Faculty Mentor-

Dr. S Anupama Kumar,
Associate Professor,
Dept of AIML, RVCE

Presented by-

Kushagra Aatre 1RV22AI025
Shreya M 1RV22AI054
Aryan Sinha 1RV22AI009

WHAT IS CLEARML?

Definition: ClearML is an open-source Machine Learning Operations (MLOps) platform designed to streamline the entire machine learning lifecycle, including experimentation, orchestration, and deployment.

Core Objective: Provide a unified, extensible platform for managing and optimizing ML workflows.

Target Audience: Designed for data scientists, ML engineers, DevOps teams, and organizations focusing on scaling and operationalizing ML workflows.

Technology Stack: Built with Python, compatible with Docker, Kubernetes, and cloud-native ecosystems.

WHO CAN USE CLEARML?

Data Scientists: To track experiments and compare results.

ML Engineers: For automating training pipelines and deploying models.

DevOps Teams: To integrate ML workflows into production systems.

Beginner ML Enthusiasts: To stay organized and avoid common pitfalls in ML projects.

CLEARML IS USED BY

COMPATIBILITY

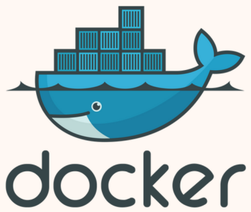
Operating Systems: Fully supports Windows, macOS, and Linux environments.



Programming Languages: ClearML SDK is Python-based but can integrate with scripts in other languages through APIs.

Integration with DevOps Tools

- Docker for containerization.
- Kubernetes for scalable deployments.
- Jenkins, GitHub Actions, and other CI/CD tools.




Cloud Platforms: AWS, GCP, Azure, and on-premise clusters (via SSH or custom setups).

FREE VS ENTERPRISE VERSION

Feature	Free Version	Enterprise Version
Experiment Tracking	✅ Full support	✅ Full support
Pipeline Orchestration	✅ Basic	✅ Advanced (custom pipelines)
Dataset Management	✅ Basic	✅ Advanced with lineage tools
Model Deployment	✅ Basic	✅ Advanced (scalable, secure)
Collaboration	✅ Limited (basic sharing)	✅ Role-based, private workspaces
Security	❌ Basic	✅ SSO, MFA, GDPR compliance
Scalability	❌ Limited by hosted server	✅ Optimized for large teams
Support	❌ Community (forums, GitHub)	✅ SLA-backed, dedicated support
Custom Storage	❌ Limited (default storage)	✅ Full customization options
Cost	Free	Custom (based on organization)

SOFTWARE REQUIREMENTS

1. Programming Language and Framework
   - Python: Version 3.9 or higher.
   - Streamlit: Version 1.26.0 or higher.

2. Key Libraries
   - scikit-learn: Version 1.3.0 or higher.
   - xgboost: Version 1.7.6 or higher.
   - pandas: Version 1.5.3 or higher.

3. Development Environment
   - Visual Studio Code: Version 1.83.0 or higher.
   - Jupyter Notebook: version 6.5.4 or higher

4. Optional Tools
   - Git: Version 2.41.0 or higher

5. System Requirements
   - Operating System: Windows, macOS, or Linux.
   - RAM: Minimum 4 GB (recommended 8 GB for larger datasets).
   - Storage: At least 250 MB free disk space.



HOW TO USE CLEAR ML?

