中国科学院大学
University of Chinese Academy of Sciences

深度学习进阶

AI DISCOVERY

目录

AI DISCOVERY

AI DISCOVERY

目录

AI DISCOVERY

AI DISCOVERY

迁移学习

AI DISCOVERY

迁移学习概念

图像中的迁移

文本中的迁移

什么是迁移学习

AI DISCOVERY

迁移学习

迁移学习

迁移学习

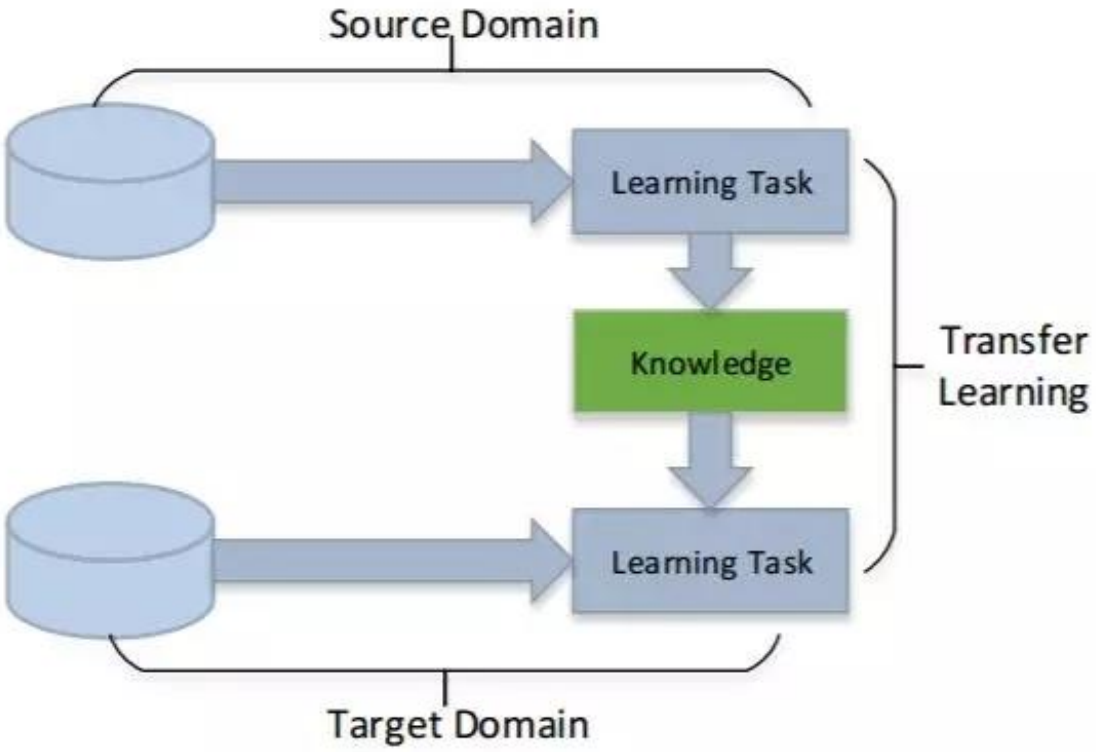
- 人类很自然就具备举一反三的迁移能力：
  - 婴儿学会爬行，学走路就很容易了；
  - 会骑自行车后，学骑摩托车就很简单了；
  - 会打羽毛球，再学打网球也就没那么难了。
- 计算机如何获得迁移能力？

AI DISCOVERY

迁移学习的定义

AI DISCOVERY

- 迁移学习是把一个领域（即**源领域，Source Domain**）的知识，迁移到另外一个领域（即**目标领域，Target Domain**），使得目标领域能够取得更好的学习效果。
- 深度迁移学习是研究如何通过深度神经网络有效地传递知识。

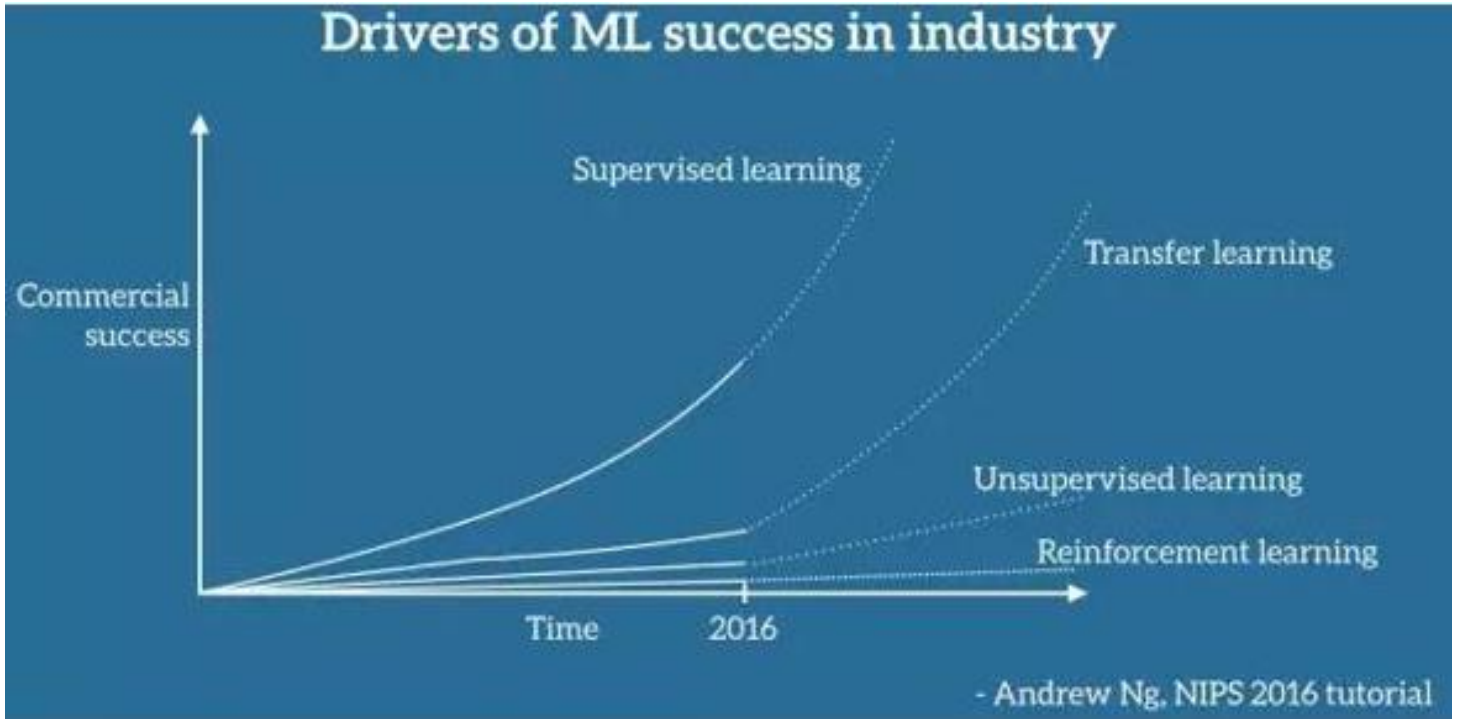


AI DISCOVERY

迁移学习的研究意义

AI DISCOVERY

- 目前大多数成功的工作都是依赖于大量有标签的数据，复用现有知识或数据，已有的大量工作不至于完全丢弃
- 不需要花费巨大代价去重新采集和标定庞大的新数据集，很多学习任务很难获得大量的有标签数据
- 对于快速出现的新领域，能够快速迁移和应用，体现时效性优势



在NIPS 2016 讲座上，吴恩达表示："在监督学习之后，迁移学习将引领下一波机器学习技术商业化浪潮。"

AI DISCOVERY

迁移学习的分类体系

AI DISCOVERY

迁移学习

特征迁移：通过源领域学习一个好的的特征表示，把知识通过特征的形式进行编码，并从源领域传递到目标领域，提升其任务效果。

样本迁移：源领域中数据的某一部分可以通过调整权重的方法重用，用于目标领域的学习。

参数迁移：任务之间共享相同的模型参数或者是服从相同的先验分布

关系知识迁移：相关领域之间的知识迁移

迁移学习

AI DISCOVERY

迁移学习概念

图像中的迁移

文本中的迁移

AI DISCOVERY

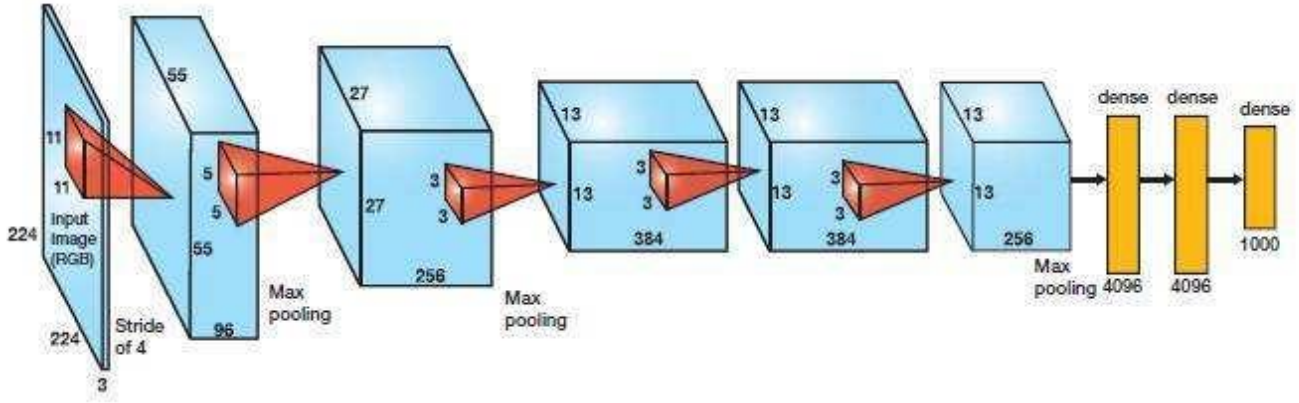
图像中的迁移学习

AI DISCOVERY

- 迁移学习是计算机视觉中的一种流行方法，它允许我们以节省时间的方式建立精确的模型。

  在计算机视觉中，通常是使用预先训练的模型来实现迁移学习。

- 预训练模型
  - 牛津VGG模型
  - 谷歌Inception模型
  - 微软ResNet模型



- 迁移学习过程

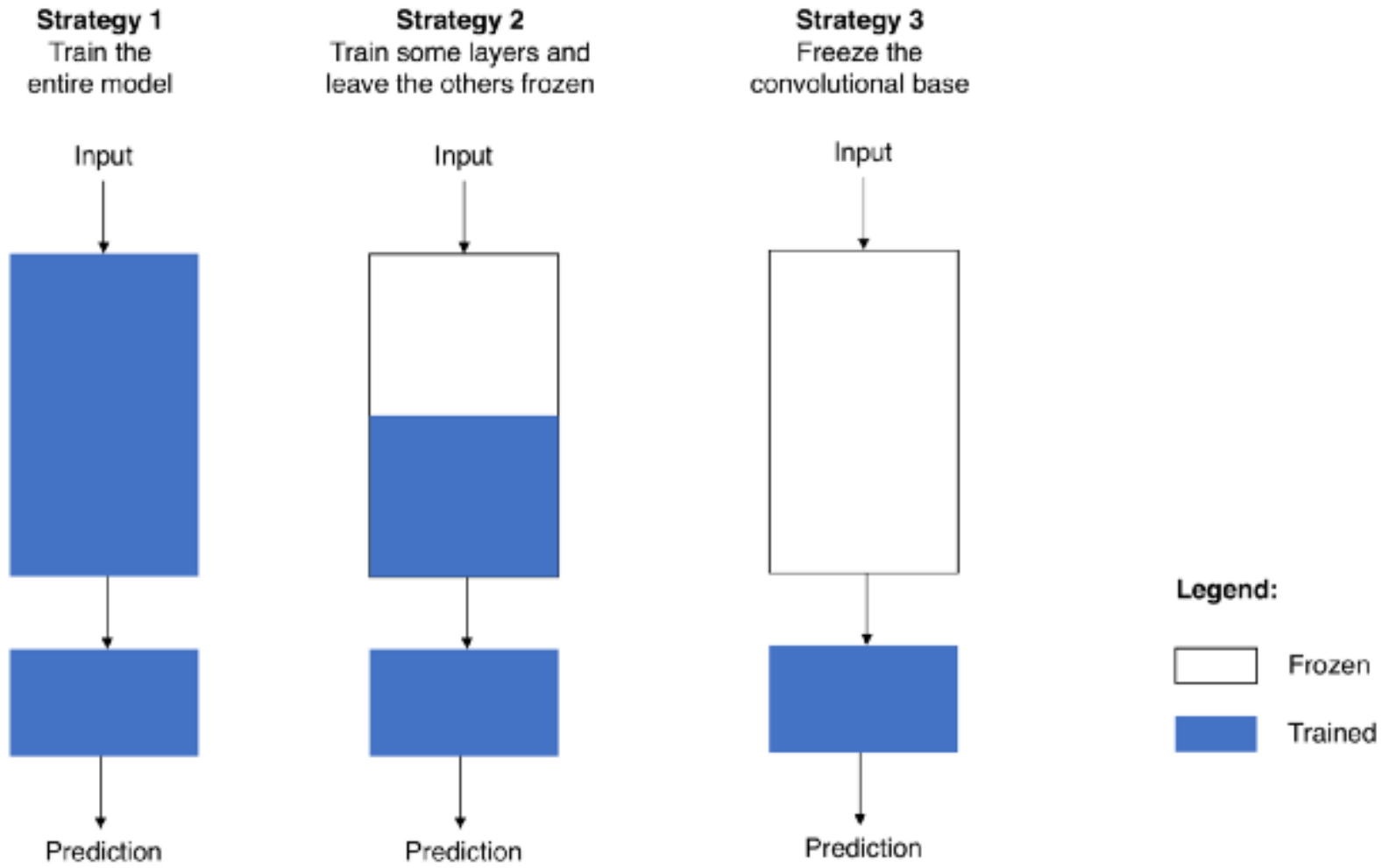
（1）选择预训练模型

（2）根据大小相似度矩阵对问题进行分类

（3）微调模型

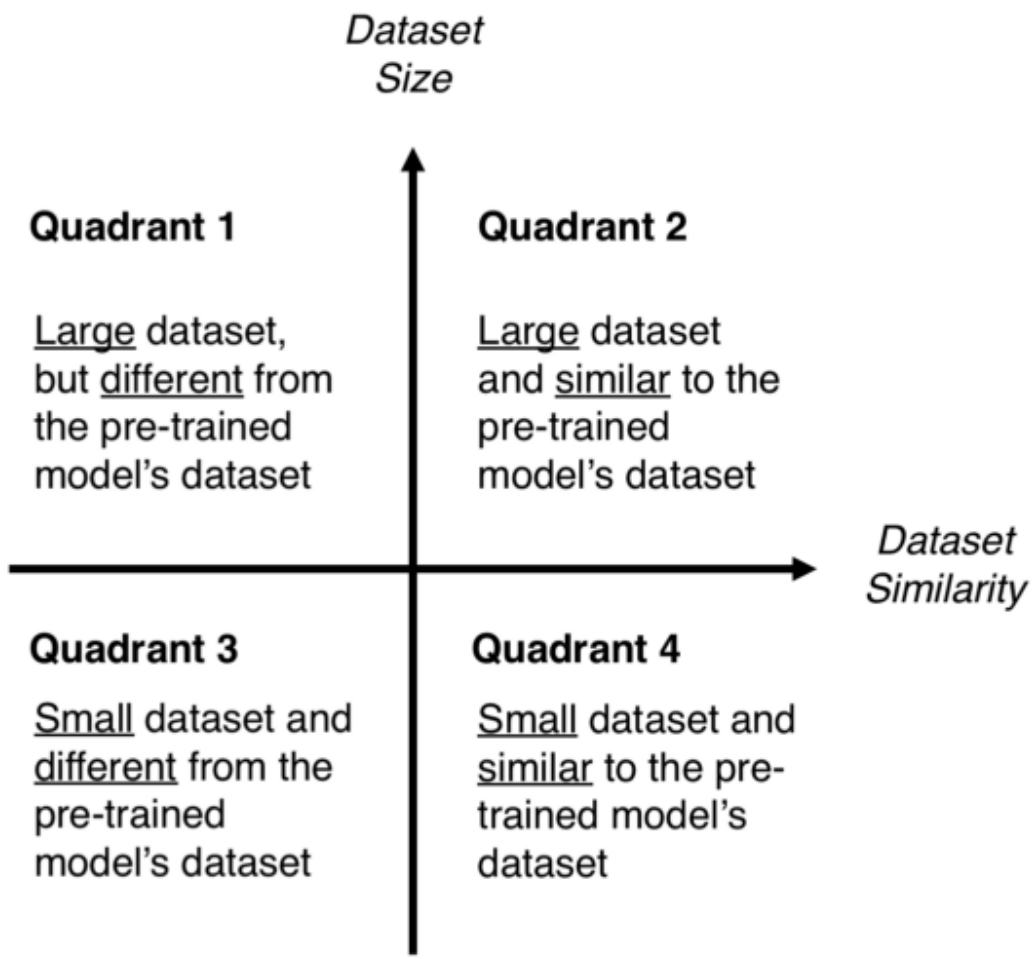
AI DISCOVERY

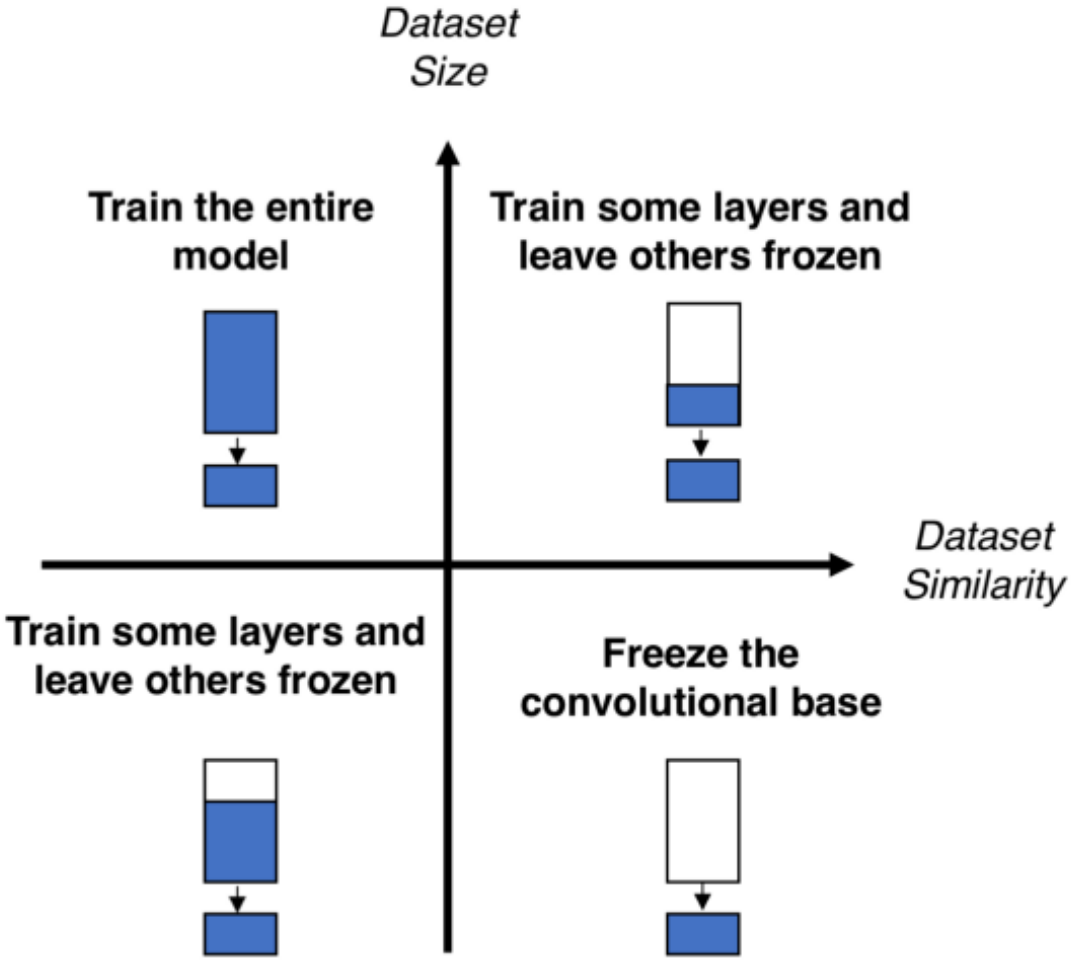
利用预训练模型的微调策略



利用预训练模型的微调策略

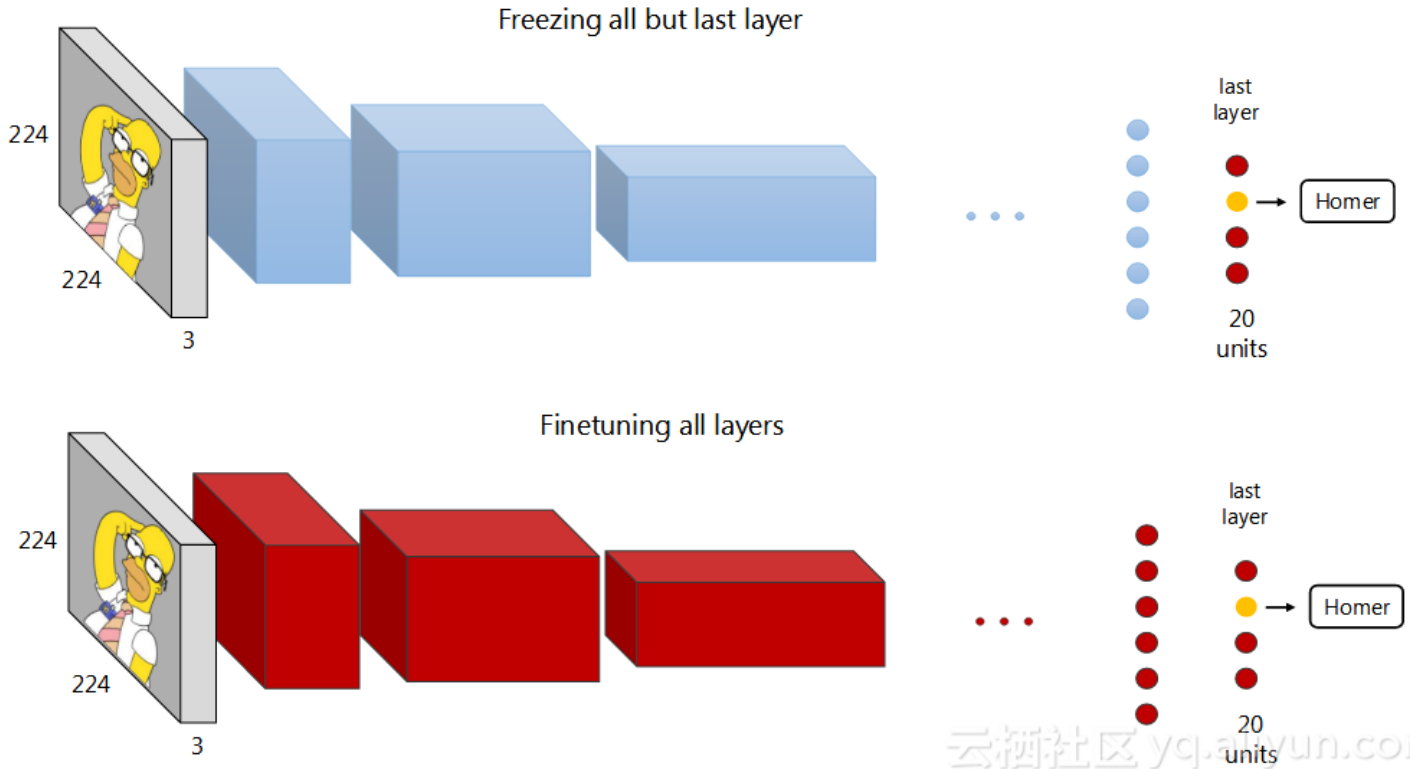
AI DISCOVERY





AI DISCOVERY

图像中的迁移学习



云栖社区 yq.aliyun.com

通常来说，迁移学习的策略有两种：

- **Fine tuning（微调）**包括在基础数据集上使用预训练网络，并在目标数据集上微调所有层。
- **Freeze and Train（冻结和训练）**包括仅冻结并训练最后一层，其他层不变（权重不更新），也可以冻结前几层，微调其他层。

