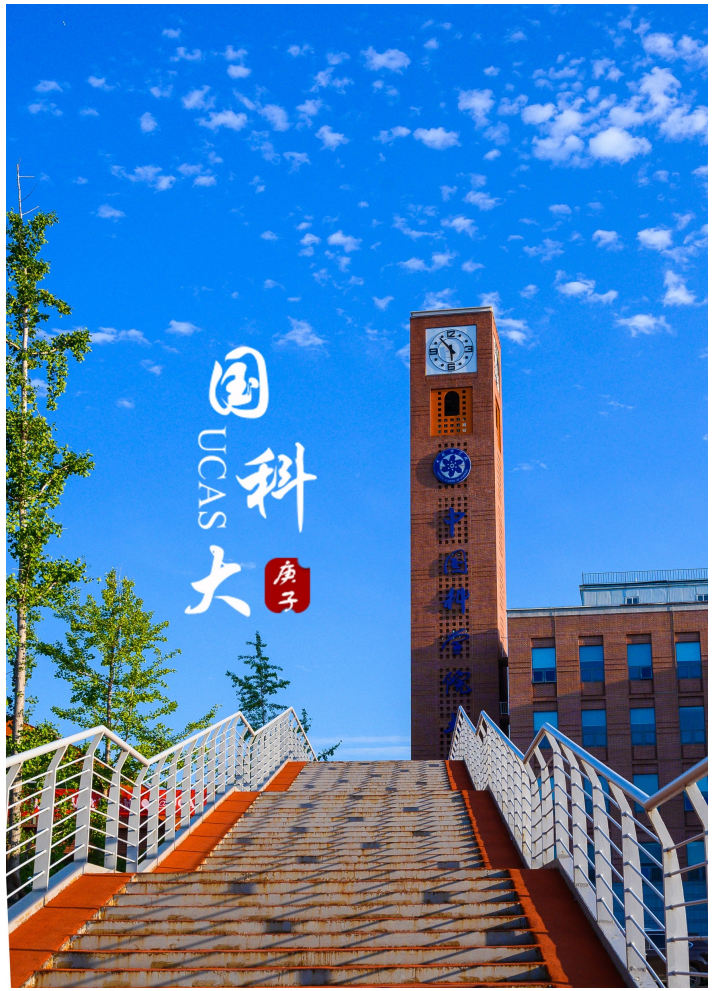
自然语言处理

第14讲 Hello again, NLP!

王 石 资康莉 刘 瑜

2026年春季课程

https://ictkc.github.io/teaching/



第十四讲

Hello again, NLP!

课件来自胡玥、曹亚男、方芳老师课程，在此感谢！

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

自然语言处理基础
Fundamentals of Natural Language Processing

授课教师：胡玥

2025.10

课程编码： 1802030839X2P2002H 课程名称：自然语言处理基础 授课团队 胡玥、曹亚男、方芳

内 容 提 要

6.1 文本分类

■ 文本分类

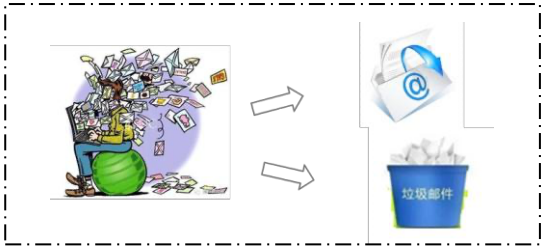
本节内容：

1. 分类任务概述
2. 序列结构文本分类
3. 图结构文本分类

1. 文本分类概述

■ 文本分类任务

利用计算机对大量的文档按照分类标准实现自动归类



分类任务	输入	任务建模	输出
	文本序列	分类模型	类别标签 类别标签根据任务定

输入：X 句子/篇章

输出：X 所属类别 Y

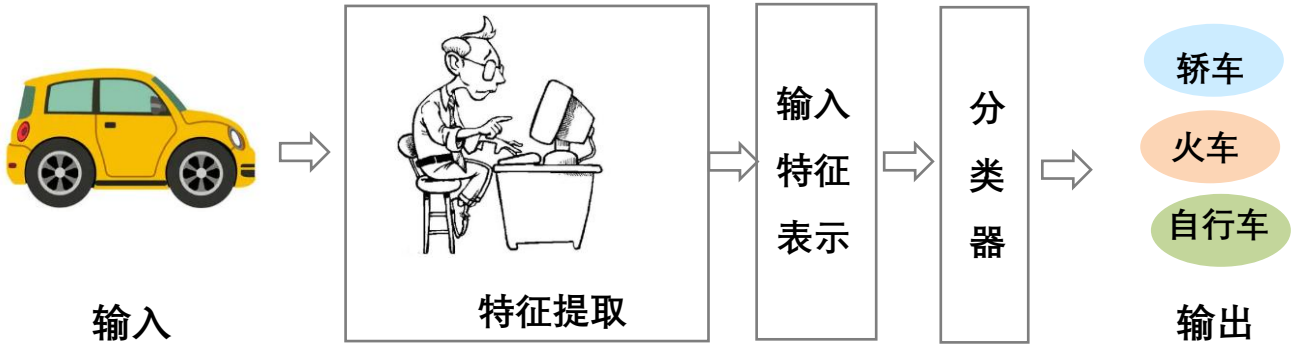
Y ∈{类别集合}

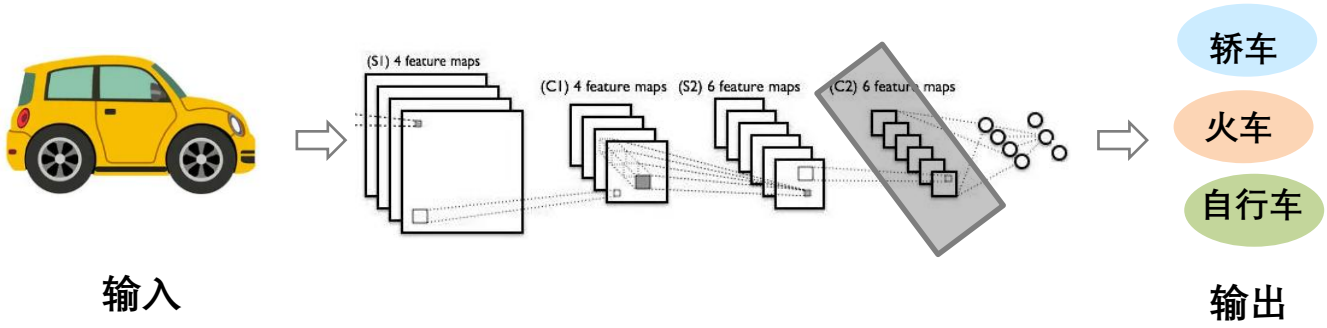
F： X → Y

1. 文本分类任务概述

■ 分类方法

概率统计时代：特征工程+算法（Naive Bayes/SVM/LR/KNN……）





深度学习时代：自动获取特征（表示学习）端到端分类

1. 文本分类概述

■ 神经网络分类方法

★ 基于词袋的文本分类

★ 基于卷积神经网络文本分类

★ 基于循环神经网络文本分类

★ 基于attention机制文本分类

★ 基于图卷积神经网络文本分类

也可以根据问题需要将上述方法结合形成混合模型

篇章级一般采用层次化的方法，先得到句子编码，然后以句子编码为输入，进一步得到篇章的表示

6.1 文本分类

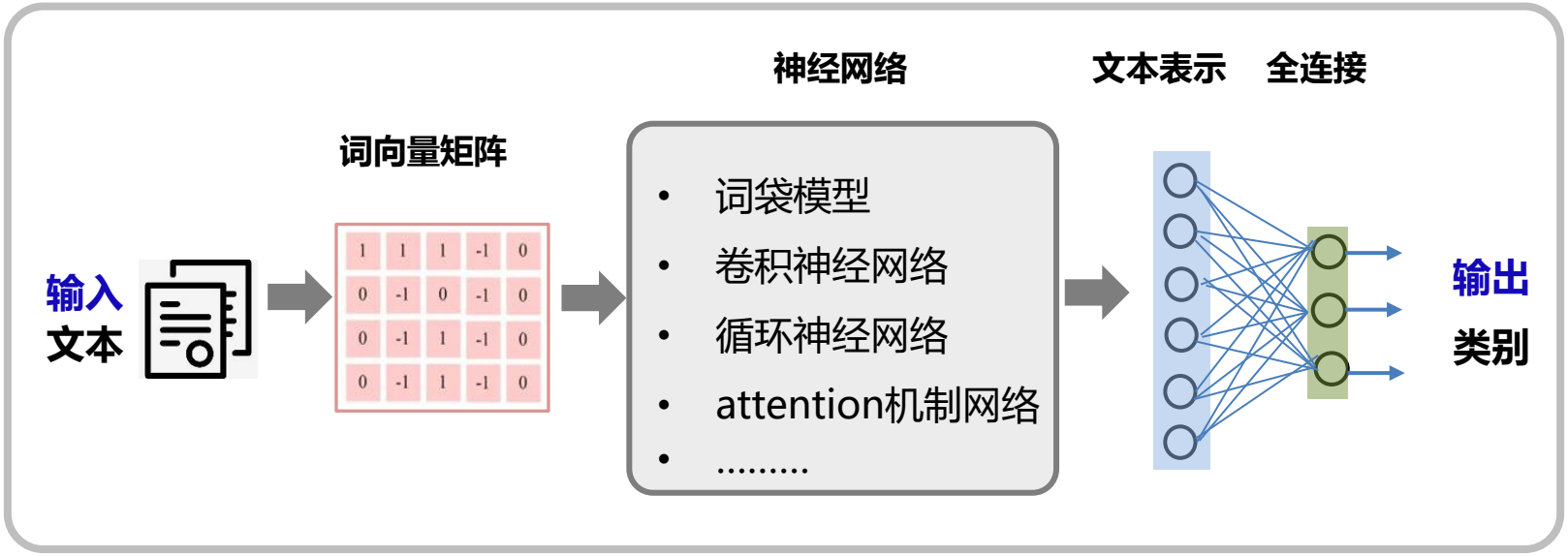
■ 文本分类

本节内容：

1. 分类任务概述
2. 序列结构文本分类
3. 图结构文本分类

2. 序列结构文本分类

■ 序列结构文本分类框架（文本整体分类）



分类任务	输入	任务建模	输出
	文本序列	分类模型	类别标签 类别标签根据任务定

关键问题：如何生成高质量的文本表示

2. 序列结构文本分类

★ FastText

Facebook提出了一种简单而有效的文本分类和表示学习方法可以在不到10分钟的时间内使用标准的多核CPU对超过10亿个单词进行快速文本训练，并在不到一分钟的时间内对312K类中的50万个句子进行分类。

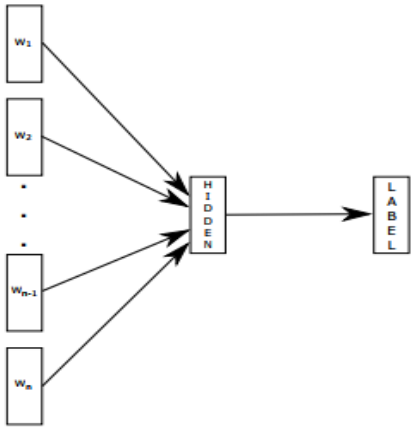


Figure 1: Model architecture for fast sentence classification.

输出层： 类别较少用softmax，

类别较多用hierarchical softmax

隐藏层： 将每个词向量相加取平均值

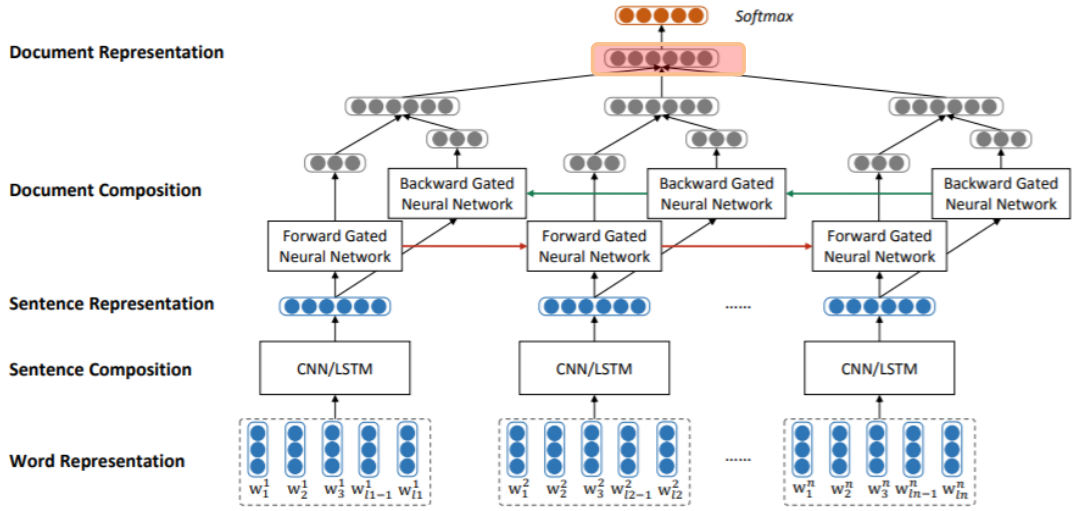
输入： Document中的每个词的词向量

Joulin et, al.Bag of Tricks for Efficient Text Classification，2017

2. 序列结构文本分类

★ LSTM/CNN-GRU（篇章级-混合模型）

- 篇章中所有句子的词向量矩阵作为输入
- 用CNN/LSTM形成句子级向量表示
- 由句向量用双向RNN 形成每句的带有上下句子信息的句子表示
- 由句向量形成篇章级向量表示
- 用篇章级向量做分类

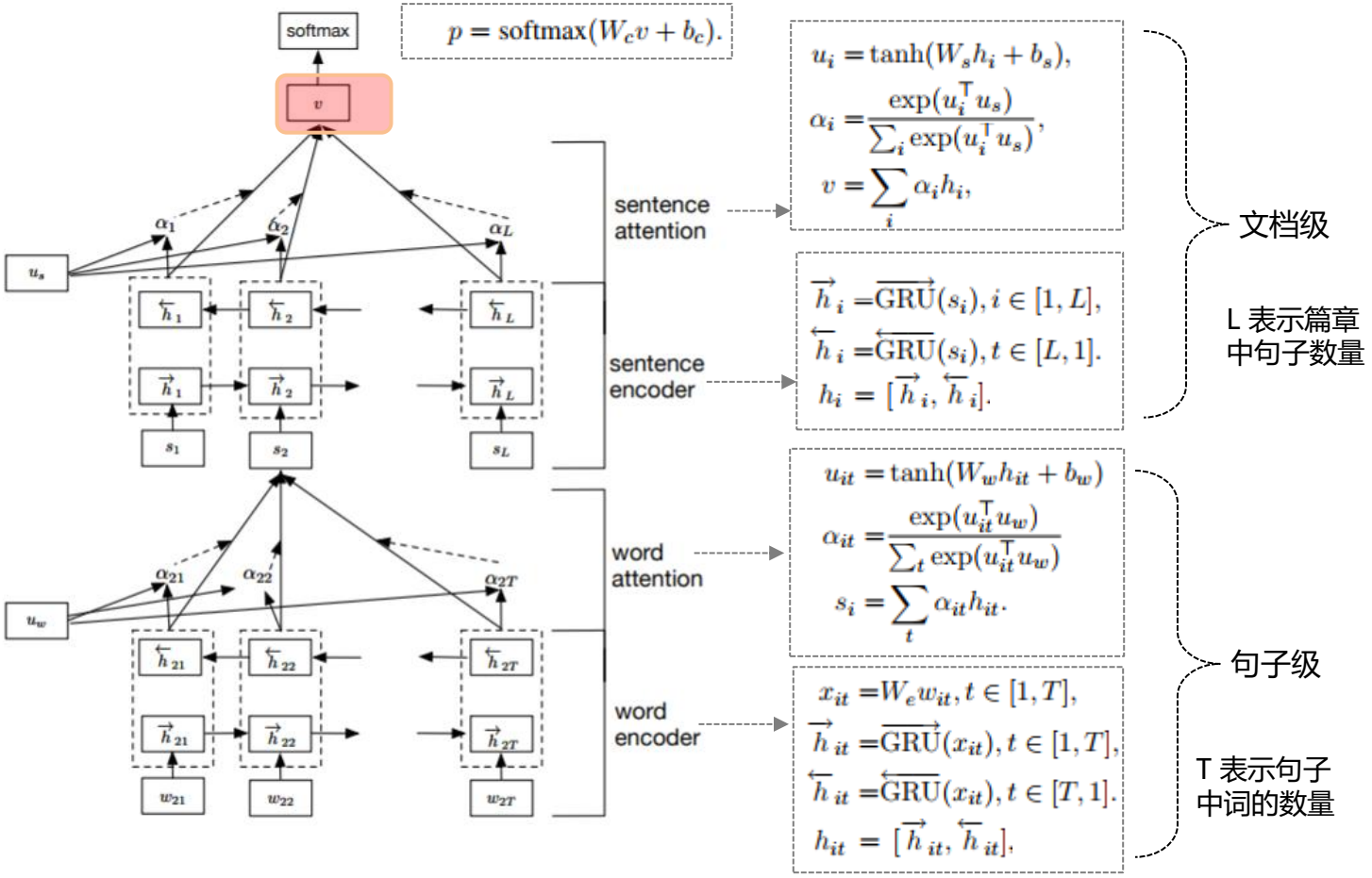


Tang et al. Document Modeling with Gated Recurrent Neural Network for Sentiment Classification, EMNLP 2015.

