一种基于句法的用于汉英翻译的预调序方法*

吴秋锋，黄书剑，戴新宇，陈家骏
南京大学 计算机软件新技术国家重点实验室，江苏 南京 210093
E-mail: {wuqf, huangsj, dxy}@nlp.nju.edu.cn; chenjj@nju.edu.cn

摘 要：本文提出一种基于句法的预调序方法来解决基于短语的汉英翻译系统中的调序问题。该方法使用训练语料的源语言句法树和词对齐信息来自动抽取调序规则，并用规则调整训练和测试语料源语言句法树，使得源语言句子的语序更加接近目标语言句子。翻译系统使用从调序后的句法树重新生成的训练和测试语料句子作为输入进行训练和翻译。该方法通过使用更多的句法信息，在一定程度上解决了一些类似方法在其他语言对上取得了效果却在汉英翻译中无效的问题。实验表明，该方法使翻译结果的BLEU值提高了大约0.8~1.2。

关键词：统计机器翻译；句法分析；词对齐；调序

A Syntax-based Pre-reordering for Phrase-based Chinese-English SMT

Wu Qiufeng, Huang Shujian, Dai Xinyu, Chen Jiajun
State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093
E-mail: {wuqf, huangsj, dxy}@nlp.nju.edu.cn; chenjj@nju.edu.cn

Abstract: We propose a pre-reordering method for phrase-based Chinese-English SMT systems in this paper. In our approach, reordering rules are extracted automatically using source side parse trees as well as word alignments. Reordering rules are applied to parse trees of source sentences in both training and testing data to match the syntax of source language more closely to that of target language. SMT systems use sentences generated from reordered parse trees as its input. Experimental results show that this approach gets an improvement of about 0.8~1.2 BLEU scores.

Keywords: statistical machine translation(SMT); parsing; word alignment; reorder

1 引言

互联网的迅速发展极大促进了世界各地人们的交流和信息的获取，但使用语言的不同却给这种交流和信息获取带来极大障碍。面对网络上不断增长的海量数据，传统人工翻译显然已不能满足人们的需要，从而导致对机器翻译需求的不断增加。统计机器翻译[1]是目前机器翻译领域的研究热点，然而现有的统计翻译系统仍有很多不足之处，翻译结果的语序问题便是目前统计翻译系统所面临的主要问题之一。不同语言在语序上往往会有很大的差别，例如，在汉语中，修饰动词的介词短语一般都置于动词短语之前（PP VP），而在英语中，人们则一般将介词短语置于它所修饰的动词短语之后（VP PP）。如果没有一个有效的调序方法，机器翻译的结果往往会很糟。

为了解决汉英翻译中的调序问题，本文提出了一种基于句法的预调序方法。本文余下内容安排如下：第2节介绍调序相关的研究工作；第3节介绍本文提出的预调序方法；第4节为实验结果和分析；最后一节是对本文所做工作的总结以及对将来工作的展望。

2 相关工作

目前，已有不少关于机器翻译中调序的研究工作。一种常见的调序方法是在翻译过程中使用调序模型。较早的调序模型是位变模型[2]，后来又提出了基于句法的调序[3]和基于层次短语的调

* 所属课题：国家自然科学基金（61003112）。

序[4]。尽管翻译过程中的调序模型能够解决部分调序问题，但为了解决更多的调序问题，各种方法或特征的不断加入，会导致翻译模型变得越来越复杂。因此，有研究把部分调序工作放在预处理步骤中进行[5]，预调序可以完全独立于翻译系统运行。

预调序方法一般都使用重写规则来改写源语言句子，使它与目标语言在语序上更加接近。调序规则的产生可以由人工总结[6]也可以从大量语料中自动学习[7~11]。但目前已有的这些预调序方法都有些不足，人工总结调序规则的方法耗时费力，而且还需要深入的语言学知识。自动学习规则的方法一般都是借助于句法树和词（短语）对齐来抽取调序规则，但很多这类方法都是用于汉英以外的语言对上的翻译，用到汉英翻译中效果并不理想，即便部分针对汉英翻译的自动学习规则的预调序方法也存在一定不足，仍有一些问题无法解决，本文第 3 节将详细分析这些问题。

