

# Youtube 의 Video Meta-Data 를 이용한 맛집 추천 서비스



저자 1 201621149 김민호

저자 2 201724613 하평안

저자 3 201824611 한정훈

지도교수 이도훈

---

# 목 차

1. 서론 .....	1
1.1 연구 배경 .....	1
1.2 기존 문제점 .....	1
1.3 연구 목표 .....	1
2. 연구 내용 .....	2
2.1 시스템 구상 및 설계 .....	3
2.2 상세 기술 .....	6
2.3 기술 설계 .....	8
2.4 기존 서비스 분석 및 차별화 .....	20
3. 연구 결과 분석 및 평가 .....	22
3.1 중간평가 .....	22
3.2 개선사항 .....	22
3.3 결과 분석 및 평가 .....	23
4. 결론 및 향후 연구 방향 .....	29
4.1 결론 .....	29
4.2 향후 연구 방향 .....	30
5. 개발 일정 및 역할 분담 .....	31
5.1 개발 일정 .....	31
5.2 역할 분담 .....	32
6. 참고 문헌 .....	32

## 1. 서론

### 1.1 연구 배경

코로나 19의 영향으로 오프라인 활동의 제한과 동시에 온라인 활동이 활발해졌다. 이와 동시에 온라인 개인 방송 플랫폼의 확산으로 온라인 동영상 플랫폼인 Youtube는 [그림 1]에서와 같이 '한국에서 가장 많이 접속한 웹사이트'로 성장했다. 사용자들은 이전과 달리 더 많은 시간을 온라인 비디오 콘텐츠 시청에 할애하고 있으며, 그중에서 다양한 주제의 먹거리와 맛집 리뷰 동영상이 많이 시청 및 생산되고 있다. 이러한 동영상은 음식의 맛, 가격, 위치, 분위기, 서비스 품질 등 소비자에게 유용한 다양한 정보를 포함하고 있다. 따라서, Youtube 동영상들의 Meta Data를 활용하면 사용자에게 더욱 최적화된 맛집 추천 서비스를 제공할 수 있을 것으로 판단하여 이와 같은 주제를 선정했다.

올해 4월 한국에서 가장 많이 접속한 웹사이트 순위 (PC+모바일)		
순위	웹사이트	누적방문자 수
1위	유튜브	3720억명
2위	구글	667억8800만명
3위	네이버	421억3700만명
4위	디자인사이드	234억1100만명
5위	나무위키	177억6900만명

자료: 셀러쉬(SEMrush)

그림 1 - 웹사이트 방문자 수

### 1.2 기존 문제점

기존의 맛집 추천 서비스는 주로 사용자의 검색어나 위치 정보 기반의 추천 서비스를 제공하고 있다. 이로 인해 추천이 상대적으로 일반화되어 사용자의 실제 취향 및 기호에 부합하지 않은 맛집이 추천된다. 또한, 기존의 서비스는 주로 식당의 정보와 해당 서비스 이용자들이 남긴 평가 점수를 기준으로 맛집을 선별한다. 그렇기에 광고, 가짜 리뷰, 정확하지 않은 정보로 인해 맛집 추천 서비스를 온전히 신뢰할 수 없는 문제가 발생한다.

### 1.3 연구 목표

이 연구의 목표는 YouTube Video Meta Data를 활용한 맛집 추천 서비스를 개발하여 기존의 맛집 추천 서비스와 차별화된 서비스를 사용자에게 제공하는 것이다. 서비스가 제공하는 기능은 다음과 같다.

- 
- 광범위한 검색 기능 : 사용자가 어떤 내용을 검색하더라도 검색 내용과 가장 유사한 맛집을 찾아서 추천한다.
  - 개인 맞춤형 기능 : YouTube 동영상의 이미지 및 관련 텍스트 정보를 분석하여 사용자에게 더욱 최적화된 추천 서비스와 상세 정보를 제공한다.

이러한 연구 목표를 달성함으로써, 사용자에게 더욱 최적화된 맛집 추천 서비스를 제공한다.

## 2. 연구 내용

### 2.1 시스템 구상 및 설계

#### 2.1.1 시스템 설계도

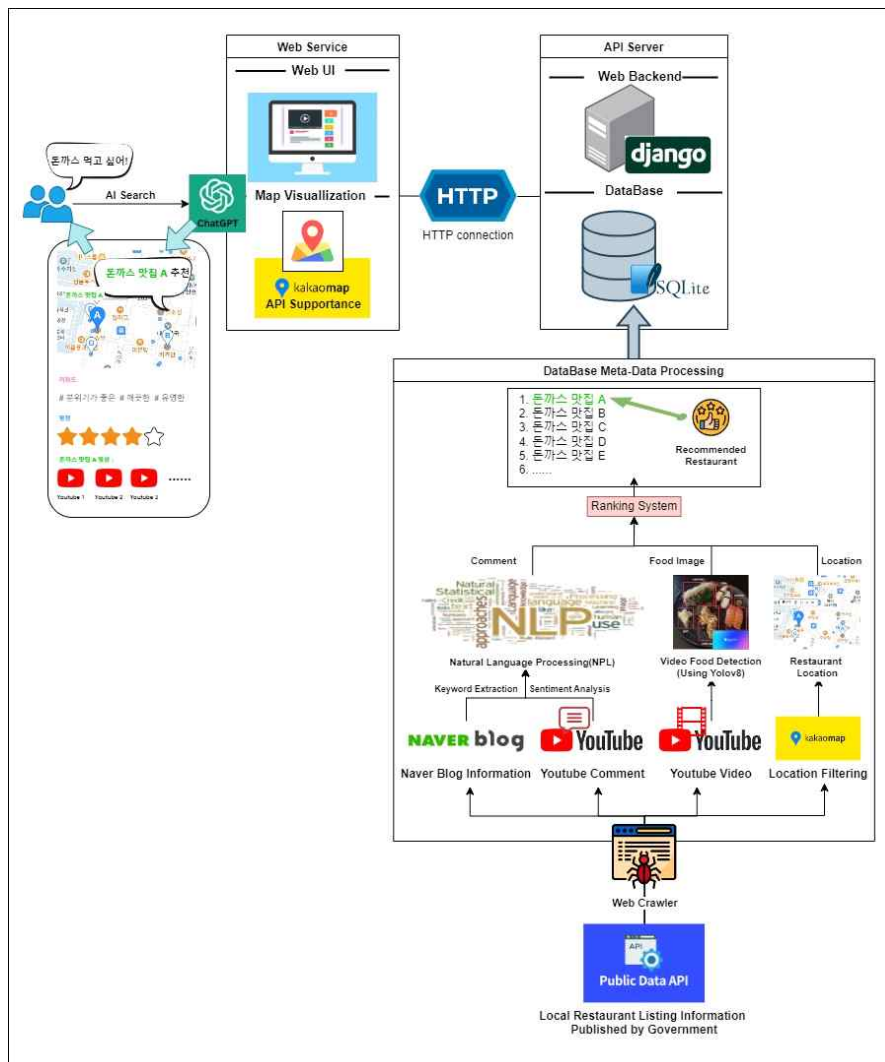


그림 2 - 시스템 설계도

#### 2.1.2 시스템 상세 설계

해당 서비스는 사용자에게 Youtube Video Meta-Data를 기반으로 맛집을 추천한다. 사용자에게 가장 알맞은 맛집을 추천하기 위해 다양한 방법을 사용하고 있는데, 방법으로

- 1) 사용자의 요구 분석
- 2) Youtube Video Meta-Data 추출 및 활용
- 3) NLP(자연어 처리) 활용

가 있다.

---

### 1) 사용자 요구 분석

먼저 사용자의 요구를 파악하는 것이 가장 우선이다. 사용자가 정확하게 원하는 식당 혹은 메뉴가 존재한다면 문제가 없지만, 예외의 경우 음식과 식당의 종류가 너무 다양해서 추천에 어려움 발생한다.

이러한 문제점을 해결할 수 있는 다양한 방안이 있는데, 가장 좋은 방법은 사용자의 취향과 이전 방문기록과 같은 프로필 데이터를 이용하는 것이다. 그러나 모든 사용자의 프로필 데이터를 구할 수 없다. 따라서 대안으로 자연어 처리를 활용하는 방안을 선택했다.

자연어 처리는 다음과 같은 방식으로 활용했다. 사람들이 식당을 선택하는 모습을 보면 키워드로 식당을 선택하는 모습을 쉽게 볼 수 있다. 예를 들어, 날이 너무 추워서 “따뜻한 음식”을 선택하거나, 데이트 코스로 “분위기가 좋은” 식당을 가는 것처럼 말이다. 이처럼 사용자의 요구에서 “따뜻한 음식”, “분위기가 좋은” 과 같은 키워드를 추출했다. ‘자연어 처리 - 키워드 추출’ 기술을 이용해 맛집과 관련된 키워드를 담은 키워드 사전을 제작하였고, Chat GPT API로 사용자의 요구 사항과 키워드 사전을 비교하여 적합한 키워드를 추출했다. 그리고 해당 키워드를 포함하고 있는 식당 목록을 검색하여 사용자에게 추천할 수 있도록 했다.

### 2) Youtube Video Meta-Data 추출 및 전처리

Youtube Video에는 이미지, 음성, 텍스트 등 수 많은 데이터가 존재한다. 타 서비스의 맛집과 관련된 데이터양은 보면 텍스트, 이미지, 영상 순으로 존재한다. 특히 영상과 관련된 데이터는 기타 데이터와 비교해 현저하게 적기 때문에 맛집 추천 알고리즘에 Youtube Video에 존재하는 영상 데이터를 사용할 수 있다면 타 서비스와 차별된 경쟁력을 가질 수 있을 것이다.

