

# 5 Techniques to fine-tune LLMs

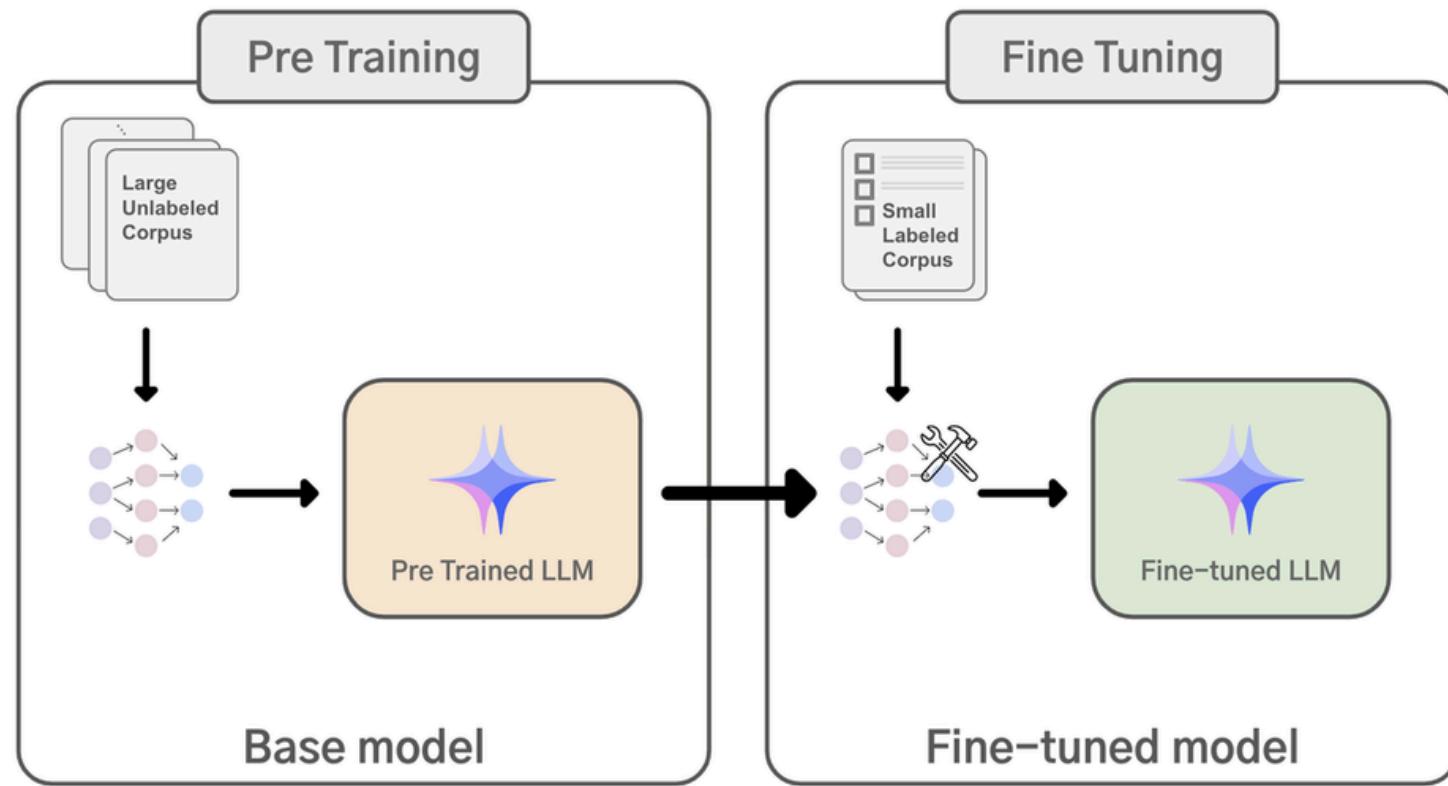


blog.DailyDoseofDS.com



# Finetuning LLM Dictionary

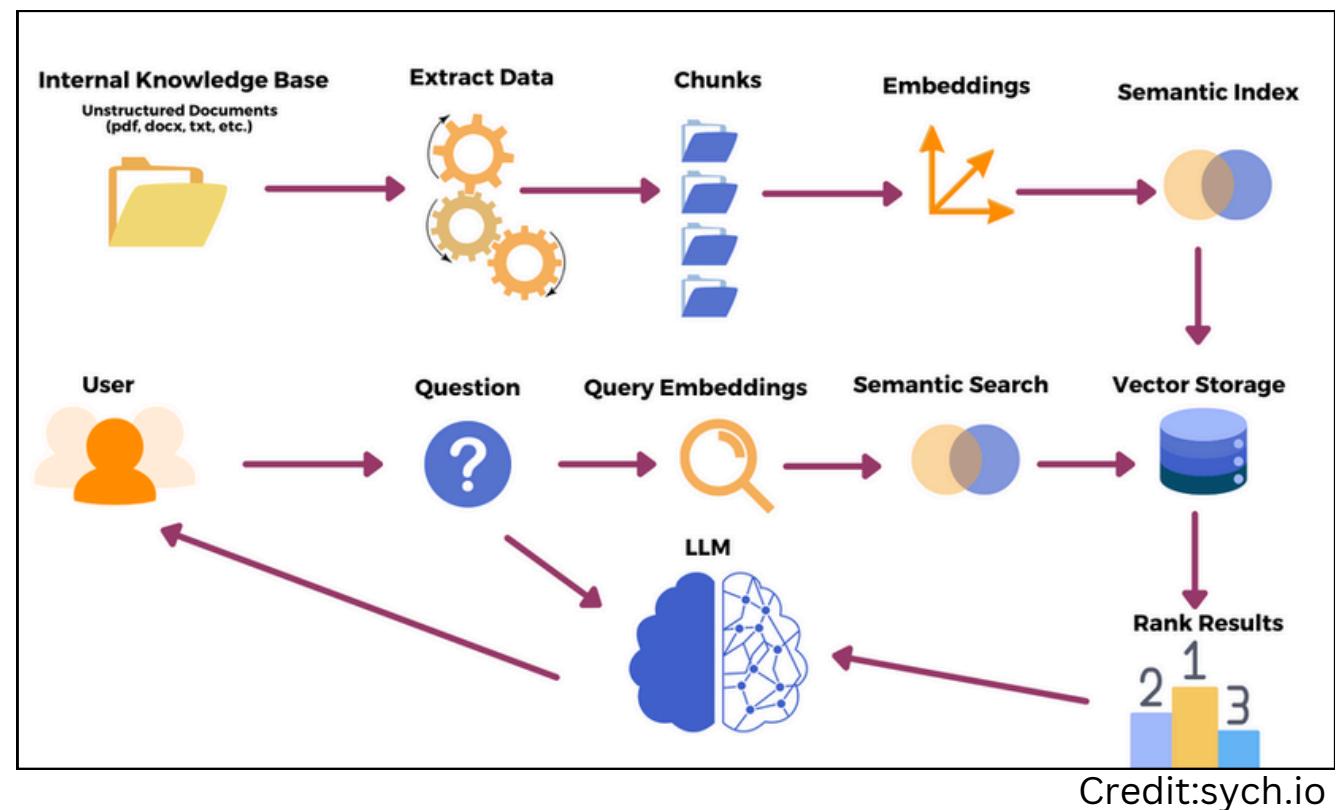
Your A-to-Z Guide to Mastering LLM Finetuning



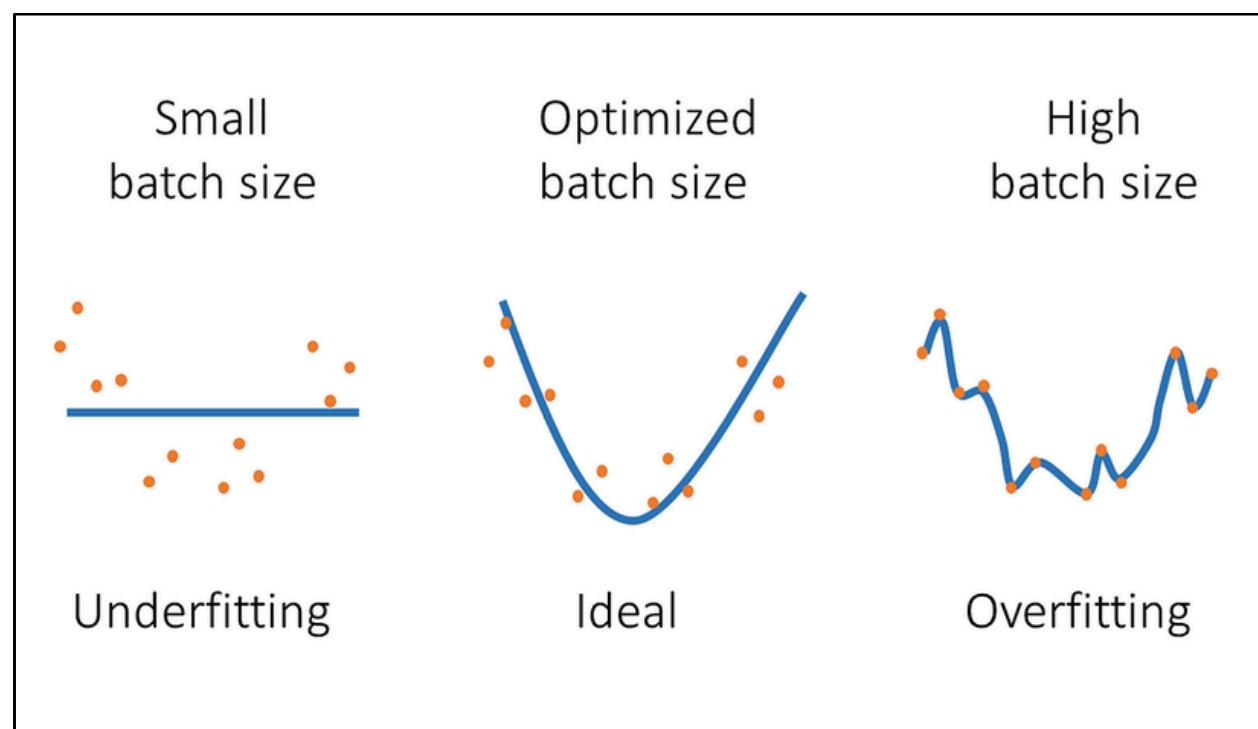
# A - Augmentation

Expanding model abilities with external data or techniques.

- Ensures better outputs by adding domain knowledge.
- Enhances reliability in fine-tuned LLMs.
- Common in adaptive learning systems.



# B - Batch Size



The number of samples processed before updating model weights.

