

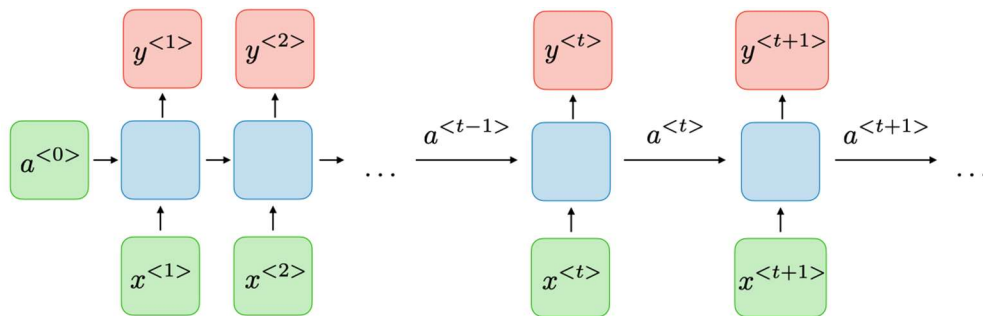
# 神经网络与深度学习第 4 次作业 实验报告

2153409 李玉泉

## 1 三种网络结构介绍

### 1.1 循环神经网络 (RNN)

RNN 是最基本的序列模型之一，其主要思想是通过循环结构对序列信息进行处理。RNN 的每个时间步都会接受一个输入和前一个时间步的隐藏状态，并生成一个输出和一个新的隐藏状态。这种结构允许信息在时间上进行传递，因此适用于处理时间序列数据。基本结构如下：

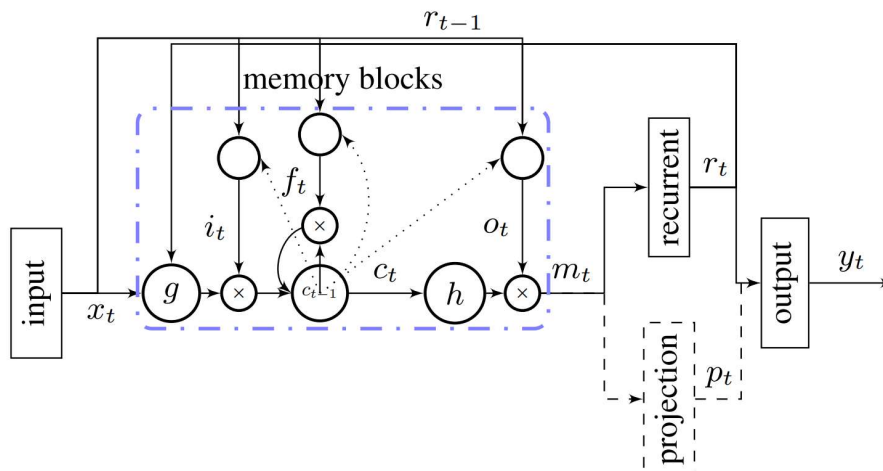


对于输入序列  $\mathbf{x} = (x^{<1>}, x^{<2>}, \dots, x^{<t>}, x^{<t+1>}, \dots)^T$ ，RNN 选取初始的隐藏状态  $a^{<0>}$ ，依次将隐藏状态与输入序列中的 token 经过多层感知机，得到第一个输出  $y^{<1>}$ ，再使用上一次的输出 token 更新 RNN 的隐藏状态，输入序列的每个 token 流经同一个多层感知机依次得到输出序列：

