基于测试集的机器翻译系统显著性检验方法

付彬[1] 薛征山[2] 张大鲲[1] 郝杰[3]

1.东芝(中国)研究开发中心 北京 100600 2.北京搜狗科技发展有限公司 北京 100084

3. 58 集团 北京 100015

E-mail: {fubin1, zhangdakun}@toshiba.com.cn , xzskmust@163.com, haojie01@58ganji.com

摘要：显著性检验常用来判断系统之间的性能差异是否来源于系统的性能改善而不是随机误差。用于机器翻译系统的显著性检验通常以句子作为基本的抽样单位，忽略了抽样样本之间的独立性假设，而且用于机器翻译系统的自动评价标准不能对句子进行准确评价，因此引入了额外的随机误差。本文详细分析了影响显著性检验的这一问题，在 Clark 方法的基础上，提出了一种以测试集为基本单位的显著性检验方法。实验结果表明，该方法进一步消除了不同抽样方法对显著性检验的影响，获得更稳定的检验结果。

关键词：机器翻译；显著性检验；重采样

Document Based Significance Testing for MT

Bin Fu[1], Zhengshan Xue[2], Dakun Zhang[1] and Jie Hao[3]

1. Toshiba (China) R&D Center, Beijing, 100600 2. Sogou.com Inc. , Beijing, 100084 3. 58.com, Beijing, 100015

E-mail: {fubin1, zhangdakun}@toshiba.com.cn, xzskmust@163.com, haojie01@58ganji.com

Abstract: Significance testing usually detects whether differences of compared systems originated from performance improvement but not randomness. For machine translation, significance testing resamples sentence as a basic unit in resample process, ignoring the dependency hypothesis of each sentence in a test set. Furthermore, automatic evaluation metrics fail to measure a single sentence. Those problems drew randomness into significance testing. We provide systematic analysis of the weakness of significance testing, and put forward a significance testing method based on Clark's method, which resamples test set as a basic unit. Results show that our method weakens influence of different resampling methods on significance testing, and achieves more reliable testing result.

Key words: Machine Translation; Significance testing; Resampling

1 引言

近年来，基于统计的机器翻译技术快速发展，如何准确快速的评价机器翻译系统之间的性能差异是一个研究的热点问题。通常，基于统计的机器翻译方法利用定义好的自动评价标准[1]，在标准的测试集上计算系统的得分（如 BLEU 值、NIST 值等），然后根据得分的相对大小来确定系统之间的差异大小。研究发现这种仅依靠得分大小来判断系统差异的方法并不完全可靠，尤其是在系统得分差异较小的情况下，需要进一步利用显著性检验来度量系统之间的差异是否具有显著性，即推断系统之间的差异是由于随机波动引起的，还是系统间真实的性能差异引起的。

常用的显著性检验方法[2]，通过 Bootstrap[3]或者 Approximate Randomization (AR) 方法对样本进行重抽样，对于机器翻译系统而言，不同的重抽样方法、自动评价标准的选取以及测试集的大小等，都会影响显著性检验的结果。由于现有的统计机器翻译系统通常由大量的参数组成，参数的训练调节过程也是影响显著性检验的主要因素之一。目前的参数训练算法会受到诸多因素影响而得到不同的局部最优结果，这些因素包括，不同的随机初始化参数、对数线性模型使用的特征数量、参考译文数量、翻译方向、译文的搜索空间以及开发集大小等等。研究者目前还没有办法去准确估计这些因素对显著性检验的影响，因此 Clark 等人[4]在进行显著性检验的过程中，选取多组参数扩大测试集样本，并进行句子级重采样，相对于仅使用一组参数结果进行抽样的方法[5]，提高了显著性检验的稳定性。

Clark 的方法以及其他用于机器翻译系

统的显著性检验通常以句子作为基本的抽样单位，对系统输出的译文进行重抽样，构造抽样样本。这种方法虽然简单易行，但是忽略了机器翻译系统通常是以测试集为单位进行自动评价的。以句子为抽样单位来构造样本，不能保证测试集中词频分布的一致性，带来随机误差。而且常用的自动评价方法无法准确地给出单个句子的翻译质量评价。因此，为了进一步消除随机误差对显著性检验的影响，本文在 Clark 方法的基础上，改变以句子为抽样基本单位的做法，提出了以测试集为抽样单位的重采样方法，从多组参数得到的结果中构造机器翻译系统的检验样本。实验证明该方法提高了机器翻译系统显著性检验的稳定性和可靠性。本文的研究主要集中在以下三点：

