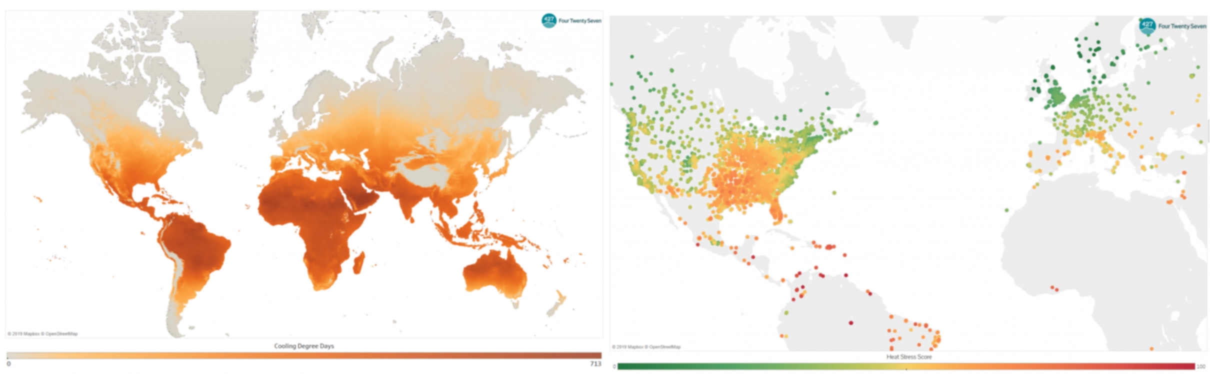
INFOSCI301
Prof. Luyao Zhang
Week 2 Assignments
Chenshuhao(Cody) Qin

Redesign in Global Data Visualization for Climate Risk, Human Trafficking Detection, and Seismic Activity

Exercise 1: Exploring Cutting-Edge Literature in Data Visualization and Aesthetics (Research)



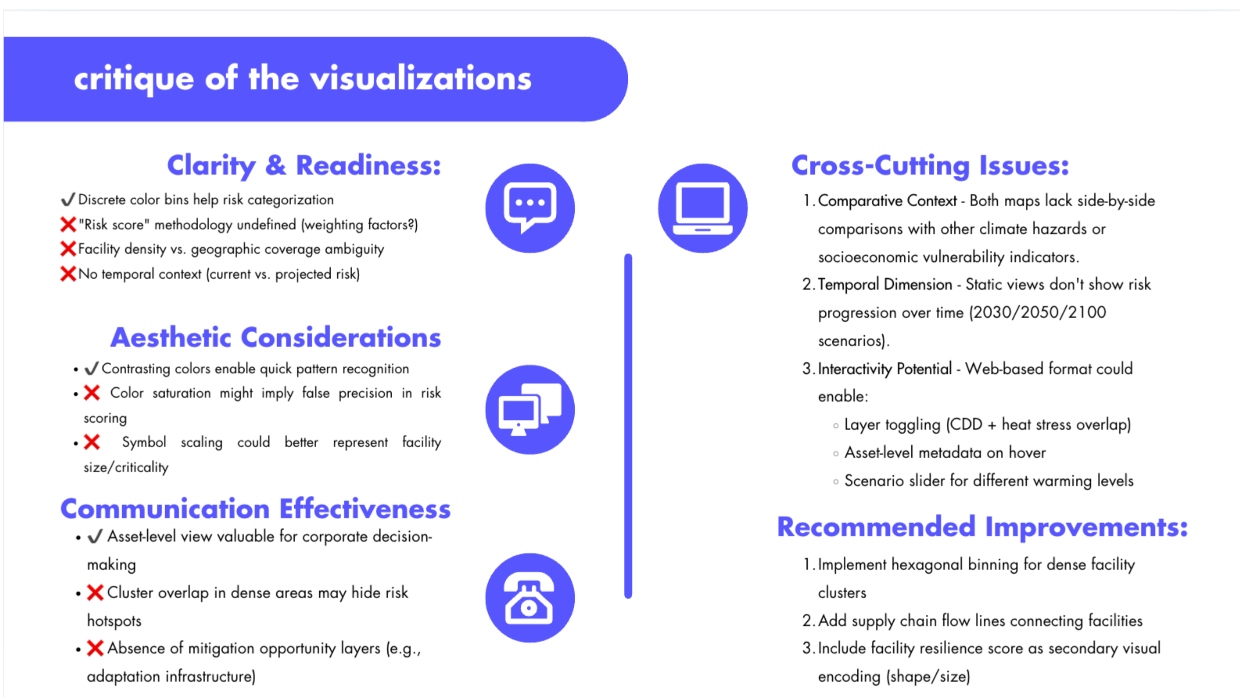


Figure 1. Flowchart of LLM's Critique (created by Canva)

