

Alania-2: Turkish Text-to-Speech with Instant and Fine-Tuned Voice Cloning

Sezgin Saygili Emre Kaplaner Oncel Ozgul Fikri San Koktas

PatientDesk AI

sezgin@patientdesk.ai <https://speech.patientdesk.ai>

Abstract

We present Alania-2, a Turkish text-to-speech (TTS) system with two built-in voices, voice design from text descriptions, and two kinds of voice cloning. Alania-2 is a 488M-parameter model with the VoxCPM architecture, which generates continuous audio latents with a diffusion head on top of a language-model backbone. We train it from scratch on 18,195 hours of Turkish speech, 90% of which is synthetic speech produced by a larger teacher model and filtered automatically. A text normalizer rewrites numbers, abbreviations and foreign names into the words a Turkish speaker would say; without it, the model misreads every sentence that contains digits. On Freya-TR-Eval, a public benchmark of 495 Turkish sentences scored with automatic speech recognition, Alania-2 reaches a word error rate of 1.44%, compared with 5.99% for our previous system. On its built-in voices, it comes within half a point of character error rate of a 2.3B-parameter model at under a quarter of the size, and a 7.6M-parameter single-voice model trained with the same recipe matches the most intelligible systems we measured, which suggests that data and text normalization matter more for intelligibility than model size. We also report practical findings on voice cloning: conditioning on the reference audio alone, rather than continuing it, cuts skipped words from 21.9% to 1.7%; 20–30 seconds of reference audio works best; and a low-rank adapter trained in about two minutes raises speaker similarity well above instant cloning. Finally, we describe inference-engine changes that bring first audio down to 32 ms for a single stream. We release our evaluation harness and three Turkish speech and text datasets.

1 Introduction

Turkish is spoken by more than eighty million people, but it remains a mid-resource language for speech synthesis. Recent zero-shot TTS models (Chen et al., 2025a,b; Du et al., 2024; Zhou et al., 2025) were trained mainly on English and Chinese speech. Multilingual successors such as VoxCPM2 (Zhou et al., 2026) support Turkish as one of 30 languages, and open Turkish systems differ by an order of magnitude in intelligibility (Section 6).

Turkish text is also hard to read aloud correctly. Numbers, dates and foreign names take case suffixes that must agree with how they are pronounced, and the dotted and dotless *i* distinguish words but are easily broken by case conversion designed for English.

We built Alania-2 for telephone voice assistants and long-form narration. These uses need reliable reading of numbers, doses, dates and names; low latency when streaming; consistent built-in voices; and voice cloning that customers can use on their own. Our previous system, Alania-1, is a 3.3B-parameter language model over 24 kHz SNAC codec tokens (Siuzdak et al., 2024). It misreads about one word in seventeen on the Freya-TR-Eval benchmark (Pamuk et al., 2026).

Figure 1 gives an overview of Alania-2. Our contributions are:

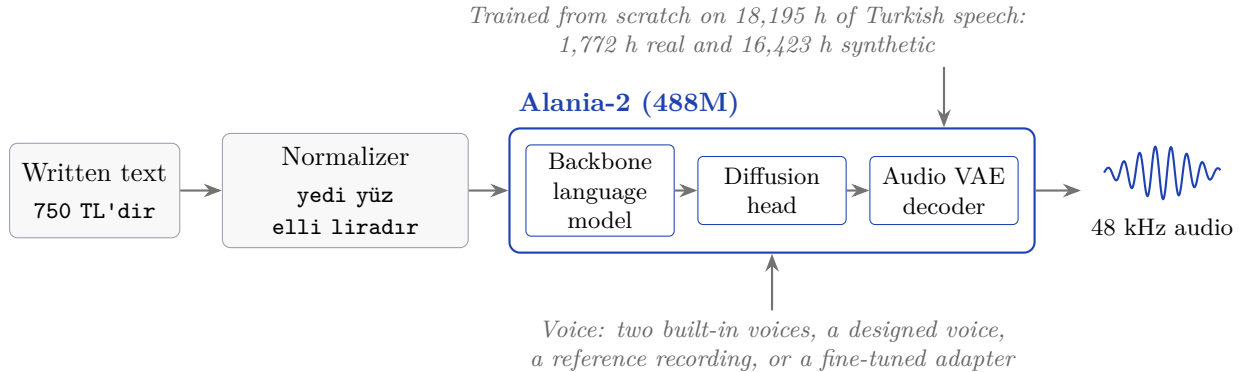


Figure 1: Overview of Alania-2. Text is first rewritten into the words a speaker would say (§4); the model then generates 48 kHz audio in the requested voice (§5, 8).

1. **A data recipe for a mid-resource language.** We train a 488M-parameter model from scratch on 90% synthetic speech and show that, on its built-in voices, it comes within half a point of character error rate of a fine-tuned model more than four times its size (Sections 3 and 5).
2. **A text normalizer** with readings reviewed by a native speaker and respelling of foreign brand names (Section 4).
3. **A comparison with 14 other systems** under one protocol (Section 6).
4. **Evidence that model size matters little for intelligibility.** A 2.3B-parameter model trained on our data is barely more intelligible than Alania-2, and a 7.6M-parameter single-voice model trained with the same recipe matches the best systems we measured (Section 7).
5. **Practical findings on voice cloning:** why continuing a reference skips words, how much reference audio helps, and fast fine-tuning with low-rank adapters (Section 8).
6. **Inference-engine changes** for low latency and for serving many fine-tuned voices from one model (Section 10).

We release the Freya-TR-Eval harness and results, a 2,398-hour Turkish caption set, a 3,411-hour synthetic speech set and a 1.44M-sentence domain text corpus. We do not release the model weights.

2 Related Work

Zero-shot TTS and voice cloning. Codec language models clone a voice by continuing a short audio prompt (Chen et al., 2025a). CosyVoice (Du et al., 2024) combines supervised semantic tokens with flow matching, and F5-TTS (Chen et al., 2025b) removes the need for explicit text–speech alignment. VoxCPM (Zhou et al., 2025) predicts continuous latents of an audio variational autoencoder (VAE) with a diffusion decoder on a language-model backbone, without a discrete speech tokenizer. VoxCPM2 (Zhou et al., 2026) extends it to 30 languages, voice design from text and a VAE that encodes at 16 kHz and decodes at 48 kHz. We use the VoxCPM2 architecture, tokenizer and VAE but train our own weights. Our previous system, Alania-1, followed the codec route with SNAC tokens (Siuzdak et al., 2024).

Turkish TTS. Yeshpanov et al. (2023) train a Tacotron 2 model on Kazakh alone and synthesize nine other Turkic languages, including Turkish, by mapping their letters through the International Phonetic Alphabet to Kazakh letters. FreyaTTS (Pamuk et al., 2026) is a 183M-parameter Turkish-

first flow-matching model released together with the Freya-TR-Eval benchmark, and Ozkose et al. (2026) compare four recent systems on Turkish, fine-tuned and zero-shot, with eighteen objective metrics.

Data and evaluation tools. We build our real-speech set from YODAS (Li et al., 2023), a multilingual corpus of YouTube audio, much of it with manual or automatic subtitles. We filter it with a language classifier trained on VoxLingua107 (Valk and Alumäe, 2021), the AST audio tagger (Gong et al., 2021), DNSMOS quality prediction (Reddy et al., 2021) and Whisper speech recognition (Radford et al., 2023). Our synthetic text comes from FineWeb-2 (Penedo et al., 2025), Wikipedia and sentences written by Qwen3 (Yang et al., 2025). For evaluation, we measure intelligibility with Whisper, naturalness with UTMOS (Saeki et al., 2022), and speaker similarity with ECAPA-TDNN embeddings (Desplanques et al., 2020; Ravanelli et al., 2021); FLEURS (Conneau et al., 2023) supplies some of the cloning references.