1) 提出以测试集为单位的采样方法，进一步提高了显著性检验对不同抽样方法的稳定性；
2) 比较不同的采样方法在不同的自动评价标准下对显著性检验的影响；
3) 探究系统初始性能差异与 p-value 值的函数关系，并给出了用于判断系统间性能差异是否显著的临界值。

本文的组织结构如下：第 2 章介绍相关背景和工作；第3章描述本文提出的显著性检验方法以及与传统方法的区别；第4章给出实验结果和相关分析；最后进行总结。

2 相关背景和工作

2.1 机器翻译系统评测

机器翻译系统评测最早由人工从流利度和忠实度两个方面对译文进行打分，这种方法可以在主观上很好的评价翻译质量，但是人工评测费时耗力，并不能满足众多研究者对译文评测的需求。所以，研究人员希望能有一种方法对翻译系统进行自动快速评测，并能与人工评测结果具有较高关联性。随着机器翻译技术的发展和各项评测会议的举办，BLEU、NIST[6]、TER[7]和 METEOR[8]等自动评测方法逐渐被人们接受并采纳，成为机器翻译评测的主流标准。

BLEU 评测是常用的机器翻译自动评测方法之一，BLEU 值计算 n 元匹配准确率的几何平均数，并乘以一个指数形式的长度惩罚因子，其对数形式如下：

$$\log BLEU = \min\left(1 - \frac{r}{c}, 0\right) + \sum_{n=1}^{N} w_n log p_n$$

其中，p_n代表 n-gram 的准确率，r 代表参考译文长度，c 代表译文长度，$w_n = 1/N$，N 通常取值为 4。BLEU 值有几个明显的缺陷[9]：

1) 对于 n-gram 的位置信息不敏感；
2) 对于不同的 n-gram 给予相同的权重，而忽略匹配难度；
3) 不能很好地评价语义信息和识别词语歧义；
4) 对单一句子进行评价时失准。

基于以上几点，NIST 值为不同的 n-gram 给予了不同的权重，一定程度上避免了缺陷 2）中描述的问题。

2.2 参数调节中的不稳定性

常用的机器翻译系统采用对数线性模型[10]，其整体的翻译得分是若干特征和权重加权组合而成的。参数调节的目的就是学习特征中的权重。通常，我们利用最小错误率训练（Minimum Error Rate Training，MERT）[11]在开发集上进行参数向量的学习，然而，这种训练方法具有不确定性，容易产生局部最优，影响系统性能。前人做了很多工作来改进 MERT 算法的不足，如 Moore 等人[12]通过选择更好的初始化参数来去除 MERT 过程中的不确定性；Cer 等人[13]通过规范化 MERT 过程提高其所得参数向量在测试集上的泛化能力。

这些方法一定程度上提高了参数调节的稳定性，但对于显著性检验而言，并不能消除参数调节的不稳定性对显著性检验的影响。图 1 是对系统 X 和系统 Y 重采样条件下的翻译质量做曲线拟合，如果选取的抽样样本分别使系统 X 和系统 Y 落入两条曲线的重叠（阴影）区域，此时我们将不能准确地评价系统 X 和系统 Y 的性能差异。随着重叠区域增大，样本会更有可能包含随机误差，使我们无法得到可靠而稳定的显著性检验结果。

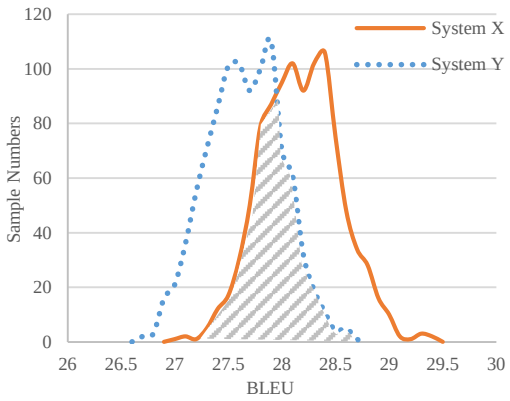

图 1 不同翻译系统重抽样的 BLEU 值分布

2.3 显著性检验及重抽样

显著性检验是根据一条假设由样本推断总体的一种方法。本文中，我们采用显著性检验来比较在同一测试集上翻译系统 y 和基线系统 x 的性能差异$S_{diff}(\mathrm{m}) = |S_x - S_y|$是否客观存在，m 代表影响性能差异的抽样误差。显著性检验通常设置一个较小的显著性水平标准∂（如 0.05、0.01），表示当空假设正确时，它被拒绝的概率不超过∂。一般来说，显著性检验计算出的 p-value 值越小，说明系统间的差异更多的是来自系统的改进而不是影响系统性能的随机变量，我们就越有理由拒绝原假设。