The critique is provided by Doubao. It offers a detailed and structured assessment of the visualizations, covering clarity, communication effectiveness, and aesthetic considerations. It highlights key strengths, such as effective use of color gradients and risk categorization, while also pointing out areas for improvement, including the lack of geographic reference points, unclear methodologies, and missing temporal context. One key area for improvement is the accessibility of the visualizations. While the critique suggests better color schemes, it would also be helpful to emphasize interactive features that allow users to explore different scenarios and risk factors dynamically. Additionally, providing more context on the real-world implications of the data—such as the impact of cooling degree days on energy infrastructure—could make the visualizations more actionable for decision-makers.

The visualizations presented in "Leveraging the Cloud for Rapid Climate Risk Assessments" effectively convey climate-related risks through visually intuitive and data-rich formats. However, they have limitations. The Cooling Degree Days map uses a color gradient to show projected increases in cooling demand but lacks clear thresholds or labeled scales, making it difficult for users to interpret precise climate projections[1]. This absence of interactive elements or data labels restricts accessibility and reusability, as viewers cannot engage with or extract meaningful insights for secondary use.

The Heat Stress Score map provides a more detailed view of heat-related risks at the facility level, using a green-to-red gradient to denote severity levels. This enhances interoperability and risk differentiation but suffers from overlapping data points in densely populated areas, such as the eastern U.S. and parts of Europe, which impairs findability[2]. Both maps are visually appealing but would benefit from improved legends, scale annotations, and dynamic features like zooming or tooltips to enhance reusability and interoperability, aligning with the FAIR data principles[3].

---

[1] Wilkinson, Mark D., et al. *The FAIR Guiding Principles for Scientific Data Management and Stewardship*. Scientific Data, vol. 3, no. 1, 2016, doi:10.1038/sdata.2016.18.
[2] Wilkinson et al., "FAIR Guiding Principles."
[3] Wilkinson et al., "FAIR Guiding Principles."

Exercise 2: Reflection on Creating Dashboards from Zero to Hero: Experiment with Amazon QuickSight (Theory & Practice)

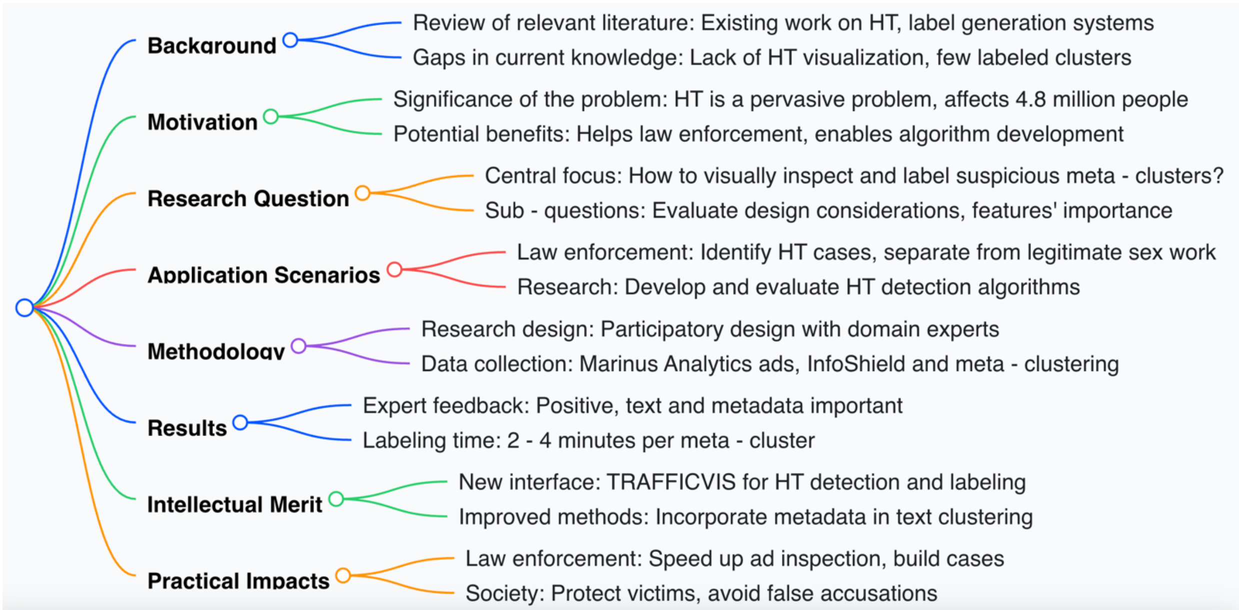


Figure 2. Flowchart of Research Process (Created by **Figma**)

