2024 전기 졸업과제 중간 보고서

팀주제 : 대규모 언어 모델(LLM)의 Prompt Engineering을 통한
한국어-한국수어 번역 서비스 개발

팀명 : 수어비전
팀 번호 : 4
팀원 : 허재성, 김민혁, 한지석

목차

1. 과제 선정 배경 및 목표

국립국어원(원장 송철의)의 '한국수어 사용 실태 조사'의 결과를 따르면, 일상적인 의사소통에서 가장 많이 사용하는 언어는 '수어'라고 응답한 농인은 69.3%로 조사되었다. 농인의 제1언어가 '수어'임을 말해주는 결과이다. 하지만 가족과의 의사소통에서는 수어 사용 비율(42.7%)이 다소 낮게 나타났는데, 이는 가족 구성원 모두가 수어에 능숙한 것은 아니기 때문인 것으로 보인다. 농인들이 수어를 일상적으로 사용하지만, 비장애인이 수어를 이해하지 못하면 소통이 어렵다는운 것이 현실이다.

"부산광역시 농아인 협회"의 자문 결과에 따르면, 비장애인과의 소통 어려움으로 가장 불편한 부분은 낯선 지역에서의 길 찾기와 공공기관 이용입니다. 예를 들어, 버스를 이용할 때 버스 기사와의 의사 소통이 어려워 목적지에 대한 정보를 얻는 것조차 어렵고, 관공서에서의 업무 처리가 소통 문제로 인해 지연되는 경우가 흔히 발생합니다.

AI Hub에서 제공하는 '수어영상' 데이터를 기반으로 'Sign Language Recognition'을 위해 'CNN' 기반 모델인 'SlowFastSign'을 학습을 한 후, 청각장애인의 수어를 카메라로 인식하고 이를 'Prompt Engineering'을 통해 텍스트로 번역하는 어플을 제공해 비장애인과 소통을 수월하게 하여 비장애인과 장애인의 소통문제를 완화하는 데에 기여하고자 한다.

2. 요구사항 및 제약사항 분석에 대한 수정사항

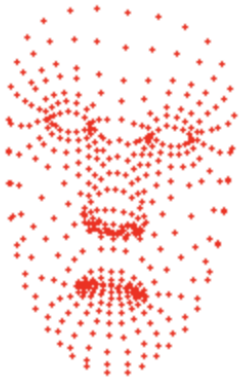
과제를 진행하는 중 예상하지 못한 제약사항들을 마주했다. 빠른 원격 회의와 연구실 조언을 통해 문제를 효과적으로 해결할 수 있었다.

2.1. KeyPoint 추출

2.1.1. OpenPose를 통한 KeyPoint 추출

OpenPose를 이용해 face, hand, body의 KeyPoint를 동시에 추출하는 것과 라이브 웹캠으로 KeyPoint를 추출하는 것은 처리 속도에서 한계가 있어 효율적이고 경량화된 모델인 MediaPipe를 사용했다.

2.1.2. MediaPipe를 통한 KeyPoint 추출 후 매핑



우리가 사용한 AI Hub에서 제공하는 '수어영상' 데이터는 OpenPose를 이용해 KeyPoint를 추출한 데이터이다. OpenPose와 MediaPipe의 hand, body KeyPoint는 유사하지만, face KeyPoint는 큰 차이가 있어 매핑 과정이 필요했다.

