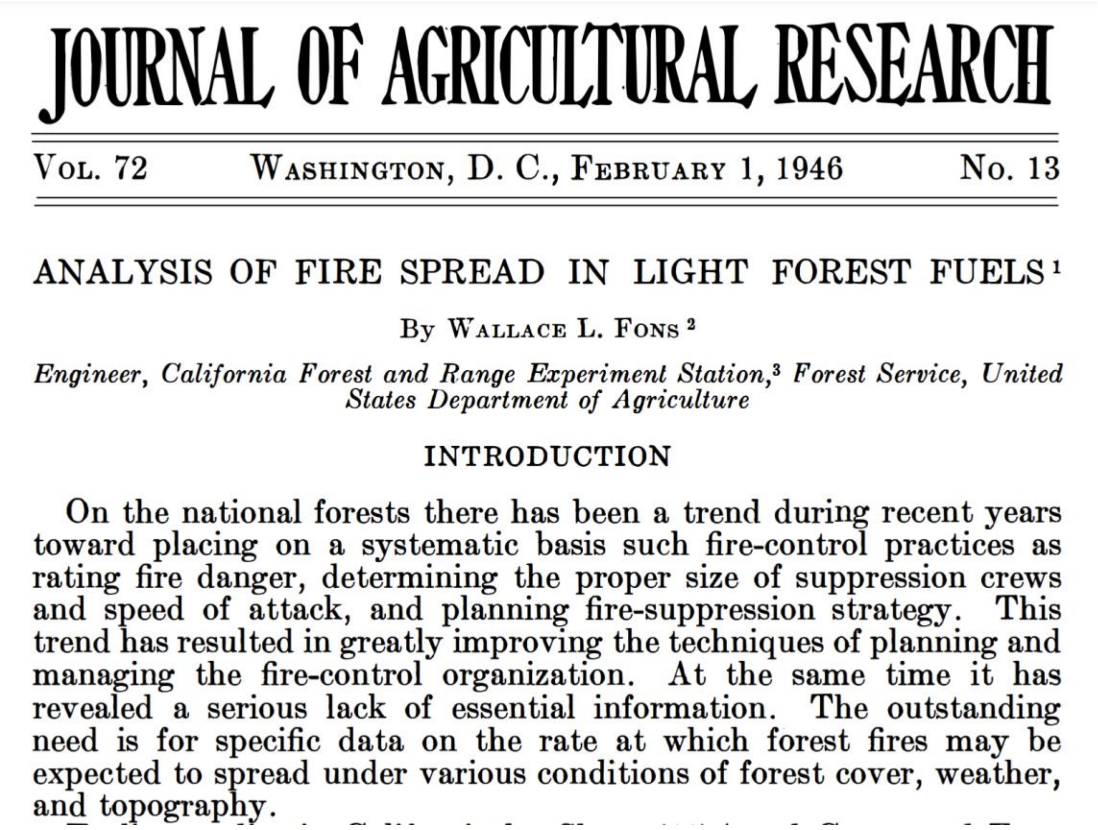
12/12/24

A new high-resolution wildfire spread machine learning dataset

G. Hulsey (UCSB), D. Alderson (NPS), J. Carlson (UCSB)

A short history of wildfire modeling



JOURNAL OF AGRICULTURAL RESEARCH

VOL. 72 WASHINGTON, D. C., FEBRUARY 1, 1946 No. 13

ANALYSIS OF FIRE SPREAD IN LIGHT FOREST FUELS[1]

By WALLACE L. FONS[2]

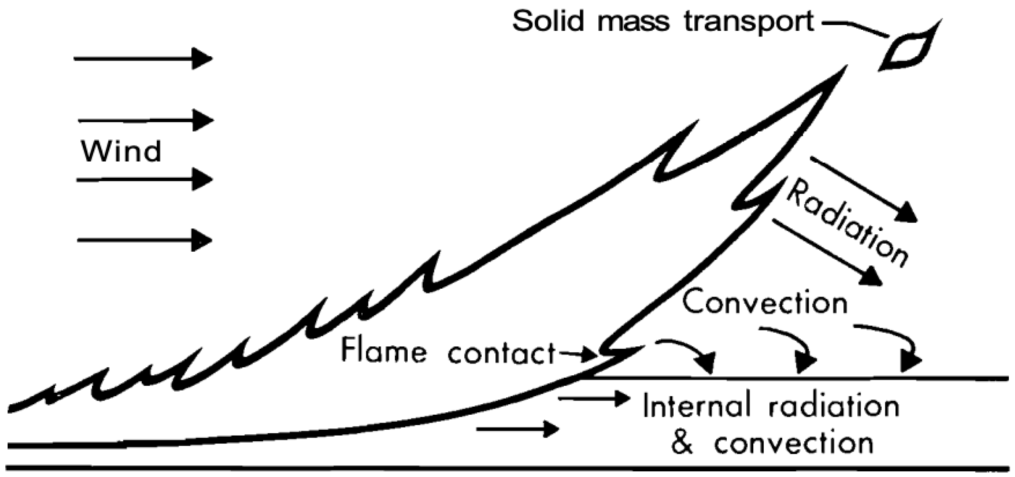
Engineer, California Forest and Range Experiment Station,[3] Forest Service, United States Department of Agriculture

INTRODUCTION

On the national forests there has been a trend during recent years toward placing on a systematic basis such fire-control practices as rating fire danger, determining the proper size of suppression crews and speed of attack, and planning fire-suppression strategy. This trend has resulted in greatly improving the techniques of planning and managing the fire-control organization. At the same time it has revealed a serious lack of essential information. The outstanding need is for specific data on the rate at which forest fires may be expected to spread under various conditions of forest cover, weather, and topography.

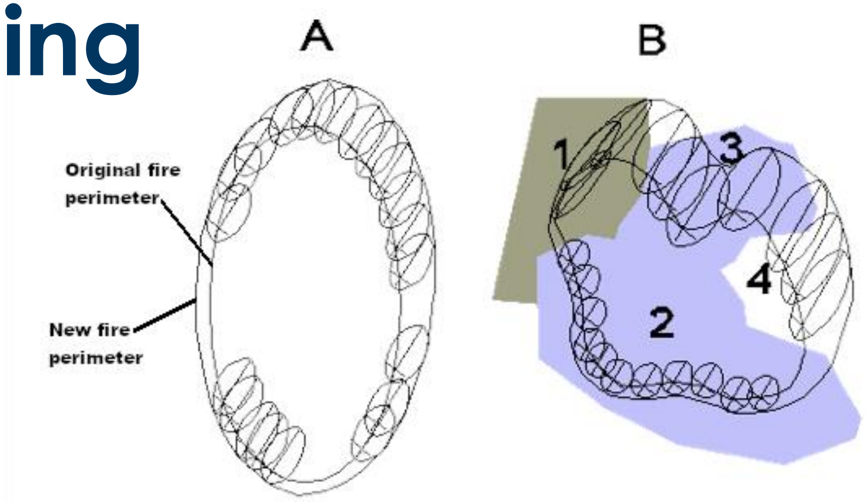
Wildfire modeling began in earnest after WWII, with the work of Fons (left, 1946).

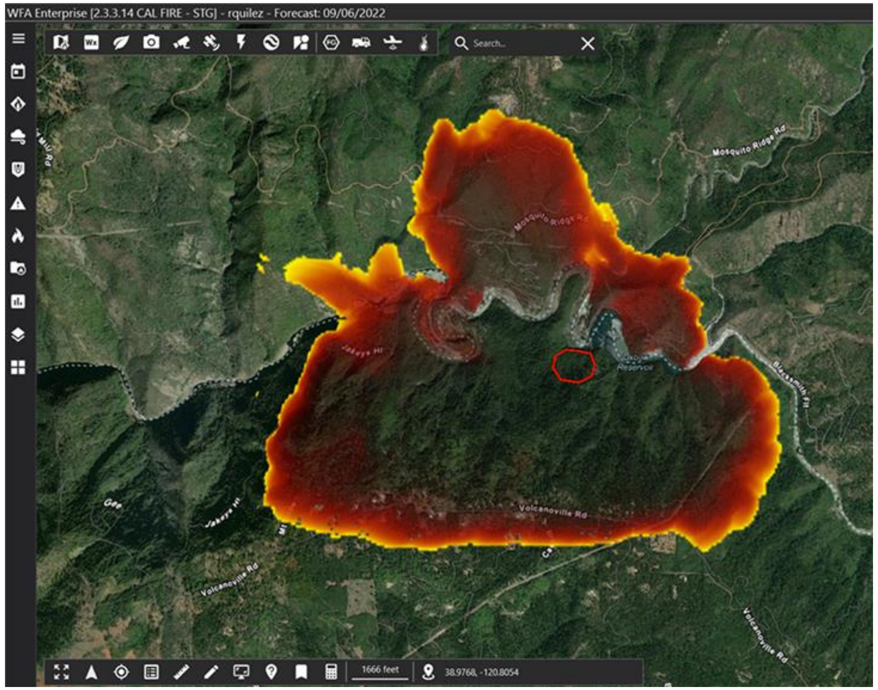
Rothermel (1972) released a simple semi-emprirical model of the rate of spread of a fire: heat in divided by heat out. This model is still in use today! (image below from Rothermel)



A short history of wildfire modeling

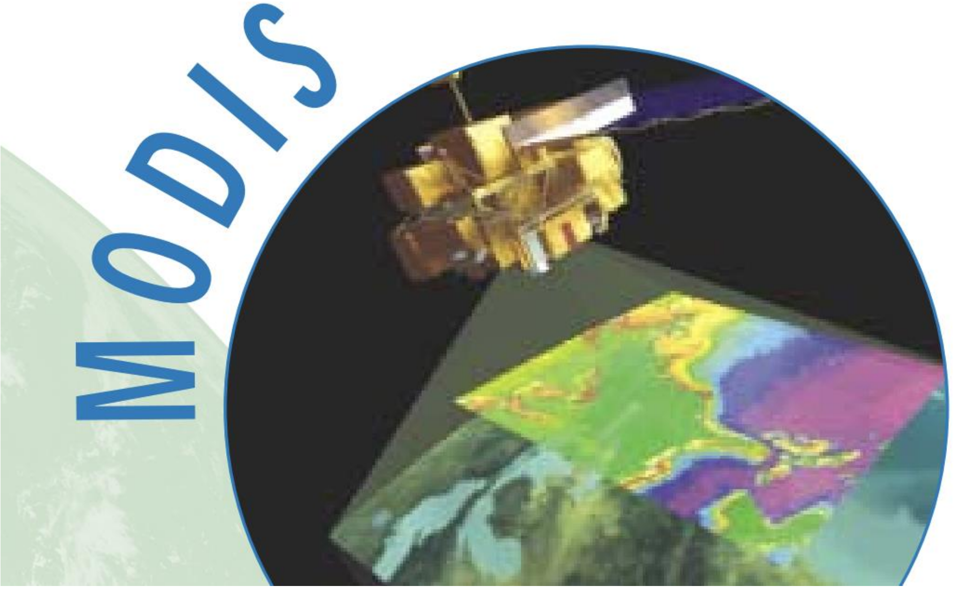
- Modern fire models include adaptations of Rothermel to a gridded 2D landscape
  - HFire
  - FARSITE (example of ellipsoidal propagation from FARSITE shown to right)
- In the 1990s, researchers began to pair Rothermel-based models with computational fluid dynamics weather models
- Commercial vendors (e.g. Technosylva) offer fire simulation software used by agencies and, while proprietary, likely have much overlap with Rothermel-based models

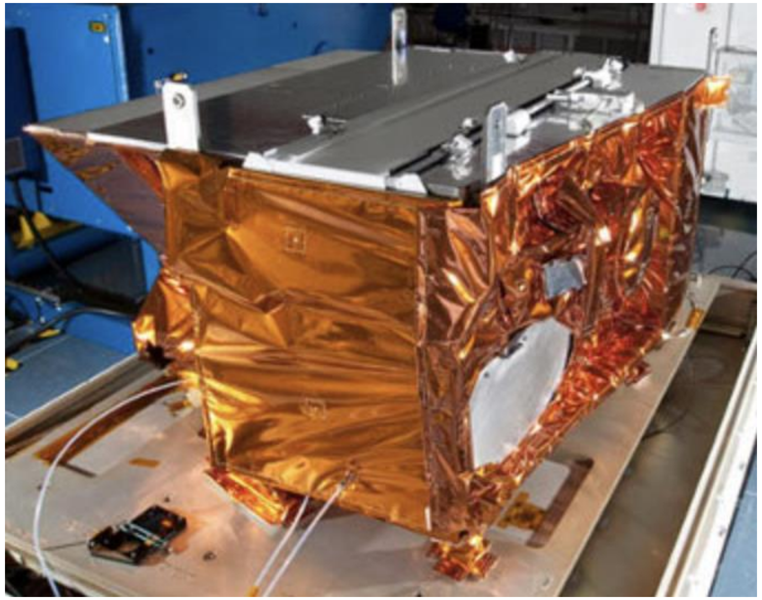




A new era of data

- NASA and other agencies maintain satellites which assist with fire detection and mapping
- The Moderate Resolution Imaging Spectroradiometer (MODIS), aboard Terra, measures EM radiation and covers most of Earth in 1-2 days
  - This includes fire detection!
- Since 2011, the Visible Infrared Imaging Radiometer Suite (VIIRS), aboard Suomi NPP (right), makes similar measurements, at higher resolution both spatially and temporally
- Remote sensing is used for wildfire mapping, but not for state-of-the-art prediction tasks.

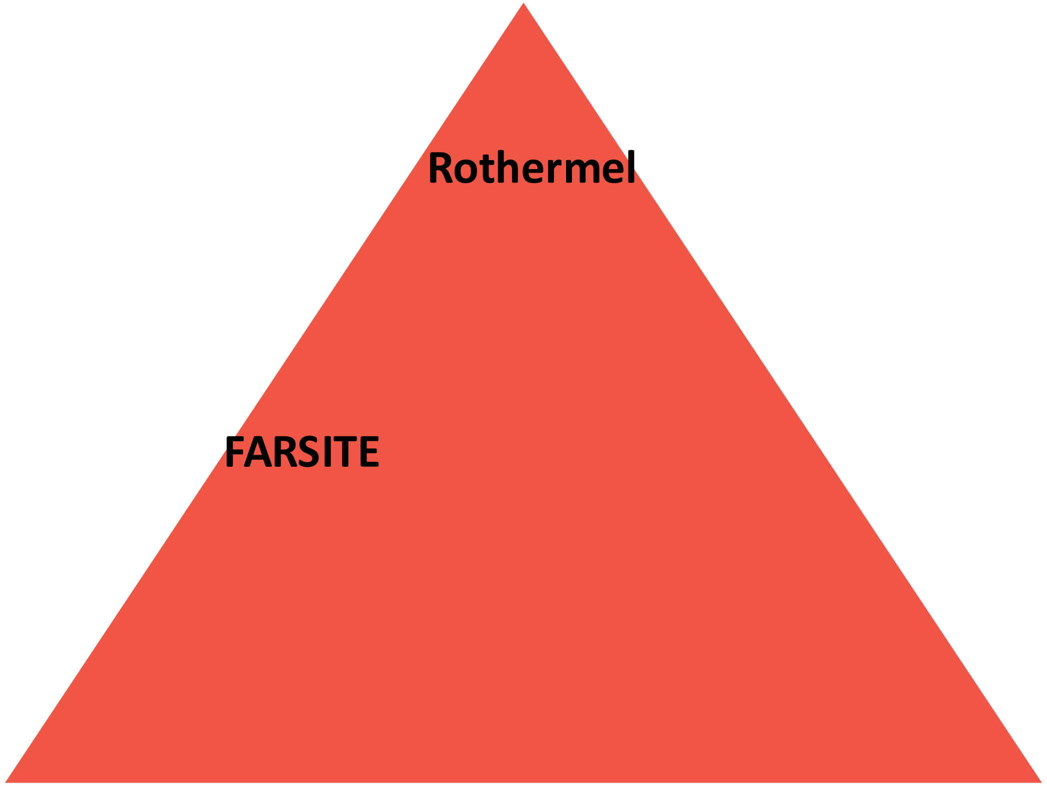




The fire (modeling) triangle

Semi-empirical models: uses fuel data, moisture data, but prescribed equations for rate of spread

We are interested in this corner!



