

네이버 리뷰 AI 분석 서비스 최종 보고서

과목명: 기계학습의 이해 (Machine Learning) 학번/이름: 202400190 최서연

1. 프로젝트 개요

1-1 무엇을 만들었나?

본 프로젝트는 네이버 플레이스의 방문자 리뷰를 긍정과 부정으로 분류하고, LLM을 이용하여 상세 내용을 분석 및 요약하는 AI 서비스입니다. 이를 위해 먼저 pre-trained된 KcBERT 모델을 fine-tuning 하였습니다. 이후 사용자가 식당 URL을 입력하면 리뷰를 자동으로 Crawling하고, 학습된 모델과 LLM이 협력하여 리뷰를 분류하고 요약하는 통합 서비스를 구축하였습니다.

1-2 어떤 문제를 해결했나?

현재 네이버 식당 리뷰는 별점 시스템이 사라지고, 광고성 리뷰가 많아 식당에 대한 정보를 얻기 위해서는 일일이 리뷰를 확인해야 하는 번거로움이 있습니다. 이러한 문제를 해결하기 위해 공개 데이터셋 (Kr3)에 더해, 제 기준에 맞게 직접 라벨링한 데이터셋으로 학습시킨 AI 모델을 활용했습니다. 웹 크롤링과 LLM API를 연동한 웹사이트를 통해, 사용자는 리뷰를 일일이 읽지 않고도 식당의 정보를 한눈에 빠르게 파악할 수 있습니다.

2. 진행 과정

2-1 주제 선정 및 문제 정의(Assignment 3)

제가 실제로 관심이 있고 자주 사용할 만한 서비스를 모색했습니다. 동시에 유의미한 성능의 모델 구축을 목표로, 후보 주제들 중 적절한 공개 데이터셋이 존재하는 주제로 최종 선정하였습니다.

2-2 데이터 수집 및 분석(Assignment 4)

Kr3(Korean restaurant review with ratings) 공개 데이터셋을 기반으로 하였으며, 그 중 일부 리뷰를 샘플링하여 문장 단위로 직접 라벨링한 데이터셋을 추가로 만들었습니다. 이후 데이터셋에 대한 다양한 분석을 수행하여 클래스 불균형을 처리할 계획을 수립하고, 하나의 리뷰 안에 긍정과 부정이 섞여 있는 복합적인 문맥을 섬세하게 처리하기 위해서 어떻게 서비스를 구축해야 할지 고민해보았습니다.

2-3 ML 모델 학습 및 평가(Assignment 5)

한국어 댓글 분석에 강한 KcBERT 기반의 pre-trained된 모델의 fine-tuning을 진행하였습니다. 데이터셋을 로드 후 전처리 과정을 거쳐 Train/Val/Test set으로 각각 분할하였습니다. 이후 pre-trained된 모델을 불러와 학습을 시킨 후 Google Drive에 저장하였습니다.

리뷰 중에서도 특히 부정 리뷰를 정확히 걸러내는 것에 집중하기 위해 Macro F1-score 및 Negative recall을 핵심 지표로 선정하였습니다. 모델을 평가해본 결과 total accuracy는 0.9650, Macro F1-score는 0.9344, Negative recall은 0.9018로 유의미한 성능을 확인하였습니다.

2-4 서비스 개발(Assignment 6)

Selenium(동적 크롤링)과 BeautifulSoup(HTML 파싱)를 이용하여 crawler.py 모듈을 구현했습니다. 이 과정에서 특수 문자를 제거하고, 정규표현식과 수동으로 입력한 예시 리스트를 이용해 순수 리뷰 텍스트만을 추출하도록 최적화하였습니다.

메인 서비스인 app.py 파일에서는 Python Streamlit을 이용하여 웹 기반 대시보드를 구축하였습니다. 학습시킨 ML 모델이 예측 결과와 함께 confidence를 도출하도록 하여 리뷰를 수치적으로 분석하였습니다. 동시에 LLM API(Gemini)를 호출하여 분류된 리뷰를 분석하고 요약하였습니다.

