Mini Project 1:

Automation for Query Development and Execution

Elena Gensch, Conrad Halle, Smilla Fox

Reflection Task 1: Query Development

How much experience in query development did the team have?
- Different levels of experience:
    some only used simple SQL queries in the past to, some have more experience

Which steps did the query development involve in each subtask?

- Subtask 1: create subqueries, nest them, check output, adjust code until final result is correct
- Subtask 2: create prompt with all necessary information (task and schema), ask ChatGPT to generate query, try output und check result, identify errors and ask ChatGPT to correct them

How time-consuming and difficult was the development?

- Understanding the problem, especially joins and combining subqueries very difficult
- 2 hours (subtask1) vs 5 minutes (subtask2)

How was the quality of the resulting implementation?

- LLM query better structured (subproblems split into functions)
- Performance of LLM query better

Reflection Task 1: Query Development

How helpful was the LLM in general and the additional explanations it provided (if any)?

- Extensive explanation not needed because we already understood the task well
- Explanations when fixing error were helpful
- For SQL beginners it was helpful to see how code can be structured better and which concepts could be used

How often did you have to prompt the LLM and were there any misunderstandings (on your side or the LLM's)?

- Asked for correction once
- It was easy to find the mistake but if we had not worked on the task before this would have been more difficult

What degree of automation did the tools you used for query development achieve on their own and in combination?

- Without LLM: Non-automatic - SQL expertise needed to translate problems and debug errors
- LLM-Assisted: Semi-automatic - LLM automates SQL generation from natural language, but requires human-guided prompt refinement and validation

Reflection Task 2: Query Extension

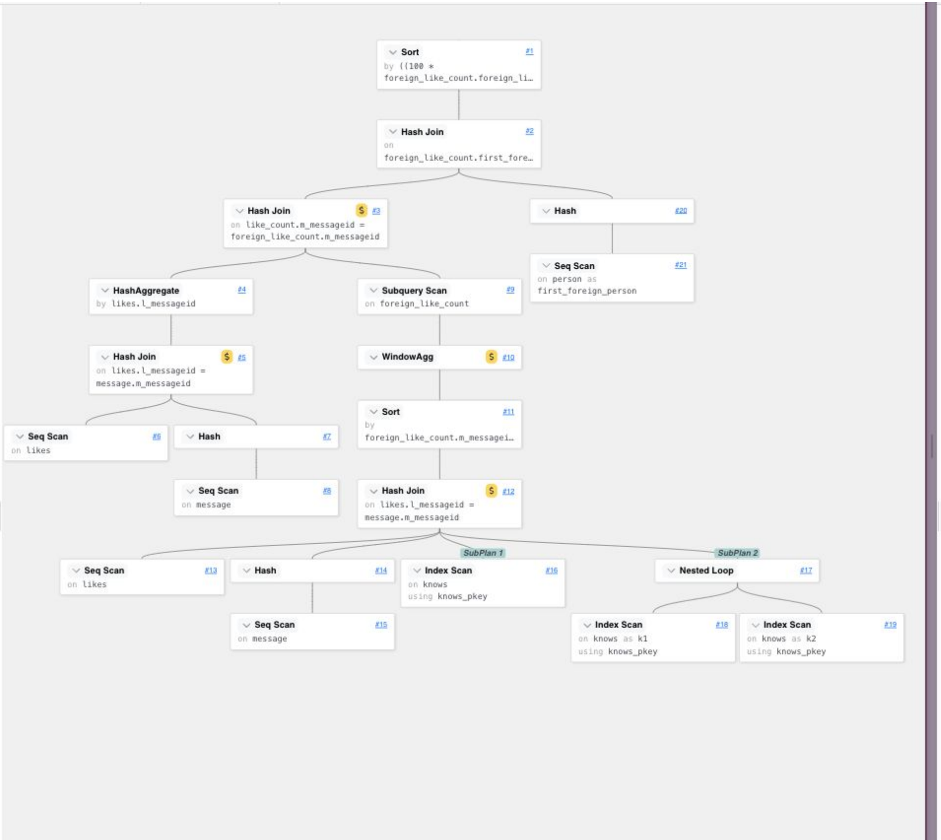
How well did the extending the query work?