1. Creating an Account on the ClearML Hosted Service

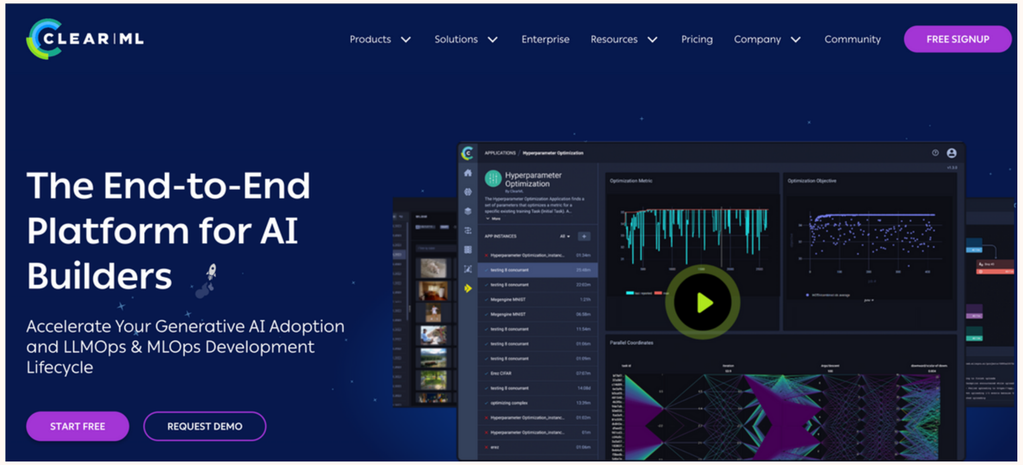
The hosted ClearML server is free to use and simplifies the setup process.

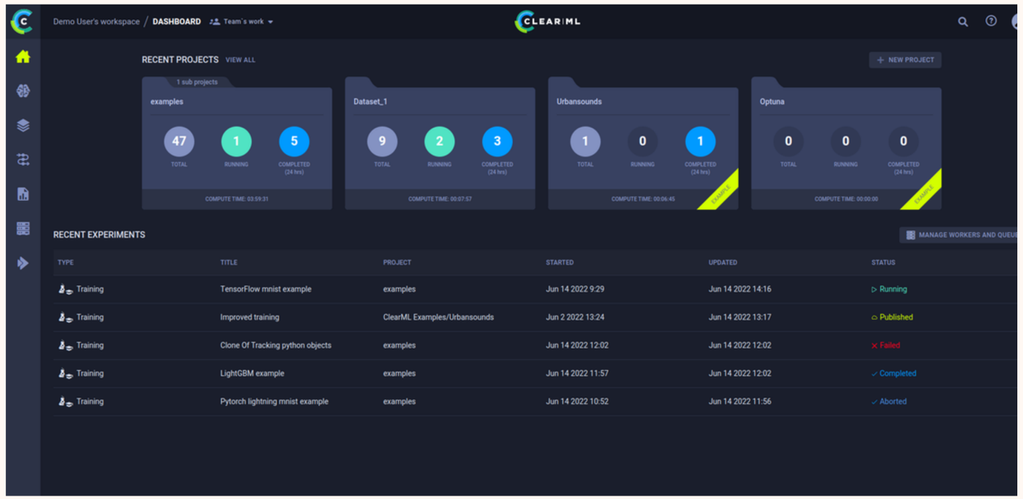
Step 1: Sign Up for ClearML

1. Go to the ClearML website and click on “FREE SIGNUP” (top-right corner).
2. Register using one of the following options:
   - Your email and password.
   - Your Google account (OAuth login).

Step 2: Access the ClearML Dashboard

1. Once signed up, log in to your account.
2. You’ll be redirected to the ClearML dashboard, which includes sections for:
   - Experiments
   - Pipelines
   - Datasets
   - Models
   - Task Queues
   - Access Key