最常见的基础数据集是 ImageNet，包含1000个类别的120万张图像

迁移学习

AI DISCOVERY

迁移学习概念

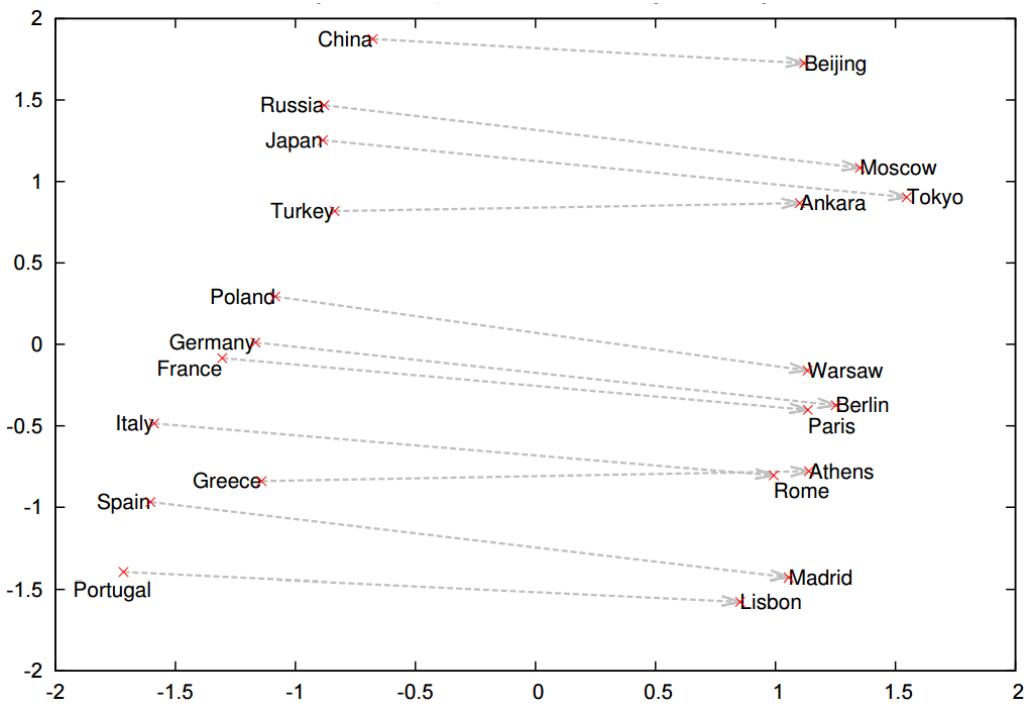
图像中的迁移

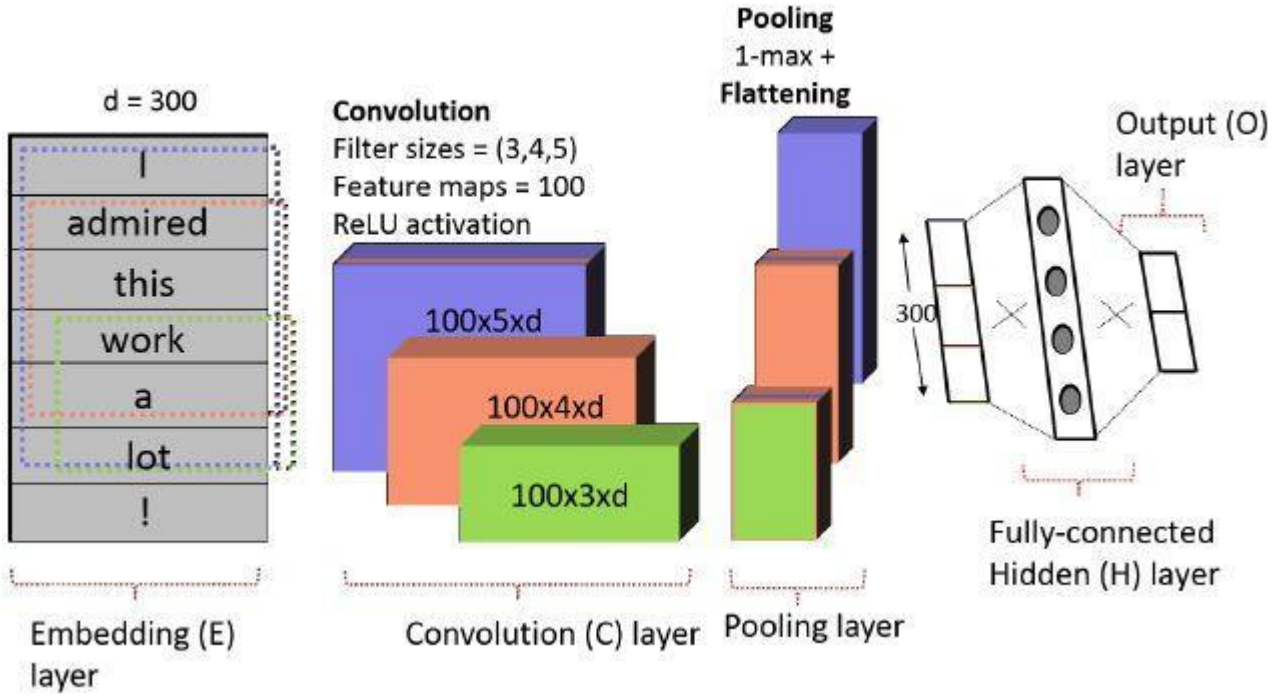
文本中的迁移

文本中的迁移学习——词级迁移

AI DISCOVERY

自然语言处理中，最初流行的迁移学习是由嵌入模型这个词带来的。对于这类问题，首先使用**词向量（Word embedding）**将单词映射到连续的高维空间，在这个高维空间中，意思相近的不同单词具有相似的向量表征。



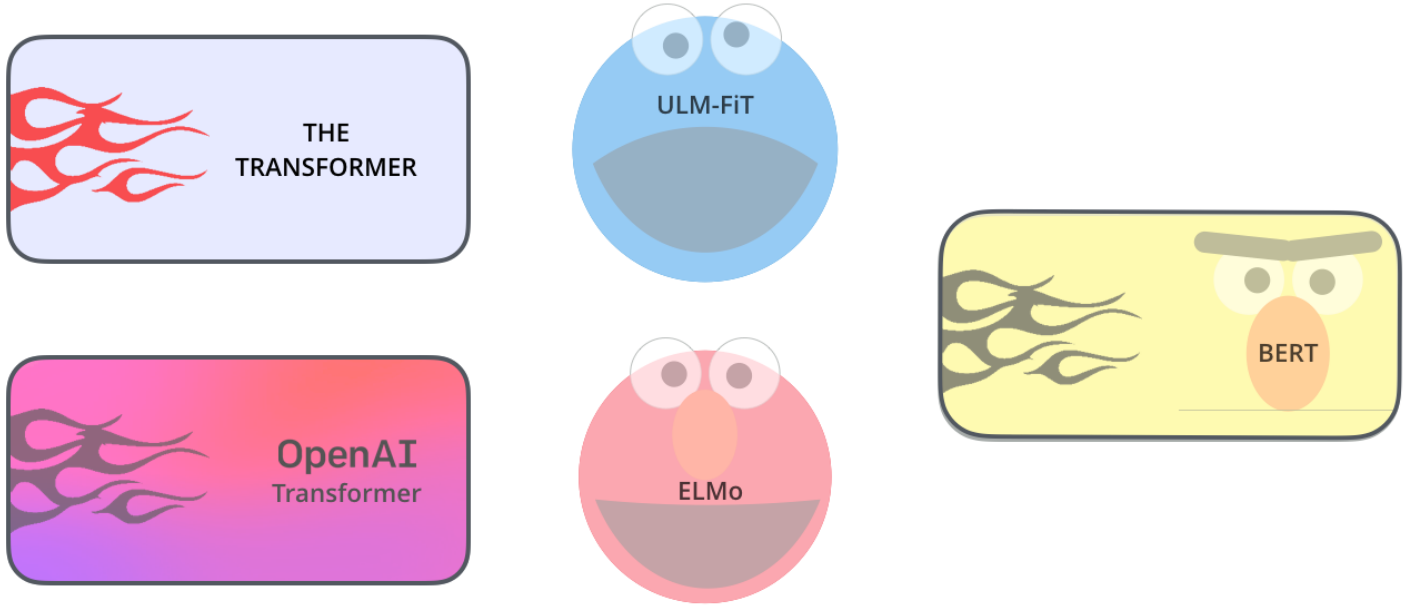


AI DISCOVERY

自然语言处理中的预训练模型

AI DISCOVERY

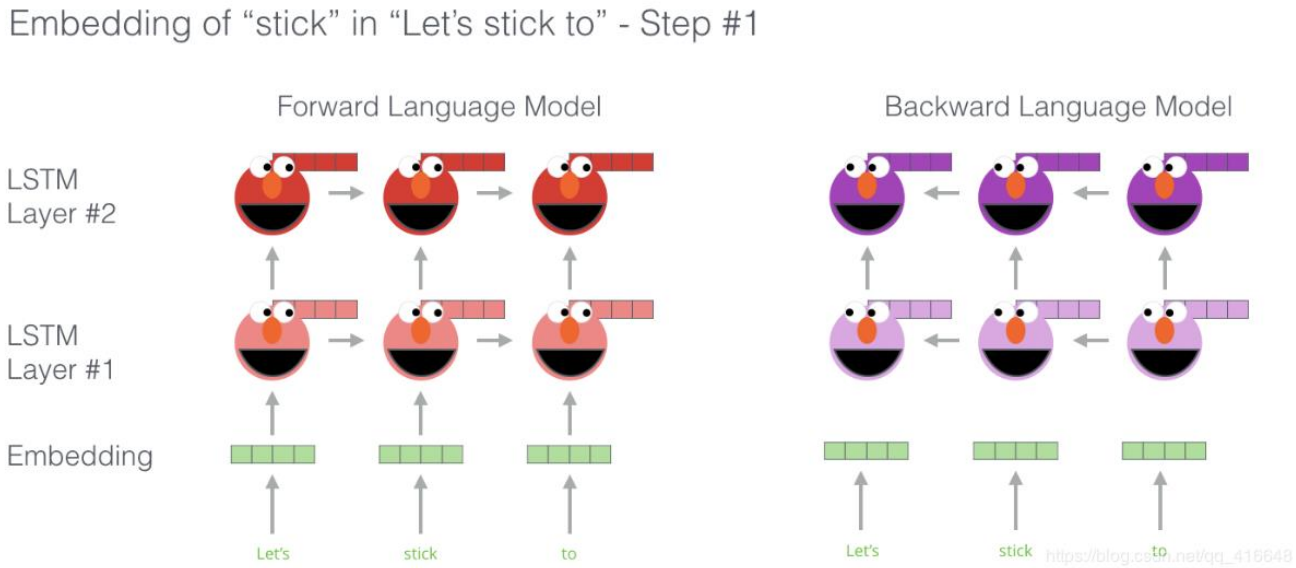
2018年是自然语言处理的转折点，**ELMo**和**BERT**等模型的提出，进一步提升了迁移学习在自然语言处理中的应用。

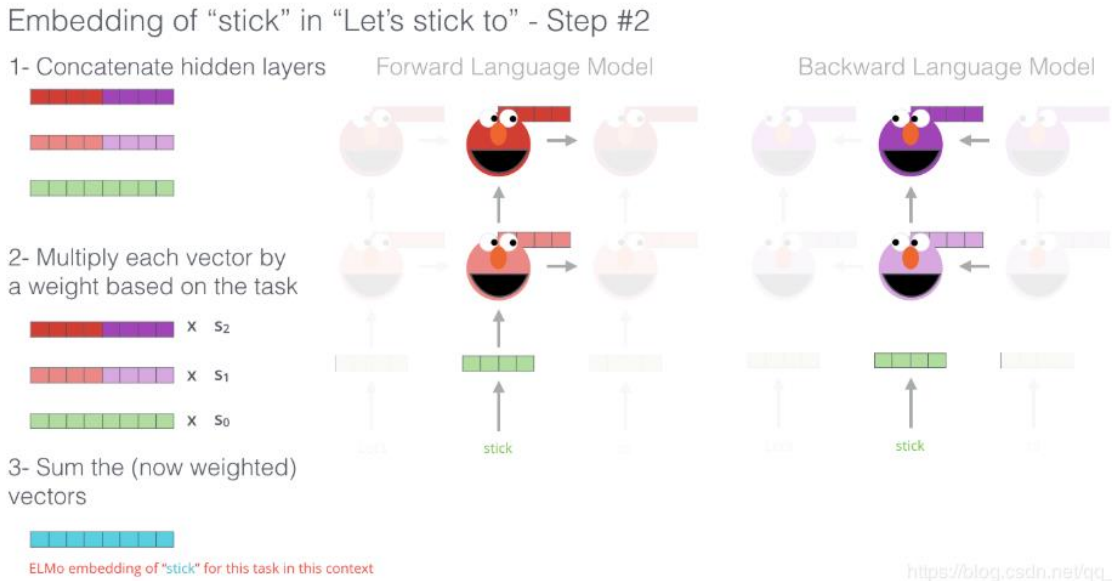


自然语言处理中的语言模型

ELMo

AI DISCOVERY





word2vec→ELMo

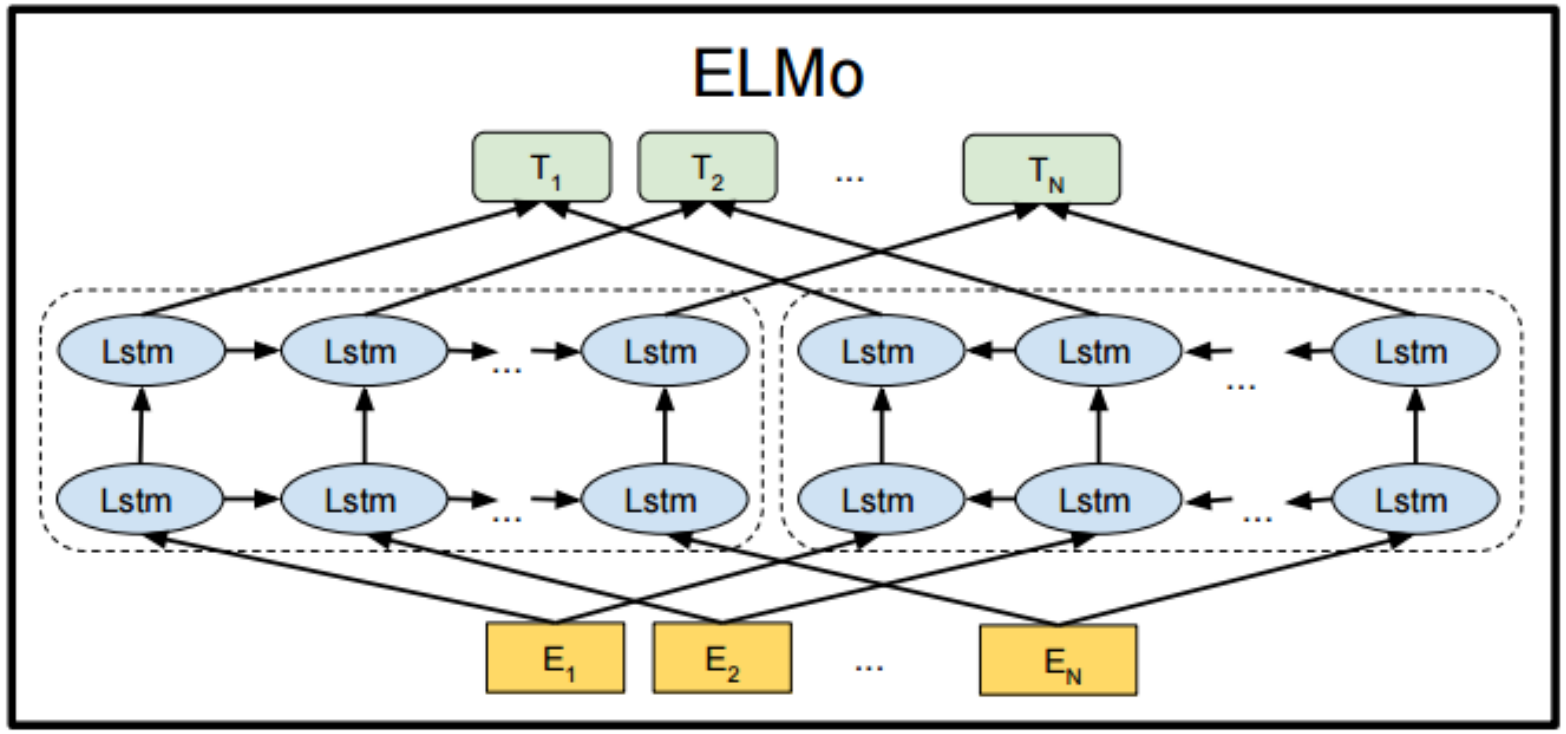
操作：为每个单词分配词向量之前先查看整个句子，然后使用bi-LSTM来训练它对应的词向量

结果：上下文无关的static向量变成上下文相关的dynamic向量，比如苹果在不同语境vector不同

EMLo为解决NLP的语境问题作出了重要的贡献，可以使用与任务相关的大量文本数据来进行训练，然后将训练好的模型用作其他NLP任务。

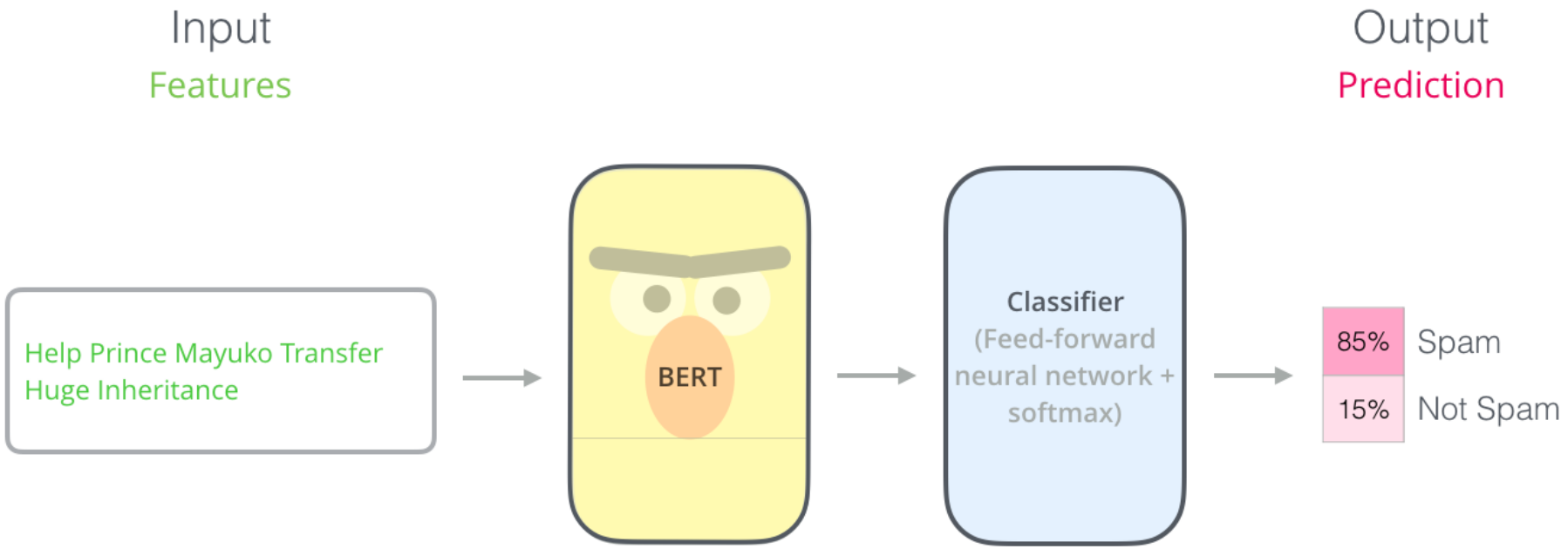
AI DISCOVERY

ELMo的模型结构图



- 在EMLo中，使用的是一个**双向的LSTM语言模型**，由一个前向和一个后向语言模型构成，目标函数就是取这两个方向语言模型的最大似然。
- 在预训练好这个语言模型之后，ELMo其实就是把这个双向语言模型的每一中间层进行求和，最简单的也可以使用最高层的表示。

文本中的迁移学习——句级迁移

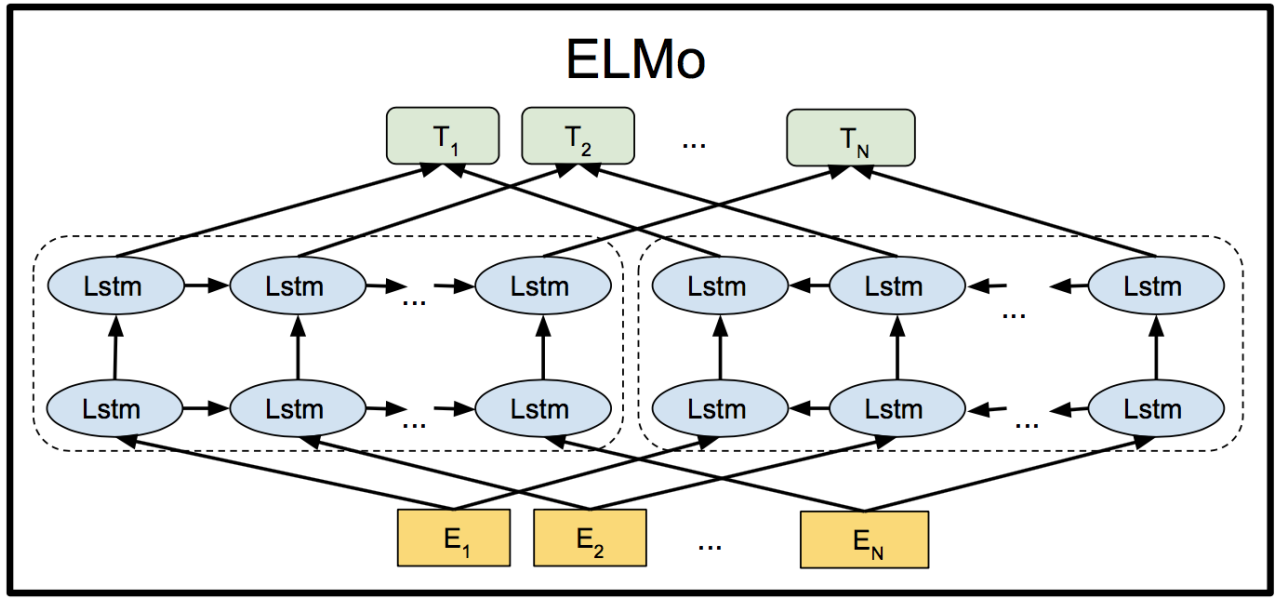


ELMo → BERT

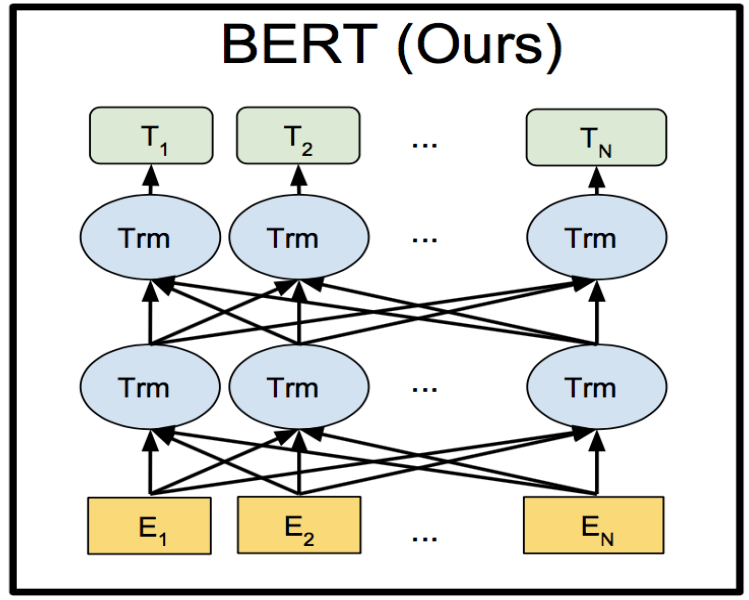
结果：训练出的word-level向量变成sentence-level的向量，具体NLP任务调用时更方便

操作：Transformer模型代替LSTM提升表达和时间上的效率，方便获得获得句子表示/句对关系

BERT的模型结构



VS.



