

# SELECTIVE CLASSIFIER-FREE GUIDANCE FOR ZERO-SHOT TEXT-TO-SPEECH

*John Zheng, Farhad Maleki*

University of Calgary, Department of Computer Science, Canada

## ABSTRACT

In zero-shot text-to-speech, achieving a balance between fidelity to the target speaker and adherence to text content remains a challenge. While classifier-free guidance (CFG) strategies have shown promising results in image generation, their application to speech synthesis are underexplored. Separating the conditions used for CFG enables trade-offs between different desired characteristics in speech synthesis. In this paper, we evaluate the adaptability of CFG strategies originally developed for image generation to speech synthesis and extend separated-condition CFG approaches for this domain. Our results show that CFG strategies effective in image generation generally fail to improve speech synthesis. We also find that we can improve speaker similarity while limiting degradation of text adherence by applying standard CFG during early timesteps and switching to selective CFG only in later timesteps. Surprisingly, we observe that the effectiveness of a selective CFG strategy is highly text-representation dependent, as differences between the two languages of English and Mandarin can lead to different results even with the same model.

**Index Terms**— Classifier-free guidance, voice cloning, text-to-speech, speech synthesis, flow matching

## 1. INTRODUCTION

**Classifier-free guidance (CFG)** [1] is a key component of iteratively denoising generative models such as diffusion or flow matching. Flow matching [2]—originally tested in image generation—has been used successfully in zero-shot text-to-speech (TTS) beginning with Voicebox [3]. Other state-of-the-art (SOTA) zero-shot TTS models continue to utilize flow matching such as F5-TTS [4], CosyVoice 3 [5], MegaTTS 3 [6], and Minimax-Speech [7].

Techniques inspired by CFG are also used in both diffusion and flow matching-based image generation models, such as weight schedules [8], different types of negative prompting [9, 10, 11], modifications to the CFG algorithm [12, 13, 14], or converting CFG to training-time target modification [15]. The last technique of training-time target modification has been already been applied to speech synthesis [16], but the other methods have not. We reduce this gap in the literature by evaluating several weight schedules [8] and the methods proposed in CFG-Zero\* on a zero-shot TTS model.

One CFG strategy used for speech synthesis in VoiceLDM [17], DualSpeech [18], and MegaTTS 3[6] is that of selectively emphasizing different input conditions by using separate CFG weights. This allows a user to trade-off between different output characteristics. However, these techniques require additional model evaluations during inference, and degradation in the non-emphasized condition remains undesirable. In this study, we propose applying regular CFG during early timesteps and switching to separated CFG to emphasize speaker conditioning in later timesteps. This approach does not require additional model evaluations compared to regular CFG.

However, we find that model and language differences affect which CFG strategies are effective. Our proposed strategy is effective for F5-TTS [4] in English, but for Mandarin with the same model, selective CFG does not improve speaker similarity. With CosyVoice 2 [19], we find that selective CFG simply improves speaker similarity without any degradation in text adherence, so our proposed strategy is unnecessary.

## 2. BACKGROUND

### 2.1. Zero-shot Text-to-Speech

Zero-shot TTS is a type of voice cloning restricted to only a single sample of the speaker’s voice, optionally also including a transcript of the sample. Zero-shot TTS is one of the most common forms of controllable TTS [20] and can be further combined with other functionalities such as emotion control or pitch control.

Two standard objective metrics for evaluating zero-shot TTS systems are similarity score (SIM) and word error rate (WER). SIM is measured by using a speaker verification system such as WavLM Large [21] to extract speaker embeddings from both the reference audio and the generated audio, with the cosine similarity between the two embeddings reported as SIM. WER is calculated by comparing the input text with the results of an automatic speech recognition (ASR) model on the generated audio and calculating the percentage of words that are added, removed, or substituted. SIM and WER are useful metrics for zero-shot TTS systems as they individually measure adherence to the reference audio and the input text, respectively.

### 2.2. Flow Matching

Flow matching [2] is a method of efficiently training continuous normalizing flows. While flow matching was proposed as a general framework for modeling smooth transformations between distributions, optimal transport (OT) flow matching is proposed as the transformation with the straightest path through the data space. This allows OT to maintain fidelity while requiring less inference steps than other iteratively denoising algorithms such as diffusion. When applied in generative models, flow matching primarily refers to the use of OT flow matching.

