Wildfire Challenge – Team PSR

Data Pre-Processing

The dataset for the h2o.ai wildfire challenge was taken from Rachael Tatman's 1.88 Million US wildfires dataset. It's available as a public dataset on Kaggle containing wildfires occurred in United States from 1992 to 2015. [1]

Dataset in the Kaggle is in the form of SQL lite database file. Database file is then converted into panda dataframe.

```
conn = sql.connect('Data.sqlite')
fire = pd.read_sql('SELECT * FROM Fires', conn)
fire.head(10)
```

After the dataframe is created the following data columns were dropped from the dataframe.

- Discovery Date (discovery_date)
- Discovery Time (discovery_time)
- Continuous Date (cont_date)
- Continuous Time (cont_time)

Acquiring Weather Data

Above dataset comprises with 1.88 million wildfire occurrences throughout the United states. The application is focused on making the predictions for a wildfire using the weather data of the incident is recorded. In order to minimize the weather data that needs to be taken from NASA Langley Research Centre [2] webpage, the geo-locations available in the dataset have been divided into a grid of 600 columns and 300 rows. The grid's corners are determined using the dataset's minimum and maximum values for latitude and longitude.

Corner of the Grid	Value
Bottom Left	17.9397, -168.87
Bottom Right	17.9397, -65.2569
Top Left	70.3306, -168.87
Top Right	70.3306, -65.2569

After the grid is created each record has been placed into the relevant cell inside and weather data for the centre point of the cell has been taken from NASA weather dataset mentioned above. Weather data for each wildfire occurrence has been taken for 7 days prior from the incident record date. Those weather data is then placed in each row with the wildfire dataframe.

Correlation with Fire_Size

Correlation matrix with fire size column has been then plotted to identify the most important features from the dataset.

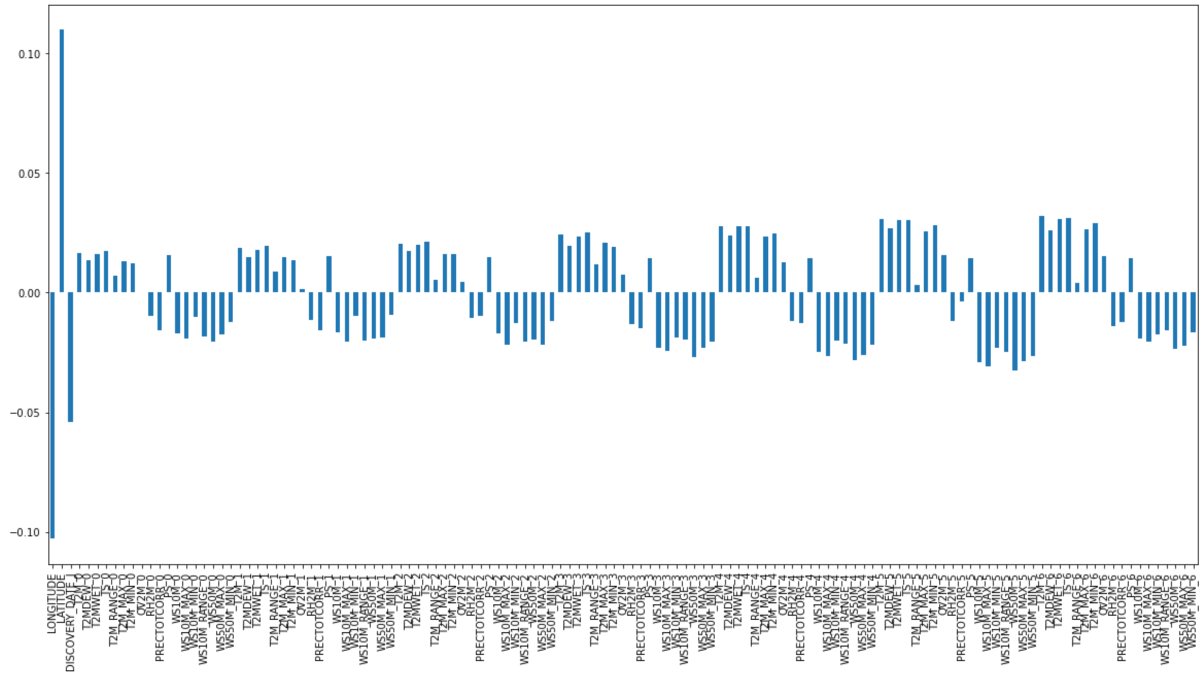

Figure 1 - Correlation Matrix

Considering the correlation values with the dependent variable (fire_size) following columns were further dropped from the dataframe.

