汉英双语库自动分段对齐研究*

王斌，刘群，张祥

（中国科学院 计算技术研究所，北京 100080）

E-mail：{wangbin,liuqun}@mtgroup.ict.ac.cn

http://www.ict.ac.cn

摘要：双语库对齐是自然语言处理研究的重要课题之一.其中，双语库段落对齐是指找出源文和译文中对应的翻译段落的对齐过程.它上承篇章对齐，下接句子对齐，在整个双语库的对齐中起着承上启下的重要作用.但由于种种原因，双语库段落对齐在当今研究中没有受到应有的重视.直接进行通用的段落自动对齐具有相当的难度，也不必要，鉴于此，提出将段落对齐转化成分段对齐进行实现.通过汉英词汇对之间的特征比较，首先找到可以用于汉英双语库分段的锚点词汇对，在此基础上，通过锚点词所在句子的匹配获得锚点句子对来进行分段.实验结果表明，该方法具有良好的使用价值，可以用于其他语言对的对齐.

关键词：自然语言处理；双语库；对齐；分段；锚点

中图法分类号：TP391 **文献标识码**：A

近年来，语料库语言学的兴起是计算语言学中的重要事件.语料库以其覆盖面广、语料真实、信息丰富而为计算机自然语言处理提供了强有力的支持.双语语料库（以下简称双语库）是一种特殊形式的语料库，它同时含有两种语言的对译语料信息，因此，加工后的双语库与单一的语料库相比，更具有两种语言之间的匹配信息，它可以用于基于统计的机器翻译（statistics based machine translation，简称 SBMT）[1]、基于实例的机器翻译（example based machine translation，简称 EBMT，亦称 memory based machine translation）[2]、机助人译[3]、双语词典和术语库的建立[4,5]、翻译知识的抽取[6]、词义排歧[7]等多种应用领域，具有很高的利用价值.

目前，基于双语库的工作主要包括两个方面：(1) 对双语库的加工，主要是对齐（alignment），即找出双语文本之间的各级对译关系[8~12]；(2) 从已经对齐的语料库中抽取知识并加以利用[1~7].双语库的对齐单位包括篇章、段落、句子、短语、单词等.

所谓段落对齐就是找出源文中的段落在译文中对应的翻译段落.由于段落上承篇章，下接句子，因此段落对齐也起着从篇章对齐到句子对齐的承上启下作用.然而，在当今双语文本对齐的研究中，段落自动对齐有意无意地受到了冷落.这种冷落主要表现为：(1) 几乎没有以段落对齐为主题的研究论文；(2) 很多学者在进行双语文本对齐的研究时，都假定双语文本已经做到了段落一级的对齐，而段落对齐的实现讨论得很少[8~12]；(3) 一些学者认为，段的对齐显然比句子对齐容易得多，因此，对段落对齐只是轻描淡写地一笔掠过，比如，Gale[9]认为可以使用与句子对齐类似的方法进行段落自动对齐.

然而，实际情况却是，(1) 现实存在的大量电子双语文本都没有做到段落对齐，如果通过手工进行段落对齐，工作量之大是显而易见的.并且，如果不进行段落对齐，下一步的对齐工作简直无法进行；(2) 由于段落的粒度较大，看上去它的对齐似乎比句子对齐更容易，实际上，它对正确率的要求远远高于句子对齐，因为一旦某个段落出现对齐错误，就会造成段内的句子以及后续段落、句子对齐的错误率急剧上升，也使后续的对齐失去了意

* 收稿日期：1999-06-10；修改日期：1999-09-10

基金项目：国家 863 高科技项目基金资助项目（863-306-03-06-2）

作者简介：王斌（1972－），男，江西波阳人，博士，副研究员，主要研究领域为自然语言处理，网络信息处理；刘群（1966－），男，江西萍乡人，副研究员，主要研究领域为自然语言处理，机器翻译；张祥（1942－），男，江苏张家港人，研究员，博士生导师，主要研究领域为计算机系统结构，自然语言处理.

义.因此,正确地进行段落自动对齐不仅十分重要而且相当必要.

