

Supplementary Information for:
“Interpretable rainfall modelling reveals rapid reorganisation of
Amazonian rainfall under vegetation loss”

Lilly Horvath-Makkos¹ and Fayyaz Minhas¹

¹Department of Computer Science, The University of Warwick, Coventry, United
Kingdom

Correspondence: Lilly Horvath-Makkos (lilly.horvath-makkos@warwick.ac.uk)

Supplementary Note 1: ConvLSTM equations

We present the ConvLSTM update equations used in our framework. Each ConvLSTM cell updates its hidden state H_t and memory cell C_t as follows:

$$\begin{aligned}i_t &= \sigma(W_{xi} * X_t + W_{hi} * H_{t-1} + b_i) \\f_t &= \sigma(W_{xf} * X_t + W_{hf} * H_{t-1} + b_f) \\C_t &= f_t \odot C_{t-1} + i_t \odot \tanh(W_{xc} * X_t + W_{hc} * H_{t-1} + b_c) \\o_t &= \sigma(W_{xo} * X_t + W_{ho} * H_{t-1} + b_o) \\H_t &= o_t \odot \tanh(C_t)\end{aligned}$$

where $*$ is convolution, \odot is the Hadamard product, and σ is the sigmoid function. Shi et al. (2015)

Supplementary Note 2: Hyperparameter selection and training protocol

Hyperparameters were tuned via 5-fold cross-validation. Candidate loss functions included standard MSE, Balanced MSE, and Smooth L1; Balanced MSE yielded the best performance for low-intensity rainfall and was selected.

For precipitation, we evaluated z-score, min-max, log transforms (offsets 1 and 10^{-6}), and log-standardisation. Log normalisation with offset 1 performed best. All other variables were standardised using z-score normalisation.

Architectural tuning included hidden dimensions for the last two ConvLSTM layers (32, 64), (32, 128), (64, 128), with (32, 64) chosen for optimal capacity-efficiency balance. Kernel sizes (2×2) , (3×3) , and (5×5) were tested; (3×3) was selected for spatial resolution and parameter efficiency.

Final choices were based on average validation performance and metric variance, with preference given to stable configurations.

The model consists of a shared 3-layer ConvLSTM encoder, followed by a 3D convolutional layer that refines the learned spatiotemporal features before branching into two task-specific heads. The classification head is a Conv3D layer that outputs the probability of dry conditions, while the regression head is a 3-layer MLP that estimates the intensity of rain.

The input at each timestep is a tensor of atmospheric and vegetation variables from the most recent hour, with shape $[(T = 1), H, W, C]$. These inputs are processed through three ConvLSTM layers with kernel size (3×3) and hidden dimensions of 14, 32, and 64. Each layer is followed by batch normalisation to improve training stability. A dropout layer is applied after the final MLP to reduce overfitting. Both output heads receive gradient updates during training, allowing the encoder to learn shared spatiotemporal representations relevant to both rainfall occurrence and intensity.

The model is trained using the composite loss described in Section 2.2.1, combining classification and regression terms. Training was conducted for 200 epochs and completed in approximately 45 minutes on a single GPU.

Optimisation is performed using the AdamW optimiser with an initial learning rate of 0.005, regulated by PyTorch's *ReduceLROnPlateau* learning rate scheduler Paszke et al. (2019).

