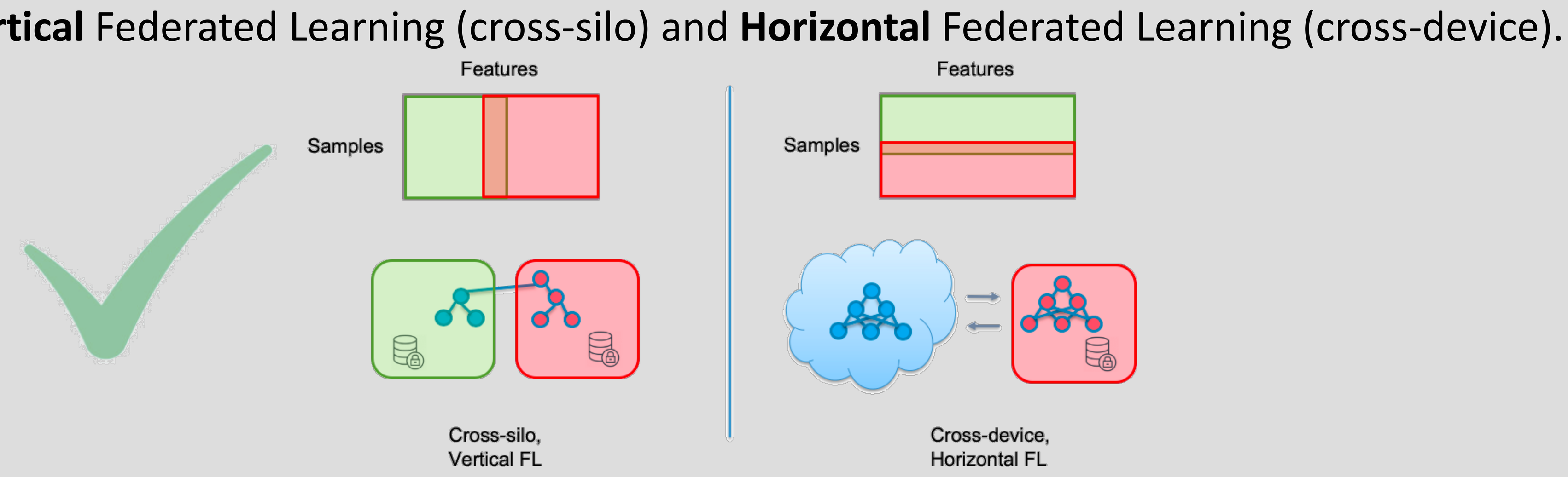
A System for Time Series Feature Extraction in Federated Learning

Siqi WANG*, Jiashu LI*, Mian LU*, Zhao ZHENG*, Yuqiang CHEN*, Bingsheng HE^
*4Paradigm Inc., ^National University of Singapore

Introduction

Federated Learning

Federated learning (FL) enables collaborative learning without revealing raw data, which exploits value of data from different entities while preserving data **privacy**.
Vertical Federated Learning (cross-silo) and **Horizontal** Federated Learning (cross-device).



Vertical FL that enables collaborations between different companies/industry/data silos, is expected to have more applications and exhibits larger value.

Popular open-sourced frameworks: FATE, PaddleFL, FedLearner, FedML, TFF, PySyft, etc..
Pros: Supports many federated learning algorithms, including horizontal, vertical, LR, GBDT, DNN and so on.
Cons: No indepth discussions on feature engineering, especially times series feature engineering. Not able to maximize the value embedded in data.

