计算机科学
COMPUTER SCIENCE

一种新的基于凸损失函数的离散扩散文本生成模型

李思慧, 蔡国永, 蒋航, 文益民

引用本文

李思慧, 蔡国永, 蒋航, 文益民. 一种新的基于凸损失函数的离散扩散文本生成模型[J]. 计算机科学, 2025, 52(10): 231-238.

LI Sihui, CAI Guoyong, JIANG Hang, WEN Yimin. Novel Discrete Diffusion Text Generation Model with Convex Loss Function [J]. Computer Science, 2025, 52(10): 231-238.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.240800147

一种新的基于凸损失函数的离散扩散文本生成模型

李思慧[1]　蔡国永[2]　蒋　航[2]　文益民[1,3]
1 桂林电子科技大学计算机与信息安全学院　广西 桂林 541004
2 广西可信软件重点实验室　广西 桂林 541004
3 桂林旅游学院广西文化和旅游智慧技术重点实验室　广西 桂林 541006
(22032201020@mails.guet.edu.cn)

摘　要　扩散语言模型采用的非自回归生成方式能显著提高推理速度,通过迭代重建过程持续优化能提高生成文本质量,因此它在文本生成任务中具有极大潜力。然而,扩散语言模型训练多采用基于极大似然估计的交叉熵损失,即便生成了正确句,也可能因为没有与参考句严格对齐被惩罚,使扩散语言模型面临严重的多模态问题,进而大大降低了文本生成质量。为了缓解多模态问题,提出了一种基于凸损失函数训练的离散扩散语言模型 ConvexDiffusion,该模型利用凸函数可以锐化最优分布这一特性,使模型更专注于高概率输出;为了进一步提高文本生成质量,降低生成词的重复率,设计了一种使噪声标记非线性变化的混合感知噪声表,并在解码过程中采用高置信度确定性去噪策略。在机器翻译、问题生成、问题阐述这 3 类文本生成任务上的实验结果表明,ConvexDiffusion 相比现有领先的扩散模型 RDM 和非自回归模型 CMLM 等,性能提升了 1～7 个 BLEU,且具有更快的生成速度。特别是在 WMT16'EN-RO 和 WMT14'EN-DE 这两个大型数据集上,ConvexDiffusion 的表现超越了目前主导文本生成领域的自回归语言模型。

关键词:扩散模型;文本生成;多模态问题;损失函数;凸损失函数
中图分类号　TP391

Novel Discrete Diffusion Text Generation Model with Convex Loss Function

LI Sihui[1],CAI Guoyong[2],JIANG Hang[2] and WEN Yimin[1,3]
1 College of Computer and Information Security,Guilin University of Electronic Technology,Guilin,Guangxi 541004,China
2 Key Laboratory of Guangxi Trusted Software,Guilin,Guangxi 541004,China
3 Guangxi Key Laboratory of Culture and Tourism Smart Technology,Guilin Tourism University,Guilin,Guangxi 541006,China

Abstract　Diffusion language models adopt a non-autoregressive generation approach that improves inference speed. Additionally, continuous refinement through an iterative refinement enhances the quality of the generated text, making it promising for text generation tasks. However, since diffusion language models are often trained using cross-entropy loss based on maximum likelihood estimation, even if the model generates a correct sentence, it may be penalized for not strictly aligning with the reference sentence, resulting in a serious multimodality problem, significantly reducing the quality of text generation. To alleviate the multimodality problem, a discrete diffusion language model ConvexDiffusion based on convex loss function training is proposed. The model leverages the property of convex functions to sharpen the optimal distribution so that the model focuses more on high-probability outputs. To further improve the quality and reduce the repetition rate of generated words, a hybrid-aware noise schedule that enabled the noise labelling to vary non-linearly is designed, along with a high-confidence deterministic denoising strategy employed during the decoding process. Experimental results on the three text generation tasks—machine translation, question generation, and question paraphrasing demonstrate that ConvexDiffusion achieves a performance improvement of 1～7 BLEU points and faster generation speed compared to leading diffusion models such as RDM and non-autoregressive models like CMLM. Especially on two large datasets, WMT16 EN-RO and WMT14 EN-DE, ConvexDiffusion surpasses the leading autoregressive models in text generation.

Keywords　Diffusion model, Text generation, Multimodality problem, Loss function, Convex loss function

到稿日期:2024-08-27　返修日期:2024-11-17
基金项目:国家自然科学基金(62366010);广西重点研发计划(桂科 AB21220023)
This work was supported by the National Natural Science Foundation of China(62366010) and Key R&D Program of Guangxi(AB21220023).
通信作者:蔡国永(ccgycai@guet.edu.cn)

1 引言

自回归(Autoregressive, AR)语言模型在语言建模领域占据主导地位,其按顺序逐个生成输出标记(token),每个 token 的生成都依赖于前序 token。虽然这种方法能够捕捉到 token 间的依赖关系,但固定的生成顺序限制了自回归模型的灵活性,并不可避免地导致了推理时的固有延迟。随着模型规模和生成句长度的不断增加,推理速度缓慢的问题更加严重。非自回归(Non-Autoregressive, NAR)语言模型基于条件独立性假设并行生成目标句的所有 token,大大降低了推理延时。虽然非自回归模型在解码速度上具有显著优势,但其只能独立学习预测每个位置的概率分布,无法建立目标 token 间的关联关系,面临多模态问题的挑战。这里的多模态与语音、文本等特征多模态不同,非自回归文本生成中的多模态问题是指对同一源句子可能存在不同但都正确的生成结果,而损失只根据单一参考句计算。因此,非自回归模型生成的文本质量与自回归模型相比,仍存在一定差距。

