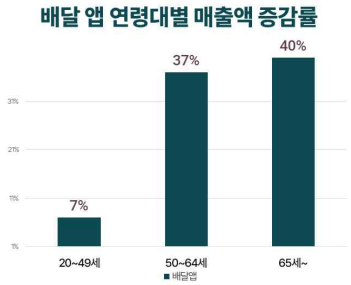
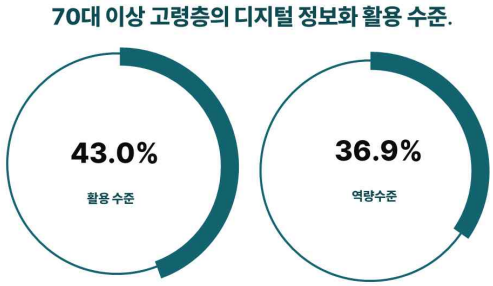
IITP 프로젝트 결과 보고서

<table>
<tr><td>제목</td><td colspan="2">고령층을 위한 음성 기반 주문 서비스</td></tr>
<tr><td>조명</td><td colspan="2">늘곁에</td></tr>
<tr><td rowspan="5">조원</td><td>성명</td><td>역할</td></tr>
<tr><td>신홍재</td><td>RAG아키텍쳐 설계, RAG/LLM 구현</td></tr>
<tr><td>김성룡</td><td>추천 가게 선정, 프로그램 설계</td></tr>
<tr><td>김원우</td><td>NLU, FSM 설계, 앱 설계</td></tr>
<tr><td>한동훈</td><td>STT, NLU 학습, TTS 구현</td></tr>
<tr><td>목적</td><td colspan="2">




Kb국민카드 data root – 시니어 소비, 2040세대보다 매출액 증가율 앞서

최근 배달 앱 시장에서 고령층의 이용 증가가 두드러지고 있습니다. 특히, 65세 이상 연령대의 배달앱 매출액 증가율은 40%로, 20~49세(7%)나 50~64세(37%)보다 높은 수치를 기록하고 있습니다. 이는 고령층의 배달 서비스 수요가 꾸준히 증가하고 있음을 보여줍니다.

그러나 70대 이상의 고령층은 디지털 정보화 활용 수준이 43%에 불과하고, 역량 수준도 36.9%로 낮은 편입니다. 이러한 디지털 역량 격차는 고령층이 기존 배달 앱을 이용하는 데에 큰 장벽이 될 수 있습니다.

이에 따라 본 프로젝트는 고령층 사용자들이 기존의 복잡한 스마트폰 배달 애플리케이션 대신, 전화 통화하듯이 음성으로 음식 주문을 할 수 있는 모바일 애플리케이션을 개발하는 것이다. 사용자가 스마트폰 앱에서 직접 말로 주문 내용을 이야기하면, 시스템이 음성을 인식하여 적절한 음식점과 메뉴를 찾아 주문을 진행하고, 결과를 다시 음성으로 안내한다. 이러한 음성 주문 비서를 통해 디지털 기기에 익숙하지 않은 노년층도 배달 서비스를 손쉽게 이용할 수 있도록 돕는 것이 목표이다.

이 프로젝트에서는 최신 AI 기술들을 활용하여 음성 인터페이스와 배달 주문 시스템을 통합하였다. 음성 인식(STT) 기술로 사용자의 말을 텍스트로 변환하고, 자연어 이해(NLU)와 대화 관리(FSM)를 통해 사용자의 의도를 파악하며 주문 절차를 단계별로 진행한다. 또한 대규모 언어모델(LLM)과 지식 검색(RAG) 기법을 결합하여 음식점 정보와 메뉴 데이터를 실시간으로 조회하고, 정확한 답변을 생성한다. 마지막으로 텍스트 합성(TTS) 기술을 이용해 시스템의 응답을 다시 음성으로 변환하여 사용자에게 제공한다.
결과적으로 본 프로젝트의 산출물은 사용자가 음성만으로 배달 음식을 주문할 수 있는 스마트폰 애플리케이션이며, 사용자 경험은 실제 전화로 음식점에 주문하는 과정과 유사하도록 설계되었다. 이를 통
</td></tr>
</table>

