Interspeech 2024
1-5 September 2024, Kos, Greece

Should you use a probabilistic duration model in TTS? Probably! Especially for spontaneous speech

Shivam Mehta[1], *Harm Lameris*[1], *Rajiv Punmiya*[2], *Jonas Beskow*[1], *Éva Székely*[1], *Gustav Eje Henter*[1]

[1]Speech, Music and Hearing, KTH Royal Institute of Technology, Sweden [2]Independent researcher

smehta@kth.se, lameris@kth.se, rajiv.punmiya@gmail.com, beskow@kth.se, szekely@kth.se, ghe@kth.se

Abstract

Converting input symbols to output audio in TTS requires modelling the durations of speech sounds. Leading non-autoregressive (NAR) TTS models treat duration modelling as a regression problem. The same utterance is then spoken with identical timings every time, unlike when a human speaks. Probabilistic models of duration have been proposed, but there is mixed evidence of their benefits. However, prior studies generally only consider speech read aloud, and ignore spontaneous speech, despite the latter being both a more common and a more variable mode of speaking. We compare the effect of conventional deterministic duration modelling to durations sampled from a powerful probabilistic model based on conditional flow matching (OT-CFM), in three different NAR TTS approaches: regression-based, deep generative, and end-to-end. Across four different corpora, stochastic duration modelling improves probabilistic NAR TTS approaches, especially for spontaneous speech.

Index Terms: Speech synthesis, probabilistic models, duration modelling, spontaneous speech, conditional flow matching

1. Introduction

A key challenge of text-to-speech is upsampling discrete text inputs, usually graphemes or phonemes, into continuous-valued acoustic outputs, often in the form of mel-spectrograms. It is of great importance to accurately model speech-sound durations in this upsampling process, particularly for the prosody of the speech. Traditionally, autoregressive (AR) neural TTS models infer these durations implicitly within their generative process, whilst non-autoregressive (NAR) TTS models often require an explicit model to create these durations. Such duration models can adopt either a deterministic regression-based approach, producing the same output for constant input, or a stochastic framework, learning a probability distribution and generating different samples from that distribution. Despite a foundation of theoretical [1] and experimental evidence [2] highlighting that only probabilistic synthesis methods can appear perfectly natural, it is only recently that advances in probabilistic modelling have demonstrated standout results in synthesising human behaviour such as motion [3, 4, 5] and speech [6, 7, 8, 9, 10, 11, 12].

Despite the apparent advantages of stochastic approaches in various domains of synthetic content generation, the adoption of probabilistic duration modelling in NAR TTS remains limited. At present, a majority of widely used NAR TTS models employ regression-based, deterministic duration modelling. This includes not only regression-based approaches like [13, 14] but also prominent examples of the latest technological advancements like diffusion models [7] and flow-matching-based models [15, 16, 6]. The end-to-end TTS model VITS [9, 17] is an exception that uses stochastic duration modelling.

This reluctance to employ stochastic duration modelling can be partly attributed to mixed empirical evidence regarding its efficacy. Some studies [9, 17] find stochastic duration modelling to improve the naturalness of synthesised speech, whilst more recent ones challenge its effectiveness [15]. Moreover, evaluations comparing deterministic and stochastic approaches often limit their focus to read-aloud speech corpora like LJ Speech [18]. This leaves open the question of how durations are to be modelled in spontaneous speech, in light of its highly diverse prosodic structure [19] and that it constitutes the most common form of human speech communication.

In this paper, we perform a comprehensive comparison between conventional regression-based duration modelling and durations sampled from a powerful probabilistic duration model based on flow matching. We perform this comparison across a variety of NAR TTS architectures, specifically a deterministic acoustic model (FastSpeech 2 [14]), an advanced deep generative acoustic model (Matcha-TTS [6]), and a probabilistic end-to-end TTS model (VITS [9]). For each architecture, deterministic and stochastic duration modelling is evaluated both objectively and through subjective listening tests on a total of four different speech corpora: two comprising read-aloud speech and two containing spontaneous speech. Our key findings are:

- Regression-based TTS approaches do not benefit from stochastic duration modelling. In contrast, the probabilistic TTS approaches had equal or improved performance.
- The differences between deterministic and probabilistic duration modelling are most evident in spontaneous speech corpora, and least apparent in the widely used LJ Speech corpus. This highlights the need for better benchmarks of how modern TTS systems handle the complexities of natural speech.