由于系统间性能差异$S_{diff}(\mathrm{m})$的准确分布无法给出，我们采取 Bootstrap 和 AR 重抽样法来扩大样本量拟合对应分布。空假设为系统之间没有差异。

Bootstrap 方法在机器翻译领域的应用十分广泛。该方法对测试集中的句子进行有放回的重复抽样，形成新的样本。Koehn[14]利用成对 Bootstrap 方法检验系统对不同语种的翻译质量；Zhang 等人[15]利用单边 Bootstrap 抽样方法检验翻译系统差异，并分析了如何构建可靠的测试集。Graham 等人[16]比较了双边、单边 Bootstrap 和 AR 抽样方法对不同评价指标的影响。实际使用中，双边 Bootstrap 的样本分布与空假设的分布虽然形状一样，但是位置会有偏移。在双边 Bootstrap 重采样中空假设分布均值为 0，所以抽样样本值通过减去本身均值即可补齐偏移量[17]。用于机器翻译的双边 Bootstrap 检验方法如表 1 所示。

与 Bootstrap 方法相比，AR 抽样方法不

表 1 双边 Bootstrap 抽样方法

1 赋值 c=0

2 计算单次 MERT 的翻译质量初始差异值

$$S_{diff} = |S_X - S_Y|$$

3 进行 B 次 Bootstrap 抽样，计算抽样样本均值

$$\mu_B = \frac{1}{B}\sum_{b=1}^{B}|S_{X_b} - S_{Y_b}|$$

4 For each bootstrap sample，b=1,…,B

1)以句子为单位，有重复地从系统 X 和 Y 的译文中抽取句子，形成新的抽样样本。

2)计算抽样样本观测值$|S_{X_b} - S_{Y_b}|$

3)如果$|S_{X_b} - S_{Y_b}| - \mu_B > S_{diff}$

c++

5 p_value = (c + 1)/(B + 1)

6 如果p_value <= ∂,则拒绝空假设H_0，反之则接受。

表 2 AR 抽样方法

1 赋值 c=0

2 计算单次 MERT 的翻译质量初始差异值

$$S_{diff} = |S_X - S_Y|$$

3 For each AR sample，r=1,…,R

1)For each sentence in test set

对系统 X 和 Y 的译文进行等概率交换

2)计算样本观测值$|S_{X_r} - S_{Y_r}|$

3)如果$|S_{X_r} - S_{Y_r}| > S_{diff}$

c++

4 p_value = (c + 1)/(B + 1)

5 如果p_value <= ∂,则拒绝空假设H_0，反之则接受。

用对样本分布进行假设，每次抽样都等概率的交换不同系统对同一句子的译文，如果测试集包含 N 个句子，该方法最多可以抽取2^N个样本对。当 N 较大时，我们一般会指定抽样次数。AR 方法在自然语言处理领域也有较多应用[18]。Riezler 等人通过分析 AR 与 Bootstrap 抽样方法对 p-value 值的影响，认为对于采用 NIST 值评分的机器翻译系统来说，Bootstrap 抽样比 AR 抽样更容易犯第一类错误（即否定了正确的原假设）。表 2 为利用 AR 抽样方法的流程图。

3 基于测试集的采样方法

3.1 抽样样本粒度分析

对用于机器翻译系统的显著性检验来说，抽样样本的粒度选择一直是被忽视的

问题。通常对于待检验的系统，我们只能得到其在特定测试集上的一组翻译结果（比如评测任务中的翻译系统），这种情况只能采用句子级抽样方法来构造检验样本。但是对于研究者自己开发的系统，或者开源系统在对比不同特征或不同算法时，我们可以通过多次参数训练得到在同一测试集上的多组测试样本，这使得选择测试集作为采样的基本单位成为可能。

一般来说，重抽样方法在进行抽样时都假设抽样样本是相互独立的[19]，以句子为基本单位的抽样方法的前提假设是，测试集中句子间的相互独立性。然而，从源语言端（源文）分析，Church[20]等人指出了几种在句子间的依赖关系。首先是单词间的联系，即一个单词的出现取决于它周围的单词，而且其词性也与周围词词性有关。其次是测试文本中出现过一次的具有实意的单词，有极大可能在后续的句子中出现，并且义项相同，尤其是对于篇章的翻译。从目标语言端（译文）分析，由于通常采用的自动评价标准 BLEU 值和 NIST 值，两种方法都不能对句子级译文进行准确评价，因此容易引入新的随机波动。

