# Contributing to mmdetection

All kinds of contributions are welcome, including but not limited to the following.

- Fixes (typo, bugs)

- New features and components

## Workflow

1. fork and pull the latest mmdetection

2. checkout a new branch (do not use master branch for PRs)

3. commit your changes

4. create a PR

Note

- If you plan to add some new features that involve large changes, it is encouraged to open an issue for discussion first.

- If you are the author of some papers and would like to include your method to mmdetection,

please contact Kai Chen (chenkaidev[at]gmail[dot]com). We will much appreciate your contribution.

## Code style

### Python

We adopt [PEP8](https://www.python.org/dev/peps/pep-0008/) as the preferred code style.

We use the following tools for linting and formatting:

- [flake8](http://flake8.pycqa.org/en/latest/): linter

- [yapf](https://github.com/google/yapf): formatter

- [isort](https://github.com/timothycrosley/isort): sort imports

Style configurations of yapf and isort can be found in [setup.cfg](../setup.cfg).

We use [pre-commit hook](https://pre-commit.com/) that checks and formats for `flake8`, `yapf`, `isort`, `trailing whitespaces`,

fixes `end-of-files`, sorts `requirments.txt` automatically on every commit.

The config for a pre-commit hook is stored in [.pre-commit-config](../.pre-commit-config.yaml).

After you clone the repository, you will need to install initialize pre-commit hook.

```

pip install -U pre-commit

```

From the repository folder

```

pre-commit install

```

After this on every commit check code linters and formatter will be enforced.

>Before you create a PR, make sure that your code lints and is formatted by yapf.

### C++ and CUDA

We follow the [Google C++ Style Guide](https://google.github.io/styleguide/cppguide.html).