Written Description

Human trafficking, especially for forced sexual exploitation, is a significant global problem affecting millions. Many victims are advertised on online escort websites, making it crucial to distinguish between legitimate sex workers and trafficking cases. Current methods face challenges in detecting human trafficking (HT) due to the complexity of the data and the presence of spam and scam ads. TRAFFICVIS is an innovative interactive application designed to address these issues by visually presenting suspicious meta-clusters (micro-clusters connected with metadata) for domain experts to label and analyze (Nair et al. 2021)[4].

4 Minji Lee et al., "INFOSHIELD: Generalizable Information-Theoretic Human-Trafficking Detection," Proceedings of the IEEE International Conference on Data Engineering (ICDE), (2021): 1116–27.

TRAFFICVIS builds on existing HT detection algorithms like InfoShield (Lee et al. 2021)[5]. InfoShield creates micro-clusters based on text similarity, using techniques such as term frequency-inverse document frequency (tf-idf) and the Minimum Description Language (MDL) principle. TRAFFICVIS then connects these micro-clusters into meta-clusters using metadata like phone numbers, emails, and social media accounts. This connection helps in identifying larger organized activities, which is a key aspect of detecting HT rings (Nair et al. 2021)[6].

The design of TRAFFICVIS was informed by months of participatory design with domain experts. The interface has several panels, each serving a specific purpose. The Micro-cluster panel shows the posting behavior of micro-clusters, allowing experts to drill down into specific ones. The Timeline panel displays metadata usage and the number of micro-clusters with posted ads over time, while the Map panel shows the geographic spread of the meta-cluster or selected micro-clusters. These panels help in visualizing complex spatio-temporal data. The Text panel is crucial for analyzing ad text, as it shows templates and individual ads, highlighting differences. The Labeling panel enables experts to quickly label meta-clusters for different modus operandi (M.O.), such as HT, spam, or scam (Nair et al. 2021)[7].

To evaluate TRAFFICVIS, expert feedback was solicited from four domain experts with varying levels of experience in HT. The experts used TRAFFICVIS to label 10 meta-clusters selected from a dataset of escort ads. The results were overwhelmingly positive. Experts found the distinction between meta-clusters and micro-clusters useful, as it allowed them to explore different levels of data. Text was the most important feature for identifying cluster behavior, with keywords like mentions of exotic ethnicities, scarcity of girls, and offering high-risk services being common indicators of trafficking. Metadata, such as phone numbers, geographic spread, and temporal distribution, also influenced their labeling decisions. TRAFFICVIS was considered accessible to various anti-HT stakeholders, easy to use, and a significant time-saver compared to

[5] Pranav Nair et al., "TRAFFICVIS: Visualizing Organized Activity and Spatio-Temporal Patterns for Detecting and Labeling Human Trafficking," (2021).
[6] Nair et al., "TRAFFICVIS."
[7] Nair et al., "TRAFFICVIS."

manual labeling. Experts could label meta-clusters in about 2–4 minutes using TRAFFICVIS, while they estimated it would take 20–30 minutes with other methods (Nair et al. 2021)[8].

However, TRAFFICVIS has limitations. There is a need to integrate more features, such as analyzing the spatial trajectories of meta-clusters over time and incorporating image data if available. The UI design could be improved based on expert suggestions, like larger font sizes and adding an additional label. Also, the labeling design makes post-processing labels for downstream tasks difficult due to overlapping labels. Future work includes soliciting more feedback from law enforcement, applying TRAFFICVIS to other domains like social media disinformation campaigns, and continuing to develop and evaluate M.O. detection algorithms using the labels generated by TRAFFICVIS (Nair et al. 2021)[9].

In conclusion, TRAFFICVIS is a high-impact system that enables label generation and saves time compared to manual labeling. It provides a useful tool for anti-HT stakeholders, especially law enforcement, to detect and analyze potential HT cases. The open-sourcing of the code and synthetic data also promotes reproducibility and further research in this area (Nair et al. 2021)[10].

Practice: Amazon QuickSight Workshop

[8] Nair et al., "TRAFFICVIS."
[9] Nair et al., "TRAFFICVIS."
[10] Nair et al., "TRAFFICVIS."

