잘못 예측하는 경우이고 TN은 실제 거짓인 값을 False

로 옳게 예측하는 경우이다. 분류모델의 가장 직관적인 평가 지표이다. 본 연구에서는 클릭 내역을 기반으로 구매할 상품을 예측했을 때 실제 구매상품과 맞는 경우에 전체 경우를 나누어서 정확도를 구했다[28].

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN}$$

4.3.2 정밀도(Precision)

정밀도은 모델이 True라고 분류한 것 중에서 실제 진실인 것의 비율이다. PPV(Positive Predictive Value)라고도 불린다. 클릭 내역이 입력으로 들어왔을 때 분류모델이 클릭 내역을 기반으로 어떤 상품을 구매할 지 예측하는 척도이다[29].

$$\mathrm{Pr}ecision = \frac{TP}{TP + FP}$$

4.3.3 재현율(Recall)

재현율은 정밀도와 반대로 실제 진실인 것 중에서 모델이 True라고 예측한 것의 비율이다. 스팸 메일을 예로 들면 실제 스팸 메일이 맞는 것을 모델이 True라고 예측한 비율이다. 이 경우 어떤 요소에 의해 확실히 구매할 상품을 예측할 수 있다면 구매할 상품만 예측할 수 있다. 구매할 상품이 확실하지 않으면 아예 예측을 하지 않고 보류하여 FP의 경우의 수를 줄여 정밀도를 극도로 끌어올릴 수 있다는 문제점이 있다. 따라서 정밀도와 재현율의 지표가 모두 높은 모델일수록 옳게 예측이 되고 있다는 것이다.

$$Recall = \frac{TP}{TP + FN}$$

4.4 적중률(Hit Ratio)

실제 추천 서비스를 이용할 경우에 한 개의 상품만을 추천해주는 것이 아니라 여러 개의 상품을 추천한다. 보통 비슷한 연관성의 상품을 여러 개 추천했을 때 사용자은 만족감을 느낀다. 대표적으로 넷플릭스와 아마존이 사용자가 선호할 만한 여러 개의 상품을 추천한다. 본 논문에서는 예측한 상품과 가까운 N개의 상품을 추천한다. 가까운 N개의 상품 안에 실제 구매 상품이 있을 경우 적중한다고 가정한다. 분류모델과 회귀모델 모두 N이 1과 5일 때 추천 정확도를 구했다. 벡터 상에서 가까운 상품들을 구할 때 유클리드 거리 공식을 이용했다. 분류모델은 N이 1인 경우에 예측한 값 그대로 나타난다.

V.실험 결과

5.1 실험 결과

5.1.1 Word2Vec 적용 전 분류모델별 결과

본 연구는 상품을 구매하기 전 클릭한 상품을 기반으로 구매 상품을 예측한다. 클릭한 상품에 대한 정보는 상품의 고유번호로 입력이 되기 때문에 머신러닝의 분류모델을 이용하여 예측하는 것이 일반적이다. 모든 학습 결과는 학습 데이터와 테스트 데이터를 다르게 설정하여 10번의 학습을 진행하였을 때 추천 정확도의 평균을 나타낸 것이다. Word2Vec를 적용하기 전 분류모델별 학습 결과를 표 5-1에서 보여준다[30]. 우선 Random Forest 분류모델의 추천 정확도는 79.87%의 결과를 보였으며, XGBoost 분류모델은 81.50%, Extra Trees 분류모델은 82.16%, Decision Tree 분류모델은 73.25%, LGBM 분류모델은 77.38%의 정확도를 보였다. 단일 모델 중 가장 결과가 좋은 모델은 Extra Trees의 분류모델이었으며, 추천 정확도가 높았던 단일 모델 3개를 결합한 하이브리드 모델이 84.23%의 정확도로 단일 모델의 성능보다 높았다. 본 연구는 가장 좋은 결과를 보였던 하이브리드 모델을 기본 모델로 학습을 진행하였으며, 이어서 Word2Vec 벡터 차원별 분류모델의 결과를 보여준다.

