

## MY WEIRD PROMPTS

Podcast Transcript

### EPISODE #196

# Beyond the Robot: The Science of Modern Voice Cloning

Published January 08, 2026 • Runtime: 23:28

<https://myweirdprompts.com/episode/voice-cloning-neural-tts/>

## EPISODE SYNOPSIS

In this meta-focused episode of My Weird Prompts, Herman and Corn peel back the digital layers of their own existence to explore the cutting-edge state of text-to-speech technology in early 2026. They move beyond the robotic, "ransom-note" style of early synthesis to discuss the power of neural generative models, explaining how modern systems utilize transformer architectures and attention mechanisms to simulate human-like prosody, rhythm, and emotion. The duo also dives deep into the practicalities of voice cloning—addressing the "average voice" problem that plagues regional accents—and offers a technical breakdown of optimizing AI workflows using serverless GPUs, cached speaker embeddings, and the trade-offs between premium APIs and lightweight open-source models like Kokoro.

## DANIEL'S PROMPT

### Daniel

I've talked about my interest in speech tech before, specifically how tools like Whisper have revolutionized the field. This podcast is actually created using text-to-speech (TTS), with the voices of Herman and Korn generated by Resemble's Chatterbox engine. I've experimented with voice cloning myself and noticed a huge difference between robotic TTS and neural-based TTS, which can infer prosody and sound much more natural. How does neural-based TTS work under the hood, and how do models manage to infer elements like prosody? What advice do you have for someone building AI workflows with TTS to get consistent, high-quality results without excessive costs?



# TRANSCRIPT

## Corn

Welcome back to My Weird Prompts. I am Corn, and today we are getting a little bit meta. It is January eighth, twenty-twenty-six, and if you have been listening to us for a while, you might have noticed that my brother and I sound, well, pretty consistent.

## Herman

Or maybe too consistent, depending on who you ask! I am Herman Poppleberry, and today is a special one because we are actually addressing the very technology that allows us to exist in your ears. Our housemate Daniel sent us a prompt about speech technology, specifically text to speech and voice cloning.

## Corn

It is funny, Daniel lives right down the hall from us here in Jerusalem, but he still sends these audio prompts in like he is a long-distance correspondent. He was talking about how he has been using tools like Whisper for transcription and Resemble for generating our voices, and he is curious about what is actually happening under the hood.

## Herman

It is a great question. We have come such a long way from the robotic, stilted voices of the early two-thousands. If you remember the old navigation systems or the early accessibility tools on computers, they sounded like a toaster trying to read the dictionary. But now, in twenty-twenty-six, we are at a point where, in many cases, the nuance can be very close to human speech.

## Corn

Right, and Daniel mentioned he tried to clone his own voice, specifically his Irish accent, and he struggled with the model defaulting back to an American cadence. So, Herman, let us start there. How did we get from those robotic phoneme-mashers to neural-based systems that can actually infer prosody and emotion?

### Herman

That is the big shift, Corn. The old systems were what we call concatenative synthesis. Basically, they had a massive database of a single voice actor recording every possible sound combination in a language. When the computer wanted to say a word, it would literally stitch those tiny audio clips together. It was technically accurate but emotionally dead because there was no flow between the sounds.

### Corn

Like a ransom note made of magazine clippings, but for your ears.

### Herman

Exactly. There was no co-articulation, which is the way our mouths prepare for the next sound while finishing the current one. Neural text to speech changed everything by moving away from stitching clips to what we call generative modeling. Instead of a database of clips, the model learns a statistical representation of what speech sounds like. Alongside transformer-based approaches, there has been growing research into alternative generative methods like diffusion and flow-matching, and some experimental models in the F-five T-T-S family explore these ideas.

### Corn

So, when we talk about a neural model for speech, are we talking about something similar to how a large language model predicts the next word?

### Herman

Very similar. In fact, many modern systems use a transformer architecture, just like the one behind the major language models we use today. The process is usually split into two main parts. First, you have the acoustic model. This takes the raw text and converts it into a visual representation of sound called a mel-spectrogram.

