Proposal for Beat the Heat - A Wildfire Prediction System

Kaylee Pham, David Shin, Saulo Rubio, Sergio Ramirez
Tyler Poplawski, David Ward, Lior Zlotikman*
{kaylee.pham.704,david.shin.904,saulo.rubio.491,sergio.ramirez.754}@my.csun.edu
{tyler.poplawski.456,david.ward.761,lior.zlotikman.183}@my.csun.edu
Department of Computer Science
California State University, Northridge
Northridge, California

ABSTRACT

While much research has been conducted on how to predict wildfires with great accuracy, wildfires are still a major ongoing issue due to climate change. There is not a strong enough model for predicting the creation and spread of wildfires. Thus, a better model is still needed to predict the creation and spread of wildfires. This paper addresses reducing the damage and effect of wildfires in California. The solution presented in this paper uses data mining to extract important factors causing the creation and spread of wildfires, feeding it to neural networks and support vector machine models to predict the creation and spread of wildfires and evaluation of the prediction result.

KEYWORDS

Neural Networks, Support Vector Machine, Big Data, Remote Sensing Data, Wildfire Prediction

1 INTRODUCTION

Throughout this paper, we position machine learning prediction as a solution for the ongoing wildfire crisis. Between January and October 2020 wildfires burned 3.75 million acres, killed 29 people, and destroyed 8,169 structures in California [1]. Machine learning prediction algorithms are presented as an effective solution through the use of neural networks and support vector machines to the creation and spread of wildfires. As machine learning is a relatively new field of research, there are limited studies to which methodologies and algorithms would be the most effective for our case study. This is evident when observing the related works for our projects as the furthest the research goes is 2016. There is not a model that produces accurate enough results to help minimize the damage and effect of wildfires. The project begins with preparing and feeding the data to the neural network and support vector machines models. These models in turn predict the creation and spread of wildfire. The final part of the design measures the quality and accuracy of the prediction.

The rest of this paper is organized as following: Section 2 discusses related work of wildfire prediction; Section 3 presents the framework of the project; Section 3.3 discusses the methodologies that will be used for this project; Section 5 goes over the timeline of the project.

*All authors contributed equally to this research.

2 RELATED WORK

There has been a sizable amount of work done on the topic already, however previous groups used different combinations of machine learning models as well as datasets with varying success.

2.1 Support Vector Machine

Researchers use a combination of meteorological data and the US forest fire database. The meteorological data consists of temperature data as it pertains to weather, humidity, rain and snowfall levels among other data. The US forest fire database included data such as geographical coordinates, area affected by fire, and severity, all in comma separated value (csv) format. The two datasets were chosen due to their reliability. Researchers demonstrate differences between some of the most popular machine learning models, and identifies SVM as having the greatest accuracy in this domain. Additionally, in this comparison, data fusion and binary and multiclass classification were used [4].

Researchers use data from Moderate Resolution Imaging Spectroradiometer (MODIS) satellite sensors. We haven't determined the exact features we will select for our database but we will likely choose features related to what they used: normalized difference vegetation index (NDVI or health of crops), land surface temperatures (LST), and thermal anomalies. This paper also compares Neural Networks against SVM, and demonstrated that in their test, a NN has 98.32% accuracy while an SVM has 97.48% [7].

2.2 Neural Network

In a research paper by Liang et al., three different neural network models were tested against each other, those being a BPNN model, a Recurrent NN model, and the LSTM model mentioned earlier. The goal was to determine which of these methods is best to build a prediction model in order to help firefighters and emergency personnel assess the risk and spread of a fire before it grows too large. The results of this paper conclude that the LSTM model produced the best predictions of the three, with an accuracy of 90.9% [3]. Contrary to this discovery, a conference paper by Lall and Mathibela used a slight modification of BPNN called the Resilient Back-Propagation algorithm (RPROP), which resulted in the system's overall performance having an accuracy of 97%, as well as 87% precision and 88% recall [2].

Both papers proved to have created successful wildfire risk prediction systems, however they also faced similar drawbacks such as ensuring the model was not overfitting the data. To avoid this, Liang et al. performed a multi-collinearity test on the data to remove factors that were proven to skew the model's interpretation

rather than benefit it. They also made note that an overall limitation of the system was due to the modeling data coming from a single area [3].

The authors of this paper, [8], propose the Particle Swarm Optimization (PSO) algorithm to train a Neural Network. For this, they use the RMSE(Root Mean Squared Error) to compare the results of the PSO algorithm with the results of a Backpropagation algorithm. This paper can be useful in our project when we compare the result of the predictions of our models.