머신러닝 모델	정확도
Random Forest	79.87%
XGBoost	81.50%
Extra Trees	82.16%
Decision Tree	73.25%
LGBM	77.38%
Hybrid Model	84.23%

〈표 5-1〉 Word2Vec 적용 전 분류모델별 정확도

5.1.2 Word2Vec 적용 후 분류모델의 결과

표 5-2은 Word2Vec 적용 후 하이브리드 모델의 분류모델의 결과를 Word2Vec 벡터차원에 따라 나타낸 표이다. Word2Vec의 차원이 한 단계씩 증가할수록 추천 정확도는 계속해서 증가하는 것을 확인했다[31]. Word2Vec의 차원 증가를 5차원까지 나타낸 것은 그 이상의 차원에서 학습을 진행할수록 결과가 떨어지는 것을 확인했기 때문에 Word2Vec의 1차원부터 5차원 벡터까지 결과를 비교했다[32]. Word2Vec의 5차원 벡터에서의 결과가 87.46%으로 Word2Vec을 적용하지 않았을 때보다 추천 정확도는 3.23% 증가한 것을 확인했다.

Word2Vec 벡터 차원	정확도
Word2Vec 1차원	85.46%
Word2Vec 2차원	86.79%
Word2Vec 3차원	86.91%
Word2Vec 4차원	87.12%
Word2Vec 5차원	87.46%

〈표 5-2〉 Word2Vec 적용 후 분류모델의 차원별 정확도

표 5-3는 위의 표와 같이 Word2Vec 적용 후 차원별 결과표지만 적중률을 적용했을 때의 결과이다. 적중률의 N 값은 5로 설정했으며, 이는 학습 모델이 데이터를 기반으로 벡터 값을 예측했을 때 해당 벡터 값과 수치상으로 가까운 상품 5개를 찾고 그 안에 실제 구매 상품이 있을 경우 적중했다라고 한다. 결과는 Word2Vec 5차원에서의 결과가 88.36%의 적중률로 가장 높았으며 1차원보다 2.30% 상승했다.

Word2Vec 벡터 차원	적중률
Word2Vec 1차원	86.06%
Word2Vec 2차원	87.27%
Word2Vec 3차원	87.52%
Word2Vec 4차원	87.61%
Word2Vec 5차원	88.36%

〈표 5-3〉 Word2Vec 적용 후 분류모델의 차원별 적중률

5.1.3 Word2Vec 적용 후 회귀모델의 결과

본 연구는 분류모델만 사용이 가능한 데이터에 Word2Vec을 이용하여 단어 간의 연관성을 벡터상의 수치로 표현했기 때문에 학습 속도가 빠른 회귀모델의 사용이 가능하다. 표 5-4는 하이브리드 모델의 회귀모델이 예측한 각 차원별 벡터 값에서 가장 가까운 상품 하나에 대한 추천 정확도에 대한 표이다. 회귀모델은 수치를 예측하기 때문에 클래스를 예측하는 분류모델과는 다르게 가장 가까운 벡터 값을 예측 값을 설정했다[33]. Word2Vec 1차원부터 5차원까지 벡터 차원이 한 개씩 늘어갈수록 정확도 또한 늘어가는 것을 확인했다. Word2Vec 1차원 벡터의 회귀모델 정확도는 분류모델보다 약 12% 떨어지지만 차원이 증가할수록 분류모델의 정확도보다 높아지는 것을 확인했다. Word2Vec 5차원 벡터의 회귀모델의 정확도는 97.92%로 분류모델의 정확도보다 약 10% 증가한 것을 확인했다.

Word2Vec 벡터 차원	정확도
Word2Vec 1차원	73.18%
Word2Vec 2차원	82.33%
Word2Vec 3차원	92.65%
Word2Vec 4차원	97.68%
Word2Vec 5차원	97.92%

