

Wildfire Risk Prediction with Remote Sensing and Transfer Learning

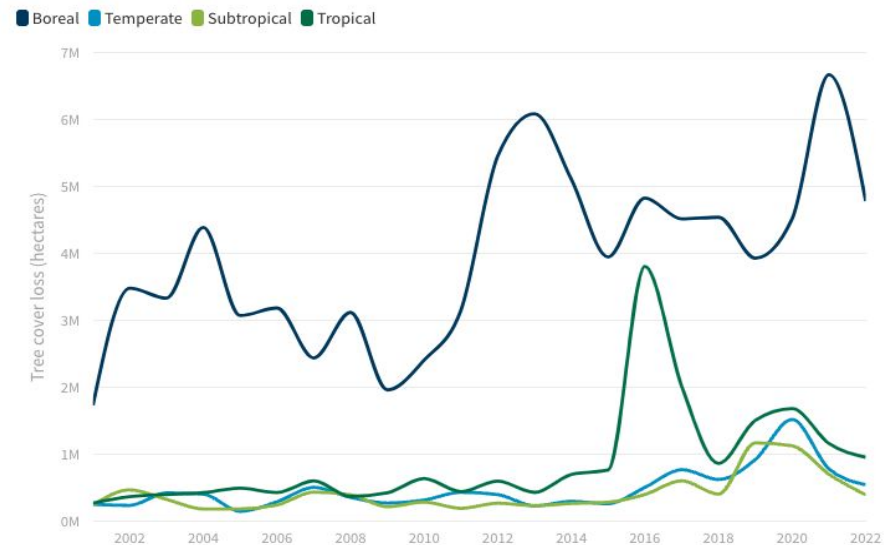
Derek Yao, Nora Povejsil, Marlon Fu
DATASCI 207 – Spring 2024



Problem Statement

- **Background:** Since the 1980s, wildfires have intensified due to factors like climate change, posing significant threats to ecosystems and economies.
- **Need for Assessment:** Early wildfire risk assessment is crucial for environmental protection and effective resource allocation by authorities.
- **Potential Impact:** By analyzing photos, machine learning models can assess areas for wildfire risk, aiding in proactive measures like vegetation management and controlled burns.

Annual tree cover loss due to fires by climate domain, 2001-2022



Key Objectives

1. Assess the effectiveness of different machine learning models in analyzing satellite imagery for wildfire risk prediction.
2. Focus on understanding the performance dynamics of various model architectures rather than developing a production-ready methodology.
3. Utilize an existing repository of pre-processed satellite images to fit 10 robust models.