2. 序列结构文本分类

★ HAN （篇章级- Attention 模型）

从句子级和文档级两个层次引入Attention机制，可识别分类决策的重要单词和句子



Yang et al. Hierarchical Attention Networks for Document Classification，2016

6.1 文本分类

■ 文本分类

本节内容：

1. 分类任务概述
2. 序列结构文本分类
3. 图结构文本分类

3. 图结构文本分类

■ 图卷积神经网络文本分类

根据任务对原文本加入附加信息并构建原文本与附加信息的关系图（将附加的结构信息融入文本），然后利用图卷积的方法提取文本有效的特征表示

图卷积文本分类步骤：

1. Graph 构建

对原文本按照附加信息的不同构建不同的图结构。附加信息可以是词的近义信息，共现信息，先验知识信息等

2. 文本Graph节点特征表示

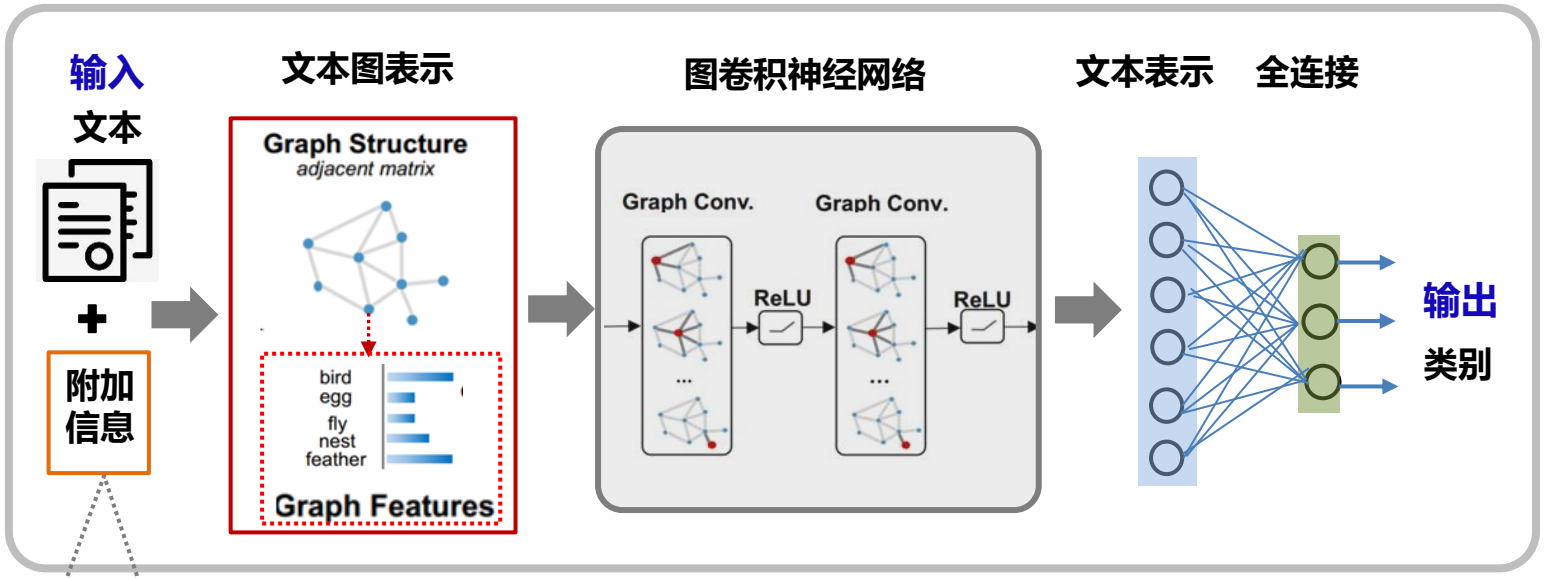
在图卷积中结点可以根据任务需要采用不同的结点表示方法。如，一般词向量， ELMO 词嵌入， Bi-LSTM 词向量嵌入，词袋词频 等方法

3. 图卷积算法

构建好输入图和图上结点表示后，可以根据不同的任务构建不同的图卷积算法。如，一般图卷积，加入注意力机制的图卷积等

3. 图结构文本分类

■ 图文本分类框架



词性POS，句法信息，关联，共现，主题，实体，实体关系，知识图谱.....

★ **图卷积文本分类步骤：**

1. Graph 构建
2. 文本Graph节点特征表示
3. 图卷积算法

关键问题：如何生成高质量的文本表示

3. 图结构文本分类

例1. 对文本进行图卷积分类

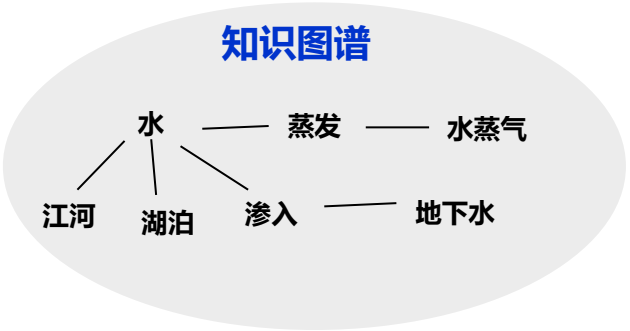
1. Graph 构建

文本

地面上的水不断蒸发变成了水蒸气，有的水在地面上汇成江河、湖泊，另外一些水渗入了地下，称为地下水.

建图

查询图谱



地面上的水不断蒸发变成了水蒸气，有的水在地面上汇成江河、湖泊，另外一些水渗入了地下，称为地下水.

邻接表形式图结构

注：附加信息可以是词性POS，句法信息，关联，共现，主题，实体，实体关系，知识图谱 等

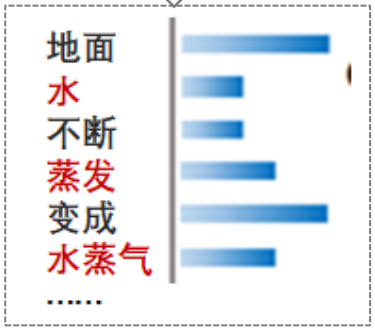
3. 图结构文本分类

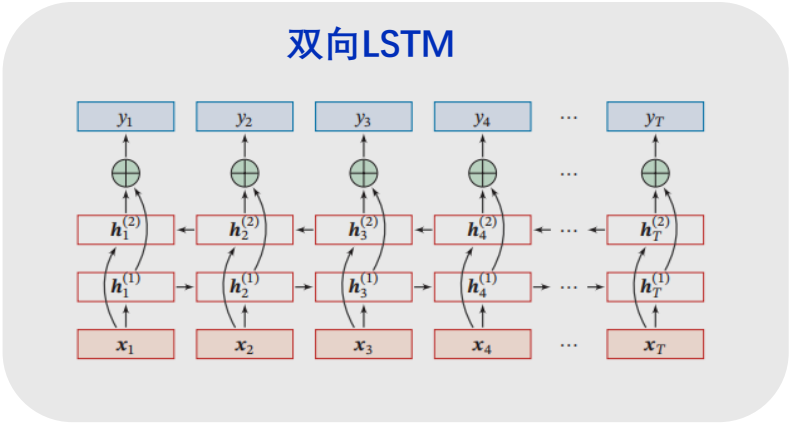
2. 文本Graph节点特征表示

地面上的水不断蒸发变成了水蒸气，有的水在地面上汇成江河、湖泊，另外一些水渗入了地下，称为地下水.

结点表示

表示方法

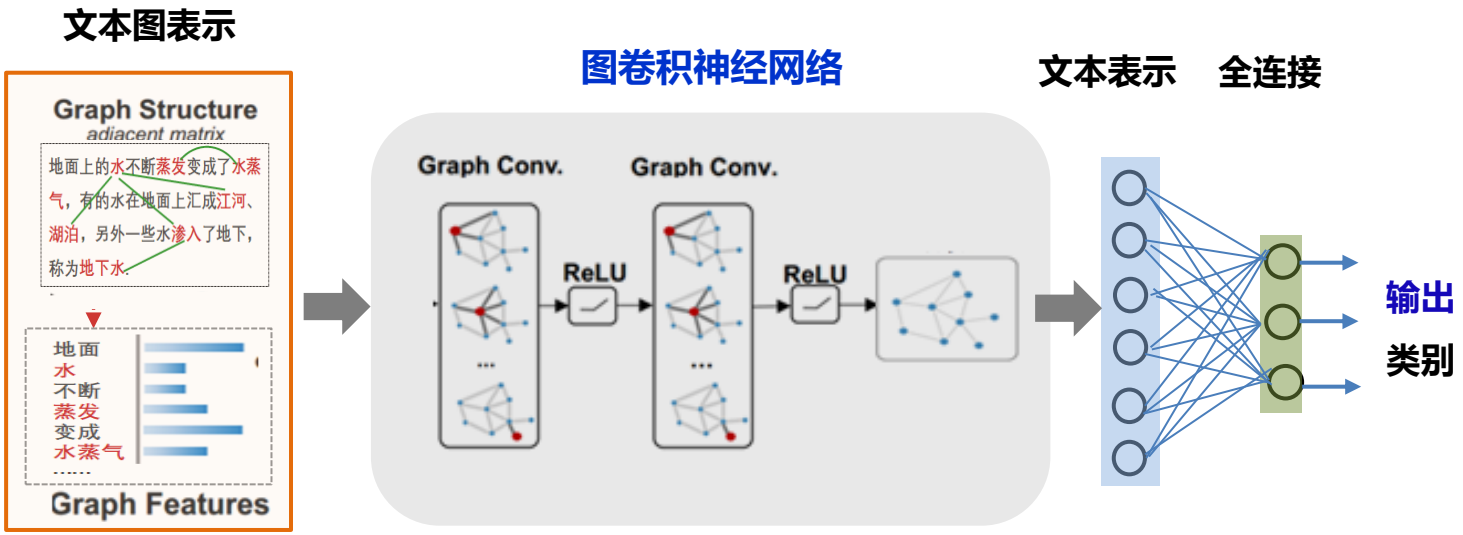




注：结点表示可根据具体问题需要采用 双向RNN， Elmo 词嵌入 等各种表示方法

3. 图结构文本分类

3. 图卷积算法



注：可以根据不同的任务构建不同的图卷积算法。如，图卷积网络文本分类算法，图卷积网络多层加权分类算法等

内 容 提 要

6.1 文本分类

6.2 文本匹配

6.3 序列标注

6.4 序列生成

6.5 综合示例

6.2 文本匹配

■ 文本匹配

本节内容：

1. 文本匹配任务概述
2. 孪生网络方法
3. 交互耦合方法

1. 文本匹配任务概述

■ 文本匹配

研究两段文本之间的关系。一般将这类问题定义为“文本匹配”问题，匹配含义根据任务的不同有不同的定义。很多自然语言处理的任务都会涉及文本匹配问题

如：

- 两个句子“感冒了是否要吃药”和“感冒了要吃什么药”

  问：两个句 子是否表达同样的意思？

- 两个句子“我正在上海旅游”和“我正在八达岭长城”

  问：这两句话是什么关系？

匹配任务	输入	任务建模	输出
	二段文本序列	匹配模型	二者关系 一般类别标签

这类问题一般可建模为“分类”或“排位”问题

1. 文本匹配任务概述

■ 与文本匹配相关的NLP任务

1.复述识别（paraphrase identification）

又称释义识别，是判断两段文本是不是表达了同样的语义，这一类场景一般建模成分类问题。

如：两个句子“感冒了是否要吃药”和“感冒了要吃什么药” 问：两个句子是否表达同样的意思？

解决方法：该问题的句子匹配是计算二个句子相似度，可建模为二分类问题

1. 文本匹配任务概述

2. 文本蕴含识别（Textual Entailment）

给定一个前提文本（text），根据这个前提去推断假说文本（hypothesis）与文本的关系，关系有：蕴含关系（entailment），矛盾关系（contradiction），蕴含关系（entailment）。这一类场景一般建模成多分类问题。

如：两个句子“我正在上海旅游”和“我正在八达岭长城” 问：这两句话是什么关系?

解决方法：该问题的句子匹配是计算二个句子之间的关系，可建模为多分类问题

1. 文本匹配任务概述

3.问答（QA）

根据Question在段落或文档中查找Answer，这类场景常常会被建模成分类问题；还有一类是根据Question从若干候选中找出正确答案，这类场景常常会被建模成排位（ ranking ）问题。

4.对话（Conversation）

与QA 类似，但是比QA更复杂，由于引入了历史轮对话，需要考虑在历史轮的限制下回复是否合理。一般建模为分类或排位问题。

5.信息检索（IR）

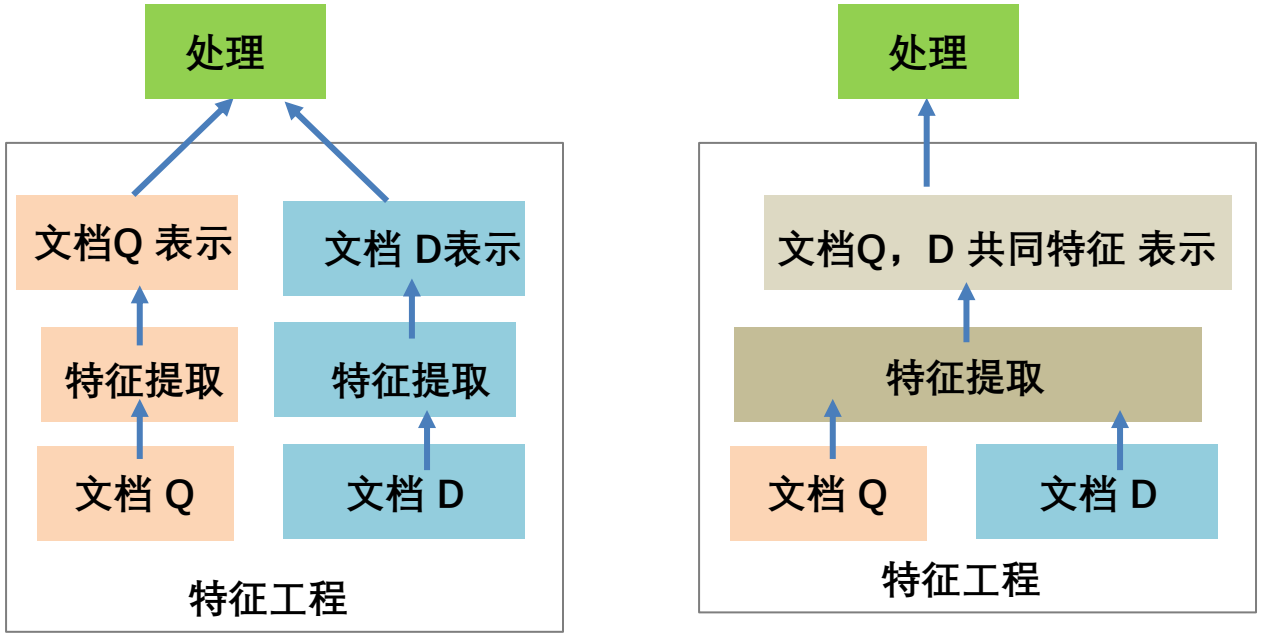
信息检索是一个更为复杂的任务，往往会有Query—Tittle，Query—Document的形式（Query可能是一个Document）检索需要计算相似度和排序一般建模为排位问题。

1. 文本匹配任务概述

■ 匹配方法

★ 统计方法：特征工程+算法（PRanking / margin/ SVM/LR……）

以上二种传统文本匹配方法主要集中在人工定义特征之上的关系学习，焦点在于如何人工提取的特征和设置合适的文本匹配学习算法来学习到最优的匹配模

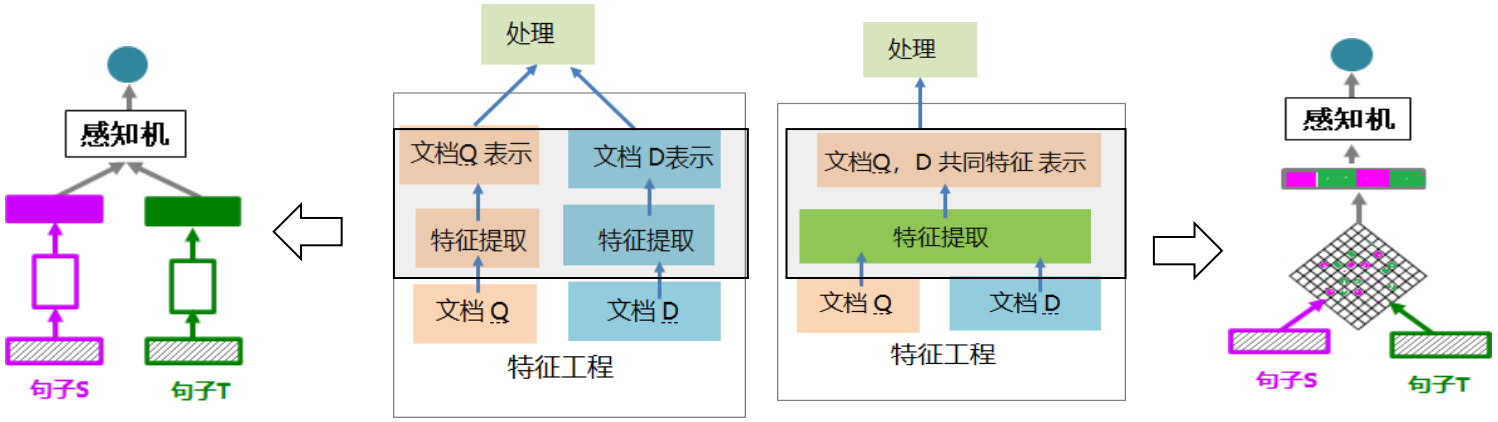


1. 文本匹配任务概述

★ 深度学习方法：

自动提取出词语之间的关系特征并结合短语匹配中的结构信息和文本匹配的层次化特性，更精细地描述文本匹配问题.