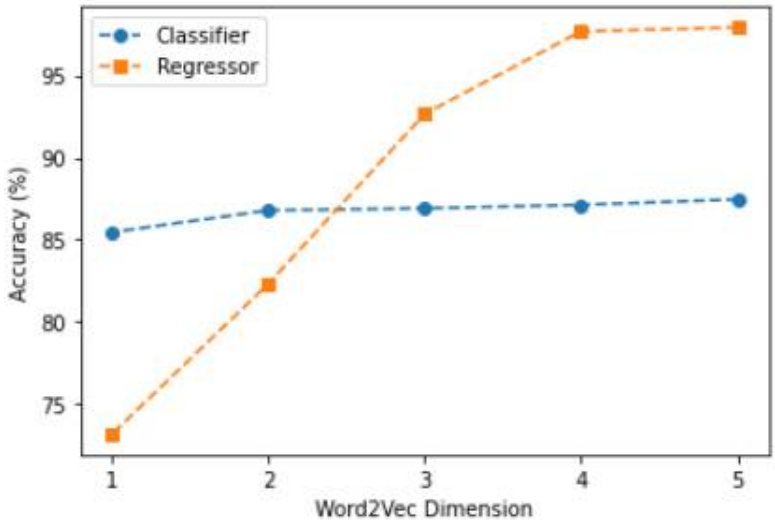
〈표 5-4〉 Word2Vec 적용 후 회귀모델의 차원별 정확도

표 5-5은 하이브리드 모델의 회귀모델이 예측한 Word2Vec 차원별 벡터 값과 가까운 상품 5개에 대한 추천 정확도이다. 전체적으로 1개의 상품을 추천했을 때보다 5개를 추천했을 때 추천 정확도가 증가했다. 분류모델의 적중률과 비교해봤을 때 Word2Vec 5차원 벡터에서 적중률이 약 11% 증가한 것을 확인했다.

Word2Vec 벡터 차원	적중률
Word2Vec 1차원	89.24%
Word2Vec 2차원	87.82%
Word2Vec 3차원	97.43%
Word2Vec 4차원	98.92%
Word2Vec 5차원	99.12%

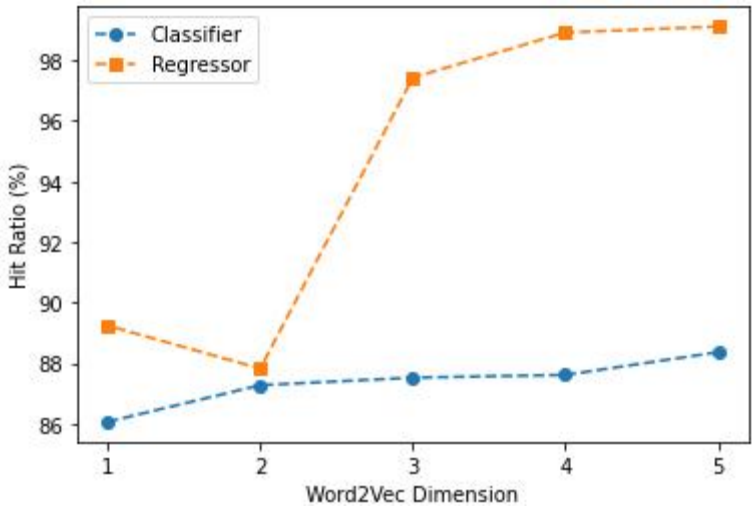
〈표 5-5〉 Word2Vec 적용 후 회귀모델의 차원별 적중률

그림 5-1는 표 5-2과 5-4를 비교한 것이다. Word2Vec 1차원과 2차원에서는 분류모델의 정확도가 회귀모델보다 높았다. 분류모델은 벡터 차원이 낮아도 86% 이상의 안정적인 결과를 보였으나, 반면 회귀모델은 벡터 차원이 낮을 때 80%이하의 정확도를 보였다. 그러나 Word2Vec 3차원 이상의 차원에서는 회귀모델의 성능이 분류모델을 뛰어넘는 것을 확인했다. 3차원 벡터부터는 회귀모델의 성능은 90%가 넘어가게 되는데, 분류모델은 87%를 유지했다. 차원이 증가함에 따라 상품들 간의 연관성이 뚜렷해지면서 회귀모델의 성능이 증가하는 것을 확인했다.



