Recent Advances on Unified Model for Voice and Audio Generation (Part 1)

2023-11-13

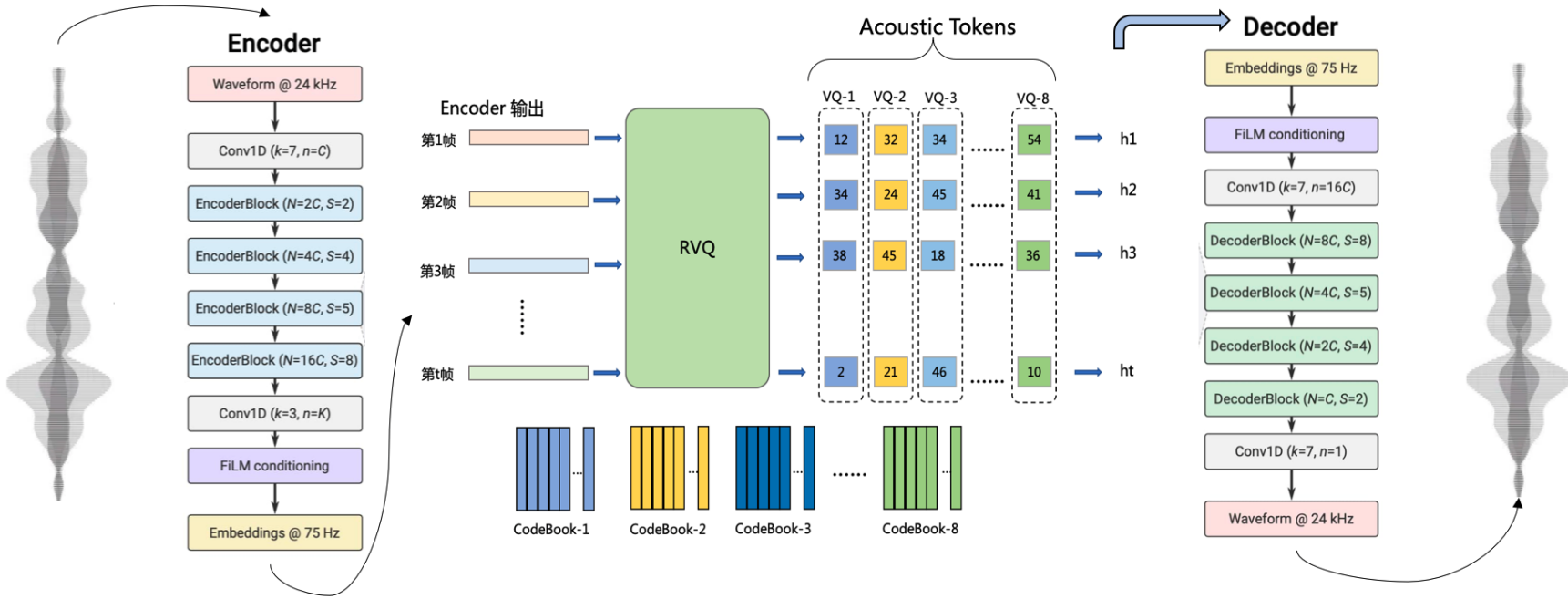
Outline

- **Part 1:** Purely LLM based Audio Generation
  - Speech-X, **Make-A-Voice / MVoice**
  - UniAudio
- **Part 2:** Incorporating Diffusion into Audio Generation
  - ☆ **AudioLDM 2**
- **Discussion :** Beyond Unified Model for Audio Generation
  - Foundation Model for Both Audio Understanding and Generation
  - Towards Final Fantasy: Any-to-Any Generation
- Appendix: Understanding Diffusion Models

Purely LLM based Audio Generation

Background

Speech/Audio Tokenization（SoundStream）

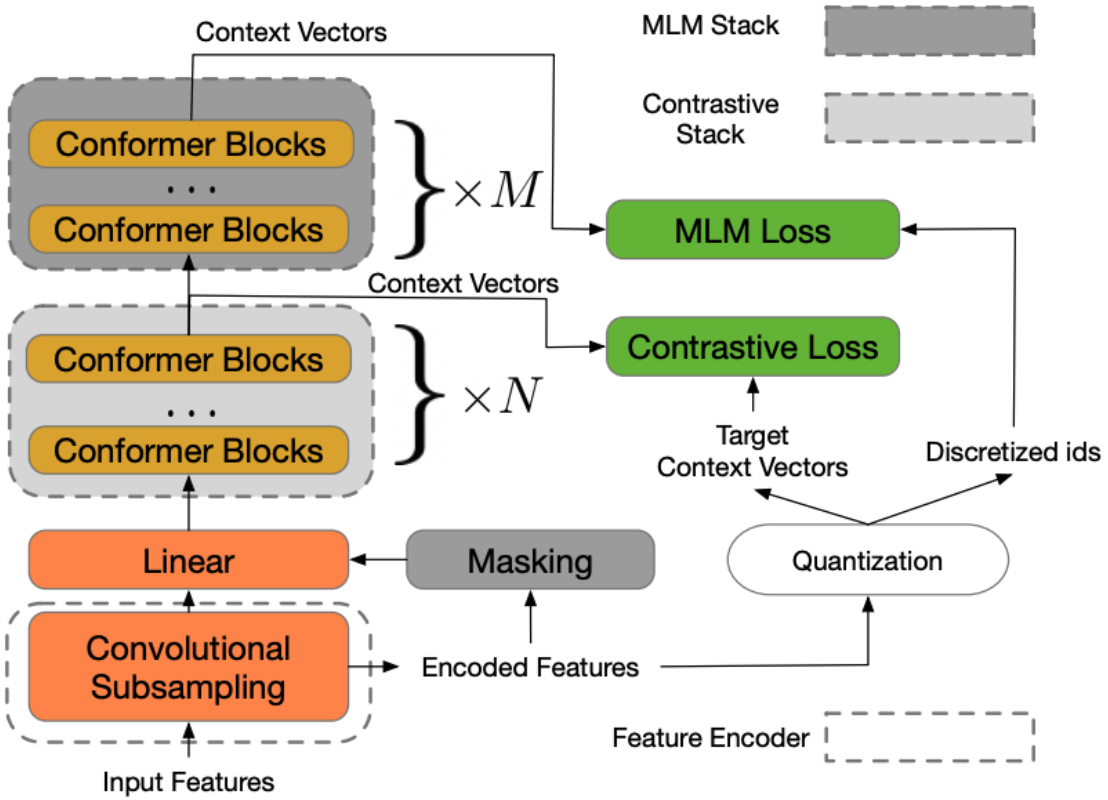


Acoustic Token

- 模型结构：RVQ-GAN
- 侧重声学层面，语音信号本身特性（音色）的细节
- 出发点：降低语音传输码率，目前主要用于各类任务下的音频离散化表征

Background

Speech/Audio Tokenization（wav2vec-BERT）



Semantic Token

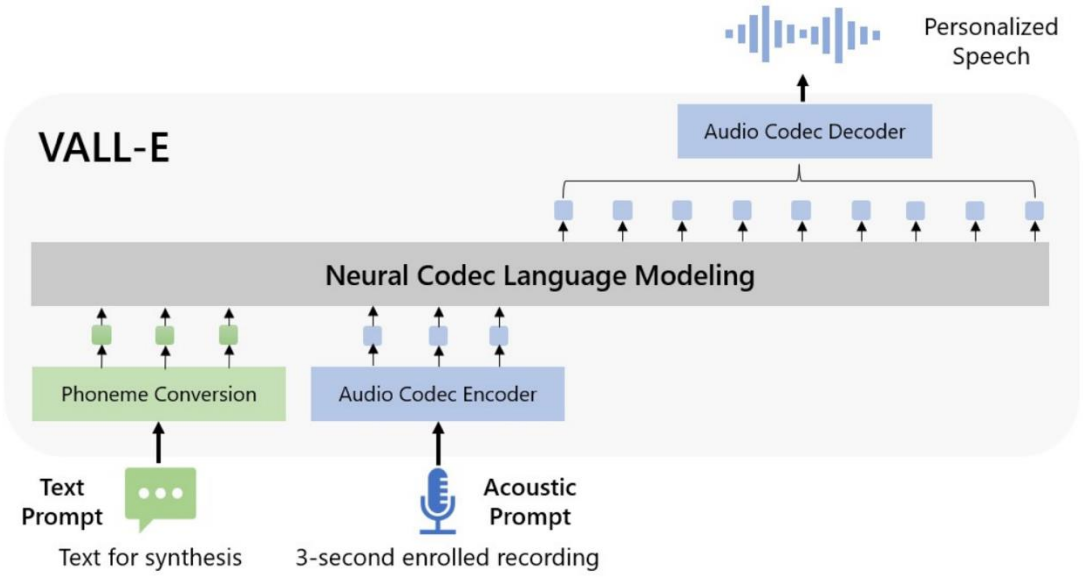
- 偏重文本语义层面，不关注波形的细节信息

Wav2vec-Bert 提取方法

- 抽取位置：MLM 部分的第 7 层输出
- 先对输出 embedding 进行正则化（均值为 0，方差为 1）
- 预设聚类个数 K（1024）进行 k-means 聚类
- 聚类后类别 id 作为离散的语义 token

Background

Background: LLM based TTS（VALL-E）



LLM 类模型的设计思路

- 基于 decoder-only Transformer 的结构
- 输入模态（文本）和输出模态（语音）序列进行拼接，需要根据目标任务设计拼接方式（比如：拼接顺序、添加特殊标记符等）
  - 文本已经是离散化的 token
  - 语音/音频需要离散化（Codec）

