챗GPT 러닝데이 & MS애저톤

#무료 #유튜브라이브

Open API를 활용한 연구원의 업무 효율화

활용

4월 25일(화)19:00

MS MVP AI **이제현**

한국에너지기술연구원

에이아이프렌즈학회

연구원의 삶

PHD Comics: The Research Cycle

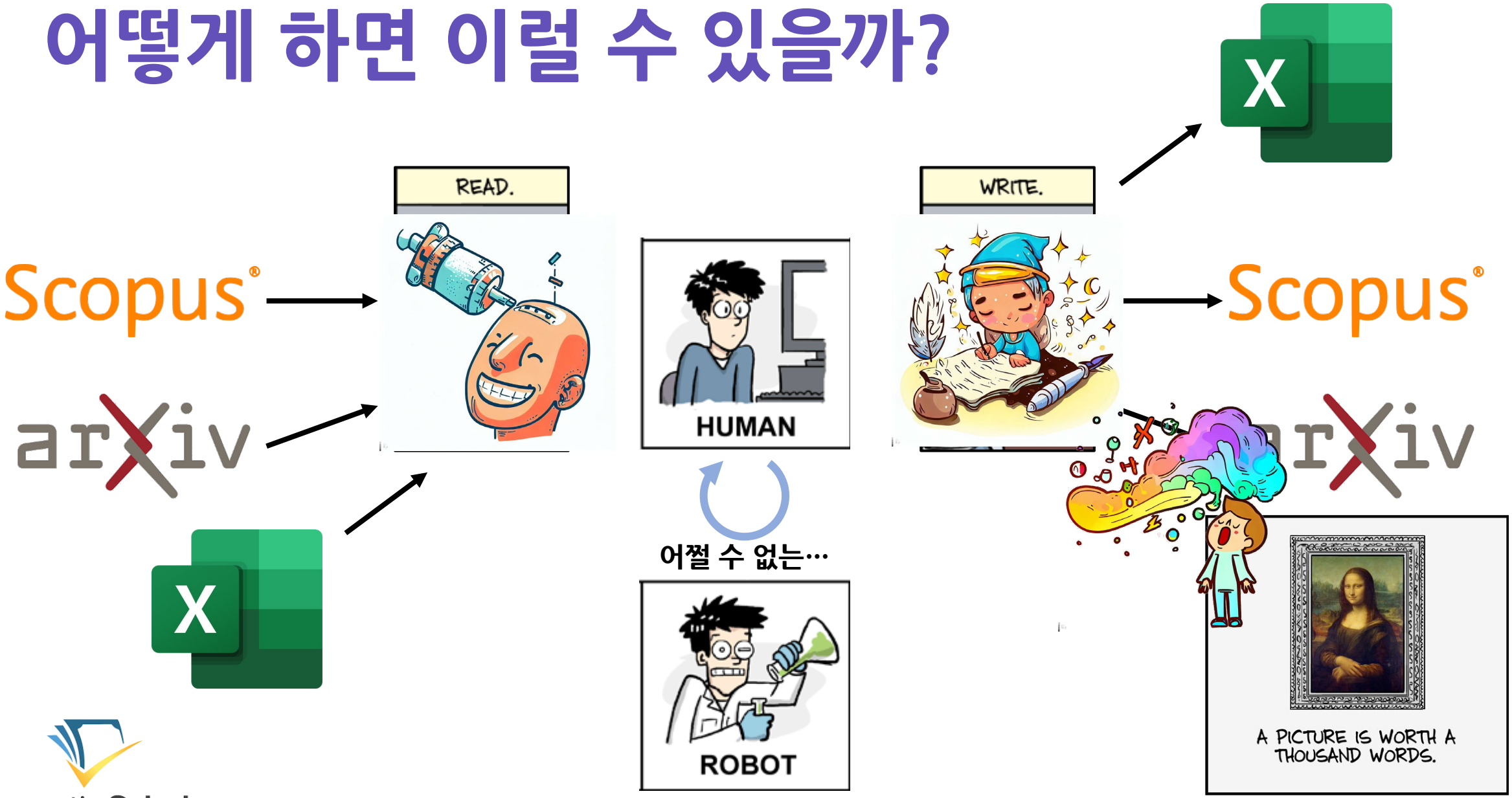
식사

안주

이러고 싶다

Bing image creator,
“cartoon of a machine injecting new knowledges on a smiling person's head”
“a kid-style cartoon of sleeping man, while a magic quill is transferring his knowledge onto a sheet of paper”
“a kid-style cartoon of man generating a rainbow-colorful chemical illustrations from his deep exhale breath”

어떻게 하면 이럴 수 있을까?



