

CSDN

首页 博客 程序员学院 下载 论坛 问答 代码 直播 电子书

最新评论

C++部署Pytorch (Libtorch) 出现问题、...
zsffuture: 可以参考我的最新博客https://blog.csdn.net/weixin_42398658/article/details/109138805 ...
C++部署Pytorch (Libtorch) 出现问题、...
qq_34543664: 我将模型路径作为参数传入进行编译报错，也就是说我并未传入模型 ...
【论文review+总结】屏幕内容图像质量...
codertalk: very good
C++部署Pytorch (Libtorch) 出现问题、...
zsffuture: 我的找bug过程不仅仅是上面提到的，我还遇到了可以进行检测，也 ...
C++部署Pytorch (Libtorch) 出现问题、...
zsffuture: 行

最新文章

Opencv显示RGB图像时出现绿色竖线的解决方法
在C++部署Pytorch (Libtorch) 模型的方法总结 (Win10+VS2017)
让Visual Studio (VS) 链接静态库 (lib) 及动态库 (dll) 文件的方法汇总

2020年 7篇 2019年 1篇

原创

Z的三次方

2020-10-17 22:35:52

1540

收藏 8

版权

分类专栏: [bug日志](#) [libtorch](#) 文章标签: [c++](#) [pytorch](#) [api](#) [接口](#) [bug](#)

错误汇总

1、由于找不到xxx.dll,无法继续执行代码，重新安装程序可能会解决此问题
(1) 由于找不到c10.dll (或其他libtorch/lib中的.dll动态库) ,无法继续执行代码
(2) 由于找不到VCRUNTIME 140_1D.dll,无法继续执行代码

2、LINK : fatal error LNK1104: cannot open file 'torch-NOTFOUND.obj' (torch-NOTFOUND.obj无法找到)

3、error C2440: “初始化”: 无法从“torch::jit::script::Module”转换为“std::shared_ptr”

4、无法定位程序输入点cudnnSetCTCLossDescriptorEx于动态链接库xxx.dll上

5、有未经处理的异常: Microsoft C++异常: c10::Error, 位于内存位置xxx处

6、引发异常: 0xC0000005:读取位置0xFFFFFFFFFFFFFFFF时发生访问冲突

7、error : c2872 std 不明确的符号

1、由于找不到xxx.dll,无法继续执行代码，重新安装程序可能会解决此问题

在运行VS时，有时会出现***“由于找不到xxx.dll,无法继续执行代码”***的报错。

这是由于当前执行程序无法找到这个动态库。这种情况可以大致分为两种以下两种：

(1) 由于找不到c10.dll (或其他libtorch/lib中的.dll动态库) ,无法继续执行代码

这是因为执行程序无法找到libtorch/lib下的这些外部动态库，这些库并不是没有，而是程序无法找到他。最简单粗暴的解决方法就是将你之前下载的libtorch库的libtorch/lib文件夹中的对应.dll文件复制到你目前工程的Example\build\Release (或debug，取决于你是采用release还是debug模式) 里，（说简单点就是和你当前工程编译后的可执行文件放在一起），缺什么补什么。

