【读后感】使用了较多的统计特征和规则特征，很多值得借鉴的地方。
【题目】基于多特征的中文文本蕴含识别方法
【作者】李妍
【单位】武汉科技大学
【期刊】硕士学位论文
【时间】2013 年 4 月

【笔记】
【摘要】
模型：支持向量机模型
特征：
统计特征；
词汇语义特征（知网、同义词林、反义词表、否定词表）；
句法特征；
事件语义特征；

中文分词：斯坦福分词工具可以通过加载不同的分词标准（中文滨州树库标准 CTB、北京大学标准 PKU），生成不同的分词结果。——基于 PKU 的斯坦福分词器的分词效果最好。
去停用词：哈工大停用词词表；

统计特征：

词集特征（针对文本对经过数据预处理的词集提取的特征）：

词重叠度；

词重叠度是指一个给定文本对（T, H）中，文本 T 和文本 H 中相同的词汇个数与文本 T 和文本 H 中词汇集合的比值。通常认为，如果 T 和 H 中相同的词越多，则 T 和 H 表达的意义越相近，具体计算方法见公式（2.1）。

$$W_{overlap}=\frac{|Words(T)\cap Words(H)|}{|Words(T)\cup Words(H)|} \quad \text{公式（2.1）}$$

公式（2.1）中，Words（T）指代文本 T 的词汇集合。

文本长度差；

Jaro-Winkler 距离；

Jaro-Winkler 距离是针对两字符串相似度的一个度量，Jaro-Winkler 值越大，表明两字符串的相似度越高。Jaro-Winkler 尤其适合短字符串相似度的度量，如专有名词中的人名、地名等。具体计算方法见公式（2.3）和公式（2.4）。

$$JW_{dis}(T,H)=\frac{m}{3*Len(T)}+\frac{m}{3*Len(H)}+\frac{m}{3*m} \quad \text{公式（2.3）}$$

$$L_{jw}(T,H)=\frac{\max(Len(T),Len(H))}{2}-1 \quad \text{公式（2.4）}$$

公式（2.3）中 m 是文本 T 和 H 匹配字符串的个数，这里“匹配”的含义是同一个字符串在指定的 L_{jw} 长度范围内同时出现在文本 T 和 H 中。

最长公共子串（LCS）相似度；

$$Sim_{LCS} = \frac{Len(LCS(T,H))}{\min(Len(T),Len(H))}$$ 公式（2.5）

公式（2.5）中，LCS（T, H）用来计算文本 T 和 H 的最长公共子串。

向量特征（为了凸显重要的词汇，用 TF*IDF 为文本中的每个词赋一个权值，将文本向量化）；

向量余弦相似度；

$$Sim_{\cos}(\vec{t},\vec{h}) = \frac{\sum_{i=1}^{n} t_i * h_i}{\sqrt{\sum_{i=1}^{n} t_i} * \sqrt{\sum_{i=1}^{n} h_i}}$$ 公式（2.6）

公式（2.6）中，$\vec{t}$ 和 $\vec{h}$ 是相对于文本 T 和 H 的向量，n 是向量维度，每个向量使用传统的 TF*IDF 方法计算得到。

每个词 TF*IDF 的结果是一个数值，$\vec{t}$ 和 $\vec{h}$ 中的元素是一个个的数值。

欧式距离；

欧式距离考虑两点间的距离，可以用欧式距离计算文本 T 与假设文本 H 对应向量之间的相似度，此相似度可以作为文本 T 与假设文本 H 之间的相似度。具体计算方法见公式（2.8）。

$$E_{dis}(\vec{t},\vec{h}) = \sqrt{\sum_{i=1}^{n}(t_i - h_i)^2}$$ 公式(2.8)

