

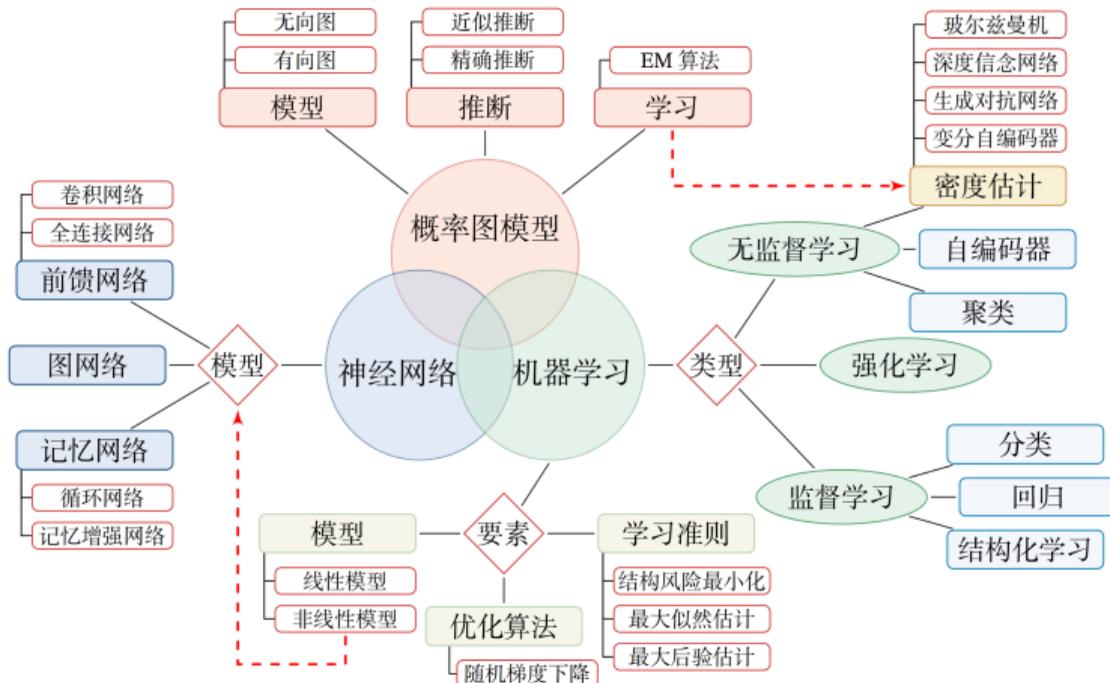
第六章 神经网络基础

修贤超

<https://xianchaoxiu.github.io>

神经网络与深度学习

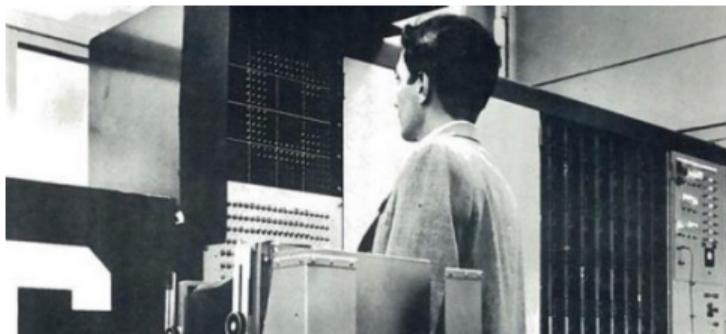
■ <https://mndl.github.io/>



- 6.1 引言
- 6.2 前馈神经网络
- 6.3 卷积神经网络
- 6.4 循环神经网络

■ 第一阶段: 模型提出

- 1943 年, Warren McCulloch 和 Walter Pitts 最早描述了一种理想化的人工神经网络, 并构建了基于简单逻辑运算的 MP 模型
- 1948 年, Alan Turing 在的论文中描述了一种 B 型图灵机
- 1951 年, Marvin Minsky 建造了第一台神经网络机 SNARC
- 1958 年, Frank Rosenblatt 提出了可以模拟人类感知能力的**感知器**



■ 第二阶段：冰河期

- 1969 年，Marvin Minsky 出版《感知器》一书，发现感知器无法处理异或回路，电脑没有足够的处理能力处理大型神经网络
- 1974 年，Paul Webos 发明了反向传播算法，但当时未受到应有的重视
- 1980 年，Kunihiko Fukushima 提出了一种带卷积和子采样操作的多层神经网络，称为新知机（Neocognitron）



■ 第三阶段：反向传播算法引起的复兴

- 1983 年, John Hopfield 对神经网络引入能量函数的概念, 并提出了用于联想记忆和优化计算的 Hopfield 网络
- 1984 年, Geoffrey Hinton 提出了玻尔兹曼机
- 1986 年, David Rumelhart 和 James McClelland 重新发明了反向传播算法
- 1986 年, Geoffrey Hinton 等人将反向传播算法到多层感知器
- 1989 年, Yann LeCun 等人将反向传播算法引入了卷积神经网络, 并在手写体数字识别上取得了很大的成功

Gradient-based learning applied to document recognition

[Y LeCun, L Bottou, Y Bengio...](#) - Proceedings of the ..., 1998 - [ieeexplore.ieee.org](#)

Multilayer neural networks trained with the back-propagation algorithm constitute the best example of a successful gradient based learning technique. Given an appropriate network

■ 第四阶段：流行度降低

- 20 世纪 90 年代中期，统计学习理论和以支持向量机为代表的机器学习模型开始兴起
- 相比之下，神经网络的理论基础不清晰、优化困难、可解释性差等缺点更加凸显，神经网络的研究又一次陷入低潮

A training algorithm for optimal margin classifiers

[BE Boser](#), [IM Guyon](#), [VN Vapnik](#) - ... of the fifth annual workshop on ..

