SK네트웍스 **Family AI 17**기 **3Team**

데이터 전처리 파인튜닝 데이터 전처리 결과서

1. 문서 개요

- 프로젝트명: ⚾BAIS⚾
- 전처리 목적: 3명의 해설위원의 말투, 어휘, 문장 구조를 모델이 학습할 수 있도록 정제된 파인튜닝용 데이터셋을 구축하는 것을 목표로 함.
- 문제 정의: AI 모델이 3명의 해설위원의 언어적 특징을 학습해 야구 하이라이트 영상에서 사용자가 선택한 해설위원의 말투로 자연스럽게 해설하도록 하기 위해, 정확하고 노이즈가 제거된 텍스트 데이터셋 구축이 필요함.

2. 데이터셋 개요

- 데이터 출처 : KBO 야구 경기 풀영상, WBC 2023 야구 경기 풀영상
- 데이터 수집 방법 :
  - KBO와 WBC 야구 경기 풀영상을 팀원들이 직접 청취하며 캐스터와 해설위원이 발화한 전체 구간 선별
  - 선별된 구간의 캐스터와 해설위원의 음성을 직접 텍스트로 전사
  - 전사된 문장을 캐스터의 발화(0)와 해설위원의 응답(1)을 구분해 CSV 파일 형태로 정리하여 저장
  - 관중 소리나 광고 등 비해설 구간은 모두 제외함
- 데이터 구성: 3명의 해설위원의 각각 발화 CSV 파일

항목명	설명	예시
speaker	화자 (캐스터/해설위원)	0 / 1
text	해설 텍스트	2루까지 서서 들어간 박민우

- 데이터 파일 명 : 예시 “기아 vs 두산 이순철 2025 04 19.csv”
- 데이터 타입 : CSV
- 데이터 양 : 총 168 Row

- 원본 데이터 샘플:

speaker	text
0	있고, 권희동 선수. 윤도현 선수가 빠지기 때문에 친구 윤도현 선수가 선발 출장을 했습니다.
1	예.
0	1번 타자 박민우가 타석에 들어왔습니다. 양현종의 초구를 받아 때립니다. 오른쪽 높게 떠가는 타구입니다. 우익수 뒤로, 담장!
1	예.
0	상단을 때리고 떨어집니다. 2루. 2루까지 서서 들어간 박민우.
1	박민우 선수는 초구부터 너무나 좋은 타구를 날렸고, 넘기지 못한 부분이 아, 아쉬울 것 같아요. 양현종 선수 입장에서 봤을 때는 이게 안 넘어간 게 다행입니다. 너무나 잘 맞은 타구였어요.
0	예.
1	그러니까 통산 전적도 박민우 선수가 이 양현종 선수에게 굉장히 강한데, 올라와서 초구를 던졌는데 너무나 좋은 타이밍으로, 예. 너무 아쉬운 타구가 됐어요. 거의 뭐, 다 넘어가는 겁니다. 창살 맞고 나
0	예. 상단에. 철조망을 맞고 나왔습니다. 1회부터 위기를 맞고 있는 양현종 선수고요. 김주원, 2번 타자를 상대로 바깥쪽에서 스트라이크를 잡습니다.
1	김주원 선수가 시범 경기 때 쭉 하는 모습을 보니까 작년 모습하고는 완전히 달라진 모습으로, 예. 시즌을 준비하고 왔더라고요.
0	예. 지난 시즌 11승 5패 기록하면서 다시, 시즌 10승 투수로 복귀한 양현종 선수입니다. 날카롭게 밀어 봤는데, 파울.
1	그러니까, 작년하고 좀 달라졌다고, 어, 달라졌다고 제가 느끼는 부분을 설명을 드리면,

3. 전처리 프로세스 개요

- 전체 흐름도:

  풀 경기 영상 수집 → 캐스터/해설위원 전체 해설 구간 청취 및 전사 → 불용어 제거 및 발화 최소 길이 제한 → 파인튜닝용 JSONL 파일로 변환

- 전처리 파이프라인 요약:

단계	목적	수행 작업	사용 도구/라이브러리
1	해설 구간 데이터 전사 및 파일로 저장	경기 풀영상 청취 및 캐스터(0)와 해설위원(1)의 발화를 구분해 전사 후 CSV 파일로 저장	수동 전사
2	파인튜닝용 데이터	파인튜닝 입력 형식에 맞게 캐스터(Q) - 해설위원(A)로 CSV → JSONL 파일 변환	Python (pandas)

4. 세부 전처리 단계

4-1. 결측치 처리

- 결측치 존재 여부: 없음

4-2. 이상치 처리

- 정의한 이상치 기준 : 발화 text 길이가 10 이하
- 처리 방식 및 영향:

항목	기준	처리 방식	제거 수
text_length	< 10자	10자 이상인 행만 저장	26건

- 처리 코드 (Python):

```
merged_df =
merged_df[merged_df['messages'][0]['content'].str.len() > 10]
```

4-3. JSONL 파일로 변환

- 0/1 → “user” / “assistant”
- “content”에 캐스터와 해설위원 발화 입력
- 변환 예시 :

```
{"messages": [{"role": "user", "content": "투볼 투스트라이크. 5구 스윙 삼진. 박세웅 출발 좋습니다. 슬라이더로 1번 타자 맨식크 삼진으로 돌려보내는 대한민국의 선발 박세웅입니다."}, {"role": "assistant", "content": "지금도 좋은 슬라이더를 던져서 스윙을 유도를 했습니다. 자, 좋은 슬라이더를 던져서 어, 투수 잡기 위해 저런 스윙을 유도해서 삼진 잡았지만 저 또한 슬라이더를 던졌기 때문에 타자가 못 쳤다가 아니라 로케이션이 좋았기 때문에 못 쳤다라는 거를 명심해야 됩니다."}]}
```

5. 학습/검증 데이터 분리

- 분리 기준 및 방법:
  - 기준: 무작위 분할
  - 비율: Train 70% / Val 15% / Test 15%
- 분리 코드:

```
def split_data(data, train_ratio=0.7, val_ratio=0.15,
test_ratio=0.15, seed=42): 로 함수 정의
```

- 분리 후 건수:

구분	데이터 수
학습 데이터	99건
검증 데이터	21건
테스트 데이터	22건

6. 전처리 결과 요약 및 평가

- 전처리 후 전체 건수: 168건 → 142건

- 품질 향상 지표:
    - 이상치 제거: 총 26건 제외
    - 레이블 정리 및 불균형 개선
- 향후 활용 방안:
    - 3명의 해설위원의 언어적 특징을 학습시키기 위한 sLLM 모델의 파인튜닝에 활용할 예정
    - 향후 생성된 해설 음성을 야구 하이라이트 영상과 결합하여 실제 해설 서비스에 적용할 계획