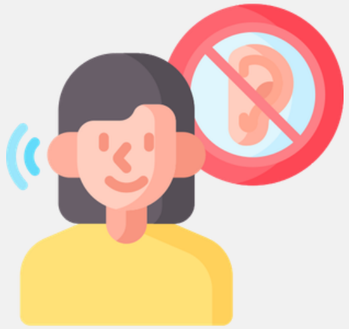
실시간 수어 번역 오픈소스 프로그램 개발

문채일 · 정민서 · 최은소

CONTENT

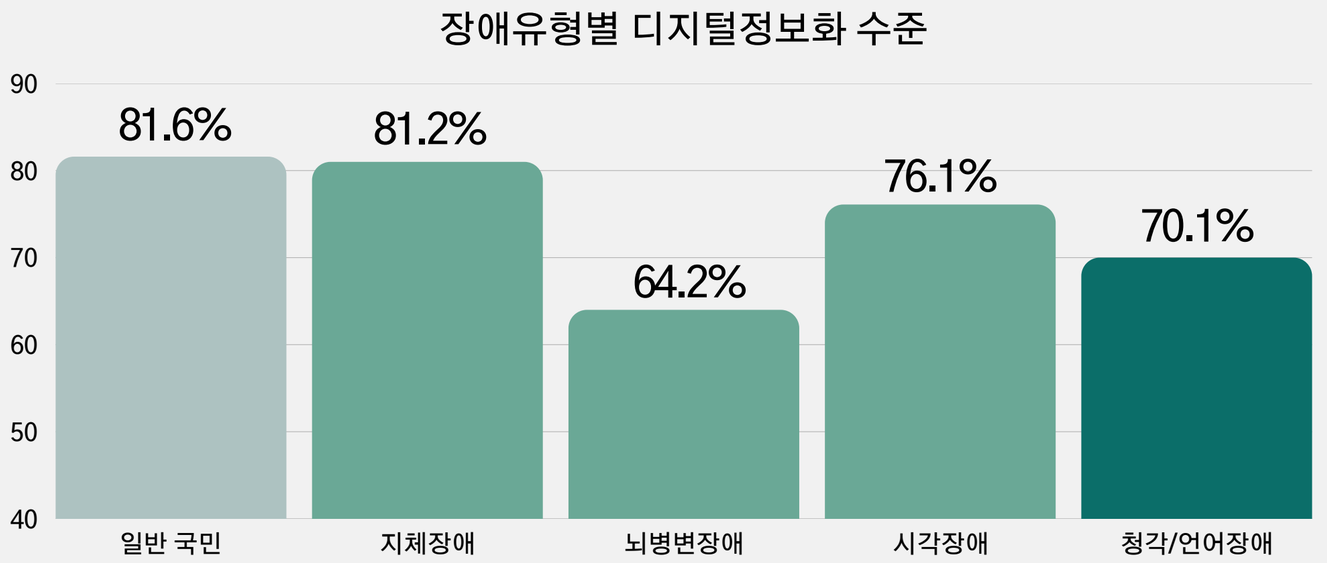
01. 프로젝트 개발 배경



국내 등록 청각장애인 수

44만 명

보건복지부 장애인등록현황 통계에 따르면,
2023년 말 기준, 국내 등록 청각장애인 수는
44만 명에 이릅니다.



청각장애인의 디지털 정보화 수준

70.1%

한국장애인개발원 연구에 따르면 청각장애인은 다른 장애 유형과 비교했을 때,
스마트 기기 활용이나 인터넷 사용에 더욱 취약한 것으로 나타났습니다.

01. 프로젝트 개발 배경

수어를 텍스트로 바로 변환해주면
청인도 바로 알아들을 수 있잖아요.

통역사가 없으면 대화가 힘든데,
화상 회의에서 수어를 번역해준다면
농인과 바로 소통할 수 있어
정말 편리할 것 같아요.

인터뷰 대상자

기존 연구

- 수어 → 한국어 번역 **상용화 서비스 X**
- [‘ㄱ’, ‘ㄴ’, ‘ㅏ’, ‘1’, ‘2’, ‘3’]와 같은 **지수어, 지문자 인식**

해결책

- 한국 수어의 형태 이해 (한국어 ≠ 한국수어)
  - 한국어대응수어: [“나”, “학교”, “가다”]
  - 한국수어: ["나", "가다", "학교"]
- 한국 수어의 **형태소(gloss) 단위 해석**
- ‘안녕하세요’, ‘반갑습니다’ 등 단일 gloss로 이루어진 문장을 번역하여 간단한 소통을 가능하게 함.

