추천 모델

이 문서는 추천 알고리즘의 기획을 위해서 작성됐다. 팀원간 의견 공유를 목적으로 만든 문서이므로 문체가 통일되지 않았지만 현재 고려 중인 사항을 전달하기 위해 제공한다.

딥러닝

처음 고려한 내용으로 사용자의 특성과 칵테일의 특성을 연관성이 존재한다는 가정하에 처음 고려된 모델이다.

신경망을 택한 이유는 다음과 같다.

1. 사용자가 입력한 선호와 칵테일이 갖고 있는 정보가 동일함을 증명할 수 없다.
   사용자가 단맛을 4 정도로 선호한다고 생각하여 초기 값을 입력했을 때 칵테일 DB에 있는 준벅의 단맛이 5이지만 선호할 가능성이 있다.
2. 여러 요인들이 복합적으로 있어 특정 점수로 판별이 불가능하다.
   도수가 높은 술에선 단맛이 강한 술을 비선호하지만 도수가 낮은 술에서 단맛이 강한 술을 선호할 수 있다.

술의 특성

알콜도수, 단맛, 상큼함

~~회의 필요 내용 : 식감, 색~~

위 요소들의 선호들을 1~5까지 총 5가지 선호를 갖고 5가 가장 높은 선호

→ 전체적인 선호를 나타냄

인공지능 모델

술의 특성들이 독립변수로 주어지고 결과가 사람들의 선호에 따라 호, 불호가 나와야 함.

유저의 특성이 반영되게 끔 인공지능 모델을 설계할려면 어떻게?

1. 임베딩(고차원 데이터를 저차원으로) : 선호도를 벡터로 표현하여 사용자와 술 종류에 따라서 각각의 선호도를 표현함
2. 프로파일링 : 비슷한 선호를 가진 다른 사용자의 선호 칵테일을 추천

3. 입력 자체를 (사용자의 선호도(단맛, 상큼한맛, 알콜도수) + 예측할 칵테일의 특징(단맛, 상큼한맛, 알콜도수))로 설정

조사 결과

1. 임베딩은 선호도 각 요소가 한 차원을 갖고 있는 것은 아니다.
2. 임베딩을 하더라도 입력 자체는 사용자의 선호도 + 예측할 칵테일의 특징이다.
3. 학습할 때 임베딩할 차원수는 사용자 마음대로 정해도 된다.
   임베딩 차원 선택 시 고려사항
   • 데이터의 다양성 및 크기: 데이터의 종류가 많고, 복잡한 관계를 학습해야 할 경우 더 높은 차원을 선택하는 것이 좋음. 반대로 데이터가 단순하고 종류가 적다면 낮은 차원도 충분함.
   • 유니크한 값의 수: 일반적인 가이드라인으로는 각 유니크한 값의 수(예: 사용자 수, 아이템 수)에 대해 임베딩 차원을 log2(유니크한 값의 수) 정도로 설정하는 것이 권장됨.
   • 모델의 목적: 모델의 복잡도에 따라 임베딩 차원을 조정할 필요가 있음. 간단한 예측이 목적이라면 낮은 차원, 복잡한 패턴을 학습해야 한다면 높은 차원을 선택할 수 있음
   • 실험적 조정: 최적의 임베딩 차원은 주어진 데이터와 문제에 따라 다르므로, 실험적으로 여러 값을 시도하여 최적의 성능을 찾는 것이 일반적

결론

- 각 사용자는 술의 특징과 일치하는 선호도를 갖고 있다. 예를 들어, 단맛, 상큼한 맛, 알콜 도수
- 술도 동일한 특징들을 갖고 있다.
- 위 두 선호도 및 특징들이 모델의 입력값(독립변수)로 들어간다.
- 선호도 및 술의 특징이라는 데이터의 내재된 특징과 관계를 표현해야 하기 때문에 임베딩 기법을 사용한다.
  사용자의 입력을 바탕으로 초기 사용자 선호도를 설정하는데 이는 사용자마다 단맛-상큼한맛 둘다 좋아한다고 해서 (5, 5)로 입력할 지 (3, 3)으로 입력할 지 확신할 수 없기 때문에 데이터 간 특징을 고려해야 하기 때문이다.
- 모델의 출력 값은 선호도이다.
  예를 들어, 단맛 5, 상큼한 맛 3, 알콜 도수 1의 선호도를 가진 사용자 그리고 단맛 5, 상큼한 맛 4, 알콜 도수 1의 특징을 가진 블루 하와이를 입력으로 넣으면 0.976 과 같이 높은 선호도를 가진 값이 나온다. (선호도는 0~1 사이 값으로 예정)

추천 알고리즘

딥러닝을 고려하더라도 칵테일의 선호는 여러 요인들을 고려해야 하므로 한계점이 있다고 파악했다. 또한 피드백 결과 기분, 날씨, 요일, 앱을 켠 시간 등의 데이터를 활용하는 방향을 고려하며 알고리즘을 구상했다.

