拼音输入法作业

2021310746-张学峰

November 2, 2021

1 作业介绍

本次作业是实现一个基于字的二元隐马尔可夫模型和 Viterbi 算法的拼音输入法。程序的输入为空格分隔开的拼音串，输出为最大概率的汉字串。另外本次实验的代码中也实现了基于字的三元模型（对应 pinyin.py 中的 Viterbi_3 函数），但是由于边权设置不合理，性能不是很理想，实验报告主要对实现的二元模型进行分析，对三元模型的分析将放在 2.8 节讨论。

2 作业内容

2.1 算法基本思路

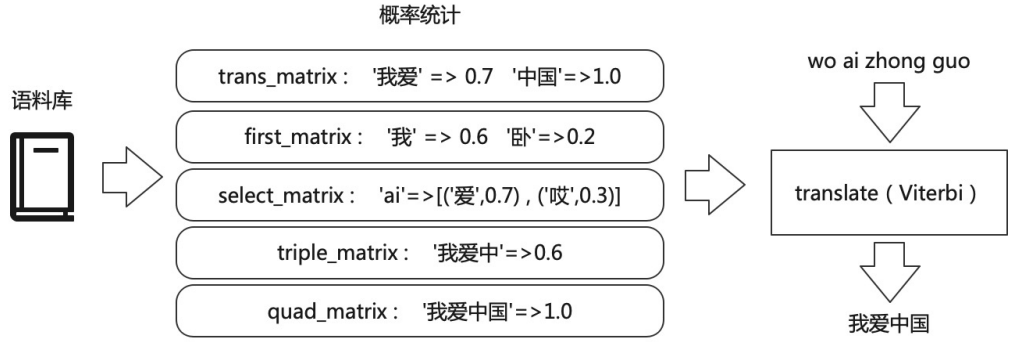

Figure 1: 算法基本思路

首先对作业任务进行分析，制作拼音输入法可以拆分成两个大体步骤，一是通过已有的词库训练出 Viterbi 算法所需要的概率关系，二是对输入的拼音串建图，使用 Viterbi 求出二元模型下概率最大的汉字串。测试性能后，再对二元模型进行改进，测试三元模型的成果。

2.2 算法设计与公式推导

2.2.1 隐马尔可夫模型

隐马尔可夫模型 (Hidden Markov Model) 是一种统计模型，用来描述一个含有隐含未知参数的马尔可夫过程，简单来说就是通过统计已知的观察序列来预测最有可能的隐含序列。

在本次作业中就是使用隐马尔可夫模型，在已知转移概率的基础上进行统计，将拼音串转换为最大概率的汉字。

在拼音输入法中，设 S 为一个句子，$w_0\ w_1...$ 为组成 S 的汉字，那么有：

$$P(S) = \prod_{i=1}^{n} P(w_i|w_0w_1...w_{i-1})$$

所以问题就改变为求解拼音串下的不同汉字组合，选择上式最大的那个汉字串。对于二元语法时，即一个汉字出现的概率只和上一个汉字有关的情况下：

$$P(S) = \prod_{i=1}^{n} P(w_i|w_{i-1})$$
$$P(w_i|w_{i-1}) = \frac{Count(w_iw_{i-1})}{Count(w_{i-1})}$$