### Corn

I have seen those. They look like heat maps of audio frequencies, right?

## Herman

Precisely. The mel-spectrogram captures the energy at different frequencies over time. It is essentially a blueprint for the voice. The neural network looks at the text, understands the context, and draws this blueprint. Then, the second part, called a vocoder, takes that blueprint and turns it into actual audio waves that your speakers can play. However, some of the newest end-to-end models, including systems like G-P-T-four-o, move away from the classic mel-spectrogram plus separate vocoder pipeline and instead generate audio through a more integrated representation, closer to predicting waveforms directly from text and conditioning signals.

## Corn

Okay, but the magic Daniel was asking about is the prosody. Prosody is the rhythm, the stress, and the intonation of speech. How does a model look at a sentence like, oh, I do not know, I never said she stole my money, and know which word to emphasize? Because that sentence has seven different meanings depending on where you put the stress.

## Herman

That is where the deep learning aspect becomes so powerful. In twenty-twenty-six, these models are not just looking at the words; they are looking at the semantic embeddings. When the model processes a sentence, it creates a mathematical representation of the meaning. It realizes that if the sentence ends in a question mark, the pitch should likely rise at the end. If it sees an exclamation point, it increases the amplitude and the sharpness of the onset of the words.

## Corn

But it goes deeper than just punctuation, does it not? I mean, we do not always punctuate our speech perfectly.

## Herman

It does. Modern models use something called attention mechanisms. While the model is generating the sound for the third word in a sentence, it is still looking back at the first word and ahead to the tenth word to ensure the emotional arc is consistent. It is essentially simulating a mental model of how a human would breathe and pause. Some of the most advanced models even have latent variables for style. You can tell the model to be whispered, or angry, or excited, and it adjusts the entire generation to match those parameters.

## Corn

This brings me back to Daniel's Irish accent problem. He said he recorded an hour of training data but the model still sounded American. Why does that happen? If the model is supposed to be cloning his voice, why is it leaking another accent?

## Herman

This is a common issue with what we call foundation models. Most of the high-end text to speech engines today are pre-trained on tens of thousands of hours of speech data. Because the vast majority of high-quality, labeled audio data on the internet is American English, the model develops a very strong prior, or a bias, toward those speech patterns. This is often called the average voice problem.

## Corn

So even if you give it an hour of Irish audio, it is trying to fit that audio into a box that was built for a Californian or a New Yorker?

## Herman

Exactly. It is like trying to paint a portrait with a brush that only wants to make certain strokes. The model sees the Irish phonemes, but when it goes to predict the prosody, the rhythm, it defaults to the patterns it learned from that massive twenty-thousand-hour dataset. To fix that, you usually need a process called fine-tuning, where you tell the model to prioritize your specific data over its pre-existing knowledge. But even then, if the foundation model was not trained on a diverse enough set of accents, it might struggle to reproduce the specific lilt of a Dublin accent. In twenty-twenty-six, some newer zero-shot models that explore flow- or diffusion-style training show promising results and can sometimes avoid the strongest American-accent bias, but performance still depends heavily on the underlying training data.

## Corn

That is fascinating. It is almost like the model has its own native language and it is trying to speak Daniel's voice with a heavy American accent. Now, Daniel also mentioned the workflow side of things. He is using Resemble and Modal for serverless graphical processing units. For someone trying to build a high-quality workflow without spending a fortune, what is the state of the art right now?

### Herman

The landscape in twenty-twenty-six is actually quite varied. You have the high-end A-P-I providers like Eleven Labs and Resemble, which are fantastic for ease of use. They handle all the heavy lifting, the hosting, and the complex inference. But as Daniel noted, the costs can add up if you are doing long-form content.

### Corn

Right, because you are paying per character or per minute of audio generated.