3 调序方法

3.1 现有预调序方法的不足

由于汉语是表义型语言，其句子结构和语序比较灵活，而且汉语中还有很多较长的流水句，汉语的这些特性使得一般的预调序方法在汉英翻译中效果不理想，主要存在如下两个问题：

1. 一般预调序方法抽取出的调序规则中有不少子节点数目较多的长规则，有些长规则明显是错误的，但有些则包含着有用的调序信息。例 1 中的短语只对 PP VP 结构进行调序，但兄弟节点的存在使得从中抽取的规则变为 *NP QP QP PP VP => NP QP QP VP PP*，这一问题最终导致抽取出的规则过于特化和稀疏，质量不高。

   例 1： *其中(NP) 一款(QP) 首次(QP) **在香港(PP)** **燃放(VP)***

   *其中(NP) 一款(QP) 首次(QP) **燃放(VP)** **在香港(PP)***

2. 在汉语中，有些短语的句法结构是一样的，但它们对应的英语翻译语序却不同，如例 2 所示，这就导致如果不对这类短语加以区分，往往会引入大量调序错误。

   例 2： *在 **过去的(DNP)** **三年(QP)** 里*

   *in the **past** **three years***

   *占 **对美出口的(DNP)** **27.2%(QP)***

   *accounting for **27.2 percent** **of the exports to the United States***

3.2 调序规则抽取方法的改进

本文针对一般预调序方法在汉英翻译中的上述两点不足进行了改进，本文使用抽取子规则的方法来解决长规则问题，并通过使用句法树中更多的句法信息来区分汉语结构相同但英语语序不同的短语，下面给出详细的改进方法和改进后完整的预调序步骤。

3.2.1 使用子规则

首先，我们给出调序规则形式的定义：**(规则原始形式，(右部节点调序后的编号序列))**，例如，可以有这样一条规则：(VP → PP VP，(2，1))。下面给出子调序规则的定义和抽取算法：

子调序规则定义：设有一条调序规则 $(X \rightarrow Y_i \cdots Y_j, (k_i\text{-}i+1, \cdots, k_j\text{-}i+1))$ $(i < j)$，该调序规则是长规则 $(X \rightarrow Y_1 Y_2 \cdots Y_n, (k_1, k_2, k_3, \cdots, k_n))$ 的子规则当且仅当 i, j, k_i, k_j 同时满足如下 4 个条件：

① $k_i > i$;

② $k_j < j$;

③ $\forall p < i, p \in \{1,2,\ldots,n\}, k_p < \min\{k_i, \ldots, k_j\}$;

④ $\forall q > j, q \in \{1,2,\ldots,n\}, k_q > \max\{k_i, \ldots, k_j\}$.

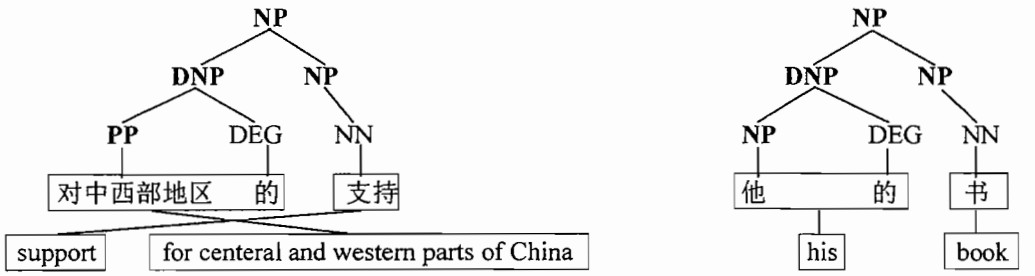
举个例子，有一条较长的调序规则：*(X → Y_1 Y_2 **Y_3 Y_4 Y_5** Y_6 **Y_7 Y_8** Y_9，(1,2,**4,5,3**,6,**8**,7,9))*，我们可以抽出两条较短的调序规则：*(X → Y_3 Y_4 Y_5，(2，3，1))* 和 *(X → Y_7 Y_8，(2，1))*。

