计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2023,32(1):348−357 [doi: 10.15888/j.cnki.csa.008910]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于生成对抗网络的文本生成图像算法①

段亚茹, 赵嘉雨, 何立明

(长安大学 信息工程学院, 西安 710018)
通信作者: 段亚茹, E-mail: dyrjiayou2022@163.com

摘　要: 文本生成图像算法对生成图像的质量和文本匹配度有很高的要求. 为了提高生成图像的清晰度, 在现有算法的基础上改进生成对抗网络模型. 加入动态记忆网络、细节校正模块 (DCM)、文本图像仿射组合模块 (ACM) 来提高生成图片的质量. 其中动态记忆网络可以细化模糊图像并选择重要的文本信息存储, 以提高下一阶段生成图像的质量. DCM 纠正细节, 完成合成图像中缺失部分. ACM 编码原始图像特征, 重建与文本描述无关的部分. 改进后的模型实现了两个目标, 一是根据给定文本生成高质量的图片, 同时保留与文本无关的内容. 二是使生成图像不再较大程度依赖于初始图像的生成质量. 通过在 CUB-200-2011 鸟类数据集进行研究实验, 结果表明相较之前的算法模型, FID (Frechet inception) 有了显著的改善, 结果由 16.09 变为 10.40. 证明了算法的可行性和先进性.

关键词: 生成对抗网络; 细节校正; 动态记忆网络; 文本描述生成图像; 文本图像仿射组合模块; 图像生成; 深度学习

引用格式: 段亚茹,赵嘉雨,何立明.基于生成对抗网络的文本生成图像算法.计算机系统应用,2023,32(1):348–357. http://www.c-s-a.org.cn/1003-3254/8910.html

Text-to-image Algorithm Based on Generative Adversarial Network

DUAN Ya-Ru, ZHAO Jia-Yu, HE Li-Ming

(School of Information Engineering, Chang'an University, Xi'an 710018, China)

Abstract: Text-to-image algorithm requires high image quality and text matching. In order to improve the clarity of generated images, a generative adversarial network model is improved based on existing algorithms. Dynamic memory network, detail correction module (DCM), and text image affine combination module (ACM) are added to improve the quality of generated images. Specifically, the dynamic memory network can refine fuzzy images and select important text information storage to improve the quality of images generated in the next stage. DCM corrects details and repairs missing parts of composite images. ACM encodes original image features and reconstructs parts irrelevant to the text description. The improved model achieves two goals. On the one hand, high-quality images are generated according to given texts, with contents that are irrelevant to the texts preserved. Second, generated images do not greatly rely on the quality of initial images. Through experiments on the CUB-200-2011 bird data set, the results show that compared with previous algorithm models, the Frechet inception (FID) has been significantly improved, and the result has changed from 16.09 to 10.40, which proves that the algorithm is feasible and advanced.

Key words: generative adversarial network (GAN); detail correction; dynamic memory network; text to image; text image affine combination module; image generation; deep learning

文本到图像的生成是近年来非常热门的研究. 其主要任务就是根据文本描述生成相应图像. 主要研究方法有变分自编码器 (variational auto-encoder, VAE) 以及生成对抗网络 (generative adversarial network,

① 收稿时间: 2022-05-11; 修改时间: 2022-06-15; 采用时间: 2022-07-18; csa 在线出版时间: 2022-09-08
CNKI 网络首发时间: 2022-11-15

GAN) 等. 其中 GAN 由于其独特的优势成为现在比较受欢迎的研究方法.

文本到图像合成要求合成的图像在逼真的基础上还要在语义层面符合文本描述. 所以训练模型之前要进行自然语言处理[1−5] 得到指示图像生成的文本特征. 之后通过 GAN 网络不断优化结果, 生成与文本匹配的图像. 文本生成图像的任务相较于其他图像合成任务, 例如图像到图像的风格转换[6]、文本问答[7]、标签的生成[8] 等挑战性更大. 一是文本描述包含更多的信息, 比标签的语义更复杂, 对合成的图像要求更高; 另外文本到图像的转换是跨模态的, 这要比图到图的风格迁移任务[9] 更复杂. 从另一个角度说, 文本生成图像其实就是文本分析和图像生成[10] 问题的结合.

为了解决上述难题相关学者不断改进 GAN 模型, 使文本到图像的生成任务取得了巨大的进展. 在多阶段的堆叠结构提出之前, GAN 模型可以根据文本生成粗略的分辨率不高的图像, 缺少必要细节. 2017 年在 StackGAN 中 Zhang 等人[11] 使用多个生成器和判别器结合, 分阶段生成图像, 提高了图像的清晰度. 在此基础上, 出现了很多改进模型提高生成图像的分辨率. 2018 年 Xu 等人提出的 AttnGAN 模型[12], 增加了跨模态的注意力机制, 首次证明可以根据词级别信息从句子中提取内容来生成图像的不同部分. 2019 年 Li 等人提出的 ControlGAN 模型[13], 加入空间注意力和通道注意力机制分辨不同属性, 实现通过操纵文本描述控制图像对应部分生成的目的. 虽然这些方法推动了文本图像生成领域的进步, 但依然存在着不可忽视的问题. 一是在分阶段生成图像时, 第 2 阶段生成的图像的质量会受到第 1 阶段的影响. 另外, 输入句子的每个单词都表明不同的重要信息, 这些模型在不同的图像细化过程中使用相同的单词表示, 没有考虑每个词对细化的重要性. 所以 2020 年 Chen 等人提出了 DM-GAN 模型[14], 加入动态内存模块存储重要信息, 在初始图像生成不佳时提高模糊图像的质量, 还利用一个响应门来自适应地融合从记忆和图像特征中读取的信息. 上述方法都并没有考虑生成图像的与文本无关的部分, 不能有效地识别其内容. 2020 年 Li 等人针对这个问题提出了 ManiGAN 模型[15], 加入了对原始图像进行编码重建与文本无关的内容的文本图像仿射组合模块 (ACM 结构), 以及可以纠正文本-图像不匹配信息, 完成缺失内容的细节校正模块 (DCM 结构). 但是此模型生成的图片清晰度不是很高, 生成图片的质量不尽如人意. 为了解决上述问题, 本文提出 DM-ManiGAN 模型. DM-ManiGAN 模型产生的图片如图 1 所示.