新兴的扩散模型[1](Diffusion Model)在图像、音视频等连续数据的生成中取得了很大成功,引起了文本生成研究者的关注。基于扩散的语言模型在每个时间步下的生成以非自回归的方式进行,提升了文本生成的效率;同时扩散模型采用的迭代重建过程可以润色先前生成的文本,从而提高文本生成的质量。相较于传统语言模型,扩散语言模型的灵活性更好,可以在生成质量和效率之间实现更优的权衡。然而,扩散语言模型同样面临多模态问题[2]。在逆转噪声逐步生成目标数据的过程中,模型需要处理复杂的高维数据分布,而给定的输入可能对应多个高概率输出,并且模型采用的非自回归生成方式使得每个时间步下的预测相对独立,难以建立 token 间的依赖关系。此外,离散扩散文本生成模型的损失通常被简化为重加权的标准交叉熵损失[3],该损失要求每个位置的生成词尽可能匹配参考文本中的对应词,位置未严格对齐便可能受到损失惩罚。这种损失函数可能使扩散语言模型的训练朝着错误的方向进行,引起更严重的多模态问题。有许多工作[4-6]试图解决这一问题,但它们大多在词级别上重构损失,缺乏对最优分布的理论保证,且主要应用于传统的非自回归语言模型,在扩散语言模型上的探索不足。

为了缓解上述多模态问题,并在保证一定生成速度的前提下提高文本生成质量,本文为离散扩散文本生成模型设计了一种基于凸函数的新型训练损失函数,结合交叉熵损失共同引导模型在句子级别上识别高概率的生成结果,并给出了对最优分布的理论保证。本文的主要贡献可归纳为以下 3 个方面:

1)设计了一类基于凸函数的新型训练损失函数,探讨了在这种训练方式下最优预测分布的理论特性,并在 3 类文本生成任务中验证了其有效性。

2)提出了一个新的混合感知噪声表,并在解码过程中采用高置信度确定性去噪策略,通过实验验证了其有效性。

3)各种文本生成任务的实验结果表明,Convex Diffusion 不仅提升了离散扩散语言模型的学习性能,也实现了比其他基线模型更优的生成质量和效率。

2 相关工作

2.1 离散扩散模型

离散扩散模型侧重于构建直接作用于离散空间的扩散过程,Sohl-Dickstein 等[7]首次提出了离散扩散模型,旨在预测连续数据的二进制表示。随后,Hoogeboom 等[8]进一步探索了具有均匀转移核状态的离散扩散过程。Austin 等[9]则提出了一个用于离散状态的扩散模型通用框架 D3PM,并首次在大规模语料库上测试了离散扩散模型。经典的离散扩散模型将潜变量 $x_T\cdots x_0$ 作为马尔可夫链,建模给定数据的概率分布,具体分为前向加噪和后向去噪两个过程。

在前向加噪过程中,从时间步长 $t=0$ 时的初始状态 x_0 开始,根据噪声表 $\beta=\{\beta_1,\cdots,\beta_T\}$ 逐步向源数据中添加噪声,经过 T 步后,最终将初始数据分布 $q(x_0)$ 转换为噪声分布 q_{noise},具体表示如下:

$$q(x_t\mid x_0)=\alpha_t x_{t-1}+(1-\alpha_t)q_{\text{noise}} \tag{1}$$

后向去噪过程从高斯分布 $p(x_T)=N(x_T;0,\mathrm{I})$ 中采样噪声后,通过参数化的去噪网络 f_θ 预测后向分布,再基于预测的分布进行随机采样得到 x_{t-1},通过不断的预测和采样过程逐步对数据进行去噪,最终得到原始数据 x_0:

$$p_\theta(x_{t-1}\mid x_t)=\mathcal{N}(x_{t-1};\mu_\theta(x_t,t),\sum_\theta(x_t,t)) \tag{2}$$

其中,$\mu_\theta(\cdot)$和$\sum_\theta(\cdot)$通过参数化去噪网络 f_θ 估算。扩散模型的训练目标是最大化数据 $\log[p_\theta(x_0)]$的边际似然,为此引入 KL 散度,按照负对数似然变分下界方法来训练扩散模型,最终得到模型参数 θ:

$$\mathcal{L}_{\text{vlb}}=\mathbb{E}_q[\underbrace{D_{\text{KL}}(q(x_t\mid x_0)\parallel p_\theta(x_T))}_{\mathcal{L}_T}]-\underbrace{\log p_\theta(x_0\mid x_1)}_{\mathcal{L}_0}+\mathbb{E}_q[\sum_{t=2}^{T}\underbrace{D_{\text{KL}}(q(x_{t-1}\mid x_t,x_0)\parallel p_\theta(x_{t-1}\mid x_t))}_{\mathcal{L}_{t-1}}] \tag{3}$$

2.2 非自回归模型和多模态问题

经典的非自回归模型基于条件独立性假设,所采用的并行生成方式大大加快了解码速度,有效降低了推理延时,在工业应用中具有很大潜力。假设源输入文本 $X=\{x_1,x_2,\cdots,x_T\}$,目标输出参考文本 $Y=\{y_1,y_2,\cdots,y_T\}$,θ 为模型参数,T 为目标句长度,非自回归语言模型按式(4)进行建模:

$$P_{\text{NAR}}(Y\mid X,\theta)=\prod_{t=1}^{T}p(y_t\mid X,\theta) \tag{4}$$

虽然这种方法可以快速解码,但多模态问题严重影响了非自回归类模型的文本生成质量。以机器翻译为例,当模型将“我必须出去打篮球了”翻译成“I have to get out and play basketball”,而参考译文是“I must get out and play basketball now”时,虽然模型的生成结果是无语法错误的通畅句子,但在传统交叉熵损失的评估下,所有位置的输出都被要求更正为参考译文中的对应词。这使得模型无法准确捕捉目标句的真正概率分布,反而可能生成“I have to out out play play basketball”这种包含重复词的语义错误句。

为了减轻多模态问题对非自回归类模型在生成性能方面的影响,研究者提出了多种改进方法,包括知识蒸馏、模型架构、训练方法等。知识蒸馏作为一种常见的策略,通过自回归教师模型的生成句来替换训练集中的目标句,从数据层面缓

解了多模态问题。Demirag 等[2]证明扩散模型同样存在多模态问题后，采用知识蒸馏方法加以解决，实验证明该方法取得了一定效果。但知识蒸馏方法高度依赖于从自回归教师模型中提取的知识，导致模型性能受限。模型架构方法通常通过增加解码器长度以实现译文长度的动态调整，但解码器长度的增加也提高了模型的计算开销。自非自回归类模型损失函数不准确的问题被指出以来，研究者逐渐将缓解多模态问题的重点转向对训练方法的改进，提出了序列级损失函数[4]、动态参考译文[5]和放宽交叉熵损失中排列限制[6]等多种损失函数。然而，这些方法往往无法有效引导模型在句子级别上重构损失，并且在扩散语言模型方面的研究仍显不足。

