
Designing a database for Pokemon!

Aashna DOGRA

May 9, 2024

1 Introduction

GenAI tools like ChatGPT and Gemini are increasingly popular for handling daily tasks. But are they good with data and database structure knowledge. In this report, I work with ChatGPT to understand data and work on SQL and NoSQL databases to see how efficiently it works. The data revolves around pokemon mechanics where various pokemons have some strengths and moves. We work with the pokemon type, move, power of move, strength and weakness and see how well can GenAI interpret this data.

2 Problem Statement

The primary study done in this report is that how GenAI works with databases. The problem statement

- Pokemon can have one or two ‘types,’ which decide whether they’re more effective or less effective against other Pokemon types.
- Every pokemon has a primary type; some also have a secondary type.
- The game involves using moves to attack other Pokemon, and each move has a certain power and type.
- Every move has a set of Pokemon who are capable of learning it; and every Pokemon has a set of moves it can learn.
- However, ‘Pokemon’ and ‘Move’ have a classic many-to-many relationship.

3 Understanding Data with GenAI

I started with explaining the problem statement to ChatGPT and shared the various pokemons and what type they are. I also gave it information what move can they use with what power.

Prompt: I started with explaining it what we are going to do, ”We’re going to simplify Pokemon to just a couple of mechanics... However, ‘Pokemon’ and ‘Move’ have a classic many-to-many relationship. How will you deal with this?”

Response: ChatGPT understood and suggested to have a junction table.

Prompt: I then asked ChatGPT to show me what tables it could make for this and how would it relate them.

Response: ChatGPT gave me pokemon, move, type tables but forgot to add the strong and weak against sections. After clearing up that error it gave me a type table which included typeID, typeName, strong and weak against.

ChatGPT understood the data pretty well and formed correct tables with proper entries and types defined.

4 GenAI working on SQL

MySQL is a relational SQL database. It creates tables and allows us to join various tables and find relations in the data.

Prompt: Then I asked ChatGPT to formulate these tables in MySQL.

Response: It responded with clean code but the tuple table with multiple entries in one row was not workable.

Prompt: So I asked it for a solution.

Response: and it suggested to make a new table typeRelationships which displayed the move and strong and weak against section.

Prompt: After the tables worked fine I asked it to "Write a query that returns all the pokemon who can learn 'Return'".

Response: It gave me a query which used JOIN on pokemon and Move table and returned the desired answer.

Prompt: Then I asked it to "Write a query that returns all the moves in the game that are powerful against Grass".

Response: ChatGPT did not perform very well here. It had to access the junction table it made but it forgot about it.

Prompt: So then I explained it the structure of all the tables again.

Response: It understood and gave me a query which worked on Move table and typeRelationships table but the query gave empty output. So I understood the error was in the way the query was retrieving the datatype of the move which was in int in typeRelationships but string in Move.

Prompt: I tried to explain the datatypes to ChatGPT so it could realize its mistake but it wasn't so I clearly mentioned what's wrong.

Response: It understood the clear instruction and generated a query which performed JOIN on Move to check the Move then Saved its TypeID from type table and then JOIN on typeRelationships to filter out the two moves 'ember' and 'wing attack' as the ones powerful against grass.

5 GenAI working on NoSQL

MongoDB is a NoSQL database. It creates collections instead of tables and does not form clear tables that can have relations. I started by sharing the problem statement with ChatGPT and asked it to code it in MongoDB but it did not perform very well so I changed my approach. **Prompt:** After generating the SQL code I asked it to convert it to NoSQL.

Response: It had some doubts in what exactly I meant with convert and clarified that

Prompt: So I asked it to write MongoDB commands to make these tables.

Response: It gave me the whole code on creating collections and documents.

Prompt: After that I asked it to write the queries.

Response: To display pokemons that can learn "Return", it gave me 2 queries to store moveType and pokemonID from collections Move and Pokemon in var

datatype and then finally use find to find them in the database. For the second query it got confused.

Prompt: It had stored strong and weak against in an array type form in the Type collection but was not able to access it. So I tried to explain it the structure again.

Response: But ChatGPT was not able to understand and generate the correct query to access all details.

6 Analysis

Analysis after the conversation:

- GenAI understands data perfectly but sometimes tends to ignore certain parts of data with huge influx at the same time
- For MySQL it created a new junction table to address the many-to-many relationships.
- For NoSQL it created an array like structure to address the many-to-many relationships.
- I noted it faces difficulty in relating datatypes while writing queries.
- It works with basic structure and does not give an optimized structure in the first go, it does so only after prompting several times.
- It used JOIN in MySQL database to find relations but wasn't able to perform very well on MongoDB NoSQL database.
- While writing structure code it performed similar in case of both SQL and NoSQL.
- But while writing the queries it worked better with relational database.

7 Conclusion

GenAI has lot of potential to work on data but does get confused with details. With this study I was able to notice how GenAI works with data and various types of databases. It follows the simple approach and does not address many-to-many relationships efficiently. GenAI works well on newly presented data and is able to interpret relationships but sometimes it does face challenges while working on those relationships with databases.

8 Appendices

ChatGPT transcript/chat: <https://chat.openai.com/share/a0a99c1b-24e3-415a-a8b3-652ffbb1faa>