- 2023-1 데이터마이닝 보고서 -

데이터마이닝을 통한  
레시피데이터 분류 및 인기 조회수 예측

국제경영학과, 20180499, 심유나

요약문

본 과제는 데이터마이닝을 통해 요리커뮤니티 서비스 내 인기 요리를 분류하고 예측한다. 요리를 위해 필요한 재료의 수, 요리 종류(분류), 요리 인분 이외에 요리 난이도, 요리명, 요리 방법 등과 같이 각각 다른 요소의 데이터를 통해 여러 특성으로 설정하여 모델을 학습시킨다. 학습된 상황별 요리 조회수를 분석하여 어떤 특징을 가진 레시피가 요리 커뮤니티 내 주목받는 레시피가 될 수 있는 확률이 높은지 연구하고, 더 많은 정보를 제공할 수 있게 하여 최근 물가상승에 따른 소비자들의 식비 절감노력에 도움이 되고자 한다. 또한 이 분석 데이터를 활용하여 만개의 레시피 서비스 내, 레시피 정보 제공 기반의 이커머스 서비스로 확장 가능성을 기대한다.

Keywords: 본 과제와 관련된 핵심 키워드

Data Mining, Classification, Decision Tree, Support Vector Machine, Recipe Data

1. 서론

1.1 데이터마이닝을 활용한 문제 해결

대한민국은 작년을 시작으로 지난해 7월 외환위기 당시 수준인 6.3%까지 치솟았던 물가 상승률을 시작으로 최근 경기침체와 물가 불안정이 연속되고 있다. 정부기관을 중심으로 물가상승을 억제하기 위해 금리를 상승하는 등의 노력을 통해 최근 물가 상승은 지난해 대비 점차 둔화하기 위해 노력하고 있어 최근 기대인플레이션은 지난해 12월 3.8%, 올해 4월 3.7% 연속으로 조금씩 둔화되고 있는 실정이다. 하지만, 소비자물가지수가 석유류 가격 하락에 힘입어 3.7%때로 하락한 것과 다르게 최근 공공요금 인상 뉴스 등으로 인해 외식이나 개인 서비스, 공업제품 등의 물가지수는 여전히 불안정한 상황이 연속되고 있고, 해당 분야는 물가 하락율이 소폭 감소하고 있다. 또한 이러한 물가 상승에 반해 최근 대한민국은 고용 불안정이 잇따르고 있으며, 물가 상승에 비례하여 임금이 상승하지 못하고 있는 실정이다. 이런 상황으로 인해 특히, 저임금 비정규직 노동자들의 생활이 위협되고 불안정이 연속되고 있는 실정이다.

이런 상황에서 최근 대한민국은 짠테크, 고물가 시대의 식비절약 집밥로그, 무지출 브이로그 등의 키워드들이 하나의 트렌드로 자리잡고 있는 상황이다. 이에 식비절약을 위한 집밥 레시피 등이 인터넷에서 관심을 받고 있다. 이런 상황에서 집밥 레시피 서비스의 중심이 되는 만개의 레시피 서비스 내, 다양한 레시피 데이터 중 어떤 것들이 사람들의 주요 관심사가 되며, 그 주요 요인들이 무엇인지 조사하고 어떤 요인들이 레시피의 인기를 이끄는 요인이 되는가에 대한 궁금증을 해결하기 위해 조사하고 데이터를 분석하기 시작했다.

본 데이터마이닝 과제는 데이터의 분류를 통해 레시피 데이터의 특성별 인기 요인을 파악하기 위해, 조회수를 반응 변수로 나머지 변수들을 예측변수로 사용한다. 이를 기반으로 모델을 비교하여 모델의 성능을 평가해 최적의 모델을 찾고, 조회수에 영향을 미치는 유의미한 변수들을 식별해 소비자들에게 좋은 레시피 데이터가 제공된다면 최근 일어나고 있는 물가상승을 사람들이 조금 더 잘 해결할 수 있을 것이라고 생각했다. 더 나아가 만개의 레시피 서비스는 해당 데이터를 활용하여 인기있는 레시피에 필요한 상품들을 이커머스 사이트 내에 집중할인 판매하는 전략을 통해 수익을 높일 수 있을 것이라고 기대했다.

1. 배경이론

본 과제에서는 Decision Tree와 Support Vector Classifier알고리즘을 활용하여 데이터 분석을 진행한다. 데이터를 활용한 연구에 앞서 본 과제에서 사용된 데이터, 알고리즘 등에 대한 배경 이론을 정리하고 이를 통해 명확한 이해를 기반으로 데이터 분석을 진행하고자 한다.

