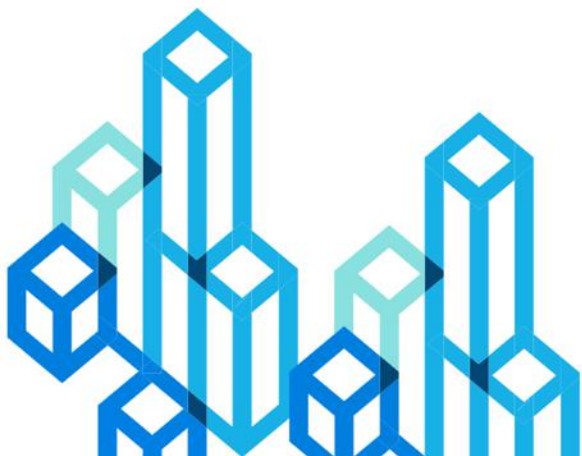


萨摩耶AI平台架构与实现



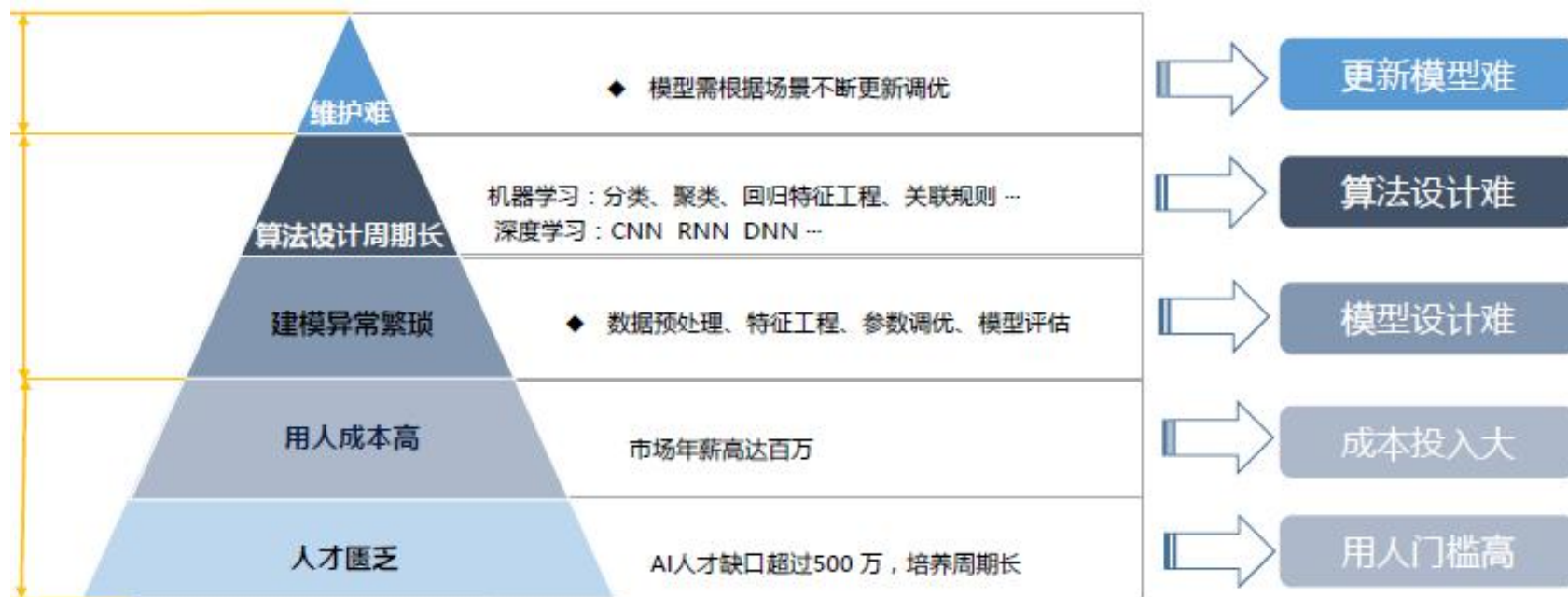
- 惠普（大数据专家）
- 2015年加入萨摩耶数科
- 现担任AI平台负责人
- 擅长的领域：实时和离线数仓、在线分析、图计算、AI平台

1. 背景介绍

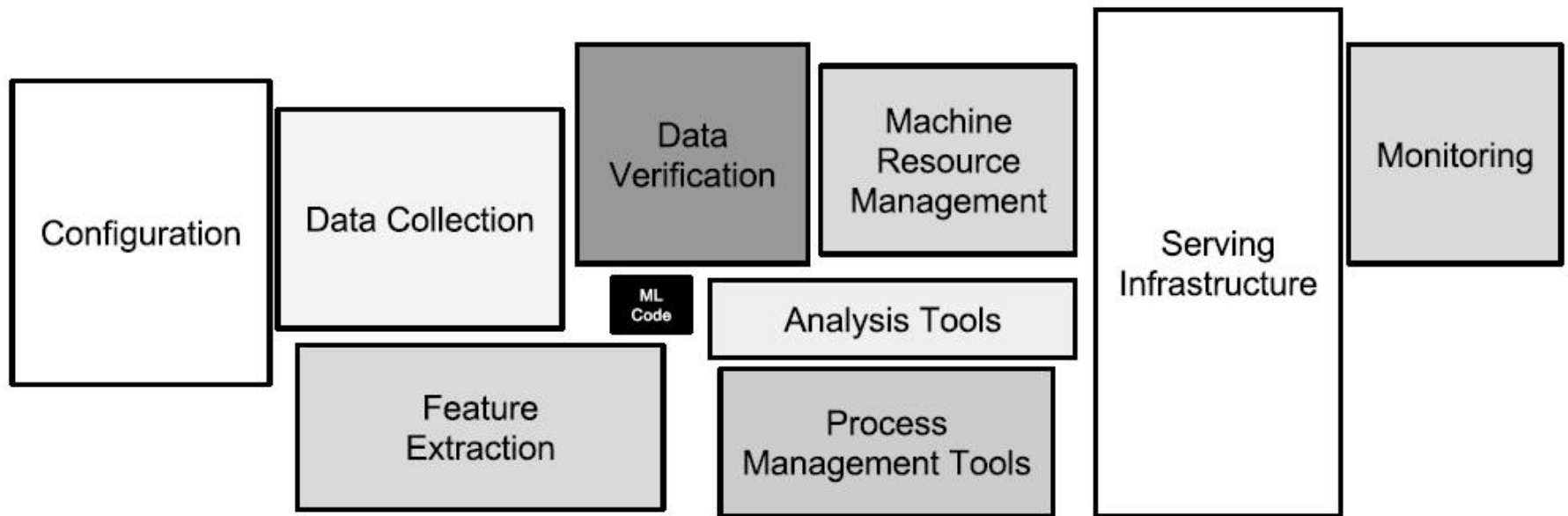
2. 平台功能

3. 架构与实现

4. 未来规划



"Only a fraction of ML systems is composed of ML code"



