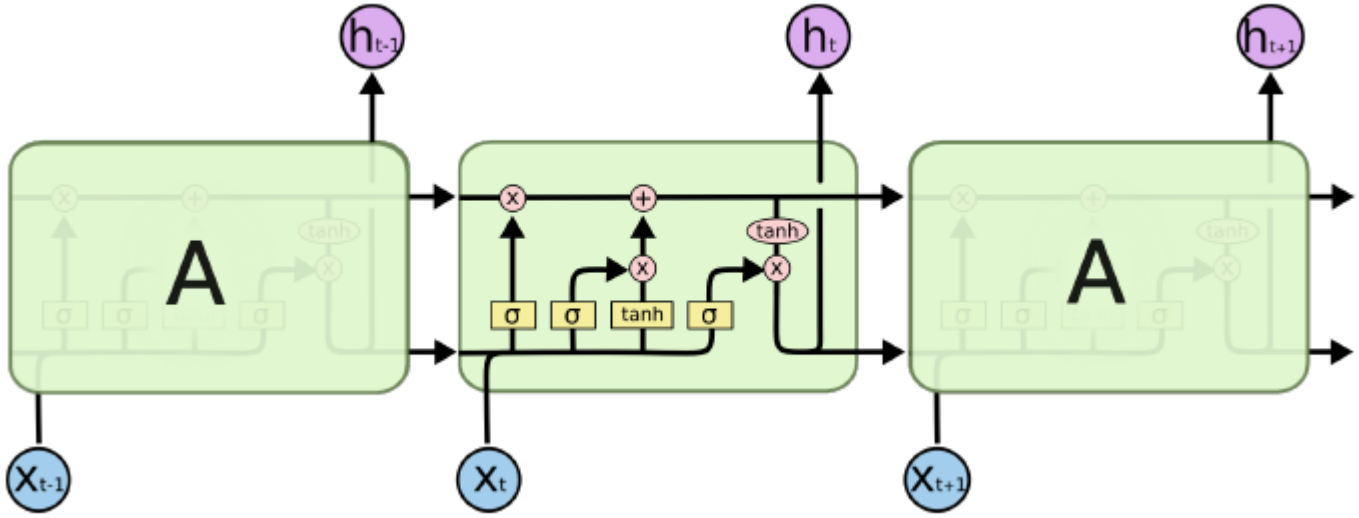
复习

- 观察模型架构，并回答以下问题：



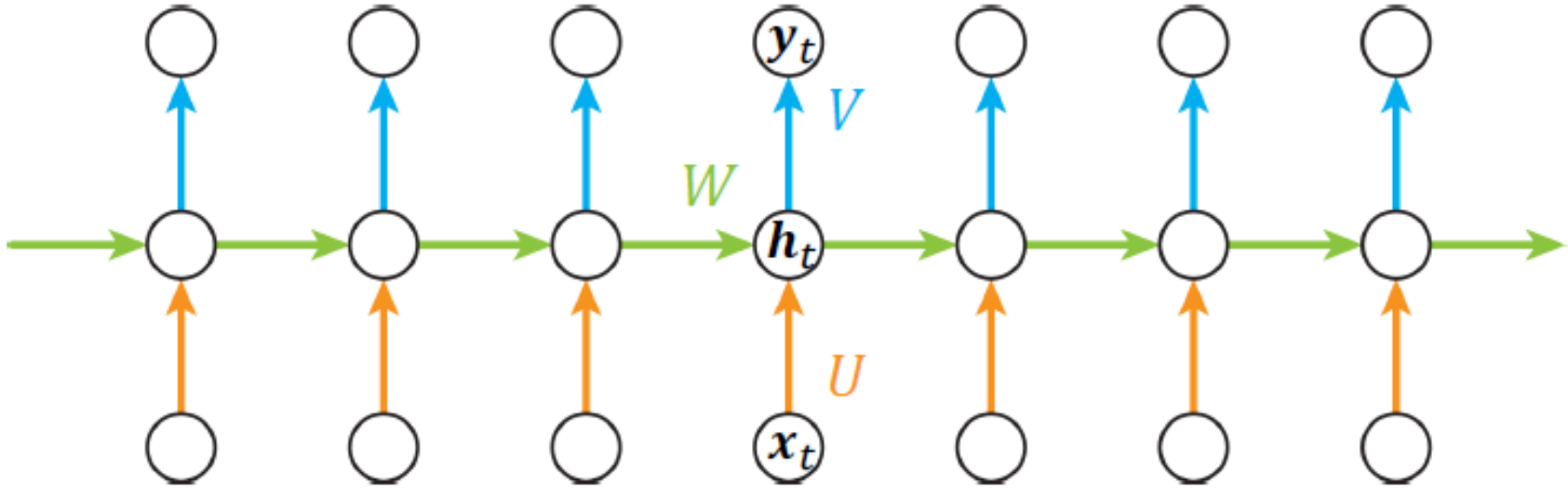
- 请写出图中所示模型的名称，并简要陈述你的判断依据；
- 图中所示模型使用了哪些门控单元？试标注在图上，并简述这些门控单元的作用。

复习

- 对于RNN而言，同步的序列到序列模式有哪些应用场景？异步的序列到序列模式有哪些应用场景?
- RNN模型存在的主要问题是什么？如何解决这一问题?
- 判断正误:
  - 与卷积神经网络相比，循环神经网络同样具有局部连接和权值共享的特点;
  - 循环神经网络在整个序列的损失可定义为不同时间步的损失之和;
  - 循环神经网络可通过随时间的反向传播算法（BPTT）进行训练，由于不同时刻的状态相互依赖，所以需要存储每个时刻的状态信息，难以并行计算;
  - 在GRU中，重置门决定先前隐藏状态单元是否被忽略，而更新门控制当前隐藏状态单元是否需要被新的隐藏状态单元更新;
  - 在GRU中，增加了一个记忆单元C_t，其在不同时刻有着可变的连接权重，以缓解梯度消失和梯度爆炸问题。