Our findings indicate that stochastic modelling of speech-sound durations can improve NAR TTS, and that flow-matching models introduce negligible overhead in terms of parameter count and synthesis speed in this regard. For audio and resources see `https://shivammehta25.github.io/prob_dur/`.

2. Background

2.1. Duration modelling in TTS

Contemporary TTS models are generally classified as either autoregressive or non-autoregressive. Autoregressive models generate output sequentially, using either neural attention mechanisms (e.g., [20, 21]) or transducers (e.g., [22, 23]) to upsample input symbols or states to output frames as they go along. Non-autoregressive models, in contrast, generate all output values in parallel. This can be faster, especially on GPUs. These models typically upsample the input text vectors using an explicit dura-

10.21437/Interspeech.2024-1582

tion model, after which the vectors are transformed into acoustic features. The duration models are trained on reference durations obtained either through external forced alignment [14, 13], or via Monotonic Alignment Search (MAS) [24, 7, 6, 16, 25], or determined through Gaussian upsampling [26, 8]. All these systems employ a regression-based duration predictor trained to typically minimise the log-domain mean square error (MSE) between predicted and reference durations. All the cited works primarily perform experiments on read-aloud speech.

Notably, there are existing studies on read speech [27] demonstrating that stochastic prosody modelling contributes to addressing the issue of oversmoothness in synthesis, particularly for styles characterised by high F0 [28]. However, these investigations did not isolate the influence of the duration modelling from other conditioning inputs like F0, nor did they examine its effects on highly conversational, spontaneous speech. Moreover, they often rely on generative modelling paradigms with restricted expressiveness, that require numerous iterations or neural network calls to produce good quality samples, which introduces significant computational overhead. Specific examples of this are VITS [9], which uses discrete-time normalising flows conditioned on speaker embeddings to generate durations, and VITS 2 [17], which wraps the aforementioned duration predictor with a discriminator to produce more realistic duration values. However, they only train the duration model after the acoustic model, as opposed to jointly with the rest of the model, for reasons of training stability.

Recent studies [15] have hinted that regression-based duration predictors underestimate the standard deviation of phoneme and silence durations, producing samples with less duration diversity and more regular durations. This could potentially be desirable for synthesising read-aloud speech, but for realistic conversational synthesis we need to recreate the variability and irregularity in phoneme and silence durations.

Recently, a new class of probabilistic generative modelling known as conditional flow matching [29] (specifically OT-CFM and the closely related rectified flows [30]) have emerged in the text-to-speech domain, delivering fast and state-of-the-art results in acoustic modelling [15, 6, 16, 25]. Flow matching is a continuous-time variant of normalising flows that, unlike discrete-time normalising flows [31], can be trained without ODE solvers and does not require limiting the architectural design to ensure bijectivity, improving training speed and model flexibility. The specific design of OT-CFM means that very few network evaluations are needed at synthesis time, making it much faster than diffusion models [29]. Despite this, only [15] among the cited works investigates flow matching for TTS duration modelling, finding no notable naturalness improvement.

2.2. Spontaneous speech

Spontaneous speech is the most common form of speech humans produce and comprehend [32], yet it has considerably different characteristics to the read-aloud speech that state-of-the-art TTS models are generally trained on. It offers interesting challenges for TTS, as it has more diversity in its F0 and speech rate compared to read speech, even if the read speech has a conversational style [19]. Additionally, it leverages phenomena not found in scripted speech, such as disfluencies, filled pauses, and breaths, all essential tools in speech planning and beneficial to information recall [33]. There is also considerably more variability in spontaneous speech depending on the communicative context, with prosodic realisation playing a role in the interpretation of the pragmatic implication of a phrase [34].

