融合数据增强与多样化解码的神经机器翻译

张一鸣，刘俊鹏，宋鼎新，黄德根*

（大连理工大学计算机科学与技术学院，辽宁 大连 116024）

摘要：基于神经机器翻译模型 Transformer，提出一种融合数据增强技术和多样化解码策略的方法来提高机器翻译的性能。首先，对训练语料进行预处理和泛化，提高语料质量并缓解词汇稀疏的现象；其次，通过 back-translation 技术构造伪双语数据，扩充双语平行语料以增强模型；最后，在解码阶段融合检查点平均、模型集成和重打分策略以提高译文质量。CCMT2020 中英新闻领域翻译任务的实验结果显示，改进后的方法较 baseline 的 BLEU 值取得了 4.89%的提升。

关键词：神经机器翻译；数据增强；多样化解码

中图分类号：TP 391　　**文献标志码：**A

近年来，随着端到端（End-to-End）结构[1]的提出，神经机器翻译获得了迅速发展。早期的神经机器翻译采用循环神经网络（RNN，Recurrent Neural Network）对句子建模，将源语言的句子压缩成一个向量来供译文生成使用。但传统的循环神经网络在训练时容易发生梯度爆炸等问题[2]，而且对于序列较长的句子而言，无法很好的处理长距离依赖的问题，导致翻译效果较差。长短期记忆网络[3]（LSTM，Long Short-Term Memory），门循环单元[4]（GRU，Gate Recurrent Unit）和注意力机制的引入可以有效捕捉长距离依赖，使得神经机器翻译系统的性能得到显著提升，超越了统计机器翻译方法的性能[5-6]。然而，考虑到循环神经网络训练的不稳定性以及串行执行的低效率，一些高效并行的网络结构被提出。其中，最为常用的模型有两个，分别是基于卷积神经网络（CNN，Convolutional Neural Networks）的 ConvS2S 模型[7]和基于自注意力（Self-Attention）机制的 Transformer 模型[8]。两者可以通过并行训练提高模型的训练效率，更好的解决句子中词与词之间的长距离依赖问题，缩短信息传递的路径，加快训练的收敛速度。相比之下，Transformer 的翻译性能更加优异，目前已成为机器翻译领域的主流模型。

面向 CCMT2020 中英新闻领域的机器翻译任务，本文主要从数据处理、数据增强和多样化解码策略三个方面介绍相关方法和技术。数据处理方面包含了语料清洗、数据泛化、BPE（Byte Pair Encoding）子词切分等方法；在数据增强方面，参考 Sennrich 等[9]和 Zhang 等[10]使用 back-translation 技术增强双语模型；在解码方面，对长度惩罚因子，beam size 等参数进行调优，综合检查点平均（Checkpoint Average）、模型集成（Model Ensemble）、重打分策略进行多样化解码，经过后处理，得到最终的翻译结果。

本文主要从以下 3 个方面改进基线系统：（1）数据泛化。基于规则识别匹配和外部资源对时间表达式、数字、人名等实体进行泛化。（2）数据增强。使用源端单语句子构造伪双语句对，通过长度比、词对齐等筛选条件对语料进行过滤，然后扩充到双语平行语料中提升翻译性能。（3）多样化解码策略。调整长度惩罚因子、beam size 参数，尝试用不同的方式结合检查点平均和模型集成来进行解码，并以 BLEU 值作为评价标准对多个候选译文进行重打分，得到最终译文。

1 Transformer

1. 1 模型结构

Transformer 模型分为编码器和解码器两部分，每一个部分都由 N 个相同层块堆叠而成。编码器的每个网络层包含两个子层：第一层是多头自注意力机制，第二层是一个全连接的前馈神经网络。

基金项目：国家自然科学基金（61672127，U1936109）资助项目

* **通信作者：**huangdg@dlut.edu.cn

而解码器的每个网络层包含三个子层：第一层采用基于掩码技术的多头注意力机制，由于解码阶段只能看到已生成词的信息，因此对未生成词的信息进行屏蔽。第二层是正常的多头注意力机制，第三层是一个全连接的前馈神经网络。为了避免层数过多导致模型难以收敛，编码器和解码器的子层之间都使用残差网络[11]和层级正则化进行连接。

