Text-to-Speech for Automatic Speech Recognition

Christoph Minixhofer

Text-to-Speech (TTS) data generation has advanced to the point where it often mimics real data convincingly, but how effective is TTS for training automatic speech recognition (ASR) systems? Despite quality improvements, a significant performance gap persists when using TTS for ASR training, even in controlled settings. This thesis explores this gap, showing that while recent non-autoregressive and probabilistic methods can reduce it, the gap remains. By investigating architectures and scaling laws of TTS-for-ASR and by evaluating TTS data along factors like prosody, speaker diversity, and acoustic environment—rather than relying on naturalness ratings—we identify key distinctions between TTS and real data. These findings are leveraged to enhance training in low-data domains, improve efficiency, and select optimal TTS data subsets.

August 26, 2024

Contents

1. Introduction

Text-to-Speech (TTS) systems have rapidly advanced in recent years, leading to synthetic output often being indistinguishable from real human speech according to human evaluators. However, despite these advancements, a critical question remains: How effective is TTS-generated speech for training automatic speech recognition (ASR) systems? This thesis addresses this question by exploring the persistent performance gap between TTS-generated data and real speech when used for ASR training.

While TTS systems have improved in producing natural-sounding speech, the effectiveness of this synthetic speech for ASR remains suboptimal. Previous studies have demonstrated that, even in controlled experimental settings, ASR systems trained on TTS data do not perform as well as those trained on real speech. This performance gap poses a significant challenge, particularly as TTS is increasingly used as a solution for generating large-scale, diverse datasets required for training robust ASR systems, especially in low-resource domains where real data is scarce.

In this work, we investigate the underlying reasons for this gap. While, our analysis reveals that while recent probabilistic methods can reduce the gap, it does not close entirely. To better understand the limitations of TTS-generated data, we propose a holistic approach that evaluates synthetic speech datasets across multiple factors, such as prosody, speaker diversity, and acoustic environment, rather than relying solely on naturalness ratings typically used in human evaluations.

By considering these factors as distributions, rather than isolated characteristics, we identify key differences between TTS-generated and real speech that contribute to the performance gap. Moreover, we demonstrate that the factors influencing human perception of naturalness do not always align with those that determine the utility of TTS data for ASR training. This discrepancy highlights the need for new evaluation metrics and methods tailored specifically to the demands of TTS-for-ASR.

In the final stage of this PhD, we aim to leverage these insights to enhance the effectiveness of TTS-generated speech for ASR. We will focus on optimizing the generation and selection of synthetic speech data in low-resource domains, predicting which TTS systems and configurations produce the most effective training data, and improving training efficiency.

1.1. Notation

In machine learning, it is convention to denote X as model input and Y as model output, with a model f with parameters θ predicting Y using X, such that $Y = f_\theta(X)$. However, in the context of speech synthesis and recognition in the same context, this can be ambiguous, as X and Y can either represent speech signals or transcripts. To maintain clarity throughout this work, we will denote speech signals as S and transcripts as T.

When both synthetic and real data are addressed in the same context, we use $\sim$ to denote the synthetic counterpart, for example $\tilde{S}$ for a set of speech signals or $\tilde{\theta}$ for model parameters derived from synthetic speech.

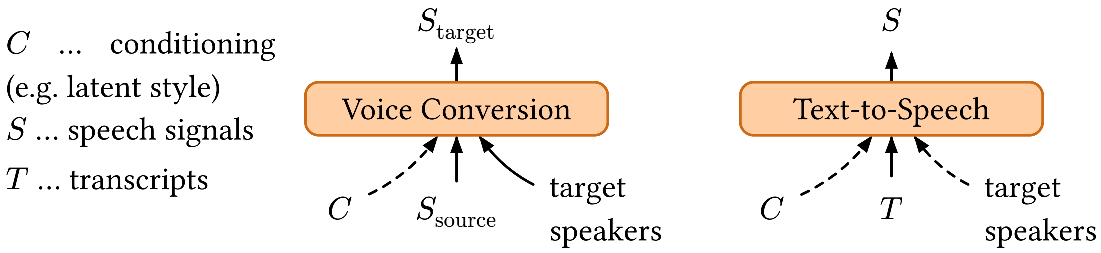


Figure 1: Text-to-Speech (TTS) compared to Voice Conversion (VC).

1.2. Synthetic Speech for Automatic Speech Recognition

Using synthetic data to train recognition systems has its origins in computer vision (Butler et al. 2012, Dosovitskiy et al. 2015, Handa et al. 2015, Mayer et al. 2015). For example, 3D scenes are rendered as static images to train models for image segmentation or for creating depth maps, or entire environments are simulated for autonomous driving (Nikolenko 2021). In computer vision, there are algorithms and programs which can simulate 3D environments, enabled by a simulation of the underlying physics, which at this point, we understand quite well (Nikolenko 2021). Speech, driven not only by the biomechanics of the vocal tract but also by neurobiological processes in the brain, is a lot harder to simulate this way. The field of *Articulatory Speech Synthesis*, which aims to generate speech by modeling the shape and movement of the vocal tract, has yet to achieve results that are comparable to natural speech. Due to this, articulatory speech synthesis is mainly used in speech and language research rather than in practical speech synthesis applications (Kröger 2023). It also has not been used to train ASR systems.

Therefore the only practical ways to generate synthetic speech for ASR are corpus-based. The tow main ways to do so, which are also shown in Figure 1, are as follows:

a) **Text-to-Speech (TTS):** TTS systems map text input T to corresponding speech waveforms S using a model $f_\theta(T) = S$, where θ represents the model parameters trained on large-scale paired text-speech datasets.
b) **Voice Conversion (VC):** VC systems transform a source speaker's speech S_{source} into a target speaker's voice S_{target} using a model $g_\varphi(S_{\text{source}}, \text{target speaker}) = S_{\text{target}}$, where φ represents the model parameters trained on paired or unpaired speech data from multiple speakers.