Microscopic, physical models: computational fluid dynamics, transport equations

Data-driven models: learn dynamics statistically from **real fire data**

Next Day Wildfire Spread (NDWS)

The existing dataset and benchmark models

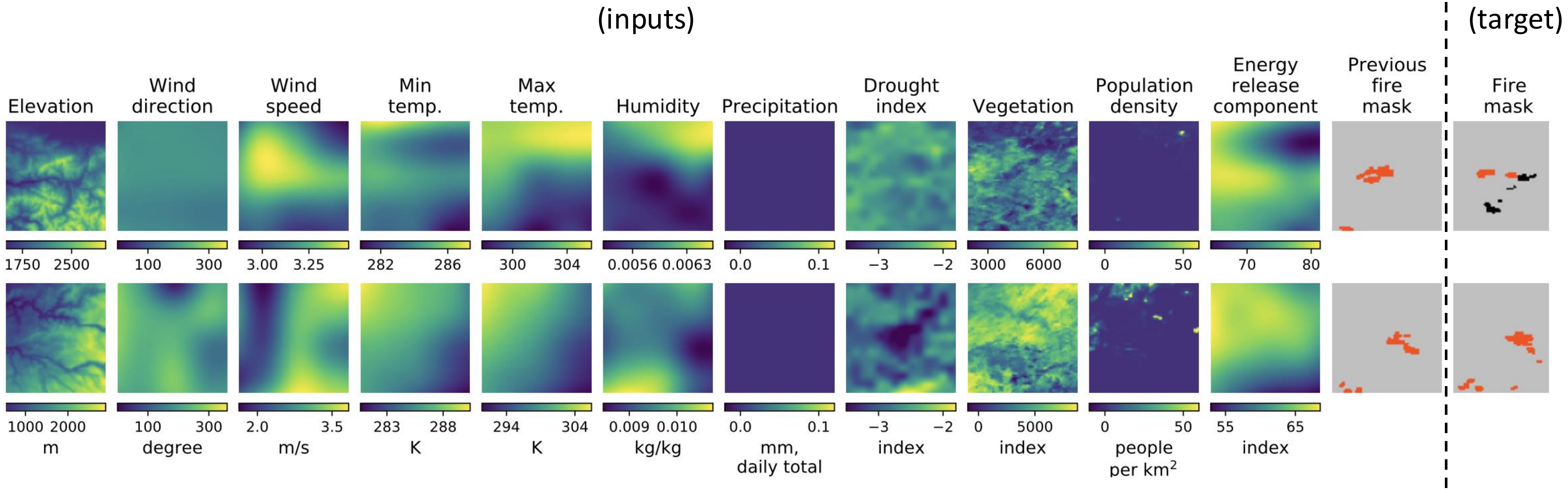
NDWS dataset (Huot et al)

Next Day Wildfire Spread:
A Machine Learning Data Set to Predict Wildfire
Spreading from Remote-Sensing Data

Fantine Huot[1], R. Lily Hu[2], Nita Goyal, Tharun Sankar, Matthias Ihme, and Yi-Fan Chen

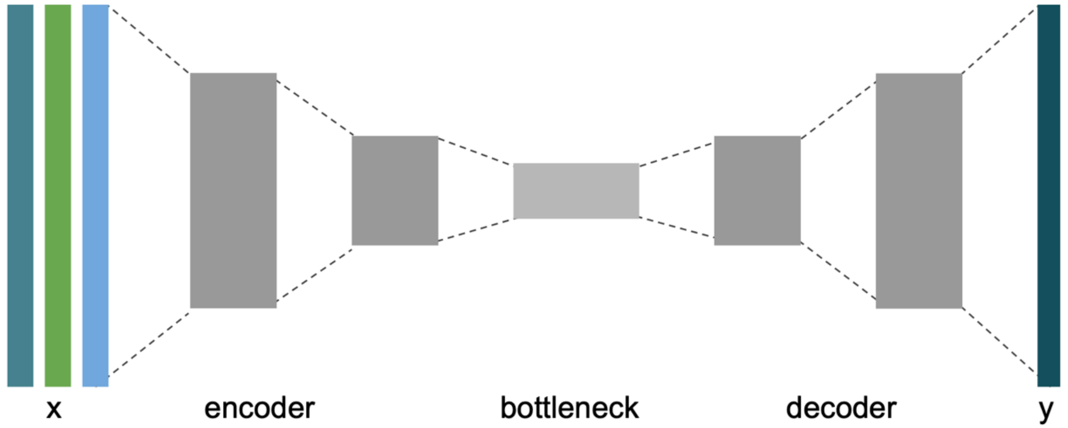
- Developed in 2022 by Google Research
- Overview: satellite fire detections & meteorological data on day *D* are used to predict fire detections in the same area on day *D + 1*
- Fire detections are from MODIS mission (1 km spatial resolution)
- Other features:
  - Environmental: elevation, drought index, veg index, population, ERC
  - Meteorological: wind speed & direction, max/min temp, humidity, precipitation

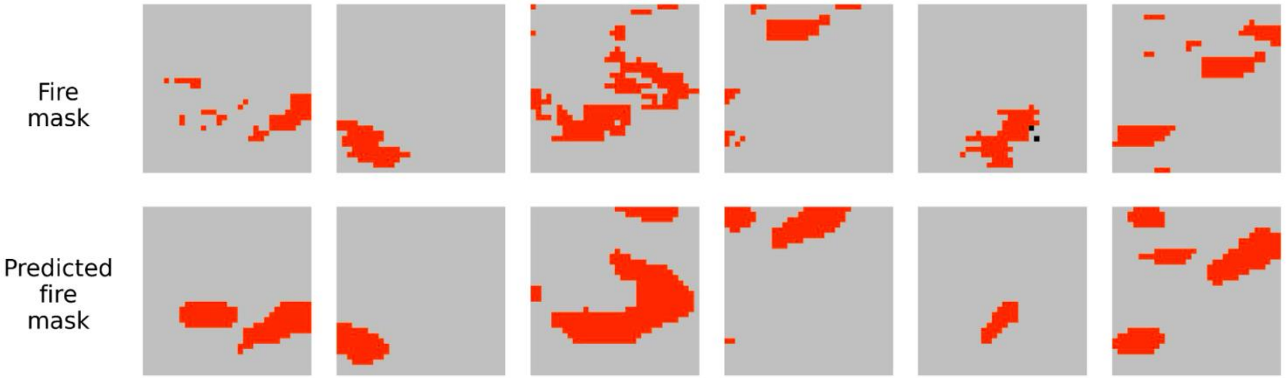
Example dataset entries from Huot et al



F. Huot, R. L. Hu, N. Goyal, T. Sankar, M. Ihme, and Y.-F. Chen, “Next Day Wildfire Spread: A Machine Learning Data Set to Predict Wildfire Spreading from Remote-Sensing Data”, arXiv preprint, 2021.

NDWS model benchmark (Huot et al)



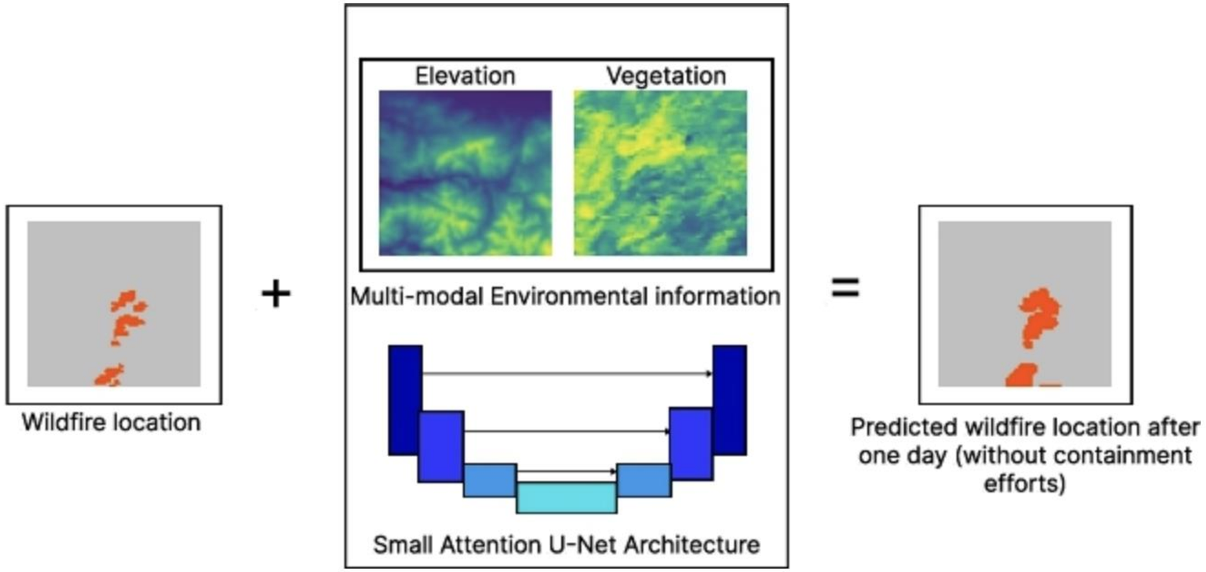


F. Huot, R. L. Hu, N. Goyal, T. Sankar, M. Ihme, and Y.-F. Chen, “Next Day Wildfire Spread: A Machine Learning Data Set to Predict Wildfire Spreading from Remote-Sensing Data”, arXiv preprint, 2021.

- Huot et al use a convolutional autoencoder (left) to predict the fire mask
  - Example predictions shown left, bottom
- The baseline in this problem is **persistence**: predicting the fire mask to be identical on days *D, D+1*
- Using the precision-recall curve, Huot et al achieve higher performance than the persistence baseline.

Attention-based models: Fitzgerald et al (2023)

- Fitzgerald et al use a U-net architecture with residual connections and attention blocks, trained on the same dataset
- Evaluation using the F1 score shows good performance but does not supersede Huot et al
- Ablation studies and hyperparameter experimentation are performed
- A different loss function (the focal loss) is used
- **Persistence baseline is hard to beat!**



Jack Fitzgerald, Ethan Seefried, James E Yost, Sangmi Pallickara, and Nathaniel Blanchard. 2023. Paying Attention to Wildfire: Using U-Net with Attention Blocks on Multimodal Data for Next Day Prediction. In ICMI '23, October 09--13, 2023, Paris, France. ACM, New York, NY, USA

Issues with the current NDWS dataset

- Relatively low spatial resolution (1 km)
  - Many features are available at much higher resolution, but MODIS is limited to 1km
- Meteorological features are aggregated as daily means
  - Some traditional fire behavior features are not present
- No data on fuel types, including impervious surfaces, are included
- Other fire behavior and topographical indices exist for inclusion
- No masking for controlled burns or non-wildfire thermal anomalies is explicitly performed

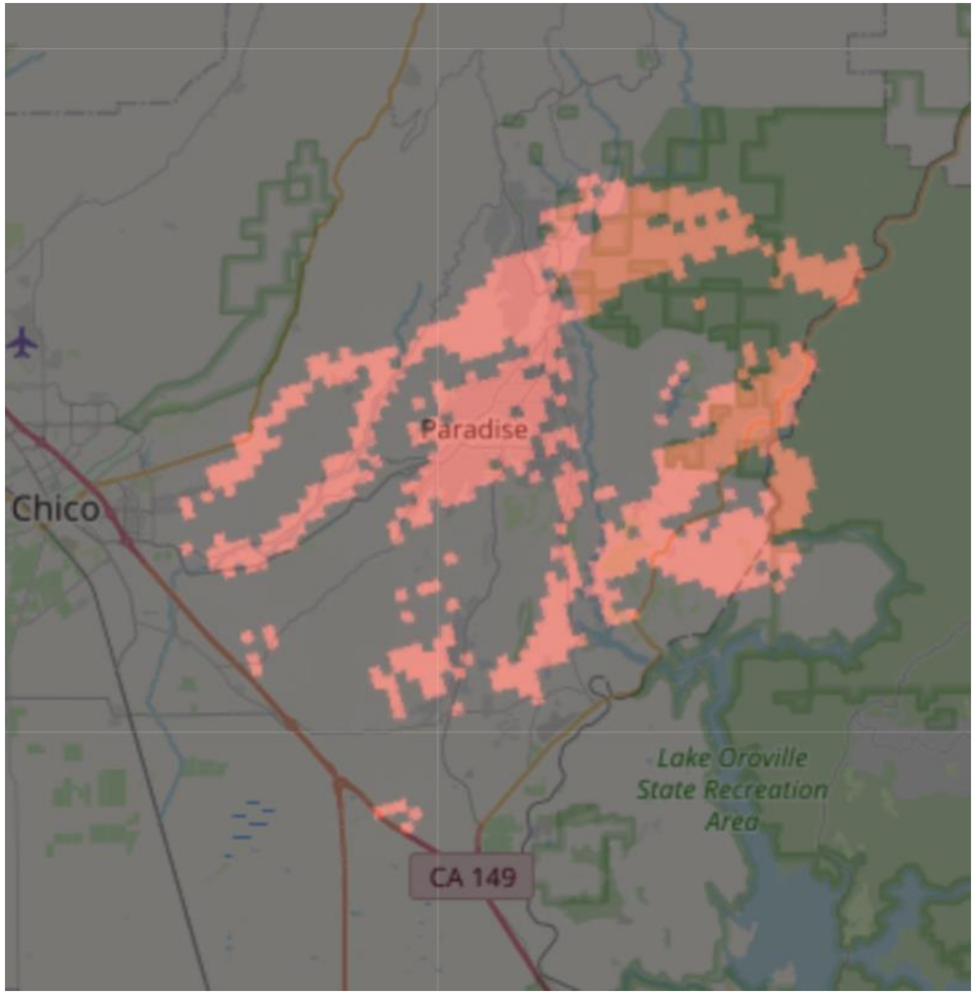
Goal: improve the NDWS dataset

- Draw fire detection data from the VIIRS mission, create a dataset at 500m spatial resolution
- Include more topographical, vegetation indices, and data on impervious landcover
- Use a high temporal resolution weather dataset and create custom aggregations for temperature, wind data
- Mask data to focus on wildland fire detections above a minimum size
- Integrate fuel type classifications from LANDFIRE

A modified NDWS dataset

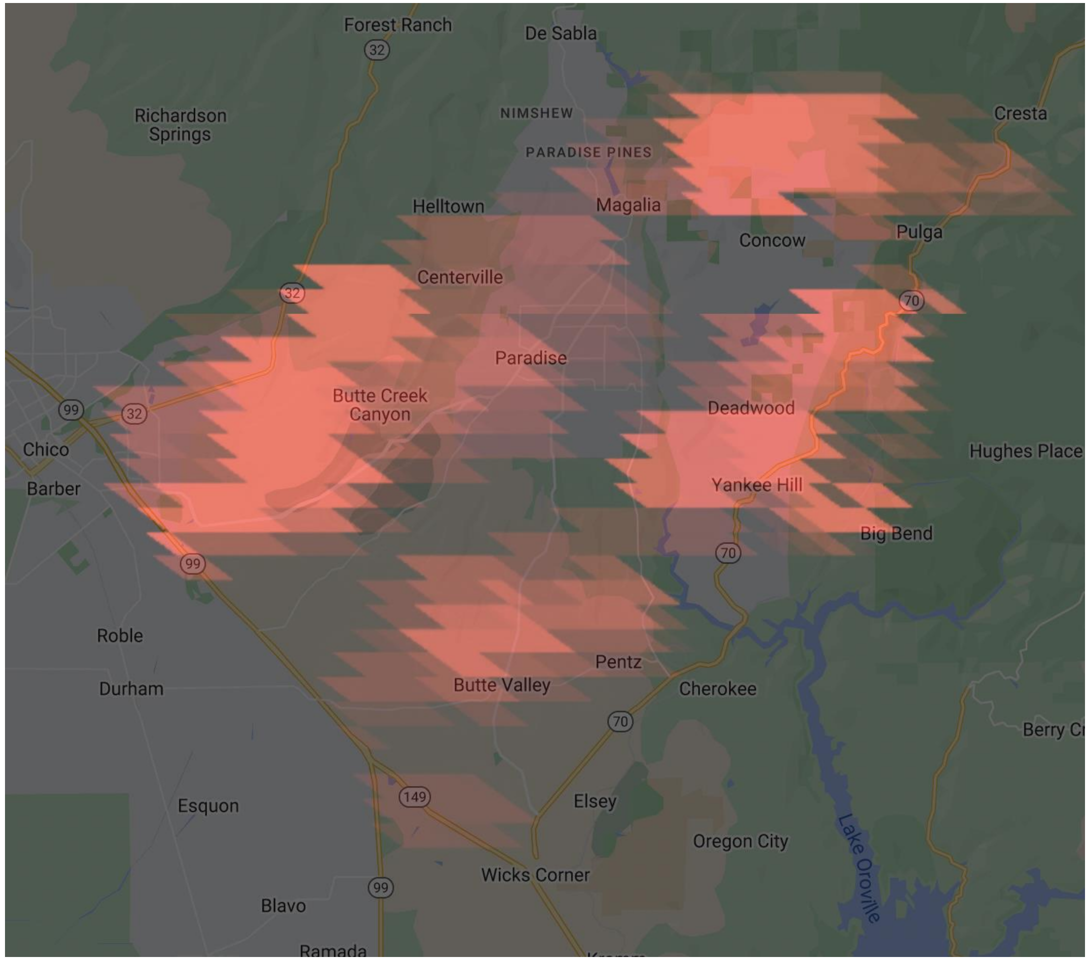
Modified NDWS

- Using the codebase developed by Huot et al, we create a pipeline to generate a new, higher resolution NDWS dataset with more features
- Fuel data is integrated using the Scott & Burgan classification in addition to an auxiliary vector embedding procedure
- VIIRS detections are used to obtain higher-resolution fire masks
  - VIIRS data is given a confidence score (poor, nominal, high) in data cleaning, aiding in detection mapping
  - MODIS measures fire radiated power (FRP), which must be thresholded to determine a detection level
- Dataset is exported at 500m resolution

