一、部署：

paddleocr接口1：

cd 进入PaddleOCR-2.9，随后启动python app.py

在app.py中修改端口号，在ocr.py中修改接口逻辑



paddleocr接口2：

cd 进入PaddleOCR-2.9，随后安装对应模块hub install ./deploy/hubserving/ocr\_system，之后输入hub serving start -c deploy/hubserving/ocr\_system/config.json启动

在./deploy/hubserving/ocr\_system/config.json中修改启动的端口号

在./deploy/hubserving/ocr\_system/params.py中修改调用的模型



二、测试：

Paddleocr接口1：

postman调试



地址http://180.169.131.147:8019/ocr/process，选择POST方法，body选择form-data，添加file字段并上传图片。

crul调试

curl -X POST http://180.169.131.147:8019/ocr/process -F "file=@（图片地址）"

返回json数据类型：

{

"content": [

{

"confidence": 1.0,

"text": "备注："

},

{

"confidence": 1.0,

"text": "门诊"

},

{

"confidence": 0.981,

"text": "医瞩名称"

}

……

……

],

"message": "File processed successfully",

"uuid": "18e959f3-476f-477a-8912-ce5e20767c95"

}

Paddleocr接口2：

postman调试



地址http://180.169.131.147:8021/predict/ocr\_system，选择POST方法，Headers添加Content-Type application/json，body为raw的json格式{"images": ["……"]}（省略号换成图片的base64编码形式）

crul调试

curl -H "Content-Type:application/json" -X POST --data "{\"images\": [\"……\"]}" http://180.169.131.147:8021/predict/ocr\_system（省略号换成图片的base64编码形式）

返回json数据类型：

{

"msg": "",

"results": [

[

{

"confidence": 0.9999859929084778,

"text": "备注：",

"text\_region": [[26,32],[57,32],[57,47],[26,47]]

},

{

"confidence": 0.9999996423721313,

"text": "门诊",

"text\_region": [[485,34],[510,34],[510,49],[485,49]]

}

……

……

]

],

"status": "000"

}