MultiTask ConvLSTM Architecture

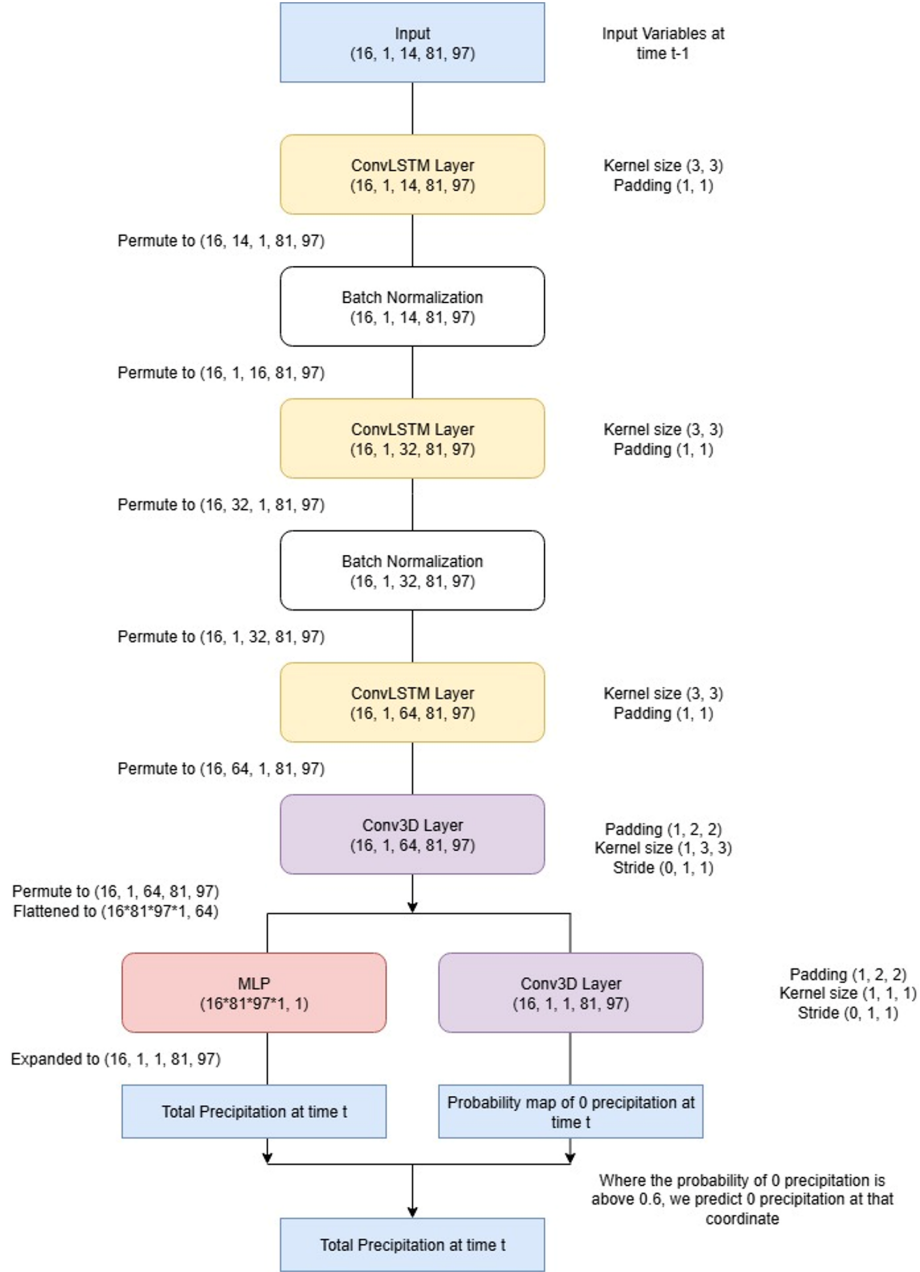


Figure S1. MultiTask ConvLSTM architecture. Schematic of the proposed architecture, with shared ConvLSTM layers and dual heads for classification and regression. The model takes as input hourly variables over latitude -15° to 5° and longitude -75° to -50° . These inputs are processed through a shared three-layer ConvLSTM encoder with batch normalization. A Conv3D layer refines the spatiotemporal features before branching into two task-specific heads: (i) a multilayer perceptron (MLP) regression head that estimates total precipitation at time t , and (ii) a Conv3D classification head that outputs the probability of zero rainfall at each grid cell. The dual-task design allows accurate discrimination between dry and rainy conditions.

