

상권 분석을 통한 입점 종목 추천 시스템 개발에 관한 연구

김준현¹, 윤상필¹, 강희성¹, 문선규¹

¹명지대학교 융합소프트웨어학부 학생

요약문

본 연구는 상권 분석을 통한 입점 종목 추천 시스템 개발에 쓰인 일련의 과정에 관해 서술하는 것을 목적으로 한다. 이를 위해 2017년부터 2022년까지의 서울시의 상권 데이터와 건축물 데이터를 이용하였으며, 상권 데이터는 생활인구를 비롯한 총 6개의 데이터에서 필요한 부분만을 전처리한 데이터이고 건축물 데이터는 건축물대장을 비롯한 총 7개의 데이터에서 결측치와 이상치를 제거한 데이터이다. 해당 시스템의 알고리즘은 기존에 있던 상권 분석 서비스와 정반대의 프로세스를 가지고 있다. 따라서 상권, 건축물 데이터로 추천되는 업종의 추정 매출액을 도출하기 위해서는 건축물 데이터로 로지스틱 회귀분석을, 분석을 통해 나온 업종 결과와 상권 데이터로 선형 회귀분석을 하여 결론적으로 추정매출액을 구하는 프로세스로 진행하였다. 다만, Decision Boundary를 조금 더 잘 나눌 수 있는 MLP(MultiLayer Perceptron)를 로지스틱 함수 대신 사용했고, 선형회귀분석의 경우에는 단순 선형 회귀를 사용하는 것과 Decision Tree의 일종인 XGBoost를 사용하여 성능을 비교해 본 결과, XGBoost와 GridSearch를 이용한 것이 30% 정도 더 좋은 결과를 내보였다. 프론트엔드는 html, css, javascript를 이용해 홈 화면 및 자리 찾기, 분석 데이터 정보 페이지를 구성했고, 백엔드는 node.js와 카카오 지도 API를 통한 시각화, 마커 생성을 구현했다. 검증 시나리오는 분류를 위한 threshold를 accuracy 85% 이상, 회귀를 위한 threshold를 R^2 값 이상이면 유의미한 결과를 낼 것이라고 판단했다. 최종적으로 XGBoost를 사용한 회귀 결과 R^2 값은 0.8630이 나와 threshold를 넘은 것을 알 수 있었고, MLP를 이용한 accuracy값은 85.2%가 나와 마찬가지로 threshold를 넘은 것을 알 수 있었다.

Keywords: 상권, 입점, 추천시스템, 회귀분석, 업종, MLP, XGBoost

I. 서론

1. 연구의 배경과 목적

장기화된 코로나19의 영향으로, 대한민국의 경제는 극심한 경기침체를 맞이하게 되었다. 비록 이전보다 상황이 많이 좋아진 현재도, 실생활에서 국민들이 체감하는 체감물가는 절대 낮아지지 않았다. 높아진 체감물가에 따른 소비감소는 서울지역의 전반적인 상권의 쇠퇴 및 공실을 증가로 이어지게 되었다. 장기간 공실 상태가 유지된다면, 건물주와 소비자 그리고 지역 경제에도 악영향을 끼치게 된다. 서울 대표 상권 중 한 곳으로 꼽혔던 홍대입구역 상권은 코로나19의 직격탄을 제대로 받은 곳으로 유명하다. 코로나 이전 2019년의 홍대입구역 인근 상권의 점포 수, 매출 건수를 2021년과 비교해볼 때, 8만여 개의 매출 건수는 5만여개로, 2000개가 넘는 점포 수 또한 500개 미만으로 줄어들었다. 이러한 좋지 못한 시국에 입점 종목을 아직 정하지 못한 자영업자에게 어떻게 하면 좀 더 괜찮은 업종을 추천할 수 있을까 하는 고민을 하게 되었다. 본 논문에서는 침체된 경기에 조금이나마 도움이 되고자 서울시 내의 상권 분석을 통하여 입점 종목을 추천해주는 시스템의 개발 과정에 관해 기술하고, 설계 및 구현에 대한 내용과 이 과정에서의 문제점, 검증 시나리오를 기술한다.

2. 연구의 범위와 논문의 구성

1) 연구의 범위

본 논문의 공간적 범위는 서울시로 한정하고 서울 열린 데이터 광장의 데이터 중 하나인 상권영역 데이터의 분류 기준에 따른다. 연구의 시간적 범위는 2017년부터 2022년까지의 데이터이며 추가로 자동화 가능한 최신데이터가 있다면 업데이트 또한 가능하다.

