John Blakkan Project Update

Rohit Nair W205

Andrea Pope

### Taxis in the Mist: Weather Impact on NYC Taxi Use

#### Summary of Project:

Our project is analyzing data provided by the New York City Taxi & Limousine Commission (TLC) covering over 1.1 billion individual taxi trips in the city between January 2009 and June 2016. We will complement this with historical weather data to model how weather impacts taxi use. We will use this data to predict upcoming NYC taxi needs based on predicted weather for the next 6 days.

#### Data Sources:

Three primary data sources are being used:

1. ***Historical Taxi Data***: Data represents taxi ridership data for three services (“Yellow”, “Green”, and “For Hire Vehicles”) for 2009 - 2016. Initial analysis was completed on publicly available sources of this data.

Each taxi service provides monthly .csv files giving fields such as pickup time, pickup location, number of passengers, etc. Note that not all the services provide data for the whole time period. For each of these files, the following batch ETL process is underway:

* Copy from public URL to AWS as .csv file
* Strip header rows
* Moved (put) to HDFS
* Map file to a HIVE table
* Transform raw data into schema suitable for processing

The initial pilot implementation covers approximately 100 Million records, representing 6 months data for all three services. More data will be incorporated following prove-out of the pilot. For the larger dataset, the following are under consideration

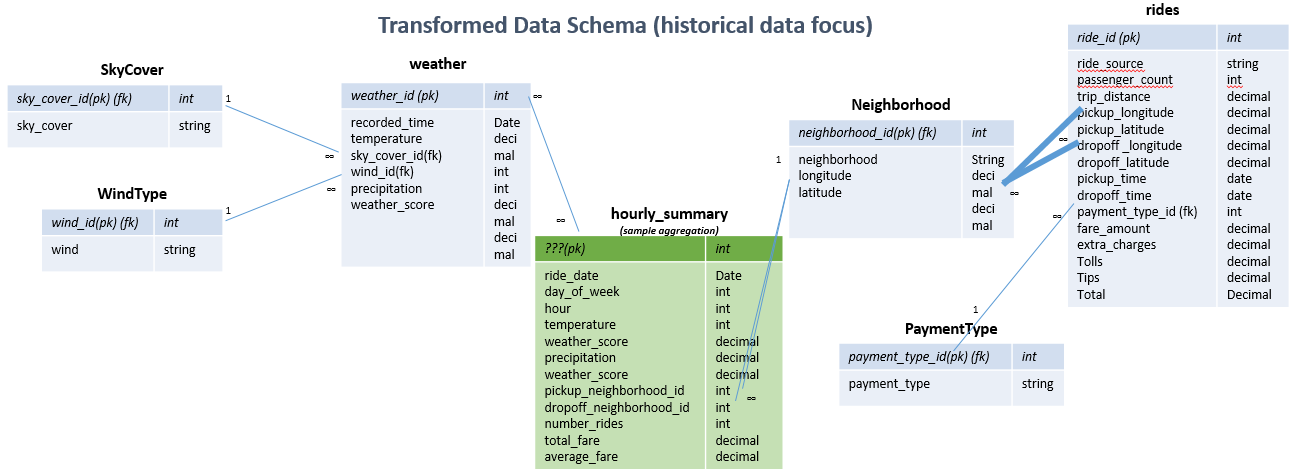
* Moving from AWS EBS to AWS S3 (for all but last year)
* Compression options within HDF
* Eliminating raw datafiles incrementally as they’re used to create the model

1. ***Historical Weather Data***: Similar to the ride data, historical weather data for NYC is stored in yearly files, representing daily weather conditions. The same basic batch process is used for these data files as the ride data, with appropriate changes to Data Definitions/Schema. The weather data will be JOINed with the taxi data, to give the inputs for building the predictive model.
2. ***Six-day Weather Forecasts***: New York City weather forecasts in XML format are available via a RESTful web interface from the US Government (NOAA). There are unique ETL concerns for this data source. The service has been seen to become unresponsive for minutes at a time, and when available, response time varies from under 2 seconds to over 15 seconds. We are planning to use local caching of data from this service (timestamped to warn of any “stale” data) to overcome this performance and reliability issue. Although low bandwidth, this is effectively a “speed layer.” As well as a reliability enhancement. ETL process is:

* Periodically pull data from interface and XML parse
* Reformat to new table DDL
* Log to local postgres table, with timestamp

Ride and Weather data schema

Ride and weather tables will be JOINED and used to build a prediction model. The image below represents the current work-in-progress (WIP) ERD supporting this transformation.