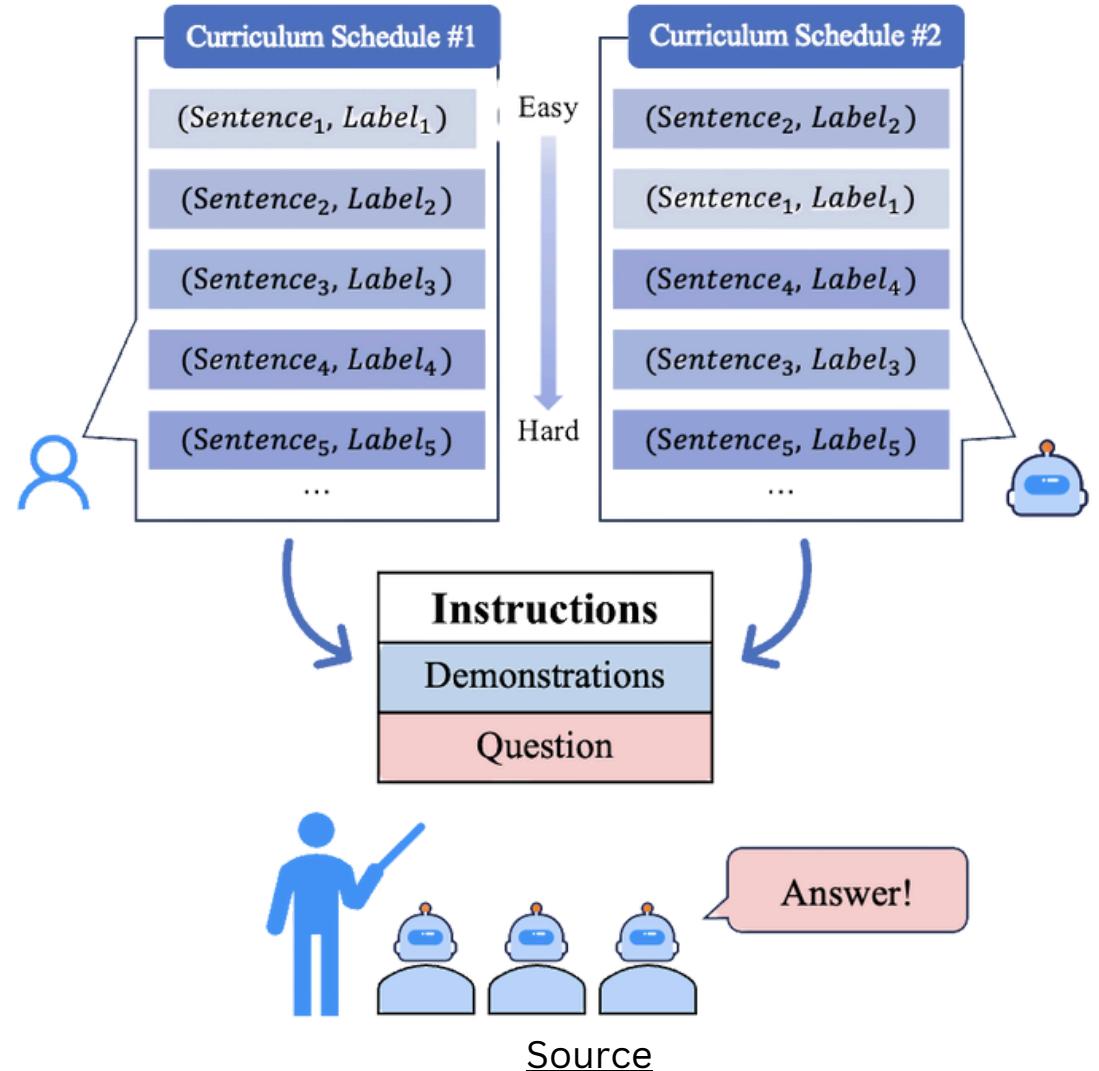
- Affects learning stability and speed.
- Balancing small and large batch sizes is key.
- Important for tuning computational efficiency.

[Source](#)

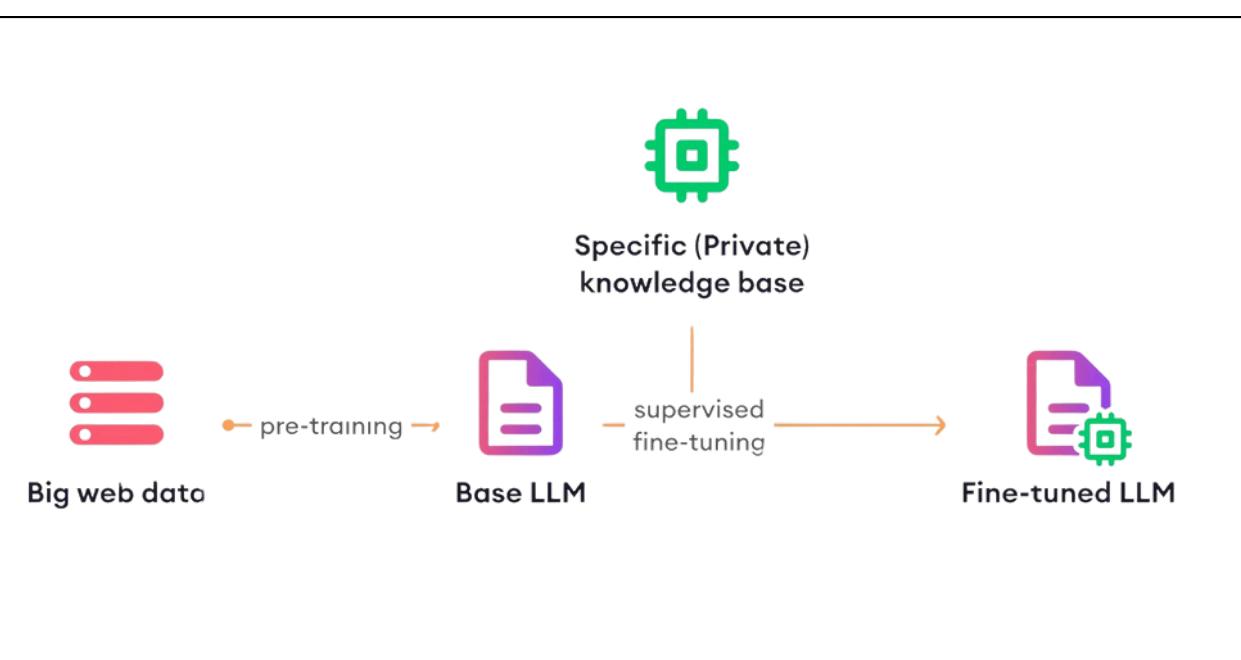
## C - Curriculum Learning

Training models step-by-step with increasing complexity.

- Speeds up convergence.
- Helps the model generalize better to tasks.
- Inspired by human learning systems.



## D - Domain-Specific Tuning



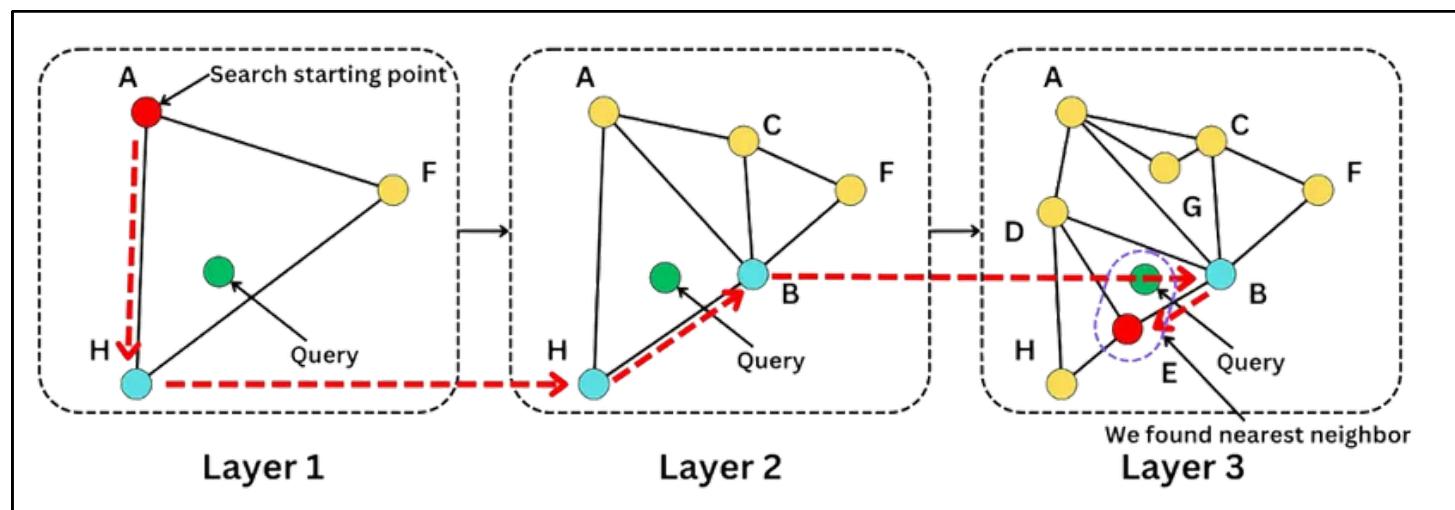
Adapting LLMs to excel in specialized fields.

- Essential for medical, legal, or finance sectors.
- Improves performance in niche applications.
- Requires high-quality labeled datasets.

