

The Long Tail of the Alphabet: Do Multilingual TTS Models Pronounce Every Letter?

Anonymous ACL submission

Abstract

042

We design TAILPROBE, a benchmark that tests whether multilingual text-to-speech (TTS) models pronounce every consonant letter of a script. TAILPROBE places each letter in three word positions and two sentence frames, and compares the model with a per-letter *human floor*: human speech scored by the same phone recogniser, so that recogniser errors are not blamed on the model. We test 7 released models in 29 languages and 16 scripts (126,194 clips) and find three failure modes. Letters with no input row are silently dropped. Letters that are rare in a small training set fall far below the human floor (MMS-TTS). The first consonant of an utterance is weaker than in human speech in all 7 models under the main recogniser. Controlled finetuning shows that a new letter is learned well only when its row gets a strong training signal. Lead-in-and-cut, a repair without training, speaks a short phrase before the sentence and cuts it from the audio; it improves utterance-initial consonants by 7.7 points across four models. Editing or retraining only the rows of a released model did not help. Pre-registration and held-out replication make the results checkable. All probes, floors, scores and code are at <https://tailprobe.pages.dev>.

1 Introduction

Neural text-to-speech (TTS) models turn each input symbol into a learned vector, its *input row*. The symbol may be a character (VITS/MMS-TTS, Kim et al., 2021; Pratap et al., 2024; F5-TTS, Chen et al., 2025), a token from a small vocabulary that also contains every character (Chatterbox, Resemble AI, 2025; XTTS, Casanova et al., 2024), or a byte-level BPE token that can split a rare letter into pieces (Ye et al., 2025; Zhu et al., 2026). Whatever the unit, the model must pronounce every letter it is given.

This is hard for rare letters. Every script has a long tail: a few letters are very common, and many are rare (Figure 2a). TTS models for such scripts

are often trained on only tens of hours of speech. Some letters fail completely. A Telugu speaker who types ఌ (ña) gets no consonant from MMS-TTS: the model has no input row for this letter, and its tokenizer drops it without warning. Earlier work shows that rare character pairs (Anand et al., 2024) and unbalanced word coverage (Taylor and Richmond, 2019) hurt TTS pronunciation, and that text LLMs have under-trained tokens (Land and Bartolo, 2024; Li et al., 2024). But no study has measured pronunciation letter by letter across released models and scripts against a human reference, separated the causes, or tested repairs.

In this paper, we design TAILPROBE, a benchmark that tests letters one by one and compares each score with human speech scored by the same recogniser (Figure 1). We use it to answer three questions. (Q1) How common are letter failures (§4)? (Q2) What causes them (§5)? (Q3) Can they be repaired (§6)? In short, every model has letters that fall clearly below human speech. They come from missing input rows, rarity in a small training set, and the start of an utterance (the *onset*). Our main contributions are:

(1) A letter-level benchmark with a human floor.

To the best of our knowledge, TAILPROBE is the first benchmark that scores TTS pronunciation letter by letter, across released models and scripts, against human speech. It covers 29 languages in 16 scripts and 1,259,561 scored human tokens (§3).

(2) A large audit. We test 7 released models from five architecture families on 126,194 clips. Letters without a row are silently dropped, and the first consonant of an utterance is weaker in all 7 models under the main recogniser (six of seven under the second; §4).

(3) A causal test of rarity. Pre-registered count analyses and controlled finetuning in two architectures show that a new letter is learned well only when its row gets a strong training signal. Then 160 examples close 95% of the gap in VITS and