图 1 DM-ManiGAN 模型产生的图片

1 相关任务

1.1 生成对抗网络

生成对抗网络 (GAN)[16] 对于生成模型的发展有重大的意义. GAN 网络结构灵活, 对生成数据的维度没有限制, 基于这些优点其模型结构被广泛应用. GAN 采用对抗的方式训练, 两个神经网络通过反向传播不断优化各自训练结果. 训练过程简单, 生成过程自由, 生成图片效率高. 而且 GAN 采样步骤简单, 可以直接对新样本采样, 减少了生成样本的时间.

GAN 具有一个生成器网络和一个判别器网络[17], 其训练可以被表述为一个双人游戏, 互相博弈. 生成器生成的图像与真实图像一起被判别器判断, 使判别器产生能区分“真假图像”的能力, 同时生成器也要优化提升, 生成更趋近真实样本的图像. 最后二者在理想状态下可以到达一个平衡——纳什平衡, 表明这个网络已经训练成功, 此时生成器的结果最优. 其模型结构如图 2 所示.

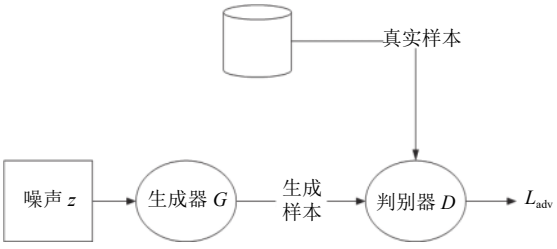

图 2 GAN 基本模型架构

生成器网络输入随机变量 z, 输出生成样本, 其训练的目标是生成与真实的图像更为相似的结果. 判别器网络输入真实样本和生成样本, 目标是分辨出其输入样本是生成的还是真实的. 判别器的结果用于计算

损失函数, 之后通过梯度更新, 不断优化生成结果.

损失函数L_{adv}如下所示:

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}}[\log D(x)] + E_{z\sim p_z}[\log(1-D(G(z)))] \quad (1)$$

在生成器G保持不变时, 最大化判别器D的判别结果, 使其能辨别出真实样本与生成样本, 即, 使得D对真实图片样本的概率$D(x)$趋近于 1, 对生成的图片样本概率预测$D(G(z))$趋近于 0. 之后, 保持判别器D不变, 让生成器G能最小化目标函数, 使其产生的样本与真实图像之间的差异最小化, 由式 (1) 可知加号之前的部分与生成器无关, 那么只考虑加号之后的部分, 让判别器D预测生成器产生的样本$D(G(z))$的概率趋近于 1. 通过上述训练, 在理想的情况下, 生成的数据分布无限接近于真实的数据.

1.2 有条件的生成对抗网络模型

有条件的生成对抗网络模型[18] 通过输入文本描述来生成图像. 就其原理来说是在上述 GAN 的基础上进行改进, 其基本模型结构如图 3 所示.

