

基于 GAN 模型优化的神经机器翻译^①



熊 伟^{1,2}, 高娟娟¹, 刘 锴¹

¹(华北电力大学(保定) 计算机系, 保定 071003)

²(华北电力大学(保定) 复杂能源系统智能计算工程研究中心, 保定 071003)

通信作者: 高娟娟, E-mail: gij18331295138@163.com

摘 要: 针对神经机器翻译和人工翻译性能的差异最小化、训练语料不足问题, 提出了一种基于生成对抗网络的神经机器翻译改进方法. 首先对目标端句子序列添加微小的噪声干扰, 通过编码器还原原始句子形成新的序列; 其次将编码器的处理结果交给判别器和解码器进一步处理, 在训练过程中, 判别器和双语评估基础值 (BLEU) 目标函数用于评估生成的句子, 并将结果反馈给生成器, 引导生成器学习及优化. 实验结果表明, 对比传统的神经机器翻译模型, 基于 GAN 模型的方法极大地提高了模型的泛化能力和翻译的精度.

关键词: 生成对抗网络; 神经机器翻译; Transformer; BLEU; 深度学习

引用格式: 熊伟, 高娟娟, 刘锴. 基于 GAN 模型优化的神经机器翻译. 计算机系统应用, 2022, 31(12): 95-103. <http://www.c-s-a.org.cn/1003-3254/8828.html>

Optimization of Neural Machine Translation with GAN Model

XIONG Wei^{1,2}, GAO Juan-Juan¹, LIU Kai¹

¹(Computer Department, North China Electric Power University (Baoding), Baoding 071003, China)

²(Engineering Research Center of Intelligent Computing for Complex Energy Systems, North China Electric Power University (Baoding), Baoding 071003, China)

Abstract: To minimize the performance difference between neural machine translation (NMT) and human translation and solve the problem of insufficient training corpora, this study proposes an improved NMT method based on the generative adversarial network (GAN). First, the sentence sequence of the target end is added with small noise interference, and then the original sentence is restored by the encoder to form a new sequence. Secondly, the results of the encoder are presented to the discriminator and decoder for further processing. In the training process, the discriminator and the bilingual evaluation understudy (BLEU) objective function are employed to evaluate the generated sentences, and the results are fed back to the generator to instruct its learning and optimization. The experimental results demonstrate that compared with the traditional NMT model, the GAN-based model greatly improves the generalization ability and translation accuracy of the model.

Key words: generative adversarial network (GAN); neural machine translation (NMT); Transformer; bilingual evaluation understudy (BLEU); deep learning

神经机器翻译 (neural machine translation, NMT) 在学术界和工业界引起了越来越多的关注^[1,2]. 与传统的统计机器翻译 (SMT)^[3]相比, NMT 在端到端框架中实现了类似甚至更好的翻译结果^[4]. LSTM/GRU^[5-7]融

合注意机制, 使 NMT 能够更好的翻译长句. Shen 等人^[8]通过采用序列级目标最大化 BLEU, 缓解了最大似然训练的限制. 尽管取得了显著的成功, 但仍存在一些弱点^[9]. 例如, NMT 通常采用极大似然估计原理进行训

① 收稿时间: 2022-03-18; 修改时间: 2022-04-14; 采用时间: 2022-04-29; csa 在线出版时间: 2022-07-25

练,即以源语句为条件,使目标语句的概率最大化.这样的翻译目标并不能保证翻译结果的自然性,准确性. Bahdanau 等人^[10]通过采用序列级来缓解最大似然训练的局限性,尽管有所改进,但这些目标仍然不能完全缩小神经机器翻译和真实人工翻译之间的差距^[11].

基于上述问题,本文采用生成对抗网络来对抗性的训练 NMT 模型^[12].在对抗性 NMT 中,用判别器来区分生成器生成的翻译和人类生成的翻译.同时,生成器试图改进其翻译结果,从而欺骗判别器.生成网络和判别网络协同训练,来相互提高性能^[13].通过学习越来越多的训练样本,可以提高判别器的判别能力,以判别器的输出为奖励,可以指导生成器学习,进而提高生成器的能力.通过这种方式,NMT 翻译结果尽可能接近真实翻译.对抗学习不仅能缩小机器翻译与人工翻译的差距,还能提升神经机器翻译的精度,获得很好的翻译性能.

目前机器翻译(MT)面临着严重的数据稀疏问题^[14],苏依拉等人^[15]针对数据匮乏导致机器翻译质量不理想的问题,提出通过利用大量单语数据使用对偶学习策略的方式进一步提升机器翻译的质量.而使用对目标端序列进行修改的动态数据扩充方法在解决 NMT 泛化能力不足的问题上获得了很好的效果,受此启发,本文还探索融合了一种训练数据的方法-动态数据扩充(DA)^[16].

本文的主要贡献如下.

(1) 提出基于生成对抗网络优化的 NMT 模型,利

用专门设计的基于卷积神经网络(CNN)^[17]的模型进行判别,以(源,目标)句子对作为输入,生成器和判别器进行对抗训练,实验结果表明对抗式 NMT 比传统 NMT 模型具有更大的词汇量和更高的模型复杂度.

(2) 使用 Transformer 模型的编码器和解码器,融合数据扩充方法,对目标端句子进行修改,结合源端语句构造新的平行句对,进而对翻译模型进行训练,较好地提高模型对抗噪声的能力.

