

绪论

- 图灵测试：机器说谎，人说实话，判断哪个是机器
对的 $P \leq 70\%$ 则机器胜

- 七个问题：
 - ① 自动的计算机
 - ② 如何编程让计算机语言
 - ③ 神经网络
 - ④ 计算规模理论 (复杂度)
 - ⑤ 自我学习提高
 - ⑥ 归纳演绎
 - ⑦ 随机性与创造力

- 1956 达特茅斯会议 麦卡锡等人 AI 领域诞生
↳ 会议上还提了 α - β 剪枝

- 神经网络** 历史 1943 麦卡洛克、皮茨 首次提出神经元数学模型

~~赫伯法则~~ \rightarrow 人在决策时并非完全理性，而是在认知能力、信息获取和计算资源受限的条件下做出“足够好”而非最优

- 自动定理证明：证明 \rightarrow 推理 在巨大推理空间搜一条公理到结论 的决策

- 下棋 1997 IBM 深蓝击败 国际象棋冠军，机器在复杂的路径智力博弈中首次战胜人类

- 自然语言处理 (NLP)

~~遗传算法~~ \rightarrow ~~强化学习~~

1955 乔姆斯基文法分层理论 0、1、2、3 型文法

- 神经元：神经网络的基本构建块，输入 + 计算 + 激活

加权求和 + 加偏置 权重发散

Hebbian rule: 2 神经元若经常被同时激活，它们之间的连接

$$\therefore w = \frac{w}{\|w\|}$$

赫伯法则

就会增强

$$\Delta w = \eta y x$$

学习率 前、后神经元 x, y



模拟思维

盲目搜索 { dfs 八皇后：每一行要放一个，失败则回退搜下一个
深度问题？深度限制（要合理）死循环？记录 path
bfs 空间消耗大，效率低 否则找不到解

第一解 dfs 并非最优 bfs 最优

启发式搜索 A^* 与 A^* 区别 $h(x) \leq h^*(x)$ 启发函数不超过真实代价

A^* $f(x) = h(x) + g(x)$ 实际代价
启发函数 min path { 欧氏距离
曼哈顿距离

$h(x)$ 评价：扩展节点数 解放到 closed

初始化 open 表 \rightarrow 找 min $f(x)$ \rightarrow 邻居 $h, g \rightarrow$ 更新 open 表
若达到目标 则退出

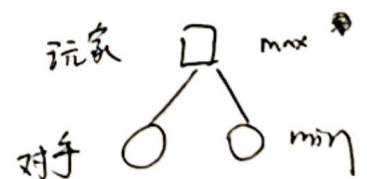
八数码
 $h(x)$ { 不在位数
离目标距离

对抗搜索 2- β 剪枝 $\min = \alpha$
 $\max = \beta$

max 层 更新 $\alpha = \max(\alpha_p, \alpha_c, \beta_c)$ min 层更新 $\beta = \min(\beta_p, \alpha_c, \beta_c)$

剪枝 $\alpha \geq \beta$

2 类 { α 剪枝 MIN 层
 β 剪枝 MAX 层



蒙特卡洛搜索：选择、扩展、模拟、反向传播

值得探索的 子节点 至结束 回溯更新节点信息
节点 加入树





模拟学习

无监督 无标签
 有监督 SL 有标签
 强化 RL 智能体 \leftrightarrow 环境

强化学习: 智能体与环境交互, 学习一条策略, 最大化长期累积回报



MDP 马尔可夫决策过程 $M = (S, A, P, R, \gamma)$ 折扣因子 $[0, 1]$

(动态 + 白盒环境)

不断试错 五元组
 $\downarrow \quad \downarrow \quad \downarrow \quad \downarrow$
 状态空间 动作空间 转 即时奖励函数

$$P(s'|s, a) \quad R(s, a) \quad R(s, a, s')$$

动/静: 可不可转移, 单/多步

白/黑盒: 变量与目标的关系有无公式

$$S_0 \xrightarrow{a_0, R(S_0, a_0)} S_1 \xrightarrow{a_1, R(S_1, a_1)} S_2 \dots$$

