

# 依赖关系三元组匹配在问题回答中的应用

林 锋<sup>1</sup> 周雅倩<sup>1</sup> 黄萱菁<sup>1</sup> 吴立德<sup>1</sup>

**摘 要** 描述了一种新的计算问题与支持答案句相似度的方法,即基于依赖关系三元组匹配的方法.该方法引入了问题中的疑问性和非疑问性部分的信息,采用了启发式规则扩展问题的依赖关系三元组,从而匹配变形的答案句.同时把问题与支持答案句的相似度作为新的特征,应用于开放领域的问题回答(Question answering, QA)任务中的答案排序.实验结果表明,引入新特征的答案排序方法与通常的基于密度的方法相比,在相对精度指标上提高了 8.2%,在平均排序倒数(Mean reciprocal rank, MRR)评价上提高了 8%.

**关键词** 问题回答, 答案排序, 依赖关系三元组  
**中图分类号** TP391

## Dependency Relation Triples Matching for Question Answering

LIN Feng<sup>1</sup> ZHOU Ya-Qian<sup>1</sup> HUANG Xuan-Jing<sup>1</sup> WU Li-De<sup>1</sup>

**Abstract** This paper presents a new method to compute the similarity between question and answer sentences, namely dependency relation triples matching. This method considers the information of question's interrogative part and non-interrogative part, and heuristic rules are used to expand question's relation triples to match metamorphosing answer sentences. Then, this similarity score is used as a new feature for answer ranking in open domain question answering (QA) track. The experiments show the new answer ranking method outperforms the common density-based approach by up to 8.2% in relative precision and 8% in mean reciprocal rank (MRR) evaluation.

**Key words** Question answering, answering ranking, dependency relation triple

目前,面向开放域的问题回答系统(Question answering, QA)得到了人们的普遍关注.与传统的文本检索系统相比,问题回答系统输入的是一个自然语言描述的问题,而返回的是一个确切的答案.问题回答系统一般包含四个主要的模块:问题分析,文档检索,答案抽取和答案排序.问题分析模块分析问题的类型并且生成查询;文档检索模块根据查询检索,并且返回相关的文档;答案抽取模块从返回的文档中抽取出若干候选答案,每个候选答案有一组包含这个答案的句子,这些句子称为支持答案句;答案排序模块对这些候选答案进行评分排序,把排在最前面的答案返回给用户.答案排序模块的优劣直接决定了问题回答系统的性能.目前的答案排序方法基本上是通过计算问题与支持答案的句子的相似度,给候选答案评分.根据计算问题与支持答案句相似度的方法的不同,答案排序方法包括基于密度的答案排序和基于语义相似度的答案排序.

基于密度的答案排序是一种传统的方法,在计算问题与支持答案句的相似度时,考察问题中的成

分在支持答案句中出现的成分,以及候选答案与这些成分的距离<sup>[1]</sup>.

基于语义相似度的答案排序是通过语义的层次来计算问题与支持答案句之间的相似,主要包括基于词的语义相似度计算和基于依赖关系的语义相似度计算.基于词的语义相似度计算主要通过计算问题中非停用词与支持答案句中非停用词的语义相似,从而计算问题与支持答案句的相似度<sup>[2]</sup>.通常利用 wordnet<sup>[3]</sup> 计算词与词之间的语义相似.基于依赖关系的语义相似度的计算考察的是问题与支持答案句中依赖关系的匹配.文献[4-5]从句法分析器的输出中抽取依赖关系,文献[6]从依赖关系树中抽取词与词之间的依赖路径,从而计算问题与支持答案句的相似度.但是,它们都没有挖掘问题的疑问部分的信息.

其他的答案排序方法还有基于依赖关系树的匹配和逻辑推理.文献[7]用依赖关系树分别描述问题与支持答案句,它用最大树匹配算法计算这两棵树之间的编辑距离,从而计算问题与支持答案句之间的相似度.这种方法的缺陷是由于采用严格匹配的方法,因此无法计算变形的答案句与问题之间的相似度.文献[8]采用逻辑推理的方法,通过推理证明候选答案是否是正确答案.这种方法的缺陷是计算量大,代价高.