## E - Embeddings

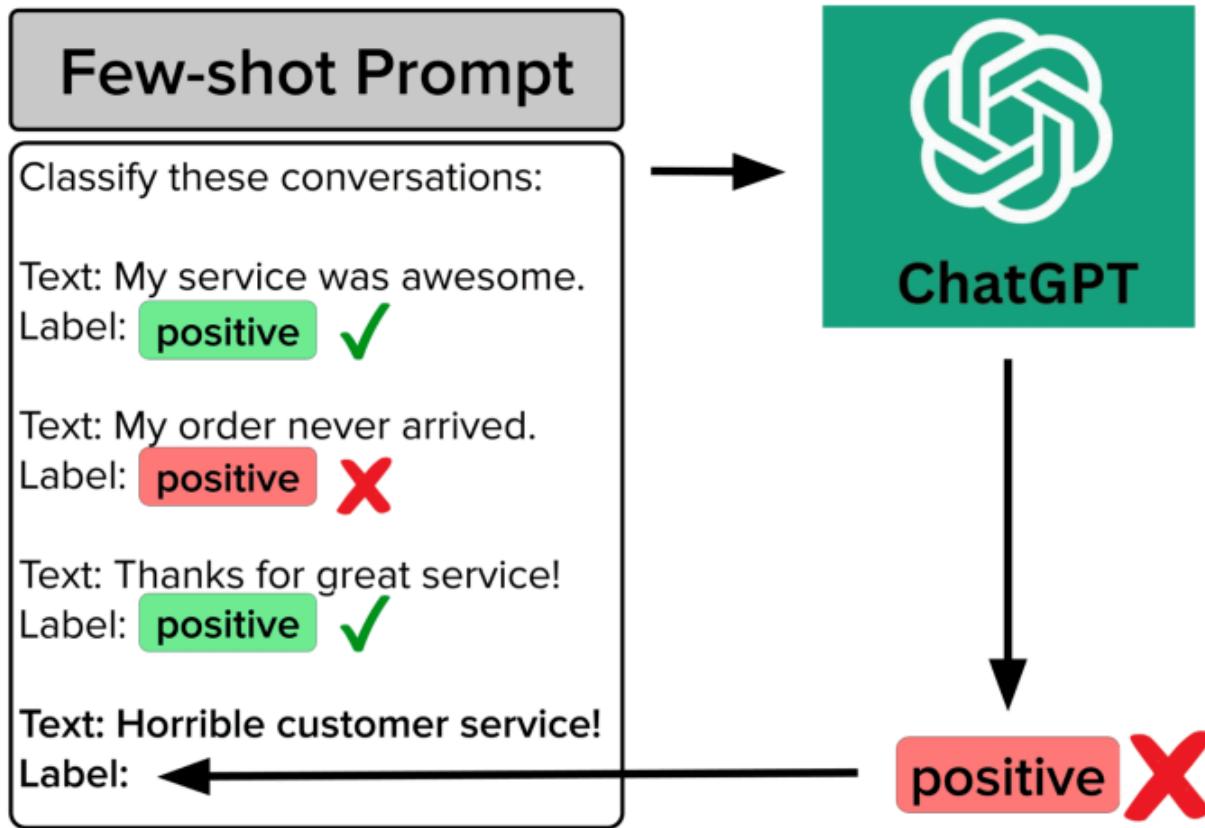
Converting text into numerical vectors for analysis.

- Key for semantic search & dense retrieval.
- Used extensively in fine-tuning pipelines.
- Powers tasks like recommendations and clustering.



Credit:Community.aws

## F - Few-Shot Learning



Fine-tuning with minimal labeled data.

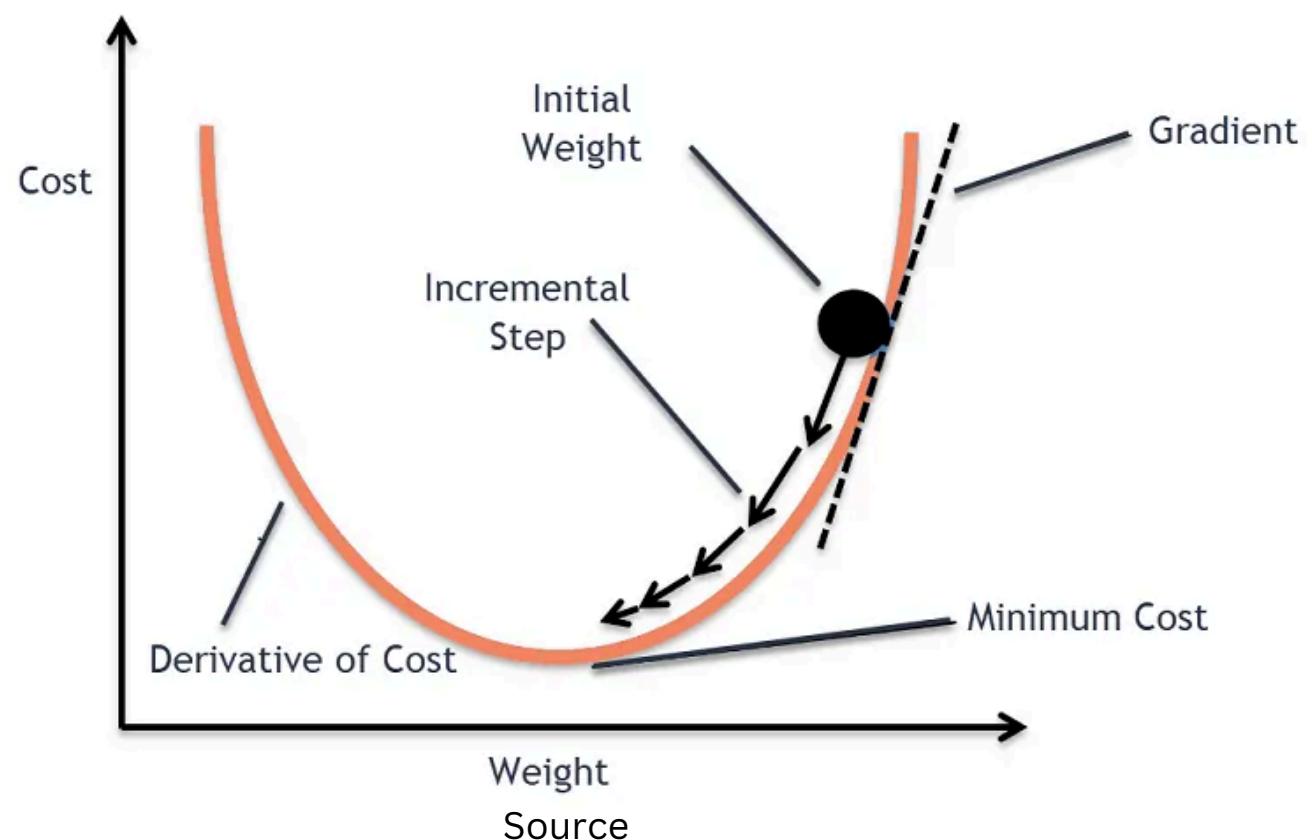
- Enables rapid adaptation to new tasks.
- Helps in low-data scenarios.
- Balances pre-training and specialized knowledge.

[Source](#)

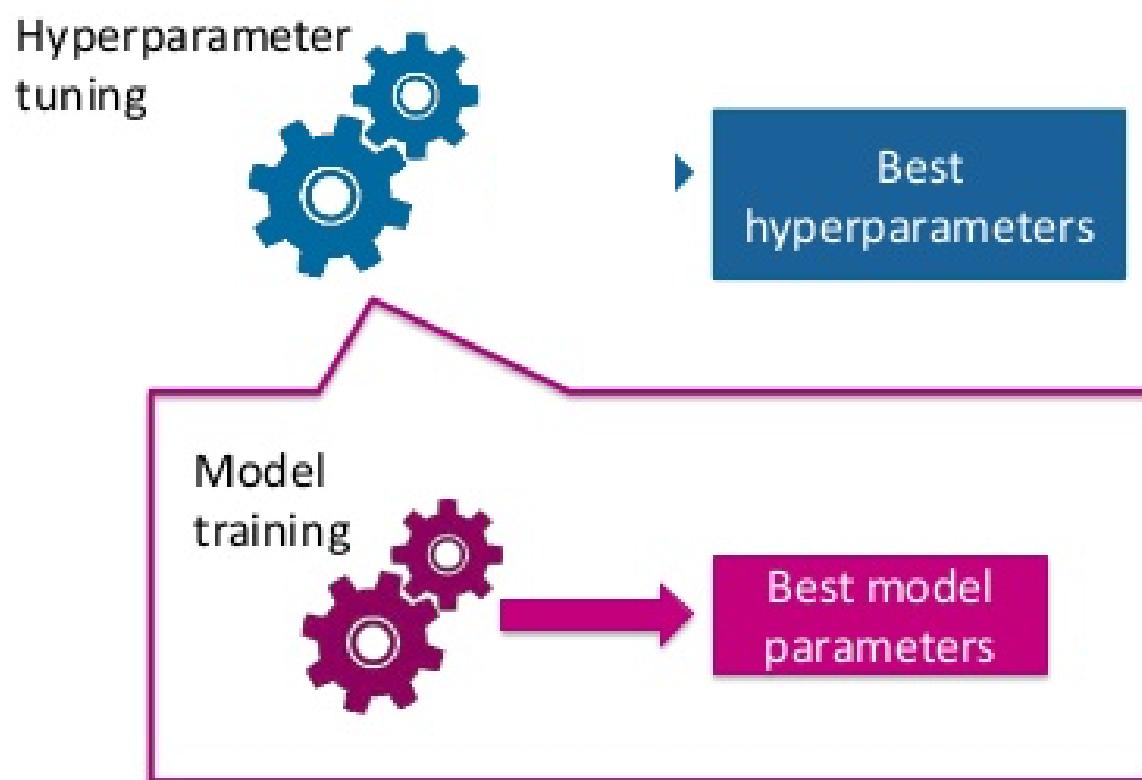
## G - Gradient Descent

Optimization method to minimize error in training.

- Drives the learning process by adjusting weights.
- Central to every fine-tuning process.
- Requires proper tuning of learning rates.



## H - Hyperparameters



Settings like batch size, learning rate, and epochs.

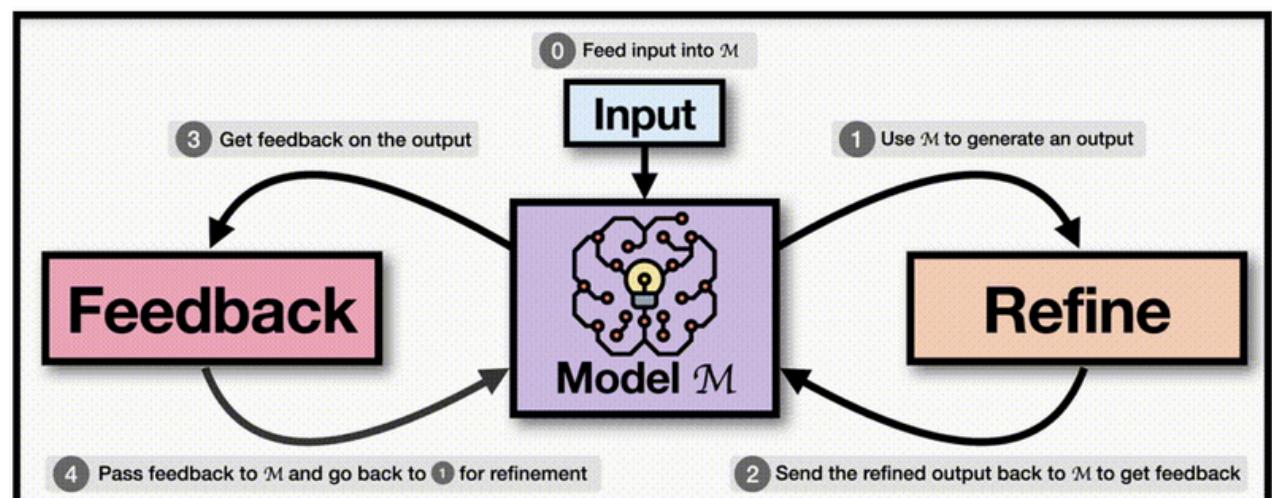
- Fine-tuning requires precise hyperparameter optimization.
- Impacts model accuracy and speed.
- Experimentation helps achieve optimal results.