### Herman

Exactly. If you want to move toward something more cost-effective, you have to look at self-hosting or using serverless infrastructure like he is doing. There are some incredible open-source models now. We have seen projects like Bark, X-T-T-S, and more recently, the Kokoro models, which are incredibly lightweight. Some of these can run on a standard consumer-grade laptop and still produce near-human quality. Kokoro is particularly impressive because it is a relatively small model—on the order of tens of millions of parameters—yet many listeners find its quality competitive with much larger systems in informal tests.

### Corn

Daniel mentioned something interesting about caching embeddings. He noticed that every time you or I speak, the system uses a numerical representation of our voice clone. Can you explain why caching that is a big deal for performance and cost?

### Herman

That is a great technical catch by Daniel. When you use a zero-shot voice cloning system, the model has to look at a reference audio clip of the target voice, say a ten-second clip of me, and turn that into a speaker embedding. This embedding is a long string of numbers that tells the model exactly what my vocal characteristics are, the rasp, the pitch, the resonance.

### Corn

And if you are generating a whole podcast, doing that calculation for every single sentence is a waste of energy.

### Herman

Precisely. If you are generating a thousand sentences for an episode, and you calculate that embedding a thousand times, you are burning through graphical processing unit cycles for no reason. By calculating the embedding once, caching it, and then just passing that small piece of data along with each new text string, you can significantly reduce the latency and the compute cost. It makes the whole pipeline much snappier. On a platform like Modal, that can noticeably reduce your compute usage and monthly bill, sometimes by a substantial margin, depending on your workload.

### Corn

Let us talk about the quality versus cost trade-off. If I am a developer building a tool that needs to read articles aloud, or maybe an interactive A-I assistant, how do I decide between an expensive A-P-I and a self-hosted open-source model?

### Herman

It really comes down to your requirements for latency and emotional range. The high-end A-P-Is are still the king of prosody. They are very good at inferring that a sentence is a joke or a sarcastic comment. If your application needs deep emotional resonance, the premium A-P-Is are usually worth the cost. However, if you are doing something like reading technical documentation or news summaries where the tone is relatively flat, the open-source models like Kokoro or F-five-T-T-S are more than sufficient.

### Corn

And what about the hardware? Daniel mentioned Modal, which is a serverless platform. Why go that route instead of just renting a dedicated server?

### Herman

Serverless is perfect for audio generation because the work is often bursty. You might need to generate ten minutes of audio all at once, and then nothing for an hour. If you have a dedicated server running twenty-four-seven, you are paying for it even when it is idle. With serverless graphical processing units, you only pay for the seconds the model is actually running. The downside is something called cold starts.

## Corn

I remember we talked about this back in episode two-hundred-six when we were discussing A-I benchmarks. A cold start is the time it takes for the system to load the model into memory before it can start working, right?

## Herman

Exactly. For a large speech model, that might take ten or fifteen seconds. If you are building a real-time voice assistant, a fifteen-second delay is a deal-breaker. But for a podcast like ours, where the generation happens in the background before the listener ever hits play, cold starts do not matter at all. That is why Daniel's setup is so smart for this use case.

## Corn

So, if someone wants to replicate this, they should look at a pipeline that uses a fast transcription model like Whisper for the input, and then a cached embedding with a neural vocoder for the output. But how do we make it sound less like a reading and more like a conversation? I notice that sometimes T-T-S models can sound a bit breathless or they do not know when to take a beat.

## Herman

That is where the pre-processing of the text comes in. A lot of people just dump raw text into these models and hope for the best. But if you want that natural flow, you need to use something called Speech Synthesis Markup Language, or S-S-M-L. It is a bit like H-T-M-L for audio. You can manually insert tags that say, break for five-hundred milliseconds here, or emphasize this specific word.

## Corn

That sounds like a lot of manual work though. Is there a way to automate that?

## Herman

In twenty-twenty-six, we are seeing large language models being used as the director. You can take your raw script and pass it through a model like G-P-T-four or a specialized smaller model, and tell it to add S-S-M-L tags based on the context of the dialogue. It can identify where a natural pause would be for a breath or where a character would be interrupted. It is essentially giving the speech model a set of acting notes.

