

拼音输入法实验报告

高寒 P18106019

问题概述

拼音输入法可以按注音符号与汉语拼音两种汉字拼音方案分成两大类。汉语拼音输入法的编码是依据汉语拼音方案进行输入的一类。早期只有全拼这种方式，即完全依照汉字的整个音节来输入。随着技术的发展，拼音输入法不仅可以简拼还出现了一种只需两键就能输入整个音节的双拼方案。

在本次作业中，我们要求同学们自己编程实现一个简单的汉语拼音输入法，即实现从拼音(全拼)到汉字(字串)内容的转换。

主要要求是使用基于字的二元模型实现拼音到汉字的转换程序。

解题思路

在二元模型的情况下，当前这一个拼音 p_i 对应的汉字 x_i 只与其前面一个拼音 p_{i-1} 对应的汉字 x_{i-1} 有关，因此可以先计算得出当前每个词组出现的概率，然后再通过相关算法进行计算一句拼音所对应的最大概率的句子即可。

算法实现

本次实验训练的材料主要来源于所提供的 sina_news 以及自己从网上找到的一些小说材料

一、 汉字对应拼音提取

因为一个汉字有可能是多音字，所以需要提取汉字的所有拼音并保存。这里使用所提供的“拼音汉字表.txt”来统计每个汉字的对应的拼音，同时在训练的时候判断每个字是不是多音字，如果是多音字，便通过python的pypinyin库函数来对当前这个多音字前后4个字进行注音，得到这个字在当前地方的正确读音，然后存到对应的字典里边，其格式为：{"汉字": [{"拼音1": 概率, "拼音2": 概率} ...] }。（例：{'塞': {'sai': 0.8, 'se': 0.2}, ...}）

二、 对每个汉字出现的频率进行统计

由于计算第一个拼音的时候无法根据其前一个拼音来进行计算，所以需要统计每个字出现的概率，其格式为：{"汉字": 频数, "汉字": 频数...}。（例：{"你": 134, "还": 56, ...}）

三、 对每个词的频率进行统计

因为是对两个连着的汉字进行统计，所以在统计的过程中需要对非汉字进行过滤，这里通过将待统计的语句通过正则匹配进行切分以后得到只有汉字的一个列表，然后在列表中的汉字进行相应的统计，其主要格式为：{ "汉字1": { 'cnt': 总频数, 'words': { "汉字1": 频数, "汉字2": 频数, ... } }, "汉字2": }。

(例： '你': { 'cnt': 80, 'words': { '好': 23, '门': 34, ... }, '我': { 'cnt': 790, 'words': { '好': 93, '取': 32, ... }, ... }, }

四、生成概率模型

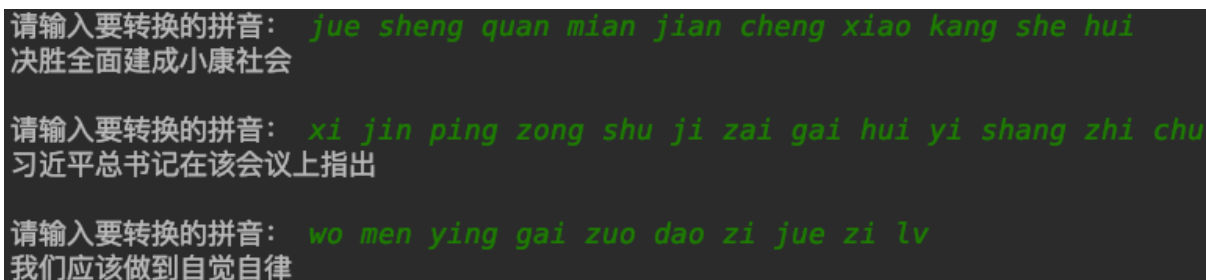
通过对上面统计的词频进行处理，得到概率模型，然后将所有的数据存到 `prob_model.json` 文件里边，为拼音转换提供模型。

五、实现拼音转汉字算法

实现拼音转汉字的算法主要使用了Viterbi 算法，该算法基于动态规划思想，并使用最大似然估计作为其概率。通过该算法计算最后便可以得到一个概率最大的语句。同时由于训练材料的限制，可能会出现两个汉字在之前并没有出现过，因此引入了一个最小概率，来提高算法在遇到这种情况时的处理。