1 相关网络模型

1.1 生成对抗网络

近年来,随着人工智能的迅速发展,神经网络生成算法成为研究热点.生成对抗网络在 2014 年最早由 Goodfellow 等人^[18]首次提出,引起很多学者关注.生成对抗网络不断优化扩展,已经成功地应用于各种计算机视觉任务以及文本生成、语音生成等领域^[19].后来由王坤峰等人^[20]对 GAN 做了进一步的研究.在此基础上,Yang 等人^[21]将生抗网络应用到神经机器翻译中,旨在判断翻译是由人工翻译的还是机器翻译的,经过对抗博弈提升机器翻译的精度. Sato 等人^[22]、Mi 等人^[23]最近的研究表明,对抗性训练可以提升机器翻译的性能.崔新明等人^[24]提出一种基于条件生成对抗网络实现情感语音的合成,实验证明利用条件生成对抗模型生成的语音具有与原语音相匹配的情感表达.受此启发,本文引入生成对抗网络.如图 1 所示.

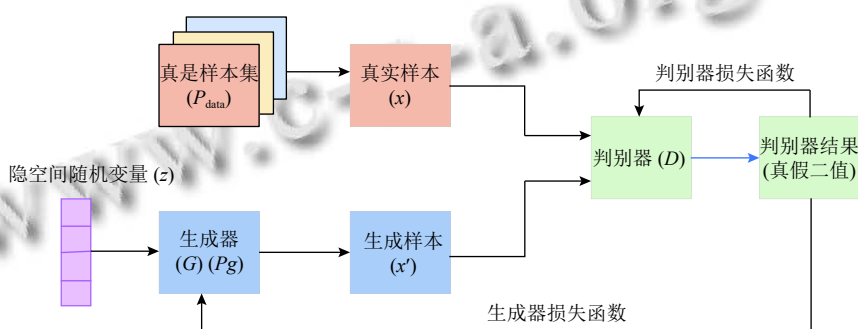


图 1 GAN 的基本框架

如图 1 所示的 GAN 模型,生成器网络,输入(记为 G)是来自隐藏空间的随机变量(记为 z),输出生成样本,其训练目标是为了提高生成样本和真实样本之间的相似性,以便该方法通过鉴别器(记为 D 网络进行区分,即令生成样本(记为 P_g)分布与真实样本(记为

P_{data})分布尽量相同. D 的输入是真实样本(标记为 x)或者生成样本(记录为 x'),输出判别结果用于训练,目的是区分真实样本和生成样本.判别结果用于计算目标函数,通过反向传播更新网络权值.在这个过程中, D 辨别真假样本的能力逐渐增加.而 G 逐渐倾向于生

成样本以达到欺骗 D 的效果. 实验结果表明, 该模型训练很大程度上降低了神经网络计算复杂度.

1.2 神经机器翻译 (NMT)

端到端神经机器翻译 (NMT) 一直是社会各界的研究热点, 到现在端到端神经机器翻译 (NMT) 取得快速发展. 神经机器翻译在多种语言对上已经超过了短语统计机器翻译^[25,26], 在大规模的语料和计算能力条件下, 神经机器翻译表现出巨大潜力, 成为目前研究机器翻译的前沿热点^[27].

基于编码器-解码器结构的神经机器翻译是一种通用模型^[28], 并不完全特定于机器翻译任务本身. 因此, 神经机器翻译还面临翻译句子长度受限、难以充分利用融合大规模单语语料、过度翻译和翻译不充分等问题. 针对上述问题, 对注意力机制改进是神经机器翻译的重大突破^[29-31]. 对比统计机器翻译, 神经机器翻译可以更有效地处理复杂结构和短句子翻译, 但是在翻译的忠实度和长句子的翻译品质上来讲, 神经机器翻译还存在一些问题和挑战. 基于此, 本文研究了基于注意力机制的机器翻译模型, 即使用 Transformer 结构进行神经机器翻译任务. 训练的目标是使模型参数在平行语料 S ($|S|$ 表示平行语料的句子数) 上取得最大似然. 如图 2 所示.

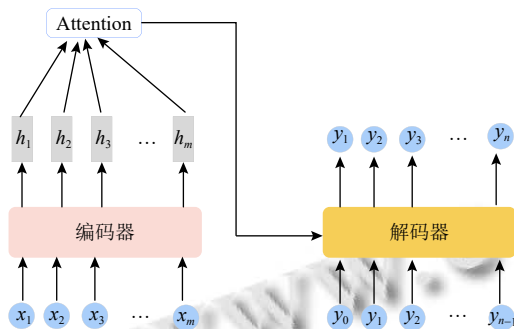


图2 带注意力机制的编码器-解码器模型

编码器首先读取输入序列 $x = (x_1, x_2, \dots, x_m)$, 并将其映射成词向量 $e(x) = [e(x_1), e(x_2), \dots, e(x_m)]$, 然后把这 M 个词向量编码成隐藏状态 h , 根据隐藏状态 h 和目标序列 N 个词的集合 $y = (y_1, y_2, \dots, y_n)$. 解码器从左到右逐个生成目标端单词的概率为:

$$p(y|x; \theta_{mt}) = \prod_{i=1}^N p(y_i | y_{<i}, x, \theta_{mt}) \quad (1)$$