(累积奖励) 总回报 $R(S_0, a_0) + \gamma R(S_1, a_1) + \gamma^2 R(S_2, a_2) \dots$

历史: H_t 一直到 t 为止所有可观测变量 策略: $\pi(a|s)$

状态: $S_t = f(H_t)$ 奖励: $R(s, a)$ 与 γ

贝尔曼方程

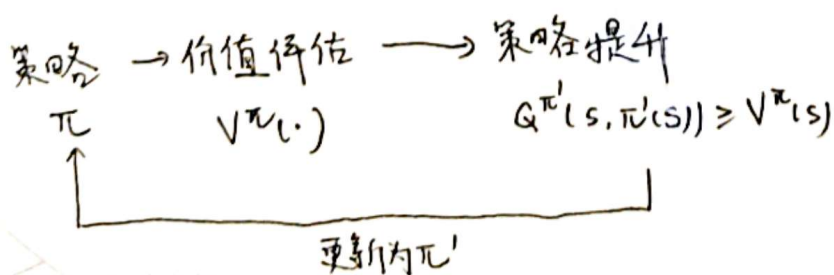
期望累积奖励 $= E[G_t(s) | S_t = s]$

$$\begin{cases}
 V_{\pi}(s) = \sum_{a \in A} \pi(s, a) q_{\pi}(s, a) & \text{价值函数 (a未知)} \\
 q_{\pi}(s, a) = \sum_{s' \in S} P(s'|s, a) [R(s, a, s') + \gamma V_{\pi}(s')] & \text{价值-动作函数 (a已知)} \\
 & = E[G_t(s) | S_t = s, A_t = a]
 \end{cases}$$

互相代入可得递推关系

蒙特卡洛价值估计: 用 π 在 S 中采 N 个样本, 用均值累积奖励估计 $V_{\pi}(s)$

策略提升定理: $\pi'(s) = \arg \max_a q_{\pi}(s, a)$ 则 $V_{\pi'}(s) \geq V_{\pi}(s) \quad \forall s$



无监督 K均值聚类 硬分配

- ① K个类, 初始质心
 - ② 归类, 欧氏距离
 - ③ 更新每个类质心
 - ④ 继续迭代 $\begin{cases} \text{max 次数} \\ \text{质心不动} \end{cases} \rightarrow \text{break}$
- (1, 1) (2, 2) (10, 10)
- $(\frac{1+2+10}{3}, \frac{1+2+10}{3})$

EM算法 软分配

(含隐变量), 以P分组, 最大似然估计参数



有监督 通过环境直接得标准答案 还可加激励 sigmoid $\sigma(x) = \frac{1}{1+e^{-x}}$

步骤: 1. 给出函数 $y = wx + b$ 表示

机器学习

分类: 交叉熵

2. 定义 Loss $L(\theta)$ θ 包含所有参数

3. 优化 optimize 更新所有参数

反向传播算法
Hinton

网络不做向更深: 过拟合!

模型容量太大, 会过度记住训练数据细节

s.t. 训练集表现优秀 新数据上泛化能力差, 性能↓
test data

强化 三类方法

基于价值 \underbrace{TD}_{\times} , \underbrace{MC}_{\times} , $\underbrace{DP}_{\checkmark}$ TD结合MC与DP优点, 单步更新无需完整轨迹

基于策略 $\underbrace{reinforce}_{\times}$ $\pi \rightarrow$ 每步求 $G_t \rightarrow L(\theta) = -G_t \cdot \log \pi_{\theta}(a_t | s_t)$

Actor-Critic \downarrow 梯度下降更新 θ

$G_t > 0$ 奖励, 更易做此策略



模拟思维(续)

知识图谱 三元组 (实体, 关系, 实体)

