

文章编号: 1003-0077(2013)06-0058-11

联合语义角色标注和指代消解

熊皓^{1,2}, 刘群¹, 吕雅娟¹

(1. 中国科学院 计算技术研究所, 北京 100190; 2. 橙译中科信息技术有限公司, 北京 100010)

摘要: 语义角色标注和指代消解是自然语言处理中两个重要的基础研究任务, 该文提出使用马尔科夫逻辑网络对两个任务进行联合学习推导。通过提出 8 条联合学习规则, 在 OntoNote5.0 的实验中证明采用联合学习推导后, 两者的标注性能都比单独标注任务有了 1.6 个 F 值的性能提升。

关键词: 语义角色标注; 指代消解; 马尔科夫逻辑网络

中图分类号: TP391 文献标识码: A

Joint Semantic Role Labeling and Coreference Resolution

XIONG Hao^{1,2}, LIU Qun¹, LV Yajuan¹

(1. Institute of Computing Technology, Beijing 100190, China; 2. Torangetek Inc., Beijing 100010, China)

Abstract: Semantic Role Labeling (SRL) and Coreference Resolution (CR) play important role in natural language processing applications. In this paper, we propose 8 rules to jointly learn and inference two tasks using markov logic network. Experimental results on OntoNote5.0 show that joint learning with markov logic network significantly improve 1.6 points in term of F score both on SRL and CR towards single systems.

Key words: semantic role labeling; coreference resolution; markov logic network

1 导论

语义角色标注和指代消解是自然语言处理中的两个重要基础研究, 相关的研究都被广泛应用于其他的自然语言处理应用中。基于直觉上来说, 这两个任务之间具有很强的联系性, 即语义角色标注的结果可以帮助改进指代消解^[1-2], 而指代消解的结果可以帮助语义角色标注。前者已经有相关的工作验证, 后面一个假设目前来说没有相关的研究工作可以参考。以 OntoNote^[3] 语料中的一篇文章片段为例:

1. Because [*I*]_{ARG0} **know** [*this situation*]_{ARG1}.
2. Right, like those whom we interviewed just now .
3. [*It*]_{ARG1} was easy to **handle** once everyone

understood .

其中第 3 个句子的“*It*”指代的是第 1 个句子的“*this situation*”。在这个例子中, 第 3 个句子的“*It*”由于出现在谓词前面, 很容易被错误的标注为“ARG0”, 但是如果能够知道“*It*”和前面的“*this situation*”指的是同一实体, 而“*this situation*”指的是事务, 更多时候是受事者, 因此“*It*”标注为“ARG1”的可能性更大。因此基于语义角色标注和指代消解相互关联这个直觉, 我们提出使用马尔科夫逻辑网对它们进行联合学习推导。

马尔科夫逻辑网络 (MLN) 是 Richardson 和 Domingos^[4] 提出的一种知识表达方式, 被广泛的应用于一些联合任务中, 如指代消解^[5-6], 语义角色标注和语义消歧^[7]。由于马尔科夫逻辑网络模型简单, 对于一个新的问题常常只需要定义一些新的逻辑表达式, 特别对于联合模型来说, 可以通过简单的

收稿日期: 2013-07-15 定稿日期: 2013-09-15

基金项目: 国家 863 计划重大项目资助(2011AA01A207)

作者简介: 熊皓(1985—), 男, 助理研究员, 研究方向是自然语言处理、机器学习; 刘群(1966—), 男, 研究员, 研究方向是自然语言处理、机器翻译、信息提取; 吕雅娟(1972—), 女, 副研究员, 研究方向是自然语言处理、机器翻译。

增加一些约束表达式来完成联合任务,因此本文选用马尔科夫逻辑网络来联合语义角色标注和指代消解。本文最后大量的实验表明,采用联合学习后,两者的标注性能相比于单独标注系统都有了 1.6 个 F 值的提升。据我们所知,这也是第一篇证明指代消解结果可以帮助语义角色标注的工作。

本文的章节组织结构如下:我们首先在第 2 节简单介绍一下马尔科夫逻辑网络的背景知识;在第 3 节将详细介绍我们的模型采用的联合学习规则;第 4 节是实验部分;最后在第 5 节我们将总结我们的工作和对未来工作做出一些展望。

2 马尔科夫逻辑网络

仅仅使用本地特征在语义角色标注任务中是不够的,一些研究者试图利用一些全局特征^[8]来提高标注的性能,或者采用联合学习方式来提高两者的性能。随着当前机器学习研究的进展,有多种方式可以实现我们的目标,例如整数线性规划、马尔科夫逻辑网络等等。在这里我们选用后者,因为其定义简单,支持判别式训练,更重要的一点是目前有很多相关的开源软件可供使用,有利于快速、高效的验证实验结果。

简单来说,马尔科夫逻辑网络可以认为是对统一一阶逻辑模型的扩充,马尔科夫模型对一阶逻辑中的公式(formula)增加了权重,用于衡量约束的强度,试图用带有权重的公式集合去描述已知知识的概率分布。

下面我们结合本文相关的任务实例来具体讲解马尔科夫逻辑网络。

逻辑谓词^①(predicate)

以指代消解为例,我们要做的决策是判别两个提及 m_i 和 m_j 是否构成共指关系。因此我们可以通过一个二元的逻辑谓词来表达这个决策: $coref(m_i, m_j)$, 并且这个逻辑谓词是隐藏的,可以叫做隐藏逻辑谓词(hidden predicates),因为在测试时我们并不知道此逻辑谓词的真假。在马尔科夫逻辑网络中还有一类逻辑谓词叫作已观察到的逻辑谓词(observed predicates),表示这些信息是我们在测试集中可以观察到的。例如我们可以定义一个相应的逻辑谓词 $headMatch(m_i, m_j)$ 表示表示两个提及的中心词是否匹配。

公式(formula)