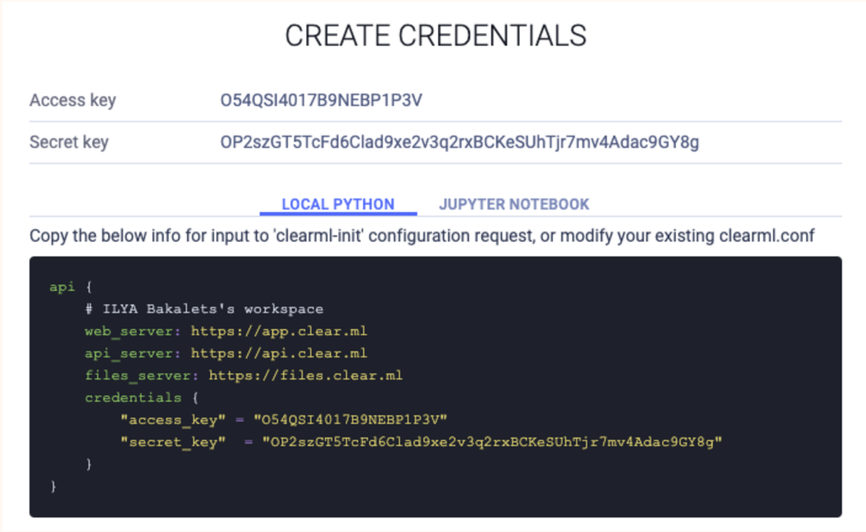


HOW TO USE CLEAR ML?

Step 3: Retrieve Your API Credentials

1. Go to Settings (top-right corner).
2. Navigate to API Keys under your profile settings.
3. Copy the following:
   - Access Key
   - Secret Key

These credentials will be used to link your local ClearML client to the hosted server.



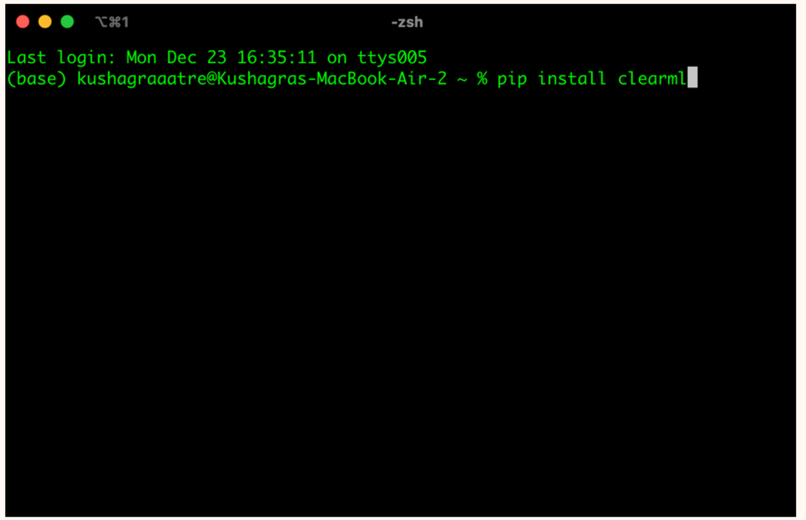
2. Setting Up ClearML Locally

This involves installing the ClearML Python SDK and configuring it to connect to the ClearML server (hosted or self-hosted).

Step 1: Install ClearML SDK

Make sure Python (3.6 or higher) is installed on your system. Then, run:

pip install clearml



HOW TO USE CLEAR ML?