其中, $\theta_{mt} = \{\theta_{enc}, \theta_{dec}\}$ 即为整个模型的参数; θ_{enc} 和 θ_{dec}

分别为编码器和解码器的参数, $y < i$ 表示在预测第 i 个目标单词时已经翻译得到的目标端单词, 给定训练语料库 $S = \{x^{(s)}, y^{(s)}\}_{s=1}^{|S|}$, 该模型在训练集 S 上定义的损失函数为:

$$L(\theta_{mt}) = \frac{1}{|S|} \sum_{(x,y) \in S} -\log P(y|x; \theta_{mt}) \quad (2)$$

1.3 DA

数据动态扩充 (DA) 与文献 [13] 等方法类似. 框架图如图 3 所示, 给定目标端序列 $y = (y_1, y_2, \dots, y_n)$, 本研究首先在目标输入句子中添加噪声化, 对每个输入目标端序列, 随机选择 15% 的单词进行覆盖, 得到带噪声的序列 $y' = (y'_1, y'_2, \dots, y'_n)$, 每句话覆盖的单词数不超过 15. 特别地, 本研究使用 3 种类型的噪声策略来得到目标端序列: 1) 以 10% 的概率替换选中的词; 2) 以 80% 的概率用 [MASK] 覆盖选中的单词; 3) 以 10% 的概率保持单词不变.

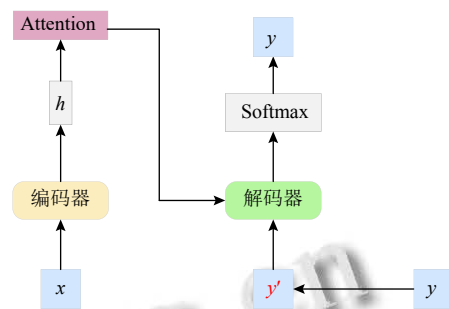


图3 数据动态扩充的 NMT 模型整体框架

假设目标端的输入序列为: “计算机系”“学生”“向”“领导”“提出”“诚恳”“建议”. 在构造带有噪声的目标端序列时分别用以上 3 个噪声策略的替换, 如表 1 所示.

表1 噪声策略示例

替换方式	示例
10%的概率替换为其他单词	“诚恳”替换为“中肯”
80%的概率替换为[MASK]	“学生”替换为[MASK]
10%的概率保持单词不变	“建议”替换为“建议”

加入噪声后, 得到的最终噪声输入序列为: 计算机系 [MASK] 向领导提出中肯建议.

对目标端序列添加噪声得到序列 $y' = (y'_1, y'_2, \dots, y'_n)$, 加噪的目的是让编码器学会提取语义特征, 将加噪的目标端句子序列解码恢复到原始句子. 通过最大化条件概率 $P(y|h, y'; \theta_{dec})$ 来训练模型获得目标端序列.

$$P(y|h, y'; \theta_{dec}) = \prod_{i=1}^n P(y_i|h, y' < i; \theta_{dec}) \quad (3)$$

通过解码器从左到右逐一生成目标词, 最终得到完整的译文 y , 该模型在每个伪平行句对 (x, y') 定义的损失函数为:

$$Loss(\theta_{mt}) = \sum_{i=1}^n -\log P(y_i|x, y' < i; \theta_{mt}) \quad (4)$$

在种子语料 (x, y) 上训练好机器翻译模型后, 用一定的策略对目标端语句 y 随机噪声化, 得到含有噪声的目标端语句 y' , 进而构建伪平行语料 (y', y) .

2 神经机器翻译模型

2.1 GAN-NMT-DA

将生成对抗网络应用于神经机器翻译中, 基于此对网络进行优化, 使用数据扩充方法, 在每次加载一批训练数据时, 通过不同的覆盖、替换等操作随机修改

句子中的单词, 得到新的目标语句和源端语句构成新的平行句对进行训练, 让 y' 表示从 NMT 系统中源句子 x 翻译出来的句子. 对抗性的神经机器翻译的目标是使 y' 与 y 相似. 生成模型 G 试图产生一个与人工翻译相似的句子来欺骗判别模型, 判别模型 D 的目标是区分翻译来自人工翻译还是机器翻译, 在训练的过程中, G 和 D 互相斗争, 判别器和 BLEU 值一起评估生成的句子, 产生的奖励信号用来指导生成器学习. 网络整体框架图如图 4 所示.

2.2 生成模型 G

对抗网络基于两个模型, 一个生成模型, 一个判别模型. 生成器 G 的目标是在源句的基础上生成与人类翻译不可区分的目标语言句子. 为了提高翻译效率, 本文的生成模型采用 Transformer 结构, 该模型在提高翻译速度的同时也提升了机器翻译的性能. Transformer 结构如图 5 所示.