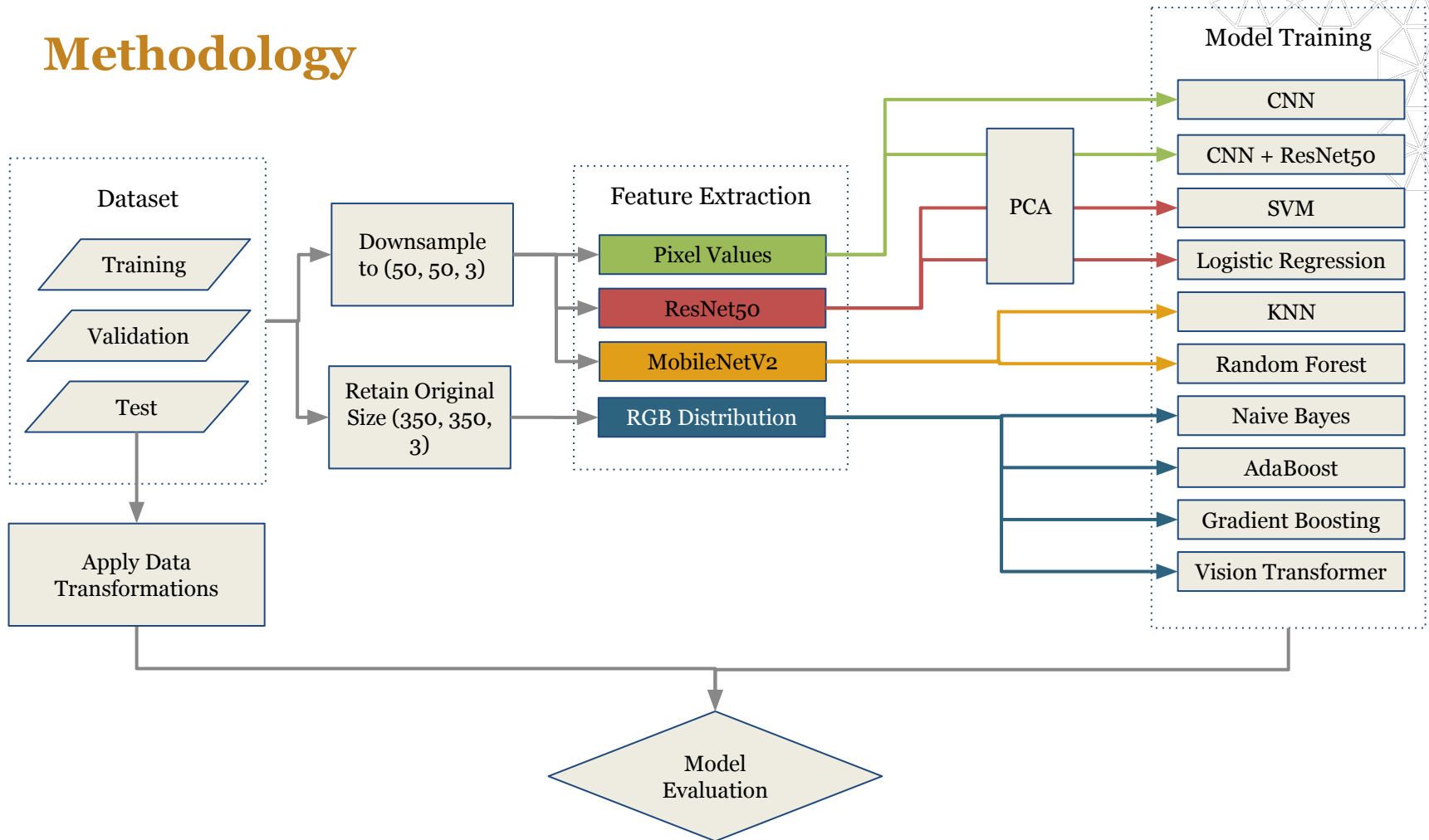
Performance Metrics

- Accuracy
 - correctly classified proportion
- Precision
 - true positives / all positive predictions
- Recall
 - true positive / all actual positives
- F1 Beta Score
 - the harmonic mean of precision and recall
- F2 Beta Score
 - weights recall 2x more than F1
 - false negatives are more costly than false positives

Success: >0.85
precision and recall.

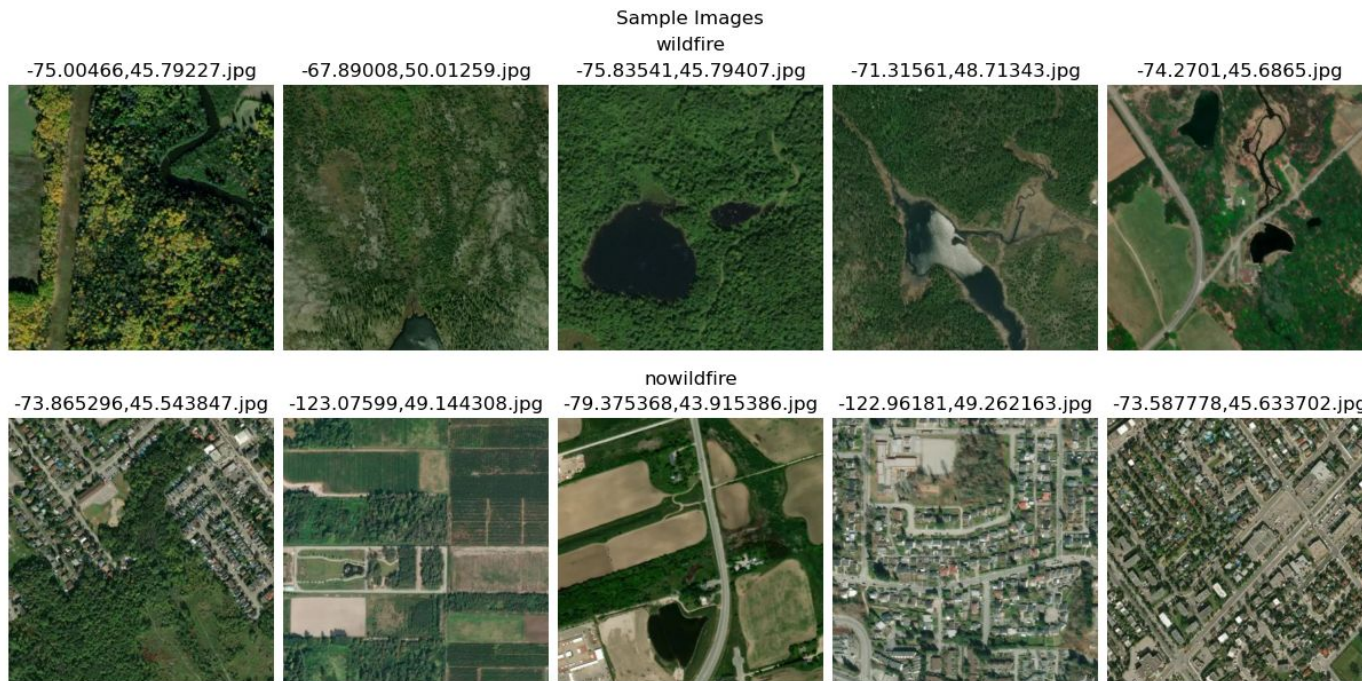
Failure: <0.75
precision and recall.