在计算问题与支持答案句相似度的方法中:传统的基于密度的方法约束太松<sup>[1]</sup>;基于词的语义相

收稿日期 2007-07-26 收修改稿日期 2008-03-11  
Received July 26, 2007; in revised form March 11, 2008  
国家自然科学基金(60435020, 60503070)资助  
Supported by National Natural Science Foundation of China (60435020, 60503070)  
1. 复旦大学计算机科学与工程系 上海 200433  
1. Department of Computer Science and Engineering, Fudan University, Shanghai 200433  
DOI: 10.3724/SP.J.1004.2008.01410

似度计算没有考虑词与词之间的依赖关系<sup>[2]</sup>; 现有基于依赖关系的语义相似度的计算没有挖掘问题疑问部分所含有的信息<sup>[4-6]</sup>, 而基于依赖关系树的硬匹配约束太紧<sup>[7]</sup>; 基于逻辑推理的方法成本过高<sup>[8]</sup>.

本文介绍一种新的计算问题与支持答案句之间的相似度的方法, 即依赖关系三元组匹配的方法来计算问题与支持答案句之间的相似度. 该方法考虑问题中疑问部分与非疑问部分的信息, 并且用启发式规则对问题中的依赖关系三元组进行扩展, 把问题与支持答案句分别用一群依赖关系三元组表示, 通过计算这两群依赖关系三元组的匹配来计算问题与支持答案句的相似度. 然后, 把这种相似度作为一种新的特征, 同时加上浅层的句法特征, 语法特征及其来自于 Web 的特征, 利用基于优先级的方法和机器学习的方法进行答案排序, 并对这两种方法进行比较.

## 1 问题/支持答案句之间依赖关系三元组匹配

问题 ( $Q$ ) 由疑问部分 ( $IQ$ ) 和陈述部分 ( $SQ$ ) 组成. 疑问部分包括疑问词和中心词. 疑问词表明问题所问的是什么; 中心词是答案的上位词, 表明答案是什么. 并不是所有的问题都有中心词. 除疑问部分之外, 剩下的属于陈述部分. 支持答案句 ( $AS$ ) 由答案部分 ( $A$ ) 和其他部分 ( $OA$ ) 组成. 答案部分对应于问题中的疑问部分, 而其他部分则对应于问题中的陈述部分. 比如, 在问题 “What country are Volvo automobiles made in?” 中, “What country” 是疑问部分, 它包括了疑问词 “What” 和中心词 “country”, “are Volvo automobiles made in” 是陈述部分. 该问题的答案是 “Sweden”, 支持答案句是 “Volvo automobiles are made in Sweden”. 在支持答案句中 “Sweden” 属于答案部分, 剩下的属于其他部分.

下面我们首先介绍三元关系组匹配的原理, 然后介绍三元关系组的抽取, 最后具体介绍三元关系组匹配的算法.

### 1.1 依赖关系三元组匹配的原理

给定一个问题或者一个支持答案句, 我们从句法分析器 `minipar`<sup>[9]</sup> 的输出中抽取依赖关系三元组. 一个依赖关系三元组定义为

$$T = (\text{slot}_1, \text{rel}, \text{slot}_2) \quad (1)$$

它由两个槽 ( $\text{slot}$ ) 和一个依赖关系 ( $\text{rel}$ ) 组成. 其中, 每个槽对应着一个短语或者实体名, 槽 ( $\text{slot}_1$ ) 与槽 ( $\text{slot}_2$ ) 之间的依赖关系是  $\text{rel}$ , 并且槽 ( $\text{slot}_1$ ) 管辖槽 ( $\text{slot}_2$ ). 在 `minipar` 中定义了 60 多种依赖

关系, 比如 *sub*, *obj* 等.

本文定义了四类依赖关系三元组:

1) 在问题中疑问部分与陈述部分之间的依赖关系三元组, 我们称为与疑问相关的依赖关系三元组:  $R_{ISQ}(IQ, SQ)$ .

2) 在问题中陈述部分内部的依赖关系三元组, 我们称为与疑问非相关的依赖关系三元组:  $R_{SQ}(SQ, SQ)$ .

3) 在支持答案句中, 答案部分与其他部分之间的依赖关系三元组, 我们称为与答案相关的依赖关系三元组:  $R_{AOA}(A, OA)$ .

4) 在支持答案句中, 其他部分内部的依赖关系三元组, 我们称为与答案非相关的依赖关系三元组:  $R_{OA}(OA, OA)$ .

若干依赖关系三元组集合在一起成为一个依赖关系三元组群. 一个问题包括与疑问相关的依赖关系三元组群和与疑问非相关的依赖关系三元组群. 一个支持答案句包括与答案相关的依赖关系三元组群和与答案非相关的依赖关系三元组群. 本文通过计算问题与支持答案句的依赖关系三元组群的匹配计算两者的相似性.

