Red Hat Enterprise Linux AI 1.1

Getting Started

Introduction to RHEL AI with product architecture and hardware requirements

Last Updated: 2024-10-23

Red Hat Enterprise Linux AI 1.1 Getting Started

Introduction to RHEL AI with product architecture and hardware requirements

Legal Notice

Abstract

This document provides introductory information for Red Hat Enterprise Linux AI.

Table of Contents

CHAPTER 1. RED HAT ENTERPRISE LINUX AI OVERVIEW 3
1.1. COMMON TERMS FOR RED HAT ENTERPRISE LINUX AI 3
1.2. INSTRUCTLAB AND RHEL AI 4
1.2.1. Introduction to skills and knowledge 4
1.2.1.1. Knowledge 4
1.2.1.2. Skills 4

CHAPTER 2. RED HAT ENTERPRISE LINUX AI PRODUCT ARCHITECTURE 6
2.1. BOOTABLE RED HAT ENTERPRISE LINUX WITH INSTRUCTLAB 6
2.1.1. InstructLab model alignment 6
2.1.2. Open source licensed Granite models 6

CHAPTER 3. RED HAT ENTERPRISE LINUX AI HARDWARE REQUIREMENTS 8
3.1. HARDWARE REQUIREMENTS FOR END-TO-END WORKFLOW OF GRANITE MODELS 8
3.1.1. Bare metal 8
3.1.2. Amazon Web Services (AWS) 8
3.2. HARDWARE REQUIREMENTS FOR INFERENCE SERVING GRANITE MODELS 9
3.2.1. Bare metal 9
3.2.2. Amazon Web Services (AWS) 9
3.2.3. IBM cloud 9

CHAPTER 1. RED HAT ENTERPRISE LINUX AI OVERVIEW

Red Hat Enterprise Linux AI is a platform that allows you to develop enterprise applications on open source Large Language Models (LLMs). RHEL AI is built from the Red Hat InstructLab open source project. For more detailed information about InstructLab, see the "InstructLab and RHEL AI" section.

Red Hat Enterprise Linux AI allows you to do the following:

- Host an LLM and interact with the open source Granite family of Large Language Models (LLMs).
- Using the LAB method, create and add your own knowledge data in a Git repository and fine-tune a model with that data with minimal machine learning background.
- Interact with the model that has been fine-tuned with your data.

Red Hat Enterprise Linux AI empowers you to contribute directly to LLMs. This allows you to easily and efficiently build AI-based applications, including chatbots.

1.1. COMMON TERMS FOR RED HAT ENTERPRISE LINUX AI

This glossary defines common terms for Red Hat Enterprise Linux AI:

InstructLab

InstructLab is an open source project that provides a platform for easy engagement with AI Large Language Models (LLM) by using the **ilab** command-line interface (CLI) tool.

Large Language Models

Known as LLMs, is a type of artificial intelligence that is capable of language generation or other processing tasks.

Synthetic Data Generation (SDG)

A process where large LLMs (Large Language Models) are used to generate artificial data that then can be used to train other LLMs.

Fine-tuning

A technique where an LLM is trained to meet a specific objective: to know particular information or be able to do a particular thing.

LAB

An acronym for "**L**arge-Scale **A**lignment for Chat**B**ots." Invented by IBM Research, LAB is a novel synthetic data-based alignment tuning and multi-phase training method for LLMs. InstructLab implements the LAB method during synthetic generation and training.

Multi-phase training

A fine-tuning strategy that the LAB method implements. During this process, a model is fine-tuned on multiple datasets in separate phases. The model evaluates the performance of all the checkpoints from a given phase. The best performing checkpoint is then used for training in the following phase. The fully fine-tuned model is the best performing checkpoint from the final phase.

Serving

Often referred to as "serving a model", is the deployment of an LLM or trained model to a server. This process gives you the ability to interact with models as a chatbot.

