自然语言处理应用实践

——暑期课程

南京大学软件学院
李传艺
费彝民楼917

第二部分：机器学习算法简介

- 什么是机器学习算法?
  - 区别于“让机器获得学习能力”，仅是指导机器从给定的数据中学习**规律**

- 机器学习算法类型——多种不同的分类方式
  - 监督学习
  - 无监督学习
  - 半监督学习
  - 强化学习

- 聚类算法 (Clustering Algorithm)
- 支持向量机 (Support Vector Machine, SVM)
- 隐马尔可夫模型 (Hidden Markov Model)、条件随机场 (Conditional Random Fields)
- 神经网络模型 (Neural Networks)、深度学习 (Deep-learning)

什么是机器学习算法

- 什么是机器学习?
  - 人工智能的目标之一：让机器获得类似人的学习能力
  - T. Mitchell (米切尔)
    - Carnegie Mellon University Machine Learning
    - – Any **computer algorithm** that lets the system perform a task more effectively or more efficiently than before.
  - H. Simon (西蒙)
    - Professor of Computer Science, Carnegie Mellon
    - – **Learning** denotes changes in the system that are adaptive in the sense that they enable the system to do the task or tasks drawn from the same population more efficiently and more effectively the next time.
  - The ability to perform a task in a situation which has never been encountered before
  - Learning = Generalization
- 什么是机器学习算法?
  - 从给定**数据**中**泛化**数学**模型**的算法；一次性的学习算法；非常依赖数据质量。

处理未来见到的数据，预测未来发生的事情

什么是机器学习算法

- 今天适合运动吗?
  - 场景：早晨起床，根据自己感受到的、看到的一些信息，判断今天是否适合跑步、打球等活动
  - 让计算机**模拟并拓展**人类的这种智能
    - 人类直观感受到的信息不多
    - 人类难以直接建立各种信息之间的联系

- 数据

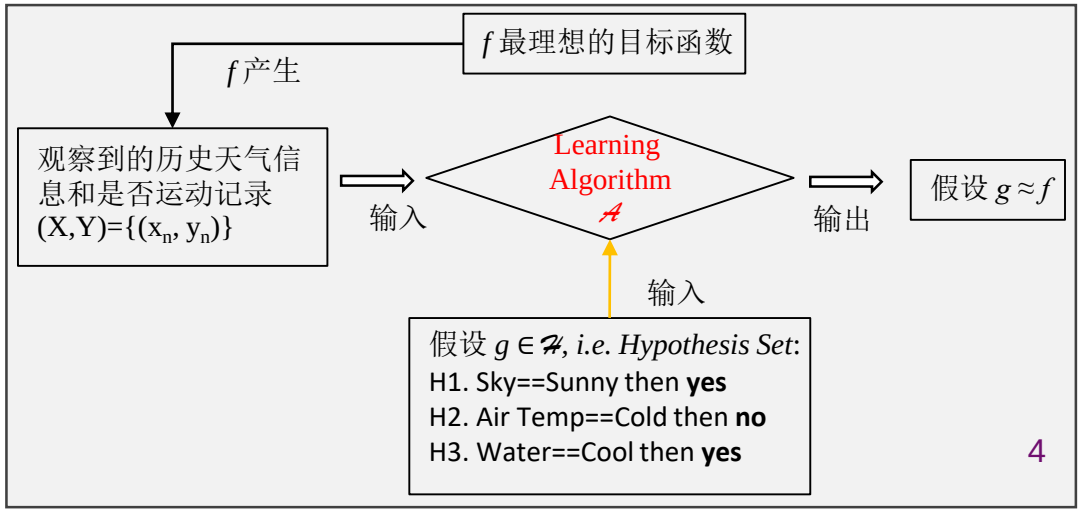
Example	Sky	Air Temp	Humidity	Wind	Water	Forecast	Enjoy Sport
0	Sunny	Warm	Normal	Strong	Warm	Same	**Yes**
1	Sunny	Warm	High	Strong	Warm	Same	**Yes**
2	Rainy	Cold	High	Strong	Warm	Change	**No**
3	Sunny	Warm	High	Strong	Cool	Change	**Yes**

- 泛化

$f : x \rightarrow y$

compute(x) → y

- 模型
  - f 是大自然的杰作 目标函数
  - 人能得到的只是一个 $g \approx f$ 假设



什么是机器学习算法

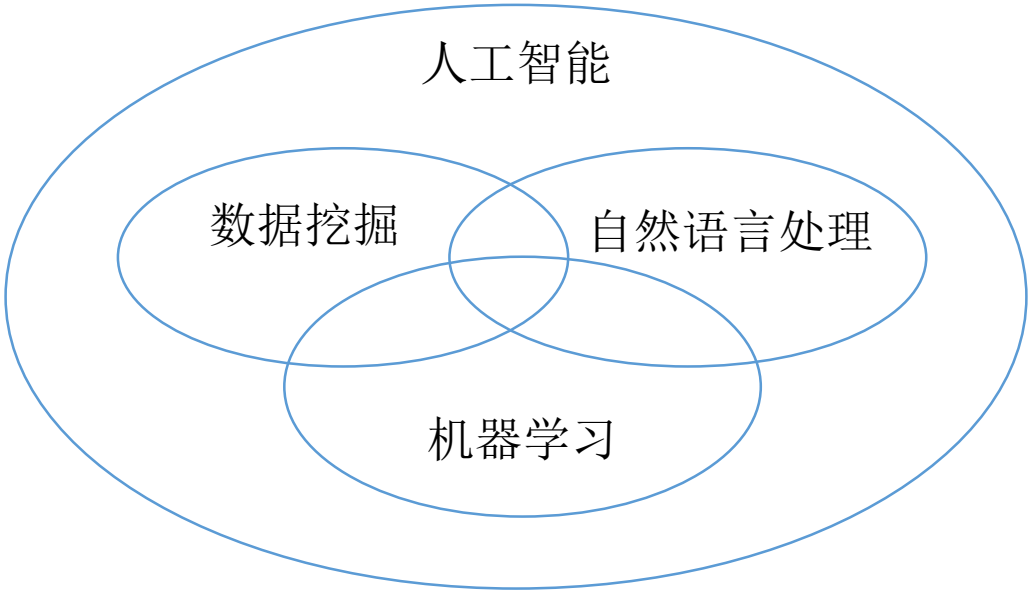
以观察到的数据为样本从假设空间中选择一个与目标函数最像的假设

$\mathcal{A}[(X,Y)] \rightarrow g \in \mathcal{H}\ (g \approx f)$

- 和“数据挖掘”的关系？——使用数据发现有趣、有用的性质
  - g可能就是有趣、有用的性质
  - 有趣、有用的性质可能帮助构造更加接近 f 的 g
  - g可能帮助发现更加有趣、有用的性质

- 和“人工智能”的关系？
  - 模拟“思维”的一种方式

- 和“自然语言处理”的关系？
  - 成为了实现自然语言处理的核心技术



机器学习算法类型

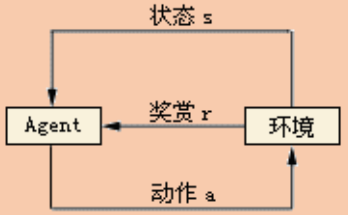
- 按照输入数据类型分
  - 监督学习
  - 无监督学习
  - 半监督学习
  - 强化学习

输入带不带Y？
输入是不是序列？

$\mathcal{A}[(X,Y)] \rightarrow g \in \mathcal{H}\ (g \approx f)$

•如何结合没有Y和少量有Y的数据构建最好效果的模型？→Few shot learning, Transfer learning

•环境（标准的为静态stationary，对应的non-stationary）
•agent（与环境交互的对象）
•动作（action space，环境下可行的动作集合，离散or连续）
•反馈（回报，reward，有了反馈，才能迭代，才学习到策略链）
•数据是序列的、交互的、并且还是有反馈的
•方案(policy)=在每个状态下，你会选择哪个动作？



- 按照预测目标分
  - 分类算法
  - 排序算法
  - 序列标注算法
  - 匹配算法
  - 生成算法

Clustering Algorithm	Support Vector Machine (SVM)	Neural Networks
Hidden Markov Model (HMM)	Conditional Random Fields (CRF)	

