WILDFIRE PREDICTION APP FOR THE UNITED STATES

Honghu Lin and Kun Ye and James Leathy and Mauricio de Paula[1]

Abstract— **Our project is inspired by a wildfire data set from Kaggle with data size of approximately 1.88 million large [1]. The objective of this project is that we intent to create an application to predict locations where wildfire will possibly happen. Given features such as temperature or latitude, for a certain range of area, the software will calculate data in a remote server using different data mining algorithms, and be able to output a labeled map that tells the user what will happen in the future.**

I. INTRODUCTION

Nowadays, global warming has become a factor that mainly causes fluctuating temperatures, various humidities and more. A recent report about California, claimed that the wildfire disaster on October 11th resulted in a massive economic loss of around 3.4 billion dollars [2]. Not only in the United States, residents in B.C this summer had both witnessed and suffered polluted air because of a serious wildfire that took place close to the province.

It is not true to say that nobody cares about wildfire, yet in most cases, some authorities had been investigating different causes and analyzing the system for decades. Sensors, cameras, unmanned aerial vehicles are distributed everywhere to monitor areas. Although there are tons of devices out in the forests, monitoring wild conditions is still inaccurate, hard to generalize and very expensive.

In consequence of huge loss in precious natural resources and significant financial drawbacks, it really brings out our attention to put more efforts in designing a software that can help the Environment Bureau to figure out possible future wildfires that could arise. We wish to solve wildfire problems by analyzing the large data we obtained using data mining algorithms.

Figure1. This is the user interface we roughly designed for our first deliverable. It will get improved and refined as project proceeds. We wish to add more features onto this app later and make it more user friendly.

II. RELATED WORKS

There has been some research regarding the use of data to predict wildfires. Two researchers have developed a spatial-temporal forecasting framework to aid in the prevention of wildfires given location [4]. Another group of researchers, used techniques such as boosted regression trees and generalized additive models, and data mined from a town in Iran [5]. This project relates to those because it will attempt to predict probable areas of future wild fires based on past data obtained from the United States.

*This work was not supported by any organization

[1]Instructor: George is with the Faculty of Computer Science, University of Victoria, Victoria, B.C.,Canada

III. PROJECT DESCRIPTION

The final deliverable for the project will be a mobile application that allows users to interact with wildfire data. This includes an interface that will allow users to view historical data and request our algorithms to make predictions about events related to wildfires.

A. Prediction and Categories

The first thing we would like to do with the data is that given the attributes listing on the dataset that we downloaded from kaggle. We want to make the following predictions.

- The causes of fire
- The number of fire
- Wildfire or no fire

The ranks of fire are identified as following:

- Smouldering ground fire
- Low vigour surface fire
- Moderately vigorous surface fire
- Highly vigorous surface fire with torching, or passive crown fire
- Extremely vigorous surface fire or active crown fire
- A blow up or conflagration; extreme and aggressive fire behavior.

The causes of fire are identified as following:

- Lightning
- Power line
- Missing/Undefined
- Equipment Use
- Structure
- Campfire
- Miscellaneous
- Arson
- Fireworks
- Smoking
- Debris Burning
- Children

Fig. 1. Prediction for cause.

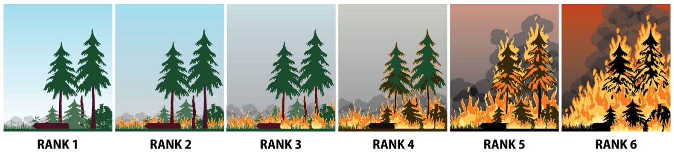

Fig. 2. Prediction for fire ranks.

B. Dataset

The dataset is downloaded from kaggle and the size of the fire is very massive (around 1.88 million entries). So when we use these dataset for testing, for some cases we will piece the data randomly with respect to different attributes.

The original file format is in SQLITE form, but for this project we are going to use .csv file for the convenience of implementation. The way we converted SQLITE to .csv is by accessing the availability of PANDA.