BERT采用双向encoding

Masked LM，类似完形填空，需要预测的词被特殊符号代替

Transformer做encoder实现上下文相关（context）

使用transformer而不是bi-LSTM做encoder，可以有更深的层数、具有更好并行性

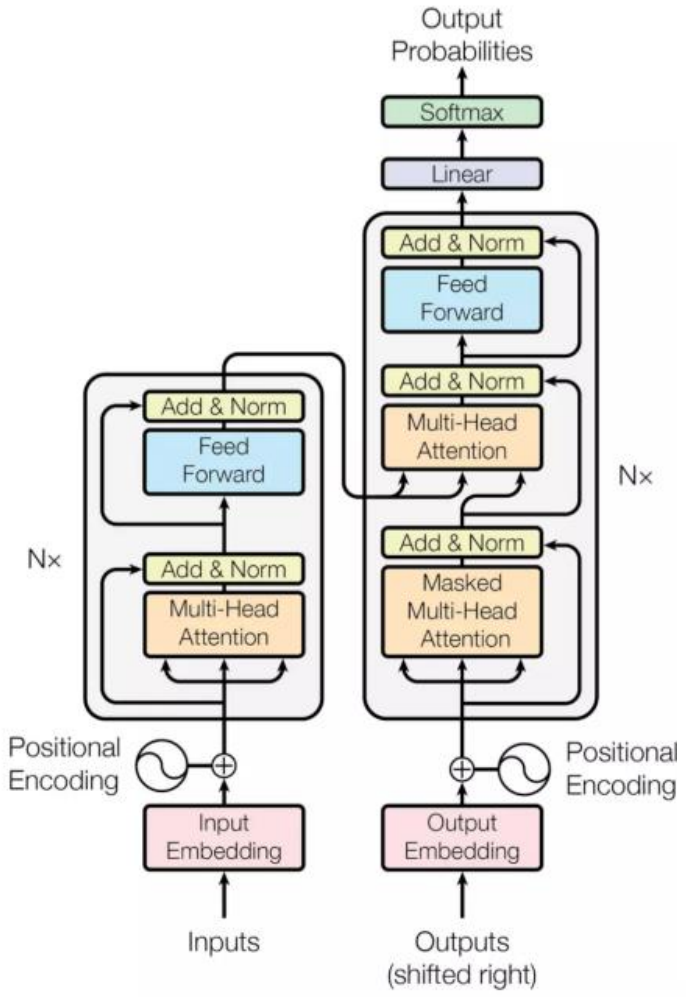
BERT的原理（1）

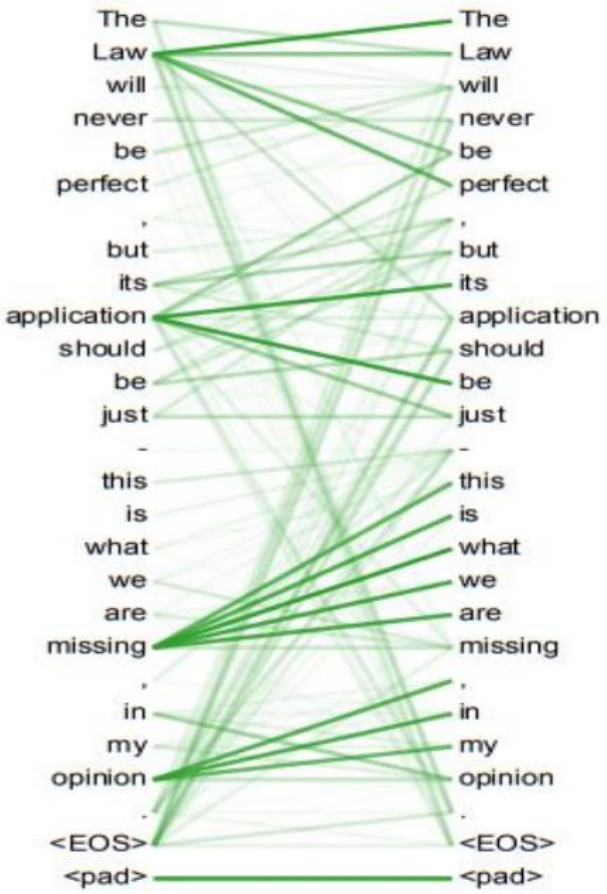
AI DISCOVERY

Transformer

- Transformer模型是2018年提出的可以替代传统RNN和CNN的一种新的架构
- 无论是RNN还是CNN，在处理NLP任务时都有缺陷
  - CNN是先天的卷积操作不很适合序列化的文本
  - RNN是没有并行化，很容易超出内存限制

针对每一个词，计算全部词与其相似度
根据相似度，将全部词进行加权求和





BERT的原理（2）

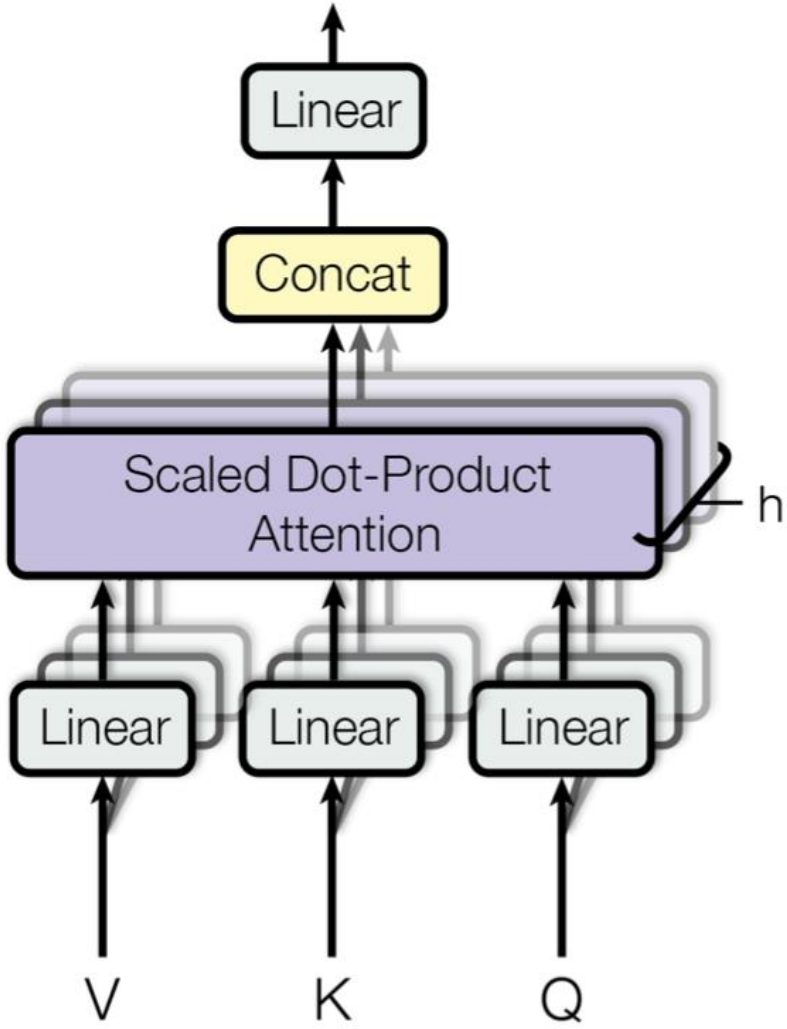
AI DISCOVERY

Multi-head Attention

- 将一个词的vector切分成h个维度，求attention相似度时每个h维度计算

- 由于词向量每一维空间都可以学到不同的特征，相邻空间所学结果更相似，相较于全体空间放到一起对应更加合理

- 比如对于vector-size=512的词向量，取h=8，每64个空间做一个attention，学到的结果更细化



BERT的原理（3）

AI DISCOVERY

position encoding

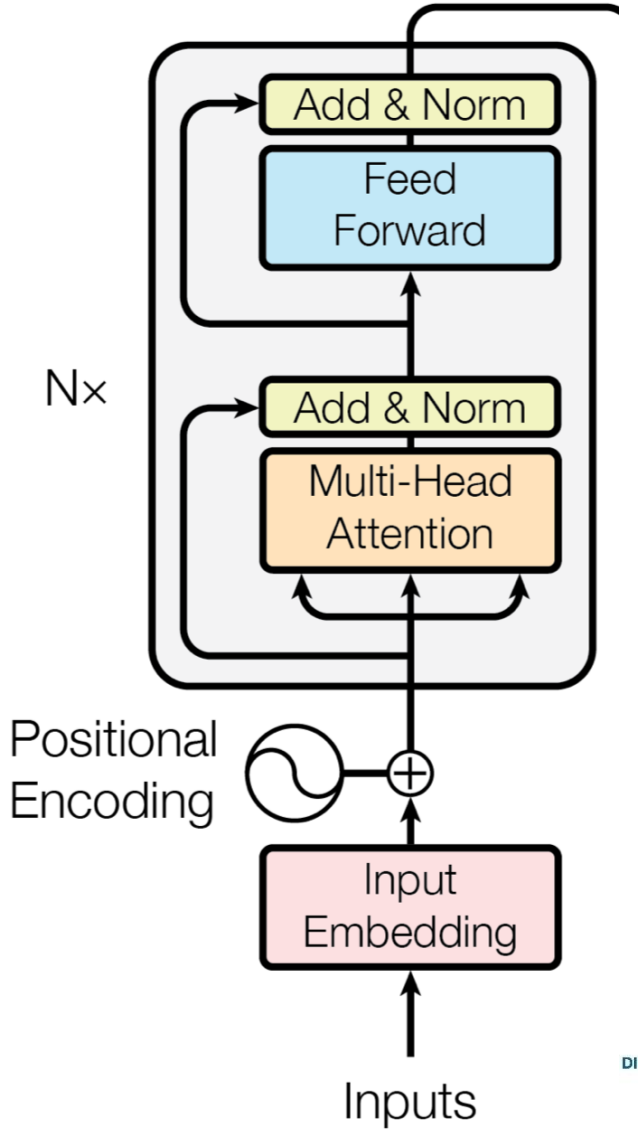
- 因为transformer既没有RNN的recurrence也没有CNN的convolution，但序列顺序信息很重要

- transformer计算token的位置信息时使用正弦波，类似模拟信号传播周期性变化

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$$

$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}})$$

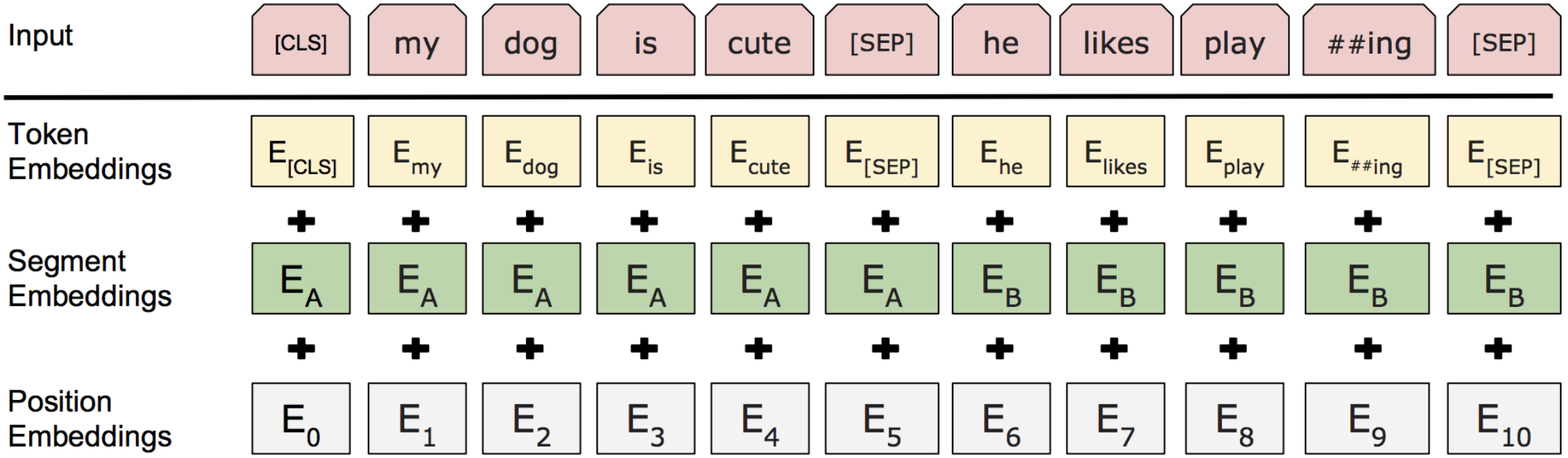
- 但BERT直接训练一个position embedding来保留位置信息，每个位置随机初始化一个向量，加入模型训练，最后就得到一个包含位置信息的embedding



DISCOVERY

模型输入

AI DISCOVERY



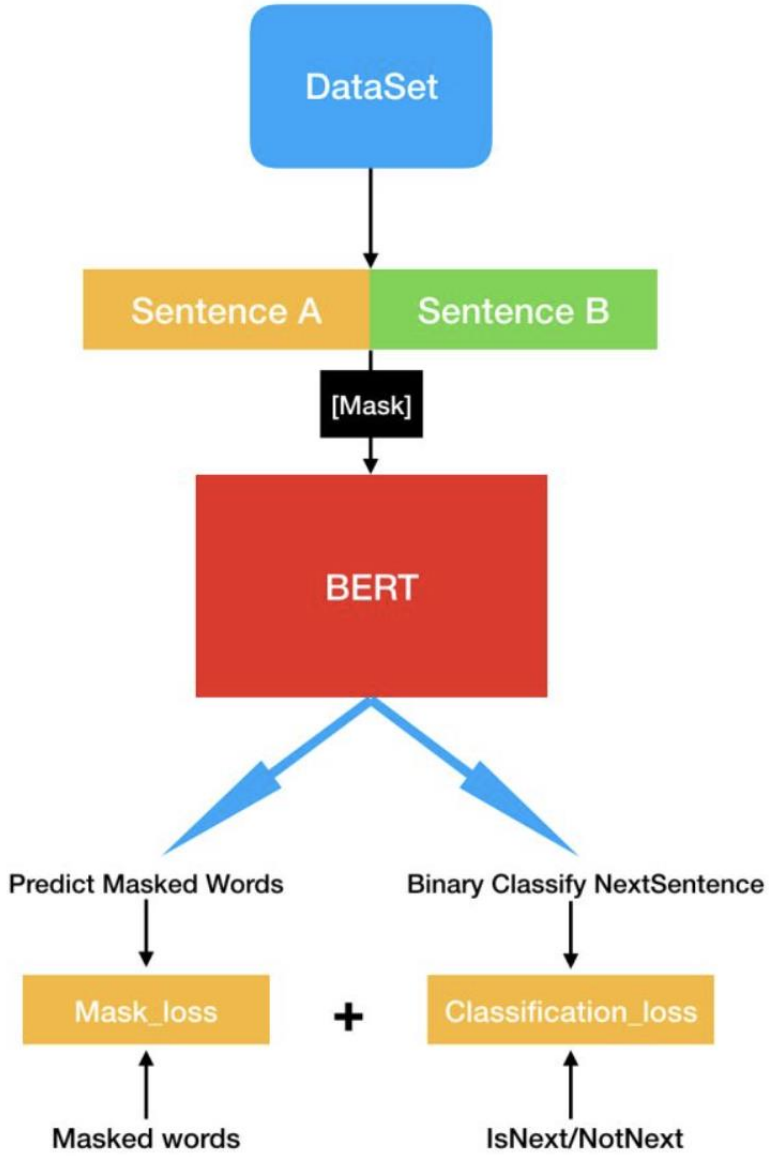
- [SEP] 是用于分割两个句子的符号
- [CLS] 作为整个句子的表征
- Position Embeddings 感知词与词之间的位置关系

AI DISCOVERY

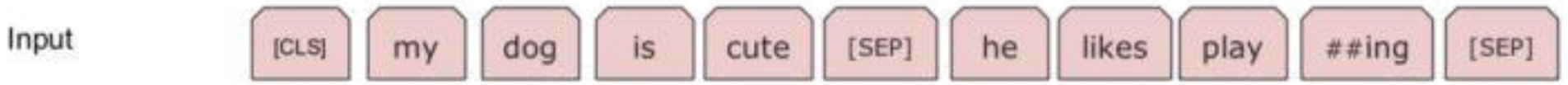
预训练——多任务学习

AI DISCOVERY

- 任务一
  - 遮盖语言模型（Masked LM）
- 任务二
  - 预测下一句话（Next Sentence Prediction）
- 目标函数
  - 两项任务的似然函数之和



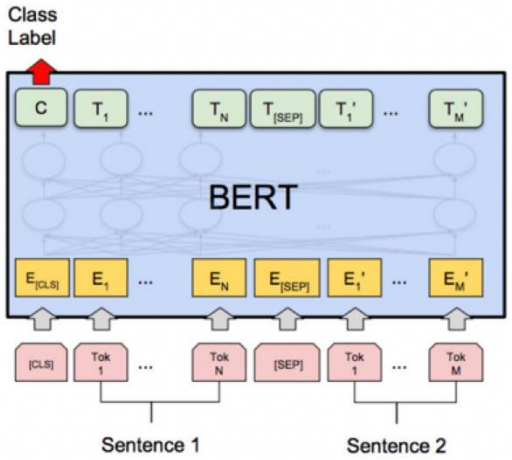
句子级表达



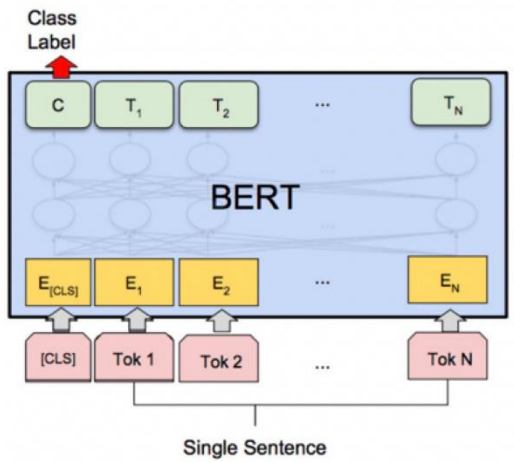
- BERT是一个句子级别的语言模型，可以直接获得一整个句子的唯一向量表示
- 它在每个input前面加一个特殊的记号[CLS]，然后让Transformer对[CLS]进行深度encoding
- 由于Transformer是可以无视空间和距离的把全局信息encoding进每个位置的，而[CLS]的最高隐层作为句子/句对的表示直接跟softmax的输出层连接，因此其作为梯度反向传播路径上的“关卡”，可以学到整个input的上层特征

BERT的应用

句子关系类任务



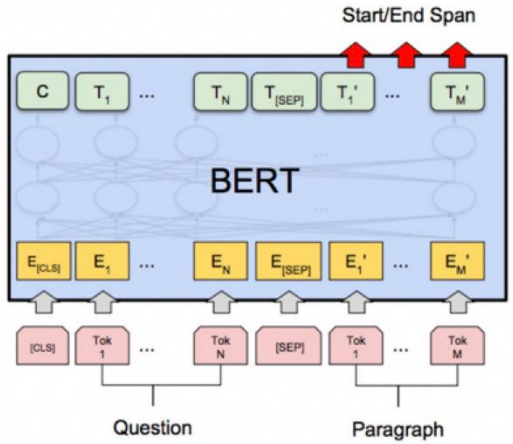
(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG



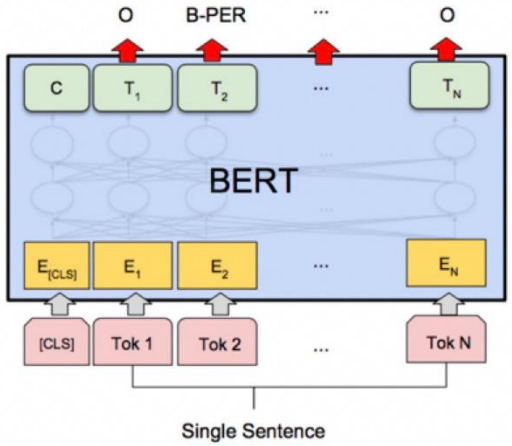
(b) Single Sentence Classification Tasks: SST-2, CoLA

句子分类任务

阅读理解类任务



(c) Question Answering Tasks: SQuAD v1.1



(d) Single Sentence Tagging Tasks: CoNLL-2003 NER

序列标注类任务

目录

AI DISCOVERY

AI DISCOVERY

生成对抗网络

AI DISCOVERY

GAN的原理

GAN的改进

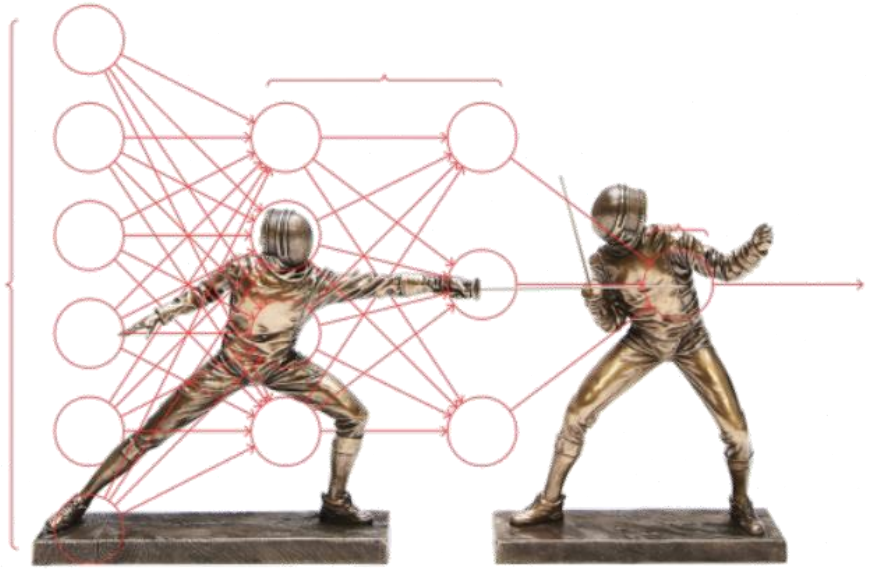
GAN的应用

AI DISCOVERY

什么是生成对抗网络（GAN）？

Generative **A**dversarial **N**etworks (**GAN**s)

- Generative 学习一个生成式模型
- Adversarial 使用对抗的方法训练
- Networks 使用神经网络



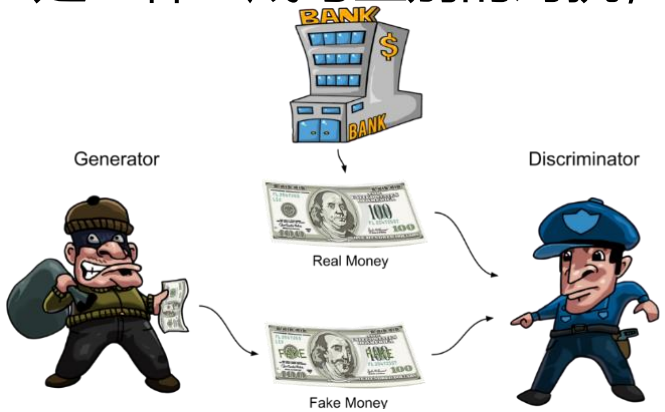
- 通过对抗的方式去学习数据分布的生成式模型
- GAN的核心思想
  - 通过生成网络G (Generator)和判别网络D (Discriminator)不断博弈，

    来达到生成类真数据的目的

对抗学习 VS 监督学习

AI DISCOVERY

GAN的思想：是一种生成与鉴别的对抗，在对抗过程中相互促进



- 监督学习：有明确的标签信息，类似于教小朋友画画
- 对抗学习：小朋友自己模仿，由大人来鉴别好坏

监督学习

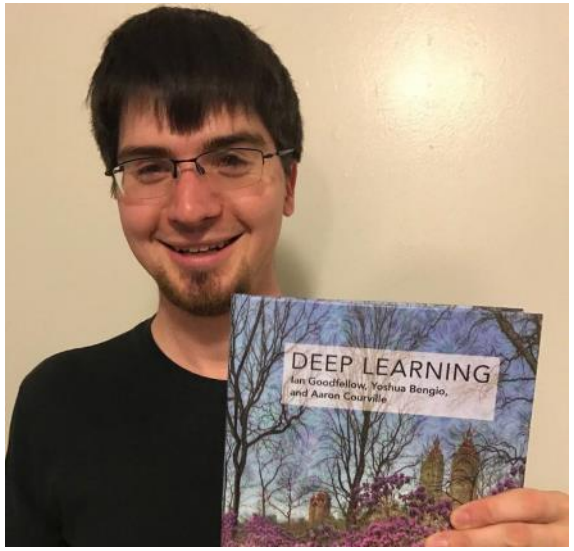
GAN

AI DISCOVERY

GAN的作者

AI DISCOVERY

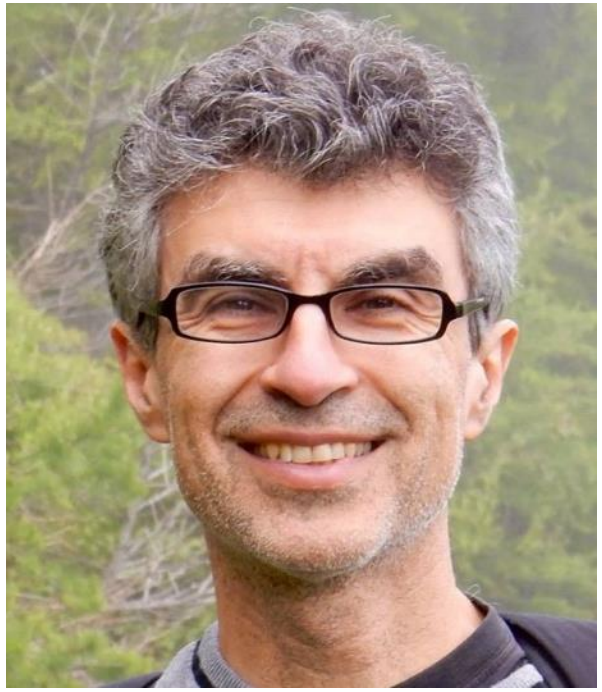
2014年，Ian Goodfellow 和蒙特利尔大学的其他研究者（包括Yoshua Bengio）提出GANs



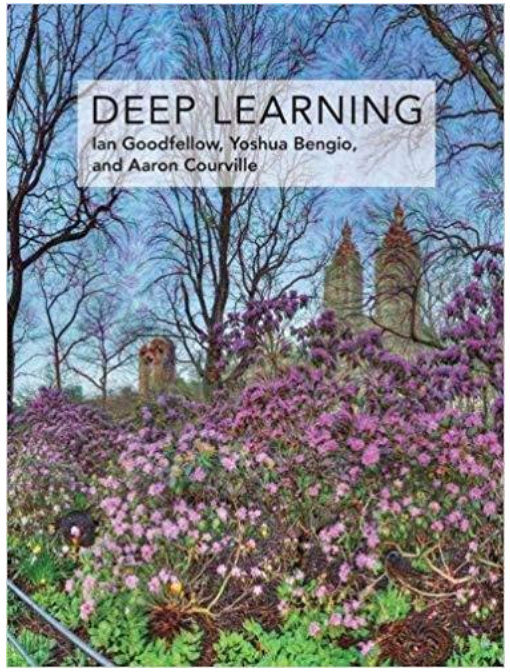
Goodfellow和他的“花书”

B.S. and M.S. Stanford University

Ph.D. Université de Montréal

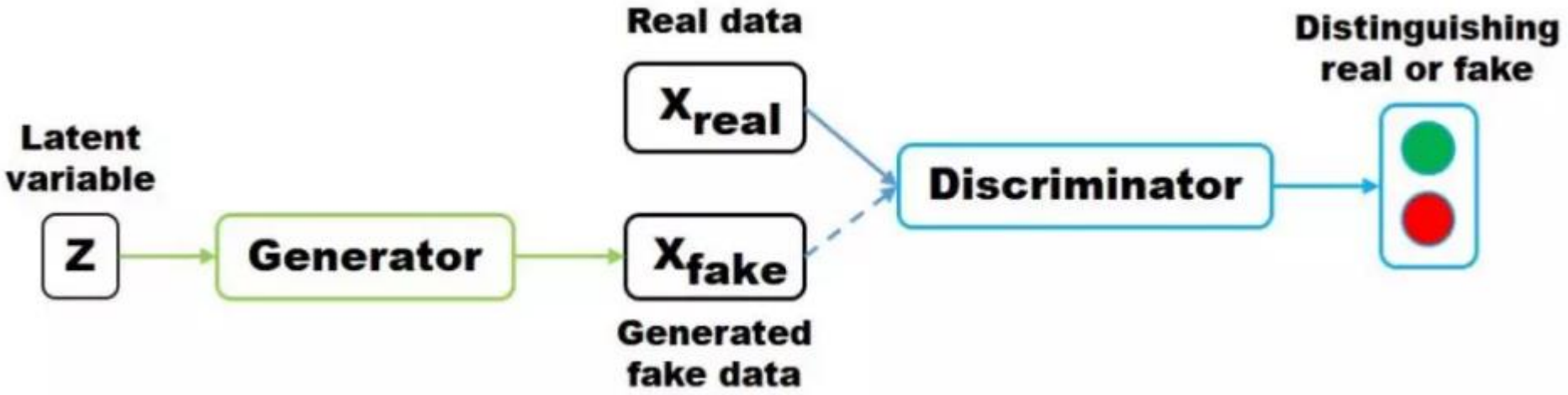


Yoshua Bengio



AI DISCOVERY

GAN的原理(1)