1.2 自注意力机制

Transformer 的自注意力层由缩放点积和多头注意力两部分构成。若是将自注意力机制抽象为一个查询向量 $\boldsymbol{Q}$，一组键向量 $\boldsymbol{K}$ 和一组值向量 $\boldsymbol{V}$ 加权求和的结果，则缩放点积可以表示为式（1）：

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (1)$$

其中d_k是 $\boldsymbol{K}$ 中向量的维度。与传统注意力机制的向量点积运算[12]方法不同，使用$\sqrt{d_k}$进行维度放缩会使训练过程更加稳定。为了更大程度的关注到不同子空间的特征信息，Transformer 使用了多头的注意力网络，将上述的 $\boldsymbol{Q}$，$\boldsymbol{K}$，$\boldsymbol{V}$ 进行均匀的分割，共生成 k 个子块。每个头独立执行缩放点积，最后将所有头生成的结果进行拼接作为最终输出，过程如式（2）和式（3）所示：

$$MultiHead(Q,K,V) = [head_1,\dots,head_k] \quad (2)$$

$$head_i = Attention(QW_i^q, KW_i^k, VW_i^v) \quad (3)$$

其中，W_i^q，W_i^k，W_i^v为参数矩阵。

由于位置信息对于机器翻译过程中语言的理解和生成有着重要作用，因此 Transformer 模型在编码器和解码器的最底层输入向量中引入了位置编码信息。Transformer 借鉴了 ConvS2S 中的位置向量（PE）的概念，采用基于不同频率的三角函数对位置信息进行编码，如式（4）和式（5）所示：

$$PE_{pos,2i} = \sin\left(\frac{pos}{10000^{2i/d}}\right) \quad (4)$$

$$PE_{pos,2i+1} = \cos\left(\frac{pos}{10000^{2i/d}}\right) \quad (5)$$

其中pos表示句子中词所在的位置，i是特征的维度，d是特征维度的大小。利用不用频率的三角函数作为位置向量可以利用三角函数的性质将距离为k的两个词PE_{pos+k}，PE_{pos}表示为线性关系。

2 数据处理

2.1 语料预处理

本次实验使用的中英新闻领域的双语平行语料，中文单语新闻语料和验证集都由 CCMT2020 提供。为了提高数据质量，在训练前对语料采取了一系列的预处理：（1）过滤掉含有乱码的句子；（2）转义字符的处理；（3）全角转半角；（4）语料去重。对于双语语料，使用长度比对双语句子进行过滤。数据处理之后，使用 NiuTrans[13]中提供的分词工具对中英文进行分词。为了精简词表，更好的解决集外词的问题，采用 Sennrich 等提出的 BPE 算法[14][1]分别将中英文词语切分成更小粒度的子词，翻译后再进行恢复。

2.2 语料泛化处理

新闻领域的语料中常包含着大量的命名实体，如人名、地名和机构名，这些命名实体出现的次数较多，但重复率不高，特别以人名为甚。为了缓解词汇稀疏的现象，本次评测对数据进行了泛化处理[15]。实体方面对训练语料进行了人名泛化，在测试阶段对中文单语进行了人名、地名、机构名的泛化。同时，对时间表达式，数字等特殊表达也进行了泛化。采用基于规则的方法对数字、日期、时间表达式进行识别和匹配，然后用“$number”、“$date”和“$time”标签对匹配项进行替换。在一个

[1] https://github.com/resennrich/subword-nmt

句子中通常存在着多个同类泛化成分，为了加以区分并降低恢复难度，在标签后面添加不同的数字编号进行区分，即“$number_i”、“$date_i”、“$time_i”(i=0,1,…,n)。实体方面采用实验室内部开发的中文实体识别工具和 Stanford Corenlp 开源工具[2]分别对中英文实体进行识别。然后，基于中英文人名词典对人名进行识别匹配。初步匹配后，根据大部分中文人名使用汉语拼音作为英文翻译这一特点，综合拼音模糊匹配以及中英文人名首字母音译规律对人名进行再次匹配。用“$name”标签对匹配项进行替换，同样加数字编号进行区分，即“$name_i”(i=0,1,…,n)。受限于外部资源，仅在测试阶段对中文单语增加了地名和机构名的泛化。通过训练集的词频统计，使用中国省份名称的中英文翻译作为标签对两类实体进行泛化，如“北京—beijing”，“天津—tianjin”等。

