인공지능 데이터 전처리 결과서

1. 데이터 개요

구분	데이터명	수집 목적	데이터 형태	데이터 출처	데이터 크기	처리 상태
DATA-001	밈 기본 정보	밈 정의, 기원, 키워드 수집	JSON	나무위키, Google 검색 (Serper API)	110건	테스트 완료 / 자동화 개발 중
DATA-002	밈 활용 예시	밈 사용 맥락, 패러디 사례	JSON	블로그, 뉴스 크롤링	밈당 3~5개	테스트 완료 / 자동화 개발 중
DATA-003	YouTube 쇼츠 정보	밈 관련 인기 영상 수집	JSON	YouTube Data API v3	밈당 상위 3개	테스트 완료 / 자동화 개발 중
DATA-004	영상 분석 데이터	밈 구간 탐지, 동작 분석	JSON	Google Gemini API	밈당 1건	테스트 완료 / 자동화 개발 중
DATA-005	오디오 분석 데이터	음성 운율, TTS 힌트 생성	WAV + JSON	YouTube 영상 추출 (yt-dlp)	밈당 1건	테스트 완료 / 자동화 개발 중
DATA-006	텍스트 파인튜닝 데이터1	캐릭터 대사 생성 및 대화 스타일 학습	TSV	Korean SmileStyle Dataset	3,706개	추가 수집 중 / 후처리 필요
DATA-007	텍스트 파인튜닝 데이터2	캐릭터 대사 생성 및 대화 스타일 학습	JSON	AI Hub - Office 상황 데이터셋	46,414개	추가 수집 중 / 후처리 필요
DATA-008	음성 파인튜닝 데이터	음성 생성 모델의 보이스 클로닝	JSON	AI Hub - 감성 및 발화스타일 음성 합성 데이터	560개	추가 수집 중 / 후처리 필요

- **밈 수집 데이터 (DATA-001~005)**: 밈 110건 기준 약 550건 (밈당 평균 5건)
- **파인튜닝 데이터 (DATA-006~008)**: 50,680개
- **데이터 수집 기간**: 2026.01.08 ~ 2026.01.09

2. 데이터 필드 상세

DATA-001: 밈 기본 정보

필드명	데이터 타입	설명	예시
meme_id	BIGINT	밈 고유 식별자 (Auto increment)	1, 2, 3
meme_name	VARCHAR	밈 이름 (UNIQUE)	나니가 스키, 랫댄스
definition	TEXT	밈 정의 (3~5문장)	"나니가 스키"는 일본어로 "뭐가 좋아?"라는...
origin	JSONB	기원 정보 {source, creator, date, platform}	{"source": "러브라이브!", "platform": "TikTok"}
key_phrase	TEXT	quotable 밈의 핵심 대사	何が好き?, Absolute Cinema
keywords	JSONB	검색 키워드 배열	["나니가스키", "러브라이브", "틱톡"]
risk_info	JSONB	위험/논란 정보 {controversies, risk_level}	{"risk_level": "low", "controversies": []}
meme_type	VARCHAR	밈 유형 (quotable | performable)	quotable

DATA-002: 밈 활용 예시

필드명	데이터 타입	설명	예시
example_id	INT	예시 고유 식별자	1, 2, 3
meme_id	BIGINT	참조 밈 ID (FK)	1
source_url	TEXT	크롤링 원본 URL	https://blog.naver.com/...
source_title	TEXT	원본 페이지 제목	나니가스키 밈 사용법
context	TEXT	사용 맥락 설명	친구에게 취향을 물을 때 사용
original_quote	TEXT	원문 발췌	"이 밈은 주로 귀여운 상황에서..."
example_type	VARCHAR	예시 유형	챌린지, 패러디, 상황극, 짤, 댓글

DATA-003: YouTube 쇼츠 정보

필드명	데이터 타입	설명	예시
video_id	VARCHAR	YouTube 영상 ID (11자)	ZMQRVtrPAWs
meme_id	BIGINT	참조 밈 ID (FK)	1
video_title	VARCHAR	영상 제목	나니가스키 챌린지 모음
youtube_url	TEXT	YouTube URL	https://youtube.com/shorts/ZMQRVtrPAWs
view_count	BIGINT	조회수	13466259

