



IBM

Generative AI on Amazon SageMaker - Deep Dive

Partha Dey

Enterprise Solutions Architect
AWS

Agenda

Evolution of LLMs

I am lost

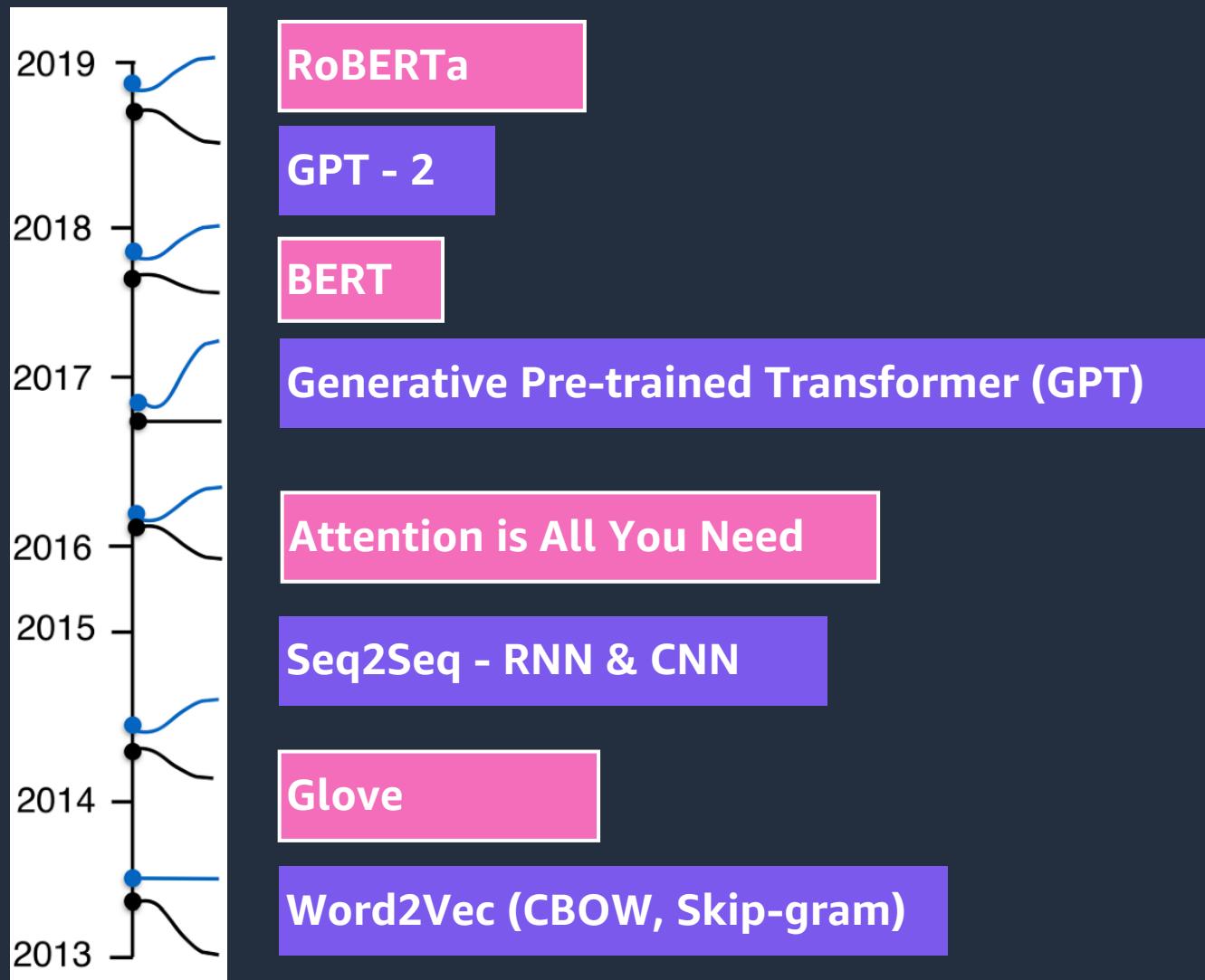
Best practices for Model Tuning

Amazon Distributed Training – for
Model Tuners and Providers



Evolution of LLMs

The timeline



Dive Deep



It started not long ago

The diagram illustrates word embeddings for four cities: Rome, Paris, Italy, and France. Each city is represented by a vector of numbers. Arrows point from the city names to their respective vectors.

Rome = [1, 0, 0, 0, 0, 0, ..., 0]

Paris = [0, 1, 0, 0, 0, 0, ..., 0]

Italy = [0, 0, 1, 0, 0, 0, ..., 0]

France = [0, 0, 0, 1, 0, 0, ..., 0]



Pros: Its numbers, computer can process
Cons: It doesn't capture the meaning



Dive Deep

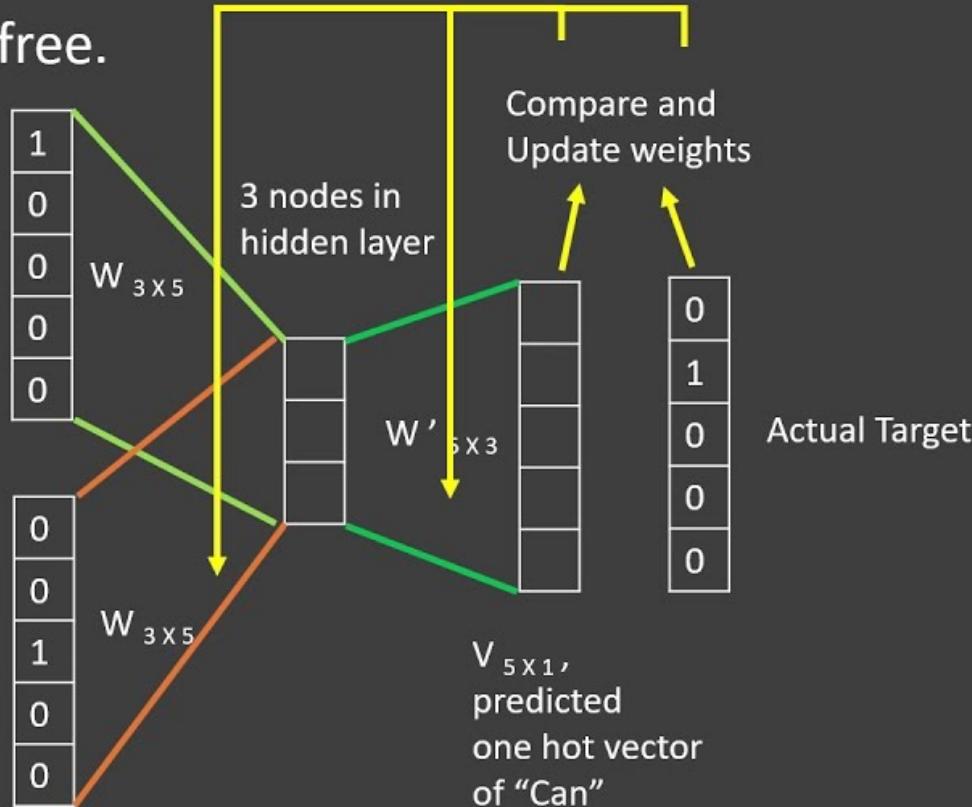


CBOW - Working

Hope can set you free.

$V_{5 \times 1}$, one hot vector of "Hope"

$V_{5 \times 1}$, one hot vector of "Set"

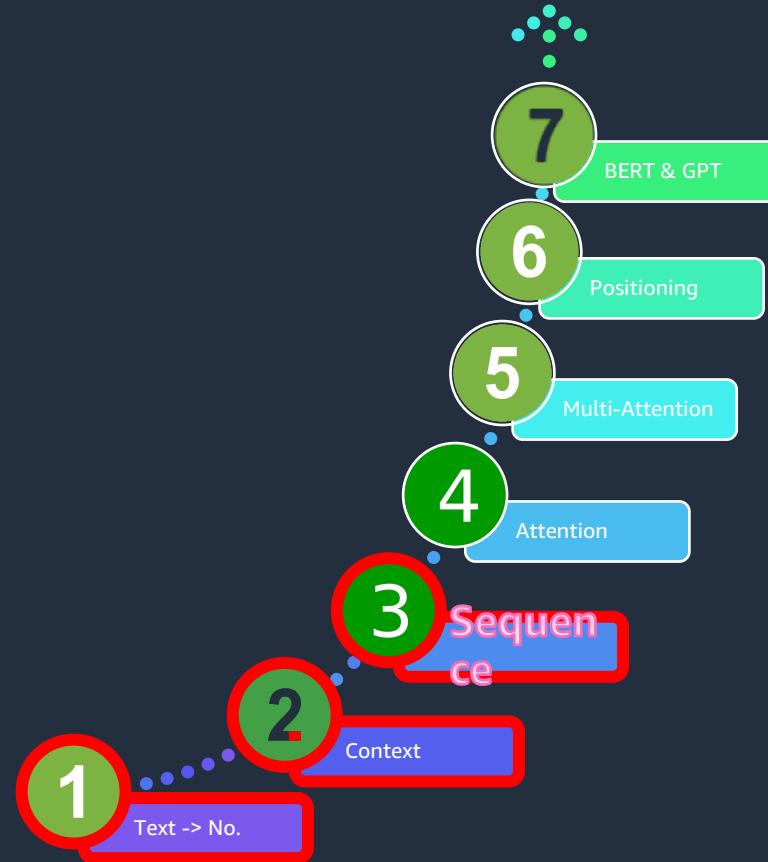


<https://www.youtube.com/watch?v=UqRCEmr1gQ>

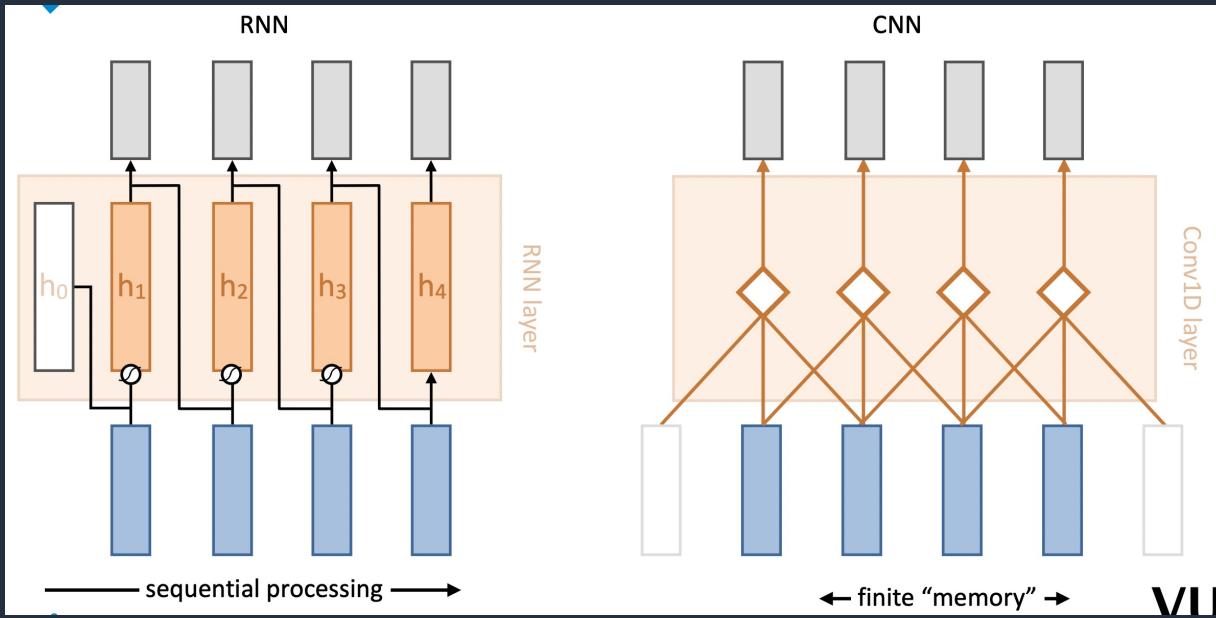
Pros: It captures meaning and context
Cons: It doesn't capture the sequence



