

20 高级循环神经网络

概要

- 深度循环神经网络
- 双向循环神经网络
- 循环神经网络结合
 - 残差网络 (ResNet)
 - 稠密连接网络 (DenseNet)
- 循环神经网络的正则化

回顾：标准RNN

- 标准RNN通过循环的隐藏状态 H_t 来处理序列信息，其核心是两个状态更新公式：
 - 隐藏状态更新: $H_t = \phi(X_t W_{xh} + H_{t-1} W_{hh} + b_h)$
 - 输出更新: $O_t = H_t W_{hq} + b_q$
- 其中， H_{t-1} 捕获了到上一步为止的所有历史信息。然而，这种“浅层”的循环结构在两个维度上存在局限：
 - 纵向深度局限: 在每个时间步 t ，从输入 X_t 到输出 O_t 的计算路径是浅的，限制了模型在单步内进行复杂特征变换的能力
 - 横向视野局限: H_t 仅依赖于过去 ($t < T$) 的信息，无法利用未来的上下文
- 接下来，我们将探讨如何从这两个维度突破这些局限

深度循环神经网络

深度循环神经网络 - 根本问题与核心目标

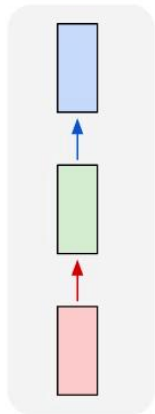
➤ 根本问题与核心目标

- 标准RNN在每个时间步的计算是“浅”的。从输入 X_t 和前一状态 H_{t-1} 到新状态 H_t 的转换仅涉及一次非线性变换。这限制了模型在单个时间步内学习复杂层次化特征的能力
 - 我们需要增加RNN的“深度”
- 增加模型在时间维度上的“垂直”表示能力
 - 通过在每个时间步堆叠多个循环层，使得网络能够学习从低级到高级的特征抽象。例如，在语音识别中，第一层可能识别音素，第二层组合音素成音节，第三层再组合成词汇

