Institut de Recherche
en Informatique de Toulouse
CNRS - INP - UT3 - UT1 - UT2J

UNIVERSITÉ TOULOUSE CAPITOLE

Model Lake: a New Alternative for Machine Learning Models Management and Governance

Moncef Garouani [1], Franck Ravat [1], Nathalie Valles-Parlangeau [2]

[1] IRIT, UMR5505 CNRS, Université Toulouse Capitole, Toulouse, France
[2] LIUPPA, Université de Pau et des Pays de l'Adour, Anglet, France

moncef.garouani@irit.fr

Slides available at mgarouani.fr/talks -> WISE 2024 Presentation

PLAN❯

1 Introduction & Context

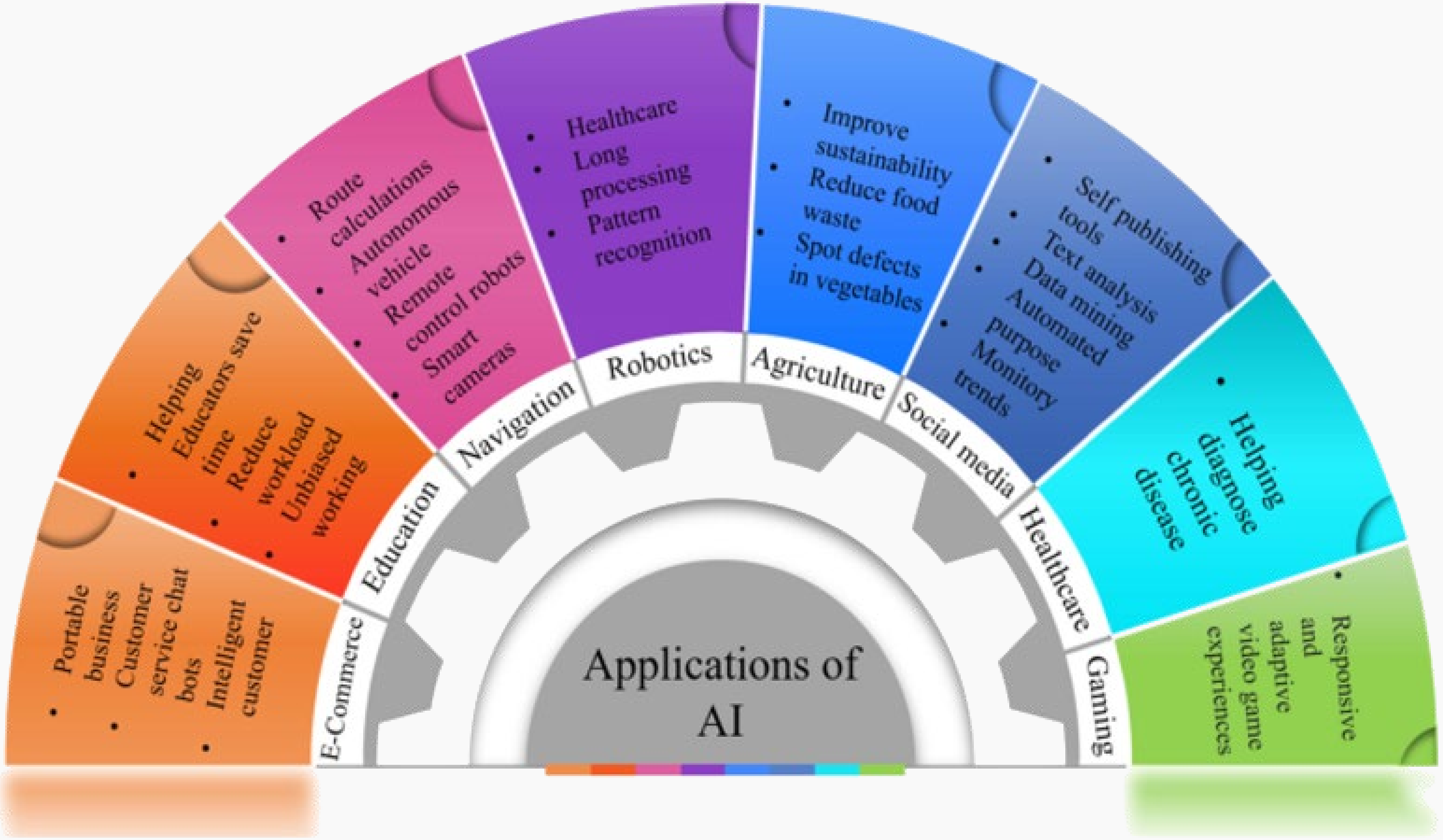
2 Problem Statement

3 Model Lake

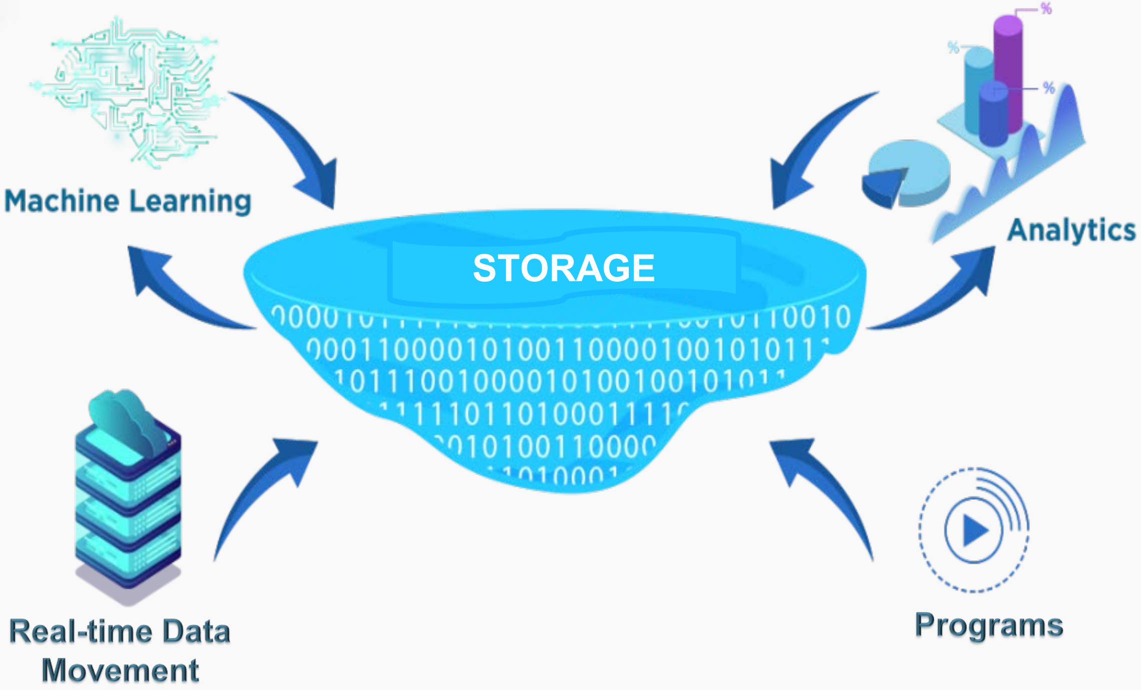
4 Model Lake Management System

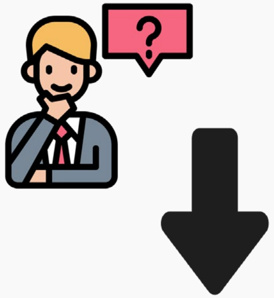
5 Conclusion and Discussion

Context: Success of Machine Learning & Deep Learning (1/2)