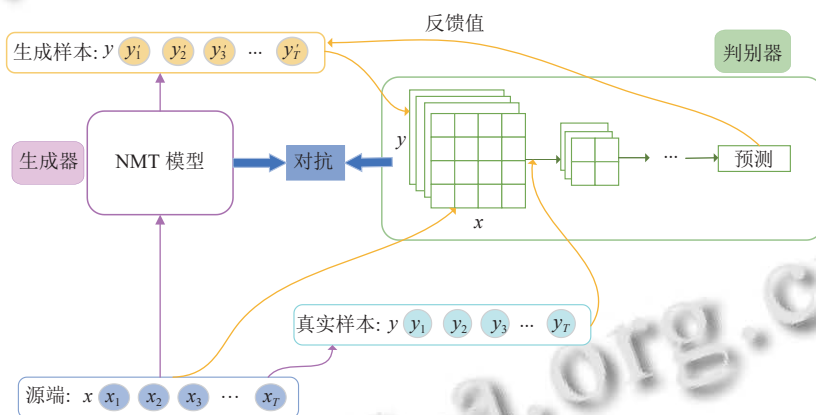


图 4 网络整体架构图

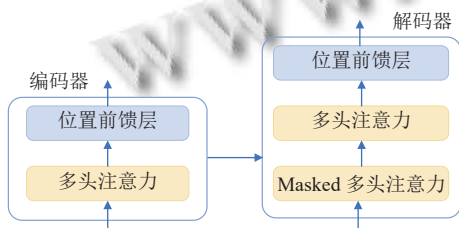


图 5 Transformer 结构图

每个编码器根据需要可以由多个相同的编码器构成. 每个编码器层又由两个环节组成, 分别是: 多头注意力和位置前馈层. 每个解码器也是由相同数量的解码器构成, 每个解码器层则是由 3 个环节构成, 分别是:

Masked 多头注意力, 多头注意力, 和位置前馈层. 在编码器和解码器的各个层之中, 核心就是其中不同的注意力机制和权重矩阵.

对于编码器而言: 不同的句子, 首先要做的就是将词向量输入到模型之中, 这就是词嵌入层获取源语种和目标语种的词向量. 由于采用的不是顺序输入的网络结构, 因此, 在词向量进入模型之前要先进行词位置的标注, 位置标注公式如下:

$$PE_{(pos, 2i)} = \sin(pos/10000^{2i/d_{model}}) \quad (5)$$

$$PE_{(pos, 2i+1)} = \cos(pos/10000^{2i/d_{model}}) \quad (6)$$

其中, d_{model} 是词向量的长度.

通过这两个公式与词向量相加得到带有位置信息的词向量. 接下来, 将带有位置信息的词向量输入到多头注意力层中, 得到每个单词的自注意力分数, 如图6所示.

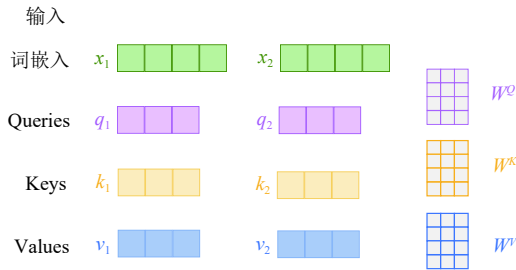


图6 self-attention 计算过程

图6中, x_1, x_2 代表的是输入的词向量, W^Q, W^K, W^V 分别代表3个权重矩阵, 且 $W \in R^{d_{\text{model}} \times V}$ (V 代表词表的长度).

将 x_i 分别与3个权重矩阵相乘得到分向量:

$$q_i = x_i \cdot W^Q, k_i = x_i \cdot W^K, v_i = x_i \cdot W^V \quad (7)$$

然后根据式(4)计算两个单词之间的相似度得分:

$$e_{ij} = \frac{1}{\sqrt{d_{\text{model}}}} q_i k_j^T \quad (8)$$

对各个单词之间的相似度得分应用 Softmax 函数:

$$a_{ij} = \frac{\exp(e_{ij})}{\sum_{i=1}^T \exp(e_{il})} \quad (9)$$

最后计算注意力得分, 即 self-attention 的得分:

$$Z_i = \sum_{j=1}^T a_{ij} v_j \quad (10)$$

其中, Z_i 是对应每个单词的注意力得分, 接下来将注意力得分输入到 feed-forward networks 中进行非线性, 采用激活函数 ReLU:

$$FFN(z_i) = \max(0, z_i W_{L1} + b) W_{L2} + b_2 \quad (11)$$

矩阵 $W_{L1}, W_{L2} \in R^{d_{\text{model}} \times V}$, 维度和词向量相同, b_1 和 b_2 是偏差. FFN层的输出作为解码器环节的输入.

2.3 判别模型 D

鉴别器 D 以源语言句子为条件, 试图区分机器生成的句子和人工翻译的句子. 因为 D 与 G 同步更新, 所以 D 可以看作是一个动态目标. 由于生成模型生成的序列长度是长短不一的, 所以先将输出的序列扩展到固定长度 T , 根据源语言序列和目标序列分别构建源矩阵 $X_{1:T}$ 和目标矩阵 $Y_{1:T}$, 公式如式(12):

$$\begin{cases} X_{1:T} = x_1; x_2; x_3; \dots; x_T \\ Y_{1:T} = y_1; y_2; y_3; \dots; y_T \end{cases} \quad (12)$$

其中, $x_T, y_T \in R^k$ 是维度为 k 的词嵌入向量; 源矩阵 $X_{1:T}, w_j \in R^{L \times k}$ 是一个长度为 L 的卷积核. 对长度为 L 的卷积核进行卷积运算, 产生一系列特征映射, 其卷积公式为:

$$C_{ji} = f(w_j \otimes X_{i:i+L-1} + b) \quad (13)$$

其中, w_j 是卷积核的权值, $X_{i:i+L-1}$ 为从 i 到 $i+L-1$ 窗口内词向量矩阵, \otimes 为元素间的卷积; b 为偏置项, f 为非线性激活函数, 本文采用线性整流函数 (ReLU) 激活函数.