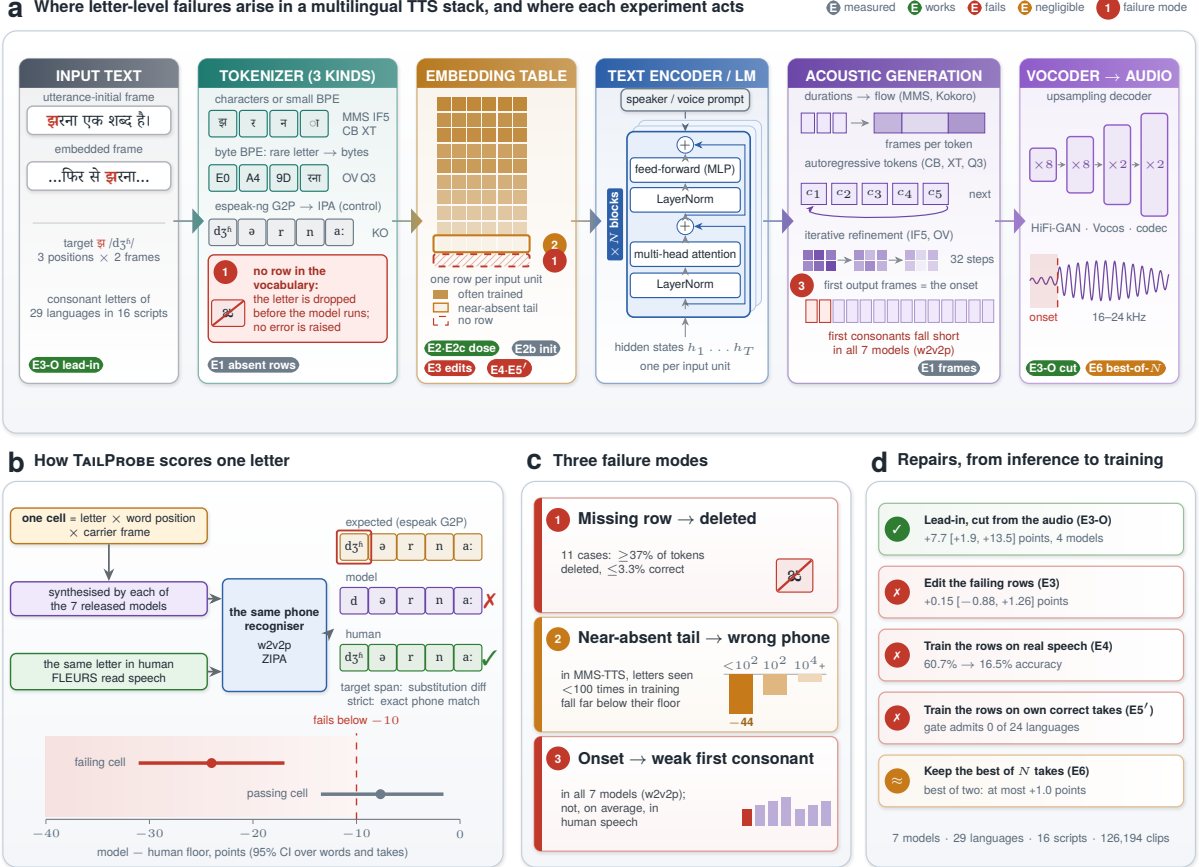


Figure 1: **Where letter failures arise, and how we test them.** (a) A multilingual TTS model: each unit from the tokenizer looks up one input row (shading: how often the row was trained), and an encoder, an acoustic model and a vocoder turn the rows into audio. Red badges mark the three problems we find: ① a letter with no row, ② a letter almost absent from training, ③ the first sound of an utterance. Pills show where each experiment acts and how it turned out. (b) Model and human speech go through the same phone recogniser; a cell fails when the model is clearly more than 10 points below humans. (c) The three problems and their size. (d) The repairs and their results. Model codes: MMS-TTS (MMS), IndicF5 (IF5), Chatterbox (CB), XTTS-v2 (XT), OmniVoice (OV), Qwen3-TTS (Q3), Kokoro (KO).

62% in F5-TTS. In released models, rarity hurts where training data are scarce (§5).

(4) Repairs at three levels. A training-free *lead-in-and-cut* repair improves the onset in all four models we tested, by 7.7 points. Editing the rows (E3) or training only the rows (E4, E5', both pre-registered) did not repair released models, and we explain why (§6).

2 Related Work

Under-trained tokens and new rows. In text LLMs, rare tokens keep near-initial embeddings and misbehave. Such tokens can be found from the weights (Land and Bartolo, 2024) and cluster in embedding space (Li et al., 2024); their effects can be reduced by correcting intermediate activations (Zhang et al., 2024) or by gated LoRA branches

(Fan et al., 2026). Rare-token embeddings collapse into a narrow cone (Gao et al., 2019), although a short norm alone does not prove under-training (Quemy, 2026). New rows can be initialised from static embeddings (Minixhofer et al., 2022), overlapping tokens (Dobler and de Melo, 2023), factorised vectors (Liu et al., 2024; Özeren et al., 2025), the convex hull of existing rows (Mundra et al., 2024) or distilled hidden states (Dobler et al., 2026). Joshi et al. (2026) compare more than twenty strategies for Hindi, and transliteration can exploit related scripts (Khemchandani et al., 2021). TTS adaptations add letter rows in the same way. IndicF5 copies randomly chosen existing rows (Varadhan et al., 2025), community adapters copy Devanagari rows (Kommanamanchi, 2026), and F5-TTS has been extended with trainable new-character rows (Chivereanu and Boros, 2025). Unseen phonemes