The training and inference algorithms for OT flow matching are as follows. Let  $x_1$  be the original audio signal and  $x_0$  be a sample of Gaussian noise. During training, a random timestep  $t \in [0, 1]$  is sampled, and the model receives as input  $x_t = x_0 + t \times (x_1 - x_0)$  and embedding for  $t$ . The prediction target is  $x_1 - x_0$  with  $L_2$  loss, which is interpreted as the derivative with respect to  $t$  of a linear function between  $(x_0, t = 0)$  and  $(x_1, t = 1)$ . The inference algorithm is an ordinary differential equation (ODE) from the initial value of pure noise  $x_0$  at  $t = 0$  to the final value of predicted speech at  $\hat{x}_1$  at  $t = 1$  by integrating from  $t = 0$  to 1 with an ODE solver, using the model prediction as the derivative.

### 2.3. Classifier-free Guidance

Classifier-free guidance (CFG) [1], originally introduced in diffusion models, involve amplifying the difference between a conditioned and unconditioned prediction to increase the effect of the conditioning. Given  $\lambda$  as the constant CFG weight, it can be implemented as replacing the conditioned model prediction  $\epsilon(x_t, c)$  with the following:

$$\hat{\epsilon}(x_t, c) = \epsilon(x_t, c) + \lambda(\epsilon(x_t, c) - \epsilon(x_t))$$

where  $\epsilon(x_t)$  is the unconditioned model prediction. The same concept can be applied to flow matching, where the predicted derivative  $\epsilon(x_t, c)$  is replaced with  $\epsilon(x_t, c) + \lambda(\epsilon(x_t, c) - \epsilon(x_t))$ . CFG was originally formalized as creating an implicit classifier based on prior work with classifier guidance [22], and explanation can be found in the original [1].

### 2.4. CFG for Image Generation

Negative prompting is an extension of concepts introduced by CFG, originally used in image generation models such as Stable Diffusion and explored in other academic works [9, 10]. Negative prompting is used to prevent the generation of certain features in image generation by replacing the unconditioned prediction in CFG with a prediction conditioned on the unwanted feature. With  $c^-$  as the unwanted condition, the modified prediction becomes:

$$\hat{\epsilon}(x_t, c, c^-) = \epsilon(x_t, c) + \lambda(\epsilon(x_t, c) - \epsilon(x_t, c^-)).$$

It was found that negative prompts have different effects on across timesteps [10], with negative prompts restricted to early timesteps potentially *introducing* the unwanted feature to an image that otherwise did not contain it, successfully removing features if used in middle timesteps, and largely having no effect if used only in later timesteps.

Perp-Neg [9] and CFG-Zero\* [13] both calculate the perpendicular component between the two prediction vectors instead of just the difference between them. Perp-Neg uses the component of the negative prediction perpendicular to the positive prediction, while CFG-Zero\* uses the component of the positive prediction perpendicular to the unconditioned prediction. (CFG-Zero\* uses a different formalization, but the resulting algorithm is identical to what we describe.) CFG-Zero\* also suggests the zero-init strategy, where ignoring the first few update steps during flow matching (i. e. beginning flow matching from  $t > 0$  while still using pure noise as the starting point) may improve generation results, especially for underfitted models.

Wang et al. [8] analyzed different CFG weight schedules for image generation and finds that linearly decreasing weight schedules, optionally clamped to not decrease below a lower bound at later timesteps, can have beneficial results.

### 2.5. Separated-condition CFG for Speech Synthesis

VoiceLDM [17] uses two conditions for environmental speech synthesis, the script for the generated speech and a description of the environmental context. It uses two separate CFG weights for the text script condition  $c_{text}$  and the description condition  $c_{desc}$ , with the guided prediction as follows:

$$\begin{aligned} \hat{\epsilon}(x_t, c_{text}, c_{desc}) &= \epsilon(x_t, c_{text}, c_{desc}) \\ &+ \lambda_{text}(\epsilon(x_t, c_{text}) - \epsilon(x_t)) \\ &+ \lambda_{desc}(\epsilon(x_t, c_{desc}) - \epsilon(x_t)) \end{aligned}$$