... **ALGORITHM** In this Section, we highlight some important aspects of the algorithm... The description is split into a discussion of the qualities of

☆ 保存 引用 被引用次数: 17836 相关文章 所有 22 个版本

Regression shrinkage and selection via the lasso

[R Tibshirani](#) - *Journal of the Royal Statistical Society Series B* ..., 1996 - acad

We propose a new method for estimation in linear models. The 'lasso' minimizes the sum of squares subject to the sum of the absolute value of the coefficients being

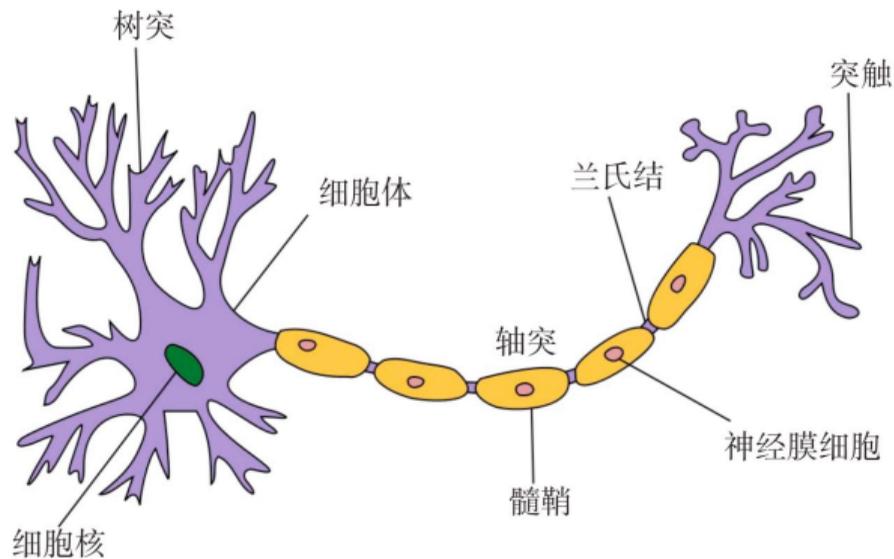
☆ 保存 引用 被引用次数: 61553 相关文章 所有 54 个版本 ✂

■ 第五阶段：深度学习的崛起

- 2006 年，Geoffrey Hinton 等人发现多层前馈神经网络可以先通过逐层预训练，再用反向传播算法进行精调的方式进行有效学习
- 2013 年，Geoffrey Hinton 的学生发明了第一个现代深度卷积网络模型 AlexNet，它是深度学习技术在图像分类上取得真正突破的开端
- 2015 年，Kaiming He 提出了残差神经网络 ResNet
- 2016 年，AlphaGo 战胜围棋世界冠军、职业九段棋手李世石
- 2022 年，OpenAI 正式发布聊天机器人 ChatGPT
- 2024 年，John Hopfield 和 Geoffrey Hinton 获得物理学诺贝尔奖

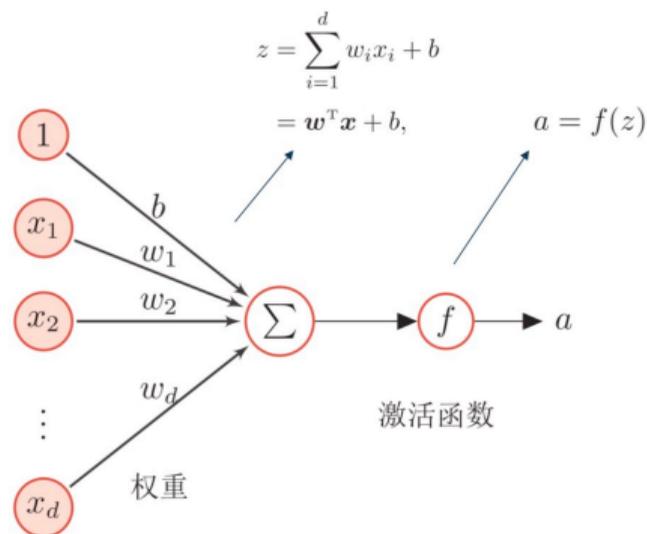
神经元

- 单个神经细胞只有两种状态——兴奋和抑制



激活函数的性质

- 连续并可导的非线性函数
- 激活函数及其导函数要尽可能的简单
- 激活函数的导函数的值域要在一个合适的区间内



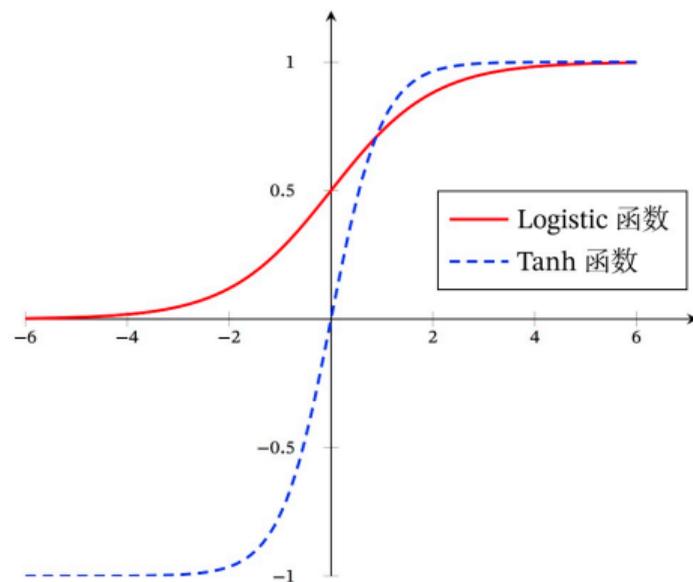
常见激活函数

■ Sigmoid 函数/Logistic 函数

$$\sigma(x) = \frac{1}{1 + \exp(-x)}$$

■ Tanh 函数

$$\tanh(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)}$$



常见激活函数

■ ReLU 函数

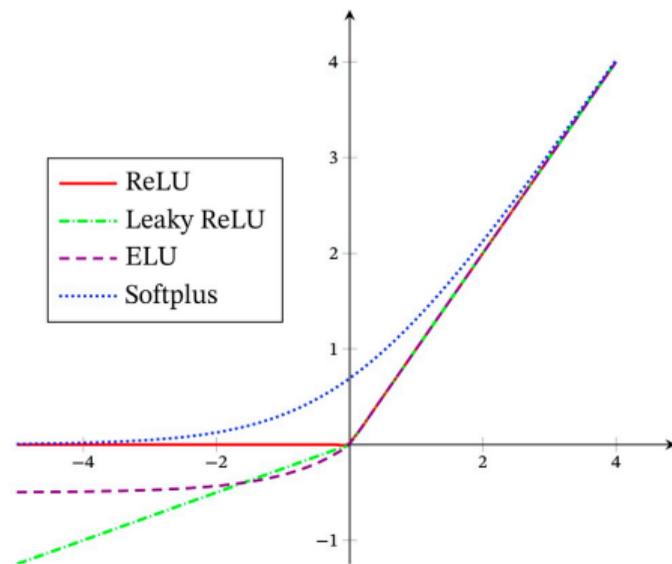
$$\text{ReLU}(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

■ LeakyReLU 函数

$$\text{LeakyReLU}(x) = \begin{cases} x, & x > 0 \\ \gamma x, & x \leq 0 \end{cases}$$