因为语料库的很多常用字概率被划分的零零散散，导致计算的得到的概率很小，为了防止在 Viterbi 算法过程中概率越来越小，需要对原来的概率取负对数：

$$P(S) = min(-\sum_{i=1}^{n} log(P(w_i|w_{i-1})))$$

对于一个二元短语，很有可能在训练语库中不存在的，这就需要引入平滑化处理其中 λ 为待定参数。

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

隐马尔可夫模型在本次作业中由三要素组成：

- 第一个是 trans_matrix 记录转移信息，本次作业用到的是最简单的一阶隐马尔科夫模型里，每个汉字的出现只和它前面的的一个汉字有关，但已经可以满足大部分情况。统计的过程就是找出语料库中每个汉字后面出现的汉字集合，并统计概率。例如' 我爱'=0.8　' 中国'=1.0
- 第二个是 select_matrix，记录对于一个给定的拼音,select_matrix 记录了这个拼音下所有的可能汉字被选择的概率例如'ai' => [(' 哎',0.7),(' 爱',0.3)]
- 第三个 first_matrix，记录一个汉字位于句首的概率，因为每个汉字可能在句子中的做的成分不同，一个汉字可能在句子中出现的次数很多但是这个字不易出现在句首，比如说汉字'的'。在 Viterbi 算法的初节点到第一层节点之间的边权，显然设置为 0 和设置为汉字出现概率都不合适，不妨使用 first_matrix。同样还需要考虑的一个问题是，可能测试的真正句首没有在语料库的句首中出现过，这需要 first_matrix 和 select_matrix 做一个平滑处理。select_matrix 的信息例如：(' 哎'=>0.7) 、(' 爱'=>0.3)。
- 同 trans_matrix，本次实验也生成了 triple_matrix、quad_matrix 来测试基于字的三元模型与四元模型，但是性能欠佳。

2.2.2 Viterbi 算法

Viterbi 算法的本质其实是一个动态规划过程，保证当前转移得到的最优解是无后效的。在本作业中的 viterbi 算法的应用就是在使用隐马尔可夫模型建图的基础上计算当前汉字串的最优路径。

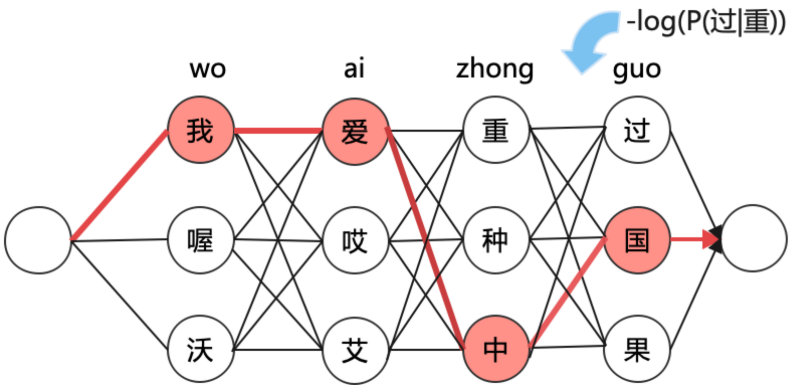

Figure 2: Viterbi 算法

$$P[i,j] = max_k\{P[i-1,k] * P(w_{i,j}|w_{i-1,k})\}$$

其中 P[i,j] 为，到第 i 层候选集合的第 j 个字的最大转移概率，k 为上层节点数，本公式是概率公式，如果使用 2.2.1 所提到的负对数，公式也要跟着修改为求负对数累加和的最小值。

2.3 算法实现

2.3.1 数据预处理

Step 1: 首先读入 sina 新闻语料库，以 gbk 格式解码后使用 utf-8 编码。

Step 2: 按照标点将句子分隔开，分隔开的每一个句子记录统计词频结果。需要统计的词频有：

- 每个汉字以及每个拼音出现的次数
- 每个汉字在句首、以及相应的拼音在句首出现的次数
- 连续两个字出现的次数

Step 3: 由于 2.2.1 小节所提到所需的三个矩阵全部都是非常稀疏的，所以全部采用字典这种数据结构存储。对于 select_matrix 采用一个 {拼音:[(汉字，概率)、[..]..]} 格式的存储结构例如'ai' => [(' 哎',0.7),(' 爱',0.3)]，其中每个汉字的概率计算方法为：

$$P(\text{我}) = \frac{Count(\text{我})}{Count(wo)}$$

first_matrix 同 select_matrix 只不过是句首的统计，统计范围远小于 select 为了更精准确定句首汉字。triple_matrix 和 quad_matrix 的处理方法同 trans_matrix，以 trans_matrix 为例子，统计方法为:

$$P(\text{' 机器'}) = \frac{Count(\text{'机器'})}{Count(\text{'机'})}$$

Step 4: 将预处理结果输出到文件

2.3.2 计算实现

Step 1: 读入预处理的的结果，读入输入文件。

Step 2: 根据存拼音的 list 建图，初始节点出发，每层包含该层拼音对应的所有汉字，每相邻层之间全相连，权值是平滑化处理后的概率取负对数。

Step 3: 在建好的图上使用 Viterbi 算法，并且在转移的过程中记录上一个转移节点以便输出统计。

Step 4: 根据输出和标准输出统计正确率。

2.4 实验

2.4.1 实验环境

本次作业的实验环境如下所示：

CPU	Apple M1
OS	macOS Big Sur 11.5.2
python	python3.9.7
pypinyin	0.43.0
训练语料库	sina news 2016.[2-11]
测试集合	sina news 2021.10

2.5 实验结果

```
qing hua da xue bi bei jing da xue hao
清华大学比北京大学好
zong tai ji zhe shi di tan fang xi ning nong mao shi chang
总台记者实地探访西宁农贸市场
bu wang chu xin lao ji shi ming
不忘初心牢记使命
zhong guo xin wen zhou kan fu ze zhao pin lao shi
中国新闻周刊负责招聘老师
bu lun zhuang bei zhi zao ye hai shi xin neng yuan chan ye
不论装备制造业还是新能源产业
```

Figure 3: 部分测试结果

因为作业中提供的测试数据 input.txt 存在很多奇奇怪怪的句子比如‘我一把把把把住了’、‘八百标兵奔北坡’、‘拒绝内卷’ 等，而训练语料库是选用 2016 年的 sina 新闻，为了测试算法本身的真实效果，本次作业爬取了 2021 年 10 月的 sina 新闻进行测试。

因为 input.txt 存在很多奇怪的句子，对于作业本身提供的 input.txt 平均正确率可以到达 82.6%，而对于自己爬取的 2021 年 10 月 sina 新闻正确率可以到达 92.5%。

测试集	正确率
作业样例 input.txt	82.6%
2021.10 sina news	92.5%

2.6 参数对性能的影响

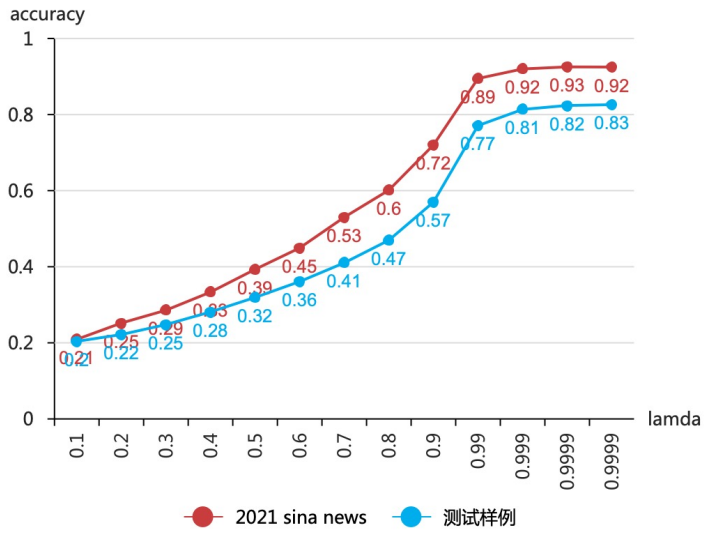

Figure 4: 不同测试集上 lamda 对性能影响 (alpha = 0.7)