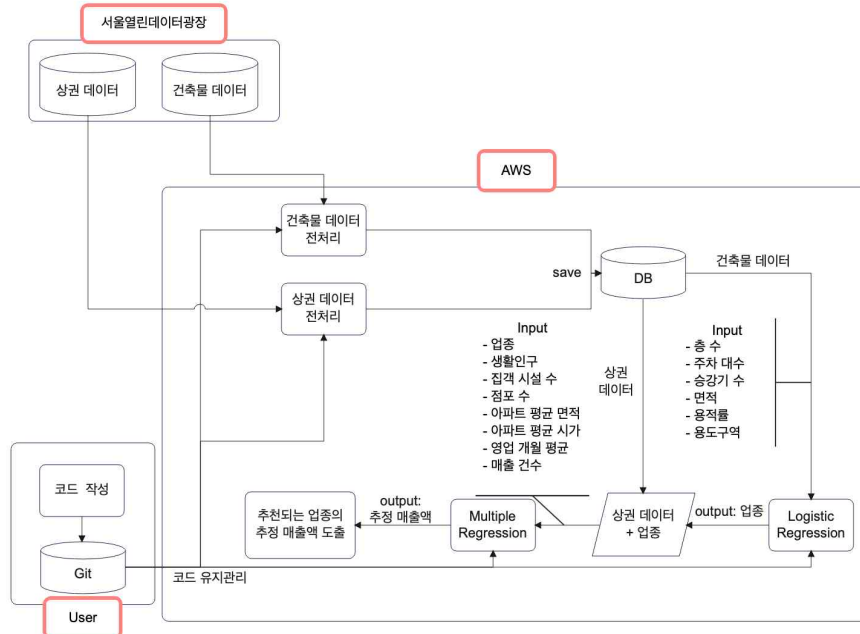
2) 논문의 구성

본 논문의 구성은 다음과 같다. 1장 서론에서는 연구의 배경과 목적, 내용과 구성을 밝히고, 2장에서는 개발 과정 및 문제 정의를 다루었다. 3장에서는 데이터와 관련된 모든 것이 담긴 솔루션에 관한 내용을 다루었고, 4장에서는 그중에서도 알고리즘을 중점적으로 다루었다. 5장에서는 데이터 분석의 결과, 마지막 6장에서는 결론 및 한계점 및 시사점을 제시한다.

II. 문제 정의

1. 상권에 입점 종목을 추천하는 시스템의 주요 과정

해당 시스템의 주요 흐름도를 보면 다음과 같다.



<그림 1> 추천시스템의 주요 흐름도

서울 열린 데이터 광장에서 가져온 상권 데이터와 건축물 데이터를 위의 흐름도를 보면 알 수 있듯이, 추천되는 업종의 추정 매출액을 도출하는 것을 최종적인 결과물로 판단한다. 서울 열린 데이터 광장에서 가져온 상권 데이터와 건축물 데이터를 분석에 용이하게 전처리를 진행한 후, 건축물 데이터 중 필요한 칼럼들인 ‘층수’, ‘주차 대수’, ‘승강기 수’, ‘면적’ 등 6개의 칼럼을 로지스틱 회귀의 입력값으로 넣은 후, 출력값으로 건축물 기반 업종이 나오게끔 설정한 다음 상권 데이터와 업종을 합친 데이터를 통해 선형 회귀 분석을 하게 된다면 최종적으로 추천되는 업종의 추정 매출액을 도출할 수 있게 된다. 코드 형상 관리는 Git을 사용하고, 서버는 AWS의 EC2 서버를 사용한다. 알고리즘 부분의 더욱 자세한 내용은 IV장에 기술된다.

2. 상권에 입점 종목을 추천하는 시스템의 평가 기준

1) 첫 번째 평가 기준

첫 번째 기준은 [1]을 통해 알 수 있다. 해당 논문은 R^2 값을 활용하는 데 정당성을 부여한다. MSE, MAE, RMSE는 모델마다 결과 값이 다르고, 0에 근접한 값으로 표현되면 좋은 성능을 발휘하지만, 0에 얼마나 근접하느냐에 따라 우수한 성능으로 판단되는지에 대해 성능 판단의 어려움이 있기 때문에 회귀분석에서 독립변수가 종속변수를 얼마나 잘 설명해주는지를 나타내는 결정계수인 R^2 를 성능 검증지표로서 사용한다. 해당 논문에서 가장 우수한 성능을 보인 R^2 값은 0.931 이다.

