文章编号：2095-2163(2020)06-0187-04 中图分类号：TP391.1 文献标志码：A

基于 Back-translation 的语法错误纠正

邓俊锋，朱聪慧，赵铁军
（哈尔滨工业大学 计算机科学与技术学院，哈尔滨 150001）

摘 要：语法错误纠正，面临着较为严重的数据稀疏问题，这给机器翻译方法在该任务上的应用带来了直接的困难。本文首次提出在 back-translation 阶段采用 sampling 解码策略，并对比基于不同解码策略合成的伪平行句对给训练语法错误纠正模型带来的影响。在标准数据集 CoNLL-2014 Test Set 上的实验结果表明，本文提出的方法能显著提升语法错误纠正的性能。
关键词：语法错误纠正；Back-translation；数据增强

Grammatical error correction based on Back-translation

DENG Junfeng, ZHU Conghui, ZHAO Tiejun
(School of Comquter Science and Technology, Harbin Institute of Technology, Harbin 150001, China)

【Abstract】 At present, the scale of "error-correction" parallel corpus for grammatical error correction is limited, which brings direct difficulties to the application of machine translation approachs in this task. To alleviate this problem, We synthesize pseudo "error-correction" parallel sentence pairs based on back-translation method. In order to introduce a variety of grammatical errors, we construct pseudo parallel sentence pairs by using sampling decoding in the back-translation, and we compare the effects of pseudo data synthesized by using different decoding strategies on training forward grammatical error correction model. The experimental results on CoNLL-2014 Test Set show that the proposed method can synthesize effective pseudo sentence pairs and improve the performance of grammatical error correction.
【Key words】 Grammatical Error Correction; Back-translation; Data Augmentation

0 引 言

近年来，采用序列到序列学习框架的神经机器翻译方法，俨然成为语法错误纠正研究的主流[1-2]。神经机器翻译研究中最新的模型不断被应用到语法错误纠正任务中，并取得远超其他方法的性能。然而，受限于“错误-纠正”平行语料的规模，拥有巨大参数空间的神经语法错误纠正模型很难被充分训练，导致模型的泛化能力大打折扣。

之前大部分研究工作仅关注于少数特定类型的语法错误的生成[3-4]。例如，不可数名词、冠词、介词等。部分最新的工作开始逐渐尝试生成全部类型的语法错误，并从句子层面考虑伪平行句对中的语法错误多样性[5-6]。

本文使用神经机器翻译中的 back-translation 方法[7]来合成伪平行句对。首先，利用种子语料训练一个语法错误生成模型，在训练时，模型的输入为“错误-纠正”平行句对中书写正确的纠正句子，输出为平行句对中含语法错误的句子。使用该反向模型，将海量书写正确的句子“翻译”成含语法错误的句子，进而构造伪“错误-纠正”平行句对。然而，Xie 等人[5]的工作表明，在反向模型的解码阶段，若直接采用 beam search 策略，生成的伪错误句子将缺乏足够的语法错误多样性。不同于 Xie 等人使用加噪的 beam search 解码策略来引入更多的语法错误，本文直接使用 sampling 解码策略。在语法错误纠正任务的标准数据集 CoNLL-2014 Test Set 上的实验结果表明：本文提出的方法能合成有效的伪“错误-纠正”平行句对，从而帮助语法错误纠正模型的训练。

1 方法

1.1 Transformer 语法错误纠正模型

1.1.1 模型结构

Transformer 包含一个编码器和一个解码器，图 1 给出了 Transformer 的模型结构。给定源端错误句子 $x = (x_1, x_2, \ldots, x_m)$，$x_i \in X$，$X$ 为源端词表，Transformer 编码器将 x 编码为连续空间中的一组隐含状态表示 $e = (e_1, e_2, \ldots, e_m)$。基于这一表示，Transformer 解码器逐时间步地生成目标端纠正句子 $y = (y_1, y_2, \ldots, y_n)$，$y_i \in Y$，$Y$ 为目标端词表。

作者简介：邓俊锋(1995-)，男，硕士研究生，主要研究方向：机器翻译、语法错误纠正；朱聪慧(1979-)，男，博士，硕士生导师，主要研究方向：自然语言处理、机器翻译。

