CS5481: Data Engineering - Assignment2

Instructions

1. Due on Tuesday, Oct. 29, 2024, 6:00 PM.
2. You can submit your answers by **a single PDF with the code and the output files** or **a jupyter notebook with output files** containing both the answers and the code.
3. For the coding questions, besides the code, you are encouraged to additionally give some descriptions of your code design and its workflow. Detailed analysis of the experimental results is also preferred.
4. Total marks are 100.
5. If you have any questions, please post your questions on the Canvas-Discussion forum, or contact Mr. Guanzhi DENG (email: guanzdeng2-c@my.cityu.edu.hk) or Mr. Mingyang LIU (email: mingyaliu8-c@my.cityu.edu.hk).

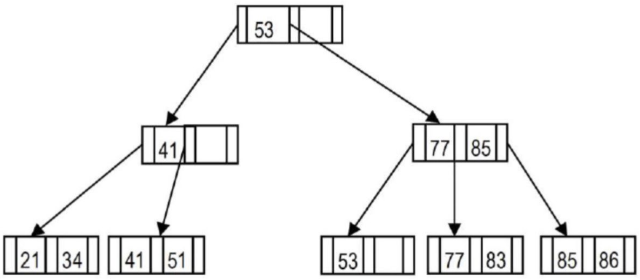
Question 1 - LLM for data engineering

(**20 marks**) LLMs' fast, articulate answers to expert questions can help data engineers discover datasets, write and debug code, document procedures, and learn new techniques as they build data pipelines. In this question, you are required to write suitable prompts for ChatGPT to achieve the following targets.

1. (5 marks) Assume you need datasets to train a recommender system that predicts user preferences for products. Use ChatGPT (GPT-4) to find relevant datasets. List the prompts you used and the corresponding outputs from ChatGPT.
2. (5 marks) Use ChatGPT to preprocess a sample from the **Movielens-1M** dataset for training a recommender system. List the prompts, inputs, and outputs from ChatGPT.
3. (10 marks) Generate a preprocessing script using ChatGPT to format the dataset for a Collaborative Filtering model using Surprise or TensorFlow. Correct any errors in the generated code, use the revised version to preprocess the dataset, and display the first 5 processed entries.

Question 2 - Data Indexing

(**25 marks**) Given the following B$^+$-tree, please answer following questions.



1. (5 marks) What is the value of p for this B$^+$-tree? (Note that p is the order of a B$^+$-tree)
2. (6 marks) Can you re-build a taller B$^+$-tree with the same value of p using the same set of search-key values in the leaf nodes of the given tree? If yes, show the steps by drawing a new diagram whenever the height of the tree increases.
3. (6 marks) Insert the search-key values 84, 19 and 32 in sequence to the given B$^+$-tree, and draw a new diagram for each insertion.
4. (8 marks) Suggest a sequence of search-key values to be deleted from the resultant B$^+$-tree in Q4.2 to shrink the tree to 2 levels with the **least** number of deletions. Show the steps by drawing a new diagram whenever a node is deleted.

Question 3 - Data Querying

(**25 marks**) The university held a coding contest where hackers submit solutions to various tasks. Each task has a bonus for the top 3 performers. You are given the following SQL tables:

- Hackers (hacker_id: INT, name: VARCHAR, bank_account: INT)
- Tasks (task_id: INT, description: VARCHAR, bonus: INT)
- Submissions (submission_id: INT, hacker_id: INT, task_id: INT, score: INT, submission_date: DATE)

Assume:

- Each task has a bonus for the top 3 submissions with the highest scores.
- If there are multiple submissions with the same score, the earliest submission (lower submission_id) is preferred.
- Hackers can submit multiple times, but only their best submission (highest score) counts for each task.

1. (**5 marks**) Write a query to print the hacker_id, name, and the number of distinct tasks each hacker participated in. Sort the result by the number of tasks in descending order, and then by hacker_id in ascending order if there’s a tie.
2. (**5 marks**) Write a query to find the task_id, description, and the total bonus awarded for each task. Sort the result by task_id in ascending order.
3. (**5 marks**) Write a query to list the submission_id, hacker_id, name, and score of the highest-scoring submission for each task submitted on 2023-01-01. If multiple submissions have the highest score for the same task, return the submission with the smallest submission_id. Sort the result by task_id in ascending order.
4. (**5 marks**) Write a query to print the hacker_id, name, and the total score each hacker achieved across all tasks. For each task, only the hacker's best score counts. Sort the result by total_score in descending order and by hacker_id in ascending order if there's a tie.
5. (**5 marks**) Write a query to find the hacker_id, name, and bank_account of hackers who did not participate in any tasks.

Question 4 - Recommender System

(**30 marks**)

1. **(8 marks)** Please write two basic approaches for recommender system and briefly explain them.
2. **(8 marks)** One common challenge in recommender systems is the filter bubble problem. Explain what the filter bubble problem is and how it affects user experience. Suggest at least two strategies to mitigate the filter bubble issue.
3. **(14 marks)** Rating prediction is an important task for a recommender system. Try to implement a recommendation model using the Goodbooks-10k dataset to predict user ratings. You have two options based on your available compute power:
   (a) Option 1 (Full Dataset): For students with sufficient compute power, use the entire dataset (6 million ratings for 10,000 books by 53,000 users).
   (b) Option 2 (Limited Dataset): For students with limited compute power, use a subset of the data. Limit the dataset to the top 5,000 users and top 3,000 books based on the number of interactions (ratings). In your submission, clearly state that you are using the subset due to computational constraints.
   (c) You can implement the model using user-based collaborative filtering, item-based collaborative filtering, or matrix factorization.
   (d) Split your chosen dataset (full or limited) into a training set and a test set.
   (e) Display the Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) on the test set.