表示学习抽取有用特征



孪生网络

交互聚合

1. 文本匹配任务概述

匹配方法：

- **基于单语义文档表达的深度学习模型（基于表示-孪生网络）**

  主要思路：首先将单个文本先表达成一个稠密向量（分布式表达）然后直接计算两个向量间的相似度作为文本间的匹配度。

- **基于多语义文档表达的深度学习模型（基于交互-交互聚合）**

  主要思路：需要建立多语义表达，更早地让两段文本进行交互， 然后挖掘文本交互后的模式特征，综合得到文本间的匹配度。

复杂问题建模中常用交互耦合方法进行序列间的耦合表示，为后继挖掘交互表示模式特征提供信息。如选择式阅读理解等

6.2 文本匹配

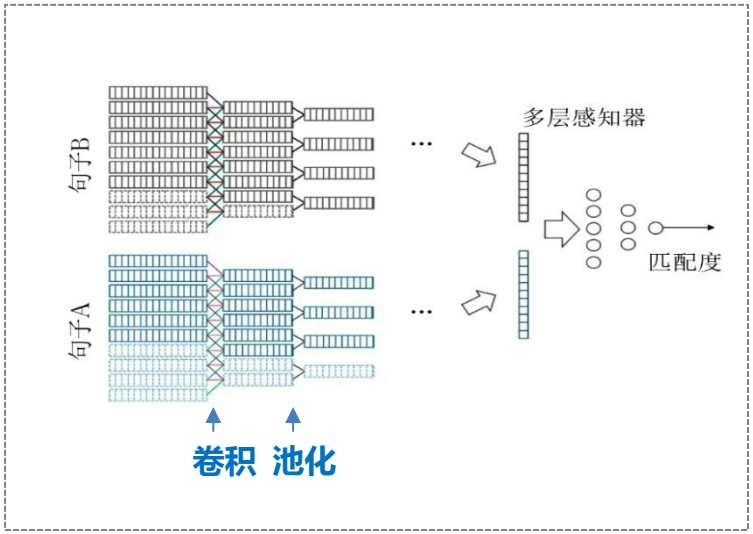
■ 文本匹配

本节内容：

1. 文本匹配任务概述
2. 孪生网络方法
3. 交互耦合方法

2. 孪生网络方法

★ ARC-I (基于CNN)



输入：句子A和B

运算关系：

1. padding句子A/B到定长
2. 分别做多轮的卷积+池化运算
3. 拼接两个向量
4. 输入给多层感知机

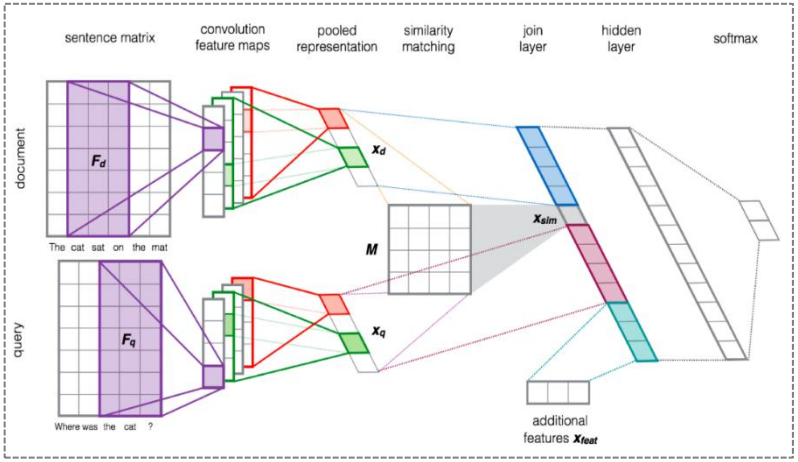
输出：句子A和B的匹配度

特点：在于将两个句子encode成句向量之后再用多层感知机进行分类，没有体现出句子之间的交互操作

Hu, Baotian, ， Convolutional Neural Network Architectures for Matching Natural Language Sentences.” NIPS (2014).

2. 孪生网络方法

★ CDNN (基于CNN)



特点：引入相似矩阵

输入：句子d和q

运算关系：

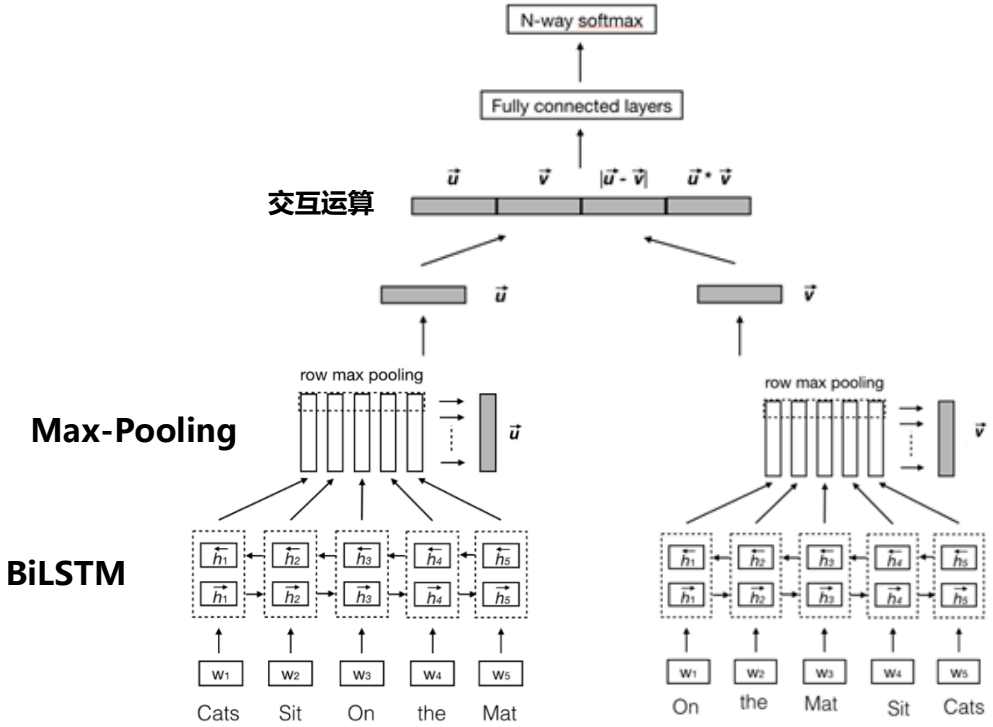
1. 分别做卷积+池化运算，得到句向量
2. 计算相似度 $\mathbf{X_{sim}} = \boldsymbol{X_q^T M X_d}$
3. 计算单词重叠数等其他特征
4. 拼接句向量、相似度和其他特征
5. 输入给多层感知机

输出：句子d和q的匹配度

Aliaksei , et al. Learning to Rank Short Text Pairs with Convolutional Deep Neural Networks.” SIGIR (2015).

2. 孪生网络方法

★ InferSent (基于RNN)



输入：句子A和B

运算关系：

1. 将A，B分别通过BiLSTM-Max表示成句向量
2. 将2个句向量交互运算输入给多层感知机

输出：句子A和B的匹配度

Alexis et al.. Supervised learning of universal sentence representations from natural language inference data. (EMNLP). 2017

6.2 文本匹配

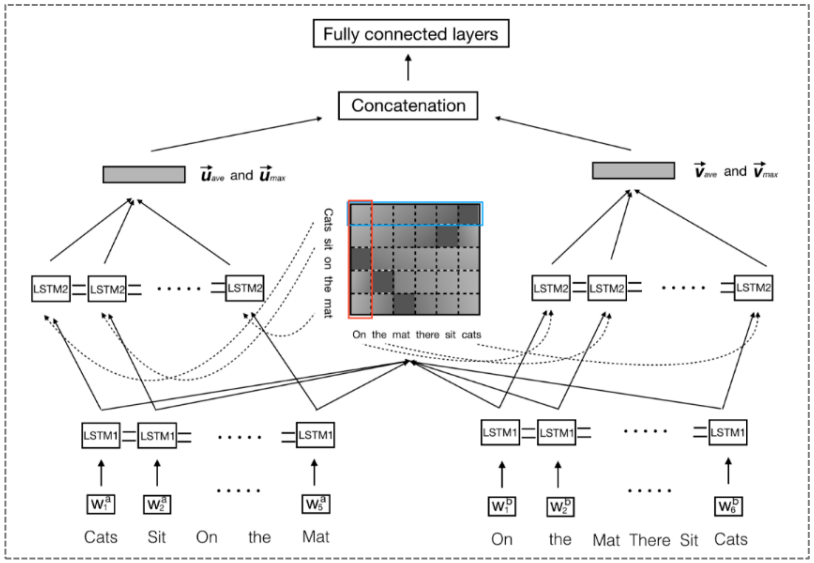
■ 文本匹配

本节内容：

1. 文本匹配任务概述
2. 孪生网络方法
3. 交互耦合方法

2. 交互聚合方法

★ ESIM (基于RNN 和 Attention)



运算关系：

1. 把句子X/Y表达成词向量序列，输入第一层BiLSTM进行编码
2. 将得到的词表示相互计算Attention得到交互矩阵
3. 将权重化的词表示与Attention值进行交互计算，再输入第二层BiLSTM
4. 将Max-Pooling和Aver-Pooling的结果串联
5. 输入给多层感知机

输入：句子X和Y

输出：句子X和Y的匹配度

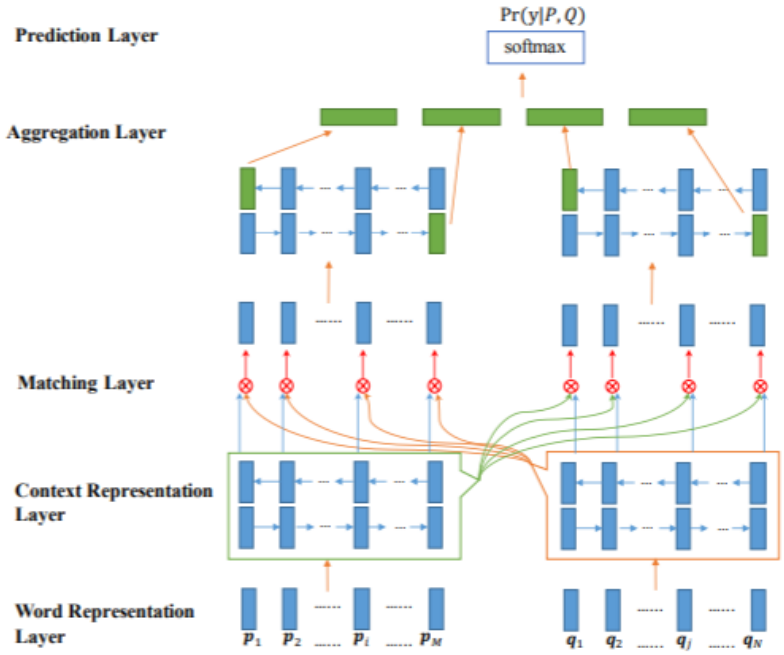
优势：Attention机制更好地建模句子关系

Qian Chen, et al. Enhanced LSTM for natural language inference. (ACL).

2. 交互聚合方法

★ BiMPM(基于RNN)

该模型主要用于做文本匹配，即计算文本相似度。创新点在于采用了双向多角度匹配，采用matching-aggregation的结构，把两个句子之间的单元做相似度计算，最后经过全连接层与softamx层得到最终的结果



ESIM模型包含五部分：

- Word Representation Layer
- Context Representation Layer
- Matching Layer
- Aggregation Layer
- Prediction Layer

Zhiguo Wang, et al. Bilateral Multi-Perspective Matching for Natural Language Sentences

内 容 提 要

6.1　文本分类

6.2　文本匹配

6.3　序列标注

6.4　序列生成

6.5　综合示例

6.3 序列标注

■ 序列标注

本节内容：

1. 序列标注问题概述

2. 隐马尔科夫模型HHM(概率统计模型)

4. 神经网络序列标注模型（深度学习模型）

1. 序列标注问题概述

问题引入

例：实体识别

问题1：将给定的输入序列中的人名识别出来（人名识别）

新任总裁罗建国宣布了对部门经理邓奇的任免通知

问题2：将给定的输入序列中的组织机构名识别出来（组织机构名）

新任总裁罗建国宣布了对远大公司经理国庆的任免通知

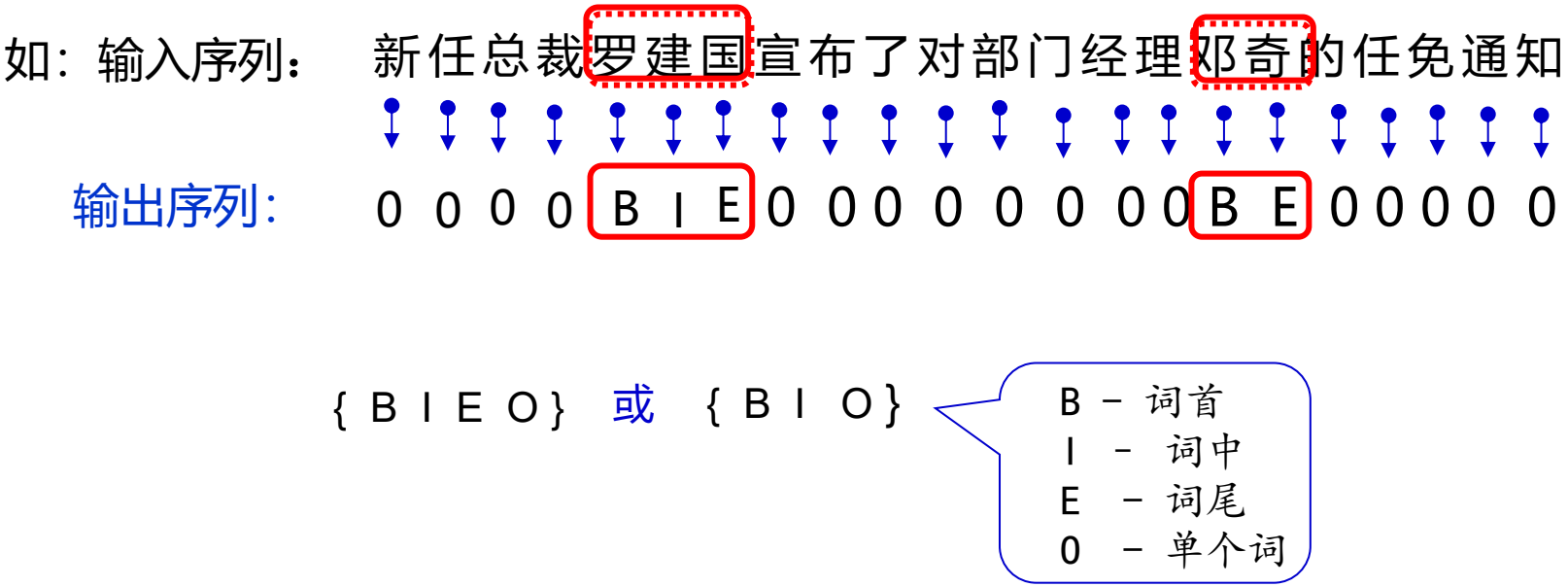
问题3：将给定的输入序列中的军事术语抽取出来

鹰式战斗机是一款极为优秀的多用途战斗机

1. 序列标注问题概述

解决方案：“将输入的语言序列转化为标注序列”，通过标注序列标签含义来解决问题。

◆ **命名实体识别（人名识别）**



1. 序列标注问题概述

◆ **命名实体识别（组织机构名识别）**

输入序列：新任总裁罗建国宣布了对远大公司经理国庆的任免通知

标注序列：O OO OOOO O OOO BIIEOOO OOOO OO

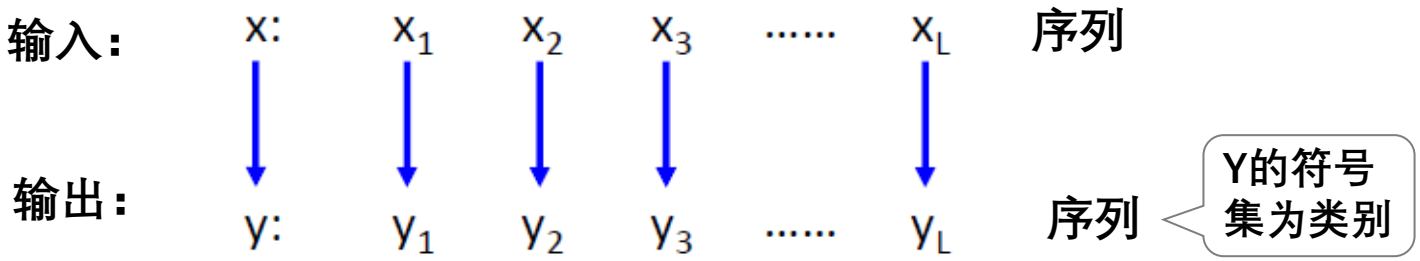
◆ **信息抽取（实体识别）**

输入：　鹰 式 战 斗 机 是 一 款 极 为 优 秀 的 多 用 途 战 斗 机

输出：　B I I I E O O O O O O O O B I I I I E

1. 序列标注问题概述

序列标注方法：



序列标注任务(方法)	输入	任务建模	输出
	非结构化文本序列	序列标注模型	标签序列

标注问题是分类问题的推广，是复杂结构预测的简单形式（监督学习）

1．序列标注问题概述

统计时代许多自然语言处理问题 均可转化为序列标注问题，如 分词、短语识别、依存分析、语义角色标注，信息抽取……

例：词性序列标注（POS）

问题：将给定的输入序列中词的词性标出来

输入：Flies like a flower

输出：N V ART N

结果：Flies/N like /V a/ART flower/N

Y的标签集｛ 单词的词性，如 N 、V 等｝

■ 序列标注

本节内容：

1. 序列标注问题概述

2. 隐马尔科夫模型HHM(概率统计模型)

4. 神经网络序列标注模型（深度学习模型）

■ 序列标注

本节内容：

1. 序列标注问题概述

2. 隐马尔科夫模型HHM(概率统计模型)

4. 神经网络序列标注模型（深度学习模型）

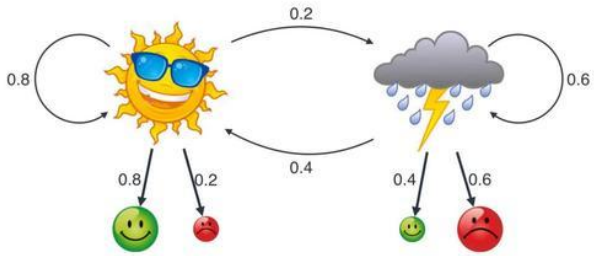
2. 隐马尔科夫模型

隐马尔可夫模型 （Hidden Markov Model, HMM）

描写：该模型是一个双重随机过程，我们不知道具体的状态序列，只知道状态转移的概率，即模型的状态转换过程是不可观察的（隐蔽的）而可观察事件的随机过程是隐蔽状态转换过程的随机函数。