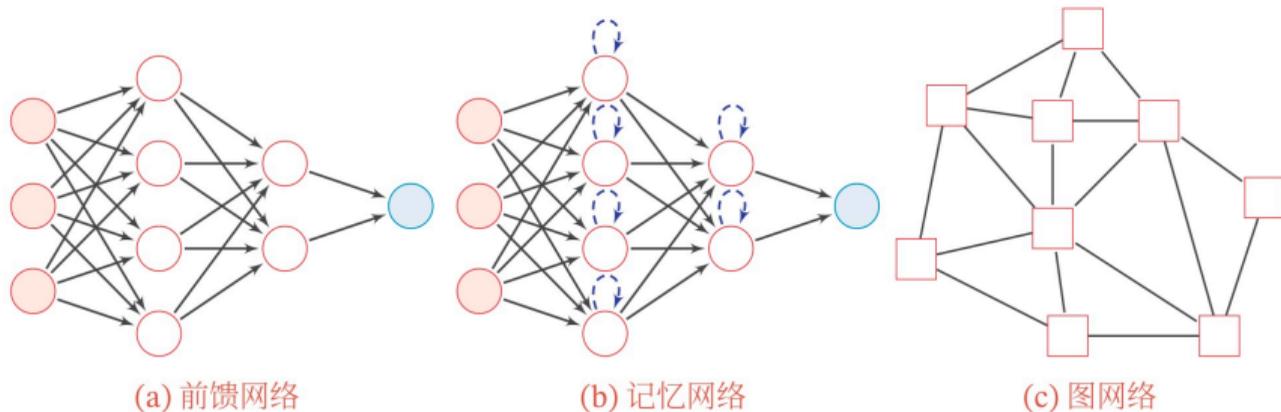
■ ELU 函数

$$\text{ELU}(x) = \begin{cases} x, & x > 0 \\ \gamma(\exp(x) - 1), & x \leq 0 \end{cases}$$



人工神经网络

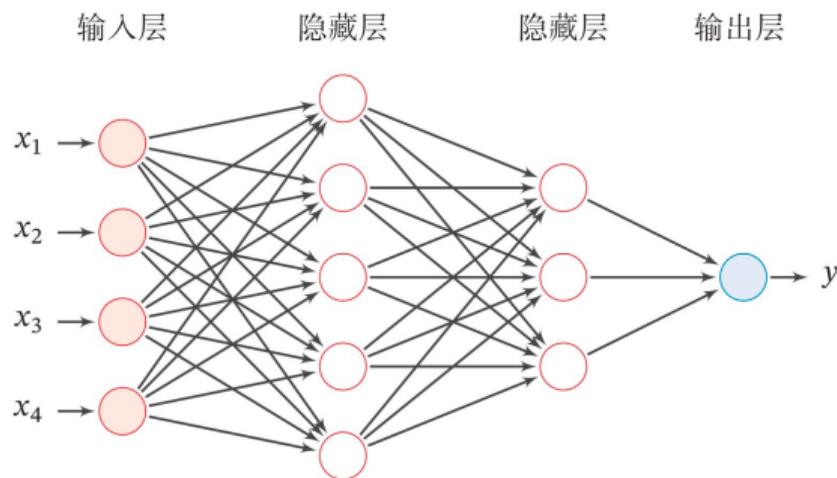
- 人工神经网络由大量的神经元以及它们之间的有向连接构成，需考虑
 - 神经元的激活规则
 - 网络的拓扑结构
 - 学习算法



- 6.1 引言
- 6.2 前馈神经网络
- 6.3 卷积神经网络
- 6.4 循环神经网络

■ 前馈神经网络（全连接神经网络、多层感知器）

- 各神经元分别属于不同的层，层内无连接
- 相邻两层之间的神经元全部两两连接
- 整个网络中无反馈，单向传播



- 前馈神经网络通过下面公式进行信息传播

仿射变换 $\mathbf{z}^{(l)} = \mathbf{W}^{(l)}\mathbf{a}^{(l-1)} + \mathbf{b}^{(l)}$

非线性变换 $\mathbf{a}^{(l)} = f_l(\mathbf{z}^{(l)})$

□ $\mathbf{z}^{(l)}$ 净活性值 (Net Activation)

□ $\mathbf{a}^{(l)}$ 活性值 (Activation)

- 前馈计算

$$\mathbf{x} = \mathbf{a}^{(0)} \rightarrow \mathbf{z}^{(1)} \rightarrow \mathbf{a}^{(1)} \rightarrow \mathbf{z}^{(2)} \rightarrow \dots \rightarrow \mathbf{a}^{(L-1)} \rightarrow \mathbf{z}^{(L)} \rightarrow \mathbf{a}^{(L)} = \phi(\mathbf{x}; \mathbf{W}, \mathbf{b})$$

通用近似定理

- 通用近似定理 (Universal Approximation Theorem), 1989
- 令 $\phi(\cdot)$ 是一个非常数、有界、单调递增的连续函数, \mathcal{J}_D 是一个 D 维的单位超立方体 $[0, 1]^D$, $C(\mathcal{J}_D)$ 是定义在 \mathcal{J}_D 上的连续函数集合。对于任意给定的一个函数 $f \in C(\mathcal{J}_D)$, 存在一个整数 M , 和一组实数 $v_m, b_m \in \mathbb{R}$ 以及实数向量 $\mathbf{w}_m \in \mathbb{R}^D$, 可以定义函数

$$F(\mathbf{x}) = \sum_{m=1}^M v_m \phi(\mathbf{w}_m^\top \mathbf{x} + b_m)$$

作为函数 f 的近似实现, 即

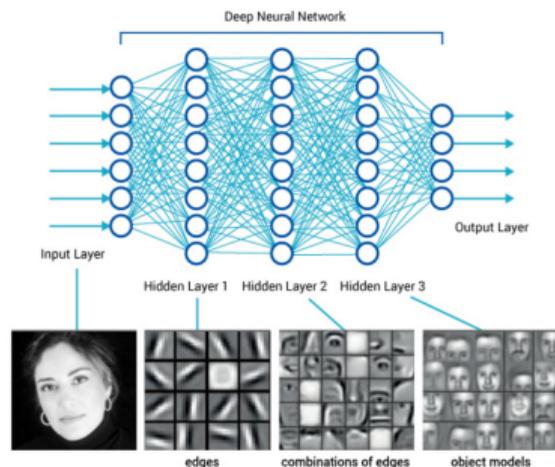
$$|F(\mathbf{x}) - f(\mathbf{x})| < \epsilon, \forall \mathbf{x} \in \mathcal{J}_D$$