本文定义了两类依赖关系三元组群的匹配:

1) 问题中与疑问相关的依赖关系三元组群  $R_{ISQ}$  和支持答案句中与答案相关的依赖关系三元组群  $R_{AOA}$  之间的匹配:  $Match(R_{ISQ}, R_{AOA})$ .

2) 问题中的与疑问非相关的依赖关系三元组群  $R_{SQ}$  和支持答案句中与答案非相关的依赖关系三元组群  $R_{OA}$  之间的匹配:  $Match(R_{SQ}, R_{OA})$ .

**定义 1.** 依赖关系三元组的匹配: 给定两个依赖关系三元组:  $T_1 = (\text{slot}_{11}, \text{rel}_1, \text{slot}_{12})$ ,  $T_2 = (\text{slot}_{21}, \text{rel}_2, \text{slot}_{22})$ . 如果  $Match(\text{slot}_{11}, \text{slot}_{21})$ ,  $Match(\text{slot}_{12}, \text{slot}_{22})$ , 并且  $\text{rel}_1 = \text{rel}_2$ , 那么  $Match(T_1, T_2) = 1$ , 否则  $Match(T_1, T_2) = 0$ . 其中,  $Match(\text{slot}_i, \text{slot}_j)$  表示:  $\text{slot}_i = \text{slot}_j$  或者  $\text{slot}_i \in \text{Synset}(\text{slot}_j)$  或者  $\text{slot}_j \in \text{Synset}(\text{slot}_i)$ ,  $\text{Synset}(\text{slot})$  是在 WordNet 中定义的  $\text{slot}$  的同义词集.

**定义 2.** 依赖关系三元组群的匹配: 给定两个依赖关系三元组群:  $R_1 = \{T_1, T_2, \dots, T_m\}$ ,  $R_2 = \{T'_1, T'_2, \dots, T'_n\}$ .  $R_1$  与  $R_2$  之间的匹配  $Match(R_1, R_2) = \sum_{i=1}^m (\omega_i \cdot Match(T_i, R_2))$ , 其中  $\omega_i$  是  $T_i$  的权重. 如果  $\exists T'_k \in R_2 (1 \leq k \leq n)$  并且  $Match(T_i, T'_k) = 1$ , 那么  $Match(T_i, R_2) = 1$ . 否则,  $Match(T_i, R_2) = 0$ .  $\omega_i$  定义为

$$\omega_i = \omega_{s1} + \omega_{s2} \quad (2)$$

$$\omega_{si} = \begin{cases} 0.5, & slot_i \text{ is answer} \\ 0.4, & slot_i \text{ is question's main verb} \\ 0.3, & slot_i \text{ is proper noun phrase} \\ 0.2, & \text{others} \end{cases} \quad (3)$$

我们评价一个候选答案是否正确回答了问题 ( $Q$ ), 可以通过考察该答案的一组支持答案句 ( $AS$ ) 与问题的相似度  $c$ . 如果两者之间的相似度越高, 则问题与支持答案句之间越相似, 从而候选答案的置信度越高.

$$c = Sim(Q, AS) = \max_i Sim(Q, AS_i) \quad (4)$$

其中  $AS_i$  是第  $i$  个支持答案句,  $Sim(Q, AS_i)$  是问题与  $AS_i$  支持答案句之间的相似度.

$$Sim(Q, AS_i) = Match(R_Q, R_{AS}^i) = Match(R_{ISQ}, R_{AOA}^i) + Match(R_{SQ}, R_{OA}^i) \quad (5)$$

## 1.2 问题分类

通常我们根据答案类型对问题进行分类. 为了更好地从不同的问题中抽取依赖关系三元组, 我们从三个方面对问题进行分类: 1) 根据问题的疑问部分; 2) 根据问题的主动词; 3) 根据问题的语态.

对于不同的问题, 我们根据其疑问部分预测候选答案将会在支持答案句中所作的成分, 如表 1 所示. 例如, 问题是 “What is the currency of Denmark?” 候选答案是 “Danish krone”. 该候选答案在支持答案句 “Danish krone is the currency of Denmark” 中作主语, 而在支持答案句 “The currency of Denmark is Danish krone” 中作表语.

对于不同的问题, 我们根据问题的主动词将问题分为两类: BE 主动词问题和 DO 主动词问题. BE 主动词问题的主动词是一个系动词, 如 am, is, are 等. DO 主动词问题的主动词是一个主动或者被动的动词.