2.2. 수어 영상을 이용한 AI 모델 학습

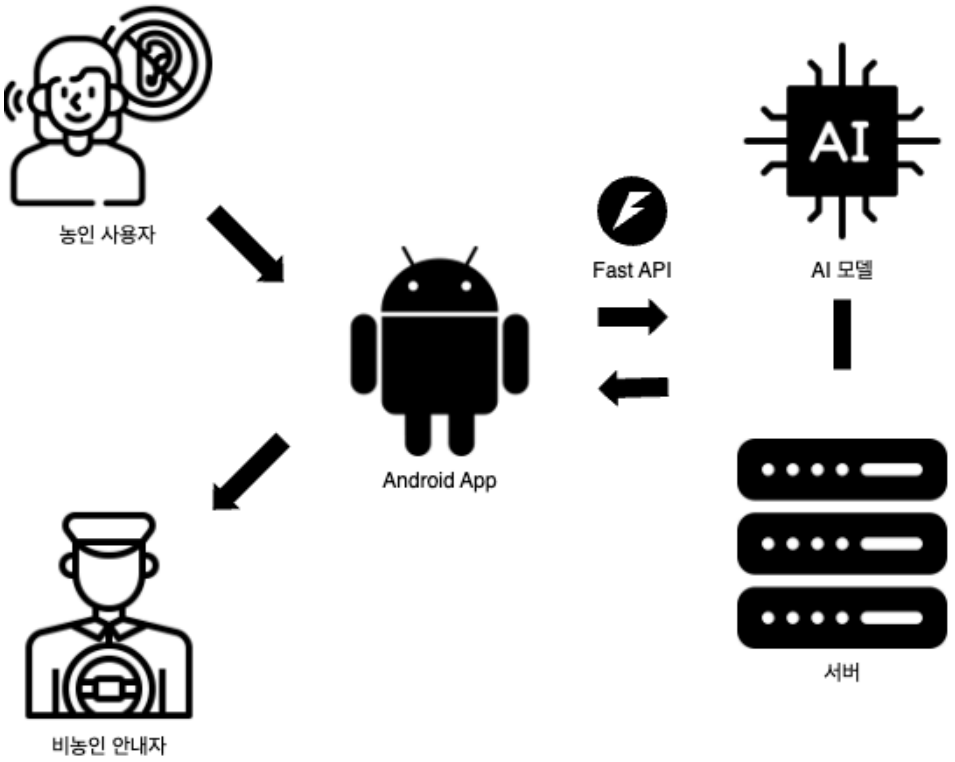
AI Hub에서 제공하는 방대한 양의 수어 데이터를 사용하여, 수어 인식을 위한 CNN(Convolutional Neural Network) 기반 모델인 SlowFastSign을 학습시켰습니다. 이 모델은 농인이 사용하는 수어 영상을 인식하고, 이를 텍스트로 변환하는 기능을 제공합니다. SlowFastSign 모델은 수어의 세부적인 움직임과 패턴을 잘 파악할 수 있도록 설계되었으며, 특히 비슷한 동작의 수어를 구분할 수 있도록 높은 정확도를 목표로 하였습니다.

2.3. 어플리케이션 서비스 개발

수어 인식 시스템의 핵심은 FastAPI를 이용한 백엔드 서버 개발이었습니다. 이 서버는 안드로이드 애플리케이션과 상호작용하여, 수어 영상을 처리하고 AI 모델을 통해 인식 결과를 제공합니다. 어플리케이션은 촬영된 수어 영상을 서버로 전송하고, 서버는 이를 처리하여 텍스트로 변환한 후 클라이언트에게 반환합니다. 이 API는 사용자에게 실시간으로 빠르고 정확한 수어 번역 결과를 제공하기 위해 설계되었습니다.

3. 설계 상세화 및 변경 내역

3.1. 전체 설계 구조



1. 영상 소스 획득
   Android 앱에서 농인 사용자가 수어 영상을 촬영합니다.
   촬영된 영상은 클라이언트 측에서 전처리 후 서버로 전송됩니다.

2. 서버 연결 및 영상 전송
   Android 앱은 FastAPI 백엔드와 연결되어, 수어 영상을 HTTP 요청을 통해 서버로 전송합니다.

3. AI 모델 실행
   서버는 수신한 영상을 전처리하여 AI 모델에 입력합니다.
   AI 모델은 영상의 특징을 분석하여, 데이터베이스에서 가장 유사한 3개의 영상 파일명을 찾습니다.

4. 유사도 분석 및 결과 반환
   서버는 유사도가 가장 높은 3개의 영상 파일명을 클라이언트에 반환합니다.

