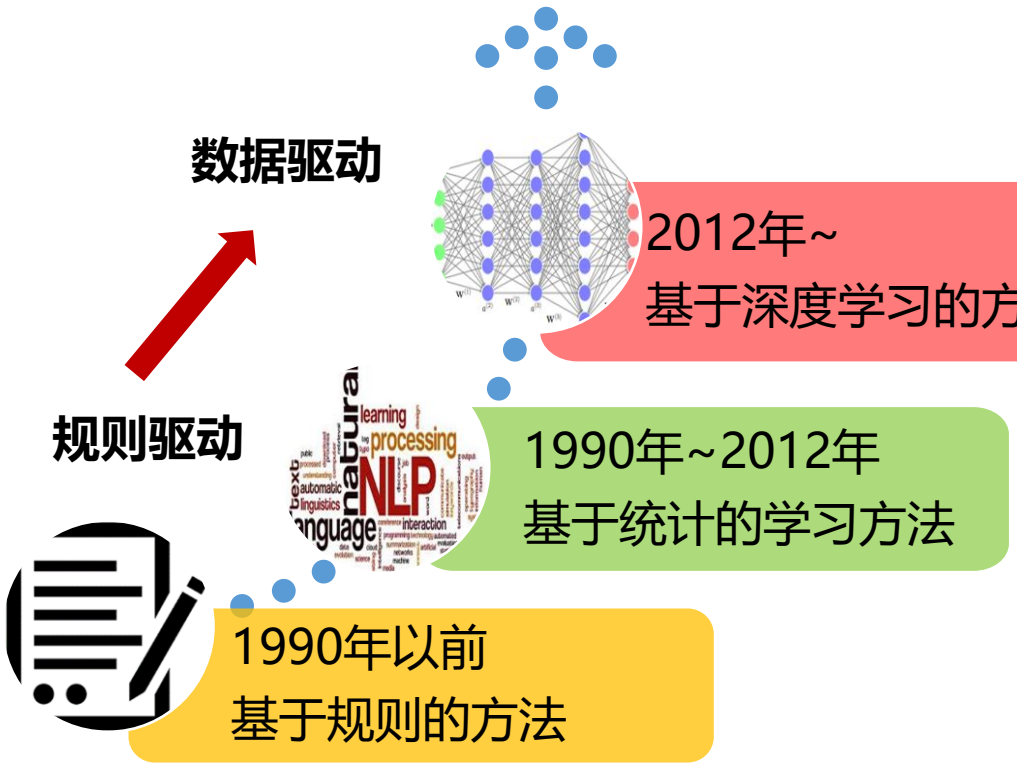
中国科学院大学
University of Chinese Academy of Sciences

循环神经网络

AI DISCOVERY

什么是自然语言处理

自然语言处理研究实现人与计算机之间用自然语言进行有效通信的各种理论和方法。自然语言处理技术发展经历了**基于规则的方法**、**基于统计学习的方法**和**基于深度学习的方法**三个阶段。自然语言处理由浅入深的四个层面分别是**形式**、**语义**、**推理**和**语用**，当前正处于由语义向推理的发展阶段。



自动分词

AI DISCOVERY

中文为什么要进行分词?

与大部分印欧语系的语言不同，汉语是以字为基本的书写单位，词语之间没有明显的区分标记，需要人为切分。

例子：我路过南京市长江大桥

我 / 路过 / 南京 / 市 / 长江 / 大桥

我 / 路过 / 南京市 / 长江大桥

} 不同的分词粒度

中文分词的核心任务是要确定词边界，将句子分解为最小意义单元，即将中文字序列转换为词序列。

中文分词是很多自然语言处理系统中的基础模块和首要环节。

AI DISCOVERY

中文分词

AI DISCOVERY

分词面临的主要问题

汉语分词困难重重

① **分词规范：**易受主观语感约束，产生不同的切分结果

② **歧义切分**

有些歧义无法在句子内部解决，
需要结合篇章上下文

- ✓ 交集歧义

  研究 / 生命 / 的 / 起源　　研究生 / 命 / 的 / 起源

- ✓ 组合歧义

  门 / 把 / 手 / 弄 / 坏 / 了　　门 / 把手 / 弄 / 坏 / 了

③ **未登录词识别**

包括中外人名、中国地名、机构组织名、事件名、货币名、缩略语、派生词、各种专业术语以及在不断发展和约定俗成的一些新词语。

确定词汇边界：PMI互信息，熵等

确定新词语义：领域词扩展等，LDA，word2vec

AI DISCOVERY

分词算法

AI DISCOVERY

基于规则的分词方法

简单易行，但歧义消解能力差

- **基本思想**：按照一定策略将待分析的汉字串与一个“充分大的”机器词典中的词条进行匹配
- **主要方法**：最大匹配、逆向最大匹配、双向最佳匹配、逐词遍历

基于统计的分词方法

效果依赖于训练语料的规模和质量

- **基本思想**：上下文中相邻的字同时出现的次数越多，就越有可能构成一个词，字与字相邻出现的概率或频率能较好地反映成词的可信度。
- **主要方法**：N 元文法模型（N-gram）、隐马尔可夫模型（Hiden Markov Model，HMM）、最大熵模型（ME）、条件随机场模型（Conditional Random Fields，CRF）等

基于理解的分词方法

需要大量的语言知识和信息

- **基本思想**：在分词的同时进行句法、语义分析，利用句法信息和语义信息来处理歧义现象，让计算机模拟人对句子的理解来进行分词。
- **主要方法**：专家系统分词、神经网络分词（LSTM,CNN）