子调序规则抽取算法（算法中的 A 和 B 分别为调序规则右部节点调序前、后的序列）：

```
INPUT: A[1:n] = { 1, 2, 3,..., n }
       B[1:n] = { k1, k2, k3,..., kn }
       CandidateRuleSet = ϕ
i ← 1
WHILE i <= n
  IF B[i] != A[i]
  THEN start ← i; end ← B[i]; i ← i + 1
       WHILE  i <= end
            IF B[i] > end
            THEN  end ← B[i]
            ELSE  i← i + 1
       产生新调序规则X → Y_B[start] ··· Y_B[end]
       将新规则加入 CandidateRuleSet
  ELSE  i ← i + 1
OUTPUT: CandidateRuleSet
```

3.2.2 使用子节点信息

汉语中有很多结构形式为“*某某的(DNP) 某某(NP)*”的名词短语，其中的 DNP 短语的结构不一样时，这些短语对应英文翻译的语序往往也不一样，图 1 显示了如何通过 DNP 节点的子节点区分两类结构相同的不同名词短语。在句法树中大多数内部节点都不止一个子节点，本文的方法中，主要使用节点的中心子节点[12]。



(a) NP → DNP(PP) NP(NN)　　(b) NP → DNP(NP) NP(NN)

图 1　两类不同的 NP → DNP NP 短语

3.2.3 使用父节点信息

父节点对于区分不同类型的短语同样十分有效。我们发现，在所用的训练数据中，有很多形如“*在 过去的(DNP) 三年(QP) 里(in the **past three years**)*”的 *QP → DNP QP* 短语都被规则 *QP → DNP QP => QP → QP DNP* 进行调序。通过分析比较，我们发现上述类型的 QP 短语一般都出现在 LCP 节点下，而需要调序的 QP 短语则往往位于其他类型的节点下。图 2 显示了如何通过父节点区分两类结构相同英文翻译语序不同的 QP 短语。

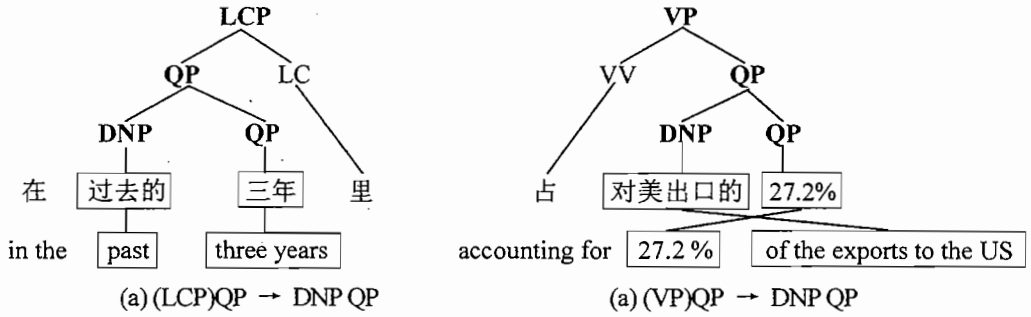


图2 两类不同的QP → DNP QP 短语

3.2.4 完整调序步骤

（1）给句法树中的每个内部节点加上子节点和父节点标记；

（2）自顶向下，遍历句法树中的所有非叶节点，将子节点数目大于 1 的节点与它的子节点组成一条规则，如图 3 所示的句法树节点可以得到这样一条规则：$X \rightarrow Y_1Y_2\cdots Y_n$；