训练阶段

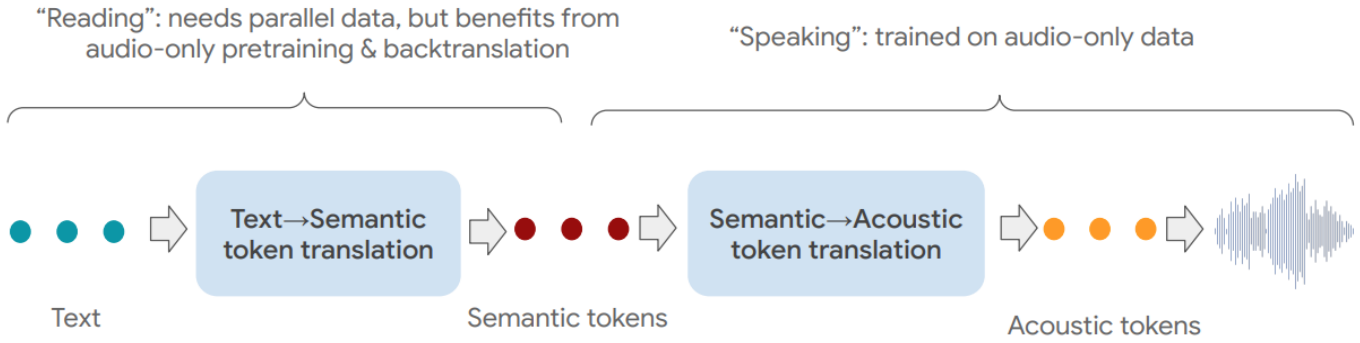
- 不区分（区分） prompt 和非 prompt 部分
- 完全采用 teacher-forcing 的方式训练

推理阶段

- 将 condition按照既定的序列拼接方式作为 prompt，自回归生成目标序列

Background

Background: LLM based TTS（Spear-TTS）



Regeneration Learning

将 $X \rightarrow Y$ 的任务拆解为 $X \rightarrow Y' \rightarrow Y$

- $X \rightarrow Y'$ 的一对多建模难度显著更低
- $Y' \rightarrow Y$ 是自监督训练，不需数据标注

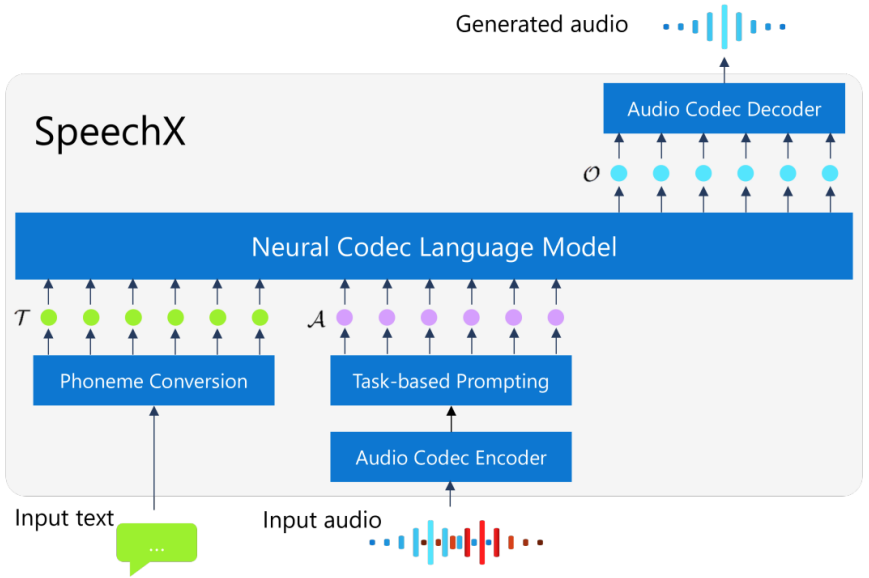
Spear-TTS：将 $X \rightarrow Y$ 的任务进一步拆解为 $X \rightarrow Y_1 \rightarrow Y_2 \rightarrow Y$

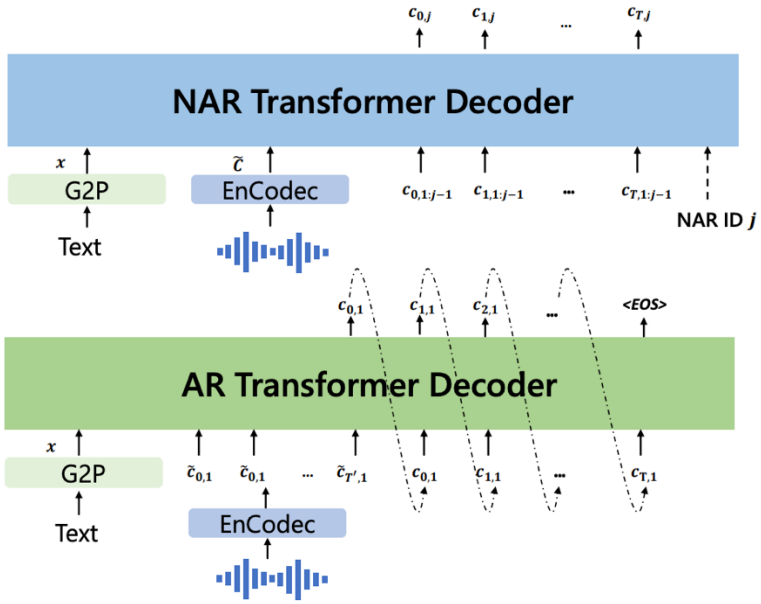
其中：X：文本，Y：音频波形，Y_1：semantic token，Y_2：acoustic token

- $X \rightarrow Y_1$：semantic token 去除了音色/录音环境等细节信息，缓解 one-to-many 问题，有监督训练 **优势一**
- $Y_1 \rightarrow Y_2$：semantic token 和 acoustic token 都是从语音中提取，可以用大量数据进行自监督训练，无需标注 **优势二**
- $Y_2 \rightarrow Y$：acoustic token 和波形之间使用 codec 建模，codec 本身的训练也是大量数据自监督，无需标注 **优势二**

Purely LLM based Audio Generation

Speech-X: Versatile VALL-E





Conditional Codec Language Modeling

- 文本 → 语音：TTS
- 语音 → 语音：降噪/去除人声/目标人声抽取/..
- 文本+语音 → 语音：语音编辑/...
- 文本 prompt：$\mathcal{T}$
- 语音 prompt：$\mathcal{A}$

Encodec $\mathcal{O} = [o_{t,l}] \in \mathbb{N}^{T \times L}$

$$\mathcal{L}_{AR} = -\sum_{t=1}^{T} \log P(o_{t,1} | \mathcal{T}, \mathcal{A}, \mathbf{o}_{<t,1}; \theta_{AR})$$

$$\mathcal{L}_{NAR} = -\sum_{l=2}^{8} \log P(\mathbf{o}_{:,l} | \mathcal{T}, \mathcal{A}, \mathbf{o}_{:,<l}; \theta_{NAR})$$

Purely LLM based Audio Generation

Speech-X: Versatile VALL-E

- Design Task-Specific Prompting
  - 任务标记 + condition 输入 + 目标输出（人工设计顺序，注意分隔符）

Task	Textual prompt $\mathcal{T}$	Acoustic prompt $\mathcal{A}$	Desired output $\mathcal{O}$
Noise suppression	G2P(text) / null	<ns>, $\mathrm{C}(s+n)$	$\mathrm{C}(s)$
Speech removal	G2P(text) / null	<sr>, $\mathrm{C}(s+n)$	$\mathrm{C}(n)$
Target speaker extraction	G2P(text) / null	$\mathrm{C}(s'_1)$, <tse>, $\mathrm{C}(s_1+s_2)$	$\mathrm{C}(s_1)$
Zero-shot TTS	G2P(text)	$\mathrm{C}(s)$	$\mathrm{C}(s')$
Clean speech editing	G2P(text)	$\mathrm{C}(s_{\text{pre}})$, <soe>, <mask>, <eoe>, $\mathrm{C}(s_{\text{post}})$	$\mathrm{C}(s_{\text{pre}}), \mathrm{C}(s_{\text{edit}}), \mathrm{C}(s_{\text{post}})$
Noisy speech editing	G2P(text)	$\mathrm{C}(s_{\text{pre}}+n_{\text{pre}})$, <soe>, $\mathrm{C}(s_{\text{mid}}+n_{\text{mid}})$, <eoe>, $\mathrm{C}(s_{\text{post}}+n_{\text{post}})$	$\mathrm{C}(s_{\text{pre}}+n_{\text{pre}}), \mathrm{C}(s_{\text{edit}}+n_{\text{mid}}), \mathrm{C}(s_{\text{post}}+n_{\text{post}})$

Model	Noise suppression			Target speaker extraction			Zero-shot TTS		Clean speech editing		Noisy speech editing		Speech removal
	WER↓	DNSMOS↑	PESQ↑	WER↓	DNSMOS↑	PESQ↑	WER↓	SIM↑	WER↓	SIM↑	WER↓	SIM↑	MCD↓
No processing	3.29	2.42	1.93	12.55	3.04	2.27	1.71	1.00	38.29	0.96	42.48	0.87	12.57
Expert model	**DCCRN [39], [40]**			**VoiceFilter [22]**			**VALL-E [16]**		**A3T [20]**		**A3T [20]**		**N/A**
	6.39	3.25	3.52	5.09	3.39	2.90	5.90	0.57	17.17	0.29	32.17	0.18	
SpeechX (random init.)	2.56	3.05	2.24	3.12	3.46	2.27	5.40	0.57	8.10	0.75	15.33	0.64	3.04
SpeechX (VALL-E init.)	2.48	3.05	2.24	2.53	3.46	2.28	4.66	0.58	5.63	0.76	13.95	0.65	3.05

Purely LLM based Audio Generation

Speech-X: Versatile VALL-E

- 增加文本 Prompt 提高模型效果

TABLE III
RESULTS OF NOISE SUPPRESSION AND TARGET SPEAKER EXTRACTION WITH OR WITHOUT TEXTUAL PROMPT.