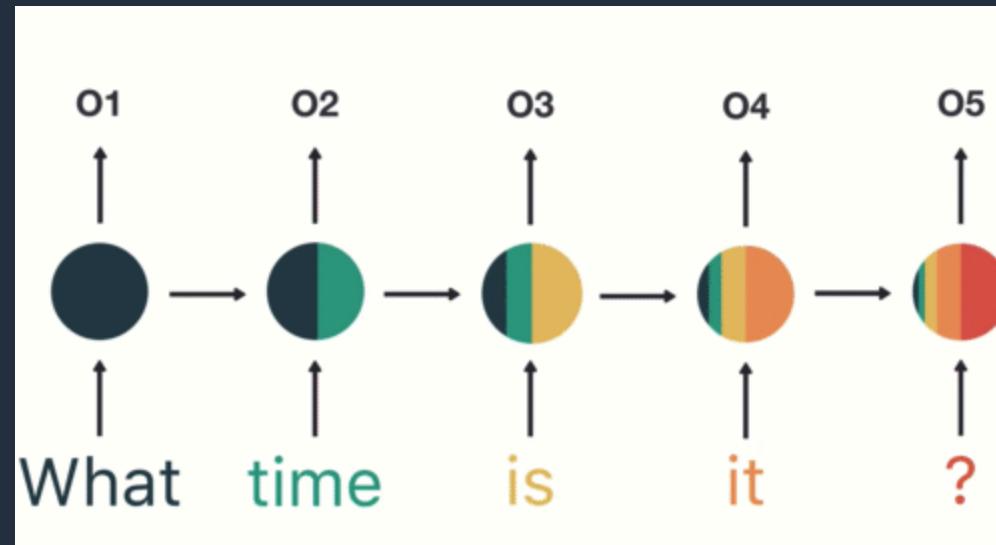
Dive Deep



RNNs & CNNs



RNNs & CNNs



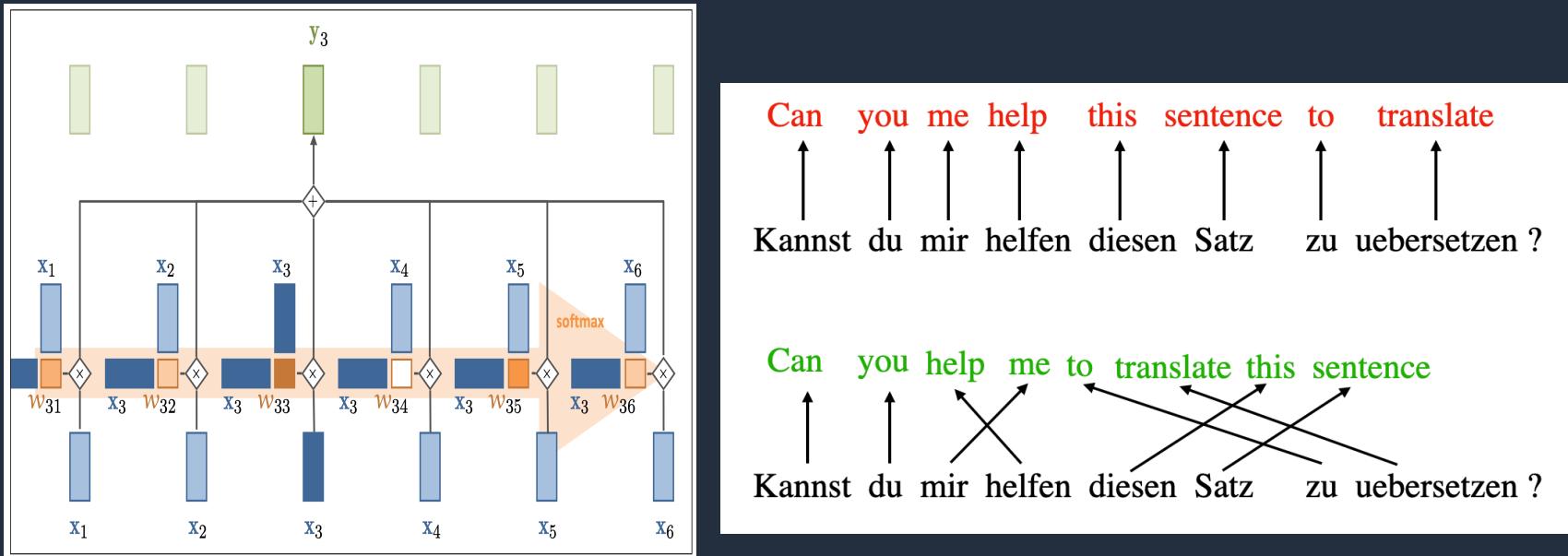
Pros: It captures the sequence
Cons: It is either slow or limited by memory



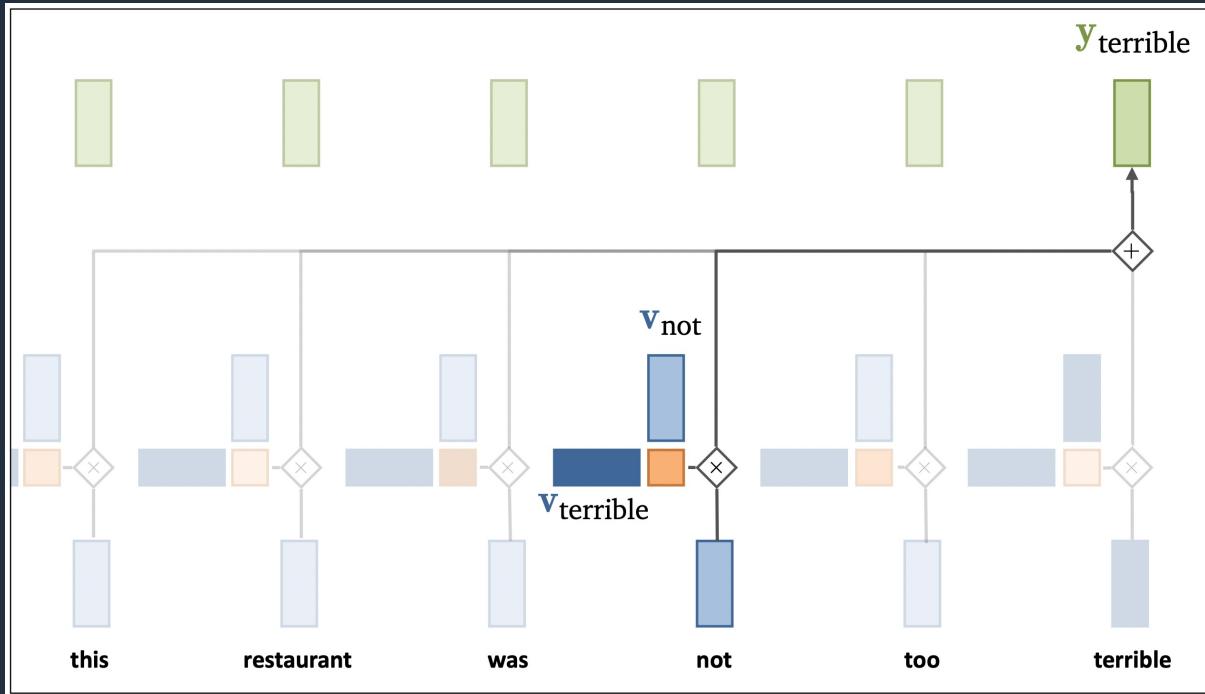
Dive Deep



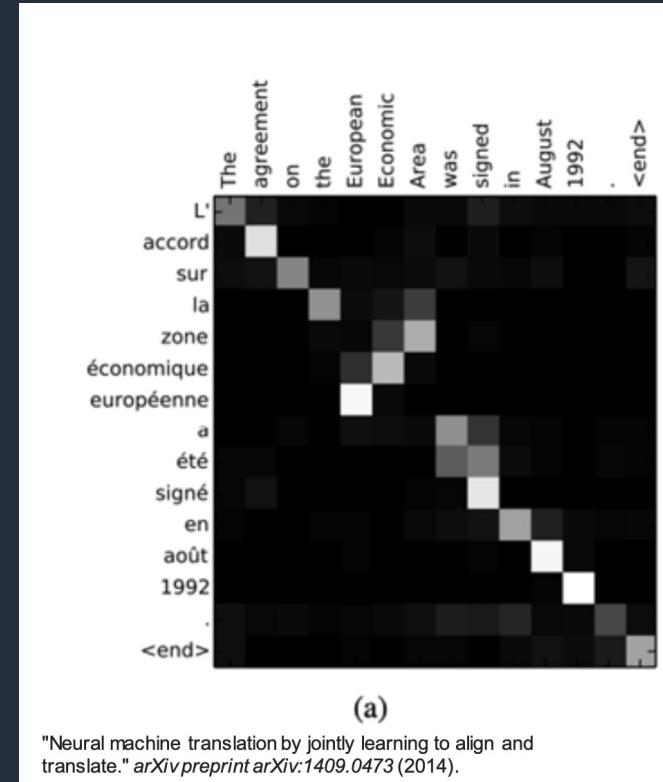
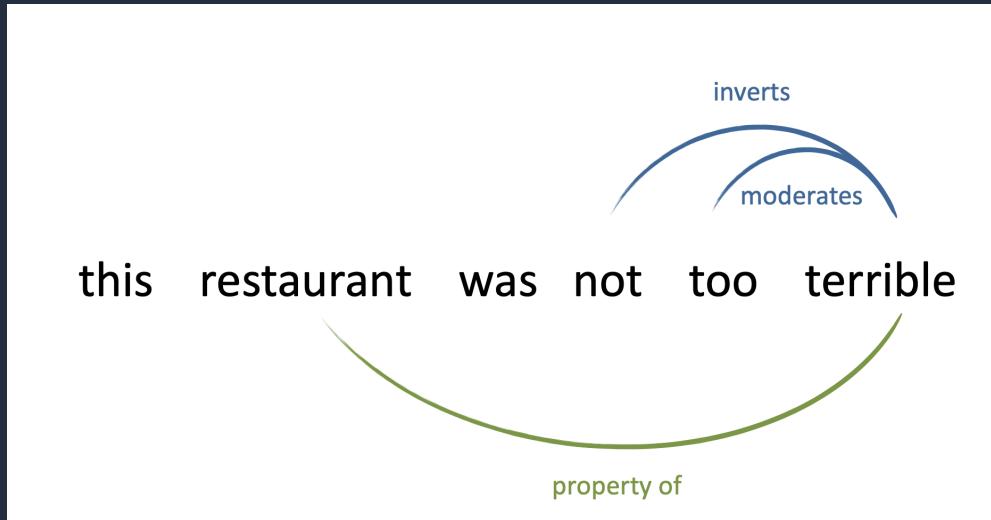
Self-Attention



Why attention is important



Why is attention important?



