

NLP常用基础模型

Essential Models

王博 天津大学智能与计算学部 2019.12

机器学习角度：

序列标注：分词、词性标注、名实体识别、问答...

分类：文本分类、情感分析、问答...

序列转换（串到串、生成）：翻译、语音识别、自动摘要、对话、问答

序列比较：语义匹配、相似度分析、推理分析、问答

机器学习角度：

序列标注： HMM, MEMM, CRF, RNN

分类： ME, SVM, CNN

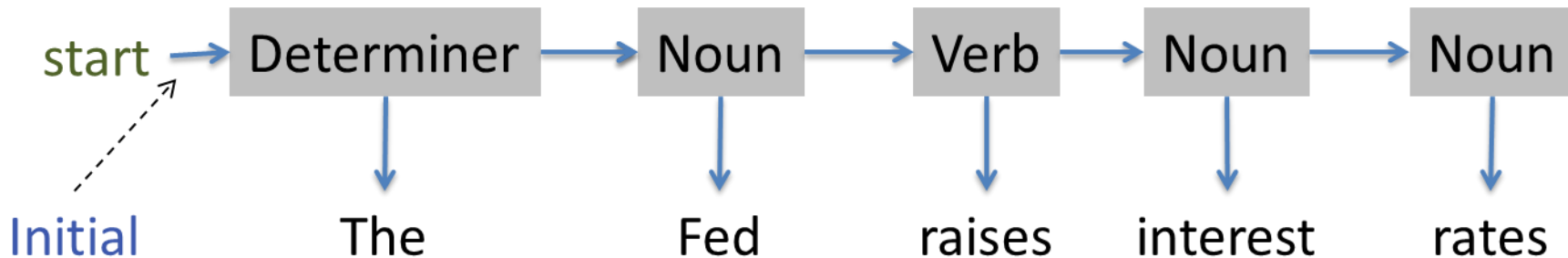
序列转换（串到串、生成）： RNN

序列比较： 编辑距离、语义编码（RNN等），语义编码+分类

EM算法：猜测隐藏在文字背后的信息

最大期望算法 (Expectation-maximization algorithm, 又译为期望最大化算法), 是在概率模型中寻找参数最大似然估计或者最大后验估计的算法, 其中概率模型依赖于无法观测的隐性变量。

通俗的表述: 观察结果依赖于隐藏状态。只能看到观察结果, 看不到隐藏状态。如何知道隐藏状态生成观察结果的概率 (模型参数) ?



➤ EM (expectation maximization)



要求出每一种硬币投掷时正面向上的概率

➤ EM (expectation maximization)

硬币	结果	统计
A	正正反正反	3正-2反
B	反反正正反	2正-3反
A	正反反反反	1正-4反
B	正反反正正	3正-2反
A	反正正反反	2正-3反

- 统计期望

$$P(A=\text{正}) = (3+1+2) / 15 = 0.4 \quad P(B=\text{正}) = (2+3) / 10 = 0.5$$

➤ EM (expectation maximization)

硬币	结果	统计
Unknown	正正反正反	3正-2反
Unknown	反反正正反	2正-3反
Unknown	正反反反反	1正-4反
Unknown	正反反正反	3正-2反
Unknown	反正正反反	2正-3反

如已知 $P(A=\text{正}) = 0.4$ $P(B=\text{正}) = 0.5$

- **MLE(Maximum likelihood estimation)**

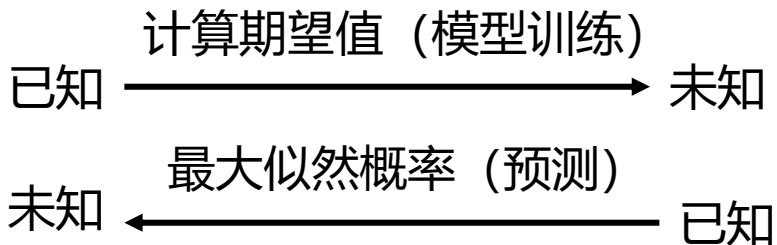
估计硬币序列为 B A A B A

➤ EM (expectation maximization)

观察值已知



隐藏状态



模型的参数

➤ EM (expectation maximization)

观察值已知



隐藏状态

未知



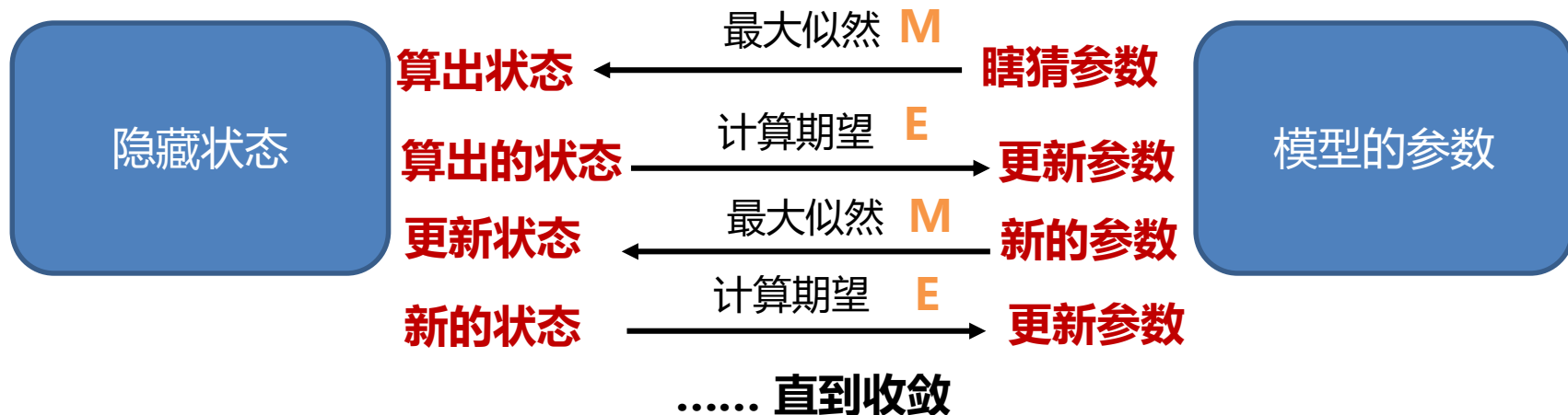
未知

模型的参数

如果只知道训练样本观察值，样本的隐藏状态和模型参数都不知道怎么办？

➤ EM (expectation maximization)

观察值已知



EM算法

➤ EM (expectation maximization)



如果我们不知道每一次投掷用的是哪一种硬币呢

硬币	结果	统计
Unknown	正正反正反	3正-2反
Unknown	反反正正反	2正-3反
Unknown	正反反反反	1正-4反
Unknown	正反反正正	3正-2反
Unknown	反正正反反	2正-3反

隐变量

➤ EM (expectation maximization)

基本思想:

- M步: 先初始化一个 $P(A), P(B)$, 然后我们拿着这个初始化的 $P(A), P(B)$ 用最大似然概率估计出每次的硬币类型;
- E步: 接下来, 知道每次的硬币类型之后就可以利用期望值计算 $P(A), P(B)$;
- 然后不断地重复E步和M步, 直到 $P(A), P(B)$ 收敛。

➤ EM (expectation maximization)

一开始随机设定 $P(A)=0.2$, $P(B)=0.7$

硬币	结果	统计
Unknown	正正反正反	3正-2反
Unknown	反反正正反	2正-3反
Unknown	正反反反反	1正-4反
Unknown	正反反正正	3正-2反
Unknown	反正正反反	2正-3反

➤ EM (expectation maximization)