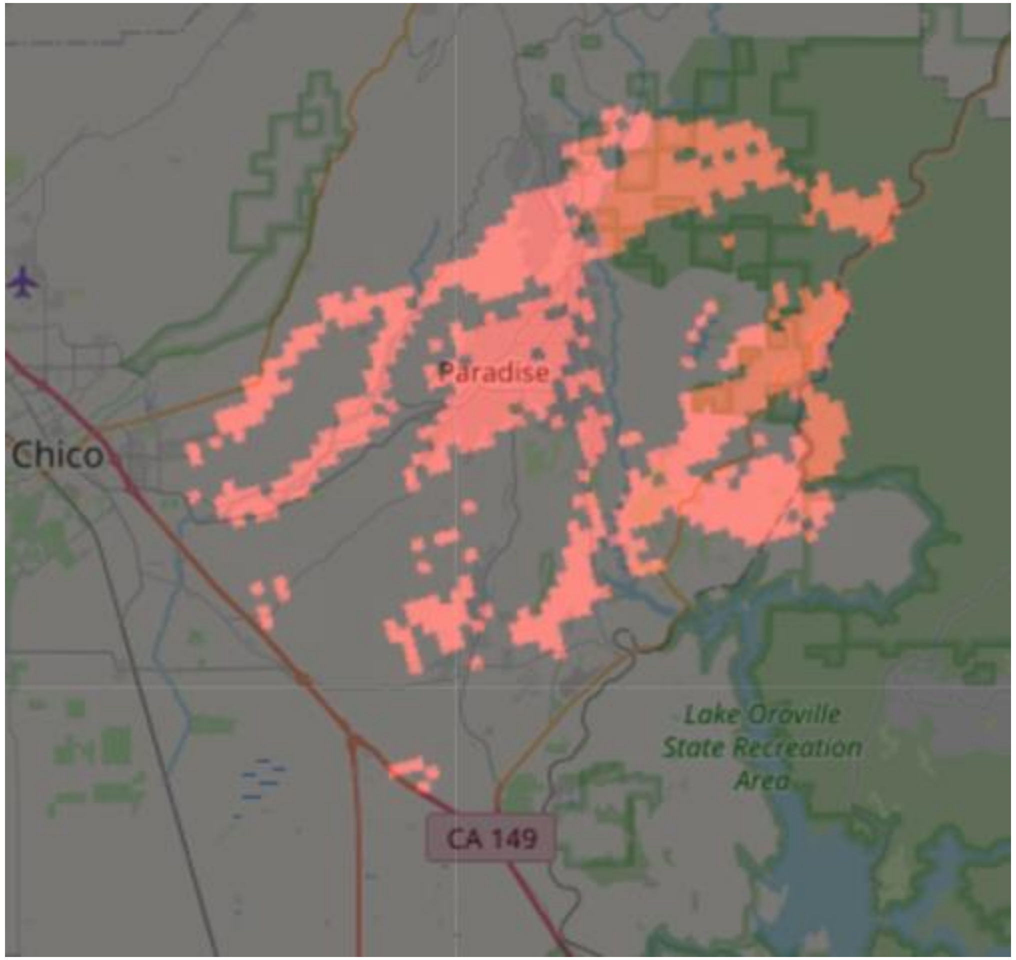


VIIRS detections of the Camp Fire on 11/09/2018, extracted using Google Earth Engine.

VIIRS vs. MODIS Thermal Anomaly Detection



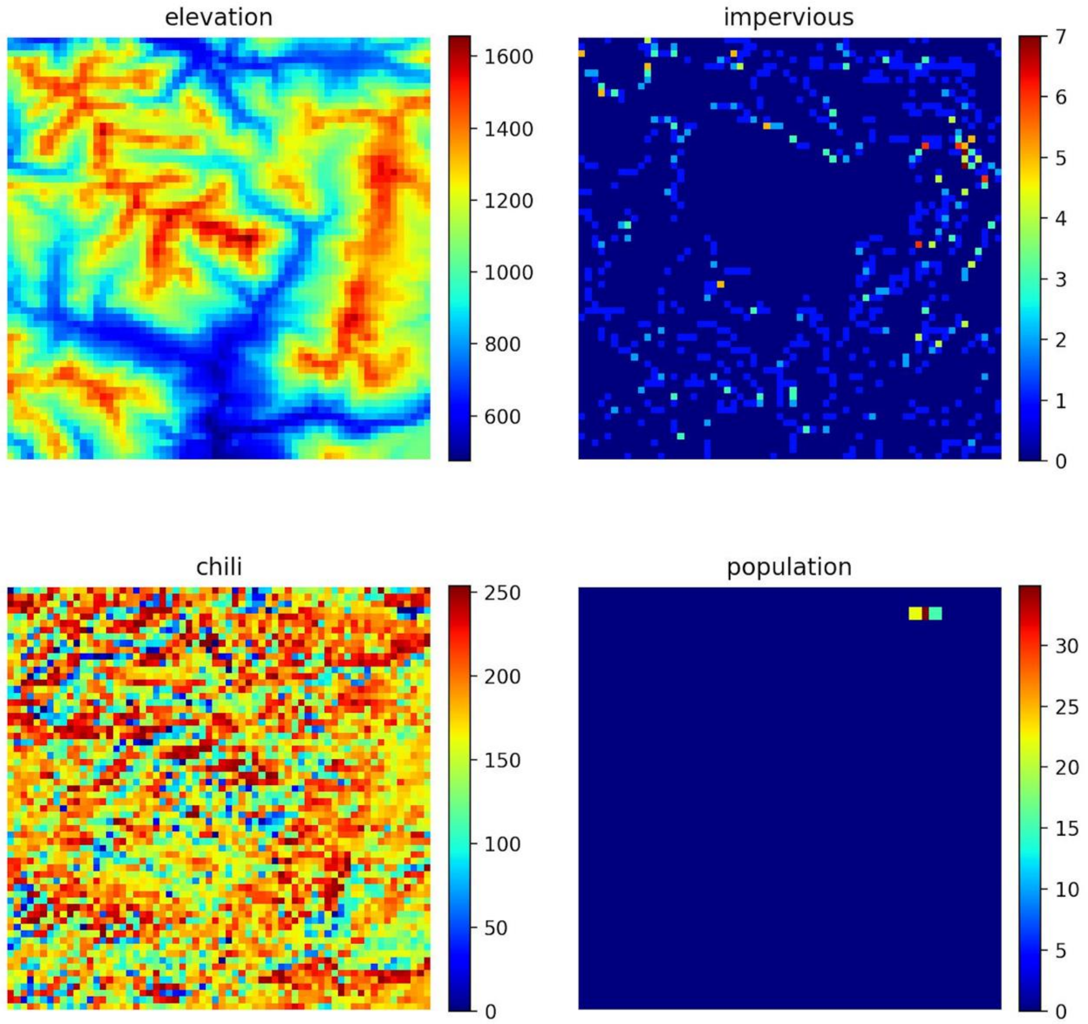
MODIS MOD14A1.061 dataset (Camp Fire, 11/09/24)



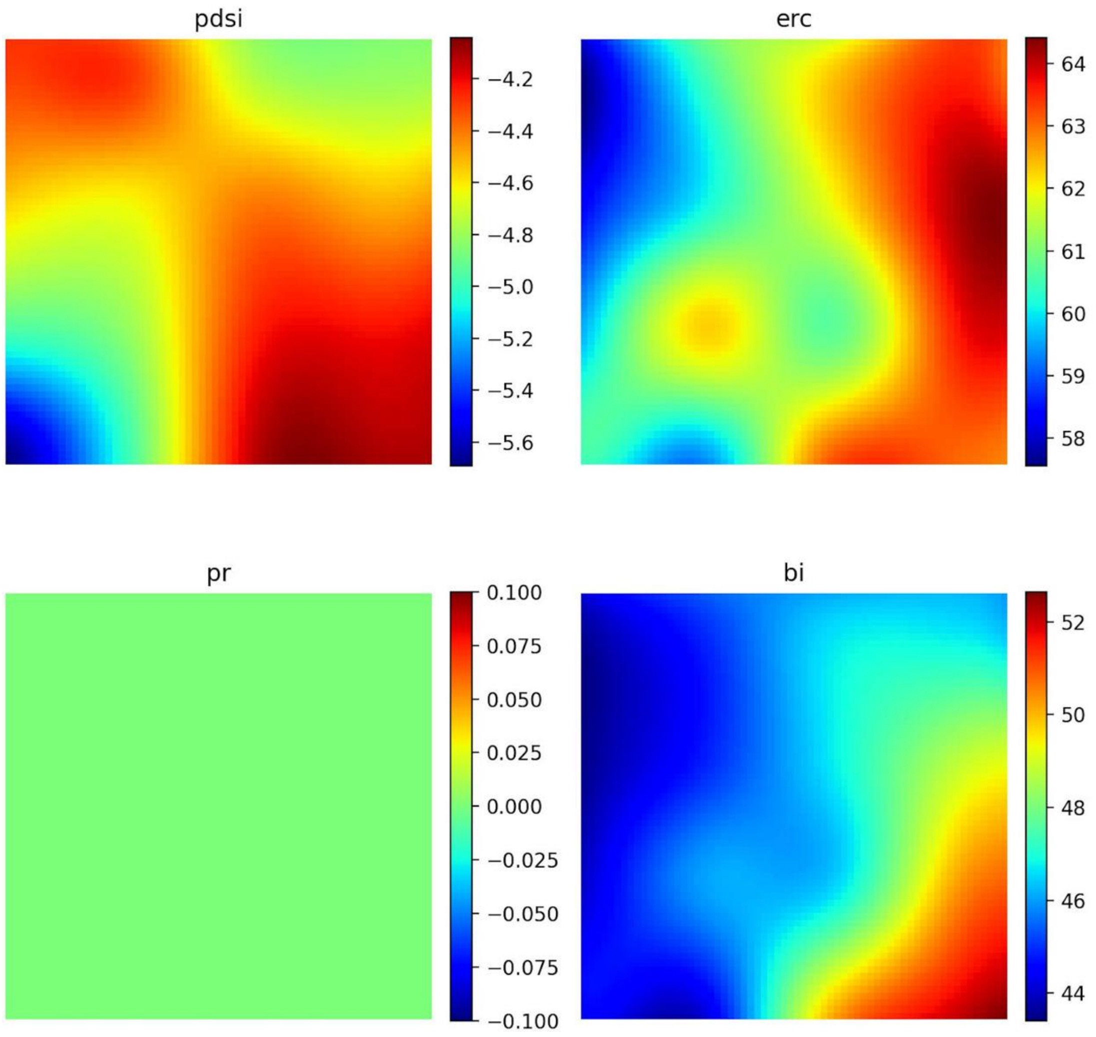
VIIRS S-NPP dataset (Camp Fire, 11/09/24)

Features: topographical/environmental

- We include:
  - Elevation
  - Population
  - **Continuous heat isolation load index (CHILI)**
  - **Impervious surfaces**
- While elevation captures topography, CHILI includes aspect information
- Impervious surfaces (roads, urban development) may present obstacles to fire spread, **or correlate with suppression efforts**
- Bolded features are new introductions



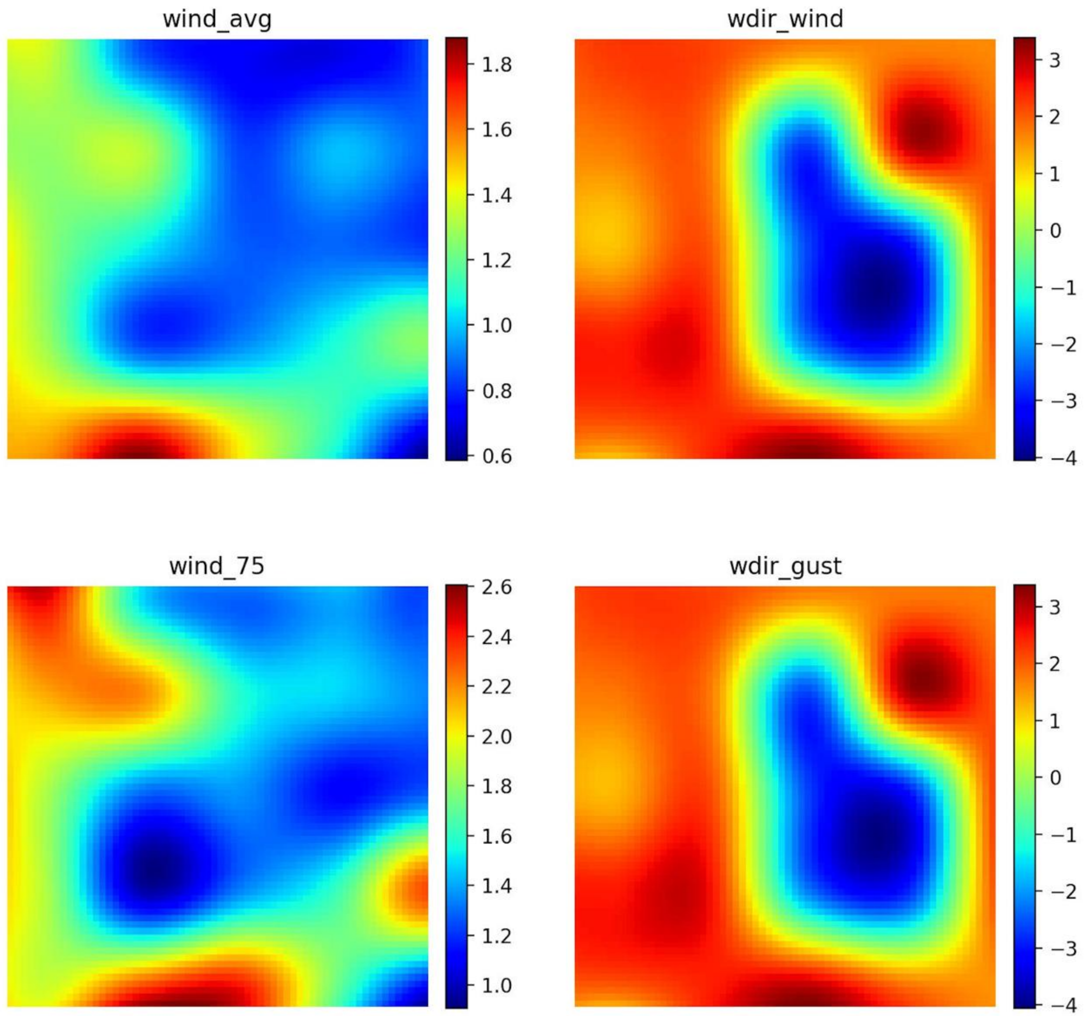
Features: vegetation dryness indices



- We include:
  - Palmer drought severity index (pdsi)
  - Energy release component (erc)
  - **Burn index** (bi)
  - Precipitation (pr)
- ERC and BI are traditional indices used in fire modeling, both for ignition risk and wildfire propagation
- Note: 100hr/1000hr dead fuel moistures are also available for sampling, but not presently included

