**RNN 部署 (Centos 6)**

**Qiang Li**

1. gcc-4.9.3 安装

需要 gcc-4.9.3.tar.bz2 安装包

# yum install libmpc-devel mpfr-devel gmp-devel

$ tar xvfj gcc-4.9.3.tar.bz2

$ cd gcc-4.9.3

$ ./configure --disable-multilib --enable-languages=c,c++

$ make -j 4

# make install

1. boost\_1\_55\_0 安装

需要 boost\_1\_55\_0.tar.gz 安装包

$ rpm -qa boost // 如果centos已经安装boost，则通过上述指令

// 会显示版本，卸载

# yum remove boost // 卸载boost, 使用第一条指令，不会显示任何信息

$ tar xzvf boost\_1\_55\_0.tar.gz

$ cd boost\_1\_55\_0

$ ./bootstrap.sh

$ ./b2

$ ./b2 install

1. CUDA\_8\_0 宏定义

Cuda\_kernel.h 文件中

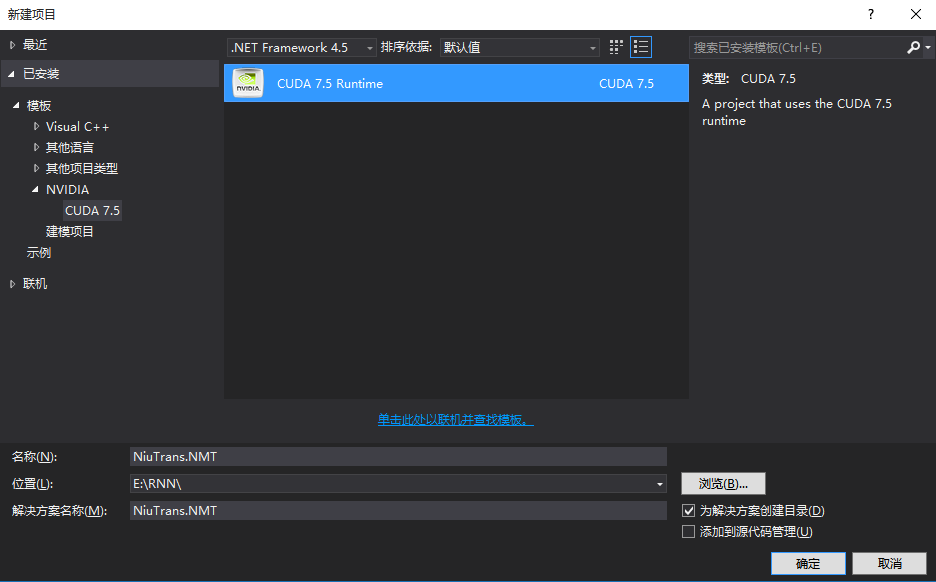
如果使用的CUDA\_8.0，则定义 #define CUDA\_8\_0

如果使用的CUDA\_7.5，则使用 //#define CUDA\_8\_0

1. Eigen

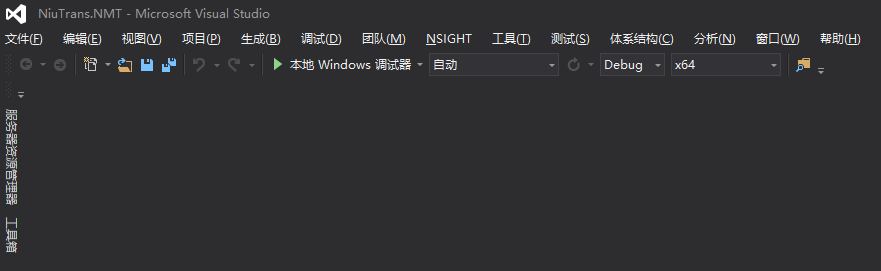
**RNN 部署 （windows, vs2013）**

1. 安装 boost\_1\_60\_0
2. 在任意文件夹下放入 Eigen
3. 安装 VS2013 (VS2015无法使用)
4. VS2013 新建cuda项目



