Fine-Tuning & Model Customisation

Unlocking the power of domain-specific AI through strategic model customisation

What is Fine-Tuning?

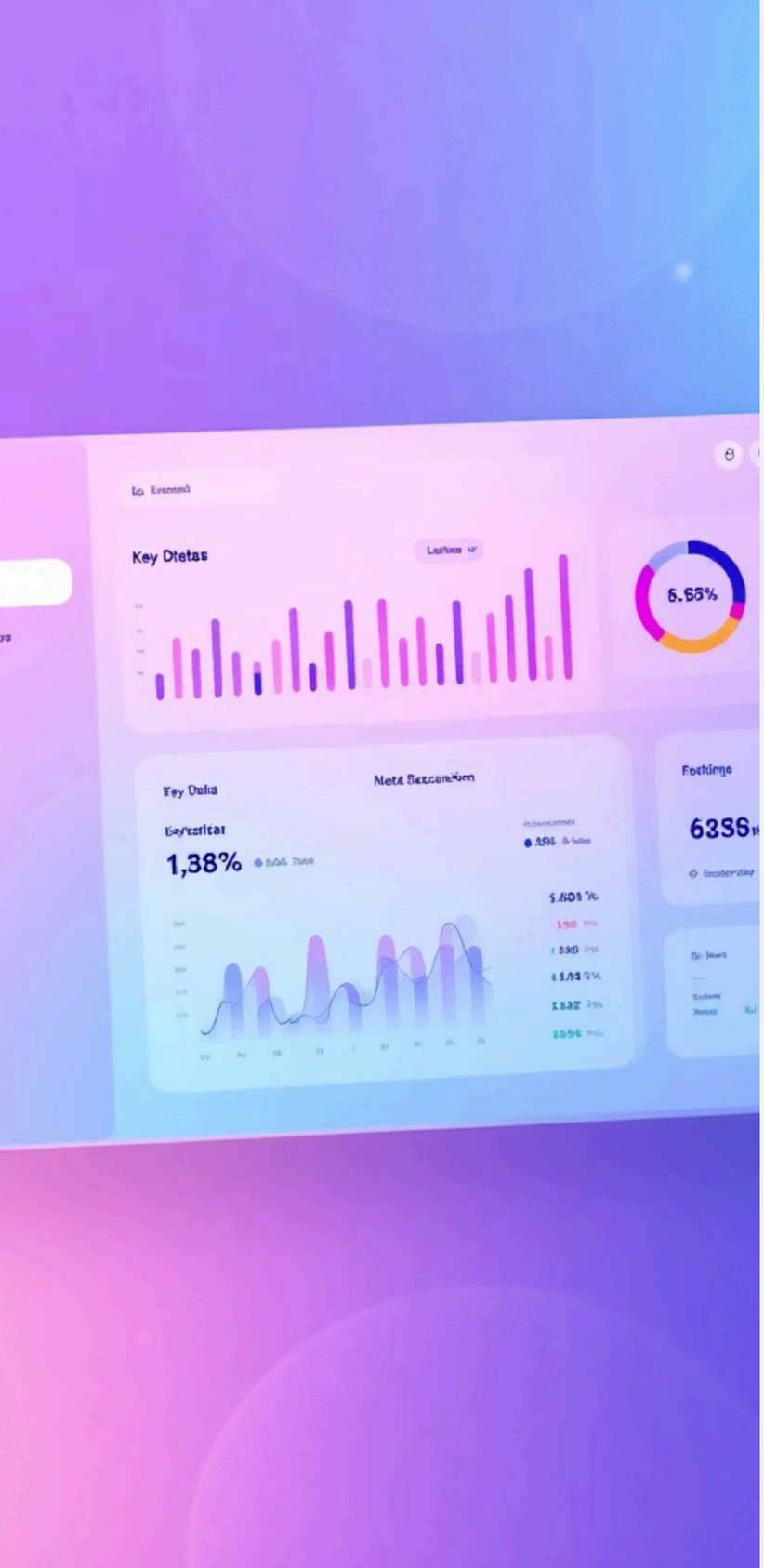
Definition

Fine-tuning involves training a pre-trained model further using your own domain-specific dataset, enabling it to adapt to your unique tasks, style, or industry requirements.