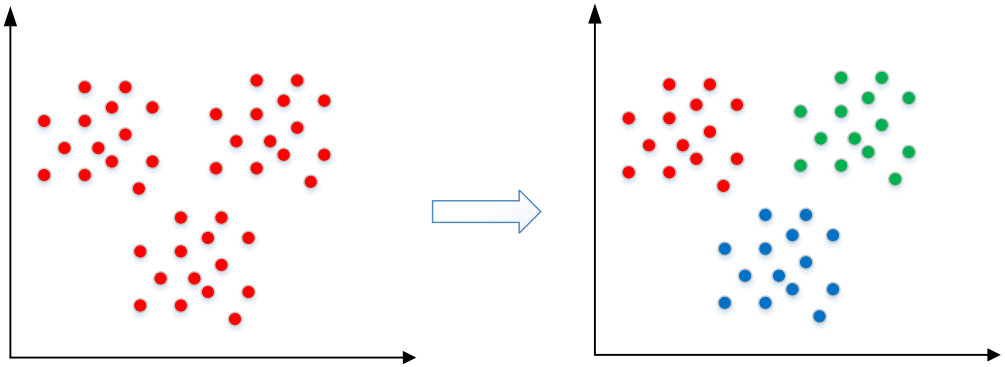
- 按照原理分
  - 判别模型算法：根据给定(x,y)数据集合，构造计算P(y|x)的模型用于预测y
  - 生成模型算法：对给定的(x,y)数据集合，试图获得联合概率分布P(x,y)，然后计算P(y|x)

$$P(y|x) = \frac{P(x,y)}{P(x)} = \frac{P(y) * P(x|y)}{P(x)}$$

聚类算法基础

- 将**没有标签的**数据样例分布到由不相交的类簇构成的子集中，达到：
  - 在同一个子集中的两个数据样例是相似的
  - 在不同子集中的两个数据样例是不相似的

无监督学习



  - 相似程度用两个样例之间的**距离**表示

- 距离函数
  - 闵可夫斯基距离(Minkowski distance)

$$L(\vec{x},\vec{y}) = (\sum_{i=1}^{m} |x_i - y_i|^P)^{\frac{1}{p}}$$

  - 欧氏距离(Euclidean distance)

$$L_2(\vec{x},\vec{y}) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2}$$

  - 曼哈顿距离(Manhattan distance)

$$L_1(\vec{x},\vec{y}) = \sum_{i=1}^{m} |x_i - y_i|$$

  - Cosine距离

$$\cos(\vec{x},\vec{y}) = \frac{\vec{x} \bullet \vec{y}}{|\vec{x}| \cdot |\vec{y}|}$$

字符串编辑距离

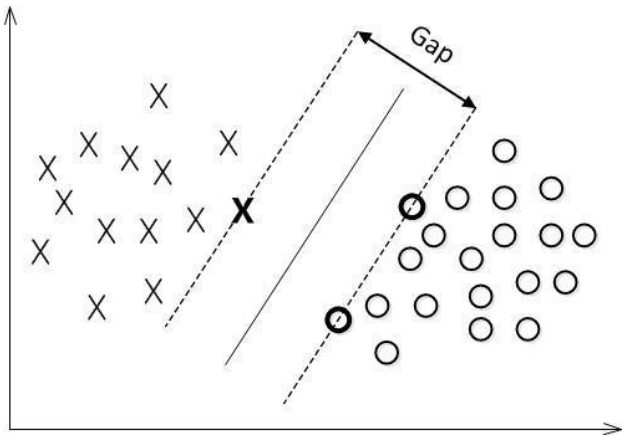
聚类算法类型(Clustering)

- 层次聚类(Hierarchical clustering)
  - 自底向上 or 自顶向下
  - AGNES：自底向上
    - 初始化：把每一个样本当作一个类簇；迭代合并

- 原型聚类(prototype-based clustering)
  - K-means：均值聚类 https://www.jianshu.com/p/4f032dccdcef
    - 目标是最小化每一个类簇里平方误差
    - 贪心策略
  - LVQ：学习向量量化
    - 假设数据样本自带类别标记，用这些标记辅助聚类
    - 有点监督学习的意思
  - 高斯混合聚类 https://zhuanlan.zhihu.com/p/81255623
    - 假设数据满足高斯分布；用EM(Expectation - Maximization)算法求这个分布；按照概率划分类簇
      https://www.cnblogs.com/jerrylead/archive/2011/04/06/2006936.html
- 密度聚类(Density-based clustering)
  - 密度直达、可达、相连概念；找到核心对象；根据核心对象确定对应的类簇；

支持向量机(Support Vector Machine, SVM)

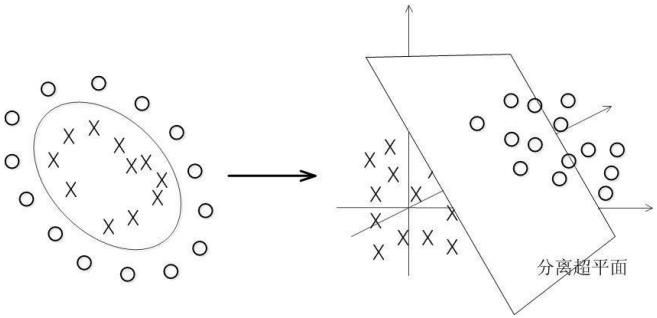
- 最大间隔分类器
  - 分类超平面距离数据的间隔越大，分类置信程度越大

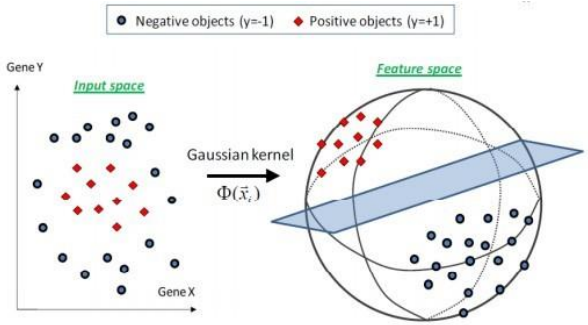


- 支持向量
  - 超平面是由距离平面最近的点决定的
  - 也就是图中在虚线上的点
  - 这些点称为“支持向量”点

- 也可以理解为
  - 找到一个平面，使得到平面距离最近的不同类型的点之间最短距离最大

CSDN非常详细的SVM介绍：https://blog.csdn.net/v_july_v/article/details/7624837





- 工具Libsvm： https://www.csie.ntu.edu.tw/~cjlin/libsvm/

隐马尔可夫模型(Hidden Markov Model)——马尔可夫模型定义

- 马尔可夫系统
  - 对有限状态自动机的扩展：状态集合、状态间转换关系集合（可加权）
  - 有一个有限的状态集合：$S=\{s_1,s_2,\ldots,s_{|s|}\}$
  - 有一个有限的时间序列：t={1, 2, 3, …, T}
  - 在每一个时间点，系统处于一个确定的状态$z_t \in \{s_1,s_2,\ldots,s_{|s|}\}$
  - 每个时间点的状态都是随机选择的
  - 当前时刻的状态决定了下一个时间点状态的概率分布
    - 状态转换概率矩阵：$A=\{a_{ij} \mid s_i \rightarrow s_j$转换的概率，0<i、j<|S|}
    - 开始状态概率：$\pi \in R^{|S|}$

