

CONNEQT

13 Open-Source Tools for Foundation Model Deployment

The Complete Guide to Self-Hosting AI Models in 2026

١٣ أداة مفتوحة المصدر لنشر نماذج الذكاء الاصطناعي

الدليل الشامل لاستضافة نماذج الذكاء الاصطناعي ذاتيا في ٢٠٢٦

13 Open-Source Tools أداة مفتوحة المصدر	55K+ vLLM GitHub Stars نجوم vLLM	30-50% Cost vs Cloud APIs أقل تكلفة من APIs السحابية	100% Data Control تحكم كامل بالبيانات
--	---	---	--

2026 Edition | conneqtme.com

Table of Contents

جدول المحتويات

1. Why Self-Host Foundation Models?	لماذا تستضيف النماذج ذاتيا؟
2. Categories of Deployment Tools	فئات أدوات النشر
3. Quick Comparison Table	جدول المقارنة السريعة
4. Inference Engines (vLLM, SGLang, llama.cpp, TGI, Ollama)	محركات الاستدلال
5. Model Serving Frameworks (TF Serving, TorchServe, BentoML)	أطر تقديم النماذج
6. MLOps Platforms (MLflow, Metaflow, MLRun, Kubeflow)	منصات MLOps
7. Production Orchestration (Seldon Core)	تنسيق الإنتاج
8. Decision Matrix: Which Tool for Which Job	مصفوفة القرار
9. Architecture Patterns	أنماط البنية
10. GCC Deployment Considerations	اعتبارات النشر في الخليج

1. Why Self-Host Foundation Models?

١. لماذا تستضيف النماذج ذاتيا؟

Cloud APIs (OpenAI, Anthropic, Google) are convenient but come with trade-offs: data leaves your infrastructure, per-token costs compound at scale, rate limits constrain throughput, and vendor lock-in limits flexibility. Self-hosting gives you 100% data control, predictable costs, no rate limits, and the ability to fine-tune models for your specific domain. For GCC businesses operating under Saudi PDPL and other data residency regulations, self-hosting may be a compliance requirement.

ضبط النماذج دقيقا لمجالك. لشركات الخليج العاملة بموجب نظام حماية البيانات الشخصية السعودي قد تكون الاستضافة الذاتية متطلب امتثال. المعدل تقيد الإنتاجية وقفل المزود يحد المرونة. الاستضافة الذاتية تمنحك تحكما كاملا بالبيانات وتكاليف متوقعة وبدون حدود معدل وقدرة برمجة التطبيقات السحابية مربحة لكنها تأتي مع مقايضات: البيانات تغادر بنيتك والتحتية وتكاليف كل رمز تتراكم على نطاق واسع وحدود واجهات

Factor	Cloud API	Self-Hosted
Data Privacy	Data sent to third-party servers	Data never leaves your infrastructure
Cost at Scale	Per-token pricing compounds (100K+/day = expensive)	Fixed GPU cost — predictable regardless of volume
Latency	Network round-trip + queue wait	Direct GPU access — lowest possible latency
Customization	Limited to provider's models	Fine-tune, quantize, optimize for your use case
Rate Limits	Tokens/min, requests/min caps	Limited only by your hardware
Compliance	Depends on provider's certifications	Full control — meet any regulatory requirement

2. Categories of Deployment Tools

٢. فئات أدوات النشر

Category	What It Does	Tools
LLM Inference Engines	High-throughput serving of large language models with optimized GPU memory, batching, and KV-cache management	vLLM, SGLang, llama.cpp, HuggingFace TGI, Ollama
Model Serving Frameworks	Package and serve any ML model (not just LLMs) as API endpoints with scaling and monitoring	TensorFlow Serving, TorchServe, BentoML
MLOps Platforms	End-to-end ML lifecycle: experiment tracking, pipeline orchestration, model registry, deployment	MLflow, Metaflow, MLRun, Kubeflow
Production Orchestration	Kubernetes-native model deployment with A/B testing, canary rollouts, monitoring, and multi-model routing	Seldon Core

Key Insight: Most production systems combine tools from multiple categories. Example: vLLM (inference engine) + MLflow (experiment tracking) + Seldon Core (Kubernetes orchestration). The tools are complementary, not competing.

