HIFI-CODEC：用于高保真音频编解码器的组残差向量量化

Dongchao Yang[1*], *Songxiang Liu* [2*]、*Rongjie Huang*[3*]、*Jinchuan Tian*[1], *Chao Weng* [2]、*Yuexian Zou* [1]

[1] 北京大学，中国[2] 腾讯 AI 实验室，中国深圳[3] 浙江大学，中国

arXiv:2305.02765v2 [cs.SD] 7 May 2023

摘要

音频编解码模型作为一种将音频压缩为离散表示的关键技术，已广泛应用于音频通信领域。如今，音频编解码模型越来越多地被用作生成领域中的中间表示。例如，AudioLM 是一种音频生成模型，它使用 SoundStream 的离散表示作为训练目标；而 VALL-E 则采用 Encodec 模型作为中间特征，以辅助文本转语音（TTS）任务。尽管这些模型具有诸多优势，但仍存在两个挑战：（1）由于缺乏公开可用的训练流程，且需要大规模数据和图形处理器（GPU），训练这些音频编解码模型可能十分困难；（2）实现良好的重建性能需要大量码本，这增加了生成模型的负担。在本研究中，我们提出了一种组残差向量量化（GRVQ）技术，并利用该技术开发了一种新型高保真音频编解码模型 HiFi-Codec，该模型仅需 4 个码本。我们使用 LibriTTS、VCTK、AISHELL 等公开可用的文本转语音数据集训练所有模型，训练数据总时长超过 1000 小时，并使用 8 个图形处理器。实验结果表明，尽管仅需 4 个码本，HiFi-Codec 在重建性能方面仍优于 Encodec。为了推动音频编解码和生成领域的研究，我们推出了 AcademiCodec，这是首个开源音频编解码工具包，为 Encodec、SoundStream 和 HiFi-Codec 提供训练代码及预训练模型。代码和预训练模型可在以下地址获取：https://github.com/yangdongchao/AcademiCodec

索引词— 向量量化，音频编解码器，音频生成

1. 引言

音频编解码器的目的是减少存储或传输音频信号所需的数据量，同时避免显著降低其质量。音频编解码器的基本原理是从音频信号中去除冗余或无关信息。音频编解码器有许多不同类型。音频编解码器种类繁多，各有优缺点。有些编解码器专为实时应用而设计，例如电话通信或流式音频，并优先考虑低延迟和低比特率。另一些则专为高品质音频制作而设计，并优先考虑保真度和准确性。除了压缩之外，许多音频编解码器还包含错误校正、降噪和动态范围压缩等特征。这些特征有助于提升音频信号的质量和可靠性，尤其是在嘈杂或带宽较低等具有挑战性的环境中。

在本研究中，我们着重探讨如何利用音频编解码模型，帮助解决音频相关领域中的生成问题，例如文本转语音（TTS）[1–4], 音乐生成 [5], 音频生成 [6–10]。Encodec [11] 和 SoundStream [12] 与我们的工作最为相关。Encodec 和 SoundStream 的核心技术是残差向量量化（RVQ），它使用多个向量量化码本来表示中间特征。Encodec 和 SoundStream 均采用编码器—解码器框架。编码器首先将波形压缩为紧凑的深层表示，然后使用残差向量量化对中间特征进行量化。最后，解码器根据量化后的表示恢复波形。

在我们的实验中，我们发现使用残差向量量化时，大部分信息都保存在第一个码本中，例如，对于语音数据，只使用一个码本就可以恢复文本信息和音色信息。后续码本保存了一些细节信息，这些信息可能会影响音频质量。然而，这些信息在隐空间中稀疏且分散。因此，这种量化方式需要许多码本才能实现良好的重建性能。例如，Encodec 需要 12 个码本才能实现高质量的重建性能。在音频生成领域，使用多个码本会给生成模型带来负担，例如，Transformer 难以对长序列进行建模。