一开始随机设定 $P(A)=0.2$, $P(B)=0.7$

M步

轮数	若是硬币A	若是硬币B
1	0.00512, 即0.2 0.2 0.2 0.8 0.8, 3正-2反	0.03087, 3正-2反
2	0.02048, 即0.2 0.2 0.8 0.8 0.8, 2正-3反	0.01323, 2正-3反
3	0.08192, 即0.2 0.8 0.8 0.8 0.8, 1正-4反	0.00567, 1正-4反
4	0.00512, 即0.2 0.2 0.2 0.8 0.8, 3正-2反	0.03087, 3正-2反
5	0.02048, 即0.2 0.2 0.8 0.8 0.8, 2正-3反	0.01323, 2正-3反

$Z=(B,A,A,B,A)$

➤ EM (expectation maximization)

$$P(A) = (2+1+2) / 15 = 0.33 \quad P(B) = (3+3) / 10 = 0.6$$

E 步

轮数	若是硬币A	若是硬币B
1	0.00512, 即0.2 0.2 0.2 0.8 0.8, 3正-2反	0.03087, 3正-2反
2	0.02048, 即0.2 0.2 0.8 0.8 0.8, 2正-3反	0.01323, 2正-3反
3	0.08192, 即0.2 0.8 0.8 0.8 0.8, 1正-4反	0.00567, 1正-4反
4	0.00512, 即0.2 0.2 0.2 0.8 0.8, 3正-2反	0.03087, 3正-2反
5	0.02048, 即0.2 0.2 0.8 0.8 0.8, 2正-3反	0.01323, 2正-3反

$$Z=(B,A,A,B,A)$$

➤ EM (expectation maximization)

$$P(A) = (2+1+2) / 15 = 0.33$$

$$P(B) = (3+3) / 10 = 0.6$$

M 步

轮数	若是硬币A	若是硬币B
1	0.00512, 即0.2 0.2 0.2 0.8 0.8, 2正2反	0.03087, 3正-2反
2	0.02048, 即0.2 0.2 0.2 0.8 0.8, 2正-3反	0.01323, 2正-3反
3	0.08192, 即0.2 0.8 0.8 0.8 0.8, 1正-4反	0.00567, 1正-4反
4	0.00512, 即0.2 0.2 0.2 0.8 0.8, 3正-2反	0.03087, 3正-2反
5	0.02048, 即0.2 0.2 0.8 0.8 0.8, 2正-3反	0.01323, 2正-3反

循环E、M步直到收敛

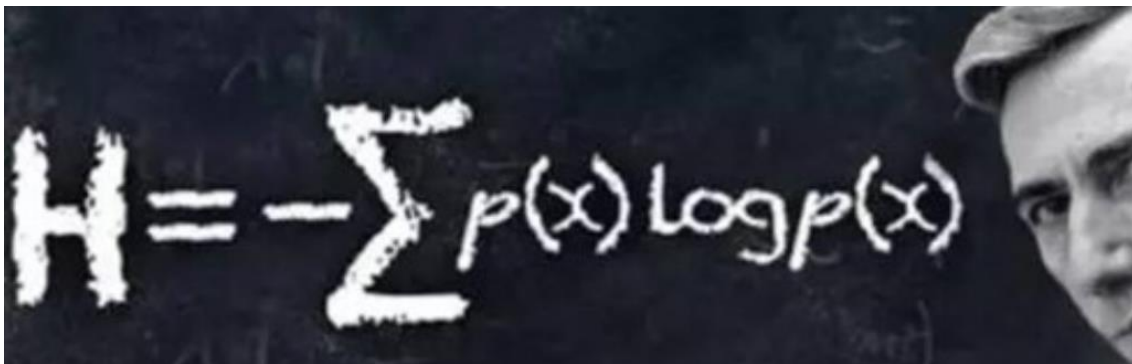
$Z=(\dots\dots\dots)$

最大熵模型：用特征去束缚语言的任意性

➤ ME (Maximum Entropy)

最大熵

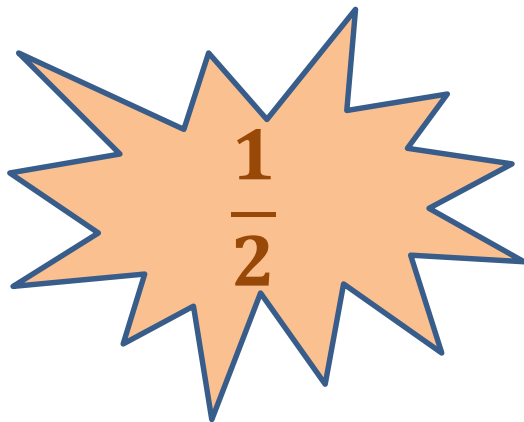
----- “信息熵”，用来描述信息的不确定程度


$$H = -\sum p(x) \log p(x)$$

➤ ME (Maximum Entropy)

最大熵

? 如果抛掷一枚硬币，你认为正面朝上和反面朝上的概率分别是多少？



➤ ME (Maximum Entropy)

最大熵

假设世界上有三种职业，请问任意一个人，每种职业的概率是多大？



$1/3$



$1/3$



$1/3$

我知道这个人是男性，而男性人口中医生和工程师一共占 $4/5$ 。那么他是每种职业的概率多大？



$2/5$



$2/5$

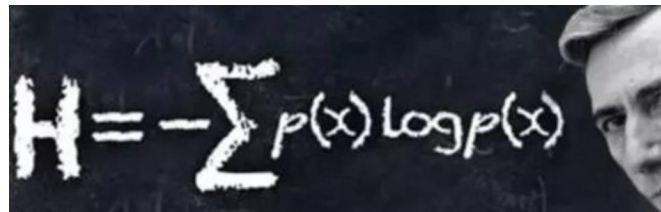


$1/5$

➤ ME (Maximum Entropy)

- **最大熵**：最大熵这个词听起来很深奥，但它的原理很简单，我们每天都在用。说白了，就是要保留全部的不确定性，把风险降到最小。
- **最大熵原理**：最大熵原理指出，需要对一个随机事件的概率分布进行预测时，我们的预测应当**满足全部已知的条件**，而**对未知的情况不要做任何主观假设**。在这种情况下，概率分布最均匀，预测的风险最小。因为这时概率分布的信息熵最大，所以人们称这种模型叫“最大熵模型”。

➤ ME (Maximum Entropy)


$$H = -\sum p(x) \log p(x)$$



如果抛掷一枚硬币，你认为正面朝上和反面朝上的概率分别是多少？

$$P(\text{正面}) = p, P(\text{背面}) = 1 - p$$

$$H(P) = -[p \log p + (1 - p) \log (1 - p)]$$

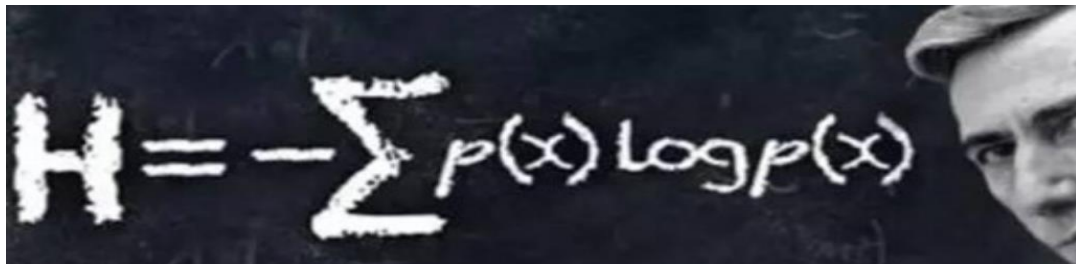
$$\text{Max} H(P): H(P)' = 0$$

$$\log p + 1 - \log(1 - p) - 1 = 0$$

$$p = 1/2$$



➤ ME (Maximum Entropy)


$$H = -\sum p(x) \log p(x)$$



1/3



1/3



1/3

在给定输入的情况下，计算输出概率，我们实际上在计算以输入为条件的条件熵

条件熵：

$$H(P) = - \sum_{x,y} \tilde{P}(x) P(y|x) \log P(y|x)$$



2/5



2/5



1/5

➤ ME (Maximum Entropy)

- (t1) “抓住”: The mother takes her child by the hand. 母亲抓住孩子的手。
- (t2) “拿走”: Take the book home. 把书拿回家。
- (t3) “乘坐”: to take a bus to work. 乘坐公共汽车上班。
- (t4) “量”: Take your temperature. 量一量你的体温。
- (t5) “装”: The suitcase wouldn't take another thing. 这个衣箱不能装别的东西了。
- (t6) “花费”: It takes a lot of money to buy a house. 买一所房子要花一大笔钱。
- (t7) “理解、领会”: How do you take this package? 你怎么理解这段话?