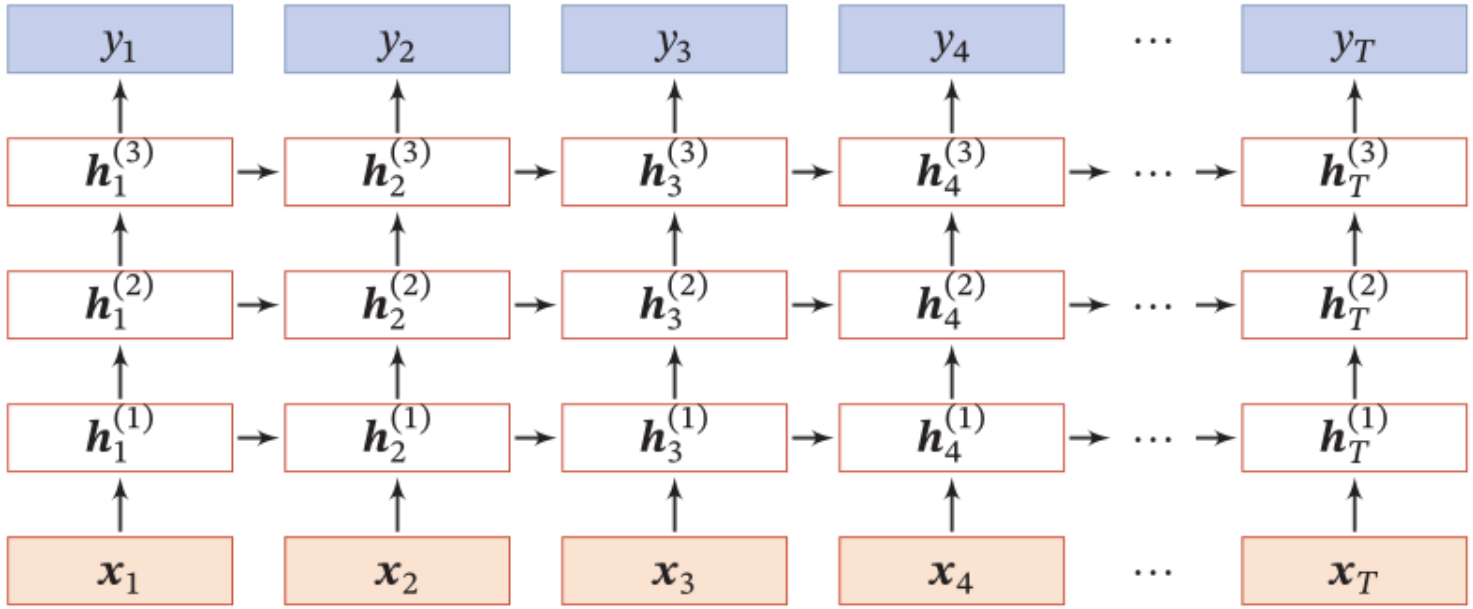
复习

- 下图给出了RNN的前向传播路径，试在下图中绘制出与x_t直接相关的反向传播路径。

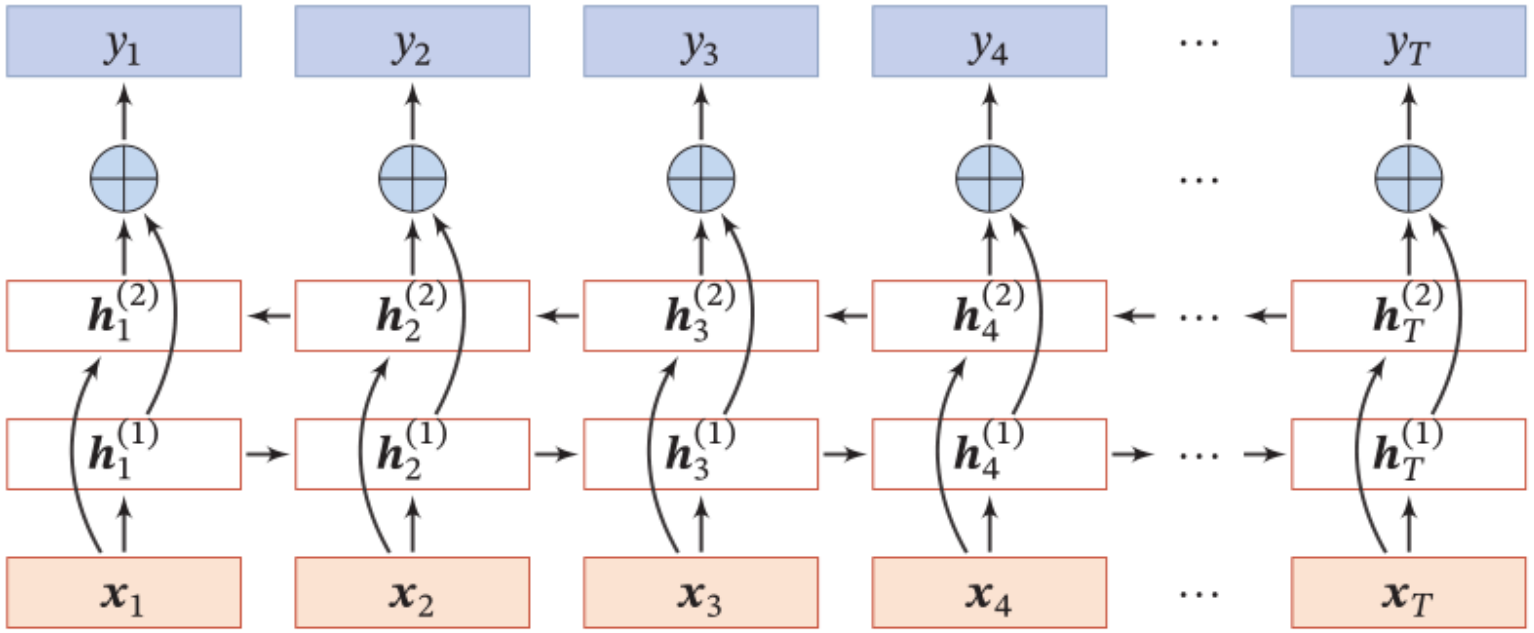


深层循环神经网络

堆叠循环神经网络


y_1 y_2 y_3 y_4 ⋯ y_T
$\boldsymbol{h}_1^{(3)}$ $\boldsymbol{h}_2^{(3)}$ $\boldsymbol{h}_3^{(3)}$ $\boldsymbol{h}_4^{(3)}$ ⋯ $\boldsymbol{h}_T^{(3)}$
$\boldsymbol{h}_1^{(2)}$ $\boldsymbol{h}_2^{(2)}$ $\boldsymbol{h}_3^{(2)}$ $\boldsymbol{h}_4^{(2)}$ ⋯ $\boldsymbol{h}_T^{(2)}$
$\boldsymbol{h}_1^{(1)}$ $\boldsymbol{h}_2^{(1)}$ $\boldsymbol{h}_3^{(1)}$ $\boldsymbol{h}_4^{(1)}$ ⋯ $\boldsymbol{h}_T^{(1)}$
$\boldsymbol{x}_1$ $\boldsymbol{x}_2$ $\boldsymbol{x}_3$ $\boldsymbol{x}_4$ ⋯ $\boldsymbol{x}_T$

双向循环神经网络


y_1
y_2
y_3
y_4
$\cdots$
y_T
$\boldsymbol{h}_1^{(2)}$
$\boldsymbol{h}_2^{(2)}$
$\boldsymbol{h}_3^{(2)}$
$\boldsymbol{h}_4^{(2)}$
$\boldsymbol{h}_T^{(2)}$
$\boldsymbol{h}_1^{(1)}$
$\boldsymbol{h}_2^{(1)}$
$\boldsymbol{h}_3^{(1)}$
$\boldsymbol{h}_4^{(1)}$
$\boldsymbol{h}_T^{(1)}$
$\boldsymbol{x}_1$
$\boldsymbol{x}_2$
$\boldsymbol{x}_3$
$\boldsymbol{x}_4$
$\boldsymbol{x}_T$

双向循环神经网络

```
I am _____
I am _____ very hungry,
I am _____ very hungry, I could eat half a pig.
```

```
I am happy.
I am not   very hungry,
I am very  very hungry, I could eat half a pig.
```