C. Mobile Application

The mobile application will provide to the user an interface to interact with our data mining algorithms and an interface to view statistical information on the history of wildfires. The mobile application will communicate over HTTP(S) to an application that is running in Heroku.

D. Data Mining Algorithms

Data mining algorithms will be used to predict events related to wildfires and to extract information from our data set of 1.8 million records. Access to these algorithms and information will be provided to a user through the mobile application.

We import the data from database to our program. Just for the simple testing of our algorithm, We use 10 percent of the dataset as training data and the rest of 90 percent for testing. For post-midterm analyzing, we are planning to increase the size of training data for a better prediction accuracy. At first, we choose the Feature to be `STAT_CAUSE_DESCR` (fire cause) in order to get the causes of fire and how fire distributes in ranges. Then We build a graph as is shown in figure 2.

This graph provides us a clear insight of what are the factors which causes the fire. Also, It helps us to make a decision about whether there would be a fire or not. We also used decision tree algorithm to do the prediction. The accuracy of the trail set which We obtain is approximately 0.30061. On the other hand, We perform a K-cross validation of fold 2 with accuracy 0.31652, and a fold of 3 with accuracy around 0.32611, a fold of 4 with accuracy equal to 0.33712, a fold of 6 equal to 0.35611. We also tried different training sets to see how accuracy behaves accordingly. In the future, we will attempts various models to do more comparisons in order to get the most reasonable and adequate result.

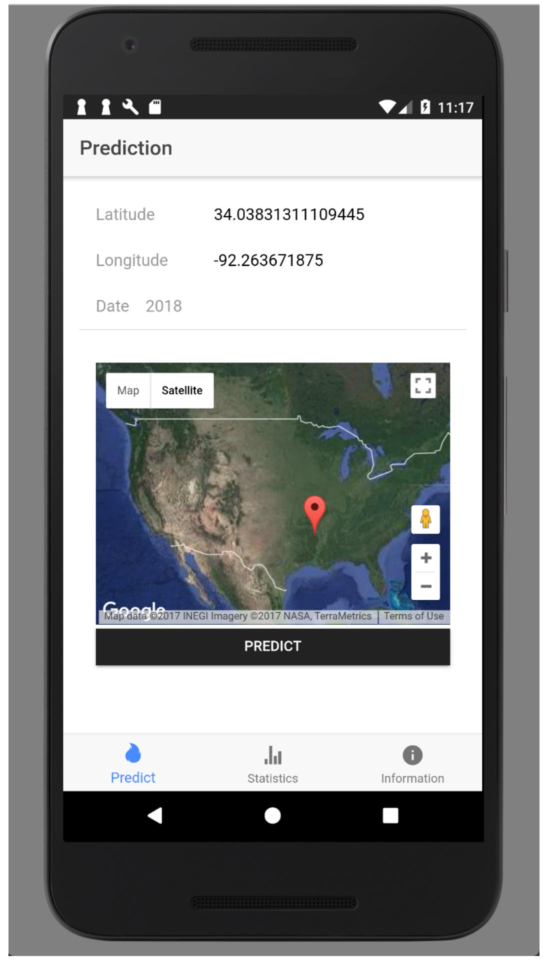

Fig. 3. App User Interface

1) Decision Tree Classifier: We tried decision tree classifier mainly because we notice that our dataset is very large and comprehensive. Also, decision tree classifier makes really fast prediction. Besides that the property of highly non-linear relationship are very friendly to our dataset.

For the fire ranks prediction, we get approximately fifty-seven percent in average when we ran a truncated data around 300,000. Meanwhile, for the same dataset we get the accuracy around sixty-one percent.