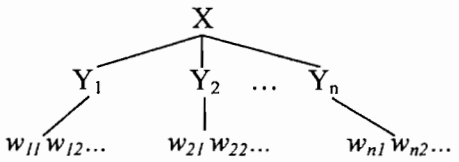


图3 “$w_{11}\,w_{12}\cdots\,w_{21}\,w_{22}\,\cdots w_{n1}\,w_{n2}\cdots$”的句法树

（3）将步骤（2）得到的规则 $X \rightarrow Y_1Y_2\cdots Y_n$ 右部的子节点重新排序。具体算法为：

对规则右部的每一个节点 Y_i，列出 Y_i 下面的所有叶节点 w_{i1}，w_{i2}，…，w_{im}，根据词对齐求出在目标语言中与这些词对应的词的位置 $a(w_{i1})$，$a(w_{i2})$，…，$a(w_{im})$，如果与 w_{ij} 对应的目标语言的词不止一个，那么就用这些词在目标语言中的平均位置作为 $a(w_{ij})$ 的值，那么 Y_i 的平均位置 $position_i$ 为：$position_i = \dfrac{\Sigma_j a(w_{ij})}{Y_i\text{节点下源语言词的个数}}$。最后，根据 $position_i$ 的大小对规则右部重新排序，得到一条新规则 $X \rightarrow Y_{k1}Y_{k2}\ldots Y_{kn}$；

（4）将步骤(3)的规则与步骤(2)中的规则原始形式比较，若不一致，则得到一个调序规则。仍以图 3 为例，我们可以得到这样一个调序规则：$(X \rightarrow Y_1Y_2\cdots Y_n,\ (k_1k_2,\ \cdots,\ k_n))$；

（5）从步骤(4)抽取出的规则中使用 3.2.1 节的算法抽取子调序规则；

（6）筛选步骤(5)抽取出的调序规则。我们定义了两个阈值作为筛选条件，一个称为数量阈值，另一个称为比例阈值。以规则 $(X \rightarrow Y_1Y_2\cdots Y_n,\ (k_1k_2,\ \cdots,\ k_n))$ 为例，该规则只有满足调序序列 $X \rightarrow Y_{k1}Y_{k2}\cdots Y_{kn}$ 出现的次数超过数量阈值并且该规则在所有原始形式为 $X \rightarrow Y_1Y_2\cdots Y_n$ 的规则中的比例超过比例阈值才算是一条有效调序规则；

（7）用抽取出的调序规则对训练和测试语料的源语言句法分析树进行调序，重新生成源语言句子。将调序后得到的新训练和测试语料作为翻译系统的输入。

4 实验与分析

4.1 实验系统与数据

实验所用的翻译系统基于开源的 MOSES 系统。预处理中使用的句法分析工具为使用概率上

下文无关文法的 Berkeley Parser，词对齐工具为 GIZA++。实验所用训练数据主要由 LDC2002~2006 的部分汉英语料组成，大约 50 万句中英文句对。测试数据是 NIST 评测语料，NIST06 为开发集，其余为测试集。训练和测试数据的详细统计信息如表 1 所示。实验中筛选调序规则的数量阈值为 100，比例阈值为 0.5。最后，总共得到 50 多条调序规则。

表 1　训练和测试数据详细信息

语料	句子数	单词数	句子平均长度
训练数据	456986	12417050（汉）14264896（英）	27（汉）31（英）
开发集（nist06）	1664	39004	23
测试集（nist03）	919	24812	27
测试集（nist04）	1788	50061	28
测试集（nist05）	1082	30440	28
测试集（nist08）	1357	33208	24

4.2　实验结果及分析

表 2 给出的是没有预调序、使用一般预调序方法以及使用本文的预调序方法调序的翻译结果的 BLEU 值，可以看出，一般预调序方法在汉英翻译中几乎没有效果，而本文的调序方法使得翻译的 BLEU 值提高了 1 点左右，这表明本文的预调序方法对汉英统计机器翻译系统而言是有效的。然而，本文的方法仍有一定的不足。如表 3 所示，在所有节点中，只有大约 5~6%的节点被调序，而且从表 4 可以看出，调序的节点中仍有大约 33.14%的错误，同时还有近 42.03%的需要调序的节点没有被调序。这些问题主要是由句法分析和词对齐的错误造成的，这些错误对调序规则的抽取带来干扰，句法分析的错误也使得部分不需要调序的节点被调序。另外，在使用子节点时，只有部分中心子节点对区分不同短语所起的效果比较明显，而那些没能发挥作用的中心子节点的存在反而增加了规则的稀疏性。