	해 디지털 기기에 익숙하지 않은 고령층도 별도의 교육 없이 자연스럽게 배달 서비스를 이용할 수 있으며, 더 나아가 음성 기술을 활용한 신규 서비스 모델의 가능성을 모색하였다.
수행 내용	# 1. 데이터 수집 및 전처리 **데이터 수집:** 본 프로젝트에서는 국내 배달앱 **요기요** 웹사이트로부터 음식점 정보, 메뉴, 리뷰 데이터를 크롤링하여 활용했습니다. Python 기반 크롤러를 구현하여 약 **1,358개 음식점**의 상세 페이지를 수집했고, 이를 통해 **약 6.3만 개의 메뉴**와 **6.7만 건의 리뷰** 데이터를 확보했습니다. 크롤링에는 Selenium을 이용해 동적으로 콘텐츠를 로드하고, BeautifulSoup으로 HTML을 파싱하여 원하는 필드를 추출하는 방식을 사용했습니다. 크롤링 과정에서 IP 차단이나 동적 콘텐츠 로딩 이슈가 발생하지 않도록 요청 간 딜레이를 주고, 필요 시 헤더 정보를 변경하는 등 안정성을 확보했습니다. **전처리 과정:** 수집한 원천 데이터에 대해 정밀한 전처리를 수행했습니다. 중복된 데이터(예: 중복 메뉴나 동일 리뷰)가 발견될 경우 ID나 내용 기준으로 **중복 제거**를 시행했습니다. 텍스트 정보(메뉴명, 리뷰 등)는 **정규화**를 거쳤는데, 특수문자나 이모티콘 제거, 일관된 어휘 통일(예: 숫자 표현 통일, 불필요한 공백 제거) 등의 작업을 포함합니다. 특히 리뷰 데이터의 경우 일부 오탈자나 비속어가 존재하여, 한글 맞춤법 교정기를 통해 가능한 한 표준어에 가깝게 정제하였습니다. **청크 단위 분리:** 후속 벡터 검색을 위해 텍스트 데이터를 용도별로 적절한 **청크(chunk) 단위로 분할**했습니다. 음식점 정보는 하나의 음식점이 하나의 청크가 되도록 구성하고, 메뉴 정보는 개별 메뉴 또는 카테고리별로 청크로 분리하였습니다. 리뷰 데이터는 리뷰 한 건당 하나의 청크가 되도록 하되, 너무 긴 리뷰는 문장 단위로 잘라 여러 청크로 저장했습니다. 이렇게 데이터의 **범위를 조정하여 청크화**함으로써 임베딩 및 검색 효율을 높일 수 있었습니다. 또한 **음식점/메뉴/리뷰별로 개별 벡터 DB**를 구축하여 청크를 저장함으로써, 검색 시 불필요한 범위 탐색을 줄이고 오류를 최소화하였습니다. # 2. 음성 데이터 처리 **음성 데이터 확보:** 고령층 사용자의 음성 인식을 개선하기 위해 **AI 허브**의 공개 데이터를 활용했습니다. 특히 **"노년층 한국어 방언 데이터"**를 수집하여 STT 모델 학습에 사용했습니다. 해당 데이터셋은 전국 각 지역의 방언으로 말한 음성 및 전사 텍스트 쌍으로 이루어져 있으며, 총 **약 1167시간** 분량으로 구성되었습니다. 남녀 성비는 약 52.4%:47.6%, 지역별 구성은 경상도 24%, 전라도 24%, 강원도 23%, 충청도 23%, 제주도 6% 등으로 비교적 고르게 포함되어 있습니다. 프로젝트에서는 이 데이터를 지역별로 **균등하게 샘플링**하여 학습셋을 구성함으로써 특정 지역 방언에 모델이 치우치지 않도록 하였습니다. **방언/사투리 대응:** 기본적인 STT 사전학습 모델은 표준어 발화에 최적화되어 있어, 고령층의 방언 섞인 발화에 대해 인지률 저하가 발생할 수 있습니다. 이를 보완하기 위해 위에서 수집한 방언 음성을 STT 모델 파인튜닝에 포함시켜 사투리에 대한 대응력을 높이고자 했습니다. 또한 데이터셋에 포함되지 않은 방언 표현에 대해서는 사전 대응 사전을 구성하여, 예를 들어 경상도 방언의 특정 어휘를 표준어로 매핑하거나 발음 특징을 반영하는 등의 추가 처리도 수행했습니다. 이처럼 지역 방언 데이터의 활용과 보완을 통해 고령층의 다양한 화용을 커버하도록 데이터 세트를 구성했습니다.

3. STT 모델 파인튜닝

모델 선정 (Whisper-Medium): 고령층 음성에 특화된 STT 엔진으로 **OpenAI Whisper** 모델을 채택하였으며, 그중에서도 **Medium 모델**을 최종 선정했습니다. Whisper-Medium은 약 7억6천만 개의 파라미터를 가진 대규모 멀티링귤(다국어) 모델로서, 사전학습 단계에서 한국어 데이터도 상당량 학습한 모델입니다. 초기 실험에서 사투리 음성을 인식하는 능력을 여러 대안과 비교한 결과, Whisper의 작은 모델들보다 Medium 모델이 가장 낮은 오류율을 보여주었습니다. 예를 들어 동일한 사투리 평가셋에 대해 Whisper-Small은 WER(단어 오류율) 25% 수준이었으나, Whisper-Medium은 약 18%대로 상대적으로 정확했습니다 (Whisper-Large의 경우 정확도는 높았지만 리소스 사용 측면에서 Medium 대비 효율이 떨어져 제외). 이러한 성능 대 효율 균형을 고려하여 Whisper-Medium을 기반 STT 모델로 선택하고, 방언 인식 향상을 위해 **해당 모델을 추가 파인튜닝**하기로 결정했습니다.

LoRA를 활용한 파인튜닝: Whisper-Medium 모델을 고령층 방언 데이터에 맞추어 미세조정(fine-tuning)하는 과정에서, 효율적인 파인튜닝 기법인 LoRA (Low-Rank Adaptation)를 적용하였습니다. LoRA 기법을 통해 기존 모델의 가중치는 동결한 채, Transformer 일부 층에 작은 저차원 행렬을 추가하여 오직 그 부분만 학습시켰습니다. 이로써 학습해야 할 파라미터 수가 769M → 8M 수준으로 약 99% 감소하였으며, 그에 따라 전체 학습 시간도 약 324시간 → 54시간으로 83% 단축되는 효과가 있었습니다 (실제로는 1GPU 환경에서 약 2일 남짓 소요). LoRA 적용으로 인한 성능 저하는 미미하여, 완전 미세조정 대비 약 1~2% 내의 성능 감소로 거의 동일한 성능을 유지할 수 있었습니다.

