Fine-Tuning Strategies: Comprehensive Comparison

Approach	Parameters Updated	Training Cost	Performance	Best Use Cases
Full Fine-Tuning Update all parameters 110M params **100%**	**100%** All 110M parameters trainable	**$500** 1 GPU-day 10K-100K examples needed	**100%** Maximum performance	• Large labeled datasets • Maximum accuracy required • Single-task focus
LoRA Low-Rank Adaptation 0.1-1M params **0.1-1%**	**0.5%** Small trainable matrices added	**$10-50** 0.02-0.1 GPU-day 1K-10K examples needed	**95-99%** Near full FT performance	• Production standard • Domain adaptation • Cost-sensitive projects • Multiple task variants
Adapter Layers Insert trainable modules 1-3M params **1-3%**	**2%** Small modules between layers	**$20-80** 0.05-0.15 GPU-day 1K-10K examples needed	**90-95%** Good performance	• Multi-task serving • Swappable tasks • Modular systems • Task isolation
Prompt Tuning Learn soft prompts 10K-100K params **0.01-0.1%**	**0.05%** Continuous vectors only	**$5-20** 0.01-0.05 GPU-day 100-1K examples needed	**80-90%** Moderate performance	• Many task variants • Minimal resources • Rapid experimentation • Limited data available

Recommendation: LoRA has become the production standard—95-99% performance at 10-50× lower cost than full fine-tuning