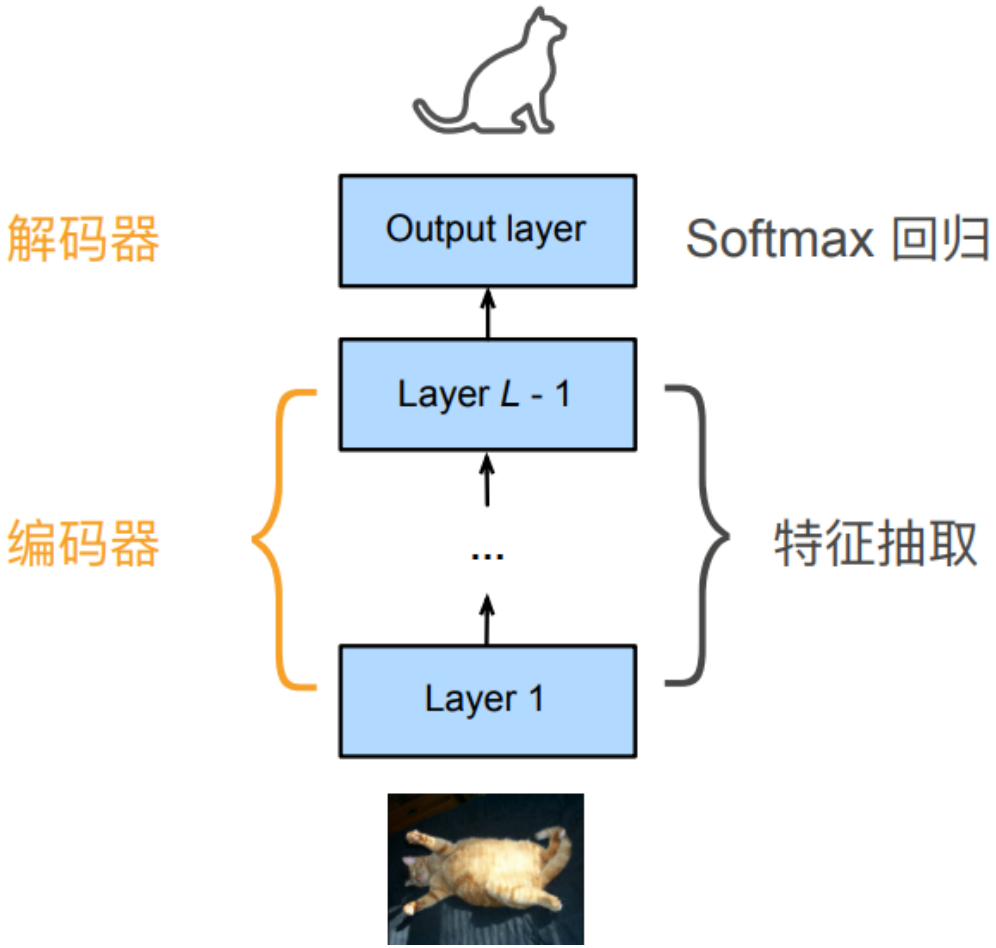
循环神经网络总结

- 优点：
  - 引入（短期）记忆
  - 图灵完备：能够近似任意可计算问题

- 缺点：
  - 长程依赖问题
  - 记忆容量问题
  - 并行能力：只能逐时间步计算

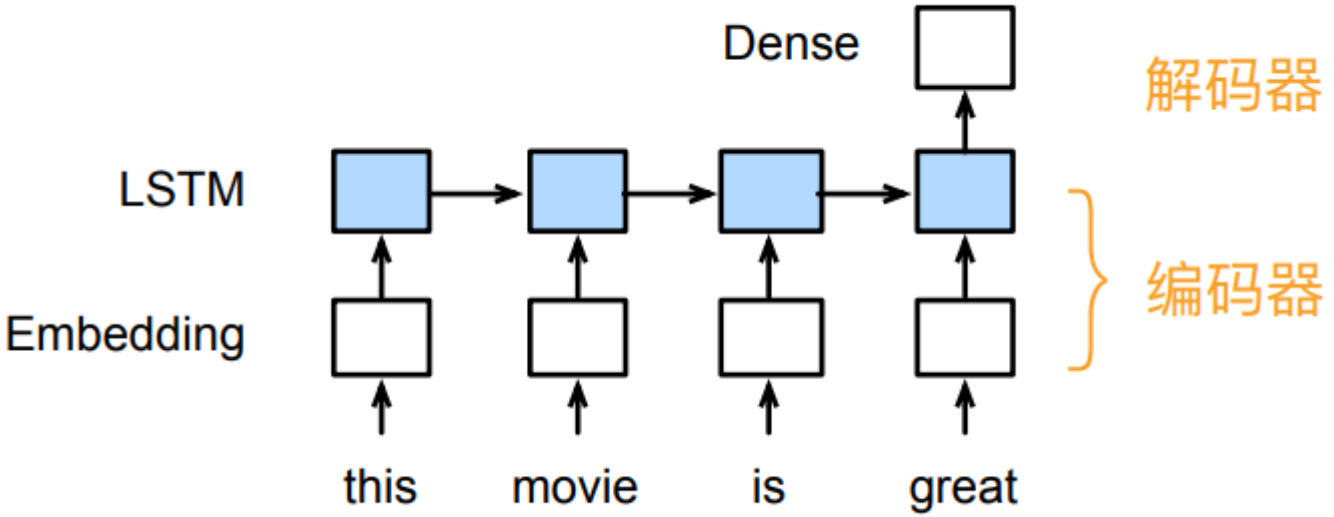
编码器-解码器架构

重新审视CNN



- 编码器（Encoder）：将输入编码为中间表达形式（特征）；
- 解码器（Decoder）：将中间表示解码成输出。

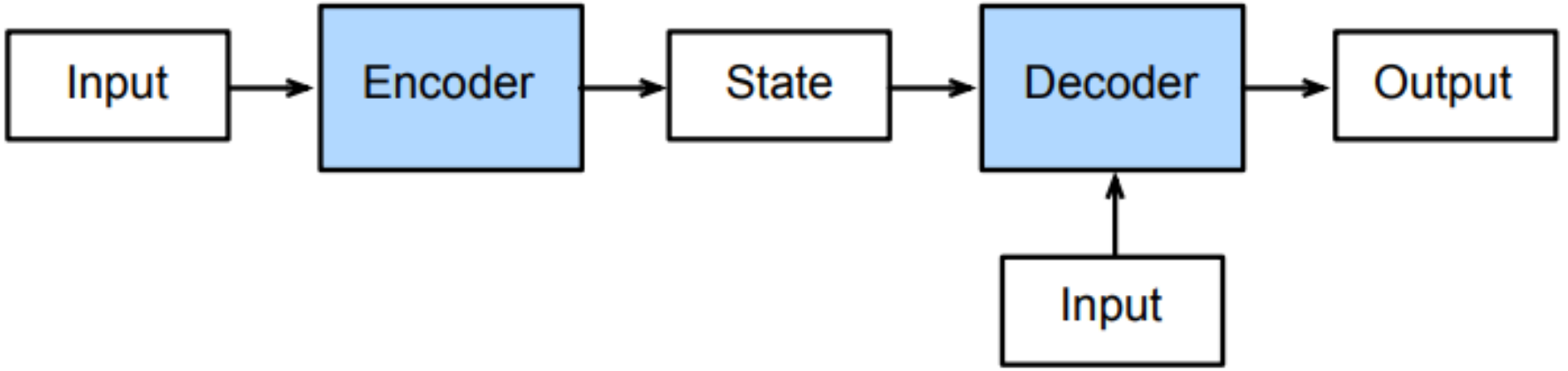
RNN中的编码器-解码器



- 编码器：将序列数据表示成向量；
- 解码器：将向量表示成输出。

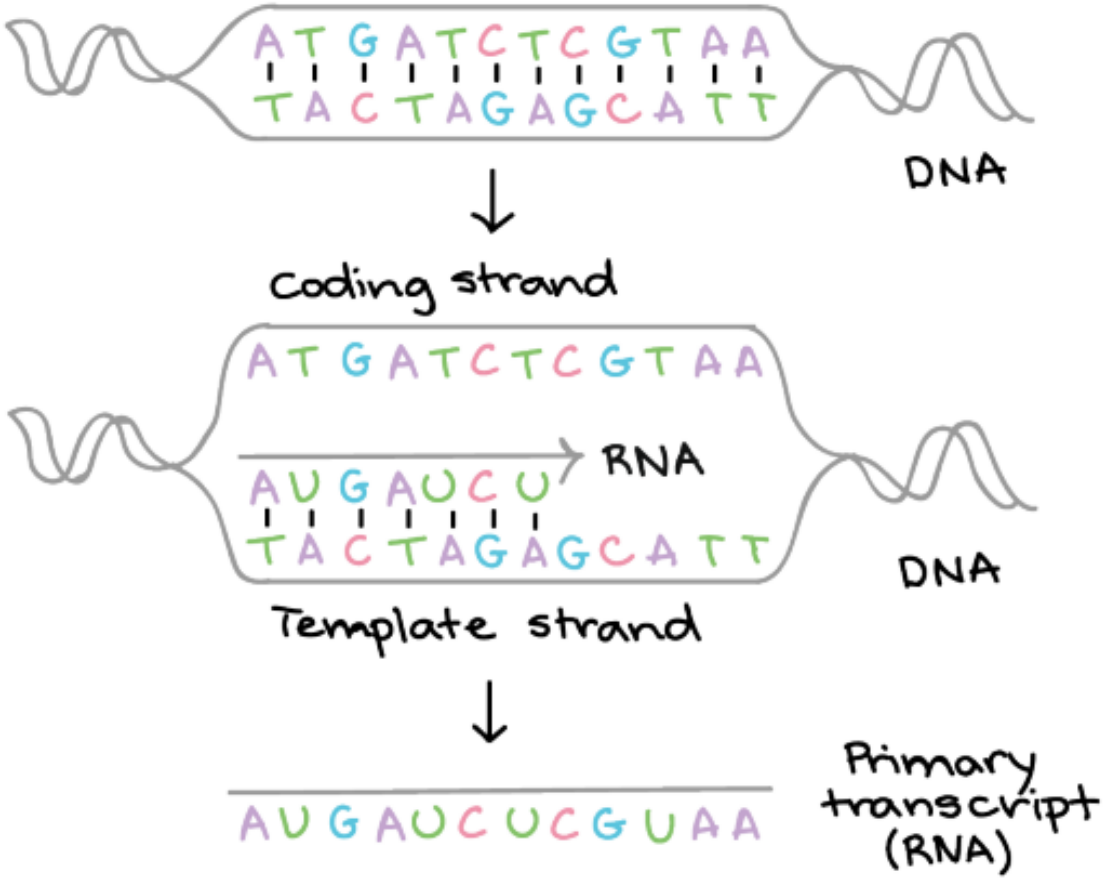
编码器-解码器架构

- 一个模型被分成两块：
  - 编码器处理输入；
  - 解码器生成输出。



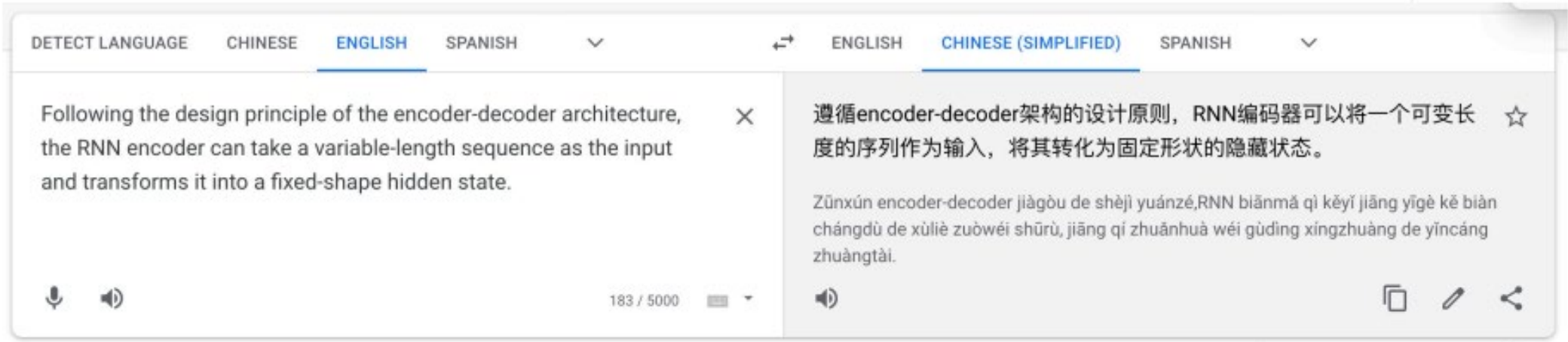
Seq2seq

从序列到序列


ATGATCTCGTAA
TACTAGAGCATT
DNA
Coding strand
ATGATCTCGTAA
RNA
AUGAUCU
TACTAGAGCATT
DNA
Template strand
AUGAUCUCGUAA
Primary transcript (RNA)