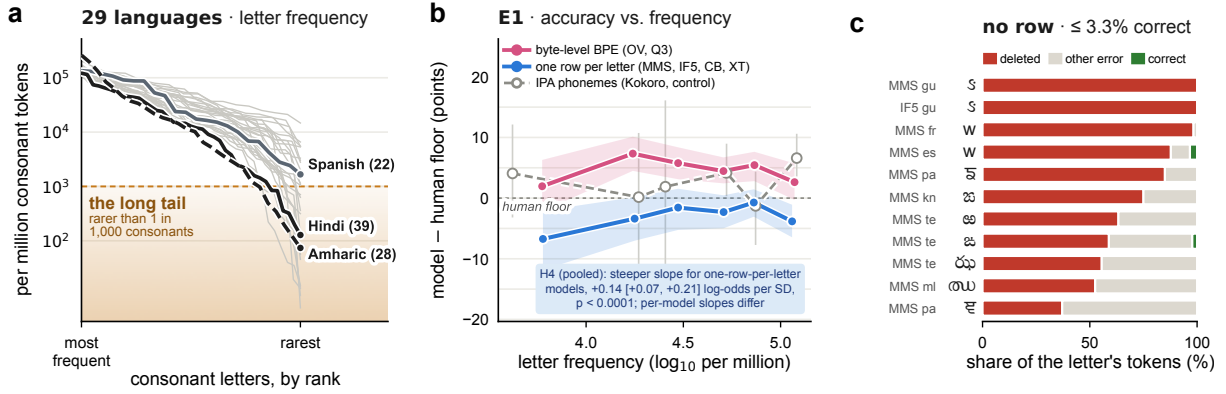


Figure 2: **The long tail of the alphabet.** (a) Written-text frequency of each probed consonant letter, by rank, in the 29 languages (three highlighted, with their numbers of letters); the shaded tail holds letters rarer than 1 in 1,000 consonants. (b) Model – human floor against letter frequency, by kind of input unit (95% intervals); the callout gives the pooled test of whether the slope differs by unit (H4). (c) The 11 letters without an input row in a model: share of their tokens heard as deleted, as another error, or as correct (at most 3.3%).

need as much data as training from scratch (Mura et al., 2026). None of this work checks, letter by letter, whether the rows of released models are usable.

Rare letters and TTS evaluation. Rare character pairs hurt Hindi TTS (Anand et al., 2024), and mispronunciations follow unbalanced word coverage (Taylor and Richmond, 2019). Multilingual embeddings may merge letters of different scripts (Rahman, 2026), and retroflex and aspiration errors in Indic TTS have been profiled as accent (Menta, 2026b). Letter embeddings steer pronunciation (Perquin et al., 2020), and pronunciation can be edited after training (Singh et al., 2026a; Lee et al., 2025; Singh et al., 2026b; Kato, 2026). Some designs avoid a letter table altogether and use bytes (Li et al., 2019; He et al., 2021), pixels (Arigala et al., 2026) or romanisation (Menta, 2026a). Error rates from one ASR system mix model errors with recogniser errors. Whisper normalisers distort Indic scripts (Manohar and Pillai, 2024), gains from choosing among takes depend on the judging ASR (Yu and Kang, 2026), and faithful measurement is called for (Yang et al., 2026). We instead test every letter against a human floor scored by the same phone recogniser (Xu et al., 2022), and test causes and repairs side by side.

3 Setup

TAILPROBE asks one question per *cell*: how often does a model pronounce consonant letter *g* correctly at word position *p* (initial, medial or final), in a word that opens the sentence (*init*) or sits inside

it (*mid*)? We say *letter* for a consonant grapheme. Appendix F.2 dates every scorer change made after looking at outputs, and the frequency claims are re-tested on held-out words under a plan fixed in advance (§5.1).

Probes. A letter is the unit a text-input model usually sees as one symbol: a code point, or a base letter with a nukta. We skip letters inside multi-letter graphemes such as Polish *sz*, and we sample the Ge’ez syllabary (Appendix B.2). For each cell we take the six most frequent words (Speer, 2022) that contain *g* exactly once, are written only in the language’s script, and have at least two grapheme clusters (The Unicode Consortium, 2026). The first three are *screening* words; the other three are held-out *reserve* words. Each word is spoken in two carrier sentences. The words next to it end and start with vowels, so no other consonant can merge with the target (58 frames, plus two English frames used only in E2; Appendix B.3). To find the target sound, we run the espeak-ng 1.52 G2P (eSpeak NG contributors, 2024) on the sentence and on two copies with *g* replaced by another consonant. The phones that change in both copies are the target span. We drop and log probes without a clean span.