AI DISCOVERY

文本分类与聚类

AI DISCOVERY

随着互联网的高速发展，海量文本数据不断产生，如何面对浩如烟海的数据进行分类、组织和管理，已经成为一个具有重要用途的研究课题，广受学术界和工业界关注。

文本分类(Text Classification)根据给定文档的内容或主题，自动分配预先定义的类别标签。

文本聚类(Text Clustering)根据文档之间的内容或主题相似度，将文档集合划分成若干个子集，每个子集内部的文档相似度较高，而子集之间的相似度较低。

应用场景：新闻自动分类、电子商务评价分类 、垃圾邮件识别等。

文本分类与聚类

AI DISCOVERY

文本分类模型

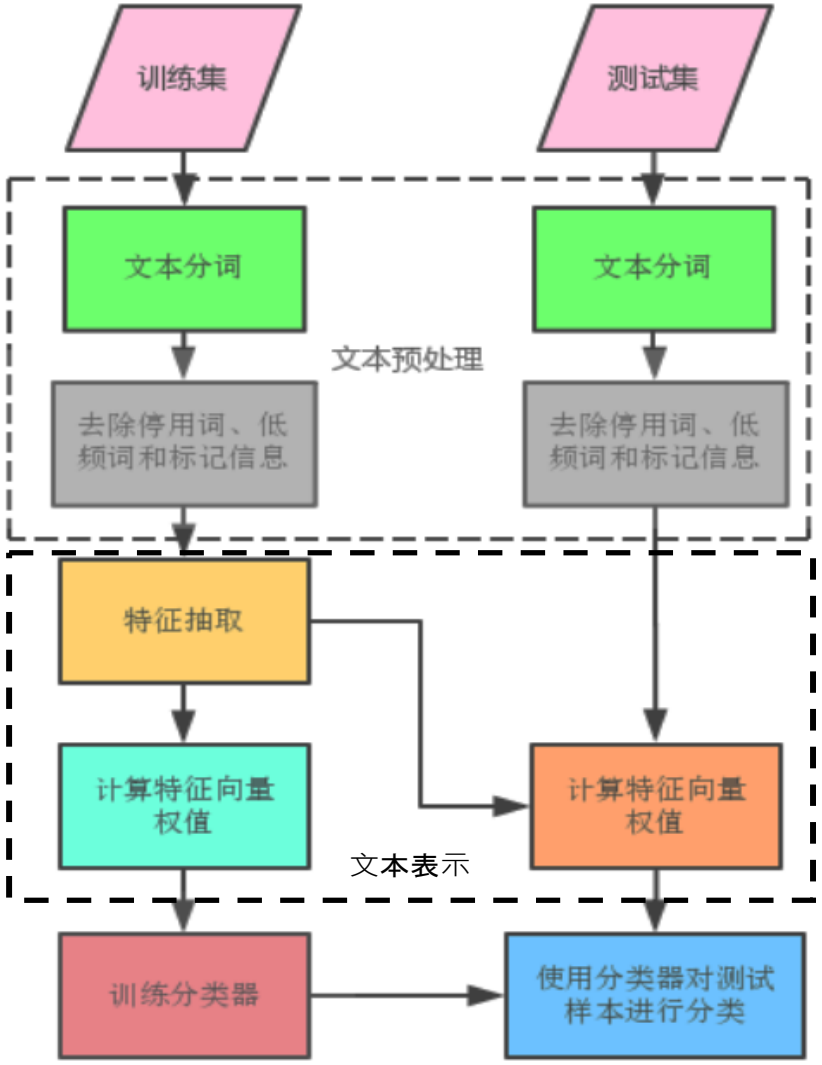
基于机器学习的分类：朴素贝叶斯 (Naive Bayes)、支持向量机(SVM)、最大熵分类器

基于神经网络的方法：多层感知机(MLP) 、卷积神经网络(CNN) 、循环神经网络(RNN)

文本聚类模型

基于距离的聚类：通过相似度函数计算语义关联度，然后根据语义关联度进行聚类，如K-means

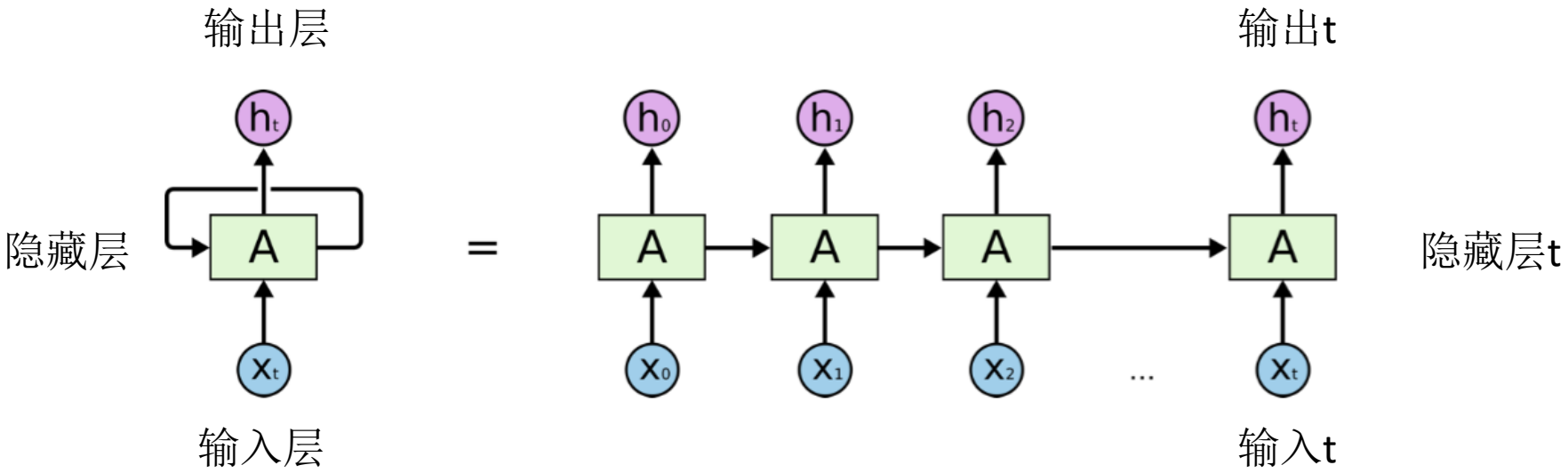
基于概率模型的聚类：假设每篇文章是所有主题上的概率分布，典型的主题模型包括 PLSA 和 LDA 等



图为有监督分类方法的一般过程，无监督分类/聚类方法将训练部分去掉即可

循环神经网络

◆**循环神经网络（Recurrent Neural Network，RNN），**也叫递归神经网络。这里为了区别与另外一种**递归神经网络（Recursive Neural Network），**我们称为**循环神经网络。**



✓循环神经网络通过使用带自反馈（隐藏层）的神经元，能够处理任意长度的序列。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。已经被广泛应用在语音识别、图像处理、语言模型以及自然语言生成等任务上。

简单循环神经网络

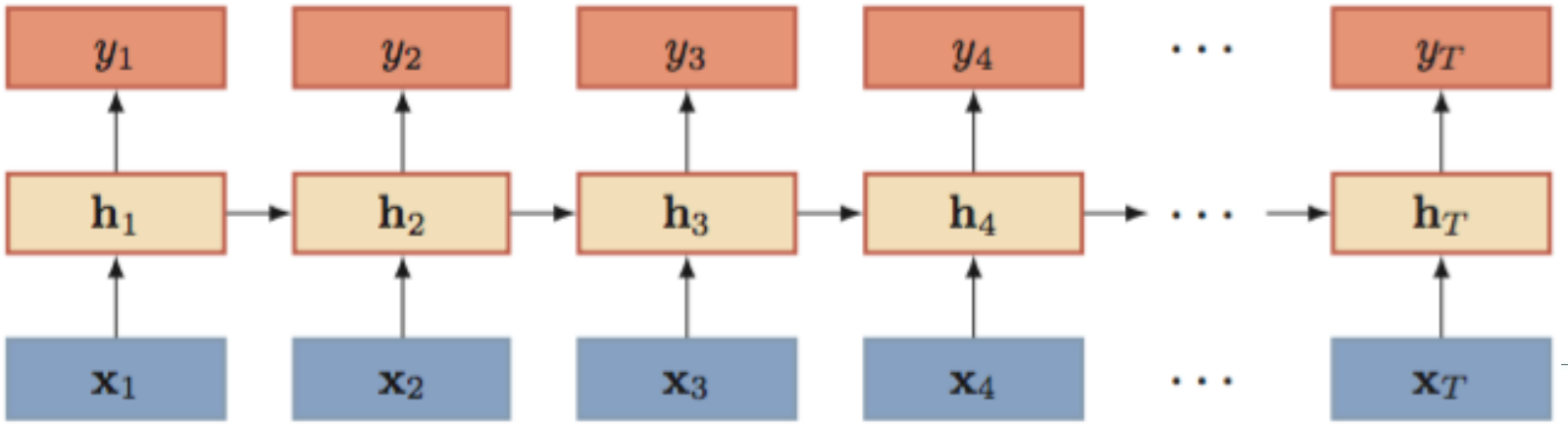
AI DISCOVERY

✓假设时刻 t 时，输入为 $\mathbf{x_t}$，隐层状态（隐层神经元活性）为 $\mathbf{h_t}$。 $\mathbf{h_t}$ 不仅和当前时刻的输入相关，也和上一个时刻的隐层状态相关。

✓一般我们使用如下函数：

$$\boldsymbol{h}_t = f(\boldsymbol{U}\boldsymbol{h}_{t-1} + \boldsymbol{W}\boldsymbol{x}_t + b) \qquad \boldsymbol{y}_t = softmax(\boldsymbol{W}^{(S)}\boldsymbol{h}_t)$$