P AN
T TP TN
F FP FN

$$rec = \frac{TP}{TP+FN}$$

$$pre = \frac{TP}{TP+FP}$$

$$acc = \frac{TP+TN}{TP+TN+FP+FN}$$

$$F_1 = \frac{2P \cdot r}{p+r}$$

HMM例子 观测集 隐状态集 发射

$\lambda = (S, O, A, B, \pi)$
初始状态分布 $(0.6, 0.4)$
转 \downarrow 隐 \rightarrow 隐

隐 S = 晴 | 雨 O = 吃几 | 冰淇淋
A $\begin{pmatrix} s \rightarrow s & s \rightarrow r \\ r \rightarrow s & r \rightarrow r \end{pmatrix}$ B $\begin{matrix} & 1 & 2 & 3 \\ s & 0 & 0 & 0 \\ r & 0 & 0 & 0 \end{matrix}$

HMM HMM对应

三类问题

	输入	输出	算法
解码	观 + 参	隐序列	维特比
评估	观 + 参	P观	前向
学习	观	参	前向-后向

AB 给 O 求 S (天气)
给 O AB 求 P(O|入)



模拟语言

双向 BERT : Transformer 的 Encoder 堆叠

词向量

训练: 自回归语言模型
单向 GPT : 生成类

理解类 (分类, 问答)

LSTM 1997
RNN 1987

transformer 2017
bert

self-attention

位置编码

输入
三嵌入
Token 语义
Segment 时间 MLM
Position 位置 掩码掩上文
让它预测

NSP
学习句间关系

潜在语义分析 LSA : 单词-文档矩阵

↓

神经网络模型 2003 本吉奥 首个 输入(词向量) → 隐藏 → 输出

word = vec one-hot 的词向量 → 词的联合分布
2013

架构 { skip-gram 中心词 → 上下文 细致
CBOW 上下文 → 中心词 快

作用: 优化词向量, 语义相近的词在向量空间更接近
通过上下文预测任务

LSTM 处理长期依赖 (防梯度消失或爆炸) x_t 输入 上一隐藏 h_{t-1}

门控机制 { 遗忘门 $f_t = G[W_f(h_{t-1}, x_t) + b_f]$
输入门 $i_t = G[W_i(h_{t-1}, x_t) + b_i]$ $\tilde{C}_t = \tanh(W_C(h_{t-1}, x_t) + b_C)$
输出门 $O_t = G[W_o(h_{t-1}, x_t) + b_o]$ $C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t$
 $h_t = O_t \odot \tanh(C_t)$ 逐元素乘 更新

Transformer NLP 革新, 取代 LSTM

序列长度不变, 词维度分份, 最后拼起来
cell 状态

输入 a^i $q^i = W^q a^i$ $k^i = W^k a^i$ $v^i = W^v a^i$ 共 i 个 token

编码 词嵌入 + 位置编码 → 多头自注意力

$$A(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

每层有 解码 masked 自注意力 (自回归) + 跨注意力 → Y-out
归一化 (Fix + x) 输出 预测的 token X-out
↓ Linear + softmax
词概率

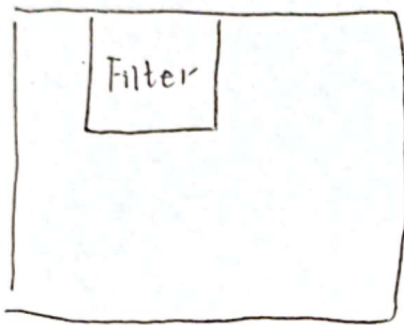


扫描全能王 创建



模拟视觉

CNN 卷积神经网络



通道数

$$out = \text{Filter 数量}$$

in RGB 3

灰 1

每个 Filter 的 channel

与原图一致

Transformer

① 编码器将输入序列变成

"上下文信息"向量

② 解码器依据输出生成目标序列

① 多头自注意力 Attention (Q, K, V)

$$\uparrow \text{很多个层} = G\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

每层之间 ~~并~~ 残差连接并归一化 X_{out}

② 掩码 & 多头自注意力

$$\text{softmax}\left(\frac{QK^T}{\sqrt{d_k}} + \text{mask}\right) \quad \begin{matrix} \text{位于 } j \text{ 时, 只考虑 } \leq j \text{ 的} \\ > j \text{ 的为 } -\infty \rightarrow \text{softmax} = 0 \end{matrix}$$