5. 사용자 선택 및 후처리
   농인 사용자는 반환된 3개의 영상 중 가장 적절한 데이터셋을 선택합니다.
   비농인 사용자는 선택된 데이터셋을 기반으로 텍스트 정보를 확인하고, 결과를 제공받습니다.

3.2. AI 모델 개발

3.2.1. 데이터 준비

AI Hub에서 제공한 536,000개의 수어 영상 데이터를 사용해 학습 데이터를 구성했습니다. 데이터셋은 수어 영상 클립, 형태소 및 비수지 요소 가공값(json 파일), 그리고 30fps 분할 이미지의 KeyPoint 값을 포함하고 있습니다.

3.2.2. 모델 학습

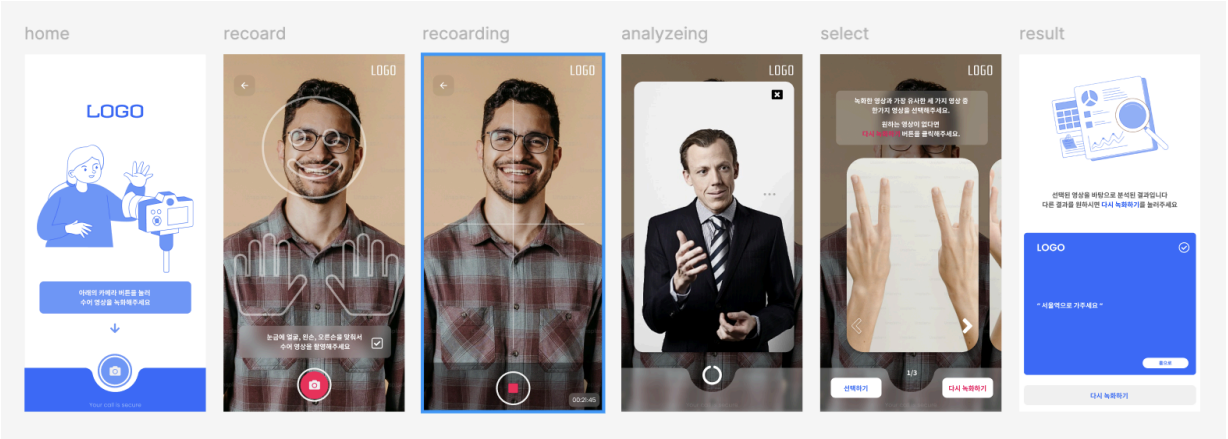
SlowFastSign 모델은 수어 영상 내의 복잡한 움직임을 인식하고, 이를 텍스트로 변환할 수 있도록 설계되었습니다. 모델은 수어의 공간적, 시간적 특징을 동시에 학습하여 높은 정확도의 인식 결과를 제공합니다.

3.2.3. 모델 평가 및 개선

학습된 모델은 검증 데이터셋을 이용해 평가되었으며, 정확도를 높이기 위해 지속적인 하이퍼파라미터 튜닝과 모델 아키텍쳐 개선이 이루어졌습니다.

3.3. Android 애플리케이션 제작

3.3.1. 전체 디자인



3.3.2. 시나리오

본 애플리케이션은 농인 사용자로부터 수어 영상을 녹화받아, 백엔드에서 AI 모델을 통해 분석하고, 결과를 사용자에게 제공하는 시스템을 제공합니다. 전체 시나리오는 다음과 같습니다.

1. 영상 녹화:

농인 사용자는 Android 애플리케이션을 통해 수어 영상을 녹화합니다.

2. 백엔드 통신:
   녹화된 영상은 FastAPI 백엔드 서버로 전송됩니다. 서버는 이 영상을 전처리하여 AI 모델을 실행하고, 데이터셋으로부터 가장 유사한 3개의 수어 영상을 찾습니다.

3. 유사한 영상 제공:
   백엔드 서버는 유사한 3개의 수어 영상 파일명을 반환합니다.
   애플리케이션에서는 해당 영상들을 재생합니다.