✓ 这里，f 是非线性函数，通常为 *sigmod*函数或 *tanh* 函数。



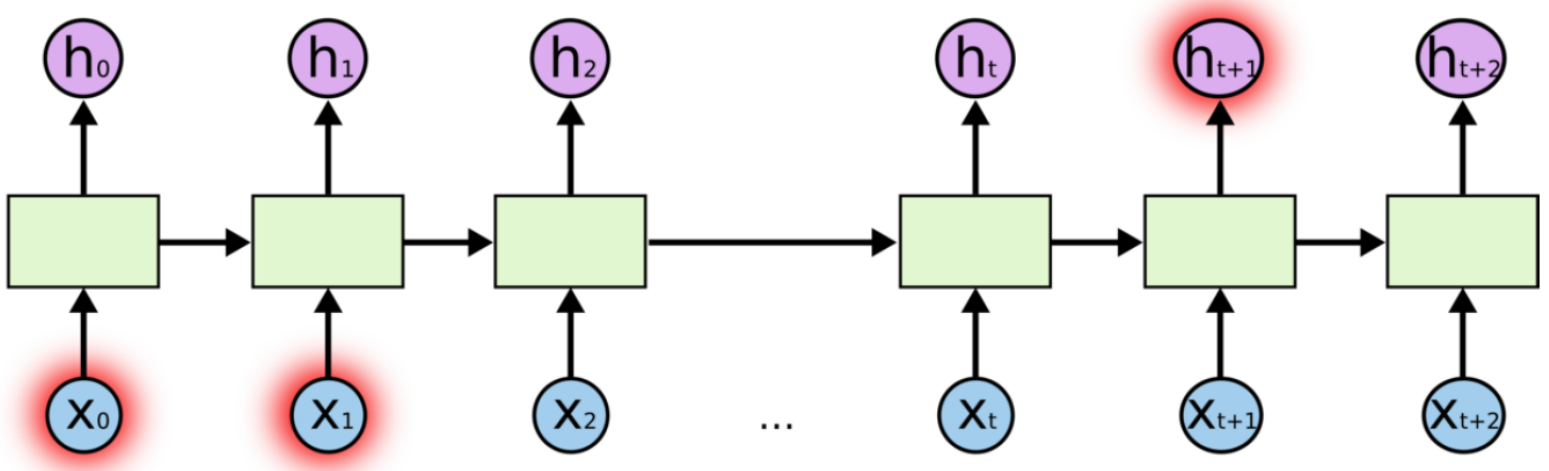
AI DISCOVERY

长期依赖的问题

AI DISCOVERY

- 很久以前的输入，对当前时刻的网络影响较小；反向传播的梯度，也很难影响很久以前的输入

- 例如：
  - The cat, which already ate a bunch of food, (was) full.
  - The cats, which already ate a bunch of food, (were) full.



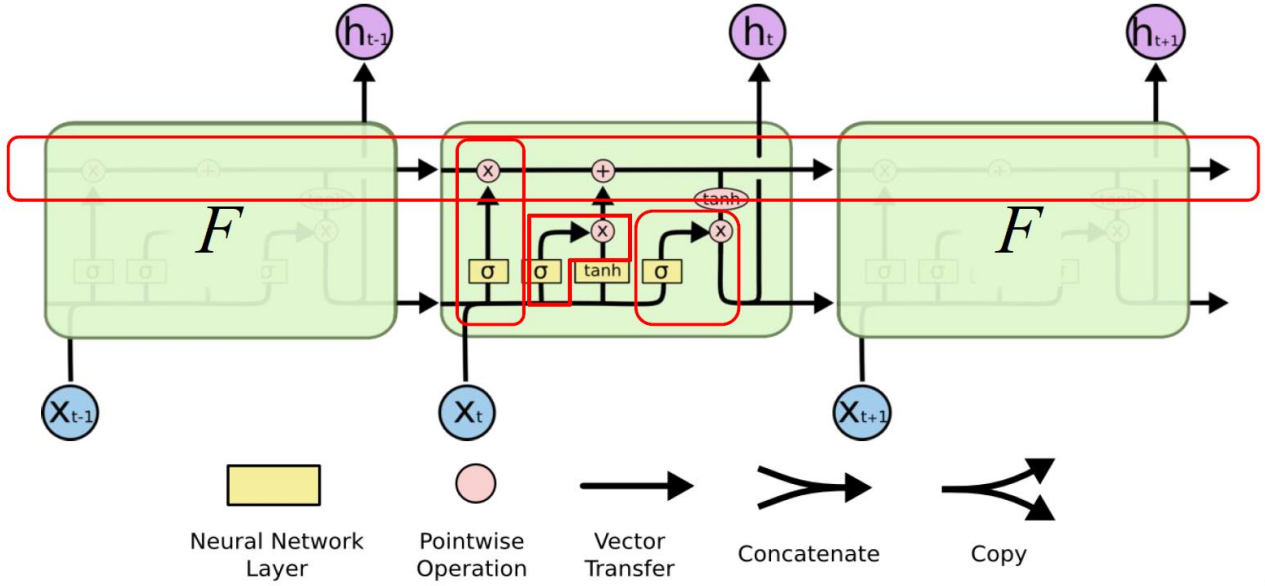
AI DISCOVERY

- 解决思路：采用ReLu函数，或采用其他模型来代替非线性激活函数

改进方案：长短时记忆神经网络LSTM

AI DISCOVERY

长短时记忆神经网络（Long Short-Term Memory Neural Network，LSTM）是循环神经网络的一个变体，可以有效地解决**长期依赖问题/梯度消失**问题。



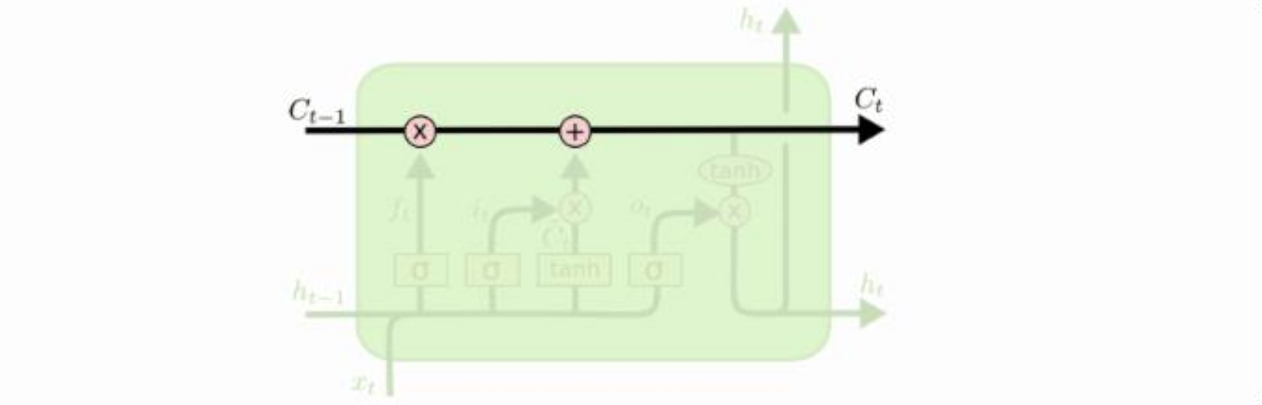
LSTM 模型的关键是引入了一组**记忆单元**（Memory Units），允许网络可以学习何时遗忘历史信息，何时用新信息更新记忆单元。在时刻 t 时，记忆单元 $\mathbf{c}_t$ 记录了到当前时刻为止的所有历史信息，并受**三个“门”控制**：输入门 $\mathbf{i}_t$，遗忘门$\mathbf{f}_t$和输出门$\mathbf{o}_t$。三个门的元素的值在 [0, 1] 之间。