DualSpeech [18] uses a similar formulation, replacing the environmental description conditioning with speaker conditioning  $c_{spk}$ .

$$\begin{aligned} \hat{\epsilon}(x_t, c_{text}, c_{spk}) &= \epsilon(x_t, c_{text}, c_{spk}) \\ &+ \lambda_{text}(\epsilon(x_t, c_{text}) - \epsilon(x_t)) \\ &+ \lambda_{spk}(\epsilon(x_t, c_{spk}) - \epsilon(x_t)) \end{aligned}$$

Notably, Mega-TTS 3 [6] changes the separated-condition CFG from DualSpeech by adding text conditioning to both conditioned and unconditioned predictions for emphasizing speaker conditioning. It uses the following formulation:

$$\begin{aligned} \hat{\epsilon}(x_t, c_{text}, c_{spk}) &= \underline{\epsilon(x_t)} \\ &+ \lambda_{text}(\epsilon(x_t, c_{text}) - \epsilon(x_t)) \\ &+ \lambda_{spk}(\underline{\epsilon(x_t, c_{text}, c_{spk})} - \epsilon(x_t, c_{text})) \end{aligned}$$

with changes from DualSpeech underlined.

The authors of Mega-TTS 3 find that as  $\lambda_{text}$  increases, it starts with poor text adherence at  $\lambda_{text} = 1$ , shifts towards accented pronunciation around  $\lambda_{text} = 1.5$  to  $\lambda_{text} = 2.5$ , and then finally towards standard pronunciation at  $\lambda_{text} = 4$  and beyond [6].

## 3. TEXT-TO-SPEECH SYSTEMS

We investigate two strong (SOTA or near-SOTA) open weight models, F5-TTS [4] and CosyVoice 2 [19]. We selected these models primarily based on availability, as many other state-of-the-art zero-shot TTS models are not publicly available (most notably Seed-TTS [23]). In addition, these models represent the two most popular flow-matching TTS paradigms: purely non-autoregressive flow matching and flow matching on autoregressively generated speech tokens.

F5-TTS is based on E2-TTS [24], primarily trained with the open YouTube-based dataset Emilia [25] and utilizing an updated architecture and sampling strategy compared to E2-TTS. The model utilizes input audio and input text as conditioning, where the input audio is an audio clip of the voice to be cloned, and the input text is the transcript of the input audio concatenated with the desired speech text. The best-performing released checkpoint of this model has 336 million parameters. The version of the model included in the original paper has 0.66 SIM score and 0.024 WER on LibriSpeech, but later updates to the model improve it to 0.676 SIM and 0.020 WER. We consider the best iteration of this model as open-weight because the exact training or fine-tuning methodology of the updated model checkpoint is unpublished. The model uses a text embedder and a Diffusion Transformer backbone with 4.3 and 332 million parameters, respectively.

CosyVoice 2 [19] is the second generation of CosyVoice models. There is a newer CosyVoice 3 [5] by the same team, but as CosyVoice 3 is not public, we use the older CosyVoice 2 for our experiments. CosyVoice 2 utilizes a pre-trained LLM, Qwen2.5-0.5B, that has been fine-tuned to autoregressively generate the intermediate representations of an ASR model based on the input text. The intermediate representations of the ASR model, referred to as semantic tokens, capture the text information. The reference audio and transcript are processed into a speaker embedding. The semantic tokens and speaker embedding are used as input to a 71 million-parameter flow matching model.

Both models use a cosine scheduler, where timesteps are not linearly distributed. For  $n$  total inference timesteps, the  $i$ th timestep is determined by the following equation:

$$t_i = 1 - \cos\left(\frac{\pi}{2n} \times (i - 1)\right)$$

This biases inference towards smaller updates at the start, when noise

levels are high. F5-TTS defaults to 32 timesteps while CosyVoice 2 defaults to 10 timesteps. Both models perform flow matching with mel-spectrograms, and the generated mel-spectrogram is then decoded by a vocoder into audio.