With the increased focus on communicative competencies in spoken interaction for agents, spontaneous speech offers a more realistic setting for the evaluation of TTS architectures [35]. However, despite its potential advantages, spontaneous speech data has not been commonly used in TTS systems, partly due to its expensive transcription process. The lower variability of read-aloud speech also benefited concatenative TTS systems by making differences across concatenation points smaller. As a consequence, there is little knowledge regarding what techniques that are appropriate for synthesising spontaneous speech. In particular, we believe our work is the first to carefully study duration modelling in spontaneous speech for NAR neural TTS.

3. Method

3.1. Data

We used two read-speech and two spontaneous-speech corpora for our study of duration modelling. 100 utterances of each corpus were withheld for validation.

The first read-speech corpus, LJ Speech (herein labelled **LJ**) [18], is a public-domain dataset that consists of a single female speaker of General American English reading passages from non-fiction books for a duration of approximately 24 h. The second read-speech corpus, RyanSpeech **(RS)** [36], is a scripted conversational corpus of 9 h of a male speaker of General American English reading texts from chatbots and dialogue systems, as well as LibriVox transcriptions. Whilst purportedly conversational, RyanSpeech does not exhibit any spontaneous speech behaviours, as the performed dialogues are all scripted.

The experiments considered two spontaneous-speech corpora. One was the Trinity Speech-Gesture Dataset II (**TSGD2**)[1] [37], a 6 h spontaneous conversational corpus of time-aligned speech and marker-based motion capture of a male Hiberno-English speaking actor. It features 25 takes of the actor speaking in a conversational style without feedback or interruptions. The speech data was extracted and segmented into breath groups of 1–10 seconds, which were transcribed using Automatic Speech Recognition (ASR) before manual correction, based on [38]. The transcripts include fillers and repetitions, along with tokens like semicolons for breaths and commas for silent pauses, to be able to elicit spontaneous behaviours at synthesis time.

Our other spontaneous corpus, AptSpeech (**AptS**), contains the speech data from the publicly available multimodal multiparty recordings from [39]. The corpus consists of 15 interactions between a single mediator and varying participants who were tasked with designing a living space (apartment) on a large touchscreen GUI. We extracted the unscripted speech data from the mediator, a male speaker of General American English, which totalled 5.7 h. Transcription and segmentation were performed identically to the first spontaneous corpus.

3.2. Duration models considered

Most contemporary NAR TTS approaches are encoder-decoder frameworks where duration modelling occupies a consistent position in the synthesis pipeline, illustrated in Figure 1. Specifically, the encoder generates an intermediate representation from the input symbols, which is fed into a duration-predictor module. This module estimates the temporal length (i.e., the number of frames) of each unit in the input sequence. The duration model is trained to predict the reference duration of each input symbol using a MSE regression loss in the log domain [40].

[1] https://trinityspeechgesture.scss.tcd.ie/

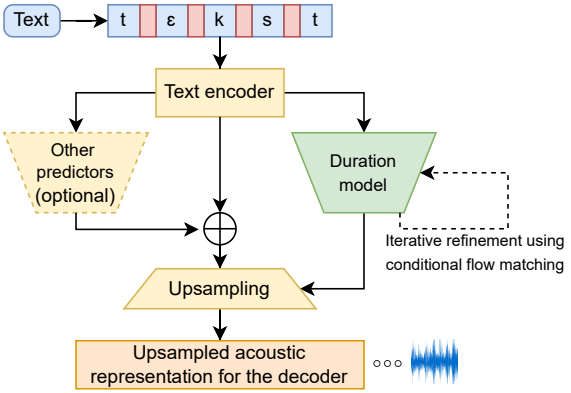


Figure 1: *Overview of NAR TTS synthesis. We replace duration prediction with flow-matching-based duration modelling.*

It thus estimates the expected log-duration of each unit, conditioned on the text-encoder output sequence. After converting durations to integers each text-encoder output vector is repeated (upsampled) as many times as the duration predictor indicates.