采用不同数量的核函数和不同的窗口大小来提取特征, 再将每个特征向量的最大值拼接在一起, 组成源序列的特征向量 C_x . 对每一个卷积核的特征映射进行最大值池化 (max pooling), 得到最终特征结果为:

$$C_x = \max\{c_{j1}, c_{j2}, \dots, c_{jT-L+1}\} \quad (14)$$

用上述方法同样地从源矩阵中提取目标序列特征向量 C_y . 通过 C_x 和 C_y 计算目标语句是真样本的概率, 计算公式如下:

$$P = \Phi(V[C_x; C_y]) \quad (15)$$

其中, V 是变换矩阵, 将 C_x 和 C_y 连接成一个二维词向量, Φ 是 Softmax 函数.

2.4 评估反馈

双语评估基础值 BLEU 作为评测方法之一, 将生成模型和判别模型合并进行 GAN 迭代训练. 预训练生成模型后, 在验证集上用 BLEU 对生成的译文进行评估. 当整个 GAN 训练结束后, 再次在验证集上用更新后的模型对译文进行评估. 此时, 两个 BLEU 得分相比, 可以得到 GAN 在优化后的生成模型上起的作用. 本文用质量评估 BLEU 值作为强化目标, 来指导生成器生成更高的 BLEU 值.

2.5 生成对抗网络模型中的对抗过程

生成对抗网络是一种基于无监督的学习方式, 它的参数更新基于连续的空间, 但是自然语言处理的数据是离散的, 因此引入了强化学习的策略梯度概念. 将强化学习和生成对抗网络相结合, 具体地, 生成模型视为强化学习的主体, 判别器用来评估序列, 并反馈评估结果, 以指导生成模型的学习. 同时模型里的参数更新借助梯度策略算法来实现. 在模型的梯度策略中, 使用蒙特卡洛树搜索 (Monte Carlo tree search, MCTS) 来实

现对不完整序列的实时评估。

具体的训练过程如下: 给定源端输入句子, 生成器的目的是从生成序列的起始状态, 生成类似于人工翻译的语句, 使最后期望的结果达到最大。形式上, 生成器的目标函数如式 (16) 所示:

$$J(\theta) = \sum_{y_{1:T}} G_{\theta}(y_{1:T}|x) \cdot R_D^{G_{\theta}}(y_{1:T-1}, x, y_T) \quad (16)$$

其中, θ 表示生成器 G 的参数, $y_{1:T}$ 表示生成器生成的句子, x 表示输入的源端句子。 $R_D^{G_{\theta}}$ 表示动作值函数, 具体含义为, 从状态 $y_{1:T-1}$ 开始, 对生成的反馈进行处理。使用判别器输出句对 $(x, y_{1:T})$ 为人工标注的概率表示动作值函数 $R_D^{G_{\theta}}$ 的值。

$$R_D^{G_{\theta}}(y_{1:T-1}, x, y_T) = D(x, y_{1:T}) - b(x, y_{1:T}) \quad (17)$$

其中, $b(x, y_{1:T})$ 是为了减少奖励值方差的基线值, 为了简单计算, 将 $b(x, y_{1:T})$ 设置为常数 0.5。

然而, 判别器只能为完整的序列提供奖励值, 对于 $D(x, y_{1:T})$ 计算来的结果没有明确的含义, 所以无法评估中间状态的作用值。因此, 需要依据策略梯度算法 G_{θ} , 在生成的序列不是完整的序列时, 通过蒙特卡洛搜索生成后续完整的单词序列。当采样到句子结束或者从初始状态 (之前的 $T-1$ 时刻生成的序列) 到生成完整序列达到预期设定的最大长度时, 表示本次搜索结束。为了减少奖励值的方差, N 次蒙特卡洛搜索的设置如式 (18) 所示:

$$\{y_{1:T_1}^1, \dots, y_{1:T_N}^N\} = MC^{G_{\theta}}((y_{1:t}, x), N) \quad (18)$$

其中, T_i 表示第 i 次蒙特卡洛搜索时的序列长度, $(y_{1:t}, x)$ 表示当前状态, $\{y_{1:T_1}^1, \dots, y_{1:T_N}^N\}$ 表示依据策略 G_{θ} 生成的 N 种有可能的单词序列, 并对这所有的 N 个序列取平均, 就可以求得整个句子序列的最终反馈函数。

$$R_D^{G_{\theta}}(y_{1:t-1}, x, y_t) = \begin{cases} D(x, y_{1:t}) - b(x, y_{1:t}), & t = T \\ \frac{1}{N} \sum_{n=1}^N (D(x, y_{1:T_n}^n) - b(x, y_{1:T_n}^n)), & t < T \end{cases} \quad (19)$$

在对抗训练阶段, 使用判别器作为奖励函数来激励生成器学习, 进一步提升生成器的性能。一旦生成器获得更新, 模型将重新训练判别器, 并令其损失函数为最小。

$$\min = E_{x,y \in p_{\text{data}}} [\log D(x,y)] - E_{x,y \in G_{\theta}} [\log(1 - D(x,y))] \quad (20)$$

