# 拼音输入法编程实验报告

人工智能 2017 年秋季学期

水博 16 刘千惠 2016310082

## 1. 算法思路与实现

算法主要包括文本训练和拼音汉字转换两个部分。拼音汉字转换需要读取文本训练得到的数据库。

### 1.1. 文本训练

第一部分是根据提供的语料统计相应的频次。因为考虑的是二元模型，所以只考虑单个字连续两个字出现的频率就可以了。最直观的想法是建立一个多叉树，如图 1.1 所示。

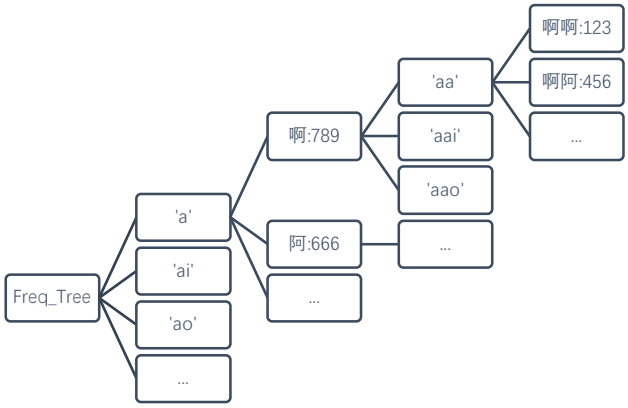


图 1.1：语料统计频次树

虽然可以根据拼音汉字表直接建立整个树，然后再去统计每个字和词出现的次数，但是这样生成的频次树就会比较大。而且考虑到中文虽然字很多，但是常用的字不多，会出现的组合也比较有限，所以根据语料从最底层建立频次树应该是比较好。

这个时候就会遇到多音字的问题。通常当一个字出现在一句话中的时候，字的读音是固定的，而通过汉字拼音表是没有办法得知字的准确读音的。但是因为算法最终的目的是实现

拼音到汉字的转换，所以可以把每个字的读音排列组合储存起来。这样会使频次树中包含一些无用信息，经过计算和尝试发现这些无用信息并不会使频次树过大，不影响使用，因此就没有进一步优化。

具体实现的时候，考虑到频次树比较简单，直接用 python 中的字典来实现。在 Mac OSX EI Capitan10.11 系统下，处理 $10^8$ 个汉字用时大约 50min，占用内存约 100M。

得到的频次树和其他数据用 json 格式输出为 Freq_Tree.json。

这一部分的代码见 trainer.py。

### 1.2. 拼音汉字转换

第二部分是从拼音转换成汉字的部分。

这一部分需要调用第一部分生成的 Freq_Tree.json，默认情况下会从根目录下找这个文件，假如根目录下没有这个文件，可以通过命令行指定文件的位置。

另外，需要读取输入文件，并定位输出文件。默认情况下会从父目录找 data/Input.txt 和 data/Output.txt 分别作为输入和输出文件，当找不到输入文件时，可以从命令行指定文件的位置，此时默认在指定文件位置建立相应的输出文件。无论是否指定输入文件的位置，当输出文件已经存在时，可以从命令行选择更换输出的目录和文件名，或者直接覆盖当前文件。

采用二元语法下的条件概率，对可能的字的排列组合进行估值。对于一串给定的拼音，最终的目标是求下式的最大值：

$$\prod_{i=1}^{n} P(w_i \mid w_{i-1})$$

其中，

$$P(w_i \mid w_{i-1}) = \frac{w_i w_{i-1}\text{同现的次数}}{w_{i-1}\text{出现的次数}}$$

并利用下式对上述条件概率做平滑处理：

$$\lambda P(w_i \mid w_{i-1}) + \lambda P(w_i) => P(w_i \mid w_{i-1})$$

从 Freq_Tree.json 获取单字和词组出现的频次时有两种情况，一种是不知道前面的汉字的情况下求两个拼音对应的汉字的情况，另一种是已知前一个汉字的情况下求后一个汉字的情况。定义 vv2ww 和 wv2ww 两个函数，处理这两种情况。图 1.2 为这两个函数的调用流程。其中还包含了对单个字的处理，以及对非拼音的输入项的处理。

