Freeway Data Quality

Mingjue Wang, Ronnie Song, Frank Sun

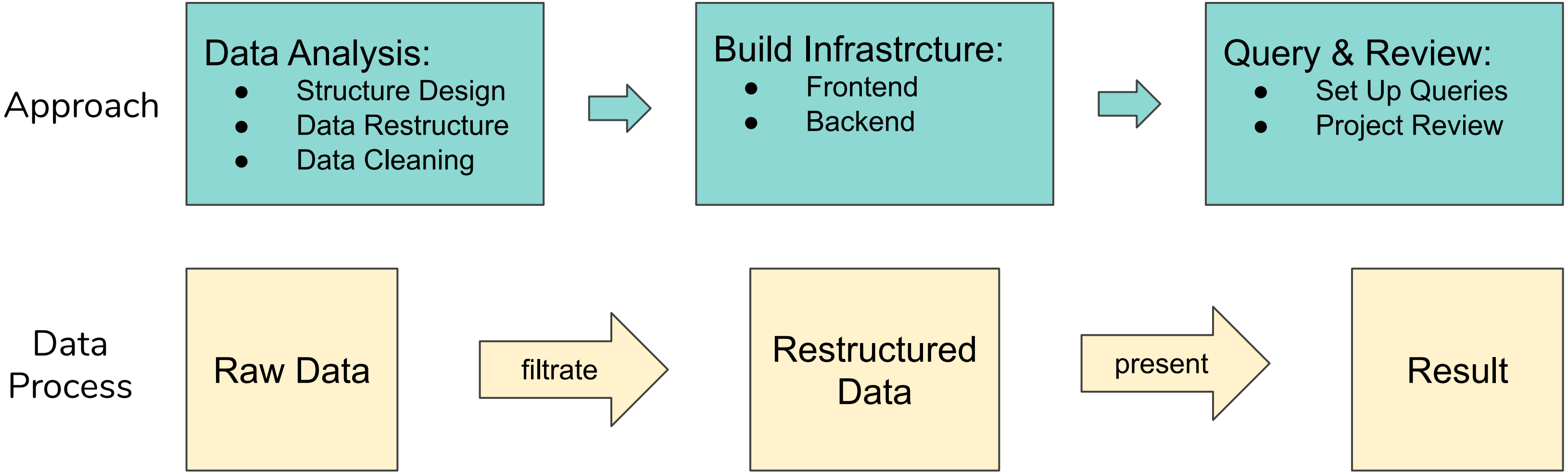
Project Goal

★ Determining the malfunctional detector, by analysis the data from highway dataset

★ Dashboard of speed detectors information: location, statistics, etc.

★ Language: JavaScript, Python

Project Development



Approach

Data Analysis:

- Structure Design
- Data Restructure
- Data Cleaning

Build Infrastrcture:

- Frontend
- Backend

Query & Review:

