# On My Way - 경로 주변 장소 검색을 한눈에







GitHub

컴퓨터학과 18학번 장세윤

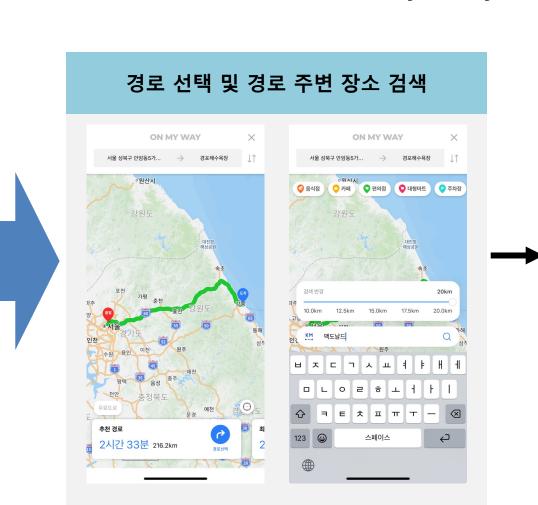
컴퓨터학과 18학번 전홍찬 환경생태공학부 18학번 이상연

# KOREA

# ■ 프로젝트 배경

네이버맵, 티맵 등 기존의 지도 서비스를 이용하는 사용자는 경로 주변 장소 검색 시 아래와 같은 반복되는 복잡한 절차를 거쳐야 한다는 Painpoint가 존재한다. 맛집, 편의시설 등 경로 주변의 장소 검색 시 발생하는 기존 서비스의 Painpoint를 개선하기 위해 OnMyWay는 인앱 환경에서 **주행 경로를 따라 필요한 장소를** 빠르게 찾고, 한눈에 비교하는 서비스를 제공하고자 한다. 나아가 LLM 리뷰 요약 서비스를 제공해 성공적인 장소 검색/추천 서비스를 제공함으로써 편리함을 도 모하고자 한다.

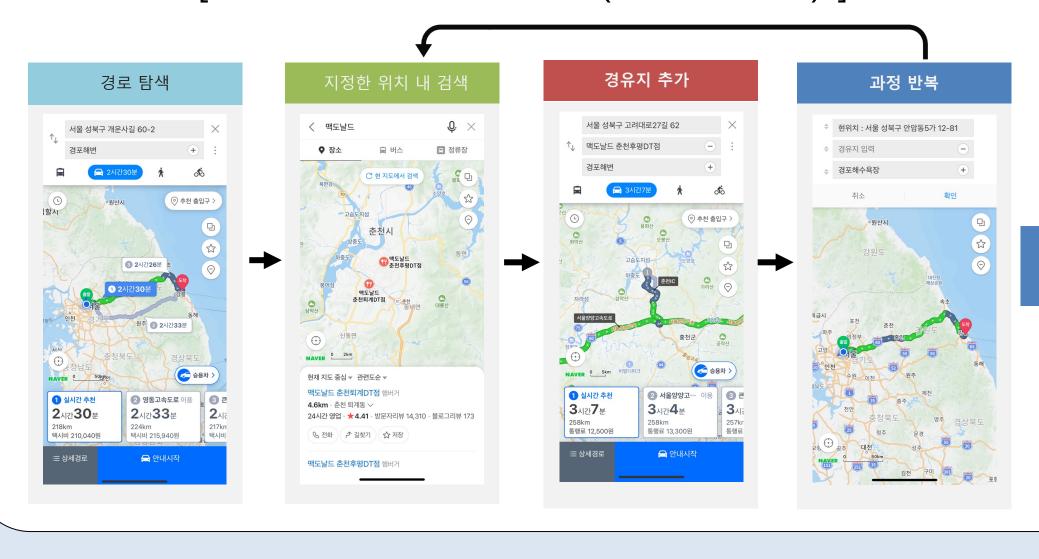
[기존 서비스 이용 시 (네이버지도)]