AI DISCOVERY

LSTM

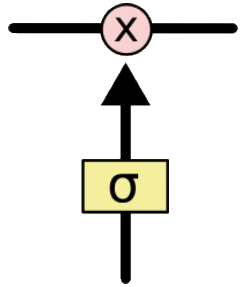
AI DISCOVERY

◆核心：记忆（细胞状态）和门机制



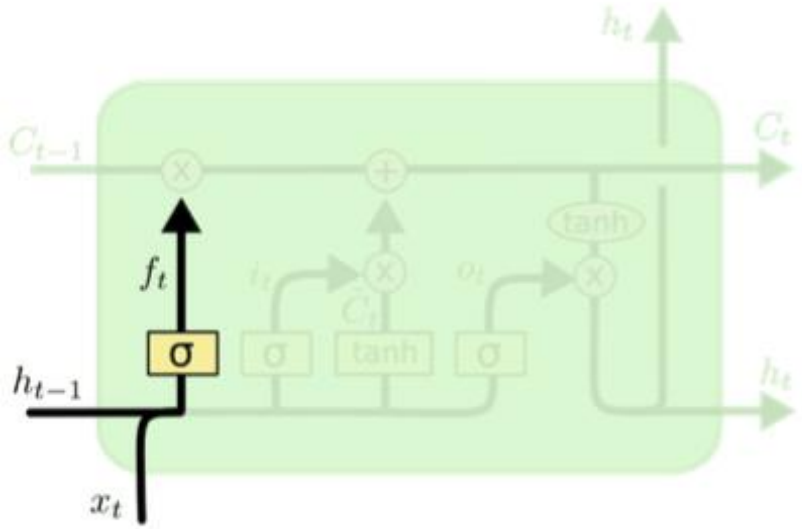
细胞的状态在整条链上运行，只有一些小的线性操作作用其上，信息很容易保持不变的流过整条链。

门(Gate)是一种可选地让信息通过的方式。 它由一个Sigmoid神经网络层和一个点乘法运算组成。



Sigmoid神经网络层输出0和1之间的数字，这个数字描述每个组件有多少信息可以通过， 0表示不通过任何信息，1表示全部通过

Forget Gate



以语言模型为例，细胞状态可能包括当前主语的性别，从而决定使用正确的代词（它/他/她），当看到一个新主语时，需要忘记旧主语的性别。

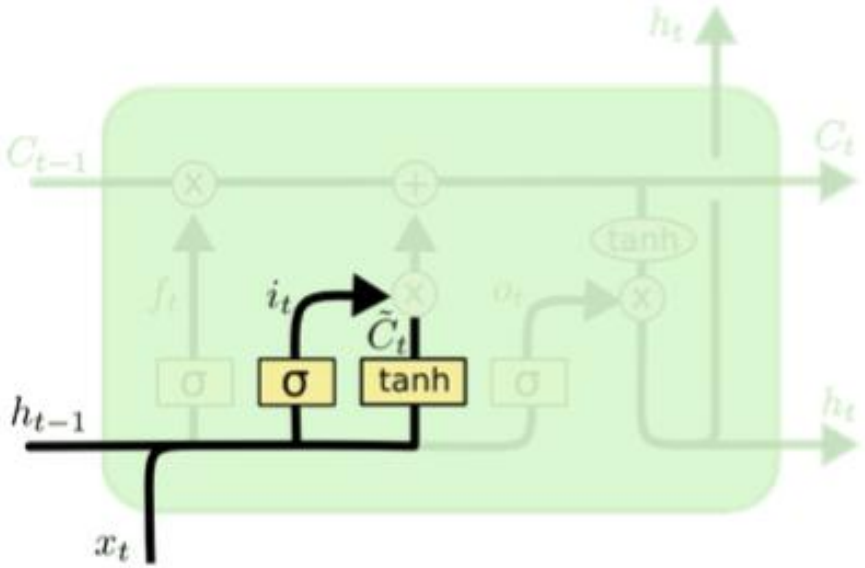
$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

- ✓ 遗忘门决定我们要从细胞状态中丢弃什么信息
- ✓ 它查看h_{t-1}(前一个隐藏状态)和x_t(当前输入)，并为状态C_{t-1}(上一个状态)中的每个数字输出0和1之间的数字
- ✓ 1代表完全保留，而0代表彻底删除

Input Gate

AI DISCOVERY

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$
$$\tilde{C}_t = tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

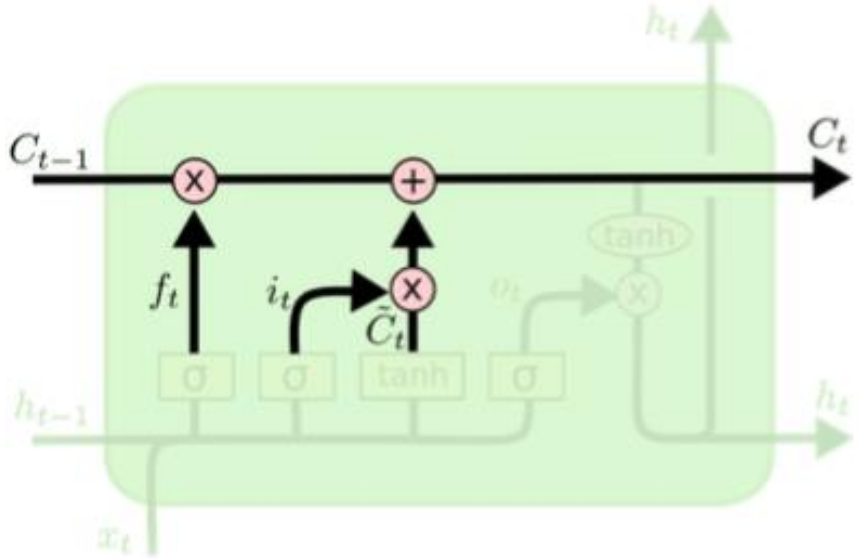


输入门决定我们要在细胞状态中存储什么信息

- ✓ 首先，输入门的Sigmoid层决定了我们将更新哪些值
- ✓ 然后，一个tanh层创建候选向量$\tilde{C}_t$,该向量将会被加到细胞的状态中
- ✓ 最后，结合这两个向量来创建更新值

AI DISCOVERY

Update Memory



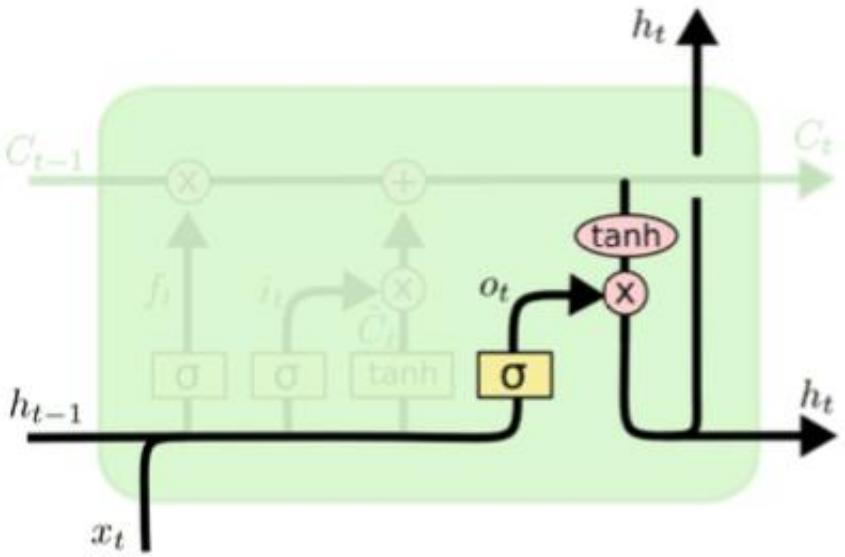
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

