

# 智能体应用文档

## 一、基础信息

- 团队名称: AutoMaticLatex
- 作品名称: Amala

## 作品简介

Amala (**AutoMatic Latex**)

- **核心目标:** 实现自动从图片中提取latex公式, 同时可选地提供公式讲解
- **应用场景:** 撰写paper时, 避免手动输入大量繁琐的latex公式
- **目标用户:** 广大科研工作者, 有使用latex公式需求的用户

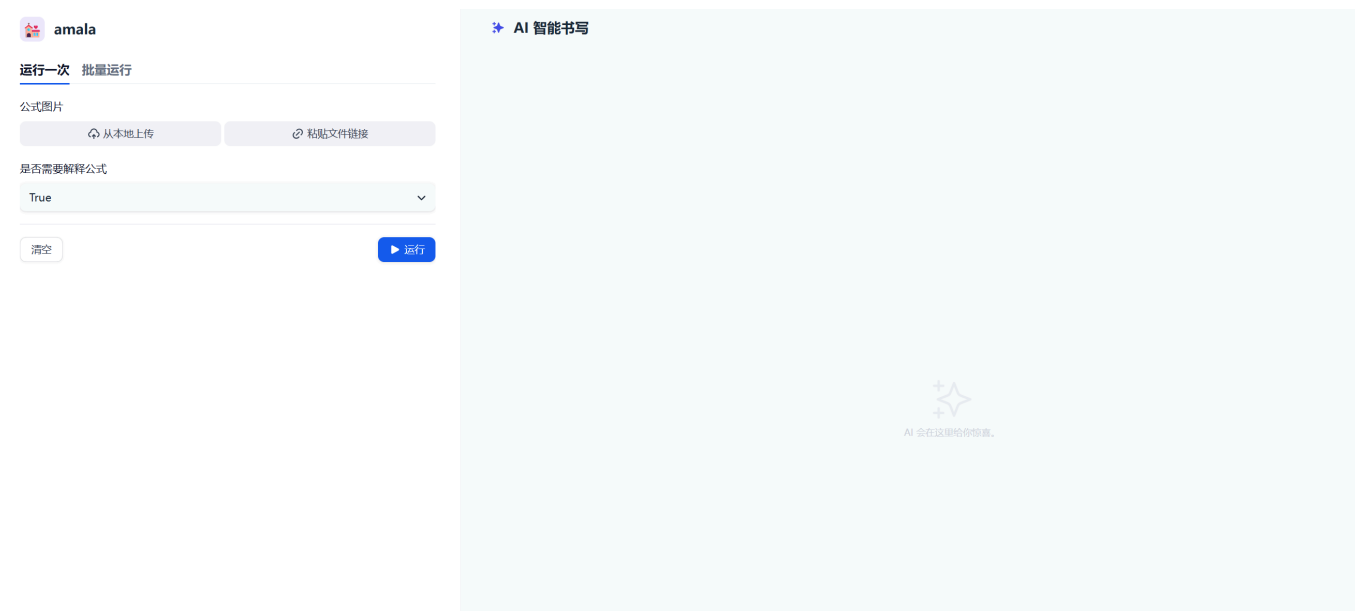
利用ocr技术, 通过ViT, 将图片转换为latex公式, 参考开源的LaTeX-OCR。

输入含有公式的图片, 通过api调用个人已经部署在hugging face的LaTeX-OCR工具, 输出latex公式, 可以直接复制使用。

如果有需要, 还可以调用大模型对公式进行解释。

## 二、核心内容

### 1. 智能体用户界面



左侧用户可以添加目标公式图片、选择是否需要大模型解释，点击运行后，右侧会根据用户的输入来输出结果。

## 2. 操作流程

### 1. 上传图片



amala

运行一次 批量运行

公式图片

🔄 从本地上传

🔗 粘贴文件链接

是否需要解释公式

True



清空

▶ 运行

- 用户可以选择从本地上传，或者直接从剪贴板中粘贴图片文件链接
2. 选择是否需要解释，默认是False
  3. 运行

## 3. 预期成果展示

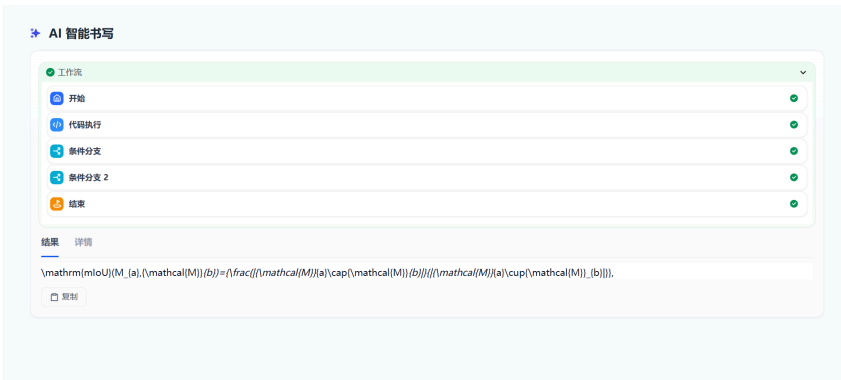
$$\text{mIoU}(\mathcal{M}_a, \mathcal{M}_b) = \frac{|\mathcal{M}_a \cap \mathcal{M}_b|}{|\mathcal{M}_a \cup \mathcal{M}_b|},$$