Source: NIPS-2015-hidden-technical-debt-in-machine-learning-systems-Paper

数据获取

- 从各个数据源高效地获取数据

数据分析

- 变量分布、相关性、重要性、缺失率等

数据处理

- 数据清洗、编码、数据集拆分等。

模型训练

- 选择算法、调参、训练模型。

模型评估

- 稳定性、精确率、召回率等

模型部署

- 微服务、边缘设备、批量服务等。

监报告警

- 性能、稳定性、有效性、资源使用等

特征管理

- 离线和在线特征

模型仓库

- 保存模型的存储空间

资源管理

- cpu、内存、磁盘、软件环境等

版本控制

- 数据集、参数、代码、执行环境、模型、日志、评估等

流水线工具

- 将建模流程标准化，实现持续集成

1. 背景介绍

2. 平台功能

3. 架构与实现

4. 未来规划

产品功能——一站式MLOps平台

建模前

数据源

jdbc

用户
密码

加密

数据集

hive

MySQL

CSV

Excel
|
自动更新

项目管理

公司

部门

数据集

组件管理

开发

共享

评价

依赖

复制

收藏

模型环境

语言

框架

类型

系统管理

私有化部署

跨平台部署

建模中

数据质量报告

缺失率

数据集
检查

报告
下载

异常
值

统计
分析

变量
分析

变量分箱

决策树
分箱

等频等
距调参

Caim调
参

Mdlp分
箱

卡方
调参

单调
分箱

特征工程

缺失值
填充

woe转
码

特征导
入

模型算法

xgboost

逻辑回
归

决策树

模型筛选

条件筛
选

模型对
比

图形化建模

拖拽式

自动部
署

断点运
行

代码生
成

发布
调度

所见即
所得

规则自动化

单变量
分析

规则自
动筛选

筛选日
志

决策树
建池

规则集
报告

规则vn
图

模型报告

Ks/auc/
roc

累积风
险报告

Psi/等
频等距

数据提
升度表

建模后

模型部署

一键
部署

文件
导出

规则
导出

模型管理

版本

启停

发布

模型监控

耗时

服务
日志

实例
列表

模型告警

邮件
短信

耗时

自定义
指标

错误
率

进程
实例

跑批

耗时

服务
日志

实例
列表

系统日志

覆盖建模全流程，一站式MLOps平台。

高度灵活和扩展性，自定义组件、推理逻辑、监控指标等。

高度开放性，平台完全基于开源打造，兼容业界标准。

兼顾易用性和专业性，支持多种建模方式，支持代码生成。

监报告警，支持qps、失败率、耗时指标。

模型管理，如发布、回滚、版本、启停、日志、实例等。

支持多种数据源，基于列式存储，兼容多种存储服务。

金融行业算法模板，自动调参，多模型输出。

模型日报，支持输入项、psi、汇总统计等指标。

专业级的工作流调度引擎，可视化ETL及定时任务。

跟特征平台和策略引擎打通，端到端全流程线上化。

数据集分析模板，数据质量报告、分箱、特征衍生。

规则自动化模板，自动搜索最优策略，供策略引擎使用。

组件分享、点赞、使用统计。

本地化部署，保证数据安全。

集成联邦学习。

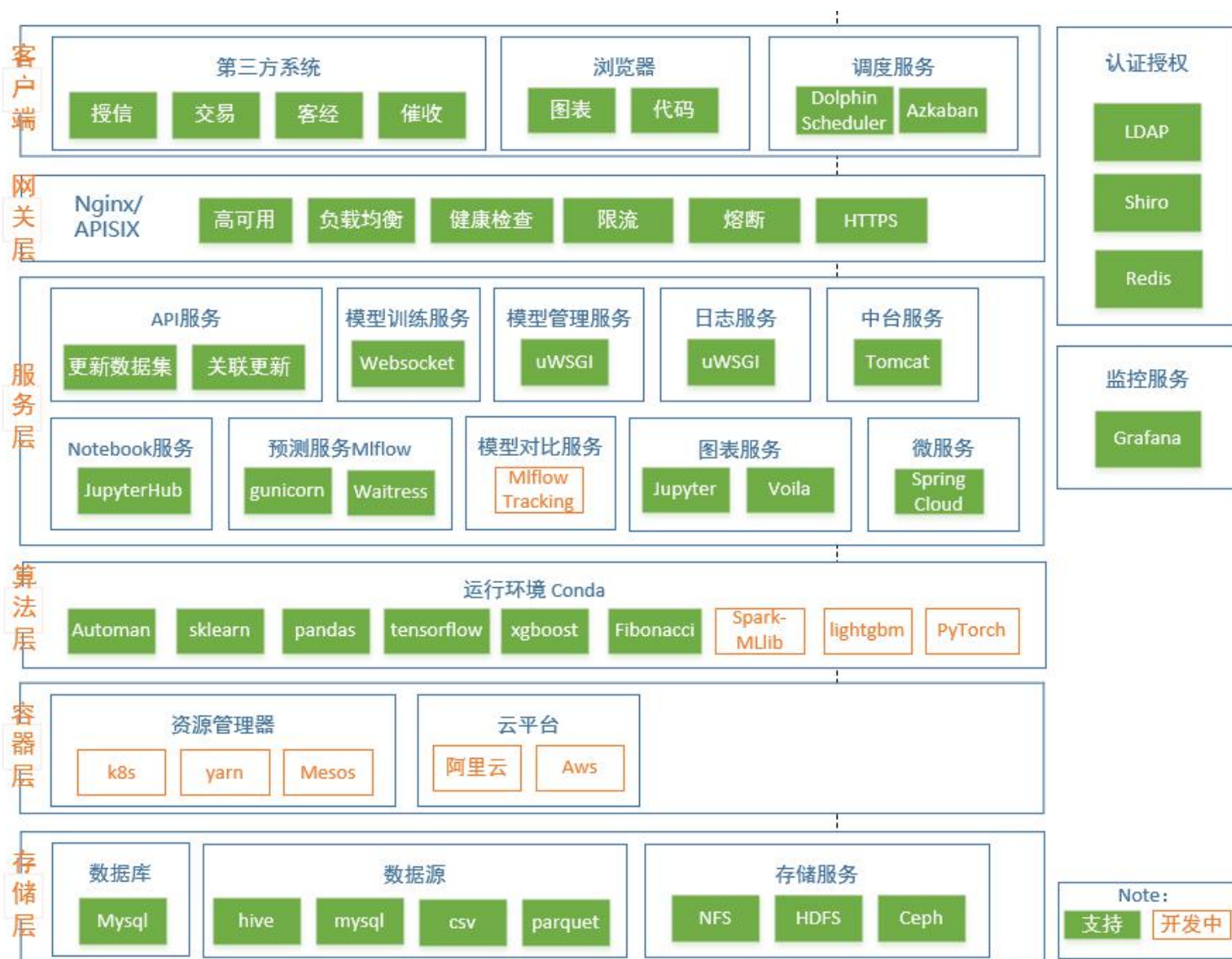
1. 从2019年上线至今，已迭代近10个大版本。
2. 支持全部公司上百个模型，应用在线上风控等场景，日调用量千万级别。
3. 线上模型类型包括分类、文本、语音、图像等。
4. 线上模型平均耗时50ms，峰值qps近100。
5. 业务人员自主上线模型，上线时间平均10分钟。
6. 使用可视化模板建模，开发效率提升一倍。
7. 模型维护效率提升三倍，包括部署、更新、监控、日报、版本和资源管理等。