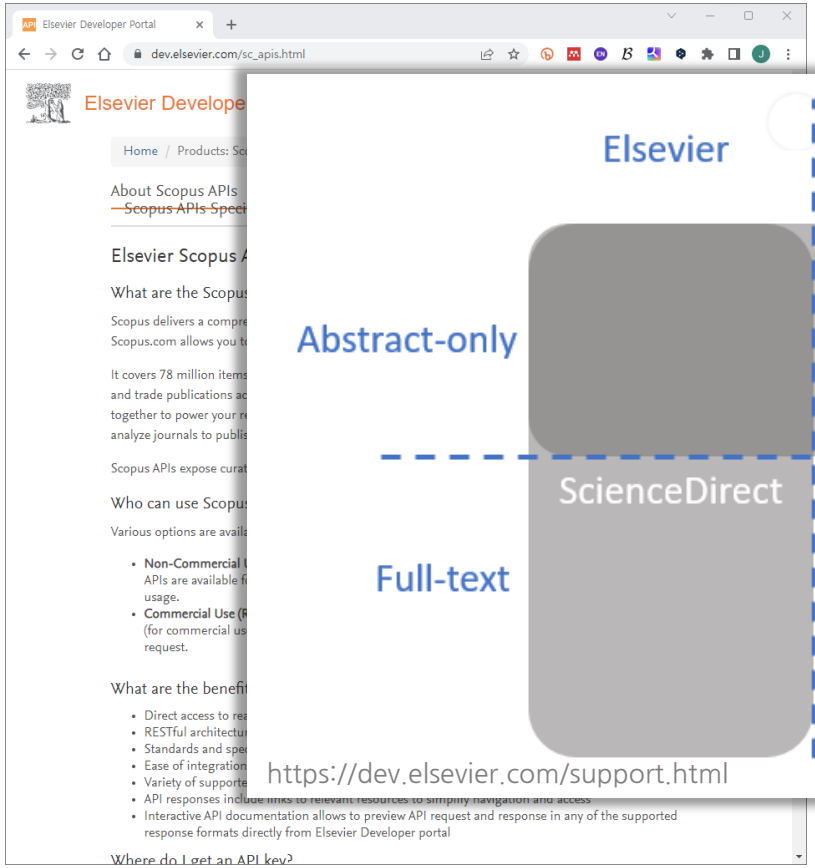
Semantic Scholar

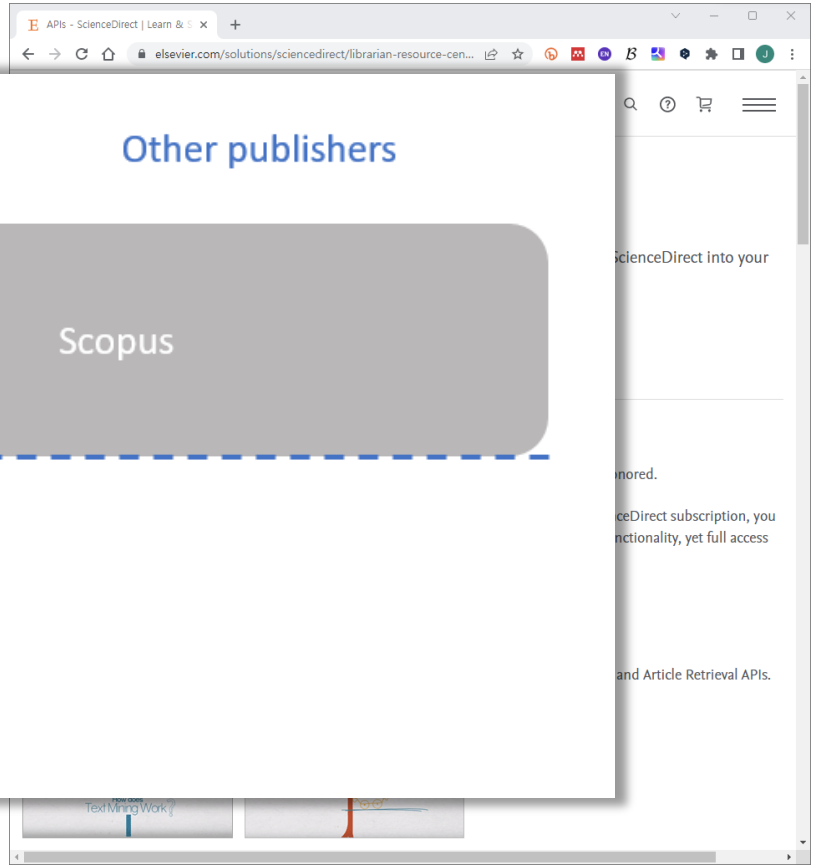
PHD Comics: Grad School Energy Levels

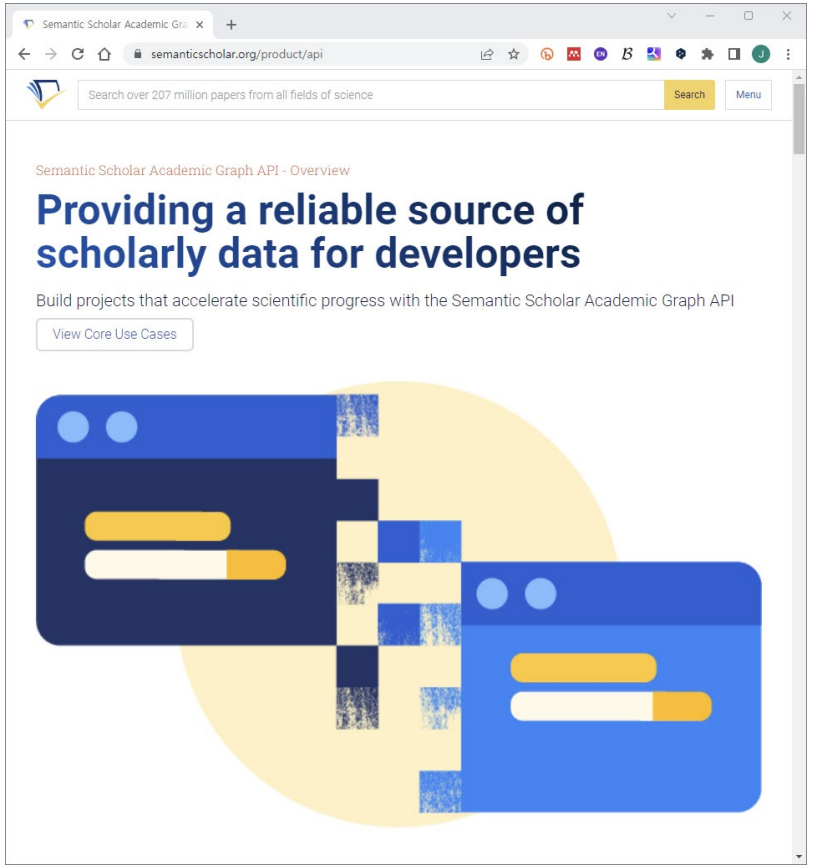
PHD Comics: Truism

Open API : Scientific Literatures

- **Scopus**
  - https://dev.elsevier.com/
- **ScienceDirect**
  - https://www.elsevier.com/solutions/sciencedirect/librarian-resource-center/api
- **Semantic Scholar**
  - https://www.semanticscholar.org/product/api

