

AI 활용도 200% 향상 #32

Google Colab으로 논문에 사용할 그래프 그리기

이봉우 (2026.01.23)

논문을 작성할 때 그래프를 이용하여 다양한 시각적 표현을 하곤 합니다. 엑셀을 이용해서 그림을 그릴 수 있겠고, Origin같은 프로그램을 이용해서 그래프를 그리기도 합니다.

이전에 AI사용팁 13, 14에서도 언급하였듯이 AI를 이용한 노코드 데이터 분석을 이용하여 그래프를 그릴 수도 있습니다. 문제는 조금 수정하려고 해도 처음부터 다시 시행해야 하는 어려움이 있습니다.

코딩을 잘 하는 분들께서는 직접 R이나 Python으로 그림을 그리는데, 최근에는 코딩 프로그램들에도 AI가 연결되어 쉽게 작업을 할 수 있습니다.

그렇지만 코딩에 익숙하지 않은 분들께서는 처음의 진입장벽을 넘어서는 어려움이 있을 수 있어 이번 문서에는 Google Colab을 이용하여 그래프를 그리는 방법을 익혀보도록 하겠습니다.

* 목표 : 주어진 데이터를 이용하여 논문에 사용할 그래프 그리기

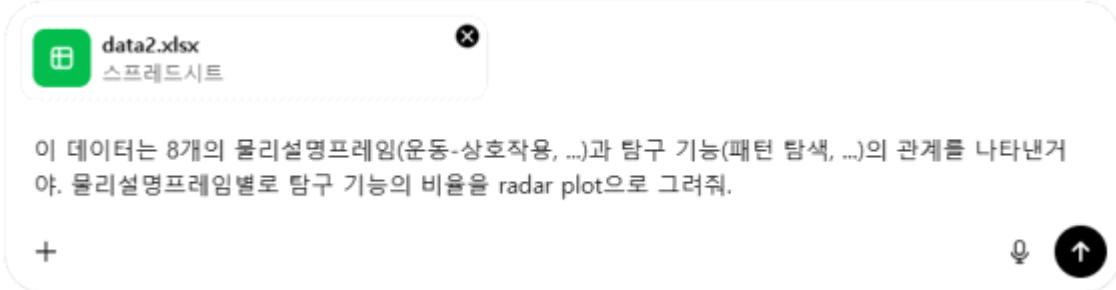
- 데이터는 제가 최근에 연구했던 결과입니다. 논문에 있는 그림을 물리 설명 프레임(8가지)과 탐구 기능(8가지)를 분석하고, 이 두 변인간의 관련성을 알아보기 위해 표로 만든 것입니다.

물리설명 프레임	탐구 기능	패턴 탐색	조건 비교	관계 규명	스케일 링	파라미 터 추정	불확실 성	분류/상	메커니 즘
운동-상호작용	15	19	21	0	6	2	6	6	
에너지 전이·변환	4	4	7	3	4	0	8	4	
공간적 분포·연속성	10	11	11	5	9	3	7	9	
주기성·전파	47	44	41	7	31	13	23	25	
매개된 상호작용	43	41	43	7	34	25	28	37	
통계적 집합행동	82	85	89	29	56	44	63	55	
구조-상태 대응	34	34	34	2	19	12	25	19	
모형 기반 비직관적	83	75	89	21	67	43	75	3	
합	318	313	335	74	226	142	235	238	

- 이 데이터로 물리설명프레임 별로 탐구 기능의 비율이 어떻게 되는지 radar plot을 만들 어보겠습니다.
- 비율을 이용해야 하기 때문에 데이터는 개수가 아니라 비율로 다시 만들어야 하지만, 우리는 AI를 이용할 것이기 때문에 원 소스데이터로 출발해도 됩니다.

■ 1단계: LLM에서 직접 그리기

- 가장 쉬운 단계로 LLM (ChatGPT, Gemini 등)에서 직접 그려달라고 해보겠습니다.



- 한글이 표시되지 않으면 이전에 했던 것과 같은 방법으로 하면 됩니다.



- 그러면 잘 그려줍니다. 이후의 과정도 다음과 같이 제안해줍니다.

