CZAR OF HACKATHON

Respira

One good turn deserves another
One good man deserves a purifier

OVERVIEW

- Introduction
- Goal
- Setting & Implementation
- Feature
- Vision
- Reference

INTRODUCTION

An optimized, customized system

- Develop a automatic scoring system.
- Develop a mathematical model for power allocation.
- Design a simple framework for data visualization, processing, scraping, and training.
- Visualization
  - Visualize air quality and air data dashboard.
  - Process history data and visualize correlation between data.

GOAL - 1

Automatic Scoring System

- Step 1 : label current air quality

  Labeling standards can be changed

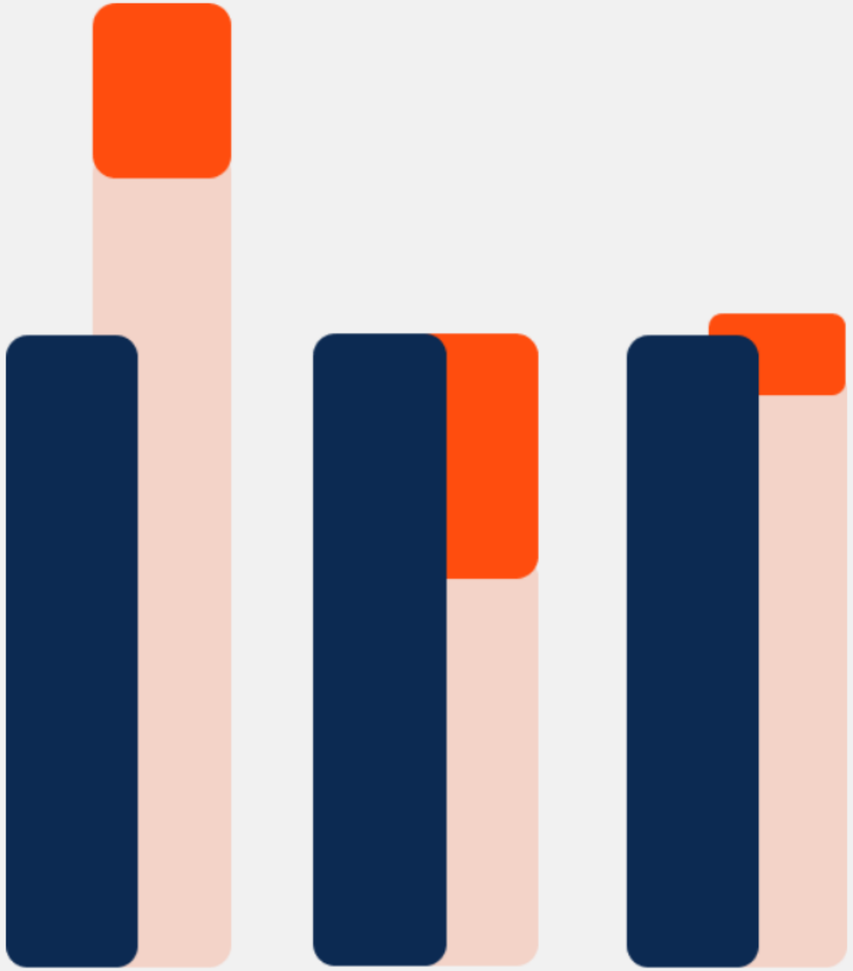
  - based on user feedback
  - based on discrete quality metrics (AQI)

- Step 2 : Use logistic regression to fit the labels

  Each air data vector will then be associated with a probability between 0~1

- Step 3 : Visualize real-time score on the webpage

GOAL - 2 POWER ALLOCATE OPTIMIZATION



- Scores after allocation
- Scores without algorithm
- Wasted Power

GOAL - 2 POWER ALLOCATE OPTIMIZATION

- Can we develop a mathematical model to better allocate power rather than divide power averagely ? Scenarios with lower scores should be allocated more power.