〔그림 5-1〕 분류모델과 회귀모델의 정확도 비교 그래프

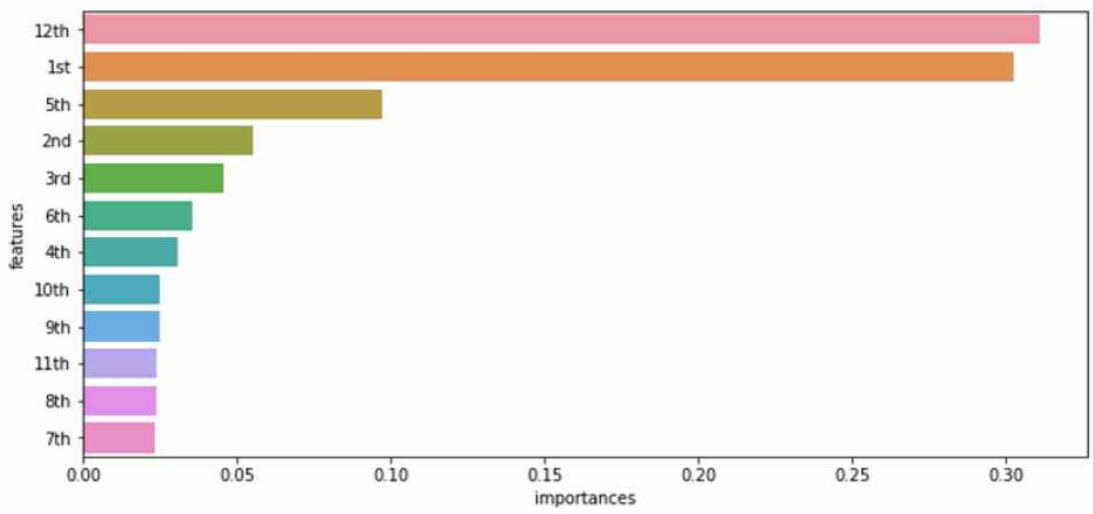
한 개의 상품을 추천하는 것보다 5개의 상품을 추천했을 때의 추천 정확도가 더 높게 나오는 것을 확인했다[34]. 그림 5-2는 하이브리드 모델의 회귀모델과 분류모델의 적중률을 비교한 그래프이다. 회귀모델의 Word2Vec 1차원 벡터에서의 정확도는 89.24%로 한 개의 상품을 추천했을 때의 정확도보다 약 16% 증가했다. 회귀모델은 Word2Vec의 저차원에서도 어느 정도 비슷한 상품을 추천했다는 것을 알 수 있다. 반면 분류모델의 5개 상품 추천 정확도는 약 2%정도 증가하지만, 어느 정도 비슷한 상품을 추천하지 못하고 예측에 실패한 경우 관련성이 먼 상품을 추천한다는 것을 알 수 있다. 결과적으로 회귀모델은 예측에 실패해도 비교적 가까운 유사도의 상품을 추천하는 반면 분류모델은 회귀모델에 비해 그렇지 못하다는 것을 알 수 있다.



〔그림 5-2〕 분류모델과 회귀모델의 적중률 비교 그래프

5.2 특징 중요도

특징 중요도는 머신러닝에서 각 열이 결과에 얼마나 영향을 주는지 확인할 수 있다[35]. 본 연구에 사용한 데이터의 열은 상품을 구매하기 전 클릭 내역이며 클릭 순서에 따라 나열한 데이터이다. 따라서 해당 데이터에 특징 중요도를 적용하면, 상품을 구매하기 전에 각각의 클릭 순서가 상품 구매에 얼마나 영향을 많이 줬는지 알 수 있다. 그림 5-3는 몇 번째 클릭 순서가 결과에 얼마나 영향을 주는 지에 대한 그래프이다. 본 데이터에서 상품을 구매하기 직전 클릭한 상품인 12번째로 클릭한 상품이 31.11%로 결과에 가장 큰 영향을 미쳤다. 다음으로, 첫 번째로 클릭한 상품이 30.28%로 결과에 영향을 미쳤다. 이후에는 5번째로 클릭한 상품이 9.72%, 2번째로 클릭한 상품이 5.53%로 영향을 미쳤다. 첫 번째로 클릭한 상품과 마지막 클릭 상품을 제외한 클릭 상품들은 10%이하의 영향력을 미쳤다. 이러한 결과를 보고 처음 클릭한 상품과 마지막 클릭 상품이 상품 구매에 가장 영향을 많이 미치는 것을 확인할 수 있다[36].