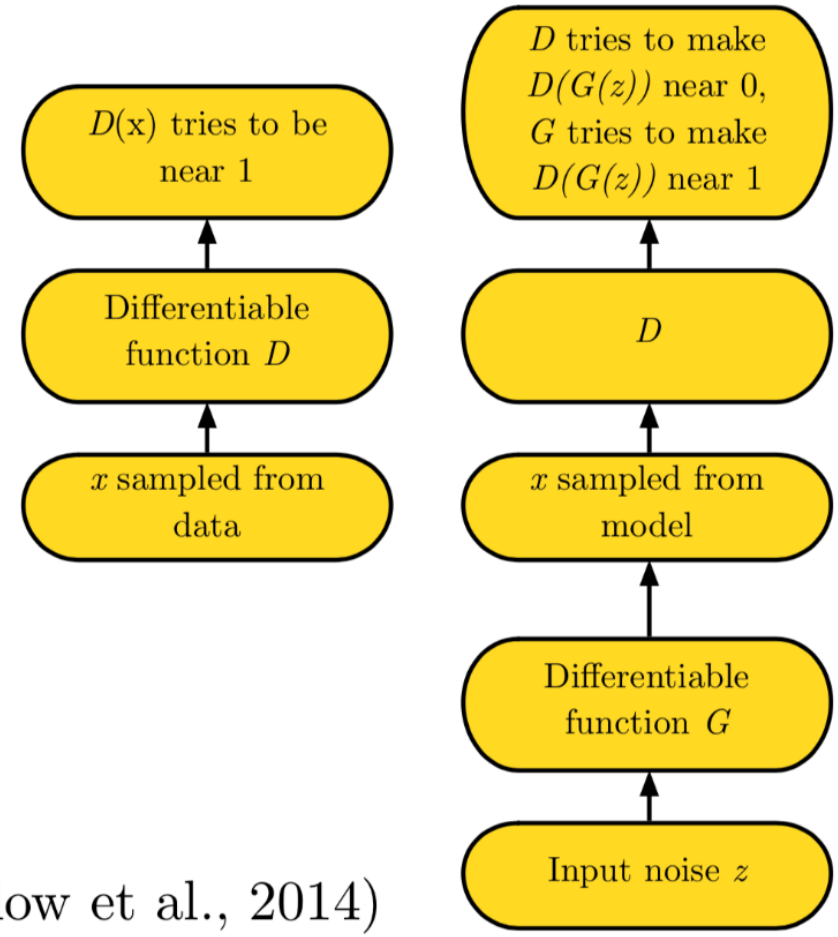


GAN 的思想启发自博弈论中的零和游戏，包含一个生成网络G和一个判别网络D

- G是一个生成式的网络，它接收一个随机的噪声Z，通过Generator生成假数据X_{fake}
- D是一个判别网络，判别输入数据的真实性。它的输入是X，输出D(X)代表X为真实数据的概率
- 训练过程中，生成网络G的目标是尽量生成真实的数据去欺骗判别网络D。而D的目标就是尽量辨别出G生成的假数据和真数据。这个博弈过程最终的平衡点是纳什均衡点

GAN的原理(2)

AI DISCOVERY



(Goodfellow et al., 2014)

假设我们有两个网络，G和D

G是一个生成图片的网络，它接收一个随机的噪声z，通过这个噪声生成图片，记做G(z)

D是一个判别网络，判别一张图片是不是“真实的”。它的输入x代表一张图片，输出D(x)代表x为真实图片的概率，如果为1，就代表100%是真实的图片，而输出为0，就代表不可能是真实的图片。

在训练过程中，生成网络G的目标就是尽量生成真实的图片去欺骗判别网络D。而D的目标就是尽量把G生成的图片和真实的图片分别开来。这样，G和D构成了一个动态的“**博弈过程**”

最后博弈的结果是什么？**在最理想的状态下，**G可以生成足以“以假乱真”的图片G(z)。对于D来说，它难以判定G生成的图片究竟是不是真实的，因此D(G(z)) = 0.5。

纳什均衡

AI DISCOVERY

这是一个博弈游戏，其**纳什均衡点**在

- $P_{data}(x) = P_{gen}(x) \ \forall x$
- $\mathrm{D}(x) = \frac{1}{2} \ \forall x$

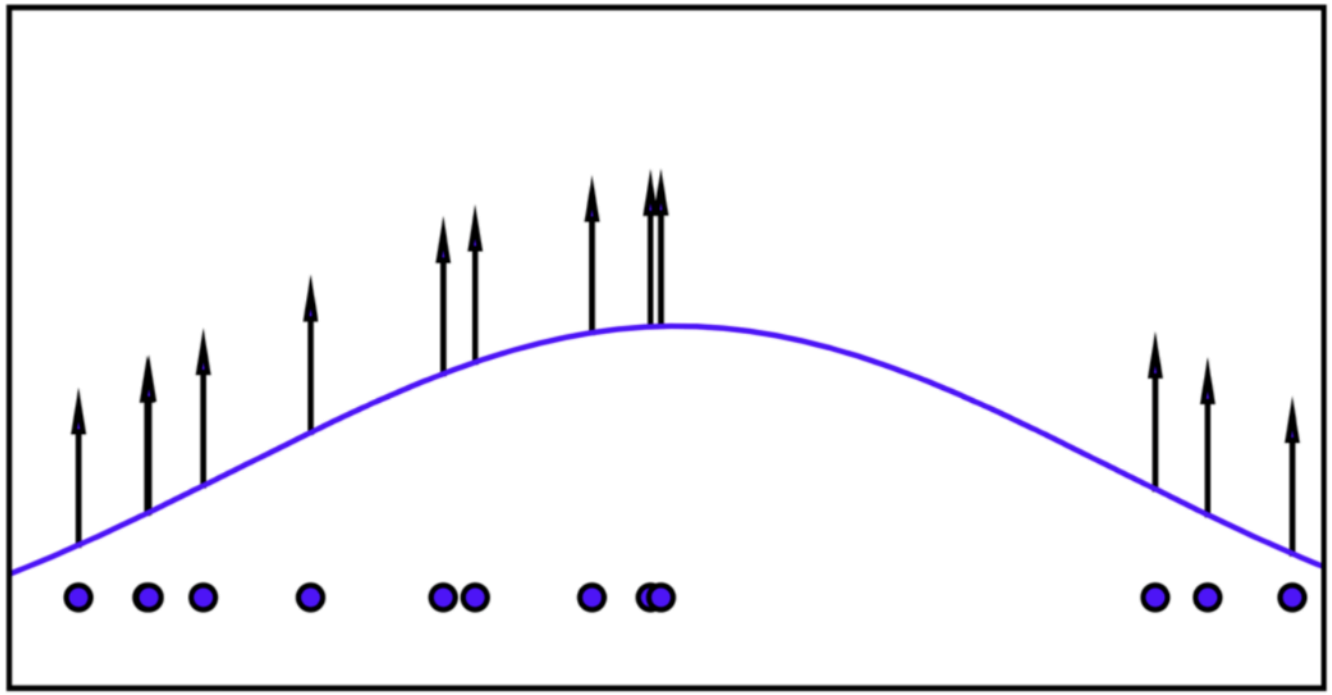
典型博弈：囚徒困境

	乙沉默（合作）	乙认罪（背叛）
甲沉默（合作）	二人同服刑半年	甲服刑10年；乙即时获释
甲认罪（背叛）	甲即时获释；乙服刑10年	二人同服刑5年

纳什均衡是指博弈中这样的局面，对于每个参与者来说，只要其他人不改变策略，他就无法改善自己的状况。

纳什均衡，又称为非合作博弈均衡，是博弈论的一个重要术语，以约翰·纳什命名。

最大似然估计



$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \mathbb{E}_{x \sim p_{\text{data}}} \log p_{\text{model}}(\boldsymbol{x} \mid \boldsymbol{\theta})$$

在统计学中，最大似然估计（英语：**M**aximum **L**ikelihood **E**stimation，缩写为MLE），也称最大概似估计，是用来估计一个概率模型的参数的一种方法

就是利用已知的样本结果信息，反推最具有可能（最大概率）导致这些样本结果出现的模型参数值

GAN的目标函数

AI DISCOVERY

GAN的核心公式

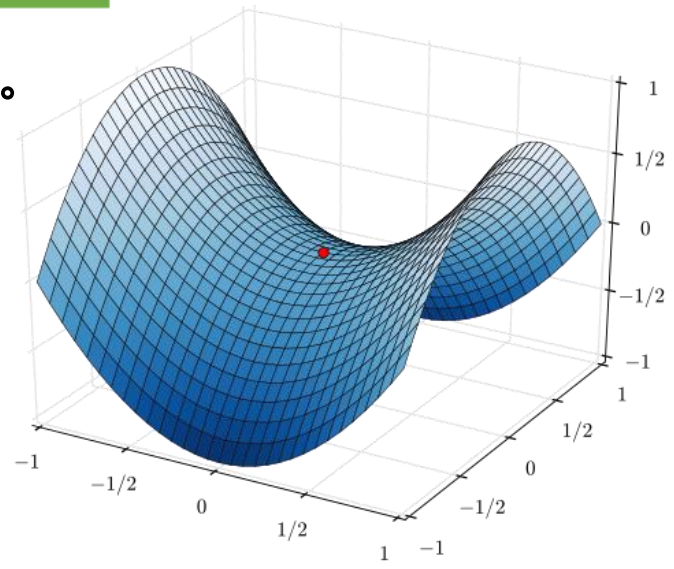
$$\min_G \max_D V(D,G)$$

从两方面来说：

- 判别器的目标是最大化它的奖励V(D, G)
- 生成器的目标是最小化判别器的奖励(最大化其损失) V(D, G)

$$V(D,G) = \mathbb{E}_{x\sim p(x)}[\log D(x)] + \mathbb{E}_{z\sim q(z)}[\log(1 - D(G(z)))]$$

- x表示真实图片，z表示输入G网络的噪声，而G(z)表示G网络生成的数据。
- D(x)表示D网络判断**真实数据是否真实的概率**，而D(G(z))是**D网络判断G生成的数据是否真实的概率**。
- **G的目的**：G希望自己生成的数据“越接近真实越好”。也就是说，G希望D(G(z))尽可能得大，这时V(D, G)会变小。因此我们看到式子的最前面的记号是min G。
- **D的目的**：D的能力越强，D(x)应该越大，D(G(x))应该越小。这时V(D,G)会变大。因此式子对于D来说是求最大(max D)



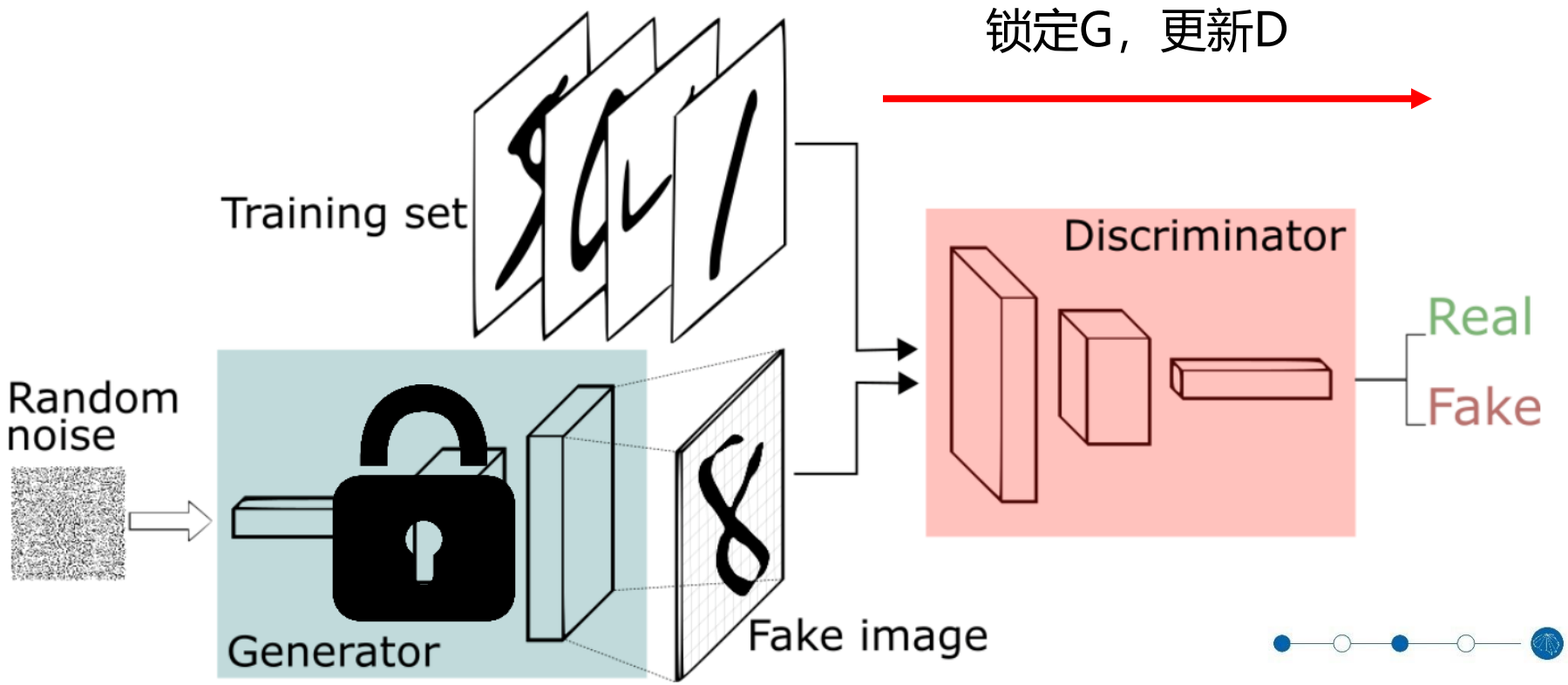
AI DISCOVERY

GAN的训练方法(1)

AI DISCOVERY

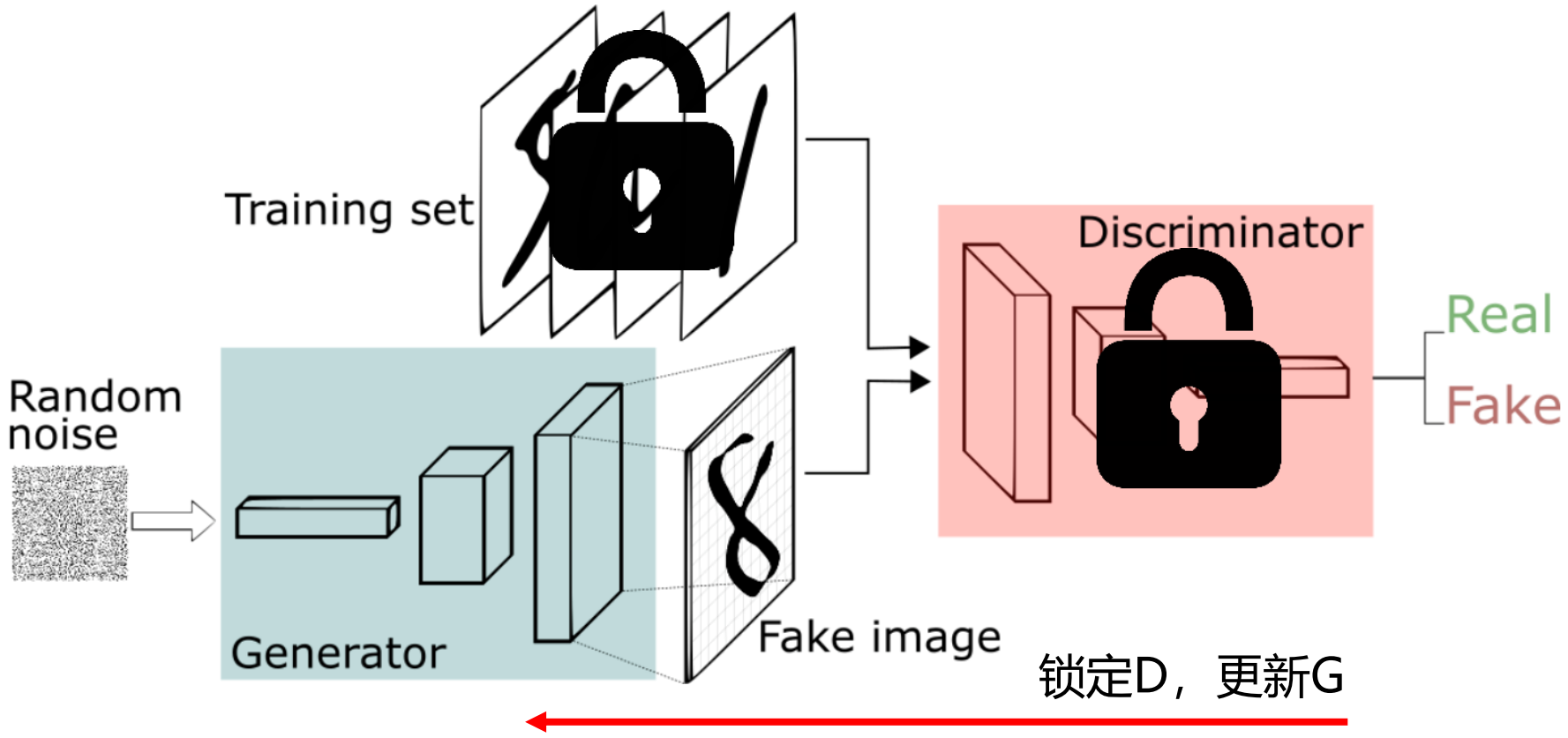
两个都需要训练，而且彼此依赖，怎么办？

- - 锁定一个，训练另一个



AI DISCOVERY

GAN的训练方法(2)


Training set
Discriminator
Real
Fake
Random noise
Generator
Fake image
锁定D，更新G

GAN的训练细节

AI DISCOVERY

Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

for number of training iterations **do**

for k steps **do**

- Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
- Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
- Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right) \right].$$

end for

Discriminator updates

- Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
- Update the generator by descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right).$$

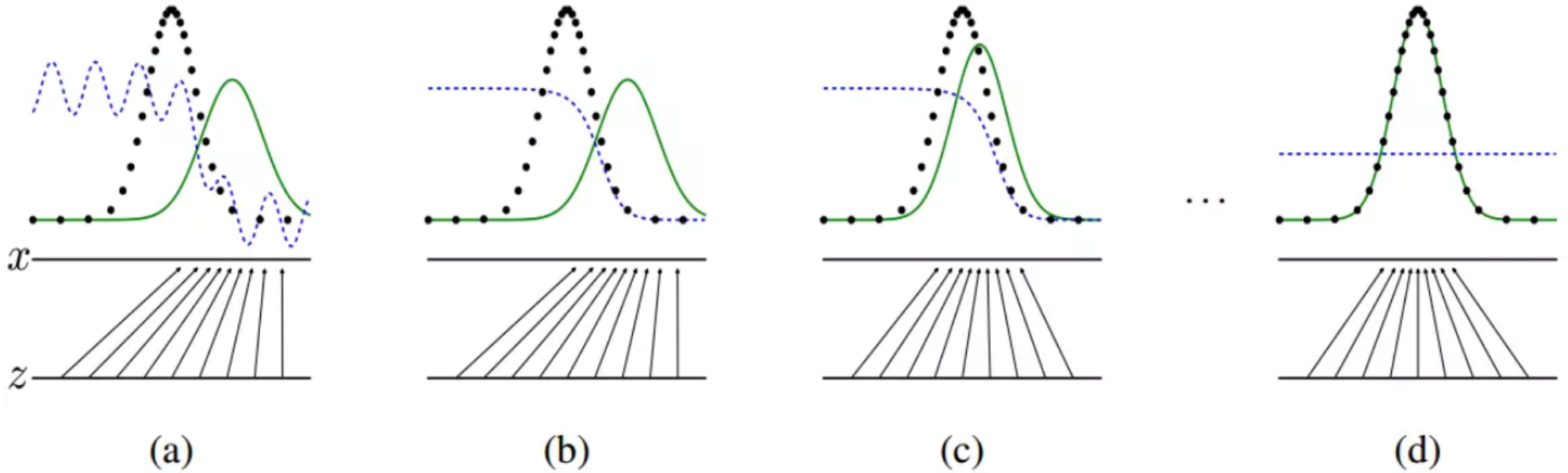
Generator updates

end for

The gradient-based updates can use any standard gradient-based learning rule. We used momentum in our experiments.

AI DISCOVERY

GAN的训练可视化



- 虚线点为真实的数据分布，蓝色虚线是判别器，绿色实线为生成器。
- 由左至右可以看到生成的分布越来越接近真实分布，而判别器的概率最后变为0.5

GAN的优点

AI DISCOVERY

摘自Ian Goodfellow在Quora的问答

- GAN是一种生成式模型，相比较其他生成模型（玻尔兹曼机和GSNs）只用到了反向传播,而不需要复杂的马尔科夫链
- 相比其他所有模型, GAN可以产生更加清晰，真实的样本
- GAN采用的是一种**无监督**的学习方式训练，可以被广泛用在无监督学习和半监督学习领域
- 相比于变分自编码器, GANs没有引入任何决定性偏置(deterministic bias),变分方法引入决定性偏置,因为它们优化对数似然的下界,而不是似然度本身,这看起来导致了VAEs生成的实例比GANs更模糊
- 相比VAE, GANs没有变分下界,如果鉴别器训练良好,那么生成器可以完美的学习到训练样本的分布.换句话说,GANs是渐进一致的,但是VAE是有偏差的
- GAN应用到一些场景上，比如图片风格迁移，超分辨率，图像补全，去噪，避免了损失函数设计的困难，不管三七二十一，只要有一个的基准，直接上判别器，剩下的就交给对抗训练了。

生成对抗网络

GAN的原理

GAN的改进

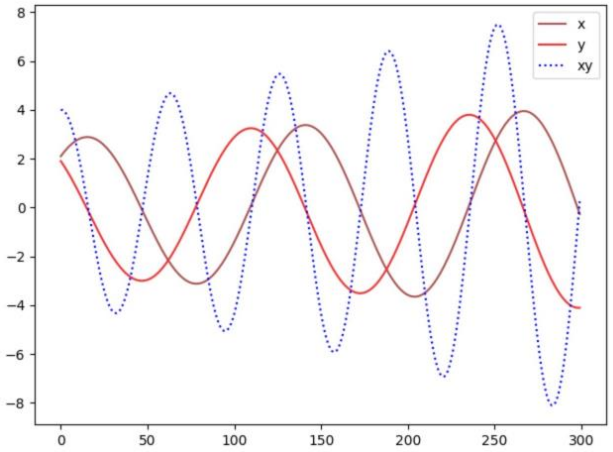
GAN的应用

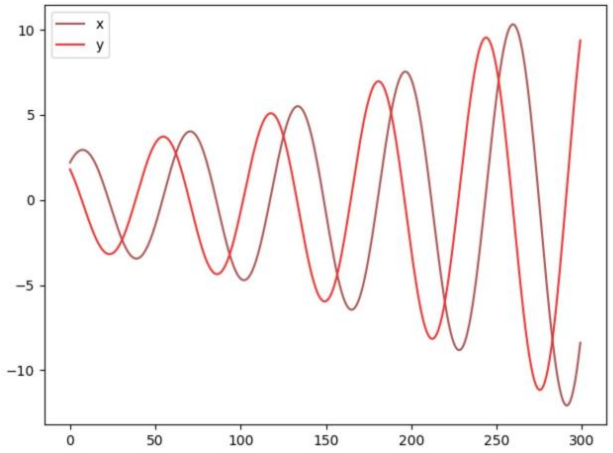
GAN存在的问题(1)