Methodology



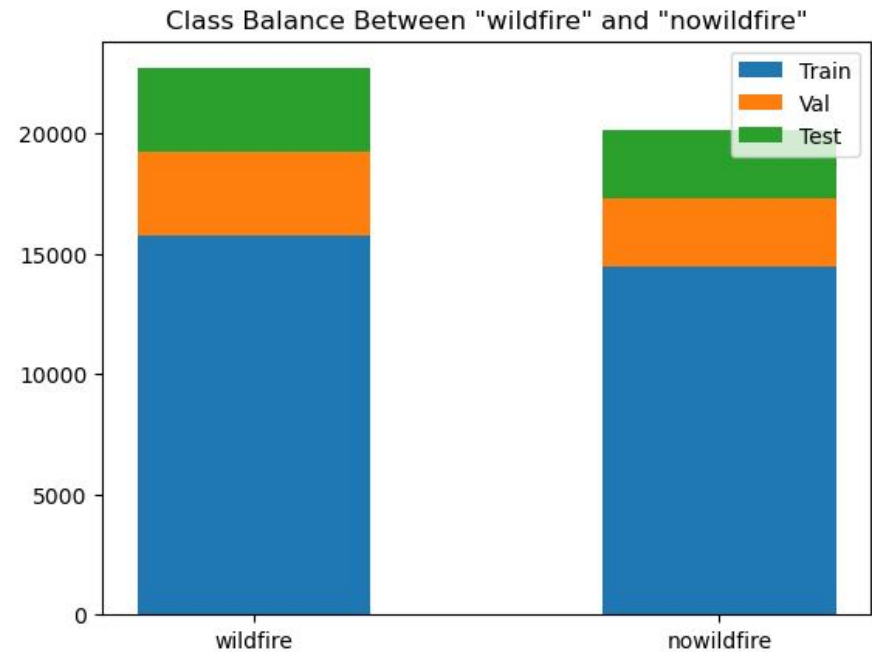
Data Overview

- For this study, we are use the [Wildfire Prediction Dataset \(Satellite Images\)](#) on Kaggle.com by Abdelghani Aaba (derived from [Forest Fires - Open Government Portal](#)).