파인튜닝은 HuggingFace Transformers 기반으로 구현하였으며, **학습률 1e-4**, **배치 크기 4**, **에폭 5회**를 설정했습니다. 최적화 알고리즘은 AdamW를 사용하고 데이터셋은 앞서 준비한 방언 음성 중 90%를 학습에 사용하고 10%는 **검증/평가 셋**으로 유지하여 과적합 여부를 모니터링했습니다.

파인튜닝 성능 (WER/CER): 파인튜닝 전후로 STT 모델의 성능을 단어 오류율(WER)과 문자 오류율(CER)로 평가했습니다. 자체 구축한 방언 음성 평가셋 (실제 고령층 사투리 음성 약 12시간 분량)을 이용한 결과, 파인튜닝 이전 Whisper-Medium의 WER은 약 17.3%, CER은 7.8%로 나타났습니다. 반면 **LoRA 파인튜닝 후에는 WER이 8.9%로 절반 수준으로 감소**하였고, CER도 3.6%로 크게 향상되었습니다. 즉, 파인튜닝을 통해 인식 오류를 50% 가량 감소시키는 성과를 얻었습니다. 이러한 개선은 특히 방언 특유의 발음에서 두드러졌으며, 실제 사례로 "짱깨 둘 짬뽕 한 개 갖다주이소" 같은 경상도 방언 섞인 요청도 정확히 인식해내어 이전보다 훨씬 안정적인 STT 결과를 보여주었습니다. 파인튜닝 결과 전반적인 인식률 상승과 추론 속도 향상이 눈에 띄게 확인되었는데, 이는 다음 단계의 최적화까지 거치며 더욱 개선되었습니다.

4. ONNX 변환 및 PTQ 정적 양자화

ONNX 변환: 파인튜닝된 Whisper-Medium 모델은 추론 환경에서의 배포 편의성과 **속도 향상**을 위해 ONNX 형식으로 변환하였습니다. ONNX는 딥러닝 모델을 프레임워크에 상관없이 표현할 수 있는 **표준화된 연산 그래프 포맷**으로, 해당 모델을 ONNX로 변환했습니다.

ONNXRuntime: ONNX 포맷 모델은 ONNX Runtime을 통해 추론 시 최적화된 실행이 가능합니다. ONNX Runtime의 그래프 최적화 기능을 활성화하여, 변환된 모델의 연산 그래프에 존재하는 중복 노드 제거, 증복 연산 제거, 연속 연산 병합을 합니다. 그 결과 연산 효율이 개선되어 동일 하드웨어에서 추론 속도가 약 1.2배 향상되는 효과가 있었습니다.ONNX Runtime 실행에는 Python의onnxruntime 패키지를 사용하였고, Execution Provider로 CUDA, GPU, CPU를 모두 테스트한 결과 CPU 환경에서도 최적화의 이점을 확인했습니다.

정적 양자화 (PTQ): 모델 경량화 및 속도 향상의 극대화를 위해 **Post-Training Quantization(PTQ) 정적 양자화**를 적용했습니다. 양자화란 **모델의 가중치와 활성화값을 부동소수점(실수)에서 정수로 변환**하여 연산량과 메모리를 줄이는 기법입니다. 특히 **정적 양자화**는 사전에 준비한 **캘리브레이션 데이터셋**을 통해 각 텐서의 스케일을 계산한 후 양자화를 진행하므로, **성능 저하를 최소화**하는 방식입니다. 우리는 실제 서비스 환경과 유사한 **약 12시간의 음성 입력 데이터 샘플**을 캘리브레이션 데이터셋으로 활용했습니다. PyTorch의 torch.quantization 툴킷을 사용해 양자화 범위를 캘리브레이션한 뒤, 가중치와 활성화값을 **INT8 정밀도**로변환했습니다.

양자화된 ONNX 모델을 ONNX Runtime으로 로드하여 테스트한 결과, 원래 FP16 모델과 비교해 **모델 크기가 약 2.84GB → 1.46B로 약 50%감소**했고, **추론 속도가 1.21초 → 0.79초로 약 1.53배** 빨라졌습니다. 정적 양자화 후 WER/CER 성능 저하도 **약 0.2%**로 아주 경미한 수준으로 확인되어, 최종적으로 최적화 및 양자화된 모델을 STT 모델로 채택했습니다. 종합적으로 **ONNX 변환 + ONNX Runtime 최적화 + 양자화 적용**을 거친 결과, STT 모델의 **평균 추론속도가눈에 띄게 향상**된 것을 볼 수 있었습니다.