Both VC (Casanova et al. 2022, Thai et al. 2019) and TTS (Li et al. 2018, Rosenberg et al. 2019, Rossenbach et al. 2020) systems have been successfully used for ASR training. However, TTS systems are more commonly used than VC systems for generating ASR training data as they directly generate speech from text, making it suitable for producing large and diverse datasets. This is helpful for generating the varied data required to train robust ASR systems, with no reference speech needed once training the TTS is complete. Additionally, TTS systems allow for greater control over the generated speech, enabling us to systematically vary factors such as prosody, speaker characteristics, and acoustic conditions, without having to account for these factors in any reference speech. This is particularly valuable for conducting controlled experiments and investigating specific aspects of ASR performance. In contrast, VC is primarily focused on altering existing speech rather than generating new speech from scratch, which limits its utility. For these reasons, this work focuses on TTS as the primary method for generating synthetic speech for ASR systems.

2. TTS-for-ASR

One of our contributions to the field of TTS-for-ASR is the quantification of the performance gap between real and synthetic speech. In this chapter, we examine previous work in the context of the Word Error Rate Ratio (WERR), a metric we introduced in Minixhofer et al. (2023).

2.1. Previous Work

When it comes to improving TTS-for-ASR performance, several approaches have emerged. Providing latent variables for the model is chief among them (Du & Yu 2020, Laptev et al. 2020, Li et al. 2018, Rossenbach et al. 2020, Sun et al. 2020) – the main techniques for this being Global Style Tokens (Wang et al. 2018) and Variational Autoencoders (Kingma 2013). Another approach to improve synthetic speech for ASR training is to increase the diversity of the output at inference time. This is most commonly done through conditioning at inference time, most commonly with speaker representations (Du & Yu 2020, Wang et al. 2020), however aspects of prosody such as duration have also been explored (Casanova et al. 2022, Rossenbach et al. 2023). The synthetic speech can also be made more suitable post-generation using data augmentation, such as adding noise or reverberation (Rossenbach et al. 2020). Different architectures and training paradigms have been explored as well, including non-autoregressive and autoregressive modeling, normalizing flows and denoising diffusion, however we will describe those in detail in Section 3.

The data most commonly used for TTS-for-ASR is read audiobook speech, such as LibriSpeech (Panayotov et al. 2015) and LibriTTS (Zen et al. 2019), and to maintain comparability with previous work, we also use this domain for our work.

2.2. Word Error Rate Ratio

Once synthetic speech $\widetilde{S}$ has been generated it can be used either in conjunction with the real speech S or on its own. The former yields better results (Rossenbach et al. 2020), which is unsurprising, since any additional real data of the target domain should improve an ASR system. However, to assess the suitability of synthetic speech for ASR training, we can conduct controlled experiments by training an ASR system solely on synthetic data and comparing the results to training with real data or synthetic data generated by different systems or with varying parameters.

To make the different works comparable, we introduce the *Word Error Rate Ratio (WERR)*. We evaluate the effectiveness of the learned ASR model parameters $\widetilde{\theta}$ derived from training an ASR system using $\widetilde{S}$ by computing the Word Error Rate on a held-out test set, and denote this as $\mathrm{WER}\left(\widetilde{\theta}\right)$. We repeat this by evaluating θ by training on real data with the same transcripts, resulting in $\mathrm{WER}\left(\theta\right)$. We then define WERR as

$$\mathrm{WERR}\left(\widetilde{\theta}, \theta\right) = \frac{\mathrm{WER}\left(\widetilde{\theta}\right)}{\mathrm{WER}\left(\theta\right)} \tag{1}$$

If $\mathrm{WERR}\left(\widetilde{\theta}, \theta\right) = 1$, we can conclude the synthetic data is functionally identical to the real data for the purpose of ASR training under the experimental conditions. Any value higher than 1 indicates the factor by which real data outperforms the synthetic data.

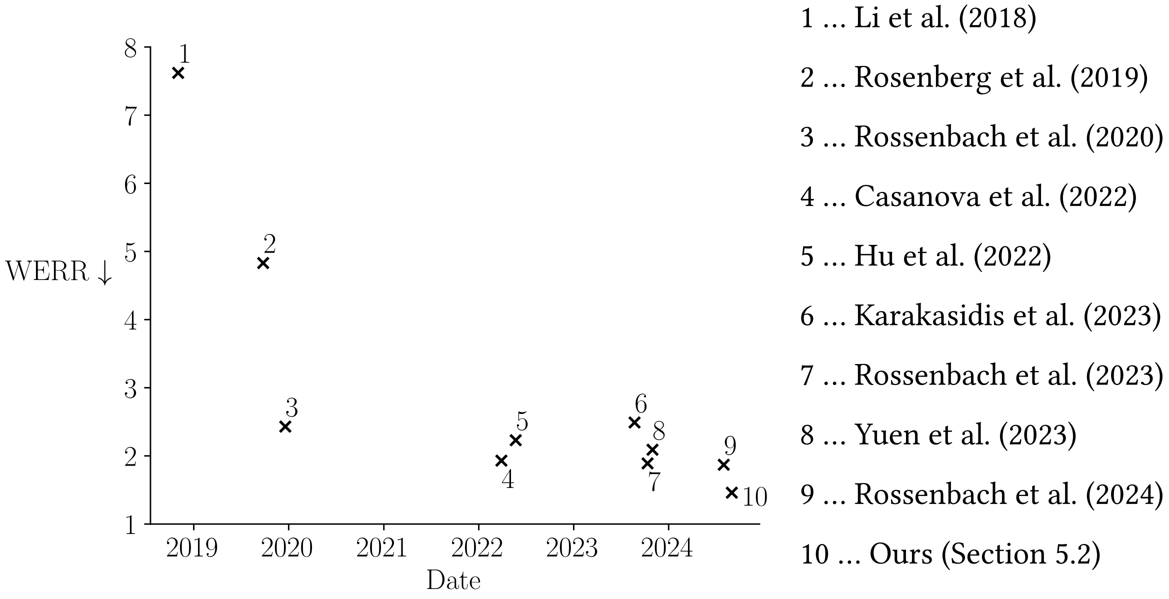


Figure 2: TTS-for-ASR performance in terms of WERR over time.

2.3. The Performance Gap

However, regardless of the modifications and techniques used for TTS-for-ASR, it lags behind synthetic speech naturalness considerably. As is shown in Figure 2, the earliest reported WERR was > 7, with only recent works consistently achieving $\text{WERR} < 2$.