2) Naive Bayes Classifier: Since NB classifier need less training data, so When we deal with small test data, we would like to try with NB classifier, because if the NB conditional independence assumption holds, then it will converge quicker than discriminative models like logistic regression. Even if the NB assumption doesnt hold, it works great in practice. On the other hand, the data is highly scalable. It scales linearly with the number of predictors and data points. When we do the testing it handles both cases ”continuous” when we use all 1.88 million dataset and ”discrete” when we use partial dataset. The most significant point is that there are a lot of off-topic attributes in the dataset, but NB clssifier is not sensitive to irrelevant features,

For the fire ranks prediction, we get approximately sixty-

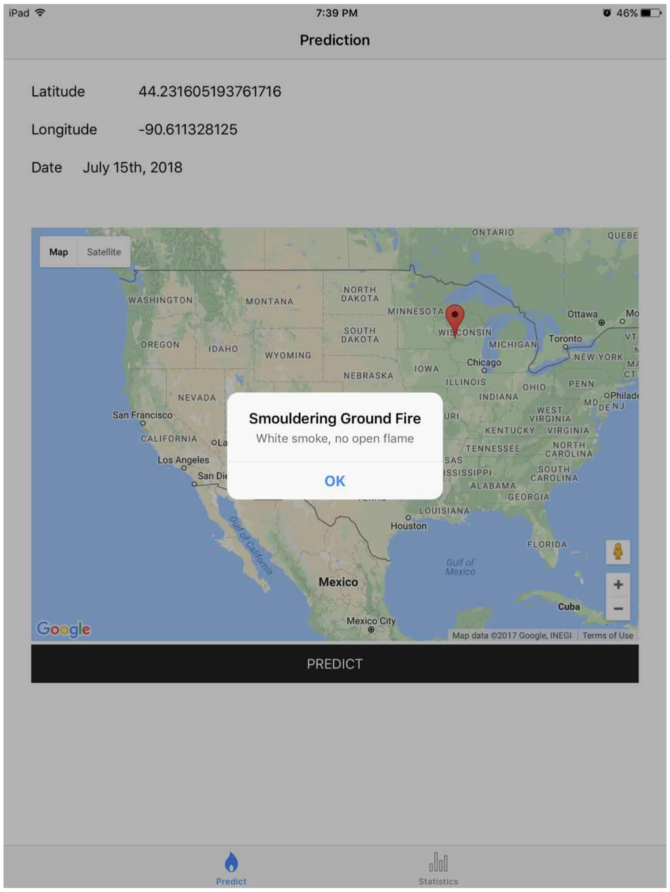

Fig. 4. App User Interface

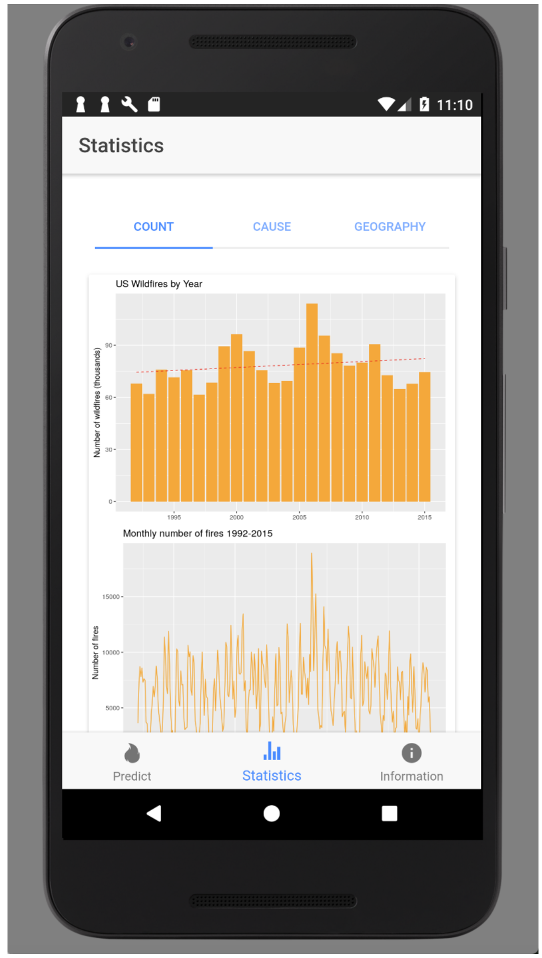

Fig. 5. App User Interface

six percent in average when we ran the full dataset. Similarly for the same dataset we get the accuracy around sixty-five percent.

