Actions

At the core of Giza Actions are Actions themselves. An action serves as a framework for coding ML inferencing workflow logic, enabling users to tailor the behaviour of their workflows. Defined as Python functions, any Python function has the potential to be transformed into an action.

Overview of Actions

Think of actions as unique types of functions. They are capable of receiving inputs, executing tasks, and producing outputs. Remarkably, transforming any standard function into a Giza Actions action is as simple as appending the@action decorator. This transformation alters the function's characteristics, bestowing several advantages:

- Each time this function is utilized, its activity is monitored, with every state change communicated to the API for efficient tracking of the action's execution.
- Input parameters undergo automatic type verification and adaptation to ensure they match the required types.
- In cases of failure, the system is equipped to retry. Time constraints can be imposed to avert excessively prolonged workflows.
- Actions leverage inherent logging capabilities, which record essential details of each action run, including its duration and conclusive state.
-

These actions includetasks , can be formulated within modules and then imported to functions in your specific action setups.

Deployments elevate individual workflows from functions that you call manually to API-managed entities.

All tasks must be called from within an action. Tasks may not be called from other tasks.

Action Runs

An action run signifies a singular instance of executing the action.

Initiating an action execution can be done manually, such as by executing a Python script or bringing the action into an interactive session for activation.

Additionally, action runs can be instigated through:

- Utilizing scheduling tools likecron
- to trigger an action function or interval executions.
- Setting up a deployment on Giza Platform.
-

Regardless of the execution method, the Giza API keeps an eye on the action execution, recording its state for easy tracking and visibility.

When an action that includes tasks or other actions is run, Giza will monitor and document how each subordinate execution is related to the main action execution.

Last updated1 day ago