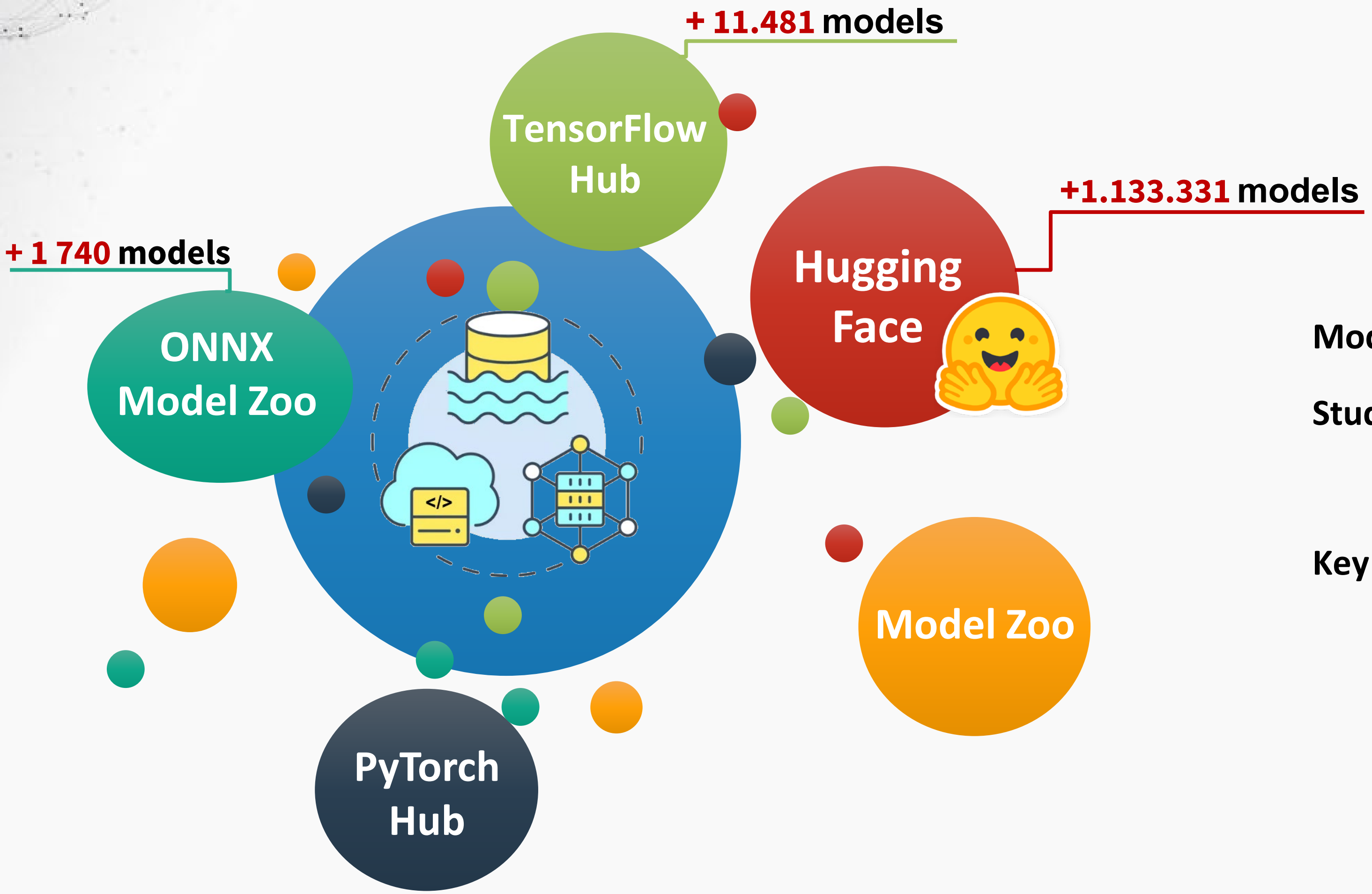
Context: Management problem of ML/DL Models (2/2)





- *How can we manage and understand many available ML models?*
- *How can we know what a model does and how it was trained?*
- *How can we ensure ethical use and trace model lineage?*
- *How can we ensure compliance with regulations?*
- *How can we improve models to avoid mistakes?*

Model Repository, Registry, Zoo, ...



Model Reuse in the Hugging Face Registry (W. Jiang et al.):

Study Approach:

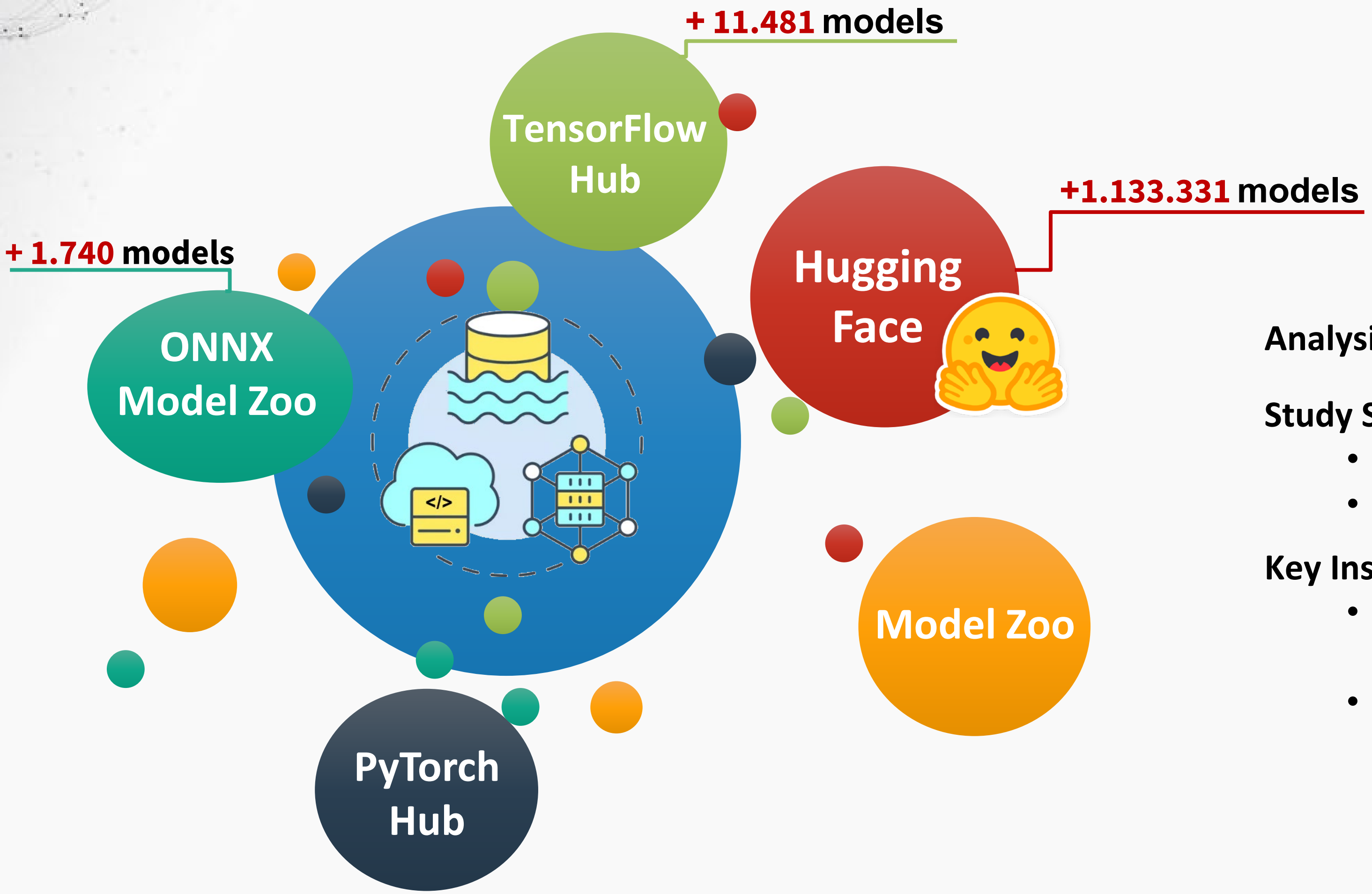
- Interviews with 12 practitioners to identify challenges.
- Analysis of 63,182 models in the registry.

Key Challenges:

- Missing attributes: model lineage, training data.
- Disparities in claimed vs. actual performance.
- Privacy and ethical concerns due to opaque data lineage.

W. Jiang et al. An Empirical Study of Pre-Trained Model Reuse in the Hugging Face Deep Learning Model Registry. 2023. arXiv: 2303.02552.

Model Repository, Registry, Zoo, …



Analysis of Hugging Face Model Cards (W. Liang et al.):

Study Scope:

- Analyzed 74.970 model repositories from 20.455 user accounts.
- Found only 32.111 repositories (44.2%) include model cards.

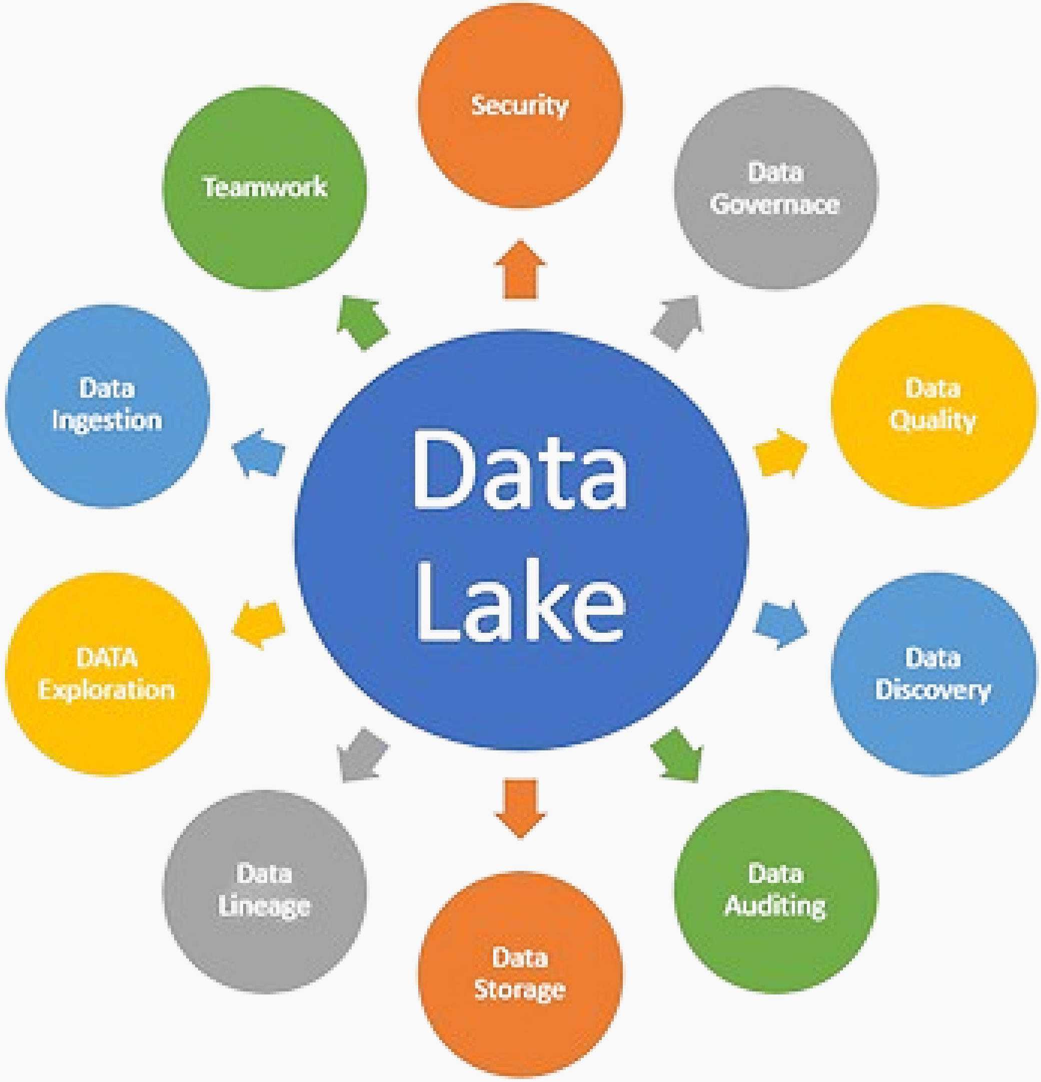
Key Insights:

- Over 56% of models lack proper documentation, reducing reliability.
- **Highlights the need for data-centric approaches to improve model quality and support responsible AI development.**

W. Liang et al. What's documented in AI? Systematic Analysis of 32K AIModel Cards. 2024. arXiv: 2402.05160.

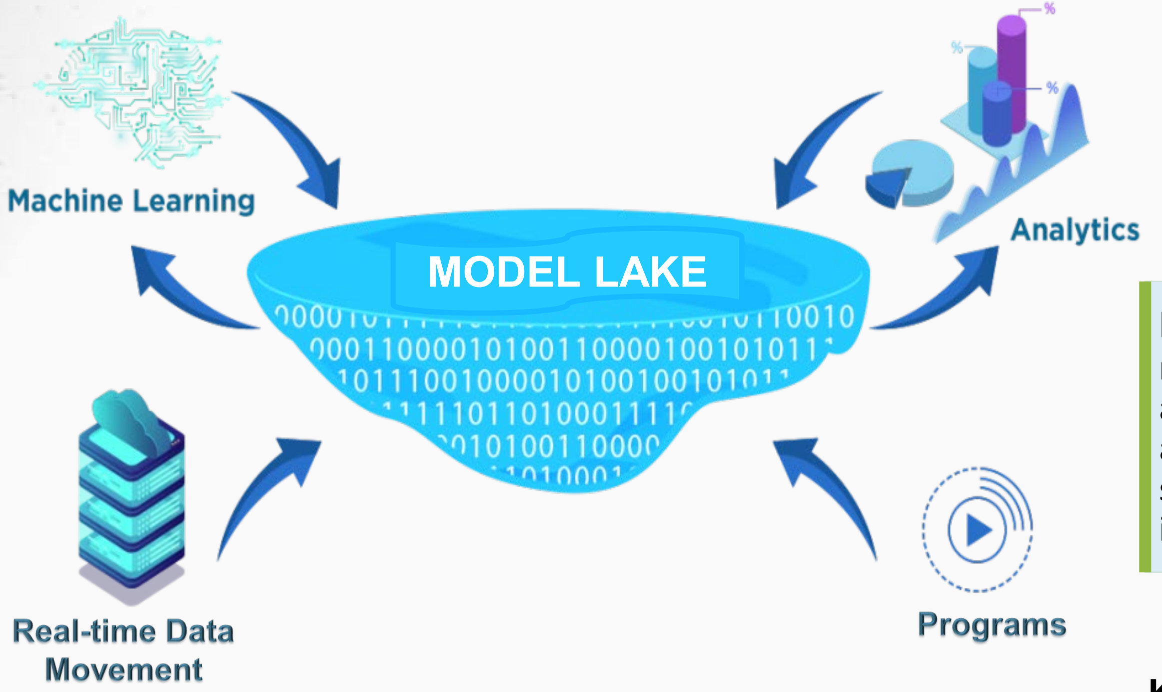
Model Lake

- ❑ Data Lake efficiency



Model Lake

❑ Model Lake Definition



Definition. Model Lake stands as an integrated ecosystem encompassing respectively the input, process, output and governance aspects of both mined data and developed models. It acts as a centralized hub and management system accommodating diverse data and model types, meeting the requirements of various stakeholders including data engineers, data scientists, data analysts, and business intelligence professionals.

Key Functionalities:

- Raw data ingestion, processing, storage, and governance.
- Model training, fine-tuning, review, and monitoring.
- Data, model, and code provenance, management, and governance.

Model Lake

❑ Model Lake Architecture

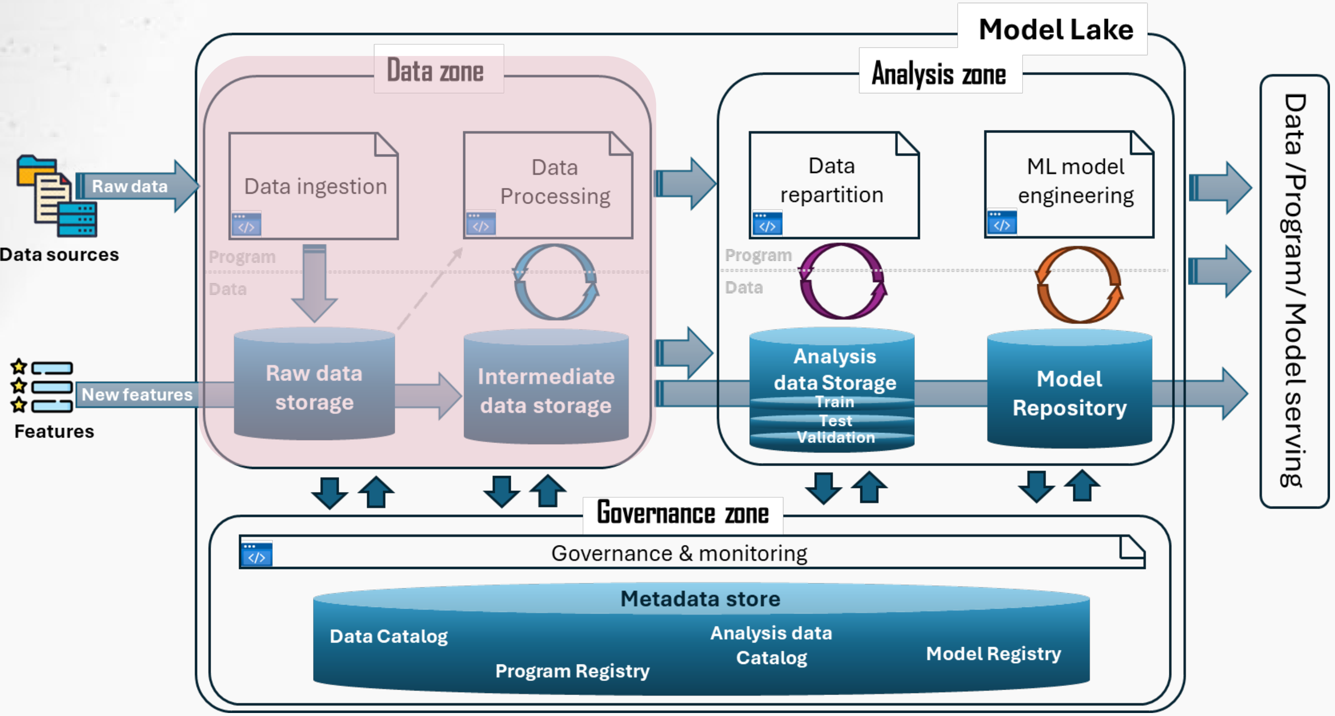


Fig. 1. The proposed Model Lake architecture.