此外，Bootstrap 抽样是重复采样，很可能多次抽中一些n元匹配度低或者重复的句子，将基于测试集的词语分布转变成趋向于某些句子的词语分布，从而使 BLEU 值或 NIST 值无法对该样本进行准确评价。句子级的AR抽样仅调换相同句子的译文，并不会改变源语言中的词语分布。但是 AR 抽样根据空假设（系统间无显著差异）来等概率交换句子，而空假设是 BLEU 值或 NIST 值对于测试集译文的整体评价建立的，并不是假设测试集中的每一个译文均无显著差异。所以如果仅调换单一译文，也会对 AR 抽样带来额外的随机误差。

因此，不管是从句子间语言层面的依存关系，还是从不同的重采样方法来看，简单地进行以句子为单位的抽样来重新组合成新的抽样样本，忽略了重抽样过程中抽样样本应该相互独立的条件，从而影响显著性检验的稳定性。

表 3 基于 Bootstrap 抽样的本文方法

1 赋值 c=0

2 选取 M 次 MERT 结果，计算翻译分数

$$S_X = \frac{1}{M}\sum_{m=1}^{M} X_m ; S_Y = \frac{1}{M}\sum_{m=1}^{M} Y_m$$

3 计算样本初始差异值$S_{diff} = |S_X - S_Y|$

4 进行 B 次 Bootstrap 抽样，计算抽样样本均值

$$\mu_B = \frac{1}{B}\sum_{b=1}^{B} |S_{X_b} - S_{Y_b}|$$

5 For each bootstrap sample，b=1,…,B

1)以文档为单位，有重复地从系统 X 和 Y 的译文中抽取，形成新的抽样样本。

2)计算抽样样本差异值

$$S_{diff} = |S_{X_b} - S_{Y_b}|$$

3)如果$|S_{X_b} - S_{Y_b}| - \mu_B > S_{diff}$

c++

6 p_value = (c + 1)/(B + 1)

7 如果p_value <= ∂,则拒绝空假设H_0，反之则接受。

表 4 基于 AR 抽样的本文方法

1 赋值 c=0

2 选取 M 次 MERT 结果，计算翻译分数

$$S_X = \frac{1}{M}\sum_{m=1}^{M} X_m ; S_Y = \frac{1}{M}\sum_{m=1}^{M} Y_m$$

3 计算样本初始差异值$S_{diff} = |S_X - S_Y|$

4 For each AR sample，r=1,…,R

1)For each translation set in M times parameter

对X_m和Y_m的译文文档进行等概率交换

2)计算抽样样本观测值$|S_{X_r} - S_{Y_r}|$

3)如果$|S_{X_r} - S_{Y_r}| > S_{diff}$

c++

5 p_value = (c + 1)/(B + 1)

6 如果p_value <= ∂,则拒绝空假设H_0，反之则接受。

3.2 基于测试集的抽样方法

为了避免这一问题，本文基于 Clark 的方法，提出了一种基于测试集的机器翻译系统显著性检验方法。首先，我们进行 M 次参数训练，计算在每次 M 个参数情况下的系统得分X_m，以 M 个翻译得分均值计算系统差异值；我们以翻译系统的译文文档作为抽样样本，Bootstrap 重采样过程中，对系统中 M 个文档进行有放回的采样；而在 AR 重采样时，等概率交换 M 个参数下

不同系统间的译文文档；重采样结束后，计算采样样本均值的差异，作为本次抽样的结果。

这种方法将细粒度的抽样样本（句子）转化为粗粒度样本（文档、测试集），使抽样样本能较好地通过自动评测方法衡量翻译系统自身性能，可以很好地避免句子之间独立性假设所引发的不稳定性。表 3、表 4 分别为基于测试集的 Bootstrap 和 AR 两种抽样方法的详细流程。

4 实验和分析

4.1 翻译系统及语料

本文选用 WMT 2006 提供的英—法语料（约 68 万句）作为实验数据。翻译系统采用 MOSES[21]基于短语的机器翻译系统，选取 14 个基本特征进行训练，包括双向短语翻译概率，双向词汇化翻译概率，语言模型，基于位置的调序概率，词汇化调序概率等。

表 5 不同语料训练的翻译系统

序号	系统名	训练语料组成	规模
1	TOTAL	全部 训练语料	688K
2	RAND1 (R1)	随机抽取 一半训练语料	344K
3	RAND2 (R2)	随机抽取 一半训练语料	
4	TOP	训练语料 前半部分	
5	BOTTOM (BOT)	训练语料 后半部分	
6	ODD	奇数行 训练语料	
7	EVEN	偶数行 训练语料	

