

## 1. RNN、LSTM 和 GRU 模型解释

RNN 通过隐藏状态将前一步的输出和当前输入一起传递到下一步，适用于处理序列数据（如文本、语音等），存在梯度消失和梯度爆炸问题，导致长期依赖关系学习困难。

LSTM 通过引入遗忘门、输入门和输出门来控制信息流动，缓解了梯度消失问题。可以捕获长期依赖关系，适合处理长序列。

GRU 是 LSTM 的简化版本，合并了遗忘门和输入门，减少了参数量。与 LSTM 效果相似，但计算效率更高。

## 2. 诗歌生成过程

读取文本数据，并将文本拆分成单个字符或词汇。使用 Counter 统计词频，建立词汇表 word2id（从词到索引的映射）和 id2word（从索引到词的映射）。使用 Embedding 层将词汇映射到固定大小的向量、SimpleRNNCell 定义 RNN 层，输出特征表示、Dense 层生成每个时间步的输出。使用 tf.data.Dataset 创建数据集，生成批次数据，使用交叉熵损失函数，结合 reduce\_avg 计算损失，用 Adam 优化器进行反向传播，更新模型参数。

以指定的 begin\_word 作为起始输入，通过 get\_next\_token() 方法逐步生成下一个词，遇到结束符 eos 时结束生成。

## 3. 生成结果

---

以 '日' 开头的诗歌：日暮云开。  
以 '红' 开头的诗歌：红蓉瀟，十载凌云起，金輿不可怜。  
以 '山' 开头的诗歌：山畔寺皆相见，不见人间人。  
以 '夜' 开头的诗歌：夜斜阳路，春风满树中。  
以 '湖' 开头的诗歌：湖水上风光，玉衣香下水。  
以 '海' 开头的诗歌：海畔凌悠悠悠悠悠悠悠悠悠悠悠悠。  
以 '月' 开头的诗歌：月。