面向统计机器翻译的重对齐方法研究*

肖桐 李天宁 陈如山 朱靖波 王会珍

东北大学 信息学院 计算机软件研究所 自然语言处理实验室，沈阳 110004

E-mail: {xiaotong,zhujingbo,wanghuizhen}@mail.neu.edu.cn, {litn,chenrs}@ics.neu.edu.cn

摘 要： 词对齐是统计机器翻译中的重要技术之一。本文提出了一种重对齐方法，它在 IBM models 获得的正反双向词对齐的基础上，确定出正反双向对齐不一致的部分。之后，对双向词对齐不一致的部分进行重新对齐以得到更好的对称化的词对齐结果。此外，本文提出的方法还可以利用大规模单语语料来强化对齐结果。实验结果表明，相比在统计机器翻译中广泛使用的基于启发信息的词对齐对称化方法，文本提出的方法可以使统计机器翻译系统得到更高的翻译准确率。

关键词： 统计机器翻译，词对齐，重对齐，IBM models

Word Realignment for Statistical Machine Translation

Xiao Tong, Li Tianning, Chen Rushan, Zhu Jingbo and Wang Huizhen

NLP Lab, Institute of Computer Software, Northeastern University, Shenyang, 110004

E-mail: {xiaotong,zhujingbo,wanghuizhen}@mail.neu.edu.cn, {litn,chenrs}@ics.neu.edu.cn

Abstract: Word alignment is one of the most important techniques in statistical machine translation (SMT). In this paper, we propose a word realignment method, which recognizes the inconsistent parts between the bidirectional alignments generated by IBM models at first. Then, the word alignment is refined by realigning the inconsistent parts. To reinforce our method, a monolingual feature is used to make benefits from large scale monolingual corpus. The effectiveness of the method is demonstrated on a state-of-the-art phrase-based SMT system. The experimental results show that compared to the wildly-adopted heuristics-based method our method can obtain higher translation accuracy.

Keywords: statistical machine translation, word alignment, word realignment, IBM models.

1 引言

词对齐是统计机器翻译的重要组成部分[1]。高质量的词对齐结果可以带来统计机器翻译系统翻译性能的提高[2][3]。现在大多数统计机器翻译系统都是利用 IBM models [4]来进行词对齐。但是在 IBM models 中一个源语言单词最多只允许对应一个目标语单词。因此 IBM models 生成的词对齐通常也被称作非对称的词对齐。为了获得对称的词对齐，需要利用 IBM models 得到源语言→目标语言（正向）和目标语言→源语言（反向）双向对齐的结果，之后在双向对齐的基础上自动得到对称的词对齐结果。在这个过程中，一个关键的问题就是解决双向对齐的不一致性。图 1 展示了一个由 IBM models 生成的双向词对齐的实例。在这个实例中，源语言单词 f_1 和目标语单词 e_1 在正反双向词对齐中均被对应上，于是我们称双向词对齐在 f_1 和 e_1 之间的对齐上是相交的或一致的。相反，f_1 和 e_2 在反向词对齐中被对应上，而在正向词对齐中却没有被对应上。这时我们称双向词对齐在 f_1 和 e_2 之间的对齐上是有歧义的或不一致的。对于这种情况，我们需要判断 f_1 和 e_2 是否在最终的词对齐结果中被对齐。在本文中，我们称片断对(f_1, e_1e_2)有相交型歧义，而(f_1, e_1e_2)被称为相交型歧义块。

可以看出，提高相交型歧义块中的词对齐的准确率将会有助于最终词对齐准确率的提高。对于这个问题，现在广泛采用的解决办法是，利用启发信息来判断有歧义对齐的正确性[1][2]。但

*本论文工作得到国家自然科学基金项目(60873091)、辽宁省自然科学基金项目(20072032)和沈阳市科学技术计划(1081235-1-00)资助。

是由于这个方法只考虑了锚点（比如：双向对齐的交集部分）和对齐点的相对位置信息，它更适用于翻译顺序相对一致的语言对，如：法语-英语。而在语序差异极大的语言之间，经常会出现翻译的远距离调序现象，比如：汉语-英语之间的远距离调续。这时仅使用启发信息并不能得到很好的对齐结果。针对这个问题，本文提出了一个对相交型歧义块进行重新对齐的方法。它使用了翻译概率，扭曲度概率和产出率概率等多个特征共同作用来得到对称化地词对齐结果。此外，本文还对这个方法进行了改进，使它能够利用大规模单语语料得到更好的词对齐结果。

