

# Real-time Weather Forecasting using Autonomous Deep Learning

Shijia Yan

# Motivation

- Traditionally weather forecasting through physical simulations(NWP) needs huge amount of computing power

BBC BBC

## Met Office and Microsoft to build climate supercomputer

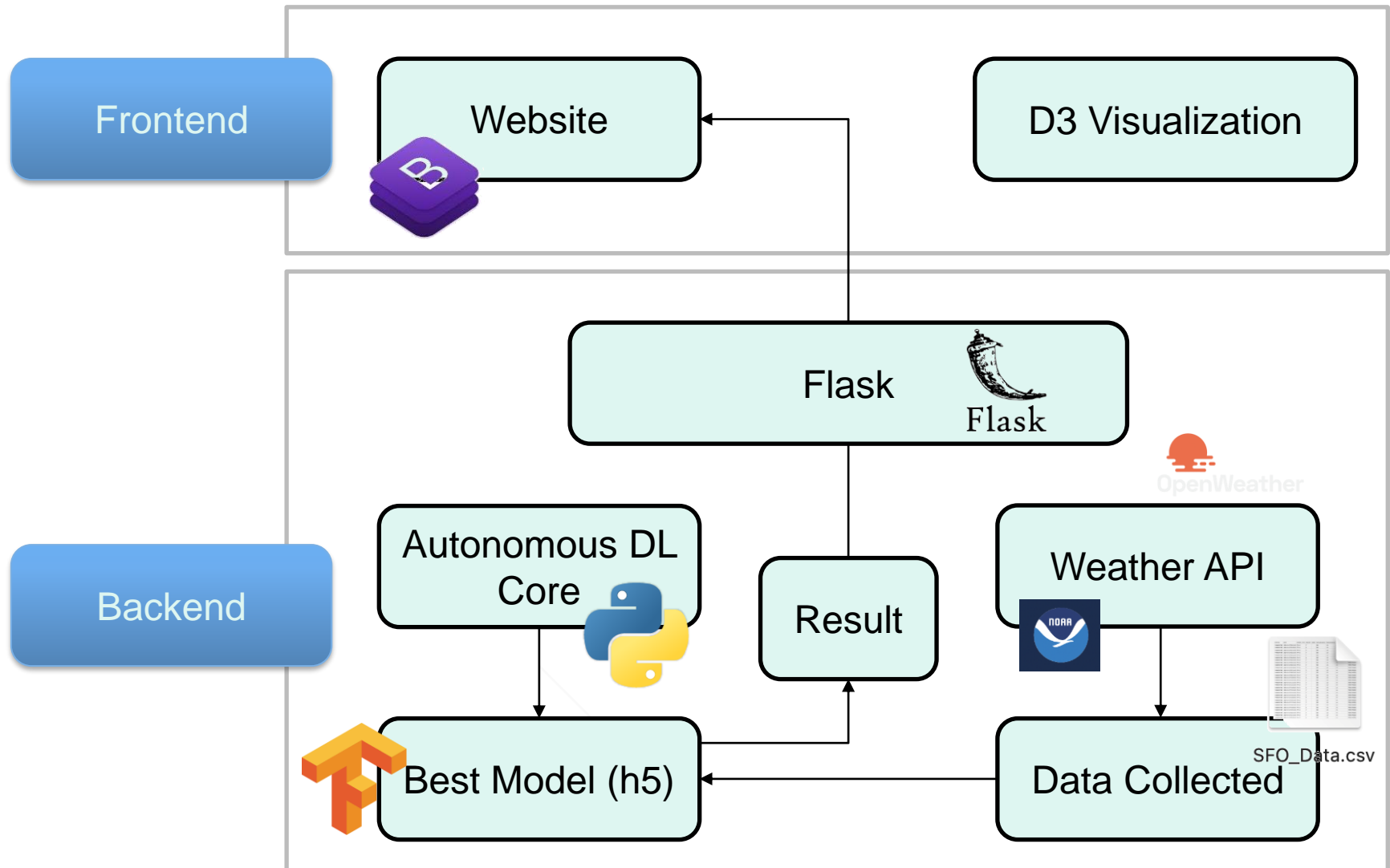
The Met Office is working with Microsoft to build a weather forecasting supercomputer in the UK. They say it will provide more accurate weather ...  
2 days ago



- Machine learning applications can achieve relatively high accuracy while using much less resources – suitable for online service

