An Overview of Voice Cloning

Transfer Learning, Representation Learning and Meta-Learning

2022-09-15

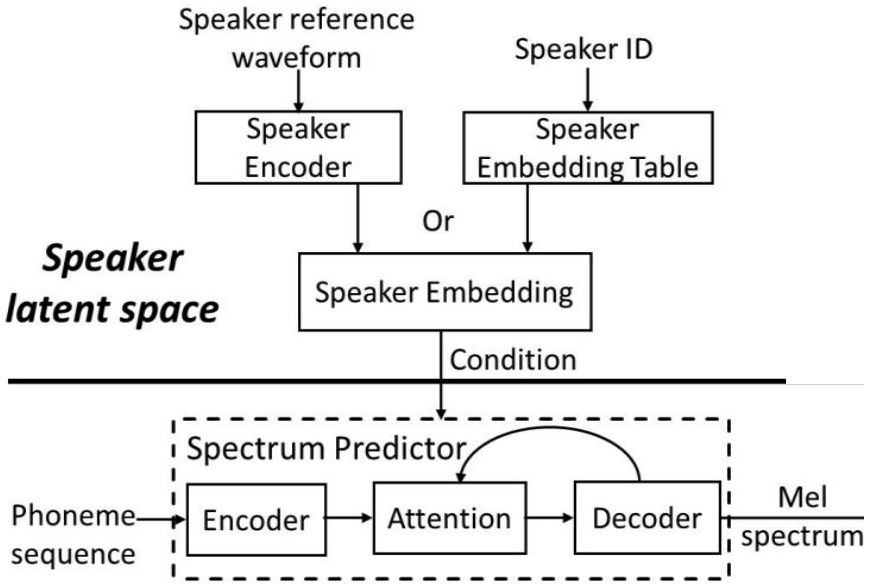
Outline

- **Background**
- **Transfer Learning (Brief Recap)**
  - Regular Adaptation
  - Speaker-Conditioned Adaptation
- **Representation Learning (Networks)**
  - Speaker Encoding
  - Style Encoding
- **Meta-Learning (Few-shot Learning)**
  - Model-Agnostic Meta-Learning (MAML)
  - Meta-TTS
- **Summary**

Background

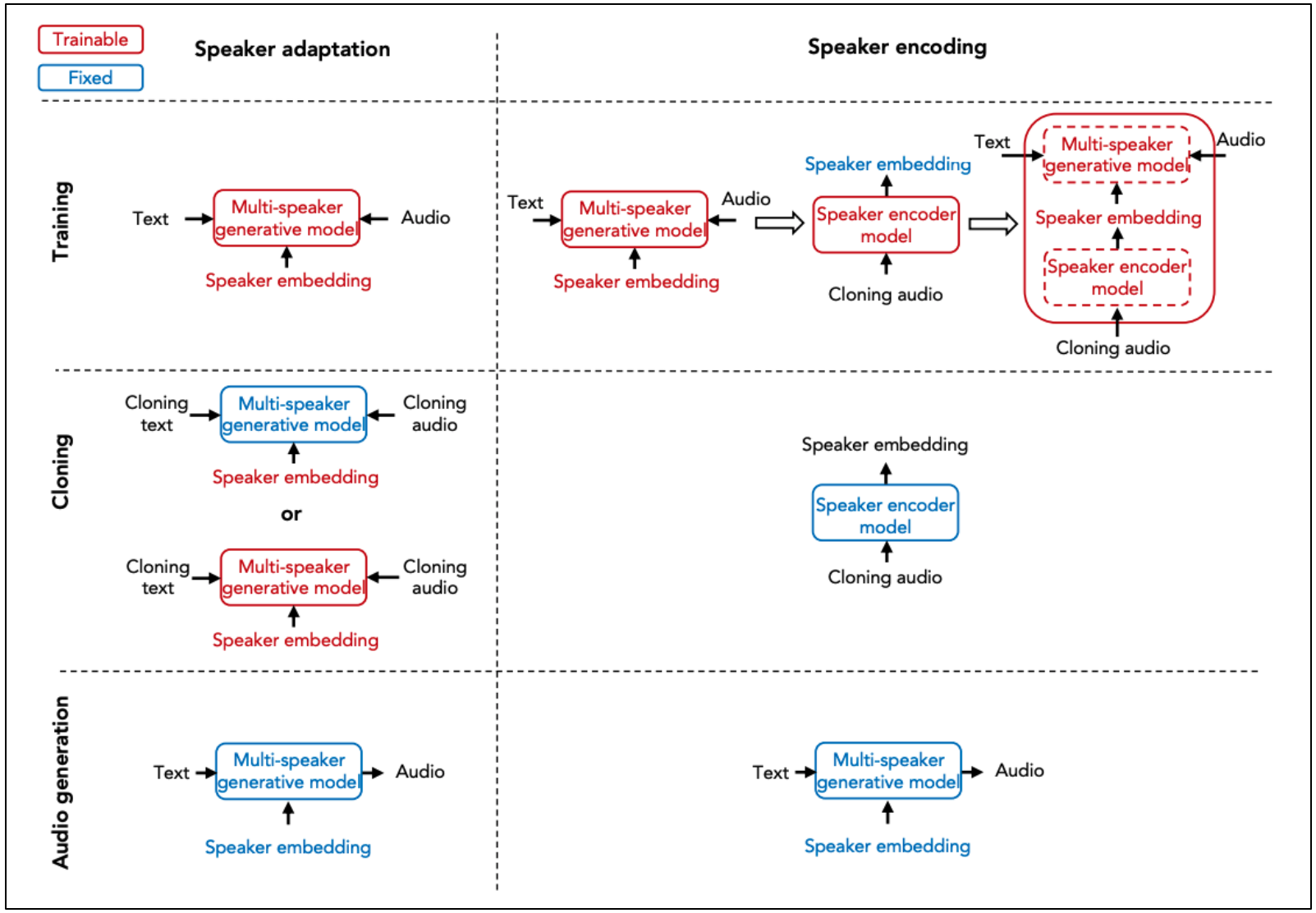
Voice Cloning (语音克隆)

- Few-Shot TTS (少样本语音合成) / Personalized TTS (个性化语音合成)
- 问题背景：使用较少样本合成目标说话人的声音
- **目标说话人的语音合成方法**
  - 样本数量很多时 → 单说话人 TTS
  - 样本数量较多时 → 多说话人 TTS
  - 样本数据很少时 → 语音克隆
- **Voice Cloning 模型的评价**
  - 语音效果 MOS / SMOS
    - 语音质量/自然度
    - 说话人音色相似度 Timbre
    - 说话人风格 Style / 韵律 Prosody 相似度
  - 成本/代价：
    - 资源消耗（单说话人的模型参数量）
    - 时间消耗（获得模型的快慢）



- **引入模型的两种方式**
  - Speaker embedding table: global embedding
  - Speaker encoder: utterance-level embedding
- **推理时的两种输入**
  - Speaker ID + 音素序列
  - Speaker 参考音频 + 音素序列

两大类思想： Transfer Learning Representation Learning



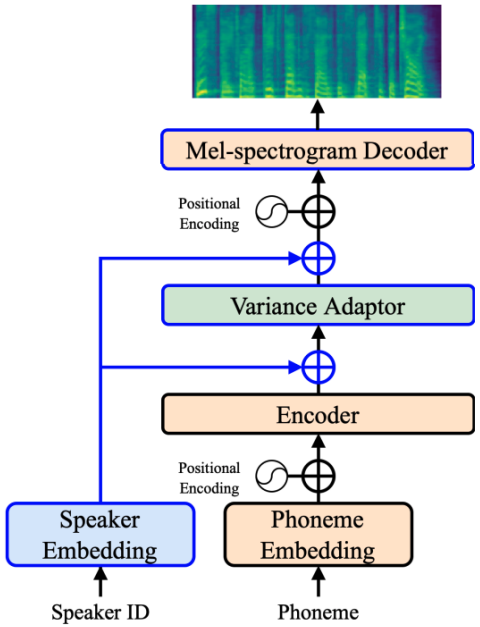
Arik, Sercan, et al. "**Neural voice cloning with a few samples**." *Advances in neural information processing systems* 31 (2018).