为了检验本文提出的方法的有效性，我们把它应用到基于短语的统计机器翻译系统中。在汉-英翻译任务上的实验结果表明，本文提出方法要优于现在广泛使用的基于启发信息的方法。此外，我们的实验结果还表明使用单语语料有助于进一步提高词对齐的性能。

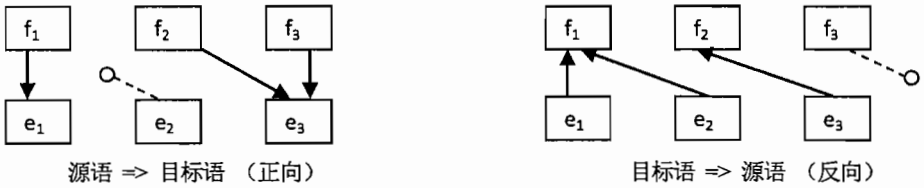

图1 正反双向词对齐实例

2 问题描述

2.1 词对齐形式化描述

假设 $\boldsymbol{f}=f_1 f_2 \ldots f_J$ 和 $\boldsymbol{e}=e_1 e_2 \ldots e_I$ 分别表示源语和目标语词序列，其中 J 和 I 表示序列长度。$\boldsymbol{f}$ 和 $\boldsymbol{e}$ 之间的词对齐可以表示为一个函数 $a: J \times I \Rightarrow (0,1)$，对于任意(j, i) ($1 \le j \le J \wedge 1 \le i \le I$)有：当 f_j 和 e_i 之间有对齐关系，$a(j,i)=1$；否则 $a(j,i)=0$。在本文中我们称 a(j, i)为对齐函数。对于任意一个(j, i)，如果 a(j, i) = 1，我们称(j, i)为一个对齐链接。如果有两个对齐函数 a_1 和 a_2 满足对任意的(j, i)，都有 $a_1(j,i) \le a_2(j,i)$，我们称 $a_1 \subseteq a_2$[1]。考虑词对齐的方向，我们使用 $a_{f\to e}$(j, i)表示从 $\boldsymbol{f}$ 到 $\boldsymbol{e}$ 的词对齐函数（正向词对齐），$a_{e\to f}$(j, i)表示从 $\boldsymbol{e}$ 到 $\boldsymbol{f}$ 的词对齐函数（反向词对齐）。在此基础上，我们定义正反双向词对齐的并集为 a_{union}，它满足 $a_{union}(j,i)=1$ iff $a_{f\to e}(j,i)=1 \vee a_{e\to f}(j,i)=1$；正反双向词对齐的交集为 a_{inter}，它满足 $a_{inter}(j,i)=1$ iff $a_{f\to e}(j,i)=1 \wedge a_{e\to f}(j,i)=1$。

2.2 相交型歧义块定义

在给出相交型歧义块的定义之前，我们先给出块（*block*）的定义。假设 $f_{j1}\ldots f_{j2}$ 和 $e_{i1}\ldots e_{i2}$ 分别是 $\boldsymbol{f}$ 和 $\boldsymbol{e}$ 中的两个词序列，a_{union} 为($\boldsymbol{f}$, $\boldsymbol{e}$)之间正反双向词对齐的并集，如果

$$\sum_{(j<j1 \vee j>j2) \wedge (i1 \le i \wedge i \le i2)} a_{union}(j,i) + \sum_{(j1 \le j \wedge j \le j2) \wedge (i<i1 \vee i>i2)} a_{union}(j,i) = 0 \qquad (1)$$

我们称序偶($f_{j1}\ldots f_{j2}$, $e_{i1}\ldots e_{i2}$)是一个翻译对或块，并把它表示成 B($f_{j1}\ldots f_{j2}$, $e_{i1}\ldots e_{i2}$)。这个定义意味着($f_{j1}\ldots f_{j2}$, $e_{i1}\ldots e_{i2}$)中没有词被对齐到($f_{j1}\ldots f_{j2}$, $e_{i1}\ldots e_{i2}$)以外。可以看出这个定义本质上与基于短语的统计机器翻译[1]中的短语对的定义是一致的。

