

최종보고발표

음식 추천 AI 챗봇

10조 민정연 김강희 김혜원 배지원 이가은

INDEX

- 1 목적과 필요성
- 2 활용 데이터
- 3 챗봇 구조

INDEX

- 4 챗봇 구현
- 5 한계점 및 추후 과제
- 6 비즈니스 모델

1. 목적과 필요성

1. 목적과 필요성

다이어트 목적의 식단 추천



자취생을 위한 균형 잡힌 식단 추천

1. 목적과 필요성

질병 유무 + 신체조건



신체조건(키, 체중, 체지방률, 근육량)

1. 목적과 필요성

1인가구수 —

1인가구비율 33.4% '21

1인가구수 716만 5,788가구 '21



1인 가구 비율 매년 증가
= 혼자서 끼니를 해결하는 자취생 증가

자취생을 위한
균형 잡힌 한 끼 식단
추천 시스템에 대한 수요

2. 활용 데이터

2. 활용데이터

공공데이터포털 제공, ‘전국통합 식품영양성분정보 표준데이터’ 882개의 음식데이터 수집

식품코드	식품명	데이터구분코드	데이터구분명	식품기원코드	식품기원명	식품대분류코드
D101-002140000-0001	국밥_돼지머리	D	음식	1	가정식(분석 함량)	01
D101-002260000-0001	국밥_순대국밥	D	음식	1	가정식(분석 함량)	01
D101-002410000-0001	국밥_콩나물	D	음식	1	가정식(분석 함량)	01
D101-003000000-0001	기장밥	D	음식	1	가정식(분석 함량)	01
D101-004000000-0001	김밥	D	음식	1	가정식(분석 함량)	01
D101-004060000-0001	김밥_김치	D	음식	1	가정식(분석 함량)	01
D101-004080000-0001	김밥_날치알	D	음식	1	가정식(분석 함량)	01
D101-004110000-0001	김밥_돈가스	D	음식	1	가정식(분석 함량)	01
D101-004250000-0001	김밥_쇠고기	D	음식	1	가정식(분석 함량)	01
D101-004370000-0001	김밥_참치	D	음식	1	가정식(분석 함량)	01
D101-004390000-0001	김밥_채소	D	음식	1	가정식(분석 함량)	01
D101-004400000-0001	김밥_치즈	D	음식	1	가정식(분석 함량)	01
D101-004430000-0001	김밥_풋고추	D	음식	1	가정식(분석 함량)	01
D101-006070000-0001	덮밥_낙지	D	음식	1	가정식(분석 함량)	01
D101-006100000-0001	덮밥_닭고기	D	음식	1	가정식(분석 함량)	01
D101-006130000-0001	덮밥_돼지고기(제육)	D	음식	1	가정식(분석 함량)	01
D101-006190000-0001	덮밥_불고기	D	음식	1	가정식(분석 함량)	01
D101-006210000-0001	덮밥_우육면	D	음식	1	가정식(분석 함량)	01

2. 활용데이터

**식품대분류명
(ex. 밥류) 기준으로
식품군 분류**

식품대분류명	대표식품코드	대표식품명	식품중분류코드	식품중분류명	식품소분류코드	식품소분류명	식품세분류코드	식품세분류명
밥류	01002	국밥	0100214	돼지머리	010021400	해당없음	00	해당없음
밥류	01002	국밥	0100226	순대국밥	010022600	해당없음	00	해당없음
밥류	01002	국밥	0100241	콩나물	010024100	해당없음	00	해당없음
밥류	01003	기장밥	0100300	해당없음	010030000	해당없음	00	해당없음
밥류	01004	김밥	0100400	해당없음	010040000	해당없음	00	해당없음
밥류	01004	김밥	0100406	김치	010040600	해당없음	00	해당없음
밥류	01004	김밥	0100408	날치알	010040800	해당없음	00	해당없음
밥류	01004	김밥	0100411	돈가스	010041100	해당없음	00	해당없음
밥류	01004	김밥	0100425	쇠고기	010042500	해당없음	00	해당없음
밥류	01004	김밥	0100437	참치	010043700	해당없음	00	해당없음
밥류	01004	김밥	0100439	채소	010043900	해당없음	00	해당없음
밥류	01004	김밥	0100440	치즈	010044000	해당없음	00	해당없음
밥류	01004	김밥	0100443	풋고추	010044300	해당없음	00	해당없음
밥류	01006	덮밥	0100607	낙지	010060700	해당없음	00	해당없음
밥류	01006	덮밥	0100610	닭고기	010061000	해당없음	00	해당없음
밥류	01006	덮밥	0100613	돼지고기(제육)	010061300	해당없음	00	해당없음
밥류	01006	덮밥	0100619	불고기	010061900	해당없음	00	해당없음

