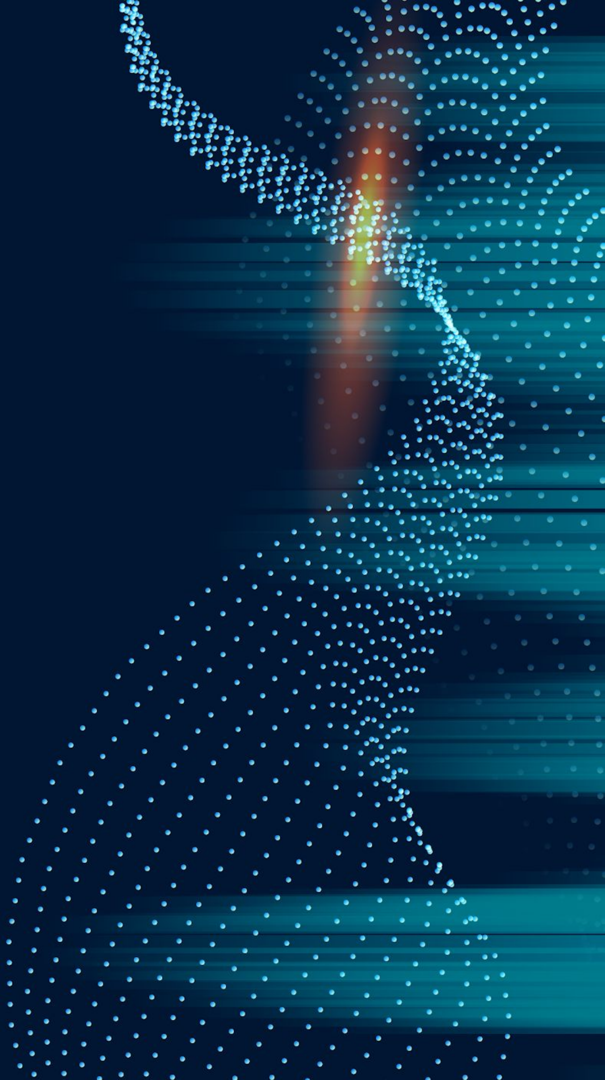
Wildfire Spread Prediction

CMPE 252

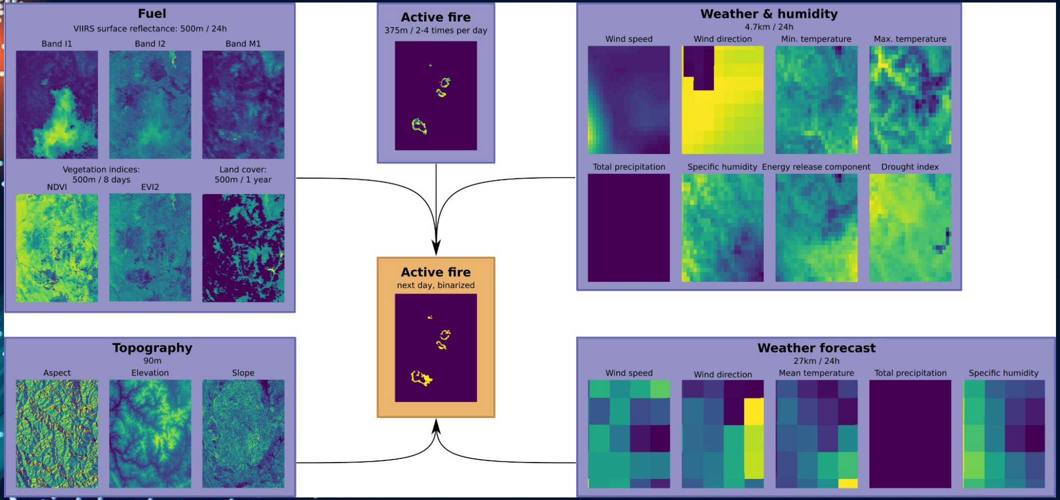
Neil Mewada, Ganesh Thampi

AGENDA

- Dataset & Pre-processing
- Model Architecture
- Training
- Model Results
- Model Performance



WildfireSpreadTS Dataset



Open dataset:

- 23 Channels
- ~50 GB Dataset
- 24 hour temporal resolution

Our subset:

- ~4.2 GB
- Continuous sequence with most fire activity

Data Pre-processing

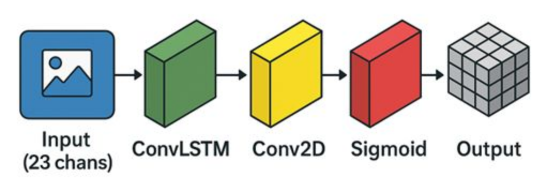
- Computed min and max range per channel.
- Normalized each channel using their min-max range.
- Downscale all images to 128x128.
- Step-function of 0.15 on ground truth (23’rd channel).
- 80-20 training and testing data split.
- Batch size of 8.
- 5 consecutive days’ input, 6th day’s output. (sliding window)

Why Convolutional LSTM?

- LSTMs are good with time-series data, i.e. sliding window.
- Convolutional LSTMs for spatio-temporal data.

Model Architecture

- Sliding window of 5 days' input, and 6th day as output.
- ConvLSTM
  - 64 hidden dimensions (output)
  - 3x3 kernel size
- Conv2D Layer
  - 64 input, 3x3 kernel size, padding = 1
- Sigmoid function



Model Architecture

```
ConvLSTM(
      input_size=(128, 128),
      input_dim=23,
      hidden_dim=[64],
      kernel_size=(3, 3),
      num_layers=1
)

nn.Sequential(
      nn.Conv2d(64, 1, kernel_size=3, padding=1),
      nn.Sigmoid()
)
```

Post-processing

- Step-function of 0.5 on model prediction.
    - Worked better in our testing

Training Process

- BCEWeightedLoss
  - pos_weight = 10, neg_weight = 1
- lr = 0.01
- Adam Optimizer
- Batch size = 8
- num_epochs = 150