Prompt	Noise suppression			Target speaker extraction		
	WER↓	DNSMOS↑	PESQ↑	WER↓	DNSMOS↑	PESQ↑
w/ text	2.48	3.05	2.24	2.53	3.46	2.28
w/o text	6.76	3.05	2.20	5.00	3.01	2.23

- 多任务同时训练带来的影响

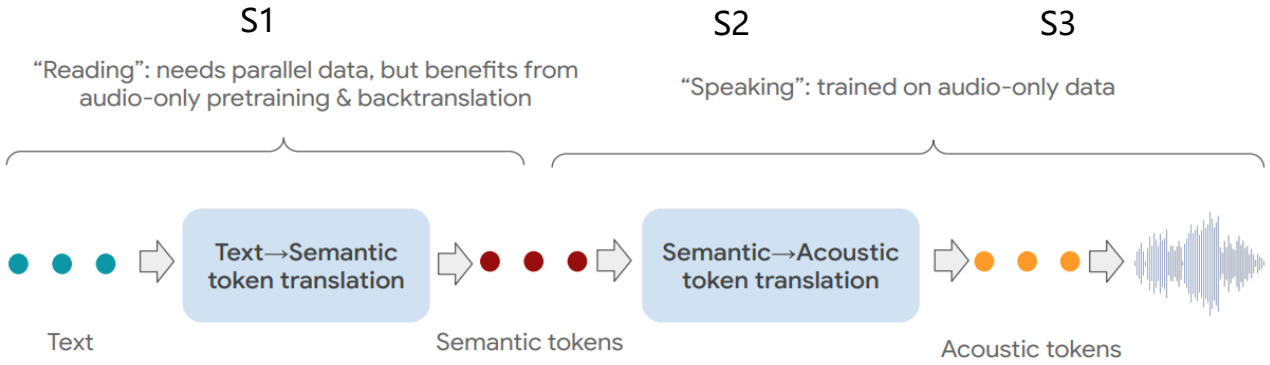
TABLE IV
EFFECTS OF ADDING TASKS DURING TRAINING. ZS: ZERO-SHOT, SE: SPEECH EDITING, NS: NOISE SUPPRESSION, SR: SPEECH REMOVAL, TSE: TARGET SPEAKER EXTRACTION.

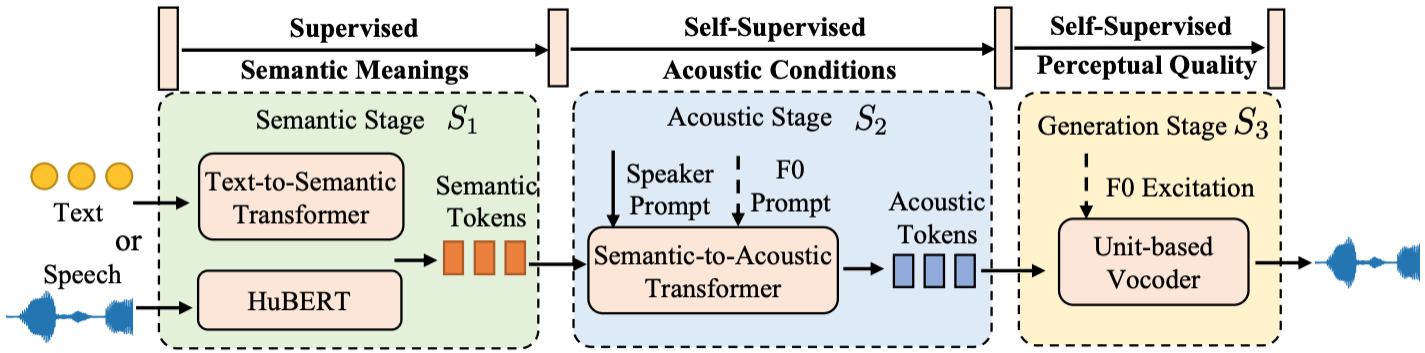
Training tasks	Zero-shot TTS		Speech editing (clean/noisy)		Noise suppression		Speech removal	Target speaker extraction	
	WER↓	SIM↑	WER↓	SIM↑	WER↓	DNSMOS↑	MCD↓	WER↓	DNSMOS↑
ZS-TTS	5.90	0.57	-	-	-	-	-	-	-
ZS-TTS + SE	4.55	0.58	5.79 / 13.80	0.76 / 0.65	-	-	-	-	-
ZS-TTS + SE + NS/SR	5.11	0.57	6.91 / 13.23	0.77 / 0.66	2.59	3.03	3.04	-	-
ZS-TTS + SE + NS/SR + TSE	4.66	0.58	5.63 / 13.95	0.76 / 0.65	2.48	3.05	3.05	2.53	3.46

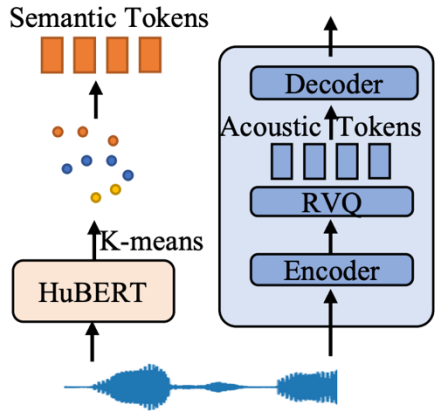
引入与噪声相关的任务，产生负面影响 →

Purely LLM based Audio Generation

Make-A-Voice: Extension of Spear-TTS







训练音频： 16 kHz

Semantic Token

- HuBert 第 12 层，1000 k-means 聚类，帧率 50 Hz
- 中英文各用各的 Hubert 模型，再一起聚类

SoundStream

- 12 个量化层，每层 codebook size 1024，帧率 50 Hz
- 训练时只用到了前 3 层，每帧按顺序摊开为 3 个 token

Purely LLM based Audio Generation

Make-A-Voice: Extension of Spear-TTS

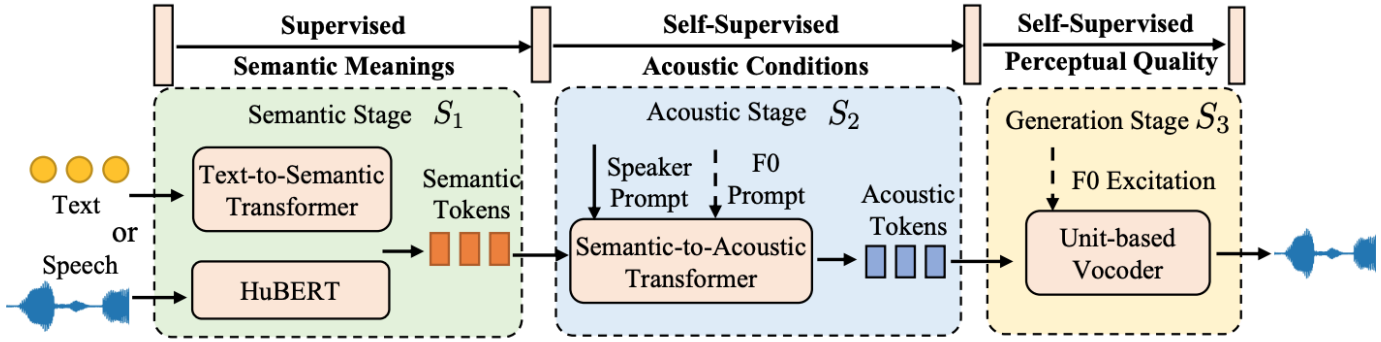
分阶段建模的第三点优势

- 语义（内容/韵律）与声学（音色/录音环境）的解耦

S1 阶段：seq2seq（phoneme → semantic token）

S2 阶段：decoder-only LM

- 选择同一条音频两个不重叠的窗，分别作为 prompt 和 target



语音任务：语音合成/语音转换

- LM 序列设计：**[prompt acoustic token] | [target semantic token]** | [target acoustic token]

→ VC 任务不需要 pair 的音频

$$p\left(\mathbf{a} \mid \mathbf{s}, \mathbf{a_p}; \theta_{AR}\right) = \prod_{t=0}^{T} p\left(\mathbf{a}_t \mid \mathbf{a}_{<t}, \mathbf{s}, \mathbf{a_p}; \theta_{AR}\right)$$

歌声任务：歌声合成

- 歌声合成需要更精确的 pitch 控制，要求更强的 condition 信息，引入了 F0 prompt
- F0 序列（帧级别）使用 YAAPT 算法从 target 音频（歌声）中提取，并在 log 尺度下进行 256 个数的整数量化
- LM 序列设计：**[prompt acoustic token] | [prompt F0] | [target semantic token]** | [target acoustic token]

→
- 训练时直接从 target 提取
- 推理时从输入的 MIDI 转换出 F0 序列

$$p\left(\mathbf{a} \mid \mathbf{s}, \mathbf{a_p}, \mathbf{F}; \theta_{AR}\right) = \prod_{t=0}^{T} p\left(\mathbf{a}_t \mid \mathbf{a}_{<t}, \mathbf{s}, \mathbf{a_p}, \mathbf{F}; \theta_{AR}\right)$$