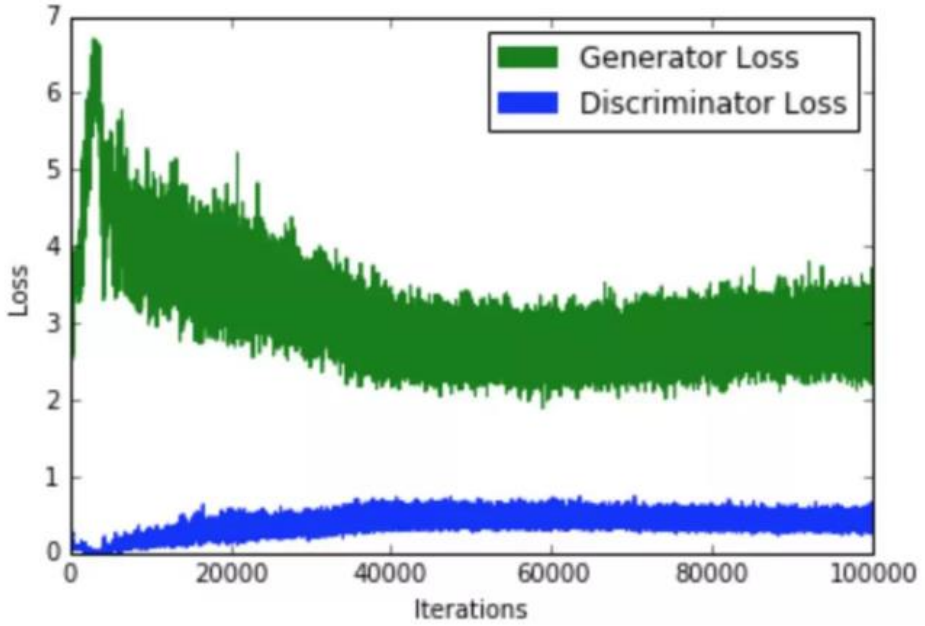
AI DISCOVERY

1. Non-Convergence （不收敛）

训练GAN需要达到纳什均衡,有时候可以用梯度下降法做到,有时候做不到.我们还没有找到很好的达到纳什均衡的方法,所以训练GAN相比VAE是不稳定的







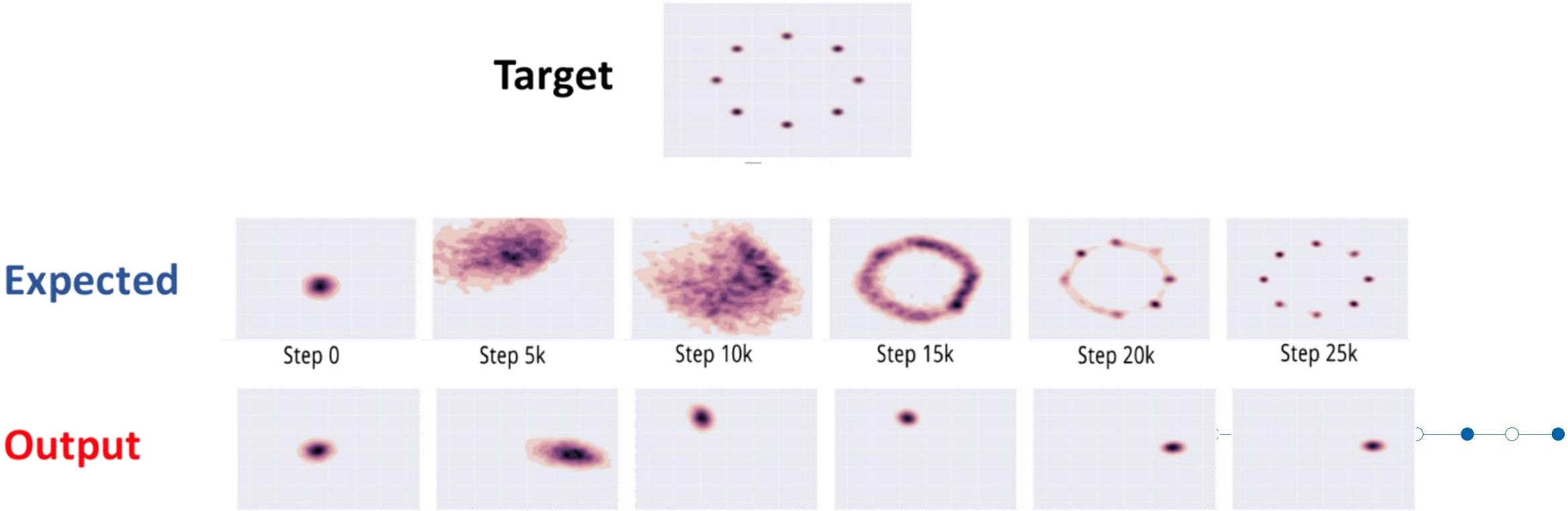
AI DISCOVERY

GAN存在的问题(2)

AI DISCOVERY

2. Mode-Collapse （模式坍塌） 可以理解为生成的内容没有多样性

一般出现在GAN训练不稳定的时候，具体表现为生成出来的结果非常差，但是即使加长训练时间后也无法得到很好的改善。



GAN存在的问题(3)

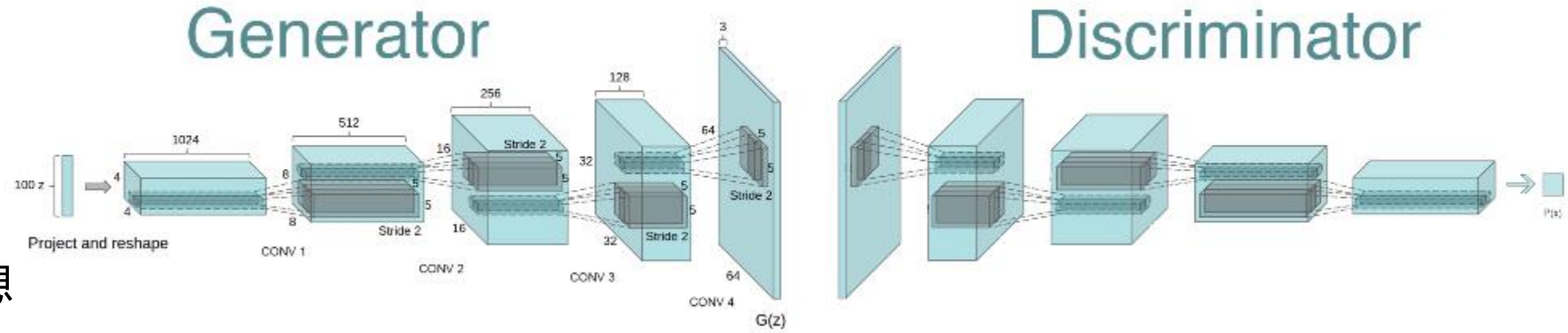
2. Mode-Collapse （模式坍塌）的原因

- GAN采用的是对抗训练的方式，G的梯度更新来自D，所以G生成的好不好，依赖于D的评价。
- 如果某一次G生成的样本可能并不是很好，但是D给出了很好的评价，或者是G生成的结果中一些特征得到了D的认可，这时候G就会认为我输出的正确的，那么接下来我就这样输出肯定D还会给出比较高的评价（实际上G生成的并不好）
- 进入一种“死循环”，最终生成结果缺失一些信息，特征不全。

GAN常见的模型结构(1)

AI DISCOVERY

Deep Convolutional GAN (DCGAN)



核心思想

1. 使用卷积层替换全连接层
2. 在每层后使用Batch Normalization。将特征层的输出归一化到一起，加速了训练，提升了训练的稳定性。（生成器的最后一层和判别器的第一层不加batchnorm）
3. G的隐藏层使用ReLU；G的输出层使用Tanh；D使用leakrelu激活函数，而不是RELU，防止梯度稀疏

上面这些 trick 对于稳定 GAN 的训练有许多帮助，自己设计 GAN 网络时也可以酌情使用

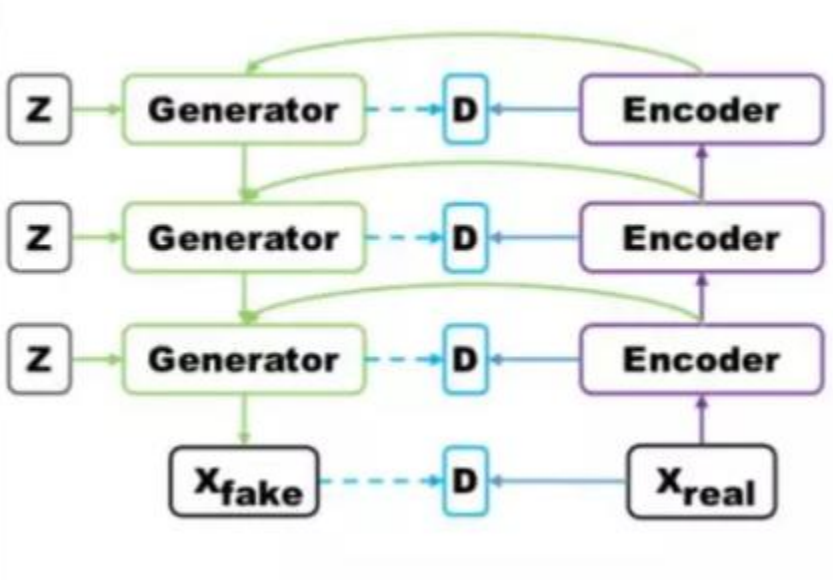
AI DISCOVERY

GAN常见的模型结构(2)

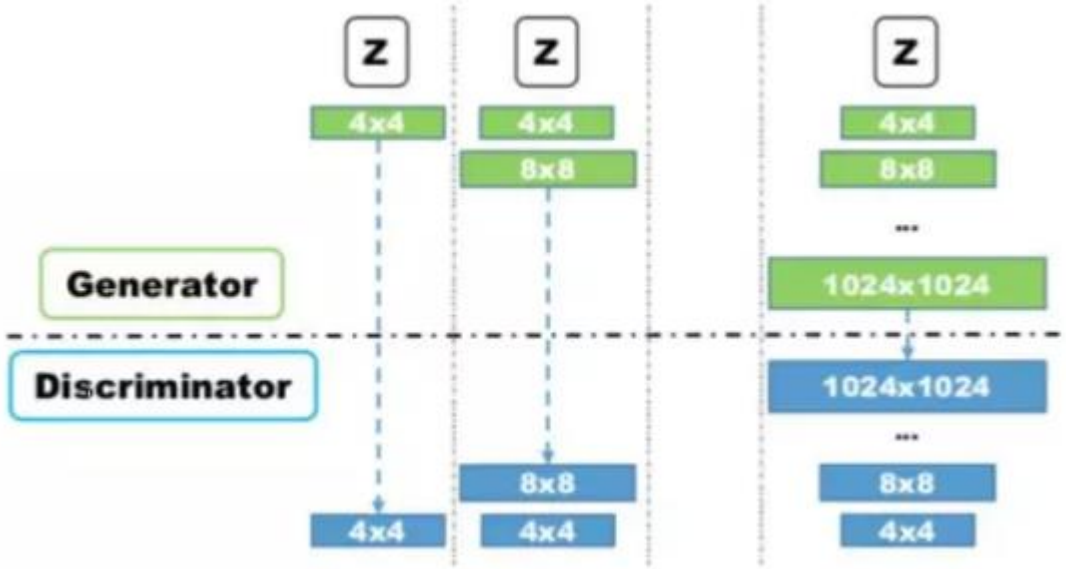
AI DISCOVERY

层级结构

GAN 对于高分辨率图像生成一直存在许多问题，层级结构的 GAN 通过逐层次，分阶段生成，一步步提生图像的分辨率。典型的使用多对 GAN 的模型有StackGAN，GoGAN。使用单一GAN，分阶段生成的有 ProgressiveGAN。



StackedGAN



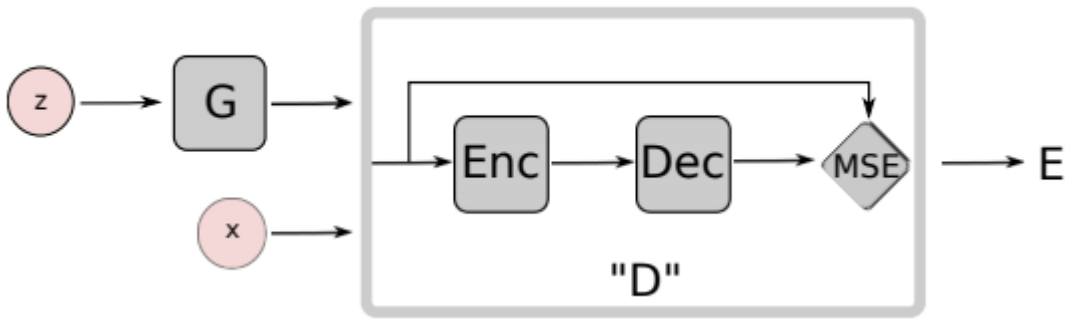
Progressive GAN

GAN常见的模型结构(3)

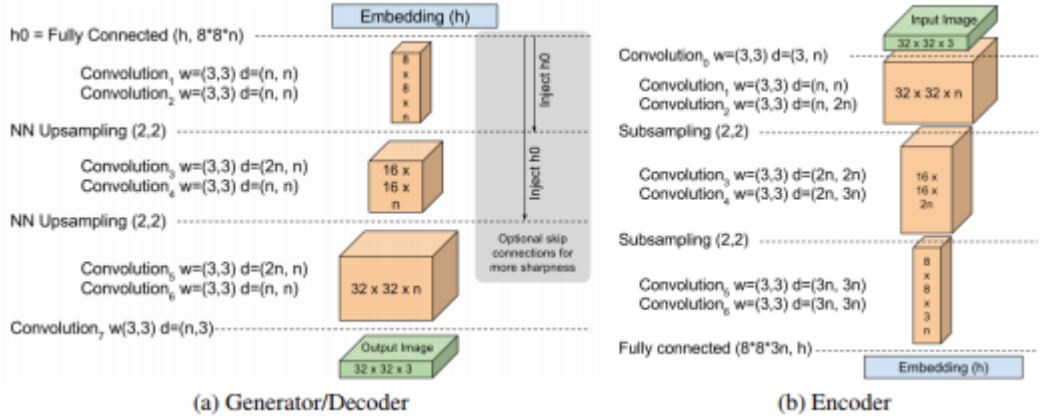
AI DISCOVERY

自编码结构

经典的 GAN 结构里面，判别网络通常被当做一种用于区分真实/生成样本的概率模型。而在自编码器结构里面，判别器（使用 AE 作为判别器）通常被当做能量函数（Energy function）。对于离数据流形空间比较近的样本，其能量较小，反之则大。有了这种距离度量方式，自然就可以使用判别器去指导生成器的学习。典型的自编码器结构的 GAN 有：BEGAN，EBGAN，MAGAN 等。



Energy-based GANs



AI DISCOVERY

Boundary Equilibrium GANs

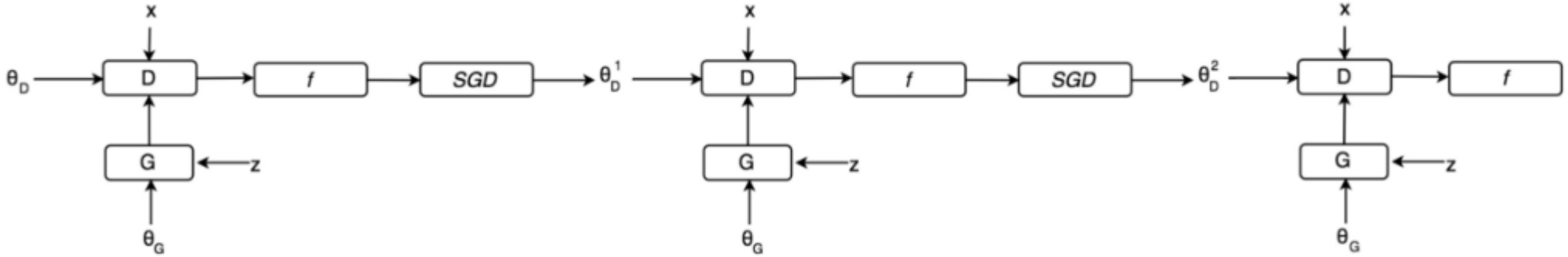
Mode Collapse的解决方案(1)

AI DISCOVERY

针对目标函数的改进方法

为了避免前面提到的由于优化 maxmin 导致 mode 跳来跳去的问题，UnrolledGAN 采用修改生成器 loss 来解决。具体而言，UnrolledGAN 在更新生成器时更新 k 次生成器，参考的 Loss 不是某一次的 loss，是判别器后面 k 次迭代的 loss。

注意，判别器后面 k 次迭代不更新自己的参数，只计算 loss 用于更新生成器。这种方式使得生成器考虑到了后面 k 次判别器的变化情况，避免在不同 mode 之间切换导致的模式崩溃问题。



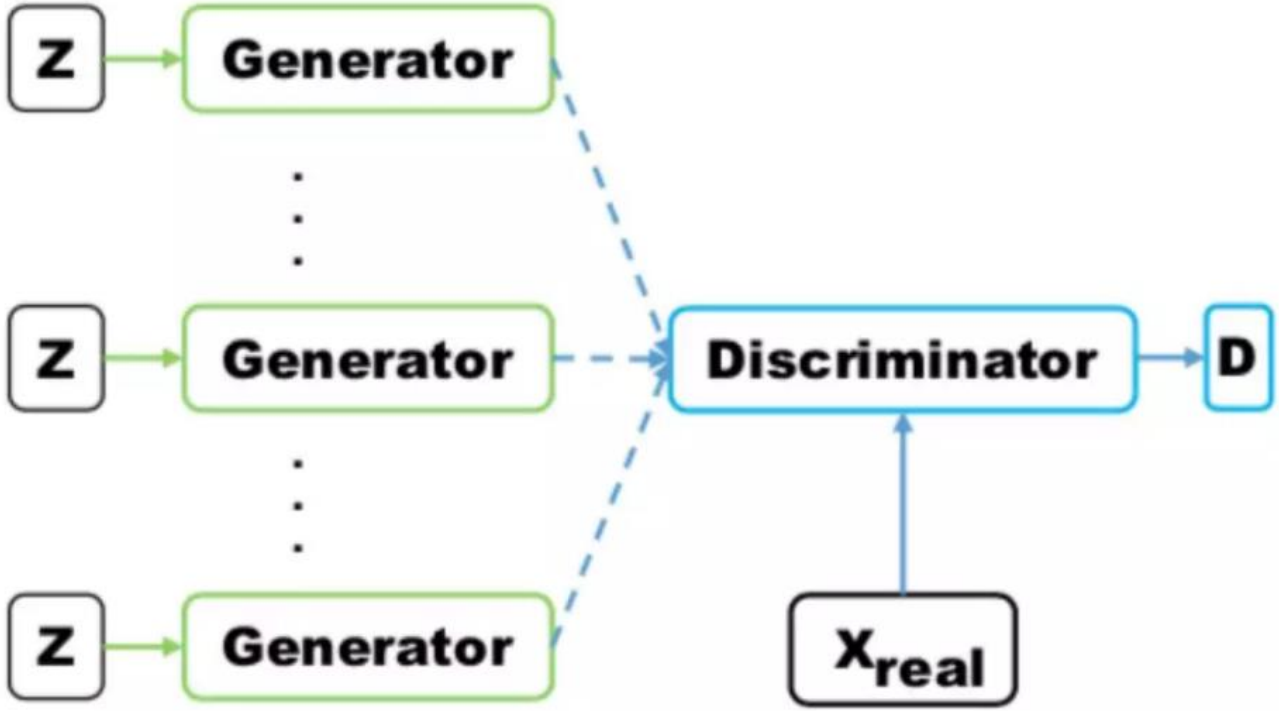
Unrolled GAN

Mode Collapse的解决方案(2)

AI DISCOVERY

针对网络结构的改进方法(1)

- Multi agent diverse GAN (MAD-GAN) 采用多个生成器，一个判别器以保障样本生成的多样性。
- 相比于普通 GAN，多了几个生成器，且在 loss 设计的时候，加入一个正则项。正则项使用余弦距离惩罚三个生成器生成样本的一致性。



MAD GAN

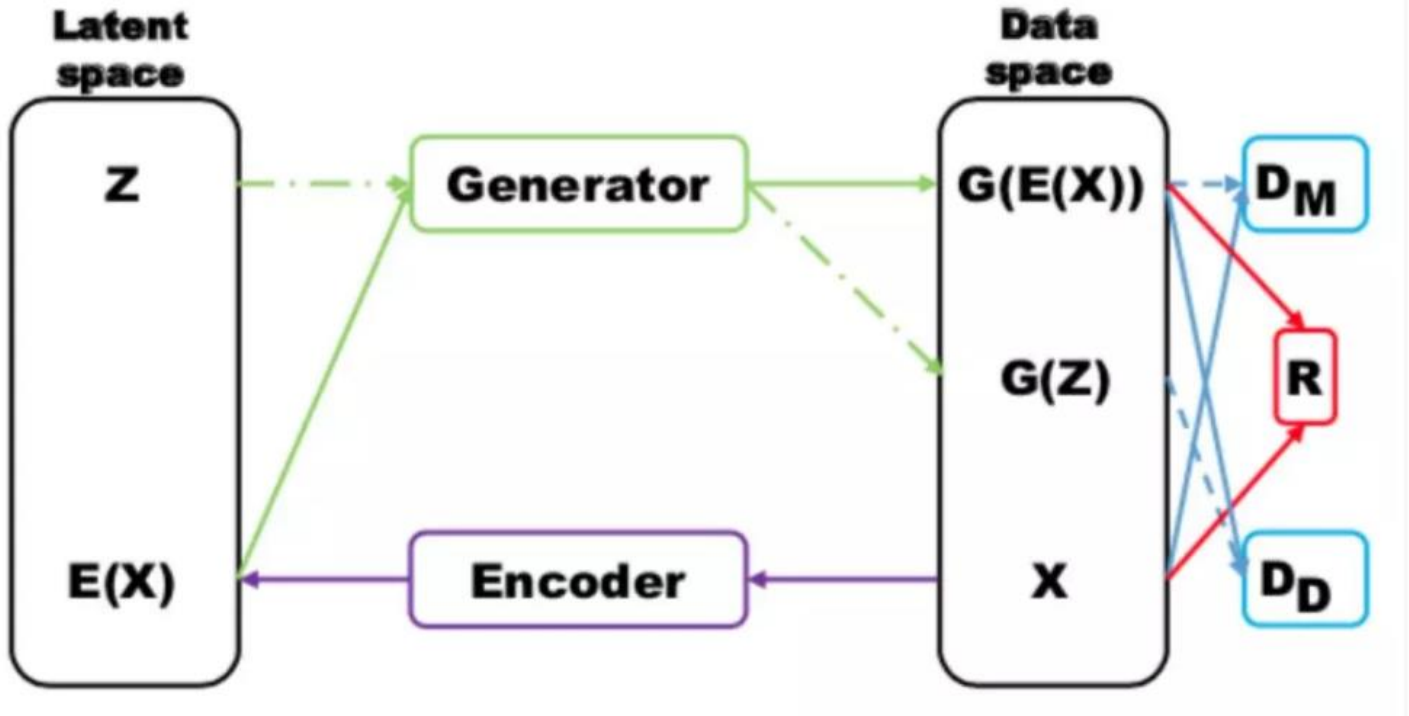
AI DISCOVERY

Mode Collapse的解决方案(3)

AI DISCOVERY

针对网络结构的改进方法(2)

- MRGAN 则添加了一个判别器来惩罚生成样本的 mode collapse 问题
- 输入样本 x 通过一个 Encoder 编码为隐变量 E(x)，然后隐变量被 Generator 重构，训练时有三个loss
- D_M 和 R （重构误差）用于指导生成 real-like 的样本。而 D_D 则对 E(x) 和 z 生成的样本进行判别



MRGAN

- 显然二者生成样本都是 fake samples，所以这个判别器主要用于判断生成的样本是否具有多样性，即是否出现 mode collapse。

AI DISCOVERY

Wasserstein GAN (WGAN)

WGAN的优点

- WGAN彻底解决GAN训练不稳定的问题，不再需要小心平衡生成器和判别器的训练程度
- 基本解决了collapse mode的问题，确保了生成样本的多样性
- 训练过程中终于有一个像交叉熵、准确率这样的数值来指示训练的进程，这个数值越小代表GAN训练得越好，代表生成器产生的图像质量越高
- 以上一切好处不需要精心设计的网络架构，最简单的多层全连接网络就可以做到

通过一系列复杂的理论分析，总结以下几点

- 判别器最后一层去掉sigmoid
- 生成器和判别器的loss不取log
- 每次更新判别器的参数之后把它们的绝对值截断到不超过一个固定常数c
- 不要用基于动量的优化算法

Wasserstein GAN (WGAN)解决的问题

AI DISCOVERY

原始GAN的问题：判别器越好，生成器梯度消失越严重。

- 根据原始GAN定义的判别器loss，可以得到最优判别器的形式；而在最优判别器的下，可以把原始GAN定义的生成器loss等价变换为最小化真实分布与生成分布之间的JS散度。
- 我们越训练判别器，它就越接近最优，最小化生成器的loss也就会越近似于最小化真实分布和生成分布之间的JS散度。

问题就出在这个JS散度上

- 如果两个分布之间越接近，它们的JS散度越小，通过优化JS散度就能将它们拉近，最终以假乱真。

Vanishing gradient strikes back again...

$$\min_G \max_D V(D,G)$$

$$V(D,G) = \mathbb{E}_{x\sim p(x)}[\log D(x)] + \mathbb{E}_{z\sim q(z)}\left[\log\left(1 - D(G(z))\right)\right]$$

$$\nabla_{\theta_G} V(D,G) = \nabla_{\theta_G} \mathbb{E}_{z\sim q(z)}\left[\log\left(1 - D\left(G(z)\right)\right)\right]$$

- $\nabla_a \log\left(1-\sigma(a)\right) = \frac{-\nabla_a \sigma(a)}{1-\sigma(a)} = \frac{-\sigma(a)\left(1-\sigma(a)\right)}{1-\sigma(a)} = -\sigma(a) = -D\left(G(z)\right)$
- Gradient goes to 0 if D is confident, i.e. $D\left(G(z)\right) \to 0$
- Minimize $-\mathbb{E}_{z\sim q(z)}\left[\log D\left(G(z)\right)\right]$ for **Generator** instead (keep Discriminator as it is)

- 以上在两个分布有所重叠的时候是成立的，但是如果两个分布完全没有重叠的部分，或者它们重叠的部分可忽略，它们的JS散度是log2
- 在（近似）最优判别器下，最小化生成器的loss等价于最小化真实分布与生成之间的JS散度，而由于与几乎不可能有不可忽略的重叠，所以无论它们相距多远JS散度都是常数，最终导致生成器的梯度（近似）为0，梯度消失。

AI DISCOVERY

Wasserstein距离(1)

AI DISCOVERY

GAN: minimize Jensen-Shannon divergence between p_X and $p_{G(Z)}$

$$JS(p_X||p_{G(Z)}) = KL(p_X||\frac{p_X + p_{G(Z)}}{2}) + KL(p_{G(Z)}||\frac{p_X + p_{G(Z)}}{2})$$

WGAN: minimize earth mover distance between p_X and $p_{G(Z)}$

$$EM(p_X, p_{G(Z)}) = \inf_{\gamma \in \prod(p_X, p_{G(Z)})} E_{(x,y)\sim\gamma}[||x - y||]$$

Wasserstein距离又叫Earth-Mover（EM）距离，直观上可以理解为在γ这个“路径规划”下把这堆“沙土”P_r挪到“位置”P_g所需的“消耗”，而$W(P_r,P_g)$就是“最优路径规划”下的“最小消耗”，所以才叫Earth-Mover（推土机）距离。