Determining corelation values less than 0.008

```
Col_to_drop = []
for index,val in Plot_data.items():
    if(abs(val) < 0.008):
        Col_to_drop.append(index)
print(Col_to_drop)
```

OUTPUT

```
['T2M_RANGE_0', 'QV2M_0', 'QV2M_1', 'T2M_RANGE_2', 'QV2M_2', 'QV2M_3', 'T2M
_RANGE_4', 'T2M_RANGE_5', 'PRECTOTCORR_5', 'T2M_RANGE_6']
```

Model Building & Training

After pre-processing the raw data, dataset was saved as a CSV file format. Then it was added to the H2O Driverless AI platform in order to determine the best suitable model and parameters.

Prior to the dataset being processed, dataset was split into train, test & validation. The percentage for each set is respectively 70%, 15% & 15%.

Figure 2 - Dataset Split

After the dataset is prepared by splitting as above, an Experiment Setup was created inside the platform. Experiment setup was targeting “fire_size” column as the dependent variable and experiment type is set to “supervised”. Afterwards the experiment setup is started and following results has been obtained by the platform.

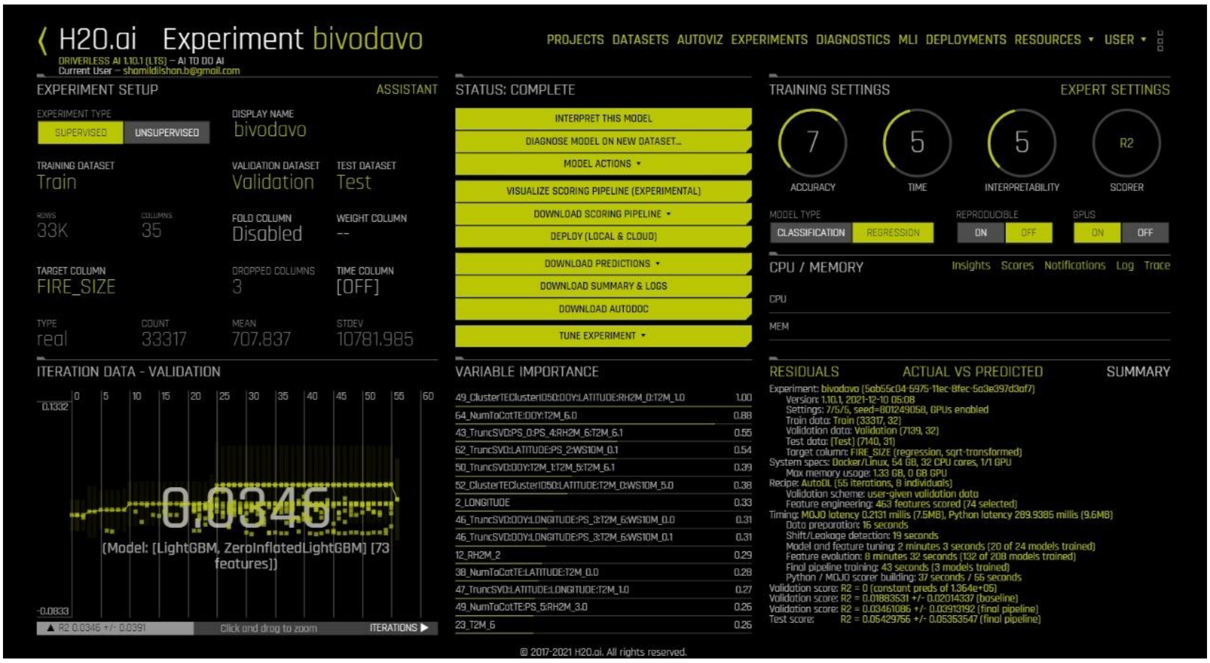

Figure 3 - Experiment Setup Results

“Autodoc” was generated and downloaded at the end of the experiment in order to identify the performance results for each model. The document results were used to determine the best model and its hyper-parameters. The best model that was considered for model building was “LightGBM”.