02. 프로젝트 개요

"화상 회의 환경에서 AI 기술을 통해 수어를 실시간으로 번역하여
농인과 비농인 간의 원활한 소통을 지원하는 웹 확장 프로그램"

실시간 수어 인식

사용자의 카메라 영상을 통해 손동작, 표정, 입 모양 등 비수지 신호를 포함한 수어 발화 인식

AI 문장 번역

파인튜닝된 Transformer 모델을 통해 인식된 gloss 단위의 수어를 번역

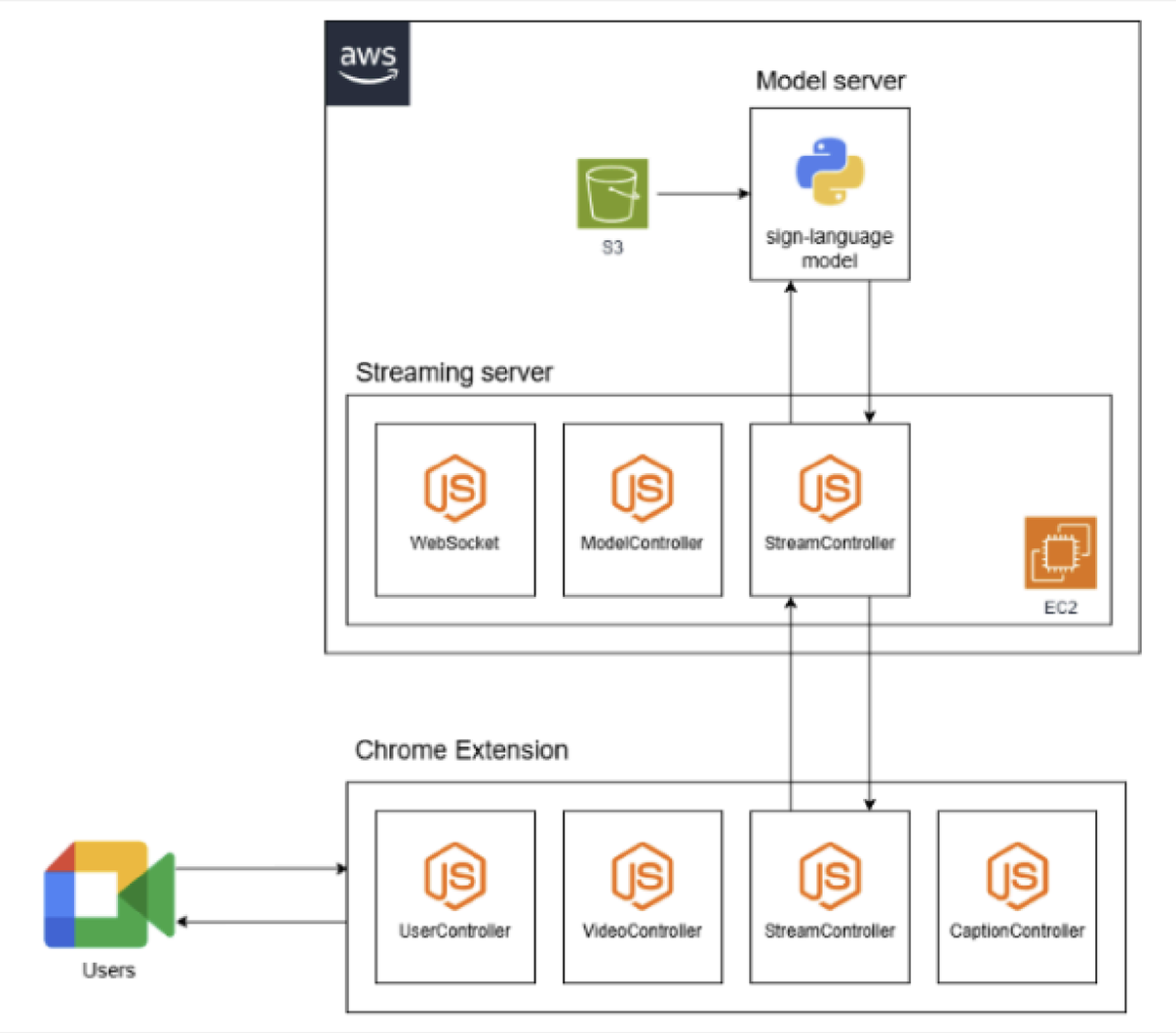
결과 시각화

번역된 내용을 화상 회의 화면에 맞게 자막 및 전체 스크립트 형태로 실시간 출력

접근성

별도 프로그램 설치 없이 기존 화상 회의 플렛폼에서 바로 사용하는 웹 확장 프로그램 방식

03. 서비스 소개



