一种基于 E-Chunk 的机器翻译模型*

李 沐, 吕学强, 姚天顺

(东北大学 计算机科学与工程研究所,辽宁 沈阳 110004)
E-mail: ics@mail.neu.edu.cn
http://www.nlplab.com

摘要: 提出了一种基于 E-Chunk 的多引擎机器翻译模型.该模型以中心语驱动的分析技术为基础,通过词汇相似特征计算 E-Chunk 的匹配代价,自底向上地完成最优 E-Chunk 覆盖的构造,并以 E-Chunk 为基本翻译单元完成机器翻译过程.初步的实验结果显示,该方法在面向领域文本的自动翻译方面是有效的.

关 键 词: E-Chunk;机器翻译;词汇相似计算
中图法分类号: TP18 文献标识码: A

随着信息技术的迅速发展和对真实文本处理需求的急剧增长,基于规则系统的机器翻译方法正面临着日益严重的困境和挑战[1],而以基于实例的机器翻译[2](example based machine translation,简称 EBMT)技术为代表的经验主义方法则成为近期机器翻译研究中的一个重要趋势.现在已经有了很多 EBMT 方法的变体和扩展,其中较重要的方面包括加标实例的使用[3]、实例的模板化[4]和将类比的基本单元由句子细化为片断[3].

与经验主义理论的实证哲学观不同,Chunk 是一种基于心理语言学模型的计算语言学文本结构单元[5].来自心理语言学和韵律学的研究数据显示,这种文本结构单元具有结构上的稳定性与功能上的无歧义性.Abney 的实验表明,基于 Chunk 的有限状态级联分析技术在处理非受限真实文本方面,其精确性和健壮性都是十分优秀的[6].

本文目的在于将作为 Chunk 理论基础的心理语言学假设引入到机器翻译的研究领域,即如果我们把人类的翻译过程相应地解释为一个双语性能词群的选取和重新组合过程,那么,机器翻译中的基本处理单元也应该是一组高度相关的、具有相对稳定的句法结构和内部语义自解释能力的词汇的集合.我们将这种机器翻译的基本处理单元称为 E-Chunk(extended chunk).基于 E-Chunk 机器翻译模型的基本框架就是以 E-Chunk 作为 EBMT 中双语实例知识的基本表示单元,在由源语句法和双语 E-Chunk 知识库共同定义的搜索空间中查找输入句子的最佳匹配 E-Chunk 集合,并通过结果集中的双语 E-Chunk 实例完成对翻译转换和目标语生成过程的驱动.

1 翻译模型

1.1 E-Chunk的基本概念

机器翻译模型中的双语 E-Chunk 在单语 E-Chunk 的基础上定义.每个单语 E-Chunk 可以形式化地表示为一个四元组:$EC=<T,h,l,r>$,其中 T 为 EC 的句法标记,h 为中心词,l 和 r 分别为由 h 左、右两侧依存子结点构成的

* 收稿日期: 2000-08-21; 修改日期: 2000-12-19
基金项目: 国家自然科学基金资助项目(69985001);国家重点基础研究 973 资助项目(G19980305011);国家教育部博士点基金资助项目(1999014503)

作者简介: 李沐(1972－),男,辽宁辽阳人,博士,副研究员,主要研究领域为自然语言处理;吕学强(1970－),男,辽宁抚顺人,博士生,讲师,主要研究领域为机器翻译;姚天顺(1934－),男,江苏苏州人,教授,博士生导师,主要研究领域为自然语言处理,信息检索,机器翻译.

有序线性列表.双语 E-Chunk 知识库依据词汇主义[7]的原则构造,称为双语 E-Chunk 词典.词典中的每个入口定义为以源语 E-Chunk 中心词汇为索引的三元组<H,G,f>,其中 H 为源语的单语 E-Chunk,G 是以 H 中非词汇参数为变元的目标语模板,f 为 H 中词汇和 G 中对应词汇之间的对译映射函数.