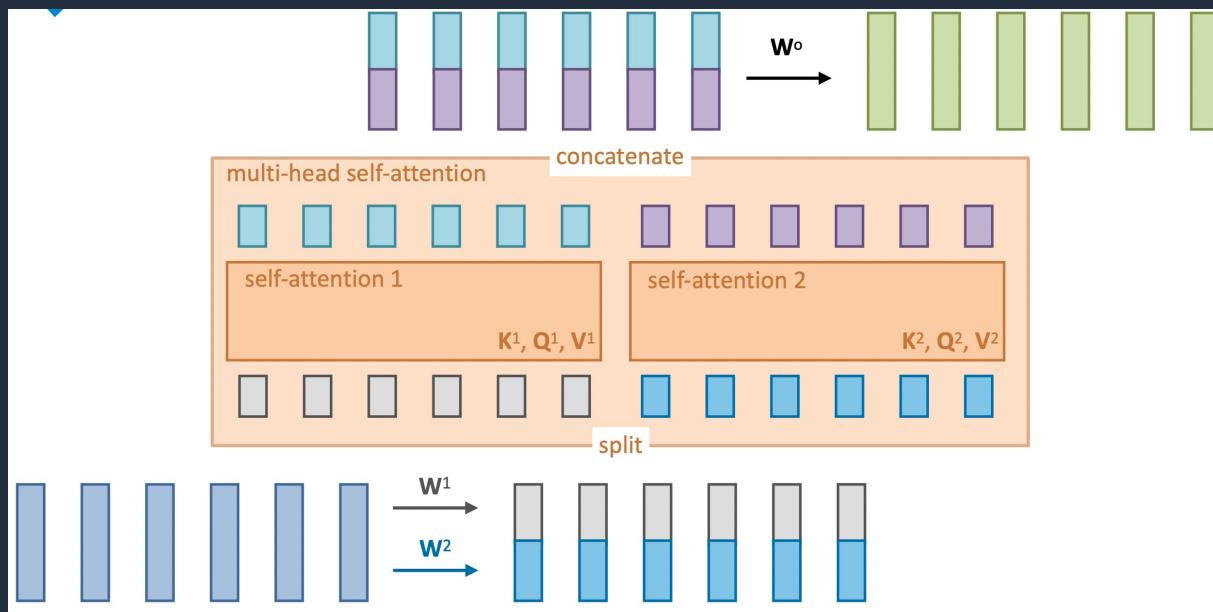
Pros: The model captures all words
Cons: One attention mechanism is not enough



Dive Deep



Multi-Head attention & Parallel Computations



this restaurant was not too terrible

inverts

moderates

property of



Pros: The model pays attention to multi-logic
Cons: It doesn't capture the word position



Dive Deep



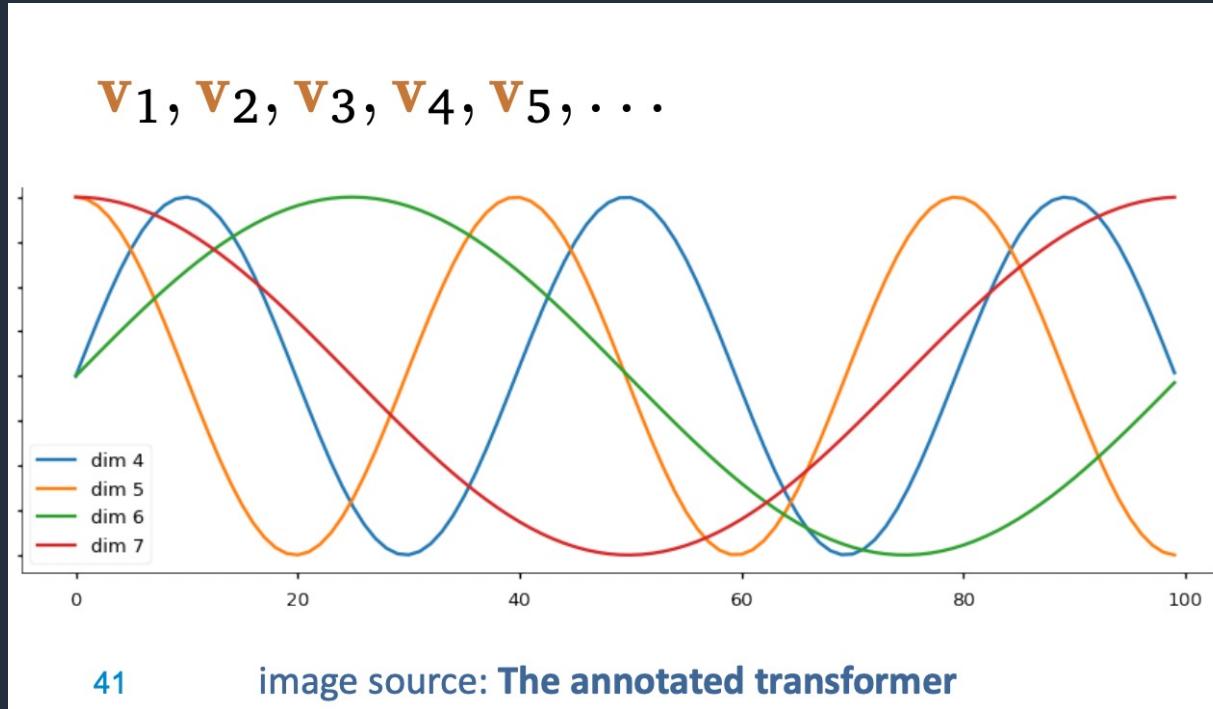
What about Position

This is not a real restaurant, it's a filthy burger joint.

This is not a filthy burger joint, it's a real restaurant.



Position Embeddings



Dive Deep



Transformer Architecture

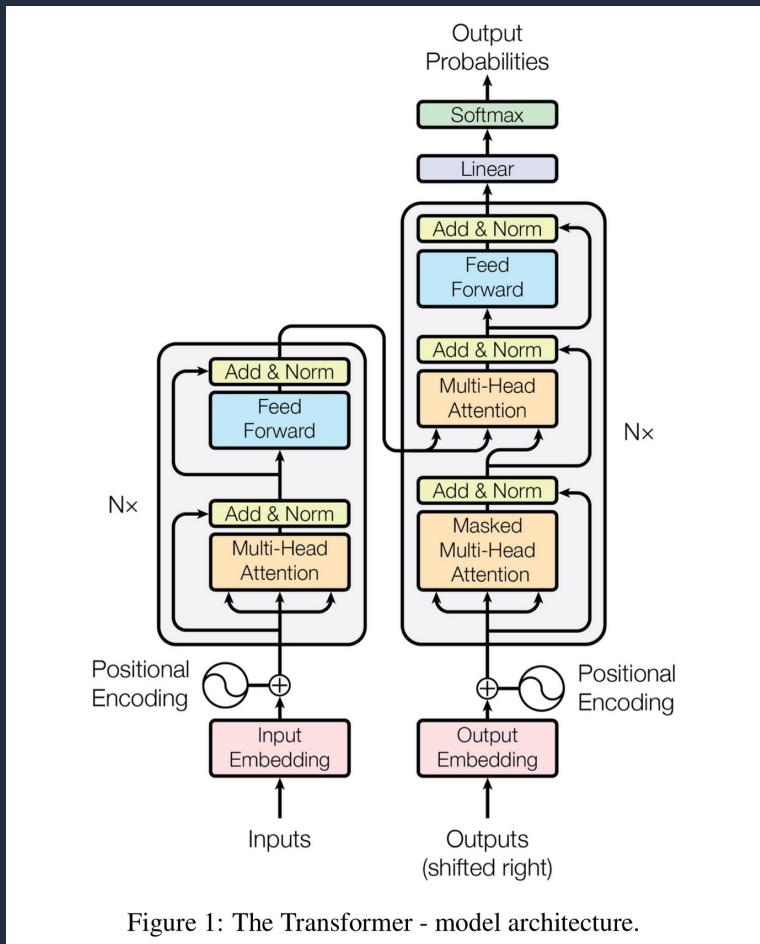


Figure 1: The Transformer - model architecture.



I am lost. It's OK.



Methods to use generative AI



Model provider

Customers who build their own foundation models from scratch



Model consumer

Start with publicly or proprietary available foundation models and fine-tune to customize for their domain



Application developer

Make API calls to third party foundation model providers

Low code option for model tuners and model consumers is to build on top of existing foundation models



Model provider

Build your own
foundation model



Amazon
SageMaker



Model tuners

Amazon SageMaker

Build on top of existing
foundation models



GPT-J



Bloom from HF



Model consumers

Generative AI



Amazon
CodeWhisperer

Light^{bulb} AI21labs



alexa stability.ai

Amazon SageMaker JumpStart



Customer challenges

Getting started with ML takes

too long...

- Importing publicly available algorithms and models into SageMaker
- Maintaining and updating SageMaker-compatible scripts
- Setting up infrastructure
- Building NLP and vision models from scratch
- Model sharing and collaboration is done manually



Amazon SageMaker JumpStart

ML hub with foundation models, built-in algorithms, and prebuilt ML solutions that you can deploy with just a few clicks



Machine learning hub

Browse through 400+ built-in algorithms with pretrained models, pretrained foundation models, solutions, and example notebooks



Pre-built training and inference scripts

Compatible with SageMaker and configurable with custom dataset



UI as well as API-based

Use the user interface for single click model deployment or API for the Python SDK-based workflow



Notebooks with examples

Jump into notebooks to use selected model with examples to guide you through the entire ML workflow



Share and collaborate within your organization

Share models and notebooks with others within your organization, and allow them to train with their own data or deploy as-is for inferencing

Why use foundation models on SageMaker JumpStart

1

Choose foundation models offered by model providers

AI21labs

Lighton
We bring Light to AI

stability.ai

cohere



alexa

2

Try out model and/or deploy



Try out models via AWS Console



Deploy the model for inference using SageMaker hosting options includes single node

3

Fine tune model and automate ML workflow



Only selected models can be fine-tuned



Automate ML workflow

Data stays in your account including model, instances, logs, model inputs, model outputs

Fully integrated with Amazon SageMaker features



SageMaker JumpStart models and features

Publicly available			Proprietary models		
stability.ai	alexa		co:here	Light	AI21labs
Models SD XL, Upscaling, Inpainting	Models AlexaTM 20B	Models Falcon, OpenLLaMA, Flan, GPT NeoXT, BloomZ 176B	Models Cohere Command	Models Lyra-Fr 10B, Mini	Models Jurassic-2 Ultra, Mid
Tasks Generate photo-realistic images from text input Improve quality of generated images	Tasks Machine translation Summarization	Tasks Machine translation Question answering Summarization	Tasks Text generation Information extraction Question answering Summarization	Tasks Text Generation Keyword extraction Information extraction Question answering Summarization	Tasks Text generation Long-form generation Summarization Paraphrasing Chat Information extraction Question answering Classification
Features Fine-tuning on SD 2.1 model		Features Instruction and domain adaptation, RAG			