本文的研究重点在于增强模型本身对真实世界数据分布的学习能力，因此未采用可能影响数据分布的知识蒸馏技术，而是从损失函数入手，针对词级交叉熵损失难以准确评估模型输出的问题，充分利用扩散模型在生成任务中的优势，设计新的凸损失函数，旨在有效引导模型在句子级别上识别高概率的生成结果，从而提高模型的生成性能。

3 基于凸损失函数的离散扩散文本生成模型

图 1 展示了基于凸损失函数的离散扩散文本生成模型 ConvexDiffusion 的总体框架。在前向过程中，采用设计的混合噪声表调度策略逐步为数据加噪，直至数据变为完全屏蔽序列。后向过程从该序列开始，在每个时间步采用设计的确定性解码策略并行预测被屏蔽的部分 token，最终生成完整的目标文本。此外，本文为 ConvexDiffusion 的训练设计了一种新的凸损失函数，后续将分别对这些设计进行详细阐述。

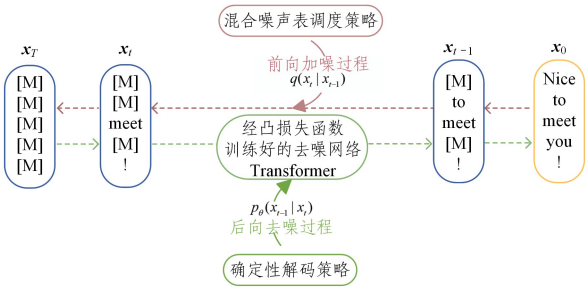


图 1 基于凸损失函数的离散扩散文本生成模型

Fig. 1 Discrete diffusion text generation model based on a convex loss function

3.1 混合噪声表调度设计

噪声表作为噪声尺度的函数，是扩散模型的一个关键组成部分。它定义了在整个训练过程中，数据样本在不同时间步下被添加的噪声量，因此噪声表的设定方式严重影响了文本生成质量。现有离散扩散模型常采用的是线性噪声表，即在前向加噪过程中，被噪声处理的 token 随时间步长线性变化。然而有研究证明，线性噪声表的噪声尺度增长过快，导致后四分之一的潜变量几乎是纯噪声，这种设定方式并非最佳选择[10]。此外，单一线性的噪声权重不符合人类学习表达的自然过程，这种简化的噪声方式会导致生成句缺乏连贯性，难以应对多模态问题。

为了弥补线性噪声表的缺陷，本文提出了一种新的混合感知噪声表，综合利用余弦函数周期性振荡和指数衰减的平滑特性，使噪声信号非线性变化，从而提供更丰富的信号形态，减缓噪声尺度的增长速度。混合噪声表中，α_t 设计如下：

$$\alpha_t=\exp\left(-\frac{1}{\gamma}\left(\frac{t+1}{T}\right)^2\right)\cdot\cos\left(\frac{\pi}{2}\cdot\frac{(t+1)}{T}\right) \tag{5}$$

其中，γ 是一个超参数，将在实验部分进行讨论。如图 2 所示，与线性噪声表相比，本文的混合噪声表在初期会快速增加噪声，防止模型过度拟合于简单模式，并帮助模型更快地适应复杂的学习环境。当噪声水平变得较高时，逐步放缓噪声的增加速度，以避免模型在高噪声下浪费太多的训练步数。因为噪声过多时，模型几乎无法学习到有用的知识。这种渐进的噪声衰减能够有效减小高频成分的影响，使噪声水平在接近总时间步长 T 的过程中逐渐变得平滑，减少突变和振荡。对于需要保持连贯性和稳定性的文本生成任务，平滑性至关重要。根据噪声表，给定目标序列长度 N，每次前向加噪过程中被屏蔽的 token 数 n_t 为：

$$n_t=N\cdot\left(\left(1-\exp\left(-\frac{1}{\gamma}\left(\frac{t+1}{T}\right)^2\right)\right)\cdot\cos\left(\frac{\pi}{2}\cdot\frac{(t+1)}{T}\right)\right) \tag{6}$$

不同于自回归模型，扩散模型这类非自回归生成式模型在解码时需要提前确定目标序列的长度 N。参考文献[11]中的处理方式，在编码器之上额外增加一个长度预测模块，为目标序列提供长度候选项。给定源输入，首先运行编码器以获得隐藏表示，将隐藏表示的所有向量平均化后，传递到线性层得到一组输出长度得分，选取得分最高的长度作为最终的目标序列长度 N。

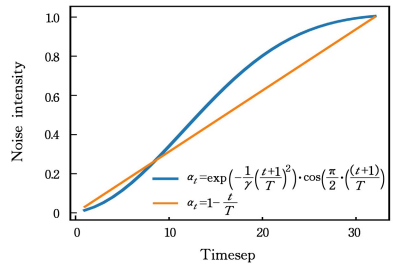


图 2 不同噪声表随时间步长变化的噪声程度

Fig. 2 Noise intensity of different noise schedules with time step variation

3.2 解码策略设计

为了进一步提高 ConvexDiffusion 的文本生成质量，在解码过程中从完全噪声的序列开始，不同于以往对所有屏蔽 token 随机去噪的统一处理方式，本文采取分类处理序列里的各个屏蔽 token。通过收集神经网络输出的最大值，为每个 token 分配一个分数，表示模型对该 token 的预测置信度。在扩散后向过程中，仅对神经网络输出中高置信度的屏蔽 token 进行去噪，对低置信度的屏蔽 token 则保持噪声状态。为了确保噪声程度随生成过程的进行而降低，设置每次迭代过程中处于去噪状态的 token 数量 $k=\cos\frac{\pi t}{2T}\cdot N$，$k$ 从 0 递增到 N。模型解码过程如算法 1 所示。