Chrome Extension

- 화상 회의 화면에서 번역 대상 영상을 캡쳐
- 영상 스트림의 실시간 전송 및 번역 결과 수신

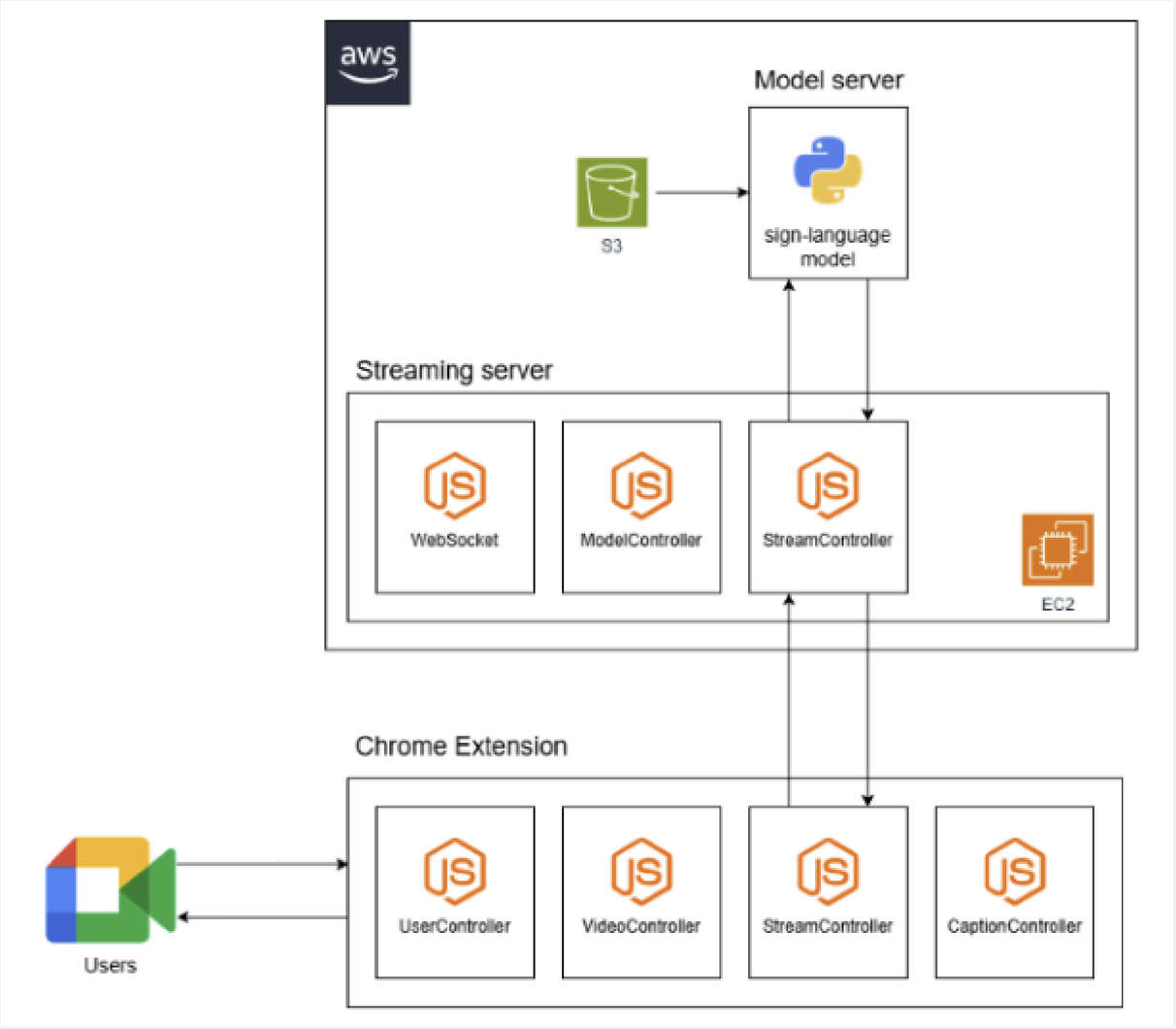
Streaming Server

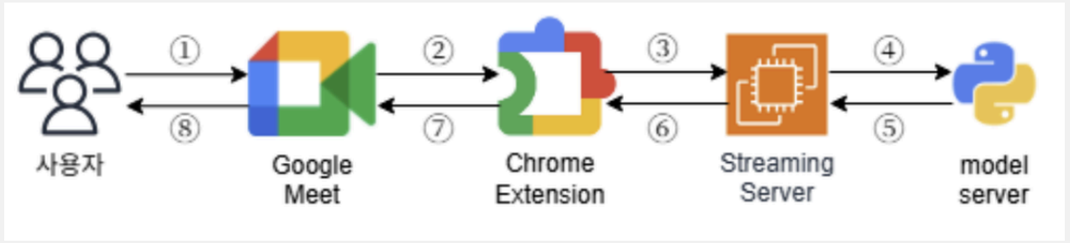
- 다수의 클라이언트로부터 영상 스트림을 수집 및 관리
- Model server 와의 번역 요청, 응답 프로세스 조율

Model Server

- 영상 입력 데이터로부터 손동작 등 수어 특징 추출
- 학습된 Transformer 기반 모델을 통한 문맥 이해 및 문장 단위 번역
- 번역 결과를 Streaming Server로 반환