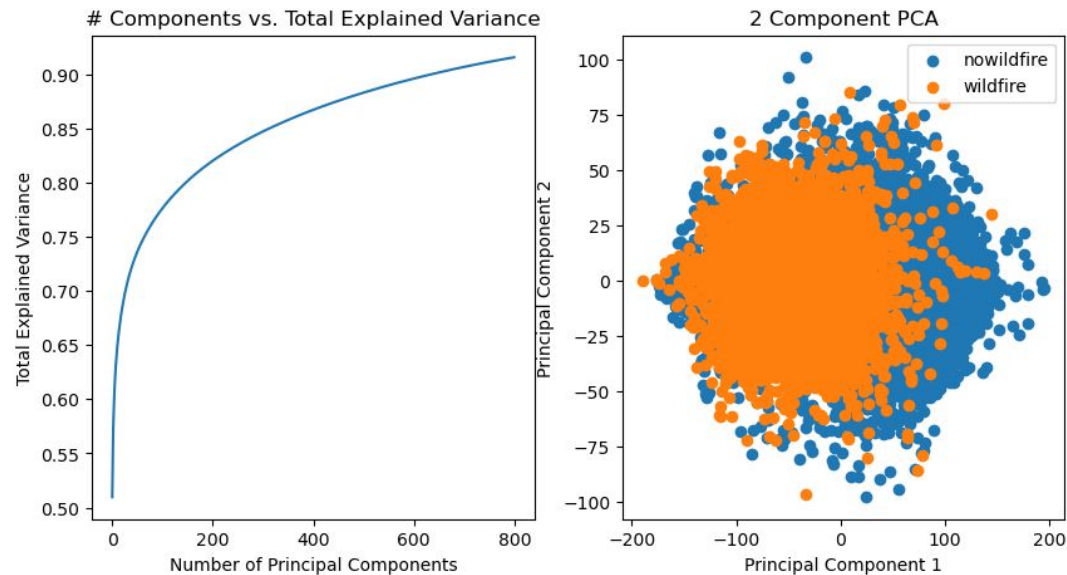
Class Balance

- Labels
 - Wildfire: 22,710 images
 - No wildfire: 20,140 images
- Train/Test/Validate:
 - Train: 30,250 images (75%)
 - Validation: 6,300 images (15%)
 - Test: 6,300 images (15%)



Raw Pixel Values as a Baseline Feature

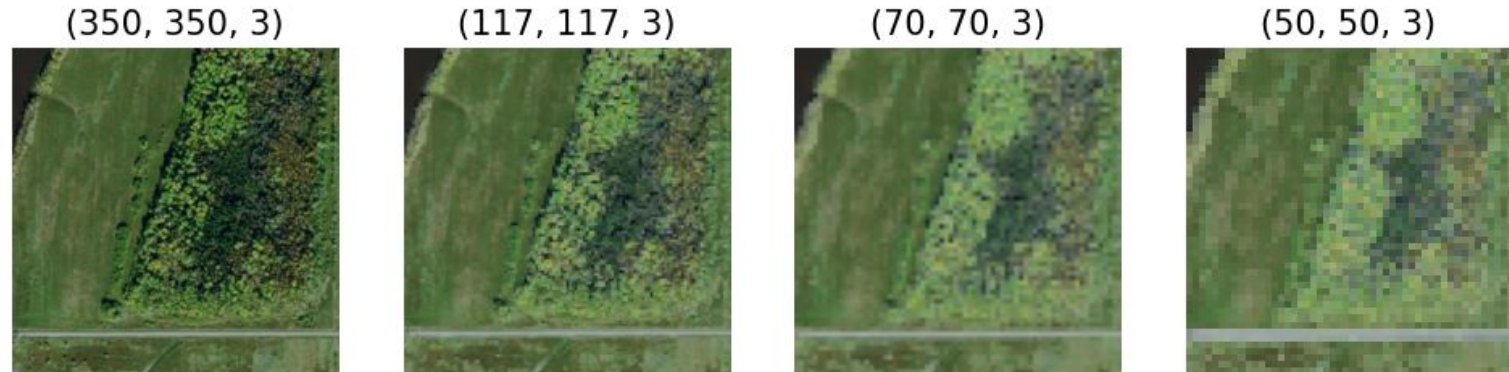
- Most simple and elementary feature
- Flattens the raw pixel values into a 1D vector for each image
- Models using this feature: Baseline Logistic Regression, Baseline CNN



Midterm Results From Baseline Models

Metric	Logistic Regression with Image Flatten	CNN w/ Sigmoid, SGD Optimizer	CNN w/ Sigmoid, Adam Optimizer	CNN w/ Relu, Adam Optimizer
Train Accuracy	90%	53%	86%	93.2%
Validation Accuracy	89%	48%	53%	55.3%