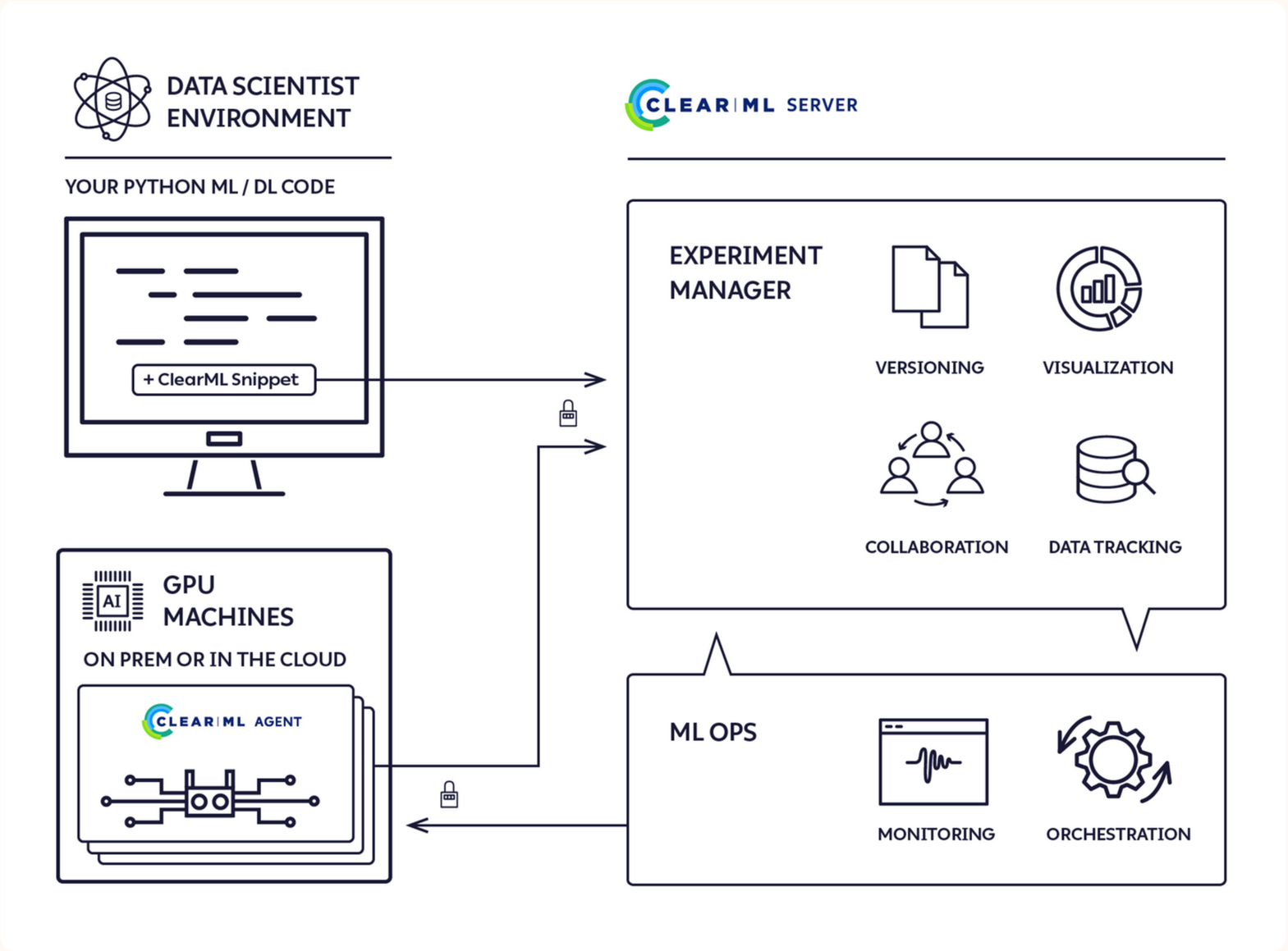
Step 2: Initialize ClearML
Run the following command in your terminal:

clearml-init

Step 3: Provide Configuration Details
During initialization, ClearML will prompt you to input the following:

1. API Server URL:
   - For the free hosted server:
   - arduino
   - Copy code
   - https://app.clear.ml
   - For a self-hosted server: Use your server's URL (e.g., http://<your-server-ip>:8080).
2. Access Key and Secret Key: Paste the credentials obtained earlier from the hosted ClearML platform or your self-hosted server.