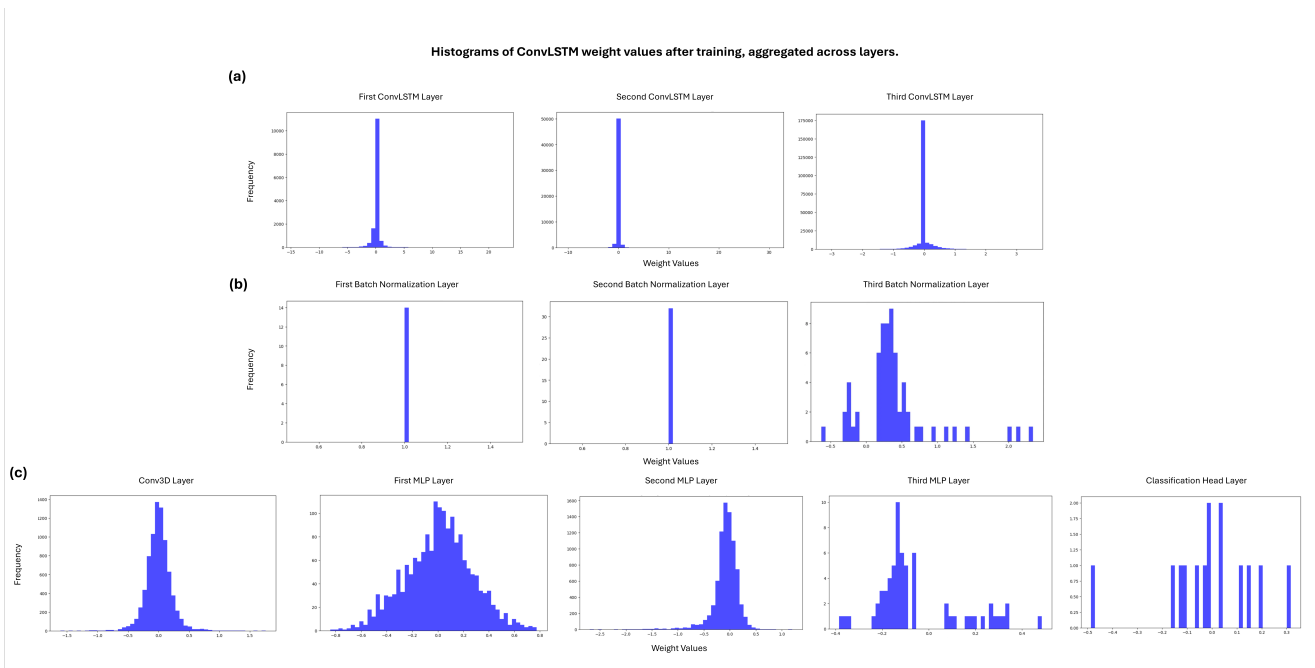


Figure S2. Histograms of learned weight distributions across layers in the MultiTask ConvLSTM, evaluated after training. (a) ConvLSTM encoder layers. Weight distributions are sharply centered near zero, with broader spread across the first and third layers. (b) Batch Normalization layers. The first two layers display nearly degenerate distributions at unity, whereas the third exhibits greater variability, including stronger rescaling and shifting during training. (c) Output layers. The MLP layers show approximately Gaussian distributions, consistent with well-regularized training. The Conv3D layer and classification head display sparser distributions, with weights spread across a broader range.

Gradient-based saliency maps of individual input variables in the ConvLSTM model evaluated on the validation set

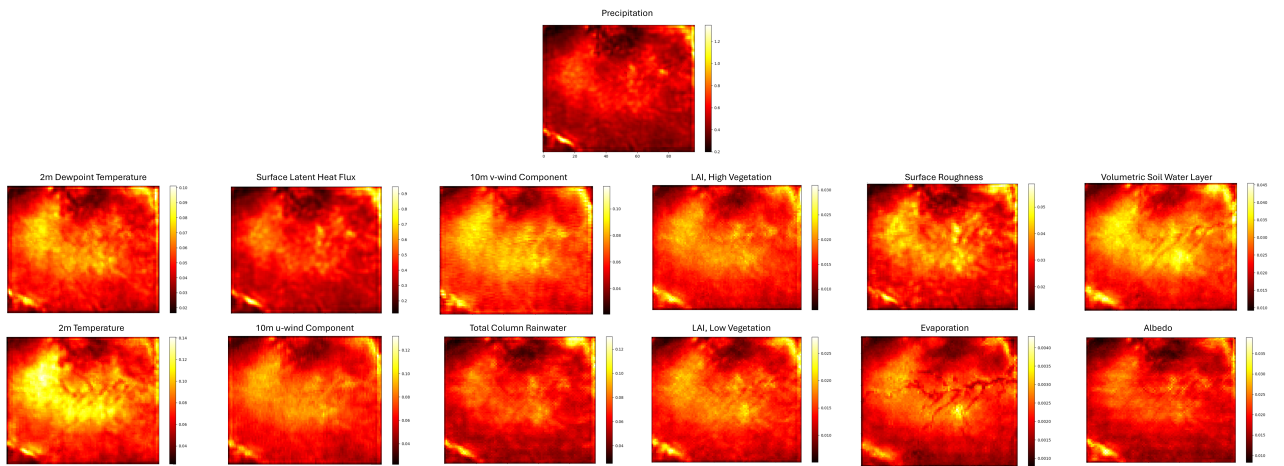


Figure S3. Gradient-based saliency maps of input variables in the MultiTask ConvLSTM, evaluated on the validation set. Each panel shows the spatial distribution of gradient magnitudes, with brighter intensities (yellow/white) indicating regions more influential for rainfall predictions. The model assigns the strongest sensitivity to the previous hour's total precipitation, reflecting its role as the primary predictor of the spatial distribution and magnitude of subsequent rainfall. Vegetation variables display lower overall saliency, about an order of magnitude smaller than atmospheric inputs, consistent with their indirect role in short-term rainfall generation. Evaporation shows near-zero saliency, likely due to its strongly diurnal variability being captured via latent heat flux. Notably, vegetation saliency maps resemble the spatial structure of Amazonian biomass, with higher influence detected in ecologically sensitive regions such as the Andes, central Amazonia, and the Atlantic coastal margin. This suggests that vegetation inputs contribute spatially structured signals, particularly at biome boundaries where vegetation–atmosphere coupling is strongest.

References

- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S.: PyTorch: An Imperative Style, High-Performance Deep Learning Library., in: Advances in Neural Information Processing Systems 32, pp. 8024–8035, 2019.
- Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-K., and Woo, W.-C.: Convolutional LSTM network: A machine learning approach for precipitation nowcasting, in: Adv. Neural Inf. Process. Syst., pp. 802–810, <https://arxiv.org/abs/1506.04214>, 2015.