Process Book

Jake Wagoner, Sudhanva Manjunath Athreya, Tin Vo

December 6, 2024

1 Feedback from TA Meeting (Proma):

1. Trends that are going to be plotted needs to have the intention
2. Create a geographic representation to show location of earthquakes

2 Project Discussions:

2.1 Initial Discussions - Regarding Datasets and Software Architecture

1. We considered using other data sets outside of earthquakes. Things such as the comparison of night and day temperatures in the state of Guanacaste in Costa Rica. Crocodiles have their gender decided when they hatch, and it is determined by the outside temperature, but with global warming, the night-time temperature is higher on average, so more female crocodiles are being born. Visualizing the difference can be done in many different ways, and I think an LLM could draw meaningful conclusions from it. In the end, we thought that earthquakes would make for more interesting visualizations.
2. For the prototype, we first decided to just put the .csv file in the code and pull it directly from there. We ended up deciding that using an API to pull data from the source between certain time periods would be interesting.
3. FastAPI was used as the back end, and data are dynamically retrieved based on user input.
4. The data was filtered and stored in JSON format, which then was returned to the user upon request.
5. We apply the filtered data based on date ranges into a shaded line chart with interactivity like hovering over data points will tell you specific information about that point, all of this is done using a VUE LineChart component with D3.