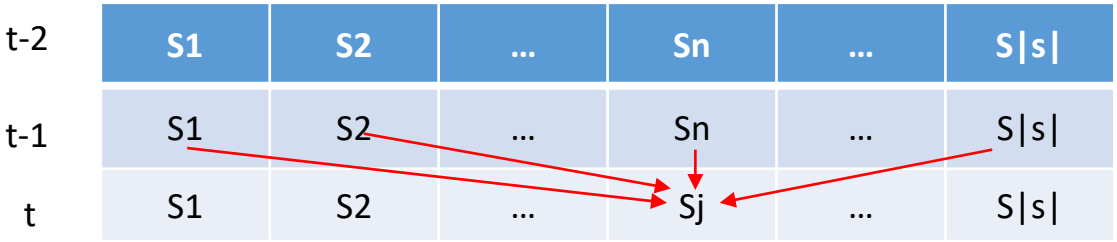
- 例子：天气变化模型，晴天、阴天、雨天、多云

- **问题一**
  - 给定一个马尔可夫系统，观测到$\vec{z}=\{z_1,z_2,\ldots,z_t,\ldots z_T\}$的概率
    - $P(\vec{z})$
    - $=P(z_1,z_2,\ldots,z_t,\ldots z_T)$
    - $=P(z_1,z_2,\ldots,z_t,\ldots z_{T-1})*P(z_T|z_1,z_2,\ldots,z_t,\ldots z_{T-1})$
    - $=P(z_1,z_2,\ldots z_{T-2})*P(z_{T-1}|z_1,z_2,\ldots,z_t,\ldots z_{T-2})*P(z_T|z_1,z_2,\ldots,z_t,\ldots z_{T-1})$
    - ……
    - $=P(z_1)*P(z_2|z_1)*\ldots*P(z_T|z_1,z_2,\ldots,z_t,\ldots z_{T-1})$
    - $= P(z_0)*P(z_1|z_0)*P(z_1|z_0)*\ldots*P(z_t|z_{t-1})*\ldots*P(z_T|z_{T-1})$
    - $=1*A_{z0z1}*A_{z1z2}*\cdots*A_{zT-1zT}$
    - $=\prod_{t=1}^{T} A_{z_{t-1}z_t}$

马尔可夫模型——特定状态概率

- **问题二**：给定模型，则t时刻观察到状态z_t=s的概率
  - P(z_t=s)
  - 设长度为t的状态序列Z是一个以z_t=s结尾的序列，则
  - P(z_t=s)=$\sum_{Z所有可能的情况}$ P(Z)

- 蛮力解法
  - 罗列Z所有可能的序列
  - 求每一个P(Z)
  - 求和



- 巧妙解法：动态规划
  - 如果知道t-1时刻处于某个状态s_i的概率，那么z_t=s_j的概率如何表示?

$$P(z_t = s_j) = \sum_{i=1}^{|s|} P(z_t = s_j \wedge z_{t-1} = s_i)$$

$$= \sum_{i=1}^{|s|} P(z_t = s_j | z_{t-1} = s_i) * P(z_{t-1} = s_i)$$

$$= \sum_{i=1}^{|s|} a_{ij} * P(z_{t-1} = s_i)$$

t	$P(z_t = s_1)$	...	$P(z_t = s_i)$	...	$P(z_t = s_{|s|})$
0					
1					
...					
T					

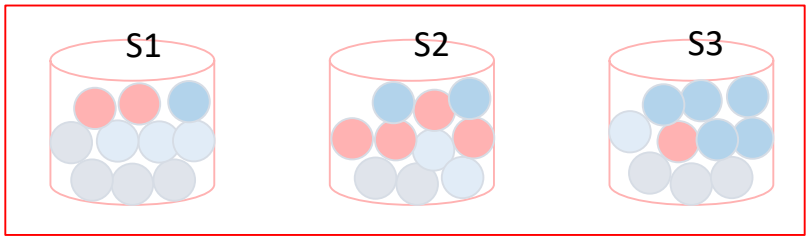
$$P(z_1 = s_j) = \sum_{i=1}^{|s|} a_{01} * P(z_0 = s_i) \qquad P(z_0 = s_i) = \begin{cases} 1, & 以si为开始状态 \\ 0, & 不以si为开始状态 \end{cases}$$

隐马尔可夫模型——定义

- 状态集合S中的状态不可见了
- 但是状态会产生一个可见结果V

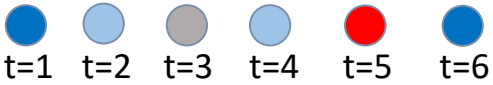
- 例如：从盒子里抓彩色球

S:



V:

$\vec{v}$:



- 隐藏状态，结果，时间，状态转换概率，状态产生结果概率
  - 有一个有限的不可见的状态集合：$S=\{s_1,s_2,...,s_{|S|}\}$
  - 有一个有限的可见的、由状态产生的结果的集合$V=\{v_1,v_2,...,v_{|V|}\}$
  - 有一个有限的时间序列：t={1, 2, 3, ..., T}
  - 在每一个时间点，系统处于一个确定的状态$z_t \in \{s_1,s_2,...,s_{|S|}\}$
  - 但是该状态是不可见的，只能看见这个状态产生的结果$v_t \in \{v_1,v_2,...,v_{|V|}\}$
  - 每个时间点的状态都是随机选择的
  - 当前时刻的状态决定了下一个时间点状态的概率分布
    - 状态转换概率矩阵：$A=\{a_{ij} \mid s_i \rightarrow s_j$转换的概率，$0<i、j<|S|\}$
    - 开始状态概率：$\pi \in R^{|S|}$

  - 每个当前结果由当前状态随机产生，产生结果的概率分布为
    - 产生结果概率矩阵：$B=\{b_{jk} \mid s_j \rightarrow v_k$产生的概率，$0<j<|S|、0<k<|V|\}$

隐马尔可夫模型——结果序列、时间点状态概率

- 给定一个HMM，观测到结果序列$\vec{x}=\{x_1,x_2,\dots,x_T\}$的概率$P(\vec{x})$， T时刻观测到结果是$x_T$的概率$P(x_T)$

$$=\sum_{\vec{z}}P(\vec{x}\wedge\vec{z})$$

$$=\sum_{\vec{z}}P(\vec{x}|\vec{z})*P(\vec{z})$$

为什么？

$$=\sum_{\vec{z}}P(x_t|z_t)*P(x_{t-1}|z_{t-1})*\cdots*P(x_1|z_1)*P(\vec{z})$$

$$=\sum_{\vec{z}}\left(\prod_{t=1}^{T}P(x_t|z_t)\right)*\left(\prod_{t=1}^{T}P(z_t|z_{t-1})\right)$$

动态规划求解

$$=\sum_{\vec{z}}P(x_T\wedge\vec{z})$$

$$=\sum_{\vec{z}}P(x_T|\vec{z})*P(\vec{z})$$

$$=\sum_{\vec{z}}P(x_T|z_T)*P(\vec{z})$$

$$=\sum_{\vec{z}}P(x_T|z_T)*\left(\prod_{t=1}^{T}P(z_t|z_{t-1})\right)$$

给定一个HMM，最可能观测到结果序列$\vec{x}=\{x_1,x_2,\dots,x_T\}$的状态序列

模型训练——马尔可夫模型（1）

- 给定一个观测到的状态序列$\vec{z}$，求最可能产生该序列的马尔可夫模型，就是求状态集合、状态转移概率
- 生成模型 / 判别模型
- 最大似然估计算法
  - 用模型参数将观测结果的概率函数表达出来，求使其取最大值的模型参数
  - 步骤：
    - 写出似然函数
    - 取对数（为什么要取对数？，为什么取对数不改变最大值点位置？）
    - 使用拉格朗日乘子法构造对偶函数（如何理解拉格朗日乘子法）
    - 对所有参数求偏导数，使其为0列出方程组
    - 解之

拉格朗日乘子法：
求f(x)在g(x)=0时最大值点，
等价于求F(x)=f(x)+α*g(x)最大值点

- 似然函数最大值表示

$$\max_A p(A) = P(\vec{z}; A) = \prod_{t=1}^{T} A_{z_{t-1}z_t}$$

- 取对数

$$\max_A l(A) = logP(\vec{z}; A) = log\prod_{t=1}^{T} A_{z_{t-1}z_t} = \sum_{t=1}^{T} logA_{z_{t-1}z_t}$$

$$= \sum_{i=1}^{|S|}\sum_{j=1}^{|S|}\sum_{t=1}^{T} 1\{z_{t-1} = s_i \wedge z_t = s_j\} logA_{ij}$$

$$s.t. \quad \sum_{j=1}^{|S|} A_{ij} = 1, i \in [1, |S|];\ A_{ij} \geq 0, i, j \in [1, |S|]$$

$$\max_A l(A) = logP(\vec{z}; A) = log\prod_{t=1}^{T} A_{z_{t-1}z_t} = \sum_{t=1}^{T} logA_{z_{t-1}z_t}$$

$$f(A) = \sum_{i=1}^{|S|}\sum_{j=1}^{|S|}\sum_{t=1}^{T} 1\{z_{t-1} = s_i \wedge z_t = s_j\} logA_{ij}$$

$$g_i(A) = 1 - \sum_{j=1}^{|S|} A_{ij}, \qquad i \in [1, |S|];$$

$$F(A) = \sum_{i=1}^{|S|}\sum_{j=1}^{|S|}\sum_{t=1}^{T} 1\{z_{t-1} = s_i \wedge z_t = s_j\} logA_{ij} + \sum_{i=1}^{|S|} \alpha_i \left(1 - \sum_{j=1}^{|S|} A_{ij}\right)$$