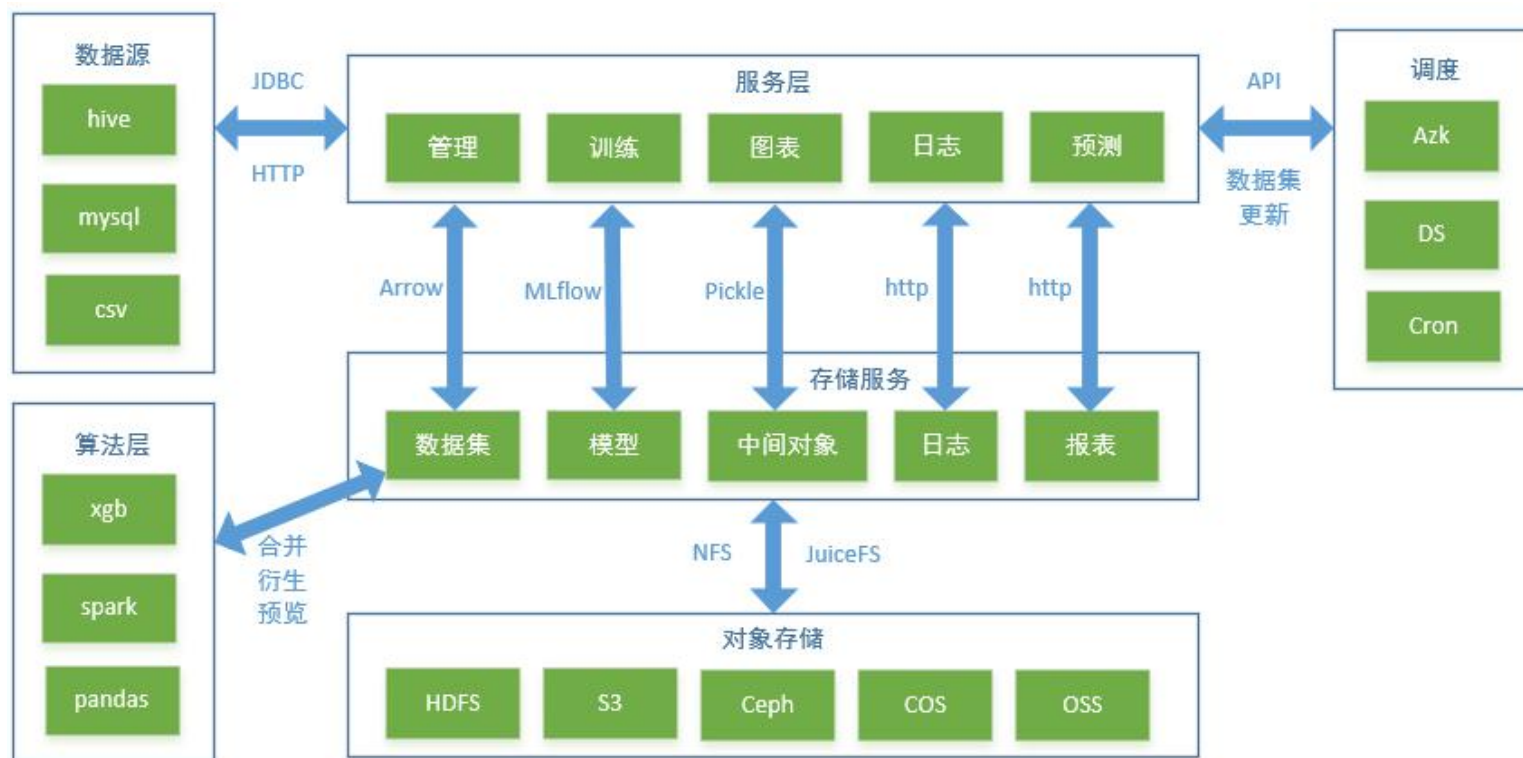
1. 背景介绍

2. 平台功能

3. 架构与实现

4. 未来规划

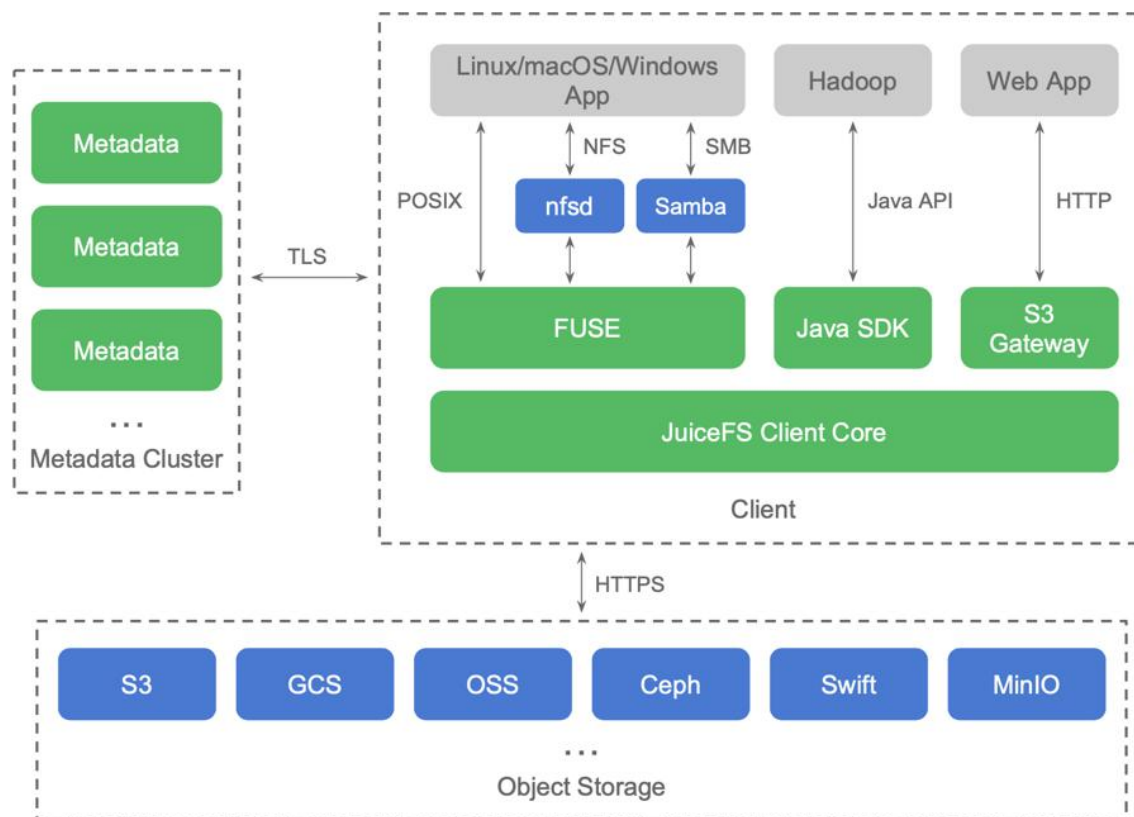




数据集存储格式为parquet有如下优点：

1. Spark默认数据格式，兼容大数据生态
2. 基于列式存储，读写性能好
3. 高效的压缩率
4. 自带元数据，不需额外存储字段信息，保证schema的一致性
5. 基于Arrow实现跨平台的高性能数据传输和读写
6. 通过Spark实现合并、衍生、预览等。

1. POSIX 兼容
2. 丰富的对象存储
3. 云原生
4. 多端共享
5. 强一致性
6. 强悍性能
7. 数据安全
8. 数据压缩



模型组件

- 算法函数，包括输入、输出、代码等

模型环境

- Python环境、镜像等

模型开发

- 可视化建模、notebook建模

模型生成

- 生成模型文件、推理代码、预处理代码等

模型编译

- 将原始信息编译打包成MLflow格式

模型发布

- 发布MLflow模型到推理服务器

模型监控

- 实时监控qps、耗时、错误率等

模型日报

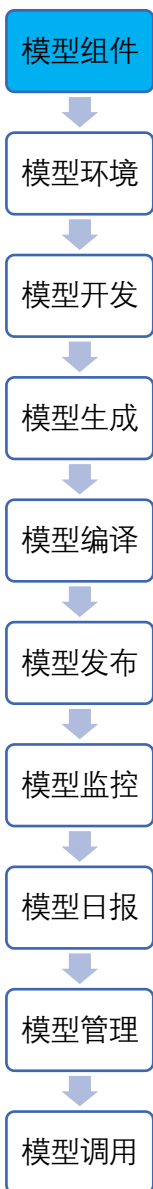
- 模型变量分布、psi、汇总统计等。

模型管理

- 版本管理、资源管理、跑批、试跑、日志、实例管理等

模型调用

- 供第三方系统、离线跑批、调度引擎调用



- 将模型算法抽象为组件，可任意组合生成建模pipeline，具备高度的灵活性与可扩展性。
- 常用算法无需重复开发，一次编写，到处运行。
- 支持主流开源框架和模型类型。
- 内置常用建模模板，提高建模效率与降低技术门槛。
- 自定义在线推理逻辑、评估图表。

入参参数列表

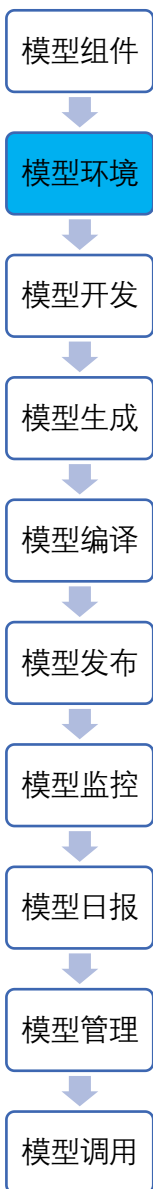
名称	类型	默认值	输入格式
data_x	DataFr...		上游传入
data_y	DataFr...		上游传入
train_size	float	0.7	文本框

函数名: `smyai_train_test_split20`

源码:

```

def smyai_train_test_split2(data_x,data_y,train_size):
    from sklearn.model_selection import train_test_split
    df_train_x,df_test_x, df_train_y, df_test_y=train_test_split(data_x,data_y,train_size=train_size)
    return df_train_x,df_train_y, df_test_x, df_test_y
    
```



- 基于Conda的虚拟环境，管理python和包的不同版本，用于模型开发和部署。
- 可使用docker镜像管理模型环境，实现容器化开发和部署。

名称	<input type="text" value="keras_tf"/>
是否默认	<input type="text" value="否"/>
第三方包信息	<input type="text" value="tensorflow==1.3.0
keras==2.1.2
h5py==2.10.0"/>
服务器列表	<input type="text" value="192.168.2.184"/>
pythonHome	<input type="text" value="/hadoop/bigdata/sai/miniconda3/envs/autom"/>
mlflowHome	<input type="text" value="/hadoop/bigdata/sai/miniconda3/envs/autom"/>
描述	<input type="text" value="newcdh04"/>

