Severity Sentinel

Predictive Analytics for Wildfire Risk from Cross-Domain Indicators

Rithvik Siddam
Technology and Information Management
University of California, Santa Cruz
Santa Cruz, CA, USA
rsiddam@ucsc.edu

Blake Roberts
Technology and Information Management
University of California, Santa Cruz
Santa Cruz, CA, USA
blmorobe@ucsc.edu

ABSTRACT

In recent years, wildfire activity has increased across the United States, prompting an increased need for tools that go beyond predicting fire occurrences. Severity Sentinel introduces a machine learning approach focused on forecasting the potential impact of any wildfire at any given location within the United States. Using the FPA FOD dataset, which contains 28 years of wildfire-related data and over 2.3 million records [1], we trained a multi-output regression neural network to estimate three key metrics: expected agricultural loss rate (EALR_PFS), expected building loss rate (EBLR_PFS), and expected population loss rate (EPLR_PFS), each expressed as a percentile. Our model, developed in PyTorch and trained on carefully cleaned and encoded features, achieved R^2 scores above 0.86 and a mean absolute error of approximately 0.06 across all outputs. These results demonstrate that predictive models can offer more accurate insights into fire severity, enabling improved disaster preparedness, policy planning, and resource allocation.

Introduction

Wildfires pose an increasingly severe threat to communities, infrastructure, and ecosystems across the United States. Traditional predictive models for wildfires have largely focused on forecasting where or when a wildfire might occur. However, such models often leave decision-makers without clear information about the severity of a fire's potential impact data that could prove essential for proactive response and mitigation efforts [2]. This project aims to address that gap by shifting the predictive focus from wildfire occurrence to wildfire impact.

In an exploration of existing research in the field of severity prediction, we found that existing models utilize decision tree models including a Boosted Regression Tree [3] and Random Forest model [4], both focusing on finding areas of high risk and explanatory variables, similar to our own project in purpose but using less complex models which may have less luck in predicting severity.

Leveraging a comprehensive dataset from the Fire Program Analysis Fire-Occurrence Database (FPA FOD), we built a deep learning model capable of estimating the potential damage of a wildfire at a specific location. The dataset includes diverse physical, biological, social, and administrative features, which we cleaned, analyzed, and reduced to a subset of 54 relevant attributes through correlation analysis and logical filtering.

Using these features, we implemented a feedforward neural network with three output nodes corresponding to percentile-based estimates for agricultural, building, and population loss. The model was trained using an 80/20 train-test split, ReLU activation functions, early stopping, and a sigmoid output layer. The resulting predictions, evaluated using mean absolute error and R^2 metrics, demonstrated strong performance across all three targets.

By providing a framework for estimating wildfire impact severity, Severity Sentinel offers a novel and actionable perspective for stakeholders involved in emergency management, environmental policy, and community resilience planning.

CCS CONCEPTS

•Computing methodologies~Machine learning~Machine learning approaches~Neural networks•Applied computing~Physical sciences and engineering~Earth and atmospheric sciences~Environmental sciences•Information systems~Information systems applications~Spatial-temporal systems~Geographic information systems•Applied computing~Operations research~Forecasting•Computing methodologies~Machine learning~Learning paradigms~Supervised learning

KEYWORDS

Wildfire prediction, Neural networks, FPA FOD dataset, Disaster modeling, Machine learning, Supervised learning, Damage estimation, Geospatial data analysis

1 Data Preparation

The data for our model was taken from the FPA FOD database, which contains 28 years of wildfire data from 1992 to 2020. This database contains the Physical, Biological, Social, and Administrative data of every tracked wildfire in the United States of America using 308 features and 2.3 million data points. We distilled this 5.02GB of data down to 0.21GB of relevant and actionable information. This took the form of 54 features with the same 2.3 million data points.

