

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

IEEE GRSS HACKATHON

PROBLEM STATEMENT - DATA DRIVEN AI IN REMOTE SENSING

DONE BY SOLAR HAWKS – SKIBIDI

INITIAL SETUP:

Visited the provided AWS login page:

AWS Login Portal

Use our assigned team name as the username and the provided password to log in.

After logging in, Create a new private JupyterLab space

Created instance

Clone the repository `git clone https://github.com/NASA-IMPACT/rsds-hackathon-24.git`

In the terminal, run `sudo apt-get update && sudo apt-get install ffmpeg libsm6 libxext6 -y`

Install the required Python packages: `pip install -r requirements.txt`

Opened the `training_terratorch.ipynb` notebook in JupyterLab.

Downloaded the development set of training data.

Created necessary directories.

Use Weights & Biases (wandb) to track progress in real-time.

Generated results and produce a TensorBoard log for monitoring.

1ST STAGE TRAINER CONFIGURATION:

Precision: Mixed-precision training (16-bit floating point) is used for faster training on supported hardware.

Patience Increased: Early stopping patience increased from 1 to 5 to avoid premature stopping and allow the model more time to converge. Mode set to min to ensure the model is stopping when the val/loss reaches a minimum.

Max Epochs Increased: Increased max_epochs from 5 to 100 to allow more training for better convergence.

Batch Size Increased: Increased batch_size from 4 to 16 to improve training efficiency (if hardware allows).

Checkpointing: Enabled checkpointing to save the model during training for recovery or further analysis.

2ND STAGE TRAINER CONFIGURATION:

Logging: Added version: "improved" to the logger for better version tracking.

log_graph: true enabled to log the computation graph in TensorBoard.

Early Stopping: Increased patience from 5 to 10 to allow more epochs before early stopping.

Set min_delta: 1e-4 to stop when the improvement in loss is smaller than 0.0001.

Model Checkpointing: Added checkpointing with the top 3 best models saved based on the minimum val/loss.

Filenames include epoch and loss: {epoch}-{val_loss:.2f} for better tracking.

Gradient Clipping: Added gradient_clip_val: 1.0 to prevent exploding gradients.

Gradient Accumulation: Set accumulate_grad_batches: 2 to accumulate gradients over 2 batches, useful when batch size is limited by memory.

Max Epochs: max_epochs: 100 retained to ensure longer training for better convergence.

Model Configuration:

Decoder: Switched to UperNetDecoder from FCNDecoder for enhanced performance, available in terratortch.

Loss Function: Changed to dice_ce, a combination of Dice and Cross-Entropy Loss, improving the handling of class imbalance.

Decoder and Convolutions: Retained decoder_channels: 16 and decoder_num_convs: 3 to maintain feature extraction capabilities.

Dropout: Increased head_dropout to 0.2 for better regularization.

Tiled Inference: Increased crop size to 1024x1024 with strides of 768x768 to accommodate larger tiles for inference.

Optimizer: Using AdamW for weight decay optimization with learning rate lr: 0.001 and weight_decay: 0.01. Added betas: [0.9, 0.999] for more control over AdamW's momentum parameters.

Learning Rate Scheduler:

OneCycleLR: Replaced CosineAnnealingLR with OneCycleLR for more aggressive learning rate scheduling. Set max_lr: 0.001, with a start of 30% (pct_start: 0.3) of training before the

learning rate begins to decay. `div_factor`: 25 and `final_div_factor`: $1e4$ to control the learning rate dynamics during training.

3RD STAGE TRAINER CONFIGURATION:

Precision: Set precision: 16-mixed for mixed-precision training to improve performance and memory efficiency.

Early Stopping: Set patience: 3 to stop training if validation loss does not improve within 3 epochs.

Model Checkpointing: Save only the best model based on val/loss by setting `save_top_k`: 1.

Max Epochs: Reduced `max_epochs` to 20 for faster convergence given the increase in early stopping.

Gradient Logging: `log_every_n_steps`: 50 ensures that logs are updated every 50 steps.

Data Module:

Batch Size: Increased `batch_size` to 8 for better gradient estimates and generalization.

Workers: Increased `num_workers`: 16 to speed up data loading.

Train Transform: Added ColorJitter for brightness, contrast, saturation, and hue augmentations to introduce more variability. Changed RandomCrop to crop at 224x224, reducing image size.

Normalization: Updated means and stds to lower values (0.01 for both) to account for potentially lower reflectance values in the dataset.

Model Configuration:

Decoder: Switched back to FCNDecoder with `decoder_channels`: 8 for simplicity and smaller decoder size. `decoder_num_convs`: 3 to increase the number of convolution layers for feature extraction.

Loss Function: Changed to dice loss for better performance on class-imbalanced segmentation tasks.

Dropout: Increased `head_dropout` to 0.2 for better regularization and preventing overfitting.