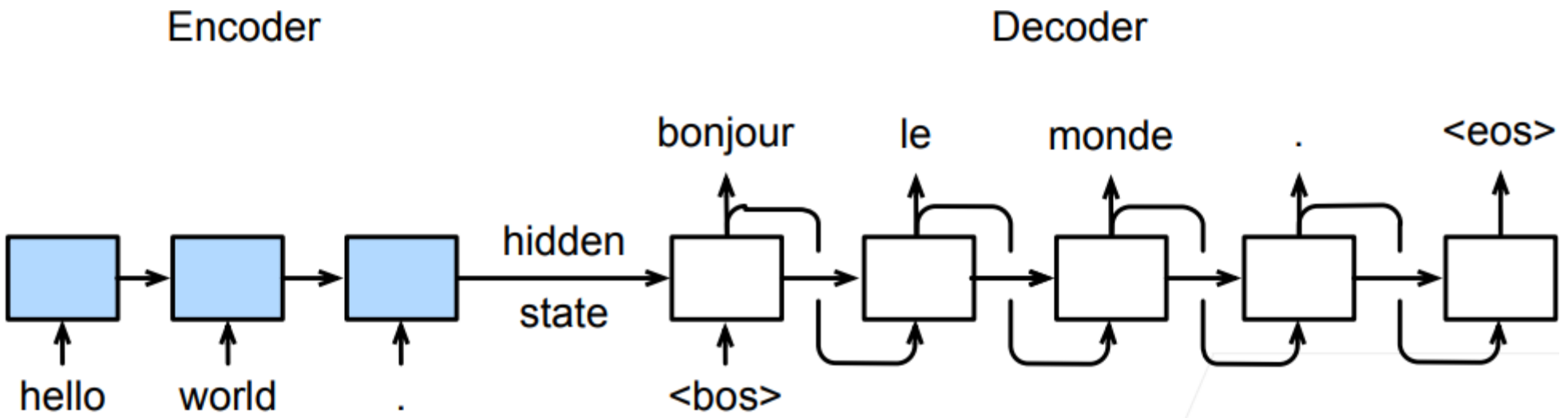
机器翻译

- 给定一个源语言的句子，自动翻译成目标语言；
- 两个句子可以有不同的长度。



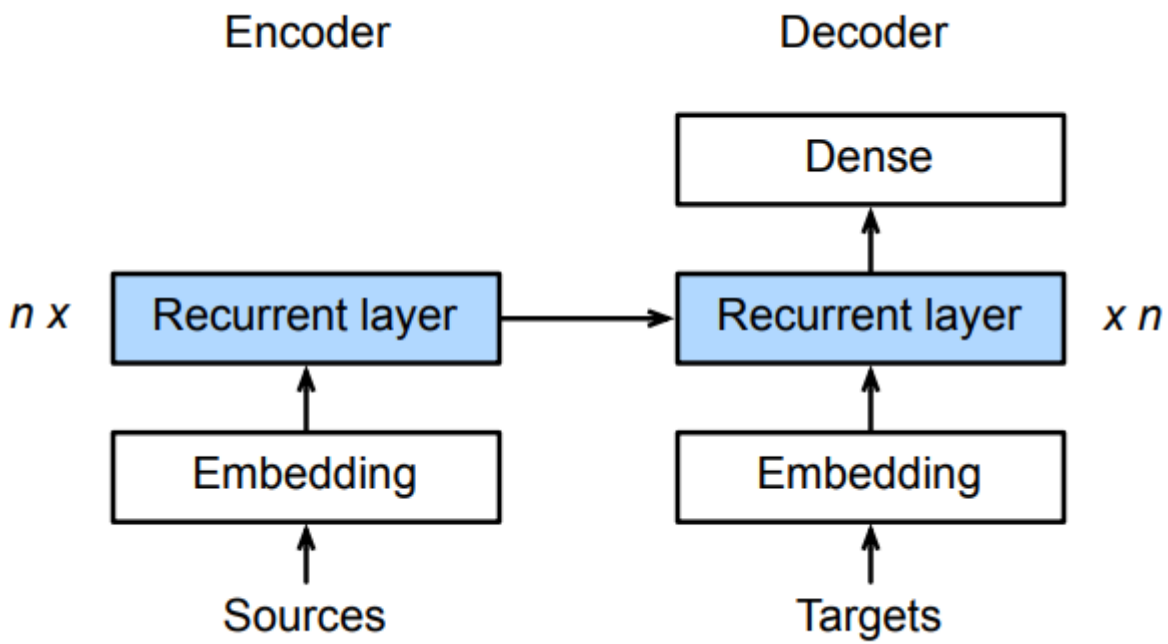
Seq2seq

- 编码器是一个RNN，读取输入的句子；
  - 此处RNN可以是双向的；
- 解码器使用另外一个RNN进行输出。



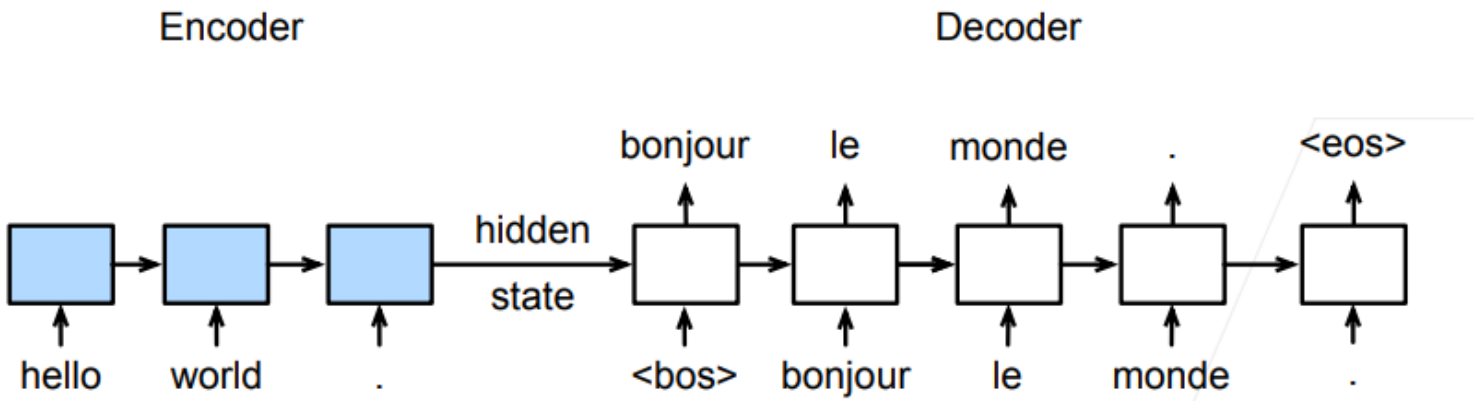
Seq2seq

- 编码器是没有输出层的RNN;
- 编码器最后时间步的隐状态用作解码器的初始隐状态。

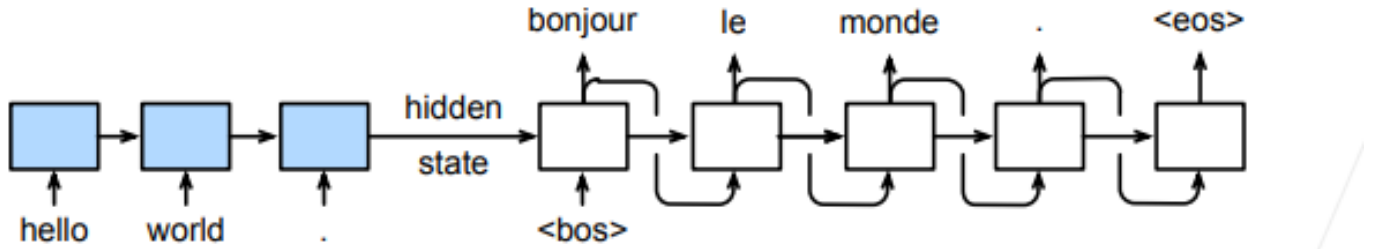


Seq2seq

- 训练阶段，解码器使用目标句子作为输入



- 推理阶段，解码器无额外输入



循环神经网络的应用

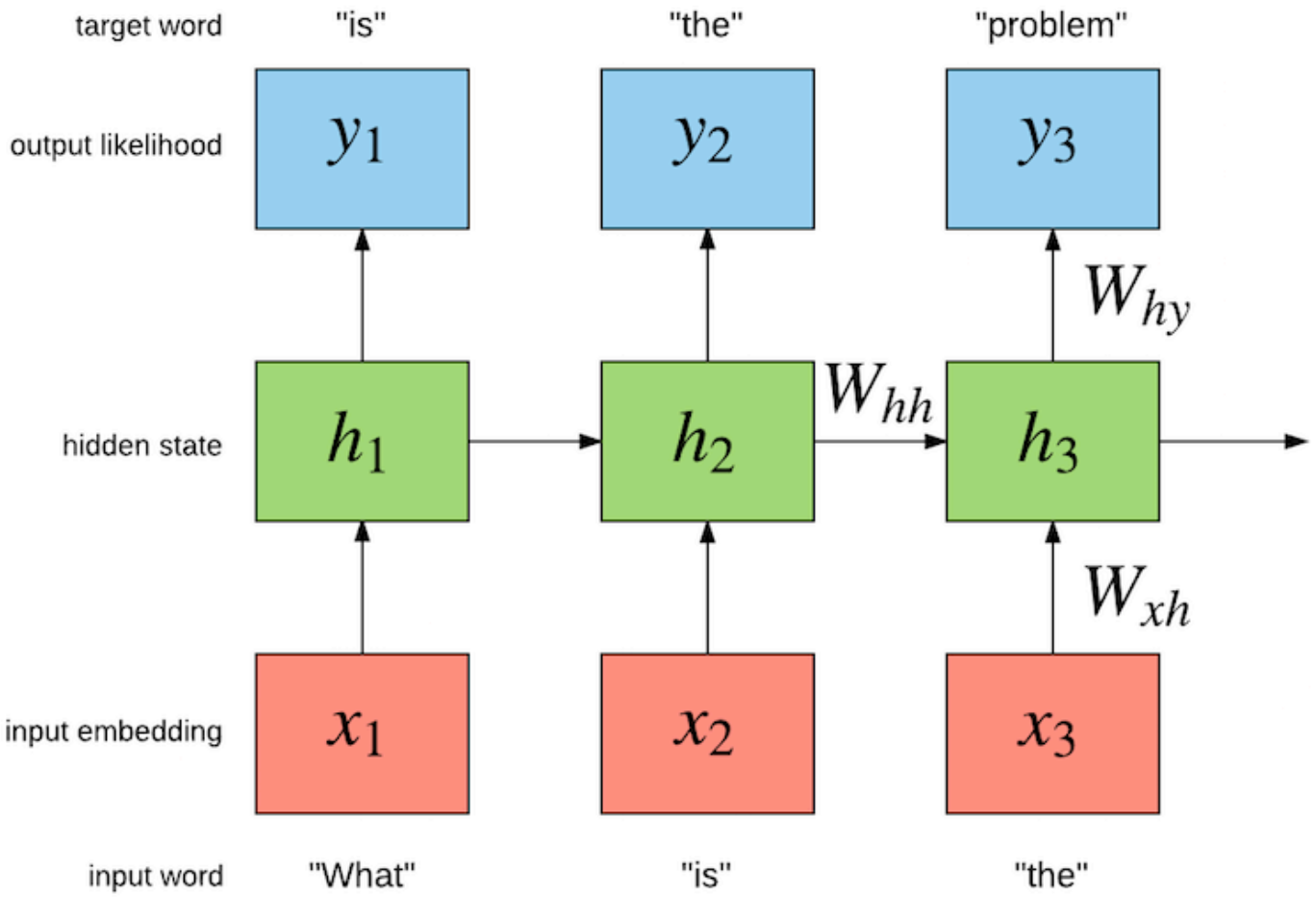
语言模型

- 自然语言理解 → 一个句子的可能性/合理性
  - ！ 在报那猫告做只 😖
  - 那只猫在作报告！ 😓
  - 那个人在作报告！ 😃

- 合理性→概率：
  - $P(x_1, x_2, \cdots, x_T)$
  - $= \prod_i P(x_i | x_{i-1}, \cdots, x_1)$
  - $\approx \prod_i P(x_i | x_{i-1}, \cdots, x_{i-n+1})$

N元语言模型

语言模型


target word
"is"
"the"
"problem"
output likelihood
y_1
y_2
y_3
W_{hy}
hidden state
h_1
h_2
h_3
W_{hh}
W_{xh}
input embedding
x_1
x_2
x_3
input word
"What"
"is"
"the"

生成LINUX内核代码

```
/*
 * If this error is set, we will need anything right after that BSD.
 */
static void action_new_function(struct s_stat_info *wb)
{
  unsigned long flags;
  int lel_idx_bit = e->edd, *sys & ~((unsigned long) *FIRST_COMPAT);
  buf[0] = 0xFFFFFFFF & (bit << 4);
  min(inc, slist->bytes);
  printk(KERN_WARNING "Memory allocated %02x/%02x, "
    "original MLL instead\n"),
    min(min(multi_run - s->len, max) * num_data_in),
    frame_pos, sz + first_seg);
  div_u64_w(val, inb_p);
  spin_unlock(&disk->queue_lock);
  mutex_unlock(&s->sock->mutex);
  mutex_unlock(&func->mutex);
  return disassemble(info->pending_bh);
}

static void num_serial_settings(struct tty_struct *tty)
{
  if (tty == tty)
    disable_single_st_p(dev);
  pci_disable_spool(port);
```

作词机

- RNN在“学习”过汪峰全部作品后自动生成的歌词
  - 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叶
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
可我们这是我们的是不要为你
我们想这有一种生活的时候

作诗

白鹭窥鱼立， Egrets stood, peeping fishes. 青山照水开. Water was still, reflecting mountains. 夜来风不动， The wind went down by nightfall, 明月见楼台. as the moon came up by the tower.	满怀风月一枝春， Budding branches are full of romance. 未见梅花亦可人. Plum blossoms are invisible but adorable. 不为东风无此客， With the east wind comes Spring. 世间何处是前身. Where on earth do I come from?