根据上面的逻辑谓词定义,我们可以使用一些

带有权重的一阶逻辑公式(first-order logic formulae)来表达我们的研究动机。例如,我们可以定义如下一条公式:

$$headMatch(m_i, m_j) \Rightarrow coref(m_i, m_j) \quad (1)$$

上面这条公式不一定在所有条件下成立,但是我们可以认为在大多数情况下成立。因此我们给上面的公式一个权重 ω , 一般来说当 ω 越大时,上面这条公式在我们的模型中成立的可能性就越大。并且 ω 的取值不需要人为指定,可以从训练语料中学习得到。

虽然上面的公式使用传统的本地分类器(local classifier)也可以达到相同的效果,但是使用马尔科夫逻辑网络我们可以完成更强的推导,例如:

$$coref(m_i, m_j) \wedge coref(m_j, m_k) \Rightarrow coref(m_i, m_k) \quad (2)$$

在上面这个公式中,我们定义了一条本地分类器无法实现的全局约束规则,即指代消解中常见的指代传递性问题^[9]。而此类全局约束规则也是本文的核心思想,通过马尔科夫逻辑网络我们能够轻松的实现这些目标。

形式化定义

形式化来说,马尔科夫逻辑网 M 是一个带权公式的集合,表示为 $\{(\Phi_i, \omega_i)\}_i$, 其中 Φ_i 为一阶逻辑公式, ω_i 为其对应的权重。马尔科夫逻辑网络试图通过调整 ω_i 的大小来描述可能世界的概率。例如假设 y 表示一个可能的世界(possible world), 或者说是基态原子(ground atoms)集合, 则这个可能的世界出现的概率 $p(y)$ 可以定义如下:

$$p(y) = \frac{1}{Z} \left(\sum_{(\Phi_i, \omega_i) \in M} \omega_i \sum_{c \in C^{n_{\Phi_i}}} f_c^{\Phi_i}(y) \right) \quad (3)$$

其中 c 是 Φ_i 中可以实例化自由变量的常量, $f_c^{\Phi_i}$ 表示一个二值特征函数, 如果使用 c 对 Φ_i 中的自由变量进行替换后, 得到的基态公式(ground formula)在世界 y 中仍然是正确的, 则返回 1, 否则返回 0。 n_{Φ_i} 是公式 Φ_i 中自由变量的个数。 $C^{n_{\Phi_i}}$ 是可以用于实例化自由变量的常量集合, Z 是归一化常量。需要说明的是, 分布 $p(y)$ 对应于一个马尔科夫网络, 其中图中的节点表示基态原子, 因子(factor)表示基态公式。例如, 在世界 y 中, 我们有两个提及实例 A, B , 并且有逻辑谓词 $headMatch$ 构成的基态原子 $headMatch(A, B), headMatch(B, C)$ 。根据上

^① 为了和语义角色标注中的谓词相区分, 我们对马尔科夫逻辑网络中的谓词添加了逻辑两个字用于区分。

面的公式(1),(2),我们可以得到如图1所示的马尔科夫图模型,其中因子1和因子2对应于公式(1),因子3对应于公式(2)。

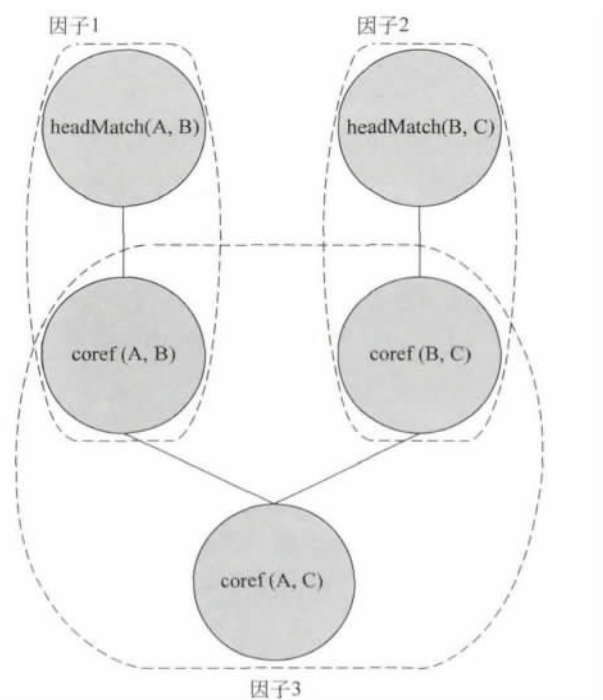


图1 根据公式(1),(2)和逻辑谓词 *headMatch* 生成的马尔科夫网络模型(其中虚线为马尔科夫图模型中的因子(factor))。

模型应用

一般来说,利用马尔科夫逻辑网络进行任务求解时,我们只需要定义好类似于公式(1),(2)的启发式规则。虽然我们需要对整个模型进行训练,例如

给公式分配不同的权重 w ,以及给定一些观察信息时,如何从图模型中推理出最可能的基态原子等,但是一般的开源工具,如 Alchemy^① 和 Markov thebeast^②都提供了相应的求解算法,因此我们需要做的仅仅是根据他们的格式定义,在训练语料中抽取相应的基态原子实例。

3 联合标注模型中使用的公式

在我们的联合任务中,语义角色的分类以及指代消解中的提及分类都是本地化的不涉及到全局特征,而语义角色的推导,指代消解中的实体推导以及两个任务之间的联合推导都是全局化的,需要使用到彼此之间的推导结果,因此我们将公式分为两种:本地公式和全局公式。通过将本地公式和全局公式融入到马尔科夫逻辑网络中实现联合分析的目的。

3.1 本地公式

本地公式由若干个已观察到的逻辑谓词和最多一个隐藏逻辑谓词组成。对于语义角色标注来说,观察到的逻辑谓词大部分来自前人工作总结的比较好的判别特征^[8, 10-13],最后结果如表1所示,隐藏逻辑谓词只有一个: *role*。对于指代消解来说,我们参照现有一些研究工作^[6, 14-16],使用的观察到的逻辑谓词如表3所示,隐藏逻辑谓词只有一个: *coref*。两个任务使用的本地公式如表2,表4所示,其中符号“+”表示针对当前公式里面可以替换的常量我们单独给予一个权重。

