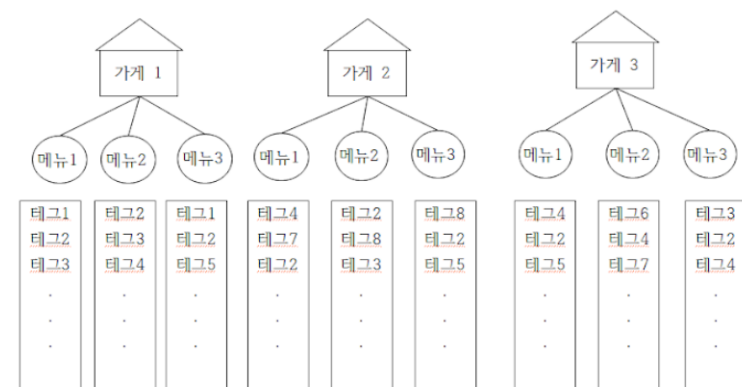
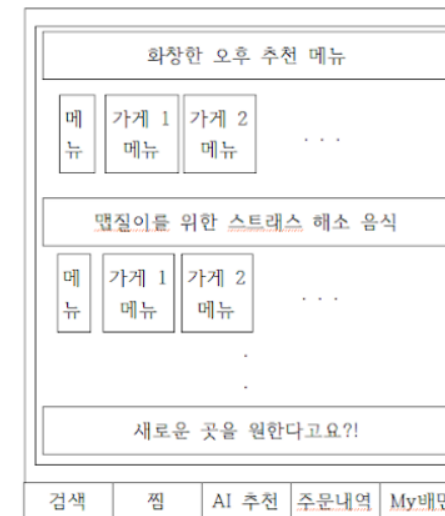


UI 예시



리뷰들을 통해서 태그 값 추출 (자연어 처리)

<필요데이터>

<과정>

1. 가게 데이터 처리

- 가게 전체 리뷰 데이터에서 사용가능한 리뷰 데이터 추출한다. 각 리뷰데이터를 자연어 처리하여 메뉴별 태그 생성.
- 가게 정보 추출 (가게 영업시간,)

2. 유저 데이터 처리

- 유저의 주문 내역 중 타지역 데이터는 제거한다. (전처리)

- 유저의 주문 내역을 통해 해당 가게의 정보 데이터와 메뉴 태그들을 상속받는다.
- 가게 정보를 기반으로 이용했던 가게들을 목록화 하고 각 이용 횟수를 더하여 백분율로 표현하고 가중치순으로 순서화한다.
- 또한 상속받은 태그는 메뉴(음식)와 속성 값 (맛, 재료 등)으로 나누어 적재한다.
- 동일한 태그들끼리 합하여 누적한 값을 백분율로 계산하여 가중치를 표현하고 순서화한다. (메뉴, 맛, 재료 등의 태그 모두 다)
- 유저 데이터는 총 2개의 매트릭스로 구성이 되는데 가게 <- 메뉴, 메뉴 <- 속성값으로 종속된다
- 첫번째 매트릭스에서 가게를 x축, 메뉴를 y축으로 구성한 뒤 관계값을 설정한다. (관계 있음 1, 관계 없음 0)
- 두번째 매트릭스에서 메뉴를 x축, 속성값을 y축으로 구성한 뒤 관계값을 설정한다.(관계 있음 1, 관계 없음0)

-

태그값 설정 이후 가게 가중치 순서도에 따라 유저의 성향을 2가지로 분류한다.

1번 매니아 :

2번 모험가

이용 가게 분포도에서 기울기가 1에 가까울수록 매니아 0에 가까울수록 모험가로 분류한다.

3. CF (협업 필터링) 기반 추천 (“당신이 좋아할만한 음식”)

- x축 (속성값), y축(메뉴), z축(가중치)로 3차원 그래프를 그려 군집화를 한다.
- 이후 분석가의 판단에 따라 차원 축소를 하여 그룹화를 진행한다.
- 비슷한 특성의 메뉴를 가진 다른 유저의 가게와 메뉴를 추천받는다. (CB 방법론)

4. CBF (컨텐츠 필터링) 기반 추천

- 사용자와 비슷한 특성을 갖는 가게의 메뉴를 추천한다. (족발 – 매운 의 조합이 유저의 랭킹 1위라면 ,,)

1번 메뉴와 관계된 가게를 뽑는다.

2번 가게 메뉴의 속성값이 유저와 같은 곳을 추천한다.

5. 통계학 기반 추천

- 사용자 별 날짜, 시간, 계절, 날씨 등에 따른 주기(패턴)을 확인
- 분류된 성향에 따라 매니아는 최대한 비슷하고 주문횟수가 많았던 곳으로 추천을 한다.
- 모험가의 경우 새로운 가게를 추천하는 비율을 높여 추천해준다.
- 패턴을 파악한 후 자주 이용했던 음식점 & 음식 그리고 그 가게와 비슷한 메뉴, 속성값을 가진 또 다른 가게를 추천한다.

6. 신규 가게 추천

신규 가게의 초기 메뉴 - 속성값은 배민 전체 가게의 메뉴 - 속성값들의 교집합으로 설정한다.

이후 통계학 기반 추천 시스템과 동일하게 사용자 패턴을 분석한다.

