循环神经网络

序列模型

1. 时序数据

- 实际中很多数据是有时序结构的
  a) 电影的评价随时间变化而变化
     拿奖后评分上升，直到奖项被忘证
     看了很多好电影后，人们的期望变高
     季节性：贺岁片、暑期档
     导演、演员的负面报道导致评分变低
  b) 音乐、语言、文本、和视频都是连续的
  c) 大地震发生后，很可能会有几次较小的余震
  d) 人的互动是连续的，从网上吵架可以看出
  e) 预测明天的股价要比填补昨天遗失的股价的更困难

2. 统计工具

- 在时间 t 观察到 x_t，那么得到 T 个不独立的随机变量
  $(x_1, \ldots x_T) \sim p(\mathbf{x})$
- 使用条件概率展开
  $p(a,b) = p(a)p(b|a) = p(b)p(a|b)$

3. 序列模型

$$p(\mathbf{x}) = p(x_1) \cdot p(x_2|x_1) \cdot p(x_3|x_1,x_2) \cdot \ldots p(x_T|x_1, \ldots x_{T-1})$$

- 对条件概率建模

$$p(x_t|x_1, \ldots x_{t-1}) = p(x_t|f(x_1, \ldots x_{t-1}))$$

对见过的数据建模，也称自回归模型

3.1 方案 A – 马尔科夫假设

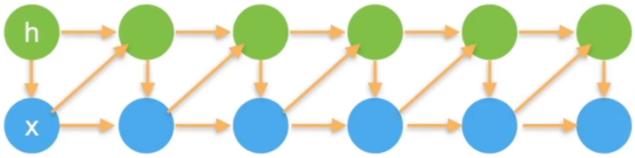
- 假设当前数据只跟τ个过去数据点相关

$$p(x_t \mid x_1, \ldots x_{t-1}) = p(x_t \mid x_{t-\tau}, \ldots x_{t-1}) = p(x_t \mid f(x_{t-\tau}, \ldots x_{t-1}))$$

例如在过去数据上训练
一个MLP模型

3.2 方案 B – 潜变量模型

- 引人潜变量 h_t 来表示过去信息 $h_t = f(x_1, \ldots x_{t-1})$
  - 这样 $x_t = p(x_t \mid h_t)$



4. 总结

- 时序模型中，当前数据跟之前观察到的数据相关
- 自回归模型使用自身过去数据来预测未来
- 马尔科夫模型假设当前只跟最近少数数据相关，从而简化模型
- 潜变量模型使用潜变量来概括历史信息

语言模型

1. 概念

- 给定文本序列 $x_1, \ldots, x_T$，语言模型的目标是估计联合概率 $p(x_1, \ldots, x_T)$
- 应用包括
  a) 做预训练模型（e.g. BERT， GPT-3）
  b) 生成文本，给定前面几个词，不断的使用$x_t \sim p(x_t|x_1, \ldots, x_{t-1})$来生成后续文本
  c) 判断多个序列中哪个更常见，e.g. **'to recognize speech'** vs. **'to wreck a nice beach'**

2. 使用计数来建模

- 假设序列长度为2，我们预测

$$p(x, x') = p(x)p(x'|x) = \frac{n(x)}{n}\frac{n(x, x')}{n(x)}$$

  - 这里 n 是总词数，$n(x), n(x, x')$ 是单个单词和连续单词对的出现次数

- 很容易拓展到长为3的情况

$$p(x, x', x'') = p(x)p(x'|x)p(x''|x, x') = \frac{n(x)}{n}\frac{n(x, x')}{n(x)}\frac{n(x, x', x'')}{n(x, x')}$$