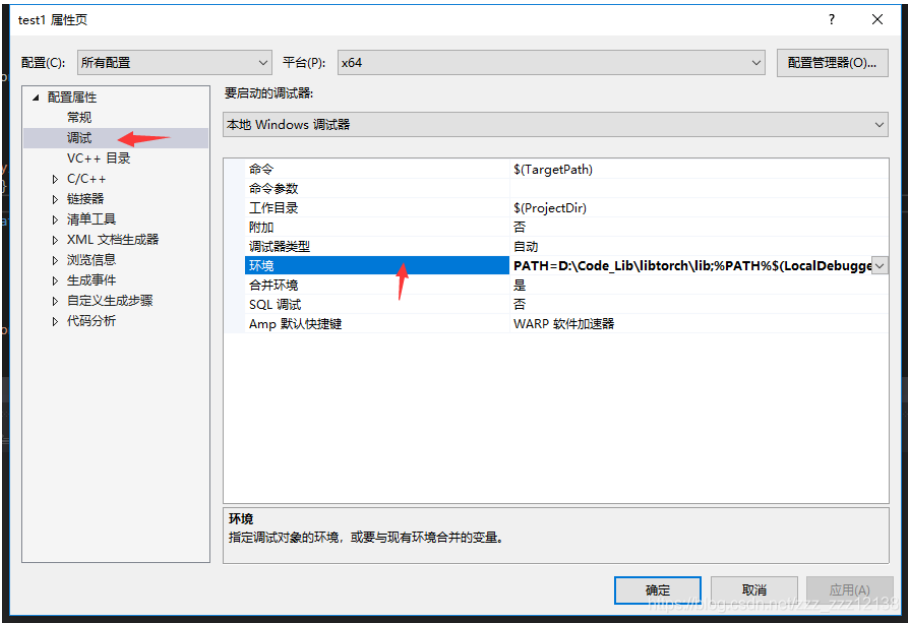
名称	修改日期	类型	大小
c10.dll	2019/5/1 9:25	应用程序扩展	340 KB
c10_cuda.dll	2019/5/1 9:25	应用程序扩展	138 KB
caffe2.dll	2019/5/1 9:42	应用程序扩展	135,537 KB
caffe2_gpu.dll	2019/5/1 10:33	应用程序扩展	416,542 KB
example-app.exe	2019/6/14 23:45	应用程序	73 KB
example-app.exp	2019/6/14 23:45	Exports Library ...	18 KB
example-app.lib	2019/6/14 23:45	Object File Library	31 KB
nvToolsExt64_1.dll	2014/5/30 1:50	应用程序扩展	47 KB
torch.dll	2019/5/1 10:37	应用程序扩展	29,731 KB

另外，我看网上还有一种做法，就是把libtorch/lib中的所有dll放到libtorch/bin中，然后把libtorch/bin加到环境变量的path中。这样让程序安装Path中的路径自己找到对应的动态库，我觉得这个方法应该是最科学有效的。但是我这么做了之后仍然出现了无法找到里面某些.dll文件的问题，所以不知道这个方法的正确操作方式应该是什么。

补充：目前找到了一个有效且简易的方法，就是在属性界面的 调试=>环境 里添加libtorch动态库的路径：

1 | PATH=D:\Code_Lib\libtorch\lib;%PATH%

https://blog.csdn.net/zzz_zzz12138/article/details/109138805[1/14/2021 7:17:28 PM]



总而言之，就是需要让我们工程编译后的可执行文件在运行时能够找到需要的动态库，而VS如果采用隐式链接的方式链接动态库，则其搜索动态库的方法可以参考我另一篇文章：
[让Visual Studio \(VS\) 链接静态库 \(lib\) 及动态库 \(dll\) 文件的方法汇总](#)

(2) 由于找不到VCRUNTIME140_1D.dll,无法继续执行代码

除了外部动态库，程序还需要一些windows系统自带的动态库，但有的时候也会出现找不到这些库的情况，这大概是因为你的系统里就是没有这些库。
解决方法也很简单，只需要去<https://www.dll-files.com/>网站下载你缺少的dll动态库，得到该文件的压缩包，然后解压把该.dll文件复制到C:\Windows\System32目录下即可。

2、LINK : fatal error LNK1104: cannot open file ‘torch-NOTFOUND.obj’ (torch-NOTFOUND.obj无法找到)

大概是因为下载的libtorch是release版本的，却在debug下编译，或者反之。
因此解决方法就是要用对应版本的libtorch进行对应方法的编译。
参考：
<https://github.com/pytorch/pytorch/issues/24305>

3、error C2440: “初始化” : 无法从 “torch::jit::script::Module” 转换为 “std::shared_ptr<torch::jit::script::Module>”