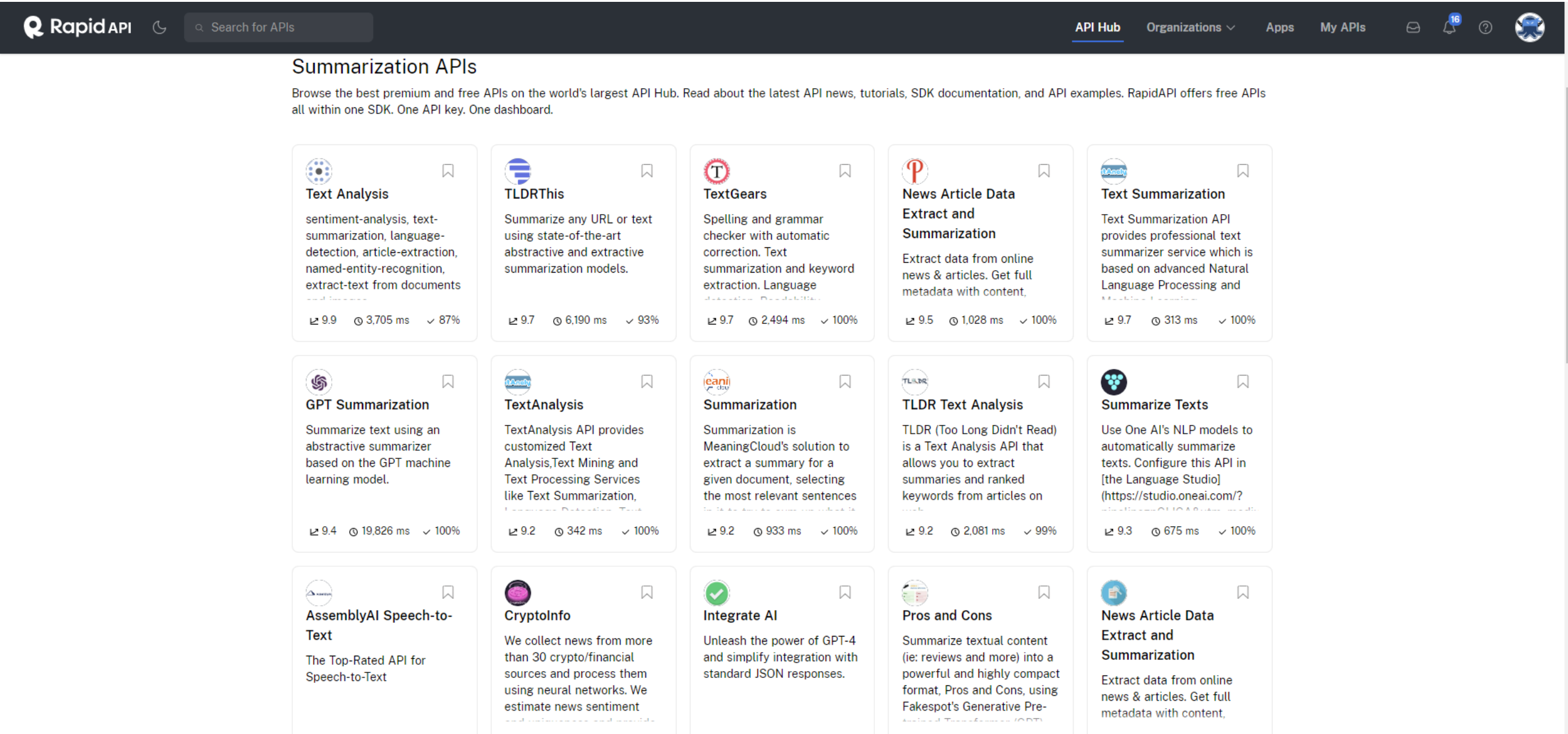




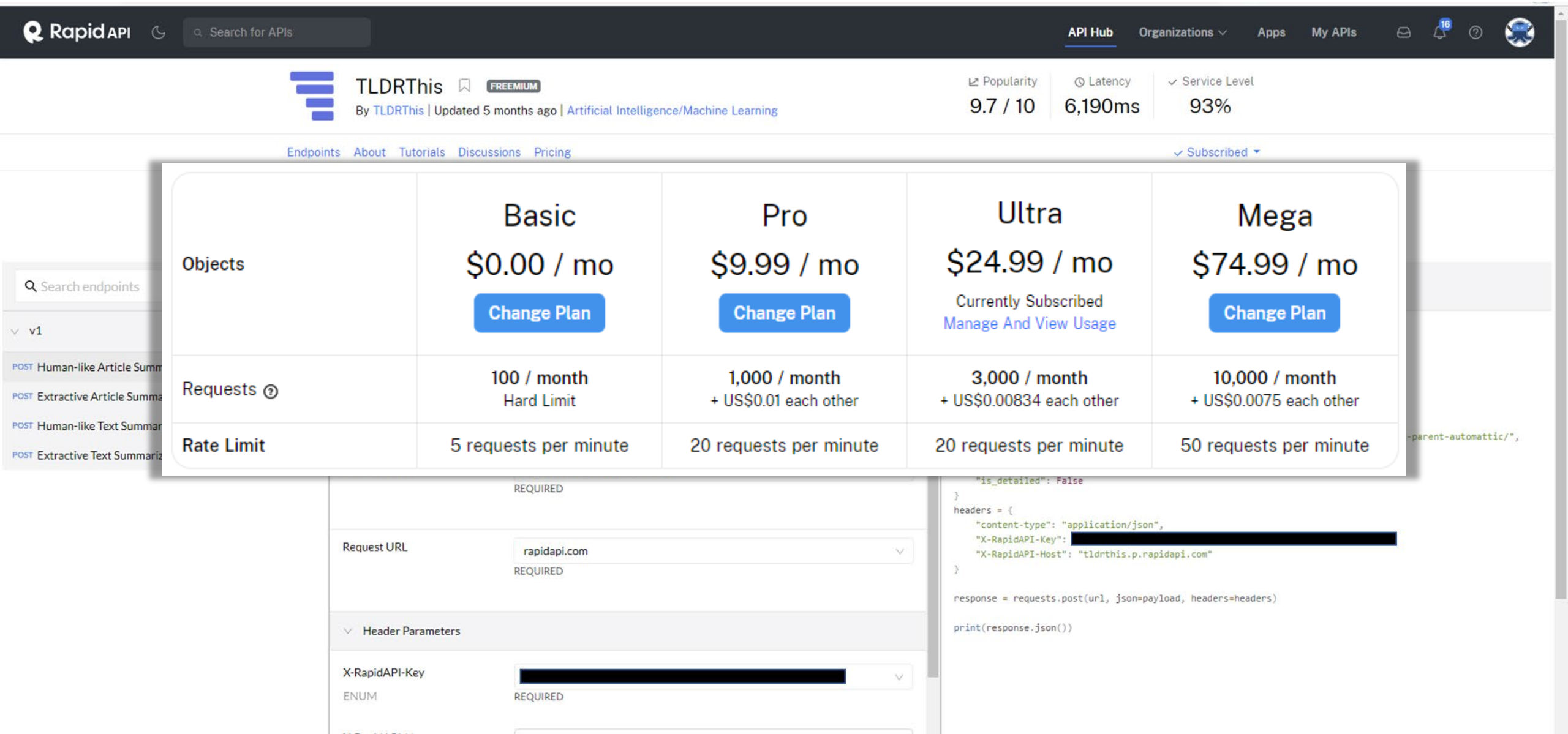


Rapid API : https://rapidapi.com/

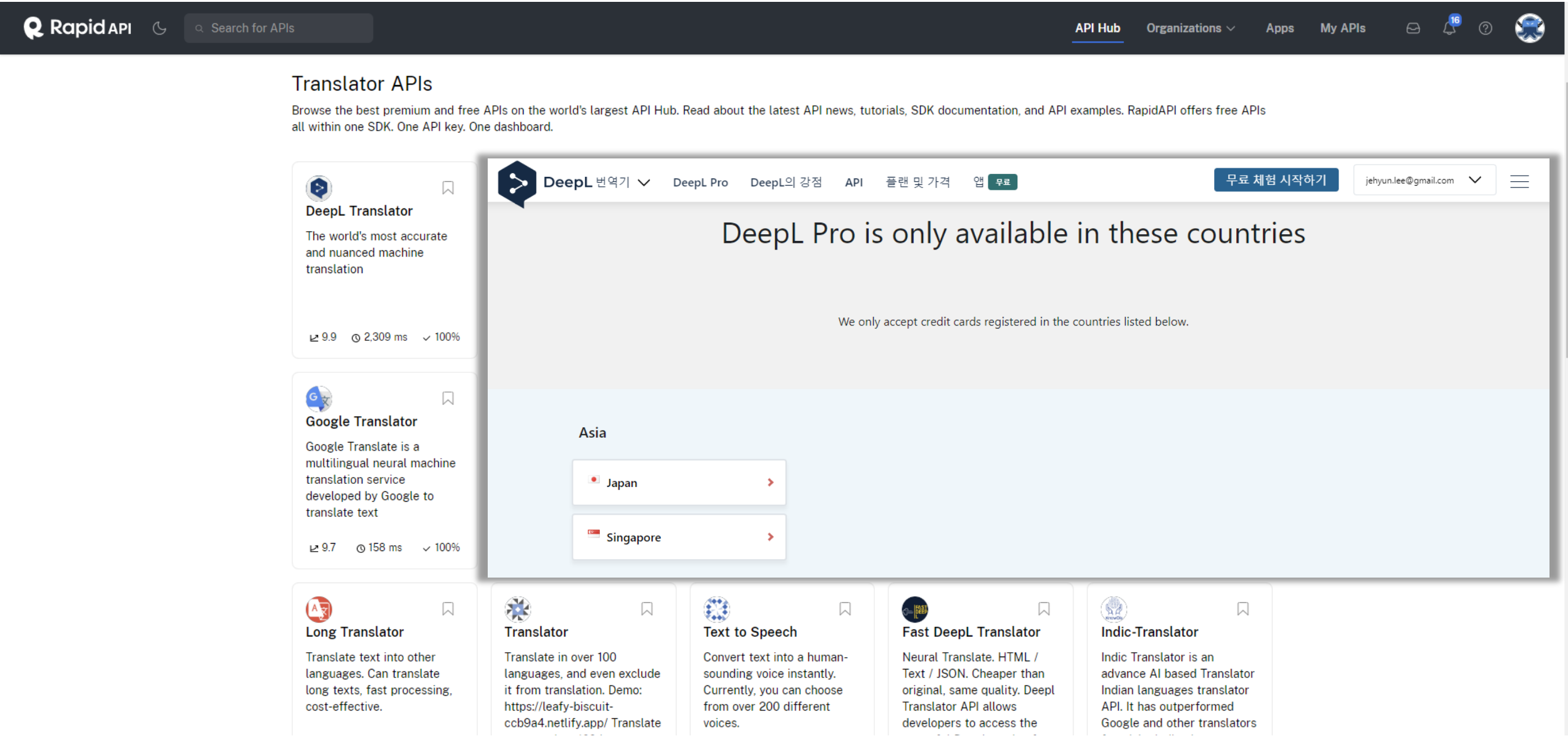
Open API : Summarization & Translation



Open API : Summarization & Translation



Open API : Summarization & Translation



Open API : ETRI

ETRI AI API, Data : https://aiopen.etri.re.kr/