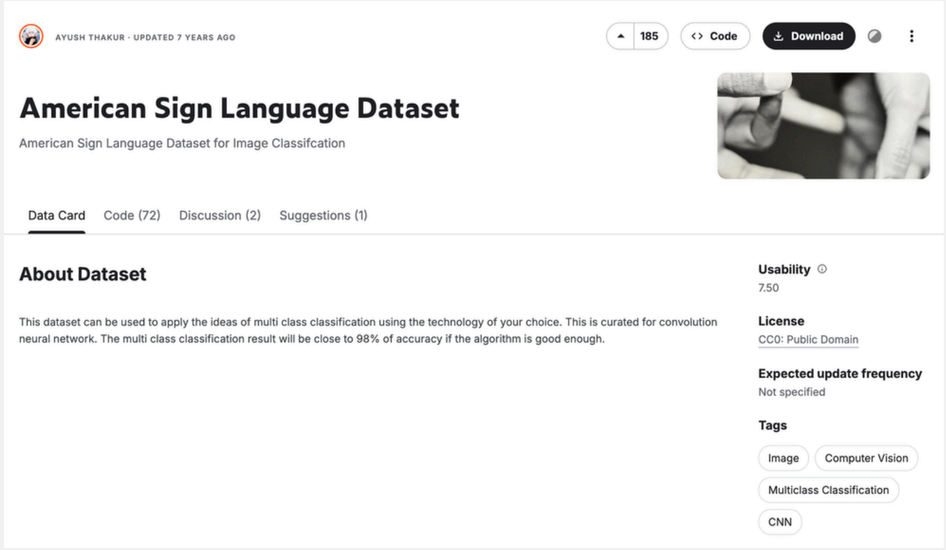
03. 서비스 소개

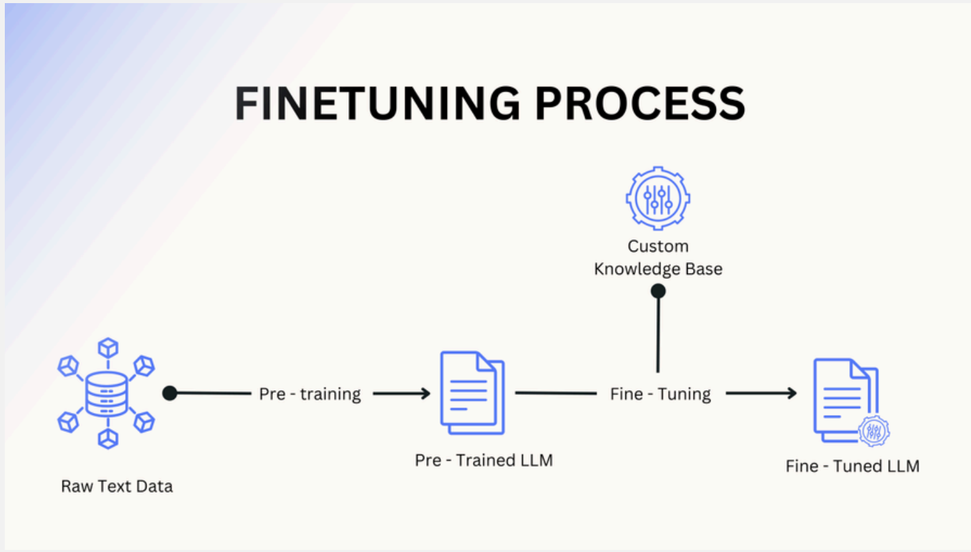




1. 사용자가 Google Meet 화상 회의에 참여해 카메라를 통해 소통한다.
2. 수어 번역 확장 프로그램을 활성화하면, 회의 화면에서 사용자의 영상을 실시간 캡처한다.
3. 캡처된 영상은 프레임 단위 스트림으로 변환되어 WebSocket을 통해 서버로 전송된다.
4. 스트리밍 서버가 받은 영상 스트림을 Model Server로 전달해 번역을 요청한다.
5. AI 모델이 영상을 분석해 수어를 문맥에 맞는 한국어 텍스트로 변환하여 스트리밍 서버로 전송한다.
6. 스트리밍 서버는 다시 WebSocket을 통해 확장 프로그램으로 전송된다.
7. 확장 프로그램이 텍스트를 회의 화면에 실시간 자막 형태로 오버레이 표시한다.
8. 사용자는 자막을 통해 농인과 실시간으로 대화할 수 있다.

03. 서비스 소개 – 핵심 기술





ASL (American Sign Language) 모델

- 대규모 데이터(56GB, 수백만 샘플) 기반으로 학습된 고성능 수어 인식 모델
- 풍부한 손·신체 움직임 표현력을 학습한 benchmark 수준의 pretrained 모델

ASL 모델 Fine-Tuning

- ASL pretrained 모델을 KSL에 전이하여 데이터 부족 문제 해결
- 적은 데이터로도 높은 성능을 달성하는 효율적 학습 전략

03. 서비스 소개 – 핵심 기술

```python
for ksl_layer in ksl_model.layers:
    asl_layer = asl_by_name.get(ksl_layer.name)
