EURASIP Journal on Audio,
Speech, and Music Processing

REVIEW

Open Access

Deep learning-based expressive speech synthesis: a systematic review of approaches, challenges, and resources

Huda Barakat[1*], Oytun Turk[2] and Cenk Demiroglu[3]

Abstract

Speech synthesis has made significant strides thanks to the transition from machine learning to deep learning models. Contemporary text-to-speech (TTS) models possess the capability to generate speech of exceptionally high quality, closely mimicking human speech. Nevertheless, given the wide array of applications now employing TTS models, mere high-quality speech generation is no longer sufficient. Present-day TTS models must also excel at producing expressive speech that can convey various speaking styles and emotions, akin to human speech. Consequently, researchers have concentrated their efforts on developing more efficient models for expressive speech synthesis in recent years. This paper presents a systematic review of the literature on expressive speech synthesis models published within the last 5 years, with a particular emphasis on approaches based on deep learning. We offer a comprehensive classification scheme for these models and provide concise descriptions of models falling into each category. Additionally, we summarize the principal challenges encountered in this research domain and outline the strategies employed to tackle these challenges as documented in the literature. In the Section 8, we pinpoint some research gaps in this field that necessitate further exploration. Our objective with this work is to give an all-encompassing overview of this hot research area to offer guidance to interested researchers and future endeavors in this field.

Keywords Speech synthesis, Expressive speech, Emotional speech, Deep learning

1 Introduction

Since the late 1950s, computer-based text-to-speech systems (TTS) have undergone significant advancements [1], culminating in the production of models that generate speech almost indistinguishable from that of a human. This progress has followed a path consisting of several stages, beginning with conventional methods named as concatenative synthesis and progressing to more advanced approaches known as statistical parametric speech synthesis (SPSS). Advanced approaches are mainly based on machine learning algorithms like hidden Markov models (HMMs) and gaussian mixture models (GMMs). Despite this progress, speech generated by these methods was still noticeably artificial. However, the emergence of deep learning (DL) as a new branch under machine learning (ML) in 2006 has led to significant improvements. Speech synthesis researchers, like many in other research fields, started incorporating deep neural networks (DNN) in their models. Initially, DNNs replaced HMMs and GMMs in SPSS models while the main structure still follows the primary framework of SPSS models as shown in Fig. 1. As discussed in [2], the deep learning-based models have overcome many limitations and problems associated with machine learning-based models.

*Correspondence:
Huda Barakat
huda.barakat@ozu.edu.tr
[1] Department of Computer Science, Ozyegin University, Istanbul 34794, Turkey
[2] Independent Consultant/Researcher, Beaverton, OR, USA
[3] Department of Electrical and Electronics Engineering, Ozyegin University, Istanbul 34794, Turkey

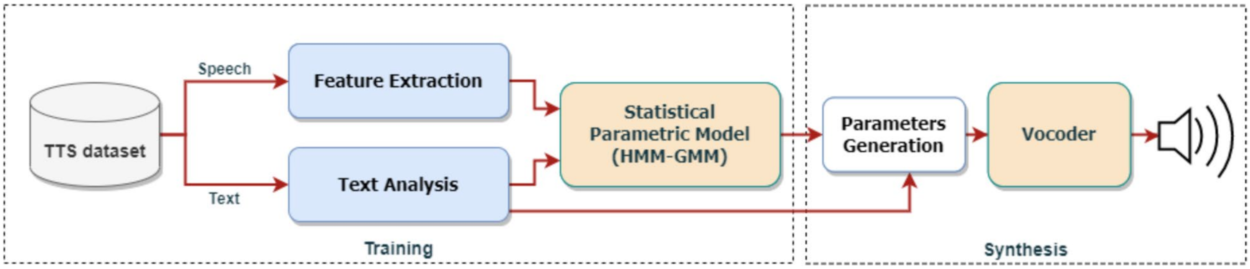

Fig. 1 Statistical parametric speech synthesis model structure

Researchers continue to aim for improved speech quality and more human-like speech despite past advancements. Additionally, they seek to simplify the framework of the text-to-speech models due to the intricate nature of the SPSS structure, which limits progress in this field to those with extensive linguistic knowledge and expertise. Deep learning advancements have brought about the simple encoder-decoder structure for TTS models as sequence-to-sequence (Seq2Seq) approaches. The proposed approaches have simplified the structure of conventional TTS with multiple components into training a single network that converts a set of input text characters/phonemes into a set of acoustic features (mel-spectrograms). A main concern in these advanced TTS models is the mapping process between the input and output sequences, which is a one-to-many problem, as the single input text can have multiple speech variations as output. In fact, there are two groups of recent TTS models, as shown in Fig. 2. The first group generates mel-spectrograms in a sequential (autoregressive) manner using soft and automatic attention alignments between input and output sequences, such as the Tacotron model [3, 4]. The second group utilizes hard alignments between the phonemes/characters and mel-spectrograms, and thus its speech generation process is parallel (non-autoregressive), as in the FastSpeech model [5, 6]. This improvement in the structure of the TTS model has encouraged rapid development in the field within the last few years, during which the proposed models produced speech that is nearly indistinguishable from human speech.

Human speech is highly expressive and reflects various factors, such as the speaker's identity, emotion, and speaking style. In addition, there are many applications in which speech synthesis can be utilized, especially expressive speech synthesis. For instance, audiobooks and podcast applications that create audio versions of eBooks and podcasts, translation applications which provide real-time translation of foreign language text, dubbing applications that generate an alternative audio track for a video with different content, speaker, or language, and content creation applications which help produce audio versions of textual content, such as blogs and news articles. E-learning applications that allow for adding voice-over audio to e-learning courses, and conversational AI applications enable machines to communicate with users in a human-like manner, such as AI chatbots and virtual assistants.

As spoken language is a crucial component in such applications, users must feel as if they are

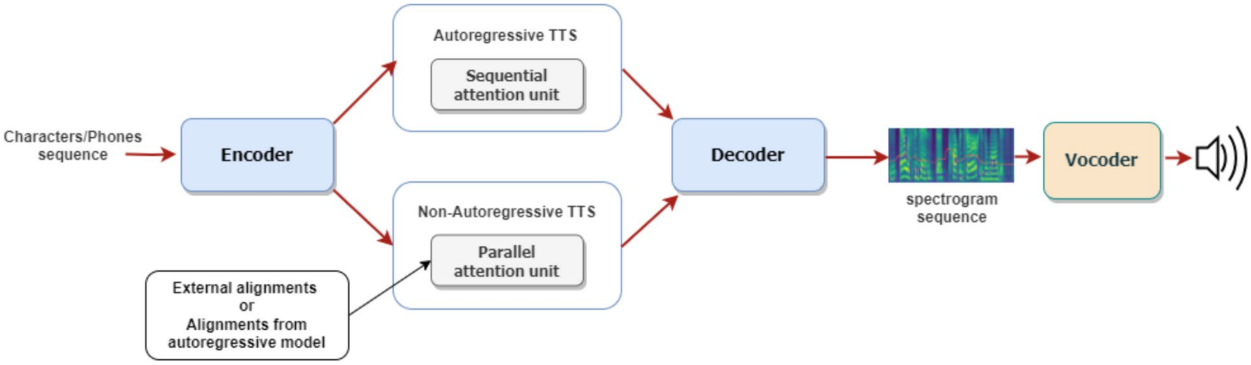
Characters/Phones sequence
Encoder
Autoregressive TTS
Sequential attention unit
Non-Autoregressive TTS
Parallel attention unit
External alignments or Alignments from autoregressive model
Decoder
spectrogram sequence
Vocoder

Fig. 2 Structure of TTS models based on deep learning. The autoregressive models follows the upper track with sequential attention mechanism, non-autoregressive models follow the lower track with parallel attention unit utilizing alignments from an external aligner or a pretrained autoregressive model

communicating with a real human rather than a machine. Therefore, the speech generated by these applications should convey appropriate emotion, intonation, stress, and speaking style to match the ongoing conversation or the content type and context of the text being read.

As a result, there has been a recent attention towards building efficient expressive speech synthesis models as another step forward in achieving human-like speech. Therefore, many studies have been devoted to expressive speech synthesis (ETTS) as a hot research area, particularly over the last 5 years. In this work, we present the findings of our systematic literature review on ETTS field from the past 5 years. Firstly, we suggest a classification schema of deep learning-based ETTS models that are proposed during this period, based on structures, and learning methods followed in each study. A summary is then provided for each category in the classification schema and main papers related to this category. After that, we outline the main challenges in the ETTS area and solutions that have been proposed to solve them from literature. Finally, we conclude with a discussion of the implications of our work and a highlight of some gaps that require further research in this area.

During our work on this review of expressive speech synthesis literature, we came across several review papers that focus on different stages of development in the speech synthesis field. The majority of these reviews concentrate on DL-based TTS approaches [7–13], while only a few papers [13, 14] cover recent TTS approaches in addition to early conventional ones. However, to the best of our knowledge, there are no review papers that cover the fast growth in the (expressive) speech synthesis area, especially in the last few years. Therefore, our main goal in this review is to provide an overview of research trends, techniques, and challenges in this area during this period. We hope that our work will offer researchers a comprehensive understanding of how and what has been accomplished in this field and the gaps that need to be filled as guidance for their future efforts.

While we were writing this paper, we came across an interesting recent review paper [15] that is similar to our work. However, the review in [15] covers emotional speech synthesis (ESS) as a sub-field of voice transformation while our work is more comprehensive as a systematic literature review that discusses approaches, challenges, and resources. Furthermore, the taxonomy we provide for the reviewed approaches differs from the one given in [15] as elaborated in the next section.

The remaining sections of this paper are structured as follows: Section 2 provides an explanation of the methodology employed for conducting this review. Sections 3 and 4 describe the different main and sub-categories of the proposed classification schema for DL-based expressive TTS models. Main challenges facing ETTS models and how they have been tackled in the literature are covered in Section 5. We then give a brief description of ETTS datasets and applied evaluation metrics in Sections 6 and 7, respectively. Finally, Section 8 concludes the paper.

2 Method

The last few years have seen rapid growth in expressive and emotional speech synthesis approaches, resulting in a large number of papers and publications in this area. Here, we present the outcomes of a systematic literature review of the last 5 years' publications within this active research area. This section describes the methodology used to conduct the review, illustrated by Fig. 3, which consists of three main stages: paper selection, paper exclusion, and paper classification.

2.1 Paper selection

For our review, we used the Scopus [16] database to retrieve papers as it encompasses most of the significant journals and conferences pertaining to the speech synthesis field. Our query criteria to find relevant papers on Scopus were twofold: (1) the paper title must include at least one of four words (emotion* OR expressive OR prosod* OR style) that denote expressive speech, and (2)

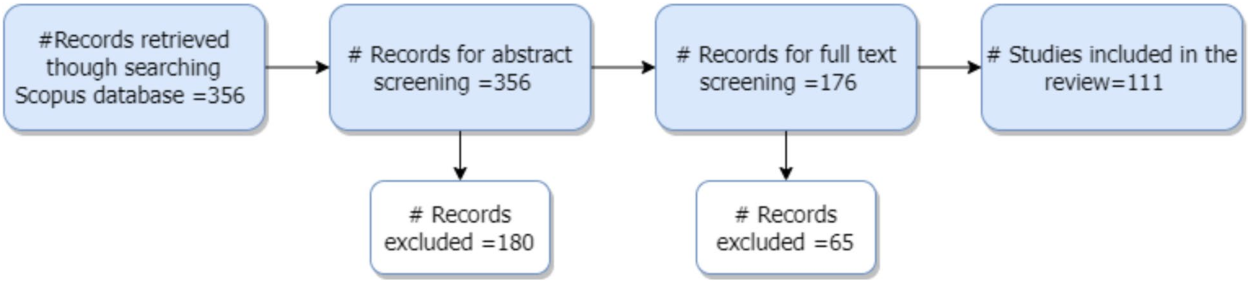
#Records retrieved though searching Scopus database =356
# Records for abstract screening =356
# Records for full text screening =176
# Studies included in the review=111
# Records excluded =180
# Records excluded =65

Fig. 3 Flowchart of methodology used for selecting publications

the paper title, abstract, or keywords must comprise the terms "speech" AND "synthesis," in addition to at least one of the above-mentioned words for expressive speech. We considered all papers written in English and published in journals or conferences since 2018. The search query was conducted in January 2023, and it yielded 356 papers. Scopus provides an Excel file containing all the primary information of the retrieved papers, which we used in the second stage of our review.

2.2 Exclusion of papers

The exclusion of papers occurred in two phases. In the first phase, we screened the abstract text, while in the second phase, we screened the full text of the paper. Five main constraints were used to exclude papers, including (1) papers that were not related to the TTS field, (2) papers that were not DL-based models, (3) papers that did not focus on expressive or emotional TTS models, (4) papers that were too specific to non-English languages, and (5) papers that lacked details about the applied method. After screening the paper abstracts, we excluded 180 papers, mostly based on the first exclusion criterion. During the second exclusion phase, in which we read the full text of each paper, we identified another 65 papers that met at least one of the five exclusion criteria. Consequently, 111 papers were included in the third stage of our review. Additionally, a group of recently published papers in this area [17–25] was hand-picked and added to the final set of selected papers. While most of the reviewed papers trained their models on English data, a few other papers used data in other languages as listed in Table 1.

2.3 Paper classification

After summarizing the approach proposed for generating expressive speech in each selected paper, we categorized the papers based on the learning approach applied in each one. Accordingly, papers are divided into two main categories, including supervised and unsupervised approaches. Under the supervised category, where labeled data is utilized, we identified three subcategories based on how models are employed expressive speech synthesis. The three proposed subcategories are (1) labels as input features, (2) labels as separate layers or models, and (3) labels for emotion predictors/classifiers.

Papers in the unsupervised approaches category are grouped into four different subcategories based on the main structure or method used for modeling expressivity in these papers. From our observation, most of the proposed methods in the last 5 years are based on three main early works in this field, namely, reference encoder [74], global style tokens [75], and latent features via variational autoencoders (VAE) [76, 77]. Specifically, proposed models in most of the papers under this category can be considered as an extension or enhancement of one of the three previously mentioned methods. Besides, we identify a fourth subcategory that includes the recent TTS models representing the new trend in the TTS area, which utilizes in-context learning. There is one factor common to all these four unsupervised models, which is that they are all based on using an audio reference/prompt. Additionally, we added a fifth subcategory (named other approaches) in which we include approaches outside the previous four main unsupervised approaches. Figure 4 illustrates the proposed classification schema for the DL-based expressive speech synthesis models.

Table 1 List of other languages than English used in the reviewed publications to train proposed ETTS models with links to related papers

Language	References	# papers
Chinese	[23, 26–40]	16
Mandarin Chinese	[17, 21, 36, 41–54]	18
Korean	[19, 55–64]	11
Japanese	[65–70]	6
Mexican-Spanish	[20, 71]	2
Telugu	[72]	1
Bahasa Indonesia	[73]	1
Multilingual	[19, 25, 36, 44, 53–55]	7

3 Supervised approaches

Supervised approaches refer to models that are trained on datasets with emotion labels. Those labels guide model training, enabling it to learn accurate weights. Early deep learning-based expressive speech synthesis systems were primarily supervised models that utilized labeled speech exhibiting various emotions (such as sadness, happiness, and anger) or speaking styles (such as talk-show, newscaster, and call-center). Note that the term style has also been used to refer to a set of emotions or a mixture of emotions and speaking styles [59, 68, 78, 79]. Generally, the structure of early conventional TTS models was built upon two primary networks: one for predicting duration and the other for predicting acoustic features. These acoustic features were then converted to speech using vocoders. Both networks receive linguistic features extracted from the input text. In supervised ETTS approaches, speech labels (emotions and/or styles) are represented in the TTS model as either input features or as separate layers, models, or sets of neurons for each specific label. The following sections explain these three representations in detail then we provide a general summary of the supervised approaches reviewed in this work in Table 2.

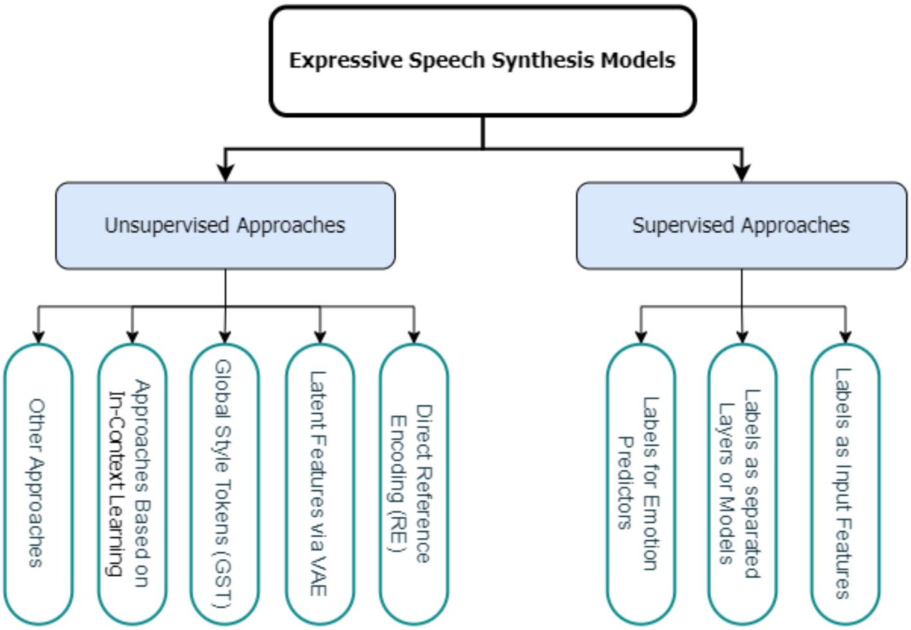
Expressive Speech Synthesis Models
Unsupervised Approaches
Supervised Approaches
Other Approaches
Approaches Based on In-Context Learning
Global Style Tokens (GST)
Latent Features via VAE
Direct Reference Encoding (RE)
Labels for Emotion Predictors
Labels as separated Layers or Models
Labels as Input Features

Fig. 4 General proposed classification schema of the deep learning based models for expressive speech synthesis

3.1 Labels as input features

The most straightforward method for representing emotion labels of annotated datasets as input to the TTS model is by using a one-hot vector. This approach entails using a vector with a size equivalent to the number of available labels. In this vector, a value of (1) is assigned to the index corresponding to the label ID, while all other values are set to (0). Many early ETTS models [43, 56, 65, 69, 78, 80, 82, 84] advocated for this direct representation of emotion labels in order

Table 2 Summary of supervised ETTS approaches. "LingF" stand for linguistic features, "EmoL" stands for emotion labels, "MelS" stands for mel-spectrograms, "PhnSq" stands for phoneme sequence, "ChrSq" stands for character sequence, "ProsF" stands for prosodic features, "LM-F" stands for features from a language model and "ET" stands for expression/emotion transplantation

Ref No.	Inputs	Emotion label representation	ET	TTS model
[80]	LingF+EmoL	One-hot vector		DL-SPSS, HMM
[65]	LingF+EmoL	One-hot vector/dependent layers	✓	DL-SPSS
[66]	LingF+EmoL	Perception vector/matrix		DL-SPSS
[41]	LingF+EmoL	One-hot vector		DL-SPSS
[42]	LingF+EmoL	Dependent layers		DL-SPSS
[81]	LingF+EmoL	One-hot vector/set of neurons	✓	DL-SPSS
[43]	LingF+EmoL	One-hot vector/dependent layers/separated Model		DL-SPSS
[82]	LingF+EmoL	One-hot vector	✓	DL-SPSS
[83]	PhnSq+LM-F+EmoL	Embedding vector		Encode-Dttention-Decoder
[28, 78]	LingF+EmoL	One-hot vector/dependent layers/separated Model	✓	DL-SPSS
[26]	PhnSq+MelS+EmoL	One-hot vector as ground truth GSTs weights		Tacotron2
[27]	PhnSq+LingF+EmoL	Embedding vector		Tacotron2
[84]	LingF+EmoL	Joint embedding with other data labels		DL-SPSS
[85]	LingF+ProsF+EmoL	Ground truth for a classifier		DL-SPSS
[86]	PhnSq+EmoL	Embedding vector		Transformer TTS
[32, 36]	ChrSq+MelS+EmoL	Ground truth for a classifier		Tacotron2
[69]	LingF+EmoL	One-hot vector/dependent layers	✓	DL-SPSS
[34]	PhnSq+MelS+EmoL	Ground truth for a classifier		Tacotron2
[64]	ChrSq+LM-F+EmoL	Ground truth for a predictor		Tacotron2
[39, 87]	PhnSq+MelS+EmoL	Ground truth for a classifier		Tacotron2

to generate speech encompassing various emotions. The one-hot emotion vector, also referred to as a style/emotion code in some studies [43, 78, 80, 82], is concatenated with the input linguistic features of the model.

