Project Requirements Document v2

Project: Wildfire Prediction

Team Name: Flare

Authors:

- Kelly Lin (Team Lead)
- Alvin Tu (Scribe)
- Nick Ong
- Shuya Shou
- Steven Chang

Sponser: PwC

Introduction

Background

Wildfires are a serious environmental concern that cause not only economic and ecological harm, but also put human lives in danger. With growing climate change, wildfires are an increasingly dangerous problem, which makes it even more imperative to be able to predict and prepare for them. Various conditions increase the likelihood of wildfires, such as the presence of fuel or residing in a dry climate. By using these variables, we can generate predictive models that estimate the long term effects of climate change on wildfire occurrences, ultimately reducing the damage they have on the environment and society.

Problem Being Solved

Currently, many areas are in danger from wildfires, with few means to detect and prevent them in the long term. As climate change worsens, we need to prepare for the increased occurrences of wildfires since it will not only help reduce business risks caused from wildfires, but also potentially save lives and protect the environment. Our team will tackle this problem by using deep learning approaches to predict potential wildfires.

Why is This Important?

Mitigating the risk and damages that occur from wildfires is urgent. Over the past ten years, the U.S. has averaged an annual 7.4 million acres burned, costing the country $2.4 billion a year. Additionally, the world's largest corporations face up to $1 trillion in damages from climate change. However, the most impacted group are individuals who have lost their homes, savings, and lives from fires. Investing in methodologies to predict wildfires would minimize the toll of these catastrophic events.

Existing Solutions

Studies have shown that the main causes of wildfires include climate change, human negligence, and natural causes like thunderstorms. Additionally, meteorological measurements such as temperature, relative humidity, rain, and wind are all known to impact forest fires. Most of the current solutions rely on a simulation method and mathematical models, which can be expensive and hard to maintain. Other common methods include using Support Vector Machines, Decision Trees, Random Forests, and genetic algorithms. However, as an increasing amount of meteorological and climate data becomes available, it becomes difficult for these methods to remain effective, thus encouraging deep learning approaches. Recurrent Neural Networks (RNN) have been used in the past to do time-series forecasting, which uses past observed values to make future predictions. However, RNNs tend to suffer from long term dependency problems. On the other hand, Long Short Term Memory (LSTM) networks are a type of RNN that reduces this issue, which is a more suitable choice for our task.

Some current solutions use the Global Wildfire Information System (GWIS) data. The GWIS data originates in Europe, but can be used to predict wildfires on a global scale. The caveat to the existing models is that they can only predict for a few weeks ahead of the current date. Our objective is to build a model that can forecast months to years in the future.

After speaking with climate scientists at the University of California, Santa Barbara (UCSB), we learned that there are climate simulations that can be used to help identify

the way the environment changes. Using these climate models, scientists at UCSB have predicted how wildfires can occur due to simulated changes in environments. Using these simulations guarantees clean, usable data that comes from an environment that is under the complete control of the user. However, with the usage of simulations, there is the chance that the results deviate from reality.

By analyzing these existing solutions, we can determine how our system should be designed to handle long-term wildfire prediction. Both the advantages and downsides of the existing solutions will be taken into account for our system's development. Using these insights, our system will be developed to be robust, while also maintaining simplicity.

Project Goals

The goal of this project is to create a LSTM neural network to predict estimated wildfire sizes using time series climate data combined with weather variables. A wildfire prediction model will be developed for Australia's states, as it was the only available open source data that fit our needs. In the future, the model can be improved to account for different locations given the availability of geographically-diverse datasets. The model will be able to forecast on a day-by-day basis, with no limit to the number of forecast days. It should be noted that the predictions may become increasingly inaccurate over time due to the autoregressive network architecture, which means that the time series predictions are generated using predictions from the previous time stamps. Finally, the model will be improved to ensure robustness against outliers through data that predicts variances in the features we use.

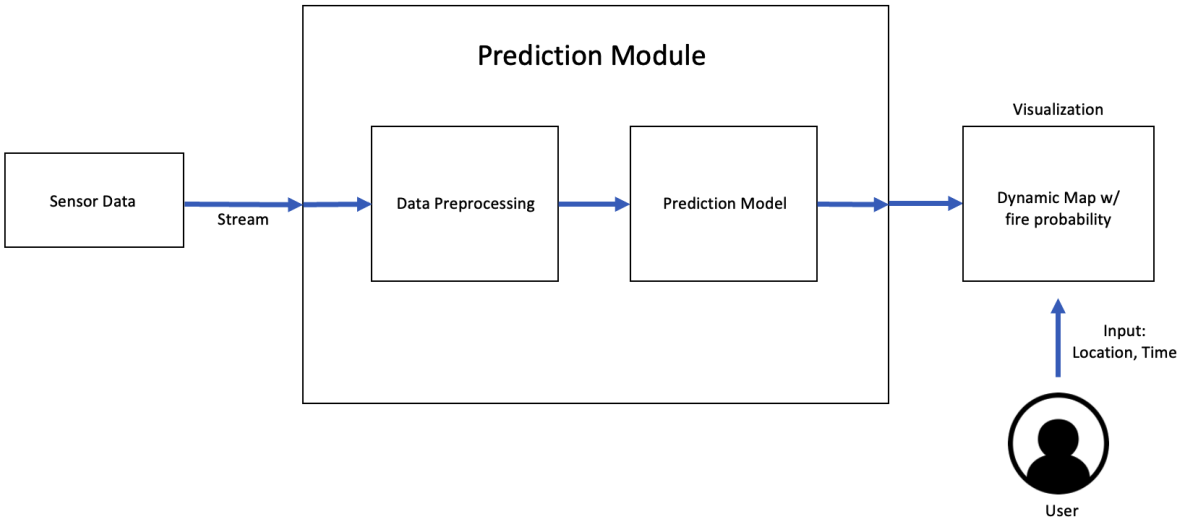
In addition to the LSTM, we want to build a graphical user interface (GUI) dashboard where our user can easily use our model to make predictions. This GUI will include a form to input regions and time ranges, which the system will then use to output forecasts for the provided location based on the given input. We would also like for this GUI dashboard to have an easy way for PwC to upload their own models and datasets.