--- 创建于20世纪70年代 ---

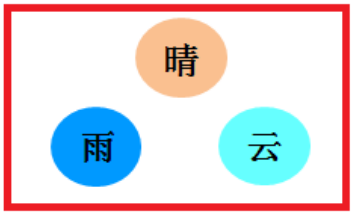
通过可见的事物的变化揭示深藏其后的内在的本质规律



2. 隐马尔科夫模型

马尔可夫模型：

S:



A:

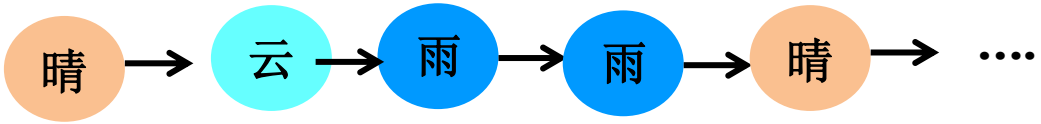
$A = [a_{ij}] =$

	雨	云	晴
雨	0.4	0.3	0.3
云	0.2	0.6	0.2
晴	0.1	0.1	0.8

π：晴 云 雨

(1，0，0)

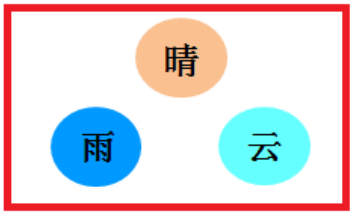
天气变化



2. 隐马尔科夫模型

隐马尔可夫模型HMM：

S:



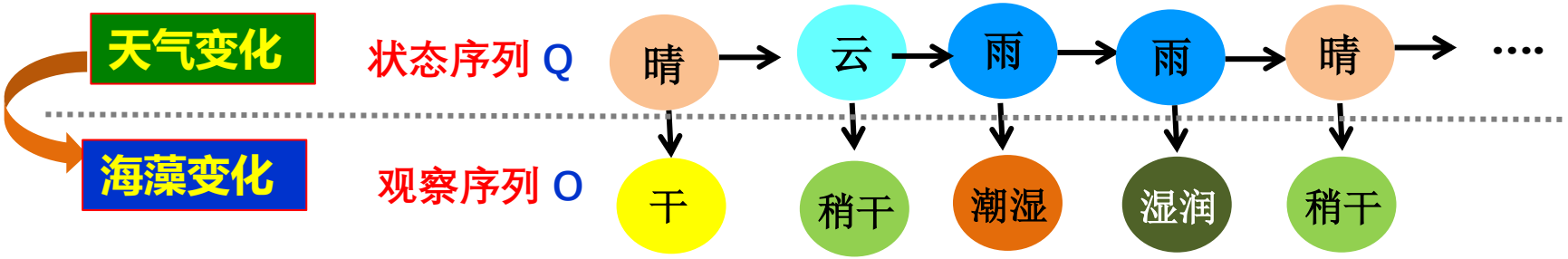
A:

$A = [a_{ij}] =$

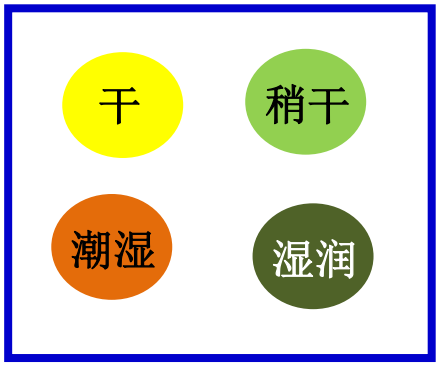
	雨	云	晴
雨	0.4	0.3	0.3
云	0.2	0.6	0.2
晴	0.1	0.1	0.8

π：晴 云 雨

(1，0，0)



O:



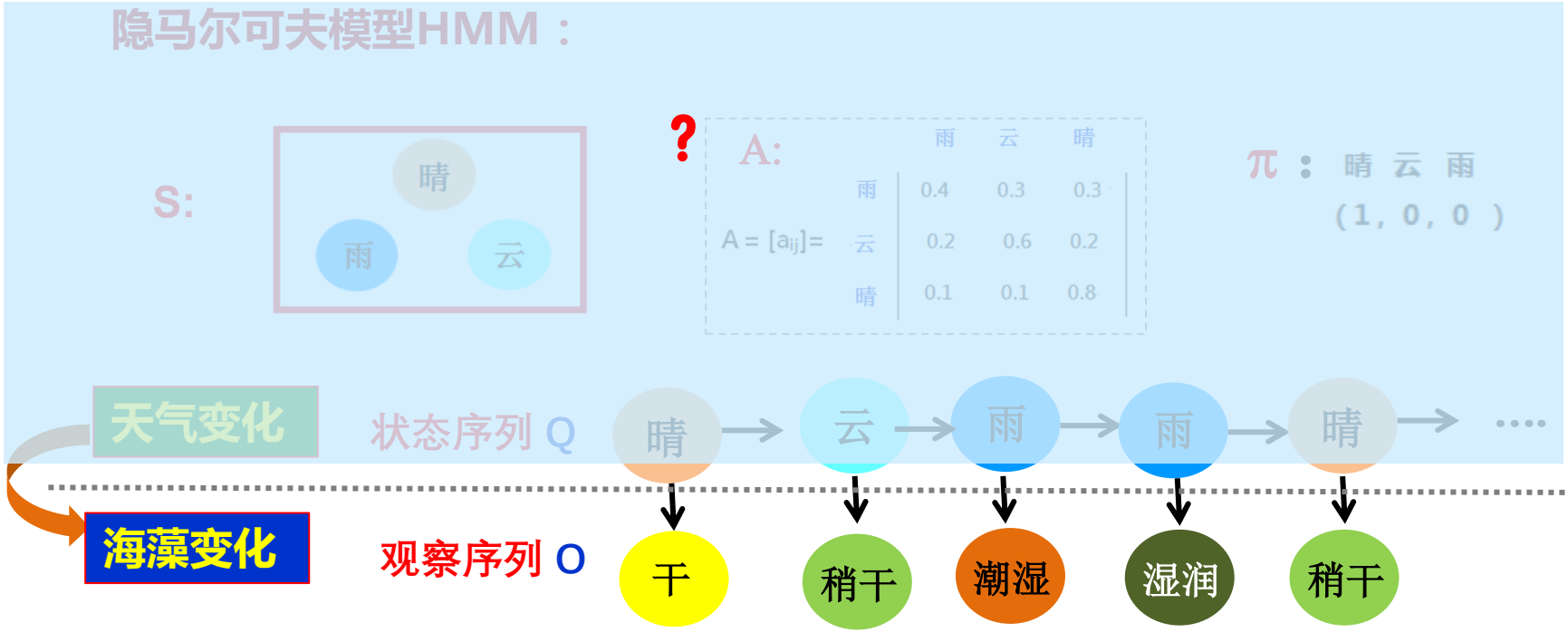
B:

天气 \ 海藻	干	稍干	潮湿	湿润
晴天	0.60	0.20	0.15	0.05
阴天	0.25	0.25	0.25	0.25
下雨	0.05	0.10	0.35	0.50

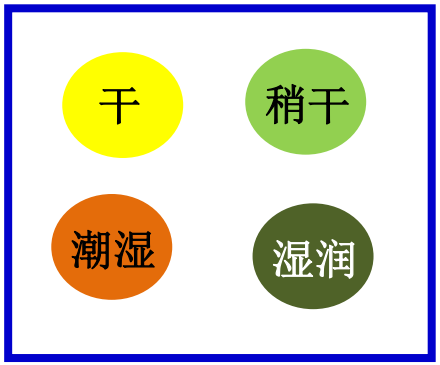
观察序列变化由状态序列变化引起

（两者相关联）

2. 隐马尔科夫模型



O:



B:

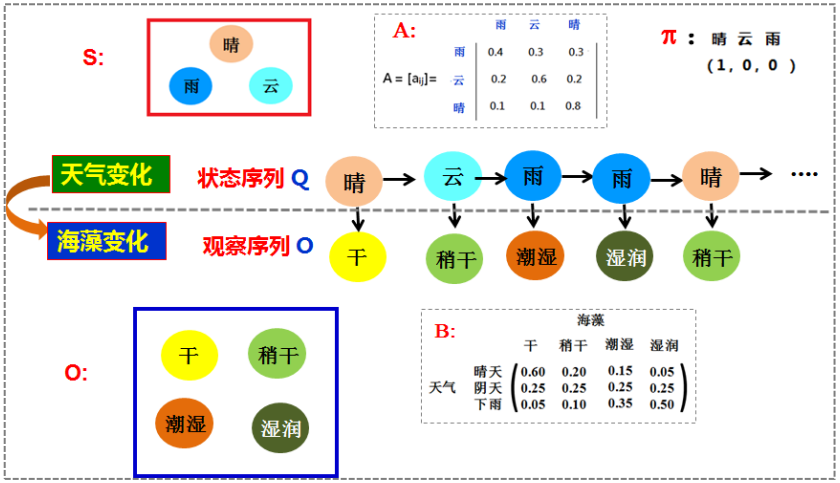
天气	海藻：干	稍干	潮湿	湿润
晴天	0.60	0.20	0.15	0.05
阴天	0.25	0.25	0.25	0.25
下雨	0.05	0.10	0.35	0.50

观察序列变化由状态序列变化引起

（两者相关联）

2. 隐马尔科夫模型

隐马尔可夫模型(HMM):



要素	含义	实例
S	模型中状态的有限集合	天气
O	每个状态可能的观察值	海藻
A	与时间无关的状态转移概率矩阵	天气转移概率矩阵
B	给定状态下，观察值概率分布	每个天气状态的海藻观测概率
π	初始状态空间的概率分布	初始时选择某天气概率

五元组 $\lambda = (S, O, \pi, A, B)$

或简写为 $\lambda = (\pi, A, B)$

HMM的特点：

- HMM的状态是不确定或不可见的，只有通过观测序列的随机过程才能表现出来
- 观察到的事件与状态并不是一 一对应，而是通过一组概率分布相联系
- HMM是一个双重随机过程，两个组成部分：
  - 马尔可夫链：描述状态的转移，用转移概率描述。
  - 一般随机函数：描述状态与观察序列间的关系， 用观察值概率描述。

2. 隐马尔科夫模型

HMM的三个假设：

对于一个随机事件，有一观察值序列： $O=O_1,O_2,\cdots O_T$

该事件隐含着一个状态序列： $Q = q_1,q_2,\cdots q_T$。

假设1： 马尔可夫性假设（状态构成一阶马尔可夫链）

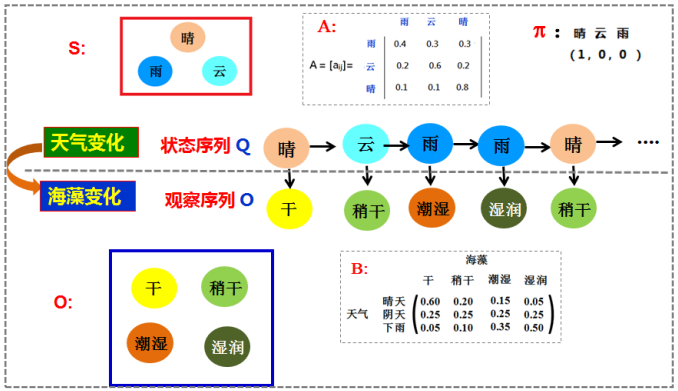
$$P(q_i|q_{i-1}\cdots q_1) = P(q_i|q_{i-1})$$

假设2： 不动性假设（状态与具体时间无关）

$$P(q_{i+1}|q_i) = P(q_{j+1}|q_j)，对任意i，j成立$$

假设3： 输出独立性假设（输出仅与当前状态有关）

$$p(O_1,\ldots,O_T \mid q_1,\ldots,q_T) = \Pi p(O_t \mid q_t)$$



2. 隐马尔科夫模型

HMM五元组说明：

1. 隐藏状态s：一个系统的(真实)状态，可以由一个马尔科夫过程进行描述（如, 天气）

3. 观察状态 o：在这个过程中‘可视’的状态（例如，海藻的湿度）

3. 状态转移概率矩阵 $A = a_{ij}$ ：包含了一个隐藏状态到另一个隐藏状态的概率 。其中,

$$\begin{cases} a_{ij} = p(q_{t+1} = S_j \mid q_t = S_i), \quad 1 \le i, j \le N \\ a_{ij} \ge 0 \\ \sum_{j=1}^{N} a_{ij} = 1 \end{cases}$$

4. 观察概率矩阵 $B=b_j(k)$: 从隐藏状态 S_j 观察到某一特定符号 v_k 的概率分布矩阵。

其中，

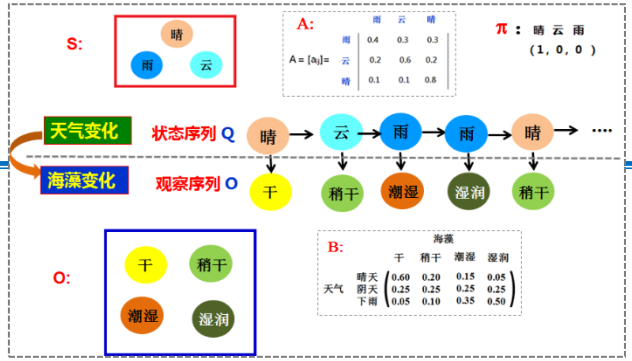
$$\begin{cases} b_j(k) = p(O_t = v_k \mid q_t = S_j), \quad 1 \le j \le N, \quad 1 \le k \le M \\ b_j(k) \ge 0 \\ \sum_{k=1}^{M} b_j(k) = 1 \end{cases}$$

2. 隐马尔科夫模型

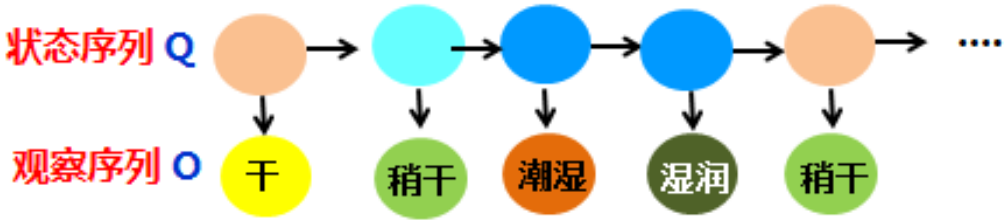
5. 初始状态的概率分布为：$\pi = \pi_i$，其中，

$$\begin{cases} \pi_i = p(q_1 = S_i), \quad 1 \le i \le N \\ \pi_i \ge 0 \\ \sum_{i=1}^{N} \pi_i = 1 \end{cases}$$

2. 隐马尔科夫模型



隐马尔可夫模型结构(HMM)：



输入： 观察序列

输出： 1. 观察序列的概率值 2. 隐状态序列

参数： $P(q_t|q_{t-1})$，$P(O_t|q_t)$

A矩阵　　B矩阵

函数关系：

(1) 观察序列的概率值 （ HMM评估问题 ）

(2) 隐状态序列 （ HMM解码问题 ）

2. 隐马尔科夫模型

HMM参数学习

隐马尔科夫模型参数

$$P(S_t \mid S_{t-1}) = \frac{P(S_{t-1}S_t)}{P(S_{t-1})} \qquad P(O_t \mid S_t) = \frac{P(O_t, S_t)}{P(S_t)}$$

训练思路：

通过观察序列 O = $O_1O_2 \cdots O_T$作为训练数据，用最大似然估计，使得观察序列 O 的概率 p(O|μ) 最大。

2. 隐马尔科夫模型

统计自然语言处理时代HMM模型在统计自然语言处理中有着广泛的应用

观察序列$O=O_1O_2\cdots O_T$： 处理的语言单位，一般为 词

状态序列 S =S1S2…ST： 与语言单位对应的句法信息，一般为 词类/词性

模型参数：初始状态概率、状态转移概率、发射概率 需要学习获得

★ 分词： HMM的评估问题：当分词出现多种可能时，求观察序列 $O=O_1O_2\cdots O_T$ 的概率， 结果取 概率最大的序列 ；解码问题：用序列标注直接进行分词

★ 词性标注： 相当HMM的解码问题。即求观察序列 $O=O_1O_2\cdots O_T$ 下，概率最大的标注序列 argmax P(Q|O ,μ)

★ 其他：如 短语识别、语音识别 …….

