马尔科夫链在古诗写作上的应用

黄予 2017210861 计研174

实验内容

机器自动作诗，用户输入第一句诗，机器输出下一句诗。

语料库

诗句对的集合，截取部分如下：

```
朝朝奉御临池上  不羡青松拜大夫
幽人听达曙  相和薛床吟
佳人忆山水  置酒在高台
降集翻翔凤  追攀绝众狙
圣主此时思共理  又应何处救苍生
化城若化出  金榜天宫开
樱桃未绽梅花老  折得柔条百尺长
```

理论根据

记号

- FS：第一句诗（First Sentence）
- SS：下一句诗（Second Sentence）
- f_i：第一句诗中的第i个字（词）
- s_i：下一句诗中的第i个字（词）
- L：诗句长度
- $w_1w_2\dots w_n$：字（词）序列，在计算语言学领域称为$n-gram$
- $C(w_1w_2\dots w_n)$：$w_1w_2\dots w_n$在语料库中出现的频数
- N_{n-gram}：语料库中$n-gram$的总数
- $C_{SS}(s_i)$：s_i在语料库第二句诗中出现的频数

注：由于本实验的实现是基于**字**，也为了说明方便，以下的说明均以**字**为单位。

原理

从概率论的角度，实验任务可描述为在给定第一句诗FS时，求条件概率最大的第二句诗SS，即求

$$\underset{SS}{argmax}\, P(SS|FS)$$

由贝叶斯公式，

$$P(SS|FS) = \frac{P(FS|SS)P(SS)}{P(FS)} \propto P(SS)P(FS|SS)$$

在上式中，由于FS已经给定，故略去；$P(SS)$表示下一句诗的概率，该式称为语言模型；$P(FS|SS)$表示给定第二句诗时生成第一句诗的的条件概率，该式称为翻译模型。将语言模型继续分解：

$$P(SS) = P(s_1 s_2 \dots s_L) = P(s_1)P(s_2|s_1)P(s_3|s_1 s_2)\dots P(s_n|s_1 s_2 \dots s_L)$$

从语言学的角度，句子中的字一般只与其附近的字有关，例如 `我爱吃红苹果` 一句，如果把 `果` 字拿掉，变成 `我爱吃红苹_` ，那么即使我们没有看到 `我爱吃` 三个字，而是仅看到 `红苹` ，我们也可以推出下一个字应为 `果` 。因此，我们假设诗句中的第i个字只受前$n-1$个字约束，即：

$$P(s_i|s_1 s_2 \dots s_{i-1}) \approx P(s_i|s_{i-n+1} \dots s_{i-1})$$

进而，

$$P(SS) = P(s_1 s_2 \dots s_L) \approx \prod_{i=1}^{L} P(s_i|s_{i-n+1} \dots s_{i-1})$$

其中当 $j < 1$ 时认为 s_j 为空，例如$P(s_2|s_{-1}s_0 s_1) = P(s_2|s_1)$。上述语言模型在计算机领域称为 $n-gram$ 语言模型，特别地，当$n = 2$时，

$$P(SS) = P(s_1 s_2 \dots s_L) \approx \prod_{i=1}^{L} P(s_i|s_{i-1}) = P(s_1)P(s_2|s_1)P(s_3|s_2)\dots P(s_L|s_{L-1})$$

即诗句第 i 个字只与第 $i-1$ 有关，而与前面的字无关，该诗句的生成过程就是一个马尔科夫链。各个“子条件概率”的计算如下：

$$P(s_n|s_1 \dots s_{n-1}) = \frac{C(s_1 \dots s_n)}{C(s_1 \dots s_{n-1})}$$

$$P(s_i) = \frac{C(s_i)}{N_{1-gram}}$$

其中，$C(s_1 \dots s_n)$指在$s_1 \dots s_n$语料库中出现的频数，N_{1-gram}指语料库中的 $1-gram$ 总数，即总字数。

