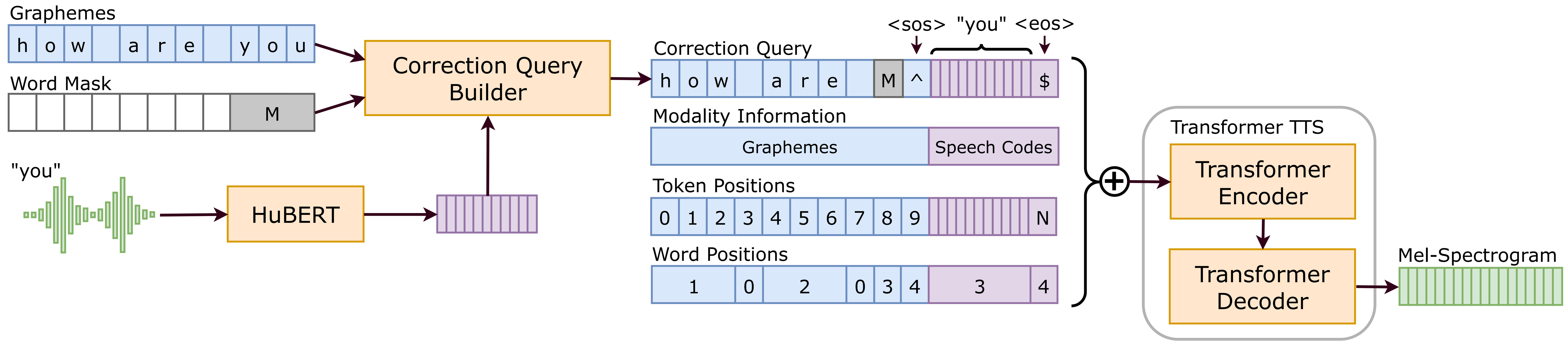
Using speech examples to correct TTS mispronunciations

THE UNIVERSITY of EDINBURGH
CSTR

Jason Fong, Daniel Lyth, Gustav Eje Henter, Hao Tang, Simon King
The Centre for Speech Technology Research, School of Informatics, The University of Edinburgh

Paper & Samples

Architecture | Speech Audio Corrector (SAC)



1 Problem: Phoneme-based pronunciations are expensive

Correct pronunciation is essential for high-quality TTS but is unachievable using only grapheme inputs.

The usual solution involves expensive pronunciation dictionaries & grapheme-to-phoneme models.
→ Therefore, TTS for low-resourced scenarios is not feasible.

Research Question:
Can we control TTS pronunciations using cheaper-to-obtain resources?

2 Solution: Use speech examples to control pronunciation

Speech examples are an alternative source of ground-truth pronunciations.

They are cheap to obtain via crowd-sourcing or extracting from found data using forced alignment.

Our solution:
Train a grapheme-based TTS model that can use speech examples to perform one-off corrections of mispronunciations when needed.

Steps:
1. Extract self-supervised speech codes for all utterances.
2. Align speech codes to word token boundaries.
3. Train a TTS model, swapping the graphemes for each word token with its speech codes with a 50% probability [1].
4. At inference time, use speech codes rather than graphemes to represent words that are mispronounced.

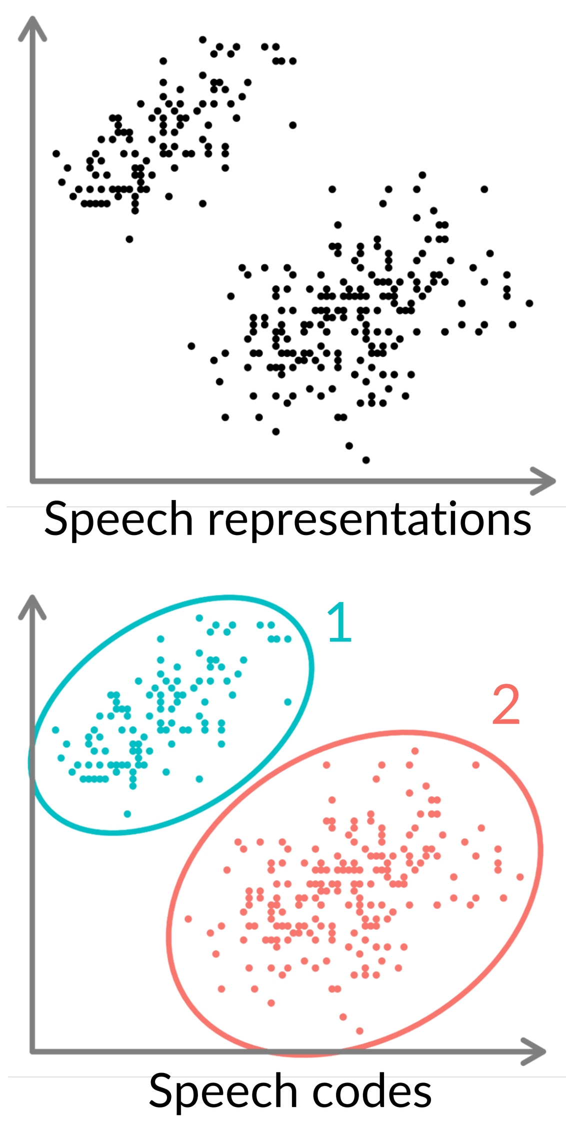
3 Why use "self-supervised speech codes"?