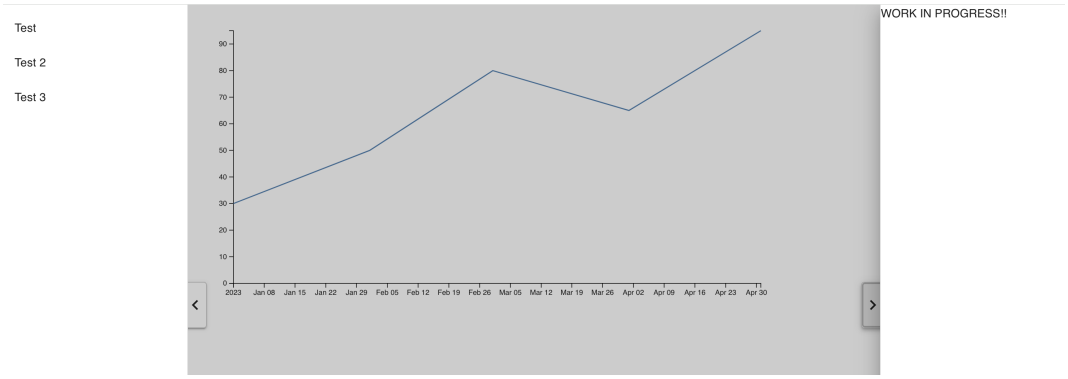

Figure 1: Line Chart Original Placements & Structure

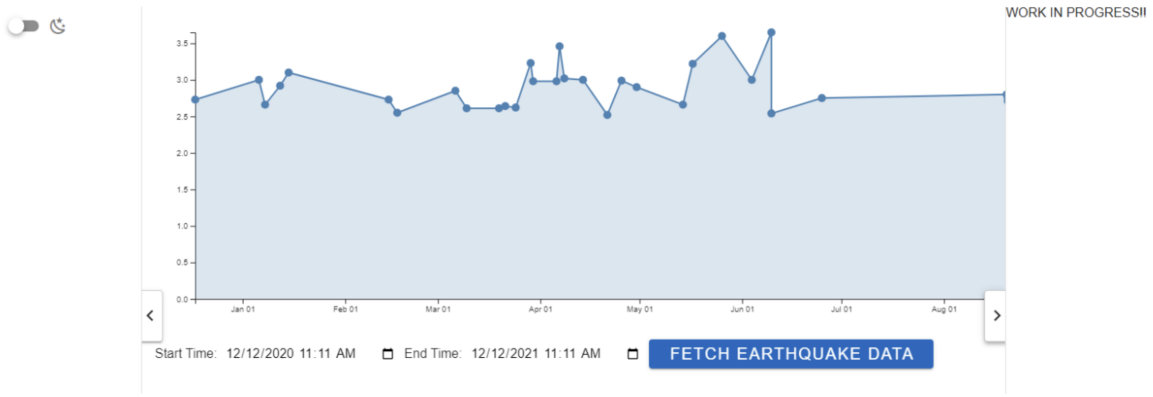

Figure 2: Line Chart after API integration

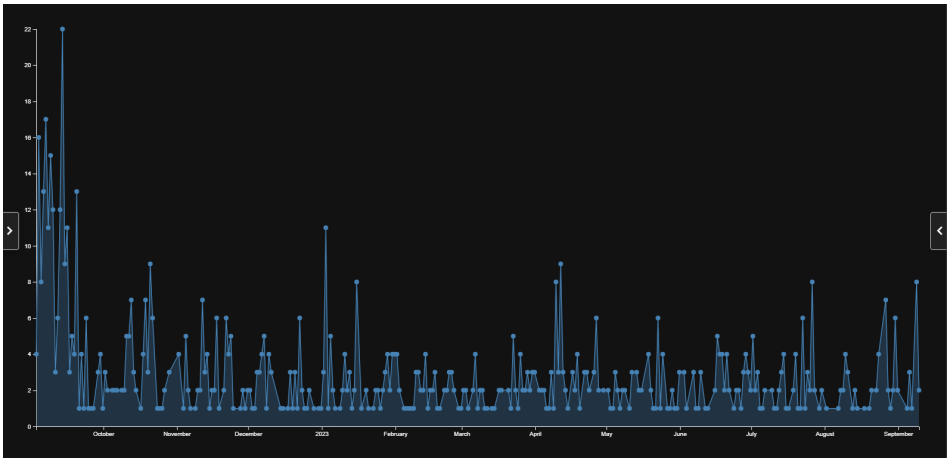

Figure 3: Dark Mode and Interactability with the charts.

2.2 Updates:

1. Added a night mode theme which shouldn't break too many rules since the color change is simple and still keeps the visualization clear, and it makes it easier on the eyes to look at.

2. Originally we wanted to do a time slider where we can pick the date ranges that way but for now, our temporary solution is to just have dropdowns to pick between two dates to visualize the data for.

3. The right tab that you can slide open is going to be used for the LLM function of our project but that is still a work in progress but the UI design was added to give a visual idea on what it would look like.

4. We tried to add another visualization type like histograms to have more options on how the data is shown but ran into a few problems because of the way the project is set up, so that is still a work in progress, similar to what other visualizations might plan to add.

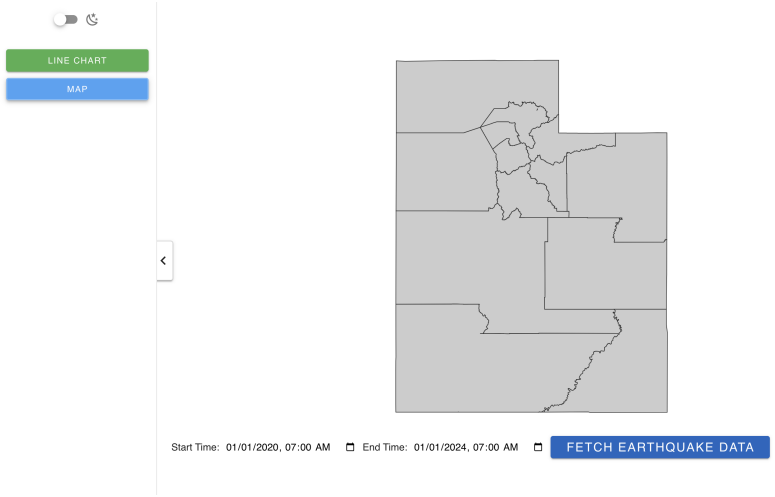

Figure 4: Map Visualization

3 Updates:

Map Visualization

- **Added map visualization**: Implemented a secondary visualization, a geographic map. This is a map of Utah with datapoints plotted in their geographic location. This allows users to view the data from a second perspective, highlighting regions and front lines rather than clusters in time.