Efficient serving. PagedAttention (Kwon et al., 2023) manages the key/value cache of language models in fixed-size blocks, and S-LoRA (Sheng et al., 2024) and Punica (Chen et al., 2024) serve many low-rank adapters (LoRA; Hu et al., 2022) on one base model. Our engine builds on Nano-vLLM (a710128, 2025; GeeeekExplorer, 2025), which uses the same block-based cache.

3 Data

Figure 2 summarizes the training data. The base mix combines 1,772 hours of filtered real speech with 16,423 hours of synthetic speech; a second, shorter stage adds voice design and style instructions.

3.1 Real speech

We start from the Turkish part of YODAS (Li et al., 2023): 38,494 recordings totalling 9,216 hours. Most of this audio is weakly labelled. Of the recordings, 97% carry automatic YouTube captions, and only 2,448 hours are labelled as Turkish at all. We therefore treat every file as untrusted. We split it into segments, check the language with a classifier trained on VoxLingua107 (Valk and Alumäe, 2021) and with Whisper (Radford et al., 2023), remove music and singing with the AST audio tagger (Gong et al., 2021), reject clipped or band-limited audio, and keep segments with a DNSMOS score of at least 2.4 (Reddy et al., 2021). Finally, we transcribe every segment again with Duyu, our Turkish speech recognizer, a fine-tune of Whisper large-v3 turbo.

This leaves 2,398 hours (835,267 segments, median length 9.9s). Qwen3-32B (Yang et al., 2025) writes a short style caption for each segment, describing the speaker and the delivery. For training, we apply stricter filters and remove segments with a low DNSMOS score (189,631), a Duyu transcript that disagrees with the YouTube caption (42,186), or an implausible speaking rate (152). The training set holds 598,213 segments (1,772 hours) from 19,969 speaker clusters; we hold out about 1% of the clusters for validation.

3.2 Synthetic speech

Turkish speech with the quality, speaker balance and text coverage we needed is not available at this scale, so we generate it with VoxCPM2 (Zhou et al., 2026), using 10 diffusion steps and a guidance scale of 2.0. One H100 GPU produces 83 seconds of audio per second.

Table 1: Checks applied to every synthetic clip. A clip is kept only if it passes all of them.

Check	Measured with	Threshold
Intelligibility	CER, faster-whisper large-v3	≤ 0.05
Speaking rate	characters per second	7–25
Voice match	ECAPA similarity to the reference	≥ 0.45 (≥ 0.35 with a style instruction)
Naturalness	UTMOS22	≥ 2.5
Pitch	median F0	consistent with the voice’s gender

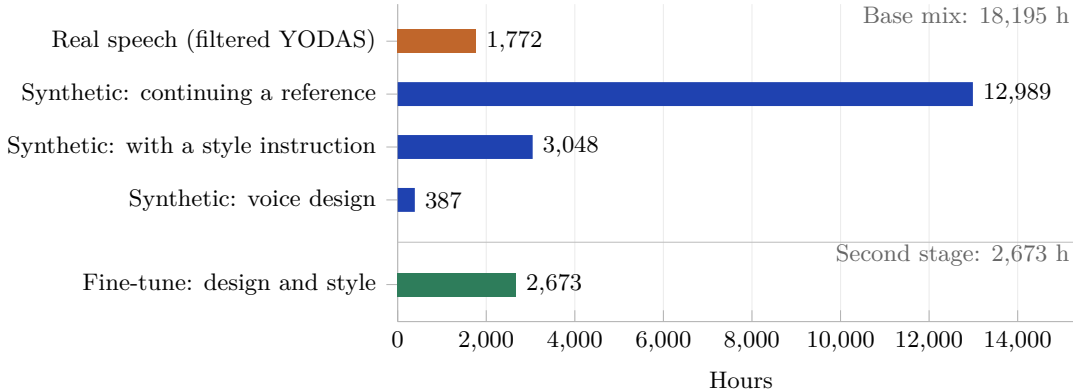


Figure 2: Training data. The base mix is 90% synthetic speech; a short second stage adds voice design and style instructions.

Text. The text comes from FineWeb-2 (Penedo et al., 2025) (40%), domain sentences written by Qwen3-30B-A3B (Yang et al., 2025) (25%), Wikipedia (22.5%) and YODAS transcripts (12.5%). After removing duplicates, 5.65M of 6.18M text units remain. Each unit is normalized (Section 4) with the normalizer version current at the time.

Voices. The speakers are our two built-in voices and a bank of 2,752 designed voices (1,283 female, 1,469 male). We create the bank from 5,000 text descriptions and drop near-duplicates whose speaker embeddings have a cosine similarity of 0.85 or more. Each clip is generated in one of three modes: continuing a reference recording (77%), following a reference together with an English style instruction (20%), or designing a voice from its description alone (3%). About 22.5% of the training clips are also passed through a simulated room, microphone, noise and codec channel.

Quality control. We keep a clip only if it passes every check in Table 1. In a manual audit of 681 generated clips, 79.2% passed.

3.3 Second stage and released data

The second-stage mix has 1.4M rows: voice descriptions, short paraphrased descriptions, style-instruction rows, and 400,000 plain rows that keep the model’s base behaviour. We release three datasets on Hugging Face:

- **Captions**¹: transcripts and style captions for the 835,267 real segments, without audio.
- **Synthetic speech**²: 3,411 hours at 48 kHz, designed voices only.

¹<https://huggingface.co/datasets/cloud0day3/alania-speech-captions-tr>

²<https://huggingface.co/datasets/cloud0day3/alania-synthetic-speech-tr>

Table 2: Examples of written input and the text Alania-2 receives.

Case	Written	Spoken form given to the model
Money with a suffix	<i>750 TL'dir</i>	<i>yedi yüz elli liradır</i>
Decimal amount	<i>12,50 TL</i>	<i>on iki lira elli kuruş</i>
Percentage with a suffix	<i>%2,5'lik</i>	<i>yüzde iki buçukluk</i>
Turkish initialism	<i>SGK</i>	<i>segeka</i>
Borrowed initialism	<i>CEO</i>	<i>siio</i>
Brand name	<i>PatientDesk</i>	<i>Peyşint Desk</i>
Brand name	<i>ChatGPT</i>	<i>Çet Ci Pi Ti</i>

- **Domain sentences**³: 1.44M sentences.

4 Text Normalization

Alania-2 reads Turkish spelling and has no phoneme input, so numbers, symbols, abbreviations and foreign words must first be rewritten into the words a speaker would say. Table 2 shows typical cases. Our normalizer is a sequence of rules, covered by 137 unit tests, for numbers, money, dates, times, ranges, phone numbers, doses, units, titles, addresses and symbols. After rewriting a number or symbol, it re-attaches the Turkish suffix that followed it and adjusts the suffix’s vowels to the new word (vowel harmony). A companion paper describes the normalizer in detail; here we summarize the parts that matter most for the model.

Initialisms. Turkish initialisms are read as Turkish letter names joined into one word, while some borrowed ones use English letter names. A native-speaker author reviewed 134 such readings and corrected 10.

Upper case. Words in capitals are lowercased with Turkish rules, so that *I* becomes *ı* and *İ* becomes *i*. Without this, VoxCPM2 read the city name *Iğdır* as *ığdır*.

Foreign words and brands. Read with Turkish spelling rules, many brand names and English words come out garbled. When a word is in mixed case or English, and rare in a 225,764-word Turkish vocabulary, we respell it in Turkish letters from its pronunciation in the CMU Pronouncing Dictionary.⁴

Lexicon. Automatic respelling is not always better than the written form, so we checked 100 common brand and English terms by hand. We generated each term both as written and respelled, and transcribed both with Duyu. A native-speaker author listened to both versions of every term and chose one, giving a lexicon of 17 respelled and 77 unchanged terms. Users can add their own entries. Table 3 shows the effect: for the 17 respelled terms, recognition similarity rises from 0.63 to 0.99.

³<https://huggingface.co/datasets/cloud0day3/alania-domain-text-tr>

⁴<https://github.com/cmuspinx/cmudict>

Table 3: Recognition of 100 brand and English terms by Duyu, with one voice and seed. Similarity is between the transcript and the term (0–1) $\uparrow$.