TTS data seems to be much worse for ASR training than for the human listener. To illustrate this, we compute the ratio of Mean Opinion Scores (MOS) equivalently to Equation 1. Using this measure, Tacotron 2 (Shen et al. 2018), which was published a year prior to the earliest system in Figure 2, achieves a factor of ≈ 1.02, still 0.65 lower than the best reported TTS-for-ASR systems' WERR. Since Tacotron 2, TTS has only gotten better, with recent models reporting MOS scores statistically inseparable from real speech MOS scores (Casanova et al. 2024, Chen et al. 2024).

> While TTS samples can achieve high opinion scores from human judges, this does not mean TTS models can completely model the training data distribution. If this were the case, WERR would be 1.

This insight leads to the main questions we seek to answer in this thesis:

1) **Performance Gap**
   a) How much is this gap constrained by factors such as architecture or amount of data?
   b) Can we quantify which factors of TTS speech contribute most to the gap?
2) **TTS-for-ASR Improvements**
   a) Can we specifically target weaker areas of TTS for better TTS-for-ASR performance?
   b) Do improvements gained translate to low-resource applications?

This report contains work relating to 1a), 1b) and 2a), while 2b) is still to be completed.

3. Modeling and Training Approaches for TTS-for-ASR

In this chapter, we investigate different training approaches and data regimes for TTS-for-ASR. As described in Section 2, in previous work, several approaches have been used, but rarely are they analyzed comparatively.

The modeling techniques in this chapter form the basis for our work in Minixhofer et al. (2024a), with the experimental results presented in Section 5.2.

3.1. Autoregressive and Non-Autoregressive Models

The two most common systems used in prior work are Tacotron 2 (Shen et al. 2018) and FastSpeech 2 (Ren et al. 2021). They represent two distinct paradigms: autoregressive and non-autoregressive models, respectively. Both aim to convert text sequences T into corresponding speech waveforms S, but they differ significantly in their architectures and training paradigm.

Tacotron 2 is an autoregressive model, which means it generates each output frame sequentially, conditioned on the previous output. Given an input text sequence $T = \{t_1, t_2, ..., t_N\}$, the corresponding mel-spectrogram sequence $S = \{s_1, s_2, ..., s_K\}$ is generated by modeling the probability distribution as:

$$p(s \mid t) = \prod_{n=1}^{K} p(s_n \mid s_{1:n-1}, t) \tag{2}$$

Here, s_k represents the output at time step k, which depends on all previous outputs $s_{1:k-1}$ and the input text sequence t. This sequential dependency ensures that Tacotron 2 captures temporal correlations in the speech signal. However, the autoregressive nature of Tacotron 2 introduces two main drawbacks:

- **Sequential Dependency:** The generation process is inherently sequential, leading to slow inference times, as each frame s_k must wait for the previous frame s_{k-1} to be generated.
- **Training-Inference Mismatch:** During training, the model is exposed to ground truth previous frames, but at inference, it relies on its own predictions, leading to potential accumulation of errors (Li et al. 2018).

In contrast, FastSpeech 2 uses a non-autoregressive approach, where the entire mel-spectrogram sequence is generated in parallel. The key idea is to remove the sequential dependency by modeling the distribution as:

$$p(s \mid t) = \prod_{n=1}^{K} p(s_n \mid t) \tag{3}$$

In this framework, each output frame s_n is generated independently, conditioned only on the input sequence t. FastSpeech 2 achieves this by leveraging a duration predictor and positional encoding to align the text and speech sequences, allowing for parallel processing and significantly faster inference. The non-autoregressive design of FastSpeech 2 offers several advantages:

- **Parallel Generation**: Since all frames are generated simultaneously, FastSpeech 2 achieves much faster inference times compared to autoregressive models.

- **Robustness:** By removing the dependency on previous outputs, FastSpeech 2 avoids the exposure bias problem, leading to more stable and consistent outputs. This also allows the model to converge with significantly less training data (Pine et al. 2022).

However, the non-autoregressive nature also introduces challenges:

- **Expressiveness:** The lack of temporal dependencies can make it harder for FastSpeech 2 to capture the fine-grained prosody and natural variations in speech, which are more naturally handled by autoregressive models like Tacotron 2.

In TTS-for-ASR, non-autoregressive and autoregressive models have been used to similar extent, however only recently have they been compared directly: Rossenbach et al. (2024) find an autoregressive system outperforms its non-autoregressive counterpart by $\approx 8\%$ (relative) WERR. However, this still does not explain the performance gap outlined in Section 2.3. Due to its robustness across dataset sizes and efficient training and generation, we use a non-autoregressive system in our work.

3.2. Stochastic Speech Synthesis with Denoising Diffusion

In the computer vision domain, which has made use of synthetic data for model training more commonly than in speech (see Section 1.2), recent work has almost exclusively utilized denoising diffusion models (Ho et al. 2020). This has led to progress in using synthetic images for model training, bringing the factor between real and synthetic data down to ≈ 1.3 (Fan et al. 2024), significantly lower than the lowest previously reported WERR values of ≈ 1.7.

Denoising diffusion for TTS can be explained as a sequence of transformations from an initial noise distribution to the desired speech distribution. Let $\boldsymbol{s}_0$ denote the target speech data, and let $\boldsymbol{s}_N$ be a sample from a Gaussian noise distribution, which can be seen as a speech signal with signal-to-noise ratio of $-\infty$ dB. The diffusion process can be described as a sequence of latent variables $\boldsymbol{s}_{n_{n=1}^{T}}$ that iteratively transform $\boldsymbol{s}_N$ into $\boldsymbol{s}_0$. This process is governed by a forward diffusion process $q(\boldsymbol{s}_n|\boldsymbol{s}_{n-1})$ and a reverse denoising process $p_{\theta(\boldsymbol{s}_{n-1}|\boldsymbol{s}_n)}$, parameterized by θ.