This paper overlaps with our project in using satellite images and past fire record data along with CNNs [10]. They created a model where satellite images of past fire data and a 2D CNN, with 2 convolutional layers w/ 32 and 64 nodes using Sigmoid and ReLU activation functions respectively.

2.3 Datasets

Three groups had a large influence on the data chosen, Sayad et al. [7] among others [6] and [9], as the same combination of data that they used was also chosen for the project: past fire records, satellite data, and meteorological data. The Sayad et al. paper was especially influential in that the source of data used in satellite imagery were also borrowed (NASA's MODIS satellite data) as well as the features selected from satellite data: land surface temperature, normalized difference vegetation index, and land surface temperatures.

A number of groups used some combination of the same datasets mentioned above. Zhang et al. [10] used a conjunction of satellite images and past fire record data. Storer and Green [8] and Liang et al. [3] both used a combination of past fire record databases and meteorological data to train their models. Lall and Mathibela [2] showed promising results using vegetation and environmental features to train their highly accurate artificial NN. Perumal and Zyl's [5] focused on satellite data, specifically the Visible Infrared Imaging Radiometer Suite (VIIRS) satellite instrument for the area of South Africa. The authors indicated for future work that MODIS data would likely improve upon their work, which also played a role in the decision to use such data.

3 DESIGN

As shown in Fig. 1, this project is composed of three major components: preparation, model building, and quality measurement. First part of preparation is gathering datasets from multiple data sources. Next, explore and clean the data, and apply feature engineering to preprocess the data into a new dataset that will be used in the next component. For model building, the new dataset will be split into two sets for training and testing. Next, the training set will be fed into the model and validated after the training phase is completed. If the model is not properly validated, it will loop back to training until it passes the validation phase. Finally, test the model against the testing dataset. This process will be applied on the Neural Network (NN) model and the Support Vector Machine (SVM) model. Output the results from testing and send it to the last phase. In quality measurement, the prediction result will be observed and analyzed. Additionally, the quality measurement phase will reprocess the data if improvement is needed. Then compare NN and SVM accuracy and evaluate them to choose a best model for deployment.

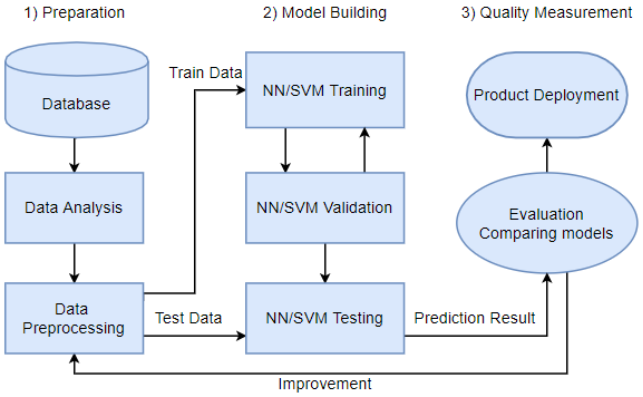

Figure 1: Design Overview

3.1 Preparation

To prepare a dataset for this project, the first step is accessing the Remote Sensing Data from the Land Processes Distributed Active Archive Center (LP DAAC). Next, select California region and the time frame from 1/1/2019 to 12/5/2020, and using data tool AppEEAR to extract the area sample data. Lastly, apply data preprocessing steps such as: data (conversion, cleaning, clipping, interpolation, extrapolation, extraction,integration).

3.2 Model Building

For model building, the first step is splitting the data into two subsets for training and testing. Next, feed the data into training model, validate the model, and test the model with the testing data subset. Finally, output the prediction result.

3.2.1 Neural Network. The neural network component to be used will follow a feed forward neural network style and make use of the backpropagation algorithm. "Feed forward" refers to a non-cyclical style of neural network, in which the data is passed through multiple layers of neurons strictly from input to output without looping back. Backpropagation is used to help identify which neurons contribute to errors in the output from the network. This provides useful insight to which weights should be modified in order to produce more accurate results.

Referring to Fig. 2, the following steps of the neural network flow can be traced. First, the training set of data is passed into the input neurons and initial weights are chosen. This data is fed forward and a linear combination of the inputs and weights is calculated, which becomes the input for the hidden layer(s). From this layer the data passes through an activation function, typically a sigmoid function is used, and compared against a threshold value to determine if that neuron's output is passed on. Linear combination is again calculated before arriving at the output layer. Here is where error checking takes place, and the backpropagation algorithm occurs if weights need to be modified. Finally, this entire process is repeated until the best fitting weights are discovered and the model's accuracy is maximally improved.