为表述方便,在下面的讨论中,将源语的单语 E-Chunk 简称为 E-Chunk.E-Chunk 是 Chunk 概念的一种扩展,这种扩展的主要动机在于其功能角色由单语分析单元向双语翻译单元的转换,而 Chunk 中严格的句法本位描述原则无法满足这种要求.一个合格的 E-Chunk 应满足如下条件:

(1) E-Chunk 具有语义自足性.输入与输出之间的意义等价性是机器翻译的基本原则,消除词汇歧义是机器翻译研究中一个基本环节.E-Chunk 作为翻译转换中的一个基本单元,本身应该包含其内部词汇元素及自身结构的消歧语境,即是消歧上下文自包含的.E-Chunk 的语义自足性来自于语义关联的局部性假设,同时也得到基于语料统计的词义消歧方法研究结果的支持.这些方法的实验结果表明[8],用于消解词汇歧义的上下文绝大部分来自于词汇自身的局部线性语境或局部结构语境.考虑名词短语 a demo file 所对应的单语 E-Chunk 表示为[NP,file,a,demo],其中 demo 和 file 的共现决定了各自的语义特征,这里,file 的含义为“文件”而不是“锉”;

(2) E-Chunk 具有结构平面性.为了满足上述语义约束条件,E-Chunk 需要包含比 Chunk 更复杂的句法结构,即 E-Chunk 在结构上可以对应于一棵句法树的部分推导.然而,为了避免树文法处理中的复杂性[9],我们将 E-Chunk 结构限制为如下两种类型的单层次依存结构:ES_L 和 ES_G.以 w 为中心词的 E-Chunk 集合(ES 含义为 E-Chunk structure) 表示为上述两种类型 E-Chunk 集合的并集 $ES(w)=ES_L(w)\bigcup ES_G(w)$ .ES_L 类型的 E-Chunk 中仅包含中心词的直接依存结点,此时,双语 E-Chunk 基本等价于 Shake 和 Bake 翻译方法[10]中的一个双语转换规则的实例表示.如果依存参数中可以包含非词汇化的句法标记,则其进一步一般化为一个多词入口的转换模板,其中的元素在匹配时没有连续性限制.ES_G 类型 E-Chunk 则取消树结构的中间表示层次,使其简化成为一个标注了中心词汇信息的表层字符串.该类型 E-Chunk 中不允许非词汇化类型的参数,其匹配对象也相应地定义为语言中的表层线性串.比如,名词短语 a file for demo 所对应的 E-Chunk 就需要表示为[NP,file,a,for]或者[NP,file,a,(for,demo)]的形式,而不是[NP,file,a,(for (demo))];

(3) 双语 E-Chunk 具有转换充分性.每个双语 E-Chunk 本身就定义为一个翻译模板,其中包含了翻译该 E-Chunk 所需的全部词汇和结构转换信息,如多词映像(many-to-many word mapping)、参数交换(argument switching)和中心词交换(head switching)[11].此外,转换充分性也涵盖了 Chunk 分析中的歧义包容原则:对于存在内部关联歧义的结构,则将其作为一个整体包含在一个 E-Chunk 中.

1.2 基于E-Chunk的机器翻译模型

基于 E-Chunk 的机器翻译,简单地说,就是给定输入句子 S,在双语 E-Chunk 词典中选取一组可以与之进行最优匹配的双语 E-Chunk ES*,应用 ES*中提供的双语转换信息实现语言的自动翻译.这种翻译模型我们称为基于 E-Chunk 的机器翻译模型(E-Chunk based machine translation,简称 ECBMT,).

定义 **1**. 片断.设 T 为输入句子 S 某个可能的依存分析树,则 T 的一个连通子图 H:<h,T,l,r>称为 T 的一个片断.其中 h,T,l,r 的含义与 E-Chunk 定义中的相同.我们使用 $E(H)$ 表示 H 中的边集,$V(H)=h\cup l\cup r$,表示结点集.

这里,我们将片断与子树的概念加以区别.子树中所有的叶子结点都是整个分析树的叶子结点,而片断则无此要求.子树是被看做是一种特殊类型的片断.

定义 **2**. 划分.设子树片断 T 中片断集合 $HS=\{H_1,\ldots,H_n\}$,若对于任意 $1\le i\le n,1\le j\le n,i\ne j$, 都有 $E(H_i)\cap E(H_j)=\varnothing,\bigcup_{1\le i\le n}H_i=T$,则称 HS 为树 T 的一个划分,记作 HS_T .

