Final Project Proposal

Team 3

Team members: Hongwei Hu, Guangnan Liang

Boston Air Quality Analysis

1. What is Geospatial Data?

The word geospatial is used to indicate that data that has a geographic component to it. This means that the records in a dataset have locational information tied to them such as geographic data in the form of coordinates, address, city, or ZIP code. GIS data is a form of geospatial data. Other geospatial data can originate from GPS data, satellite imagery, and geotagging.

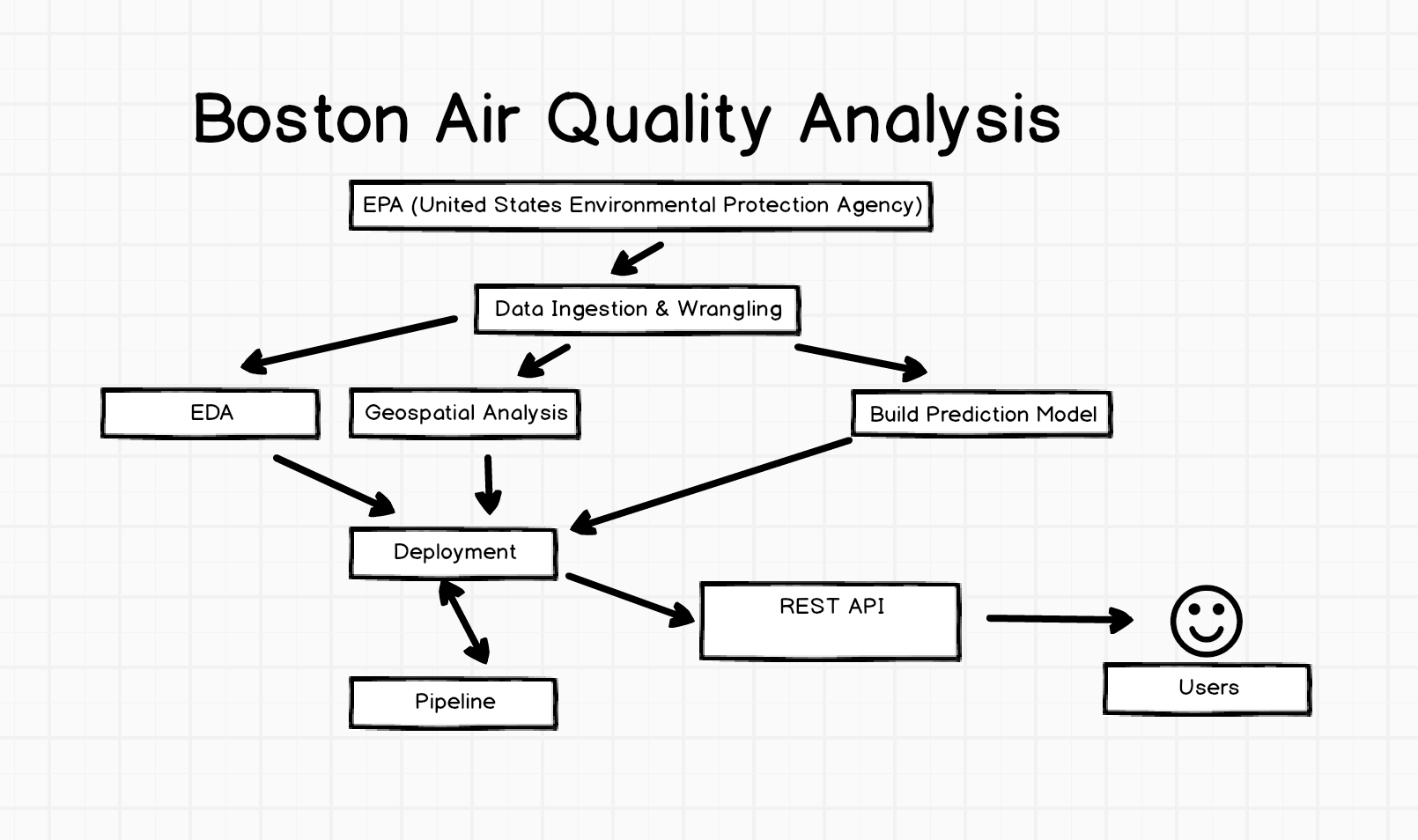
2. For Whom?

For people who are interested in air quality of Boston. Of course, it’s easy to implant for other locations if you have the air quality data of them.

3. Dataset:

* Data source: EPA (United States Environmental Protection Agency)
* Outdoor Air Quality Data:
  + url: <https://www.epa.gov/outdoor-air-quality-data/download-daily-data>
  + api: <https://aqs.epa.gov/api>
    - The daily summaries come directly from the AQS (Air Quality System) Data Mart which contains ambient air pollution data collected by EPA, state, local, and tribal air pollution control agencies.
    - Pollutants:
      * CO, Pb, NO2, Ozone, PM10, PM2.5, SO2

4. Workflow



5. Process details:

1. Data Ingestion and Wrangling:
   * Preprocess:
     1. EDA: missing data, and etc.
     2. For Prediction: Pick up one station (Boston-Cambridge-Newton, MA-NH, Suffolk), collect the past five years’ pollutants on this station, and merge all the information to produce a new dataset for prediction.
   * This step could be automated, and data will be uploaded to S3.
2. Geospatial Analysis:
   * ArcGIS: using ArcGIS online mapping tools to work with maps and geographic information. It is used to create and use maps, compile geographic data, and analyze mapped pollutants’ information.
3. EDA: create a Jupyter notebook to perform EDA on each pollutant dataset and analyze the relationship among pollutants in the corresponding period.
4. Build model to predict Air Quality (can be digital values or categories) based on its air quality data:
   * Linear Regression/Classification
   * Decision Forest Regression/Classification
   * Neural Network Regression/Classification
5. REST API:
   * Notebook: the APIs provide links to jupyter notebook so user can view the virtualization results.
   * Prediction service: using flask to design some APIs that present the results of prediction.
   * Docker (optional): allow user to download the deployed docker image to local system
6. Deployment
   * Azure: model, notebook
   * pythonanywhere or AWS: API
   * Docker: whole project
7. Pipeline:
   * Build pipeline to get newest data every month, and upload it to S3.
   * Retrain models every month based on newest air quality data.

6. User Interface

