复用模块代码说明文档

* 主要实现功能：数据下载并转换为tfrecord格式、调用。
* 数据下载并转换为tfrecord格式接口：
* Cifar10：download\_and\_convert\_cifar10.py中最后if \_\_name\_\_ == '\_\_main\_\_'函数；
* 输入：dataset\_dir（存贮数据库的文件夹目录）；
* 输出：tfrecord格式的数据库。
* flowers：download\_and\_convert\_ flowers.py中最后if \_\_name\_\_ == '\_\_main\_\_'函数；
* 输入：dataset\_dir（存贮数据库的文件夹目录）；
* 输出：tfrecord格式的数据库。
* 数据库调用接口：
* dataset\_factory.py中get\_data(hyp, is\_training)函数：
* 输入：hyp超参数，包括数据库名称、训练分割名称（‘train/test’）、数据库所在地址、训练所用batch size大小、归一化后图像尺寸；is\_training：bool类型，是否为训练过程；
* 输出：batch size大小的图像信息，batch size大小的标签信息。
* 简单实例

超参数hyp配置:

if FLAGS.dataset\_name == 'cifar10':

self.DATA\_PATH = '/mnt/data1/liuhongli/data2/nasnet\_tensorflow\_master/data/cifar10'

self.DATA\_TRAIN\_NUM = 50000

self.DATA\_TEST\_NUM = 10000

self.NUM\_CLASSES = 10

self.IMAGE\_SIZE = 32

self.train\_split\_name = 'train'

self.test\_split\_name = 'test'

from dataset\_factory import get\_data

训练调用：images\_train, labels\_train = get\_data(hyp, is\_training=True)

测试调用：images\_test, labels\_test = get\_data(hyp, is\_training=False)

batch\_x, batch\_y = sess.run([images\_test, labels\_test])