맛집 관련 데이터를 얻기 위해서 먼저 맛집과 관련된 영상을 찾고, 해당 영상에서 필요한 데이터를 추출하는 과정이 필요하다. Youtube에는 API로 가져올 수 있는 데이터와 그렇지 않은 Raw-Data가 모두 존재한다. 먼저 Youtube API를 통해 얻을 수 있는 데이터는 [동영상 제목, 동영상 설명, 태그, 조회 수, 좋아요, 싫어요, 업로드 일자, 댓글 및 댓글 수, 채널 관련 정보 등]이 있다. 이 데이터만으로 맛집 관련 영상을 판별과 필요한 데이터 추출까지 모두 가능하다면 좋겠지만 문제가 있다. 주어진 데이터 중 대부분이 영상 제작자가 직접 입력하는 데이터이기 때문에, 영상 제작자가 영상에 대

---

한 충분한 설명과 데이터를 입력하지 않았거나 잘못된 데이터를 입력했을 경우 해당 영상에서 필요한 데이터를 얻을 수 없게 된다.

이러한 문제를 해결하기 위해 YOLOv8을 활용해 맛집 관련 영상을 판별하고 '자연어 처리-감성 분석' 기술로 추가적인 데이터를 추출했다. 자세한 과정으로는, 먼저 Youtube API를 이용해 맛집 이름을 검색했을 때 나오는 영상 목록과 영상의 정보들을 수집했다. 영상 목록은 Youtube에서 설정한 '관련성'을 기준으로 정렬되었기 때문에 종종 맛집과 관련 없는 영상이 섞여 있다. 2번의 필터링 과정을 통해 맛집 관련 영상을 판별했는데, 먼저 '영상 제목'과 검색어를 비교했다. 다음으로 영상에서 YOLOv8로 Object Detection을 수행하여 해당 영상에서 맛집과 관련된 데이터가 얼마나 많이 노출되는지를 확인하여 판별을 진행했다.

위의 과정을 거쳐 맛집 관련 영상을 판별한 후, '조회 수', '좋아요', '댓글' 데이터를 가공하여 필요한 데이터를 추출했다. 마지막으로 모든 데이터를 종합하여 맛집 순위를 매긴 후 사용자에게 맛집을 추천한다.

### 3) NLP(자연어 처리) 활용

맛집과 관련된 데이터는 YouTube 이외의 다른 플랫폼에서도 풍부하게 제공된다. 특히 Naver, Kakao, Google Maps와 같은 서비스에서 리뷰, 별점, 블로그 게시물과 같은 다양한 데이터를 얻을 수 있다. YouTube 데이터만으로 맛집을 추천하기에는 한계가 있다. 따라서 다른 플랫폼에 있는 데이터를 크롤링과 NLP 기술을 통해 추가로 수집 및 가공하려 한다. NLP 기술 중 '키워드 추출'을 블로그 데이터에, '감성 분석' 기술을 Youtube 댓글에 활용하여 데이터를 추출한다.

먼저, '키워드 추출'은 크롤링한 블로그 데이터에서 중요한 키워드를 식별하고 추출하는 기술이다. 형태소 분석을 통해 의미를 포함하고 있는 단어를 식별하고, 반복적으로 사용되는 단어를 추출하는 과정을 통해 키워드 추출이 이루어진다. 이때, 맛집과 관련된 블로그 데이터만을 대상으로 키워드 추출을 진행하여, 맛집과 관련된 키워드만 추출한다. 마지막으로 추출한 키워드를 모아서 키워드 사전을 만든다. 이후 맛집에 대한 블로그 데이터와 키워드 사전을 비교해 해당 맛집에 대한 키워드를 추출할 수 있다.

다음으로 '감성 분석'은 주어진 텍스트에 대해 해당 텍스트가 포함하고 있는 감성을 분석하는 기술이다. 텍스트의 긍정 혹은 부정을 판단하는 것부터

---

놀람, 기쁨, 슬픔과 같은 감정을 분석할 수도 있다. 이 기술을 YouTube 댓글 데이터에 활용하여 맛집 및 맛집 관련 영상에 대한 감성을 파악할 수 있다.

위의 두 가지 기술을 이용해 맛집과 관련된 추가 데이터를 추출하여 사용자에게 더욱 유용한 데이터를 제공할 수 있도록 한다.

위에서 소개한

- 1) 사용자 요구 분석
  - 2) Youtube Video Meta-Data 추출 및 전처리
  - 3) NLP(자연어처리) 활용
- 3가지 방법을 통해 사용자에게 가장 적합하고 어울리는 맛집을 추천하는 서비스를 제공한다.

### 2.1.3 시스템 시나리오

- User Interface

- 1) 맛집 검색과 관련된 내용을 입력한다.
- 2) GhatGPT API로 입력된 내용에서 핵심 키워드를 추출한다.
- 3) 키워드와 관련된 맛집을 평점을 기준으로 추천한다.
- 4) 링크를 통해 해당 맛집에 대한 세부정보 및 관련 Youtube Video를 제공한다.

- Create DataBase

- 1) 일반음식점 공공데이터를 Kakao Data와 비교하여 필터링한 식당 목록을 얻는다.
- 2) NLP와 Object Detection으로 식당에 대한 데이터 획득 및 평가를 수행한다.
- 3) 추출한 Meta-Data로 카테고리별 추천 맛집을 선정한다.

## 2.2 상세 기술

### 2.2.1 REST API

REST는 URI와 HTTP Method를 활용해 자원과 행위를 표현하는 아키텍처의 한 형식이다. URI를 통해 자원을 명시하고, HTTP Method(POST, GET, PUT, DELETE, PATCH)로 자원에 대한 CRUD Operation을 수행한다. REST API는 REST 방식을 따라 설계된 API를 의미한다.

### 2.2.2 Django

Django는 Python으로 작성된 웹 프레임워크 중 하나로 MVT(Model -



---

View - Template) 패턴을 따른다. Django는 데이터베이스 관리, URL 라우팅, 사용자 인증, 템플릿 엔진, REST API 작성을 위한 풍부한 기능 등을 제공하여 개발자들의 웹 애플리케이션 개발 및 유지·보수를 돕는 역할을 한다.

### 2.2.3 SQLite

SQLite는 DBMS(Database Management System) 중 하나로 프로그램에 포함되어 사용하는 비교적 가벼운 DBMS이다. 별도의 서버가 존재하지 않기 때문에 대규모 작업에는 적합하지 않지만, 작은 규모의 데이터베이스를 효율적으로 관리하는 데 유용하다.

### 2.2.4 YOLOv8

YOLO (You Only Look Once)은 실시간 객체 탐지를 위한 딥 러닝 알고리즘이다. 객체 탐지를 위한 다양한 알고리즘이 존재하지만, YOLO는 기존 모델보다 빠른 속도와 상대적으로 높은 정확도를 보여준다.

YOLO의 동작 원리는 아래와 같다.

1. 먼저 입력한 이미지를 확인한다.
2. 이미지를  $N \times N$ 의 그리드로 나눈다.
3. 각 그리드에 대해 이미지 분류 및 지역화 작업을 수행한다.
4. 객체가 어디에 있는지 확인하고, 식별해야 하는 객체에 Bbox를 그린다.
5. bBox와 각 개체의 클래스 확률을 통해 객체를 인식하고 예측합니다.

### 2.2.5 NLP(Natural Language Processing)

NLP(자연어 처리, Natural Language Processing)는 일상에서 사용하는 자연어를 기계가 이해, 해석, 생성하는 AI 기술 중 하나이다.

NLP로 수행할 수 있는 작업 중에는 감성 분석과 키워드 추출이 있다.

- 감성 분석 (Sentiment Analysis):

감성 분석은 텍스트의 감성 또는 의견을 파악하는 기술이다. 주로 텍스트의 긍정, 부정, 중립을 파악하여 제품 및 서비스의 피드백을 위한 용도로 사용된다.

- 키워드 추출 (Keyword Extraction):

텍스트에서 중요한 단어나 구절을 식별하여 해당 문서 또는 문서 집합의 핵심 주제를 파악하는 기술이다. 대량의 텍스트 데이터에서 카테고리 할당 및 특정 데이터 검색에 유용하게 사용된다.



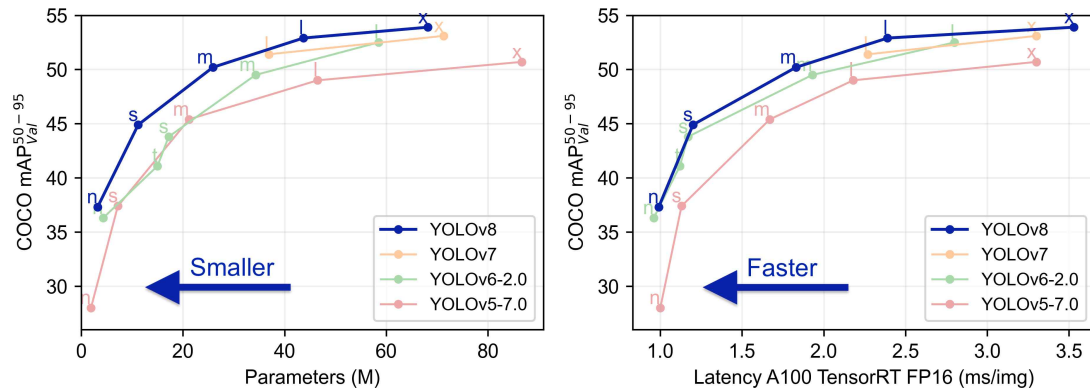


그림 4 - 버전 비교 그래프

서비스 제작을 위해 이전 버전과 비교해서 더 높은 성능과 더 간편한 사용법을 가지고 있는 YOLOv8를 채택하여 사용했다.