根据问题的语态, 我们将问题分为主动问题与被动问题.

表 1 问题中疑问部分与候选答案在支持答案句中的语义角色对应表

Table 1 The semantic mapping table between question interrogative part and candidate answer

疑问部分	语法成分	举例
What/Which	subject, object, appo	What is the currency of Denmark?
What/Which + NP	subject, object, appo	What country is Mt. Everest in?
Who/Whom	subject, object, appo	Who was Horus' father?
Prep + what/which + NP	modifier	In what country did Khmer Rouge take place?
Prep + whom	subject	By whom were Harlem Globe Trotters founded?

## 1.3 依赖关系三元组的扩展

我们把依赖关系三元组的扩展分为关系 ( $rel$ ) 的扩展和槽 ( $slot$ ) 的扩展. 在一个依赖关系三元组中, 关系的扩展只对依赖关系进行扩展, 槽的扩展只对槽进行扩展. 本文只讨论对问题的依赖关系三元组的扩展, 因为一个问题对应若干个支持答案句, 扩展问题的依赖关系三元组才能更好地匹配不同变形的支持答案句. 下面, 我们将讨论问题中与疑问相关的依赖关系三元组的扩展和与疑问非相关的依赖关系三元组的扩展.

### 1.3.1 与疑问相关的依赖关系三元组的扩展

通常疑问词在问题中的语义角色与候选答案在支持答案句中的语义角色不一样, 所以从问题疑问部分抽取的三元关系组很难与从支持答案句中的答案部分抽取的三元关系组匹配. 例如在例 1 中, 支持答案句正确回答了问题. 但是疑问词 “What” 在问题中充当了主语, 而候选答案 “red” 在支持答案句中充当了表语, 两者所对应的依赖关系三元组无法由定义 1 匹配, 所以我们要对问题的与疑问相关的依赖关系三元组进行扩展.

#### 例 1. 依赖关系三元组举例

Q: What is Crips' gang color?

$R_{ISQ}$ : (color, subject, what)

$R_{SQ}$ : (be, pred, color), (color, gen, Crips)

A: red

AS: Crips' gang color is red.

$R_{AOA}$ : (red, subject, color)

$R_{OA}$ : (be, pred, color), (color, gen, Crips)

在第 1.2 节中, 我们从三个方面对问题进行分类. 对于不同类型的问题, 我们用不同的疑问部分三元关系组的扩展方法.

在本节中, 我们将讨论 “What” 问题的扩展方法, 其他类的问题可以用相同的方法进行扩展. 对于主动语态的 “What” 问题, 我们主要进行如下三类的扩展:

### 1) What + BE + ACTIVE

这类“*What*”问题的主动词是系动词, 语态是主动语态. 对于图 1 中的例子, 我们参考表 1, 预测候选答案在支持答案句中会充当 *subject* 或者 *object* 或者 *appo* 的语义角色. 疑问词“*What*”对应支持答案句中的答案部分. 我们把候选答案用记号 *A* 标记, 那么疑问词“*What*”也可以用 *A* 替代. 根据启发式规则, 问题中与疑问相关的依赖关系三元组群  $R_{ISQ}$  可以扩展为:  $\{(color, subject, A), (A, subject, color), (A, appo, color), (color, appo, A)\}$ . 这样  $(A, subject, color)$  可以与支持答案句中的  $(red, subject, color)$  匹配.

### 2) What + NP + BE + ACTIVE

这类“*What*”问题有中心词, 主动词是 BE 动词, 而且它的语态是主动语态, 该类问题的与疑问相关的依赖关系三元组扩展如下. 例如, 问题“*What nationality is architect Frank Gehry?*”中, 与疑问相关的依赖关系三元组群是  $\{(nationality, det, what), (Frank Gehry, subject, nationality)\}$ . 正确的候选答案应该是中心词“*nationality*”的下位词, 而且该中心词通常不出现在支持答案句中, 所以把问题中与疑问相关的依赖关系三元组群中的中心词用 *A* 替换, 同时根据表 1 对三元关系组进行扩展为:  $\{(Frank Gehry, subject, A), (A, subject, Frank Gehry)\}$ .

### 3) What + NP + DO + ACTIVE

这类“*What*”问题有中心词, 主动词是 DO 动词, 语态是主动语态. 例如, 问题“*What film introduced Jar Jar Binks?*”中, 与疑问相关的依赖关系三元组群是  $\{(film, det, what), (introduce, subject, film)\}$ . 同样, 我们将它扩展为:  $\{(introduce, subject, A), (introduce, by, A)\}$ .