The forward diffusion process adds Gaussian noise to the data at each time step n:

$$q(\boldsymbol{s}_n|\boldsymbol{s}_{n-1}) = \mathcal{N}\left(\boldsymbol{s}_n; \sqrt{1-\beta_n}\boldsymbol{s}_{n-1}, \beta_n \boldsymbol{I}\right) \tag{4}$$

where β_n is a variance schedule controlling the amount of noise added at each step. The reverse process, parameterized by the model, aims to denoise the noisy data back to the original speech signal:

$$p_{\theta(\boldsymbol{s}_{n-1}|\boldsymbol{s}_n)} = \mathcal{N}\left(\boldsymbol{s}_{n-1}; \mu_{\theta(\boldsymbol{s}_n,n)}, \Sigma_{\theta(\boldsymbol{s}_n,n)}\right) \tag{5}$$

The training objective is to minimize the evidence lower bound (ELBO) (Kingma 2013), which corresponds to the negative log-likelihood of the data, and can be used to derive the loss:

$$\mathcal{L}_{\text{diffusion}} = \sum_{n=1}^{N} \mathbb{E}_{q(\boldsymbol{s}_n|\boldsymbol{s}_0)}[\| \boldsymbol{s}_0 - \boldsymbol{s}_{n-1} \|^2] \tag{6}$$

4. Evaluating Synthetic Speech: Beyond Naturalness

As we discussed in Section 2.3, the performance gap for TTS-for-ASR cannot solely be explained by architecture or training procedure, and a large remainder of the gap remains unexplained. In this chapter, we look at the gap from the opposite direction, the MOS scores which have been within $\approx 2\%$ of real speech since Shen et al. (2018) and have recently been reported as having reached parity with real speech (Casanova et al. 2024, Chen et al. 2024). The methodology behind MOS scores has been debated and criticized extensively – for example, for the fact that they change significantly when evaluated years apart (Le Maguer et al. 2022) and are applied inconsistently (Kirkland et al. 2023).

> We pose that the performance gap between MOS ratings and TTS-for-ASR performance could go both ways – what if MOS ratings *overestimate* the current capabilities of TTS systems, and TTS-for-ASR performance is symptomatic of this fact?

A key limitation of subjective evaluation, particularly in the context of TTS-for-ASR, is its sample-based nature. For example, if a highly realistic sample is produced repeatedly, its average MOS would still be high. However, this would lead to very low diversity, rendering the resulting dataset unsuitable for training ASR models. Objective evaluation methods, in contrast, can assess the quality and diversity of TTS systems across entire datasets rather than individual samples. In this chapter, we introduce our Text-to-Speech Distribution Score (TTSDS) to evaluate the synthetic speech based on its distribution across a variety of factors. This chapter provides an overview of our work in Minixhofer et al. (2024b).

4.1. Objective Evaluation in Speech and other Domains

With generative capabilities of models increasing, namely for image and text generation, a body of work concerned with objective evaluation of these models has emerged.
For *Large Language Models (LLMs)*, task-based evaluation has become standard – a model will be subjected to a number of tasks aimed at testing different aspects of the model, such as coding, reasoning and recall abilities (Wang et al. 2019). This approach has not been applied to TTS, mainly due the lack of definable "tasks" in speech generation.
In *Computer Vision*, Fréchet Inception Distance (FID) (Heusel et al. 2017) has become the defacto standard. It uses the latent representations of a deep neural network of the synthetic and real data and estimates the distance between the synthetic and real distributions, rather than the individual samples.

4.2. TTSDS – Distribution Score for Synthetic Speech Datasets

In our work, we extend these ideas to the evaluation of TTS systems by introducing the Text-to-Speech Distribution Score (TTSDS), which assesses synthetic speech quality across multiple factors, including prosody, speaker identity, intelligibility, and environmental conditions, inspired by the task-based evaluation of LLMs. By computing the distributional distance be-

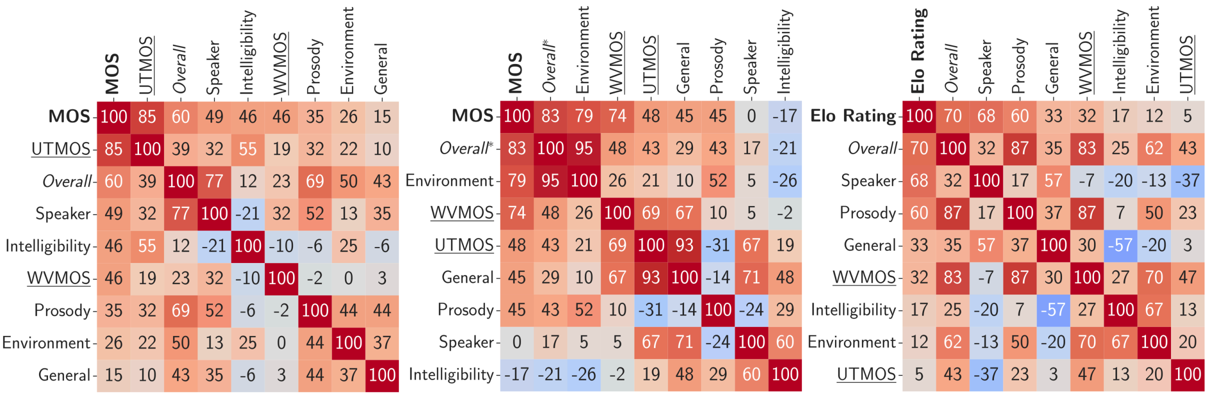


Figure 3: The correlation of UTMOS and WVMOS automatic MOS prediction compared to TTSDS factors and overall score (Minixhofer et al. 2024b).

tween synthetic and real speech datasets, TTSDS can evaluate whole synthetic speech datasets rather than individual samples.

We identify and define several key factors that contribute to the overall quality and effectiveness of synthetic speech, particularly in the context of training automatic speech recognition (ASR) systems. These factors are designed to capture various aspects of the synthetic speech. for both human perception and TTS-for-ASR.