When dealing with large number of labels, the one-hot representation becomes both high-dimensional and sparse. Moreover, in other scenarios, merging label vectors with input features instead of concatenation can lead to length mismatch issues. In both situations, the embedding layer offers a solution by creating a continuous representation for each label, known as embedding vectors. Unlike the one-hot vector, which is constrained in size based on the number of labels, an emotion embedding can have any dimension, regardless of the number of available labels.

For instance, in [84], each sample in the training dataset has three separated labels including speaker, style (emotion), and cluster. In this context, the cluster value indicates the consistency in speech quality of a given speaker and style pair. If one-hot vector is used to represent each unique combined label of each sample, the resulting label vector will be high dimensional (which in this case is 67). Therefore, the three one-hot vectors representing the given three labels are combined and passed as input to an embedding layer to reduce its dimension (in this case 15). On a different note, [41] utilizes an embedding layer to expand concise binary one-hot label vectors to match with the dimensions of the input features to be added together as input to the TTS model.

To address the potential disparities between a talker's intent and a listener's perception when annotating emotional samples, in [66], a different methodology for representing labels is introduced. In the context of N emotion classes, each sample from the talker may be perceived by the listener as one of the N emotions. In response to this, the paper suggests the adoption of a singular vector termed the 'perception vector,' with N dimensions. This vector represents how samples from a specific emotion class are distributed among the N emotions, based on the listener's perception. Furthermore, in the context of multiple listeners, each emotion class can be represented as a confusion matrix that captures the diverse perceptions of samples belonging to that emotion class by multiple listeners.

3.2 Labels as separate layers/models

In this approach, to represent emotion or style labels in TTS models, each label is associated with either a separate instance of the DNN model, an emotion-specific layers, or a set of emotion-specific neurons within a layer. Initially, the model is trained using neutral data, which typically has larger size. Subsequently, in the first approach, multiple copies of the trained model are fine-tuned using emotion-specific data of small size [43, 78]. In the second approach, instead of creating an individual model for each emotion, only specific model layers (usually the uppermost or final layers) from the employed DNN model are assigned to each emotion [43, 65, 69, 78] as shown by Fig. 5. While shared layers are adjusted during training using neutral data, output layers corresponding to each emotion are modified exclusively when the model is trained with data from the respective emotion.

Alternatively, when dealing with limited data for certain emotions/styles, the model can initially undergo training for emotions with large amount of data. Following this step, the weights of the shared layers within the model are fixed, and only the weights of the top layers are fine-tuned using the limited, emotion-specific data [42]. Another method for representing emotion labels involves allocating specific neurons from a layer within the DNN model for each emotion. In this approach, the hidden layers of the model could be expanded by introducing new neurons. Then, as outlined in [81], particular neurons from this expanded set are assigned to represent each distinct emotion. Importantly, the associated weights of these specific neuron subsets are adjusted solely during the processing of data relevant to the corresponding emotion. Furthermore, by substituting the subset of neurons dedicated to a particular emotional class with a different set, the model becomes capable of generating speech imbued with the desired emotional class. This capability holds true even for new speakers who only possess neutral data, and in this case, it is known as expression/emotion transplantation.

3.3 Labels for emotion predictors/classifiers

Another common approach to utilize emotion labels is to use them directly or via emotion predictor or classifier to support the process of extracting emotion/prosody embedding. For example, in [26] emotion labels represented as one-hot vectors are used as targets for the weight vectors of GSTs (explained in Section 4.3) where a cross entropy loss between the two vectors is added to the total loss function. Yoon et al. [64] proposes a joint emotion predictor based on the Generative Pre-trained Transformer (GPT)-3 [88]. The proposed predictor produces two outputs including emotion class and emotion strength based on features extracted from input text by (GPT)-3. A joint emotion encoder is then used to encode the predictor outputs into a joint emotion embedding. The joint emotion predictor is trained with the guidance of the emotion labels and emotion strength values obtained via a ranking support vector machine (RankSVM) [89].

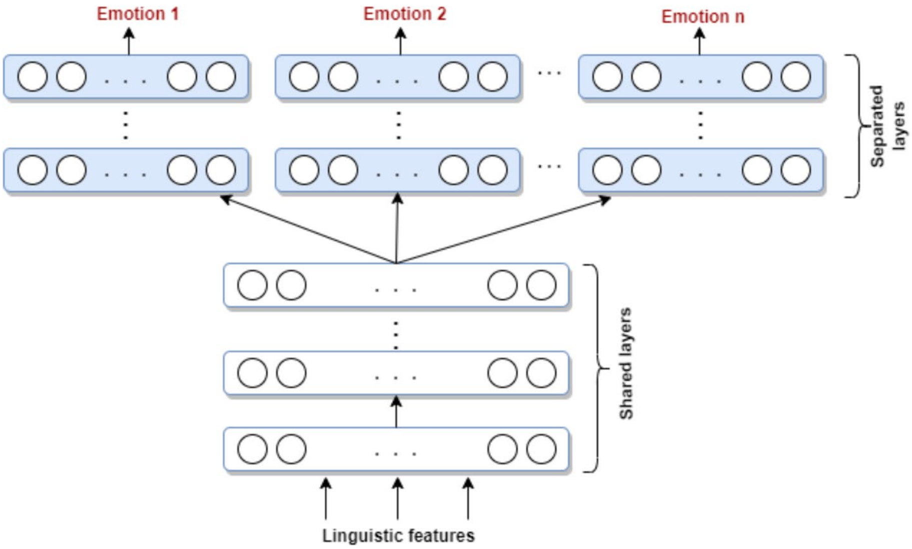

Fig. 5 Labels represented as multiple separated layers, the shared layers are trained with data from all emotions, the emotion specific layers are trained with emotion related data only

In [32], an emotion classifier is used to produce more discriminative emotion embeddings. Initially, the input Mel-spectrogram features from the reference-style audio and those predicted by the proposed TTS model are passed to two reference encoders (explained in Section 4.1) to generate reference embeddings. Both embeddings are then fed to two emotion classifiers, which consist of intermediate fully connected (FC) layers. The output of the second FC layer from both classifiers is considered as the emotion embedding. Apart from the loss of the classifiers, an additional loss function is established between the resulting emotion embeddings from the two classifiers. Similarly, an emotion classifier is also employed in [36] to reduce irrelevant information in the generated emotion embedding from an emotion encoder with reference speech (Mel-spectrogram) as input.

Several other studies [34, 36, 39] that support multiple speakers also suggest utilizing a speaker classifier in addition to the emotion classifier. This approach aims to improved the speaker embedding derived from speaker encoders. Moreover, these studies introduce an adversarial loss between the speaker encoder and the emotion classifier using a gradient reversal layer (GRL) [90]. The purpose of this is to minimize the potential transfer of emotion-related information into the speaker embedding. The GRL technique involves updating the weights of the speaker encoder by utilizing the inverse of the gradient values obtained from the emotion classifier during the training process.

4 Unsupervised approaches

Due to the limited availability and challenges associated with collecting or preparing labeled datasets of expressive speech, as discussed in Section 6, many researchers tend to resort to unsupervised approaches for generating expressive speech. Within these approaches, models are trained to extract speaking styles or emotions from expressive speech data through unsupervised methods. Unsupervised models typically utilize reference speech as an input to the TTS model, which extracts a style or prosody embedding which is then used to synthesize speech resembling the input style reference. In the literature, three primary structures emerge as baseline models for unsupervised ETTS models: including reference encoders, global style tokens, and variational autoencoders, which are explained in the following three sections. In addition, we identify the recent TTS models that utilize in-context learning as another group of unsupervised approaches. The last subcategory under the unsupervised approaches involves other individual approaches. We then provide a general summary of all the unsupervised approaches reviewed in this work in Table 3.

4.1 Direct reference encoding

The main approach, based on a reference or prosody encoder, can be traced back to an early Google paper [74]. The paper suggests using a reference encoder to produce a low-dimensional embedding for a given style

Table 3 Summary of unsupervised approaches. "RE" refers to direct reference encoding, "VAE" refers to approaches based on VAEs, "GST" refers to approaches based on GSTs and "ICL" refers to approaches based on in-context learning. Prosody level "U" stands for utterance, "Se" stands for sentence, "Pr" stands for phrase, "W" stands for word, "Sy" stands for syllable, "Pn" stands for phoneme, "C" stands for character and "F" stands for frame

Ref No	Group	TTS Model	Prosody Level	Ref No	Group	TTS Model	Prosody Level
[58, 75, 91–94]	GST	Tacotron	U	[61, 95–98]	RE	FastSpeech2	Pn
[60, 62, 99–101]	RE	Tacotron2	U	[74, 102–104]	RE	Tacotron	U
[57, 105, 106]	GST	Tacotron2	U	[53, 77, 107]	VAE	Tacotron2	U
[35, 47]	VAE	FastSpeech	Pn	[48, 108]	OTHER	Tacotron2	C
[109, 110]	VAE	CHiVE	Se,W,Sy	[49, 111]	RE	Tacotron2	U,Pn
[112, 113]	RE	Tacotron	Pn	[59]	GST	Tacotron2	Se
[114]	GST	Tacotron2	Pn	[45]	OTHER	Tacotron2	Se
[115]	OTHER	Tacotron-like	Pn	[31]	RE	Tacotron	U,Pn
[55]	RE	Tacotron	Pn,F	[29]	OTHER	Tacotron2	U
[116]	RE	FastSpeech2	W	[68]	VAE	DL-SPSS	U,Pr,W
[38]	OTHER	FastSpeech	U,F	[117]	GST	Transformer TTS	U
[67]	VAE	DL-SPSS	Pr	[37]	RE	Tacotron2	U,Se
[118]	RE	FastSpeech2	U	[19]	RE	Tacotronr2	U
[76]	VAE	Voice-loop	U	[119]	VAE	NTTS	Pn
[120]	RE	Tacotron2	U,F	[17]	GST	FastSpeech2	Se
[121]	OTHER	Tacotron	U	[122]	RE	Tacotron2	Sy
[123]	VAE	Tacotron2	W,Pn	[71]	OTHER	Tacotron2	Pn
[124]	OTHER	Tacotron-like	Pr,W	[33]	RE	Tacotron/2	U,Sy
[51]	RE	Tacotron/2	U	[125]	OTHER	Tacotron	Pn
[126]	VAE	Tacotron	Pn	[52]	RE	FastSpeech	U
[72]	OTHER	Prosody-TTS	Pn	[70]	RE	Fastspeech2	C
[18]	ICL	NaturalSpeech2	F	[127]	RE	CopyCat, Tacotron2	W
[79]	RE	FastSpeech	U,Pn	[30]	OTHER	GraphPB	U,Pr,W
[128]	OTHER	FastSpeech2	U,Pn	[129]	VAE	DurIAN	Se
[130]	VAE	Tacotron-like	U,Pn	[44]	RE	AlignTTS	Pn
[131]	VAE	Tacotron-like	Se	[132]	OTHER	AdaSpeech 3	Pn
[63]	RE	Transformer TTS	U,Pn	[40]	OTHER	Transformer TTS	U,W
[54]	OTHER	Tacotronr2	W	[20]	OTHER	Tacotronr2	Pn
[21]	RE	InstructTTS	Se	[22]	ICL	VALL-E	F
[23]	RE	VITS	U,F	[24]	VAE	FastSpeech 2	U,W
[25]	ICL	Voicebox	F	[46]	GST	Tacotron2	W
[73]	GST	Tacotron	Sy	[133]	GST	Tacotron	Pn
[134]	RE	FastSpeech2	U,Pn	[135]	OTHER	DL-SPSS	Se,W,Sy,Pn
[136]	GST	FastSpeech2	U	[50]	GST	FastSpeech2	U,Se,Sy
[137]	GST	Tacotron	Pn	[138]	OTHER	DL-SPSS	F
[56]	RE	DL-SPSS	U				

reference audio, which is called a prosody embedding. This encoder takes spectrograms as input to represent the reference audio. The generated prosody embedding is then concatenated with the text embedding derived from the text encoder of a Seq2Seq TTS model such as Tacotron [3, 4]. Figure 6 shows reference encoder integrated to the TTS model.

Various features have been employed in the literature as inputs for the reference encoder. For example, in the work [85], MFCC features extracted using the openSMILE toolkit [139] are fed into one of the encoders within its style extraction model, which is composed of a multi-modal dual recurrent encoder (MDRE). In another study [31], the reference encoder is proposed as a ranking function model, aimed at learning emotion strength at the phoneme level. This model leverages the OpenSMILE toolkit to extract 384-dimensional emotion-related features from segments of reference audio, derived

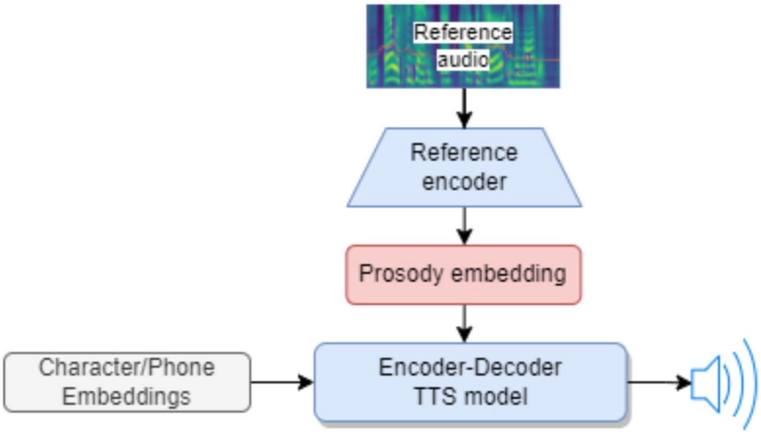

Fig. 6 Baseline model (1) for unsupervised approaches: Direct reference encoding integrated to an encoder-decoder TTS model

using a forced alignment model for phoneme boundaries. Furthermore, in work [63], a word-level prosody embedding is generated. This is achieved by extracting phoneme-level F0 features from reference speech using the WORLD vocoder [140] and an internal aligner operating with the input text.

A prosody-aware module is proposed in [37] which extracts other prosody-related features. The prosody-aware module consists of an encoder, an extractor, and a predictor. The encoder receives the three phoneme-level features including logarithmic fundamental frequency (LF0), intensity, and duration from the extractor as input and generates the paragraph prosody embedding with the assistance of an attention unit. Simultaneously, the predictor is trained to predict these features at inference time based on the input text embedding only.

In Daft-Exprt TTS model [118], the prosody encoder receives pitch, energy and spectrogram as input. The prosody encoder then uses FiLM conditioning layers [141] to carry out affine transformations to the intermediate features of specific layers in the TTS model. A slightly modified version of the FastSpeech2 [6] model is utilized in this work where the phoneme encoder, prosody predictor and the decoder are the conditioned components. The prosody predictor is similar to the variance adaptor of FastSpeech2 but without the length regulator, and it estimates pitch, energy and duration at phoneme-level.

A pre-trained Wav2Vec model [142] has also been utilized for extracting features from the reference waveform. These features serve as input to the reference encoders of the proposed Emo-VITS model [23], which integrates an emotion network into the VITS model [143] to enhance expressive speech synthesis. In fact, the emotion network in the Emo-VITS model comprises two reference encoders. The resulting emotion embeddings from these encoders are then combined through a feature fusion module that employs an attention mechanism. Wav2vec 2.0-derived features from the reference waveform in this work are particularly suitable for attention-based fusion and contribute to reducing the textual content within the resulting embeddings [23].

In contrast, [60] proposes a an image style transfer module to generate input for reference encoder. The concept of image style transfer involves altering the artistic style of an image from one domain to another while retaining the image's original content [144]. In specific research, the style reconstruction module from VGG-19 [145], a deep neural network primarily used for image classification, is employed to extract style-related information from the Mel-spectrogram used as input image. Subsequently, the output of this module is fed into the reference encoder to generate the style embedding.

4.2 Latent features via variational auto-encoders

The goal of TTS models under this is to map input speech from the higher dimensional space to a well-organized and lower-dimensional latent space utilizing variational auto-encoders (VAEs) [146]. VAE is a generative model that is trained to learn the mapping between observed data x and continuous random vectors z in an unsupervised manner. In detail, VAEs learn a Gaussian distribution denoted as the latent space from which the latent vectors representing the given data x can be sampled. A typical variational autoencoder consists of two components. First, the encoder learns the parameters of the z vectors (latent distribution), namely the mean $\mu(x)$ and variance $\sigma^2(x)$, based on the input data x. Second, the decoder regenerates the input data x based on latent vectors z sampled from the distribution learned by the encoder. In addition to the reconstruction loss between the model input and the data, variational autoencoders are also trained to minimize a latent loss, which ensures that the latent space follows a Gaussian distribution.

Utilizing VAEs in expressive TTS models as shown by Fig. 7, allows for mapping the various speech styles within the given dataset to be encoded as latent vectors, often referred to as prosody vectors, within this latent space. During inference, these latent vectors can be sampled directly or with the guidance of reference audio from the VAE's latent space. Furthermore, the latent vectors offer the advantage of disentangling prosody features, meaning that some specific dimensions of these vectors independently represent single prosody features such as pitch variation or speaking rate. Disentangled prosody features allow for better prosody control via manipulating the latent vectors with different operations such as interpolation and scaling [77]. The two early papers, [76, 77], can be regarded as the baseline for latent feature-based approaches. The former study [76] introduces VAE within the VoiceLoop model [147], while the latter [77] incorporates VAE into Tacotron2 [4] as an end-to-end TTS model for expressive speech synthesis.

In the same direction of modeling the variation of the prosodic features in expressive speech, studies [109, 110] propose a hierarchical structure for the baseline variational autoencoder, known as Clockwork Hierarchical Variational AutoEncoder (CHiVE). Both the encoder and decoder in the CHiVE model have several layers to capture prosody at different levels based on the input text's hierarchical structure. Accordingly, linguistic features are also used alongside acoustic features as input to the model's encoder. The model's layers are dynamically clocked at specific rates: sentence, words, syllables, and phones. The encoder hierarchy goes from syllables to the sentence level, while the decoder hierarchy is in the reversed order. The CHiVE-BERT model in [110], differs from the main model in [109] as it utilizes BERT [148] features for input text at the word-level. Since the features extracted by the BERT model incorporate both syntactic and semantic information from a large language model, CHiVE-BERT model is expected to have improved the prosody generation.