进行段落自动对齐的困难还在于:(1) 现实的许多电子文本并没有明显的段落边界标志,或者根本没有段落之分;(2) 即使电子文本具有明显的段落边界,但由于段落的粒度较大,翻译者在进行翻译时,对译文段落进行段落重组的可能性加大,加上翻译人员的个人喜好和随意性,也会造成多种可能的复杂的翻译模式.由于上述原因,要建立一个通用的段落自动对齐工具是相当困难的.鉴于此,本文提出一个将文本依照翻译块(translation block)重新进行分段的方法,既避免了段落自动对齐的难度,又达到了段落对齐的真正目的.

1 对齐的形式化定义

"对齐"这个词,既用于表示寻找不同语言文本之间互译片断的过程(align),也常常用于表示该过程产生的结果(alignment).一个结果意义上的对齐可以形式化定义如下:

定义 1. 假设源文文本 S 以及其对应的译文文本 T 分别可以看成是 n 个源文片段集合 $S=\{s_1,s_2,s_3,\ldots,s_n\}$ 及 m 个译文片断集合 $T=\{t_1,t_2,t_3,\ldots,t_m\}$,$\mathscr{P}(S)$,$\mathscr{P}(T)$分别是 S 和 T 的幂集,$\mathscr{P}(S)\times\mathscr{P}(T)$是两者的笛卡尔积,$\mathscr{P}(S)\times\mathscr{P}(T)$的任一个子集 $\mathscr{A}=\{\mathscr{A}_1,\mathscr{A}_2,\mathscr{A}_3,\ldots,\mathscr{A}_l\}$,$\mathscr{A}_i=\langle\mathscr{A}_{iS},\mathscr{A}_{iT}\rangle$ $(i=1,2,\ldots,l,\mathscr{A}_{iS}\in\mathscr{P}(S),\mathscr{A}_{iT}\in\mathscr{P}(T))$,如果 $\mathscr{A}$ 满足:

(1) 若 $S\cup T\neq\varnothing$,则对任一 i,$\mathscr{A}_{iS}\cup\mathscr{A}_{iT}\neq\varnothing$;

(2) (无二义性)任意 i,j,若 $i\neq j$,则 $\mathscr{A}_{iS}\cap\mathscr{A}_{jS}=\varnothing$,$\mathscr{A}_{iT}\cap\mathscr{A}_{jT}=\varnothing$;

(3) (覆盖性)对于所有的 i,满足:

$$\bigcup_{i=1}^{l}\mathscr{A}_{iS}=S,\quad \bigcup_{i=1}^{l}\mathscr{A}_{iT}=T,$$

则称 $\mathscr{A}$ 为源文 S 和译文 T 的一个对齐.集合 $\mathscr{A}$ 的大小 $l=|\mathscr{A}|$称为对齐的长度.三元组$\langle S,T,\mathscr{A}\rangle$称为一段双语文本(bitext),$\mathscr{A}$ 的每个元素$\langle\mathscr{A}_{iS},\mathscr{A}_{iT}\rangle$称为一个双语片断(bisegment)或者一个双语串(bead),$|\mathscr{A}_{iS}|:|\mathscr{A}_{iT}|$称为一个匹配模式(matching pattern).

特别地,如果每个 $\mathscr{A}_{iS}$或 $\mathscr{A}_{iT}$包含的是 0 个、1 个或多个段落,则 $\mathscr{A}$ 是一个段落对齐.

性质(1)保证在非空双语文本的对齐中,没有空到空的双语片断.

性质(2)保证源文文本和译文文本的任一片断只存在于对齐的某一唯一的双语片断中.

性质(3)保证源文文本和译文文本的任一片断必定存在于某双语片断中,并且任一双语片断均由源文片断和译文片断组成.

为了说明以上概念,我们举如下的一个例子(见表 1).

Table 1 An example of alignment

表 1 一个对齐的例子

s_1	它是把数据转换成一种不用秘密的解密钥匙就不能读出的形式的过程	t_1	It is the process of converting data into a form that is unintelligible without the secret decryption key.
s_2	首先,让我们解释几个术语上的拦路虎.在密码学中,没有加密的任何形式的文件叫做普通文本,而加密的数据则叫做密码文本.	t_2	First, let's get a few terms out of the way.
		t_3	In cryptography, a file of any type that isn't encrypted is called plaintext; encrypted data is called ciphertext.

