Fine Tuning LLMs

Agenda

- Introduction to Fine Tuning
- Prompt Tuning
- Parameter Efficient Fine Tuning
- Adapter Layers
- LoRA

Great Learning
POWER AHEAD

Let’s begin the discussion by answering a few questions.

Fine Tuning Quiz

Which factors are critical decision points when considering whether to fine-tune a pre-trained model?

A. The presence of labeled data and the initial model performance on the target task.

B. The size of the pre-trained model and the speed of the training process.

C. The complexity of the model architecture and the number of available GPUs

D. The availability of high-performance computing resources and the total number of parameters in the model.

Fine Tuning Quiz

Which factors are critical decision points when considering whether to fine-tune a pre-trained model?

A. The presence of labeled data and the initial model performance on the target task.

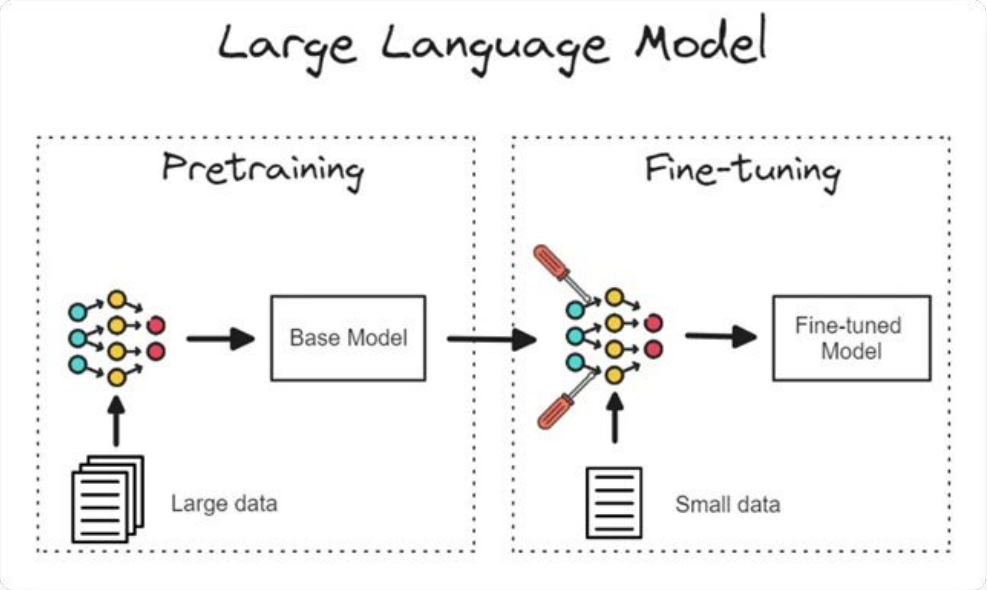
B. The size of the pre-trained model and the speed of the training process.

C. The complexity of the model architecture and the number of available GPUs

D. The availability of high-performance computing resources and the total number of parameters in the model.

Introduction to Fine Tuning

- The base model performance on a specific task establishes a baseline to refer to
- If the baseline model underperforms on the task, fine-tuning might yield improvements
- High-quality labeled data is critical for successful fine-tuning
- Adequate quantity of labeled examples is essential for effective learning of the target task



Fine Tuning Quiz

What is the main idea behind prompt tuning in large language models?

A. Updating all the model weights during fine-tuning

B. Adding small trainable vectors to the input to guide the model without changing most weights

C. Compressing the model to reduce memory usage

D. Manually crafting input instructions and examples

Fine Tuning Quiz

What is the main idea behind prompt tuning in large language models?