判别器更新完之后, 重新训练生成器。目标函数

$J(\theta)$ 对生成器参数的计算如式 (21) 所示:

$$\begin{aligned} \nabla J(\theta) &= \sum_{t=1}^T \sum_{y_t} R_D^{G_{\theta}}(y_{1:t-1}, x, y_t) \cdot \nabla_{\theta} (G_{\theta}(y_t|y_{1:t-1}, x)) \\ &= \sum_{t=1}^T (R_D^{G_{\theta}}(y_{1:t-1}, x, y_t) \cdot \nabla_{\theta} \log(G_{\theta}(y_t|y_{1:t-1}, x))) \end{aligned} \quad (21)$$

对抗训练过程简单来说, 生成网络和判别网络是交替更新的。生成网络 G_{θ} 通过借助强化学习中的策略梯度方法和利用判别网络 D 得到的蒙特卡洛搜索的奖励值来实现生成器的更新, 使得生成器生成和人工翻译相近的句子达到迷惑判别器的效果。判别网络 D 借助人工翻译的真实数据和生成网络生成的数据来共同训练判别器, 以提高判别能力, 进而达到对抗的效果。对抗学习过程的训练算法如算法 1 所示。

算法 1. 对抗学习过程中的训练算法

描述: 生成器 G_{θ} , 判别器 D , 机器翻译平行语料 (X, Y)

1. 在 (X, Y) 上使用极大似然估计预训练 G_{θ}
2. 使用生成器 G_{θ} 生成负样本 (X, Y)
3. 将 (X, Y) 作为正样本, 和负样本 (X, Y) 一起预训练判别器 D
4. for 生成器训练轮次 do
5. 给定源端句子 X , 经过生成器 G_{θ} 生成目标端句子 Y'
6. 使用判别器 D 计算负样本的奖励值
7. 通过策略梯度均衡更新生成器参数
8. end for
9. for 判别器训练轮次 do
10. 给定源端句子 X , 经过生成器 G_{θ} 生成目标端句子 Y'
11. 用正样本 (X, Y) 和负样本 (X, Y') 训练判别器 D
12. end for

3 测试与分析

3.1 数据集

对于 English-German 数据, 数据集采用的是斯坦福自然语言处理实验室公开的英-德双语翻译数据集, 该数据集包括训练集, 测试集, 验证集。从 WMT14 En-De 中随机选择了 100 万条英-德对照句子对作为训练集。选取 newstest2014 作为测试集, 选取 newstest2013 作为验证集。

对于 Chinese-English 数据, 训练数据由 1.25 万个句对组成, 其中 27.9 万个中文单词和 34.5 万个英文单词。选取 NIST 2002, 2003 数据集作为测试集, 选取 NIST 2006 数据集作为验证集。

3.2 实验设置

实验基于 OpenNMT, CPU 为 Intel Core i7-9700U

@4.20 GHz×4, 内存 16 GB. 使用 Python 3.5 编程, 在 PyTorch 框架下使用 Python 语言, 本文采用的是 Transformer 模型架构, 编码器解码器都有 6 层的网络结构, 编码器和解码器的隐藏状态维度设置为 512. 实验采用 Adam 方法对网络进行优化, 将单词嵌入维度设置为 100, 学习率初始值设置为 0.005, 为了防止训练过程中出现过拟合的情况, 引入 dropout 机制, 并且将其设置为 0.3, epoch 设置为 60.

3.3 实验结果分析

1) 模型损失: 本文实验使用最大似然估计损失函数进行训练, 得到的训练损失与验证损失如图 7 所示. 途中蓝色曲线为模型的训练损失, 橘色曲线为验证损失.

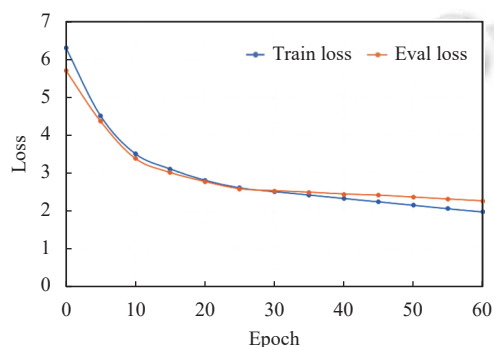


图 7 损失函数曲线

图 7 反映了数据迭代次数与损失值变化情况的关系. 可以看出, 随着迭代次数的增加, 两个模型的损失值均逐渐减小, 迭代到 25 轮左右时, 验证集的损失值减速放缓, 而训练集的损失值仍有下降空间. 为了防止迭代次数过多导致模型过拟合, 停止对模型继续迭代训练.

2) 数据集英德翻译、中英翻译结果分别如表 2、表 3 所示.