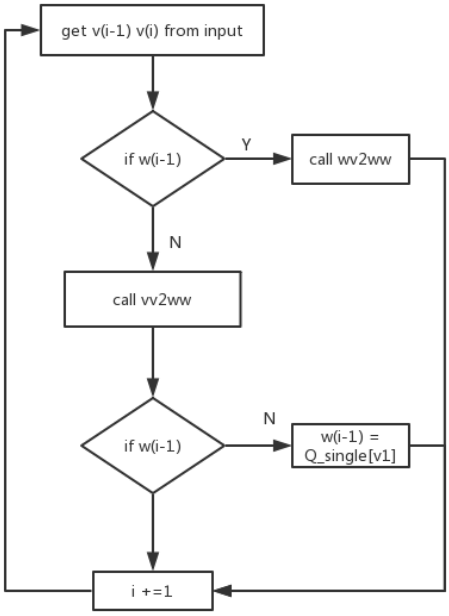


图 1.2：vv2ww 和 wv2ww 的调用流程

对于每一次循环，对所有可能的结果进行排列组合，找出条件概率之积最大的组合。考虑到在统计频次的时候，并没有忽略掉出现频次很低的词汇组合。由于出现的频次很低，其频率可能接近 1，从而导致计算得到的概率较高的词组并不是想要的词组。为了避免这种，设置限制参数 `Q_bottom_lim`，当词组的频次低于这一限制时，词组将不被考虑。

当对结果进行配对组合时，设定限制参数 `itm_limit` 和 `tmp_list_limit`，限制参与组合的字符串个数，按照当前计算得到的频率进行排序，选择频率较高的前几位进行组合。这个参数一开始的设计是为了避免排列组合太多，导致运行速度太慢。但是实验中发现，对于这

次使用的数据，即使不做限制也不会出现这样的问题。但这两个参数对于程序的扩展是很有意义的。

这一部分的代码见 PinYin.py。

### 1.3. 编译环境说明

本报告中实验所涉及代码使用 pycharm community 5 + python3.6.3 编译，在 Mac OSX EI Capitan10.11 系统下成功运行。输出文件的编码方式均为 utf -8。

## 2. 实验结果及参数分析

首先对已给出的四个输入算例子进行转换，表 2.1 中列出了得到的转换结果。

表 2.1: 基本例子转换结果

| 拼音 | 期望得到的汉字 | 转换得到的汉字 |
|---|---|---|
| qing hua da xue ji suan ji xi | 清华大学计算机系 | 清华大学计算机系 |
| wo shang xue qu le | 我上学去了 | 我上学去了 |
| jin tian hui jia bi jiao wan | 今天回家比较晚 | 今天回家比较完 |
| liang hui zai bei jing zhao kai | 两会在北京召开 | 两会在北京召开 |

从表中的结果可以看到，大多数的转换结果是令人满意的，只有第三条的“晚”被转换成了“完”。调出频次库中的数据查看发现，在整个语料库中，“较”字出现的次数为 260402 次，“较完”这个组合出现的次数为 1049 次，“较晚”这个词组出现的次数为 847 次。对于“wan”这个发音，“完”出现的次数也大于“晚”出现的次数。对于实验采用的算法而言，这两方面的原因导致了“今天回家比较完”这句话的估值一定大于“今天回家比较晚”这句话。

对于这个例子，语料库是导致转换不准的主要原因之一。相比于“较完”，“较晚”明显是更加高频的词汇，在实际使用中，后者的频次是远远大于前者的，但是在语料库中前者的频

次却大于后者。另一方面，采用二元模型对拼音进行转换，也可能是导致结果出准确的原因。假如分析语料库中“比较完”和“比较晚”这两个词组的频次，后者应该是会远远大于前者的，也就不会出现这个例子中的问题了。

在调整参数的过程中发现，平滑参数对于转换结果影响不大，只要参数取值大致合理，得到的结果都是相同的。而限制频次下限的参数 Q_bottom_lim 对结果的影响非常大。