Object Detection을 수행하기 위해서는 학습된 모델이 필요하다. 기본적으로 YOLOv8에서 제공하는 모델은 그림 5에서 볼 수 있는 것처럼 5가지가 존재한다.

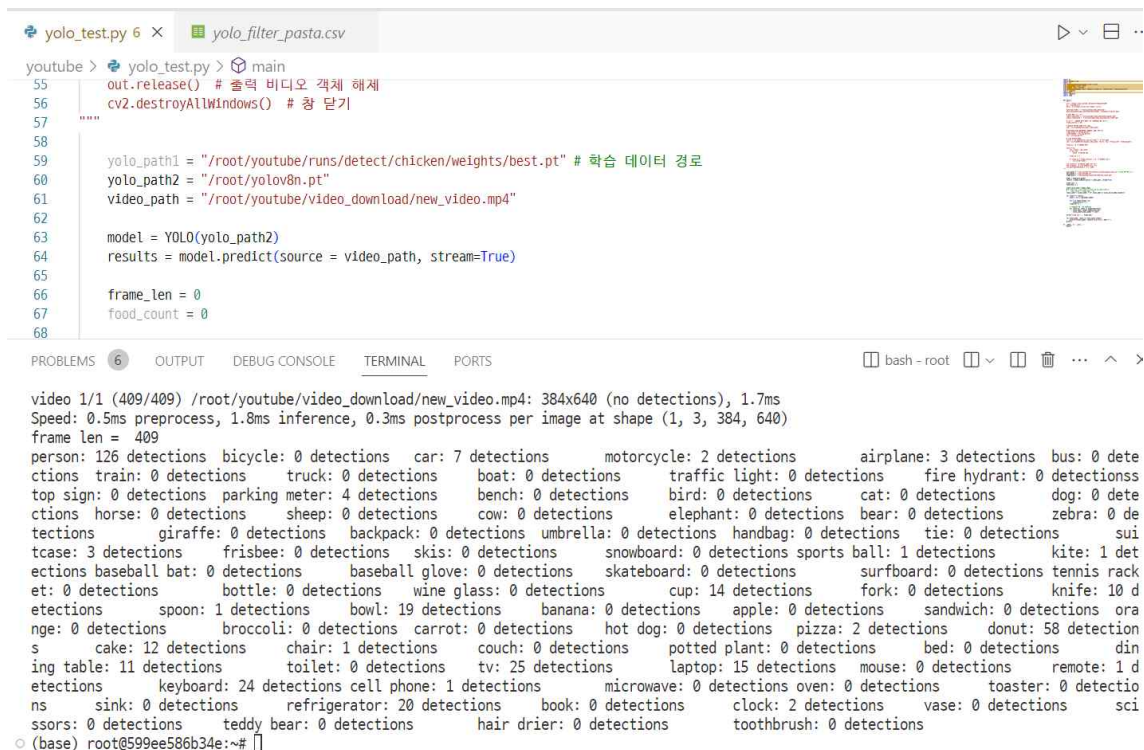
Model	size (pixels)	mAP <sup>val</sup> 50-95	Speed CPU ONNX (ms)	Speed A100 TensorRT (ms)	params (M)	FLOPs (B)
<a href="#">YOLOv8n</a>	640	37.3	80.4	0.99	3.2	8.7
<a href="#">YOLOv8s</a>	640	44.9	128.4	1.20	11.2	28.6
<a href="#">YOLOv8m</a>	640	50.2	234.7	1.83	25.9	78.9
<a href="#">YOLOv8l</a>	640	52.9	375.2	2.39	43.7	165.2
<a href="#">YOLOv8x</a>	640	53.9	479.1	3.53	68.2	257.8

그림 5 - YOLOv8 모델 비교

#### - 학습 모델 테스트

먼저 맛집 관련 영상에 대해 Object Detection이 얼마나 정확하게 수행되

는지 확인하기 위해 테스트를 진행했다. 학습 모델은 데이터 크기를 기준으로 n부터 x까지 5개가 존재하는데, 테스트에는 YOLOv8n 모델을 사용했다. 그림 6은 Youtube 영상(<https://www.youtube.com/watch?v=4wouxIMru90>)을 YOLOv8n 모델로 detection을 수행한 결과이다. 영상은 치킨과 관련된 영상으로 10프레임마다 1회씩 Detection을 수행했으며, 결과는 "class 이름 : detection 횟수" 형식으로 출력했다.



```

yolo_test.py 6 x yolo_filter_pasta.csv
youtube > yolo_test.py > main
55 out.release() # 출력 비디오 객체 해제
56 cv2.destroyAllWindows() # 창 닫기
57
58
59 yolo_path1 = "/root/youtube/runs/detect/chicken/weights/best.pt" # 학습 데이터 경로
60 yolo_path2 = "/root/yolov8n.pt"
61 video_path = "/root/youtube/video_download/new_video.mp4"
62
63 model = YOLO(yolo_path2)
64 results = model.predict(source = video_path, stream=True)
65
66 frame_len = 0
67 food_count = 0
68
PROBLEMS 6 OUTPUT DEBUG CONSOLE TERMINAL PORTS
video 1/1 (409/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
Speed: 0.5ms preprocess, 1.8ms inference, 0.3ms postprocess per image at shape (1, 3, 384, 640)
frame len = 409
person: 126 detections bicycle: 0 detections car: 7 detections motorcycle: 2 detections airplane: 3 detections bus: 0 detections
train: 0 detections truck: 0 detections boat: 0 detections traffic light: 0 detections fire hydrant: 0 detections
top sign: 0 detections parking meter: 4 detections bench: 0 detections bird: 0 detections cat: 0 detections dog: 0 detections
horse: 0 detections sheep: 0 detections cow: 0 detections elephant: 0 detections bear: 0 detections zebra: 0 detections
giraffe: 0 detections backpack: 0 detections umbrella: 0 detections handbag: 0 detections tie: 0 detections suitcase: 3 detections
frisbee: 0 detections skis: 0 detections snowboard: 0 detections sports ball: 1 detections kite: 1 detections
baseball bat: 0 detections baseball glove: 0 detections skateboard: 0 detections surfboard: 0 detections tennis racket: 0 detections
bottle: 0 detections wine glass: 0 detections cup: 14 detections fork: 0 detections knife: 10 detections
spoon: 1 detections bowl: 19 detections banana: 0 detections apple: 0 detections sandwich: 0 detections orange: 0 detections
broccoli: 0 detections carrot: 0 detections hot dog: 0 detections pizza: 2 detections donut: 58 detections
cake: 12 detections chair: 1 detections couch: 0 detections potted plant: 0 detections bed: 0 detections dining table: 11 detections
toilet: 0 detections tv: 25 detections laptop: 15 detections mouse: 0 detections remote: 1 detections
keyboard: 24 detections cell phone: 1 detections microwave: 0 detections oven: 0 detections toaster: 0 detections
sink: 0 detections refrigerator: 20 detections book: 0 detections clock: 2 detections vase: 0 detections scissors: 0 detections
teddy bear: 0 detections hair drier: 0 detections toothbrush: 0 detections
(base) root@599ee586b34e:~#

```

그림 6 - YOLOv8n

그림 6의 결과에서 YOLOv8n 모델의 class를 보면 맛집 혹은 음식과 관련된 class가 거의 없는 것을 볼 수 있다. 따라서 탐색 결과를 사용할 수 없을 것으로 판단했고, 이후에 진행되는 테스트에서는 맛집 관련 학습 모델을 사용했다.

```
PROBLEMS 7 OUTPUT DEBUG CONSOLE TERMINAL PORTS
bash - root

video 1/1 (400/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (401/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.6ms
video 1/1 (402/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.6ms
video 1/1 (403/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (404/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (405/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.6ms
video 1/1 (406/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (407/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.6ms
video 1/1 (408/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (409/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
Speed: 0.5ms preprocess, 1.7ms inference, 0.2ms postprocess per image at shape (1, 3, 384, 640)
frame len = 409
Braised lotus roots: 0 detections Dongchimi: 0 detections Japchae: 0 detections Kimchi stew: 0 detections Kimchi: 0
detections Korean rib: 0 detections Korean style raw beef: 0 detections Seasoned bean sprouts: 0 detections Ud
on: 0 detections baek Kimchi: 0 detections banquet noodles: 0 detections bean sprout soup: 0 detections beef-bone soup
: 0 detections bellflower greens: 0 detections bibimbap: 0 detections boiled fish paste soup: 0 detections budaejjigae: 0
detections bulgogi: 0 detections chonggak Kimchi: 0 detections cucumber kimchi: 0 detections cup rice: 1 detections dri
ed pollack soup: 0 detections eel: 0 detections fried rice: 0 detections green onion Kimchi: 0 detections
grilled mackerel: 0 detections janjorim: 0 detections jeyuk bokkeum: 0 detections memil soba: 0 detections
miso soup: 0 detections mixed rice: 0 detections nabak Kimchi: 0 detections naengmyeon: 0 detections pickled s
esame leaf: 0 detections pig hocks: 0 detections pork belly: 0 detections radish Kimchi: 0 detections radish kimchi
: 0 detections rice ball: 1 detections rice roll: 1 detections rice: 1 detections seasoned bean sprouts: 0 detections
seasoned bellflower root: 0 detections seasoned bracken: 0 detections seasoned zucchini: 0 detections seaweed soup: 0 de
tections seaweed: 0 detections shellfish soup: 0 detections spicy yuke jang: 0 detections spinach greens: 0 detection
s stewed mackerel: 0 detections stir-fried anchovies: 0 detections young radish Kimchi: 0 detections
(base) root@599ee586b34e:~#
```

그림 7 - Korean Food

그림 7은 Roboflow에서 가져온 Korea Food 학습 모델을 사용하여 탐색을  
진행한 결과이다. 한식과 관련된 dataset이기 때문에 YOLOv8n 모델보다 높  
은 정확도를 보여줄 것으로 기대했다.