表 2　使用预调序前后的 BLEU 值对比

	nist06	nist03	nist04	nist05	nist08
没有预调序	33.48	33.76	33.71	33.08	25.10
一般调序方法	33.68(+0.20)	33.68(-0.08)	33.37(-0.34)	32.82(-0.26)	25.23(+0.13)
本文调序方法	34.71(+1.23)	34.58(+0.82)	34.58(+0.87)	33.92(+0.84)	26.05(+0.95)

表 3　训练和测试语料中的调序节点数

	训练数据	nist06	nist03	nist04	nist05	nist08
调序节点数	860586	2549	1484	3476	1908	1939
节点总数	13153618	41599	25879	53174	31444	36182

表 4　调序的准确率与召回率

准确率	召回率
66.86%	57.97%

5　结束语

本文提出了一种用于基于短语的汉英翻译系统的预调序方法。该方法使用自动抽取的调序规则对汉语句法树进行调整，使得汉语句子的语序更接近英语句子。该方法在汉英翻译系统上使翻译结果的 BLEU 值提高了大约 0.8~1.2。然而，该方法仍有一些有待提高之处。调序中的一些错误

可能会使得训练数据出现一定程度的不一致性，从而带来负面影响，在今后工作中，我们将尝试找出这些错误会带来怎样的影响。同时，还将设法找到一个能够更好利用子节点信息的方法，并尝试使用更多的句法信息以提高调序的准确率和召回率。此外，我们还将在更大的数据集和其他语言对上试验我们的方法。

参 考 文 献

[1] 刘群. 统计机器翻译综述. 中文信息学报, 2003: 1-12.

[2] Peter F. Brown, John Cocke, Stephen A. Della Pietra, 等. A Statistical Approach to Machine Translation. Computational Linguistics, 16(2): 79-85.

[3] Kenji Yamada and Kevin Knight. A Syntax-based Statistical Translation Model. ACL, 2001: 523-530.

[4] David Chiang. A Hierarchical Phrase-based Model for Statistical Machine Translation. ACL, 2005: 263-270.

[5] Simon Zwarts and Mark Dras. Syntax-Based Word Reordering in Phrase-Based Statistical Machine Translation: Why Does it Work? Proceedings of MT Summit. 2007.

[6] Chao Wang, Michael Collins and Philipp Koehn. Chinese Syntactic Reordering for Statistical Machine Translation. Joint Conference on EMNLP and CoNLL, 2007: 737-745.

[7] Fei Xia and Michael McCord. Improving a Statistical MT System with Automatically Learned Rewrite Patterns. COLING 2004.

[8] Dmitriy Genzel. Automatically Learning Source-side Reordering Rules for Large Scale Machine Translation. COLING 2010: 376-384.

[9] Karthik Visweswariah, Jiri Navratil, Jeffrey Sorensen, 等. Syntax Based Reordering with Automatically Derived Rules for Improved Statistical Machine Translation. COLING 2010: 1119-1127.

[10] Yuqi Zhang, Richard Zens and Hermann Ney. Chunk-Level Reordering of Source Language Sentences with Automatically Learned Rules for Statistical Machine Translation. SSST'07 Proceedings of the NAACL-HLT 2007/AMTA Workshop on Syntax and Structure in Statistical Translation, 2007: 1-8.

[11] Chi-Ho Li, Dongdong Zhang, Mu Li 等. A Probabilistic Approach to Syntax-based Reordering for Statistical Machine Translation. ACL, 2007: 720-727.

[12] Michael Collins. Head-Driven Statistical Models for Natural Language Parsing. Ph.D. thesis, University of Pennsylvania, 1999.