IBM翻译模型作业报告

人工智能82班 刘志成 2183511589

相关代码已上传至我的github仓库：
https://github.com/zchliu/2020-Fall

问题引入

使用IBM model1和IBM model2求解最优对齐，输出为每一个中文单词及其对应的对其单词，语料库位于：
http://www.datatang.com/data/14779/

计算模型

IBM model1

EM算法用极大似然估计法来计算每一个英文词所对应的中文词，在每一轮迭代中更新词对的概率值，直到收敛。

IBM model1先用EM算法求出每个英文词的对应中文词的概率，在生成对应关系的时候给定一个英文词，遍历每一个英文词所对应中文词的概率，找到其最大值所表示的词即为对应的中文词

伪码如下：

```
input:set of sentence pairs (e,f)
output:translation prob t(e|f)

initialize t(e|f) uniformly
while not converged do:
    count(e|f) = 0 for all e,f
    total(f) = 0 for all f
    for all sentence pairs (e,f) do:
        for all words e in e do:
            s-total(e) = 0
            for all words f in f do:
                s-total(e) += t(e|f)
            end for
        end for

        for all words e in e do:
            for all words f in f do:
                count(e|f) += t(e|f) \ s-total(e)
                total(f) += t(e|f) \ s-total(e)
            end for
        end for
    end for

    for all foreign words f do:
        for all english words e do:
            t(e|f) = count(e|f) / total(f)
        end for
    end for
end while
```

假设有两个词条如下：

- 他吃狗 -- he eat dog
- 他吃猪 -- he eat pig

初始化矩阵为$t(e|f) = 1$

在每次迭代时$count(e|f) = 0, total(f) = 0$

t	he	eat	dog	pig
他	1	1	1	1
吃	1	1	1	1
狗	1	1	1	1
猪	1	1	1	1

第一次迭代

count	he	eat	dog	pig
他	0	0	0	0
吃	0	0	0	0
狗	0	0	0	0
猪	0	0	0	0

经过第一个词条

count	he	eat	dog	pig
他	0.33	0.33	0.33	0
吃	0.33	0.33	0.33	0
狗	0.33	0.33	0.33	0
猪	0	0	0	0

经过第二个词条

count	he	eat	dog	pig
他	0.66	0.66	0.33	0.33
吃	0.66	0.66	0.33	0.33
狗	0.33	0.33	0.33	0
猪	0.33	0.33	0	0.33

更新t矩阵

t	he	eat	dog	pig
他	0.33	0.33	0.17	0.17
吃	0.33	0.33	0.17	0.17
狗	0.33	0.33	0.33	0
猪	0.33	0.33	0	0.33

第二次迭代

经过第一个词条

count	he	eat	dog	pig
他	0.33	0.33	0.25	0
吃	0.33	0.33	0.25	0
狗	0.33	0.33	0.5	0
猪	0	0	0	0

经过第二个词条

count	he	eat	dog	pig
他	0.66	0.66	0.25	0.25
吃	0.66	0.66	0.25	0.25
狗	0.33	0.33	0.5	0
猪	0.33	0.33	0	0.5

更新t矩阵

t	he	eat	dog	pig
他	0.36	0.36	0.14	0.14
吃	0.36	0.36	0.14	0.14
狗	0.29	0.29	0.42	0
猪	0.29	0.29	0	0.42

问题为求解给定一个中英文语料库的对齐方式，假设$\vec{a}$表示一种对齐方式，$\vec{e}$为英文句子，$\vec{f}$ 为中文句子，推导如下

$$\begin{aligned} & P(\vec{a}|\vec{f},\vec{e}) \\ &= \alpha P(\vec{a},\vec{e}|\vec{f}) \\ &= \alpha P(\vec{e}|\vec{a},\vec{f})P(\vec{a}|\vec{f}) \end{aligned} \tag{1}$$

在IBM model1中，$P(\vec{a}|\vec{f})$为相等值，因此，上式为：

$$\begin{aligned} &\alpha P(\vec{e}|\vec{a},\vec{f}) \\ &= \alpha \sum_i P(e_i|f_{a_i}) \end{aligned} \tag{2}$$

IBM model2

IBM model2 在IMB model1 的基础上考虑了中文和英文对应句子的长度以及两个句子中每一个词的位置关系，用q矩阵来描述，在生成对应关系的时候和IBM model1相同。

伪代码如下：

```
input:set of sentence pairs (e,f)
output:translation prob t(e|f)

initialize t(e|f) uniformly
initialize q(k,j,l,m) uniformly
while not converged do:
    count(e|f) = 0 for all e,f
    total(f) = 0 for all f
    count_num(k,j,l,m) = 0 for all e,f
    total_num(j,l,m) = 0 for all f
    for all sentence pairs (e,f) do:

        for all words e in e do:
            for all words f in f do:
                denomitor = denomitor + q(k,j,l,m) * t(e|f)
            for all words f in f do:
                delta = q(k,j,l,m) * t(e|f) / denomitor
                count(e|f) += delta
                total(f) += delta
                count_num(k,j,l,m) += delta
                total_num(j,l,m) += delta
            end for
        end for
    end for

    for all foreign words f do:
        for all english words e do:
            t(e|f) = count(e|f) / total(f)
            q(k,j,l,m) = count_num(k,j,l,m) / total_num(j,l,m)
        end for
    end for
end while
```

模型评估

总的来说，在实践中发现，IBM model2 比 IBM model1 有更好的表现。大概可以分成三种情况：

1. 在简单句子中，对应关系比较少，IBM model1 和 IBM model2 表现得同样好

比如句对：Do you eat? -- 你 吃 了 吗？

IBM model1	IBM model2
Do: 吗 you: 你 eat: 吃	Do: 吗 you: 你 eat: 吃

2. 一般来说，排在相同位置的词更容易产生对应，因此，在较为复杂得句子中，由于IBM model2 考虑到了词的长度以及在句子中位置的因素，产生的对应效果会更好些。

比如句对 Boys are your hands clean, 孩子 们 你们 的 手 干净 吗？

IBM model1	IBM model2
Boys: 干净 are:们 your: 你们 hands: 手 clean: 干净	Boys: 孩子 are:孩子 your: 你们 hands: 手 clean: 干净

3. 在更复杂的句子中，即使考虑到了词的位置关系，两个模型的效果也都不好，这时候只能考虑用更复杂的模型来训练

A science fiction cannot be regarded as a mere entertainment, but in fact it tells the reader much more, 科幻小说 不能 简单 地 看成是 供 消遣 的 ， 而 实际上 它 给 读者 展示 更 深刻 的 内容

IBM model1	IBM model2
A: 展示 science: 科幻小说 fiction: 科幻小说 cannot: 不能 be: 供 regarded: 科幻小说 as: 看成是 a: _ mere: 科幻小说 entertainment: 消遣 but: 科幻小说 in: 实际上 fact: 实际上 it: 它 tells: 科幻小说 the: 地 reader: 读者 much: 科幻小说 more: 更	A: 科幻小说 science: 科幻小说 fiction: 科幻小说 cannot: 不能 be: 不能 regarded: 的 as: 展示 a: 看成是 mere: _ entertainment: 科幻小说 but: 实际上 in: 深刻 fact: 实际上 it: 它 tells: 给 the: _ reader: 读者 much: 实际上 more: 更

最后，可以发现两个模型在有关虚词的对应上都做得不好，这可能跟虚词占词条的比重大有关