被套路 ™ 的 NLP 序列标注问题

Liam Huang*

2017 年 9 月 28 日

*liamhuang0205@gmail.com

套路

我们都见过什么套路?

- 电信诈骗的套路：利用网银贵金属功能;
- 男女朋友*调情*的套路:

  男 为什么老鼠会飞?

  男 因为老鼠吃了印度飞饼.

  男 那么接下来为什么蛇会飞?

  女 哈哈这个我知道，因为它吃了吃了飞饼的老鼠.

  男 对，那你知道鹰为什么会飞?

  女 你这是大直路，一点也不急转弯．因为它吃了会飞的蛇.

  男 笨蛋，因为鹰本来就会飞.

  女 拉黑!

城市套路深，我要回农村

套路究竟是什么？

- 深刻规律的总结；
- 再然后结合实践运用出来.

例子（受力分析）：

- 重力；
- 弹力；
- 摩擦力.

例子（三种不同的红色，古代军事地理）：

- 地形；
- 交通要道.

让套路拯救*被骗钱骗*、*被女朋友拉黑*，*还买不起房的*苦哔的你.

机器学习的套路

具体到某个机器学习的算法，有三个维度的问题需要思考：

- 模型结构：描述实际问题；
- 优化问题：「好坏」的标准，损失函数；
- 求解方法：解最优化问题.

每个维度有三个等级的问题：

- **WHAT**：它是怎样的？
- **HOW**：怎样达成目的？
- **WHY**：为什么它能较好地达成目的？

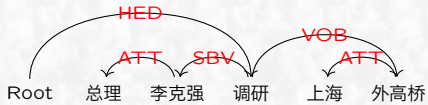
对应人工智能的三重境界：

- 全是人工，没有智能：只会套各种模型；
- 部分人工，有点智能：调参圣手；
- 很少人工，非常智能：根据实际情况设计新模型.

自然语言处理任务

从浅到深：

- 中文分词：
  总理/李克强/调研/上海/外高桥
- 词性标注：
  总理/n 李克强/nh 调研/v 上海/ns 外高桥/ns
- 命名体识别：
  总理/n [李克强 人名] 调研/v [上海外高桥 地名]
- 句法分析：



- 语义分析：语义角色标注、语义依存图、抽象语义表示

结构化预测：NLP 任务的描述

要点：

- 预测结构化对象，而不是离散或连续的单个值；
- 输出序列中各个元素互相有关联.

序列标注统一所有线性方向上的有限标注任务：

- 分词；
- 词性标注；
- 命名体识别；
- 语义角色标注.

总理/n 李克强/nh 调研/v 上海/ns 外高桥/ns

解析算法统一所有将句子转换为树或图的任务：

- 句法分析；
- 语义依存图；
- 抽象语义表示.

序列标注

要点：

- 输出与输入的长度相同；
- 将序列打散成连续的部分；
- 每一部分打上一个标签；
- 标签的候选集是有限的；
- 各部分之间的标签互相影响.

分词示例：

- 输入：总理李克强调研上海外高桥
- 输出：总B 理I 李B 克I 强I 调B 研I 上B 海I 外B 高I 桥I
- 标注：B 表示词的开始，I 表示词的内部

NER 示例：

- 输入：总理李克强调研上海外高桥
- 输出：总O 理O 李B 克I 强I 调O 研O 上B 海I 外I 高I 桥I
- 标注：B 命名实体的开始，I 命名实体的内部，O 明明实体外部

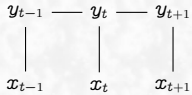
机器学习方法

按照损失函数的定义方式，可以分为以下两类

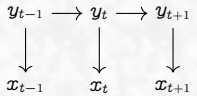
- 基于图的方法;
- 基于转移的方法.（按下不表）

基于图的方法

- 状态序列（标注结果）：$\ldots, y_{t-1}, y_t, y_{t+1}, \ldots$
- 观察序列（真实句子）：$\ldots, x_{t-1}, x_t, x_{t+1}, \ldots$



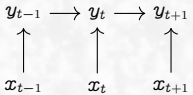
HMM 状态之间存在转移，状态生成观测结果.



$$P(X, Y) \stackrel{\text{def}}{=} \prod_{t=1}^{T} P(y_t \mid y_{t-1}) \cdot P(x_t \mid y_t).$$

- 有监督学习
  - 训练数据：$S \stackrel{\text{def}}{=} \{(X_1, Y_1), (X_2, Y_2), \ldots, (X_s, Y_s)\}$.
  - $\arg\max_\theta \sum_S P(X, Y \mid \theta)$.
  - 极大似然估计.
- 无监督学习
  - 训练数据：$S \stackrel{\text{def}}{=} \{X_1, X_2, \ldots, X_s\}$.
  - 含有隐变量的概率模型：$P(X \mid \theta) = \sum_Y P(X \mid Y, \theta) \cdot P(Y \mid \theta)$.
  - $\arg\max_\theta P(X \mid \theta)$.
  - Baum-Welch 算法（EM 算法 + HMM）.