为了评价翻译系统的差异性，我们从 68 万训练集中抽取语料，组成新的训练数据，并以此训练出 7 个性能不同的翻译系统以产生对比系统（如表 5 所示）。利用这种方法构造的翻译系统在算法上具有一致性，从直观上考虑系统 1（TOTAL）和其他 6 个系统具有显著性差异，系统 2（R1）和系统 3（R2）不具有显著差异，后续的实验结果也验证了这一点。

同时，我们选用 MERT、MIRA[22]和 PRO[23]方法在开发集上进行参数调节，每个方法各取 20 次最优结果，共 60 组参数向量，用以消除参数调整的随机性对系统性能的影响。每个系统对 3064 句测试集进行翻译。7 个系统进行两两组合，在系统间分别采用三种检验方法（Riezler 方法、Clark 方法和本文方法）逐一进行显著性检验。表 6 为三种检验方法的区别比较。

表 6 三种检验方法区别

方法	参数调节次数	抽样样本粒度	样本大小
Riezler 方法	1 次	句子	3064
Clark 方法	60 次	句子	3064*60
本文方法	60 次	文档 (测试集)	60

4.2 置信区间

对于待检验的机器翻译系统而言，对样本进行多次重采样后，所形成的分数置信区间重叠越大，就越有可能造成显著性检验的判断错误。本节中我们利用三种不同的检验方法计算出 7 个系统在 Bootstrap 抽样下各自性能的置信区间（置信度 95%），并对系统间置信区间的重叠区域进行比较。

首先给出重叠区域的计算方法。假设我们已经获得了系统 x 置信区间(n_x, m_x)和样本的近似分布$D_x = N(\mu_x, \sigma_x)$， 系统 x 与 y 的总区域S_{total}由下式得出：

$$S_{total} = \int_{n_x}^{m_x} D_x dx + \int_{n_y}^{m_y} D_y dy$$

假设系统 x 和 y 的样本分布有一个唯一的交点 j，且$\mu_x < \mu_y$,系统间的重叠区域$S_{overlap}$计算方法由下式得出

$$S_{overlap} = \int_{j}^{m_x} D_x dx + \int_{n_y}^{j} D_y dy$$

为了更直观的比较重叠区域，我们计算其重叠比$P_{overlap}$：

$$P_{overlap} = \frac{S_{overlap}}{S_{total} - S_{overlap}}$$

表 7 七个翻译系统的 Bootstrap 抽样置信区间

Sys.	BLEU			NIST		
	Riezler 方法	Clark 方法	本文方法	Riezler 方法	Clark 方法	本文方法
TOTAL	**[27.44, 28.82]**	**[28.11, 28.28]**	**[28.17, 28.22]**	**[7.26, 7.45]**	**[7.34, 7.36]**	**[7.34, 7.36]**
R1	[26.86, 28.22]	[27.51, 27.69]	[27.55, 27.64]	[7.19, 7.38]	[7.26, 7.28]	[7.26, 7.28]
R2	[26.89, 28.25]	[27.49, 27.66]	[27.53, 27.61]	[7.19, 7.38]	[7.26, 7.28]	[7.26, 7.28]
TOP	[26.68, 28.02]	[27.18, 27.35]	[27.23, 27.29]	[7.16, 7.35]	[7.23, 7.25]	[7.23, 7.25]
BOT	**[27.37, 28.74]**	**[27.81, 27.99]**	**[27.86, 27.93]**	**[7.24, 7.43]**	**[7.31, 7.33]**	**[7.31, 7.32]**
ODD	[26.94, 28.30]	[27.52, 27.70]	[27.57, 27.65]	[7.20, 7.38]	[7.30, 7.33]	[7.27, 7.29]
EVEN	[27.32, 28.67]	[27.74, 27.91]	[27.79, 27.86]	[7.21, 7.40]	[7.27, 7.30]	[7.28, 7.29]

表 7 包含了 7 个系统在三种不同检验方法下的置信区间。我们发现，在表 7 中，对于所有 7 个系统，本文的方法得到的置信区间在两种自动评测方法下，均小于前两种方法。其中系统 TOTAL 与 BOT，如果使用 Riezler 方法，以 BLEU 值评测为例，将会得到相近的置信区间 TOTAL [27.44,28.82] 和 BOT[27.37,28.74]，而在其余方法中置信区间却没有交集（本文方法中 TOTAL [28.17,28.22]；BOT[27.86,27.93]）。因为 Riezler 方法的样本仅是一组参数的翻译结果，样本数量小，而且 MERT 的局部最优致使参数向量具有随机性，对这种样本进行抽样会引入随机误差，造成系统的置信区间跨度很大，从而可能造成重叠区域。而其余两种方法因为选取了多组参数向量进行翻译，扩大了样本空间，减小了随机误差对系统性能的影响，所以抽样样本的置信区间相对稳定，重叠区域也比较小。