AI API·DATA

서비스 안내 | API 개발가이드 | API 데모 | 모델 및 데이터 | 고객지원 | API Key

- 사전 준비사항
- 사이트 소개
- 정책 및 약관

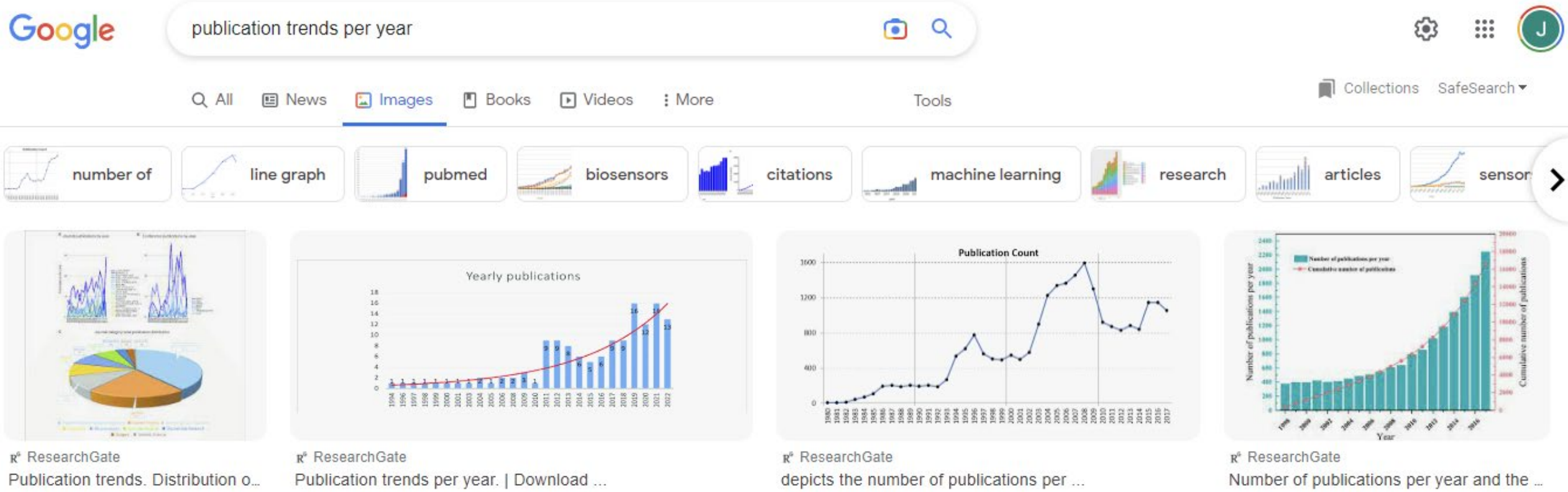
오픈API 목록

기술명	API명		1일 허용량
언어 분석 기술 (문어)	· 형태소 분석 API · 동음이의어 분석 API · 의존 구문분석 API	· 개체명 인식 API · 다의어 분석 API · 의미역 인식 API	5,000건/일 (1회 사용 시 입력은 1만 글자 이하)
언어 분석 기술 (구어)	· 형태소 분석 API	· 개체명 인식 API	5,000건/일 (1회 사용 시 입력은 1만 글자 이하)
어휘관계 분석 기술	· 문장 패러프레이즈 인식 API · 동음이의어 정보 API · 어휘간 유사도 분석 API · 상호참조 해결 API	· 어휘 정보 API · 다의어 정보 API · 개체 연결 API	5,000건/일
질의응답 기술	· 질문분석 API · 위키백과 QA API · 행정문서 QA API	· 기계독해 API · 법률 QA API	5,000건/일 (1회 사용 시 입력은 1만 글자 이하)
	· 한국어 인식 API	· 영어 인식 API	1,000건/일 (최대 20초/건당)
	· 중국어 인식 API	· 일본어 인식 API	