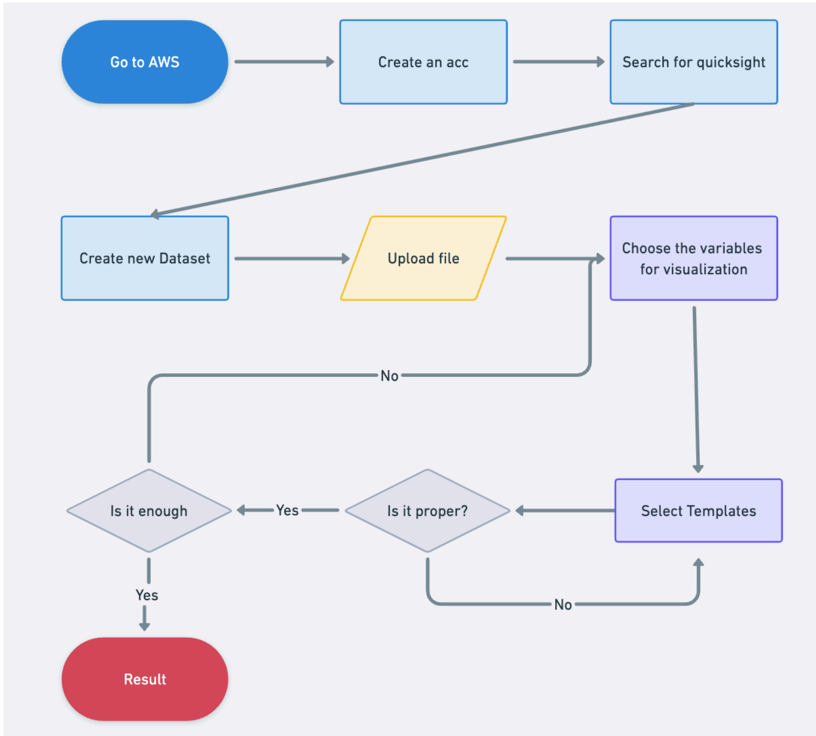


Figure 3. Flowchart for Dashboard (Created by **Whimsical**)