Time series Feature Engineering

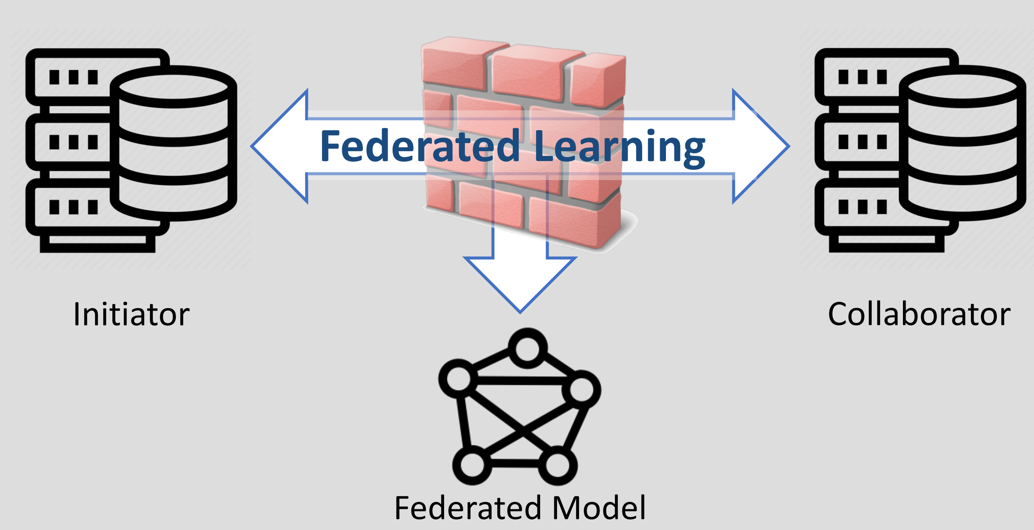
In actual industrial scenarios, data generally contains **time series** information. Example scenarios are Internet of Things, economics and finance, environmental monitoring, manufacturing, agriculture, etc. **Time-series feature extraction** also plays an important role in 4Paradigm's industrial products, including HyperCycle ML, OpenMLDB, …

We focus on window features, where the feature is extracted together from multiple historical samples. A sliding window is constructed to produce summary statistics.

Federated Learning with TSFE

System Topology

We consider vertical FL scenario where the **Initiator** intends to build a better quality model with the help of the **Collaborator** through federated learning.