2.1 Decision Tree

Decision Tree 알고리즘은 Support Vector Machine 분야에서 논리 체계를 Tree 모양으로 구성하는 알고리즘이다. 분류와 회귀 문제를 해결하기 위해 주로 사용되는 지도학습 알고리즘의 하나로 직관적인 구조와 해석 가능성, 데이터 전처리요구 최소화의 장점으로 인해 널리 사용되고 있는 방법이다. 또한 이는 앙상블 학습 알고리즘인 랜덤 포레스트의 기본 구성요소라고 볼 수 있다.

해당 알고리즘의에서 가장 중요한 것은 데이터를 분류하는 기준을 찾는 것인데, 이를 위해 주로 엔트로피와 정보이득이라는 개념을 사용한다. 엔트로피는 주어진 데이터의 불확실성을 나타내는 지표이며, 정보이득은 분류 전후의 엔트로피 차이를 나타낸다. Decision Tree 알고리즘은 복잡한 문제를 다 때 적합하며, 일반적으로 분류와 회귀에 모두 사용된다. 하지만, 본 과제에서는 분류에 집중해서 데이터를 분류한 결과를 알아볼 때 사용하려고 한다. Decision Tree는 데이터 전처 과정이 단순하며, 결과 해석과 시각화에 용이합니다. 또한 큰 규모의 데이터에 적용이 가능하며, 다른 종류의 모델과 함께 사용하여 정확도를 높일 수 있다는 점이 장점이다. 하지만, Decision Tree는 과적합(OverFitting)문제가 발생할 수 있으며, 어떤 조건을 먼저 분류하는지에 따라 결과가 달라질 수 있는 문제를 갖고 있다.

Decision Tree 알고리즘은 다음과 같은 단계를 통해 수행된다.

(1) 알고리즘 최초의 시작은 초기화 값이 설정된 Data Set과 Data Set attribute의 집합의 root에서부터 적용한다.   
(2) Data Set과 Data Set attribute를 파라미터로 받아 Data Set을 검토하여 A의 값이 모두 같으면 그 값을 현재 노드에 기입하고 return한다. 만약 A 값이 같지 않으면 (3)을 진행한다.

(3) Data Set을 검토해 Data Set attribute B를 Data Set attribute로부터 택해 B를 현재의 노드에 기록한다.

(4) A가 취할 수 있는 모든 값에 대하여 Data Set의 여집합과 Data Set attribute 의 여집합과을 다음과 같이 계산하여 (2)항로 Parameter Passing하여 Recursive 하게 진행된다.

2.2 Support Vector Machine

Support Vector Machine(SVM) 알고리즘은 분류(classification) 회귀(regression), 특이점 판별에 쓰이는 지도학습 머신러닝 방법 중 하나다. 본 과제에서는 분류를 중심으로 데이터 분석을 진행하기 때문에 Support Vector Machine(SVM)의 분류 문제와 관련된 이론을 소개한다.

Support Vector Machine(SVM)는 두가지 클래스 중 하나로 데이터를 분류하는 문제를 다룬다. 이 알고리즘의 핵심 아이디어는 데이터를 선형으로 구분하는 결정 경계(Decision Boundary)를 찾는 것이다. Support Vector Machine(SVM)는 이런 결정 경계를 최적화하여 최대 마진을 갖도록 만드는데 주로 중점을 둔다. 마진은 결정 경계와 가장 가까운 데이터 포인트(서트 벡터)간의 거리로 정의되면, 이 거리가 최대가 되게 만드는 경계를 찾는다.

SVM알고리즘은 선형 및 비정형 분류 문제에 모두적용 가능하다는 장점을 가지고 있다. 또한 최적 분류 경계를 찾기 위해 마진을 최대화하는 성향 덕분에 일반화 성능을 향상시키는 알고리즘이라고 할 수 있다. 더불어 상대적으로 적은 수의 서포트 벡터만 사용하여 효율적인 예측이 가능하다, 다만, 데이터의 특성 수가 샘플 수 보다 훨씬 많을 경우 성능저하가능성이 존재하며, 매개변수 설정에 민감하고 최적 매개변수 선택이 어렵다는 단점이 있다.

Support Vector Machine(SVM)의 주요 이론과 개념은 다음과 같다.