AI DISCOVERY

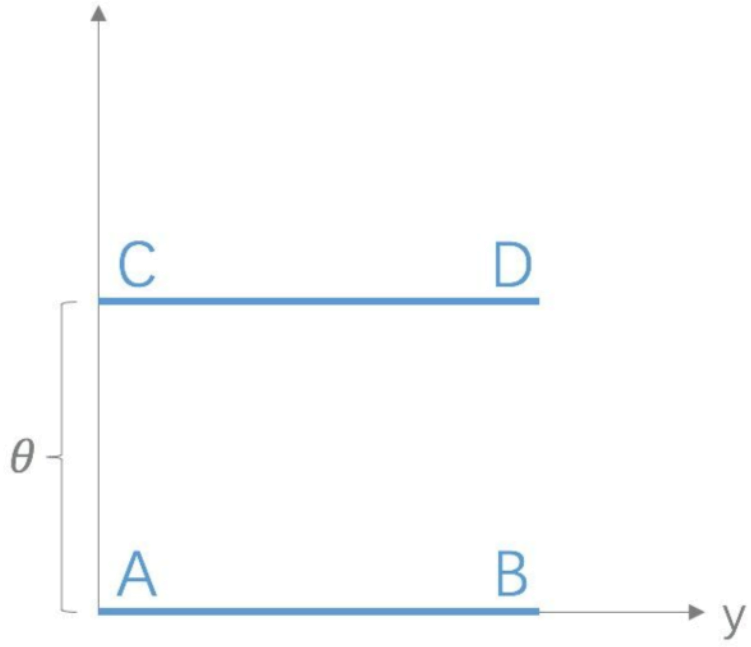
Wasserstein距离(2)

AI DISCOVERY

Wasserstein距离相比KL散度、JS散度的优越性在于，即便两个分布没有重叠，Wasserstein距离仍然能够反映它们的远近。

考虑分布P1和P2，P1在线段AB上均匀分布，P2在线段CD上均匀分布，

通过参数θ可以控制两个分布的远近



$$KL(P_1||P_2) = KL(P_1||P_2) = \begin{cases} +\infty & \text{if } \theta \neq 0 \\ 0 & \text{if } \theta = 0 \end{cases} \quad \text{(突变)}$$

$$JS(P_1||P_2) = \begin{cases} \log 2 & \text{if } \theta \neq 0 \\ 0 & \text{if } \theta - 0 \end{cases} \quad \text{(突变)}$$

$$W(P_0, P_1) = |\theta| \quad \text{(平滑)}$$

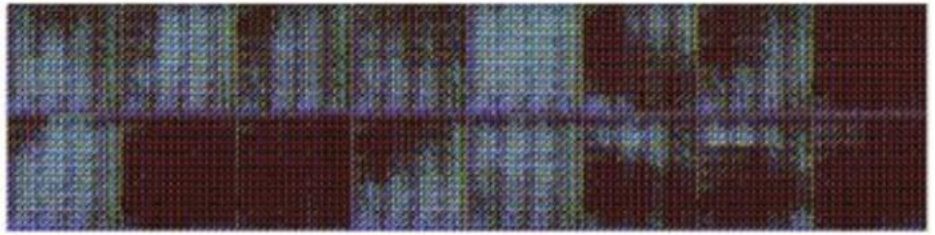
更多参考 https://zhuanlan.zhihu.com/p/25071913

AI DISCOVERY

原始GAN VS. WGAN

WGAN如果用类似DCGAN架构，生成图片的效果与DCGAN差不多

拿掉Batch Normalization的话，DCGAN不能生成正常图片



如果WGAN和原始GAN都使用多层全连接网络MLP，不用CNN，WGAN质量会变差些，但是原始GAN不仅质量变得更差，而且还出现了collapse mode，即多样性不足

生成对抗网络

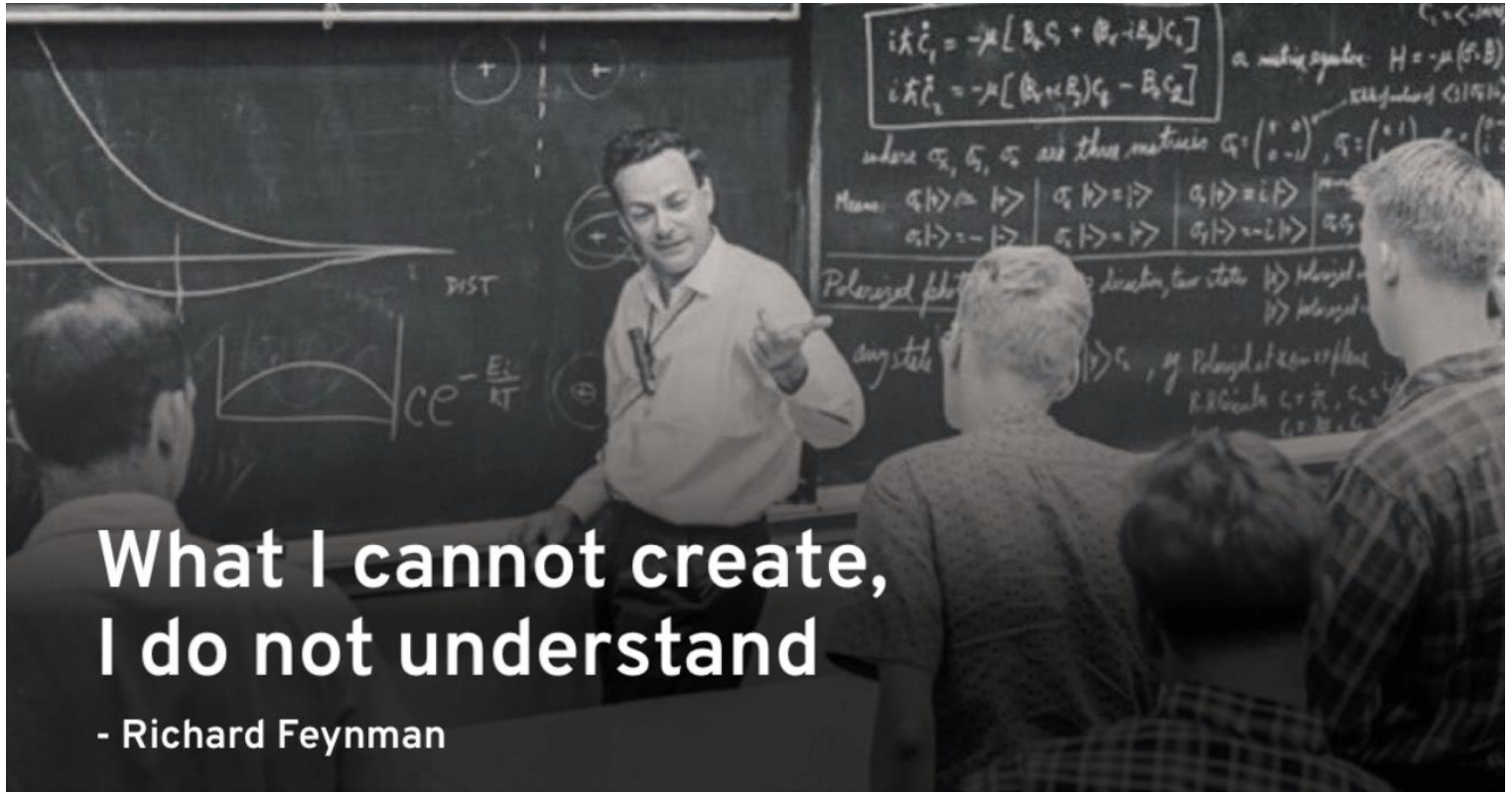
GAN的原理

GAN的改进

GAN的应用

GAN的应用场景

AI DISCOVERY

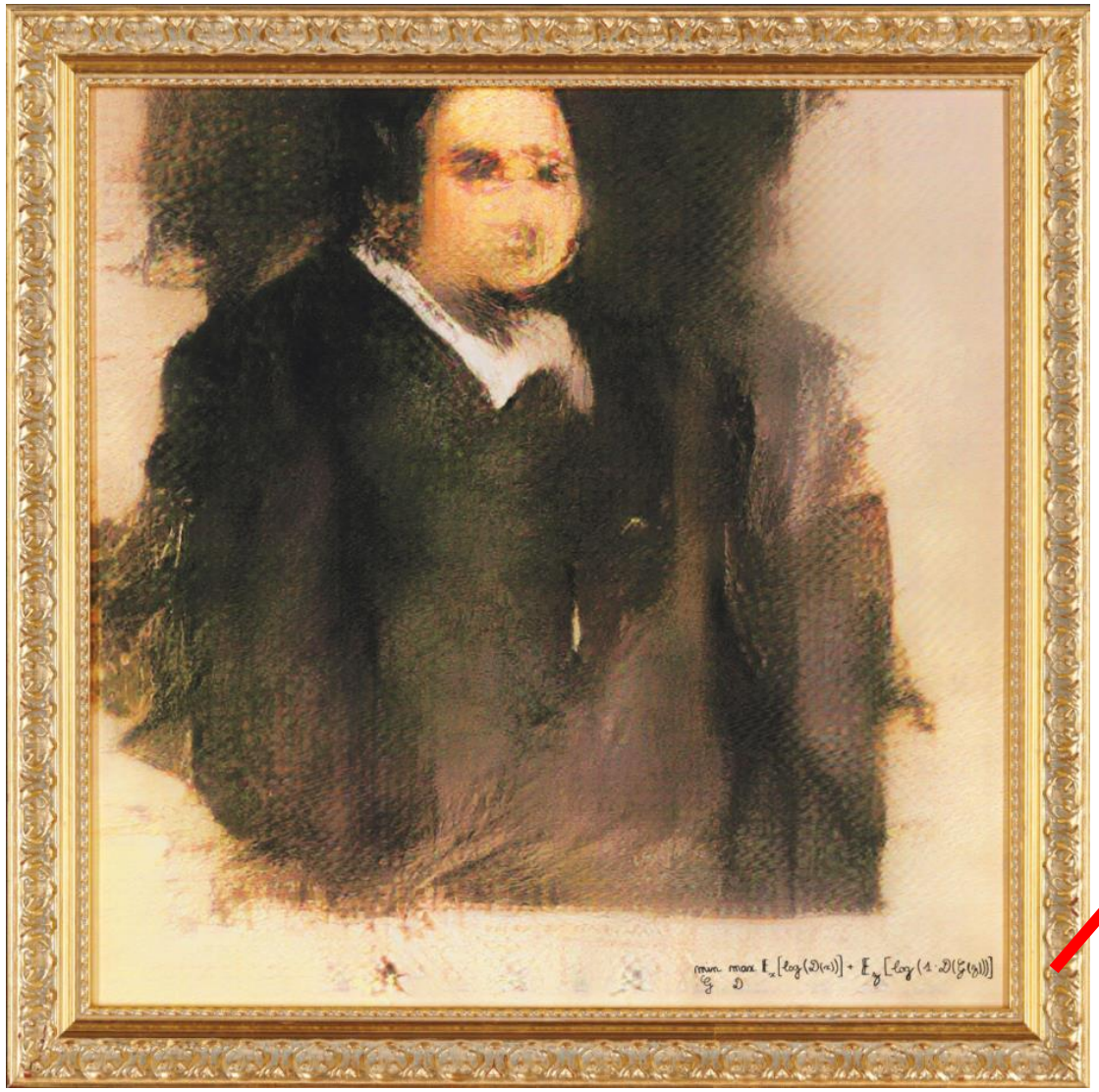


GAN的潜力巨大，因为它们能去学习模仿任何数据分布，因此，GANs能被教导在任何领域创造类似于我们的世界，比如图像、音乐、演讲、散文。在某种意义上，他们是机器人艺术家，他们的输出令人印象深刻，甚至能够深刻的打动人们。

AI DISCOVERY

GAN作画(1)

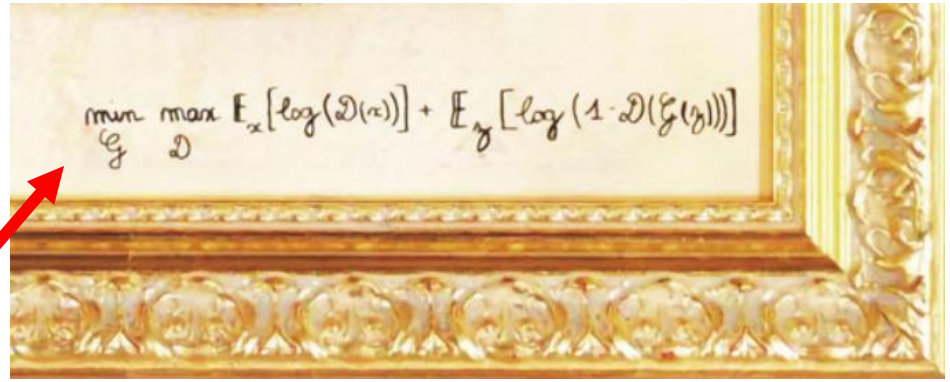
AI DISCOVERY



《埃德蒙·贝拉米画像》(Portrait of Edmond Belamy)

纽约佳士得拍卖会

43.25万美元 (约300万人民币)，成交。



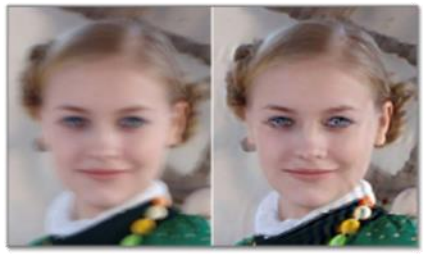
AI DISCOVERY

GAN作画(2)

《贝拉米伯爵夫人 - La Comtesse de Belamy》

超分辨率图像生成

Low-res to high-res



Blurry to sharp

Thermal to color

Synthetic to real

LDR to HDR

Noisy to clean

Image to painting

Day to night

Summer to winter

- Bad weather to good weather
- Greyscale to color
- …

序列生成

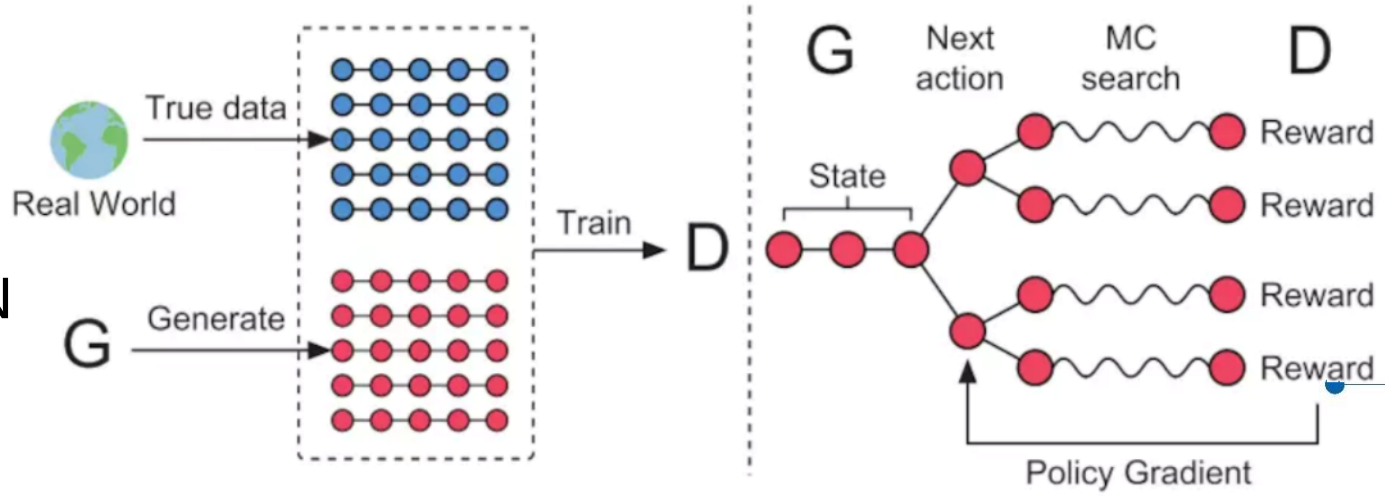
AI DISCOVERY

相比于 GAN 在图像领域的应用，GAN 在文本，语音领域的应用要少很多。主要原因有两个：

1. GAN 在优化的时候使用 BP 算法，对于文本，语音这种离散数据，GAN 没法直接跳到目标值，只能根据梯度一步步靠近。
2. 对于序列生成问题，每生成一个单词，我们就需要判断这个序列是否合理，可是 GAN 里面的判别器是没法做到的。除非我们针对每一个 step 都设置一个判别器，这显然不合理。

为了解决上述问题，强化学习中的策略梯度下降（Policy gradient ）被引入到 GAN 中的序列生成问题。

Sequence GAN



可以用于对话生成、诗句生成、机器翻译等自然语言生成

AI DISCOVERY

目录

AI DISCOVERY

AI DISCOVERY

强化学习

AI DISCOVERY

强化学习的概述

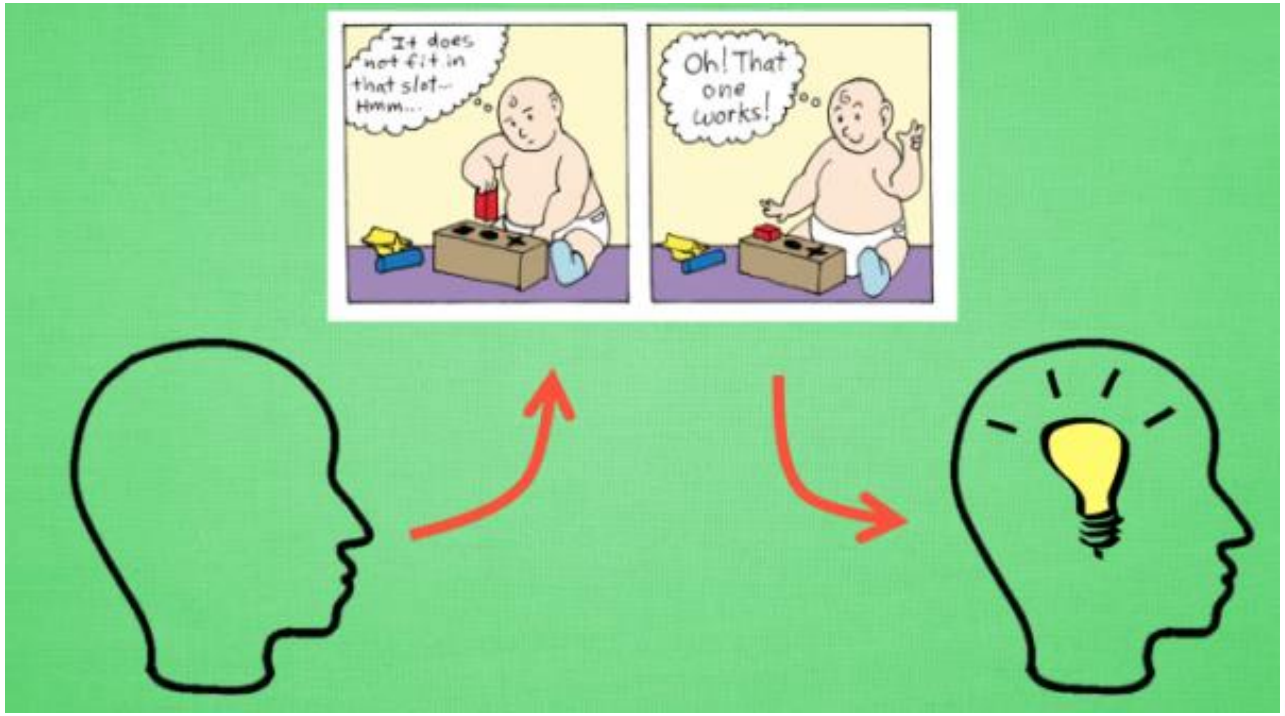
深度强化学习

强化学习的应用

AI DISCOVERY

什么是强化学习

强化学习是一类算法, 是让计算机实现从一开始什么都不懂, 脑袋里没有一点想法, 通过不断地尝试, 从错误中学习, 最后找到规律, 学会了达到目的的方法， 这就是一个完整的强化学习过程



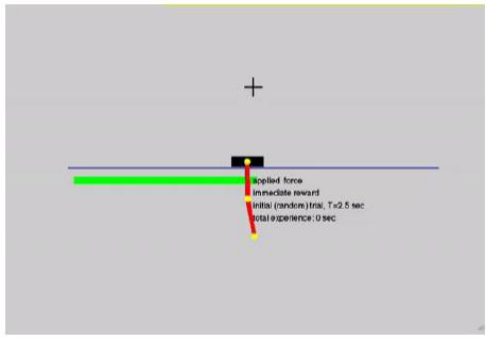
通过什么来学习

AI DISCOVERY

- 计算机需要一位虚拟的老师, 这个老师比较吝啬, 他不会告诉你如何移动, 如何做决定, 他为你做的事只有给你的行为打分
- 算法只需要记住那些高分, 低分对应的行为, 下次用同样的行为拿高分, 并避免低分的行为
- 强化学习具有分数导向性，这种导向性使得模型达到学习的目的

强化学习解决的问题

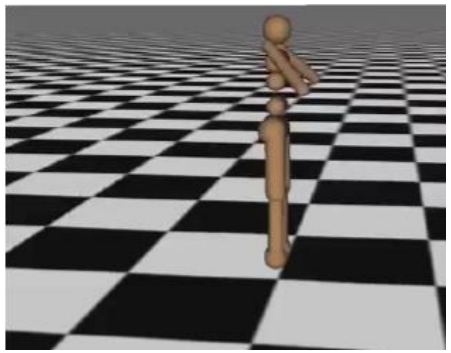
AI DISCOVERY



非线性控制

AlphaGo下围棋

视频游戏

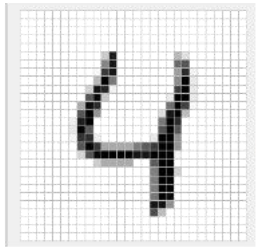


机器人控制

除了非线性控制、视频游戏、下棋、机器人，强化学习还可用于人机对话、无人驾驶、机器翻译、文本序列预测等领域。

强化学习解决的是**智能决策问题**

深度学习解决的是**智能感知问题**



智能感知

4

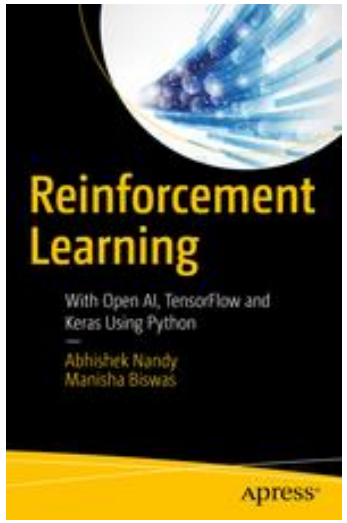
智能决策

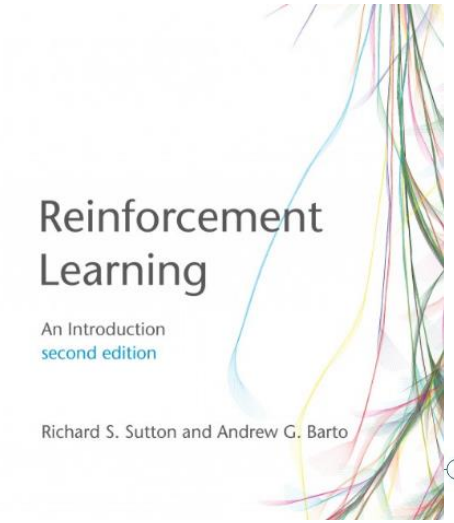
AI DISCOVERY

强化学习定义

AI DISCOVERY

- **Reinforcement learning** (**RL**) is an area of machine learning concerned with how software agents ought to take *actions* in an *environment* so as to maximize some notion of cumulative *reward*.
- **强化学习**是机器学习中的一个领域，强调如何基于环境而行动，以取得最大化的预期利益。其灵感来源于心理学中的行为主义理论，即有机体如何在环境给予的奖励或惩罚的刺激下，逐步形成对刺激的预期，产生能获得最大利益的习惯性行为。



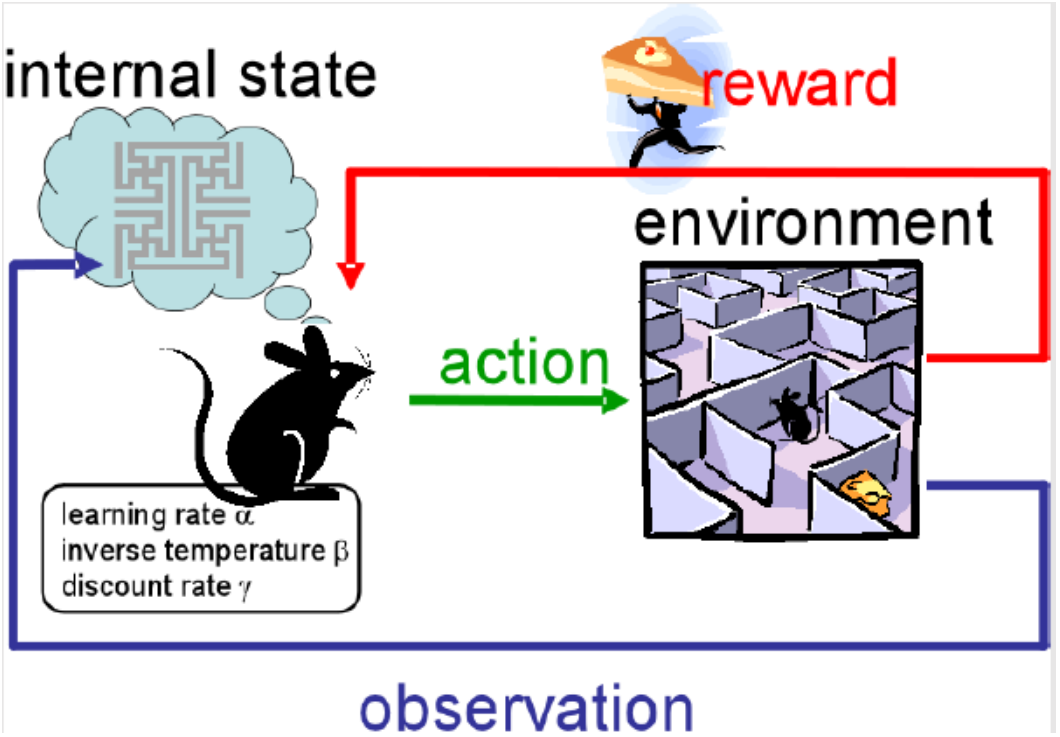


AI DISCOVERY

强化学习 VS. 有监督学习、无监督学习

AI DISCOVERY

- **有监督学习**
  - 主要应用于回归、分类、排序等问题
  - 需要利用带标注的数据学习
- **无监督学习**
  - 主要应用于聚类、降维、密度估计等问题
  - 不需要标注数据
- **强化学习**
  - 主要应用智能决策问题
  - 学习过程可以与环境交互
  - 可以利用延迟、累计回报优化模型

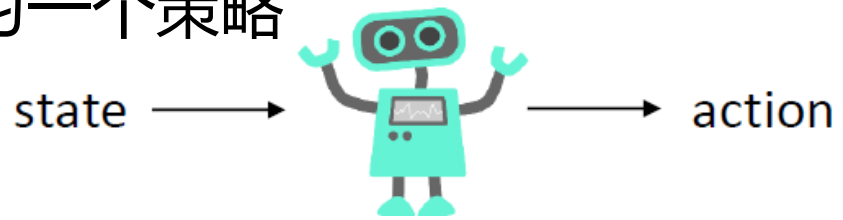


AI DISCOVERY

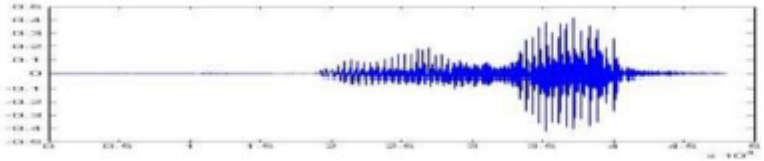
One-short Decision VS. Sequential Decisions

AI DISCOVERY

- 都可以看成是一个Agent学习一个策略



- 有监督学习

$f($  $)=$ “How are you”

$f($  $)=$ “Cat”

- 强化学习

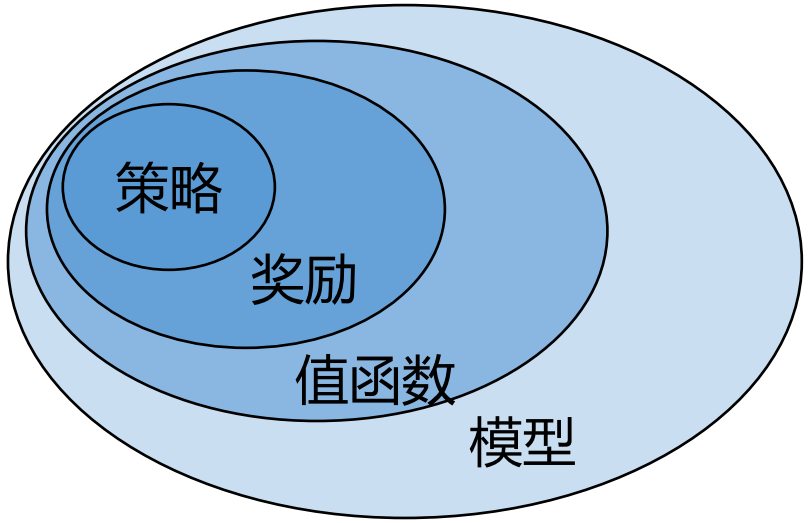



Take a move ..... Take a move ..... Win!