Raw speech contains information often unrelated to pronunciation such as speaker ID and pitch.

Self-supervised models such as wav2vec 2.0 and HuBERT extract representations that better separate different types of speech information.

These representations perform very well in ASR, demonstrating an ability to capture phonetic content [2, 3].

Moreover, they can be discretised into "speech codes" using k-means clustering. This further discards non-phonetic information [4, 5].



4 Experiment: Compare graphemes with speech codes

Data:
LJ Speech (24 hours, single female US speaker)

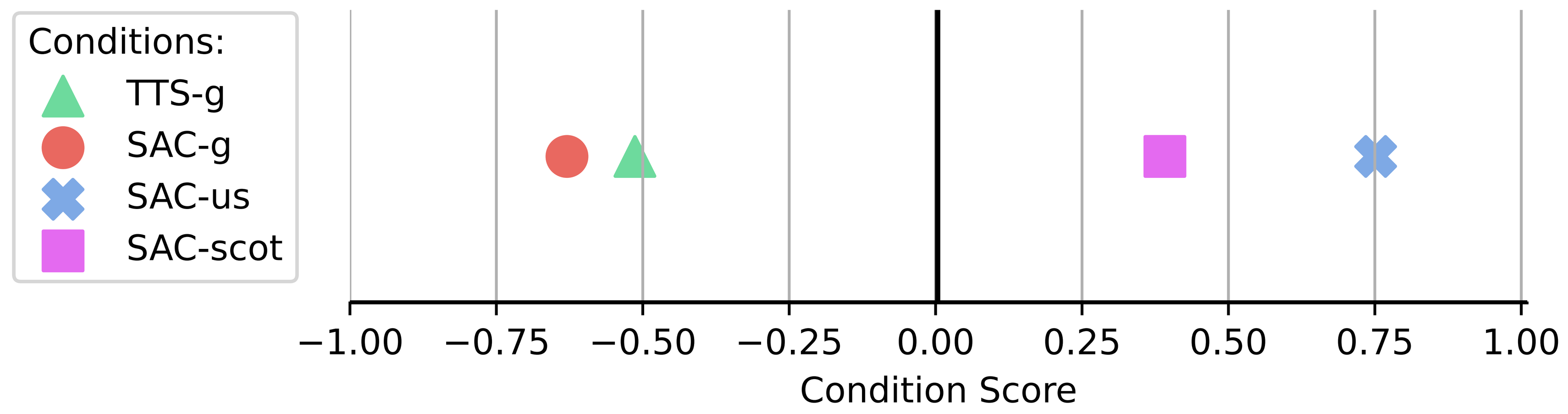
Models:
- Transformer TTS
- HuBERT-Base-LS960h
- Montreal forced aligner

Systems:
- TTS_G: Transformer TTS using grapheme inputs
- SAC_G: SAC using grapheme inputs
- SAC_{US}: SAC using US female speech code inputs
- SAC_{Scot}: SAC using Scottish female speech code inputs

Test set stimuli:
78 held-out words that are mispronounced by SAC_G, contained in the carrier sentence "How is … pronounced?".

5 Results

Subjective AB preference tests:



Other observations:
- TTS_G slightly preferred over SAC_G. Possibly as 7 out of 78 test words are pronounced correctly by TTS_G.
- SAC_{Scot} more likely to mispronounce words than SAC_{US} (24% vs 15% mispronounced). Possibly due to Scottish speech being from a different data distribution, which was unseen during training.
- Using Scottish speech doesn't noticeably affect speaker identity.
- US-based raters preferred US pronunciations over Scottish ones. E.g.: derby, mobile, bother, comedy.

6 Conclusions

Speech examples can control the pronunciation of TTS models. Also works using mismatched accents.

Potential future work:
- Increase robustness to accent mismatch.
- Control syllable stress.
- Control non-segmental aspects such as prosody.
- Use for multilingual code-switching.

References

[1] Kastner, Kyle, et al. "Representation mixing for tts synthesis." *ICASSP 2019*
[2] Baevski, Alexei, et al. "wav2vec 2.0: A framework for self-supervised learning of speech representations." *Advances in neural information processing systems*
[3] Hsu, Wei-Ning, et al. "Hubert: Self-supervised speech representation learning by masked prediction of hidden units." *IEEE/ACM Transactions on Audio, Speech, and Language Processing*
[4] van Niekerk, Benjamin, et al. "A comparison of discrete and soft speech units for improved voice conversion." *ICASSP 2022*
[5] Polyak, Adam, et al. "Speech resynthesis from discrete disentangled self-supervised representations." *arXiv preprint*