〔그림 5-3〕 특징 중요도

VI.결론

Word2Vec를 이용하여 쇼핑 상품 추천 정확도를 증가시켜, 사용자들에게 보다 정확한 상품 추천을 제공한다. 사용자들의 상품 클릭 내역을 수집하여 학습시키고 비슷한 상품 클릭 내역을 가진 다른 사용자에게 학습시킨 것을 기반으로 상품을 추천한다. 사용들의 상품 클릭 내역은 클릭 순서에 따라 나열한 이후에 Word2Vec를 적용했기 때문에 데이터 프레임 순서대로 연관성을 찾아주는 Word2Vec을 이용할 수 있었다. 상품들 간의 연관성을 찾고 학습 이후 예측을 진행했을 때 추천 정확도가 증가할 것이라는 가정으로 연구를 진행했다. 클릭 순서별로 구매할 상품에 영향을 주는 정도를 특징 중요도를 이용하여 구했다.

기존 데이터는 상품의 고유 번호를 나타냈기 때문에 머신러닝의 분류모델만 사용 가능한 상태였다. Word2Vec를 이용하여 상품들 간의 연관성을 숫자로 표현했기 때문에 머신러닝의 회귀모델 또한 사용이 가능했다. 그리하여 사용 가능해진 회귀모델의 성능과 분류모델의 성능을 Word2Vec 1차원 벡터부터 5차원 벡터까지의 추천 정확도를 비교했다. 그 결과 회귀모델의 추천 정확도는 Word2Vec 벡터 차원이 증가할수록 증가의 폭이 컸다. 한 개의 상품만을 추천했을 때의 정확도는 1차원 벡터와 2차원 벡터에서는 분류모델의 성능이 좋았지만 3차원 벡터 이후부터는 회귀모델이 뛰어넘는 것을 확인했다. 실제 추천 시스템 상황과 비슷하게 5개의 상품을 추천하여 정확도를 비교했을 때는 모든 차원에서 회귀모델의 성능이 분류모델의 성능보다 높게 나오는 것을 확인했다.

참 고 문 헌

[1] Hao, N.; Wang, H.H.; Zhou, Q. The impact of online grocery shopping on stockpile behavior in Covid-19. China Agric. Econ. Rev. 2020, 12, 459-470.

[2] STATISTICS, K. Annual Report, 2020. Available online: https://kostat.go.kr

[3] Gomez-Uribe, C.A.; Hunt, N. The Netflix Recommender System: Algorithms, Business Value, and Innovation. ACM Trans. Manag. Inf. Syst. 2016, 6.

[4] Sardianos, C.; Ballas Papadatos, G.; Varlamis, I. Optimizing parallel collaborative filtering approaches for improving recommen┐dation systems performance. Information 2019, 10, 155.

[5] Park, S.; Seong, D.; Byun, Y. A Hybrid Collaborative Filtering based on Online Shopping Patterns using XGBoost and Word2Vec. J. Korean Inst. Inf. Technol. 2020, 18, 1 - 8.

[6] Basilico, Justin, and Thomas Hofmann. "Unifying collaborative and content-based filtering." Proceedings of the twenty-first international conference on Machine learning. 2004.

[7] Lika, Blerina, Kostas Kolomvatsos, and Stathes Hadjiefthymiades. "Facing the cold start problem in recommender systems." Expert Systems with Applications 41.4 (2014): 2065-2073.

[8] Su, Xiaoyuan, and Taghi M. Khoshgoftaar. "A survey of collaborative filtering techniques." Advances in artificial intelligence 2009.

[9] He, Xiangnan, et al. "Neural collaborative filtering." Proceedings of the 26th international conference on world wide web. 2017.

[10] Li, Yu, Liu Lu, and Li Xuefeng. "A hybrid collaborative filtering method for multiple-interests and multiple-content recommendation in E-Commerce." Expert systems with applications 28.1 (2005): 67-77.

[11] Goldberg, Y.; Levy, O. word2vec Explained: Deriving Mikolov et al.'s negative-sampling word-embedding method. arXiv 2014, arXiv:1402.3722.

[12] McCormick, C. Word2vec Tutorial-the Skip-Gram Model. 2016. Available, http://mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model

[13] Ling, W.; Dyer, C.; Black, A.W.; Trancoso, I. Two/too simple adaptations of word2vec for syntax problems. In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, CO, USA, 31 May - 5 June 2015; pp. 1299 - 1304.