- **Intelligibility** refers to how easily the content of the speech can be understood by a listener or transcribed by an ASR system. It is a critical factor in determining the usefulness of TTS-generated speech for ASR training. Intelligibility is typically measured using Word Error Rate (WER), where lower WER indicates higher intelligibility. However, real speech does not always result in low WER; for instance, children's speech or speech with strong accents may naturally have a higher WER. Therefore, in our evaluation, we compare the distribution of WER across the TTS-generated dataset with that of real speech instead of assigning a higher score to lower average WER.
- **Prosody** has many definitions, but can encompass the rhythm, stress, and intonation of speech, which are vital for conveying meaning and emotion. Realistic prosody is a significant component of natural-sounding speech, making it an essential factor in TTS evaluation. To measure prosody, we analyze several features, including pitch contours, segmental durations, and self-supervised learning (SSL) representations of prosody (Wallbridge et al. 2024). By comparing these features between synthetic and real speech, we assess how closely the TTS system can replicate natural prosodic patterns, which are crucial for both human listeners and the performance of ASR systems trained on synthetic data.
- **Speaker Identity** refers to the ability of a TTS system to generate speech that convincingly mimics a specific speaker's voice. This factor is particularly important in applications where the consistency of speaker characteristics is necessary, such as voice cloning or personalized TTS systems. To evaluate speaker identity, we use speaker embeddings derived from speaker verification models, such as d-vectors (Wan et al. 2018) and WeSpeaker (Wang et al. 2023) representations. These embeddings allow us to measure how closely the synthetic speech matches the reference speaker's characteristics, ensuring that the TTS system can accurately capture the nuances of individual speaker voices.
- **Environmental Conditions** consider the background noise, reverberation, and other artifacts present in the speech. These conditions can significantly affect the perceived quality

and intelligibility of speech. For TTS systems, it is crucial to generate speech that either replicates realistic environmental conditions or produces clean audio suitable for various applications. We assess environmental conditions using correlates such as Signal-to-Noise Ratio (SNR) and measures derived from noise reduction algorithms like VoiceFixer (Liu et al. 2021). By comparing these features between synthetic and real speech, we determine the TTS system's ability to handle or replicate environmental variations.
- **General Speech Quality** is a broader measure that captures the overall alignment of the synthetic speech distribution with that of real speech, without focusing on specific attributes like prosody or intelligibility. This factor is measured using self-supervised learning (SSL) representations from models like Hubert (Hsu et al. 2021) and wav2vec 2.0 (Baevski et al. 2020). These representations are designed to capture high-level features of speech, allowing us to assess the overall fidelity of the synthetic speech in comparison to real speech. The General Speech Quality factor provides a holistic view of how well the TTS system generates speech that is consistent with the complex distribution of natural speech.

We compute an overall score by averaging the individual factor scores.

4.3. Correlation with MOS

To validate the robustness of our TTSDS score, we evaluate its correlation with MOS for three datasets from different time periods – *The Blizzard Challenge 2008* (King et al. 2008), *Back to the Future* (Le Maguer et al. 2022) (2013-2022) and crowdsourced *TTS Arena*[1] scores (2023-2024). These datasets include early unit selection systems, hybrid HMM systems, deep neural networks and the latest token-based TTS systems. We find simply averaging the factors leads to significant correlation with human evaluation for each dataset, although different factors are of different importance.

We use the UTMOS (Saeki et al. 2022) and WVMOS (Andreev et al. 2022) state-of-the-art MOS prediction systems as baselines. Our analysis reveals that the correlation between TTSDS and MOS scores is generally strong, while any correlations with predicted MOS are inconsistent between time periods, and for the latest systems, there is no significant correlations between the baselines and human evaluation at all. We also find the importance of TTSDS factors evolves as TTS technology advances. For example, earlier systems, such as those evaluated in the Blizzard Challenge 2008, show a higher correlation with intelligibility and environmental factors, reflecting the technology's focus on these aspects at the time. In contrast, more recent systems, such as those in TTS Arena, exhibit higher correlations with prosody and speaker identity factors, indicating a shift in human evaluators' priorities as TTS systems have improved in intelligibility and reduced artifacts.

These findings underscore the robustness and adaptability of the TTSDS benchmark across different eras of TTS technology, with significant correlations to human evaluations ranging from early unit selection systems to the latest token-based models. While individual factors such as intelligibility, prosody, and speaker identity each played varying roles depending on the time period, their combined evaluation consistently outperformed traditional MOS prediction systems, as can be seen in Figure 3.

[1] https://huggingface.co/spaces/TTS-AGI/TTS-Arena°

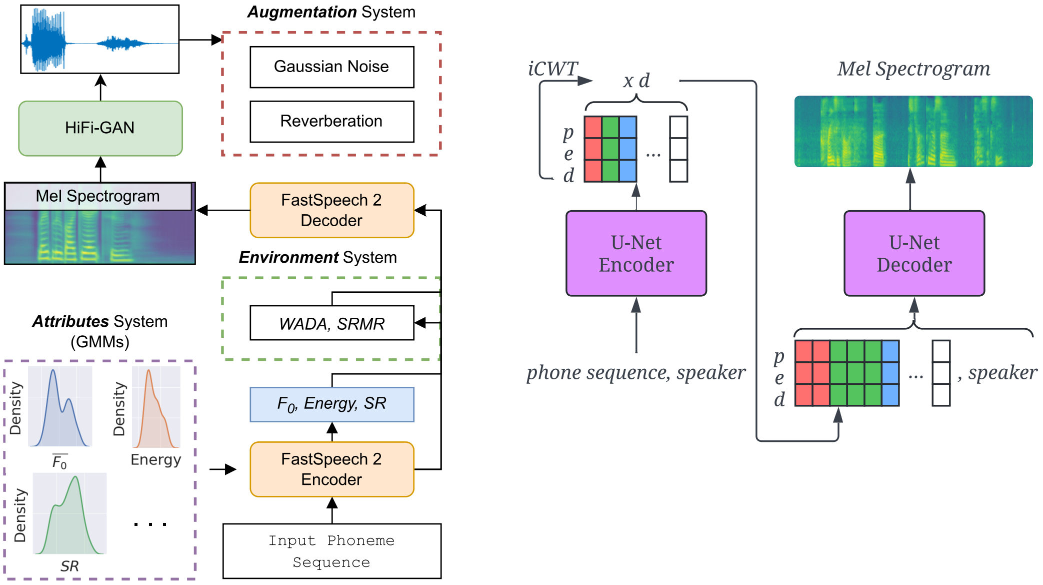


Figure 4: TTS system used for external conditioning in Minixhofer et al. (2023) (*left*) and diffusion training in Minixhofer et al. (2024a) (*right*)

5. Leveraging Key Factors to improve TTS-for-ASR

In this section, we apply our previous findings to actively improve TTS-for-ASR. We do so using external conditioning, post-generation augmentation, stochastic generation methods and model selection based on TTSDS scores. External conditioning is outlined in Section 5.1 and corresponds to Minixhofer et al. (2023). Evaluating the model and training approaches introduced in Section 3 is covered in Section 5.2 and summarizes Minixhofer et al. (2024a). We finally present a preliminary experiment for selecting TTS systems for ASR based on TTSDS (Minixhofer et al. 2024b) in Section 5.3.