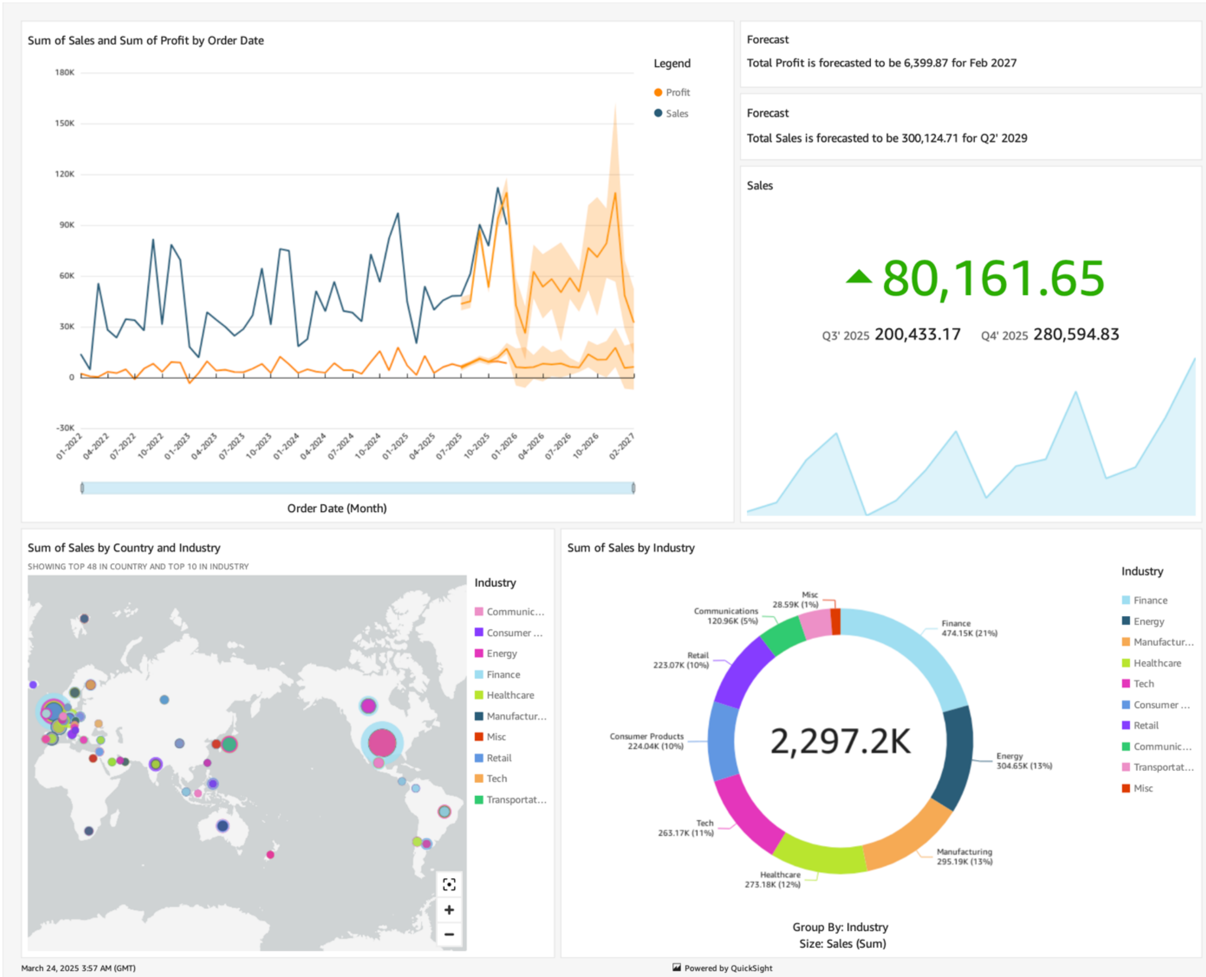


Figure 4. Dashboard (Created by **QuickSight**)

This dashboard is crafted using Amazon QuickSight[11], harnessing the SaaS - Sales dataset available on the platform. The dataset encompasses multiple variables, including order date, which allows for temporal analysis of sales and profit. Sales and profit themselves are the key numerical variables that drive the financial analysis. Additionally, variables such as industry and country are included, enabling segmentation of data by business sector and geographical region. Although the exact size of the dataset in terms of rows and columns is not specified, it is rich enough to support comprehensive visualizations.

The visualization reveals several notable trends and patterns. In the line chart titled "Sum of Sales and Sum of Profit by Order Date", both sales and profit exhibit fluctuating trends over time. There are distinct peaks and troughs, indicating periods of high and low performance. This

[11] Amazon Web Services, Amazon QuickSight User Guide (2025).

can help in identifying seasonal trends or the impact of specific events on sales and profit. The forecast sections on the right - hand side project future values for profit in February 2027 and sales in Q2 2029, providing insights into potential future performance.

The circular chart "Sum of Sales by Industry" shows the proportional contribution of different industries to the total sales, which amounts to 2,297.2K. This clearly highlights which industries are major drivers of sales in the SaaS domain. The world map visualization, "Sum of Sales by Country and Industry", uses colored circles to represent sales within various countries and industries, allowing for the identification of geographical hotspots for sales.

These visualizations are invaluable for decision - making. By understanding the trends in sales and profit over time, companies can plan their budgets, allocate resources, and time marketing campaigns more effectively. Identifying the leading industries and geographical regions in terms of sales can guide strategic expansion efforts, such as targeting new markets or investing more in high - performing industries. Overall, the dashboard transforms raw data into an intuitive format that simplifies the understanding of complex relationships between industry, region, sales, and profit, facilitating data - driven decision - making.

Amazon QuickSight provides an effective platform for creating visually appealing dashboards, as demonstrated by the report above. The tool's ability to integrate diverse visualization types, such as line charts, geographic maps, and pie charts, allows users to analyze data trends, regional sales distributions, and industry-specific performance in a single view. Its forecasting capabilities, powered by machine learning, add predictive insights to the dashboard, enabling users to make informed decisions based on projected sales and profits. Furthermore, QuickSight’s interactive features, such as drill-downs and filters, enhance the user experience by allowing deeper exploration of data.

However, despite these strengths, QuickSight has limitations that impact its usability. The interface can feel unintuitive for first-time users due to its complex navigation between datasets and visualizations. Additionally, while the visualizations are functional, they often lack advanced customization options for aesthetics, which can limit the ability to tailor dashboards to specific branding or design needs. For example, the pie chart and map visuals in the report could benefit from more refined formatting options to improve clarity and visual appeal.

