ProsodyFlow: High-fidelity Text-to-Speech through Conditional Flow Matching and Prosody Modeling with Large Speech Language Models

Haoyu Wang, Sizhe Shan, Yinlin Guo, Yuehai Wang*
College of Information Science
Electronic Engineering,
Zhejiang University, Hangzhou, China
22331169@zju.edu.cn, 22360174@zju.edu.cn, 22231138@zju.edu.cn, wyuehai@zju.edu.cn

Abstract

Text-to-speech (TTS) has seen significant advancements in high-quality, expressive speech synthesis. However, achieving diverse and natural prosody in synthesized speech remains challenging. In this paper, we propose ProsodyFlow, an end-to-end TTS model that integrates large self-supervised speech models and conditional flow matching to model prosodic features effectively. Our approach involves using a speech LLM to extract acoustic features, mapping these features into a prosody latent space, and then employing conditional flow matching to generate prosodic vectors conditioned on the input text. Experiments on the LJSpeech dataset show that ProsodyFlow improves synthesis quality and efficiency compared to existing models, achieving more prosodic and expressive speech synthesizing. [1]

1 Introduction

Text-to-Speech (TTS) aims to synthesize high-quality, natural-sounding speech from input text. Recent advancements in TTS have led to the development of non-autoregressive models capable of generating high-quality speech (Kim et al., 2021; Ren et al., 2020). However, as TTS models are applied to more complex scenarios, generating speech that captures natural and diverse prosodic attributes remains a significant challenge. Although various strategies have been proposed, such as explicit pitch and energy prediction (Valle et al., 2020; Ren et al., 2020), variational inference methods (Lee et al., 2020), and using reference prosody encoder (Oh et al., 2024; Li et al., 2024; Ren et al., 2022). They share common issues: difficulty in fully extracting rich prosodic information and a tendency for models to learn the average distribution to generate speech without diversity. Recent advances in self-supervised speech language models, such as wav2vec 2.0 (Baevski et al., 2020), HuBERT (Hsu et al., 2021) and WavLM (Chen et al., 2021), have demonstrated substantial improvements in various aspects of speech processing, including understanding speech content, capturing semantic information and extracting prosodic features from speech. These models leverage large-scale pre-training on diverse and unlabeled speech data, enabling them to learn robust and comprehensive speech representations. Such representations are highly effective in prosody modeling, as they capture both local and global variations in speech, such as pitch, rhythm, and intonation, which are key components of prosody representations. Furthermore, flow matching-based TTS models have emerged as a promising approach to achieving both high-quality and fast-speed synthesis. Unlike traditional diffusion methods (Popov et al., 2021; Huang et al., 2022) that rely heavily on complex probabilistic frameworks, flow matching-based models simplify the training process by learning to match distributions more directly (Lipman et al., 2022), leading to significant improvements in training stability. These models (Le et al., 2023; Guo et al., 2023; Mehta et al., 2024) achieve higher efficiency and reduce overall training costs without compromising the quality of the generated speech.

Therefore, to address these challenges in prosody modeling, we propose **ProsodyFlow**, an end-to-end TTS model that combines self-supervised pre-trained models and conditional flow matching. Our work contributes in two main ways:

1. We leverage the self-supervised WavLM model to extract acoustic features and map them into the prosody latent space.

2. We use conditional flow matching to learn the distribution of prosody and sample prosody vectors conditioned on texts.

*Corresponding author.

[1]The audio demos are available at https://szczesnys.github.io/prosodyflow/

Proceedings of the 31st International Conference on Computational Linguistics, pages 7748–7753
January 19–24, 2025.