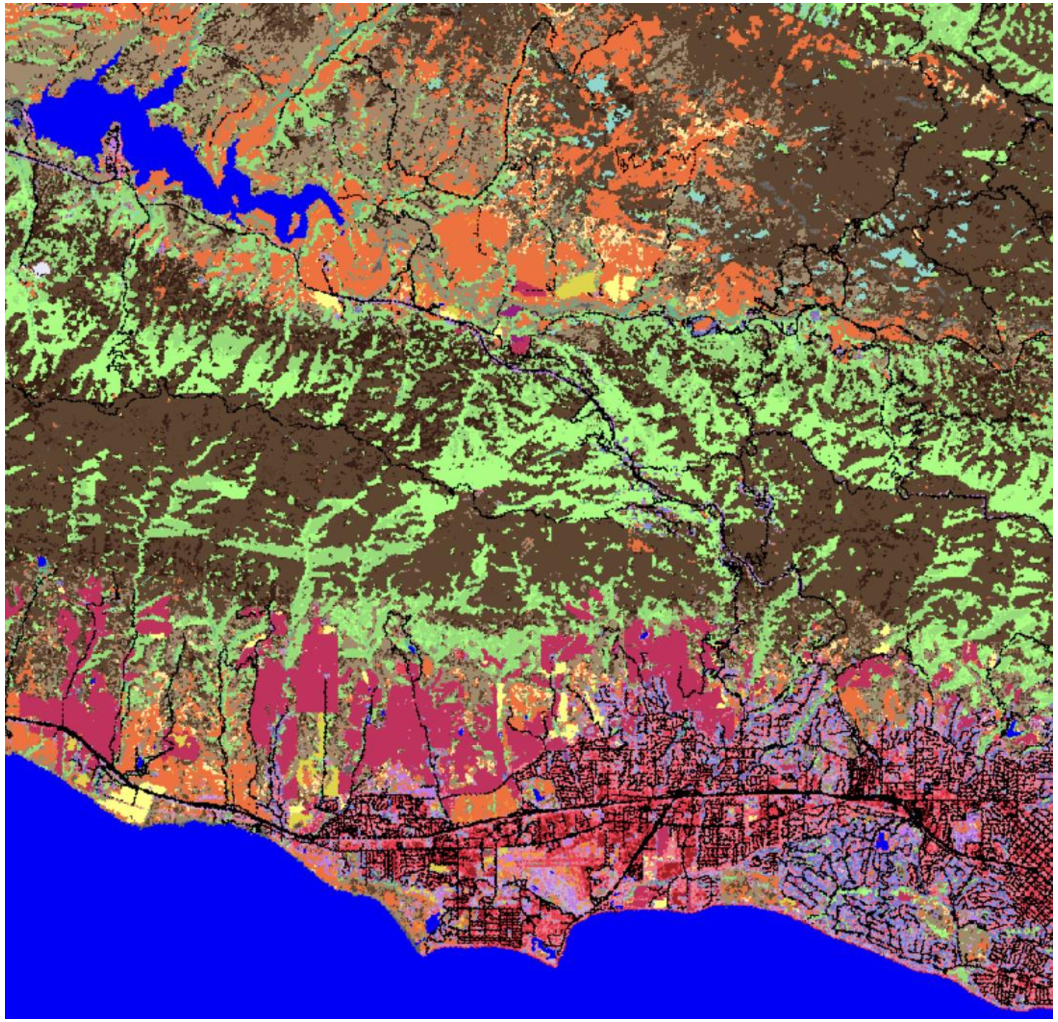
Features: wind speed and direction

- Instead of using daily averages, we use the RTMA dataset with hourly data at 500m resolution
- We include both **wind gust** and wind speed
  - Aggregated by quantile and average
- Wind direction is aggregated as an average weighted by wind/gust speed
- Custom aggregations of hourly data allow more complex meteorological features to be integrated



Integrating fuel data

Standard fuel classifications



- Many types of fuel exist, and often fuel types contain many categories
- Fire behavior modelers have developed some standard categorizations:
  - 13 Anderson Fire Behavior fuel models (FBFM)
  - 40 Scott & Burgan FBFM
  - Canadian Forest Fire Fuel Danger rating system
- LANDFIRE provides access to these classifications (at 30m resolution)
  - Left: Scott & Burgan fuel models around UCSB

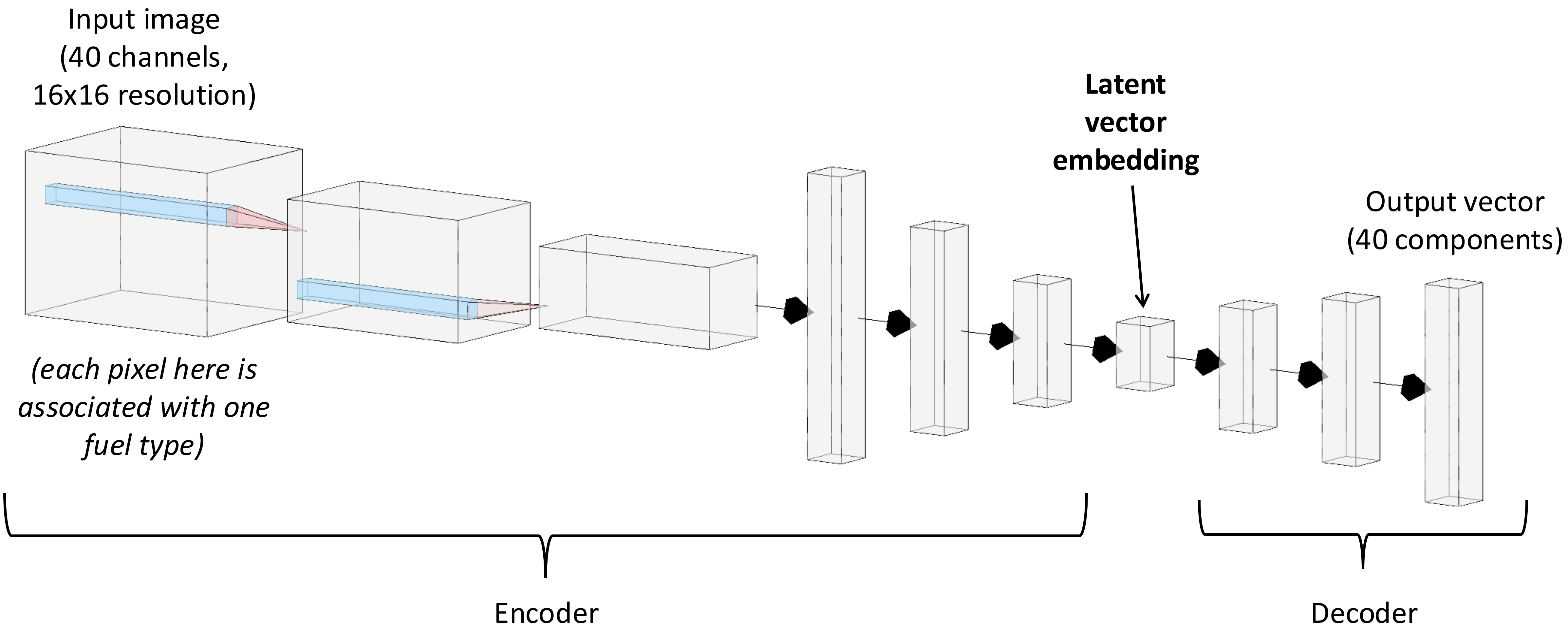
Issue: categorical encoding for neural networks is hard

- Categorical features are notoriously difficult for neural networks
- Why: neural networks are continuous, differentiable models. Cannot ordinally encode categories.
- Normal approach: “one-hot” encoding (all categories are binarized into present/not present)
  - This results in the creation of as many features as there are categories!
- Experts usually collapse fuel categories to a smaller set based on domain knowledge
- Alternative approach: **learn a vector embedding of fuel categories.**

Vector embedding of fuel categories

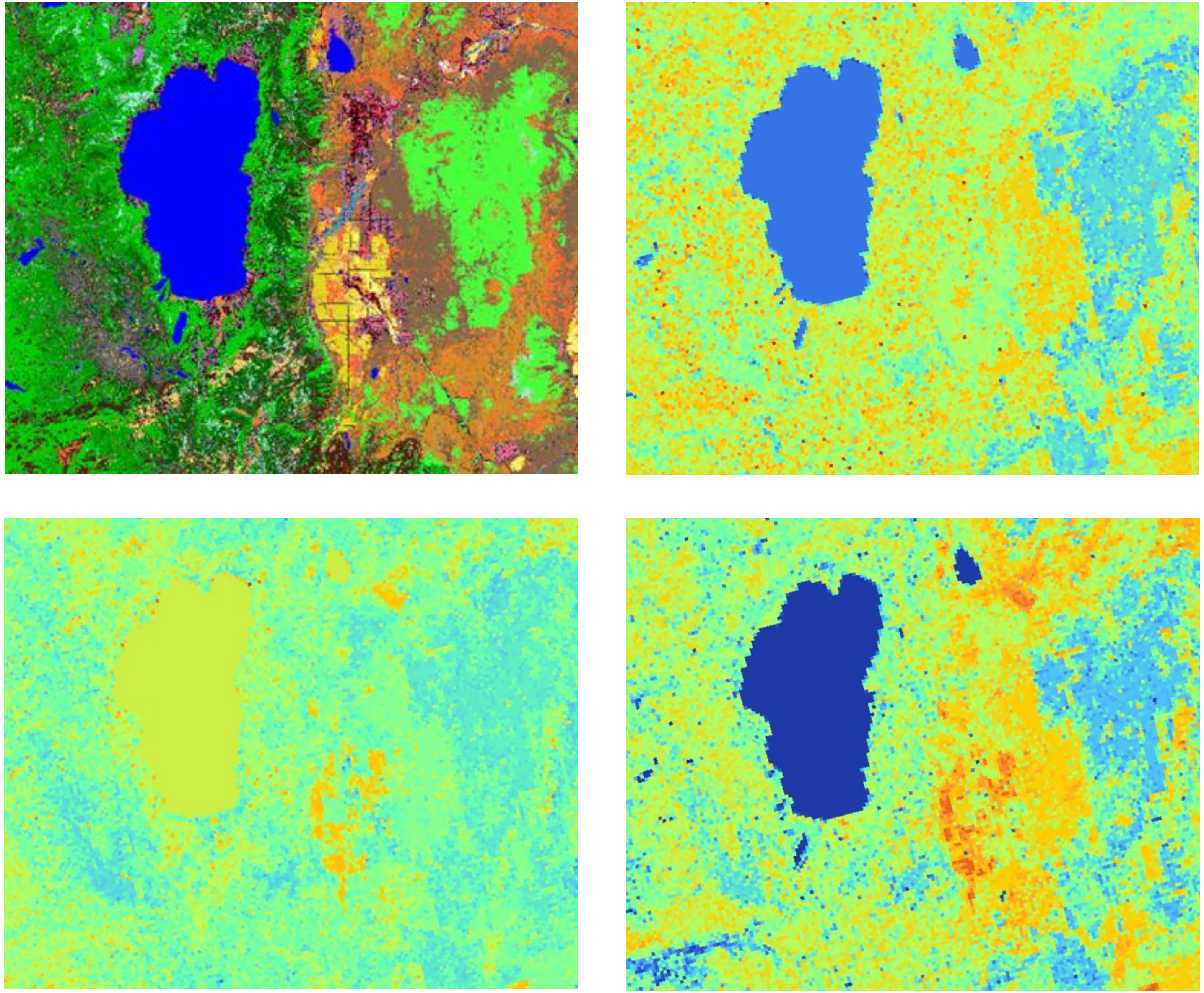
- We use the Scott & Burgan FBFM40 data
- For the new NDWS dataset, need to up-sample the LANDFIRE fuel maps from 30m to 500m resolution
  - Approximately a mapping from a 16x16 image to a vector of 40 category proportions
- Idea: use an autoencoder architecture
  - Encoder uses convolutional layers to featurize the 16x16 input, which is then mapped to a vector
  - Vector is embedded into a low-dimensional latent space (d = 3 here)
  - Decoder maps the latent vector to the vector of 40 category proportions

Architectural schematic: fuel autoencoder



Model training

- We train a model with conv layers {32,32,32}, max pooling, and a fully-connected autoencoder {32,16,**3**,8,16,32,40} where **3** is the latent vector embedding of interest
- We use a dataset of 20,000 random samples from CONUS of the FBFM40 fuel model maps from 2014
- After some experimentation, the best performing model is used to encode FBFM40 maps over CONUS into three-band images
- These images are uploaded to Earth Engine and sampled to form the new NDWS dataset.



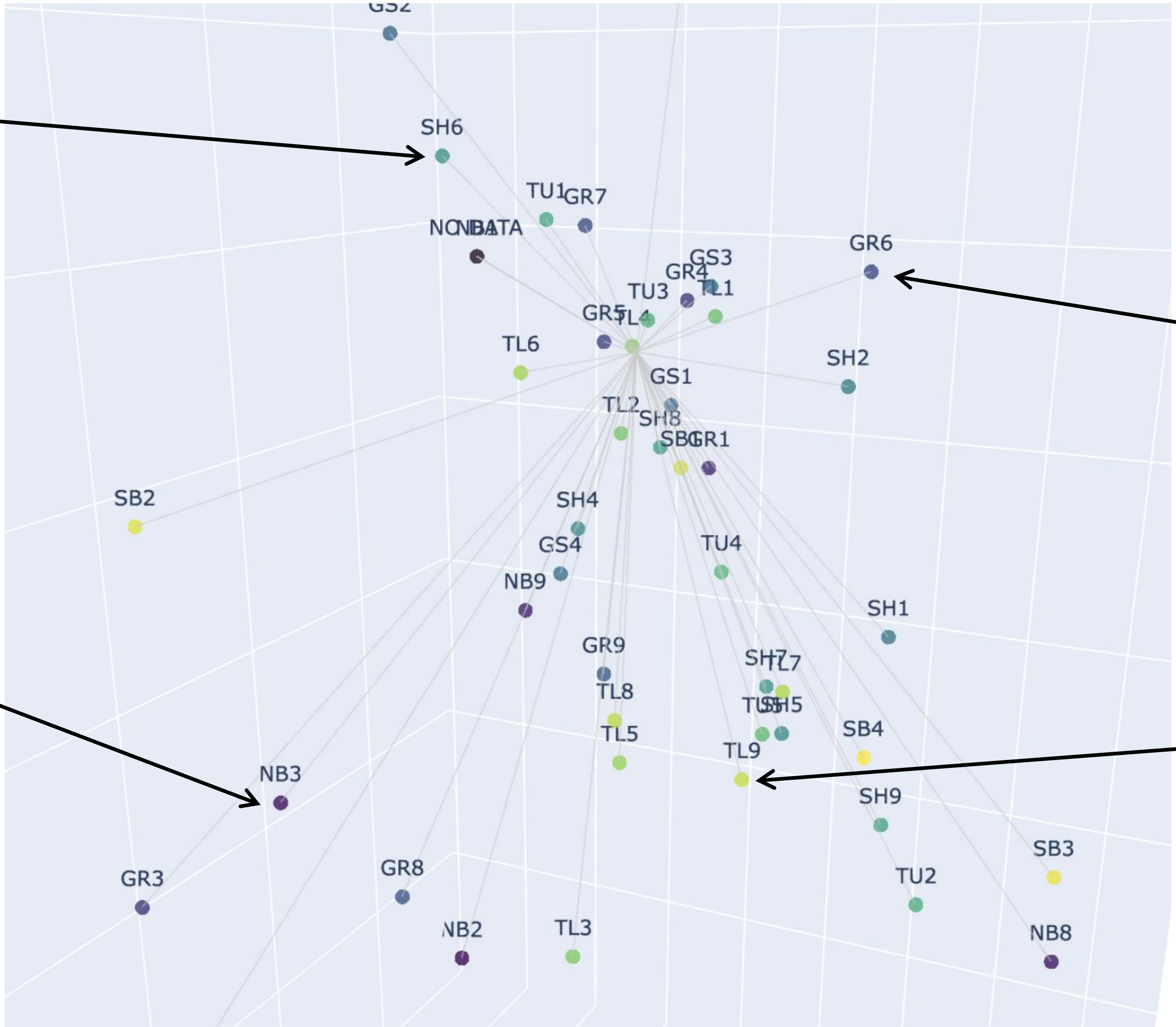
Example embedding

- Top left: FBFM40 fuel map for an area around Lake Tahoe (30m resolution)
- Other images: each component of the latent vector embedding of the fuel data, upscaled to the higher resolution (500m)

Visualization of the latent space

"Shrubs: dense shrubs, little or no herb fuel, depth about 2 feet. Spread rate high; flame length high."

"Grass: Dryland grass about 1 to 2 feet tall. Spread rate very high; flame length very high"

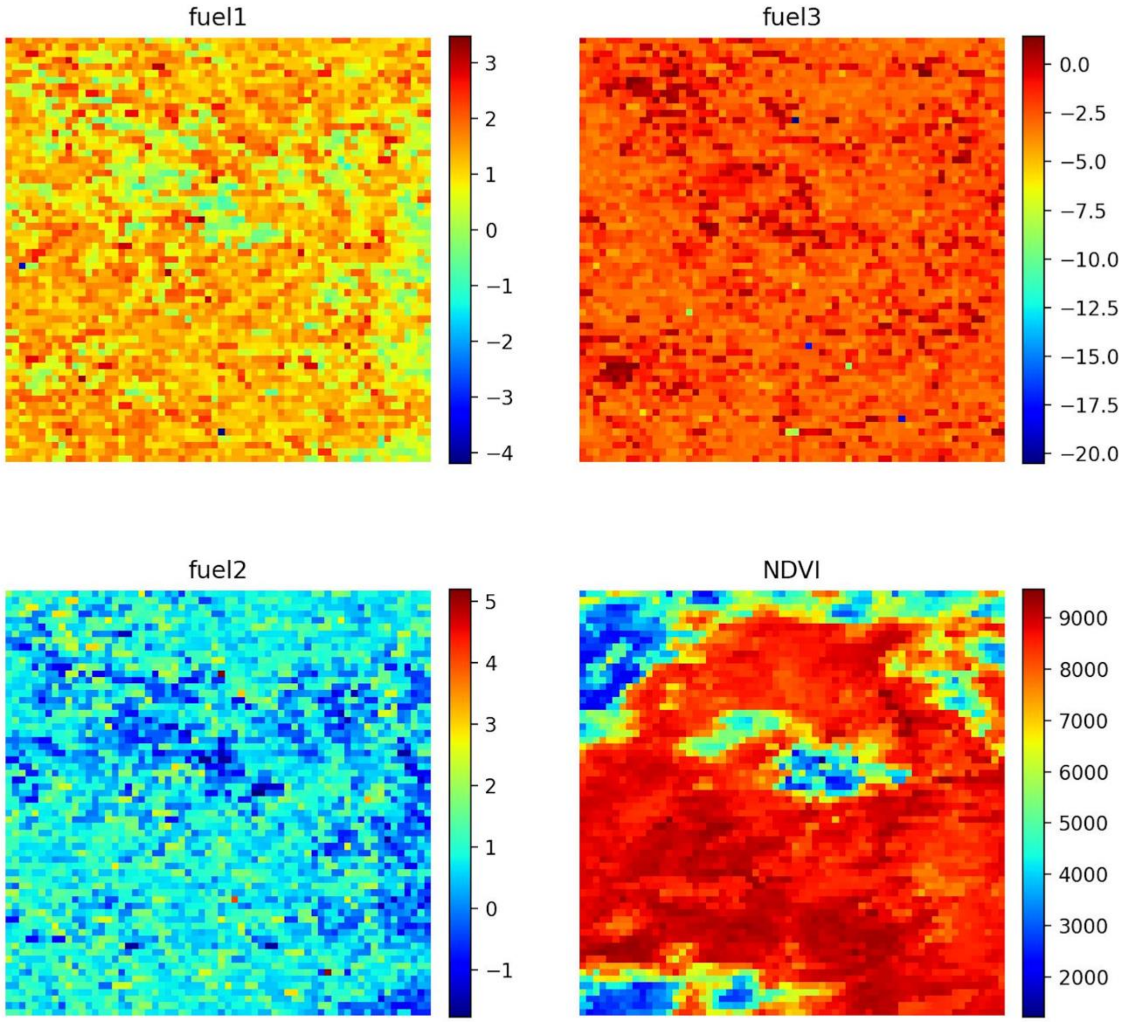


"Non-burnable: Agricultural field, maintained in nonburnable condition."