Data Zone :

- ❑ **Purpose**: Manages data ingestion, processing, and intermediate storage.

  •**Ingestion**: Connects to data sources for extraction and change tracking.
  •**Processing**: Standardizes raw data with operations like integration, cleaning, transformation, and reduction.
  •**Intermediate Storage**: Saves processed data and metadata for lineage tracking.

- ❑ **Key Role**: Prepares data for analysis.

Model Lake

❑ Model Lake Architecture

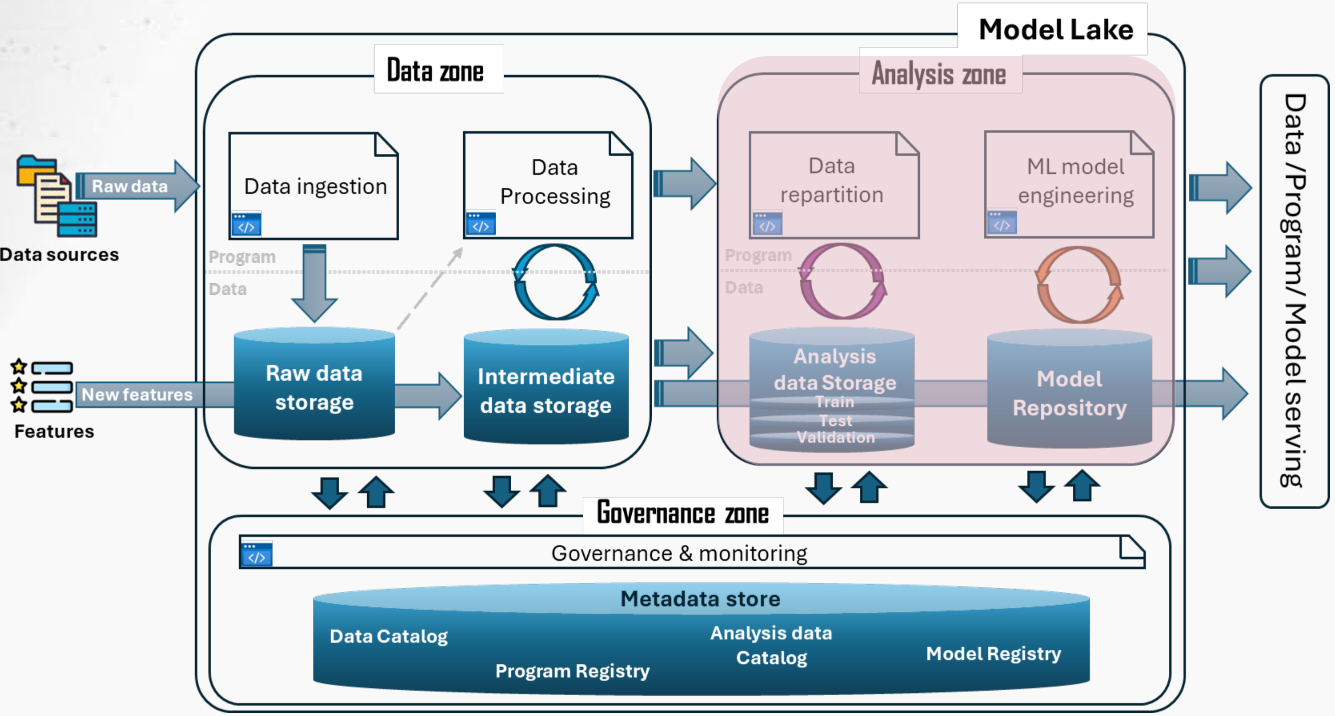


Fig. 1. The proposed Model Lake architecture.

Analysis Zone :

❑ **Purpose**: Central hub for data exploration and ML model development.

❑ **Key Features:**

•Advanced data exploration (meta-features, attributes, transformations).

•ML model development and evaluation.

•Model storage for production.

❑ **Additional Capabilities:**

•Continuous monitoring and feedback for performance and reliability.

•Lineage tracking, model comparison, auditing, and compliance.

Model Lake

❑ Model Lake Architecture

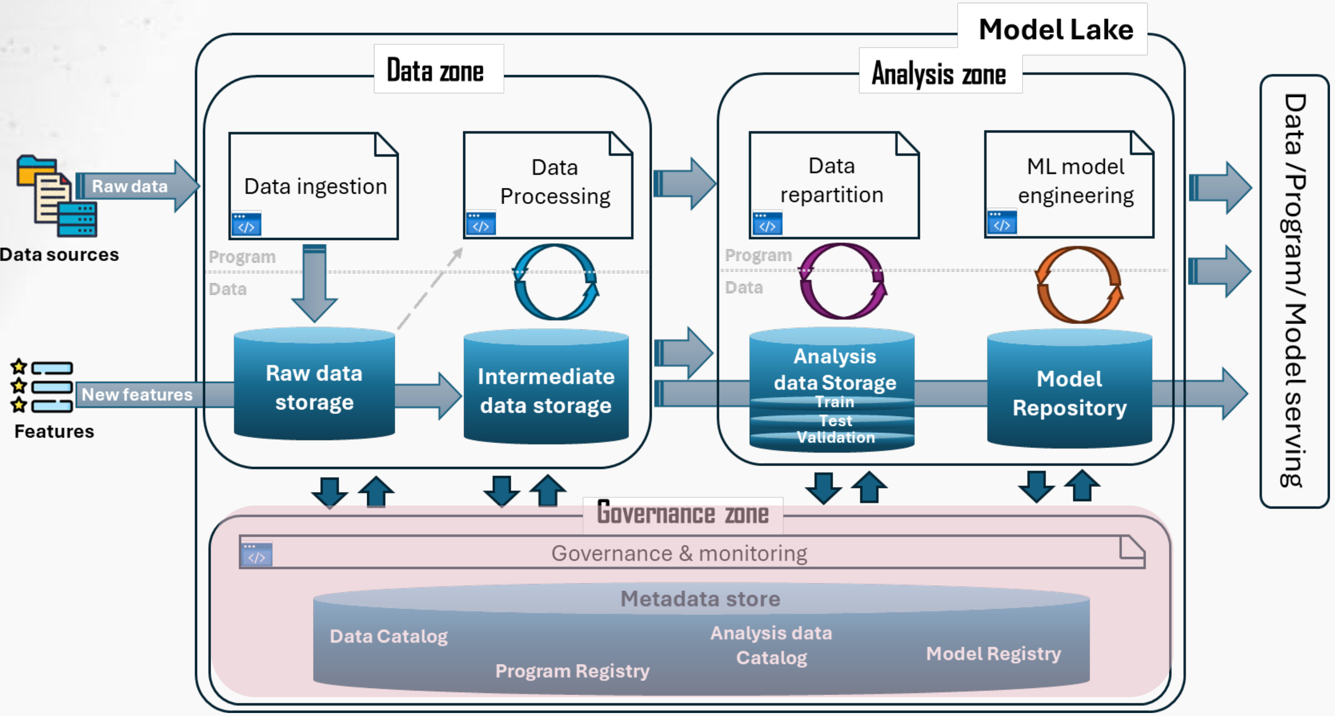


Fig. 1. The proposed Model Lake architecture.