其中 $\epsilon > 0$ 是一个很小的正数

- 神经网络可以作为一个“万能”函数进行复杂的特征提取

$$y = g(\phi(\mathbf{x}), \theta)$$

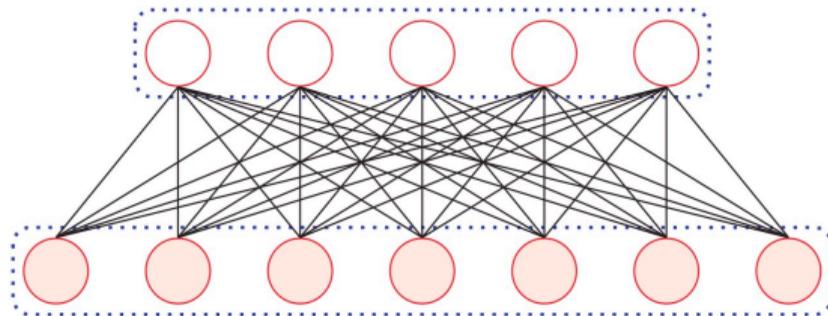
其中 $g(\cdot)$ 为线性或非线性的分类器，如 Logistic 函数、Softmax 函数



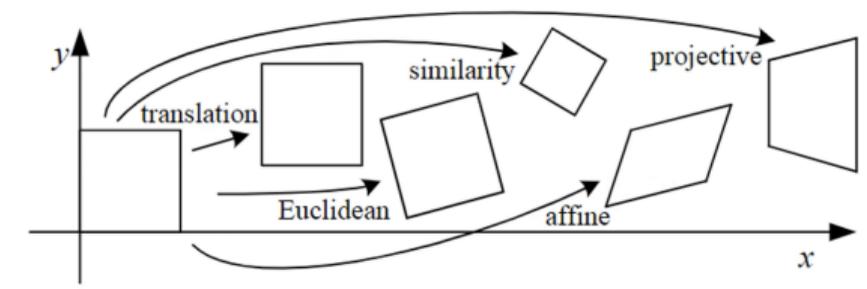
- 6.1 引言
- 6.2 前馈神经网络
- 6.3 卷积神经网络
- 6.4 循环神经网络

全连接前馈神经网络

- 权重矩阵的参数非常多



- 自然图像中的物体都具有局部不变性特征



- 卷积神经网络 (Convolutional Neural Networks CNN)
 - 一种前馈神经网络
 - 感受野: 视网膜上的特定区域, 只有这个区域内的刺激才能激活该神经元
- 卷积神经网络的特性
 - 局部连接
 - 权重共享
 - 空间或时间上的次采样

卷积

- 图像以二维矩阵的形式输入到神经网络，卷积定义为

$$Y = W * X$$

⇓

$$y_{ij} = \sum_{u=1}^U \sum_{v=1}^V w_{uv} x_{i-u+1, j-v+1}$$

1	1	1	1	1
-1	0	-3	0	1
2	1	1	-1	0
0	-1	1	2	1
1	2	1	1	1

*

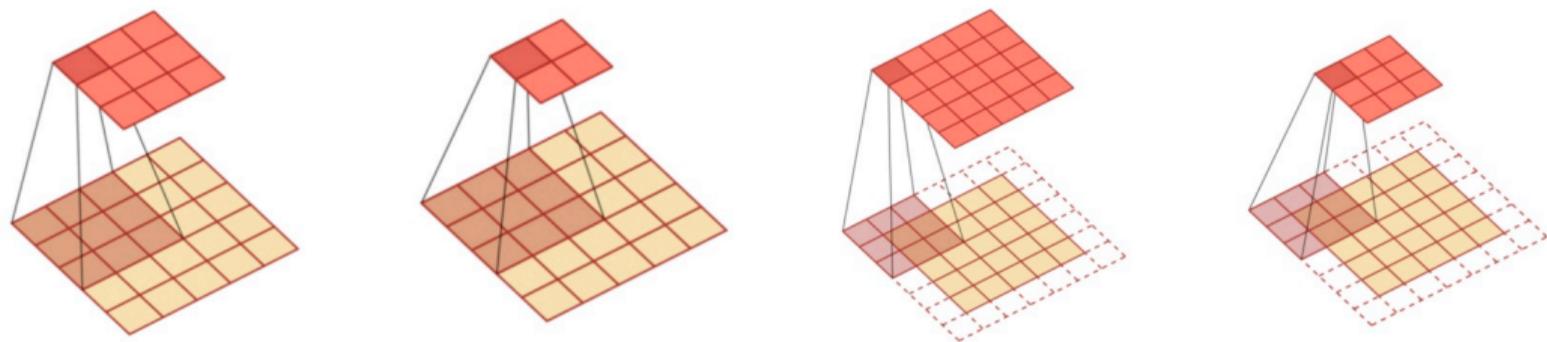
1	0	0
0	0	0
0	0	-1

=

0	-2	-1
2	2	4
-1	0	0

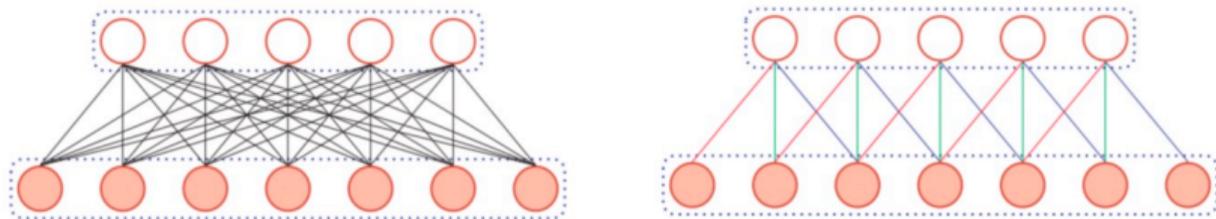
卷积

- 引入滤波器的滑动步长 T 和零填充 P , 卷积的结果可以分为三类
 - 窄卷积: 步长 $T = 1$, 两端不补零 $P = 0$
 - 宽卷积: 步长 $T = 1$, 两端不补零 $P = K - 1$
 - 等宽卷积: 步长 $T = 1$, 两端不补零 $P = (K - 1)/2$
- 目前的文献中, 卷积一般默认为等宽卷积



