Prototyping

By: Andrew Snyder, Alex Polston, Alek Norris,
Eamon Collins, James Byrd, Svyatoslav Varnitskyy

sdmay25-20

HADES: Holistic Astronomical
Database Exploration System

Project Overview

- **Objective:** Develop a system to manage and analyze simulated binary star data
- **Key Features:**
  - Import multivariate time-series simulation data into relational database
  - Provide sample SQL queries
  - Enable custom queries through natural language processing
- **Deliverables:**
  - Relational database
  - User Interface for writing and viewing SQL queries
  - Sample SQL queries

UI Prototyping

UI Reflection

Why We Need This:

- Users will utilize this page as a home/query page

Goals in Creating a Prototype:

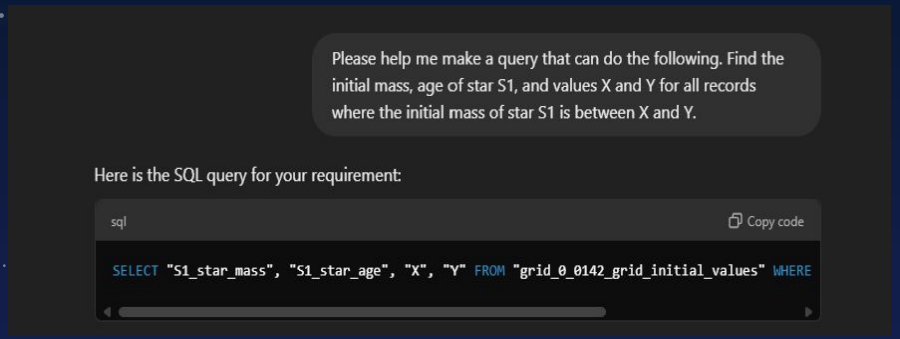
- Demonstrate overall styling and design pattern
- Determine if UI is intuitive

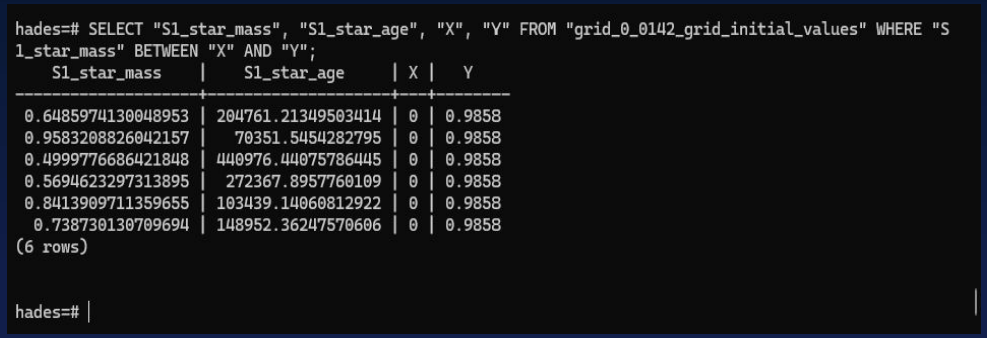
Outcomes:

- Styling looks professional and intuitive
- We need to expand on the navigation bar to include more tools
- Overall experience is positive

NLP SQL ProtoTyping

Still awaiting funding from Iowa State To use API so “free” version of GPT with prompt training was used to demonstrate functionality.





Prompt: Please help me make a query that can do the following. Find the initial mass, age of star S1, and values X and Y for all records where the initial mass of star S1 is between X and Y.

Generated query correctly and returns relevant data.

Query: SELECT *"S1_star_mass", "S1_star_age", "X", "Y"*
FROM *"grid_0_0142_grid_initial_values"*
WHERE *"S1_star_mass" BETWEEN "X" AND "Y";*

NLP SQL ProtoTyping

Prompting

Im wanting to query "public.grid_0_0142_grid_run0_final_profile1" to look at the energy, where the mass is greater than 5.

To query the data for research purposes based on your requirement, you can use the following SQL command:

1

```sql
SELECT energy
FROM "grid_0_0142_grid_run0_final_profile1"
WHERE mass > 5;
```

nothing showed up, could you show me one, including "logRho" where mass is greater than 1.5?