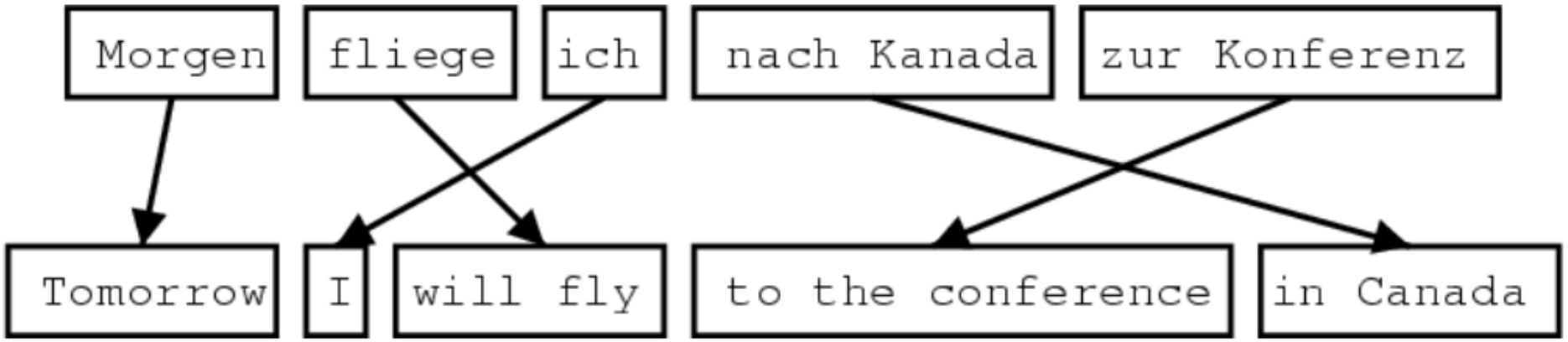
案例

Suno AI

火山写作 (writingo.net)

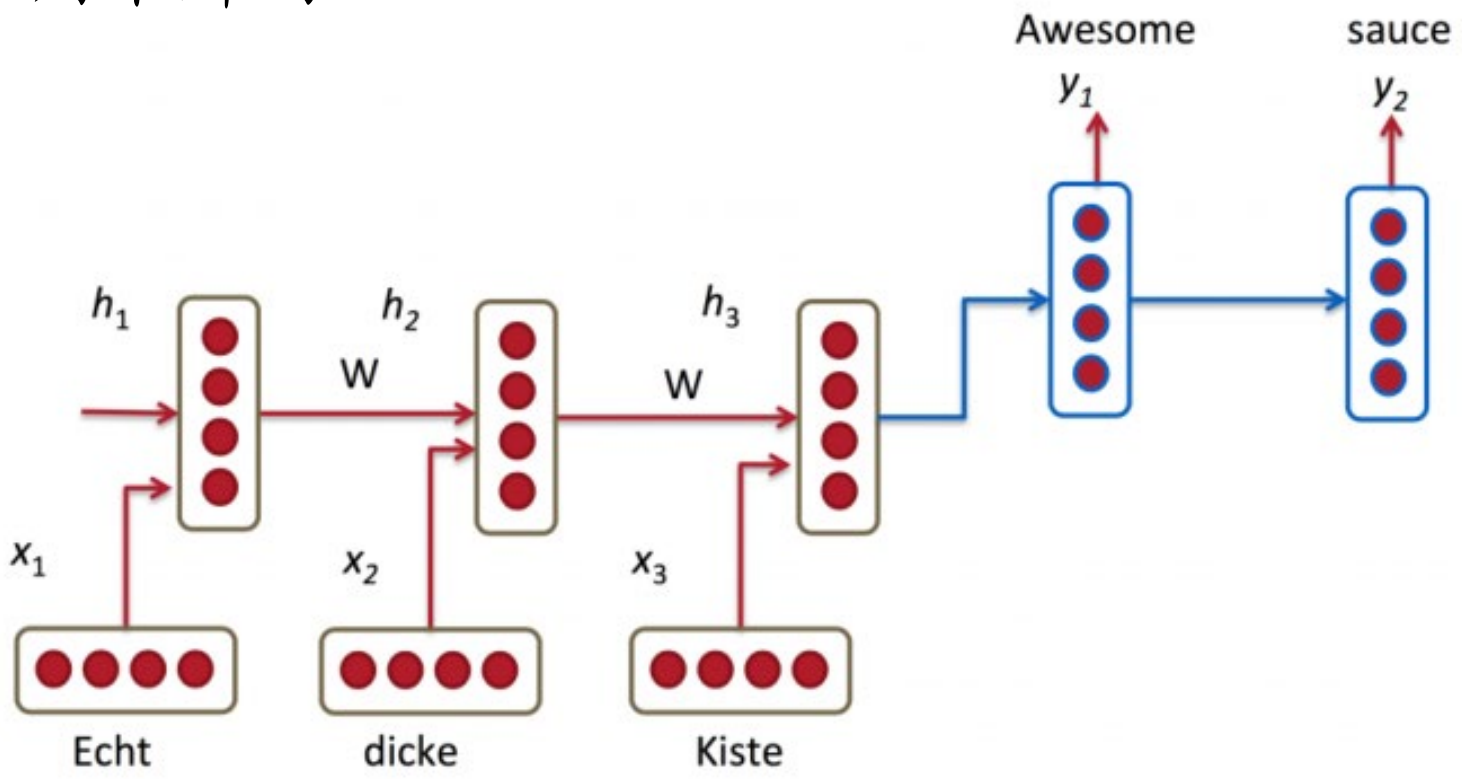
传统统计机器翻译

- 源语言：f
- 目标语言：e
  - 模型：$\hat{e} = \text{argmax}_e\, p(e|f) = \text{argmax}_e\, p(f|e)p(e)$
  - p(f|e): 翻译模型
  - p(e) : 语言模型

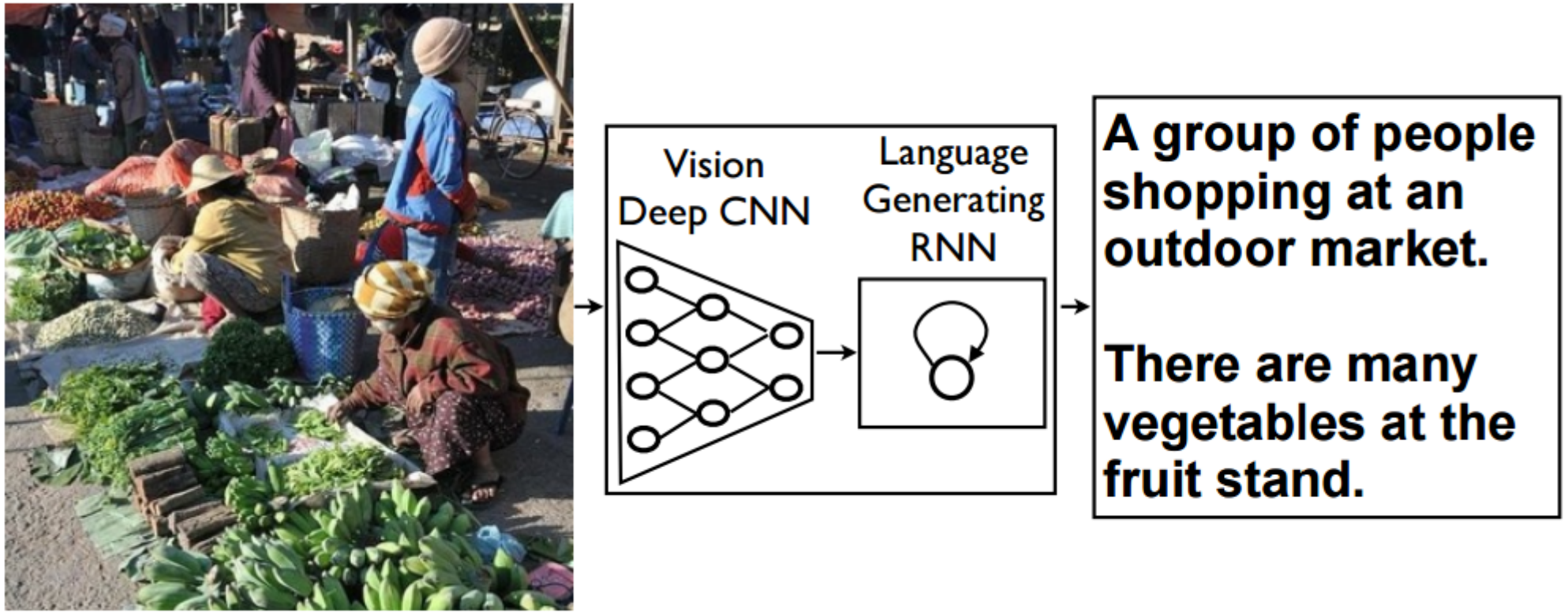


基于序列到序列的机器翻译

- 编码器-解码器结构
  - 一个RNN用来编码
  - 另一个RNN用来解码



看图说话


Vision
Deep CNN
Language
Generating
RNN
A group of people shopping at an outdoor market.
There are many vegetables at the fruit stand.