테스트 영상은 그림 6과 같은 영상을 사용했다. 결과를 보면 YOLOv8n과  
Korean Food 모두 영상의 전체 프레임 수는 409이며, 기대와는 다르게  
Korean Food 학습 모델에 치킨과 관련된 class가 없어서 여전히 필요한 데  
이터를 추출하지 못하는 모습을 볼 수 있다.

```
PROBLEMS 6 OUTPUT DEBUG CONSOLE TERMINAL PORTS
bash - root

video 1/1 (395/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (396/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (397/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (398/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (399/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (400/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (401/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (402/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (403/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (404/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (405/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (406/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (407/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (408/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
video 1/1 (409/409) /root/youtube/video_download/new_video.mp4: 384x640 (no detections), 1.7ms
Speed: 0.5ms preprocess, 1.8ms inference, 0.2ms postprocess per image at shape (1, 3, 384, 640)
frame len = 409
fried-chicken: 25 detections
(base) root@599ee586b34e:~#
```

그림 8 - Chicken

---

마지막은 그림 8은 치킨과 관련된 학습 모델을 사용하여 탐색을 수행한 결과이다. 학습 모델은 역시 Roboflow에서 치킨과 관련된 학습 모델을 가져와서 사용했다.

결과를 보면 이전 학습 모델들과는 다르게 치킨에 대한 탐색이 이루어진 것을 확인할 수 있다. 전체 409프레임에 대해 25번의 탐색이 이루어졌으며, 실제 영상을 보면 2분 16초 중에서 약 30초 정도 치킨이 노출된다. 비율을 계산해보면 탐색이 진행된 프레임 비율은  $25/409 = 6\%$ 이고, 실제 노출 비율은 22%로 탐색과 실제 사이에 간격이 존재한다. 그러나 실제 영상에서 탐색이 어려운 장면이 있는 것을 고려했을 때, 완전한 탐색이 아닌 해당 영상이 맛집 관련 영상인지 판단하는 수준의 탐색은 가능하다고 볼 수 있다.

위의 결과를 통해, 영상의 관련성 판단을 위해서는 해당 영상과 관련된 학습 모델이 필요함을 볼 수 있다. 따라서 음식 항목으로 맛집을 구분하여 각각 다른 학습 모델을 통해 탐색을 진행했다. 학습 모델의 유무와 영상 데이터의 유무를 고려하여, 우선 “파스타, 피자, 햄버거, 치킨, 초밥” 총 5가지 항목에 대해 작업을 수행했다.

위의 작업을 위해 리스트에 있는 식당 중에서 해당 5가지 항목에 포함되는 식당을 분류했다. 식당 분류는 일반음식점 공공데이터에서 제공하는 데이터를 이용했다. 일반음식점 공공데이터는 아래 그림 9와 같은 형식으로 사업장과 항목 등 다양한 데이터를 제공한다. 그러나 항목을 보면 사업자가 개업 시 구청에 신고한 데이터가 저장되어 있어 일관성이 없는 것을 확인할 수 있다. 따라서 새로운 카테고리를 만들어 그림 10과 같이 식당 목록을 수정했다. 이후 위에서 언급한 5가지 항목에 대한 식당을 항목별로 재추출하여 그림 11과 같이 분류했다.



이름	항목	3	4
자갈치브라더스	호프,요리주점	부산 북구 상하	<a href="https://place.m">https://place.m</a>
미소김밥	분식	부산 북구 만덕	<a href="https://place.m">https://place.m</a>
푸주옥	설렁탕	부산 금정구 금	<a href="https://place.m">https://place.m</a>
도스커피	카페	부산 금정구 정	<a href="https://place.m">https://place.m</a>
잉클밥	양식	부산 금정구 금	<a href="https://place.m">https://place.m</a>
오븐에꾸운닭 서	치킨	부산 금정구 서	<a href="https://place.m">https://place.m</a>
홍매사천요리	중식	부산 금정구 수	<a href="https://place.m">https://place.m</a>

그림 9 - 일반음식점 공공데이터 원본

사업장명	최종수정시점	데이터갱신구분	데이터갱신일자	업태구분명
자갈치브라더스	2021-07-15 15:	U	2021-07-17 02:	분식
미소김밥	2022-12-01 13:	U	2022-12-03 02:	한식
푸주옥 설렁탕 도	2018-01-25 17:	I	2018-08-31 23:	한식
도스커피(does)	2018-08-29 15:	I	2018-08-31 23:	경양식
잉클밥	2019-08-27 13:	U	2019-08-29 02:	한식
대일반점	2014-01-10 11:	I	2018-08-31 23:	중국식
오꾸닭	2021-11-09 11:	U	2021-11-11 02:	한식
금수네 밥집	2017-09-18 14:	I	2018-08-31 23:	경양식
홍매사천요리	2019-09-05 12:	U	2019-09-07 02:	중국식
코코치노	2018-12-12 13:	U	2018-12-14 02:	한식

그림 10 - 일반음식점 공공데이터 수정본

<ul style="list-style-type: none"> <li>result_csv</li> <li>&gt; filter_csv</li> <li>original_csv</li> <li>burger.csv</li> <li>chicken.csv</li> <li>pasta.csv</li> <li>pizza.csv</li> <li>sushi.csv</li> <li>&gt; yolo_csv</li> </ul>	<ol style="list-style-type: none"> <li>토츠, "토츠, <a href="https://www.youtube.com/watch?v=HRyE815am4c">https://www.youtube.com/watch?v=HRyE815am4c</a>, 725,16,12,1880,746,0"</li> <li>잉클밥, "잉클밥, <a href="https://www.youtube.com/watch?v=0Wwez-ytt08">https://www.youtube.com/watch?v=0Wwez-ytt08</a>, 342,8,0,3218,1092,0", "잉클밥</li> <li>파머쉬(PHARMACY)</li> <li>제멜리아</li> <li>델라고 부산대점</li> <li>에픽키친, "에픽키친, <a href="https://www.youtube.com/watch?v=sEZF4dogobV">https://www.youtube.com/watch?v=sEZF4dogobV</a>, 10085,102,21,910,641,0",</li> <li>베드빌, "베드빌, <a href="https://www.youtube.com/watch?v=kMf7Ie6ZPb8">https://www.youtube.com/watch?v=kMf7Ie6ZPb8</a>, 164,1,0,355,227,0"</li> <li>이음</li> <li>초원, "초원, <a href="https://www.youtube.com/watch?v=pgREAc_q0aM">https://www.youtube.com/watch?v=pgREAc_q0aM</a>, 4736,137,9,2259,48,0"</li> <li>수제왕대박돈까스</li> </ol>
--	---

그림 11 - 일반음식점 공공데이터 재배포치

다음은 위와 같이 분류작업이 끝난 식당들을 기준으로 관련 영상을 추출할 차례이다. Youtube API를 이용해 식당 이름을 Youtube 검색했을 때 나오는 식당 목록과 관련 데이터를 얻었다. 그리고 Object Detection을 수행하기 전, Pytube를 이용하여 Youtube 영상을 오프라인에 저장 후 작업을 수행했다. Url을 이용해서 YOLOv8을 바로 영상에 적용할 수도 있지만, 성능을 봤을 때 모든 프레임에 대해 Object Detection을 수행하는 것이 비효율적이며 또한 url을 이용해 접근하는 경우 제한된 영상과 같은 다양한 문제가 발생했다. 따라서 저장이 가능한 영상에 대해 오프라인으로 영상을 저장하고 필요에 맞게 프레임 수를 조정하여 Object Detection을 수행했다.

표 1은 항목별 Object Detection 수행 결과이다. 학습 모델에 따라 항목별로 탐색 결과에 큰 차이가 나타났다.

항목	Youtube 추천 영상	YOLO-판단 영상	실제 관련 영상
파스타	57	33	32
햄버거	2	1	1
치킨	36	2	2
피자	0	0	0
초밥	70+	0	0

표 1 - 영상 판단 결과

표의 결과에서 순서대로 Youtube에서 추천해준 영상의 수, YOLO를 이용해서 Object Detection 후 관련 영상으로 판단된 수, 그중 실제 관련 영상 수를 나타낸 것이다.

1) 파스타는 높은 정확도를 보여주고 있다.

2) 햄버거와 피자의 경우 Youtube에 있는 영상 데이터가 거의 없어서 결과를 판단하기 어렵다. 해당 항목에 대해 영상 데이터가 거의 없는 것은 프랜차이즈 식당이 아닌 개인 식당을 우선으로 작업을 진행했는데, 금정구 인근에 피자과 햄버거를 판매하는 개인 식당이 거의 없어서 위와 같은 문제가



---

발생했다.

3) 치킨과 초밥은 Youtube에서 제공한 영상 목록 중에서 관련 영상으로 판단된 영상이 거의 없는 것을 볼 수 있다. 먼저 치킨의 경우 프랜차이즈를 제외한 개인 식당 중 대부분 단순히 튀긴 치킨이 아닌 닭과 관련된 다른 메뉴를 판매하고 있는 것을 확인했다. 예를 들어, 닭볶음탕, 닭구이 등과 같은 메뉴를 판매하고 있어서 튀긴 치킨을 학습한 모델의 Object Detection에 대해 안 좋은 결과를 보인다. 마지막으로 초밥 역시 학습 모델에 있는 초밥의 모양과 실제 식당에서 나오는 초밥의 모양의 차이로 인해 제대로 된 결과가 나오지 않는 것을 확인했다.