表1 语义角色任务中使用的已观察到逻辑谓词

谓词 <i>pre</i> 的逻辑谓词 <i>voice(i)</i> : 谓词是主动还是被动
<i>subCategory(i, sub)</i> : 谓词父亲节点的子范畴
<i>phraseType(i, pt)</i> : 候选节点的句法功能标记
<i>predicateLemma(i, pl)</i> : 谓词的词根
<i>pathLen(i, plen)</i> : 从候选节点到谓词节点需要经过多少个节点
<i>missSub(i)</i> : 是否缺少主语
候选节点 a_j 的逻辑谓词
<i>headWord(j, hw)</i> : 候选节点的中心词
<i>headPos(j, hp)</i> : 候选节点的词性
<i>firstWord(j, fw)</i> : 候选节点的第一个单词
<i>firstPOS(j, fp)</i> : 候选节点的第一个单词词性
<i>lastWord(j, lw)</i> : 候选节点的最后一个单词

① <http://alchemy.cs.washington.edu/>

② <http://code.google.com/p/thebeast/>

续表

lastPOS(j,lp):	候选节点的最后一个单词词性
prevWord(j,pw):	候选节点前一个单词
nextWord(j,nw):	候选节点后一个单词
prevTwoWord(j,ptw):	候选节点前两个单词
nextTwoWord(j,ntw):	候选节点后两个单词
leftSister(j,ls):	候选节点左节点的句法功能标记
rightSister(j,rs):	候选节点右节点的句法功能标记
leftSisterHeadWord(j,lshw):	候选节点左节点的中心词
leftSisterHeadPos(j,lshp):	候选节点左节点的中心词词性
rightSisterHeadWord(j,rshw):	候选节点右节点的中心词
rightSisterHeadPos(j,rshp):	候选节点右节点的中心词词性
parent(j,pa):	候选节点父亲节点的句法功能标记
parentHeadWord(j,phw):	候选节点父亲节点的中心词
parentHeadPOS(j,php):	候选节点父亲节点的中心词词性
ppParentHW(j,pphw):	如果候选节点父亲节点被标记为介词短语则返回其中心词
ppnpHW(j,ppnphw):	如果上面特征生效,则返回其最右边 NP 节点的中心词词性
ppnpHP(j,ppnphp):	如果上面特征生效,则返回其最右边 NP 节点的中心词词性
prevPred(j,prep):	候选节点的前面一个谓词
nextPred(j,nextp):	候选节点的后一个谓词

谓词 *pre*_{*i*} 和候选节点 *a_j* 关联的逻辑谓词

cand(i,j):	节点是谓词的候选节点
partialPath(i,j,ppath):	从候选节点到谓词最近公共祖先节点的路径
projectPath(i,j,prop):	候选节点到控制谓词的最顶层 VP 节点的路径
path(i,j,path):	从候选节点到谓词节点的句法路径
position(i,j):	候选节点在谓词前或者后

3.2 全局公式

由于我们是联合语义角色标注和指代消解任务,首先我们给出语义角色推导中使用的一些全局约束,然后在指代消解任务中,我们参考宋洋等人的工作^[6]使用全局约束的规则,最后我们定义新的联合两个任务的全局公式。

3.2.1 语义角色标注全局公式

在语义角色标注中,一个节点只可能被标注为一个角色,因此我们有如下的公式:

$$role(pre, a, r_1) \wedge r_1 \neq r_2 \Rightarrow \neg role(pre, a, r_2) \quad (4)$$

此外对于 ARG0~ARG5 来说,在一个谓词中只出现一次。

$$role(pre, a, r_1) \wedge unique(r) \wedge a_1 \neq a_2 \Rightarrow \neg role(pre, a_2, r) \quad (5)$$

其中 *unique(r)* 为全局谓词,其中的常量为 ARG0~ARG5。此外对于 R-* 和 C-* 来说,他们出现时必然有对应的核心角色出现。

$$role(pre, a_1, r_1) \wedge rrole(r_1) \wedge a_1 \neq a_2 \Rightarrow \neg role(pre, a_2, r_2) \wedge unique(r_2) \quad (6)$$

$$role(pre, a_1, r_1) \wedge crole(r_1) \wedge a_1 \neq a_2 \Rightarrow \neg role(pre, a_2, r_2) \wedge unique(r_2) \quad (7)$$

其中 *rrole* 和 *crole* 为全局谓词,其中的常量分别为 R-ARG0~R-ARG5 和 C-ARG0~C-ARG5。

3.2.2 指代消解标注全局公式

在指代消解问题的推导过程中,可以采取最近优先和最好优先两种策略,前者搜索当前提及前面的第一个共指提及,后者搜索前面概率最高的共指提及。在这里我们采用最好优先策略,因此定义下面的全局公式:

$$coref(i, j) \Rightarrow \neg coref(k, j), \forall i, k < j (i \neq k) \quad (8)$$

此外在指代消解的过程中如果不加入任何约束条件,容易出现如下矛盾情况,如对于 3 个提及 *i, j, k* 来说,可能分类的结果是 *coref(i, j)* 和 *coref(i, k)*,但是 *j* 和 *k* 分类为非共指,按照一般的算法判别的结果就是 *i, j, k* 都是指向同一实体,这样就产生了矛盾。为了解决这个问题,Luo 等人^[9]通过使用 Bell-Tree 来搜索可以加入共指链的提及,在这里我们通过加入几条全局公式来约束生成的共指链不存在上述的矛盾关系。

$$\text{coref}(i, j) \wedge \text{coref}(k, j) \wedge i < k < j \Rightarrow (i, k) \quad (9)$$

$$\text{coref}(i, k) \wedge \text{coref}(i, j) \wedge i < k < j \Rightarrow (k, j) \quad (10)$$

$$\text{coref}(i, k) \wedge \text{coref}(k, j) \wedge i < k < j \Rightarrow (i, j) \quad (11)$$