曼哈顿距离；

曼哈顿距离是在欧几里得空间的固定直角坐标系上两点所形成的线段对轴产生的投影的距离总和。例如，在平面上，坐标（x_1, y_1）的点 P1 与坐标（x_2, y_2）的点 P2 的曼哈顿距离为$|x_1-x_2|+|y_1-y_2|$。曼哈顿距离可以被用来计算文本对之间的相似度。具体计算方法见公式（2.7）。

$$M_{dis}(\vec{t},\vec{h}) = \sum_{i=1}^{n} |t_i - h_i|$$ 公式（2.7）

语义特征：

基于知网语义相似度

文本对间词汇的语义相似度越高，它们之间存在蕴涵关系的可能性越大。基于知网语义相似度特征可以使用公式（2.9）来计算得到。

$$LS_{Hownet} = \frac{1}{2}\left(\frac{1}{m}\sum^{m}\max\{sim_w(w_{1i}, w_{2j}) | 1 \le j \le n\} + \frac{1}{n}\sum^{n}\max\{sim_w(w_{1i}, w_{2j}) | 1 \le i \le m\}\right)$$ 公式（2.9）

公式（2.9）中，$\{w_{1i}|1\le i\le m\}$代表文本对（T, H）中文本 T 的词汇集，$\{w_{2j}|1\le j\le n\}$代表文本 H 的词汇集。Sim_w（w_{1i}, w_{2j}）代表 w_{1i} 与 w_{2j} 的词汇相似度，这里的词汇相似度计算是基于知网[35]进行的。

[35]刘群，李素建。基于《知网》的词汇语义相似度计算。计算语言学及中文信息处理。

基于同义词林语义相似度

文本 T 和 H 中的同义词可以提高识别双向蕴涵关系的识别率，基于同义词林的语义相似度特征可以使用公式（2.10）来计算得到。

$$LS_{Cilin}=\frac{1}{2}\left(\frac{1}{m}\sum_{i=1}^{m}\max\{sim_w(w_{1i},w_{2j})|1\le j\le n\}+\frac{1}{n}\sum_{j=1}^{n}\max\{sim_w(w_{1i},w_{2j})|1\le i\le m\}\right) \quad \text{公式（2.10）}$$

公式（2.10）中词义相似度 Sim_w（w_{1i}, w_{2j}）的计算采用文章[36]中提到的方法。

[36]田久乐，赵蔚。基于同义词词林的词语相似度计算方法。吉林大学学报（信息科学版）

反义词特征

文本 T 和 H 中反义词对在一定程度上可以反映出两者的矛盾关系，利用反义词特征可以提高矛盾的识别率，系统利用公式（2.11）来计算两个文本间的反义词特征。

$$f_{antonym}=\begin{cases}1 & (n\neq 0)\\ 0 & (n=0)\end{cases} \quad \text{公式（2.11）}$$

公式（2.11）中，n 是计算（Words（T）-（Words（T）∩Words（H））词汇集与（Words（H）-（Words（T）∩Words（H））词汇集之间的反义词对数，这样比直接计算文本 T 的词汇集与文本 H 的词汇集之间的反义词对数效果更好，这种方法可以避免文本因为分词错误导致的反义词识别错误，具体的例子如表 2.2 所示。

否定词特征

除反义词对外，文本 T 和 H 中出现的否定词也可以在一定程度上反映出两者间的矛盾关系，系统使用公式（2.12）来计算两个文本间的否定词特征。

$$f_{neg}=\begin{cases}0 & (n_1\%2=n_2\%2)\\ 1 & otherwise\end{cases} \quad \text{公式（2.12）}$$

公式（2.12）中，n_1 和 n_2 分别指文本 T 和 H 中否定词的数目。

词义重叠比

在下面的例 2.2 中，文本 T11 和 H11 相同的词汇很多，但由于 T11 的文本比 H11 的文本长很多，于是词重叠度特征计算出来并不是特别高，这种文本对往往会被错误的判断为独立关系，但实际上 T11 和 H11 为正向蕴涵关系。