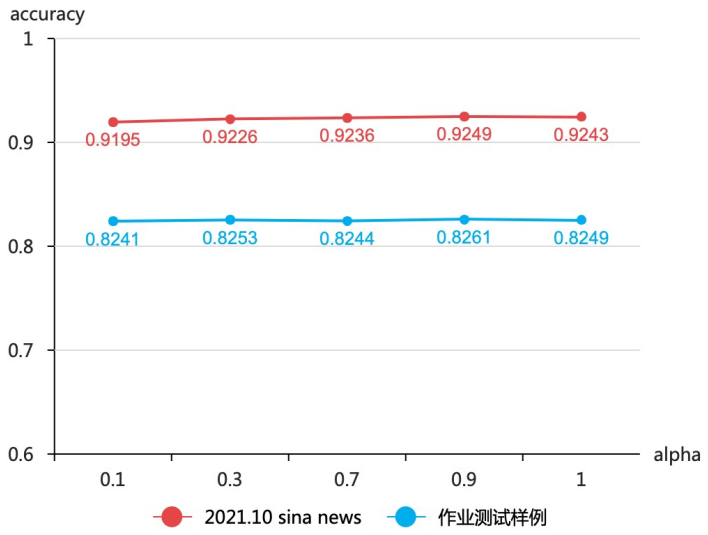

Figure 5: 不同测试集上 alpha 对性能影响 (lamda = 0.99999)

本次实验有两个待定参数，如 2.2.1 小节所属，第一个是为了避免句中的连续两个字没有在语料库中出现而进行平滑化处理设置的参数 lamda。第二个是句首字的平滑化处理 alpha，上图分别为 2021sina news 测试集上两个参数对性能的影响分析。

另外本次实验还测试了在 2021.10 sina news 数据集上不同输入长度对正确率的影响，宏观上

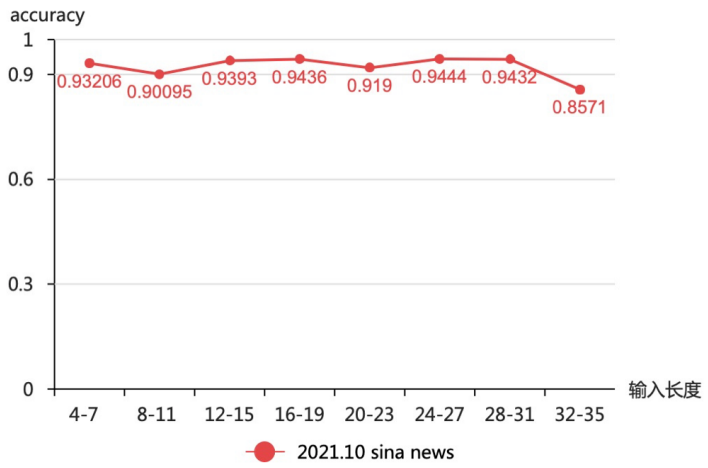

Figure 6: 不同输入长度对性能的影响

看其实句子长度并不会大幅影响正确率，二元语法只考虑相邻两字的相关性，句子越长会导致整句内出现错字的概率变高，但不会影响句子内单个汉字识别的正确率。

2.7 例子与局限性分析

对于大部分句子表现性能较好，且对于部分长句子也有较好的表现，如图 3 所示，本部分主要对错误例子和局限性进行分析。

```
yi qing fang kong
疫情防控
yi qing fang kong ling dao xiao zu
疫情防控领导小组
he suan jian ce
核酸检测
he suan jian ce cheng xian yang xing
核酸检测称咸阳性
yi qing qi jian
以庆期间
yi qing qi jian yao yan ge ba guan
以庆期间要严格把关
e er duo si
厄尔多斯
```

Figure 7: 语料库限制对疫情或地名表现不好

1. 受训练语料库限制的错误，比如本次训练的语料库是 2016 年的新浪新闻摘要，本次作业的测试集从新浪新闻爬下了 2021 年近期的几篇新闻进行测试，发现对于疫情防控等话题翻译效果并不佳，原因就是 2016 年那时候还没有疫情，包括最近’ 鄂尔多斯高薪招聘中小学教师’ 的新闻，因为训练语料库没有鄂尔多斯或者一些地点专用名词，识别难免会错误。2. 句式正确难以避免的错误：

此类错误输出的句式也是正确句式，判断对错主要和输入者的心理有关或者和语境有关。另外因为采用的是简单的二元语法模型，所以就会有局部的用词还挺合理但是整个句子效果就不太

