|  |
| --- |
| **기계학습을 활용한 써브웨이 샌드위치**  **조합 추천 모바일 애플리케이션** |
|  |
| **Subway sandwich combination recommendation**  **mobile application using machine learning** |
| **경희대학교 컴퓨터 공학과**  2017103999 신민식  2018110653 이민영 |
| **요 약**  세계에서 매장 수가 가장 많은 패스트푸드 전문점인 “써브웨이”는 빵, 소스 등의 재료 등을 전부 고객이 직접 선택하게 하는 것이 특징이다. 재료 선택이 복잡해서 불편함을 느끼던 사람들을 위해 사용자 개인의 입맛 취향을 알아내고, 취향에 맞는 샌드위치 재료 조합을 추천해주는 모바일 애플리케이션을 제안한다. 이 때, 유사도 분석과 기계학습을 이용한 학습모델을 통해 재료 조합 추천 기능을 구현한다. |

**1. 서 론**

**1.1 연구 배경**

세계에서 가장 매장 수가 많은 패스트푸드점인 “써브웨이”는 다양한 속재료, 빵, 소스 등을 하나하나 고객이 직접 선택해야 한다는 점이 특징이다. 이 특징이 단점으로 오히려 작용하여 주문 시간이 길어지고, 써브웨이 이용을 기피하는 고객들도 있을 것이라는 생각을 하게 되었다.

이것의 사실여부를 확인하기 위하여 써브웨이의 고객과 비고객(총 32명)을 대상으로 사전 설문조사(9/16~9/23 진행)를 진행했고, 총 답변자수 32명 중 써브웨이를 이용하지 않는 17명, 그 중에서 “메뉴 선택이 복잡해서”를 이유로 꼽은 사람은 12명으로, 써브웨이를 이용하지 않는 가장 주된 이유는 “메뉴선택이 복잡”하기 때문이라는 결과가 나왔다. 즉, 선택이 복잡해서 주문이 어렵고, 입맛에 맞는 샌드위치 조합을 맛보기도 힘든 것이다.

따라서, 사람들 각자의 입맛에 맞는 써브웨이 샌드위치 재료 조합을 추천해주는 애플리케이션을 고안하게 되었다. 또한, 이 애플리케이션이 써브웨이의 복잡한 메뉴 선택에 어려움을 겪던 기존 써브웨이 고객층과 비고객층 모두에게 도움을 주기를 기대한다.

**1.2 연구 목표**

이 연구의 목표는 써브웨이의 메뉴 조합을 선택할 때, 개인 취향에 맞춰 샌드위치 재료의 조합을 추천해주는 모바일 앱을 구현하는 것이다.

**2. 관련 연구(기존의 유사한 앱)**

써브웨이 코리아 공식 애플리케이션인 ‘써브웨이’1의 경우는 회사 차원에서 제공하는 기본적인 추천 조합이 존재하지만, 고객 “개인”의 입맛에 맞게 조합을 추천해주는 기능은 없다. 플레이스토어 애플리케이션인 ‘미식큐’2 와 ‘맛있는 취향’3은 일반적인 식사메뉴를 개인의 입맛에 따라 추천해준다. 다만, 한 번 취향이 입력되면 수정하는 것이 어려워 입맛의 변화를 적용하기 힘든 단점이 있다. 그러므로 우리는 기존에 없던 “써브웨이 샌드위치의 재료 조합을 사용자 개인의 취향에 맞춰 추천해주는 앱”을 구현하려고 한다.

**3. 프로젝트 내용**

**3.1 써브웨이 샌드위치의 주문 방법**

써브웨이 매장에서 샌드위치를 주문하는 순서는 다음과 같다.4

1) 샌드위치의 주 재료를 선택한다.(18종류 중 택1)

2) 빵의 종류를 선택한다.(6종류 중 택1)

3) 치즈의 종류를 선택한다.(3종류 중 택1)

4) 추가할 재료가 있다면 선택한다.(7종류 중에서 선택)

5) 제외할 야채가 있다면 선택한다.(8종류 중에서 선택)

6) 소스를 선택한다.(15종류 중에서 복수 선택 가능)

**3.2 시나리오**

이 프로젝트의 주요 기능은 애플리케이션 사용자 각자의 입맛에 맞춰 써브웨이 샌드위치의 재료 조합(주 재료 + 소스)을 추천하는 것이다. 앱의 작동 순서는 아래와 같다.

1) 주 재료 추천[그림1]

코사인 유사도 분석을 통해 주 재료를 추천한다.

첫째, 앱 사용자에게 가장 선호하는 식재료와 가장 좋아하는 맛을 물어보고 답변(문자열 형태)을 받는다.

둘재, 위에서 받은 답변과 샌드위치 주재료(18종) 각각의 유사도를 분석한다.

셋째 , 가장 높은 유사도를 가진 주재료 1,2,3 순위를 뽑는다.