算法 1 采样 Convex Diffusion

输入：已训练好的神经网络 $p(x_{t,n};\theta)$

输出：生成的去噪目标数据 $x_{0,1:N}$

1. forn=1,2,…,N do
2. 初始化 $x_{T,n}\sim q_{noise}$;
3. end for
4. /* 将噪声序列 $x_{t,1:N}$ 输入神经网络,取输出最大值代表每个 token 的分数 */
5. for t=T,…,1 do
6. for n=1,2,…,N do
7. Draw $\tilde{x}_{t,n}\sim$Categorical($p(x_{t,n};\theta)$);
8. $score_{t,n}=\max\limits_{1\leqslant j\leqslant K} p_j(x_{t,n};\theta)$;
9. /* 确定 $x_{t,1:N}$ 中所有处于非 mask 状态的 token,将 $\tilde{x}_{t,1:N}$ 中相应位置的 token 分数设为正无穷 */
10. if $x_{t,1:N}$ is not mask then
11. $setscore_{t,n}=+\infty$;
12. end if
13. /* 选取前 k 个 mask token 进行去噪,跳过已处于去噪状态的 token */
14. if $score_{t,n}\in \underset{1\leqslant n\leqslant N}{topk}\{score_{t,n}\}_{n=1}^{N}\neq+\infty$ then
15. $x_{t-1,n}=\tilde{x}_{t,n}$;
16. else
17. $x_{t-1,n}$保持不变;
18. end if
19. end for
20. end for/* 直到 t=0 时结束循环,此时所有 token 都处于去噪状态 */
21. Return $x_{0,1:N}$

这一总体策略具有更大的信息量,通过比较得分确定屏蔽 token 的状态,使模型能在丰富的双向语境中反复考虑词语选择,并更精细地捕捉 token 间的关系。

3.3 凸损失函数的设计及示例

为了缓解多模态问题,本节提出了一类基于凸函数的新型训练损失函数,以更好地利用扩散模型满足文本生成任务的特定要求。直观地说,传统的标准交叉熵损失函数基于对数概率,而对数函数是凹函数,其梯度随模型预测概率的增加而减少。这意味着当模型已经对某个样本预测出较高概率时,再进一步提高该概率不会显著减少损失函数的值。相反,凸函数的梯度随模型预测概率的增加而增加,因此模型会倾向于预测高概率输出,从而使预测分布更加集中。严格的数学推导已经证明,当拟合函数是递增凸函数时,最小化损失的最优分布 p_l 是独热(One-hot)分布[12],即模型会收敛到比真实数据分布更清晰的最优分布,这种特性对于追求确定性输出的封闭式文本生成任务尤为有利。

因此,为了缓和整体损失函数的凹性,本文提出了一种凸优化方法,将递增凸函数 f 与原始凹函数 g 结合起来,新损失函数结构表示如下:

$$\mathcal{L}(\theta)=-\left(1-\frac{t-1}{T}\right)\sum_{n=1}^{N}p_{\text{data}}(x_{0,n})\cdot fg(p(x_{t,n};\theta)) \quad (7)$$

其中,$p_{\text{data}}(x_{0,n})$为真实数据的概率分布,$p(x_{t,n};\theta)$为模型预测的概率分布,$g=\log$ 是基于交叉熵损失的预测概率拟合函数,f 为本文要进行复合的递增凸函数。

推论 1 证明了在这一损失函数组合框架下,基于凸损失函数的最优分布 p_{fg} 的香农熵小于或等于原始交叉熵损失函数的最优分布 p_g 的香农熵,模型的最优分布得到了理论保障。

推论 1 p_{fg} 的香农熵小于或等于 p_g 的香农熵。

证明:香农熵用于衡量分布 p 的平均不确定性,定义为 $H_p=-\sum\limits_x p(x)\log p(x)$,$p(x)$为事件 x 发生的概率,熵越大,分布的不确定性越高。$H=-(x_1\log x_1+x_2\log x_2)$为概率值 x_1 和 x_2 的香农熵贡献。定义函数 $h(\Delta x)=-(x_1+\Delta x)\log(x_1+\Delta x)-(x_2-\Delta x)\log(x_2-\Delta x)$表示对概率值 x_1 增加 Δx 并相应对 x_2 减小 Δx 后的熵变化,该函数的一阶导数为:

$$\begin{aligned}h'(\Delta x)&=\frac{\mathrm{d}}{\mathrm{d}\Delta x}[-(x_1+\Delta x)\log(x_1+\Delta x)-(x_2-\Delta x)\log(x_2-\Delta x)]\\&=\log(x_2-\Delta x)-\log(x_1+\Delta x)\end{aligned} \quad (8)$$

假设 $x_1\geqslant x_2$,那么对于任意的 $\Delta x>0$,有 $h'(\Delta x)<0$。因此当增大 x_1 并相应减小 x_2 时,香农熵会减小。由于凸损失函数的梯度特性,模型倾向于增加高概率输出的权重,每次调整实质上都是在重新分配概率质量,即把低概率输出的概率转移到高概率输出上,从而使得整个分布更加集中。因此,从 p_g 到 p_{fg} 的转换可以视为一系列类似对 $h(\Delta x)$的调整,经过多次这样的调整后,分布 p_{fg} 的香农熵必然小于或等于初始分布 p_g 的香农熵。通过在损失函数中引入凸函数,香农熵得以有效降低,这一变化意味着模型的最优分布变得更加清晰,降低了生成多个模式的可能性。

前述理论分析证明了在损失函数中引入凸函数的有效性,接下来给出凸损失函数在离散扩散文本生成模型中的实际示例。$(-\infty,0]$上常见的递增凸函数有两种,分别是指数函数 $f(x)=e^{kx}$,$k\geqslant 0$ 和幂函数 $f(x)=-(-x)^k$,$0\leqslant k\leqslant 1$。根据实验结果,选择结合指数函数与原始对数概率形成凸复合函数 $fg(p(x_{t,n};\theta))=p(x_{t,n};\theta)^k$,超参数 k 用于调整复合损失函数的凸性。凸复合损失函数的梯度为 $f'(g(p(x_{t,n};\theta)))\cdot g'(p(x_{t,n};\theta))$,相较于原始损失函数 $g(p(x_{t,n};\theta))$的梯度 $g'(p(x_{t,n};\theta))$,多了一个附加项 $f'(g(p(x_{t,n};\theta)))=k\cdot p(x_{t,n};\theta)^k$,这可以解释为损失的权重。对于更确定的生成 token,该权重会更大,从而引导模型更专注于生成高概率输出。实验中也尝试了结合幂函数作为凸复合损失函数,然而结果表明,幂函数形式在训练过程中的表现较差。在离散扩散文本生成模型的训练过程中,最终形成的凸损失函数为:

$$\mathcal{L}(\theta)=-\left(1-\frac{t-1}{T}\right)\sum_{n=1}^{N}p_{\text{data}}(x_{0,n})(p(x_{t,n};\theta))^k \quad (9)$$