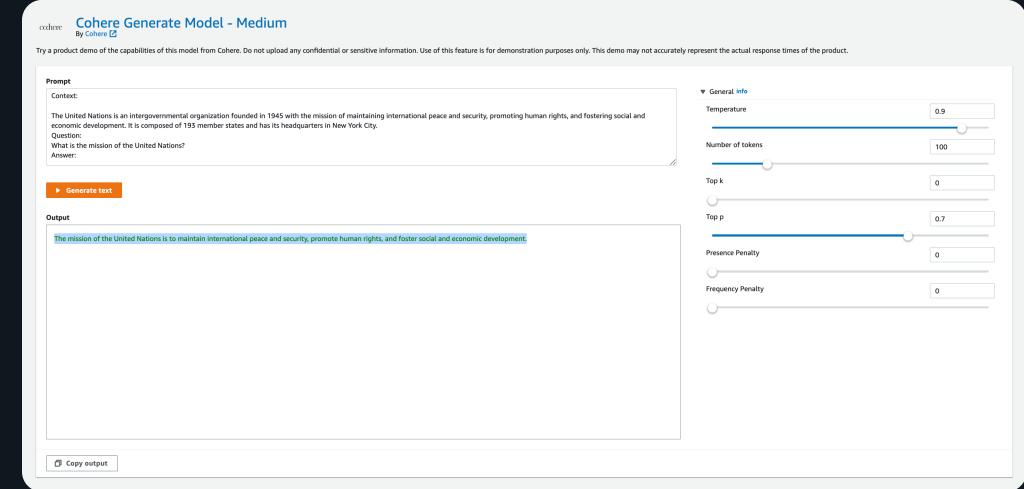
Build on top of existing foundation models using Amazon SageMaker JumpStart



The screenshot shows the Amazon SageMaker JumpStart landing page. At the top, there's a navigation bar with links like Overview, Features, Pricing, FAQs, By Role, By ML Lifecycle, Getting Started (which is underlined), Customers, and Partners. Below the navigation is a breadcrumb trail: Products / Machine Learning / Amazon SageMaker JumpStart. The main title is "Getting started with Amazon SageMaker JumpStart". A brief introduction states: "Amazon SageMaker JumpStart is a machine learning (ML) hub that can help you accelerate your ML journey. Explore how you can get started with built-in algorithms with pretrained models from model hubs, pretrained foundation models, and prebuilt solutions to solve common use cases. To get started, see documentation or example notebooks that you can quickly execute." On the left, there are filter options for "Reset Filters", "Product Type" (Foundation Model, Model, Solution), and "Text Tasks" (End-to-end Solution, Text Classification, Text Embedding, Text Generation, Text Summarization, Named Entity Recognition, Question Answering, Zero-Shot Classification). There's also a search bar labeled "Search for content" and a "Sort By" dropdown set to "Popularity". The main content area displays a grid of four foundation models:

- Falcon 40B Instruct BF16** (Huggingface): Model ID: huggingface-textgeneration-falcon-40b-instruct-bf16. Falcon-40B-Instruct is a 40B parameters causal decoder-only model built by TII based on Falcon-40B and finetuned on a mixture of Baize.
- Open LLaMa** (Huggingface): Model ID: huggingface-textgeneration-open-llama. This is a Text Generation model built upon a Transformer model from Hugging Face. It is a permissively licensed (Apache-2.0) open source reproduction of Meta AI's LLaMA 7B trained
- Stable Diffusion XL Beta V0.8** (StabilityAI): Model ID: stable-diffusion-xl-beta-v0.8. Extend beyond just text-to-image prompting. Stable Diffusion XL offers several ways to modify the images: Inpainting - edit inside the image, Outpainting - extend the image outside of the original
- Cohere Command** (Cohere): Generative model that responds well with instruction-like prompts. This model provides businesses and enterprises with best quality, performance and accuracy in all generative tasks. And with our intuitive SDK, unlocking the full potential of LLMs for your applications has

Try-out experience



The screenshot shows the Cohere Generate Model - Medium interface. On the left, there's a 'Prompt' section with a context about the United Nations and a question 'What is the mission of the United Nations?'. Below it is an 'Output' section displaying the generated response: 'The mission of the United Nations is to maintain international peace and security, promote human rights, and foster social and economic development.' To the right, there are several configuration sliders under a 'General info' heading: Temperature (0.9), Number of tokens (100), Top k (0), Top p (0.7), Presence Penalty (0), and Frequency Penalty (0).

- Try out the models and model prompts without running code or incurring costs
- Available for proprietary models in Top 10 in HELM benchmarks and public models for comparison purposes
- This is a shared environment in a SageMaker escrow account



Choosing the right instance for hosting

Size of model (# of parameters)	Large 3B–10B	Mega 11B–20B	Massive 100B+*
Task Type	Image generation Simple text classification (Short form)	Natural language understanding (NLU)	Natural language generation (NLG) (long form)
Minimum instance required	p3.2xlarge g5.2xlarge	p3.8xlarge g5.12xlarge	p4de.24xlarge p4d.24xlarge
Pricing	\$4/hr \$2/hr	\$15/hr \$9/hr	\$47/hr \$38/hr

Scale vertically (larger instances) to improve latency

Scale horizontally (more instances) to support higher traffic

*P4d instances will have limited availability, escalate to S-Team for support



Instance type/size recommendation for models

Model	Instance
Text Generation	
J2 Ultra	g5.48xlarge
J2 Mid	g5.12xlarge
Cohere Command Medium	g5.xlarge
Cohere Command XL	p4d.24xlarge
FLAN T5 XL	g5.2xlarge
FLAN T5 XXL	g5.12xlarge
FLAN UL2	g5.12xlarge
GPT-J 6B	g5.12xlarge
GPT NeoX	g5.24xlarge
Image Generation	
Stable Diffusion 2.1 base	g5.2xlarge
SD Upscaling	g5.2xlarge



Selecting a foundation model



- Modality (text, image, multi-modal...)
- Training datasets (multi-lingual, Java...)
- Size
- Accuracy
- Throughput / latency
- Context length
- Inference cost

Center for Research on Foundation Models HELM Models Scenarios Results Raw runs

Core scenarios

The scenarios where we evaluate all the models.

[Accuracy | Calibration | Robustness | Fairness | Efficiency | General information | Bias | Toxicity | Summarization metrics]

Accuracy

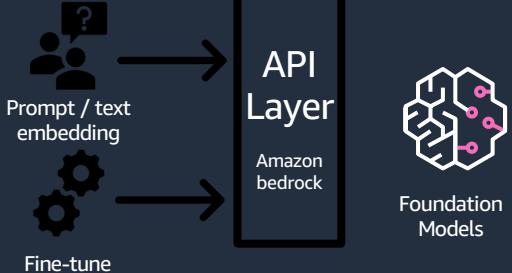
Model/adapter	Mean win rate ↑ [sort]	MMLU - EM ↑ [sort]	BoolQ - EM ↑ [sort]	NarrativeQA - F1 ↑ [sort]	NaturalQuestions book) - F1 ↑ [sort]
Cohere Command beta (52.4B)	0.93	0.452	0.856	0.752	0.372
text-davinci-002	0.93	0.568	0.877	0.727	0.383
text-davinci-003	0.898	0.569	0.881	0.727	0.406
TNLG v2 (530B)	0.855	0.469	0.809	0.722	0.384
Anthropic-LM v4-s3 (52B)	0.842	0.481	0.815	0.728	0.288
J1-Grande v2 beta (17B)	0.806	0.445	0.812	0.725	0.337
Luminous Supreme (70B)	0.783	0.38	0.775	0.711	0.293
Cohere Command beta (6.1B)	0.762	0.406	0.798	0.709	0.229
Cohere xlarge v20221108 (52.4B)	0.74	0.382	0.762	0.672	0.361
OPT (175B)	0.687	0.318	0.793	0.671	0.297
Cohere xlarge v20220609 (52.4B)	0.649	0.353	0.718	0.65	0.312
davinci (175B)	0.628	0.422	0.722	0.687	0.329
GLM (130B)	0.6	0.344	0.784	0.706	0.148
J1-Jumbo v1 (178B)	0.592	0.259	0.776	0.695	0.293
Luminous Extended (30B)	0.582	0.321	0.767	0.665	0.254
BLOOM (176B)	0.528	0.299	0.704	0.662	0.216
OPT (66B)	0.522	0.276	0.76	0.638	0.258

Instance types for fine-tuning foundation models on SageMaker JumpStart

	G4dn	P3	P3dn	P4d
Ideal for	Accelerated training of small to medium sized models with less than 100M parameters	Training medium to large models with 100M to 300M parameters Good for single node distributed training	Training large models with more than 300M parameters Spot Training may provide better price-performance than P4d Good for multi-node distributed training	Customer looking for best training performance on the cloud Training large models with more than 300M parameters Good for multi-node distributed training
Key features	16 GB/GPU PCIe only 25–50 Gbps networking 100 Gbps on bare-metal	16 GB/GPU 200–300 GB/s NVLink (4, 8 GPUs) 10–25 Gbps networking	32 GB/GPU 300 GB/s NVLink (8 GPUs) 100 Gbps networking	40 GB/GPU 600 GB/s NVLink (8 GPUs) 400 Gbps networking
GPU config	1, 4, or 8 NVIDIA Tesla T4s	1, 4, or 8 NVIDIA Tesla V100s	8 NVIDIA Tesla V100s	8 NVIDIA Tesla A100s (latest)
Recent launches	Habana Gaudi	Trainium	g5	p4de*
	Broadest and most complete set of Distributed Training Infrastructure choices			

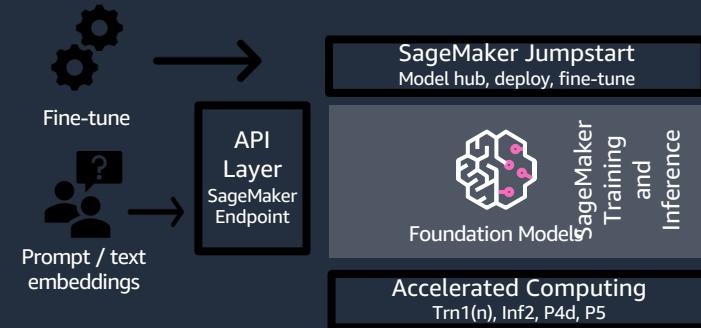


How do I access foundation models?



