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.



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.

Chart

Description automatically generated

Figure - Correlation Matrix

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



**PARAMETER EXPLANATION**

|  |  |  |
| --- | --- | --- |
| Name | Unit | Description |
| T2M\_RANGE\_0 | C - Celsius | Temperature at 2-meter range  (0 = present day) |
| QV2M\_0 | g/kg | Specific Humidity at 2 meters  (0 = Present day) |
| QV2M\_1 | g/kg | Specific Humidity at 2 meters  (1 = One day before) |
| T2M\_RANGE\_1 | C - Celsius | Temperature at 2-meter range  (1 = One day before) |
| QV2M\_2 | g/kg | Specific Humidity at 2 meters  (2 = Two days before) |
| QV2M\_3 | g/kg | Specific Humidity at 2 meters  (3 = Three days before) |
| T2M\_RANGE\_4 | C - Celsius | Temperature at 2-meter range  (4 = Four days before) |
| T2M\_RANGE\_5 | C - Celsius | Temperature at 2-meter range  (5 = Five days before) |
| PRECTOTCORR\_5 | mm/day | Precipitation Corrected (Rainfall)  (5 = Five days before) |
| T2M\_RANGE\_6 | C - Celsius | Temperature at 2-meter range  (6 = Six days before) |

# Weather Forecasting

The Wildfire prevention application is currently designed to predict wildfire based upon selected day’s weather dataset. The weather dataset is obtained from separate website in the application. Currently the application is only limited to selected time range which the predictions can be obtained on. The reason behind limiting the time range is because currently the application is not fitted with the function to fetch weather data for relevant day or date that are in future.

In order to address the limitation of having to fetch weather data every time and to predict future wildfire risks, possibility to design **accurate weather prediction model** has been tested as following.

## Weather forecasting model

Free to use dataset was available on ‘MeteoBlue’ web page [3]. The dataset was based in Basel a city in Switzerland. The data which were acquired were having 6 weather parameters in daily basis and hourly basis. The daily basis weather data were taken for a period of 9 years between 2010 January to 2018 December. Then the hourly basis weather taken only for 4-year period between 2015 January to 2018 December. Weather data which came as hourly basis contained 24 values per day sub-dataset for representing each hour in a day, but the daily weather data set only had the mean values regarding the parameters which were requested when downloading from web page

The RNN model has been fed only with one input at a time, but it has modelled to predict several type of forecasts such as precipitation, temperature and wind speed. The model has been fed with two set of datasets, where one set was daily record based and other with hourly record based. The model, which is built on daily data, is capable of predicting 1 day ahead of weather forecast considering past data which belongs to 3 days. The hourly based model evaluates 24 hours of past data to predict 1 hour ahead weather forecast.

The model has been built with Long Short-Term Memory (LSTM) cells included in every dense layer. Input layer consists of 120 neurons and the output layer only with 1 unit. The dense layer set with 3 layers, and 4 drop out layers were used to model both RNN models for daily and hourly dataset. Dropout layer in a RNN structure means that, if the threshold value of dropout is passed, the relevant neuron will be deactivated. Combined with this model it allows neurons, not to depend on other neurons out of order weights. Following table represent the parameters which have been selected to build the model along with training and testing dataset selection criteria.

Table - RNN Model Specification

|  |  |  |  |
| --- | --- | --- | --- |
| Layer Rank | Layer Name | Layer Size (Neurons) | Dropout Rate |
| 1 | Input Layer | 120 | - |
| 2 | Dropout Layer 1 | - | 0.2 |
| 3 | Dense Layer 1 | 120 | - |
| 4 | Dropout Layer 2 | - | 0.2 |
| 5 | Dense Layer 2 | 120 | - |
| 6 | Dropout Layer 3 | - | 0.2 |
| 7 | Dense Layer 3 | 120 | - |
| 8 | Dropout Layer 4 | - | 0.2 |
| 9 | Output Layer | 1 | - |
| Optimizer = 'Adam'; loss = 'Mean Squared Error' | | | |
| Epochs = 25; Batch size = 500 (hourly set), 100 (daily set) | | | |

Following table represents the total number of samples used to train both hourly and daily weather forecasting models along with the time-period it was taken.

Table - Dataset sizes used on RNN Model

|  |  |  |  |
| --- | --- | --- | --- |
| RNN Model | Data Type | Time Period of Dataset | Total Samples |
| Hourly based model | Training Dataset | 2017 January to December | 8736 |
| Testing Dataset | 2018 January to December | 8736 |
| Daily based model | Training Dataset | 2010 January to 2017 December | 2920 |
| Testing Dataset | 2018 January to December | 365 |

## Model Results & Evaluation

Following figures represents the prediction results and actual data for model which predicts weather one day ahead. Parameters which can be organized in a time series array has only been trained with the model.

A picture containing graphical user interface

Description automatically generated

Figure - Daily Precipitation Forecast using RNN

Chart, line chart

Description automatically generated

Figure - Daily Temperature Forecast using RNN

Graphical user interface, chart

Description automatically generated

Figure - Daily Wind Speed Forecast using RNN

Following figures represents the prediction results and actual data for model which predicts relevant weather parameter one hour ahead.

Chart, histogram

Description automatically generated

Figure - Hourly Precipitation Forecasting RNN Model

Graphical user interface, chart, line chart

Description automatically generated

Figure - Hourly Temperature Forecasting RNN Model Comparison

Chart, histogram

Description automatically generated

Figure - Hourly Wind Speed Forecasting using RNN [Scaled]

The Recurrent Neural Network model with LSTM cells has performed well throughout the testing phase. It has been shown that a well-trained RNN model can provide accurate results on weather forecasting. The RNN model does only consider the present and past data set to predict the behavior of the weather in future time. When comparing both daily based and hourly based models, it could be seen that the hourly predicting model, in other words short-term predicting model perform well.

## Final Thoughts on Weather Forecasting Model

However above results clearly shows that precipitation, temperature & windspeed which are in a time series form can be predicted using above model with a quite good accuracy. The issue arises when the Wildfire application requires more weather parameters which cannot be predicted with RNN model as above. **Specific humidity (g/kg)** and **precipitation (mm/day)** parameters are holding dependency with more advance metrological parameters such as cloud coverage, air pressure and require more advanced neural networks for prediction. Thus, designing a separate machine learning model dedicated to predicting weather data instead of properly utilizing a third party service for fetching relevant weather data, could lead to less accurate result in wildfire prediction application.

# 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%.

A screenshot of a computer

Description automatically generated with medium confidence

Figure - 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.

Text

Description automatically generated

Figure - 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”.

Table

Description automatically generated

Figure - 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.

Text

Description automatically generated

## 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.**

## Graphical user interface, text Description automatically generated

Figure - 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 - 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 |

A picture containing graphical user interface

Description automatically generated

Graphical user interface, application, website

Description automatically generated

Figure - 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)

A picture containing graphical user interface

Description automatically generated

Figure - Output from the Application (Severity)

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

Table - 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. |