实验效果分析

较好的例子：



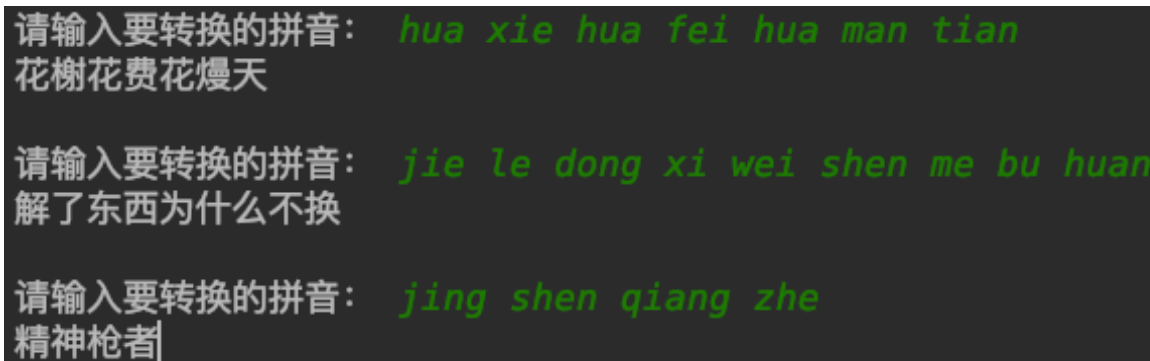
请输入要转换的拼音： *jue sheng quan mian jian cheng xiao kang she hui*
决胜全面建成小康社会

请输入要转换的拼音： *xi jin ping zong shu ji zai gai hui yi shang zhi chu*
习近平总书记在会议上指出

请输入要转换的拼音： *wo men ying gai zuo dao zi jue zi lv*
我们应该做到自觉自律

由上面的图片中我们可以看出在对于一些比较正式的句子中，能够正确的翻译过来，这个主要是因为训练材料主要是新闻材料的原因

较差的例子



请输入要转换的拼音： *hua xie hua fei hua man tian*
花榭花费花慢天

请输入要转换的拼音： *jie le dong xi wei shen me bu huan*
解了东西为什么不换

请输入要转换的拼音： *jing shen qiang zhe*
精神枪者

通过对一些诗句和某些句子进行翻译时，会发现结果不是很理想，这个主要的原因是因为训练材料量少，对于每个词的统计量并不是很均匀，由于训练语料的类别不同，便会导致对不同类别的语句的识别概率不同。

对比参数选择及性能分析

一、 相关参数选择

在进行拼音转汉字的时候，发现有的词并没有出现过，会导致无法准确的计算出结果，因此引入了一个最小概率参数，刚开始尝试将该最小参数设置为在第一个字之后出现一次另外一个字时的概率，后来通过测试发现这样计算其准确度并不高，后来将其设置为未出现过的第二个字的概率乘以 $1e-5$ 来进行计算，发现能够有效的提高准确度

二、性能分析

通过拼音转汉字算法的实现，目前该算法的准确率在所提供的测试集的准确率为：77%左右，可以通过调节参数来进行微调（无其他测试集，不确定是否能微调）

总结及改进方案

总结

通过本次实验，让我对python的使用更加的熟悉了，同时在编写代码的同时也了解到了这种基于模型的算法的实现过程，提高了我对课程相关知识的理解。同时也知道了在一样的数据集下，通过不同的训练方法训练出来的模型所得到的结果会是不一样的，同时也有各种不同的算法来进行相关的计算。也让我学到了一些相关的基础算法。

改进方案

1. 由于当前方案只是根据统计句子两个字之间的频率，导致统计的一些词组根本不是常见的词组，因此，后续改进可以将待统计的句子先进行一个简单的分词处理，然后在进行一个统计
2. 在进行转换的时候，如果遇到之前未出现的词的时候原方案是采用当前字出现的频率在进行计算的，但是该计算效果并不会很好，因此可以找出一种合适的参数来进行处理
3. 在二元模型的计算结果中，有时会由于某个词出现的频率特别高，便导致总的结果偏向于有这个词的语句，相应的改进便是针对于频率次数过高的词进行一个降频的操作，但是前提条件是在不影响正确结果的条件下进行降频。