Assumptions

Our system will be built under several assumptions. The most significant assumption is that climate change is one of the most important causes of wildfires, which is why the model will be trained with climate data. Additionally, because we are using a LSTM model for time-series forecasting, we are assuming that there is a long term trend in the climate data that directly correlates with wildfire occurrences. Using an LSTM also has an assumption that forecasts are more accurate towards the beginning, and as the forecast gets further from the start date the less accurate it becomes. We also assume that wildfire area in the Australian regions is an approximate estimate to the day-to-day wildfire occurrences.

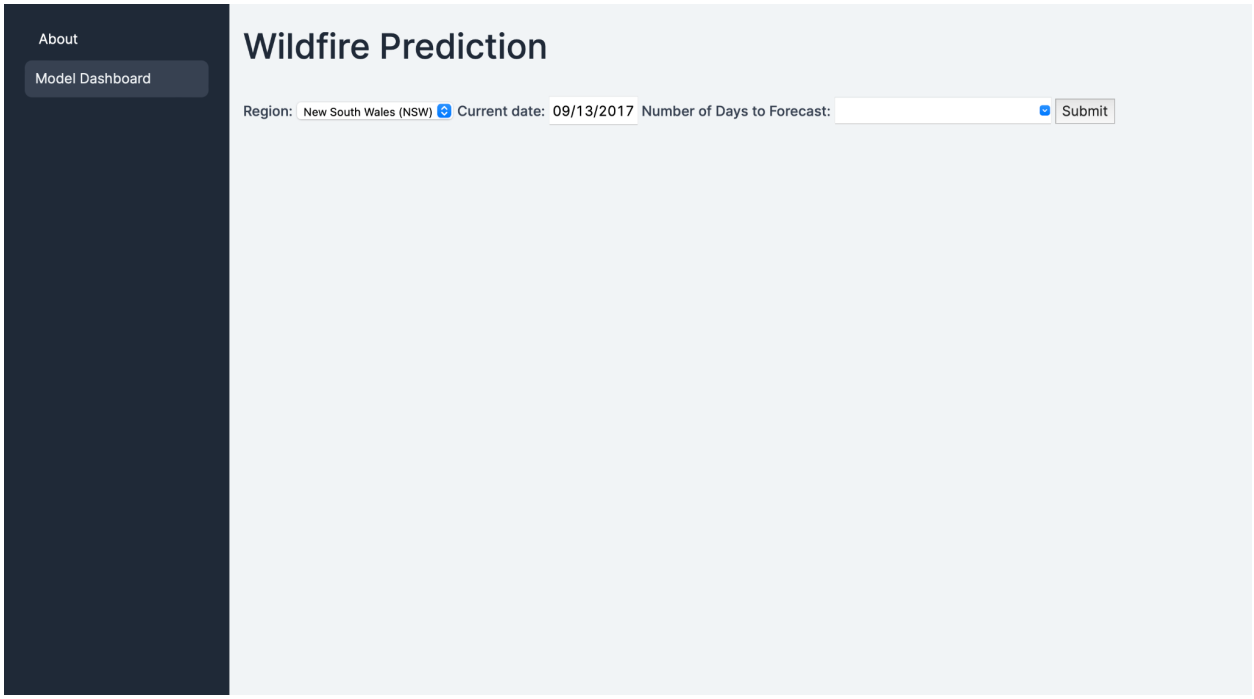
Overview of System Architecture & User Interaction

The diagram below shows an overview of our system. On the left side, we have live sensor data regularly streaming into our prediction module. Our prediction module contains two parts – a data preprocessing module and a prediction model. The data preprocessing module takes the incoming sensor data and processes them so that they can be used by the prediction model to make predictions. The prediction module is a network that is pre-trained using 10+ years of climate data for time-series forecasting. Given a specific location and time, the prediction module can then output the probability of wildfires in the area for a specified amount of time in the future. Users can specify a location and time to visualize the model's output onto a dynamic map.