用于生成序列 $\rightarrow Y_{out}$

同时
每层之间

残 + 归

② 交叉多头自注意力

$$X_{out} \text{ 与 } Y_{out} \rightarrow Z_{out}$$

$$\begin{matrix} Q \leftarrow Y_{out} \\ K, V \leftarrow X_{out} \end{matrix} \quad \begin{matrix} \downarrow \\ \text{Linear} + \text{softmax} \end{matrix}$$

词根无字

③ 输出预测的 token

VGG, Res Net

3x3 $F(X) + X$
maxpool

转置卷积 $H_{in} = \frac{H_{out} + 2p - f}{\text{stride}} + 1$

(上采样过程, 恢复分辨率)

ViT 图像切 patch 加位置编码和 class token

给 Encoder (Transformer 架构)

最后输出的是 class token 预测

GAN 生成式对抗网络 对抗训练: 生成问题 \rightarrow min-max 博弈

图片数据化, 用神经网络去拟合 \rightarrow 步骤: train data; 设计网络; 优化参数; 生成新图

battle { 生成器 $G(z; \theta_g)$
判别器 $D(x; \theta_d)$

① 取 data 1. 真样本 $\rightarrow 1$ 2. 假样本 $\rightarrow 0$

$$\min_G \max_D V(D, G) = E_{x \sim p_{data}} [\log D(x)] +$$

$$E_{z \sim p_z} [\log (1 - D(G(z)))]$$

④ 训练 G 固定一个更新另一个 交替随机梯度下降



扫描全能王 创建

VAE 变分自编码器

① 编码: 输入 \rightarrow 概率分布 (不是固定值描述特征)

连续:

② 解码: 采样 \rightarrow 重建与原始输入相似的数据

不笑 \leftrightarrow 笑

损失函数 2 部分 1. 生成尽可能像原始 2. 分布尽可能接近 $N(0,1)$ 变化过程 \checkmark

~~diffusion~~ diffusion 扩散模型 加噪, 去噪
(前向) (反向)

加噪 U-Net



$$x_t = \sqrt{\alpha_t} x_0 + \sqrt{1 - \alpha_t} \epsilon, \quad \epsilon \sim N(0, I)$$

模拟听觉

sound power 单位 t 产生的 E, 单位 W
sound intensity 单位 S 的 power, 单位 W/m^2

分帧: 语音短时间内
"近似平稳"

$$dB = dB(I) = 10 \log_{10} \left(\frac{I}{I_{70dB}} \right)$$

欧拉公式 $e^{i\theta} = \cos\theta + i\sin\theta$
$$g(t) = \int c_f \cdot e^{i2\pi f t} dt$$

傅里叶变换: 时域信号 \rightarrow 不同 f 的正弦波叠加, 得到频域表示

频率 f 处正弦分量的幅度 df 和相位 φ_f

$$\begin{cases} \arg \max_{\varphi \in [0, 1)} \left| \int s(t) \sin(2\pi(f t - \varphi)) dt \right| \\ \max_{\varphi \in [0, 1)} \left| \int s(t) \sin(2\pi(f t - \varphi)) dt \right| \end{cases}$$

Mel 谱: 人耳对声音的感知是对数型的 (低 f 分辨率高, 高 f 低)

$$Mel(f) = 2595 \log_{10} \left(1 + \frac{f}{700} \right)$$

\sqrt{DCT} 离散余弦变换

MFCC: 对 Mel 频谱做进一步压缩, 提取语音谱包络特征

音高 pred
能量 pred
(每块到帧)

Fast Speech 2 非自回归, 可控的 TTS 模型

Embedding \rightarrow FFT $E \rightarrow$ 时长预测器
 \downarrow
Mel 谱 \leftarrow FFT $D \leftarrow$ 长度管理 (控语速)

tacotron

字符序列 \rightarrow Mel 谱图

编码器 \rightarrow 语音波形

Encoder-Attention-Decoder

(TTS) 自回归
输入 \rightarrow 输出

输入

输出

语速不可控, 生成慢



扫描全能王 创建