天善智能
TIANSHAN SOFT

AI深度学习之自然语言处理顶级实战课程

四、依存句法和语义依存分析

讲师：aopu

自我介绍

- 天善商业智能和大数据社区**aopu** 讲师
- 天善社区ID- **aopu**主页
- https://www.hellobi.com – 学习过程中有任何相关的问题都可以提到技术社区 人工智能 版块。

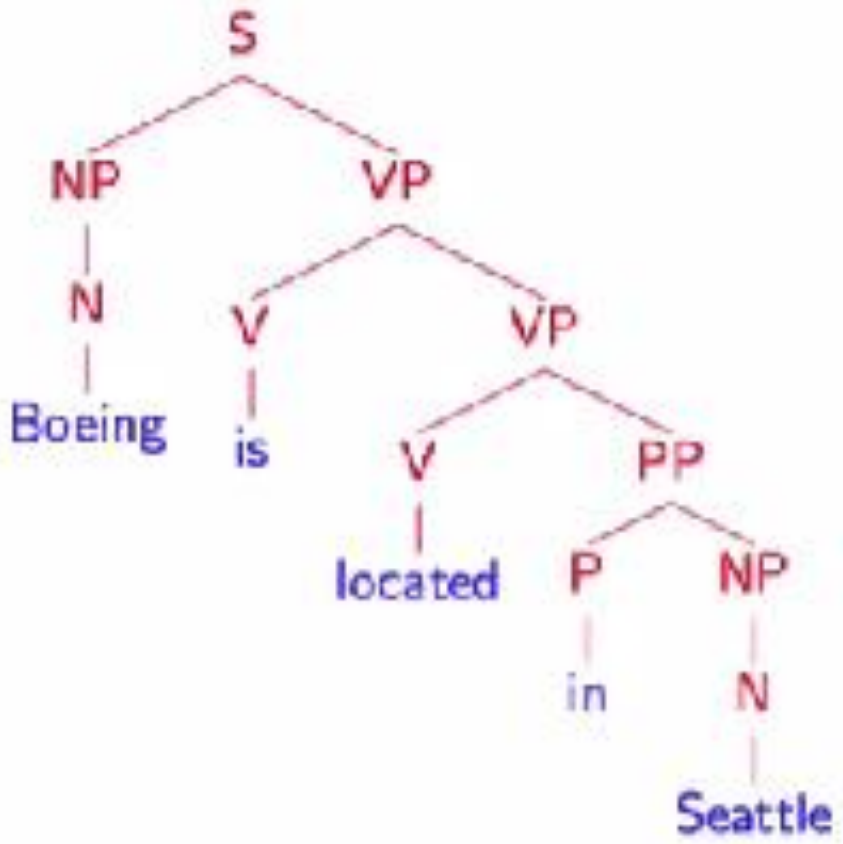
4、4自定义语法与CFG

什么是语法解析？

- 在自然语言学习过程中，每个人一定都学过语法，例如句子可以用主语、谓语、宾语来表示。在自然语言的处理过程中，有许多应用场景都需要考虑句子的语法，因此研究语法解析变得非常重要。
- 语法解析有两个主要的问题，其一是句子语法在计算机中的表达与存储方法，以及语料数据集；其二是语法解析的算法。
- 对于第一个问题，我们可以用树状结构图来表示，如下图所示，S表示句子；NP、VP、PP是名词、动词、介词短语（短语级别）；N、V、P分别是名词、动词、介词。

什么是语法解析？

Boeing is located in Seattle.



4、4自定义语法与CFG

- **上下文无关语法（Context-Free Grammer）**
- 为了生成句子的语法树，我们可以定义如下的一套上下文无关语法。
- 1）N表示一组非叶子节点的标注，例如{S、NP、VP、N...}
- 2）Σ表示一组叶子结点的标注，例如{boeing、is...}
- 3）R表示一组规则，每条规则可以表示为

$$X \to Y_1, Y_2, \ldots, Y_n, X \in N, Y_i \in (N \cup \Sigma)$$

- 4）S表示语法树开始的标注
- 举例来说，语法的一个语法子集可以表示为下图所示。当给定一个句子时，我们便可以按照从左到右的顺序来解析语法。例如，句子the man sleeps就可以表示为(S (NP (DT the) (NN man)) (VP sleeps))。