The Analogy

Pre-trained model: A general graduate with broad knowledge

Fine-tuning: Providing specialised professional training in law, medicine, finance, or customer support



Why Fine-Tuning Matters

Fine-tuning becomes essential when you need precision, consistency, and domain-specific intelligence that general models cannot provide.

Domain Expertise

Medical, legal, finance, or technical data requiring specialised knowledge

Task Specialisation

Classification, summarisation, translation, or sentiment analysis

Tone Consistency

Brand voice, call centre tone, or industry-specific communication style

Performance Benefits

Higher accuracy than prompting or RAG alone, with predictable outputs

Cost Efficiency

Lower inference costs with smaller, optimised custom models

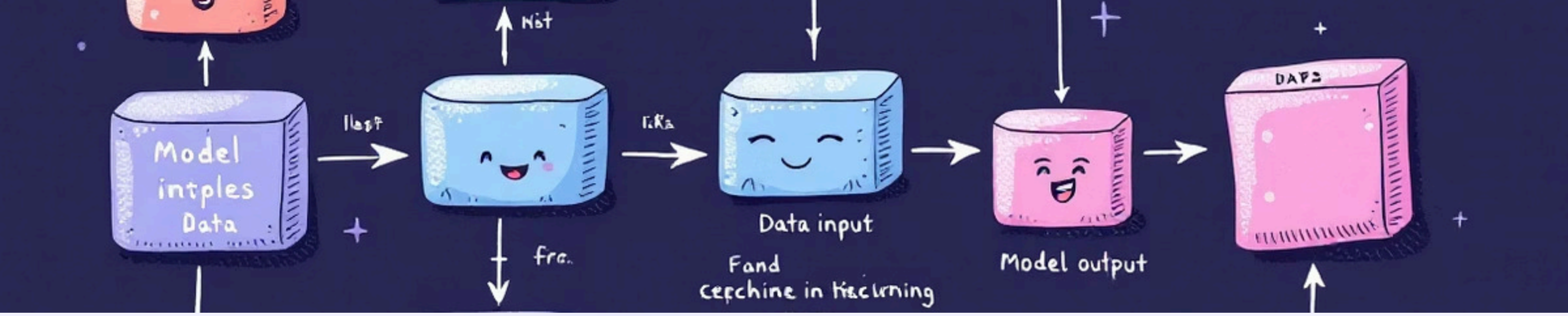
When to Fine-Tune Your Model

✓ Use Fine-Tuning When:

- You repeatedly prompt the model for the same task
- You want the model to learn entirely new behaviour patterns
- You need consistent and structured outputs
- RAG is insufficient (requires reasoning patterns or style adaptation)

▯ Avoid Fine-Tuning When:

- Task only needs factual retrieval — RAG is more suitable
- Data is small or inconsistent — prompt engineering may suffice
- Budget or compute resources are severely limited
- Rapid iteration and experimentation are priorities



Types of Model Customisation

Understanding the spectrum of fine-tuning approaches helps you select the right strategy for your use case, balancing performance, cost, and implementation complexity.

Full Fine-Tuning

Train all model parameters comprehensively

Partial Fine-Tuning

Freeze early layers, train task-specific layers

PEFT Methods

Parameter-efficient techniques like LoRA and adapters

Full Fine-Tuning

What It Is

Full fine-tuning involves training all parameters of the model from top to bottom, providing complete control over model behaviour and maximum adaptation capability.