模型训练——马尔可夫模型（2）

$$F(A) = \sum_{i=1}^{|S|}\sum_{j=1}^{|S|}\sum_{t=1}^{T} 1\{z_{t-1} = s_i \wedge z_t = s_j\} log A_{ij} + \sum_{i=1}^{|S|} \alpha_i \left(1 - \sum_{j=1}^{|S|} A_{ij}\right)$$

$$\frac{\partial F(A)}{\partial A_{ij}} = \frac{1}{A_{ij}} \sum_{t=1}^{T} 1\{z_{t-1} = s_i \wedge z_t = s_j\} - \alpha_i \Rightarrow A_{ij} = \frac{1}{\alpha_i} \sum_{t=1}^{T} 1\{\ldots\ldots\}$$

$$\frac{\partial F(A)}{\partial \alpha_i} = 1 - \sum_{j=1}^{|S|} A_{ij} \Rightarrow \alpha_i = \sum_{t=1}^{T} 1\{z_{t-1} = s_i\}$$

$$A_{ij} = \frac{\sum_{t=1}^{T} 1\{z_{t-1} = s_i \wedge z_t = s_j\}}{\sum_{t=1}^{T} 1\{z_{t-1} = s_i\}}$$

使用输入的观测序列计算A_{ij}

模型训练——隐马尔可夫模型（1）

- 给定一个观测到的结果序列$\vec{x}$，求最可能产生该序列的隐马尔可夫模型，就是求结果集合V、状态集合S、状态转移概率A和结果产生概率B
- 最大似然估计算法
  - 用模型参数将观测结果的概率函数表达出来，求使其取最大值的模型参数
  - 步骤：
    - 写出似然函数
    - 取对数（为什么要取对数？，为什么取对数不改变最大值点位置？）
    - 使用拉格朗日乘子法构造对偶函数（如何理解拉格朗日乘子法）
    - 对所有参数求偏导数，使其为0列出方程组
    - 解之

拉格朗日救不了了，怎么办？

$$maxP(\vec{x}) = max\sum_{\vec{z}} P(\vec{x}\wedge\vec{z}) = max\sum_{\vec{z}}\left(\prod_{t=1}^{T} P(x_t|z_t)\right) * \left(\prod_{t=1}^{T} P(z_t|z_{t-1})\right)$$

- 还有EM算法——用于求参数极大似然估计
  - 若$f(\mathrm{X}) \geq g(\mathrm{X})$，在满足取等号的情况下，求$g(x,y)$最大值，就是在取等情况下$f(x,y)$的最大值

- Jensen不等式
  - 若$f(\mathrm{X})$凹或凸函数，即二次导数恒大于等于0或者小于等于0，则$f(E(\mathrm{X})) \geq E\big(f(X)\big)$,且当且仅当X是常量取等

模型训练——隐马尔可夫模型（2）

- 基于$max\sum_{\vec{z}}P(\vec{x}\wedge\vec{z})$构造上述不等式

$$\max P(\vec{x}) = max\sum_{\vec{z}}P(\vec{x}\wedge\vec{z}) = max\sum_{\vec{z}}\left(\prod_{t=1}^{T}P(x_t|z_t)\right)*\left(\prod_{t=1}^{T}P(z_t|z_{t-1})\right)$$

$$s.t.\sum_{j=1}^{|S|}A_{ij}=1, A_{ij}\geq 0, 1\leq i\leq |S|; \sum_{k=1}^{|V|}B_{jk}=1, B_{jk}\geq 0, 1\leq k\leq |S|$$

- 令$Q(\vec{z})$是每一个能够产生$\vec{x}$的$\vec{z}$出现的可能性，则$Q(\vec{z}) = p(\vec{z}|\vec{x})$
- 且$\sum_{\vec{z}}Q(\vec{z}) = 1$
- $\sum_{\vec{z}}P(\vec{x}\wedge\vec{z}) = \sum_{\vec{z}}Q(\vec{z})*\frac{P(\vec{x}\wedge\vec{z})}{Q(\vec{z})}$
- 令$\mathrm{X} = \frac{P(\vec{x}\wedge\vec{z})}{Q(\vec{z})}$，则$\mathrm{E}(X) = \sum_{\vec{z}}Q(\vec{z})*[\frac{P(\vec{x}\wedge\vec{z})}{Q(\vec{z})}]$
- 令$f(X) = \log(X)$(是凹函数)，则$f\big(E(X)\big) = \log(\sum_{\vec{z}}Q(\vec{z})*[\frac{P(\vec{x}\wedge\vec{z})}{Q(\vec{z})}])$

$$\geq E\big(f(x)\big) = \mathrm{E}\left(\log\left(\left[\frac{P(\vec{x}\wedge\vec{z})}{Q(\vec{z})}\right]\right)\right) = \sum_{\vec{z}}Q(\vec{z})*\log([\frac{P(\vec{x}\wedge\vec{z})}{Q(\vec{z})}])$$

$$\log(\sum_{\vec{z}}P(\vec{x}\wedge\vec{z})) = \log(\sum_{\vec{z}}Q(\vec{z})*\left[\frac{P(\vec{x}\wedge\vec{z})}{Q(\vec{z})}\right]) \geq \sum_{\vec{z}}Q(\vec{z})*\log(\left[\frac{P(\vec{x}\wedge\vec{z})}{Q(\vec{z})}\right])$$

模型训练——隐马尔可夫模型（3）

$$\log(\sum_{\vec{z}} P(\vec{x}\wedge\vec{z})) = \log(\sum_{\vec{z}} Q(\vec{z}) * \left[\frac{P(\vec{x}\wedge\vec{z})}{Q(\vec{z})}\right]) \geq \sum_{\vec{z}} Q(\vec{z}) * \log(\left[\frac{P(\vec{x}\wedge\vec{z})}{Q(\vec{z})}\right])$$

- 如何求最大值？后半部分最大值有用吗？取等号才有用
- 能取等吗？
- $Q(\vec{z}) = p(\vec{z}|\vec{x})$则$\frac{P(\vec{x}\wedge\vec{z})}{Q(\vec{z})} = \frac{P(\vec{x}\wedge\vec{z})}{p(\vec{z}|\vec{x})} = P(\vec{x})$，恰好是一个常数当A和B确定的时候
- 再把其看成关于AB的函数，则可以求一个最大值和对应的A，B值

$$\max \sum_{\vec{z}} Q(\vec{z}) * \log(\left[\frac{P(\vec{x}\wedge\vec{z})}{Q(\vec{z})}\right]) = \max (\sum_{\vec{z}} Q(\vec{z}) * logP(\vec{x}\wedge\vec{z}) - \sum_{\vec{z}} Q(\vec{z}) * log\big(Q(\vec{z})\big))$$