Amazon Bedrock

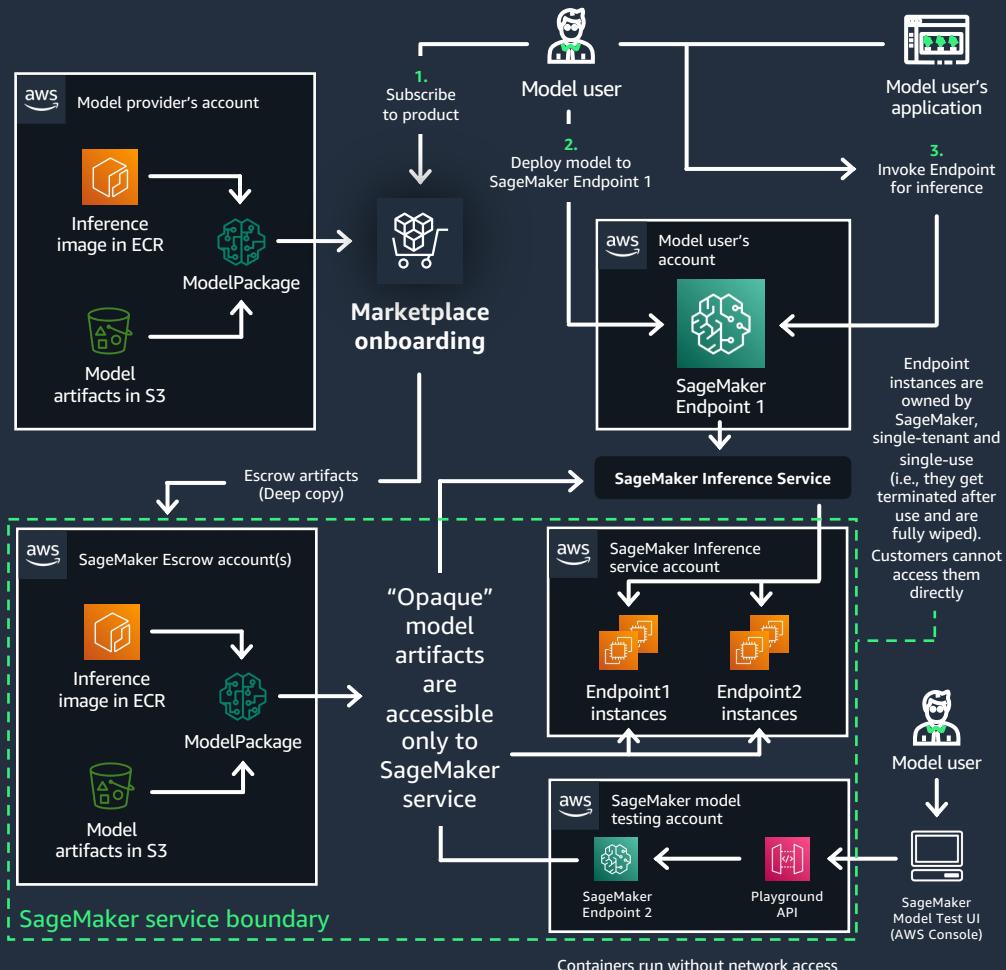
- The easiest way to build and scale generative AI applications with foundation models (FMs)
- Access directly or fine-tune foundation model using API
- Serverless



Amazon SageMaker JumpStart

- Machine learning (ML) hub with foundation models, built-in algorithms, and prebuilt ML solutions that you can deploy with just a few clicks
- Deploy FM as SageMaker Endpoint (hosting)
- Fine-tuning leverages SageMaker Training jobs
- Choose SageMaker managed accelerated computing instance





SageMaker JumpStart protects your data and model provider IP

- Proprietary model package and endpoint is hosted in SageMaker owned escrow account
- Containers have no outbound network access; user data and model provider IP is protected the same time
- No data is used to update/train the base model that JumpStart provides to customers



HuggingFace



Hugging Face: the largest collection of open source models and datasets

Hugging Face Search models, datasets, users...

Click here to log in through Single Sign-On to view activity within the huggingface.org.

Models 193,365 Filter by name Full-text search Sort: Most Downloads

Tasks Libraries Datasets Languages Licenses Other

Filter Tasks by name

Multimodal

- Feature Extraction Text-to-Image
- Image-to-Text Text-to-Video
- Visual Question Answering
- Document Question Answering
- Graph Machine Learning

Computer Vision

- Depth Estimation Image Classification
- Object Detection Image Segmentation
- Image-to-Image Unconditional Image Generation
- Video Classification Zero-Shot Image Classification

Natural Language Processing

- Text Classification Token Classification
- Table Question Answering Question Answering
- Zero-Shot Classification Translation
- Summarization Conversational
- Text Generation Text2Text Generation
- Fill-Mask Sentence Similarity

Audio

- Text-to-Speech Automatic Speech Recognition
- Audio-to-Audio Audio Classification
- Voice Activity Detection

bert-base-uncased (jonatasgrosman/wav2vec2-large-xlsr-53-english)

gpt2 (xlm-roberta-base)

microsoft/resnet-50 (facebook/dino-vitb16)

openai/clip-vit-large-patch14 (roberta-base)

facebook/convnext-large-224 (microsoft/resnet-18)

facebook/convnext-base-224 (facebook/dino-vits8)

distilbert-base-uncased (microsoft/layoutlmv3-base)

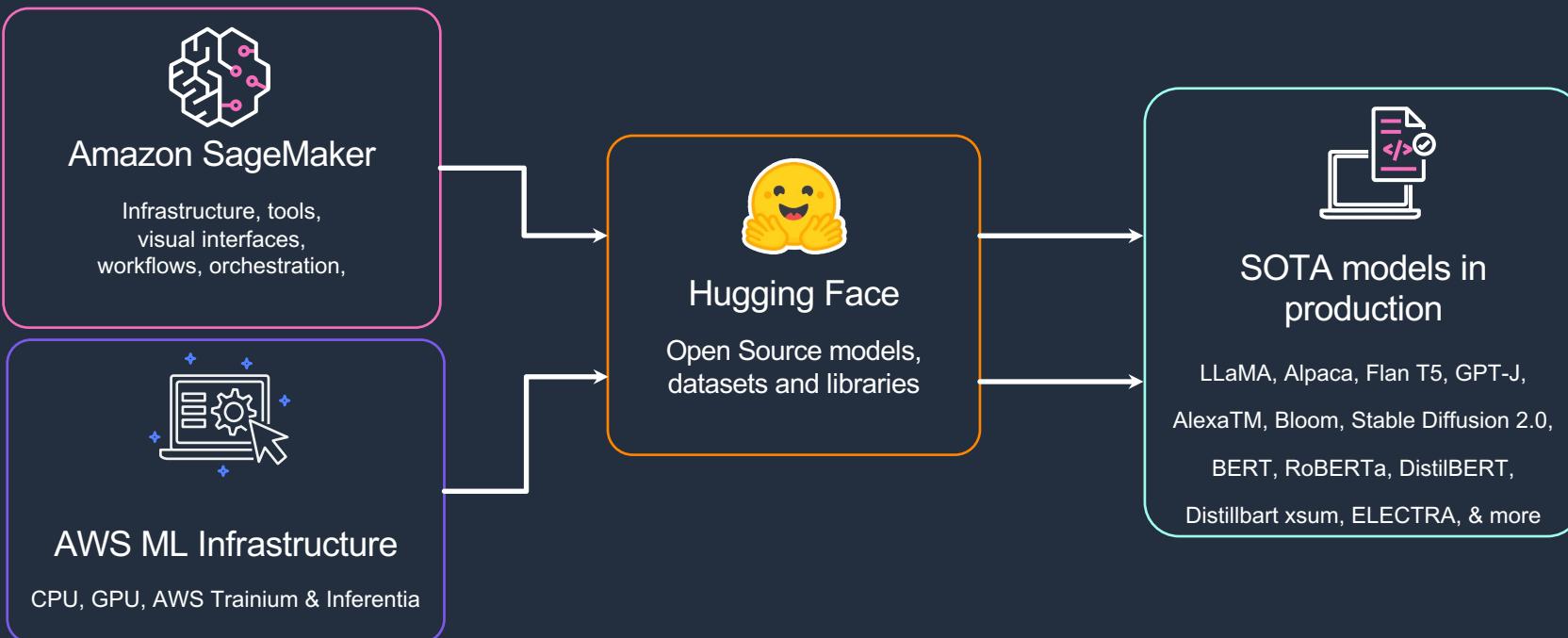
t5-base (prajjwali/bert-tiny)

bert-base-multilingual-cased (bert-base-cased)

xlm-roberta-large (distilroberta-base)



Hugging Face and AWS collaborate to simplify state-of-the-art AI



Hugging Face open source libraries

- Transformers: Transformer models for Pytorch, TensorFlow, and JAX
- Diffusers: image and audio generation models for PyTorch
- Accelerate: simple distributed training (CPU, GPU, TPU) for PyTorch
- Peft: Parameter Efficient Fine-Tuning lets you train larger models on the same GPU
- SetFit: few-shot learning for Sentence Transformers
- Optimum: hardware acceleration for Transformers and Diffusers
 - Optimum Intel: Intel Neural Compressor, Intel OpenVINO
 - Optimum Habana: training and inference for Habana Gaudi/Gaudi2
 - Optimum Neuron: AWS Trainium and Inferentia2



End-to-end ML with Hugging Face on AWS

Hugging Face models,
datasets, and libraries

Hugging Face Expert Acceleration Program (EAP)



Experiment on Hugging Face

Hugging Face
Spaces



Q2'23

Train and deploy on Amazon EC2

Hugging Face
DLAMI



Train and deploy on Amazon SageMaker

Hugging Face
DLCs



Deploy on Hugging Face

Hugging Face
Inference Endpoints



Q2'23

AWS Infrastructure (CPU, GPU, Trainium, Inferentia)



Technical resources

Hugging Face documentation



[Documentation](#)

SageMaker documentation

A screenshot of the Amazon SageMaker documentation page. The page title is "What is Amazon SageMaker?". It features a large heading "Amazon SageMaker" with a sub-subtitle "Machine learning made easy". Below the heading is a detailed description of what SageMaker is and how it works. The page includes several sections: "Amazon SageMaker Overview", "Amazon SageMaker Features", and "Amazon SageMaker Benefits". There are also links to "Amazon SageMaker Pricing" and "How to Use SageMaker". The page has a standard navigation bar at the top and a footer at the bottom.

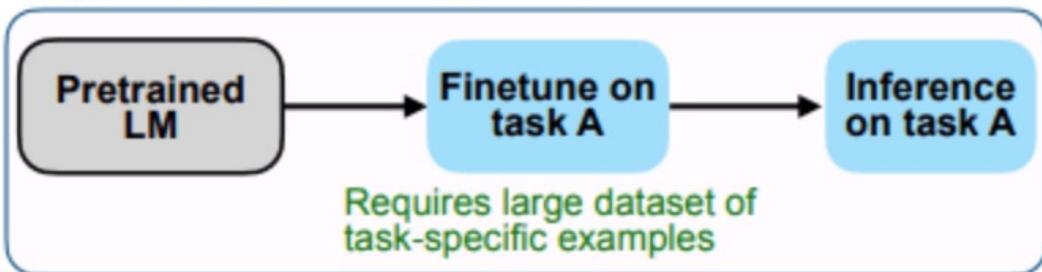
[Documentation](#)