	As written	Respelled
Mean similarity, all 100 terms	0.911	0.961
Terms recognized exactly	72	85
Mean similarity, the 17 respelled terms	0.627	0.989

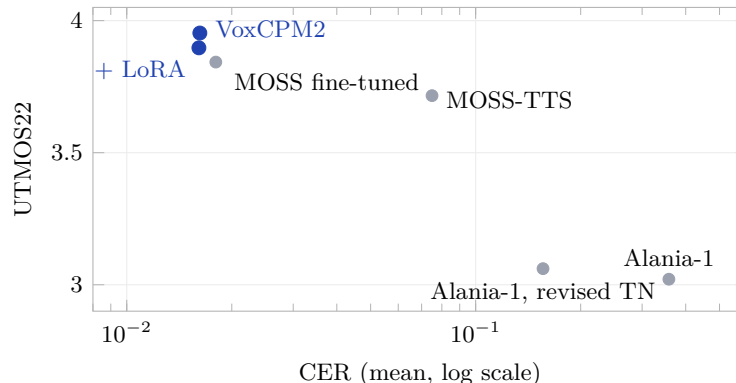


Figure 3: Choosing an architecture: intelligibility (CER, lower is better) against naturalness (UTMOS22, higher is better) on 120 Turkish prompts, 240 renders per system. The best systems sit at the top left.

Effect of normalizing at all. Normalization is not optional for Alania-2. On the companion paper’s 150-sentence test set, which covers numbers, dates, units, phone numbers, initialisms, brands and addresses, the model given raw written text had a word error rate of 66.7%, against 6.4% with normalized text. All key spans were read correctly in 7% of sentences with raw text and in 81% with normalized text. With raw text, none of the 96 sentences containing digits came out right: the model replaced numbers with invented words. Zero-shot VoxCPM2 degrades much less (CER 0.116 raw against 0.056 normalized on our adversarial prompts).

5 Model

5.1 Choosing an architecture

We compared candidate architectures on 120 Turkish prompts with two seeds each (Figure 3). Zero-shot VoxCPM2 was the most intelligible (CER 0.016), the most natural by UTMOS (3.95) and the most consistent between renders of the same voice (speaker similarity 0.83). Unlike Alania-1, which ran on past the end of the text in 7.5% of renders, it never did. Its VAE also preserved speaker identity through encoding and decoding far better than the SNAC codec (similarity 0.93 against 0.71). We therefore adopted the VoxCPM architecture.

5.2 Architecture and training

Alania-2 keeps VoxCPM2’s text tokenizer and its frozen audio VAE (Zhou et al., 2026), which encodes 16 kHz audio into 64-dimensional latents at 25 frames per second and decodes them to 48 kHz. A language-model backbone reads the text and the latents generated so far. A diffusion transformer then predicts the next patch of four latent frames (0.16 s of audio) in 10 flow-matching

Table 4: Alania-2 architecture, excluding the frozen audio VAE. Attention heads stay 128-dimensional to match VoxCPM’s position encoding.

Component	Layers $\times$ width	Heads	Params
Backbone LM	16×1024	8 (2 KV)	268M
Residual LM	6×1024	8 (2 KV)	72M
Feature encoder	8×768	6	69M
Diffusion decoder	8×768	6	72M
Total			488M

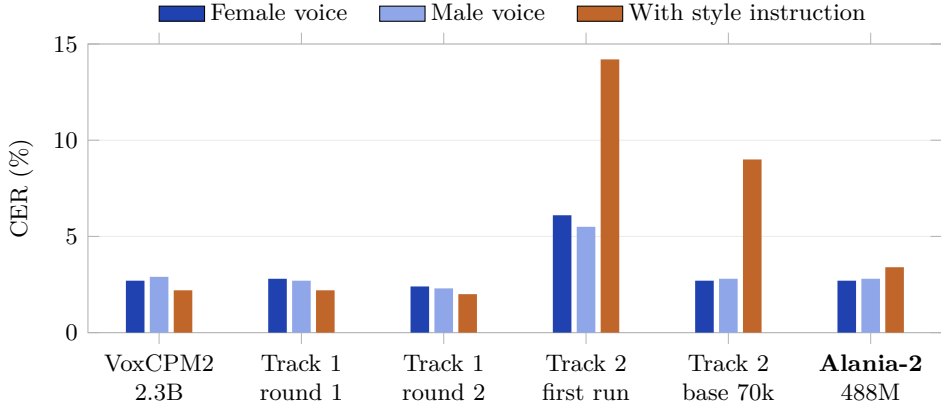


Figure 4: CER of the candidate models on our 292-clip suite (one seed; Whisper large-v3), lower is better. On its built-in voices, Alania-2 is close to the 2.3B models at under a quarter of their size, and the second training stage fixed most of the base model’s errors with style instructions. Full results are in Appendix A.

steps with classifier-free guidance. Our model is smaller than VoxCPM2 (Table 4), and we train all its weights from random initialization with the VoxCPM training code.

We pursued two tracks in parallel. **Track 1** continued training the full VoxCPM2 on our data (2.3B parameters in the released checkpoint, excluding the VAE). **Track 2** trained our own smaller model from scratch. A model trained from scratch needs many steps before it learns to align text with audio: a 1.24B-parameter variant never did within 54k steps (CER 0.84–0.88), while the 488M model did. The released model was trained in three stages:

1. about 64k steps on an earlier 8,904-hour mix;
2. 70k steps on the 18,195-hour base mix, with a batch of 32 per GPU, two gradient-accumulation steps, a learning rate of 10^{-4} and weight decay of 0.05;
3. the second stage on design and style data at a learning rate of 5×10^{-5} , stopped at step 4,000.

Training took on the order of 210 H200 GPU-hours, not counting data generation.

5.3 Comparison and choice

Figure 4 compares the candidates. On the two built-in voices, Alania-2’s CER is within half a point of VoxCPM2 and of both Track 1 models. With style instructions its CER is higher (3.4% against 2.0–2.2% for Track 1). It produced the requested gender for 30 of 32 voice designs, as often as any candidate. Its UTMOS is 0.2–0.3 lower than VoxCPM2’s, and it is weaker at cloning a voice from an English recording (Section 8). We chose Alania-2 because it is far cheaper to serve at nearly the same intelligibility: on one H200, it keeps up with real time at 12 requests per second, against about 8 for Track 1.

5.4 Voices, styles and design

The two built-in voices are designed voices, not recordings of real people. We chose them in a blind comparison of 12 candidates. For each voice we store a reference clip, an average speaker embedding and a pitch range (199.6–238.6 Hz for the female voice, 109.1–131.6 Hz for the male voice). At serving time, we use the pitch range to reject renders that drift away from the voice.

Style control comes from the captioned training rows. At first, the model sometimes read the instruction aloud: Turkish instructions on every take, and long English ones on 4 of 6. We now rewrite each instruction into a short English tag with a small LLM, after which all 54 test takes were clean.

Ready-made styles (warm, calm, energetic, smiling) keep CER between 0.33% and 1.82% and widen the pitch range by 0.5–1.4 semitones. Whispering and slow speech are beyond the model. For voice design we use a guidance scale of 3.0, chosen from a sweep between 1.5 and 4.0 in which higher guidance raised UTMOS (from 1.79 to 2.93) but lowered gender accuracy (from 30 to 27 of 32).

6 Evaluation

6.1 Freya-TR-Eval

Freya-TR-Eval (Pamuk et al., 2026) is a public Turkish benchmark of 495 sentences of 3–13 words, given as raw written text. It measures intelligibility: each system reads every sentence, a speech recognizer transcribes the audio, and the benchmark reports the word error rate (WER) against the original text. We re-ran it for 15 systems under one protocol, plus the bare Alania-2 model without our serving checks, and released the harness and results. Appendix B lists the exact checkpoint of every system.⁵ Each system reads every sentence three times. We band-limit the audio to 8 kHz, resample it to 16 kHz, transcribe it with Whisper large-v3 (Radford et al., 2023) through faster-whisper⁶ (beam size 5) and score it with the benchmark’s own text normalization. We also report UTMOS22 on the full-band audio and the real-time factor on one L4 GPU.