Models and languages. We test seven released models that differ in what an input row is (Appendix Table 3; Appendix B.4 draws each one). Each runs only in languages it officially supports, with its released decoding settings. MMS-TTS (Pratap et al., 2024) is a per-language VITS model (Kim et al., 2021); its tokenizer silently deletes characters outside its vocabulary. IndicF5 (Varadhan et al., 2025)

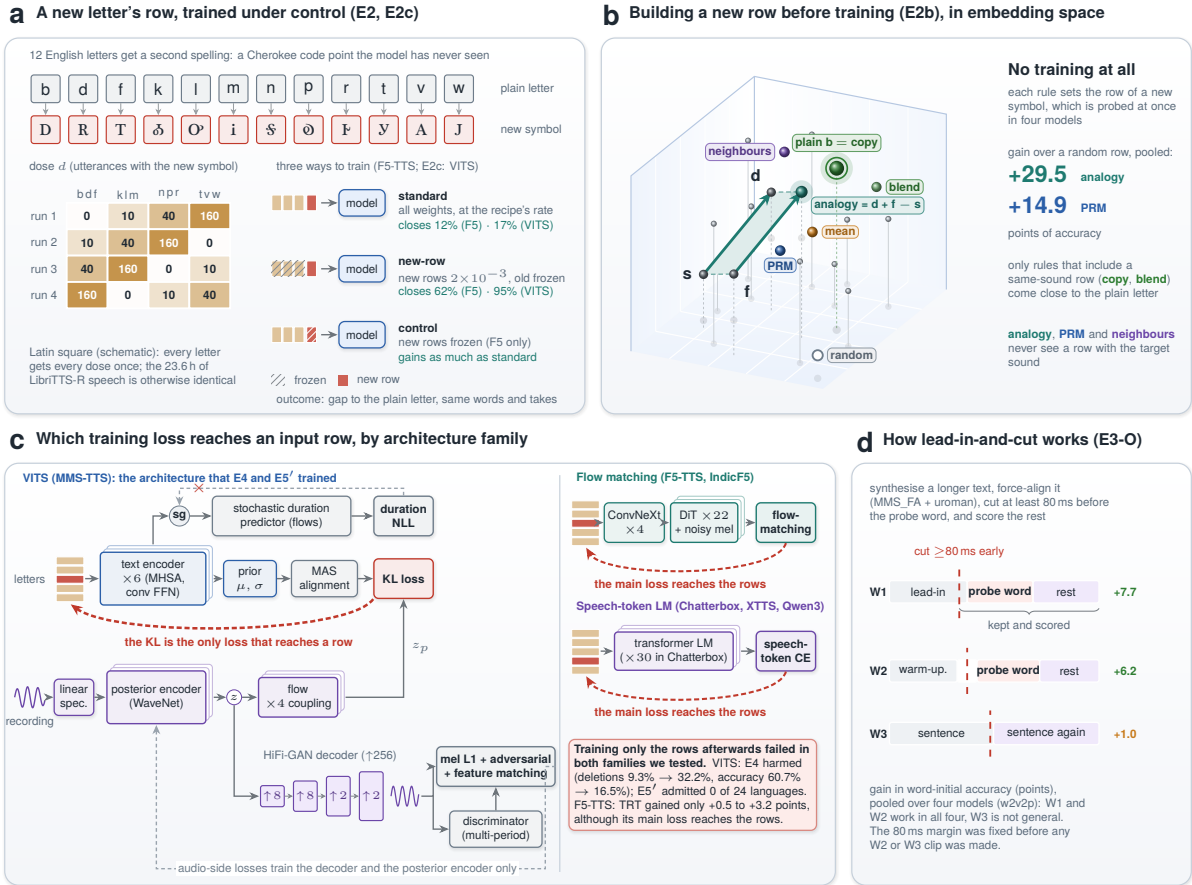


Figure 3: **How the key experiments work.** (a) E2/E2c: new spellings for twelve letters, a Latin square of doses and three arms. (b) E2b: seven starting rules in a sketch of embedding space (illustrative). (c) Which loss reaches an input row: only the KL in VITS, the main loss in other families. (d) Lead-in-and-cut and the other remedies.

is an F5-TTS model (Chen et al., 2025) finetuned on Indian languages, with one row per code point (2,545 entries). Chatterbox multilingual (Resemble AI, 2025) has only 265 BPE merges, all English or IPA, so non-Latin text gets one row per code point. XTTS-v2 (Casanova et al., 2024) uses a 6,681-token BPE that contains every character. OmniVoice (Zhu et al., 2026) and Qwen3-TTS (Hu et al., 2026) share the Qwen3 byte-level BPE. Kokoro-82M (hexgrad, 2025), a StyleTTS2 model (Li et al., 2023), reads espeak-ng IPA instead of letters, so it has no letter rows and serves as a control. Voice-cloning models get one native-speaker FLEURS recording as the voice prompt, which we leave out of the floors; MMS-TTS and Kokoro use their built-in voices. We generate five takes per item for MMS-TTS and Kokoro and two for the other models. The 29 languages cover 16 scripts: nine languages use Brahmic scripts, four Arabic or Hebrew script, five other non-Latin alphabets and ten Latin script, and Amharic uses the Ge'ez syllabary (Appendix B.2).