To enhance the user experience, Amazon QuickSight could focus on improving interface intuitiveness by simplifying workflows and adding more customization options for charts and dashboards. Expanding its library of visualization types and refining its natural language processing (NLP) capabilities for querying data would also make it more competitive with other BI tools like Tableau or Power BI. Research in data visualization emphasizes that intuitive design and flexibility in customization are critical for effective communication of insights (Few 2006)[12].

Exercise 3: Infovis Redesign Pilot Experiment (Research & Innovation)

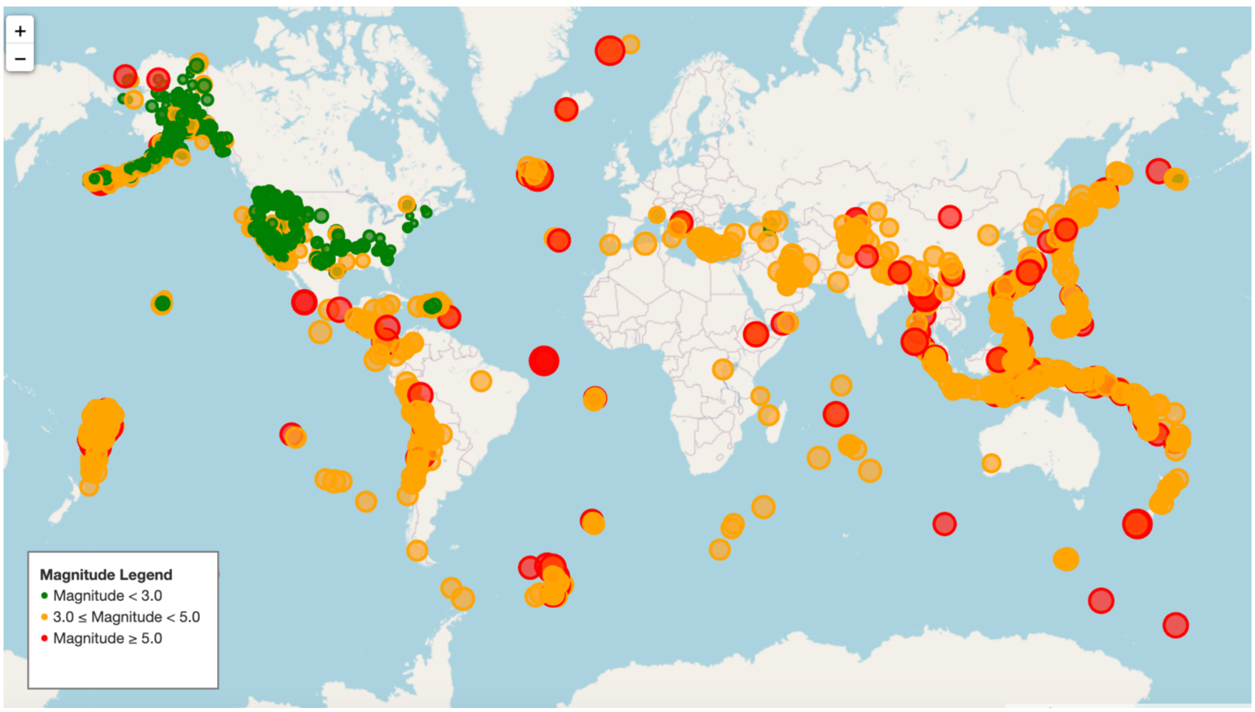


Figure 5. EarthQuake Visualization Redesign colored with Level Index Ranges

[12] Few, Stephen. *Information Dashboard Design: The Effective Visual Communication of Data*. O'Reilly Media, 2006.