Figure 5 shows the results. Through our API, Alania-2 reaches 1.44% WER, seventh of the 15 systems. The two best are compact models of under 9M parameters: EMA Lightning at 1.04% and our Antalia-2 Mini at 1.05%. The benchmark authors report 1.43% for ElevenLabs v4, which we did not run. Differences of a tenth of a point are within the spread between the three readings. Our serving checks and retries (Section 9) lower the bare model’s 1.51% to 1.44%. Compared with Alania-1, Alania-2 cuts WER four-fold and character error rate (CER) six-fold (Table 5).

6.2 What we did not measure

This report does not yet include a listening test with many listeners. Decisions that needed listening, such as choosing the architecture and the built-in voices, were made in blind comparisons by one native speaker, and our only naturalness measure is UTMOS22, which was trained on English. Intelligibility results depend on the recognizer and on how text is normalized before scoring, so we name the recognizer with every result.

⁵<https://huggingface.co/datasets/PatientdeskAI/freya-tr-eval-rerun>

⁶<https://github.com/SYSTRAN/faster-whisper>

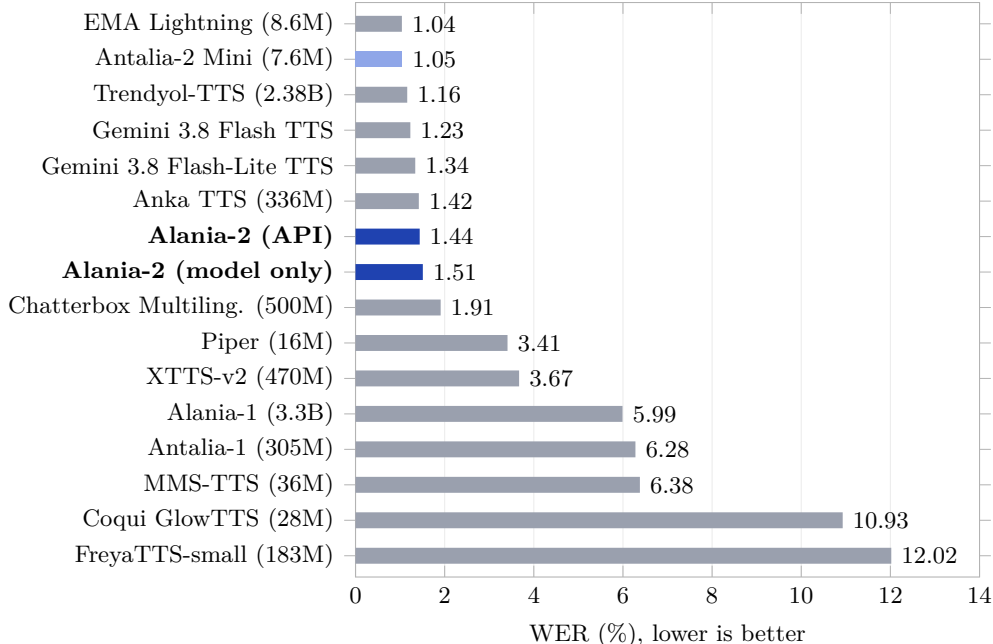


Figure 5: Freya-TR-Eval word error rate (495 sentences, three readings each, faster-whisper large-v3), lower is better. Alania-2 through our API is seventh of the 15 systems; the bare model without serving checks is shown for reference. Antalia-2 Mini is our 7.6M-parameter single-voice model (Section 7).

Table 5: Alania-1 and Alania-2 on Freya-TR-Eval, both through our API. RTF is the real-time factor on one L4 GPU (Alania-1 measured on 100 sentences).

	Alania-1	Alania-2
Architecture	LLM over SNAC tokens	VoxCPM
Parameters	3.3B	488M
Sample rate	24 kHz	48 kHz
WER (%) ↓	5.99	1.44
CER (%) ↓	1.81	0.28
UTMOS22 ↑	3.19	3.67
RTF ↓	2.49	0.151

7 How Large Does the Model Need to Be?

Alania-2 has 488M parameters, while open speech models of several billion parameters are available, so it is fair to ask why we did not build on a larger one. We did try: in our experiments, the larger models were barely more intelligible, and a far smaller model trained with the same recipe matched the best systems we measured.

7.1 Larger models trained on our data

Table 6 compares Alania-2 with the larger models from Section 5. Continuing to train the full VoxCPM2, at 2.3B parameters, improved CER on the built-in voices by only 0.3–0.5 points over Alania-2. The larger model did keep clear advantages elsewhere. It followed style instructions more accurately (CER 2.0% against 3.4%), it sounded somewhat more natural (UTMOS 0.16 higher on the female voice), and it cloned voices from English recordings far better (similarity 0.64 against

Table 6: Model size against quality on our 292-clip suite (one seed; CER from Whisper large-v3, lower is better) and serving capacity. Training: “continued” means further training of VoxCPM2 on our data, “scratch” random initialization. Requests/s is the request rate at which about 99% of streams keep ahead of real time on one H200, higher is better.

Model	Params	Training	CER (%) ↓			UTMOS ↑	Requests/s ↑
			Female	Male	Style		
VoxCPM2, zero-shot	2.3B	released	2.7	2.9	2.2	3.77	–
Track 1, round 2	2.3B	continued	2.4	2.3	2.0	3.73	8
Track 2-L	1.24B	scratch	did not align within 54k steps (CER 84–88%)				
Alania-2	488M	scratch	2.7	2.8	3.4	3.57	12

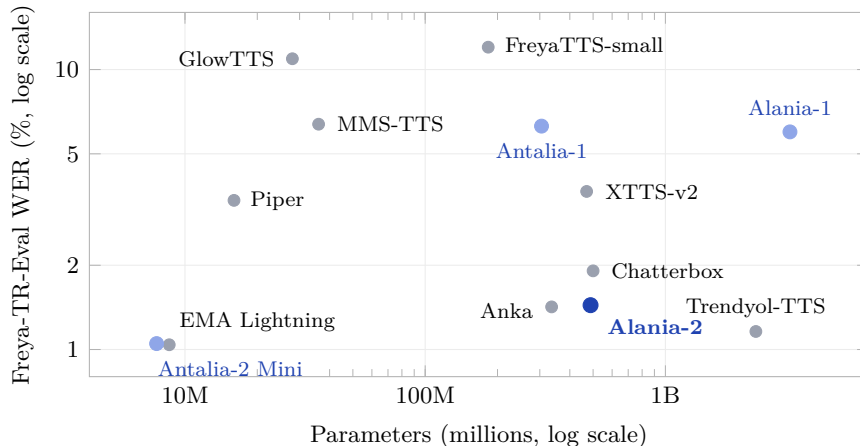


Figure 6: Freya-TR-Eval word error rate against model size for the 13 systems with published sizes (both axes logarithmic). Our models are in blue. Size does not predict intelligibility.

0.41, measured before Alania-2’s second training stage). Training a larger model from scratch did not help either: our 1.24B-parameter variant had still not learned to align text with audio after 54k steps, when the 488M model already had. At serving time, the smaller model handles 1.5 times as many requests per GPU.

7.2 Size and intelligibility across systems

The same holds across the systems in our Freya-TR-Eval re-run (Figure 6). Model size does not predict word error rate: the three most intelligible systems with published sizes span 7.6M to 2.4B parameters, and our 3.3B-parameter Alania-1 misreads more than four times as many words as the 488M Alania-2. Training data and text handling separate these systems far more than size does.