Source

# I - Iterative Training

Refining models through repeated training cycles.

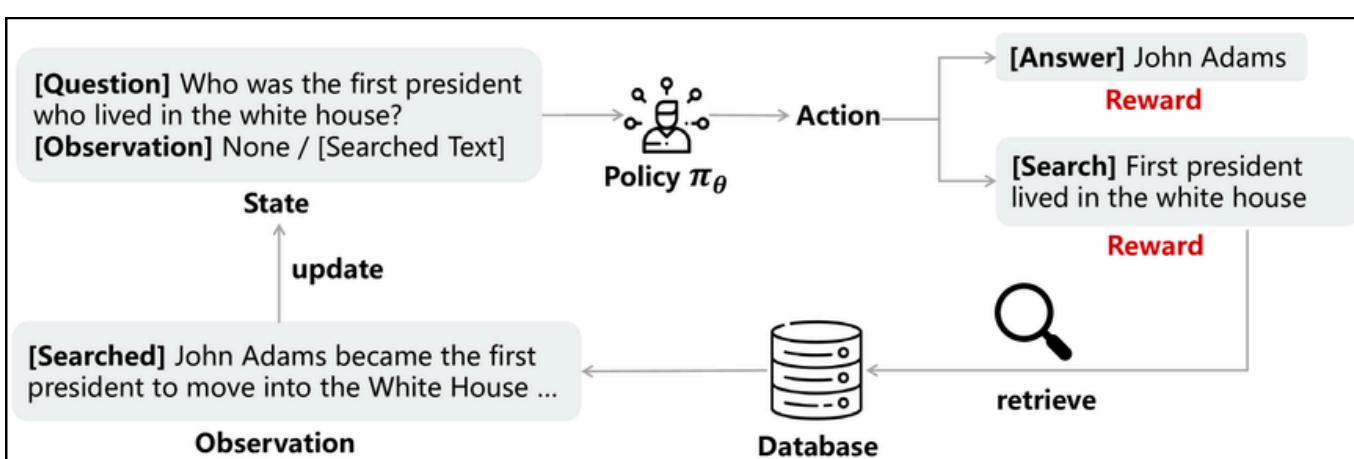
- Enhances performance step by step.
- Key for achieving state-of-the-art results.
- Reduces overfitting by monitoring progress.



Source

# J - Joint Learning

Trains retrieval and generation components simultaneously.



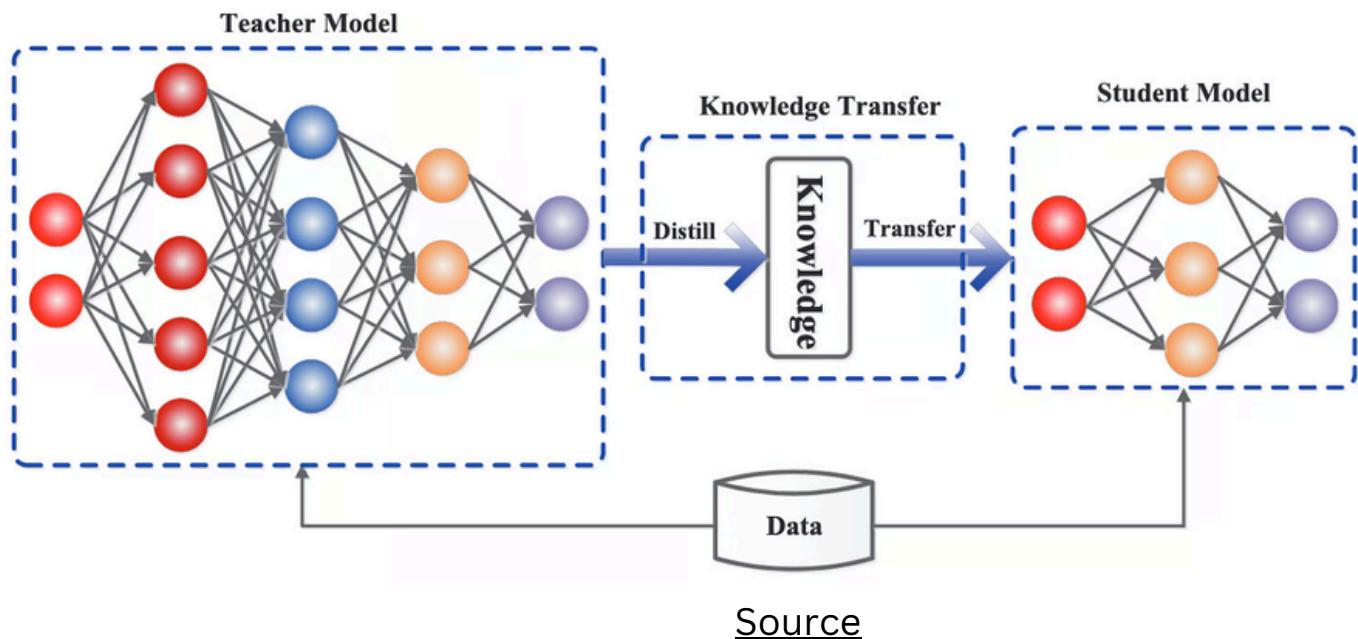
Credit:arxiv.org

- Ensures better synergy between retrieved data and generated outputs.
- Reduces the need for separate fine-tuning.
- Often leads to improved overall system performance.

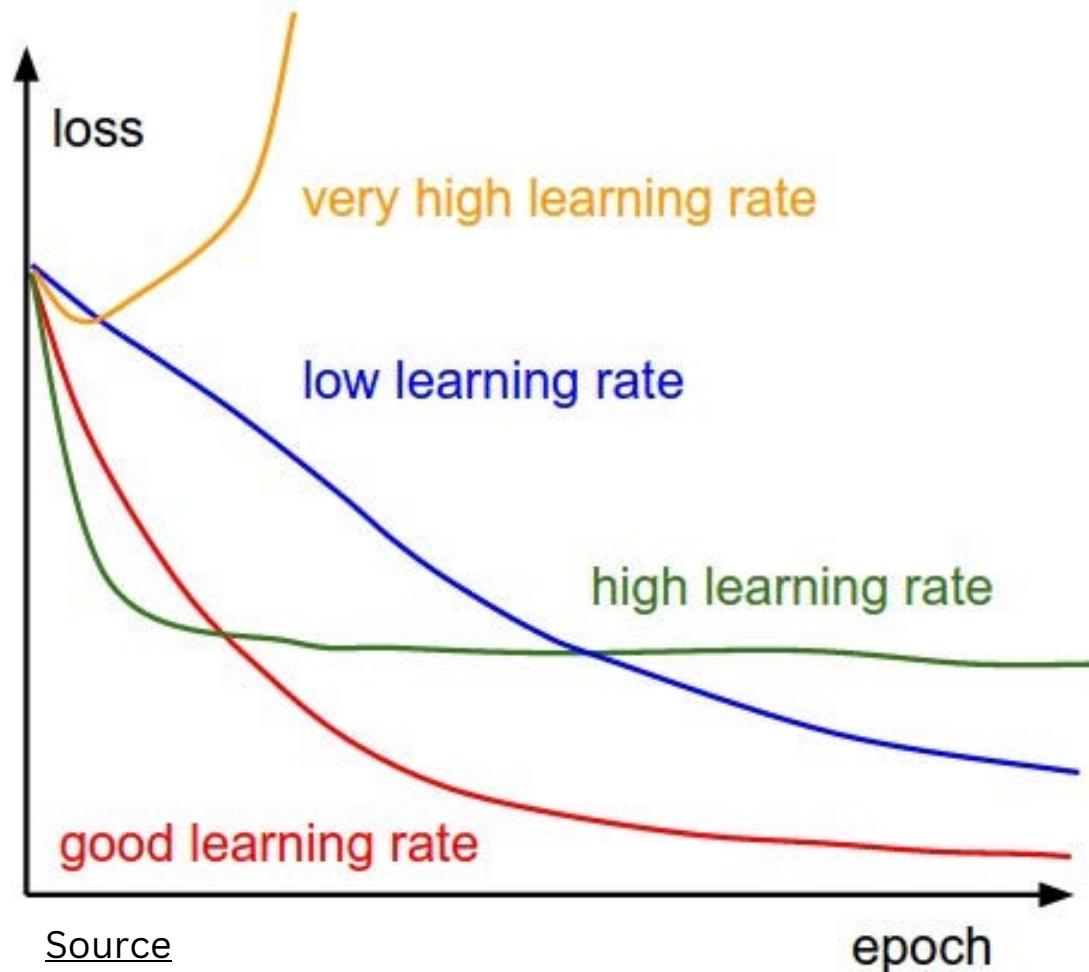
# K - Knowledge Distillation

Transferring knowledge from large models to smaller ones.

- Makes models more efficient for deployment.
- Retains essential capabilities with fewer resources.
- Common in low-resource environments.



# L - Learning Rate



The speed at which the model learns.

- Critical to balance between slow and fast learning.
- Improper tuning can lead to overfitting or underperformance.
- Often adjusted dynamically during training.

# M - Model Weights

Parameters learned during training.

- Define how the model processes input.
- Fine-tuning adjusts weights for specific tasks.
- Pretrained weights act as a foundation.