图 2 和图 3 列出了重叠比例较高的比较系统。从图 2 和图 3 中可以发现 Riezler 方法的重叠比例总是最高，即显著性检验的随机误差最大。Clark 方法在某些情况（如 R1-ODD，BLEU）下重叠区域的比例和 Riezler 方法相近，这说明 Clark 方法在某些情况下，也会产生较大的随机误差，主要是因为 Clark 方法使用的句子级别的抽样方法会因为句子并不相互独立，造成样本中词语分布的偏移，影响自动评测方法对翻译性能的判断。而本文提出的方法，由于抽样样本以测试集为单位，相互之间独立，其置信区间和重叠区域总是小于另外两种句子级抽样方法。说明本文方法可以很好的控制由抽样粒度对词语分布带来的影响，形成更加稳定的测试集样本，利于自动评测方法对系统性能进行真实评测。

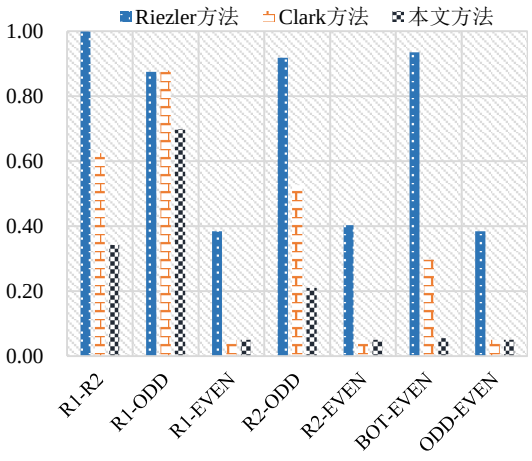

图 2 置信区间重叠比例（BLEU 值）

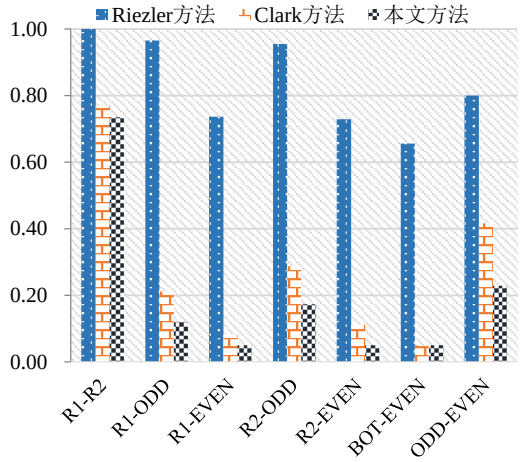

图 3 置信区间重叠比例（NIST 值）

表 8 系统间的显著性检验结果

比较系统	方法	BLEU		NIST	
		AR	Bootstrap	AR	Bootstrap
R1-R2	Clark 方法	0.077	0.234	0.268	0.291
	本文方法	0.324	0.121	0.719	0.265
R1-ODD	Clark 方法	0.666†	0.369	***0.001***	***0.069***
	本文方法	0.783	0.303	0.005	0.028
R2-ODD	Clark 方法	***0.031***†	***0.199***	***0.001***	***0.101***
	本文方法	0.117	0.082	0.037	0.045
BOT-EVEN	Clark 方法	***0.001***	***0.115***	0.001	0.001
	本文方法	0.018	0.006	0.001	0.001
ODD-EVEN	Clark 法	0.001	0.001	***0.004***	***0.154***
	本文方法	0.001	0.001	0.134	0.089

4.3 系统间的显著性检验

对于已有的7个系统，我们依次配对形成 21 组比较系统，表 8 列出其中有代表性的 5 组使用 Clark 方法和本文方法计算得到的 p-value 值，其余组的 p-value 值均为 0.001。重采样次数为 10000。

从表8中可以看出，对于相同的自动评价准则，以 0.05 为显著性水平标准，本文方法使用 Bootstrap 重采样可以得到与 AR 重采样一致的判断结果，并不会因为不同的采样方法而影响显著性检验结果。而 Clark 方法，尽管通过采用多组参数消除了参数调节引起的随机误差，但由于忽略了句子级抽样方法的独立性问题，得到的 Bootstrap 和 AR 抽样在 5 组数据上（表 8 中粗斜体）都出现了相反的判断，造成显著性检验的不稳定结论。