7.3 Going much smaller

To test how far the recipe scales down, we trained Antalia-2 Mini, an openly released model with a single voice. It has 7.62M parameters: a 3.69M-parameter acoustic model that generates mel spectrograms with few-step flow matching, and a 3.93M-parameter vocoder that produces 48 kHz audio. Like Alania-2, it learns from synthetic speech, here 1,578 hours of one voice generated by VoxCPM2, and it reads text through the same normalizer (Section 4). On Freya-TR-Eval it reaches 1.05% WER and 0.20% CER, the lowest CER of the 15 systems we ran. Its teacher, VoxCPM2

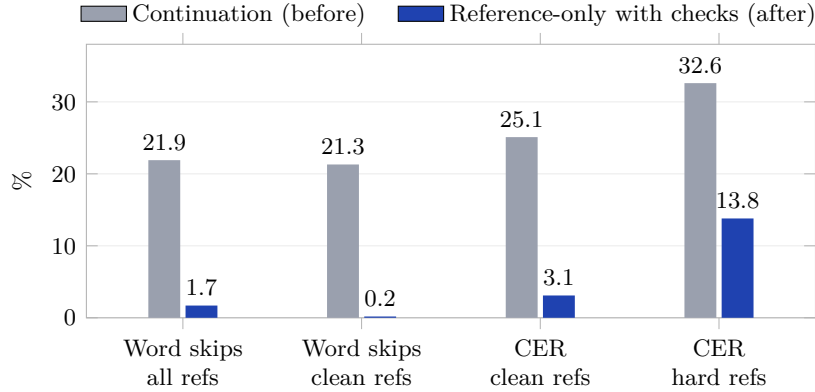


Figure 7: Streamed instant clones in continuation mode (before) and reference-only mode with per-chunk checks (after); lower is better. 200 streams each, with the same seeds and texts of 121–249 characters. “Hard” references contain reverberation, music, a laptop microphone, noise or MP3 compression.

speaking the same voice, scored 0.92% WER and 0.20% CER in a single run on our development harness. A model about 300 times smaller thus keeps nearly all of its teacher’s intelligibility.

Antalia-2 Mini also shows what the extra parameters pay for. It has one voice and no cloning, voice design or style control, and it sounds less natural than Alania-2 (UTMOS 3.25 against 3.67). For reading Turkish correctly in a fixed voice, a few million parameters suffice when the data and the text normalization are right. The larger capacity of Alania-2 goes into flexibility: many voices, cloning from a short recording, voice design and style instructions. We chose 488M as the smallest size we tried that kept these abilities while serving the most requests per GPU.

8 Voice Cloning

Alania-2 offers two kinds of cloned voice. An **instant clone** is conditioned on a short reference recording at request time. A **professional clone** is a LoRA adapter trained for one voice on one to five minutes of speech. The reference recordings in this section come from consenting contributors to a Turkish speech corpus and from FLEURS (Conneau et al., 2023). We measure CER and skipped words with Whisper large-v3 and word alignment, and speaker similarity as the cosine similarity of ECAPA embeddings between the generated audio and a held-out recording of the same speaker.

8.1 Continuing a reference skips words

VoxCPM can use a reference recording in two ways. In **continuation** mode, the reference audio and its transcript come before the new text, and the model continues the recording. In **reference-only** mode, the reference audio sets the voice and no transcript is given.

We first served clones in continuation mode, and users reported skipped words, silent gaps and loops. With references of 20s or more, continuation skipped about half of every text, often whole chunks, and a wrong reference transcript had the same effect at any length. Switching to reference-only mode, together with per-chunk checks (Section 9), fixed this without losing speaker identity (Figure 7). Over 200 streams with the same seeds and texts, skipped words fell from 21.9% to 1.7%, and to 0.18% on clean references. Speaker similarity on clean references was unchanged (0.569 against 0.568). The near-silent chunks that users heard as gaps disappeared (from 45 to 0), and the longest pause where two chunks join mid-sentence fell from 1.04s to 0.37s.

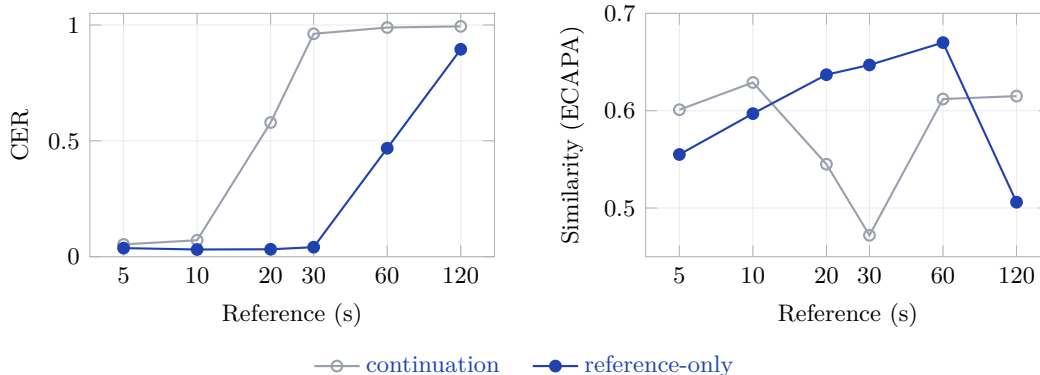


Figure 8: Instant clones as the reference grows, using the first N seconds of each recording: CER (left, lower is better) and speaker similarity (right, higher is better). Reference-only mode is stable up to 30 s.

8.2 How much reference audio helps

Figure 8 compares the two modes as the reference grows from 5 to 120 s (four speakers, three texts, three seeds). In reference-only mode, similarity rises from 0.56 at 5 s to 0.64–0.67 at 20–30 s. Beyond that, the model breaks down, with a CER of 0.47 at 60 s and 0.90 at 120 s. Continuation mode is already unusable at 20 s. Choosing the cleanest 20-second window of a recording is better than taking its first 20 seconds (similarity 0.672 against 0.637, CER 0.024 against 0.032). We therefore accept references of 3–30 s, recommend 20–30 s, and automatically pick the cleanest 20-second window from longer uploads. Encoding the reference takes 31 ms at 5 s and 404 ms at 120 s, while first audio stays between 83 and 114 ms.

Chaining chunks does not help. We also tried conditioning each chunk of a long text on the end of the previous chunk. Over 80 paired renders, this lowered CER by 0.035–0.069 but also lowered similarity by 0.058–0.074, and the voice drifted from chunk to chunk. Every chunk therefore stays conditioned on the original reference.

Reference quality. We convert saved references to mono 48 kHz at -20 dBFS and check them for background noise, echo, clipping and implausible transcripts. Background noise only 12 dB below the speech made the model skip 29–49% of words.

8.3 Professional clones

A professional clone is a rank-32 LoRA adapter (Hu et al., 2022) on the backbone and the diffusion decoder. The training pipeline splits the upload into 3–15 s utterances with a signal-to-noise ratio above 15 dB, transcribes them, holds out 10% for evaluation, and trains for 200 steps on 80 s of speech, rising linearly to 400 steps for five minutes. The adapter is used only if it beats the instant clone made from the best 20-second reference on the held-out speech: similarity must rise by at least 0.02, and CER must stay below $\max(0.05, 1.5c)$, where c is the instant clone’s CER.

Training takes 84–126 s on one A100, and a whole job finishes in 3–4.5 minutes. Figure 9 shows large similarity gains in every run. One speaker’s adapter trained on 580 s of speech (similarity 0.68) scored below the same speaker’s 78-second adapter (0.72), which suggests that matching the recording session matters more than the amount of audio at this scale. Cloning a voice from an English recording into Turkish speech remains weak: similarity 0.41, against 0.64 for Track 1, measured before the second training stage.