在表 1 所示的例子中,源文 $S=\{s_1,s_2\}$,译文 $T=\{t_1,t_2,t_3\}$.该对齐的形式化表示为 $\mathscr{A}=\{(\{s_1\},\{t_1\}),(\{s_2\},\{t_2,t_3\})\}$,$\mathscr{A}$ 的长度为 2,两个双语片断$(\{s_1\},\{t_1\})$,$(\{s_2\},\{t_2,t_3\})$的匹配模式分别为 1:1,1:2.

从定义 1 中不难看出,对于同一源文和同一译文,满足定义的对齐不止一个,实际上,我们所希望找到的对齐满足"最小片断"原则,即在正确的前提下尽可能地使对齐片断所含的对齐单位数较少.举个简单的例子,在进行句子级的对齐时,我们就尽可能地使双语片断所含的句子数较少,如果我们能找到句子 s_1 与 t_1,s_2 与 t_2 互为翻译,那么在最后的对齐中希望出现双语片断$(\{s_1\},\{t_1\})$,$(\{s_2\},\{t_2\})$,而不希望出现双语片断$(\{s_1,s_2\},\{t_1,t_2\})$.

需要说明的是,由于交叉依赖的复杂性,实际求得的对齐 $\mathscr{A}=\{\mathscr{A}_1,\mathscr{A}_2,\mathscr{A}_3,\ldots,\mathscr{A}_l\}$,$\mathscr{A}_i=\langle\mathscr{A}_{iS},\mathscr{A}_{iT}\rangle$ 还要满足以下假设(次序性假设):

任意 $\mathscr{A}_i,\mathscr{A}_j$,若 $i<j$,则对任意 $s_a\in\mathscr{A}_{iS}$,$s_b\in\mathscr{A}_{jS}$,$t_c\in\mathscr{A}_{iT}$,$t_d\in\mathscr{A}_{jT}$,有 $a<b$,$c<d$.

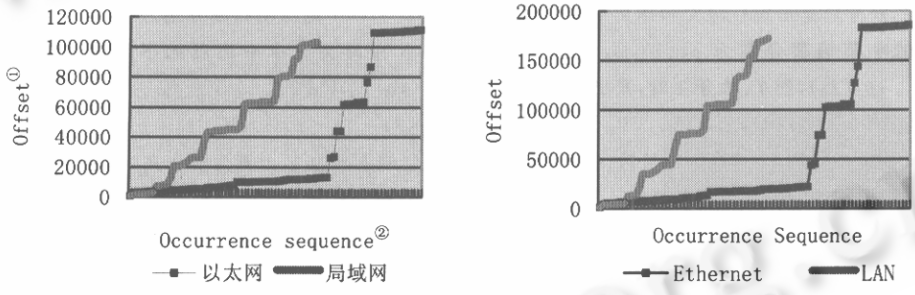
2 分段对齐的锚点选择

所谓分段对齐是指,将源文和译文重新划分成多个相互翻译块(我们也将这个块称为段)的过程.分段对齐实现的关键技术是分段,而分段的原则又是使分出来的段落互相对齐,即对齐是分段的目的.它与段落对齐的最大不同之处在于它忽视已有的段落边界,而根据对齐的目的来重新组织段落.它们的相同之处都是以输出对齐的翻译段落为目的.分段对齐的优越之处在于它不受是否已有段落边界、已有边界是否清晰的限制,而是根据对齐这个根本的目的重新组织段落,具有很强的通用性.

分段对齐的任务就是寻找可以用于分段的边界,这个边界也常常称为"锚点"(anchor).我们寻找的锚点往往是双语文本中具有这样属性的词、短语、句子或者段落对,即它们具有说明自己是一对互译片断的相对明显的特征.

为了选取词汇级的锚点,本文选取出现频率较高的翻译固定词汇作为考察对象,进行了如下统计实验.

首先,我们对词的频率信息进行了统计考察.在我们的实验汉英对照文本中,汉语词"以太网"、"局域网"分别出现了88次和57次,相应地,"Ethernet"和"LAN"分别出现了87次和48次,可见,翻译相对固定的词对的出现频率比较相近.其次,我们对词的位置信息进行了考察."以太网"在汉语文本中的字节偏移构成一个维数为88的偏移向量〈718,2405,3061,3148,3180,...,111026,111268〉,"局域网"的偏移向量为〈931,1850,2289,2311,2478,...,102951,102967〉,长度为58.相应地,"Ethernet"和"LAN"分别对应于维数为87和48的偏移向量〈1326,4147,5277,5417,5448,...,184983,185903〉和〈1689,3979,4001,4272,4313,...,170485,172462〉.它们在文本中出现的示意图如图1所示.