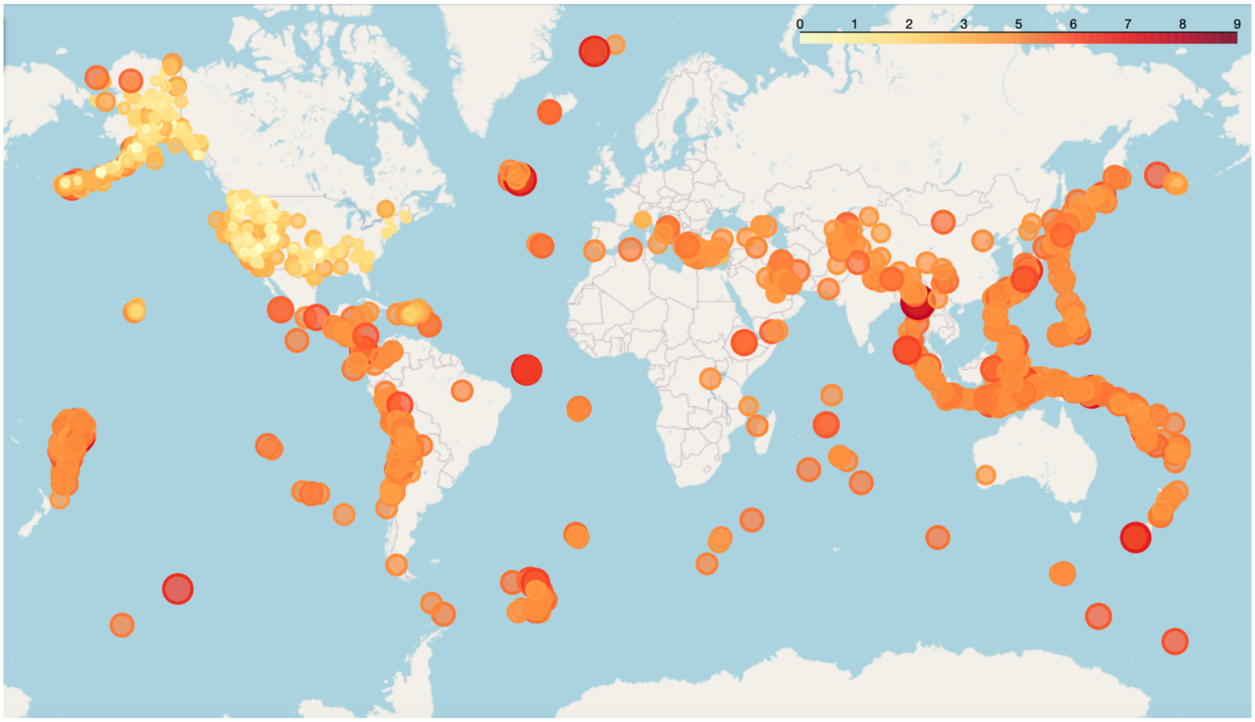


Figure 6. EarthQuake Visualization Redesign colored with Gradual Level Index

Figure 6. EarthQuake Visualization Details Example when touching the Signal

Introduction

This aims to visualize real-time earthquake data on an interactive world map to enhance public awareness and promote deeper insights into global seismic activity. Through the integration of open-source data and Python-based visualization tools, we demonstrate how geospatial information can be effectively communicated and explored.

Github: https://github.com/CodyQin/Visualization-Earthquack

Data Source

The data is sourced from the **United States Geological Survey (USGS)**[13], which provides open access to real-time earthquake information via its public GeoJSON API. The dataset used in this project includes all earthquakes that occurred globally within the past 30 days, offering rich metadata such as location, time, magnitude, and depth.

- Source: USGS Earthquake Feed – GeoJSON

Tools Used

- **Python** – Programming language for data collection, analysis, and visualization.
- **Requests** – For retrieving GeoJSON data from the USGS API.
- **Pandas** – For data processing and structuring.
- **Folium** – For creating interactive leaflet maps in Python.
- **Jupyter Notebook** – As the development environment for experimenting and prototyping.

Significance

This project demonstrates how **open geospatial data** and **Python-based visualization** can be combined to:

- Enhance accessibility and understanding of natural hazard data.
- Provide an interactive and visual method to analyze spatial patterns.
- Support decision-making for emergency response, education, and policy-making.

By displaying the **magnitude and location** of recent earthquakes on an interactive map, users can easily identify clusters, high-risk zones, and recent seismic trends.

[13] United States Geological Survey, "Real-time Earthquake Map," last modified April 2025, https://earthquake.usgs.gov/earthquakes/feed/v1.0/geojson.php.

Progress and Achievements

- Successfully connected to a live data source and retrieved structured earthquake data.
- Built an interactive world map with dynamic markers scaled by earthquake magnitude.
- Implemented a clean and accessible UI with popups that show real-time earthquake information.
- Generated a reusable tool (HTML map) that can be shared or embedded into other platforms.

Areas for Improvement and Future Work

While the current version achieves basic functionality, there are several directions for future development:

1. **Time-Based Animation**:
Add time-based layers or animation to visualize how earthquake events evolve over days or weeks.

2. **Severity Color Coding**:
Implement dynamic color scales to better represent different magnitude levels visually.

3. **Historical Comparison**:
Incorporate historical earthquake datasets to analyze trends and anomalies over years.

4. **Additional Layers**:
Integrate other geospatial datasets, such as population density or tectonic plates, for richer contextual understanding.

5. **Web App Integration**:
Deploy the map as part of a responsive web or mobile application to broaden its accessibility.