Governance and Management Zone:

•**Purpose**: ensure Data, Program, Model security, lifecycle management, access, and metadata management.

•**Preventing Model Swamp**:

- •Maintains accessibility and usability of the Model Lake.

•**Metadata Store**:

- •Records metadata for all ML workflow tasks and iterations.
- •Tracks job details (e.g., training date, artifact sources).

•**Model Lineage**:

- •Combines data, model, and code lineage.
- •Tracks metadata like feature data sources, parameters, and performance metrics.
- •Ensures full traceability for each registered model.

Model Lake Metadata Management

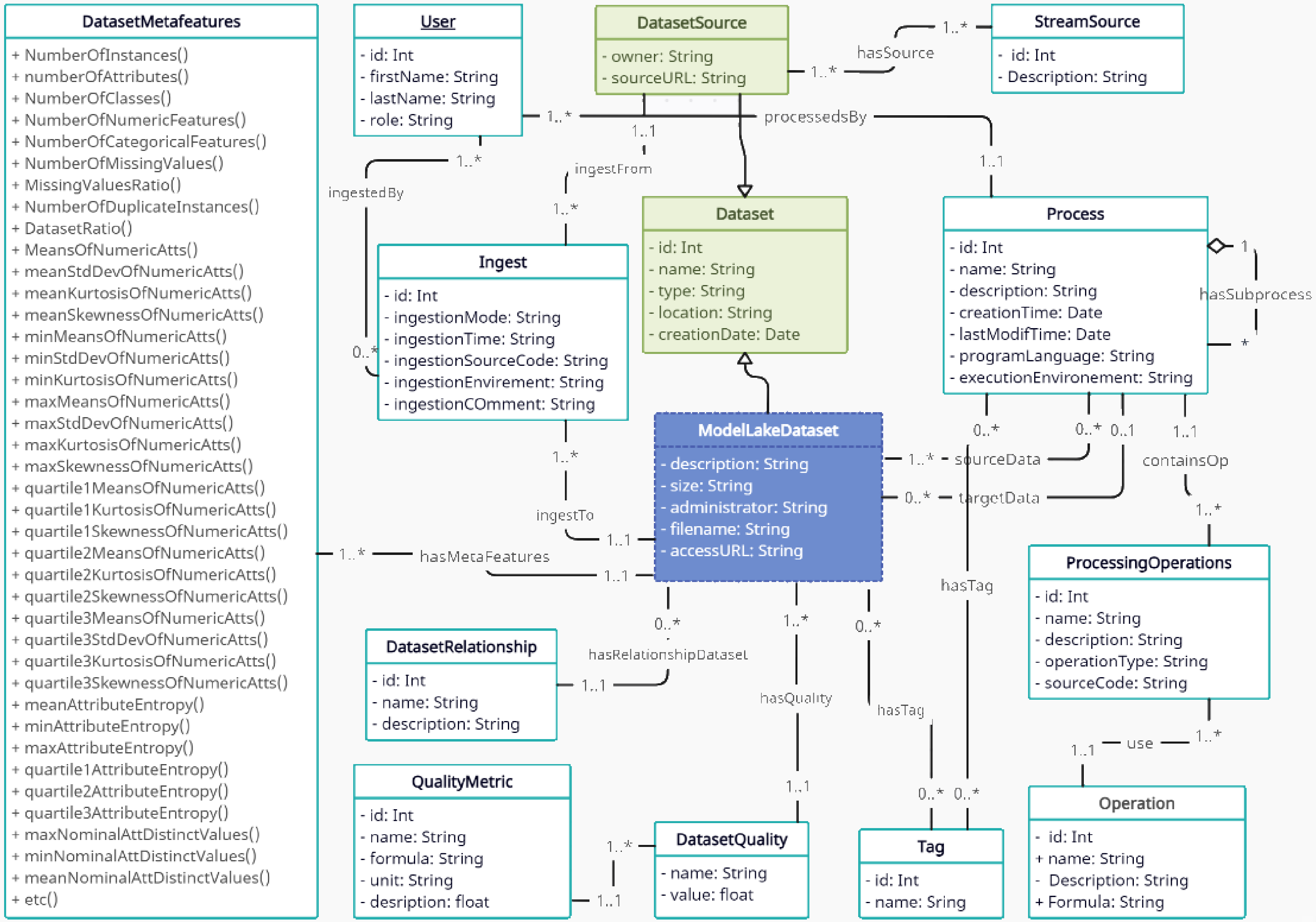
❑ Metadata Model on Data zone

We adopt the 5W1H (What, Who, Where, When, why, how) method to facilitate a systematic understanding of data ingestion and processing. This method prompts the following inquiries:

– **What**: Identifying external data sources and the nature of ingestion activities (ingested datasets, their quality, security level, and interrelations).
– **Who**: Determining ownership of the source data, as well as the individuals responsible for data ingestion and processing.
– **Where**: Locating the storage sites for ingested and processed datasets and associated data ingestion/processing code.
– **When**: Establishing timelines for the ingestion and processing of datasets.
– **Why**: Understanding the purpose behind the data processing activities.
– **How**: Understanding the ingestion and processing operations.

Model Lake Metadata Management

❑ Metadata Model on Data zone



DatasetMetafeatures
- + NumberOfInstances()
- + numberOfAttributes()
- + NumberOfClasses()
- + NumberOfNumericFeatures()
- + NumberOfCategoricalFeatures()
- + NumberOfMissingValues()
- + MissingValuesRatio()
- + NumberOfDuplicateInstances()
- + DatasetRatio()
- + MeansOfNumericAtts()
- + meanStdDevOfNumericAtts()
- + meanKurtosisOfNumericAtts()
- + meanSkewnessOfNumericAtts()
- + minMeansOfNumericAtts()
- + minStdDevOfNumericAtts()
- + minKurtosisOfNumericAtts()
- + maxMeansOfNumericAtts()
- + maxStdDevOfNumericAtts()
- + maxKurtosisOfNumericAtts()
- + maxSkewnessOfNumericAtts()
- + quartile1MeansOfNumericAtts()
- + quartile1KurtosisOfNumericAtts()
- + quartile1SkewnessOfNumericAtts()
- + quartile2MeansOfNumericAtts()
- + quartile2KurtosisOfNumericAtts()
- + quartile2SkewnessOfNumericAtts()
- + quartile3MeansOfNumericAtts()
- + quartile3StdDevOfNumericAtts()
- + quartile3KurtosisOfNumericAtts()
- + quartile3SkewnessOfNumericAtts()
- + meanAttributeEntropy()
- + minAttributeEntropy()
- + maxAttributeEntropy()
- + quartile1AttributeEntropy()
- + quartile2AttributeEntropy()
- + quartile3AttributeEntropy()
- + maxNominalAttDistinctValues()
- + minNominalAttDistinctValues()
- + meanNominalAttDistinctValues()
- + etc()