5.1. External Conditioning and Post-Generation Augmentation

As discussed in Section 2, conditioning refers to incorporating additional context or external information into the TTS model to improve the quality and relevance of the generated speech. This context can include factors such as speaker identity, prosody, and environmental conditions, which can significantly increase the utility of the synthetic speech for ASR training.

In the literature, conditioning techniques have been explored extensively. For instance, Global Style Tokens (GST) (Wang et al. 2018) and Variational Autoencoders (VAEs) (Kingma & Welling 2022) are commonly used to capture and control variations in speaker style, prosody, and other speech characteristics. These methods allow the model to generate speech with specific attributes, improving the alignment between synthetic and real speech distributions.

In our work, we extend these techniques by incorporating a broader range of conditioning factors, such as those related to the acoustic environment and speaker characteristics. We employ Gaussian Mixture Models (GMMs) to sample realistic attribute values for these factors during inference, allowing our TTS system to generate more diverse and contextually appropriate speech. Additionally, we apply data augmentation techniques, such as adding noise and reverberation, to further align the synthetic speech with real-world conditions.

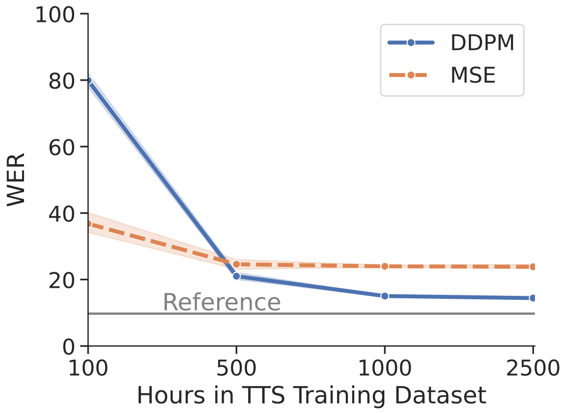


Figure 5: Scaling properties of Denoising Diffusion Probabilistic Model (DDPM) compared to equivalent model trained using Mean Squared Error (MSE) for TTS-for-ASR.

This approach not only improves the quality of the generated speech but also enhances its effectiveness as training data for ASR systems, particularly in low-resource settings. By targeting the weaker areas of TTS and incorporating external conditioning, we aim to reduce the performance gap and improve the generalization of ASR models trained on synthetic data.

5.2. Evaluation of MSE and Diffusion Loss

Using a non-autoregressive U-Net model (Ronneberger et al. 2015), we investigate the effect of, while keeping all other factors equivalent, substituting the standard MSE loss used in FastSpeech 2 and others with the diffusion loss described in Section 3.2. In particular, we investigate the scaling behavior with respect to training using differing amounts of data – the general trend could tell us to what diffusion TTS models could be scaled to reduce the performance gap for TTS-for-ASR.

As can be seen in Figure 5, the diffusion model is initially worse but keeps improving beyond 500 hours of training data, although improvements beyond 1000 hours are diminishing. Our final 2500 hour model yields a new lowest WERR of 1.46, hinting a big part of the performance gap was due to model and training data constraints. However, the diminishing returns beyond 1000 hours show that the remainder of the gap is still unexplained.

For more details on training procedure and model architecture, see Minixhofer et al. (2024a).

5.3. Predicting ASR Performance Based on Factor Analysis

In preliminary experiments, we have extended our TTSDS benchmark to include 20 of the latest TTS systems[2], and measured the correlation of the TTSDS factors to TTS-for-ASR performance on a small subset of just 10 minutes of data. We find a significant correlation of the overall TTSDS scores with WERR ($r = -0.31, p < 0.05$), however, the environment score, which proved insubstantial for MOS correlation, seems to be much more significant for WERR ($r = -0.63, p < 0.001$).

[2] https://huggingface.co/spaces/ttsds/benchmark°

6. Future Work & Timeline

In this chapter, we present experimental work yet to be conducted, as well as a timeline for the remainder of the work in this PhD.

6.1. Predicting TTS-for-ASR Performance with TTSDS

We plan to extend the work presented in Section 5.3 with hours rather than minutes of training data, and to hold out a number of TTS systems to test if these correlations can be used to select TTS systems for TTS-for-ASR based on some combination of TTSDS factors alone, which would allow for efficient selection of TTS models for ASR training.

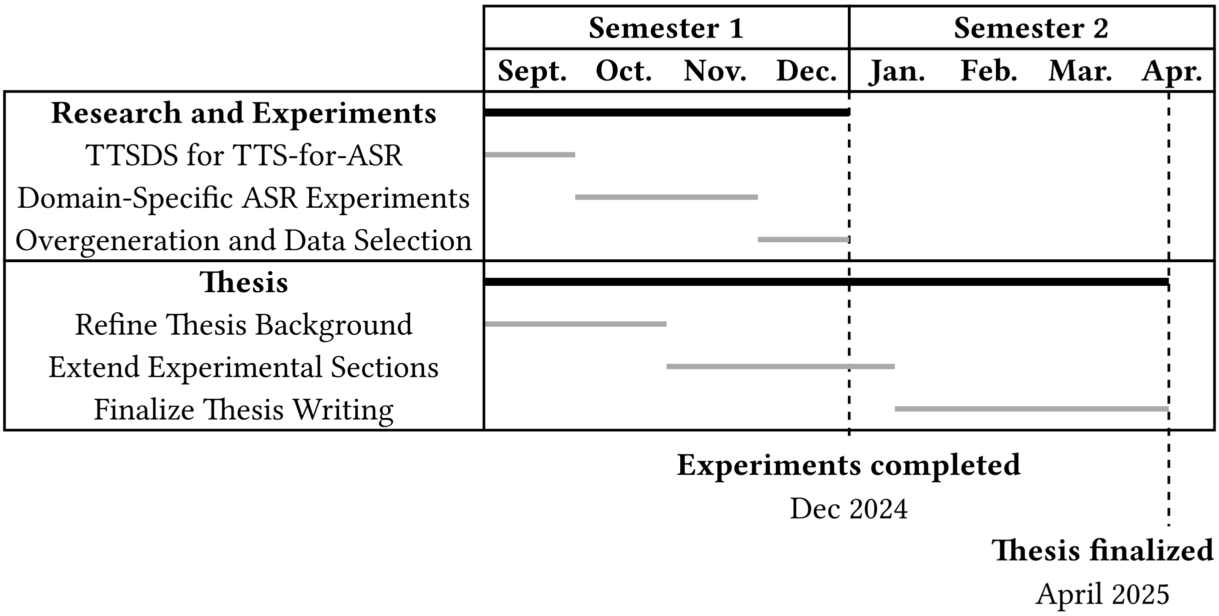
6.2. Application to Domain-Specific ASR with Limited Data