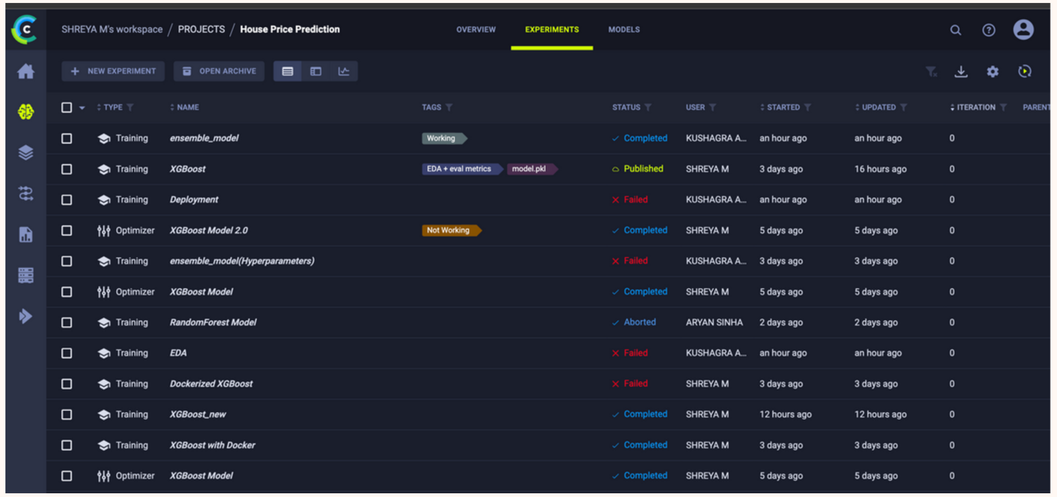
KEY FEATURES OF CLEARML



KEY FEATURES OF CLEARML

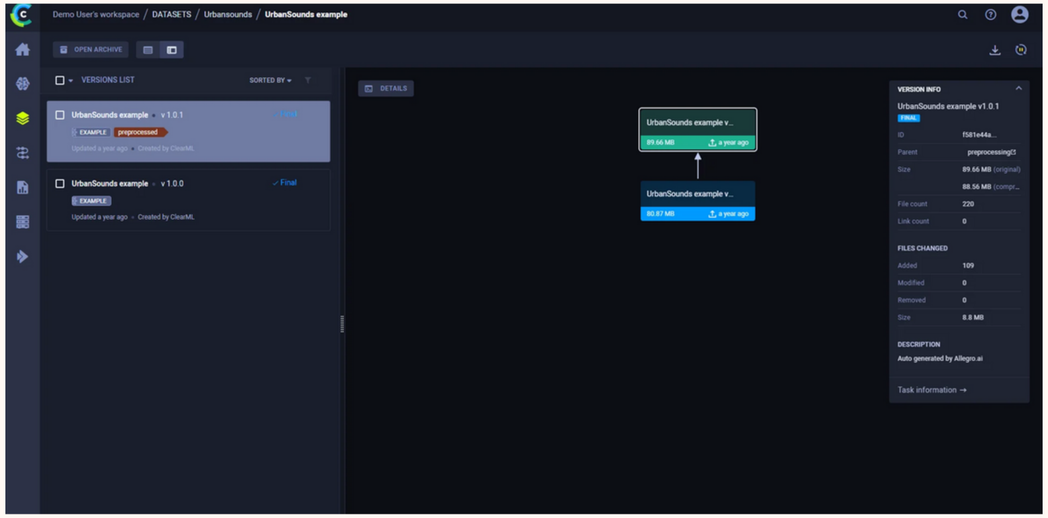
Experiment Tracking

- Logs all experiment metadata:
  - Code version (Git hash).
  - Parameters (hyperparameters, configurations).
  - Metrics (loss, accuracy, etc.).
  - Outputs (model checkpoints, logs).
- Real-time dashboards for interactive tracking and comparison of multiple experiments.
- Benefit: Ensures reproducibility and transparency.



Dataset Versioning

- Manage datasets as immutable entities with metadata and storage references.
- ClearML Data API allows:
  - Dataset creation, upload, and versioning.
  - Dataset lineage tracking (which model used which dataset).
- Benefit: Ensures consistency and simplifies dataset sharing.