    if asl_layer is None:
        continue
    asl_w, ksl_w = asl_layer.get_weights(), ksl_layer.get_weights()
    if all(a.shape == k.shape for a, k in zip(asl_w, ksl_w)):
        ksl_layer.set_weights(asl_w)
    # else: stem_conv/classifier는 shape 불일치 → 랜덤 초기화
```

ASL, KSL 구조가 거의 같지만 레이어 입력/출력 크기가 달라 그대로 사용 불가함

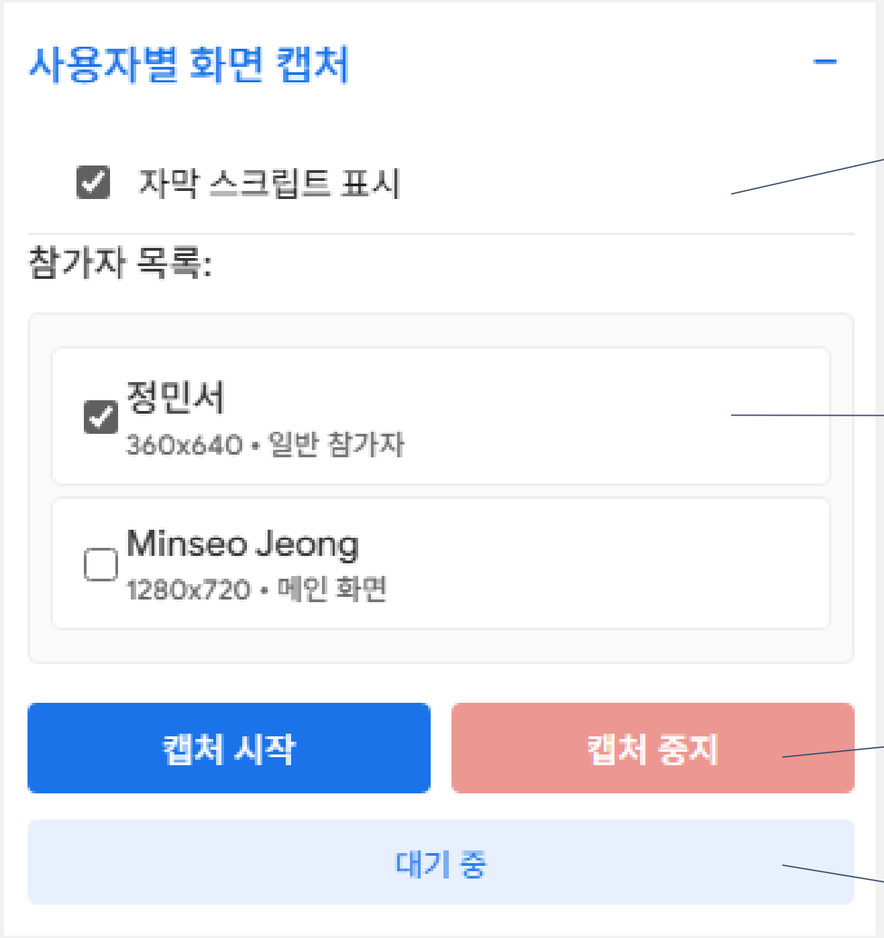
- Conv1DBlock 6개, TransformerBlock 2개 → 파라미터 그대로 전이
- stem_conv, classifier → 새로 학습 (KSL에 맞게 재적응)

```python
# Stage 1 (LR=1e-3): stem_conv + classifier만 학습 (backbone 동결)
freeze_layers(model, frozen_names=['stem_bn','mbblock_1~6','top_conv'])

# Stage 2 (LR=1e-4): 상위 절반 backbone 해동
freeze_layers(model, frozen_names=['stem_bn','mbblock_1','mbblock_2','mbblock_3']

# Stage 3 (LR=5e-6): 전체 파인튜닝 + label smoothing
unfreeze_all(model)
model.compile(loss=SparseCategoricalCrossentropy(label_smoothing=0.1))
```