Inference

When serving and chatting with a model, inferencing is when a model can process and produce outputs from input data.

Taxonomy

The LAB method is driven by taxonomies, an information classification method. On RHEL AI, you can customize a taxonomy tree that enables you to create models fine-tuned with your own data.

Granite

An open source (Apache 2.0) Large Language Model trained by IBM. On RHEL AI you can download the **granite-7b-starter** model as a base LLM for customizing.

PyTorch

An optimized tensor library for deep learning on GPUs and CPUs.

vLLM

A memory-efficient inference and serving engine library for LLMs.

DeepSpeed

A Python library for optimized and distributed LLM training and fine-tuning.

1.2. INSTRUCTLAB AND RHEL AI

InstructLab is an open source AI project that facilitates contributions to Large Language Models (LLMs). RHEL AI takes the foundation of the InstructLab project and builds an enterprise platform for LLM integration on applications. Red Hat Enterprise Linux AI targets high performing server platforms with dedicated Graphic Processing Units (GPUs). InstructLab is intended for small scale platforms, including laptops and personal computers.

InstructLab implements the LAB (Large-scale Alignment for chatBots) technique, a novel synthetic data-based fine-tuning method for LLMs. The LAB process consists of several components:

- A taxonomy-guided synthetic data generation process
- A multi-phase training process
- A fine-tuning framework

RHEL AI and InstructLab allow you to customize an LLM with domain-specific knowledge for your distinct use cases.

1.2.1. Introduction to skills and knowledge

Skill and knowledge are the types of data that you can add to the taxonomy tree. You can then use these types to create a custom LLM model fine-tuned with your own data.

1.2.1.1. Knowledge

Knowledge for an AI model consists of data and facts. When creating knowledge sets for a model, you are providing it with additional data and information so the model can answer questions more accurately. Where skills are the information that trains an AI model on how to do something, knowledge is based on the model's ability to answer questions that involve facts, data, or references. For example, you can create a data set that includes a product's documentation and the model can learn the information provided in that documentation.

1.2.1.2. Skills

IMPORTANT

RHEL AI version 1.1 currently does not support customizing skills in your taxonomy.

A skill is a capability domain that intends to train the AI model on submitted information. When you make a skill, you are teaching the model how to do a task. Skills on RHEL AI are split into categories:

- Composition skill: Compositional skills allow AI models to perform specific tasks or functions. There are two types of compositional skills:
  - Freeform compositional skills: These are performative skills that do not require additional context or information to function.
  - Grounded compositional skills: These are performative skills that require additional context. For example, you can teach the model to read a table, where the additional context is an example of the table layout.
- Foundation skills: Foundational skills are skills that involve math, reasoning, and coding.

CHAPTER 2. RED HAT ENTERPRISE LINUX AI PRODUCT ARCHITECTURE

Red Hat Enterprise Linux AI contains various distinct features and consists of the following components.

2.1. BOOTABLE RED HAT ENTERPRISE LINUX WITH INSTRUCTLAB

You can install RHEL AI and deploy the InstructLab tooling using a bootable RHEL container image provided by Red Hat. The current supported installation methods for this image are on Amazon Web Services (AWS), IBM Cloud, and bare-metal machines with NVIDIA GPUs.

This RHEL AI image includes InstructLab, RHEL 9.4, and various inference and training software, including vLLM and DeepSpeed. After you boot this image, you can download various Red Hat and IBM developed Granite models to serve or train. The image and all the tools are compiled to specific Independent Software Vendor (ISV) hardware. For more information about the architecture of the image, see Installation overview

IMPORTANT

RHEL AI currently only includes bootable images for NVIDIA accelerators.

2.1.1. InstructLab model alignment

The Red Hat Enterprise Linux AI bootable image contains InstructLab and its tooling. InstructLab uses a novel approach to LLM fine-tuning called LAB (Large-Scale Alignment for ChatBots). The LAB method uses a taxonomy-based system that implements high-quality synthetic data generation (SDG) and multi-phase training.