3. 모델을 서비스로 만든 구조

3-1 시스템 파이프라인

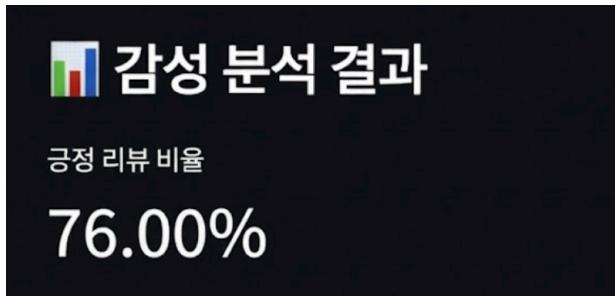
- User Input : 사용자가 네이버 플레이스 URL 입력
- Crawler Engine : Selenium이 Headless 모드로 브라우저를 띄워 실시간 리뷰 수집
- Preprocessor : 텍스트 정제(이모티콘, 특수문자, 광고성 키워드 등 제거)
- Dual AI Engine
 - Engine A(PyTorch) : assignment5_final_model이 각 리뷰의 긍정 확률(confidence) 계산
 - Engine B(Gemini API) : 리뷰의 유효성 검증 및 핵심 키워드 기반 문장 요약
- Visualization : Altair 차트 및 Streamlit 데이터프레임으로 최종 결과 시각화

3-2 핵심 기술적 특징

- gdown 라이브러리를 활용하여 Google Drive에서 학습된 모델 자동으로 다운로드
- ML 모델과 LLM API의 협력 : 문맥 이해와 창의성이 필요한 요약은 Gemini가 담당
- 텍스트 수집 후 정교한 전처리 과정을 통해 순수 리뷰 텍스트 추출 : 크롤러 단계에서 7단계의 필터 와, 제외할 키워드 리스트를 통해 1차 정제. 앱 단계에서 LLM이 최종적으로 검토 진행
- 부정 리뷰를 우선적으로 시각화 및 분석하는 알고리즘 구현

4. 실제로 사용하며 개선

[1차 개선]



ex) 총 리뷰 두 개 중 한 개가 긍정 -> 50%



ex) 총 리뷰 두 개가 각각 긍정 70%, 90% -> 80%

감정 분석 결과를 보여줄 때 단순히 긍정 리뷰가 몇 퍼센트인지 보여주는 대신, 리뷰들의 긍정 Confidence의 평균을 보여주는 방식으로 변경. 같은 긍정 리뷰라고 해도 긍정에 대한 confidence가 다를 수 있는데, 이러한 상황을 좀 더 섬세하게 표현해줄 수 있음. 또한, 크롤링한 리뷰가 전부 긍정으로 나오는 경우에도 문맥 속에 섞여 있는 작은 불만이나 아쉬운 점들이 점수에 반영되어 더 디테일한 결과를 보여줄 수 있음.

[2차 개선]

	Sentiment	Confidence	Review
0	긍정 😊	96.25%	맘스터치 외대점햄버거알림받기
1	부정 😥	99.21%	감자튀김에 무슨 치킨 튀김가루인지 찌거기도 많고, 농축함은 어제 만든거 꺼내먹는
2	부정 😥	99.82%	가보았던 맘스터치매장 싸이버거중 제일 맛이없어요. 양상추는 신선했으나 소스발
3	긍정 😊	99.78%	치킨이 맛있고 매장이 청결해요
4	부정 😥	99.66%	감자튀김을 시켰더니 밀가루덩어리인지 뭔지를 서비스로 주네요
5	긍정 😊	99.75%	엄마의 손길... 너무 맛있네요...
6	긍정 😊	97.60%	음식 맛있게 잘 먹었습니다. 배달을 주력으로 하시는지 매장을 좀 어수선했습니다. ↪
7	부정 😥	94.79%	조리 중에 크림치즈볼의 치즈가 다 흘러나간 듯합니다. 4개 중 2개가 텅 비어있는 상
8	부정 😥	96.04%	맛은 너무 좋은 느낌을 주는데 위생이 shittttt 위생만 좀 신경써 주세요 치즈볼은 치즈
9	부정 😥	99.14%	ㅁ맘스터치 오랜만에 왔는데 패티도 껍질이 너무 두껍고 감자튀김도 한지 오래되고

<문제점>

- 크롤링 과정에서 리뷰가 아닌 것이 리뷰로 들어옴 (ex 맘스터치 외대점햄버거알림받기)
- 원본 리뷰 텍스트를 가져오니 길이나 말투가 통일되지 않아 읽기 불편함
- Sentiment나 Confidence를 정렬해서 보여주는 것이 훨씬 정보 전달이 잘 될듯 함