3) K Nearest Neighbour Classifier: The advantage of using a k nearest neighbour classifier is that it works greats when trying to ignore outliers. It can be considerably effective with a large dataset such as the one we obtained. The speed of the algorithm depends on the number of near neighbours the classifier is looking at. If this number is extremely large, it would take an immense amount of time on a typical personal computer to run the classifier. In order to make this classifier feasible on a dataset like ours, we would have to make the compromise of lowering the number of neighbors it looks at to keep the running time low. This leads to less accuracy of the classifier but makes the classifier more practical.

For our project, we decided that this classifier would work best in one specific case after testing different classifiers. We decided that for a future mobile application update we will allow the user to enter latitude, longitude, state, cause of fire and time of the year (day) and have our classifier predict the size of the fire. This feature would be extremely useful for government institutions and fire departments. If a fire has started somewhere, they would be able to use our classifier to predict the size of the fire and properly allocate the necessary resources to put out a fire of such size.

The K nearest neighbour classifier we have developed splits the data into a 70% training set and a 30% testing set. We were able to use the entire data set of 1.88 million entries after cleaning up and mapping some entries from strings to numbers using python. After a lot of testing we decided that a nearest neighbour value of 5 for the classifier would be a resonable compromise of accuracy for speed. We achieved a final accuracy of 59% with this classifier.

4) Algorithms Implementation: Scikit-learn is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, and k-means, and is designed to inter-operate with the Python numerical and scientific libraries NumPy and SciPy. We used it to build all of our classifiers. Pandas was also used to manage and organize our dataset.

E. Backend

The backend will be implemented using Go, Python and Heroku. An API will created using Go and will allow the user to request information. The request will result in reading

Fig. 6. App User Interface

information or in execution of a Python program (our data mining algorithms).

F. PROJECT TASKS

Main project tasks can be broken into:

- Cleaning the data from the source
- Developing the backend
- Developing the mobile application
- Developing the data mining algorithms

G. TOOLS, LANGUAGES, PLATFORMS

For tracking the history of software development we will use Git. For tracking project progress we will use Trello. The framework that we will use for building the mobile application will be Ionic. The Backend will be built using Go, Python and Heroku. Our code will be hosted in BitBucket for remote access.

H. PROJECT TIMELINE

1) Design: The design of our project can be broken down into three main components:

- Data mining algorithms
- Backend infrastructure
- Mobile application

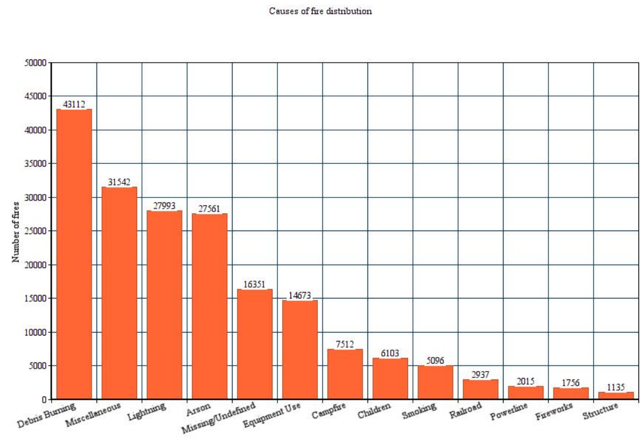

Fig. 7. Diagram for causes of fire distribution.

The mobile application will communicate with the backend over HTTP(S) enabling the user to access our data mining algorithms and view wildfire information.

The design of each component will be specified later in the project.

2) Implementation: The mobile application will be implemented using the Ionic Framework. The backend will be implemented using Heroku, Go and Python. The data mining algorithms will be implemented using different prediction algorithms learned throughout the semester.

3) Evaluation: For the evaluation, we will mainly focus on the classification. we will build a model with training data, and we use this model for test data. After these processes, we will have the accuracy.we will improve our model in order to get high accuracy.We will also be focusing on the training time[3]. If the training takes a long time, we can make the change in our algorithm, model, and dataset.

4) Final Presentation: The final presentation will be a powerpoint slides analyzing the performance of our data mining algorithms along with an interactive demo of the mobile application.