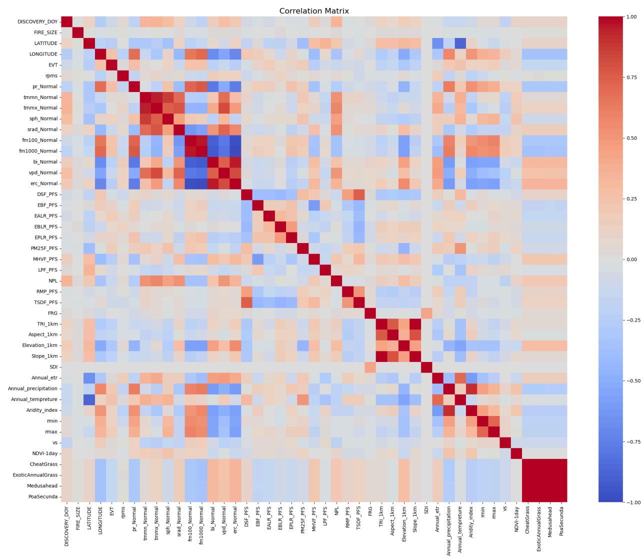

Figure 1: **Correlation Matrix of Cleaned Features**

A visual representation of each feature's correlation with one another. A strong red square indicates a strong positive correlation while a strong blue square indicates a strong negative correlation.

1.1 Cleaning

We cleaned the database in several stages, following a combination of intuition and model testing to ensure the final features contributed to a successful model. In general, we, followed these stages:

1. Clean by description: Remove any data that couldn't logically affect the severity of a wildfire (e.g., identifiers like fire ID or cannot influence fire severity) and any data that is reported twice (e.g., temperature as a mean and temperature as a percentile both report the same data over the same period of time so any patterns in one will be reflected in the other).
2. Clean by data: Remove any data that either isn't easily processed by the model or has too many missing values to be interpolated.
3. Clean by correlation: Based on correlation scores found alongside the correlation matrix, remove any data points that are both low correlation with our target variables and whose values are logically unimportant enough to overlook (e.g., population density was low correlation and there was enough data for predicting Building Loss and Population Loss without it. Cheat Grasses had a similarly low correlation but there were fewer agriculturally significant variables, so they were kept after ensuring the model's performance did not change greatly).

After following the steps for data cleaning, we split our features into categorical, numerical, and target variables. The final step of cleaning the data was to get our categorical data into a form that was processable by the model. We did this by using one hot encoding: splitting every possible output of each categorical variable into a separate binary variable.

2 Model Building

Our model is a feedforward deep neural network. In order to create this, we depended on many python libraries: pandas, pytorch, scikit-learn, and matplotlib. These libraries provided the following value to our model:

1. pandas: Tabular Data Management. This was utilized to load and manipulate our data and allowed for easy selection of our features and target variables.
2. pytorch: Deep Learning Framework. We defined our custom model using torch.nn.Module and computed our numerical operations using torch.tensor. Training and data handling were also handled by pytorch.
3. scikit-learn: Preprocessing and Evaluation. This was utilized to preprocess our variables, split our data into test and training data, and evaluate our results using mean absolute error and R^2.
4. matplotlib: Visualization. We utilized matplotlib before our model for the Correlation Matrix and after our model for visualizing our predictions as well as the loss curve.

We trained our model on 80% of the cleaned data, leaving the other 20% for validation, converting this data into pytorch tensors to enable model training. PyTorch tensors are required for building neural networks since it makes computation more efficient, enables gradient tracking, splits data into batches, and even allows faster processing using GPU acceleration.

Our model architecture consists of an input layer, 3 hidden layers, and an output layer. The input layer is straightforward, taking in all of our cleaned features. Our first hidden layer transforms our data into 256 dimensions based on a learned matrix multiplication. Then that layer drops 30% of the neurons during training to prevent overfitting. Additionally, the layer uses a ReLU activation to introduce non-linearity using max(0, x). The subsequent hidden layers do the same but instead reduce dimensionality from 256 to 128 and then from 128 to 64. Our output layer projects the output into our three target variables: EALR_PFS (Expected agricultural loss rate), EBLR_PFS (Expected building loss rate), and EPLR_PFS (Expected population loss rate). This layer uses a sigmoid activation function to ensure the outputs are between 0 and 1, which is perfect for our outputs since they are percentiles.