<1,2 해결> LLM API 활용

	Sentiment	Confidence	Keyword_Summary
0	부정 😞	99.21%	감튀 뉙눅하고 햄버거는 차가워요.
1	부정 😞	99.82%	싸이버거가 맛없고 매장이 지저분해요.
2	긍정 😊	99.78%	치킨이 맛있고 매장이 청결해요.
3	부정 😞	99.66%	감튀에서 밀가루 덩어리가 나왔어요.
4	긍정 😊	99.75%	정말 맛있어요.
5	긍정 😊	97.60%	음식은 맛있지만 매장이 어수선하고 위생이
6	부정 😞	94.79%	크림치즈볼 치즈가 다 흘러나와 아쉬워요.

LLM(Gemini)을 활용하여 원본 리뷰 텍스트에서 키워드를 추출한 후, 그 키워드를 기반으로 적절한 길이의 자연스러운 문장 생성 -> 통일된 길이와 말투로 훨씬 보기 편해짐

★ LLM 일괄 처리 중... (키워드 추출 및 태그 필터)

- ④ LLM이 태그로 판단하여 제거: 맘스터치 외대점햄버거알림받기...
- ④ LLM이 태그로 판단하여 제거: Byrus 러스에 의한 바이러스...

웹 크롤링에서 미처 걸러내지 못 한 리뷰가 아닌 텍스트는 LLM이 마지막으로 검토 후 제거

<3 해결> 부정(내림차순) -> 긍정(오름차순) 순으로 정렬

 상세 분석 결과 (키워드 요약)

	Sentiment	Confidence	Keyword_Summary
0	부정 😞	99.82%	싸이버거 맛없고 매장도 지저분했어요.
1	부정 😞	99.66%	감튀 대신 밀가루 덩어리를 줬어요.
2	부정 😞	99.45%	감튀는 식었고, 직원들이 너무 바빠요.
3	부정 😞	99.25%	콜라는 김 빠지고 햄버거는 차가워요.
4	부정 😞	99.21%	매장 관리 안 되고 음식 맛도 변했어요.
5	부정 😞	99.21%	감튀 찌꺼기 많고 뉙눅, 버거도 차가웠어요.
6	부정 😞	99.14%	패티 두껍고 감튀 오래돼서 아쉬워요.
7	부정 😞	96.04%	맛은 좋으나 위생과 치즈볼이 아쉬워요.
8	부정 😞	94.79%	크림치즈볼 2개가 텅 비어 있었어요.
9	긍정 😊	55.62%	콘샐러드 1개가 누락되었어요.

[3차 개선]

다음은 리뷰에서 주로 언급된 긍정적인 내용 3가지입니다.

1. 다양한 메뉴의 빙수와 디저트가 맛있습니다.
2. 직원들의 친절한 응대와 세심한 서비스에 만족합니다.
3. 넓고 쾌적하며 깔끔한 매장 환경이 좋습니다.

사람들이 주로 언급하는 내용은 다음과 같습니다.

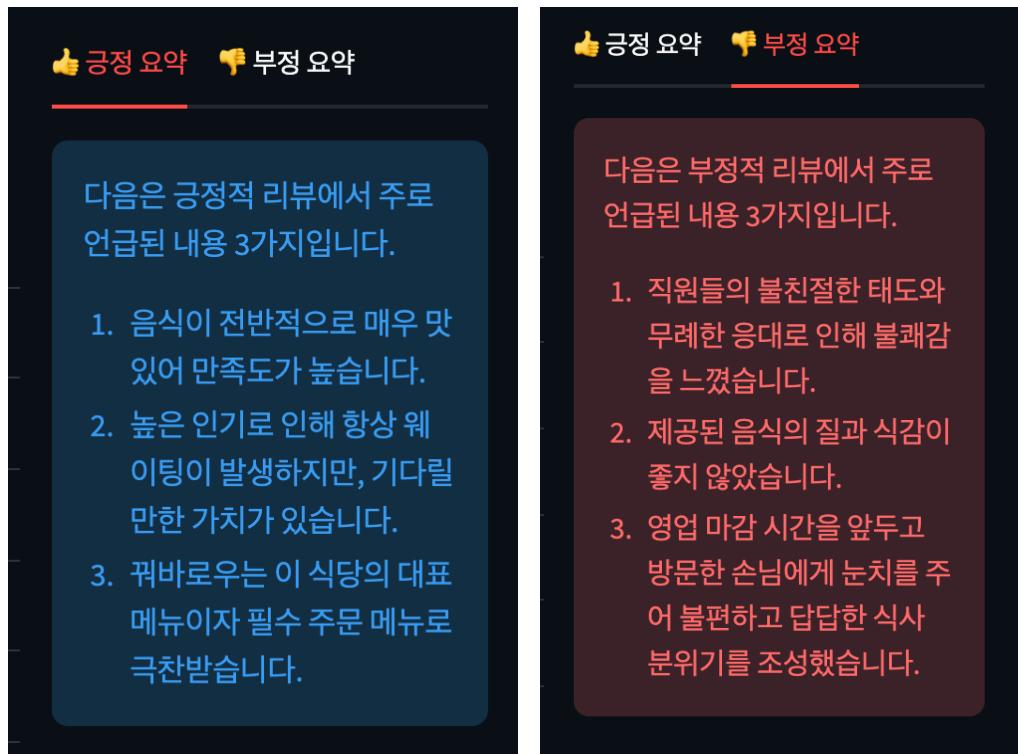
- 주차 공간 부족 및 불편
- 식기류 위생 불량
- 포장 시 보냉재 미제공으로 인한 상품 손상