Codes Reference：

```
import requests
import pandas as pd
import folium

# Step 1: Fetch earthquake data from USGS GeoJSON API
url = 'https://earthquake.usgs.gov/earthquakes/feed/v1.0/summary/all_month.geojson'
response = requests.get(url)
data = response.json()

# Step 2: Extract relevant fields
earthquake_list = []
for feature in data['features']:
```

```
    properties = feature['properties']
    geometry = feature['geometry']

    if geometry['type'] == 'Point':
        coords = geometry['coordinates']
        earthquake_list.append({
            'Location': properties['place'],
            'Magnitude': properties['mag'],
            'Longitude': coords[0],
            'Latitude': coords[1],
            'Time': pd.to_datetime(properties['time'], unit='ms')
        })

# Step 3: Convert to a DataFrame
earthquakes = pd.DataFrame(earthquake_list)

# Step 4: Define a function to assign color based on magnitude
def get_color(mag):
    if mag < 3.0:
        return 'green'
    elif 3.0 <= mag < 5.0:
        return 'orange'
    else:
        return 'red'

# Step 5: Create base map
earthquake_map = folium.Map(location=[0, 0], zoom_start=2)#, tiles="CartoDB positron"

# Step 6: Add enhanced markers
for _, row in earthquakes.iterrows():
    mag = row['Magnitude']
    if pd.notnull(mag):
        color = get_color(mag)
        folium.CircleMarker(
            location=[row['Latitude'], row['Longitude']],
            radius=max(3, mag * 2),
            color=color,
            fill=True,
            fill_color=color,
            fill_opacity=0.7,
            popup=folium.Popup(
                f"<strong>Location:</strong> {row['Location']}<br>"
                f"<strong>Magnitude:</strong> {mag:.2f}<br>"
                f"<strong>Time:</strong> {row['Time']}<br>"
                f"<strong>Coordinates:</strong> ({row['Latitude']:.2f}, {row['Longitude']:.2f})",
                max_width=300
```

```
        )
    ).add_to(earthquake_map)

# Step 7: Add a legend (as a custom HTML element)
legend_html = '''
<div style="position: fixed;
    bottom: 30px; left: 30px; width: 180px; height: 130px;
    background-color: white; z-index:9999; font-size:14px;
    border:2px solid grey; padding: 10px;">
<b>Magnitude Legend</b><br>
<span style="color:green;">●</span> Magnitude < 3.0<br>
<span style="color:orange;">●</span> 3.0 ≤ Magnitude < 5.0<br>
<span style="color:red;">●</span> Magnitude ≥ 5.0
</div>
'''
earthquake_map.get_root().html.add_child(folium.Element(legend_html))

# Step 8: Save the map
earthquake_map.save("enhanced_earthquake_map.html")
print("Enhanced earthquake map saved as 'enhanced_earthquake_map.html'")
```

Bibliography

Amazon Web Services. 2025. *Amazon QuickSight User Guide*. Seattle: Amazon Web Services.

Few, Stephen. 2006. *Information Dashboard Design: The Effective Visual Communication of Data*. Sebastopol, CA: O'Reilly Media.

Lee, Minji, Cristian Vajiac, Aashish Kulshrestha, Steve Levy, Nathan Park, Cole Jones, Reihaneh Rabbany, and Christos Faloutsos. 2021. "INFOSHIELD: Generalizable Information-Theoretic Human-Trafficking Detection." In *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, 1116–1127. Chania, Greece: IEEE. https://doi.org/10.1109/ICDE51399.2021.00104.

Nair, Pranav, Cristian Vajiac, Duen Horng Chau, Min-Chieh Lee, Andreas Olligschlaeger, Ryan Mackenzie, Yuhan Li, Nathan Park, Reihaneh Rabbany, and Christos Faloutsos. 2021. "TRAFFICVIS: Visualizing Organized Activity and Spatio-Temporal Patterns for Detecting and Labeling Human Trafficking." *IEEE Transactions on Visualization and Computer Graphics* 28 (1): 586–596. https://doi.org/10.1109/TVCG.2021.3114831.

United States Geological Survey (USGS). 2025. "Real-time Earthquake Map." Last modified April 2025. https://earthquake.usgs.gov/earthquakes/feed/v1.0/geojson.php.

Wilkinson, Mark D., Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, et al. 2016. "The FAIR Guiding Principles for Scientific Data Management and Stewardship." *Scientific Data* 3 (1). https://doi.org/10.1038/sdata.2016.18.