#### 4. METHODOLOGY

We propose three selective CFG strategies. In the first, the `input_text` condition replaces the unconditioned prediction with a prediction partially conditioned on the input text, as shown below:

$$\hat{\epsilon}(x_t, c_{text}, c_{spk}) = \epsilon(x_t, c_{text}, c_{spk}) + \lambda(\epsilon(x_t, c_{text}, c_{spk}) - \epsilon(x_t, c_{text})).$$

It is also equivalent to setting  $\lambda_{text} = 1$  in the separated-condition CFG used by MegaTTS 3.

The second condition we propose is `def_text`, which uses standard CFG for early timesteps below a certain threshold  $t_{threshold}$  and uses the same strategy as `input_text` for all timesteps above  $t_{threshold}$ . This design is motivated by listening to an extrapolated generation at each timestep, where the extrapolated signal is defined as  $x_t + (1-t)\epsilon(x_t, c)$ . We observe that the words become audible very quickly, at around 6 timesteps with  $t \approx 0.04$ . We hypothesize that CFG for text adherence may not be necessary after the initial steps, since the conditioned model has already captured sufficient text information. The timestep-dependent nature of conditioning has also been reported in previous work on negative prompting [10]. We evaluated candidate values of  $t_{threshold} \in \{0.02, 0.04, 0.06, 0.08\}$  on F5-TTS and found that  $t_{threshold} = 0.08$  achieves a good balance between improved SIM while minimizing WER increases.  $t_{threshold}$  is between the 9th and 10th timesteps for F5-TTS and the 3rd and 4th timesteps for CosyVoice 2. The `input_text` and `def_text` conditions are evaluated using both F5-TTS and CosyVoice 2 on LibriSpeech [26] and the English and Mandarin subsets of Seed-TTS-eval [23].

We define a third condition `input_audio` which, similarly to `input_text`, replaces the unconditioned prediction with a prediction partially conditioned on the input audio. This is only evaluated using F5-TTS on LibriSpeech [26] due to its poor performance.

For Seed-TTS-eval [23], we use the provided English and Mandarin cross-sentence prompt lists. For LibriSpeech [26], we use the 1127-sample prompt list provided by F5-TTS. Experiments with F5-TTS follow the original evaluation protocol of taking the average of three seeded trials [4], but differences between seeds are insignificant. For experiments with CosyVoice 2 we perform only one trial.

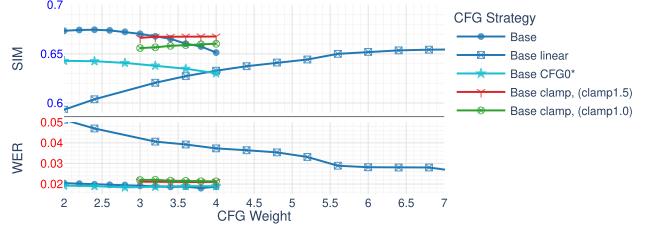
We also evaluate the weight schedules proposed by [8]—namely the linearly increasing schedule and the clamped-minimum linearly increasing schedule—as well as the perpendicular re-weighting and zero-init strategies introduced in CFG-Zero\* [13], using F5-TTS on the LibriSpeech [26] test set.

For F5-TTS,  $c_{spk}$  is treated as the input audio and  $c_{text}$  is treated as the transcript text concatenated with the input text. The transcript text could arguably be considered part of  $c_{spk}$ , but we leave this for future research. It is worth noting that F5-TTS is trained in fully conditioned, text only, and fully unconditioned modalities [4].

For CosyVoice 2,  $c_{spk}$  is treated as the speaker embedding and  $c_{text}$  as the output semantic tokens of the Qwen2.5-0.5B model.

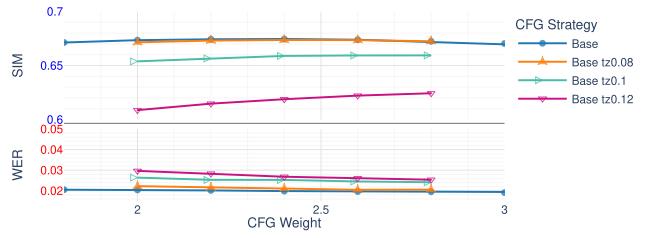
#### 5. RESULTS

The weight schedules from [8] and the perpendicular re-weighting from CFG-Zero\* [13] perform worse than the baseline, as shown in Figure 1. However, we note that high early CFG weight heavily degrade generation quality more than high CFG weight in late



