

Project G4: Next-Day Wildfire Spread Prediction on mNDWS

- **Start simple:** Implement per-pixel logistic regression on normalized features first—if this achieves $F1 > 0.40$, quantify spatial context lift before U-Net.
- Deep segmentation is warranted for spatial fire dynamics, but justify U-Net over simpler FCN or even patch-based ResNet classifier with ablation showing $< 5\%$ $F1$ gap.
- Skip MA-Net or DeepLabV3+ initially—compact U-Net with ResNet-18 encoder ($\sim 11M$ params) is sufficient for 500m resolution given prior $F1 \sim 0.64$ benchmarks.

Recommended actions before Milestone-2

- **Dataset verification:** Download mNDWS TFRecords, write loader script, verify train/val/test splits exist, and display 5 sample tiles with burn masks overlaid.
- **Create validation split:** If only train/test provided, hold out 10% of train for validation (stratified by transect/region) to tune hyperparameters without test set leakage.
- **Logistic regression baseline:** Train classifier with L2 regularization, report $F1$ /IoU/precision@90% recall on val set within 1 week.
- **Define U-Net spec:** ResNet-18 encoder (ImageNet pretrained), 4 decoder blocks, input=all 15 channels or subset (justify), output=sigmoid, loss=focal($\gamma=2$)+Dice.
- **Reproducibility setup:** Fix seeds (torch 42, numpy 42), [recommended:] log to W&B or MLflow (hyper-params, metrics per epoch, training time), save configs as YAML.

Ablations

- **Wind on/off:** Expect $\geq 10\%$ $F1$ drop without wind features (Shadrin: $F1 \sim 0.51$ without wind); if drop $< 5\%$, question wind's learned importance vs correlation with other features.
- **Loss function:** Focal($\gamma=2$)+Dice vs weighted-BCE+Dice—hypothesis is focal better handles 3% imbalance, yielding $+2\text{--}3\%$ $F1$ by focusing on hard negatives.
- **Fuel embeddings:** Compare fuel1–3 vs empty channels—expect $+3\text{--}5\%$ $F1$ gain, testing whether 30m→500m LANDFIRE compression retains predictive signal.

Risks & mitigations

- **Risk:** 3% class imbalance causes trivial “predict no-burn everywhere” baseline—**Mitigation:** Use focal loss ($\gamma=2$), monitor precision/recall separately, and oversample positive tiles 3:1.
- **Risk:** U-Net overfits to spatial location rather than learning fire dynamics—**Mitigation:** Apply random crops, horizontal/vertical flips, and early-stop on validation $F1$ (patience=5).
- **Risk:** Full mNDWS may exceed storage/memory—**Mitigation:** Start with max-side-800px version (160GB), precompute 256×256 crops, cache to SSD.

Open questions

- What are mNDWS tile dimensions ($H \times W$) in TFRecords, and will you crop/pad or train at native resolution with adaptive pooling?
- How will you prevent temporal leakage (same fire across days in train/val/test)—are splits already temporally stratified, or must you implement fire-ID-based splits?