(1) 선형 SVM : 이 경우, 결정 경계가 선형인 모델을 만든다. 데이터가 선형적으로 구분 가능한 경우 해당 모델을 사용할 수 있다.

(2) 비선형 SVM : 실제 데이터가 완전히 선형적으로 구분되지 않는 경우가 많습니다. 이 경우, Kernel Trick 이라는 기법을 사용하여 데이터를 고차원 공간으로 매핑한 뒤 선형 결정 경계를 찾는다.

(3) 이진 분류 확장 : SVM은 기본적으로 두 클래스를 구분하는 문제에 적용된다. 하지만, 멀티 클래스 분류 문제에 확장할 수 있는 두가지 기법이 있다. 이에는 일대다(One-vs-All) 및 일대일(One-vs-One) 방식이 있다.

1. 연구 방법

**3.1 데이터 수집 및 전처리 개요**

본 과제에서는 분류기를 학습하기 위해 KADX(농식품 빅데이터 거래소)에서 제공되는 만개의 레시피 서비스를 통해 수집된 레시피 데이터를 활용한다. 본 데이터는 레시피 일련번호, 레시피 제목, 요리명, 등록자 ID, 스크랩수, 요리 소개, 요리 재료 등 총 18개의 파라미터로 구성되어 있으며, 데이터 특성 및 주요 자료를 확인하여 ‘요리명, 조회수, 추천수, 스크랩수, 요리방법별명, 요리상황별명, 요리재료별명 등 총 12개의 주요한 파라미터를 활용해 데이터를 분석했다.

<Table 1> Original Data Set텍스트, 영수증, 폰트, 화이트이(가) 표시된 사진

자동 생성된 설명

또한 약 130,000개의 데이터로 구성된 데이터를 특성을 파악하고, 컬럼의 값들 중 문자로 된 카테고리을 각각 라벨값 부여를 통해 숫자로 정의했다. 또한, 결측된 값 중 평균데이터로 대체할 수 없는 문자 값들은 삭제하고 데이터 분석을 진행했다. 그 외의 갑들은 평균 값으로 대체하여 결측치를 제거하는 과정을 통해 데이터를 분석할 수 있게 전처리하는 과정을 거쳤다.

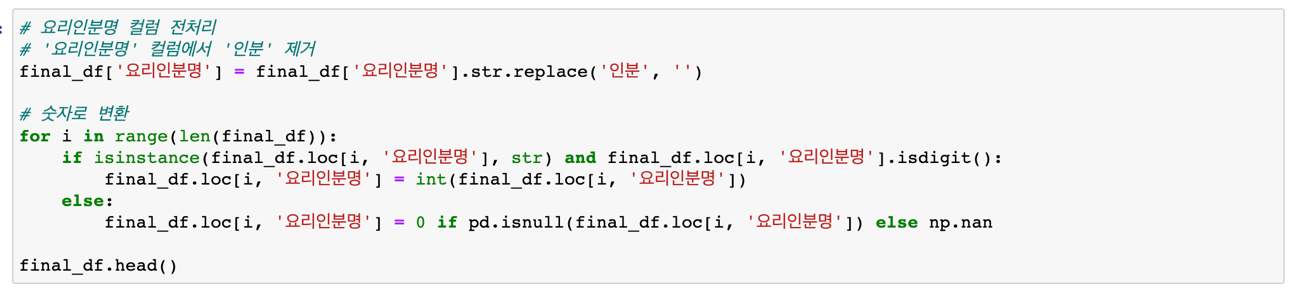
**3.2 주요 파라미터 전처리**

<Table 2> Data Set텍스트, 스크린샷, 영수증, 화이트이(가) 표시된 사진

자동 생성된 설명

< Table 1> 은 수집한 데이터의 특성을 보여주는 표이다. 초기 18개였던 파라미터 값들을 확인한 후, 레시피 번호, 최초등록 일시와 같이 데이터 전처리에 필요하지 않다고 판단되는 파라미터는 삭제하였다. 또한, 데이터 마이닝 수업 내에서 배우지 배우지 않았기 때문에 심도있게 다루지 못하는 비정형 텍스트로 구성된 요리소개 컬럼 또한 삭제하였다. 요리명, 조회수, 스크랩 수와 같은 내용들은 그대로 유지하였으며, 요리 재료의 내용은 | 를 기준으로 요리재료의 개수를 count하여 전처리하는 과정을 거쳤다. 그 외 주요 파라미터들 역시 .unique를 통해 컬럼에서 고유하게 언급되는 종류를 파악하여 항목별로 라벨을 정리하는 과정을 거쳤다.