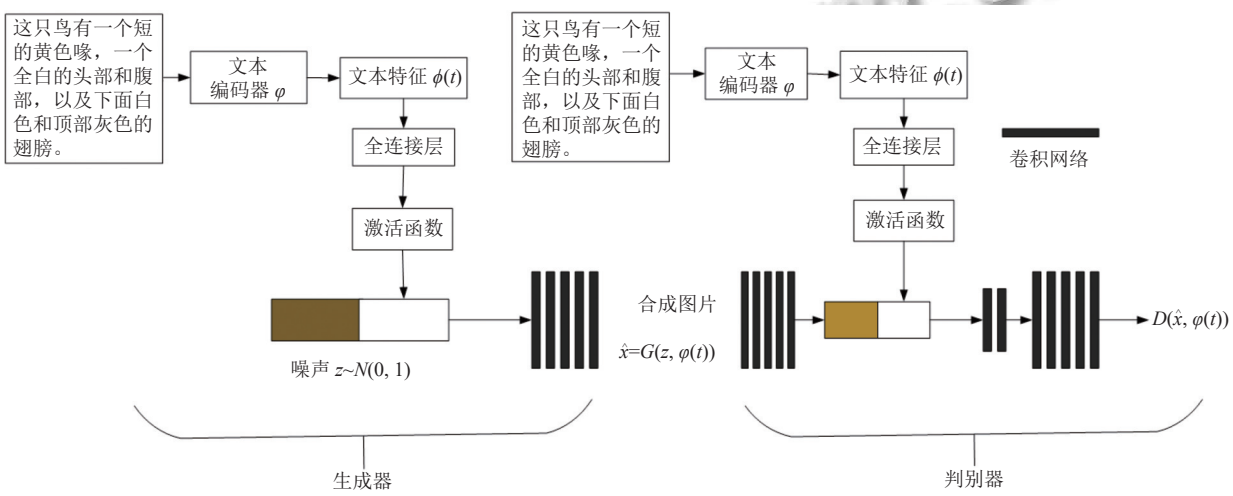

图 3 条件生成对抗网络模型

图 3 中文本编码器φ对文本编码得到的$\varphi(t)$经过全连接层压缩后与随机向量z连接, 输入生成网络. 判别器接收生成器生成的图片和文本向量, 输出判别结果.

1.3 文本特征的提取

Radford 等人首次证明了条件生成网络 (DC-GAN)[18] 可以根据文本描述生成接近真实的图像. Zhang 等人[11] 提出的 StackGAN 堆叠了几个 GAN, 利用不同阶段生成不同分辨率的图像. 上述模型都是使用全局句子向量与噪声向量z拼接后经卷积神经网络上采样来生成图像. 这种对整句文本进行编码生成的图像中的细粒度信息会被忽略. 这些细粒度信息由词层面的语义信息决定, 如颜色. Xu 等人[12] 在 AttnGAN 中提出了一种新的文本特征提取结构—注意力模型 (DAMSM). 这个结构可以把单词内容与绘制样本的区域进行高度匹配. DAMSM 通过同时训练不同的神经网络来计算图像生成的细粒度损失. 首先使用双向 LSTM 编码提取文本的词特征向量和句子特征向量; 其次使用图像编码器衡量词语与图像是否匹配. 本文对文本特征的提取将会使用同样的方式. 同时还加入了动态记忆网络模型, 通过动态记忆网络[19] 之间的重要转换来生成高质量的图像.

2 DM-ManiGAN 模型

本文模型在初始图像生成阶段, 通过一个文本编码器将输入的文本描述转换为句向量s和单词向量w. 然后, 根据句子向量和随机噪声向量预测具有大致形状比较粗糙的初始图像I_0. 其中, 随机噪声服从正态分布z, $R_0 = G_0(z,s)$, R_0是图像特征. 第 2 个阶段, 在初始图像中添加更细致的特征, 生成逼真的图像I_i: $I_i, R_i = G_i(R_{i-1}, w)$, 其中$R_{i-1}$是上一阶段的图像特征. 细化阶段可以重复多次, 以检索更相关的信息, 并生成具有细粒度更细节的高分辨率图像. 模型结构如图 4 所示.

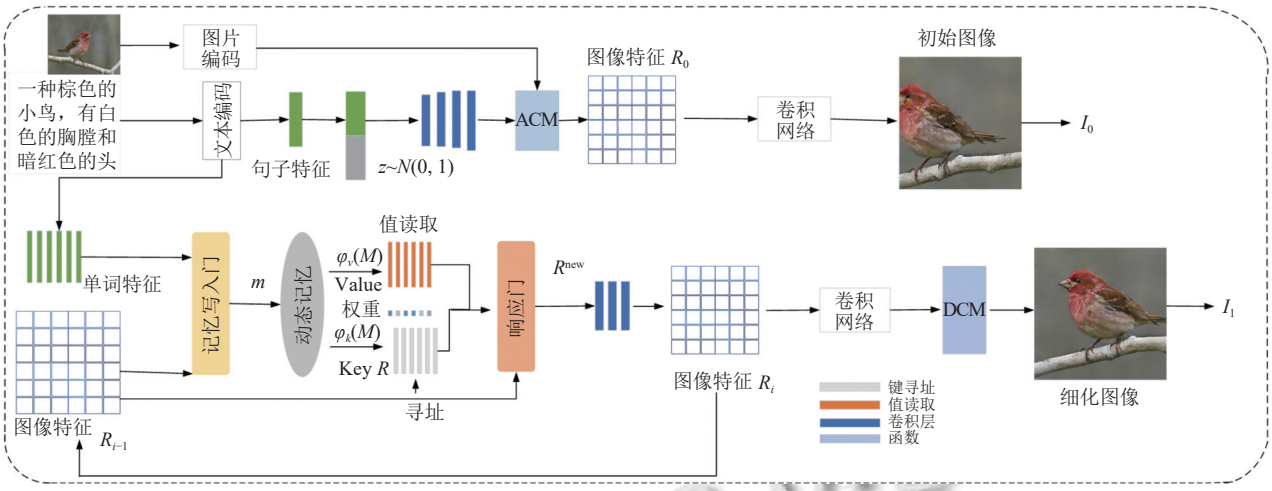

图 4 DM-ManiGAN 模型架构

2.1 动态记忆网络

输入单词特征w和图像特征R_{i-1}:

$$\begin{cases} W=\{w_1,w_2,w_3,\cdots,w_T\},\ w_i\in R^{N_w} \\ R_i=\{r_1,r_2,r_3,\cdots,r_N\},\ r_i\in R^{N_r} \end{cases} \tag{2}$$

其中, T为单词的数量, N_w是单词特征向量的维度; N是图片的像素数, 其向量维度是N_r. 此过程的目的是使用更有效的方法融合图像和文本信息. 基于动态内存的图像细化阶段由 4 个部分组成: 内存写入、键寻址、值读取、响应.