在没有任何限制的条件下，最大熵原理认为翻译成任何一种解释都是等概率的（如同硬币正反面都是1/2）。

$$p(t1|x)=p(t2|x)=\dots\dots=p(t7|x)=1/7$$

如果我们增加一个特征：如果下一个词是“bus”，则 $p(t3|x)=4/5$ 。此时，最大熵的结果就是在满足这个约束的情况下，剩余6种语义评分1/5的概率。

$$p(t3|x)=4/5 \quad p(t1|x)=p(t2|x)=p(t4|x)=\dots\dots=p(t7|x)=1/30$$

➤ ME (Maximum Entropy)

求解带约束（特征函数）的最优化问题

$$\max_{P \in \mathcal{C}} -H(P) = \sum_{x,y} \tilde{P}(x) P(y|x) \log P(y|x)$$

$$s.t. E_P(f_i) - E_{\tilde{P}}(f_i) = 0, i = 1, 2, \dots, n$$

$$\sum_y P(y|x) = 1$$

在下面的约束下求熵最大的输出分布：

所有特征函数的模型期望等于要求的期望

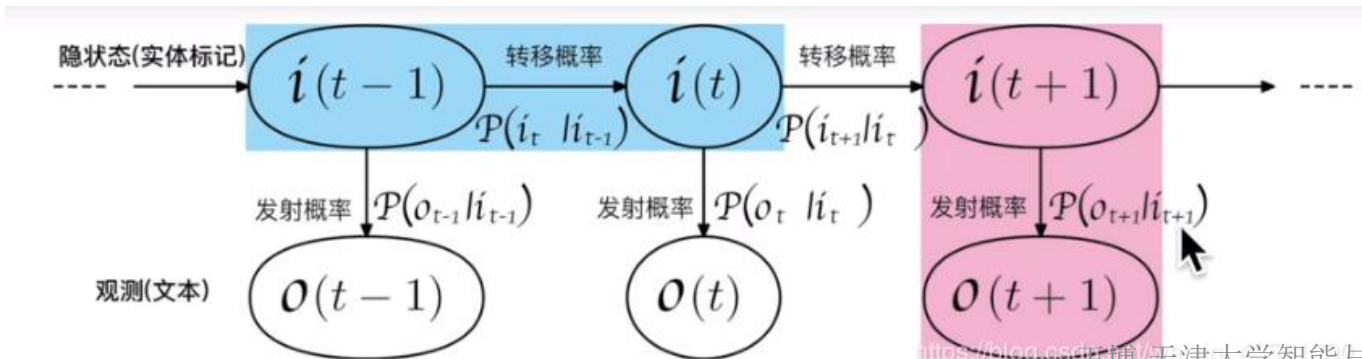
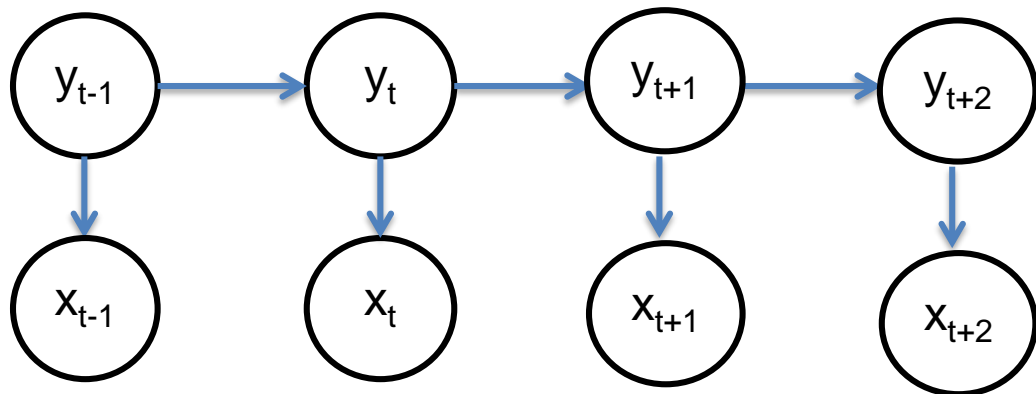
引入拉格朗日乘子，定义拉格朗日函数，转化为特征加权和

$$\max_{\lambda} \psi(\lambda) = \sum_{i=1}^n \lambda_i E_{\tilde{P}}(f_i) - \sum_x \tilde{P}(x) \log Z_{\lambda}(x)$$