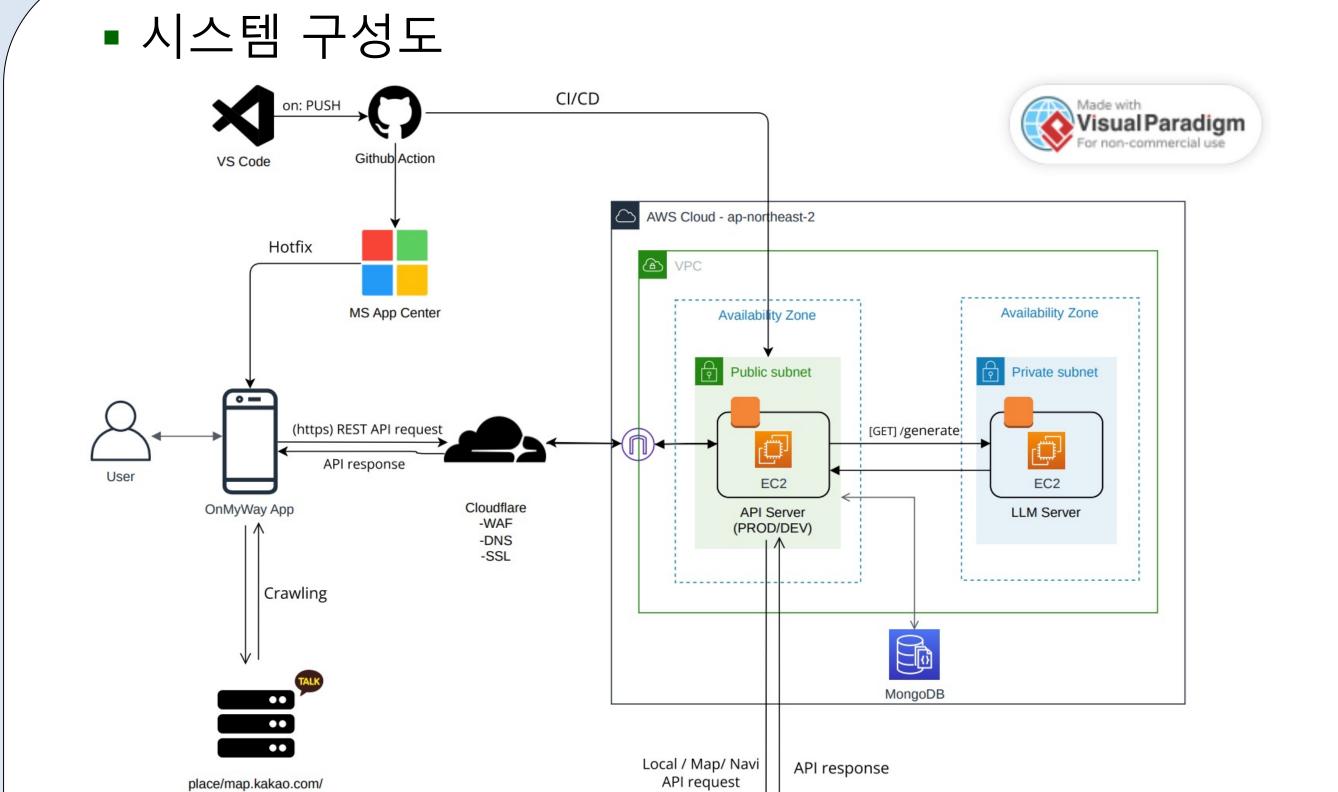


# [OnMyWay 이용 시]

# ■ 유사 서비스 비교

	OnMyWay	티맵	카카오맵	네이버맵	트리플
경로 주변 장소 검색	О	Δ	X	X	X
AI 리뷰 요약	О	X	X	X	O
경유지별 추가 되는 시간 표시	O	0	O	O	X
장소 결과 핕터링	O	X	X	X	X





Kakao Map/Local API server

■ 개발 일정

# ■ 주요 기능 및 구현

#### - 경로 주변 장소 탐색 기능

[설명] 원하는 키워드 입력 / 카테고리 선택 시 현재 주행 경로를 기준으로 일정 반경 내의 검색 결과를 지도상에서 확인, 지도 위 마커를 클릭하면 사용자 리뷰 등 자세한 정보를 한눈에 확인 [구현] 이동 경로에서 최대 20개의 좌표 값을 추출한 뒤, 각 좌표에서 최대 20km 반경으로 Kakao Local Search API 응답을 활용해 이동 거리 별로 검색 결과 수 조정 및 중복 제거

#### - 검색 결과 정렬 및 필터링

[설명] 별점, 리뷰, 영업 중, 주차가능 여부를 기준으로 사용자가 원하는 장소만 지도에서 확인 [구현] 클라이언트 디바이스에서 크롤링 후 filtering / sorting 알고리즘 적용