算法 1. 初始图像生成算法

输入: 原始图片和文本信息
输出: 粗糙图像

步骤 1. 通过一个文本编码器将输入的文本描述转换为句向量s和单词向量w.
步骤 2. 单词特征将会成为下一阶段的输入. 句子向量经过条件增强与随机噪声向量拼接.
步骤 3. 拼接结果经过全连接层与句子特征一起输入文本图像仿射组合模块 (ACM), 得到图片特征R_0.
步骤 4. R_0经过 3×3 的卷积网络生成粗糙图像.

算法 2. 图像细化算法

输入: 上一阶段生成的粗糙图像
输出: 细化之后的图像

步骤 1. 经过动态记忆网络, 融合图像文本信息.
(1) 内存写入: 选择相关单词优化细节. 得到单词特征与图像子区域之间的相关概率.
(2) 键寻址、值读取: 通过 (1) 得到的相关概率计算每个子区域占的权重o_j.
(3) 自适应门根据o_j利用自适应门控制图像特征的更新, 得到新的图像特征R^{new}.
步骤 2. 新的图像特征经过上采样和两个残差模块得到最终图像特征R_i.
上述步骤 1 和步骤 2 可以多次执行.
步骤 3. R_i经过 3×3 的卷积网络后输入 DCM 网络增强合成图像中的细节, 补充缺失内容. 最终输出细化之后的图像.

2.2 门控记忆写入

记忆写入门细化初始图像, 筛选重要单词优化细节. g_i^w结合上一阶段的图像特征R_i和单词特征w计算每个单词的重要性:

$$g_i^w(R,w_i)=Sigmoid\left(A\times w_i+B\times\frac{1}{N}\sum_{i=1}^{N}r_i\right) \tag{3}$$

其中, A 是$1\times N_w$的矩阵, B 是$1\times N_r$的矩阵.

$$m_i=M_w(w_i)\times g_i^w+M_r\left(\frac{1}{N}\sum_{i=1}^{N}r_i\right)\times\left(1-g_i^w\right) \tag{4}$$

其中, $M_w(\cdot)$和$M_r(\cdot)$表示 1×1 的卷积, 作用是进行维度转换, 使单词特征和图像特征嵌入到相同特征空间. m_i表示记忆特征空间, 是单词特征w与图像子区域r_i之间的相关性.

2.3 键寻址、值读取

根据式 (5) 计算得到第i个记忆空间和第j个子区域间的相似概率:

$$\alpha_{ij}=\frac{\exp(\phi_k(m_i)^T r_j)}{\sum_{l=1}^{T}\exp(\phi_k(m_l)^T r_j)} \tag{5}$$

根据式 (6) 计算每一个图像子区域r_j的注意力权重.

$$o_j=\sum_{i=1}^{T}\alpha_{ij}\phi_v(m_i) \tag{6}$$

式 (5) 和式 (6) 的ϕ_k、ϕ_v都是 1×1 的卷积.

2.4 自适应门

利用上述得到的权重o_j, 更新图像特征:

$$\begin{cases} g_i^r = Sigmoid(W[o_j, r_j] + b) \\ r_i^{\text{new}} = o_j \times g_i^r + r_i \times (1 - g_i^r) \end{cases} \tag{7}$$

其中, g_i^r为信息融合的响应门, $[\cdot,\cdot]$是连接操作, W和b分别为参数矩阵和偏置项.

2.5 文本图像仿射组合模块 (ACM)

ACM 作用是融合文本图像跨模态表示. 如图 5 所示, 文本特征经卷积层后得到隐藏特征h. ACM 进一步将h与原始区域图像特征r有效结合, 选定图像样本中与文本相关区域, 这部分将会与文本信息关联. 另外 ACM 还对原始图像表示进行编码, 以进行稳定重建. 模型结构如图 5 所示.

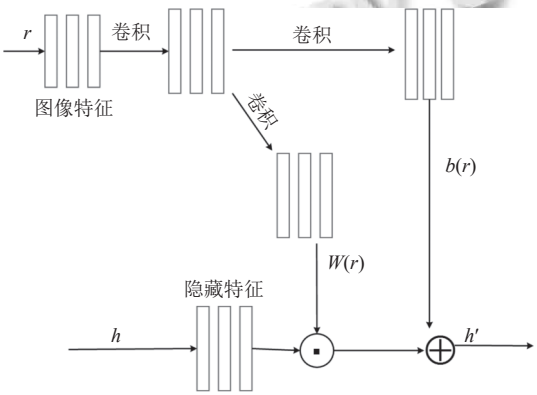

图 5 ACM 模型

公式如下:

$$h' = h \odot W(r) + b(r) \tag{8}$$

其中, $\odot$表示对应元素相乘, 作用是使文本特征重新对图像特征进行加权, 在选择子区域时, 帮助模型精确识别与给定文本匹配的属性, 同时建立子区域与语义词之间的相关性; h表示隐藏特征, $h \in \mathbb{R}^{C\times H\times D}$, 其中$C$、$H$、$D$分别代表通道数、特征图的高度和宽度; v是图像特征, $v \in \mathbb{R}^{256\times 17\times 17}$; $b(r)$记录与文本无关特征, 为图像的完整生成提供帮助. $W(r)$和$b(r)$是v经过上采样之后用 3×3 的卷积进一步处理得到. 总而言之, $W(r)$和$b(r)$将输入图像编码为具有语义信息的特征.