模型组件

模型环境

模型开发

模型生成

模型编译

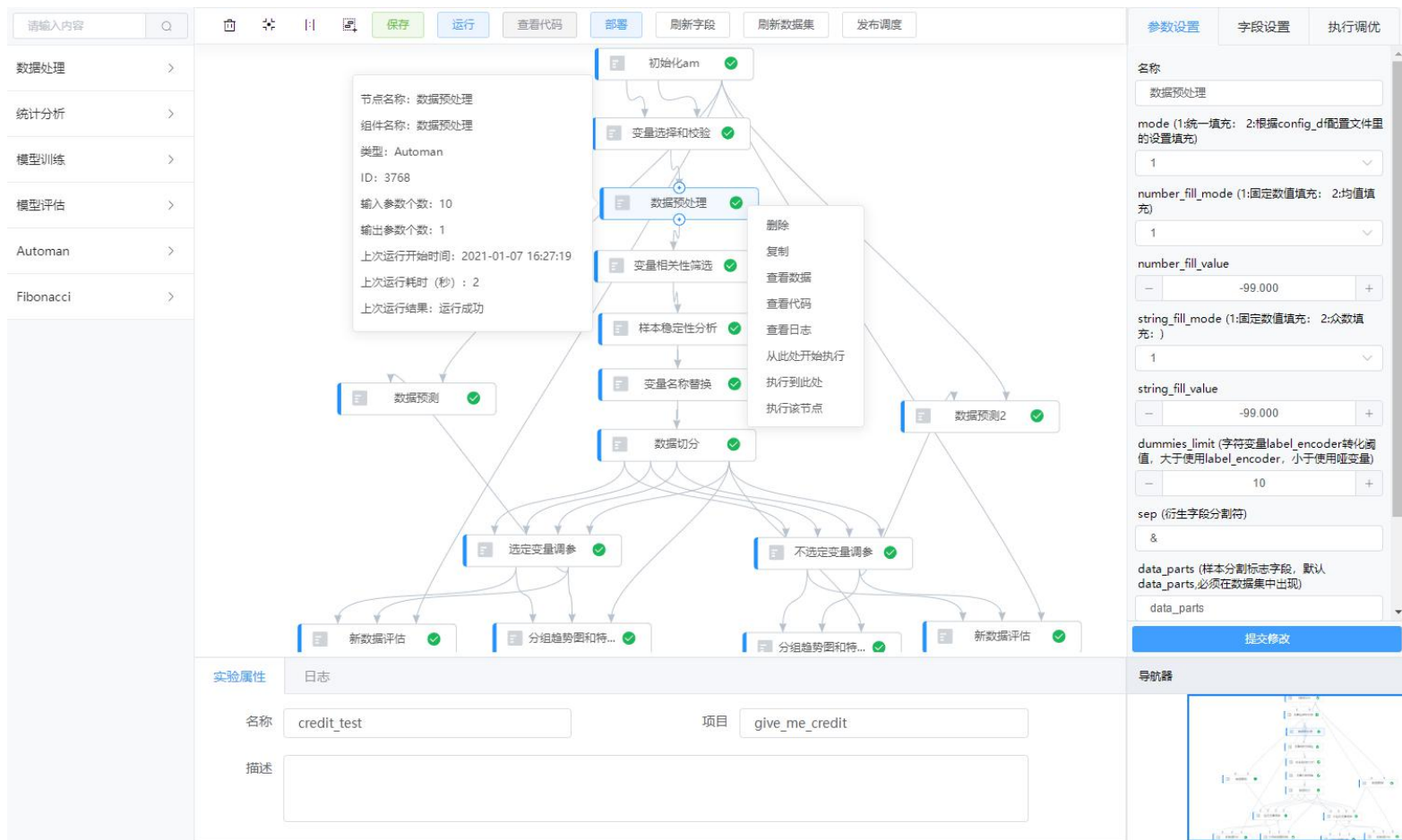
模型发布

模型监控

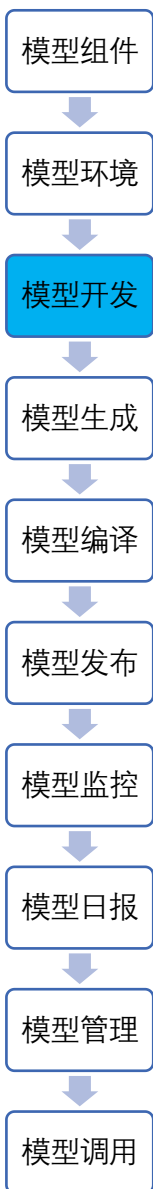
模型日报

模型管理

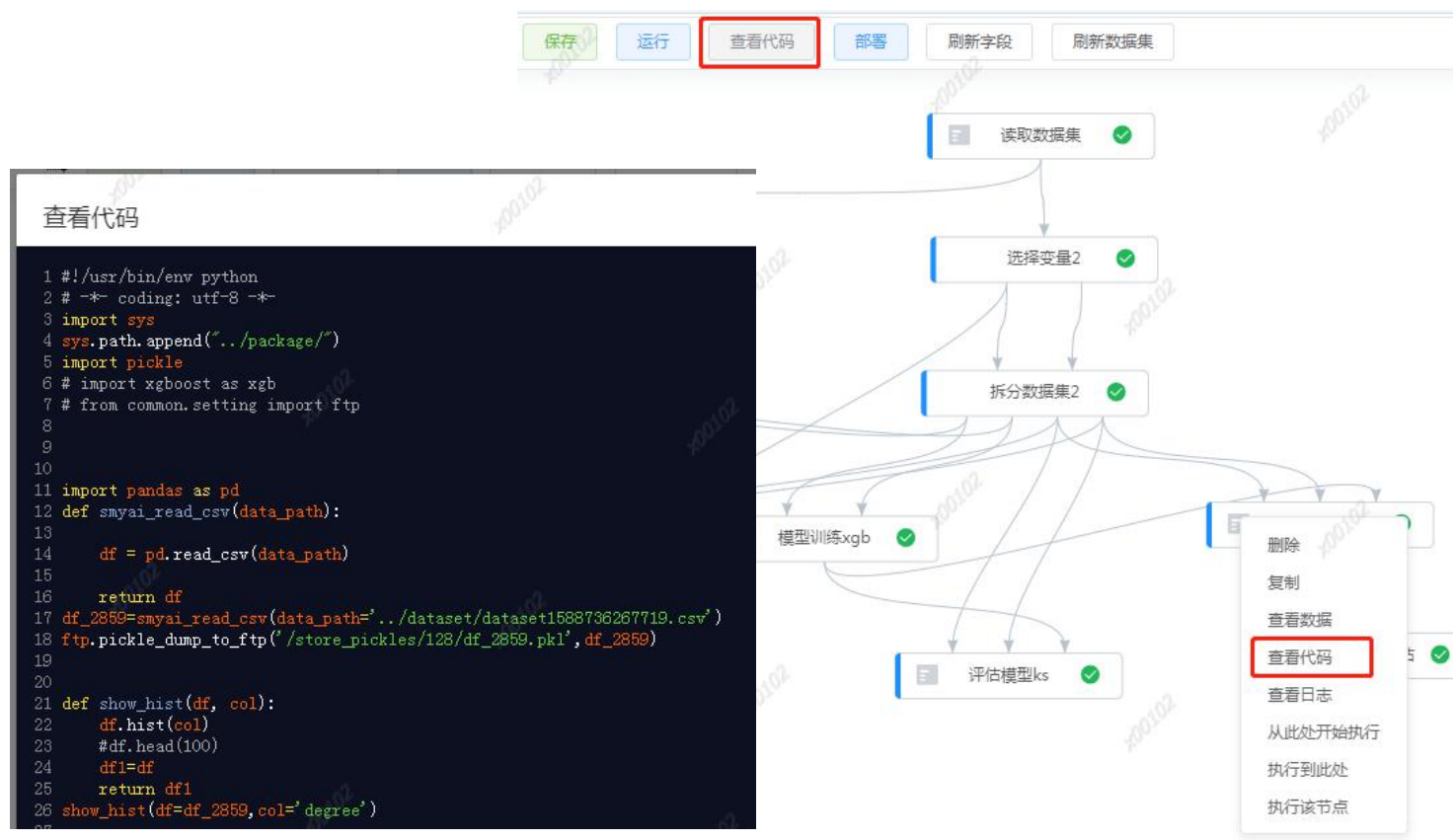
模型调用

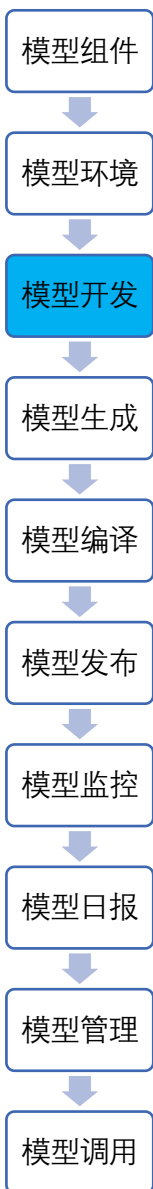


The screenshot displays the GIAC visualization modeling interface. On the left is a sidebar with a search bar and a list of components: 数据处理 (Data Processing), 统计分析 (Statistical Analysis), 模型训练 (Model Training), 模型评估 (Model Evaluation), Automan, and Fibonacci. The main workspace shows a workflow diagram with nodes: 初始化am (Initialize am), 变量选择和校验 (Variable Selection and Validation), 数据预处理 (Data Preprocessing), 变量相关性筛选 (Variable Correlation Selection), 样本稳定性分析 (Sample Stability Analysis), 变量名称替换 (Variable Name Replacement), 数据切分 (Data Splitting), 数据预测 (Data Prediction), 数据预测2 (Data Prediction 2), 选定变量调参 (Selected Variable Parameter Tuning), 不选定变量调参 (Unselected Variable Parameter Tuning), 新数据评估 (New Data Evaluation), and 分组趋势图和特... (Group Trend and Characteristics). A context menu is open over the '数据预处理' node, showing options like 删除 (Delete), 复制 (Copy), 查看数据 (View Data), 查看代码 (View Code), 查看日志 (View Log), 从此处开始执行 (Start Execution Here), 执行到此处 (Execute Here), and 执行该节点 (Execute This Node). A tooltip for the '数据预处理' node provides details: 节点名称: 数据预处理, 组件名称: 数据预处理, 类型: Automan, ID: 3768, 输入参数个数: 10, 输出参数个数: 1, 上次运行开始时间: 2021-01-07 16:27:19, 上次运行耗时 (秒): 2, 上次运行结果: 运行成功. The bottom section contains '实验属性' (Experiment Properties) with fields for 名称 (Name: credit_test), 项目 (Project: give_me_credit), and 描述 (Description). On the right, the '参数设置' (Parameter Settings) panel shows configurations for '数据预处理', including mode, number_fill_mode, number_fill_value, string_fill_mode, string_fill_value, dummies_limit, sep, and data_parts. A '提交修改' (Submit Changes) button is at the bottom of this panel. A small thumbnail of the workflow is visible in the bottom right corner.