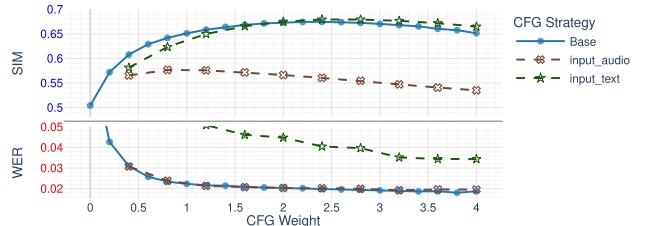
**Fig. 1:** Results of using F5-TTS with some CFG methods which reported improved image generation quality.

timesteps. Zero-init from CFG-Zero\* [13] is implemented by starting from a later timestep instead of skipping inference steps, and this also does not improve generation quality as shown in Figure 2.



**Fig. 2:** Evaluation of late timestep method from CFG-Zero\* [13], with the value  $tz$  controlling the starting timestep value. The starting timestep value is  $1 - \cos(\frac{\pi}{2} \times tz)$ , as the value  $tz$  was used before the cosine scheduler was applied.

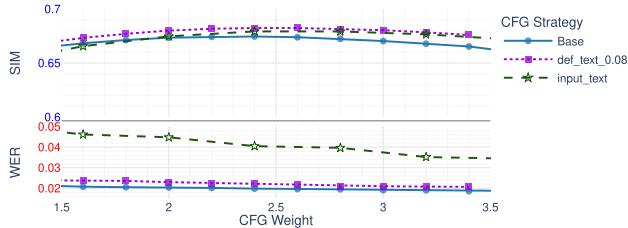
In Figure 3, we find that `input_audio` does not have lower WER. However, `input_text` does achieve a trade-off of higher SIM at the cost of worse WER.



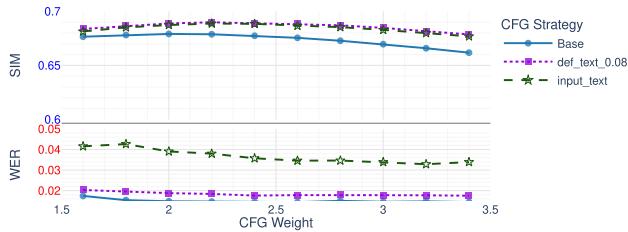
**Fig. 3:** Adding some conditioning to the unconditioned prediction, and therefore not emphasizing those conditions with CFG.

We find that `def_text` with a threshold  $t_{threshold} = 0.08$  achieves a good balance of increasing SIM with minimal impact to WER, as shown in Figure 4. This approach also works for the English subset of Seed-TTS-eval [23] as shown in Figure 5. However, using the exact same model, neither `input_text` nor `def_text` improve SIM for the Mandarin dataset and only increases WER, as shown in Figure 6.

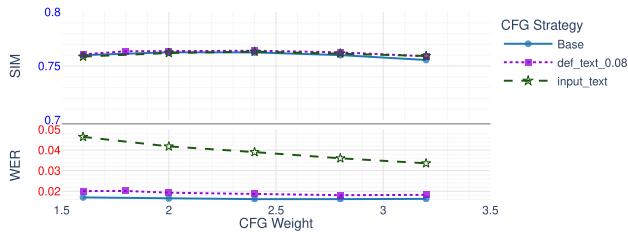
From Figure 7, we observe that with CosyVoice 2, the `input_text` condition does not degrade WER as it did with F5-TTS. We also did not observe significant language differences between English and Mandarin, unlike F5-TTS. Also we note that while CosyVoice 2 defaults to a CFG strength of 0.7, higher SIM scores appear to be achieved at CFG strength of around 1.0.