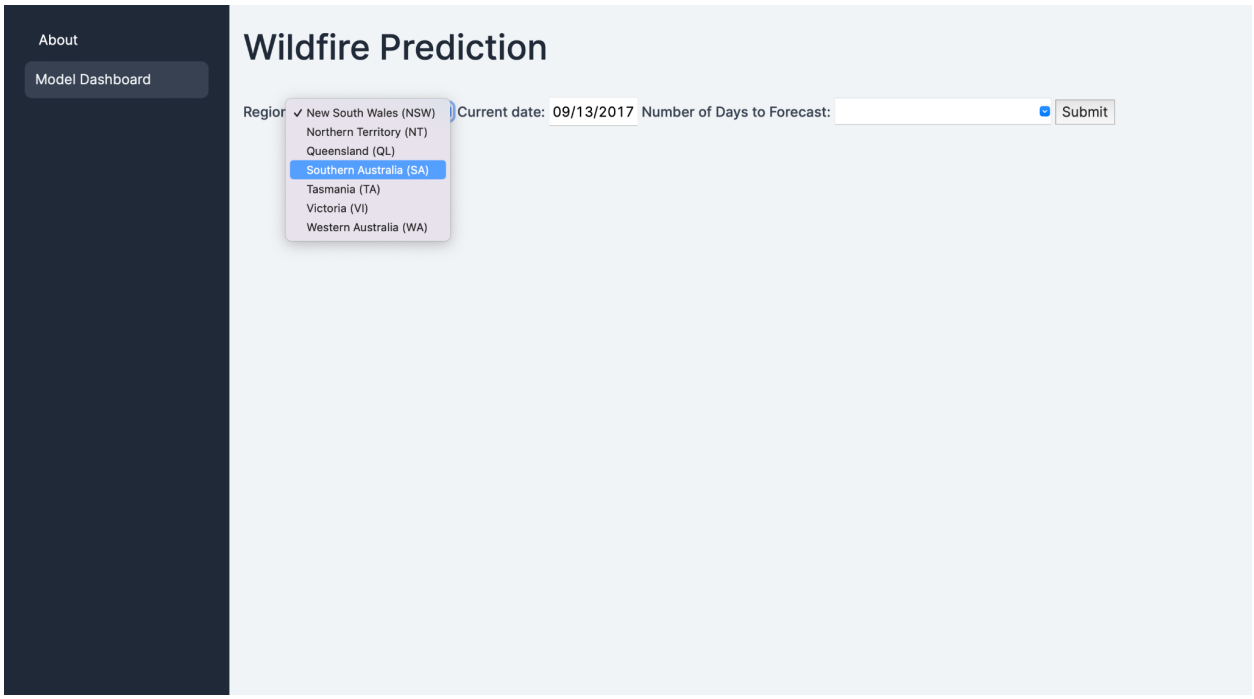


Front End UI/UX

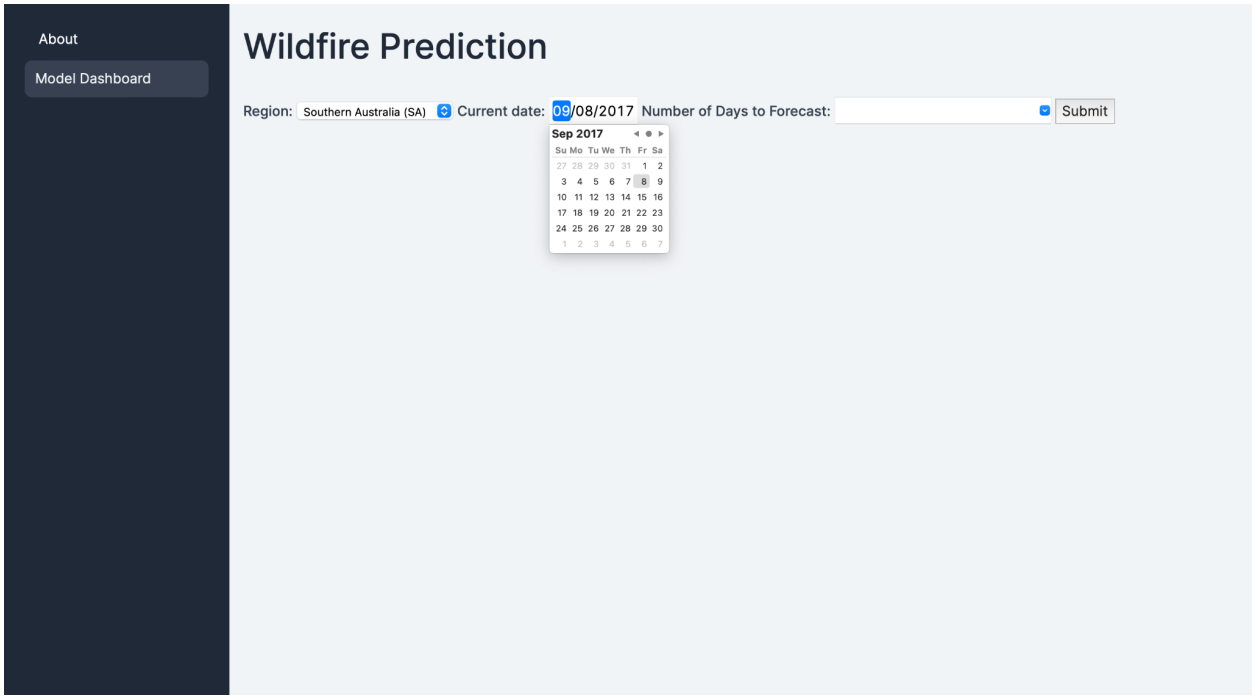
The interface has three inputs that are used to determine where and when the system makes predictions for.



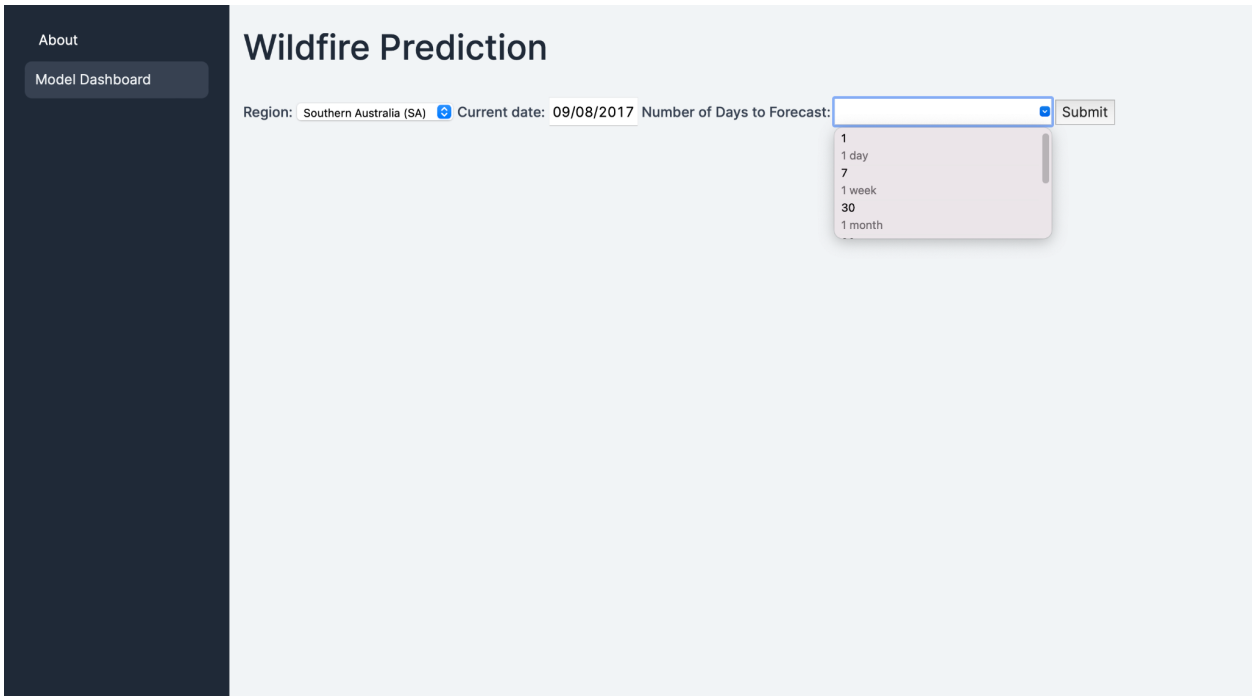
The region dropdown shows the available locations that the system can make predictions for. From the list, the states of Australia can be chosen.