결과적으로 제대로 된 학습 모델만 있으면, 파스타 항목과 같이 제대로 된 영상 판별이 가능하다는 것을 확인할 수 있었다.

### 2.3.2 NLP - Sentiment Analysis

NLP(자연어 처리)에는 다양한 기술이 존재한다. 그중에서 감성 분석과 키워드 추출을 활용하여 맛집 추천 서비스를 제작했다.

먼저 감성 분석(Sentiment Analysis)에 관한 내용이다. 감성분석은 텍스트에 나타난 사람들의 태도, 의견, 성향과 같은 주관적인 데이터를 분석하는 자연어 처리 기술이다. 대표적인 예시로는 IMDb(Internet Movie Database)를 이용한 영상물 리뷰 및 분석이다. IMDb는 영상물에 대한 다양한 정보를 제공하는 dataset인데, 해당 dataset을 이용해 영상 리뷰를 분석하고 별점을 매겨 순위를 제공하는 서비스이다. 그림 12는 IMDb를 이용해 TV 영상물의 평점 및 순위를 보여준다

[ 편집 ]

[ 편집 ]

9.

### 그림 12 - IMDb Rating

IMDb의 예시와 같이 감성 분석을 이용하면 해당 영상에 대한 시청자들의 반응을 분석할 수 있다. 이 기술을 Youtube 댓글에 활용하여 영상과 관련 식당에 대한 사용자들의 반응을 긍정 또는 부정으로 분석했다.

Youtube 댓글 감성 분석을 위한 학습 데이터로는 KNU 감정 사전을 이용했다. 댓글 내용에서 형태소를 분석하여 KNU 감정 사전과 대조하여 일치하는 단어를 추출했다. 그리고 추출된 단어 중에서 긍정어와 부정어의 비율을 통해 댓글의 호감도를 계산하였다. 호감도는 0에서 100까지로 나타나며 숫자가 클수록 호감도가 높은 영상이다.

```
positive_counts: 136
기대하는, 기대, '맛있게', '성공', '맛있게', '잘', '맛있다', '함께', '잘', '재미있게', '잘', '좋아하는', '귀엽고', '중
기대하는, '잘', '맛있게', '귀엽고', '기대', '먼저', '잘', '함께', '사람', '잘', '좋은', '잘', '좋은', '덕', '잘', '만족
있게, '잘', '맛있게', '귀엽고', '행복', '맛있게', '잘', '잘', '좋아하는', '좋은', '잘', '좋은', '만족
'잘', '최고', '성공', '만고', '행복하다', '맛있게', '하루하루', '만', '행복한', '맛있게', '건강한', '그리운', '맛있게'
'잘', '최고', '잘', '최고', '좋은', '감사', '맛있게', '최고', '맛있게', '만하게', '즐거고', '기대', '잘', '최고', '맛있게'
사랑, '좋은', '잘', '좋은', '잘', '잘', '최고', '좋아하는', '즐거하고', '잘', '사랑', '함께', '맛있게', '행복', '맛있게', '사랑', '맛있게', '사랑
'좋은', '잘', '중요한', '위트', '맛있게', '잘', '잘', '사랑', '열심히', '유명한', '유명한'
negative_counts: 15
['모시', '모시', '힘들어', '부담', '포기', '모시', '걱정', '걱정', '모시', '없다', '해', '글욕', '해', '도둑', '지지']
rate : 90.1 %
```

그림 13 - 입짧은햇님 이재모피자 리뷰 영상(90.1%)

```
shalomha@shapyeong-an-ui-MacBookAir emotion % ./usr/bin/python3 /Users/shalomha/Desktop/악과/출업/emotion/comments_positive.py
positive counts: 183
['잘', '조화', '최상급', '멋진', '최고', '멋진', '인정', '멋진', '인정', '멋진', '인심', '맛있는', '맛있는', '잘', '최고', '맛들', '감들', '유익한', '잘',
'조화', '최상급', '인심', '부드럽고', '필수', '맛있는', '인정하게', '맛있', '칭찬', '맛있는', '부자', '맛있는', '흥이', '유익한', '멋진',
'멋진', '잘', '감들', '최고', '멋진', '맛있게', '멋진', '칭찬', '예정', '최고', '유익한', '유명한', '맛있게', '좋은', '맛있게', '~~~~~',
'잘', '맛', '최고', '가치', '최고', '맛있는', '인기', '잘', '맛들', '잘', '멋진', '잘', '맛있다', '맛들', '대가', '좋다', '좋다', '기대',
'잘', '맛', '극락', '맛있게', '감추', '진리', '잘', '좋아하는', '인정', '잘', '멋진', '최고', '잘', '보람', '최고', '맛있는', '오츠', '맛',
'맛있는', '맑음', '멋진', '신뢰', '잘', '맑음', '오츠', '유명한']
negative counts: 37
['은', '은', '잘못', '우는', '아쉬움', '부럽다', '죽음', '외부러', '우는', '벌하는', '집', '실망', '죽음', '느끼하다', '우는', '심한', '지옥', '후회',
'은', '도둑', '무섭다', '변태', '까비', '우는', '오츠', '우는', '포기', '모르는', '실망', '우는', '미친', '실망', '도둑', '최악', '바보', '죽', '견방지',
'etc']
cat: 73.6 %
```

그림 14 - 정육왕 톤쇼우 리뷰 영상(73.6%)



용해 맛집을 검색하는데 사용할 수 있도록 했다.

다음은 키워드 추출의 과정이다. 먼저, konlpy 자연어 처리를 통해 블로그 리뷰의 형태소를 분석하여 가장 높은 빈도수를 보이는 키워드 10개를 추출했다.

```
표본 댓글 /랑 이식당
scan vocabs ...
num vocabs = 1172
done = 10
[('너무', 1), ('맛있', 1), ('정말', 1), ('만두', 1), ('음식', 1), ('진짜', 1), ('먹고', 1), ('제주', 1), ('밀퐁유', 1), ('천절', 1)]
```

그림 16 - 1차 키워드 추출 결과

그러나 추출한 키워드를 보면 맛집과 관련된 키워드가 거의 없는 것을 확인할 수 있었다. 또 추출한 키워드를 이용해서 맛집을 분류하기에는 키워드가 너무 일반적인 키워드라는 문제점이 있었다.

이러한 문제점을 해결하기 위해 키워드 사전을 제작했다. 맛집과 관련된 키워드 사전을 제작해 블로그 리뷰와 대조하여 키워드를 추출하면 훨씬 높은 정확도를 보일 것으로 예상했다.

키워드 사전을 제작하기 위해 많은 리뷰 데이터를 보유하고 있으며, 서로 다른 카테고리의 식당 20개를 선정했다. 이후 모든 식당에 대해 블로그 리뷰 30개와 평점 리뷰 500개를 크롤링해 키워드를 수집했다.

```
done = 10
['너무', '맛있', '칼국수', '장칼', '한번에', '해이팅', '음식', '강릉', '가다', '주문', '맛은', '테이블', '국물', '맛이', '먹었', '그냥', '대기', '말이', '들어', '먹고', '만두', '소속', '밖에서', '진짜', '사람', '김치', '맛집', '나오', '손님', '정말', '조금', '다른', '별로', '있는', '그리고', '처음', '오래', '같이', '바로', '먹는', '한끼', '근데', '양도', '먹을', '30분', '줄서서', '먼저', '다시', '양이', '1시간', '저는', '얼른', '감아오', '진한', '말은', '약간', '40분', '아니', '유명한', '저희', '생각', '출아', '짜고', '하나', '여기', '모든', '안에', '뉴', '없이', '시간', '개인으로', '맛도', '면은', '그래도', '공기', '해서', '그런지', '먹어', '면이', '정도', '칼칼', '해장', '방문', '평일', '면도', '매운', '아주', '좋았', '추가', '20분', '알아서', '먹기', '하고', '갔는데', '시스템', '나왔', '나름', '제일', '늦게', '입맛', '절대', '느낌', '나와서', '계속', '먹으', '솔직히', '매뉴', '같은', '미리', '리뷰', '한시간', '하는', '끌어서', '나왔', '나름', '가게', '순서', '중국산', '불친절', '엄청', '고추장', '이상', '면을', '있으니', '받아', '줄', '깃', '사장님', '국수', '매장', '전체', '시키', '직원', '주차', '있었', '짜여요', '받고', '실망', '보통', '없어', '한참', '시원', '현대', '받아', '자리', '남자', '잘끔', '없음', '아침', '다음', '기본', '이런', '원래', '없고', '두번', '시케', '자극적', '어떤', '잘먹', '입장', '지나', '가고', '이유', '완전', '고기', '매워', '땀기', '합니다', '나온', '것도', '열치욕', '없는', '했어요', '모르겠', '이렇게', '굳이', '열테이블', '다신', '구구', '가면', '앉아', '서서', '회전', '아쉬', '먹은', '기대', '방식이', '주말', '있었', '나고', '재방문', '강은', '추천', '맛집', '매콤', '기분', '짜요', '말은', '4', '곳이', '게더운', '보드', '하저', '핵인', '부족', '같이', '그럼', '여행', '친절하', '12', '마지막', '아닌', '말아', '있어', '배추', '왔다', '먹은', '최악', '처우', '근처', '10분', '적어', '드시', '가게', '1', '들어', '정신', '서버', '11', '반반', '불편', '식당', '했다', '일바', '힘들', '친구', '갈습', '양되', '주방', '식사', '서빙', '편찮', '놀러', '면발', '가능', '중심', '참고', '멜로이', '안중', '감자', '감니', '했는', '일장', '형제', '했지만', '재료', '예전', '특별', '분들', '있고', '썩썩', '9천원', '실었', '전하', '후추', '돈독', '운영', '구수', '조리', '호박', '느껴', '아쉽', '아무', '하시', '덜매', '3.', '방송', '립립', '말씀', '장국', '호불호', '다르', '오정어']
```

그림 17 - 키워드 사전

다음은 추출한 키워드를 카테고리별로 구분했다. 우선 음식(메뉴), 맛, 분위기, 서비스, 가격 카테고리로 키워드를 구분하고 중복되는 키워드 및 관련 없는 키워드를 제거하여 카테고리별 키워드 사전을 제작했다.