- 再通过$Q(\vec{z})$构造使得不等式取等

- EM算法：
  - E-step：对给定的AB计算能够使得不等式取等的$Q(\vec{z})$
  - M-step：求不等式右边最大值，调整AB

- EM算法核心
  - 构造一个不等式，而这个不等式利用了期望的性质

$$\max (\sum_{\vec{z}} Q(\vec{z}) * logP(\vec{x}\wedge\vec{z}))$$

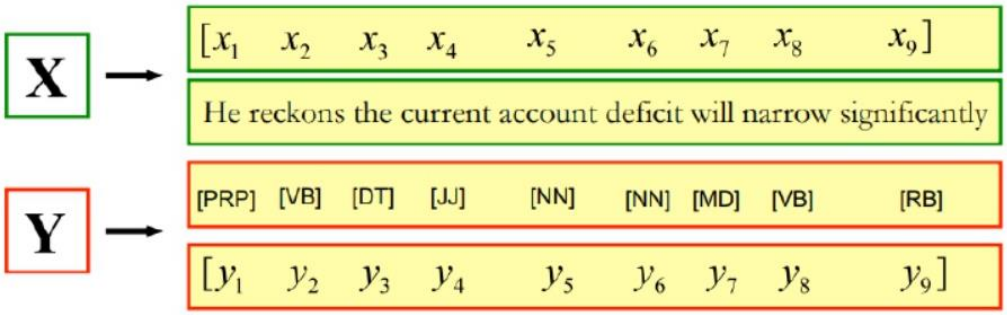
$$= \max \sum_{\vec{z}} Q(\vec{z}) \log(\left(\prod_{t=1}^{T} P(x_t|z_t)\right) * \left(\prod_{t=1}^{T} P(z_t|z_{t-1})\right))$$

$$= \max \sum_{\vec{z}} Q(\vec{z}) * \sum_{t=1}^{T} logB_{z_t x_t} + logA_{z_{t-1} z_t}$$

后面继续用拉格朗日乘子法解

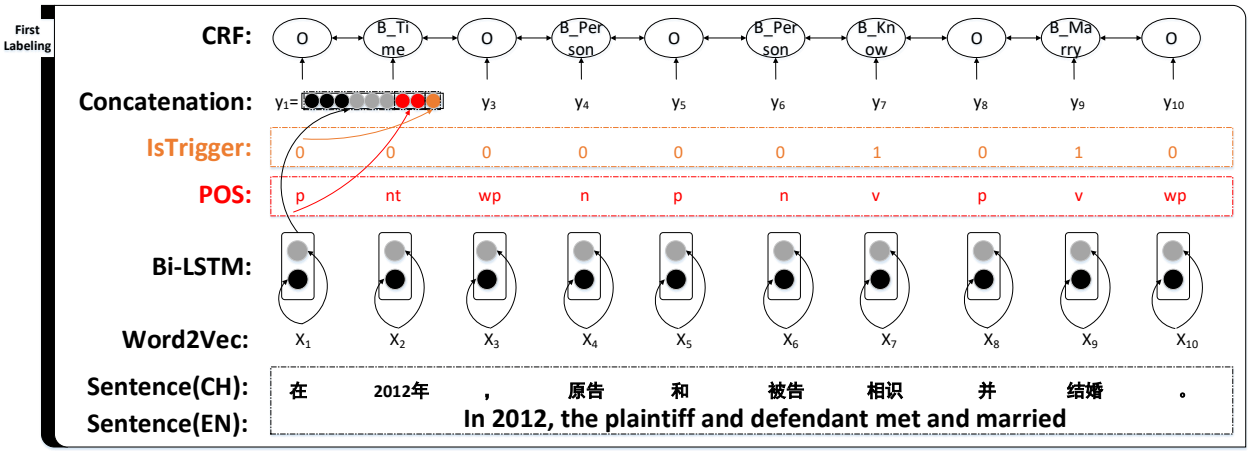
条件随机场-CRF

- CRF 是一个序列化标注算法（sequence labeling algorithm），接收一个输入序列如 [公式] 并且输出目标序列[公式]，也能被看作是一种seq2seq模型。这里使用大写 X,Y 表示序列。



- 使用特征表示X序列中的每一个Token，训练根据Token特征预测其Y中对应Label的模型

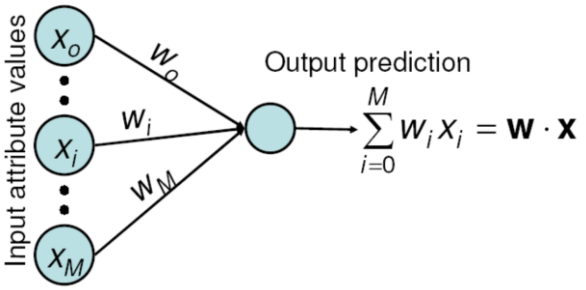
- 工具CRF++：https://taku910.github.io/crfpp/

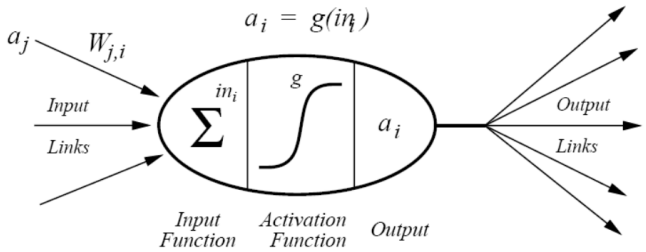


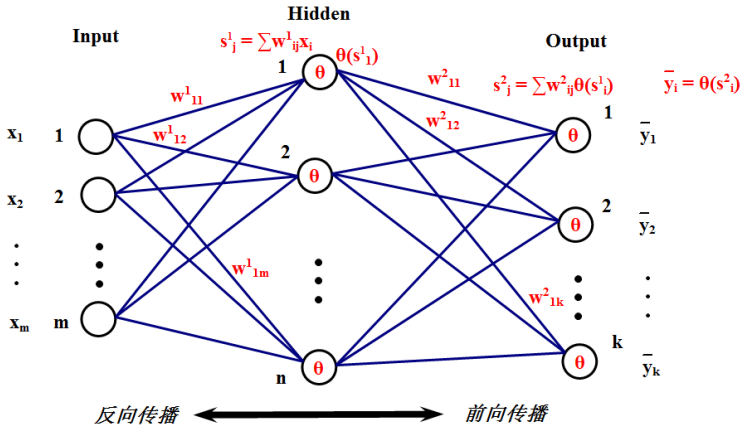
深入阅读，知乎：https://zhuanlan.zhihu.com/p/79719127

神经网络模型

- 模拟人脑神经元对信息的传递
  - 由神经元链接而成
  - 能够模拟任意函数——假设的类型是什么?
- 结果对“假设”的一般表达
  - 层数，每层的神经元个数
  - 每两层间的连接变量W
  - 变换函数
- 目标：将模型预测结果尽可能和正确结果对应上
- 损失函数
  - 衡量预测结果和正确结果之间有差距时，预测结果遭受到的损失——两个结果之间的差别
  - 每一个预测结果都有一个损失值
  - 应该有下界，在所有预测结果都准确时取得