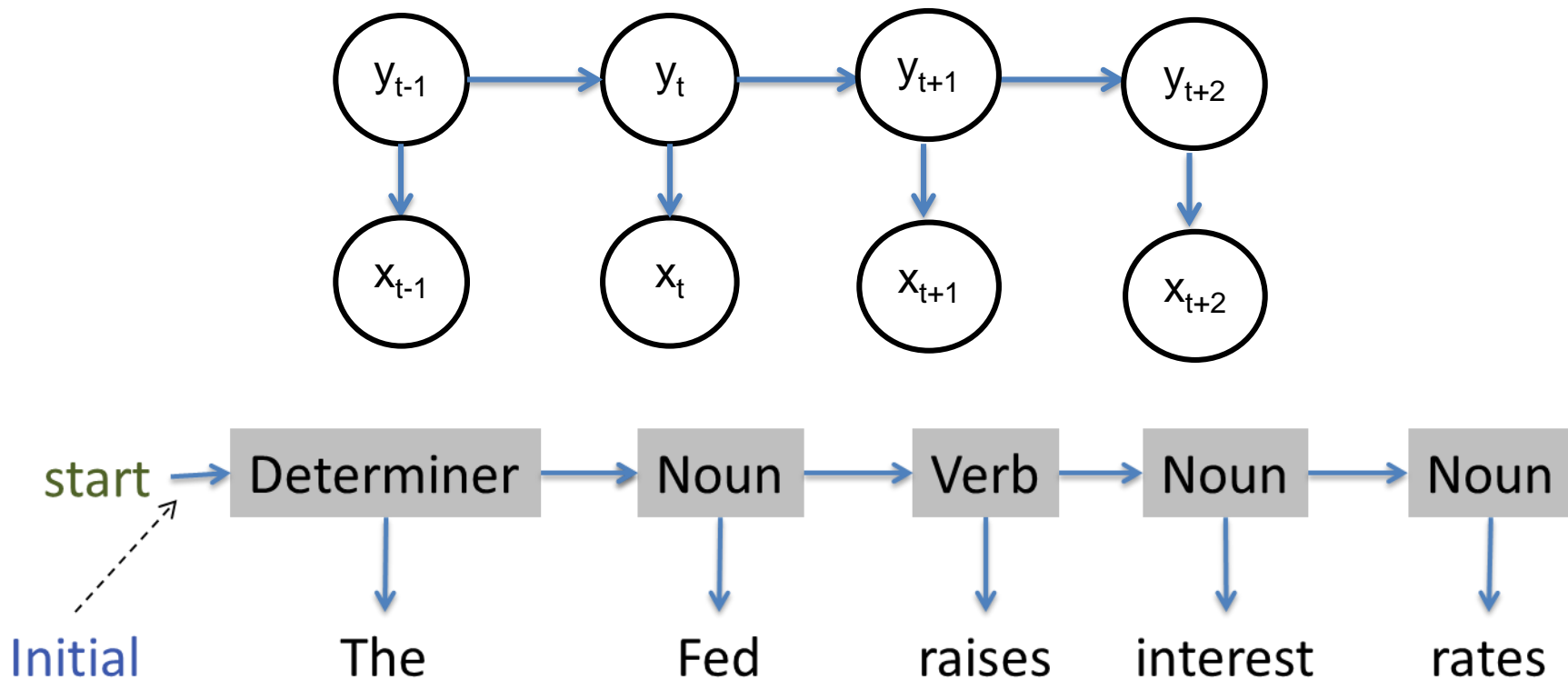
隐马尔可夫链：语言是一个串

隐马尔可夫模型 HMM

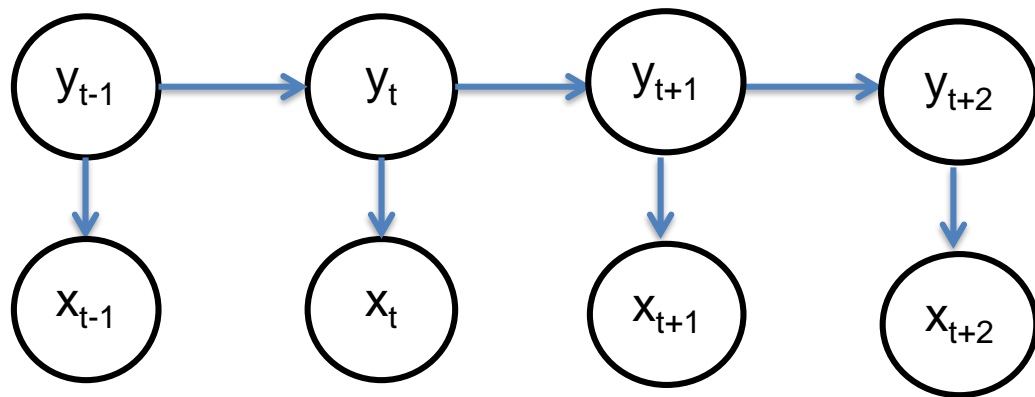
隐马尔可夫链：一个隐状态序列产生一个观察值序列。每个隐状态依赖于前一个隐状态



隐马尔可夫模型 HMM



隐马尔可夫模型 HMM



HMM的模型参数:

- 状态种类数量 = K , 观察值种类数量 = M
- π : 每种状态做为初始状态的概率 (K dimensional vector)
- A : 状态之间的转移概率 Transition probabilities ($K \times K$ matrix)
- B : 状态到观察值的发射概率 Emission probabilities ($K \times M$ matrix)

HMM 的三个问题

1. 估计问题：给定观察序列 x_1, x_2, \dots, x_n 和模型参数 (π, A, B) , 观察序列的概率有多大？（根据语言模型判断一句话是不是人话）

Estimation

2. 序列问题：给定观察序列 x_1, x_2, \dots, x_n 和模型 (π, A, B) , 最可能的状态序列是什么？（根据语言模型给一句话的每个词进行类别标注）

Inference

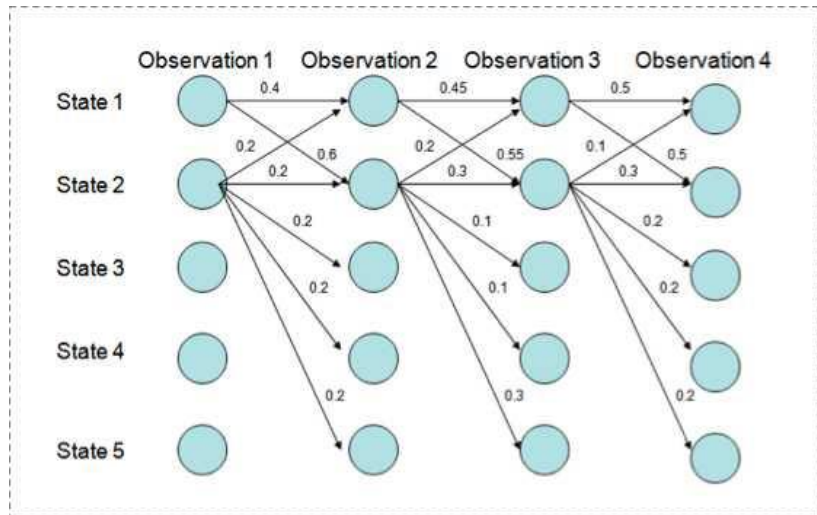
3. 训练问题：给定观察序列 x_1, x_2, \dots, x_n , 最佳的模型参数 (π, A, B) 是什么？（训练获得语言模型）

Learning

HMM 的三个问题

1. 估计问题：给定观察序列, x_1, x_2, \dots, x_n 和模型参数 (π, A, B) , 观察序列的概率有多大？（根据语言模型判断一句话是不是人话）

Estimation



在左图中从左至右高效遍历所有可能的状态序列，并计算每个状态序列产生观察序列的概率，最后求和即可。

左图利用动态规划优化算法（向前算法）高效解决上述问题，复杂度 $O(N^2T)$, N 为所有可能的状态数， T 为序列长度。

HMM 的三个问题

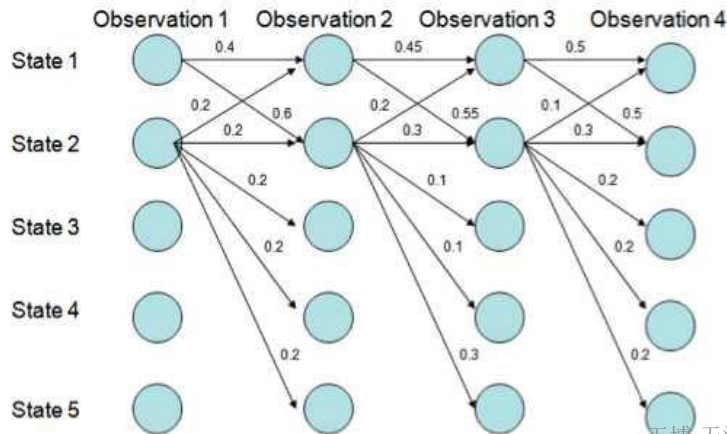
2. 序列问题：给定观察序列 x_1, x_2, \dots, x_n 和模型 (π, A, B) , 最可能的状态序列是什么？（根据语言模型给一句话的每个词进行类别标注）

Inference

$$\arg \max_{\mathbf{y}} P(\mathbf{y} | \mathbf{x}, \pi, A, B)$$

同样在左图中从左至右高效搜索能以最大概率产生观察序列的状态序列。

左图利用动态规划优化算法（向前算法）高效解决上述问题，复杂度 $O(N^2T)$, N 为所有可能的状态数， T 为序列长度。



HMM 的三个问题

3. 训练问题：给定观察序列 x_1, x_2, \dots, x_n ，最佳的模型参数 (π, A, B) 是什么？
(训练获得语言模型)