MEMM 保留 HMM 的状态转移，定义条件概率，最大熵求解.

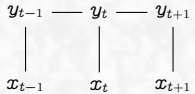


$$P(Y \mid X) \stackrel{\text{def}}{=} \prod_{t=1}^{T} \frac{1}{Z_{y_{t-1},x_t}} \exp \left\{ \begin{array}{l} \sum_j \lambda_j f_j(y_t, y_{t-1}) \\ + \sum_k \mu_k g_k(y_t, x_t) \end{array} \right\},$$

$$\theta^* \stackrel{\text{def}}{=} \arg\max_{\theta} - \sum_{x,y} \tilde{P}(x) P(y \mid x) \ln P(y \mid x).$$

求解方法：拟牛顿法（BFGS）.

CRF 忽略状态转移而以随机场描述，定义条件概率求解.



$$P(Y \mid X) \stackrel{\text{def}}{=} \frac{1}{Z_Y} \prod_{t=1}^{T} \exp \left\{ \begin{array}{l} \sum_j \lambda_j f_j(y_t, y_{t-1}) \\ + \sum_k \mu_k g_k(y_t, x_t) \end{array} \right\},$$

$$\theta^* \stackrel{\text{def}}{=} \arg\max_{\theta} - \sum_{x,y} \tilde{P}(X, Y) \ln P(Y \mid X).$$

求解方法：拟牛顿法（BFGS）.

神经网络方法

CRF 的问题

- 需要人工提取特征（函数），做特征工程；
- 无法输入大量数据进行训练；
- 模型的训练受限于序列长度（T）.

可能的方案：

- 人工神经网络（特征工程、数据量受限）；
- 卷积神经网络（序列长度限制）；
- 循环神经网络（序列长度限制）.

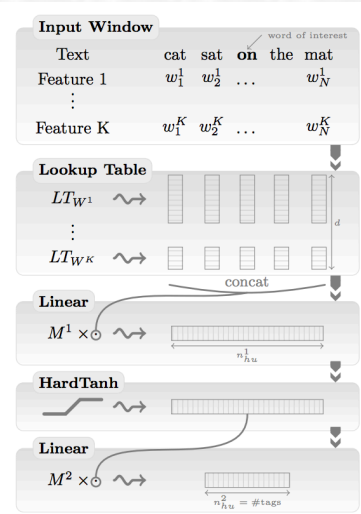
MLP + Window

特点:

- Word Embedding;
- 每次标注一个词;
- 为此每次读入固定大小的窗口内的词;
- 最终以 softmax 得到一个多分类结果.

缺点:

- 窗口大小有限，上下文信息不够;
- 对于 SRL，和谓词有关;
- 如果谓词在窗口之外，则失败.



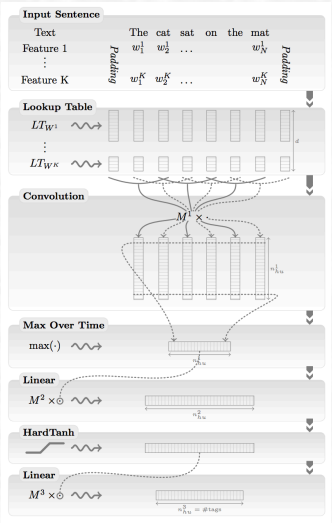
MLP + Sentence

特点:

- Word Embedding;
- 每次标注一个词;
- 读入整句;
- 卷积处理边长序列，获得定长向量;
- 最终以 softmax 得到一个多分类结果.

缺点:

- 逐词判断;
- 不是结构化的问题;
- 实际还是分类问题.



