Data Aggregation and Visualization Technique for Traffic Sensor Data

Anwesh Tuladhar, Sulav Malla, Ghulam Jilani Quadri, Paul Rosen
Department of Computer Science and Engineering
Email: {atuladhar, sulavmalla, ghulamjilani}@mail.usf.edu, prosen@usf.edu

University of South Florida

ABSTRACT

A wealth of information is captured by traffic sensors but extracting and representing the said information is a challenge. We developed a data processing tool in Apache Spark to aggregate the data points recorded by the sensors and enrich it with geographical information as well. We also developed a tool in Processing to aid the visual analysis of this data set. It plots the *paths* identified in the transformed data as a subway map, while still preserving the relative locations of each sensor. The transformed data is also suitable for further analysis using existing tools such as Tableau. We use all three of these tools in conjunction to solve the VAST challenge 2017 - mini challenge 1.

1 INTRODUCTION

The goal of mini challenge 1 is to find out any traffic patterns within the Lekagul Preserve that might have an impact in the decline of the Rose-Crested Pipit bird population. The data provided includes the map of the preserve, which also marks the locations of the various sensors within the park, and the data collected by each sensor. The sensors record the car-id, car type and time-stamp for each passing vehicle. The challenge is to provide a visual analytic tool that is capable of identifying daily patterns, patterns spanning multiple days and any unusual patterns to help in identifying the decline in population of the petit bird.

2 DATA AGGREGATION

2.1 Map

We developed a tool in Processing[1] to represent the given map as a weighted graph. First, we scan the map to find all the sensor locations, which represents the nodes of the graph. Then we perform a modified Depth First Search to find all the paths between all the nodes, which represents the edges. The edges are weighted by the distance between the two nodes. Using this graph, we can plot any path on the map. We also export data from this graph as a csv.

2.2 Sensor Data

We developed another tool in Spark[2] to aggregate the sensor data and combine it with the graph data from section 2.1. We group the sensor data by car-id to trace the path followed by each car in a day. Each such record now represents a trip for that car. We prevent dangling trips by considering edge cases, where a trip spans two days, by applying a heuristic that a trip can end either at a camping site, ranger-base or an entrance. We use the graph data to calculate the distance travelled, time taken and average speed for each trip. As a heuristic, we choose the path where the speed of travel is closest to the speed limit when multiple paths exist between sensors. We also maintain start gate, end gate and the day of week for each trip.

[1] `https://processing.org/`
[2] `https://spark.apache.org/`

We also noticed that many trips follow the same path in forward and reverse directions. So, we calculate the hash of the forward and reverse paths for easier grouping during visualization. This forms the data set for single day pattern analysis.

For the analysis of patterns spanning multiple days, we further enrich the data by combining the trips that have not yet exited the preserve. We supplement this with daily records information, end destination of each day spent in the preserve and total days spent in the preserve.

We export both the transformed data sets as JSON files.

3 VISUALIZATION

3.1 Using Existing Tools

Since the transformed data set is rich enough and available in simple JSON format, we do not need to develop custom tools to aid visual analysis. We can simply load the JSON file into a tried and trusted tool like Tableau[3] to create a wide range of visualizations quite easily.

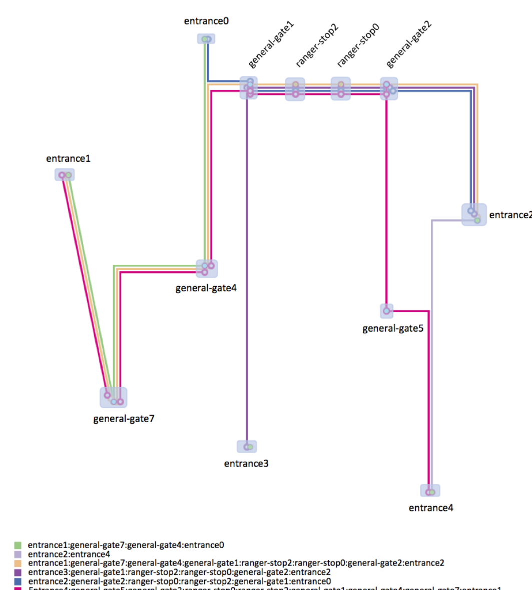

Figure 1: Top 6 Paths as a Subway Map

3.2 Subway Maps

Although the tool we developed in section2.1 is capable of plotting any path on a map, it was still not easy to visualize multiple paths together for comparison, especially when the different paths overlap. For this, we developed another tool in Processing to plot each path as a subway map. The subway map shows each path as a separate line, plotted with different colors. This makes it easy to distinguish each