A. Updating all the model weights during fine-tuning

B. Adding small trainable vectors to the input to guide the model without changing most weights

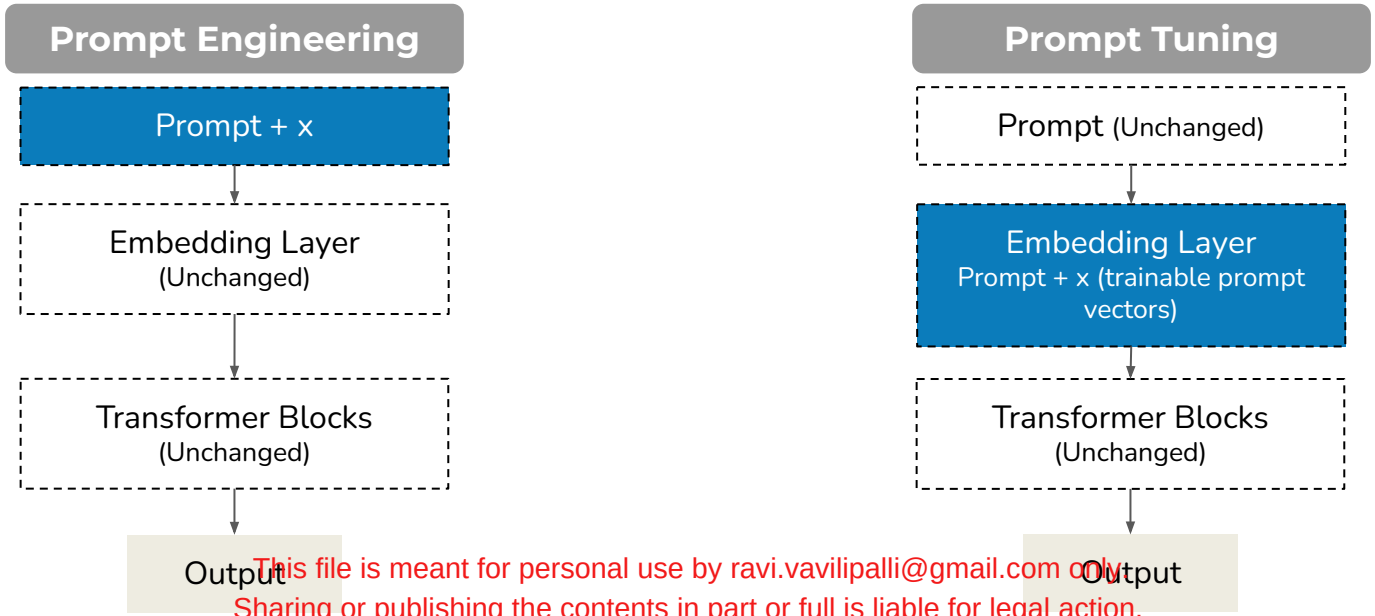
C. Compressing the model to reduce memory usage

D. Manually crafting input instructions and examples

Prompt Tuning

A method where instead of updating all model weights, we attach a few trainable prompt vectors to the input, while keeping the core model frozen

These vectors “nudge” the model towards better performance on a specific task



Fine Tuning Quiz

What is the main idea behind Parameter Efficient Fine Tuning methods?

A. Fine-tune all the billions of parameters in the LLM

B. Replace pre-training with fine-tuning

C. Only update a small number of additional parameters while keeping most weights frozen

D. Train models faster by skipping optimization

Fine Tuning Quiz

What is the main idea behind Parameter Efficient Fine Tuning methods?

A. Fine-tune all the billions of parameters in the LLM

B. Replace pre-training with fine-tuning

C. Only update a small number of additional parameters while keeping most weights frozen

D. Train models faster by skipping optimization

Parameter Efficient Fine Tuning

A technique to adapt large language models (LLMs) to specific tasks by training only a small fraction of the model's parameters, rather than retraining the entire model.

Efficiency Gains

Reduces training cost and memory requirements by avoiding updates to all model parameters.

Minimal Updates

Only about 0.01–1% of parameters are tuned, though the percentage varies by the approach taken.

Common Methods

Prompt Tuning, Prefix Tuning, Adapter Layers, LoRA, QLoRA

Fine Tuning Quiz

How do adapter layers enable efficient fine-tuning?

A. By adding small trainable layers between existing model layers

B. By removing unused hidden layers from the model

C. By replacing embeddings with domain-specific tokens

D. By compressing the model into fewer parameters

Fine Tuning Quiz

How do adapter layers enable efficient fine-tuning?

A. By adding small trainable layers between existing model layers

B. By removing unused hidden layers from the model

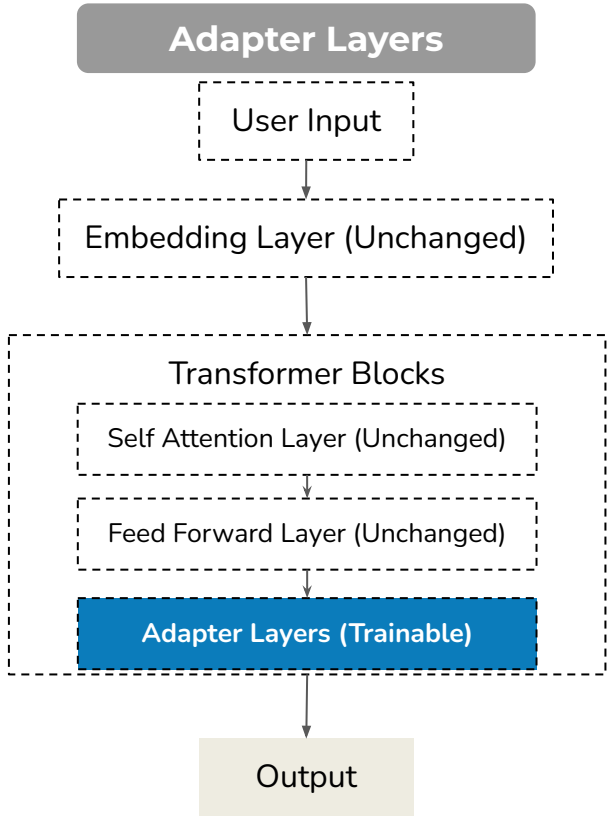
C. By replacing embeddings with domain-specific tokens

D. By compressing the model into fewer parameters

Adapter Layers

The Embedding layer, Self Attention Layer and Feed Forward Layer remain unchanged.