如果采用网络上流行的libtorch测试代码：

```
1 #include <torch/script.h> // One-stop header.
2
3 #include <iostream>
4 #include <memory>
5
6 int main() {
7     // Deserialize the ScriptModule from a file using torch::jit::load().
```

```
8   std::shared_ptr<torch::jit::script::Module> module = torch::jit::load("D:/Desktop/pytorch1_0/model.pt
9   ");
10
11   assert(module != nullptr);
12   std::cout << "ok\n";
13   // Create a vector of inputs.
14   std::vector<torch::jit::IValue> inputs;
15   inputs.push_back(torch::ones({ 1, 3, 224, 224 }));
16
17   // Execute the model and turn its output into a tensor.
18   at::Tensor output = module->forward(inputs).toTensor();
19
20   std::cout << output.slice(/*dim=*/1, /*start=*/0, /*end=*/5) << '\n';
21   while (1);
22 }
```

在生成解决方案时，会报错： error C2440: “初始化”: 无法从“torch::jit::script::Module”转换为“std::shared_ptr<torch::jit::script::Module>”

这是因为采用的Libtorch版本的问题。

`std::shared_ptr`是libtorch测试版本使用的变量类型，

而libtorch **version1.2**已经进行了修改：

[jit] `script::Module` is now a reference type

To better align with the [PyTorch C++ API philosophy](#), `script::Module` and `script::Method` are now reference types. Our APIs have been updated to use `script::Module` instead of `std::shared_ptr<script::Module>`.

Version 1.1

```
using torch::jit::script::Module;

std::shared_ptr<Module> m = torch::jit::load("my_model.py");
m->forward(...);
```

Version 1.2

```
using torch::jit::script::Module;

Module m = torch::jit::load("my_model.py");
m.forward(...);
```

https://blog.csdn.net/zzz_zzz12138

因此，应将：

```
1 | std::shared_ptr<torch::jit::script::Module> module = torch::jit::load("../xxx.pt");
```

改为：

```
1 | using torch::jit::script::Module;
2 | Module module = torch::jit::load("../xxx.pt");
```

同时，`module`已经不再是指针，所以也需要将：

```
1 | at::Tensor output = module->forward(inputs).toTensor();
```

改为：

```
1 | at::Tensor output = module.forward(inputs).toTensor();
```

参考：

[1] : <https://github.com/pytorch/pytorch/issues/22382>

[2] : <https://github.com/pytorch/pytorch/releases/tag/v1.2.0>

[3] : https://blog.csdn.net/qq_33507306/article/details/104427134

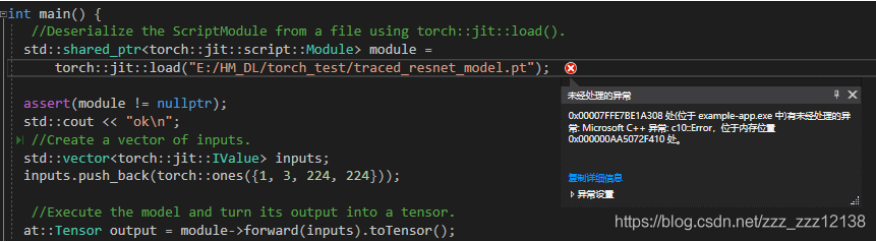
4、无法定位程序输入点cudnnSetCTCLossDescriptorEx于动态链接库xxx.dll上

这个问题我没有找到解决方法，之前在CMake时是同时链接了libtorch库和OpenCV库。我去掉OpenCV库之后重新CMake了一遍就没再出现这个问题。

但是在同时链接这两个库的时候如何解决这个问题我暂时没有找到解决方法。

不过在网上查阅的时候，<https://blog.csdn.net/wb175208/article/details/90479037>提出了一个解决“无法定位程序输入点xxx于动态链接库xxx.dll上”的方法。

5、有未经处理的异常：Microsoft C++异常：c10::Error，位于内存位置xxx处



这个错误出现在用torch::jit::load加载.pt文件时，出现的内存错误。
如果出现这个报错大概率是因为用Python生成.pt文件时所采用的pytorch的版本和C++采用的Libtorch版本不一致导致的。致使C++的libtorch接口无法读取.pt文件。

