Proceedings of the 4th International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/2026.37211

Discrete Global Style Tokens for Expressive Text-to-Speech Modelling

Dzhan Maks He

Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Canada
dmhe@uwaterloo.ca

Abstract. Text-to-speech systems typically lack emotional expressiveness, and systems capable of expressive synthesis rely either on inferring emotions from text or accepting human-written labels from a small handful of emotion classes that are trained into the model. This work proposes an alternative based on learning a coarse discrete sentence-level representation for speech style, in the form of style tokens, learned in an unsupervised fashion, eliminating the need to pick a small number of emotion classes to model ahead of time. This can be used to augment an existing corpus with style labels to efficiently fine-tune existing text-to-speech models to condition speech generation on speaking style without the need for significant modification for training pipelines. The resulting representations and models can be used to perform speech style transfer using a reference clip as a source of style tokens to append to target text, achieving example-based speech style transfer without relying on predetermined style categories.

Keywords: Text-to-speech, representation learning, style transfer

1. Introduction

Most text-to-speech models are trained on datasets such as Librispeech [1], where the majority of speech is read in a relatively neutral tone. Some models train on expressive datasets, but if the input text provides no explicit conditioning for speaking style, the model is forced to infer the speaking style implied by the text, which does not provide any explicit controllability of the output style. This is an especially significant limitation when many speech synthesis models have limited ability to infer emotions due to possessing limited world knowledge and reasoning ability. Many also only support a very small context length which limits how much information the model can make use of to infer emotions [2, 3]. To address this, some models introduced explicit labels and directions that the user can insert into text prompts to direct the generated audio's style, however such directions are limited to a small number of classes that are available in the training data, and textual natural-language-based directions possess inherent limitations when it comes to portrayal of acoustic details.

These limitations can be eliminated by learning a style embedding instead of using fixed supervised labels. Existing attempts to inject style embeddings into TTS systems, such as Global Style Tokens [4], have demonstrated that tokenized embeddings can condition a speech model with a small global utterance-level embedding to produce stylized speech. In the original GST work, style embeddings are constructed as a weighted sum of a fixed set of style tokens using attention which

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2026.37211

are trained end-to-end within a Tacotron system. Prosody of generated speech can then be controlled using style transfer, requiring a reference audio clip as a source. Alternatively, an embedding can construct manually as a weighted sum of the learned tokens, requiring manual inspection and analysis of the effects of individual tokens. However, this technique requires a specialized, modified architecture and end-to-end training pipelines, and using continuous embeddings is less convenient for end-users and application developers than text prompts.

Emotion2vec also provides a fully unsupervised model for learning continuous representations of utterance-level style and emotion features [5], without being coupled to an end-to-end speech synthesis system. This study builds on emotion2vec to create a discrete emotion representation that could be used to condition text-to-speech systems in addition to classification.

This work aims to bridge this gap by training a discrete representation using vector-quantization that can be converted into textual tokens for convenient fine-tuning and deployment within existing text-to-speech pipelines [6]. The style tokens can be copied and used like textual instructions, without requiring supervised style and emotion labels for training.

2. Methodology

While this technique does not require a labeled dataset, a labeled dataset allows for easy objective evaluation of the result using the labels and provides a convenient source of diverse emotional expression.

using the largest Emotion2vec model [5], an autoencoder with a vector-quantized bottleneck was trained on emotion embeddings extracted from the Expresso dataset [7]. The encoder and decoder each consisted of a layer of dimensions 512 and 128, with a vector quantizer as the bottleneck layer. Different autoencoders with different quantizer configurations were trained and evaluated by training an emotion probe on the discrete representation of clips from the RAVDESS dataset and using the evaluation accuracy of the probe to compare autoencoders [8]. A good autoencoder should preserve enough information to be able to classify the emotion of the source clip with a small codebook, as a large codebook would be difficult for a text-to-speech model to learn and would require a much larger fine-tuning dataset.

After a suitable autoencoder was selected, Emotion2Vec features from Expresso were compressed with the autoencoder. The discrete codes from the autoencoder's quantizers were assigned textual tokens, which were then inserted into text transcripts. The resulting dataset was then used to fine-tune Orpheus [3], Spark-TTS [2], OuteTTS [9], and SpeechT5 [10, 11]. Then, using a standardized fine-tuning pipeline from Unsloth for the first three and Hugging Face's Transformers for the latter [10]. The training also included a small portion of original, unconditioned clips to prevent catastrophic forgetting.

In addition, the effect of introducing contrastive training and contrastive decoding was evaluated on the best performing model. Contrastive training was implemented as an additional loss term rewarding the difference between predictions conditioned on ground-truth embeddings and embeddings from a randomly chosen clip, and contrastive decoding by applying classifier-free-guidance to the logits [12].

The fine-tuned text-to-speech models were evaluated by classifying the emotion of synthesized audio clips and comparing the classification to the target emotion. A fixed set of 100 RAVDESS reference clips was used [8], with each reference clip converted to discrete emotion codes which were used to condition synthesis of a common utterance for each model. The resulting clips were then evaluated by comparing the ground-truth emotion class of each clip's reference clip to the predicted emotion of the generated audio clip using Emotion2Vec's built-in emotion classifier [5].

DOI: 10.54254/2755-2721/2026.37211

3. Results

Smaller quantizers performed better, as large sparse codebooks are more difficult to learn on a small dataset. A configuration of four size-32 codebooks had the highest accuracy, however a more compact configuration of only two codebooks would achieve performance within the margin of error with half the tokens per utterance. The second-best configuration of two size-64 codebooks was chosen as its variance is lower than the similarly-performing configuration of two 32-wide codebooks. Table 1 shows the accuracy of different autoencoder quantizer configurations.