卷积神经网络

- 用卷积层代替全连接层



- 卷积操作的目标提取特征 \Rightarrow **翻转是不必要的！**
- 除非特别声明，卷积一般指“互相关”

$$y_{ij} = \sum_{u=1}^U \sum_{v=1}^V w_{uv} x_{i+u-1, j+v-1}$$

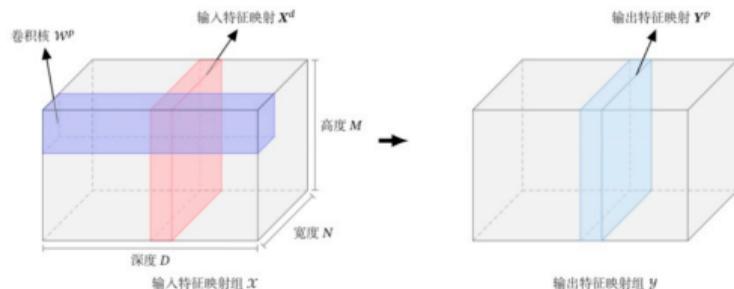
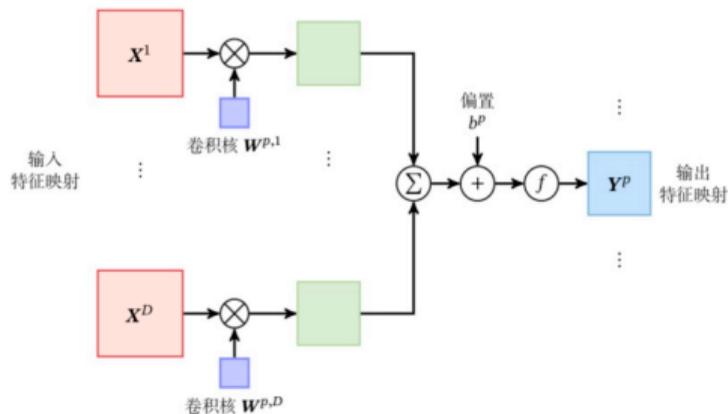
多个卷积核

■ 特征映射 (Feature Map)

□ 输入: D 个特征映射 $M \times N \times D$

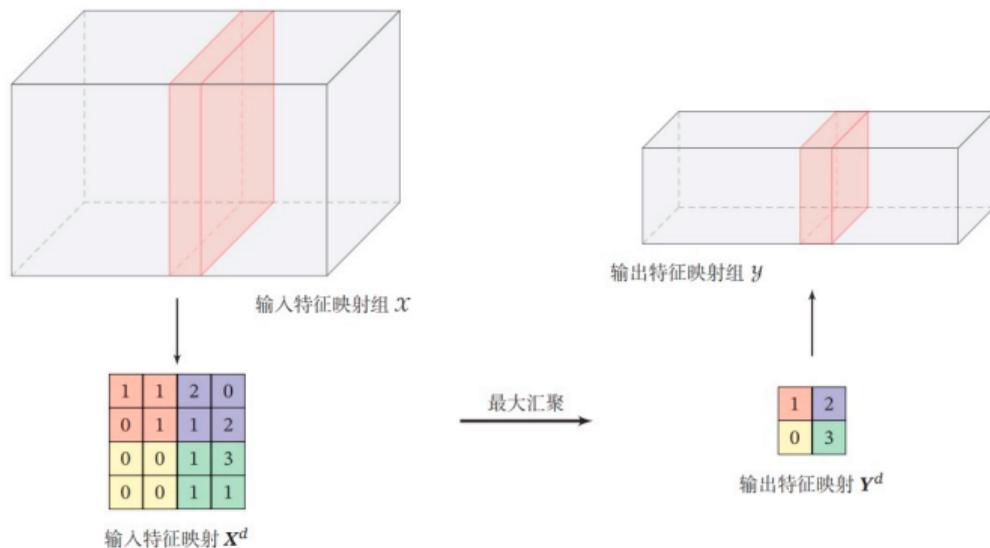
□ 输出: P 个特征映射 $M \times N \times P$

$$\mathbf{Z}^p = \sum_{d=1}^D \mathbf{W}^{p,d} \otimes \mathbf{X}^d + \mathbf{b}^p, \mathbf{Y}^p = f(\mathbf{Z}^p)$$



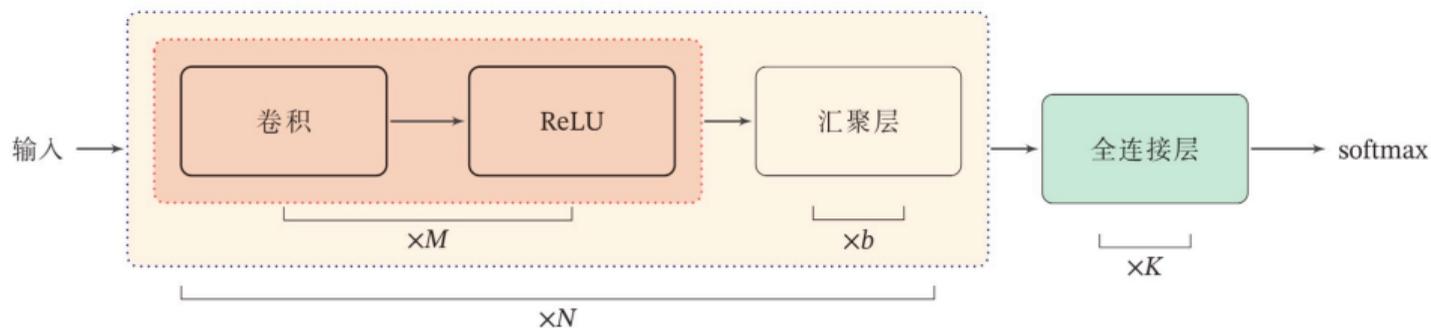
汇聚层 (Pooling Layer)

- 卷积层虽然可以显著减少连接的个数，但是每一个特征映射的神经元个数并没有显著减少
- 汇聚层，又称池化层或子采样层



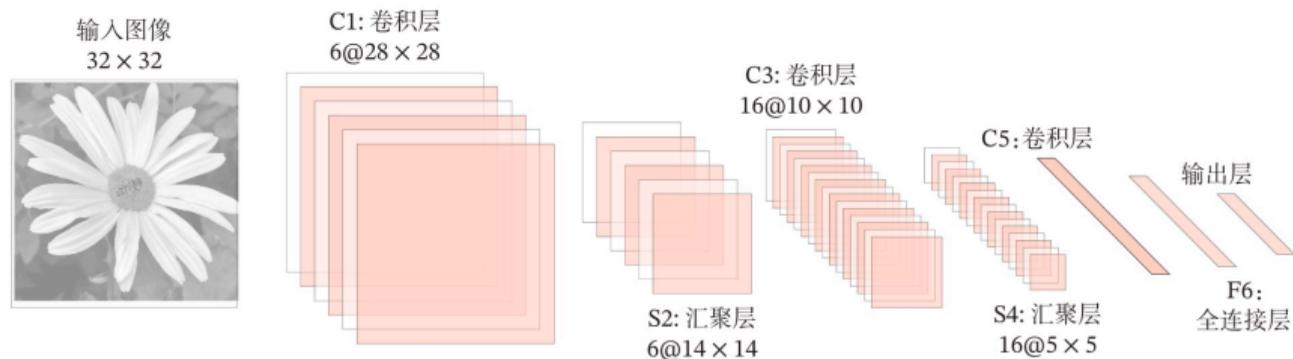
卷积网络结构

- 卷积网络是由卷积层、汇聚层、全连接层交叉堆叠而成
 - 小卷积大深度
 - 全卷积
- 典型结构: $M = 2 - 5$, $b = 0 - 1$, $N = 1 - 100$, $K = 0 - 2$