- Model Index: 0 has a weight of 0.5 in the final ensemble

Type	grow policy	index	learning rate	max depth	Split Type	max leaves	colsample bytree	subsample	model class name	tree method
LightGBM Model	lossguide	0	0.03	5	External	32	0.45	0.7	LightGBM Model	

Figure 4 - Hyper Parameter for LightGBM model

Above hyper-parameters were then utilized to build the model LGB regression model. Following code snippet represents the relevant section of the notebook which was used to create the model.

```
import lightgbm as lgb
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn import metrics

data = pd.read_csv('Data_final_Dropped.csv')
data.head()

# To define the input and output feature
x = data.drop(['FIRE_SIZE','DISCOVERY_DATE'],axis=1)
y = data.FIRE_SIZE
# train and test split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.3,random_state=42)

#Model parameters got based on Driverless AI output
model = lgb.LGBMRegressor(learning_rate=0.03,max_depth=5,random_state=42,num_leaves=32,subsample=0.7,colsample_bytree=0.45)
model.fit(x_train,y_train,eval_set=[(x_test,y_test),(x_train,y_train)],verbose=20,eval_metric='l2')

pred = model.predict(x_test)
print("Testing Accuracy (R Squre Value),")
print(metrics.r2_score(y_test,pred))

lgb.plot_importance(model)
lgb.plot_metric(model)

model.booster_.save_model("wildfire_detector.model")
```

Model Results

Following model results were taken after the model is saved. Training accuracy and testing accuracy in the following figure represents the **R2 Score.**

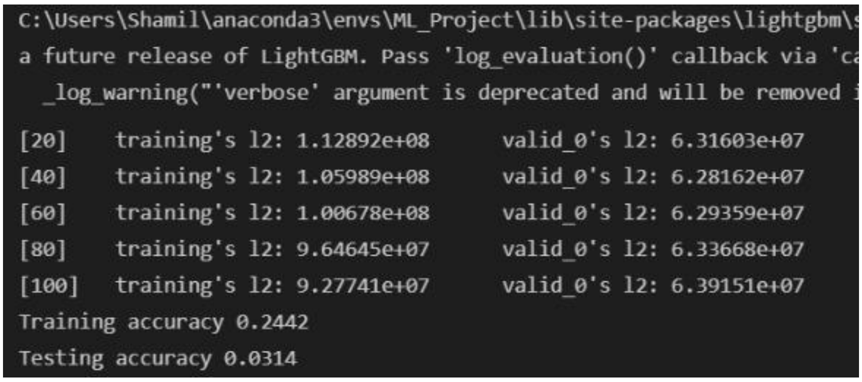

Figure 5 - Model Results

Prediction Application

Inputs to the model

The prediction system accepts several inputs if the user needs to get predictions for a wildfire. Following image shows the user interface for inputting data which is required for the system (Model) to predict the severity parameter for the given geo-location.

Geo-Location can be viewed using the "**Show Map**" button and selecting the desired point using the graphical map of the region. After the point is selected, coordination data can be input to the system manually using Latitude & Longitude.

The system needs user to pick a date which the user wants the system to predict the wildfire on the desired geo-location. Ideally the above application can be modified to fetch the current date and time from the Internet thus the user doesn't get to input the date. Main reason the application has designed in this way manner is due to complexity in updating weather data automatically. After the user interface provide necessary values to fetch the weather data from a pre-saved dataset which is not being automatically updated.