表 2 English-German BLEU

系统	Tst2013	Tst2014	平均值
RNNsearch	21.57	22.29	21.93
Transformer-base	27.85	28.34	28.09
Transformer-big	28.20	28.72	28.46
GAN+Transformer	28.95	29.31	29.13
GAN+Transformer+DA	31.32	32.40	31.86

为验证提出的 GAN-NMT-DA 方法的有效性, 在英德数据集上分别进行 5 组实验, 表 2 给出了英德翻译的结果. 可以看出, 与传统的 RNNsearch 模型相比,

Transformer 方法在优化 NMT 训练目标方面取得了更好的效果, 使机器翻译的结果和人类翻译的真实翻译结果相似, 然而 Transformer 模型没有考虑语言相关联系. GAN+Transformer 模型引入了生成对抗网络, 考虑了语言的相关性, 缓解了训练不足的问题, 使得翻译更加准确. 上述方法都是用源端数据进行训练, 然而, NMT 比其他模型需要更多的数据, 这使得方法更难适应低资源情况. 本文所提 GAN+Transformer+DA 方法, 通过动态的数据扩充, 在目标端通过不同的覆盖、替换等操作随机修改句子中的单词, 得到新的目标句子, 然后和源端语句构成新的平行句对, 对翻译模型进行训练; 通过约束解码器重构原始目标语句, 提高模型对抗噪声的能力. 实验结果表明, 提出的模型改善了翻译性能, 在数据扩充 DA 方法的基础上, 分别提升了 2.37 和 3.09 的 BLEU 值, 表现较好的性能, 同时提升了机器翻译的精度.

表 3 Chinese-English BLEU

模型	NIST06	NIST02	NIST03	平均值
RNNsearch	33.24	35.76	33.93	34.31
RNNsearch+GAN	36.72	39.01	37.66	37.79
Transformer	45.66	46.89	46.04	46.20
GAN+Transformer	46.87	47.95	46.48	47.10
GAN+Transformer+DA	48.16	49.56	49.19	48.97

对于 Chinese-English, 表 3 显示了翻译结果, 可以看出在 RNNSearch 模型上引入 GAN, RNNSearch+GAN 在中英测试集上 BLEU 平均提高了 3.48. Transformer 基准系统实现了翻译性能提升. 我们发现, 在 Transformer 中引入对抗式训练可以带来进一步的改进. 与 Transformer 模型相比, 加入 GAN 的 GAN+Transformer 结构系统在测试集上 BLEU 提高了 0.90. 在使用 DA 方法后, 效果更佳. GAN+Transformer+DA 的效果大幅度提升, 方法在中英翻译方面相比之前的方法 BLEU 提高了 1.87. 综上所述, 本文提出的模型训练准确率不断提高, 模型的学习能力和泛化能力更强.

4 总结与展望

论文提出了将 GAN 思想引入机器翻译中, 融合态数据扩充方法, 通过在目标端构造干扰噪声, 把添加了干扰噪声的序列作为解码器的输入, 约束编码器还原原始句子, 让生成器和判别器对抗训练, 提高模型的语言表达能力和鲁棒性. 在英-德, 中-英翻译任务上, 方法

都获得了较好的 BLEU 值的提升,证明了 DA 技术增强训练数据的合理性。

本文的方法还适用于语言稀疏和多语言翻译任务上^[32],在接下来的工作中将会逐一尝试。GAN 方法目前已经广泛应用于图像处理,以及文本分类相关的领域。未来将继续进行生成对抗网络在自然语言处理领域的研究,让机器真正的理解自然语言,进一步提升机器翻译。