2. 활용데이터

식품 별 100g당 칼로리와
필수 영양소 정보

식단 내 영양소
비율 조절에 활용

영양성분함량기준량	에너지(kcal)	수분(g)	단백질(g)	지방(g)	회분(g)	탄수화물(g)	당류(g)	식이섬유(g)	칼슘(mg)
100g	137	71.6	6.70	5.16	0.63	15.94	0.16	0.7	9
100g	75	83.7	3.17	2.28	0.48	10.38	0.17	1.3	9
100g	52	86.8	1.45	0.24	0.56	10.93	0.00	1.2	6
100g	166	59.1	3.44	0.57	0.15	36.77	0.00	1.5	4
100g	140	69.5	4.84	4.55	1.11	19.98	0.00	1.4	36
100g	130	71.2	4.30	4.03	1.29	19.17	0.29	1.8	44
100g	177	59.8	6.10	4.26	1.13	28.66	1.43	2.1	20
100g	202	55.8	5.77	5.81	0.95	31.64	1.33	2.1	15
100g	179	61.2	6.46	5.56	1.00	25.78	0.03	1.5	30
100g	174	64.2	7.00	7.22	1.32	20.26	0.71	1.8	51
100g	158	63.8	4.60	3.65	1.28	26.65	0.14	3.4	38
100g	177	63.3	6.24	7.03	1.33	22.10	0.21	2.6	6
100g	169	61.9	4.88	4.41	1.32	27.52	0.09	3.3	19
100g	150	65.9	5.88	3.34	0.80	24.12	2.16	2.3	15
100g	125	70.8	11.40	2.18	0.76	14.82	1.07	1.1	21
100g	202	62.0	9.43	10.77	0.95	16.86	1.98	1.4	11
100g	182	60.2	6.60	5.31	0.97	26.96	0.07	2.8	6

2. 활용데이터

	A	B	C	D	E	F
1	식품명	식품대분류	대표식품	영양성분	에너지(kcal)	단백질(g)
2	국밥_돼지머리국밥	밥류	국밥	100g	137	6.7
3	국밥_순대국밥	밥류	국밥	100g	75	3.17
4	국밥_콩나물국밥	밥류	국밥	100g	52	1.45
5	기장밥	밥류	기장밥	100g	166	3.44
6	김밥	밥류	김밥	100g	140	4.84
7	김밥_김치	밥류	김밥	100g	130	4.3
8	김밥_날치	밥류	김밥	100g	177	6.1
9	김밥_돈가시	밥류	김밥	100g	202	5.77
10	김밥_쇠고기	밥류	김밥	100g	179	6.46
11	김밥_참치	밥류	김밥	100g	174	7
12	김밥_채소	밥류	김밥	100g	158	4.6
13	김밥_치즈	밥류	김밥	100g	177	6.24
14	김밥_꽃고추	밥류	김밥	100g	169	4.88
15	덮밥_낙지	밥류	덮밥	100g	150	5.88
16	덮밥_닭고기	밥류	덮밥	100g	125	11.4
17	덮밥_돼지	밥류	덮밥	100g	202	9.43

[예시]
‘돼지머리 국밥’