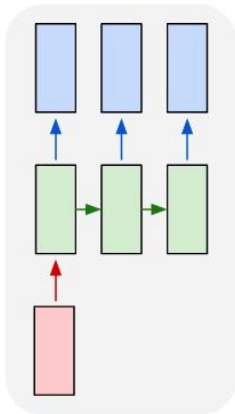
使用循环神经网络

- 编码序列
- 解码序列
- 同时做到这两点

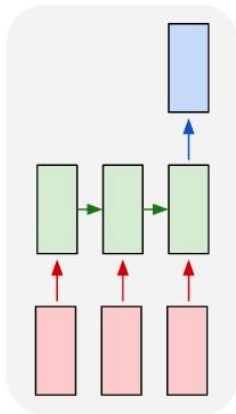
one to one



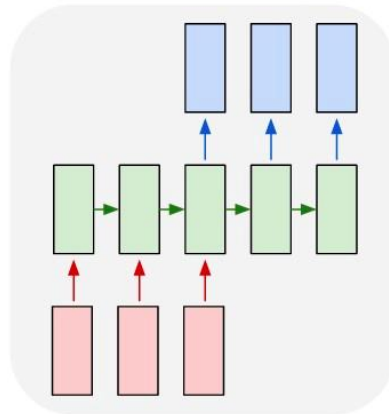
one to many



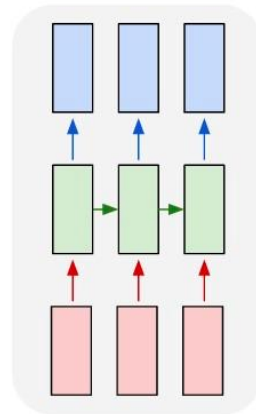
many to one



many to many

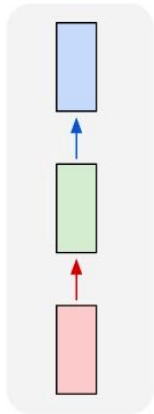


many to many

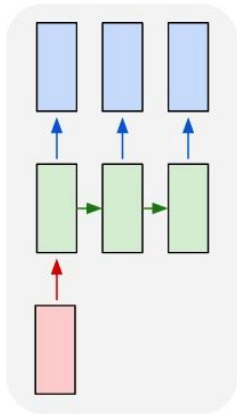


使用循环神经网络

one to one

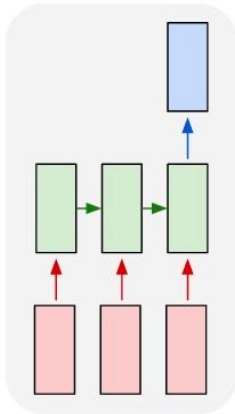


one to many



编写诗歌

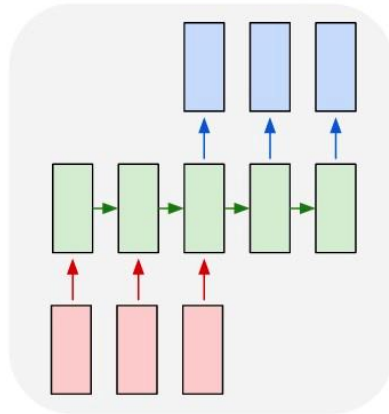
many to one



情感分析

文本分类

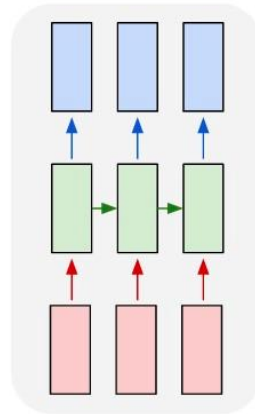
many to many



自动问答

机器翻译

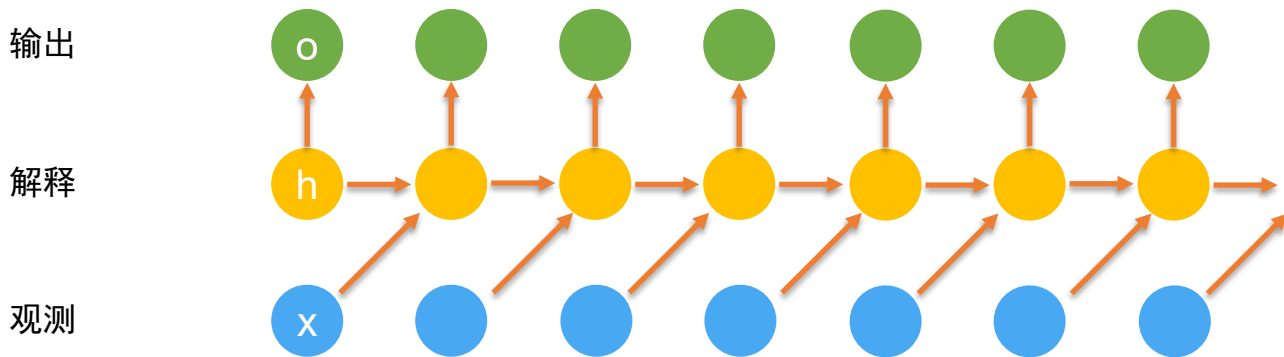
many to many



命名实体标记

(图片来源: karpathy.github.io)

回顾 - 循环神经网络



➤ 隐含状态更新

$$\mathbf{h}_t = \phi(\mathbf{W}_{hh}\mathbf{h}_{t-1} + \mathbf{W}_{hx}\mathbf{x}_{t-1} + \mathbf{b}_h)$$