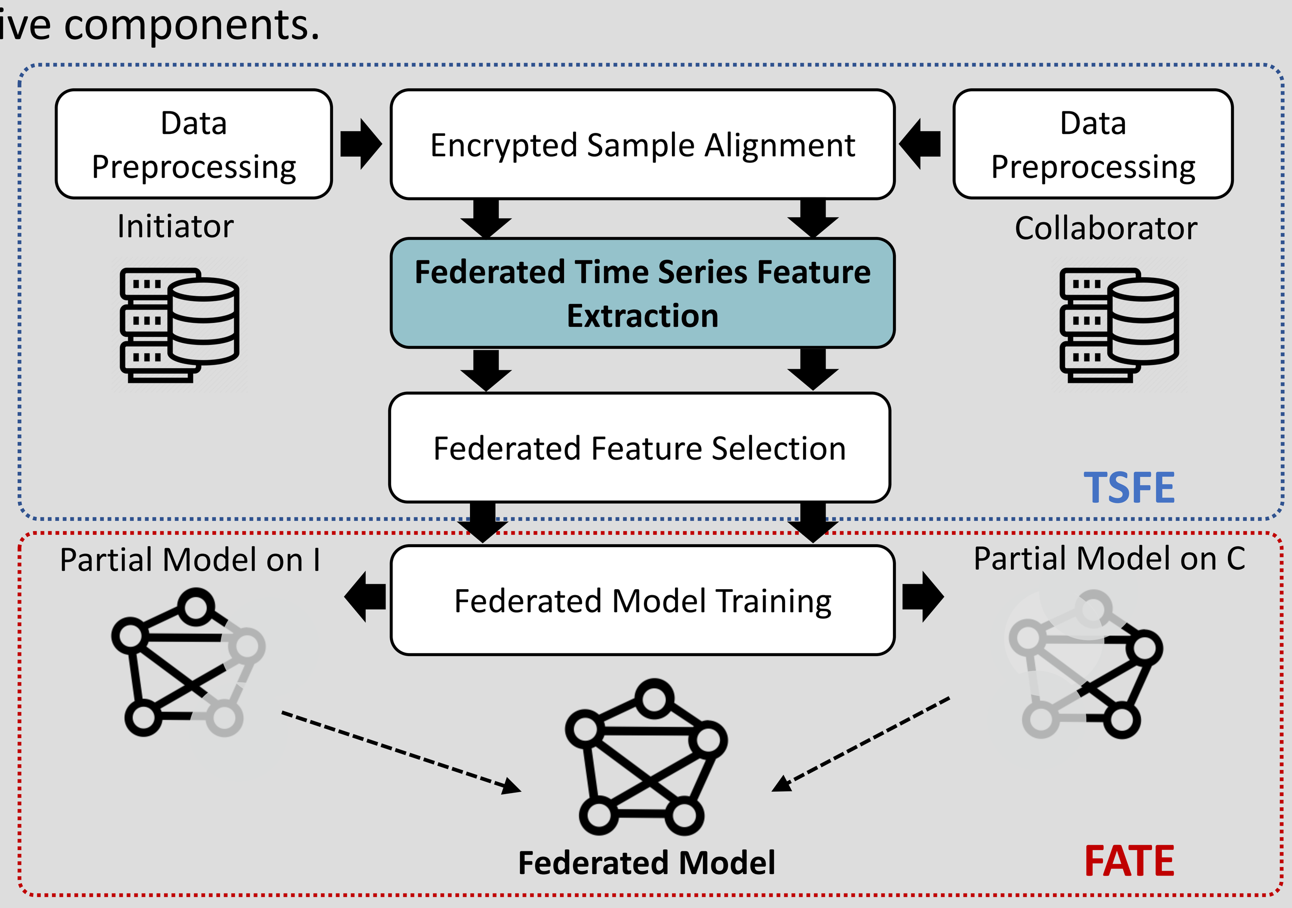


Workflow Block Diagram

Federated time series feature extraction is enabled by two industry solutions: OpenMLDB and **FATE**. TSFE is engaged for data processing and FATE for model training.

The key component in TSFE is the **Federated Time Series Feature Extraction** component. This component is implemented base on OpenMLDB, and the rest are modified from FATE's native components.

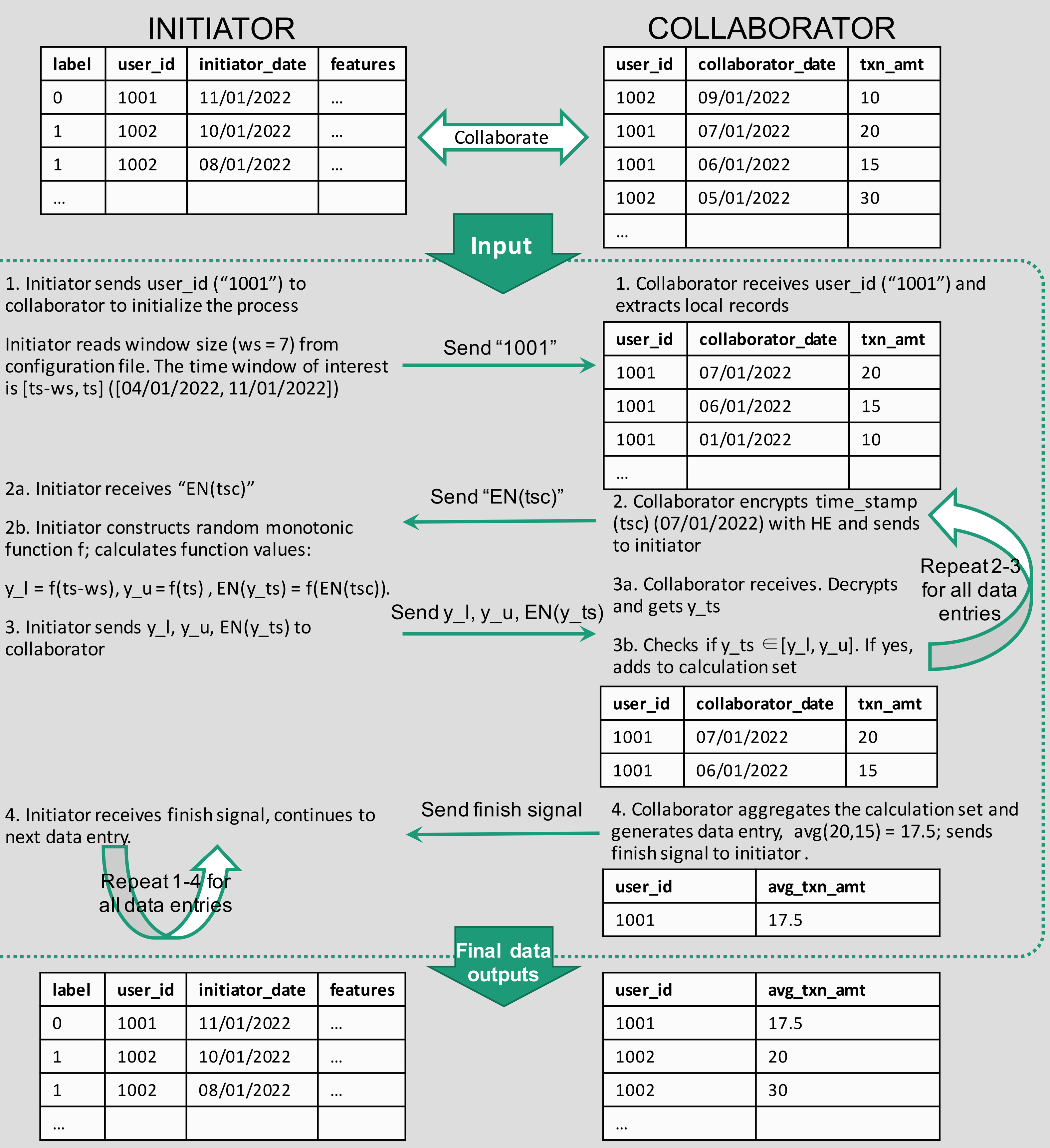
This block is implemented as a self-contained functional block. which can be directly connected to any other functional blocks in FATE.