- 现在是时候去更新上一个状态值C_{t-1}了，将其更新为C_t
- 将上一个状态值C_{t-1}乘以f_t，以此表达期待忘记的部分。之后将得到的值加上 $i_t * \tilde{C}_t$。这个得到的是新的状态值

Output Gate

AI DISCOVERY

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$
$$h_t = o_t * tanh(C_t)$$



- ✓ 输出门决定我们要输出什么， 此输出将基于当前的细胞状态
- ✓ 首先，通过一个sigmoid层，决定了我们要输出细胞状态的哪些部分。
- ✓ 然后，将细胞状态通过tanh（将值规范化到-1和1之间），并将其乘以Sigmoid门的输出，至此完成了输出门决定的那些部分信息的输出。

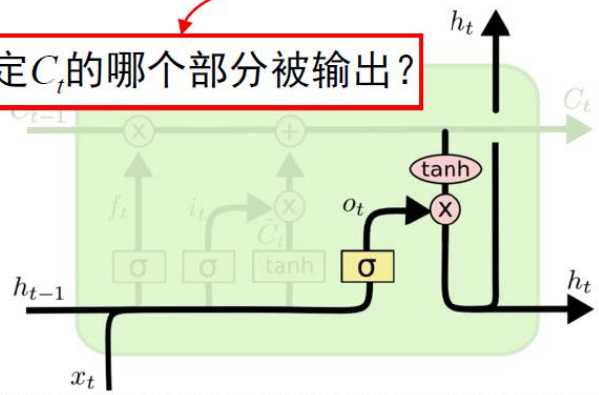
举个语言模型的例子，当看到一个主题词，考虑到后面可能出现的词，可能需要输出与动词相关的信息，比如单数还是复数，需要根据主题信息来决定具体输出什么。

AI DISCOVERY

Long Short Term Memory (LSTM)

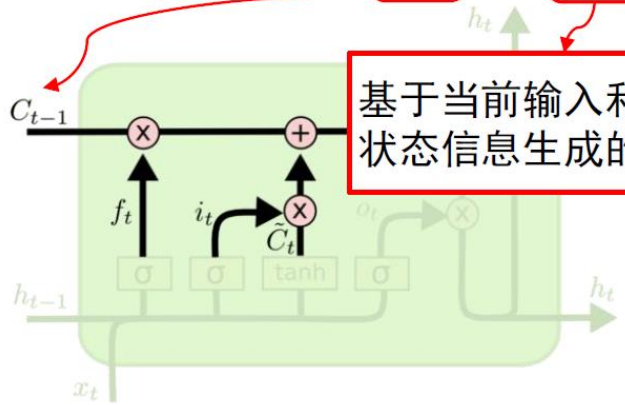
- 输出： $h_t = o_t * \tanh(C_t)$
- 输出门： $o_t = \sigma(W_o\,[h_{t-1}, x_t] + b_o)$

决定C_t的哪个部分被输出？



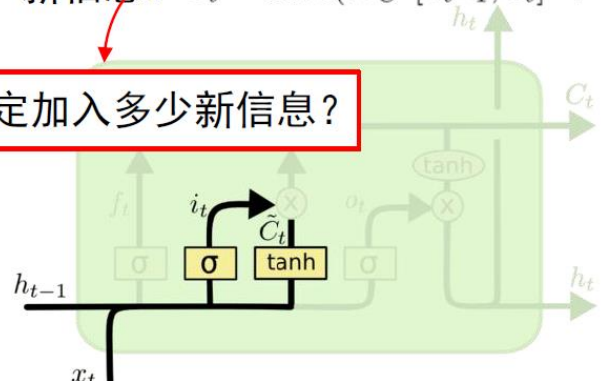
- 细胞状态： $C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$

基于当前输入和上个隐状态信息生成的新信息



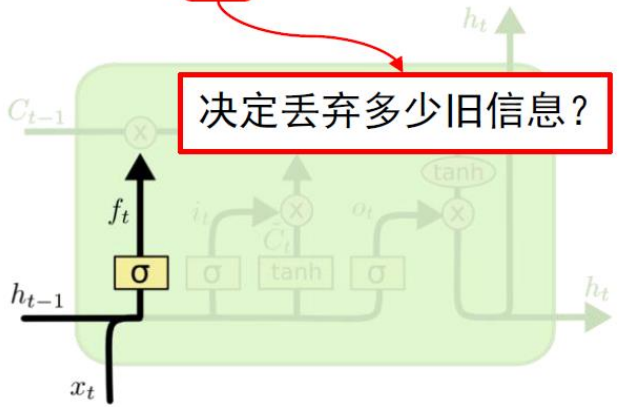
- 输入门： $i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$
- 新信息： $\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$

决定加入多少新信息？



- 遗忘门： $f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$

决定丢弃多少旧信息？

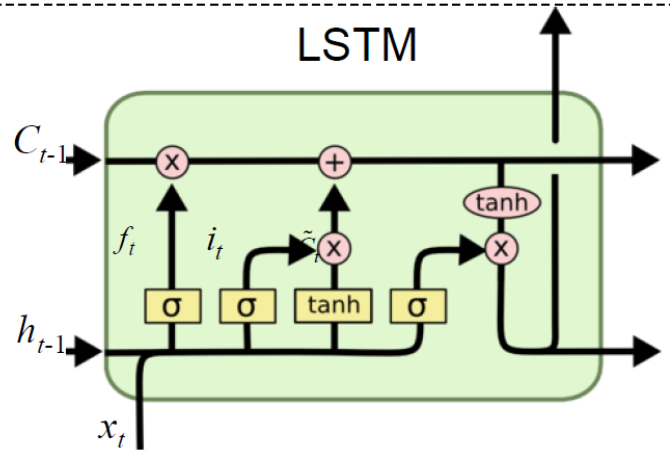


门限循环单元：GRU

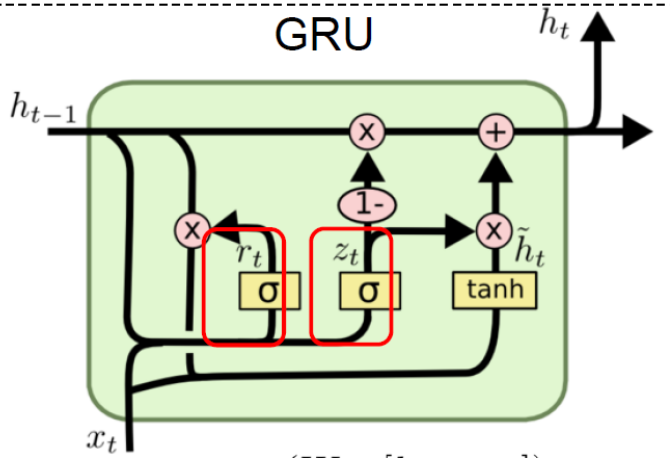
AI DISCOVERY

- 有单独的细胞状态
- 用输入门和遗忘门决定保留或放弃
- 新信息 $\tilde{C_t}$ 来源于 h_{t-1} 和 x_t
- 输出门控制细胞状态的输出

- 没有单独的细胞状态
- 用更新门决定保留或放弃
- $\tilde{h_t}$ 由重置门决定来自 h_{t-1} 的信息
- 直接输出隐状态



- 遗忘门 $f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right)$
- 输入门 $i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] + b_i\right)$
- 新信息 $\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$
- 细胞状态 $C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$
- 输出门 $o_t = \sigma\left(W_o\ [h_{t-1}, x_t] + b_o\right)$
- 隐状态 $h_t = o_t * \tanh\left(C_t\right)$