Transfer Learning

基础方法：Speaker Adaptation

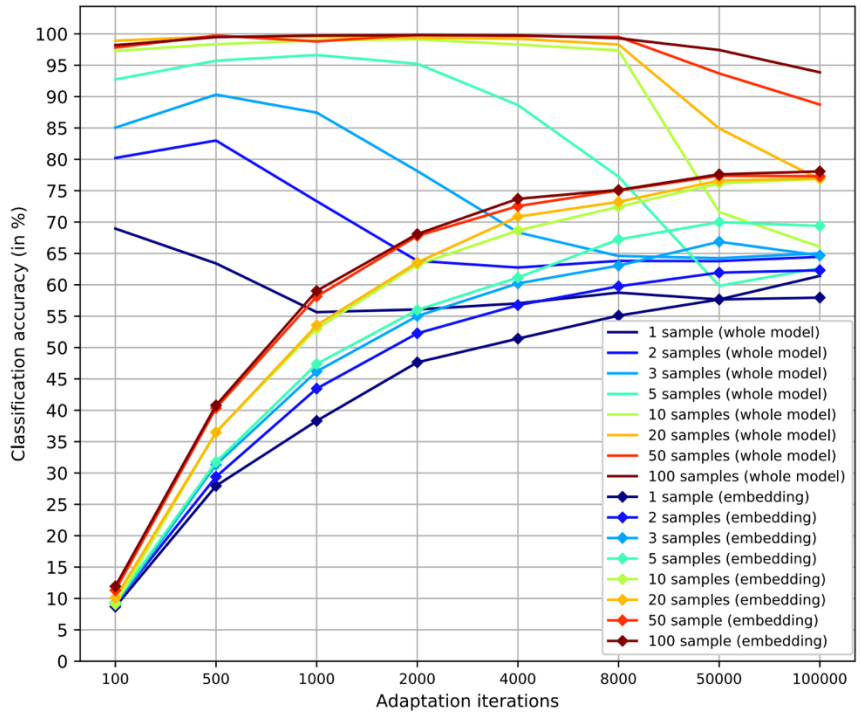
影响因素的实验性分析

- 样本数量：1, 2, 3, 5, 10, 20, 50, 100
- 微调参数量：spk embedding / 整个模型
- 模型迭代次数 (iteration)



(a) Multi-speaker FastSpeech 2.

纵坐标：使用测试集的 (新增) 说话人音频训练说话人分类器后，语音克隆合成的语音的分类准确率。

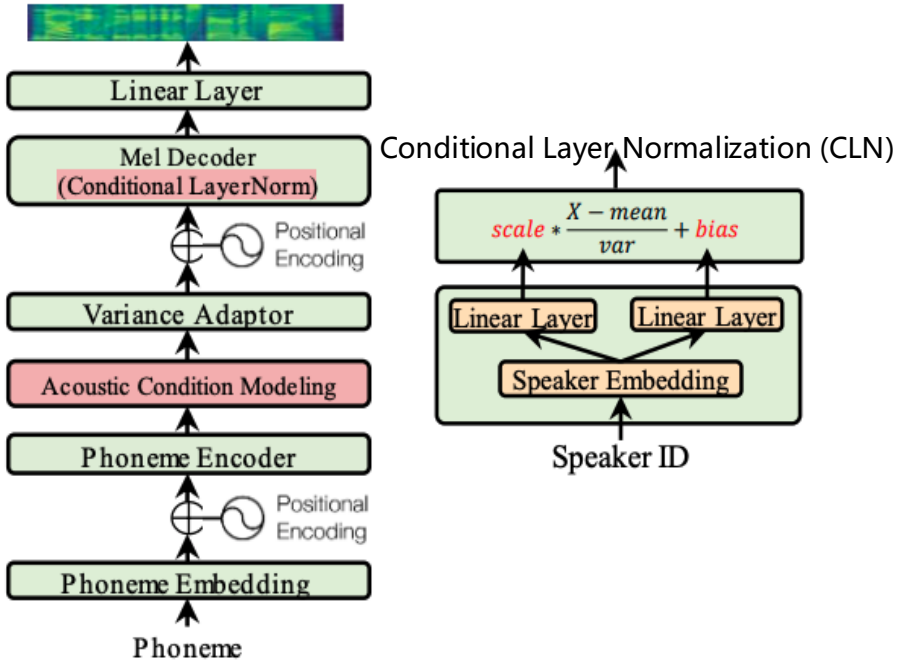


Arik, Sercan, et al. "**Neural voice cloning with a few samples**." *Advances in neural information processing systems* 31 (2018).

Transfer Learning

改进版方法：Speaker Conditioned Adaptation

- Speaker adaptation：微调参数量大，个性化模型参数多，速度慢
- Speaker conditioned adaptation ： 尽可能降低微调参数量，同时合成效果尽量不受影响
- CLN 的思想：增加以说话人信息作为输入 (condition) 的模块，只微调与说话人有关的参数



Metric	Setting	# Params/Speaker	LJSpeech	VCTK	LibriTTS
MOS	*GT*	/	3.98 ± 0.12	3.87 ± 0.11	3.72 ± 0.12
	GT mel + Vocoder	/	3.75 ± 0.10	3.74 ± 0.11	3.65 ± 0.12
	Baseline (spk emb)	256 (256)	2.37 ± 0.14	2.36 ± 0.10	3.02 ± 0.13
	Baseline (decoder)	14.1M (14.1M)	3.44 ± 0.13	3.35 ± 0.12	3.51 ± 0.11
	AdaSpeech	1.2M (4.9K)	3.45 ± 0.11	3.39 ± 0.10	3.55 ± 0.12
SMOS	*GT*	/	4.36 ± 0.11	4.44 ± 0.10	4.31 ± 0.07
	GT mel + Vocoder	/	4.29 ± 0.11	4.36 ± 0.11	4.31 ± 0.07
	Baseline (spk emb)	256 (256)	2.79 ± 0.19	3.34 ± 0.19	4.00 ± 0.12
	Baseline (decoder)	14.1M (14.1M)	3.57 ± 0.12	3.90 ± 0.12	4.10 ± 0.10
	AdaSpeech	1.2M (4.9K)	3.59 ± 0.15	3.96 ± 0.15	4.13 ± 0.09

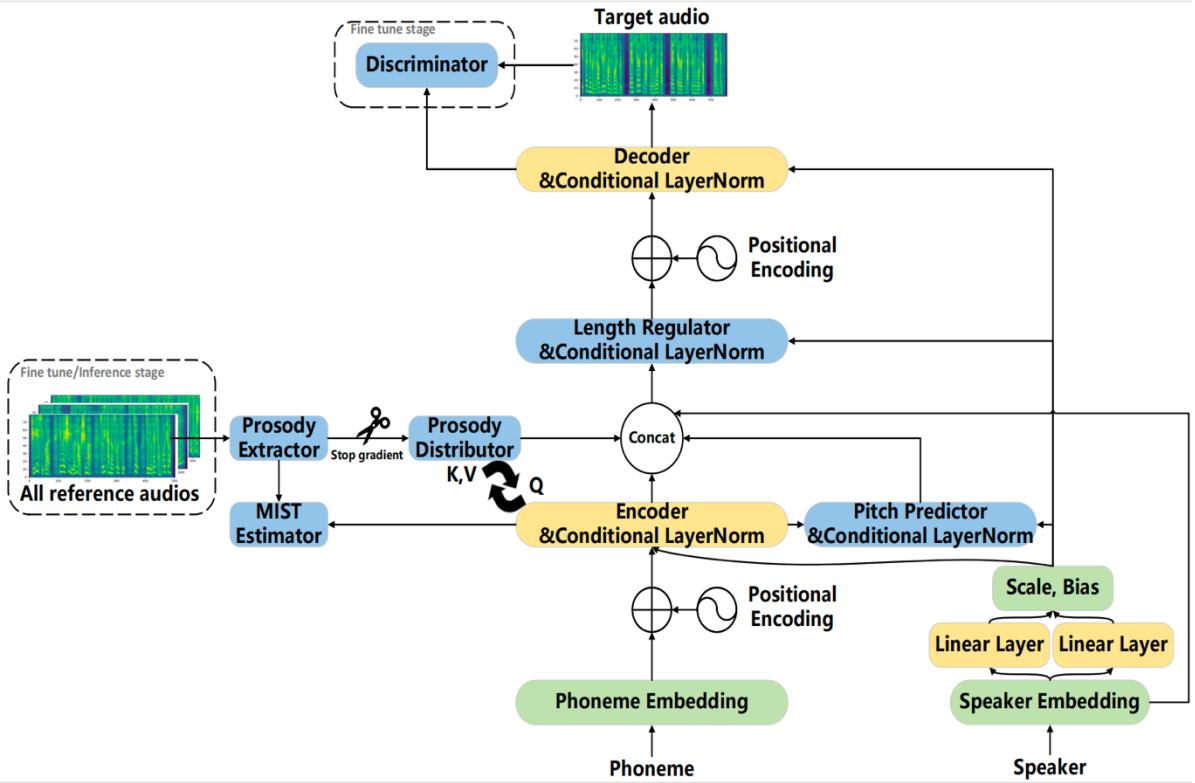
Chen, Mingjian, et al. "**AdaSpeech: Adaptive Text to Speech for Custom Voice**." *International Conference on Learning Representations*. 2020.

Transfer Learning

改进版方法：Speaker Conditioned Adaptation

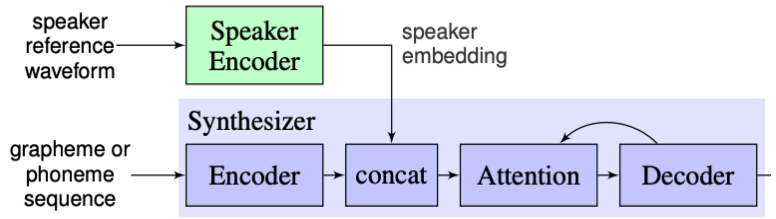
增强版：增加更多与说话人相关的模块

- Variance Adaptor 加入 Conditional Layer Norm
- 作用：引入说话人信息，增加 fine-tune 时可变参数量
- 目标：模型 adaptation 后，语速/韵律上与目标说话人更接近



Yi, Yuanhao, et al. "**Prosodyspeech: Towards Advanced Prosody Model for Neural Text-to-Speech.**" *ICASSP* 2022.