Optimizer:

Adam Optimizer:

Changed to Adam from AdamW, with a learning rate (lr) of 0.001 and reduced `weight_decay` to 0.001 for finer weight updates and less aggressive regularization.

Learning Rate Scheduler:

ReduceLROnPlateau:

Switched to ReduceLROnPlateau, reducing the learning rate when the validation loss plateaus.

Set patience: 5, allowing the model to have more epochs without improvement before reducing the learning rate, aiding fine-tuning.

Tiled Inference:

Tile Size and Stride: Reduced crop size and strides to 512x512 with 256x256 strides, ensuring better overlap for smoother results when reconstructing images from smaller patches.

DIFFERENT APPROACH:

Seed: seed_everything: 0 ensures consistency and reproducibility across all runs.

Early Stopping: patience increased to 5 to avoid premature stopping, allowing the model to improve.

Max Epochs: Increased to 100 for more thorough training and better convergence.

Data Module:

Batch Size: batch_size: 16 for better gradient estimates and more efficient training.

Train Transform: RandomCrop size increased to 256x256 to capture more detailed features.

Added augmentations: VerticalFlip with p: 0.5.

RandomRotate90 with p: 0.5 for increased data variability.

RandomBrightnessContrast to adjust brightness and contrast for more robust training.

Normalization: Updated means and stds based on typical reflectance values, improving image standardization.

Model Configuration:

Decoder: decoder_channels increased to 16 for better feature extraction.

Loss Function: Changed to dice_cross_entropy to handle class imbalance, combining Dice loss and cross-entropy loss.

Added class_weights: [1.0, 8.0] to prioritize detecting burn scars by giving more weight to the burned class.

Model Regularization: head_dropout set to 0.1 for regularization. decoder_num_convs increased to 3 to improve segmentation performance.

Optimizer:

AdamW: Switched back to AdamW for better weight decay behavior, with a reduced lr of 0.001 for stable training.

Learning Rate Scheduler:

CosineAnnealingLR: Replaced ReduceLROnPlateau with CosineAnnealingLR to gradually reduce the learning rate over time.

Set T_max: 50, which is half the number of max_epochs, ensuring smooth decay with minimal learning rate (eta_min: 1e-6).

Tiled Inference:

Tiled Inference Parameters: Adjusted h_stride and w_stride to 496 for better overlap between image tiles, enhancing segmentation accuracy at the tile boundaries.

NEW BASE APPROACH

Seed: Still set to 0 for reproducibility.

Logger: Enabled log_graph: true to log the computation graph. Disabled the default hyperparameter metric (default_hp_metric: false).

Early Stopping: Set patience to 3, meaning training will stop if no improvement after 3 epochs.

Model Checkpointing: Saves the top-performing model based on val/loss and stores it as best-checkpoint.

Max Epochs: Set to 100 for more training iterations.

Logging: Logs every 20 steps for more frequent updates.

Data Module:

Batch Size: Set to a smaller batch_size: 4 for better handling of smaller datasets and improving gradient estimates.

Transforms: Simpler augmentations: RandomCrop with 224x224 and HorizontalFlip for data variability.

Means and Standard Deviations: Specific values for each band reflecting their actual distributions.

Classes: Kept num_classes: 2 for a binary segmentation task.

Model:

Backbone: Uses the prithvi_vit_100 backbone with FCNDecoder for the semantic segmentation task. Increased the decoder channels to 256 for better feature extraction.

ViT-specific Parameters: patch_size: 16, embed_dim: 768, num_heads: 12 optimize the Vision Transformer architecture to extract more meaningful features from images.

Loss Function: Uses dice loss for handling class imbalance during segmentation tasks.

Tiled Inference: Performs inference with 224x224 crop size and 112 strides, improving prediction overlap and accuracy.

Optimizer:

Adam: Very small learning rate (lr: 1.3e-5) to ensure stable and slow convergence.

betas: [0.9, 0.999] with no weight decay (weight_decay: 0.0).

Learning Rate Scheduler:

ReduceLROnPlateau: Reduces the learning rate by a factor of 0.1 after 2 epochs if val/loss does not improve, with a minimum learning rate of 0.0.

1ST DERIVED BASE:

Removed Early Stopping Callback: The new configuration no longer includes the EarlyStopping callback, which stops training when val/loss does not improve after a certain patience.

Checkpointing: No change in model checkpointing, still saving the best model based on val/loss.

Transforms: No additional augmentations beyond RandomCrop and HorizontalFlip—simplified compared to the previous setup, which had more transformations like brightness/contrast adjustments and rotations.

Optimizer: No change in optimizer (still using Adam with the same learning rate and betas).

Removed Class Weights: The new configuration does not specify class weights in the loss function. The previous setup had a weighted loss function to handle class imbalance.

2ND DERIVED BASE:

Training Settings:

Max Epochs: Increased from 100 to 150 to allow more training time.