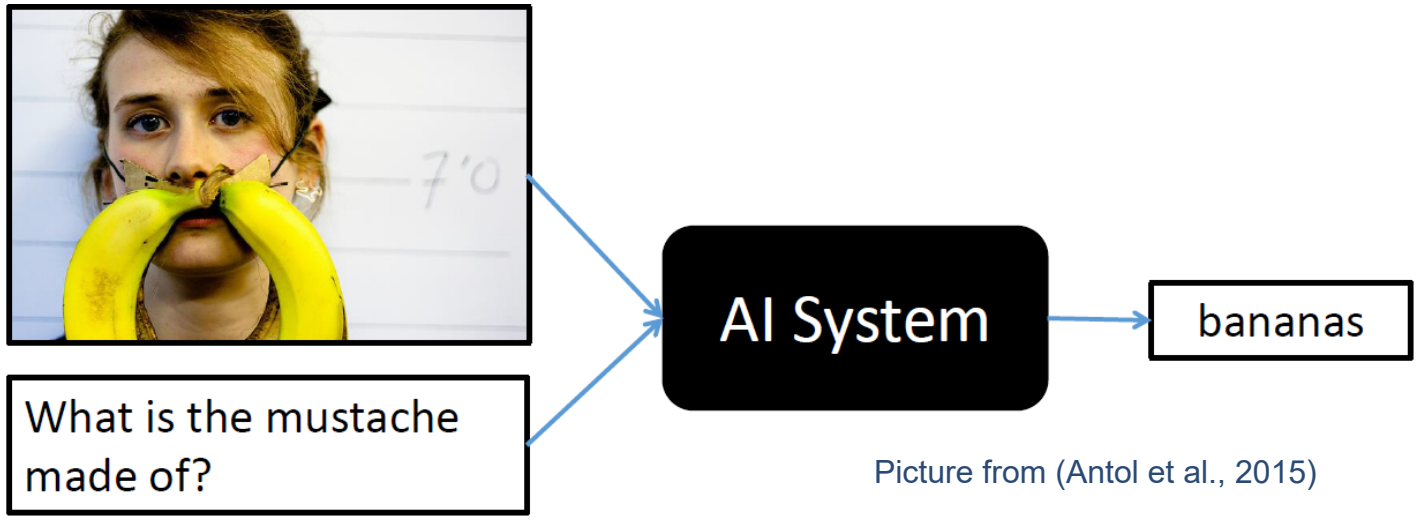
看图说话

Figure 5. A selection of evaluation results, grouped by human rating.

Visual Question Answering (VQA)

Demo Website

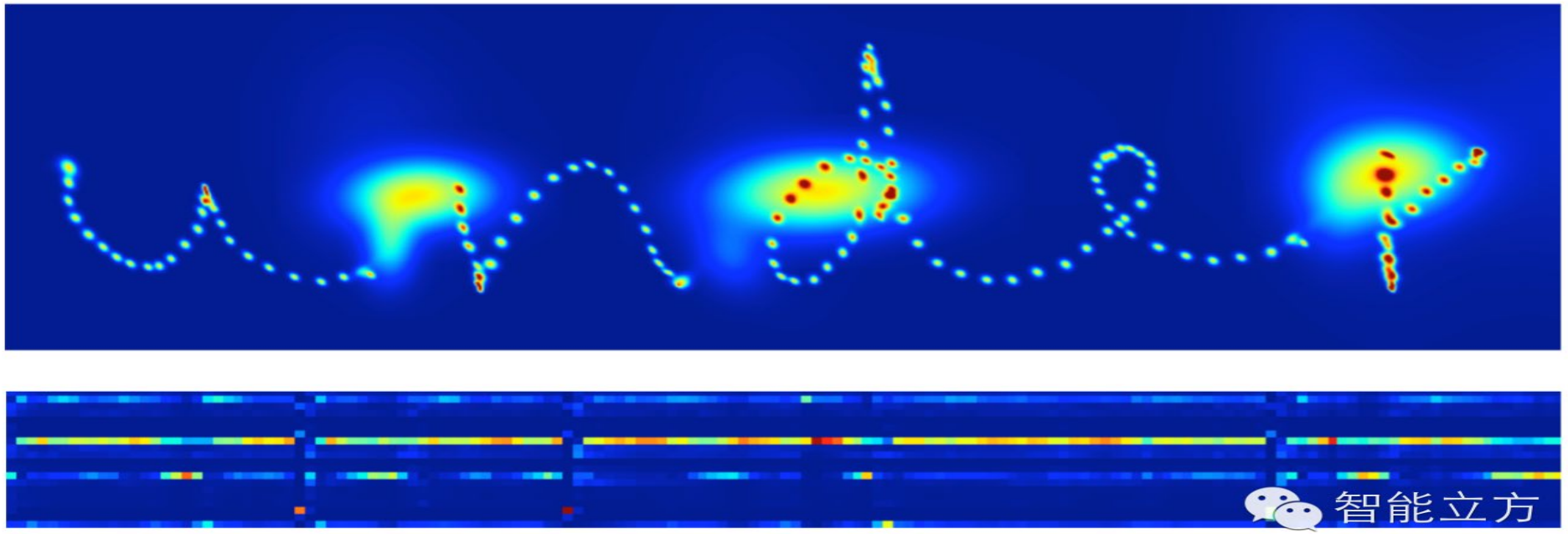
VQA: Given an image and a natural language question about the image, the task is to provide an accurate natural language answer



Picture from (Antol et al., 2015)

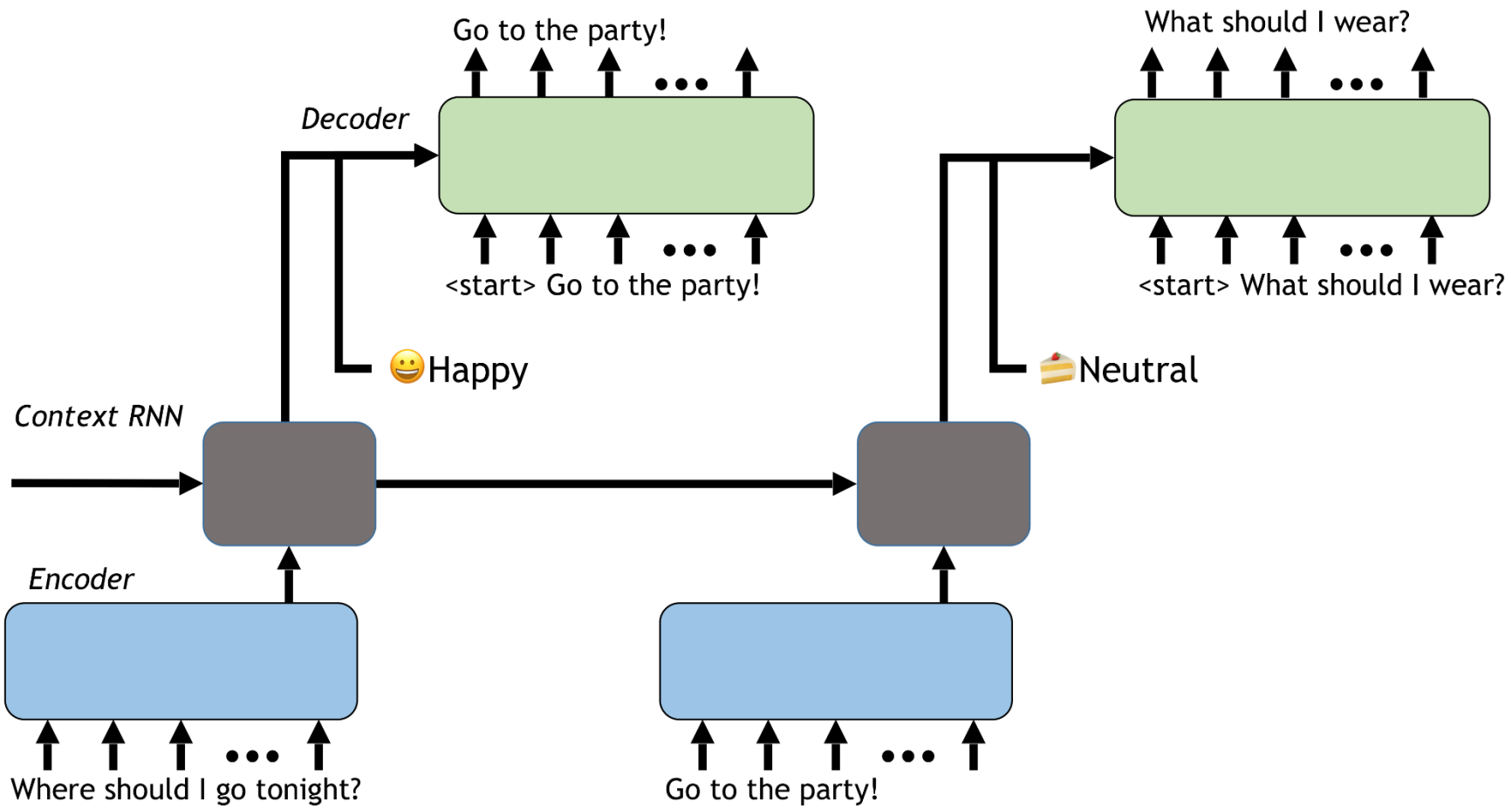
写字

- 把一个字母的书写轨迹看作是一连串的点。一个字母的“写法”其实是每一个点相对于前一个点的偏移量，记为(offset x, offset y)。再增加一维取值为0或1来记录是否应该“提笔”。