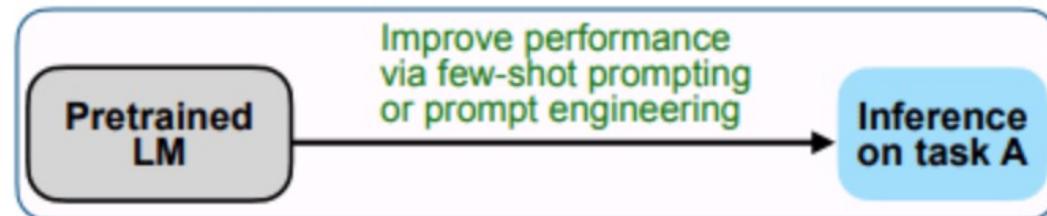
Best practices for Model Tuning

Learning Paradigms

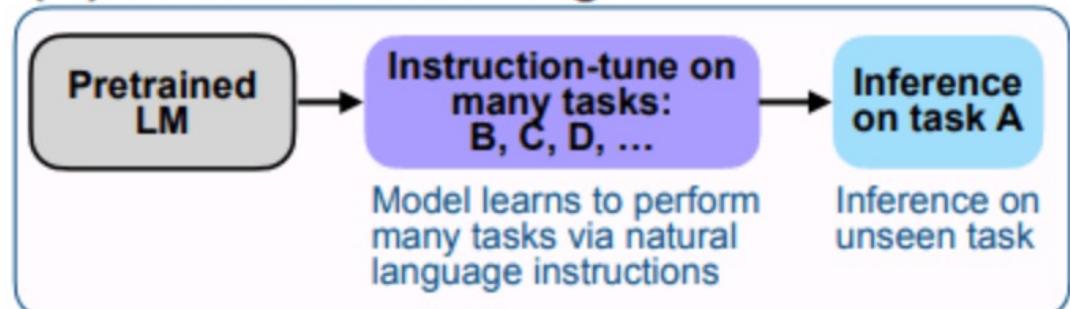
(A) Pretrain–finetune



(B) Prompting



(C) Instruction tuning



Using Foundation Models – User Journey Example



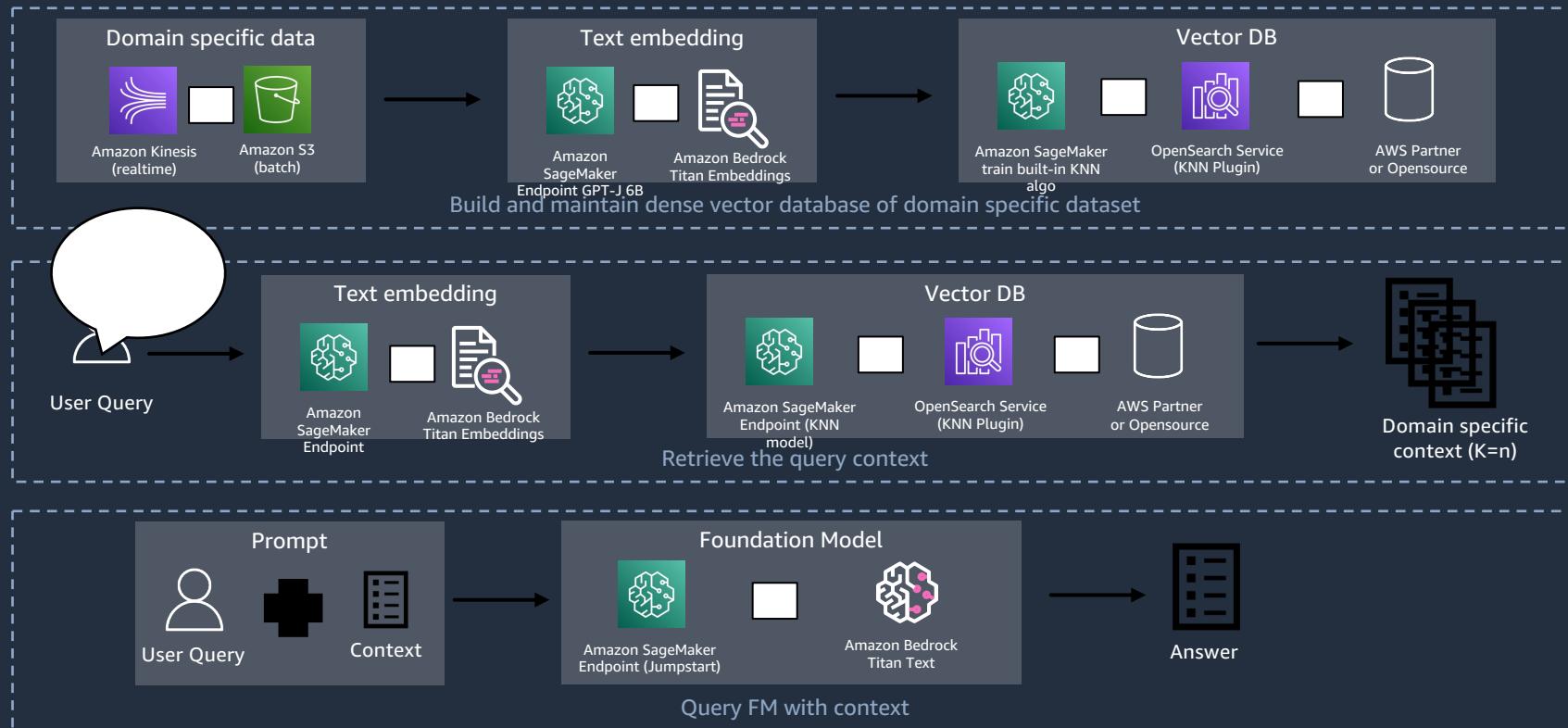
Q&A on proprietary or domain specific data

(e.g., internal Wiki, FAQs, product documentation)

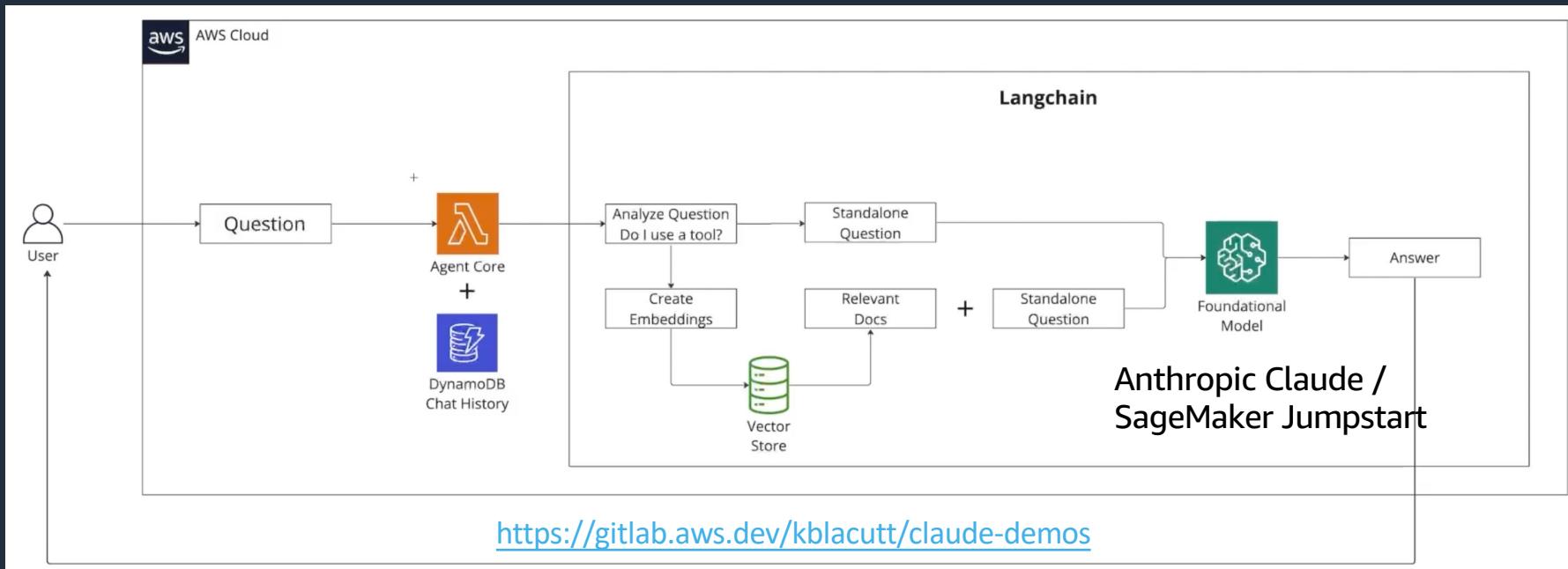
Knowledge Augmentation

Retrieval-Augmented Generation (RAG)

EXAMPLE ARCHITECTURE



Bored of going through transaction history?
Now you can talk literally talk to your bank
account!



Credit: Kenton Blacutt



Examples: RAG and domain adaptation/fine-tuning

QUESTION AND ANSWER USING DOMAIN SPECIFIC DATASET

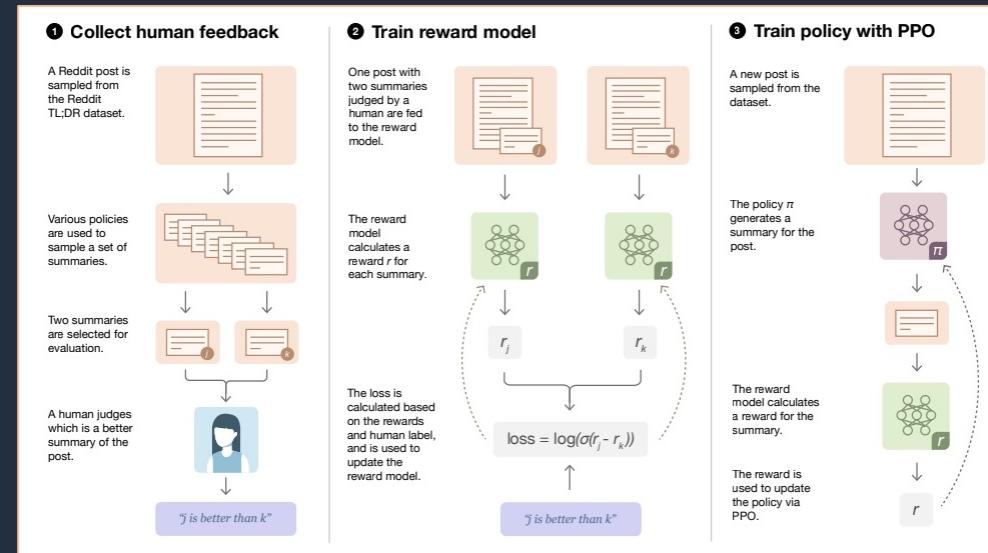
- [Amazon SageMaker Jumpstart + VectorDB as Amazon SageMaker KNN and Opensource \(langchain\)](#)
- [Amazon SageMaker Jumpstart + VectorDB as Amazon OpenSearch](#)
- [Domain Adaption Fine Tuning using Amazon SageMaker JumpStart on Financial Data](#)



Reinforcement Learning from Human Feedback (RLHF)

- **Why use RL with human feedback for LLMs?**
 - Reduces the need for *explicit supervision*
 - Improves the model's ability to generate more *helpful* and *context-aware* responses
 - Enables fine-tuning to *align* with human values and preferences
- **Process**
 1. Gather model-generated responses
 2. Obtain human rankings based on quality
 3. Train a reward model using ranked data
 4. Predict reward values for potential responses
 5. Fine-tune with *Proximal Policy Optimization* (PPO)