①字节偏移,②出现次序.

Fig. 1 Offset of Chinese and English word in texts

图1 汉英词汇字节偏移图

从图1中可以看出,"以太网"与"Ethernet"、"局域网"与"LAN"的字节偏移分布十分相似,如果说这种相似还不明显的话,我们再来计算这些词的出现间隔向量,即由相邻的两次出现的字节偏移之间的差值组成的向量.对于以上各词,它们的出现间隔向量分别是:

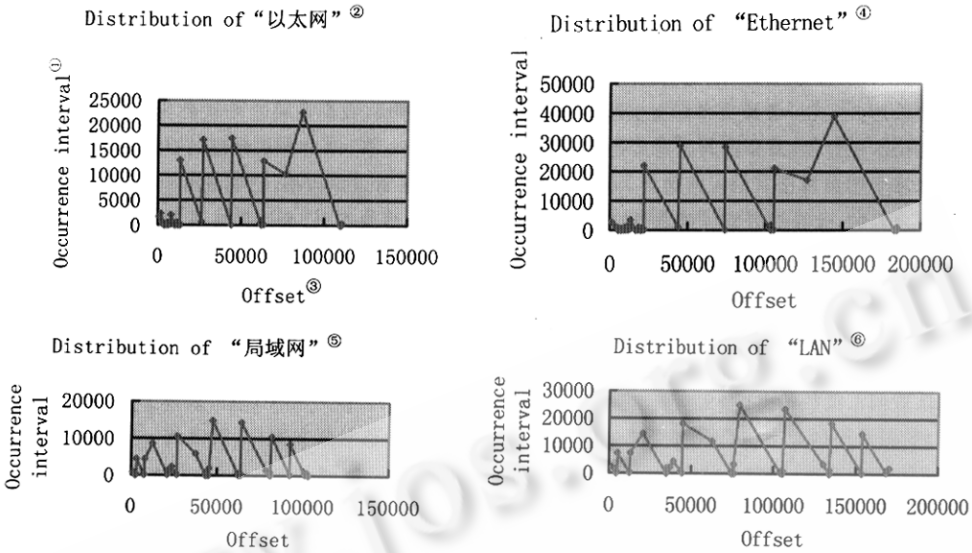
"以太网":〈1687,656,87,32,...,242〉,向量维数为87,

"局域网":〈919,439,22,167,...,16〉,向量维数为57,

"Ethernet":〈2821,1130,140,31,...,920〉,向量维数为86,

"LAN":〈2290,22,271,41,...,1977〉,向量维数为47.

如果以词的出现偏移向量为X轴值,该词的出现间隔向量为Y轴值,则可以画出如图2所示的词汇出现间隔分布曲线图.在图2中,"以太网"与"Ethernet"、"局域网"与"LAN"的分布曲线惊人地相似,而"以太网"与"LAN"的分布曲线又有明显的不同.



①出现间隔,②"以太网"分布,③字节偏移,④"Ethernet"分布,⑤"局域网"分布,⑥"LAN"分布.

Fig. 2　Occurrence interval distribution of Chinese and English words

图 2　汉英词汇出现间隔分布

有了以上特征,我们使可以通过计算词对出现间隔向量的相似度来达到选取词汇级锚点的目的.

设 A,B 分别是出现在源文和译文中的单词,它们在各自文本中的出现频率分别为 f_A 和 f_B,A 在源文文本中出现的字节偏移分别为 $p_A(1),p_A(2),\ldots,p_A(f_A)$,$B$ 在译文文本中出现的字节偏移分别为 $p_B(1),p_B(2),\ldots,p_B(f_B)$,当 f_A 大于 1 时,其出现间隔向量 $V_A=\langle V_A[1],V_A[2],\ldots,V_A[f_{A-1}]\rangle$,当 f_B 大于 1 时,出现间隔向量 $V_B=\langle V_B[1],V_B[2],\ldots,V_B[f_{B-1}]\rangle$,$V_A[i]=p_A(i+1)-p_A(i)$,$V_B[j]=p_B(j+1)-p_B(j)$,$i=1,2,\ldots,f_A-1$,$j=1,2,\ldots,f_B-1$. 显然,出现间隔向量分量值的下界为 1,上界为所在文本的长度.