Purely LLM based Audio Generation

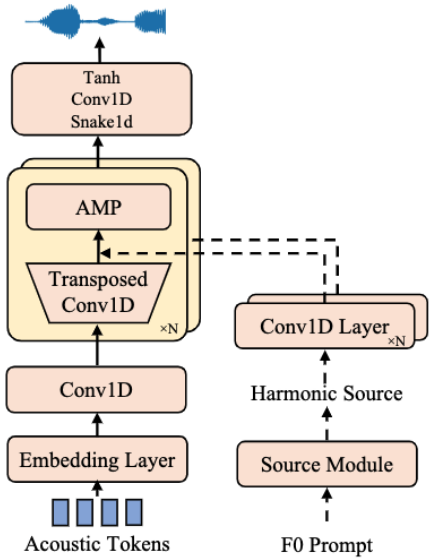
Make-A-Voice: Extension of Spear-TTS

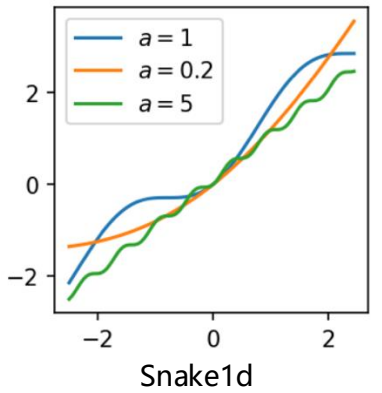
S3 阶段：Acoustic Token → Wave

- **问题**：推理时只能生成前 3 层 acoustic token，直接用 SoundStream 的 Decoder，波形恢复效果较差
- **解决方案**：强行用前 3 层 acoustic token 恢复波形细节（Unit Vocoder）
- Unit Vocoder 的输入是 Acoustic Token（3 层），输出是波形（Vocoder 能够恢复出细节）
  - 思考：直接只训练 3 层的 SoundStream 会更好吗？
- 模型结构主要参考的是 BigVGAN
  - 核心改进：激活函数引入周期性归纳偏置 → Snake1d

$$\text{Snake}_a := x + \frac{1}{a}\sin^2(ax) = x - \frac{1}{2a}\cos(2ax) + \frac{1}{2a}$$

  - a 越大，可以建模更高频的周期变化；a 是一组可学习参数，不同 channel 对应不同的参数
- **语音任务：语音合成/语音转换**
  - 不需要额外信息，直接根据 S2 预测出的 acoustic token 恢复波形
- **歌声任务：歌声合成**
  - F0 序列也作为额外的输入，用于提高生成歌声的稳定性，避免 glitches (小瑕疵)

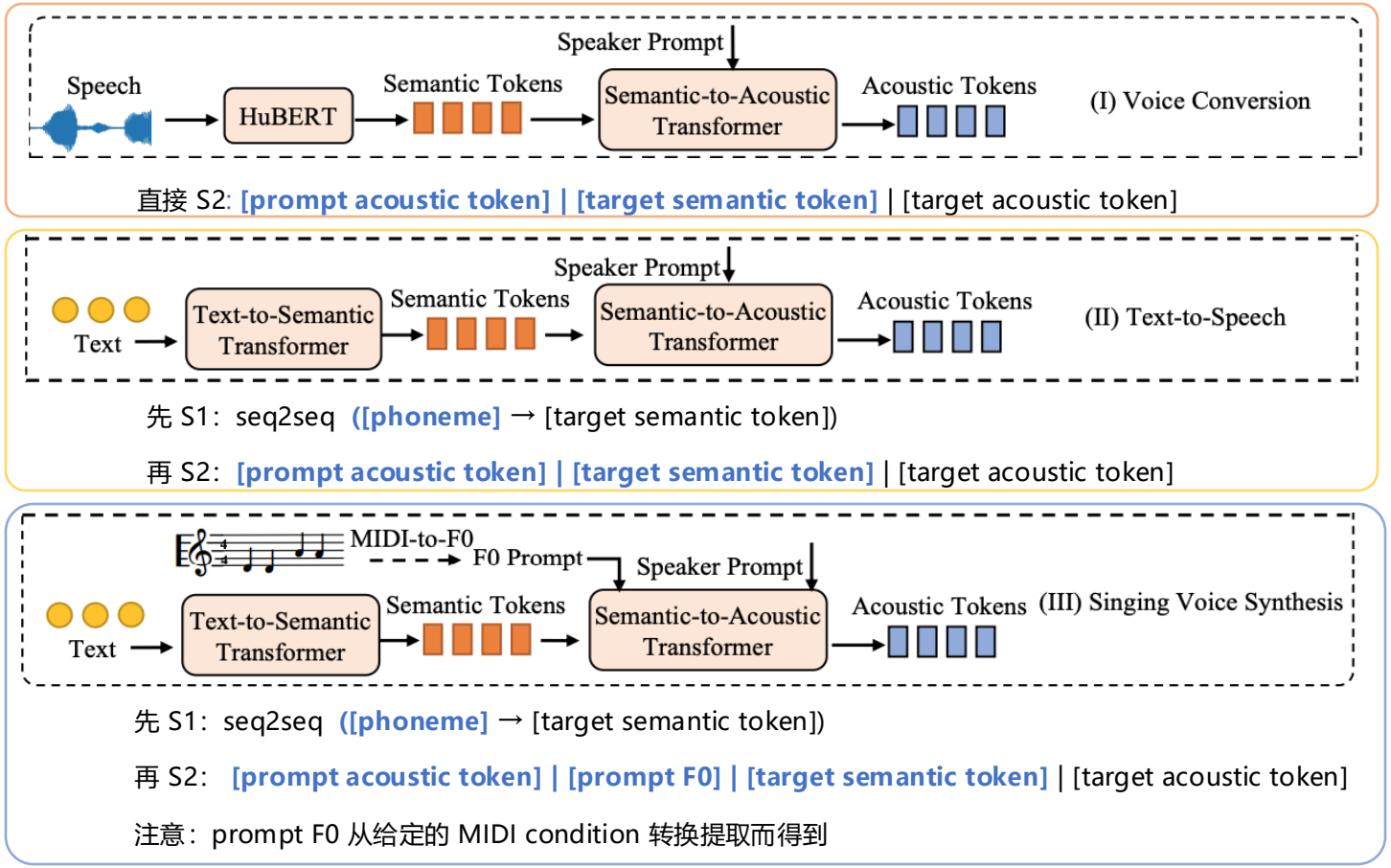




Snake1d

Purely LLM based Audio Generation

Make-A-Voice: Extension of Spear-TTS



Purely LLM based Audio Generation

Make-A-Voice: Extension of Spear-TTS

Table 1: Dataset usage in training and inference stages.

Language	S_1	S_2/S_3	Application	Zero-Shot Testing
English	LibriTTS train	LibriLight	TTS, VC	LibriTTS test
Chinese	OpenCPOP train	OpenCPOP+OpenSinger+CSMSC	SVS	M4Singer test

Table 2: Quality and style similarity of generated samples in zero-shot text-to-speech.

Model	MOS (↑)	SMOS (↑)	CER (↓)	Cos (↑)
GT	4.23±0.09	/	0.030	/
Meta-StyleSpeech	3.84±0.08	3.76±0.09	**0.065**	0.73
Zero-Shot Text-to-Speech				
GenerSpeech	3.99±0.08	3.77±0.08	0.059	0.75
YourTTS	3.89±0.08	3.72±0.06	0.143	0.72
Make-A-Voice (TTS)	**4.04±0.07**	**3.81±0.08**	0.068	**0.77**
Small-Scale Subjective Test				
VALL-E	3.92±0.12	3.81±0.07	/	/
SPEAR-TTS	3.98±0.06	**3.84±0.06**	/	/
Make-A-Voice (TTS)	**4.05±0.10**	3.83±0.08	/	/

Purely LLM based Audio Generation

Make-A-Voice: Extension of Spear-TTS

Table 3: Quality and style similarity of generated samples in zero-shot voice conversion.

Model	MOS (↑)	SMOS (↑)	Cos (↑)
GT	4.26±0.06	/	/
NANSY	3.89±0.08	3.73±0.10	0.68
PPG-VC	3.97±0.06	**3.80±0.07**	0.78
Zero-Shot Voice Conversion			
Make-A-Voice (VC)	**4.07±0.06**	3.77±0.07	**0.80**

Hubert Semantic Token
可能存在音色泄露?

Table 5: Ablation studies.

Model	CER (↓)	STOI (↑)	MCD (↓)
HuBERT-10	0.54	/	/
HuBERT-11	0.44	/	/
HuBERT-12	**0.39**	/	/
S_3: SoundStream	/	0.92	1.90
S_3: Unit Vocoder	/	**0.93**	**1.56**

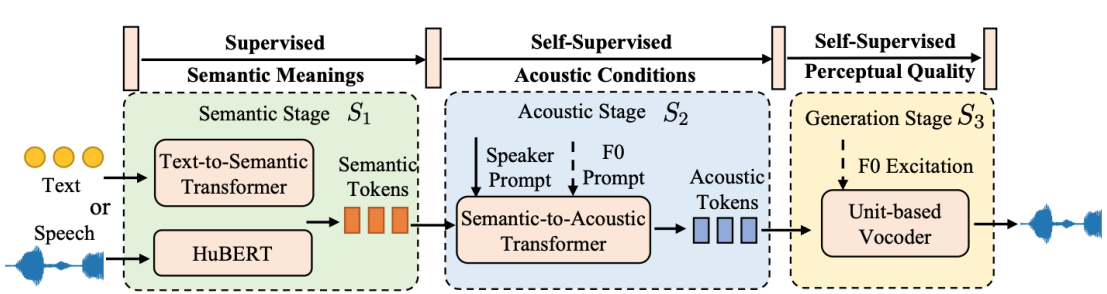
Table 4: Quality and style similarity of generated samples in zero-shot singing voice synthesis.