We explore the effects of replacing the MSE-based duration predictor in existing NAR TTS approaches with a log-domain duration model based on conditional flow matching, specifically OT-CFM [29] with $\sigma{=}10^{-4}$ as in [6]. This learns a probability distribution by learning to predict log-domain reference durations (no dequantisation) from noisy versions of the same, and from the text-encoder output. Synthesis iteratively transforms a sequence of Gaussian noise values into output durations, then converts to the linear domain and rounds to the nearest integer.

4. Experiments

For our experiments, we selected three strong and widely-recognised NAR TTS approaches: FastSpeech 2 (**FS2**), a deterministic acoustic model; Matcha-TTS (**Matcha**), a probabilistic acoustic model; and VITS (**VITS**), an end-to-end probabilistic TTS model. For each approach, we studied the effect of replacing its conventional deterministic duration predictor (**DET**) with the OT-CFM-based duration model (**FM**) described in Sec. 3.2. Whilst Matcha-TTS and VITS can learn alignments jointly with learning to speak, FastSpeech 2 was supplied with precomputed reference alignments from Matcha during its training phase. We trained each of these architectures with each duration-model type (DET and FM) on the four different corpora from Sec. 3.1. for 500k updates. This resulted in a total of $3{\times}2{\times}4{=}24$ distinct systems trained. Each system was trained on a single NVIDIA-3090 GPU using batch size 32.

All systems used Phonemizer[2] with `espeak-ng` to convert input graphemes to IPA phones. Following [24, 9, 6], we consistently interleaved each phone with a blank token, so as to represent each phone by two encoder vectors. This improves acoustic-model granularity. We used the default hyperparameters for each architecture as specified in their original publications, with the sole deviation being the incorporation of an additional convolution-based causal post-net for FS2, a feature prevalent in most open-source implementations of FastSpeech 2.[3] Notably, all approaches use the same network architecture for the duration predictor, consisting of a layer norm sandwiched between two convolution layers followed by a projection layer, cumulatively amounting to approximately 400k parameters.

[2] `https://github.com/bootphon/phonemizer/`
[3] `https://github.com/facebookresearch/fairseq/`

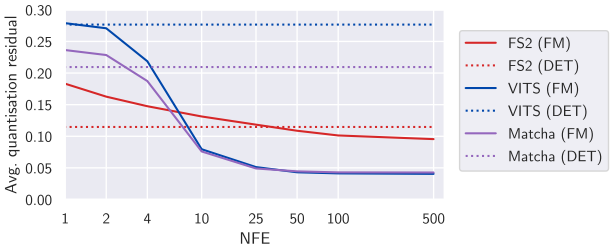


Figure 2: *Duration quantisation residual, averaged over four corpora for FS2, VITS, and Matcha-TTS vs. the number of function evaluations for FM. DET has constant residual.*

For the FM duration models, we maintained the architecture of the original, deterministic duration predictor unchanged but incorporated noisy durations and an embedding of the current iteration step as extra inputs, the latter using the time-step embedding architecture from the acoustic decoder of [6]. This only added another 100k parameters, equating to 0.6% or less of model size. We empirically tuned the noise temperature to 0.667 during synthesis, finding that VITS (but not Matcha) output degraded above this value. We opted to use 10 Neural Function Evaluations (NFEs) during synthesis, which resulted in a negligible increase in of 0.001 in Real Time Factor (RTF) over DET. Figure 2 graphs the average magnitude of the residual when quantising the duration-model output to the nearest integer as a function of on the NFE, indicating how accurately the synthesis process is able to generate integer outputs.

4.1. Stimuli and objective evaluation

To ensure that all stimuli are understandable as standalone utterances and their content is appropriate for online listening tests – which is not true for most corpora – we generated 100 new test sentences per dataset in the same style as the original corpus (to avoid domain mismatch). This was done by putting 100 sentences from each corpus into GPT-4 [41], with instructions to mimic the speaking style and breaths and pauses for the spontaneous corpora, and create another 100 sentences related to everyday topics like visiting the zoo, going to a shopping centre, and going to school. All prompts and sentences are provided on our webpage. Five random realisations per sentence were synthesised for the stochastic duration models, with objective scores being calculated as an average across all of these, whilst the subjective evaluation used one realisation per sentence.