3.2.3 联合任务全局公式

值得注意的是,在我们的语义角色标注任务中,每一个和谓词没有区间交叉的节点都可以认为是谓词的候选标注节点,而在指代消解任务中候选提及一般都是名词短语,并且大部分提及实际上都不是一个语义角色。如果我们直接将两个模型联合起来求解,很有可能在指代的影响下导致大量的非语义角色节点最后标注了语义角色。因此我们首先利用表2的本地公式首先进行一次角色分类,将第一次分类标为语义角色的节点保留为候选节点,然后再联合两个模型起来求解。

首先我们认为如果两个节点是指代关系,则他们更可能标注为同一个角色,于是有如下公式:

$$\text{coref}(a_1, a_2) \wedge \text{role}(pre, a_1, r+) \Rightarrow (pre, a_2, r+) \quad (12)$$

并且我们可以定义的更细致一些,约束到同一类谓词:

$$\text{coref}(a_1, a_2) \wedge \text{cand}(pre, a_1) \wedge \text{cand}(pre, a_2) \wedge \text{role}(pre, a_1, r+) \Rightarrow (pre, a_2, r+) \quad (13)$$

另外我们发现角色相对于谓词的顺序以及谓词是否被动对于标注为 ARG0 还是 ARG1 影响很大,因此添加下面三条公式:

$$\text{coref}(a_1, a_2) \wedge \text{cand}(pre, a_1) \wedge \text{cand}(pre, a_2) \wedge \text{voice}(pre, v_1+) \wedge \text{voice}(pre, v_2+) \wedge \text{role}(pre, a_1, r+) \Rightarrow \text{role}(pre, a_2, r+) \quad (14)$$

$$\text{coref}(a_1, a_2) \wedge \text{cand}(pre, a_1) \wedge \text{cand}(pre, a_2) \wedge \text{position}(pre, a_1, o_1+) \wedge \text{position}(pre, a_2, o_2+) \wedge \text{role}(pre, a_1, r+) \Rightarrow \text{role}(pre, a_2, r+) \quad (15)$$

$$\text{coref}(a_1, a_2) \wedge \text{cand}(pre, a_1) \wedge \text{cand}(pre, a_2) \wedge \text{voice}(pre, v_1+) \wedge \text{voice}(pre, v_2+) \wedge \text{position}(pre, a_1, o_1+) \wedge \text{position}(pre, a_2, o_2+) \wedge \text{role}(pre, a_1, r+) \Rightarrow \text{role}(pre, a_2, r+) \quad (16)$$

此外,如果两个提及都标注为同样的角色,则他们共指的可能性也很大,因此我们考察提及前后两个谓词的标注角色是否一致:

$$\text{role}(pre_1, a_1, r+) \wedge \text{prevPred}(pre_1, a_1) \wedge \text{role}(pre_2, a_2, r) \wedge \text{prevPred}(pre_2, a_2) \Rightarrow \text{coref}(a_1, a_2) \quad (17)$$

$$\text{role}(pre_1, a_1, r+) \wedge \text{nextPred}(pre_1, a_1) \wedge \text{role}(pre_2, a_2, r) \wedge \text{nextPred}(pre_2, a_2) \Rightarrow \text{coref}(a_1, a_2) \quad (18)$$

表2 语义角色任务中使用的本地公式

$\text{cand}(i, j) \wedge \text{voice}(i) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{subCategory}(i, \text{sub}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{phraseType}(i, \text{pt}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{predicateLemma}(i, \text{pl}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{pathLen}(i, \text{plen}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{headWord}(j, \text{hw}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{headPos}(j, \text{hp}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{_rstWord}(j, \text{fw}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{_rstPOS}(j, \text{fp}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{lastWord}(j, \text{lw}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{lastPOS}(j, \text{lp}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{prevWord}(j, \text{pw}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{nextWord}(j, \text{nw}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{prevTwoWord}(j, \text{ptw}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{nextTwoWord}(j, \text{ntw}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{leftSister}(j, \text{ls}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{rightSister}(j, \text{rs}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{leftSisterHeadWord}(j, \text{lshw}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{leftSisterHeadPos}(j, \text{lshp}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{rightSisterHeadWord}(j, \text{rshw}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{rightSisterHeadPos}(j, \text{rshp}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{parent}(j, \text{pa}+) \Rightarrow \text{role}(i, j, r)$
$\text{cand}(i, j) \wedge \text{parentHeadWord}(j, \text{phw}+) \Rightarrow \text{role}(i, j, r)$

续表

$\text{cand}(i,j) \wedge \text{parentHeadPOS}(j, \text{php}+) \Rightarrow \text{role}(i,j,r)$
$\text{cand}(i,j) \wedge \text{ppParentHeadWord}(j, \text{pphw}+) \Rightarrow \text{role}(i,j,r)$
$\text{cand}(i,j) \wedge \text{ppnpHeadWord}(j, \text{ppnphw}+) \Rightarrow \text{role}(i,j,r)$
$\text{cand}(i,j) \wedge \text{ppnpHeadPOS}(j, \text{ppnphp}+) \Rightarrow \text{role}(i,j,r)$
$\text{cand}(i,j) \wedge \text{partialPath}(i,j, \text{ppath}+) \Rightarrow \text{role}(i,j,r)$
$\text{cand}(i,j) \wedge \text{projectPath}(i,j, \text{prop}+) \Rightarrow \text{role}(i,j,r)$
$\text{cand}(i,j) \wedge \text{path}(i,j, \text{path}+) \Rightarrow \text{role}(i,j,r)$
$\text{cand}(i,j) \wedge \text{position}(i,j) \Rightarrow \text{role}(i,j,r)$
$\text{cand}(i,j) \wedge \text{predicateLemma}(i, \text{pl}+) \wedge \text{path}(i,j, \text{path}+) \Rightarrow \text{role}(i,j,r)$
$\text{cand}(i,j) \wedge \text{predicateLemma}(i, \text{pl}+) \wedge \text{headWord}(j, \text{hw}+) \Rightarrow \text{role}(i,j,r)$
$\text{cand}(i,j) \wedge \text{predicateLemma}(i, \text{pl}+) \wedge \text{phraseType}(i, \text{pt}+) \Rightarrow \text{role}(i,j,r)$
$\text{cand}(i,j) \wedge \text{voice}(i) \wedge \text{position}(i,j) \Rightarrow \text{role}(i,j,r)$
$\text{cand}(i,j) \wedge \text{predicateLemma}(i, \text{pl}+) \wedge \text{ppParentHeadWord}(j, \text{pphw}+) \Rightarrow \text{role}(i,j,r)$
$\text{cand}(i,j) \wedge \text{missSub}(i) \wedge \text{path}(i,j, \text{path}+) \Rightarrow \text{role}(i,j,r)$