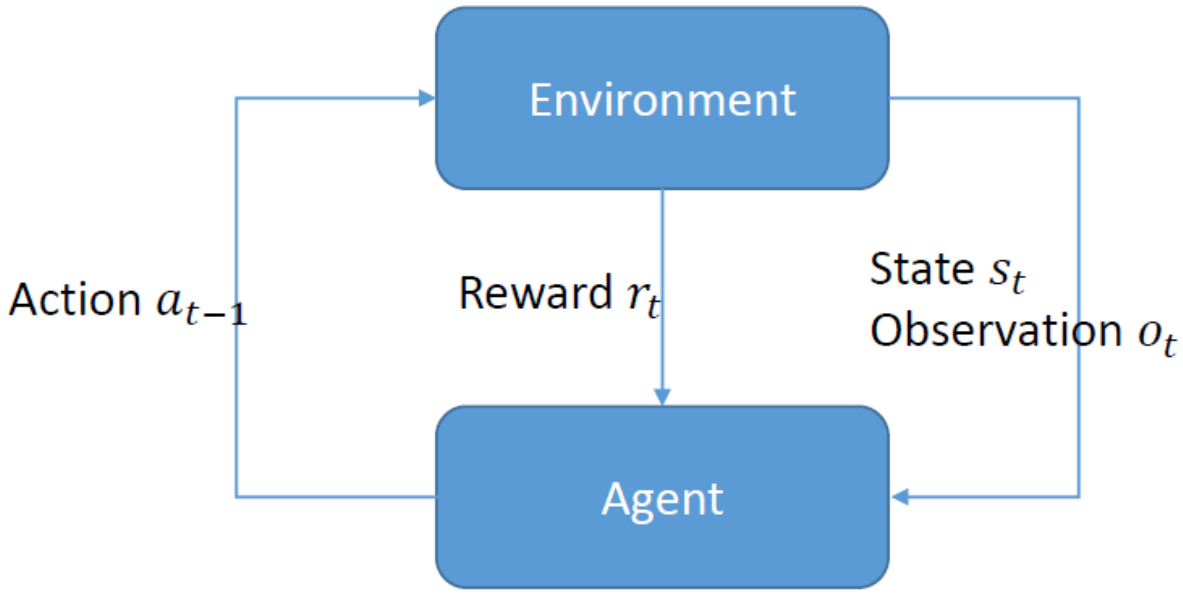
I DISCOVERY

强化学习的基本要素(1)

- **策略 (Policy)**
  - 从状态到动作的一个映射函数，决定了Agent的行为
- **奖励 (Reward Signal)**
  - 每一步环境给Agent的反馈
- **值函数 (Value Function)**
  - Agent对当前状态下未来所能得到的全部奖励的评估
- **模型 (Model, optional)**
  - Agent对于环境的建模，可选

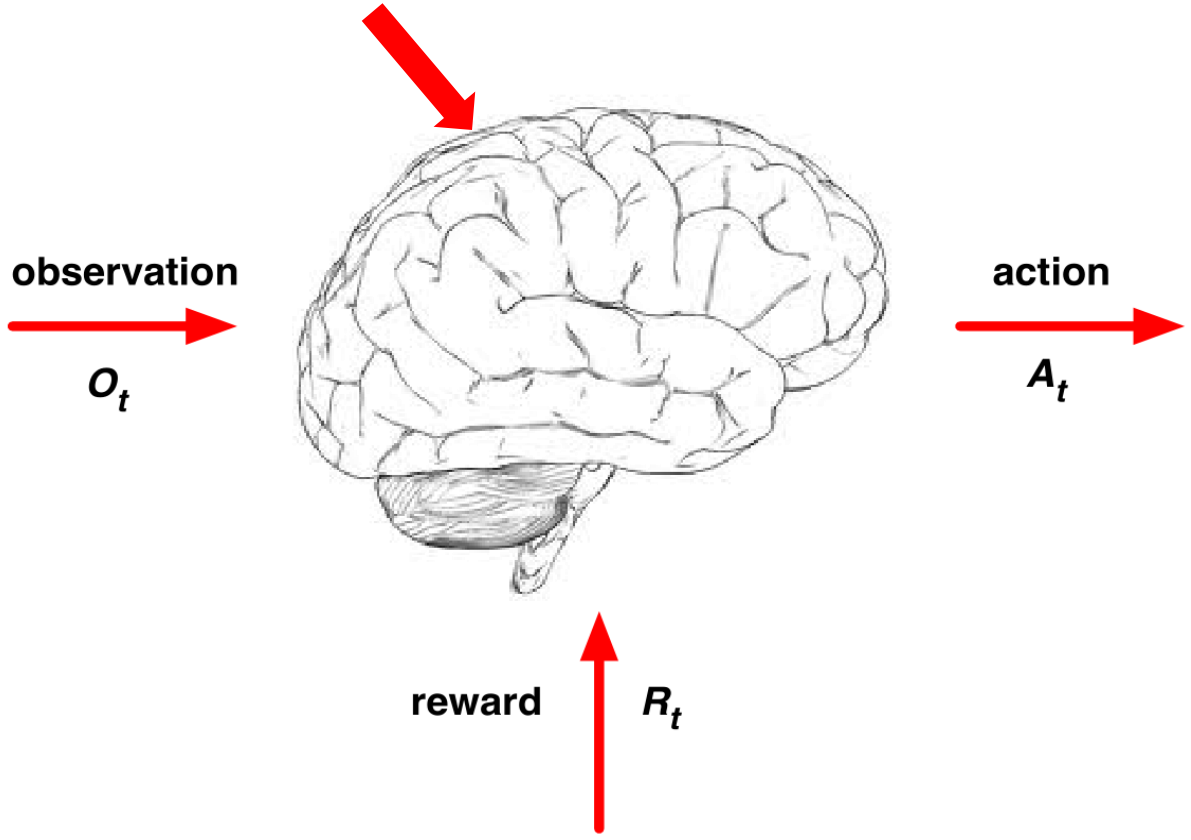


强化学习的基本要素(2)



- 状态S：对环境的描述，可以是离散的或连续的
- 动作A：对智能体行为的描述，Agent的动作集
- 策略P：一系列的状态转移规律
- 回报R：奖励机制及奖励值
- 观测O：观测值

建立Agent的模型



目标：最大化长期累积回报（奖励）

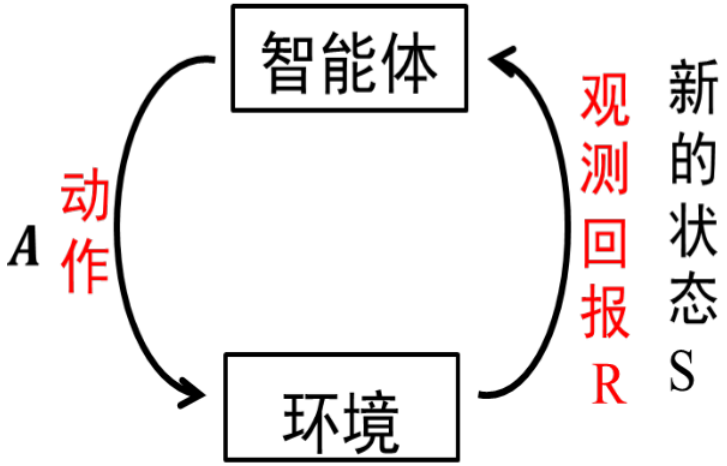
强化学习的工作流程

- **工作流程：**

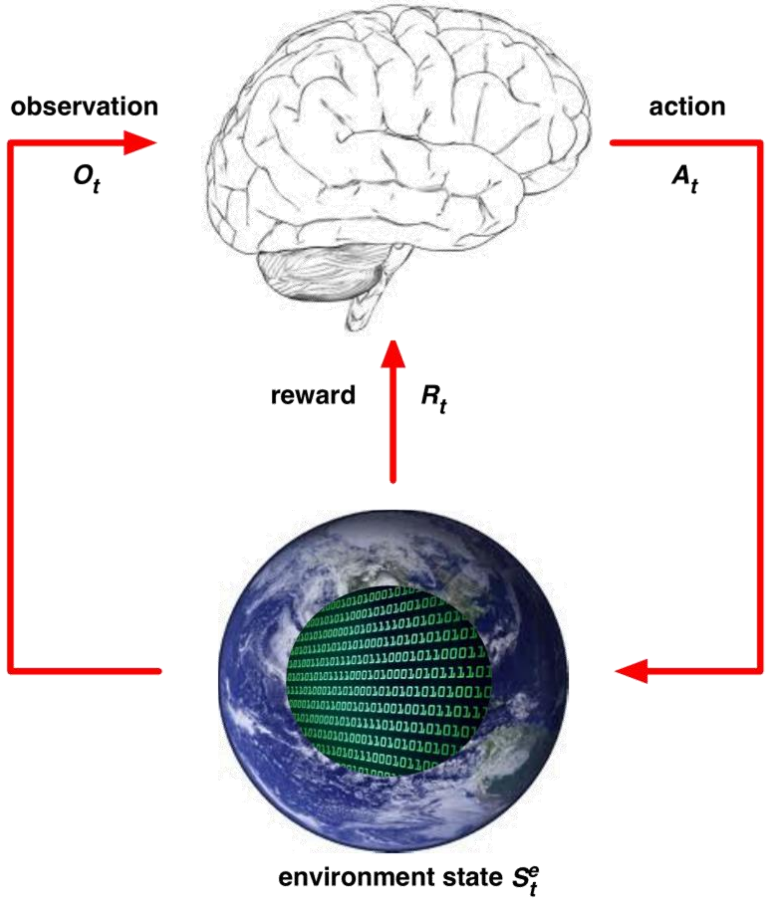
1. 与环境交互，产生数据流（试错学习trial and error）

$$s_0 \xrightarrow{a_1} (r_1, s_1) \xrightarrow{a_2} (r_2, s_2) \rightarrow \cdots$$

2. 智能体利用数据进行学习（延迟的累积回报），优化自身行为（学习一个最优的从状态到动作的策略），期望最终的累积回报最大



环境的观测性与建模



- 状态S_t^e是环境的私有表示，对Agent不可见
- 状态S_t^e可以是任何一个环境用来产生下一个观察或奖励的函数

- **完全可观测**
  - $O_t = S_t^a = S_t^e$
  - 可建模为马尔科夫决策过程（MDP）
- **部分可观测**
  - Agent内部的状态表示不同于环境的状态表示，即$S_t^a \neq S_t^e$
  - S_t^a是Agent自行建立的
- **都是序列决策制定任务**

策略 Policy

AI DISCOVERY

- 智能体的策略（policy）就是智能体如何根据环境状态s来决定下一步的动作a，通常可以分为下面两组：
  - •确定性策略：从状态空间到动作空间的映射函数π：S→A
  - •随机性策略：表示再给定环境状态时，智能体选择某个动作的概率分布

$$\pi(a|s) \triangleq p(a|s), \qquad \sum_{a \in \mathcal{A}} \pi(a|s) = 1.$$

- 通常情况下，强化学习一般使用随机性的策略，因为随机性策略有很多优点：
  - •在学习时可以通过引入一定随机性更好地探索环境
  - •使得策略更加的多样性

AI DISCOVERY

奖励 Reward

AI DISCOVERY

- 给定策略π(a|s)，智能体和环境一次交互过程所收到的累积奖励称为总回报

$$G(\tau) = \sum_{t=0}^{T-1} r_{t+1} = \sum_{t=0}^{T-1} r(s_t, a_t, s_{t+1})$$

•假设环境中有一个或多个特殊的终止状态（terminal state），当到达终止状态时，一个智能体和环境的交互过程就结束了

•这一轮交互的过程称为一个回合（episode）或试验（trial

- 如果环境中没有终止状态（比如终身学习的机器人），即T = ∞，称为持续性强化学习任务，其总回报也可能是无穷大。为了解决这个问题，可以引入一个折扣率来降低远期回报的权重。
- 折扣回报定义为 $G(\tau) = \sum_{t=0}^{T-1} \gamma^t r_{t+1}$
- 其中γ ∈ [0, 1]是折扣率。当γ 接近于0时，智能体更在意短期回报；而当γ 接近于1时，长期回报变得更重要。

值函数 Value-Function - 策略的评估(1)

AI DISCOVERY

- 为了评估一个策略π 的期望回报，我们定义两个值函数：**状态值函数**和**状态-动作值函数**
- **状态值函数**
  - 一个策略π的期望回报可以分解为

$$\mathbb{E}_{\tau\sim p(\tau)}[G(\tau)] = \mathbb{E}_{s\sim p(s_0)}\left[\mathbb{E}_{\tau\sim p(\tau)}\left[\sum_{t=0}^{T-1}\gamma^t r_{t+1}|\tau_{s_0}=s\right]\right]$$
$$= \mathbb{E}_{s\sim p(s_0)}[V^\pi(s)],$$

  - Vπ(s)称为状态值函数（state value function），表示从状态s开始，执行策略π得到的期望总回报
  - 根据马尔科夫性Vπ(s)可展开，经过一系列计算最后得到：

$$V^\pi(s) = \mathbb{E}_{a\sim\pi(a|s)}\mathbb{E}_{s'\sim p(s'|s,a)}\left[r(s,a,s') + \gamma V^\pi(s')\right]$$

  - 上式也称贝尔曼方程，表示当前状态的值函数可以通过下个状态的值函数计算
  - 所以，给定策略π，状态转移概率p和奖励r，可以通过迭代的方式来计算Vπ(s)

值函数 Value-Function - 策略的评估(2)

AI DISCOVERY

- 为了评估一个策略π 的期望回报，我们定义两个值函数：**状态值函数**和**状态-动作值函数**
- **状态-动作值函数**

  •贝尔曼方程中的第二个期望是指初始状态为s 并进行动作a，然后执行策略π 得到的期望总回报，称为状态-动作值函数（state-action value function）

$$Q^{\pi}(s,a) = \mathbb{E}_{s' \sim p(s'|s,a)} \left[r(s,a,s') + \gamma V^{\pi}(s') \right]$$

  - 状态-动作值函数也经常称为Q函数（Q-function）
  - 状态值函数Vπ(s)是Q函数Qπ(s, a)关于动作a的期望：

$$V^{\pi}(s) = \mathbb{E}_{a \sim \pi(a|s)} \left[Q^{\pi}(s,a) \right]$$

强化学习的分类(1)

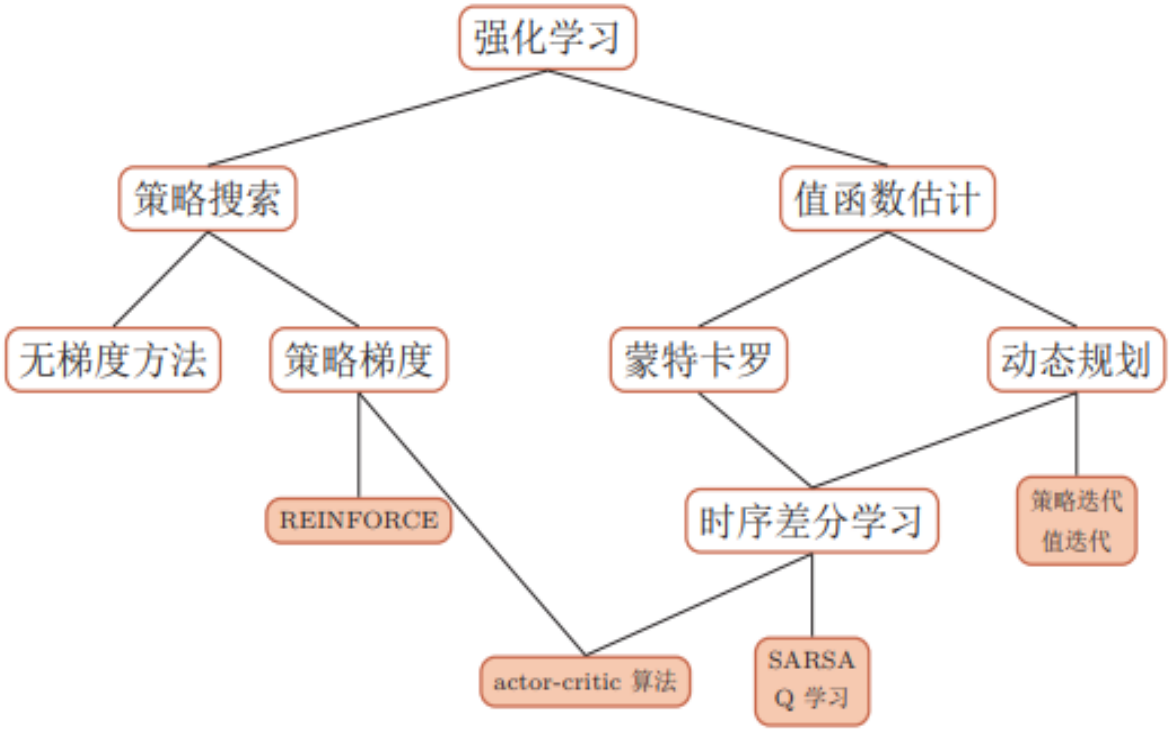
AI DISCOVERY

- 根据是否依赖模型：
  - 基于模型的强化学习（Model-based）
  - 无模型的强化学习（Model-free）
- 根据策略更新和学习方法：
  - 基于价值的强化学习（Value-based） （Q learning， Sarsa）
  - 基于概率的强化学习（Policy-based）(Policy Gradients)
  - 结合价值和概率优势的AC算法(Actor Critic)
- 根据更新的时机：
  - 回合更新（Monte-carlo learning, basic policy gradients）
  - 单步更新（Q-learning, Sarsa, Advanced policy gradients）
- 根据是否亲自参与：
  - 在线学习（On-Policy） （Sarsa）
  - 离线学习（Off-Policy） （Q learning， Deep-Q-Network）

AI DISCOVERY

强化学习的分类(2)

AI DISCOVERY



- 常用分类
  - 基于值函数的方法
  - 基于策略函数的方法
  - 融合两种的方法
- 基于值函数的方法策略更新时可能会导致值函数的改变比较大，对收敛性有一定影响，而基于策略函数的方法在策略更新时更加更平稳些。
- 但后者因为策略函数的解空间比较大，难以进行充分的采样，导致方差较大，并容易收敛到局部最优解。
- Actor-Critic算法通过融合两种方法，取长补短，有着更好的收敛性。

AI DISCOVERY

强化学习

AI DISCOVERY

强化学习的概述

深度强化学习

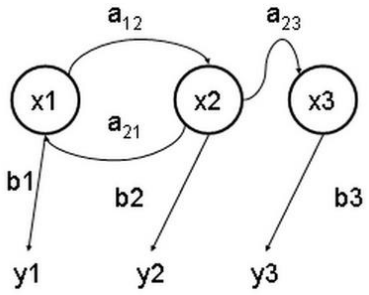
强化学习的应用

AI DISCOVERY

早期强化学习 VS. 深度强化学习

AI DISCOVERY

- 早期的强化学习算法主要关注于状态和动作都是离散且有限的问题，可以使用表格来记录这些概率。但在很多实际问题中，有些任务的状态和动作的数量非常多，还有些任务的状态和动作是连续的。
- 相比于强化学习，深度强化学习通常利用卷积层中隐藏神经元来探测到各式各样的state（取决于卷积层的权重分配），由全连接层的权重分配决定action。
- 用强化学习来定义问题和优化目标，用深度学习来解决策略和值函数的建模问题，然后使用误差反向传播算法来优化目标函数。

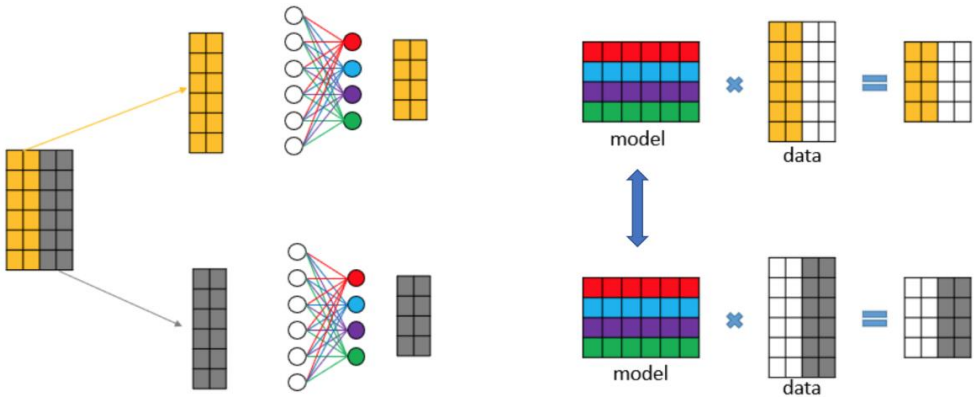


AI DISCOVERY

深度学习 VS. 强化学习

AI DISCOVERY

- 深度学习的基本思想是通过多层的网络结构和非线性变换，组合低层特征，形成抽象的、易于区分的高层表示，以发现数据的分布式特征表示。因此 深度学习方法**侧重于对事物的感知和表达。**
- 强化学习的基本思想是通过最大化智能体（agent）从环境中获得的累计奖赏值，以学习到完成目标的最优策略。因此强化学习方法更加**侧重于学习解决问题的策略。**
- 随着人类社会的飞速发展，在越来越多复杂的现实场景任务中，需要利用 深度学习来自动学习大规模输入数据的抽象表征，并以此表征为依据进行自我激励的强化学习，优化解决问题的策略。



深度Q网络 Deep Q-learning Network

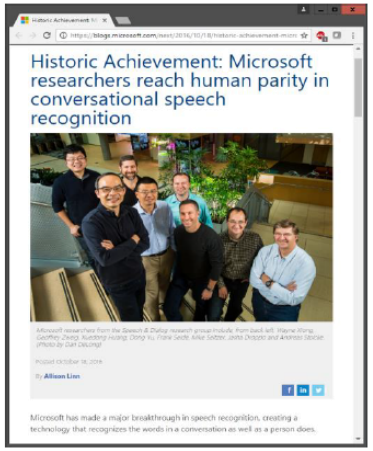
AI DISCOVERY

Q-learning算法是1989年Watkins提出来的，2015年nature论文所提出的DQN就是在Q-learning的基础上修改得到的。

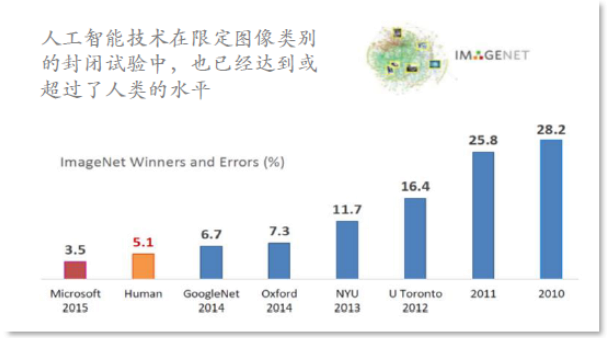
Q-learning方法采用异策略和时间差分。

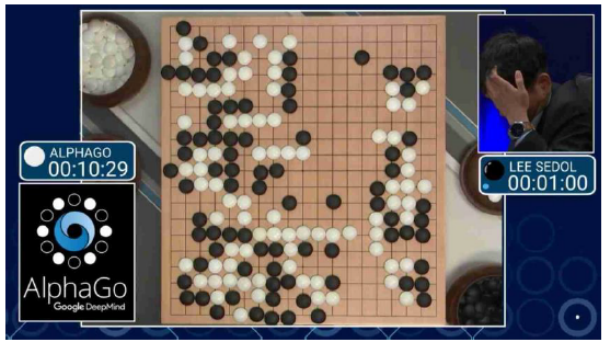
- 异策略，是指行动策略（产生数据的策略）和要评估的策略不是一个策略。
- 时间差分方法，是指利用时间差分目标来更新当前行为值函数。

Deep Learning Is Making Break-through!