- 将实验对应的有向无环图(DAG)编译成代码。
- 用于后台或开发人员调用。



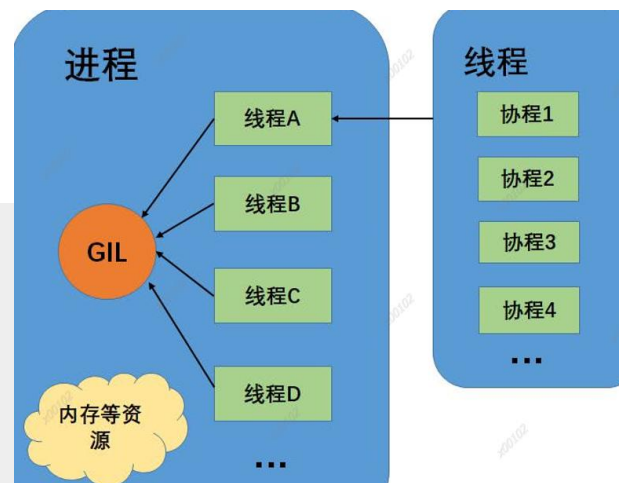


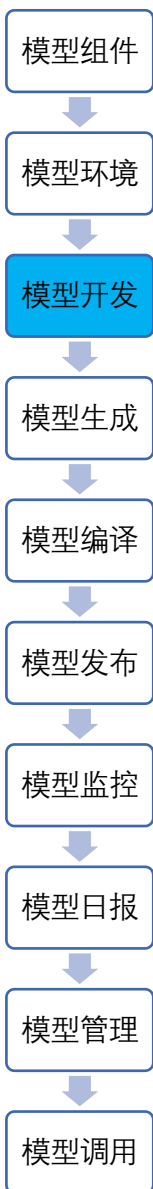
- 前后端交互使用websocket协议，保证消息的实时收发。
- 前端发送运行命令，实时接收运行状态和日志等。
- 后端基于协程的异步编程，相关库都需要基于asyncio，如mysql，http，file。
- 基于CAS算法和数据库锁的并发控制。

```
import asyncio
import websockets
```

```
async def echo(websocket, path):
    async for message in websocket:
        await websocket.send(message)
```

```
start_server = websockets.serve(echo, "localhost", 8765)
```

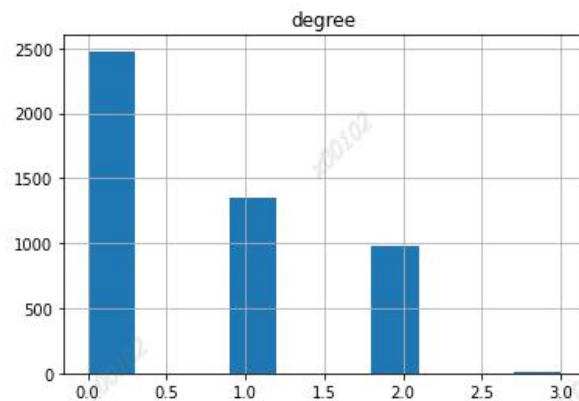


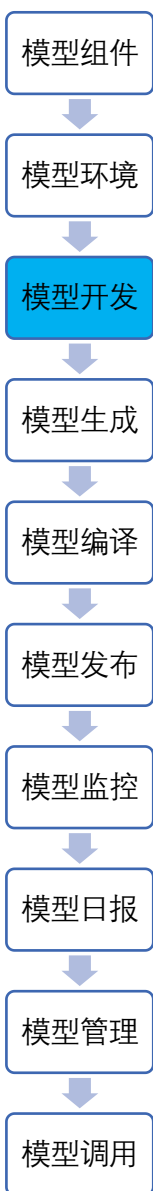


- 基于开源Voila的交互式图表服务，支持模型评估、统计分析、模型日报等图表的生成和展示。
- 通过组件可以自定义任意图表，兼容主流浏览器和Python框架，高度的灵活性和扩展性。

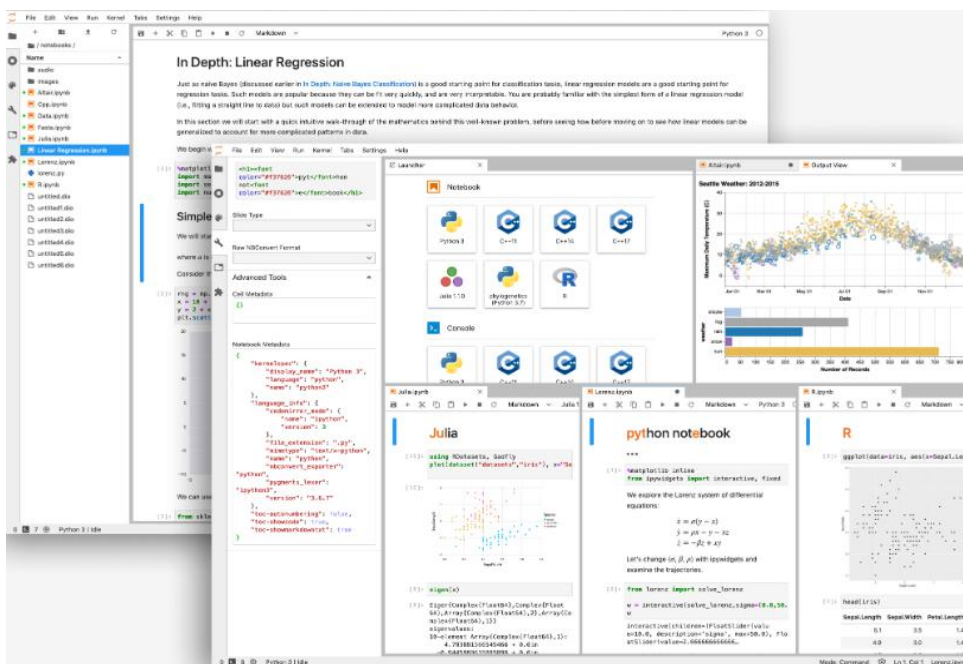


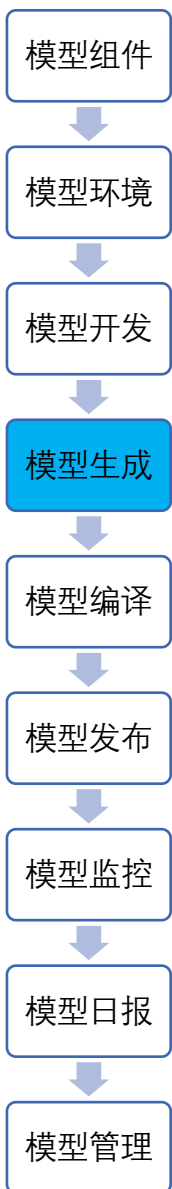
```
1 def show_hist(df, col):
2     df.hist(col)
3     #df.head(100)
4     df1=df
5     return df1
```



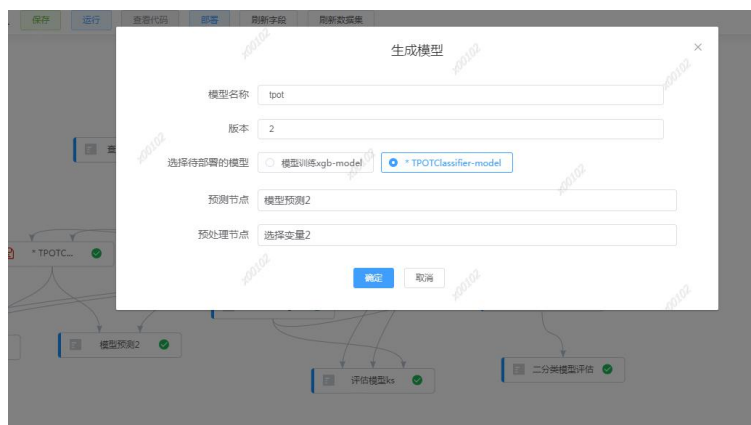


- 基于开源Jupyter的云原生多租户notebook。
- 基于docker镜像在k8s集群中动态创建notebook，按需申请cpu、内存等资源，共享存储。
- 一键部署模型，基于平台托管。





