关于汽车评分标签获取接口安装说明

1. 环境说明：

Python3.x+flask

因为使用到flask框架，所以在生产环境中需要安装uwsgi网关，这样才能保证服务的稳定性，这部分内容的安装调试可以请教运维的同事；

1. 软件的获取：

# git clone <https://github.com/xiaobo0903/tjmodel>

仓库中有三个目录，web1目录下就是程序使用目录；其它两个目录web、train可以删除；

运行的主web程序是: tj\_predict\_api.py

python3 -u tj\_predict\_api.py

启动后的接口

<http://domain:port/tj_predict/get_tag>

使用post方法提交；

1. 提交数据格式说明（method=post）：

|  |
| --- |
| {      "id": "111111111122222222223333333",      "debug":false,      "detail": [{          "bg": 1990,          "onebest": "行没问题，就如果说您订不了的话，后期我再联系，你，有活动的话我再通知您"      },      {          "bg": 6640,          "onebest": "然后您也可以关注咱们魔比商城，要是价格有变动的话，呢您在家里拿手机一看就一目了然了"      }]  } |

具体的提交格式可参考call\_test.py中的代码；

1. 返回格式说明（json）：

|  |
| --- |
| 成功返回  {      "code": 0,      "detail": [          {              "bg": 6640,              "match": [                  {                      "regex": "关注咱们魔比",                      "tag": "感兴趣车辆"                  }              ],              "mflag": true,              "onebest": "然后您也可以关注咱们魔比商城，要是价格有变动的话，呢您在家里拿手机一看就一目了然了"          }      ],      "id": "111111111122222222223333333"  }  请求包含 debug为true时,在返回值上会携带更多的信息如begin-time 、use-time等  {      "begin-time": "2021-03-08 17:25:13.477357",      "code": 0,      "detail": [          {              "bg": 1990,              "mflag": false,              "onebest": "行没问题，就如果说您订不了的话，后期我再联系，你，有活动的话我再通知您"          },          {              "bg": 6640,              "match": [                  {                      "regex": "关注咱们魔比",                      "tag": "感兴趣车辆"                  }              ],              "mflag": true,              "onebest": "然后您也可以关注咱们魔比商城，要是价格有变动的话，呢您在家里拿手机一看就一目了然了"          }      ],      "end-time": "2021-03-08 17:25:13.478959",      "id": "111111111122222222223333333",      "use-time": "0:00:00.001602秒"  }  异常请求时的返回值  {      "code": 1,      "message": "程序运行期异常, 请稍后再试"  } |