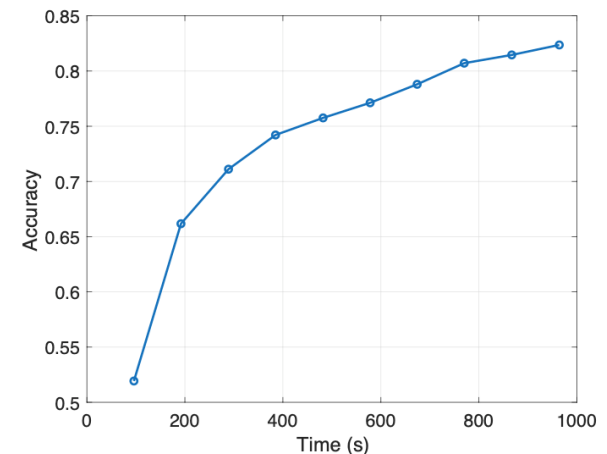
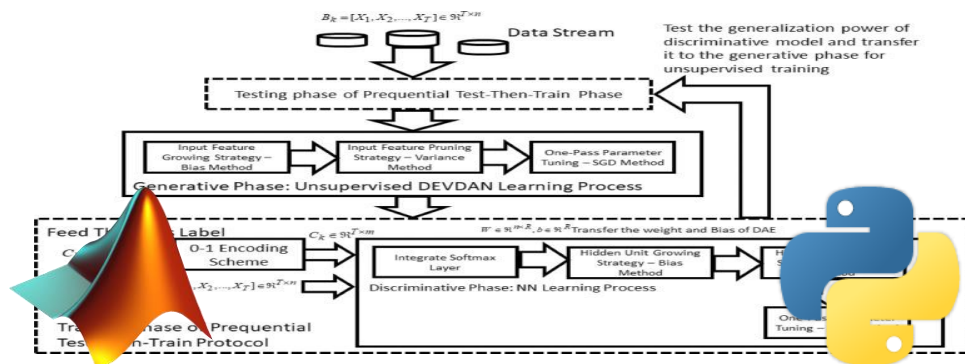
Architecture	Reference	Recent Developments
convolutional neural network (CNN)	AlexNet (Alex <i>et al.</i> 2012) VGG (Simonyan & Zisserman, 2013) ResNet (He <i>et al.</i> 2015) GoogleLeNet (Szegedy <i>et al.</i> 2015)	VGG (Shi <i>et al.</i> 2018 [66]) ResNet (Pothineni <i>et al.</i> 2019 [68]) Vgg, ResNet (Wen <i>et al.</i> 2020 [70]) Inception blocks (Kleinert <i>et al.</i> 2021 [72])
long short-term memory network (LSTM)	RNN (Bengio <i>et al.</i> 1994) LSTM (Gómez <i>et al.</i> 2003)	LSTM (Qing & Niu, 2018) PhyDNet (Le Guen, 2020)
variational autoencoder (VAE)	Vanilla VAE, 2013	None
generative adversarial neural network (GAN)	Vanilla GAN (Goodfellow <i>et al.</i> , 2014)	MD-GAN (Xiong, 2018) conditional GAN (Schmidt <i>et al.</i> 2020)

# Workflow overview

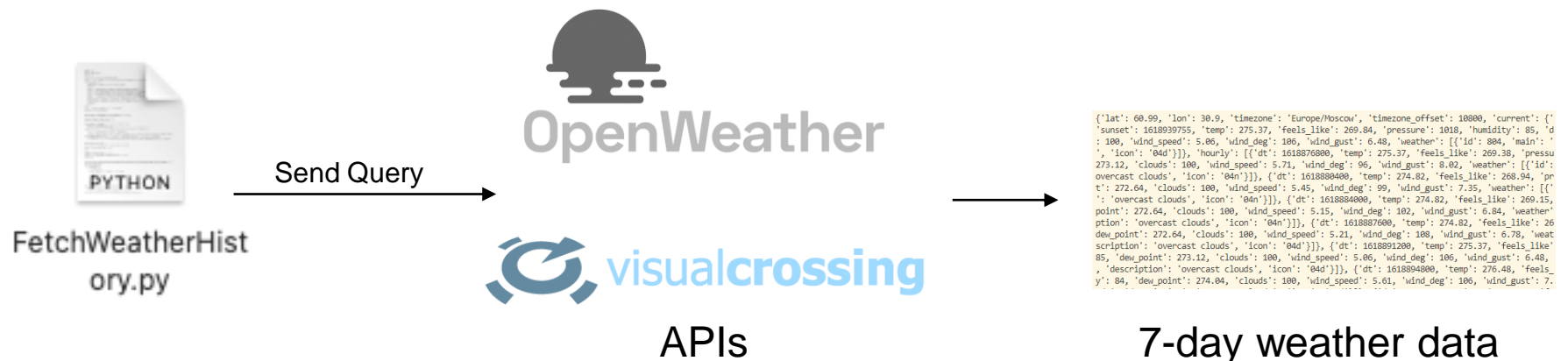


# Deep Learning Core

- Implemented DEV DAN with weather data(originally in Matlab, rebuilt in Python 2.7)
- Training dataset: NOAA historical data, 3 years, 20 locations, more than 20,000 days of data
- Weather data: daily reported data, consisting of precipitation, snow, wind, sea-level pressure, cloud
- Data chunk: 14-days each chunk
- Best model saved in .h5, ready for prediction