收稿日期：2019-06-20

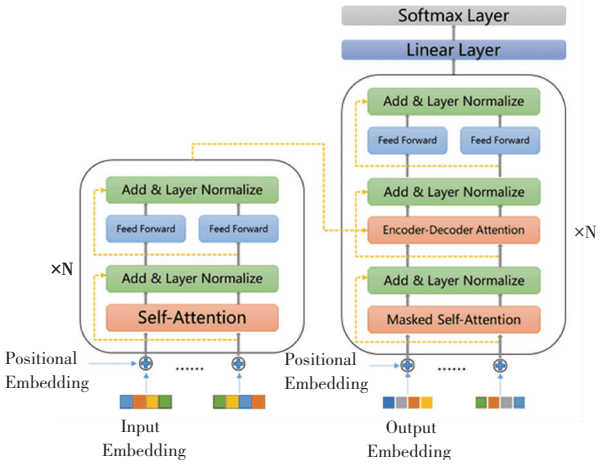

图 1 Transformer 模型结构

Fig. 1 Transformer model architecture

1.1.2 模型训练

语法错误纠正模型建模以下条件概率分布：

$$p(y \mid x) = \prod_{t=1}^{n} p(y_t \mid x, y_{1:t-1}; \theta). \quad (1)$$

其中，θ 为语法错误纠正模型的参数。训练时，使用极大似然估计学习模型参数：

$$\theta = \text{argmax} \sum_{t=1}^{n} \log p(y_t \mid x, y_{1:t-1}; \theta). \quad (2)$$

1.1.3 解码策略

给定输入的错误句子 x，采用 beam search 解码生成目标端纠正句子 y_{hyp}，在每一个时间步，保留得分最高的前 k 个候选前缀句子。此外，为了抑制模型偏向于输出较短句子的行为，在原始的似然得分中引入长度惩罚项，具体计算公式为：

$$\text{score}(y_{hyp}, x) = \log(p(y_{hyp} \mid x)) / LP(y_{hyp}), \quad (3)$$

$$LP(y_{hyp}) = (5 + \mid y_{hyp} \mid)^{\alpha} / (5 + 1)^{\alpha}, \ \alpha \in [0.6, 0.7]. \quad (4)$$

1.2 面向语法错误多样性的 back-translation

使用 back-translation 方法，对书写正确的句子 Y_{clean} 施加“噪声”，以构建伪平行句对 $(Y_{corrupt}, Y_{clean})$。之后，组合种子语料 (X, Y) 和生成的 $(Y_{corrupt}, Y_{clean})$，将其记作混合语料 $(X_{mixtcure}, Y_{mixture})$，训练语法错误纠正模型。

1.2.1 语法错误生成模型

使用“错误-纠正”平行句对 (X, Y) 训练反向神经语法错误生成模型。模型结构采用 Transformer，给定错误句子 $x = (x_1, x_2, \ldots x_m)$ 和对应的纠正句子 $y = (y_1, y_2, \ldots, y_n)$，语法错误生成模型建模“加噪”概率：

$$p(x \mid y) = \prod_{t=1}^{m} p(x_t \mid y, x_{1:t-1}; \theta_{backward}). \quad (5)$$

模型损失函数定义为：

$$Loss(\theta_{backward}) = \sum_{t=1}^{m} -\log p(x_t \mid y, x_{1:t-1}; \theta_{backward}). \quad (6)$$

学习目标是最大化模型在种子语料上的似然：

$$\theta_{backward} = \text{argmax} \sum_{t=1}^{m} \log p(x_t \mid y, x_{1:t-1}; \theta_{backward}). \quad (7)$$

在种子语料 (X, Y) 上训练好语法错误生成模型后，用其“翻译”书写正确的句子 Y_{clean}，得到含语法错误的句子，进而构建伪平行句对 $(Y_{corrupt}, Y_{clean})$。