Table 1 - Input Parameters for the Application by User

Parameter	Description	Input Range
Latitude	Latitude of the geo-location	17.9397 to 70.3306
Longitude	Longitude of the geo-location	-178.8026 to -65.2569
Date	Date to fetch the weather data from pre-saved dataset	07/01/2021 to 01/12/2021

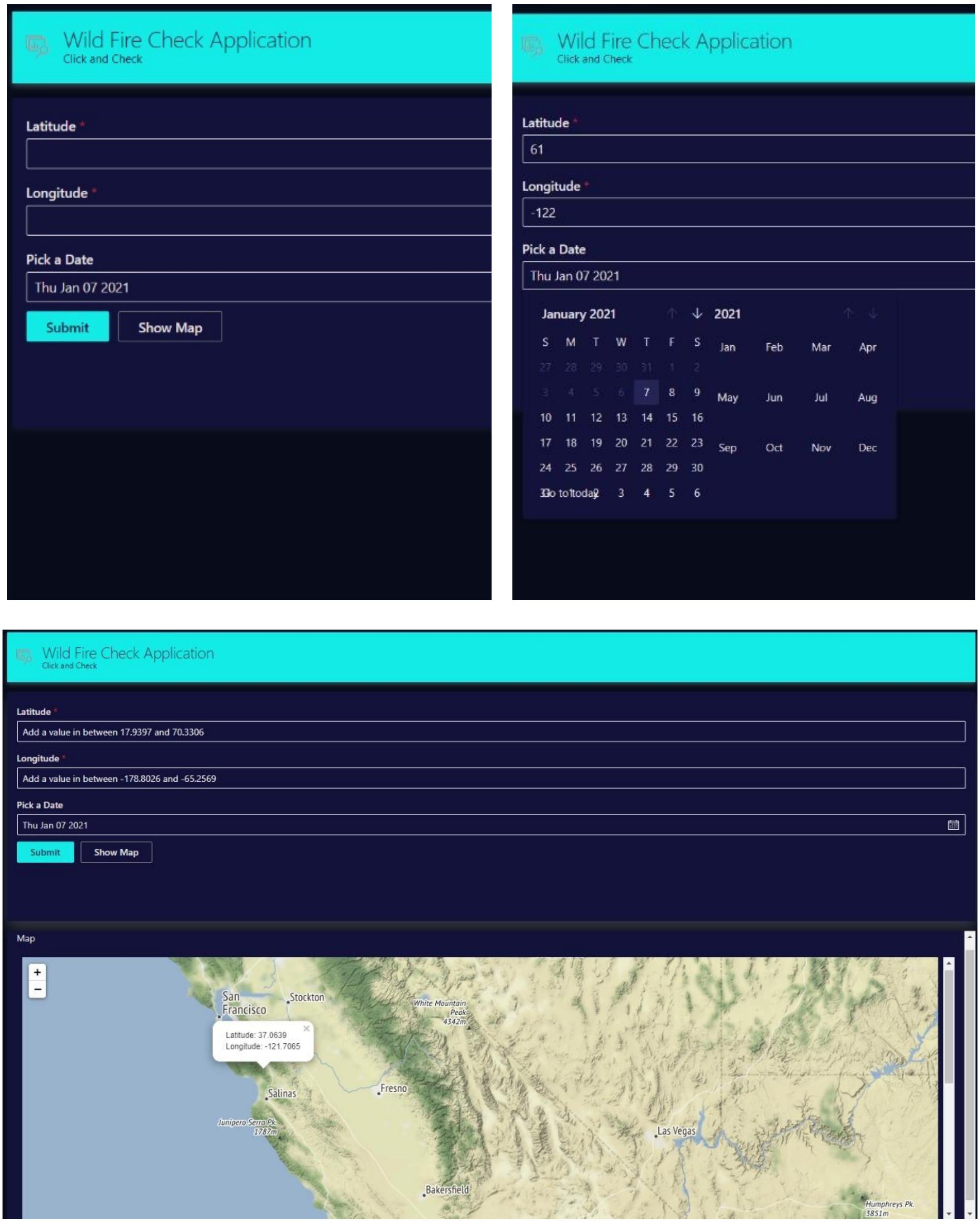

Figure 6 - User Interface of the Application

Prediction from the Application

As the user provides necessary values to the application, model which has been trained in the application fetches the relevant weather data. The application requires weather data for past 7 days from the user picked date. Application is defined to collect all the weather data into one row and then provide them into the model.

