

## SK네트웍스 Family AI과정 14기

# 데이터 수집 및 저장 수집 데이터 보고서

산출물 단계	데이터 수집 및 저장
평가 산출물	수집 데이터 보고서
제출 일자	2025.08.22
깃허브 경로	<a href="#">SKN14-Final-1Team</a>
작성 팀원	김준기, 김재우, 안윤지, 이나경, 이원지희, 정민영

### 1. 수집 데이터 개요

데이터명	수집 대상	수집 목적	사용 예정 기능	출처/저작권
Google API Docs	Google API Docs 의 11개 API 공식 문서	프롬프트 튜닝,  모델 응답 테스트	키워드 검색,  문맥 기반 질의응답	Google for developers
회사 사내 문서	프론트엔드팀 15장  백엔드팀 15장  DATA&AI팀 15장  CTO 15장	사내문서 축적	사내 규정 기반 지침 QA,  권한 기반 문서 검색,  팀 운영 문서	OpenAI  합성 데이터셋 (자체 생성)

## 2. 수집 방법 및 자동화 절차

### ● 수집 방식

- Google api 문서 : 웹 크롤링 → 텍스트 추출 → 텍스트 파일 저장
- 사내문서 : OpenAI API 프롬프트 합성 → 텍스트 파일 저장

### ● 수집 도구 또는 스크립트 설명:

- 사용한 언어/라이브러리
  - 구글 API 문서 크롤링 : python, Selenium, lxml, beautifulsoup 등
  - 사내 내부 문서 : python, openai 등
- 자동화 여부 및 주기: 일주일마다 수집 예정

### ● 예시 스크립트 또는 흐름도 첨부 (이미지, 순서도 또는 코드)

- 구글 API 문서 Selenium 웹 크롤링

```
1 # ===== 메인 =====
2 try:
3     full_start_url = ensure_hl_ko(urljoin(BASE_URL, START_URL))
4     driver.get(full_start_url)
5
6     print("사이드바의 링크를 수집 중...")
7     urls_to_crawl = collect_sidebar_links()
8     urls_to_crawl.insert(0, full_start_url)
9     # 중복 제거
10    urls_to_crawl = sorted(list(dict.fromkeys(urls_to_crawl)))
11    print(f"총 {len(urls_to_crawl)}개의 유효한 페이지 링크를 수집했습니다.")
12
13    for i, url in enumerate(urls_to_crawl, 1):
14        try:
15            print(f"\n({i}/{len(urls_to_crawl)}) 크롤링 중: {url}")
16            driver.get(url)
17
18            try:
19                article_element = wait.until(EC.presence_of_element_located((By.TAG_NAME, "article")))
20            except TimeoutException:
21                # 폴백
22                article_element = wait.until(EC.presence_of_element_located((By.TAG_NAME, "body")))
23
24            final_page_text = extract_page_text()
25
26            # 파일 저장
27            path = url.split("?")[0].replace(BASE_URL, "")
28            filename = re.sub(r'[/\?%*:|"<>]', "_", path).strip("_") + ".txt"
29            filepath = os.path.join(OUTPUT_DIR, filename)
30
31            with open(filepath, "w", encoding="utf-8") as f:
32                f.write(f"Source URL: {url}\n\n{final_page_text}")
33            print(f"저장 완료: {filepath}")
34
35        except Exception as e:
36            print(f"페이지 처리 중 오류 발생: {url} - {e}")
37
38        time.sleep(0.8)
39
40    finally:
41        driver.quit()
42        print("\n크롤링 완료! 브라우저를 종료합니다.")
```

○ 자동화 흐름

- **1. 원문 크롤링** : 지정된 링크에서 원문 데이터를 크롤링하여 확보
- **2. 원문 벡터 데이터베이스 삭제** : 재크롤링한 원문이 중복되지 않도록 기존 벡터DB에 저장된 원문 중 특정 태그를 가진 문서를 삭제 (메타 필터링)
- **3. 원문 벡터 DB에 저장** : 원문 벡터DB를 로드하여 새로 크롤링한 원문을 모두 ChromaDB에 저장
- **4. QA 데이터 생성 및 파일 저장** : 생성된 QA 데이터는 JSON 파일로 저장
- **5. QA 벡터 데이터베이스 삭제** : 새로 생성된 QA 데이터가 중복되지 않도록 기존 벡터DB에 저장된 QA 중 특정 태그를 가진 문서를 삭제 (메타 필터링)
- **6. QA 벡터 DB에 저장** : 새로 생성된 QA 데이터를 QA 벡터DB에 저장