전이된 backbone 가중치는 대용량 ASL 데이터로 학습된 좋은 초기값임
소량의 KSL 데이터로 처음부터 모든 레이어를 동시에 학습하면 이 초기값이
무너져버리므로 이를 막기 위해 단계적으로 해동하여 학습함

전이학습을 통해 안정적으로 수렴하며, 일반화 성능을 유지한 정확도 약 59%의 모델을 확보

04. 주요 기능 소개



자막 스크립트 조작

체크박스를 활성화/비활성화 하는 것을 통해 사용자가 상황에 맞게 자막 스크립트를 켰다가 껐다가 할 수 있음

참가자 목록

회의에 참여한 사용자 중 카메라가 켜져 있어 접근 가능한 사용자 목록을 자동으로 보여줌. 체크박스를 클릭하여 어느 사용자의 화면을 녹화할 것인지 선택 가능

해석 시작 / 해석 중지 버튼

참가자 목록에서 선택한 사용자의 화면을 해석 시작 및 중지 할 수 있음

상태 표시란

스트리밍 서버와의 상태를 표시함

05. 기대 효과

수어 사용자의 디지털 자립성 향상	• 비대면 환경에서 의사소통 장벽을 해소해 학습권·노동권을 보장 • 통역사 없이 스스로 소통하며 디지털 자립성 강화
교육 분야	• 청각장애 학생이 통역사 유무와 관계없이 원격 수업에 참여 • 포용적 스마트 학습 환경 구축 가능
업무 및 비즈니스 분야	• 농인 근로자가 팀 내 협업에 자연스럽게 참여 • 기업은 농인 고객 대상 비대면 응대·영업 원활히 수행 가능
공공 및 생활 편의 분야	• 관공서, 은행, 병원 등에서 통역사 없이도 업무·진료 가능 • 비대면 서비스 접근성과 자율성 향상

06. 기술적 시도 – 직접 학습

백본 구조 (Backbone Structure)

Input (Sequence Input: max_len, feature_dim)
↓
Stem Layer (Dense + BatchNorm)
↓
Conv1D Blocks × 3
↓
Transformer Block
↓
Conv1D Blocks × 3
↓
Transformer Block
↓
Feature Projection (Dense)
↓
Global Average Pooling
↓
Late Dropout
↓
Classifier (Softmax)