同理, 我们可以采用相似的方法对被动语态的“*What*”问题进行依赖关系三元组扩展.

#### 1.3.2 与疑问非相关的依赖关系三元组的扩展

通常问题的陈述部分在支持答案句中也会变形, 这样两者抽取的依赖关系三元组无法正确匹配. 例如, 问题“*Who founded the Black Panthers organization?*”中与疑问非相关的依赖关系三元组群是  $\{(found, object, organization), (organization, nn, Black Panters)\}$ , 而在支持答案句“*In October 1966 Bobby Seale founded the Black Panther for self-defense*”中其他部分的依赖关系三元组群是  $\{(found, object, Black Panther) \dots\}$ , 两者无法直接匹配. 我们用启发式规则扩展与疑问非相关的依赖关系三元组来解决这类问题. 这类规则总共有 12 条, 其中两条如下:

**规则 1.**  $R$  是依赖关系三元组群, 如果  $\exists(VP, subject, NP) \in R$ , 则在  $R$  中增加依赖关系三元组  $(VP, by, NP)$ , 其中  $VP$  是动词短语,  $NP$  是名词短语,  $by$  表示被动的依赖关系.

**规则 2.**  $R$  是依赖关系三元组群, 如果  $\exists(NP_1, nn, NP_2) \in R$ , 并且  $\exists(VP, rel, NP_1) \in R$ , 则在  $R$  中增加依赖关系三元组  $(VP, rel, NP_2)$  和  $(VP, rel, NP_2NP_1)$ , 其中  $nn$  表示名词修饰名词的依赖关系.

#### 1.4 依赖关系三元组群的匹配

问题与支持答案句之间的依赖关系匹配由两部分组成: 1) 问题中与疑问相关的依赖关系三元组群与支持答案句中答案相关的依赖关系三元组群的匹配; 2) 问题中与疑问非相关的依赖关系三元组群与支持答案句中答案非相关的依赖关系三元组群的匹配.

依赖关系三元组群的匹配算法如下所示.

##### 依赖关系三元组群的匹配算法

- 1) 在问题中抽取依赖关系三元组群  $R_{ISQ}$ :  
 $\{q_{00}, q_{10}, \dots, q_{10}\}$ ;
- 2) 在  $R_{ISQ}$  上进行关系的扩展和槽的扩展, 得到新的依赖关系三元组群  $\bar{R}_{ISQ}$ :  
 $\{\{q_{00}, q_{01}, \dots\}, \dots, \{q_{m0}, q_{m1}, \dots\}\}$
- 3) 计算每个依赖关系三元组的权重  $w_{jk}$ ;
- 4) 在支持答案句中抽取  $R_{AQ}^i$ :  
 $\{as_0, \dots, as_n\}$ , 并设相似得分  $S = 0$ ;
- 5) 计算  $\bar{R}_{ISQ}$  与  $R_{AQ}^i$  的匹配如下:
- 6) for  $j \leftarrow 0$  to  $m$
- 7)     for  $k \leftarrow 0$  to  $num(q_j)$
- 8)         for  $t \leftarrow 0$  to  $n$
- 9)             if  $Match(q_{jk}, as_t)$
- 10)                 then  $S = S + w_{jk}$ , goto Step 6
- 11) return  $S$

我们可以计算问题中与疑问相关的依赖关系三元组群与支持答案句中答案相关的依赖关系三元组群之间的匹配  $Match(R_{ISQ}, R_{AOA}^i)$ . 同理, 我们可以计算问题中与疑问非相关的依赖关系三元组群和支持答案句中答案非相关的依赖关系三元组群之间的匹配得分  $Match(R_{SQ}, R_{OA}^i)$ . 这样我们可以用式 (5) 和 (6) 计算问题与支持答案句之间的相似度  $Sim(Q, AS_i)$ :

$$Sim(Q, AS_i) = \frac{Sim(Q, AS_i)}{\sum_{j=0}^m w_{j0} + \sum_{k=0}^n w_{k0}} \quad (6)$$

其中  $m$  和  $n$  分别为  $R_{ISQ}$  和  $R_{SQ}$  的大小. 最后, 我

们用式 (4) 计算候选答案的置信度.