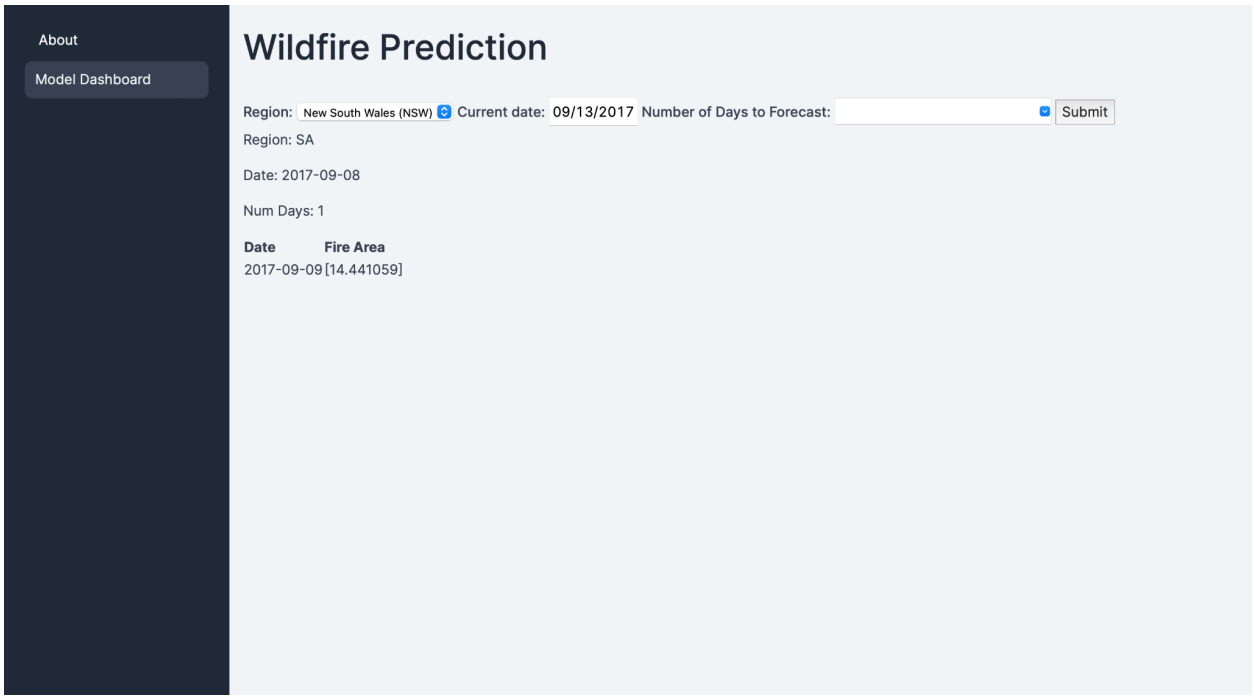
The calendar dropdown is used to select the present day. The system will make predictions for time periods after this selected day.



The number of days dropdown is used to select the number of days for the system to forecast after the selected date. The available forecast amounts are a day, a week, a month, three months, six months, and a year.



Upon form submission, the system outputs the dates in the future and the associated predicted fire areas of the selected region. The amount of dates that are shown depends on the chosen number of days to forecast. The system also shows the inputs given by the user.



Requirements

Note: We are filling in user stories with tasks because a machine learning model does not have many user stories

---

Research Tasks

1. Research past work on wildfire prediction using machine learning
    a. **Acceptance criteria:** Writing up detailed notes about past wildfire prediction research papers and how we can apply this knowledge.
    b. Github issue
2. Research open source climate/wildfire datasets
    a. **Acceptance criteria:** Writing up detailed notes about datasets with pros and cons of each in order to pick one or mix multiple.
    b. Github issue
3. Research climate specific domain knowledge
    a. **Acceptance criteria:** Writing up detailed notes on climate specific variables from research papers that affect wildfires to better understand how to analyze data

    b. Github issue
4. Research Pytorch and machine learning tools
    a. **Acceptance criteria:** Writing up detailed notes on tools and libraries that could aid in a wildfire prediction model and teaching the other members how to use them.
    b. Github issue

Model Building

5. Data analysis and feature selection
    a. **Acceptance criteria:** Thoroughly analyze datasets and select most relevant and impactful features to use to train models.
    b. Scenario 1: Select data from only one dataset
        i. Given dataset is thorough
        ii. We will select and analyze data from what we deem as the most relevant dataset
    c. Scenario 2: Select data from multiple datasets
        i. Given multiple datasets have useful data
        ii. We will pick and choose data from multiple datasets to combine into one dataframe for data analysis
    d. Github issue
6. Build LSTM model
    a. **Acceptance criteria:** Build a basic LSTM machine learning model using the analyzed dataset mentioned in user story 5 with training and validation samples.
    b. Github issue
7. Build model pipeline
    a. **Acceptance criteria:** Create a pipeline to streamline model training and hyperparameter tuning to easily create multiple models
    b. Github issue
8. Build testing pipeline
    a. **Acceptance criteria:** Create a pipeline to streamline testing
    b. Github issue
9. Individual model: Alvin