Representation Learning



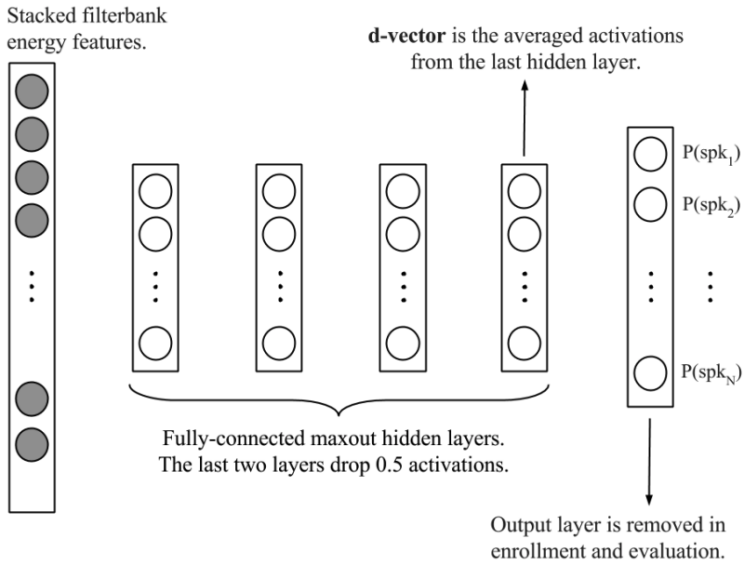
Speaker Encoding

基于 reference audio 参考音频获取 speaker embedding

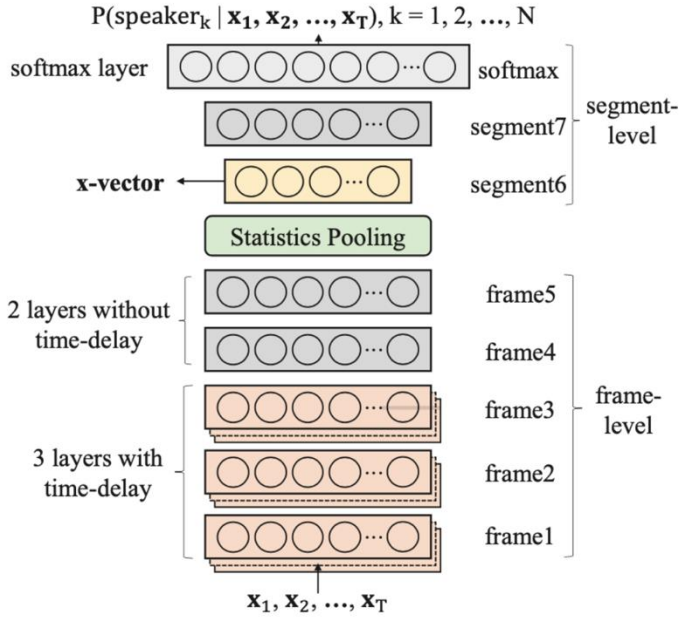
思想：以说话人分类任务为指导，将神经网络隐含层向量作为说话人的表征

Speaker Encoder 的两种使用方法

- 预训练说话人识别模型：特征提取器，给定音频，提取网络某一隐层信息作为说话人特征向量
- 多任务学习：说话人识别模型结构，增加额外的说话人分类任务，与 TTS 模型联合训练



d-vector



x-vector

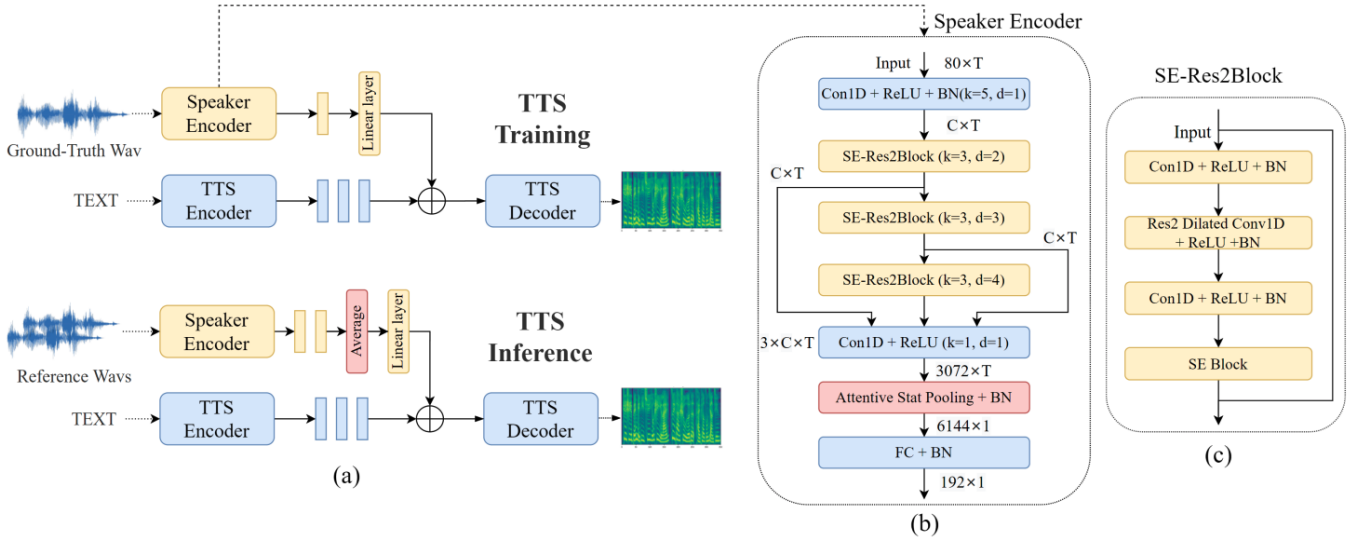
Jia, Ye, et al. "**Transfer learning from speaker verification to multispeaker text-to-speech synthesis.**" *Advances in neural information processing systems* 31 (2018).
Cooper, Erica, et al. "**Zero-shot multi-speaker text-to-speech with state-of-the-art neural speaker embeddings.**" *IEEE ICASSP*. IEEE, 2020.

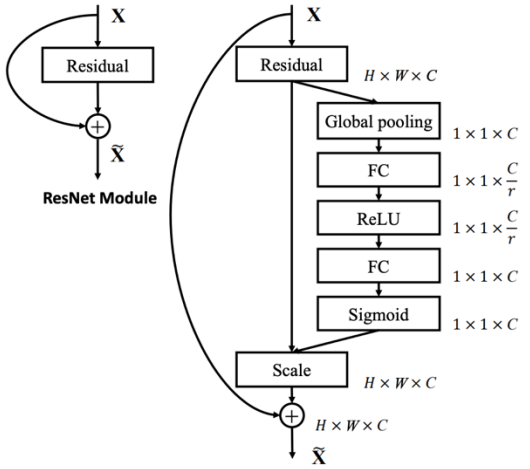
Representation Learning

Speaker Encoding

基于 reference audio 参考音频获取 speaker embedding

思想：以说话人分类任务为指导，将神经网络隐含层向量作为说话人的表征





Squeeze and excitation：在 channel 维度增加 attention，强化重要通道的特征

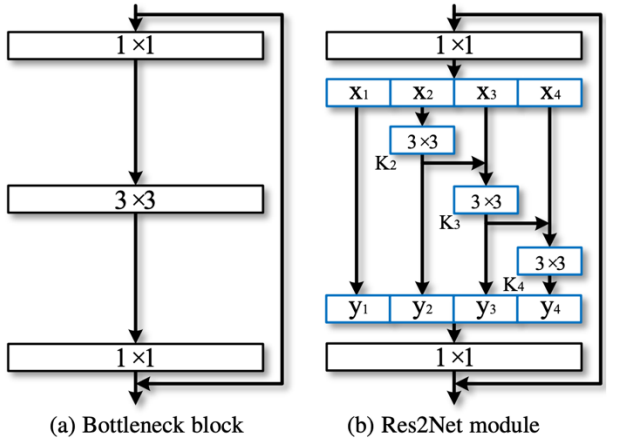


Fig. 2. Comparison between the bottleneck block and the proposed Res2Net module (the scale dimension $s = 4$).

Res2Net模块的输出包含不同感受野大小的组合，该结构有利于提取全局和本地信息。

Xue, Jinlong, et al. "**ECAPA-TDNN for Multi-speaker Text-to-speech Synthesis**." *arXiv preprint arXiv:2203.10473* (2022).

