Voice Puppetry with FastPitch

Voice Puppetry with FastPitch

Emelie Van De Vreken[1], *Korin Richmond*[2], *Catherine Lai*[2]

[1]Institute for Language, Cognition and Computation (ILCC), University of Edinburgh
[2]Centre for Speech Technology Research (CSTR), University of Edinburgh
{emelie.vandevreken, korin.richmond, c.lai}@ed.ac.uk

Abstract

Affective speech synthesis is an active research area, but recent approaches usually lack the full, fine-grained controllability to produce utterances with any exact affect intended by the user. We propose a puppetry tool based on FastPitch to help model output convey any required suprasegmental meanings. Users can choose any trained FastPitch model, and which features should be mimicked, making the approach fine-grained and language-independent.

Index Terms: speech synthesis, voice puppetry

1. Introduction

In recent years, TTS research has increased its focus on models producing more varied prosody, to convey a wider array of affective states. Despite ongoing efforts, model output does not always sound exactly *how* we want it to, even if perfectly intelligible.

There are currently two related but separate uses of the phrase 'voice puppetry'. It can denote the use of voice to direct synthesised facial expressions [1, 2] or more appropriate here, the use of voice to direct synthesised speech, as in [3, 4].

In this case the trained model should mimic *how* a user, i.e. the puppeteer, pronounces the exact same utterance. We want to keep same vocal characteristics of the training speaker, while changing the paralinguistic information, as conveyed by pitch, energy, durations, and voice quality.

1.1. Use cases

Allowing puppetry at inference time might appear to defeat the purpose of speech synthesis. A trained speech synthesis model allows the creation of audio material without needing to have a voice talent present. The model output may be suboptimal for its context, in terms of its suprasegmental features, conveying paralinguistic information.

The puppetry approach provides user-friendly, fine-grained control to refine synthesised output for practical applications. Puppetry can also be used to create stimuli for experiments on speech synthesis evaluation.

2. Adapting FastPitch

The speech synthesis model used for puppetry is FastPitch (version 1.1) [5]. FastPitch has individual predictors of pitch, energy, and duration per input symbol. At inference time, these predictions can be replaced (see Figure 1).

By default, FastPitch uses mixed grapheme and phoneme input, henceforth referred to as 'input symbols'.

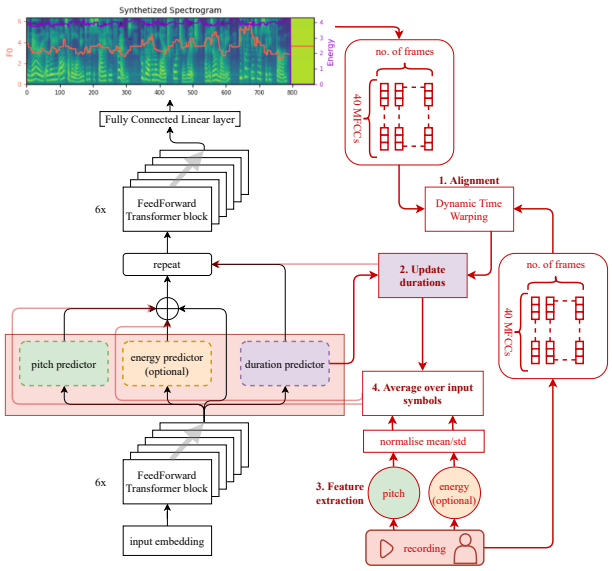

Figure 1: *Diagram of the inference process in FastPitch with puppetry. On the left (black), is the original setup. On the right (red), are the steps of puppetry inference.* $\oplus$ *represents the addition of the encoder output with pitch and/or energy embeddings.*

2.1. Alignment

The first step is to align a new recording to the model output. This is done by calculating the Dynamic Time Warping (DTW) path between the MFCC vectors. For a 22,050Hz sampling rate, 40 MFCCs are calculated on 1,024 sample sliding window frames with 256 samples overlap ($\frac{1}{4}^{th}$ the window size). Based on the DTW path the durations are updated.

2.2. Calculating the durations

Durations are a vector with, for each input symbol, the number of frames it takes up, i.e. the alignment between text and audio. This alignment is unknown for the reference recording.

New durations are calculated by translating the start and end of the model's durations into frame indices. These are in turn translated into frame indices of the reference recording, to divide up all the frames into new input symbol durations.

Per input symbol, the maximum allowed duration is reduced from the default 75 to 20 frames, to avoid distortion. 20 frames is roughly the maximum produced by a model trained on the LJ dataset. High durations tend to be caused by suboptimal DTW alignments, which in turn is sensitive to excessive silence.

2.3. Reference feature extraction

Pitch is zero-mean normalised, as in FastPitch training. Energy is updated to match the training set mean and standard deviation. This is especially important if recordings are much quieter than the training set.

In order to get embeddings that can be added to the encoder output, only one pitch and one energy value is expected per input symbol. Using the updated durations, pitch and energy values of frames are averaged per symbol.