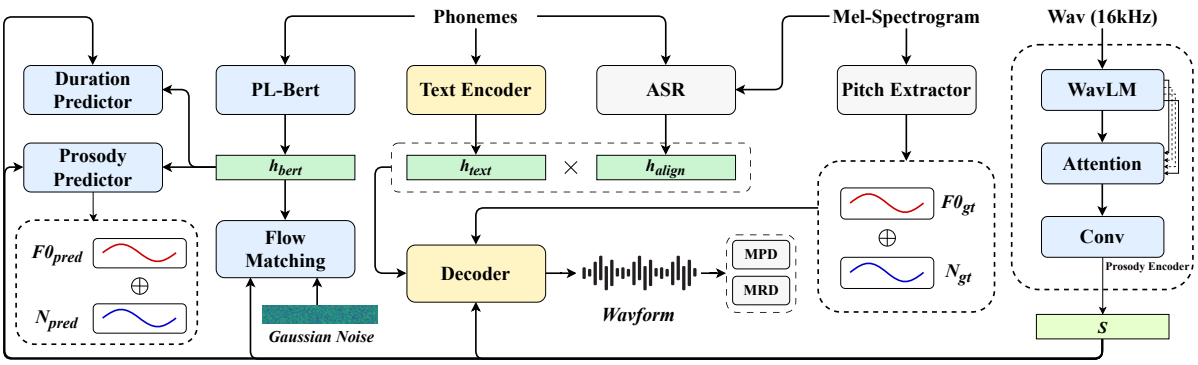

Figure 1: Training diagram of the ProsodyFlow model. The pitch extractor and ASR modules are pre-trained with frozen parameters. The prosody encoder utilizes the outputs from each layer of WavLM.

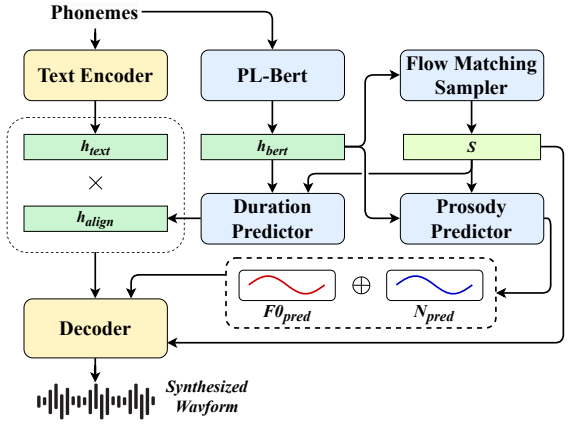

Figure 2: Inference diagram of the ProsodyFlow model.

Our experimental results demonstrate that on the single-speaker LJSpeech dataset (Ito and Johnson, 2017), ProsodyFlow achieves human-level TTS, with a MOS score of 4.23 (±0.08). Additionally, ProsodyFlow exhibits significantly faster synthesis speed compared to autoregressive and diffusion-based TTS models.

2 Methods

ProsodyFlow is a non-autoregressive, end-to-end TTS architecture that leverages a pre-trained WavLM model to extract prosody and style information s from the recording. The prosody s is integrated into the decoder, duration, and pitch predictor by Adaptive Instance Normalization (AdaIN)(Huang and Belongie, 2017). Furthermore, conditioned flow matching generates the predicted prosody vector s'. This approach enables the synthesis of high-quality speech with diverse prosodic styles. The training and inference structures of ProsodyFlow are illustrated in Figure 1 and Figure 2.

2.1 Overview

ProsodyFlow improves upon the StyleTTS2 framework which supports end-to-end training, however, to ensure stable training and accelerate the process, the training is divided into two stages followed by StyleTTS2.

In the first stage, the encoder-decoder structure of the model is trained through the loss function given by $\mathcal{L}_{first} = \mathcal{L}_{\text{mel}} + \mathcal{L}_{\text{GAN}}$. Let t denote the text inputs, x the mel-spectrograms, and w the waveforms. The text encoder processes the phonemes into hidden representations h_{text}. Simultaneously, a pre-trained ASR model is used to obtain the ground-truth alignment $\text{align} = \text{ASR}(t, x)$ and the aligned phoneme encoding $h_{\text{align}} = \text{align} \cdot h_{\text{text}}$. Concurrently, the WavLM prosody encoder extracts the prosody vector s from the waveform. A pre-trained pitch extractor retrieves the ground truth F_0 (pitch) and energy N from the mel-spectrogram. The improved decoder then generates the waveform as $\text{Decoder}(s, F_0, N, h_{\text{align}})$. Multi-Period Discriminator (MPD) and Multi-Resolution Discriminator (MRD) (Lee et al., 2022) are employed as discriminators to enhance the quality of the synthesized speech.