The model out is produced as the severity value which starting from zero and zero represents that there is no risk of a wildfire event considering the geo-location and weather data for past 7 days from the user picked date. If there is a severity value presented it can be further understood by referring following table. (Table 2 - Wildfire Size Categorization)

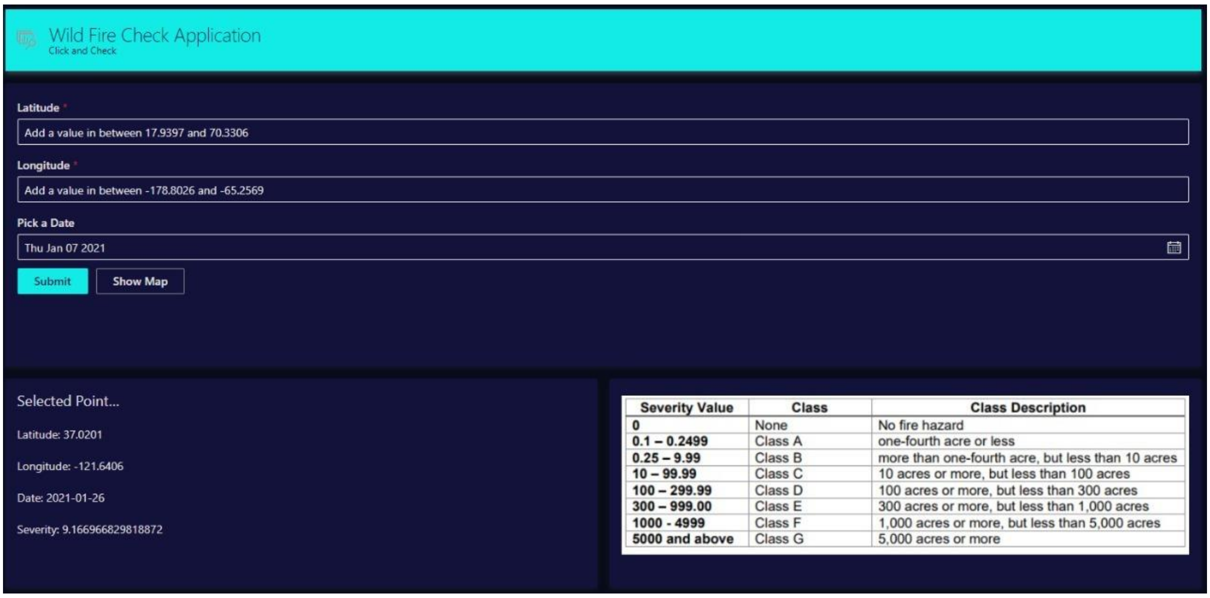

Figure 7 - Output from the Application (Severity)

Note: Initial app launch is taking little longer to load since the pre-saved weather data has to be loaded into the app.

Table 2 - Wildfire Size Categorization

Severity Value	Class	Class Description
0	None	No fire hazard
0.1 – 0.2499	Class A	one-fourth acre or less
0.25 – 9.99	Class B	more than one-fourth acre, but less than 10 acres
10 – 99.99	Class C	10 acres or more, but less than 100 acres
100 – 299.99	Class D	100 acres or more, but less than 300 acres
300 – 999.00	Class E	300 acres or more, but less than 1,000 acres
1000 - 4999	Class F	1,000 acres or more, but less than 5,000 acres
5000 and above	Class G	5,000 acres or more

References

[1] R. Tatman, “1.88 Million US Wildfires,” 13 05 2020. [Online]. Available: https://www.kaggle.com/rtatman/188-million-us-wildfires.

[2] NASA Langley Research Center, [Online]. Available: https://power.larc.nasa.gov/data-access-viewer/.

[3] National Wildfire Coordinating Group, “Size Class of Fire,” [Online]. Available: https://www.nwcg.gov/term/glossary/size-class-of-fire.