3. 隐马尔科夫模型（应用）

例1： HMM模型在词性标注中的应用

设，有如下从语料库训练得到的词性转移概率矩阵和词语生成概率矩阵

词性转移概率

词性	估计
PROB(ART | ø)	0.71
PROB(N | ø)	0.29
PROB(N | ART)	1
PROB(V | N)	0.43
PROB(N | N)	0.13
PROB(P | N)	0.44
PROB(N | V)	0.35
PROB(ART | V)	0.65
PROB(ART | P)	0.74
PROB(N | P)	0.26

词语生成概率

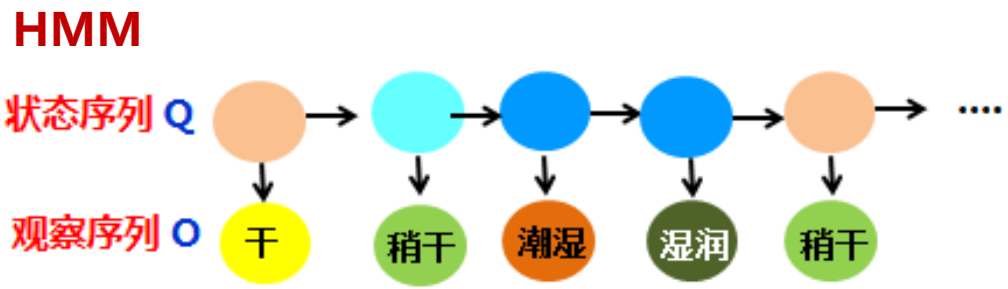
PROB(*the* | ART)	0.54
PROB(*flies* | N)	0.025
PROB(*flies* | V)	0.076
PROB(*like* | V)	0.1
PROB(*like* | P)	0.068
PROB(*like* | N)	0.012
PROB(*a* | ART)	0.360
PROB(*a* | N)	0.001
PROB(*flower* | N)	0.063
PROB(*flower* | V)	0.05
PROB(*birds* | N)	0.076

试对“ flies like a flower “ 进行词性标注

3. 隐马尔科夫模型（应用）

解：　问题求解目标：对每个词标出其词性

该问题属于序列标注问题，可用HMM模型进行标注



观察集（词集）： flies， like， a， flower

状态集（词性集）： N，V，P，ART

- 共有 256 种可能标注结果
- 可用 Viterbi 搜索算法 解码

词性转移概率

词性	估计
PROB(ART | ø)	0.71
PROB(N | ø)	0.29
PROB(N | ART)	1
PROB(V | N)	0.43
PROB(N | N)	0.13
PROB(P | N)	0.44
PROB(N | V)	0.35
PROB(ART | V)	0.65
PROB(ART | P)	0.74
PROB(N | P)	0.26

词语生成概率

PROB(*the* | ART)	0.54
PROB(*flies* | N)	0.025
PROB(*flies* | V)	0.076
PROB(*like* | V)	0.1
PROB(*like* | P)	0.068
PROB(*like* | N)	0.012
PROB(*a* | ART)	0.360
PROB(*a* | N)	0.001
PROB(*flower* | N)	0.063
PROB(*flower* | V)	0.05
PROB(*birds* | N)	0.076

Viterbi 搜索算法

1. 初始化： $\delta_1(i)=\pi_i b_i(O_1)$， $\varphi_1(i)=0$， $1\le i\le N$

2. 递归： $\delta_t(j)=[\max_{1\le i\le N}\delta_{t-1}(i)a_{ij}]b_j(O_t)$， $2\le t\le T, 1\le j\le N$

   $\varphi_t(j)=\arg\max_{1\le i\le N}[\delta_{t-1}(i)a_{ij}]b_j(O_t)$， $2\le t\le T, 1\le j\le N$

3. 终结： $p^*=\max_{1\le i\le N}[\delta_T(i)]$ ， $q_T^*=\arg\max_{1\le i\le N}[\delta_T(i)]$

4. 路径回溯： $q_t^*=\varphi_{t+1}(q_{t+1}^*)$， $t=T-1,T-2,\ldots,1$

观察序列（词）

flies　like　a　flower

状态（词性）

V

N

P

ART

词性转移概率

词性	估计
PROB(ART | ø)	0.71
PROB(N | ø)	0.29
PROB(N | ART)	1
PROB(V | N)	0.43
PROB(N | N)	0.13
PROB(P | N)	0.44
PROB(N | V)	0.35
PROB(ART | V)	0.65
PROB(ART | P)	0.74
PROB(N | P)	0.26

词语生成概率

PROB(*the* | ART)	0.54
PROB(*flies* | N)	0.025
PROB(*flies* | V)	0.076
PROB(*like* | V)	0.1
PROB(*like* | P)	0.068
PROB(*like* | N)	0.012
PROB(*a* | ART)	0.360
PROB(*a* | N)	0.001
PROB(*flower* | N)	0.063
PROB(*flower* | V)	0.05
PROB(*birds* | N)	0.076

Viterbi 搜索算法

1. 初始化： $\delta_1(i)=\pi_i b_i(O_1)$， $\varphi_1(i)=0$， $1\le i\le N$

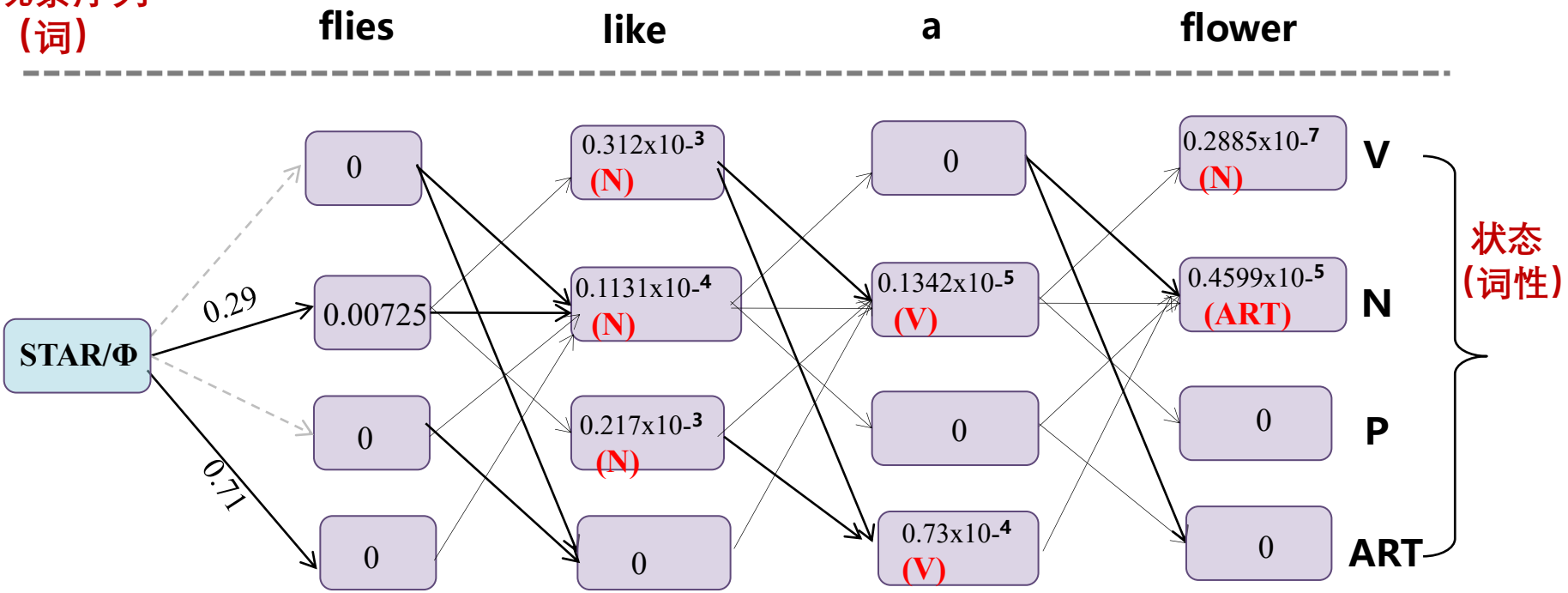
2. 递归： $\delta_t(j)=[\max_{1\le i\le N}\delta_{t-1}(i)a_{ij}]b_j(O_t)$， $2\le t\le T, 1\le j\le N$

   $\varphi_t(j)=\arg\max_{1\le i\le N}[\delta_{t-1}(i)a_{ij}]b_j(O_t)$， $2\le t\le T, 1\le j\le N$

3. 终结： $p^*=\max_{1\le i\le N}[\delta_T(i)]$ ， $q_T^*=\arg\max_{1\le i\le N}[\delta_T(i)]$

4. 路径回溯： $q_t^*=\varphi_{t+1}(q_{t+1}^*)$， $t=T-1,T-2,\ldots,1$

观察序列（词）



词性转移概率

词性	估计
PROB(ART | ø)	0.71
PROB(N | ø)	0.29
PROB(N | ART)	1
PROB(V | N)	0.43
PROB(N | N)	0.13
PROB(P | N)	0.44
PROB(N | V)	0.35
PROB(ART | V)	0.65
PROB(ART | P)	0.74
PROB(N | P)	0.26

词语生成概率

PROB(*the* | ART)	0.54
PROB(*flies* | N)	0.025
PROB(*flies* | V)	0.076
PROB(*like* | V)	0.1
PROB(*like* | P)	0.068
PROB(*like* | N)	0.012
PROB(*a* | ART)	0.360
PROB(*a* | N)	0.001
PROB(*flower* | N)	0.063
PROB(*flower* | V)	0.05
PROB(*birds* | N)	0.076

Viterbi 搜索算法

1. 初始化： $\delta_1(i)=\pi_i b_i(O_1)$， $\varphi_1(i)=0$， $1\le i\le N$

2. 递归： $\delta_t(j)=[\max_{1\le i\le N}\delta_{t-1}(i)a_{ij}]b_j(O_t)$， $2\le t\le T, 1\le j\le N$

   $\varphi_t(j)=\arg\max_{1\le i\le N}[\delta_{t-1}(i)a_{ij}]b_j(O_t)$， $2\le t\le T, 1\le j\le N$

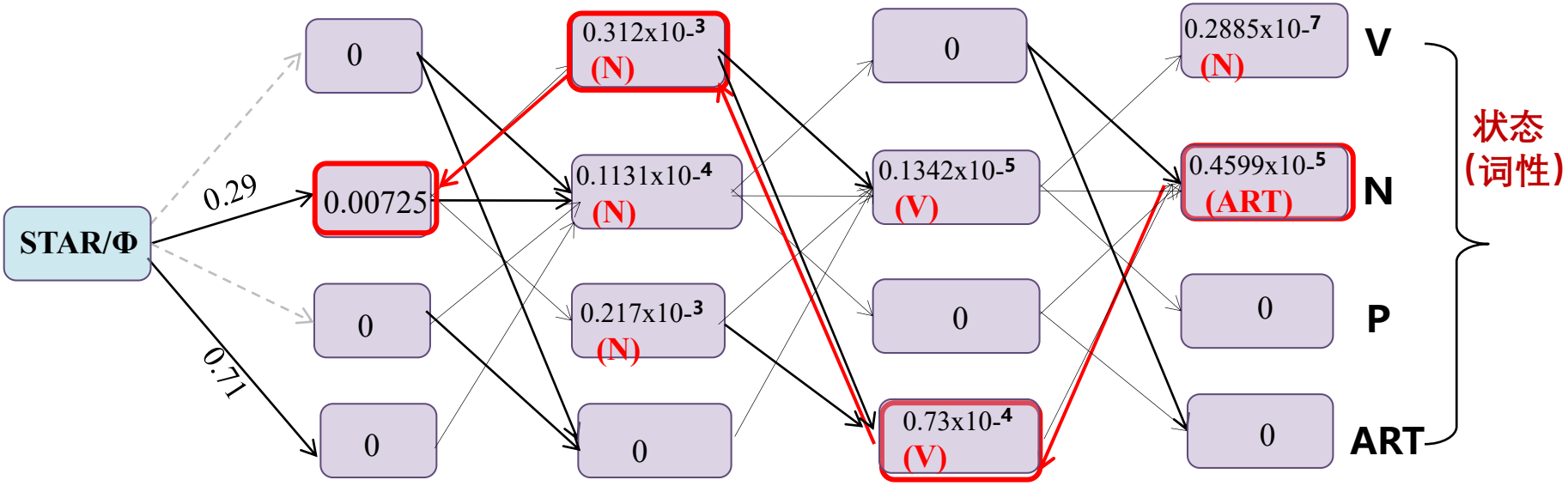
3. 终结： $p^*=\max_{1\le i\le N}[\delta_T(i)]$ ， $q_T^*=\arg\max_{1\le i\le N}[\delta_T(i)]$

4. 路径回溯： $q_t^*=\varphi_{t+1}(q_{t+1}^*)$， $t=T-1,T-2,\ldots,1$

观察序列（词）

flies /N　　like /V　　a /ART　　flower /N

结果



2. 隐马尔科夫模型

例2： 用HMM实现简单的中文分词

例 . 输入： 北京是中国的首都

输出：北京 是 中国 的 首都 (词序列)

解： 用单字序列标注方法

{ 词首/B，词内/I，词尾/E，单字词/O }

模型HMM:

S：状态集合， { B，I，E，O}

O：观察值集合，{单个汉字： 人、民、中.....}

A：状态转移概率矩阵

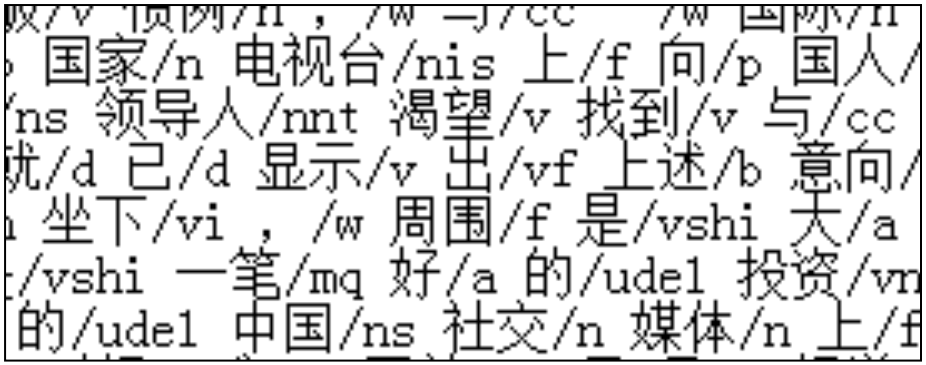
B：给定状态下，观察值的概率分布

π：初始状态空间的概率分布

2. 隐马尔科夫模型

参数学习

语料：



训练语料：

国/B 家/E 电/B 视/I 台/E 上/O 向/O 国/B人/E 领/B 导/I 人/E....

2. 隐马尔科夫模型

训练语料：

国/B 家/E 电/B 视/I 台/E 上/O 向/O
国/B人/E 领/B 导/I 人/E....

假设，语料中不重复的中文单字共8000个

- $A =$

	B	I	E	O
B	0	0.3	0.7	0
I	0	0.4	0.6	0
E	0.4	0	0	0.6
O	0.5	0	0	0.5

$A \in R^{4\times4}$，每行元素之和为1

- $B =$

	国	家	电	视	台	上	向	国	人	....
B	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	
I	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	....
E	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	....
O	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	....

$B \in R^{4\times8000}$，每行元素之和为1

- $\pi = [XXX, 0, 0, XXX]^T$ $\pi \in R^4$，元素之和为1

2. 隐马尔科夫模型

用最大似然估计学习参数：

有观察序列$O=O_1O_2...O_T$ 和 状态序列$Q=q_1q_2.....q_T$

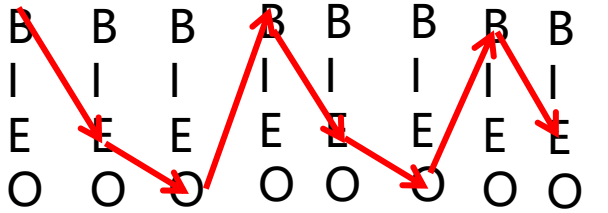
用极大似然估计

- $\pi_i = \frac{\sum_{t=1}^{T} \delta(q_t, S_i)}{T}$, $(S_0=\mathrm{B}, S_1=\mathrm{I}, S_2=\mathrm{E}, S_3=\mathrm{O})$
- $a_{ij} = \frac{\sum_{t=1}^{T-1} \delta(q_t, S_i) \times \delta(q_{t+1}, S_j)}{\sum_{t=1}^{T-1} \delta(q_t, S_i)}$
- $b_{jk} = \frac{\sum_{t=1}^{T} \delta(q_t, S_j) \times \delta(o_t, v_k)}{\sum_{t=1}^{T} \delta(q_t, S_j)}$

2. 隐马尔科夫模型

预测-分词

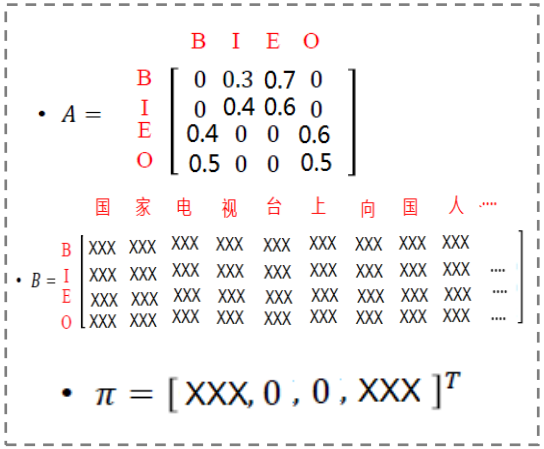
Viterbi算法



输入： 北 京 是 中 国 的 首 都

输出： B E O B E O B E

分词结果： 北京/ 是/ 中国/ 的/ 首都



注意： 分词和词性标注虽均用HMM模型， 但 状态集 观察集 不同，训练语料标注不同，模型参数不同

其它相关概率统计模型

最大熵模型：

$$p_\lambda(y|x) = \frac{1}{Z_\lambda(x)} \exp\left(\sum_i \lambda_i f_i(x,y)\right)$$

其中：

$$Z_\lambda(x) = \sum_y \exp\left(\sum_i \lambda_i f_i(x,y)\right)$$

称为归一化因子。

特点：可以综合上下文信息

条件随机场 CREF：

$$p(y|x,\lambda) = \frac{1}{Z(x)} \exp\left(\sum_{ji} \lambda_j t_j(y_{i-1}, y_i, x, i) + \sum_{ki} \mu_k s_k(y_i, x, i)\right)$$

其中：$$Z(x) = \sum_y \exp\left(\sum_{ji} \lambda_j t_j(y_{i-1}, y_i, x, i) + \sum_{ki} \mu_k s_k(y_i, x, i)\right)$$

