Transfer Learning Tools
这篇文档主要是为了介绍一下在师兄的easyME基础上改的模型训练和预测工具，由于条件有限，我会尽量详细的写出记的比较清楚的部分内容。

目录

1总体介绍
这个工具是为了完成吕新波师兄的课题而在实验室原有easyME工具的基础上修改而来，主要可以完成训练语料和标注语料的工作。其中训练语料部分和原有easyME部分完全相同，标注语料部分根据论文《A Comparative Study of Methods for Transductive Transfer Learning》的transductive transfer部分编写，根据论文中的原理，标注程序会同时读入源语料，目标语料和训练模型，通过统计源语料和目标语料的期望比来调整训练模型中的特征的权重，并根据更新后的模型来标注目标语料，达到迁移学习的目的。
工程的完整代码在实验室服务器的/home/zhuxi/Recouse/MyeasyME/easyME目录下，里面已经有编译好的程序，可以直接运行。
在后面的介绍中均将训练语料视为源语料，测试语料视为目标语料，特做说明。

2语料处理
工具对语料的要求与easyME的要求完全一致，每行为一个事件，每个事件由若干个字符串组成，这些字符串由空格或制表符分开，字符串本身不带空格。
每个事件的第一个字符串为标注结果，这个在训练语料中是必须的；在测试语料中若没有标准标注可以用任意字符串代替，但如果这么做的话程序本身自带的准确率统计就是无效的。
事件的第二到第N个字符串为此事件的特征，一般可以由当前词，当前词的词性，当前词的构词特征，前一词，或是一些特征的组合等等来构成，训练语料和测试语料的特征选择最好是一致的以获得比较好的效果。
在工程目录下应该有我自己试验用的训练语料和测试语料，名为new_train_file和new_test_file，主要用的是病历文献，其中选取了当前词，当前词的词性，当前词是否为描述蛋白质的一部分，当前词的后4个字母，当前词的后2个字母，当前词的词形这6个方面的特征，train和test的格式完全一样。这里因为时间有限所以只是简单的提取了一些特征，更好的方法是利用几个特征之间的组合来构造新的特征，应该会加强标注效果。

3用法
训练语料：和easyME完全一样，可以使用以下命令
./train "train_file_name" "model_file_name"
其中"train_file_name"是训练语料的名称；"model_file_name"是要建立的训练结果模型的名称，可以自己定义。
在这两个参数后面还可以加一些其他的训练时用到的参数，比如收敛方法，迭代参数等等，具体可以到/examples/train.cpp这个文件中查看。

测试语料：使用如下命令
./test "model_file_name" "train_file_name" "test_file_name"
其中model_file_name为刚才train程序训练出的模型的名称。
最后测试会在一个叫"…"文件的每一行输出一个字符串代表每个事件测试的结果。假如测试语料中有标准的标注的话程序还能显示测试结果和标准标注比较的准确率。
另外我还写了几个python脚本"swap.py"和"eval…"和"…"来处理测试结果，具体流程大概是先swap.py将test_file的标准标注移到每个事件的最后，然后是用"…"将测试结果加入test_file的对应事件的最后，最后使用"eval…"统计实体标注的准确率，召回率和F值。（更具体的描述有点记不清了，有"…"的地方表示忘了那个文件名是什么了，要是可以看到文件的话就可以知道怎么用了）

4相关函数
/src/dataManager.cpp：
添加了getSourceExpected和getTargetExpected函数，分别计算了源语料和目标语料每个特征的期望值。
添加了getAllProbsNew函数，用于计算每个事件中每个可能的标注在新的模型下出现的概率。
/src/maxEntModel.cpp：
修改了loadModel函数，主要是添加了读取源语料和目标语料的功能，然后调用了getSourceExpected和getTargetExpected函数。
添加了testModel函数，主要用于使用新的特征权重来对语料进行测试和输出结果。
/examples/test.cpp:
修改了主函数以正确调用以上函数。