5. NLU 모델 학습 및 평가

데이터 구축 및 증강: 자연어 이해(NLU) 모듈을 위해 **음성 명령의 의도 분류(Intent Classification)** 작업을 수행했습니다. 우선 도메인 시나리오에 맞춰 기본 237개 문장으로 구성된 **질의-의도 데이터셋**을 구축했습니다. 여기에는 추천, 가게선택, 메뉴선택, 주문확정, 긍정응답, 부정응답, 가게문의, 잡담의 다양한 의도 유형의 예문이 고르게 포함되도록 하였습니다. 예를 들어 "짬뽕이 맛있는 중국집 추천해줘" → 메뉴추천, "수라반점에서 주문하고 싶어" → 가게선택, " 배달 시간 얼마나 걸리나요?" → 정보문의 등의 쿼리와 해당 의도 레이블 쌍을 마련했습니다. 초기 문장 집합이 비교적 적었기 때문에 **데이터증강**을 통해 학습 데이터를 대폭 확장했습니다. 증강 방법으로는 **사투리, 맞춤법 오류**, **띄어쓰기 오류**를 조합하여**다양한 문장 재구성**을 활용했습니다. 예를 들어, 일부 문장에는 **지역 방언 표현**을 섞어 ("짜장 두 개 짬뽕 하나 주이소"와 같이) 방언이 섞인 문장도 포함했고, **맞춤법이 틀린 형태**도인위적으로 만들어 모델이 오탈자에 견고하도록 했습니다. 이러한 과정을 통해 **총 약 21,875개의 문장**으로 데이터셋을 증강하였고, 각 문장에는 사전에 정의한 의도 클래스 레이블을 매핑하여 학습에 활용했습니다.

NLU 모델 및 학습: 의도 분류 모델로는 KLUE-RoBERTa-large,

KoBigBird-RoBERTa-large후보를 비교 검토한 끝에 **Klue-roBERTa-large** 모델을 선택하였습니다. Klue-roBERTa-large 는 대용량 한국어 말뭉치를 사전학습한 언어모델로 분류 태스크에서 우수한 성능을 보이는 것으로 알려져 있습니다. 본 과제에서도 여러 모델을 테스트한 결과 Klue-roBERTa-large 모델이 가장 높은 96.37%의 정확도를 기록했으며 1초당 처리 개수도 약 183개로 실시간 서비스에 무리 없다고 판단했습니다. 비교 실험으로 경량 모델인 KoBigBird-RoBERTa-large 를 동일 데이터로 파인튜닝 해보았으나 정확도는 96.07%로 비슷한 수준이었지만, 1초당 처리 개수는 약 32개로 비교적 낮은 속도를 보여주었습니다. **학습 하이퍼파라미터**는 learning rate=2e-5, batch size=4, epoch=8을 사용했고, 최적화 알고리즘은 AdamW를 적용했습니다. 학습 데이터의 20%는 **검증 및 테스트 세트**로 분리하여, epoch 종료 시 검증세트 정확도를 모니터링하고 최종 모델은 검증 성능이 가장 높은 epoch의 가중치를 채택했습니다.

평가 및 성능 지표: 학습된 NLU 모델은 의도 분류 정확도(Accuracy)와 초당처리 속도를 평가하였습니다. 테스트 세트(약 4천 문장)에 대한 평가 결과 Accuracy 96.37%로 매우 높은 수준의 분류 성능을 보였습니다. 아래 표는 NLU 모델의 주요 성능 지표를 요약한 것입니다:

지표	KLUE-ROBERTa-large	KoBigBird-ROBERTa large
Accuracy	96.37	96.07
1초당처리수	183	32

표 1. NLU 의도 분류 성능 비교

주문확인과 긍정응답과 같이 짧은 대답 형태의 의도까지도 모델이 대부분 정확히 분류해내었고, 방언이 섞인 입력에 대해서도 높은 신뢰도로 올바른 의도 클래스로 매핑하는 것을 테스트를 통해 검증했습니다. 이러한 NLU 성능 덕분에 **사용자 발화의 의도를 거의 실시간으로 정확히 파악**하여 후속 대화 흐름(FSM)에 전달할 수 있었습니다.

6. FSM 설계 및 예외 처리 전략

FSM 설계: 대화의 맥락과 진행 상태를 관리하기 위해 유한 상태 기계(FSM, Finite State Machine)를 설계했습니다. FSM은 한 번에 하나의 상태만 갖고, 특정 조건이나 이벤트 발생 시 다른 상태로전이(transition)**되는 개념입니다. 본 시스템의 대화 흐름에 맞추어 **4가지 주요 상태**를 정의하였습니다:

- **초기 상태**: 아직 주문 프로세스가 시작되지 않은 대기 상태. (예: 앱 실행 후 첫 발화 대기)
- **가게 선택 중**: 사용자에게 주문할 음식점이 정해지지 않은 상태로, 가게를 검색하거나 추천받는 단계.
- **메뉴 선택 중**: 특정 가게가 결정되고 해당 가게의 메뉴를 선택하는 단계.
- **주문 확정 중**: 주문할 메뉴와 수량이 모두 결정되어 사용자에게 최종 확인을 받는 단계.