对话系统

https://github.com/lukalabs/cakechat

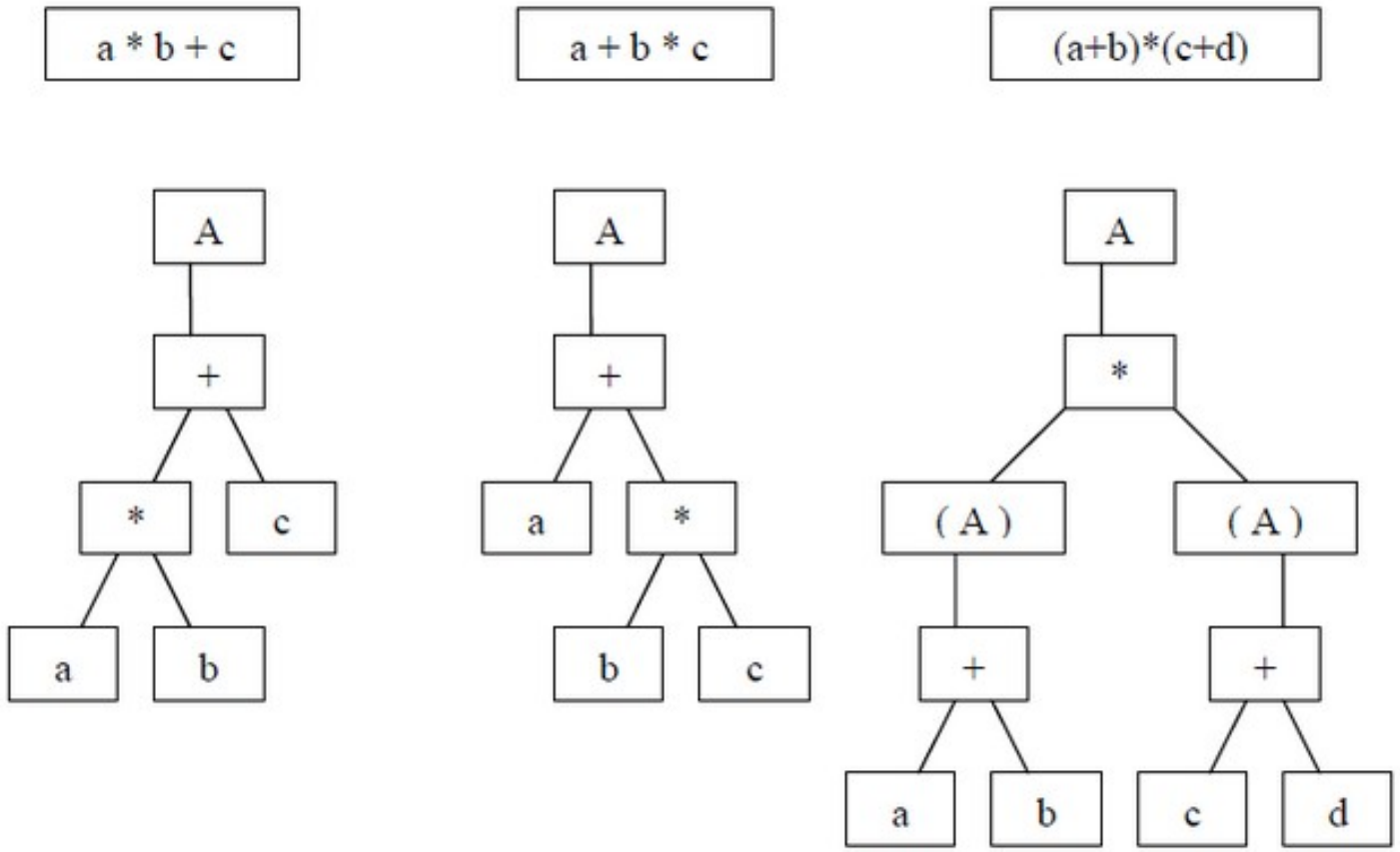


扩展到图结构

扩展到图结构

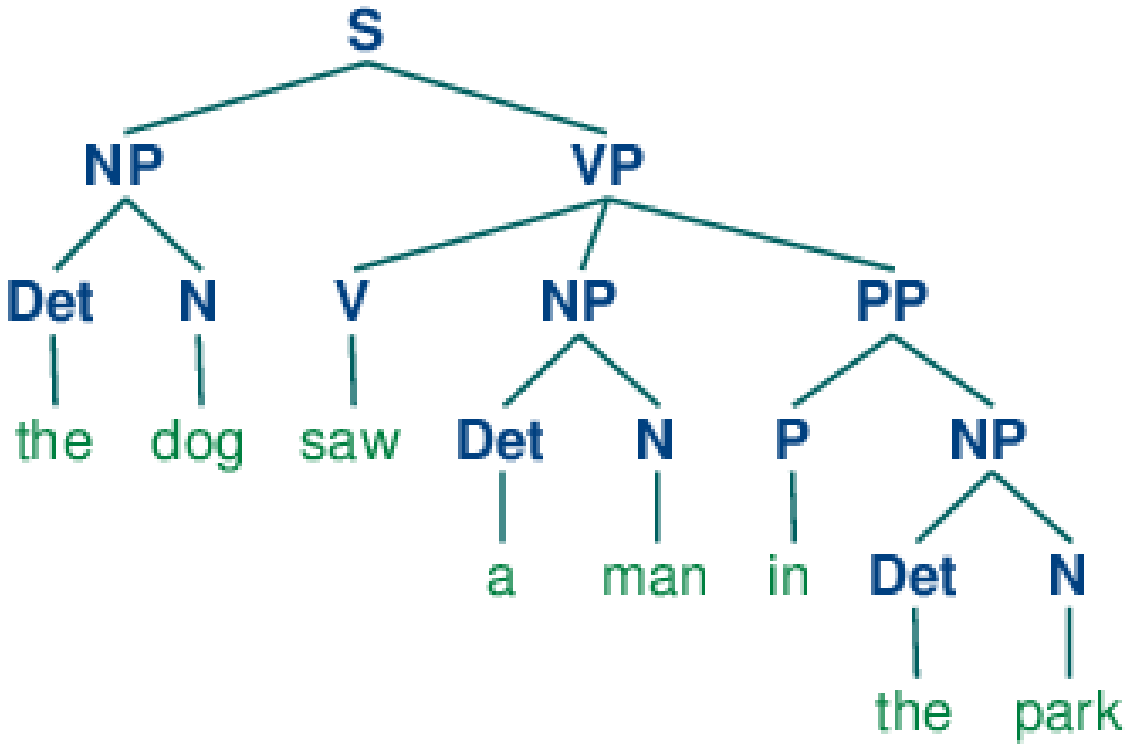
树结构

- 程序语言的句法结构



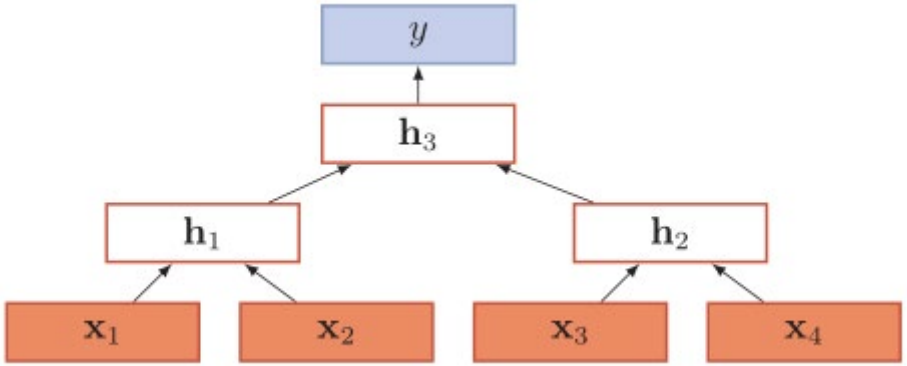
树结构

- 自然语言的句法结构



递归神经网络 Recursive Neural Network

- 递归神经网络：在一个有向无环图上共享一个组合函数



$$\mathbf{h}_i = f(\mathbf{h}_{\pi_i})$$

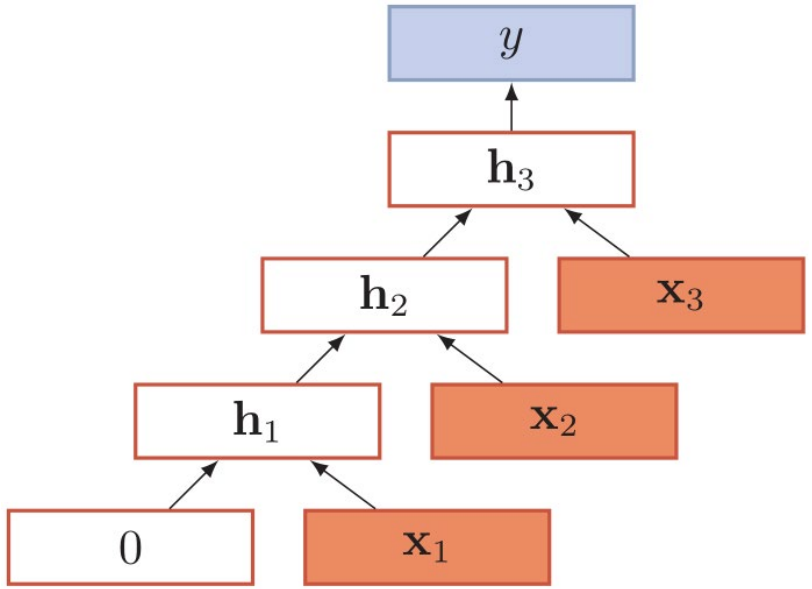
$$\mathbf{h}_1 = f(W \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} + \mathbf{b}),$$

$$\mathbf{h}_2 = f(W \begin{bmatrix} \mathbf{x}_3 \\ \mathbf{x}_4 \end{bmatrix} + \mathbf{b}),$$

$$\mathbf{h}_3 = f(W \begin{bmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \end{bmatrix} + \mathbf{b}),$$