2) 주 재료에 어울리는 소스 추천[그림2]

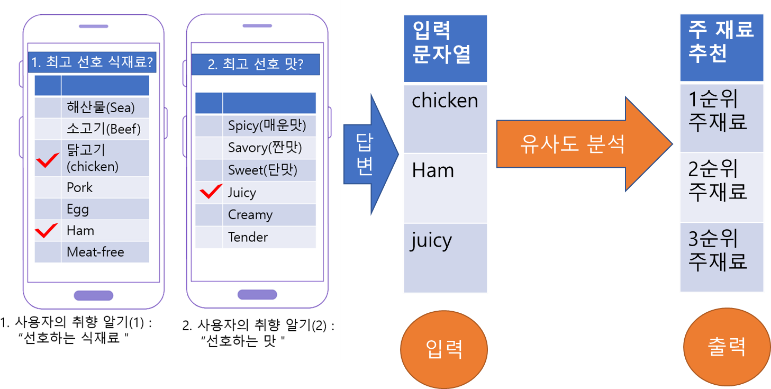
직접 제작한 기계학습(Multi-label Classification)모델을 통해 주재료에 어울리는 소스를 추천한다.

첫째, 1)에서 뽑힌 1순위 주재료를 기계학습 모델에 입력하고, 1순위 주재료에 어울리는 소스를 출력한다.

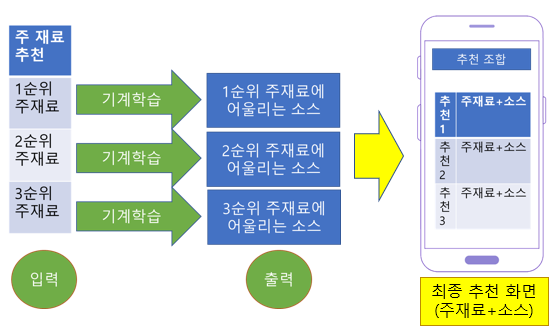
둘째, 2순위 주재료에 어울리는 소스를 위와 같은 방식으로 출력한다.

셋째, 3순위 주재료에 어울리는 소스를 위와 같은 방식으로 출력한다.

3) “1순위 주재료 + 어울리는 소스”, “2순위 주재료 + 어울리는 소스”, “3순위 주재료 + 어울리는 소스” 를 최종적으로 애플리케이션 화면에 출력한다. [그림2]



[그림1] 주 재료를 추천하는 과정



[그림2]주재료에 어울리는 소스를 추천하고,

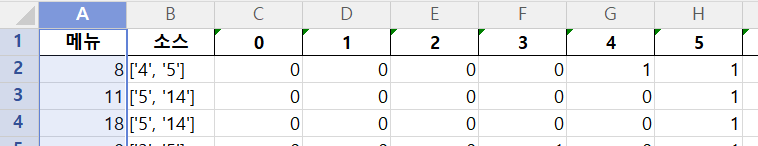
최종 추천 조합(주재료 + 소스)을 화면에 출력하는 과정

**3.3 데이터 수집(기계학습에 사용)**

써브웨이 이용자들이 추천하는 “주재료 + 소스”의 조합들을 데이터로 수집한다. 수집한 데이터는 엑셀로 정리해 기계학습 모델(주재료에 어울리는 소스추천)에 사용한다.[그림2,3]



[그림2]주재료/소스 종류 각각에 번호를 매긴다.



[그림3]정리한 데이터의 예시

**3.4 코사인 유사도를 이용한 주재료 선정**

**• TF-IDF 벡터화**

TF-IDF(Term Frequency – Inverse Document Frequency) 인코딩은 단어의 개수를 그대로 세지 않고 모든 문서에 공통적으로 포함되어 있는 단어의 경우, 문서 구별 능력이 떨어진다고 판단해 오히려 가중치를 축소하는 방법이다. 따라서 TF-IDF 값이 낮으면 중요도가 낮은 것이며, TF-IDF 값이 크면 중요도가 큰 것이다.

아래와 같은 예시가 있다고 가정해보자.

***문서1 : 먹고 싶은 사과  
문서2 : 먹고 싶은 바나나  
문서3 : 길고 노란 바나나 바나나  
문서4 : 저는 과일이 좋아요***

그렇다면 일반적인 DTM(Document-Term Matrix, 문서 단어 행렬)은 다음과 같다.



[그림4] 예시 문서에 대한 DTM

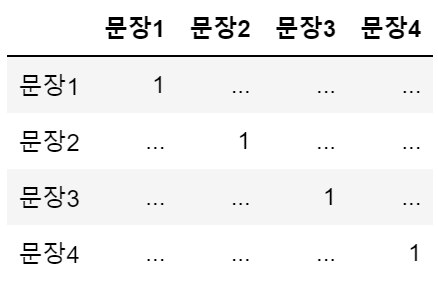
TF-IDF계산 결과는 다음과 같다.



[그림5] 예시 문서에 대한 TF-IDF

이를 구현하기 위해 Scikit-learn API의 TfidfVectorizer 를 이용한다.

TF-IDF Matrix가 생성되면 Scikit-learn API의 linear\_kernel을 이용하여 코사인 유사도를 계산할 수 있게 된다. 결과는 다음과 같은 형태를 가진다.