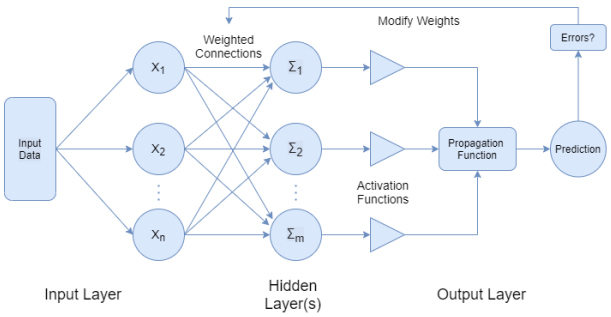

Figure 2: Neural network workflow

3.2.2 Support Vector Machine. Support Vector Machines (SVM) are commonly used for classification and regression problems. The main objective of an SVM is to find the optimal hyperplane for the classification of two classes. An easy way to explain how an SVM works is by using a 2D space representation of two different classes linearly separable. The optimal hyperplane is the line passing between these two classes, creating a maximum margin that serves as the host of the support vectors. Unfortunately, not all datasets can be linearly separable; in a multidimensional space, Kernel functions are used to shape the hyperplane. To find the optimal hyperplane in a multidimensional space, a Support Vector Machine(SVM) has the following major components:

- **Feature Selection**: helps create new features so that the Kernel function can transform and find boundaries in the dataset.
- **SVM Kernel**: functions used to shape the hyperplane in a multidimensional space.
- **Classifier (SVM)**: finds the optimal hyperplane and the support vectors that define the maximum margin.

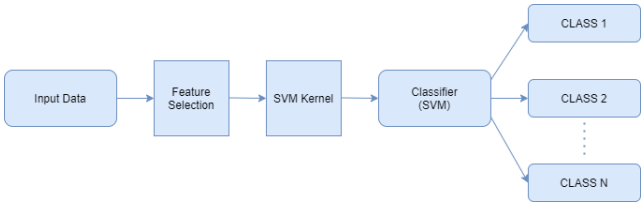

Figure 3: Support Vector Machine

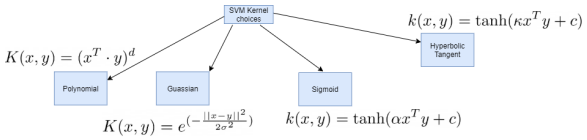

Figure 4: SVM Kernels

SVM Kernel. For the SVM algorithm we choose a kernel, a map from often single dimensional space to n-dimensional space. This is done in the hopes that a higher dimensional space will allow the data to be classified more efficiently. And as previously mentioned in the section above, being able to find a plane that will linearly separate the data is an extremely integral part of the SVM model. For our data and design we found that the following 4 kernels are potential candidates: polynomial, Gaussian, Sigmoid, and Hyperbolic Tangent. In figure 3 above we can see that for each given kernel, we have an associated function. And interestingly, choosing any such kernel will allow us to make predictions based only on our original feature selections.

We found that the polynomial kernel is often associated with having high measure of accuracy in the realm of of image processing. The Gaussian kernel is looked at as being more general purpose. That is to say that it primarily operates without being given prior knowledge of input data. Moreover, the remaining two kernels, Sigmoid and Hyerbolic Tangent, form an excellent pair with neural networks. What we hope to accomplish by selecting a kernel model, is to choose one such model that will help us deliver the most accurate and reliable results.

3.3 Quality Measurement

The Quality Measurement phase is the last phase in the design and the work flow can be observed in Fig. 5. In this phase, we will be collecting the prediction results from our trained models, applying evaluation methods, comparing and finalizing our results.

The results from the prediction phase will be collected and the accuracy will be calculated for both the Neural Network and Support Vector Machine Model. The accuracy can be measured in one of two ways. It can be measured through the summation of True Positive and True Negative predictions, divided by the total summation of True Positive, True Negative, False Positive and False Negative predictions. Alternatively, the accuracy can be measured by using a statistical model (RMSE, MSE, MAE, etc).

- **True Positive** - actual and predicted outcomes are the true.
- **True Negative** - actual and predicated outcomes are false.
- **False Positive** - actual outcome is true, but predicted to be false.
- **False Negative** - actual outcome is false, but predicted to be true.

The statistical models calculate the average distance of error between the predicted and the actual outcome. After the accuracy has been calculated, we will compare the results.

The accuracy of each model be compared to each other as well as the scores from previous works to see if adjustments are needed. If adjustments should be made, we will go back into the Data Preparation Phase or more specifically, the Data Preprocessing Phase, as shown in Figure 5, to make adjustments to parameters. The adjustments will be making are either to change the weight of the parameters or to replace them. After we have reached the desired results, we will finalize them by writing our report on our process and findings.