$$\mathbf{y} = (y^{<1>}, y^{<2>}, \dots, y^{<t>}, y^{<t+1>}, \dots)^T$$

### 1.2 长短期记忆网络 (LSTM)

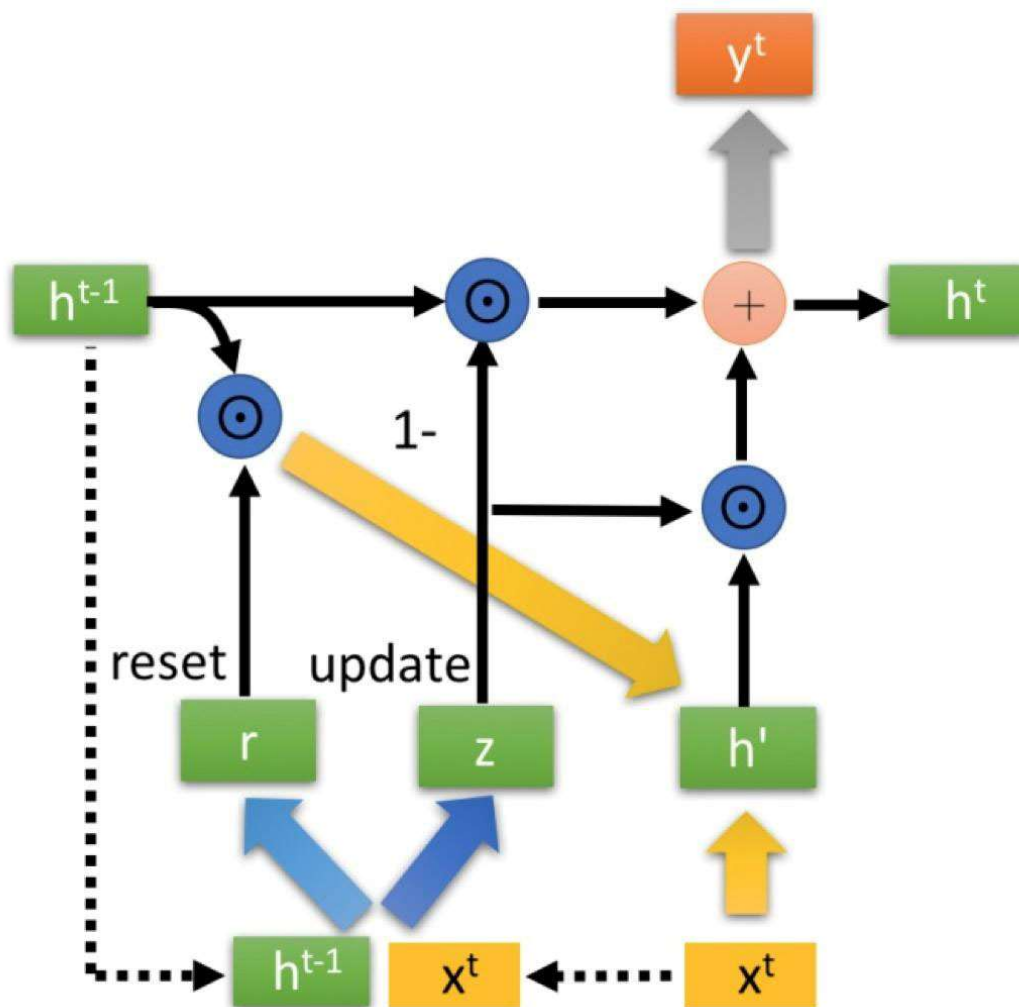
LSTM 是为了解决 RNN 中梯度消失和梯度爆炸问题而提出的。它引入了门控机制，包括输入门、遗忘门和输出门，以及一个记忆单元，用来控制信息的流动。通过这些门控制机制，LSTM 能够更有效地捕捉长期依赖关系。输入门决定哪些信息可以进入记忆单元，遗忘门决定哪些信息可以被遗忘，输出门决定哪些信息可以输出给下一层或下一个时间步。基本结构如下：



### 1.3 门控循环单元（GRU）

GRU 是另一种用来解决梯度消失和梯度爆炸问题的 RNN 变体。与 LSTM 类似，GRU 也引入了门控制机制，包括更新门和重置门。更新门控制着信息的流动，决定了前一时刻的隐藏状态如何与当前的输入进行整合，以及新的信息应该流入隐藏状态的多少。重置门则决定了如何使用历史信息，它帮助网络决定多少历史信息应该被忽略，从而减少梯度消失问题。