特点：综合上下文信息

并且建立输出之间联系

详情略

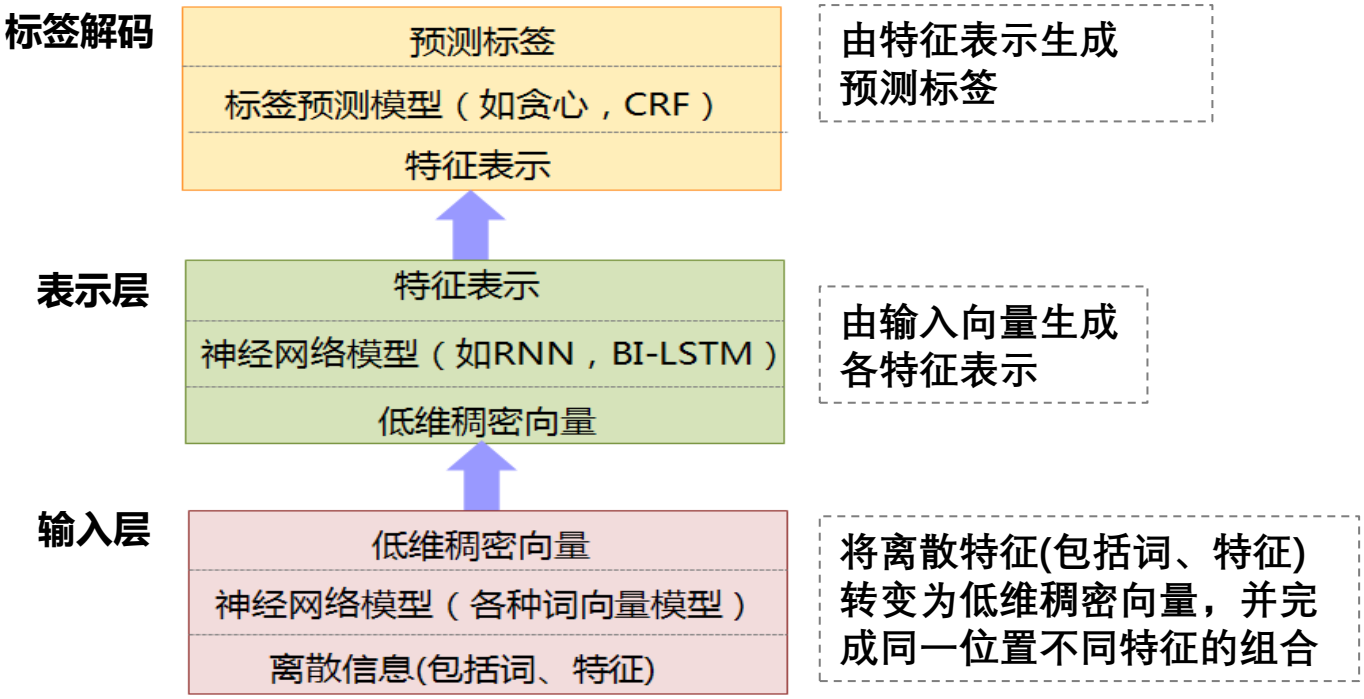
■ 序列标注

本节内容：

1. 序列标注问题概述
2. 隐马尔科夫模型HHM(概率统计模型)
3. 神经网络序列标注模型（深度学习模型）
   （1）双向RNN+softmax 模型
   （2）双向RNN+CRF 模型

(1) 双向RNN+softmax 模型：

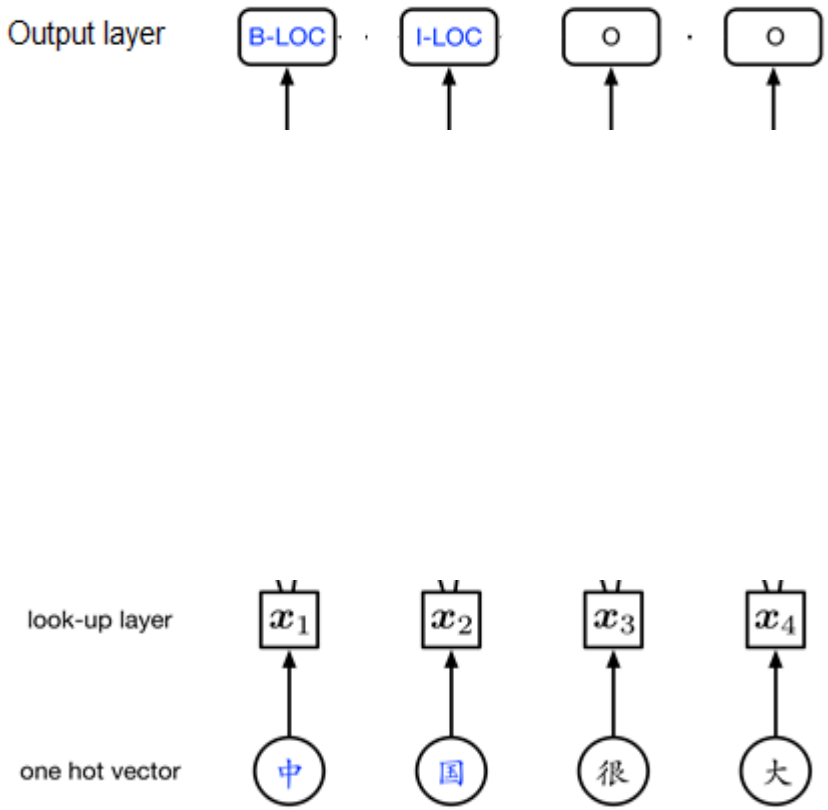
神经网络序列标注模型架构



(1) 双向RNN+softmax 模型：

(1) BiRNN+softmax 模型：

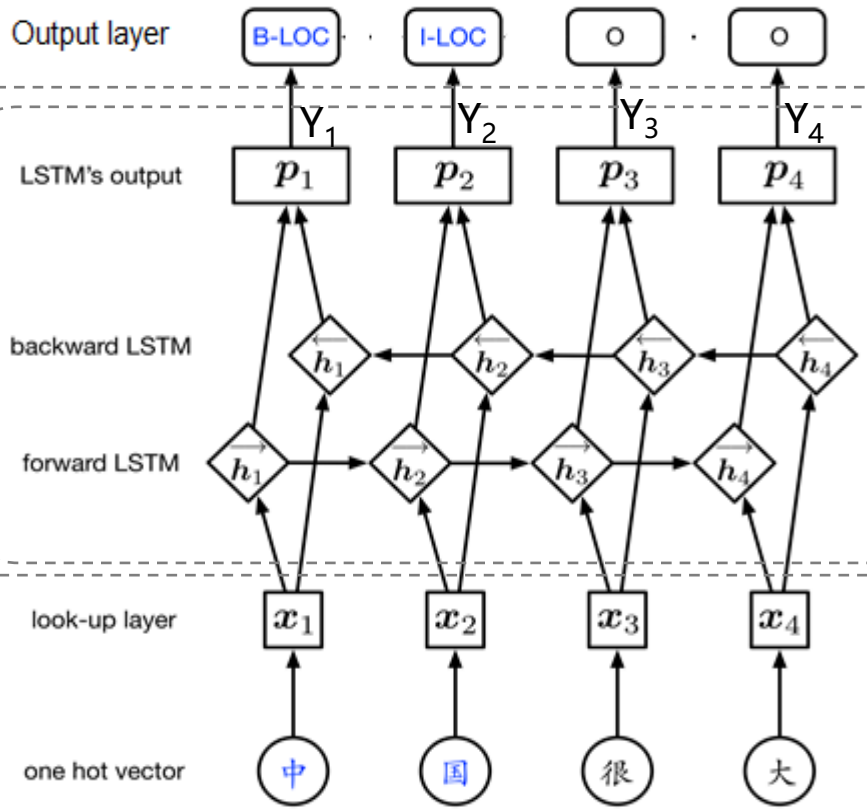
- **模型结构：**



(1) 双向RNN+softmax 模型：

(1) BiRNN+softmax 模型：

■ **模型结构：**



输出层： $Out_i = Max(y_i)$ 标签序列

$Y_i = softmax(p_i)$ (标签分布)

$p_i = \tanh(W_C[F(h_i), B(h_i)])$

反向隐藏层：

$$B(h_i) = \sigma((W2 \bullet X_i) + U2 \bullet B(h_{i+1}))$$

正向隐藏层：

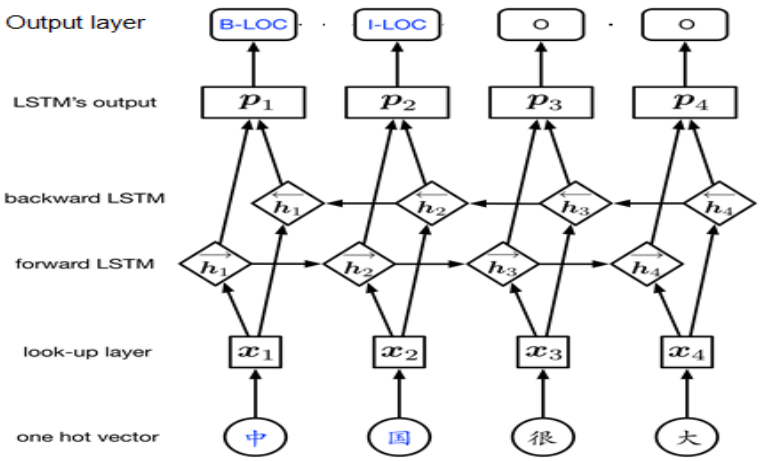
$$F(h_i) = \sigma((W1 \bullet X_i) + U1 \bullet F(h_{i-1}))$$

输入层： X_i ： 字 w_i 的字向量序列

参数： W1,U1, W2, U2 ,Wc,

(1) 双向RNN+softmax 模型：

■ 模型学习 (有监督)



$\hat{Y}$ 格式：(10000) (01000) (00000) (00100)

$\hat{Y}$: B-PER I-PER O B-Loc

X: 张 三 在 北 京

如 标人名： 训练数据（有标注训练集）

张/B-PER 三/I-PER 在/O 北/B-Loc 京/I-Loc 旅/O 游/O …..

标签集：{B-PER， I-PER， B-Loc， I-Loc， O}

(1) 双向RNN+softmax 模型：

■ 模型学习 (有监督)

- 定义损失函数

交叉熵损失：$J(\theta;x,y)=-\sum_{j=1}^{k} y_j \log((y_{pred})_j)$　　k 标签数

整体损失：$J(\theta)=\frac{1}{m}\sum_{i=1}^{m} J(\theta;x^{(i)},y^{(i)})$　　$\theta=\{W1,U1,W2,U2,Wc,\}$

- 用BPTT算法训练参数 W1,U1, W2, U2, Wc

■ 序列标注

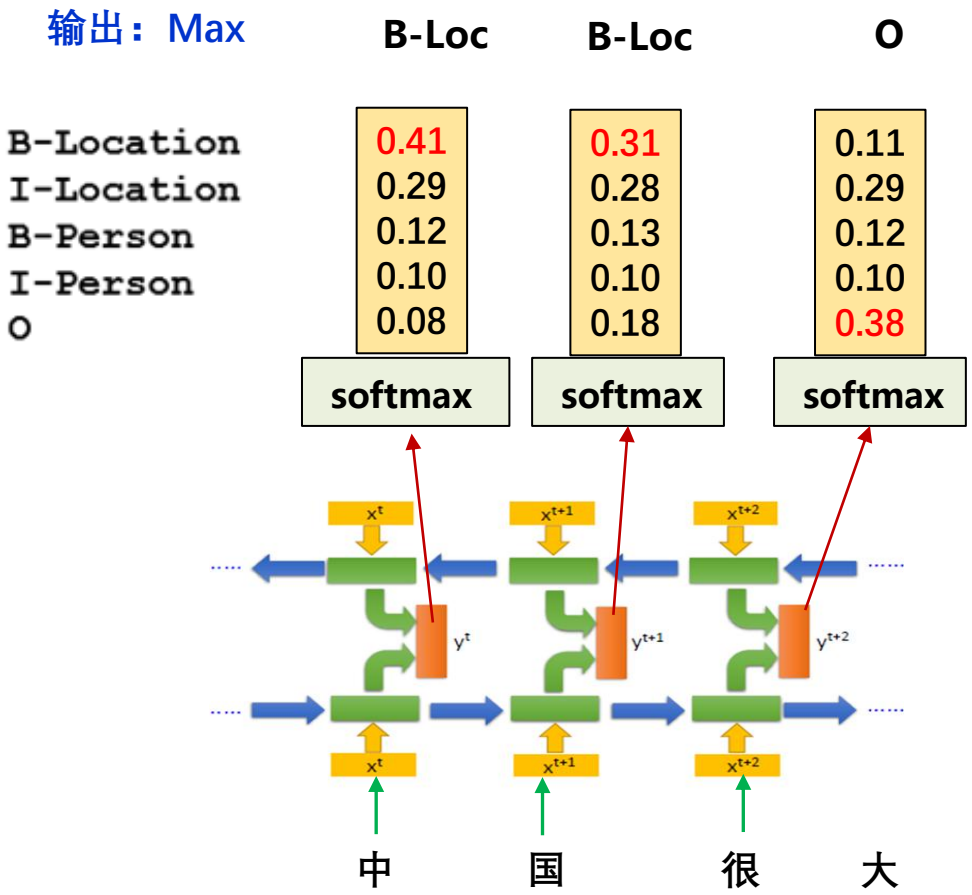
本节内容：

1. 序列标注问题概述
2. 隐马尔科夫模型HHM(概率统计模型)
3. 神经网络序列标注模型（深度学习模型）
   （1）双向RNN+softmax 模型
   （2）双向RNN+CRF 模型