例 2.2：

T11：据他所知，这是查尔斯首次参加悉尼-霍巴特帆船赛，而查尔斯一向是注重安全、非常谨慎的人，他更想参加 2000 年悉尼奥运帆船赛。

H11：2000 年奥运在悉尼举办。

为了解决这一问题，提出了词义重叠度这一词汇语义特征，如公式（2.13）所示。

$$CWR(T,H)=\frac{|SW(Words(T),Words(H))_{similarity=1}|}{\min(Len(T),Len(H))} \quad \text{公式（2.13）}$$

公式（2.13）中，SW（Words（T）,Words（H））$_{similarty=1}$ 为出现在文本 T 和 H 中相同词和同义词集合，同义词基于同义词林来认定。

句法特征：

例 2.3：

T12：南亚海啸已在南亚及东南亚八国和三个非洲国家索马里、肯尼亚、坦桑尼亚造成至少五万五千人丧生。

H12：南亚海啸造成至少五万五千人丧生。

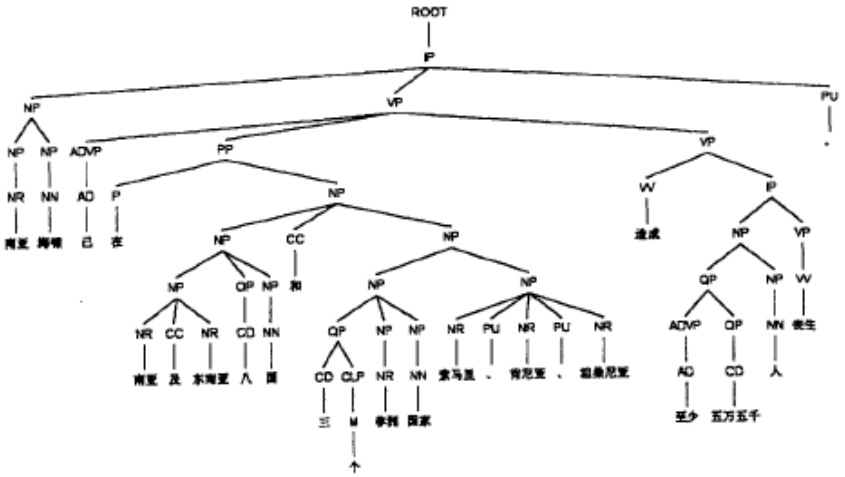

图2.2　T12的句法结构树

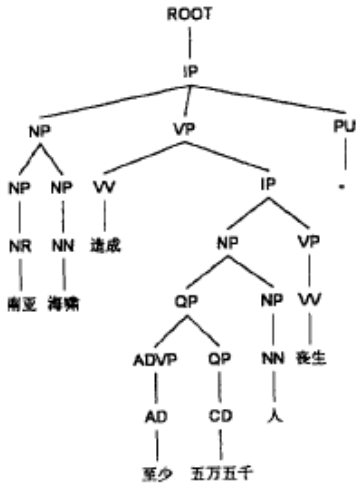

图2.3　H12的句法结构树

在图 2.2 和图 2.3 中，H12 句法结构树与 T12 句法结构树的一部分非常相似。实际上，T12 和 H12 为正向蕴涵关系。

句法依存树可以转化为一个三元组的依存关系，记为 rel（w_1-loc_1,w_2-loc_2），其中，rel 指词 w_1 和词 w_2 的依存关系，loc_1 和 loc_2 分别指词 w_1 和词 w_2 在句子中的位置，例 2.4 中列举了例 2.3 文本的依存关系集合。