有监督的情形：指导观察序列对应的状态值，直接对训练语料进行统计计数即可，即最大似然

$$\pi_s = \frac{\text{count}(\text{start} \rightarrow s)}{n}$$

$$A_{s',s} = \frac{\text{count}(s \rightarrow s')}{\text{count}(s)}$$

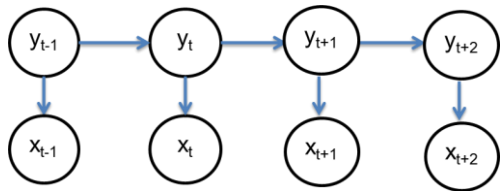
$$B_{s,x} = \frac{\text{count} \left(\begin{array}{c} s \\ \downarrow \\ x \end{array} \right)}{\text{count}(s)}$$

HMM 的三个问题

3. 训练问题：给定观察序列 x_1, x_2, \dots, x_n ，最佳的模型参数 (π, A, B) 是什么？
(训练获得语言模型)

无监督的情形： 不知道观察序列对应的状态值，只指导可能的状态集合

EM算法！



- 随机初始化 (π, A, B)
- M步：解HMM的第二问题（序列问题）求出状态序列
- E步：基于求出的状态序列重新计算 (π, A, B) （和有监督的参数估计相同，计数即可），然后回到M步。
- 不断迭代直到收敛。

生成与判别：纵观全局还是聚焦一处

机器学习（包括NLP）有两大类模型：**生成式模型** vs. **判别式模型**

HMM是典型的生成式模型，而最大熵是判别式模型。

生成模型：学习得到联合概率分布 $P(x,y)$ ，即特征 x 和标记 y 共同出现的概率，然后求条件概率分布。能够学习到数据生成的机制。

判别模型：学习得到条件概率分布 $P(y|x)$ ，即在特征 x 出现的情况下标记 y 出现的概率。

生成式与判别式模型

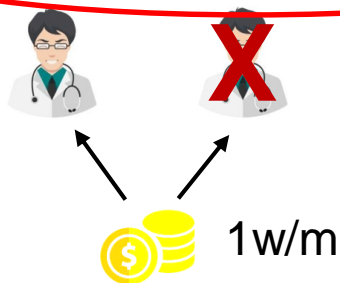
我想做一个模型知道一个人是不是医生，以收入为特征。

判别式：输入收入，输出一个人是医生概率（局部概率分布）？ 不需要知道其他职业的概率

生成式：首先得到每种收入值和每个职业的共现概率（完整的联合概率密度），看看指定输入和医生的概率是所有职业中最大？

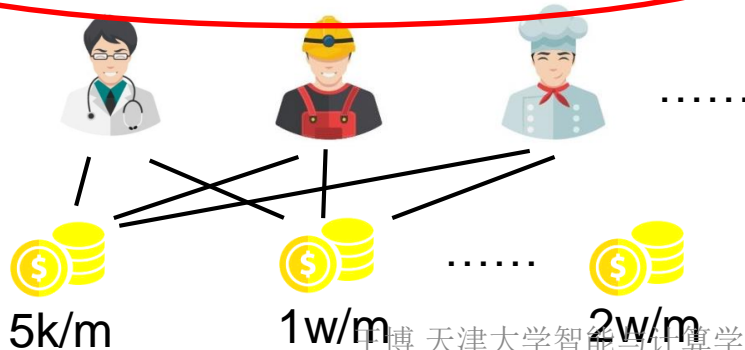
归一

$$P(\text{医生}|1w/m) = 0.6 \quad P(\text{非医生}|1w/m) = 0.4$$



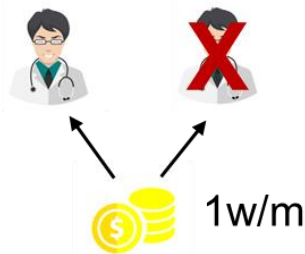
归一

$$\begin{aligned} P(\text{医生}, 1w/m) &= 0.3 & P(\text{工程师}, 1w/m) &= 0.2 & \dots\dots \\ P(\text{医生}, 5k/m) &= 0.1 & P(\text{工程师}, 5k/m) &= 0.2 & \dots\dots \end{aligned}$$



生成式与判别式模型

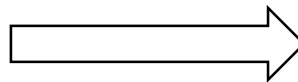
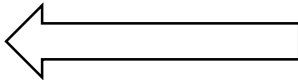
$$P(\text{医生}|1\text{w/m})=0.6 \quad P(\text{非医生}|1\text{w/m})=0.4$$



优点：所需数据量小，计算量小，对单一类别判定准确率高。可随意增加新特征。

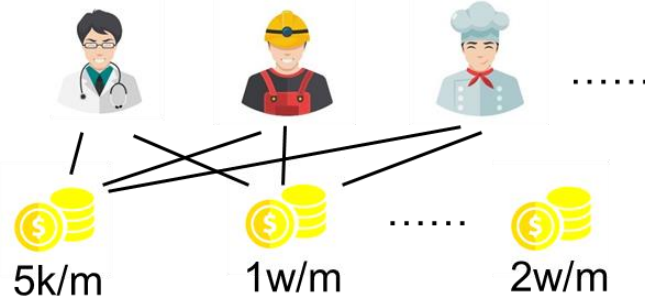
缺点：无法全局优化，应用范围受限。

已知生成模型可以得到各个判别模型



已知判别模型无法的到生成模型，除非已知所有可能的判别关系。

$$\begin{aligned} P(\text{医生}, 1\text{w/m}) &= 0.3 & P(\text{工程师}, 1\text{w/m}) &= 0.2 & \dots\dots \\ P(\text{医生}, 5\text{k/m}) &= 0.1 & P(\text{工程师}, 5\text{k/m}) &= 0.2 & \dots\dots \end{aligned}$$