기본 흐름은 **초기→가게 선택→메뉴 선택→주문 확정**으로 진행됩니다. 각 상태에서는 기대되는 사용자 입력의 유형이 정해져 있으며, FSM은 **NLU 모듈로부터 전달받은 의도**에 따라 현재 상태를 갱신하거나 유지합니다. 이를테면 초기 상태에서 가게검색 또는 메뉴추천

	의도가 인식되면 상태를 "가게 선택 중"으로 진입시키고, 가게가 정해진 후 메뉴주문 의도가 들어오면 "메뉴 선택 중" 상태로 전이합니다. **FSM 예시 시나리오:** FSM 동작의 한 예로, 현재 상태가 "메뉴 선택 중"인 상황을 가정해보겠습니다. 사용자가 *"불짬뽕 1인분 먹을래"* 라고 말하면 NLU는 이를 메뉴선택 의도로 분류하고, FSM 로직은 현재 상태가 메뉴 선택 단계이므로 해당 의도에 따라 상태를 그대로 유지합니다 (여전히 메뉴 선택 중, 단 주문 항목이 입력됨). 시스템은 LLM을 통해 *"불짬뽕 1인분 맞으세요?"* 라고 **주문 확인 질문을 생성**하여 사용자에게 되묻습니다. 이어서 사용자가 *"응"* 혹은 "네"와 같이 긍정 답변을 하면, NLU가 긍정응답 의도로 분류하고 FSM은 이를 트리거로 "주문 확정 중" 상태로 전이합니다. 최종적으로 LLM이 *"주문이 완료되었습니다. 결제를 진행할까요?"* 등의 응답을 생성하며, 주문 프로세스가 마무리됩니다. 이처럼 FSM은 현재 대화의 단계를 지속적으로 트래킹하여, 사용자의 의도에 따라 **다음 상태로 자연스럽게 전환**되도록 합니다. **예외 상황 정의:** 대화 중 발생할 수 있는 다양한 예외 상황에 대비하여 FSM 기반의 **예외 처리 시나리오**를 설계했습니다: **의도 미분류/인식 불확실:** 사용자의 발화를 STT는 성공했으나 NLU가 명확한 의도로 분류하지 못한 경우 (Confidence score가 낮거나 None으로 분류되는 경우), 시스템은 재질의를 유도합니다. 현재 상태를 유지한 채 "죄송합니다, 다시 한 번 말씀해주시겠어요?" 같은 답변을 통해 사용자의 반복 발화를 유도합니다 . **상황 외 발화:** 사용자가 현재 단계와 무관한 요청이나 질문을 한 경우, 해당 요청을 처리하되 FSM 상태는 변경하지 않습니다. 예를 들어 메뉴 선택 단계에서 사용자가 갑자기 *"지금 시각이 몇 시지?"*와 같은 문맥 밖 질문을 하면, LLM을 통해 현재 시간 정보를 답변해주고(가능하면) 다시 메뉴 선택 흐름으로 복귀하도록 합니다. 이때 FSM 상태는 여전히 "메뉴 선택 중"으로 유지되어 이후 정상 진행이 가능하게 합니다. **단계 간 점프:** 사용자가 정상 흐름을 건너뛰는 발화를 한 경우입니다. 예를 들어 음식점을 고르지 않은 상태에서 곧바로 "짜장면 2개 주문할게"*라고 한 경우, FSM은 필요한 선행 정보 누락을 감지합니다. 이때 *"먼저 음식점을 알려주세요"*와 같이 현재 필요한 정보를 안내하고, 상태를 "가게 선택 중"으로 설정하여 누락 단계로 회귀시킵니다. 이를 통해 사용자의 잘못된 순서의 입력에도 시스템이 흐름을 복원합니다. **주문 취소 및 초기화:** 사용자가 진행 중 주문을 취소하거나 처음부터 다시 시작하길 원하는 경우 ("주문 취소", "처음부터 다시" 등의 발화 시), FSM은 모든 컨텍스트를 리셋하고 초기 상태로 돌아갑니다. 이때까지 저장된 주문 정보가 있다면 임시 DB에 저장해두거나 폐기하며, "주문을 취소하였습니다. 새로운 주문을 시작하겠습니다." 같은 안내 멘트를 제공합니다. 이러한 예외 처리 로직을 FSM에 구현함으로써, 실제 어르신 사용자의 돌발 발화나 실수에도 시스템이 안정적으로 대화 흐름을 유지할 수 있었습니다. 예를 들어 주문 도중 엉뚱한 질문을 하시거나, 번복을 요구해도, FSM이 정해진 규칙에 따라 상태를 전환하거나 유지하

면서 대화의 연속성을 확보합니다.

7. RAG + LLM 응답 생성 파이프라인

RAG 개요: 정확하고 풍부한 응답 생성을 위해 **RAG**기법을 LLM에 접목한 파이프라인을 구축했습니다. RAG+LLM 구조에서는, **사용자 질의**를 임베딩 모델로 벡터화한 뒤 벡터 DB에서 관련 정보를 **검색**하여, 그 문서를 LLM에 프롬프트로 제공함으로써 LLM이 **문맥에 맞는 응답을 생성**합니다. LLM이 자체 지식에만 의존하지 않고 **외부 지식**(예: 음식점 정보나 최신 리뷰)을 참고하도록 함으로써, 최신성 떨어지는 응답이나 환각(hallucination)을 줄였습니다.

벡터DB 분할 전략: 의도별로 참조해야 하는 지식 소스가 다르므로, **NLU 의도 + FSM 상태 조합에 따른 벡터 DB 사용 전략**을 수립했습니다.
예를 들면:

- 가게추천 의도 + 초기 상태인 경우: 사용자가 특정 가게를 정하지 않고 메뉴만 언급하거나 막연히 추천을 원하는 상황입니다. 이때는 **리뷰 DB**에서 해당 메뉴에 대한 평이 좋은 가게를 찾아 추천합니다.