例 2.4：

T12：root（ROOT-0, 南亚-1），dep（南亚-1, 海啸-2），advmod（造成-20, 已-3），prep（造成-20, 在-4），dep（在-4, 南亚-5），dep（在-4, 东南亚-7），conj_及（南亚-5, 东南亚-7），num（南亚-5, 八-8），dep（南亚-5, 国-9），num（非洲-13, 三-11），dep（三-11, 个-12），dep（在-4, 非洲-13），conj_和（南亚-5, 非洲-13），dep（非洲-13, 国家-14），dep（坦桑尼亚-19, 索马里-15），dep（坦桑尼亚-19, 、-16），dep（坦桑尼亚-19, 肯尼亚-17），dep（坦桑尼亚-19, 、-18），dep（非洲-13, 坦桑尼亚-19），dep（五万五千-22, 至少-21），num（人-23, 五万五千-22）

H12：root（ROOT-0, 南亚-1），dep（南亚-1, 海啸-2），dep（五万五千-5, 至少-4），num（人-6, 五万五千-5）

利用文本的依存关系，可以计算两个文本之间的句法相似度，具体计算方法见公式（2.14）。

$$Sim_{syntree}=\frac{\sum_{p_H\in S_H}\max_{p_T\in S_T}\{sim_p(p_T,p_H)\}}{|S_H|} \qquad \text{公式(2.14)}$$

公式（2.14）中，S_T 与 S_H 是文本对（T, H）中文本 T 和文本 H 的句法依存关系集合，p_T 和 p_H 是依存关系集合 S_T 与 S_H 中的一个依存关系。Sim_p（p_T, p_H）代表 p_T 与 p_H 之间的相似度，具体计算方法见公式（2.15）。

$$sim_p(p_T,p_H)=\frac{1}{2}(\max\{sim_w(w_1,w_1')+sim_w(w_2,w_2'),sim_w(w_1,w_2')+sim_w(w_2,w_1')\}) \qquad \text{公式(2.15)}$$

公式（2.15）中，$w_1,w_2\in p_T$ 且 $w_1',w_2'\in p_H$，sim_w（w_1,w_2）代表词 w_1 与词 w_2 之间的词语相似度，具体计算方法见公式（2.16）。

$$sim_w(w_1,w_2)=\begin{cases}1 & w_1=w_2\\ 0 & otherwise\end{cases} \qquad \text{公式(2.16)}$$

