Experiencing ProvLake to Manage the Data Lineage of AI Workflows

Leonardo Guerreiro Azevedo, Renan Souza,
Raphael Melo Thiago, Elton Soares, Marcio Moreno

[1]IBM Research
Av. Pasteur, 146 – 22.290-240 – Rio de Janeiro – RJ – Brazil

{lga, rfsouza, raphaelt, mmoreno}@br.ibm.com, eltons@ibm.com

Abstract. *Machine Learning (ML) is a core concept behind Artificial Intelligence systems, which work driven by data and generate ML models. These models are used for decision making, and it is crucial to trust their outputs by, e.g., understanding the process that derives them. One way to explain the derivation of ML models is by tracking the whole ML lifecycle, generating its data lineage, which may be accomplished by provenance data management techniques. In this work, we present the use of* `ProvLake` *tool for ML provenance data management in the ML lifecycle for Well Top Picking, an essential process in Oil and Gas exploration. We show how* `ProvLake` *supported the validation of ML models, the understanding of whether the ML models generalize respecting the domain characteristics, and their derivation.*

1. Introduction

Machine Learning (ML) is a core concept behind Artificial Intelligence systems being applied to several domains, like finance, health, and climate [Gil et al. 2018, Rodrigues et al. 2018]. ML systems work driven by data and generate ML models. These models are derived by processing input datasets and are used to make forecasts and guide decisions. The trust in the outputs of these ML models is crucial, and understanding the derivation process is an effective way to accomplished this.

This work presents the use of `ProvLake`[1] tool [Souza et al. 2019b] for understanding how ML models were derived utilizing provenance data management, *i.e.*, capture, storage, and querying. By using this tool, we aim to answer the question: "How to allow the understanding of data transformations of ML lifecycle – from the raw data to the trained models?", underpinning the validation of the trained models and understanding how they generalize respecting domain characteristics.

Historically, provenance data management has been used to capture, represent, store, and query data lineage in data-driven workflows, like the ML lifecycle [Herschel et al. 2017]. While capturing the data being processed in the lifecycle, `ProvLake` logically integrates and ingests them into a provenance database, named `ProvLake Data View` (`PLView`), ready for analyses at runtime [Souza et al. 2019b]. The tool captures provenance of the three phases of ML lifecycle: data curation, data preparation for learning, and learning [Souza et al. 2019a]. It then gives an integrated view of domain data, execution data, and ML data in multiworkflows supporting queries and analysis on such data.

[1]`http://ibm.biz/provlake`

2. Machine Learning Lifecycle

We can divide the ML lifecycle into three major phases [Souza et al. 2019b]: *data curation*, *data preparation for learning*, and *learning* (Fig. 1).

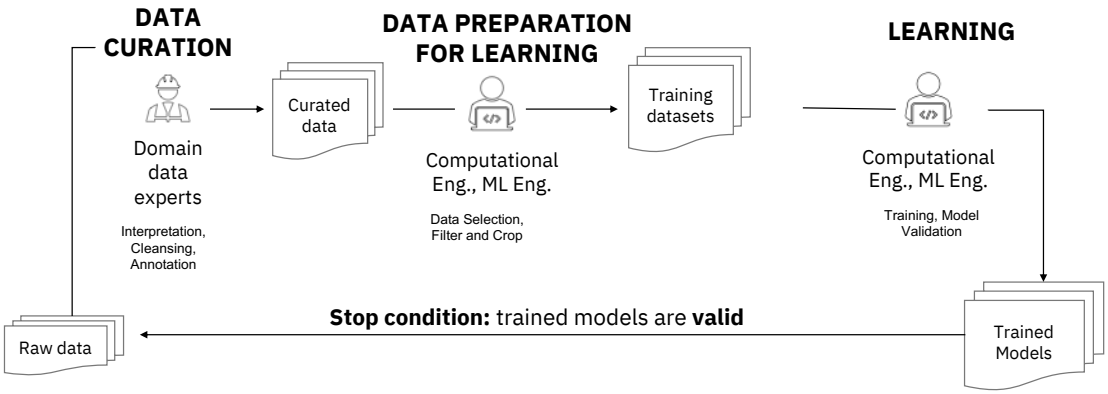

Figure 1. The ML lifecycle [Souza et al. 2019b].