由于不同词的间隔向量一般具有不同的维数,所以,我们在计算它们之间的距离时,引入了模式识别中常用于频谱信号匹配的动态时间伸缩算法 DTW(dynamic time wrapping). DTW 算法实际上是动态规划(dynamic programming)的一种. 在这种算法中,在两个模式进行匹配时,一个模式可以看成另一个模式的失真,两者的失真度(距离)就能衡量两者之间的相似度,失真度越低,相似度越高. DTW 算法通过部分失真的累积计算两个模式的总失真.

形式地,两个词 A 和 B 出现间隔向量的相似度可以通过向量间的最短距离 $DIV(A,B)$ 来计算,$DIV(A,B)$ 的计算过程如下:

(1) 初始化

$Dist(0,j)=0$,$Dist(i,0)=0$,$Dist(1,1)=d(1,1)$

$Dist(1,t)=\min(Dist(1,t-1),d(1,t))$,$Dist(s,1)=\min(Dist(s-1,1),d(s,1))$

$i=0,1,2,\ldots,M$,$j=0,1,2,\ldots,N$,$s=1,2,3,\ldots,M$,$t=1,2,3,\ldots,N$

$M=\dim(V_A)$,$N=\dim(V_B)$,M,N 分别为 V_A,V_B 的维数

$d(s,t)=|c\times V_A(s)-V_B(t)|$

$Dist(s,t)$ 表示向量 $V_A[1\ldots s]$ 与向量 $V_B[1\ldots t]$ 的最短距离

$d(s,t)$ 表示分量 $V_A[s]$ 与分量 $V_B[t]$ 的距离

c 为归一化常数,本算法中取两个文件长度比值.

(2) 递推计算

$$Dist(m,n)=\min_{\substack{1\leqslant a,b\leqslant 2,ab\neq 4\\ \text{且}Dist(m-a,n-b)\geqslant 0\\ \text{且}(m-a+1,n-b+1)\\ \text{满足路径限制}}}(Dist(m-a,n-b)+d(m-a+1,n-b+1)).$$

若 $Dist(m,n)\geqslant 0$,则

$$Path(m,n)_x=\arg\min_a Dist(m,n),\ Path(m,n)_y=\arg\min_b Dist(m,n),$$

$$m=2,3,\ldots,M,n=2,3,\ldots,N.$$

(3) 递推结束

$$DIV(A,B)=Dist(M,N).$$

(4) 路径回溯

$$\psi_1=\langle M+Path_x(M,N)-1,N+Path_y(M,N)-1\rangle,$$

$$\psi_{r-1}=\langle x_{r-1},y_{r-1}\rangle,$$

$$\psi_r=\langle x_r=(x_{r-1}-1)+Path_x(x_{r-1}-1,y_{r-1}-1)-1,y_r=(y_{r-1}-1)+Path_y(x_{r-1}-1,y_{r-1}-1)-1\rangle.$$

输出路径(长度为 h):$(\psi_h,\psi_{h-1},\ldots,\psi_2,\psi_1)$

该路径经过的点依次为

$$\langle A[x_h],B[y_h]\rangle,\langle A[x_{h-1}],B[y_{h-1}]\rangle,\ldots,\langle A[x_1],B[y_1]\rangle,$$

$$r=2,3,\ldots,h+1,x_{h+1}=0 \text{ 或者 } y_{h+1}=0.$$

平均出现间隔向量距离为

$$ADIV(A,B)=DIV(A,B)/h$$

根据以上过程,可以对任意一对分别出现在汉英文本中词汇对的平均出现间隔向量进行计算,比如,通过计算可得

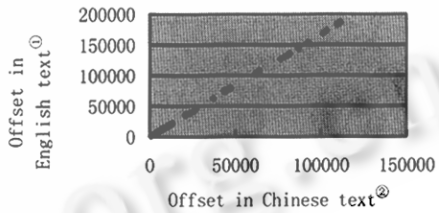
$$ADIV(\text{“以太网”},\text{“Ethernet”})=73.31,$$