在本研究中，我们专注于为生成任务设计音频编解码模型。我们的目标主要有两个：1）良好的重建性能；2）只需少量码本。为实现这些目标，我们提出了分组残差向量量化（GRVQ）方法。对于任意潜在特征$\boldsymbol{z} \in \mathbb{R}^N$，我们首先将 $\boldsymbol{z}$ 拆分为若干组，例如 $\{\boldsymbol{z}_1, \boldsymbol{z}_2\}$。

Dongchao Yang、Songxiang Liu 和 Rongjie Huang 是本项目的主要贡献者。相关工作是在腾讯 AI 实验室实习期间完成的

然后，我们分别使用残差向量量化（RVQ）对 z_1 和 z_2 进行量化。最后，我们结合两组 RVQ 的信息来解码波形。我们的动机是，希望第一层的码本能够保存更多信息，这样我们就可以使用更少的残差块。当我们将特征拆分为若干组，并在第一层使用更多码本时，这些码本便能在压缩过程中发挥更重要的作用。在实验中，我们发现，将特征拆分为 2 组并使用 2 个残差层，相较于预训练的 Encodec 模型 [1]，能够带来更好的重建性能。

2. 相关工作

2.1. 音频表征学习。

自监督学习（SSL） [13, 14] 已在音频相关领域取得巨大成功，例如自动语音识别（ASR）和音频压缩 [11,12]。受向量量化（VQ）技术 [15], 的启发，SoundStream [12]提出了一种残差向量量化（RVQ）架构，用于生成承载语义信息的高层表征。类似地，许多研究 [6, 16] 尝试使用 VQ-VAE 模型，将时频谱图（例如梅尔频谱图）压缩为高层表征。

2.2. 使用音频编解码器进行语音与音频生成

近期，许多研究 [1, 2, 5, 7, 8, 17, 18] 提出借助音频编解码器，在离散潜在空间中对语音和音频进行建模。其核心思想是使用音频编解码模型将语音或声音压缩为一组离散词元，然后使用生成模型生成这些词元。例如 AudioLM [8]利用音频编解码模型将波形编码为离散词元，然后使用语言模型（LM）对生成过程进行建模。类似地，InstructTTS[2] 使用离散扩散模型生成离散词元，再使用音频编解码模型还原波形。

2.3. 音频编解码器

对低码率参数化音频编解码器的研究可追溯至 [19, 20], ，但其质量往往有限。近期，研究人员提出了多种基于神经网络的音频编解码器，并取得了令人期待的结果 [11, 12, 21–24]。这些方法通常使用编码器在潜在空间中提取深层特征，随后对其进行量化，再输入解码器。与我们最相关的工作是 Sound-Stream [12] 和 Encodec [11] 模型。这些方法提出使用带有残差向量量化（RVQ） [25,26] 层的全卷积编码器-解码器架构。这些模型通过重建损失和对抗感知损失进行优化。为了加快处理过程

[1] https://github.com/facebookresearch/encodec

算法1： 组残差向量量化

输入： $\boldsymbol{y} = encoder(x)$ 编码器的输出，
向量量化器 Q_i 用于 $i = 1..2 * N_q$
输出： 量化后的 $\hat{\boldsymbol{y}}$

$\hat{y}_1 \leftarrow 0.0$
$\hat{y}_2 \leftarrow 0.0$
将 $\boldsymbol{y}$ 拆分为两组；
残差$_1 \leftarrow \boldsymbol{y}_1$
残差$_2 \leftarrow \boldsymbol{y}_2$
对于 $i = 1$到N_q **do**
  $\hat{y}_1 \mathrel{+}= Q_i$(残差$_1$)
  残差$_1 \mathrel{-}= Q_i$(残差$_1$)
for $i = N_q$ 至 $2 * N_q$ **do**
  $\hat{y}_2 \mathrel{+}= Q_i$(残差$_2$)
  残差$_2 \mathrel{-}= Q_i$(残差$_2$)
$\hat{\boldsymbol{y}} = concat(\hat{\boldsymbol{y}}_1, \hat{\boldsymbol{y}}_2)$
return $\hat{y}$