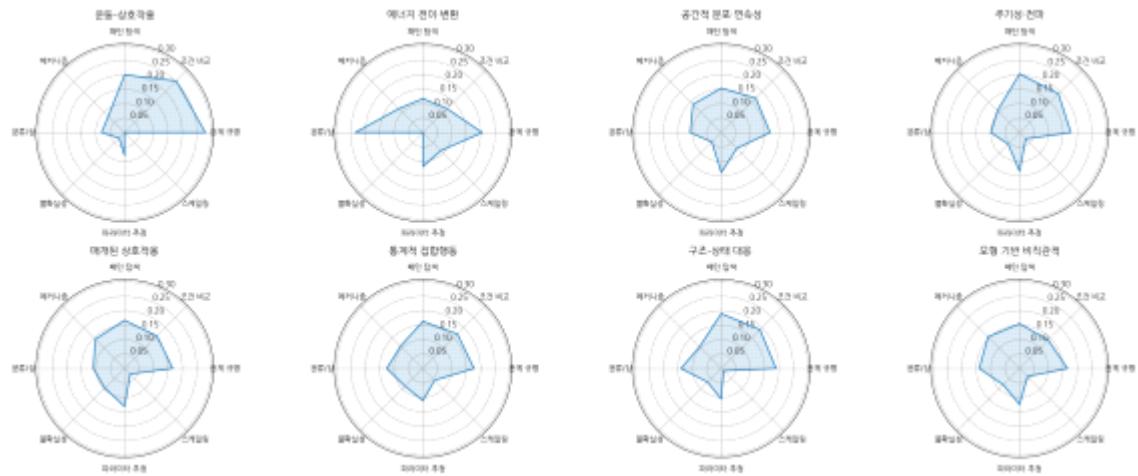
5 다음으로 도와드릴 수 있는 것

원하시면 바로 이어서:

- radar plot 논문용 스타일(단일 Figure, 서브플롯 2x4)
- 색상·투명도·축 정규화 통일(프레임 간 비교 최적화)
- Figure caption + 본문 해석 문단 (APA 스탠다드)
- 탐구 기능 다양성 지수 계산 코드

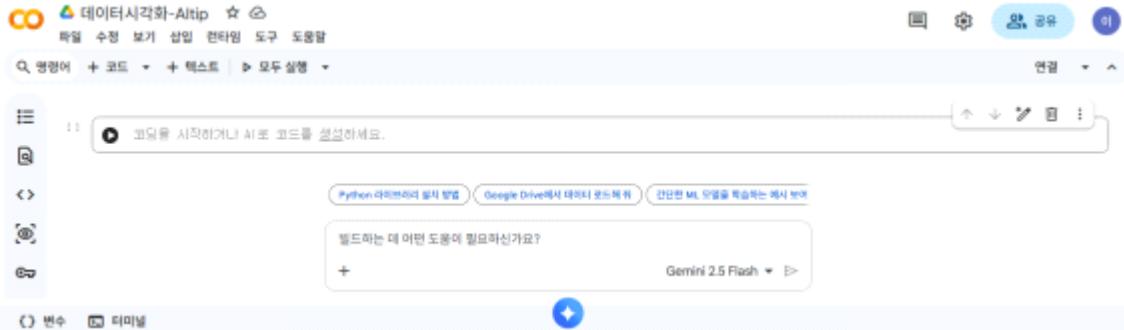
중에서 바로 이어서 도와드릴게요. ↗

- 원하는 형태가 나오도록 추가로 수정을 해달라고 하면 됩니다. 이와 같은 방법을 이용해도 되지만, 우리는 google colab을 이용해서 직접(아니 AI에게 시켜서) 코딩을 해보겠습니다.



■ 2단계: Google colab을 이용하기

google colab으로 접속해서 새 코드를 만들기 시작하는 것까지는 생략하겠습니다. 예전에는 바로 코드를 입력하는 창이 나왔는데, 이제는 아래에 AI와의 대화창이 나옵니다.



① 처음 시작하기

- 처음에 아무것도 없으면 막막하죠? google colab에서 AI의 도움으로 시작할 수도 있지만, 처음 시행하시는 분들은 중간에 오류가 나오면 진행하기 어려울 수 있으니까 처음은 LLM으로 시작하죠. 다음과 같이 ChatGPT에게 물어봅니다.

google colab에서 엑셀 파일을 업로드하여 이것으로 시각화하는 코드를 만들고 싶어. 처음 부분을 만들어줘!

+ Google 드라이브



- 그러면 코드를 잘 만들어줍니다. 아예 시각화 예시도 만들어주세요. 우리는 이중에서 엑셀 파일을 업로드하는 부분까지만 복사하겠습니다. 주석을 제외한 부분만 옮기면 다음과 같습니다.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# =====
# 1) 엑셀 파일 업로드
# =====
from google.colab import files
uploaded = files.upload() # 업로드 창이 뜹니다.
excel_filename = list(uploaded.keys())[0]
print("업로드된 파일:", excel_filename)

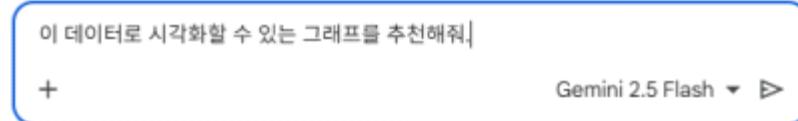
# =====
# 2) 엑셀 읽기 (시트 선택 가능)
# =====
xls = pd.ExcelFile(excel_filename, engine="openpyxl")
print("시트 목록:", xls.sheet_names)