Seldon Core + (تنسيق Kubernetes). الأدوات مكملة وليست متنافسة. فئات متعددة. مثال: vLLM (محرك استدلال) + MLflow (تتبع التجارب) رؤية: معظم أنظمة الإنتاج تجمع أدوات من

3. Quick Comparison: All 13 Tools

٣. مقارنة سريعة: جميع الأدوات الـ ١٣

Tool	Category	Best For	Difficulty
vLLM	Inference Engine	Production LLM serving at scale — the industry default	Medium
SGLang	Inference Engine	Multi-turn chat, RAG, structured output — fastest TTFT	Medium
llama.cpp	Inference Engine	Local/edge deployment on CPU/consumer GPUs/mobile	Low
HF TGI	Inference Engine	Easy production serving — zero-config GPU optimization	Low
Ollama	Inference Engine	Local development, experimentation, desktop AI	Very Low
TF Serving	Model Serving	TensorFlow models in production (Google ecosystem)	Medium
TorchServe	Model Serving	PyTorch models in production (Meta ecosystem)	Medium
BentoML	Model Serving	Framework-agnostic model packaging and deployment	Low
MLflow	MLOps	Experiment tracking, model registry, reproducibility	Low
Metaflow	MLOps	ML pipelines for data scientists (Netflix-built)	Low
MLRun	MLOps	Full MLOps automation with serverless functions	Medium
Kubeflow	MLOps	Enterprise ML on Kubernetes — full platform	High
Seldon Core	Orchestration	K8s model deployment with A/B, canary, monitoring	High

4. Inference Engines

٤. محركات الاستدلال

Inference engines are the most critical choice — they determine throughput, latency, GPU utilization, and cost. These five tools handle the actual running of LLMs.

الأكثر أهمية — تحدد الإنتاجية والتأخير واستخدام GPU والتكلفة. هذه الأدوات الخمس تتعامل مع التشغيل الفعلي للنماذج اللغوية الكبيرة. محركات الاستدلال هي الاختيار

vLLM

vLLM

Category: LLM Inference Engine | Stars: 55K+ | License: Apache 2.0

What It Does

The industry-standard production LLM serving engine. Uses PagedAttention for GPU memory management (borrowed from OS virtual memory concepts) and continuous batching (new requests slot into running batches instantly). Powers xAI's Grok, Microsoft Azure endpoints, LinkedIn AI, Cursor's code completion, and Oracle Cloud. Speaks the OpenAI API format natively — swap in vLLM without rewriting client code.

لعمله اذا ما

Grok من xAI ونقاط Azure ومميزات LinkedIn AI واستكمال كود Cursor وOracle Cloud. يتحدث تنسيق OpenAI API أصليا. محرك تقديم LLM المعياري للإنتاج. يستخدم PagedAttention لإدارة ذاكرة GPU والدمج المستمر. يشغل

When to Use It

Production deployments serving 50+ concurrent users. Multi-tenant serving where many users share GPU fleet. When you need the largest ecosystem, most model support, and proven reliability at scale. The default choice unless you have a specific reason to pick something else.

هملختستى تم

عندما تحتاج أكبر نظام بيئي ودعم نماذج وموثوقية مثبتة على نطاق واسع. الخيار الافتراضي ما لم يكن لديك سبب محدد لاختيار شيء آخر. نشر الإنتاج الذي يخدم أكثر من 50 مستخدما متزامنا. التقديم متعدد المستأجرين.

Key Strengths

Largest community and ecosystem. Supports every major model (Llama, Mistral, Qwen, Falcon, etc.). Built-in quantization (AWQ, GPTQ, FP8). Llama 70B runs on a single 40GB GPU. OpenAI-compatible API means zero migration cost.

تيسيرلا قوقلا طاقدا

كل نموذج رئيسي. تكميم مدمج. Llama 70B يعمل على GPU واحد 40 جيجابايت. واجهة متوافقة مع OpenAI تعني صفر تكلفة ترحيل. أكبر مجتمع ونظام بيئي. يدعم

Limitations

Slightly slower first-token latency than SGLang for multi-turn conversations. NVIDIA-focused (AMD/Intel support is improving but lags). Requires GPU — no CPU-only mode for development.

دويقللا

تأخير أول رمز أبطأ قليلا من SGLang للمحادثات متعددة الأدوار. يركز على NVIDIA. يتطلب GPU — لا وضع CPU فقط للتطوير.

SGLang

SGLang

Category: LLM Inference Engine | Stars: 15K+ | License: Apache 2.0

What It Does

The fastest-growing inference engine, developed at UC Berkeley. Introduces RadixAttention — treats the KV cache as a tree (Radix Tree) for incredibly efficient prefix caching. If 10 users ask questions about the same document, SGLang processes that document once and reuses cached attention for all 10. Hits 80-120ms TTFT on single requests — 30-40% faster than vLLM for interactive use.