解决方法就是采用相同版本的pytorch和libtorch。

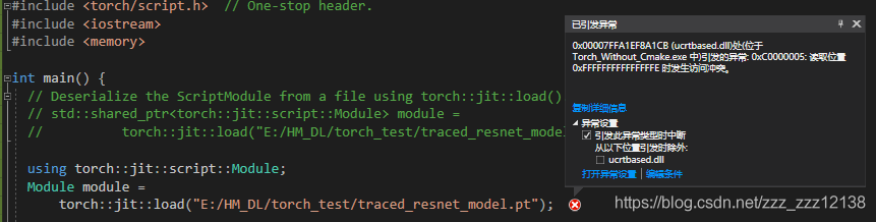
-----2020/12/30更新-----

根据后续大家补充，出现c10::Error不止这一个原因。详情可见评论区。
但总的来说，出现c10::Error基本是由加载的.pt模型有问题造成的，包括但不限于：生成.pt模型的pytorch和libtorch版本不一致；.pt模型的挂载设备（CPU or CUDA）与C++中tensor挂载设备不一致；使用的libtorch支持的CUDA与计算机的CUDA不一致……（欢迎大家在评论区补充讨论）

参考：

[1] 使用vs2019编译c++ libtorch 的 yolov5项目出现的问题，尤其是 Microsoft C++ 异常: c10::Error，位于内存位置 0x0000000D606FE350

6、引发异常：0xC0000005:读取位置0xFFFFFFFFFFFFFFFFFE时发生访问冲突












出现这个问题的原因和前面的“torch-NOTFOUND.obj无法找到”是一样的。也是因为下载的libtorch是release版本的, 却在debug下编译, 或者反之。