定义 **3**. 匹配.给定 E-Chunk EC:<T,h,l,r> 和子树片断 T 中的某个片断 H:<h',T',l',r'>, 若有 $h=h',T=T',l=l',r=r'$,则称 H 是 EC 的一个完全匹配.若仅有 $T=T'$,则称 H 为 EC 的一个部分匹配.完全匹配和部分匹配统称为匹配,记为 $\psi(H)=EC$.对于 E-Chunk EC 的每个匹配 H,都有一个衡量其间相似程度的匹配代价 $\theta_\psi(EC\mid H)$ 与之相关联.匹配代价越低,两者之间的相似程度越高.该代价定义为 E-Chunk 与匹配对象的函数,具体的计算方法我们将在下节加以讨论.

定义 **4**. 覆盖.设 T 为一个子树片断, $HS_T=\{H_1,\ldots,H_n\}$ 是 T 的一个划分,若存在 E-Chunk 集合

$ES=\{EC_1,\ldots,EC_n\}$ 满足 $\psi(H_i)=EC_i(1\le i\le n)$,则称 ES 为 T 的一个覆盖,记为 $ES_T=\Psi(T)=\bigcup_{1\le i\le n}\psi(H_i)$.覆盖 Ψ 的代价 $\theta_\Psi(ES\,|\,T)$ 定义为其中所有匹配的代价之和:

$$\theta_\Psi(ES\,|\,T)=\sum_{\psi\in\Psi}\theta_\psi(EC_i\,|\,H_i)\,.$$

若 T 为句子 S 的一个可能的分析结果,则称 ES 为输入句子 S 的一个覆盖,记为 $ES_s=\Psi(T,S)$.因此,寻找输入句子 S 最优匹配 E-Chunk 集合的问题可以形式化地表述为寻找输入句子最低代价的覆盖 ES^*:

$$ES^*=\arg\min_{\Psi}\theta_\Psi(ES\,|\,S)=\arg\min_{\Psi}\sum_{\psi\in\Psi}\theta_\psi(EC_i\,|\,H_i).$$

1.3 系统结构

整个翻译模型由源语分析引擎、E-Chunk 匹配引擎和转换引擎共同驱动,其结构框架如图 1 所示.

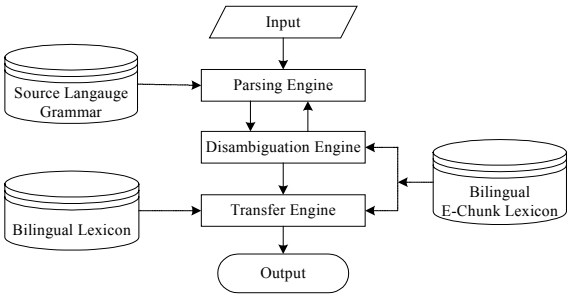

Fig.1 Architecture of the model

图 1 模型结构

我们说这是一种多引擎机器翻译模型,是因为在翻译模型处理过程中,使用了包括源语句法知识、双语词典和双语 E-Chunk 词典等多种资源,同时调动了多种处理机制共同参与控制决策:输入句子根据源语单语句法规则进行自底向上的结构分析,在分析的过程中同时进行 E-Chunk 匹配和代价计算,并以此作为部分分析结果的优选评价机制,从小到大逐步构造部分句法结构和相应的最优 E-Chunk 覆盖.分析过程结束后,选择整个句子的最优覆盖 ES^*,通过其中双语 E-Chunk 提供的转换信息,完成源语言向目标语言的翻译过程.

2 基于词汇相似的 E-Chunk 匹配代价计算

事实上,任何一种基于经验的语言学方法都难以避免数据稀疏问题,发现一个 E-Chunk 的完全匹配的机率是很低的.因此,如何有效地计算 E-Chunk 的匹配代价,确定类比推理中的相似原则,是 ECBMT 中的一个核心问题.本文建议使用基于词汇相似的编辑距离来表示 E-Chunk 的匹配代价.

2.1 词汇相似评估

词汇相似是指以一种上下文语境共享为特征的词汇二阶关联.这里,我们采取一种知识和统计方法相结合的相似计算的原则,分为 3 种情况进行讨论.

首先,我们将词汇分为语法词 W_F 和词汇词 W_C 两种基本类型.语法词是指语言中担任一些重要句法功能的封闭类型词汇的集合,如英语中的助动词、联系动词和介词;词汇词则是语法词的补集.因为语法词对句法结构起着重要的支撑作用,不能被替换或忽略,所以其被定义为不与其除本身以外的任何词汇相似.