Other studies [24, 53] propose Vector-Quantized Variational Auto-Encoder (VQ-VAE) to achieve discretized latent prosody vectors. In vector quantization (VQ) [149], latent representations are mapped from the prosody latent space to a codebook of a limited number of prosody codes. Specifically, during training, the nearest neighbor lookup algorithm is applied to find the nearest codebook vector to the output of the reference encoder and used to condition TTS decoder. To further improve the quality of latent prosody vectors and consequently the expressiveness of the generated speech, Diff-Prosody[24] proposes a diffusion-based VQ-VAE model. In the proposed model a prosody generator that utilizes a denoising diffusion generative adversarial networks (DDGANs) [150] is trained to generate the prosody latent vectors based only on text and speaker information. At inference time, the prosody generator is used to produce

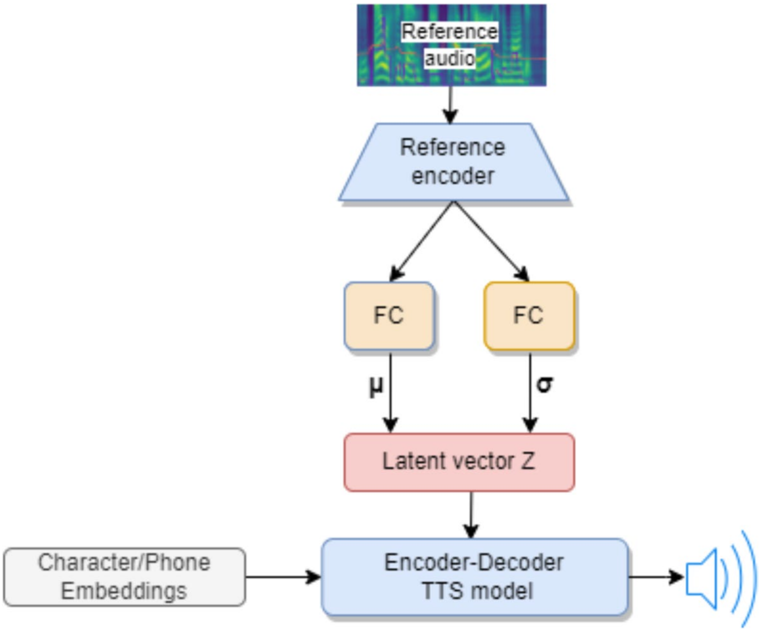

Fig. 7 Baseline model (2) for unsupervised approaches: variational autoencoder (VAE) integrated to encoder-decoder TTS model. VAE learns latent space variables mean $\mu(x)$ and variance $\sigma^2(x)$ and samples latent prosody vector(z) from the learned prosody space

prosody vectors based on input text and with no need for an audio reference which improves both quality and speed of speech synthesis.

While most of the studies in this category follow the baseline model and use mel-spectrograms to represent the reference audio, other studies extract correlated prosody features as input to the VAE. For instance, frame-level F0, energy, and duration features are extracted from the reference speech as basic input for the hierarchical encoder of the CHiVE model [109]. These same features are also used as input for the VAE encoder in work [35], but at the phoneme level. In work [68], multi-resolution VAEs are employed, each with acoustic and linguistic input vectors. The acoustic feature vectors for each encoder include 70 mel-cepstral coefficients, log F0 value, a voiced/unvoiced value, and 35 mel-cepstral analysis aperiodicity measures.

4.3 Global Style Tokens

The Global Style Tokens (GST) approach for expressive synthesis was first introduced in [75]. The paper proposes a framework to learn various speaking styles (referred to as style tokens) in an unsupervised manner within an end-to-end TTS model. The proposed approach can be seen as a soft clustering method that learns soft style clusters for expressive styles in an unlabeled dataset. In detail, GST, as shown by Fig. 8, extends the approach introduced in [74] by passing the resulting style embedding from the reference encoder to an attention unit, which functions as a similarity measure between the style embedding and a bank of randomly initialized tokens. During training, the model learns the style tokens and a set of weights, where each style embedding is generated via a weighted sum of the learned tokens. In fact, the obtained weights represent how each token contributes to the final style embedding. Therefore, each token will represent a single style or a single prosody-related feature, such as pitch, intensity, or speaking rate.

At inference time, a reference audio can be passed to the model to generate its corresponding style embedding via a weighted sum of the style tokens. Alternatively, each individual style token can be used as a style embedding. In addition, GSTs offer an enhanced control over the speaking style through various operations. These include manual weight refinement, token scaling with different values, or the ability to condition different parts of the input text with distinct style tokens.

The GST-TTS model can be further enhanced by modeling different levels of prosody to improve both expressiveness and control over the generated speech. For instance, [46] proposes a fine-grained GST-TTS model where word-level GSTs are generated to capture local style variations (WSVs) through a prosody extractor. The WSV extractor consists of a reference encoder and a style token layer, as described in [75], along with an attention unit to produce the word-level style token

In [133] a hierarchical structure of multi-layer GSTs with residuals is proposed. The model employs three GST layers, each with 10 tokens, resulting in a better interpretation of the tokens of each level. Upon tokens analysis, it was found that the first-layer tokens learned speaker representations, while the second-layer tokens captured various speaking style features such as pause position, duration, and stress. The third-layer tokens,

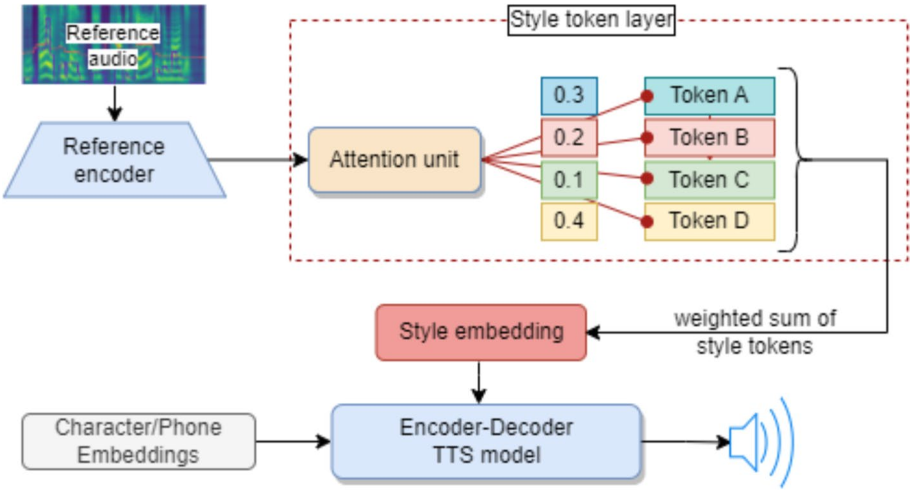

Fig. 8 Baseline model (3) for unsupervised approaches: Global Style Tokens (GSTs) integrated to encoder-decoder TTS model. Style token layer learns a single weight for each token and generates the style embedding from summation of all the tokens multiplied by their corresponding weights

however, were able to generate higher-quality samples with more distinct and interpretable styles. Similarly, in [50], a multi-scale GST extractor is proposed to extract speaking style at different levels. This extractor extracts style embeddings from the reference mel-spectrogram using three style encoders at global, sentence, and sub word levels, and combines their outputs to form the multi-scale style embedding.

With only a small portion of the training dataset labeled with emotions, [26] proposes a semi-supervised GST model for generating emotional speech. The model applies a cross-entropy loss between the one-hot vectors representing the emotion labels and the weights of GSTs, in addition to the GST-TTS reconstruction loss. The semi-GST model is trained on a dataset in which only 5% of the samples are labeled with emotion classes, while the rest of the dataset is unlabeled. After training, each style token represents a specific emotion class from the training dataset and can be used to generate speech in the corresponding emotion.

Furthermore, in [92], a speech emotion recognition (SER) model is proposed with the GST-TTS to generate emotional speech while acquiring only a small labeled dataset for training. The paper formulates the training process as reinforcement learning (RL). In this framework, the GST-TTS model is treated as the agent, and its parameters serve as the policy. The policy aims to predict the emotional acoustic features at each time step, where these features represent the actions. The pre-trained SER model then provides feedback on the predicted features through emotion recognition accuracy, which represents the reward. The policy gradient strategy is employed to perform backpropagation and optimize the TTS model to achieve the maximum reward.

In contrast, the Mellotron model [114] introduces a unique structure for the GSTs, enabling Mellotron to generate speech in various styles, including singing styles, based on pitch and duration information extracted from the reference audio. This is achieved by obtaining a set of explicit and latent variables from the reference audio. Explicit variables (text, speaker, and F0 contour) capture explicit audio information, while latent variables (style tokens and attention maps) capture the latent characteristics of speech that are hard to extract explicitly.

4.4 Approaches based on in-context learning

These is a group of recent TTS models that are trained on a large amounts of data using in-context learning strategy. During in-context learning (also called prompt engineering), the model is trained to predict missing data based its context. In other words, the model is trained with a list of input-output pairs formed in a way that represents the in-context learning task. After training, the model should be able to predict the output based on a given input.

For the TTS task, the provided style reference (referred to as prompt) is considered as part of the entire utterance to be synthesized. The TTS model training task is to generate the rest of this utterance following the style of the provided prompt as shown by Fig. 9. By employing this training strategy, recent TTS models such as VALL-E [22], NaturalSpeech 2 [18], and Voicebox [25] are capable of producing zero-shot speech synthesis using only a single acoustic prompt. Furthermore, these models demonstrate the ability to replicate speech style/emotion from a provided prompt [18, 22] or reference [25] to the synthesized speech.

In VALL-E [22], a language model is trained on tokens from Encodec [151], and the input text is used to condition the language model. Specifically, the Encodec model tokenizes audio frames into discrete latent vectors/codes, where each audio frame is encoded with eight codebooks. VALL-E employs two main models: the first one is an auto-regressive (AR) model that predicts the first code of each frame, and the second is non-auto-regressive (NAR) model that predicts the other seven codes of the frame.

Instead of discrete tokens used in VALL-E, NaturalSpeech 2 [18] represents speech as latent vectors from a neural audio codec with residual vector quantizers. The latent vectors are then predicted via a diffusion model, conditioned on input text, pitch from a pitch predictor, and input speech prompt.

Another example of in-context training is Voicebox [25] which is a versatile generative model for speech trained on a large amount of multilingual speech data. The model

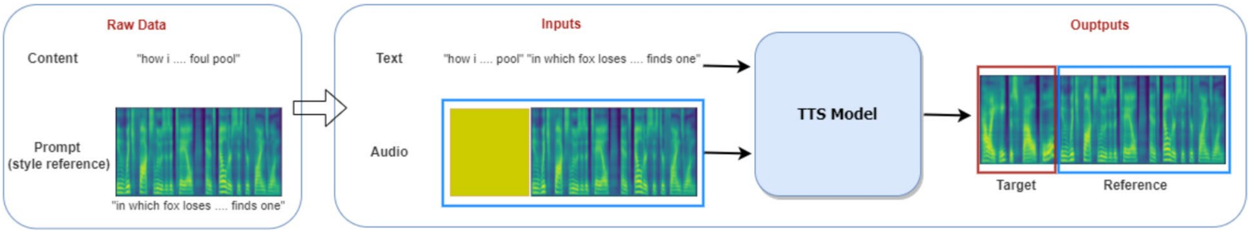

Fig. 9 Utilizing in-context learning for training speech synthesis models, adapted from [25]

is trained on a text-guided speech infilling task, which gives it the flexibility to perform various speech tasks such as zero-shot TTS, noise removal, content editing, and diverse speech sampling. Voicebox is modeled as a non-autoregressive (NAR) flow-matching model with the ability to consider future context.

4.5 Other approaches

This category containes reviewed papers that propose individual techniques or methods which cannot be categorized under any of the previously mentioned unsupervised approaches. For instance, in [121], a neural encoder is introduced to encode the residual error between the predictions of a trained average TTS model and the ground truth speech. The encoded error is then used as a style embedding that conditions the decoder of the TTS model to guide the synthesis process. Raitio and Seshadri [128] improves prosody modeling of FastSpeech2 model [6] with an additional variance adaptor for utterance-wise prosody modeling.

As context information is strongly related to speech expressivity, [45] proposes using multiple self-attention layers in Tacotron2 [4] encoder to better capture the context information in the input text. The outputs of these layers in the encoder are combined through either direct aggregation (concatenation) or weighted aggregation using a multi-head attention layer. Additionally, there are some papers that propose using only input text to obtain prosody-related representations/embeddings without any style references, and those are further discussed in Section 5.2.4.

5 Main challenges of ETTS models

In this section, we list and explain the most important challenges that face expressive TTS models and the main solutions that have been proposed in the literature to overcome these challenges. We then provide a summary of papers addressing each challenge in Table 5.

5.1 Irrelevant information leakage

One main problem in unsupervised approaches that rely on having a style reference or a prompt, is the leakage of irrelevant information, like speaker or text related information, into the generated style or prosody embedding. This irrelevant information within the speech style can lead to degradation in the quality of the synthesized speech. As a result, many studies have investigated this problem, and several solutions have been proposed as outlined below.

5.1.1 Adversarial training

Adversarial training [90] is one of the widely used techniques to confront the information leakage problem. Typically, a classifier is trained to distinguish the type of unwanted information (such as speaker or content information) that is leaking from the prosody reference audio into the generated prosody embedding. During the training process, the weights of the employed prosody encoder/extractor from the reference audio are modified with gradient inversion of the proposed classifier. In other words, the classifier penalizes the prosody encoder/extractor for any undesired information in its output. A gradient reversal layer (GRL) is usually used to achieve the inversion of the classifier gradients.

Several studies utilize adversarial training to prevent the flow of either speaker or content-related information from the given reference audio to the resulting prosody embedding. For instance, the VAE-TTS model proposed in [47] learns phoneme-level 3-dimensional prosody codes. The VAE is conditioned on speaker and emotion embeddings, besides the tone sequence and mel-spectrogram from the reference audio. Adversarial training using a gradient reversal layer (GRL) is applied to disentangle speaker and tone from the resulting prosody codes. Similarly, adversarial training is introduced to the style encoder of the cross-speaker emotion transfer model proposed in [19] to learn a speaker-independent style embedding, where the target speaker embedding is provided from a separate speaker encoder.

The STYLER model in [97] employs multiple style encoders to decompose the style reference into several components, including duration, pitch, speaker, energy, and noise. Both channel-wise and frame-wise bottleneck layers are added to all the style encoders to eliminate content-related information from the resulting embeddings. Furthermore, as noise is encoded individually by a separate encoder in the model, other encoders are constrained to exclude noise information by employing either domain adversarial training or residual decoding.

In [111], prosody is modeled at the phone-level and utterance-level by two separate encoders. The first encoder consists of two sub-encoders: a style encoder and a content encoder, besides two supporting classifiers. The first classifier predicts phone identity based on the content embedding, while the other classifier makes the same prediction but based on the style embedding. The content encoder is trained via collaborative training with the guidance of the first classifier, while adversarial training is used to train the style encoder, utilizing the second classifier.

On the other hand, [102] proposes adversarial training for the style reference by inverting the gradient of an automatic speech recognition (ASR) model. The proposed model introduces a shared layer between an ASR and a reference encoder-based model. Specifically, a single BiLSTM layer from the listener module of a pre-trained ASR model serves as the prior layer to the reference encoder. The process starts by passing the reference Mel-spectrogram to the shared layer to produce the shared embedding as input to both the reference encoder and the ASR model. A gradient reversal layer (GRL) is employed by the ASR model to reverse its gradient on the shared layer. Accordingly, the reference encoder parameters are modified so that the ASR model fails to recognize the shared embedding, and thus content leakage to the style embedding from the reference encoder is reduced.

5.1.2 Prosody classifiers

This is a supporting approach used by some studies to produce more discriminative prosody embeddings by passing them to a prosody classifier. This method can be applied when the training data is labeled with emotion or style labels. In the two consecutive studies [32, 34] from the same research group, an auxiliary reference encoder is proposed and located after the decoder of the baseline TTS model [74]. The two reference encoders in the model are followed by emotion classifiers to further enhance the discriminative nature of their resulting embeddings. However, the emotion embedding that is passed to the TTS model is the output of an intermediate hidden layer of the classifiers. In addition to the classification loss, an additional style loss is also applied between the two emotion embeddings from the two employed emotion classifiers.

In [36], alongside the text encoder, two encoders are introduced to generate embeddings for speaker and emotion from a reference audio. To further disentangle emotion, speaker, and text information, both speaker and emotion encoders are supported with a classifier to predict speaker and emotion labels, respectively. Similarly, in paper [39], a model with two encoders and two classifiers is proposed to produce disentangled embeddings for speakers and emotions from a reference audio. However, the paper claims that some emotional information is lost during the process of disentangling speaker identity from the emotion embedding. As a result, an ASR model is introduced to compensate for the missing emotional information. The emotion embedding is incorporated within a pre-trained ASR model through a global context (GC) block. This block extracts global emotional features from the ASR model's intermediate features (AIF). Subsequently, a prosody compensation encoder is utilized to generate emotion compensation information from the output of the AIF layer, which is then added to the emotion encoder output.

5.1.3 Information bottleneck

The information bottleneck is a technique used to control information flow via a single layer/network. It helps prevent information leakage as it projects input into a lower dimension so that there is not enough capacity to model additional information and only important information is passed through it. In other words, the bottleneck can be seen as a down-sampling and up-sampling filter that restricts its output and generates a pure style embedding. Several prosody-reference based approaches, as in [86, 93, 97, 101, 130], have employed this technique to prevent the flow of speaker or content-related information from the reference audio to the prosody embedding.

In [93], a bottleneck layer named sieve layer is introduced to the style encoder in GST-TTS to generate pure style embedding. Similarly, in the multiple style encoders model STYLER [97], each encoder involves a channel-wise bottleneck block of two bidirectional-LSTM layers to eliminate content information from encoders' output. Another example is the cross-speaker-style transfer Transformer-TTS model proposed in [86] with both speaker and style embeddings as input to the model encoder. The speaker-style-combined output from the encoder is then passed to a prosody bottleneck sub-network, which produces a prosody embedding that involves only prosody-related features. The proposed bottleneck sub-network consists of two CNN layers, a squeeze-and-excitation (SE) block [152], and a linear layer. The encoder output is then concatenated with the resulting prosody embedding and used as input to the decoder.

The Copycat TTS model [130] is a prosody transfer model via VAE. The model applies three techniques to disentangle the source speaker information from the prosody embedding. One of these techniques is to use a temporal bottleneck encoder [153] within the reference encoder of the model. The prosody embedding that is sampled from the latent space is passed to the bottleneck to reduce speaker identity-related information in the prosody embedding before it flows to the model decoder. Similarly, the model proposed in [101] produces a style embedding with less irrelevant style information by adding a variational information bottleneck (VIB) [154] layer to the reference encoder. The idea behind this layer is to introduce a complexity constraint on mutual information (MI) between the reference encoder input and output so that it only flows out style-related information.

5.1.4 Instance normalization

Batch normalization (BN), first introduced in [155], is utilized in deep neural networks to accelerate the training process and increase its stability. Essentially, a batch normalization layer is added before each layer in deep neural networks to adjust the means and variances of the layer inputs, as illustrated by Eq. (1):

$$IN(x) = \gamma \left[\frac{x - \mu(x)}{\sigma(x)} \right] + \beta \tag{1}$$

where γ, β are affine parameters learned from data and μ, σ are the mean and standard deviation which are calculated for each feature channel across the batch size. Instance normalization (IN) also follows equation (1); however, it calculates means and variances across spatial dimensions independently for each channel and each sample (instance). In the field of computer vision, stylization approach is significantly improved by replacing (BN) layers with (IN) layers [156]. Consequently, researchers in the expressive speech field have started to apply IN to extract better prosody representations. For example, an instance normalization (IN) layer is used at the reference encoder in [130], at the prosody extractor in [93], and at the style encoder in [96] to remove style/prosody irrelevant features (such as speaker identity features) and enhance the learned style/prosody embedding.

5.1.5 Mutual information minimization

For a pair of random variables, mutual information (MI) is defined as the information obtained on one random variable by observing the other. Specifically, if X and Y are two variables, then $MI(X; Y)$ shown by Venn diagram in Fig. 10, can be seen as the KL-divergence between the joint distribution (P_{XY}) and the product of the marginals (P_X, P_Y) as in equation (2). If the two random variables X and Y represent linguistic and style vectors, applying MI minimization between these two vectors helps to produce style vectors with less information from the content vector.

$$MI(X; Y) = DL_{KL}(P_{(X,Y)} \| P_X \otimes P_Y) \tag{2}$$

For example, in [137], the Mutual Information Neural Estimation algorithm (MINE) [157] is employed to estimate the mutual information between the content and style vectors. The algorithm uses a neural network that is trained to maximize the lower bound of the mutual information between the style and content vectors. Simultaneously, the TTS model aims to minimize the reconstruction loss, making the overall problem a max-min problem. Alternatively, in [21], the CLUB method [158], which computes an upper bound as the MI estimator, is used to prevent the leakage of speaker and content information into the style embedding.