所以解决方法就是在release下编译release版本的libtorch，用debug编译debug版本的libtorch。

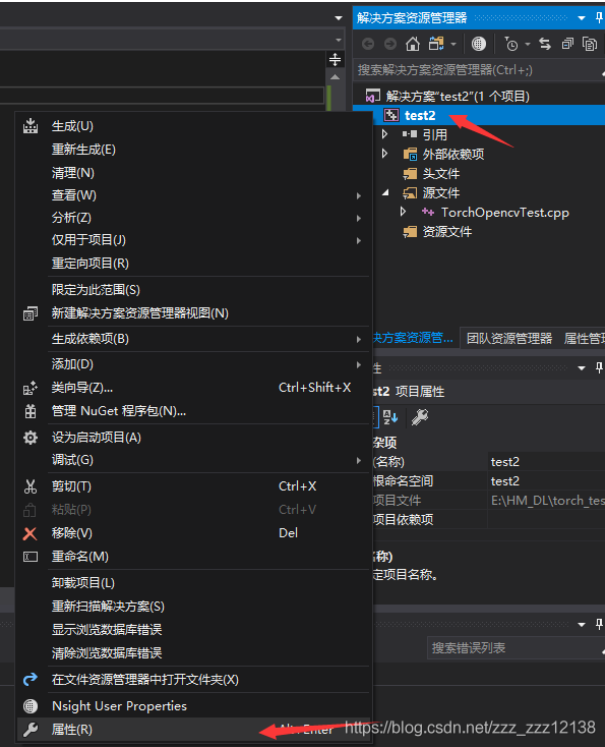
7、error：c2872 std 不明确的符号

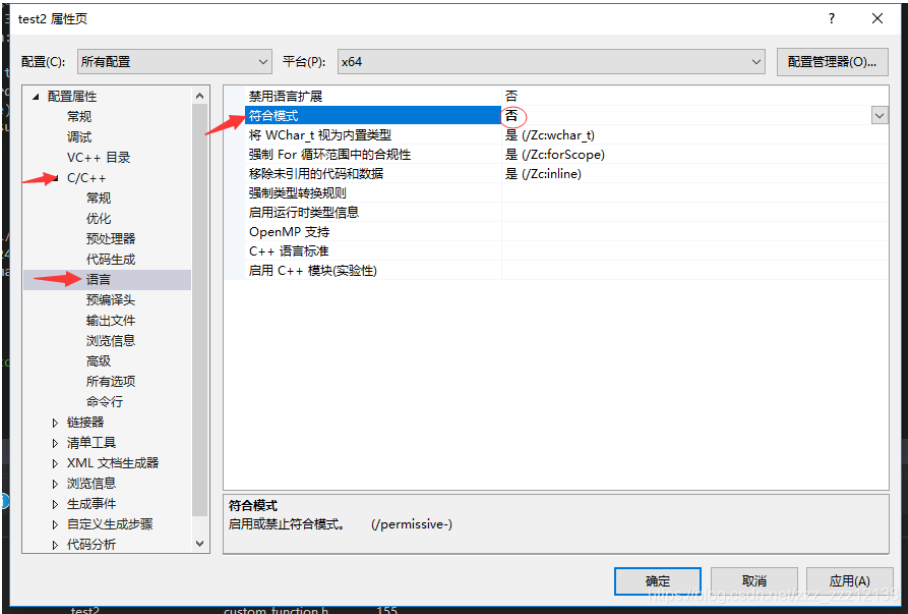
我通过直接配置VS让项目连接libtorch和opencv库，而不是采用CMake的方式。在配置完包含目录、库目录、链接器附加依赖项后，在debug x64下编译项目。报了如下错误：

error：c2872 std 不明确的符号

	C2872	"std": 不明确的符号	test2	module.h	403
	C2872	"std": 不明确的符号	test2	custom_function.h	155
	C2872	"std": 不明确的符号	test2	custom_function.h	156
	C2872	"std": 不明确的符号	test2	custom_function.h	157
	C2872	"std": 不明确的符号	test2	custom_class.h	249
	C2872	"std": 不明确的符号	test2	queue.h	78
	C2872	"std": 不明确的符号	test2	queue.h	79
	C2872	"std": 不明确的符号	test2	queue.h	80
	C2872	"std": 不明确的符号	test2	sequencers.h	108

解决方法：将 属性》C/C++》语言》符合模式 改为否，问题解决。





 声网 **agora**

视频通话SDK

声网Agora，实时音视频 RTC SDK，服务全球覆盖200多个国家和地区，免费接入，每月10000分钟免费

广告 www.agora.io

打开

0x7795DB52 处(位于 test.exe 中)有未经处理的异常: Microsoft C++ 异常: cv::Exception，位于内存位置 0x00
vs2017+opencv3.4在做数字识别的时候为什么总出现0x7795DB52 处(位于 test.exe 中)有未经处理的异常: Microsoft C++ 异常: cv::Exception，

qq_34543664: 我将模型路径作为参数传入进行编译报错，也就是说我并未传入模型，该程序在VS上编译失败，提示c10::ArrayRef<c10::IValue>处，但我已在英伟达的板上编译通过，程序可以通过传参运行 5天前 回复 **


爱码士-zsffuture 回复： 可以参考我的最新博客https://blog.csdn.net/weixin_42398658/article/details/112602722 3小时前 回复 **


爱码士-zsffuture:


```
1 cv::cvtColor(img, img, cv::COLOR_BGR2RGB); // BGR -> RGB
2 img.convertTo(img, CV_32FC3, 1.0f / 255.0f); // normalization 1/255
3 auto imgTensor = torch::from_blob(img.data, { 1, img.rows, img.cols, img.channels() }).to(device_type);
4 imgTensor = imgTensor.permute({ 0, 3, 1, 2 }).contiguous(); // BHWC -> BCHW (Batch, Channel, Height, Width)
5 std::vector<torch::jit::IValue> inputs;
6 inputs.emplace_back(imgTensor);
7 // preds: [?, 15120, 9]
8 torch::jit::IValue output = module.forward(inputs);
9
1
0 //我的死在下面这行了，报错为: 0x00007FFF95D9A388 处(位于 libtorch.exe 中)有未经处理的异常:
1
1 // Microsoft C++ 异常: c10::Error，位于内存位置 0x0000000D606FE350 处。
```


```
1
2      auto preds = output.toTuple()->elements()[0].toTensor();
```

