计算机科学与技术学院神经网络与深度学习课程实验报告

实验题目：RNN		学号：201918130222
日期：2021/11/28	班级： 智能	姓名： 魏江峰
Email：2257263015@qq.com		

实验目的：

学习 RNN

实验软件和硬件环境：

硬件环境：

处理器：Intel core i7 9750-H

电脑：神州 z7m-ct7nk

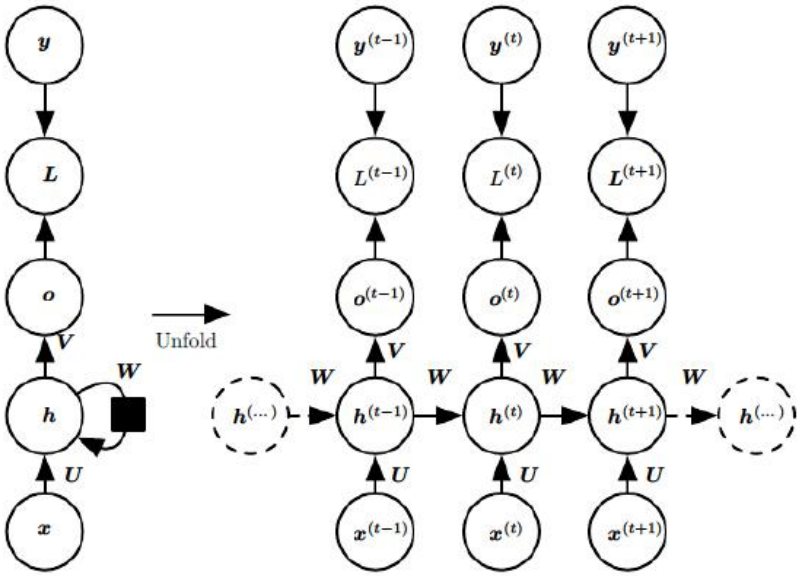
软件环境：

Pycharm 与 jupyter notebook

实验原理和方法：

我们从基础的神经网络中知道，神经网络包含输入层、隐层、输出层，通过激活函数控制输出，层与层之间通过权值连接。激活函数是事先确定好的，那么神经网络模型通过训练“学“到的东西就蕴含在“权值“中。

基础的神经网络只在层与层之间建立了权连接，RNN 最大的不同之处就是在层之间的神经元之间也建立的权连接。如图。



这是一个标准的 RNN 结构图，图中每个箭头代表做一次变换，也就是说箭头连接带有权值。左侧是折叠起来的样子，右侧是展开的样子，左侧中 h 旁边的箭头代表此结构中的“循环“体现在隐层。

在展开结构中我们可以观察到，在标准的 RNN 结构中，隐层的神经元之间也是带有权值的。也就是说，随着序列的不断推进，前面的隐层将会影响后面的隐层。图中 O 代表输出，y 代表样本给出的确定值，L 代表损失函数，我们可以看到，“损失“也是随着序列的推荐而不断积累的。

实验步骤：（不要求罗列完整源代码）

Part1:

1，RNN 语言模型对其输出分布使用 softmax 激活函数

在每个时间步。可以通过将 logits 乘以 a 来修改分布

常数α:

y = softmax(α z)

在这里，1/α 可以被认为是一个“温度”，即较低的 α 值对应于

“更热”的分布。其实，α 越大，概率分布越分散，α 越小，概率分布 越集中。，

2，**sample 函数：输入一个字母的索引 seed_ix，利用当前 RNN 模型，根据该字母创建整个句子，然后返回句子中出现的字母对应的索引列表 ixes**

```
def sample(h, seed_ix, n, alpha):
    # Start Your code
    x = np.zeros((vocab_size, 1))
        x[seed_ix] = 1
    ixes = []
    for t in range(n):
        h = np.tanh(np.dot(Wxh, x) + np.dot(Whh, h) + bh)
        y = np.dot(Why, h) + by
        p = np.exp(alpha * y) / np.sum(np.exp(alpha * y))
        ix = np.random.choice(range(vocab_size), p=p.ravel())
         x = np.zeros((vocab_size, 1))
         x[ix] = 1
        ixes.append(ix)
    return ixes
    # End your code
```

3，comp 函数：给定一个长度为 m 的字符串，将长度为 n 个字符的字符串补全

```
    y = np.dot(Why, h) + by
    p = np.exp(y) / np.sum(np.exp(y))
```

```
    ix = np.random.choice(range(vocab_size), p=p.ravel())
    x = np.zeros((vocab_size, 1))
    x[ix] = 1
```