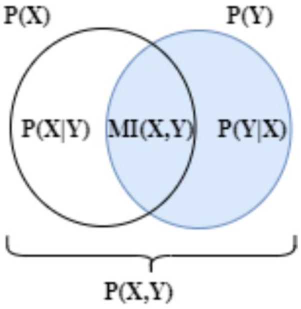

Fig. 10 Venn diagram of two random variables *X* and *Y* where *P*(*X*) and *P*(*Y*) represent their entropies, *P*(*X*|*Y*) is the conditional entropy of *X* given *Y* and *P*(*Y*|*X*) is the conditional entropy of *Y* given *X*, *H*(*X*,*Y*) is the joint entropy of *X* and *Y* and *MI*(*X*,*Y*) is their mutual information

A new approach is proposed in [117] for MI estimation and minimization to reduce content/speaker information transfer to the style embedding in a VAE based approach. Typically, the model needs to estimate MI between latent style embeddings and speaker/content embeddings. To avoid the exponentially high statistical variance of the finite-sampling MI estimator, the paper suggests using a new algorithm for information divergence named Rényi divergence. Two variations from the Rényi divergence family are proposed, including minimizing the Hellinger distance and minimizing the sum of Rényi divergences.

5.1.6 Wav2Vec features

Wav2Vec [142] model converts speech waveform into context-dependent vectors/features. The model is trained via self-supervised or in-context training algorithms which are explained in Section 4.4. Features generated by wav2vec and similar models such as HuBERT [159] provide better representations of speech and its lexical and non-lexical information. Therefore, these models are utilized nowadays in different speech processing tasks such as speech recognition, synthesis, and downstream emotion detection.

Some studies such as [23, 120] use Wav2vec 2.0 as a feature extractor to provide input to the reference encoder instead of spectrum features or raw audio waveform. Figure 11 illustrates the framework of the wav2vec technique and how it is utilized as a feature extractor with TTS models. The wav2vec model converts the continuous audio features into quantized finite set of discrete representations called tokens. This is done using a quantization module that maps the continuous feature vectors into a discrete set of tokens from a learned codebook. As those tokens are more abstract, they reduce the complexity of the features by retaining important features while filtering out all the irrelevant information. Because of that abstraction, it is harder to reconstruct audio from the wav2vec features, which means leakage of linguistic content into feature vectors is significantly lower compared to other features such as MFCCs.

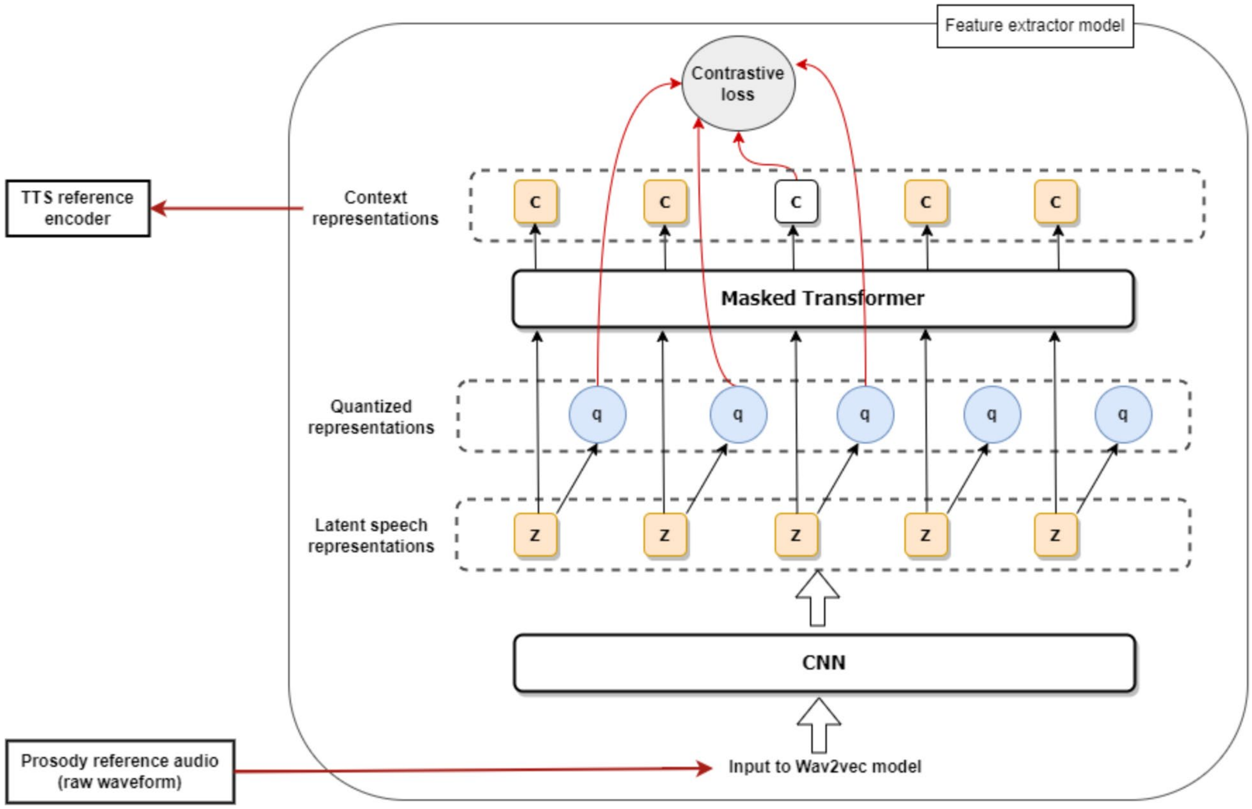

Fig. 11 The general framework of wav2vec technique and its utilization as a feature extractor for generating speech representations as input to the TTS model

5.1.7 Orthogonality loss

Studies [34, 39] propose a model with two separate encoders to encode speaker and emotion information through speaker and emotion classification loss, along with gradient inversion of the emotion classification loss in the speaker encoder. Additionally, to disentangle the source speaker information from the emotion embedding, the emotion embedding is made orthogonal to the speaker embedding with an orthogonality loss shown in equation (3). An ablation study in [34] showed that applying an orthogonality constraint helped the encoders learn both speaker-irrelevant emotion embedding and emotion-irrelevant speaker embedding.

$$L_{orth} = \sum_{i=1}^{n} \|S_i - e_i\|_F^2 \tag{3}$$

where $\|.\|_F$ is the Frobenius norm, e_i is the emotion embedding and s_i is the speaker embedding.

5.2 Inference without reference audio

A main drawback of the unsupervised approaches (Section 4) is that they require a reference audio for the desired prosody or style of the generated speech. However, prosody references are not always available for the desired speaker, style, or text. Besides, using prosody reference introduces the leakage problem as discussed in Section 5.1. As a result, different techniques have been proposed that enable unsupervised expressive speech synthesis without prosody references. Some techniques utilize the reference audio at training phase while at inference phase speech synthesis can be done with or without a reference audio. Other techniques depend on input text only to generate prosody embedding at both training and inference phases. In the following three sections, we will describe techniques for inference without reference audio applied with each of the three main unsupervised ETTS approaches. In Section 5.2.4, we will discuss some ETTS approaches that are based on text only. Then in Table 4, we summarize main approaches that are used to extract text-based features with related papers links.

5.2.1 Direct reference encoding without reference audio

In several studies, prosody predictors are trained jointly with the proposed reference encoder to bypass the requirement for reference audio at inference time. The prosody predictors are trained to predict either the

Table 4 Applied models and techniques in literature for extracting features from textual input of the TTS model with papers' links in which they are applied. Extracted features are utilized in the ETTS model for three purposes: inference in reference-based ETTS models when lacking reference audio, inputs to ETTS models trained to be based on text only or as additional features to the ETTS model

Model/method	**Utilized for:**		
	Inference without reference audio	**ETTS based on text only**	**Additional ETTS features**
BERT language model	[35, 44, 46, 50, 59, 73, 129]	[29, 40, 54]	[62, 87, 100, 110, 127, 131, 136, 138]
ELECTRA language model		[125]	
ELMo language model		[83]	
RoBERTa language model			[21, 70]
XLNet language model	[17, 50]		
(GPT)-3 language model		[64]	
Parsing trees	[129]		
Prosody boundaries in text			
Constituency trees			[131]
Sentiment analysis model		[30]	
Stanford Sentiment Parser		[135]	
Syntax-related features (such as POS: part of speech)			[127]
Word emotion lexicon		[40]	
Term Frequency-Inverse Document Frequency (TF-IDF) (TF-IDF)			[99]
Character/phoneme embedding	[20, 33, 37, 44, 47, 48, 63, 71, 72, 91, 94–96, 103, 111]		

prosody embeddings generated by reference encoders [50, 96, 111, 116], or the acoustic features used as input to reference encoders [37, 63]. As input to these prosody predictors, most studies utilize the phoneme embeddings [37, 63, 96, 111].

Alternatively, features extracted from input text can also be used as input for prosody predictors. In [50], the prosody predictor has a hierarchical structure that utilizes contextual information at both the sentence and paragraph levels to predict prosody embeddings. The input features for this predictor are in the form of 768-dimensional phrase embeddings extracted by the pre-trained language model XLNet [160]. Sentence embeddings are initially predicted from the input features using an attention network. Then a second attention network is used to predict the paragraph-level prosody embedding.

Furthermore, in [33], emotion is modelled at three levels: global, utterance, and syllable (local). The model employs three prosody encoders, each with a predictor trained to predict the corresponding prosody embedding based on input text. The global-level predictor functions as an emotion classifier, where the output of its final softmax layer serves as the global emotion embedding. The emotion label's embedding is used as the ground truth for this emotion classifier. Both the utterance and local prosody encoders receive level-aligned mel-spectrograms as input and produce utterance prosody embedding and local prosody strength embedding, respectively. Similarly, two prosody predictors are used to predict utterance and local-level embeddings based on the output from the text encoder of the TTS model.

In contrast, the prosody predictor proposed in paper [44] learns multiple mixed Gaussian distributions model (GMM) for prosody representations. Therefore, the final outputs of the prosody predictor involve three parameters: mean, variance, and weight of multiple mixed Gaussian distributions from which prosody representations can be sampled at inference time. As input, the predictor receives two phoneme-level sequences including embeddings from the text encoder and embeddings from a pre-trained language model. Similar work is proposed in [95] where only phoneme embeddings are used as input to the prosody predictor. GMM in both studies is modeled via the mixture density network [161].

5.2.2 VAE-based approaches without reference audio

Sampling from the latent space without reference audio results in less controllability of style. In addition, it can also introduce naturalness degradation and inappropriate contextual prosody with regard to the input text [68, 129]. Therefore, to avoid sampling the latent space without a reference, authors of [131] proposed utilizing the same prosody embedding of the most similar training sentence to input sentence at inference time. The selection process is based on measuring cosine similarity between sentences' linguistic features. Three methods are proposed for extracting sentence linguistic information including (1) calculating the syntactic distance between words in the sentence using constituency trees [162], (2) averaging the contextual word embeddings (CWE) for the words in the sentence using BERT, and (3) combining the previous two methods.

Other studies approach the problem in alternative ways, seeking to enhance the sampling process either through refining the baseline model structure or by incorporating text-based components into the baseline. Regarding the improvement of the baseline structure, study [68] suggests the combination of multiple variational autoencoders to generate latent variables at three distinct levels: utterance-level, phrase-level, and word-level. Furthermore, they apply a conditional prior (CP) to learn the latent space distribution based on the input text embedding. To account for dependencies within the input text, they employ Autoregressive (AR) latent converters to transform latent variables from coarser to finer levels.

An alternative approach is proposed in [126] by replacing the conventional VAE encoder with a residual encoder that leverages phoneme embedding and a set of learnable free parameters as inputs. With this modified structure, the model learns a latent distribution that represents various prosody styles for a specific sentence (i.e., the input text), in addition to capturing potential global biases within the applied dataset (represented by the free parameters). At the same time, with this modification, the problem of speaker and content leakage into prosody embedding is addressed.

Various studies propose training a predictor for the latent prosody vectors based on features extracted from the input text [35, 47]. The proposed model in [47] generates fine-grained prosody latent codes of three dimensions at phoneme-level. These prosody codes are then used to guide the training process of a prosody predictor that receives phoneme embeddings as input, in addition to emotion and speaker embeddings as sentence-level conditions. In [35], the predicted mean values of the latent space distribution are employed as prosody codes. Similarly, a prosody predictor is trained to predict these prosody codes using two text-based inputs, including sentence-level embeddings from a pre-trained BERT model and contextual information considering BERT embeddings of a few of surrounding k sentences given the current sentence.

Alternatively, study [129] proposed training a sampler, i.e., Gaussian parameters, to sample the latent space using features extracted from the input text. Three different structures are investigated for the sampler based on the input features it receives. The applied text-based features include BERT representations of a sentence (semantic information), the parsing tree of the sentence (syntactic information) after it is fed to a graph attention network, and the concatenation of outputs from the previous two samplers.

5.2.3 GST-based approaches without reference audio

There are GST-TTS models that utilize text-based features from pre-trained language models such as BERT to guide expressive speech synthesis at inference time without a reference. In [59], the training dataset is labeled with short phrases that describe the style of the utterance and are known as style tags. A pre-trained Sentence BERT (SBERT) model is used to produce embeddings for each style tag as input to a style tag encoder. The style embedding from the GST-TTS model is used as ground truth for the style tag encoder. During inference, either a reference audio or a style tag can be used to generate speech.

Alternatively, pre-trained language models are used to extract features from input text and train a prosody predictor to predict the style embedding based on these text-based features [17, 46, 50, 73, 91, 94]. In [94], the baseline model [75] is extended with a prosody predictor module that extracts time-aggregated features from the output of the baseline text encoder. Two pathways are suggested for the targets of the predictor output: either using the weights of the GSTs or the final style embedding. Similarly, in [73], two prosody predictors are investigated, using different inputs from a pre-trained multi-language BERT model. While the first predictor utilizes BERT embeddings for the sub-word sequence of input text, the other predictor employs only the CLS token from the sentence-level information extracted by the BERT model. Both inputs provide rich information for the predictors to synthesize prosodic speech based solely on input text.

The multi-scale GST-TTS proposed in [50] which employs three style encoders, also introduces three style predictors that employ hierarchical context encoders (HCE). The input to the first predictor is the BERT sub word-level semantic embedding sequence. The attention units in the HCE, however, are used to aggregate the resulting context embedding sequence from lower level as input to higher-level predictors. Additionally, the output of higher-level predictor is used to condition the lower-level predictor. BERT embeddings are also used in

[46] but at word-level and are passed as input to the proposed prosody predictor. The style embedding which is generated via word-level GSTs is used to guide the prosody predictor during model training.

A Context-aware prosody predictor is proposed in [17] which considers both text-side context information and speech-side style information from preceding speech. This predictor comprises two hierarchical components: a sentence encoder and a fusion context encoder. The context-aware input to the predictor includes word-level embeddings from XLNet [160] for each word in the current sentence, as well as the N preceding and following sentences. The sentence encoder focuses on learning low-level word meanings within each sentence, while the fusion context encoder captures high-level contextual semantics between the sentences. Additionally, style embeddings from previous sentences are integrated into the fusion context encoder input to account for speech-side information.

In [91] Speech emotion recognition model (SER) is employed as a style descriptor to learn the implicit connection between style features and input text. Deep style features for both synthesized speech and reference speech are obtained from a small intermediate fully connected layer of a pre-trained SER model during training. The extracted style features are compared where an additional loss is introduced to the GST-TTS model loss. At inference time only text is used to synthesize expressive speech.

5.2.4 ETTS approaches based only on text

This category involves approaches that depend solely on input text to obtain prosody-related representations/embeddings during TTS model training. Several features related to speech prosody have been proposed by various studies for extraction from input text and subsequent transmission to a DNN-based module to generate prosody representations. For instance, the features extracted by the pre-trained language models can capture both semantic and syntactic relationships with the input text, making them effective representations for prosody. In [83], input text word-level embeddings are extracted by the Embeddings from Language Models (ELMo) model [163] and used to generate context-related embeddings via a context encoder. Similarly, in [29], BERT is employed to extract embeddings for utterance sentences and pass them to a specific context-encoder to aggregate these embeddings and form a final context vector.

Other studies, such as [30, 40, 54], utilize graph representations of input text, which can also reflect semantic and syntactic information about the given text. In [30], the graphical representations of prosody boundaries in Chinese text are passed to a graph encoder based on Graph Neural Networks (GNN) to generate prosodic information for the input text. The prosody boundaries of the Chinese language can be manually annotated or predicted using a pre-trained model. In contrast, [54] combines BERT-extracted features for input text with its graph dependency tree to produce word-level prosody representations. Specifically, the input text is passed through both BERT and a dependency parsing model to extract the dependency tree for word-level BERT embeddings. A Relational Gated Graph Network (RGGN) is used to convert this dependency tree into word-level semantic representations upon which the decoder of the TTS model is conditioned.

Different text-based features have been extracted from input text to obtain prosody (style) embeddings in [40]. The paper utilizes an emotion lexicon to extract word-level emotion features, including VAD (valence, arousal, dominance) and BE5 (joy, anger, sadness, fear, disgust). Additionally, the [CLS] embedding by BERT for each utterance is also extracted. The obtained features are then passed to a style encoder to produce a style embedding.

Other models under this category train a prosody encoder/predictor jointly with an autoregressive TTS model such as Tacotron 2, to encode some prosody related features utilizing text-based features. The trained encoder is then used at inference time to encode prosody-related features based on input text to the TTS model. The text-based input to these prosody encoders in most of the studies is the text's character/phoneme embeddings [20, 48, 71, 72, 103], while some studies use features extracted from the input text [64, 125]. For instance, [125] employs four ToBI (Tones and Break Indices) features as word-level prosody tags that are combined with the phoneme embedding as input to the TTS model. A ToBI predictor is jointly trained to predict four ToBI features based on grammatical and semantic information extracted from the input text using a self-supervised language representation model ELECTRA [164].

In addition to the previously mentioned features, several other prosodic features are also proposed as the output of the prosody predictors in other studies. For example, the prosody predictor in [103] predicts a set of utterance-wise acoustic features, including log-pitch, log-pitch range, log-phone duration, log-energy, and spectral tilt. In [48], the proposed pitch predictor outputs a continuous pitch representation, which is converted into discrete values using Vector Quantization (VQ) [149]. Furthermore, studies [20, 71] propose predicting the three prosody-related features, i.e., F0, energy, and duration, either by a single acoustic features predictor (AFP) [71] or via three separated predictors [20].

Another type of emotion embedding is sentiment feature embedding, which is utilized to produce expressive speech by extracting sentiment information from the input text. This is demonstrated in work [135], where the Stanford Sentiment Parser is used to generate vector embeddings or sentiment probabilities based on the tree structure of the sentence. To synthesize expressive speech, different combinations of probabilities and vector embeddings (for individual words or word-context) are added to the linguistic features as inputs to the TTS model.

5.3 Prosody controllability

Text-to-speech is a one-to-many mapping problem, i.e., for one piece of text there could be many valid prosody patterns because of speaker-specific variations. Accordingly, providing a kind of controllability over prosody-related features in synthesized speech is essential for generating expressive speech with different variations. However, it's not always easy to mark-up prosody or even to define boundaries between prosody events, i.e., duration boundaries can vary depending on segmentation, pitch contour prediction is error-prone, and prosody features may not always correlate well with what listeners perceive.

Several studies in literature have addressed the controllability issue in terms of selecting an emotion/style class or intensity level and adjusting prosody-related features at different speech levels. In this section, we discuss studies considering prosody controllability.

5.3.1 Modeling-specific prosody styles

This group of studies provides individual representations of expressive styles/emotions, enabling the control of prosody in synthesized speech by offering the ability to select from available representations or adjust their values. In some studies [55, 70, 116], style is modeled at a single speech/text level, while in other studies [68, 79, 133] a multi-level or hierarchical model of expressive styles is used to allow for a better capture of prosody variation in expressive speech.