➤ 观测更新

$$\mathbf{o}_t = \phi(\mathbf{W}_{ho}\mathbf{h}_t + \mathbf{b}_o)$$

➤ 怎么添加更多非线性的层？

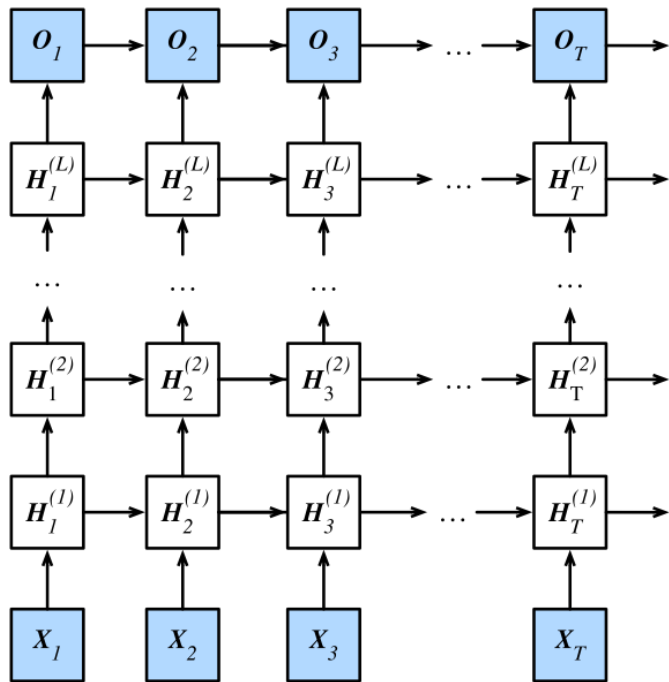
深度循环神经网络

➤ 浅度循环神经网络

- 输入
- 隐含层
- 输出

➤ 深度循环神经网络

- 输入
- 隐含层
- 隐含层
- ...
- 输出



深度循环神经网络

$$\text{➤ } \mathbf{H}_t^1 = f_1(\mathbf{H}_{t-1}^1, \mathbf{X}_t)$$

$$\text{➤ } \mathbf{H}_t^j = f_j(\mathbf{H}_{t-1}^j, \mathbf{H}_t^{j-1})$$

$$\text{➤ } \mathbf{O}_t = g(\mathbf{H}_t^L)$$

$$\text{➤ } \mathbf{H}_t^1 = f_1(\mathbf{H}_{t-1}^1, \mathbf{X}_t)$$

$$\text{➤ } \mathbf{H}_t^j = f_j(\mathbf{H}_{t-1}^j, \mathbf{H}_t^{j-1})$$

$$\text{➤ } \mathbf{O}_t = g(\mathbf{H}_t^L)$$

双向循环神经网络

“未来”的重要性

- 我 _____ 。
- 我 _____ 非常饿。
- 我 _____ 非常饿，我可以吃下一只猪。

“未来”的重要性

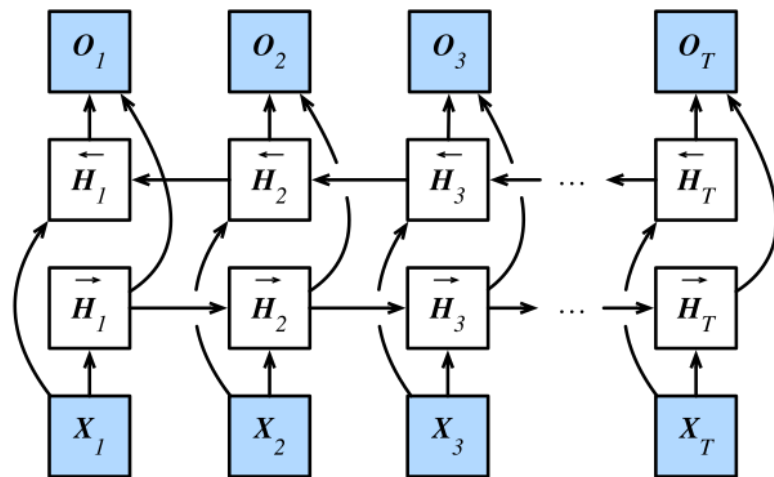
- 我 饿 。
- 我 不是 非常饿。
- 我 非常 非常饿，我可以吃下一只猪。

“未来”的重要性

- 我 饿 。
 - 我 不是 非常饿。
 - 我 非常 非常饿，我可以吃下一只猪。
-
- 填写完全不同的单词，取决于单词的上下文
 - 到目前为止，RNN 只关注上文
 - 也可以使用下文来填写