    a. **Acceptance criteria:** Model that achieves high forecasting accuracy and will eventually be integrated with Kelly and Shuya's model
    b. Github issue
10. Individual model: Kelly
    a. **Acceptance criteria:** Model that achieves high forecasting accuracy and will eventually be integrated with Alvin and Shuya's model
    b. Github issue
11. Individual model: Shuya
    a. **Acceptance criteria:** Model that achieves high forecasting accuracy and will eventually be integrated with Kelly and Alvin's model
    b. Github issue
12. Unit test
    a. **Acceptance criteria:** Unit tests for frontend and backend components that successfully pass and are thorough
    b. Github issue
    c. Github commit (backend)
    d. Github commits (frontend)
13. Integration test:
    a. **Acceptance criteria:** Frontend and backend successfully integrate and opens up a web UI that can make forecasts based on saved models
    b. Github issue
    c. Github commit

Using Model

14. As a PwC climate scientist, I can input data that matches the required features we have chosen
    a. **Acceptance criteria:** Successfully have method to input data
    b. Scenario 1: User inputs correct data
        i. Given the model is currently working
        ii. Given there is a method to input data
        iii. The user will be given long-term wildfire predictions for given data
    c. Scenario 2: User inputs incorrect data
        i. Given that model is currently working

ii. Given model has been trained on specific data
iii. The user will be given an error that the data is not correct

Visualizing Data

15. As a PwC climate scientist, I can view areas of wildfire risks on a dynamic map
    a. **Acceptance criteria:** Map successfully outputs areas of wildfire risks in an easy to read format
    b. Scenario 1: User opens up map
        i. Given model is working correctly
        ii. Given model has made up-to-date predictions
        iii. The user will be able to open the map and explore wildfire probabilities in different locations
    c. Scenario 2: User zooms in on map
        i. Given map is up-to-date
        ii. The user will be able to see more specific cities and locations
    d. Scenario 3: User clicks on specific area
        i. Given map is up-to-date
        ii. Given data is open source
        iii. The user will be able to see specific details as to why a probability for the clicked location was given

GUI

16. As a user, I can click on a drop down to select a region in Australia to forecast for
    a. **Acceptance criteria:** Successfully drop down a menu of seven Australian regions, all of which are selectable
    b. Scenario 1: User selects one of seven regions
        i. Given only seven regions to select from
        ii. Given user cannot type in any text
        iii. Region will be selected and shown on the drop down menu bar with the drop down menu minimized
    c. Scenario 2: User selects a new region after a region is already selected
        i. Given region has been selected

    ii. The drop down menu will show again, and the user will be able to select a new region, updating the current region and minimizing the drop down menu
  d. Scenario 3: User selects the same region after region is already selected
    i. Given region has been selected
    ii. The drop down menu will show again, and the user will be able to select a region. The current region will stay, and the drop down menu will be minimized
  e. Github issue
  f. Github commit
17. As a user, I can select a date from the calendar for forecasting
  a. **Acceptance criteria:** Calendar successfully displays and allows user to select dates available from dataset
  b. Scenario 1: User selects a date
    i. Given date is in dataset
    ii. Calendar display will show, and the user can select a date from the display. The date will then be displayed while the calendar display is minimized.
  c. Scenario 2: User selects a new date
    i. Given date is in dataset
    ii. Given date is already selected
    iii. Calendar display will show, and the user can select a new date from the display. The new date will replace the old date displayed while the calendar display is minimized.
  d. Scenario 3: User selects same date
    i. Given date is in dataset
    ii. Given date is already selected
    iii. Calendar display will show, and the user attempts to select the same date from the display. The date displayed remains unchanged, and the calendar display minimizes.
  e. Github issue
  f. Github commit

18. As a user, I can input the number of days after the selected date to forecast a wildfire
    a. **Acceptance criteria:** Text field is able to take in any integer
    b. Scenario 1: User input is an integer
        i. The form can be submitted successfully given that the other fields are filled out
    c. Scenario 2: User selects a pre-existing option from the dropdown provided
        i. The text field is filled with the selection
    d. Scenario 3: User input is not an integer
        i. Web page displays an error message indicating that the text field must be an integer
    e. Github issue
    f. Github commit
19. As a user, I can view the forecasts in a list after submitting the necessary information
    a. **Acceptance criteria:** Forecasts display with date and estimated fire area in list
    b. Scenario 1: All forecasts display properly
        i. Given all inputs are valid
        ii. The user will be given data with estimated fire area in kilometers squared with corresponding dates
    c. Scenario 2: Forecasts fails
        i. Given inputs are not valid
        ii. The user will be given data with “nan” values as inputs are not valid
    d. Github issue
    e. Github commit

Loading Model to Frontend

20. As a PwC scientist, I can load different models that the frontend interacts with
    a. **Acceptance criteria:** Different LSTM models can be used and successfully generate predictions based on the user input
    b. Scenario 1: Web page loads predictions using the model
        i. Given all inputs are valid