## Corn

That is such a cool concept. The language model understands the emotion, and the speech model executes the sound. It is a two-stage process of understanding and then performing.

## Herman

It really is a performance. And that brings us to the future of this tech. We are moving toward what is called end-to-end multi-modal models. Instead of having a separate text model, a separate acoustic model, and a separate vocoder, we are starting to see single models that can take in text and output raw audio directly.

## Corn

Like the G-P-T-four-o model we saw glimpses of, where the latency is so low it feels like a real-time conversation.

## Herman

Exactly. When you remove the boundaries between these stages, the model can maintain much more of the nuance. It does not lose information as it passes data from one part of the pipeline to the next. The prosody becomes much more organic because the model is predicting the audio waveforms directly from the meaning of the words.

## Corn

I want to go back to the ethical side of this for a second. Daniel mentioned he was doing this for fun and for the podcast, but voice cloning has a bit of a reputation. What are the safeguards in twenty-twenty-six?

## Herman

It is a major topic of discussion. Many leading commercial providers now advertise consent and verification safeguards. For example, some require you to record specific scripts or confirm you have rights to a voice, and they prohibit cloning public figures, although exact policies and enforcement vary by provider. There are also digital watermarking technologies being integrated into the audio itself. These are subtle changes to the frequencies that a human cannot hear, but a piece of software can detect to prove that the audio was A-I-generated.

### Corn

Like a digital fingerprint hidden in the sound waves.

### Herman

Exactly. We discussed something similar in episode one-hundred-forty-six when we were looking at A-I guardrails. It is a cat-and-mouse game, but the goal is to make sure that as this technology becomes more accessible, it remains clear what is human and what is synthesized.

### Corn

So, to summarize for Daniel and anyone else building these workflows. If you want high quality and low cost, you should look at caching your speaker embeddings, using serverless infrastructure for your heavy lifting, and maybe using a language model to add some performance notes to your script before it goes to the speech engine.

### Herman

And do not be afraid to experiment with the open-source models. The gap between the paid A-P-Is and the top-tier open-source projects is closing faster than anyone expected. If you have the technical chops to set up a container on a platform like Modal, you can get very close to Eleven Labs' quality for a fraction of the price.

### Corn

And hopefully, by this time next year, those models will have a better grasp of the Irish accent so Daniel can finally hear himself properly!

### Herman

One can only hope. It is all about the data, Corn. As more diverse datasets are open-sourced, these models will get better at representing the whole world, not just the folks in Silicon Valley.

**Corn**

Well, this has been a bit of a trip, talking about our own digital guts. It makes me appreciate the work Daniel puts into making us sound as good as we do.

**Herman**

It really does. I feel quite polished today, I must say.

**Corn**

You always do, Herman. You always do. Before we wrap up, I want to remind everyone that if you are enjoying our deep dives into the weird and the technical, we would really appreciate it if you could leave us a review on your podcast app or on Spotify. It genuinely helps us reach more curious minds like yours.

**Herman**

It really does make a difference. And remember, you can find all our past episodes and a way to get in touch with us at our website, [myweirdprompts dot com](http://myweirdprompts.com). We love hearing what you want us to explore next.

**Corn**

Huge thanks to Daniel for sending in this prompt. It was a fun one to tackle from the inside out. This has been My Weird Prompts.

**Herman**

I am Herman Poppleberry, and we will talk to you next week.

**Corn**

Thanks for listening. Stay curious.

**Herman**

That was a good one, Corn. Do you think they realized we were talking about ourselves the whole time?

**Corn**

I think they have a pretty good idea, Herman. I think they have a pretty good idea.

**Herman**

Fair enough. Let us go see what Daniel is cooking for dinner. I smell something good.

**Corn**

Probably just more data for the model, Herman.

**Herman**

Ha! Very funny. See you everyone.