双向循环神经网络

- 一个 RNN 正向传递
- 另一个 RNN 反向传递
- 将两种隐含状态组合在一起以生成输出



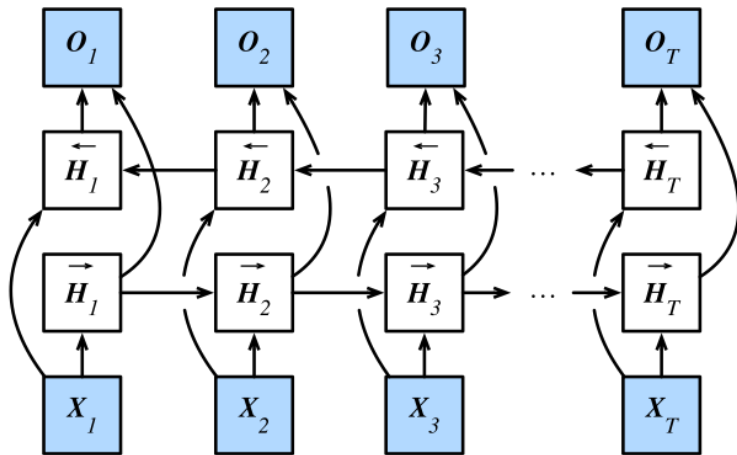
双向循环神经网络

$$\vec{\mathbf{H}}_t = \phi \left(\mathbf{x}_t \mathbf{W}_{xh}^{(f)} + \vec{\mathbf{H}}_{t-1} \mathbf{W}_{hh}^{(f)} + \mathbf{b}_h^{(f)} \right)$$

$$\rightrightarrows \overleftarrow{\mathbf{H}}_t = \phi \left(\mathbf{x}_t \mathbf{W}_{xh}^{(b)} + \overleftarrow{\mathbf{H}}_{t+1} \mathbf{W}_{hh}^{(b)} + \mathbf{b}_h^{(b)} \right)$$

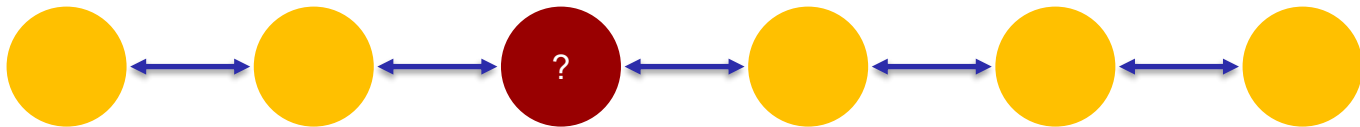
$$\mathbf{H}_t = [\vec{\mathbf{H}}_t, \overleftarrow{\mathbf{H}}_t]$$

$$\mathbf{O}_t = \mathbf{H}_t \mathbf{W}_{hq} + \mathbf{b}_q$$

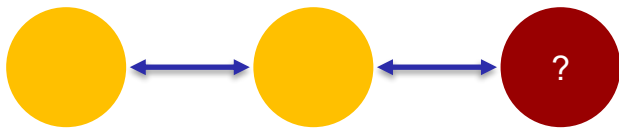


推理

➤ 训练阶段



➤ 推理（不合适）



➤ 仍然可以使用它来编码序列

总结

➤ 两种增强RNN能力的高级结构

➤ 深度循环神经网络 (Deep RNN)

- 目的: 增强模型在每个时间步的表示能力和特征抽象层次
- 机制: 垂直堆叠多个循环层，上层RNN将下层RNN的输出作为输入
- 类比: 从“思考不深”到“深度思考”

➤ 双向循环神经网络 (Bi-RNN)

- 目的: 融合过去和未来的上下文信息，生成更全面的序列表示
- 机制: 并行运行一个前向RNN和一个后向RNN，并合并它们的隐藏状态
- 类比: 从“单向阅读”到“双向阅读理解”

➤ 这两种结构分别从“深度”和“广度”（上下文视野）两个维度扩展了RNN的能力，是构建现代序列处理模型的关键组件