此外，不管是 Clark 方法还是本文方法，比较系统 R1 和 R2 显示两个系统性能不存在显著差异（p-value 大于 0.05），则理论上系统 R1 和 ODD，以及系统 R2 和 ODD 应该表现出具有一致的显著性差异。对于 BLUE 值自动评价标准，系统 R1-ODD 和系统 R2-ODD，Clark 方法在使用 AR 抽样时得到了相反的判断（表 8 中†标示数值）。这是因为 AR 抽样过程，尽管我们假设系统间翻译质量相似，但并不是平均体现在每个句子上，也就是说单一句子并不能反映其性能。所以进行句子级的交换可能会将翻译质量不好的句子交换到另一个系统，为显著性检验带来误差。本文方法以测试集为单位进行交换，样本间相互独立，交换的样本可以衡量系统性能，所以并没有这种情况发生。

4.4 初始差异值的随机性

无论使用哪一种方法进行显著性检验，我们都需要计算系统的初始差异值 $S_{diff} = |S_x - S_y|$。对于只选择一组参数结果进行显著性比较的 Riezler 方法，其初始差异值的随机性较大，容易引起不稳定的结论。图 4 以 TOTAL 系统和 BOT 系统为例，给出了两个系统在 60 组参数条件下的性能拟合。从图中可以看出，与样本抽样分布类似，TOTAL 与 BOT 系统的差异期望值为 0.3BLEU，如果我们每次显著性检验时只取单一参数的结果作为初始样本，由于样本的取值范围跨度较大，很可能会造成不稳定的S_{diff}值，影响显著性检验的可靠性。而对于本文方法，由于是对多组参数下的译文进行综合评价，极大减小了初始差异值的不稳定性。一般来说，选择越多的参数调节数量会得到更稳定的初始差异值。

为了研究初始差异值对翻译系统显著性检验的影响，我们选取了表 8 中前 4 组系统（其余系统 p-value 值在 BLEU 作为评价指标时皆为 0.001），观察其 p-value 值随着S_{diff}变化的趋势。实验结果表明(图 5)，p-value 值的大小随着系统初始差异值S_{diff}呈单调变化趋势，即初始差异值S_{diff}越大，

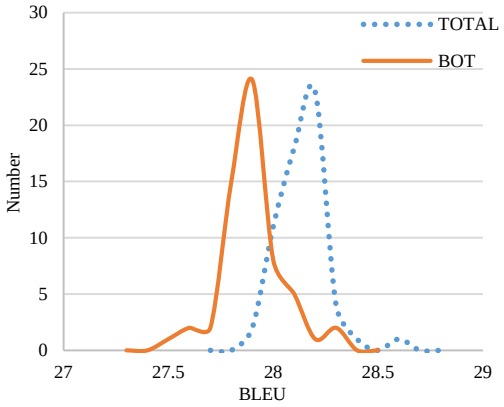

图 4　多组参数下系统初始性能分布

其 p-value 值越小，就越有可能判断系统间的差异是显著的。因此，通过拟合出的S_{diff}在特定段的分布情况，我们可以找到 p-value 值为 0.05 对应的差异值点，为以后在相同测试集上的系统间显著性检验提供参考。在图 5 的 Bootstrap 方法中，当 p-value 为 0.05 时，$S_{diff} \approx 0.05\,BLEU$，也就是说当$S_{diff} > 0.05$时，我们就能大致认为系统间存在显著性差异，从而快速对系统之间差异性做出粗略判断。当然，随着测试集中句子数量的上升，系统会在更小的S_{diff}值上得到 p-value 为 0.05 的结果(Berg-Kirkpatrick [24])，因此这种方法仅适用于在确定测试集的条件下，比如研究者对于新方法开发阶段的系统进行快速评价。

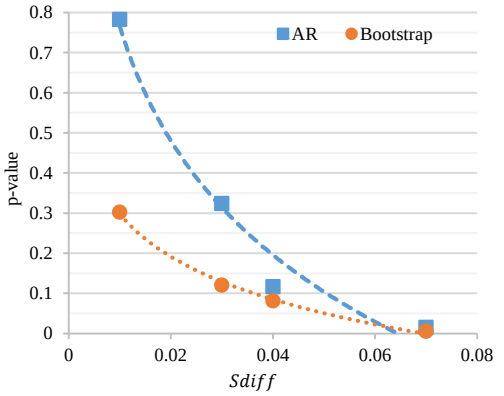

图 5　系统间 p-value 与初始差异值关系

5 结论

用于机器翻译领域的显著性检验方法通常是以句子为单位进行抽样。本文研究发现，以句子为单位进行抽样，忽略了抽样样本之间相互独立的条件，影响样本的词语分布，以及 AR 抽样的交换稳定性，使得自动评测方法无法通过 Bootstrap 或 AR 抽样形成的样本稳定地评测系统性能，从而影响显著性检验的稳定性。因此，本文在 Clark 方法的基础上，提出了一种以测试集为单位的采样方法，利用多次参数调节的结果进行抽样。该方法减小了句子级采样在 Bootstrap 和 AR 方法中引入的随机误差，提高了机器翻译系统间显著性检验的稳定性和可靠性。