2.6 细节校正模块 (DCM)

该模块利用了单词级别特征与图像信息结合. 如图 6 所示.

图 6 中h_{last}是图像特征R_i经过卷积网络的输出. 空间和通道注意力特征$s \in \mathbb{R}^{C'\times H'\times D'}$和$c \in \mathbb{R}^{C'\times H'\times D'}$, 分别与$h_{\text{last}}$连接产生中间特征$a$, a有助于细化与文本有关的视觉属性. 为了从输入样本引入详细的视觉特征利用预训练的VGG网络提取图像特征v', v'经过上采样之后与特征a通过 ACM 模块融合为$\tilde{a}$. 最后, 用两个残差模块对$\tilde{a}$进行细化, 并再一次将图像特征v'与细化结果通过 ACM 模块, 添加文本描述缺失部分的图像, 生成最终的图像.

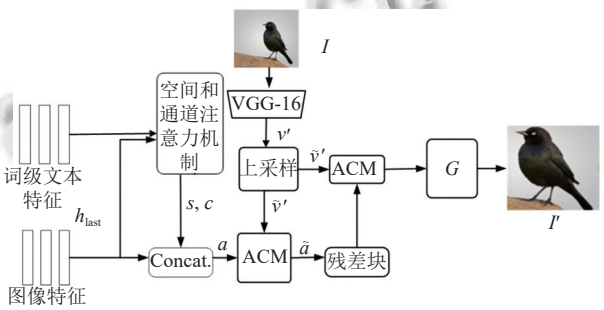

图 6 DCM 模型

DCM 模型中的空间注意力机制和通道注意力机制介绍如下.

空间注意力的本质就是定位目标并进行一些变换或者获取权重在文献 [12] 中有详细说明.

通道注意力机制计算过程如图 7 所示. 其中, H_k, W_k是h_{last}的高度和宽度. 单词特征w首先通过感知层F_k映射到与视觉特征h_{last}相同的特征空间, 产生$\tilde{w}_k$, 其中$F_k \in \mathbb{R}^{(H_k\times W_k)\times D}$. 转换后的单词特征$\tilde{w}_k$与视觉特征$h_{\text{last}}$相乘, 计算出通道注意力矩阵$m^k \in \mathbb{R}^{C\times L}$, 表示为$m^k = \tilde{w}_k h_{\text{last}}$. 因此, m^k是所有空间位置的通道和单词之间的相关性值. 接下来, 将m^k通过 Softmax 函数进行归一化, 生成归一化的通道注意矩阵α^k:

$$\alpha_{i,j}^k = \frac{\exp(m_{i,j}^k)}{\sum_{l=0}^{L-1} \exp(m_{i,l}^k)} \tag{9}$$

注意力权重$\alpha_{i,j}^k$表示视觉特征h_{last}中的第 i 个通道与第 j 个单词之间的相关性, 值越高表示相关性越大. 利用注意矩阵α^k, 得到最终的通道注意特征$f_k^\alpha \in \mathbb{R}^{C\times(H_k\times W_k)}$, 记为$f_k^\alpha = \alpha^k(\tilde{w}_k)^T$. f_k^α中的每个通道都由单词和视觉特征中相应通道之间的相关性加权. 因此, 相关值高的通道被增强, 可以对相应单词产生高响应. 通过产生较低的相关性来减少不相关内容的影响.

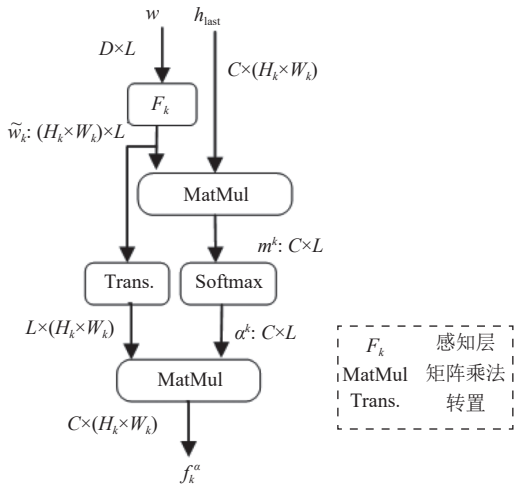

图 7 通道注意力机制计算流程图

2.7 目标函数

生成网络的目标函数定义为:

$$L=\sum_i L_{G_i}+\lambda_1 L_{CA}+\lambda_2 L_{\mathrm{DAMSM}}+\lambda_3 L_p \tag{10}$$

其中, λ_1、λ_2和λ_3分别为条件反射增强损失、DAMSM 损失和感知损失的相应权值. G_0为初始阶段的生成器. G_i表示图像细化阶段的迭代i次的生成器.