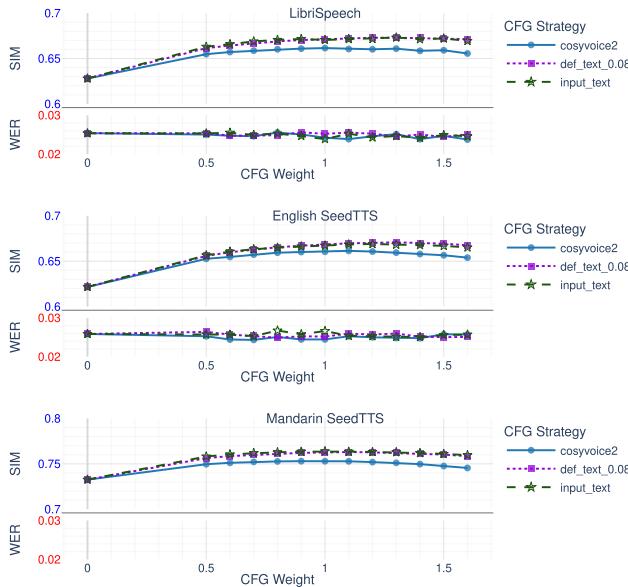
**Fig. 4:** Comparison of baseline, def\_text, and input\_text strategies on LibriSpeech with F5-TTS.



**Fig. 5:** Results on English subset of Seed-TTS-eval with F5-TTS.



**Fig. 6:** Results on Mandarin subset of Seed-TTS-eval with F5-TTS show no gain in SIM with input\_text or def\_text compared to baseline.



**Fig. 7:** Results for CosyVoice 2. Note that WER is low even without CFG (at CFG strength of 0).

## 6. DISCUSSION

Model	LibriSpeech		Seed-TTS-en		Seed-TTS-zh	
	SIM	WER	SIM	WER	SIM	WER
F5-TTS (Base)	0.675	0.020	0.679	0.018	0.763	0.017
F5-TTS (def_text)	0.682	0.022	0.690	0.018	0.764	0.019
CosyVoice 2 (Base)	0.661	0.025	0.660	0.024	0.753	0.017
CosyVoice 2 (input_text)	0.671	0.025	0.666	0.026	0.763	0.018
Minimax-Speech [7]			0.738	0.019	0.799	0.010
Seed-TTS [23]			0.762	0.022	0.796	0.011
CosyVoice 3-1.5B [5]			0.720	0.022	0.781	0.012

**Table 1:** Comparison of state-of-the-art zero-shot TTS models. Italicized results are experimentally obtained; others are reported.

We observed that the tested CFG strategies developed for image generation, namely weight schedules [8], perpendicular reweighting [13], and zero-init [13], do not generalize well to zero-shot TTS with F5-TTS as shown in Figure 1. This could be due to differences in conditioning and output modality. We suggest that future research assess the contribution of these techniques across different target modalities, such as various audio codecs [20].

Language differences have an impact on CFG strategy effectiveness for F5-TTS, but not for CosyVoice 2. This may be due to differences in text representation. CosyVoice 2 uses a 506 million-parameter LLM [19], so generated semantic tokens allow the model to achieve strong text adherence even without CFG as seen in Figure 7. F5-TTS relies on a much smaller 4.3 million-parameter ConvNeXt 2 module to process text, which may result in English and Chinese text acting as different conditioning modalities. However, the language differences do not arise in WER but rather through a failure to improve SIM when using either input\_text or def\_text conditions, which is not where language differences may be expected to arise. Another possible cause of the language difference for F5-TTS is training methodology or dataset differences, but as the training procedure for the best-performing checkpoint is not published, this remains speculation.

Even though our proposed CFG strategy can improve SIM for F5-TTS and CosyVoice 2, these improvements do not completely close the gap of state-of-the-art results reported by other closed-source models as listed in Table 1. However, given that our methods can find significant improvements with only an inference-time hyperparameter sweep, they are added gains without difficulties of training large models or collecting large, high-quality datasets.

## 7. CONCLUSION

Here, we confirm prior results [17, 18, 6] that a trade-off between speaker similarity and text adherence can be achieved by separately emphasizing the speaker conditioning with CFG. We show that by starting with regular CFG and switching to selective CFG after the several timesteps, as inspired by prior investigations into negative prompting [10], may minimize WER increases while still improving SIM. However, we demonstrate that the efficacy of separated-condition CFG depends on both the language and the model being used.