$Total\ of\ m\ scenarios: 1 \sim m$
$A\ data\ vector\ at\ i_{th}\ location\ is\ denoted\ as\ x^i = [x_1^i \quad x_2^i \quad \dots\dots x_n^i]\,, x^{i_o}\ is\ the\ initial\ measurement$

$Suppose\ we\ wish\ to\ allocate\ a\ total\ of\ power\ of\ P\ to\ m\ locations.$

$The\ power\ allocated\ to\ the\ i_{th}\ location\ is\ a_i W, hence \sum_{i=1}^{m} a_i = 1.$

$We\ model\ the\ relation\ between\ power\ and\ the\ score\ as\ linear, that\ is:$

$$x^{i_f} = x^{i_o} + a_i V, \qquad V = [V_1 \quad V_2 \quad \dots\dots V_n]\,.$$

$The\ scaling\ factor\ V\ can\ be\ determined\ using\ past\ collected\ air\ purifier\ data.$

GOAL - 2 POWER ALLOCATE OPTIMIZATION

$$Objective:$$

$$[a_1 \quad a_2 \quad \dots\dots a_n] = \ argmin \ Var\left[\ f\left(x^{i_f}\right)\right], \text{subject to} \sum_{i=1}^{m} a_i = 1. f(x) \ is \ the \ score \ function.$$

$$We \ solve \ this \ by \ using \ the \ lagrange \ multiplier \ method.$$

$$T = \ Var\left[\ f\left(x^{i_f}\right)\right] + \ \gamma(1 \ - \sum_{i=1}^{m} a_i)$$