其中,$x_{t,1:N}:=\{x_{t,n}\}_{n=1}^{N}$表示第 t 个时间步长的 token 序列,其中 $x_{t,n}$是第 n 个 token,N 是序列长度。模型训练过程如算法 2 所示。

算法 2 训练 ConvexDiffusion

输入:神经网络 $p(x_{t,n};\theta)$,初始数据分布 $p_{\text{data}}(x_{0,1:N})$

输出:模型参数 θ

1. 初始化神经网络 $p(x_{t,n};\theta)$的参数 θ;
2. While not converged do
3. Draw $x_{0,1:N}\sim p_{\text{data}}(x_{0,1:N})$; /* 从初始数据分布 $p_{\text{data}}(x_{0,1:N})$中抽取样本序列 $x_{0,1:N}$ */
4. Draw t∈Uniform({1,…,T}); /* 从{1,…,T}的均匀分布中抽取时间步长 t */
5. /* 分别对当前时间步下的每一个样本从条件分布 $q(x_{t,n}|x_{0,n})$中抽取得到 $x_{t,n}$ */

6. for n=1,2,…,N do
7. Draw $x_{t,n}\sim q(x_{t,n}|x_{0,n})$;
8. end for
9. /*将 $x_{t,n}$输入神经网络 $p(x_{t,n};\theta)$，预测概率分布并计算损失*/
10. $\mathcal{L}(\theta)=-(1-\frac{t-1}{T})\sum_{n=1}^{N}p_{data}(x_{0,n})(p(x_{t,n};\theta))^k$;
11. /*通过最小化损失函数$\mathcal{L}(\theta)$反向传播更新参数θ，直到收敛*/
12. backpropagate and update parameters($\mathcal{L}(\theta)$);
13. end while

4 实验

4.1 实验设置

在机器翻译、问题生成、问题阐述3类不同的文本生成任务中评估了ConvexDiffusion的性能。所有实验均使用Fairseq库工具包，不使用知识蒸馏技术，对比baseline的结果均为原对应论文中报告的最优结果。在所有数据集上，沿用RDM中的参数[3]，总扩散步长 $T=32$，凸损失函数超参数 $k=0.7$，采用混合噪声表进行学习，$\gamma=0.45$，其他超参数设置如表1所列。为了获得稳健的结果，各实验取最后5个检查点的均值作为模型参数。解码时，在翻译实验中，使用5个长度候选项并行解码，选择模型评分最高的序列作为最终输出。在问题生成和阐述任务中，遵循DiffuSeq[13]的方法，使用包含10个样本的MBR解码，以确保公平对比。

硬件环境上，所有实验均在NVIDIA Tesla V100 GPU服务器上进行，使用4块GPU并行进行训练，使用1块GPU进行解码。软件环境上，使用Python 3.8及PyTorch深度学习框架进行编程，操作系统为Ubuntu 22.04。

表1 超参数设置

Table 1 Hyperparameter setting

Hyper-parameter	IWSLT14'DE-EN	WMT16'EN-RO	WMT14'EN-DE	QG	QQP
Hidden size	512	512	512	512	512
Number of warm-up steps	30000	15000	10000	10000	10000
Number of attention heads	4	8	8	8	8
Number of training steps	300000	120000	300000	70000	70000
Label smoothing	0.1	0.1	0.1	0.1	0.1
Optimizer	Adam	Adam	Adam	Adam	Adam
Dropout	0.3	0.3	0.1	0.2	0.2
Hidden size in FFN	1024	2048	2048	1024	1024
Weight decay rate	0.01	0.01	0.01	0.01	0.01

4.2 机器翻译任务

4.2.1 实验数据

在机器翻译任务中，选用IWSLT14'DE-EN[14]，WMT16'EN-RO[15]，WMT14'EN-DE[16]这3个数据集进行实验，分别由160000/7000/7000，610000/2000/2000，4×10^6/3000/3000个句子对用于训练、验证和测试。

4.2.2 对比模型及评价指标

在机器翻译任务中，考虑4组模型作为基线。

1)非自回归模型：CMLM[11]，CMLMC[17]，CMLM+SMRAT[18]，Disco[19]。CMLM模型是迭代式非自回归模型中最具代表性的框架之一，由条件掩蔽概率模型和掩码预测解码算法组成。CMLMC和CMLM+SMRAT分别在CMLM的基础上引入了不同的自我纠错机制。Disco引入多样性和覆盖率概念，提高了翻译结果的准确性。

2)连续扩散模型：CDCD[20]。采用分步去噪和自条件化等技术，提高了扩散语言模型的生成质量。

3)离散扩散模型：Absorbing Diffusion[21]和RDM[3]。Absorbing Diffusion修改了扩散过程，以适应离散数据特性。RDM重参数化了离散扩散模型的训练过程。

4)自回归模型：Transformer[22]。采用编码器-解码器架构，以自回归的方式生成目标句。

本文使用标准指标BLEU[23]评估模型生成的准确性。

4.3 问题生成与阐述任务

4.3.1 实验数据

问题生成任务选用Question Generation(QG)数据集[24]，旨在给定上下文作为输入时生成问题，分别使用117000/2000/10000个问题对来进行训练、验证和测试。问题阐述任务选用Quora Question Pairs(QQP)数据集[25]，旨在使用同一语言生成相同语义内容的替代形式，分别由145000/2000/2500个问题对来进行训练、验证和测试。

4.3.2 对比模型及评价指标

在问题生成和阐述任务中，考虑3组模型作为基线。

1)自回归语言模型：GRU-attention[26]和Transformer[22]。GRU-attention融合了门控机制与注意力机制。Transformer采用编码器-解码器架构，通过并行处理提高了训练效率。