#### - 특정 장소 경유 시 변경사항 안내, 티맵 / 네이버맵 네비게이션 연동

[설명] 선택된 장소를 경유할 시 변경 되는 경로 및 추가 시간 안내와 더불어 네이버맵, 티맵 네비게이션으로 이어 안내 받을 수 있음

[구현] Kakao Navi API 활용. 경유지 최적화 알고리즘 구현, 네이버맵 / 티맵 서비스 SDK를 네이티브 모듈화 후 React Native에 적용

### - LLM 리뷰 봇 : 장소 리뷰 요약

[설명] 선택된 장소에 대해 리뷰를 남긴 사용자들의 리뷰를 최대 300자 내외로 요약 [구현] 클라이언트에서 수행한 크롤링 데이터 전처리 후 Fine-tuned LLM 활용

# ■ 개발 스택

[FE]





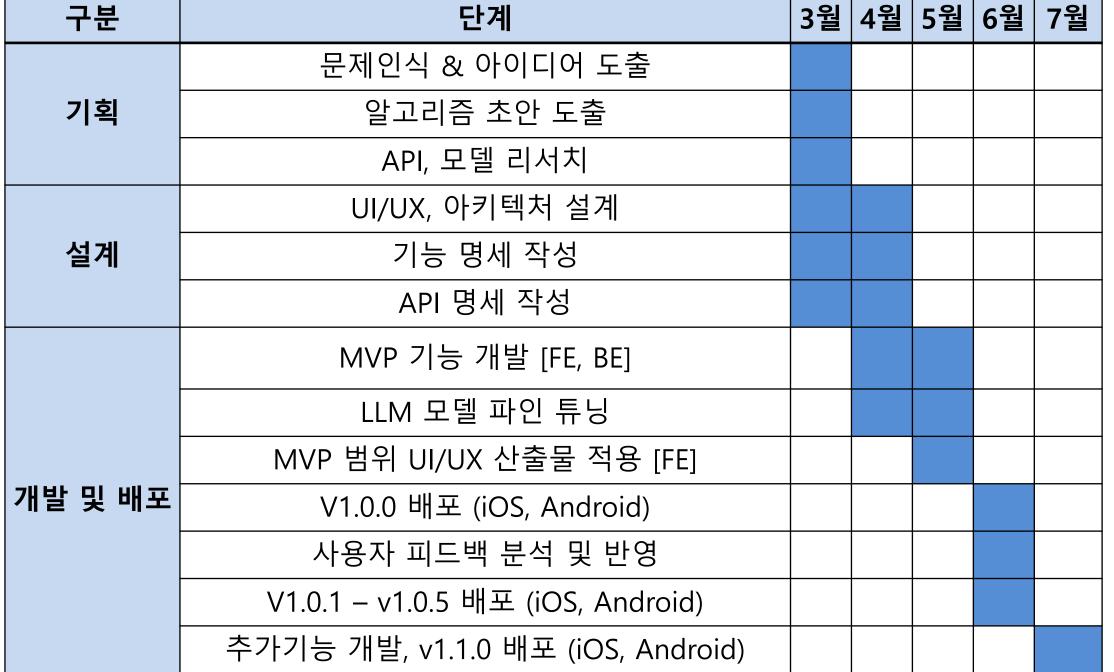
[CI/CD]



[기타]







# ■ 출시 현황 (iOS, Android)

날짜	버전	주요 업데이트 내용	
	v1.0.0	출시	
24.06.04	v1.0.1-v1.0.2	Minor Bug fix	
24.06.14	v1.0.3	주차 가능여부로 결과 필터링 기능 추가	
24.06.15	v1.0.4	카테고리로 검색 기능 추가 경로 선택 시 '무료도로' 옵션 추가	
24.06.16	v1.0.5	(Android) 티맵 연동 기능 추가(Android) 티맵 안내 시 최적 경로 적용iOS, Android 통합 다운로드 수 100+	
24.06.17	v1.0.6		
현재	v1.0.6		
7월 중	v1.0.1	(iOS) 티맵 연동 기능 추가 LLM 리뷰 요약 봇 기능 출시 Business Model 도입 (광고, 멤버십)	

# Fine-tuned LLM: Gemma-7b-ko-sum



# 1. LLM 파트 목표

# 1) Open source LLM 지향

앱 배포 중에도 업데이트를 지속적으로 할 수 있는, 개발자가 예상 범위 안에서 통제할 수 있는 모델을 지향한다. 그래서 이번 모델은 open source 기반 LLM 중에서 선택했다. 그 중 Gemma 모델은 오픈소스이기에 수정이 가능하고 앱 배포 중 수정 및 성능 개선 가능하다는 점에서 이번 프로젝트의 목적에 부합한다. 비록 GPT, Claude 같은 비 오픈소스 모델과 오 픈 소스 모델의 성능 차이가 있지만, GPT는 타사 api를 그대로 불러오기에 예상 외 상황에 대처할 수 없다는 단점이 있다.