[3] `https://www.tableau.com/`

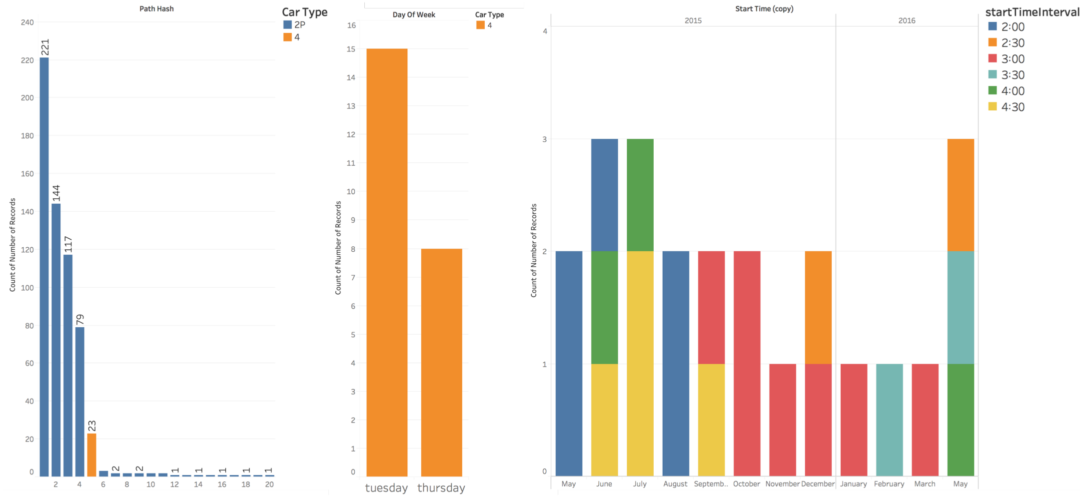

Figure 2: (a) All the paths which include a gate. Color indicates the car type. (b) The day of week when cars of type 4 travel this path. (c) The time of travel breakdown by months.

path even with overlaps. Our tool avoids line intersections between multiple paths when possible and allows for strategic placement of each node to minimize such intersections. In addition to this, our graph representation of the map also allows us to detect any non-existent paths.

4 Pattern Analysis

After the source data is pre-processed using our tools, visualization and analysis is very simple. We use Tableau to filter and plot visualizations as necessary. In figure 3, we simply plot the number of records for each path hash and color it based on car type and sort it based on the count. This visualization helps us to easily distinguish the top 10 paths from the rest. Figure 1 shows the top 6 paths together in a subway. For this, we simply feed the corresponding data to our subway map tool from section 3.2).

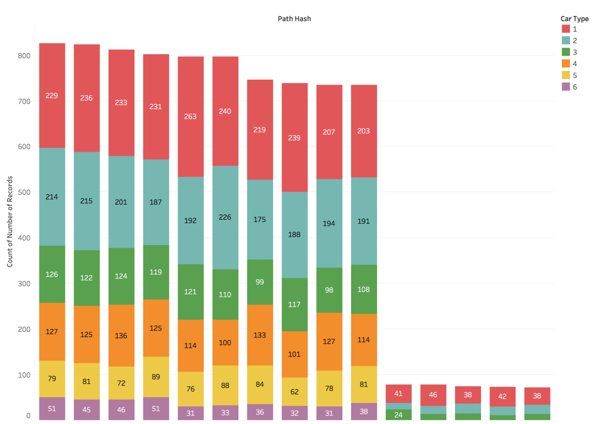

Figure 3: Top 15 Daily Patterns through the Preserve

Traffic flow is very likely tied with the time of day, day of week and/or the month. Our transformed data set already incorporates these information and hence, we can easily breakdown any pattern we find based on these as well. The preserve rules state that only ranger vehicles of type *2P* should pass certain *gates*. We plot all the paths passing such *gates* in figure 2a and find that 23 cars of type 4 also pass these gates. In figure 2b, we see that these cars take this same path always on Tuesdays and Thursdays. And from figure 2c, we see that these paths are travelled throughout the year and around the same time from 2 am to 4:30 am, suggesting that something fishy is happening in this route in those days.

5 Conclusion

In this way, we can use our tool chain to easily aggregate the sensor and geographical data into a single, rich data set and use it for visual analysis using Tableau and subway maps. We were able to answer all the questions posed in VAST challenge 2017's mini challenge 1. We refer the readers to our complete answer[4] and video[5] for further information.

[4]https://anwesht.github.io/vast2017/index.htm
[5]https://www.youtube.com/watch?v=c3DY_hK7NhQ&t=5s