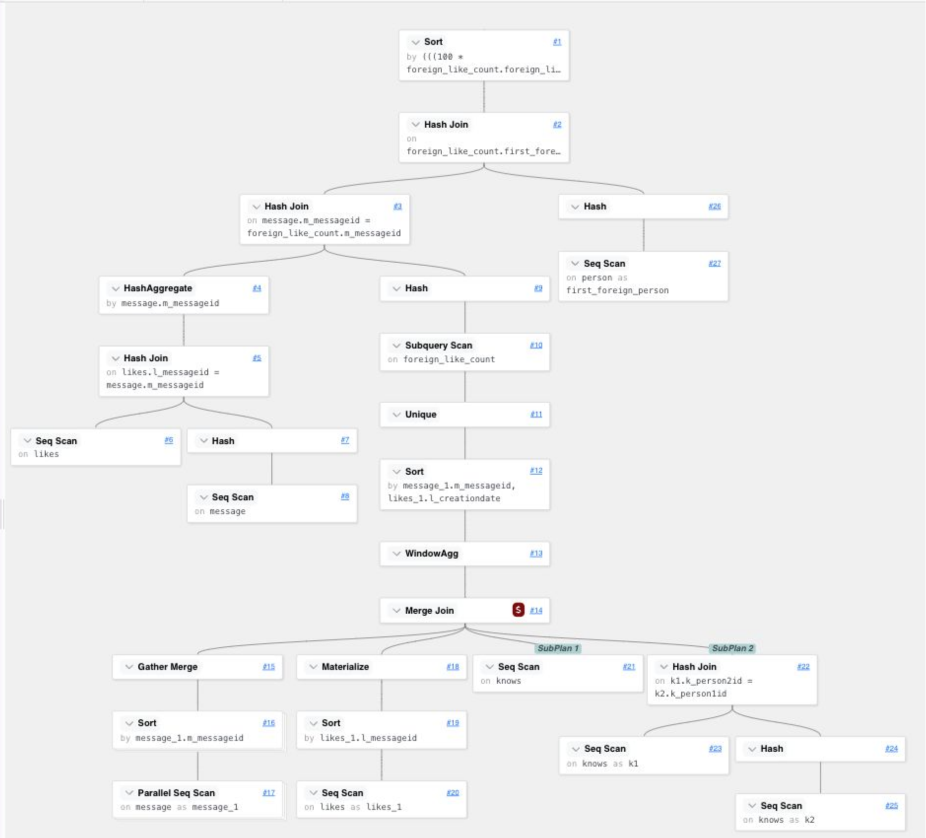
- With LLM:
    - correct when updating its own code
    - incorrect when updating our query
- Manually
    - depends on SQL knowledge
        - for SQL beginner: easier with more structured ChatGPT query
        - otherwise: easier to extend own query because it takes time to understand ChatGPT generated query

How could automation as provided by the LLM be integrated holistically into an iterative query development process?

- Integrate LLM into IDE
- Give the schema as context and previous query
- LLM could fix errors by itself

Task 3: Query Execution





Reflection Task 3: Query Execution

- The LLM-generated plan is somewhat plausible, but a lot of it is halluzination
- Further explanations were general and not really helpful
- Query Execution is fully automated in SQL

How well can the tasks be automated?

- Tasks 1 and 2:
    - No knowledge about the database contents except schema necessary
    - Many examples of SQL code online, therefore included in LLMs training data
    - Cannot be fully automated because output can always have errors
- Task 3:
    - Query plan can be computed correctly based on clear rules -> fully automatable