In single-level prosody modeling approaches, [55] is one of the early studies that extends a baseline with fine-grained control over the speaking style/prosody of synthesized speech. The proposed modification involves adding an embedding network with temporal structure to either the speech-side or text-side of the TTS model. Accordingly, the resulting prosody embedding is of variable length, and it is used to condition input to either encoder or decoder based on the position of the embedding network. Speech-side prosody embedding provides adjustment of prosody at frame-level, while text-side prosody embedding enables phoneme-level prosody control.

Single-level prosody embeddings can be converted into discrete embeddings as in [70, 116]. Discrete prosody representations are easier to control and analyze and provide a better interpretation of prosodic styles. In [116], a word-level prosody embedding is proposed based on decision trees and a GMM. A word-level reference encoder is first used to obtain word-level prosody embedding from reference audio. A binary decision tree is employed to cluster embeddings with their identities based on their phonetic information. Prosody embeddings of words in each leaf node will differ only in their prosodies. Then prosody embeddings of each leaf can be clustered via a GMM model where clusters represent prosody tags. If the applied GMM consists of five components and a tree of ten leaf nodes, a set of 50 prosody tags is produced. At inference time, prosody tags can be selected manually or via a prosody predictor that is trained to select appropriate prosody tags based on input text.

In [70], an audiobook speech synthesis model is proposed. The model uses a character-acting-style extraction module based on ResCNN [165] to extract different character acting styles from the input speech. Discrete character-level styles are obtained via vector quantization (VQ) [149], which maps them to a codebook, limiting the number of styles. At inference, the discrete character-acting-styles are predicted via a style predictor. The character-level style predictor uses both character embeddings from Skip-Gram [166] and text-based features from RoBERTa [167] as input.

Regarding multi-level prosody modeling, some studies propose enhancing prosody control in the baseline models [74, 75, 77] by modifying their single-level prosody modeling to multiple levels. For instance, [133] proposes a hierarchical structure of [75] with multiple GST layers. Three GST layers are employed in the proposed model, each consisting of 10 tokens, which were found to yield better token interpretation. Tokens of the first and second layers were found to learn different speakers and styles, but these representations were not easily interpreted. Interestingly, the tokens in the third layer were able to generate higher quality samples with more distinct and interpretable styles. Specifically, third-layer styles exhibit clear differences in their features, including pitch, stress, speaking rate, start offset, rhythm, pause position, and duration.

Model in [77] is further extended in [68] with three VAEs to generate three different levels (utterance, phrase, and word) of latent variables with varying time resolutions. Acoustic features and linguistic features are passed as input to the three VAEs. Initially, a conditional prior (CP) is applied to learn a distribution for sampling utterance-level latent variables based on linguistic features from the input text. The generated latent variables are passed to other levels via auto-regressive (AR) latent converters that convert latent variables from coarser-level to finer-level with input text condition. In fact, the utterance-level latent variables can be used to control the generated speech styles, regardless of latent variables of other levels, as they are predicted based on the utterance-level latent variables.

The Controllable Expressive Speech Synthesis (ConEx) model in [79] proposes modeling prosody at two levels, utterance-level (global) and phone-level (local), using reference encoders [74]. However, the global prosody embedding is used to condition the local prosody embedding, resulting in an integrated prosody embedding. The local embeddings are 3D vectors that are converted into discrete local prosody embeddings (codes) via vector quantization (VQ) [149]. At inference time, the integrated prosody embedding is predicted by an auto-regressive (AR) prior model trained to predict categorical distributions for each of the discrete codes utilizing global prosody embedding and the phoneme embedding as inputs. While global prosody embedding can be obtained from training samples or from an audio reference, local prosody embeddings for a given global prosody embedding are achieved via the AR prior model. Fine-grained prosody control can be achieved by selecting a specific phoneme to start adjusting prosody from. The AR prior model will first generate the top *k* prosody options for this phoneme. Then, the local prosody sequence will be generated autoregressively for each of the first top *k* options by the AR prior model.

5.3.2 Modeling-specific prosody features

This group of studies provides individual representations of prosody-related features. Control over prosody of the synthesized speech is provided via selecting or adjusting a specific representation of a specific prosody-related feature. Some studies in this direction model prosody features at the global or utterance-level [97, 128], while other studies propose modeling at fine-grained levels [48, 63, 71, 122, 138], such as phoneme, syllable, or word-level.

The STYLER model [97], for example, employs multiple style encoders to factor speech style into several components, including duration, pitch, speaker, energy, and noise. This structure enables STYLER to generate controllable expressive speech by adjusting each of the individually modeled features. Furthermore, with the explicit noise encoding, other encoders can be constrained to exclude noise information as a style factor, and thus the model can generate clean speech even with noisy references. Adjusting the style factors, various styles of speech can be generated from STYLER.

Adjusting several features at fine-grained levels can be a difficult task. For example, FastSpeech2 [6] provides fine-grained control over pitch range, duration, energy, which are modeled at the phone-level (phone-wise), and it is not easy to adjust these features to achieve a specific prosodic output. Raitio and Seshadri [128] improves FastSpeech2 with an utterance-wise (coarse-grained) prosody model using an additional variance adaptor. That second variance adaptor is the same as the original one, but it models five features at the utterance-level: pitch, pitch range, duration, energy, and spectral tilt. These features are then concatenated with the corresponding output of the first variance adaptor. Such utterance-wise prosody model enables easier control of prosody while still allowing modification at the phone-level. To control high-level prosody, a bias is added to the corresponding utterance-wise prosody predictions. A phone-level prosody control is achieved by directly modifying the phone-wise features.

Fine-grained control over a specific prosody-feature can also be required specially for strong speaking styles. To that end, in [71], a predictor is proposed to predict F0, energy, and duration features at the phoneme-level. During inference, the predicted features are generated based on the input text alone; however, they can also be provided externally and modified as desired.

Furthermore, two prosody modeling levels are proposed in [63]: the local level (word-level) and global level (utterance-wise). The global prosody embedding is the emotion embedding obtained by a reference-based encoder. The local prosody embedding is obtained from a predictor of the F0 features at the word-level with global prosody embedding and the phoneme embedding as inputs. Both embeddings are then passed to a multi-style encoder to form the final multi-style prosody embedding. Therefore, modifying the predicted F0 values can provide control of prosody at the utterance, word, and phoneme levels.

More flexibility in controlling the F0 feature is provided in the controllable deep auto-regressive model (C-DAR) model [138] which allows for F0 contour adjustment by the user. To achieve this goal, three strategies are used: 1) context awareness by conditioning the model on the preceding and following speech during training, 2) conditioning the model on some random segments of ground truth F0, and 3) predicting F0 values in reverse order. Additionally, several text-based features are used as input to the model, including word embeddings derived from BERT, V/UV label, one-hot vector for the nearby punctuation, and phoneme encodings. At inference, F0 values specified by the user are used as alternatives for the ground truth F0 segments, and the model predicts the rest of the utterance's F0 contour through context awareness.

Discrete fine-grained representations for prosody features as in [48, 122] are also useful to limit the number of the obtained representations. Both studies [48, 122] utilize VQ [149] to map each prosody embedding to the closest discrete representation from a predefined codebook. In [48], a pitch predictor is used to predict character-level continuous pitch representation using character embeddings from the text encoder as input. Zhang et al. [122], however, produces syllable-level prosody embeddings from a reference encoder that takes F0, intensity, and duration features from reference audio as input. The resulting prosody embeddings are then mapped to a predefined codebook to extractb discrete prosody codes. Resulting prosody codes in [48] represent the pitch and other suprasegmental information that can be adjusted via a specific bias value to generate speech with different pitch accents. The codes in [122], can be interpreted as representing some prosody features such as pitch and duration. The prosody variation at the syllable-level can be manually controlled by assigning each syllable the desired prosody code from the codebook.

In [125], ToBI features, which involve a set of conventions used for transcribing and annotating speech prosody, are used. The applied ToBI features are four word-level tags: pitch accents, boundary tones, phrase accents, and break indices. The extracted ToBI tags are used as input to TTS model. Simultaneously, a ToBI predictor is trained to predict these prosody tags based on grammatical and semantic information extracted from the input text using a self-supervised language model. The resulting model had the ability to control the stress, intonation, and pause of the generated speech to sound natural, utilizing only ToBI tags from the text-based predictor.

5.3.3 Modeling prosody strength

This group of studies focus on regulating the strength of emotion or prosody. For instance, [61] utilizes the distance between emotion embeddings and the neutral emotion embeddings to identify scalar values for emotion intensity. It proposes a phoneme-level emotion embedding and a fine-grained emotion intensity. The emotion embedding is first obtained via a reference encoder. The emotion intensity is then generated by an intensity extractor that takes the emotion embedding as input. The intensity extractor produces intensity as a scalar value based on the distance between the emotion embedding and the centroid of a pre-defined cluster for neutral emotion embeddings. The resulting emotion intensity values are quantized into pseudo-labels that serve as the index for an intensity embedding table.

Another method for learning emotion strength values in an unsupervised manner is by using ranking functions. Studies [27, 31, 33, 64] utilize a ranking function-based method named relative attributes [89] for this purpose. In [33], prosody is modeled at three levels: global-level representation by emotion embedding, utterance-level represented by prosody embedding from a reference-based encoder, and the local-level represented by emotion strength. The study trains an emotion strength extractor at the syllable-level based on input speech utilizing the ranking function. Simultaneously, a predictor of emotion strength is trained based on features extracted from input text via BERT model. Besides changing emotion label and emotion reference audio, the model provides manual control of the emotion strength values in the synthesized speech.

Alternatively, the reference encoder in [31] functions as a ranking function to learn a phoneme-level emotion strength (descriptor) sequence. The proposed ranking function [89] receives its input from fragments of target reference audio obtained via a forced alignment model to phoneme boundaries. The OpenSMILE [139] tool is then used to extract 384-dimensional emotion-related features from these reference speech fragments as input to the ranking function. Similarly, the proposed ranking function in [27] takes a set of acoustic features extracted from the input speech via OpenSMILE tool but at the utterance-level as input. The ranking function leverages the difference between neutral samples and samples associated with each emotion class in the dataset. The training process is formulated as solving a max-margin optimization problem. The resulting emotion strength scalars can be manually adjusted or predicted based on text or reference speech.

In [64], both emotion class and emotion strength value are obtained via a joint emotion predictor based only on the input text. The input to the predictor is features extracted from input text via the Generative Pre-trained Transformer (GPT)-3 [88]. Emotion class and emotion strength are the two outputs of the predictor where the former is represented as a one-hot encoded vector and the latter is presented as a scalar value. Emotion labels and emotion strength values which are also obtained via [89], are used as ground truth for predictor training.

Another ranking method is proposed in [19] using the ranking support vector machine. The model generates style embedding and speaker embedding via two separate encoders. Both style and speaker embeddings at inference time are represented by centroids of each single speaker and style embeddings. However, a linear SVM is trained with the model to provide the ability for style embedding adjustment. The proposed SVM model is trained to classify between neutral emotion and a specific emotion embedding, where the learned hyperplane is utilized to move(scale) the style vectors in a direction towards/opposite to the hyperplane.

Another type of control that contributes to generating speech with a better representation of local prosodic variation is introduced in [124]. The proposed model suggests an unsupervised approach to obtain word-level prominence and phrasal boundary strength features. For this purpose, continuous wavelet transform (CWT) [168] is utilized to extract continuous estimates of word prominence and boundary information from the audio signal. First, the three prosodic signals f0, energy, and duration are extracted and combined as input to the CWT. Then, the combined signal is decomposed via CWT into scales that represent prosodic hierarchy. Word and phrase-level prosody are then obtained by following ridges or valleys across certain scales. The continuous word prominence and boundary estimates are achieved via the integration of the resulting lines aligned with the textual information. With manually identified intervals, the continuous values of prominence and boundary strength are then discretized.

5.3.4 Prosody clustering

In this section, methods for selecting the appropriate prosody embedding for the referenced-based ETTS models are described. To begin with, clustering methods are utilized in [57, 58] to generate representative prosody embeddings for each emotion class when the GST-TTS model is trained with a labeled dataset. Initially, the resulting emotion embeddings are clustered in a 2d space. In [57], the centroid of each cluster is used as the weights of the GSTs to generate emotion embedding for each emotion class. In [58], the weight vector that represents each emotion cluster is obtained by considering the inter and intra distances between emotion embedding clusters. Specifically, an algorithm is used for minimizing each embedding distance to the target emotion cluster and maximizing its distance to other emotion clusters.

Similarly, clustering algorithms are applied in [112, 113] to achieve discrete prosody embeddings but for two specific prosody-related features. The two studies employ K-means algorithm to cluster F0 and duration features extracted for each phoneme. The centroids of the clusters are then used as discrete F0 and duration values/tokens for each phoneme. work [112] applies a balanced clustering method with duration features to overcome degradation in voice quality that appeared in [113] during duration control. Moreover, to keep phonetic and prosodic information separate during training, an attention unit is introduced to map prosody tokens to decoder hidden states and generate prosody context vectors. The resulting discrete tokens for F0 and duration features provide a fine-grained level of control over prosody by changing the corresponding prosodic tokens for each phoneme.

In [105], a cross-domain SER model with the GST-TTS model is proposed to obtain emotion embeddings for an unlabeled dataset. The cross-domain SER model is trained using two datasets including: 1) an SER dataset (source) labeled with emotions, and 2) a TTS dataset (target) that is not labeled. Simultaneously, the SER model trains an emotion classifier that generates soft labels for the unlabeled TTS dataset. These soft labels are then used to train an extended version of the baseline in [74] with an emotion predictor. In the training process, the weights of the style tokens layer are passed as input to the predictor, which employs the learned soft labels as ground truth values. At inference time, weights vectors for each emotion class are averaged to obtain the emotion class embedding. However, since the predicted labels for the TTS dataset are soft labels, and thus not entirely reliable, only the top K samples with the highest posterior probabilities are selected.

5.4 Speech synthesis for unseen speakers and unseen styles

Building a speech synthesis model that supports multiple speakers or styles can be achieved by training TTS model with a multi-speaker multi-style dataset. However, generating speech for an unseen speaker or style is a challenging task for which several solutions have been proposed in the literature. A popular approach is to fine-tune the averaged TTS model with some samples from the unseen target speaker or style. The fine-tuning process may require a single sample from the unseen speaker or style (referred to as one-shot models) or a few samples (referred to as few-shot models). There are also models that do not require any fine-tuning steps, and these are known as zero-shot TTS models.

For instance, the fine-tuning process proposed in [112] focused on sentences used in the process to ensure phonetic coverage, meaning that each phoneme should appear at least once in these sentences. The proposed model requires about 5 minutes of recordings from the unseen target speaker to clone the voice and allow for manipulation of some voice features (such as F0 and duration) by the model at the phoneme-level.

Another approach to address the problem of unseen data is to employ specific structures in the TTS model, as proposed in [52, 96, 97, 107]. As an example, in [107], a cycle consistency network is proposed with two Variational Autoencoders (VAEs). The model incorporates two training paths: a paired path and an unpaired path. The unpaired path refers to training scenarios where the reference audio differs from the output (target) speech in terms of text, style, or speaker. Two separate style encoders are utilized in the model, with one dedicated to each path. This structure facilitates style transfer among intra-speaker, inter-speaker, and unseen speaker scenarios.

In [52], the U-net structure proposed for the TTS model supports one-shot speech synthesis for unseen styles and speakers. The U-net structure is used between the style encoder and the mel decoder of the TTS model, with an opposite flow between them. Both the style encoder and decoder consist of multiple modules with the main building unit as ResCnn1D and instance normalization (IN) layers. The decoder receives phoneme embedding and produces the Mel-spectrogram as output. In parallel, the style encoder receives the reference audio and produces its linguistic content with guidance from the content (text) encoder. The style encoder modules produce latent variables, i.e., mean, and standard deviation, for the hidden inputs in the IN layers. These latent variables are used to bias and scale the normalized hiddens of the corresponding module layers in the decoder.

A separate encoder (reference encoder) has been used in [96] to extract speaker-related information besides the prosody encoder (extractor) that encodes prosody features into the prosody embedding. A prosody predictor is also trained to predict the prosody embedding based on the phoneme-embedding. While the instance normalization (IN) layer is utilized by the prosody extractor to remove global (speaker) information and to keep prosody-related information, the speaker encoder is designed with a special structure (Conv2D layers, residual blocks (GLU with fully connected layers), and a multi-head self-attention unit) for better extraction of speaker information. Moreover, instead of concatenation or summation with the decoder input, the speaker embedding is adaptively affine transformed to the different FFT blocks of the decoder through a Speaker-Adaptive Linear Modulation (SALM) network that is inspired by Feature-wise Linear Modulation (FiLM) [141]. The speaker encoder and conditioning of decoder blocks with speaker embedding allow the model to generate natural speech for unseen speakers with only a single reference sample (zero-shot).

The attention unit used in seq2seq TTS models aims at mapping the different length between text and audio pairs. However, it can get unstable when the input is not seen during training [97]. The STYLER model has addressed this issue by using a linear compression or expansion of the audio to match the text's length via a method named Mel Calibrator. With this simplification of the alignment process as a scaling method, the unseen data robustness issue is alleviated and all audio-related style factors become dependent only on the audio.

Similarly, in [119], the Householder Normalizing Flow [169] is incorporated into the VAE-based baseline model [77]. The Householder normalizing flow applies a series of easily invertible affine transformations to align the VAE's latent vectors (style embeddings) with a full covariance Gaussian distribution. As a result, the correlation among the latent vectors is improved. Generally, this architecture enhances the disentanglement capability of the baseline model and enables it to generate embedding for unseen style with just a single (one-shot) utterance of around one second length.

The Multi-SpectroGAN TTS model proposed in [98] is a multi-speaker model trained based on adversarial feedback. The model supports the generation of speech for unseen styles/speakers by introducing adversarial style combination (ASC) during the training process. Style combinations result from mixing/interpolating style embeddings from different source speakers. The model is then trained with adversarial feedback using mixed-style mel-spectrograms. Two mixing methods are employed: binary selection or manifold mix-up via linear combination. This training strategy enables the model to generate more natural speech for unseen speakers.

Lastly, recent TTS models based on in-context learning [18, 22, 25] all share the capability to perform zero-shot speech synthesis, as explained in Section 4.4. In fact, the in-context training strategy underlies the ability of these models to synthesize speech given only a style prompt with the input text. Specifically, the synthesis process treats the provided prompt/reference as part of the desired output speech. Therefore, the model's goal is to predict the rest of this speech in the same style as the given part (prompt) and with the input text. In Table 5 we list papers addressing each challenge.

6 Datasets and open source codes

Deep learning models, including TTS models, rely heavily on the availability of data in terms of size and diversity. Furthermore, the quality of synthesized speech by TTS models is closely tied to the quality and size of the data used for model training. ETTS models face even greater challenges in this regard, as they require data to be not only high-quality and clean but also to accurately represent the numerous available speaking styles and emotions.

A main limitation in the domain of expressive speech synthesis is the inadequate availability of expressive speech datasets. Although there are several emotional and expressive speech datasets publicly available, they still fall short in terms of size, accuracy, and diversity required to train effective ETTS models. As a result, current ETTS models still suffer from performance degradation and poor generalization. In [170], which introduces

Table 5 List of papers addressing main expressive speech synthesis challenges. "IL" stands for information leakage, "LR" is a shortcut for inference that lack reference audio, "PC" stands for prosody controllability and "US" stands for unseen style/speaker

References	Challenges addressed			
	IL	LR	PC	US
[62]	✓	✓	✓	✓
[59, 96, 126]	✓	✓		✓
[97, 112, 120]	✓		✓	✓
[18, 22, 25, 52, 93, 98, 107, 119]	✓			✓
[23, 54, 101, 102, 117, 130, 137]	✓			
[33, 63, 68, 70, 116]		✓	✓	
[21, 47, 111]	✓	✓		
[19]	✓		✓	
[31, 48, 49, 53, 55, 57, 58, 61, 71, 72, 79, 99, 105, 113, 122–125, 128, 133, 138]			✓	
[17, 35, 37, 44, 46, 50, 73, 91, 94, 100, 129, 131]		✓		

a recent multi-emotion and multi-speaker dataset, it provides a concise summary of the majority of the available emotional speech datasets. Additionally, there are open expressive datasets, such as the Blizzard datasets [171], which are larger in size but lack any labels. Furthermore, widely used TTS datasets, such as LJSpeech [172] with a single speaker, and VCTK [173], and LibriTTS [174] with multiple speakers, are also employed in expressive TTS models to address prosodic features generally, control issues, or learning different speakers' styles. In Table 6, we list the main open-source databases used in the papers covered by this review.

