Analytics Vidhya

Difference between RAG & Finetuning

Aspect	RAG	FineTuning
[illegible]	✓	✗
[illegible]	✓	✗
[illegible]	✗	✓
[illegible]	✓	✗
[illegible]	✓	✗
[illegible]	✗	✓
[illegible]	✗	✓
[illegible]	✓	✗
[illegible]	✗	✓
[illegible]	✓	✗

Analytics Vidhya

Purpose

RAG

Combines retrieval and generation for enhanced responses.

FineTuning

Specializes a pre-trained model for a specific task.

Analytics Vidhya

Functionality

RAG

Retrieves relevant documents and generates responses based on them.

FineTuning

Adjusts model parameters using task-specific data.

Analytics Vidhya

Use of External Knowledge

RAG

Uses external knowledge sources at inference time.

FineTuning

Embeds knowledge during training, no external data used at inference.

Analytics Vidhya

Training Process

RAG

Retriever and generator can be trained separately or together.

FineTuning

Involves updating the entire model with new data.

Analytics Vidhya

Adaptability to New Information

RAG

Adaptable; external knowledge can be updated without retraining.

FineTuning

Requires retraining to incorporate new information.

Analytics Vidhya

Application Examples

RAG

Open-domain question answering, customer support bots.

FineTuning

Sentiment analysis, text classification, machine translation.

Analytics Vidhya

Complexity

RAG

More complex due to the need to integrate retrieval and generation components.

FineTuning

Simpler; just requires fine-tuning the model with new data.

Analytics Vidhya

Real-time Information Access

RAG

Yes, real-time access to updated knowledge sources.

FineTuning

No, relies only on pre-trained data.

Analytics Vidhya

Model Update Frequency

RAG

External knowledge updates are independent of the model.

FineTuning

Must retrain to update model knowledge.

Implementation

RAG

Involves both retrieval and generation architectures.

FineTuning

Involves updating a single architecture.

Analytics Vidhya

A Summary

Aspect	RAG	FineTuning
Retrieval + Generation	✓	✗
External Knowledge	✓	✗
Full Model Retraining	✗	✓
Adaptable Without Retraining	✓	✗
Real-time Info Access	✓	✗
Task Specialization	✗	✓
Simpler Implementation	✗	✓
Open-domain Tasks	✓	✗
Task-specific Tasks	✗	✓
Integrated Components	✓	✗