Adapter layers, which are trainable, are added within the transformer blocks.



Fine Tuning Quiz

In the context of business applications, what role does Low Rank Adaptation (LoRA) play in fine-tuning large language models (LLMs)?

A. LoRA facilitates rapid deployment of LLMs in cloud environments.

B. LoRA enables LLMs to achieve state-of-the-art performance without extensive computational resources

C. LoRA ensures seamless integration of LLMs with existing enterprise software systems

D. LoRA enhances the interpretability of LLMs for regulatory compliance purposes

Fine Tuning Quiz

In the context of business applications, what role does Low Rank Adaptation (LoRA) play in fine-tuning large language models (LLMs)?

A. LoRA facilitates rapid deployment of LLMs in cloud environments.

B. LoRA enables LLMs to achieve state-of-the-art performance without extensive computational resources

C. LoRA ensures seamless integration of LLMs with existing enterprise software systems

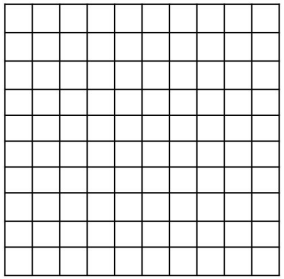
D. LoRA enhances the interpretability of LLMs for regulatory compliance purposes

Low Rank Adaptation (LoRA)

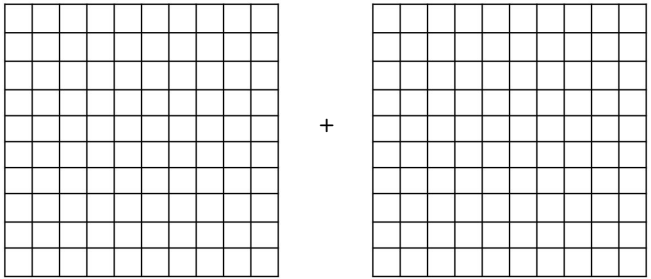
Consider the query matrix (Q) with 1 million parameters in an attention block within a transformer layer of an LLM

Full fine-tuning would require us to train all 1 million parameters - requires extensive compute resources!

LoRA works by adding another similar larger matrix delta Q (1 M parameters) to the original query matrix (Q)



Q: 1000 * 1000



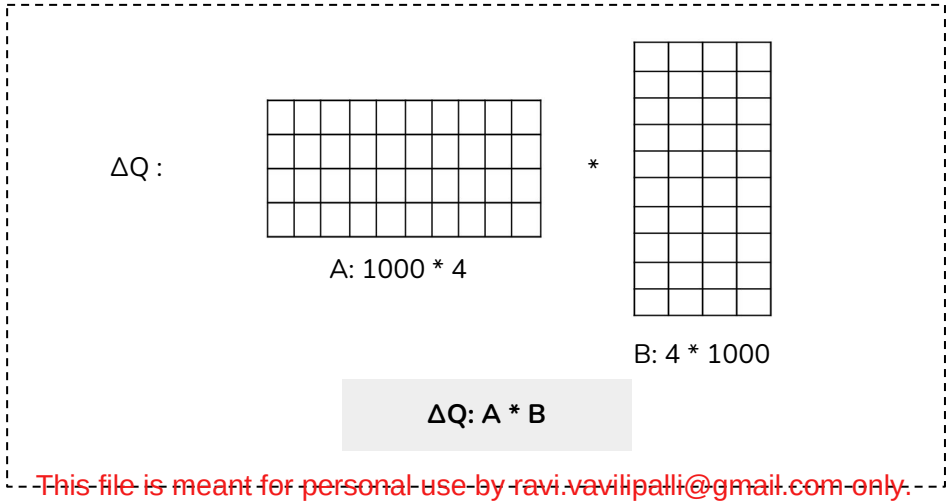
Q: 1000 * 1000 ΔQ: 1000 * 1000

$Q^* : Q + \Delta Q$

Low Rank Adaptation (LoRA)

The reduction in the number of trainable parameters is obtained by decomposing ΔQ into two smaller matrices (A and B) of size 1000 x r and r x 1000

For the above example, the number of trainable parameters reduces from 1 M to 8000 for r=4



Great Learning

Power Ahead!