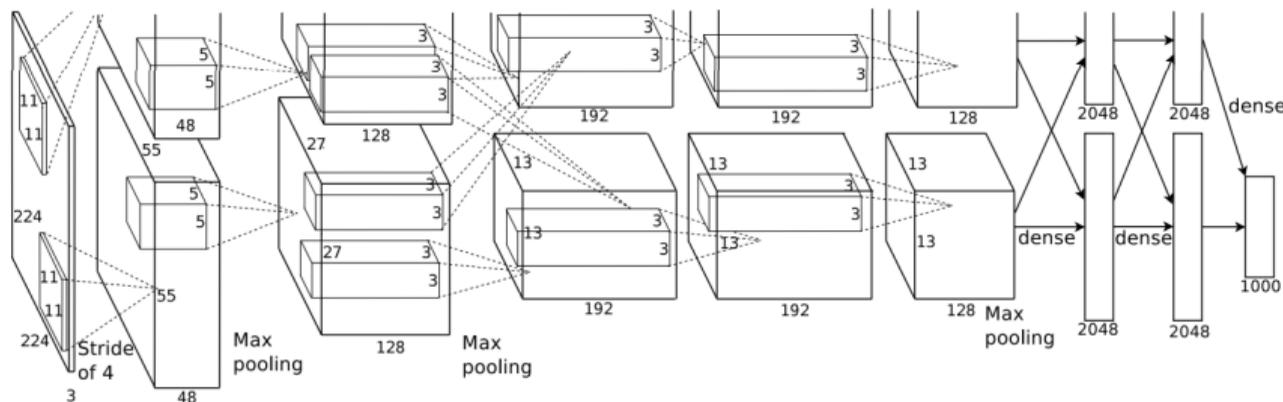
代表: LeNet-5

- LeNet-5 是一个非常成功的神经网络模型
 - 基于 LeNet-5 的手写数字识别系统在 90 年代被美国很多银行使用，用来识别支票上面的手写数字
 - 共有 7 层，输出层由 10 个径向基函数 (Radial Basis Function, RBF) 组成

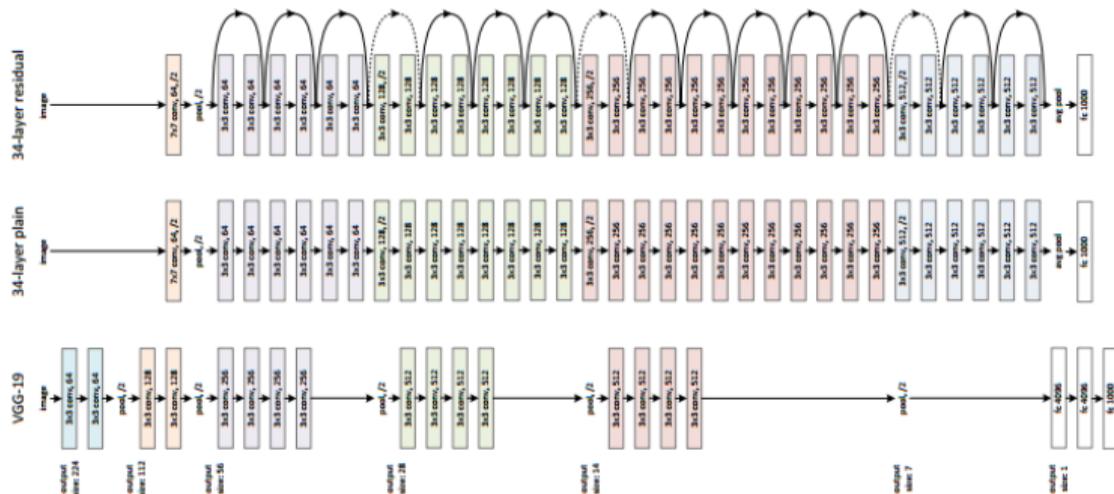


■ 第一个现代深度卷积神经网络模型

- 使用 GPU 进行并行训练, 采用 ReLU 作为非线性激活函数、Dropout 防止过拟合、数据增强
- 5 个卷积层、3 个汇聚层和 3 个全连接层

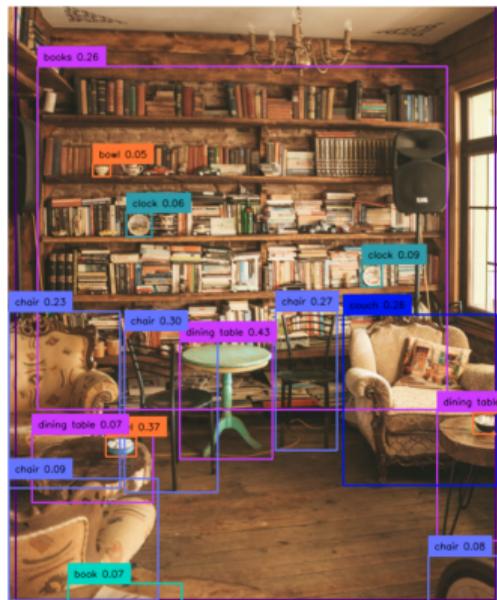
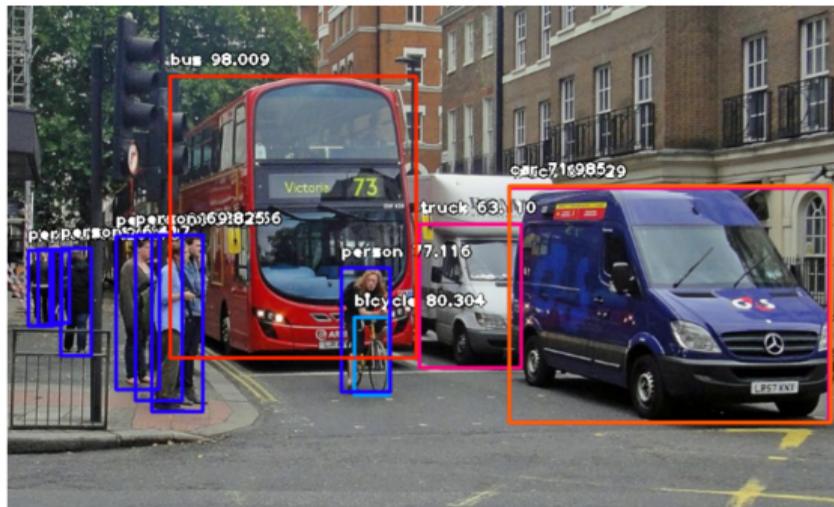