- 输入的公式图片：
- 输出结果：

- 选择需要解释公式：



- 选择不需要解释公式：



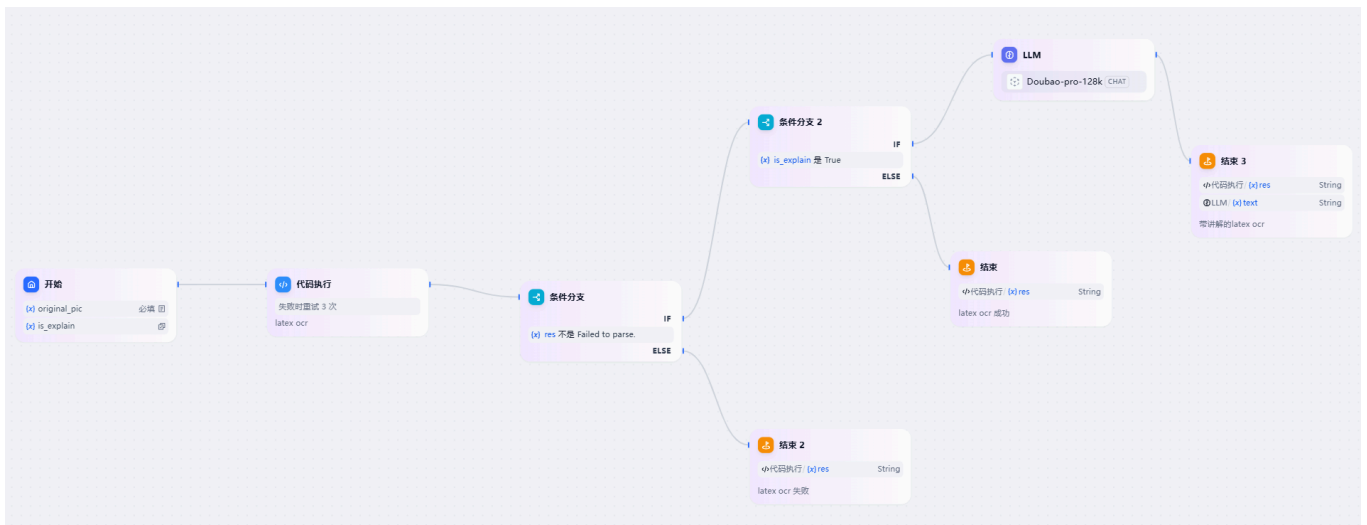
- 将模型输出的latex公式进行渲染：

$$\mathrm{mIoU}(M_a, M_b) = \frac{|M_a \cap M_b|}{|M_a \cup M_b|}$$

$$\mathrm{mIoU}(M_a, M_b) = \frac{|M_a \cap M_b|}{|M_a \cup M_b|}$$

发现成功对原公式图片进行了识别

## 4. 智能体实现方法



1. 利用平台提供的输入接口，添加图片输入和选择是否需要，公式图片输入必填

### 输入字段

(x) original\_pic · 公式图片

必填

(x) is\_explain · 是否需要解释公式



2. 代码执行是核心模块，用于向个人部署在hugging face上的space发送post请求，调用latex-ocr模型，然后解析响应，获取返回值 res

latex ocr

输入变量

+

img\_path

开始 / original\_pic.url String



PYTHON3



```
1 import requests
2 import re
3
4 def main(img_path):
5     # Define the URL and the data for the POST request
6     url = 'https://seanxunx-amala.hf.space/gradio_api/call/predict'
7     data = {
8         "data": [
9             {"path": f"{img_path}", "meta": {"_type": "gradio.FileData"}}
10        ]
11    }
12
13    res = "Failed to parse."
14
15    # Send a POST request to get the EVENT_ID
16    response = requests.post(url, json=data, headers={"Content-Type":
17        "application/json"})
18
19    if response.status_code == 200:
20        res_data = response.json()
21        event_id = res_data['event_id']
22        if event_id:
23            get_url = f"https://seanxunx-amala.hf.space/gradio_api/call/
24                predict/{event_id}"
25            get_res = requests.get(get_url, stream=True)
26            if get_res:
```

输出变量

+

res

String



- 根据 res 的值，和用户选择的是否需要解释公式 is\_explain，进行条件分支判断，进行调用大模型解释等后续工作