df = pd.read_excel(excel_filename, sheet_name=xls.sheet_names[0], engine="openpyxl")
```

- 이 부분을 google colab에 복사한 이후에 실행 버튼(▶)을 클릭합니다.

② 그래프 선택하기

- 우리는 무엇을 그릴 것인지 이미 정해 놓았지만, 혹시나 싶어 어떤 그래프가 좋을지 물어봤습니다. 그랬더니 막대그래프와 히트맵을 추천하고, 바로 막대그래프 코드를 만들어 주었습니다.
- 파이썬에는 matplotlib 또는 seaborn과 같은 좋은 그래프 라이브러리가 있습니다. 이곳을 찾아보시면 어떤 그래프가 가능한지도 확인할 수 있으니 한번 꼭 보시기를 추천합니다.

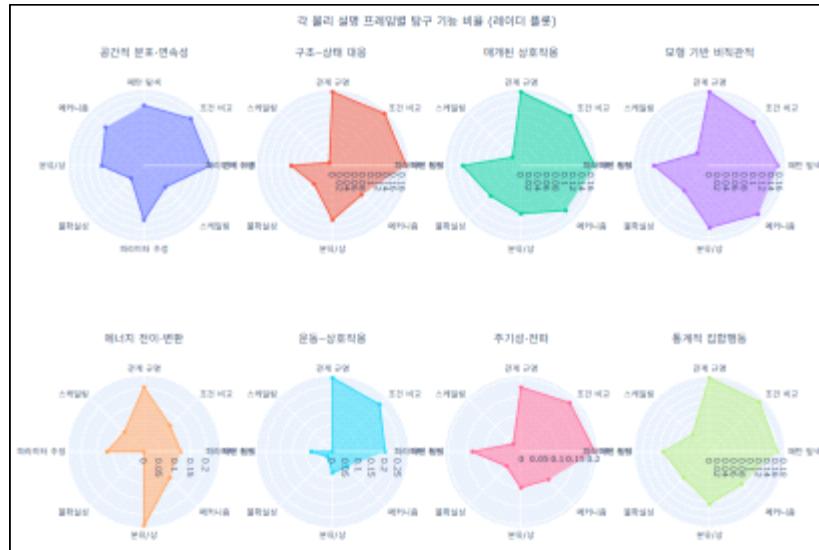
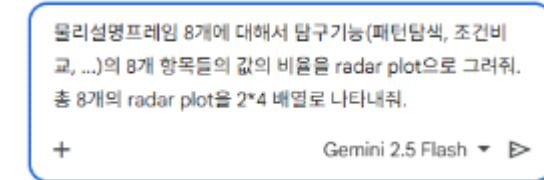


- 스택형 막대 그래프:** 각 '물리 설명 프레임'별로 모든 수치형 열의 합계 및 구성을 한 번에 비교할 수 있습니다.
- 히트맵:** '물리 설명 프레임'과 모든 수치형 열 간의 관계를 색상 강도로 시각화하여 패턴을 파악할 수 있습니다.

- 그런데 한글이 잘 나오지 않습니다. Colab이 스스로 한글이 잘 안나온다는 것을 인지하고 스스로 코드를 수정해 주기도 합니다. 사실 한글 구현이 계속 문제가 될 수 있습니다. 런타임을 다시 시작하라는 명령이 나오기도 하는데, 메뉴를 보면 '런타임'이 있습니다.
- 한글 구현은 시스템마다 달라서 직접 해보시면서 오류가 생기면 계속 부탁하면 됩니다. 자기가 계속 문제를 해결해가면서 결과를 보여주니까 충분히 원하시는 한글 구현을 할 수 있으리라 생각합니다. 혹시 어려움이 있으면, 일단 이번 문서는 그래프 그리기가 목적이니까 영어로 바꿔서(AI에게 바꿔달라고 하세요) 그래프를 그려보세요.

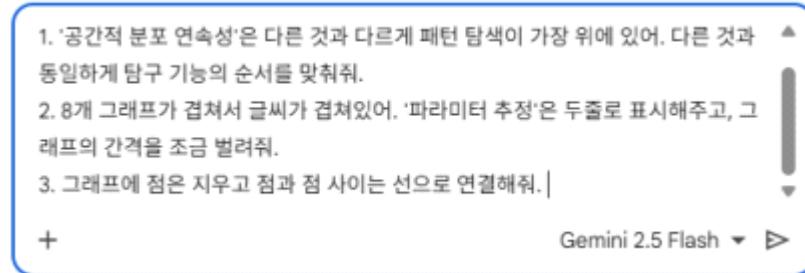
③ 레이더 플롯 그리기

- 어느 정도 나오기는 하는데, 몇 가지 수정할 부분이 보이네요.



④ 수정하기 1

- 수정하는 방법은 크게 두 가지입니다. AI와 대화하면서 수정하는 방법으로 먼저 해보겠습니다.



⑤ 수정하기 2

- 수정을 요청하면 어떻게 수정했는지 화면에 보여줍니다. 이 값을 보고 직접 수정하면 됩니다. 물론 파이썬 코드를 알면 바로 할 수 있겠지만, 주석을 보면서 수행하면 어렵지 않습니다.
- AI에게 대화를 통해 수정을 해보면 어떤 부분을 수정했는지 보여주는데, 그것을 보면 눈치를챌 수도 있습니다. 예로 저는 레이더 범위를 0.3까지, 간격도 0.3까지 표시하도록 수정해보았습니다.

```
range=[0, 0.3], # 레이더 범위 설정  
ticknode='array',  
tickvals=[0, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3], # 특정 간격으로 레이블 표시
```

⑥ 최종 완성본