User
- - id: Int
- - firstName: String
- - lastName: String
- - role: String

DatasetSource
- - owner: String
- - sourceURL: String

StreamSource
- - id: Int
- - Description: String

Ingest
- - id: Int
- - ingestionMode: String
- - ingestionTime: String
- - ingestionSourceCode: String
- - ingestionEnvirement: String
- - ingestionCOmment: String

Dataset
- - id: Int
- - name: String
- - type: String
- - location: String
- - creationDate: Date

Process
- - id: Int
- - name: String
- - description: String
- - creationTime: Date
- - lastModifTime: Date
- - programLanguage: String
- - executionEnvironement: String

ModelLakeDataset
- - description: String
- - size: String
- - administrator: String
- - filename: String
- - accessURL: String

ProcessingOperations
- - id: Int
- - name: String
- - description: String
- - operationType: String
- - sourceCode: String

DatasetRelationship
- - id: Int
- - name: String
- - description: String

QualityMetric
- - id: Int
- - name: String
- - formula: String
- - unit: String
- - desription: float

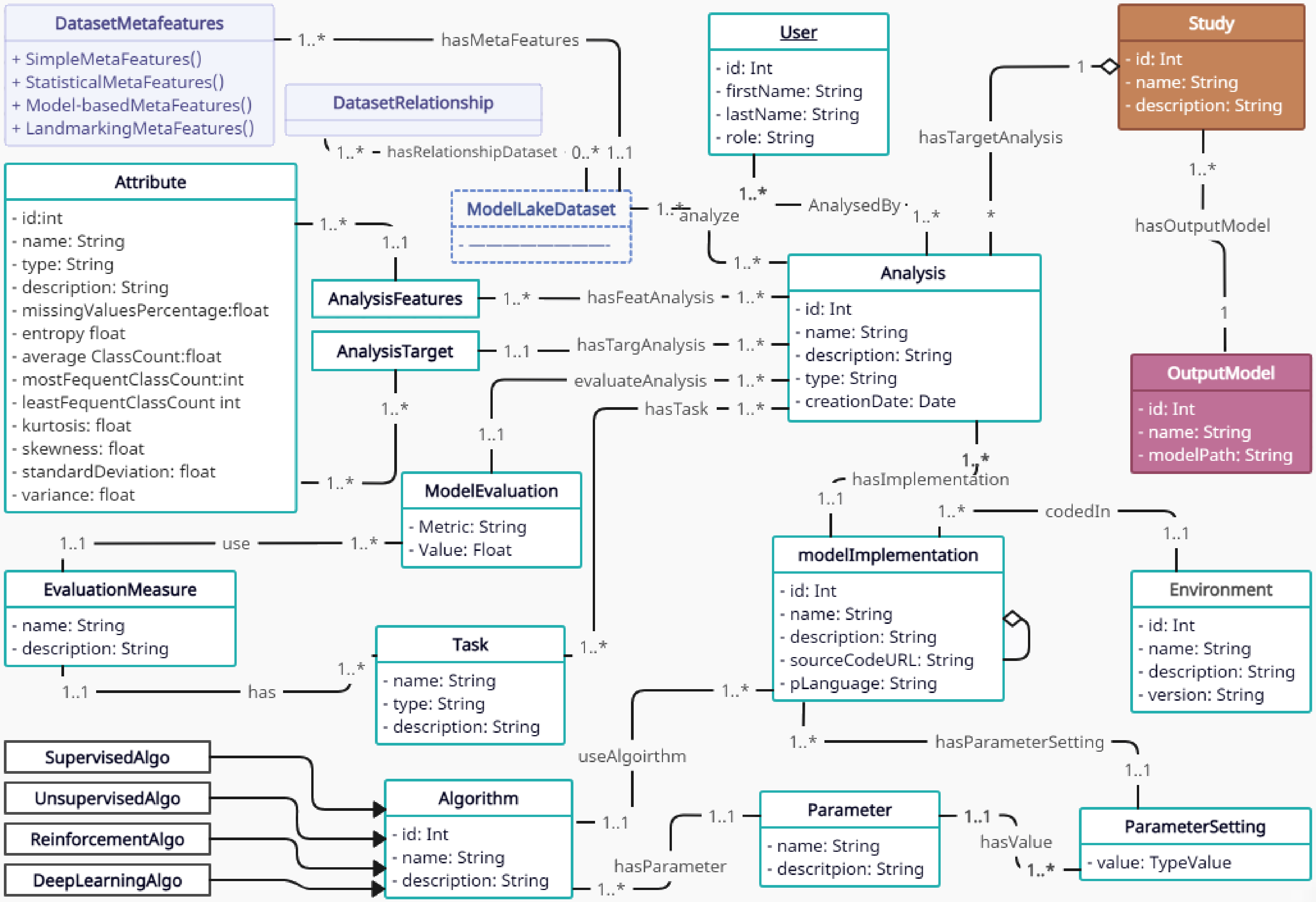
DatasetQuality
- - name: String
- - value: float

Tag
- - id: Int
- - name: Sring

Operation
- - id: Int
- + name: String
- - Description: String
- + Formula: String

Model Lake Metadata Management

❑ Metadata Model on Data Analysis



DatasetMetafeatures
+ SimpleMetaFeatures()
+ StatisticalMetaFeatures()
+ Model-basedMetaFeatures()
+ LandmarkingMetaFeatures()

DatasetRelationship

User
- id: Int
- firstName: String
- lastName: String
- role: String

Study
- id: Int
- name: String
- description: String

Attribute
- id:int
- name: String
- type: String
- description: String
- missingValuesPercentage:float
- entropy float
- average ClassCount:float
- mostFequentClassCount:int
- leastFequentClassCount int
- kurtosis: float
- skewness: float
- standardDeviation: float
- variance: float

ModelLakeDataset
- ——————————

AnalysisFeatures

AnalysisTarget

Analysis
- id: Int
- name: String
- description: String
- type: String
- creationDate: Date