实验结果表明，本文的方法与具体的重采样方法无关，可以在 Bootstrap 和 AR 抽样上获得一致的检验结果。同时，以测试集为单位进行抽样，降低了系统置信区间跨度和重合率，使得初始的系统差异值选取更加稳定，更容易获得稳定的显著性检验结果。本文的方法对机器翻译方法本身没有限制，可以应用于不同架构的机器翻译系统。

参考文献

[1]. 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.

[2]. Kumar S, Byrne W. Minimum bayes-risk decoding for statistical machine translation [R]. Johns Hopkins University Baltimore MD center for Language and Speech Processing (CLSP), 2004.

[3]. Efron B, Tibshirani R J. An introduction to the bootstrap [M]. CRC press, 1994.

[4]. Clark J H, Dyer C, Lavie A, et al. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability [C]//Proceedings of the 49th AMACL: Human Language Technologies: short papers-Volume 2. Association for Computational Linguistics, 2011: 176-181.

[5]. Riezler S, Maxwell J T. On some pitfalls in automatic evaluation and significance testing for MT [C]//Proceedings of the ACL workshop on intrinsic and extrinsic evalu-

ation. Measures for machine translation and/or summarization. 2005: 57-64.

[6]. Doddington G. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics [C]//Proceedings of the second international conference on Human Language Technology Research. Morgan Kaufmann Publishers Inc., 2002: 138-145.

[7]. Snover M, Dorr B, Schwartz R, et al. A Study of Translation Error Rate with Targetted Human Annotation [J]. LAMP-TR-126, CS-TR-4755, UMIACS-TR-2005-58, University of Maryland, College Park, MD, 2005.

[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, 29: 65-72.

[9]. Callison-Burch C, Osborne M. Reevaluating the role of BLEU in machine translation research [C]//In EACL. 2006.

[10]. Koehn P, Och F J, Marcu D. Statistical phrase-based translation [C]//Proceedings of the 2003 Conference of NACACL on Human Language Technology-Volume 1. Association for Computational Linguistics, 2003: 48-54.

[11]. Och F J. Minimum error rate training in statistical machine translation [C]// Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1. Association for Computational Linguistics, 2003: 160-167.

[12]. Moore R C, Quirk C. Random restarts in minimum error rate training for statistical machine translation [C]//Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1. Association for Computational Linguistics, 2008: 585-592.

[13]. Cer D, Jurafsky D, Manning C D. Regularization and search for minimum error rate training [C]//Proceedings of the Third Workshop on Statistical Machine Translation. Association for Computational Linguistics, 2008: 26-34.

[14]. Koehn P. Statistical Significance Tests for Machine Translation Evaluation [C]// EMNLP. 2004: 388-395.

[15]. Zhang Y, Vogel S. Significance tests of automatic machine translation evaluation metrics [J]. Machine Translation, 2010, 24(1): 51-65.

[16]. Graham Y, Mathur N, Baldwin T. Randomized significance tests in machine translation [C]//Proceedings of the ACL 2014 Ninth Workshop on Statistical Machine Translation. 2014: 266-274.

[17]. Resampling B. Computer-intensive methods for testing hypotheses: an introduction [J]. Computer, 1989.

[18]. Chinchor N, Lewis D D, Hirschman L. Evaluating message understanding systems: an analysis of MUC-3 [J]. Computational linguistics, 1993, 19(3): 409-449.

[19]. Yeh A. More accurate tests for the statistical significance of result differences [C]// Proceedings of the 18th conference on Computational linguistics-Volume 2. Association for Computational Linguistics, 2000: 947-953.

[20]. Church K W, Mercer R L. Introduction to the special issue on computational linguistics using large corpora [J]. Computa-tional linguistics, 1993, 19(1): 1-24.

[21]. Koehn P, Hoang H, Birch A, et al. Moses: Open source toolkit for statistical machine translation [C]//Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions. Association for Computational Linguistics, 2007: 177-180.

[22]. Cherry C, Foster G. Batch tuning strategies for statistical machine translation [C]//

Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, 2012: 427-436.

[23]. Hopkins M, May J. Tuning as ranking [C]. Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2011: 1352-1362.

[24]. Berg-Kirkpatrick T, Burkett D, Klein D. An empirical investigation of statistical significance in NLP [C]//Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Association for Computational Linguistics, 2012: 995-1005.