Benefits

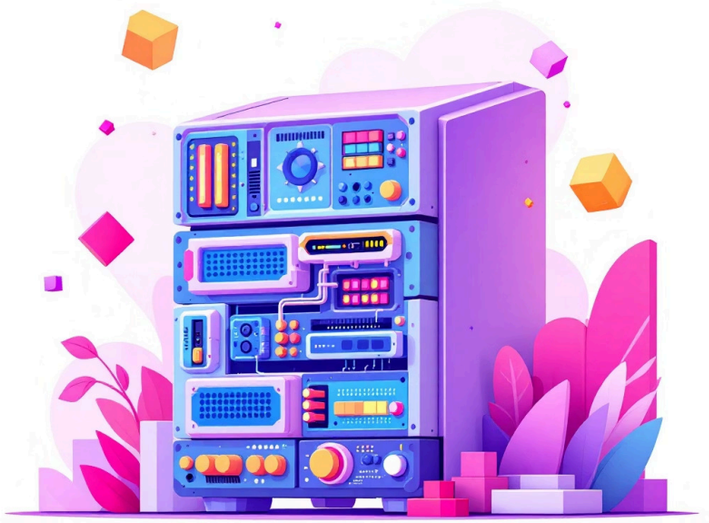
- Best overall performance and accuracy
- Learns new domain behaviour comprehensively
- Complete customisation of model outputs

Drawbacks

- Requires very large GPUs and memory
- Expensive and time-consuming training process
- Higher risk of overfitting on small datasets

When to Use

- Large labelled dataset available (10K+ examples)
- Need major behavioural change from base model
- Enterprise training pipelines with sufficient resources



Partial Fine-Tuning (Layer Freezing)

The Middle Ground Approach

Partial fine-tuning freezes early layers—which store general language knowledge—and trains only the last few layers that are task-specific. This balanced approach maintains core capabilities whilst enabling targeted adaptation.

Key Benefits

- Faster training compared to full fine-tuning
- Reduced compute requirements
- Preserves core language abilities
- Moderate adaptation flexibility

Ideal Scenarios

- Medium dataset size (1K-10K examples)
- Task resembles pre-training tasks
- Need moderate domain adaptation
- Limited but adequate GPU resources

PEFT: Parameter-Efficient Fine-Tuning

The Modern Standard

PEFT represents the most popular fine-tuning approach today. It trains only a small set of additional parameters whilst keeping the base model unchanged, delivering exceptional efficiency.

95%

Cost Reduction

Compared to full fine-tuning approaches

1

GPU Needed

Can fine-tune huge models on single consumer GPU

∞

Adapters

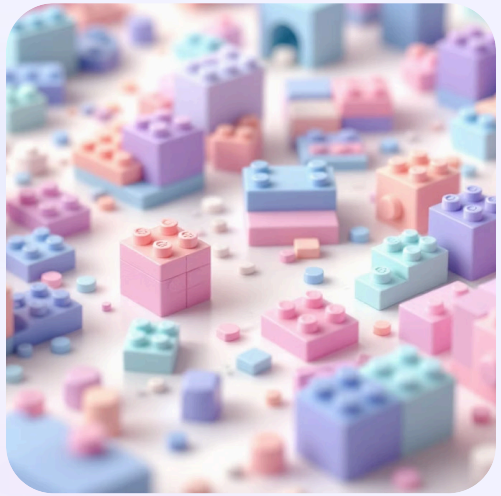
Store multiple adapters for different use cases

> PEFT democratises fine-tuning, making advanced model customisation accessible to organisations of all sizes.

Types of PEFT Methods

LoRA (Low-Rank Adaptation)

Injects low-rank matrices into model layers. Very cost-effective and represents state-of-the-art performance. Most widely adopted PEFT method.



Adapters

Small neural modules inserted between layers. You can load and remove adapters per task, enabling multi-task serving from one base model.

Prefix / Prompt Tuning

Trains only task-specific virtual tokens prepended to inputs. Best suited for instruction following and style adaptation tasks.

Choosing the Right Method

Select your fine-tuning approach based on dataset size, compute budget, and customisation requirements. Each method offers distinct trade-offs between performance, cost, and implementation complexity.

Full Fine-Tuning	💸💸💸	New behaviour, huge datasets, complete customisation	Low compute environments, small datasets
Partial Fine-Tuning	💸💸	Moderate customisation, balanced approach	High novelty tasks, extreme adaptation
PEFT (LoRA, Adapters)	💸	Most tasks, small datasets, cost efficiency	Large structural changes, radical behaviour shifts
Prompt Tuning	💰	Style/tone tasks, instruction following	Deep reasoning changes, complex adaptations