## 2 答案排序模型

答案排序模块根据候选答案的排序得分 ( $\gamma$ ) 进行排序, 排序的计算模型定义为  $\gamma = H(f_1^i, \dots, f_5^i)$ , 其中  $f_j^i$  是第  $i$  个候选答案  $A_i$  的特征,  $H$  是排序函数. 本文介绍了 5 种类型的特征: 问题与支持答案句的依赖关系三元组匹配得分 (见第 1 节), 候选答案是否是问题中心词的下位词, 问题成分在支持答案句中匹配得分, 候选答案在支持答案句中与其他成分的距离得分, 以及候选答案在 Web 上出现的频率得分.

候选答案是否是问题中心词  $FW$  的下位词得分参考式 (7):

$$SR(A, FW) = \begin{cases} 1, & FW \neq \emptyset, A \in Hyponymy(FW) \\ 0.5, & FW = \emptyset \\ 0, & FW \neq \emptyset, A \notin Hyponymy(FW) \end{cases} \quad (7)$$

其中  $Hyponymy(FW)$  是中心词在 WordNet 中定义的下位词.

问题成分在支持答案句中的匹配得分  $QTM$  参考式 (8) 和式 (9):

$$QTM(Q, AS) = \max QTM(Q, AS_i) \quad (8)$$

$$QTM(Q, AS_i) = \frac{50 - NC \times 10}{50} \quad (9)$$

其中  $NC$  是问题的成分在支持答案句中未出现的个数.

候选答案在支持答案句中与其他成分的距离得分  $DIS$  与文献 [1] 中定义的距离得分相似. 我们假定在支持答案句中, 问题的成分与候选答案的距离越近, 则该候选答案获得的置信度越高. 与文献 [1] 不同的是, 我们仅考虑问题的五种成分: 主语、谓语、直接宾语、间接宾语和状语. 定义为

$$DIS(A, Q) = \min DIS(A, AS_i, Q) \quad (10)$$

$$DIS(A, AS_i, Q) = \frac{\sqrt{d_1^2 + \dots + d_5^2}}{5} \quad (11)$$

其中  $d$  为问题中每个成分  $C$  在支持答案句中的距离得分, 即

$$d = \begin{cases} 1, & C \in Q, C \notin AS \\ 0.5, & C \notin Q \\ D(A, C, AS), & \text{其他} \end{cases} \quad (12)$$

最后我们考虑候选答案在 Web 中出现的频率得分, 如果候选答案出现的频率越高, 那么该候选答案获得的置信度越高.

本文采用了两种答案排序的方法: 基于优先级的排序和基于机器学习方法的排序.

1) 基于优先级的排序. 根据特征的优先级进行排序, 首先用低优先级的特征排序, 然后用高优先级的特征排序. 特征的优先级从高到低为: 候选答案是否是问题中心词的得分, 问题与支持答案句的依赖关系三元组的匹配得分, 问题成分在支持答案句中匹配的得分, 候选答案在 Web 中出现的频率得分, 以及候选答案在支持答案句中与其他成分的距离得分.

2) 基于机器学习方法的排序. 候选答案可以分为两类: 正确与错误, 所以答案排序问题可以看作是一个分类问题. 本文尝试用机器学习方法进行答案排序, 在第 3 节中, 我们用支持向量机 (Support vector machine, SVM) 的机器学习方法进行答案排序.

## 3 实验分析

国际文本检索会议 (Text retrieval conference, TREC)<sup>1</sup> 是文本检索领域最具有权威性的评测会议. 我们采用了该会议的问题回答子任务中的问题作为实验数据, 大会提供了正确答案的模板. 本文中, 基于四类问题进行实验: What 问题、Which 问题、Who 问题和 Whom 问题.

本文采用 Google 搜索引擎进行 Web 检索. 因为在系统中 Web 检索模块和候选答案抽取模块存在错误累积, 所以本文用相对精度 (*Relative precision*) 来评价答案排序的性能:

$$Relative\ precision = \frac{\text{排在第一的候选答案的精度}}{\text{所有候选答案的精度}} \quad (13)$$

此外, 准确率 (Precision) 和平均排序倒数 (Mean reciprocal rank, MRR) 值也作为评测标准.

本节中, 我们主要讨论两组实验. 在第一组实验中, 我们基于所讨论的 5 类特征, 对基于优先级的排序方法与基于 SVM 的排序方法进行比较, 找出相对比较好的排序方法. 然后我们提出假设, 即问题与支持答案句依赖关系三元组匹配这一特征能显著提高答案排序的性能, 并且通过第二组实验证明这一假设.

在第一组实验中, 我们对基于优先级的排序方法 (Competition based method, CBM) 与基于 SVM 的排序方法进行比较.