1.2.2 面向语法错误多样性的 back-translation

在机器翻译中使用 back-translation 方法生成伪源语言句子时，反向模型一般采用 greedy search 或者 beam search 解码。关于 back-translation 的最新研究[8]表明，采用 sampling 或者加噪的 beam search 解码能取得更好的效果。因为在构造伪平行句对时，greedy search 或者 beam search 解码生成的伪源语言句子缺乏足够的多样性，且无法全面呈现反向模型建模的概率分布 $P(Source \mid Target)$。相比之下，采用 sampling 或加噪的 beam search 解码能在生成的伪源语言句子中引入更多的多样性，从而为后续正向模型的训练提供更强的学习信号。

面向语法错误纠正的“错误-纠正”平行语料，其源端错误句子和目标端纠正句子往往存在大量的重复。无论是语法错误纠正模型，还是语法错误生成模型，使用具有这种特性的语料进行训练，模型往往趋于“保守”。若在构造伪平行句对时，反向模型采用 greedy search 或者 beam search 解码策略，在生成的伪错误句子中，只会包含极少的语法错误，这样构造出的伪平行句对只能提供微弱的学习信号。在早期实验中发现，若采用这两种解码策略，在反向模型输出的伪错误句子中，有相当一部分和输入的书写正确的句子完全相同。Xie 等人受神经对话生成研究[9]的启发，在做 back-translation 时，采用加噪的 beam search 解码来生成伪错误句子，并证实有效。本文指出，采用 sampling 解码构造的伪平行句对同样能帮助训练，且效果比 greedy search 解码策略更好。

2 实验

2.1 实验数据与评价指标

本文在实验中，使用 NUCLE[10] 和 Lang-8[11] 作为训练语料。原始的 Lang-8 语料包含约 80 多种语言的句子，可使用语言识别工具 langid.py，过滤掉原始语料中的非英语句子，以及源端错误句子和目标

端纠正句子完全相同的平行句对。对于不同平行句对中源端错误句子相同的情况，仅保留其中之一，至此，筛选出大约 120 万条“错误-纠正”平行句对。进一步分析 Lang-8 语料发现，在一部分平行句对的目标端纠正句子中，包含修订者额外给出的评注。例如，“maybe you could say XXX”、“XXX is ok，but it sounds a little strange”，对于这样的平行句对，可利用手工设计的一些匹配模式，以及一些启发式规则（例如，要求目标端纠正句子和源端错误句子的长度比值不能超过 1.5），将其过滤掉。最终，用于模型训练的 Lang-8 语料的句对数为 930428。NUCLE 是 CoNLL-2013、CoNLL-2014 语法错误纠正评测任务提供的语料，官方已经对其进行预处理，包含 57151 条平行句对。此外，在使用 back-translation 方法合成伪平行句对时，用到了 WMT-2017 提供的英语单语语料 News Crawl 2013，原始语料包含约 1500 万个句子，实验中仅使用其中前 100 万个句子。表 1 给出了实验中使用的训练数据的统计信息。

表 1　训练数据

Tab. 1　Training Data

类型	语料	句对(子)数
平行语料	Lang-8	930428
平行语料	NUCLE	57151
单语语料	News Crawl 2013	100 万

测试数据为 CoNLL-2014 Test Set[12]，使用官方提供的 M^2 打分器[13]，评估指标为 $F_{0.5}$ 值。为了方便和之前的工作进行对比，在此选用 CoNLL-2013 Test Set[14] 作为开发集。

表 2　测试集 & 评价指标

Tab. 2　Test Set & Metrics

测试数据	句对数	评价指标
CoNLL-2014 Test Set	1 312	$F_{0.5}$

2.2　实验设置

2.2.1　语法错误生成模型训练设置

基于开源库 tensor2tensor 实现 Transformer，采用 $\text{Transformer}_{\text{base}}$ 模型；编码器、解码器各包含 6 个相同的网络层；各层输入、输出的维度，以及 Embedding 维度设置为 512；多头注意力层使用 8 个头，在单个头中，查询向量、键向量、值向量的维度均设置为 64；前向神经网络子层的隐含层维度设置为 2048，在模型的 Embedding 层、以及各子层的输出处应用 dropout，dropout 率设置为 0.3；label smoothing 率设置为 0.1；使用带学习率衰减的 Adam 优化算法，初始学习率设置为 0.000 3；warmup 步数设置为 16000；在 4 块 GeForce RTX 2080 Ti 上训练模型；batch 的大小设置为 256；最大句子长度设置为 50；超过该长度的部分直接截断，更新约 30 000 步后停止。源端、目标端使用不同词表，分别取各自出现最频繁的前 30000 个 BPE[15]（byte pair encoding）子词单元。