其次,根据文本的领域特征,为一些相互间相似度非常高的词汇词,如数字,{morning,afternoon},{Monday, Tuesday,…,Sunday}等,预定义了一些等价类 C_E.属于同一个等价类的两个词汇,认为它们完全可以互换,其相似度 μ 的值定义为 1.

最后,两个不同的词汇词 w_1 和 w_2 之间的相似系数根据语料统计数据进行计算[12].令 $D(w_1\,\|\,w_2)$ 表示 w_1 和 w_2 的 Kullback-Leibler(KL)距离:

$$D(w_1 \| w_2) = \sum_w p(w \mid w_1) \log \frac{p(w \mid w_1)}{p(w \mid w_2)}.$$

令 $S(w_1)$ 表示与 w_1 在上述相似尺度上最为相似的元素的集合,则 w_1 和 w_2 之间的相似系数 $\mu(w_2, w_1)$ 定义为 $S(w_1)$ 中元素分布的加权平均:

$$\mu(w_2, w_1) = \sum_{w \in S(w_1)} p(w_2 \mid w) \frac{W(w, w_1)}{\sum_{w \in S(w_1)} W(w, w_1)},$$

其中 $S(w_1)$ 是根据 k-nearest 原则构造的,即给定经验参数 k,t,使 $|S(w_1)| \leq k$ 且对于 $\forall w \in S(w_1)$ 满足 $D(w_1 \| w_2) < t$. $W(w, w_1)$ 是词汇 w 的加权值,可以根据 w 与 w_1 的相似程度给定:

$$W(w_1, w_2) = \exp(-\beta D(w_1 | w)).$$

参数β控制着 $S(w_1)$ 中元素的相对分布对 W 的影响,β越大,则 S 中与 w 距离越远的词将获得更大的权值.

2.2 编辑距离

编辑距离(edit distance)是用来衡量两个字符串间相似程度的一种有效的方法[13].两个有限字符串 S 和 T 之间的编辑距离定义通过单符号增加(insertion)、删除(deletion)和替换(substitution)操作将 S 转换为 T 所需要的最小代价.编辑距离 $d_c(s^u, t^v)$ 的值可以由以下公式递归定义:

$$d_c(s^u, t^v) = \min \begin{cases} c(s_u, t_v) + d_c(s^{u-1}, t^{v-1}) \\ c(s_u, \varepsilon) + d_c(s^{u-1}, t^v) \\ c(\varepsilon, t_v) + d_c(s^u, t^{v-1}) \end{cases}.$$

其中 c 是替换操作代价函数.特殊地, $c(a, \varepsilon)$ 表示删除代价,而 $c(\varepsilon, a)$ 表示增加代价.因此 3 种操作的代价参数可以由一个大小为 $(|\Sigma|+1) \times (|\Sigma|+1)$ 的二维矩阵 C 统一包含,其中 $|\Sigma|$ 为词汇表的大小.矩阵中第 0 行和第 0 列对应空字符ε,矩阵中元素 $C(w,w')$表示将符号 w 替换为 w'需要的代价,而第 0 行元素表示增加代价,第 0 列元素表示删除代价.

作为初始条件 $d_c(s^j, t^0) = \sum_{k=0}^{i} C(t_k, 0)$, $d_c(s^0, t^i) = \sum_{k=0}^{i} C(0, t_k)$, $C(0,0) = 0$.应用动态规划算法,上述计算过程可以在 $O(m \times n)$ 的时间内完成.

我们根据词汇间的相似特征指派编辑距离计算中的操作代价.

(a) 替换代价.词 w 和 $w'(w \neq w')$ 之间的替换代价如下定义: $c(w, w') = 1 - \mu(w, w')$.显然,其相似程度越高,替换代价就越小.

(b) 增删代价.增删操作可以看做是 w 与空字符 ε 进行的一种特殊的替换操作,操作代价取决于 w 的句法功能和语义贡献.对英语而言,一般限定词、程度副词通常不对词汇语义和句法结构产生决定性影响,可以为其指定较低的增删代价.其余词汇的增删代价设为 1.

