



Red Hat Enterprise Linux AI 1.2

CLI reference

RHEL AI command line interface (CLI) reference

Red Hat Enterprise Linux AI 1.2 CLI reference

RHEL AI command line interface (CLI) reference

Legal Notice

Copyright © 2024 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 a reference for the CLI commands used in RHEL AI.

Table of Contents

CHAPTER 1. RED HAT ENTERPRISE LINUX AI COMMAND LINE INTERFACE REFERENCE	3
1.1. RED HAT ENTERPRISE LINUX AI CLI COMMANDS	3
1.1.1. ilab config	3
1.1.1.1. ilab config init	3
1.1.1.2. ilab config show	3
1.1.1.3. ilab config edit	3
1.1.2. ilab data	3
1.1.2.1. ilab data generate	4
1.1.2.2. ilab data list	4
1.1.3. ilab model	4
1.1.3.1. ilab model chat	4
1.1.3.2. ilab model download	5
1.1.3.3. ilab model evaluate	5
1.1.3.4. ilab model list	5
1.1.3.5. ilab model train	5
1.1.3.6. ilab model serve	5
1.1.4. ilab system	6
1.1.4.1. ilab system info	6
1.1.5. ilab taxonomy	6
1.1.5.1. ilab taxonomy diff	6

CHAPTER 1. RED HAT ENTERPRISE LINUX AI COMMAND LINE INTERFACE REFERENCE

This reference provides descriptions and examples for the Red Hat Enterprise Linux AI CLI (**ilab**) commands.

1.1. RED HAT ENTERPRISE LINUX AI CLI COMMANDS

1.1.1. **ilab config**

Command Group for Interacting with the configuration of InstructLab

Example usage

```
# Prints the usable commands in the config group
ilab config
```

1.1.1.1. **ilab config init**

Initializes environment for InstructLab

Example usage

```
# Set up the InstructLab environment
ilab config init
```

1.1.1.2. **ilab config show**

Displays current state of the config file stored at `~/.config/instructlab/config.yaml`

Example usage

```
# Shows the `config.yaml` file on your system
ilab config show
```

1.1.1.3. **ilab config edit**

Allows you to edit the config stored at `~/.config/config.yaml`

Example usage

```
# Opens a vim shell where you can edit your config file
ilab config edit
```

1.1.2. **ilab data**

Command Group for Interacting with the data generated by InstructLab

Example usage

```
# Prints the usable commands in the data group
ilab data
```

1.1.2.1. ilab data generate

Runs the synthetic data generation (SDG) process for InstructLab

Example usage

```
# Runs the SDG process on the default model, the default model is specified in the
`~/config/config.yaml`
ilab data generate

# Runs the SDG process on a selected model
ilab data generate --model <model-name>

# Runs the SDG process on the customized taxonomy path
ilab data generate --taxonomy-path <path-to-taxonomy>

# Edits the `config.yaml` to use a specified number of GPUs in SDG
ilab data generate --gpus <num-gpus>
```

1.1.2.2. ilab data list

Displays every dataset in the datasets directory, `~/local/instructlab/datasets`, on your machine

Example usage

```
# List every dataset in the datasets directory
ilab data list
```

1.1.3. ilab model

Command Group for Interacting with the models in InstructLab

Example usage

```
# Prints the usable commands in the model group
ilab model
```

1.1.3.1. ilab model chat

Run a chat using the modified model

Example usage

```
# Creates a virtual environment to chat with the model
ilab model chat

# Creates a virtual environment to chat with a specified model
ilab model chat --model <model-name>
```


1.1.3.2. ilab model download

Downloads the model(s)

Example usage

```
# Downloads the default models
ilab model download

# Downloads the models from a specific repository
ilab model download --repository <name-of-repository>
```

1.1.3.3. ilab model evaluate

Runs the evaluation process on the model

Example usage

```
# Runs the evaluation process on the MMLU benchmark
ilab model evaluate --benchmark mmlu

# Runs the evaluation process on the MT_BENCH benchmark
ilab model evaluate --benchmark mt_bench

# Runs the evaluation process on the MMLU_BRANCH benchmark
ilab model evaluate --benchmark mmlu_branch

# Runs the evaluation process on the MT_BENCH_BRANCH benchmark
ilab model evaluate --benchmark mt_bench_branch
```

1.1.3.4. ilab model list

Lists all the models installed on your system

Example usage

```
* List all the installed models
ilab model list
```

1.1.3.5. ilab model train

Runs the training process on the model

Example usage

```
# Runs the training process on the default model from the config.yaml
ilab model train

# Runs the training process on a specified model
ilab model train --model-name <name-of-model>
```

1.1.3.6. ilab model serve

Serves the model on an endpoint

Example usage

```
# Serves the default model to the server
ilab model serve

# Serves the specified model to the server
ilab model serve --model-path <path-to-model>

# Serves the default model using a specified number of GPUs
ilab model serve --gpus <num-gpus>
```

1.1.4. ilab system

Command group for all system-related commands

Example usage

```
# Prints the usable commands in the system group
ilab system
```

1.1.4.1. ilab system info

Displays the hardware specifications of your system

Example usage

```
#Prints the hardware specifications of your machine
ilab system info
```

1.1.5. ilab taxonomy

Command Group for Interacting with the taxonomy path of InstructLab

Example usage

```
# Prints the usable commands in the taxonomy group
ilab taxonomy
```

1.1.5.1. ilab taxonomy diff

Lists taxonomy files that you changed and verifies that the taxonomy is valid

Example usage

```
# Prints the taxonomy files you changed and verifies that the taxonomy is valid
ilab taxonomy diff

# Prints the taxonomy files in a specified path and verifies that the taxonomy is valid
ilab taxonomy diff --taxonomy-path <path-to-taxonomy>
```