表 3 指代消解任务中使用的已观察到逻辑谓词

单个提及 m_i 的逻辑谓词	
$\text{type}(i,t)$:提及的类别:命名实体(NAM),代词(PRO),名词(NOM)	
$\text{entityType}(i,e)$:命名实体类型:Person, ORG, LOC 等	
$\text{gdType}(i,g)$:性别类型:MALE, FEMALE, NEUTRAL, UNKNOWN	
$\text{numType}(i,n)$:数量类型:SINGULAR, PLURAL, UNKNOWN	
$\text{hasHed}(i,h)$:提及的中心词	
$\text{firstMention}(i)$:是否是当前句子第一个提及	
$\text{relexive}(i)$:是否反身代词	
$\text{possessive}(i)$:是否所有格	
$\text{de_nite}(i)$:是否确定性名词短语	
$\text{inde_nite}(i)$:是否非确定性名词短语	
$\text{demonstrative}(i)$:是否指示词	
<hr/>	
提及 m_i 和 m_j 关联的逻辑谓词	
$\text{mentionDist}(i,j,m)$:两个提及之间隔了多少个其他提及	
$\text{sentenceDist}(i,j,s)$:两个提及之间隔了多少个句子	
$\text{bothMath}(i,j,b)$:两个提及的性别和数量是否匹配:YES, NO, UNKNOWN	
$\text{strMatch}(i,j)$:字符串是否匹配	
$\text{pronounStrMatch}(i,j)$:都为代词且字符串匹配	
$\text{nopronounStrMatch}(i,j)$:都为非代词且字符串匹配	
$\text{properStrMatch}(i,j)$:都为专有名词且字符串匹配	
$\text{subStrMatch}(i,j)$:存在子串关系	
$\text{headMatch}(i,j)$:中心词匹配	
$\text{animacyMatch}(i,j)$:是否都为动物性词	
$\text{nested}(i,j)$:是否有嵌套关系	
$\text{sameSpeaker}(i,j)$:是否为同一个说话人	
$\text{subStrMatch}(i,j)$:存在子串关系	
$\text{alias}(i,j)$:是否存在别名关系	
$\text{entityTypeMatch}(i,j)$:命名实体类型是否匹配	
$\text{_rstRunMatch}(i,j)$:开源工具的结果是否为指代关系	
$\text{oneSent}(i,j)$:两个提及是否在同一个句子中	

根据篇章向中对齐理论,对于在同一个句子中的提及来说,对于句子中的每一个谓词,如果他们标

注的角色都相同,那么他们共指的可能性也很大^[2]:

$role(pre, a_1, r+) \wedge role(pre, a_2, r+) \wedge$
 $oneSent(a_1, a_2) \Rightarrow coref(a_1, a_2)$ (19)
 由于我们使用了多条联合推理公式, 当数据集

很大时速度很慢, 因此我们使用开源工具先进行初步的指代消解, 然后将构成实体的提及作为我们的候选提及, 采用马尔科夫逻辑网络重新训练学习。

表 4 指代消解任务中使用的本地公式

词汇特征	
$type(i, t1+) \wedge type(j, t2+) \wedge strMatch(i, j) \wedge i \neq j \Rightarrow coref(i, j)$	
$type(i, t1+) \wedge type(j, t2+) \wedge pronounStrMatch(i, j) \wedge i \neq j \Rightarrow coref(i, j)$	
$type(i, t1+) \wedge type(j, t2+) \wedge properStrMatch(i, j) \wedge i \neq j \Rightarrow coref(i, j)$	
$type(i, t1+) \wedge type(j, t2+) \wedge nopronounStrMatch(i, j) \wedge i \neq j \Rightarrow coref(i, j)$	
$type(i, t1+) \wedge type(j, t2+) \wedge headMatch(i, j) \wedge i \neq j \Rightarrow coref(i, j)$	
$type(i, t1+) \wedge type(j, t2+) \wedge subStrMatch(i, j) \wedge i \neq j \Rightarrow coref(i, j)$	
$head(i, h1+) \wedge head(j, h2+) \wedge i \neq j \Rightarrow coref(i, j)$	
语法特征	
$type(i, t1+) \wedge type(j, t2+) \wedge gdType(i, g1+) \wedge gdType(j, g2+) \wedge i \neq j \Rightarrow coref(i, j)$	
$type(i, t1+) \wedge type(j, t2+) \wedge numType(i, n1+) \wedge numType(j, n2+) \wedge i \neq j \Rightarrow coref(i, j)$	
$type(i, t1+) \wedge type(j, t2+) \wedge bothMatch(i, j, b+) \wedge i \neq j \Rightarrow coref(i, j)$	
$type(i, t1+) \wedge type(j, t2+) \wedge animacy(i, j) \wedge i \neq j \Rightarrow coref(i, j)$	
$type(i, t1+) \wedge type(j, t2+) \wedge nested(i, j) \wedge i \neq j \Rightarrow coref(i, j)$	
$(type(i, t1+) \vee type(j, t2+)) \wedge i \neq j \Rightarrow coref(i, j)$	
$(reexive(i) \vee reexive(j)) \wedge i \neq j \Rightarrow coref(i, j)$	
$(possessive(i) \vee possessive(j)) \wedge i \neq j \Rightarrow coref(i, j)$	
$(definite(i) \vee de_nite(j)) \wedge i \neq j \Rightarrow coref(i, j)$	
$(indefinite(i) \vee inde_nite(j)) \wedge i \neq j \Rightarrow coref(i, j)$	
$(demonstrative(i) \vee demonstrative(j)) \wedge i \neq j \Rightarrow coref(i, j)$	
距离特征	
$type(i, t1+) \wedge type(j, t2+) \wedge menDist(i, j, m1+) \wedge i \neq j \Rightarrow coref(i, j)$	
$type(i, t1+) \wedge type(j, t2+) \wedge sentDist(i, j, s+) \wedge i \neq j \Rightarrow coref(i, j)$	
$(firstMention(i) \vee firstMention(j)) \wedge i \neq j \Rightarrow coref(i, j)$	
语义特征	
$type(i, t1+) \wedge type(j, t2+) \wedge alias(i, j) \wedge numType(i, n2+) \wedge i \neq j \Rightarrow coref(i, j)$	
$type(i, t1+) \wedge type(j, t2+) \wedge sameSpeaker(i, j) \wedge i \neq j \Rightarrow coref(i, j)$	
$type(i, t1+) \wedge type(j, t2+) \wedge entityTypeMatch(i, j) \wedge i \neq j \Rightarrow coref(i, j)$	
$(entityType(i, e1+) \vee entityType(j, e2+)) \wedge i \neq j \Rightarrow coref(i, j)$	
$firstRunMatch(i, j) \Rightarrow coref(i, j)$	

