CS230

Satellite Imagery based Wildfire Detection: CS230-Winter 2020 Final Paper

Emily Wu
Department of Computer Science
Stanford University
emilyw12@stanford.edu

Glynnis Millhouse
Department of Computer Science
Stanford University
glynnis@stanford.edu

Olamide Oladeji
Management Science & Engineering
Stanford University
oladeji@stanford.edu

Abstract

In this project, we explore the use of one major global satellite - the Sentinel-2 - in order to address wildfire detection. Unlike previous work in literature which focus on the use of Landsat (30m - 60m resolution), a less granular satellite imagery service, our approach uses deep learning on multiple features extracted via the Sentinel satellite (10m - 40m), such as the true color, vegetation index, and moisture content in order to detect these wildfires. This paper describes results of three models. Our baseline model only uses the true color satellite imagery layer with a simple CNN model on 11,000 images for wildfire detection, obtaining an accuracy of 70.63%. Our second and best model adds normalized difference vegetation index (NDVI), Burned Area Index (BAI), and normalized burn ratio (NBR-RAW) layers for a total of 8 channels and we obtain a test set accuracy of 95.09% after hyperparameter tuning and regularization. Finally, we find that a transfer learning model based on the ResNet-50 CNN architecture on 8-channel images led to a reduced test accuracy of 64.09%.

1 Introduction

In light of the increased prevalence and severity of wildfires in recent times, the minimization of response delay has become even more crucial for improved wildfire mitigation. Recent advances in deep learning and computer vision coupled with increased proliferation of frequent satellite mapping across the world can be leveraged in order to address these challenges.

To this end, we propose a deep learning wildfire detection system which is trained on images of historical wildfire locations from the Sentinel-2 satellite. Based on the gaps identified in literature of deep learning for wildfire detection, we explore the use of several image layers obtainable from Sentinel-2's satellite imagery service. In particular, we focus on the use of the true color, the normalized difference vegetation index (NDVI), Burned Area Index (BAI), and the Normalized Burn Ratio (NBR-RAW) layers as channels for data points and train them using various deep learning architectures with sigmoidal (binary) predictions for wildfire existence or not.

,

2 Related Work and Motivation

A number of examples involving the use of various machine learning methods for wildfire detection exist in literature. Most of these machine learning approaches so far have been predominantly on non-image data on historical wildfire incidents. For example, methods such as SVM and logistic regression have been used been used for wildfire detection and predictive risk modeling as seen in [9][10][11].
More recently, a number of related works have tried to address the wildfire detection using deep learning focusing primarily on remotely sensed satellite imagery data [7][12]. N. Toan et al. in [12] build deep learning models using output of the Geospatial Operating Environmental Satellites-16 (GEOS-16) - a very powerful 16-channel satellite with very granular temporal (every 60 seconds to every 10 minutes) and spatial (up to <1km) resolution - to detect wildfires achieving an F-1 score of 94% on the test set. While this paper shows the promise of deep learning and satellite imagery on wildfire detection, its replicability and applicability are limited by both the cost and engineering pipeline challenges associated with dealing with GEOS-16 satellite data. Transfer learning approaches using networks such as Inception-V3 and ResNet-50 have also been applied to the wildfire detection problem with varying degrees of success [8][13][14].

In [14], the authors explore the use of deep learning models in detecting wildfires based on satellite imagery. They specifically focus on using Landsat-8 as their satellite imagery source. Landsat has a temporal resolution of every 16 days, and a spatial resolution that varies between 30m and 60m. Out of 11 bands retrieved from Landsat, 6 channels were used per data point: 3 RGB channels and 3 Infra-Red (IR) bands. Using over 40,000 satellite image datapoints (including augmented images), with each image at 224 x 224 x 6, an 86% training accuracy and 87% test accuracy was reported.
This work while sharing similar motivations with our project, has some significant limitations which our project sought to overcome. First, although free and easy to access, the temporal (every 16 days) and spatial (30 m - 60m) resolutions of Landsat are relatively poor and could significantly affect the validity of the model predictions [3]. The Sentinel satellite, which we use in our project, is similarly free to access but provides better temporal and spatial resolution.
Another potential limitation of the approach of [14] is that the training data of satellite wildfire imagery was sourced using Landsat's own computationally/mathematically estimated fire locations. There is thus a potential for bias due to this coupling and any high test accuracy values may simply be the model learning to detect Landsat's computational estimations rather than the true fire locations. By first retrieving wildfire location-date pairs from the US government database GEOMAC, and subsequently extracting satellite images for the pair, our approach decouples these two processes and mitigates such bias.
Finally, the work in [14] fails to explore other potentially useful bands/layers available from satellites such as Landsat or Sentinel. For example, in addition to IR, Sentinel is also able to provide layers such as Normalized Differential Vegetation Index (NDVI), Moisture Index (MSI), Burned Area Index (BAI) and NBI (Normalized Burn Index), among others. By including these additional layers, it may be possible to develop better performing models that can be deployed for use by wildfire response agencies.