Using the RHEL AI command line interface (CLI), which is built from the InstructLab CLI, you can create your own custom LLM by tuning a Granite base model on synthetic data generated from your own domain-specific knowledge.

For general availability, the RHEL AI LLMs customization workflow consists of the following steps:

1. Installing and initializing RHEL AI on your preferred platform.
2. Using a CLI and Git workflow for adding skills and knowledge to your taxonomy tree.
3. Running synthetic data generation (SDG) using the **mixtral-8x7B-Instruct** teacher model. SDG can generate hundreds or thousands of synthetic question-and-answer pairs for model tuning based on user-provided specific samples.
4. Using the InstructLab to train the base model with the new synthetically generated data. The **prometheus-8x7B-V2.0** judge model evaluates the performance of the newly trained model.
5. Using InstructLab with vLLM to serve the new custom model for inferencing.

2.1.2. Open source licensed Granite models

With RHEL AI, you can download the open source licensed IBM Granite family of LLMs.

Using the **granite-7b-starter** model as a base, you can create your model using knowledge data. You can keep these custom LLMs private or you can share them with the AI community.

Red Hat Enterprise Linux AI also allows you to serve and chat with Granite models created and fine-tuned by Red Hat and IBM.

CHAPTER 3. RED HAT ENTERPRISE LINUX AI HARDWARE REQUIREMENTS

Various hardware accelerators require different requirements for serving and inferencing as well as installing, generating and training the **granite-7b-starter** model on Red Hat Enterprise Linux AI.

3.1. HARDWARE REQUIREMENTS FOR END-TO-END WORKFLOW OF GRANITE MODELS

The following charts show the hardware requirements for running the full InstructLab end-to-end workflow to customize the Granite student model. This includes: synthetic data generation (SDG), training, and evaluating a custom Granite model.

3.1.1. Bare metal

Hardware vendor	Supported accelerators (GPUs)	Aggregate GPU memory	Recommended additional disk storage
NVIDIA	2xA100 4xA100 8xA100	160 GB 320 GB 640 GB	1 TB
NVIDIA	2xH100 4xH100 8xH100	160 GB 320 GB 640 GB	1 TB
NVIDIA	4xL40S 8xL40S	192 GB 384 GB	1 TB

3.1.2. Amazon Web Services (AWS)

Hardware vendor	Supported accelerators (GPUs)	Aggregate GPU Memory	AWS Instance	Recommended additional disk storage
NVIDIA	8xA100	640 GB	p4de.24xlarge	1 TB
NVIDIA	8xH100	640 GB	p5.48xlarge	1 TB

3.2. HARDWARE REQUIREMENTS FOR INFERENCE SERVING GRANITE MODELS

The following charts display the minimum hardware requirements for inference serving a model on Red Hat Enterprise Linux AI.

3.2.1. Bare metal

Hardware vendor	Supported accelerators (GPUs)	minimum Aggregate GPU memory	Recommended additional disk storage
NVIDIA	A100	80 GB	1 TB
NVIDIA	H100	80 GB	1 TB
NVIDIA	L40S	48 GB	1 TB
NVIDIA	L4	24 GB	1 TB

3.2.2. Amazon Web Services (AWS)

Hardware vendor	Supported accelerators (GPUs)	Minimum Aggregate GPU Memory	Recommended additional disk storage
NVIDIA	A100	80 GB	1 TB
NVIDIA	H100	80 GB	1 TB
NVIDIA	L40S	48 GB	1 TB
NVIDIA	L4	24 GB	1 TB

3.2.3. IBM cloud

Hardware vendor	Supported accelerators (GPUs)	Minimum Aggregate GPU Memory	Recommended additional disk storage
NVIDIA	L40S	48 GB	1 TB
NVIDIA	L4	24 GB	1 TB