2)预训练语言模型：GPT-2和GPVAE-T5[27]。GPT-2是基于Transformer架构的大语言模型，适用于多种语言任务。GPVAE-T5结合了变分自编码器的原理，提高了文本生成的多样性。

3)非自回归语言模型：LevT[28]。这是一个广泛使用且性能领先的迭代式非自回归模型。

4)扩散模型：DiffuSeq[13]和RDM[3]。两者分别是连续扩散过程和离散扩散过程中的领先模型。

本文使用BLEU[23]，ROUGE-L[29]，BERTScore[30]这3个指标来评估模型生成的准确性，较高的分数反映了更好的性能。此外，使用不重复单词比例(Dist-1)[31]来评估生成句的多样性，较低的Dist-1表明生成句中包含更多重复词。

5 实验结果分析

5.1 机器翻译实验结果分析

机器翻译基准上的实验结果如表2所列，粗体为最佳

结果。由表 2 可知，现有离散扩散模型在机器翻译任务中的表现欠佳，且在处理大型数据集时的扩展性较差。所有数据集中，ConvexDiffusion 相较于原有领先的非自回归模型及其他离散、连续扩散模型均有一定的性能提升(1～7 BLEU)。此外，本方法有效地将离散扩散语言模型扩展到了更大规模的数据集，在 WMT'16 En-Ro 和 WMT'14 En-De 数据集上，ConvexDiffusion 取得了与自回归基线相媲美的生成结果。

不同模型在 WMT'16 En-Ro 数据集上生成速度与质量的权衡曲线如图 3 所示，使用测试集的推理时间来衡量生成速度。相较于其他基线模型，本方法能够在保持生成质量的同时显著提升生成速度，仅用约一半的时间成本即可生成质量超越自回归基线的结果。

表 2 在 IWSLT'14 DE-EN, WMT'16 EN-RO 和 WMT'14 EN-DE 数据集上的 BLEU 分数比较

Table 2 Comparison of BLEU scores between IWSLT'14 DE-EN, WMT'16 EN-RO, and WMT'14 EN-DE

Type	Model	Iterations	IWSLT14 DE-EN	WMT'16 En-Ro	WMT'14 En-De
Non-autoregressive Models	CMLM	16	32.18	32.90	25.00
	CMLMC	10	34.28	34.14	26.40
	CMLM+SMRAT	10	30.74	32.71	25.10
	Disco	Adaptive	—	—	25.64
Continuous Diffusion	CDCD	200	—	—	20.0
Discreet Diffusion	Absorbing Diffusion	4	26.93	29.16	19.48
		10	28.32	30.41	21.62
		16	28.38	30.79	22.07
		25	28.93	30.56	22.52
	RDM	4	31.47	32.60	24.26
		10	33.91	33.38	26.96
		16	34.41	33.82	27.58
		25	34.49	33.99	27.59
	ConvexDiffusion(Ours)	4	31.52	32.45	23.51
		10	33.64	33.63	26.66
		16	34.20	34.02	26.99
		25	34.48	**34.34**	**27.63**
Auto-regressive Models	Transformer-base	n. a.	**34.51**	34.16	27.53

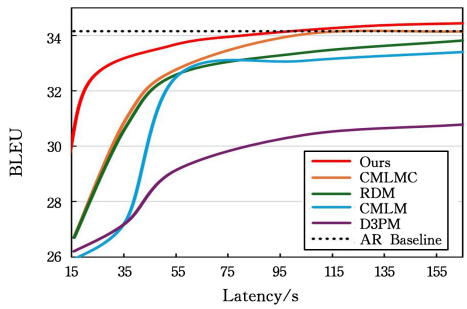


图 3 在 WMT'16 En-Ro 上生成速度与质量的权衡曲线

Fig. 3 Trade-off curves between generation speed and quality on WMT'16 En-Ro

5.2 问题生成和阐述实验结果分析

在问题生成和阐述这两个基准上的实验结果如表 3 所列，粗体为最佳结果。由表 3 可知，在准确性的 3 个评估指标中，ConvexDiffusion 至少在一个指标上领先。在多样性评估指标 Dist-1 上，ConvexDiffusion 的表现均优于强大的连续扩散基线 DiffuSeq、离散扩散基线 RDM 以及非自回归基线 LevT。这表明新的凸损失函数减少了生成句中重复词的出现，有效缓解了多模态问题。

预训练语言模型在大规模语料库上进行训练，能够更好地捕捉语言的丰富性和多样性，因此在 Dist-1 指标上的表现更为出色。相比之下，ConvexDiffusion 在生成多样性方面存在一定不足，后续仍需进行进一步的改进和优化。

表 3 问题生成和阐述上的实验结果

Table 3 Experimental results on QG and QQP

Task	Type	Model	BLEU	ROUGE-L	BERTScore	Dist-1
QG	Auto-regressive Models	GRU-attention	0.0651	0.2617	0.5222	0.7930
		Transformer-base	0.1663	0.3441	0.6307	0.9309
	Pre-trained language Models	GPT2-large FT	0.1110	0.3215	0.6346	**0.9670**
		GPVAE-T5	0.1251	0.3390	0.6308	0.9381
	Non-autoregressive Models	LevT	0.0930	0.2893	0.5491	0.8914
	Continuous Diffusion	DiffuSeq	0.1731	**0.3665**	0.6123	0.9056
	Discreet Diffusion	RDM	0.1802	0.3550	0.6310	0.9082
		Ours	**0.1814**	0.3556	**0.6354**	0.9205
QQP	Auto-regressive Models	GRU-attention	0.1894	0.5129	0.7763	0.9423
		Transformer-base	**0.2722**	0.5748	0.8381	0.9748
	Pre-trained language Models	GPT2-large FT	0.2059	0.5415	0.8363	0.9819
		GPVAE-T5	0.2409	0.5886	0.8466	0.9688
	Non-autoregressive Models	LevT	0.2268	0.5795	0.8344	0.9790
	Continuous Diffusion	DiffuSeq	0.2413	0.5880	0.8365	0.9807
	Discreet Diffusion	RDM	0.2498	0.5886	0.8466	0.9817
		Ours	0.2585	**0.5941**	**0.8514**	**0.9824**