(i) **Data Curation** transforms raw data into useful data for ML processing. In some scenarios, datasets may be massive (*e.g.*, terabytes) and can have geospatial-temporal data, stored in different data formats (*e.g.*, HDF5) or may be domain-specific (*e.g.*, SEG-Y for seismic data, in the case of the Oil and Gas (O&G) industry). In such cases, data processing may require industry-specific software and domain knowledge to inspect, visualize, and understanding.

(ii) **Learning data preparation** selects relevant parts of the curated data for learning. After selecting the data, model designers develop scripts that transform the data into training datasets. Typical transformations include image crop, quantization, scale, among others.

(iii) **Learning** selects the input training datasets, optionally choosing validation datasets and training parameters, and execution of the training process. A training process often generates multiple trained models, and one is chosen as the "best" depending on evaluation metrics (*e.g.*, accuracy, or any other user-defined metric).

3. `ProvLake` Tool

`ProvLake` [Souza et al. 2019b] allows the tracking of data lineage of ML lifecycle. It stores tracked data in a provenance database, supporting an integrated view and making easier the decision-making processes that analyze data of computational processes.

`ProvLake` is a system for runtime analysis of multiworkflow data. It logically integrates and ingests multiworkflow data into a provenance database, named `ProvLake Data View` (`PLView`), ready for data analyses at runtime. During a multiworkflow execution, instrumented code captures and ingests data in the PLView, including: domain data extracted from data stored in multiple stores; explicit data relationships datasets distributed across the multiple stores; and, the multiworkflow data relationships. The PLView is materialized in a DBMS.

`ProvLake` provides a lightweight data tracker API to be added to workflow codes, like scripts. Also, a query API supports runtime analytical queries that integrate

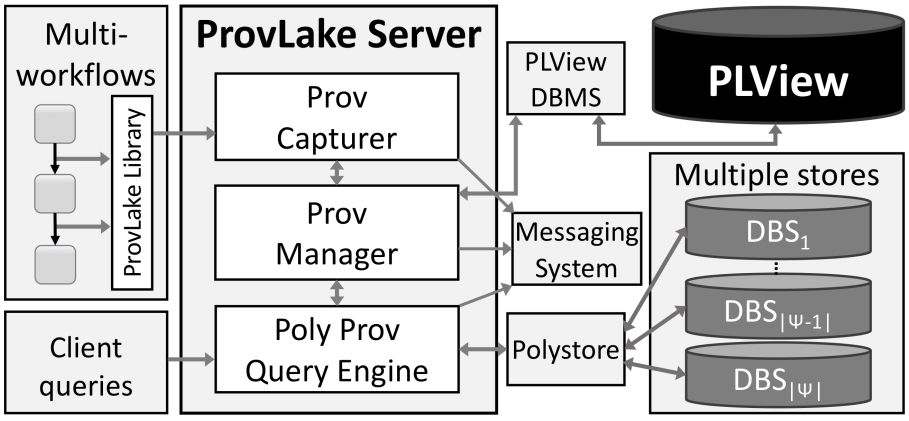

Figure 2. `ProvLake`'s Architecture [Souza et al. 2019b]

.

multiple stores' data at runtime. Those stores may be homogeneous or heterogeneous, considering storage and query technology. The query API supports direct access to the multiple stores jointly with their provenance data. `ProvLake`'s provenance data model follows W3C PROV[2] standards for provenance data representation, and extends PROV-DM[3].

The `ProvLake-Server` is the main component, and it has three subcomponents: `ProvCapturer`, `ProvManager`, and `PolyProvQueryEngine`.

To populate the `PLView`, the team responsible for the workflow instrument workflow code using `ProvLake Lib`, which, during workflow execution, captures data and sends to `ProvCapturer`, which transforms the data into provenance data following ProvLake's data representation. Then, the `ProvCapturer` sends provenance data to `ProvManager`, which inserts data into the DBMS managing the `PLView`.