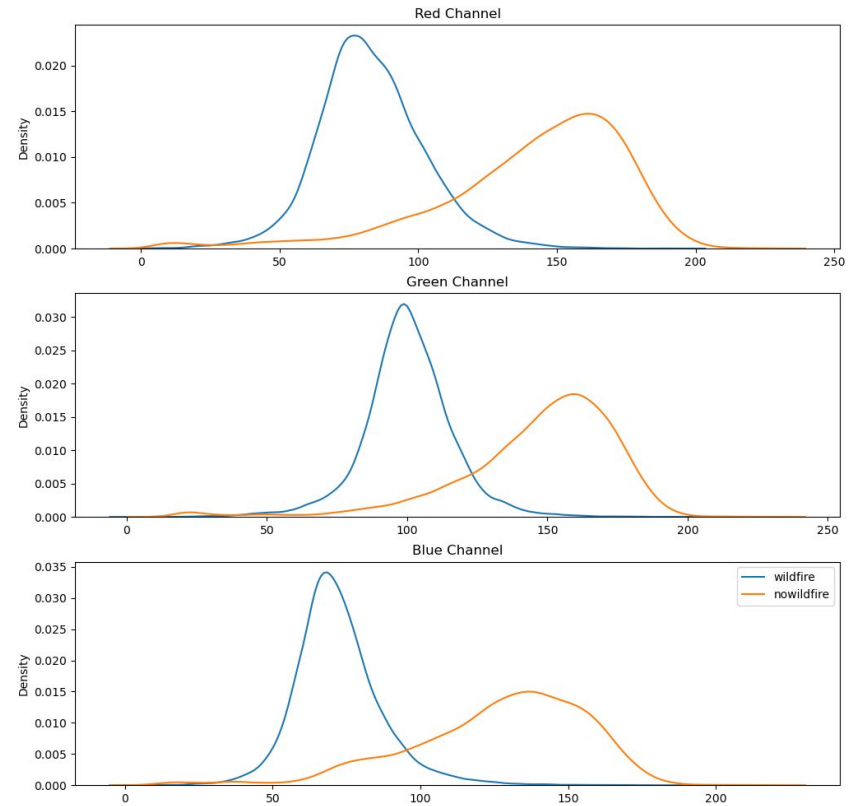
Image Downsampling



- Used max pooling on varying block sizes
- Downsampled to (50, 50, 3) based on visual inspection
- Aids in speeding up computation time for downstream tasks without losing too much information

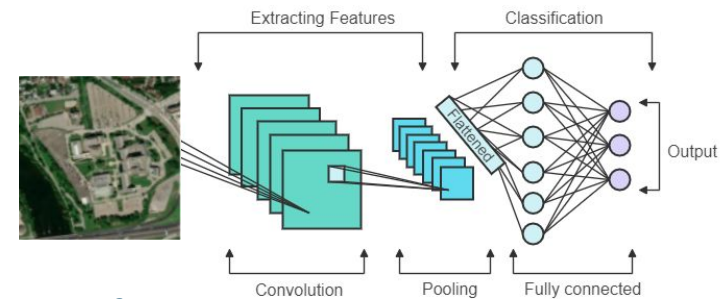
Feature: RGB Distribution

- Simple and intuitive representation of RGB color distributions
- Normalized histograms for all 3 color channels constructed as a single vector
- Robust to variations in images
- Low computational cost
- Models using these features: Naive Bayes, AdaBoost Classifier, Gradient Boosting Classifier



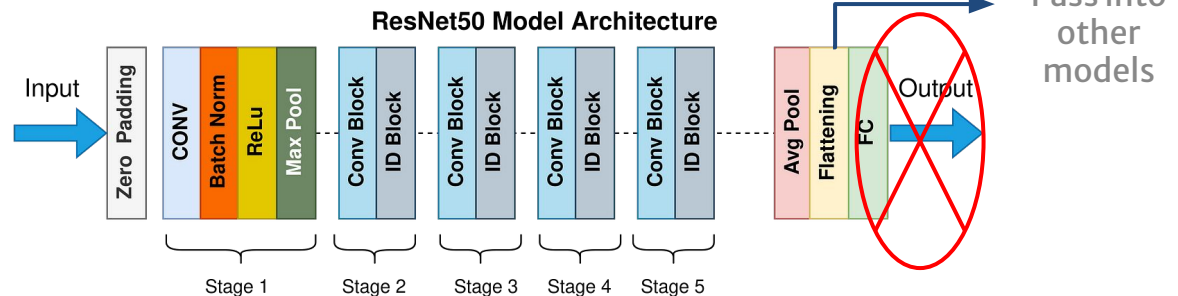
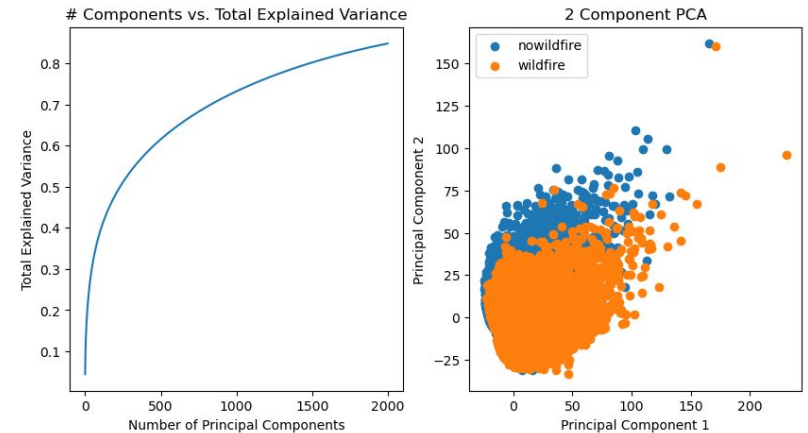
Feature Extraction: MobileNetV2