- Set Up Queries
- Project Review

Data Process

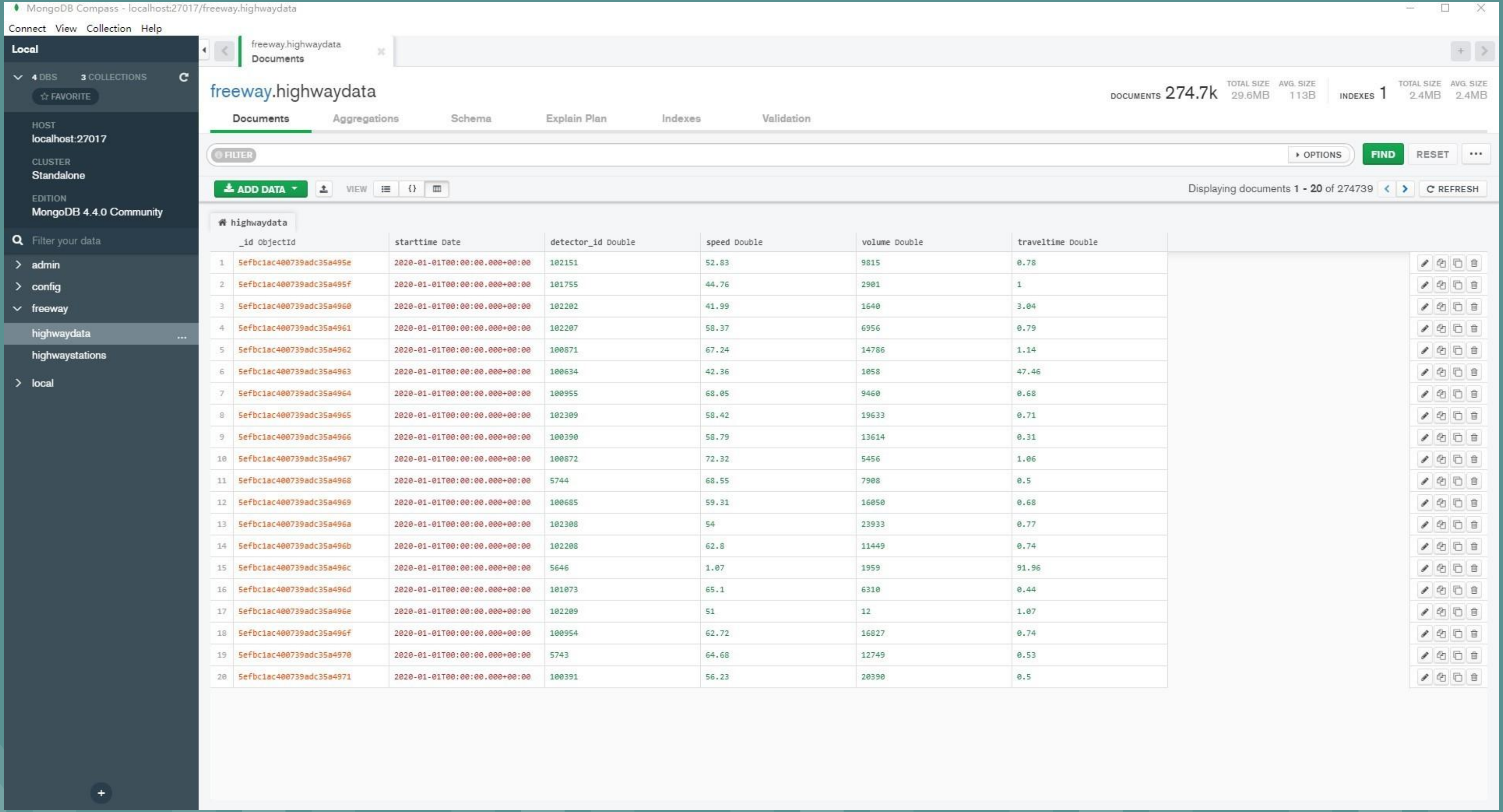
Raw Data

filtrate

Restructured Data

present

Result



HOST
localhost:27017

CLUSTER
Standalone

EDITION
MongoDB 4.4.0 Community

freeway.highwaydata

DOCUMENTS 274.7k TOTAL SIZE 29.6MB AVG. SIZE 113B INDEXES 1 TOTAL SIZE 2.4MB AVG. SIZE 2.4MB

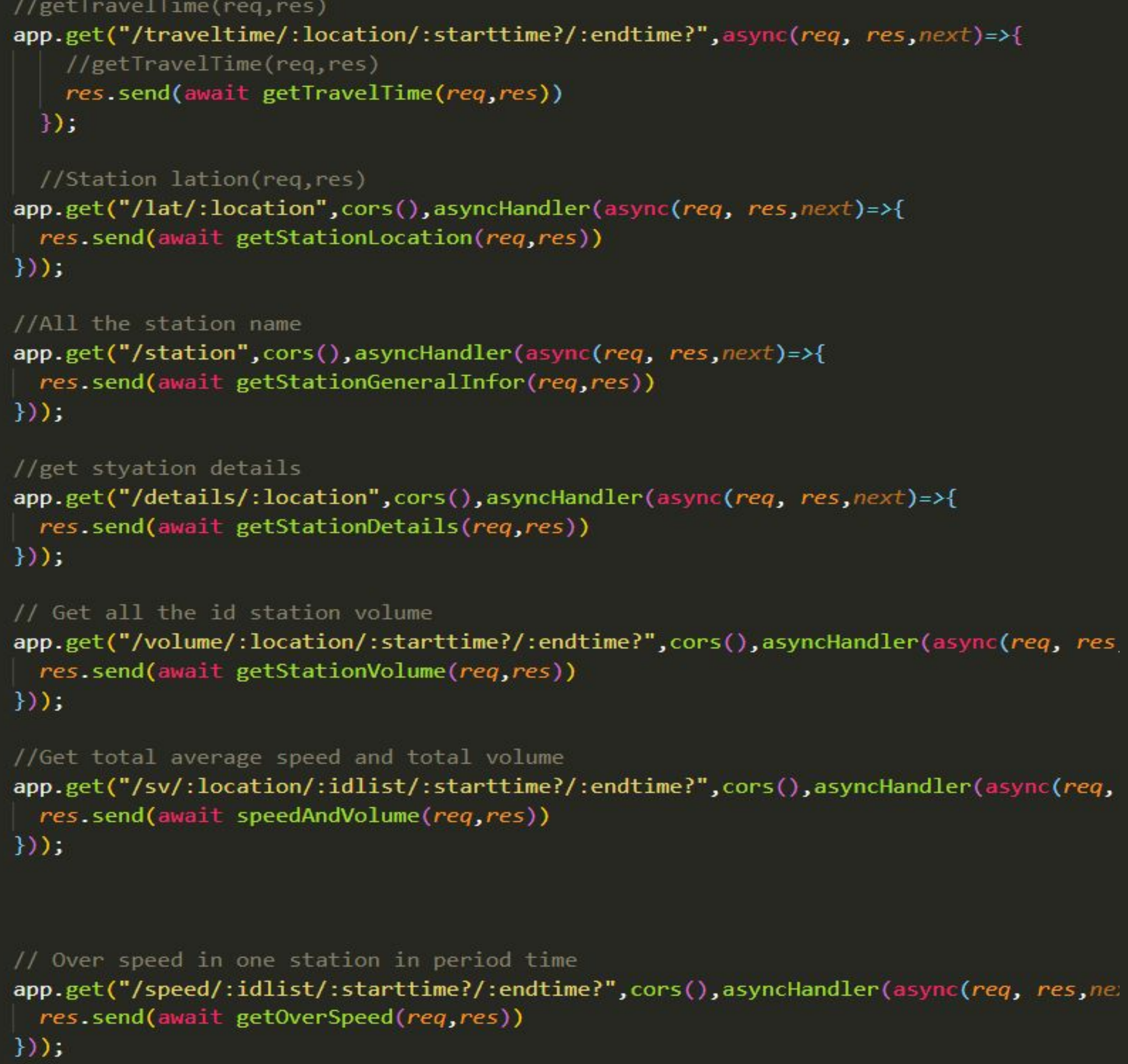
Displaying documents 1 - 20 of 274739

	_id ObjectId	starttime Date	detector_id Double	speed Double	volume Double	traveltime Double
1	5efbc1ac400739adc35a495e	2020-01-01T00:00:00.000+00:00	102151	52.83	9815	0.78
2	5efbc1ac400739adc35a495f	2020-01-01T00:00:00.000+00:00	101755	44.76	2901	1
3	5efbc1ac400739adc35a4960	2020-01-01T00:00:00.000+00:00	102202	41.99	1640	3.04
4	5efbc1ac400739adc35a4961	2020-01-01T00:00:00.000+00:00	102207	58.37	6956	0.79
5	5efbc1ac400739adc35a4962	2020-01-01T00:00:00.000+00:00	100871	67.24	14786	1.14
6	5efbc1ac400739adc35a4963	2020-01-01T00:00:00.000+00:00	100634	42.36	1058	47.46
7	5efbc1ac400739adc35a4964	2020-01-01T00:00:00.000+00:00	100955	68.05	9460	0.68