但是相较于 LSTM，GRU 的结构更加简单，去掉了 LSTM 中的记忆单元，减少了参数数量和计算复杂度。虽然 GRU 比 LSTM 更简单，但在某些任务上表现也同样优秀。基本结构如下：



## 2 生成诗歌过程

本实验的诗歌生成过程是先将开头字符输入到模型，再依次将得到的输出字符作为下一次输入继续迭代，直到输出训练过程中预先规定好的结束 token 为止，认为生成过程结束。

具体实现方法是：首先将希望作为开头的汉字字符单独作为输入序列，输入到训练好的模型中，模型会得到与输入序列长度相同的输出序列，也就是单个字符的输出序列，输出序列的最后一个字符认为是模型预测的结果，拼接到输入序列之后，作为下一次迭代的输出序列，重复这个过程，直到模型的预测结果是规定好的结束 token，则认为生成诗歌的过程结束。

### 3 诗歌生成结果

本实验完成了 Pytorch 版的代码，模型训练过程截图如下：

```
终端 1 × main2.py × main.py × rnn.py ×
b_y [318, 318, 201, 74, 85, 0, 375, 186, 261, 60, 13, 1, 110, 1295, 115, 283, 28, 0, 622, 713, 1187, 770, 471, 1, 8, 196, 200, 54, 707, 0, 7, 3
44, 99, 123, 714, 1, 1307, 137, 277, 452, 107, 0, 371, 690, 987, 347, 76, 1, 111, 38, 105, 2450, 393, 0, 99, 609, 1182, 2989, 121, 1, 101, 324, 214, 6
92, 789, 0, 46, 187, 61, 2528, 5, 1, 3, 3]
*****
epoch 29 batch number 343 loss is: 6.237895488739014
prediction [10, 318, 179, 318, 393, 0, 46, 11, 91, 16, 115, 1, 10, 296, 11, 57, 17, 0, 83, 74, 99, 9, 13, 1, 72, 303, 617, 119, 8, 0, 46, 7, 311, 145,
46, 1, 7, 12, 546, 74, 8, 0, 12, 18, 10, 11, 134, 1, 4, 19, 9, 47, 42, 0, 9, 11, 117, 729, 30, 1, 3, 21, 4, 54, 41, 0, 15, 64, 49, 12, 281, 1, 3, 3]
b_y [1279, 1279, 1310, 154, 72, 0, 206, 206, 43, 293, 1497, 1, 2085, 3029, 452, 209, 8, 0, 60, 237, 92, 546, 13, 1, 1219, 983, 7, 14, 177, 0, 2
20, 75, 964, 145, 76, 1, 137, 78, 444, 309, 245, 0, 178, 1762, 392, 87, 535, 1, 678, 357, 28, 260, 114, 0, 857, 11, 99, 1110, 121, 1, 6, 1523, 2101, 4
, 51, 0, 35, 1775, 644, 922, 1979, 1, 3, 3]
*****
epoch 29 batch number 344 loss is: 6.174890518188477
prediction [10, 7, 19, 44, 159, 0, 39, 406, 43, 22, 84, 1, 4, 17, 7, 62, 29, 0, 6, 330, 17, 54, 79, 1, 7, 32, 4, 31, 8, 0, 6, 45, 31, 144, 22, 1, 8, 4
8, 7, 9, 70, 0, 5, 21, 8, 31, 46, 1, 7, 228, 9, 34, 17, 0, 7, 18, 117, 15, 88, 1, 39, 17, 26, 119, 74, 0, 7, 30, 10, 39, 21, 1, 3, 3]
b_y [196, 330, 69, 44, 70, 0, 581, 451, 18, 714, 457, 1, 8, 169, 5, 24, 73, 0, 6, 40, 100, 54, 626, 1, 185, 11, 32, 1148, 1148, 0, 249, 16, 63,
236, 236, 1, 118, 48, 168, 450, 378, 0, 6, 21, 101, 289, 615, 1, 22, 50, 259, 232, 252, 0, 33, 364, 117, 93, 594, 1, 233, 161, 88, 201, 943, 0, 86, 3
0, 43, 39, 632, 1, 3, 3]
*****
epoch 29 batch number 345 loss is: 6.337967395782471
prediction [10, 18, 13, 186, 70, 0, 84, 8, 37, 163, 7, 1, 10, 52, 117, 57, 42, 0, 10, 49, 10, 20, 32, 1, 8, 72, 28, 7, 16, 0, 6, 340, 10, 74, 178, 1,