For the objective evaluation, we calculated the word error rate (WER) of automatic speech recognition on synthetic stimuli, and also performed automatic MOS prediction to estimate TTS quality. WERs were obtained using Whisper `medium.en` [42], as the WER of contemporary ASR correlates well with speech intelligibility to human listeners [43]. MOS prediction used the off-the-shelf AutoMOS system described in [44].

4.2. Subjective evaluation

For the subjective evaluation, we conducted four Comparative Mean Opinion Score (CMOS)-style web-based listening tests, with each test including stimuli from all models, but only one specific corpus. In these tests, listeners were asked to "choose how natural these two versions sound in comparison" on a 7-point integer Likert scale ranging from "ver 1 much better" to "ver 2 much better"; zero meant no difference. Each audio stimulus pair compared the same sentence synthesised using the same architecture (FS2, VITS, or Matcha), the only difference being the use of either DET or FM for durations. We recruited

Table 1: *ASR WER, AutoMOS scores, and CMOS values. Positive CMOS means FM is preferred over DET. Significant CMOS differences favouring FM are bold, ones favouring DET italic.*

Model	Dataset	WER% (↓)		AutoMOS (↑)		CMOS
		DET	FM	DET	FM	95% conf. int.
FS2	LJ	**2.07**	2.27	**4.35**	4.29	*−0.25±0.15*
	RS	2.89	**2.27**	**4.35**	4.29	*−0.49±0.13*
	TSGD2	**17.24**	23.52	**2.65**	2.44	*−0.59±0.10*
	AptS	**27.61**	34.17	**3.15**	2.88	*−1.66±0.11*
VITS	LJ	2.67	**2.40**	4.31	**4.51**	**0.23±0.17**
	RS	2.46	**2.31**	4.52	**4.70**	**0.47±0.16**
	TSGD2	13.27	**9.26**	3.38	**3.94**	**0.48±0.15**
	AptS	10.62	**8.31**	4.07	**4.37**	**0.69±0.14**
Matcha	LJ	2.39	**1.53**	4.40	**4.53**	0.02±0.14
	RS	1.94	**1.70**	4.46	**4.56**	−0.04±0.16
	TSGD2	9.00	**6.15**	3.12	**3.56**	**0.47±0.15**
	AptS	11.50	**5.11**	3.55	**4.15**	**0.69±0.14**

160 self-reported native English speakers via Prolific, exactly 40 for each test. Each listener was paid 3 GBP after test completion (median duration 15 min). An additional 7 listeners were rejected for failing two or more of our three attention checks.

To familiarise listeners with the speaker and speaking style of the corpus used in the test, they first listened five natural utterances from the test set, before proceeding with the CMOS test. Each CMOS test for a specific listener comprised 10 stimulus pairs per architecture (30 pairs per person). Ultimately 4 sets of 30 stimulus pairs per corpus were evaluated, for an overall total of 400 ratings for each of the 12 DET-FM system pairings.

5. Results

Results of the objective and subjective experiments are reported in Table 1. All CMOS results except LJ and RS for Matcha were significantly different from zero ($p < 0.05$). We find that FS2, the regression-based TTS model, produced worse speech in all aspects when changing from the deterministic to the stochastic duration model. Conversely, VITS (an end-to-end architecture based on discrete-time normalising flows) demonstrated notable improvements with the stochastic duration predictor. This enhancement was observed for both read and spontaneous speech, aligning with the findings in the original VITS paper [9]. Matcha-TTS, being architecturally flexible with a strong probabilistic foundation, synthesised read speech similarly well using both DET and FM (consistent with prior results on read speech in [15]), but showed significant improvement for spontaneous speech with stochastic duration generation.