2.1 Optimization

Our strategy also utilized an optimization strategy. We utilized Smooth L1 Loss, also known as Huber Loss, which helps regression tasks involving potential outliers. It dynamically behaves like Mean Squared Error for small differences but like Mean Absolute Error for larger differences, causing the model to have reduced sensitivity to outliers. We additionally utilized the Adam (Adaptive Moment Estimation) optimizer with a learning rate of 0.001. This combines the benefits of adaptive learning rates and momentum-based updates, allowing for fast convergence and more stable updates.

Another key optimization strategy we used was a Learning Rate Scheduler—ReduceLROnPlateau. To improve training stability, we chose this scheduler to monitor validation loss and adjust the learning rate automatically based on signs of local minima or plateaus. By tracking our validation loss, we could catch if the validation did not improve for 5 epochs, leading to the automatic reduction of our learning rate in an attempt to free the model, leading to better convergence and generalization.

2.2 Training

The training strategy for our neural network consisted of 300 epochs with early stopping set to trigger after 10 consecutive epochs of no validation improvement. For each epoch, we used backpropagation on mini-batches and evaluated validation loss at the end of each. This means that each epoch, the model is performing a forward pass to generate predictions, computing loss, and then performing a backward pass to compute gradients which updated the model's parameters using Adam. We additionally found that model checkpointing, saving the best performing model as the model ran, helped eliminate the effects of overfitting. This allowed our results to reflect the model at its best state, rather than the last state the model ends up in. Additionally, we prevented unnecessary computation with early stopping. We set the model to stop training if validation loss did not improve for 10 consecutive epochs. In general, we saw the model stop near 250 epochs.

3 Results

From the correlation testing in our data cleaning stages, we found a few key features correlated highly with the values of our target variables. The most significant of these per each target are as follows:

- EALR_PFS - Vegetation type and precipitation
- EBLR_PFS - Diesel particulate matter and proximity to hazardous waste
- EPLR_PFS - Dead fuel moisture, precipitation and energy burden

These findings parallel the findings of previous models which further validate that we did not accidentally clean any core variables [2, 3].

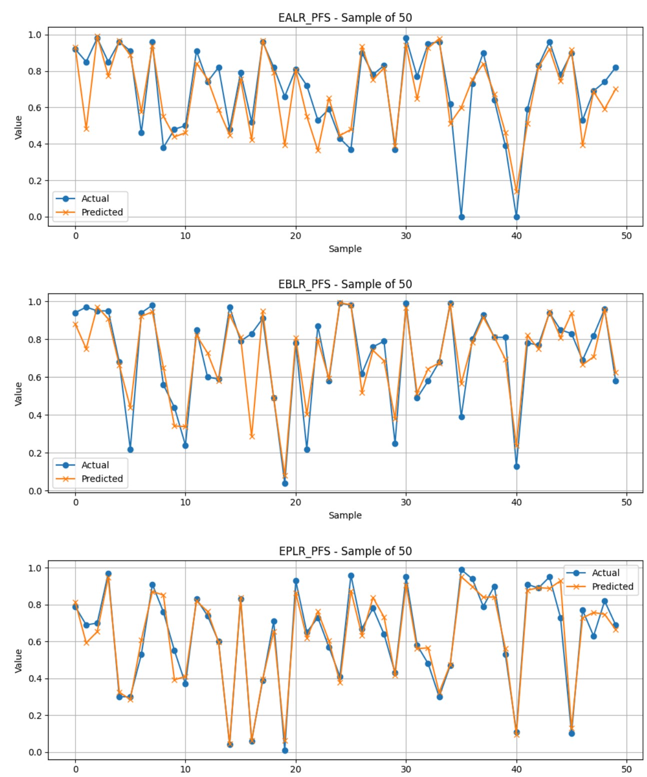

Figure 2: **Target Variables Actual vs. Expected Value Graph**

A visual representation of each target variable's performance. Orange lines are the predicted values, while blue lines are the actual values..