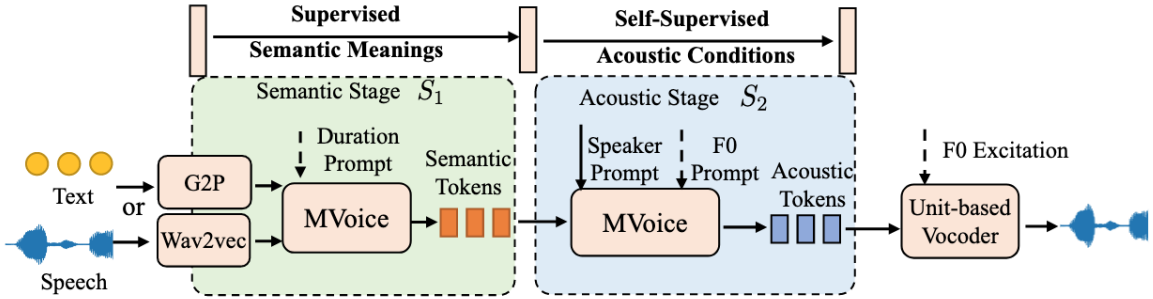
Model	MOS (↑)	SMOS (↑)	Cos (↑)	FFE (↓)
GT	4.08±0.08	/	/	/
FFT-Singer	3.83±0.09	3.91±0.08	0.93	0.17
Diffsinger	3.98±0.07	**3.98±0.07**	**0.94**	0.08
Zero-Shot Singing Voice Synthesis				
Make-A-Voice (SVS)	**3.99±0.08**	3.96±0.07	0.88	**0.05**

训练和推理阶段有一些重复的 speaker

Purely LLM based Audio Generation

MVoice: Enhanced Version of Make-A-Voice





改进点：更多语种 + 更多任务

- Semantic Token: XLSR-53（53 种语言训练的 wav2vec 2.0），k-means 聚类中心数为 1000, 帧率 50 Hz
- S1：从 seq2seq 改为自回归 LM 建模
  - **text to semantic**（spear-tts S1）
  - **text to semantic with duration prompt**
    - Sing Voice Synthesis 需要 phoneme 级别的对齐信息，扩充为帧级别的 phoneme 序列
  - **speech to semantic** 将语音的 wav2vec continuous vector 转换为 semantic token
- S2：模型结构从普通 Transformer 修改为 Multi-Scale Transformer
  - **semantic to acoustic** 将 semantic token 序列转换为 acoustic token 序列 (speaker/ F0 prompt 与 Make-A-Voice 相同)

$$p\left(\mathbf{s} \mid \mathbf{c}_{\mathbf{s}}; \theta_{AR}\right) = \prod_{t=0}^{T} p\left(\mathbf{s}_t \mid \mathbf{s}_{<t}, \mathbf{c}_{\mathbf{s}}; \theta_{AR}\right)$$

Purely LLM based Audio Generation

MVoice: Enhanced Version of Make-A-Voice

Applications	S_1	S_2
TTS	Text-to-semantic	Semantic-to-acoustic
SVS	Text-to-semantic with duration	Semantic-to-acoustic with $\mathbf{F_0}$
VC	K-means	Semantic-to-acoustic
SVC	K-means	Semantic-to-acoustic with $\mathbf{F_0}$
S2ST	Speech-to-semantic	Semantic-to-acoustic

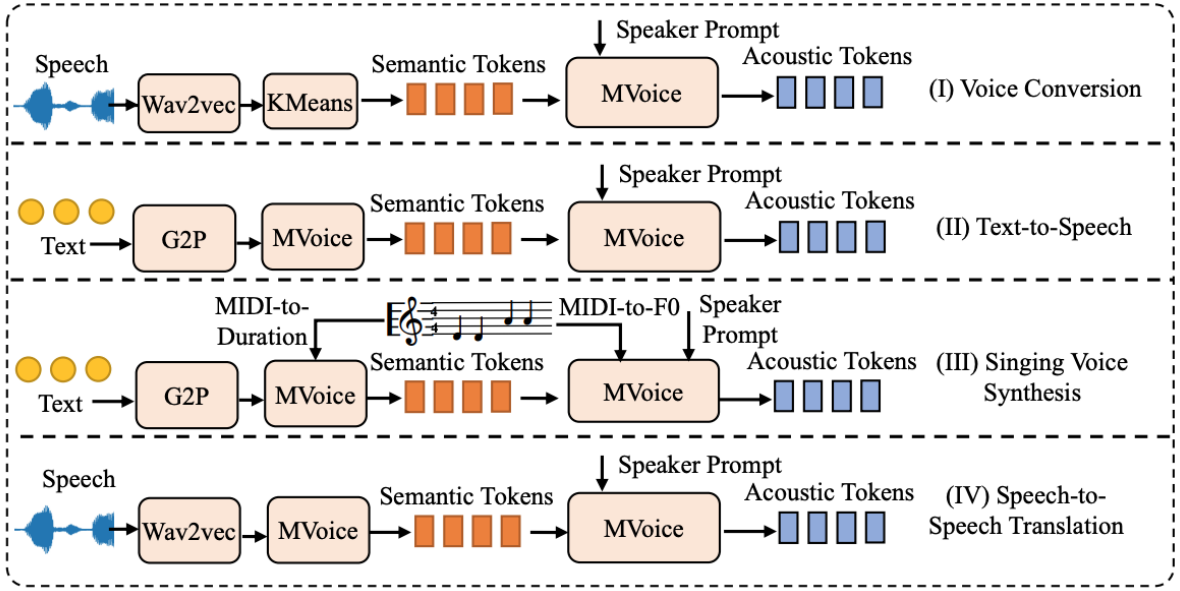


Table 3: Dataset usage in training and inference stages.

Tasks	Language	Dataset	Testing set
TTS/VC	Ja, De, Fr, En, Es, Zh	Librilight, Gigaspeech, WenetSpeech, CSS, AISHELL	LibriTTS/VCTK
SVS	En, Zh	OpenSinger, M4Singer, CSD, Kiritan	Opencpop
S2ST	Fr, En, Es, De	SpeechMatrix	SpeechMatrix test

Model	TTS-WER	TTS-SIM	VC-SIM
Base	9.8	0.84	0.75
Medium	8.3	0.86	**0.76**
Large	**6.1**	**0.87**	**0.76**

模型参数：160M, 520M, 1.2B

Table 7: Zero-shot VC and SVC.

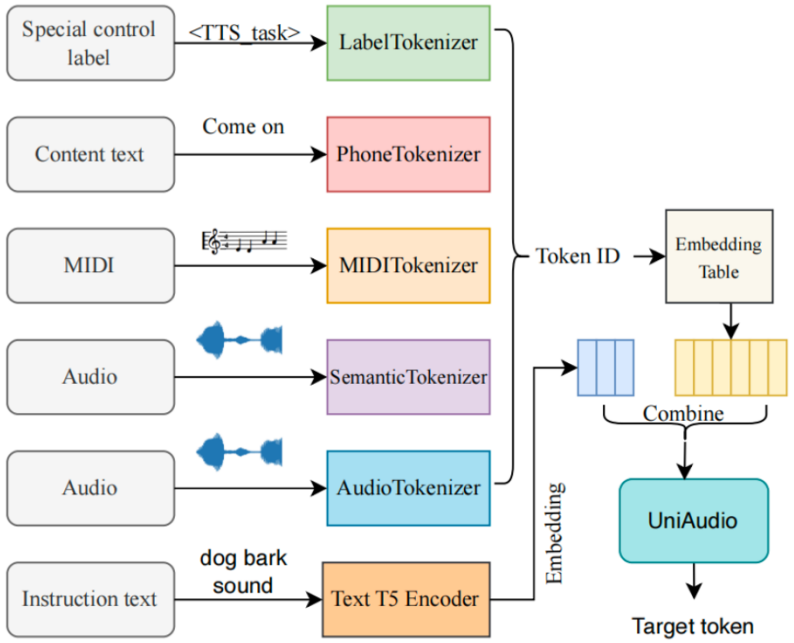
Model	MOS (↑)	SMOS (↑)	SIM (↑)
Voice Conversion			
Prompt	4.26±0.06	/	/
NANSY	3.89±0.08	3.73±0.10	0.68
PPG-VC	3.97±0.06	**3.82±0.05**	0.78
MVoice (Zero-shot)	**4.02±0.08**	3.78±0.06	**0.80**
Singing Voice Conversion			
Prompt	4.21±0.05	/	/
MVoice	3.96±0.06	3.72±0.05	**0.76**

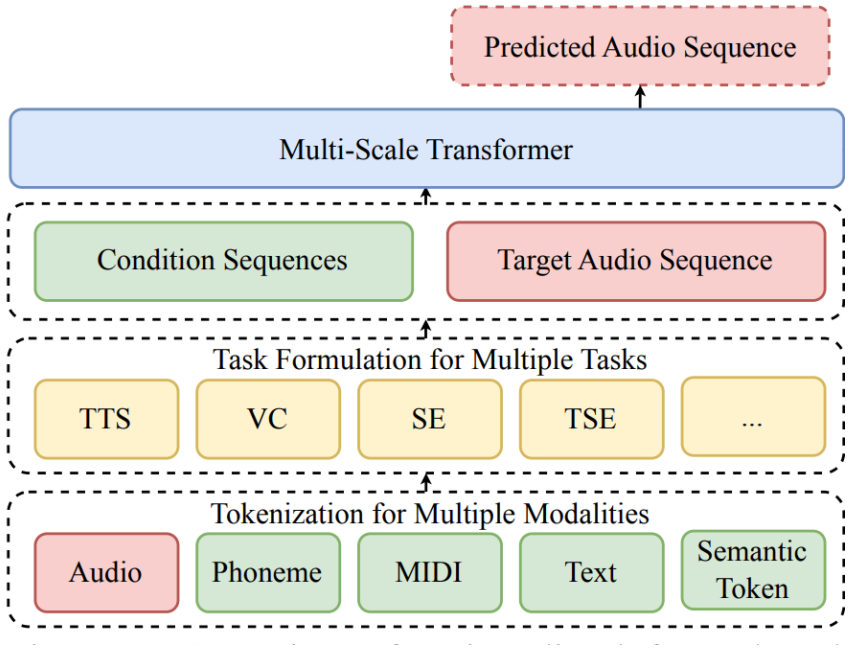
Towards LLM Based Audio Generation Foundation Model