给定正反双向词对齐 $a_{f\to e}$，$a_{e\to f}$和一个块 B($f_{j1}\ldots f_{j2}$, $e_{i1}\ldots e_{i2}$)，如果有一个(j, i) ($j_1 \le j \le j_2 \wedge i_1 \le i \le i_2$)满足 $a_{f\to e}(j,i)=1 \wedge a_{e\to f}(j,i)=1$，我们称 $a_{f\to e}$(j, i)和 $a_{e\to f}$(j, i)在(j, i)上是相交的或一致的。如果 $a_{f\to e}(j,i) \neq a_{e\to f}(j,i)$，我们称 f_j 和 e_i 之间的对齐链接有（相交型）歧义。如果 B($f_{j1}\ldots f_{j2}$, $e_{i1}\ldots e_{i2}$)包含歧义链接，而且 B($f_{j1}\ldots f_{j2}$, $e_{i1}\ldots e_{i2}$)中没有其它块包含歧义链接，我们就称 B($f_{j1}\ldots f_{j2}$,

[1]这里我们借用了集合的饱含关系的表示方法。如果把一个对齐函数中满足 a(j, i) = 1 的所有(j, i)看成一个集合，这种表示也是正确的。

$e_{i1}...e_{i2}$)为相交型歧义块(*overlapping ambiguous block*, *OAB*)，并把它记为 OAB($f_{j1}...f_{j2}$, $e_{i1}...e_{i2}$)。实际上，OAB 的定义保证了任意一个 OAB 不能嵌套的包含其它 OAB，而且与其它任何 OAB 都不相交。例如，在图 1 中所示的对齐实例中包含两个 OAB：OAB(f_1, e_1e_2)和 OAB(f_2f_3, e_3)。

2.3 重对齐

在本文中，重对齐是指根据正反双向词对齐的结果重新对双语句对进行对齐，以得到完整的对称化的词对齐结果。显然，对于双语句对中的非 OAB 的部分来说来说重对齐是比较简单的，因为我们只需把双向词对齐的交集部分作为最终的对齐结果即可。而对于 OAB 来说，重对齐任务要难得多，因为我们要对其中的每个歧义链接进行消歧。而本文的工作也正是集中在对 OAB 的重对齐任务上。在这个任务中有如下两个问题需要解决：

a) 给定双语句对(***f***, ***e***)和它们之间的正反双向词对齐结果，如何得到所有的 OAB。

b) 如何定义 OAB 上最优的词对齐，如何高效地搜索最优词对齐。

我们把第一个问题称作相交歧义块识别问题，把第二个问题称作 OAB 的重对齐问题。

3 相交歧义块识别

根据 OAB 的定义，我们给出了一个能够得到所有 OAB 的 *OAB 快速识别算法*。描述如下：

输入: 双语句对(***f***, ***e***) 和正反双向词对齐结果 $a_{f\to e}$ 和 $a_{e\to f}$
输出: (***f***, ***e***)包含的所有 OAB

```
Step1: 得到所有歧义链接，把它们保存在 ambilink[0...l]中
Step2: for k = 0 to l - 1 do
        if Checked(f[ambilink[k].j]) do next
            FSegStart = FSegEnd = ambilink[k].j
            ESegStart = ESegEnd = ambilink[k].i
            FoundOABFlag = false
            while not FoundOABFlag do
                Step2.1: 根据 a_{e->f}，ESegStart，ESegEnd 和 e 更新 FSegStart 和 FSegEnd
                Step2.2: 根据 a_{f->e}，FSegStart，FSegEnd 和 f 更新 ESegStart 和 ESegEnd
                Step2.3: 如果无更新，把 FoundOABFlag 设为 true
            把 B(FSegStart, FSegEnd, ESegStart, ESegEnd)存入 OABList
            把 f[FSegStart...FSegEnd] 标记为"checked"
Step3: for i = 0 to OABList.length – 1 do
        if OABList[i]没有被 OABList 中的其它元素覆盖 do 输出 OABList[i]
```