It is interesting to compare the numerical results to the quantisation residuals in Figure 2. These residuals indicate that the stochastic FM models successfully learnt to produce near-integer outputs during sampling, but that 10 or more NFEs (i.e., Euler-forward ODE-solver steps) are required to recover this property during sampling; this validates our choice of NFE=10 for the experiments. The DET models only require a single NFE to compute their output. As they are trained to predict average values, which typically are not integers, it makes sense that they have greater quantisation residuals (dashed lines). The one exception is FS2, where FM duration modelling converges much more slowly and to a higher residual value. Although the graph only shows averages, the same shapes and trends hold for all individual corpora. This suggests that OT-CFM struggled to learn accurate duration distributions inside the FS2 architecture, making it perform worse than DET there. Separately, the WER and AutoMOS results suggest that FS2 typically achieved worse intelligibility and speech quality than corresponding VITS and Matcha systems. FS2 FM might improve by using a synthesis temperature near zero, but this essentially makes output deterministic like DET, defeating the purpose of using flow matching.

Our results indicate that, although deterministic acoustic models may not benefit from stochastic duration modelling, probabilistic TTS approaches often improved and were never adversely affected (some cases showed no statistically significant difference). This reinforces the potential benefits of advanced probabilistic duration modelling, especially using low-overhead flow-matching techniques. Further, the disparity between deterministic and probabilistic modelling paradigms was most pronounced on spontaneous speech corpora, underscoring the critical role of accurate duration modelling (and potentially broader prosody modelling) for this diverse and irregular speech type. This finding, combined with the observed advantages in models like VITS and Matcha-TTS, suggest that spontaneous speech and probabilistic models are promising direction for future TTS research focused on enhancing the naturalness and expressiveness of synthesised speech.

Interestingly, LJ Speech, the most widely used TTS corpus, consistently exhibited the smallest CMOS difference, highlighting its limitations as a benchmark for duration modelling, and possibly other prosodic features as well. This strongly suggests a need for more varied and rigorous benchmarks and corpora to accurately evaluate the performance of advanced TTS methods.

6. Conclusion

In this study, we explored the impact of deterministic versus probabilistic duration modelling within the framework of non-autoregressive TTS, across three categories of such TTS architectures: regression-based, deep generative, and end-to-end. Crucially, our study included not only read-aloud but also spontaneous speech, for which duration modelling has hitherto been severely underexplored. We discovered significant naturalness improvements from probabilistic duration modelling over conventional deterministic duration prediction on spontaneous speech synthesised by probabilistic TTS paradigms. Objective metrics likewise improved. Our collective findings highlight the advantages of probabilistic models of durations and acoustics in achieving lifelike and expressive speech synthesis.

Additionally, our analysis suggests that the LJ Speech corpus may not be the most appropriate benchmark for evaluating the nuances of duration and prosody modelling, due to its limited representation of the variability and complexities inherent in natural speech. Spontaneous speech corpora may offer more interesting and relevant material for benchmarking TTS.

7. Acknowledgements

Research funded by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation, Swedish Research Council proj. VR-2020-02396, and the Industrial Strategic Technology Development Program (grant no. 20023495) funded by MOTIE, Korea.

8. References

[1] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *J. Mach. Learn. Res.*, vol. 13, no. 25, pp. 723–773, 2012.

[2] G. E. Henter, T. Merritt, M. Shannon, C. Mayo, and S. King, "Measuring the perceptual effects of modelling assumptions in speech synthesis using stimuli constructed from repeated natural speech," in *Proc. Interspech*, 2014, pp. 1504–1508.

[3] S. Alexanderson, R. Nagy, J. Beskow, and G. E. Henter, "Listen, denoise, action! Audio-driven motion synthesis with diffusion models," *ACM Trans. Graph.*, vol. 42, no. 4, 2023, art. no. 44.

[4] S. Mehta, S. Wang, S. Alexanderson, J. Beskow, É. Székely, and G. E. Henter, "Diff-TTSG: Denoising probabilistic integrated speech and gesture synthesis," in *Proc. SSW*, 2023.