在压缩和解压缩方面，Encodec 提出了一种用于预测词元的语言模型方法。MQ-TTS [17] 也提出使用多个码本对中间特征进行量化，但该方法假设说话人信息可通过说话人标签明确获得，且未使用残差向量量化来保留更多音频信息。尽管以往的工作在重建性能和压缩率方面取得了巨大成功，但由于需要使用许多码本才能维持良好的重建性能，这些方法可能不适用于生成任务。

3. 提出的方法

在本节中，我们将介绍 HiFi-Codec 的具体细节。首先介绍 HiFi-Codec 模型概述，然后讨论 HiFi-Codec 各个部分的具体细节。

3.1. 概述

我们考虑一个时长为 d 的单通道音频信号 $\boldsymbol{x}$，并将其表示为一个序列 $\boldsymbol{x} \in \mathcal{R}^T$，其中 $T = d * sr$，且 sr 为音频采样率。HiFi-Codec 模型由三个主要部分组成：(1) 编码器网络 E，用于接收输入音频并生成潜在特征表示$\boldsymbol{z}$；(2) 分组残差量化层 Q，用于生成压缩表示 $\boldsymbol{z}_q$；以及 (3) 解码器 G，用于从压缩潜在表示 $\boldsymbol{z}_q$ 重建音频信号 $\hat{\boldsymbol{x}}$。该模型采用端到端方式进行训练，通过优化同时作用于时域和频域的重建损失，以及以不同分辨率下运行的判别器形式实现的感知损失。所提出方法的可视化描述见图1。

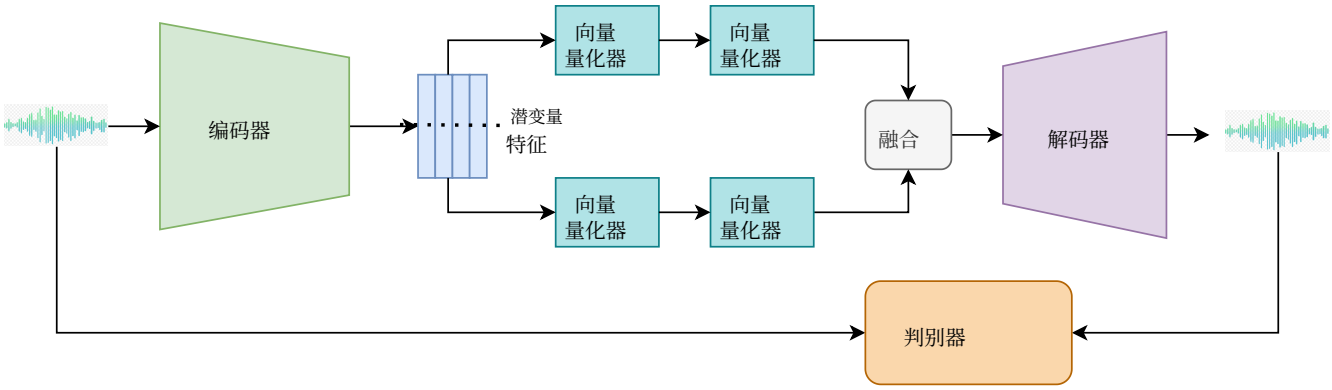


图 1：HiFi-Codec 模型概览。

3.2. 编码器和解码器

我们的编码器和解码器架构借鉴了 Encodec [11] 和 SoundStream [12]的设计。该架构基于卷积框架，并在编码器端和解码器端均针对潜在表示应用序列建模。编码器模型 E由一个具有 C 个通道、卷积核大小为 7 的一维卷积层构成，后接 B 个卷积块。每个卷积块包含一个残差单元，其后连接一个下采样层，该下采样层由一个步幅为 S、卷积核大小为 K （即步幅的两倍）的步幅卷积构成。残差单元由两个卷积核大小为 3 的卷积层和一个跳跃连接组成。每当进行下采样时，通道数便加倍。随后，卷积块后接一个用于序列建模的双层长短期记忆网络（LSTM），以及一个卷积核大小为 7、具有 D 个输出通道的最终一维卷积层。在本研究中，我们探索了 C、 B和S的不同设置，例如 $C = [32, 48, 64]$、$B = 4$和 $S =$ (2,4,5, 8)、(2, 4, 5, 6) 或 (2, 2, 2, 4)。解码器采用与编码器对称的架构，并使用转置卷积代替步幅卷积，其步幅顺序与编码器相反。解码器输出最终的音频信号。