식품대분류: 밥류
대표식품: 국밥
영양성분 및 에너지: 137Kcal/100g
단백질: 6.7g

2. 활용데이터

식품 성격 태깅: 5가지의 태그

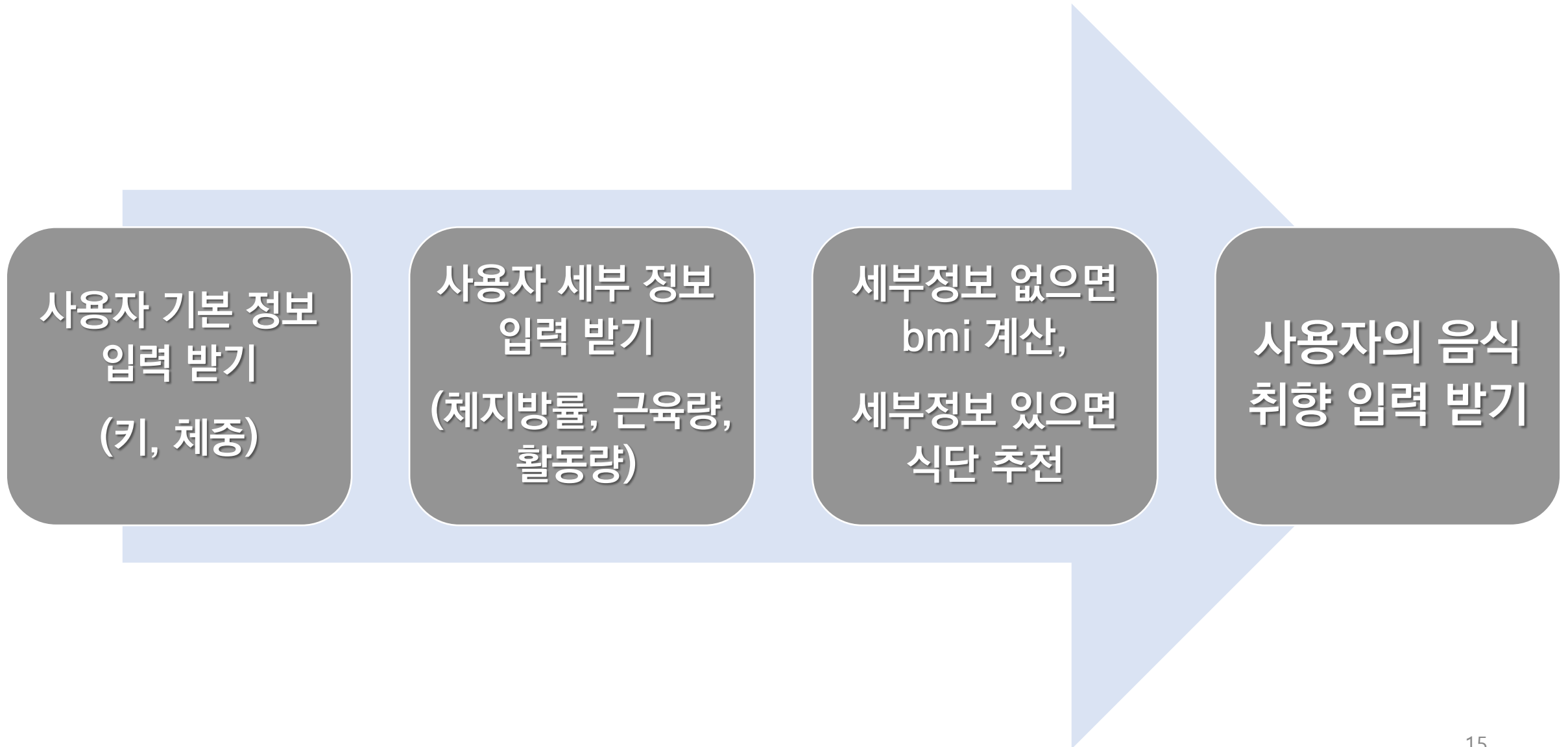
- 가벼움
- 매움
- 든든
- 기름진
- 빠리

한 음식 당 평균 1-2개의 태그
사용자 선호 반영 식단 설계에 활용

I	J	K
태그		
가벼움, 매움, 든든, 기름진, 빠리		
가벼움, 매움, 든든, 기름진, 빠리		
가벼움, 매움, 든든, 기름진, 빠리		
가벼움, 매움, 든든, 기름진, 빠리		
가벼움, 매움, 든든, 기름진, 빠리		
가벼움, 매움, 든든, 기름진, 빠리		
가벼움, 매움, 든든, 기름진, 빠리		
가벼움, 매움, 든든, 기름진, 빠리		
가벼움, 매움, 든든, 기름진, 빠리		
가벼움, 매움, 든든, 기름진, 빠리		
가벼움, 매움, 든든, 기름진, 빠리		
가벼움, 매움, 든든, 기름진, 빠리		
가벼움, 매움, 든든, 기름진, 빠리		
가벼움, 매움, 든든, 기름진, 빠리		
가벼움, 매움, 든든, 기름진, 빠리		
가벼움, 매움, 든든, 기름진, 빠리		

3. 챗봇 구조

3. 챗봇 구조



3. 챗봇 구조

입력 태그와 가깝고, 사용자의 건강 정보에 적합한 맞춤 식단 구성

식단 확인 후 마음에 들면 '확인', 마음에 안들면 '식단 변경'

'확인'된 식단은 선호도 점수 높게 측정하여 고빈도 출력

추후 음식 취향 입력으로 되돌아감

4. 챗봇 구현

4. 챗봇 구현

1. 패키지 설치

#패키지 설치

```
!pip install -q -U sentence-transformers
import pandas as pd
from sentence_transformers import SentenceTransformer
from sklearn.metrics.pairwise import cosine_similarity
model = SentenceTransformer('jhgan/ko-sroberta-multitask')
```

4. 챗봇 구현

2. 음식에 단 태그 벡터화

```
#파일 불러들이고 수작업한 '태그' 벡터값 추출
df = pd.read_csv('원본 수정 ver.csv', encoding='CP949')
df=df[~df['태그'].isna()]
df.loc[0, '태그']
model.encode(df.loc[0, '태그'])

#추출한 '태그' 벡터값 열 추가해서 새 csv 파일로 저장
df['vector']=pd.Series([[]]*len(df))
df['vector']=df['태그'].map(lambda x: list(model.encode(x)))
df.to_csv('벡터값 포함 데이터.csv', index=False)