8	5efbc1ac400739adc35a4965	2020-01-01T00:00:00.000+00:00	102309	58.42	19633	0.71
9	5efbc1ac400739adc35a4966	2020-01-01T00:00:00.000+00:00	100390	58.79	13614	0.31
10	5efbc1ac400739adc35a4967	2020-01-01T00:00:00.000+00:00	100872	72.32	5456	1.06
11	5efbc1ac400739adc35a4968	2020-01-01T00:00:00.000+00:00	5744	68.55	7908	0.5
12	5efbc1ac400739adc35a4969	2020-01-01T00:00:00.000+00:00	100685	59.31	16050	0.68
13	5efbc1ac400739adc35a496a	2020-01-01T00:00:00.000+00:00	102308	54	23933	0.77
14	5efbc1ac400739adc35a496b	2020-01-01T00:00:00.000+00:00	102208	62.8	11449	0.74
15	5efbc1ac400739adc35a496c	2020-01-01T00:00:00.000+00:00	5646	1.07	1959	91.96
16	5efbc1ac400739adc35a496d	2020-01-01T00:00:00.000+00:00	101073	65.1	6310	0.44
17	5efbc1ac400739adc35a496e	2020-01-01T00:00:00.000+00:00	102209	51	12	1.07
18	5efbc1ac400739adc35a496f	2020-01-01T00:00:00.000+00:00	100954	62.72	16827	0.74
19	5efbc1ac400739adc35a4970	2020-01-01T00:00:00.000+00:00	5743	64.68	12749	0.53
20	5efbc1ac400739adc35a4971	2020-01-01T00:00:00.000+00:00	100391	56.23	20390	0.5

