Contribution ID: **71** Type: **Talk (15min + 5min)**

MLentory: A Machine Learning model registry with natural language queries

Wednesday 26 February 2025 11:00 (20 minutes)

The rapid increase of Machine Learning (ML) models and the research associated with them has created a need for efficient tools to discover, understand, and utilize these resources. Researchers often need help traversing the large collection of ML repositories and finding models that align with their specific requirements, such as open-source availability, FAIR principles, and performance metrics. MLentory addresses this challenge by providing a comprehensive solution for ML model discovery.

MLentory is a system that extracts, harmonizes, and stores metadata from diverse ML model repositories, including Hugging Face and OpenML. This metadata is harmonized using RDA FAIR4ML schema, stored into FAIR Digital Objects (FDOs), and indexed to enable efficient natural language-based search. By leveraging different information retrieval techniques, MLentory enables researchers to discover, compare, and dive into ML models tailored to their needs.

The core components of MLentory are an ETL pipeline, a backend service, and a frontend interface. The ETL pipeline, implemented using Python scripts, extracts metadata from various sources, transforms it into a standardized format, and loads it into a PostgreSQL database for historical tracking, a Virtuoso database for RDF-based knowledge representation, and an Elasticsearch module for efficient data indexing. Each stage of the pipeline operates independently within its own container.

Then there is the backend module, built with FastAPI, which serves as the query engine, enabling users and other systems to retrieve information from the different data stores in MLentory. The natural language-based search leverages Elasticsearch for initial retrieval and then employs a self-hosted LLM powered by Ollama to refine search results through Retrieval Augmented Generation (RAG).

Finally, the frontend module, developed using Vue3.js, provides a user-friendly interface that allows users to explore models using natural language and different search filters, and then delve into their version history. MLentory maintains a history of metadata changes for each model, allowing users to track their evolution and identify versions with the right compatibility for their needs.

One of the main features of MLentory is its highly decoupled architecture, where each component runs in its own Docker container, and Apache Kafka is used as a common framework for asynchronous communication between containers. Apache Kafka is built on the idea of having queues where publishers can write messages and consumers can read them. This modular design facilitates independent scaling, flexible technology choices, and isolated error handling.

The downside of a decoupled architecture is that maintenance and code standards become more difficult to enforce. Then, to ensure the quality and reliability of the system, a testing framework was implemented. It encompasses unit, integration, and coverage tests. Additionally, linting checks are employed to maintain code style and consistency. This framework is automated through a continuous integration (CI) pipeline deployed on CircleCI, guaranteeing that tests are executed after every code commit.

I want to participate in the youngRSE prize

yes

Primary author: QUINONES VIRGEN, Nelson David (ZB MED)

Co-authors: CASTRO, Leyla Jael (ZB MED Information Centre for Life Sciences); REBHOLZ-SCHUHMANN, Dietrich (ZB MED - Information Centre for Life Sciences, Cologne, Germany)

Presenter: QUINONES VIRGEN, Nelson David (ZB MED)

Session Classification: Large Language Models(LLMs) in RSE

Track Classification: Research Software: AI and ML in a research context