목적

prediction 일지 ranking, 이 두가지 분야에 대해서 고민해야 함. 현 Mixby는 유저의 선호도를 정확하게 예측하는 것이 목적이여야 하므로 preiction version of problem으로 접근해본다.

종류

협업 필터링

Collaborative Filtering

프로파일링처럼 비슷한 관심사를 가진 다른 사용자의 데이터를 바탕으로 추천해 주는 방식

메모리 기반

neighborhodd-based collaborative filtering algorithm 이라고 부르기도 함.

사용자 기반 추천

유저 간의 유사도가 높을 수록 가중치를 부여하는 방식, 같은 그룹의 다른 유저가 선호하는 아이템을 추천함.

아이템 기반 추천

한 아이템과 유사한 아이템을 선정하여 그 아이템 set으로 유저의 선호도를 예측

모델 기반

context를 기반하여 머신러닝이나 데이터 마이닝을 통해서 예측

한계

1. 콜드 스타트 : 데이터가 없는 상태에서는 제대로 동작하지 않음(신규 상황에서 동작 x)
2. 계산 효율 저하 : 사용자의 수가 많아질수록 계산 시간이 더 길어짐
3. 롱테일 : 소수의 인기 있는 항목에만 관심을 보여서 저조한 항목은 추천되지 못함

콘텐츠 기반 추천 시스템

사용자가 과거에 경험했던 아이템 중 비슷한 아이템을 현재 시점에서 추천하는 것

feature extraction, vector representation → 유저 선호 프로필이 파악 → cosine 유사도를 이용하여 유사 아이템을 선택

한계

콜드 스타트 문제를 해결할 수 있지만 다양한 항목을 추천하기에는 어려움

지식 기반 추천 시스템

하이브리드 방식

분석

칵테일 취향은 흑백 논리로 펼칠 수 없기 때문에 다방면으로 분석해야 하는 경우, 아래와 같은 경우가 발생한다.

> *나는 상큼한 준벅같은 칵테일도 좋아하지만 위스키 샷이나 갓파더와 같은 칵테일도 좋아하는데 어떻게 추천해야 할까?*

위 경우 다음과 같은 방식을 사용한다.

하이브리드 추천 알고리즘

콘텐츠 기반 필터링 + 협업 필터링

콘텐츠 기반 : 상큼한 술과 독한 술의 특성을 파악해서 각 특성 칵테일을 추천

협업 필터링 : 위 케이스와 비슷하게 상큼한 술과 독한 술 모두 좋아하는 다른 사용자의 취향을 반영하여 추천

어떤 필터링에 가중치를 주냐에 따라 어떤 술을 추천할지 갈림 (다중 취향 반영 모델이라고 함)

→ 예를 들어, 상큼한 칵테일을 먹은 연속된 날을 가중치로 둔다면 상큼한 칵테일을 먹다가 쿨이 돌면 독한 칵테일을 추천할 수 있음

클러스터링 기반 추천

사용자를 다양한 취향 그룹으로 분류

/ 상큼파 / 독주파 / 중도파 /

해당 그룹들로 나누고 소속된 클러스터에 맞는 칵테일을 추천

1. 사용자와 칵테일의 특성 데이터를 바탕으로 K-means, GMM(가우시안 혼합 모델)과 같은 클러스터링 알고리즘 적용
2. 취향이 유사한 사용자 그룹으로 나누어, 그 그룹 내에서 다른 사용자가 선호하는 칵테일을 추천 (협업 필터링)

결론

두 가지 결론이 존재

클러스터링 기반 + 협업 필터링

- 사용자를 다양한 취향 그룹으로 분류
- 각 칵테일을 어떤 그룹인지 분류 (신경망 써야 할듯?)
- 각 그룹내에서 어떤 칵테일을 추천할지 우선순위가 있어야 함 (재료 → 도수 → ... 이런 느낌으로)
- 우선순위의 가중치를 두는 방향을 고려한다면 신경망으로 어느 요인의 비중을 크게 두는게 가능

하이브리드 / 클러스터링 기반 + 벡터

- 도수, 당도, 산도의 값을 토대로 모든 레시피를 3차원 공간에 표현
- 위 세 값을 담은 벡터와 "테이스팅 노트"를 기반으로 한 가중치를 사용해 최적의 레시피를 추천
- 날씨 및 요일 별 가중치, 사용자의 취향 가중치를 통해 벡터의 시작점 위치 변경, 구체화된 레시피 목록을 도출할 수 있음