3.3. 组残差向量量化（GRVQ）

在本研究中，我们希望设计一种包含更少量化器，同时具有良好重建性能的音频编解码模型。我们认为，RVQ 的一个缺点在于，RVQ 中第一层的码本能够保存大部分信息，而剩余的码本只能保存少量信息。因此，我们提出在第一层中增加更多码本。具体而言，对于任意潜在特征表示$\boldsymbol{z}$，我们首先将其平均划分为若干组（在本研究中，我们将 $\boldsymbol{z}$ 划分为两组， $\boldsymbol{z}_1$， $\boldsymbol{z}_2$），然后使用多个 RVQ 分别对每组特征进行量化。最后，我们将多个分组 RVQ 的输出合并，以获得最终的量化结果。整个过程可以总结为算法1。

3.4. 判别器

在本研究中，我们使用了三种判别器：一种基于多尺度短时傅里叶变换（MS-STFT）的判别器，用于 En- codec [11]；一种来自 HiFi-GAN 声码器 [27]的多周期判别器（MPD），以及一种多尺度判别器（MSD）。

对于 MS-STFT 判别器，它由结构相同的网络组成，作用于多尺度复值短时傅里叶变换，并将实部和虚部拼接在一起。我们为每个子网络采用与 Encodec 相同的配置：先经过一个二维卷积层，再经过若干个在时间维度上膨胀率递增（1、2 和 4）、并在频率轴上步幅为 2 的二维卷积。最后，使用一个卷积核大小为 3 x 3、步幅为 (1, 1) 的二维卷积生成最终预测。我们采用五种不同尺度，其 STFT 窗口长度为 [2048, 1024, 512, 256, 128]。对于多周期判别器和多尺度判别器，我们保持与 HiFi-GAN 相同的结构，但减少通道数，使判别器的参数量与 MS-STFT 判别器相近。

3.5. 训练损失

我们的方法基于 GAN目标，同时对生成器和判别器进行优化。具体而言，我们联合优化重建损失项、感知损失项（通过判别器实现）以及生成器的 GRVQ 承诺损失。生成器的训练目标由多个损失项组成，包括时域项、频域项、三个判别器损失，以及作为感知损失的相应特征损失项和 GRVQ 承诺损失。判别器损失基于对抗铰链损失函数。

3.5.1. 重建损失

我们的重建损失包含两个方面：（1）时域损失和（2）时频损失。对于时域损失，我们直接使用 L1距离损失来优化 $\boldsymbol{x}$ 和 $\hat{}\boldsymbol{x}$。

表 1：性能比较。

方法	采样率（K Hz）	下采样次数	码本数量	感知语音质量评估 ↑	短时客观可懂度 ↑
Encodec（Facebook）	24	320	8	3.01	0.94
Encodec（Facebook）	24	320	12	3.21	0.95
Encodec（我们的方法）	24	240	8	3.62	0.94
Encodec（我们的方法）	24	32	2	3.08	0.91
Encodec（我们的方法）	16	320	8	3.04	0.93
SoundStream（本文方法）	16	320	12	3.26	0.95
HiFi-Codec	24	240	4	3.63	0.95
HiFi-Codec	24	240	8	**3.92**	**0.95**
HiFi-Codec	24	320	4	3.64	0.95
HiFi-Codec	16	320	4	3.22	0.94