2.7.1 对抗性损失

G_i的对抗损失定义如下:

$$L_{G_i}=-\frac{1}{2}\left[E_{x\sim P_{G_i}}\log D_i(x)+E_{x\sim P_{G_i}}\log D_i(x,s)\right] \tag{11}$$

其中, 第 1 项是无条件损失, 使生成的图像尽可能真实, 第 2 项是使图像与输入句子匹配所产生的条件损失. 或对每个鉴别器D_i的对抗性损失也可以定义为:

$$L_{D_i}=-\frac{1}{2}\left[L_{Di1}+L_{i2}\right] \tag{12}$$

其中, $L_{D_{i1}}$为无条件损失, 计算公式为:

$$L_{Di1}=E_{x\sim P_{\mathrm{data}}}\log D_i(x)+E_{x\sim P_{G_i}}\log(1-D_i(x)) \tag{13}$$

其中, $L_{D_{i2}}$为条件损失, 计算公式为:

$$L_{Di2}=E_{x\sim P_{\mathrm{data}}}\log D_i(x,s)+E_{x\sim P_{G_i}}\log(1-D_i(x,s)) \tag{14}$$

无条件损失被用来区分生成的图像和真实的图像, 条件损失决定了图像和输入句子的匹配度.

2.7.2 条件增强损失

通过从一个独立的高斯分布中重新采样输入句子向量来增强训练数据, 避免过拟合. 因此, 条件增强损失表示的是 Kullback-Leibler (KL)散度:

$$L_{CA}=D_{\mathrm{KL}}\left(N\left(\mu(s),\sum(s)\right)\|N(0,I)\right) \tag{15}$$

其中, $\mu(s)$和$\sum(s)$为句子特征的均值和对角线协方差矩阵. $\mu(s)$和$\sum(s)$由全连接层计算. $N\left(\mu(s),\sum(s)\right)$是条件高斯分布, $N(0,I)$是标准高斯分布.

2.7.3 DAMSM 损失和感知损失

DAMSM 损失可以衡量文本与生成图像间的相关性, 促使生成的图像更接近于文本描述. 计算方式与文献 [12] 一致.

感知损失在一定程度限制了生成结果. 如果不对与文本无关的区域 (如背景) 添加任何约束, 生成的结果可能是高度随机的, 也可能无法与语义上的其他内容一致. 为了减轻这种不可靠性, 使用 VGG 预训练网络提取图像特征信息. 训练此网络的数据集是 ImageNet 数据集. 从生成图像I'和真实图像I中分别提取语义向量定义其损失为:

$$L_p(I',I)=\frac{1}{C_iH_iW_i}\|\varphi_i(I')-\varphi_i(I)\|_2^2 \tag{16}$$

其中, H_i是特征向量的高度, W_i是宽度, $\varphi_i(I)$为 VGG 网络第 i 层的激活值.

目标函数 (10) 中的λ_1、λ_2和λ_3是控制不同损失的超参数.

3 实验与结果分析

文中初始图像生成阶段首先合成分辨率为 64×64 的图像. 然后, 图像细化阶段将图像细化到 128×128 和 256×256 的分辨率. 由于 GPU 内存的限制, 实验只对动态内存模块重复细化过程两次. 对低分辨率图像 (即 16×16、32×32) 引入动态内存并不能进一步提高性能. 为了提高文本生成图像的质量每次在判别器卷积之后都会使用谱归一化, 用来避免梯度异常. 默认情况下, 将N_w=256、N_r=64 和N_m=128 分别设置为文本、图像和记忆特征向量的维数. 实验中超参数的设置参考了基础模型 DMGAN, 损失函数的超参数λ_1=1、λ_2 = 5、λ_3=1. 生成网络和对抗网络的学习率都设置为 0.000 2. 上述模型所有的网络结构都是 ADAM 优化器. 分阶段合成图像的平滑指数分别为*GAMMA*1=4.0, *GAMMA*2=5.0, *GAMMA*3=10.0, *LAMBDA*=5.0. 训练模型的*batch_size*= 10, *epochs*= 800. DCM 模块与主模块分开进行训练. DCM 模块训练时的参数设置与主模块一致.

本文在 CUB-200-2011 鸟类数据集进行研究实验, 验证方法的可行性与优越性. CUB-200-2011 鸟类数据集包含了 11 788 张 200 种鸟类的图片. 其中 8 855 张

图片用于训练, 2 933 张图片用于测试. 每张图片形态各异, 并且每一个都有相应的 10 个文本描述. 数据使用之前会对进行预处理. 最终有 3 万张生成图像来进行评估.

本文的评估方式采用了FID (Frechet inception) 和 IS (inception score)[20].

FID度量的是真实样本与生成样本的特征向量的距离. 根据 Inception v3 图像分类模型进行计算. 分数越低则表明生成样本越接近真实图像.

IS使用预先训练的 Inception v3网络来计算条件分布和边缘分布之间的散度. 直接针对生成图像, 指标值越大说明生成的图像越接近真实图像.

3.1 实验步骤

本文实验步骤分为 3 步, 即预训练、图片生成以及测试生成图像质量.

(1) 预训练阶段是本文的重点内容, 通过分阶段训练不同的子任务. 获取子任务中文本与图像之间的关系.

阶段 1. 使用双向 LSTM 文本编码器对文本进行编码, 每个单词的特征都对应着两个方向的隐藏状态. 句子向量是根据词向量生成. 由于训练过程与文献 [12] 相同所以直接使用其预训练好的模型进行实验. 必须首先对文本编码, 因为这是之后模型训练的基础.

阶段 2. 运行 main.py 函数训练 DM-ManiGAN 的主模型. 运行的 batch_size=10, max_epoch= 800. 训练完成将模型保存. 这部分训练结果就是没有加 DCM 模块的模型该模型也可以用来生成图片, 但是效果不太好, 将会在第 3.3 节进行对比.