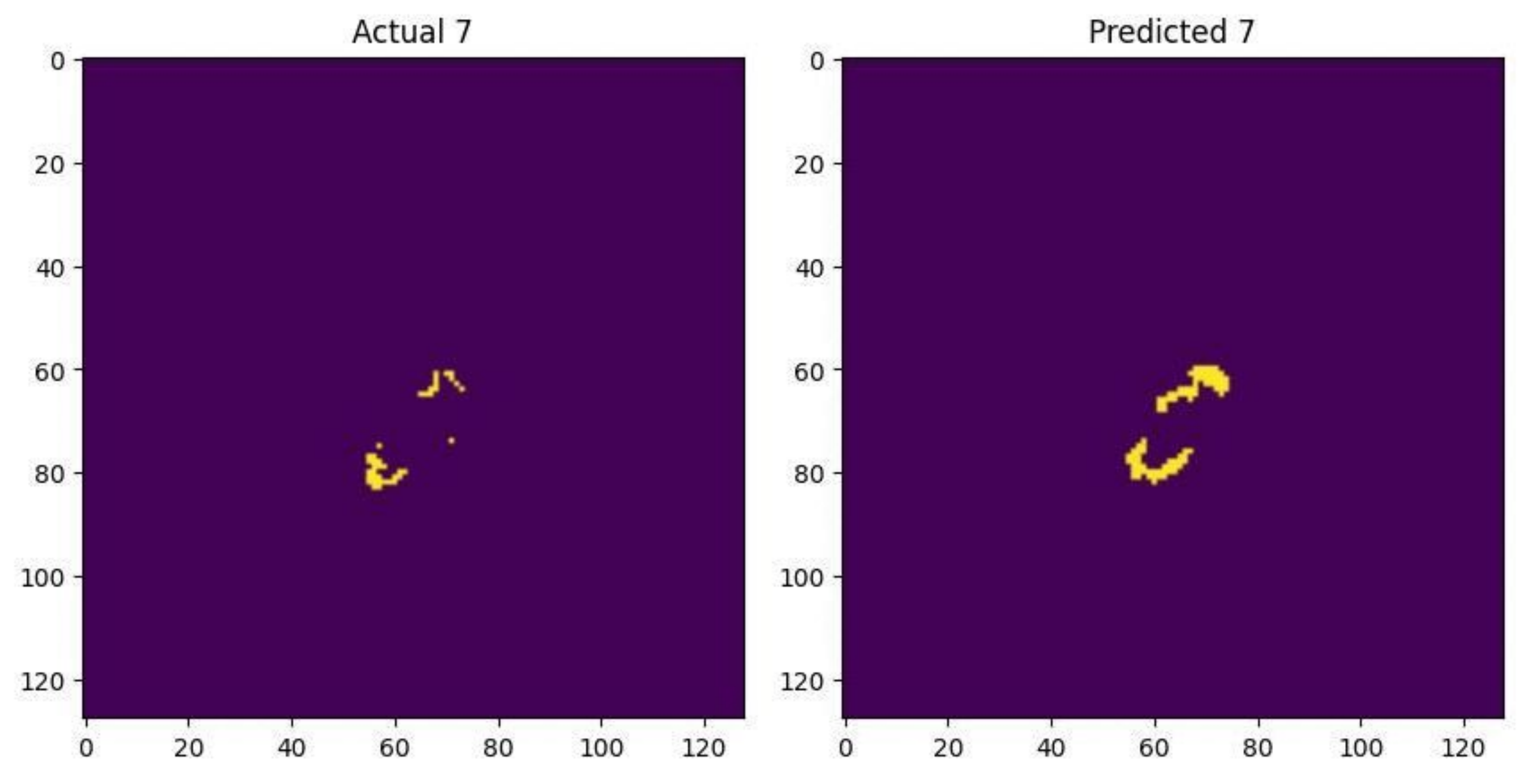
Training Process

Trained locally on a **Desktop PC**:

- AMD Ryzen 9 9950X CPU 16 Core / 32 Threads @ 5.7 GHz
- 32 GB DDR5 RAM @ 5600MT/s
- Nvidia RTX 5070 Ti - 16 GB VRAM
- Dataset stored on a Nvme Gen4 SSD