82, 23, 5, 5, 28, 0, 83, 178, 4, 17, 178, 1, 10, 27, 9, 393, 534, 0, 4, 49, 10, 178, 20, 1, 3, 488, 9, 4, 42, 0, 42, 0, 10, 178, 4, 96, 5, 123, 5, 1,
3, 3]
b_y [457, 152, 475, 147, 70, 0, 223, 8, 37, 313, 1117, 1, 302, 457, 231, 652, 1824, 0, 222, 704, 70, 147, 5, 1, 457, 48, 84, 88, 16, 0, 201, 72
, 1766, 372, 1209, 1, 88, 16, 112, 594, 93, 0, 12, 57, 56, 23, 79, 1, 149, 278, 80, 309, 2542, 0, 28, 175, 254, 217, 1113, 1, 271, 488, 450, 135, 43,
15, 42, 0, 12, 65, 564, 475, 540, 157, 29, 1, 3, 3]
*****
epoch 29 batch number 346 loss is: 6.445919513702393
prediction [10, 23, 179, 11, 0, 8, 48, 0, 10, 87, 46, 117, 10, 178, 48, 1, 10, 11, 19, 34, 21, 16, 74, 0, 10, 7, 79, 7, 48, 44, 107, 1, 30, 62, 4, 8,
16, 41, 41, 0, 84, 11, 4, 48, 4, 26, 104, 1, 30, 5, 4, 163, 32, 80, 393, 0, 10, 117, 15, 82, 4, 268, 107, 1, 3, 92, 5, 151, 56, 0, 29, 0, 10, 62, 4, 3
7, 9, 6, 330, 1, 3, 3]
b_y [188, 682, 804, 242, 143, 242, 70, 0, 10, 242, 297, 41, 10, 242, 100, 1, 342, 71, 191, 6, 240, 60, 154, 0, 555, 85, 1335, 1335, 69, 97, 107
, 1, 139, 284, 325, 196, 4, 149, 95, 0, 36, 11, 99, 73, 144, 37, 381, 1, 38, 5, 112, 163, 1572, 241, 867, 0, 182, 7, 1076, 1076, 243, 134, 434, 1, 441
, 70, 296, 457, 869, 108, 20, 0, 729, 131, 907, 117, 66, 6, 539, 1, 3, 3]
*****
epoch 29 batch number 347 loss is: 6.312870502471924
error
```

```
终端 1 × main2.py × main.py × rnn.py ×
epoch 29 batch number 342 loss is: 6.187593460083008
prediction [10, 318, 154, 74, 48, 0, 10, 186, 82, 13, 154, 1, 10, 41, 5, 13, 8, 0, 4, 37, 4, 10, 379, 1, 4, 119, 1742, 17, 177, 0, 7, 592, 16, 15, 12,
1, 10, 74, 57, 7, 393, 0, 7, 178, 8, 34, 28, 1, 47, 466, 9, 4, 272, 0, 4, 29, 82, 5, 134, 1, 4, 19, 84, 18, 45, 0, 26, 7, 4, 117, 5, 1, 3, 3]
b_y [318, 318, 201, 74, 85, 0, 375, 186, 261, 60, 13, 1, 110, 1295, 115, 283, 28, 0, 622, 713, 1187, 770, 471, 1, 8, 196, 200, 54, 707, 0, 7, 3
44, 99, 123, 714, 1, 1307, 137, 277, 452, 107, 0, 371, 690, 987, 347, 76, 1, 111, 38, 105, 2450, 393, 0, 99, 609, 1182, 2989, 121, 1, 101, 324, 214, 6
92, 789, 0, 46, 187, 61, 2528, 5, 1, 3, 3]
*****
epoch 29 batch number 343 loss is: 6.237895488739014
prediction [10, 318, 179, 318, 393, 0, 46, 11, 91, 16, 115, 1, 10, 296, 11, 57, 17, 0, 83, 74, 99, 9, 13, 1, 72, 303, 617, 119, 8, 0, 46, 7, 311, 145,
46, 1, 7, 12, 546, 74, 8, 0, 12, 18, 10, 11, 134, 1, 4, 19, 9, 47, 42, 0, 9, 11, 117, 729, 30, 1, 3, 21, 4, 54, 41, 0, 15, 64, 49, 12, 281, 1, 3, 3]
