Approaches toward a Unified Framework for Data Management

Data-driven methods are becoming increasingly prevalent, while the data itself is becoming more dynamic. There is a clear need to move away from adhoc workflows toward a more scalable solution that integrates well with existing development processes. This poster aims to illustrate the current research effort toward such a framework.

Metadata and Provenance

Collaborative, adhoc data science workflows pose a challenge for traditional lifecycle management systems. Version control systems are based on a low-level file abstraction and lack capabilities to reason about data contained within versions and relationships between datasets.

ProvDB proposes a unified system to track version lineages of generated datasets, scripts and models to enable introspection and debugging of the workflows.

User challenge

Define a schema for the provenance information a priori

Capture metadata in a passive manner with minimal effort

Generate useful insights from captured metadata

Proposed solution

A small base schema is fixed, but users can add arbitrary semi-structured metadata (“schema-later”)

Suite of provenance ingesters for popular frameworks (scikit-learn, caffe), default ingester for UNIX shell commands

Logical data model is mapped into a Neo4J property graph to enable Cypher queries and visual exploration

Continuous Data Cleaning

Data-driven businesses need to ensure data quality not only for static data with fixed constraints, but also in dynamic environments where data changes frequently and constraints evolve over time. Inconsistencies can arise from incorrect data values or stale constraints.

The proposed system is able to operate on data streams by considering incremental changes instead of starting the repair process from scratch. It automatically suggests repairs and involves a human-in-the-loop to validate application semantics.

TFX: toward an end-to-end platform

TensorFlow Extended (TFX) is a machine learning platform implemented at Google. Driven by the need to keep track of experiment history in a centralized database and be resilient against disruptions from inconsistent data, TFX integrates aforementioned approaches into an end-to-end platform with shared configuration.

It allows continuous training and validation over evolving data, captures data transformation pipelines and exposes a simple interface for users of various levels of expertise to monitor and debug their workflows.

Conclusion

While TFX is built on top of TensorFlow for the machine learning use case, it provides a solid reference architecture for a general-purpose data management platform, including traditional ETL processes.

An open standard for metadata models and stricter formalization of data workflow activities through clear interfaces would not only enable researchers to contribute their work directly (e.g. classifiers for data cleaning), but also create opportunities for sharing metadata and generating new insights.

Collectively, this could be a step away from adhoc glue code, custom scripts and fragile systems with high technical debt toward more resilient, transparent and reproducible processes.

References