"Timber litter: very high load, fluffy. Spread rate moderate; flame length moderate"

Features: latent fuel variables

- After the fuel encoding process, the latent fuel variables are geospatially sampled with the rest of the features
- The NDVI (normalized difference vegetation index) contains information about fuel moisture/greeness
- The latent fuel variables **theoretically** have information about fuel types: the model must learn to extract this info effectively



Dataset preparation

- Focus on western CONUS region between 2018-2023
- All feature values clipped to reasonable ranges
- Feature mean, standard deviation computed over training set for standardization
- Instances with fewer than 5 detects or more than 25% impervious surface are removed.

- After cleaning, about 3% of pixels contain fire detections.
  - About the same proportion as in the Huot et al dataset

A main focus of this work is **creating an easy-to-use, robust codebase** to generate similar datasets over arbitrary regions and timeframes. The code is available on GitHub, with auxiliary data available in public Google cloud storage.

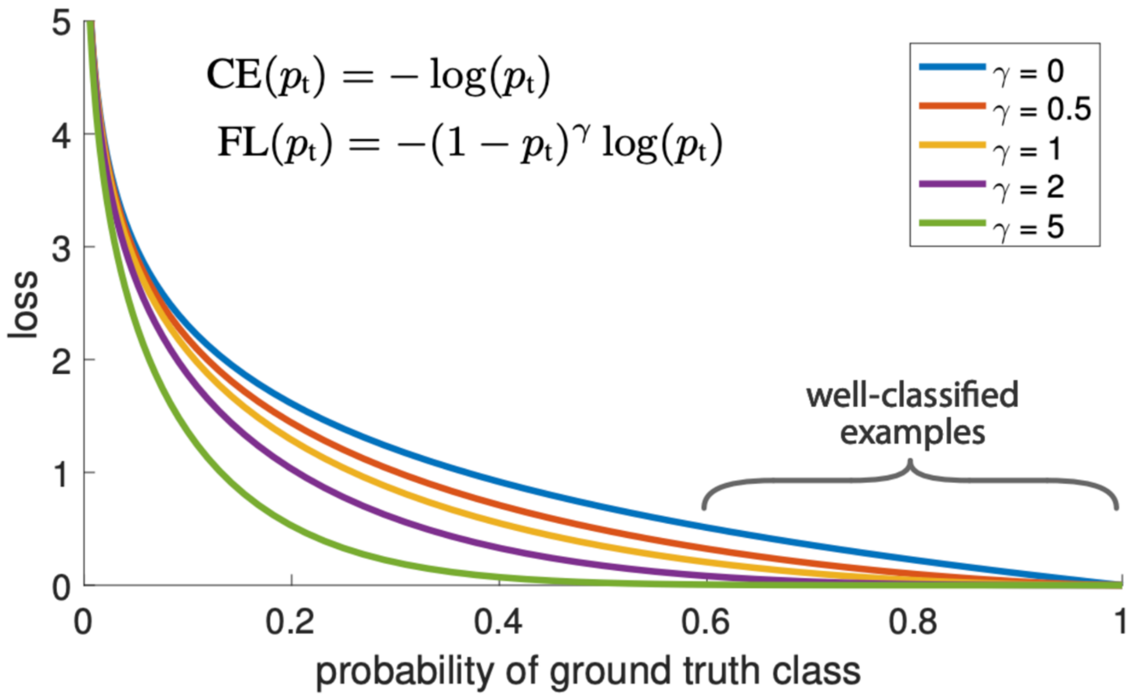
Results

Notes on model evaluation

- Class imbalance: use area under the precision-recall curve to evaluate (AUC-PR), F1 score
  - F1 score is the harmonic mean of *precision* (% of correct fire cells out of all fire cells) and *recall* (% of true fire cells predicted)
- Baseline is set by the **persistence** model: predict next day fire mask to be identical to given fire mask
  - On present dataset, this gives F1 = 0.399 (decently high!).
- Model training w/ Adam and various subsets of features, hyperparameters
  - Much more extensive experimentation & compute needed to find optimal hyperparameter combos
- Some ablation studies should be performed
  - Limited by compute power & time as of this presentation

Other notes on training

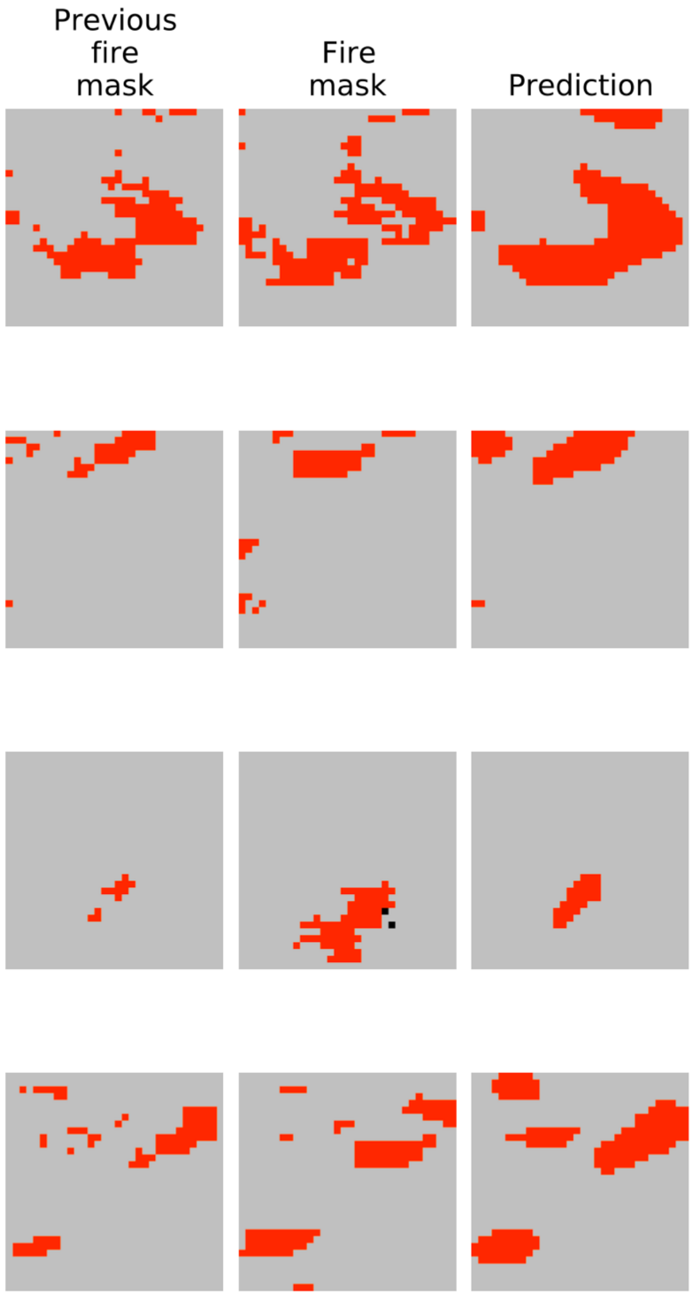
- Prediction target is binary segmentation: predict next day fire mask
  - Other options: 3-class prediction: predict non-burning pixels, predict pixels that go out, predict pixels that ignite
- Loss function: using weighted focal loss (right)
- BatchNorm to stabilize, Dropout to combat overfitting
- Group pooling creates invariant features
- L1 penalty can be applied to prediction masks to encourage sparsity
- Data augmentation: random crops, random rotation, random flips



Lin, T.-Y., Goyal, P., Girshick, R., He, K. & Dollár, P. Focal Loss for Dense Object Detection. Preprint at https://doi.org/10.48550/ARXIV.1708.02002 (2017).

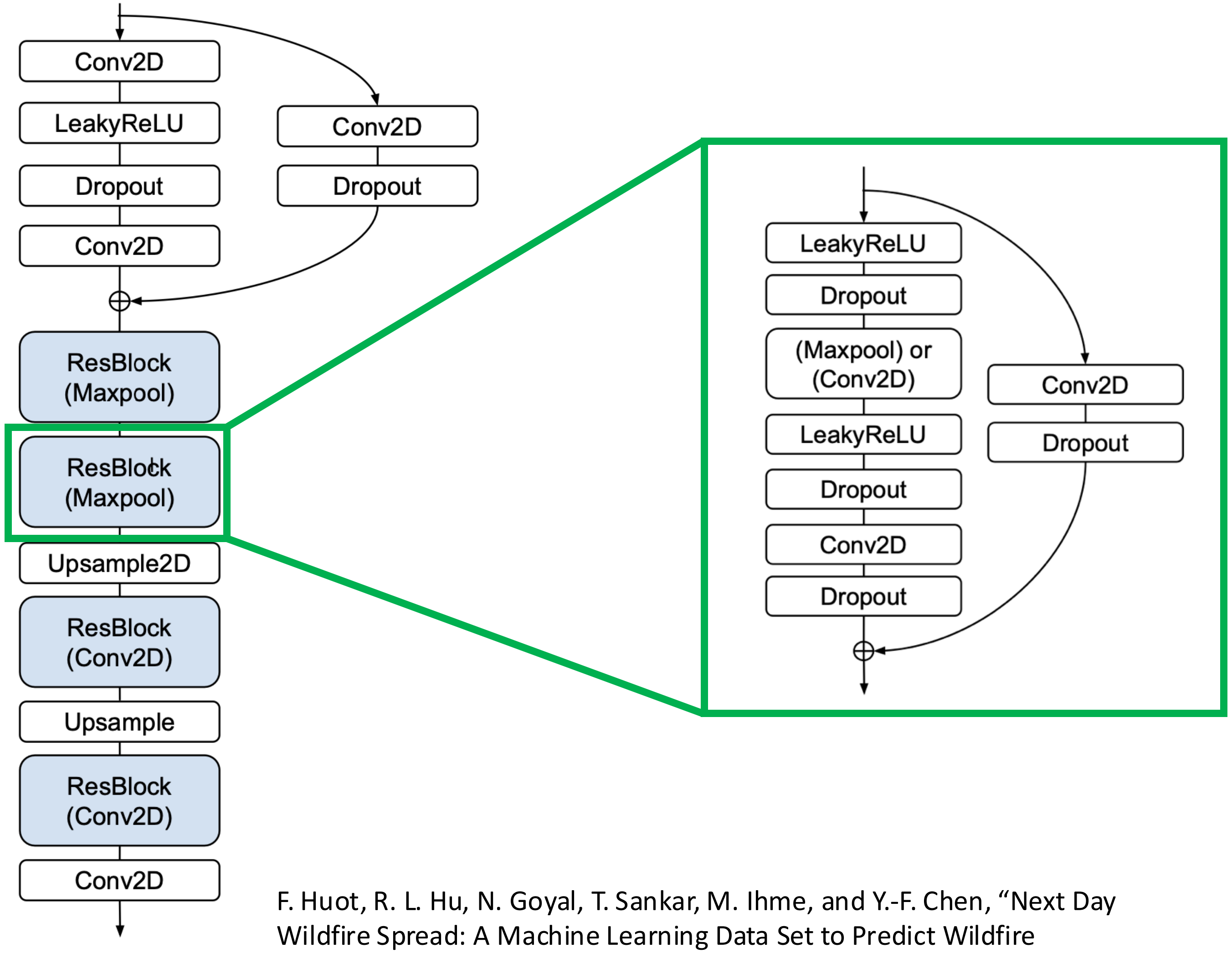
Huot et al model

- Original NDWS metrics:
  - Persistence F1 = 0.309
  - Best model F1 = 0.377
- F1 score is threshold-dependent: here, choose threshold that maximizes F1
- Huot et al model predictions shown to right
  - CNN predictions are generally smooth & consolidated
- Observation: model seems to roughly “fill out” the previous fire mask input



F. Huot, R. L. Hu, N. Goyal, T. Sankar, M. Ihme, and Y.-F. Chen, “Next Day Wildfire Spread: A Machine Learning Data Set to Predict Wildfire Spreading from Remote-Sensing Data”, arXiv preprint, 2021.

Huot et al architecture

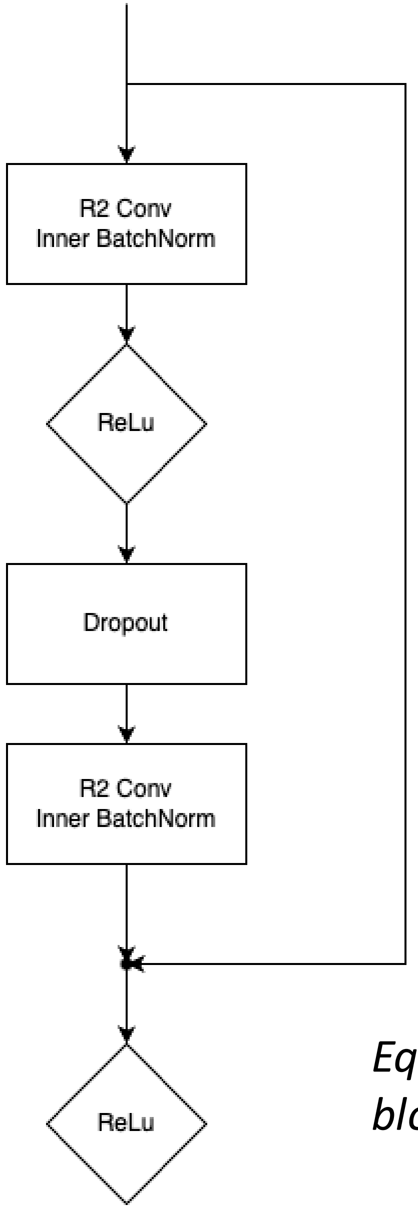


- Convolutional autoencoder
- Residual blocks form backbone of autoencoder structure
- Pooling in encoder, upsampling in decoder

F. Huot, R. L. Hu, N. Goyal, T. Sankar, M. Ihme, and Y.-F. Chen, "Next Day Wildfire Spread: A Machine Learning Data Set to Predict Wildfire Spreading from Remote-Sensing Data", arXiv preprint, 2021.