In the second stage, we jointly train all modules through the loss function given by $\mathcal{L}_{jointly} = \mathcal{L}_{\text{mel}} + \mathcal{L}_{\text{dur}} + \mathcal{L}_{F_0} + \mathcal{L}_{N} + \mathcal{L}_{\text{CFM}} + \mathcal{L}_{\text{GAN}}$. we leverage the pre-trained language model PLBert (Li et al., 2023) to extract rich semantic information from the text, which allows us to decouple the Text Encoder and Predictor, following the approach in StyleTTS2. We denote the output of PLBert as $h_{\text{Bert}} = \text{PLBert}(t)$. Both h_{Bert} and s are used as inputs to train the predictors. The

predictors generate the predicted duration, pitch, and energy as d', F_0', and N' respectively, where $d', F_0', N' = \text{Predictor}(h_{\text{Bert}}, s)$. The predicted aligned text embedding is computed as $h_{\text{pred}} = h_{\text{text}} \cdot d'$, and the synthesized speech is then produced as waveform $= \text{Decoder}(s, F_0', N', h_{\text{pred}})$. Additionally, conditional flow matching is employed to learn ordinary differential equations (ODEs) that flow between a noise distribution and the target distribution in latent prosody space, to predict the prosody vector s'.

This two-stage process stabilizes training and improves the ability to capture prosody.

2.2 WavLM Prosody Encoder

WavLM is a self-supervised speech model that learns rich representations from large-scale unlabelled data. WaveLM uses a 12-layer Transformer (Vaswani, 2017) architecture. We take the output from each layer and average it along the sequence length dimension. These averaged outputs are then processed through a self-attention module, resulting in a feature vector for the input speech. Subsequently, a convolutional mapping layer applies a series of downsampling convolutional blocks to transform these attended features into a fixed-size prosody vector space. This mechanism enhances the representation by capturing complex patterns and relationships within the semantic information. This process effectively extracts and condenses the most relevant information, resulting in a compact prosody representation that can be used for expressive speech synthesis.

2.3 Conditional Flow Matching

Conditional Flow Matching (CFM) extends the flow matching framework by incorporating conditioning information (h_{bert}) into the generative process. Instead of learning a flow that transforms a base distribution to a target distribution unconditionally, CFM learns a conditional vector field that effectively maps input conditions and target data characteristics. Given a conditional vector field $\mathbf{v}(\mathbf{x}, t \mid \mathbf{c})$, where $\mathbf{c}$ is the condition (e.g., text or phonetic input), the flow can be expressed as an ODE:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{v}(\mathbf{x}(t), t \mid \mathbf{c}), \tag{1}$$

where $\mathbf{x}(t)$ represents the sample state at time t, and $\mathbf{v}$ is learned to minimize the transport cost between distributions.

To train the conditional flow model, we define a loss function that ensures the learned vector field $\mathbf{v}(\mathbf{x}, t \mid \mathbf{c})$ approximates the true conditional vector field $\mathbf{u}(\mathbf{x} \mid \mathbf{c})$ along the probability path. The loss function for Conditional Flow Matching (CFM) is given by:

$$L_{\text{CFM}}(\theta) = \mathbb{E}_{t, q(\mathbf{x}_1), p_t(\mathbf{x}|\mathbf{x}_1)} \left\| \mathbf{u}(\mathbf{x} \mid \mathbf{x}_1) - \mathbf{v}(\mathbf{x}; \theta) \right\|^2, \tag{2}$$

where $t \sim U[0, 1]$ is uniformly sampled from the interval [0, 1], $q(\mathbf{x}_1)$ is the data distribution, and $p_t(\mathbf{x} \mid \mathbf{x}_1)$ is the conditional probability density function at time t. Here, $\mathbf{v}(\mathbf{x}; \theta)$ is a neural network parameterized by θ. This loss replaces the intractable marginal probability densities and vector fields with conditional probability densities and conditional vector fields, making the learning process more tractable. Importantly, the gradients of $L_{\text{CFM}}(\theta)$ with respect to θ are identical to those of the original Flow Matching loss $L_{\text{FM}}(\theta)$. Definitions mainly follow Lipman et al. (2022).

3 Experiments and Results

3.1 Experimental Settings

We trained a single-speaker model on the LJSpeech dataset, containing approximately 13,100 audio clips (24 hours). The dataset was split into training (12,500), validation (100), and testing (500) sets. Texts were converted to phonemes using Phonemizer (Bernard and Titeux, 2021). We used the improved iSTFTNet (Kaneko et al., 2022) as the decoder to generate waveforms directly. The model was first trained for 150 epochs, followed by joint training of all modules for another 100 epochs. we chose the WavLM-Base-plus version, which is pretrained on 94,000 hours of unlabelled speech data, and the parameters of WavLM are fixed throughout the entire training process. Both stages utilized the AdamW optimizer (Loshchilov, 2017) with $\beta_1 = 0$, $\beta_2 = 0.99$, weight decay $\lambda = 10^{-4}$, learning rate $\gamma = 5 \times 10^{-5}$, and batch size of 8. For training the CFM, we applied a first-order Euler method to solve ordinary differential equations (ODEs). The number of function evaluations (NFE) was randomly sampled from 5 to 10 during training for computational efficiency and fixed at 8 during inference for higher quality.