5.3 凸损失函数的有效性分析

本节将凸优化损失函数与传统离散扩散模型的损失函数进行了对比，并探讨了不同类型凸函数在损失中的复合效果，$\mathcal{L}(\theta)=(1-\frac{t-1}{T})\sum_{n=1}^{N}p_{\text{data}}(x_{0,n})(-\log p(x_{t,n};\theta))^{k}$，$0\leqslant k\leqslant 1$ 为使用幂函数复合的凸损失函数，取 $k=0.85$ 时效果最佳，结果如表 4 所列。由表 4 可知，改进后的训练方案通过引导模型将大部分概率分配给所有适当候选中的最佳者，显著提升了模型性能，相比于原始损失函数，模型的 BLEU 分数提高了约 4 个百分点。

表 4 不同损失函数在 WMT'16 En-Ro 上的实验结果

Table 4 Experimental results of different loss functions on WMT'16 En-Ro

损失函数类型	BLEU
原始损失函数	30.56
交叉熵损失函数	33.99
幂函数复合的凸损失函数	34.21
指数函数复合的凸损失函数	**34.34**

此外，与指数函数相比，幂函数在离散扩散文本生成模型的凸损失框架中表现不佳，我们认为这是幂函数的数学特性对梯度的附加影响所致。如图 4 所示，在凸复合损失框架中应用幂函数时，梯度的附加项 $f'(g(p(x_{t,n};\theta)))$ 为 $k\cdot(-\log(p(x_{t,n};\theta)))^{k-1}$，当 k 接近 1 时，由于幂函数的凸性减弱，梯度增益会相应减少，$f'(g(p(x_{t,n};\theta)))$ 在不同预测概率的情况下都接近常数 1；当 k 接近 0 时，$f'(g(p(x_{t,n};\theta)))$ 会从接近 0 的极小值爆发性增长到极大值。这些因素会导致梯度爆炸，使得训练过程极其不稳定。而应用指数函数时，梯度能够稳定增长，不会出现爆炸式变化或常数值问题。因此，幂函数不适合作为凸优化损失框架中的复合函数。

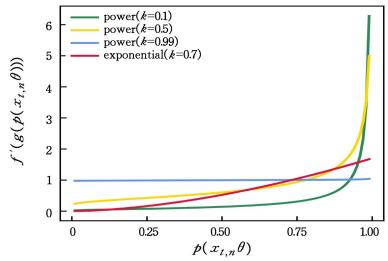


图 4 不同凸函数下梯度的变化曲线

Fig. 4 Gradient curves under different convex functions

此外，本节也探讨了指数超参数 k 对 ConvexDiffusion 模型文本生成准确性的影响。表 5 列出了模型在不同超参数 k 值下的 BLEU 分数。通过对比分析发现，当 $k=0.7$ 时，模型在各数据集下均实现了最佳性能。这一结果表明，指数超参数 k 在一定程度上决定了模型生成的文本质量，其选择对于提升模型的整体性能至关重要。

表 5 问题生成和阐述上随 k 变化的 BLEU 分数

Table 5 BLEU on QG and QQP test sets with varying k

k	QG	QQP
不使用凸优化策略	0.1744	0.2507
k-Power=0.3	0.1725	0.2391
k-*Power*=0.5	0.1746	0.2462
k-*Power*=0.7	**0.1814**	**0.2585**
k-*Power*=0.75	0.1765	0.2512

5.4 混合噪声表的有效性分析

图 5 显示了在 IWSLT'14 De-En 数据集上，随着迭代次数的增加，使用混合噪声表不同超参数 γ 下模型的 BLEU 分数。实验结果表明，所提出的混合噪声表在各个迭代次数上的文本生成质量均显著优于原有的线性噪声表。在离散扩散文本生成模型 ConvexDiffusion 的学习过程中，采用具有平滑过渡特性的混合噪声表显著提升了模型性能。当 $\gamma=0.45$ 时，模型在所有迭代次数中均表现出了一定程度的领先性能。

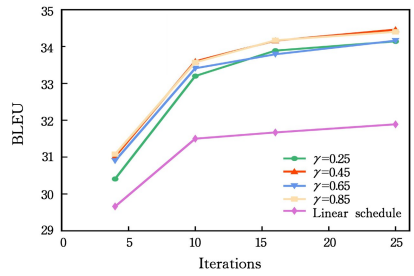


图 5 不同超参数 γ 在 IWSLT'14 De-En 上的实验结果

Fig. 5 Results of different hyperparameters γ on IWSLT'14 De-En

5.5 确定性解码策略的有效性分析

为了进一步验证所提确定性解码策略的有效性，本文在数据量大的 WMT14'EN-DE 数据集上进行了消融实验。在实验中，统一采用了新提出的损失函数和混合噪声表调度设计，并分别在解码过程中使用随机性和确定性去噪策略，以全面比较两种策略在文本生成质量上的表现，实验结果如表 6 所列。结果表明，确定性去噪策略有效提升了模型的生成准确率，在迭代至第 25 次时，模型的 BLEU 分数提升了 1.19。

表 6 不同解码策略在 WMT14'EN-DE 上的实验结果

Table 6 Experimental results of different decoding strategies on WMT14'EN-DE

去噪策略	迭代次数		
	10	16	25
随机性去噪策略	26.12	26.38	26.44
确定性去噪策略	26.66	26.99	27.63

结束语 本文提出了一种新颖的离散扩散文本生成模型 ConvexDiffusion。通过设计一种基于凸函数的新型训练损失函数，并结合混合噪声表和高置信度确定性去噪的双重策略，有效缓解了多模态问题，进一步提升了离散扩散语言模型的性能。通过在机器翻译、问题生成、问题阐述这 3 类文本生成任务的 5 个数据集上进行实验，充分证明了 ConvexDiffusion 的有效性。特别是在大型数据集上，ConvexDiffusion 的性能超越了领先的自回归基线模型。

然而，ConvexDiffusion 在文本生成多样性方面仍存在一定局限性。未来的研究工作将重点尝试使用一些数据增强技术，进一步降低 ConvexDiffusion 在生成过程中的词汇重复率，力求达到预训练语言模型在生成多样性方面的先进水平。

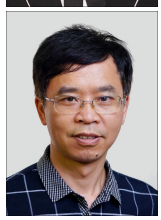
参 考 文 献

[1] YANZ H，ZHOU C B，LI X C. A review of research on generative diffusion models[J]. Computer Science，2024，51(1)：273-283.

