

机器翻译系统融合中词对齐优化方法*

姚振宇, 杨沐昀, 李生, 赵铁军

(哈尔滨工业大学计算机科学与技术学院, 黑龙江哈尔滨 150001)

摘要: 词对齐是机器翻译系统融合中关键的一步, 词对齐结果的好坏直接决定融合结果的质量。针对现在广泛使用的多种词对齐方法结果中的差异, 本文探索了多种词对齐优化的策略。该策略的核心思想是对基于翻译编辑距离, 基于 GIZA++和基于 METEOR 三种方法的对齐结果进行取交, 取并和投票操作。实验结果表明, 在系统融合中我们提出的方法相对三种词对齐方法的机器翻译结果的质量上有所提升。

关键词: 词对齐; 系统融合; 机器翻译

Optimizing Word Alignment for Machine Translation System Combination

YAO Zhenyu, YANG Muyun, LI Sheng, ZHAO Tiejun

(School of Computer Science and Technology, Harbin Institute of Technology, Harbin,
Heilongjiang 150001, China)

Abstract: Word alignment is an important procedure in machine translation system combination. The quality of the alignment result will affect the translation result directly. We found that the results of the most commonly used word alignment methods are different. This paper combined different word alignment methods and proposed a strategy to optimize the alignment result. This method conduct union, intersection and voting operations on the result of TER, GIZA++ and METEOR word alignment results. The experiment result shows that the translation result on the optimized alignment results is better than any of these three methods.

Key words: word alignment; system combination; machine translation

1 引言

系统融合技术成功的应用到多个领域中, 并且取得了较好的性能提升。受到语音处理领域 ROVER[1]的启发, 2001年 Bangalore 将基于混淆网络的系统融合技术引入到机器翻译领域[2]。在机器翻译系统融合技术中, 基于混淆网络的系统融合技术相比于其他融合方法, 例如句子级系统融合等, 体现出了很大的性能优势[3]。

通常来说, 机器翻译中基于混淆网络的系统融合方法主要有四个过程: (1) 选取对齐骨架: 从所有翻译假设中选取对齐骨架。(2) 翻译假设对齐: 在对齐骨架和翻译假设之间建立词的对齐关系。(3) 构建混淆网络: 根据各个假设之间的对齐关系, 建立混淆网络。(4) 混淆网络解码: 从混淆网络中解码出最优的翻译结果。在四步中, 翻译假设对齐是至关重要的一个过程。翻译假设对齐结果将会是构建混淆网络的依据, 从而影响最终的融合结果, 而其中的核心就是不同译文单词之间的词对齐过程。

为了得到更好的词对齐结果, 近些年来人们已经提出了各种方法。机器翻译系统融合基于词错误率准则 (Word Error Rate, WER) 的词对齐方法利用了两个句子之间的编辑距离来

* 收稿日期: 2013-9-27 定稿日期: 2013-10-13

基金项目: 国家自然科学基金(No. 61272384 & 61105072); 国家高技术研究发展计划(863 计划) 重点项目(No. 2011AA01A207)。

作者简介: 姚振宇, 男, 硕士, 主要研究方向为机器翻译; 杨沐昀, 男, 副教授, 主要研究方向为机器翻译以及自然语言处理; 李生, 男, 教授, 主要研究方向为机器翻译以及自然语言处理; 赵铁军, 男, 教授, 主要研究方向为机器翻译以及自然语言处理。