数据泛化阶段，双语语料的泛化需要保证中英两侧泛化标签的一致性，若存在单侧识别不匹配的情况，则保持原有形式不做处理；单语语料的泛化则需要对所有匹配项进行泛化处理。由于在测试阶段对单语进行了泛化处理，所以解码后的译文中包含泛化标签，根据标签对应关系对泛化部分进行恢复后才能得到最终译文。对于数字、日期、时间表达式来说，统计常用中英文表达的转换规律，根据这些规律编写固定的翻译规则进行恢复；对于人名、地名、机构名来说，使用外部词典进行还原。对人名来说，若词典中无匹配结果，则使用中文人名的拼音作为英文翻译结果。

3 数据增强

3.1 伪双语语料的构建

通常情况下，机器翻译任务主要基于对齐的双语语料进行模型的训练，而对单语语料缺乏足够的关注。Sennrich 等[9]提出了一种使用单语句子构造伪双语句对的数据增强技术（back-translation），可以有效扩充训练语料，提升翻译质量。多数情况下，back-translation 使用目标端的单语数据生成伪双语句对。不同于此，本次评测只采用了源端的单语数据进行实验。为了提高伪双语句对的质量，在预处理的基础上对语料进行再次过滤，包括：（1）特殊符号的过滤；（2）过滤字符长度小于 11 和中文字符长度占比大于 0.5 的句子。（3）以“；”和“。”作为切分点，对长句进行了切割。过滤后，先使用双语平行语料训练一个中文到英文的神经翻译模型，然后使用生成的中英模型将单语中文句子翻译成对应的英文句子，生成初始的伪双语语料。

3.2 伪双语语料过滤

为了保证伪双语语料的质量，基于长度比和词对齐对生成的伪双语语料进行过滤。首先，将长度比限定在 0.4-1.6 的范围内，剔除句子长度差距过大的句对以减少干扰。其次，使用 GIZA++工具[3]对伪双语句对进行词对齐，去掉词对齐比率过低的句子。过滤后，将伪双语语料扩充到双语平行语料中形成新的训练集。融合后训练集的组成部分如表 1 所示，其中关于双语平行语料和伪双语语料的统计均为过滤之后实际用于训练的数量，在此基础上训练数据增强后的中英神经翻译模型。

表 1 双语平行语料和伪双语语料的数量统计

Tab.1 Statistics of bilingual parallel corpus and pseudo-bilingual corpus

语料类型	句子数量
双语平行语料	6.7 M
伪双语语料	6.8 M

[2] http://nlp.stanford.edu/software/stanford-english-corenlp-2018-10-05-models.jar

[3] http://code.google.com/p/giza-pp/downloads/detail？name=giza-pp-v1.0.7.tar.gz

4 解码策略

实验融合检查点平均[16]、模型集成、重打分方法在解码阶段提高译文质量。下面将分别对这 3 个方面进行介绍：

1）检查点平均

检查点平均是指将同一模型在不同时刻保存的参数进行平均。保存的参数通常选择模型基本收敛时对应的最后 N 个时刻的参数，防止引入其他噪声。以同等的权重对 N 个检查点的参数进行平均，得到鲁棒性更强的模型参数。

2）模型集成

模型集成是利用多个机器翻译系统协同进行解码的方法，在神经翻译领域有着广泛的应用[16-17]。集成解码使用的模型可以使用同构或者异构的系统，一般来说，结构和初始化均不同的模型通常更具有差异性，能够带来更大的提升。

3）重打分

解码阶段，通过调整长度惩罚因子和 beam size 参数的设置可以产生多组译文结果。单独计算每一个句子的 BLEU 值时发现，对部分源句子而言，BLEU 值得分最高的译文句子分布在不同的译文结果中，使得单一译文结果无法达到最优。为了缓解这种现象，实验中使用句子级别的 BLEU 值作为评分标准，通过重打分将不同的译文结果中得分最高的句子重组作为最终输出。