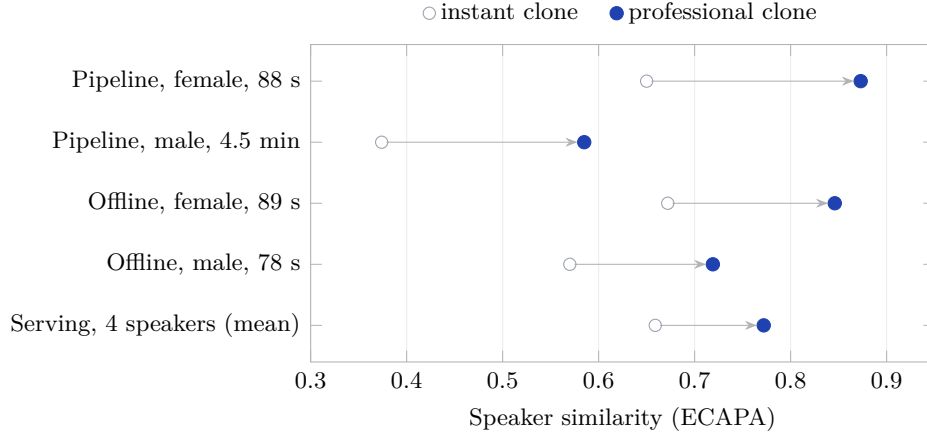


Figure 9: Speaker similarity of professional clones (LoRA adapters) against instant clones of the same speakers, on held-out speech; higher is better. Each row is one run; the last is the mean over four speakers through the serving engine.

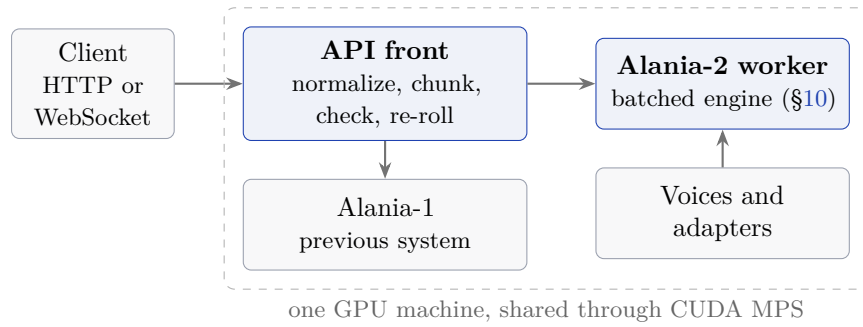


Figure 10: Serving layout on each GPU machine.

8.4 Verification

Because a professional clone reproduces a voice closely, we require the voice’s owner to be present. The speaker reads a random Turkish sentence aloud in the browser. The sentence is drawn from about three million combinations and expires after 15 minutes. Duyu must recognize the sentence, and the speaker embedding of the reading must match the uploaded recordings. On 13 consenting speakers (84 same-speaker pairs across recordings and 2,460 different-speaker pairs), a threshold of 0.45 rejects 0.5% of genuine readings and accepts 0.45% of impostor pairs. In a live test, a different speaker’s reading scored 0.20 and was rejected within 3s. Our API also refuses to synthesize any pending challenge sentence, so the check cannot be passed by generating the reading with an instant clone through our own service.

9 Serving

Each GPU machine runs the Alania-2 worker and an API front that also serves Alania-1 (Figure 10). CUDA’s Multi-Process Service (MPS) lets the two models share the GPU fairly: without it, four Alania-1 streams next to an Alania-2 load fell from 100% to 3% keeping up with real time; with it, they held 100% next to 24 Alania-2 streams.

From text to audio. The front normalizes the text (Section 4) and splits it into chunks of at most 140 characters, preferring sentence and then clause boundaries. It then resolves the voice (built-in, designed, instant clone or professional clone) and streams the first chunk as it is generated, in steps of 0.16 s.

Checks and retries. Later chunks are generated up to two chunks ahead of playback and checked before they are sent. A chunk is rejected if it is empty; if it runs on too long (more than 2.0 s plus 0.11 s per character); if its length does not fit its text; if it contains a pause of more than 1 s; if it has no voiced speech; or, for the built-in voices, if its median pitch is outside the voice’s range. A rejected chunk is generated again with a new seed, up to twice, and the best attempt is kept. If waiting would make playback stall, the chunk is sent unchecked. When joining chunks, we trim their edge silence and insert 350 ms of silence after a sentence and 120 ms after a clause. Every response is labelled as AI-generated in its headers.

Deployment. We run on spot RTX PRO 6000 machines in one EU region, with on-demand A100 machines as a floor that is never preempted. A new machine serves Alania-2 81 s after boot. Professional clones are trained on a separate, always-on GPU machine in the EU.

10 Inference Engine

The worker runs Alania-2 on Nano-vLLM-VoxCPM (a710128, 2025), an inference engine for VoxCPM based on Nano-vLLM (GeeekExplorer, 2025). It batches concurrent streams and keeps the backbone’s key/value cache in fixed-size blocks (Kwon et al., 2023). We changed the engine in three places to cut first-audio latency and raise throughput, and in several more to serve many fine-tuned voices from one model. We checked each change against the unmodified engine on the same seeds and texts.

10.1 Latency and throughput

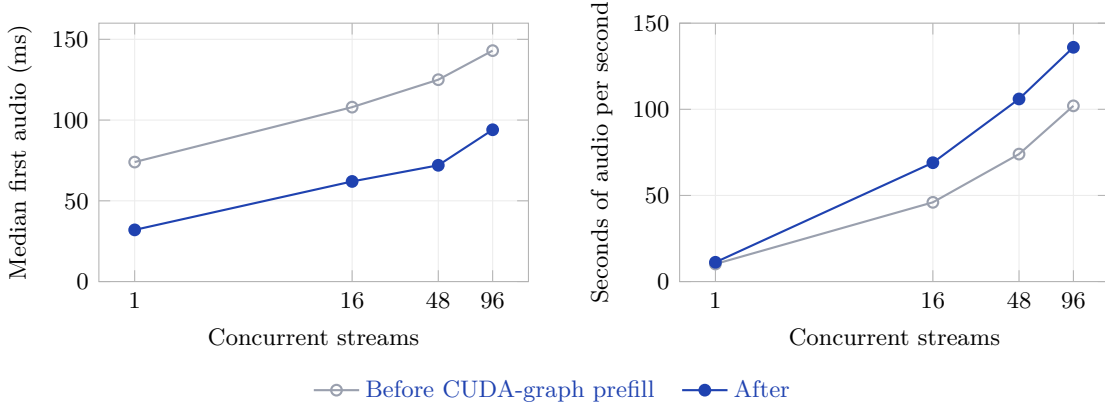
Table 7 lists the three changes:

- **Batched VAE decoding.** The audio decoder ran once per stream. Running it once per batch cut decoding at 96 streams from 110 to 27 ms per step. At 48 streams, throughput rose from 41 to 74 seconds of audio per second, and the share of streams that kept ahead of real time rose from none to all.
- **CUDA-graph prefill.** We capture the prompt step as CUDA graphs, as the engine already did for later steps. Single-stream first audio fell from 74 to 32 ms, and the output stayed bit-identical (23 of 23 renders).
- **Kernel warm-up.** A batch size the engine had not seen before stalled every stream for about 1.4 s. We now run the common batch sizes once at start-up.

Figure 11 shows the effect of the prefill change as load grows, with batched decoding already in place. With all three changes, median first audio is 32 ms for one stream and 94 ms for 96 concurrent streams, which together produce 136 seconds of audio per second. Through the API on the same machine, a single stream starts in 97 ms for a fixed voice, 109 ms for a clone and 78 ms for a designed voice. On an H200 with Poisson arrivals of ~ 5 -s requests, median first audio is 51 ms at 2 requests per second and 75 ms at 12 requests per second, where 98.6% of streams still keep ahead of real time.

Table 7: Engine changes, measured on one A100 (40 GB) at the worker, excluding the network.

Change	Metric	Before	After
Batched VAE decoding	VAE decode per step, 96 streams (ms) ↓	110	27
	Audio seconds per second, 48 streams ↑	41	74
CUDA-graph prefill	First audio, 1 stream (ms) ↓	74	32
	First audio, 96 streams (ms) ↓	143	94
Kernel warm-up	Stall on a new batch size (s) ↓	~1.4	none

**Figure 11:** First-audio latency (left) and throughput (right) on one A100 as concurrent streams grow, before and after CUDA-graph prefill. Batched VAE decoding is on in both.