$$ADIV(\text{“局域网”},\text{“LAN”})=821.15,$$

$$ADIV(\text{“以太网”},\text{“LAN”})=2696.51,$$

$$ADIV(\text{“局域网”},\text{“Ethernet”})=912.53.$$

由于“以太网”与“Ethernet”、“局域网”与“LAN”的平均出现间隔向量距离比“以太网”与“LAN”、“局域网”与“Ethernet”的平均出现向量距离短,因此可以认为,“以太网”与“Ethernet”、“局域网”与“LAN”更可能互为翻译.将它们的最短路径所经过的点重新还原为字节偏移向量,然后分别以这些点的汉语文本中字节偏移为 X 轴值,以这些点的英语文本中字节偏移为 Y 轴值,画出汉英词汇字节偏移对应图(如图 3 所示).



①英语文本中字节偏移,②汉语文本中字节偏移.

Fig. 3 Offset distribution of Chinese-English anchors

图 3 汉英锚点偏移分布图

根据以上过程,可以将求得的平均间隔向量距离较小的词汇对作为我们的词汇级锚点,再根据它们的匹配路径(即匹配的位置对)所在的句子的互译信息率(句子对中互为翻译的词汇字节数占总字节数的比率)和长度关系的检查,便可以选取句子级别的锚点,从而实现汉英双语库的分段对齐.

3 分段对齐的算法实现

分段对齐的算法实现过程如下:

(1) 扫描汉语及英语文本,对汉语文本进行切分、标注;

(2) 计算汉语名词、二元单字接续对、三元单字接续对的出现频率、字节偏移向量及出现间隔向量,计算每个英语单词的出现频率、字节偏移向量以及出现间隔向量;

(3) 对满足限制条件的任意一对汉英词对 A 和 B,计算其平均出现间隔向量距离;

(4) 根据间隔向量距离的大小,从小到大输出满足一定阈值 t_d 的相应词对及其匹配路径;

(5) 检查匹配路径所在句子对的互译信息率及长度关系(如比值),输出其中互译信息率大于阈值 t_c、长度关

系值满足阈值 t_l 的句子对；

(6) 根据这些锚点句子将双语文本分段.

为了减少比较的词对数，算法的步骤(2)、(3)在实现时选取的汉英词汇满足下列的限制：

(1) 只统计汉语的名词和可能的未定义词(二元接续对、三元接续对)，不统计含有助词的汉语接续对；

(2) 不统计英语的冠词、常用介词、人称代词.

为了防止过多的词对进行比较，本算法实现时引入了以下限制条件：

(1) (频率限制)任意出现频率小于某一预定阈值 t_f 的词不参加间隔向量距离的比较；

(2) (频率关系限制)两个词 A 和 B，如果其中一个词的出现频率大于另一个词的出现频率的两倍，可以认为它们是相互翻译的可能性极小，这种情况不用计算它们之间的间隔向量距离；

(3) (偏移关系限制)两个词 A 和 B，如果其中一个词的首次字节偏移与另一个词的首次字节偏移之差大于1/2文本长度(汉或英)，则它们是相互翻译的可能性较小，在这种情况下也不用计算它们之间的间隔向量距离；

(4) (分布关系限制)两个词 A 和 B，假设它们的出现间隔向量的平均值分别是 c_A 和 c_B，均方差分别是 σ_A 和 σ_B，如果值 $(c\times c_A-c_B)^2-(c\times\sigma_A-\sigma_B)^2$ 大于一定的阈值，则认为它们是相互翻译的可能性较小而不用计算它们的间隔向量距离.

以上限制条件可以大大加快算法的运行速度.

4 实验结果及结论

利用以上算法，我们对总共 347K 的汉英对照文本进行了实验(其中汉语 132K，英语 215K)，通过相似性度量输出如表 2 所示的结果(前 7 项)：

Table 2 The most similar output word pairs

表 2 实验中相似度较大的词对输出

Chinese word①	Frequency②	English word③	Frequency	Average DIV④
以太网	88	Ethernet	87	72.78
服务器	188	Server	188	95.92
用户	179	User	152	126.90
数据	227	Data	197	145.34
网络	214	Network	232	157.23
系统	202	System	158	236.60
软件	95	Software	80	259.12

①汉语单词，②出现频率，③英语单词，④平均距离.