Representation Learning

结论一：Speaker Encoding 与 Speaker Identification 的相关性

- Speaker Identification : 相同后端分类器条件下，不同网络结构的效果
  - ecapa > x-vector > d-vector > i-vector
- 用于多说话人 TTS 的 Speaker Encoder， 指标上**有类似的规律**

Table 1: *EER and MinDCF performance of all systems on the standard VoxCeleb1 and VoxSRC 2019 test sets.*

Architecture	# Params	VoxCeleb1		VoxCeleb1-E		VoxCeleb1-H		VoxSRC19
		EER(%)	MinDCF	EER(%)	MinDCF	EER(%)	MinDCF	EER(%)
E-TDNN	6.8M	1.49	0.1604	1.61	0.1712	2.69	0.2419	1.81
E-TDNN (large)	20.4M	1.26	0.1399	1.37	0.1487	2.35	0.2153	1.61
ResNet18	13.8M	1.47	0.1772	1.60	0.1789	2.88	0.2672	1.97
ResNet34	23.9M	1.19	0.1592	1.33	0.1560	2.46	0.2288	1.57
ECAPA-TDNN (C=512)	6.2M	1.01	0.1274	1.24	0.1418	2.32	0.2181	1.32
ECAPA-TDNN (C=1024)	14.7M	**0.87**	**0.1066**	**1.12**	**0.1318**	**2.12**	**0.2101**	**1.22**

对比实验

- reconstruct：GT mel + HiFi-GAN
- baseline：使用 speaker embedding table
- x-vector：FastSpeech2 + 预训练 x-vector Encoder
- ecapa： FastSpeech2 + 预训练 ECAPA-TDNN Encoder

Table 2: *The results of the subjective MOS tests for naturalness and speaker similarity.*

Model	method	Seen VCTK	Unseen VCTK	Unseen LibriTTS
MOS	ground-truth	4.19	4.20	4.21
	reconstruct	4.09	4.11	4.10
	baseline	**3.70**	-	-
	x-vector	3.51	3.51	3.38
	ecapa	3.62	**3.62**	**3.47**
Similarity	reconstruct	4.65	4.69	4.66
	baseline	3.89	-	-
	x-vector	3.71	3.65	3.08
	ecapa	**3.93**	**3.66**	**3.18**

结论

- 之前 Speaker Encoding 不如 Speaker Embedding Table 的效果
- 但 ECAPA 在集内说话人上达到了更好的 SMOS
- 表征能力更强的 speaker embedding，在集外新说话人的泛化能力越好

Xue, Jinlong, et al. "**ECAPA-TDNN for Multi-speaker Text-to-speech Synthesis**." *arXiv preprint arXiv:2203.10473* (2022).

Representation Learning

结论二：Speaker Encoding 与 Speaker Adaptation 的优缺点对比

Approach	Sample count				
	1	2	3	5	10
Ground-truth (same speaker)	3.91±0.03				
Ground-truth (different speakers)	1.52±0.09				
Speaker adaptation (embedding-only)	2.66±0.09	2.64±0.09	2.71±0.09	2.78±0.10	2.95±0.09
Speaker adaptation (whole-model)	2.59±0.09	2.95±0.09	3.01±0.10	3.07±0.08	3.16±0.08
Speaker encoding (without fine-tuning)	2.48±0.10	2.73±0.10	2.70±0.11	2.81±0.10	2.85±0.10
Speaker encoding (with fine-tuning)	2.59±0.12	2.67±0.12	2.73±0.13	2.77±0.12	2.77±0.11

语音克隆方法	Speaker Adaptation	Speaker Encoding
优势	微调后新增说话人的合成效果不错	不需要额外微调模型(微调效果反而可能更不好)， 只需要新增说话人的参考音频
不足	微调整个模型时效果最好，但每个说话人一套模型，资源要求高 微调时模型迭代次数成千上万次才能达到不错的效果，耗时较长	泛化能力差，对训练集说话人的丰富程度要求高 新增说话人和训练集差异较大时，容易出现 mismatch 的问题

Arik, Sercan, et al. "**Neural voice cloning with a few samples**." *Advances in neural information processing systems* 31 (2018).

Representation Learning

结论三：Speaker Encoding 与 Speaker Adaptation 相辅相成

- Speaker Encoding 提供了音频级别丰富的说话人音色/风格层面的信息
- 在 Speaker Adaptation 的模型中增加 Speaker Encoding 提供高维信息
- 扩充说话人表征的来源 (Speaker-ID 和音频都作为输入)
- 微调时只更新和说话人更直接相关的参数，更快更省地达到更好的效果

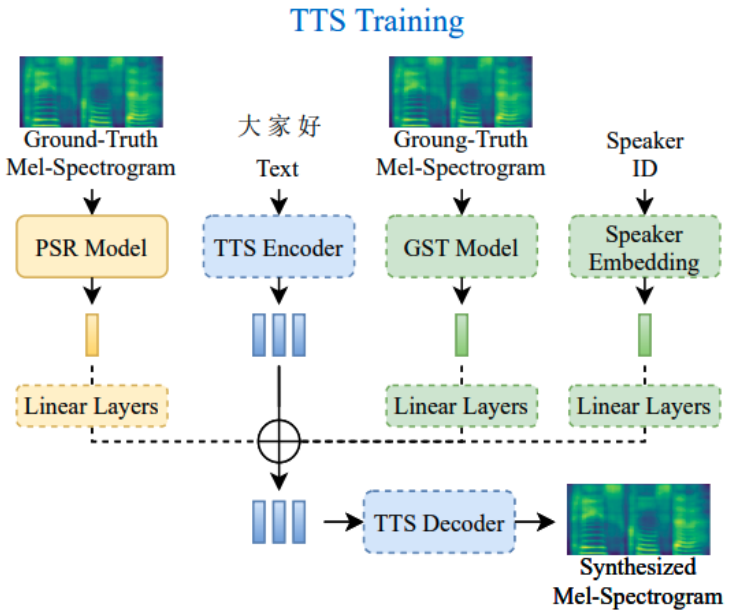


Table 1: The results of the objective speaker verification tests for TTS models with different types of speaker representations.

Model	Speaker Representation					Results	
	Pretrained			Learnable		SV Accuracy	
	d-vec	x-vec	VC	embed	GST	Track 1	Track 2
(a) Tacotron 2	✓					.772	.367
		✓				.785	.377
			✓			.942	.727
				✓		.630	.703
					✓	.102	.050
(b) FastSpeech2	✓					.977	.323
		✓				.973	.623
			✓			.980	.837
				✓		.988	.490
					✓	.778	.340
(c) FastSpeech2	✓		✓			.978	.747
			✓	✓		.983	**.937**
			✓		✓	.982	.783
			✓	✓	✓	.988	.897
	✓	✓	✓	✓	✓	.990	.887

* The colored row is the model used for the final submission to the ICASSP 2021 M2VoC challenge. Due to the time limitation, we did not submit our best model.

Chien, Chung-Ming, et al. "**Investigating on incorporating pretrained and learnable speaker representations for multi-speaker multi-style text-to-speech.**" IEEE *ICASSP*, 2021.

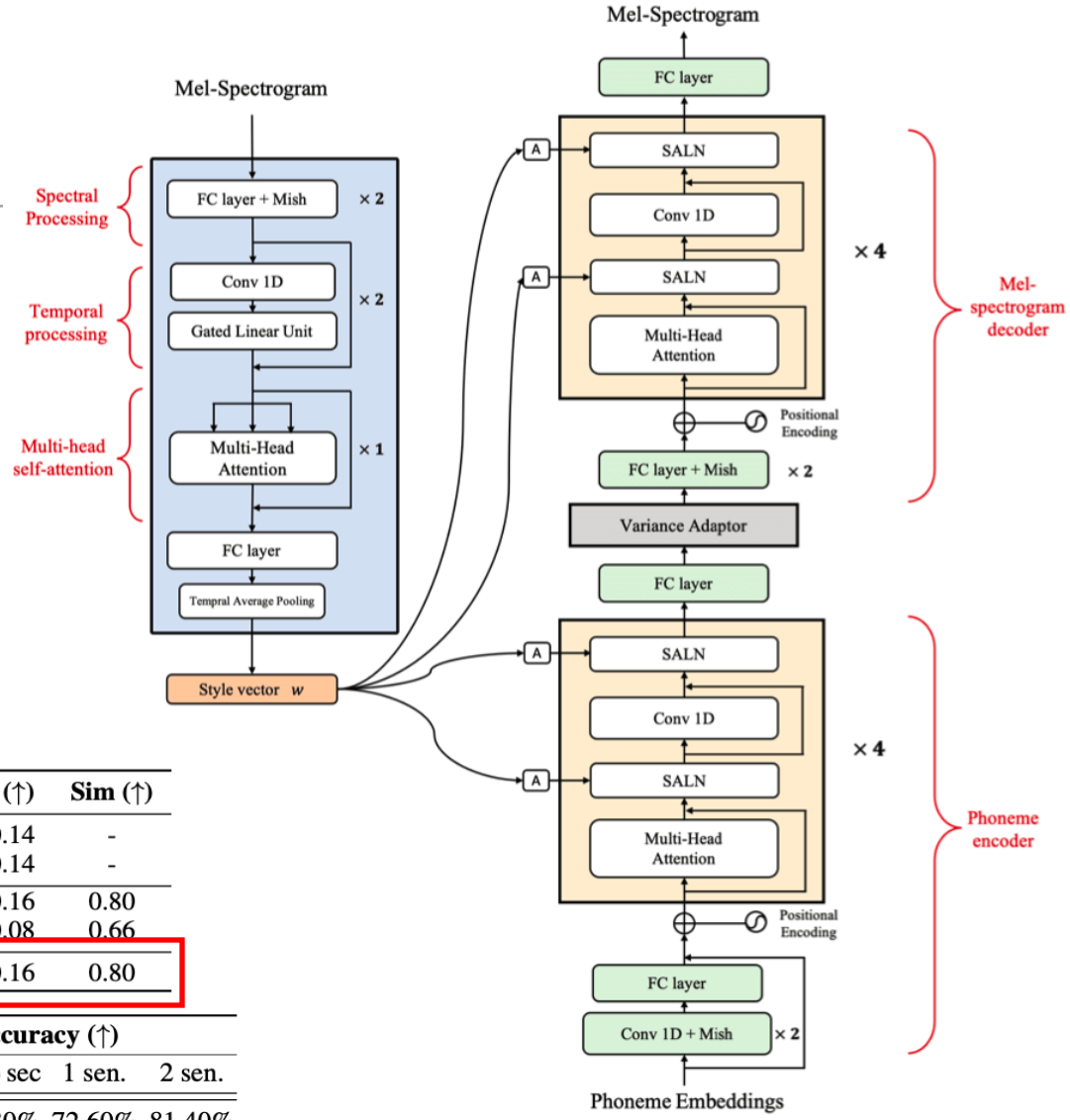
Representation Learning

从 Speaker Encoder 到 Style Encoder

StyleSpeech=Mel-Style Encoder + Speaker-Adaptive LayerNorm (SALN)