3 Dataset and Features

In order to adequately detect wildfires with deep learning methods, we propose the integration of several datasets. The typical variables which may be significantly correlated wildfire include fuel/vegetation, topography, wind and other weather-related variables (which may affect its spread), as well as features correlated to heat or burn colors [4][5][6]. We sought to integrate those features extractable from satellite imagery. We first retrieved historical wildfire perimeter data from a US Geological Survey database known Wildland Fire Support Geospatial Multi-Agency Coordination (GeoMAC) [1]. The database contains location, perimeter polygons and date of all wildfires in the United States and is stored as 30m x 30m raster level (.shp) shapefiles. We wrote a web-scraping script to download the shapefiles from the GeoMAC website.
We focused on wildfire perimeter data from 2016 to 2019, resulting in 20946 historical wildfire incidents stored in a geodataframe, ultimately using 5000 of these incidents to generate our 11000-image data set. Starting with wildfire incident geodata, we obtain the satellite imagery data of the true color and other layers as follows:

- We use the Sentinel-2 satellite as the source of these satellite images, leveraging on a Web Map Service (WMS) provided by Sentinel-Hub [15]. The WMS also allows us to query and obtain other satellite extracted images/features for those wildfire location-date pairs that correspond to information on Vegetation, Normalized Burn Index and Burned Area Index, and Moisture content.
- For each of the historical wildfire data points, we used the perimeter polygon and date properties to define bounding boxes that are then used to generate corresponding satellite imagery for that wildfire at that location.
- We make sure to standardize the zoom/resolution level for each historical wildfire perimeter polygon and date pair to a 0.5 km radius from the perimeter polygon's inner centroid, retrieving an array of layers (channels) corresponding to a 256 x 256 image for that data point.

As our problem is a classification problem, we also obtain data for the non-wildfire class in two ways. First, using the satellite imagery WMS service, we extract images for the same wildfire location but 150 days prior (to a date for which we assume there was no wildfire). Secondly, to avoid over-fitting to the same locations, we also add additional satellite image tiles of random, non-fire locations in the United States at random dates, to complete our non-fire dataset and serve as 10% of the whole data. It should also be noted that, for the 3 channel RGB-only model, we also augment 10% of the images with horizontal flipping and right rotations.
Our first (baseline) model build uses only the true color RGB images, with each (input) satellite image, for both classes thus having a dimension of (256,256,3). Our second model uses 8 channels of satellite data, for both classes thus having a dimension of (256, 256,8). Our final model uses the same 8-channel dataset as our second model.
Including the two classes, the dataset here thus has 11000 images with 45.5% : 54.5% split between wildfire and non-wildfire classes. Next we do a 80-10-10 training-validation-test split of the data. A sample wildfire image is visualized below.

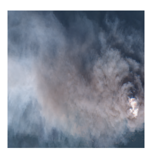


Figure 1: Sample 256 x 256 RGB Satellite Image showing Wildfire

4 Approach

4.1 Baseline Model

Our baseline model is a simple Convolutional Neural Network consisting of 3 convolutional layers. The model uses an Adam optimizer and a binary cross-entropy loss function:

$$\frac{1}{m}\sum_{i=1}^{m}(-y(log(\hat{y})) + (1-y)(log(1-\hat{y})))$$

The following table outlines the model:

Layer	Operation	Kernel/Pool Size	Feature Maps	Stride	Activation
1	Convolution	5×5	64	5	ReLU
	Max Pooling	2×2		1	
2	Convolution	3×3	32	3	ReLU
	Max Pooling	2×2		1	
3	Convolution	3×3	64	3	ReLU
	Max Pooling	2×2		1	
4	Convolution	3×3	128	3	ReLU
	Flatten				
	Dense		128		ReLU
	Dense		1		Sigmoid