只考虑了依存关系中成分的相似度。

SVM 分类模型：

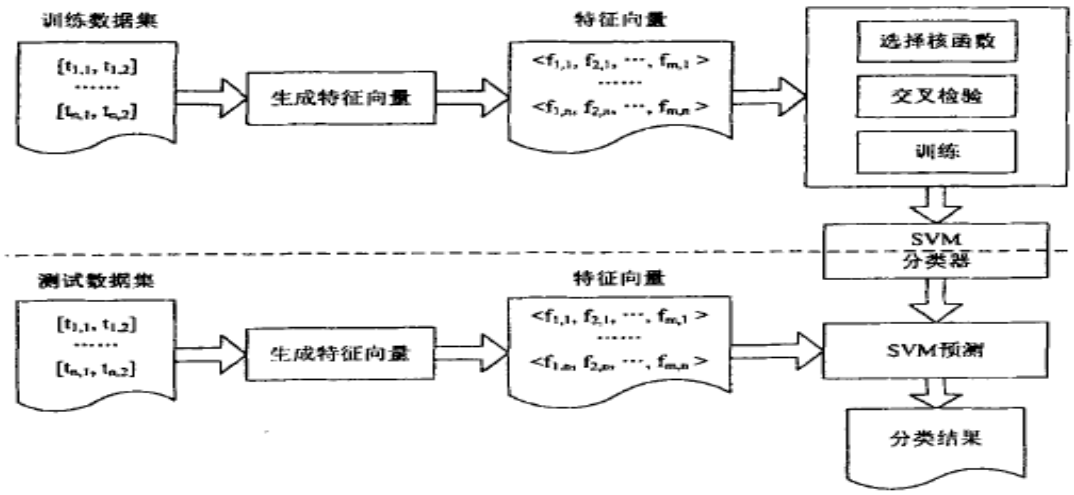

图2.4　　SVM分类

特征向量：数据预处理和特征提取的结果。

分类性能影响因素：

误差惩罚参数 C；

核函数形式及其参数的选择；

RBF（Radial Basis Function）核函数：

$$K(x,x_i)=\exp(-\gamma^*\|x-x_i\|^2)$$　　公式（2.17）

上式中，γ 为核函数的宽度。

参数选择（交叉验证）：采用网格搜索法（Grid Search）确定惩罚因此 C 与核函数参数 γ。

多分类器（一对一/一对多/有向无环图 SVMs）：采用一对一方法进行多分类器的构造，为每两类构造一个二分类器，即 K 类就有 k(k-1)/2 个二分类器，本文构造十个二分类器。

基于事件语义特征的系统结构图：

在之前系统基础上增加了基于事件标注语料的事件语义特征与基于事件语义规则的修正模块。

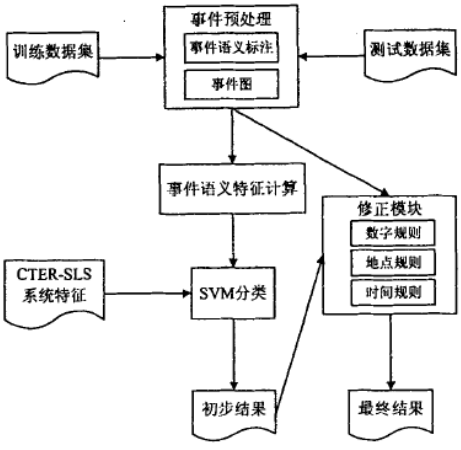

图3.1　　CTER-ES系统结构图

事件预处理模块对测试数据集和训练数据集进行事件语义标注，并根据标注文本生成事件图；

事件语义特征计算模块的作用是基于生成的事件图，将文本的蕴涵关系转化为图的蕴涵关系；

将 CTER-SLS 系统特征和事件语义特征相结合，利用 SVM 分类模块，生成初步结果；

将初步结果基础上基于事件语义规则进行修正，生成最终结果。

事件语义角色结构图：

原子事件划分完毕之后，接着对原子事件进行标注，每个原子事件只有一个事件谓词，标记为“EP”。原子事件语义角色一般分为核心和附加语义角色两大类。其中核心语义角色又分为主体、客体和时空语义角色三类。目前主体语义角色包括施事、致事和主事，标记分别为“A”、“Cau”和“Th”，其中主事是主体语义角色的回收站，所有不能标为施事、致事的主体语义成分都被标为主事；而客体语义角色包括受事、与事、结果和系事，标记分别为“P”、“D”、“R”和“Re”；时空语义角色包括时间和地点，标记分别为“T”和“L”。附加语义角色则更多，包括工具、材料、方式、原因、目的以及范围，标记分别为“I”、“Ma”、“M”、“Rn”、“Ai”和“Ra”等，事件语义角色结构图如图 3.2 所示。