Visualization Enhancements

Commit `e68aa23` introduced several critical changes:

- **Visualization Enhancements**: Commit `e68aa23` introduced tooltip and legend improvements, making data interpretation more intuitive. To accommodate the dark mode feature, the tooltip background opacity and text dynamically change color to maintain high contrast with the plot and background.
- **Time Slider**: A dynamic time slider was added to LineChart, allowing users to explore data across time ranges. This shows the line chart over the entire fetched data range. The user can select a subsection of this line chart to select a date range within the global dataset. Initially, no dates were added to the time slider, but this lead to ambiguity when selecting a date range, as the user would need to reference the above line chart to see what date ranges were selected. Then, months were added to the x-axis, but for datasets that span a large range of time, the axis quickly became cluttered. It was decided that having year markers would be the best mix of information density and readability/usability.
- **Categorical Encoding**: To separate the types of data entries the two major types were encoded with shape and color. Mining explosions were denoted with a green triangle, and natural tectonic activity is denoted with a blue circle. A legend was added to the top right of both visualizations.
- **Item Opacity**: The individual data point opacity was reduced, specifically for the map visualization. Before this reduction, a cluster of data points became ambiguous, as dots were covering eachother entirely. By reducing the opacity of the individual points, the density at a given overlap can be seen. This allows the user to interpret the results better.

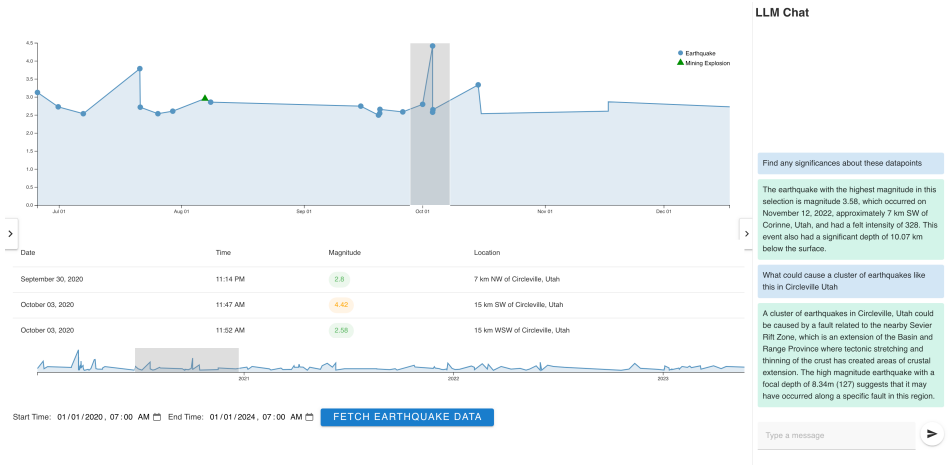

Figure 5: LLM Inference Sample Output

LLM enhancements

1. LLM initializer pipeline was built.
2. Different Quantization strategies were tried and tested.
3. 4-bit LLaMa:3b seems to work well, but is slow. Quantizing it to 2-bit results in loss of information.
4. User selections were passed through to the LLM. This allows the LLM to use contextual information and provide better details about user interactions.
5. **System prompts**: By adding system prompts (pre-prompting), the LLM was directed to answer in more direct, concise, and informative responses.

State Management

A new state management system was introduced to keep track of user selections and interactions, providing seamless updates across components.

Real-Time Communication

WebSocket functionality was debugged and expanded in `5bd8c59`, ensuring robust communication between the client and server.

Addressing Required and Optional Features

Every required feature noted in the project proposal has been included in the final application. Additionally, many of the optional features have been included. Some of the features of the final application also represent additional features not noted in the original proposal.

Some optional features were deemed to be either impossible or unlikely to be accomplished due to time constraints. For example, using multiple datasets. While this is entirely possible, the overhead required to generalize the infrastructure, acquire the data, implement the backend API calls for fetching this new data, and the integration into the frontend visualizations simply would take too much time. This feature, and related optional features, simply exceed the scope of the project in relation to this class.