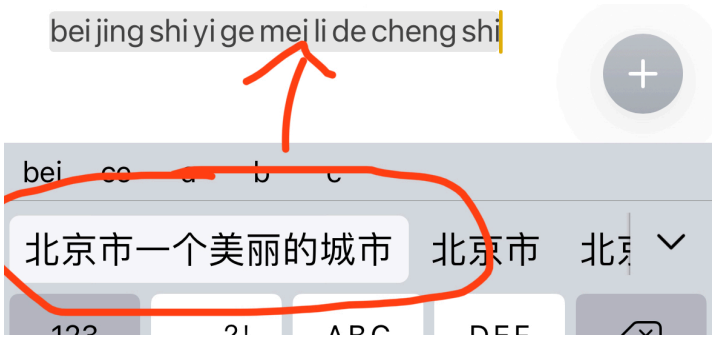

Figure 8: 苹果输入法也有相同错误

Table 1: 句式正确难以避免的错误

Stdout	Output	Accuracy
北京是一个美丽的城市	北京市一个美丽的城市	90%
信息资讯	信息咨询	50.00%
蔬菜经营户岳女士	蔬菜经营户约女士	87.50%
都带编制	都带贬值	50.00%
在近几年转型探索中	在紧急辗转型探索中	66.66%
范明告诉中国新闻周刊	樊铭告诉中国新闻周刊	80.00%
对于近日市场内蔬果价格出现的波动	对于今日市场内蔬果价格出现的波动	93.75%
西宁市民程女士对记者表示	西宁市民称女士对记者表示	91.66%
其他城市未必有这样的底气	其他城市未必有这样的第七	83.33%
获评市级创新创业人才团队	获评实际创新创业人才团队	83.33%
除了高薪招聘教师	除了高新招聘教师	87.5%
出现连锁反应	出现连锁反映	83.33%
平台核心业务至少涉及两类平台业务	平台和新业务至少涉及两类平台业务	87.5%
高薪挖人能否从根本上提升本地教育水平	高新伍人能否从根本上提升本地教育水平	88.88%
五六十万年薪确实非常诱人	五六十万年薪却是非常有人	75%
记者在莫家街综合农贸市场内看到有市民在采买周末用的新鲜食材	记者在莫家界综合农贸市场内看到有市民在采买周末用的新鲜食材	96.5%
他们已经对个别趁疫情哄抬物价的经营户进行了严厉查处和相应处罚	他们已经对个别陈奕清哄抬物价的经营户进行了严厉查处和相应处罚	90%
蔬菜经营户岳女士的摊位经常有外卖前来代买新鲜蔬果配送到家	蔬菜经营户约女士的摊位经常有外卖前来代买新鲜蔬果配送到家	96.4%
但是今天来买菜的人少了	但是今天来买彩的人少了	90.9%
直接吸引发达地区的产业人才可能不现实	直接吸引发达地区的产业人才可能不显示	88.88%

好，还有如果句子中包含人名或者姓氏也比较难判断，这类错误较难更改，但是这类错误的特点是不会造成整个句子的正确率降低太多，如 Table 1 所示，另外我发现苹果的输入法也存在某些错误。

3. 多音字造成的错误，这类错误跟数据预处理部分关系比较大，在处理语料库时无法准确判断一个汉字的多个发音中到底是哪个，例如:" 给予"," 参差"," 银行 (hang)/发行 (xing)"。

Table 2: 多音字错误

Stdout	Output	Accuracy
蔬果自给率不高	蔬果自己率不高	85.71%
参差不齐	岑茨补齐	33.33%
银行发行钞票	引航发行钞票	66.66%

4. 对于二元语法，因为把前后两个字的关联作为主要的判断依据，所以无法判断前后本就联系不大的字，例如下图中的' 但在昨天竟有一百多单' 这一句，其中的 '竟' 和前后联系都不大，所以输出的结果成了' 但在昨天经有一百多单'。如果我们把输入把' 竟' 改为' 竟然' 增加句子的关联度，那么程序就可以完全识别正确。

```
dan zai zuo tian jing you yi bai duo dan
但在昨天经有一百多单
dan zai zuo tian jing ran you yi bai duo dan
但在昨天竟然有一百多单
```