b_y [1279, 1279, 1310, 154, 72, 0, 206, 206, 43, 293, 1497, 1, 2085, 3029, 452, 209, 8, 0, 60, 237, 92, 546, 13, 1, 1219, 983, 7, 14, 177, 0, 2
20, 75, 964, 145, 76, 1, 137, 78, 444, 309, 245, 0, 178, 1762, 392, 87, 535, 1, 678, 357, 28, 260, 114, 0, 857, 11, 99, 1110, 121, 1, 6, 1523, 2101, 4
, 51, 0, 35, 1775, 644, 922, 1979, 1, 3, 3]
*****
epoch 29 batch number 344 loss is: 6.174890518188477
prediction [10, 7, 19, 44, 159, 0, 39, 406, 43, 22, 84, 1, 4, 17, 7, 62, 29, 0, 6, 330, 17, 54, 79, 1, 7, 32, 4, 31, 8, 0, 6, 45, 31, 144, 22, 1, 8, 4
8, 7, 9, 70, 0, 5, 21, 8, 31, 46, 1, 7, 228, 9, 34, 17, 0, 7, 18, 117, 15, 88, 1, 39, 17, 26, 119, 74, 0, 7, 30, 10, 39, 21, 1, 3, 3]
b_y [196, 330, 69, 44, 70, 0, 581, 451, 18, 714, 457, 1, 8, 169, 5, 24, 73, 0, 6, 40, 100, 54, 626, 1, 185, 11, 32, 1148, 1148, 0, 249, 16, 63,
236, 236, 1, 118, 48, 168, 450, 378, 0, 6, 21, 101, 289, 615, 1, 22, 50, 259, 232, 252, 0, 33, 364, 117, 93, 594, 1, 233, 161, 88, 201, 943, 0, 86, 3
0, 43, 39, 632, 1, 3, 3]
*****
epoch 29 batch number 345 loss is: 6.337967395782471
prediction [10, 18, 13, 186, 70, 0, 84, 8, 37, 163, 7, 1, 10, 52, 117, 57, 42, 0, 10, 49, 10, 20, 32, 1, 8, 72, 28, 7, 16, 0, 6, 340, 10, 74, 178, 1,
82, 23, 5, 5, 28, 0, 83, 178, 4, 17, 178, 1, 10, 27, 9, 393, 534, 0, 4, 49, 10, 178, 20, 1, 3, 488, 9, 4, 42, 0, 42, 0, 10, 178, 4, 96, 5, 123, 5, 1,
3, 3]
b_y [457, 152, 475, 147, 70, 0, 223, 8, 37, 313, 1117, 1, 302, 457, 231, 652, 1824, 0, 222, 704, 70, 147, 5, 1, 457, 48, 84, 88, 16, 0, 201, 72
, 1766, 372, 1209, 1, 88, 16, 112, 594, 93, 0, 12, 57, 56, 23, 79, 1, 149, 278, 80, 309, 2542, 0, 28, 175, 254, 217, 1113, 1, 271, 488, 450, 135, 43,
15, 42, 0, 12, 65, 564, 475, 540, 157, 29, 1, 3, 3]
*****
epoch 29 batch number 346 loss is: 6.445919513702393
prediction [10, 23, 179, 11, 0, 8, 48, 0, 10, 87, 46, 117, 10, 178, 48, 1, 10, 11, 19, 34, 21, 16, 74, 0, 10, 7, 79, 7, 48, 44, 107, 1, 30, 62, 4, 8,
16, 41, 41, 0, 84, 11, 4, 48, 4, 26, 104, 1, 30, 5, 4, 163, 32, 80, 393, 0, 10, 117, 15, 82, 4, 268, 107, 1, 3, 92, 5, 151, 56, 0, 29, 0, 10, 62, 4, 3
7, 9, 6, 330, 1, 3, 3]
b_y [188, 682, 804, 242, 143, 242, 70, 0, 10, 242, 297, 41, 10, 242, 100, 1, 342, 71, 191, 6, 240, 60, 154, 0, 555, 85, 1335, 1335, 69, 97, 107
, 1, 139, 284, 325, 196, 4, 149, 95, 0, 36, 11, 99, 73, 144, 37, 381, 1, 38, 5, 112, 163, 1572, 241, 867, 0, 182, 7, 1076, 1076, 243, 134, 434, 1, 441
, 70, 296, 457, 869, 108, 20, 0, 729, 131, 907, 117, 66, 6, 539, 1, 3, 3]
```