DATA-004: 영상 분석 데이터

필드명	데이터 타입	설명	예시
video_id	VARCHAR	YouTube 영상 ID (PK, FK)	ZMQRVtrPAWs
time_stamp	JSONB	밈 구간 {start, end, detected_text}	{"start": 2.8, "end": 3.4, "detected_text": {...}}
motion_prompt_hint	TEXT	영상생성용 프롬프트 (50-100 words)	Cute character tilts head, raises hand...
motion_sequence	JSONB	시간별 동작 분해	[{"time": "0-2s", "action": "머리 기울임"}]
body_parts	JSONB	부위별 동작 {head, face, arms, hands, torso}	{"head": "tilts right", "face": "smile"}
style_keywords	JSONB	스타일 키워드 배열	["cute", "kawaii", "anime"]
camera_motion	VARCHAR	카메라 움직임	static, slow_zoom_in, subtle_pan

DATA-005: 오디오 분석 데이터

필드명	데이터 타입	설명	예시
video_id	VARCHAR	YouTube 영상 ID (PK, FK)	ZMQRVtrPAWs
audio_file	TEXT	추출된 오디오 파일 경로	data/raw/audio/ZMQRVtrPAWs.wav
audio_json.prosody	JSONB	운율 분석 {pitch_mean, speaking_rate}	{"pitch_mean": 220.5, "speaking_rate": 4.2}
audio_json.ssml	TEXT	생성된 SSML	`<speak><prosody rate="fast">...</prosody></speak>`
audio_json.detected_text	TEXT	음성 인식 텍스트	何が好き

DATA-006: 텍스트 파인튜닝 데이터1 (SmileStyle)

필드명	데이터 타입	제약 조건	설명
utterance_id	INTEGER	NOT NULL	원본 데이터 식별자
character	TEXT	NOT NULL	캐릭터 유형 (부장: azae / 사원: gentle)
utterance	TEXT	NOT NULL	발화 텍스트

DATA-007: 텍스트 파인튜닝 데이터2 (Office)

필드명	데이터 타입	제약 조건	설명
utterance_id	INTEGER	NOT NULL	원본 데이터 식별자
domain	VARCHAR(30)	NOT NULL	대화 유형 구분 (daily, task)
system_utterance	TEXT	NOT NULL	시스템 발화 내용

DATA-008: 음성 파인튜닝 데이터

필드명	데이터 타입	제약 조건	설명
audio_id	SERIAL	PK	음성 데이터 고유 식별자
filename	VARCHAR(30)	-	음성 파일 이름
character	VARCHAR(10)	-	캐릭터 (부장/신입)
speaker_id	INT	-	성우 ID
speaker_age	INT	-	성우 나이
speaker_gender	VARCHAR(10)	-	성우 성별
origin_text	VARCHAR(200)	-	문장 원문
ptr	VARCHAR(200)	-	발음 전사
tr	VARCHAR(200)	-	철자 전사
emotion	TEXT	-	감정 (예: 분노)
intensity	INT	-	감정의 강도
style	VARCHAR(10)	-	스타일 (예: 독백체)
sub_style	VARCHAR(10)	-	세부 스타일 (예: 애니체)
duration	VARCHAR(20)	-	음성 구간 길이
file_duration	VARCHAR(20)	-	음성 파일 구간 길이
votes	JSONB	-	투표 정보
created_at	TIMESTAMP	DEFAULT NOW()	생성 일시

데이터 수집 및 활용의 적법성 검토

1. 적법성 검토 결과

구분	상업 이용	학습 허용	크롤링 준수	개인정보 보호	비고
DATA-001	X	O	O	O	출처 명시 필수 (https://namu.wiki/robots.txt)
DATA-002	O	O	O	O	공개 게시물만 수집
DATA-003	O	O	O	O	YouTube Developer Policies 준수
DATA-004	O	O	O	O	2차 가공 데이터
DATA-005	O	O	O	O	음성 원본 미저장 (구간만 추출)
DATA-006	O	O	-	O	상업적 사용 시 smilegate_ai@smilegate.com 문의
DATA-007	O	O	-	O	원본 라이선스 범위 내 2차 가공 후 사용 가능
DATA-008	O	O	-	O	원본 라이선스 범위 내 2차 가공 후 사용 가능