4 实验设计

4.1 实验数据

由于常用的语义角色标注库 PropBank^[17] 数据集没有指代标注, 我们选用 OntoNote 数据集作为我们的实验数据。我们使用 CoNLL-2012^[18] 的数据切分进行实验对比, 由于我们的系统不需要调节参数, 因此在实验部分本文只给出测试集上的实验结果。

对于语义角色标注任务我们采用准确率、召回率和 F 值三个评价指标; 而对于指代消解我们采用 CoNLL-2012 使用的 MUC^[19], BCUB^[20], CEAF^[21], BLANC^[22] 以及 MUC, BCUB 和基于实体级别的 CEAFE 三者的平均值 Avg 作为评价指标。

此外我们使用开源工具 *thebeast*^① 进行马尔科夫逻辑网训练和推导, 并且使用平均感知机^[23] 算法, 迭代 10 轮进行参数训练, 其他都使用默认参数。

① <http://code.google.com/p/thebeast/>

4.2 基准系统

4.2.1 语义角色标注基准系统

我们首先利用表 2 使用的本地特征以及公式(4)、(5)、(6)使用 *thebeast* 进行一次语义角色标注任务,以此来生成联合标注的候选节点。表 5 给出了测试集上使用本地特征在标准句法分析结果和自动句法分析结果上面进行标注的性能,以及不考虑标注的结果是否和参考答案匹配,仅仅考虑其是否为一个语义角色的性能,其中系统上限为基准系统标注的语义角色节点作为候选节点可能达到的性能上限。

表 5 语义角色标注基准系统性能

	准确率 /%	召回率 /%	F 值 /%
人工标注句法树基准系统	86.82	85.53	86.17
Che 等 ^[7]			84.36
人工标注句法树基准系统上限	99.92	99.10	99.51
自动标注句法树基准系统	74.12	72.58	73.34
Pradhan 等 ^[18]	81.47	61.56	70.13
自动标注句法树基准系统上限	99.95	98.09	99.01

虽然到目前为止,还没有相关的研究工作汇报

表 6 指代消解基准对比系统性能

	提及识别(P,R,F)/%			MUC/%	BCUB/%	CEAFM/%	CEAFE/%	BLANC/%	Avg/%
<i>sys1</i>	78.5	73.9	76.1	66.79	70.17	59.42	47.08	75.56	61.35
<i>sys2</i>	75.3	72.2	73.7	64	68.97	57.67	45.38	74.78	59.45
<i>sys3</i>	79.1	75.2	77.1	67.91	70.7	60.3	47.95	75.79	62.19
<i>sys4</i>	76.0	73.5	74.7	65.17	69.5	58.53	46.19	75.01	60.29

从表 6 中可以看到我们使用人工标注和自动标注的结果分别进行指代消解,自动标注的信息对于指代消解的性能还是略微有影响,差距大概在 1~2 个百分点左右,和 Pradhan 等人的结论相当^[18]。不过总体而言,Stanford-CoreNLP 的水平还是处于世界最好水平之一,使用自动标注时,MUC, BCUB, CEAFE 三个指标的平均得分 59.45%比 CoNLL-2012 的系统最好水平 63.37%^[24]相差 4 个点,但是与其他系统性能水平相差不是很大^[18],差距也主要在于 Stanford-CoreNLP 是 2011 年设计的系统,CoNLL-2012 较 CoNLL-2011 系统水平有了很大改进。不过我们采用二次消解的基准系统仍然取得了 60.29%的平均水平相比于 Stanford-CoreNLP 仍

在 CoNLL-2012 上面语义角色标注的性能,但是我们可以参考 CoNLL-2012 官方报告^[18]给出的自动句法分析树上面标注的语义角色 F 值为 70.13%,低于我们的标注性能,他们使用的标注器是 *Assert*^① 开源标注器,并且在 OntoNote 上面使用训练集重新训练,但是最后的性能还是比我们的基准系统要低。此外我们可以参考 Che 等人^[7]利用 Onto-Note3.0 进行联合语义角色标注和语义消歧的工作,他们将人工标注的句法树转为依存树进行标注,使用他们自己的切分方案得到的语义角色标注最好 F 值为 84.36%。从对比性能上来看,我们的基准系统很强,因此基于此对比系统进行的性能提升很有说服力。

4.2.2 指代消解基准系统

此外对于指代消解,我们也先使用 Stanford-CoreNLP^② 进行初步指代消解任务,然后在其基础上我们使用表 4 的本地特征以及公式(8)、(9)使用 *thebeast* 进行更进一步的指代消解任务,表 6 给出了系统的提及识别性能以及 5 个指代消解评价指标的得分,其中 *sys1* 为使用全部人工标注的句法树、命名实体识别等信息,*sys2* 为使用自动标注的识别信息。*sys3* 为使用 *sys1* 的结果进行马尔科夫逻辑网训练推导的结果,*sys4* 为使用 *sys2* 的结果进行马尔科夫逻辑网训练推导的结果。