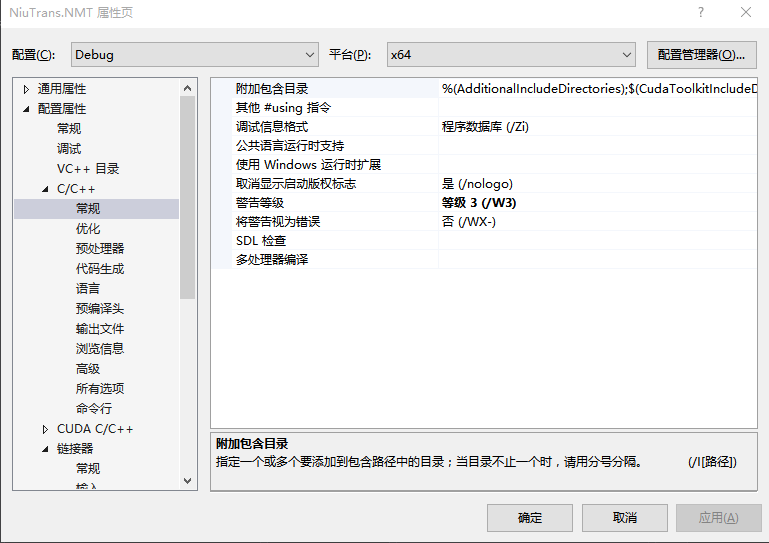
删除自动生成代码，使整个解决方案为空

1. VS2013 导入所有代码



选择 x64 模式

1. VS2013 配置参数修改

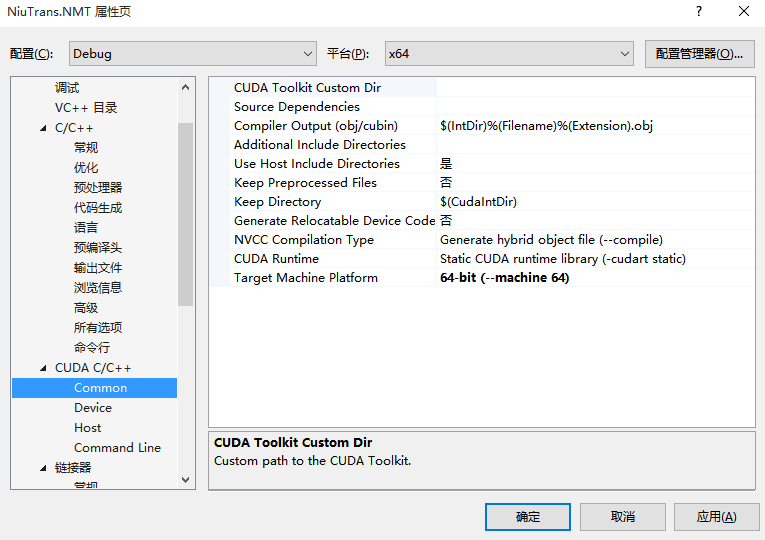


在 c/c++ 常规 → 附加包含目录 末尾添加

;C:\boost\_1\_60\_0\;E:\RNN\eigen-forcuda\eigen-win

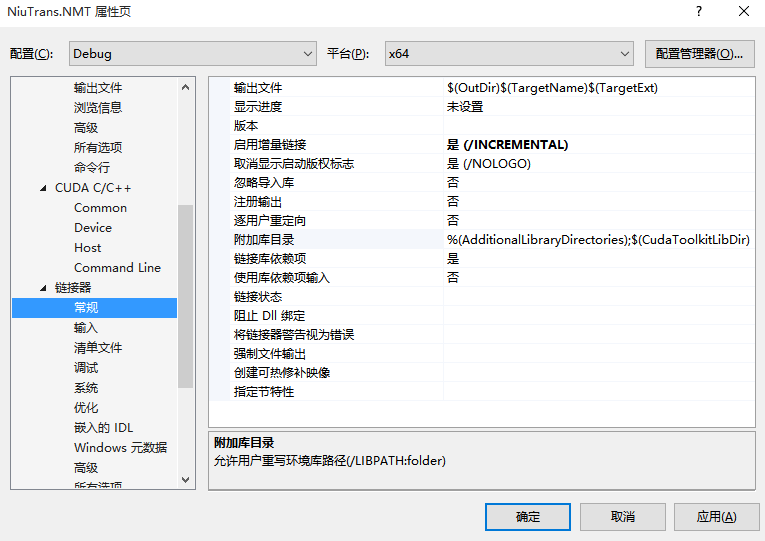
这里我们安装boost\_1\_60\_0于C盘

Eigen放在E盘



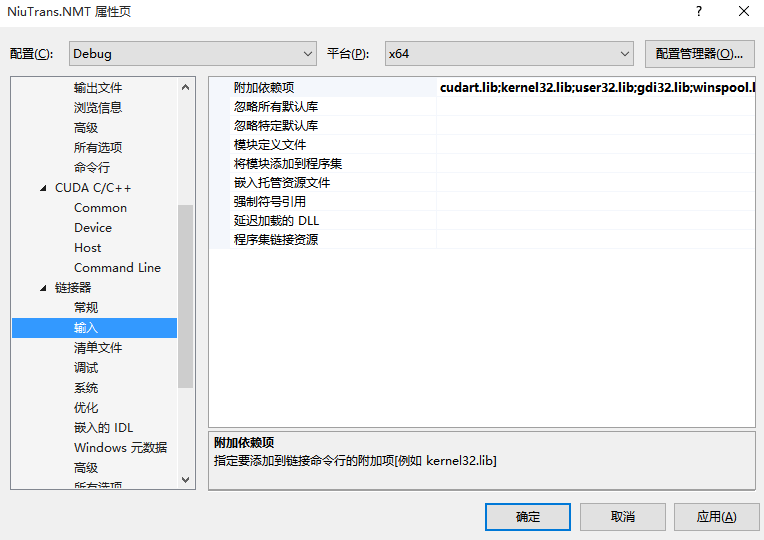