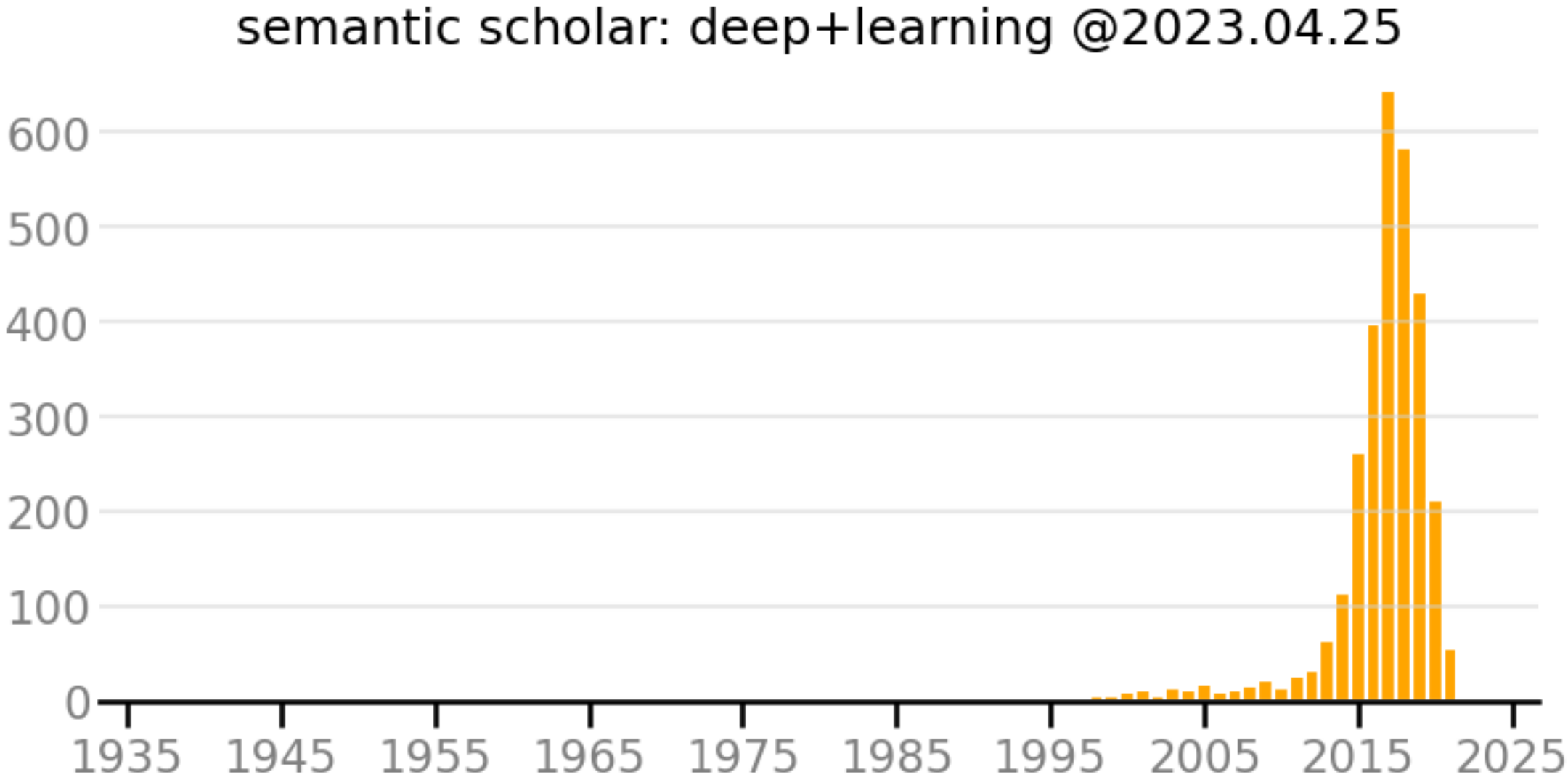
실습 Google Colab

- https://bit.ly/3HaUIgy

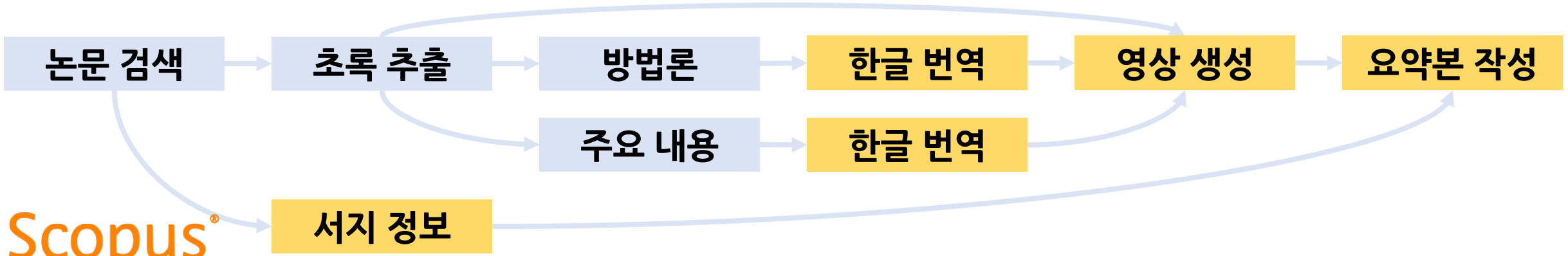
1. 년도별 출판 건수 시각화



1. 년도별 출판 건수 시각화



2. 논문 요약본 생성



Scopus®

라이선스 필요

Semantic Scholar

라이선스 불필요

Semantic Scholar

arXiv

라이선스 불필요

OpenAI

DALL·E 2

2. 논문 요약본 생성

230415_애저톤.ipynb

파일 수정 보기 삽입 런타임 도구 도움말 모든 변경사항이 저장됨

목차

1. 라이브러리 설치
2. 기능 구현
   - 2-1. 논문 검색 (semantic scholar)
     - 논문 출판 동향 (Matplotlib)
   - 2-2. 초록 요약, 질의 응답 (langchain)
     - 일괄 초록 요약, 질의 응답 (pd.Series.apply)
   - 2-3. 한글 번역 (Google Translator)
   - 2-4. 한글 번역 (DeepL)
     - 일괄 방법론, 메시지 번역 (pd.Series.apply)
3. 보고서 생성 (python-docx)
   - 3-1. hyperlink 삽입 코드
   - 3-2. 방법론과 메시지를 함축한 그림 생성 (DALL.E)
   - **3-3. MS Word 보고서 생성**

섹션

+ 코드 + 텍스트

[46] 31 fig = get_figure(methodology, message)

3-3. MS Word 보고서 생성

[48]
```
1 from docx import Document
2 from docx.shared import Inches
3 from docx.shared import Pt
4 from docx.shared import RGBColor
5 from docx.enum.text import WD_ALIGN_PARAGRAPH
6
7 # document generation
8 document = Document()
9
10 # Title
11 report_title = f"semantic scholar @{datetime.strftime(datetime.now(), '%Y.%m.%d')}:\n{keywords}"
12 document.add_heading(f'{report_title}', 0)
13 document.add_page_break()
14
15 # Each Article
16 for i in range(df_paper_abstract20.shape[0]):
17   # article data
18   url = df_paper_abstract20["urls"].iloc[i]
19   title = df_paper_abstract20["title"].iloc[i]
20   journal = df_paper_abstract20["journals"].iloc[i]
21   volume = df_paper_abstract20["volumes"].iloc[i]
22   pages = df_paper_abstract20["pages"].iloc[i]
23   abstract = df_paper_abstract20["abstracts"].iloc[i]
24