优点：信息全面，可实现全局优化。

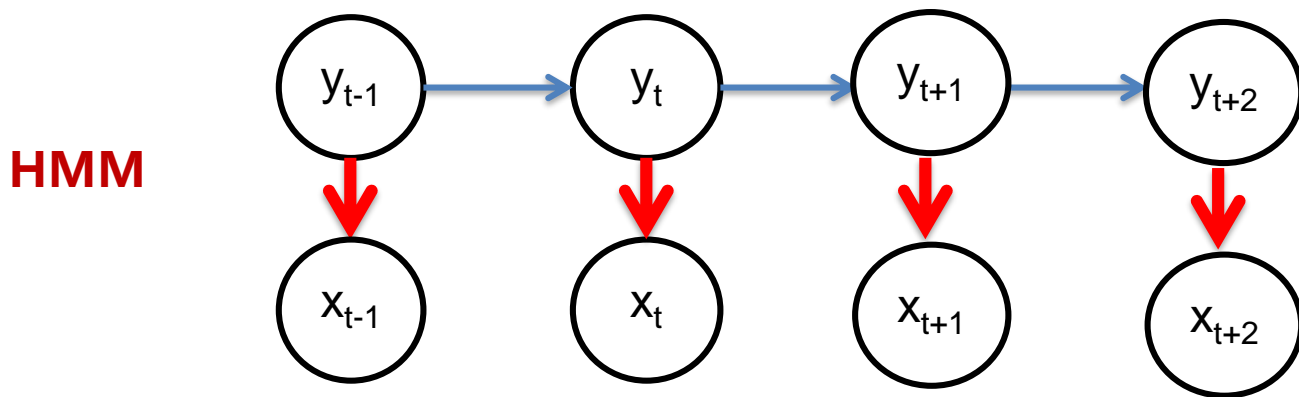
缺点：所需数据量大，计算量大，增加新特征的成本高。

为什么HMM是生成式模型？ 建模所有状态之间的转移关系和状态与所有词汇的发射关系

MEMM: 最大熵 + HMM

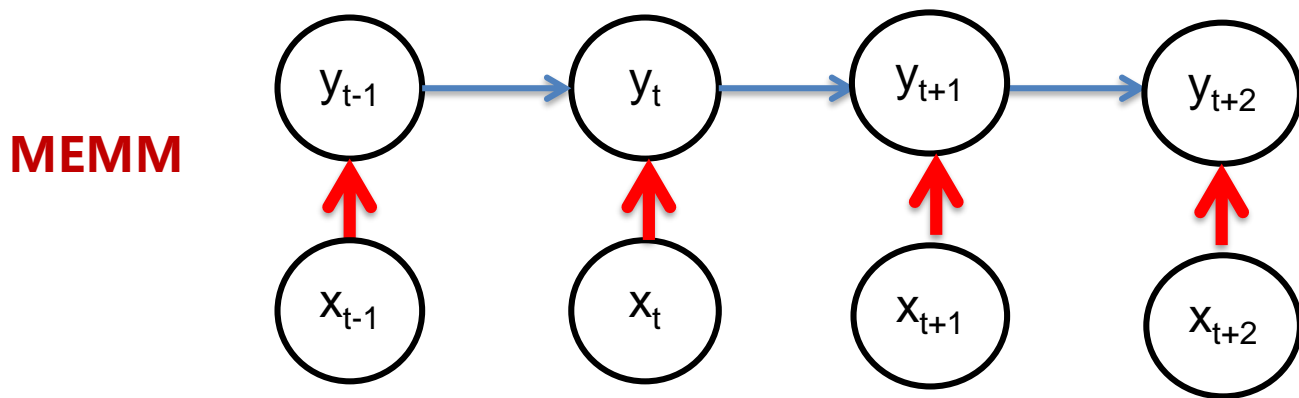
最大熵隐马 MEMM

在解决序列标注问题时，HMM的输入信息只有参数(π , A , B)和观察序列（即每个状态对应的文字），**没有办法接受更丰富的特征**（例如更多的上下文文字等）。为了解决这个问题，提出了MEMM模型。



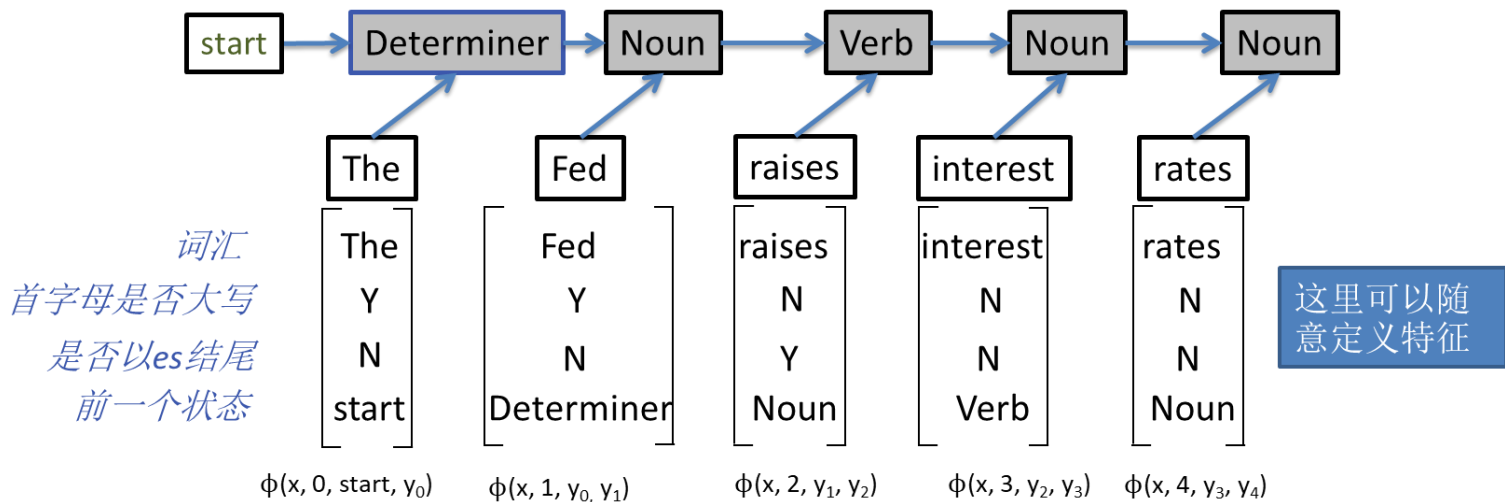
最大熵隐马 MEMM

在解决序列标注问题时，HMM的输入信息只有参数(π, A, B)和观察序列（即每个状态对应的文字），**没有办法接受更丰富的特征**（例如更多的上下文文字等）。为了解决这个问题，提出了MEMM模型。



最大熵隐马 MEMM

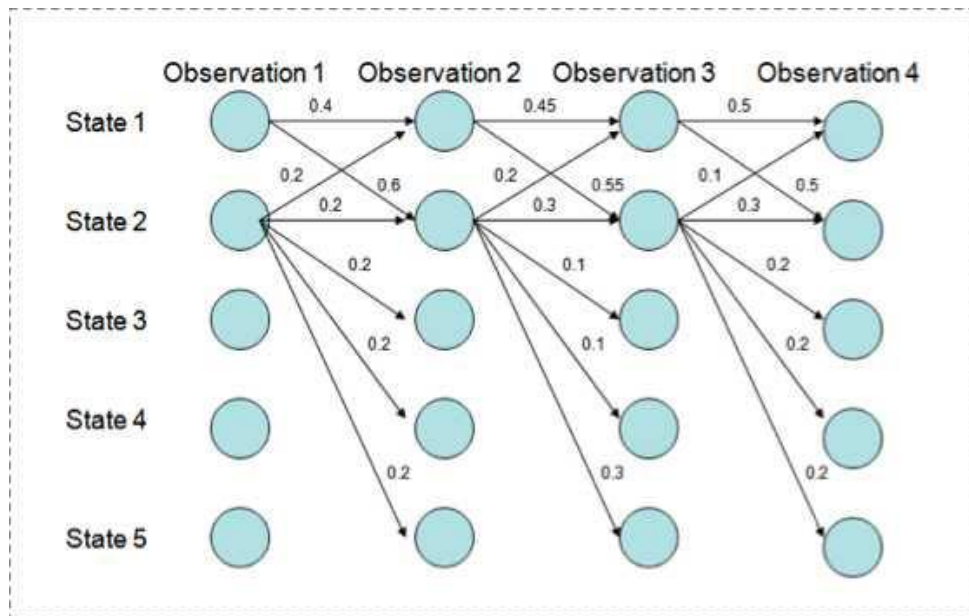
MEMM模型在预测当前状态时，将前一个状态和与当前观察值相关的一组特征一起做为最大熵模型的输入，来预测当前状态。MEMM与HMM不同，是判别式模型。



$$P(y_i | y_{i-1}, \mathbf{x}) \propto \exp(\mathbf{w}^T \phi(\mathbf{x}, i, y_i, y_{i-1}))$$