لماذا

المستند يعالجه SGLang مرة واحدة ويعيد استخدام الانتباه المخزن للعشرة. يصل إلى ٨٠-١٢٠ مللي ثانية — أسرع ٣٠-٤٠% من vLLM. كاليفورنيا بيركلي. يقدم RadixAttention — يعامل ذاكرة KV كشجرة للتخزين المؤقت الفعال جدا. إذا سأل ١٠ مستخدمين عن نفس أسرع محرك استدلال نمو من جامعة

When to Use It

RAG pipelines, multi-turn chatbots, and structured output (JSON mode). Applications where shared context is common (same system prompt, same documents). When first-token latency matters more than raw batch throughput.

مختصة

المحادثة متعددة الأدوار والمخرجات المهيكلة. التطبيقات ذات السياق المشترك. عندما يكون تأخير أول رمز أهم من إنتاجية الدفعة الخام. خطوط أنابيب RAG وروبوتات

Key Strengths

Best prefix caching in any engine. 30-40% faster TTFT than vLLM. Native function abstractions for multi-stage inference. Excellent structured output performance.

تيسيرا قوقلا طاقدا

أفضل تخزين مؤقت للبادئة في أي محرك. أسرع ٣٠-٤٠% في TTFT. تجريدات دوال أصلية للاستدلال متعدد المراحل.

Limitations

Smaller production footprint than vLLM. Newer ecosystem with fewer battle-tested deployments. Model support lags vLLM slightly for bleeding-edge releases.

دويقلا

بصمة إنتاج أصغر من vLLM. نظام بيئي أحدث مع نشرات أقل اختبارا. دعم النماذج يتأخر قليلا عن vLLM.

llama.cpp

llama.cpp

Category: Local/Edge Inference | Stars: 74K+ | License: Apache 2.0

What It Does

Pure C/C++ implementation for running LLMs on anything — CPUs, consumer GPUs, Raspberry Pi, Android phones, iPhones, and even web browsers via WebAssembly. Created by Georgi Gerganov in 2023, it's become the universal runtime for local AI. Supports GGUF quantization format (2-bit to 8-bit), enabling 70B models to run on 32GB RAM. The 1-bit weight support merged in 2026 enables massive memory savings.

لماذا

وأيفون وحتى متصفحات الويب. يدعم تكميم GGUF من ٢ إلى ٨ بت. دعم الأوزان ١ بت المدمج في ٢٠٢٦ يمكن توفيراً ضخماً في الذاكرة. تنفيذ ++C/C++ خالص لتشغيل النماذج على أي شيء — معالجات و GPU استهلاكية و Raspberry Pi ومواتف أندرويد

When to Use It

Local development and testing. Edge deployment (mobile, IoT, embedded). Running models on consumer hardware without enterprise GPUs. Privacy-sensitive applications where data must never leave the device.

مختبرتي

النشر على الحافة (جوال وإنترنت الأشياء). تشغيل النماذج على أجهزة استهلاكية بدون GPU مؤسسي. التطبيقات الحساسة للخصوصية. التطوير والاختبار المحلي.

Key Strengths

Runs on literally everything (CPU, GPU, mobile, browser). Smallest memory footprint of any engine. No NVIDIA dependency. Massive community (74K stars). Powers Ollama under the hood.

تيسيراً وقولاً طاقداً

يعمل حرفياً على كل شيء. أصغر بصمة ذاكرة. لا اعتماد على NVIDIA. مجتمع ضخم (٧٤ ألف نجمة). يشغل Ollama من الداخل.

Limitations

Not designed for high-concurrency production serving. Continuous batching is real but caps out faster than vLLM. Quantization trades accuracy for speed — not suitable for all use cases.

دويقلاً

غير مصمم لتقديم إنتاج عالي التزامن. الدمج المستمر حقيقي لكنه يصل لحدده أسرع من vLLM. التكميم يبادل الدقة بالسرعة.

Hugging Face TGI

Hugging Face TGI

Category: Production Inference | Stars: 10K+ | License: Apache 2.0

What It Does

Hugging Face's production inference server. Trades raw performance for ease of use — auto-handles tensor parallelism across GPUs with zero configuration. Built-in safety features: sampling distributions, repetition penalties, stop tokens. Hits ~2,500 tokens/sec on A100 for Llama 70B. Slightly slower than vLLM but dramatically easier to set up and maintain.