3.2 Results

We conducted a subjective evaluation: mean opinion score (MOS) to measure human perception of

Model	MOS(CI)	MCD↓	WER↓	RTF↓
Ground-truth	4.25(±0.10)	—	1.27%	—
FastSpeech 2	3.83(±0.09)	5.77	5.47%	**0.0143**
VITS	3.92(±0.09)	5.49	3.61%	0.0376
Diffprosody	4.05(±0.10)	5.27	2.25%	0.0543
GradTTS (n=8)	3.97(±0.09)	5.41	2.13%	0.0532
StyleTTS 2 (n=8)	4.18(±0.09)	4.93	1.71%	0.0231
Proposed (n=8)	**4.23(±0.08)**	**4.63**	**1.33%**	0.0191

Table 1: **Metrics comparing with other models.** We measure the performance with MOS(↑) with 95% confidence intervals, MCD(↓) , WER(↓), and RTF(↓). The Mel-spectrograms are converted to waveforms using iSTFTnet. Diffusion and flow steps are set to 8.

speech quality. We randomly selected 50 samples from the test set. 20 professional raters were employed to rate these samples on a scale from 1 to 5. In addition, we used three objective evaluations to assess speech quality: the Mel-Cepstral Distortion (MCD)[2] calculated through Dynamic Time Warping (DTW), the Word Error Rate (WER) computed using the ASR system Whisper Medium (Radford et al., 2023), and the Real-Time Factor (RTF). All metrics were computed on randomly selected samples. Specifically, it achieves the highest MOS of 4.23 and the lowest MCD of 4.63 among baseline models, indicating higher speech quality. Additionally, the lowest WER of 1.33% indicates the intelligibility of the generated speech, and the RTF of 0.0191 reflects the efficiency of ProsodyFlow. Overall, the results outlined in Table 1 demonstrate that our method significantly improves the speed while synthesizing speech with rich prosodic features.

3.3 Prosody Flow Matching

We conducted comparative experiments to evaluate the performance of the proposed method with different NFEs and all other model configurations are identical. Table 2 details the results. The experiments show even with $n = 1$, the proposed model can achieve results that are competitive with baseline models, which demonstrates the high efficiency of flow matching. Therefore, we chose $n = 8$ as a balance between speed and quality.

3.4 Ablation Study

We conducted ablation studies to validate the effectiveness of the proposed method, with results converted into Comparative Mean Opinion Scores (CMOS) to assess differences in speech quality, as shown in Table 3. Replacing prosody flow matching with a reference encoder as in Ren et al. (2022) resulted in a CMOS of -0.27, emphasizing the importance of flow matching for prosody diversity. Removing WavLM led to a CMOS of -0.18, demonstrating the capability of speech LLMs in capturing prosody. Substituting flow matching with diffusion caused only a slight CMOS change, suggesting that flow matching performs comparably to diffusion in prosody modeling but with greater efficiency.

[2] https://github.com/SamuelBroughton/Mel-Cepstral-Distortion

Model	MOS(CI)	MCD↓	WER↓	RTF↓
proposed-1	3.92(±0.09)	5.11	2.61%	0.0114
proposed-4	4.17(±0.09)	4.83	1.97%	0.0154
proposed-8	4.23(±0.08)	4.63	1.33%	0.0191
proposed-16	4.23(±0.10)	4.58	1.29%	0.0456

Table 2: **Metrics comparing with different NFEs.** We denote different NFE conditions as proposed-n.

Model	CMOS
w/o prosody flow	-0.27
w/o wavlm	-0.18
w/o flow w/ diffusion	-0.04

Table 3: Ablation study of the proposed method.

4 Conclusion

In this paper, we introduce ProsodyFlow, an end-to-end TTS model that combines self-supervised speech models and conditional flow matching to model prosodic features effectively. ProsodyFlow achieves expressive prosodic speech synthesis while reducing computational costs. Ablation studies highlight the importance of flow matching and WavLM in achieving these results. ProsodyFlow addresses the challenge of diverse and natural prosody in TTS, and we believe that this approach shows promising potential for prosodic speech synthesis.