(2) 双向RNN+CRF 模型

BiRNN+softmax 模型存在问题：

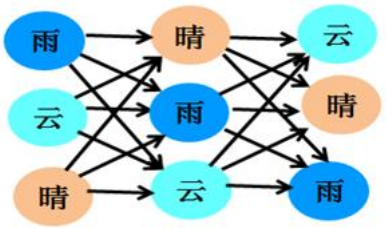
例如：



原因：输出独立

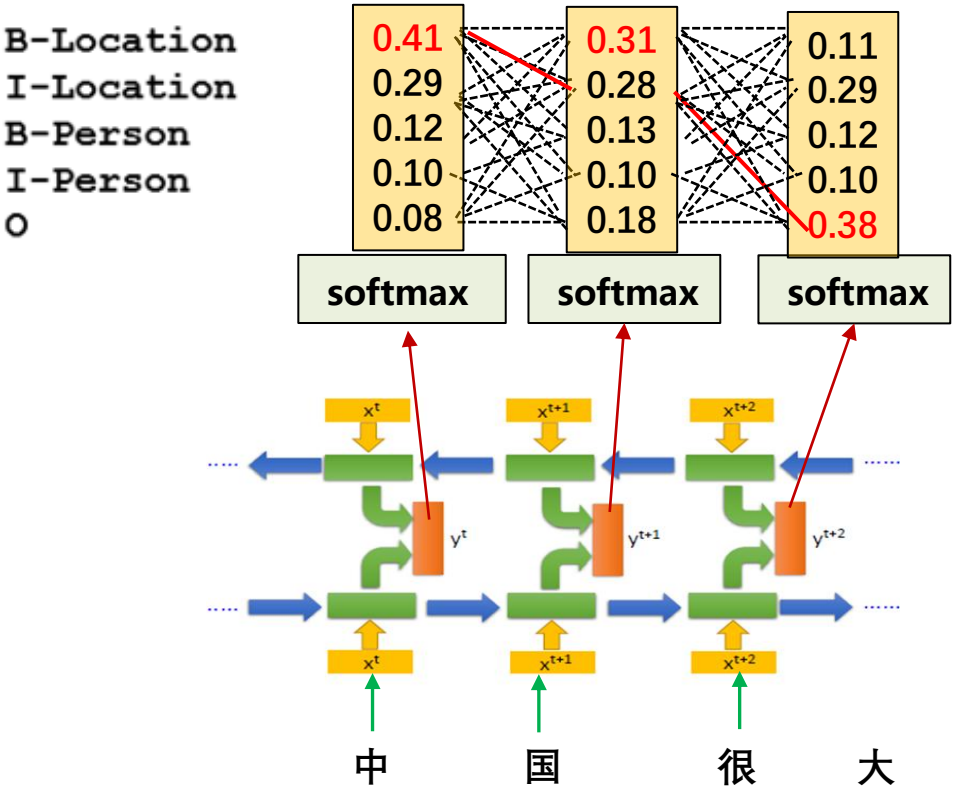
(2) 双向RNN+CRF 模型

对BiRNN+softmax 模型改进：



改进思路： 建立输出之间的关系

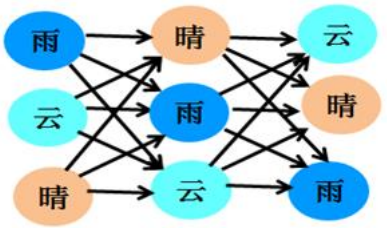
输出：概率最大的序列　B-Loc　I-Loc　O



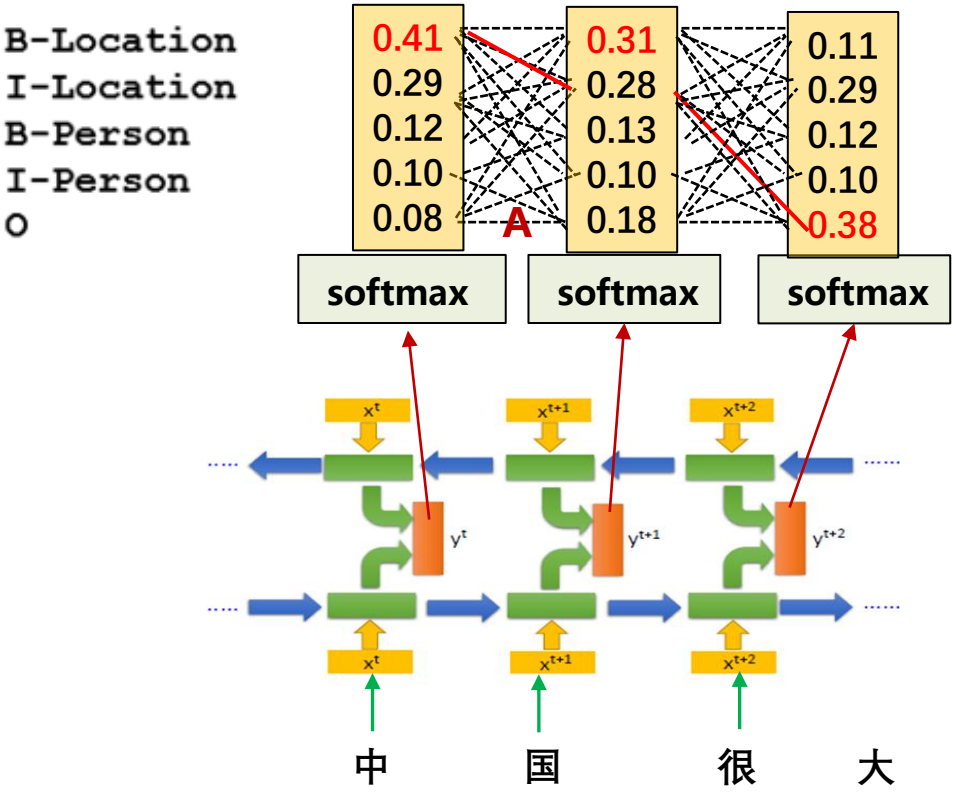
(2) 双向RNN+CRF 模型

对BiRNN+softmax 模型改进：

改进思路： 建立输出之间的关系

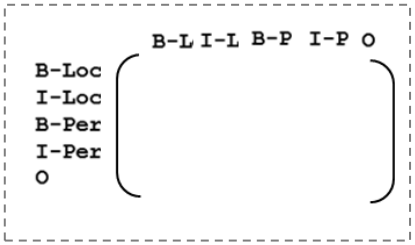


输出： 概率最大的序列　B-Loc　I-Loc　O



方法： 设一组参数A学习标签间的转移概率

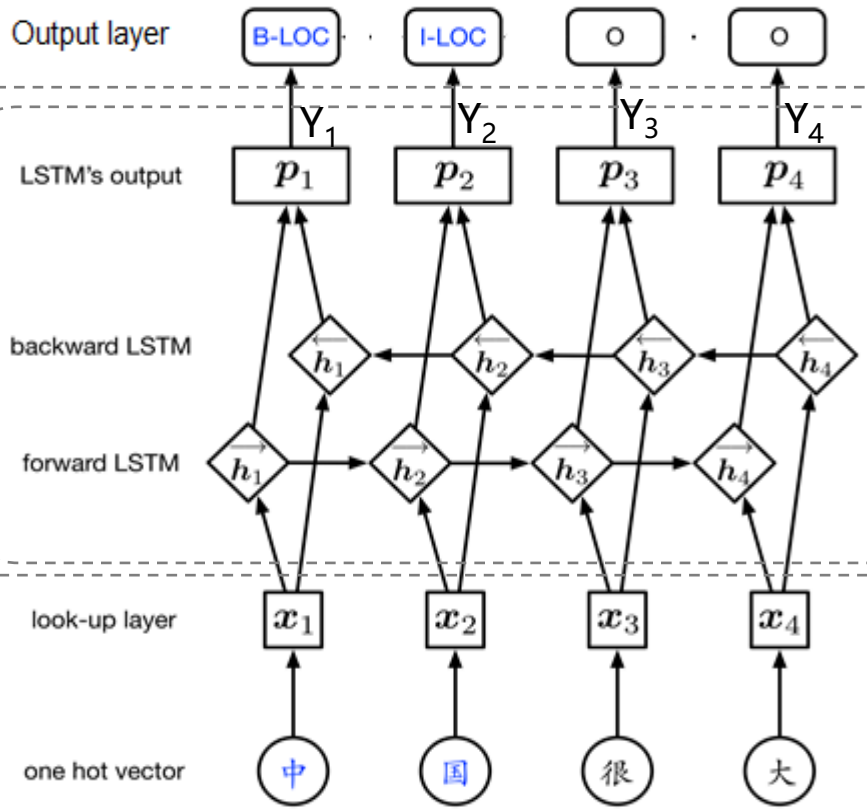
A： k× k方阵



K： 标签数

(2) 双向RNN+CRF 模型

如何改进BiRNN+softmax 模型（建立输出间联系）？



输出层： $Out_i = Max(y_i)$ 标签序列

$Y_i = softmax(p_i)$ (标签分布)

$p_i = \tanh(W_C[F(h_i), B(h_i)])$

反向隐藏层：

$$B(h_i) = \sigma((W2 \bullet X_i) + U2 \bullet B(h_{i+1}))$$

正向隐藏层：

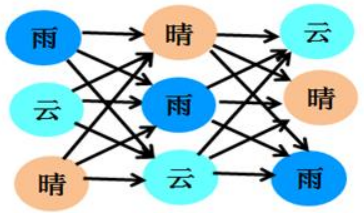
$$F(h_i) = \sigma((W1 \bullet X_i) + U1 \bullet F(h_{i-1}))$$

输入层： X_i ： 字 w_i 的字向量序列

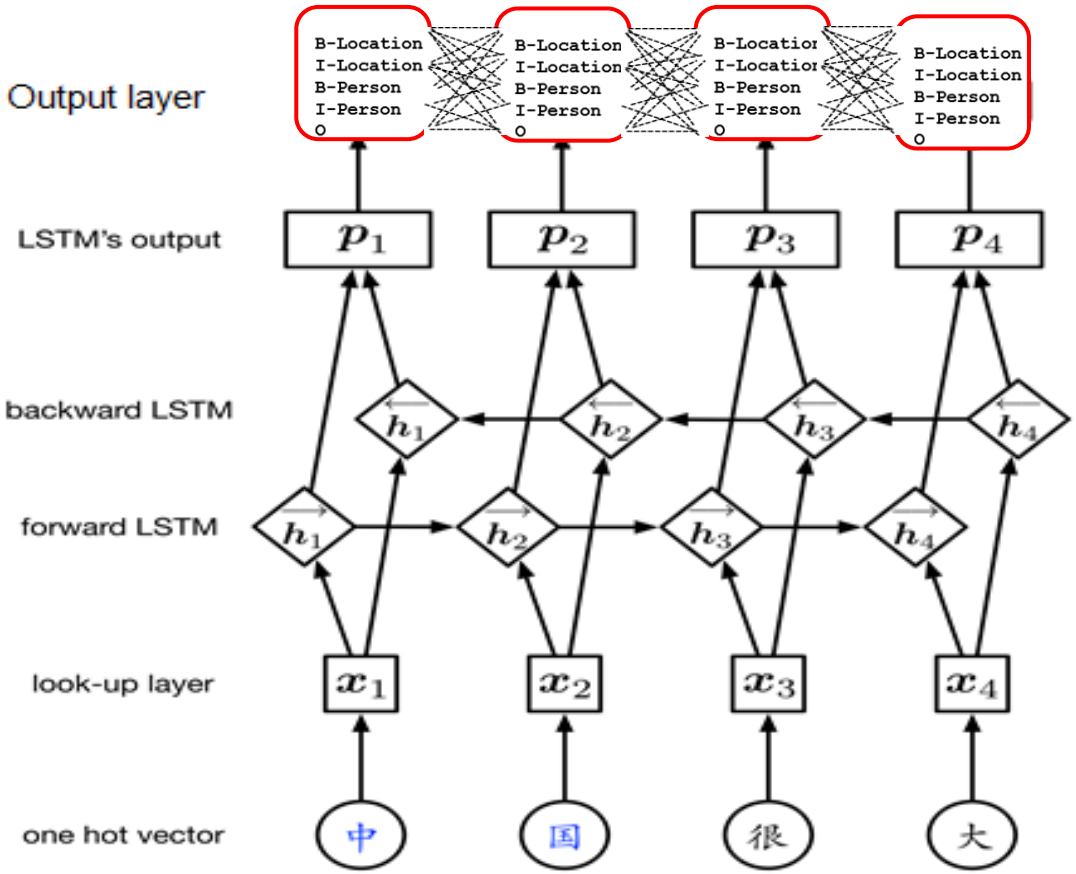
参数： W1,U1, W2, U2 ,Wc,

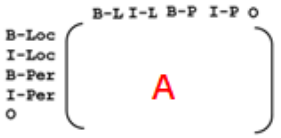
(2) 双向RNN+CRF 模型

如何改进BiRNN+softmax 模型（建立输出间联系）？



输出：概率最大的序列





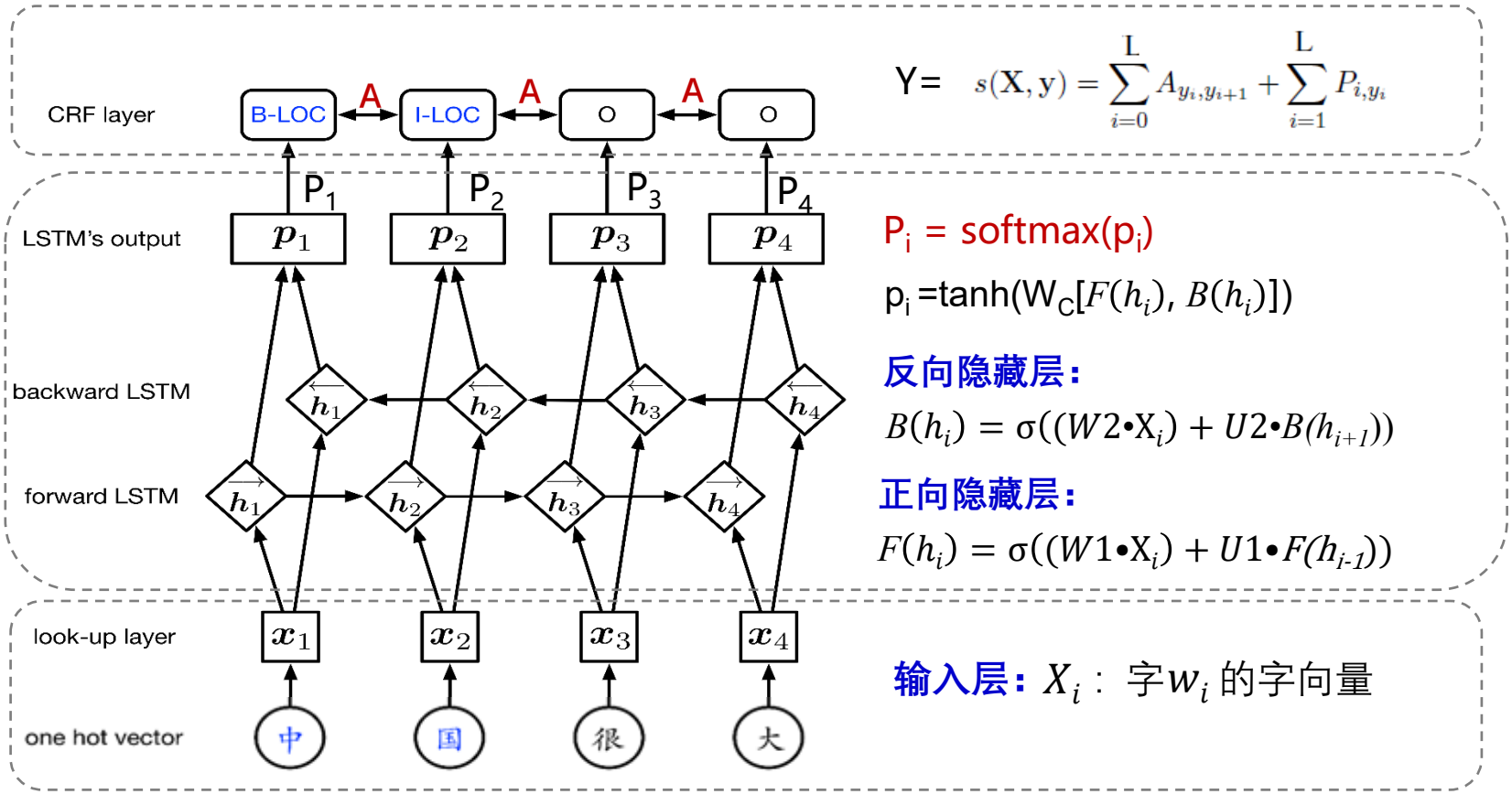
$$Y= \quad s(\mathrm{X},\mathrm{y}) = \sum_{i=0}^{\mathrm{L}} A_{y_i,y_{i+1}} + \sum_{i=1}^{\mathrm{L}} P_{i,y_i}$$

(2) 双向RNN+CRF 模型

(2) BiRNN+CRF 模型：

- **模型结构：**

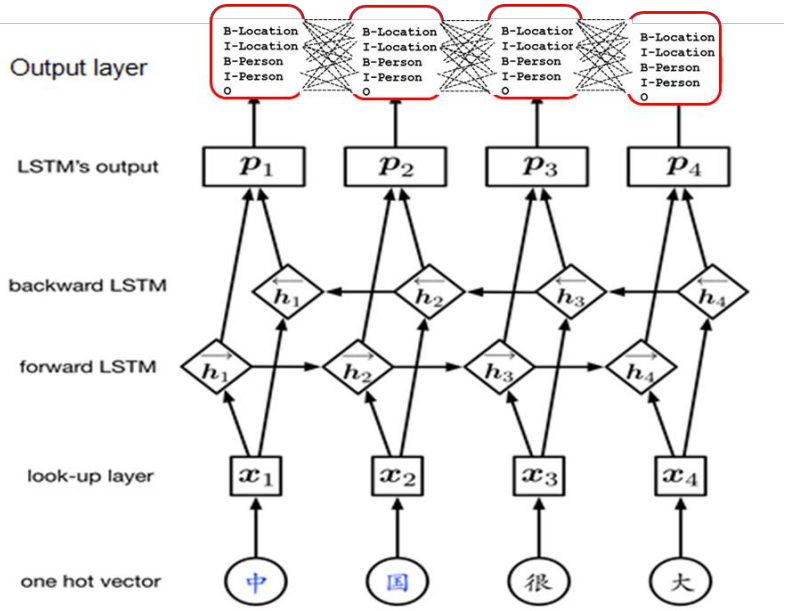
输出层： $y^* = \underset{\tilde{y} \in Y_X}{\operatorname{argmax}}\, s(X, \tilde{y})$.



参数： W1,U1, W2, U2 ,Wc, **A**

(2) 双向RNN+CRF 模型

■ 模型学习 (有监督)



$\hat{Y}$ 格式： （ 0 0 0 … 1 0 … ）

$\hat{Y}$: B-PER I-PER O B-Loc

X: 张 三 在 北 京

如 标人名： 训练数据（有标注训练集）

张/B-PER 三/I-PER 在/O 北/B-Loc 京/I-Loc 旅/O 游/O …..

标签集：{B-PER， I-PER， B-Loc， I-Loc， O}

(2) 双向RNN+CRF 模型

■ 模型学习 (有监督)

- 损失函数：交叉熵损失

- 优化目标 $p(\mathbf{y}|\mathbf{X}) = \frac{e^{s(\mathbf{X},\mathbf{y})}}{\sum_{\tilde{\mathbf{y}}\in\mathbf{Y_X}} e^{s(\mathbf{X},\tilde{\mathbf{y}})}}.$ （ $s(\mathbf{X},\mathbf{y}) = \sum_{i=0}^{\mathrm{L}} A_{y_i,y_{i+1}} + \sum_{i=1}^{\mathrm{L}} P_{i,y_i}$ ）

  最大化 $\log(p(\mathbf{y}|\mathbf{X})) = s(\mathbf{X},\mathbf{y}) - \log\left(\sum_{\tilde{\mathbf{y}}\in\mathbf{Y_X}} e^{s(\mathbf{X},\tilde{\mathbf{y}})}\right) = s(\mathbf{X},\mathbf{y}) - \underset{\tilde{\mathbf{y}}\in\mathbf{Y_X}}{\text{logadd}}\, s(\mathbf{X},\tilde{\mathbf{y}})$

  其中，$\mathrm{Y_X}$是所有可能 的输出序列

- 用BPTT算法训练参数 θ=[A,W1,U1, V1, W2, U2, V2 ,Wc]

■ 模型预测： $\mathbf{y}^* = \underset{\tilde{\mathbf{y}}\in\mathbf{Y_X}}{\text{argmax}}\, s(\mathbf{X},\tilde{\mathbf{y}}).$

内 容 提 要

6.1　文本分类

6.2　文本匹配

6.3　序列标注

6.4　序列生成

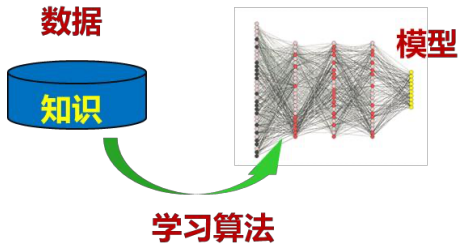
6.5　综合示例

6.4 序列生成

本节主要内容：

- 序列生成模型
  - 1 概述
  - 2 模型结构
    - 2.1 纯生成式模型
      - 基本RNN架构
      - RNN+注意力架构
      - Transformer架构
    - 2.2 选择式模型
      - 指针网(Pointer network)
    - 2.3 选择-生成式模型
      - PG-Net
  - 3 训练策略
    - 自编码
    - 对抗生成网络
    - 扩散生成模型
  - 4 序列生成模型评价指标
  - 5 序列生成模型存在问题