Algorithm Illustration

Federated Time Series Feature Extraction

The algorithm is based on homomorphic encryption and monotonic function to ensure data privacy. Features extracted by the collaborator remain within the collaborator.



Experiments and Results

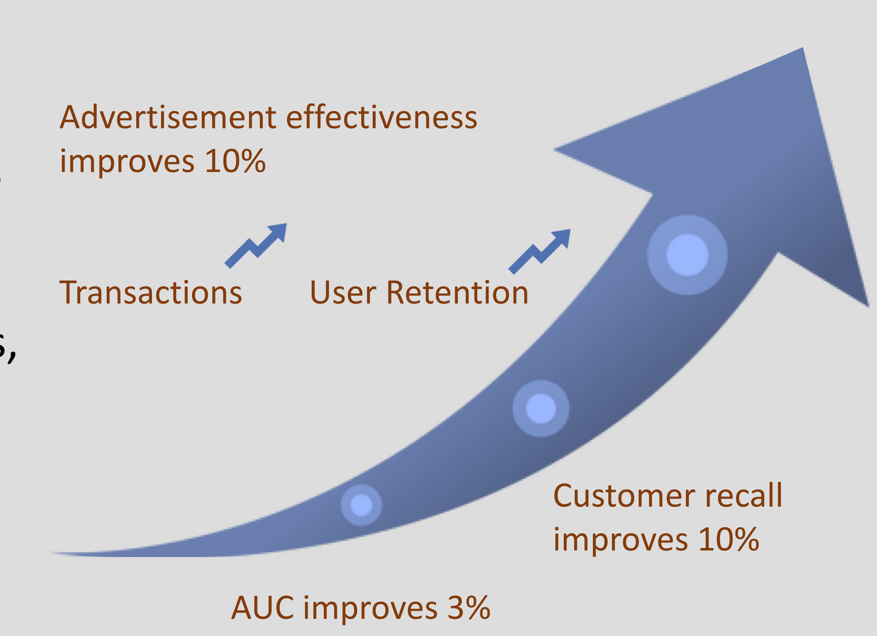
Model Accuracy: AUC improvement and its effect

Scenarios: 3 real-world datasets

- Task: customer purchase intention; credit risk assessment; content recommendation.
- Initiator: main table, **identifier**, **time stamps**, **label**
- Collaborator: **identifier**, additional features, **time stamps**