Our neural network model resulted in an overall R^2 of 0.8763 and an overall Mean Average Error of 0.0613. For the individual variables, it resulted in an R^2 and Mean Average Error of 0.8568 and 0.0632, of 0.8731 and 0.0599, and 0.8991 and 0.0607 for EALR_PFS, EBLR_PFS, and EPLR_PFS respectively. This is also reflected in Figure 2 where we can see clearly that EALR_PFS had the hardest time predicting the findings, showing clear outliers in performance near sample 35. We tested the model again with more of the removed variables and saw clear downgrades in both accuracy and efficiency. We tested the model again without including the CheatGrass variables but found no significant improvement, implying that the performance issues stem from something else.

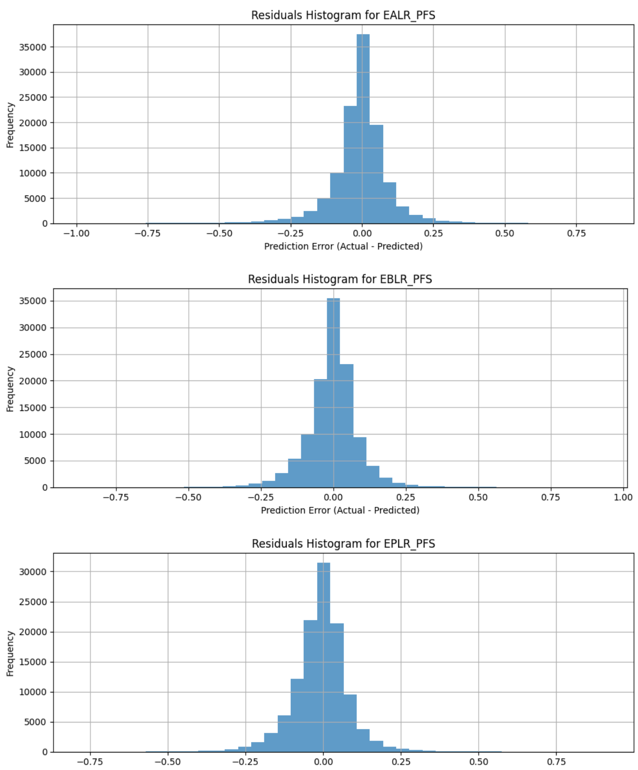

Figure 3: **Target Variables Residuals Histogram**

These results show that we are able to successfully predict the severity of wildfires since our R^2 value of 0.8763 shows that our model explains 87.63% of the variance in wildfire severity, which is very strong, especially in a real world scenario with the high likelihood of unexplored variables. Additionally the MAE score of 0.0613 suggests that we are off the mark by an average of 6% since our model is predicting between 0 and 1.

In comparison to the existing models, we can see that we have significantly improved the best predictions in the field. Though it is a different metric, the area under the curve (AUC) of the existing decision tree model achieved was only 0.72 on average. Our MAE of 0.0613 is also on the scale of 0 to 1 and it can very easily be concluded that our model has improved predictions since a perfect score for an AUC is 1 and a perfect score for an MAE is 0.

3.1 **Limitations**

We found that agricultural loss was the most difficult to predict, possibly due to the fact that we found less highly correlated variables related to it. This is very likely a sign that more features need to be measured in related to agricultural contexts. It is also possible that the measures of loss are not consistent enough, since it is much easier to count building loss in terms of dollars or population loss in terms of relocations and deaths than it is to measure the loss of agriculture.

Some limitations that prevent our project from being immediately actionable is the lack of personal actionable insights, which is the one place we feel our model falls short of previous models. Previous models showed impacts per region and created actionable visualizations to help regulators more easily provide support in areas where they are necessary [3]. Additionally, providing scores as a percentile without an easy means of conversion to a monetary value limits our appeal to investors. If we were to launch this project or present our findings to a government agency, we would first need to solve those gaps in our service.

3.2 **Findings**

The objective of this model was to move beyond traditional wildfire occurrence prediction and provide a more reliable estimation of wildfire severity than the existing models. We targeted severity across three critical dimensions: agricultural, building, and population loss. The findings from our model strongly support this goal and suggest that our multi-output neural network can offer meaningful, actionable predictions that can inform preparedness, policy, and resource planning.