4.2 Adding additional channels

In addition to the three RGB channels from the true color layer as used in our baseline model, we add three other layers of data for every image to get a total of 8 channels per image. We specifically add the following layers:

- NDVI: Normalized Differential Vegetation Index $NDVI = \frac{NIR-Red}{NIR+Red}$
  NIR here represents the Near Infra-Red band. The NDVI obtained has 3 channels.
- BAI: This refers to the Burned Area Index.

  $$BAI = (1 - \sqrt{\frac{B06 * B07 * B8A}{B4}}) * (\frac{B12 - B8}{\sqrt{B12 + B8A}} + 1)$$

  The BAI layer combines a band ratio in the red-edge spectral domain (which extracts out vegetation features) with a band ratio in the Short Wave Infra-Red (SWIR) spectral domain which helps in extracting out hot/burned areas. The BAI has one channel.
- NBR RAW Normalized Burned Ratio Raw: $NBR = \frac{B8-B12}{B8+B12}$
  B8 and B12 are refer to pixel values of the 8th and 12th bands of the Sentinel-2 satellite. The NBR_RAW has one channel.

Using these bands, each image then becomes 256 x 256 x 8.

We then build and train a series of models focusing on this multi-channel image data. First, we explored the same CNN as presented in 4.1, with identical data collection, transformation, and split into training, test and validation. We also added dropout layers to the CNN. Second, we explore a transfer learning approach with ResNet-50 on the 8 channel image data.

For both the 3 channel and 8 channel versions, we iterated on different learning rates and optimization algorithms. We iterated on the learning rate on a log-scale, and ultimately decided on 0.01. For optimization algorithms, we also tried RMSProp, and ultimately chose ADAM as the best optimizer. We also experimented with adding dropout layers, which we did decide to include. When making these decisions, we used the classification matrix as a guide.

5 Results

Our baseline model with a simple 3-channel RGB image and standardized zoom levels, achieved the below accuracy metrics:

	Training	**Validation**	**Test**
Accuracy	71.33%	70.03%	70.63%

	Precision	**Recall**	**F1**	**Support**
0.0	0.81	0.60	0.69	596
1.0	0.64	0.83	0.72	504
weighted avg	0.73	0.71	0.70	1100

We therefore have very low variance, but some significant bias in our baseline 3-channel CNN.

It should be noted that running this baseline model without standardizing the resolution of input images led to a higher test accuracy of 79.8%, but we discarded this approach of not standardizing the input images resolution as we believe resolution standardization is more representative of how the model would be used if deployed.

When we added additional channels to our data, we achieved the below accuracy metrics:

	Training	**Validation**	**Test**
Accuracy	78.27%	79.65%	78.65%

	Precision	**Recall**	**F1**	**Support**
0.0	0.70	0.91	0.79	581
1.0	0.85	0.57	0.68	519
weighted avg	0.77	0.75	0.74	1100

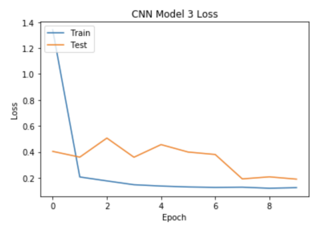
The additional channels thus significantly increased accuracy, reducing bias with no loss in our low variance.

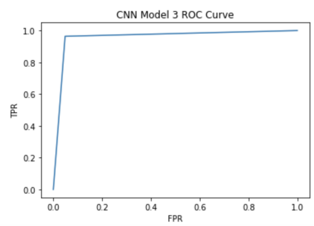
Finally, adding dropout layers (with probability 0.2) led to achieving even better accuracy:

	Training	Validation	Test
Accuracy	95.84%	94.73%	95.09%

	Precision	Recall	F1	Support
0.0	0.98	0.93	0.95	596
1.0	0.92	0.98	0.95	504
weighted avg	0.95	0.95	0.95	1100

Although dropout is a regularization technique and thus is primarily intended to reduce variance, this drastic reduction in our model bias as well was an interesting and welcome result. Our interpretation of this is that since we were only able to train for limited - 10 - epochs in either case, i.e. with and without dropout, our training without dropout was not near convergence. This non-dropout training accuracy may have been higher than the 95.84% obtained with dropout if we had sufficient computational resources to train for longer epochs. However, the dropout may have made sped up training such that the accuracy obtained after only 10 epochs may be the near-converged training accuracy with dropout. Below is the Loss and ROC Curves for our model with 8 channel data and dropout.