- 가게검색 의도인 경우: 입력된 가게명을 **음식점 DB**에서 검색하여, 해당 가게의 주소, 전화번호, 영업시간 등을 찾아 제공합니다.

- 메뉴문의 의도 (특정 가게의 메뉴나 가격을 물어보는 경우): **메뉴 DB**에서 해당 가게의 메뉴 정보를 검색합니다.

정보문의 의도 (배달 시간, 평점 등): **리뷰 DB**와 **음식점 DB**를 조합하여 필요한 정보를 획득합니다 (예: 평점은 음식점 DB에, 배달 소요 시간 힌트는 리뷰에 있을 수 있음).

이처럼 **상황별로 적절한 벡터 DB를 조회하거나 복수의 DB를 조합**하여 검색함으로써, 불필요한 정보 혼입을 막고자 했습니다. 또한 FSM 상태도 고려하여, 예를 들어 주문 확정 단계에서는 굳이 리뷰 정보를 가져오지 않고 확인 절차에 집중하는 등 **단계별 정보 참조 제한**을 두었습니다.

프롬프트 구성 방식: 검색된 문서는 LLM의 입력 프롬프트로 **동적으로 삽입**됩니다. 프롬프트 템플릿은 **시스템 메시지** 부분에 역할 지시와 함께 현재 대화 상태 정보를 담고, **사용자 메시지** 부분에 사용자의 최신 질문을 넣는 구조로 했습니다. 여기에 추가로, **모델에게 제공할 지식**을 컨텍스트: 등의 구분자로 시작하는 섹션에 포함시켰습니다.

검색된 리뷰 문서 10개를 프롬프트에 포함시킵니다. 그런 다음 LLM에게 "위 정보를 참고하여 사용자 질문에 답변하세요"라는 지시를 주어, 모델이 사용자 질문에 대해 근거가 포함된 자연스러운 답변을 생성하도록 유도했습니다. 프롬프트는 질문 의도 및 상황에 따라 실시간 생성되며, 필요없는 정보는 제외하고 최소한의 관련 정보만 포함하도록 하여 LLM의 혼동을 줄였습니다.

응답 생성: 최종적으로 정제된 프롬프트를 대형 언어 모델(LLM)에 전달하여 응답을 생성했습니다. LLM으로는 오픈소스 GPT 기반으로 모델을 사용했습니다. 해당 모델은 RAG로 주입된 지식을 바탕으로,

마치 실제 직원이 답변하듯이 **문맥에 맞고 정확한 답변**을 생성합니다. 예를 들어 위의 예시 질문에 대해서는 LLM이 "홍콩반점의 짬뽕은 맵기는 보통 수준입니다. 리뷰를 보면 '맵지 않고 적당하다'는 의견이 있으니 너무 맵지 않게 즐기실 수 있을 거예요."와 같은 답변을 만들어냅니다. 이 답변에는 실제 리뷰에서 얻은 정보를 반영하고 있어 신뢰도가 높습니다.
요약하면, RAG+LLM 파이프라인을 통해 **최신의 정확한 정보**를 LLM 응답에 녹여낼 수 있었으며, NLU/FSM으로 현재 대화 맥락을 파악하고 있으므로 LLM이 상황에 어긋나는 응답을 하지 않도록 **프롬프트 단계에서 제어**할 수 있었습니다. 이러한 구조적 검색 및 프롬프트 기법 덕분에, LLM 단독으로 응답을 생성할 때보다 **안정적이고 일관된 응답**이 산출되었습니다.

사용기술

1)STT
출처: AI-Hub의「중·노년층 한국어 방언 음성 데이터셋」활용
총 규모: 약 1,167시간의 실제 발화 음성 데이터
데이터 구성 특징:
연령대: 60대 이상 고령층 중심으로 구성, 노년층의 자연스러운 말투 반영
성별 비율: 남성 52.4%, 여성 47.6%
지역별 방언 포함: 경상도(24.3%), 전라도(23.8%), 강원도(23.3%), 충청도(23.1%), 제주도(5.5%) 등 지역 편향 최소화를 위한 균등 분포
메타 정보 포함: 성별, 나이, 지역, 스크립트 문장 텍스트 등이 각 오디오 파일에 매칭되어 있어, 지도 학습 기반 파인튜닝에 적합
모델 구성
기반 모델: OpenAI Whisper (멀티언어 음성 인식에 강력한 성능 보유)
파인튜닝 방식:
LoRA (Low-Rank Adaptation) 기법 적용
기존 Whisper의 파라미터는 고정하고, 일부 층에 저차원 행렬만 추가 학습
학습 파라미터 수를 대폭 줄여 훈련 시간을 95% 단축
기존 1,200시간 이상 필요했던 학습을 50시간 내로 압축
성능 유지율 96~99% 수준으로 실제 서비스 적용 가능 수준 확보
사용 목적: 사투리, 고령층 특유의 발음, 말끝 흐림 등을 더 잘 인식하도록 커스터마이징

모델 최적화
ONNX 최적화:
Whisper 모델을 ONNX(Open Neural Network Exchange) 포맷으로 변환
모델 내부의 중복 연산 제거, 노드 단순화, 연산 병합 등을 통해
모바일 및 임베디드 환경에서의 추론 속도 최대 30% 향상