## 8. COMPLIANCE WITH ETHICAL STANDARDS

This research was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC), Discovery Grant RGPIN-2024-04966. The authors have no other relevant financial interests to disclose. This study was conducted using code and model checkpoints from F5-TTS [4], code and model checkpoints from CosyVoice 2 [19], Seed-TTS-eval [23], and LibriSpeech [26] under public domain, share-alike, or non-commercial licenses.

## 9. REFERENCES

- [1] Jonathan Ho and Tim Salimans, “Classifier-free diffusion guidance,” 2022.
- [2] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le, “Flow matching for generative modeling,” in *The Eleventh International Conference on Learning Representations*, 2023.
- [3] Matthew Le, Apoorv Vyas, Bowen Shi, Brian Karrer, Leda Sari, Rashel Moritz, Mary Williamson, Vimal Manohar, Yossi Adi, Jay Mahadeokar, et al., “Voicebox: Text-guided multilingual universal speech generation at scale,” *Advances in Neural Information Processing Systems*, vol. 36, pp. 14005–14034, 2023.
- [4] Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, Jian Zhao, Kai Yu, and Xie Chen, “F5-TTS: A fairytaler that fakes fluent and faithful speech with flow matching,” 2025.
- [5] Zhihao Du, Changfeng Gao, Yuxuan Wang, Fan Yu, Tianyu Zhao, Hao Wang, Xiang Lv, Hui Wang, Chongjia Ni, Xian Shi, Keyu An, Guanrou Yang, Yabin Li, Yanni Chen, Zhifu Gao, Qian Chen, Yue Gu, Mengzhe Chen, Yafeng Chen, Shiliang Zhang, Wen Wang, and Jieping Ye, “CosyVoice 3: Towards in-the-wild speech generation via scaling-up and post-training,” 2025.
- [6] Ziyue Jiang, Yi Ren, Ruiqi Li, Shengpeng Ji, Boyang Zhang, Zhenhui Ye, Chen Zhang, Bai Jionghao, Xiaoda Yang, Jialong Zuo, Yu Zhang, Rui Liu, Xiang Yin, and Zhou Zhao, “MegaTTS 3: Sparse alignment enhanced latent diffusion transformer for zero-shot speech synthesis,” 2025.
- [7] Bowen Zhang, Congchao Guo, Geng Yang, Hang Yu, Haozhe Zhang, Heidi Lei, Jialong Mai, Junjie Yan, Kaiyue Yang, Mingqi Yang, Peikai Huang, Ruiyang Jin, Sitan Jiang, Weihua Cheng, Yawei Li, Yichen Xiao, Yiyi Zhou, Yongmao Zhang, Yuan Lu, and Yucen He, “MiniMax-Speech: Intrinsic zero-shot text-to-speech with a learnable speaker encoder,” 2025.
- [8] Xi Wang, Nicolas Dufour, Nefeli Andreou, Marie-Paule Cani, Victoria Fernández Abrevaya, David Picard, and Vicky Kalogeiton, “Analysis of classifier-free guidance weight schedulers,” *Transactions on Machine Learning Research Journal*, 2024.
- [9] Mohammadreza Armandpour, Ali Sadeghian, Huangjie Zheng, Amir Sadeghian, and Mingyuan Zhou, “Re-imagine the negative prompt algorithm: Transform 2D diffusion into 3D, alleviate Janus problem and beyond,” 2023.
- [10] Yuanhao Ban, Ruochen Wang, Tianyi Zhou, Minhao Cheng, Boqing Gong, and Cho-Jui Hsieh, “Understanding the impact of negative prompts: When and how do they take effect?”, in *European Conference on Computer Vision*. Springer, 2024, pp. 190–206.
- [11] Felix Koulischer, Johannes Deleu, Gabriel Raya, Thomas Deemeester, and Luca Ambrogioni, “Dynamic negative guidance of diffusion models,” in *The Thirteenth International Conference on Learning Representations*, 2025.
- [12] Hyungjin Chung, Jeongsol Kim, Geon Yeong Park, Hyelin Nam, and Jong Chul Ye, “CFG++: Manifold-constrained classifier free guidance for diffusion models,” in *The Thirteenth International Conference on Learning Representations*, 2024.