对比实验

- GT mel + Vocoder: GT 特征 + MelGAN 声码器
- Multi-speaker FS2: FastSpeech2 (no SALN)
- Multi-speaker FS2+d-vector: FastSpeech2 + Speaker Encoder
- StyleSpeech: Mel-Style Encoder + SALN



Model	MOS (↑)	MCD (↓)	WER (↓)
GT	4.37±0.15	-	-
GT *mel* + Vocoder	4.02±0.15	-	-
Multi-speaker FS2(*vanilla*)	3.53±0.13	4.50±0.21	16.49
Multi-speaker FS2+*d*-vector	3.46±0.13	4.53±0.21	16.51
StyleSpeech	3.84±0.14	4.49±0.22	16.79

Model	SMOS (↑)	Sim (↑)
GT	4.75±0.14	-
GT *mel* + Vocoder	4.57±0.14	-
Multi-speaker FS2(*vanilla*)	3.41±0.16	0.80
Multi-speaker FS2+*d*-vector	2.42±0.08	0.66
StyleSpeech	3.83±0.16	0.80

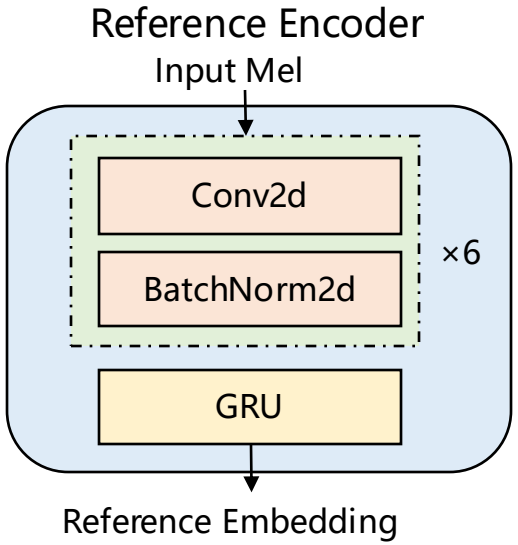
Metric	SMOS (↑)				Sim (↑)				Accuracy (↑)			
Length	<1 sec	1~3 sec	1 sen.	2 sen.	<1 sec	1~3 sec	1 sen.	2 sen.	<1 sec	1~3 sec	1 sen.	2 sen.
Multi-speaker FS2(*vanilla*)	3.14±0.17	3.63±0.16	3.31±0.14	3.36±0.12	0.713	0.735	0.775	0.773	64.80%	73.80%	72.60%	81.40%
Multi-speaker FS2+*d*-vector	1.85±0.12	2.08±0.16	2.11±0.16	2.12±0.14	0.601	0.603	0.619	0.616	2.40%	3.80%	5.60%	5.60%
StyleSpeech	3.32±0.16	4.13±0.16	**3.50±0.10**	3.46±0.12	0.725	0.756	0.791	0.795	77.60%	85.00%	83.46%	85.19%

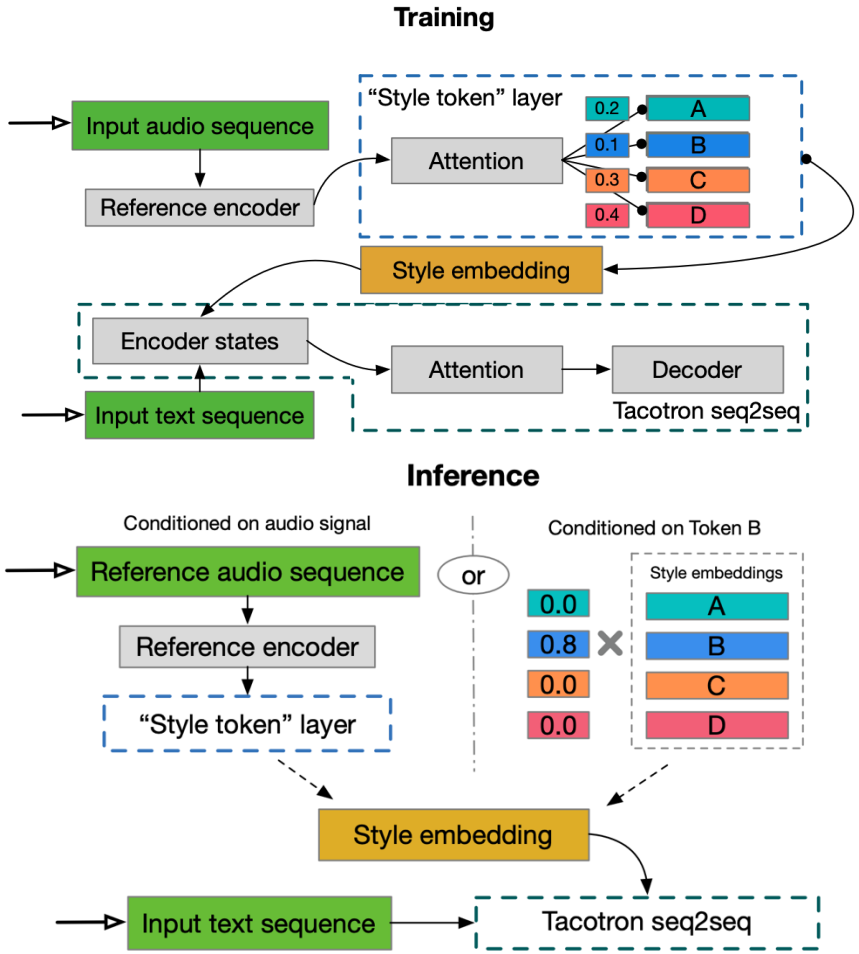
Min, Dongchan, et al. "**Meta-stylespeech: Multi-speaker adaptive text-to-speech generation**." *International Conference on Machine Learning*. PMLR, 2021.

Representation Learning

从 Speaker Encoder 到 Style Encoder

Global Style Tokens (GST)





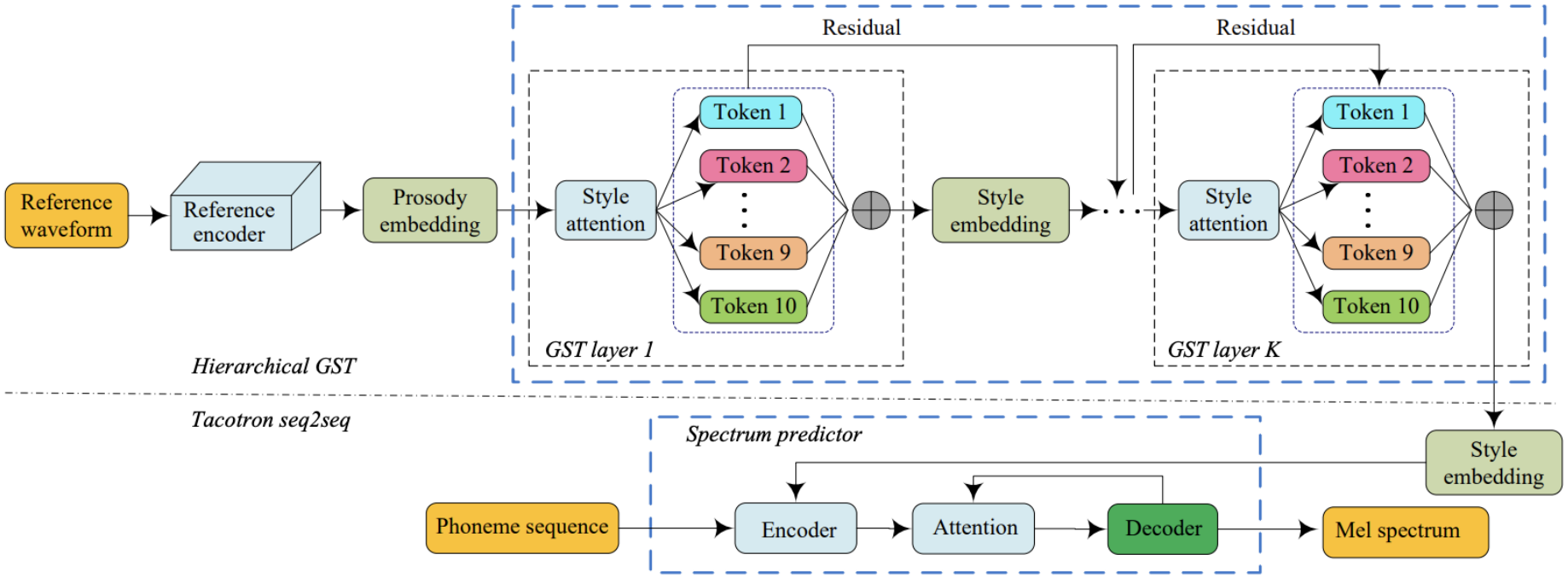
Wang, Yuxuan, et al. "**Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis**." *ICML*. PMLR, 2018.