- Neural Network
- Expansion and depth-wise convolutional layers
- Linear activation function (to prevent the model from becoming too complex)
- Lightweight & accurate models
- Captures both fine-grained details and high-level patterns for classification tasks.
- Models using these features: Random Forest, KNN, and



Feature Extraction: ResNet-50

- Remove fully connected layer of the ResNet50 model, keep only the convolutional base.
- Pass satellite images through the ResNet50 convolutional layers, to extract features.
- Apply global average pooling to reduce the spatial dimensionality.
- Obtain a feature vector representing each input image.
- Pass the extracted features into classifiers for wildfire prediction
- **Models using this feature:** Logistic regression, SVM



Source: [Suvaditya Mukherjee, Towards Data Science](#)

Model comparison

	Model	Accuracy	Precision	Recall	F1	F2
1	Gradient Boosting Classifier	0.969	0.980	0.964	0.972	0.967
2	CNN + ResNet50	0.961	0.968	0.961	0.964	0.962
3	Vision Transformer (ImageNet-21k)	0.916	0.957	0.941	0.927	0.956
4	AdaBoost Classifier	0.957	0.968	0.954	0.960	0.956
5	Logistic Regression + ResNet50	0.945	0.961	0.939	0.949	0.943
6	Naive Bayes	0.824	0.767	0.979	0.860	0.927
7	CNN	0.918	0.938	0.911	0.917	0.925
8	SVM + ResNet50	0.918	0.937	0.913	0.925	0.917
9	K-Nearest Neighbors + MobileNetV2	0.895	0.912	0.897	0.905	0.901
10	Random Forest Classifier + MobileNetV2	0.889	0.903	0.895	0.899	0.896

[*Reference for Accuracy Metrics](#)

Top 3 Models based on F2 Score

1. Gradient Boost Classifier
2. CNN + ResNet 50
3. AdaBoost Classifier

Model 1: Gradient Boosting Classifier

- Used sklearn's Gradient Boosting Classifier
- Trains a robust classifier sequentially, with each new weak learner trained to correct the errors made by the previous ones.
- Gradient Boosting fits new weak learners to the residuals of the current ensemble
- Typically achieves high accuracy in datasets with noise

Gradient Boosting Classifier with Downsampling -> RGB feature extraction

Training Accuracy: 0.9585785123966942

Validation Accuracy: 0.956031746031746

Test Accuracy: 0.9625396825396826

Test Precision: 0.9674351585014409

Test Recall: 0.9646551724137931

F1 Beta: 0.9660431654676259

F2 Beta: 0.9652098907418057

Confusion Matrix (Test):

[[2707 113]

[123 3357]]

Gradient Boosting Classifier with RGB feature extraction

Training Accuracy: 0.9658512396694214

Validation Accuracy: 0.9658730158730159

Test Accuracy: 0.9696825396825397

Test Precision: 0.9804265264387964

Test Recall: 0.964367816091954

F1 Beta: 0.9723308706359554

F2 Beta: 0.9675373349478175

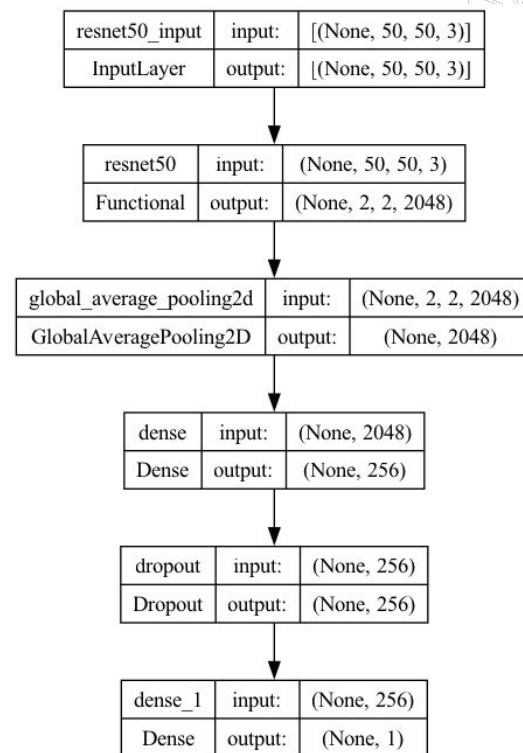
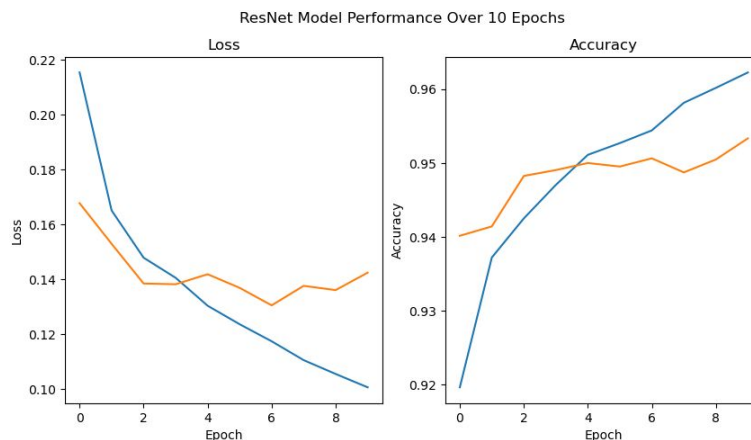
Confusion Matrix (Test):