10.2 Serving many fine-tuned voices

Professional clones (§8) are LoRA adapters on one shared base model, as in S-LoRA (Sheng et al., 2024) and Punica (Chen et al., 2024). Each request names its adapter. Up to eight adapters stay on the GPU, and requests with different adapters, or none, share a batch. Adapters load from storage on first use and are cached in host memory and on disk. A broken adapter falls back to the instant clone of the same voice.

Prefix-cache correctness. The engine reuses cached key/value blocks when a new request starts with the same tokens as an earlier one. Its cache key covered only the tokens, so a request for a fine-tuned voice could reuse blocks that a plain request had computed from the same reference and text. We saw 8 of 9 prompt blocks reused this way, and 3 of 3 fine-tuned renders differed from renders with an empty cache. We now seed the cache key with the adapter’s name and a digest of its weights. Fine-tuned renders then match empty-cache renders bit for bit, and plain requests keep their existing keys.

Two smaller fixes. We replaced a LoRA kernel that would not compile on our Triton version; outputs agree to within 2×10^{-7} . Batches with a single adapter now run as plain matrix products, which cut the adapter’s first-audio overhead from 33 to 13 ms.

With these fixes, quality through the engine matches the reference implementation (Table 8). In mixed batches, every render was closest to its own speaker (60 of 60), and throughput with every stream on an adapter is 4–15% below plain clones.

Table 8: Professional clones through the engine (4 speakers $\times$ 5 texts $\times$ 3 seeds). Speaker similarity $\uparrow$ / CER $\downarrow$.

Path	Instant clone	Professional clone
Reference implementation	0.651 / 0.019	0.770 / 0.026
Our engine	0.659 / 0.019	0.772 / 0.028
Our engine, mixed batches	0.676 / 0.023	0.780 / 0.026

11 Conclusion

We presented Alania-2, a 488M-parameter Turkish TTS model trained mostly on synthetic speech. On its built-in voices, it comes within half a point of CER of a fine-tuned model more than four times its size, generates 48 kHz audio, and starts streaming in tens of milliseconds. Larger models brought little intelligibility, and a 7.6M-parameter model trained with the same recipe matched the best systems we measured; extra parameters pay for flexibility rather than accuracy. Several of the most useful lessons concerned the system around the model. Text normalization decides whether numbers and names are read correctly. Cloning by continuing a reference fails with long references, while conditioning on the reference audio alone does not, and 20–30 seconds of reference audio works best. A low-rank adapter trained in about two minutes gives a close voice match, and serving many such adapters needs care with caching. Next, we plan a listening test with many native listeners, audio watermarking, live verification for instant clones, and retraining on text from the current normalizer.

12 Limitations

Our evaluation has several gaps. First, naturalness rests on UTMOS22, a predictor trained on English speech, and on blind comparisons by one native listener. Second, intelligibility depends on the speech recognizer and on the text normalization used for scoring, and the same recognizers also helped build our training data. Third, ECAPA speaker similarity separates speakers on our data, but differences below about 0.05 should not be over-read: a style instruction alone moves similarity to the plain voice from about 0.91 to between 0.64 and 0.80. Fourth, most results come from a single seed or a single run.

The model itself has limits. Ninety percent of its training audio comes from VoxCPM2, so Alania-2 inherits its teacher’s prosody and artefacts, including narrow high-frequency tones in some voices that we filter out of the synthetic data but do not yet suppress at serving time. The base training text was also normalized before the native-speaker review of Section 4, so the model now receives some readings, including all brand respellings, that it never saw in training. Finally, exact GPU-hours and some early training configurations were not recorded.

13 Ethics and Safety

Training data. The real speech comes from public YouTube videos distributed in YODAS under Creative Commons terms. Our synthetic text includes Wikipedia (CC BY-SA) and FineWeb-2 (ODC-By). The built-in voices are designed voices that do not belong to any real person, and our released synthetic speech contains designed voices only. We do not release the model weights.

Cloning. Cloning someone’s voice without their agreement can enable fraud, harassment and disinformation. Every clone requires the user to state that they have the speaker’s consent, and we record this statement with the voice. Professional clones also require the live reading and speaker match of Section 8, with separate consent to the biometric comparison. Speaker embeddings are computed in memory and never stored, uploaded recordings are deleted seven days after training, customer voice data stays in the EU, and every response is labelled as AI-generated.

These controls have gaps. Instant clones have no live verification, so only the user’s statement of consent stands in the way of misuse. We do not yet watermark generated audio (San Roman et al., 2024), and we have no specific policy on the voices of public figures beyond requiring consent. Every voice in this report’s cloning experiments belongs to a consenting contributor or comes from FLEURS.

References

- a710128. Nano-vLLM-VoxCPM: An inference engine for VoxCPM based on Nano-vLLM. <https://github.com/a710128/nanovllm-voxcpm>, 2025.
- Edresson Casanova, Kelly Davis, Eren Gölge, Gökem Gökner, Iulian Gulea, Logan Hart, Aya Aljafari, Joshua Meyer, Reuben Morais, Samuel Olayemi, and Julian Weber. XTTS: a massively multilingual zero-shot text-to-speech model. In *Interspeech 2024*, pages 4978–4982, 2024. doi: 10.21437/Interspeech.2024-2016.
- Lequn Chen, Zihao Ye, Yongji Wu, Danyang Zhuo, Luis Ceze, and Arvind Krishnamurthy. Punica: Multi-tenant LoRA serving. In *Proceedings of Machine Learning and Systems (MLSys)*, volume 6, pages 1–13, 2024.
- Sanyuan Chen, Chengyi Wang, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, Lei He, Sheng Zhao, and Furu Wei. Neural codec language models are zero-shot text to speech synthesizers. *IEEE Transactions on Audio, Speech and Language Processing*, 33:705–718, 2025a. doi: 10.1109/TASLPRO.2025.3530270.
- Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, Jian Zhao, Kai Yu, and Xie Chen. F5-TTS: A fairytaler that fakes fluent and faithful speech with flow matching. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6255–6271, 2025b. doi: 10.18653/v1/2025.acl-long.313.
- Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna. FLEURS: Few-shot learning evaluation of universal representations of speech. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 798–805, 2023. doi: 10.1109/SLT54892.2023.10023141.
- Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynek. ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in TDNN based speaker verification. In *Interspeech 2020*, pages 3830–3834, 2020. doi: 10.21437/Interspeech.2020-2650.
- Zhihao Du, Qian Chen, Shiliang Zhang, Kai Hu, Heng Lu, Yexin Yang, Hangrui Hu, Siqi Zheng, Yue Gu, Ziyang Ma, Zhifu Gao, and Zhijie Yan. CosyVoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens. arXiv:2407.05407, 2024.
- GeeekExplorer. Nano-vLLM: A lightweight vLLM implementation built from scratch. <https://github.com/GeeekExplorer/nano-vllm>, 2025.