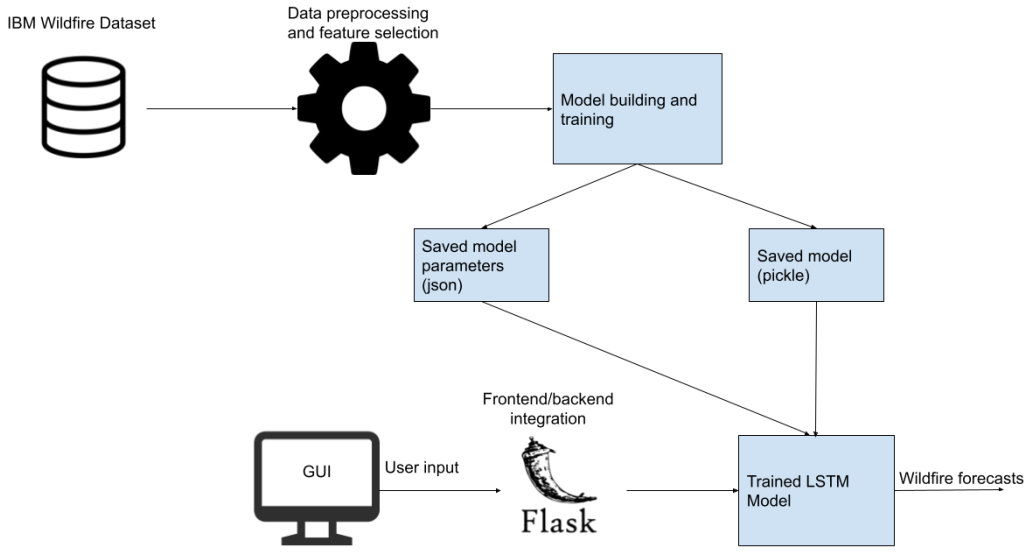
ii. The backend code for the web page is generalized enough to load in different models with their parameters

c. Scenario 2: Web page uses default model to generate predictions
   i. Given all inputs are valid
   ii. The model or parameters aren't suitable to be used with the frontend and gracefully alerts the user.
   iii. The backend defaults to a predefined model that is guaranteed to work.