<문제점> 같은 식당 리뷰 분석인데, 구조나 말투 면에서 통일성이 부족

<해결> LLM 호출 프롬프트 수정(템플릿 지정 등)

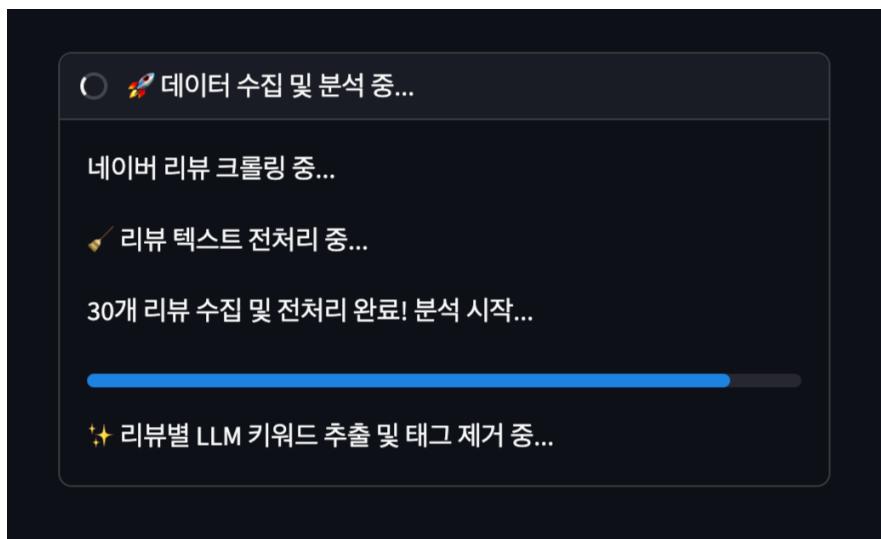
```
prompt = f"""\n\n다음은 식당에 대한 '{sentiment_type}' 리뷰들입니다.\n이 리뷰들을 분석해서 사람들이 주로 언급하는 내용을 3가지 항목으로 요약해주세요.\n결과는 반드시 한국어로 작성하고, 각 항목은 한 줄로 간결하게 써주세요.\n\n->
```

```
try:  
    genai.configure(api_key=api_key)  
    gemini_model = genai.GenerativeModel("gemini-2.5-flash")  
    text_chunk = "\n".join(reviews[:30])  
  
    prompt = f"""\n\n다음은 식당에 대한 '{sentiment_type}' 리뷰들입니다.\n이 리뷰들을 분석하여 공통적인 핵심 내용 3가지를 추출하고, 아래 형식에 맞춰 작성하세요.  
  
[출력 형식]  
다음은 {sentiment_type} 리뷰에서 주로 언급된 내용 3가지입니다.  
  
1. (핵심 내용 1) ~습니다.  
2. (핵심 내용 2) ~습니다.  
3. (핵심 내용 3) ~습니다.  
  
[제약 사항]  
- 반드시 위의 출력 형식을 그대로 따르세요.  
- 각 문장의 어미는 반드시 '~습니다' 체(하십시오체)로 끝나야 합니다. (예: 맛있습니다, 친절합니다)  
- 서두나 맺음말 없이 위 형식의 텍스트만 출력하세요.  
  
[리뷰 데이터]  
{text_chunk}  
.....
```



