

RNN 实验报告

一、RNN、LSTM、GRU 模型

1. RNN

核心思想：

处理序列数据（如文本、时间序列），通过隐藏状态传递历史信息。

循环连接：当前状态 h_t 由当前输入 x_t 和前一状态 h_{t-1} 决定。

参数共享：所有时间步共用同一组权重。

公式： $h_t = \sigma(Wx_t + Wh \cdot h_{t-1} + bh)$

优点：适合变长序列，理论可建模任意长度依赖。

缺点：梯度消失/爆炸，难以学习长期依赖。

适用于简单文本生成、短时间序列预测

2. LSTM

核心思想：解决 RNN 的长期依赖问题，通过门控机制控制信息流动。

细胞状态（Cell State）：长期记忆通道

三门机制：遗忘门、输入门、输出门：

优点：有效缓解梯度消失，适合长序列任务。

缺点：参数量大，计算成本高。

适用于机器翻译、语音识别、文档摘要生成

3. GRU

核心思想：

LSTM 的简化版，合并细胞状态和隐藏状态。

双门机制：重置门（ r_t ）、更新门（ z_t ）。

优点：计算效率高，性能接近 LSTM。

缺点：对极长序列建模略弱于 LSTM。

适用于文本分类、股票价格预测、用户行为序列建模

二、诗歌生成过程

初始化：

给定起始字符，将其转换为索引输入模型。

自回归生成：

模型预测下一个字符的概率分布，通过 `np.argmax` 选择最可能的字符。

将新字符添加到诗中，作为下一步的输入，直到生成结束符 `E` 或超过最大长度。

结果格式化，移除起止符，按句分割打印。

三、生成诗歌

```
FullModel.load_state_dict(torch.load('model.pth'))
不见一心事，无人不见人。
error
initial linear weight
不见一时心，无令此心在。
error
initial linear weight
error
initial linear weight
何处望中去，无令见一生。
error
initial linear weight
一时望水来，一望一千年。
error
initial linear weight
一时望水来，一望一千年。
error
initial linear weight
一时望水来，一望一千年。
error
initial linear weight
何处更无事，今年不见春。
```

出现了未知错误，未找到原因