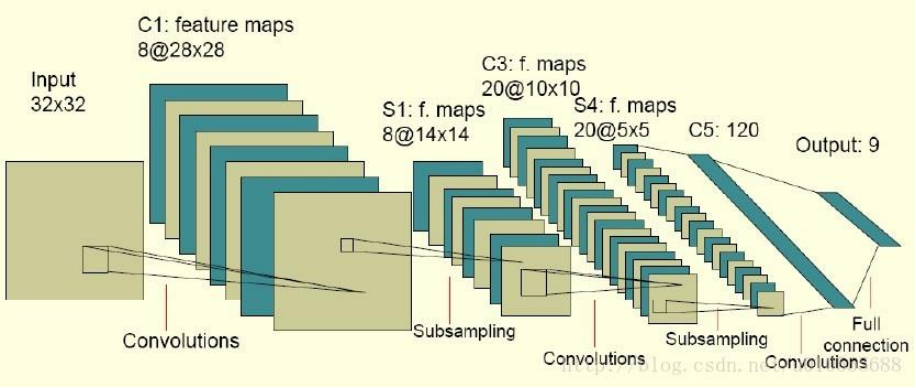


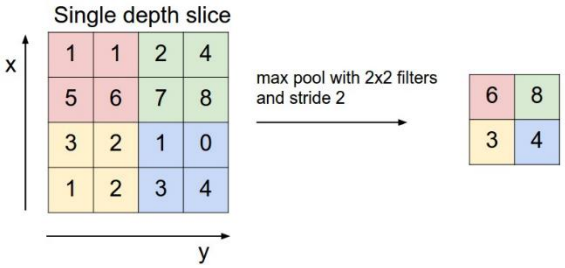


$$\mathcal{L}(\Theta) = \frac{1}{n}\sum_{i=1}^{n} L(f(x_i;\Theta), y_i)$$

神经网络模型——CNN

- 卷积神经网络CNN





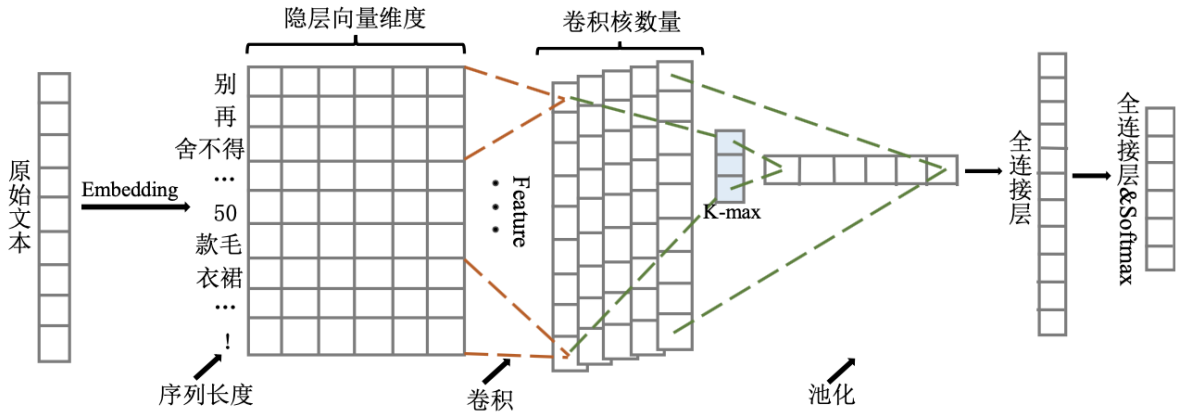
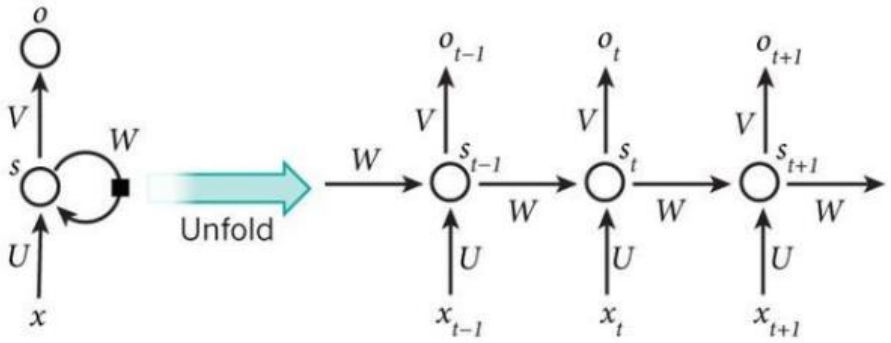
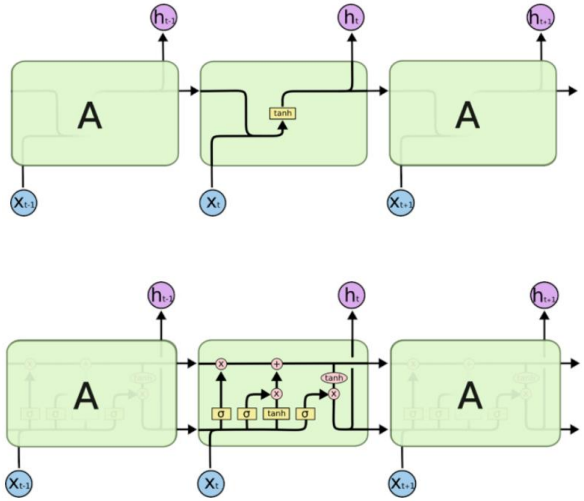