사용자 속성값과 비슷한 속성값을 가진 '신규' 가게 + 메뉴를 추천한다.

<요약>

가게 별 속성값을 가게 - 메뉴 - 속성값별로 분류하여

각 메뉴, 속성값 별 관계도를 매트릭스로 설정하며

고안한 알고리즘에 따라 자동으로 메뉴 추천을 한다.

또한 신규 가게의 초기값 설정을 자동화 할 수 있는 알고리즘을 고안하였다.

<리뷰 분석에 사용할 알고리즘>

BPE(Byte Pair Encoding): 서브워드(sub-word)단위로 분리하는 알고리즘

<태그 생성 과정>

사용자가 메뉴를 주문

-> 메뉴의 태그가 사용자에게 축적

-> 태그가 중복될수록 가중치가 증가

-> 사용자의 선호 태그 랭킹화

2. 이웃추천, 가게추천, 패턴추천 방식

[이웃추천]

비슷한 태그를 갖는 사용자 군집화 (여기서 사용자끼리 태그의 랭킹이 다를 텐데 랭킹이 비슷한 사람들끼리 2차 군집화?)

비슷한 태그를 갖는 이웃의 사용 빈도가 높은 가게 추천(if 사용 빈도 높은 가게가 없다면 해당 태그를 갖는 음식점을 추천)

[가게 추천]

- 사용자의 태그와 같은/비슷한 태그를 가진 가게를 추천

<사용할 알고리즘>

CBF(contents based filtering)

[패턴 추천]

- 사용자의 구매 패턴 분석 -> 특정 패턴을 보이는 메뉴/가게 추천
- 날씨, 시간대, 계절에 따른 메뉴/가게 패턴 분석
- 특정 주기를 가지는 패턴으로 개인화 추천 진행

<사용할 알고리즘>

둘 중에 더 결과가 좋은 모델 사용:

다중퍼셉트론(MLP) 모형 VS 순환신경망(RNN) 모형

반복 구매제품의 재구매시기 예측을 위한 두 모형의 성능비교

3. '매니아'와 '모험가'의 성향 분류

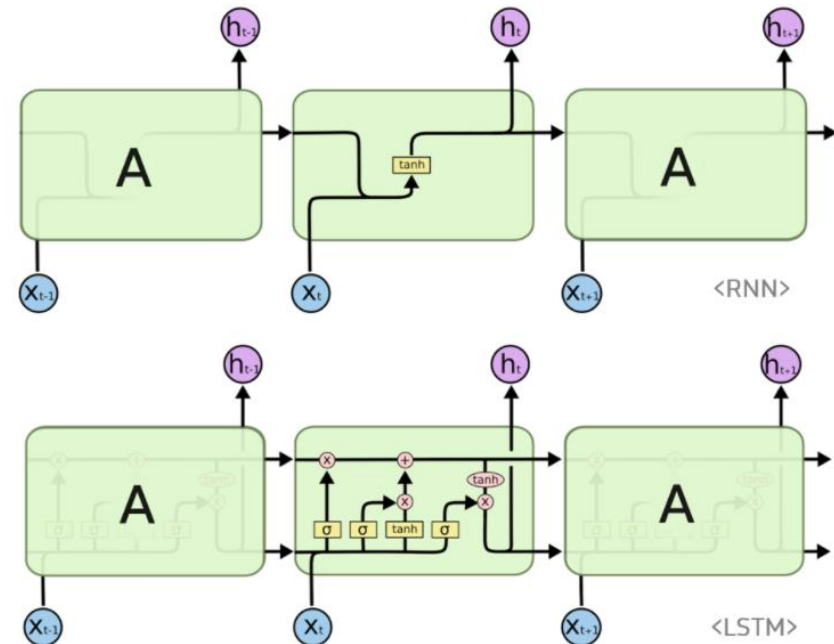
- 매니아: 똑같은/비슷한 음식을 계속/자주 먹는 사용자
- '모험가': 과거와 다른/새로운 음식을 먹는 사용자

<사용할 알고리즘>

LSTM(Long short term memory): RNN의 "vanishing gradient problem"을 극복하기 위해 고안된 알고리즘

RNN의 hidden state에 cell-state를 추가한 구조

- 사용자가 주로 단골집에서 자주 시키는지
 - 다양한 가게의 다양한 메뉴에 도전을 즐기는지
 - 과거의 데이터와 현재의 데이터에 가중치 차이를 두지 않는다.
-
- 사용자가 주문한 횟수, 즉 평균 주문 수를 계산 (전체)
 - 각 가게마다 가중치를 적용 (평균에서 얼마나 떨어져 있는지에 따라)
- ex) '매니아'
- 사용자가 A, B, C 가게에서 각각 10, 4, 1번 주문
- > 전체 평균 주문수 = 5번
- > 평균에서 얼마나 어떻게 떨어져 있는지 따라 분류 (분산)



-> A: +5 / B: -1 / C: -4 의 가중치를 둔다.

- ex) '모험가'

사용자가 A, B, C, D, E, F 가게에서 각각 1번씩 주문

-> 전체 평균 주문수 = 1번

-> 분산

-> A: 0 / B: 0 / C: 0 / D: 0 / E: 0 / F: 0

-> 가중치의 평균이 0에 가까움 = 모험가