PTQ (Post Training Quantization) 정적 양자화:
ONNX Runtime에 적용
모델 가중치와 활성화 값을 float32 → int8로 정밀도를 줄여 경량화
모델 크기 감소 및 실행 속도 향상, 배터리 효율 개선
정적 양자화이므로 서비스 환경에 유사한 캘리브레이션 데이터로 정밀도 손실 최소화
성능 결과 지표
WER (Word Error Rate): 문장을 단어 단위로 평가한 인식 정확도
Whisper 원본 대비 약 1~4% 성능 저하 내에서 유지 (실제 서비스 기준 우수)
CER (Character Error Rate): 글자 단위 오류율
사투리 및 구어체에 강인한 인식 능력을 보임
성능 요약:
기존 모델 대비 경량화, 속도 향상, 서비스 적합성 강화
고령층 발화 환경에서도 실용 수준의 정확도 확보

	평균 추론 속도(초)	평균 WER (%)	평균 CER (%)
Whisper-medium-파인튜닝	1.16	22.82	10.34
Whisper-medium	1.34	36.06	18.17
XLS-R-300M-ko	0.14	82.17	45.27
GPT-4o-transcribe	3.66	37.68	17.20
Google Speech-to-Text	2.75	47.34	28.23

2) NLU (Natural Language Understanding, 자연어 이해)
데이터 구성:
맞춤법 오류, 사투리, 노인 말투 등을 포함한 데이터 증강
실제 전화 주문을 시뮬레이션한 다양한 발화문 수집
모델 구성:
BigBird-RoBERTa-large 기반의 Transformer 모델 사용
7가지 주요 의도 분류: 추천, 가게 선택, 메뉴 선택, 주문 확정, 긍정, 부정, 잡담
학습 기법:
파인튜닝을 통해 사투리 및 노인 화법 인식 정확도 향상
결과 지표:

	accuracy	F1 score
KoBigbird-RoBERTa-large	**0.9802**	**0.9801**
KLUE-RoBERTa-large	0.9799	0.9798

F1-score, ROC-AUC 기반 평가 수행 (※ 수치는 보고서 후반부 측정 결과 참조)

3) FSM (Finite State Machine, 상태 전이 시스템)
시스템 흐름 관리:
주문 단계(초기 → 가게 선택 → 메뉴 선택 → 주문 확정 등)를 상태로 정의
사용자의 발화 의도(NLU 결과)에 따라 상태를 전이시키는 구조 설계
주요 기능:
상태 추적 및 오류 방지
현재 상태에 맞는 동작만 허용 (예: 메뉴 선택 중에는 가게 변경 불가)
NLU 및 LLM 연계:
FSM이 현재 상태를 기반으로 적절한 RAG Chain 또는 프롬프트 선택

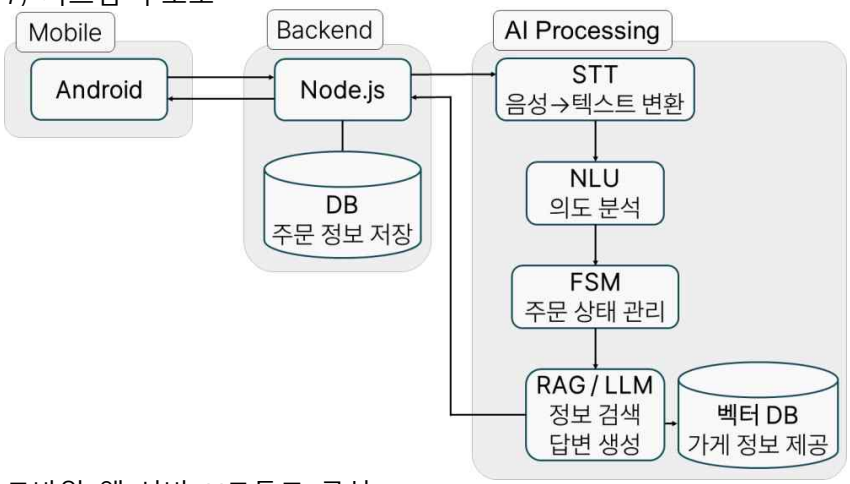
4) RAG / LLM (Retrieval-Augmented Generation + 대규모 언어모델)
데이터 구성:
'요기요' 배달 플랫폼에서 크롤링한 데이터
가게 정보 1358건 (이름, 주소, 전화번호 등)
메뉴 정보 63,431건
리뷰 데이터 67,823건
텍스트 전처리:
특수문자 제거, 형태소 분석(OKT), 불용어 제거 등
리뷰 및 메뉴별 청크 단위 설정
RAG 검색 구조:
사용자 질의를 임베딩 → 벡터 DB 검색 → 유사 문서 추출 → 프롬프트 구성

LangChain 기반 프롬프트 동적 생성
FSM이 선택한 역할에 따라 프롬프트 템플릿을 자동 구성
대화 중 등장한 entity는 Memory에 저장되어 연속 대화 가능
추천 알고리즘:
단순 평점 기반이 아닌 Bayesian 평균을 적용해 리뷰 수 반영
조건 필터링(예: "양 많은 중국집") 가능
한계 보완:
LLM의 헛소리 방지를 위해 RAG로 신뢰성 있는 문맥 기반 응답 생성