[2] DEMIRAG Y，LIU D，NIEHUES J. Benchmarking Diffusion Models for Machine Translation[C]// Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics：Student Research Workshop. 2024：313-

324.
[3] ZHENG L, YUAN J, YU L, et al. A reparameterized discrete diffusion model for text generation[J]. arXiv:2302.05737, 2023.
[4] LI Y, CUI L, YIN Y, et al. Multi-granularity optimization for non-autoregressive translation[J]. arXiv:2210.11017, 2022.
[5] SHAO C, ZHANG J, ZHOU J, et al. Rephrasing the reference for non-autoregressive machine translation[C]// Proceedings of the AAAI Conference on Artificial Intelligence. 2023: 13538-13546.
[6] GHAZVININEJAD M, KARPUKHIN V, ZETTLEMOYER L, et al. Aligned cross entropy for non-autoregressive machine translation[C]// International Conference on Machine Learning. PMLR, 2020: 3515-3523.
[7] SOHL-DICKSTEIN J, WEISS E, MAHESWARANATHAN N, et al. Deep unsupervised learning using nonequilibrium thermodynamics[C]// International Conference on Machine Learning. PMLR, 2015: 2256-2265.
[8] HOOGEBOOM E, NIELSEN D, JAINI P, et al. Argmax flows and multinomial diffusion: Learning categorical distributions[J]. Advances in Neural Information Processing Systems, 2021, 34: 12454-12465.
[9] AUSTIN J, JOHNSON D D, HO J, et al. Structured denoising diffusion models in discrete state-spaces[J]. Advances in Neural Information Processing Systems, 2021, 34: 17981-17993.
[10] LIN S, LIU B, LI J, et al. Common diffusion noise schedules and sample steps are flawed[C]// Proceedings of the IEEE/CVF Winter Confe-rence on Applications of Computer Vision. 2024: 5404-5411.
[11] GHAZVININEJAD M, LEVY O, LIU Y, et al. Mask-predict: Parallel decoding of conditional masked language models[C]// Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. 2019: 6112-6121.
[12] SHAO C, MA Z, ZHANG M, et al. Beyond MLE: convex learning for text generation[J]. Advances in Neural Information Processing Systems, 2023, 36: 8913-8936.
[13] GONG S, LI M, FENG J, et al. Diffuseq: Sequence to sequence text generation with diffusion models[J]. arXiv: 2210.08933, 2022.
[14] CETTOLO M, NIEHUES J, STÜKER S, et al. Report on the 11th IWSLT evaluation campaign[C]// Proceedings of the 11th International Workshop on Spoken Language Translation: Evaluation Campaign. 2014: 2-17.
[15] BOJAR O, CHATTERJEE R, FEDERMANN C, et al. Findings of the 2016 conference on machine translation (wmt16)[C]// First Conference on Machine Translation. Association for Computational Linguistics. 2016: 131-198.
[16] BOJAR O, BUCK C, FEDERMANN C, et al. Findings of the 2014 workshop on statistical machine translation[C]// Proceedings of the 9th Workshop on Statistical Machine Translation. 2014: 12-58.
[17] HUANG X S, PEREZ F, VOLKOVS M. Improving non-autoregressive translation models without distillation[C]// International Conference on Learning Representations. 2022.
[18] HUANG S, DONG L, WANG W, et al. Language is not all you need: Aligning perception with language models[M]// Advances in Neural Information Processing Systems. 2024.
[19] KASAI J, CROSS J, GHAZVININEJAD M, et al. Non-autoregressive machine translation with disentangled context transformer[C]// International Conference on Machine Learning. PMLR, 2020: 5144-5155.
[20] DIELEMAN S, SARTRAN L, ROSHANNAI A, et al. Continuous diffusion for categorical data[J]. arXiv: 2211.15089, 2022.
[21] AUSTIN J, JOHNSON D D, HO J, et al. Structured denoising diffusion models in discrete state-spaces[J]. Advances in Neural Information Processing Systems, 2021, 34: 17981-17993.
[22] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]// Proceedings of the 31st International Conference on in Neural Information Processing Systems. 2017: 6000-6010.
[23] PAPINENIK, ROUKOS S, WARD T, et al. Bleu: a method for automatic evaluation of machine translation[C]// Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. 2002: 311-318.
[24] DHINGRA B, MAZAITIS K, COHEN W W. Quasar: Datasets for question answering by search and reading[J]. arXiv: 1707.03904, 2017.
[25] SHARMA L, GRAESSER L, NANGIA N, et al. Natural language understanding with the quora question pairs dataset[J]. arXiv: 1907.01041, 2019.
[26] ZHANG B, XIONG D, SU J. A GRU-gated attention model for neural machine translation[J]. IEEE Transactions on Neural Networks and Learning Systems, 2020, 31(11): 4688-4698.
[27] RAFFEL C, SHAZEER N, ROBERTS A, et al. Exploring the limits of transfer learning with a unified text-to-text transformer[J]. Journal of Machine Learning Research, 2020, 21(140): 1-67.
[28] GU J, WANG C, ZHAO J. Levenshtein transformer[C]// Proceedings of the 33rd International Conference on Neural Information Processing Systems. Red Hook, NY: Curran Associates Inc., 2019: 11181-11191.
[29] LIN C Y. Rouge: A package for automatic evaluation of summaries[C]// Text Summarization Branches Out. ACL, 2004: 74-81.
[30] ZHANG T, KISHORE V, WU F, et al. Bertscore: Evaluating text generation with bert[J]. arXiv: 1904.09675, 2019.
[31] DESHPANDE A, ANEJA J, WANG L, et al. Fast, diverse and accurate image captioning guided by part-of-speech[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 10695-10704.

LI Sihui, born in 1999, postgraduate, is a member of CCF (No. W1626G). Her main research interests include natural language processing and non-autoregressive language modelling.



CAI Guoyong, born in 1971, Ph.D, professor, Ph. D supervisor, is a distinguished member of CCF (No. 12524D). His main research interests include multimodal affective computing, trustable AI theory and techniques.

(责任编辑:喻藜)