You can use the following SQL query to look at `logRho`, `mass`, and `energy` values where `mass` is greater than 1.5 from the `"public.grid_0_0142_grid_run0_final_profile1"` table:

2

```sql
SELECT mass, logRho, energy
FROM "grid_0_0142_grid_run0_final_profile1"
WHERE mass > 1.5;
```

Queries and Results

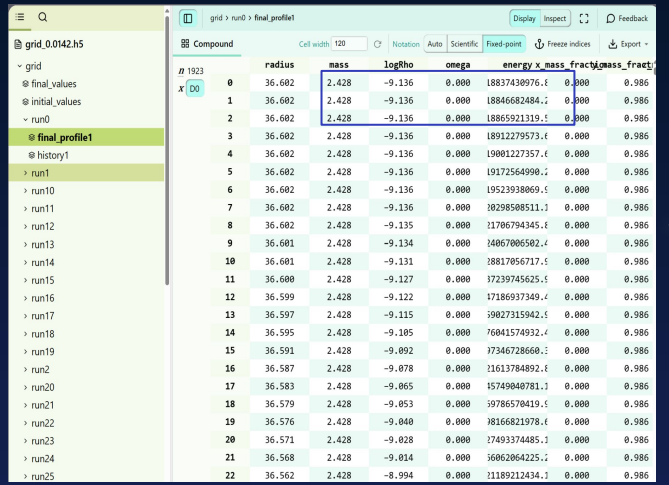
1

```
hades=# SELECT mass, energy FROM "grid_0_0142_grid_run0_final_profil
e1" where mass > 5;
 mass | energy
------+--------
(0 rows)
```

2

```
hades=# SELECT mass, "logRho", energy FROM "grid_0_0142_grid_run0_fi
nal_profile1" where mass > 1.5;
        mass        |        logRho         |        energy
--------------------+-----------------------+----------------------
 2.4282737796159032 |    -9.136390283830577 |  1918837430976.8416
 2.4282737961569523 |    -9.136385738294923 |  1918846682484.2263
 2.4282737961536687 |    -9.136376285548819 |   1918865921319.566
  2.428273796145799 |    -9.136353507185277 |  1918912279573.6323
  2.428273796126791 |    -9.136309798045636 |   1919001227357.688
  2.428273796094218 |    -9.136225588910927 |  1919172564990.2378
 2.4282737960273955 |    -9.13605283777916  |  1919523938069.9016
 2.4282737958902247 |    -9.135671753942669 |  1920298508511.1755
  2.428273795576972 |    -9.134977642395501 |  1921706794345.8872
 2.4282737950683777 |    -9.133810604257508 |   1924067006502.471
 2.4282737941908974 |    -9.131446150132005 |  1928817056717.969
  2.428273792243294 |     -9.1271758655197  |  1937239745625.9265
  2.428273789032749 |    -9.121944620825284 |  1947186937349.4182
  2.428273785828184 |    -9.115405319722937 |  1959027315942.9255
 2.4282737808556893 |    -9.105310190363538 |   1976041574932.489
  2.428273772867065 |    -9.092210443503726 |  1997346728660.3992
 2.4282737634462848 |    -9.078021173265473 |  2021613784892.8313
  2.428273753204602 |    -9.065011749167072 |  2045749040781.1973
  2.42827374459265  |    -9.053217905603713 |  2069786570419.9033
 2.4282737335339396 |    -9.040464185947844 |  2098166821978.6426
```

Native Data



1. Initial generated Query that returned no matching data
2. Updated Query and results

NLP SQL Reflection

Why We Need This:

- Not all users know how to write proper SQL queries

Goals in Creating a Prototype:

- To demonstrate how Machine Learning can be used to translate natural language to SQL queries

Outcomes:

- Natural language to SQL statement translation is possible
- Prompt training helped improve translation accuracy
- Further, stricter prompt training is required

Implications and Next Steps

- We have confidence that the NLP SQL generation is feasible and workable
- We have confidence that we can meld the AI SQL generation with the PostgreSQL database
- Expand on UI to include other pages and begin integration with Natural Language Processing
- Begin incorporating Database connection into UI
- There is still further work to do on the prompt training of the AI SQL generation module, however after tinkering expectations are high
- The integration of the Data into the database seems to be working, further work needs to be done to facilitate the normalization of the data.