UniAudio: Towards a Foundation Model for Audio Generation

通用的音频生成框架

- **输出模态固定为音频模态**（语音/歌声/音乐/各种声音）→ Universal Audio Codec
- LLM 支持基于各种类型 prompt 的生成任务（Task-Based Prompting）
- 将所有音频生成统一同一框架下，只用一个模型建模（Unified Task Formulation + 硬 train 一发）



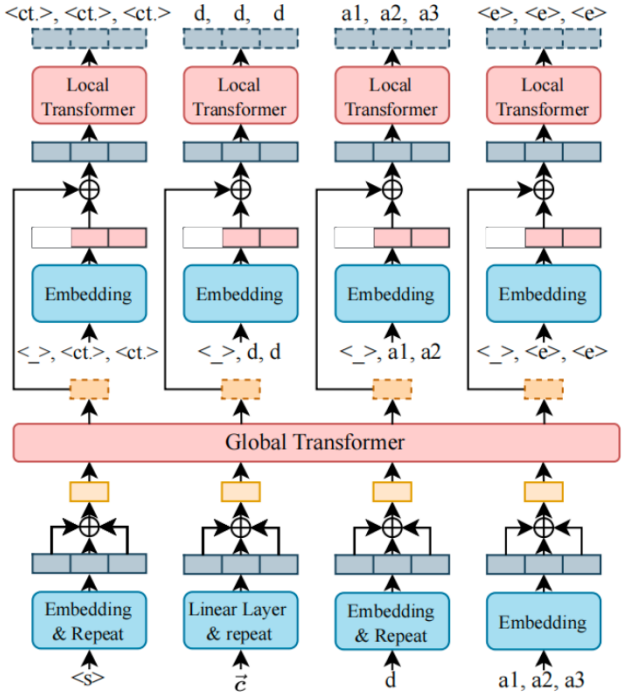


Towards LLM Based Audio Generation Foundation Model

UniAudio: Towards a Foundation Model for Audio Generation

通用的音频生成框架

- 输出模态固定为音频模态（语音/歌声/音乐/各种声音）→ Universal Audio Codec
- LLM 支持基于各种类型 prompt 的生成任务（Task-Based Prompting）
- 将所有音频生成统一同一框架下，只用一个模型建模（Unified Task Formulation + 硬 train 一发）



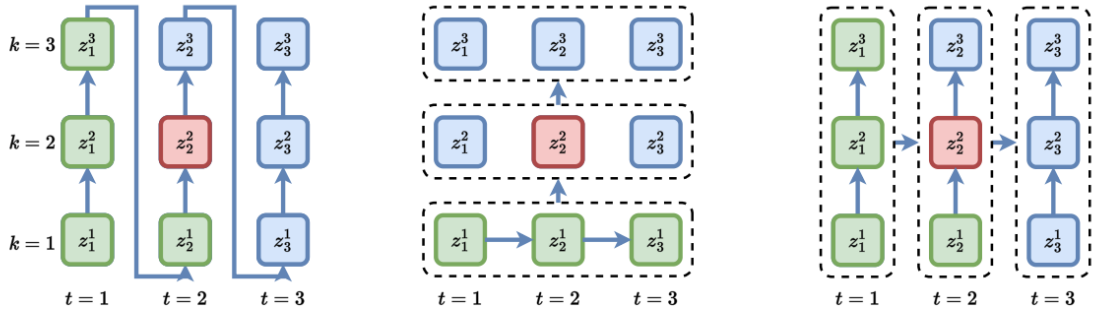


Figure 3: Order of token prediction for SPEARTTS (left), VALL-E (middle) and the proposed multi-scale Transformer (right). Assume $n_q = 3$ and $T = 3$. Current token prediction (red) is conditioned on prior tokens (in green).

Towards LLM Based Audio Generation Foundation Model

UniAudio: Towards a Foundation Model for Audio Generation

Task	Conditions	Audio Target
Text-to-Speech (TTS) (Wang et al., 2023a)	phoneme, speaker prompt	speech
Voice Conversion (VC) ♣ (Wang et al., 2023e)	semantic token, speaker prompt	speech
Speech Enhancement (SE) ♣ (Wang et al., 2023b)	noisy speech	speech
Target Speech Extraction (TSE) ♣ (Wang et al., 2018)	mixed speech, speaker prompt	speech
Singing Voice Synthesis (SVS) (Liu et al., 2022)	phoneme (with duration), speaker prompt, MIDI	singing
Text-to-Sound (Sound) (Yang et al., 2023c)	textual description	sounds
Text-to-Music (Music) (Agostinelli et al., 2023)	textual description	music
Audio Edit (A-Edit) ♣◇ (Wang et al., 2023d)	textual description, original sounds	sounds
Speech dereverberation (SD) ♣◇ (Wu et al., 2016)	reverberant speech	speech
Instruct TTS (I-TTS)◇ (Guo et al., 2023)	phoneme, textual instruction	speech
Speech Edit (S-Edit) ◇ (Tae et al., 2021)	phoneme (with duration), original speech	speech

Table 6: Data statistics

Dataset	Type	Annotation	Volume (hrs)
Training			
LibriLight (Kahn et al., 2020)	speech	-	60k
LibriTTS (Zen et al., 2019)	speech	text	1k
MLS (Pratap et al., 2020)	speech	-	20k
AudioSet (Gemmeke et al., 2017)	sound	-	5.8k
AudioCaps (Kim et al., 2019)	sound	text description	500
WavCaps (Mei et al., 2023)	sound	text description	7k
Million Song Dataset (McFee et al., 2012)	music	text description	7k
OpenCPOP (Wang et al., 2022)	singing	text, MIDI	5.2
OpenSinger (Huang et al., 2021a)	singing	text, MIDI	50
AISHELL3 (Shi et al., 2020)	speech	text	85
PromptSpeech (Guo et al., 2023)	speech	text, instruction	200
openSLR26,openSLR28 (Ko et al., 2017)	Room Impulse Response	-	100
Test			
LibriSpeech test-clean Panayotov et al. (2015)	speech	text	8
VCTK (Veaux et al., 2017)	speech	text	50
TUT2017 Task1 (Mesaros et al., 2017)	Noise	-	10
Cloth (Drossos et al., 2020)	Sound	text description	3
MusicCaps (Agostinelli et al., 2023)	Music	text description	15
M4Singer(Zhang et al., 2022)	singing	text, MIDI	1

Type				Speech (VCTK) (Veaux et al., 2017)		Sound (cloth) (Drossos et al., 2020)		Music (musiccaps) (Agostinelli et al., 2023)		Sing (m4sing) (Zhang et al., 2022)		Average -	
Model	n_q	FPS	TPS	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI
Encodec	8	75	600	2.18	0.79	2.23	0.48	1.86	0.57	1.95	0.76	2.05	0.65
Ours	3	50	150	2.96	0.85	2.42	0.49	1.99	0.57	3.13	0.85	2.62	0.69
Ours	4	50	200	3.11	0.86	2.5	0.51	2.08	0.59	3.27	0.86	2.73	0.71
Ours	8	50	400	**3.36**	**0.88**	**2.67**	**0.54**	**2.31**	**0.65**	**3.49**	**0.89**	**2.95**	**0.74**

Table 7: Dataset adoption of all tasks

Task	Training dataset	Test set	Train Volume (hrs)
Training Stage			
TTS	Librilight	LibriSpeech clean-test	60k
VC	Librilight	VCTK	60k
SE	MLS, Audioset	TUT2017 Task1, VCTK	20k
TSE	MLS	Libri2Mix test set	10k
Sound	AudioCaps, WavCaps	Cloth test set	7k
Music	MSD	MusicCaps	7k
Singing	OpenCPOP, OPenSinger, AISHEELL-3	M4Singer test set	150
Fine-Tuning Stage			
I-TTS	PromptSpeech	PromptSpeech test set	200
Speech dereverberation	LibriTTS, openSLR26, openSLR28	LibriTTS test set	100
Speech edit	LibriTTS	LibriTTS test set	100
Audio edit	AudioCaps, WavCaps	AudioCaps test set	500
Sum	-	-	166k

Towards LLM Based Audio Generation Foundation Model

问题一：音频的离散化

- 音频离散化 + LLM 建模，成为一种主流选择
  - 抛弃连续表征，效果能够达到足够好?
  - Encodec 效果、不太行

问题二：Unified Model 的必要性

- 单个模型解决所有任务 or 各模型
  - 假设：不同类型的音频之间是有一些共享的信息的
  - 单个任务数据量足够，是否还需要 UniAudio?

问题三：更好地解决长序列建模?