OutputModel
- id: Int
- name: String
- modelPath: String

ModelEvaluation
- Metric: String
- Value: Float

EvaluationMeasure
- name: String
- description: String

Task
- name: String
- type: String
- description: String

modelImplementation
- id: Int
- name: String
- description: String
- sourceCodeURL: String
- pLanguage: String

Environment
- id: Int
- name: String
- description: String
- version: String

SupervisedAlgo

UnsupervisedAlgo

ReinforcementAlgo

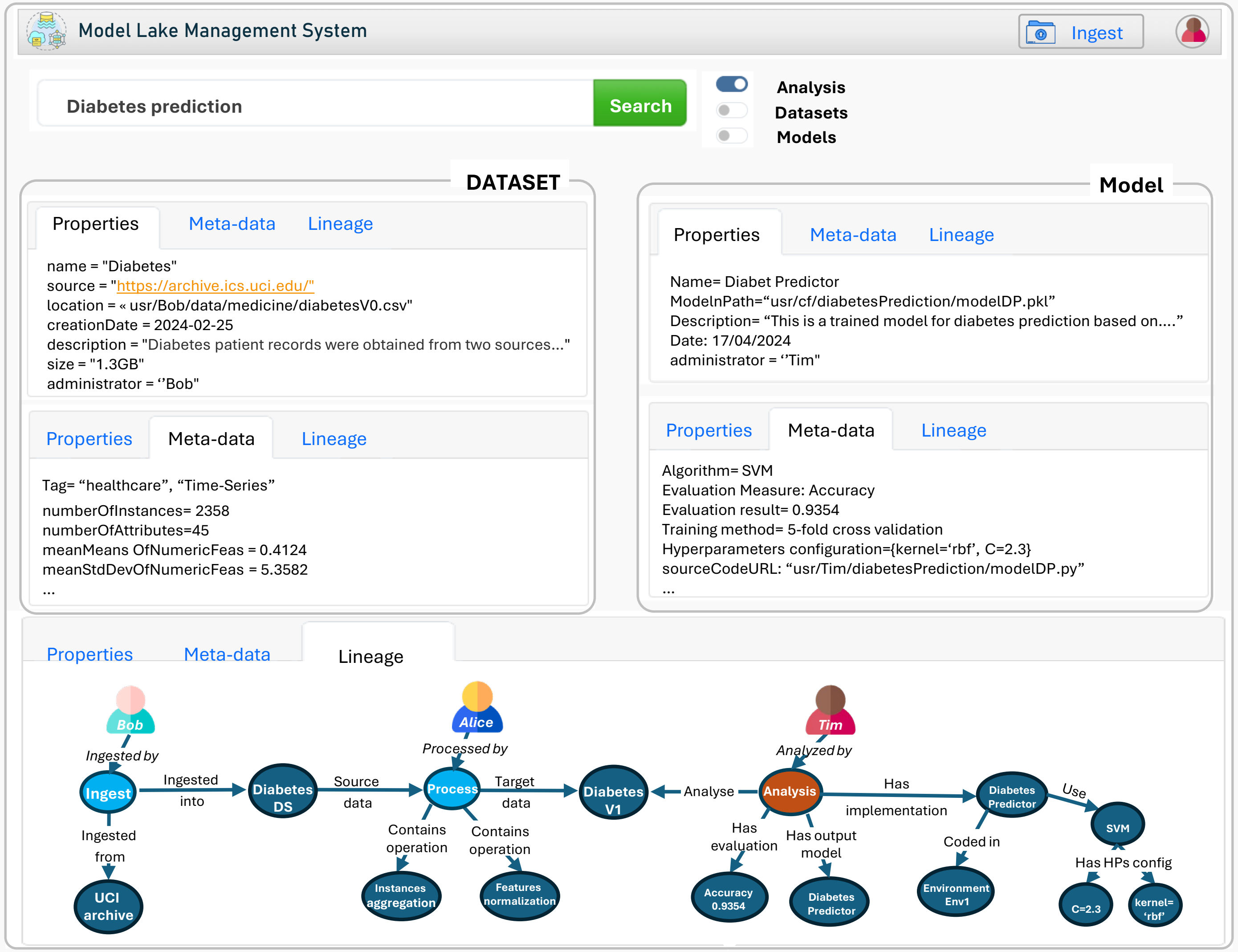
DeepLearningAlgo

Algorithm
- id: Int
- name: String
- description: String

Parameter
- name: String
- descritpion: String

ParameterSetting
- value: TypeValue

Model Lake Metadata Management



Diabetes prediction

DATASET

Properties | Meta-data | Lineage

name = "Diabetes"
source = "https://archive.ics.uci.edu/"
location = « usr/Bob/data/medicine/diabetesV0.csv"
creationDate = 2024-02-25
description = "Diabetes patient records were obtained from two sources..."
size = "1.3GB"
administrator = ''Bob"

Properties | Meta-data | Lineage

Tag= “healthcare”, “Time-Series”
numberOfInstances= 2358
numberOfAttributes=45
meanMeans OfNumericFeas = 0.4124
meanStdDevOfNumericFeas = 5.3582
...

Model

Properties | Meta-data | Lineage

Name= Diabet Predictor
ModelnPath=“usr/cf/diabetesPrediction/modelDP.pkl”
Description= “This is a trained model for diabetes prediction based on....”
Date: 17/04/2024
administrator = ‘’Tim"

Properties | Meta-data | Lineage

Algorithm= SVM
Evaluation Measure: Accuracy
Evaluation result= 0.9354
Training method= 5-fold cross validation
Hyperparameters configuration={kernel=‘rbf’, C=2.3}
sourceCodeURL: “usr/Tim/diabetesPrediction/modelDP.py”
...

Properties | Meta-data | Lineage

Conclusion & Perspectives

Conclusion

- Rapid ML model growth presents both opportunities and challenges,
- Lack of standardized management risks limiting their full potential.
- **Model Lakes**: A promising solution for centralized, scalable model management.
- Success requires collaboration across the ecosystem and commitment to responsible AI.

Perspectives

- Expand model lake system to include additional analysis types and ML pipeline artifacts.
- Develop a **recommender system** to enhance data and model search and discovery.

THANK YOU!

IRIT Institut de Recherche en Informatique de Toulouse
CNRS - INP - UT3 - UT1 - UT2J

UNIVERSITÉ TOULOUSE CAPITOLE

Model Lake: a New Alternative for Machine Learning Models Management and Governance

Moncef Garouani [1], Franck Ravat [1], Nathalie Valles-Parlangeau [2]

[1] IRIT, UMR5505 CNRS, Université Toulouse Capitole, Toulouse, France
[2] LIUPPA, Université de Pau et des Pays de l’Adour, Anglet, France

moncef.garouani@irit.fr

Slides available at mgarouani.fr/talks -> WISE 2024 Presentation