2. 준수 사항

- **저작권 준수**: 각 데이터의 이용약관을 검토하여 상업적 이용 및 2차 가공 가능 여부 확인 완료

- **크롤링 이용약관 준수**: robots.txt 규정에 따라 서버 부하를 최소화하는 방식으로 크롤링 수행
- **개인정보 보호**: 수집 과정에서 개인 식별 정보(이름, 연락처 등)는 즉시 제외 또는 비식별 처리

수집 자동화

1. 수집 방식 및 도구

구분	수집 방식	사용 도구	자동화 주기
DATA-001	검색 API + 크롤링	Serper API, BeautifulSoup	매일 00:00 (Phase 1)
DATA-002	웹 크롤링	BeautifulSoup, requests	밈 수집 시 동시 실행
DATA-003	API 호출	YouTube Data API v3	밈 수집 시 동시 실행
DATA-004	영상 분석	Google Gemini API	주간 콘텐츠 생성 시
DATA-005	오디오 추출 + 분석	yt-dlp, ffmpeg, librosa	주간 콘텐츠 생성 시
DATA-006	공개 데이터셋 다운로드	GitHub, Web Browser	1회성 수집
DATA-007	공공 데이터셋 다운로드	AI Hub Downloader	1회성 수집
DATA-008	공공 데이터셋 다운로드	AI Hub Downloader	1회성 수집

2. 수집 자동화 코드 (Data 01~05)

```
# Agent v8 실행 (DATA-001 ~ DATA-005 통합 수집)
uv run python -m scripts.agent.meme_agent_v8 "밈이름"

# 배치 실행 (다중 밈 수집)
uv run python -m scripts.agent.batch_runner
```

3. 데이터 유효성 검증 (Data 01~05)

검증 단계	검증 방법	예외 처리
1차: 필수 필드 검증	meme_name, definition, source_url 존재 확인	누락 시 수집 실패 로그 기록
2차: 중복 검증	meme_name UNIQUE 제약 조건	중복 시 기존 데이터 업데이트
3차: 출처 검증	usage_examples의 source_url 유효성 확인	유효하지 않은 URL 제외
4차: 영상 ID 검증	YouTube video_id 11자 형식 확인	형식 불일치 시 제외
5차: 밈 구간 검증	time_stamp.start < time_stamp.end 확인	역전 시 전체 오디오 분석으로 fallback

전처리 프로세스

1. 전체 흐름도

① 수집 → ② 결측치 처리 → ③ 이상치 탐지 → ④ 정규화 → ⑤ 데이터 분리/저장

2. 밈 데이터 전처리 (DATA-001 ~ 005)

단계	목적	수행 작업	사용 도구
결측치 처리	누락값 제거/대체	Null 행 제거, 기본값 대체	pandas, PostgreSQL
이상치 처리	비정상 데이터 제거	조회수 0 영상 필터링, 60초 초과 영상 제외	numpy

단계	목적	수행 작업	사용 도구
정규화	텍스트 전처리	중복 키워드 제거, 특수문자 정제	re, nltk
분류	밈 타입 분류	quotable / performable 자동 분류	LLM (GPT-4o)
저장	DB 적재	PostgreSQL memedb 스키마 저장	psycopg2, SQLAlchemy

데이터별 전처리 상세

구분	전처리 작업
DATA-001	나무위키/웹검색 결과 LLM 요약, 키워드 중복 제거, 논란 키워드 자동 감지
DATA-002	크롤링 원문에서만 예시 추출 (할루시네이션 방지), source_url 필수 검증
DATA-003	Shorts 영상만 필터링 (60초 이하), 조회수 상위 3개 선정, 중복 제거
DATA-004	Gemini Vision 분석 결과 JSON 파싱, motion_prompt 50단어 이상 검증
DATA-005	ffmpeg 밈 구간 추출, librosa 운율 분석, LLM SSML 생성