لماذا

يتعامل تلقائياً مع التوازي المنسوج عبر GPUs بدون تكوين. ميزات أمان مدمجة. يصل لحوالي ٢٥٠٠ رمز/ثانية على A100 لـ Llama 70B. خادم الاستدلال الإنتاجي من Hugging Face. يبادل الأداء الخام بسهولة الاستخدام —

When to Use It

Teams that want production inference without vLLM's operational overhead. When ease of deployment matters more than squeezing every last token/second. Organizations already deep in the Hugging Face ecosystem.

٥مدختستىتم

بدون عبء تشغيل vLLM. عندما تكون سهولة النشر أهم من استخراج كل رمز/ثانية. المنظمات العميقة في نظام Hugging Face البيئي. الفرق التي تريد استدلال إنتاجي

Key Strengths

Easiest production setup of any engine. Auto tensor parallelism. Deep Hugging Face Hub integration — pull any model instantly. Container or standalone.

٤يسينرلا قوقلا طاقذ

أسهل إعداد إنتاجي بين جميع المحركات. توازي منسوج تلقائي. تكامل عميق مع Hugging Face Hub.

Limitations

15-20% slower than vLLM at high concurrency. Fewer optimization knobs for advanced users. Model support can lag behind vLLM/llama.cpp for newest releases.

دويقللا

أبطأ ١٥-٢٠% من vLLM عند التزامن العالي. أقل خيارات تحسين للمستخدمين المتقدمين.

Ollama

Ollama

Category: Local Development | Stars: 100K+ | License: Apache 2.0

What It Does

The Docker of LLMs. One command to download and run any model locally: `ollama run llama3`. Wraps llama.cpp with a beautiful CLI, model library, API server, and desktop app. Powers local AI for hundreds of thousands of developers. The easiest way to get a model running on your machine in under 60 seconds.

ل عقيادام

وتشغيل أى نموذج محليا: `ollama run llama3`. يغلف llama.cpp بواجهة سطر أوامر جميلة ومكتبة نماذج وخادم API وتطبيق سطح مكتب. دوكر النماذج اللغوية الكبيرة. أمر واحد لتنزيل

When to Use It

Local development and prototyping. Personal AI assistants. When you need a model running locally in under 60 seconds. Developers who want to experiment with different models quickly without infrastructure setup.

٥مدختستىتم

بالذكاء الاصطناعي. عندما تحتاج نمودجا يعمل محليا في أقل من ٦٠ ثانية. للمطورين الذين يريدون التجريب مع نماذج مختلفة بسرعة. التطوير والنمذجة المحلية. المساعدون الشخصيون

Key Strengths

Lowest barrier to entry. Beautiful DX. Huge model library. Runs on Mac, Linux, Windows. Built-in model management. REST API for integration.

٤يسينرلا قوقلا طاقذ

أقل حاجز دخول. تجربة مطور ممتازة. مكتبة نماذج ضخمة. يعمل على ماك ولينكس وويندوز.

Limitations

Not designed for production multi-user serving. No continuous batching at vLLM scale. Limited GPU optimization compared to dedicated serving engines.

دويقلا

غير مصمم لتقديم إنتاج متعدد المستخدمين. لا دمج مستمر بحجم vLLM. تحسين GPU محدود مقارنة بمحركات التقديم المخصصة.

5. Model Serving Frameworks

5. أطر تقديم النماذج

TensorFlow Serving

TensorFlow Serving

Category: TensorFlow Model Serving | Stars: 6.5K+ | License: Apache 2.0

What It Does

Google's production serving system for TensorFlow models. Handles model versioning, A/B testing, and auto-scaling. Designed for high-throughput, low-latency serving of any TensorFlow model (not just LLMs) — vision models, recommendation systems, structured prediction. Used internally at Google for serving billions of predictions daily.

لماذا

مع إصدارات النماذج واختبار A/B والتوسع التلقائي. مصمم لتقديم أي نموذج TensorFlow — نماذج الرؤية وأنظمة التوصية والتنبؤ المهيكلة. نظام تقديم الإنتاج من جوجل لنماذج TensorFlow. يتعامل

When to Use It

TensorFlow models in production. Computer vision, recommendation, and structured prediction workloads. Google Cloud / GKE environments. When you need battle-tested model versioning and rollback.

مختبر على نطاق واسع

نماذج TensorFlow في الإنتاج. أعباء الرؤية الحاسوبية والتوصية. بيئات Google Cloud. عندما تحتاج إصدارات نماذج مختبرة.