2.2.2　语法错误纠正模型训练设置

使用语法错误生成模型“翻译”News Crawl 2013 语料中前 100 万个英语句子，分别采用 greedy search 和 sampling 解码策略，设置生成的伪错误句子长度不超过 50 个词，这样构造出 100 万条伪“错误-纠正”平行句对（$Y_{corrupt}$，Y_{clean}），和原始种子语料（X，Y）一起，从头开始训练语法错误纠正模型。由于实际训练语料规模增大，和反向模型相比，多更新 18 000 步，其余训练设置保持不变。

解码时，采用带长度惩罚项的 beam search，beam size 设置为 8，长度惩罚项中的 α 参数设置为 0.6，设置生成的纠正句子最大长度为 300。

2.3　实验结果与分析

表 3 给出了在 CoNLL-2014 Test Set 上的实验结果。由此可见，本文提出的方法能显著提升语法错误纠正的性能。

在使用 back-translation 合成伪“错误-纠正”平行句对时，分别采用 greedy search 和 sampling 解码策略，并用混合语料训练 Transformer。在 $F_{0.5}$ 值上，sampling 的结果比 baseline 高出了约 1.8 个点，比 greedy 的结果高出 6.1 个点。这说明：(1) 基于 sampling 解码策略的 back-translation 数据增强方法，能有效利用外部单语语料，为语法错误纠正模型的训练提供额外的学习信号，从而显著提升模型的性能。(2) 在使用反向模型生成伪错误句子时，采用 sampling 解码策略比 greedy search 解码策略更好。这归因于采用 sampling 解码，能提高生成伪错误句子中的语法错误多样性。

Xie 等人[5] 的工作在方法上和本文最为接近，同样利用 back-translation 方法来合成伪平行句对。而与本文方法的不同之处在于：为了在生成的伪错误句子中引入更多的语法错误多样性，文献[5] 中使用了三种加噪的 beam search 解码策略（rank penalty noising、top penalty noising、random noising）。而本文直接使用 sampling 解码，尽管该方法带来的绝对提升（1.82）有所不及，但本文 baseline 系统的结果比其最好的系统（$\text{base+BT}_{\text{random BS}}$）还要高出 2.18 个点。

表 3 CoNLL-2014 Test Set 实验结果

Tab. 3 Experimental result on CoNLL-2014 Test Set

System	P	R	$F_{0.5}$
baseline	61.21	30.91	51.18
greedy	53.19	31.83	46.90
sampling	**63.17**	**32.23**	**53.00**
base(CharMLConv+LM)	52.7	27.5	44.5
base+BT_{BS}	54.7	29.6	46.8
base+$BT_{rank\ BS}$	54.3	29.3	46.4
base+$BT_{top\ BS}$	50.9	34.7	46.6
base+$BT_{random\ BS}$	54.2	35.4	49.0

3 结 论

本文将机器翻译中的 back-translation 方法应用到语法错误纠正中,充分利用外部单语语料,以缓解该任务面临的数据稀疏问题。首次提出使用 sampling 解码策略来构造伪“错误-纠正”平行句对,并在两个标准数据集上均取得显著提升。进一步的工作将考虑如何利用外部资源中的弱监督信号(如维基百科的编辑历史)来提升语法错误纠正的性能。

参考文献

[1] YUAN Z, BRISCOE T. (2016). Grammatical error correction using neural machine translation. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (pp. 380-386).

[2] CHOLLAMPATT S, NG H T. (2018, April). A multilayer convolutional encoder-decoder neural network for grammatical error correction. In Thirty-Second AAAI Conference on Artificial Intelligence.