[14] Loh, W.Y. Classification and regression trees. Wiley Interdiscip. Rev. Data Min. Knowl. Discov. 2011, 1, 14 - 23.

[15] Sidiropoulos, Nicholas D., et al. "Tensor decomposition for signal processing and machine learning." IEEE Transactions on Signal Processing 65.13 (2017): 3551-3582.

[16] Batista, Gustavo EAPA, Ronaldo C. Prati, and Maria Carolina Monard. "A study of the behavior of several methods for balancing machine learning training data." ACM SIGKDD explorations newsletter 6.1 (2004): 20-29.

[17] Safavian, S. Rasoul, and David Landgrebe. "A survey of decision tree classifier methodology." IEEE transactions on systems, man, and cybernetics 21.3 (1991): 660-674.

[18] Liaw, Andy, and Matthew Wiener. "Classification and regression by randomForest." R news 2.3 (2002): 18-22.

[19] Ke, Guolin, et al. "Lightgbm: A highly efficient gradient boosting decision tree." Advances in neural information processing systems 30 (2017): 3146-3154.

[20] Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 2016.

[21] Ahmad, Muhammad Waseem, Jonathan Reynolds, and Yacine Rezgui.

"Predictive modelling for solar thermal energy systems: A comparison of support vector regression, random forest, extra trees and regression trees." Journal of cleaner production 203 (2018): 810-821.

[22] He, Jinlu. "사용자 평점과 리뷰 유사도를 이용한 협업 필터링 기반 영화 추천시스템." 국내석사학위논문 경희대학교 대학원, 2021

[23] 남기백 and 박상원. "머신러닝 기반 고객 재구매 상품 예측." 한국정보처리학회 학술대회논문집 24.2 (2017): 421-423.

[24] 황민기. "Item2vec을 이용한 하이브리드 협업필터링." 국내석사학위논문 서강대학교 정보통신대학원, 2018. 서울

[25] Dawood, Edel Goreil. Geo-locating UEs Using Multi-output Decision Tree Regressor. Diss. 2019.

[26] Shahbazi, Z.; Hazra, D.; Park, S.; Byun, Y.C. Toward Improving the Prediction Accuracy of Product Recommendation System Using Extreme Gradient Boosting and Encoding Approaches. Symmetry 2020, 12, 1566.

[27] Sudharsan, Bharath, John G. Breslin, and Muhammad Intizar Ali. "Edge2train: A framework to train machine learning models (svms) on resource-constrained iot edge devices." Proceedings of the 10th International Conference on the Internet of Things. 2020.

[28] García, Salvador, et al. "A study of statistical techniques and performance measures for genetics-based machine learning: accuracy and interpretability." Soft Computing 13.10 (2009): 959.

[29] Davis, Jesse, and Mark Goadrich. "The relationship between Precision-Recall and ROC curves." Proceedings of the 23rd international conference on Machine learning. 2006.

[30] Kotsiantis, Sotiris B., I. Zaharakis, and P. Pintelas. "Supervised machine learning: A review of classification techniques." Emerging artificial intelligence applications in computer engineering 160.1 (2007): 3-24.

[31] Ma, Long, and Yanqing Zhang. "Using Word2Vec to process big text data." 2015 IEEE International Conference on Big Data (Big Data). IEEE,

2015.

[32] Alshari, Eissa M., et al. "Improvement of sentiment analysis based on clustering of Word2Vec features." 2017 28th international workshop on database and expert systems applications (DEXA). IEEE, 2017.

[33] Trawiński, Bogdan, et al. "Nonparametric statistical analysis for multiple comparison of machine learning regression algorithms." International Journal of Applied Mathematics and Computer Science 22 (2012): 867-881.

[34] Bennett, James, and Stan Lanning. "The netflix prize." Proceedings of KDD cup and workshop. Vol. 2007. 2007.

[35] Altmann, André, et al. "Permutation importance: a corrected feature importance measure." Bioinformatics 26.10 (2010): 1340-1347.

[36] Valko, Michal, and Milos Hauskrecht. "Feature importance analysis for patient management decisions." Studies in health technology and informatics 160.Pt 2 (2010): 861.