4. 영상 선택:
   농인 사용자는 제공된 3개의 영상 중에서 가장 적절하다고 생각되는 영상을 선택합니다.

5. 텍스트 디스플레이:
   선택된 영상의 텍스트 정보를 백엔드에서 받아와서 비농인 안내자에게 보여줍니다.

3.3.3 아키텍처

본 애플리케이션은 MVVM (Model-View-ViewModel) 패턴을 적용하여 기능적이고 확장성 있는 구조를 제공합니다. 애플리케이션은 총 4개의 주요 액티비티로 구성되어 있으며, 각각의 액티비티와 그에 대응하는 ViewModel, 그리고 패키지 구조는 다음과 같습니다.

1. MVVM 패턴 적용

Model: 애플리케이션의 데이터와 비즈니스 로직을 포함합니다. 모델은 데이터 클래스와 API 호출을 처리하는 로직을 담당합니다.
View: 사용자 인터페이스(UI)를 정의하며, 액티비티와 프래그먼트가 포함됩니다. View는 UI를 구성하고 사용자와의 상호작용을 처리합니다.
ViewModel: UI 관련 데이터와 로직을 관리하며, View와 Model 간의 연결을 제공합니다. ViewModel은 데이터 처리 및 비즈니스 로직을 View에 제공하며, UI 상태를 관리합니다.

2. 액티비티 및 ViewModel

HomeActivity:
홈 화면으로, 설명용 수어 영상과 녹화 버튼을 포함합니다. 녹화 버튼을 누르면 RecordActivity로 이동합니다.

RecordActivity:

녹화 화면으로, self-calibration과 촬영 방법에 대한 간단한 설명용 수어 영상과 확인 버튼을 포함한 dialog를 포함합니다. dialog 확인 후 사용자는 눈금에 맞춰 얼굴, 왼손, 오른손을 위치시킨 뒤 녹화 버튼을 눌러 안내자에게 전달하고자 하는 수어를 녹화합니다.