- 更新门 $z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$
- 重置门 $r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$
- 新信息 $\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$
- 隐状态 $h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$

保留哪些旧状态

接收哪些新状态

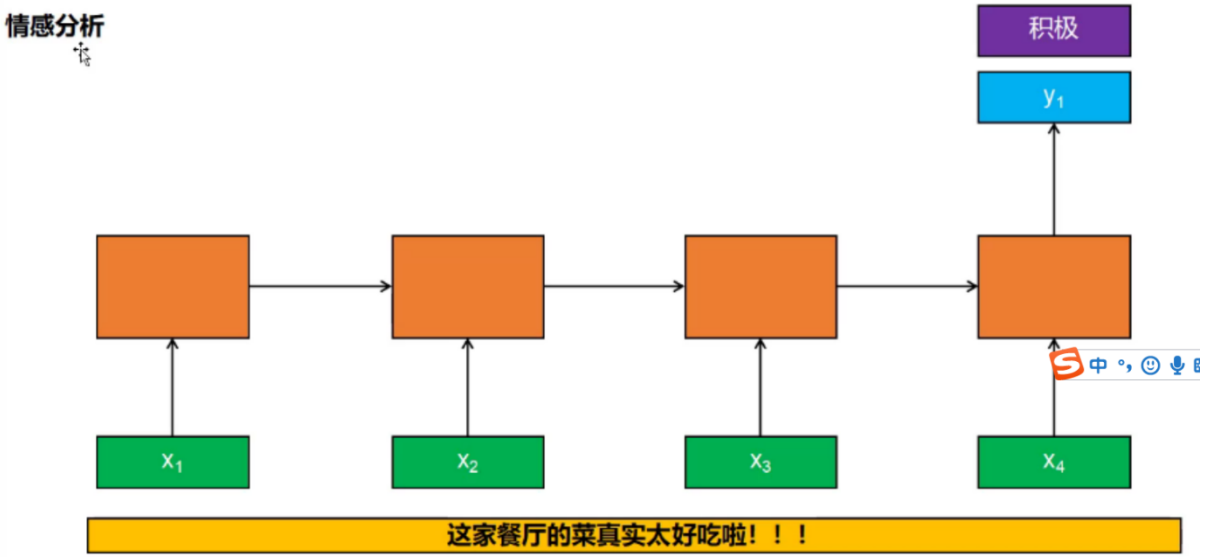
门限循环单元（Gated Recurrent Unit，GRU）是一种比 LSTM 更加简化的版本。在 LSTM 中，输入门和遗忘门是互补关系，因为同时用两个门比较冗余。GRU 将输入门与和遗忘门合并成一个门：更新门（Update Gate），同时还合并了记忆单元和隐藏神经元。

