

诗歌生成实验报告

一、RNN、LSTM 和 GRU 模型解释

1. RNN（循环神经网络）

原理：RNN 是一种用于处理序列数据的神经网络。它通过引入循环结构，使得网络能够记忆之前的信息，并将其用于当前的计算。RNN 的基本单元是一个循环层，每个时间步的输出不仅取决于当前输入，还取决于前一时间步的输出。

优点：能够处理任意长度的序列数据，适用于文本生成、语音识别等任务。

缺点：在处理长序列时，容易出现梯度消失或梯度爆炸的问题，导致网络难以学习到长距离的依赖关系。

2. LSTM（长短期记忆网络）

原理：LSTM 是 RNN 的一种改进版本，通过引入门控机制来解决梯度消失的问题。LSTM 的每个单元包含三个门：遗忘门（Forget Gate）、输入门（Input Gate）和输出门（Output Gate）。遗忘门控制丢弃哪些信息，输入门控制新信息的写入，输出门控制单元状态的输出。

优点：能够有效地学习长距离的依赖关系，适用于需要捕捉长序列依赖的任务，如文本生成、机器翻译等。

缺点：结构复杂，计算量较大，训练速度相对较慢。

3. GRU（门控循环单元）

原理：GRU 是 LSTM 的一种简化版本，通过引入更新门（Update Gate）和重置门（Reset Gate）来控制信息的流动。更新门决定新信息的写入程度，重置门决定旧信息的保留程度。GRU 的结构比 LSTM 更简洁，但仍然能够有效地捕捉长距离依赖关系。