One of the most promising areas for future work is the application of TTS data generation techniques to domain-specific ASR systems, particularly in low-resource settings. By leveraging our TTSDS analysis, we aim to tailor TTS systems to generate synthetic speech that is more effective for training ASR models in specific domains, such as children's speech, accented speech, and low-resource languages. These domains often suffer from a scarcity of annotated data, making them good candidates for the use of synthetic speech.

6.3. Overgeneration and Optimal Data Selection Strategies

Another key area of future exploration involves the concept of overgeneration – creating more synthetic speech data than needed and selectively using only the most effective subsets for ASR training. Building on the work of (Sun et al. 2020), who successfully utilized oversampling in TTS, we plan to employ our TTSDS scores as a metric to guide the selection of the most useful synthetic data. This approach will help us discard less effective data, thereby optimizing the ASR training process. Additionally, we aim to extend our evaluation measures to operate at a finer granularity, such as on a sample or mini-batch level, to facilitate easier and more precise data selection. This could further enhance the efficiency and effectiveness of ASR models trained with synthetic data.

6.4. Timeline for Remaining Work



External Resources and Links

In this PhD, we put emphasis on making our work available and useful to a wider community than can be reached by research papers alone. The following is a non-exhaustive list of open source software which has been published as part of this PhD.

- **Alignments**, a library to easily access speech datasets with forced alignments.
  https://github.com/minixc/alignments°
- **Masked Prosody Model**, a self-supervised model for prosody representations.
  https://huggingface.co/cdminix/masked_prosody_model°
- **LightningFastSpeech2**, an improved implementation of FastSpeech2.
  https://github.com/minixc/LightningFastSpeech2°
- **Phones**, a library for accessing IPA phone feature vectors and more.
  https://github.com/minixc/phones°
- **TTSDS Benchmark**, a leaderboard with evaluation scores for the latest TTS systems.
  https://ttsdsbenchmark.com/leaderboard°
- **Vocex**, a model for estimation of energy, pitch, signal-to-noise ratio and reverberation.
  https://github.com/minixc/vocex°

Declaration of Content

This review includes research from the following publications, which are also appended to this document.

- Minixhofer C, Klejch O, Bell P. 2023. Evaluating and reducing the distance between synthetic and real speech distributions. *Interspeech*
  - Section 2.2, Section 5.1; Figure 4 (*left*)
- Minixhofer C, Klejch O, Bell P. 2024a. Beyond Oversmoothing: Evaluating DDPM and MSE for Scalable Speech Synthesis in ASR. *To be submitted to ICASSP, attached to this document.*
  - Section 3.2, Section 5.2; Figure 5
- Minixhofer C, Klejch O, Bell P. 2024b. TTSDS – Text-to-Speech Distribution Score. *Under review for SLT 2024, attached to this document.*
  - Section 4.2, Section 5.3; Figure 3
- Wallbridge S, Minixhofer C, Lai C, Bell P. 2024. Systematicity in prosody beyond lexical content: a study of self-supervised learning. *Not published yet, attached to this document.*
  - Part of the prosody measure in Section 4.2.

For better coherence, the following publication has been excluded from this review.

- Minixhofer C, Klejch O, Bell P. 2021. Mask-combine Decoding and Classification Approach for Punctuation Prediction with real-time Inference Constraints. *arXiv:2112.08098*

Bibliography

Andreev P, Alanov A, Ivanov O, Vetrov D. 2022. HiFi++: a unified framework for neural vocoding, bandwidth extension and speech enhancement. *arXiv:2203.13086*

Baevski A, Zhou Y, Mohamed A, Auli M. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *NeurIPS*

Butler D J, Wulff J, Stanley G B, Black M J. 2012. A Naturalistic Open Source Movie for Optical Flow Evaluation. *ECCV*

Casanova E, Davis K, Gölge E, Göknar G, Gulea I, et al. 2024. XTTS: a Massively Multilingual Zero-Shot Text-to-Speech Model. *arXiv:2401.02839*

Casanova E, Shulby C, Korolev A, Junior A C, Soares A d S, et al. 2022. A single speaker is almost all you need for automatic speech recognition. *arXiv:2204.00618*

Chen S, Liu S, Zhou L, Liu Y, Tan X, et al. 2024. VALL-E 2: Neural Codec Language Models are Human Parity Zero-Shot Text to Speech Synthesizers. *arXiv:2406.05370*

Dosovitskiy A, Fischer P, Ilg E, Hausser P, Hazirbas C, et al. 2015. FlowNet: Learning Optical Flow with Convolutional Networks. *ICCV*

Du C, Yu K. 2020. Speaker augmentation for low resource speech recognition. *ICASSP*

Fan L, Chen K, Krishnan D, Katabi D, Isola P, Tian Y. 2024. Scaling laws of synthetic images for model training… for now. *CVPR*

Handa A, Patraucean V, Badrinarayanan V, Stent S, Cipolla R. 2015. SceneNet: Understanding Real World Indoor Scenes with Synthetic Data. *arXiv:1511.07041*

Heusel M, Ramsauer H, Unterthiner T, Nessler B, Hochreiter S. 2017. GANs trained by a two time-scale update rule converge to a local nash equilibrium. *NIPS*

Ho J, Jain A, Abbeel P. 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*

Hsu W-N, Bolte B, Tsai Y-H H, Lakhotia K, Salakhutdinov R, Mohamed A. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *TASLP*