다음은 제작한 키워드 사전의 일부이다.

```
menu = ['돈까스', '피자', '치킨', '파스타', '흑돼지', '버거', '수제', '삼겹살', '국물',  
        '초밥', '국밥', '튀김', '마라탕']  
taste = ['시원', '시원한', '신선한', '따뜻', '건강', '풍미', '진한', '부드러운', '이국', '고  
        소', '배부']  
atmosphere = ['편한', '고급']  
service = ['친절', '정성']  
price = ['가성비', '혜자']
```

이렇게 만들어진 키워드 사전을 이용해 다시 키워드 추출을 진행했다. 맛집 블로그 리뷰와 제작한 키워드 사전을 대조하여 맛집에 대한 키워드를 추출했다. 아래는 그림은 키워드 추출 결과이다.

```
shalomha@hapyeong-an-ui-MacBookAir emotion % /usr/bin/python3 /Users/s  
scan vocabs ...  
num vocabs = 2234  
done = 10  
scan vocabs ...  
num vocabs = 2830  
done = 10  
alls : ['수제', '진한', '깔끔', '시원', '풍미', '부드러운', '친절']
```

그림 18 - 만게츠(라멘집)

```
num vocabs = 1699  
done = 10 Early stopped.  
alls : ['피자', '파스타', '수제', '고소한', '배부', '친절', '가성비']
```

그림 19 - 이재모 피자

```
num vocabs = 225  
done = 10  
alls : ['초밥', '수제', '튀김', '국물', '고소한', '시원', '배부', '풍미', '고급', '친절', '정성', '가성비']
```

그림 20 - 스시심

```

scan vocabs ...
num vocabs = 101
done = 10
alls : ['우동', '국물', '치킨', '따뜻한']

```

그림 21 - 겐로쿠 우동

위와 같은 방식으로 모든 식당에 대한 키워드 추출을 진행했다.

## 2.4 기존 서비스 분석 및 차별화

### 2.4.1 기존 서비스 분석

이미 맛집 추천 서비스가 많이 존재하기 때문에 기존 서비스와 차별화된 서비스를 설계하는 것은 필수적이다. 차별화된 서비스를 설계하기에 앞서 기존 서비스의 문제점과 개선점을 파악하는 것이 가장 우선이라 판단했다. 아래 표 2는 기존 서비스들의 특징을 분석하고 장점 및 개선해야 할 점을 작성한 내용이다.

서비스명	특징
망고플레이트	<ul style="list-style-type: none"> <li>- 가장 기본적인 맛집 추천 서비스</li> <li>- 식당에 대한 다양한 정보 제공</li> <li>- 사용자들의 평가를 데이터베이스로 보유</li> </ul>
다이닝코드	<ul style="list-style-type: none"> <li>- 네이버, 다음 포털의 블로그 내용 기반</li> <li>- 빅데이터를 기반으로 맛집 추천</li> <li>- 광고성 내용이 평가에 영향을 미치지 않음</li> </ul>
식신	<ul style="list-style-type: none"> <li>- 다양한 매체와 협업하여 맛집 추천 서비스 제공</li> <li>- 실제 방문한 식당에 대한 평가만 제공</li> <li>- 원격 예약 서비스 제공</li> </ul>
배달의 민족	<ul style="list-style-type: none"> <li>- 배달 및 포장 업체 대상 서비스</li> <li>- 리뷰 데이터가 많이 존재</li> <li>- 광고, 이벤트로 인한 신뢰할 수 없는 데이터</li> </ul>
종합	<ul style="list-style-type: none"> <li>- 다양한 플랫폼에 존재하는 빅데이터 기반 추천 서비스</li> <li>- 실제 방문객 대상 리뷰 데이터 수집 필요 (광고 및 이벤트 리뷰 필터링 작업 수행)</li> <li>- 이미지 및 영상과 같은 시각적 데이터 필요</li> <li>- 식당에 대한 분석 뿐 아니라 사용자의 취향과 개성 고려</li> <li>- 서비스에 대한 접근성과 편의성의 향상</li> </ul>

표 2 - 기존 서비스 분석

---

#### 2.4.2 차별화된 서비스 설계

기존 서비스 분석한 내용을 기반으로 차별화된 서비스를 설계했다. 가장 우선은 서비스의 제목과 같이 Youtube의 Video Meta-Data를 활용한 것이다. 기존 서비스들의 데이터는 자체 서비스, 네이버, 카카오, 구글과 같은 플랫폼에 있는 텍스트 및 이미지 데이터를 기반으로 제작되었다. 그러나 서론에서 언급한 것과 같이 계속해서 영상 데이터에 대한 수요와 공급이 늘어나고 있는 상황에서 기존의 데이터에 영상 데이터를 더하여 맛집 추천 서비스를 제공하면 분명히 차별화된 서비스가 될 것이다.

다음은 개인 맞춤형 서비스이다. 맛집 추천 서비스에 존재하는 딜레마는 사용자에게 대한 충분한 데이터가 없는 것과 서비스에 대한 접근성이 떨어진다는 것이다. 자세하게 설명하자면, 네이버, 구글과 같은 대형 플랫폼에서도 서비스를 제공하고 있기에 사용자들은 접근성이 떨어지는 맛집 추천 서비스를 잘 이용하지 않는다. 이로 인해 맛집 추천 서비스는 사용자에게 대한 자세하고 충분한 데이터를 쌓을 수 없게 되고, 이로 인해 사용자에게 알맞은 식당 추천이 이루어지지 않는다. 그리고 사용자는 자신에게 알맞은 식당 추천이 이루어지지 않는 서비스를 이용하지 않을 것이고 계속해서 악순환이 반복된다. 이러한 문제점을 해결하기 위해 Chat GPT Open AI를 활용하여 사용자의 요구에 유연하게 대응할 수 있도록 서비스를 구현했다. 사용자가 어떠한 내용을 입력하는지와 상관없이 적합한 식당을 추천하는 것이다.

위와 같은 방법으로 기존의 서비스와 차별화되고, 사용자가 유용하게 사용할 수 있는 맛집 추천 서비스를 제작했다.

---

### 3. 연구 결과 분석 및 평가

#### 3.1 중간 평가

##### 3.1.1 자체 평가

Youtube 영상을 분석하여 사용자에게 영상 내에 존재하는 정보를 제공하는 것이 초기 목표였지만, 맛집과 관련된 모든 영상을 완벽히 분석하기에는 기술적인 한계가 있다. 또 위치, 카테고리를 기반으로 식당을 추천해주는 알고리즘은 다소 평이하고 타 플랫폼 서비스와 유사하여 차별화된 맛집 추천 알고리즘이 필요하다.

##### 3.1.2 멘토 평가

멘토님께서 언급하신 내용의 핵심은 차별점과 신뢰성 2가지이다.

1. 서비스가 제공하는 추천 맛집 알고리즘이 다른 유사 서비스와 어떻게 다른지?

멘토 의견 : 다양한 플랫폼에 존재하는 리뷰 및 별점 데이터 중 신뢰할 수 있는 데이터를 맛집 추천 알고리즘에 활용.

2. 서비스가 제공하는 데이터를 사용자가 얼마나 신뢰할 수 있는지?

멘토 의견 : NLP 기법으로 Youtube 댓글의 긍정과 부정 비율을 예측할 수 있다. 이를 활용해 다른 서비스의 별점과 비교하는 등의 방식을 통해 평가 신뢰도 향상.

#### 3.2 개선사항

기존에는 Youtube Video Meta-Data만을 활용해서 맛집 추천 서비스를 제작하려고 했다. 그러나 신뢰할 수 있는 양질의 데이터를 제공하기 위해 타 플랫폼에 있는 데이터를 크롤링하여 사용했다.

크롤링 데이터의 활용 방안으로는,

1. 키워드 추출

블로그와 방문자 리뷰 데이터를 이용해 맛집과 관련된 키워드를 추출했다.

2. 감성 분석



Youtube 댓글 데이터를 이용해 평가 알고리즘의 신뢰도를 높였다.

### 3. Open AI API 검색

장소, 카테고리 검색 기반 서비스와 차별화된 검색 서비스를 제공한다.

### 3.3 결과 분석 및 평가

다음은 맛집 추천 서비스의 결과물이다.