对于时频损失，我们采用与 Encodec 类似的方法，在多个时间尺度上对梅尔频谱图施加损失项。

3.5.2. 判别器损失

对抗损失用于提升感知质量。我们使用三种类型的判别器，其中 MS-STFT 判别器试图使频谱图级重建结果尽可能接近原始结果。MPD 和 MSD 判别器则试图使波形级重建结果尽可能接近原始结果。为了训练判别器，我们可以优化以下目标函数：

$$\mathcal{L}_d = \frac{1}{K}\sum_{i=1}^{K} max(0, 1 - D_k(\boldsymbol{x})) + max(0, 1 + D_k(\hat{\boldsymbol{x}})) \tag{1}$$

其中，K 表示判别器的数量。此外，我们可以将对抗损失定义为这些判别器逻辑值上的铰链损失：

$$\mathcal{L}_{adv} = \frac{1}{K}\sum_{i=1}^{K} max(0, 1 - D_k(\hat{\boldsymbol{x}})) \tag{2}$$

此外，特征损失通过计算判别器对生成音频和对应真实音频的内部层输出之间的平均绝对差值得出。

$$\mathcal{L}_{feat} = \frac{1}{KL}\sum_{k=1}^{K}\sum_{l=1}^{L} \frac{||D_k^l(\boldsymbol{x}) - D_k^l(\hat{\boldsymbol{x}})||_1}{mean(||D_k^l(\boldsymbol{x})||_1)} \tag{3}$$

3.5.3. GRVQ 承诺损失

对于第 i 组的第 c 个残差量化器，我们可以根据以下公式计算承诺损失：

$$\mathcal{L}_c = \sum_{i,c} ||\boldsymbol{z}_{i,c} - q_{i,c}(\boldsymbol{z}_{i,c})||_2^2 \tag{4}$$

基于之前的讨论，我们可以使用以下公式训练生成器。

$$Loss_G = \lambda_{\text{adv}}\mathcal{L}_{adv} + \lambda_{\text{feat}} \cdot \mathcal{L}_{feat} + \lambda_{\text{rec}} \cdot \mathcal{L}_{rec} + \lambda_c \cdot \mathcal{L}_c \tag{5}$$

其中，$\mathcal{L}_{adv}$ 表示对抗损失。$\mathcal{L}_{feat}$ 表示特征损失。$\mathcal{L}_{rec}$ 表示重建损失。λ_{adv}、λ_{feat}、λ_{rec} 和 λ_c 是用于控制训练目标函数的超参数。在我们的实验中，我们通过调整这些超参数的取值，尽量平衡各项损失。

4. 实验

4.1. 评估指标

在本研究中，我们通过测量重建音频与目标音频之间的差距来评估音频编解码模型的性能。我们采用语音增强领域的指标（如感知语音质量评估和短时客观可懂度）来评估性能。

4.2. 数据集

我们使用文本转语音数据集来训练音频编解码器模型。我们的训练数据来自公开数据集，例如 LibriTTS、VCTK 和 AISHELL，主要包含英语和中文语音。

4.3. 实验结果

表 1 展示了实验结果。可以看出，我们提出的 HiFi-Codec 仅使用 4 个码本，就实现了良好的重建性能。当下采样次数设置为 240、码本数量设置为 8 时（我们将每层设置为包含 4 个码本，并使用两个残差层），可以获得最佳性能。此外，我们复现的模型（Encodec 和 SoundStream）也取得了与 Encodec [11] 相当的性能。我们强烈建议读者在尝试训练生成模型时使用具有 4 个码本的 HiFi-Codec 模型。

5. 结论

在本研究中，我们提出了一种组残差向量量化方法，并构建了一种新型音频编解码模型：HiFi-Codec，该模型专为生成任务而设计。即使仅使用 4 个码本，HiFi-Codec 也能获得比 Encodec 更好的重建性能。此外，我们还公开了 Encodec 和 SoundStream 模型的训练流程，这有助于读者训练自己的编解码模型。未来，我们将继续优化 HiFi-Codec 模型，并尝试训练性能更好的 Encodec 和 SoundStream 模型。我们希望本项目能够推动音频生成任务领域的研究。

