# AutoCV模型使用说明书

## 权限控制编译说明

### 1.1编译依赖

1.目前的模型包整体上只在CentOS 7.5上进行过运行测试，不能完全保证在其他Linux发行版或者非Linux操作系统正常运行。

2.程序使用Python执行，版本建议使用3.6及以上，并且使用了部分Python未预安装的模块，可以使用下面命令安装：

|  |
| --- |
| pip install cython |

### 1.2编译方法

#### 文件说明

编译前分发包中的文件列表如下：

|  |
| --- |
| ├── autocv\_classifier\_v\_0\_1.tar  ├── build\_python.sh  ├── main.py  ├── predict\_url.py  ├── run\_docker.py  └── setup.py |

run\_docker.py：镜像加载程序模块，主要功能是用户权限控制和启动docker,主要给main.py提供接口,该文件是我们进行权限控制编译的对象；

main.py：主程序文件，用来启动训练、测试、推理以及模型服务化；

predict\_url.py：通过调用模型服务的URL接口实现推理，该文件可以提供给用户，方便用户进行二次开发使用；

autocv\_classifier\_v\_0\_1.tar：模型程序的docker镜像文件压缩包（文件名会根据具体的发布版本而变动）；

setup.py：Python编译文件；

build\_python.sh：编译执行脚本文件，使用setup.py编译run\_docekr.py文件。

#### 编译过程

1. 使用编辑软件打开用户权限控制控制文件run\_docker.py，文件中包含如下代码段：



1. 根据不同的权限控制需求，需要修改不同的字段，AUTH\_CONTROL\_TYPES字段说明了目前支持的权限控制类型，AUTH\_CONTROL用来打开或关闭权限控制。
2. 如果只需要控制时间权限，则将AUTH\_CONTROL设置为True，然后将AUTH\_CONTROL\_TYPE设置为time，将AUTH\_START\_TIME设置为当前时间或者预计程序开始使用的时间（时间格式必须与上图中一致），将AUTH\_TIME设置为需要授权的时间天数，如上图中的100天；
3. 如果只需要控制服务器主机权限，则将AUTH\_CONTROL设置为True，将AUTH\_CONTROL\_TYPE设置为host，然后将 HOST\_MAC\_ADDR设置为**用户提供**的服务器MAC地址，注意该MAC地址必须是物理网卡MAC地址，而不是loop或docker虚拟的网卡地址；
4. 如果需要同时控制时间和主机权限，则将AUTH\_CONTROL设置为True，将AUTH\_CONTROL\_TYPE设置为time&host，然后同时设置这两种权限相关的字段，具体参考上面a和b的内容。
5. 运行如下命令便可完成编译，注意：该命令会自动删除无需提供给用户的文件，所以在执行前建议备份，或者编辑build\_python.sh文件，禁用掉删除操作，并在编译完成后手动删除对应的文件：

|  |
| --- |
| bash build\_python.sh |

完成编译后，在用户主机上安装时的打包文件只需要包含如下图所示的四个文件即可。

|  |
| --- |
| ├── autocv\_classifier\_v\_0\_1.tar  ├── main.py  ├── predict\_url.py  └── run\_docker.so |

## 2.系统安装说明

### 2.1依赖

1. 目前的模型包整体上只在CentOS 7.5上进行过运行测试，不能完全保证在其他Linux发行版或者非Linux操作系统正常运行。

2．本次提供的模型程序是依赖Docker容器运行的，所以宿主机上需运行Docker client和Docker server，并需要安装nvidia-docker插件，安装方法见程序主页（https://github.com/NVIDIA/nvidia-docker）。注意：nvidia-docker要求Docker版本在19.03及以上。

3. 本程序需要依赖GPU环境运行，请提前安装Nvidia GPU驱动程序，目前程序只在Nvidia 10180Ti上测试通过，显卡驱动版本为410.78。

4. 程序使用Python执行，Python版本建议在3.6及以上，并且使用了部分Python未预安装的模块，使用下面命令安装：

|  |
| --- |
| pip install ntplib requests |

### 2.2安装

## 程序包结构说明

下面是程序包解压后的目录结果:

|  |
| --- |
| [yang@m7-model-gpu15]$ tree  ├── autocv\_classifier\_v\_0\_1.tar  ├── main.py  ├── predict\_url.py  └── run\_docker.so |

下面是以上文件的详细说明:

run\_docker.so：镜像加载程序库，用户权限控制和启动docker,主要给main.py文件提供调用；

main.py：主程序文件，用来启动训练、测试、推理以及模型服务化；

predict\_url.py：通过调用模型服务的URL接口实现推理，该文件可以提供给用户，方便用户进行二次开发使用；

autocv\_classifier\_v\_0\_1.tar: 该文件是模型程序的镜像文件压缩包（具体文件名会根据具体的发布版本而变动），使用下面命令将该镜像包上传到本地镜像仓库:

|  |
| --- |
| docker load -i autocv\_classifier\_v\_0\_1.tar |

## 服务启动

使用以下命令，启动服务：

|  |
| --- |
| python main.py start\_server --model\_path `pwd`/records --port 5000 --visible\_devices\_list 5,6 |

其中，--port、--model\_path和--visible\_devices\_list指定程序依赖的参数，具体含义和其他参数可以使用如下命令查看：

|  |
| --- |
| python main.py start\_server --help |