Figure 9: 关联度低造成的错误

Table 3: 单个字关联度过低导致的错误

Stdout	Output	Accuracy
做肯定比不做强	作肯定比不做强	85.71%
但在昨天竟有一百多单	但昨天经有一百多单	90%
西宁市商务局等部门采取措施确保肉菜等生活必需品供应充足	西宁市商务局等部门采取措施确保肉菜等生活必须品供应充足	96.2%

5. 某些句子是古诗或者绕口令，训练语料库没有这些句子会导致准确率非常低。

2.8 模型改进方法与深入思考

- 考虑到一个汉字在句首的概率和在句中的概率不能一视同仁，本次作业将句首汉字概率单独处理为一个矩阵作为初节点到第一层节点的权值，对于不在句首出现的汉字同样采取平滑化处理，测试结果证明确实能对正确性带来提高。
- 基于字的二元模型的最大局限性在于一个字的选择与否只与前一个字有关，本实验尝试了三元模型，但是在边权设置上出现了疑问。对于建好的图，连续三个字在语料库中出现的概率

Table 4: 训练语料库没有出现导致错误

Stdout	Output	Accuracy
遇事不决量子力学	是不觉良资理学	25%
八百标兵奔北坡	把白标兵本被迫	28.4%
校服上除了校徽别别别的	小幅上出了销毁瘪瘪瘪的	27.7%
山有木兮木有枝	山游目系目优质	14.2%
心悦君兮君不知	新月均细菌不知	28.4%

其实并不大，那么就要和两个字连续出现做一个加权平均，同理两个连续字可能也不存在，那么就要（连续三个字、连续两个字、一个字）做加权平均，另外一个问题就是在三元模型下前两层边权的初始化问题。ddl 眼看就要到了，没时间再改了，三元部分测试结果如下。

- 程序的输入只能是相隔的拼音输入，但是人们真正使用过程中是不会打完一个字打空格再打下一个字的，所以我感觉可以用同样的隐马尔可夫 + 动态规划的思想对拼音进行划分。
- 对于拼音不同、差别很大的多音字，可以在数据预处理时进行标注，这样的字本身也不是很多。
- 程序训练所花时间太长，可以根据题材场景将训练语料库分类，然后根据应用场景针对性选择语料库训练，这样可以实现性能和效率的 trade off。

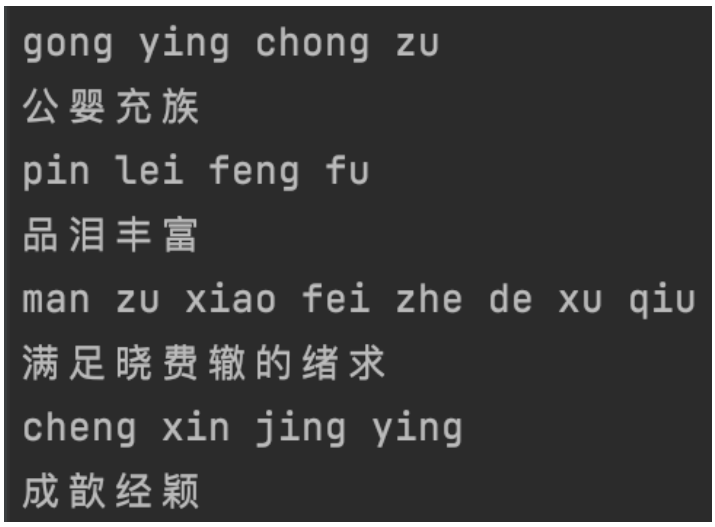

Figure 10: 基于字的三元语法（效果欠佳）

2.9 实验收获

本次实验实现了基于二元隐马尔可夫模型和 Viterbi 算法的拼音输入法制作，本次实验最大的收获就是学习到了马尔可夫过程的思想，同时也了解并实现了简单输入法制作的过程和原理，在实验的最后进行了对现有模型的局限性分析，想要制作一个精细准确的输入法还需要考虑很多问题，通过本次实验真切的感受到了语言处理有很多需要思考和入手的问题比如语句成分分析、筛选等。