Numerous internal expressive speech datasets are utilized in many studies in literature. Some of these datasets are of large size and exhibit good quality, with high diversity, including multiple speakers, styles, and emotions. However, they are not open to the research community. Additionally, replicating the work presented in these studies or making further improvements is challenging. Constructing an expressive speech dataset is, in fact, a demanding endeavor compared to collecting neutral speech datasets, due to several factors.

First of all, differences among speakers in portraying different speech styles or emotions pose the first challenge. Some speakers may overact, while others may misinterpret or blend acting styles or emotions. Secondly, there are variations in emotional interpretation among different listeners who annotate the same expressive speech, which can impact the accuracy and consistency of these datasets. Notably, [66] highlighted the differences in emotional reception among listeners for the same utterance, as explained in Section 3.1.

Moreover, the wide range of human emotions and speaking styles introduces further complexities. In the literature, emotions are defined and classified based on various criteria [175]. One common classification approach distinguishes between discrete emotions, which are considered basic emotions recognizable through facial expressions and biological processes, and dimensional emotions, which are identified based on dimensions such as valence and arousal [176, 177]. A well-known study conducted by Paul Ekman and Carroll Izard [178] involved cross-cultural studies and identified six main basic emotions, including anger,

Table 6 List of main open source expressive TTS databases according to publications reviewed in this work

Database	Language	Multi Emotion	Multi Speaker
Blizzard Challenge 2012, 2013, 2016, 2019	English		
VCTK	English		✓
LibriSpeech	English		✓
IEMOCAP	English	✓	✓
CMU ARCTIC	English		✓
LJSpeech	English		
LibriTTS	English		✓
Chinese Standard Mandarin Speech Copus(CSMSC)	Mandarin Chinese		
Aishell-3	Mandarin Chinese		
Korean Emotional Speech (KES) dataset	Korean	✓	
English conversation corpus (ECC)	English		✓
IndicTTS database	Indian		
Emotional Speech Dataset (ESD)	English/Mandarin Chinese	✓	✓
Japanese Kamishibai and audiobook corpus (J-KAC)	Japanese		
Multilingual LibriSpeech	Multilingual		✓
Korean Single Speaker (KSS)	Korean		✓

disgust, fear, happiness, sadness, and surprise. In fact, although the available emotional datasets diverge in the set of emotions they cover, as shown in [170], most of the emotions considered in these datasets belong to the six basic emotion classes identified by [178].

Additionally, when considering different languages and multiple speakers, the challenge becomes more intricate. However, with the new trend that introduces the language modeling approach to the field of speech synthesis, it becomes possible to train TTS models on a large amount of data using an in-context learning strategy. This vast amount of data provides diversity in speakers, speaking styles, and prosodies, and it can be used for training despite noisy speech and inaccurate transcriptions. In fact, recent TTS models based on language modeling, such as VALL-E [22], NaturalSpeech 2 [18], and Voicebox [25], have been successful in various speech-related tasks, especially zero-shot speech synthesis. Besides, they have shown promising results in expressive speech synthesis, as they are able to replicate the speech style and emotion provided in a single input acoustic prompt to the synthesized speech.

Table 7 List of main open source implementations for TTS models with related links

Source name	Link
Espnet	github.com/espnet/espnet
coqui	github.com/coqui-ai
Mozilla	github.com/mozilla
NeMo (NVidia)	github.com/NVIDIA/NeMo
espeak-ng	https://github.com/espeak-ng/espeak-ng
marytts	github.com/marytts
CSTR-Edinburgh	github.com/CSTR-Edinburgh
Hugging Face	huggingface.co/docs/transformers/tasks/text-to-speech

As for open-source codes, several implementations and repositories are publicly available. Table 7 list some main open-source implementations for expressive speech synthesis models.

7 Evaluation metrics

An essential step in building any generative model is to evaluate its performance and compare it to the previous state-of-the-art models. In addition to using the same datasets, standard evaluation metrics are also needed to compare different approaches with each other. While evaluation metrics applied for general TTS models focus on speech quality in terms of intelligibility and naturalness, the assessment of ETTS models' performance goes beyond that, focusing on other aspects. Evaluation metrics of ETTS models measure more aspects like emotion or style expressiveness, prosodic features, and controllability over all these aspects. Tables 8 and 9 list the common objective and subjective metrics applied for evaluating TTS models' performance, respectively.

In fact, all the mentioned objective and subjective evaluation metrics have been applied by the studies covered in this review. However, in many studies, these metrics have been applied differently to assess aspects related to expressivity. In other words, these metrics have been applied to samples representing different emotions, speaking styles, and their varying levels of strength or intensity. Furthermore, samples can represent various speech synthesis scenarios, such as parallel/non-parallel style transfer and seen/unseen styles or speakers.

On the other hand, various additional methods have been proposed in the papers to evaluate either the effectiveness of the proposed models or the expressiveness of the synthesized speech. For instance, emotion and style classifiers as in [57, 64] and speech emotion recognition (SER) models as in [19, 58] which are used to measure classification accuracy, reflecting the efficiency of the

Table 8 Objective evaluation metrics for expressive speech synthesis models

Metric	Description
Mel-Cepstral Distortion (MCD)	Sums the squared differences between the Mel-Frequency Cepstrum Coefficients (MFCC) from the ground truth and synthesized sample.
Gross Pith Error (GPE)	Calculates percentage of voiced frames that deviate in pitch by more than 20% compared to the ground truth samples.
Voice Decision Error (VDE)	Measures the difference of voiced/unvoiced decision between the ground truth and the synthesized sample.
F0 Frame Error (FFE)	Combines GPE and VDE by measuring the percentage of frames that either contain a 20% pitch error (GPE) or a voicing decision error (VDE) in ground truth and synthesized samples.
Word Error Rate (WER)	Measures word error rate of the synthesized speech's transcription with respect to the input text. Public automatic speech recognition (ASR) models are used for transcribing synthesized speech.
Band APeriodicity Distortion (BAPD)	Measures over linearly spaced band aperiodicity coefficients between the ground truth and the synthesized samples.
Root Mean Square Error (RMSE)	Measure the root mean square error of F0 or energy of the synthesized samples compared to their ground truth.

Table 9 Subjective evaluation metrics for expressive speech synthesis models

Metric	Description
Mean Opinion Score (MOS)	Listeners to scores quality (naturalness and intelligibility) of synthesized speech with a five-point scoring system.
Comparison Mean Opinion score (CMOS)	Compares MOS values between models under test and the baseline via comparing ground truth and synthetic samples from each model.
Differential mean opinion score (DMOS)	Listeners score samples from one to five based on its similarity to a specific emotion or style.
AB preference test	Listeners score same sentence synthesized by the two models and select the one that fulfills the given condition more than the other.
ABX preference test	Listeners hear three samples A, B and X ,where X represents the target speech, and they should score the one that is more close to target speech.
MUltiple Stimuli Hidden Reference and Anchor (MUSHRA)	Listeners are presented with mixed samples including synthesized sample, natural speech samples (named proper reference) and total loss sample (named anchor). Listeners score each sample from 0 to 100 through a double-blind listening test.

proposed model in generating emotional speech. Furthermore, visualization and plotting of different prosodic features, variables, or embeddings are also employed in several studies [27, 63, 71, 105, 112, 122] to evaluate the expressivity of generated samples and compare different approaches or synthesizing scenarios. Additionally, ablation studies as in [19, 97] have also been conducted to measure the effectiveness of each component in the proposed model and how it affects the expressivity of generated speech.

8 Discussion

This systematic review of ETTS models within the last 5 years has shown a wide variety of methods and techniques that have been applied in this field, particularly DL-based approaches. However, current ETTS models are still far away from achieving their goal of generating human-like speech in terms of expressiveness, variability, and control flexibility. The main contribution of this review to the literature is to provide a full picture of the efforts that have been conducted in this field for newcomers or beginner researchers, helping them to identify their roadmap within this area.

On the other hand, we hope that the provided information and summaries in this review including methods taxonomy, modeling challenges, datasets and evaluation metrics can be of good support and guidance for researchers in this area to compare and identify state-of-the-art models on one side, and to spot gaps yet to be filled on the other side. Our focus in this review was to identify the main methods and structures applied in literature for ETTS besides challenges and problems that they encounter. Nevertheless, papers covered here can be further investigated to analyze models' performance and compare their results utilizing the same datasets and evaluation metrics.

Additionally, based on our investigation in this review, we would like in this discussion to highlight some research gaps within this research area that need to be considered in future work.

- *Terminology Identification*: During the course of this work, we observed a lack of clear definitions for main terminologies used in this research area, such as "expressive," "emotion," "prosody," and "style". Early studies, as in [78, 80, 82], often used the terms "emotion" and "style" interchangeably, encompassing data with different emotions (happy, sad, etc.) or a blend of emotion and style (e.g., happy call center, sad technical support). Furthermore, the term "expressive speech" is used in a general sense to describe speech that is natural-sounding and resembles human speech overall, as in studies [35, 40, 54, 76, 94, 97, 114, 119, 134, 135]. However, it is also utilized in other studies to describe speech with different labels for emotions [26, 28, 49, 64, 91], styles (newscaster, talk-show, call-center, storytelling.) [45, 106, 115, 118], or combinations of emotions and styles [19, 59, 68, 78, 79, 84]. On the other hand, a single style itself can encompass speech featuring multiple emotions and variable prosody attributes, as exemplified by the Blizzard2013 dataset [171], which includes data in a storytelling style. Many studies [50, 74, 93, 94, 121, 122] employ the Blizzard2013 dataset to train their TTS models and generate expressive speech. The resulting speech from these models in this case exhibits varying prosody and conveys different emotions. In general, the existing literature lacks a distinct differentiation among these terminologies and their associated variations, which can lead to confusion among researchers and complicated comparisons between models. Therefore, it is highly recommended to conduct further investigation to establish clear and comprehensive definitions for these terms. Specifically, each term needs to be accurately delineated, specifying its respective types, attributes, and speech features.

- *Controllable Expressive Speech*: Providing control over the expressiveness of synthesized speech can be considered an advanced step in this area of research. As we have discussed in Section 5.3 several recent studies have addressed different aspects to provide more control over expressivity in synthesized speech. The aspects covered in these studies include selection and adjusting different prosody-related features at coarse levels (utterance, paragraph, sentence, etc.) as well as fine-grained levels (word, syllables, phonemes, etc.). However, the proposed controlling techniques with their achieved results are still considered small steps in this important research area of expressive speech synthesis, and more efforts are needed and expected in this direction in the near future. In fact, bridging this research gap is a crucial step towards the goal of speech synthesis research to produce human-like speech.
- *Evaluation metrics*: Despite the existence of several metrics applied in the literature to evaluate the performance of ETTS models, no general and standard metrics have been identified to facilitate the comparison process among different approaches. Furthermore, since the evaluation of expressive speech is more sophisticated and challenging, there is still a high demand for more accurate metrics capable of capturing various aspects of expressiveness in speech. Additionally, with the increased attention on building controllable ETTS models, the need arises for efficient evaluation metrics for controllability related aspects.
- *Datasets*: As discussed in Section 6, availability of inclusive, high quality and large size expressive dataset is crucial for achieving efficient ETTS models. However, building a comprehensive emotional speech dataset that encompasses a wide range of emotions, styles, speakers, and languages with high quality remains a formidable objective in the expressive speech synthesis field. The challenges extend beyond the issues mentioned in Section 6 and encompass aspects such as time and cost. Language modeling-based approaches could be the future of the field, overcoming these challenges, but they are still in the early stages, and further research in this direction is necessary.

9 Conclusion

This paper presents the findings of our systematic literature review on expressive speech synthesis over the past 5 years. The main contribution of this article is the development of a comprehensive taxonomy for DL-based approaches published in this field during that specific time frame. The approaches are classified into three primary categories based on the learning method, followed by models within each category. Further subcategories are identified at the lower levels of the taxonomy, considering the methods and structures applied to achieve expressiveness in synthesized speech. In addition to the ETTS approaches taxonomy, we provide descriptions of the main challenges in the ETTS field and proposed solutions from the literature. Furthermore, we support the reader with brief summaries of ETTS datasets, performance evaluation metrics, and some open-source implementations. The significance of our work lies in its potential to serve as an extensive overview of the research conducted in this area from different aspects, benefiting both experienced researchers and newcomers in this active research domain.

Some main directions for future work in this area involve collection of large expressive datasets in different languages, going from acted expressive style to realistic style. Further evaluation metrics are still needed in this area for assessing models' performance such as evaluation of prosody controllability. Efficient metrics are also required for monitoring performance and guiding loss evaluation during the training process. These need to be lightweight and fast in order not to slow down training but still reliable. Another suggestion for future investigations is to take cultural differences in perception of expressions into account for multi-language, multi-speaker expressive TTS applications. Moreover, as speech is just one modality for expressions, multi-modal approaches that combine facial expressions, eye movements, body movements, gestures, non-verbal clues, etc., will be required to reach human-level expressiveness. Training several modalities together could be beneficial as the model can transfer useful information from one modality to another in a self-supervised fashion.

Abbreviations

TTS	Text to speech
ETTS	Expressive text to speech
HMM	Hidden Markov model
SPSS	Statistical parametric speech synthesis
GMM	Gaussian mmixture models
DL	Deep learning
VAE	Variational auto-encoder
CNN	Convolutional neural network
DNN	Deep neural network
FFT	Feed-forward transformer
LSTM	Long short-term memory networks

GRL	Gradient reversal layer
VQ	Vector quantization
AR	Auto-regressive
BERT	Bidirectional encoder representations from transformers
RoBERTa	Robustly optimized BERT pre-training approach
MFCC	Mel-frequency ceptral coefficients
NLTK	Natural language toolkit
TF-IDF	Term frequency-inverse document frequency
MSE	Mean square error
ASR	Automatic speech recognition
SVM	Support vector machine
GRU	Gated recurrent unit
SER	Speech emotion recognition
GNN	Graph neural network
GPT	Generative pre-trained transformers
NLP	Natural language processing
DDGAN	Denoising diffusion GANs

Acknowledgements
Not applicable.

Authors' contributions
HB: main writing and editing. OY: review. CD: review and supervision. All authors read and approved the final manuscript.

Funding
Not applicable.

Availability of data and materials
Not applicable.

Competing interests

Competing interests
The authors declare that they have no competing interests.

Received: 9 June 2023 Accepted: 22 January 2024
Published online: 12 February 2024

References

1. Wikipedia. Speech Synthesis - Wikiversity — en.wikiversity.org. https://en.wikiversity.org/wiki/Speech_Synthesis. Accessed 09 Jun 2023
2. H. Ze, A. Senior, M. Schuster, in *2013 ieee international conference on acoustics, speech and signal processing*. Statistical parametric speech synthesis using deep neural networks (IEEE, 2013), pp. 7962–7966. https://doi.org/10.1109/icassp.2013.6639215
3. Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R.J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, et al., in *Proc. Interspeech 2017*. Tacotron: Towards end-to-end speech synthesis (2017), pp. 4006–4010. https://doi.org/10.21437/Interspeech.2017-1452
4. J. Shen, R. Pang, R.J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan, et al., in *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions (IEEE, 2018), pp. 4779–4783. https://doi.org/10.1109/icassp.2018.8461368
5. Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, T.Y. Liu, Fastspeech: Fast, robust and controllable text to speech. Adv. Neural Inf. Process. Syst. **32**, 3171–3180 (2019)
6. Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, T.Y. Liu, Fastspeech 2: Fast and high-quality end-to-end text to speech. (2020). arXiv preprint arXiv:2006.04558
7. Y. Kumar, A. Koul, C. Singh, A deep learning approaches in text-to-speech system: a systematic review and recent research perspective. Multimed. Tools Appl. **82**(10), 15171–15197 (2023)
8. F. Khanam, F.A. Munmun, N.A. Ritu, A.K. Saha, M. Firoz, Text to speech synthesis: A systematic review, deep learning based architecture and future research direction. J. Adv. Inform. Technol. **13**(5), 398–412 (2022)
9. Z. Mu, X. Yang, Y. Dong, Review of end-to-end speech synthesis technology based on deep learning. (2021). https://doi.org/10.48550/arXiv.2104.09995
10. Y. Ning, S. He, Z. Wu, C. Xing, L.J. Zhang, A review of deep learning based speech synthesis. Appl. Sci. **9**(19), 4050 (2019)
11. Z.H. Ling, S.Y. Kang, H. Zen, A. Senior, M. Schuster, X.J. Qian, H.M. Meng, L. Deng, Deep learning for acoustic modeling in parametric speech generation: A systematic review of existing techniques and future trends. IEEE Signal Process. Mag. **32**(3), 35–52 (2015)
12. O. Nazir, A. Malik, in *2021 2nd International Conference on Secure Cyber Computing and Communications (ICSCCC)*. Deep learning end to end speech synthesis: a review (IEEE, 2021), pp. 66–71. https://doi.org/10.1109/icsccc51823.2021.9478125
13. X. Tan, T. Qin, F. Soong, T.Y. Liu. A survey on neural speech synthesis (2021). arXiv preprint arXiv:2106.15561
14. N. Kaur, P. Singh, Conventional and contemporary approaches used in text to speech synthesis: A review. Artif. Intell. Rev. **2022**, 1–44 (2022)
15. A. Triantafyllopoulos, B.W. Schuller, G. İymen, M. Sezgin, X. He, Z. Yang, P. Tzirakis, S. Liu, S. Mertes, E. André et al., An overview of affective speech synthesis and conversion in the deep learning era. Proc. IEEE (2023), vol. 111, no. 10, pp. 1355–1381
16. Scopus. Scopus — scopus.com. https://www.scopus.com/. Accessed 7 Jan 2023
17. S. Lei, Y. Zhou, L. Chen, Z. Wu, S. Kang, H. Meng, in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Context-aware coherent speaking style prediction with hierarchical transformers for audiobook speech synthesis (IEEE, 2023), pp. 1–5. https://doi.org/10.1109/icassp49357.2023.10095866
18. K. Shen, Z. Ju, X. Tan, Y. Liu, Y. Leng, L. He, T. Qin, S. Zhao, J. Bian, Naturalspeech 2: Latent diffusion models are natural and zero-shot speech and singing synthesizers. (2023). arXiv preprint arXiv:2304.09116
19. S. Jo, Y. Lee, Y. Shin, Y. Hwang, T. Kim, in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Cross-speaker emotion transfer by manipulating speech style latents (IEEE, 2023), pp. 1–5. https://doi.org/10.1109/icassp49357.2023.10095619
20. T.H. Teh, V. Hu, D.S.R. Mohan, Z. Hodari, C.G. Wallis, T.G. Ibarrondo, A. Torresquintero, J. Leoni, M. Gales, S. King, in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Ensemble prosody prediction for expressive speech synthesis (IEEE, 2023), pp. 1–5. https://doi.org/10.1109/icassp49357.2023.10096962
21. D. Yang, S. Liu, R. Huang, G. Lei, C. Weng, H. Meng, D. Yu, Instructtts: Modelling expressive tts in discrete latent space with natural language style prompt. (2023). arXiv preprint arXiv:2301.13662
22. C. Wang, S. Chen, Y. Wu, Z. Zhang, L. Zhou, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li, et al., Neural codec language models are zero-shot text to speech synthesizers. (2023). arXiv preprint arXiv:2301.02111
23. W. Zhao, Z. Yang, An emotion speech synthesis method based on vits. Appl. Sci. **13**(4), 2225 (2023)
24. H.S. Oh, S.H. Lee, S.W. Lee, Diffprosody: Diffusion-based latent prosody generation for expressive speech synthesis with prosody conditional adversarial training. (2023). arXiv preprint arXiv:2307.16549
25. 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. (2023). arXiv preprint arXiv:2306.15687
26. P. Wu, Z. Ling, L. Liu, Y. Jiang, H. Wu, L. Dai, in *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. End-to-end emotional speech synthesis using style tokens and semi-supervised training (IEEE, 2019), pp. 623–627. https://doi.org/10.1109/apsipaasc47483.2019.9023186
27. X. Zhu, S. Yang, G. Yang, L. Xie, in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. Controlling emotion strength with relative attribute for end-to-end speech synthesis (IEEE, 2019), pp. 192–199. https://doi.org/10.1109/asru46091.2019.9003829
28. X. Zhu, L. Xue, Building a controllable expressive speech synthesis system with multiple emotion strengths. Cogn. Syst. Res. **59**, 151–159 (2020)