Key Strengths

Battle-tested at Google scale. Excellent model versioning. gRPC + REST API. Automatic batching. Native Kubernetes support.

مختبر على نطاق واسع

مختبر على نطاق جوجول. إصدارات نماذج ممتازة. واجهة gRPC + REST. دمج تلقائي. دعم Kubernetes أصلي.

Limitations

TensorFlow-only. Not optimized for LLM-specific workloads. Heavier setup than BentoML.

مختبر على نطاق واسع

مقتصر على TensorFlow. غير محسن لأعباء LLM. إعداد أثقل من BentoML.

TorchServe

TorchServe

Category: PyTorch Model Serving | Stars: 4.5K+ | License: Apache 2.0

What It Does

Meta and AWS's production serving solution for PyTorch models. Provides model management, multi-model serving, A/B testing, logging, and monitoring. Supports custom handlers for pre/post-processing. Increasingly important as PyTorch is the dominant training framework.

لماذا

من ميتا و AWS لنماذج PyTorch. يوفر إدارة النماذج والتقديم متعدد النماذج واختبار A/B والتسجيل والمراقبة. يدعم معالجات مخصصة. حل تقديم الإنتاج

When to Use It

PyTorch models in production. AWS SageMaker environments. When you need tight integration with the PyTorch ecosystem and AWS infrastructure.

ملخص سريع

نماذج PyTorch في الإنتاج. بيئات AWS SageMaker. عندما تحتاج تكامل وثيق مع نظام PyTorch البيئي وبنية AWS التحتية.

Key Strengths

Native PyTorch integration. AWS backing. Multi-model serving. Built-in model management. Good logging and metrics.

تيسيرا قوقلا طاقذ

تكاملا PyTorch أصلي. دعم AWS. تقديم متعدد النماذج. إدارة نماذج مدمجة.

Limitations

PyTorch-only. Less community momentum than vLLM for LLM workloads. More complex setup than BentoML.

دويقل

مقتصر على PyTorch. زخم مجتمعي أقل. إعداد أكثر تعقيدا من BentoML.

BentoML

BentoML

Category: Framework-Agnostic Model Serving | Stars: 8K+ | License: Apache 2.0

What It Does

Framework-agnostic model packaging and deployment. Works with any ML framework (PyTorch, TensorFlow, scikit-learn, XGBoost, LightGBM, Hugging Face). Packages models as 'Bentos' — standardized containers that include the model, pre/post-processing code, dependencies, and API definitions. Deploy anywhere: Docker, Kubernetes, or BentoCloud (managed).

ل عقي اذام

على إطار. يعمل مع أي إطار ML. يغلف النماذج كـ 'Bentos' — حاويات معيارية تشمل النموذج وكود المعالجة والتبعيات وتعريفات API. تغليف ونشر نماذج لا يعتمد

When to Use It

Multi-framework environments. When your stack includes models from different frameworks. Teams that want the simplest path from notebook to production. Startups and small teams that can't afford complex Kubernetes setups.

ملخص سريع

بيئات متعددة الأطر. عندما يشمل مكدسك نماذج من أطر مختلفة. الفرق التي تريد أبسط مسار من الدفتر إلى الإنتاج.

Key Strengths

Works with ANY ML framework. Easiest packaging workflow. Built-in adaptive batching. Python-native (feels natural). Active community.

تيسيرا قوقلا طاقذ

يعمل مع أي إطار ML. أسهل سير عمل تغليف. دمج تكييفي مدمج. أصلي ببايثون.

Limitations

Less raw throughput than vLLM for LLM-specific workloads. Smaller ecosystem than TF Serving/TorchServe for their respective frameworks.

دويقلدا

إنتاجية خام أقل من vLLM لأعباء LLM. نظام بيئي أصغر لكل إطار.

6. MLOps Platforms

٦. منصات MLOps

MLflow

MLflow

Category: Experiment Tracking & Model Registry | Stars: 19K+ | License: Apache 2.0

What It Does

The most popular open-source MLOps platform. Tracks experiments (parameters, metrics, artifacts), manages model versions in a central registry, and supports deployment to multiple targets. Originally from Databricks, now used by thousands of organizations. Works with any ML framework and any deployment target.

