



Red Hat Enterprise Linux AI 1.5

Getting Started

Introduction to RHEL AI with product architecture

Red Hat Enterprise Linux AI 1.5 Getting Started

Introduction to RHEL AI with product architecture

Legal Notice

Copyright © 2025 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

This document provides introductory information for Red Hat Enterprise Linux AI. This includes an overview of RHEL AI and the product architecture

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 5

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 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 or skills data in a Git repository. Then, fine-tune a model on that data with minimal machine learning background.
- Interact with the model that is fine-tuned with your data.

Red Hat Enterprise Linux AI empowers you to contribute directly to Large Language Models (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 (LLMs) 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 task processing.

Synthetic Data Generation (SDG)

A process where large LLMs (Large Language Models) are used, with human generated samples, 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 to do a particular task.

LAB

An acronym for "Large-Scale Alignment for ChatBots." Invented by IBM Research, LAB is a novel synthetic data-based and multi-phase training fine-tuning 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 trains in multiple phases called epochs, which save as checkpoints. 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, deduct, and produce outputs based on 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 family models 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.

FSDP

An acronym for [Fully Shared Data Parallels](#). The Pytorch tool FSDP can distribute computing power across multiple devices on your hardware. This optimizes the training process and makes fine-tuning faster and more memory efficient. This tool shares the functionalities of DeepSpeed.

DeepSpeed

A Python library for optimizes LLM training and fine-tuning by distributing computing resources on multiple devices. This tool shares the functionalities of FSDP. Deepspeed is currently the recommended hardware off loader for NVIDIA machines.

1.2. INSTRUCTLAB AND RHEL AI

InstructLab is an open source AI project that facilitates contributions to Large Language Models. 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 with greater accuracy. 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

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:

- **Compositional 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 layout of the table.
- **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.

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 Vendors (ISV) hardware. For more information about the architecture of the image, see [Installation overview](#).

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 starter Granite model as a base, you can create your model using knowledge or skills 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.