通常情况下，当使用目标到源的翻译系统来重构源到目标的翻译结果到源句子时，很难将较差的翻译结果复制到源句子中。根据这个特点，使用目标到源的翻译系统将多个候选译文重构成对应的多个重构句子，然后基于 BLEU 值对这些重构句子进行打分，选取得分最高的译文作为输出。当输入一个源句子进行解码时，首先通过调整参数的方式生成 N 个候选译文，形成候选列表。重构阶段，为了避免单一模型的局限性，可以采用不同的训练方式生成 k 个目标到源的翻译模型。使用这些模型对候选列表中的每一个译文句子进行 k 次重构，生成 k 个重构句子。对每一个译文句子来说，分别计算 k 个重构句子与源句子的 BLEU 值，然后以 k 个翻译模型在验证集上的文档级 BLEU 值分数比作为权重对 k 个 BLEU 值进行加权求和，得到译文句子的评价分数。对比 N 个候选译文的评价分数，选择得分最高的译文句子作为最终输出。其中，单个句子的重打分过程如算法 1 所示:

算法 1: 单个句子的重打分过程

Input: S: source sentence

Output: T: target sentence

Under different parameter settings, decode S to $C = \{C_1,...,C_n\}$，Index_C = {1,...,n}

Train target to source models: $M_1,...,M_k$，Index_M = {1,...,k}

Weight_sum = $\sum_{g=1}^{k}$ Document_level_BLEU(M_g)

for i in Index_C do

    for j in Index_M do

        $L_j = M_j(C_i)$

        Weight_j = Document_level_BLEU(M_j) / Weight_sum

        Value_j = Sentence_level_BLEU(S, L_j)

        Score_i += Weight_j * Value_j

    end for

    Add Score_i to Score_list

end for

Based on the Score_list, select the candidate T with the highest score from C

return T

5 实验结果

5.1 实验参数

本次实验的基线系统使用开源框架 THUMT[4]中提供的 Transformer 模型，实验参数如下：编码器与解码器的层数均为 6 层，词向量与隐层状态维度均为 512，前馈神经网络中的隐层状态维度为 2048，多头注意力机制使用 8 个头。训练阶段中的每个 batch 包含 6250 个 token，模型训练 20 万 steps，每 2000steps 保存一个 checkpoint，并在训练过程中保存最优的 10 个 checkpoint。损失函数使用极大似然估计，并使用 Adam 梯度优化算法，初始学习率为 1.0，warmup 为 4000。训练集双语语料使用 BPE 算法进行切分，中英文词表大小均限制为 32K，且两者不共享词表。解码阶段，使用集束搜索算法和长度惩罚因子对模型进行调优。实验过程中，GPU 方面使用两个 TITAN Xp 进行训练。

模型方面，首先通过随机初始化参数的方式训练了 4 个中英模型，然后选取每个模型中 BLEU 值得分最高的 3 个 checkpoint 进行检查点平均，最后对 4 个平均模型进行模型集成来进行最后的解码。在重打分阶段引入了 4 组不同参数设置下生成的译文结果作为候选项，训练了 3 个不同的英中模型用于对译文结果的重打分。

5.2 实验结果与分析

系统在验证集 newstest2019 上的结果如表 2 所示，评测指标采用大小写不敏感的 BLEU 值，使用 multi-bleu[5]作为评测工具。

表 2 newstest2019 验证集 BLEU 值

Tab.2 BLEU value of newstest2019 validation set

系统	BLEU
baseline	26.11
+synthetic	29.59
+average	30.06
+ensemble	30.70
+reranking	31.00

其中，+synthetic 为加上伪双语语料后的结果，+average 是指经过检查点平均后的结果，+ensemble 是经过模型集成后的结果，+reranking 是重打分后的结果。从实验结果可以看出，在 baseline 的基础上加入 back-translation、检查点平均、集成译码、重打分这些方法对系统 BLEU 值的提高均有帮助。其中，back-translation 提升的效果较为显著，说明单语数据的引入对机器翻译的性能提升有很大帮助。