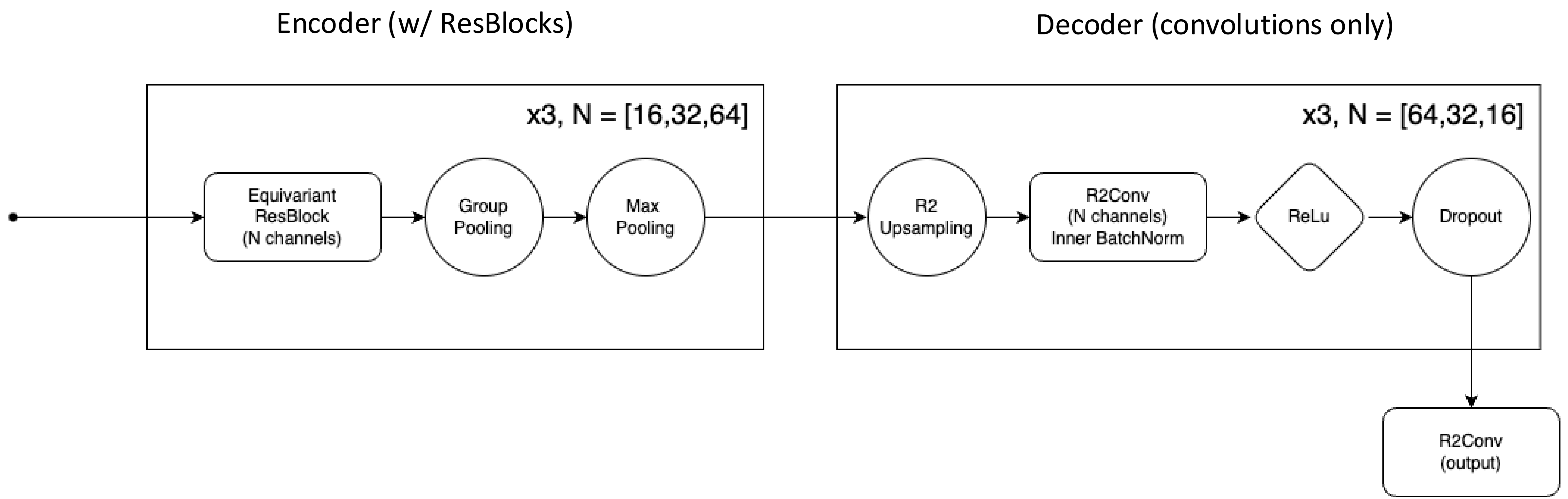
Architecture notes

- Maintain a simplified ResBlock autoencoder structure
- Upgrade ResBlocks to be **equivariant to rotations**
  - No inherent orientation to fire data: rotations of input should correspond to rotations of output
  - Data augmentation can help do this at the cost of more parameters and larger dataset
- Add attention/convolutional output blocks
- Maintain more direct residual connections throughout



Equivariant residual block diagram.

Initial architecture: equivariant autoencoder



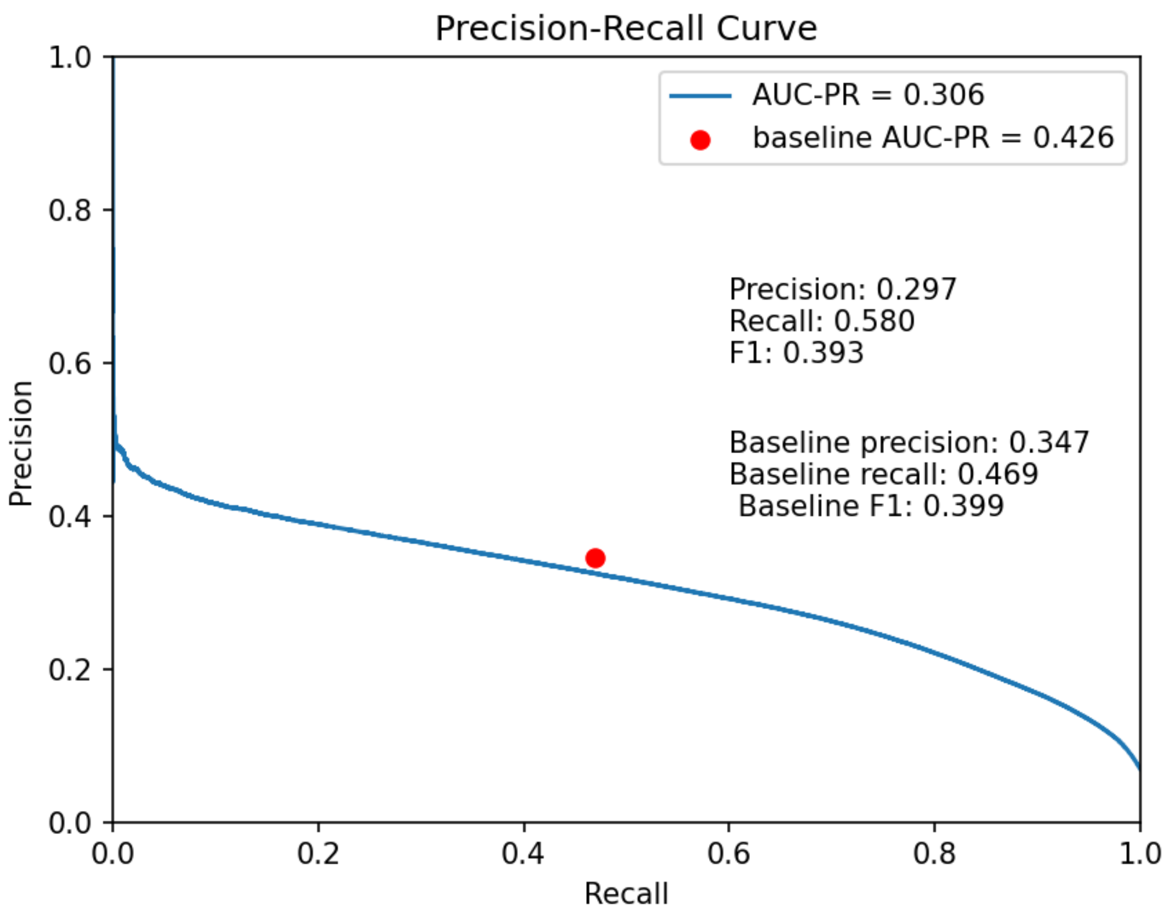
Experiment 1: feature selection with equivariant autoencoder

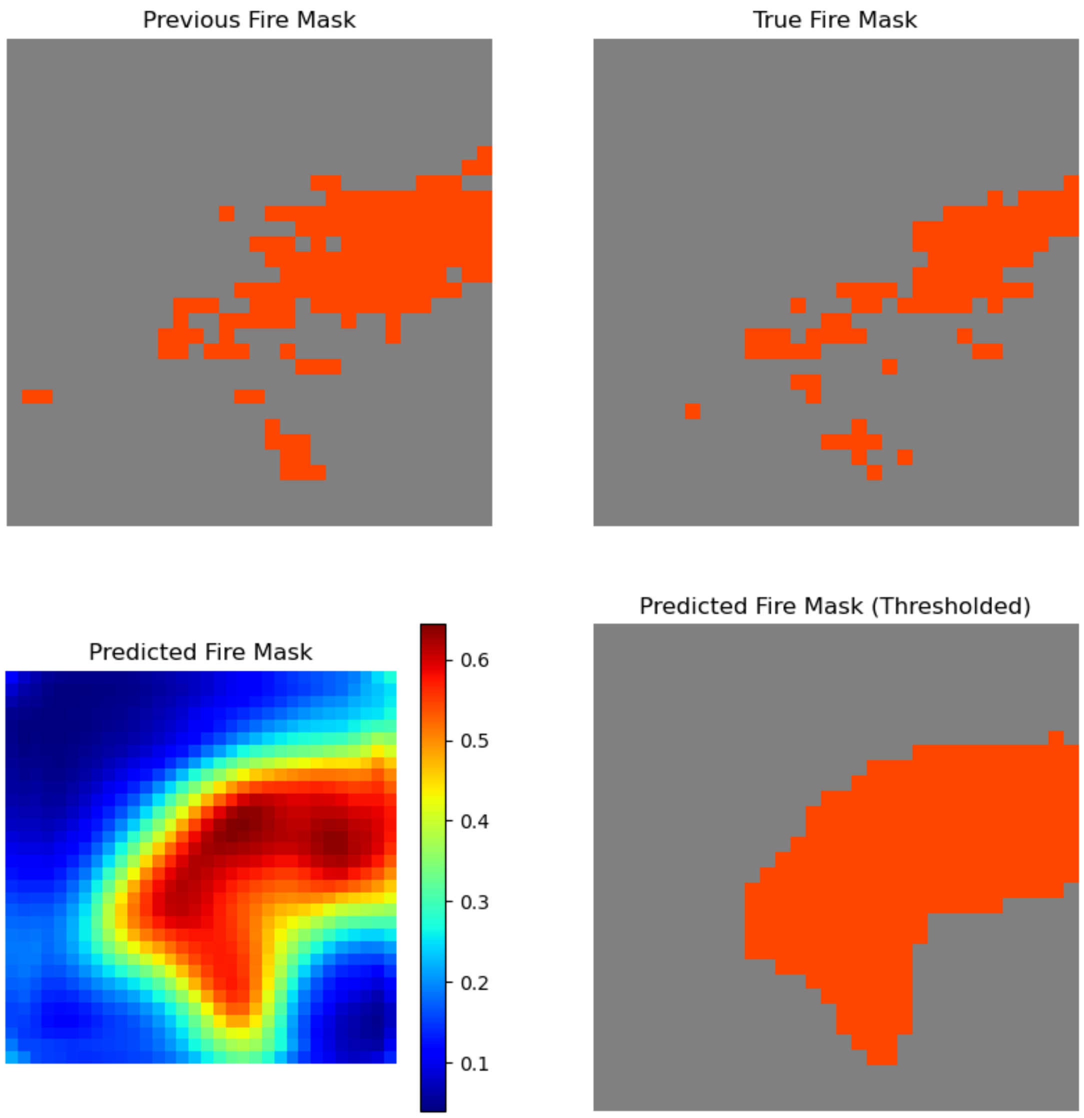
- Base features:
  - elevation
  - chili
  - impervious
  - population
  - NDVI
  - pdsi
  - erc
  - avg_sph
  - tmp_75
  - wind_avg
  - wdir_wind
  - viirs_PrevFireMask

(all models trained with focal loss $\gamma = 0.1$, for 25 epochs)

- Model A, 12 inputs:
  - (base features only)
- Model B, 15 inputs:
  - (base features)
  - Latent fuel variables: fuel1, fuel2, fuel3
- Model C, 18 inputs:
  - (base features)
  - Latent fuel variables: fuel1, fuel2, fuel3
  - Burn index, 75% percentile wind, weighted gust direction

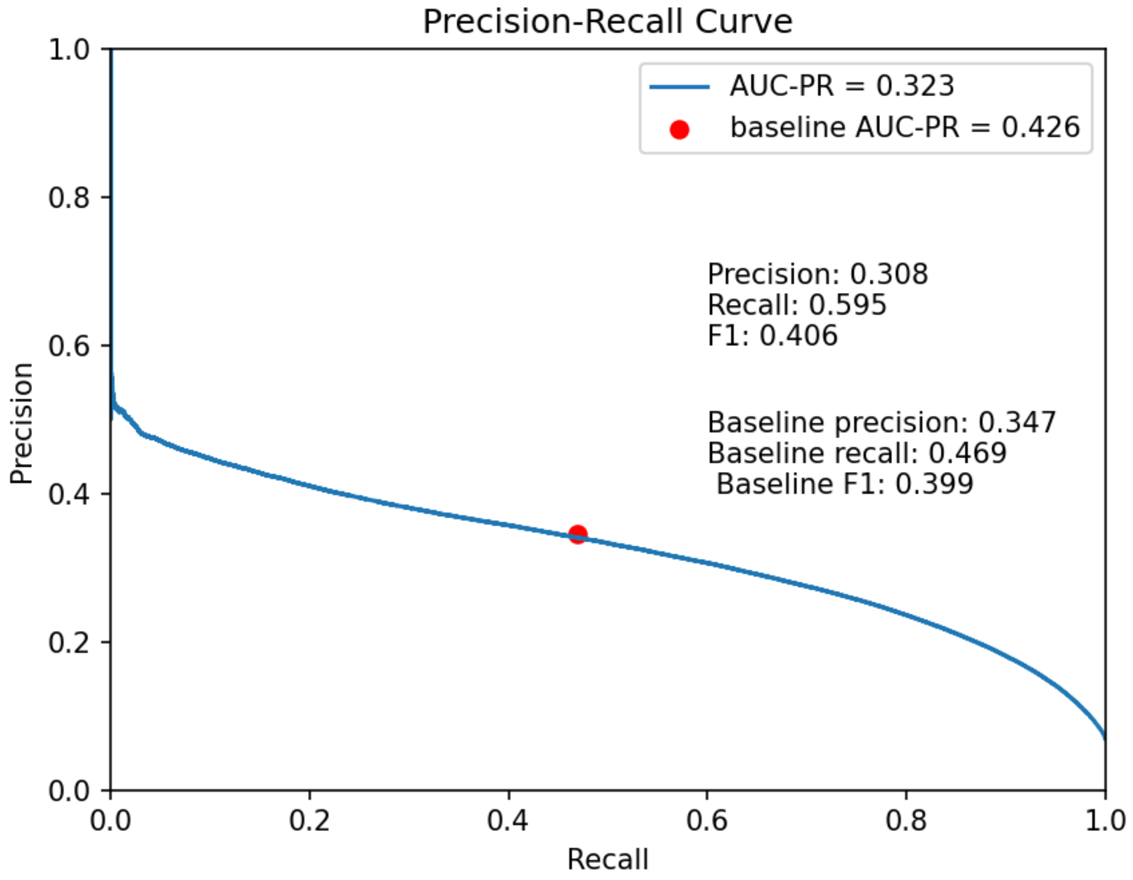
Model A: 12 features



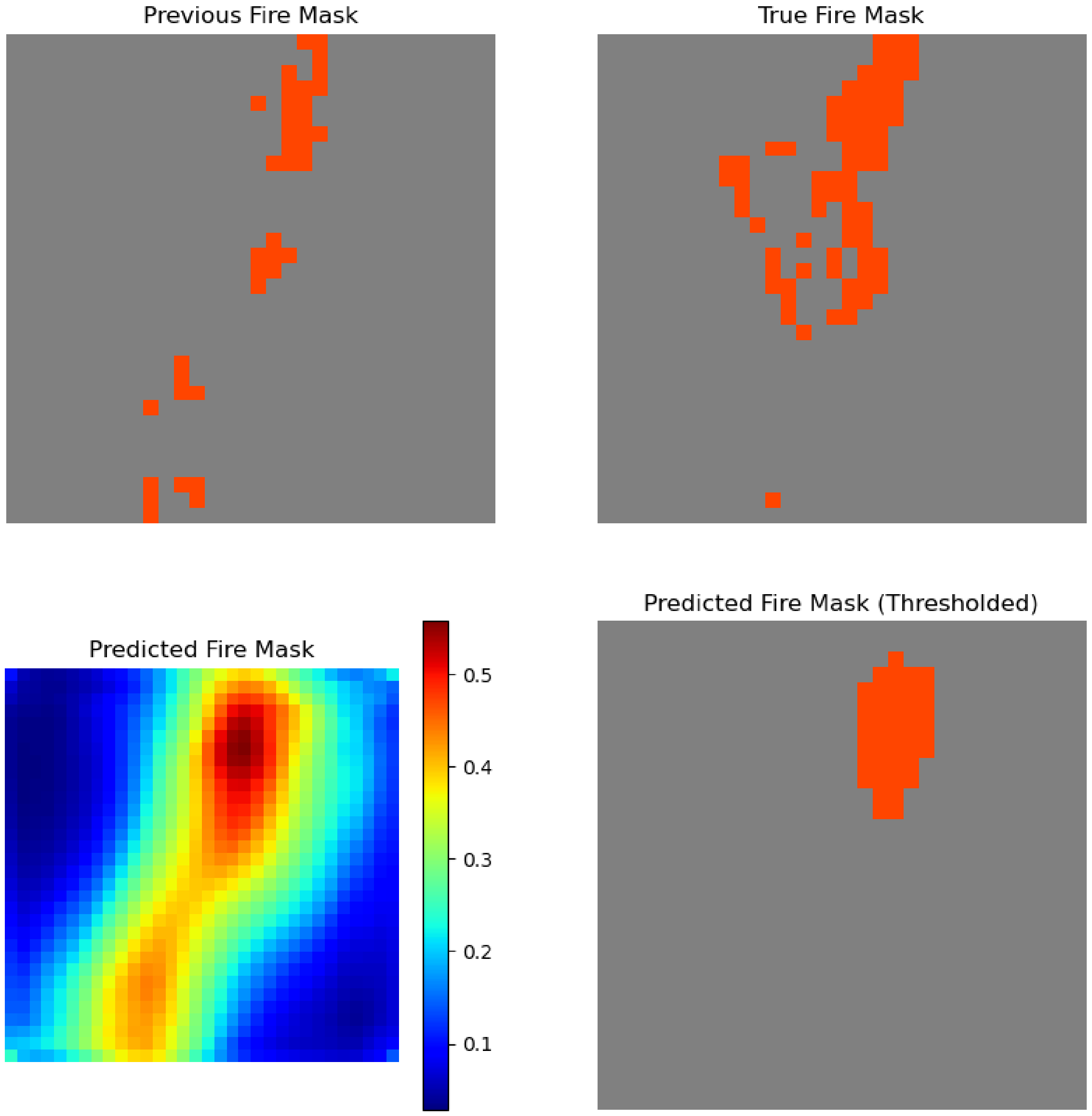


F1 score does not beat persistence baseline, and model predictions appear very consolidated and low-skill.