2.3 E-Chunk匹配代价计算

E-Chunk EC=<T,h,l,r>与其匹配 H:<h',T',l',r'>之间的匹配代价根据其依存参数间的编辑距离定义:

$$\theta(EC \mid H) = d_c(EC, H) = d_c(l, l') + d_c(r, r') + c(h, h'),$$

其中 $c(h,h')$ 为中心词的替换代价,H 参数 l'和 r'的值需要根据 EC 的类型而定:若 $EC \in ES_L(h)$,则定义为其直接子结点,否则为其表层的字符串实现.式中考虑了中心词必须互相匹配的情况,采用了其左右参数进行了单独计算然后求和的方法.事实上,由于 E-Chunk 的平面性构造原则,只有最大高度为 1 的片断或者子树片断与给定的 E-Chunk 有匹配意义,因此,其他类型的片断与任何 E-Chunk 之间的匹配代价均定义为∞.

但是,仅比较编辑距离的总值无法体现 EBMT 中常用的最大匹配原则[14].解决的方法是将编辑距离 d 分解为由增加距离α,删除距离β和替换距离γ组成的三维向量,当两组编辑距离 d 与 d'的差值小于给定的阈值δ时,则需要按照$\beta > \alpha > \gamma$的原则确定其优先关系,从而保证匹配长度较长的 E-Chunk 可以获得较高的优先级.

3 中心语驱动的分析策略

因为 E-Chunk 的匹配对象定义为分析树中的片断,所以寻找最优 E-Chunk 覆盖 *ES**的过程首先是源语的分析过程.在 ECBMT 模型中,我们以 ECTRAN 英汉机器翻译系统[15]中的分析子系统为基础,采用词汇化的中心语驱动图分析方法(head-driven chart parsing)[16]实现源语句法分析.这种分析方法的优势在于可以在自底向上的分析过程中,同时进行递增式的 E-Chunk 匹配代价计算和部分分析结果剪枝,而不必在分析结束之后,递归遍历整个分析森林来搜索 E-Chunk 的最优覆盖.

3.1 算法框架

设输入字符串 $S = w_1 w_2 \ldots w_n$,分析调度表(agenda)C 定义为大小为 $n \times n$ 的二维数组.一个自底向上的图分析过程基本上可以分为如下 3 个部分:

(1) 初始化.对于 $w_i (1 \le i \le n)$,将其相关规则加入 $C[i,i]$中,并将其初始化为起始状态.

(2) 成分组合.以自底向上的方式,将较小的成分组合为较大的成分,直至分析过程完成.

```
1.    for w=1 to n
2.        for start=1 to n–w
3.            end=start+w
4.            for mid=start to end−1
5.                foreach  sig1 ∈ C[start, mid]
6.                    foreach  sig2 ∈ C[smid+1, end]
7.                        Discover (sig1, sig2)
```

(3) 完成.组合过程完毕后,$C[1,n]$中所有处于完成态的部分分析就是整个句子的一个分析结果.

各种自底向上形式的表格分析算法的差异主要存在于上面算法的 Discover 子例程中.Discover 的任务在于将分析表中的两个部分分析 sig_1 和 sig_2 合并成跨度为(start,end)的更大的部分分析 sig.

3.2 成分组合算法

为结合 E-Chunk 匹配代价计算机制,调度表中的每个部分分析扩充表示为一个六元组:sig:<h,R,l,r,ES,c>,其中 R 为规则集中规则的惟一标识;h 为中心词;l 和 r 分别为当前需要匹配的左侧和右侧依存参数指针,并使用 $l \triangleright$ 和 $r \triangleright$ 表示其后继结点;ES 和 c 分别为与该部分分析对应的最优覆盖及其匹配代价.

在中心语驱动的图分析算法中,分析调度表中新的部分分析是由中心语开始,对其依存参数向两侧进行双向匹配生成的.设分析表中存在如下两个相邻的部分分析:

$$sig_1:<h_1,R_1,l_1,r_1,ES_1,c_1> \in C[\text{start,mid}],$$
$$sig_2:<h_2,R_2,l_2,r_2,ES_2,c_2> \in C[\text{start+1,mid}],$$

则 Discover 例程可以分为如下 3 个部分:

(1) 右向匹配.如果 $r_1 \ne nil$, $l_2 = r_2 = nil$,且 r_1 可以与 sig_2 相匹配,则将 sig_2 作为 sig_1 的一个子结点组成新的部分分析 sig_1:<h_1,R_1,l_1,$r_1 \triangleright$,$ES_1 \cup ES_2$,c_1+c_2>,并将其加入 C[start,end].此时如果有 $l_1 = r_1 \triangleright = nil$,则从 E-Chunk 词典中检索出 h_1 的相关 E-Chunk 集合 $ES(h_1)$,计算 sig 的最优覆盖及其匹配代价.sig 的最优覆盖一种可能是由一个与 sig 匹配代价最小的 ESL 类型的 E-Chunk EC_l 与 sig 子结点的最优覆盖组合而成,匹配代价由 EC_l 与 sig 的匹配代价与 sig 子结点的匹配代价求和得到.因为在部分分析的构造过程中已经将 sig 子结点的信息累计到 sig 之中,所以上述操作仅需在原有数据基础上加入 EC_l 的对应信息即可.另外一种情况则是一个 ES_G 类型的 E-Chunk EC_g 单独组成,匹配代价 $c = \theta(EC_g \mid sig)$.此时,需要使用 EC_g 和 c 替换 sig 中现存的子结点累计信息.

(2) 左向匹配.左向匹配的分析过程与右向匹配完全对称.

(3) 剪枝.我们使用部分分析最优覆盖的匹配代价作为动态规划的剪枝条件.如果一个分析表格中存在如下两个部分分析:

$$sig_1=<h,R,l,r,ES_1,c_1> \in C[i,j],$$
$$sig_2=<h,R,l,r,ES_2,c_2> \in C[i,j].$$

若有 $c_2 > c_1$,则可以安全地将 sig_2 从 $C[i, j]$ 中删除.

4 相关工作及讨论

面向数据的处理技术(data oriented processing,简称 DOP)[17]是另外一种基于经验数据的句法分析技术.DOP 中建议使用从实例中抽取得到的树库通过概率化的拼接操作进行句法分析.但 DOP 模型中的片断是完全结构化的,包含了句法分析树中所有可能的子图;Bod 提出的基于片断拼接操作的随机树替换文法(stochastic tree substitution grammar,简称 STSG)模型本质上也是对 ERF(expected rule frequency)模式的 PCFG(probabilistic context free grammar)的一个扩展.基于树结构的片断匹配操作是非常昂贵的,Sima'an[9]证明,所有基于树文法的语言模型,包括 TAG(tree adjoining grammar)和 STSG,求解其最可能分析(most probable parse,简称 MPP)的计算都是 NP 完全的.E-Chunk 的平面性结构原则是在多项式时间复杂度下求解全局最优 E-Chunk 匹配代价的一种有效方法.

目前一些基于分析树匹配的 EBMT 模型通常试图计算整个句子的最佳匹配实例[14],或者构造输入句子的最优划分[18].ECBMT 模型的特点在于,有效地结合了中心语驱动的分析过程与词汇化实例最优覆盖的构造过程,在这个过程中 E-Chunk 不但作为双语转换的知识源,同时也作为源语分析中消除词汇和结构歧义的知识源.此外,ECBMT 模型中还明确定义了匹配单元的构造原则:即或者是 EBMT 的最大化原则(覆盖整个子树片断),或者是 S&B 方法中的词汇化原则.

5 实验结果分析

我们从英汉双语法律文本中选取了 124 篇作为实验语料,计 1 436 句,约 32 600 词.其中的 120 篇被用于构造一个包含 1 521 个双语 E-Chunk 的小型双语 E-Chunk 词典,其余 4 篇计 57 句用于结果测试.词汇相似模型的参数则从全部的 305 篇英文法律文本中进行训练.测试中采用 ECTRAN 翻译系统中的分析子系统作为分析工具,并根据语料特点增加了一部分分析规则以实现对测试语料句法现象的基本覆盖.

实验的主要目的在于测试模型在源语分析和翻译转换中的效果,因此我们选取了以下几个性能指标:

(1) 源语分析的标记正确率(labled precision,简称 LP).LP 是指分析结果中正确成分占正确分析中总成分数的百分比,该指标用于衡量 E-Chunk 的剪枝优选机制对分析效果的影响.

(2) 译文中的缺失词率(missing word rate,简称 MWR).MWR 是指因 E-Chunk 近似匹配而导致译文中缺译的词汇占总词数的百分比.

(3) 译文中的冗余词率(redundant word rate,简称 RWR).RWR 指译文中冗余词汇占总词数的百分比.MWR 和 RWR 用于衡量 E-Chunk 对测试句子的覆盖程度.