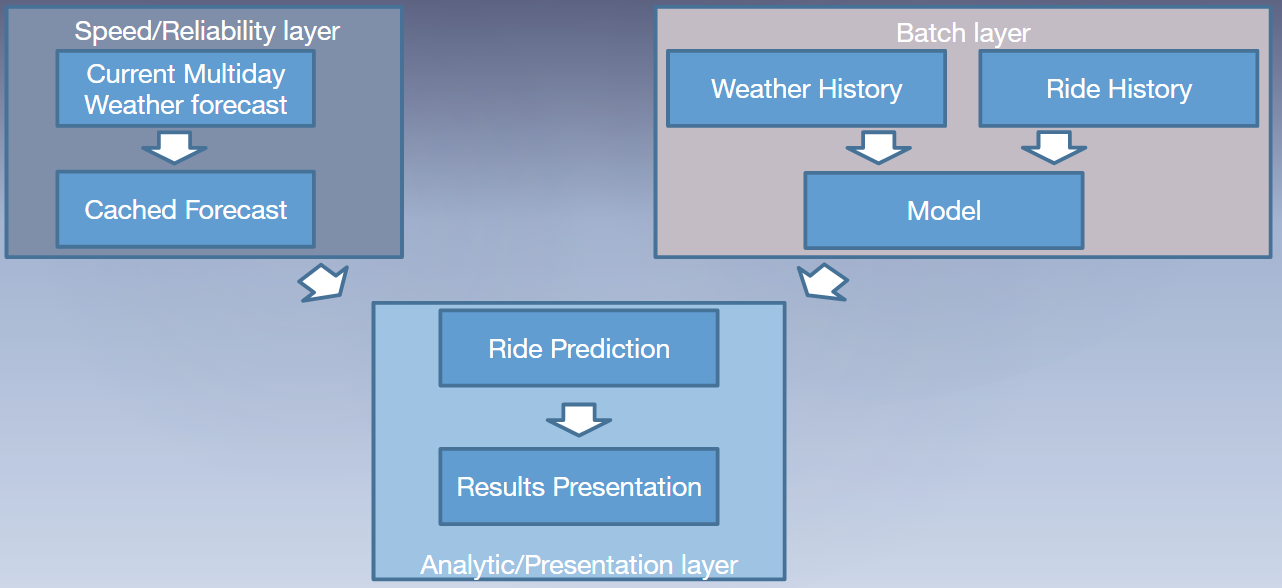


Prediction model/analytics/results presentation:

The model will return the predicted number of pickups and average number of passengers as a function of: hour-of-day, day-of-week, week-of-year (to capture seasonality), temperature, location, and precipitation. Pickup location data will be aggregated into a small number of districts (current plan is to use a simple North, Mid, and South division, since the intent is to preposition cabs in general areas, not specific neighborhoods.).

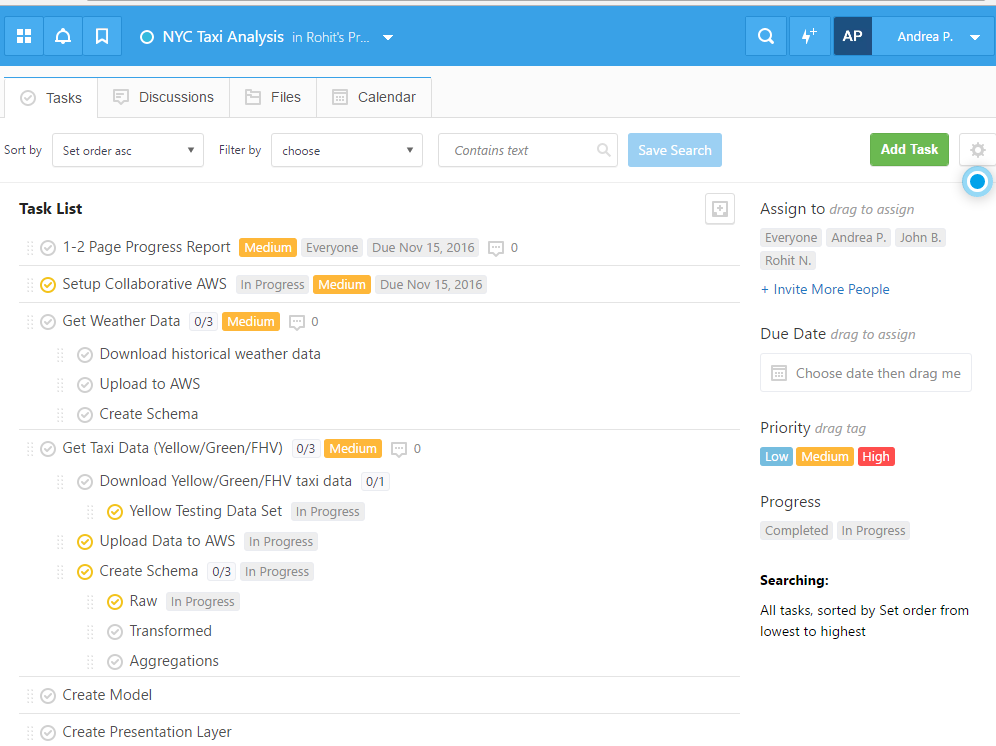
The final predicted taxi needs, for each of the 6 days of weather forecast will be presented using Tableau.

The following image is a representation of this process:



#### Project management and status:

The team is using a combination of FreedCamp and github to work together. Freedcamp allows us to update/add project tasks, and monitor team progress. Github allows us to share files and code with each other and the AWS instance.



#### Appendix

As a reference point, the raw data files (external tables) have the following structure (also WIP):