An optional feature which represents a reasonable future expansion of this project is the addition of an overlaid density plot on the line chart. This could be in the form of a KDE or strip plot. The design of such an addition would be inspired by Allen et al. Raincloud plots: a multi-platform tool for robust data visualization.

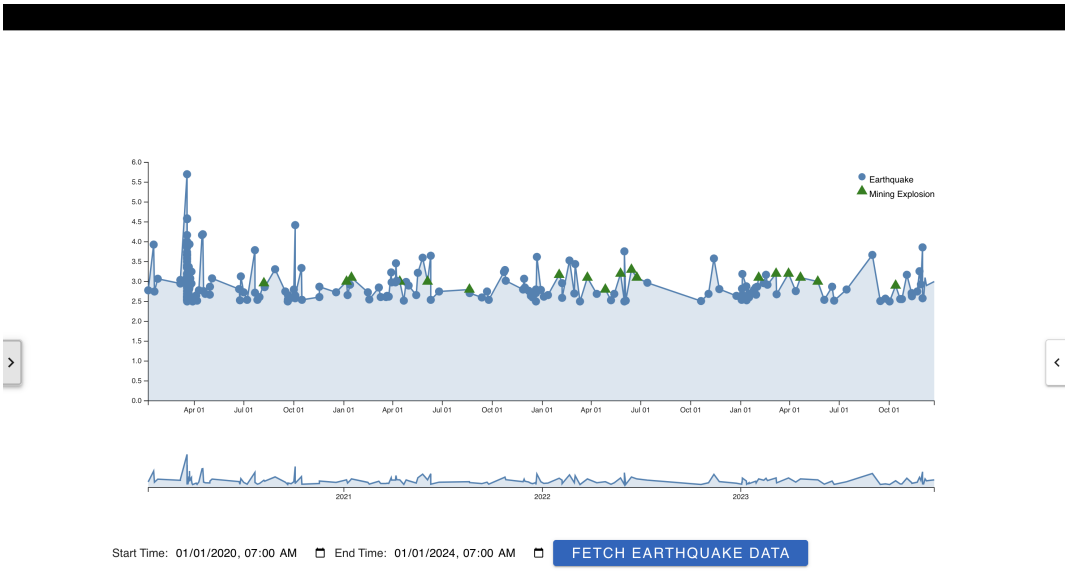

Figure 6: Final Line Chart View

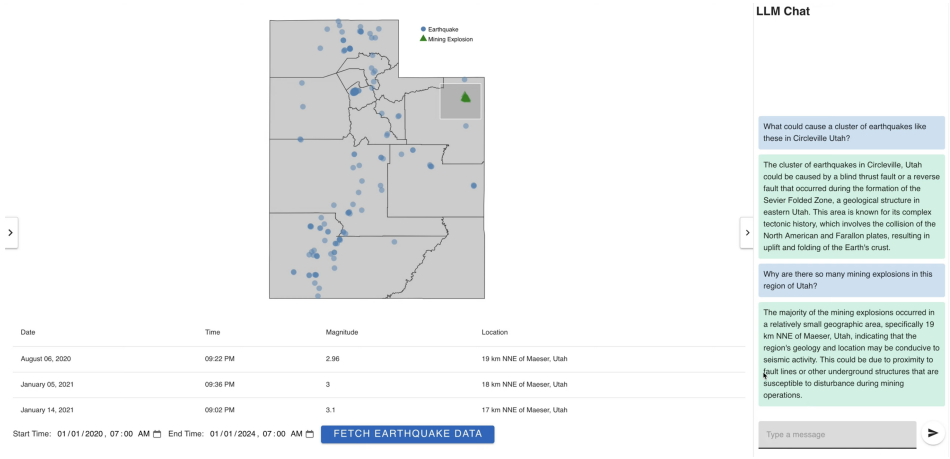

Figure 7: Final Map View with LLM chat integration