2) 두 번째 평가 기준

두 번째 기준은 [2]를 통해 알 수 있다. 해당 논문은 부동산의 실거래가를 예측하는 방법을 담은 논문으로 DNN과 Boosting 방법 3가지를 활용하여 예측한다. 해당 논문의 프로세스는 전처리 후 다중 공산성을 제거하고, 변수를 선정하는 과정을 거친 후 모델에 적용하는 방법이다.

<표 1> Variable Selection Method & Fit Result by Model [2]

	model	r2_score	mean_squared_error	mean_absolute_error	rmse
Variable selection	Linear Regression	-0.4224	0.0026	0.0332	0.0515
	XGBoost	0.7965	0.0008	0.0182	0.0291
	CatBoost	0.8464	0.0006	0.0155	0.0255

해당 표는 변수 선택 과정 이후에 모델을 적용했을 때 나타난 지표들이다. 표를 보면 XGBoost와 CatBoost가 성능이 괜찮게 나온 것을 볼 수 있고, 그중에서 구현 난이도가 좀 더 쉬운 XGBoost를 상권에 입점 종목을 추천하는 시스템의 주요 알고리즘으로 사용했다.

3) 세 번째 평가 기준

마지막 기준은 [3]을 통해 알 수 있다. [3]논문은 점포의 특성에 따라 상권을 추천해주는 알고리즘에 대한 연구라서 해당 논문의 시스템과 유사한 부분이 많지만, 추천 알고리즘의 과정이 정반대로 되어있기 때문에 감안하고 판단해야 한다. (현재 업종 시스템은 건물의 특성을 기반으로 업종을 추천하는 방식) 그러나 순방향의 과정이 성립한다면 역방향의 과정 또한 예측이 가능하다. 다음 표는 [3] 논문의 모델 정확도에 관한 표이다.

<표 2> Model Accuracy [3]

알고리즘	Accuracy	
Decision Tree	Train set	0.93
	Test set	0.85
Xgboost Regression	선유로운	0.99
	양재천길	0.95
	합마르뜨	0.99
	장충단길	0.95
	오류버들	0.98

4) 세 가지 기준에 대한 판단

참고논문 [1],[2],[3]을 참고했을 때, 분류를 위한 threshold를 accuracy 85% 이상, 회귀를 위한 threshold를 R^2 값 0.85 이상이면 유의미한 결과를 낼 것이라 판단했다.

III. 솔루션

1. 상권 데이터

해당 논문에서 사용한 상권 및 건축물 데이터는 서울 열린 데이터 광장에서 가져왔다. 상권 데이터는 생활인구, 집객시설, 점포, 아파트, 추정 매출, 상권변화지표 총 6개의 데이터에서 각각 2017년부터 2022년까지 6년 치 데이터를 가지고 필요한 칼럼들을 분석하기 용이하게 전처리를 진행하였다. 같은 연도의 같은 상권 코드를 가지고 있어도, 분기별로 다른 총 생활 인구 수가 다르기 때문에 해당 데이터를 연도별 데이터로 변환하였고, 특수문자 및 기능 상실 상권을 제거했다. 또한 기존의 서비스 업종 코드의 개수는 총 100개였지만, 큰 카테고리로는 묶은 업종의 개수를 16개로 줄였다. 새로 만들어진 업종 분류표는 다음과 같다.