df.head()
```

태그

짜짜

짜짜

짜짜

짜짜

빨리

vector

[0.047852244, -0.32964087, -0.10950571,
0.3504...

[0.047852244, -0.32964087, -0.10950571,
0.3504...

[0.047852244, -0.32964087, -0.10950571,
0.3504...

[0.047852244, -0.32964087, -0.10950571,
0.3504...

[-0.28290483, -0.04207116, -0.22384472,
-0.665...

4. 챗봇 구현

3. 사용자 정보 입력



#챗봇 첫 시작

```
print("안녕하세요, 음식 추천 AI 챗봇 한끼똑딱입니다.")
height=float(input("당신의 신장(cm)을 입력해주세요.> "))
weight=float(input("당신의 체중(kg)을 입력해주세요.> "))
fat = float(input("당신의 체지방률(%)을 입력해주세요.> (모르면 0을 입력해주세요) "))
muscle = float(input("당신의 근육량(kg)을 입력해주세요.> (모르면 0을 입력해주세요) "))
act = float(input("당신의 활동량은 얼마나 되나요?적으면 25, 보통은 30~35, 많으면 40을 입력해주세요"))
```

```
if fat == 0:
    BMI=round(weight/(height/100*height/100), 1)
    print("당신의 BMI는", BMI,"입니다.")
    if BMI < 18.5:
        print("저체중이므로 고단백질 위주의 식단을 추천드리겠습니다.")
    elif 18.5 <= BMI < 23:
        print("정상체중이므로 하루 평균 영양에 맞춰 식단을 추천드리겠습니다.")
    elif BMI >= 23:
        print("과체중이거나 비만이므로, 저탄수화물, 고단백질 위주의 식단을 추천드리겠습니다.")

else:
    if fat >= 32 and muscle > 39.2:
        print("저탄수화물 위주의 식단을 추천드리겠습니다.")
    if muscle <= 39.2 and fat < 32:
        print("고단백질 위주의 식단을 추천드리겠습니다.")
    if fat >= 32 and muscle <= 39.2:
        print("저탄수화물과 고단백질 위주의 식단을 추천드리겠습니다.")
    if fat < 32 and muscle > 39.2:
        print("정상 수치이므로 하루 평균 영양에 맞춰 식단을 추천드리겠습니다.")
```

