pytorch/.github

> NOTE: This README contains information for the `.github` directory but cannot be located there because it will overwrite the repo README.

This directory contains workflows and scripts to support our CI infrastructure that runs on Github Actions.

Workflows

- Pull CI (`pull.yml`) is run on PRs and on master.
- Trunk CI (`trunk.yml`) is run on trunk to validate incoming commits. Trunk jobs are usually more expensive to run so we do not run them on PRs unless specified.
- Scheduled CI (`periodic.yml`) is a subset of trunk CI that is run every few hours on master.
- Binary CI is run to package binaries for distribution for all platforms.

Templates

Templates written in Jinja are located in the `.github/templates` directory and used to generate workflow files for binary jobs found in the `.github/workflows/` directory. These are also a couple of utility templates used to discern common utilities that can be used amongst different templates.

(Re)Generating workflow files

You will need `jinja2` in order to regenerate the workflow files which can be installed using:

```
pip install -r .github/requirements.txt
```

Workflows can be generated / regenerated using the following command:

```
.github/regenerate.sh
```

Adding a new generated binary workflow

New generated binary workflows can be added in the `.github/scripts/generate_ci_workflows.py` script. You can reference examples from that script in order to add the workflow to the stream that is relevant to what you particularly care about.

Different parameters can be used to acheive different goals, i.e. running jobs on a cron, running only on trunk, etc.

ciflow (trunk) The label `ciflow/trunk` can be used to run `trunk` only workflows. This is especially useful if trying to re-land a PR that was reverted for failing a `non-default` workflow.

Infra

Currently most of our self hosted runners are hosted on AWS, for a comprehensive list of available runner types you can reference `.github/scale-config.yml`.

Exceptions to AWS for self hosted: * ROCM runners

Adding new runner types

New runner types can be added by committing changes to `.github/scale-config.yml`. Example: https://github.com/pytorch/pytorch/pull/70474

> NOTE: New runner types can only be used once the changes to `.github/scale-config.yml` have made their way into the default branch