实验中,我们还比较了模型在单独应用某种类型 E-Chunk 实例情况下的性能.实验数据见表 1.

Table 1 Experimental results

表 **1** 测试结果

	ES_L (%)	ES_G (%)	$ES_L \cup ES_G$ (%)
LP	87.6	80.1	84.2
MWR	11.5	15.4	9.8
RWR	16.7	17.6	14.2

在初步的实验结果中,LP,MWR 和 RWR 的情况都基本令人满意,一定程度上表明本文的方法在领域文本的处理中是有效的.我们从表中还可以看出,在同时应用两种类型的 E-Chunk 时,MWR 和 RWR 指标都要好于任何一种类型的单独应用时的情况.因为比较 E-Chunk 匹配代价时采用了最长匹配优先的原则,模型倾向于接受长度更大的 E-Chunk,RWR 的值要高于 MWR 是符合理论期望的.在独立应用 ESG 类型 E-Chunk 时 LP 较低是因为 ESG 类型 E-Chunk 具有较大的匹配粒度,句子中总成分数量比较少的缘故.

6 结 论

E-Chunk 定义为具有语义自足性和转换充分性的机器翻译基本单元.本文提出了一种基于 E-Chunk 的机器翻译模型.在该模型中,我们将词汇相似技术应用于 E-Chunk 匹配代价计算,并提出了一种与中心语驱动分析技术相结合的有效的最优 E-Chunk 覆盖求解方法.在法律文本自动翻译方面的初步实验表明这种方法是可行的.我们今后的工作将主要集中在以下两个方面:(1) 扩大双语 E-Chunk 词典的规模,提高 E-Chunk 实例对语言事实的覆盖程度;(2) 改进词汇相似模型,并增加训练语料容量,提高模型参数的可信度.

References:

[1] Arnold, D., Balkan, L., Humphreys, R.L., *et al*. Machine Translation, an Introductory Guide. Machesster-Oxford: NCC Blackwell, 1994.

[2] Nagao, M. A framework of a mechanical translation between Japanese and English by analogy principle. In: Elithorn, A., Banerji, R., eds., Artificial and Human Intelligence. Amsterdam, New York: Elsevier Science Publishers Corporation, 1984. 173~180.

[3] Sato, Satoshi, Makoto, Nagao. Toward memory-based translation. In: Proceedings of the 13th International Conference on Computational Linguistics (COLING'90), Vol.3. Helsinki, Finland: Helsingiensis Universitas, 1990. 247~252.

[4] Kaji, Hiroyuki, Yuuko, Kida, Yasutsugu, Morimoto. Learning translation templates from bilingual text. In: Proceedings of the 15th [sic] International Conference on Computational Linguistics (COLING'92). Nates: the Association, ICCL, 1992. 672~678.

[5] Abney, S. Parsing by chunk. In: Tenny, B.A., eds. Principle-Based Parsing. Nowell, MA: Kluwer, 1991.

[6] Abney, S. Partial parsing via finite-state cascades. Natural Language Engineering, 1996,2(4):337~344.

[7] Pollard, C., Ivan, S. Head-Driven Phrase Structure Grammar. Centre for the Study of Language and Information, Stanford University, 1994.

[8] Yael, K., Edelman, S. Learning similarity-based word sense disambiguation. Computational Linguistics, 1998,24(1):41~60.

[9] Sima'an, Khalil. Computational complexity of probabilistic disambiguation by means of tree-grammars. In: Proceedings of the COLING'96. Copenhagen: the Association, Morristown, NJ, 1996.

[10] Whitelock, Pete. Shake and bake translation. In: Rupp, C.J., Rosner, M.A., Johnson, R.L., eds. Constraints, Language and Computation. London: Academic Press, 1994. 339~359.

[11] Dorr, B. Machine Translation: a View from Lexicon. Cambridge, MA: MIT Press, 1993.

[12] Lee, L.J. Similarity-Based approaches to natural language processing [Ph.D. Thesis]. Harvard University, 1997.

[13] Hall, P., Dowling, G. Approximate string matching. Computing Surveys, 1980,12(4):381~402.

[14] Sato, Satoshi. CTM: an example-based translation aid system using the character-based best match retrieval method. In: Proceedings of the COLING'92. Nantes: the Association, ICCL, 1992.