阶段 3. 运行 DCM.py 函数训练 DCM 模型. batch_size=10, max_epoch= 500. 其余参数设置与主模型一致. 运行模型保存.

(2) 图片生成: 训练的全部模型路径写入 eval_bird.yml 文件, 此文件记录了代码运行所需要的全部参数. 在 main.py 中加载 eval_bird.yml 模型参数, 生成图像.

(3) 测试生成图像质量: 将生成的图片与原始图片输入 fid_score.py 和 IS.py 中分别得到生成图片的 FID 和 IS 值.

算法 3. 训练 DM-ManiGAN 的主模型部分参数设置

```
TRAIN:
    BATCH_SIZE: 10
    MAX_EPOCH: 800
SNAPSHOT_INTERVAL: 10
DISCRIMINATOR_LR: 0.0002
GENERATOR_LR: 0.0002
NET_E: 'text_encoder200.pth'
SMOOTH:
GAMMA1: 4.0
GAMMA2: 5.0
GAMMA3: 10.0
LAMBDA: 5.0
GAN:
DF_DIM: 32
GF_DIM: 64
Z_DIM: 100
R_NUM: 2
TEXT:
EMBEDDING_DIM: 256
CAPTIONS_PER_IMAGE: 10
```

3.2 实验结果

本文利用此数据集对其他模型进行复现, 产生的结果与 DM-ManiGAN 模型的结果进行比较. 由表 1 可知, 本文方法在 CUB 数据集上的评价指标 IS 由 4.75 提高到 5.37, FID 由 16.09 下降到 10.40. 结果表明相较于其余方法本文的方法的确要更优. 也就是本文模型相对于基础模型 DMGAN 生成的图像更接近真实图像, 图像内容更加丰富, 质量也有所提高.

表 1 不同模型生成图片的评价指标结果

模型	FID↓	IS↑
AttnGAN	23.98	4.36±0.3
ControlGAN	30.72	4.58±0.09
DMGAN	16.09	4.75±0.7
ManiGAN	24.29	4.79±0.7
本文	10.40	5.37±0.2

3.3 消融试验

只有动态记忆网络, 动态记忆网络+ACM, 与动态记忆网络+ACM+DCM 实验得出结果进行比较, 结果如表 2 所示, 可以看出加入 DCM、ACM 模型的确是可以提高生成图像的质量. 并且加入 DCM 的网络结构 IS 更高也就是图片更加清晰, 因为其纠正了细节部分使得生成图片展现的内容质量更高; 加入 ACM 的模型 FID 更高也就是与原始图片更为相似, 因为重建了与文本无关内容.

记忆网络+ACM 产生的图片进行比较如图 8 所示, 可以看出来大多数情况下加入 ACM 的网络即使对于文本描述的鸟类的生成结果不理想也可以生成无关的背景图.

记忆网络+DCM 产生图片结果如图 9 所示. 对于

文本描述的部分可以生成以外还看到细节部分也得到了很好的纠正, 各种鸟类的花纹可以清晰的生成. 但是生成图像的完整性不足, 也就是对于背景图大多数情况不能很好地实现.

表 2 消融试验结果对比图

模型	FID↓	IS↑
动态记忆网络	35.72	3.27±0.09
动态记忆网络+DCM	30.71	5.27±0.3
动态记忆网络+ACM	15.46	4.24±0.7
动态记忆网络+ACM+DCM	10.40	5.37±0.2

图 8 消融试验, 记忆网络+ACM 可视化图

图 10 显示了 AttnGAN、ControlGAN、DMGAN、ManiGAN 以及本文模型 DM-ManiGAN 之间的可视化比较. 由图 10 可知, 本文的方法的确在文本生成图像时有一定优越性, 生成结果质量有一定的提高. 在合成图像和文本描述之间保持高度的语义一致性, 同时也保留了与文本无关的区域, 例如背景, 站立姿势等. 并且图片中的鸟类清晰度更高, 不论是站在树枝上还是铁丝上, 不论是转头还是正面各种动作都可以生成, 具有更好的视觉效果.

图 9 消融试验, 记忆网络+DCM 可视化图

图 10 AttnGAN、ControlGAN、DMGAN、ManiGAN 以及本文 DM-ManiGAN 模型的可视化

4 结束语

针对文本生成图像存在的无关区域不能很好的重建, 生成样本受第 1 阶段生成样本质量影响以及生成图像质量不高的问题, 本文提出了新的模型 DM-ManiGAN. DM-ManiGAN 大多数情况下生成的图片都具有较高的清晰度和更生动的动作. 通过动态记忆模块弥补了视觉内容与自然语言处理之间的差距; 通过 ACM, DCM 结构重建了文本描述缺失的内容. 实验结果表明了本文方法在图像生成方面的有效性和产生高质量图像的优越性. 虽然本文的方式产生了比较清晰的图片, 但是还是有一些问题存在, 比如生成图片的多样性不足; 产生重影等, 以后的研究任务将会沿着这个方向展开.