[[2753 67]

[124 3356]]

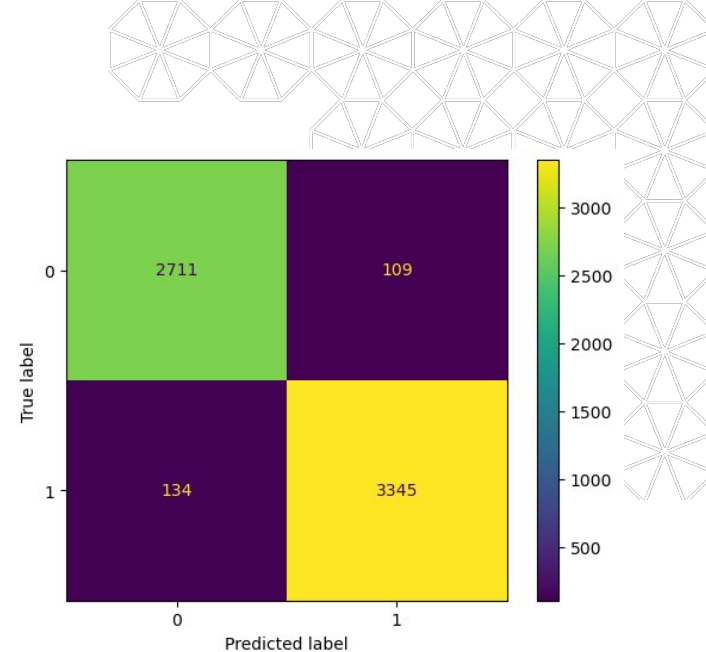
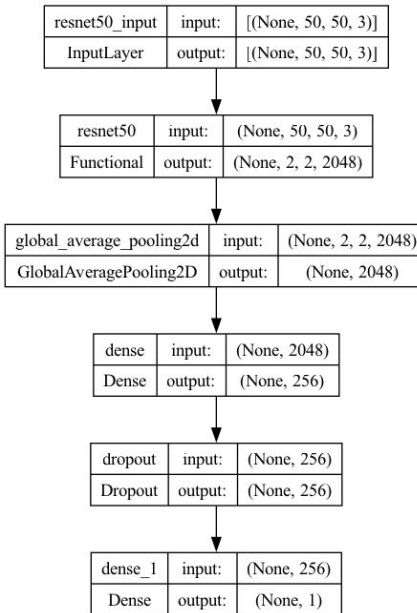
CNN with ResNet50 Pretrained Base Model

- Wrapped CNN around ResNet50 architecture using TensorFlow/Keras.
 - Utilized transfer learning by loading pre-trained ResNet50 model with ImageNet weights.
 - Extracted features using Global Average Pooling layer to reduce dimensionality.
 - Added dense layers for classification with dropout regularization.
- 24,112,513 total parameters
- 524,801 trainable parameters



Model 2: CNN + ResNet 50

- Wrapped CNN around ResNet50 architecture using TensorFlow/Keras.
 - Loaded pre-trained ResNet50 model with ImageNet weights.
 - Extracted features using Global Average Pooling.
 - Added dense layers for classification with dropout regularization.
- 24,112,513 total parameters
- 524,801 trainable parameters