[15] Yao, Tian-shun, Li, Jing-jiao, Liu, Dong-li, *et al*. Natural Language Understanding. Beijing: Tsinghua University Press, 1995 (in Chinese).

[16] Alshawi, H. Head automata and tree tiling: translation with minimal representations. In: Proceedings of the Association for Computational Linguistics. Santa Cruz, CA: Morgan Kaufmann Publishers, 1996. 167~176.

[17] Bod, R. Using and annotated corpus as stochastic grammar. In: Proceedings of the EACL'93. Utrecht: the Association, Morristown, NJ, 1993.

[18] Nirenburg, Sergei, Constantine, Demashnev, Grannes, D.J. Two approaches to matching in EBMT. In: Proceedings of the 5th International Conference on Theoretical and Methodological Issues in Machine Translation. Amsterdam: IOS Press, 1993.

附中文参考文献:

[15] 姚天顺,李晶皎,刘东立,等.自然语言理解.北京:清华大学出版社,1995.

A Machine Translation Model Based on E-Chunk*

LI Mu, LÜ Xue-qiang, YAO Tian-shun

(*Institute of Computer Science and Engineering*, *Northeastern University*, *Shenyang* 110006, *China*)

E-mail: ics@mail.neu.edu.cn

http://www.nlplab.com

Abstract: In this paper, a new E-Chunk based multi-engine machine translation model is proposed. The model is composed of a head-driven lexicalized parser, a word-similarity based E-Chunk match engine and a bilingual E-Chunk based transfer engine. The optimal E-Chunk tiling is constructed in a bottom-up style efficiently. Preliminary experimental results show that it is effective in domain oriented machine translation.

Key words: E-Chunk; machine translation; word similarity computation

* Received August 21, 2000; accepted December 19, 2000

Supported by the National Natural Science Foundation of China under Grant No.69985001; the National Grand Foundamental Research 973 Program of China under Grant No.G19980305011; the National Research Foundation for the Doctoral Program of Higher Education of China under Grant No.1999014503

2002 年全国开放式分布与并行计算学术会议(DPCS 2002)

征 文 通 知

2002 年全国开放式分布与并行计算学术会议(简称 DPCS 2002)由中国计算机学会开放系统专业委员会主办,华中科技大学计算机科学与技术学院承办,湖北省、武汉市计算机学会协办,定于 2002 年 10 月 24~26 日在武汉召开.

一、征文范围

(1) 开放式分布与并行计算系统,包括系统体系结构、算法与优化、语言与编译、存储方法与数据结构、操作系统与数据库、多机与群集系统、系统平台与程序设计环境以及性能分析与评价等;

(2) 开放式分布异构环境的处理技术,包括这类环境的信息集成、互操作以及环境安全等技术;

(3) 开放式网络技术与应用,主要包括网络计算与分布式计算、移动计算与代理技术、数据挖掘与数据仓库，以及网络安全技术等;

(4) 开放式多媒体处理与并行计算技术,主要包括图形图像理论与算法,语音、视频处理与人机交互、模式识别等;

(5) 上述领域的发展趋势和综合评论.

二、征文要求

(1) 论文应是未正式发表的、或者未正式等待刊发的研究成果;

(2) 论文格式仿照《计算机研究与发展》刊物的格式,应包含题目、摘要、关键词、正文和参考文献;

(3) 论文中、英文均可,一般不超过 5000 字,一律用 Word2000 格式排版,提供 A4 激光打印稿一式 2 份,并随寄软盘;

(4) 论文稿件和软盘不论录取与否,恕不退稿,请自留底稿;

(5) 邮寄论文时须在信封左下角或 E-mail 主题中注明《DPCS2002》;

(6) 经程序委员会审查合格的论文,将收录在会议论文集或者推荐到相关期刊发表.论文一律寄给武汉地区联系人.

三、重要日期与联系方式

论文截稿日期: 2002 年 6 月 30 日　　录用通知日期: 2002 年 8 月 1 日

武汉地区联系人: 卢正鼎,李瑞轩　湖北武汉华中科技大学计算机科学与技术学院(430074)

电话: 027−87544285;87543884　传真: 027−87545004

E-mail: rxli@public.wh.hb.cn

北京地区联系人: 陈炳从　北京 619 信箱 63 号(100083)

电话: 010−62311951

会议主页: http://cs.hust.edu.cn/dpcs2002