<표 3> 업종 분류 표

분류(개수)	1	2	3	4	5	6	7	8	9	10
음식업 및 주점업	한식음식점	호프-간이주점	치킨전문점	패스트푸드점	제과점	양식음식점	일식음식점	중국음식점	분식전문점	커피-음료
식료품 도(소)매업	반찬가게	청과상	수산물판매	육류판매	미곡판매	주류판매	편의점	슈퍼마켓		
보건업	동물병원	한의원	치과의원	일반의원						
전문 서비스업	세무사 사무소	법무사 사무소	인테리어	기타법무서비스	회계사사무소	통번역서비스	변리사사무소			
교육 및 사업지원 서비스업	예술학원	외국어학원	일반교습학원	컴퓨터학원	어행사					
스포츠 및 오락 서비스업	골프연습장	스포츠클럽	볼링장	스포츠강습	pc방	기타오락장	노래방	복권방	dvd방	전자게임장
개인 서비스업	세탁소	피부관리실	미용실	사진관	녹음실	네일숍				
기타 서비스업	건축물청소	독서실								
생활용품 도(소)매업	일반의류	운동/경기용품	유아의류	한복점	서적					
가전제품 및 통신기기 도(소)매업	핸드폰	컴퓨터및주변장치 판매	가전제품							
종교상품 도(소)매업	재생용품 판매점	종교차판매	종교가구							
기타 도(소)매업	화초	섬유제품	완구	미용재료	화장품	문구	주유소	전자상거래업	예원동물	
수리업	모터사이클및부품	자동차부품	가전제품 수리	모터사이클수리	자동차미용	자동차수리	통신기기수리	가방	신발	
숙박업	고시원	여관	게스트하우스							
부동산 및 임대업	부동산중개업	가정용품임대	비디오/서적임대	의류임대						
제조업	안경	조명용품	의료기기	의약품	시계및거금속	가구	악기	철물점	자전거 및 기타운송장비	

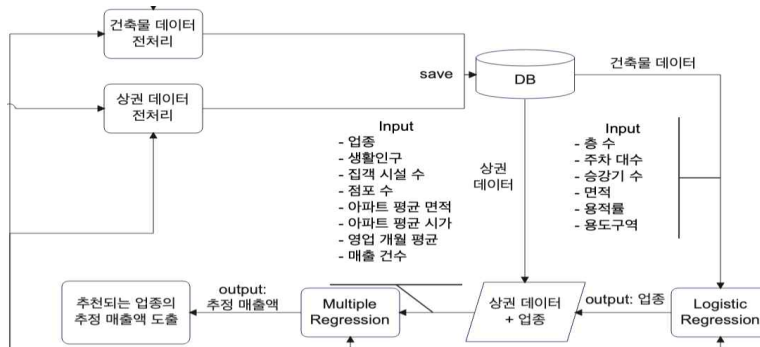
2. 건축물 데이터

건축물 데이터 또한 서울 열린 데이터 광장에서 가져왔다. 건축물 데이터는 건축물대장, 표제부, 총괄 표제부, 층별 개요정보, 지역 지구 구역, 전유부, 전유 공유면적 총 7개의 데이터에서 결측치, 이상치를 제거하였다. 건축물대장 데이터에서 대지 구분, 특수지, 블록 등 Nan 값이 많고 추천 알고리즘과 연관성이 없는 칼럼들을 제거하였고, 표제부 데이터에서는 구조 구분, 기타 구조, 지붕 구분 등 건물이 지어진 방법에 대한 내용 또한 추천 알고리즘과 연관성이 없어서 제거하였다. 이 외에도 매우 많은 데이터의 상태들의 단위, 숫자가 서로 일치하지 않는 경우가 많았기 때문에 따로 각각 비교하여 차이를 찾아야 했다.

IV. 알고리즘

1. 전체적인 알고리즘 개요

그림 2는 해당 시스템의 주요 흐름도 중 알고리즘 부분만 확대한 그림이다.



<그림 2> 추천시스템의 알고리즘

현존하는 상권분석 서비스는 이용자가 먼저 업종을 고른 후, 업종에 맞는 상권을 찾아주는 시스템이었다면 해당 시스템은 먼저 상권을 고른 후, 상권에 맞는 업종을 추천해주는 시스템이다. 따라서 기존의 상권분석 서비스에서 사용된 알고리즘을 똑같이 사용할 수 없다. <그림 2>를 보면 상권, 건축물 데이터로 추천되는 업종의 추정 매출액을 도출하기 위해서는 건축물 데이터로 로지스틱 회귀분석을, 분석을 통해 나온 업종 결과와 상권 데이터로 선형 회귀분석을 하여 결론적으로 추정매출액을 구하는 프로세스로 진행되었다는 것을 알 수 있다. 크게 보면 두 단계의 분석단계로 나눌 수 있는데, 자세한 부분별 내용은 다음과 같다.

2. 로지스틱 회귀

로지스틱 회귀 부분에서는 단순한 로지스틱 함수가 아닌 결정 경계를 좀 더 잘 나눌 수 있는 MLP(MultiLayer Perceptron)를 사용한다. MLP는 로지스틱 함수보다 비선형성을 좀 더 잘 표현할 수 있고, hidden layer가 여러 개 있기 때문에 비교적 간단한 Sigmoid 함수를 쓰며 단일 계층 모델인 로지스틱 함수보다는 좀 더 잘 분류할 수 있을 것이라 판단한다.