○ 사내 내부 문서 OPENAI API 생성 흐름

- **문서 사양 정의** : 부서별로 15개씩 카테고리·제목을 리스트에 정리
- **프롬프트 생성** : 각 문서 사양에 맞춰 실행 지침과 섹션 구조를 포함한 프롬프트 작성
- **OpenAI API 호출** : 프롬프트를 넣어 문서 본문을 생성
- **파일 저장** : 생성된 문서를 직급/제목 규칙에 맞게 .txt 파일로 저장
- **인덱스 집계** : 생성된 파일 목록을 INDEX.txt로 정리
- **반복 구조** : 부서별(4개) × 문서별(15개) 루프를 돌며 전체 60개 문서를 자동 생성

○ 사내 내부 문서 OPENAI API 생성 프롬프트 코드

```
63 def make_user_prompt(category: str, title: str, today: str) -> str:
64     base = f"""\# {title}
65     (분류: {category}) | 회사: CodeNova | 버전: v1.0 | 작성일: {today}
66
67     ---
68     **작성 지침**
69     - 언어: 한국어, Markdown 형식
70     - 분량: A4 1장(약 550~750단어) 내외
71     - CTO가 전략·리스크 관점에서 참고할 수 있도록 작성
72     - 실행 단계/체크리스트/의사결정 포인트 포함
73     """
74     if category == "인사·조직 기밀":
75         body = dedent(
76             """
77             ## 포함 섹션
78             1. 개요 및 목적
79             2. 적용 범위/대상
80             3. 실행 계획 (단계별)
81             4. 리스크 및 대응 방안
82             5. 검증 포인트 (CTO 관점)
83             6. 기밀 유지 지침
84             7. 개정 이력 (v1.0 – 오늘)
85             """
86         ).strip()
```

### 3. 데이터 설명 및 구성

#### ✓ 파일 및 필드 설명

- 구글 API 문서 : txt 파일 (문서 상단에 Source URL)

Source URL: <https://cloud.google.com/firestore/docs/security/rules-structure?hl=ko>  
Title: 보안 규칙 구조화

의견 보내기

컬렉션들 사용해 정리하기

내 환경설정들 기준으로 콘텐츠들 저장하고 분류하세요.

보안 규칙 구조화

Firestore 보안 규칙을 통해 데이터베이스의 문서와 컬렉션에 대한 액세스를 제어할 수 있습니다. 유연한 규칙 구문을 사용하면 전체 데이터베이스에 대한 모든 쓰기 작업부터 특정 문서에 대한 작업까지 어떠한 상황에 맞는 규칙이라도 작성할 수 있습니다.

이 가이드에서는 보안 규칙의 기본적인 구문과 구조를 설명합니다. 이 구문과 보안 규칙 조건 (<https://cloud.google.com/firestore/native/docs/security/rules-conditions?hl=ko>)을 결합하면 완전한 규칙 세트가 생성됩니다.

참고: 서버 클라이언트 라이브러리는 모든 Firestore 보안 규칙을 우회하고 대신 Google 애플리케이션 기본 사용자 및 권한 정보 (<https://cloud.google.com/docs/authentication/production?hl=ko>)를 통해 인증합니다. 서버 클라이언트 라이브러리, REST 또는 RPC API를 사용하는 경우 Firestore의 Identity and Access Management (IAM) (<https://cloud.google.com/firestore/docs/security/iam?hl=ko>)를 설정해야 합니다.

서비스 및 데이터베이스 선언

Firestore 보안 규칙은 항상 선언으로 시작됩니다.

```
service cloud.firestore {
  match /databases/{database}/documents {
    allow write: if <condition>;
    service cloud.firestore {
      match /databases/{database}/documents {
```

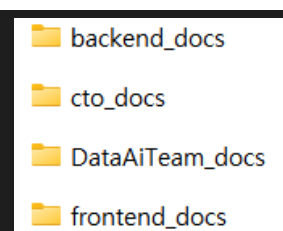
Firestore의 데이터는 문서 컬렉션으로 정리되며, 각 문서는 하위 컬렉션을 통해 계층구조를 이룰 수 있습니다. 계층적 데이터에 보안 규칙이 어떻게 적용되는지 이해하는 것이 중요합니다.