# 2) On-device LLM 지향

이번 프로젝트에서는 서버에 업로드/보관이 용이한 sLLM을 사용하고자 한다. 더욱이, LLM의 용량이 서버 유지비용에 영 향을 주는 점이 sLLM 채택에 크게 작용했다. 예시로, GPT-4같은 경우에는 파라미터 수가 1조 7000억 개 수준으로 추정이 되지만, 이번 프로젝트에서 사용한 Gemma-7b-it은 70억 개 수준의 파라미터를 가지고 있다. 즉, 240배 이상의 용량 차이 를 가지고 있다는 것이다. 이는 프로젝트의 앱 배포에 있어 서버 유지에 영향을 줄 수 있는 정도의 차이이다. 따라서 최대 한 small LLM 사용을 지향하되 높은 성능을 가진 LLM과 "리뷰 요약 태스크"에 한해 비슷한 결과물을 출력할 수 있는 방 향을 채택하게 되었다.

# 2. LLM 파인 튜닝 과정

# 1) 데이터셋 사용

네이버 뉴스 요약문 데이터셋 22K, Alhub 요약문 데이터셋 11k, 총 33k 크기의 한국어 요약 데이터셋을 사용했다. 상이한 요약문 데이터 제공 방식을 고려해 (제목, 본문, 요약) 칼럼을 추출 후 전처리를 진행했다.

# 2) 프롬프트 조정

요약성능을 개선하기 위해 학습 시 프롬프트를 조정했다. 우선 가게 리뷰를 요약한다는 태스크를 모델이 자연스럽게 수행할 수 있도록 Instruction 설정은 "당신은 친절한 여행 가이드입니다. 여행 가이드는 장점 / 단점을 명확히 말해야 합니다."로 잡았다. 다만 가이드의 역할을 모델이 잘 이해하지 못했기에 뉴스 요약문 텍스트 3개를 따로 추출해 각각 **가이드의 말투로 직접 수정**하여 프롬프트에 예시로 추가하는 과정이 필요했다.

#### LLM 제작/배포 Pipeline

(O)

Sagemaker

# 3. LLM 배포 과정

- 배포 서버 후보군 선정 및 Ollama 사용 시연

우선 lambda layer와 EC2를 사용해 업로드하는 안을 고안했다. EC2 에서 해당 모델을 업로드, 유지하는 비용이 최소 시간 당 USD 0.1344 정도 청구되어 유지비용만 한 달에 13만원, 1년에 160만원정도가 필 요했다. 지속적인 배포를 유지하기에는 어려움이 따르지만 배포 과정 자체는 어렵지 않았다.

두 번째 안은 sagemaker를 사용하는 것이었다. 개발, 학습, 배포까 지 진행할 수 있는 클라우드 서비스였으나 시간 당 USD 0.2 정도 청 구되었고 1년에 240달러 정도가 필요했다. 따라서 sagemaker가 아닌 EC2를 이용한 배포를 LLM 배포 방향으로 채택하게 되었다.

EC2를 이용해 배포하는 것은 GPU 가속을 지원하는 g5.12xlarge (NVIDIA A10G GPU, 12GB) 인스턴스를 사용한 배포가 적합하다. 다 만 계속 클라우드 이용료를 고려해 추후 배포 예정이다.

# def generate\_prompt(example):

prompt\_list = [] for i in range(len(example['document'])):

prompt\_list.append(r"""<bos><start\_of\_turn>user 당신은 친절한 여행 가이드입니다. 여행가이드는 장점과 단점을 명확히 말해야 합니다.

{}<end\_of\_turn> <start\_of\_turn>model

{}<end\_of\_turn><eos>""".format(example['document'][i], example['summary'][i])) 📍 return prompt\_list

프롬프트 설정