CSIT5900 - Artificial Intelligence

Project Description

Objectives

To give you a first-hand experience in using generative AI to solve non-trivial problems.

Project Description

This is a group project. By default you need to work as a group of two. Contact us if you cannot find a partner.

Your group will have to select one of the tasks described below to work on. You'll need to submit a copy of your slides that you'll present in class. If you solve the task by a Python code that uses generative AI tools by their APIs, then you'll also need to submit your code. On the other hand, if you solve the task manually with the help of some generative AI tools, like prompt some LLMs through their interactive web interface, then you need to submit a sequence of screen shots showing the entire process of how you solve the task. See below for more details and the due dates.

Generative AI tools

You can use any generative AI tools that you can find. Our university provides such of these tools on https://genai.ust.hk. Furthermore, all OpenAI models can be accessed by APIs on Microsoft Azure platform: HKUST GenAI API.

Tasks

You should choose one of the following three. To avoid everyone working on the same task, we will limit each task to at most 15 groups. We will do the task assignment based on your preferences. See below for when to send in your preferences and how the task assignment is done.

Task 1

This is a modified problem of the Missionary and Cannibals Problem:

30 sisters and 30 brothers must cross a river using a boat which can carry at most four people, under the constraint that, for both banks, if there are sisters present on the bank, they cannot be out-numbered by brothers. How do they cross the river?

To solve the problem, you can only use the above description as input to your generative AI tools. You can use general prompts such as "solve the problem step by step". You should not suggest solutions such as "start by moving 2 brothers and 2 sisters to the other side, then bring 2 sisters back and so on". The final solution needs to be a sequence of moves.

Task 2

Ten Students are working on a group project. They want to find some time to meet. They have the following constraints (morning means from 8 AM to 12 PM, and afternoon means from 1 PM to 5PM):

Student A: I have classes on Monday and Wednesday from 2 PM to 4 PM, club activities on Tuesday and Thursday evenings from 7 PM to 9 PM, and I'm available on Friday afternoons.
Student B: I have a lab on Monday from 9 AM to 11 AM, classes on Tuesday and Thursday afternoons from 1 PM to 3 PM, and I'm free all day on Friday, but I might have to work late in the evening after 10 PM.
Student C: I have classes every weekday morning from 8 AM to 10 AM. I may have to work on weekends from 3 PM to 5 PM. If A calls me on Friday morning, we'll go play tennis in the afternoon 6 PM – 8 PM. I'm free otherwise.
Student D: I have classes on Monday and Friday from 10 AM to 12 PM, commitments on Tuesday afternoons from 3 PM to 5 PM.
Student E: I have group discussions on Wednesday evenings from 7 PM to 9 PM, and I participate in volunteer activities on weekends, usually free from 2 PM to 5 PM.
Student F: I have labs every afternoon from 1 PM to 3 PM, and I'm free on Saturday and Sunday mornings, but I might go out on Sunday afternoons. I'm working with student D and student I on another project. We have regular meetings Wednesdays 5 PM to 6 PM. However, we can reschedule our meeting if needed.
Student G: I have classes on Monday and Tuesday evenings from 6 PM to 8 PM, training on Thursday afternoons from 4 PM to 6 PM, and I'm usually free on Wednesday and Friday afternoons.
Student H: I have classes from Monday to Friday from 9 AM to 11 AM, and I usually go to the gym on weekend afternoons from 1 PM to 3 PM.
Student I: I have a meeting on Thursday afternoons from 3 PM to 5 PM; I'm available at other times, but I might travel on the weekend.
Student J: I study every evening from 8 PM to 10 PM, have no classes on Thursday and Sunday afternoons, but I will gather with my family on Saturday.

Now find all possible times between 8 AM and 10 PM each day that at least 8 of them

are free to meet. Again, like for Task 1, you must use the English descriptions as given above as input to your generative AI tools.

Task 3

The distributed video shows someone used a browser to go to HKEX webpage https://www.hkex.com.hk/?sc_lang=en, moused over 'Market Data', typed '1' in "stock quote lookup" search box, selected "CK HUTCHISON HOLDINGS LTD." from the shown result, waited for the new page to load and then clicked on 'Export to Excel' to save the displayed price data as an excel file (the demonstration in the video works on a desktop, and may not work on a mobile device). Your task here is to use some generative AI tools to help you write a Python code to do the same thing: going to the HKEX website to download today's price data for the company "CK HUTCHISON HOLDINGS LTD.". It'll be ideal if you can use your AI tool to write the code using the video and some text prompts. But this may be too difficult if not impossible. For this project, it is okay for you to watch the video yourself, describe what needs to be done in a natural language, and ask your AI tool to write the code based on your description.

Schedule:

October 17: Group information and task preference
Form a group of 2 and sign up on canvas.   Enter your group's preference on tasks in the form of [i, j], meaning Task i is your first preference and Task j the second.
October 20: Task assignment notification
We'll honor your first choice as much as possible. However, if more than 15 groups have the same top choice, then we'll do a random tie-breaking.
November 25: Due date for presentation slides.
November 29: Presentation sessions (attendance required).
December 6: Code or screenshots due.

References:

- Azure Machine Learning documentation
- HKUST ChatGPT
- HKUST ChatGPT API
- Github - Cortex : how to deploy and manage large models on Azure.
- Solving math word problems with ChatGPT and an external solver. https://arxiv.org/pdf/2304.09102.
- Solving reasoning problems with ChatGPT and an external theorem prover. https://arxiv.org/abs/2304.01771
- Improving the programming quality of ChatGPT (e.g., runtime and memory usage). https://www.sciencedirect.com/science/article/pii/S2666827024000021

- AI-Driven Financial Analysis: Exploring ChatGPT's Capabilities and Challenges. https://www.researchgate.net/publication/380862587_AI-Driven_Financial_Analysis_Exploring_ChatGPT's_Capabilities_and_Challenges
- Using knowledge graph (extra knowledge) to reduce the hallucination in Large Language Models. https://arxiv.org/pdf/2307.07697