这个算法的核心思想是，根据每个歧义链接进行扩展，直到得到包含它的 OAB。算法中的 Step2.1 和 Step2.2 实际上就是对当前得到的含有歧义链接的块进行判断，如果在这个块的外部仍有歧义链接对应到块中的某些词(源语词或者目标语词)就更新块的范围使其包含这个歧义链接。当这个块无法被更新时，表示得到 OAB，退出循环。这个算法的时间复杂度为$\Theta(I \cdot J)$。相比最直接的遍历方法（时间复杂度为$\Theta(I^2 \cdot J^2)$），它具有更高的运行效率。

4 重对齐模型

4.1 模型 1

首先，为了简化 OAB 的重对齐问题，我们假设：

a) OAB 中的词对齐是上下文无关的。对于一个 OAB，其它 OAB 不会影响它的对齐结果。

b) OAB 中的词对齐a与双向词对齐的并集 a_{union} 是兼容的，即 $a \subseteq a_{union}$。

根据这两个假设，我们定义 OAB($f_{j1}...f_{j2}$, $e_{i1}...e_{i2}$)上最优的词对齐为：

$$a_{best} = max\, arg_{a \subseteq a_{union}}\ Score(a, OAB) \tag{2}$$

其中 *Score*(a, OAB)是一个函数用来评价对齐 a 的好坏程度。由于直接在整体上对 OAB 中的词对齐进行评价是比较困难的，我们把 *Score*(a, OAB)定义为如下形式：

$$Score(a, OAB) = \prod_{k=1...K} Score\left(factor_k(a, OAB)\right) \quad (3)$$

其中{$factor_1$(a, OAB), … , $factor_K$(a, OAB)}表示影响词对齐的各个因素的集合。沿用经典的 IBM models[4]的思路，本文定义了三个影响 OAB 中的词对齐因素，它们是：翻译概率（translation equivalent probabilities），扭曲度概率（distortion probabilities）和产出率概率（fertility probabilities）。于是我们得到，

$$Score(a, OAB) = Score\left(link(a, OAB)\right) \times Score\left(fertility(a, OAB)\right) \times Score\left(distortion(a, OAB)\right) \quad (4)$$

其中，

$$Score\left(link(a, OAB)\right) = \prod_{(j,i)\in OAB \wedge a(j,i)=1} t\left(f_j, e_i\right) \times \prod_{(j,i)\in OAB \wedge a(j,i)\neq 1} \left(1 - t\left(f_j, e_i\right)\right) \quad (5)$$

$$Score\left(fertility(a, OAB)\right) = \prod_{j=j1}^{j2} n(\emptyset_{f_j} | f_j) \times \prod_{i=i1}^{i2} n(\emptyset_{e_i} | e_i) \quad (6)$$

$$Score\left(distortion(a, OAB)\right) = \prod_{(j,i)\in OAB \wedge a(j,i)=1} d(j, i) \quad (7)$$

这里 *Score*(link(a, OAB))表示整个词对齐所对应的翻译概率，$t(f_i, e_i)$表示 f_j和 e_i互为翻译的联合概率。*Score*(fertility(a, OAB))表示 OAB 生成一定数量链接的概率，这里$n(\emptyset_w | w)$表示一个单词 w 对应$\emptyset_w$条链接的概率。*Score*(distortion(a, OAB))表示词对齐所对应的整体调序概率，这里 d(j, i)表示源语言第 j 个词与目标语上第 i 个词之间有链接的概率。

对于模型的参数估计，我们直接使用 IBM model 3 得到$n(\emptyset_w | w)$的估计。而对于$t(f_i, e_i)$，我们在训练语料的 a_{union} 上，使用极大似然估计(Maximum likelihood estimation , MLE)的方法对其进行估计，即：$t(f_i, e_i)$=count(f_i和e_i之间有链接)/count(f_i和e_i共现)。对于 d(j, i)，我们采用了一个简单的估计方法（或者说定义），$d(j, i) = \alpha^{|i - I \cdot j/J|}$。这里$|i - I \cdot j/J|$表示在对齐矩阵中(j, i)与对角线的相对距离，距离越远表示调序的程度越大。$\alpha < 1.0$ 是调解因子，在本文中我们通过实验的方法得到 α 的最优值 0.9。