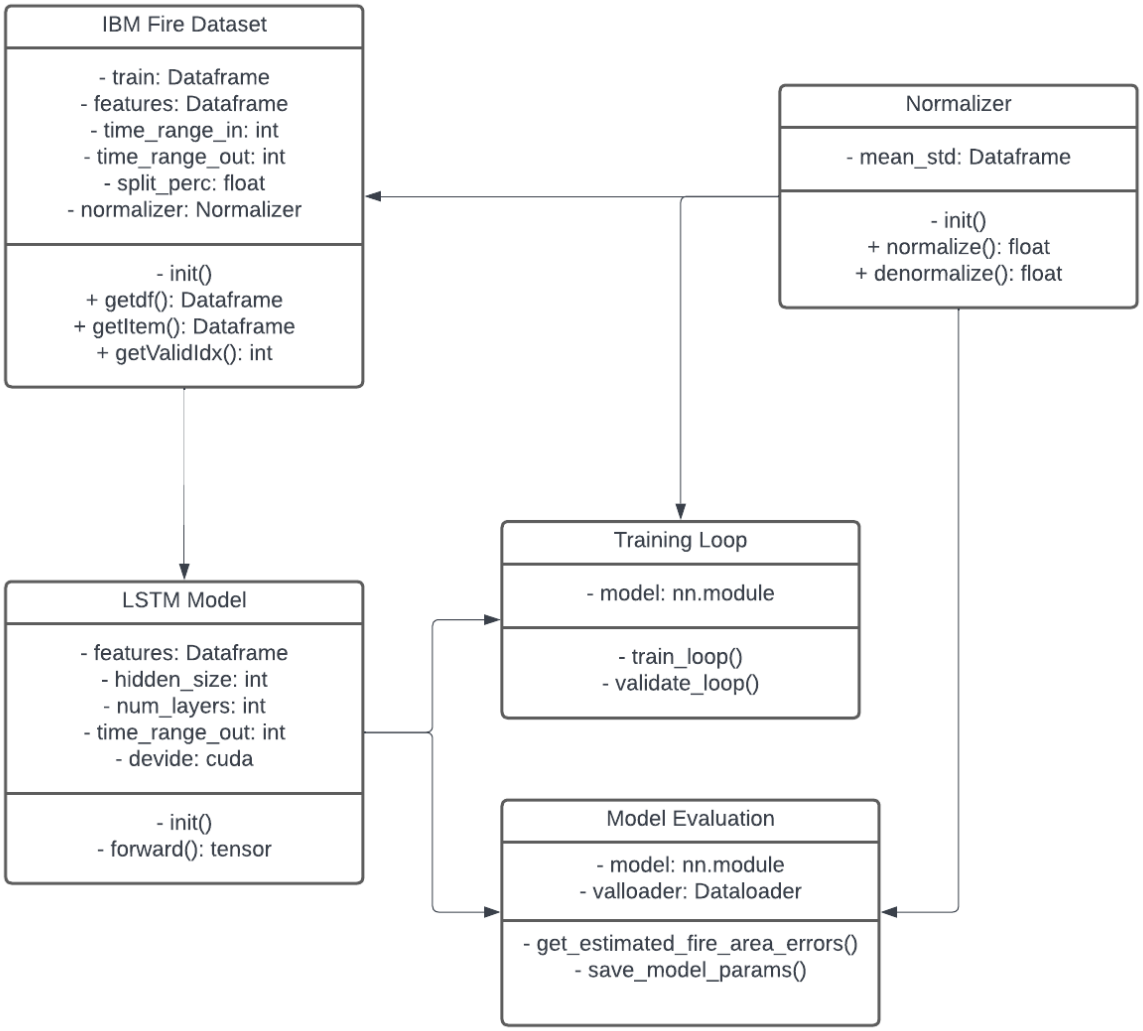
d. Github issue

System Models

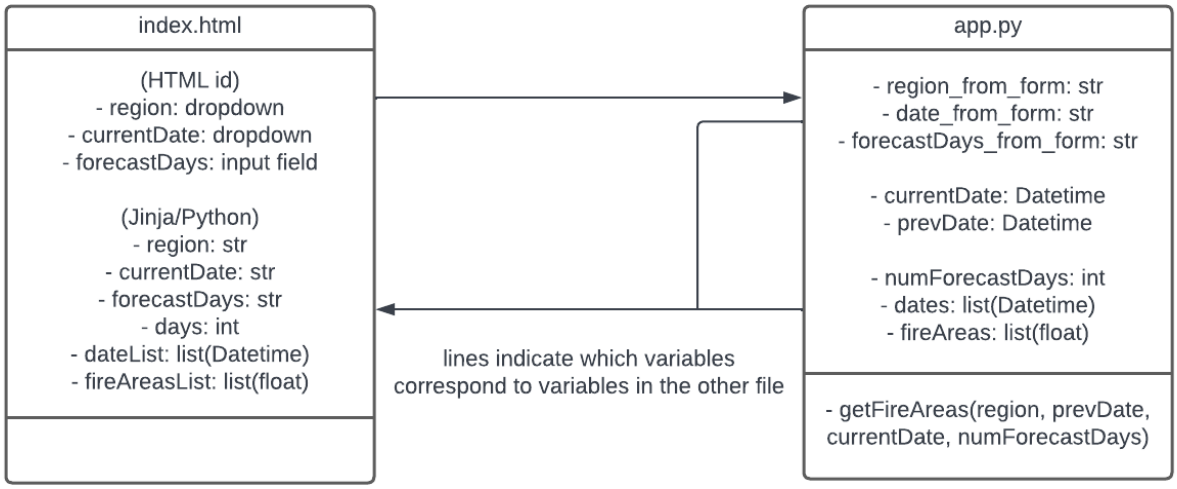
High Level Diagram



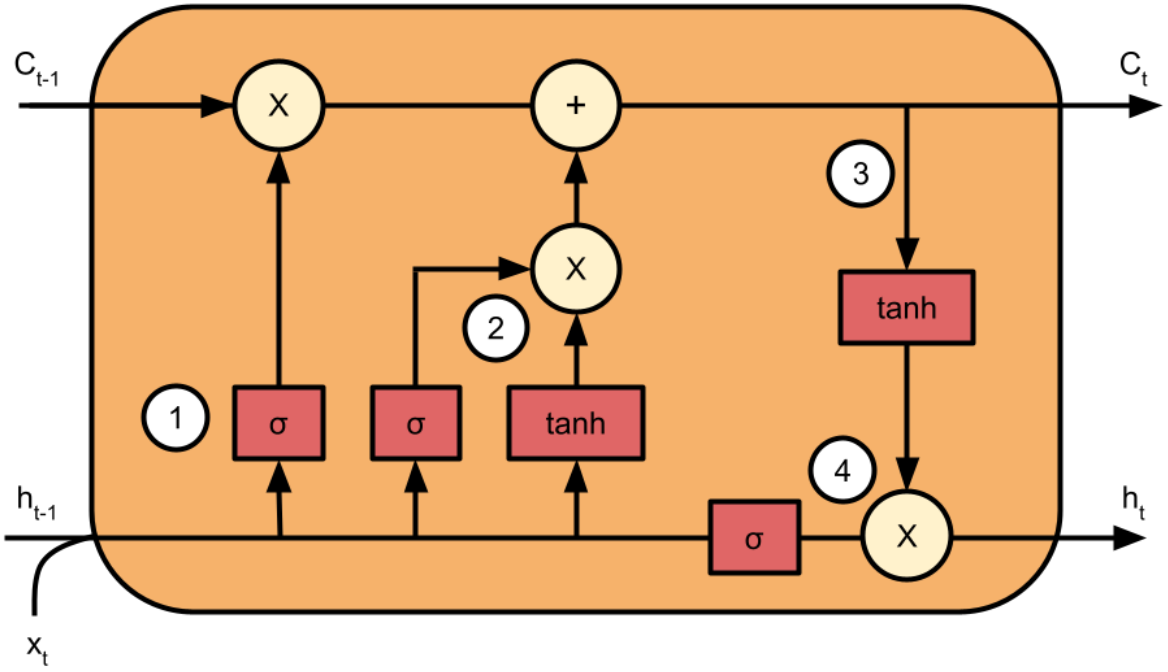
Backend UML Diagram



Frontend UML Diagram



LSTM Network Architecture



LSTM networks are a type of recurrent neural network that are organized into cells. C_{t-1} and C_t represent the input and output of the cell, with C_{t-1} being the output of the previous cell. Information relevancy comes in through h_{t-1} and x_t, in which the sigmoid (σ) and tanh functions determine their importance to the cell's state. To put it broadly, the top horizontal line represents the information stored, and the bottom line determines how that information is transformed within the cell and beyond.

1) This sigmoid function is the forget layer where the output is between 0 and 1, with 0 meaning none of the information is remembered and 1 indicating that all of it should persist. This output is multiplied by C_{t-1} and determines how much of the previous information is remembered.

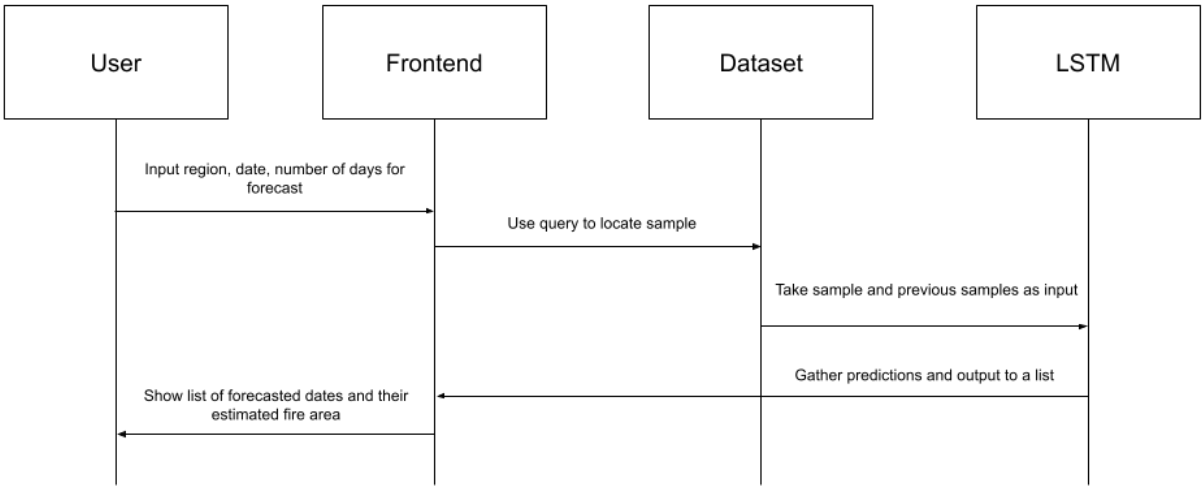
2) This sigmoid function is known as the input gate layer and determines which values will be updated. The tanh function next to it creates new candidate values. The product of these functions represent the new information that is added on top of the previous information, if persisted.