Note: You will need Nvidia GPU to run the model

Model Results

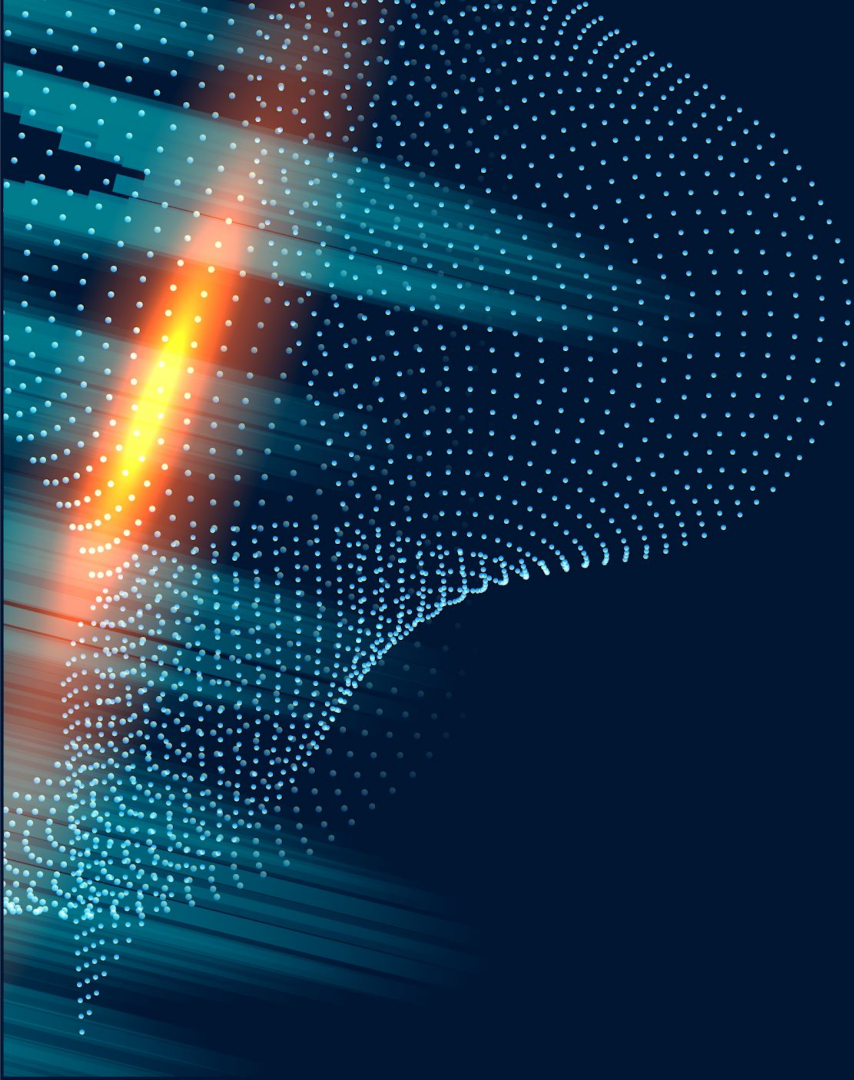


More results in “Prediction Visualizations.pdf”

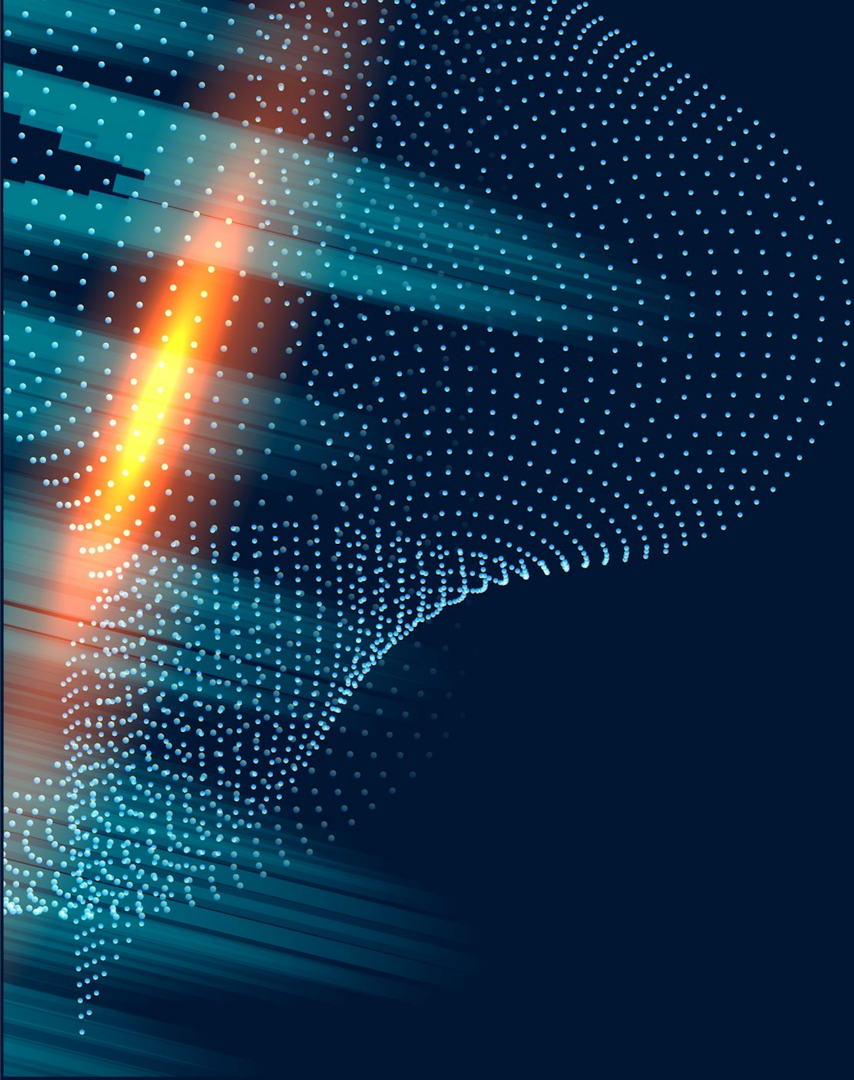
Model Performance

Average Precision	Intersection over Union	Dice	Recall	F1	Accuracy*
0.16341	0.12337	0.18755	0.29537	0.21041	~99%*

*Accuracy is irrelevant for sparse binary data



Q&A



Thank You