Our model achieved an overall R^2 score of 0.8763 and a mean absolute error of 0.0613, indicating that it can explain a significant proportion of the variance in fire severity outcomes. This high level of predictive performance, especially in the context of a real-world dataset with considerable variability, highlights the potential of our approach to modeling complex environmental phenomena. The breakdown by target variables (EALR_PFS: 0.8568, EBLR_PFS: 0.8731, EPLR_PFS: 0.8991) further illustrates the model's capacity to capture the nuances of different kinds of wildfire impact.

Ultimately, these results affirm that it is possible to improve the current predictions of wildfire severity across multiple impact categories. The implications are transformative: emergency responders can better allocate resources, policymakers can identify high-risk communities before disasters strike, and long-term planning can integrate localized predictions of wildfire impact into land use, environmental protection, and climate adaptation strategies.

Conclusion

Severity Sentinel demonstrates that deep learning models can meaningfully fill the gap of wildfire prediction between occurrences and anticipated impact, offering stakeholders a more accurate tool for disaster preparedness and response. The existence of occurrence prediction is important, but by introducing this reliable severity predictor alongside existing models, we can maximize our wildfire fighting potential. We can plan both legislation and reallocate resources off of these predictions. By leveraging a personalized version of a large and trusted dataset and applying a carefully optimized neural network, we successfully predicted percentile-based estimates of agricultural, building, and population loss with high accuracy, achieving an overall R^2 of 0.8763 and a mean absolute error of just 0.0613. Our results show that deep learning can outperform traditional models in this space, while also re-emphasising key environmental, social, and economic factors that drive wildfire severity. These improved metrics are more actionable for disaster response, policy planning, and long-term resilience efforts. Though we can still improve our project as a whole by implementing metrics that are more actionable and collecting more agricultural data, we still believe that Severity Sentinel is a powerful tool that improves the current wildfire severity prediction accuracy and conveys the importance of tracking and acting upon certain key indicators which should be and incorporated into current wildfire management strategies to increase preparedness, mitigate losses, and support data-driven policy development.

Contributions

Rithvik was in charge of coding, the Neural Network, and filtering the data based on labels and data types. Blake further cleaned the database using correlation scores and helped test the Neural Network with different fine tuning parameters.

Github

Spring2025 TIM147-Group7
Contact rithvik.siddam@gmail.com for access
https://github.com/RithvikSiddam/TIM147/tree/main

REFERENCES

[1] Pourmohamad, Y., Abatzoglou, J., Belval, E., Short, K., Fleishman, E., Reeves, M., Nauslar, N., Higuera, P., Henderson, E., Ball, S., AghaKouchak, A., Prestemon, J., Olszewski, J., & Sadegh, M. (2023). Physical, social, and biological attributes for improved understanding and prediction of wildfires: FPA FOD-Attributes dataset. Zenodo. https://doi.org/10.5281/zenodo.8381128.

[2] Dansbecker. 2020. Wildfire Analysis. Kaggle. https://www.kaggle.com/code/dansbecker/wildfire-analysis. (Accessed April 27, 2025).

[3] Parks, S. A., Holsinger, L. M., Panunto, M. H., Jolly, W. M., Dobrowski, S. Z., & Dillon, G. K. (2018). High-severity fire: evaluating its key drivers and mapping its probability across western US forests. Environmental Research Letters, 13(4), 044037. https://doi.org/10.1088/1748-9326/aab791.

[4] U.S. Forest Service Rocky Mountain Research Station. Fire Severity Mapping: Past Fires and Predicting Future. https://research.fs.usda.gov/rmrs/products/multimedia/webinars/fire-severity-mapping-past-fires-and-predicting-future. (Accessed June 8, 2025).

[5] Pourmohamad, Y., Abatzoglou, J., Belval, E., Short, K., Fleishman, E., Reeves, M., Nauslar, N., Higuera, P., Henderson, E., Ball, S., AghaKouchak, A., Prestemon, J., Olszewski, J., & Sadegh, M. (2024). Supplement of Physical, social, and biological attributes for improved understanding and prediction of wildfires: FPA FOD-Attributes dataset. Earth System Science Data. https://doi.org/10.5194/essd-16-3045-2024-supplement.