<sup>1</sup>TREC 会议是美国国家标准技术局 (National Institute of Standards and Technology) 和国防部高级研究计划局 (Defence Advanced Research Projects Agency) 组织召开的文本检索会议.

我们用 TREC-12 中问题回答子任务的问题以及我们系统抽取的候选答案作为 SVM 方法的训练数据, 用 TREC-13 中问题回答子任务的 119 个问题作为这两种方法的测试数据. 第 2 节介绍的 5 类特征都用在了这组实验中. 在表 2 中, 基于优先级的排序方法比基于 SVM 的排序方法在相对精度上提高了 6%, 在 MRR 值上提高了 3%. 在这组实验中, 基于 SVM 的方法结果并不是很好, 主要原因来自于训练语料的数据稀疏导致了实验的过拟合.

表 2 两种答案排序方法的性能比较

Table 2 The performance comparison between two answer ranking methods

方法	精度	相对精度	MRR
SVM	25.2 %	44.8 %	0.31
CBM	28.6 %	50.8 %	0.34

在第二组实验中, 本文用 TREC 8-13 的问题回答子任务中的 1488 个问题作为实验数据, 其中 What 问题有 1217 个. 并且, 本文采用基于优先级的排序方法.

在表 3 中, 第 1 步, 本文使用 3 类常规特征作为 *baseline*: 问题成分在支持答案句中匹配的得分, 候选答案在支持答案句中与其他成分的距离得分, 候选答案是否是问题中心词的得分. 第 2 步, 我们加入候选答案在 Web 中出现的频率得分这一特征. 第 3 步, 为了与现有的基于依赖关系的语义相似度方法进行比较, 本文加入不挖掘问题疑问部分信息的依赖关系三元组的匹配得分这一特征. 第 4 步, 本文加入问题与支持答案句的依赖关系三元组的匹配得分这一特征. 分析表 3 的结果可以看到, 第 2 步比第 1 步在相对精度上提高了 3.7%, 在 MRR 值上提高了 3%. 第 3 步比第 1 步在相对精度上提高了 5.6%, 在 MRR 值上提高了 5%. 第 4 步比第 1 步在相对精度上提高了 8.2%, 在 MRR 值上提高了 8%; 比第 3 步在相对精度上提高了 2.6%, 在 MRR 值上提高了 3%. 由此可见, 加入问题与支持答案句依赖关系三元组匹配能明显提高答案排序的性能, 并且挖掘了问题疑问部分所含有信息的依赖关系三元组的匹配优于传统的依赖关系三元组匹配的方法, 由此证明了本文的假设.

在表 4 中, 本文对在回答 What 问题时, 基于不同的特征空间的性能进行了比较. 可见, 新的特征的加入明显提高了回答 What 问题的性能.

挖掘了问题疑问部分所含有信息的依赖关系三

元组匹配的方法有效地反映了支持答案句与问题在语义层次上的相似, 从而反映了候选答案是否正确地回答了问题. 在错误分析中, 本文发现错误主要来自于三方面: 1) 在 Web 上一些错误的句子与问题很相似, 这样抽取出来的候选答案虽然是错的, 但是却排在了前面; 2) Minipar 错误的输出导致抽取错误的依赖关系三元组; 3) 人工编写的启发式规则不能解决一些复杂的变形的支持答案句与问题的匹配.

表 3 基于不同的特征空间的性能的比较

Table 3 The performance comparison with different feature spaces

特征	精度	相对精度	MRR
$f_1, f_2, f_3$	24.4 %	39.3 %	0.31
$f_1, f_2, f_3, f_4$	26.7 %	43.0 %	0.34
$f_1, f_2, f_3, f_4, f_5$	27.9 %	44.9 %	0.36
$f_1, f_2, f_3, f_4, f_5, f_6$	29.5 %	47.5 %	0.39

说明:  $f_1$ : 问题成分在支持答案句中的出现得分;

$f_2$ : 候选答案在支持答案句中与其他成分的距离得分;

$f_3$ : 候选答案是否是问题中心词的下位词得分;

$f_4$ : 候选答案在 Web 中出现的频率得分;

$f_5$ : 现有的依赖关系三元组的匹配得分 (不挖掘问题疑问部分的信息);

$f_6$ : 问题与支持答案句的依赖关系三元组的匹配得分.