## Finetuned Weights

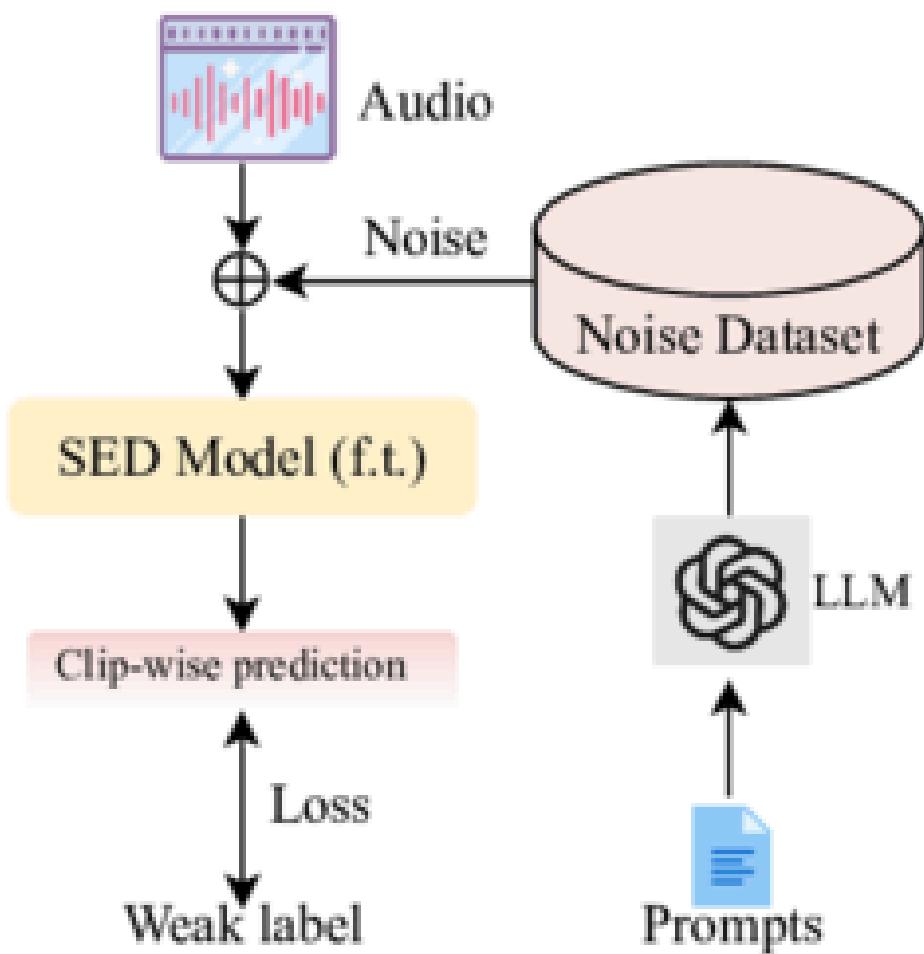
$$W_{\text{ft}} = W_{\text{pt}} + \Delta W$$

## Pretrained Weights

[Source](#)

## Weight Update

# N - Noise Handling



Addressing noisy or low-quality data in fine-tuning.

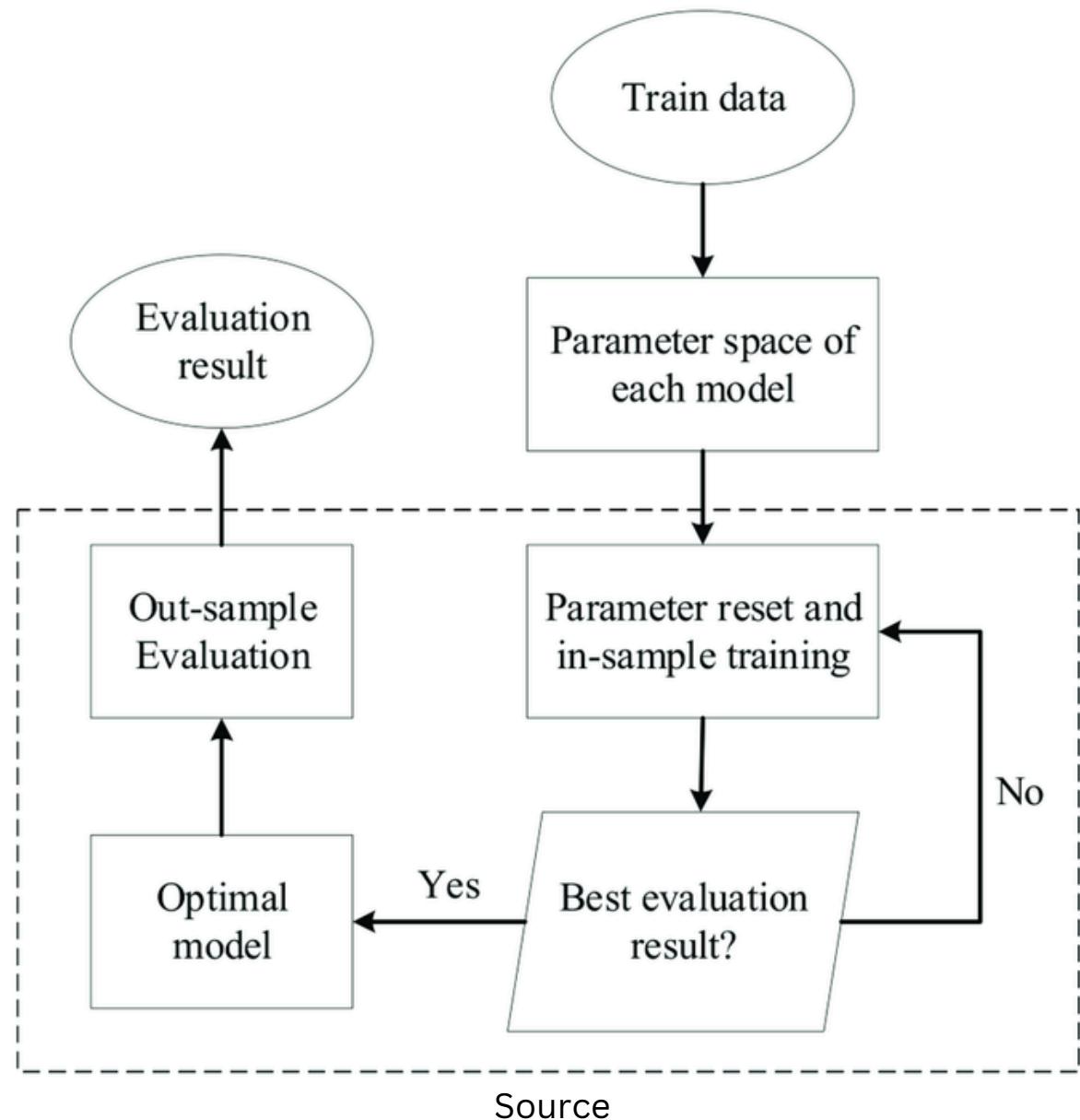
- Improves the model's ability to generalize.
- Requires robust preprocessing pipelines.
- Common in messy, real-world datasets.

[Source](#)

# O - Optimization

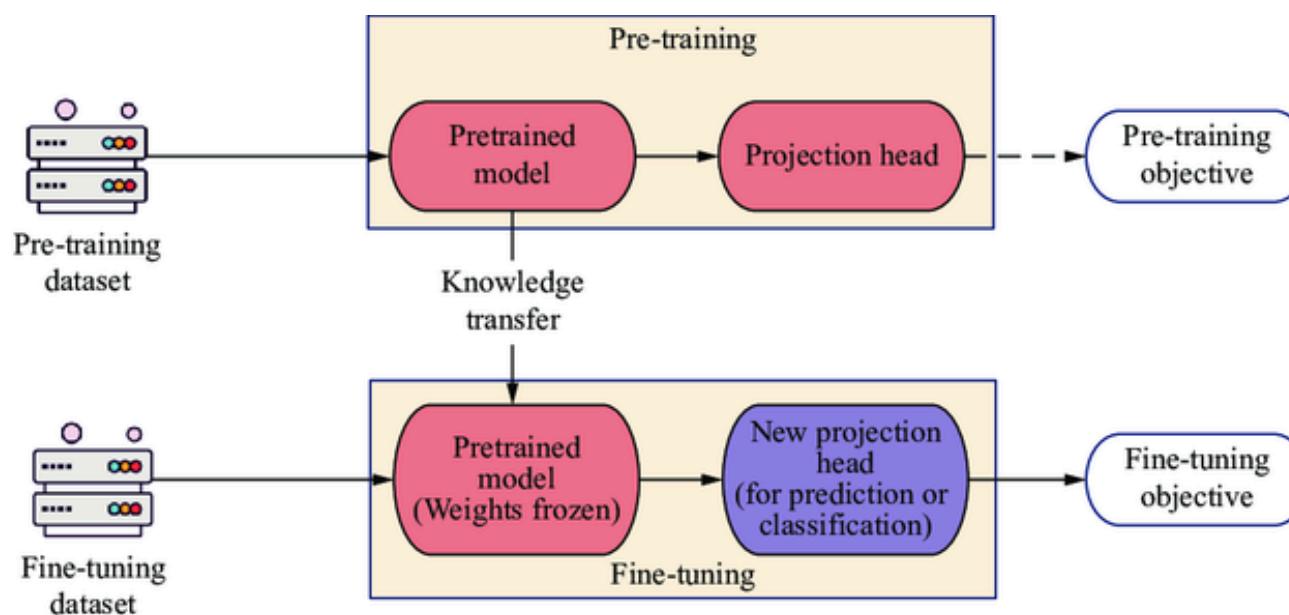
The process of improving model performance.

- Involves methods like Adam or SGD.
- Essential for faster convergence in fine-tuning.
- Balances trade-offs between accuracy and efficiency.



[Source](#)

# P - Pretraining



Training on large datasets before fine-tuning.

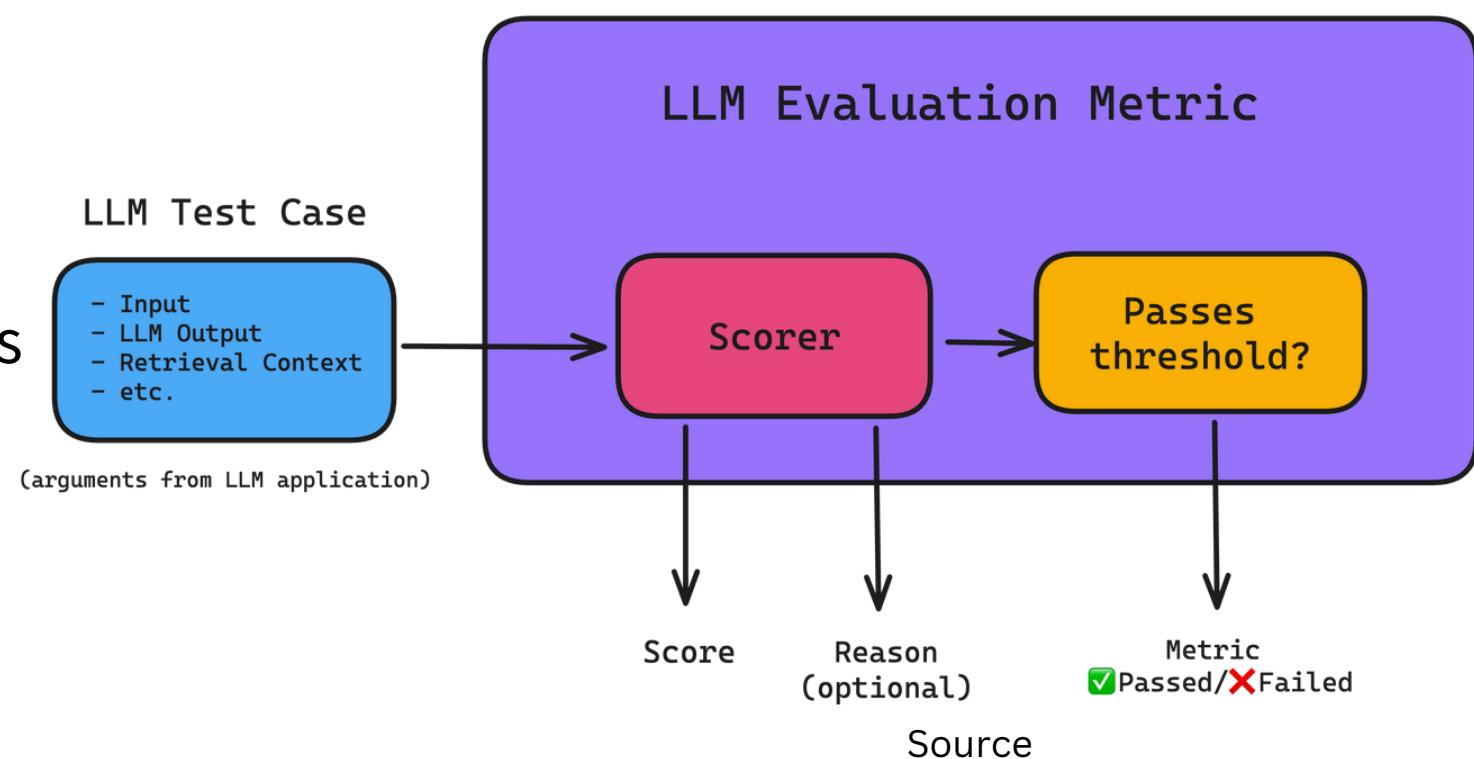
- Provides general knowledge to the model.
- Reduces data requirements for fine-tuning.
- Speeds up task-specific adaptation.