<Code1> 텍스트, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

<Code2>

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

**3.3 최종 데이터의 결측치 처리 및 숫자 타입 확인 및 변환**

모델을 돌리기 전, 최종적으로 선택된 데이터들의 정보를 확인했다. 결측치가 있는지 확인하고 전처리하는 작업을 수행했다. 평균으로 대체가 가능한 값은 평균값을 활용하여 주요 값을 대체했다.

하지만, 텍스트이기 때문에 평균이나 대체할 수 없는 값들에 대해서는 행자체를 제거하여 모델을 돌리는데 문제가 없을 수 있도록 처리하는 과정을 거쳤다.

<Code3>

텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

마지막으로 info()를 통해, 데이터의 기본정보를 확인했다. 각 컬럼의 데이터 타입을 확인하고 모델을 돌리고, 분석에 적합한 데이터 타입으로 타입을 변환하는 과정을 거쳤다.

**3.4 최종 데이터의 분포 및 관계 파악**

**(1) 데이터간 상관관계 파악(correlation)**

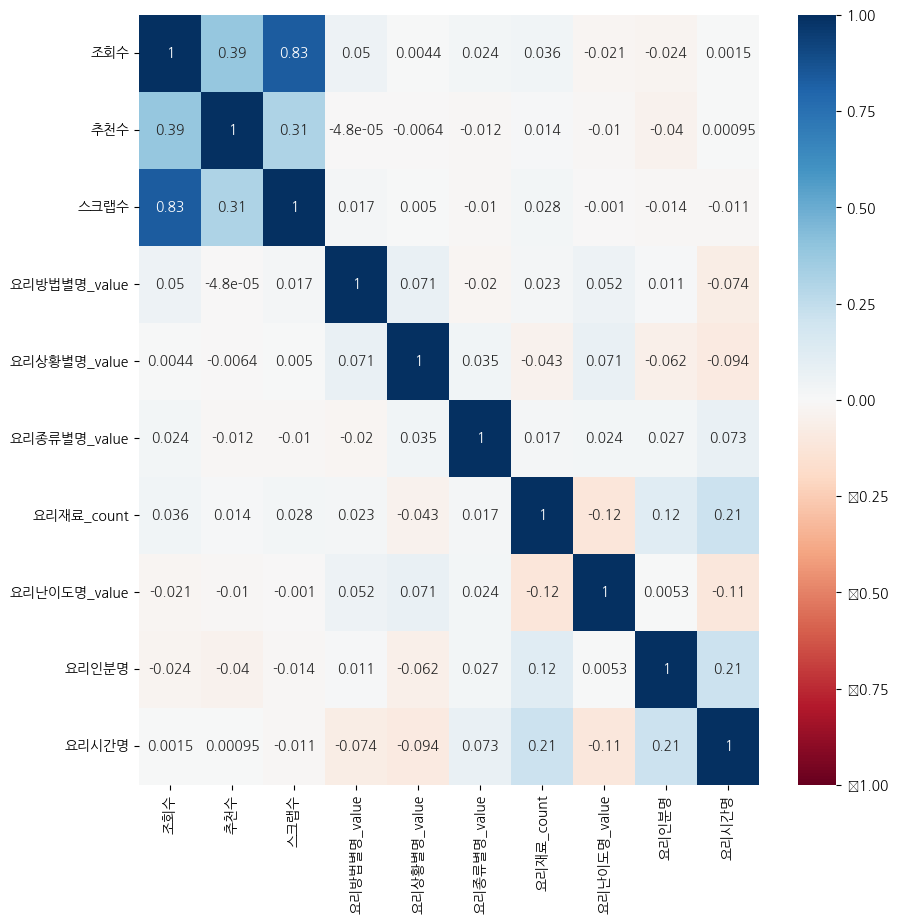
상관관계는 그 값이 0보다 큰지 작은지를 비교를 통해 두 Attribute 사이의 양과 음의 관계를 알아보는데 사용한다. R 값인 상관계수는 X와 y가 완전히 동일하면 +1, 전혀 다르다면 0(무상관관계), 반대 방향으로 완전히 동일하면 -1을 가진다. 즉, 1과 -1에 가까울수록 두 변수는 밀접하게 관계가 있다고 볼 수 있다.

분류하려는 target이 조회수이기 때문에, 다른 attribute와 조회수 간의 상관관계를 살펴보았다.