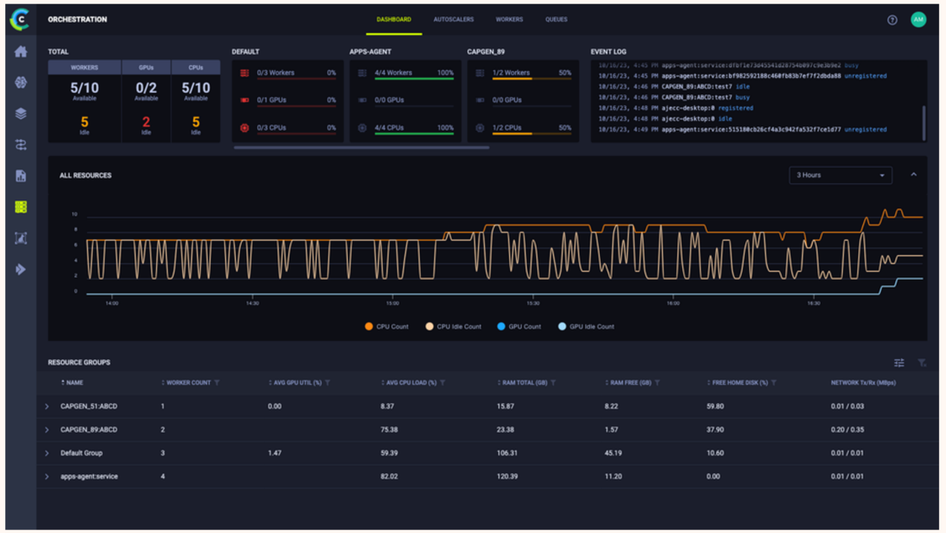
KEY FEATURES OF CLEARML

Team Collaboration

- Multi-user environment with centralized dashboards.
- Share experiments, metrics, and datasets with teammates effortlessly.
- Benefit: Promotes transparency and teamwork.

Task Orchestration

- Automate ML workflows using task dependencies.
- Schedule and execute tasks with ClearML Agents on local or remote machines.
- Supports hyperparameter optimization via grid search or random search.
- Benefit: Reduces manual intervention and allows for scalable execution.



KEY FEATURES OF CLEARML

Model Deployment and Monitoring

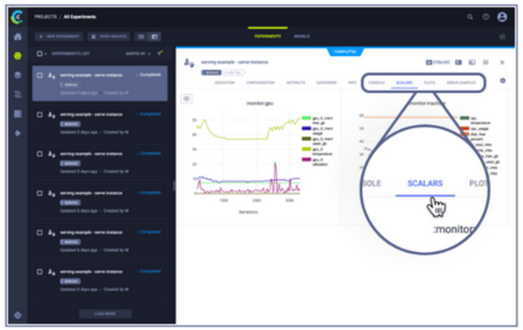
- Simple model registration and deployment pipelines.
- Real-time monitoring of model performance in production environments.
- Integrates with Docker and Kubernetes for scalable deployments.
- Benefit: Enables continuous monitoring and improvement of production models.

Remote Execution

- Execute training jobs on remote machines without modifying the codebase.
- ClearML Agent handles environment setup (cloning repo, installing dependencies).
- Supports multiple queues for prioritized execution.
- Benefit: Efficient resource utilization and faster experimentation.

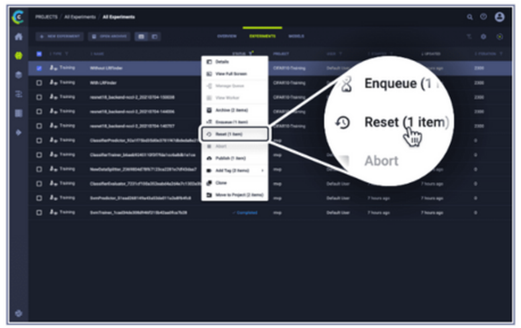
KEY FEATURES OF CLEARML

TRACK EXPERIMENT RESULTS



Go to any experiment results tabs to see its logs, generated artifacts, reported metrics and debug samples

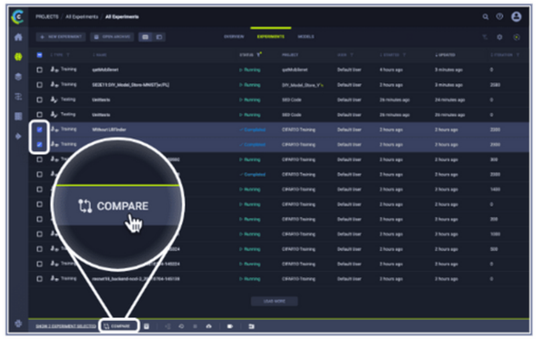
RESET EXPERIMENTS



You can reset any non-published experiment, setting it back to 'draft' status.