[5] S. Mehta, R. Tu, S. Alexanderson, J. Beskow, Éva Székely, and G. E. Henter, "Unified speech and gesture synthesis using flow matching," in *Proc. ICASSP*, 2024, pp. 8220–8224.

[6] S. Mehta, R. Tu, J. Beskow, É. Székely, and G. E. Henter, "Matcha-TTS: A fast TTS architecture with conditional flow matching," in *Proc. ICASSP*, 2024, pp. 11 341–11 345.

[7] V. Popov, I. Vovk, V. Gogoryan, T. Sadekova, and M. Kudinov, "Grad-TTS: A diffusion probabilistic model for text-to-speech," in *Proc. ICML*, 2021, pp. 8599–8608.

[8] N. Chen, Y. Zhang, H. Zen, R. J. Weiss, M. Norouzi, N. Dehak, and W. Chan, "WaveGrad 2: Iterative refinement for text-to-speech synthesis," in *Proc. Interspeech*, 2021, pp. 3765–3769.

[9] J. Kim, J. Kong, and J. Son, "VITS: Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," in *Proc. ICML*, 2021, pp. 5530–5540.

[10] N. Chen, Y. Zhang, H. Zen, R. J. Weiss, M. Norouzi, and W. Chan, "WaveGrad: Estimating gradients for waveform generation," in *Proc. ICLR*, 2021.

[11] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro, "DiffWave: A versatile diffusion model for audio synthesis," in *Proc. ICLR*, 2021.

[12] V. Popov, I. Vovk, V. Gogoryan, T. Sadekova, M. S. Kudinov, and J. Wei, "Diffusion-based voice conversion with fast maximum likelihood sampling scheme," in *Proc. ICLR*, 2021.

[13] A. Łańcucki, "FastPitch: Parallel text-to-speech with pitch prediction," in *Proc. ICASSP*, 2021, pp. 6588–6592.

[14] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "FastSpeech 2: Fast and high-quality end-to-end text to speech," in *Proc. ICLR*, 2021.

[15] M. Le, A. Vyas, B. Shi, B. Karrer, L. Sari, R. Moritz, M. Williamson, V. Manohar, Y. Adi, J. Mahadeokar *et al.*, "Voicebox: Text-guided multilingual universal speech generation at scale," in *Proc. NeurIPS*, 2023.

[16] Y. Guo, C. Du, Z. Ma, X. Chen, and K. Yu, "VoiceFlow: Efficient text-to-speech with rectified flow matching," in *Proc. ICASSP*, 2024, pp. 11 121–11 125.

[17] J. Kong, J. Park, B. Kim, J. Kim, D. Kong, and S. Kim, "VITS2: Improving quality and efficiency of single-stage text-to-speech with adversarial learning and architecture design," in *Proc. Interspeech*, 2023, pp. 4374–4378.

[18] K. Ito and L. Johnson, "The LJ Speech dataset," `https://keithito.com/LJ-Speech-Dataset/`, 2017.

[19] H. Lameris, S. Mehta, G. E. Henter, J. Gustafson, and É. Székely, "Prosody-controllable spontaneous TTS with neural HMMs," in *Proc. ICASSP*, 2023.

[20] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen *et al.*, "Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions," in *Proc. ICASSP*, 2018, pp. 4779–4783.

[21] N. Li, S. Liu, Y. Liu, S. Zhao, and M. Liu, "Neural speech synthesis with transformer network," in *Proc. AAAI*, vol. 33, no. 01, 2019, pp. 6706–6713.

[22] S. Mehta, É. Székely, J. Beskow, and G. E. Henter, "Neural HMMs are all you need (for high-quality attention-free TTS)," in *Proc. ICASSP*, 2022, pp. 7457–7461.

[23] S. Mehta, A. Kirkland, H. Lameris, J. Beskow, É. Székely, and G. E. Henter, "OverFlow: Putting flows on top of neural transducers for better TTS," in *Proc. Interspeech*, 2023, pp. 4279–4283.