#### 3.3.1 초기화면

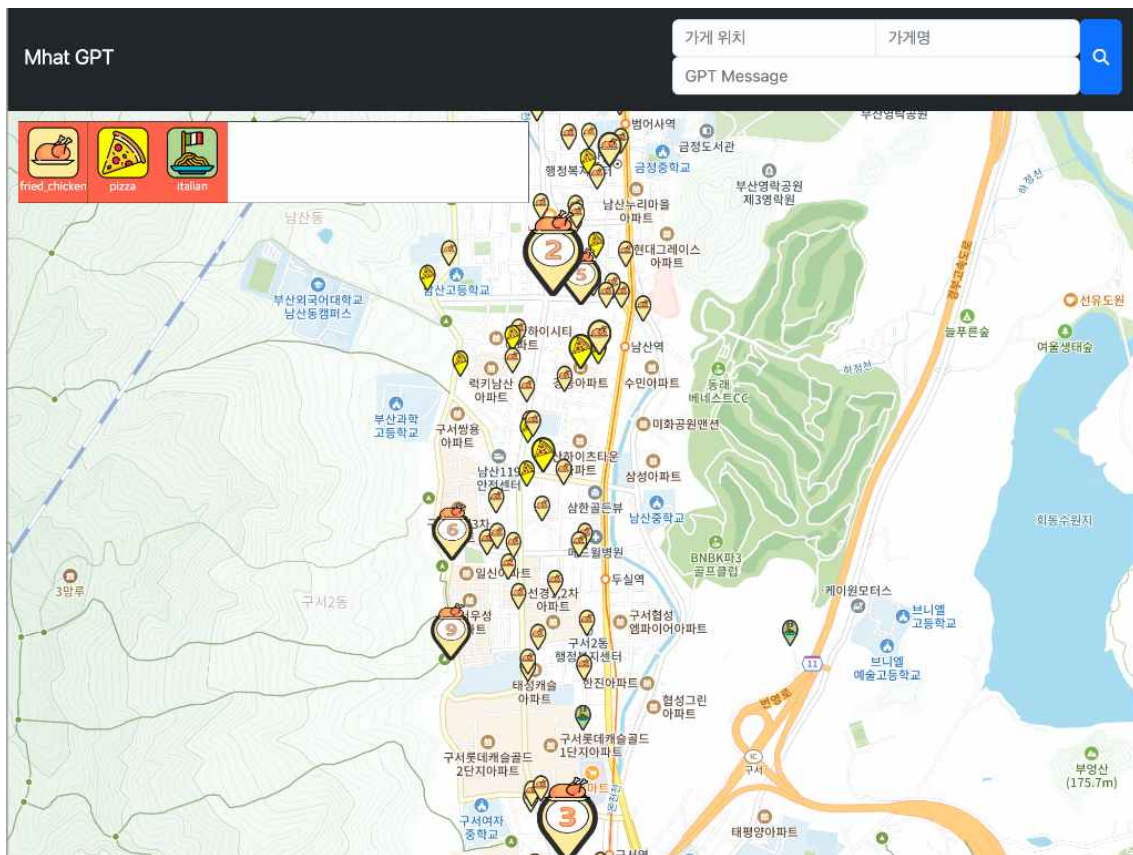


그림 22 - 초기화면

- 화면의 좌측 상단에는 카테고리를 선택할 수 있는 버튼이 있다.
- 화면 우측 상단에서 장소, 가게명, AI 검색이 가능하다.
- 지도 화면에 추천 맛집의 위치와 카테고리가 핀을 통해 제공된다.
  - ↳ 핀의 크기와 숫자는 맛집 추천 순위를 나타낸다.
  - ↳ 가장 초기화면은 모든 식당에 대한 추천 순위를 나타낸다.

### 3.3.2 위치 기반 검색 (부산대학교 정문)

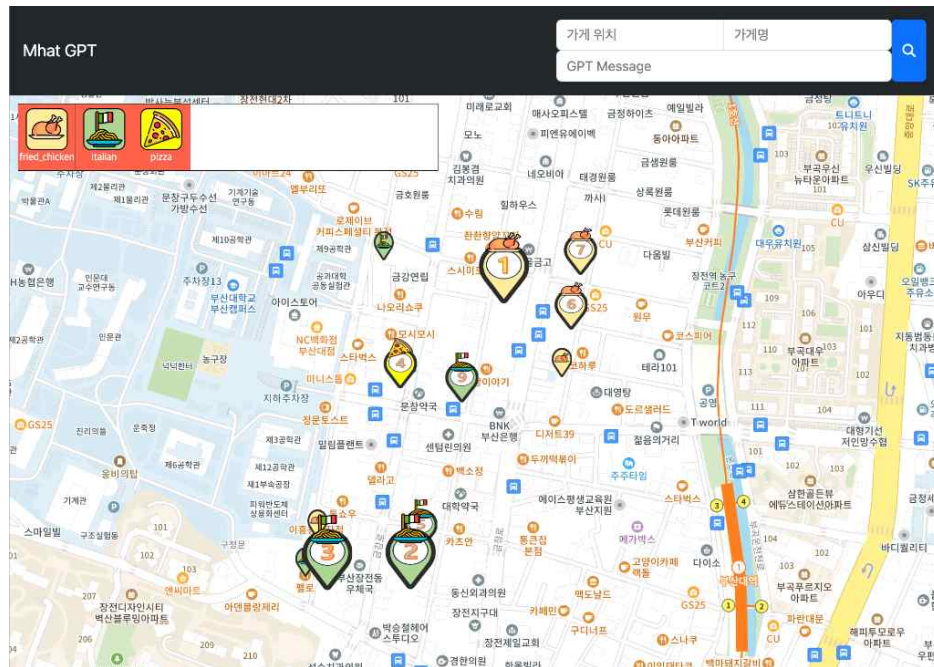


그림 23 - 위치 기반 검색

- 위치를 기반으로 검색한 화면이다
  - ↳ 그림 23은 부산대학교 정문을 기준으로 검색한 결과이다.
- 왼쪽 상단 카테고리 목록으로 추천 맛집의 카테고리를 확인할 수 있다.
- 초기화면과 마찬가지로 추천 맛집 순위가 핀을 통해 제공된다.

### 3.3.3 식당 상세정보 - 미리보기

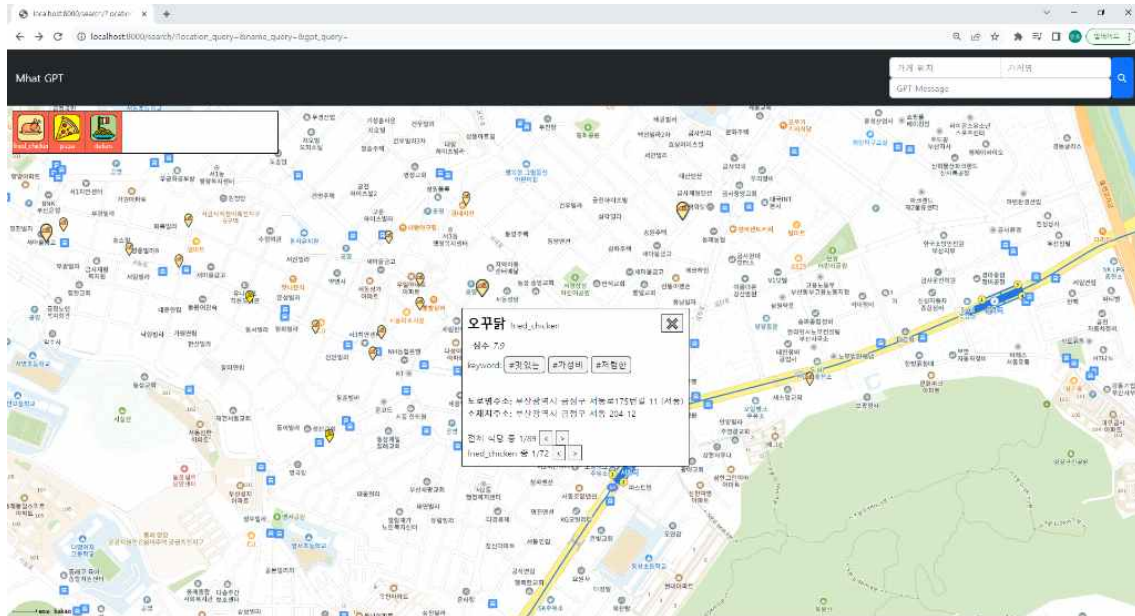


그림 24 - 상세정보 미리보기

- 맛집의 핀 위에 마우스를 올리면 해당 맛집에 대한 상세정보가 간략하게 제공된다.
- 관련 키워드, 위치, 맛집 추천 순위 정보가 제공된다.

