# 中文词性标注程序

**马琛\_1120131903\_07111304**

**一、项目概要**

**1.**本项目（ver0.8）实现了对asci编码（GB2312）的中文文章进行词性标注（文章中可以出现半角字符）。本项目是在之前的分词程序基础上开发而成（ver0.7）

**2.**通过正向最大匹配（fmm），逆向最大匹配（bmm），隐马尔可夫模型（hmm）分词。分词完了用隐马尔可夫模型进行词性标注

**3.**词性标注的隐马尔可夫可由语料库训练生成（此处用的清华的）

**4.**对分词结果的正确率、召回率即时间进行评价。

**5.**本项目采取面向对象的编程思路，由c++编写完成，同时兼顾效率，方便后期添加功能

**二、开发环境**

vs2010 ，c++10

**三、代码设计与类**

DivideWord

|BFMM

|BMM

|FMM

|HMM

|TagWord

|HMMword

Viterb

TagWord：词性标注的基类，其中有分词器、词性标签枚举型，提供词性标注接口与词性标注虚函数

HMMword：隐马尔可夫模型的词性标注（tagword\_sentence），提供训练（train\_hmm）、存储、读取功能

其他类与上一个分词版本(v0.7)相差不大，只是修正了一些bug和优化内部细节，接口没变，其他类的说明详见中文分词程序v0.7（大作业1）。

各个类函数与算法具体细节详见src目录工程代码中的注释。

**四、我的感想**

本项目由我独立完成。参考了网上的各种算法资料，包括SVM和CRF算法等，其中这几个网站对我帮助挺大<http://www.52nlp.cn/>和<http://www.hankcs.com/nlp/> 。也考虑过基于错误驱动的方法，但是个人感觉还是隐马尔可夫模型更直观些，而且之前把维特比算法单独抽出来实现，降低了系统耦合性。由于这样就可以继承原来的项目继续开发，很多代码可以复用，就决定用隐马尔可夫模型了。

词性标注原理与分词原理类似，只是把BMES（隐藏态）变为各个词性标签，把单个字（观察态）变为词语，训练方法与分词系统hmm相同。与之前不同的是，这次的发射矩阵很大(55000\*40\*8byte)，而系统默认堆栈大小1m，运行之后程序崩溃，而又不想放到全局变量、静态变量或是堆区，于是查了一下资料，用了一条编译预处理调整堆栈大小#pragma comment(linker, "/STACK:5000000000 ")解决了问题。虽然也想过稀疏矩阵压缩方法，但是还是要追求速度，这里就用空间换时间了。这里做了一点优化，词语用了map进行hash值索引，提高效率。此程序还存在的问题是，对未登录词语处理效果不是很好。

做完这个项目后，感觉收获很多。我一开始并不对自然语言理解很感兴趣，感觉很多算法很不好理解，而且在实现算法之前的预处理也是很麻烦，再加上统计和概率相关的程序不是很直观，debug略困难，所以之前进度很慢。但是在2015年的最后一天，陪伴我的是hmm（当然也有wa2）。看了一遍看不懂，再看，硬着头皮往下看，慢慢的也就理解了。发现那些思想很是巧妙，算法也很有意思，我开始喜欢上自然语言理解了。

我尝试着去构建我的自然语言理解系统(NatureLU)，一开始很丑陋，但是在一次次的修改中工程架构变得逐渐清晰，逐渐形成了自己的风格，耦合性降低。面向对象的编程风格真的很漂亮，各个对象有机的结合在一起，非常方便后期去添加功能和维护。

感觉困难的问题，不要去拒绝，迎着头皮往下做，当你理解了，那满满的成就感，你会喜欢上的。

**附：本项目的历史版本记录**