其中总共产生 251 对路径词对(带位置信息的词对)，检查路径词对所在的句子，通过互译信息率(使用双语词典及输出锚点词对)和长度关系的检查，剔除那些可能不是分段界点的句子对，最后输出了 203 对锚点句子.这些句子将汉英文本各分成 204 段，平均每段汉语文本约有 0.65K 字节，平均每段英语文本约有 1.1K 字节.经手工检查，这些句子对互为翻译的正确率为 100%.因此，根据这些锚点句子对进行双语文本分段对齐的正确率为 100%.

显然，本方法十分适合于具有较多高频固定词的双语文本的分段对齐(例如，中英对照专业文献)，对于只具有较少高频固定词的双语文本，本方法可能会遇到一定的数据稀疏问题，造成划分的“段”太“粗”.下一步，我们将一方面通过数据平滑方法，另一方面通过引入其他锚点(如数字、问句、标题)，来达到将划分的段落“细化”的目的.

本文方法为后续细粒度的对齐提供了重要而坚实的基础，显然也可以用于术语的抽取等其他应用.其思路适合于其他语言对的自动分段对齐，具有重要的实用价值.

References:

[1] Brown, P. F., Cocke, J., Della Pietra, S. A., *et al*. A statistical approach to machine translation. Computational Linguistics, 1990,16(2):79～85.

[2] Sato, S., Nagao, M. Towards memory-based translation. In: Proceedings of the 13th International Conference on Computational Linguistics (COLING'90). Helsinki, Finland, 1990. 247~252.

[3] Isabelle, P. Bi-Textual aids for translators. In: Proceedings of the 8th Annual Conference of the UW Center for the New OED and Text Research. Waterloo, Canada, 1992.

[4] Langlois, L. Bilingual concordancers: a new tool for bilingual lexicographers. In: Proceedings of the 2nd Conference of the Association for Machine Translation in the Americas (AMTA'96). Montreal, Canada, 1996.

[5] Wu, D., Xia, X. Large-Scale automatic extraction of an English-Chinese translation lexicon. Machine Translation, 1995,9(3~4):285~313.

[6] Guvenir, H.A., Cicekli, I. Learning translation templates from examples. Information Systems, 1998,23(6):353~363.

[7] Gale, W., Church, K., Yarousky, D. A method for disambiguating word senses in a large corpus. Computer and Humanities, 1993,(26):415~439.

[8] Fung, P., McKeown, K. Aligning noisy parallel corpora across language groups: word pair feature matching by dynamic time warping. In: Proceedings of the 15th Conference of the Association for Machine Translation in the Americas (AMTA'94). Columbia, USA, 1994. 81~88.

[9] Gale, W.A., Church, K.W. A program for aligning sentences in bilingual corpora. Computational Linguistics, 1993, 19(2):75~102.

[10] Simard, M., Plamondon, P. Bilingual sentence alignment: balancing robustness and accuracy. Machine Translation, 1998,13(1):59~80.

[11] Kay, M., Röscheisen, M. Text-Translation alignment. Computational Linguistics, 1998,13(1):59~80.

[12] Ker, S.J., Chang, J.S. A class-based approach to word alignment. Computational Linguistics, 1997,23(2):313~343.

Automatic Chinese-English Paragraph Segmentation and Alignment

WANG Bin, LIU Qun, ZHANG Xiang

(*Institute of Computing Technology, The Chinese Academy of Sciences, Beijing* 100080, *China*)

E-mail: {wangbin,liuqun}@mtgroup.ict.ac.cn

http://www.ict.ac.cn

Received June 16, 1999; accepted September 16, 1999

Abstract: The research on bilingual corpora is one of the most important topics in the field of natural language processing. Paragraph alignment is to find the translation paragraph pairs between source text and target text. Paragraph alignment is a necessary and important phase between section alignment and sentence alignment, but it is disregarded at present. This paper aims at dealing with this process. Because it is difficult and unnecessary to align paragraphs directly, this problem is changed into a "segmentation and alignment" one. This paper finds some anchors for segmentation first, and then matches the sentences which contain these anchors, finally checks the validity of the results and segments the texts into translation blocks. The experimental results show that the method can achieve good results and can also be applied to other language pairs.

Key words: NLP (natural language processing); bilingual corpora; alignment; paragraph segmentation; anchor