然有 0.8 个百分点左右的性能提升。并且我们的方法可以应用于任何单系统,不局限于使用某一个开源工具。

最重要的一点是,通过单独进行语义角色标注和指代消解两个任务,我们可以初步过滤掉大部分候选节点,并且保持非常高的系统性能上限(指代消解任务的提及识别召回率即为联合推导系统上限),因此在后面的联合推导模型中将在去除大量噪音的基础上推导出更好的结果。

① <http://cemantix.org/assert.html>

② <http://www-nlp.stanford.edu/software/corenlp.shtml>

4.3 实验结果

在前面一章中提到,本文一共使用了8条联合推导规则,其中前5条是指代作用于语义角色标注,后3条相反。根据使用公式的不同,我们对如下几个系统进行实验,需要说明的是,我们的实验都是基于自动标注的结果上进行的,因为人工标注的结果对于实际应用来说用处不是很大,而上文给出的人工标注结果只是作为和其他研究工作对比的参考依据。

- *sys1*: 使用公式(12)。
- *sys2*: 使用公式(13)。
- *sys3*: 使用公式(14)~(16)。

- *sys4*: 使用公式(17)~(18)。
- *sys5*: 使用公式(19)。
- *sys6*: 使用公式(12)~(16)。
- *sys7*: 使用公式(17)~(19)。
- *sys8*: 使用所有联合推导公式。
- *gold1*: 类似系统 *sys6* 并且使用人工标注的指代进行语义角色标注。
- *gold2*: 类似系统 *sys7* 并且使用人工标注的语义角色标注进行指代消解。

我们先进行系统 *gold1* 和 *gold2* 的实验,用于查看我们的系统所能达到的最好性能。表7给出了两个系统的语义角色标注和指代消解的结果。

表7 使用列表4.3的 *gold1* 和 *gold2* 系统进行的实验结果

	指代消解							SRL
	提及识别/%	MUC/%	BCUB/%	CEAFM/%	CEAFE/%	BLANC/%	Avg/%	F值/%
<i>gold1</i>								75.67
<i>gold2</i>	77.36	68.98	72.12	62.13	48.93	77.84	63.34	

从表7中我们可以看到指代消解可以帮助语义角色标注性能达到75.67%,相比于表5基准系统73.34%的性能有了2.3%的F值提升,主要提升来自于篇章级别的语义角色标注一致性,避免了指代相同的提及分配的角色不一致的问题。此外使用语义角色标注改进指代消解在CoNLL-2012综合得分可以达到63.34%,接近于当前数据集的最好实验水平,不过我们使用的是人工标注的语义角色标注结果。实验的结果也基本和以往的研究工作结论相近^[1-2]。此外我们发现从两者的贡献度来分析,语义

角色标注对于指代消解的帮助更大,可以解释的原因是语义角色本身已经具有比较高的正确率,并且指代消解的贡献多集中于ARG0, ARG1以及ARGM, TMP这几个数量较多的角色,然而基准系统对于这几个角色的标注准确率已经性能不错了,因此提升空间受限。尽管如此,能够提升2+的性能对于语义角色标注来说已经是很显著的性能提升。

我们继续对本文提出的8条联合推导公式进行单独和联合的实验,表8给出了详细的实验结果。

表8 使用列表4.3的系统进行的对比实验结果

	指代消解							SRL
	提及识别/%	MUC/%	BCUB/%	CEAFM/%	CEAFE/%	BLANC/%	Avg/%	F值/%
<i>sys1</i>	74.72	65.17	69.5	58.53	46.19	75.01	60.28	73.85
<i>sys2</i>	74.72	65.17	69.5	58.53	46.19	75.01	60.28	74.17
<i>sys3</i>	74.72	65.17	69.5	58.53	46.19	75.01	60.28	74.32
<i>sys4</i>	75.56	66.33	70.22	59.56	47	75.76	61.18	73.34
<i>sys5</i>	75.31	65.99	70.01	59.25	46.77	75.61	60.92	73.34
<i>sys6</i>	74.72	65.17	69.5	58.53	46.19	75.01	60.28	74.51
<i>sys7</i>	75.89	66.84	70.58	60.01	47.33	76.05	61.58	73.34
<i>sys8</i>	76.15	67.17	70.79	60.31	47.65	76.23	61.85	74.99

从表8的结果中首先我们可以发现本文提出的8条联合推导规则在自动标注的情况下仍然有效。其次由于我们优化时是单独对隐含变量 *role* 和

coref 进行优化的,所以只有混合了所有规则的 *sys8* 才同时改变了两个任务的性能,而前面的组合只能单独改变某一任务性能。另外从性能提升上面来

看,其中在自动指代消解的情况下,可以提升语义角色标注最多 1.2 个点,当联合起来推导时可以提升 1.65 个点;反方向的实验结果也类似,当使用联合推导时性能依然能够提升。但是我们也可以发现,语义角色标注最好的结果为 74.99%,低于可能达到的最好性能 75.67%,而指代的差距就更大了,最好的结果三个指标平均值只有 61.85%,远低于可能达到的最好性能 63.34%。最大原因在于由于我们是联合篇章级别的语义角色标注,因此一类谓词的标注错误,将导致同一链条上的指代错误,所以由于角色标注本身性能并不是很好,最后能提升的结果距离性能上限有所差距。

总体而言,使用马尔科夫逻辑网络联合推导后,两个任务的性能都有了显著的提升,由于语义角色标注本身的基准系统已经是已知研究工作中达到的最好结果,所以在其基础上所做的提升仍然是单系统的最好性能。而指代消解的最终结果也仅次于今年 CoNLL-2012 的最好系统性能,具有很强的可对比性。