HMM计算产生整个观察序列的最优状态序列，是全局最优。

MEMM计算单个观察值判定的单个最优状态，是局部最优。

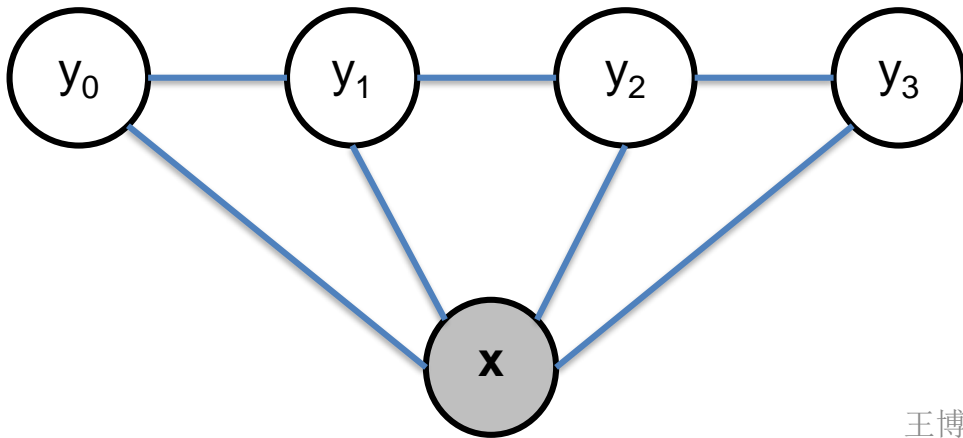


CRF：在更加宽广的上下文上进行判别

MEMM计算单个观察值判定的单个最优状态，**是局部最优**。

CRF计算由整个观察序列判定的最优状态序列，**是全局最优**。其中，每个可能的状态序列的概率这样计算：对于序列中的每个状态计算一组特征函数值，然后计算所有状态的特征函数值之和并归一化。

$$P(y|x) = \frac{1}{Z} \exp\left(\sum_j \sum_{i=1}^{n-1} \lambda_j t_j(y_{i+1}, y_i, x, i) + \sum_k \sum_{i=1}^n u_k s_k(y_i, x, i)\right)$$



HMM, MEMM, CRF

求联合概率

只用到转移概率和生成概率

HMM

$$P(x_1, y_1, \dots, x_n, y_n) = P(y_1) \prod_{i=2}^n P(y_i | y_{i-1}) P(x_i | y_i)$$

局部归一

可使用各种特征

求条件概率

MEMM

$$P(y|x) = P(y_1) \prod_{i=2}^n P(y_i | y_{i-1}, x_i)$$

$$P_{y_{i-1}}(y_i | x_i) = \frac{1}{Z(x_i, y_{i-1})} \sum_a^m (\lambda_a f_a(x_i, y_i))$$

$$P_{y_{i-1}}(y_i | x_{1:n}) = \frac{1}{Z(x_{1:n}, y_{i-1})} \sum_a^m (\lambda_a f_a(x_{1:n}, y_i))$$

求条件概率

全局归一

可使用各种特征

可使用各种特征

CRF

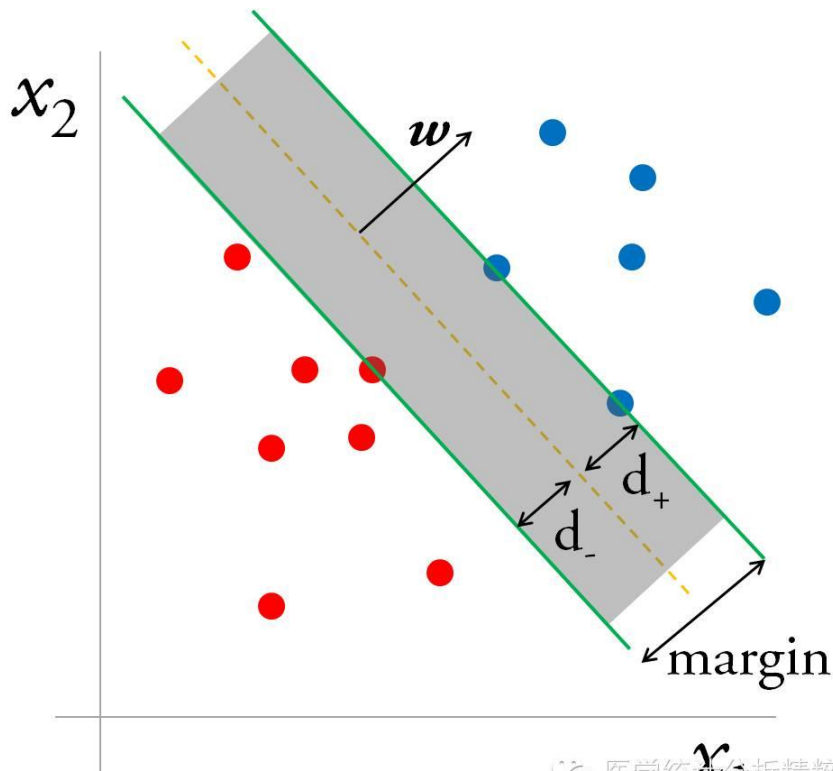
$$P(y|x) = \frac{1}{Z} \exp\left(\sum_j \sum_{i=1}^{n-1} \lambda_j t_j(y_{i+1}, y_i, x, i) + \sum_k \sum_{i=1}^n u_k s_k(y_i, x, i)\right)$$

SVM：从线性到非线性分类

支持向量机 SVM

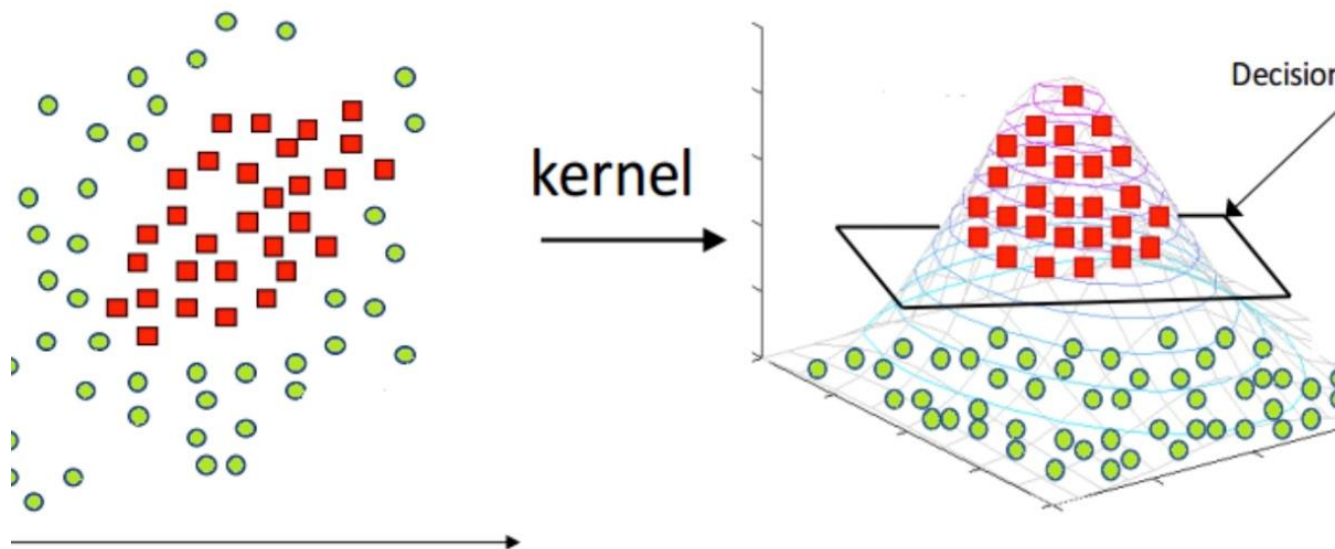
- 支持向量机 (SVM) 同最大熵一样是一种常用的线性 (Log linear) 分类器。
- SVM求使得Margin最大的分类面，并将Margin上的向量称为支持向量。
- 通过计算样本与哪一类支持向量的内积更大来判断样本类别。