We additionally did human labeling on 40 images and achieved an 85% accuracy. We believe that we were able to achieve better than human accuracy due to the addition of more channels of data. Additionally, the true Bayes Optimal Error may be much lower if a wildfire expert were to do the classification.

5.0.1 Transfer Learning

We also explored transfer learning by used the pre-trained ResNet-50 model through Keras. The results below were worse than our more simple model:

	Training	Validation	Test
Accuracy	82.61%	63.49%	64.09%

	Precision	Recall	F1	Support
0.0	0.79	0.47	0.59	601
1.0	0.57	0.85	0.68	499
weighted avg	0.69	0.64	0.63	1100

The worse results from this more complex model is likely due to overfitting, since we do see a large widening of the variance as well as poorly performance overall. Therefore, we stuck with our original model, with 8 channel data, ADAM optimization, and dropout layers, for our final results.

6 Conclusion

The algorithms that worked best included all eight channels, which is intuitive because because the additional data provided relevant information not present in an RGB image. The baseline models with a relatively smaller number of layers worked surprisingly well, while, adding additional layers with the application of ResNet-50 had a lower test accuracy and high variance suggesting an overfitting problem. Finally, adding dropout layers to our custom CNN architecture added a significant boost to accuracy under the limited 10 epochs.

If we had more time, we would have wanted to try transfer learning using VGG. In addition, we would have wanted to train without dropout for much longer to see the achievable training and test set error.

7 Contributions

Olamide Oladeji contributed to data collection, processing, model specification and interpretation. Emily Wu contributed to experimentation and modeling. Glynnis Millhouse contributed to writing, results interpretation, and some modeling.

Please find our code attached at our Github repository: https://github.com/glynnismillhouse/cs230-project/tree/master/Code.

References

[1] U.S. Department of Agriculture, Wildland Fire Support Geospatial Multi-Agency Coordination (GeoMAC) https://www.geomac.gov/services.shtml

[2] United States Geological Service, GloViS https://glovis.usgs.gov/app?fullscreen=0

[3] DevelopmentSeed. Landsat-util https://pythonhosted.org/landsat-util/

[4] NOAA, Climate Data Online (CDO). https://www.ncdc.noaa.gov/cdo-web/

[5] National Renewable Energy Laboratory (NREL). Wind Integration Data (WIND) Toolkit https://www.nrel.gov/grid/wind-toolkit.html

[6] United States Geological Service. 3D Elevation Program. https://www.usgs.gov/core-science-systems/ngp/3dep

[7] D. Radke, A. Hessler and D. Ellsworth. FireCast: Leveraging Deep Learning to Predict Wildfire Spread. Proceedings of the Twenty-Eight International. Joint Conference on Artificial Intelligence, 2019. [Available Online] https://www.ijcai.org/Proceedings/2019/0636.pdf

[8] A. Khan et al. A Survey of the Recent Architectures of Deep Convolutional Neural Networks. ArXiv. 2019. https://arxiv.org/pdf/1901.06032

[9] P. Cortez et. al. A Data Mining Approach to Predict Forest Fires using Meteorological Data Proceedings of the 13th Portuguese Conference on Artificial Intelligence

[10] E. J. S. Nhongo, et. al. Probabilistic modelling of wildfire occurrence based on logistic regression, Niassa Reserve, Mozambique, Geomatics, Natural Hazards and Risk, 2019. 10:1, 1772-1792, DOI: 10.1080/19475705.2019.1615559.

[11] Y. Chang. et al. Predicting fire occurrence patterns with logistic regression in Heilongjiang Province, China. Landscape Ecology, 2013.

[12] N. T. Toan, P. T. Cong. N Quoc, V. Hong, J. Jo. A deep learning approach for early wildfire detection from hyperspectral satellite images 2019 7th International Conference on Robot Intelligence Technology and Applications (RiTA) November 1 - 3, 2019, Daejeon, Korea.

[13] M. Souza, A. Moutinho. and M. Almeida. Wildfire detection using transfer learning on augmented datasets. Expert Systems with Applications, vol. 142, Elsevier. March 2020.

[14] J. Lin, and Y Zhu. Environment-Agnostic Wildfire Detection with Satellite Imagery, Stanford CS230 Class Project, Winter 2018. [Available Online] https://cs230.stanford.edu/files_winter_2018/projects/6935030.pdf

[15] Modified Copernicus Sentinel data [2020] / Sentinel Hub Sentinel Hub, https://sentinel-hub.com/, Sinergise Ltd.