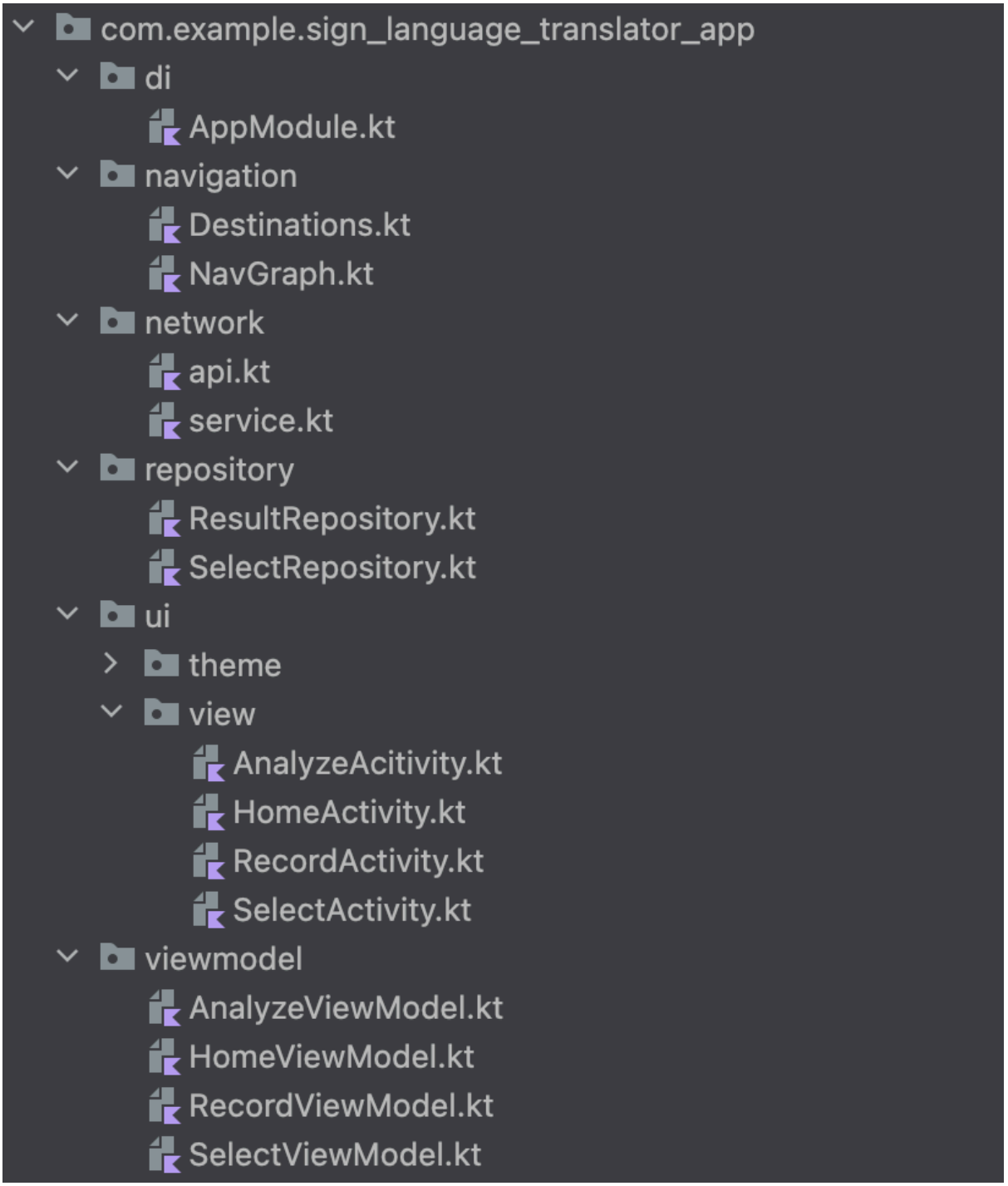
AnalyzeActivity:
백엔드 서버에서 AI 모델을 실행하는 동안 대기하는 분석 로딩 화면입니다. 곧 나타날 3개의 영상 중 가장 적절한 영상을 선택하라는 내용의 설명용 수어 영상을 재생합니다.

SelectActivity:
백엔드로부터 받아 온 영상 데이터들을 보여주는 화면입니다. 사용자는 horizontal Pager를 통해 영상을 확인하고, 자신의 의도와 가장 잘 맞는 수어 영상을 선택합니다. 영상을 선택할 시 ResultActivity로 전환되고, 다시하기 버튼을 클릭할 시 RecordActivity로 전환됩니다.

ResultAcitivty:
백엔드로부터 농인 사용자가 선택한 영상 데이터의 설명을 받아와서 화면에 보여줍니다. 농인 사용자는 비농인 안내자에게 결과 텍스트를 보여줌으로써 의사를 전달할 수 있습니다.

3. 패키지 설명

- di (Dependency Injection): 의존성 주입을 통해 앱의 구성 요소를 관리합니다.
- navigation: 앱 내 화면 간 내비게이션 관련 파일들을 관리합니다.
- network: 백엔드와의 통신과 관련된 파일들을 관리합니다.
- view: 유저에게 보여지는 Activity들을 관리합니다.
- viewmodel: ui와 repository 사이의 로직을 관리합니다.
- repostiory: 백엔드로부터 받아온 데이터들을 관리합니다.



3.3.4. FastAPI를 이용한 백엔드 구현

안드로이드 애플리케이션을 통해 촬영된 수어 영상 파일을 FastAPI 백엔드 서버로 전송합니다. 서버는 이 영상을 받아 AI 모델을 통해 분석하고, 분석 결과를 애플리케이션에 반환합니다

1. 영상 파일 업로드: 안드로이드 애플리케이션은 사용자가 촬영한 수어 영상 파일을 FastAPI 서버로 전송합니다. 이 과정은 HTTP POST 요청을 통해 이루어지며, 파일은 Multipart 형식으로 전송됩니다.