在 CUDA C/C++ → Common → Additional Include Directories 添加

C:\boost\_1\_60\_0\;



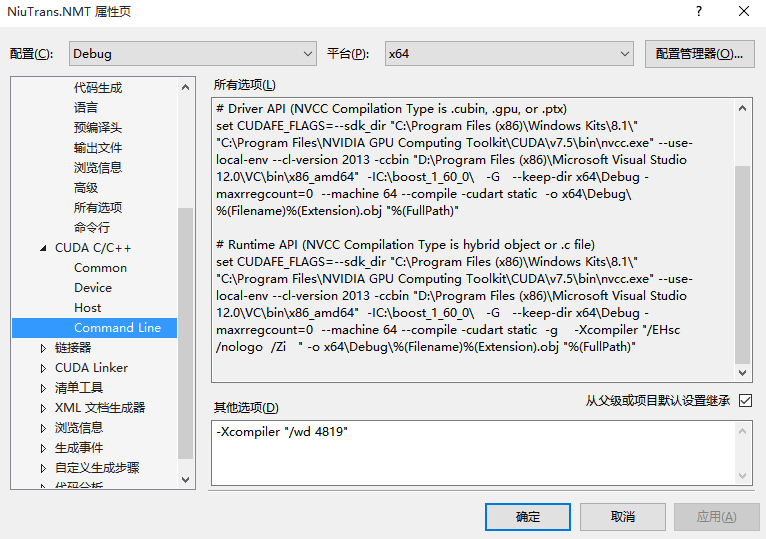
在 链接器 → 常规 → 附加库目录 末尾添加

;C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v7.5\lib\x64\cublas.lib;$(CudaToolkitLibDir);C:\boost\_1\_60\_0\bin\vc12\_x64\lib\;



在 链接器 → 输入 → 附加依赖项 头部添加

curand.lib;cublas.lib;



在 CUDA C/C++ → Command Line → 其他选项 中添加

-Xcompiler "/wd 4819"