$$d(X^T) = \sum_{i=1}^l y_i \alpha_i X_i X^T + b_0$$



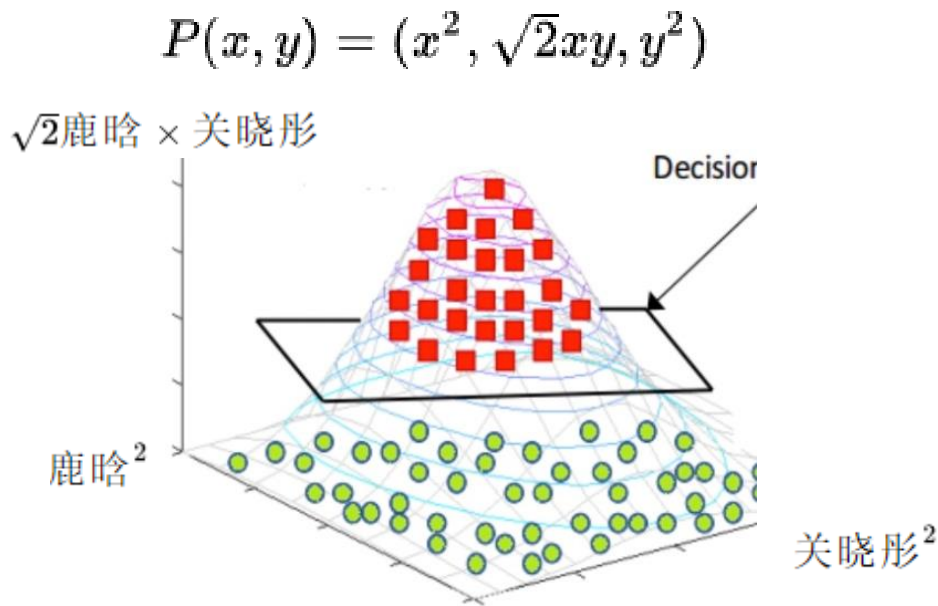
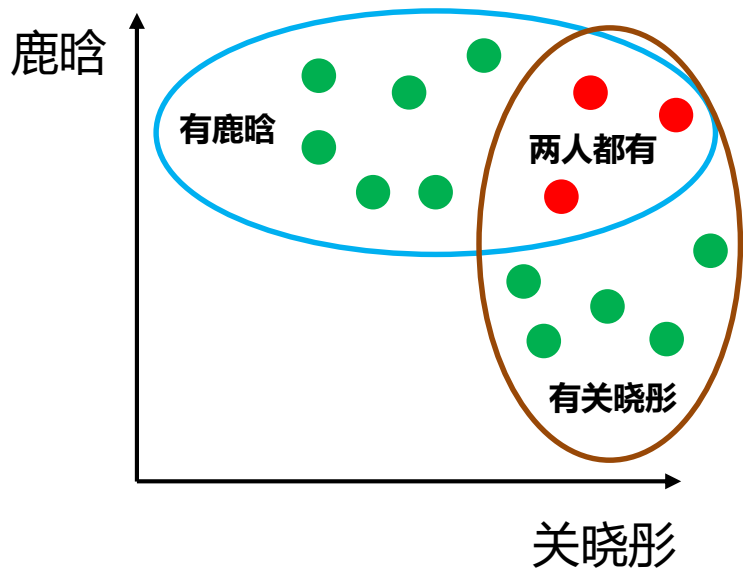
支持向量机 SVM

- 对于线性不可分的样本集，可以将其投射到高维空间中分割。
- SVM用核函数来方便计算高维空间中样本点之间的内积：核函数可以在原空间中计算高维空间中的内积。



支持向量机 SVM

- 要判断一则新闻是否会火，选了两个特征： x =是否有鹿晗， y =是否有关晓彤。
- 实际情况是只有同时包含鹿晗和关晓彤的时候才会火。



支持向量机 SVM

- 要判断一则新闻是否会火，选了两个特征： x =是否有鹿晗， y =是否有关晓彤。
- 实际情况是只有同时包含鹿晗和关晓彤的时候才会火。

为了计算新空间中两个样本点的内积，定义核函数如下，即内积的平方

$$K(v_1, v_2) = \langle v_1, v_2 \rangle^2$$

由下式可见，原空间中两个样本向量的核函数的值正好等于两个向量的高维投影的内积

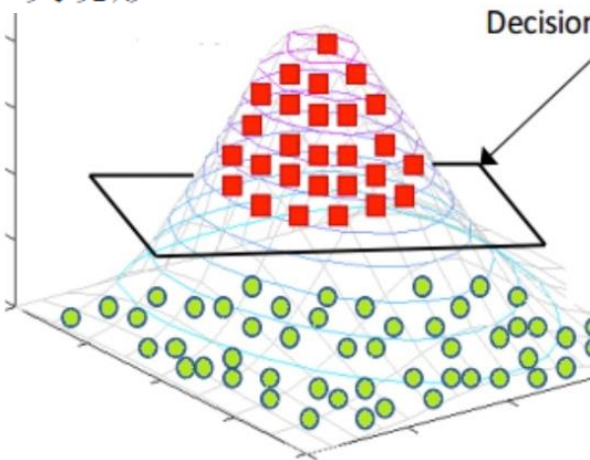
$$\begin{aligned}\langle P(v_1), P(v_2) \rangle &= \langle (x_1^2, \sqrt{2}x_1y_1, y_1^2), (x_2^2, \sqrt{2}x_2y_2, y_2^2) \rangle \\ &= x_1^2x_2^2 + 2x_1x_2y_1y_2 + y_1^2y_2^2 \\ &= (x_1x_2 + y_1y_2)^2 \\ &= \langle v_1, v_2 \rangle^2 \\ &= K(v_1, v_2)\end{aligned}$$

$$P(x, y) = (x^2, \sqrt{2}xy, y^2)$$

$\sqrt{2}$ 鹿晗 \times 关晓彤

鹿晗²

关晓彤²



常用核函数及适用范围

A Practical Guide to Support Vector Classification

<https://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf>

- 线性核: $k(x_i, x_j) = x_i^T x_j$
- 多项式核: $k(x_i, x_j) = (x_i^T x_j)^n, n \geq 1$ 为多项式的次数
- 高斯核: $k(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{2\sigma^2}}, \sigma > 0$ 为高斯核的带宽 (width)
- 拉普拉斯核: $k(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{2\sigma}}, \sigma > 0$
- Sigmoid核: $k(x_i, x_j) = \tanh(\beta x_i^T x_j + \theta)$, \tanh 为双曲正切函数, $\beta > 0, \theta > 0$

通过函数组合:

- 若 k_1 和 k_2 为核函数, 则对任意正数 λ_1, λ_2 , 其线性组合: $\lambda_1 k_1 + \lambda_2 k_2$
- 若 k_1 和 k_2 为核函数, 则核函数的直积: $k_1 \otimes k_2(x, z) = k_1(x, z)k_2(x, z)$
- 若 k_1 为核函数, 则对于任意函数 $g(x)$: $k(x, z) = g(x)k_1(x, z)g(z)$

A decorative graphic consisting of several dark blue lines. A horizontal line at the top left extends to the right, then a vertical line drops down. Another horizontal line is below it, starting further to the right. A vertical line drops down from the end of this second horizontal line. A diagonal line starts from the bottom left and extends towards the center. A horizontal line at the bottom left extends to the right, ending at the vertical line that drops from the second horizontal line.

Thanks !