- 구글 API QA 문서

파일명 또는 테이블명	필드명	데이터 타입	설명	예시
Google API Docs	-	TEXT	원문	
Google API Docs QA	question	string	질문 텍스트	Firestore 감사 로그에서 요청 호출자를 식별하기 위해 어떤 필드를 참조해야 하나요?
	answer	string	답변 텍스트	요청 호출자를 식별하기 위해 `AuditLog` 객체 내의 `AuthenticationInfo` 필드를 참조해야 하며, 여기에는 사용자의 `principalEmail`이 포함될 수 있습니다.
	source	array of string	출처 URL 리스트	["https://cloud.google.com/firestore/docs/audit-logging?hl=ko"]
	tags	string	태그(주제/분류)	firebase
	last_verified	string	최종 검증일	2025-08-19
	source_file	string	원문 파일명	cloud.google.com_firestore_docs_audit-logging_hl=ko.txt

- 사내 내부 문서 : 아래와 같이 직급별 폴더 안에 문서별 txt 파일 존재







1	# 팀 운영 문서   AI팀 주간 업무 계획	
2		
3	작성일: 2025-08-29	
4	회사: CodeNova   대상: 데이터/AI팀	
5		
6	---	
7	# AI팀 주간 업무 계획	
8	(분류: 팀 운영 문서)   회사: CodeNova   버전: v1.0   작성일: 2025-08-29	
9		
10	---	
11		
12	## 1. 이번 주 최우선 과제	
13	목표   성공 기준   담당자   마감일	
14	----- ----- ----- -----	
15	모델 성능 개선   AUC 점수 0.85 이상 달성   이지훈   2025-09-01	
16	데이터 정제 프로세스 확립   정제된 데이터셋 1000건 확보   김하늘   2025-09-02	
17	API 문서화   모든 API 엔드포인트 문서화 완료   박지민   2025-09-03	
18		
19	## 2. 팀원별 주요 작업	
20	### 이지훈	
21	- **작업**: 모델 성능 개선을 위한 하이퍼파라미터 튜닝	
22	- **의존성**: 데이터셋 업데이트 완료 필요	
23	- **리스크**: 성능 개선이 예상보다 낮을 경우 재조정 필요	
24		
25	### 김하늘	
26	- **작업**: 데이터 정제 작업 및 품질 검증	
27	- **의존성**: 기존 데이터셋의 품질 문제 해결 필요	
28	- **리스크**: 정제 과정에서 데이터 손실 발생 가능성	



## ✓ 데이터 양

- 전체 수집 데이터 건수:
  - 구글 API 문서: 약 2000개 문서 (txt 기준)
  - 회사 내부 문서: 60개의 txt 문서 (각 부서별 15개씩)
- 추출된 고품질 데이터 건수 (필터링 후 기준)
  - 구글 API 문서 QA: 약 14800개의 QA 데이터셋 (jsonl)

## ✓ 저장 위치 및 포맷

- 구글 API 문서 & QA, 회사 내부 문서
  - 저장 경로: SKN14-Final-1Team
  - 저장 포맷: txt(문서) / JSONL(QA) / Vector DB (임베딩)
  - 인코딩: UTF-8
- Chroma Vector DB
  - [구글 API 문서 & QA]
    - 구글 API 문서:  chroma\_text\_api\_final
    - 구글 API QA:  chroma\_qa\_db
  - [회사 내부 문서]
    - 프론트팀:  com\_front\_chroma\_db
    - 백엔드팀:  com\_backend\_chroma\_db
    - Data&AI팀:  com\_data\_ai\_chroma\_db
    - CTO:  com\_cto\_vector\_db \

## 5. 법적·윤리적 검토

- 개인정보 포함 여부
  - 미포함 (Google API 문서는 기술 문서로, 개인 식별 정보 없음)
- 비식별화 조치 여부
  - 해당없음 (개인정보가 없으므로 별도의 비식별화 불필요)
- 출처 및 사용권
  - Google API 공식 문서 활용
  - 원문 그대로 재배포하지 않고 QA 데이터셋으로 가공하여 내부 연구/교육 목적으로만 사용
- 공개 여부
  - 내부사용 한정
- 라이선스 또는 약관 검토 여부
  - Google 개발자 문서 이용 약관 및 robots.txt 확인 완료
  - 허용된 범위 내에서만 수집 및 활용