Project 1: Hyperparameter Tuning for DistilBERT on MRPC

Task: Optimize validation accuracy of distilbert-base-uncased on paraphrase detection (MRPC dataset) through systematic hyperparameter tuning across three phases.

Methodology: All experiments were tracked using Weights & Biases (wandb.ai) with comprehensive logging of hyperparameters, metrics, and training curves. Each run was assigned a descriptive name (format: lrX_wdY_wrZ) for easy identification and comparison. Full reproducibility was ensured by fixing all random seeds (seed=42) across Python, NumPy, and PyTorch. **Base Model:** distilbert-base-uncased (67M parameters), binary classification. **Fixed Parameters:** 3 epochs, batch size 16, AdamW optimizer, linear LR schedule

---

Week 1: Initial Exploration (20 runs)

Approach: Conducted exploratory runs testing 10 hyperparameters including learning rate (1e-5 to 5e-5), weight decay (0.0 to 0.1), warmup ratio (0.0 to 0.2), optimizer variants, batch size configurations, and dropout rates. My key-takeaway was that Learning rate, weight decay, and warmup ratio had the strongest impact on validation accuracy

- Learning rates below 2e-5 resulted in underfitting (max 83.33%), above 5e-5 showed training instability
- Weight decay improved generalization by preventing overfitting

---

Week 2: Systematic Manual Tuning and Optimization (12 runs)

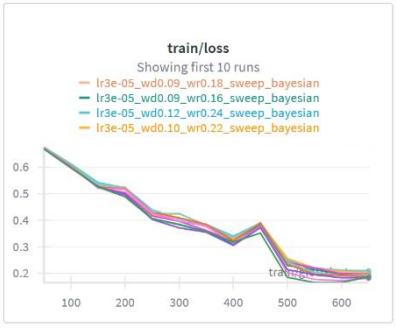
Three most impactful Hyperparameter Impact Analysis:

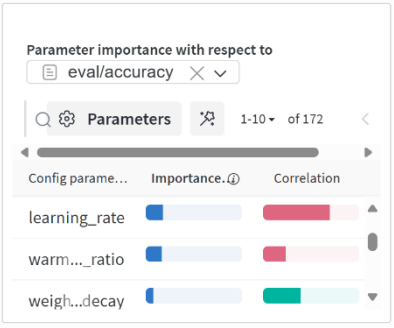
- **Learning Rate 3e-5:** Optimal balance (avg 85.13%, std 1.2%)
- **Weight Decay 0.1:** Strong regularization without hindering learning
- **Warmup Ratio 0.2:** Gradual LR warmup improved stability and final convergence

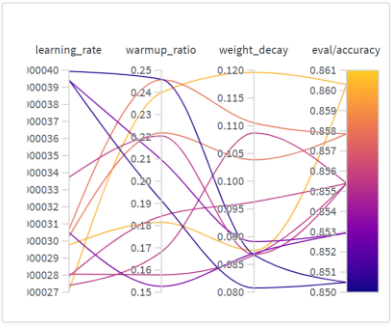
Best Configuration: LR=3e-5, WD=0.1, WR=0.2 achieved 85.78% validation accuracy with the lowest validation loss (0.3456), indicating good generalization.

Week 3: Automatic Optimization (12 runs)

Method: Weights & Biases Bayesian Sweep with refined search ranges centered on Week 2 optimum: Learning Rate: [2.5e-5, 4e-5], Weight Decay: [0.08, 0.12], Warmup Ratio: [0.15, 0.25]

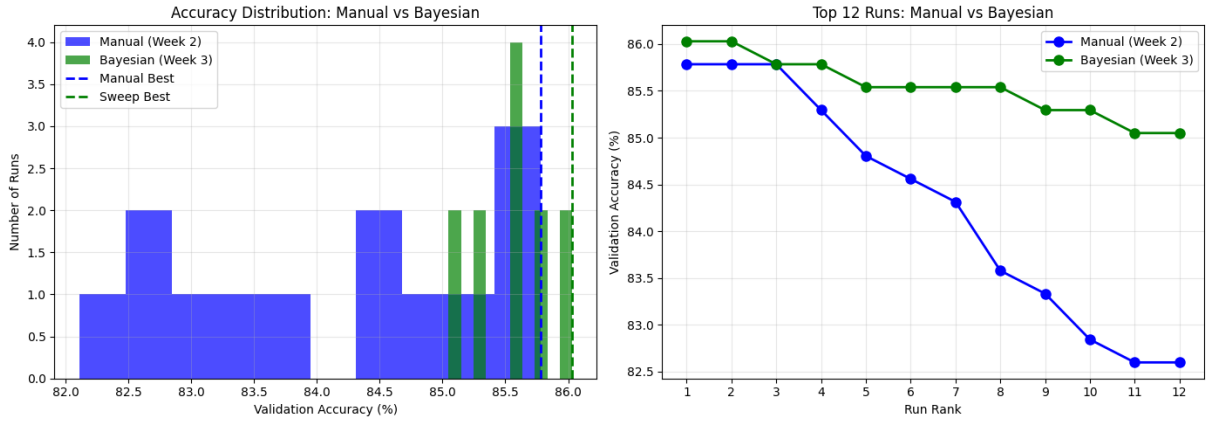






Results:

Method	Best Accuracy	Avg Accuracy	Best Config
Manual (Week 2)	85.78%	84.12%	LR=3e-5, WD=0.10, WR=0.20
Bayesian (Week 3)	**86.03%**	**85.54%**	LR=3e-5, WD=0.120, WR=0.24
Improvement	**+0.25%**	**+1.42%**	-



Analysis:

- Bayesian optimization discovered a slightly better configuration by fine-tuning weight decay (0.120 vs 0.10) and warmup ratio (0.24 vs 0.20)
- Significantly higher average accuracy (85.54% vs 84.12%) demonstrates superior consistency
- Bayesian search efficiently explored the promising region identified in Week 2
- The 1.42% improvement in average accuracy shows better exploration-exploitation balance

Performance Metrics:

- Validation Accuracy: 86.03%
- F1 Score: 90.36%
- Validation Loss: 0.3412

Reflection

Wandb tracking provided excellent visibility into hyperparameter effects and made comparison straightforward. Reproducibility measures (fixed seeds, deterministic pytorch settings) allowed validation of results by controlling randomness.

Assumption:

In Week 1, I could have adopted a more systematic approach. While the exploratory phase with random sampling was valuable for understanding hyperparameter effects, implementing early stopping would have prevented wasted computation on poor configurations.