그 결과 상관계수가 양수인 attribute와 음수인 atrribute를 구할 수 있었으며, 상관계수의 절댓값이 0.2 이상으로 어느 정도의 상관관계가 있으리라 예상되는 Attribute 역시 추출하였다.

< Figure 1> < Table 3>

|  |  |
| --- | --- |
| **기준** | **Attribute** |
| Corr> 0 | 추천수, 스크랩수, 요리방법, 요리상황별명\_value, 요리종류 별명 \_value,요리재료 \_count |
| Corr> 0 | 요리인분명, 요리난이도명 |
| |Corr| > 0.2 | **추천수, 스크랩수** |



상관관계를 기반으로 데이터셋을 하나씩 확인해본 결과, 스크랩 수 attribute는 상관관계가 조회수랑 매우 높은 관계를 보이는 것으로 나타났다. 하지만, 사실상 조회수가 높을 수록 스크랩수가 높을 가능성이 높기 때문에 모델 평가에서 실질적으로 영향을 미치는 컬럼이 무엇인지 정확히 확인하기 어려울 수 있다는 점을 감안해 모델을 돌리기 전 데이터셋 정의에서 스크랩 수 컬럼을 제외하였다.

**(2) 컬럼별 데이터 값의 분포 확인**

이후, target값과 높은 상관관계를 보인 추천수, 스크랩수의 box-plot을 그려 데이터의 산포를 확인했다. 더불어 target 값으로 생각하고 있는 조회수는 다른 attribute에 비해 더 나은 분포를 갖고 있는지 확인하기 위해 box plot을 그려 데이터의 분포를 확인하였다. 추천수, 스크랩수, 조회수에 대한 분포는 < Figure 2>와 같이 각각 regular scale과 log scale로 나누어 두가지 케이스로 시각화했다.

텍스트, 라인, 도표, 그래프이(가) 표시된 사진

자동 생성된 설명

< Figure 2>

텍스트, 도표, 라인, 스크린샷이(가) 표시된 사진

자동 생성된 설명

< Figure 3>

텍스트, 도표, 라인, 그래프이(가) 표시된 사진

자동 생성된 설명

< Figure 4>

< Figure 4>와 같이 다른 데이터 분포에 비해 조회수는 log scale 값으로 변화 했을 때, 이상치가 많다는 점을 제외하면 데이터의 분포가 적절하게 분포되어 있다는 것을 확인할 수 있었다.

**3.5 분류 모델을 위한 Target값 구분**

앞서 4분위수 기반의 Box-Plot을 조회수, 추천수 등과 같은 값에 대해 그려보았다. 위에서 본 것과 같이 다른 데이터는 log scale로 변환했음에도 좋지 않은 분포를 갖고 있어 조회수를 주요로 삼아 데이터를 분류하는 것이 적합하다고 판단했다.

< Table 2> 데이터세트를 확인하면, 데이터의 값들이 연속형 변수로 나타나 있기 때문에 분류에 적합하도록 조회수를 4분위수를 기준으로 범위를 나눠 구분하는 과정을 거쳤다. 먼저 조회수 사분위수를 확인하여 나온 값들은 각각 4165.0(Q1) 8487.0(Q2) 17317.75(Q3) 352628.0(Q4) 13152.75(Q3-Q1) 의 값으로 이루어져 있는 것을 확인했다. 해당 값들을 기준으로 for문을 활용하여, 조회수 값이 작은 것(Q1:1분위수)는 가장 낮은 4등급, 조회수가 가장 높을 때(Q4: 4분위수)는 1분위수를 주는 규칙을 만들어 분류 값을 할당하는 과정을 수행했다. 이후, < Table 3>과 같이 규칙에 의해 구분된 데이터를 ‘조회수\_분류값’이라는 파생 컬럼으로 만들어 원 데이터셋에 결합했다.

<Table 3>

텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

1. 실험 결과

Data가 label을 갖고 있기 때문에 Supervised Learning을 진행했다. 순서대로 Decision Tree를 진행한 후, Support Vector Machine을 활용해 분류를 진행했다.

**4.1 Decision Tree (의사결정 나무)**

우선, 그리드서치(GridSearchCV)를 사용하여 결정트리(DecisionTree) 모델의 최적 하이퍼파라미터를 탐색하고, 탐색 결과를 데이터프레임에 저장해 평균 테스트 점수에 따라 정렬하는 과정을 거쳤다.