[3] BROCKETT C, DOLAN W B, GAMON M. (2006, July). Correcting ESL errors using phrasal SMT techniques. In Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics (pp. 249-256). Association for Computational Linguistics.

[4] FELICE M, YUAN Z. (2014). Generating artificial errors for grammatical error correction. In Proceedings of the Student Research Workshop at the 14th Conference of the European Chapter of the Association for Computational Linguistics (pp. 116-126).

[5] XIE Z, GENTHIAL G, XIE S, NG A, et al. (2018, June). Noising and denoising natural language: Diverse backtranslation for grammar correction. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers) (pp. 619-628).

[6] GE T, WEI F, ZHOU M. (2018, July). Fluency boost learning and inference for neural grammatical error correction. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 1055-1065).

[7] SENNRICH R, HADDOW B, BIRCH A. (2016). Improving Neural Machine Translation Models with Monolingual Data. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (Vol. 1, pp. 86-96).

[8] EDUNOV S, OTT M, AULI M, et al. (2018). Understanding Back-Translation at Scale. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (pp. 489-500).

[9] LI J, MONROE W, SHI T, et al. (2017, September). Adversarial Learning for Neural Dialogue Generation. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (pp. 2157-2169).

[10] DAHLMEIER D, NG H T, WU S M. (2013). Building a large annotated corpus of learner English: The NUS corpus of learner English. In Proceedings of the eighth workshop on innovative use of NLP for building educational applications (pp. 22-31).

[11] MIZUMOTO T, HAYASHIBE Y, KOMACHI M, et al. (2012). The effect of learner corpus size in grammatical error correction of ESL writings. Proceedings of COLING 2012: Posters, 863-872.

[12] NG H T, WU S M, BRISCOE T, et al. (2014). The CoNLL-2014 shared task on grammatical error correction. In Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task (pp. 1-14).

[13] DAHLMEIER D, NG H T. (2012, June). Better evaluation for grammatical error correction. In Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (pp. 568-572). Association for Computational Linguistics.

[14] NG H T, WU S M, WU Y, et al. (2013). The CoNLL-2013 Shared Task on Grammatical Error Correction. In Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task (pp. 1-12).

[15] SENNRICH R, HADDOW B, BIRCH A. (2016). Neural Machine Translation of Rare Words with Subword Units. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (Vol. 1, pp. 1715-1725).

(上接第 186 页)

参考文献

[1] 徐淑琼,陈升平,潘文炜. 基于概率核模糊聚类剪枝的工业机器人在线控制研究[J]. 数字技术与应用,2019,37(9):1-3.

[2] 杨明,张如昊,张军,等. SCARA 四轴机器人控制系统综述[J]. 电气传动,2020,50(1):14-23.

[3] 江楠,司丽娜,蔡增玉. 基于无线网络安全的农业机器人自动避障系统[J]. 农机化研究,2020,42(2):238-242.

[4] 冯迎宾,赵小虎,何震,等. 油浸式变压器内部检测球形机器人的深度悬停控制研究[J]. 控制与决策,2020,35(2):375-381.

[5] 赵轩,王东海,韩宇泽,等. 基于 ZigBee 无线自组网技术的综合管廊机器人控制系统研究[J]. 电子设计工程,2019,27(23):22-26.

[6] 邱素贞,李庆年,卢志翔,等. 基于机器视觉检测的码垛机器人控制系统设计[J]. 包装工程,2019,40(3):207-211.

[7] 徐建明,吴蜀魏,吴小文,等. 基于 ROS 和 IgH EtherCAT 主站的 SCARA 机器人控制系统[J]. 高技术通讯,2019,29(9):876-885.

[8] 刘广志,杨林,郭志愿,等. 基于 WinCE 的经济型机器人控制系统研究[J]. 机械工程与自动化,2019,63(1):157-159,164.

[9] 孟利华,蔡雨欣,闵琴,等. 智能机器人控制系统技术在环境监测中的应用[J]. 科学技术创新,2019,42(33):83-84.

[10] 梁守志,赵虹,陶鑫钰,等. 基于单片机的服务机器人控制系统设计[J]. 轻工科技,2019,35(3):66-69.