2. 서버에서 영상 처리: FastAPI 서버는 요청에서 영상 파일을 수신하고, 서버의 지정된 위치에 파일을 저장합니다. 이후 저장된 파일을 AI 모델에 입력하여 유사도 분석을 수행합니다.

3. 유사도 분석: AI 모델은 입력된 영상 파일을 분석하여 가장 유사한 영상 목록을 생성합니다. 이 과정에서 모델은 미리 학습된 데이터셋을 기반으로 유사도를 평가하고, 결과를 도출합니다.

4. 결과 반환: 분석이 완료되면, FastAPI 서버는 유사한 영상 목록을 포함한 응답을 JSON 형식으로 안드로이드 애플리케이션에 반환합니다. 애플리케이션은 이 응답을 기반으로 사용자에게 유사한 영상 목록을 표시합니다.

4. 갱신된 과제 추진 계획

4.1. KeyPoint 추출의 최적화

OpenPose와 MediaPipe의 성능을 비교하여 최적의 성능을 발휘할 수 있는 도구를 선택하였습니다. MediaPipe의 경량화된 모델을 사용하여 KeyPoint 추출의 속도를 개선했습니다.

4.2. AI 모델의 성능 향상

SlowFastSign 모델의 정확도를 높이기 위해 추가적인 학습과 검증을 진행하였으며, 하이퍼파라미터 튜닝을 통해 성능을 최적화했습니다.

5. 구성원별 진척도

김민혁(안드로이드 애플리케이션 개발)

- 앱 시나리오 및 디자인 구상 완료
- 앱 프로젝트 구성 및 GitHub 연동 완료
- 앱 아키텍처 설정 완료
- 백엔드 구현 관련 리서치 완료

허재성(AI 모델 개발)

- SlowFastSign 모델 초기 학습 및 검증 완료
- 모델 성능 개선을 위한 하이퍼파라미터 튜닝 진행 중
- AI 모델 결과의 정확도 평가 및 리포트 작성

한지석(AI 모델 개발)

- MediaPipe를 이용한 수어 keypoint 추출 및 OpenPose Keypoint로 매핑
- 데이터 분석 및 전처리
- SlowFastSign 모델 초기 학습 및 검증 완료

6. 과제 수행 내용 및 중간 결과

6.1. 데이터 분석

AI Hub에서 제공하는 '수어영상' 데이터는 536,000개의 수어영상 클립(.mp4 파일), 536,000개의 수어 영상에 대한 형태소 및 비수지 요소 가공값(json 파일), 536,000개의 수어 영상에 대한 30fps 분할 이미지에 대한 keypoint 값(json 파일)로 구성되어 있다.

6.2. 데이터 전처리

데이터 전처리 과정에서 수어 영상의 KeyPoint를 정확하게 추출하는 것이 중요했습니다. OpenPose와 MediaPipe를 비교한 결과, MediaPipe를 사용하여 처리 속도와 효율성을 높였습니다. 전처리된 데이터는 AI 모델의 학습에 사용되었으며, 데이터의 품질을 유지하기 위해 노이즈 제거와 정규화 작업도 수행하였습니다.

6.3. keypoint 추출

사용자(농인)의 수어를 실시간으로 텍스트로 번역하기 위해서 우선 keypoint를 추출해야 한다. keypoint를 추출하기 위해 AI Hub의 '수어영상' 데이터를 만들 때 사용한 OpenPose를 사용하려고 했으나, 데이터 처리 속도 문제로 인해서 경량화된 모델인 MediaPipe를 이용해 keypoint를 추출한다.

6.4. Sign Language Recognition

SlowFastSign 모델을 통해 실시간으로 수어를 인식하고 텍스트로 변환하는 과정을 구현했습니다. 모델의 성능은 학습 데이터의 품질과 양에 크게 의존하며, 이를 개선하기 위해 다양한 수어 영상을 사용하여 학습을 진행했습니다. 또한, 모델의 정확도를 높이기 위해 지속적인 검증과 튜닝 작업을 진행했습니다.