3. Trained models

The FastPitch model was trained using the default parameters and the LJ dataset, for 1,000 epochs on 4 GeForce RTX 2080 Ti GPUs. Additionally, FastPitch was trained for 1,500 epochs on LJ combined with the MELD dataset, which contains scenes from the Friends TV show [6], in order to sound less monotone. The rest of the setup remained identical.

4. Inference

Inference has been tested using the LJ test set, and simple SVO sentences in English, with emphasis varying between the subjects, verbs, and objects (taken from [7]). The puppeteer speaker is a non-native female, with a self-reported British-American accent. An example from this inference setup is provided in Figure 2. More examples and code are available at https://evdv.github.io/2022/06/30/voice_puppetry.html .

Note that puppetry can be done with any recorded utterance with the same lexical content as passed as input to the FastPitch model. Secondly, there is free choice of which features to include in the puppetry: pitch, energy, and/or durations.

The system produces two audio files: the original synthesised utterance, and the puppetry version.

5. Future work

There is currently a degradation in signal post-puppetry, which we theorise to originate from the model a) being trained on relatively monotonous data, and b) training in an end-to-end fashion. The predictors might predict incorrect values, which the decoder learns to compensate for. One solution to this problem is to pre-train the predictors separately.

This approach also needs to be evaluated with listening tests to ensure that the identity of the training speaker has been preserved, and the intent of the puppeteer has been conveyed.

6. Conclusion

Voice puppetry is a useful tool to adjust the suprasegmental features of synthesised speech. By adapting FastPitch we can encourage it to mimic pitch, energy, and duration features from a reference recording, at inference time. The quality of the puppetry depends on the quality of the reference recording, however further evaluation is required to understand the impact of the choice of puppeteer speaker.

7. Acknowledgements

This work was supported in part by the UKRI Centre for Doctoral Training in Natural Language Processing, funded by the UKRI (grant EP/S022481/1) and the University of Edinburgh, School of Informatics and School of Philosophy, Psychology &

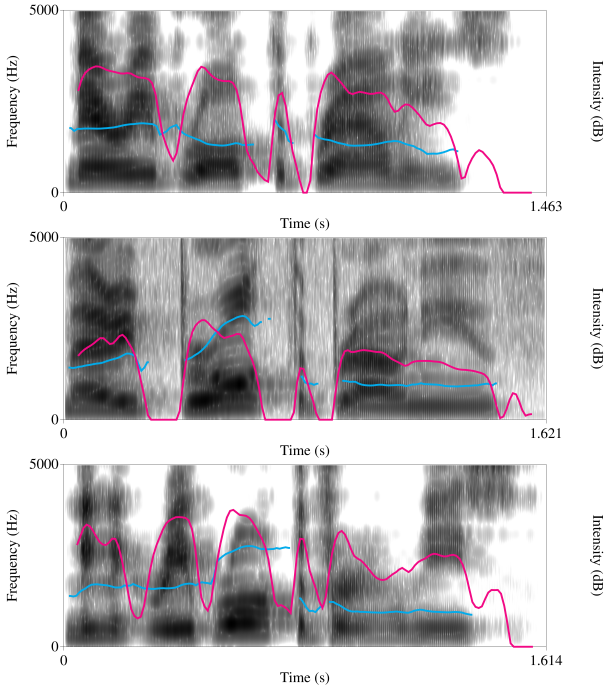

Figure 2: *Spectrograms of FastPitch output (top), original puppeteer speech (middle) and FastPitch output with puppetry (bottom). The input sentence is 'John brought the cookies'. Pitch (in Hz) is shown in blue, energy (in dB) in magenta.*

8. References

[1] M. Brand, "Voice puppetry," in *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, 1999, pp. 21–28.

[2] J. Thies, M. Elgharib, A. Tewari, C. Theobalt, and M. Nießner, "Neural voice puppetry: Audio-driven facial reenactment," in *European conference on computer vision*. Springer, 2020, pp. 716–731.

[3] M. P. Aylett, D. A. Braude, C. J. Pidcock, and B. Potard, "Voice puppetry: Exploring dramatic performance to develop speech synthesis," in *Proc. 10th ISCA Speech Synthesis Workshop*, 2019, pp. 117–120.

[4] M. P. Aylett and Y. Vazquez-Alvarez, "Voice puppetry: Speech synthesis adventures in human centred ai," in *Proceedings of the 25th International Conference on Intelligent User Interfaces Companion*, ser. IUI '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 108–109. [Online]. Available: https://doi.org/10.1145/3379336.3381478

[5] A. Łańcucki, "Fastpitch: Parallel text-to-speech with pitch prediction," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6588–6592.

[6] S. Poria, D. Hazarika, N. Majumder, G. Naik, E. Cambria, and R. Mihalcea, "Meld: A multimodal multi-party dataset for emotion recognition in conversations," *arXiv preprint arXiv:1810.02508*, 2018.

[7] E. Gutierrez, P. Oplustil-Gallegos, and C. Lai, "Location, Location: Enhancing the Evaluation of Text-to-Speech synthesis using the Rapid Prosody Transcription Paradigm," in *Proc. 11th ISCA Speech Synthesis Workshop (SSW 11)*, 2021, pp. 25–30.