4、4自定义语法与CFG

- **上下文无关语法（Context-Free Grammer）**

N = {S, NP, VP, PP, DT, Vi, Vt, NN, IN}

S = S

Σ = {sleeps, saw, man, woman, telescope, the, with, in}

R =

S	→	NP	VP
VP	→	Vi	
VP	→	Vt	NP
VP	→	VP	PP
NP	→	DT	NN
NP	→	NP	PP
PP	→	IN	NP

Vi	→	sleeps
Vt	→	saw
NN	→	man
NN	→	woman
NN	→	telescope
DT	→	the
IN	→	with
IN	→	in

4、4自定义语法与CFG

- **概率分布的上下文无关语法（Probabilistic Context-Free Grammar）**
- 上下文无关的语法可以很容易的推导出一个句子的语法结构，但是缺点是推导出的结构可能存在二义性。

由于语法的解析存在二义性，我们就需要找到一种方法从多种可能的语法树种找出最可能的一棵树。一种常见的方法既是PCFG（Probabilistic Context-Free Grammar）。如下图所示，除了常规的语法规则以外，我们还对每一条规则赋予了一个概率。对于每一棵生成的语法树，我们将其中所以规则的概率的乘积作为语法树的出现概率。

4、4自定义语法与CFG

- **概率分布的上下文无关语法（Probabilistic Context-Free Grammar）**

S	⇒	NP	VP	1.0
VP	⇒	Vi		0.4
VP	⇒	Vt	NP	0.4
VP	⇒	VP	PP	0.2
NP	⇒	DT	NN	0.3
NP	⇒	NP	PP	0.7
PP	⇒	P	NP	1.0

Vi	⇒	sleeps	1.0
Vt	⇒	saw	1.0
NN	⇒	man	0.7
NN	⇒	woman	0.2
NN	⇒	telescope	0.1
DT	⇒	the	1.0
IN	⇒	with	0.5
IN	⇒	in	0.5

- 当我们或得多颗语法树时，我们可以分别计算每颗语法树的概率p(t)，出现概率最大的那颗语法树就是我们希望得到的结果，即arg max p(t)。

4、4自定义语法与CFG

- **训练算法**
- 我们已经定义了语法解析的算法，而这个算法依赖于CFG中对于N、Σ、R、S的定义以及PCFG中的p(x)。上文中我们提到了Penn Treebank通过手工的方法已经提供了一个非常大的语料数据集，我们的任务就是从语料库中训练出PCFG所需要的参数。
- 1）统计出语料库中所有的N与Σ；
- 2）利用语料库中的所有规则作为R；
- 3）针对每个规则A -> B，从语料库中估算p(x) = p(A -> B) / p(A)；
- 在CFG的定义的基础上，我们重新定义一种叫Chomsky的语法格式。这种格式要求每条规则只能是X -> Y1 Y2或者X -> Y的格式。实际上Chomsky语法格式保证生产的语法树总是二叉树的格式，同时任意一棵语法树总是能够转化成Chomsky语法格式。

4、4自定义语法与CFG

- **语法树预测算法**
- 假设我们已经有一个PCFG的模型，包含N、Σ、R、S、p(x)等参数，并且语法树总数Chomsky语法格式。当输入一个句子x1, x2, ...，xn时，我们要如何计算句子对应的语法树呢？
- 第一种方法是暴力遍历的方法，每个单词x可能有m = len(N)种取值，句子长度是n，每种情况至少存在n个规则，所以在时间复杂度O(m*n*n)的情况下，我们可以判断出所有可能的语法树并计算出最佳的那个。

4、4自定义语法与CFG

- **语法树预测算法**
- 第二种方法当然是动态规划，我们定义w[i, j, X]是第i个单词至第j个单词由标注X来表示的最大概率。直观来讲，例如xi, xi+1, ...，xj，当X=PP时，子树可能是多种解释方式，如(P NP)或者(PP PP)，但是w[i, j, PP]代表的是继续往上一层递归时，我们只选择当前概率最大的组合方式。特殊情况下。因此，动态规划的方程可以表示为
- 关于动态规划方法，leetcode里有不少案例可以说明。
- 语法解析按照上述的算法过程便完成了。虽说PCFG也有一些缺点，例如：1）缺乏词法信息；2）连续短语（如名词、介词）的处理等。但总体来讲它给语法解析提供了一种非常有效的实现方法。

更多商业智能BI和大数据精品视频尽在 www.hellobi.com

BI、商业智能
数据挖掘 大数据
数据分析师
R语言 Python
机器学习
深度学习
人工智能
Hive Hadoop
Tableau
BIEE ETL
数据科学家
PowerBI

天善智能
TIANSHAN SOFT