# Training Logs and Model Evaluation Notes

## Training Log Table

| Model Variant | Key Changes | Accuracy (Val) | Loss (Val) | Notes and Observations |
|---|---|---|---|---|
| **LSTM v1** | 2 stacked LSTM layers (128 + 64), batch size = 128 | 86.3% | 0.41 | Strong accuracy, but training time was long; showed signs of overfitting. |
| **GRU v1** | Replaced LSTM with GRU layers (same size) | 84.9% | 0.45 | Trained ~25% faster; slight accuracy drop; less overfitting. |
| **LSTM v2** | Reduced to 1 LSTM layer (64 units), batch size = 256 | 85.7% | 0.39 | Faster training and more stable validation loss; improved runtime efficiency. |
| **LSTM v3 + XLA** | Enabled XLA JIT + mixed precision | 85.5% | 0.40 | Training speed improved significantly with no loss in accuracy. |
| **LSTM v4 (final)** | Same as v3 with learning rate tuning (0.001 → 0.0005) | 86.8% | 0.38 | Best performance; balanced training time, accuracy, and generalisation. |

---

## Observations

- Increasing dropout to 0.4 led to underfitting; validation accuracy dropped to ~83%.
- Using mixed precision and XLA together resulted in ~40% reduction in training time.
- GRU layers offered a faster alternative, but LSTM consistently performed better on WISDM data.
- Larger batch sizes (256) helped improve training throughput without hurting performance.
- Early stopping with patience = 3 helped prevent overfitting and saved training time.

| **Parameter** | **Values Tried** | **Best Performing Value** | **Parameter** |
|---|---|---|---|
| Learning Rate | 0.01, 0.001, 0.0005 | 0.0005 | Learning Rate |
| Dropout Rate | 0.2, 0.3, 0.4 | 0.3 | Dropout Rate |
| Batch Size | 64, 128, 256 | 256 | Batch Size |
| Optimizer | Adam, RMSprop | Adam | Optimizer |
| **Parameter** | **Values Tried** | **Best Performing Value** | **Parameter** |

This table summarizes the hyperparameter tuning process carried out during model optimization. The selected best-performing values were chosen based on validation accuracy and training time trade-offs. These results demonstrate my iterative tuning and model refinement efforts