Topic : 用AI (GPT)協助演講者編講稿

Project Overview :

隨著openAI 的發展，我們希望能有機會

Key Objective :

提高resume競爭力、

練習openai , MLP model實作、

訓練團隊作業跟協調能力。

Architecture：

Part A : 著重在建prompt

抓slide上的資料，並試著組建擬定講稿的Prompt (Model)後，丟給GPT API (openai Library) 並拿到AI協助的講稿。

Part B : 著重在訓練下prompt

優化Part 1的 prompt – 利用MLP MODEL (stable\_baselines3 Library)去學習如何下prompt

Model, Env –

創照一個環境(Env)，此環境會有多位評審(Reward)，這些評審是由GPT模擬出來的。

接著使用ChatGPT常使用的英文連接詞(Action Taken)去拼湊建議講稿的prompt，最後將此prompt丟給ChatGPT並拿到新講稿。

然後在使用評審的prompt去評估分數及要修改的點，然候總結多位評審得建議並改善講稿(Observation)。

依據得分的高低去學習如何下此專案slide的prompt。

最後測試此MLP Model 是否穩定。

FrameWork & Spec :

MacBook M1 chip,

AWS(EC2 - MySQL),

Python,

GPT API

Step:

Part A :

用pptx 去搜集slide上的資料

用pandas去存資料，並建立指定的格式

研究prompt (OpenAI API) 的基礎模型，要如何排版才能得好的結果。

並搜集回傳的值 (Excel? , DataBase?)

Part B :

用stable\_baselines3 Library去建立一個Environment，並訓練修改script的prompt。

Environment & Training –

Action Taken :

使用一個script模板，並在其中放入不同的字串 (英文單字)，然後傳給GPT模擬的多位評審評分。(2~3位，各為感興趣的, 不感興趣的, 中立的)

Reward :

然後傳給GPT模擬的多位評審評分。(2~3位)

各個評審分數變高為正獎勵，變低為負獎勵。

Observation :

查看各個評審分數，並總結各個評審對可以改進script的review。

improve後再丟回給GPT。

Done :

此Script達到一定成績。

Testing –

用Gym 去進行步數測試，測試訓練前及訓練後，每步內給的評分有沒有變高。