递归神经网络

- 退化为循环神经网络



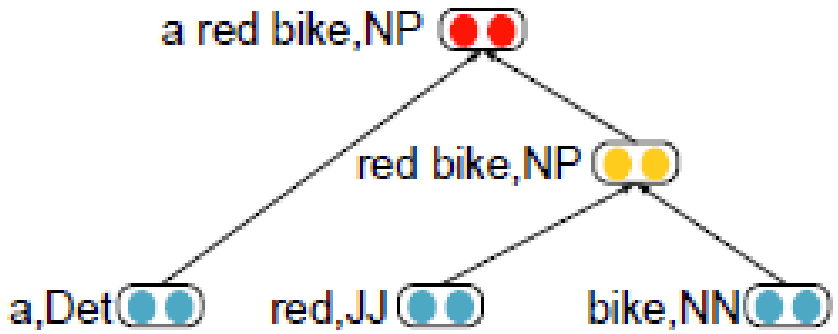
$$\mathbf{h}_t = f(\mathbf{h}_{t-1}, \mathbf{x}_t),$$

递归神经网络

给定一个语法树，

$p_2 \rightarrow ap_1$,

$p_1 \rightarrow bc$.



$$\mathbf{p}_1 = f(\mathbf{W}\begin{bmatrix}\mathbf{b}\\ \mathbf{c}\end{bmatrix}),$$

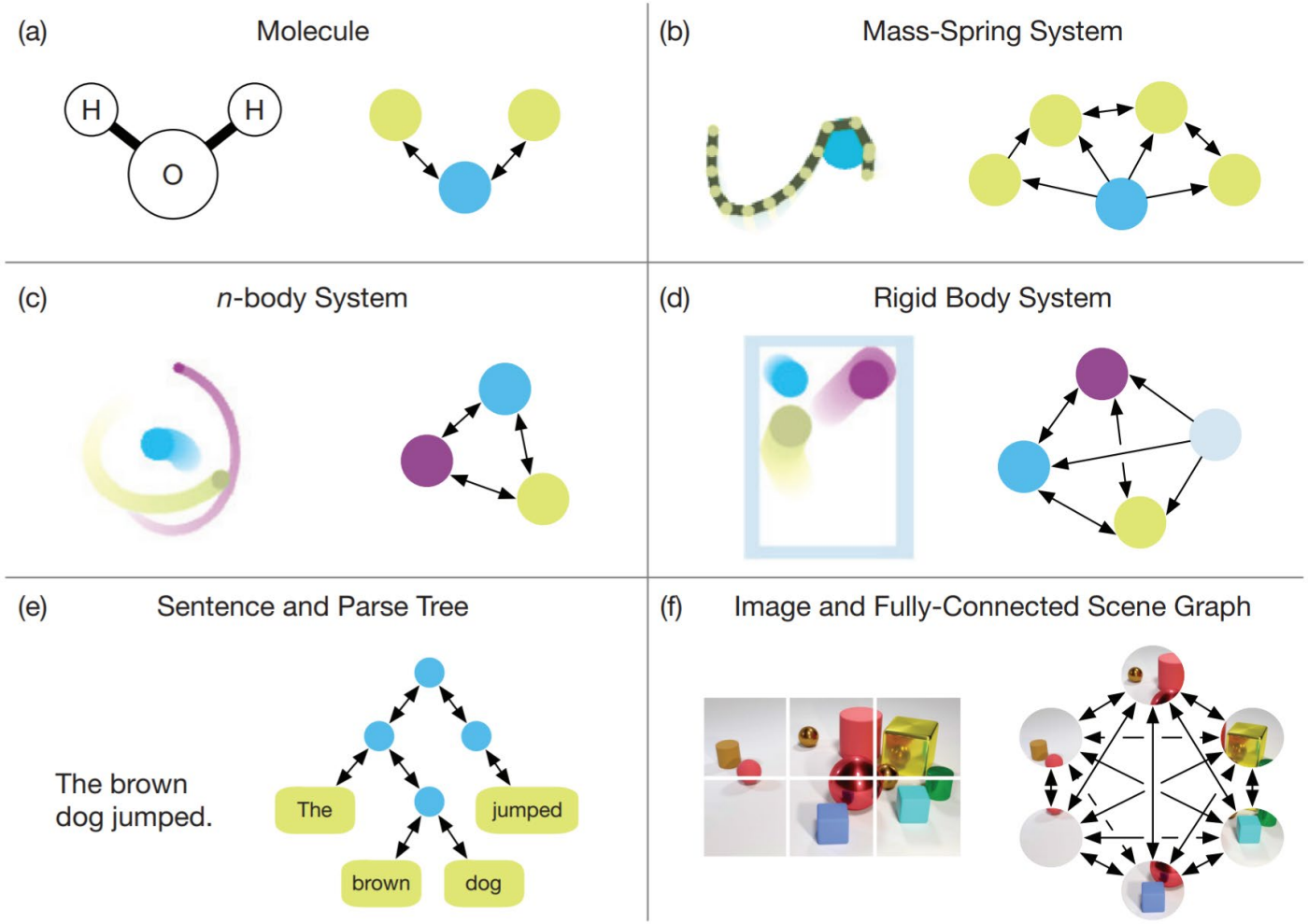
$$\mathbf{p}_2 = f(\mathbf{W}\begin{bmatrix}\mathbf{a}\\ \mathbf{p}_1\end{bmatrix}).$$

图网络

- 在实际应用中，很多数据是图结构的，比如知识图谱、社交网络、分子网络等。而前馈网络和循环网络很难处理图结构的数据。

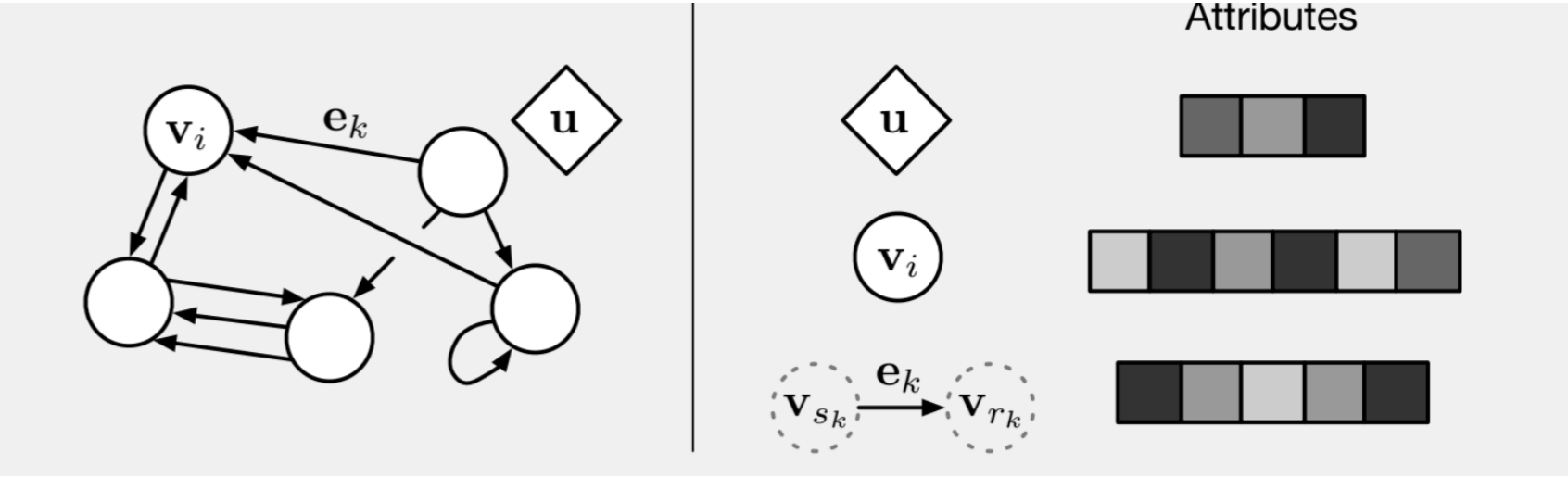
图数据

https://arxiv.org/pdf/1806.01261.pdf



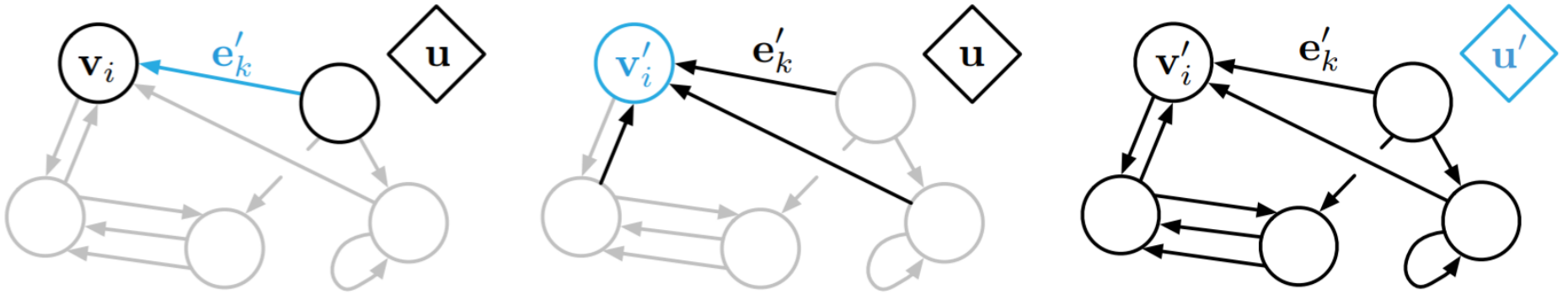
图网络

https://arxiv.org/pdf/1806.01261.pdf



图网络

https://arxiv.org/pdf/1806.01261.pdf



(a) Edge update (b) Node update (c) Global update

图网络

- 对于一个任意的图结构G(V,E)
  - 更新函数

$$\mathbf{m}_t^{(v)} = \sum_{u \in N(v)} f\left(\mathbf{h}_{t-1}^{(v)}, \mathbf{h}_{t-1}^{(u)}, \mathbf{e}^{(u,v)}\right)$$

$$\mathbf{h}_t^{(v)} = g\left(\mathbf{h}_{t-1}^{(v)}, \mathbf{m}_t^{(v)}\right)$$

  - 读出函数

$$\mathbf{y}_t = g\left(\{\mathbf{h}_T^{(v)} | v \in \mathcal{V}\}\right)$$

PyTorch实现循环神经网络

谢 谢

https://nndl.github.io/