- 选择待部署的模型、预测节点和预处理节点，根据上下文信息，自动生成一条模型记录。
- 模型记录包含了预处理代码、推理代码、模型文件等。



```

def model_preprocess(model_input, ext_obj):
    df_2686=model_input

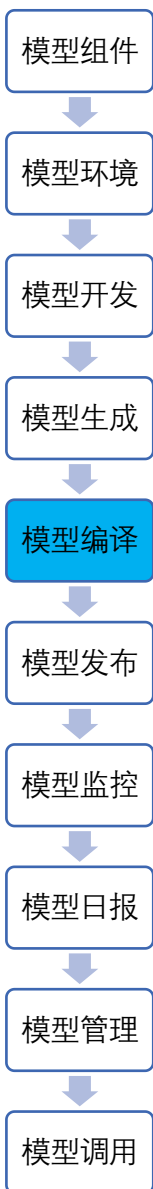
    def sai_select_df_x(df_input, col_x,):
        df_input= df_input[col_x]
        return df_input

    df_2726=sai_select_df_x(df_input=df_2686
        ,col_x=['cust_sex','cust_age','degree'])

    return df_2726
    
```

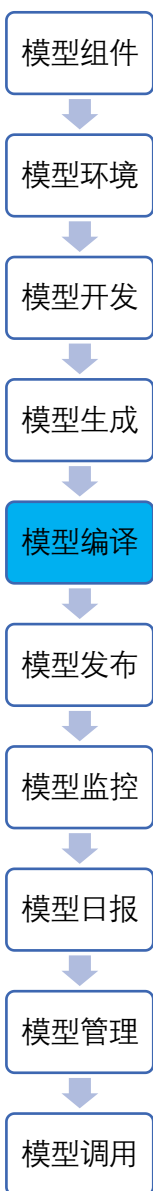
```

1 def model_predict(model, model_input, ext_obj):
2
3     df_2726=model_input
4     model_2694=model
5
6
7     def model_predict_proba(model, df):
8         return model.predict_proba(df)[: ,1]
9
10    predict_proba_2725=model_predict_proba(
11        model=model_2694,df=df_2726)
12
13    return predict_proba_2725
    
```

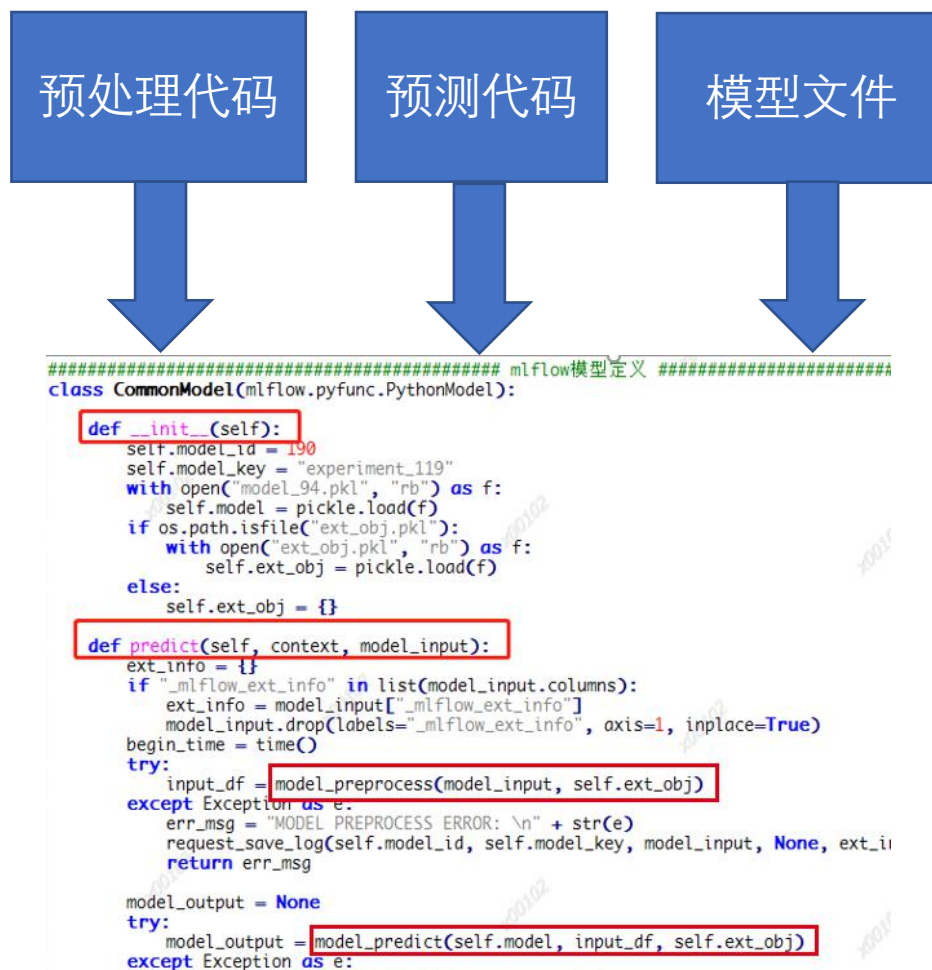


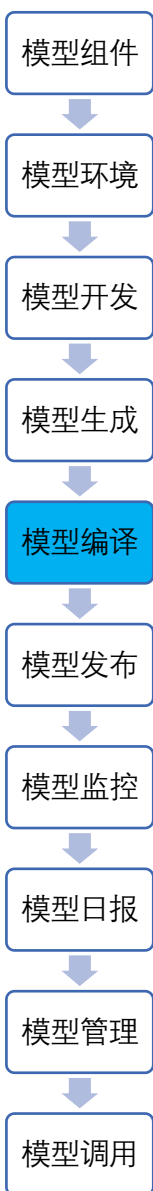
- Databrick公司开源，机器学习全流程管理平台。
- 支持主流开源框架，以及自定义模型。
- 模型跟踪对比，记录实验的参数、代码、结果，通过UI展示。
- 提供了可重复运行的模型包装格式。
- 支持多种在线和离线部署，包括rest api、Azure ML、Amazon SageMaker、Spark UDF、docker镜像等。





- Mlflow的Pyfunc模块提供了自定义模型能力，可以将可视化建模生成的模型封装成Mlflow的格式，跟AI平台无缝整合。
- 支持自定义的数据预处理和在线推理逻辑。





- 默认的Pyfunc模块只支持pickle格式，平台扩展支持了所有主流格式，如tensorflow、Keras等。

```
def load_model(self):
    if not hasattr(self, 'model'):
        _logger.info(f"--CommonModel.load_model: {self.model_path}")
        self.model = sai_load_pickle(self.model_path)
    else:
        _logger.info(f"--CommonModel.model already exists")

if python_model_subpath is:
    raise MlflowException(
        "Python model path"
    )
with open(os.path.join(model_path, python_model_subpath)) as f:
    python_model = cloudpickle.load(f)

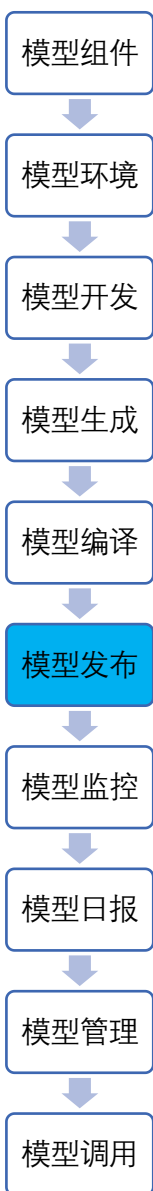
python_model.load_model()
```

- 日期自动转换bug。

```
lib/python3.6/site-packages/mlflow/pyfunc/scoring_server/_init_.py
return pd.read_json(json_input, orient=orient, dtype=False,
                    convert_dates=False)
```

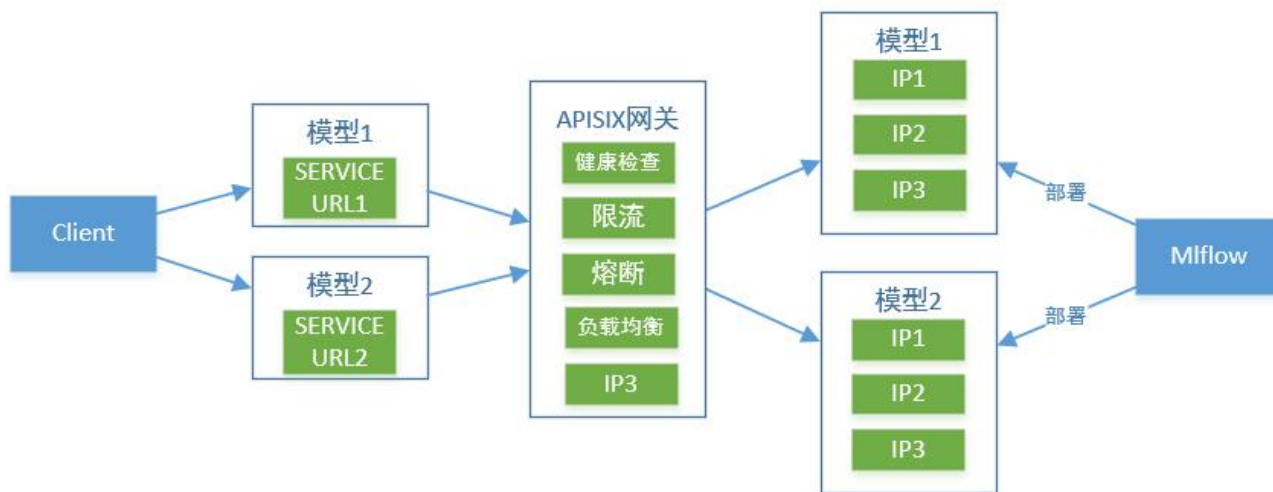
- 自定义的JSONEncoder用于解析np.float32类型，通过o.item()返回了native类型，导致四舍五入失效。