25   methodology_ko = df_paper_abstract20["methodology_ko"].iloc[i]
26   methodology = df_paper_abstract20["methodology"].iloc[i]
27   message_ko = df_paper_abstract20["message_ko"].iloc[i]
28   message = df_paper_abstract20["message"].iloc[i]
29
30   # title
31   title_ = document.add_paragraph(style='List Number').add_run(f'"{title}"')
32   title_.font.size=Pt(14)
33   title_.font.italic=True
34
35   # journal
36   if journal:
```

3. *"Prediction of Heart Disease Using a Combination of Machine Learning and Deep Learning"*↵

   Computational Intelligence and Neuroscience (2021)↵

- https://www.semanticscholar.org/paper/31cf4c96c5dd4ac5a6bbb4ac7b6bab763651624a↵

- **방법론 (국문) :** 이 글의 방법론은 다양한 머신러닝 알고리즘과 딥러닝을 UCI 머신러닝 심장 질환 데이터 세트에 적용하고, 격리 포레스트(Isolation Forest)를 사용하여 관련 없는 특징을 처리하고, 데이터를 정규화하고, 정확도 및 혼동 행렬을 사용하여 결과를 검증하는 것입니다.↵
- 방법론 (영문) : The methodology in this article is to apply different machine learning algorithms and deep learning to the UCI Machine Learning Heart Disease dataset, handle irrelevant features using Isolation Forest, normalize the data, and validate the results using accuracy and confusion matrix.↵
- **주요내용 (국문) :** 이 글의 주요 메시지는 머신러닝 알고리즘과 딥러닝을 사용하여 UCI 머신러닝 심장병 데이터 세트를 분석하여 심장병을 예측하고 94.2%의 정확도로 유망한 결과를 얻을 수 있다는 것입니다.↵
- 주요내용 (영문) : The main message of this article is that machine learning algorithms and deep learning can be used to analyze the UCI Machine Learning Heart Disease dataset in order to predict heart disease and achieve promising results with an accuracy of 94.2%.↵
- ABSTRACT : The correct prediction of heart disease can prevent life threats, and incorrect prediction can prove to be fatal at the same time. In this paper different machine learning