- 通过给非线性的卷积层增加直连边的方式来提高信息的传播效率
 - 假设在深度网络中，使用非线性单元 $f(\mathbf{x}, \theta)$ 去逼近目标函数 $h(\mathbf{x})$
 - 将目标函数拆分为恒等函数和残差函数 $h(x) = \mathbf{x} + (h(\mathbf{x}) - \mathbf{x})$



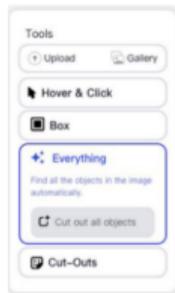
应用：目标检测

■ <https://pjreddie.com/darknet/yolo>



应用: 图像分割

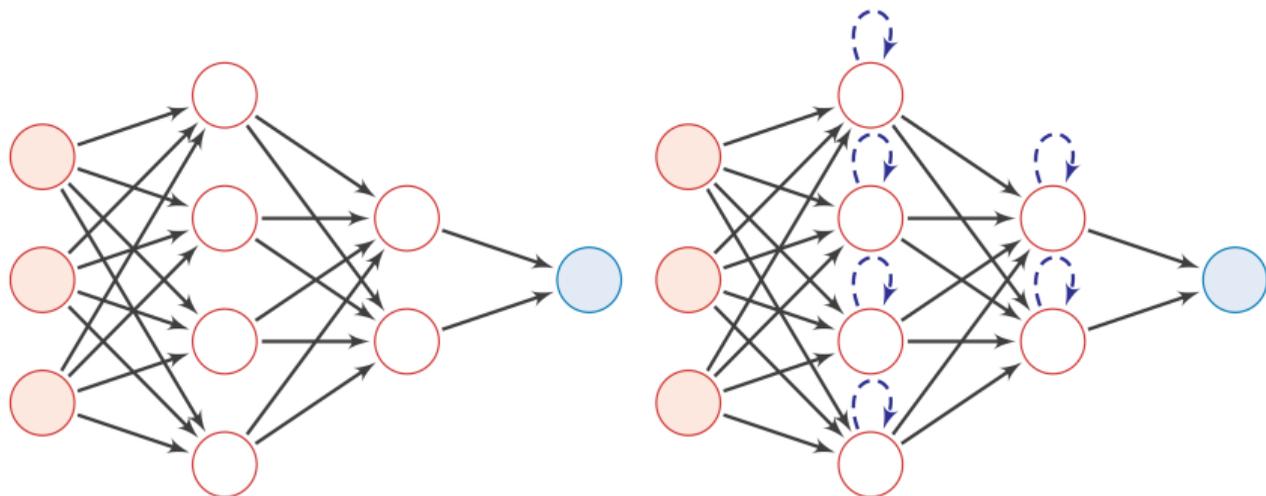
- <https://segment-anything.com/demo>



- 6.1 引言
- 6.2 前馈神经网络
- 6.3 卷积神经网络
- 6.4 循环神经网络

前馈网络

- 连接存在层与层之间，每层的节点之间是无连接的
 - 输入和输出的维数都是固定的，不能任意改变
 - 无法处理变长的序列数据

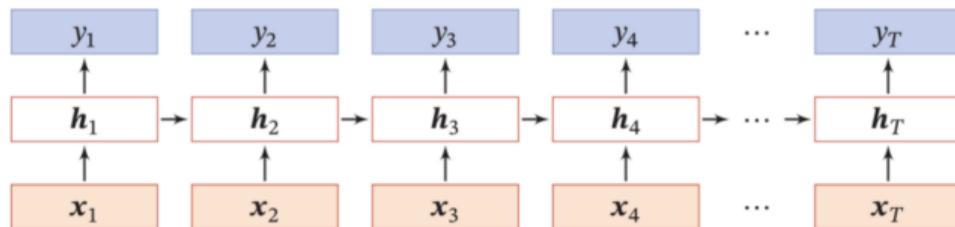
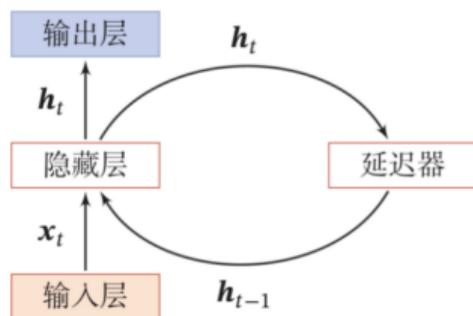


循环神经网络

- 循环神经网络通过使用带自反馈的神经元，能够处理任意长度的时序数据

$$\mathbf{h}_t = f(\mathbf{h}_{t-1}, \mathbf{x}_t)$$

- 可以看成是一个动力系统， \mathbf{h}_t 也称为状态 (State)
- 循环神经网络比前馈神经网络更加符合生物神经网络的结构
- 广泛应用于语音识别、语言模型以及自然语言生成等任务上