[그림6] 코사인 유사도 예시

이 프로젝트에서는 사용자의 취향을 문자열 형태로 입력 받아서 각 메뉴에 대한 설명 문자열과 비교한 후, 유사도가 가장 높은 주재료를 선정한다.

**3.5 Multi-label Classification을 이용한 소스 선정**

머신 러닝 모델의 입력은 메뉴이고, 이에 대한 출력은 소스이다. 하지만 하나의 메뉴에도 여러 종류의 소스가 들어갈 수 있다.따라서 우리는 Multi-label classification을 할 필요가 있다. 한 영화가 액션, 어드벤처, 로맨스 등 여러 개의 장르에 동시에 해당될 수 있는 것처럼 Multi-label classification은 결과 값이 여러 label에 해당될 수 있다.

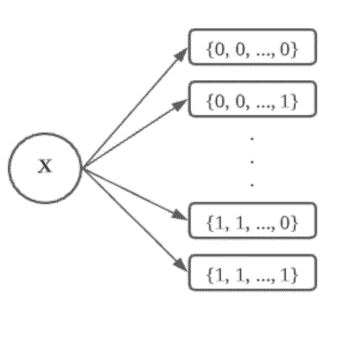
Multi-label classification는 여러 가지 방법으로 구현이 될 수 있는데, 대표적으로Binary Relevance, Classifier chain, Label Powerset 등이 있다.

**• Binary Relevance**

Binary Relevance는 결과에 해당하는 label들을 분리하여 각각의 single class로 표현한다. 그 다음, 입력에 대해 각 label 별로 single-class classification을 한 후에 이를 합치게 된다. 하지만 이 방법의 문제점은 각 label이 독립적이지 않을 수 있음에도 이를 무시한다는 것이다. 이를 보완하기 위한 방법으로 label powerset, classifier chain 등이 있다.

**• Label Powerset**

Label Powerset은 label의 가능한 모든 조합들을 각각 하나의 class로 치환하여 classification을 수행한다. 이를 그림으로 표현하면 다음과 같다.

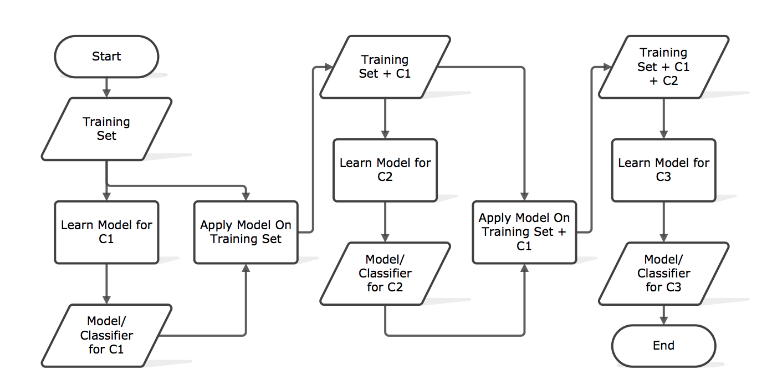


[그림7] Label Powerset

이렇게 label 집합을 예측하면 label들 간의 관계도 포함하여 예측이 가능하다는 장점이 있다. 하지만 공간 복잡도가 높다는 단점이 있다. Label이 d개 있다면 가능한 label 집합은 2d개가 된다.

**• Classifier Chain**

Classifier Chain에서는 여러 개의 classifier을 사용한다. 순서에 따라 레이블들을 예측하는데 한 레이블을 예측할 때, 이전에 예측한 레이블들까지 포함하여 예측에 활용한다. 이를 그림으로 표현하면 다음과 같다.



[그림8] Classifier Chain

이 방법은 레이블 간의 관계도 담아내면서도 공간 복잡도를 줄일 수 있다는 장점이 있다. 하지만 label 예측 순서에 따라 성능이 달라질 수 있다는 단점이 있다.

**3.6 개발환경**

운영체제: 안드로이드 5.1 (API 22)

개발언어: JAVA , Python

개발도구: Android Studio, Jupyter Notebook, VS Code

**4. 결론**

샌드위치 재료선택의 복잡함 때문에 써브웨이 이용을 꺼리는 사람들이 많이 있다. 이를 해소하기 위해, 본 프로젝트에서는 샌드위치의 주재료와 그 주재료에 어울리는 소스를 각각 추천해주는 앱을 제안한다. 이 때, 주재료는 유사도 분석을 통해서 추천하고, 주재료에 어울리는 소스는 Multi-label Classification모델을 제작해서 추천한다.

**5. 참고자료**

1. 써브웨이 공식 애플리케이션:

<https://play.google.com/store/apps/details?id=kr.co.subway.preorder>

1. 맛있는 취향 애플리케이션 : <https://play.google.com/store/apps/details?id=com.mytaste>
2. 미식큐 애플리케이션 : <https://play.google.com/store/apps/details?id=kr.misiq>
3. 써브웨이 코리아 공식 사이트(2022-09-15) : <https://www.subway.co.kr/>
4. 머신러닝: https://ikkison.tistory.com/51?category=785277
5. 본 논문 프로젝트 github  
   <https://github.com/MS1027/subwayapp/>