Hu T-Y, Armandpour M, Shrivastava A, Chang J-H R, Koppula H, Tuzel O. 2022. SYNT++: Utilizing Imperfect Synthetic Data to Improve Speech Recognition. *ICASSP*

Karakasidis G, Robinson N, Getman Y, Ogayo A, Al-Ghezi R, et al. 2023. Multilingual TTS Accent Impressions for Accented ASR. *Text, Speech, And Dialogue.* Springer Nature Switzerland

King S, Clark R A, Mayo C, Karaiskos V. 2008. The blizzard challenge 2008. *The Blizzard Challenge Workshop*

Kingma D P, Welling M. 2022. Auto-Encoding Variational Bayes

Kingma D. 2013. Auto-Encoding Variational Bayes. *arXiv:1312.6114*

Kirkland A, Mehta S, Lameris H, Henter G E, Szekely E, Gustafson J. 2023. Stuck in the MOS pit: A critical analysis of MOS test methodology in TTS evaluation. *SSW*

Kröger B J. 2023. Articulatory Speech Synthesis in the Context of Speech Research and Speech Technology: Review and Prospect. *ESSV*

Laptev A, Korostik R, Svischev A, Andrusenko A, Medennikov I, Rybin S. 2020. You Do Not Need More Data: Improving End-To-End Speech Recognition by Text-To-Speech Data Augmentation. *CISP-BMEI*

Le Maguer S, King S, Harte N. 2022. Back to the Future: Extending the Blizzard Challenge 2013. *Interspeech*

Li J, Gadde R, Ginsburg B, Lavrukhin V. 2018. Training neural speech recognition systems with synthetic speech augmentation. *arXiv:1811.00707*

Liu H, Kong Q, Tian Q, Zhao Y, Wang D, et al. 2021. VoiceFixer: Toward general speech restoration with neural vocoder. *arXiv:2109.13731*

Mayer N, Ilg E, Hausser P, Fischer P, Cremers D, et al. 2015. A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation. *arXiv:1512.02134*

Minixhofer C, Klejch O, Bell P. 2021. Mask-combine Decoding and Classification Approach for Punctuation Prediction with real-time Inference Constraints. *arXiv:2112.08098*

Minixhofer C, Klejch O, Bell P. 2023. Evaluating and reducing the distance between synthetic and real speech distributions. *Interspeech*

Minixhofer C, Klejch O, Bell P. 2024a. Beyond Oversmoothing: Evaluating DDPM and MSE for Scalable Speech Synthesis in ASR. *To be submitted to ICASSP, attached to this document.*

Minixhofer C, Klejch O, Bell P. 2024b. TTSDS – Text-to-Speech Distribution Score. *Under review for SLT 2024, attached to this document.*

Nikolenko S I. 2021. *Synthetic Data for Deep Learning*. Springer International Publishing

Panayotov V, Chen G, Povey D, Khudanpur S. 2015. Librispeech: an ASR corpus based on public domain audio books. *ICASSP*

Pine A, Wells D, Brinklow N, Littell P, Richmond K. 2022. Requirements and Motivations of Low-Resource Speech Synthesis for Language Revitalization. *ACL*

Ren Y, Hu C, Tan X, Qin T, Zhao S, et al. 2021. FastSpeech 2: Fast and High-Quality End-to-End Text to Speech. *ICLR*

Ronneberger O, Fischer P, Brox T. 2015. U-net: Convolutional networks for biomedical image segmentation. *MICCAI*

Rosenberg A, Zhang Y, Ramabhadran B, Jia Y, Moreno P, et al. 2019. Speech recognition with augmented synthesized speech. *ASRU*

Rossenbach N, Hilmes B, Schlüter R. 2023. On the Relevance of Phoneme Duration Variability of Synthesized Training Data for Automatic Speech Recognition. *ASRU*

Rossenbach N, Schlüter R, Sakti S. 2024. On the Problem of Text-To-Speech Model Selection for Synthetic Data Generation in Automatic Speech Recognition. *arXiv:2407.21476*

Rossenbach N, Zeyer A, Schlüter R, Ney H. 2020. Generating synthetic audio data for attention-based speech recognition systems. *ICASSP*

Saeki T, Xin D, Nakata W, Koriyama T, Takamichi S, Saruwatari H. 2022. UTMOS: UTokyo-SaruLab System for VoiceMOS Challenge 2022. *INTERSPEECH*

Shen J, Pang R, Weiss R J, Schuster M, Jaitly N, et al. 2018. Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions. *ICASSP*

Sun G, Zhang Y, Weiss R J, Cao Y, Zen H, et al. 2020. Generating Diverse and Natural Text-to-Speech Samples Using a Quantized Fine-Grained VAE and Autoregressive Prosody Prior. *ICASSP*

Thai B, Jimerson R, Arcoraci D, Prud'hommeaux E, Ptucha R. 2019. Synthetic Data Augmentation for Improving Low-Resource ASR. *WNYISPW*. IEEE

Wallbridge S, Minixhofer C, Lai C, Bell P. 2024. Systematicity in prosody beyond lexical content: a study of self-supervised learning. *Not published yet, attached to this document.*

Wan L, Wang Q, Papir A, Moreno I L. 2018. Generalized end-to-end loss for speaker verification. *ICASSP*

Wang A, Pruksachatkun Y, Nangia N, Singh A, Michael J, et al. 2019. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. *NIPS*

Wang G, Rosenberg A, Chen Z, Zhang Y, Ramabhadran B, et al. 2020. Improving speech recognition using consistent predictions on synthesized speech. *ICASSP*

Wang H, Liang C, Wang S, Chen Z, Zhang B, et al. 2023. WeSpeaker: A research and production oriented speaker embedding learning toolkit. *ICASSP*

Wang Y, Stanton D, Zhang Y, Ryan R-S, Battenberg E, et al. 2018. Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis. *ICML*

Yuen K C, Haoyang L, Siong C E. 2023. ASR Model Adaptation for Rare Words Using Synthetic Data Generated by Multiple Text-To-Speech Systems. *APSIPA ASC*

Zen H, Dang V, Clark R, Zhang Y, Weiss R J, et al. 2019. LibriTTS: A Corpus Derived from LibriSpeech for Text-to-Speech. *Interspeech*

Index of Figures