[Source](#)

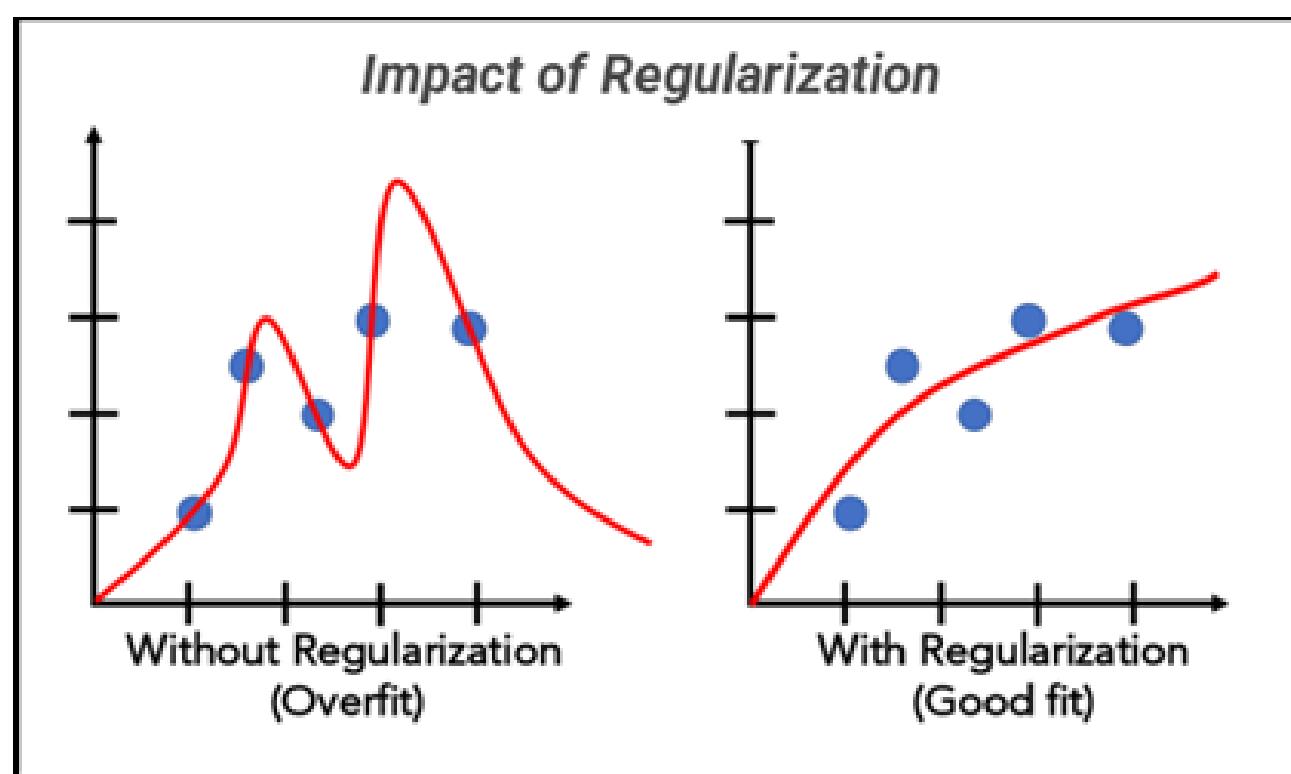
# Q - Quality Evaluation

Measuring performance with metrics like BLEU, ROUGE, or accuracy.

- Validates the effectiveness of fine-tuning.
- Guides iterative improvements.
- Helps identify issues like overfitting.



# R - Regularization



Techniques to prevent overfitting.

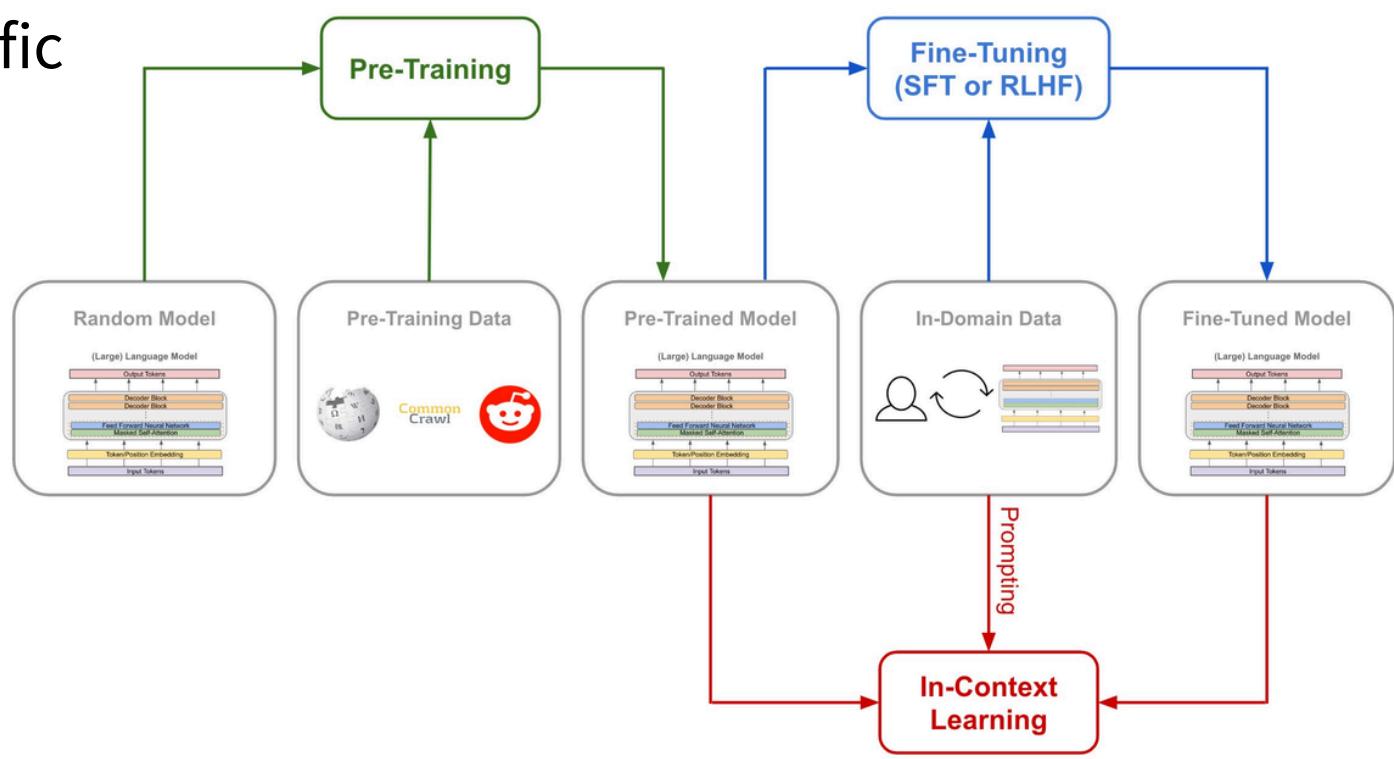
- Includes dropout, weight decay, or early stopping.
- Improves generalization on unseen data.
- Key for robust fine-tuned models.

[Source](#)

# S - Supervised Fine-Tuning

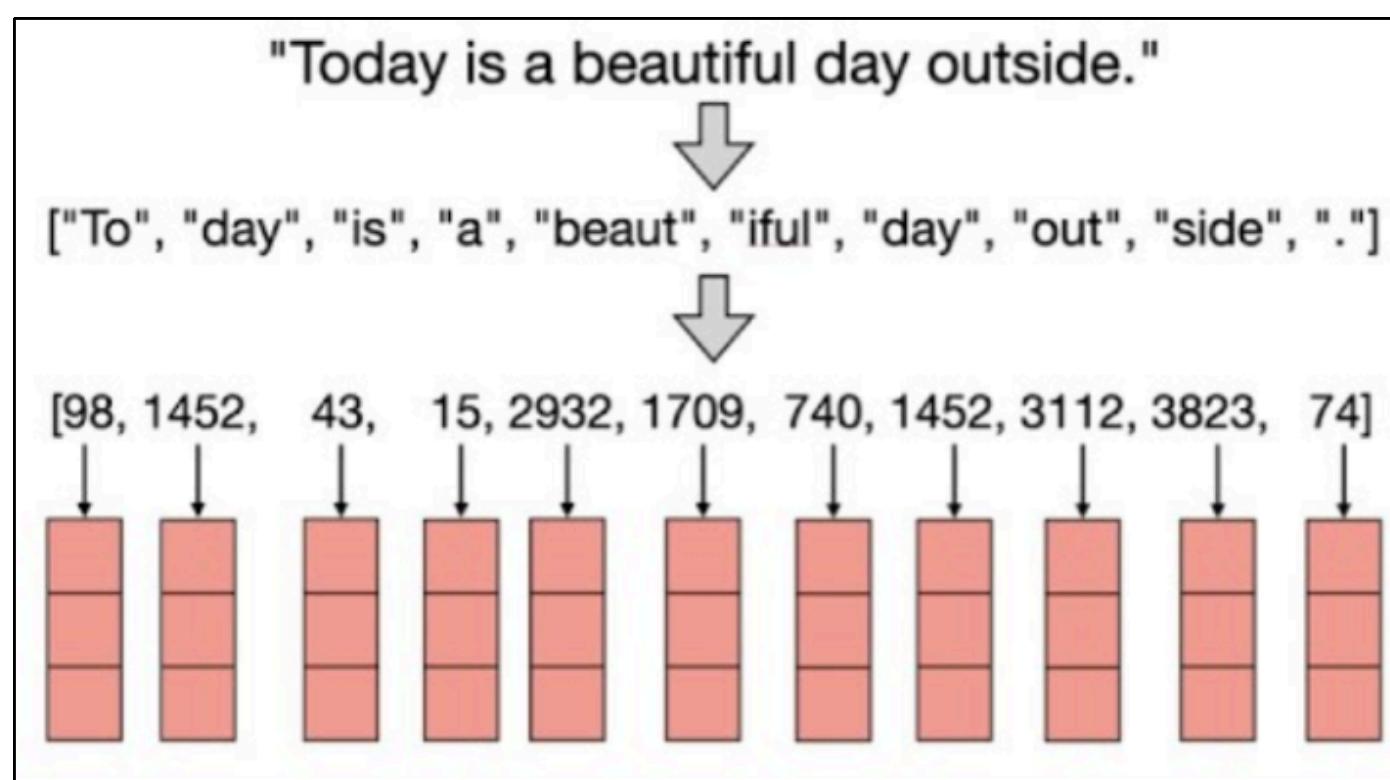
Using labeled data to teach specific tasks.