لماذا

والمنتجات وإدارة إصدارات النماذج في سجل مركزي ودعم النشر لأهداف متعددة. من Databricks الأصل وتستخدمها آلاف المنظمات. أشهر منصة MLOps مفتوحة المصدر. تتبع التجارب (المعلمات والمقاييس

When to Use It

ANY ML project that needs experiment tracking. When you need a model registry for versioning and governance. Teams that want reproducibility without heavy infrastructure. The default choice for experiment tracking in 2026.

مختصة

أي مشروع ML يحتاج تتبع تجارب. عندما تحتاج سجل نماذج للإصدارات والحوكمة. الخيار الافتراضي لتتبع التجارب في ٢٠٢٦.

Key Strengths

Universal — works with everything. Lightweight (pip install mlflow). Excellent UI. Huge community. Databricks integration.

مميزات

عالمي — يعمل مع كل شيء. خفيف الوزن. واجهة ممتازة. مجتمع ضخم. تكامل Databricks.

Limitations

Not an inference engine — you still need vLLM/TGI for serving. Scaling the tracking server requires some ops work.

عيوب

ليس محرك استدلال — لا تزال تحتاج vLLM/TGI للتقديم. توسيع خادم التتبع يتطلب بعض العمل التشغيلي.

Metaflow

Metaflow

Category: ML Pipelines (Netflix-Built) | Stars: 8.5K+ | License: Apache 2.0

What It Does

Built by Netflix for their ML infrastructure, then open-sourced. Designed to make it easy for data scientists to build and deploy ML pipelines without DevOps expertise. Write Python, decorate with @step, and Metaflow handles versioning, dependency management, and cloud scaling automatically.

لماذا

للباحثين بدون خبرة DevOps. اكتب بايثون وزخرف بـ step@ و Metaflow يتعامل مع الإصدارات وإدارة التبعيات والتوسع السحابي تلقائياً. بنته Netflix لبنيتهم التحتية للتعلم الآلي ثم فتحت مصدره. مصمم لتسهيل بناء ونشر خطوط أنابيب ML

When to Use It

Data science teams that want pipeline orchestration without Kubernetes complexity. When data scientists need to deploy without ops support. AWS-heavy environments.

مختصة في

البيانات التي تريد تنسيق خطوط الأنابيب بدون تعقيد Kubernetes. عندما يحتاج علماء البيانات للنشر بدون دعم عمليات. بيئات AWS. فرق علوم

Key Strengths

Python-native — feels like writing regular code. Built-in versioning. Netflix-proven at scale. AWS integration (Step Functions, Batch). Human-friendly.

تيسيراً قوفاً طاقداً

أصلي بايثون. إصدارات مدمجة. مثبت من Netflix على نطاق واسع. تكامل AWS.

Limitations

AWS-centric (GCP/Azure support exists but is weaker). Not a full MLOps platform (no model registry). Smaller community than MLflow.

دقيقاً

يركز على AWS. ليس منصة MLOps كاملة (لا سجل نماذج). مجتمع أصغر من MLflow.

MLRun

MLRun

Category: Full MLOps Automation | Stars: 1.5K+ | License: Apache 2.0

What It Does

An open-source MLOps framework for building and managing ML applications across their lifecycle. Provides serverless functions, auto-scaling pipelines, feature store, model monitoring, and automated retraining. The most comprehensive single-platform MLOps solution in the open-source space.

لحفظ اذام

وإدارة تطبيقات ML عبر دورة حياتها. يوفر دوال بدون خادم وخطوط أنابيب تتوسع تلقائياً ومخزن ميزات ومراقبة نماذج وإعادة تدريب آلية. إطار MLOps مفتوح المصدر لبناء

When to Use It

Teams that want an all-in-one MLOps platform. When you need feature store + pipeline + monitoring in one tool. Organizations willing to invest in a comprehensive platform.

مختصة في

الفرق التي تريد منصة MLOps شاملة. عندما تحتاج مخزن ميزات + خط أنابيب + مراقبة في أداة واحدة.

Key Strengths

Most complete open-source MLOps platform. Built-in feature store. Auto-scaling serverless. Model monitoring with drift detection.

تيسيراً قوفاً طاقداً

أكمل منصة MLOps مفتوحة المصدر. مخزن ميزات مدمج. بدون خادم يتوسع تلقائياً. مراقبة النماذج مع كشف الانحراف.

Limitations

Steeper learning curve than MLflow. Smaller community. Requires Kubernetes for full capabilities.

دويقلًا

منحنى تعلم أكثر حدة. مجتمع أصغر. يتطلب Kubernetes للقدرات الكاملة.