生成诗歌的结果如下：

```
root@autodl-container-8abd47b3fd-ed0d512b:~/tangshi# python main.py
error
initial linear weight
/root/tangshi/rnn.py:71: UserWarning: Implicit dimension choice for
  out = self.softmax(out)
日月明时起，一枝花落不成雪。
青山不见时，一望无人知。
error
initial linear weight
青山不见此，一望不相逢。
何处无枝下，千年一望春。
error
initial linear weight
玉人来往事，一为一年年。
君子无人见，何人问我心。
error
initial linear weight
夜风如欲落，风光不在人。
高人不见客，不是有清风。
error
initial linear weight
一月不知人，千年不见人。
何人更何事，不见人间人。
error
initial linear weight
青山如有事，一醉不成花。
何处无枝下，何时见此时。
error
initial linear weight
山中无见日，不是更何人。
root@autodl-container-8abd47b3fd-ed0d512b:~/tangshi#
```

## 4 实验总结

完成本次实验对 RNN 与 LSTM 的网络结构有了进一步的了解，也实践了 LSTM 模型相关的数据预处理、模型搭建、训练与预测的全过程。在训练过程中可以发现，相比于全连接神经网络、卷积神经网络等不涉及序列任务的模型结构，诸如 LSTM 等的循环神经网络由于为了适应序列任务，在模型中引入了循环更新隐状态的结构，明显地导致了模型收敛的速度降低，收敛的稳定性也下降。相比于 CNN 网络，LSTM 网络的损失函数经过长时间多轮训练只是出现了一定的收敛趋势，但是最终也没有像 CNN 那样，出现损失函数的显著降低，而且在每轮训练中，损失函数的收敛也并不稳定，会出现频繁波动的情况，初步分析这可能都与其模型结构中的长时序依赖有关。但根据最终生成诗歌的结果可见，训练的 LSTM 网络仍然学习到了诗歌数据中相当一部分的特征，说明 LSTM 的结构确实具有较好的拟合与重建序列信息的能力。