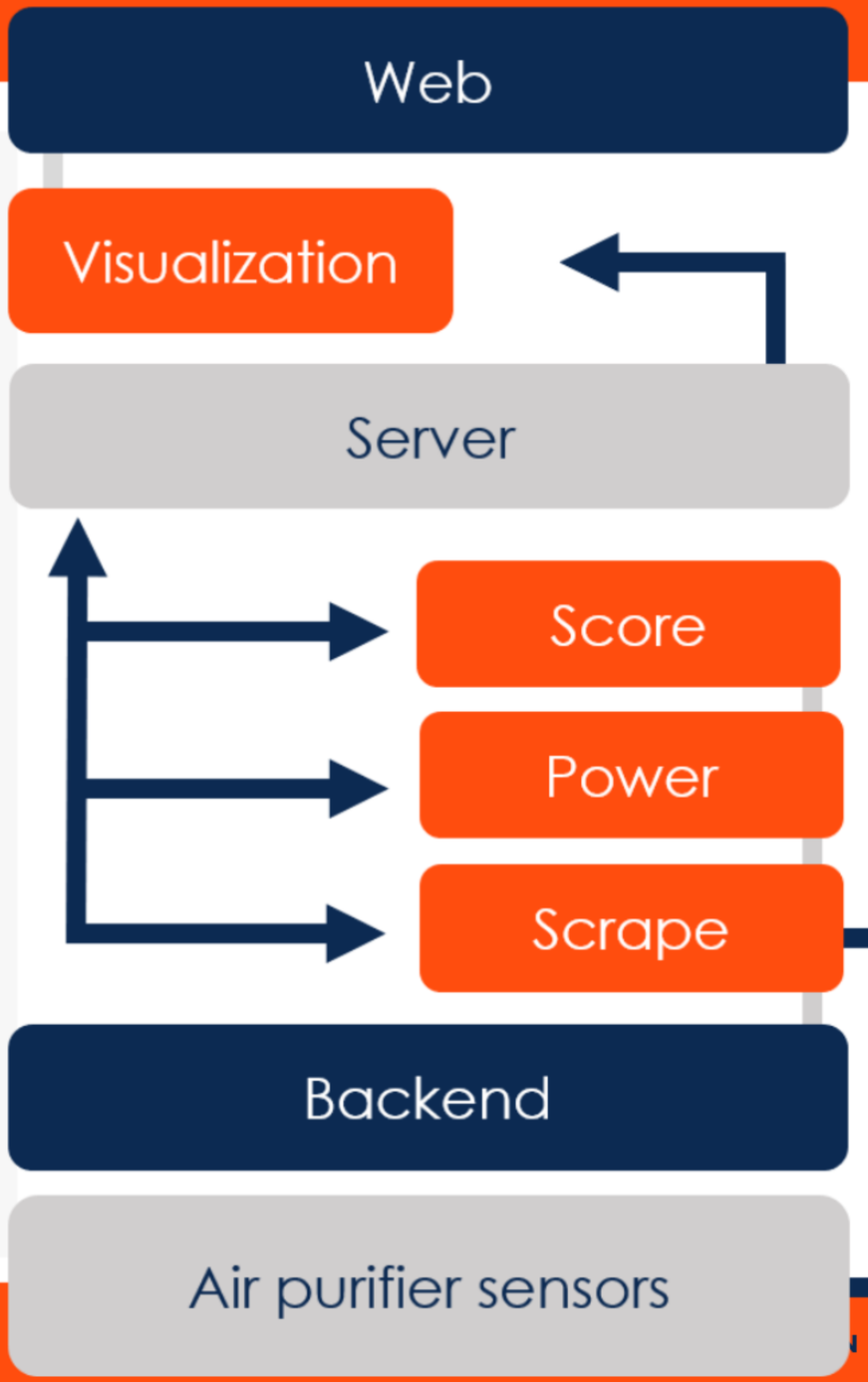
$$\frac{\partial T}{\partial a_i} = 0$$

$$\frac{\partial T}{\partial v} = \ (1 \ - \sum_{i=1}^{m} a_i) = 0$$

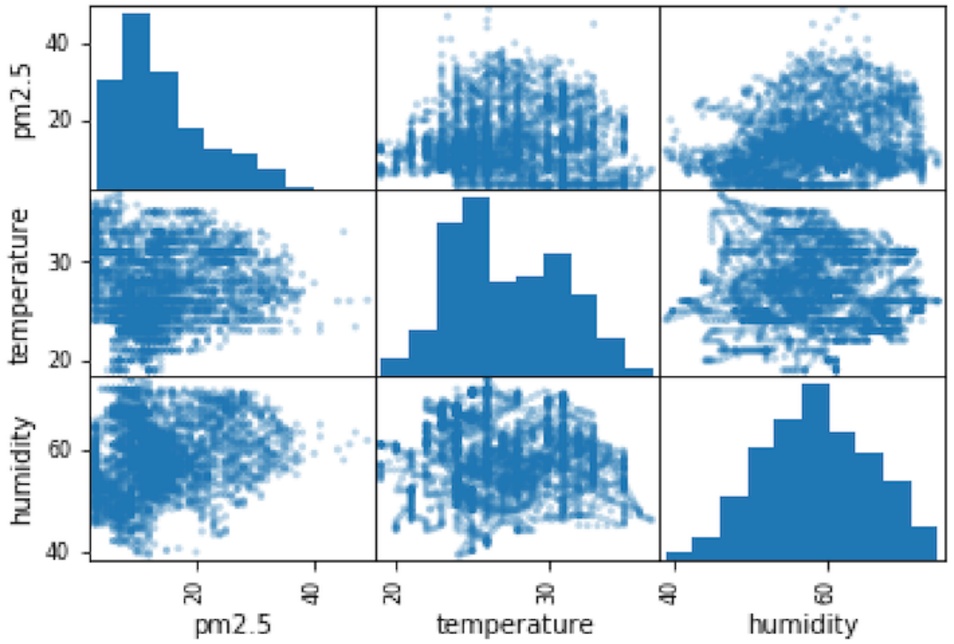
GOAL - 3

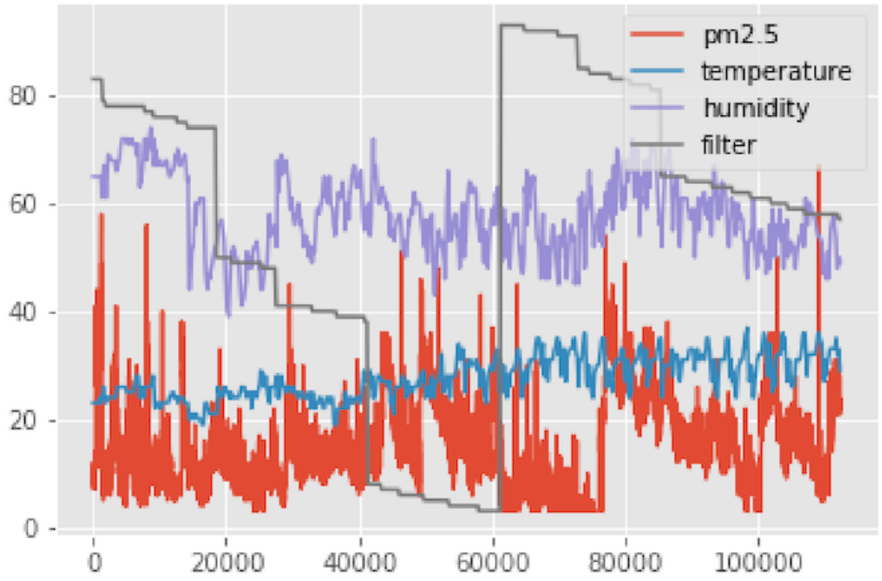
Framework

```
├── Respira
    ├── web
    │   ├── index.html
    │   ├── js
    │       ├── ...
    │
    ├── server.py
    │
    ├── collected_data
    │   ├── location_1
    │   ├── ...
    │
    ├── backend
    │   ├── score
    │   │   ├── training
    │   │   │   ├── model.py
    │   │   │   ├── ...
    │   │   ├── compute_score.py
    │   │
    │   ├── power
    │   │    ├── allocate_power.py
    │   │    ├── ...
    │   │
    │   ├── scrape.py
```



GOAL - 4 Visualization





Settings & Implementations

