# 机器学习模型服务

林智鑫（组长），汪静雅，张值玮，郝浩淼，任玿呈，胡梦箫

## 概述

### 项目说明

本项目仿照 DaaS 平台，提供模型部署上线一站式解决方案。

### 任务分工

前端 - 林智鑫

模型信息读取 - 汪静雅

模型增删查接口 - 张值玮&汪静雅

使用模型预测 - 林智鑫

服务部分 - 林智鑫、郝浩淼

任务部分 - 林智鑫、任玿呈

### 成果

**仓库地址**：https://github.com/Ethkuil/Machine-Learning-Model-Serving

注：答辩时，服务部分和任务部分的docker实现都未完成，使用了多线程来将就实现类似功能，答辩时只是按设想来讲。答辩后两天，这部分仍未完成，因此保留了多线程的实现。

|  |  |
| --- | --- |
| **GitHub账号名** | **姓名** |
| Ethkuil | 林智鑫 |
| Janet731 | 汪静雅 |
| chachaw | 张值玮 |
| haohaomiao | 郝浩淼 |
| rsc-sojung | 任玿呈 |
| matthew-mx | 胡梦箫 |

**在线接口文档**：https://www.apifox.cn/apidoc/shared-333a8de7-4ea7-4c19-b85d-d7c30147af8d

## 实现功能

**基础要求与细节优化：**

* 模型处理
  + - 上传模型，支持 pmml 与 onnx 格式
    - 上传后能查看模型信息，包括输入和目标变量（字段名、类型、取值、维数）

细节优化：

* 完备的表单校验与文件检验
* 支持对已上传的模型排序、筛选
* 上传成功后自动跳转到模型详情页面，并提供返回箭头
* 测试模型
  + 支持通过表单输入数据（表单项可以是文本也可以是文件）
  + 支持直接输入 JSON 格式的数据
  + 提交后预测结果会显示在界面中

细节优化：

* 支持一键清空表单、一键复制结果
* JSON格式数据支持使用base64编码
* 部署模型为网络服务
  + 支持模型服务的暂停，启动，删除
  + 显示当前服务的状态

细节优化：

* 部署成功后自动跳转到服务详情页面，并提供返回箭头
* 可查看当前所有的服务，也可查看与某模型关联的服务
* 使用快速返回接口测试服务
  + 可在前端使用快速返回接口测试部署
* 等待返回接口
  + 接受批量数据，比如 zip包、csv文件，先立即返回任务id，之后可通过任务 id 查找结果

**加分项：**

* 检验模型文件的有效性
* 任务管理查看界面

## 三、技术框架

前后端完全分离，只通过API通信。

前后端各自开发、各自测试（前端使用Mock数据、后端使用接口用例），双方均测试通过后再进行前后端联调。

**前端**：Vue 2 + Element UI。发送请求使用axios。

**后端**：Python Flask。requirements.txt使用pipreqs库生成。

**接口管理**：Apifox

项目的主要内容的结构如下：

### 前端

### 后端

### 接口

## 四、实现思路

### 前端下载文件、后端传递文件

从后端接收二进制流，用其创建1个Blob对象。再为该对象创建1个链接，并设置其download属性为后端提供的文件名（headers['content-disposition']），然后调用click()方法开始下载。

后端使用Flask框架提供的send\_file函数传递文件，并在Content-Disposition头中设置为附件形式、添加参考文件名，并设置'Access-Control-Expose-Headers'头解决跨域问题。

### 前端处理表单中的文件

监测文件的变化，不自动上传，而是先保存添加的文件。

提交表单时，新建1个FormData对象，将需要提交的内容append到其中，这其中就包括文件的内容。

添加完毕后，设置Content-Type为multipart/form-data，发送请求。

### 模型信息读取

* PMML文件

因为缺少相关的库（pypmml只能测试无法得到变量名称、类型等信息），所以我采用了对直接对PMML文件的文本信息进行正则匹配+字符串处理的方式，从DataDictionary中提取所有变量的名称、类型、测量和取值，从MiningField中得到哪些是输入变量哪些是目标变量。

* ONNX文件

调用onnxruntime.InferenceSession()中的get\_inputs()和get\_outputs()函数可以直接得到输入和输出节点的名称、类型和维度。onnx.checker.check\_model(onnx\_model)可以验证输入模型的有效性

### 将请求体数据转化为输入变量的字典

form-data格式不支持复杂的列表等结构，会以字符串形式传递。使用eval()函数可将其转化为Python数据类型。图片数据则经过调整大小、灰度化、归一化最后reshape成模型所需的格式。最后将这些数据合并为1个字典即可。

json格式不支持直接传递二进制数据，本项目采用传递base64编码的办法，并要求采用base64编码的文件须表示为{"type":"base64", "value": base64编码字符串}的格式。后端检测到该格式后，使用base64库的b64decode方法将其解码为二进制数据，再用ByteIO包装（提供read()方法）以便其他库读取处理。

### 批量预测

pypmml库支持直接使用csv文件预测，使用pandas库读入csv文件后调用之即可。

onnx模型，则使用自行实现的读入csv文件转化为字典列表的函数，预测后将结果列表再自行写入新建的csv文件。处理zip数据集思路类似，只是使用了zipfile包实现在不解压的前提下读取数据。