안녕하세요, 음식 추천 AI 챗봇 한끼똑딱입니다.

당신의 신장(cm)을 입력해주세요.> 180

당신의 체중(kg)을 입력해주세요.> 100

당신의 체지방률(%)을 입력해주세요.> (모르면 0을 입력해주세요) 40

당신의 근육량(kg)을 입력해주세요.> (모르면 0을 입력해주세요) 20

당신의 활동량은 얼마나 되나요?적으면 25, 보통은 30~35, 많으면 40을 입력해주세요25
저탄수화물과 고단백질 위주의 식단을 추천드리겠습니다.

4. 챗봇 구현

4. 사용자 취향과 음식 태그 사이 유사도 측정

```
#사용자 대답 입력 받고 대답 벡터화
user_answer=input("원하는 종류의 음식을 입력하세요.(예시: 가벼운 게 먹고 싶어)>")
embedding=model.encode(user_answer)

#사용자 대답과 '태그'의 벡터값(vector 열) 사이 유사도 사이 거리 측정
df['similarity'] = df['vector'].map(lambda x: cosine_similarity([embedding], [x]).squeeze())
df.head()
```

원하는 종류의 음식을 입력하세요.(예시: 가벼운 게 먹고 싶어)>가벼운 게 먹고 싶어

	식품명	식품대분 류명	대표식품 명	영양성분함량기 준량	에너지 (kcal)	단백질 (g)	지방 (g)	탄수화물 (g)	태 그	vector	similarity
0	국밥_돼지 머리	밥류	국밥	100g	137	6.70	5.16	15.94	든 든	[0.047852244, -0.32964087, -0.10950571, 0.3504...	0.289628
1	국밥_순대 국밥	밥류	국밥	100g	75	3.17	2.28	10.38	든 든	[0.047852244, -0.32964087, -0.10950571, 0.3504...	0.289628
2	국밥_콩나 물	밥류	국밥	100g	52	1.45	0.24	10.93	든 든	[0.047852244, -0.32964087, -0.10950571, 0.3504...	0.289628
3	기장밥	밥류	기장밥	100g	166	3.44	0.57	36.77	든 든	[0.047852244, -0.32964087, -0.10950571, 0.3504...	0.289628
4	김밥	밥류	김밥	100g	140	4.84	4.55	19.98	빨 리	[-0.28290483, -0.04207116, -0.22384472, -0.665...	0.327036

4. 챗봇 구현

5. 식품분류 별로 밥, 국, 반찬1, 반찬2 데이터 프레임 생성

```
#밥 국 반찬 별로 데이터 프레임 생성
```

```
bab_dataframe=df.loc[df['식품대분류명']=='밥류', ['식품명', '단백질(g)', '지방(g)', '탄수화물(g)', 'similarity']]
```

```
guk_dataframe=df.loc[df['식품대분류명'].isin(['면 및 만두류', '죽 및 스프류', '국 및 탕류', '찌개 및 전골류']), ['식품명', '단백질(g)', '지방(g)', '탄수화물(g)', 'similarity']]
```

```
ban1_dataframe=df.loc[df['식품대분류명'].isin(['나물·숙채류', '생채·무침류', '김치류', '장아찌·절임류']), ['식품명', '단백질(g)', '지방(g)', '탄수화물(g)', 'similarity']]
```

```
ban2_dataframe=df.loc[df['식품대분류명'].isin(['찜류', '구이류', '전·적 및 부침류', '볶음류', '조림류', '튀김류', '수·조·어·육류']), ['식품명', '단백질(g)', '지방(g)', '탄수화물(g)', 'similarity']]
```

