EclipseCon 2023

Applied AI with Eclipse Graphene

Martin Welß
Project Lead Graphene
Graphene Portal

- Catalog
- Design Studio
- Deployment Services

Portal stores only links to docker images and data

Execution Environments

Public Playground
  Kubernetes

Company IT
  Kubernetes
  GPU

HPC
  Slurm/Docker/Apptainer
  GPU
AI Model Catalog

- Catalog of re-usable AI Modules
- The AI-Modules are Docker Containers
- Work in teams
- Support for commercial and open source licenses for models
Compose AI Pipelines in the Design Studio

- Training pipeline with Tensorboard Integration
- Shared-folder concept
- Show matching models
- Application Pipeline
AI Playground

- Execution Environment
- One-Click-Deployment
- Based on Kubernetes
- Playground-App is a Graphene repo
- One-Click Update: Docker images are re-pulled on Reset
Container Specification for composable AI-Modules

- Docker Container with additional properties
- public interface in protobuf / grpc
- optional Web-UI (highly recommended)
- No Lock-In: it continues to be a normal Docker container
- If done right, the same Docker container can be used across different execution environments (e.g. GPU or no GPU)
Generic Parallel Orchestrator

- The orchestrator executes the pipeline according to the topology file (generated by the design studio)
- The orchestrator dispatches the messages among the nodes following the edges
- Thanks to protobuf/grpc the communication stubs can be generated on the fly
- Support for grpc streaming:
  - Sensor-Input
  - UI-Input
  - Media-Streams
  - The edges can be served in parallel
- Different orchestrator implementations possible
Pipeline Definition (Topology)

- A JSON-file which contains nodes and edges (technology neutral)
- Is generated by the Design Studio, but can be generated elsewhere (e.g. for AutoML)
- During deployment, the specific configurations for an execution environment are generated (MLOps)
- Different orchestrators are possible
- Shared Folder support
Databroker

- A Databroker is a shallow component that makes data available to the pipeline
- Usually the first node of a pipeline
- Specific for an execution environment (e.g. company IT)
- Can access the data source (credentials)
- Does not contain the dataset!
- We have already a prototype of an Eclipse Dataspase Connector
What’s New in Graphene Release 1.1?

- Installation uses Kubernetes 1.26, Calico 3.26, Helm 3.12.3, Ingress-Nginx 4.7.1
- Playground-App is faster
- Support for more protobuf features:
  - enums
  - oneof
  - recursive message definitions
  - nested definitions
- Better support for directed cyclic topologies
- Jupyter-Connect
- First-Steps for Reproducibility
- And Benchmarking (Metrics)

```python
syntax = "proto3";

message Expression {
  // As in s-expression, an Expression is either an atom or list representation.
  repeated Expression list = 2;

  Atom atom = 1;
  // If the 'atom' field is empty, then the expression is a list of
  // typically representing the application of some arguments to a
  // For instance `(+ 1 3)' (at-robot l1)' `(>= (battery_level) 2)

  string type = 3;
  // Kind of the expression, specifying the content of the expression.
  // This is intended to facilitate parsing of the expression.

  ExpressionKind kind = 4;
}
```
Cognitive Architectures

- To fully leverage AI power, we need to combine LLM with other AI technologies:
  - Planning, Logic, Knowledge Graphs, Filter, Image Object Detection, OCR, Audio
  - Constant Learning, Feedback loops
  - Cognitive Architectures require directed cyclic graphs (supported by Graphene 1.1)
  - And recursive data structures (supported by Graphene 1.1)

Maze-Planner
If the user deploys a single model
- Graphene automatically generates a suitable Jupyter-Lab container
- and adds the protobuf interface definition
- and a shared folder
- for comfortable interaction with a model / AI-Module
Capture Metadata about:

- System (CPU/GPU/RAM)
- Datasets
- Docker Image Checksums
- Different Metrics
- See Container-Spec

```
"checksum": "docker-pullable://cicd.ai4eu-dev.eu:7444/tutorials/house_price_prediction/hpp_databroker@sha256:6d8c5f203a4bcc8a794b54d82a7428b36ffeb9f53b164968f576442e2bdde89f",

"dataset_features": {
    "type": "aiod-dataset/v1",
    "datasetname": "The Reuters Dataset",
    "description": "http://kdd.ics.uci.edu/databases/reuters21578/README.txt",
    "size": "4MB",
    "DOI_ID": "Not available"
}

"metrics": [
    {
        "type": "classification-training-metrics/v1",
        "date_time": "2023-09-07 07:28:51",
        "accuracy": 0.9216,
        "validation_loss": 0.9244,
        "status_text": "success"
    },
    {
        "type": "classification-testing-metrics/v1",
        "date_time": "2023-09-07 07:28:51",
        "F1 Score": 0.85,
        "Specificity": 0.90,
        "ROC-AUC": 0.92,
        "status_text": "success"
    }
]```

"running_time": "345s",
"system_info": {
    "system_name": "mwtest",
    "fqdn": "test.playground.org",
    "cpu": "10",
    "gpu": "...",
    "memory": "66851412Ki"
}
Challenges (Planned)

- Challenges are special types of catalogs

<table>
<thead>
<tr>
<th>CATALOG NAME</th>
<th>PUBLISHER NAME</th>
<th>SELF-PUB</th>
<th>ACCESS TYPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>acemos-int-fhg Internal</td>
<td>AI4EU Experiments</td>
<td>No</td>
<td>Restricted</td>
</tr>
<tr>
<td>AI4EU Experiments Public</td>
<td>acemos-int-fhg.ai4eu.eu</td>
<td>No</td>
<td>Public</td>
</tr>
<tr>
<td>Classification Challenge</td>
<td>DEV AI4EU Experiments</td>
<td>No</td>
<td>Public</td>
</tr>
</tbody>
</table>
Leaderboard (Planned)

- Order based on Metrics
LLM Pipelines: Grounding LLM with Knowledge Graphs

Work in progress
The issue aims at automatically generating documents using large language models and LangChain. By doing so, we can effectively streamline the creation of comprehensive and informative documentation.

The initial overview of the process is shown below,

- **Structure/Format of the document**
  - Content headers - Introduction, Databroker Specifications, Metrics Aggregation, gRPC Communication, Usage, etc.

- **LangChain Integration**
  - OpenAI API Selection

- **Data Pre-processing**
  - Concatenate all the input - Existing README docs, Various code snippets, etc.

- **Generate document**
  - 1. Version Control
  - 2. Automatic Updates - Based on the changes in the code
  - 3. User feedback and improvisation

Work in progress
Graphene Use Cases

- General: Building AI-pipelines from re-usable AI-modules
- AI Research: model training, benchmarking, Jupyter-Connect
- Education: practical live demonstrations and student exercises in the playground
- MLOps: simplify deployment, training pipelines, data cleaning pipelines, can support many tools
- Model showcase, catalog (internal, external)
- Application: let non ai-experts compose and deploy pipelines for their business domains
- Collaboration: work in mixed teams on PoCs
- Automated certification
Supporting Projects: Graphene Funding at least until 2028

- KI.NRW
- AI4Europe
- HumanE-AI Net
- West-AI
- Deploy-AI (starting 2024)
Useful Links

- AI4EU Experiments: https://aiexp.ai4europe.eu/#/home
- KI.NRW AI-Lab: https://www.ai-lab.nrw/
- Eclipse Graphene Project: https://projects.eclipse.org/projects/technology.graphene
- Eclipse Graphene Gitlab: https://gitlab.eclipse.org/eclipse/graphene
- Tutorials: https://gitlab.eclipse.org/eclipse/graphene/tutorials
Contact

Martin Welß
Senior AI Solution Architect
martin.welss@iais.fraunhofer.de

Fraunhofer-Institut für intelligente Analyse- und Informationssysteme IAIS
Schloss Birlinghoven 1
53757 Sankt Augustin
Germany

www.iais.fraunhofer.de