On Roadmap

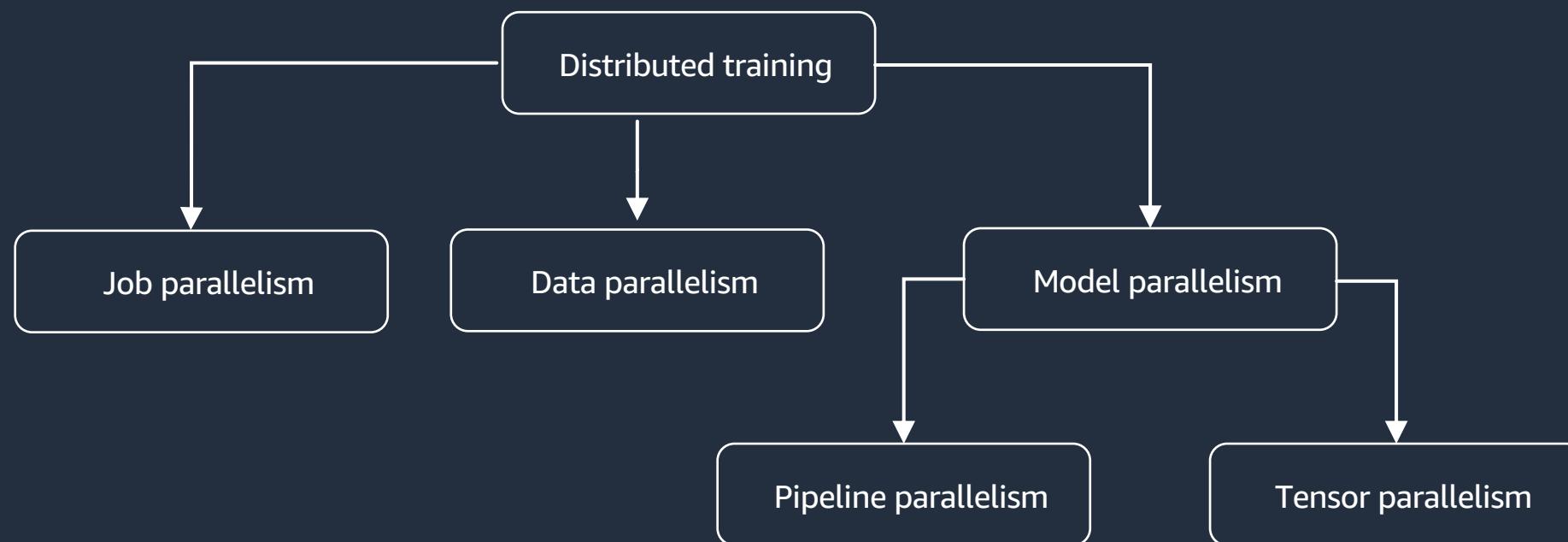


Amazon Distributed Training – for Model Tuners and Providers

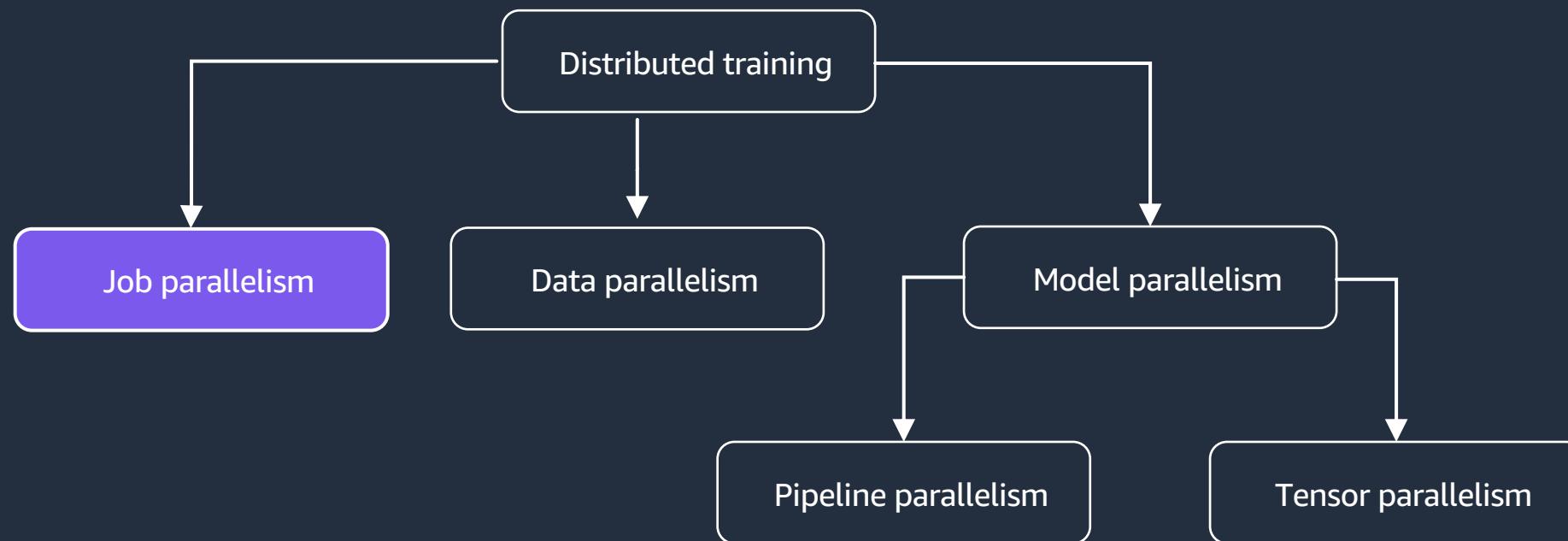
Distributed Training Types



There are many kinds of distributed training



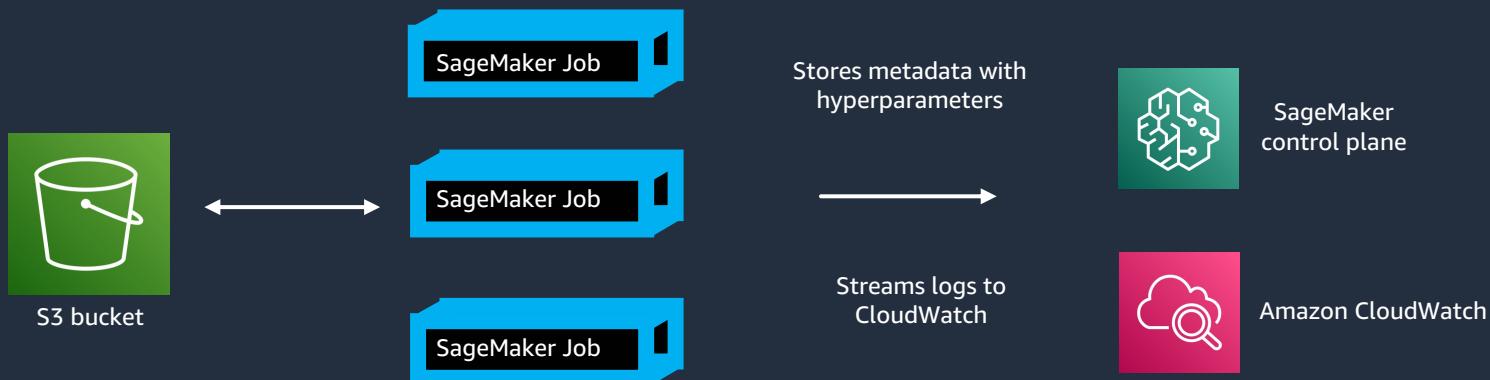
There are many kinds of distributed training



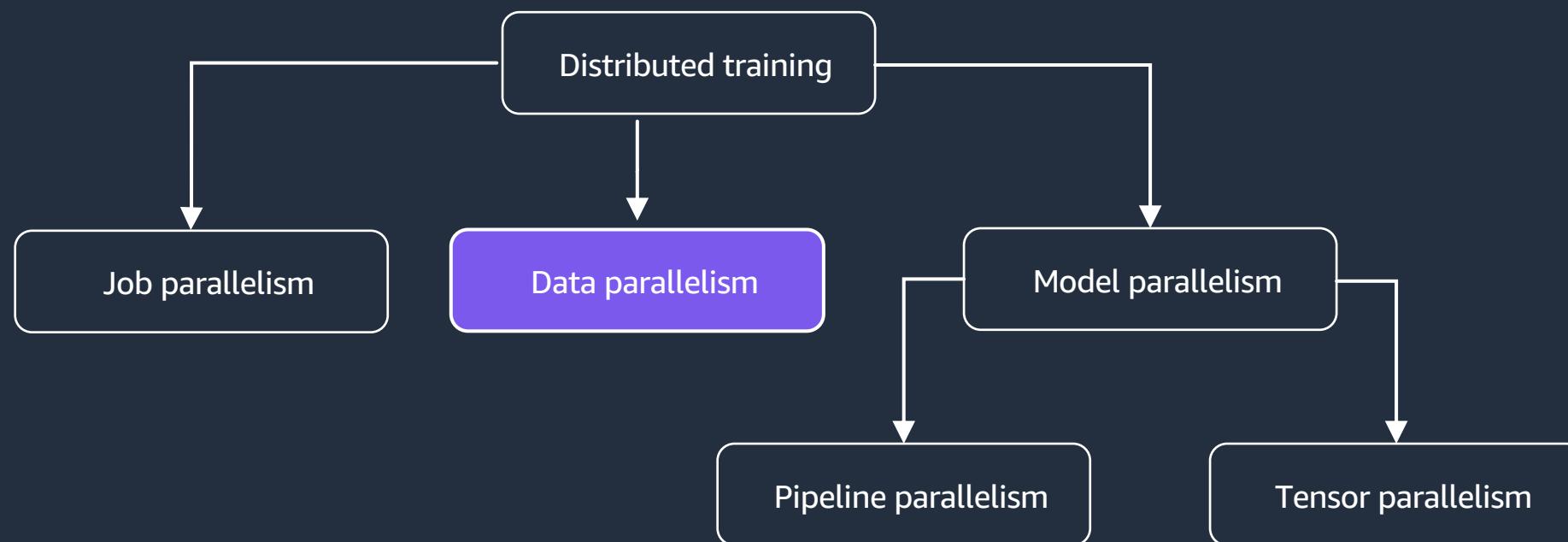
Train with parallel jobs at high frequency

1. Each job can train as many models as you need.
2. You can use *warm pools* to retrain as quickly as possible

```
• for model in list_of_models:  
•     s3_input = get_data(model)  
•     s3_output = get_location(model)  
•     estimator = get_estimator(model, s3_output)  
•     estimator.fit(s3_input, wait=False)
```



There are many kinds of distributed training



Data Parallelism



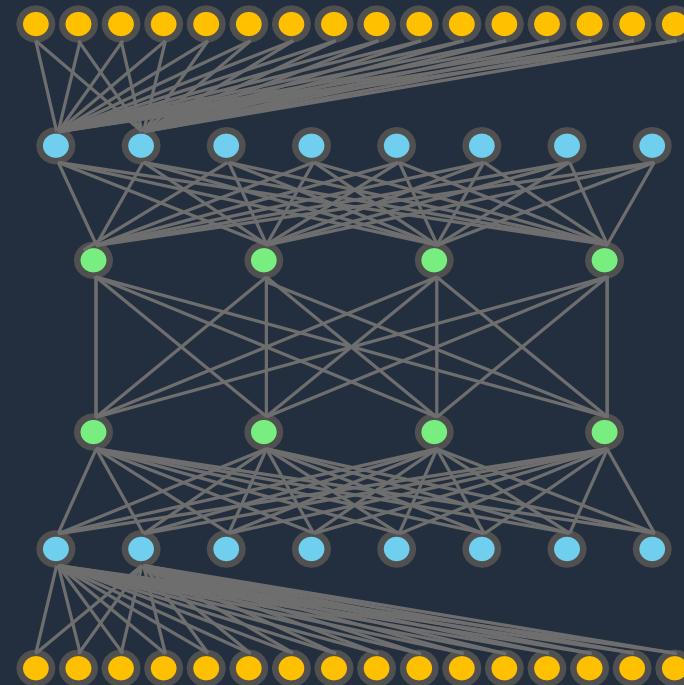
Amazon SageMaker Distributed Data Parallel