We will explain how we built the mobile application, data mining algorithms and backend.

Assigned tasks for following weeks below:

I. Weekly Milestone

- Week of October 18: High level requirement and design specification of the three main project components: (Mobile application, Backend, and data mining algorithms). In this stage the RATs (Riskiest Assumption Tests) will be defined. Setup of repositories and discussion of technologies to be used will also take place.
- Week of October 23: Imported the data from the dataset using scikit learn to test the dataset. Picking different features from the dataset, and building charts for the features. Weekly group discussion on algorithms and user interface.

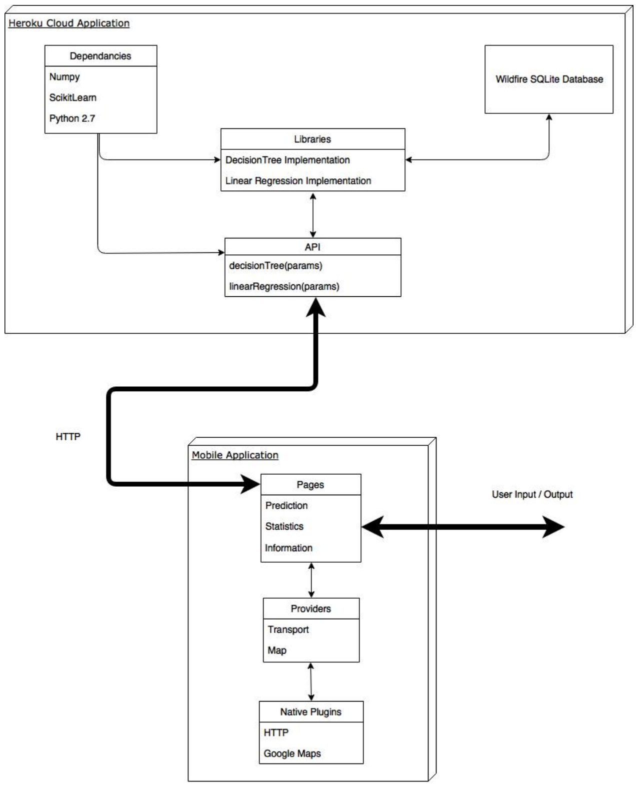

Fig. 8. Diagram for Screen shot for high-level overview.

- Week of October 30: Group discussion on the backend information. Discussing some methods that connects server from the app. Making a decision on the interface and functionality of the app. Deciding what programming language we are going to use in order to implement the app.
- Week of November 6: Figuring out the relationship between the app and database. and building a graph to show this relationship. Check the structure of the back end. List all the possible algorithm we can use for the training set.
- Week of November 13: Splitting the testing data and training data from the dataset. using the decision tree algorithm and do the predicting. Using k-fold validation for the evaluation. Using different training data, and comparing the accuracy to make the model better. Implementing the app.
- Week of November 20: Algorithms testing and accuracies measurement.
- Week of November 26: More Algorithms testing and more accuracies measurement. Implementation server, backend, user interface, project integration and configurations.
- Week of November 28: Final Presentation and final report writing.

J. ROLE AND ASSIGNMENT

1) Honghu Lin: Honghu Lin is an undergraduate student in Computer Science. When everything becomes more

```
import os
from flask import Flask, request
import sys
sys.path.append('/app/server/libs/decisiontreeExample/')
import decisiontree

app = Flask(__name__)

@app.route("/")
def test():
    return "test"

@app.route("/tree")
def tree():
    x = decisiontree.DecisionTree()
    mytree = x.createDecisionTreeModel()

    return str(mytree)

@app.route("/examplePost", methods=['POST'])
def postExample():
    json = request.get_json(force=True)
    return str(json['example'])



if __name__ == "__main__":
    port = int(os.environ.get("PORT", 5000))
    app.run(host='0.0.0.0', port=port)
~
```