[24] J. Kim, S. Kim, J. Kong, and S. Yoon, "Glow-TTS: A generative flow for text-to-speech via monotonic alignment search," in *Proc. NeurIPS*, 2020, pp. 8067–8077.

[25] S. Kim, K. Shih, J. F. Santos, E. Bakhturina, M. Desta, R. Valle, S. Yoon, *et al.*, "P-Flow: A fast and data-efficient zero-shot TTS through speech prompting," in *Proc. NeurIPS*, 2023.

[26] J. Shen, Y. Jia, M. Chrzanowski, Y. Zhang, I. Elias, H. Zen, and Y. Wu, "Non-attentive Tacotron: Robust and controllable neural TTS synthesis including unsupervised duration modeling," *arXiv preprint arXiv:2010.04301*, 2020.

[27] S. Ogun, V. Colotte, and E. Vincent, "Stochastic pitch prediction improves the diversity and naturalness of speech in Glow-TTS," in *Proc. Interspeech*, 2023.

[28] G. Zhang, T. Merritt, S. Ribeiro, B. T. Vecino, K. Yanagisawa, K. Pokora *et al.*, "Comparing normalizing flows and diffusion models for prosody and acoustic modelling in text-to-speech," in *Proc. Interspeech*, 2023.

[29] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le, "Flow matching for generative modeling," in *Proc. ICLR*, 2023.

[30] X. Liu, C. Gong *et al.*, "Flow straight and fast: Learning to generate and transfer data with rectified flow," in *Proc. ICLR*, 2022.

[31] D. P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1x1 convolutions," in *Proc. NeurIPS*, 2018, pp. 10 236–10 245.

[32] E. Shriberg, "Spontaneous speech: How people really talk and why engineers should care," in *Proc. Eurospeech*, 2005.

[33] É. Székely, G. E. Henter, J. Beskow, and J. Gustafson, "Breathing and speech planning in spontaneous speech synthesis," in *Proc. ICASSP*, 2020, pp. 7649–7653.

[34] S. Herment and L. Leonarduzzi, "The pragmatic functions of prosody in English cleft sentences," in *Proc. Speech Prosody*, 2012.

[35] M. Marge, C. Espy-Wilson, N. G. Ward, A. Alwan, Y. Artzi, M. Bansal *et al.*, "Spoken language interaction with robots: Recommendations for future research," *Comput. Speech Lang.*, vol. 71, p. 101255, 2022.

[36] R. Zandie, M. H. Mahoor, J. Madsen, and E. S. Emamian, "RyanSpeech: A corpus for conversational text-to-speech synthesis," in *Proc. Interspeech*, 2021, pp. 2751–2755.

[37] Y. Ferstl, M. Neff, and R. McDonnell, "ExpressGesture: Expressive gesture generation from speech through database matching," *Comput. Animat. Virt. W.*, p. e2016, 2021.

[38] É. Székely, G. E. Henter, and J. Gustafson, "Casting to corpus: Segmenting and selecting spontaneous dialogue for TTS with a CNN-LSTM speaker-dependent breath detector," in *Proc. ICASSP*, 2019, pp. 6925–6929.

[39] D. Kontogiorgos, V. Avramova, S. Alexanderson, P. Jonell, C. Oertel, J. Beskow *et al.*, "A multimodal corpus for mutual gaze and joint attention in multiparty situated interaction," in *Proc. LREC*, 2018, pp. 119–127.

[40] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "FastSpeech: Fast, robust and controllable text to speech," in *Proc. NeurIPS*, 2019.

[41] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman *et al.*, "GPT-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[42] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *Proc. ICML*, 2023, pp. 28 492–28 518.

[43] J. Taylor and K. Richmond, "Confidence intervals for ASR-based TTS evaluation," in *Proc. Interspeech*, 2021, pp. 2791–2795.

[44] E. Cooper, W.-C. Huang, T. Toda, and J. Yamagishi, "Generalization ability of MOS prediction networks," in *Proc. ICASSP*, 2022, pp. 8442–8446.