- No real-time data for visualization: we model various air data as a **gaussian distribution**. The parameters are calculated using **Maximum Likelihood** Estimation based on history data.
- Length mismatch between air mearsurements: simply augment data using interpolation.

Feature

VISUALIZE OPTIMIZATION

1. Eco-friendly

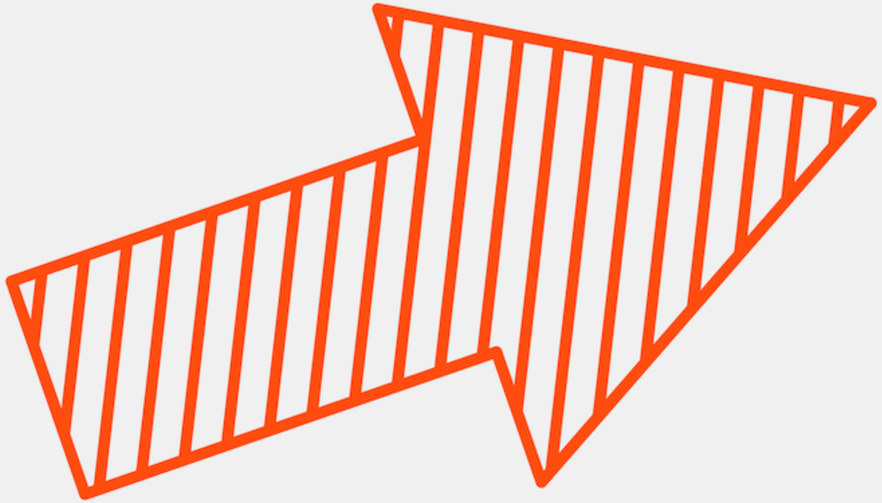
2. Danger Detecting

3. Economic

VISION

Olfactory Reality

Generate different types of air



Filter different types of air

REFERENCE

WHO Air pollution

https://www.who.int/airpollution/en/

Environmental Protection Administration Execution Yuan

https://taqm.epa.gov.tw/taqm/tw/default.aspx

III DSI

https://www.facebook.com/iii.dsi/

Q & A Time

Thanks for listening