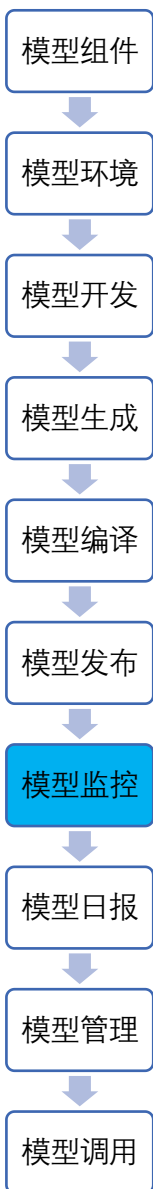
```
class NumpyEncoder(json.JSONEncoder):
    """ Special json encoder for numpy types.
    Note that some numpy types doesn't have native python equivalence,
    hence json.dumps will raise TypeError.
    In this case, you'll need to convert your numpy types into its closest python equivalence.
    """
    def default(self, o): # pylint: disable=E0202
        if isinstance(o, np.generic):
            return np.asscalar(o)
        return json.JSONEncoder.default(self, o)
```



- 平台将Mlflow模型部署为多个REST API服务，通过API网关对外部系统提供服务。
- API网关可以选用nginx或APISIX等，实现负载均衡、高可用、https、健康检查、限流、熔断、灰度发布等功能。
- APISIX具备动态路由和插件热加载，特别适合AI平台的API管理，实现对外URL一键生成。
- 运行mlflow命令进行在线推理服务的启停：

```
mlflow models serve --no-conda --host 0.0.0.0 -p 10028 -w 2 -m
```





- 实时监控模型指标，如平均耗时、qps、失败率、实例状态等，支持自定义指标。
- 满足规则条件会触发告警，根据等级发送邮件、短信等。
- 告警日志查询。

To: Me

标题：[P3 报警]模型 (model_test1) 模型平均耗时大于

模型key：model_test1

告警设备：

监控指标：模型平均耗时

当前值：2000 ms

报警说明：模型平均耗时 > 100 ms

触发时间：2021-05-20 20:34:00

规则名称

是否启用

模型key

规则描述

规则名称

是

选择

模型平均耗时

告警级别

是否全局

通知对象

P1 (邮件+电话+短信)

是

选择

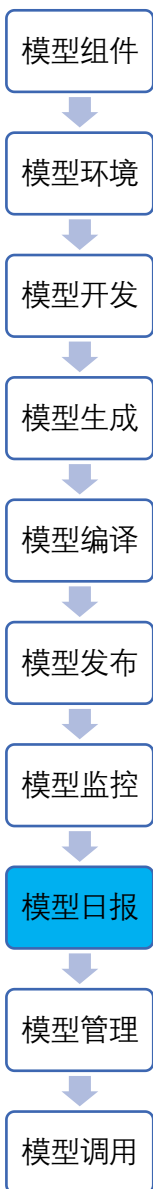
小张
小刘

>

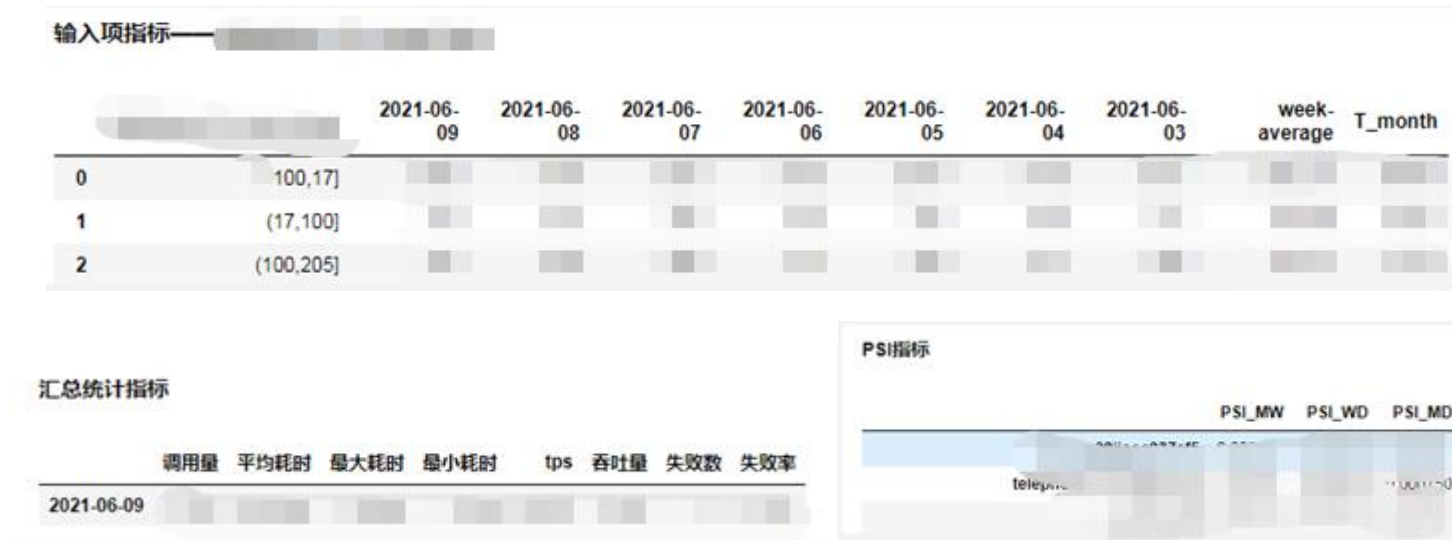
ms

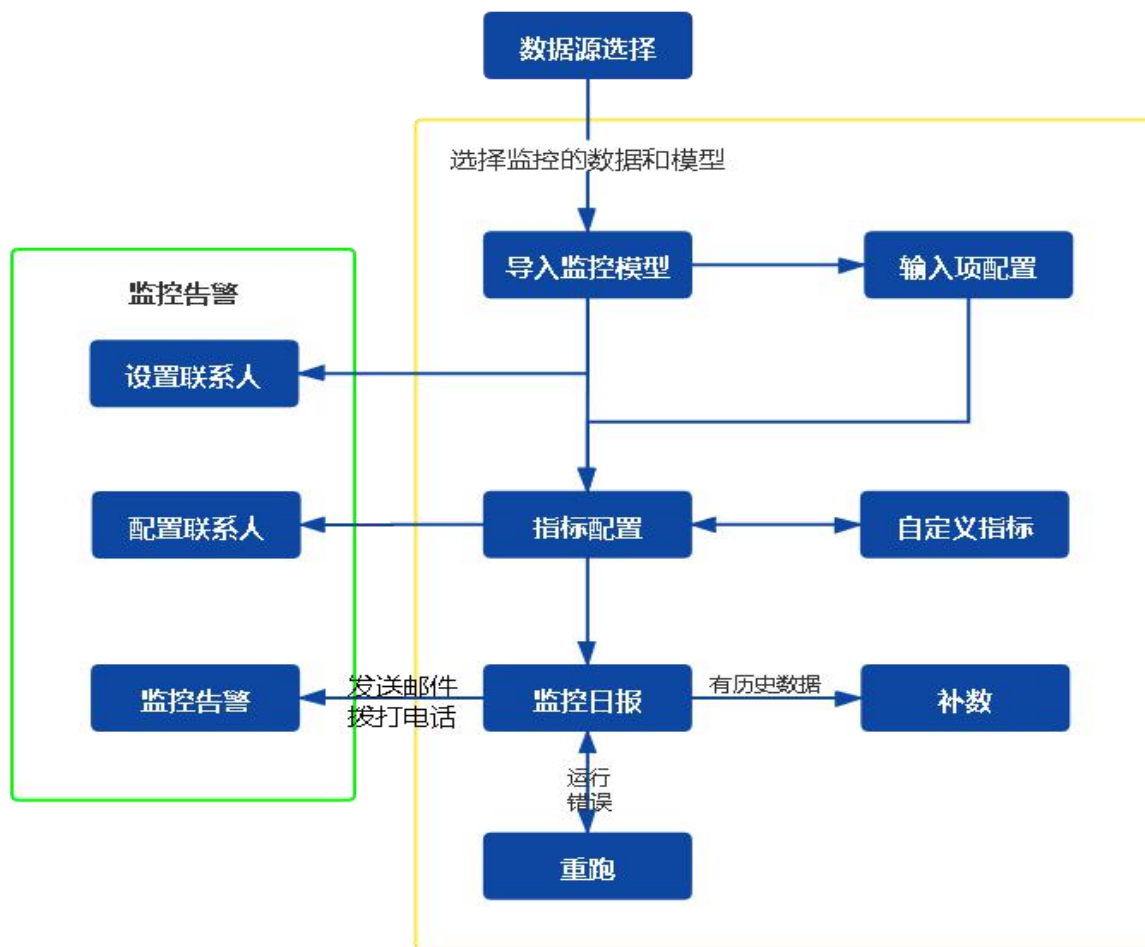
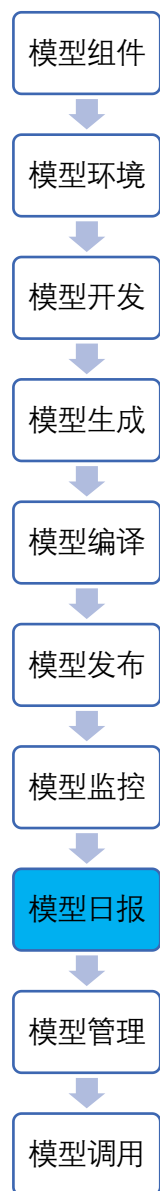
+添加规则