- Boosts performance in well-defined use cases.
- Requires high-quality annotations.
- Common in domain-specific applications.



[Source](#)

# T - Tokenization



Splitting text into smaller units (tokens).

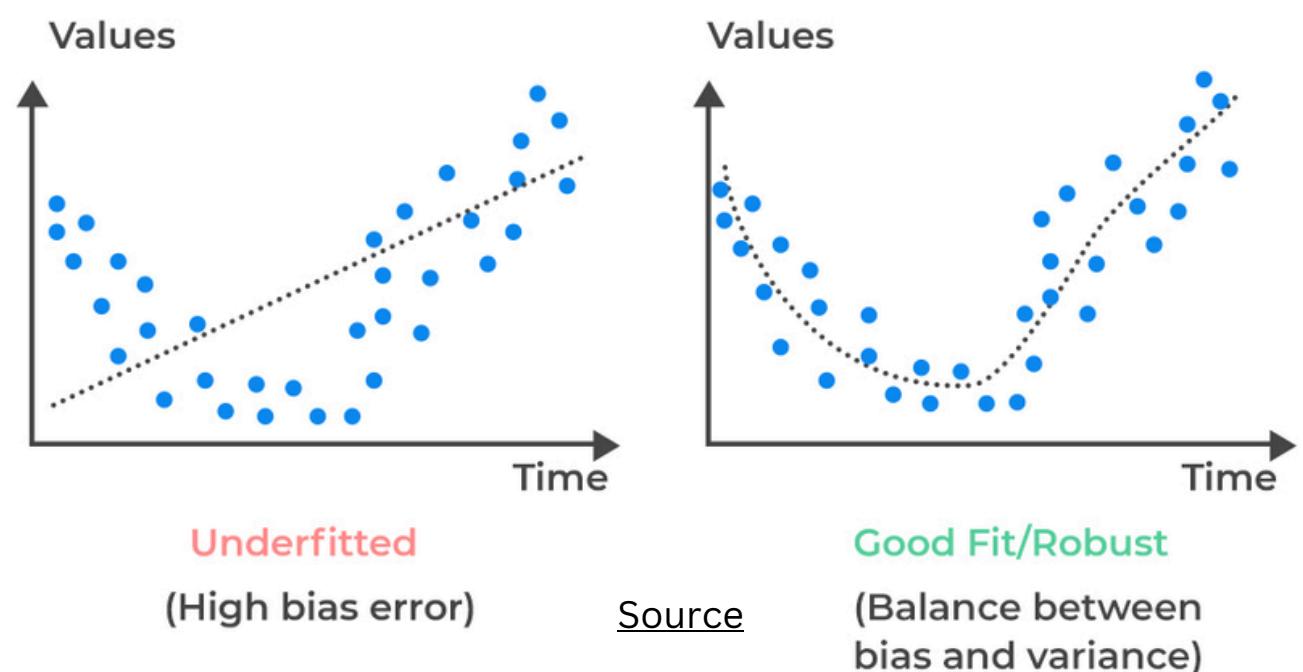
- Prepares data for model consumption.
- Handles variations like punctuation and casing.
- Fundamental for both training and inference.

Credit:Youtube.com

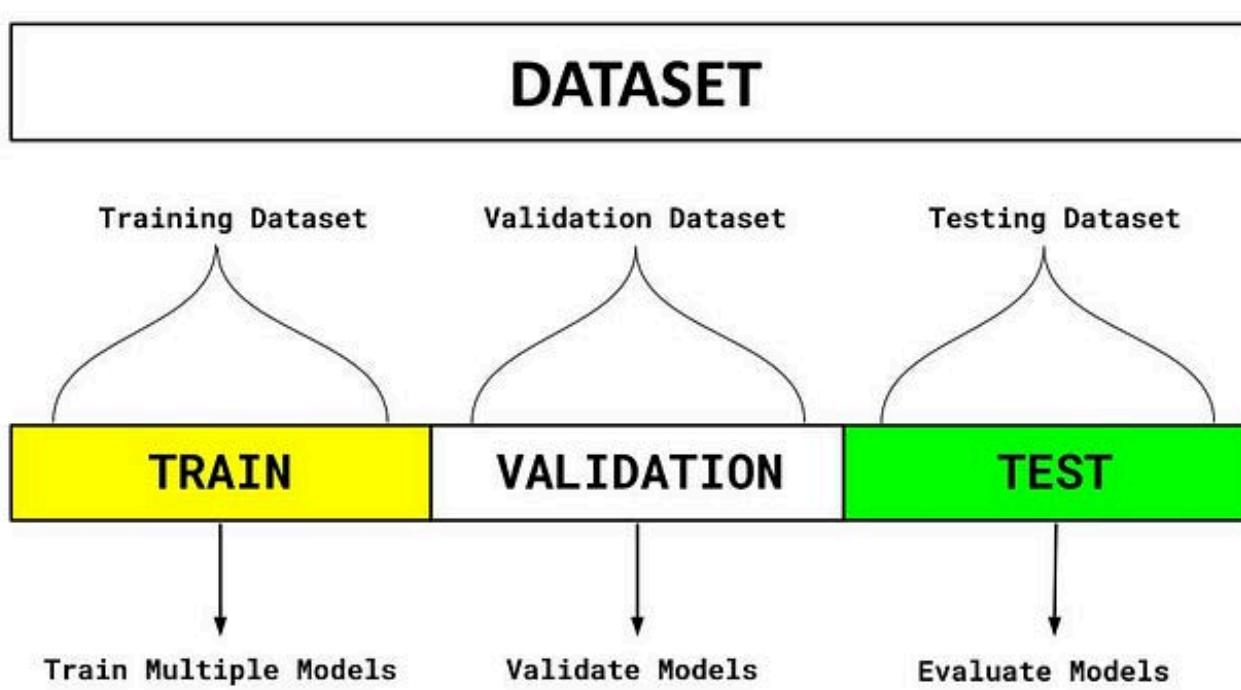
# U - Underfitting

A model failing to capture data patterns.

- Often due to insufficient training.
- Addressed by increasing complexity or data size.
- Opposite of overfitting.



# V - Validation Set



Dataset used to monitor training performance.

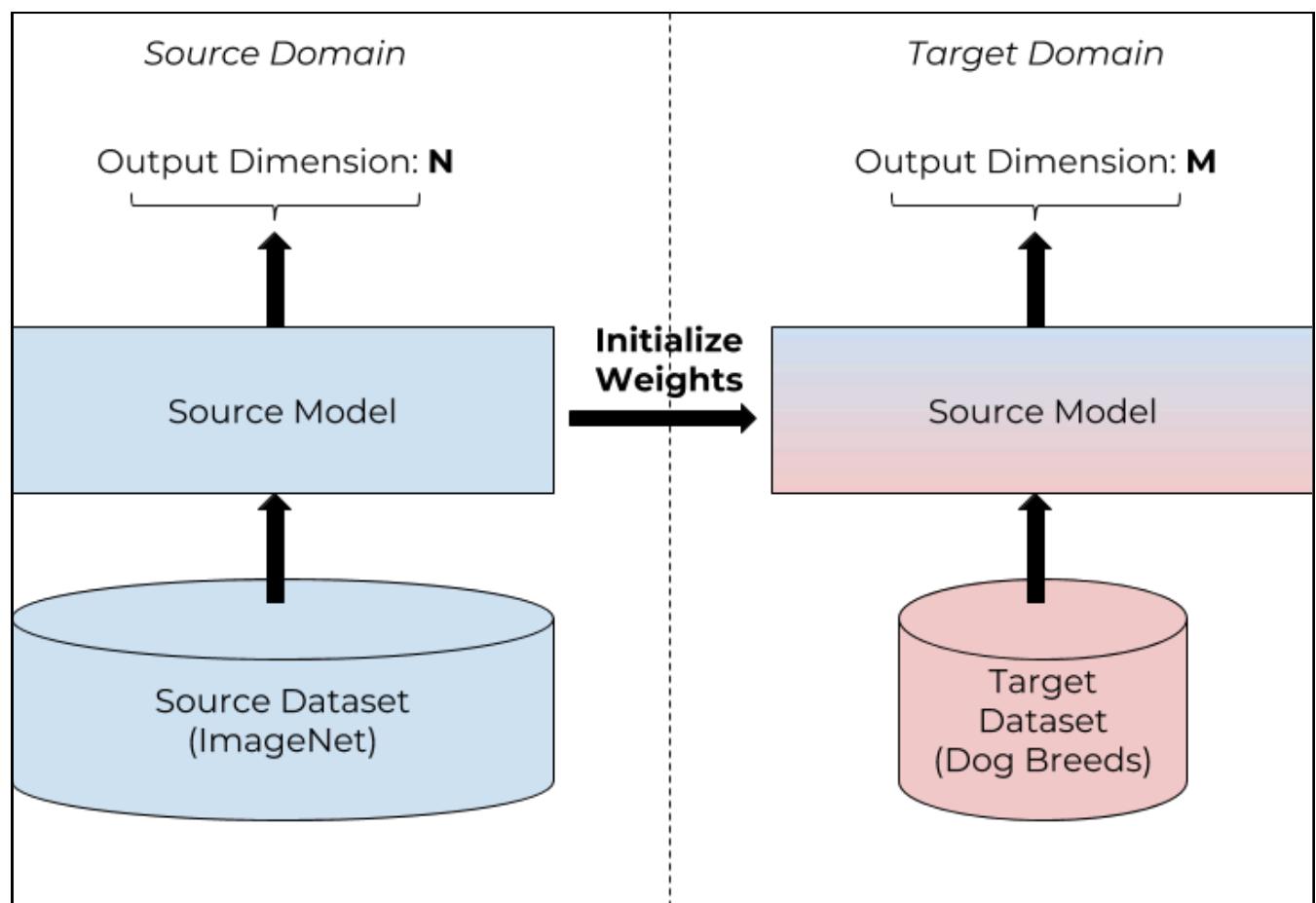
- Helps prevent overfitting during fine-tuning.
- Guides decisions on hyperparameter tuning.
- Ensures the model generalizes well.