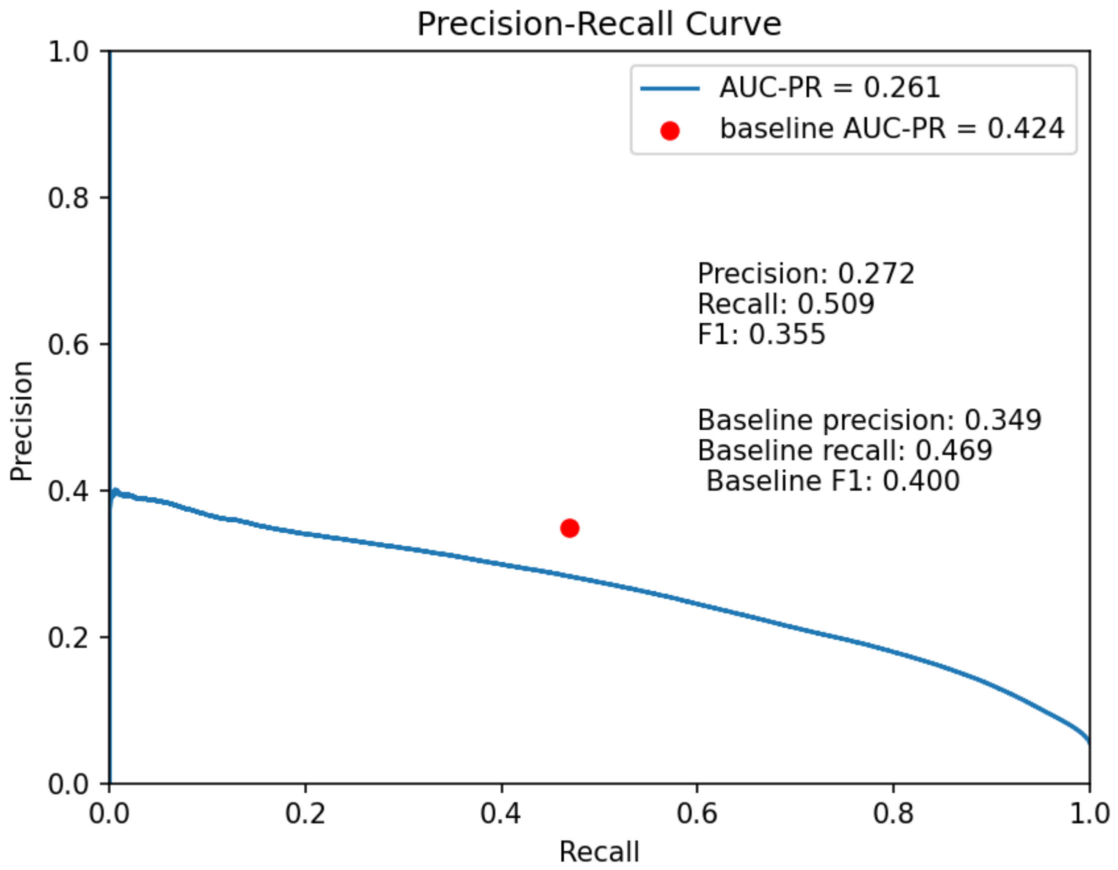
Model B: 15 features

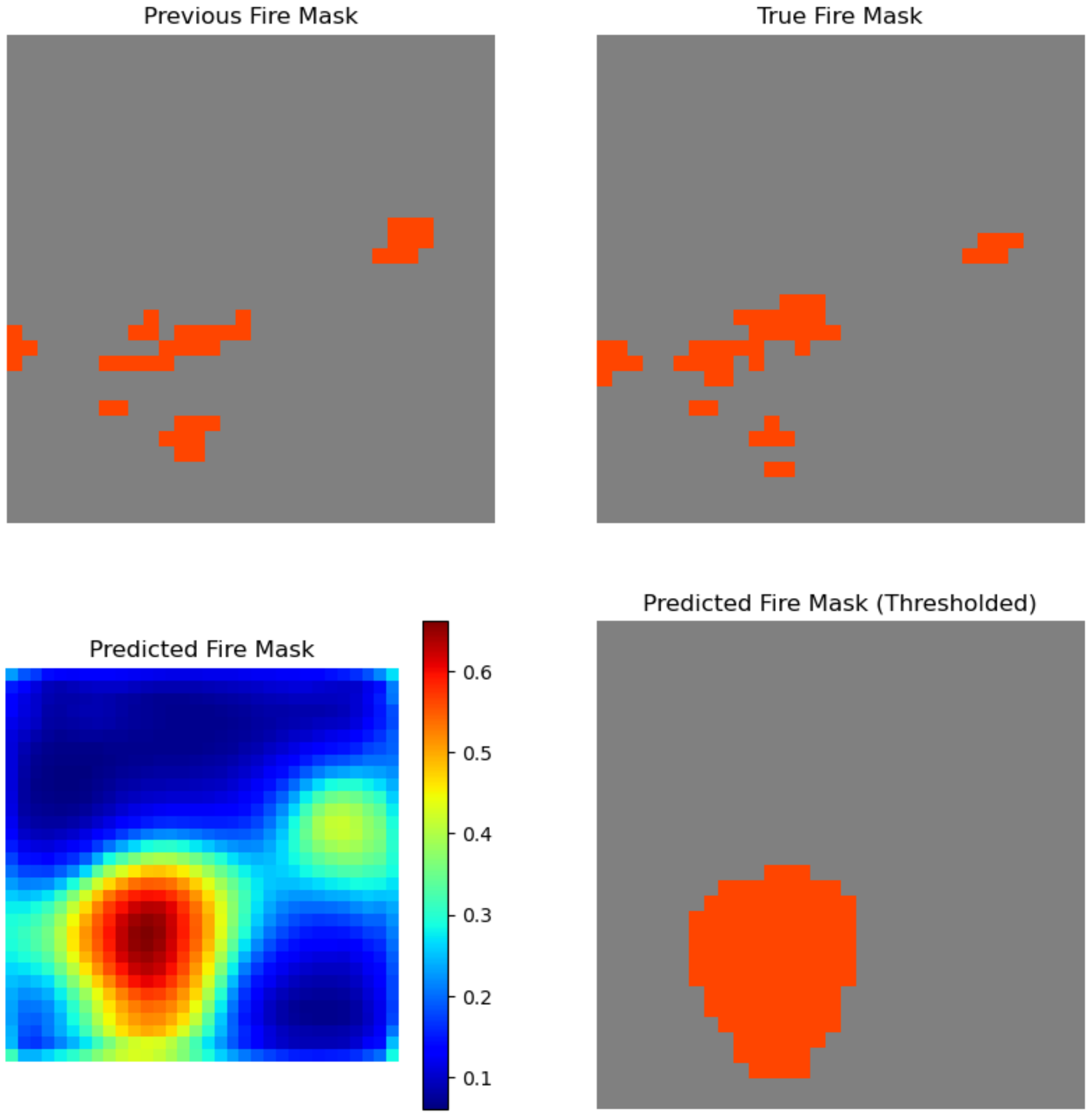


Baseline is slightly superseded, but model predictions still lack sparsity that characterizes the ground truth fire masks.



Model C: 18 features





With 18 features, performance is significantly diminished. This suggests that more procedural ablation studies be performed to identify an optimal feature subset.

Alternative evaluation: predicting new pixels

- Instead of grading based on predicting the next day fire mask exactly, consider model skill in predicting **only new ignitions**
  - Remove pixels which are ignited in the previous fire mask
  - By construction, the persistence model has no skill at this task.
- Evaluate remaining pixels by AUC-PR, F1 score
- Model B still demonstrates best performance.

Model	AUC-PR	F1 score at threshold
Model A (12 features)	0.179	0.276
Model B (15 features)	**0.192**	**0.286**
Model C (18 features)	0.146	0.236

Evaluation on new ignition prediction

Takeaways

- Feature selection is important: adding some features can improve performance while adding too many may diminish performance
- With low focal loss weight and a standard encoder architecture, predictions generally fail to beat the persistence baseline
- To beat persistence, let it be an easily learnable pathway through the network
- Focus on sparsity of predictions: penalize overprediction of ignited pixels

Experiment 2

- Modify architecture to include more residual connections
  - In particular, bring previous fire mask directly to output convolutional blocks

- Focal loss with $\gamma = 2$ (focus more on hard-to-classify examples)

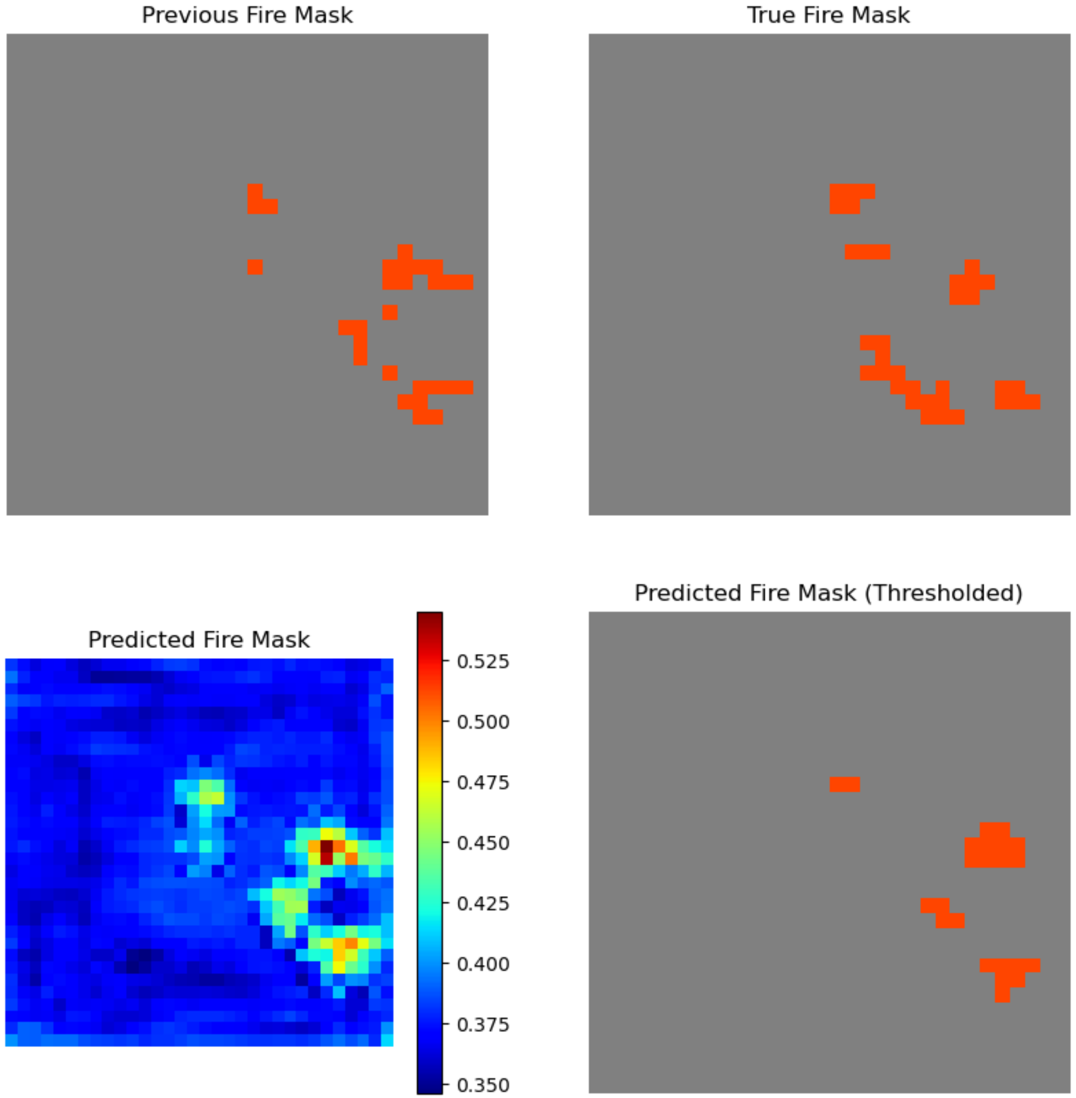
- Use an L1 penalty on predictions to encourage sparsity ($\lambda = 0.001$)
  - Loss = (focal loss) + λ*mean(abs(predicted fire pixels))