3. 파인튜닝 데이터 전처리 (DATA-006 ~ 008)

3.1. 전처리 목적

- 수집한 대화 데이터를 학습용으로 정제
- 캐릭터에 맞게 일관성을 유지하도록 전처리 수행
- 상황에 맞지 않는 발화는 OpenAI로 캐릭터 기준에 맞게 수정

3.2. 이상치 탐지 및 처리

이상치 기준:

- 대화 길이가 지나치게 짧은 데이터 (글자수 14개 이하)
- 의미 없는 반복 문자 또는 이모지로만 구성된 발화
- 캐릭터 설정(회사원)과 어긋나는 말투·태도·경험 언급

처리 방법:

- 대화 길이 기준으로 필터링
- 정규식 기반으로 반복 문자 및 의미 없는 패턴 탐지
- OpenAI 기반 LLM을 활용하여 상황에 맞지 않는 대사를 캐릭터 기준에 부합하도록 수정

처리 결과: 전체 수집 데이터 3,706건 중, 이상치 2,002건을 제거하여 최종적으로 1,704건의 유효 데이터 확보

3.3. 결측치 처리

대상 결측 필드:

- user_utterance
- system_utterance

결측치 발생 건수(비율):

구분	총 데이터 수	결측 user_utterance	비율	결측 system_utterance	비율
1st	7,427	1,492	20.09%	38	0.51%
2nd	5,726	8	0.14%	1	0.02%
3rd	2,814	819	29.10%	0	0.00%
task	30,447	6,830	22.43%	10,580	34.75%

처리 방법:

- user_utterance는 전체적으로 데이터 길이가 짧고 결측 비율이 높아 학습에 부적합

- system_utterance는 학습 핵심 발화로 활용
- 결측치가 포함된 system 발화는 제거

처리 결과: 최종적으로 학습용 데이터셋은 **system_utterance** 필드만 사용

3.4. 데이터 정제

(1) 학습 데이터 구조 정규화 적용

- **기준**: 파인튜닝 학습에 적합한 instruction / input / output 구조로 데이터 포맷 통일
- **처리 과정**:
  - 원본 대화 맥락을 기반으로 지시문(instruction)을 새로 정의
  - 상황 설명 및 지문을 input 필드로 재구성
  - 캐릭터 성격·말투가 반영된 대사를 output 필드로 매핑
- **활용 방안**: 정규화된 데이터 구조를 파인튜닝 학습 데이터셋으로 활용하여 일관된 캐릭터 말투와 성격을 출력하는 생성 모델 학습에 적용

(2) 특수문자 처리

- **기준**: !@#$% 등 의미 없는 특수문자와 😆와 같은 이모지는 제거하고, 문맥에 영향을 주는 구두점은 유지
- **처리 과정**:
  - 정규식 기반으로 의미 없는 특수문자 패턴을 탐지
  - 반복 이모지 및 장식용 특수문자 제거
- **활용 방안**: 문장 구조와 말투는 보존한 상태로 텍스트 품질을 개선하여 캐릭터 성격·화법 학습의 정확도 향상에 기여

3.5. 파인튜닝 데이터 전처리 파이프라인 요약

단계	목적	수행 작업	사용 도구
결측치 처리	결측 및 불필요 필드 제거	user_utterance 전체 제외, system_utterance 결측치 제거	OpenAI API
이상치 처리	비정상 데이터 제거	단어 수 기준 이상치 제거, 상황에 맞지 않는 발화는 OpenAI로 재구성	OpenAI API
정규화	텍스트 전처리 및 형식 통합	특수문자 제거, 구두점 유지, JSON 구조 통일	OpenAI API

4. 파이프라인 도식화

```
P34 --> P5
P5 --> P6
P6 --> P7
P7 --> COND
COND -->|YES| P8
COND -->|NO| P10
P8 --> P9
P9 --> P10
P10 --> DB
```