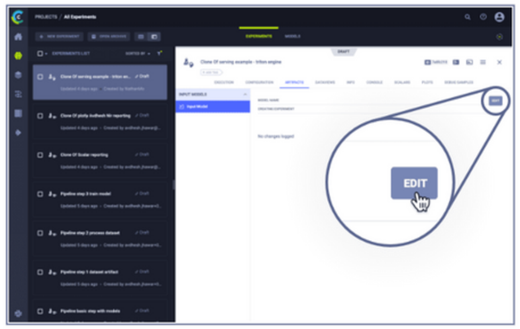
Use this to modify erroneous configurations and rerun failed experiments

COMPARING EXPERIMENTS



Select the experiments you want to compare, then compare them using the bottom bar.

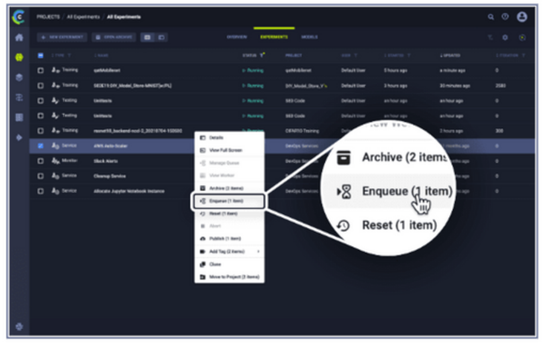
COMPARE MODEL BEHAVIOR



Clone a previously run experiment and use a different input model.

Execute the reconfigured new experiment to see how it performs

EXECUTING EXPERIMENTS REMOTELY



Enqueue draft tasks for execution by a worker using right-click or the task context menu.

EXPERIMENT QUEUES

Use queues to schedule experiment execution and utilize your training machines.

You can create any number of queues to match your queuing policy, then use them to enqueue tasks for execution by workers.

CLEARML IN THE MLOPS WORKFLOW

Reproducibility

ClearML automatically logs all experiment details, ensuring models can be reproduced precisely.

Automation

Pipeline orchestration and remote task execution automate repetitive ML workflows, reducing time-to-deployment.

Scalability

ClearML Agents and Kubernetes support enable scaling experiments and deployments across multiple machines or cloud instances.

CLEAR | ML

Centralized Management

A unified platform for experiment tracking, dataset management, and deployment simplifies collaboration and management.

Integration with CI/CD

Integrates seamlessly into DevOps pipelines for continuous model retraining and deployment.

LIMITATION AND DRAWBACKS

1. Dependency on Python

- Challenge: ClearML is primarily designed for Python-based workflows.
- Impact: Teams working with other programming languages (e.g., R, Java) may find it less useful or face integration issues.
- Mitigation: API extensions or custom integrations may be required for non-Python environments.

2. Free vs. Paid Features

- Challenge: While the free hosted version is powerful, some advanced features (e.g., enterprise-grade support, custom branding, advanced resource management) are available only in the paid version.
- Impact: Teams may find themselves limited in scaling or requiring features only available in premium plans.
- Mitigation: Evaluate your team's needs and budget carefully before committing.

3. Hardware and Resource Requirements for Self-Hosting

- Challenge: Hosting ClearML on-premises requires setting up and maintaining a server with sufficient resources (CPU, memory, storage).
- Impact: Small teams or organizations without DevOps expertise may struggle with deployment and maintenance.
- Mitigation: Use the free hosted ClearML server if privacy and control are not critical concerns.

4. Learning Curve

- Challenge: New users may face a steep learning curve, especially if they are unfamiliar with MLOps concepts like task queues, agents, or pipeline orchestration.
- Impact: It might take time to set up and fully utilize ClearML's features, especially for small teams with limited MLOps expertise.
- Mitigation: Comprehensive tutorials and documentation are available but require dedicated effort to study.

THANK YOU