4.2 模型 2

在语言的使用中，我们常常会用多个连续的词来表达一个概念，比如汉语和英语中的名词短语。如果一个连续的词序列频繁地共现，那它们很有可能在集中描述一个概念，在对齐中被作为一个单元的可能性就越大。比如，如果源语言句子中的某个词序列中的每个词都对应到目标语句子的相同部分，这个词序列就应该构成一个对齐单元。如果我们能很好的度量一个词或词序列表达同一个概念的可能性大小，那么这个信息就可以帮助我们得到更好的词对齐结果。基于这个想法，我们在模型 1 的基础上引进了一个新的单语特征，用它来度量在对齐中每个对齐单元的好坏程度。定义如下：

$$Score_{model\ 2}(a, OAB) = Score_{model\ 1}(a, OAB) \times Socre\left(mono(a, OAB)\right) \quad (8)$$

其中，

$$Score\left(mono(a, OAB)\right) = \prod_{j\in[j1,j2]\wedge|a(f_j)|>0} m\left(a(f_j)\right) \times \prod_{i\in[i1,i2]\wedge|a(e_i)|>0} m\left(a(e_i)\right) \times$$

$$\prod_{j\in[j1,j2]\wedge|a(f_j)|=0} m(f_j) \times \prod_{i\in[i1,i2]\wedge|a(e_i)|=0} m(e_i) \quad (9)$$

这里 a(w)表示所有与 w 有对齐关系的词的集合，|a(w)|表示与 w 有对齐关系的词的数量。m(a(w))是对 a(w)作为一个对齐单元可靠性的度量。本文中，我们把 m(a(w))定义为(以 m(a(f_j))作为实例，对于 m(a(e_i))可以同理推得)，

$$m\left(a(f_j)\right) = \begin{cases} Pr(e_p \dots e_q) + 1, & a(f_j)\text{对应一个序列}e_p \dots e_q \\ \prod_{e_k \in a(f_j)} Pr(e_k) + 1, & otherwise \end{cases} \quad (10)$$

这里 $Pr(e_k)$和 $Pr(e_p..e_q)$表示 e_k 和 $e_p...e_q$ 在单语（目标语）中出现的概率，它们可以直接通过 MLE 在单语语料上进行估计。

4.3 搜索

对于一个 OAB，如果它包含 l 个 a_{union} 的链接，那么搜索最佳对齐的搜索空间为 2^l。对于大部分 OAB 来说，l 都是一个比较小的值，这时我们可以直接使用全搜索的方法来得到 a_{best}。而对于 l 比较大的情况（$l > l_{max}$），我们使用了一个基于栈的 decoder 来搜索 a_{best}。它联合使用了翻译概率和扭曲度概率作为启发函数来对 a_{best} 进行搜索。这里 l_{max} 是一个阈值，我们用它来限定需要剪枝处理的链接数下限。

此外，还可以利用 a_{inter} 来进一步缩小搜索空间。通常 a_{inter} 包含的都是准确的词对齐，因此可以把它作为对齐锚点。这样，在提高搜索效率的同时，可能会进一步提高性能。利用 a_{inter} 作为锚点后搜索空间可以被进一步限制在$a_{inter} \subseteq a \subseteq a_{union}$ 范围内。

5 实验

5.1 测试方法及实验用数据

我们把本文提出的方法应用到实际的汉-英统计机器翻译系统中来验证它的有效性，并使用大小写不敏感的 BLUE4 做为翻译质量的评价指标。

我们训练和测试数据是 SSMT2007 官方提供汉-英机器翻译任务用数据，包括：训练语料约 80 万句对，开发集 500 句（每句 4 个参考译文），测试集 1002 句（每句 4 个参考译文）。在使用前，我们首先用 NEU NLP Lab 所开发的中文分词工具[2]对中文句子进行分词，并用一个基于规则的 tokenizer 对英文句子进行切分，此外我们还去掉了英文单词的大小写信息。我们使用了部分 LDC 提供的语料作为训练重对齐模型 2 所使用的源语和目标语的单语语料，包括大约 180 万句的中文和 180 万句英文单语语料。