Kubeflow

Kubeflow

Category: Enterprise ML on Kubernetes | Stars: 14.5K+ | License: Apache 2.0

What It Does

The most comprehensive Kubernetes-native ML platform. Provides Jupyter notebooks, pipeline orchestration (Kubeflow Pipelines), hyperparameter tuning (Katib), model serving (KServe), and feature management. Originally from Google, now a CNCF project. The enterprise standard for organizations running Kubernetes.

لغير اذام

دفاثر Jupyter وتنسيق خطوط الأنابيب وضبط المعلمات الفائقة وتقديم النماذج وإدارة الميزات. من جوجل الأصل وهو الآن مشروع CNCF. أشمل منصة ML أصلية على Kubernetes. يوفر

When to Use It

Enterprise organizations with Kubernetes expertise. When you need a full ML platform on top of existing K8s infrastructure. Large teams with dedicated ML engineering resources.

مدمختستى تم

ذات خبرة Kubernetes. عندما تحتاج منصة ML كاملة فوق بنية K8s التحتية الحالية. الفرق الكبيرة ذات موارد هندسة ML مخصصة. منظمات المؤسسات

Key Strengths

Most feature-complete ML platform. CNCF backing. Enterprise-grade. Katib for HPO. KServe for model serving. Pipeline orchestration.

ةيسيرلًا قوقلًا طاقذ

أكمل منصة ML ميزات. دعم CNCF. مستوى مؤسسات. Katib لتحسين المعلمات. KServe لتقديم النماذج.

Limitations

Extremely complex to set up and maintain. Requires dedicated DevOps/MLOps team. Overkill for small-medium teams. Steep learning curve.

دويقلًا

معقد للغاية للإعداد والصيانة. يتطلب فريق DevOps/MLOps مخصص. مبالغ فيه للفرق الصغيرة والمتوسطة.

7. Production Orchestration

٧. تنسيق الإنتاج

Seldon Core

Seldon Core

Category: K8s Model Deployment Orchestration | Stars: 4.5K+ | License: Apache 2.0

What It Does

Kubernetes-native platform for deploying, scaling, and monitoring ML models in production. Provides advanced deployment strategies (A/B testing, canary rollouts, shadow deployments, multi-armed bandits), model explainability, and outlier detection. Works with any inference engine — wrap vLLM, TGI, or any model in a Seldon deployment for production-grade orchestration.

لماذا

نماذج ML في الإنتاج. توفر استراتيجيات نشر متقدمة (اختبار A/B والنشر التدريجي والنشر الظلي) وتفسير النماذج وكشف القيم الشاذة. منصة أصلية على Kubernetes لنشر وتوسيع ومراقبة

When to Use It

Production ML systems that need advanced deployment strategies. When you need A/B testing, canary rollouts, or shadow deployments. Organizations with existing Kubernetes infrastructure and multiple models to manage.

ملخص

أنظمة ML الإنتاجية التي تحتاج استراتيجيات نشر متقدمة. عندما تحتاج اختبار A/B والنشر التدريجي والنشر الظلي.

Key Strengths

Best deployment strategies in any open-source tool. Model explainability built-in. Outlier detection. Multi-model routing. Prometheus/Grafana monitoring.

تفسيرات وقواعد

أفضل استراتيجيات نشر في أي أداة مفتوحة المصدر. تفسير النماذج مدمج. كشف القيم الشاذة. توجيه متعدد النماذج.

Limitations

Kubernetes-only. Complex setup. Requires DevOps expertise. Not needed for simple single-model deployments.

دوكلات

مقتصر على Kubernetes. إعداد معقد. يتطلب خبرة DevOps. غير مطلوب لنشر نموذج واحد بسيط.

8. Decision Matrix: Which Tool for Which Job

٨. مصفوفة القرار: أي أداة لأي مهمة

Scenario	Use This	Why
Production LLM API (high traffic)	vLLM + MLflow	Industry default. Proven at scale. Best ecosystem.
RAG chatbot / multi-turn	SGLang + MLflow	RadixAttention = best prefix caching. Fastest TTFT.
Local dev / prototyping	Ollama	60 seconds to running model. Best DX.
Edge / mobile deployment	llama.cpp	Runs on anything. Smallest footprint.
Easy prod serving (small team)	TGI + BentoML	Zero-config GPU optimization. Simple packaging.
Multi-model ML platform	Kubeflow + Seldon	Full platform + advanced deployment strategies.
Data science pipelines	Metaflow + MLflow	Python-native pipelines + experiment tracking.
PyTorch models (AWS)	TorchServe + MLflow	Native PyTorch + AWS integration.
TensorFlow models (GCP)	TF Serving + MLflow	Native TF + Google ecosystem.
GCC / Saudi data residency	vLLM on local GPU + MLflow	100% data control. PDPL compliant.