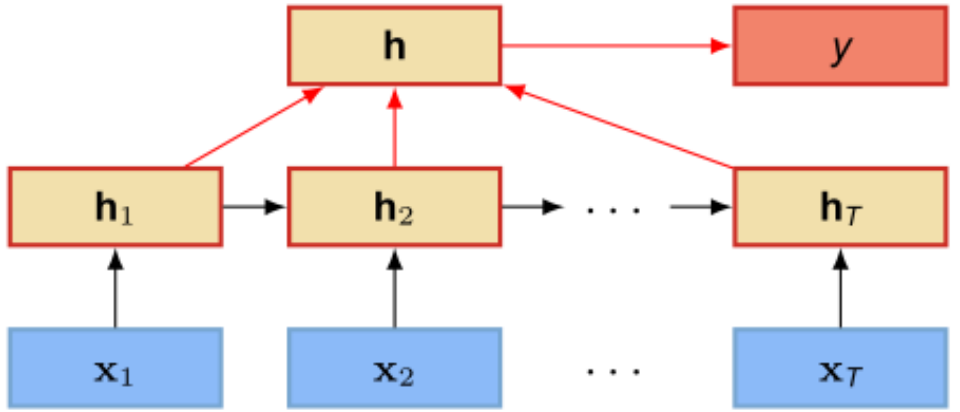
AI DISCOVERY

RNN应用：序列到类别

AI DISCOVERY

✓输入为序列，输出为类别。比如在文本分类中，**输入数据为单词的序列，输出为该文本的类别。**

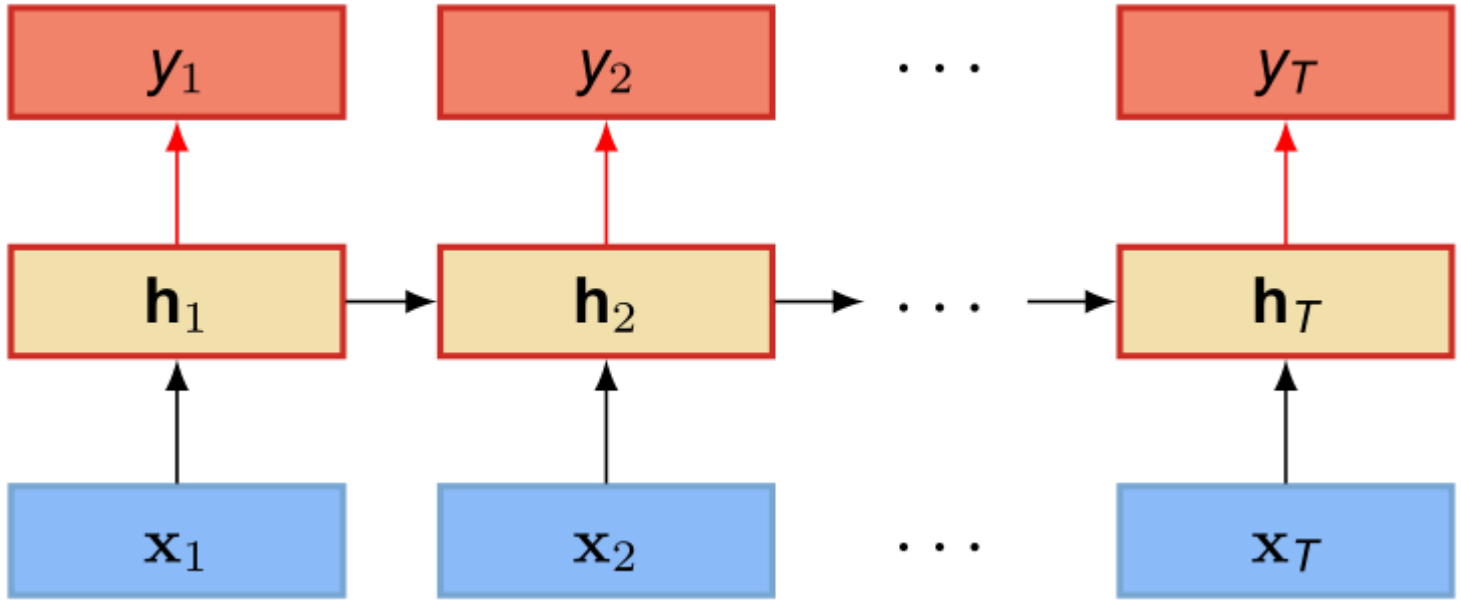




AI DISCOVERY

应用：同步序列到序列

✓**输入和输出同步，即每一时刻都有输入和输出**。比如在序列标注问题，每个时刻的输入都需要有一个输出。**输入序列和输出序列的长度相同**。



同步序列到序列——中文分词

AI DISCOVERY

中文分词：

任何网购退款均无需提供

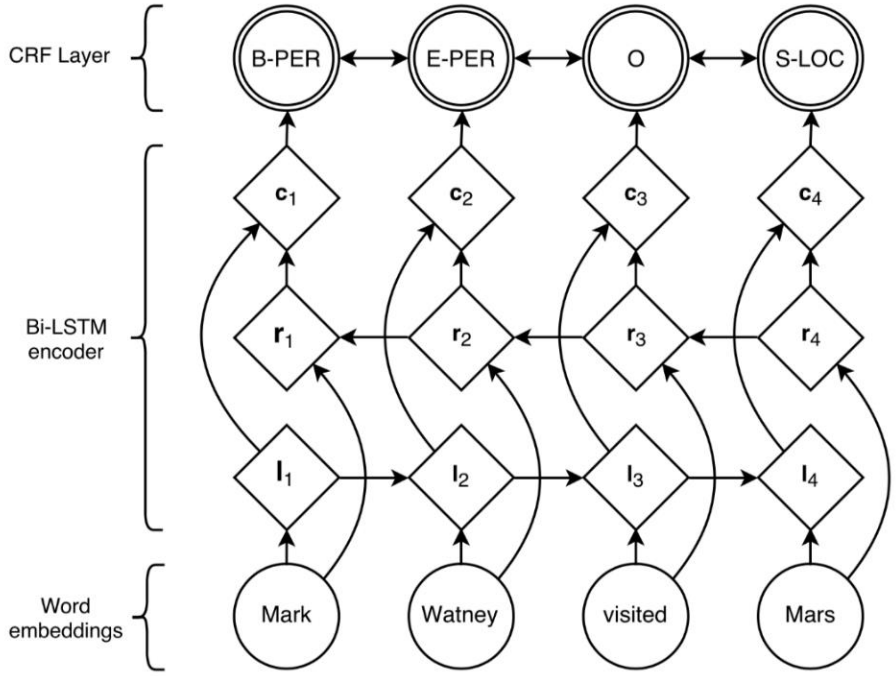
任何 | 网购退款 | 均 | 无需 | 提供

任(B)何(E) 网(B)购(I)退(I)款(E)均(O)无(B)需(E)提(B)供(E)