算法 4. DM-ManiGAN 算法

for epoch in range(start_epoch, max_epoch) do:
  while (step < num_batches) do:
    (1) 准备训练数据并计算单词向量值
    (2) 生成图像
      阶段 1 生成图像 : $z, R_0 \leftarrow G_0(z,s)$
      阶段 2 生成图像 : $I_i, R_i \leftarrow G_i(R_{i-1}, w)$
    (3) 更新判别器网络
      for i in range(len(netsD)) do: // len(netsD)←3
        errD(判别器损失) $L_{D_i} \leftarrow -\frac{1}{2}\left[L_{D_{i1}}+L_{i2}\right]$
      $L_{D_{i1}}=E_{x\sim P_{\text{data}}}\log D_i(x)+E_{x\sim P_{G_i}}\log(1-D_i(x))$
      $L_{D_{i2}}=E_{x\sim P_{\text{data}}}\log D_i(x,s)+E_{x\sim P_{G_i}}\log(1-D_i(x,s))$
      (4) 更新生成器网络: maximize log($D(G(z)$))
        计算训练 G 的总损失
        $step \leftarrow step+1$
        $gen\_iterations \leftarrow gen\_iterations+1$
      $errG\_total, G\_log\ s \leftarrow \left(-\frac{1}{2}\left[E_{x\sim P_{G_i}}\log D_i(x)+E_{x\sim P_{G_i}}\log D_i(x,s)\right]\right)$
        $kl\_loss \leftarrow D_{\text{KL}}(N(\mu(s),\sum(s))\|N(0,I))$
        $perceptual\_loss \leftarrow \frac{1}{C_i H_i W_i}\|\varphi_i(I')-\varphi_i(I)\|_2^2$
        $errG\_total \mathrel{+}= (kl\_loss+perceptual\_loss)$
      (5) 反向传播更新参数
        if epoch % 20 == 0: //20 个 epoch 保存一次生成的模型
          save_model(netG, avg_param_G, netsD, epoch)
        End if
      save_model(netG, avg_param_G, netsD, smax_epoch)
  End while
End for

参考文献

1 Chen JB, Shen YL, Gao JF, *et al*. Language-based image editing with recurrent attentive models. Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 8721–8729.

2 Cheng Y, Gan Z, Li YT, *et al*. Sequential attention GAN for interactive image editing. Proceedings of the 28th ACM International Conference on Multimedia. Seattle: ACM, 2020. 4383–4391.

3 Dong H, Yu SM, Wu C, *et al*. Semantic image synthesis via adversarial learning. Proceedings of 2017 IEEE International Conference on Computer Vision. Venice: IEEE, 2017. 5707–5715.

4 El-Nouby A, Sharma S, Schulz H, *et al*. Tell, draw, and repeat: Generating and modifying images based on continual linguistic instruction. Proceedings of 2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019. 10303–10311.

5 Nam S, Kim Y, Kim SJ. Text-adaptive generative adversarial networks: Manipulating images with natural language. Proceedings of the 32nd Conference on Neural Information Processing Systems. Montréal: NIPS, 2018. 42–51.

6 Jing YC, Yang YZ, Feng ZL, *et al*. Neural style transfer: A review. IEEE Transactions on Visualization and Computer Graphics, 2020, 26(11): 3365–3385. [doi: 10.1109/TVCG.2019.2921336]

7 Miller AH, Fisch A, Dodge J, *et al*. Key-value memory networks for directly reading documents. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. Austin: ACL, 2016. 1400–1409.

8 Torkzadehmahani R, Kairouz P, Paten B. DP-CGAN: Differentially private synthetic data and label generation. Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. Long Beach: IEEE, 2019. 98–104.

9 Richardson E, Alaluf Y, Patashnik O, *et al*. Encoding in style: A StyleGAN encoder for image-to-image translation. Proceedings of 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville: IEEE, 2021. 2287–2296.

10 Karras T, Laine S, Aittala M, *et al*. Analyzing and improving the image quality of StyleGAN. Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 8107–8116.

11 Zhang H, Xu T, Li HS, *et al*. StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks. Proceedings of 2017 IEEE International Conference on Computer Vision. Venice: IEEE, 2017. 5908–5916.

12 Xu T, Zhang PC, Huang QY, *et al*. AttnGAN: Fine-grained text to image generation with attentional generative

adversarial networks. Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 1316–1324.

13 Li BW, Qi XJ, Lukasiewicz T, *et al*. Controllable text-to-image generation. Proceedings of the 33rd International Conference on Neural Information Processing Systems. Vancouver: NIPS, 2019. 2065–2075.

14 Chen ZL, Wang C, Wu HM, *et al*. DMGAN: Discriminative metric-based generative adversarial networks. Knowledge-based Systems, 2020, 192: 105370. [doi: 10.1016/j.knosys.2019.105370]

15 Li BW, Qi XJ, Lukasiewicz T, *et al*. ManiGAN: Text-guided image manipulation. Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 7877–7886.

16 Mirza M, Osindero S. Conditional generative adversarial nets. arXiv:411.1784, 2014.

17 刘建伟, 谢浩杰, 罗雄麟. 生成对抗网络在各领域应用研究进展. 自动化学报, 2014, 46(12): 2500–2536. [doi: 10.16383/j.aas.c180831]

18 Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv:1511.06434, 2015.

19 杜会芳, 王昊奋, 史英慧, 等. 知识图谱多跳问答推理研究进展、挑战与展望. 大数据, 2021, 7(3): 60–79.

20 Heusel M, Ramsauer H, Unterthiner T, *et al*. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach: Curran Associates Inc., 2017. 6629–6640.

(校对责编: 孙君艳)