3.3.1 Experiments. We will be experimenting with different feature spaces, weights, and pre-processing techniques.

The feature space will go through multiple iterations until the optimal group is found. The correlation of each feature to the target feature will be measured as well as correlations to each feature to

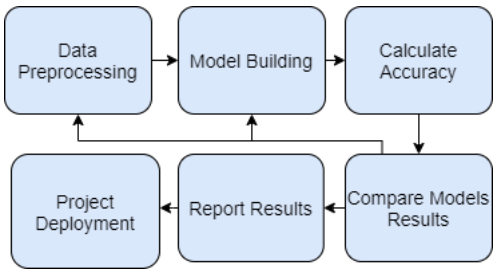

Figure 5: Quality Measurement Design which shows the work flow for this phase

see what features should be kept in order to avoid the Curse of Dimensionality.

In addition to designing the feature space, the weight of each feature will be adjusted to identify the weights needed for the neural network to optimize performance.

Finally, the different pre-processing techniques will be used to see how missing values should be handled. Also, since we are using raster data, we may need to use pre-processing techniques to clean the data in order to make it usable.

4 TIMELINE

The timeline table shown in Table 1, shows the estimated start and end dates of each major phase. Each of the major phases will be divided into smaller sub-phases. The Preparation Phase will be divided into Data Exploration and Data Preprocessing, Model Building will be divided into Training, Validating and Testing the model, and the Quality Measurement will divided into Evaluation and Finalizing sub-phases.

Table 1: Timeline Table

Major Phases	Start Date	End Date
Preparation	9/1/20	2/15/21
Model Building	2/17/21	3/31/21
Quality Measurement	4/1/21	4/8/21

ACKNOWLEDGMENTS

We would like to thank Professor Xijang and California State University of Northridge for the resources and assistance provided throughout the duration of this research.

REFERENCES

[1] 2020 (accessed October 1, 2020). *2020 Incident Archive*. https://www.fire.ca.gov/incidents/2020/

[2] Shruti Lall and Bonolo Mathibela. 2017. The application of artificial neural networks for wildfire risk prediction. *International Conference on Robotics and Automation for Humanitarian Applications, RAHA 2016 - Conference Proceedings* (2017). https://doi.org/10.1109/RAHA.2016.7931880

[3] Hao Liang, Meng Zhang, and Hailan Wang. 2019. A Neural Network Model for Wildfire Scale Prediction Using Meteorological Factors. *IEEE Access* 7 (2019), 176746–176755. https://doi.org/10.1109/ACCESS.2019.2957837

[4] By Hariharan Naganathan, Sudarshan P Seshasayee, Jonghoon Kim, Wai K Chong, and Jui-sheng Chou. 2016. Wildfire Predictions : Determining Reliable. 16, 4 (2016).

[5] Rylan Perumal and Terence L.Van Zyl. 2020. Comparison of Recurrent Neural Network Architectures for Wildfire Spread Modelling. *2020 International SAUPEC/RobMech/PRASA Conference, SAUPEC/RobMech/PRASA 2020* 2020-Janua (2020). https://doi.org/10.1109/SAUPEC/RobMech/PRASA48453.2020.9078028

[6] David Radke, Anna Hessler, and Dan Ellsworth. 2019. Firecast: Leveraging deep learning to predict wildfire spread. *IJCAI International Joint Conference on Artificial Intelligence* 2019-Augus (2019), 4575–4581. https://doi.org/10.24963/ijcai.2019/636

[7] Younes Oulad Sayad, Hajar Mousannif, and Hassan Al Moatassime. 2019. Predictive modeling of wildfires: A new dataset and machine learning approach. *Fire Safety Journal* 104, September 2018 (mar 2019), 130–146. https://doi.org/10.1016/j.firesaf.2019.01.006

[8] Jeremy Storer and Robert Green. 2016. PSO trained Neural Networks for predicting forest fire size: A comparison of implementation and performance. *Proceedings of the International Joint Conference on Neural Networks* 2016-Octob (2016), 676–683. https://doi.org/10.1109/IJCNN.2016.7727265

[9] Dieu Tien Bui, Quang Thanh Bui, Quoc Phi Nguyen, Biswajeet Pradhan, Haleh Nampak, and Phan Trong Trinh. 2017. A hybrid artificial intelligence approach using GIS-based neural-fuzzy inference system and particle swarm optimization for forest fire susceptibility modeling at a tropical area. *Agricultural and Forest Meteorology* 233 (feb 2017), 32–44. https://doi.org/10.1016/j.agrformet.2016.11.002

[10] Guoli Zhang, Ming Wang, and Kai Liu. 2019. Forest Fire Susceptibility Modeling Using a Convolutional Neural Network for Yunnan Province of China. *International Journal of Disaster Risk Science* 10, 3 (2019), 386–403. https://doi.org/10.1007/s13753-019-00233-1