计算词错误率,利用递归搜索出词错误率最低的词对齐方案。编辑操作包括词的插入(ins),删除(del)和替换(sub)三种。Sim et al. 提出利用翻译编辑率(Translation Edit Rate, TER)进行翻译假设之间的对齐[4]。这种基于翻译编辑距离的方法在上述三种编辑操作的基础上,加入了语句块的移动(shift)。Matusov et al. 提出了直接使用统计机器翻译中双语词对齐工具包 GIZA++进行单语句对的词对齐训练方法[5]。这种方法实际上是将句子间对齐关系当作统计机器翻译中的翻译模型进行训练。He et al. 提出一个基于间接隐马尔科夫模型(Indirect HMM)的翻译假设对齐方法[6],这个方法能够很好的处理同义词匹配和词序问题,同时避免了隐马尔科夫模型需要大量训练数据的问题。Krakos et al. 提出一个基于反向转录文法(ITG)的翻译假设对齐方法[7],这种对齐方法是计算 invWER 翻译质量评价尺度时产生的一种单语句对的词对齐。这种方法引入了句法信息这一语言学知识。METEOR[8]方法则是增量式的利用词之间的词形的匹配,词干信息和语义信息进行词之间的匹配,这便引入了词义这一信息到对齐过程中。

对于现有的这些方法,有些研究工作对他们进行了总述,也有对其中一些方法的对比分析研究。在现有的方法中,并没有一种方法能产生各方面性能均优于其它方法的对齐结果。其原因是在对齐过程中,不同方法使用了不同的知识作为对齐依据。在引入正常对齐结果同时,也会不同程度上的引入错误对齐信息。所以,在现有的对齐结果的基础上进行后处理,以最大限度集中不同对齐方法产生的正确对齐结果,同时去掉各自方法所带有的错误对齐结果为目的,便可以使得词对齐结果的质量得到提升,从而优化系统融合结果。

本文内容组织如下:第二章详细介绍 TER-based, GIZA++和 METEOR 这三种词对齐方法,并给出对不同词对齐方法结果的后处理优化方法;第三章描述实验设置及实验结果,并给出实验结果分析;第四章为总结。

2 词对齐方法及其结果优化方法

2.1 词对齐方法

基于翻译编辑距离的词对齐方法

基于翻译编辑距离(Translation Edit Rate, TER)的词对齐方法,指的是在翻译假设之间对齐的时候,不断搜索可能的对齐方案,选出编辑次数最少的一个作为最终的对齐方案。其中每个词的插入,删除,替换和移位都算作一次编辑操作。这种方法所得到的对齐方案为词之间一对一的对齐关系。

翻译编辑率的计算方法为:

$$TER = \frac{Num\ of\ edits}{Num\ of\ skltion\ words}$$

由于计算最优的带词调序的编辑距离的方法已经被证明是 NP-完全问题,所以,在计算 TER 时,用近似方法仅仅得到最优解。算法具体分为两步。对于不包括语块移位的其它三种操作可以利用动态规划得到;利用一种贪心搜索得出最优的转移策略,即反复地找出能减少插入、删除和替换三种编辑次数的转移策略,直到其他三种编辑次数不能再减少为止,从而确定最优转移策略。

基于 GIZA++的词对齐方法

Matusov et al. 提出一个基于统计模型的翻译假设对齐方法。这种方法主要特点是利用了整个测试集的语料信息来得出最优的对齐。与统计机器翻译中训练翻译模型的过程相同,这种方法首先假设同一个源句子的不同翻译假设之间是相互“对齐”的,由此创建出单语言的平行语料,再使用 GIZA++训练统计“对齐模型”,得出翻译假设对齐概率表,而最终的翻译假设的对齐方案是所有的对齐中概率最大的一个。与训练翻译模型过程的主要区别是,翻译模型的源语言与目标语言通常为两种不同语言,而将这个�方法利用到翻译假设对齐上,就成了同一语言之间的对齐模型的概率模型训练。因此,这种方法最终得到词之间的对齐关系是多

对一的。

对于给出的翻译假设 E_M ，用条件概率 $P(E_N|E_M)$ 来表示翻译假设 E_N 与 E_M 的对齐关系。这个条件概率通过引入一个隐含的词对齐关系 A 来计算：

$$P(E_N|E_M) = \sum P(E_N, A|E_M)$$

这个条件概率之后会被分解为对齐概率（alignment probability） $P(A|E_M)$ 和词典概率（lexicon probability） $P(E_N|A, E_M)$ ：

$$P(E_N, A|E_M) = P(A|E_M) * P(E_N|A, E_M)$$

为了得到这个概率，就像是统计机器翻译中一样，这里使用 IBM 第一模型和隐马尔可夫模型（Hidden Markov Model, HMM）来建立对齐模型。模型中的参数可以利用 EM 算法来进行训练。