- Yuan Gong, Yu-An Chung, and James Glass. AST: Audio spectrogram transformer. In *Interspeech 2021*, pages 571–575, 2021. doi: 10.21437/Interspeech.2021-698.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.
- Jaehyeon Kim, Sungwon Kim, Jungil Kong, and Sungroh Yoon. Glow-TTS: A generative flow for text-to-speech via monotonic alignment search. In *Advances in Neural Information Processing Systems*, volume 33, pages 8067–8077, 2020.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles (SOSP)*, pages 611–626, 2023. doi: 10.1145/3600006.3613165.
- Xinjian Li, Shinnosuke Takamichi, Takaaki Saeki, William Chen, Sayaka Shiota, and Shinji Watanabe. YODAS: Youtube-oriented dataset for audio and speech. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–8, 2023. doi: 10.1109/ASRU57964.2023.10389689.
- Yunus Emre Ozkose, Alperen Kahraman, and Ali Haznedaroglu. A comprehensive objective evaluation of modern text-to-speech for Turkish using speech quality assessment models. In *Speech and Computer (SPECOM 2026)*, Lecture Notes in Computer Science. Springer, 2026.
- Ahmet Erdem Pamuk, Ömer Yentür, Ahmet Tunga Bayrak, Yavuz Alp Sencer Öztürk, and Mustafa Yavuz. FreyaTTS: A compact tokenizer-free flow-matching transformer for Turkish-first speech synthesis. arXiv:2607.09530, 2026.
- Guilherme Penedo, Hynek Kydlíček, Vinko Sabolčec, Bettina Messmer, Negar Foroutan, Amir Hossein Kargaran, Colin Raffel, Martin Jaggi, Leandro Von Werra, and Thomas Wolf. FineWeb2: One pipeline to scale them all – adapting pre-training data processing to every language. arXiv:2506.20920, 2025.
- Vineel Pratap, Andros Tjandra, Bowen Shi, Paden Tomasello, Arun Babu, Sayani Kundu, Ali Elkahky, Zhaoheng Ni, Apoorv Vyas, Maryam Fazel-Zarandi, Alexei Baevski, Yossi Adi, Xiaohui Zhang, Wei-Ning Hsu, Alexis Conneau, and Michael Auli. Scaling speech technology to 1,000+ languages. arXiv:2305.13516, 2023.
- Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, pages 28492–28518, 2023.
- Mirco Ravanelli, Titouan Parcollet, Peter Plantinga, Aku Rouhe, Samuele Cornell, Loren Lugosch, Cem Subakan, Nauman Dawalatabad, Abdelwahab Heba, Jianyuan Zhong, Ju-Chieh Chou, Sung-Lin Yeh, Szu-Wei Fu, Chien-Feng Liao, Elena Rastorgueva, François Grondin, William Aris, Hwidong Na, Yan Gao, Renato De Mori, and Yoshua Bengio. SpeechBrain: A general-purpose speech toolkit. arXiv:2106.04624, 2021.
- Chandan K. A. Reddy, Vishak Gopal, and Ross Cutler. DNSMOS: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors. In *ICASSP 2021 – 2021 IEEE*

- International Conference on Acoustics, Speech and Signal Processing*, pages 6493–6497, 2021. doi: 10.1109/ICASSP39728.2021.9414878.
- Takaaki Saeki, Detai Xin, Wataru Nakata, Tomoki Koriyama, Shinnosuke Takamichi, and Hiroshi Saruwatari. UTMOS: UTokyo-SaruLab system for VoiceMOS challenge 2022. In *Interspeech 2022*, pages 4521–4525, 2022. doi: 10.21437/Interspeech.2022-439.
- Robin San Roman, Pierre Fernandez, Alexandre Défossez, Teddy Furon, Tuan Tran, and Hady Elsahar. Proactive detection of voice cloning with localized watermarking. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pages 43180–43196, 2024.
- Ying Sheng, Shiyi Cao, Dacheng Li, Coleman Hooper, Nicholas Lee, Shuo Yang, Christopher Chou, Banghua Zhu, Lianmin Zheng, Kurt Keutzer, Joseph E. Gonzalez, and Ion Stoica. SLoRA: Scalable serving of thousands of LoRA adapters. In *Proceedings of Machine Learning and Systems (MLSys)*, volume 6, pages 296–311, 2024.
- Hubert Siuzdak, Florian Grötschla, and Luca A. Lanzendörfer. SNAC: Multi-scale neural audio codec. arXiv:2410.14411, 2024.
- Jörgen Valk and Tanel Alumäe. VoxLingua107: A dataset for spoken language recognition. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pages 652–658, 2021. doi: 10.1109/SLT48900.2021.9383459.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiayi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report. arXiv:2505.09388, 2025.
- Rustem Yeshpanov, Saida Mussakhoyeva, and Yerbolat Khassanov. Multilingual text-to-speech synthesis for Turkic languages using transliteration. In *Interspeech 2023*, pages 5521–5525, 2023. doi: 10.21437/Interspeech.2023-249.
- Yixuan Zhou, Guoyang Zeng, Xin Liu, Xiang Li, Renjie Yu, Ziyang Wang, Runchuan Ye, Weiyue Sun, Jiancheng Gui, Kehan Li, Zhiyong Wu, and Zhiyuan Liu. VoxCPM: Tokenizer-free TTS for context-aware speech generation and true-to-life voice cloning. arXiv:2509.24650, 2025.
- Yixuan Zhou, Guoyang Zeng, Xin Liu, Xiang Li, Renjie Yu, Jiancheng Gui, Jiaheng Wu, Ziyang Wang, Xudong Shen, Runchuan Ye, Zhisheng Zhang, Jiuyang Zhou, Bingsong Bai, Weiyue Sun, Mengyuan Deng, Qundong Shi, Zhiyong Wu, and Zhiyuan Liu. VoxCPM2 technical report. arXiv:2606.06928, 2026.

A Full Candidate Results

B Systems in the Freya-TR-Eval Re-run

Table 9: Candidates on the 292-clip suite (one seed). CER from Whisper large-v3, UTMOS22, ECAPA similarity to the voice reference; “gender” counts voice designs that matched the requested gender.

Model	Female voice			Male voice			Style			Design
	CER	UTMOS	Sim	CER	UTMOS	Sim	CER	UTMOS	Sim	Gender
VoxCPM2, zero-shot (2.3B)	.027	3.77	.748	.029	3.53	.812	.022	2.96	.542	23/32
Track 1, round 1 (2.3B)	.028	3.77	.690	.027	3.54	.806	.022	3.66	.667	25/32
Track 1, round 1 LoRA	.024	3.78	.707	.028	3.60	.810	.037	3.71	.714	23/32
Track 1, round 2 (2.3B)	.024	3.73	.761	.023	3.50	.737	.020	3.13	.534	27/32
Track 2, first run (488M)	.061	3.48	.768	.055	3.12	.723	.142	3.21	.663	30/32
Track 2, base 70k (488M)	.027	3.64	.773	.028	3.28	.739	.090	3.29	.666	24/32
Alania-2 (fine-tune 4k)	.027	3.57	.766	.028	3.24	.741	.034	3.23	.633	30/32
Fine-tune 12k	.027	3.62	.769	.028	3.26	.736	.044	3.22	.612	30/32

Table 10: Systems in our Freya-TR-Eval re-run (Figure 5), with the exact checkpoint or service we ran. Model identifiers are Hugging Face repositories unless noted. Parameter counts are those in our released results; “–” means not published.

System	Params	Checkpoint or service	Paper
EMA Lightning	8.6M	ema-lightning 1.0.1 (PyPI), canberkkkkkk/ema-lightning	–
Antalia-2 Mini	7.62M	1.0.0, cloud0day3/antalia-mini	Section 7
Trendyol-TTS	2.38B	Trendyol/Trendyol-TTS	–
Gemini 3.8 Flash TTS	–	Vertex AI, gemini-3.8-flash-tts , voice Kore	–
Gemini 3.8 Flash-Lite TTS	–	Vertex AI, gemini-3.8-flash-lite-tts , voice Kore	–
Anka TTS	336M	anka-tts v0.1 (PyPI)	–
Alania-2	488M	our API, and the bare model	this report
Chatterbox Multilingual	500M	resemble-ai/chatterbox (GitHub)	–
Piper	16M	tr_TR-dfki-medium from rhasspy/piper-voices	–
XTTS-v2	470M	coqui/XTTS-v2	Casanova et al. (2024)
Alania-1	3.3B	our API	–
Antalia-1	305M	cloud0day3/antalia-1	–
MMS-TTS	36M	facebook/mms-tts-tur	Pratap et al. (2023)
Coqui GlowTTS	28M	tts_models/tr/common-voice/ glow-tts (Coqui TTS)	Kim et al. (2020)
FreyaTTS-small	183M	freyavoice/Freya-TTS	Pamuk et al. (2026)