通用近似定理

- 如果一个完全连接的循环神经网络有足够数量的 Sigmoid 型隐藏神经元，它可以以任意的准确率去近似任何一个非线性动力系统

$$\mathbf{s}_t = g(\mathbf{s}_{t-1}\mathbf{x}_t)$$

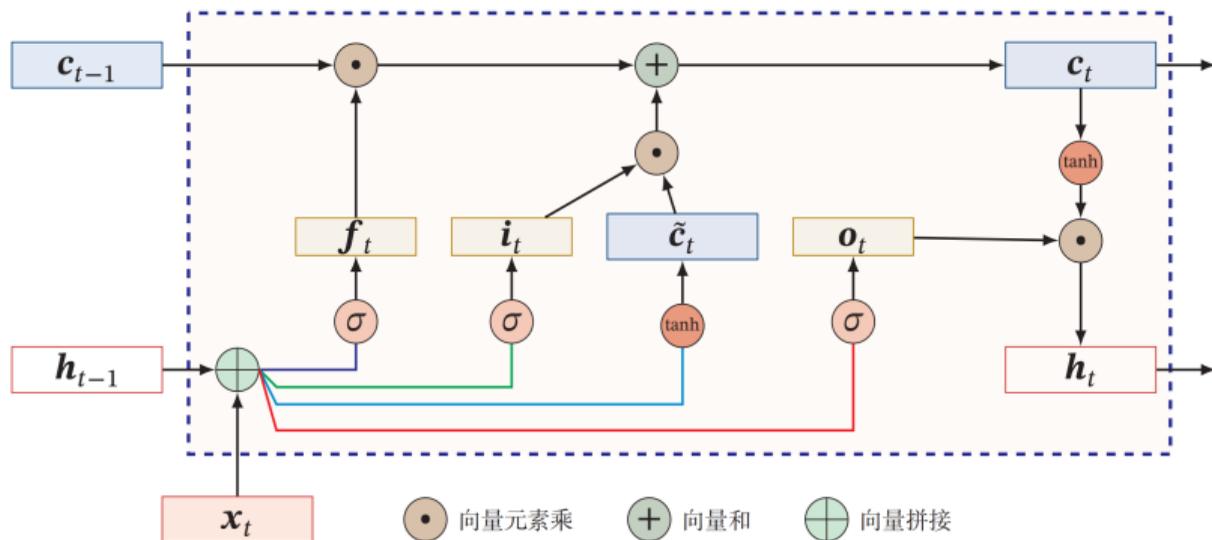
$$\mathbf{y}_t = o(\mathbf{s}_t)$$

其中 \mathbf{s}_t 为每个时刻的隐状态， \mathbf{x}_t 是外部输入， $g(\cdot)$ 是可测的状态转换函数， $o(\cdot)$ 是连续输出函数，并且对状态空间的紧致性没有限制

- 图灵完备是指一种数据操作规则，比如一种计算机编程语言，可以实现图灵机的所有功能，解决所有的可计算问题。目前主流的编程语言（比如 C++、Java、Python 等）都是图灵完备的
- 所有的图灵机都可以被一个由使用 Sigmoid 型激活函数的神经元构成的全连接循环网络来进行模拟

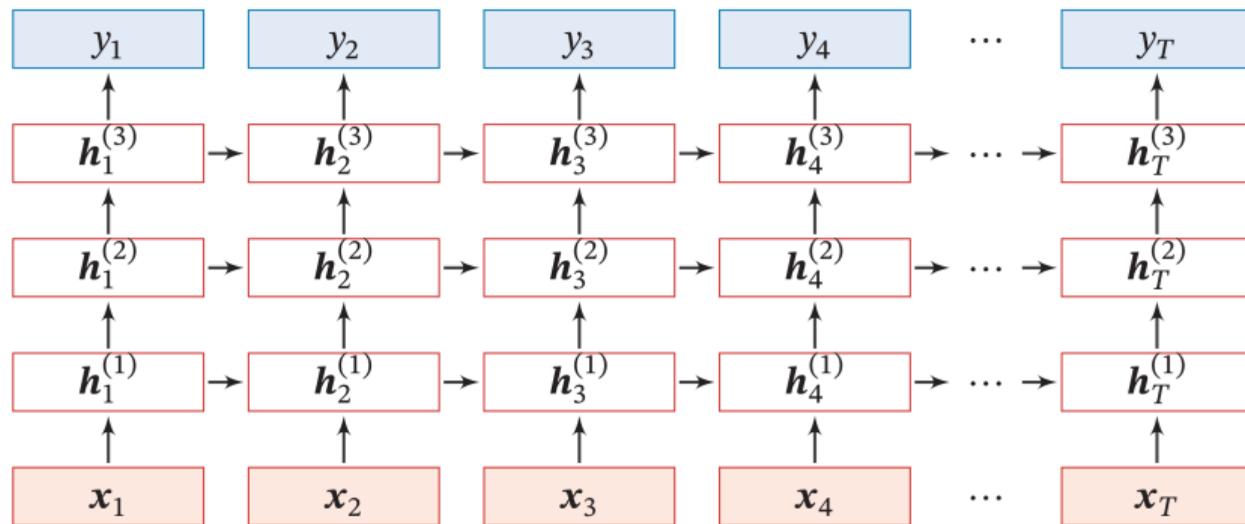
代表: 长短期记忆网络

- 引入门控机制来控制信息的累积速度，改善循环神经网络的长程依赖问题



代表: 堆叠循环神经网络

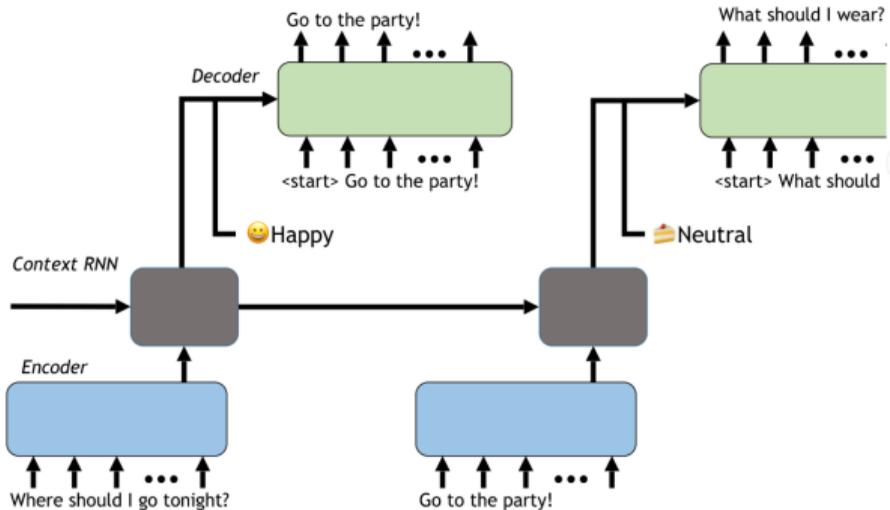
- 通过增加循环神经网络的深度，增强循环神经网络的能力



- <https://github.com/phunterlau/wangfeng-rnn>

我在这里中的夜里
就像一场是一种生命的意义
就像我的生活变得在我一样
可我们这是一个知道
我只是一天你会怎吗
可我们这是我们的的是不要为你
我们想这有一种生活的时候

■ <https://chatgpt.com>



ChatGPT 4o mini ▾

登录

模仿汪峰写一首歌



追逐梦想

(副歌) 在那遥远的天边, 星光闪烁不停歇,
我在风中奔跑, 心跳与梦想交织。
无畏艰难险阻, 勇敢面对每一次,
让我们一起飞, 追逐那永恒的光辉。

(第一节)
曾经的日子, 孤独在心里流淌,
跌倒无数次, 依然不肯放弃希望。
回首往昔, 泪水伴着笑容绽放,
每一次坚持, 都是生命的乐章。

循环神经网络总结

■ 优点

- 引入记忆
- 图灵完备

■ 缺点

- 长程依赖问题
- 记忆容量问题
- 并行能力

■ 更多

- 注意力机制/Transformer/大模型
- 深度强化学习

Q&A

Thank you!

感谢您的聆听和反馈