사용 데이터셋: AIHub 수어 영상

- 영상을 Openpose로 추출한 2D 키포인트
- 총 데이터셋 크기: 500,000

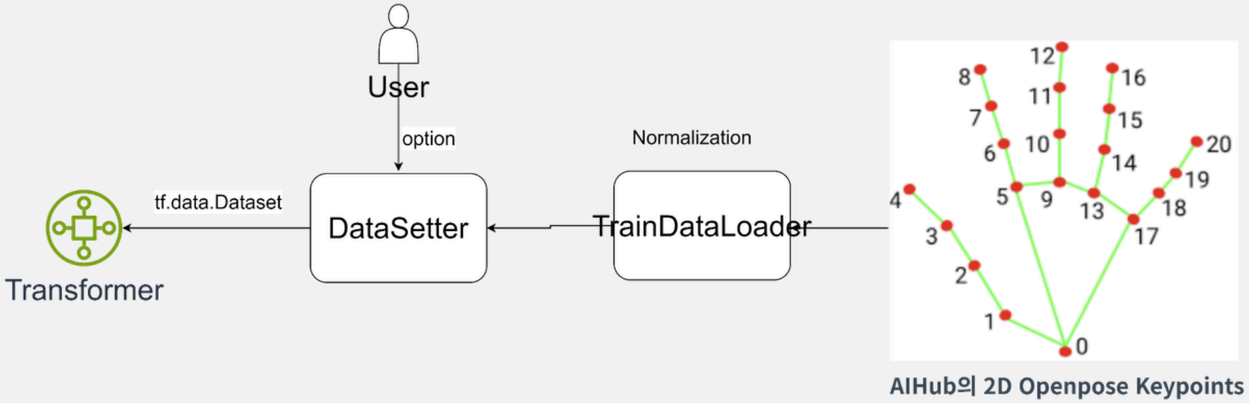
→ 사용 데이터셋 크기: 80*33 = 2640개

  - 클래스 당 키포인트 비디오 80개
  - 단일형태소 클래스 33개

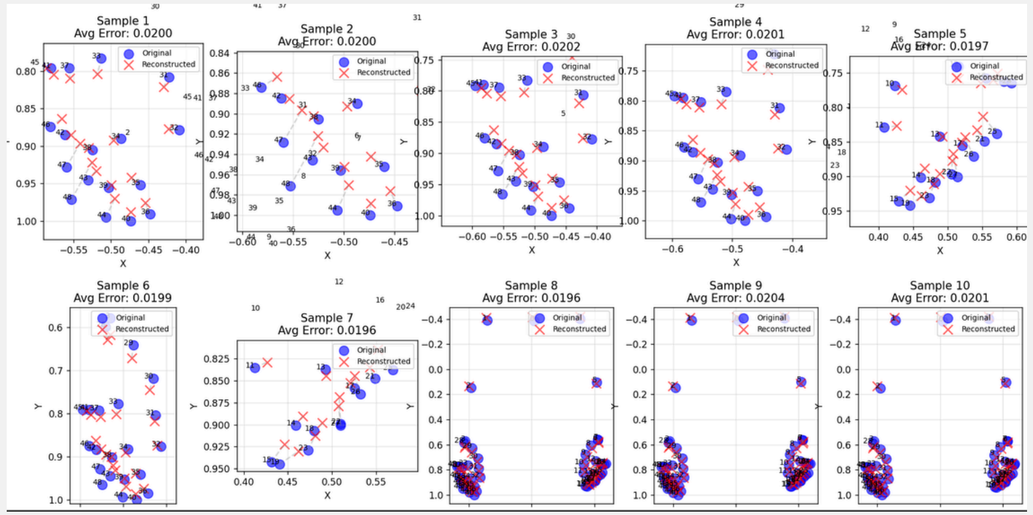
사용 모델: CNN-Transformer

- 시퀀스 데이터를 시계열 단위로 분석해 글로스(gloss) 번역
- Self-Attention으로 프레임 간 관계를 동시에 고려
  - Attention Score를 통해 핵심 키포인트에 집중
- 전역적 범위의 키포인트 이해로 CNN의 지역적 해석 극복
  - 길고 복잡한 수어 글로스 해석에 우위

06. 기술적 시도 – 직접 학습



AIHub의 2D Openpose Keypoints



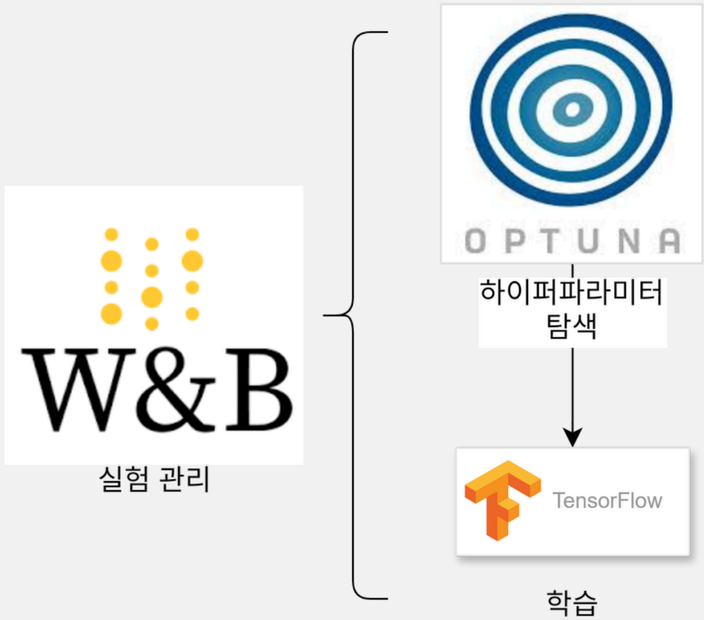
데이터셋 생성 파이프라인

- **TrainDataLoader:** 실제 클라우드 저장소(AWS S3/GCP GCS)에서 AIHub의 2D Openpose 키포인트 데이터를 가져와 정규화 처리
- **DataSetter**: 사용자 옵션에 따라 **데이터 차원수 축소/훈련 모델별**로 전처리된 데이터를 **numpy 배열/tensorflow 데이터셋** 형태로 가져옴.