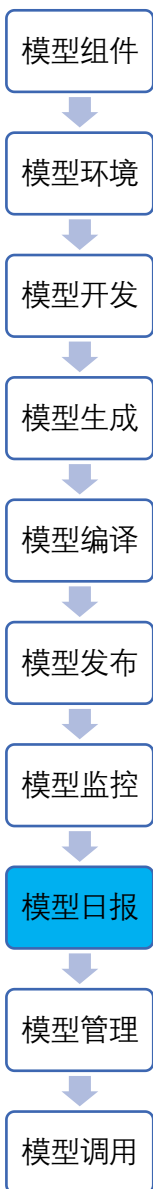
模型平均耗时	>	100ms	删除
模型最大耗时	>	1000ms	删除
模型失败率	>	1%	删除
模型qps	>	300	删除
模型状态	=	异常	删除



- 数据量大：半年区间，亿级数据量
- 计算加速层：统计中间表，单日秒级耗时
- 展示层基于voila，兼容主流python库和浏览器
- 历史日报保存为html文件，方便回溯
- 自动生成分箱数据







- 自定义计算函数和图表函数，抽象接口，标准数据对接，高度灵活性和可扩展性。
- 预置常用指标psi、变量分布、iv等。

指标计算函数： 计算指标，返回指标展示所需要的dataframe以及当天的指标计算结果。

'''

rpt_stats_df: 历史指标(不包含T-1日)计算结果（按天）

model_log_df: T-1日的模型调用的入参记录

bin_df: 输入项列表及其分箱结果

display_df: 指标展示所需要的dataframe信息

rpt_stat_df: 根据model_log_df计算的T-1的指标结果

'''

```
def calculate_metrics(rpt_stats_df, model_log_df, bin_df):
```

```
    pass
```

```
    return display_df, rpt_stat_df
```

指标展示函数： 用于展示指标计算结果

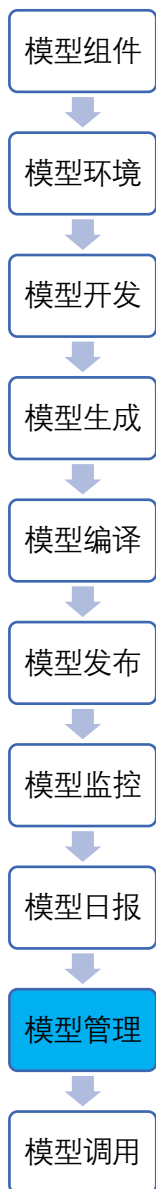
'''

display_df: 指标计算代码计算的结果

'''

```
def display_metrics(display_df):
```

```
    pass
```

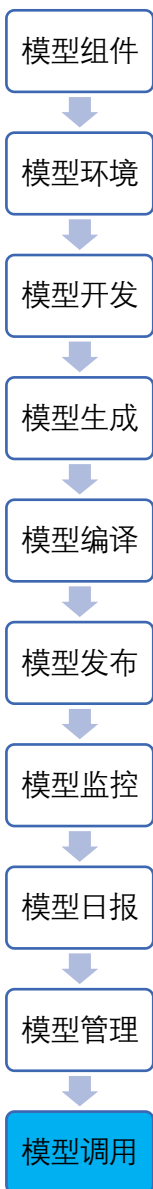


模型管理

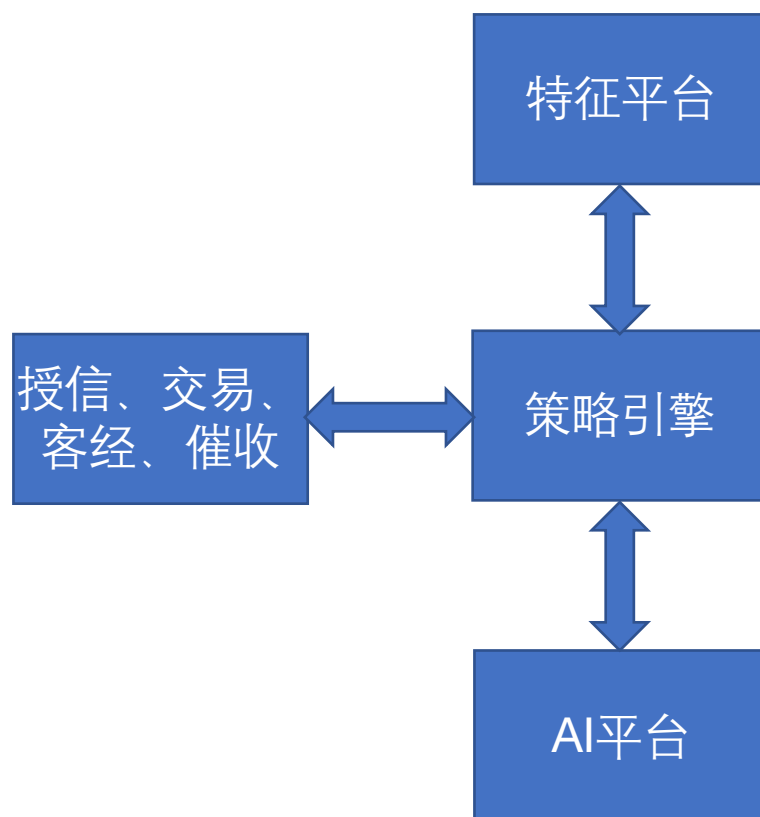
+ 新增

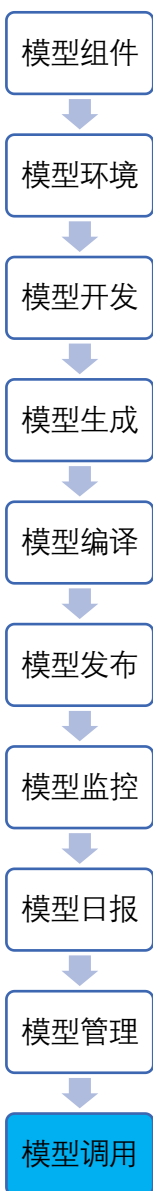
删除

<input type="checkbox"/>	模型key	模型名称	版本	来源	状态	环境	端口	操作
<input type="checkbox"/>	model12	model12	1	用户上传	已发布	am	10002	详情 试跑 发布 启动 停止 跑批1 监控 日报 输入项
<input type="checkbox"/>	测试模型	测试模型	1	用户上传	新建	am	5002	详情 试跑 发布 启动 停止 跑批1 监控 日报 输入项
<input type="checkbox"/>	model99	model99	1	用户上传	停止	am	10001	详情 试跑 发布 启动 停止 跑批1 监控 日报 输入项



- 直接使用特征平台的输入项，避免重复开发
- 策略引擎按需配置模型调用，实现端对端线上化





- 集成DolphinScheduler工作流调度引擎。
- 支持实验pipeline一键发布，以及账号和项目的自动同步。
- 可用于模型的定时训练更新，数据集更新，ETL任务等。
- 后台自动登录，与AI平台无缝整合。



1. 背景介绍

2. 平台功能

3. 架构与实现

4. 未来规划

- 容器化训练和部署。
- 分布式建模。
- AB测试。
- 边缘部署，支持部署到不同设备。
- 离线部署，集成到数仓。
- 在线学习。
- 特征管理与计算。
- 联邦学习。
- 模型解释性。
- 模型跟踪和运行效果对比。

THANKS

徐磊 From 萨摩耶数科
xulei@smyfinancial.com





麦思博(msup)有限公司是一家面向技术型企业的培训咨询机构，携手2000余位中外客座导师，服务于技术团队的能力提升、软件工程效能和产品创新迭代，超过3000余家企业续约学习，是科技领域占有率第1的客座导师品牌，msup以整合全球领先经验实践为己任，为中国产业快速发展提供智库。



高可用架构公众号主要关注互联网架构及高可用、可扩展及高性能领域的知识传播。订阅用户覆盖主流互联网及软件领域系统架构技术从业人员。高可用架构系列社群是一个社区组织，其精神是“分享+交流”，提倡社区的人人参与，同时从社区获得高质量的内容。