Fig. 9. Code Snapshot for transport provider for HTTP requests

```
import { Injectable } from '@angular/core';
import { HTTP} from '@ionic-native/http';
import 'rxjs/add/operator/map';


@Injectable()
export class TransportProvider {

BASE_URL = "https://wfa-server.herokuapp.com/"


  constructor(private http: HTTP) {

    }

  get(endpoint: string) {
        return new Promise((resolve, reject) => {
            this.http.get(this.BASE_URL + endpoint, {}, {}).then(data => {
                        resolve(data.data)
                }).catch(error => {
                        reject(error)
                });
        });
  }
```

Fig. 10. Code Snapshot for prediction page with the map

and more data-oriented, Honghu believes that learning data mining is really the core to understand informations. As a programmer with a knowledge base of different algorithms in data mining, he would like to be participating every bits of the project, mostly in design and evaluation stages.

2) Kun Ye: Kun Ye comes from China, he is interested to the data science area. Currently, he is a student of computer science major. In the past, He did a lot of studying on algorithm and mathematics. The reason he joins this project is that he wants to get more practice with his implementation of algorithms. So He will be doing the algorithm analysis and prediction for this project.

3) James Leathy: James Leahy is a Software Engineering student. He is interested in distributed applications and computer communications. He is excited to help implement the backend and mobile application.

```
import {TransportProvider} from '../../providers/transport/transport'

declare var google;

@Component({
  selector: 'home-page',
  templateUrl: 'home.html',
  providers: [TransportProvider]
  })

export class HomePage {

  @ViewChild('map') mapElement: ElementRef;
  map: any;
  latitude : any;
  longitude : any;



  constructor(public platform: Platform, private transport: TransportProvider, private googleMaps: GoogleMaps) {
  }


  ionViewDidLoad(){
        this.initializeMap();
  }

 initializeMap() {
       this.platform.ready().then(() => {
        let minZoomLevel = 5;
        let mapOptions = {
           zoom: minZoomLevel,
           center: new google.maps.LatLng(38.50, -90.50),
           mapTypeId: google.maps.MapTypeId.TERRIAN
        }
        this.map = new google.maps.Map(this.mapElement.nativeElement, mapOptions);
              var position = new google.maps.LatLng("48.4283", "123.3649");
              var dogwalkMarker = new google.maps.Marker({position: position, title: "Testing"});
              dogwalkMarker.setMap(this.map);
           });

   }

  sampleHttpRequest(){

         this.transport.get('tree').then((data => {
              console.log("here is the model: " + data)
       }));
  }
}
```

Fig. 11.   Code Snapshot for map part 2

4) Mauricio de Paula: Mauricio is a fourth year Computer Science with Software Engineering option student. He really enjoys programming and has always had an interest in Science. There is such an immense amount of data available today that allows Science to make future predictions. Mauricio believes that he can use his previous programming and Statistics experience to help implement a prediction algorithm.

REFERENCES

[1] 1.88 Million US Wildfire - 24 years of geo-referenced wildfire records, https://www.kaggle.com/rtatman/188-million-us-wildfires, [On-line;accessed 18-October-2017]

[2] California wildfire disaster could bring fiscal pain to communities for years to come,https://www.cnbc.com/2017/10/13/california-wildfire-disaster-could-bring-local-fiscal-pain-for-years.html[On-line;accessed 18-October-2017]

[3] Data Mining: Concepts and Techniques http://rizalespe.lecture.ub.ac.id/files/2015/10/DM-04-1-Klasifikasi.pdf[On-line;accessed 18-October-2017]

[4] Applications of Spatio-Temporal Data Mining and Knowledge Discovery (STDMKD) for Forest Fire Prevention, "http://www.isprs.org/proceedings/XXXVI/part7/PDF/148.pdf"[On-line; accessed 18-October-2017]

[5] Investigation of general indicators influencing on forest fire and its susceptibility modeling using different data mining techniques, "http://www.sciencedirect.com.ezproxy.library.uvic.ca/science/article/pii/S1470160X15007359?via"[On-line; accessed 18-October-2017]

IV. NOTE

Our team member, a graduate student named Onyekachi Nwamuo, dropped the class after the midterm but our group will continue working on this project.