5.2 基准系统

本文采用基于短语的统计机器翻译系统 moses[3]作为实验的基准系统。其中，我们用基于 IBM models 的 GIZA++[4]获得了正反双向词对齐结果。为了对比本文提出的方法，我们采用了三种基于启发信息的词对齐对称化方法“intersection”，“union” 和“refined”作为从正反双向词对齐获得对称的词对齐的 baseline 方法[2]。这里“intersection”，“union” 和“refined”分别是指：双向对齐的交集，双向对齐的并集，在双向对齐的交集的基础上利用启发信息进行扩展并考虑对齐矩阵中对角线位置的信息这三种方法[5]。此外我们利用 SRLIM 工具在实验用的英文单语语料上训练了 5-gram 语言模型。对于短语抽取和 decoder，我们都使用了 moses 工具包所提供的程序，并采用缺省设置。此外，我们使用了最小错误率训练来对参数进行优化。

5.3 Baseline vs. 重对齐

在进行重对齐之前，我们首先进行了双向词对齐，之后识别出了训练语料中所有的 OAB。

---

[2] http://www.nlplab.com/

[3] http://www.statmt.org/moses/

[4] http://www.fjoch.com/GIZA++.html

[5]“refined”方法是 moses 工具包所使用的缺省方法。它也被称作“intersect-diag-grow”方法。

我们发现平均每个句对包含 1.65 个 OAB。这表明 OAB 在汉-英机器翻译中的词对齐任务中是很常见的。此外，绝大多数的 OAB（>80%）包含的连接数小于等于 15。因此，在随后的所有实验中我们均设置 $l_{max}=15$ 来对包含链接数大于 15 的搜索进行剪枝。

在识别 OAB 之后我们分别利用本文提出的重对齐模型 1 和模型 2 进行了重对齐，并把得到的结果用于基准统计机器翻译系统中。在实验中，我们并没有利用锚点 a_{inter} 对搜索进行剪枝。实验结果如表 1 所示。可以看出在三种 baseline 方法中，"refined"方法取得了最好的性能，其次是"union"方法。不过，"intersection"方法却取得了比前两种方法差很多的性能。这主要是由于，"intersection"方法会产生非常稀疏的对齐结果，这会导致短语表中噪声的增加，并最终降低翻译质量。相比 baseline 方法，本文提出的方法得到了更高的 BLUE 值。重对齐模型 1 和模型 2 比最高的 baseline 方法分别高出 0.59 和 0.68 个点。这说明了我们方法的有效性。此外，模型 2 比模型 1 取得了更好的性能，这也说明了使用单语语料也可以进一步改善词对齐的质量，并间接提高统计机器翻译系统的性能。

方法	BLUE4(%)
Baseline1 ("intersection")	17.15
Baseline2 ("union")	20.66
Baseline3 ("refined")	21.00
重对齐模型 1	21.59
重对齐模型 2	21.68

表 1 Baseline 及重对齐模型性能

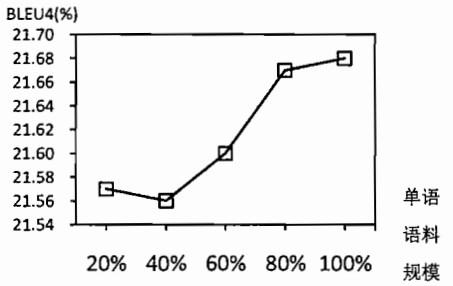

图 2 单语训练语料规模对模型 2 性能的影响

方法	BLUE4(%)
模型 1("a_{inter} anchoring")	21.68
模型 2("a_{inter} anchoring")	21.66

表 2 使用锚点信息后的性能

5.4 锚点信息的使用

根据 4.3 节的论述，我们可以使用 a_{inter} 作为锚点来缩小搜索空间。表 2 给出了模型 1 和模型 2 使用锚点信息后的翻译性能。有趣的是，对于模型 1，锚点信息的使用带来了翻译性能的进一步提高（相比 5.3 节的实验结果又提高了 0.09 个点）。而模型 2 在使用锚点信息之后翻译性能却下降了 0.02 个点。这个实验结果说明，在 OAB 中，模型的最优解 a_{best} 不一定总包含 a_{inter}。虽然在整体上 a_{inter} 中的对齐准确率较高，但它并不一定能带来 OAB 重对齐性能的提高。