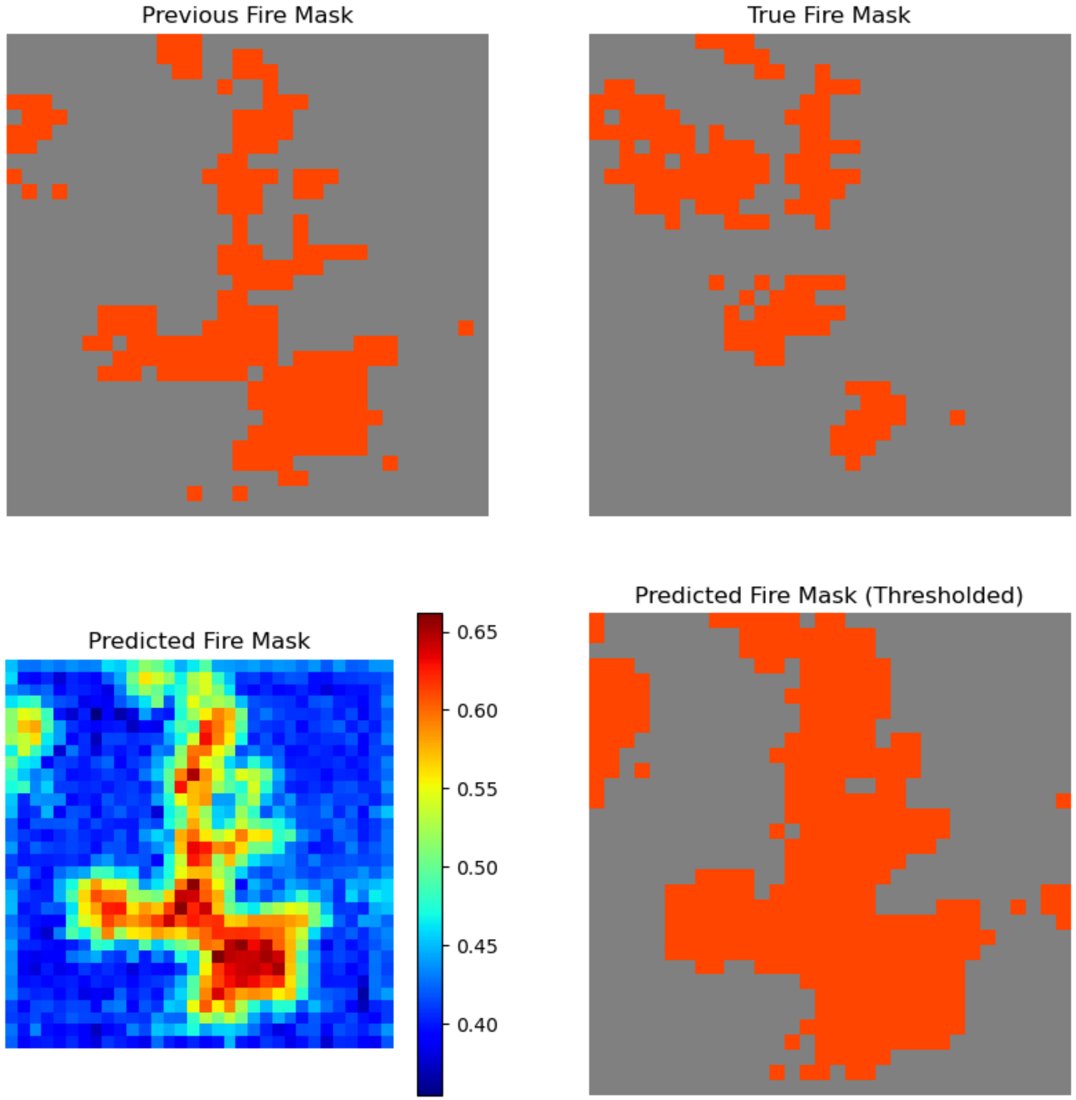
- Repeat a feature selection experiment

Results: experiment 2

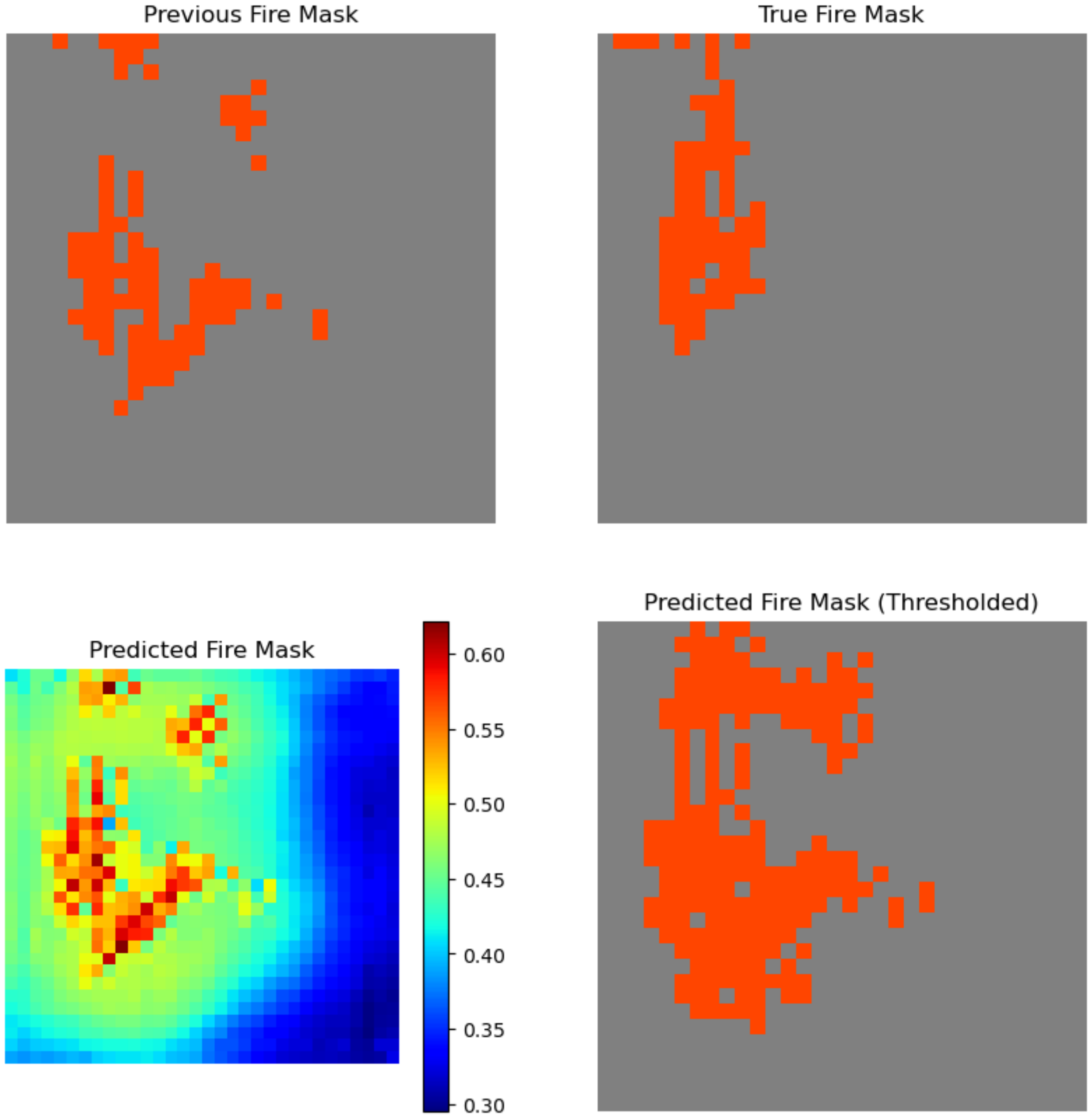
Model	Features	AUC-PR (full prediction)	F1 score (full prediction)	AUC-PR (new ignitions)	F1 score (new ignitions)
Persistence (baseline)	Previous fire mask only	n/a	0.399	n/a	0
Model A	Baseline features w/o temperature	0.267	**0.414**	0.163	**0.272**
Model B	Baseline features + latent fuels	**0.316**	0.402	**0.176**	0.264
Model C	Baseline features + latent fuels + wind gust data	0.254	0.401	0.140	0.249

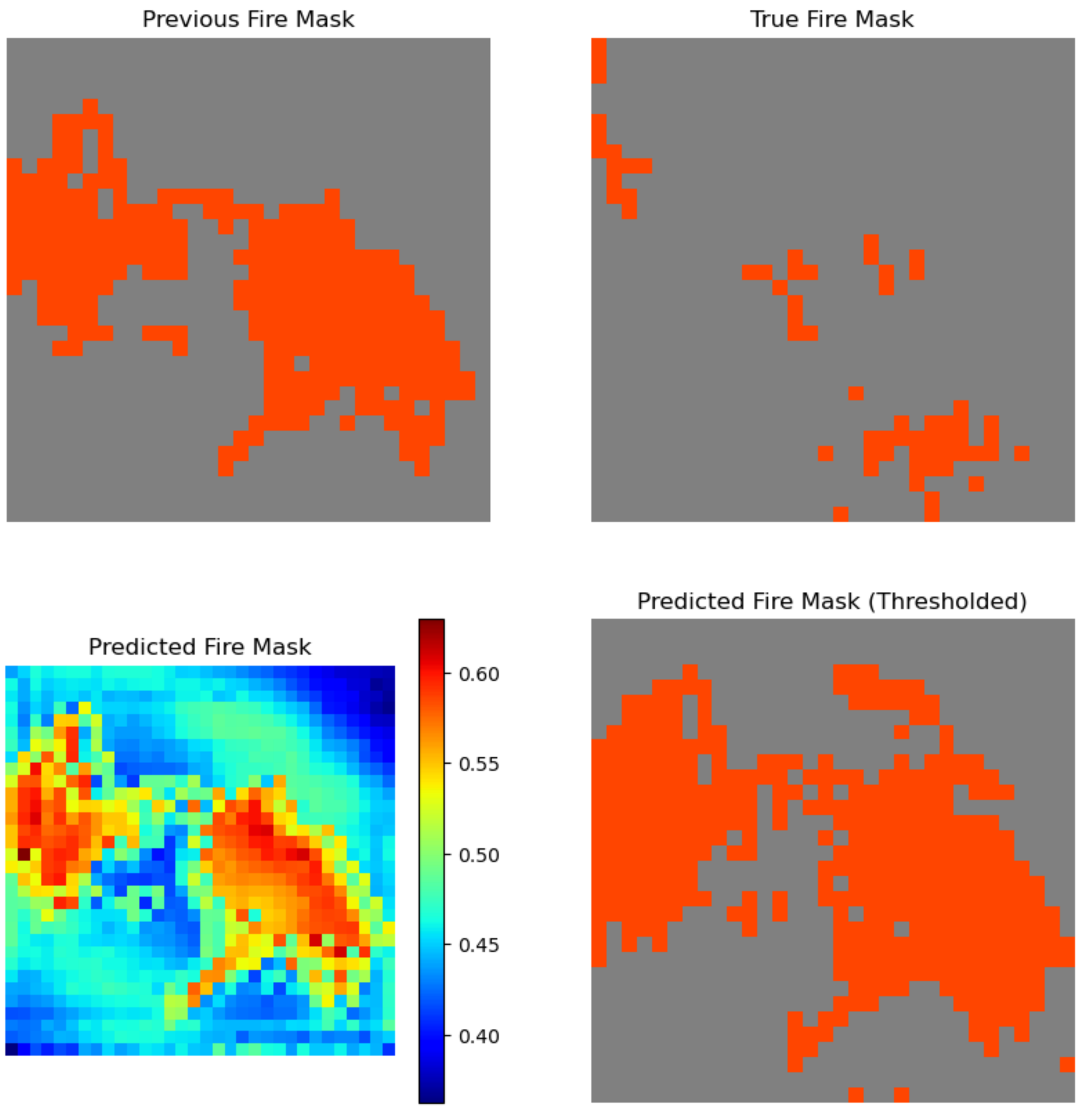
Model A: example predictions





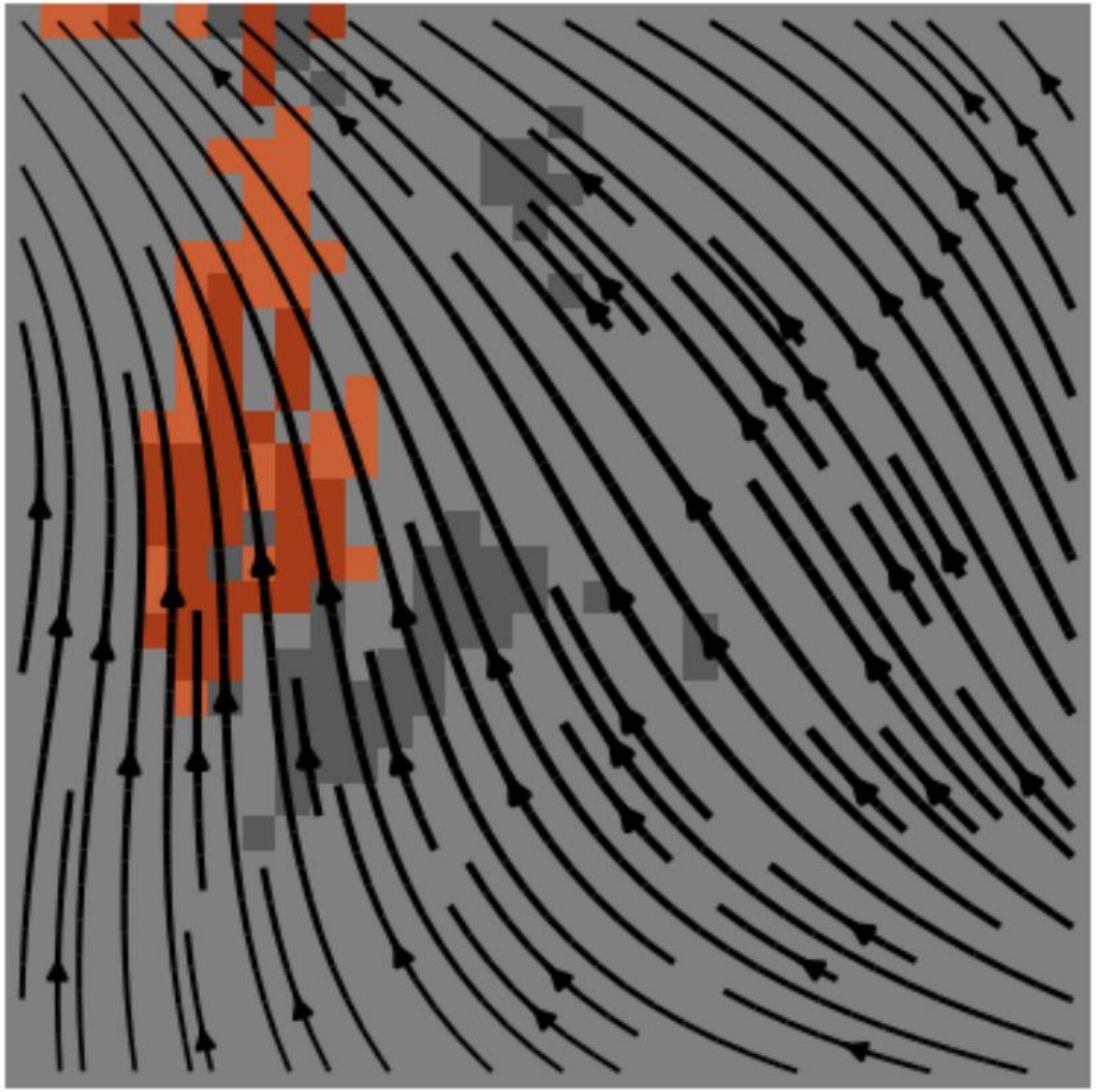
Model B: example predictions



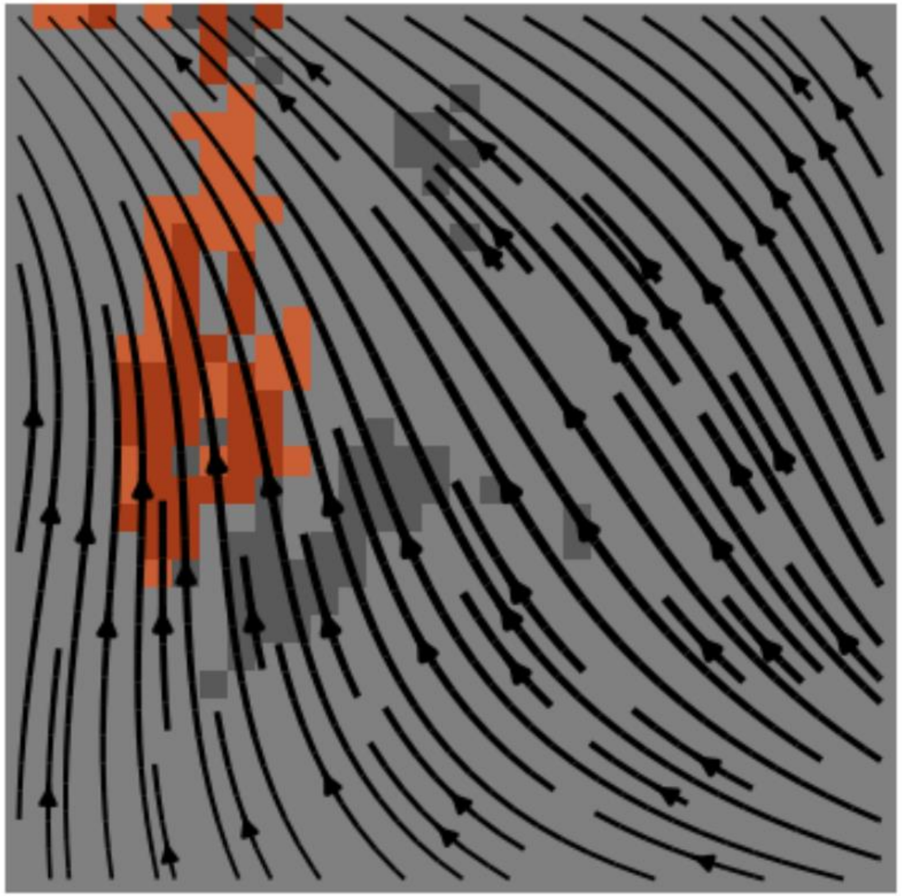


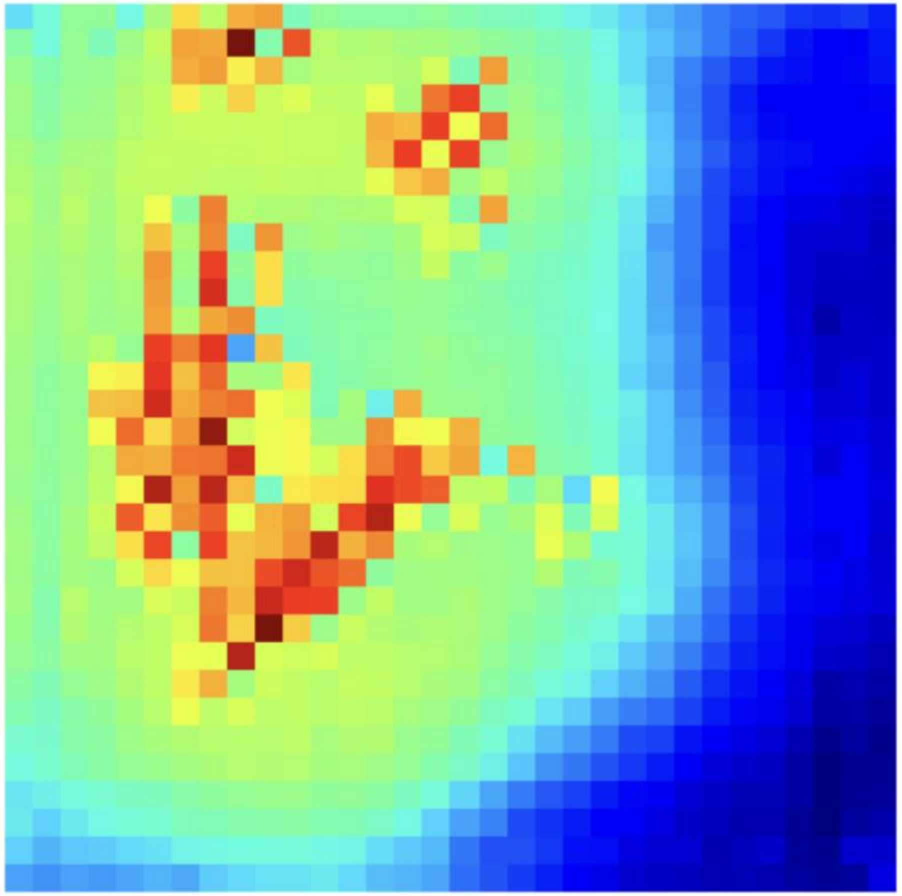
Is the model learning e.g. wind-driven spread?

- While the models can beat the persistence baseline, it's not anecdotally clear if they are learning non-trivial spread dynamics.
- Wind is known to be a factor in fire spread, and we can visualize some wind-driven fires that exist in the dataset (right).
    - Orange: next day
    - Gray: previous day
    - Orange-red: persisted
- Does the model predict spread in a direction corresponding to the wind, as can be seen in the ground truth?



The model does not appear to be learning the wind effect.





Conclusions

- Models beat the persistence baseline, but do not display much skill at the prediction task.
- Enforcing sparsity and using focal loss seem to improve model performance.
- Architecture optimization and feature selection will play an important role in future experimentation.
- Though limited by available compute here, a larger dataset and larger model is likely to help improve inferential performance.

Next steps

- A larger modified NDWS dataset, exported over the western CONUS region, will be publicly available for training new models.
- The codebase used to generate the dataset will be available on GitHub so modifications can be made as desired.
  - Relevant LANDFIRE and VIIRS assets will also be made accessible.
- More compute, more engineering is needed to realize the potential, if any, of this approach to wildfire spread modeling.
- Higher temporal resolution fire detection data is imperative if deep learning approaches are to compare to conventional fire models.

UC SANTA BARBARA