6. 局限性

尽管 HiFi-Codec 模型的重建性能优于 Encodec 模型，但仍存在一些局限性。(1) 我们没有使用大规模数据集来训练通用音频编解码器，因此其泛化能力尚未得到充分验证。(2) 我们发现，客观评估指标可能无法准确评估重建性能。主观评估始终是最佳选择，但本研究未包含这一部分。(3) HiFi-Codec 旨在助力生成任务，但我们没有提供足够的下游任务来评估其性能。我们将在未来的工作中继续探索这一方向。

7. 参考文献

[1] Chengyi Wang、Sanyuan Chen、Yu Wu、Ziqiang Zhang、Long Zhou、Shujie Liu、Zhuo Chen、Yanqing Liu、Huaming Wang、Jinyu Li 等人，《神经编解码器语言模型是零样本文本转语音合成器》，*arXiv* 预印本 *arXiv:2301.02111*，2023年。[2]Dongchao Yang、Songxiang Liu、Rongjie Huang、Guangzhi Lei、Chao Weng、Helen Meng 和 Dong Yu，《Instructtts：在离散潜在空间中使用自然语言风格提示对富有表现力的文本转语音进行建模》，*arXiv* 预印本 *arXiv:2301.13662*，2023年。

[3] RongjieHuang、Max WY Lam、Jun Wang、Dan Su、Dong Yu、Yi Ren 和 Zhou Zhao，《Fastdiff：一种用于高质量语音合成的快速条件扩散模型》，*arXiv* 预印本 *arXiv:2204.09934*，2022年。

[4] Rongjie Huang、Zhou Zhao、Huadai Liu、Jinglin Liu、Chenye Cui 和 Yi Ren，《Prodiff：用于高质量文本转语音的渐进式快速扩散模型》，收录于第*30*届 *ACM*多媒体国际会议论文集，2022年，第2595–2605页。[5] Andrea Agostinelli、Timo I Denk、Zal´an Borsos、Jesse Engel、Mauro Verzetti、Antoine Caillon、Qingqing Huang、Aren Jansen、Adam Roberts、Marco Tagliasacchi 等，《Musiclm：根据文本生成音乐》，*arXiv* 预印本 *arXiv:2301.11325*，2023年。

[6] Dongchao Yang、Jianwei Yu、Helin Wang、Wen Wang、Chao Weng、Yuexian Zou 和 Dong Yu，《Diffsound：用于文本到声音生成的离散扩散模型》，*arXiv*预印本 *arXiv:2207.09983*，2022年。

[7] Felix Kreuk、Gabriel Synnaeve、Adam Polyak、Uriel Singer、Alexandre D´efossez、Jade Copet、DeviParikh、Yaniv Taigman 和 Yossi Adi，“Audiogen：文本引导的音频生成”，*arXiv* 预印本 *arXiv:2209.15352*，2022年。

[8] Zal´an Borsos, Rapha¨el Marinier, Damien Vincent, Eu- gene Kharitonov, Olivier Pietquin, Matt Sharifi, Olivier Teboul, David Grangier, Marco Tagliasacchi 和 Neil Zeghidour，“Audiolm：一种用于音频生成的语言建模方法”，*arXiv* 预印本 *arXiv:2209.03143*，2022年。

[9] Rongjie Huang、Jiawei Huang、Dongchao Yang、Yi Ren、Luping Liu、Mingze Li、Zhenhui Ye、Jinglin Liu、Xiang Yin 和 Zhou Zhao，“Make-an-audio：利用提示增强扩散模型进行文本到音频生成”，*arXiv* 预印本 *ar Xiv:2301.12661*，2023年。

[10] Rongjie Huang、Zhou Zhao、Jinglin Liu、Huadai Liu、Yi Ren、Lichao Zhang 和 Jinzheng He，《Transpeech：采用双边扰动的语音到语音翻译》，*arXiv* 预印本 *arXiv:2205.12523*，2022年。

[11] Alexandre D´efossez、JadeCopet、Gabriel Synnaeve 和 Yossi Adi，《高保真神经音频压缩》，*arXiv* 预印本 *arXiv:2210.13438*，2022年。