4，结果：

当α = 5：

irst Senater and the the the the the the the the the shall the the the the the son and the the the the the the the we the the the the the the

the count the the would the so the the the the the the the the so so t----

irst Senater and the the the the the the the the the shall the the the the the son and the the the the the the the we the the the the the the

the count the the would the so the the the the the the the the so so t

当α = 1 ：

irst Cough movble thention-word hour had, gequin' the that your so barted Jake crow,--

mys's they as the pay, that pamantaius faude in a for labort gur, vifle befe,--thenes, whero dis sharked urril'd

----

irst Cough movble thention-word hour had, gequin' the that your so barted Jake crow,--

mys's they as the pay, that pamantaius faude in a for labort gur, vifle

befe,--thenes, whero dis sharked urril'd

当α = 0.1 ：

O'DAWQP:UOoHWvudh.otggloq!bwm,kle!dbnlKz:mG sJty

gioraVeyy vlSCB&sour MutiuspeBngy do,Cnr

'ruoa'??o'tkpe,?YchdIN

WcuqEpBr!:v;xiV-HlyVI.cg.

Kalu;s tVQC

NF:delb

'pmAJjEub;gs?NOn cu,rxigeqtewh LanKlewin ----

O'DAWQP:UOoHWvudh.otggloq!bwm,kle!dbnlKz:mG sJty

gioraVeyy vlSCB&sour MutiuspeBngy do,Cnr

'ruoa'??o'tkpe,?YchdIN

WcuqEpBr!:v;xiV-HlyVI.cg.

Kalu;s tVQC

NF:delb

'pmAJjEub;gs?NOn cu,rxigeqtewh LanKlewin

Part2:

在已经给定了一段 string 之后，生成后续看似合理的连续文本。为了能够这么做，要计算起始字符串的末尾字符的 hiddenn state。然后生成新的文本。

```
for t in range(m):
```

```
        # Start Your code
        # -------------------------
        h = np.tanh(np.dot(Wxh, x) + np.dot(Whh, h) + bh)
        ix = inputs[word_index + 1]
        word_index += 1
        x = np.zeros((vocab_size, 1))
        x[ix] = 1
        ixes.append(ix)
        # End your code
# Start Your code
  y =     np.dot(Why, h) + by
  p = np.exp(y) / np.sum(np.exp(y))
  ix = np.random.choice(range(vocab_size), p=p.ravel())
  x = np.zeros((vocab_size, 1))
  x[ix] = 1
```

当 m = 780，n = 200

Context:

---

rvee

Thus the of fon but to steicpedy.

SICINIUS:

Atsie of ko Sess you, as eim them.

SICINIUS:

Fut a tullss the the no rey se's opt come have a word it are eaving leads eat, him

Trates store to of theim me prace firte enemime bood, I this geed.

SICINIUS:

Aff, afp.

.....

当 m = 50 ， n = 500

Context:

---

lde ing,

Your Citizenseshor that a caders, would fo

Continuation:

---

ly my mothind

'fayizeas?

VOLUMNIA:

O cright; would fach you.

.....

当 m = 2，n = 500

Context:

---

r,

Continuation:

---

trit,

How, and not on day.

COOLOt: I have pait,

I lave were to had exeing Vage.

As all he man: onged trees my our's wient him, that long:

Milowh send slalled's the the not i my him we the the with a must should a ongenitor do and hiles that eyest hel hoen an he the so quenes dint rvee, go'an a be oaal froad apo, tide the of beguon searett.

Part3:

当 current char 是“：” 的时候，会有很大的概率出现“ ”或者是 newline。

现在不妨先生成一个 one-hot 编码的“：”的表示

x = np.zeros((vocab_size, 1))

x[char_to_ix[':']] = 1

经过查证 x = 9；

‘ ‘ 对应的是第 3 个元素表示成 1

‘\n’ 对应的是第 1 个元素表示成 1

并且 x 表示成：一个只有第 10 个元素是 1 的，总共 62 维度的向量。

所以 weight matrix “WXh” 和 x 的点乘法的时候：是该矩阵的第 10 列在起作用。之所以会产生这种效果（在“：“之后总是出现 ‘ ’ 或者是 newline ） 是因为经过 softmax 之后，output 的第一个位置或者是第 3 个位置的值最大，即概率最大，此时就会产生上述的情况。

结论分析：

α 增大，概率分布趋于平滑，后面词出现的概率趋于相同，训练模型时出现次数频繁的词耿荣服役被选择。

α 减小， 概率分布愈发尖锐，神经网络的聚鼎鑫增大。

循环神经网络可以往前看任意多个输入值。但有时这样会存在问题。

就实验过程中遇到和出现的问题，你是如何解决和处理的，自拟 1 - 3 道问答题：