您好作者，麻烦您看看我死在这里是怎么回事呢？ 15天前 回复 ●●●

- 

zsfuture 回复： 我的找bug过程不仅仅是上面提到的，我还遇到了可以进行进行检测，也返回正确结果。但是在进行非极大值抑制时又报了类似的错误，我这里提醒一下大家问题发生的根源，是因为我调用GPU进行计算，返回的结果也是GPU的，此时仍然在GPU下，但是非极大值抑制的代码在GPU运行就出问题了，原因分析后发现是因为非极大值抑制代码的变量是CPU下的，因此直接运行会报错，报错结果类似，此时我增添了使用CPU运行，程序便可以顺利跑通，整个工程也顺利跑通，我用的版本是 torch 1.7.1 CUDA:0 (GeForce RTX 2060 SUPER, 8192MB)，对应得libtorch也是最新版本 15天前 回复 ●●●
- 

zsfuture 回复 Z的三次方： 行 15天前 回复 ●●●
- 

Z的三次方 回复： 学习了👍我在c10的error里补充一下，引用一下你的这篇可以不 15天前 回复 ●●●
- 

zsfuture 回复： 我的问题解决了，可以参考我的这篇博客https://blog.csdn.net/weixin_42398658/article/details/111954760 15天前 回复 ●●●



苏三福： 博主你好，我是在前向计算报错torch::Tensor preds = module.forward({ imgTensor }).toTuple()->elements()[0].toTensor(); 版本一致，不知道什么原因 16天前 回复 ●●●

登录 查看 28 条评论

Windows环境下使用 LibTorch (1.4) Challovactor 的博客 2717
最近使用LibTorch在调用分类模型，在配置环境时，用到了1.4版本的PyTorch，按照网上的一些教程无法成功，便把自己部署成功的经验分享出来。 1. 从...

C++调用libtorch新版本报错、出错 m0_37729957 的博客 1249
libtorch新的版本如何调用.pt文件一、在pytorch1.2版本中，我们使用Visual Studio时调用.pt文件时，会报错，如下1、不存在用户定义的从"torch::jit::script::...

在C++部署Pytorch(Libtorch)模型的方法总结(Win10+VS2017) 12-10
运行成功的话会生成解决方案,选择合适的编译方式,生成可执行文件,然后运行。如有报错,参考我的C++部署Pytorch(Libtorch)出现问题、错误汇总 ...

windows10用c++部署libtorch过程中遇到的一些问题_jack... 12-26
libtorch1.0 vs2017 CMake3.14 windows10 无cuda 用c++调用pytorch模型官网上有详细教程,也有很多博客,可以参考以下链接:https://blog.csdn.net/dou3...

C语言运行时库详解 jxth152913的专栏 1690

运行时库是程序在运行时所需要的库文件，通常运行时库是以LIB或DLL形式提供的。C运行时库诞生于20世纪70年代，当时的程序世界还很单纯， ...

从TorchScript生成模型到windows+vs2019+LibTorch调用模型 耳东小白的赤兔秘籍 526
从TorchScript生成模型到windows+vs2019+LibTorch调用模型 文章目录从TorchScript生成模型到windows+vs2019+LibTorch调用模型References环境配...

关于windows10用c++部署libtorch过程中遇到的一些问题... 12-25
libtorch1.0 vs2017 CMake3.14 windows10 无cuda 用c++调用pytorch模型官网上有详细教程,也有很多博客,可以参考以下链接:https://blog.csdn.net/dou3...

pytorch c++部署编译?libtorch库加载报错?_thankyvision 1-7
pytorch的c++接口也发布了(libtorch),便于c++平台环境和使用条件下的部署,但是,如果你是從官网上直接下载的release版本的话,有可能在你的系统下不能...