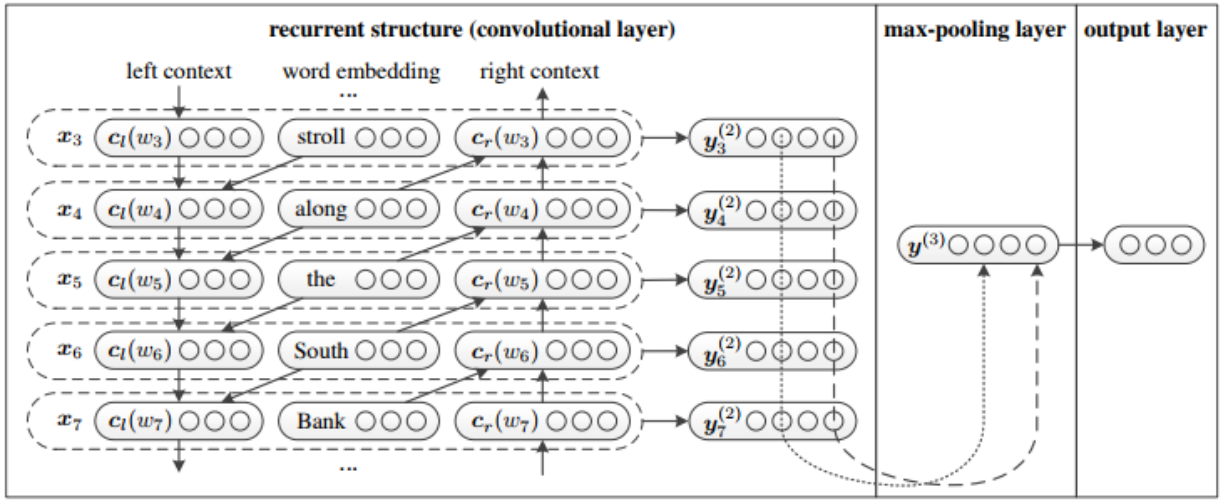
图 3-2: 词语通道 CNN 的主体架构图

神经网络模型——LSTM

- 循环神经网络RNN——长短期记忆网络LSTM



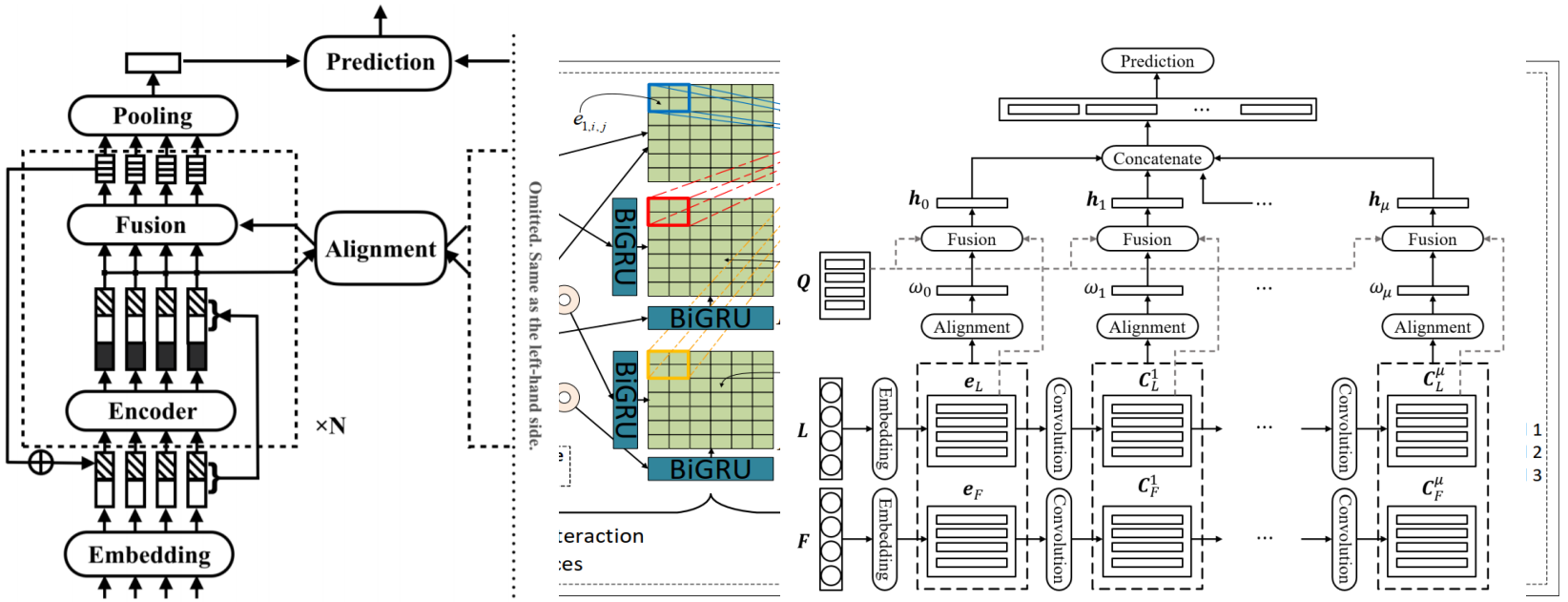




Text Matching

- 快速上手的中文博客资料 https://blog.csdn.net/xiayto/article/details/81247461

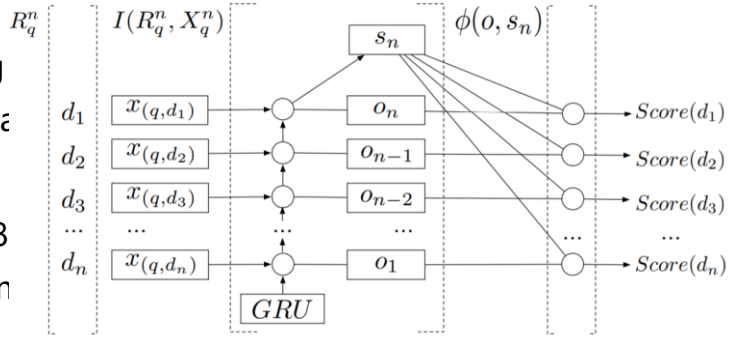
- 相关论文
  - Yu Wu, Wei Wu, Can Xu, and Zhoujun Li. 2018. Knowledge Enhanced Hybrid Neural Network for Text Matching. AAAI 2018, AAAI Press, 5586–5593.
  - Runqi Yang, Jianhai Zhang, Xing Gao, Feng Ji, and Haiqing Chen. 2019. Simple and effective text matching with richer alignment features. arXiv preprint arXiv:1908.00300 (2019)

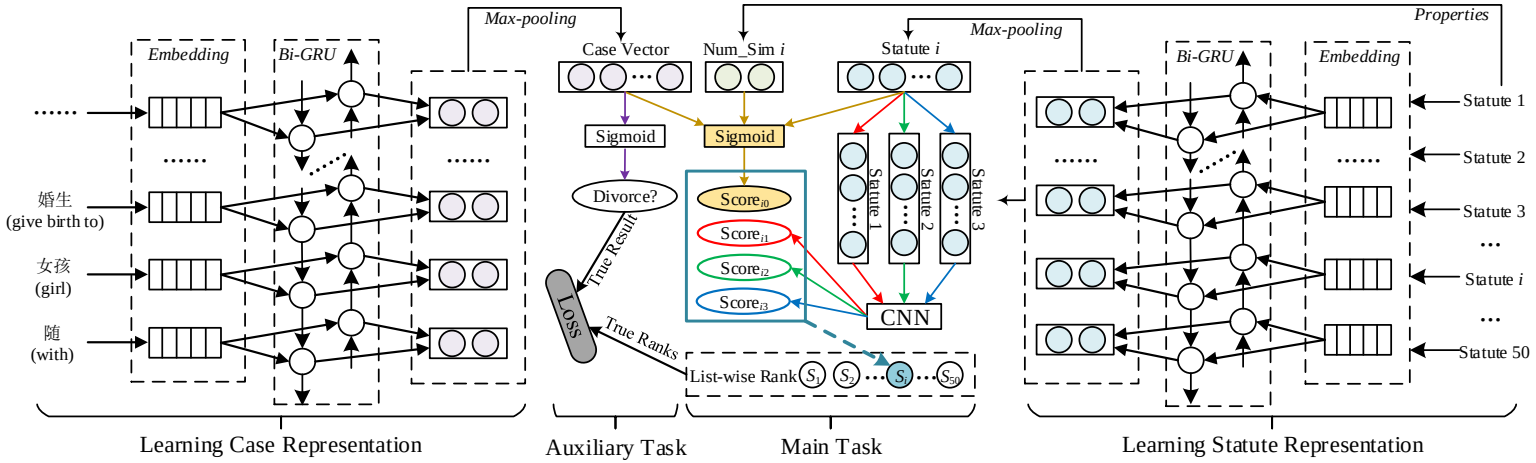


List-wise Learn to Rank

- 快速上手博客：
  - https://www.cnblogs.com/kemaswill/archive/2013/06/01/3109497.html 介绍Learn to Rank
  - https://my.oschina.net/u/4284877/blog/3808078 看List wise部分

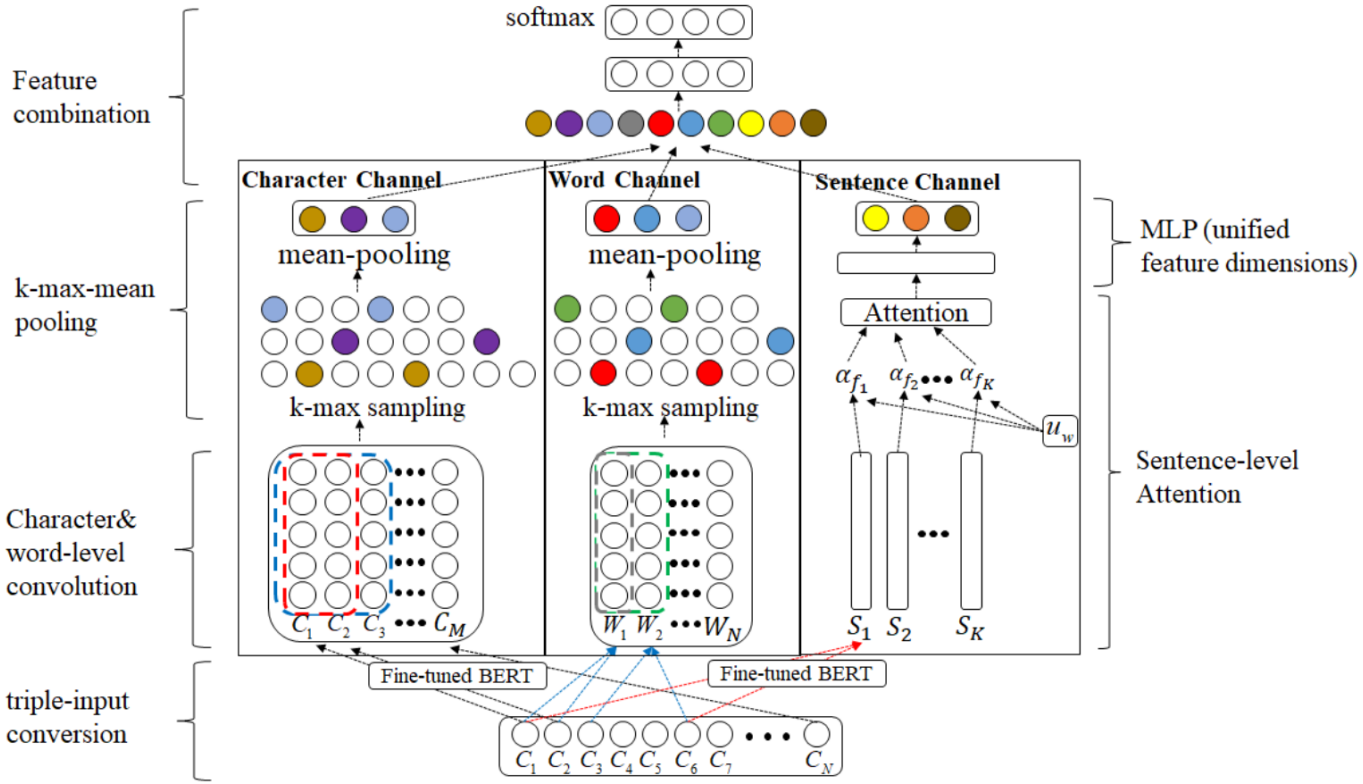
- 相关论文
  - Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang theory and algorithm. In Proceedings of the 25th internationa 1199.
  - Qingyao Ai, Keping Bi, Jiafeng Guo, and W Bruce Croft. 2018 ranking refinement. In The 41st International ACM SIGIR Con Information Retrieval. ACM, 135–144.