Source

## W - Warm-Start

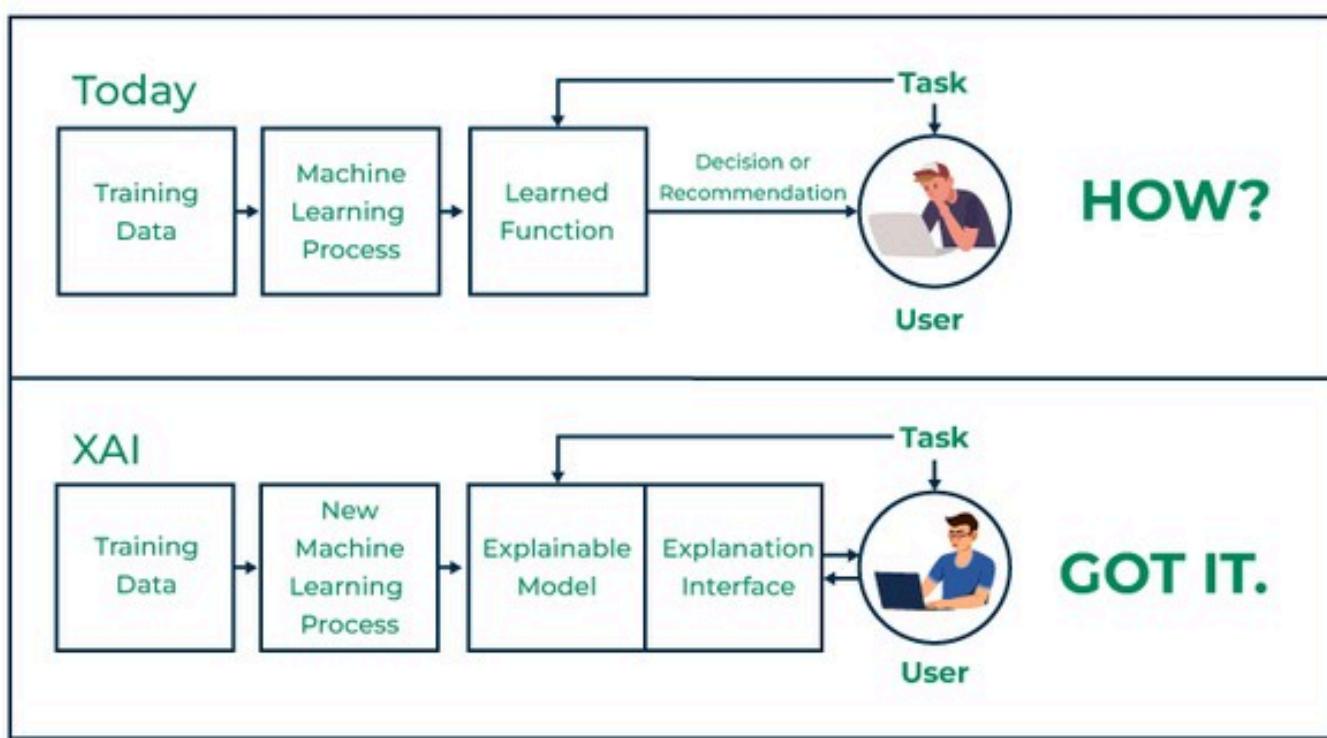
Initializes retrieval systems with pre-trained embeddings or models.

- Speeds up convergence and improves early-stage performance.
- Reduces training time for new tasks.
- Common in transfer learning scenarios.



Credit:determined.ai

## X - Explainability



Understanding model predictions.

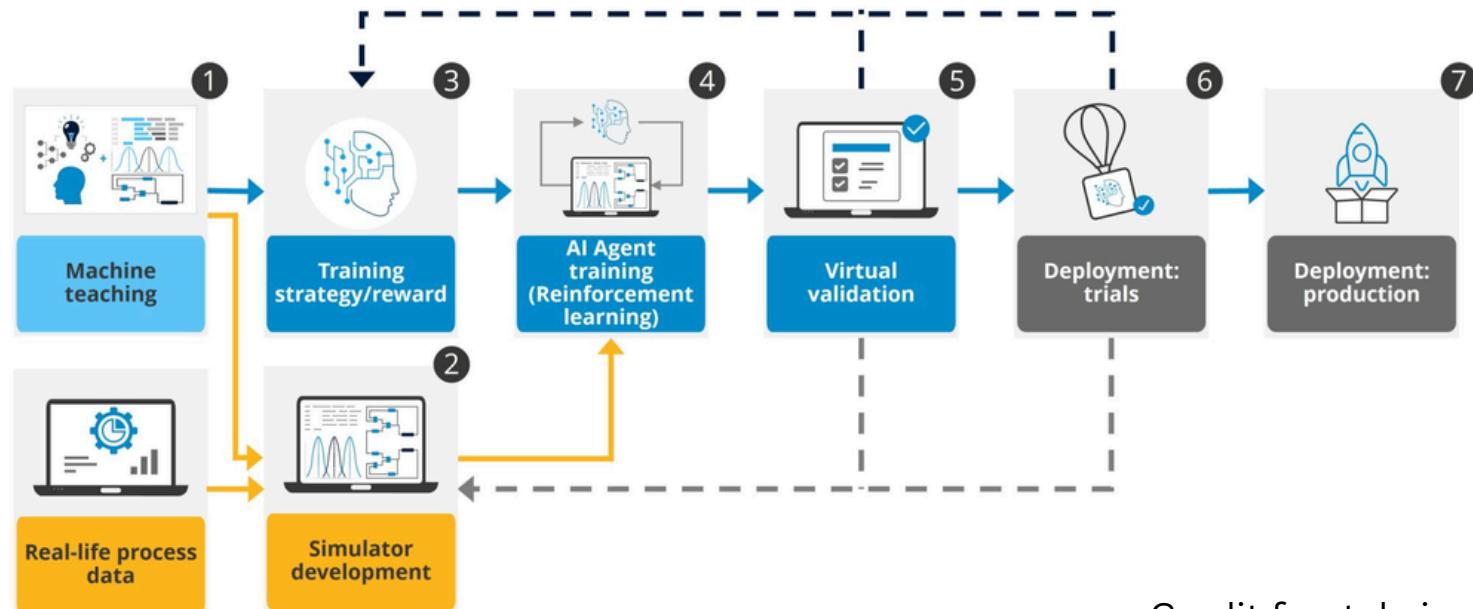
- Essential for building trust in AI systems.
- Identifies biases or issues in fine-tuned outputs.
- Key for high-stakes applications like healthcare.

Credit:Medium.com

# Y - Yield Optimization

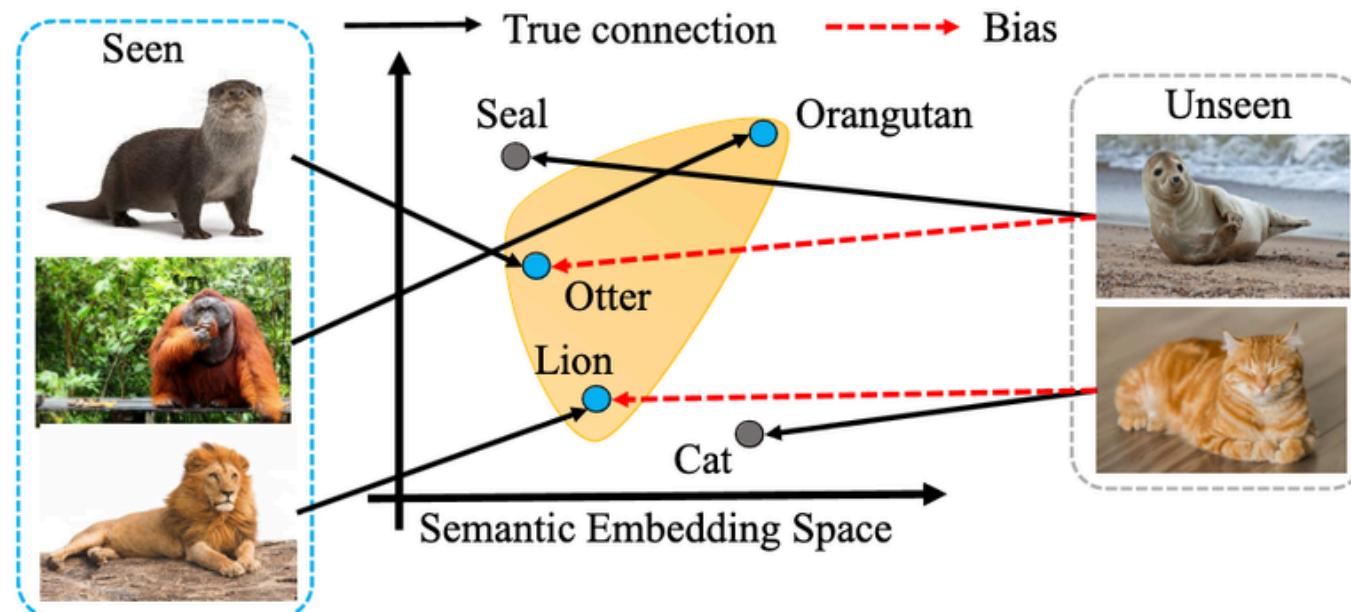
Maximizing output relevance and efficiency.

- Improves response quality for fine-tuned models.
- Involves iterative adjustments and monitoring.
- Enhances user satisfaction in real-world use.



Credit:fractal.ai

# Z - Zero-shot learning



Performing tasks without task-specific fine-tuning.

- Leverages general knowledge from pretraining.
- Useful for quick adaptation to new domains.
- A hallmark of advanced LLMs.

[Source](#)



**Follow to stay updated on  
Generative AI**



**SAVE**



**LIKE**



**REPOST**