METEOR

METEOR 在建立词之间的对齐关系时，会增量式的利用一系列不同的词之间的对齐方法来实现最终的词之间的对齐。首先，精确匹配模块（exact module）会将两个具有相同词形的单词进行对齐。然后利用一个同词干匹配模块（porter stem module）来匹配在利用 Porter stemmer 工具处理后具有相同词干的词。最后利用同词义匹配模块（WN synonymy module）在 WordNet 工具处理后将具有同词义的两个词进行对齐。与基于翻译编辑距离的词对齐方法相同，这种方法会产生一对一的词对齐结果。

这种方法使用 WordNet 同义词典来处理词义相似的单词：包括同义词和不同词性的同根词。通过查词典对参与对齐的两个翻译假设中出现的单词词条进行相互求交处理，来判断它们是否为同义词。

2.2 词对齐结果的优化方法

在对齐过程中，不同方法使用了不同的知识作为对齐依据，如基于 TER 的方法只利用的词形、位置信息；基于 GIZA++利用了文档集语料的信息贡献；基于 METEOR 的方法利用了词形、词干信息，还有语义这一语言学知识。在产生正确对齐结果同时，也会不同程度上的引入错误对齐信息。所以，在现有的对齐结果的基础上进行后处理，以最大限度集中不同对齐方法产生的正确对齐结果，同时去掉各自方法所带有的错误对齐结果为目的，便可以使得词对齐结果的质量得到提升，从而优化系统融合结果。

为了综合利用各种有利信息，获得准确率相对较高的词对齐结果，可以对三种方法的对齐结果进行取交操作，即只保留其中相同的对齐结果。通过这种方法，使得优化结果的词对齐准确率相对高于原有所有方法的对齐结果。

为了在对齐过程中，尽量多的利用对齐信息，使得对齐过程中有更多的规则可以依赖，可以对基于 TER、GIZA++和 METEOR 三种方法的对齐结果进行取并操作，即保留所有对齐结果。这样最大限度的考虑了对齐过程中的各种特征。

取交操作虽然能提高词对齐结果的准确率，但是会丢掉部分有用的对齐信息；相反，取并操作会最大限度利用各种对齐信息，但是一定程度上就降低了词对齐的准确率。综合两种方法，可以对三种方法产生的对齐结果进行投票，即保留在三种对齐结果中出现至少两次¹的对齐结果。

3 实验与结果

3.1 实验设置

实验中使用的数据集是 WMT2011 系统融合捷克语到英语评测数据。训练集和测试集分别包含 1003 句和 2000 句英文句子。单个系统输出包括 8 个系统的机器翻译提交结果，各个系统的 BLEU[9]得分如表 3-1 所示。三种词对齐方法在 MEMT[10]系统下得到结果得分如表

¹当出现三次时即对为三种方法取交操作的结果。

3-2 所示。各种词对齐方法得到对齐结果时均使用默认参数。MEMT 实验系统使用默认参数得到最终融合结果。

	DEV	TEST
online-b	28.22	29.46
udein	23.53	23.29
jhu	21.58	21.7
uppsala	22	21.15
cu-bojar	21.11	20.78
systran	19.65	19.05
cst	19.42	18.13
cu-zeman	18.01	17.62

表 3-1 单个系统得分 (BLEU%)

	DEV		TEST	
	对齐词数	BLEU%	对齐词数	BLEU%
TER	694965	25.05	1070731	25.1
GIZA++	763803	24	1406739	24.43
METEOR	584323	29.35	1282552	29.74