参考文献

- 1 Kulkarni P, Bhalariao P, Nayek K, *et al.* Trends and advances in neural machine translation. Proceedings of 2020 IEEE International Conference for Innovation in Technology (INOCON). Bangluru: IEEE, 2020. 1–6.
- 2 冯洋, 邵晨泽. 神经机器翻译前沿综述. 中文信息学报, 2020, 34(7): 1–18. [doi: 10.3969/j.issn.1003-0077.2020.07.001]
- 3 Singh TD, Hujon AV. Low resource and domain specific English to Khasi SMT and NMT systems. Proceedings of 2020 International Conference on Computational Performance Evaluation (ComPE). Shillong: IEEE, 2020. 733–737.
- 4 Poncelas A, Popovic M, Shterionov DS, *et al.* Combining SMT and NMT back-translated data for efficient NMT. Proceedings of the Recent Advances in Natural Language Processing. Varna, 2019. 922–931.
- 5 Mai SJ, Xing SL, Hu HF. Analyzing multimodal sentiment via acoustic- and visual-LSTM with channel-aware temporal convolution network. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2021, 29: 1424–1437. [doi: 10.1109/TASLP.2021.3068598]
- 6 Zhao F, Zhang T, Wu Y, *et al.* Antidecay LSTM for siamese tracking with adversarial learning. IEEE Transactions on Neural Networks and Learning Systems, 2021, 32: 4475–4489. [doi: 10.1109/TNNLS.2020.3018025]
- 7 Zhang B, Xiong DY, Xie J, *et al.* Neural machine translation with GRU-gated attention model. IEEE Transactions on Neural Networks and Learning Systems, 2020, 31: 4688–4698. [doi: 10.1109/TNNLS.2019.2957276]
- 8 Shen SQ, Cheng Y, He ZJ, *et al.* Minimum risk training for neural machine translation. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. Berlin: ACL, 2016. 1683–1692.
- 9 Koehn P, Knowles R. Six challenges for neural machine translation. Proceedings of the 1st Workshop on Neural Machine Translation. Vancouver: ACL, 2017. 28–39.
- 10 Bahdanau D, Brakel P, Xu K, *et al.* An actor-critic algorithm for sequence prediction. Proceedings of the 5th International Conference on Learning Representations. Toulon: OpenReview.net, 2017.
- 11 Wu LJ, Xia YC, Tian F, *et al.* Adversarial neural machine translation. Proceedings of the 10th Asian Conference on Machine Learning. Beijing: PMLR, 2018. 534–549.
- 12 Wang DL, Gong CY, Liu Q. Improving neural language modeling via adversarial training. Proceedings of the 36th International Conference on Machine Learning (ICML). Long Beach: PMLR, 2019. 6555–6565.
- 13 明玉琴, 夏添, 彭艳兵. 基于 GAN 模型优化的神经机器翻译. 中文信息学报, 2020, 34(4): 47–54. [doi: 10.3969/j.issn.1003-0077.2020.04.006]
- 14 侯强, 侯瑞丽. 机器翻译方法研究与发展综述. 计算机工程与应用, 2019, 55(10): 30–35, 66. [doi: 10.3778/j.issn.1002-8331.1810-0063]
- 15 苏依拉, 王昊, 贺玉玺, 等. 基于对抗学习的蒙汉神经机器翻译. 计算机系统应用, 2022, 31(1): 249–258. [doi: 10.15888/j.cnki.csa.008283]
- 16 刘志东, 李军辉, 贡正仙. 一种简单的神经机器翻译的动态数据扩充方法. 厦门大学学报(自然科学版), 2021, 60(4): 680–686.
- 17 Roy SK, Krishna G, Dubey SR, *et al.* HybridSN: Exploring 3-D–2-D CNN feature hierarchy for hyperspectral image classification. IEEE Geoscience and Remote Sensing Letters, 2020, 17(2): 277–281. [doi: 10.1109/LGRS.2019.2918719]
- 18 Goodfellow IJ, Pouget-Abadie J, Mirza M, *et al.* Generative adversarial nets. Proceedings of the 27th International Conference on Neural Information Processing Systems. Montreal: MIT Press, 2014. 2672–2680.
- 19 邹秀芳, 朱定局. 生成对抗网络研究综述. 计算机系统应用, 2019, 28(11): 1–9. [doi: 10.15888/j.cnki.csa.007156]
- 20 王坤峰, 苟超, 段艳杰, 等. 生成式对抗网络 GAN 的研究进展与展望. 自动化学报, 2017, 43(3): 321–332. [doi: 10.16383/j.aas.2017.y000003]
- 21 Yang Z, Chen W, Wang F, *et al.* Improving neural machine translation with conditional sequence generative adversarial nets. Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. New Orleans: ACL, 2018. 1346–1355.
- 22 Sato M, Suzuki J, Kiyono S. Effective adversarial regularization for neural machine translation. Proceeding of the 57th Annual Meeting of the Association for Computational Linguistics. Italy: ACL, 2019. 204–210.

- 23 Mi CG, Xie L, Zhang YN. Improving adversarial neural machine translation for morphologically rich language. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2020, 4(4): 417–426. [doi: [10.1109/TETCI.2019.2960546](https://doi.org/10.1109/TETCI.2019.2960546)]
- 24 崔新明, 贾宁, 周洁美慧. 基于条件生成式对抗网络的情感语音生成模型. *计算机系统应用*, 2022, 31(1): 322–326. [doi: [10.15888/j.cnki.csa.008246](https://doi.org/10.15888/j.cnki.csa.008246)]
- 25 Bentivogli L, Bisazza A, Cettolo M, *et al.* Neural versus phrase-based machine translation quality: A case study. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin: ACL, 2016. 257–267.
- 26 Nishimura Y, Sudoh K, Neubig G, *et al.* Multi-source neural machine translation with missing data. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2019, 28: 569–580.
- 27 Sharma S, Diwakar M, Singh P, *et al.* A review of neural machine translation based on deep learning techniques. *Proceedings of 2021 IEEE 8th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*. Dehradun: IEEE, 2021. 1–5.
- 28 孙晓骞, 苏依拉, 赵亚平, 等. 基于编码器-解码器重构框架的蒙汉神经机器翻译. *计算机应用与软件*, 2020, 37(4): 150–155, 163. [doi: [10.3969/j.issn.1000-386x.2020.04.025](https://doi.org/10.3969/j.issn.1000-386x.2020.04.025)]
- 29 Vaswani A, Shazeer N, Parmar N, *et al.* Attention is all you need. *Proceedings of the 31st International Conference on Neural Information Processing Systems*. Long Beach, 2017. 6000–6010.
- 30 Zhang B, Xiong DY, Su JS. Neural machine translation with deep attention. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020, 42(1): 154–163. [doi: [10.1109/TPAMI.2018.2876404](https://doi.org/10.1109/TPAMI.2018.2876404)]
- 31 Li XT, Liu LM, Tu ZP, *et al.* Attending from foresight: A novel attention mechanism for neural machine translation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2021, 29: 2606–2616. [doi: [10.1109/TASLP.2021.3097939](https://doi.org/10.1109/TASLP.2021.3097939)]
- 32 Hung JW, Lin JR, Zhuang LY. The evaluation study of the deep learning model transformer in speech translation. *Proceedings of 2021 7th International Conference on Applied System Innovation (ICASI)*. Chiayi: IEEE, 2021. 30–33.

(校对责编: 牛欣悦)