优点：结构简单，计算量小，训练速度较快，同时能够有效地解决梯度消失问题。

缺点：在某些复杂的长序列任务中，性能可能略逊于 LSTM。

二、诗歌生成过程

1. 数据预处理

数据来源：使用唐诗数据集（tangshi.txt），包含大量的唐诗文本。

数据清洗：

去除诗中的标点符号、特殊字符和多余的空格。

筛选长度在一定范围内的诗歌（如 5 到 80 个字符）。

在每首诗的开头添加起始标记（如 G），在结尾添加结束标记（如 E）。

构建词汇表：

统计所有诗歌中出现的字符及其频率。

按频率排序，构建字符到索引的映射（word_to_int）和索引到字符的映射（vocabularies）。

生成训练数据：

将每首诗转换为索引序列。

将索引序列划分为输入序列和目标序列。目标序列是输入序列向右移动一个位置的结果。

2. 模型构建

词嵌入层：将输入的索引序列映射到低维的词嵌入空间。

RNN 层：使用 LSTM 或 GRU 作为循环层，捕捉序列中的时间依赖关系。

全连接层：将 RNN 的输出映射到词汇表的大小，输出每个字符的概率分布。

激活函数：使用 Softmax 函数将输出转换为概率分布。

3. 训练过程

损失函数：使用负对数似然损失（NLLLoss）计算预测概率分布与真实标签之间的差异。

优化器：使用 RMSprop 优化器进行梯度下降。

训练步骤：

- 将数据划分为多个批次。

- 对每个批次的输入序列进行前向传播，计算损失。

- 反向传播计算梯度，并更新模型参数。

- 每隔一定步数保存模型，并打印当前的损失值和预测结果。

4. 生成诗歌

初始化：指定一个起始字符（如 日、红、山 等）作为输入。

生成过程：

- 将起始字符转换为索引，并输入到模型中。

- 模型输出下一个字符的概率分布，选择概率最高的字符作为下一个输入。

- 重复上述步骤，直到生成的诗歌达到一定长度或遇到结束标记。

三、实验结果

1. 训练过程

以下是训练过程中的部分输出截图：

```
*****
epoch 29 batch number 241 loss is: 5.6793107986450195
prediction [10, 7, 7, 0, 552, 0, 22, 928, 69, 309, 32, 1, 10, 30, 4, 48, 653, 0, 4, 34, 9, 123, 74, 1, 11, 57, 4, 187, 11, 0, 36, 58, 77, 13, 211, 1, 7, 5, 9, 200, 17, 0, 101, 564, 9, 13, 116, 1, 15, 68, 9, 46, 104, 0, 15, 205, 4, 51, 62, 1, 3, 3]
b_y [25, 541, 122, 320, 1216, 0, 215, 229, 5, 674, 398, 1, 231, 30, 72, 372, 646, 0, 39, 114, 390, 201, 1425, 1, 483, 827, 117, 36, 11, 0, 79, 559, 180, 237, 211, 1, 20, 14, 524, 6, 17, 0, 851, 96, 1451, 46, 945, 1, 1391, 163, 324, 518, 104, 0, 1373, 205, 54, 1, 021, 24, 1, 3, 3]
*****
epoch 29 batch number 242 loss is: 5.602120399475098
prediction [10, 7, 106, 228, 0, 0, 46, 37, 10, 43, 235, 1, 10, 14, 9, 8, 29, 0, 50, 595, 9, 11, 153, 1, 10, 638, 4, 114, 7, 0, 6, 7, 1, 0, 16, 35, 1, 10, 7, 4, 9, 57, 0, 50, 16, 106, 7, 59, 1, 15, 108, 226, 9, 104, 0, 37, 5, 4, 4, 423, 1, 3, 3]
b_y [199, 186, 1058, 20, 1796, 0, 3098, 262, 168, 101, 235, 1, 334, 198, 310, 157, 894, 0, 50, 760, 109, 154, 756, 1, 289, 865, 901, 34, 647, 0, 46, 56, 25, 1796, 78, 1, 211, 224, 1128, 921, 269, 0, 75, 284, 375, 352, 118, 1, 555, 961, 300, 1915, 1578, 0, 241, 373, 168, 31, 156, 1, 3, 3]
*****
epoch 29 batch number 243 loss is: 5.705140590667725
prediction [10, 64, 19, 56, 5, 0, 4, 23, 54, 63, 16, 1, 15, 21, 6, 150, 17, 0, 6, 53, 4, 21, 165, 1, 4, 112, 9, 748, 0, 0, 10, 14, 4, 148, 94, 1, 15, 67, 6, 62, 508, 0, 15, 30, 10, 24, 11, 1, 15, 637, 1838, 389, 45, 0, 15, 21, 9, 7, 5, 1, 3, 3]
b_y [15, 42, 628, 20, 357, 0, 76, 554, 679, 21, 109, 1, 204, 1105, 19, 25, 192, 0, 7, 269, 12, 627, 627, 1, 291, 469, 915, 989, 38, 0, 24, 32, 1014, 245, 446, 1, 263, 19, 208, 281, 508, 0, 27, 1625, 244, 36, 11, 1, 15, 8, 157, 135, 306, 0, 737, 65, 25, 41, 30, 1, 3, 3]
*****
epoch 29 batch number 244 loss is: 5.529922008514404
prediction [10, 95, 179, 37, 0, 0, 13, 748, 264, 237, 116, 1, 10, 116, 10, 435, 32, 0, 4, 112, 9, 25, 7, 1, 10, 15, 4, 149, 8, 0, 4, 7, 3, 9, 4, 164, 1, 15, 15, 4, 149, 0, 0, 4, 399, 9, 14, 528, 1, 15, 68, 47, 35, 5, 0, 4, 39, 4, 149, 151, 1, 3, 3]
b_y [4619, 4619, 5211, 2163, 1044, 0, 4360, 4360, 839, 2347, 3259, 1, 339, 339, 4867, 435, 32, 0, 289, 289, 843, 46, 219, 1, 47, 15, 4, 595, 885, 0, 560, 335, 328, 728, 3736, 1, 47, 15, 4, 1288, 20, 0, 141, 335, 1972, 728, 1460, 1, 327, 1277, 282, 2436, 225, 0, 511, 337, 4, 599, 1141, 1, 3, 3]
*****
epoch 29 batch number 245 loss is: 5.835883140563965
prediction [10, 57, 264, 48, 0, 0, 0, 0, 10, 179, 84, 184, 4, 8, 50, 1, 10, 7, 4, 8, 17, 100, 983, 0, 13, 7, 4, 8, 10, 200, 12, 1, 10, 407, 9, 146, 80, 192, 309, 0, 36, 8, 5, 67, 10, 134, 21, 1, 4, 145, 10, 0, 124, 96, 96, 0, 4, 172, 5, 193, 10, 103, 15, 1, 3, 3]
*****
epoch 29 batch number 344 loss is: 5.627809047698975
prediction [10, 7, 155, 44, 45, 0, 15, 451, 10, 22, 457, 1, 4, 169, 9, 144, 29, 0, 6, 330, 17, 145, 532, 1, 7, 6, 35, 4, 826, 0, 11, 1, 6, 17, 28, 40, 1, 17, 85, 9, 9, 64, 0, 6, 21, 4, 4, 331, 1, 4, 228, 671, 122, 383, 0, 7, 18, 130, 67, 13, 1, 128, 161, 9, 119, 74, 0, 9, 24, 4, 140, 35, 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, 30, 43, 39, 632, 1, 3, 3]
*****
epoch 29 batch number 345 loss is: 5.8000359535217285
prediction [10, 18, 8, 100, 0, 0, 10, 8, 4, 103, 328, 1, 10, 52, 117, 39, 192, 0, 4, 288, 30, 147, 20, 1, 10, 18, 6, 7, 16, 0, 5, 32, 10, 94, 178, 1, 10, 23, 4, 6, 14, 0, 46, 178, 4, 17, 29, 1, 10, 172, 39, 190, 42, 0, 4, 43, 10, 22, 35, 1, 15, 488, 9, 1693, 42, 0, 42, 0, 4, 164, 4, 96, 10, 5, 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, 1, 113, 1, 271, 488, 450, 135, 43, 15, 42, 0, 12, 65, 564, 475, 540, 137, 29, 1, 3, 3]
*****
epoch 29 batch number 346 loss is: 5.910001277923584
prediction [10, 23, 4, 1026, 48, 97, 21, 0, 10, 178, 12, 85, 10, 178, 21, 1, 12, 6, 13, 34, 191, 293, 638, 0, 4, 399, 375, 383, 10, 97, 44, 1, 10, 45, 4, 115, 10, 149, 41, 0, 10, 11, 4, 48, 15, 26, 5, 1, 68, 5, 4, 39, 9, 28, 5, 0, 4, 6, 4, 695, 4, 4, 124, 1, 3, 20, 4, 35, 5, 0, 794, 0, 10, 131, 4, 18, 10, 6, 17, 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, 1, 335, 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, 1, 076, 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: 5.779188632965088
```

