

# CLIP模型加载与转换

问题描述：








本地要根据.bin文件转换生成.pth文件，供后续使用

```
1 pretrained = 'laion2B-s34B-b88K'
2 model, _, _ = create_model_and_transforms(name_mapping[args.model],
    pretrained=pretrained)
```

加载CLIP权重.bin模型错误>>网络连接错误>>无法下载/laion/CLIP-ViT-B-16-laion2B-s34B-b88K/resolve/main/open\_clip\_pytorch\_model.bin

解决方法：

1.从url链接获取位置，去huggingface网站上找到准确的。下载必要的几个文件（.bin必要）

 merges	2023/12/22 16:08	文本文档	513 KB
 open_clip_config	2023/12/22 16:08	JSON 源文件	1 KB
 open_clip_pytorch_model.bin	2023/12/22 15:33	BIN 文件	584,502 KB
 special_tokens_map	2023/12/22 16:08	JSON 源文件	1 KB
 tokenizer	2023/12/22 16:08	JSON 源文件	2,172 KB
 tokenizer_config	2023/12/22 16:08	JSON 源文件	1 KB
 vocab	2023/12/22 16:08	JSON 源文件	843 KB

2.把这一堆，最好放在/home/ybk/.cache/huggingface/hub/指定文件夹，虽然感觉没啥用。一定要放在一个找得到的文件夹里

3.改pretrained的指定路径，指定到.bin文件

```
1 pretrained='/home/ybk/.cache/huggingface/hub/models--laion--CLIP-ViT-B-16-laion2B-s34B-b88K/open_clip_pytorch_model.bin'#12.22指定路径
2 model, _, _ = create_model_and_transforms(name_mapping[args.model],
    pretrained=pretrained)
```

4.ok

---

## CLIP模型加载与本地读取

问题描述：以CLIPDINO为例，其中调用了MaskCLIP。MaskCLIP需要使用CLIP模型

## 1.原代码/逻辑

```
1 self.tokenizer = get_tokenizer(f'{model_prefix}/{clip_model}')
2 model, _ = create_model_from_pretrained(f'{model_prefix}/{clip_model}')
```

2.观察报错信息--本来要去huggingface上找 url:

```
1 laion/CLIP-ViT-B-16-laion2B-s34B-b88K
```

我要去hg网站上手动把这里的文件全都下载下来，把这一堆，最好放在/home/ybk/.cache/huggingface/hub/指定文件夹

### 3.1对于tokenizer

```
1 def get_tokenizer(...)#获取tokenizer
2     if model_name.startswith(HF_HUB_PREFIX):#是huggingface开头，从hf上下载config
3         model_name = model_name[len(HF_HUB_PREFIX):]#laion/CLIP-ViT-B-16-laion2B-s34B-b88K(把hf-hub: 删掉)
4         try:
5             config = _get_hf_config(model_name)['model_cfg']
6             ...
7
8 def _get_hf_config(...):
9     #从huggingface上下载config
10    #此处需要下载的是'open_clip_config.json'
11    config_path = download_pretrained_from_hf(model_id,
12        filename='open_clip_config.json', cache_dir=cache_dir)#laion/CLIP-ViT-B-16-laion2B-s34B-b88K#从hf上下载
13    with open(config_path, 'r', encoding='utf-8') as f:
14        config = json.load(f)
15    return config
```

### 3.2直接指定本地的config\_path，注意是open\_clip\_config.json的路径

```
1 def _get_hf_config(model_id, cache_dir=None):
```

```

2     config_path='/home/ybk/.cache/huggingface/hub/models--laion--CLIP-ViT-B-16-laion2B-
    s34B-b88K/open_clip_config.json'
3     with open(config_path, 'r', encoding='utf-8') as f:
4         config = json.load(f)
5     return config

```

#### 4.1对于model

```

1 def create_model_from_pretrained(...)
2     ...
3     model = create_model()
4     ...
5
6 def create_model(...)
7     if has_hf_hub_prefix:#是huggingface开头
8         model_id = model_name[len(HF_HUB_PREFIX):]
9         checkpoint_path = download_pretrained_from_hf(model_id, cache_dir=cache_dir)
10        config = _get_hf_config(model_id, cache_dir)
11        ...

```

#### 4.2把checkpoint\_path直接指定本地的模型.bin， 注意是open\_clip\_pytorch\_model.bin的路径

```

1 checkpoint_path = '/home/ybk/.cache/huggingface/hub/models--laion--CLIP-ViT-B-16-
    laion2B-s34B-b88K/open_clip_pytorch_model.bin'
2 config = _get_hf_config(model_id, cache_dir)#前面已经改过了， 依旧是open_clip_config.json

```