- 方案一：自回归+非自回归
  - VALL-E / Speech-X
- 方案二：减少建模的序列长度
  - Make-A-Voice / UniAudio （RVQ 层数为 3）
- 方案三：使用更适合长序列建模的 Transformer
  - MVoice / UniAudio

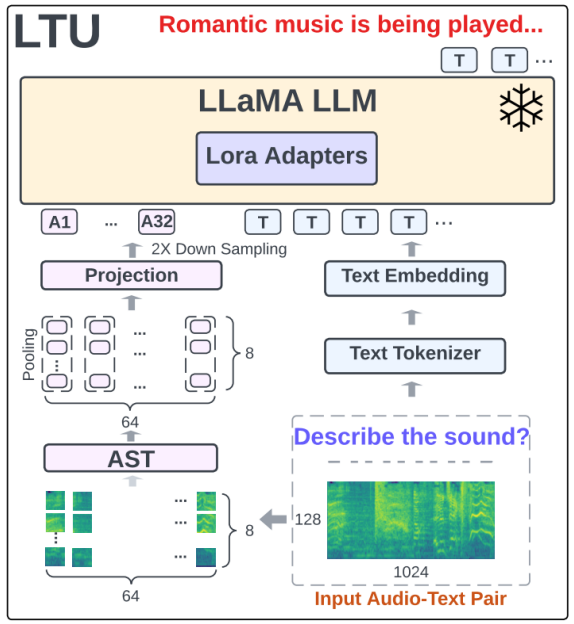
Task	Model	Objective Evaluation Metrics	Objective Evaluation Results	Subjective Evaluation Metrics	Subjective Evaluation Results
Text-to-Speech (TTS)	Shen et al. (2023)	SIM(↑) / WER(↓)	0.62 / 2.3	MOS(↑) / SMOS(↑)	**3.83±0.10** / 3.11±0.10
	UniAudio (Single)		0.64 / 2.4		3.77±0.06 / 3.46±0.10
	UniAudio		**0.71 / 2.0**		3.81±0.07 / **3.56±0.10**
Voice Conversion (VC)	Wang et al. (2023e)	SIM(↑) / WER(↓)	0.82 / 4.9	MOS(↑) / SMOS(↑)	3.41±0.08 / 3.17±0.09
	UniAudio (Single)		0.84 / 5.4		3.45±0.07 / 3.44±0.07
	UniAudio		**0.87 / 4.8**		**3.54±0.07 / 3.56±0.07**
Speech Enhancement (SE)	Richter et al. (2023)	PESQ(↑) / VISQOL(↑) / DNSMOS(↑)	**3.21 / 2.72** / 3.29	MOS(↑)	3.56±0.08
	UniAudio (Single)		2.35 / 2.30 / 3.45		3.65±0.08
	UniAudio		2.63 / 2.44 / **3.66**		**3.68±0.07**
Target Speaker Extraction (TSE)	Wang et al. (2018)	PESQ(↑) / VISQOL(↑) / DNSMOS(↑)	**2.41 / 2.36** / 3.35	MOS(↑)	3.43±0.09
	UniAudio (Single)		1.97 / 1.61 / 3.93		3.58±0.08
	UniAudio		1.88 / 1.68 / **3.96**		**3.72±0.06**
Singing Voice Synthesis (SVS)	Liu et al. (2022)	-	-	MOS(↑) / SMOS(↑)	3.94±0.02 / **4.05±0.06**
	UniAudio (Single)				**4.14±0.07** / 4.02±0.02
	UniAudio				4.08±0.04 / 4.04±0.05
Text-to-Sound (Sound)	Liu et al. (2023a)	FAD (↓) / KL (↓)	4.93 / 2.6	OVL (↑) / REL (↑)	61.0±1.9 / 65.7±1.8
	UniAudio (Single)		3.84 / 2.7		60.0±2.1 / 61.2±1.8
	UniAudio		**3.12 / 2.6**		**61.9±1.9 / 66.1±1.5**
Text-to-Music (Music)	Copet et al. (2023)	FAD (↓) / KL (↓)	4.52 / **1.4**	OVL (↑) / REL (↑)	**73.3±1.5 / 71.3±1.7**
	UniAudio (Single)		5.24 / 1.8		64.4±2.1 / 66.2±2.4
	UniAudio		**3.65** / 1.9		67.9±1.7 / 70.0±1.5

Task	Model	Evaluation Metrics	Evaluation Results
Audio Edit (A-Edit)	AUDIT (Wang et al., 2023d)	FD (↓) / KL (↓)	20.78 / 0.86
	UniAudio (scratch)		19.82 / 0.92
	UniAudio (tuning)		**17.78 / 0.77**
Speech Dereverb. (SD)	SGMSE+ Richter et al. (2023)	PESQ(↑) / DNSMOS(↑)	**2.87** / 3.42
	UniAudio (scratch)		1.23 / 3.18
	UniAudio (tuning)		2.13 / **3.51**
Instructed TTS (I-TTS)	GroundTruth	MOS(↑) / SMOS(↑)	**3.77±0.07 / 3.85±0.08**
	UniAudio (scratch)		3.62±0.07 / 3.67±0.08
	UniAudio (tuning)		3.61±0.09 / 3.71±0.09
Speech Edit (S-Edit)	TTS system regeneration	MCD(↓) / MOS(↑)	6.98 / 3.69±0.08
	UniAudio (scratch)		5.26 / 3.73±0.07
	UniAudio (tuning)		**5.12 / 3.82±0.06**

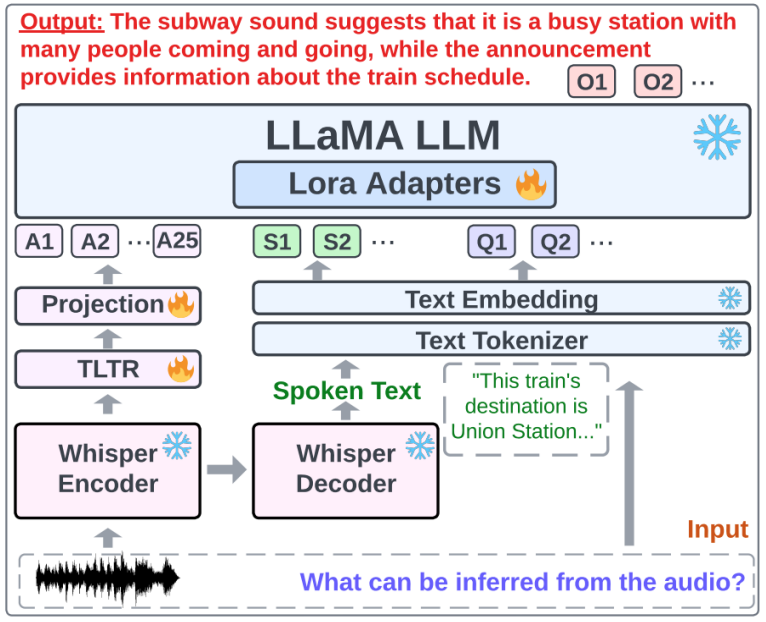
Foundation Model for Audio Understanding and Generation

Foundation Model for Audio Understanding and Generation

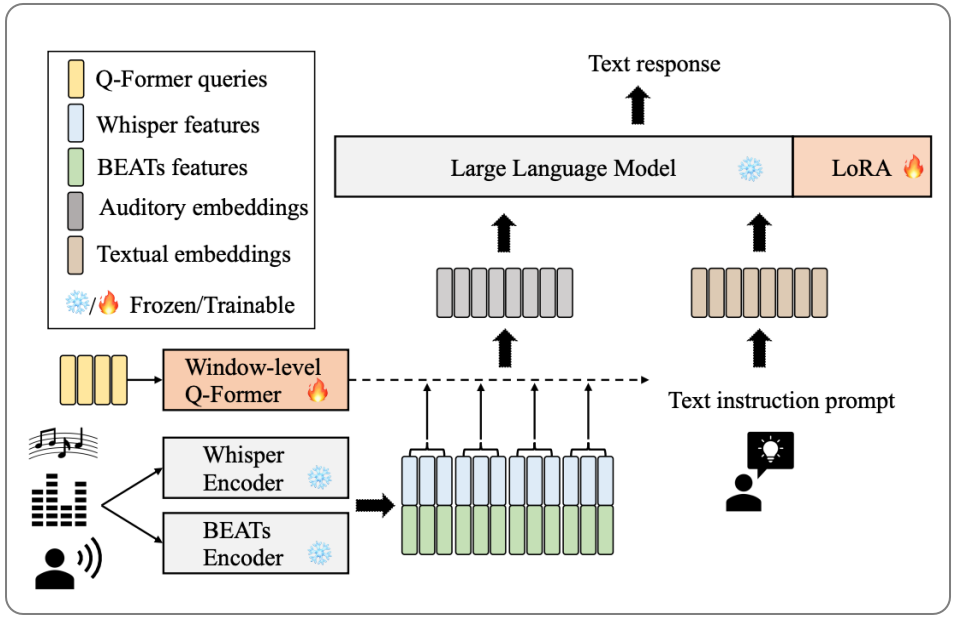
Background: Foundation Model for Speech and Audio Understanding



LTU



LTU-AS



SALMONN

语音/音频理解的 Foundation Model

- 输出模态为文本；输入模态为语音/音频+Text Instruction
- 将语音/音频模态通过 Encoder (Whisper / CLAP / BEATS)，映射到 LLaMA 的 embedding 维度
- Finetune 方式：LoRA
- 目标：使得 LLaMA （GPTx）具有理解 prompt 中语音/音频内容的能力