모델을 돌릴 파라미터 값들을 사전 타입으로 정의하여 max\_depth를 특정 값으로 지정했을 때 나오는 값들을 확인하여 해당 값들을 기반으로 홀수이면서, 너무 깊지 않은 5를 max\_depth로 지정하여 모델을 돌렸다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 메뉴, 번호이(가) 표시된 사진

자동 생성된 설명

<code> Hyperparameter tuning <Table 4> Results

아래 <Figure 5>는 max\_depth= 5를 기준으로 Decision Tree를 활용하여 데이터를 분류하고, 해당 규칙을 시각화한 그림이다. Attribute 들이 수치형 자료들과 category가 다수인 경우가 많기 때문에 gini 대신 entropy를 기준으로 Decision Tree를 만들었다. 그림에서 확인할 수 있는 것처럼 가장 조회수와 관계가 높으며 상위 노드에 해당하는 것은 ‘추천수’이고 그 다음으로 주요하게 데이터들을 분류하는 기준이 되는 항목은 ‘요리상황별명\_value’와 ‘요리인분명’, ‘요리재료 개수’가 있었다.

라인, 도표, 화이트, 디자인이(가) 표시된 사진

자동 생성된 설명

<Figure 5> Decision Tree Visualization

이후, Decision Tree 분류기에 대해 5-fold 교차검증을 수행했다. 모델은 각 폴드에서 정확도 점수가 순서대로 ‘0.585’, ‘0.583’, ‘0.579’, ‘0.584’, ‘0.584’가 나왔다. 이 모델의 전체적인 정확도는 0.583으로 평균 정확도이며, 표준편차는 0.004이다. 결과 값을 통해, 대체적으로 모델의 정확도는 평범한 편이지만 모델의 안정성은 높은 편이라고 판단할 수 있었다.

텍스트, 영수증, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

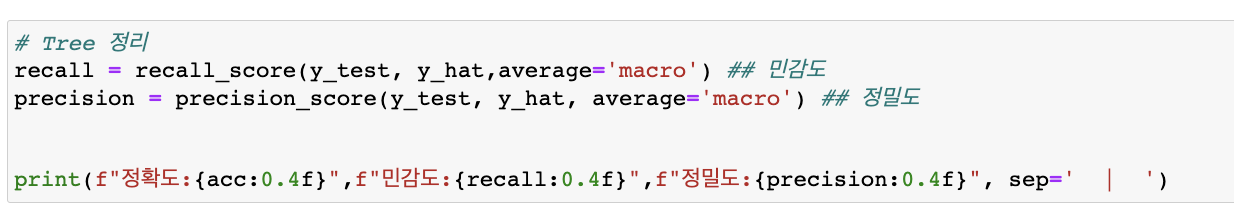
<Figure 6> Decision Tree 5-fold cross-validation Result

해당 5-fold 교차검증 기반의 정확도를 시각화해 나온 값은 <Figure 7>과 같다.

텍스트, 스크린샷, 라인, 도표이(가) 표시된 사진

자동 생성된 설명

<Figure 7> Decision Tree 5-fold cross-validation Visualization



<Figure 8>Code Decision Tree Result

결론적으로 Decision Tree model을 활용해 데이터셋을 분류한 결과에 대한 정확도, 민감도, 정밀도는 다음과 같았다. 정확도는 0.518, 민감도는 0.258, 정밀도는 0.2520으로 매우 불완전한 모델 성능을 보였다. 이를 통해 다른 모델을 활용하는 것이 적합할 것이라는 판단을 내렸다.

**4.2 SVM (Support Vector Machine)**

우선, SVM 모델링 전 scikit-Learn의 하이퍼파라미터 튜닝 도구인 ParmeterGrid를 활용하여 복수 파라미터를 최적화하는 과정을 거쳤다. 각각 6개의 후보자에 대해 5개의 폴드르 이용하여 총 30번의 학습을 수행했다.