Model 3: Adaptive Boosting Classifier

- Used Sklearn's AdaBoost Classifier Package
- Robust classifier trained by iteratively combining the predictions of multiple weak learners
- Each subsequent model assigns higher weights to the misclassified samples from the previous iteration
- AdaBoost is surprisingly robust with RGB feature extraction vs downsampling

AdABoost Classifier with Downsampling -> RGB feature extraction

Training Accuracy: 0.9439338842975207

Validation Accuracy: 0.9480952380952381

Test Accuracy: 0.949047619047619

F1 Beta: 0.9538726828567323

F2 Beta: 0.9537904477268807

Confusion Matrix (Test):

```
[[2660 160]
```

```
[ 161 3319]]
```

AdaBoost Classifier with RGB feature extraction

Training Accuracy: 0.9498842975206612

Validation Accuracy: 0.9517460317460318

Test Accuracy: 0.9571428571428572

Test Precision: 0.967930029154519

Test Recall: 0.9540229885057471

F1 Beta: 0.9609261939218524

F2 Beta: 0.956772334293948

Confusion Matrix (Test):

```
[[2710 110]
```

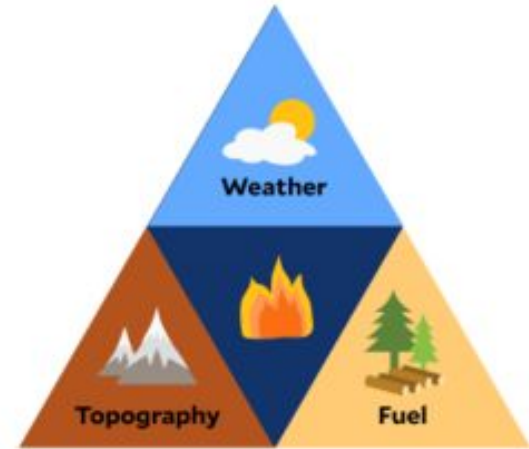
```
[ 160 3320]]
```

Bias and Fairness

- It is well-established that predictive AI algorithms can often be biased when applied to human beings.
- There is little research being done at the moment on geospatial biases and predictive algorithms' unintended biases in disaster prediction.
- Sampling bias:
 - Wildfire boundaries are commonly documented by forests services.
 - “nowildfire” labels need to be sampled from anywhere not within those boundaries.
 - Methods for sampling must ensure comprehensive representation of topographies, land uses, etc.
- Satellite images on the visual spectrum can only capture so much (for example, no vertical information, all birds-eye-view), and are prone to biases because of this.

Constraints and Limitations

- This study does not include data collection methodology
 - Key limitation of Kaggle data: time between photo taken and time of fire is unspecified
 - Methods for sampling images is out of our control
- Wildfire prediction largely dependent on additional factors beyond the visual spectrum
 - Infrared bands
 - Temperature, wind, humidity
- We recognize that this simplified approach to wildfire prediction



Fire Behavior Triangle

The factors involved in the severity, intensity, duration, size, and season of wildfires

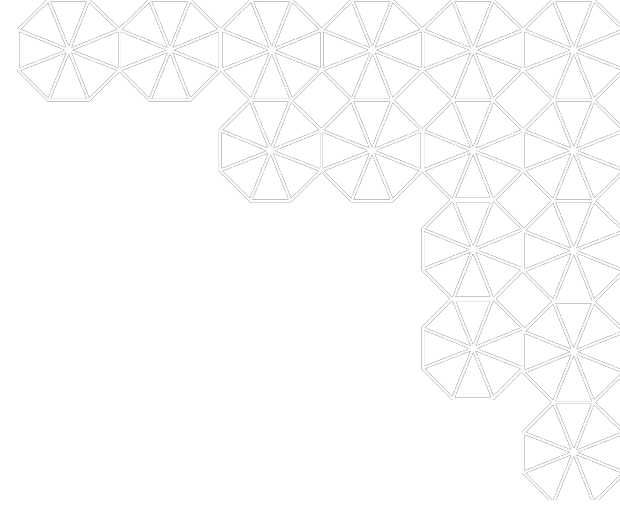
[Source: Yale Sustainability](#)

Final Takeaways

- Binary classification is suitable for wildfire prediction using visual spectrum satellite images alone.
- **All models exceeded our initial goal of >0.85 precision and recall.**
 - The Gradient Boosting Classifier performed the best (**0.964 precision, 0.972 recall, 0.967 F2**)

However

- These numbers only describe how well the models fit and generalize on the data available.
- Being able to predict wildfires in production will require more work in the data collection and sampling methodologies.



Thank you!