Data Source: https://portal.its.pdx.edu/home

Data Restructure

- Convert relational data into nosql format
- Pre-process the Data



```
//getTravelTime(req,res)
app.get("/traveltime/:location/:starttime?/:endtime?",async(req, res,next)=>{
  //getTravelTime(req,res)
  res.send(await getTravelTime(req,res))
 });

 //Station lation(req,res)
app.get("/lat/:location",cors(),asyncHandler(async(req, res,next)=>{
 res.send(await getStationLocation(req,res))
}));

//All the station name
app.get("/station",cors(),asyncHandler(async(req, res,next)=>{
 res.send(await getStationGeneralInfor(req,res))
}));

//get styation details
app.get("/details/:location",cors(),asyncHandler(async(req, res,next)=>{
 res.send(await getStationDetails(req,res))
}));

// Get all the id station volume
app.get("/volume/:location/:starttime?/:endtime?",cors(),asyncHandler(async(req, res
 res.send(await getStationVolume(req,res))
}));

//Get total average speed and total volume
app.get("/sv/:location/:idlist/:starttime?/:endtime?",cors(),asyncHandler(async(req,
 res.send(await speedAndVolume(req,res))
}));



// Over speed in one station in period time
app.get("/speed/:idlist/:starttime?/:endtime?",cors(),asyncHandler(async(req, res,ne
 res.send(await getOverSpeed(req,res))
}));
```

Project Demo

Lesson

- Not only data itself tell us the fact, whether data exist itself also tell us the fact
- Relational data convert to NoSQL data takes time, and the barrier between them reduce the efficiency of data processing
- Two stage require most coding: data processing & data presenting
- BTW: Working on something complex enough that we were forced to quickly learn new things

Thank You

IS THIS DATA ACCURATE?
YOU DON'T GO TO WAR WITH THE DATA YOU NEED. YOU GO TO WAR WITH THE DATA YOU HAVE.
DID YOU JUST MAKE IT SOUND NOBLE TO USE BAD DATA?
AND HEROIC.
DILBERT.COM
@SCOTTADAMSSAYS
5-7-20 2020 Scott Adams, Inc./Dist. by Andrews McMeel

Reference

Project GitHub: https://github.com/data-science-pdx/freeway-analysis

Data Source: https://portal.its.pdx.edu/home

Additional Slides

Database Related Feature

- ★ The feature insert data from CSV file into mongoDB
- ★ Restructure the data from relation data scheme to NoSQL database
- ★ Optimize all the queries that reduce the response time from 30 sec to fail within 3 sec which improve user experience, and provide better performance.
- ★ Database partitions.

The performance was improved by optimize the structure of the restructured data.

Backend Feature

- ❖ Using Node.JS as the backend to hook with MongoDB to design APIs to process the necessary data.
- ❖ The backend will process the data and pass the required data to frontend, including bad data and reasons.

APIs:

- Speed restrict APIs: Overspeed (more than 100), Low Speed (less than 5), Normal Speed
- Station details API: Provide Station details information such as latitude, longitude, station name, etc.
- Detectors info API: Provide the relationship between detectors and stations
- Other APIs: Fetch all the station information, more than ten. (e.g. travel time, capacity)

Things we tried but failed:

We want to use Elasticsearch and MongoDB together at beginning. However after we did a lot of searches, we cannot make it happen.

There are a few possible ways to do it.

- First way is use the third part mongo connect, but it does not support Elasticsearch 7.X and Mongo 4.X.
- Second way is the logstash, it only supports the Elasticsearch, there is no official MongoDB version.

If we have more time

- **We will try to do more data serialization.**
  - The current data are present street forward, the more organized data present format is what we want to achieve, which bring user better idea how detectors works.
- **We will separate them by using ML.**
  - The current design is we leave bad (which missing some values, or the data is meaningless) and good data in the same data table, it will drop down the performance.