MLP + Sentence + CRF

特点:

- 每个词在各个分类上有一个分;
- 借此定义输入序列和输出序列的计分函数

$$s(X, Y, \tilde{\theta});$$

- 借此定义类 CRF 的形式

$$\ln\Big[p(y \mid x, \tilde{\theta})\Big] \stackrel{\text{def}}{=} s(x, y, \tilde{\theta}) - \ln\Big[\sum_{j \in Y} \exp\Big(s(x, j, \tilde{\theta})\Big)\Big],$$

$$s(x, y, \tilde{\theta}) \stackrel{\text{def}}{=} \sum_{t=1}^{T} \Big(A_{[y]_{t-1}[y]_t} + f_{\tilde{\theta}}(x, [y]_t, t)\Big).$$

小结

- POS 效果和机器学习方法差不多;
- NER/SRL 效果和机器学习方法仍有差距;
- 词间关系在 POS 上意义不大;
- 词间关系在 NER、SRL 上意义重大;
- 训练效率（时间、空间）提升巨大.

Approch	POS	NER	SRL
Benchmark	97.24	89.31	77.92
MLP + Sentence	96.31	79.53	55.40
MLP + Sentence + CRF	96.37	81.47	70.99

System	RAM (MiB)	Time (s)
Toutanova, 2003	1100	1065
Shen, 2007	2200	833
SENNA	32	4

RNN

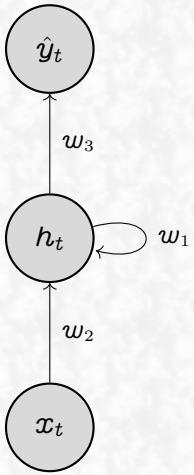
特点：

- 时间序列上循环迭代

$$h_t \overset{\text{def}}{=} \tanh\big(w_1 h_{t-1} + w_2 x_t\big),$$
$$\hat{y}_t \overset{\text{def}}{=} \text{softmax}\big(w_3 h_t\big);$$

- 在时间序列上展开成 FNN，使用反向传播 (BPTT)

$$\frac{\partial L}{\partial w_1} = \sum_{t=1}^{T} \frac{\partial L_t}{\partial w_1},$$
$$\frac{\partial L_t}{\partial w_1} = \sum_{k=1}^{t} \frac{\partial L_t}{\partial \hat{y}_t} \frac{\partial \hat{y}_t}{\partial h_t} \frac{\partial h_t}{\partial h_k} \frac{\partial h_k}{\partial w_1};$$



缺点：

- $\frac{\partial h_t}{\partial h_k}$ 依赖双曲正切的导数；
- 梯度消失 $\leftarrow$ ReLU 作为激活函数.

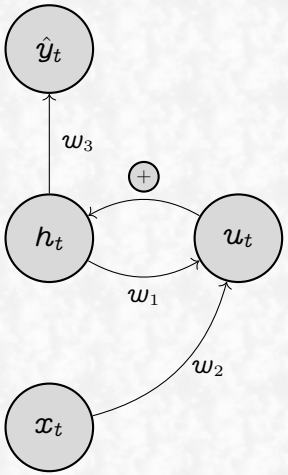
Additively RNN

特点:

- 为了更好地解决梯度消失问题

$$
\begin{aligned}
u_t &\overset{\text{def}}{=} \tanh\big(w_1 h_{t-1} + w_2 x_t\big), \\
h_t &\overset{\text{def}}{=} h_{t-1} + u_t, \\
\hat{y}_t &\overset{\text{def}}{=} \mathsf{softmax}\big(w_3 h_t\big);
\end{aligned}
$$

- 此时 $\frac{\partial h_t}{\partial h_k} = 1 + \frac{\partial u_t}{\partial h_k} \geqslant 1.$



Additively Gating RNN (LSTM)

特点:

- 循环计算时，考虑上一步输出的权重和当前输入的权重

$$
\begin{aligned}
\hat{y}_t &\stackrel{\text{def}}{=} U \cdot h_t, \\
h_t &\stackrel{\text{def}}{=} o_t \cdot \tanh(c_t), \\
c_t &\stackrel{\text{def}}{=} f_t \cdot c_{t-1} + i_t \cdot u_t, \\
u_t &\stackrel{\text{def}}{=} \tanh(w h_{t-1} + v x_t), \\
f_t &\stackrel{\text{def}}{=} \sigma(w_f h_{t-1} + v_f x_t), \\
i_t &\stackrel{\text{def}}{=} \sigma(w_o h_{t-1} + v_o x_t), \\
o_t &\stackrel{\text{def}}{=} \sigma(w_o h_{t-1} + v_o x_t).
\end{aligned}
$$

更多扩展:

- 双向 LSTM;
- 深层双向 LSTM;
- LSTM + CRF.

UGA

The opal stop codon.