29. G. Xu, W. Song, Z. Zhang, C. Zhang, X. He, B. Zhou, in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Improving prosody modelling with cross-utterance bert embeddings for end-to-end speech synthesis (IEEE, 2021), pp. 6079–6083. https://doi.org/10.1109/icassp39728.2021.9414102
30. A. Sun, J. Wang, N. Cheng, H. Peng, Z. Zeng, L. Kong, J. Xiao, in *2021 IEEE Spoken Language Technology Workshop (SLT)*. Graphpb: Graphical representations of prosody boundary in speech synthesis (IEEE, 2021), pp. 438–445. https://doi.org/10.1109/slt48900.2021.9383530
31. Y. Lei, S. Yang, L. Xie, in *2021 IEEE Spoken Language Technology Workshop (SLT)*. Fine-grained emotion strength transfer, control and prediction for emotional speech synthesis (IEEE, 2021), pp. 423–430. https://doi.org/10.1109/slt48900.2021.9383524
32. T. Li, S. Yang, L. Xue, L. Xie, in *2021 12th International Symposium on Chinese Spoken Language Processing (ISCSLP)*. Controllable emotion transfer for end-to-end speech synthesis (IEEE, 2021), pp. 1–5. https://doi.org/10.1109/iscslp49672.2021.9362069
33. Y. Lei, S. Yang, X. Wang, L. Xie, Msemotts: Multi-scale emotion transfer, prediction, and control for emotional speech synthesis. IEEE/ACM Trans. Audio Speech Lang. Process. **30**, 853–864 (2022)
34. T. Li, X. Wang, Q. Xie, Z. Wang, L. Xie, Cross-speaker emotion disentangling and transfer for end-to-end speech synthesis. IEEE/ACM Trans. Audio Speech Lang. Process. **30**, 1448–1460 (2022)
35. N.Q. Wu, Z.C. Liu, Z.H. Ling, in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Discourse-level prosody modeling with a variational autoencoder for non-autoregressive expressive speech synthesis (IEEE, 2022), pp. 7592–7596. https://doi.org/10.1109/icassp43922.2022.9746238
36. K. He, C. Sun, R. Zhu, L. Zhao, in *2022 7th International Conference on Intelligent Computing and Signal Processing (ICSP)*. Multi-speaker emotional speech synthesis with limited datasets: Two-stage non-parallel training strategy (IEEE, 2022), pp. 545–548. https://doi.org/10.1109/icsp54964.2022.9778768
37. L. Xue, F.K. Soong, S. Zhang, L. Xie, Paratts: Learning linguistic and prosodic cross-sentence information in paragraph-based tts. IEEE/ACM Trans. Audio Speech Lang. Process. **30**, 2854–2864 (2022)
38. Y. Lei, S. Yang, X. Zhu, L. Xie, D. Su, Cross-speaker emotion transfer through information perturbation in emotional speech synthesis. IEEE Signal Process. Lett. **29**, 1948–1952 (2022)
39. T. Li, X. Wang, Q. Xie, Z. Wang, M. Jiang, L. Xie, Cross-speaker emotion transfer based on prosody compensation for end-to-end speech synthesis. IEEE/ACM Trans. Audio Speech Lang. Process. **30**, 1448–1460 (2022). arXiv preprint arXiv:2207.01198
40. Y. Wu, X. Wang, S. Zhang, L. He, R. Song, J.Y. Nie, Self-supervised context-aware style representation for expressive speech synthesis. Proc. Annu. Conf. Int. Speech Commun. Assoc. pp. 5503–5507 (2022). arXiv preprint arXiv:2206.12559
41. R. Li, Z. Wu, Y. Huang, J. Jia, H. Meng, L. Cai, in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Emphatic speech generation with conditioned input layer and bidirectional lstms for expressive speech synthesis (IEEE, 2018), pp. 5129–5133
42. X. Wu, L. Sun, S. Kang, S. Liu, Z. Wu, X. Liu, H. Meng, in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Feature based adaptation for speaking style synthesis (IEEE, 2018), pp. 5304–5308. https://doi.org/10.1109/icassp.2018.8462178
43. L. Xue, X. Zhu, X. An, L. Xie, in *Proceedings of the Joint Workshop of the 4th Workshop on Affective Social Multimedia Computing and first Multi-Modal Affective Computing of Large-Scale Multimedia Data (ASMMC-MMAC)*. A comparison of expressive speech synthesis approaches based on neural network (ACM, 2018), pp. 15–20. https://doi.org/10.1145/3267935.3267947
44. Z. Zeng, J. Wang, N. Cheng, J. Xiao, in *Proc. Interspeech 2020*. Prosody learning mechanism for speech synthesis system without text length limit, vol. 2020 (2020), pp. 4422–4426. arXiv preprint arXiv:2008.05656
45. F. Yang, S. Yang, Q. Wu, Y. Wang, L. Xie, in *Proc. Interspeech 2020*. Exploiting deep sentential context for expressive end-to-end speech synthesis., vol. 2020 (2020), pp. 3436–3440. arXiv preprint arXiv:2008.00613
46. Y.J. Zhang, Z.H. Ling, Extracting and predicting word-level style variations for speech synthesis. IEEE/ACM Trans. Audio Speech Lang. Process. **29**, 1582–1593 (2021)
47. C. Lu, X. Wen, R. Liu, X. Chen, in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Multi-speaker emotional speech synthesis with fine-grained prosody modeling (IEEE, 2021), pp. 5729–5733. https://doi.org/10.1109/icassp39728.2021.9413398
48. C. Gong, L. Wang, Z. Ling, S. Guo, J. Zhang, J. Dang, in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Improving naturalness and controllability of sequence-to-sequence speech synthesis by learning local prosody representations (IEEE, 2021), pp. 5724–5728. https://doi.org/10.1109/icassp39728.2021.9414720
49. X. Li, C. Song, J. Li, Z. Wu, J. Jia, H. Meng, Towards multi-scale style control for expressive speech synthesis. (2021). arXiv preprint arXiv:2104.03521
50. S. Lei, Y. Zhou, L. Chen, Z. Wu, S. Kang, H. Meng, in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Towards expressive speaking style modelling with hierarchical context information for mandarin speech synthesis (IEEE, 2022), pp. 7922–7926. https://doi.org/10.1109/icassp43922.2022.9747438
51. F. Yang, J. Luan, Y. Wang, in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Improving emotional speech synthesis by using sus-constrained vae and text encoder aggregation (IEEE, 2022), pp. 8302–8306. https://doi.org/10.1109/icassp43922.2022.9746994
52. R. Li, D. Pu, M. Huang, B. Huang, in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Unet-tts: Improving unseen speaker and style transfer in one-shot voice cloning (IEEE, 2022), pp. 8327–8331. https://doi.org/10.1109/icassp43922.2022.9746049
53. Y. Wang, Y. Xie, K. Zhao, H. Wang, Q. Zhang, in *2022 IEEE International Conference on Multimedia and Expo (ICME)*. Unsupervised quantized prosody representation for controllable speech synthesis (IEEE, 2022), pp. 1–6. https://doi.org/10.1109/icme52920.2022.9859946
54. Y. Zhou, C. Song, J. Li, Z. Wu, Y. Bian, D. Su, H. Meng, in *Proc. Interspeech 2022*. Enhancing word-level semantic representation via dependency structure for expressive text-to-speech synthesis, vol. 2022 (2022), pp. 5518–5522. arXiv preprint arXiv:2104.06835
55. Y. Lee, T. Kim, in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Robust and fine-grained prosody control of end-to-end speech synthesis (IEEE, 2019), pp. 5911–5915. https://doi.org/10.1109/icassp.2019.8683501
56. H. Choi, S. Park, J. Park, M. Hahn, in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Multi-speaker emotional acoustic modeling for cnn-based speech synthesis (IEEE, 2019), pp. 6950–6954. https://doi.org/10.1109/icassp.2019.8683682
57. O. Kwon, I. Jang, C. Ahn, H.G. Kang, An effective style token weight control technique for end-to-end emotional speech synthesis. IEEE Signal Process. Lett. **26**(9), 1383–1387 (2019)
58. S.Y. Um, S. Oh, K. Byun, I. Jang, C. Ahn, H.G. Kang, in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Emotional speech synthesis with rich and granularized control (IEEE, 2020), pp. 7254–7258. https://doi.org/10.1109/icassp40776.2020.9053732
59. M. Kim, S.J. Cheon, B.J. Choi, J.J. Kim, N.S. Kim, in *Proc. ISCA Interspeech 2021*. Expressive text-to-speech using style tag, vol. 2021 (2021), pp. 4663–4667. arXiv preprint arXiv:2104.00436
60. S. Moon, S. Kim, Y.H. Choi, Mist-tacotron: end-to-end emotional speech synthesis using mel-spectrogram image style transfer. IEEE Access **10**, 25455–25463 (2022)
61. C.B. Im, S.H. Lee, S.B. Kim, S.W. Lee, in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Emoq-tts: Emotion intensity quantization for fine-grained controllable emotional text-to-speech (IEEE, 2022), pp. 6317–6321. https://doi.org/10.1109/icassp43922.2022.9747098
62. Y. Shin, Y. Lee, S. Jo, Y. Hwang, T. Kim, in *Proc. Interspeech 2022*. Text-driven Emotional Style Control and Cross-speaker Style Transfer in Neural TTS (2022), pp. 2313–2317. https://doi.org/10.21437/Interspeech.2022-10131

63. C. Kim, S.Y. Um, H. Yoon, H.G. Kang, in *Proc. Interspeech 2022*. Fluenttts: Text-dependent fine-grained style control for multi-style tts, vol. 2022 (2022), pp. 4561–4565. https://doi.org/10.21437/Interspeech.2022-988
64. H.W. Yoon, O. Kwon, H. Lee, R. Yamamoto, E. Song, J.M. Kim, M.J. Hwang, in *Proc. Interspeech 2022*. Language Model-Based Emotion Prediction Methods for Emotional Speech Synthesis Systems (2022), pp. 4596–4600. https://doi.org/10.21437/Interspeech.2022-11133
65. K. Inoue, S. Hara, M. Abe, N. Hojo, Y. Ijima, in *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. An investigation to transplant emotional expressions in dnn-based tts synthesis (IEEE, 2017), pp. 1253–1258. https://doi.org/10.1109/apsipa.2017.8282231
66. J. Lorenzo-Trueba, G.E. Henter, S. Takaki, J. Yamagishi, Y. Morino, Y. Ochiai, Investigating different representations for modeling and controlling multiple emotions in dnn-based speech synthesis. Speech Commun. **99**, 135–143 (2018)
67. T. Koriyama, T. Kobayashi, in *Proc. Interspeech 2019*. Semi-supervised prosody modeling using deep gaussian process latent variable model. (2019), pp. 4450–4454. https://doi.org/10.21437/Interspeech.2019-2497
68. Y. Hono, K. Tsuboi, K. Sawada, K. Hashimoto, K. Oura, Y. Nankaku, K. Tokuda, in *Proc. ISCA Interspeech 2020*. Hierarchical multi-grained generative model for expressive speech synthesis, vol. 2020 (2020), pp. 3441–3445. arXiv preprint arXiv:2009.08474
69. K. Inoue, S. Hara, M. Abe, N. Hojo, Y. Ijima, Model architectures to extrapolate emotional expressions in dnn-based text-to-speech. Speech Commun. **126**, 35–43 (2021)
70. W. Nakata, T. Koriyama, S. Takamichi, Y. Saito, Y. Ijima, R. Masumura, H. Saruwatari, in *Proc. Interspeech 2022*. Predicting VQVAE-based Character Acting Style from Quotation-Annotated Text for Audiobook Speech Synthesis (2022), pp. 4551–4555. https://doi.org/10.21437/Interspeech.2022-638
71. D.S.R. Mohan, V. Hu, T.H. Teh, A. Torresquintero, C.G. Wallis, M. Staib, L. Foglianti, J. Gao, S. King, in *Interspeech 2021*. Ctrl-p: Temporal control of prosodic variation for speech synthesis, vol. 2021 (2021), pp. 3875–3879. arXiv preprint arXiv:2106.08352
72. G. Pamisetty, K. Sri Rama Murty, Prosody-tts: An end-to-end speech synthesis system with prosody control. Circ. Syst. Signal Process. **42**(1), 361–384 (2023)
73. L. Zhao, J. Yang, Q. Qin, in *2020 3rd International Conference on Algorithms (ACAI '20), Computing and Artificial Intelligence*. Enhancing prosodic features by adopting pre-trained language model in bahasa indonesia speech synthesis (ACM, 2020), pp. 1–6. https://doi.org/10.48550/arXiv.2102.00184
74. R. Skerry-Ryan, E. Battenberg, Y. Xiao, Y. Wang, D. Stanton, J. Shor, R. Weiss, R. Clark, R.A. Saurous, in *international conference on machine learning*. Towards end-to-end prosody transfer for expressive speech synthesis with tacotron (PMLR, 2018), pp. 4693–4702. https://proceedings.mlr.press/v80/skerry-ryan18a.html
75. Y. Wang, D. Stanton, Y. Zhang, R.S. Ryan, E. Battenberg, J. Shor, Y. Xiao, Y. Jia, F. Ren, R.A. Saurous, in *International Conference on Machine Learning*. Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis (PMLR, 2018), pp. 5180–5189. https://proceedings.mlr.press/v80/wang18h.html
76. K. Akuzawa, Y. Iwasawa, Y. Matsuo, Expressive speech synthesis via modeling expressions with variational autoencoder. (2018). arXiv preprint arXiv:1804.02135
77. Y.J. Zhang, S. Pan, L. He, Z.H. Ling, in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Learning latent representations for style control and transfer in end-to-end speech synthesis (IEEE, 2019), pp. 6945–6949. https://doi.org/10.1109/icassp.2019.8683623
78. S. Suzié, T. Nosek, M. Sečujski, D. Pekar, V. Delié, in *2019 27th Telecommunications Forum (TELFOR)*. Dnn based expressive text-to-speech with limited training data (IEEE, 2019), pp. 1–6. https://doi.org/10.1109/telfor48224.2019.8971351
79. T. Cornille, F. Wang, J. Bekker, in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Interactive multi-level prosody control for expressive speech synthesis (IEEE, 2022), pp. 8312–8316. https://doi.org/10.1109/icassp43922.2022.9746654
80. S. Suzic, T.V. Delic, S. Ostrogonac, S. Duric, D.J. Pekar, Style-code method for multi-style parametric text-to-speech synthesis. SPIIRAS Proc. **5**(60), 216 (2018). https://doi.org/10.15622/sp.60.8
81. J. Parker, Y. Stylianou, R. Cipolla, in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Adaptation of an expressive single speaker deep neural network speech synthesis system (IEEE, 2018), pp. 5309–5313. https://doi.org/10.1109/icassp.2018.8461888
82. S. Suzić, T. Delić, D. Pekar, V. Delić, M. Sečujski, Style transplantation in neural network based speech synthesis. Acta Polytech. Hungarica **16**(6), 171–189 (2019)
83. N. Prateek, M. Łajszczak, R. Barra-Chicote, T. Drugman, J. Lorenzo-Trueba, T. Merritt, S. Ronanki, T. Wood, in *NAACL HLT 2019*. In other news: A bi-style text-to-speech model for synthesizing newscaster voice with limited data. (2019). arXiv preprint arXiv:1904.02790
84. M. Secujski, D. Pekar, S. Suzic, A. Smirnov, T.V. Nosek, Speaker/style-dependent neural network speech synthesis based on speaker/style embedding. J. Univers. Comput. Sci. **26**(4), 434–453 (2020)
85. Y. Gao, W. Zheng, Z. Yang, T. Kohler, C. Fuegen, Q. He, in *Proc. Interspeech 2020*. Interactive text-to-speech system via joint style analysis, vol. 2020 (2020), pp. 4447–4451. arXiv preprint arXiv:2002.06758
86. S. Pan, L. He, in *Proc. Annu. Conf. INTERSPEECH 2021*. Cross-speaker style transfer with prosody bottleneck in neural speech synthesis, vol. 2021 (2021), pp. 4678–4682. arXiv preprint arXiv:2107.12562
87. J. He, C. Gong, L. Wang, D. Jin, X. Wang, J. Xu, J. Dang, in *Proc. Interspeech 2022*. Improve emotional speech synthesis quality by learning explicit and implicit representations with semi-supervised training (2022), pp. 5538–5542. https://doi.org/10.21437/Interspeech.2022-11336
88. T. Brown, B. Mann, N. Ryder, M. Subbiah, J.D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell et al., Language models are few-shot learners. Adv. Neural Inf. Process. Syst. **33**, 1877–1901 (2020)
89. D. Parikh, K. Grauman, in *2011 International Conference on Computer Vision*. Relative attributes (IEEE, 2011), pp. 503–510. https://doi.org/10.1109/iccv.2011.6126281
90. Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, V. Lempitsky, Domain-adversarial training of neural networks. J. Mach. Learn. Res. **17**(1), 2096–2030 (2016)
91. R. Liu, B. Sisman, G. Gao, H. Li, Expressive tts training with frame and style reconstruction loss. IEEE/ACM Trans. Audio Speech Lang. Process. **29**, 1806–1818 (2021)
92. R. Liu, B. Sisman, H. Li, in *Proc. Annu. Conf. Int. Speech Commun. Assoc. 2021*. Reinforcement learning for emotional text-to-speech synthesis with improved emotion discriminability. (2021). pp. 4648-4652. arXiv preprint arXiv:2104.01408
93. X. Dai, C. Gong, L. Wang, K. Zhang, Information sieve: Content leakage reduction in end-to-end prosody for expressive speech synthesis. (2021). arXiv preprint arXiv:2108.01831
94. D. Stanton, Y. Wang, R. Skerry-Ryan, in *2018 IEEE Spoken Language Technology Workshop (SLT)*. Predicting expressive speaking style from text in end-to-end speech synthesis (IEEE, 2018), pp. 595–602. https://doi.org/10.1109/slt.2018.8639682
95. C. Du, K. Yu, in *Proc. ISCA Interspeech 2021*. Rich prosody diversity modelling with phone-level mixture density network, vol. 2021 (2021), pp. 3136–3140. arXiv preprint arXiv:2102.00851
96. Z. Lyu, J. Zhu, in *2022 12th International Conference on Information Science and Technology (ICIST)*. Enriching style transfer in multi-scale control based personalized end-to-end speech synthesis (IEEE, 2022), pp. 114–119. https://doi.org/10.1109/icist55546.2022.9926908
97. K. Lee, K. Park, D. Kim, in *Proc. Interspeech 2021*. Styler: Style factor modeling with rapidity and robustness via speech decomposition for expressive and controllable neural text to speech, vol. 2021 (2021), pp. 4643–4647. arXiv preprint arXiv:2103.09474
98. S.H. Lee, H.W. Yoon, H.R. Noh, J.H. Kim, S.W. Lee, in *Proceedings of the AAAI Conference on Artificial Intelligence*. Multi-spectrogan: High-diversity and high-fidelity spectrogram generation with adversarial style combination for speech synthesis, AAAI, vol. 35 (2021), pp. 13198–13206. https://doi.org/10.1609/aaai.v35i14.17559
99. X. Luo, S. Takamichi, T. Koriyama, Y. Saito, H. Saruwatari, in *2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. Emotion-controllable speech synthesis using