5 本文总结和未来展望

在本文中,我们提出了利用马尔科夫逻辑网络进行联合语义角色标注和指代消解的任务。通过借鉴以往的研究工作,以及本文综合创新提出的 8 条联合推导规则后,本文的实验证明,通过分析篇章级别的语义角色之间的指代关系,我们能够显著的提升角色标注的性能;此外通过语义角色标注的特征用于帮助指代消解任务,实验也验证了前人研究工作提出的相关结论。本文最后通过联合两个任务的特征,使用联合马尔科夫逻辑规则推导,在两个任务中都得到了性能的显著提升。此外本章的另外贡献在于给出了在 OntoNote5.0 数据集上使用 CoNLL-2012 切分方案所能达到的语义角色标注最好性能,以及具有性能可对比性的指代消解实验结果。

在未来的工作中我们将在中文中尝试联合标注,并且尝试不同语种之间的联合学习标注。

参考文献

[1] Simone Paolo Ponzetto and Michael Strube. Exploiting semantic role labeling, wordnet and Wikipedia for coreference resolution[C]//Proceedings of the Main Conference on Human Language Technology Confer-

ence of the North American Chapter of the Association of Computational Linguistics. Sydney, Australia: ACL Publication Chairs, 2006: 192-199.

- [2] Fang Kong, GuoDong Zhou, and Qiaoming Zhu. Employing the centering theory in pronoun resolution from the semantic perspective[C]//Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing. Singapore: ACL Publication Chairs, 2006: 987-996.
- [3] Eduard Hovy, Mitchell Marcus, Martha Palmer, et al. Ontonotes: the 90% solution[C]//Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers. Sydney, Australia: ACL Publication Chairs, 2006: 57-60.
- [4] Matthew Richardson and Pedro Domingos. Markov logic networks[J]. Machine learning, 2006, 62(1): 107-136.
- [5] Hoifung Poon and Pedro Domingos. Joint unsupervised coreference resolution with markov logic[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Waikiki, Honolulu, Hawaii: EMNLP Publication Chairs, 2008: 650-659.
- [6] Yang Song, Jing Jiang, Wayne Xin Zhao, et al. Joint learning for coreference resolution with markov logic[C]//Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Jeju Island, Korea: EMNLP Publication Chairs, 2012: 1245-1254.
- [7] Wanxiang Che and Ting Liu. Jointly modeling wsd and srl with markov logic[C]//Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010). Beijing, China: Coling Publication Chairs, 2010: 161-169.
- [8] Kristina Toutanova, Aria Haghighi, and Christopher D Manning. A global joint model for semantic role labeling[J]. Computational Linguistics, 2008, 34(2): 161-191.
- [9] Xiaoqiang Luo, Abe Ittycheriah, Hongyan Jing, et al. A mention synchronous coreference resolution algorithm based on the bell tree[C]//Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04). Barcelona, Spain: ACL Publication Chairs, 2006: 135-142.
- [10] Daniel Gildea and Daniel Jurafsky. Automatic labeling of semantic roles [J]. Computational Linguistics, 2002, 28(3): 245-288.
- [11] Mihai Surdeanu, Sanda Harabagiu, John Williams, and Paul Aarseth. Using predicate-argument structures for information extraction[C]//Proceedings of the 41st Annual Meeting on Association for Computa-

- tional Linguistics-Volume 1. Sapporo, Japan: ACL Publication Chairs, 2003: 8-15.
- [12] Sameer Pradhan, Wayne Ward, Kadri Hacioglu, et al. Shallow semantic parsing using support vector machines[C]//Proceedings of HLT/NAACL. Boston, USA: NAACL Publication Chairs, 2004: 233.
- [13] 刘挺,车万翔,李生. 基于最大熵分类器的语义角色标注[J]. 软件学报, 2007,18(3): 565-573.
- [14] Wee Meng Soon, Hwee Tou Ng, and Daniel Chung Yong Lim. A machine learning approach to coreference resolution of noun phrases[J]. Computational linguistics, 2001, 27(4): 521-544.
- [15] Vincent Ng and Claire Cardie. Improving machine learning approaches to coreference resolution[C]//Proceedings of 40th Annual Meeting of the Association for Computational Linguistics. Philadelphia, Pennsylvania, USA: ACL Publication Chairs, 2002: 104-111.
- [16] Emili Sapena, Lluís Padró, and Jordi Turmo. Relaxor participation in conll shared task on coreference resolution[C]//Proceedings of the 15th Conference on Computational Natural Language Learning: Shared Task. Portland, Oregon, USA: ACL Publication Chairs, 2011: 35-39.
- [17] Martha Palmer, Daniel Gildea, and Paul Kingsbury. The proposition bank: An annotated corpus of semantic role[J]. Computational Linguistics, 2005, 31(1): 71-106.
- [18] Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, et al. Conll-2012 shared task: Modeling multi-lingual unrestricted coreference in ontonotes[C]//Joint Conference on EMNLP and CoNLL-Shared Task. Jeju Island, Korea: ACL Publication Chairs, 2012: 1-40.
- [19] Marc Vilain, John Burger, John Aberdeen, et al. A model theoretic coreference scoring scheme[C]//Proceedings of the 6th conference on Message understanding. MIT, Cambridge, Massachusetts, USA, Proceedings. Morgan Kaufmann Publisher, 1995: 45-52.
- [20] Amit Bagga and Breck Baldwin. Algorithms for scoring coreference chains[C]//The first international conference on language resources and evaluation workshop on linguistics coreference. Granada, Spain: LREC Publication Chairs, 1998: 563-566.
- [21] Xiaoqiang Luo. On coreference resolution performance metrics[C]//Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing. Michigan, USA: ACL Publication Chairs, 2005: 25-32.
- [22] Marta Recasens and Eduard Hovy. Blanc: Implementing the rand index for coreference evaluation[J]. Natural Language Engineering, 2011, 17(4): 485.
- [23] Michael Collins and Terry Koo. Discriminative reranking for natural language parsing[J]. Computational Linguistics, 2005, 31(1): 25-70.
- [24] Eraldo Fernandes, et al. Latent structure perceptron with feature induction for unrestricted coreference resolution[C]//Joint Conference on EMNLP and CoNLL-Shared Task. Jeju Island, Korea: ACL Publication Chairs, 2012: 41-48.