C# ChineseSpell.DLL 简体汉字转拼音 06-04
ChineseSpell.DLL 的汉字转拼音，支持多音字，支持音标。 若要查看函数注释，请同时复制 ChineseSpell.xml 文件到 ChineseSpell.DLL 的路径下

GetLastError返回代码含义 无意搞花 3万+
参考[http://msdn.microsoft.com/en-us/library/ms681381\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/ms681381(v=vs.85).aspx) 【0x0】-操作成功完成。 【0x1f1】-函数不正确。 【0x2f2】-系统找不到指定的文...

C++端部署pytorch, libtorc使用教程(精简版)_baidu_341... 1-9
的方便,但是实际部署却以C++端为主,所以如何让自己的模型很好的再C++端运行(除了C++调python代码)一直以来是个重要研究点,C++部署方法很多,这里...

PyTorch开发部署时5个常见错误。_Memory1011的博客 1-8
点赞Mark关注该博主，随时了解TA的最新博文 pytorch怎么使用c++调用部署模型? 豆芽菜 2万+ pytorch的部署——把pytorch模型集成到so库 Haward 6131...



"Unhandled exception in app.exe (QtGuid4.dll): 0xC0000005: Access Violation"错误解决	当凌绝顶,俯瞰天下 1544
本文要解决的Qt调试问题截图如下：起因：在代码中添加类型为QColor的二维数组，存储图片每个像素。现象：调试时出现上图所示错误。原因：...	
Linux&Windows下用C++跑pytorch模型	JR_Chan的博客 801
前一篇博客讲了怎么用pytorch实现一个简单的图像分类器，这一篇的主要目的则是将前面训练好的模型用C++跑起来。模型转换 第一步需要python...	
Pytorch C++部署 之 TorchScript 踩坑记录_II_cookie的...	1-10
小白一枚,最近尝试用c++调pytorch模型,网上搜了一对资料,大同小异,都是需要先将模型文件转换成torch script。 pytorch官网提供两种方式,1.trace 2.注释。	
pytorch1.3版本转C++ API(libtorch)踩坑记录_ryERO的博客	1-4
最近需要把之前写的一套模型部署到服务器,需要从pytorch->C++,参考官方提供的方案基于libtorch1.3版本。 先列几个我觉得写的比较好的教程: ...	
【pytorch】加载模型出现的bug	ly771857509的博客 2287
在模型训练完后再进行测试加载模型后出现bug，显示如下错误 据了解是由于pytorch版本导致的错误，可能与自己训练阶段保持的模型方式有关，训练...	
pytorch 在加载模型时出现的错误	xixilidehuoshan的博客 2960
在训练完后，在另一个.py中想导入模型进行使用。导入之后出现这样的错误： pytorch AttributeError: Can't get attribute 'Net1' on <module 'main' &...	
ExcelDll.rar	05-19
用于读取Excel的DLL文件	
win8.1安装Python3.5以上提示缺失api-ms-win-crt-runtime-l1-1-0.dll解决方法	qq_36709271的博客 3521
操作系统win8.1（x64bit），在安装Python3.5以下时是能运行的，3.5及以上时就出现api-ms-win-crt-runtime-l1-1-0.dll丢失的问题经过百度，曾作出以下尝...	
opencv342+dll.zip	12-20
opencv342+dll 包含了opencv342编辑前文件，可以使用cmake-3.11（或者其版本）进行编译，在编译过程中会出现卡dll下载的情况，这个opencv_ffmpeg...	
C10_动态内存分配	Clare_BiJuan的博客 761
// // main.m // C10_动态内存分配 // // Created by dllo on 15/7/13. // Copyright (c) 2015年 Clare. All rights reserved. // #import void func(){ int x = 5; in	
云计算正在“抹杀”开源？	CSDN资讯 1万+
【编者按】开源之初，无数开发者满怀热情，无偿分享自己的代码，希望为世界贡献自己的一份力量。然而当开源加入商业模式，开发者们慢慢意识到，...	
Android开发入门60个小案例+源代码	08-24
适合初学者，大量简单小例子，完整源代码。	
©2020 CSDN 皮肤主题: 1024 设计师: 上身试试 返回首页	