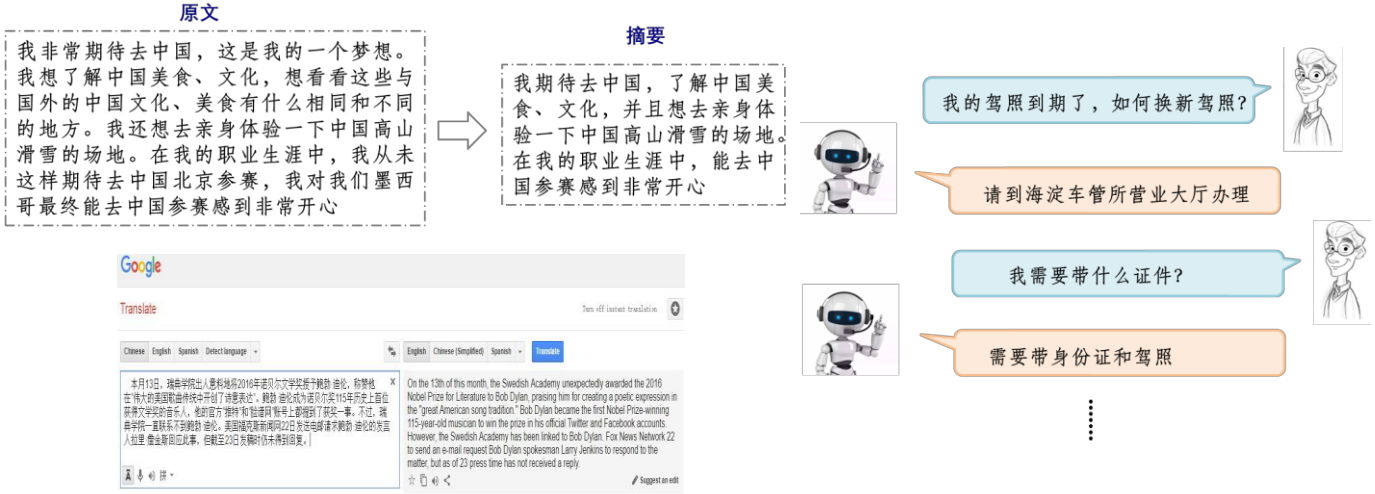
附录
- 基本问题
  - 词表问题
  - 解码问题
  - 训练问题



1. 概述

■ 序列生成

序列生成问题是NLP中常见和重要的研究内容，应用非常广泛，例如机器翻译，自动文摘、机器阅读理解、对话生成 、自动生成字幕等多项任务。



生成任务	输入	任务建模	输出
	文本序列	生成模型	文本序列

输入： X 序列　$x_t \in$ { 输入标识词典 }

输出： Y 序列　$y_t \in$ { 输出表示词典 }

1. 概述

■ 序列生成基本概念及术语

序列： 包含一系列标识符的有序列表，表示为 $y = (y_1, y_2, \cdots y_T)$ $y_t \in v$ 其中，y_t 是离散的值，V 代表输出序列标识词典，Y 可以是任意长序列，T 代表序列长度。

自回归序列生成: 用历史序列信息来预测序列中的下一个值的生成

$$P(y) = \sum_{t=1}^{r} P(y_t | y_{<t})$$

条件序列生成： 根据输入的内容 X 生成一串特定的序列 Y

$$P(y) = \sum_{t=1}^{r} P(y_t | y_{<t}, x)$$

有监督任务

其中，x 为输入文本，y 为生成文本，$P(y)$为生成 y 的概率，y_t为 t 步生成的词，$y_{<t}$为前t 步已生成的词序列，$P(y_t | y_{<t}, x)$ 表示y_t的生成概率，r为生成文本长度。x 和 Y 可以有不同的表示空间和标识词典

1. 概述

可控序列生成： 根据输入的内容X生成符合属性C的序列Y

$$P(y) = \sum_{t=1}^{r} P(y_t | y_{<t}, x, C)$$

其中，$P(y_t | y_{<t}, x, C)$ 表示y_t的生成概率，r为生成文本长度，C为要控制的本部属性。 如，关键实体，情感极性，主题，风格，人物角色等

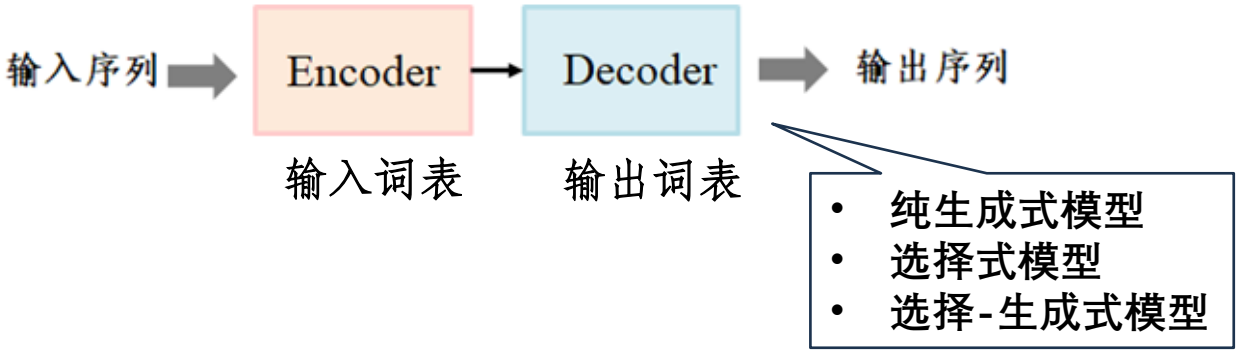
注： 常规的文本生成任务对生成文本的内容（content）没有强制性的约束，而受控文本生成任务会要求生成文本的内容必须满足一些既定的约束条件。如风格，主题等。受控文本生成更能满足实际需求

1. 概述

■ 生成模型建模 （Seq2Seq模型）

深度学习法建模：构建一个联合的神经网络，以端到端的方式将一个序列化数据映射成另一个序列化数据。简称 Sequence-to-Sequence Generation（Seq2Seq）模型。主流的Seq2Seq模型通常基于Encoder-Decoder框架实现

	输入	任务建模	输出
生成任务	文本序列	生成模型	文本序列



1. 概述

Seq2Seq模型按输出生成方式分为：

- **生成式模型Decoder：** 根据编码端形成的输入表示和先前时刻输出tokens，生成词表token的概率分布，并根据该分布产生当前输出词（编码端和解码端有各自词表，二者可相同或不同 。解码端需处理集外词OOV，一般用UNK 代替）

- **选择式模型Decoder：** 根据编码端形成的输入表示和先前时刻产生成的输出tokens，从输入端选择一个token作为输出 token （解码端和编码端词表相同）

- **选择-生成式模型Decoder：** 前两种方式结合，输出可以从输入中选择也可以 由 Decoder 端生成。（可处理输出端的OOV 问题）

总结

■ 基本任务类型归纳

分类任务	输入	任务建模	输出
	文本序列	分类模型	类别标签 类别标签根据任务定

匹配任务	输入	任务建模	输出
	二段文本序列	匹配模型	二者关系 一般类别标签

序列标注任务(方法)	输入	任务建模	输出
	非结构化文本序列	序列标注模型	标签序列 通过标签序列找问答答案

生成任务	输入	任务建模	输出
	文本序列	生成模型	文本序列

各类任务可统一为生成任务

生成任务	输入	任务建模	输出
	文本序列	生成模型	文本序列

问题：哪类模型可以完成所有的任务？

内 容 提 要

6.1　文本分类

6.2　文本匹配

6.3　序列标注

6.4　序列生成

6.5　综合示例

6.5 综合示例

■ 信息抽取任务

信息抽取: 从指定文档中或者海量文本中抽取出用户感兴趣的信息。

（1）实体识别与抽取

（2）关系抽取

（3）事件抽取

6.5 综合示例

（1）实体识别与抽取问题

6.5 综合示例

（1）传统实体识别与抽取

从指定非结构化文档中或者海量文本中抽取出简单实体构成结构化信息

非结构化文本

2011年7月25日，在上海举办的游泳世锦赛上，年仅15岁的叶诗文的以2分08秒90的成绩勇夺女子200米混合泳冠军，成为最年轻的单项世界冠军获得者。

抽取

结构化知识

时间：2011年7月25日

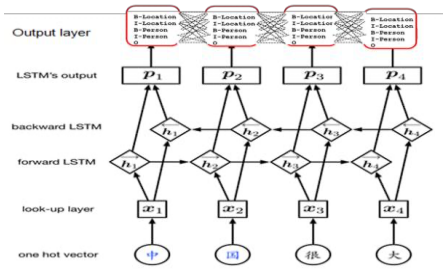
地名：上海

人名：叶诗文

任务：

	输入	任务建模	输出
序列标注任务(方法)	非结构化文本序列	序列标注模型	标签序列 通过标签序列找问答答案

模型：

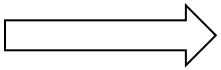


（2）实体关系抽取

实体关系抽取（关系抽取）：识别实体之间的语义关系，即从非结构化文本即纯文本中抽取实体关系三元组。（关系抽取是构建知识图谱非常重要的一环）

比尔·盖茨作为微软公司的创始人，以他的创新思维和领导力改变了整个科技行业。他在推动科技进步方面取得了巨大成就

实体间关系抽取



1. 实体：比尔盖茨, 微软

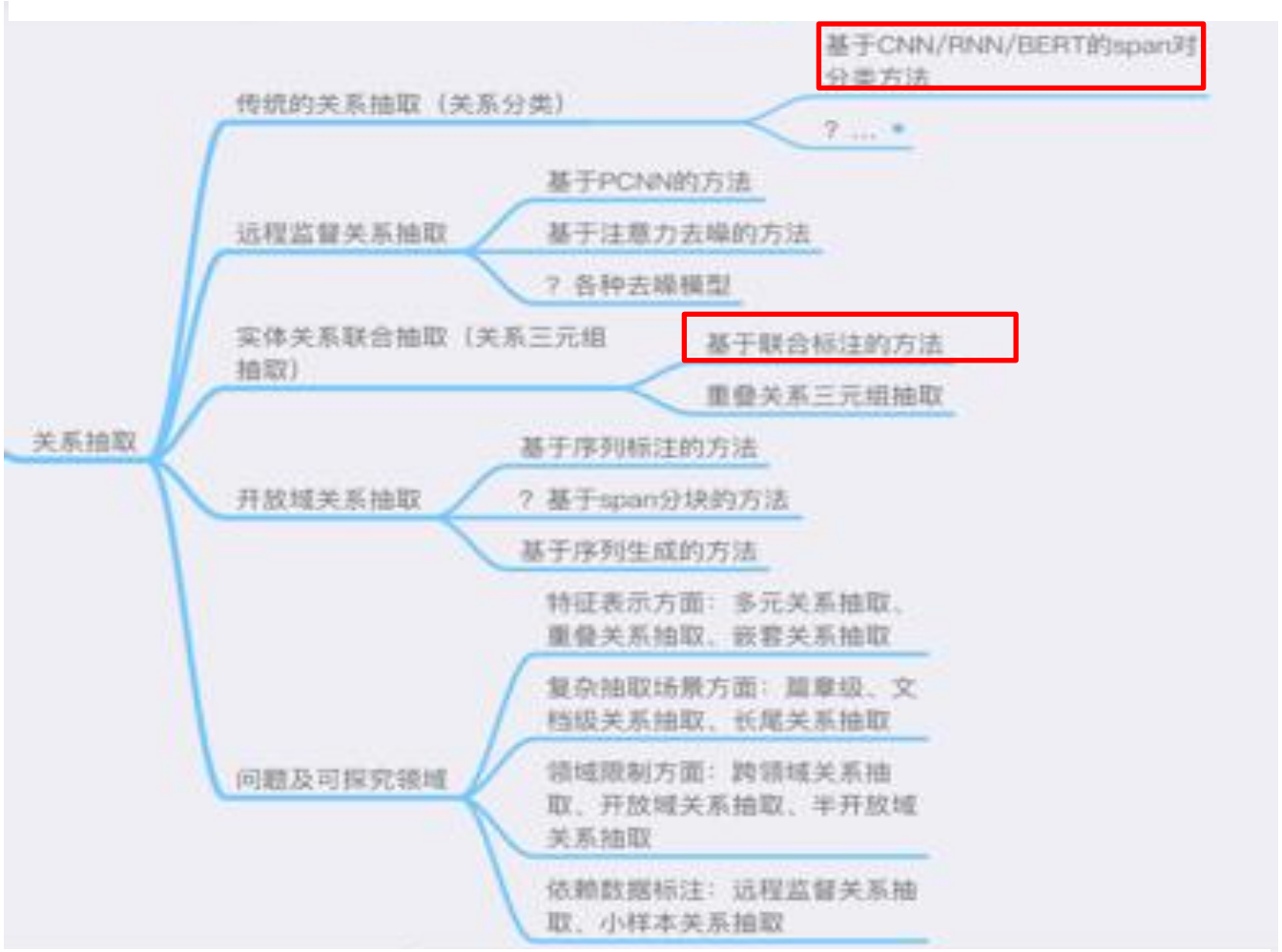
2. 比尔盖茨和微软的关系：

CEO(比尔盖茨, 微软)

关系分类：通常可以看做是一个多分类任务

6.5 综合示例

（2）实体关系抽取方法



(2) 实体关系抽取方法

方法一：流水线方法

Step1：抽取实体

Step2：实体间关系分类

例： 早期比尔·盖茨作为微软公司的创始人，以他的创新思维和领导力改变了整个科技行业。他在推动科技进步方面取得了巨大成就

Step1：抽取实体（序列标注）

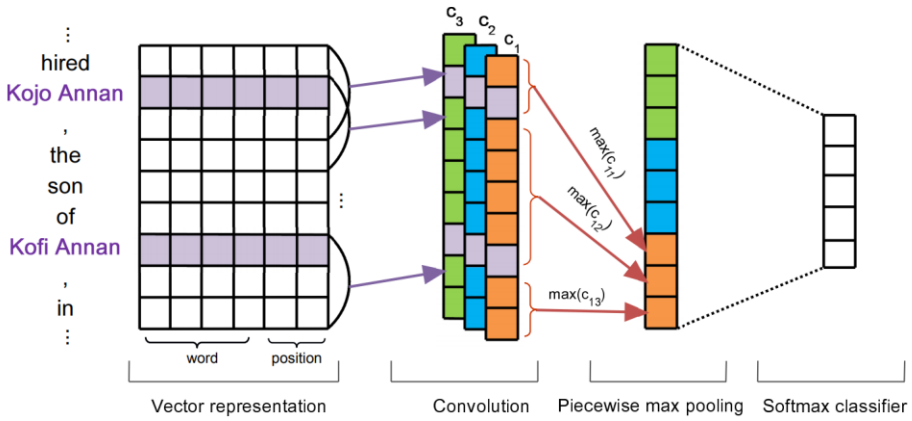
早期比尔·盖茨作为微软公司的创始人，以他的创新思维和领导力改变了整个科技行业。他在推动科技进步方面取得了巨大成就

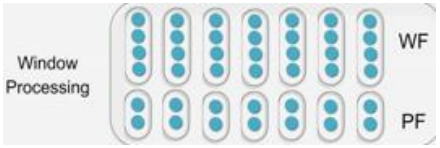
6.5 综合示例

Step2：实体间关系分类　　关系 ∈{关系类别集合}

输入： 早期比尔·盖茨作为微软公司的创始人

输出： CEO (比尔盖茨, 微软)





$$[\mathrm{WF}, \mathrm{PF}]^T$$

- WF（word）：词向量
- PF(position)：句子中每个词到两个实体的距离[d1, d2].

如：　早期比尔·盖茨作为微软公司的创始人

PF：　　作为：[1,-1]

任务涉及： 标注+分类

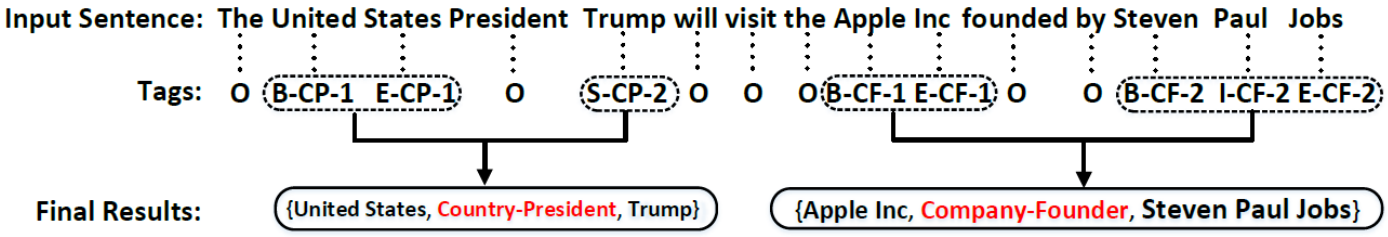
方法二：联合抽取方法

基本思想： 实体识别和关系抽取同时建模。

方法： 在实体标注中加入关系的类型信息 和 实体在关系中主/客体信息

格式： B/I/E/S/O - 关系类型 - 主/客体

其中，B：实体开始 I：中间 E：结尾 S：单个实体；关系类型：定义好的关系类型 (如，CP:Country-President、CF:Company-Founder)；1：在关系中作为主语 2：在关系中作为宾语；



当一个句子中有两个或更多的三元组有相同的关系类型，那么根据就近原则组合实体

(该方法适用一个实体只属于一个三元组的情况)

任务涉及： 标注

Zheng, Suncong, et al. "Joint extraction of entities and relations based on a novel tagging scheme."

6.5 综合示例

（3）事件抽取任务

事件抽取（Event Extraction, EE）：从非结构化文本中识别特定事件的信息，并将其转化为结构化形式。其核心目标是捕捉现实世界中事件发生的“谁、何时、何地、什么、为什么”等关键要素

(3) 事件抽取任务

句子级事件抽取方法

事件抽取步骤：

1. 触发词检测：抽取描述事件发生的核心动词或名词并确定其类别。
2. 确定事件论元集合：对指定事件确定需要抽取哪些角色
3. 抽取/识别事件论元：抽取参与事件的实体及其角色（如“参与者”“时间”“地点”等）

如： 毛泽东1893年出生于湖南湘潭

事件：
Trigger: 出生
Type: Life, Subtype: Be-Born
Person: 毛泽东
Time: 1893年
Place: 湖南湘潭

本节复习

- 文本分类
- 文本匹配
- 序列标注
- 文本生成

大作业

- 学习基于深度学习的NER模型，写学习报告（包含代码结构、模型结构等），并设计一个实际场景调用
  - https://github.com/thunlp/OpenNRE
- 思考：大模型时代传统NLP还有没有价值?

致 谢

- 胡玥、曹亚男、方芳：国科大《自然语言处理基础》
- 曹亚男、任昱冰：国科大《深度学习与自然语言处理概述》


THANKS
https://ictkc.github.io/teaching/2026spring-nlp