表 3-2 各种方法的融合结果

3.2 词对齐结果优化的实验结果及分析

三组实验的实验结果如下面三张表所示。其中表 3-3 给出了三种词对齐方法两两进行取并操作、三种方法同时取并之后的文档规模, 对齐词对的个数和利用取并得到的对齐结果融合后的结果的 BLEU 得分。表 3-4 给出了三种词对齐方法两两进行取交操作、三种方法同时取交之后的文档规模, 对齐词对的个数和利用取交的到的对齐结果融合后的结果的 BLEU 得分。表 3-5 给出了在三种方法的词对齐结果中进行投票, 利用得到的词对齐结果所得的融合结果的 BLEU 得分。

	DEV		TEST	
	对齐词数	BLEU%	对齐词数	BLEU%
TUG	874966	24.69	1608799	23.4
TUM	765641	24.31	1408515	24.07
GUM	812841	25.21	1489846	24.55
TUGUM	897426	25.19	1664017	24.86

表 3-3 对对齐结果取并的实验结果

	DEV		TEST	
	对齐词数	BLEU%	对齐词数	BLEU%
T∩G	583802	26.03	1080492	26.87
T∩M	513647	29.21	944772	29.48
G∩M	535285	29.39	987626	29.95
T∩G∩M	496069	29.05	916883	29.7

表 3-4 对对齐结果取交的实验结果

	DEV		TEST	
	对齐词数	BLEU%	对齐词数	BLEU%
Voting	552863	29.35	1015525	28.69

表 3-5 对对齐结果进行投票的实验结果

表 3-3 中数据表明，对所得到的对齐结果两两取并，利用所得的词对齐结果进行融合，最终的 BLEU 得分不能相比于原来的词对齐方法有较大幅度的提升。而三种方法同时取并的结果同样也没有较大提升。

从表 3-4 中可以看出，对所得到的对齐结果两两取交操作的融合结果相对于原来单独对齐方法均有性能提升。从数据表现来看，其中 GIZA++ 与 METEOR 两种方法的词对齐结果融合所得结果的表现最好，得分高于 METEOR 词对齐的融合结果得分，但是并没有通过显著性校验。基于 TER 和基于 GIZA++ 两种对齐结果取交之后所得词对齐的融合结果明显优于处理之前的基于 TER 和基于 GIZA++ 单独的性能表现。而这两种词对齐结果，与 METEOR 所得的对齐结果取交之后，都分别优于各自的融合性能。将三种词对齐方法所得的词对齐结果同时取交，得到的词对齐进行融合，所得结果的得分与 METEOR 方法的融合结果的得分较为接近。

从表 3-5 数据中可以看出，投票所得的结果相比于 GIZA++ 和 TER 方法，有所提升。与 METEOR 词对齐方法结果较为接近，并没有大幅度提升最终融合结果的得分。

分析其中原因，基于 TER、基于 GIZA++ 和 METEOR 三种词对齐方法在词对齐过程中，利用了不同的特征，所以会产生不同的词对齐结果。在所得的词对齐词对中，有些是正确的对齐信息，有利于提升最后融合结果的 BLEU 得分。相反由于对齐方法尚不完善，产生的对齐信息有些并不利于产生高质量的融合结果，甚至会降低最终的融合结果得分（这些对齐结果称之为“噪声”）。三种方法的对齐结果中，均含有不同的有利对齐对和噪声，但是由于对齐过程中所依赖的对齐特征不同，所以产生的噪声也不完全相同。

取并操作的结果相对于原来方法基本没有提升，是因为增加对齐对数量的同时也增加了大量噪声。这说明现有的词对齐方法所产生的词对齐对的数量已经足够，通过对现有词对齐结果取并，增大词对齐对的数量，并不能提升最终融合结果的质量。

