Contribution Guidelines

We strongly encourage our users to create tutorials for their open-source ML projects using the Giza Platform and its components. When your PR for a new tutorial in Giza Hub gets merged, you become eligible for OnlyDust grants!

For consistency and clarity, we ask all user tutorials to follow certain criteria and guidelines, which will be described below. Tutorials such as theMNIST Model are great examples of content and style.

Giza Tutorials

Giza tutorials are meant to be complete, open-source projects built by the Giza Data Science team and Giza Users, to showcase the use of the Giza Framework in an end-to-end setting, as well as to illustrate potential ZKML use cases and inspire new ideas. Given the relative immaturity of the ZKML ecosystem, we believe there are many ZKML use cases yet to be discovered and we welcome everyone to be a part of the first explorers!

Giza Tutorial Project Requirements

General

- Not all good ML use cases are good ZKML use cases.
- Given the additional complexity and costs, verifiability of the model inferences must be a requirement or must provide a significant advantage for the given task. As Giza, we expect all submitted tutorials to have clear ZKML use cases.
- Tutorials are created within Python Virtual Environments and are reproducible, using arequirements.txt
- ,poetry.lock
- in the root dir.
- (If the project uses data that is not available through Giza Datasets) : Project contains data collection scripts to load/fetch the data inside the project.Do not push data files in Giza Hub Repo.
- Similarly,do not push any model files. Model files include .pt, .onnx, or verifiable models (Cairo, EZKL, etc.).
-

Model Development

- It is encouraged to use theactions decorator
- within the model development process.
- Given some of the current restrictions in model complexity, it is expected to have low performance for some of the ZKML models. However, we expect all projects to have a fitting set of performance metrics given the task, a section as part of the documentation on how to interpret them, as well as steps that can potentially be taken from a model architecture point of view to improve the model performance.
- We expect the model development to be well documented and follow the general data science conventions and order of operations.
-

Readme

The project repository must contain a readme file with the following sections:

We strongly encourage the tutorial authors to use code snippets for the Project Installation, Overview of Model Development, Model Performance, and Giza Integration Sections. Showing individual commands and lines of code is always more informative than instructions. * Problem Setting - Introduce the task and the problem setting. Why is ZKML useful/necessary? Are there any papers/resources that can be read to learn more about the problem? * Project Installation - How can another developer reproduce the project, install dependencies etc. * Overview of Model Development - The crucial steps of the model development process. Not every step is necessary, but model architecture and/or model training scheme probably are. * Model Performance - The description of the model performance metric, as well as the measurements from the developed model from the testing. (Here is a good point to also discuss possible improvements to the model architecture.) * Giza Integration - How can we use Giza CLI & Actions to make the model verifiable? What are the individual Giza components that are used along the way? * Affiliations - If affiliated with any team/organization, make sure to have that it be stated in the readme for clarity. *

How to contribute?

After making sure your tutorials follow the guidelines above, simply create a new branch inGiza-Hub Github repository, and push your model underawesome-giza-actions folder. After pushing your tutorial, make sure to hop on ouDiscord and tell us about your project! If your tutorial is original, reproducible, fits the guidelines and is a valuable addition to our zkML use-case collection, you will be eligible forOnlyDust rewards!

Previous Quickstart Next Actions

Last updated1 day ago