Parametric Umap 기반 차원 축소

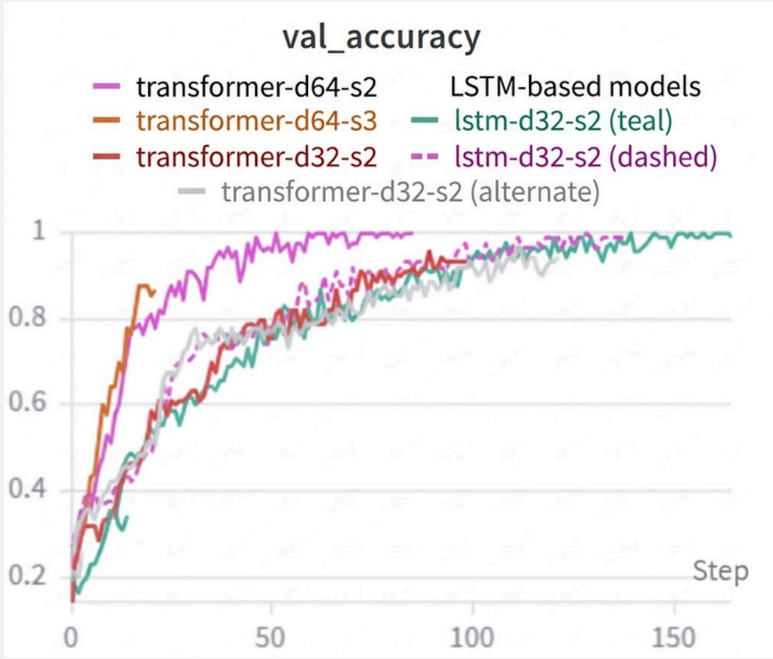
- 2D 키포인트 데이터를 저차원 벡터로 변환
- 비선형 구조 보존을 위해 UMAP 알고리즘 적용
- 전역·국소 구조를 함께 유지해 핵심 특징 효율적 추출
- 노이즈 제거 및 학습 효율·번역 정확도 향상

06. 기술적 시도 – 직접 학습의 결과



모델 학습 자동화 파이프라인

- **실험 관리**: WandB를 통한 모델의 하이퍼파라미터, 학습 과정 및 결과 기록 & 표 작성
- **최적 하이퍼파라미터 탐색**: Optuna 라이브러리를 이용하여 학습에 중요한 하이퍼파라미터들의 범위를 설정하여 `val_loss`를 최소화하는 방향으로 학습 유도
- **학습**: 도출된 최적 파라미터로 학습



실험 결과

- UMAP 적용을 하지 않은 Transformer 모델이 가장 정확률이 높음.
- 학습 정확도가 90%에 가깝지만 실제 사용 환경에서는 인식률이 낮음.
- 너무 작은 데이터셋 규모(Train에 사용되는 데이터셋 클래스 33개 기준 2400개)에 비해 복잡한 모델 구조가 원인으로 파악됨.
- 모델이 전반적인 키포인트 흐름에 대해서 학습했으나 특정한 연구 환경 혹은 개인에 과적합됐다고 볼 수 있음

06. 기술적 시도 – 향후 방향성

데이터 증강

- 가우시안 노이즈 추가
- 키포인트 마스킹 - 누락을 통한 오클루션(Occlusion) 환경 적응

3D 키포인트 활용

- z축을 활용하여 깊이도 함께 학습함
- 2D 키포인트만으로는 구분하기 어려운 자세 차이를 더 정확하게 표현 가능

문맥 단위 Transformer 개발

- 현재는 하나의 형태소를 기준으로 한 문장/단어에 대해서만 학습을 진행한 상태로, 여러 가지의 형태소로 이루어진 문장을 학습하는 키포인트 분류 모델 개발로 확장 가능.
- 같은 단어여도 다른 문장으로 해석하는 데 비언어적 표현이 많이 사용되는데, 비언어적 표현이 사용자마다 매우 다르다는 것을 감안하여 대규모의 데이터 학습이 필요할 것으로 보임.