对基于 TER 和基于 GIZA++ 的对齐结果进行取交操作，在一定程度上避免了各种方法由于各自缺陷所产生的噪声，最大限度的保留了正确信息，所以相比于原来方法会有所提升。说明了在各种词对齐方法中所产生的词对齐结果中仍然包含大量噪声，通过取交操作过滤掉部分噪声，降低对齐对的数量，最大限度保留正确对齐信息，提升词对齐结果的准确率，会使得融合结果质量得到提升。

进行投票的方法，减少了各种方法中独自存在的对齐结果，这样一来虽然减少了噪声的影响，但是同时也很大幅度的裁剪掉了有用对齐信息，所以其对齐结果不能使最终融合结果得分有大幅度提升。相比于取交操作，投票加入了在三种方法中出现两次的对齐结果，加入了更多的对齐信息，结果反而变坏。这也说明通过加入一些准确率不够高的对齐信息，扩大对齐对数量，不仅不能提升结果质量，反而会使得最终融合结果的质量下降。

4 结论

本文对基于机器翻译系统融合中的词对齐方法进行了概述，并在其中基于翻译编辑距离，基于 GIZA++ 和 METEOR 等三种方法的基础上，对词对齐结果进行了优化。实验结果表明，对不同的词对齐结果两两取交和投票，相比于参与相应后处理的原有两种词对齐方法会有不同程度上的性能改善。同时，METEOR 方法与 GIZA++ 两种方法的词对齐结果进行取交之后所产生的融合结果的表现优于后处理之前的所有方法。

目前来看，机器翻译中系统融合技术已经被证明能有效地改善翻译质量，而词对齐技术又是基于混淆网络的系统融合技术中的关键步骤。一个好的对齐结果将会使得最终融合结果

有很大的质量提升。本文中提出的方法表明，现有的词对齐方法所产生的词对齐对的数量已经足够，若想提升词对齐结果的质量，提升词对齐对的准确率才是关键。对于以后的工作，需要进一步分析对齐结果，查看相应的特点和不足之处，避免产生错误结果，提升对齐对的准确率，从而提升最终融合结果的质量。

参考文献

- [1] Jonathan G. Fiscus. A Post-Processing System to Yield Reduced Word Error Rates: Recogniser Output Voting Error Reduction (ROVER) [C]//In Proc. of IEEE ASRU Workshop. 1997: 347-352.
- [2] Bangalore B, Bordel G, Riccardi G. Computing consensus translation from multiple machine translation systems[C]//Automatic Speech Recognition and Understanding, 2001. ASRU'01. IEEE Workshop on. IEEE, 2001: 351-354.
- [3]Chen B, Zhang M, Li H, et al. A comparative study of hypothesis alignment and its improvement for machine translation system combination[C]//In Proc. of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP: Volume 2-Volume 2. ACL, 2009: 941-948.
- [4] Sim K C, Byrne W J, Gales M J F, et al. Consensus network decoding for statistical machine translation system combination[C]//Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on. IEEE, 2007, 4: IV-105-IV-108.
- [5] Matusov E, Ueffing N, Ney H. Computing Consensus Translation for Multiple Machine Translation Systems Using Enhanced Hypothesis Alignment[C]//EACL. 2006.
- [6] He X, Yang M, Gao J, et al. Indirect-HMM-based hypothesis alignment for combining outputs from machine translation systems[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2008: 98-107.
- [7] Karakos D, Eisner J, Khudanpur S, et al. Machine translation system combination using ITG-based alignments[C]//Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers. Association for Computational Linguistics, 2008: 81-84.
- [8] Banerjee S, Lavie A. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments[C]//Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization. 2005: 65-72.
- [9]Heafield K, Lavie A. Combining Machine Translation Output with Open Source: The Carnegie Mellon Multi-Engine Machine Translation Scheme[J]// The Prague Bulletin of Mathematical Linguistics, 2010, 93(1): 27-36.
- [10] Papineni K, Roukos S, Ward T, et al. BLEU: a method for automatic evaluation of machine translation[C]//Proceedings of the 40th annual meeting on association for computational linguistics. Association for Computational Linguistics, 2002: 311-318.