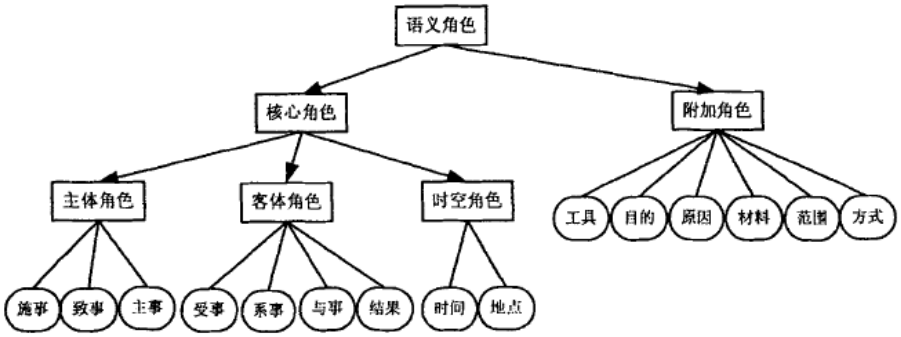

图3.2　事件语义角色结构图

事件图：

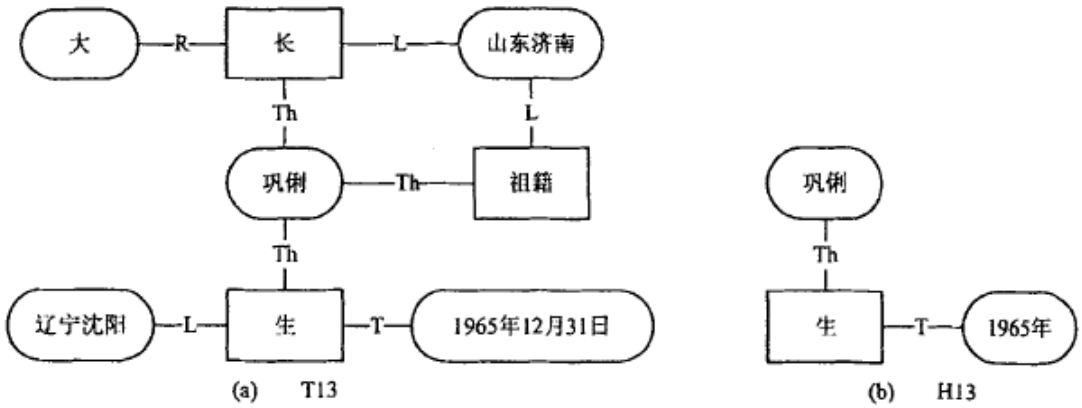

图3.3　T13与H13事件图

事件语义特征：

利用生成的事件图计算基于最大公共子图的图相似度。具体分为三步：构造图结构、

求取最大公共子图、图相似度计算。

图结构：

这里的图结构主要表示为一个三元组，即 G=（Node, Edge, Weight），其具体计算步骤如下所示：

（1）从事件标注文本对（T, H）生成的事件图中提取所有的节点组成节点集合 $Node_T$ 和 $Node_H$，集合 $Node_T$ 和 $Node_H$ 是由节点的内容词 w 组成，其公式见（3.1）。

$$Node = \{w_1, w_2, ..., w_n\} \quad \text{公式（3.1）}$$

（2） 从事件标注文本对（T, H）生成的事件图中提取所有的边组成边集合 $Edge_T$ 和 $Edge_H$，集合 $Edge_T$ 和 $Edge_H$ 是一个三元组集合，由两个节点 w_i 和 w_j，边的语义角色标签 $label_{ij}$ 组成，其公式见（3.2）。

$$Edge = \{E_{12}, E_{13}, ..., E_{ij}, ...\}, E_{ij} = < w_i, w_j, label_{ij} > \quad \text{公式（3.2）}$$

（3） 利用得到的节点集合和边集合，得到边的权重集合 $Weight_T$ 和 $Weight_H$，集合 $Weight_T$ 和 $Weight_H$ 是一个二元组集合，由边 E_{ij} 和边的权重 $wt(E_{ij})$ 组成，其公式见（3.3），其中权重 $wt(E_{ij})$ 的计算公式见（3.4）。

$$Weight = \{W_{12}, W_{13}, ..., W_{ij}, ...\}, W_{ij} = < E_{ij}, wt(E_{ij}) > \quad \text{公式(3.3)}$$

$$wt(E_{ij}) = freq(w_i, w_j) / (freq(w_i) + freq(w_j) - freq(w_i, w_j)) \quad \text{公式（3.4）}$$