字级别的序列标注

输入：汉字序列

输出：标签序列



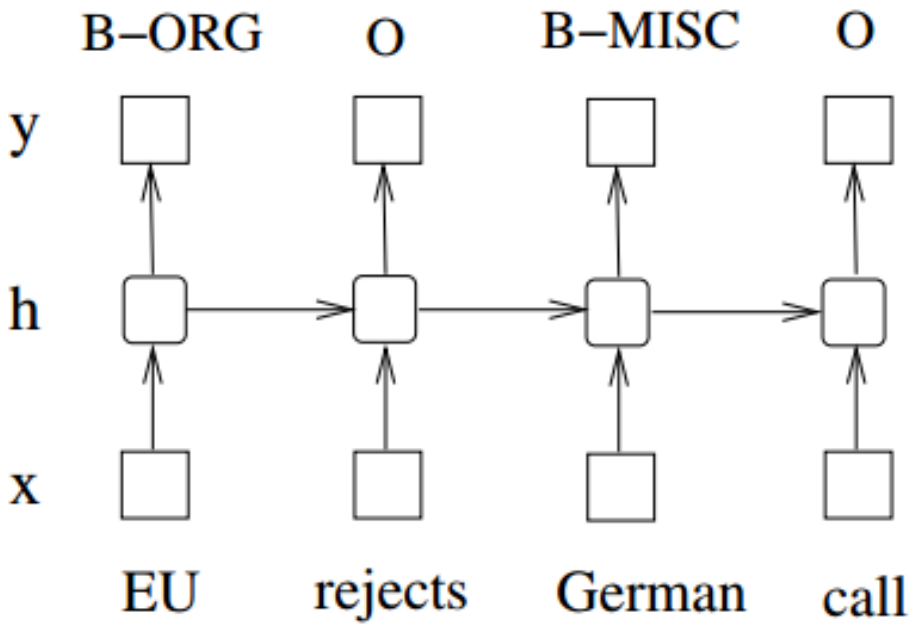
AI DISCOVERY

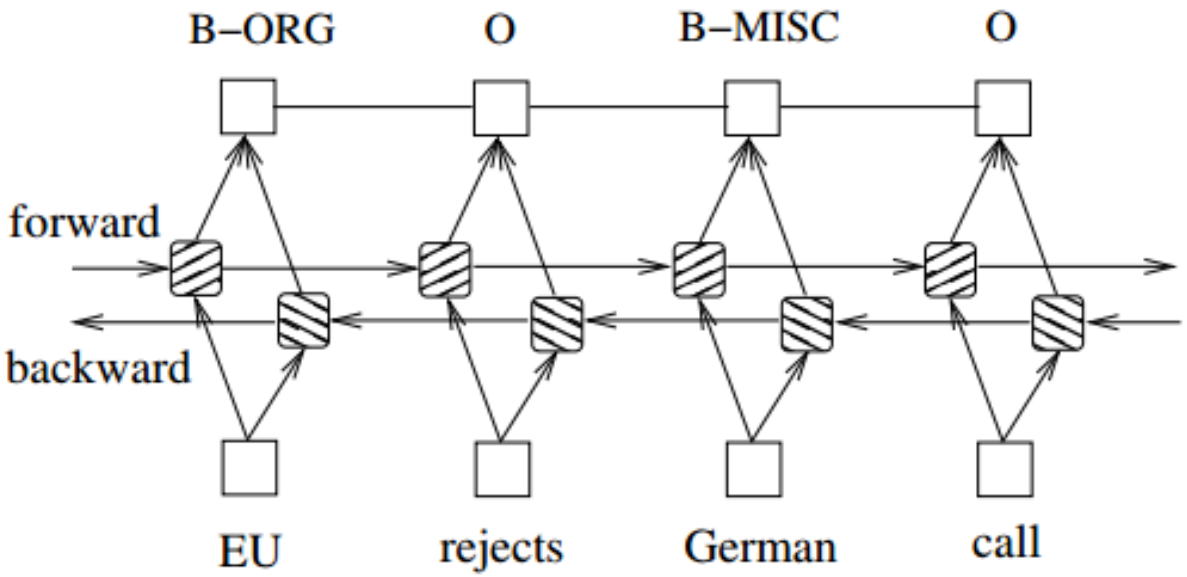
同步序列到序列——命名实体识别

AI DISCOVERY

输入：单词序列，每个时刻的输入状态是一个单词

输出：为每个单词输出一个标签，标识每个单词是否是特定命名实体的起始位置、中间位置、结束位置等

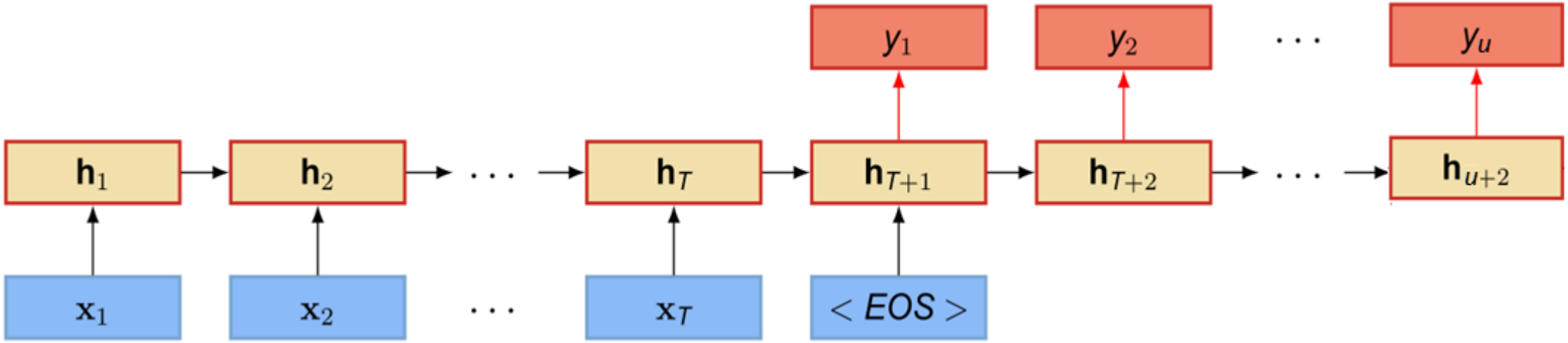




应用：异步序列到序列

AI DISCOVERY

✓**输入和输出不需要有严格的对应关系。**比如在机器翻译中，输入为源语言的单词序列，输出为目标语言的单词序列。输入和输出序列并不需要保持相同的长度。



异步序列到序列——机器翻译

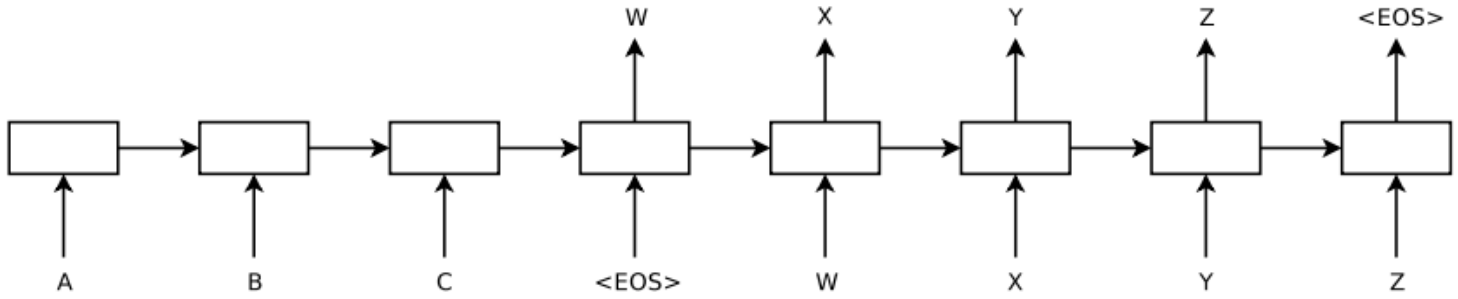
AI DISCOVERY

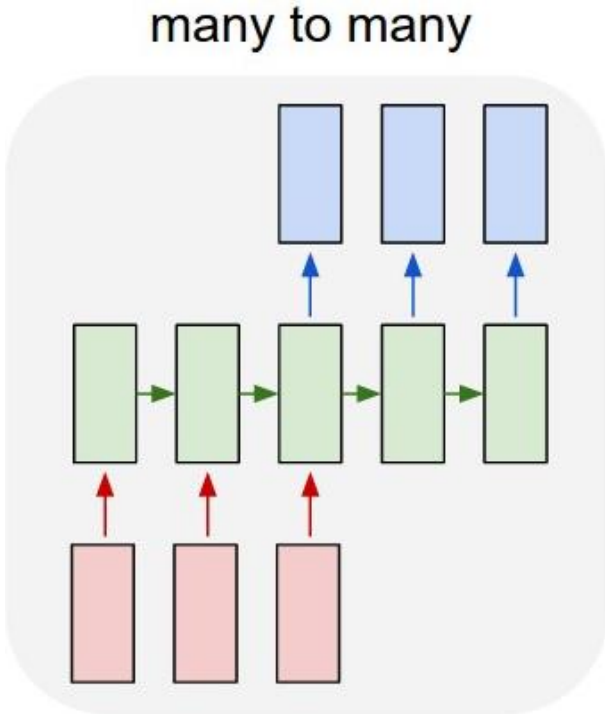
◆基于神经网络的机器翻译模型

- ✓一个 RNN 用来编码
- ✓另一个 RNN 用来解码

seq2seq

encoder-decoder





编码：读入源句子（变长向量），转换成一个固定的上下文向量c

解码：给定上下文和之前预测的词，预测下一个翻译的词

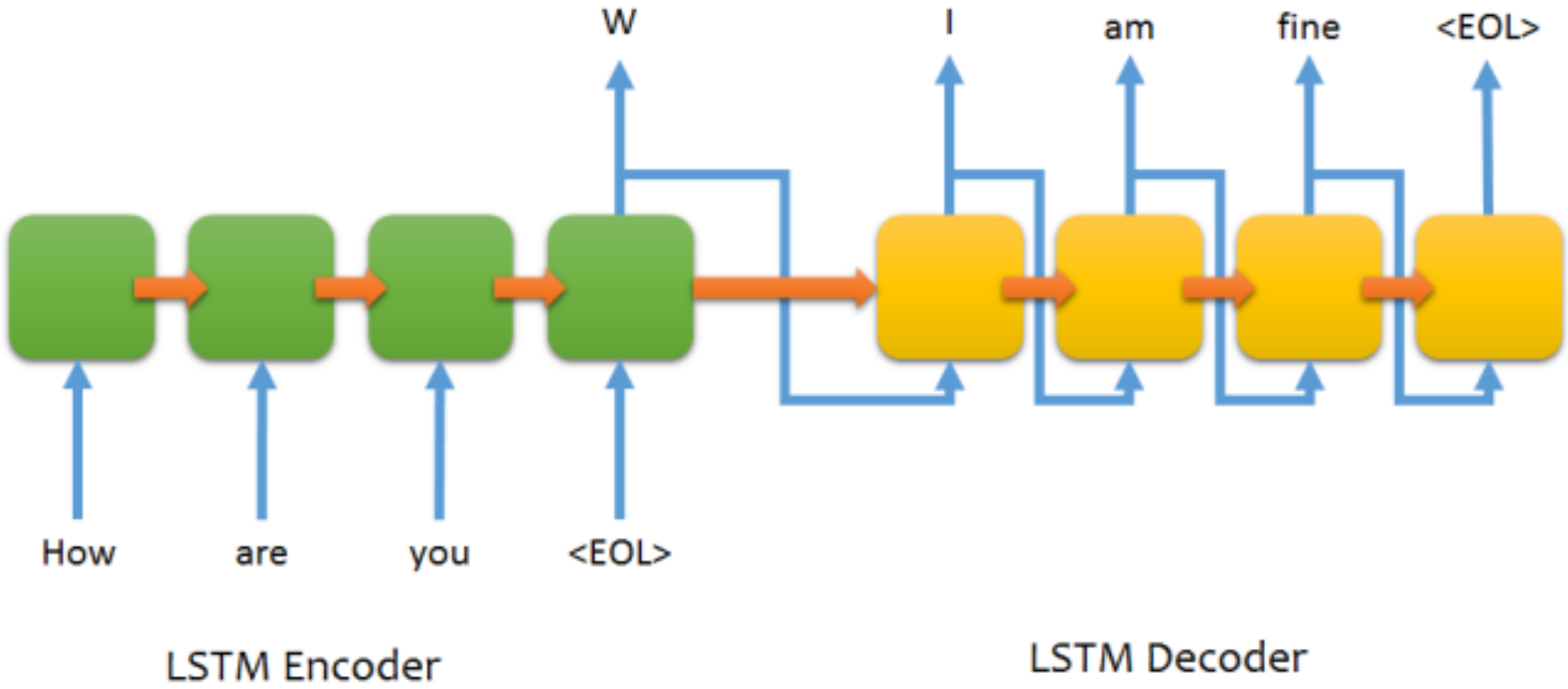
AI DISCOVERY

异步序列到序列——对话系统

AI DISCOVERY

Encoder端：对话中的上文（问句），例如 How are you

Decoder端：对话中的下句（回复句），例如 I am fine



AI DISCOVERY