Optimized backend for distributed training of deep learning models in TensorFlow, PyTorch

Accelerates training for network-bound workloads

Built and optimized for AWS network topology and hardware

20%–40% faster and cheaper than NCCL and MPI-based solutions. **Best performance on AWS for large clusters.**

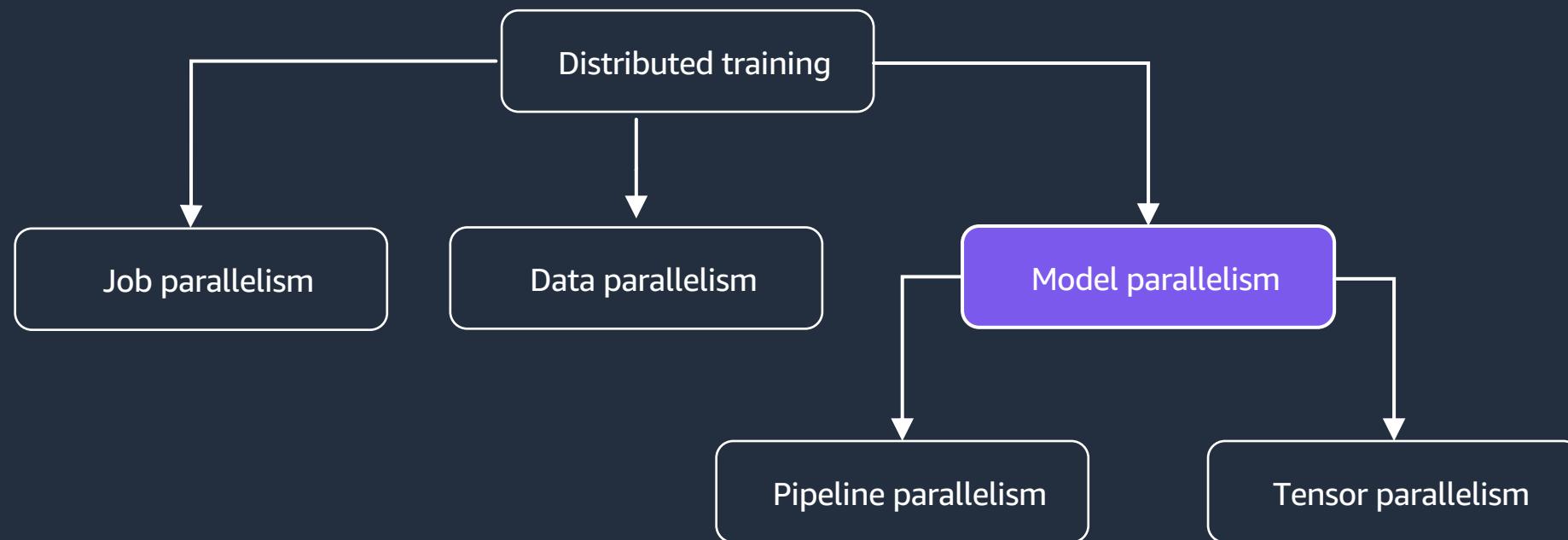


Amazon SageMaker Distributed Data Parallel

Model	Setup	Throughput			Scaling Efficiency		
		PT-DDP	SageMaker	Speed up	PT-DDP	SageMaker	Improvement
BERT Large (seqs/sec)	2 node p3dn.24xl	1752	2479	41%	64%	90%	26%
	4 node p3dn.24xl	3017	4603	52%	55%	84%	29%
	8 node p3dn.24xl	7409	8551	15%	67%	78%	11%
MaskRCNN (samples/sec)	2 node p3dn.24xl	152	158	4%	82%	85%	3%
	4 node p3dn.24xl	258	307	19%	70%	83%	13%
	8 node p3dn.24xl	545	617	13%	74%	84%	10%



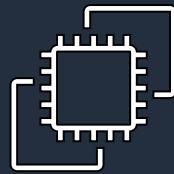
There are many kinds of distributed training



Model parallel, think “massive models”



Model parallelism on Amazon SageMaker (SMP)



Automated
model partitioning



Interleaved
pipelined training



Managed
SageMaker training



Clean
framework integration



Warm Pools: Faster startup time

Before: Multi-minute wait between script updates



After: Multi-second wait between script updates

`Keep_alive_period_in_seconds=600`



*TJ=Training Job

71



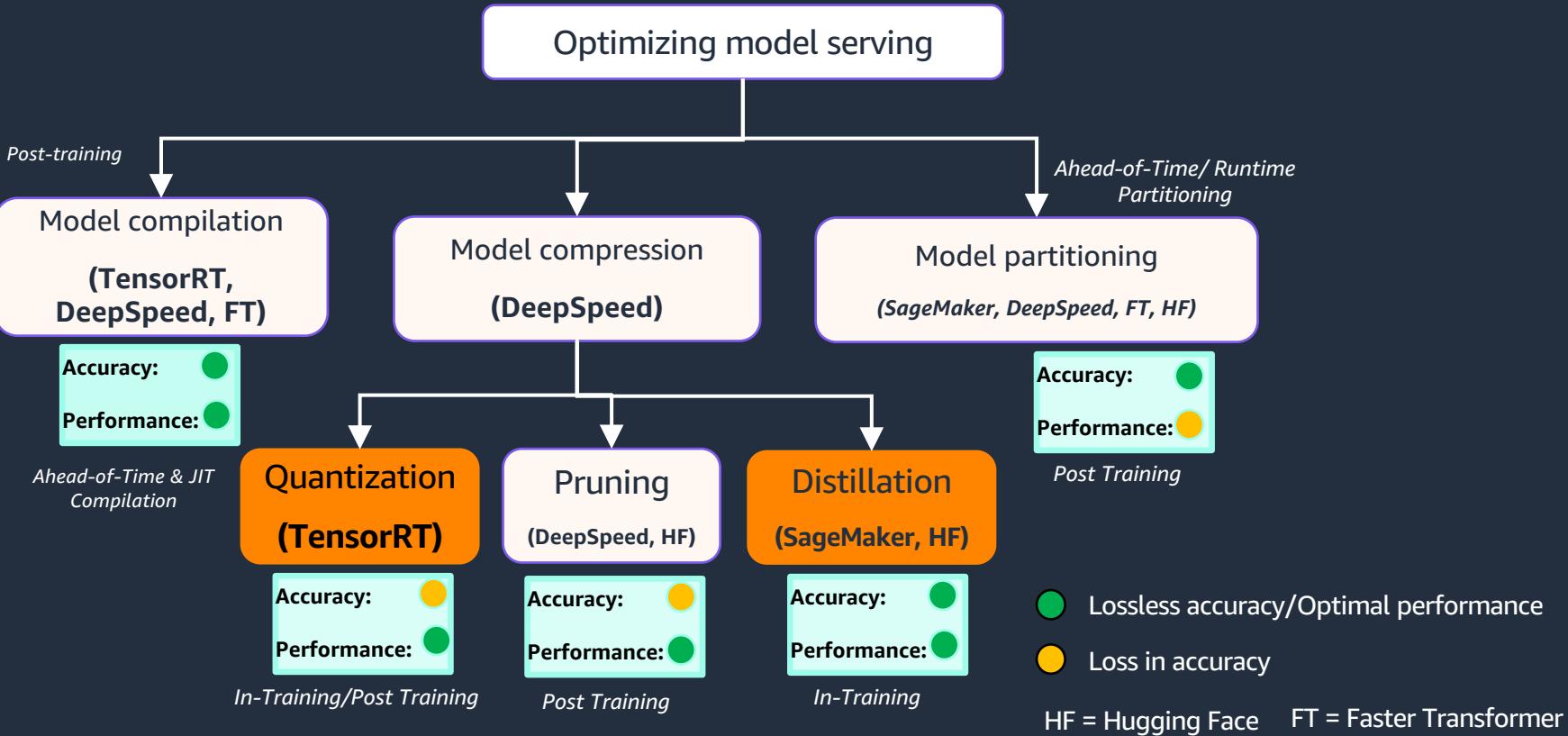
Deploy large models



Large model hosting challenges

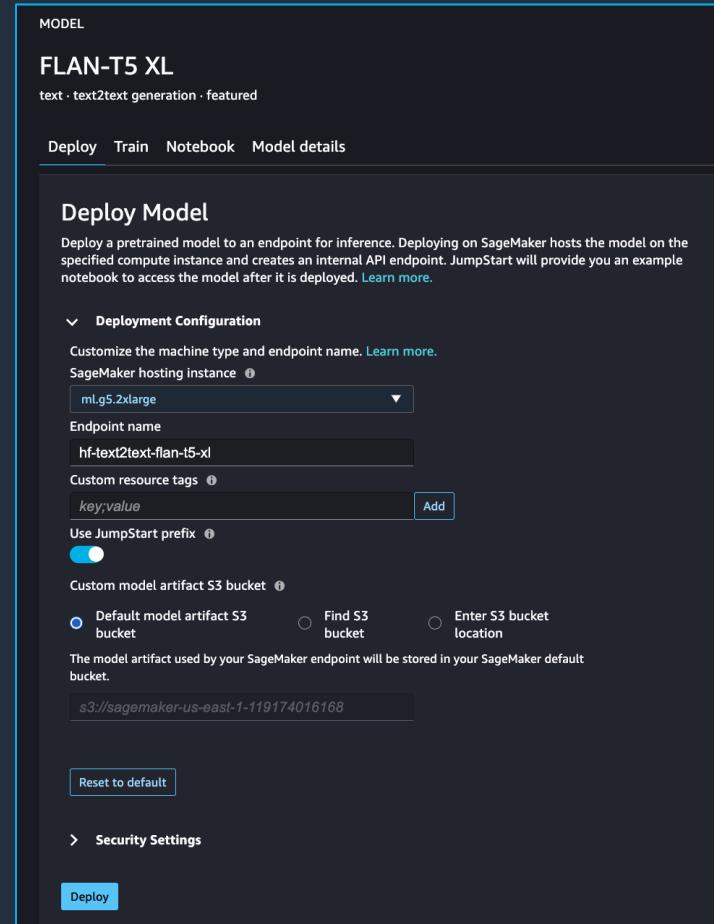


Large model inference optimization



Hosting Large Language Models

- **Scalability & Cost Efficiency:** Sagemaker JumpStart endpoints scale to handle any traffic, with a pay-as-you-go model
- **Rapid Deployment & High Availability:** Deploy ML models quickly, with built-in failover for high availability
- **Real-Time/Batch Inference & Monitoring:** Perform real-time and batch inference and monitor your model's performance
- **GPU Support & High-Performance:** Optimized performance for deep learning models and low-latency capabilities
- **Model Versioning & A/B Testing:** Switch between versions and compare performance with A/B testing
- **Autoscaling & Routing:** Autoscale based on policies and route traffic based on model weights
- **Seamless Integration:** Integrate with other AWS services and SageMaker components to build complex NLP pipelines





IBM

Thank you

Partha Dey

Enterprise Solutions Architect
AWS