2.1 N 元语法

- 当序列很长时，因为文本量不够大，很可能 $n(x_1, \ldots, x_T) \leq 1$
- 使用马尔科夫假设可以缓解这个问题
  - 一元语法：
$$p(x_1, x_2, x_3, x_4) = p(x_1)p(x_2)p(x_3)p(x_4) = \frac{n(x_1)}{n}\frac{n(x_2)}{n}\frac{n(x_3)}{n}\frac{n(x_4)}{n}$$
  - 二元语法：
$$p(x_1, x_2, x_3, x_4) = p(x_1)p(x_2|x_1)p(x_3|x_2)p(x_4|x_3) = \frac{n(x_1)}{n}\frac{n(x_1, x_2)}{n(x_1)}\frac{n(x_2, x_3)}{n(x_2)}\frac{n(x_3, x_4)}{n(x_3)}$$
  - 三元语法：$p(x_1, x_2, x_3, x_4) = p(x_1)p(x_2|x_1)p(x_3|x_1, x_2)p(x_4|x_2, x_3)$

3. 总结

- 语言模型估计文本序列的联合概率
- 使用统计方法时常采用 n 元语法

循环神经网络 RNN

1. 潜变量回归模型

- 使用潜变量h_t总结过去信息

$$h_t = f(x_t, h_{t-1})$$

2. 循环神经网络

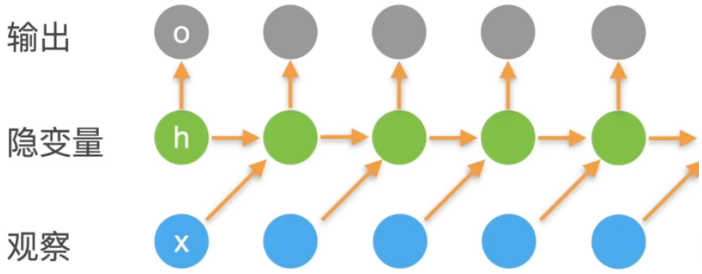
- 当前时间步隐藏变量由当前时间步输入与前一个时间步的隐藏变量一起计算得出

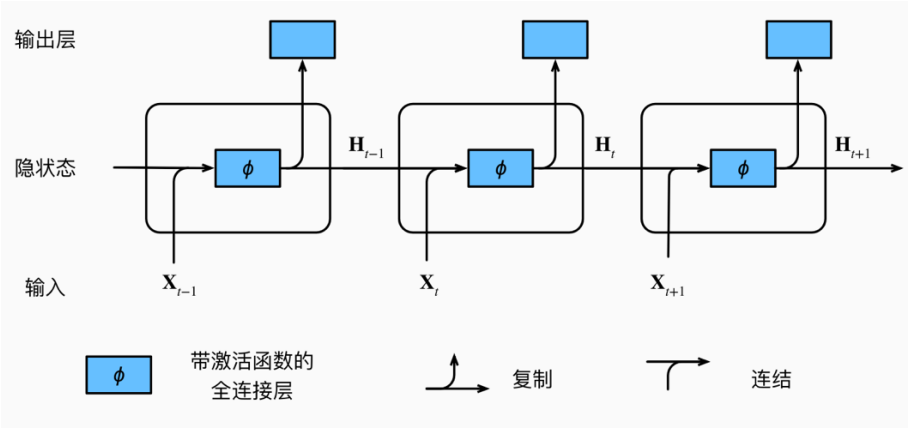
$$\boldsymbol{H_t} = \emptyset(\boldsymbol{X_t W_{xh}} + \boldsymbol{H_{t-1} W_{hh}} + \boldsymbol{b_h})$$