4. 챗봇 구현

6. 유사도 높은 음식 리스트 제공

```
#영양소 고려 안하고 유사도 가장 높은 음식 리스트 모두 도출
bab_M=max(bab_dataframe['similarity'])
bab_recommendation=bab_dataframe.loc[bab_dataframe['similarity']==bab_M]
guk_M=max(guk_dataframe['similarity'])
guk_recommendation=guk_dataframe.loc[guk_dataframe['similarity']==guk_M]
ban1_M=max(ban1_dataframe['similarity'])
ban1_recommendation=ban1_dataframe.loc[ban1_dataframe['similarity']==ban1_M]
ban2_M=max(ban2_dataframe['similarity'])
ban2_recommendation=ban2_dataframe.loc[ban2_dataframe['similarity']==ban2_M]
print(bab_recommendation)
print(guk_recommendation)
print(ban1_recommendation)
print(ban2_recommendation)
```

```
▶
443      식품명  단백질(g)  지방(g)  탄수화물(g)  similarity
445 초밥_(삼삼한밥상(VII))채소 초밥  2.71  0.49  31.30  0.61452
581      삼각김밥_고추장불고기  3.66  1.58  34.11  0.61452
582      삼각김밥_숯불갈비  4.00  1.67  32.55  0.61452
583      삼각김밥_참치마요네즈  3.94  3.20  31.95  0.61452
584      새우튀김롤  4.82  6.06  32.09  0.61452
585      연어를  6.19  5.13  24.77  0.61452
590      주먹밥_멸치  6.00  4.40  36.20  0.61452
```

```
      식품명  단백질(g)  지방(g)  탄수화물(g)  similarity
446      군만두_(눈치덕)군만두  8.89  17.78  24.44  0.685868
448 기타_(구브라고 숯불두마리 치킨)우동사리  3.71  1.26  40.00  0.685868
```

```
      식품명  단백질(g)  지방(g)  탄수화물(g)  similarity
340      가지나물무침  1.42  1.50  4.18  0.621923
341      고구마줄기나물무침  0.98  7.00  4.10  0.621923
342      고사리나물무침  3.78  5.52  6.28  0.621923
343      고춧잎나물무침  4.41  4.69  7.14  0.621923
344      갯잎나물무침  2.68  5.18  5.04  0.621923
...
872      마늘종장아찌  1.96  0.14  27.34  0.621923
873      마늘쭀장아찌_간장  2.00  1.00  10.80  0.621923
874      마늘쭀장아찌_고추장  2.60  1.80  21.00  0.621923
875      매실장아찌  1.65  0.43  58.11  0.621923
876      무장아찌  1.47  0.76  16.76  0.621923
```

[169 rows x 5 columns]

```
      식품명  단백질(g)  지방(g)  탄수화물(g)  similarity
877      대구포  31.97  0.40  2.33  0.621923
878      물회  4.91  2.70  7.90  0.621923
879      물회_생선  4.29  1.54  10.31  0.621923
```