[12] Neil Zeghidour、Alejandro Luebs、Ahmed Omran、Jan Skoglund 和 Marco Tagliasacchi，《Soundstream：一种端到端神经音频编解码器》，*IEEE/ACM*音频、语音与语言处理汇刊，第30卷，第495–507页，2021年。

[13] Alexei Baevski、Yuhao Zhou、Abdelrahman Mohamed 和 Michael Auli，《wav2vec 2.0：一种用于语音表征自监督学习的框架》，《神经信息处理系统进展》，第 33 卷，第 12449–12460 页，2020年。

[14] Alexei Baevski、Wei-Ning Hsu、Qiantong Xu、Arun Babu、Jiatao Gu 和 Michael Auli，《Data2vec：一种用于语音、视觉和语言自监督学习的通用框架》，载于国际机器学习会议。PMLR，2022年，第1298–1312页。

[15] Aaron Van Den Oord、Oriol Vinyals 等，《神经离散表示学习》，《神经信息处理系统进展》，第 30 卷，2017年。

[16] Vladimir Iashin 和 Esa Rahtu，《驯服视觉引导的声音生成》，载于英国机器视觉会议（*BMVC*），2021年。

[17] Li-Wei Chen、Shinji Watanabe 和 Alexander Rudnicky，“基于真实世界自发语音的文本转语音合成矢量量化方法”，*arXiv*预印本 *arXiv:2302.04215*，2023年。[18] Rongjie Huang、Mingze Li、Dongchao Yang、Jiatong Shi、Xuankai Chang、Zhenhui Ye、Yuning Wu、Zhiqing Hong、Jiawei Huang、Jinglin Liu 等，“Audiogpt：理解并生成语音、音乐、声音和会说话的人头”，*arXiv*预印本 *arXiv:2304.12995*，2023年。[19] Bishnu S Atal 和 Suzanne L Hanauer，“通过语音波的线性预测进行语音分析与合成”，《美国声学学会杂志》，第50卷，第2B期，第637–655页，1971年。[20] Biing-Hwang Juang 和 A Gray，“用于语音编码的多级矢量量化”，载于*ICASSP’82. IEEE*声学、语音与信号处理国际会议。IEEE，1982年，第7卷，第597–600页。[21] W Bastiaan Kleijn、Felicia SC Lim、Alejandro Luebs、Jan Skoglund、Florian Stimberg、Quan Wang 和 Thomas C Walters，“基于WaveNet的低码率语音编码”，载于*2018*年*IEEE*声学、语音与信号处理国际会议（*ICASSP*）。IEEE，2018年，第676–680页。[22] Jean-Marc Valin 和 Jan Skoglund，“使用LPCNet实现1.6 kb/s实时宽带神经声码器”，*arXiv*预印本 *arXiv:1903.12087*，2019年。[23] Ahmed Omran、Neil Zeghidour、Zal´an Borsos、F´elix de Chaumont Quitry、Malcolm Slaney 和 Marco Tagliasacchi，“在神经音频编解码器中分离语音与周围环境”，*arXiv*预印本 *arXiv:2203.15578*，2022年。[24] Tejas Jayashankar、Thilo Koehler、Kaustubh Kalgaonkar、Zhiping Xiu、Jilong Wu、Ju Lin、Prabhav Agrawal 和 Qing He，“具有可配置计算复杂度的可变比特率神经语音编解码器架构”，载于*ICASSP 2022-2022 IEEE*声学、语音与信号处理国际会议（*ICASSP*）。IEEE，2022年，第861–865页。[25] Robert Gray，“向量量化”，《*IEEE*信号处理学会杂志》，第1卷，第2期，第4–29页，1984年。[26] A Vasuki 和 PT Vanathi，“向量量化技术综述”，《*IEEE Potentials*》，第25卷，第4期，第39–47页，2006年。[27] Jungil Kong、Jaehyeon Kim 和 Jaekyoung Bae，“Hifi-gan：用于高效高保真语音合成的生成对抗网络”，神经信息处理系统进展，第33卷，第17022–17033页，2020年。