### 3.3.4 식당 상세정보

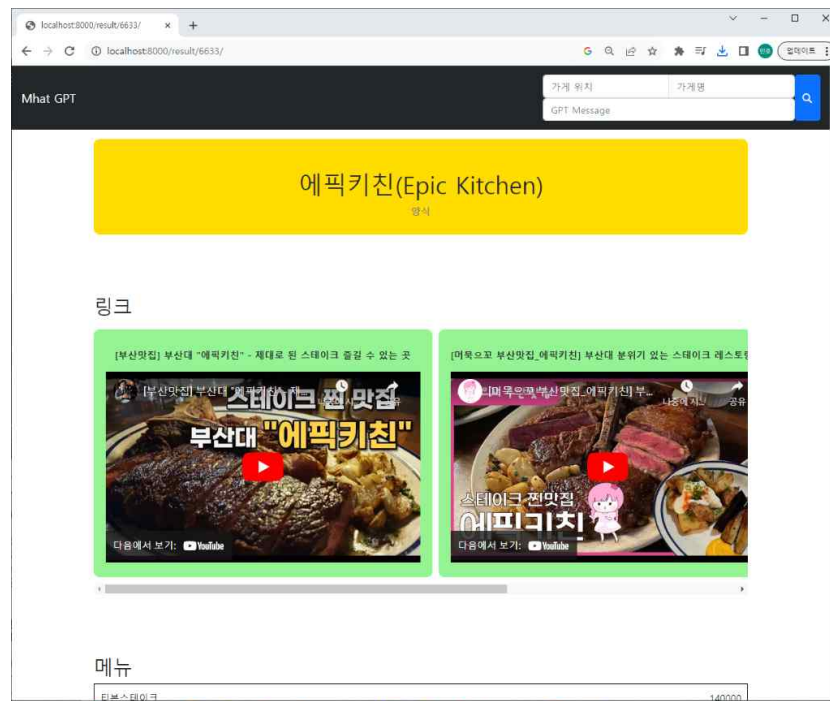


그림 25 - 상세정보 1

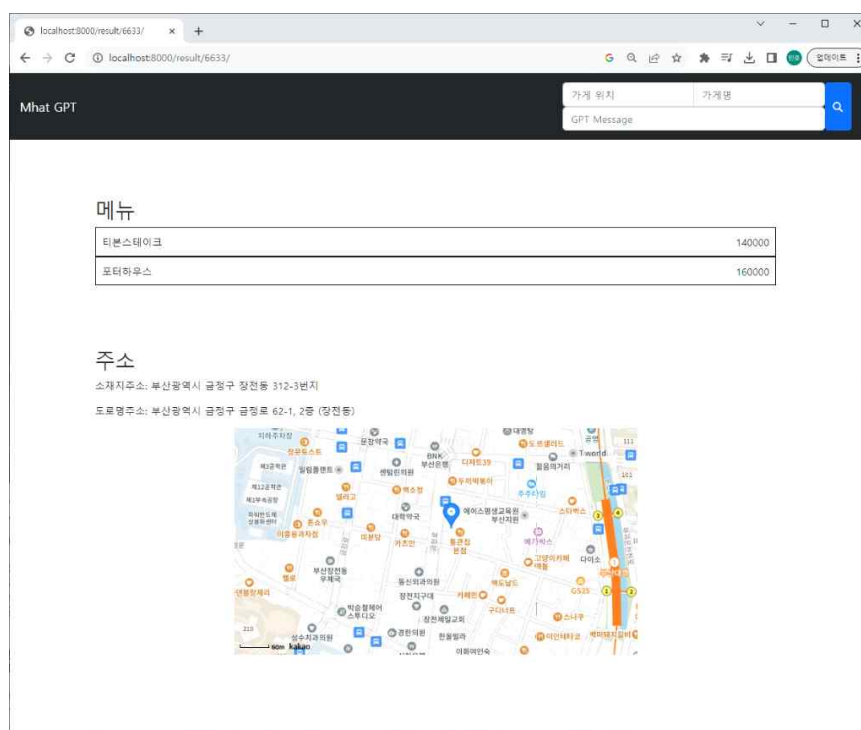


그림 26 - 상세정보 2

- 
- 식당 상세정보 화면에서는 관련 Youtube 영상, 메뉴, 주소 등 식당의 상세정보가 제공된다.
  - 관련 Youtube 영상은 YOLOv8을 이용해 확인된 관련 영상이 제공된다.

---

## 4. 결론 및 향후 연구 방향

### 4.1 결론

많은 사용자를 보유하고 있는 영상 플랫폼 Youtube를 이용해 맛집 추천 서비스를 구현하는 것이 목표였다. 목표를 달성하기 위해 Youtube 영상에 존재하는 Raw-Data를 활용해야 했고, 이를 위해 선택한 방법은 Object Detection에 사용되는 YOLO였다.

원활한 Object Detection을 수행하기 위해 학습 모델 또는 학습을 위한 충분한 이미지 데이터가 필요했다. 그러나 음식의 종류에 따라 얻을 수 있는 데이터와 모델의 편차가 심해 모든 음식에 대해 정확도 높은 탐색을 수행하는 것은 어려웠다.

또 조리 단계와 플레이팅에 따라 같은 메뉴라도 너무 대조되는 모습을 보이는 메뉴들이 존재했다. 따라서 완벽한 Object Detection을 수행하려면 메뉴 혹은 카테고리로 구분하여 학습 모델을 적용하는 것이 아니라, 모든 식당의 모든 메뉴에 대해 각각 학습 데이터를 만들어서 탐색을 수행해야 할 것으로 생각했다.

그러나 현실적으로 모든 식당 모든 메뉴에 대한 학습 모델을 생성할 수 없기에 다른 데이터를 함께 사용하는 방법을 선택했다. 다른 플랫폼에 존재하는 리뷰 데이터와 Youtube API 데이터를 이용해 문제점을 보완했다.

이를 위해 NLP, 바로 자연어 처리 기술을 활용했다. 키워드 추출을 이용해 리뷰 데이터에서 맛집에 대한 키워드를 추출했다. 또 크롤링을 통해 맛집에 대한 상세정보 및 다른 유사 서비스에서 평가하는 별점을 수집했다. 마지막으로 Youtube 댓글 내용을 감성 분석하여 맛집에 대한 반응을 수집했다.

이 과정에서도 역시 어려움은 있었다. 유명한 맛집은 수집할 수 있는 데이터가 많이 있어 맛집에 대한 자세한 정보와 신뢰할 수 있는 추천이 가능하다. 그러나 유명하지 않은 식당의 경우 수집할 수 있는 데이터가 없어서, 기본적인 정보를 제외하고 식당에 대한 자세한 설명 및 추천이 어렵다는 문제점이 있었다.

기술 및 데이터의 한계로 추천할 수 있는 맛집이 제한적이라는 것이 아쉬움으로 남는다. 그러나 이후 충분한 데이터를 얻을 수 있게 된다면 지금보다 사용자에게 더 많은 맛집을 추천할 수 있을 것으로 기대한다.

---

## 4.2 향후 연구 방향

향후 연구 방향은 기술과 편의 2가지 방향으로 구분했다.

먼저 기술적인 개선사항이다. 기술적인 부분은 다시 키워드 추출, 감성 분석, Object Detection으로 구분할 수 있다.

1) 키워드 추출에서는, 맛집에 알맞은 키워드를 추출하는 것이 핵심이다. 이를 위해 맛집과 관련된 더 많은 리뷰를 분석하여 키워드 사전을 발전시킬 예정이다. 또한 '맛있다'와 같이 너무 흔하게 사용되는 키워드를 처리하는 방법도 추가할 예정이다.

2) 감성 분석에서는, 긍정적 어휘와 부정적 어휘가 혼합된 댓글을 더 정확하게 분석할 수 있도록 개선할 예정이다. 특히, 댓글이 영상에 대한 댓글, 식당에 대한 댓글로 나뉠 수가 있는데 정확히 어떤 대상에 대한 댓글인지 구분하여 감성 분석의 정확도를 높일 예정이다.

3) Object Detection에서는, 더 높은 정확도를 보여주는 학습 모델을 생성하거나 찾아볼 예정이다. 이때, 메뉴에 대한 구분을 더욱 세분화하여 학습 모델을 적용하여 높은 정확도를 나타낼 수 있도록 할 계획이다. 마지막으로 맛집의 실제 음식 이미지를 얻을 수 있는 경우 학습 모델을 생성하는 과정에 이미지를 추가하여 플레이팅과 같은 이유로 탐색이 안 되는 문제가 발생하지 않도록 개선할 예정이다.

다음은 편의성을 위한 개선사항이다. 사용자가 서비스를 더욱 편하게 사용할 수 있도록 개선할 예정이다.

이를 위한 방안으로는, UI를 더욱 개선할 예정이다. 추천 맛집을 한눈에 볼 수 있도록 맛집 리스트를 제공할 것이다. 그리고 추천 맛집에 대한 정보를 더욱 직관적으로 알 수 있도록, 또 왜 서비스가 해당 맛집을 추천하는지 알 수 있도록 UI를 수정할 계획이다.

UI뿐만 아니라, 상세정보 페이지에도 더 많은 데이터를 추가할 예정이다. 유사한 맛집 추천 기능과 계절, 날씨, 기념일 등의 상황에 어울리는 상황별 맛집 추천 기능을 추가할 예정이다.

위의 개선사항으로 맛집 추천 서비스를 사용자들이 더욱 편하고 유용하게 사용할 수 있도록 발전시켜 나갈 계획이다.

## 5. 개발 일정 및 역할 분담

### 5.1 개발 일정

6월				7월				8월				9월				10월		
1주	2주	3주	4주	1주	2주	3주	4주	1주	2주	3주	4주	1주	2주	3주	4주	1주	2주	3주
데이터 수집																		
		웹 사이트 및 DB 구축																
					학습모델 생성 및 수집													
							이미지 데이터 전처리											
							관련 영상 판별											
								보고서 작성										
									웹 UI 개선									
										식당 추천 알고리즘 구현								
											서비스 기능 결합							
												오류 수정 및 최적화						
													데이터 추가 수집					
														NLP 기능 구현				
																수정사항 적용		
																	보고서 작성	



---

## 5.2 역할 분담

이름	역할
김민호	<ul style="list-style-type: none"><li>- Django를 이용한 웹 서비스 UI 제작</li><li>- SQLite를 이용한 DB 설계</li><li>- 기능 통합 작업 수행</li></ul>
하평안	<ul style="list-style-type: none"><li>- 웹 크롤링 통한 Data 수집</li><li>- NLP(자연어 처리 - 키워드 추출 및 감정분석) 기법을 통한 데이터 처리</li><li>- 맛집 추천 알고리즘 제작</li></ul>
한정훈	<ul style="list-style-type: none"><li>- YOLOv8 Custom 학습을 통한 이미지 탐색 수행</li><li>- 시장 조사 및 차별화된 서비스 개발</li><li>- 시스템 구조 설계 및 기술 문서 작성</li></ul>

## 6. 참고 문헌

- [1] Ultralytics YOLOv8 Docs, Available : <https://docs.ultralytics.com/>
- [2] Roboflow Universe, Available : <https://universe.roboflow.com/>
- [3] Youtube, Availalbe : <https://www.youtube.com/>
- [4] Kakao Maps, Available : <https://apis.map.kakao.com/web/>
- [5] Django Documentation, Available: <https://docs.djangoproject.com/en/3.2/>
- [6] Bootstrap, Available : <https://startbootstrap.com/template/sb-admin>
- [7] Github, Available : <https://github.com/park1200656/KnuSentiLex>