- [13] Weichen Fan, Amber Yijia Zheng, Raymond A. Yeh, and Ziwei Liu, “CFG-Zero\*: Improved classifier-free guidance for flow matching models,” 2025.
- [14] Tero Karras, Miika Aittala, Tuomas Kynkänniemi, Jaakko Lehtinen, Timo Aila, and Samuli Laine, “Guiding a diffusion model with a bad version of itself,” *Advances in Neural Information Processing Systems*, vol. 37, pp. 52996–53021, 2024.
- [15] Zhicong Tang, Jianmin Bao, Dong Chen, and Baining Guo, “Diffusion models without classifier-free guidance,” 2025.
- [16] Yuzhe Liang, Wenzhe Liu, Chunyu Qiang, Zhikang Niu, Yushen Chen, Ziyang Ma, Wenxi Chen, Nan Li, Chen Zhang, and Xie Chen, “Towards flow-matching-based TTS without classifier-free guidance,” 2025.
- [17] Yeonghyeon Lee, Inmo Yeon, Juhan Nam, and Joon Son Chung, “VoiceLDM: Text-to-speech with environmental context,” in *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 12566–12571.
- [18] Jinhyeok Yang, Junhyeok Lee, Hyeong-Seok Choi, Seunghoon Ji, Hyeongju Kim, and Juheon Lee, “DualSpeech: Enhancing speaker-fidelity and text-intelligibility through dual classifier-free guidance,” in *Proc. Interspeech 2024*, 2024, pp. 4423–4427.
- [19] Zhihao Du, Yuxuan Wang, Qian Chen, Xian Shi, Xiang Lv, Tianyu Zhao, Zhifu Gao, Yixin Yang, Changfeng Gao, Hui Wang, Fan Yu, Huadai Liu, Zhengyan Sheng, Yue Gu, Chong Deng, Wen Wang, Shiliang Zhang, Zhijie Yan, and Jingren Zhou, “CosyVoice 2: Scalable streaming speech synthesis with large language models,” 2024.
- [20] Tianxin Xie, Yan Rong, Pengfei Zhang, Wenwu Wang, and Li Liu, “Towards controllable speech synthesis in the era of large language models: A survey,” 2025.
- [21] Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, et al., “WavLM: Large-scale self-supervised pre-training for full stack speech processing,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.
- [22] Prafulla Dhariwal and Alexander Nichol, “Diffusion models beat GANs on image synthesis,” *Advances in neural information processing systems*, vol. 34, pp. 8780–8794, 2021.
- [23] Philip Anastassiou, Jiawei Chen, Jitong Chen, Yuanzhe Chen, Zhuo Chen, Ziyi Chen, Jian Cong, Lelai Deng, Chuang Ding, Lu Gao, Mingqing Gong, Peisong Huang, Qingqing Huang, Zhiying Huang, Yuanyuan Huo, Dongya Jia, Chumin Li, Feiya Li, Hui Li, Jiaxin Li, Xiaoyang Li, Xingxing Li, Lin Liu, Shouda Liu, Sichao Liu, Xudong Liu, Yuchen Liu, Zhengxi Liu, Lu Lu, Junjie Pan, Xin Wang, Yuping Wang, Yuxuan Wang, Zhen Wei, Jian Wu, Chao Yao, Yifeng Yang, Yuanhao Yi, Junteng Zhang, Qidi Zhang, Shuo Zhang, Wenjie Zhang, Yang Zhang, Zilin Zhao, Dejian Zhong, and Xiaobin Zhuang, “Seed-TTS: A family of high-quality versatile speech generation models,” 2024.
- [24] Sefik Emre Eskimez, Xiaofei Wang, Manthan Thakker, Canrun Li, Chung-Hsien Tsai, Zhen Xiao, Hemin Yang, Zirun Zhu, Min Tang, Xu Tan, et al., “E2 TTS: Embarrassingly easy fully non-autoregressive zero-shot TTS,” in *2024 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2024, pp. 682–689.
- [25] Haorui He, Zengqiang Shang, Chaoren Wang, Xuyuan Li, Yicheng Gu, Hua Hua, Liwei Liu, Chen Yang, Jiaqi Li, Peiyang Shi, et al., “Emilia: An extensive, multilingual, and diverse speech dataset for large-scale speech generation,” in *2024 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2024, pp. 885–890.
- [26] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, “LibriSpeech: an asr corpus based on public domain audio books,” in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.