表 2.2 列出了这个参数取不同值的情况下的转换结果。

表 2.2：不同参数下的转换结果对比

| Q_bottom_lim | 拼音 | 期望得到的汉字 | 转换得到的汉字 |
|---|---|---|---|
| 450 | qing hua da xue ji suan ji xi | 清华大学计算机系 | 清华大学计算机系 |
| | wo shang xue qu le | 我上学去了 | 我上学去了 |
| | jin tian hui jia bi jiao wan | 今天回家比较晚 | 今天回家比较完 |
| | liang hui zai bei jing zhao kai | 两会在北京召开 | 两会在北京召开 |
| 300 | qing hua da xue ji suan ji xi | 清华大学计算机系 | 氰化大学计算机系 |
| | wo shang xue qu le | 我上学去了 | 我上学去了 |
| | jin tian hui jia bi jiao wan | 今天回家比较晚 | 今天回家比较完 |
| | liang hui zai bei jing zhao kai | 两会在北京召开 | 两会在北京召开 |
| 500 | qing hua da xue ji suan ji xi | 清华大学计算机系 | 清华大学计算机系 |
| | wo shang xue qu le | 我上学去了 | 我上学区了 |
| | jin tian hui jia bi jiao wan | 今天回家比较晚 | 今天回家比较完 |
| | liang hui zai bei jing zhao kai | 两会在北京召开 | 两会在北京召开 |

当 Q_bottom_lim 比较小时，一些出现频率较低，并包含不常见汉字的专有名词，会很容易替代想要的词语。比如当参数小于 300 时，“qing hua”会被转化为“氰化”。而且由于这

个词出现的条件概率接近 1，且“氰”字单独出现的频率并不低，并不能通过光滑处理排除。而提高 Q_bottom_lim 则可以有效避免这一问题。

另外，当 Q_bottom_lim 较大时，又容易将一些本来不是词组的词强行组合。比如当这个参数取 500 时，本来应该是“上学去了”，却被转化成“上学区了”，这是因为当前一个字是“学”时，“学区”这个组合出现的频次很高，而“学去”出现的频次低于限定值，被排除掉了。之后即便“区了”出现的频次远远低于“去了”，也无法得到期望的结果。

由于训练采用的语料库仅来自于 2016 年的新浪新闻，在对常用的、逻辑性较强的短语和短剧进行转换时，得到的结果都比较理想。但其他类型的语句拼音转换则会出现较大的问题。比如文学性较强的歌词类的文本，或者新出的流行词汇，转换得到的结果和原文差别非常大，甚至还会出现多音字错误识别的问题。选取了较新的两首歌的歌词（不可能整句出现在语料库中）进行转换，得到的结果见表 2.3。

表 2.3: 不理想的转换结果

| 原文 | 拼音转换结果 | 文本来源 |
|---|---|---|
| 别犹豫 别偶遇 别相遇 | 别由于 别区域 别向于 | 薛之谦 |
| 别一个人去看喜剧 | 别以个人去看戏剧 | 《别》 |
| 捡起地上的谎 | 捡起的上的黄 | 孙燕姿 |
| 满足了欲望 | 满足了与网 | 《我很愉快》 |

从结果中可以明显看到上面描述的问题。解决这些问题的关键点，是使用更完善的语料库进行训练，语料来源越丰富越好，还可以用包含文字准确发音的语料资源进行训练，从而得到更好的训练文本。

## 3. 总结

总体而言，在有充足的语料库的情况下，利用字的二元模型在大多数情况下可以很好地实现拼音到汉字的转换。语料库的质量会很大程度上影响到转换的效果。另外，补充额外信息，比如多音字在不同情况下的发言，会更有利于提高转换成功率。

这次实验布置得比较早，所以很早之前就开始想要怎么写，从开始写到完成基本的代码大概用了一天半的时间，之后又花了一些时间完善细节，比如单个字的处理、出现非拼音字符串的处理等等。我之前写代码的经验比较少，通过完成这次实验也学习到了不少知识，不仅仅加深了对搜索算法的认识，更学到了许多编程技能。