同时，在实验过程中对以下 3 个方面进行了分析：

1）back-translation 分析

在 back-translation 阶段，基于长度比和词对齐对回译生成的伪双语句对进行了过滤。为了探索语料过滤手段的有效性，分别将过滤前和过滤后的伪双语语料与双语平行语料进行融合生成不同的训练集，然后各自训练生成不同的中英翻译模型。对比分析两个模型在验证集上的表现以验证语料过滤的有效性，结果如表 3 所示：

[4] http://github.com/THUNLP/THUMT

[5] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/multibleu.perl

表 3 伪双语过滤前后 BLEU 值对比

Tab.3 Comparison of BLEU value before and after pseudo-bilingual filtering

系统	BLEU
baseline	26.11
+synthetic_all	28.90
+synthetic_fil	29.59

其中，baseline 为只使用双语语料的基线系统结果，+synthetic_all 是将所有伪双语扩充到训练集时得到的结果，+synthetic_fil 是将过滤后的伪双语扩充到训练集时得到的结果。从表中结果可以看出，伪平行语料的加入对双语模型的提升有着很大帮助。同时，伪双语语料的过滤也能有效的消除语料中的噪声，进一步提高语料质量。

2）长度惩罚因子分析

在调参阶段，尝试在 back-tranlation 实验的基础上探索了不同的长度惩罚因子 α 对实验的影响。首先，将 beam size 设定为 12，然后调整长度惩罚因子的值来进行实验，结果如表 4 所示：

表 4 长度惩罚因子对 BLEU 值的影响

Tab.4 Influence of length normalization on BLEU value

α	1.1	1.3	1.5	1.6	1.8	1.9
beam = 12	29.02	29.28	29.58	29.59	29.22	28.25

随着长度惩罚因子的增加，BLEU 值呈现出先增后减的趋势，说明在一定范围内调整长度惩罚因子会对 BLEU 值的提高有所帮助，过大的长度惩罚因子可能会导致束搜索无法选择正确的结果。

3）beam size 和重打分分析

同时，实验也探索了不同的 beam size 对实验的影响，将长度惩罚因子固定为 1.6，调整不同的 beam size 进行了实验，结果如表 5 所示：

表 5 束搜索大小对 BLEU 值的影响

Tab.5 Influence of beam size on BLEU value

beam size	12	15	20	30
α = 1.6	29.59	29.73	29.76	29.80

随着 beam size 的增加，BLEU 值有所提高。对比 beam size 设置为 12 和 15 时得到的译文结果时发现，beam size 设置为 15 时整体 BLEU 值虽然有所提高，但是会使部分句子的 BLEU 值较 beam size 设置为 12 时变低，导致两种设置下存在不同的高分句子，使得译文结果无法达到最优。由于条件限制，实验过程中只对 2 组参数设置进行了对比分析，具体如表 6 和表 7 所示：

表 6 不同参数对比下各自 BLEU 值得分最高的句子数量统计

Tab.6 Statistics of the number of sentences with the highest BLEU score under different parameter settings

beam size	句子数量
12	121
15	194

表 7 不同译文结果的 BLEU 值分析

Tab.7 BLEU value analysis of different translation results

译文结果	BLEU
Beam-12	29.59
Beam-15	29.73
Beam-mix	29.95

其中，Beam-12 表示 beam size 设置为 12 时得到的译文结果，Beam-15 表示 beam size 设置为 15 时得到的译文结果，Beam-mix 表示将两种参数设置下各自得分最高的译文句子综合在一起后得到的译文结果。当 beam size 的设置从 12 变到 15 时，虽然会使 194 个译文句子的 BLEU 值上升，但也使得 121 个译文句子的 BLEU 值下降。由实验可知，综合不同参数设置下 BLEU 值得分最高的句子生成的译文结果比单一参数设置下译文结果的 BLEU 值有所提升，可以缓解部分句子在参数调整阶段得分降低的情况，提升整体的翻译质量。

6 总结

本文介绍了大连理工大学自然语言处理机器翻译实验室在 CCMT2020 中英新闻领域机器翻译任务上使用的主要方法和技术。使用 Transformer 作为基线系统，从数据处理、数据增强、多样化解码策略三个方面进行了改进。实验融合了包括 back-translation、模型平均、集成解码、重打分等多种技术来提高翻译性能。实验结果显示，这些方法能够明显提高译文质量。

受限于时间和计算资源，还有许多方法有待尝试。通过本次评测，发现了一些问题和不足，翻译模型和系统仍存在很大的提升空间，有待于今后的进一步研究。

参考文献：

[1] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks[C]//Advances in neural information processing systems. 2014: 3104-3112.