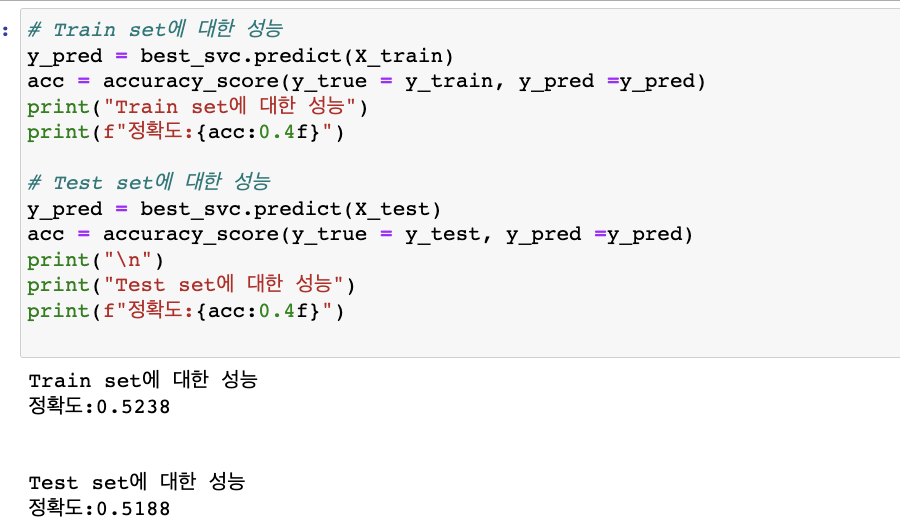
텍스트, 스크린샷, 폰트, 영수증이(가) 표시된 사진

자동 생성된 설명

<Figure 9> Code Decision Tree Result

해당 수행에 대한 결과는 <Figure 9>와 같다. 이후 ParmeterGrid를 활용해서 나온 결과를 기반으로 최적의 서포트 벡터 머신 모델을 사용하여 훈련 데이터인 X\_train과 y\_train을 학습했다.

이를 기반으로 Train set에 대한 성능(정확도)와 Test set에 대한 성능(정확도)를 확인했다. 확인 결과 나온 값은 <Figure 10>과 같다.



<Figure 10> SVM train set, test set accurcy

1. 결과 해석

본 과제에서는 분류기를 학습하기 위해 KADX(농식품 빅데이터 거래소)에서 제공되는 만개의 레시피 서비스를 통해 수집된 레시피 데이터를 활용했다. Decision Tree와 Support Vector Machine 두가지 모델을 활용하여 데이터 분류를 학습하는 과정을 거쳤다.

결과적으로 나온 데이터 값들을 확인했을 때, Decision Tree와 Support Vector Machine 의 결과가 모두 좋지 않아, 해당 데이터를 분류하는 모델로는 적합하지 않았을 수 있다는 생각을 했다. 실제 모델 성능 평가 결과는 다음과 같다.

텍스트, 폰트, 라인, 스크린샷이(가) 표시된 사진

자동 생성된 설명<Figure 11> Decision Tree

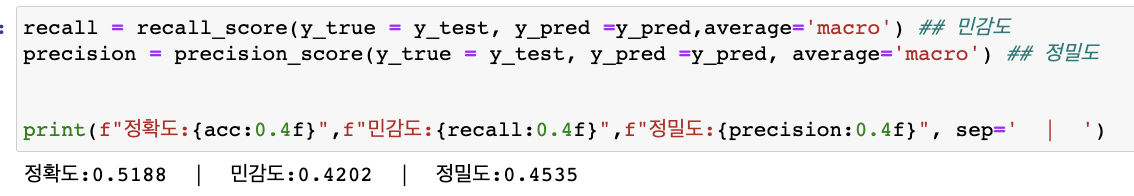
텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

<Figure 12> Decision Tree

Decision Tree의 전체 적인 모델 정확도는 0.52로 나타났다. 각 클래스 1~4에 대한 성능 자체는 클래스 1에 대한 정밀도 0.63을 제외하고, 2,3,4에서 모두 0.5수준에 미치는 값으로 나타나고 있다. 종합적으로 전체 모델의 성능이 상대적으로 낮은 수준으로 평가된다고 볼 수 있다. 클래스 4에 대해서는 비교적으로 높은 재현율과 F1-점수를 보여주고 있지만, 다른 클래스들은 상대적으로 낮은 성능을 보이고 있다.

다음으로 Support Vector Machine의 결과는 다음과 같다.



<Figure 13> Support Vector Machine

텍스트, 영수증, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

<Figure 14> Support Vector Machine

Support Vector Machine의 전체적인 정확도는 0.52로 나타났다. 앞선 모델의 결과와 동일하게 종합적으로 전체 모델의 성능이 상대적으로 낮은 수준으로 평가된다고 볼 수 있다. 클래스 4에 대해서는 비교적으로 높은 재현율과 F1-점수를 보여주고 있지만, 다른 클래스들은 상대적으로 낮은 성능을 보이고 있다.