3. 선형 회귀

선형회귀분석의 경우에는 단순 선형회귀를 사용하는 것과 Decision Tree의 일종인 XGBoost를 사용하여 성능을 비교해본 결과, XGBoost와 GridSearch를 이용한 것이 30% 정도 더 좋은 결과를 내보였다. XGBoost는 eXTreme Gradient Boosting의 줄임말로, 비선형성 변환과 변수 간 상호작용을 잘 통합할 수 있는 강점이 있기 때문에 해당 모델을 사용한다.

V. 분석 결과

XGBoost Regression model은 상권 데이터로부터 업종에 대한 매출을 예측하기 위한 모델이고, 업종에 대한 예측을 좀 더 간편히 수행하고자 업종명을 one-hot-encoding을 수행했다. 그렇게 도출된 하나의 모델로 16가지 모든 업종의 예상 매출을 구했다. 구하는 과정에서 GridSearchCV를 사용하여 여러 하이퍼파라미터의 조합을 확인을 했고, R^2 값이 0.85를 넘지 못하는 상황이 발생했다. 그래서 칼럼값 중에서 값의 크기가 상대적으로 컸던 ‘아파트_평균_시가’, ‘매출_금액’ 칼럼값에 log를 취하여 값을 변환한 결과, R^2 값이 0.8638이 넘어 평가 기준을 넘은 것을 확인할 수 있었다. 아래의 그림은 XGBoost의 training 결과이다.

```
XGB's best parameters: , XGBRegressor(base_score=None, booster=None, callbacks=None,
    colsample_bylevel=None, colsample_bynode=None,
    colsample_bytree=0.9, early_stopping_rounds=None,
    enable_categorical=False, eval_metric=None, feature_types=None,
    gamma=0, gpu_id=0, grow_policy=None, importance_type=None,
    interaction_constraints=None, learning_rate=0.1, max_bin=None,
    max_cat_threshold=None, max_cat_to_onehot=None,
    max_delta_step=None, max_depth=5, max_leaves=None,
    min_child_weight=None, missing=nan, monotone_constraints=None,
    n_estimators=5000, n_jobs=None, num_parallel_tree=None,
    predictor=None, random_state=None, ...)
XGB 예측 정확도: 0.8638
training time: 65.18909120559692
```

<그림 3> XGBoost 실행 후 R^2 값

MLP model은 매물로부터 층 수를 받아서 해당 매물이 어떤 업종에 가장 적합한지를 알아내는 모델이다. 데이터를 한꺼번에 학습하지 않고, 단위별로 쪼개서 학습시키는 minibatch 기법을 사용하였고, 차원을 1차원으로 줄여주는 Flatten(), 선형변환 시켜주는 모듈인 Linear(), 활성화 함수로 ReLU를 사용하였고, Dropout()을 통해 서로 연결된 신경망을 0부터 1 사이의 확률로 노드를 제거하여 과적합을 방지하는 용도로 사용했다. 그 결과 Accuracy는 약 85.2%를 기록하였고, 평가 기준을 넘는 것을 확인할 수 있었다. <그림 4>는 MLP의 구조를, <그림 5>는 Accuracy결과이다.

```

class MLP(torch.nn.Module):
    def __init__(self):
        super(MLP, self).__init__()
        self.linear_relu_stack = nn.Sequential(
            nn.Flatten(),
            nn.Linear(3, 121),
            nn.ReLU(),
            nn.Dropout(p = 0.5),
            nn.Linear(121, 121),
            nn.ReLU(),
            nn.Dropout(p = 0.5),
            nn.Linear(121, 121),
            nn.ReLU(),
            nn.Dropout(p = 0.5),
            nn.Linear(121, 14),
        )
    def forward(self, x):
        logits = self.linear_relu_stack(x)
        return logits