Representation Learning

从 Speaker Encoder 到 Style Encoder

Hierarchical GST

- 第一层 GST： Speaker-ID 对应的音色
- 第二/三层 GST： 说话人风格或者情感变化

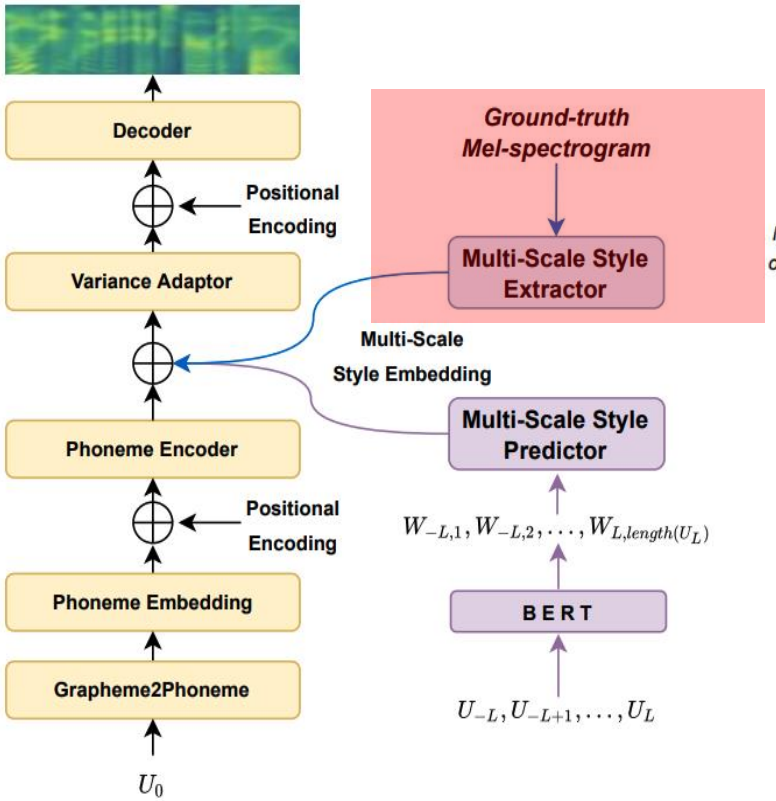


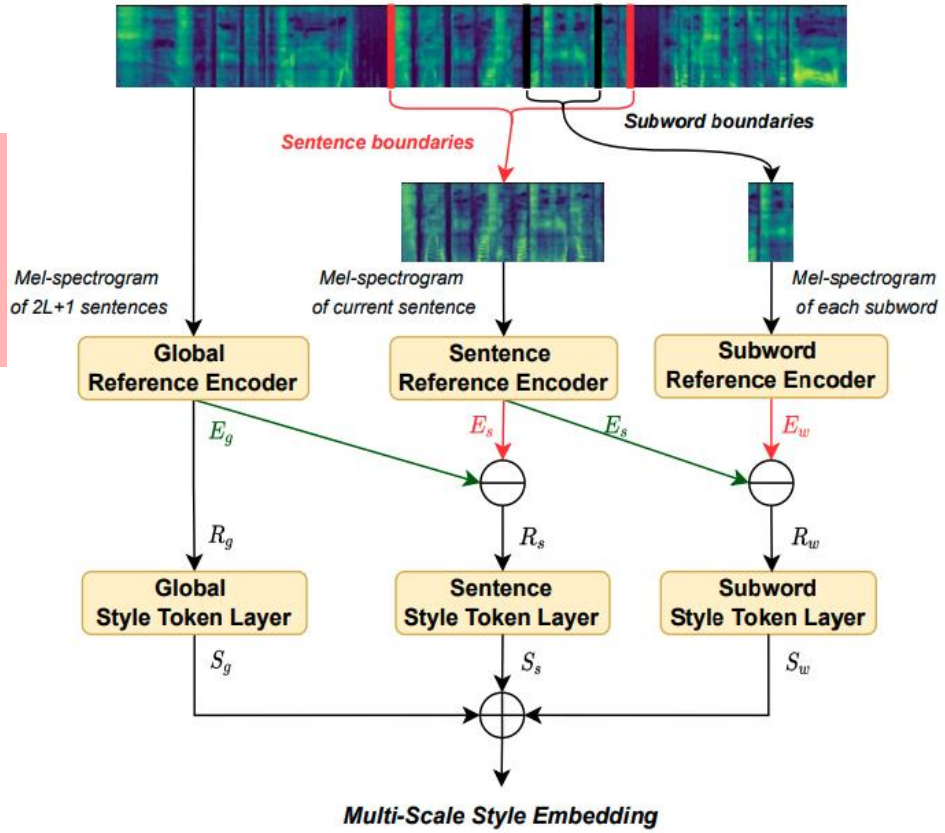
An, Xiaochun, et al. "**Learning hierarchical representations for expressive speaking style in end-to-end speech synthesis**." *IEEE ASRU,* 2019.

Representation Learning

从 Speaker Encoder 到 Style Encoder

Multi-Scale GST





denoted as a subword reference embedding E_w. The lower-level embedding E_w may contain redundant style information which has already been covered in the higher-level embedding E_s, and similarly for E_s and E_g. To reduce such overlapping, the residuals between three reference embeddings are represent as the style variation, which can be described as:

$$R_g = E_g \quad (1)$$

$$R_s = E_s - E_g \quad (2)$$

$$R_w = E_w - E_s \quad (3)$$

where R_g, R_s and R_w is the residual style embedding of global-level, sentence-level and subword-level respectively.

Model	CMOS
Proposed	0
-global-level style	−0.428
-multi-scale framework	−0.640
-residual style embedding	−0.516

Lei, Shun, et al. "**Towards Expressive Speaking Style Modelling with Hierarchical Context Information for Mandarin Speech Synthesis**." *ICASSP* IEEE, 2022.

Representation Learning

总结一：Speaker Encoder 和 Style Encoder 的多角度对比

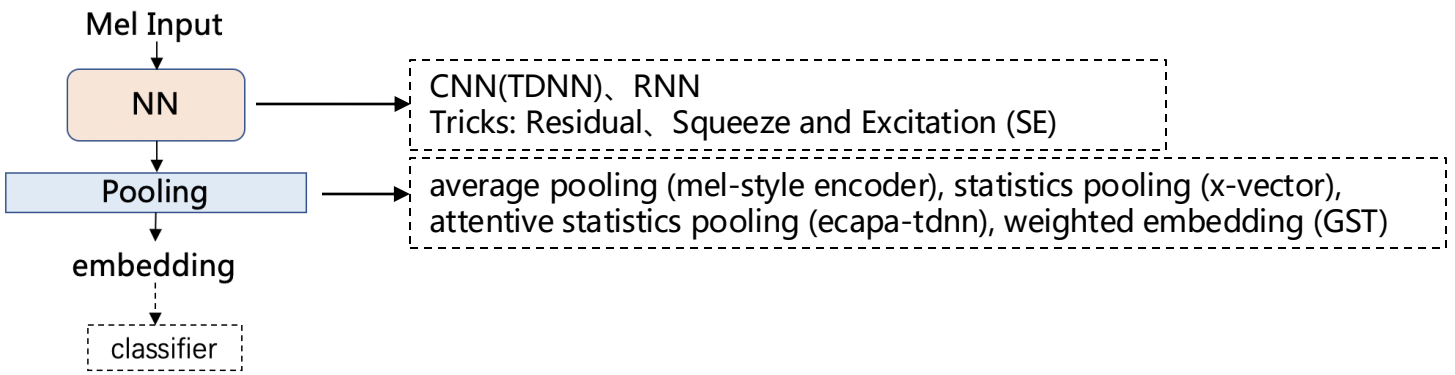
1. 建模目的

- 相同点：utterance 级别的表征学习，embedding 作为模型的 conditional 输入
- Speaker Encoder 更强调说话人**音色**属性，但不排除糅合了说话人的音色/风格等多方面信息
- Style Encoder 更强调说话人的**风格**属性，但在 Voice Cloning 任务上也可以作为说话人的整体表征

2. 训练方式

- Style Encoder 和声学模型联合训练时，**没有额外的损失函数**，监督信息完全来自声学特征预测的目标
- Speaker Encoder 和声学模型联合训练时，需要增加额外的说话人分类损失函数，进行**多任务学习**