2016年10月，微软的语音识别系统在日常对话数据上，达到了5.9%的单词错误率，首次取得与人类相当的识别精度





机器翻译新突破，微软中英新闻翻译达人类水平

原创 2018-03-15 camel AI科技评论

翻译没有唯一标准答案，它更像是一种艺术。

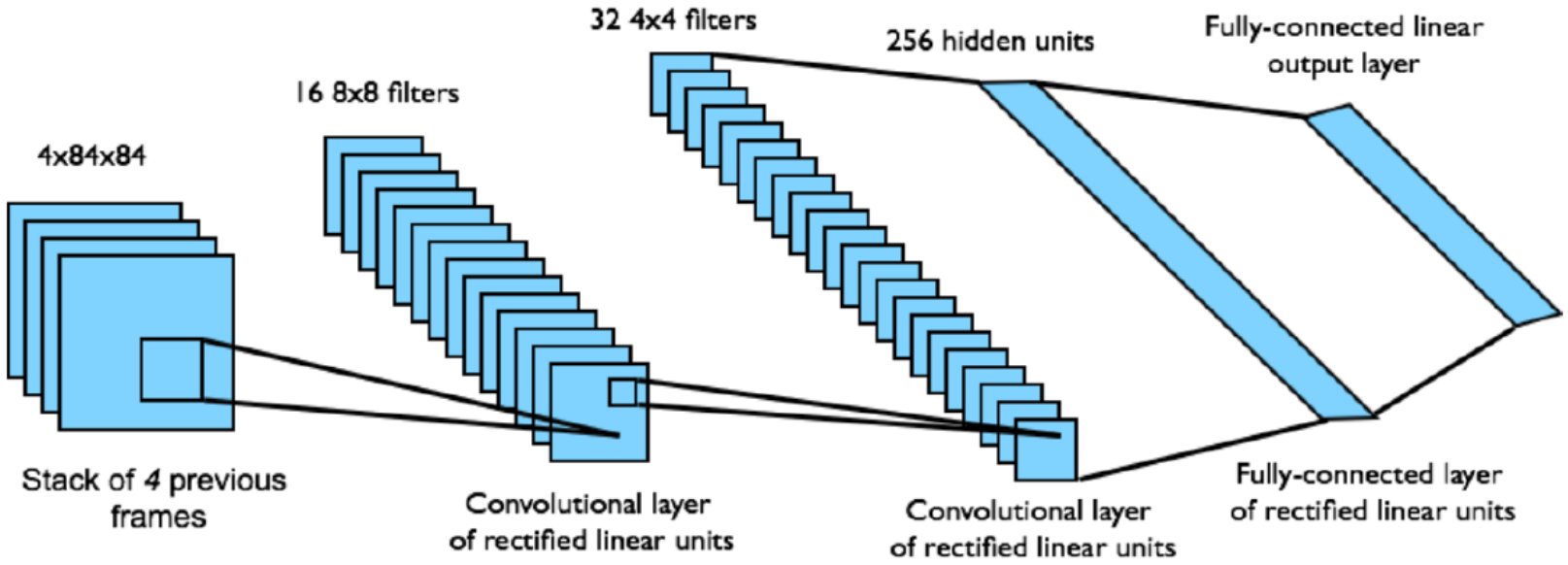
AI科技评论消息： 14 日晚，微软亚洲研究院与雷德蒙研究院的研究人员宣布，其研发的机器翻译系统在通用新闻报道测试集newstest2017 的中-英测试集上，达到了可与人工翻译媲美的水平；这是首个在新闻报道的翻译质量和准确率上可以比肩人工翻译的翻译系统。

AI DISCOVERY

DQN建模

AI DISCOVERY

- 针对Atari Games中的每个像素end-to-end学习
- 输入状态由最后4帧的所有像素组成
- 输出是所有操作的
- 回报是对应当时的游戏分数变化



Q-Learning

- 值函数由参数是　的深度Q网络学习得到：

$$Q(s,a;\theta) \approx Q^{\pi}(s,a)$$

- 优化目标：

$$L(\theta) = E\left[\left(r + \gamma \max_{a'} Q(s',a';\theta) - Q(s,a;\theta)\right)^2\right]$$

- Q-Learning 梯度：

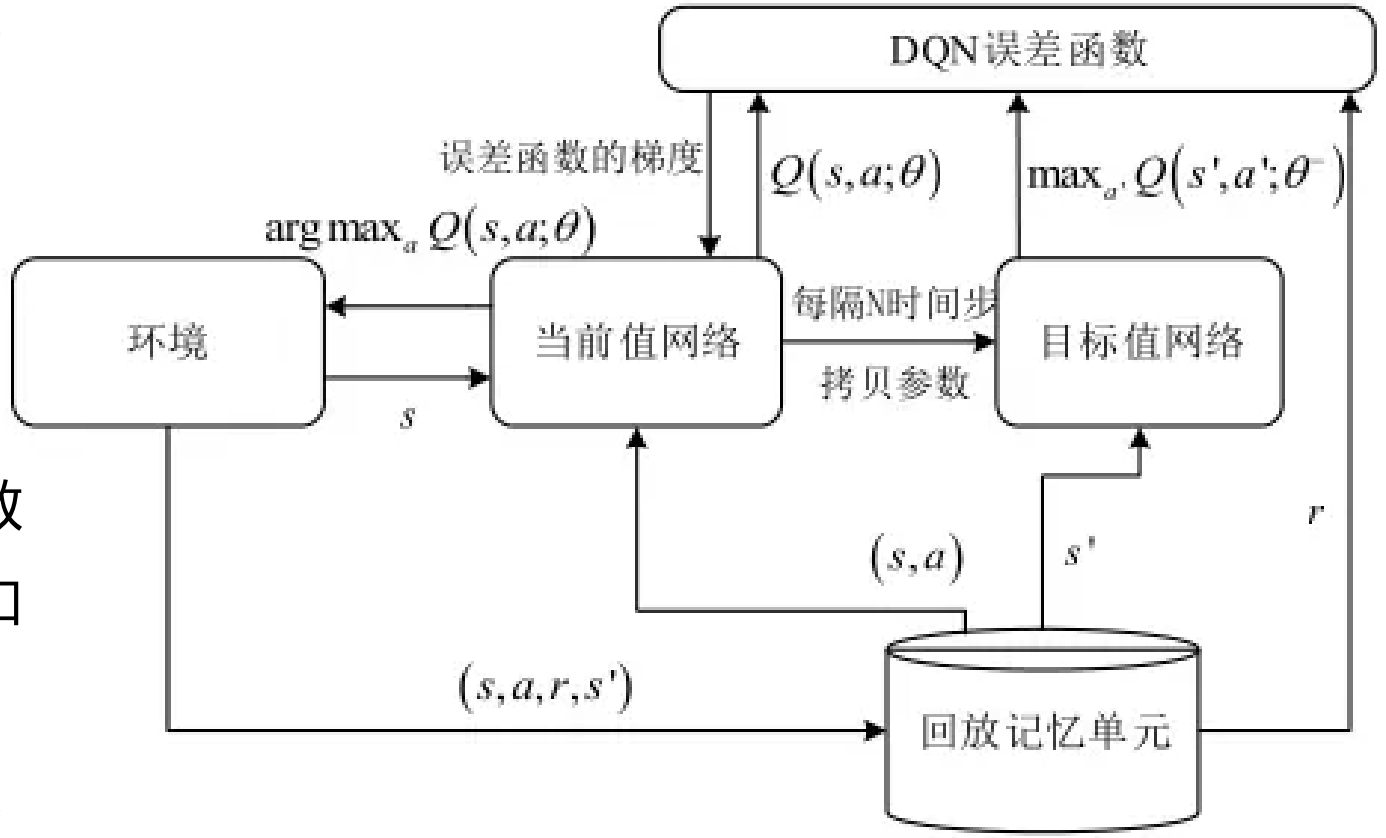
$$\frac{\partial L(\theta)}{\partial \theta} = E\left[\left(r + \gamma \max_{a'} Q(s',a';\theta) - Q(s,a;\theta)\right)\frac{\partial Q(s,a;\theta)}{\partial \theta}\right]$$

- 利用SGD优化

DQN的训练流程

AI DISCOVERY

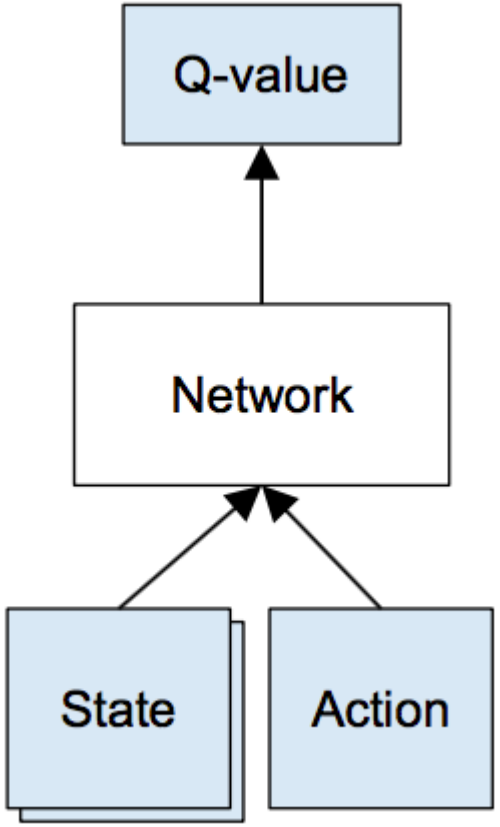
- 在训练过程中使用经验回放机制在线处理得到的转移样本
- 除了使用深度卷积网络近似表示当前的值函数之外，还单独使用了另一个网络来产生目标 Q 值。当前值网络的参数 θ 是实时更新的，每经过 N 轮迭代，将当前值网络的参数复制给目标值网络。通过最小化当前 Q 值和目标 Q 值之间的均方误差来更新网络参数。
- 将奖赏值和误差项缩小到有限的区间内，保证了 Q 值和梯度值都处于合理的范围内，提高了算法的稳定性。



AI DISCOVERY

DQN的稳定性问题

- 朴素的DQN很难优化，波动比较大
- 数据是序列的
  - 相互之间有联系
  - 不是独立的
- 策略变化较快
  - 策略比较震荡
  - 数据的分布会从一个极端变化到另一个极端



DQN稳定性问题的解决方法

- 经验回放
  - 打破数据关联，回归数据独立同分布假设
  - 从所有的历史策略中学习
  - 离线的Q-Learning

$$L(\theta) = E_{s,a,r,s' \sim D}\left[\left(r + \gamma \max_{a'} Q(s', a'; \theta) - Q(s, a; \theta)\right)^2\right]$$

- 固定目标Q网络
  - 避免数据震荡
  - 打破Q-网络和目标的关联

$$L(\theta) = E_{s,a,r,s' \sim D}\left[\left(r + \gamma \max_{a'} Q(s', a'; \theta^-) - Q(s, a; \theta)\right)^2\right]$$

强化学习

AI DISCOVERY

强化学习的概述

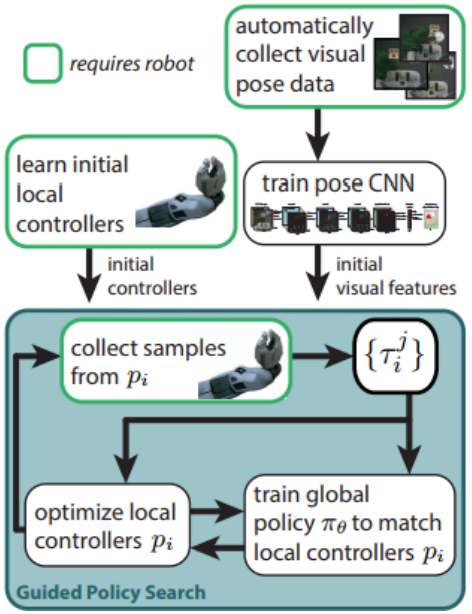
深度强化学习

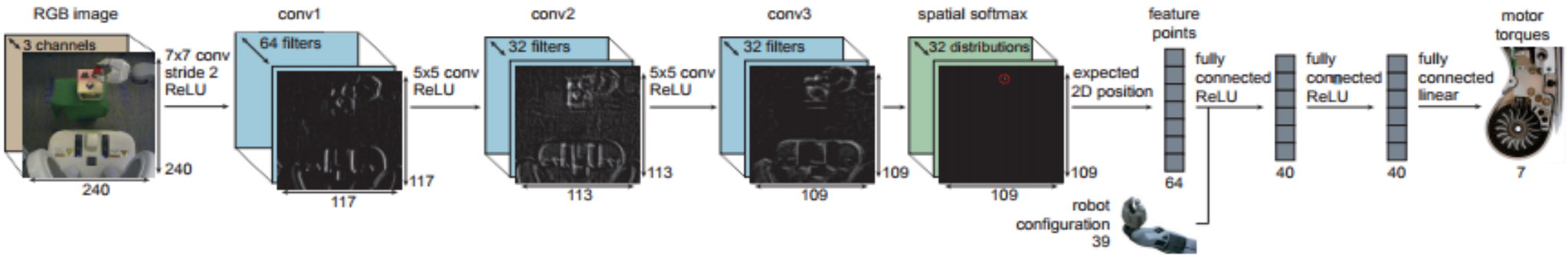
强化学习的应用

机器人

AI DISCOVERY

- End to end training of deep visuomotor policies





AI DISCOVERY

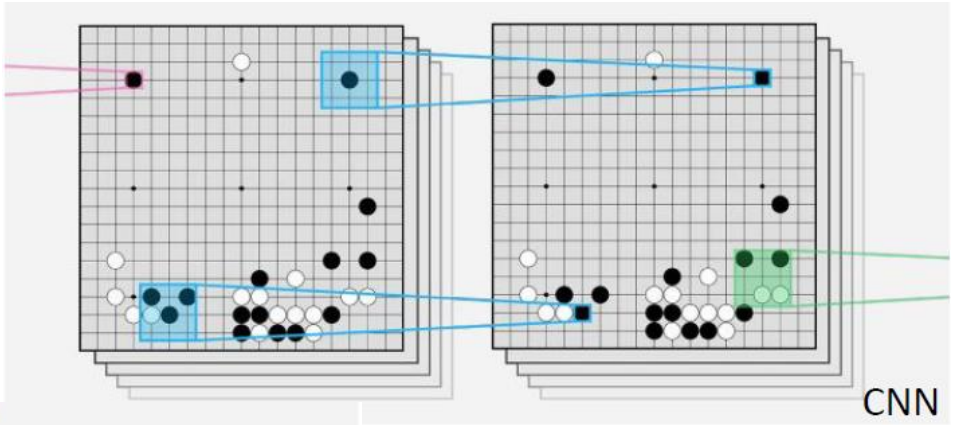
Levine S, Finn C, Darrell T, et al. End-to-end training of deep visuomotorpolicies[J]. Journal of Machine Learning Research, 2016, 17(39): 1

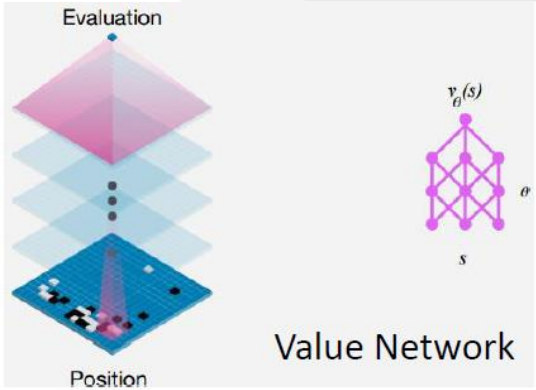
游戏

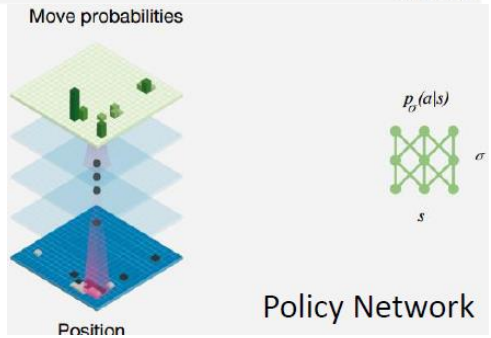
AI DISCOVERY

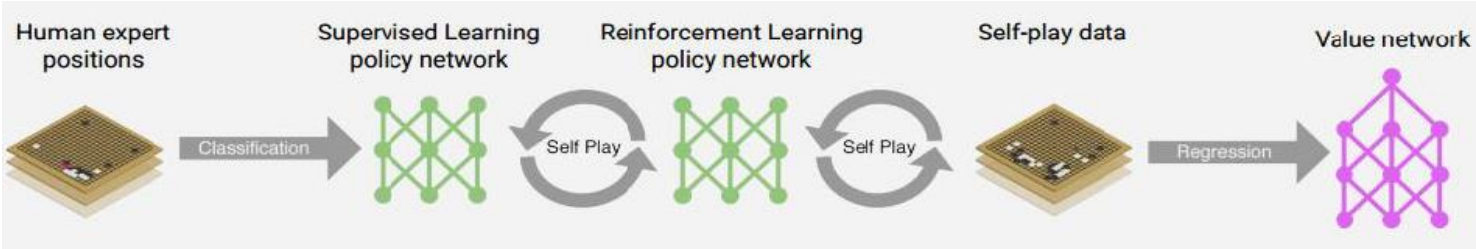
- **Atari Games**
  - **AlphaGo 4:1**
  - **Master 60:0**

AlphaGo









AI DISCOVERY

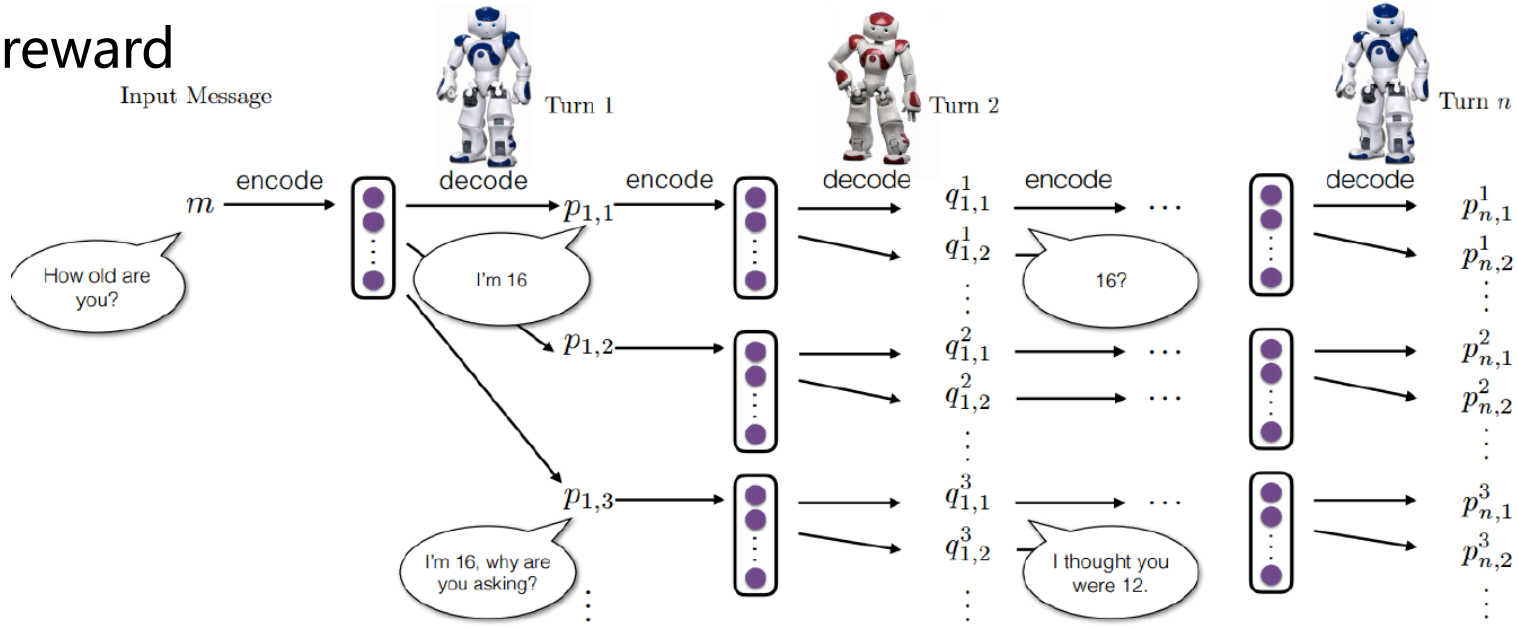
http://icml.cc/2016/tutorials/AlphaGo-tutorial-slides.pdf

多轮对话生成

AI DISCOVERY

- 本工作使用深度强化学习解决多轮对话问题。首先使用Seq-to-Seq模型预训练一个基础模型，然后根据作者提出的三种Reward来计算每次生成的对话的好坏，并使用policy network的方法提升对话响应的多样性、连贯性和对话轮次。

- 文章最大的亮点就在于定义了三种reward
  - Ease of answering
  - Information Flow
  - Semantic Coherence
- 分别用于解决
  - dull response
  - repetitive response
  - ungrammatical response



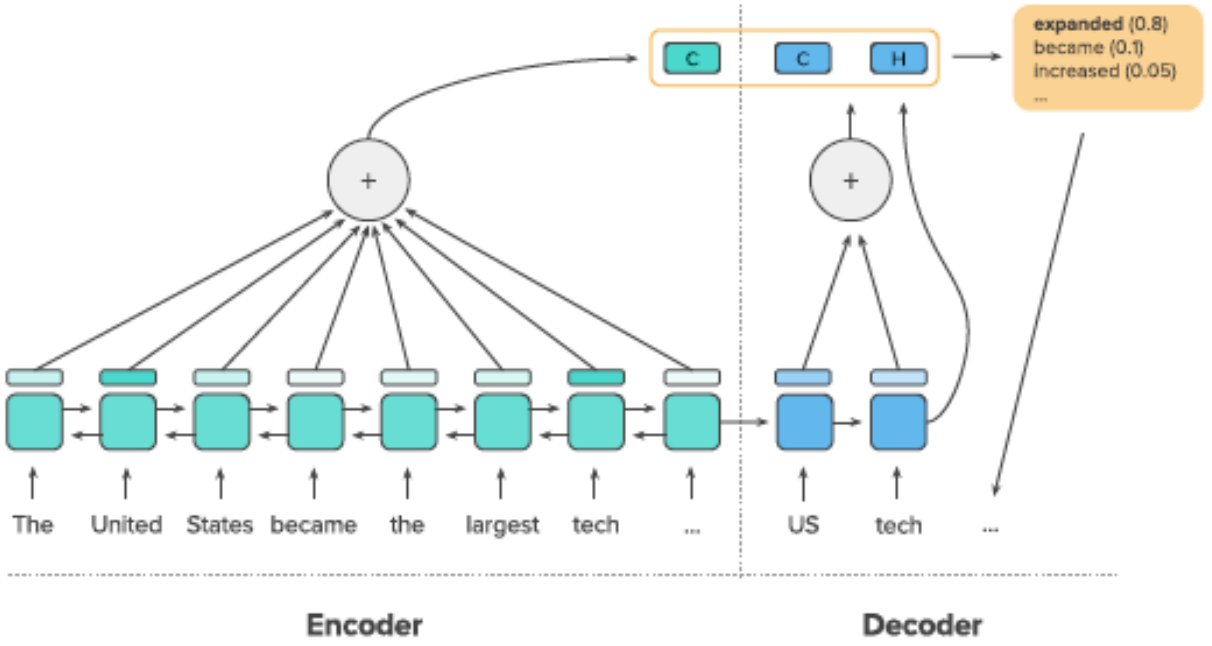
AI DISCOVERY

Li J, Monroe W, Ritter A, et al. Deep Reinforcement Learning for Dialogue Generation[J].

摘要生成

- 强化学习是一个序列预测过程，只要某个问题能转化成序列预测过程，并且每次预测都有收益，就可以使用强化学习，比如自动摘要。

- 长文本摘要问题中，已有模型会输出重复的语句和不连贯的短语。
- 本工作的解决方法是通过分别在输入和输出引入内部注意力机制进行解决。
- 词到词的监督式学习和极大似然目标函数会使得模型存在偏置问题，即训练时有监督的教，预测时可能会造成累计错误传播。



- 所以在训练时引入ROUGE指标，通过比较参考摘要和生成的摘要，给出摘要的评价。但由于ROUGE不可导，无法直接对ROUGE进行梯度计算。因此，可以考虑用强化学习将ROUGE指标加入训练目标，reward函数用Rouge函数替代。

Paulus R, Xiong C, Socher R. A deep reinforced model for abstractive summarization[J]. 2017

目录

AI DISCOVERY

AI DISCOVERY

课程实践

实践：手写数字生成