```

<그림 4> MLP 구조

```

loss: 0.877843 [ 0/22686]
loss: 0.492179 [ 6400/22686]
loss: 0.788748 [12800/22686]
loss: 0.548839 [19200/22686]
test Error:
Accuracy: 85.2%, Avg loss: 0.661993

loss: 0.572046 [ 0/22686]
loss: 0.628272 [ 6400/22686]
loss: 0.797976 [12800/22686]
loss: 0.688307 [19200/22686]
test Error:
Accuracy: 85.2%, Avg loss: 0.659376

loss: 0.463003 [ 0/22686]
loss: 0.790571 [ 6400/22686]
loss: 0.577131 [12800/22686]
loss: 0.619370 [19200/22686]
test Error:
Accuracy: 85.2%, Avg loss: 0.664101

loss: 0.732376 [ 0/22686]
loss: 0.972838 [ 6400/22686]
loss: 0.819949 [12800/22686]
loss: 0.599951 [19200/22686]
test Error:
Accuracy: 85.2%, Avg loss: 0.679694

loss: 0.592382 [ 0/22686]
loss: 0.863150 [ 6400/22686]
loss: 0.451276 [12800/22686]
loss: 0.989916 [19200/22686]
test Error:
Accuracy: 85.2%, Avg loss: 0.656537

loss: 0.625621 [ 0/22686]
loss: 0.722949 [ 6400/22686]
loss: 0.673076 [12800/22686]
loss: 0.650201 [19200/22686]
test Error:
Accuracy: 85.2%, Avg loss: 0.660241

```

<그림 5> MLP 이후 Accuracy

VI. 결론

본 연구는 상권 분석을 통한 입점 종목 추천 시스템 개발에 쓰인 일련의 과정에 관해 서술하는 것을 목적으로 한다. 이를 위해 사용된 상권 데이터와 건축물 데이터는 모두 서울 열린 데이터 광장에서 참조했으며, 각각의 데이터는 결측치와 이상치, 그리고 무의미한 칼럼은 제거한 데이터이다. 해당 시스템의 알고리즘은 기존에 있던 상권 분석 서비스와 정반대의 프로세스를 가지고 있다. 따라서 상권, 건축물 데이터로 추천되는 업종의 추정 매출액을 도출하기 위해서는 건축물 데이터로 로지스틱 회귀분석을, 분석을 통해 나온 업종 결과와 상권 데이터로 선형 회귀분석을 하여 결론적으로 추정매출액을 구하는 프로세스로 진행하였다. 다만, Decision Boundary를 조금 더 잘 나눌 수 있는 MLP(MultiLayer Perceptron)를 로지스틱 함수 대신 사용했고, 선형회귀분석의 경우에는 단순 선형회귀를 사용하는 것과 Decision Tree의 일종인 XGBoost를 사용하여 성능을 비교해본 결과, XGBoost와 GridSearch를 이용한 것이 30% 정도 더 좋은 결과를 내보였다. 프론트엔드는 html, css, javascript를 이용해 홈 화면 및 자리 찾기, 분석 데이터 정보 페이지를 구성했고, 백엔드는 node.js와 카카오 지도 API를 통한 시각화, 마커 생성을 구현했다. 검증 시나리오는 분류를 위한 threshold를 accuracy 85% 이상, 회귀를 위한 threshold를 R^2 값 이상이면 유의미한 결과를 낼 것이라고 판단을 하였고 최종적으로 XGBoost를 사용한 회귀 결과 R^2 값은 0.863이며, MLP를 이용한 accuracy 값은 85.2%가 나와 각각 threshold를 넘은 것을 알 수 있었다.

모델의 종합적 분석결과로 볼 때, 입점 종목을 추천할 때는 추정매출, 건축물 대장 등 다양한 요소들이 필요하고, 변수간 복잡한 상관관계 또한 존재하는 것을 알 수 있다. 또한 MLP, XGBoost보다 더 최신모델, 더 좋은 사양의 GPU, 더 많은 연구시간이 동원되었다면 더욱더 효과적이고 도움이 되는 분석이 되었을 것이다.

비록 부족한 점이 많은 연구였음에도 불구하고, 해당 연구에서 도출한 추천시스템 모델을 통해 소상공인의 창업 활동에 조금이나마 도움이 될 수 있을 것이라 기대한다.

REFERENCES

- [1] Sangmin Kim & Chankwon Park & Jieun Lee. (2022). XGBoost 회귀를 활용한 편의점 계약전력 예측 모델의 최적화에 대한 연구
DOI : 10.9716/ KITS.2022.21.4.091
- [2] Hakhyun Kim & Hwankyoo Yoo & Hayoung Oh (2023) Apartment Price Prediction Using Deep Learning and Machine Learning
DOI : 10.3745/ KTSDE.2023.12.2.59
- [3] Jiyeon Kim & Hyoseon Jang & Minseo Park (2023). Local Brand Alley Commercial Area Recommendation System Design Using Machine Learning
DOI : 10.17703/ JCCT.2023.9.1.101