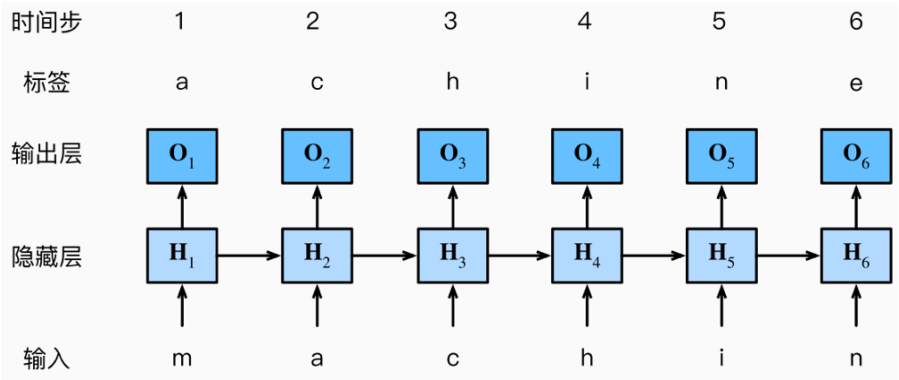
- 输出:

$$\boldsymbol{O} = \boldsymbol{H W_{hq}} + \boldsymbol{b_q}$$

例如，分类问题可以采用$softmax(\boldsymbol{O})$来计算输出类别的概率分布







3. 困惑度 perplexity

- 衡量一个语言模型的好坏可以用平均交叉熵

$$\pi = \frac{1}{n}\sum_{i=1}^{n} -\log p(x_t \mid x_{t-1}, \ldots)$$

p 是语言模型的预测概率，x_t 是真实词

- 历史原因NLP使用困惑度 $\exp(\pi)$ 来衡量，是平均每次可能选项
  - 1表示完美，无穷大是最差情况

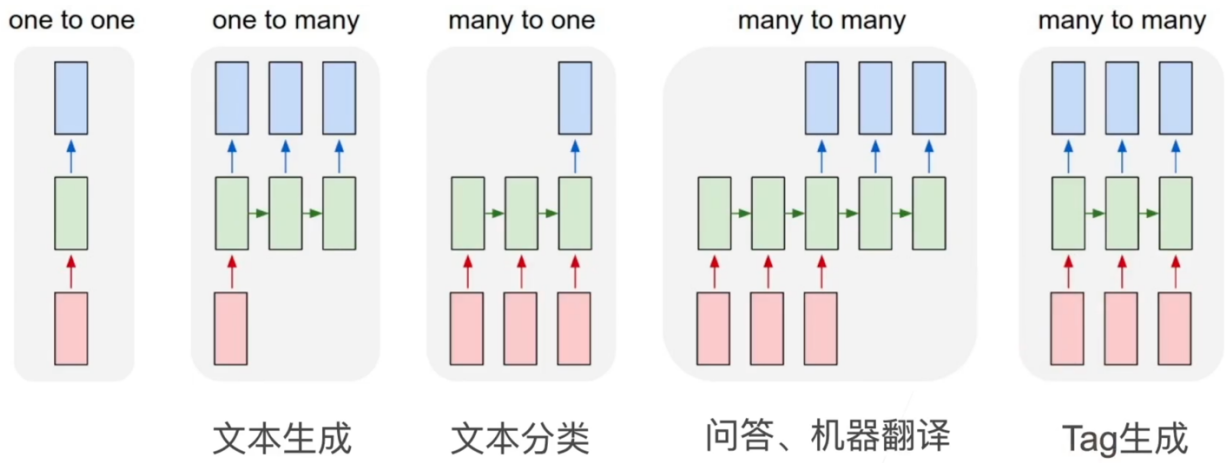
4. 梯度裁剪

- 迭代中计算这T个时间步上的梯度，在反向传播过程中产生长度为 $O(T)$的矩阵乘法链，导致数值不稳定
- 梯度裁剪能有效预防梯度爆炸
  - 如果梯度长度超过 θ，那么拖影回长度θ

$$\mathbf{g} \leftarrow \min\left(1, \frac{\theta}{\|\mathbf{g}\|}\right)\mathbf{g}$$

其中，$\boldsymbol{g}$表示的是所有层的梯度。

5. 更多的 RNN 应用



6. 总结

- 循环神经网络的输出取决于当下输入和前一时间的隐变量
- 应用到语言模型中时，循环神经网络根据当前词预测下一次时刻词
- 通常使用困惑度来衡量语言模型的好坏