人工智能导论
拼音输入法作业报告

高天宇 2016011348
gaotianyu1350@126.com

1 算法简介

我采用的是字的二元、三元模型相结合的算法。

- 分析老师提供的语料库。统计每一个二元组在语料库中出现的次数。
- 取出现次数排名前 10^6 的二元组，统计包含这些二元组的所有三元组在语料库中出现的次数。保留出现次数排名前 10^6 的三元组。
- 进行 DP 算法。令 $f[i][cur][last]$ 表示第 i 个拼音处，汉字为 cur，上一个汉字为 $last$，这种情况出现的最大概率。则在二元模型下，有:

$$f[i][cur][last] = max(f[i-1][last][j] \times p(last, cur))$$

其中 $p(last, cur)$ 表示在上一个字符是 $last$ 的前提下，当前字符为 cur 的概率，有:

$$p(last, cur) = \frac{cnt(last, cur)}{cnt(last)}$$

其中 $cnt(last)$ 表示语料库中 $last$ 出现的次数，$cnt(last, cur)$ 表示 $last$ 和 cur 一起出现的次数。由于可能出现这两项为 0 的情况，我们将上式改成:

$$p(last, cur) = \alpha \cdot \frac{cnt(cur)}{num\_single} + \beta \cdot \frac{cnt(last, cur)}{cnt(last)}$$

其中 num_single 是汉字个数，α，β 是我要自己设置的系数。若 $cnt(last)$ 为 0，则该多项式的第二项取 0。

- 通过二元模型计算出 f 数组后，从 $f[i][cur]$ 中挑选出数值从大到小排名前 NUM_SUM 个 $f[i][cur][last]$。通过多次尝试，平衡运行速度与准确率，我取 $NUM\_SUM = 5$。重新计算这 NUM_SUM 个 $f[i][cur][last]$:

$$f[i][cur][last] = max(f[i-1][last][j] \times p(j, last, cur))$$

其中，$p(j, last, cur)$ 是在上上个字符为 j，上个字符为 $last$ 的前提下，当前字符为 cur 的概率，有:

$$p(j, last, cur) = \frac{cnt(j, last, cur)}{cnt(j, last)}$$

仍然是由于 cnt 可能为 0，则上式应改为:

$$p(j, last, cur) = \alpha \cdot \frac{cnt(cur)}{num\_single} + \beta \cdot \frac{cnt(last, cur)}{cnt(last)} + \theta \cdot \frac{cnt(j, last, cur)}{cnt(j, last)}$$

其中 num_single 的含义，以及分母为 0 情况的处理同二元模型。

- 最终我们可以先找到最大的 $f[n][cur][last]$（n 为拼音序列长度），然后从这里开始，运用递归倒序处理，获取答案序列。

2 效果展示

2.1 好的例子

- mei jun fang cheng bu cheng ren zhong guo dong hai fang kong shi bie qu

  美军方称不承认中国东海防空识别区

- zhong guo guo jia dui zhan sheng han guo guo jia dui

  中国国家队战胜韩国国家队

- zhi neng ji shu yu xi tong guo jia zhong dian shi yan shi

  智能技术与系统国家重点实验室

- shen du shen jing wang luo dui ji suan zi yuan de xiao hao hen da

  深度神经网络对计算资源的消耗很大

- te lang pu xi wang bu jiu he zhong guo guo jia zhu xi mian dui mian hui wu

  特朗普希望不久和中国国家主席面对面会晤

- dui ran se ti ren gong he cheng de gong zuo ji yu le gao du ping jia

  对染色体人工合成的工作给予了高度评价

2.2 不好的例子

- ta yang le yi zhi qing wa dang chong wu

  他养了**一致青瓦**当宠物

- qu xiao huo ting zheng de she ji ge ren deng shi xiang de xing zheng shi ye xing shou fei bao kuo

  取消或**听证**的涉及个人等事项的行政事业性收费包括

- ni de shi jie hui bian de geng jing cai

  你的**时节**会变得更精彩

- zhong guo shi ren min min zhu zhuan zheng de she hui zhu yi guo jia

  中国**诗人**民民主专政的社会主义国家

2.3 缺陷分析

- 语料库的选择问题。由于使用的是新浪新闻，所以会倾向于将词语识别成新闻中出现频率高的词。如“青蛙”被识别为“青瓦”(“青瓦台”出现频率高），“听证”比“停征”比例高，“时节”比“世界”比例高。

- “目光短浅”。在现有规则下每个字只能和前两个字和后两个字发生联系，句子内容无法呼应。如第二句“收费”应该对应“停征”，而非“听证”。
- 句子成分的错误。没有理解每个字/词的词性和在句子中的成分。如最后一句缺少谓语，而且用“诗人”这个名词做了定语。

3 参数调整

3.1 NUM_SUM 调整

NUM_SUM	30	20	5	3
运行时间（每百句）	50s	20s	4s	4s
单字准确率	85.9%	85.7%	85.3%	83.5%

最终选取 $NUM\_SUM = 5$。

3.2 p 计算公式系数调整

只以二元情况为例，取 $\beta = 1 - \alpha$，有

α	10^{-50}	10^{-40}	10^{-30}	10^{-25}	10^{-20}
单字准确率	85.08%	85.08%	85.08%	85.08%	84.99%

最终选取 $\alpha = 10^{-25}$。

4 二元、三元方法对比

首先，三元模型有两种实现形式，一是我上面采取的方法，二是直接省掉二元模型的步骤，用三元模型去计算 f 数组。第二种方法在测试中几乎不可行，速度极慢，并且正确率很低，即使是很简单的句子也无法给出像样的结果。经过分析，主要原因在于收集的三元数据实在太少。

二元与三元的比较（以下识别结果均第一条二元，第二条二元 + 三元）：

- jin yong de wu xia xiao shuo fei chang jing cai

  仅用的武侠小说非常精彩

  金庸的武侠小说非常精彩

- qing da jia xuan ze ni jue de ke yi de shi jian

  请大家选择你觉得可以**地时间**

  请大家选择你觉得可以**的时间**

- xiao peng you men dou xi huan qu jiao you

  小朋友们都喜欢**区矫有**

  小朋友们都喜欢**去郊游**

分析:

算法	二元	二元 + 三元
单字准确率	79.83%	85.08%

可见加入三元算法后，准确率提升了不少。其成功将“的”前的“仅用”矫正为“金庸”，将“时间”前的“地”矫正为了“的”。同时也令常见三字词语的识别率增高，如“去郊游”。

5 总结

通过对不同方法结果的比较，以及对一些识别错误情况的分析，我总结出了以下几点:

- 二元和三元结合效率和准确率最高。
- 如果想要进一步提高准确率，可以尝试
  - 语料库选取应足够大，最好能涵盖各个方面，只选取新闻会有局限性。
  - 录入词典的词库，尝试以“词”为单位而非以“字”为单位。

  - 分析语料库可以尝试不仅分析相邻的字，也分析整句中词语之间的关系。
  - 尝试分析句子中各个词的词性和成分，避免出现一些“荒谬”的错误。

6 备注

- pinyin.py 为二元与三元结合算法，pinyin_easy.py 为二元算法。
- 请直接在 bin 目录下运行 pinyin.py，若在其他目录下运行，可能会出现路径错误的问题。
- pinyin.py 和 pinyin_easy.py 前期加载语料库分析结果的速度较慢，但真正运算的部分较快，请耐心等待。