[2] Pascanu R, Mikolov T, Bengio Y. On the difficulty of training recurrent neural networks[C]//International conference on machine learning. 2013: 1310-1318.

[3] Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural computation, 1997, 9(8): 1735-1780.

[4] Cho K, Van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[C]//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014: 1724-1734,

[5] Koehn P, Och F J, Marcu D. Statistical phrase-based translation[R]. UNIVERSITY OF SOUTHERN CALIFORNIA MARINA DEL REY INFORMATION SCIENCES INST, 2003.

[6] Chiang D. A hierarchical phrase-based model for statistical machine translation[C]//Proceedings of the 43rd annual meeting of the association for computational linguistics (acl'05). 2005: 263-270.

[7] Gehring J, Auli M, Grangier D, et al. Convolutional sequence to sequence learning[C]//Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, 2017: 1243-1252.

[8] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]//Advances in neural information processing systems. 2017: 5998-6008.

[9] Zhang J, Zong C. Exploiting source-side monolingual data in neural machine translation[C]//Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. 2016: 1535-1545.

[10] Sennrich R, Haddow B, Birch A. Improving Neural Machine Translation Models with Monolingual Data[C]//In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 1.

2016: 86-96.
[11] He K, Zhang X, Ren S, et al. Identity mappings in deep residual networks[C]//European conference on computer vision. Springer, Cham, 2016: 630-645.
[12] Luong T, Pham H, Manning C D. Effective Approaches to Attention-based Neural Machine Translation[C] //Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. 2015: 1412-1421.
[13] Xiao T, Zhu J, Zhang H, et al. NiuTrans: an open source toolkit for phrase-based and syntax-based machine translation[C]//Proceedings of the ACL 2012 System Demonstrations. 2012: 19-24.
[14] Sennrich R, Haddow B, Birch A. Neural Machine Translation of Rare Words with Subword Units[C]//In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2016: 1715-1725.
[15] 刘俊鹏,宋鼎新,张一鸣,等.多种数据泛化策略融合的神经机器翻译系统[J].江西师范大学学报(自然科学版),2020,(01):39-45.
[16] Sennrich R, Haddow B, Birch A. Edinburgh neural machine translation systems for wmt 16[C]//Proceedings of the First Conference on Machine Translation, Volume 2: Shared Task Papers, 2016: 371–376.
[17] Wang Y, Cheng S, Jiang L, et al. Sogou neural machine translation systems for WMT17[C]//Proceedings of the Second Conference on Machine Translation. 2017: 410-415.

Improving Neural Machine Translation with Data Enhancement and Diverse Decoding

ZHANG Yiming, LIU Junpeng, SONG Dingxin, HUANG Degen*

(College of Computer Science and Technology, Dalian University of Technology, Dalian Liaoning 116024, China)

Abstract: Based on the neural machine translation model Transformer, the paper proposes a method to improve the performance of machine translation by fusing data enhancement technology with diverse decoding strategies. Firstly, the training corpus is preprocessed and generalized to improve the quality of the corpus and alleviate the phenomenon of sparse vocabulary. Then, back-translation technology is used to construct pseudo bilingual data, and the model is enhanced by expanding the bilingual parallel corpus. Meanwhile, translation system integrates checkpoint averaging, ensemble, and rescoring in the decoding stage. It improves the quality of translations. Experimental results on CCMT2020 Chinese-English news translation task show that the proposed methods achieve an increase of 4.89% compared to the BLEU value of baseline system.

Keywords: neural machine translation; data enhancement; diverse decoding