결론적으로, 두 모델 모두 낮은 성능을 보이고 있으며 추후 모델 성능을 향상시키기 위해 파라미터를 바꾸거나 축소시켜 모델을 돌려보는 방식의 수행을 통해 성능 개선을 기대해볼 수 있을 것이다. 혹은 랜덤 포레스트 등을 고려하거나, 앞선 Support Vector Machine의 커널 타입이나 c값을 조금 더 조정해본다면 보다 나은 결과가 나올 수도 있을 것이라고 생각한다.

1. 결론(프로젝트 고찰)

본 과제에서는 요리커뮤니티 서비스(이하, 만개의 레시피) 내 레시피 데이터를 활용해 인기 요리를 분류하는 과정을 진행하였다.

위 과정을 수행하며 우리는 데이터를 단순히 코드를 입력하고, 모델을 돌려 숫자를 확인하는 역할이 아니라는 것을 분명히 이해할 수 있었다. 제약된 시간상Decision Tree와 Support Vector Machine두가지만을 활용했지만 팀프로젝트 혹은 조금 더 장기 프로젝트로 꼼꼼히 확인할 수 있는 시간이 있었다면 해당 자료들을 조금 더 촘촘하게 전처리하고, 다양한 방식으로 파라미터들을 수정하며 모델을 돌리고 수치를 시각화해보는 시간을 가질 수 있었을 것이라고 생각했다.

끝으로, 약 12000개의 데이터가 존재하여 다양한 방식으로 데이터를 활용할 수 있다는 장점이 있었지만, 텍스트형 데이터로 이루어진 컬럼들을 라벨링 값으로 변환하여 데이터를 분석해야 했기 때문에 더 깊은 의미의 데이터들 안의 관계나 규칙을 알 수 없다는 점이 아쉬웠다. 실제로 유사한 서비스 유저인터뷰 결과를 보면, 요리 커뮤니티를 통해 정보를 얻으려는 사용자들이 고려하는 항목 중 주요한 것은 ‘레시피 재료의 주요 항목’ 이었다. 이는 사용자들이 가정 내 냉장고 속 식품들을 활용하여 조리할 수 있는 식품인지 확인하거나 쉽게 구할 수 있는 재료 여부에 따라 해당 레시피를 조회하거나, 스크랩 할 수 있다는 것을 의미하기 때문에 데이터에 분명한 영향을 끼쳤을 것이라고 생각한다. 하지만 본 과제에서는 데이터를 텍스트 기반으로 분석한 것이 아닐 뿐만 아니라 재료들을 단순 개수로 전처리해 분석했기 때문에 데이터가 내포하고 있는 실제 의미를 놓쳤을 수 있다고 생각한다.

다양한 모델을 비교하며 모델의 예측 성능을 평가하여 최적의 모델을 찾고, 예측 결과를 해석하는 과정에서 유의미한 변수들을 식별하여 개인 사용자들이 흥미를 느끼는 레시피 데이터를 이해하는데 유의미한 결과를 도출하고자 했는데, 초기 목표에 비해 모델을 돌리기 위한 전처리를 거치며 실제 텍스트 데이터 내에 내포하고있을 내용들을 놓친 것 같아 아쉬웠다. 이를 통해 텍스트 마이닝을 활용해 사람들의 실제 언어를 기반으로 데이터를 분석하는 방법을 공부한다면, 동일한 데이터를 활용해 실제 데이터가 내포하고 있는 다양한 정보들을 파악할 수 있을 것이라고 생각했다. 더 나아가 본 데이터를 활용한 텍스트 마이닝 결과물에 해당 레시피 글에 달린 텍스트들을 분석할 수 있다면, 실제 앞서 과제를 진행하면서 기대했던 커머스 영역으로의 확장 뿐만 아니라 사용자에게 더 유의미하고 도움이 되는 양질의 자료들을 분류해서 제공하고 이를 통해 더 나은 서비스로 개선할 수 있을 것이라고 생각했다.

1. 참고문헌

[1] 정원찬, 최경수, 김정호 .(1993).Decision Tree 를 이용한 Machine Learning.[ETRI] 전자통신동향분석,8(4),0-0.

[2] 오승환, (2019), ’파이썬 머신러닝 판다스 데이터분석’, 정보문화사

데이터 출처: <https://kadx.co.kr/product/detail/0c5ec800-4fc2-11eb-8b6e-e776ccea3964>