Foundation Model for Audio Understanding and Generation

Audio Understanding

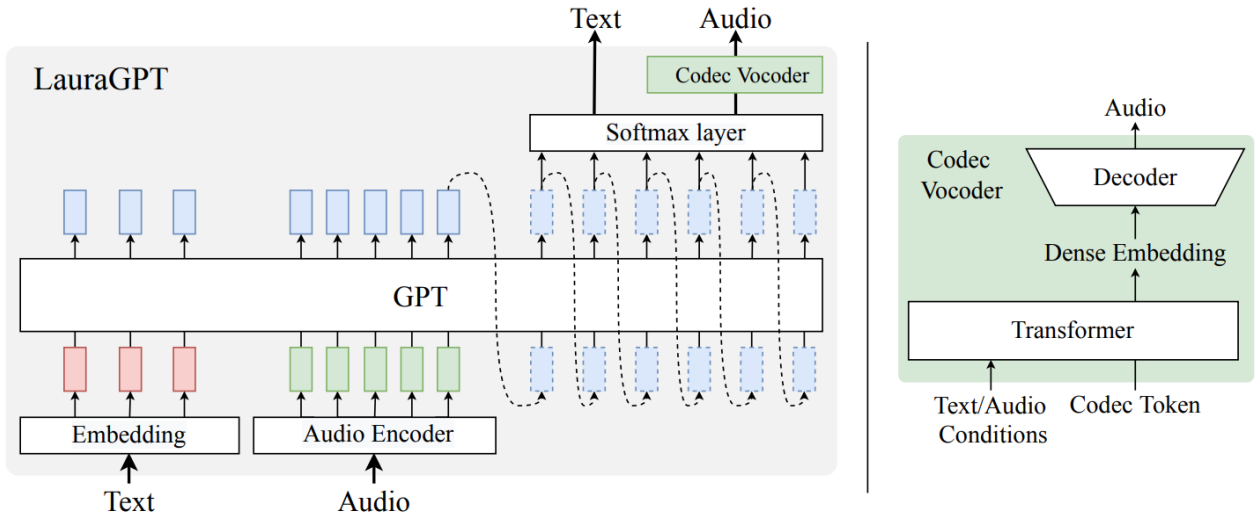
- 输入模态：文本 / 音频 / ...
- 输出模态：文本

Audio Generation

- 输入模态：文本 / 音频 / ...
- 输出模态：音频

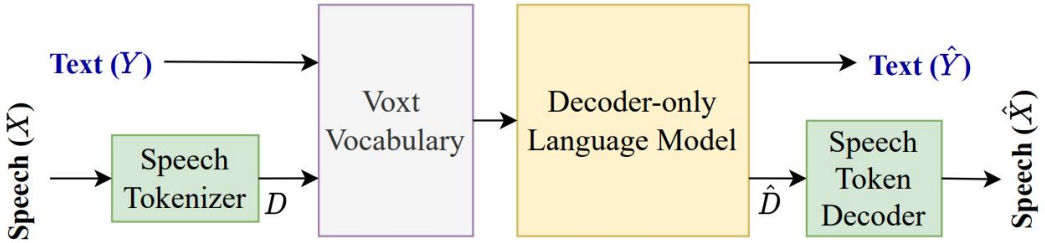
共同点

- 输入端 Whisper / CLAP / BEATS / ImageBind 都可以直接使用
- 输出端通常是离散化的 token（分类的交叉熵损失函数）
- 音频基于离散化的 token 可以直接恢复波形

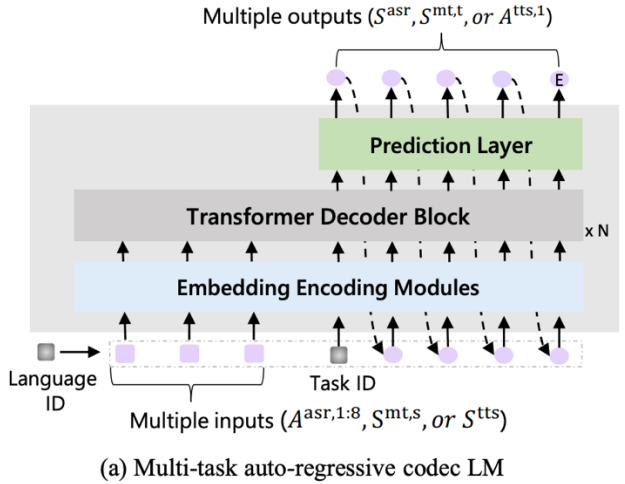


Foundation Model for Audio Understanding and Generation

VoxtLM

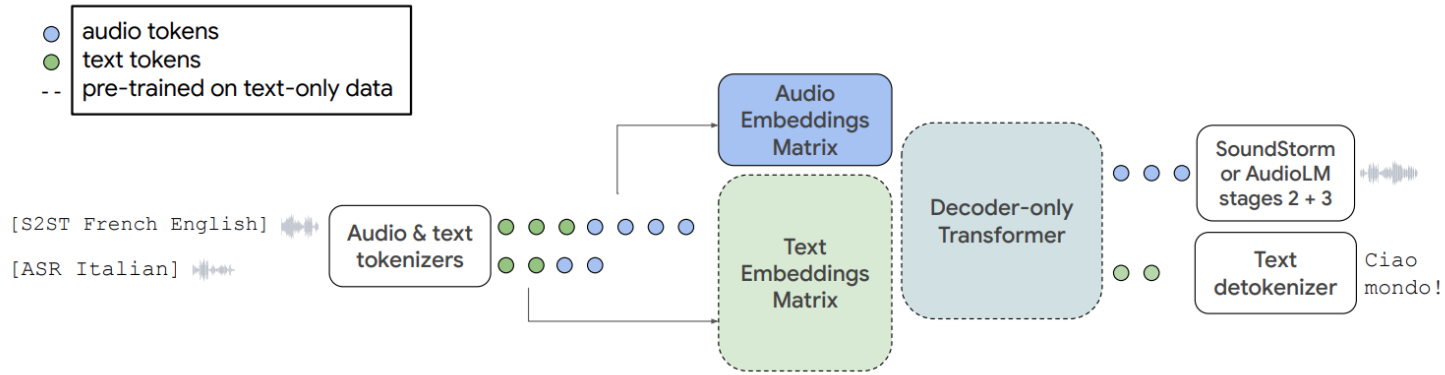


SpeechGPT

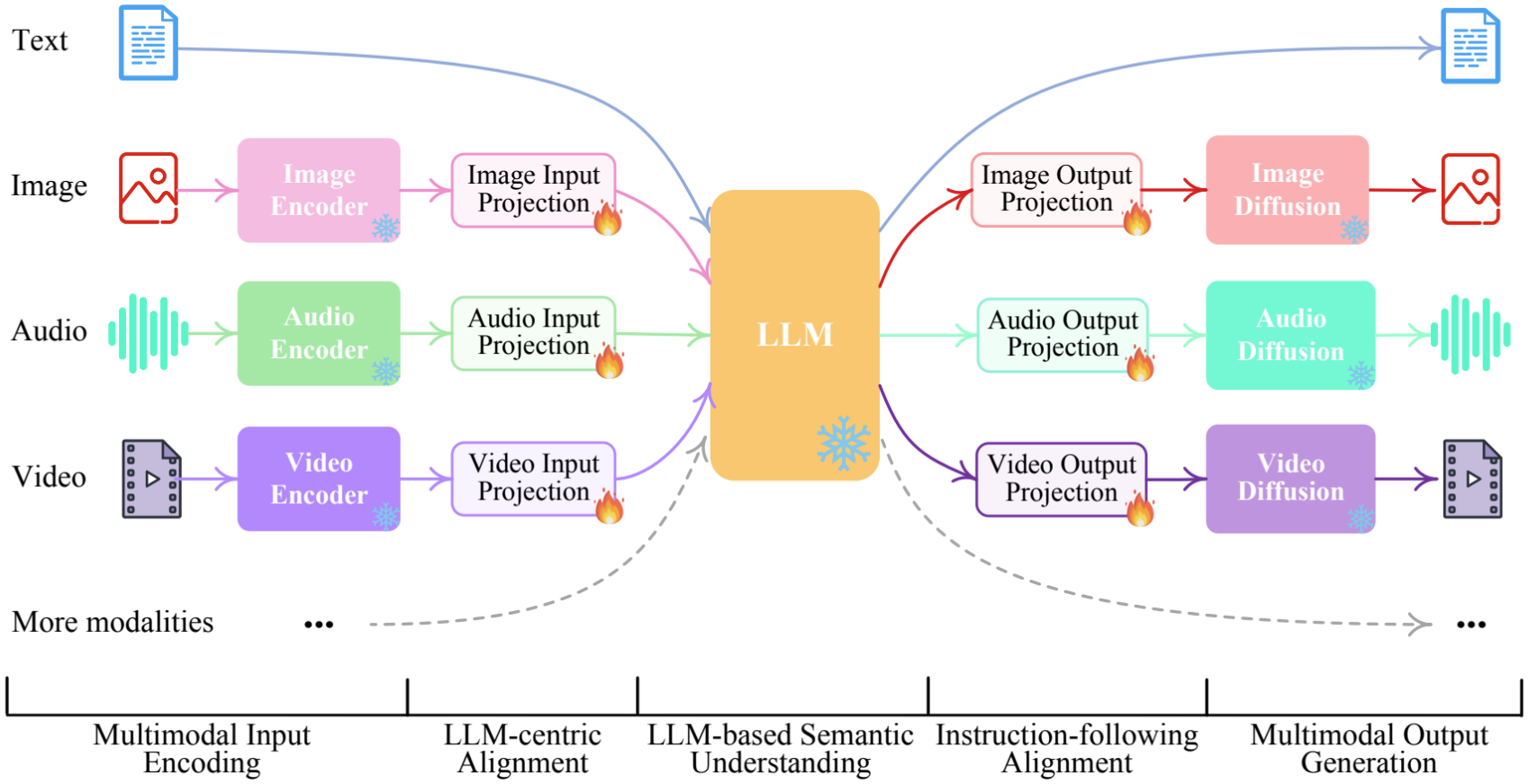


(a) Multi-task auto-regressive codec LM

AudioPaLM



Final Fantasy: Any-to-Any Generation



References

名称	论文题目	链接	训练数据量（小时）	是否开源
Speech-X	SpeechX: Neural Codec Language Model as a Versatile Speech Transformer	https://arxiv.org/pdf/2308.06873.pdf	60k+	否
Make-A-Voice	Make-A-Voice: Unified Voice Synthesis With Discrete Representation	https://arxiv.org/pdf/2305.19269.pdf	60k+	否
MVoice	Multilingual Unified Voice Generation With Discrete Representation at Scale	https://openreview.net/forum?id=eGdhD93hZr	≈200k	否
UniAudio	UniAudio: An Audio Foundation Model Toward Universal Audio Generation	https://arxiv.org/pdf/2310.00704.pdf	165k	https://github.com/yangdongchao/UniAudio

Thanks!