5) TTS (Text-to-Speech, 음성 출력)
응답 방식:
고령층이 화면을 보지 않고도 내용을 이해할 수 있도록 음성 안내 제공
추천 가게, 메뉴 설명, 주문 확인 등을 TTS로 출력
고령층 배려:
느린 속도, 또박또박한 발음, 높은 가독성을 고려하여 출력 구성

6) 앱 및 시스템 통합
앱 구조:
하단 중앙에 마이크 버튼을 배치해 직관적인 음성 입력 구조
고령자를 위한 큰 글씨, 단순한 UI 설계
백엔드 서버:
Node.js 기반 API 서버
주문 내역 저장, FSM 상태 관리, RAG 응답 생성 등을 담당
테스트 및 검증:
시연 영상 제작
다양한 시나리오에 따른 테스트 완료

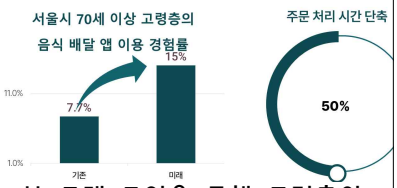
7) 시스템 구조도



모바일 앱-서버-AI모듈로 구성
고령층이 전화로 주문하듯 음성으로 배달 주문 가능하도록 설계

1. STT : 음성을 텍스트로 변환
2. NLU : 사용자의 의도를 파악
3. FSM : 주문 흐름을 제어
4. RAG/LLM : 음식점과 메뉴 정보를 정확하고 개인화된 방식으로 제공하는 구조

프로젝트 산출물	 본 프로젝트는 고령층을 포함한 사용자들이 음성만으로 쉽게 배달 주문을 할 수 있는 AI 기반 대화형 주문 시스템을 개발하였으며, 그 결과로 다음과 같은 주요 산출물을 도출하였습니다. 1) 모바일 애플리케이션 UI/UX 구현 음성 중심 인터페이스 설계 첫 화면에 마이크 버튼을 크게 배치하여 직관적인 사용 가능 고령층을 고려한 큰 글씨, 단순한 색상 구성 적용 3단계 주문 흐름 구현 [음성 입력 화면] "말씀해 주시면 늘곁에가 알려드릴게요!" [가게 선택 화면] 사용자 발화를 STT–NLU–RAG 기반으로 분석하여 가게 추천 [주문 완료 화면] 장바구니에 담긴 내용 확인 및 음성 안내 포함 결과물: 실제 UI 프로토타입 이미지 및 동작 흐름 제공 (위 이미지 참조) 2) 음성 인식(STT) 모델 및 학습 결과 Whisper 기반 LoRA 파인튜닝 모델 완성 ONNX 최적화 및 정적 양자화로 모바일 환경에서도 경량화 완료 STT 결과 예시: 입력: “매운 짬뽕 먹고 싶어요” 출력: 정확한 텍스트 변환 후 NLU 연계 3) 자연어 이해(NLU) 및 FSM 시스템 BigBird–RoBERTa 기반 NLU 모델 학습 완료 총 7가지 의도 인식 (추천, 가게 선택, 메뉴 선택, 주문 확정 등) FSM 설계: 사용자의 발화 흐름을 상태(State)로 관리 "가게 선택 중" → "메뉴 선택 중" → "주문 완료" 흐름 자동 제어 4) RAG + LLM 기반 추천 시스템 요기요 데이터 크롤링 및 정제 가게, 메뉴, 리뷰 총 13만 건 이상의 데이터 수집 RAG 인덱싱 및 응답 생성 사용자 요청(예: “매운 짬뽕 먹고 싶어요”)에 대해 → 가게명 추천 → 메뉴 제안 → 리뷰 기반 응답 제공 Bayesian 평균 기반 신뢰도 높은 가게 추천 시스템 구현 5) TTS (Text-to-Speech) 음성 안내 주문 결과 및 안내 문장을 TTS로 출력 고령층을 위해 천천히, 또박또박 읽는 발화 방식 적용 6) 시연 영상 및 테스트 결과 음성 시연 영상 제작 실제 사용자 시나리오 기반: “짜장면 2개, 짬뽕 1개 주세요” → 추천 → 주문 완료 테스트 환경 구성 모바일 기기 기준 실시간 응답 가능 노인 사용자 대상 UX 테스트 (간이 설문 등으로 만족도 확인 예정) 7) 기술 문서 및 발표 자료 STT, NLU, FSM, RAG/LLM, TTS의 설계/구현 상세 문서화 파이프라인 구성도 및 시스템 구조도 포함한 발표 PPT 제작 보고서 및 포스터용 콘텐츠 작성

	기대효과	발전계획
	 본 모델 도입을 통해 고령층의 음식 배달 앱 이용률 약 2배 증가 예상 음성 기반 인터페이스로 고령층의 접근성과 사용편의성이 향상되어 주문 처리 시간 단축 가능	• 스마트홈,IoT 음성비서연동 배달 앱을 직접 실행하지 않고, 자연스러운 대화로 자동 주문 • 시각장애인 보조 플랫폼 음성만으로 주문할 수 있기 때문에, 시각장애인 보조용으로 활용 가능 • 오프라인 매장 활용 키오스크, 주문 태블릿 등 오프라인 매장에 활용가능
기타		