-> 통일감이 생겨 정보 전달이 잘 되고, 보기 편함

[4차 개선]



<문제점> LLM 기반으로 리뷰 키워드 추출 및 자연스러운 문장을 생성하는 과정이 시간이 너무 오래 걸림

<해결> LLM batch 처리를 통한 성능 최적화

1. 개별 데이터 처리 시 발생하는 응답 지연 문제를 해결하기 위해 LLM Batch Processing 도입
2. 수집된 데이터를 일정 단위로 묶어 단위 프롬프트로 통합 후 LLM에 일괄 전송
3. LLM이 다수의 데이터에 대한 키워드 필터링 및 요약 작업을 동시에 수행하도록 지시
4. 연산 결과는 구조화된 JSON 포맷으로 한 번에 반환되도록 하여 후처리 효율성 확보
5. 불필요한 API 호출을 최소화하고, 처리 속도를 전에 비해 눈에 띄게 단축

[5차 개선] - 디자인

- 감성 분석 결과 긍정 및 부정 막대 그래프 색 구별되도록 변경
- 분석 중인 식당 이름도 크롤링 목록에 포함하고, 웹에 표시
- 이모티콘을 통해 정보가 한 눈에 들어오도록 함

[최종]

맘스터치 외대점

感悟 分析 결과

평균 긍정 확률
66.5%

총 28개 리뷰 중 긍정 19개

Sentiment	Count
긍정 😊	19
부정 😞	9

다음은 긍정적 리뷰에서 주로 언급된 내용 3가지입니다.

- 음식이 매우 맛있습니다.
- 직원들이 친절하고 서비스가 신속합니다.
- 매장이 청결하며 편안하게 이용할 수 있습니다.

무난한 식당

상세 분석 결과

	Sentiment	Confidence	Keyword_Summary
0	부정 😞	99.82%	싸이버거가 맛없고 매장도 지저분해요.
1	부정 😞	99.66%	서비스로 받은 것이 밀가루 덩어리였다.
2	부정 😞	99.45%	식은 감튀와 알바 부족으로 매장이 바빠 보여요.
3	부정 😞	99.25%	콜라 김이 빠지고 햄버거가 차갑다.
4	부정 😞	99.21%	매장 관리와 맛이 아쉬워 개선이 필요하다.
5	부정 😞	99.21%	감자튀김과 싸이버거 모두 상태가 매우 안 좋음.
6	부정 😞	99.14%	패티 껍질이 두껍고 감튀가 오래되어 성의가 없다.
7	부정 😞	96.04%	맛은 좋으나 위생과 치즈볼이 아쉽다.

5. 배운 점 및 개선 방향

5-1 배운 점

- 유의미한 모델 성능을 확보하기 위해서는 데이터의 양과 질이 정말 중요한 것 같습니다.
- 단순히 정확도 수치만 보는 것이 아니라, 모델이 어떤 케이스에서 틀리는지 분석하는 과정(오류 분석)이 모델의 성능을 개선하는 데에 있어서 생각보다 큰 영향력을 가진다는 점을 알게 되었습니다.
- 데이터 수집부터 서비스 구축까지 전체 과정을 직접 구현하는 경험을 해보았습니다.
- PyTorch(모델 학습), Streamlit(웹 구현), Selenium(크롤링), LLM API 등의 다양한 프레임워크를 활용하여 하나의 서비스를 구축해보는 경험을 통해 그 사용법을 익혔습니다.
- LLM API를 호출하는 과정에서 응답을 JSON 포맷으로 제어하는 프롬프트 엔지니어링 기술을 익히고, API 비용과 속도를 고려한 최적화 방법을 학습하였습니다.

5-2 개선 방향

- 현재 모델은 한국어에 특화된 KcBERT 모델을 기반으로 하여, 한국어가 아닌 영어 리뷰에서는 정확도가 떨어집니다. 추후 다국어 데이터셋을 추가 학습시키거나 Multilingual BERT를 도입하여 다국어 리뷰의 처리도 해보고 싶습니다.
- 현재는 정보성이나 애매한 리뷰에도 긍정 또는 부정으로 이진 분류를 진행하는데, 중립 등의 클래스를 추가하여 다중 분류 모델을 만든다면 더 섬세하게 분석할 수 있을 것 같습니다.