Table 1. Comparison of autoencoders with different quantization configurations

VQ configuration	Mean accuracy
4x32	82.64% ± 1.50
2x64	81.94% ± 0.80
2x32	81.94% ± 1.59
1x64	80.90% ± 2.67
2x128	80.73% ± 2.27
4x2048	72.40% ± 1.88

Table 2 shows the performance of different models. It also includes several baselines for comparison. The random baseline is the expected accuracy of ignoring the conditioning and choosing an emotion at random, while the majority-class baseline represents always generating the most common emotion. White-noise and silence baselines are expected accuracy for generating white noise and silence respectively.

Of all the tested models, Orpheus performed the best and Speech-T5 the worst after fine-tuning [3, 11], however, the accuracy of Orpheus was still below the majority-class baseline. Introducing a contrastive loss increased accuracy to slightly above the baseline, which combined with classifier-free-guidance brought the accuracy to significantly above baseline levels at strength 1.5. Increasing the strength of classifier-free-guidance to 2 decreased the performance to below Speech-T5 [11].

Table 2. Comparison of different models

TTS condition	Emotion accuracy
Orpheus, contrastive, CFG 1.5	31%
Orpheus, contrastive, no CFG	27%
Majority-class	26%
Orpheus	23%
OuteTTS	21%
Spark-TTS	19%
SpeechT5	18%
Orpheus, contrastive, CFG 2.0	18%
Random	14.3%
White-noise	13.78%

DOI: 10.54254/2755-2721/2026.37211

Table 2. (continued)

Silence	13.18%

4. Conclusion

This work has shown that discrete, unsupervised utterance-level embeddings can be used to condition text-to-speech models on speaking style. This allows convenient training of expressive text-to-speech models using only augmented text prompts. Potential applications include introducing additional example-based controls for artistic speech synthesis. Further research could fine-tune language models and speech-to-text models on augmented text prompts and audio clips respectively, to create a STT-LLM-TTS pipeline that avoids some of the loss of non-lexical tonal information that typically occurs in such pipelines. These types of pipelines are more convenient to set up and fine-tune and exhibit more intelligence than end-to-end audio language models, thus this could be used to bridge the gap between the native audio understanding of audio language models and STT-LLM-TTS pipelines.

References

[1] Panayotov, V., Chen, G., Povey, D., & Khudanpur, S. (2015). Librispeech: An ASR corpus based on public domain audio books. *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 5206–5210. https://doi.org/10.1109/ICASSP.2015.7178964

[2] Wang, X., Jiang, M., Ma, Z., Zhang, Z., Liu, S., Li, L., Liang, Z., Zheng, Q., Wang, R., Feng, X., Bian, W., Ye, Z., Cheng, S., Yuan, R., Zhao, Z., Zhu, X., Pan, J., Xue, L., Chen, Y., Li, Z., Xie, C., Guo, Y., & Xue, W. (2025). Spark-TTS: An efficient LLM-based text-to-speech model with single-stream decoupled speech tokens. arXiv. https://doi.org/10.48550/arXiv.2503.01710

[3] Canopy AI. (2025). Orpheus TTS: Towards human-sounding speech. *GitHub*. https://github.com/canopyai/Orpheus-TTS

[4] Wang, Y., Stanton, D., Zhang, Y., Skerry-Ryan, R. J., Battenberg, E., Shor, J., Xiao, Y., Jia, Y., Ren, F., & Saurous, R. A. (2018). Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis. *Proceedings of the 35th International Conference on Machine Learning, 80*, 5180–5189.

[5] Ma, Z., Zheng, Z., Ye, J., Li, J., Gao, Z., Zhang, S., & Chen, X. (2024). Emotion2vec: Self-supervised pre-training for speech emotion representation. *Findings of the Association for Computational Linguistics: ACL 2024*.

[6] van den Oord, A., Vinyals, O., & Kavukcuoglu, K. (2017). Neural discrete representation learning. *Advances in Neural Information Processing Systems, 30*, 6306–6315.

[7] Nguyen, T. A., Hsu, W.-N., D'Avirro, A., Shi, B., Gat, I., Fazel-Zarani, M., Remez, T., Copet, J., Synnaeve, G., Hassid, M., Kreuk, F., Adi, Y., & Dupoux, E. (2023). Expresso: A benchmark and analysis of discrete expressive speech resynthesis. *Proceedings of Interspeech 2023*, 4823-4827. https://doi.org/10.21437/Interspeech.2023-1905

[8] Livingstone, S. R., & Russo, F. A. (2018). The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English. *PLOS ONE, 13*(5), e0196391. https://doi.org/10.1371/journal.pone.0196391

[9] OuteAI. (n.d.). OuteTTS: Interface for OuteTTS models. *GitHub*. https://github.com/edwko/OuteTTS

[10] Han, D., Han, M., & Unsloth team. (2023). Unsloth. *GitHub*. https://github.com/unslothai/unsloth

[11] Ao, J., Wang, R., Zhou, L., Wang, C., Ren, S., Wu, Y., Liu, S., Ko, T., Li, Q., Zhang, Y., Wei, Z., Qian, Y., Li, J., & Wei, F. (2022). SpeechT5: Unified-modal encoder-decoder pre-training for spoken language processing. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, 5723-5738. https://doi.org/10.18653/v1/2022.acl-long.393

[12] Ho, J., & Salimans, T. (2022). Classifier-free diffusion guidance. arXiv. https://doi.org/10.48550/arXiv.2207.12598