# Data Fetching



- Visualcrossing provides 1000 free results per day with 50 years of history
- Open Weather only provides past 5 days data for free
- API key was obtained by creating account at these websites
- Data was then stored and filtered to only high-low temperature, wind, cloud, precipitation and pressure
- Use `tf.Session()` from keras to load model and generate prediction results

# Challenges

- Challenges:
  - Volume: large amount of data from weather history for training.
    - Data filtering using python, stored in csv file → ready for training
    - Store pre-trained model into .h5, avoid training on-the-fly with whole dataset
  - Velocity: long training time
    - Use pre-trained model to generate prediction results
    - Update model by training new models in the background once in a while
  - Variety: worldwide locations coverage
    - Hard to obtain worldwide weather data
    - Data in most US cities are available through multiple API websites

# Online Forecasting

```
{
  "lat": 33.44,
  "lon": -94.04,
  "timezone": "America/Chicago",
  "timezone_offset": -21600,
  "current": {
    "dt": 1618317040,
    "sunrise": 1618282134,
    "sunset": 1618333901,
    "temp": 284.07,
    "feels_like": 282.84,
    "pressure": 1019,
    "humidity": 62,
    "dew_point": 277.08,
    "uvi": 0.89,
    "clouds": 0,
    "visibility": 10000,
    "wind_speed": 6,
    "wind_deg": 300,
    "weather": [
      {
        "id": 500,
        "main": "Rain",
        "description": "light rain",
        "icon": "10d"
      }
    ],
    "rain": {
      "1h": 0.21
    }
  }
},
```

Typical response  
from OpenWeather

```
{
  "dt": 1618317040,
  "temp": 284.07,
  "pressure": 1019,
  "clouds": 0,
  "wind_speed": 6,
  "rain": 0.21
}
```

Data extracted in json()

numpy

np.array

tf.Session()

model.predict()

Also normalizing all  
data to [0,1]  
e.g.: temp data  
normalized with range  
(100,350) (K)

One-hot vector:  
['Sunny', 'Cloudy',  
'Rainy', 'Foggy',  
'Snowy']

```
model =  
load_model('best_model.h5')
```

best\_model.h5

Tensorflow

run\_service.py

Flask

weather\_forecastin  
g\_website.html

weather\_forecastin  
g\_results.html

# Weather Forecasting Page

## Choose your location

(Please choose one of the several locations from below)

✓ Choose...

New York

San Francisco

Seattle

Albany

Boston

Atlanta

San Jose

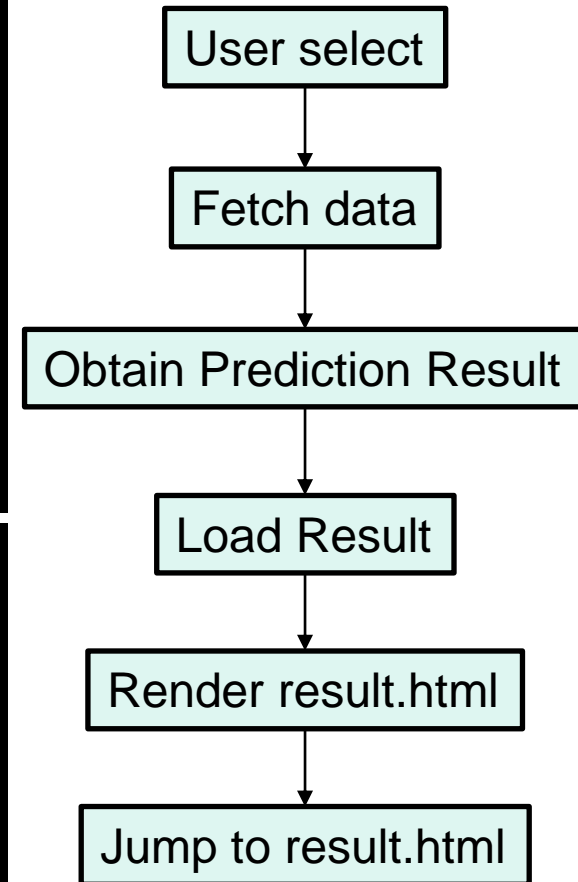
Chicago

## Your Weather Forecasting Result

My autonomous deep learning gave this prediction:

Rank	Weather Type	Possibility
Most likely	Sunny	0.95
2nd most likely	Rainy	0.03
3rd most likely	Cloudy	0.01

Try again?



I haven't complete the code using D3.js so we're stuck with this ☹



# Places for improvements

- Result visualization with D3: creating interactive chart page
- Construct higher-dimensional data to improve prediction accuracy
- Obtain data from more sources
- Develop into online Autonomous ML Platform for various other applications
- Q & A