5.5 单语语料规模大小对性能的影响

在最后一组实验中，我们对单语语料规模大小对重对齐模型 2 的性能的影响进行了研究。我们分别用 20%，40%，60%，80%和 100%的单语语料训练，得到了模型 2 的对齐结果。实验结果如图 3 所示。可以看出，除了在 40%时性能略有下降外，翻译性能基本上是随着单语语料数量的增大而提高。这说明增大单语训练语料的规模有助于提高模型 2 的性能。从图 2 中我们还可以发现，当单语语料规模达到一定大小后（比如我们实验用单语语料的 80%）翻译性能增长缓慢。这表明，在我们的方法中，单纯地增大单语语料规模并不能有效地提高翻译翻译准确性。

6 相关工作

Koehn 等人[1] 以及 Och 和 Ney[2]研究了利用双向非对称的词对齐得到对称的词对齐的方法。在他们的方法中，首先把正反双向词对齐的交集部分固定，之后利用启发信息来扩展固定部分。但是这种基于启发信息的方法更倾向于含有局部调序的对齐，而对于语序相差很大的语言间的对

齐的性能并不是很好。Liang 等人[5]提出了一种利用最大化正反双向对齐的一致部分的似然概率的方法来得到更好的词对齐结果。与 Liang 等人工作不同，我们的工作集中在正反双向对齐不一致的部分。也就是说我们更关心重新对齐那些正反双向训练下 IBM modes 不能达成一致的对齐。

还有其它一些工作主要集中在利用判别模型来进行词对齐[6][7][8][9]。他们把词对齐转化为有指导或半指导的分类任务，并利用多个特征共同作用得到对齐结果。不过，这些方法均需要人工标注的词对齐的训练语料，训练数据的构造代价比较昂贵。

此外，本文工作与其它相关工作的另一个重要不同是，我们提出的重对齐方法可以利用单语语料来进一步提高性能。而这个问题在以前的工作中并没有被很好地讨论过。

7 讨论

在模型的参数估计方面，本文分别对不同的参数采用了不同的估计方法。这么做的好处在于，方法简单，而且系统易于实现。实际上，也可以考虑利用 EM 等无指导的学习方法，来最大化词对齐在整个训练集上的似然概率（可以把目标函数看做词对齐的可能性的度量），同时得到更好的参数估计结果。

此外，在本文提出的模型中，所有特征的权重都是相等的（均为 1）。实际上，我们可以通过调整特征的权重使模型取得更好的性能。不过，这需要使用少量带有人工标注词对齐的开发集来优化这些权重。

8 结论及未来工作

本文提出了一种重对齐方法，它在 IBM models 生成正反双向词对齐的基础上，对双向对齐有歧义的部分进行重新对齐，最终得到完整的对称的词对齐结果。此外，这个方法可以利用单语语料来进一步改进词对齐结果。相比在统计机器翻译中广泛使用的基于启发信息的词对齐对称化方法，文本提出方法可以使统计机器翻译系统得到更高的翻译准确率。在以后的工作中，我们会对重对齐模型的参数估计和模型最优解的搜索等问题做进一步研究。

参 考 文 献

[1] Philipp Koehn, Franz Josef Och ,and Daniel Marcu. Statistical Phrase-Based Translation. In *Proc. of HLT/NAACL*, 2003, 48~54.

[2] Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 2003, 29(1):19~51.

[3] Alexander Fraer and Daniel Marcu. Measuring word alignment quality for statistical machine translation. In Technical Report ISI-TR-616, 2006, ISI/University of Southern California.

[4] Peter F. Brown, Stephen A. Della Piatra, Vincent J. Della Pietra, and R. L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 1993, 19(2):263~311.

[5] Percy Liang, Ben Taskar, and Dan Klein. Alignment by agreement. In *Proc. of HLT/NAACL*, 2006, 104~111.

[6] Yang Liu, Qun Liu, and Shouxin Lin. Log-linear models for word alignment. In *Proc. of ACL*, 2005, 459~466.

[7] Alexander Fraer and Daniel Marcu. Semi-Supervised Training for Statistical Word Alignment. In *Proc. of ACL*, 2006, 769~776.

[8] Abraham Ittycheriah and Salim Roukos. A maximum entropoy word aligner for Arabic-English machine translation. In *Proc. of HLT/EMNLP*, 2005, 89~96.

[9] Ben Taskar, Simon Lacoste-Julien, and Dan Klein. A discriminative matching approach to word alignment. In *Proc. of HLT/EMNLP*, 2005, 73~80.