2. 生成的诗歌

以下是使用不同起始词生成的诗歌：

以“日”开头：

日和月，飞鸟声中晓色中。
何处不得西风起，一枝何处一相寻。

日来长如何，一半一为一时。

以“红”开头：

红桃花发露春，青山一望水山头。
如今不见人间事，不是人间不得无。

红树带风烟，玉树浮花露月收。

红然然老地无难难远长结叶琉叶结叶琉叶结叶琉叶结叶杯前成青何处用。

以“山”开头：

山上无人死，不是人间不得名。
不是此中无一事，不能惆得是闲人。

山雨高不在，山前水上雨。

以“夜”开头：

夜来看月中，一树风落一时开。

夜深夜月还如此，不得还家一万年。

夜来相断，不得山中一何。

以“湖”开头：

湖海为人醉，一半千山一醉衣。
不是此中无一事，不能惆得是闲人。

湖上雪长水，青山一见一时归。

湖水不留春，江海分苏古道同。

以“海”开头：

海去无人，一半春风入洞庭。

海门前，风起风光早相见。

海木万国年年何似国江不复江不复江不须清E。

以“月”开头：

月已无名，不能闲去更何如。
如今一别无人见，不是人间不得无。

月明水流，云外风前玉树头。
一夜春风满天上，一声声入凤池头。

月断发，风流红药满朝枝。
平生不见无人见，不得无人不可寻。

四、实验总结

1. 实验效果

通过本次实验，成功实现了基于 RNN 的诗歌生成模型。模型能够根据给定的起始词生成具有一定韵律和意义的诗歌。生成的诗歌在一定程度上符合唐诗的风格，但也存在一些不足之处，如部分诗句的连贯性和逻辑性有待提高。

2. 模型改进方向

增加数据量：扩大训练数据集，增加模型对不同风格和主题的诗歌的学习能力。

调整模型结构：尝试使用更复杂的 RNN 结构（如多层 LSTM 或 GRU），或引入注意力机制，提高模型对长序列的建模能力。

优化训练过程：调整学习率、优化器等超参数，提高模型的训练效果。

3. 实验体会

本次实验让我对 RNN 模型有了更深入的理解，同时也体会到了自然语言处理任务的复杂性和挑战性。通过实际操作，我掌握了如何使用 PyTorch 框架构建和训练循环神经网络模型，并能够将其应用于文本生成任务。未来将继续探索更先进的模型和算法，进一步提高模型的性能和生成效果。