表 4 基于 What 问题的性能的比较

Table 4 The performance comparison based on "What" question

特征	精度	相对精度	MRR
$f_1, f_2, f_3$	24.5 %	39.4 %	0.31
$f_1, f_2, f_3, f_4$	27.0 %	43.4 %	0.35
$f_1, f_2, f_3, f_4, f_5$	28.1 %	45.2 %	0.37
$f_1, f_2, f_3, f_4, f_5, f_6$	29.8 %	47.9 %	0.40

说明:  $f_1 \sim f_6$  含义同表 3.

## 4 结论与展望

本文介绍了一种新的计算问题与支持答案句的相似度的方法, 即基于依赖关系三元组的匹配的方法, 并把该相似度应用在问题回答的答案排序中. 实验证明, 新的答案排序方法优于传统的基于密度排序的方法, 并且新的特征的加入明显提高了排序的性能. 在将来的工作中将会尝试用机器学习的方法学习更多的依赖关系三元组扩展的规则, 从而能够

处理复杂的支持答案句与问题的匹配.

## References

- 1 Kwok C, Etzioni O, Weld D S. Scaling question answering to the web. *ACM Transactions on Information Systems*, 2001, **19**(3): 242–262
- 2 De Boni M, Manandhar S. The use of sentence similarity as a semantic relevance metric for question answering. In: *Proceedings of the AAAI Symposium on New Directions in Question Answering*. Stanford, USA: AAAI, 2003. 138–144
- 3 Miller G A, Beckwith R, Fellbaum C, Gross D, Miller K J. Introduction to wordnet: an on-line lexical database. *International Journal of Lexicography*, 1990, **3**(4): 235–244
- 4 Attardi G, Cisternino A, Formica F, Simi M, Tommasi A, Zavattari C. PIQASso: pisa question answering system. In: *Proceedings of the 10th Text Retrieval Conference*. Gaithersburg, USA: NIST, 2001. 599–607
- 5 Katz B, Lin J. Selectively using relations to improve precision in question answering. In: *Proceedings of the EACL 2003 Workshop on Natural Language Processing for Question Answering*. Budapest, Hungary: EACL, 2003. 1–8
- 6 Sun R X, Cui H, Li K Y, Kan M Y, Chua T S. Dependency relation matching for answer selection. In: *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Salvador, Brazil: ACM, 2005. 651–652
- 7 Punyakanok V, Roth D, Yih W. Mapping dependencies trees: an application to question answering. In: *Proceedings of the 8th International Symposium on Artificial Intelligence and Mathematics*. Fort Lauderdale, USA: AI and MATH, 2004. 1–10
- 8 Moldovan D, Pasca M, Harabagiu S, Surdeanu M. Performance issues and error analysis in an open-domain question answering system. *ACM Transactions on Information Systems*. 2003, **21**(2): 133–154
- 9 Lin D K. Dependency-based evaluation of minipar. In: *Proceedings of Workshop on the Evaluation of Parsing Systems*. Granada, Spain: McNamara, 1998. 298–312



**林 锋** 复旦大学计算机科学与工程系博士研究生. 2004 年获得上海大学计算机工程与科学学院硕士学位. 主要研究方向为自然语言处理, 信息检索. 本文通信作者.

E-mail: fenglin@fudan.edu.cn

(**LIN Feng** Ph.D. candidate at Fudan University. He received his master degree from School of Computer Engineering and Science, Shanghai University in 2004. His research interest covers natural language processing and information retrieval. Corresponding author of this paper.)



**周雅倩** 复旦大学计算机科学与工程系讲师. 2005 年获得复旦大学计算机科学与工程系博士学位. 主要研究方向为自然语言处理.

E-mail: zhouyaqian@fudan.edu.cn

(**ZHOU Ya-Qian** Lecturer at Fudan University. She received her Ph.D. degree from Computer Science and Engineering Department, Fudan University in 2005. Her main research interest is natural language processing.)



**黄萱菁** 复旦大学计算机科学与工程系教授. 1998 年获得复旦大学计算机科学与工程系博士学位. 主要研究方向为自然语言处理.

E-mail: xjhuang@fudan.edu.cn

(**HUANG Xuan-Jing** Professor at Fudan University. She received her Ph.D. degree from Computer Science and Engineering Department, Fudan University in 1998. Her main research interest is natural language processing.)



**吴立德** 复旦大学计算机科学与工程系教授. 1958 年毕业于复旦大学数学系数学专业. 主要研究方向为自然语言处理, 图像识别.

E-mail: ldwu@fudan.edu.cn

(**WU Li-De** Professor at Fudan University. He graduated from the Mathematics Department, Fudan University in 1958. His research interest covers natural language processing and image processing.)