同样地，翻译模型$P(FS|SS)$可分解为：

$$P(FS|SS) = P(f_1 f_2 \dots f_n|s_1 s_2 \dots s_n) \approx \prod_{i=1}^{n} P(f_i|s_i)$$

$$P(f_i|s_i) = \frac{C(match(f_i, s_i))}{C_{SS}(s_i)}$$

其中，$C_{SS}(s_i)$指s_i在语料库的所有第二句诗中出现的频数，$C(match(f_i, s_i))$指f_i与s_i在语料库中的匹配次数。

计算机程序

- 见code文件夹

运行

在 `code` 文件夹下启动命令行，输入命令：

```
python SMT.py
```

回车即可运行，用户可依次输入第一句诗，程序会自动给出下一句诗，如下：

```
Last login: Sun Dec 24 17:23:15 on ttys004
HYdeMacBook-Pro:code apple$ python SMT.py
Reading data...
Training...
Training Language Model...
Traning Translation Model...
输入诗句: 缀帘金翡翠
输出诗句: 宫殿玉鸳红
输入诗句: 不堪明月里
输出诗句: 不可入云中
输入诗句: 祝老师万事如意
输出诗句: 流年是千年似情
输入诗句: exit
HYdeMacBook-Pro:code apple$
```

结果与分析

语料库切分为训练集和测试集，训练集用于训练模型，测试集用于测试。模型在测试集上的结果截取如下，格式为 `第一句诗 模型根据第一句诗的输出 |Ref:第一句诗原来所对的诗` ：

```
此日令人肠欲断 何时见我眼初开 |Ref:不堪将入笛中吹
弄闲时细转 寻静处轻轻 |Ref:争急忽惊飘
明月峡添明月照 白云峰减暗风吹 |Ref:蛾眉峰似两眉愁
世危肯使依刘表 生古障令仿谢溟 |Ref:山好犹能忆谢公
宁愬羽觞迟 不仁鳞句疾 |Ref:惟欢亲友会
胡雁哀鸣夜夜飞 楚猿悲宿朝朝落 |Ref:胡儿眼泪双双落
独有成蹊处 多无作堤时 |Ref:秾华发井傍
别马连嘶出御沟 行人断伯入宫桑 |Ref:家人几夜望刀头
枝逐清风动 叶随白露生 |Ref:香因白雪知
蒐于岐阳骋雄俊 术于违阴当雌英 |Ref:万里禽兽皆遮罗
影摇江汉路 声入塞胡尘 |Ref:思结潇湘天
五侯轩盖行何疾 千里客旌坐几迟 |Ref:零陵太守登车日
雪貌潜凋雪发生 云心暗损冰颜死 |Ref:故园魂断弟兼兄
远山应见繁华事 寒水不闻喧彩情 |Ref:不语青青对水流
殷勤照永夜 莫学吹式秋 |Ref:属思未成眠
还希驻辇问 更问停宫知 |Ref:莫自叹冯唐
新秋日后晒书天 旧春风前洗酒地 |Ref:白日当松影却圆
荀谢年何少 胡为岁几多 |Ref:韦平望已久
凉为开襟至 秋作回首犹 |Ref:清因作颂留
遥遥望左右 远近听右左 |Ref:日入未回车
三展蜀笺皆郢曲 九成秦笔尽湘弦 |Ref:我心珍重甚琼瑶
居士近依僧 寂寥深逐客 |Ref:青山结茅屋
远岸牧童吹短笛 孤舟移女湿长砧 |Ref:蓼花深处信牛行
伤哉绝粮议 感激多椽朋 |Ref:千载误云云
...
```

可以看到，很多诗句对的还是不错的，例如`枝逐清风动 叶随白露生`，`三展蜀笺皆郢曲 九成秦笔尽湘弦`等等

注：完整结果见附件`poemSMT_lmn3_sm0.100_lmw0.300_be10.txt`

Reference

- Ming Zhou, Long Jiang, and Jing He. 2009. Generating Chinese Couplets and Quatrain Using a Statistical Approach.
- Philipp Koehn, Franz Josef Och, Daniel Marcu. 2003. Statistical Phrase-Based Translation