emotion soft labels and fine-grained prosody factors (IEEE, 2021), pp. 794–799

100. C. Gong, L. Wang, Z. Ling, J. Zhang, J. Dang, in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Using multiple reference audios and style embedding constraints for speech synthesis (IEEE, 2022), pp. 7912–7916. https://doi.org/10.1109/icassp43922.2022.9747801
101. S. Liang, C. Miao, M. Chen, J. Ma, S. Wang, J. Xiao, in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Unsupervised learning for multi-style speech synthesis with limited data (IEEE, 2021), pp. 6583–6587. https://doi.org/10.1109/icassp39728.2021.9414220
102. K. Zhang, C. Gong, W. Lu, L. Wang, J. Wei, D. Liu, in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Joint and adversarial training with asr for expressive speech synthesis (IEEE, 2022), pp. 6322–6326. https://doi.org/10.1109/icassp43922.2022.9746442
103. T. Raitio, R. Rasipuram, D. Castellani, in *Interspeech 2020*. Controllable neural text-to-speech synthesis using intuitive prosodic features, vol. 2020 (2020), pp. 4432–4436. arXiv preprint arXiv:2009.06775
104. D.R. Liu, C.Y. Yang, S.L. Wu, H.Y. Lee, in *2018 IEEE Spoken Language Technology Workshop (SLT)*. Improving unsupervised style transfer in end-to-end speech synthesis with end-to-end speech recognition (IEEE, 2018), pp. 640–647. https://doi.org/10.1109/slt.2018.8639672
105. X. Cai, D. Dai, Z. Wu, X. Li, J. Li, H. Meng, in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Emotion controllable speech synthesis using emotion-unlabeled dataset with the assistance of cross-domain speech emotion recognition (IEEE, 2021), pp. 5734–5738. https://doi.org/10.1109/icassp39728.2021.9413907
106. R. Chung, B. Mak, in *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. On-the-fly data augmentation for text-to-speech style transfer (IEEE, 2021), pp. 634–641. https://doi.org/10.1109/asru51503.2021.9688074
107. L. Xue, S. Pan, L. He, L. Xie, F.K. Soong, Cycle consistent network for end-to-end style transfer tts training. Neural Netw. **140**, 223–236 (2021)
108. S.J. Cheon, J.Y. Lee, B.J. Choi, H. Lee, N.S. Kim, Gated recurrent attention for multi-style speech synthesis. Appl. Sci. **10**(15), 5325 (2020)
109. T. Kenter, V. Wan, C.A. Chan, R. Clark, J. Vit, in *International Conference on Machine Learning*. Chive: Varying prosody in speech synthesis with a linguistically driven dynamic hierarchical conditional variational network (PMLR, 2019), pp. 3331–3340. https://proceedings.mlr.press/v97/kenter19a.html
110. T. Kenter, M.K. Sharma, R. Clark, in Proc. Interspeech 2020. Improving prosody of rnn-based english text-to-speech synthesis by incorporating a bert model, vol 2020 (2020), pp. 4412–4416
111. D. Tan, T. Lee, in *Proc. Annu. Conf. Int. Speech Commun. Assoc. 2020*. Fine-grained style modeling, transfer and prediction in text-to-speech synthesis via phone-level content-style disentanglement, vol. 2020 (2020), pp. 4683–4687. arXiv preprint arXiv:2011.03943
112. N. Ellinas, M. Christidou, A. Vioni, J.S. Sung, A. Chalamandaris, P. Tsiakoulis, P. Mastorocostas, Controllable speech synthesis by learning discrete phoneme-level prosodic representations. Speech Commun. **146**, 22–31 (2023)
113. A. Vioni, M. Christidou, N. Ellinas, G. Vamvoukakis, P. Kakoulidis, T. Kim, J.S. Sung, H. Park, A. Chalamandaris, P. Tsiakoulis, in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Prosodic clustering for phoneme-level prosody control in end-to-end speech synthesis (IEEE, 2021), pp. 5719–5723. https://doi.org/10.1109/ICASSP39728.2021.9413604
114. R. Valle, J. Li, R. Prenger, B. Catanzaro, in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Mellotron: Multispeaker expressive voice synthesis by conditioning on rhythm, pitch and global style tokens (IEEE, 2020), pp. 6189–6193. https://doi.org/10.1109/ICASSP40776.2020.9054556
115. G. Huybrechts, T. Merritt, G. Comini, B. Perz, R. Shah, J. Lorenzo-Trueba, in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Low-resource expressive text-to-speech using data augmentation (IEEE, 2021), pp. 6593–6597. https://doi.org/10.1109/ICASSP39728.2021.9413466
116. Y. Guo, C. Du, K. Yu, in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Unsupervised word-level prosody tagging for controllable speech synthesis (IEEE, 2022), pp. 7597–7601. https://doi.org/10.1109/ICASSP43922.2022.9746323
117. D. Paul, S. Mukherjee, Y. Pantazis, Y. Stylianou, in *Interspeech 2021*. A universal multi-speaker multi-style text-to-speech via disentangled representation learning based on rényi divergence minimization (2021), pp. 3625–3629. https://doi.org/10.21437/Interspeech.2021-660
118. J. Zaïdi, H. Seuté, B. van Niekerk, M.A. Carbonneau, in *Proc. Interspeech 2022*. Daft-exprt: Cross-speaker prosody transfer on any text for expressive speech synthesis, vol. 2022 (2021), pp. 4591–4595. arXiv preprint arXiv:2108.02271
119. V. Aggarwal, M. Cotescu, N. Prateek, J. Lorenzo-Trueba, R. Barra-Chicote, in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Using vaes and normalizing flows for one-shot text-to-speech synthesis of expressive speech (IEEE, 2020), pp. 6179–6183. https://doi.org/10.1109/icassp40776.2020.9053678
120. L.W. Chen, A. Rudnicky, in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Fine-grained style control in transformer-based text-to-speech synthesis (IEEE, 2022), pp. 7907–7911. https://doi.org/10.1109/icassp43922.2022.9747747
121. X. Wu, Y. Cao, M. Wang, S. Liu, S. Kang, Z. Wu, X. Liu, D. Su, D. Yu, H. Meng, in *Interspeech 2018*. Rapid style adaptation using residual error embedding for expressive speech synthesis. (2018), pp. 3072–3076. https://doi.org/10.21437/Interspeech.2018-1991
122. G. Zhang, Y. Qin, T. Lee, in *Interspeech 2020* Learning syllable-level discrete prosodic representation for expressive speech generation (2020), pp. 3426–3430. https://doi.org/10.21437/Interspeech.2020-2228
123. G. Sun, Y. Zhang, R.J. Weiss, Y. Cao, H. Zen, Y. Wu, in *ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. Fully-hierarchical fine-grained prosody modeling for interpretable speech synthesis (IEEE, 2020), pp. 6264–6268. https://doi.org/10.1109/icassp40776.2020.9053520
124. A. Suni, S. Kakouros, M. Vainio, J. Šimko, in *10th International Conference on Speech Prosody 2020*. Prosodic prominence and boundaries in sequence-to-sequence speech synthesis. (2020). pp. 940–944. arXiv preprint arXiv:2006.15967
125. Y. Zou, S. Liu, X. Yin, H. Lin, C. Wang, H. Zhang, Z. Ma, in *Proc. Interspeech 2021*. Fine-Grained Prosody Modeling in Neural Speech Synthesis Using ToBI Representation (2021), pp. 3146–3150. https://doi.org/10.21437/Interspeech.2021-883
126. I. Vallés-Pérez, J. Roth, G. Beringer, R. Barra-Chicote, J. Droppo, in *Interspeech 2021*. Improving multi-speaker tts prosody variance with a residual encoder and normalizing flows, vol. 2021 (2021), pp. 3131–3135. arXiv preprint arXiv:2106.05762
127. Z. Hodari, A. Moinet, S. Karlapati, J. Lorenzo-Trueba, T. Merritt, A. Joly, A. Abbas, P. Karanasou, T. Drugman, in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Camp: a two-stage approach to modelling prosody in context (IEEE, 2021), pp. 6578–6582. https://doi.org/10.1109/icassp39728.2021.9414413
128. T. Raitio, J. Li, S. Seshadri, in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Hierarchical prosody modeling and control in non-autoregressive parallel neural tts (IEEE, 2022), pp. 7587–7591. https://doi.org/10.1109/icassp43922.2022.9746253
129. S. Karlapati, A. Abbas, Z. Hodari, A. Moinet, A. Joly, P. Karanasou, T. Drugman, in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Prosodic representation learning and contextual sampling for neural text-to-speech (IEEE, 2021), pp. 6573–6577. https://doi.org/10.1109/icassp39728.2021.9413696
130. S. Karlapati, A. Moinet, A. Joly, V. Klimkov, D. Sáez-Trigueros, T. Drugman, in *Proc. Interspeech 2020*. Copycat: Many-to-many fine-grained prosody transfer for neural text-to-speech, vol. 2020 (2020), pp. 4387–4391. arXiv preprint arXiv:2004.14617
131. S. Tyagi, M. Nicolis, J. Rohnke, T. Drugman, J. Lorenzo-Trueba, in *Proc. Annu. Conf. Int. Speech Commun. Assoc. 2019*. Dynamic prosody generation for speech synthesis using linguistics-driven acoustic embedding selection. (2019). pp. 4407–4411. arXiv preprint arXiv:1912.00955
132. Y. Yan, X. Tan, B. Li, G. Zhang, T. Qin, S. Zhao, Y. Shen, W.Q. Zhang, T.Y. Liu, in *INTERSPEECH 2021*. Adaspeech 3: Adaptive text to speech for spontaneous style, vol. 2021 (2021), pp. 1–5. arXiv preprint arXiv:2107.02530

133. X. An, Y. Wang, S. Yang, Z. Ma, L. Xie, in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. Learning hierarchical representations for expressive speaking style in end-to-end speech synthesis (IEEE, 2019), pp. 184–191. https://doi.org/10.1109/asru46091.2019.9003859
134. Y. Feng, P. Duan, Y. Zi, Y. Chen, S. Xiong, in *2022 IEEE International Conference on Multimedia and Expo (ICME)*. Fusing acoustic and text emotional features for expressive speech synthesis (IEEE, 2022), pp. 01–06. https://doi.org/10.1109/icme52920.2022.9859769
135. I. Jauk, J. Lorenzo Trueba, J. Yamagishi, A. Bonafonte Cávez, in *Interspeech 2018: 2-6 September 2018, Hyderabad*. Expressive speech synthesis using sentiment embeddings (International Speech Communication Association (ISCA), 2018), pp. 3062–3066. https://doi.org/10.21437/interspeech.2018-2467
136. J. Li, Y. Meng, C. Li, Z. Wu, H. Meng, C. Weng, D. Su, in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Enhancing speaking styles in conversational text-to-speech synthesis with graph-based multi-modal context modeling (IEEE, 2022), pp. 7917–7921. https://doi.org/10.1109/icassp43922.2022.9747837
137. T.Y. Hu, A. Shrivastava, O. Tuzel, C. Dhir, in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Unsupervised style and content separation by minimizing mutual information for speech synthesis (IEEE, 2020), pp. 3267–3271. https://doi.org/10.1109/icassp40776.2020.9054591
138. M. Morrison, Z. Jin, J. Salamon, N.J. Bryan, G.J. Mysore, in *Proc. Interspeech 2020*. Controllable neural prosody synthesis, vol. 2020 (2020), 4437–4441. arXiv preprint arXiv:2008.03388
139. F. Eyben, F. Weninger, F. Gross, B. Schuller, in *Proceedings of the 21st ACM international conference on Multimedia*. Recent developments in opensmile, the munich open-source multimedia feature extractor (ACM, 2013), pp. 835–838. https://doi.org/10.1145/2502081.2502224
140. M. Morise, F. Yokomori, K. Ozawa, World: a vocoder-based high-quality speech synthesis system for real-time applications. IEICE Trans. Inf. Syst. **99**(7), 1877–1884 (2016)
141. E. Perez, F. Strub, H. De Vries, V. Dumoulin, A. Courville, in *Proceedings of the AAAI Conference on Artificial Intelligence*. Film: Visual reasoning with a general conditioning layer, AAAI, vol. 32 (2018). https://doi.org/10.1609/aaai.v32i1.11671
142. A. Baevski, Y. Zhou, A. Mohamed, M. Auli, wav2vec 2.0: A framework for self-supervised learning of speech representations. Adv. Neural Inf. Process Syst. **33**, 12449–12460 (2020)
143. J. Kim, J. Kong, J. Son, in *International Conference on Machine Learning*. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech (PMLR, 2021), pp. 5530–5540. https://proceedings.mlr.press/v139/kim21f.html
144. L.A. Gatys, A.S. Ecker, M. Bethge, in *Proceedings of the IEEE conference on computer vision and pattern recognition*. Image style transfer using convolutional neural networks (IEEE, 2016), pp. 2414–2423. https://doi.org/10.1109/CVPR.2016.265
145. K. Simonyan, A. Zisserman, in *ICLR 2015*. Very deep convolutional networks for large-scale image recognition. (2015). arXiv preprint arXiv:1409.1556
146. D.P. Kingma, M. Welling, in *ICLR 2014*. Auto-encoding variational bayes. (2014). arXiv preprint arXiv:1312.6114
147. Y. Taigman, L. Wolf, A. Polyak, E. Nachmani, in *ICLR 2018*. Voiceloop: Voice fitting and synthesis via a phonological loop. (2018). arXiv preprint arXiv:1707.06588
148. J. Devlin, M.W. Chang, K. Lee, K. Toutanova, in *Proceedings of NAACL 2019*. Bert: Pre-training of deep bidirectional transformers for language understanding. (2019). pp. 4171–4186. arXiv preprint arXiv:1810.04805
149. A. Van Den Oord, O. Vinyals et al., Neural discrete representation learning. Adv. Neural Inf. Process. Syst. **30**, 6306–6315 (2017)
150. Z. Xiao, K. Kreis, A. Vahdat, in *International Conference on Learning Representations 2022*. Tackling the generative learning trilemma with denoising diffusion gans. (2022). arXiv preprint arXiv:2112.07804
151. A. Défossez, J. Copet, G. Synnaeve, Y. Adi, High fidelity neural audio compression. (2022). arXiv preprint arXiv:2210.13438
152. J. Hu, L. Shen, G. Sun, in *Proceedings of the IEEE conference on computer vision and pattern recognition*. Squeeze-and-excitation networks (IEEE, 2018), pp. 7132–7141. https://doi.org/10.1109/cvpr.2018.00745
153. K. Qian, Y. Zhang, S. Chang, X. Yang, M. Hasegawa-Johnson, in *International Conference on Machine Learning*. Autovc: Zero-shot voice style transfer with only autoencoder loss (PMLR, 2019), pp. 5210–5219. https://proceedings.mlr.press/v97/qian19c.html
154. A.A. Alemi, I. Fischer, J.V. Dillon, K. Murphy, in *Proc. Int. Conf. Learn. Representations 2017*. Deep variational information bottleneck. (2017). arXiv preprint arXiv:1612.00410
155. S. Ioffe, C. Szegedy, in *International conference on machine learning*. Batch normalization: Accelerating deep network training by reducing internal covariate shift (PMLR, 2015), pp. 448–456. https://proceedings.mlr.press/v37/ioffe15.html
156. D. Ulyanov, A. Vedaldi, V. Lempitsky, in *Proceedings of the IEEE conference on computer vision and pattern recognition*. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis (IEEE, 2017), pp. 6924–6932. https://doi.org/10.1109/cvpr.2017.437
157. M.I. Belghazi, A. Baratin, S. Rajeshwar, S. Ozair, Y. Bengio, A. Courville, D. Hjelm, in *International conference on machine learning*. Mutual information neural estimation (PMLR, 2018), pp. 531–540. https://proceedings.mlr.press/v80/belghazi18a.html
158. P. Cheng, W. Hao, S. Dai, J. Liu, Z. Gan, L. Carin, in *International conference on machine learning*. Club: A contrastive log-ratio upper bound of mutual information (PMLR, 2020), pp. 1779–1788. https://proceedings.mlr.press/v119/cheng20b.html
159. W.N. Hsu, B. Bolte, Y.H.H. Tsai, K. Lakhotia, R. Salakhutdinov, A. Mohamed, Hubert: Self-supervised speech representation learning by masked prediction of hidden units. IEEE/ACM Trans. Audio Speech Lang. Process. **29**, 3451–3460 (2021)
160. Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R.R. Salakhutdinov, Q.V. Le, Xlnet: Generalized autoregressive pretraining for language understanding. Adv. Neural Inf. Process. Syst. **32**, 5753–5763 (2019)
161. C.M. Bishop, Technical report, Aston University 1994. Mixture density networks (1994)
162. Y. Shen, Z. Lin, C.W. Huang, A. Courville, in *Proceedings of ICLR 2018*. Neural language modeling by jointly learning syntax and lexicon. (2018). arXiv preprint arXiv:1711.02013
163. J. Sarzynska-Wawer, A. Wawer, A. Pawlak, J. Szymanowska, I. Stefaniak, M. Jarkiewicz, L. Okruszek, Detecting formal thought disorder by deep contextualized word representations. Psychiatry Res. **304**, 114135 (2021)
164. K. Clark, M.T. Luong, Q.V. Le, C.D. Manning, in *ICLR 2020*. Electra: Pre-training text encoders as discriminators rather than generators. (2020). arXiv preprint arXiv:2003.10555
165. C. Li, X. Ma, B. Jiang, X. Li, X. Zhang, X. Liu, Y. Cao, A. Kannan, Z. Zhu, Deep speaker: an end-to-end neural speaker embedding system. (2017). arXiv preprint arXiv:1705.02304
166. M. Azab, N. Kojima, J. Deng, R. Mihalcea, in *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*. Representing movie characters in dialogues. Association for Computational Linguistics, Hong Kong, China. (2019), pp. 99–109. https://doi.org/10.18653/v1/K19-1010
167. Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach. (2019). arXiv preprint arXiv:1907.11692
168. A. Suni, J. Šimko, D. Aalto, M. Vainio, Hierarchical representation and estimation of prosody using continuous wavelet transform. Comput. Speech Lang. **45**, 123–136 (2017)
169. J.M. Tomczak, M. Welling, in *NIPS Workshop: Bayesian Deep Learning 2016*. Improving variational auto-encoders using householder flow. (2016). arXiv preprint arXiv:1611.09630
170. K. Zhou, B. Sisman, R. Liu, H. Li, Emotional voice conversion: Theory, databases and esd. Speech Commun. **137**, 1–18 (2022)
171. cstr. The blizzard challenge. https://www.cstr.ed.ac.uk/projects/blizzard/. Accessed 15 Sept 2023
172. K. Ito, L. Johnson, The lj speech dataset. (2017). https://keithito.com/LJ-Speech-Dataset/. Accessed 15 Sept 2023
173. cstr. Voice cloning toolkit. https://datashare.ed.ac.uk/handle/10283/3443. Accessed 15 Sept 2023
174. H. Zen, R. Clark, R.J. Weiss, V. Dang, Y. Jia, Y. Wu, Y. Zhang, Z. Chen, in *Interspeech*. Libritts: A corpus derived from librispeech for text-to-speech (2019). https://arxiv.org/abs/1904.02882. Accessed 15 Sept 2023

175. Wikipedia. Emotion classification - Wikipedia — en.wikipedia.org. https://en.wikipedia.org/wiki/Emotion_classification. Accessed 30 May 2023
176. M.M. Bradley, M.K. Greenwald, M.C. Petry, P.J. Lang, Remembering pictures: pleasure and arousal in memory. J. Exp. Psychol. Learn. Mem. Cogn. **18**(2), 379 (1992)
177. J.A. Russell, A circumplex model of affect. J. Pers. Soc. Psychol. **39**(6), 1161 (1980)
178. P. Ekman, E. Revealed, Emotions revealed: Recognizing faces and feelings to improve communication and emotional life. (Holt Paperback, 2003), vol. 128, no. 8, pp. 140–140

Publisher's Note