Text Classification

- SVM文本分类
- 基于神经网络的文本分类
  - 基本：CNN，RNN，LSTM，GRU，Bi-LSTM, Bi-GRU, ……
  - 文本表示
    - One-hot
    - Work2vec
    - BERT
- 多通道的文本分类

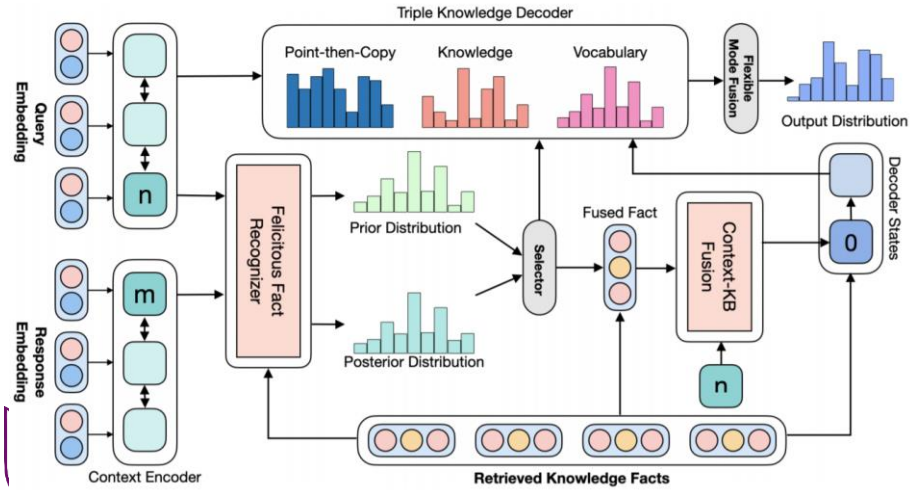


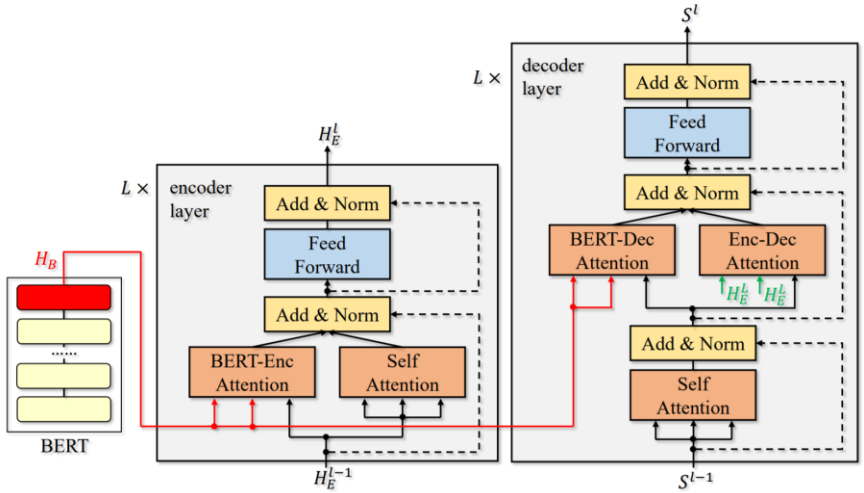
Sequence to Sequence

- 快速上手Seq2Seq： https://blog.csdn.net/qq_32241189/article/details/81591456

- 几乎所有NLG都基于Seq2seq：翻译、对话生成、摘要、问答等

- 相关论文
  - Wu, S., Li, Y., Zhang, D., Zhou, Y., & Wu, Z. (2020, July). Diverse and informative dialogue generation with context-specific commonsense knowledge awareness. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (pp. 5811-5820).
  - Zhu, Jinhua, Yingce Xia, Lijun Wu, Di He, Tao Qin, Wengang Zhou, Houqiang Li, and Tie-Yan Liu. "Incorporating bert into neural machine translation." arXiv preprint arXiv:2002.06823 (2020).

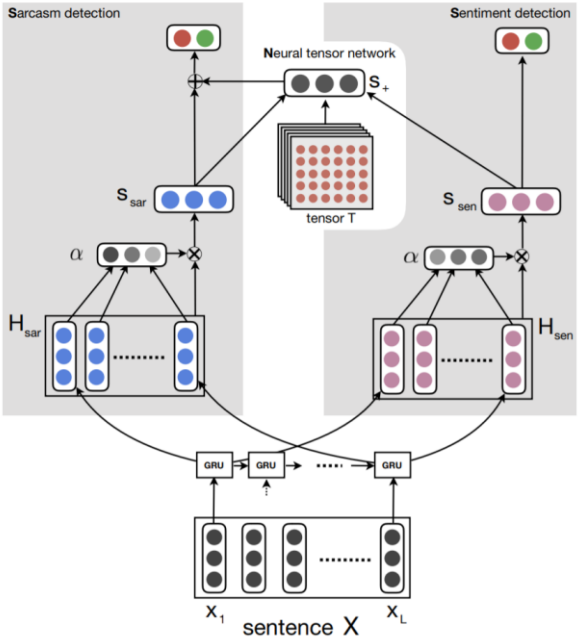


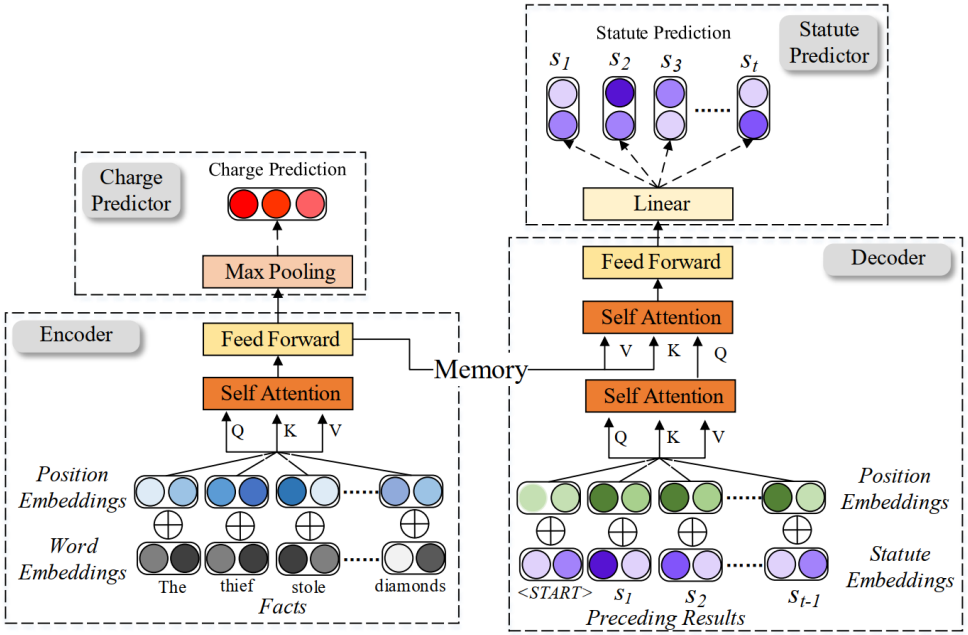


Multi-task/Joint Learning

- 快速上手博客： https://zhuanlan.zhihu.com/p/27421983

- 相关论文
  - 综述文章（各种各样的多任务学习）：Yu, Tianhe, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. "Gradient surgery for multi-task learning." arXiv preprint arXiv:2001.06782 (2020).
  - Majumder, Navonil, Soujanya Poria, Haiyun Peng, Niyati Chhaya, Erik Cambria, and Alexander Gelbukh. "Sentiment and sarcasm classification with multitask learning." IEEE Intelligent Systems 34, no. 3 (2019): 38-43.





谢谢！