Clients send API query requests to `PolyProvQueryEngine`, which connects to `ProvDataManager` to query the `PLView` and employs a polystore to query data directly in the multiple stores, jointly with their provenance data.

The tool is deployed in IBM Cloud and has been used in internal projects of the IBM Research Brazil lab and projects for clients. *E.g.*, a joint project between IBM and Galp to build an Artificial Intelligence advisor to speed up the appraisal of oil and gas prospects and discoveries[4].

4. Use case

`ProvLake` was used for understanding the training process for Well Top Picking, which is an essential process in the O&G exploration process. The use case included a geoscientist who knows the basics of ML and has a dataset of 500 well logs. Picking tops in each one of the wells is time-consuming. However, considering an O&G AI Workbench with ML models for well top picking, these models can be applied to the well logs to pick the well tops for the dataset.

[2]`https://www.w3.org/TR/prov-overview/`

[3]`https://www.w3.org/TR/2013/REC-prov-dm-20130430/`

[4]https://www.galp.com/corp/en/media/press-releases/press-release/id/834/galp-and-ibm-build-cutting-edge-artificial-intelligence-advisor-to-speed-up-appraisal-of-oil-and-gas-prospects-and-discoveries

In the use case, the Geoscientist starts visualizing some properties of a well log file, and, using ProvLake, s/he retrieves the best ML model for Well Top picking from the O&G AI Workbench platform. S/he applies the model and visualizes ProvLake did not pick the tops correctly. Hence, s/he performs queries, using ProvLake, to understand the reasons for the errors through evaluating hypotheses.

The first hypothesis was checking the distance between the query well log and the well logs used to train the model. S/he could get these well logs because ProvLake stored the wells used in the data curation phase. Through plotting these wells, s/he discovered wells are near and rejected this hypothesis.

The second hypothesis was checking if the query well log and the well logs belong to the same basing. S/he got the basins by using the provenance data and plotting them. The result was the well logs are in different basins, and the chosen model did not fit the scenario. After retrieving the best model in the same basin of the query well log, and applying it, s/he got the right picks. As a result, s/he can apply this approach to the whole dataset.

5. Conclusion

This work presented the use of ProvLake tool in ML. ML is a core concept behind Artificial Intelligence used in many areas in the industry and academia. It essential to have trust in the results produced by the application of ML models, which ProvLake helps to reach through provenance capture and consumption (*e.g.*, query executions). We applied the tool in an ML use case in O&G industry to demonstrate the benefits of ProvLake.

Considering performance, we validated ProvLake in another ML scenario, also in O&G industry, evaluating the performance with 48 GPUs in parallel, and the data capture overhead was less than 1% [Souza et al. 2019a].

More details about ProvLake's components and the structure of the data stored in the system is presented in [Souza et al. 2019b].

References

Gil, Y., Pierce, S. A., Babaie, H., and Banerjee, A. *et al*. (2018). Intelligent systems for geosciences: an essential research agenda. *Comm. of the ACM*.

Herschel, M., Diestelkämper, R., and Ben Lahmar, H. (2017). A survey on provenance: What for? what form? what from? *VLDB Journal*.

Rodrigues, E., Oliveira, I., Cunha, R., and Netto, M. (2018). DeepDownscale: a deep learning strategy for high-resolution weather forecast. In *IEEE Int. Conf. on eScience*.

Souza, R., Azevedo, L., Lourenço, V., Soares, E., Thiago, R., Brandão, R., Civitarese, D., Brazil, E., Moreno, M., Valduriez, P., Mattoso, M., and Netto, M. (2019a). Provenance data in the machine learning lifecycle in computational science and engineering. In *IEEE/ACM WORKS@Supercomputing*, pages 1–10.

Souza, R., Azevedo, L., Thiago, R., Soares, E., Nery, M., Netto, M., Brazil, E. V., Cerqueira, R., Valduriez, P., and Mattoso, M. (2019b). Efficient runtime capture of multiworkflow data using provenance. In *IEEE Int. Conf. on eScience*, pages 1–10.