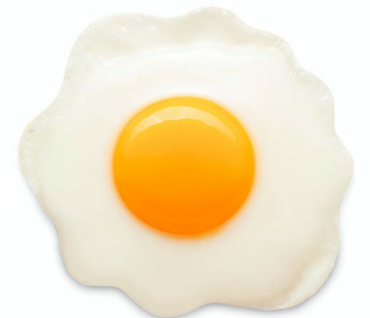
5. 한계점 및 추후 과제

5-1 한계점

1. 밥, 국, 반찬 2개의 조합이 복잡해서 영양소를 고려해 챗봇 구현을 완성하지 못했던 부분 존재



VS



5-1 한계점

2. 추천 메뉴가 한식에 제한되어 있음



<https://blog.naver.com/jiniher720/220882959933>



https://m.blog.naver.com/hansik_culture/221505645101



https://m.blog.naver.com/hansik_culture/221505645101

5-2 추후 과제

1. 메뉴간 조화 고려하기
2. 메뉴 추천의 정확도 높이기
3. 사용자가 직접 사용 가능한 채팅 형식의 UI 구현하기

6. 비즈니스 모델

6. 비즈니스 모델

1. CPM (Cost Per Mile)

- 노출된 건수만큼 기업에게 대가를 지불 받는 형태



<https://www.ottogi.co.kr/main/main.asp>



<https://play.google.com/store/apps/details?id=com.coupang.mobile.eats&hl=ko&gl=US>

6. 비즈니스 모델

2. CPC (Cost Per Click)

- 광고를 클릭한 횟수만큼 광고주로부터 돈을 지불 받는 방식



https://play.google.com/store/apps/details?id=kr.or.khealth.smhc&hl=en_US&gl=US

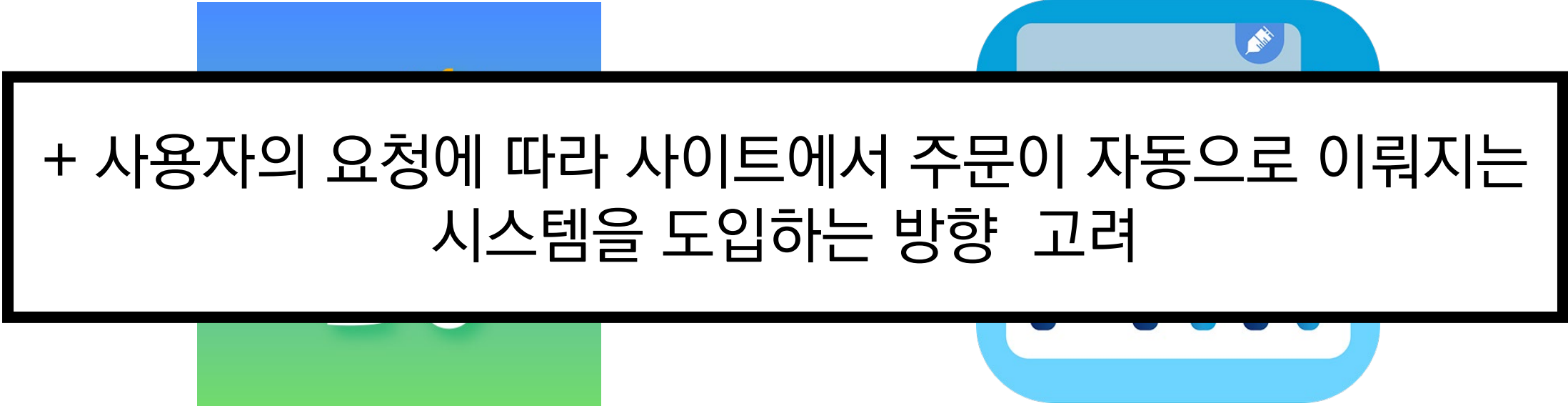


<https://play.google.com/store/apps/details?id=or.kr.khis.phr&hl=ko&gl=US>

6. 비즈니스 모델

2. CPC (Cost Per Click)

- 광고를 클릭한 횟수만큼 광고주로부터 돈을 지불 받는 방식



+ 사용자의 요청에 따라 사이트에서 주문이 자동으로 이뤄지는 시스템을 도입하는 방향 고려

https://play.google.com/store/apps/details?id=kr.or.khealth.smhc&hl=en_US&gl=US

<https://play.google.com/store/apps/details?id=or.kr.khis.phr&hl=ko&gl=US>

감사합니다