3) Together, the addition of the historical and new information make up the output C_t and are transferred through the right-most tanh function. This tanh function decides the magnitude of the current information and how it will affect the next cell.

4) The sigmoid function determines what part of the cell state will be outputted. It is then multiplied by output of the tanh function to become the h_t, one of the inputs of the next cell state.

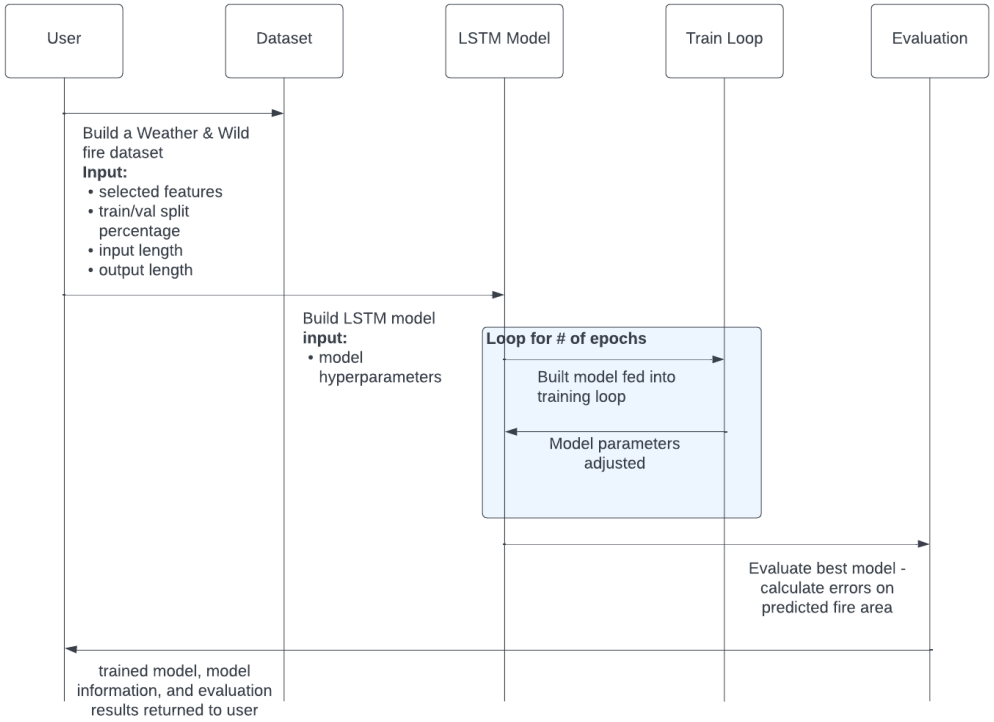
Sequence Diagrams

Frontend:

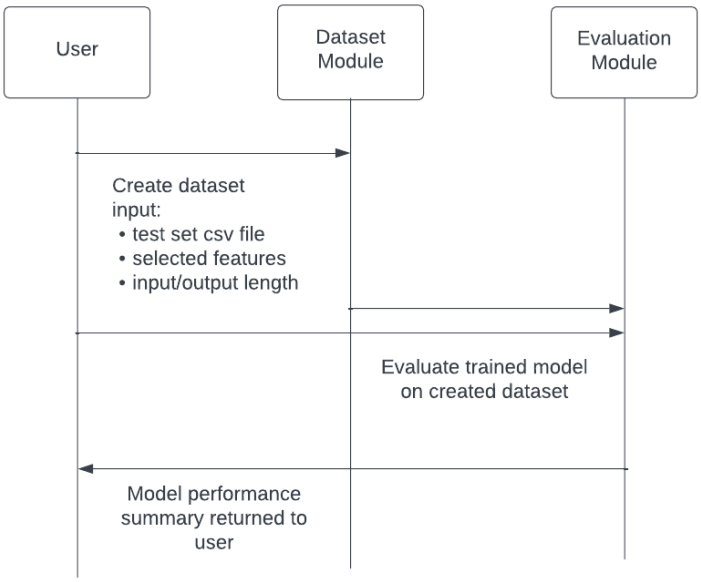


Backend:

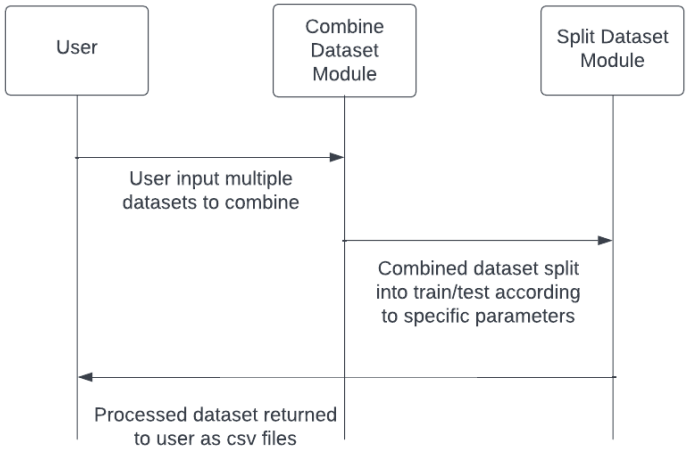
User: ML engineer training from scratch



User: ML engineer evaluating existing models



User: Data scientist inspecting and processing datasets



Appendices

Technologies

- Python
    - Pytorch
    - Pandas
    - Flask
- Google APIs (required for certain datasets)
- Github
- Heroku

References

- Understanding LSTM Networks -- colah's blog
- IMB dataset