Key Insight: Start simple: Ollama for development, vLLM for production. Add MLflow for tracking. Only add Kubeflow/Seldon when your team and model count justify the operational complexity.

فقط عندما يبرر حجم فريقك وعدد نماذجك التعقيد التشغيلي.
و vLLM للإنتاج. أضف MLflow للتتبع. أضف Kubeflow/Seldon
رؤية: ابدأ بسيطاً: Ollama للتطوير

9. Architecture Patterns & 10. GCC Considerations

٩-١٠. أنماط البنية واعتبارات الخليج

Pattern 1: Simple Production Stack

Ollama (dev) → vLLM (prod) → MLflow (tracking) → NGINX (load balancer). This covers 80% of use cases. Deploy on a single server with 1-4 GPUs. Cost: \$2-8K/month on cloud GPU instances or one-time hardware investment.

النمط ١: مكس إنتاج بسيط

يغطي ٨٠% من حالات الاستخدام. انشر على خادم واحد مع ١-٤ (NGINX) → (تتبع) MLflow → (إنتاج) vLLM → (تطوير) Ollama. موازن تحميل). GPUs.

Pattern 2: Enterprise ML Platform

Kubeflow (platform) → Seldon Core (orchestration) → vLLM/TGI (inference) → MLflow (registry) → Prometheus/Grafana (monitoring). For organizations with 10+ models, multiple teams, and Kubernetes infrastructure. Requires dedicated MLOps engineering.

النمط ٢: منصة ML مؤسسية

متعددة وبنية) Prometheus/Grafana → (سجل) MLflow → (استدلال) vLLM/TGI → (تنسيق) Seldon Core → (منصة) Kubeflow. للمنظمات ذات أكثر من ١٠ نماذج وفرق Kubernetes. مراقبة).

Pattern 3: Edge + Cloud Hybrid

llama.cpp (edge/mobile for fast inference) → vLLM (cloud for heavy workloads) → API gateway (routes requests based on complexity). Small models run locally for privacy and speed; complex queries escalate to cloud GPUs.

النمط ٣: هجين حافة + سحابة

الصغيرة تعمل محليا للخصوصية والسرعة) API سحابة للأعباء الثقيلة) → (بوابة) vLLM → (حافة/جوال للاستدلال السريع) llama.cpp. توجه الطلبات حسب التعقيد). النماذج.

Saudi PDPL Compliance

Self-hosting on Saudi-based infrastructure ensures full PDPL compliance. Deploy vLLM on local GPU servers or Saudi cloud providers. Use MLflow for audit trails. All data stays within the Kingdom — no cross-border transfer concerns.

امتثال نظام حماية البيانات السعودي

انشر vLLM على خوادم GPU محلية أو مزودي سحابة سعوديين. استخدم MLflow لمسارات التدقيق. جميع البيانات تبقى داخل المملكة. الاستفادة الذاتية على بنية تحتية في السعودية تضمن امتثال PDPL الكامل.

Arabic Language Models

Fine-tune open models (Qwen, Llama, Falcon) on Arabic data using these tools. vLLM and SGLang support all major Arabic-capable models. llama.cpp enables running Arabic models on consumer hardware for testing. The GCC has a unique opportunity to build Arabic-first AI infrastructure.

نماذج اللغة العربية

وSGLang يدعمان جميع النماذج الرئيسية القادرة على العربية. لدى الخليج فرصة فريدة لبناء بنية تحتية للذكاء الاصطناعي عربية أولا. اضبط النماذج المفتوحة دقيقا (Falcon وLlama وQwen) على بيانات عربية باستخدام هذه الأدوات. vLLM.

Need help deploying AI models for your business?

Conneqt builds self-hosted AI infrastructure for GCC businesses — from model selection to production deployment.

Visit conneqtme.com or contact us for a free consultation.

هل تحتاج مساعدة في نشر نماذج الذكاء الاصطناعي لعملك؟
كونكت تبني بنية تحتية للذكاء الاصطناعي مستضافة ذاتيا لشركات الخليج.
قم بزيارة conneqtme.com أو تواصل معنا لاستشارة مجانية.

conneqtme.com | 2026 Edition

كونكت | نسخة ٢٠٢٦