5 Limitation

ProsodyFlow demonstrates excellent performance on single-speaker datasets, but it has yet to be validated in multi-speaker scenarios. Additionally, the model's architecture is relatively complex. Future work will focus on extending its application to multi-speaker settings and simplifying the model to enhance its efficiency and usability.

References

Alexei Baevski, Henry Zhou, Abdel rahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *ArXiv*, abs/2006.11477.

Mathieu Bernard and Hadrien Titeux. 2021. Phonemizer: Text to phones transcription for multiple languages in python. *Journal of Open Source Software*, 6(68):3958.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Micheal Zeng, and Furu Wei. 2021. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16:1505–1518.

Yiwei Guo, Chenpeng Du, Ziyang Ma, Xie Chen, and K. Yu. 2023. Voiceflow: Efficient text-to-speech with rectified flow matching. *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 11121–11125.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM transactions on audio, speech, and language processing*, 29:3451–3460.

Rongjie Huang, Max WY Lam, Jun Wang, Dan Su, Dong Yu, Yi Ren, and Zhou Zhao. 2022. Fastdiff: A fast conditional diffusion model for high-quality speech synthesis. *arXiv preprint arXiv:2204.09934*.

Xun Huang and Serge Belongie. 2017. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision*, pages 1501–1510.

Keith Ito and Linda Johnson. 2017. The lj speech dataset.

Takuhiro Kaneko, Kou Tanaka, Hirokazu Kameoka, and Shogo Seki. 2022. istftnet: Fast and lightweight mel-spectrogram vocoder incorporating inverse short-time fourier transform. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6207–6211. IEEE.

Jaehyeon Kim, Jungil Kong, and Juhee Son. 2021. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In *International Conference on Machine Learning*, pages 5530–5540. PMLR.

Matt Le, Apoorv Vyas, Bowen Shi, Brian Karrer, Leda Sari, Rashel Moritz, Mary Williamson, Vimal Manohar, Yossi Adi, Jay Mahadeokar, and Wei-Ning Hsu. 2023. Voicebox: Text-guided multilingual universal speech generation at scale. *ArXiv*, abs/2306.15687.

Sang-gil Lee, Wei Ping, Boris Ginsburg, Bryan Catanzaro, and Sungroh Yoon. 2022. Bigvgan: A universal neural vocoder with large-scale training. *arXiv preprint arXiv:2206.04658*.

Yoonhyung Lee, Joongbo Shin, and Kyomin Jung. 2020. Bidirectional variational inference for non-autoregressive text-to-speech. In *International conference on learning representations*.

Yinghao Aaron Li, Cong Han, Xilin Jiang, and Nima Mesgarani. 2023. Phoneme-level bert for enhanced prosody of text-to-speech with grapheme predictions. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Yinghao Aaron Li, Cong Han, Vinay Raghavan, Gavin Mischler, and Nima Mesgarani. 2024. Styletts 2: Towards human-level text-to-speech through style diffusion and adversarial training with large speech language models. *Advances in Neural Information Processing Systems*, 36.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. 2022. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*.

I Loshchilov. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Shivam Mehta, Ruibo Tu, Jonas Beskow, Éva Székely, and Gustav Eje Henter. 2024. Matcha-tts: A fast tts architecture with conditional flow matching. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 11341–11345. IEEE.

Hyung-Seok Oh, Sang-Hoon Lee, and Seong-Whan Lee. 2024. Diffprosody: Diffusion-based latent prosody generation for expressive speech synthesis with prosody conditional adversarial training. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Vadim Popov, Ivan Vovk, Vladimir Gogoryan, Tasnima Sadekova, and Mikhail Kudinov. 2021. Grad-tts: A diffusion probabilistic model for text-to-speech. In *International Conference on Machine Learning*, pages 8599–8608. PMLR.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR.

Yi Ren, Chenxu Hu, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. 2020. Fastspeech 2: Fast and high-quality end-to-end text to speech. *arXiv preprint arXiv:2006.04558*.

Yi Ren, Ming Lei, Zhiying Huang, Shiliang Zhang, Qian Chen, Zhijie Yan, and Zhou Zhao. 2022.

Prosospeech: Enhancing prosody with quantized vector pre-training in text-to-speech. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7577–7581. IEEE.

Rafael Valle, Jason Li, Ryan Prenger, and Bryan Catanzaro. 2020. Mellotron: Multispeaker expressive voice synthesis by conditioning on rhythm, pitch and global style tokens. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6189–6193. IEEE.

A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.