实验报告

2014080043 计62 姜东槿

算法基本思想

$$P(\text{sentence}) = P(W_1) \times P(W_2|W_1) \times P(W_3|W_1W_2) \times \ldots \times P(W_m|W_{m-N} \ldots W_{m-2}W_{m-1}) \times \ldots$$

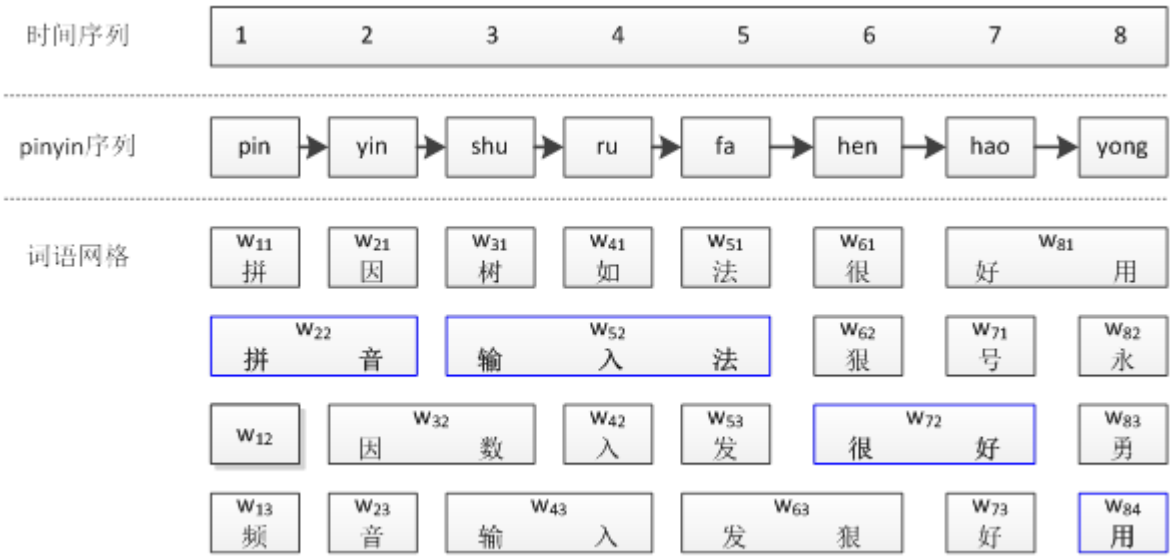
上面的公式中$P(\cdot)$的就是通过 N 元语言模型训练算法得到的参数。

对于二元语言模型（bigram），上面的概率简化为：

$$P(\text{sentence}) = P(W_1) \times P(W_2|W_1) \times P(W_3|W_2) \times \ldots \times P(W_m|W_{m-1}) \times \ldots$$

对于一元语言模型（unigram），上面的概率还可以简化为：

$$P(\text{sentence}) = P(W_1) \times P(W_2) \times P(W_3) \times \ldots \times P(W_m) \times \ldots$$



代码如此

```
For t = 1 : T Do

For i = 1 : M Do
```

```
MaxProb = 0;

Index = 0;

For j = 1 : M Do

ThisProb = Prob[t-1,j]*Bigram(w[t-1,j],w[t,i]);

If ThisProb > MaxProb

MaxProb = ThisProb;

Index = j;

End If

End For

Prob[t,i] = MaxProb;

Ptr[t,i] = j;

End For

End For
```

效果展示

参数：

使用 SinaNews 语料训练，保留最低频率 1e-7
每步保留最优解 10 个

效果好：新闻腔，官方表述
效果不好：不知所云的，专业术语多的

每部取最优解的个数

个数	时间	字距离	句距离
2	18	0.612	0.113
5	44	0.701	0.186
10	73	0.632	0.196
20	153	0.658	0.212

分析： 做的输入法的看训练了什么，因为训练新闻它对日常生活中的语句都不会对政治社会的语句很准，模型效果和训练数据很重要，越高的模型频率越低。

收获机终结：

我的写代码能力很弱，这次作业用了非常多的时间。在网上查看了很多很多的资料，最后还是能够做出了结果（并不是很好），通过这次实验得到了很多写代码和对人工智能的知识，怎样去计算频率，距离等知识。以后更加努力！