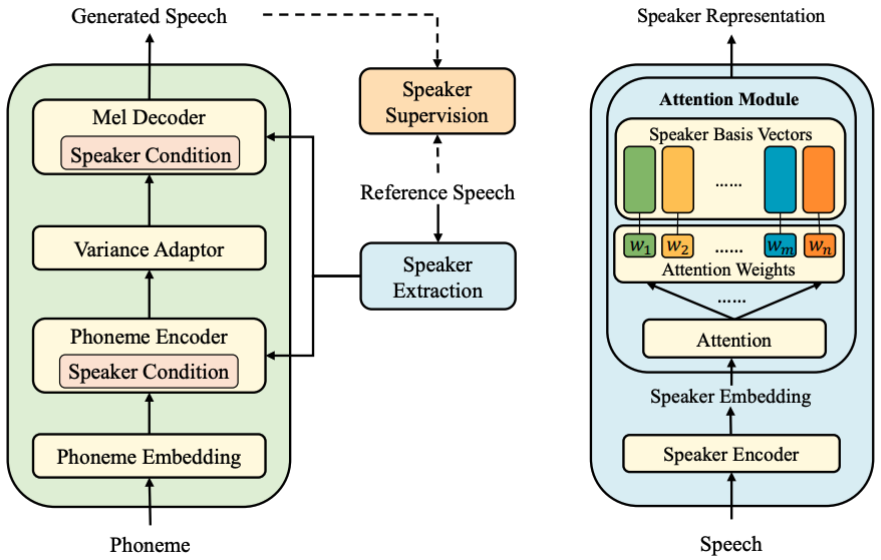
3. 模型结构设计



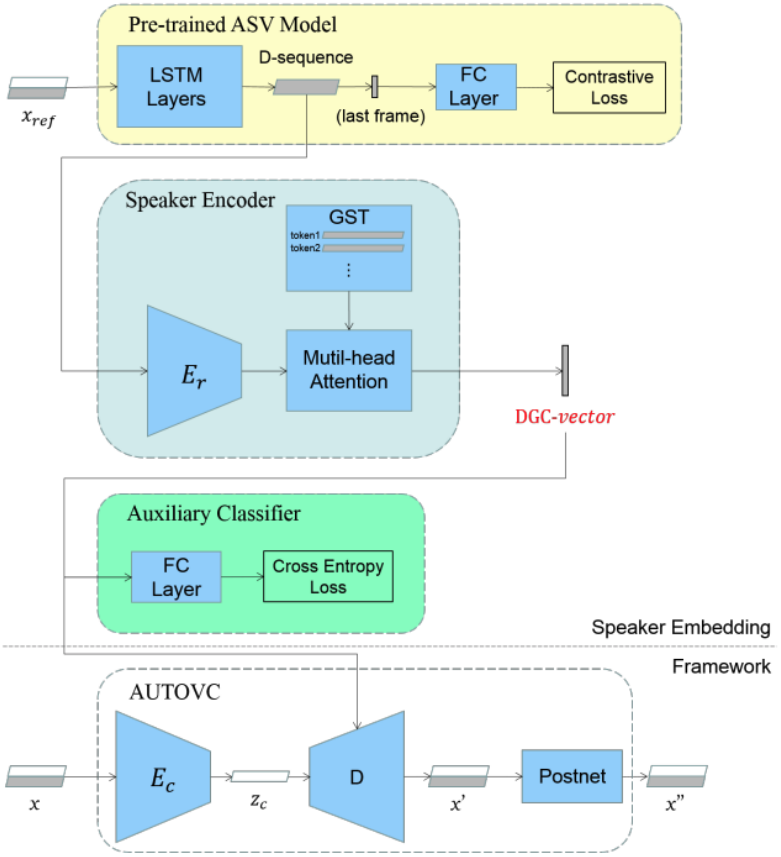
Representation Learning

总结二：GST 作为 Speaker Encoder (说话人分解)

- 将说话人用一组基向量张成的空间来建模，基向量加权后作为说话人表征向量
- 将 GST 的 token 视为一组说话人基底向量，输入的 speaker embedding 被分解为基向量的线性加权，权重来自 speaker embedding 和基底向量的 attention 计算



正交基限制



Wu, Yihan, et al. "**Adaspeech 4: Adaptive text to speech in zero-shot scenarios**." Interspeech 2022.
Xiao, Ruitong, Haitong Zhang, and Yue Lin. "**DGC-vector: A new speaker embedding for zero-shot voice conversion**." *ICASSP* IEEE, 2022.

Meta-TTS: Meta-Learning for Few-Shot Speaker Adaptive Text-to-Speech

TASLP 2022

Meta-Learning

Meta-Learning 元学习

- 思想：学习如何学习，**learning to learn**
- 不是学习一个直接用于预测的数学模型
- 而是学习“如何**更快更好**地学习一个数学模型”

授人以鱼不如授人以渔
传授给人已有的知识
不如传授给人学习知识的方法

问题设定

1. 数据量少：少样本学习、语音克隆
2. 学习更快：Speaker Adaptation 迭代次数多，希望减少迭代次数
3. 泛化性更好：提高新说话人上的合成效果

- 小任务：模仿 apple、banana发音来学习；对于新单词 strawberry，必须听到**发音**才能学会
- 元学习：目标仍然是学习 strawberry 的发音，但元学习不强调特定某个单词的学习目标，而是**学习音标的发音**，之后只要根据音标**很快**就能掌握新单词的发音
- 不限于一种语言，总结出**国际音标 IPA**，在 IPA 基础上加以变通，可以**更快**学会新语言的发音

MAML: Model-Agnostic Meta-Learning

MAML: 模型无关的元学习

Model Agnostic： 模型无关的/模型不可知的、不对模型进行任何假设

核心目标： 更好的模型初始化权重，学习快速 Adaptation 到其他 task 的能力

基础概念：

Model-agnostic meta-learning for fast adaptation of deep networks
C Finn, P Abbeel, S Levine - … on machine **learning**, 2017 - proceedings.mlr.press
… for **meta-learning** that is **model-agnostic**, in the sense that it is compatible with any **model** trained with gradient descent and applicable to a variety of different **learning** problems, …
☆ Save 99 Cite Cited by 6954 Related articles All 15 versions »

专业名词	说明
	任务：在 C1-C10 上预训练模型，再在T1-T5上微调得到目标任务的模型
Base-Learner	图像分类任务：对标签未知的图片在 T1-T5 上进行分类 (真正的目标)
Meta-Learner	基于现有数据，元学习一个 base 模型 (预训练得到初始化权重)
Meta-Train Classes	C1-C10：10 个类别，每个类别 30 张图片，共 300 张图片
Meta-test Classes	T1-T5：5 个类别，每个类别 20 张图片，共 100 张图片
N-way K-shot	N 表示类别的个数，K 表示每个类别的样本数
5-way 10-shot	**5-way：** 预训练/Meta-train 阶段，从 Meta-train 数据 C1-C10 10个类别中，随机选择 5 类，用这 5 类的 K-shot 数据作为一次分类**任务 T**； **10-shot：** 每个类别选择 10 个样本作为任务 T 的训练集，剩余样本作为任务 T 的测试集 训练集称为任务 T 的 **support set**, 测试集称为任务 T 的 **query set**
Task (meta-task)	每个 N-way K-shot 的分类任务即为 **meta-task**

MAML: Model-Agnostic Meta-Learning

MAML: 模型无关的元学习

MAML 的训练数据

- MAML 每个 N-way K-shot 的任务 (meta-task)，相当于普通深度学习概念下的一个样本
- MAML 的 batch 是由一组 task (每个 task 都有 support set 和 query set) 构成的

MAML 的预训练

4-7: 对于每个 task，复制模型得到副本，进行内层优化及参数更新

5: N 类每个类别 K 个 support 样本，基于 N×K 个 support 样本计算梯度

6: 基于上一步的梯度结果，更新每个副本模型的参数

8: Meta-Learning 的核心步骤，称为 **meta-update**

- Loss 是在是各 task 的 query set 用相应的更新后的副本模型上计算得到，梯度下降更新参数作用于原始的预训练模型上

MAML 的微调

- 预训练的模型参数作为初始参数
- 理论上只有一个 K-shot 的 task，没有 query set，所以只有内部优化：使用 task 的 support set 微调模型

Algorithm 1 Model-Agnostic Meta-Learning

Require: $p(\mathcal{T})$: distribution over tasks
Require: α, β: step size hyperparameters
1: randomly initialize θ
2: **while** not done **do**
3: Sample batch of tasks $\mathcal{T}_i \sim p(\mathcal{T})$
4: **for all** $\mathcal{T}_i$ **do**
5: Evaluate $\nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta)$ with respect to K examples
6: Compute adapted parameters with gradient descent: $\theta_i' = \theta - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta)$
7: **end for**
8: Update $\theta \leftarrow \theta - \beta \nabla_\theta \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_i'})$
9: **end while**

Meta-TTS

Speaker Adaptation + Meta-Learning

- 出发点：Speaker Adaptation 效果好，但需要多次迭代，时间和计算资源消耗高，难以满足实际落地需要
- Meta-Learning 解决的问题 / 优势：**Fast Adaptation**，模型微调更高效
- **Meta-TTS = Speaker Adaptation + Meta-Learning**，优势互补，更快实现高质量的语音克隆