Batch Size: Increased from 4 to 8 for better gradient stability.

Augmentation:

New augmentations added:

ShiftScaleRotate for random shifting, scaling, and rotating images.

RandomBrightnessContrast for adjusting brightness and contrast during training.

Model:

Decoder: Changed from FCNDecoder to DeepLabV3, a more advanced segmentation model.

Head Dropout: Increased dropout rate from 0.1 to 0.2 for better regularization.

Loss Function: Changed from dice to dice_cross_entropy, combining Dice and Cross-Entropy for improved performance.

Tiling:

Tiling Parameters: Increased crop size from 224x224 to 512x512 for larger image patches. Reduced stride from 112 to 480 for more overlap between tiles.

Optimizer:

Learning Rate: Slightly increased from $1.3e-5$ to $3e-5$.

Weight Decay: Added $1e-5$ weight decay for regularization.

Learning Rate Scheduler:

Scheduler: Changed from ReduceLROnPlateau to CosineAnnealingLR, which provides a smoother decay of the learning rate.

FINAL TRAINER CONFIG:

Max Epochs: Reduced back from 150 to 100.

Batch Size: Batch size is returned to 4 (from 8 in the previous version)

Augmentations: The new augmentations (ShiftScaleRotate and RandomBrightnessContrast) added in the previous version are removed. The augmentations are simplified to only include RandomCrop and HorizontalFlip.

Model:

Decoder: Changed back from DeepLabV3 to FCNDecoder.

Head Dropout: Decreased back to 0.1 from 0.2.

Loss: Returned to dice instead of dice_cross_entropy.

Tiling:

The stride values are changed again for tiling. The stride for height and width is set to 496 (previously 480).

Optimizer:

Learning rate decreased back to $1.3e-5$ (from $3e-5$).

Weight decay removed (0.0 instead of $1e-5$).

Learning Rate Scheduler:

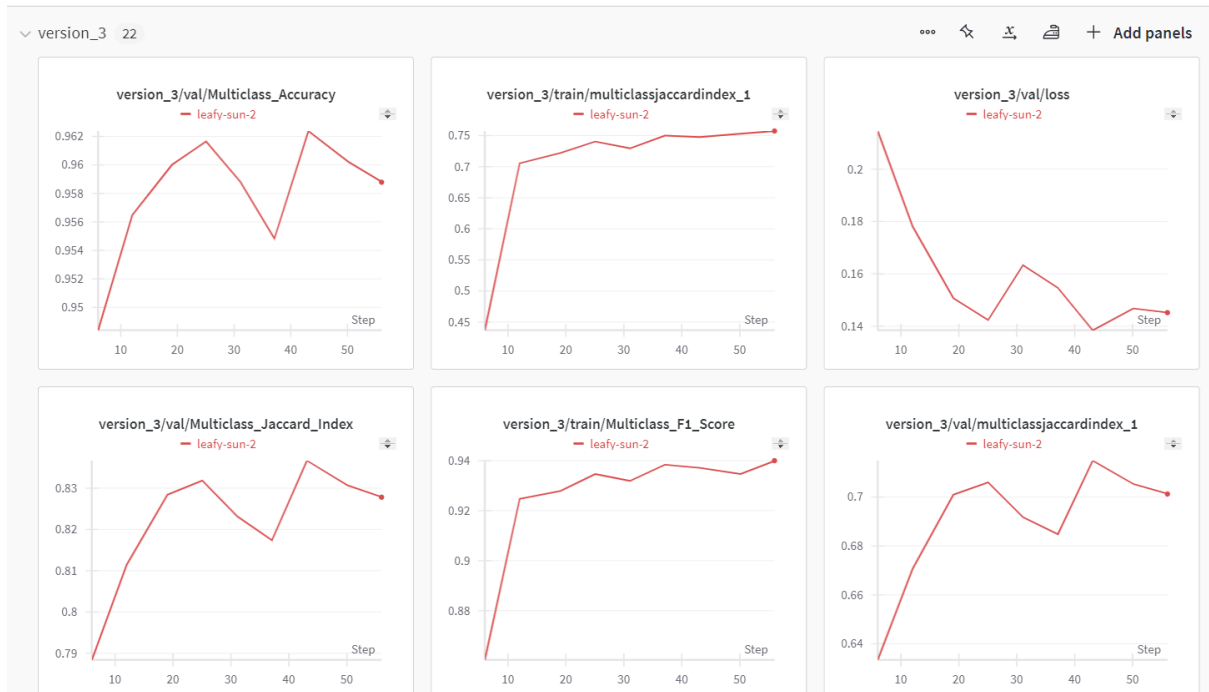
Scheduler reverted to ReduceLROnPlateau from CosineAnnealingLR.

RESULT AND ANALYSIS:

Version-2



Version-3



Version-11