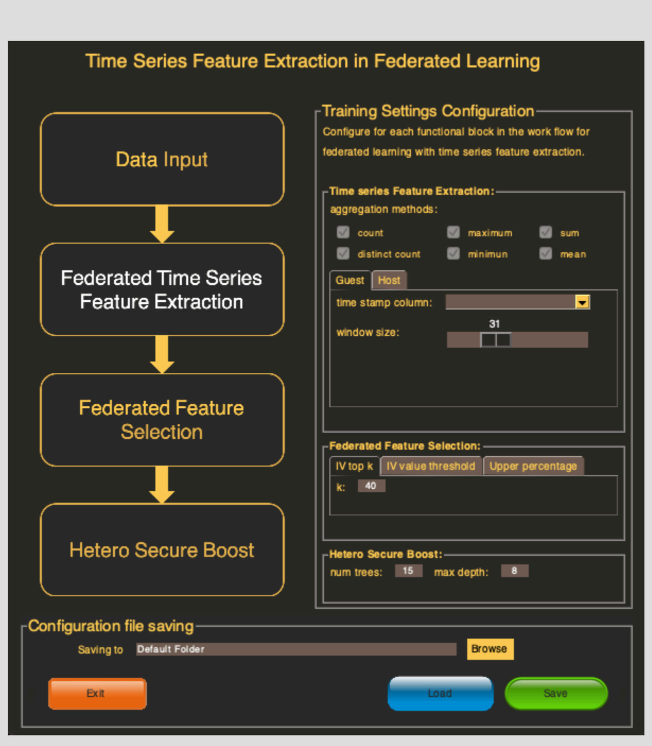
Results:

- AUC on average improves by 3%.
- Under the same quantile setting, precision is higher, recall is higher: 0.66 → 0.72

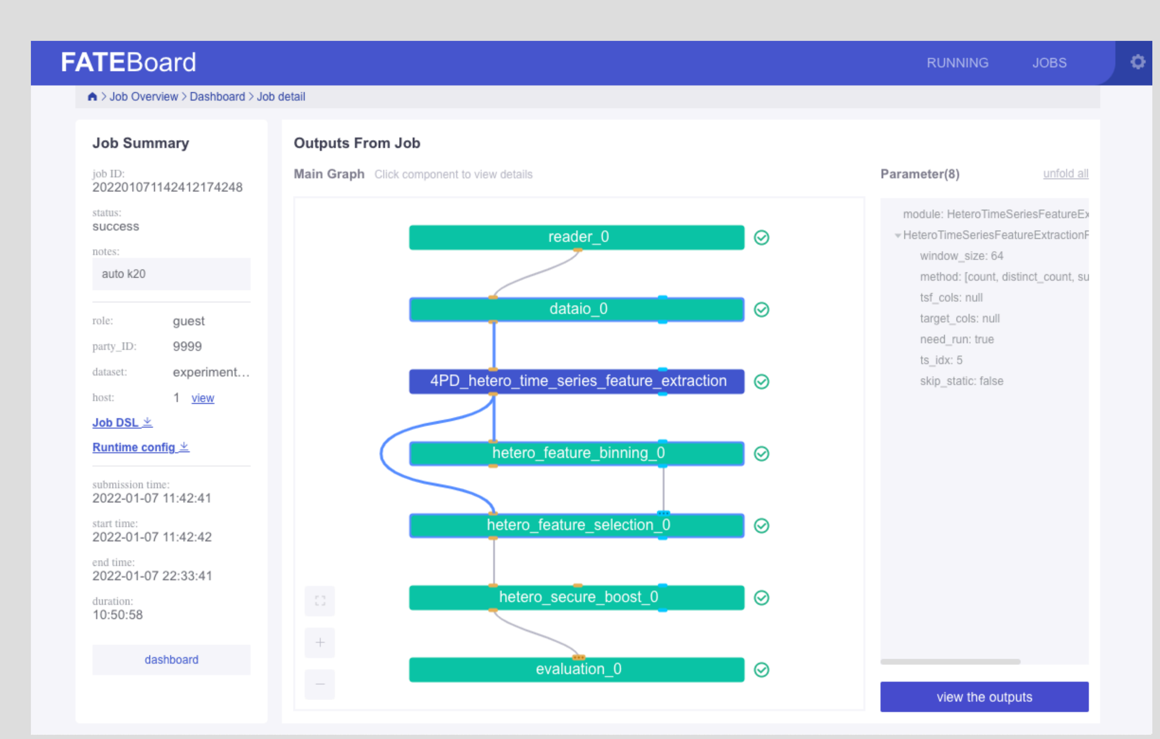


User Interfaces

Parameter Setting UI



FATEBoard Job Monitor



Contact: wangsiqi@4paradigm.com
Project open-sourced at
https://github.com/4paradigm/tsfe