Meta-TTS 的修改

- Meta-learning 内层优化时默认更新全部参数

- Meta-TTS 内层优化：只更新 speaker 相关的参数，与 finetune 时操作一致

$$\boldsymbol{\theta}_i = \{\boldsymbol{\theta}_E, \boldsymbol{\theta}_{VA,i}, \boldsymbol{\theta}_{D,i}, E_{S,i}\}$$

$$\boldsymbol{\theta}_{VA,i}, \boldsymbol{\theta}_{D,i}, E_{S,i} \coloneqq \underset{\boldsymbol{\theta}_{VA}, \boldsymbol{\theta}_D, E_S}{\arg\min} \mathcal{L}\left(\boldsymbol{\theta}, \mathcal{S}_i\right)$$

- Meta-TTS 外层优化：更新模型的全部参数

$$\boldsymbol{\theta} = \{\boldsymbol{\theta}_E, \boldsymbol{\theta}_{VA}, \boldsymbol{\theta}_D, E_S\}$$

Algorithm 1 MAML for Few-Shot Learning

Require: $p(\mathcal{T})$: distribution over tasks
Require: α, β: step size hyperparameters
1: randomly initialize $\boldsymbol{\theta}$
2: **while** not done **do**
3: Sample batch of tasks $\mathcal{T}_i \sim p(\mathcal{T})$
4: **for all** $\mathcal{T}_i$ **do**
5: Sample K datapoints $\mathcal{S}_i = \{(\mathbf{x}_j^{\mathcal{S}_i}, \mathbf{y}_j^{\mathcal{S}_i})\}_{j=1}^{K}$ from $\mathcal{T}_i$
6: Let $\boldsymbol{\theta}_i \leftarrow \boldsymbol{\theta}$
7: **for** N gradient decent steps **do**
8: Inner update: $\boldsymbol{\theta}_i \leftarrow \boldsymbol{\theta}_i - \alpha \nabla_{\boldsymbol{\theta}_i} \mathcal{L}\left(\boldsymbol{\theta}_i, \mathcal{S}_i\right)$
9: **end for**
10: Sample Q datapoints $\mathcal{Q}_i = \{(\mathbf{x}_j^{\mathcal{Q}_i}, \mathbf{y}_j^{\mathcal{Q}_i})\}_{j=1}^{Q}$ from $\mathcal{T}_i$ for the meta-update
11: **end for**
12: Meta-update: $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \beta \nabla_{\boldsymbol{\theta}} \mathbb{E}_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}\left(\boldsymbol{\theta}_i, \mathcal{Q}_i\right)$
13: **end while**

Meta-TTS

Meta-TTS 的实验设计

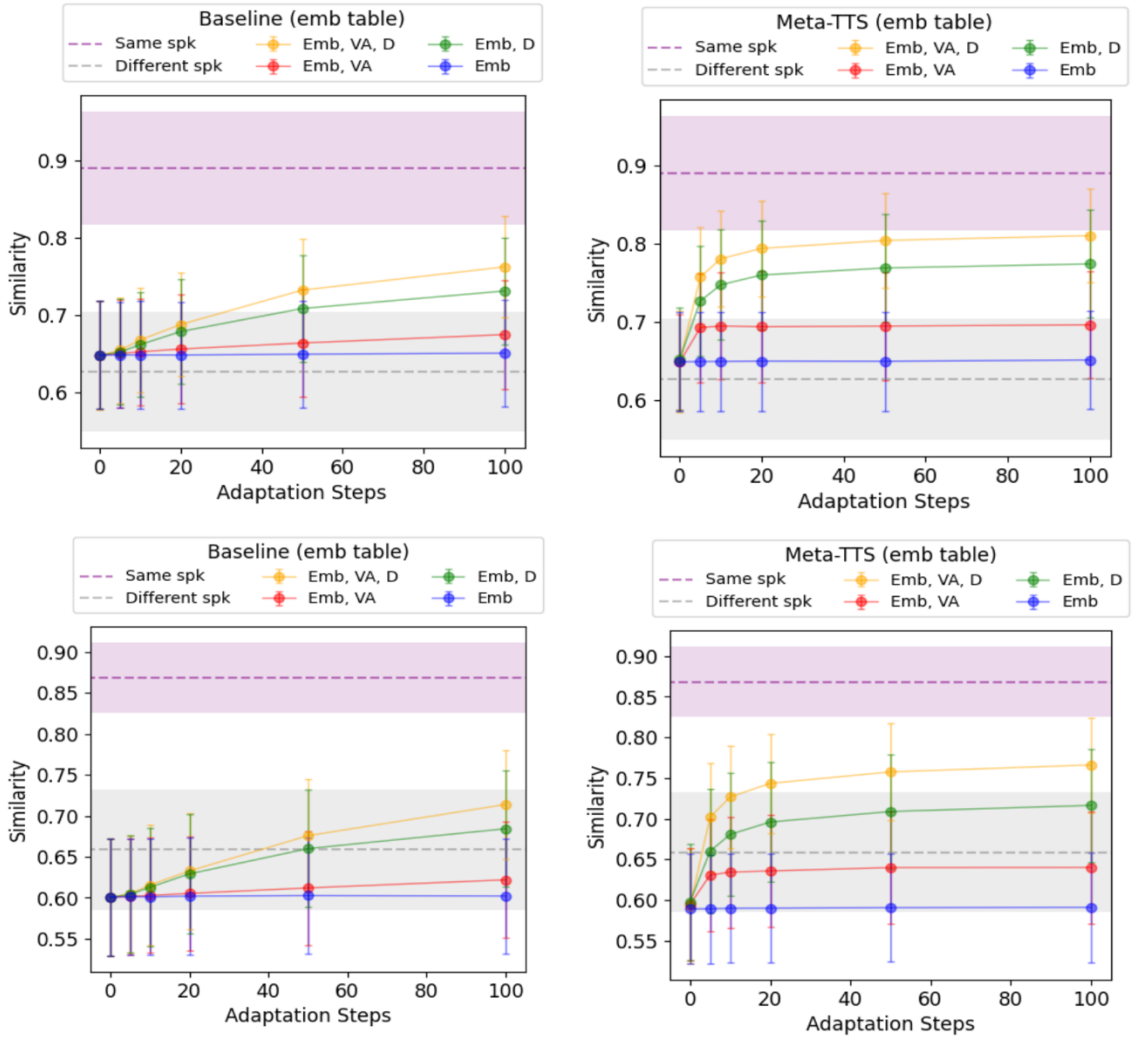
论文以 5-shot Voice Cloning 为例

- **Meta-train**：LibriTTS (train-clean-100)， 247 个说话人，54 小时音频
- **Meta-test**：LibriTTS 30 个其他说话人；VCTK 80 个说话人，每个说话人 1 个 meta-task
- 每个 meta-task：support set = 5，query set = 5
  - support set 学习该说话人的特性，微调声学模型
  - query set 预测该说话人的语音特征
  - support 和 query 样本**来自同一个说话人**
    - 先从 Meta-train 中随机选择一个说话人
    - 再从说话人的样本中，随机选择 5 + 5 = 10 条音频作为 task 的 support / query set
- 每个 batch 8 个 task，实际有 **8 × (5 + 5) = 80** 条音频；baseline 的 speaker adaptation 同样设置 batch_size=80
- 声码器使用 **MelGAN**

Meta-TTS

Meta-TTS 的实验细节

Approach	Adaptation	LibriTTS	VCTK
		MOS	MOS
Ground truth			
Real utterances		4.10 ± 0.16	4.04 ± 0.11
Reconstructed		3.47 ± 0.30	3.38 ± 0.15
With embedding table			
Baseline	10 steps	2.92 ± 0.23	3.40 ± 0.15
Meta-TTS	10 steps	$\mathbf{2.95 \pm 0.19}$	$\mathbf{3.44 \pm 0.14}$



Summary

总结

1. Speaker Encoder / Style Encoder / Speaker Adaptation 的组合
2. GST 结构的尝试
    - 基于 GST 的说话人空间建模，将输入说话人信息分解为基向量的自动加权表征
    - 不同结构的尝试：Hierarchical GST、Multi-Scale GST
3. MAML 元学习应用于模型的预训练

扩展

1. 模型的快速个性化适应 (语音识别/口语评测)
2. GST 变体用于音频特征的提取或者 embedding 的进一步分解

谢 谢!

An, Xiaochun, et al. "**Learning hierarchical representations for expressive speaking style in end-to-end speech synthesis**." *IEEE ASRU,* 2019.

Q & A

Hierarchical GST

- 动机：对于很多说话人、每个说话人风格多样的语音数据，论文希望通过多层 GST，***将说话人的基础信息(音色)和说话人的风格信息「解耦」***
- 在说话人、风格甚至情感这些不同的层次上，用**不共享**的 token embedding 来建模，而不希望只用一层 token 个数很多的 GST 来表示。

关于图中两个残差具体如何实现的，论文没有任何细节表述，但是论文给出的动机是：*the residuals can increase the style information for decomposition*，残差能够给深层GST的细粒度风格分解补充一些风格信息。