公式（3.4）中，freq（w_i）为节点 w_i 出现在文本中的频率，freq（w_i, w_j）为节点 w_i 和节点 w_j 同现在文本中的频率。

（4） 将事件图 G 转化为三元组（Node, Edge, Weight），其公式见（3.5）。

$$G = \{g_1, g_2, ..., g_k, ...\}, g_k = < w_i, w_j, E_{ij}, W_{ij} > \quad \text{公式(3.5)}$$

最大公共子图：

根据图结构 G_T 和 G_H 求解最大公共子图 G_C 的步骤如下所示：

（1） 分别遍历图结构 G_T 和 G_H，若 G_T 和 G_H 之间存在相同的节点集合，则将相同的节点集合作为最大公共子图的节点集合 $Node_C$；判断节点相同，只需判断 $w_i = w_j$，其中 $w_i \in Node_T$，$w_j \in Node_H$。

（2）如果集合 $Node_C$ 中任两个节点之间存在一条边 E_{ij}，且 E_{ij} 等于某一条既属于 $Edge_T$ 又属于 $Edge_H$ 的边，则将 E_{ij} 加入 G_C 边集合 $Edge_C$ 中。

（3） 根据得到的 $Node_C$ 和 $Edge_C$ 计算 $Weight_C$，其中 $Weight_C$ 中的二元组集合 W_{ij} 的计算公式见（3.6）。

$$W_{ij} = \frac{\min(W_T(E_{ij}), W_H(E_{ij}))}{\max(W_T(E_{ij}), W_H(E_{ij}))}, E_{ij} \in Edge_C \quad \text{公式（3.6）}$$

公式（3.6）中，$W_T(E_{ij})$为边 E_{ij} 在图 G_T 中存在一条相等的边 E_{xy} 的权重 $wt(E_{xy})$；同理，$W_H(E_{ij})$为边 E_{ij} 在图 G_H 中存在一条相等的边 E_{ab} 的权重 $wt(E_{ab})$。

以上求解最大公共子图的伪代码如下所示：

算法 3.1 求解最大公共子图

输入：

G_T=（$Node_T$, $Edge_T$, $Weight_T$）

G_H=（$Node_H$, $Edge_H$, $Weight_H$）

输出：

G_C=（$Node_C$, $Edge_C$, $Weight_C$）

//$w_i \in Node_T$ （$Node_T$ 含有 n 个节点）

//$w_j \in Node_H$ （$Node_H$ 含有 m 个节点）

```
1: for（i = 0; i < n; i++） {
2:       for（j = 0; j < m; j++） {
3:            if（wi==wj）
4:                 NodeC ={wi} ∪ NodeC
5:       }
6: }
7: if（wi, wj∈NodeC && Eij∈（EdgeT∩EdgeH））
8:      EdgeC={Eij} ∪ EdgeC
9:      WeightC={Wij} ∪ WeightC
10: else
11:      wi, wj 没有边
```

图相似度：

根据求得的最大公共子图，计算图相似度的公式如（3.7）所示。

$$sim(G_T, G_H) = \beta \frac{sizeof(Node_C)}{\max(sizeof(Node_T), sizeof(Node_H))} + (1-\beta) \frac{\sum W_C(E_{ij})}{\max(sizeof(Edge_T), sizeof(Edge_H))} \quad \text{公式（3.7）}$$

其中，sizeof（$Node_T$）为 G_T 中节点的个数，sizeof（$Edge_T$）为 G_T 中边的条数，$W_C(E_{ij})$ 为图 G_C 中边 E_{ij} 的权重 $wt(E_{ij})$；β 为综合加权因子，当 β=0.5 时，图 G_T 和图 G_H 中节点和边对图相似度的影响程度相同，当 β=0 时，不考虑节点对图相似度的影响，当 β=1 时，不考虑边对图相似度的影响。本文中，β 等于 0.6，综合考虑了边和节点对图相似度的影响。

基于事件语义规则的修正：

1、基于数字的语义规则：数值不同、单位不同、数字范围不同；

2、基于地点的语义规则：地点不同；

3、基于时间的语义规则：时间不同；