Scoring. Our main ruler is a multilingual phone recogniser, wav2vec2-xlsr-53-espeak-cv-ft (w2v2p; Xu et al., 2022), which outputs the same espeak IPA symbols as the G2P. A normaliser removes stress, tone and length marks, joins split affricates, and treats three pairs that the recogniser does not separate as equal (Appendix B.5). Every clip, model or human, gets 250 ms of silence at both ends. Without this padding the recogniser misses the first phone of model audio, because synthesisers often start speaking at the very first sample. We align expected and recognised phones with Needleman–Wunsch, using PanPhon feature distances (Mortensen et al., 2016). A token is *correct* when every consonant of the target span is recognised exactly.

Human floors. No automatic ruler is perfect, so “the model is wrong” must mean “worse than a human under the same ruler”. We therefore score every consonant in the FLEURS (Conneau et al., 2023) test and dev recordings with exactly the same

pipeline. The *floor* of letter g at position p is the human accuracy on these consonants (pooled over positions when fewer than 20 occur). A cell *fails* when the whole 95% interval of model – floor lies below -10 points; intervals come from a cluster bootstrap over words and takes.

Checks and statistics. A second phone recogniser with a larger inventory, ZIPA-CR-NS-Large (Zhu et al., 2025), scores every clip with its own floors. Whisper large-v3 (Radford et al., 2023) is used only to choose repairs, so that choosing and scoring use different instruments (Yu and Kang, 2026). Planted errors with known answers are falsely accepted in 18.8% of cases; aspiration is the weak spot, so no claim rests on it. FLEURS-like noise lowers MMS-TTS accuracy by only 0.3 and 2.3 points on average at $+30$ and $+20$ dB. Only Arabic drops by more than 10 points at $+20$ dB (Appendix B.6). For frequency, we fit a logistic mixed model per model, with the floor as an offset (Bates et al., 2015); Appendix D.1 gives the details.

4 How Common Is It?

Human floors are far from 100%. The audit scores 126,194 clips over 97 model–language pairs and 13,596 cells, with floors from 1,259,561 FLEURS consonant tokens (Appendix C.1). Averaged over the probed cells of a language, the human floor is only 40–94% (median 58%). A raw accuracy of 50% can therefore be perfect for one letter and a failure for another, so every result below is relative to the letter’s own floor.

Letters without a row are silently dropped. In 11 cases a model has no input row for a letter: MMS-TTS for Telugu ఌ య ఱ (ña, jha, ra), Punjabi ਌ ਊ (ña, ña), Kannada ಌ (ña), Malayalam ൐ (jha), Gujarati ળ (ña) and the Spanish and French w , and IndicF5 for Gujarati ળ (Figure 2c). The tokenizer drops such a letter, so the model never receives it. At most 3.3% of its tokens are correct, at least 37% are heard as deleted, and nothing reports the problem.

Every model has failing letters. 644 of the 13,596 cells fail. That is 2.6–8.8% of a model’s cells, most for MMS-TTS, which was trained on one book per language (Table 1).

The start of an utterance is weaker in (almost) every model. The first consonant of a word is less accurate when the word opens the utterance than

Model	Langs	Clips	Acc.	Floor	Fail. %	Fail. % _{hc}
MMS-TTS	24	55,470	55.0	61.5	8.8	9.1
IndicF5	9	10,444	50.5	48.6	2.6	0.6
Chatterbox-MTL	13	10,228	70.4	72.2	4.3	3.1
XTTS-v2	11	8,968	70.1	73.2	4.5	4.6
OmniVoice	29	26,644	65.5	61.9	2.8	1.1
Qwen3-TTS	6	4,220	84.7	85.2	3.4	3.6
Kokoro-82M	5	10,220	77.1	74.5	3.8	6.0

Table 1: Results per model (strict). Acc. and Floor: mean accuracy of the model and of human speech over cells (%); Fail. %: share of failing cells; *hc*: human-calibrated criterion. Per language: Figure 17.

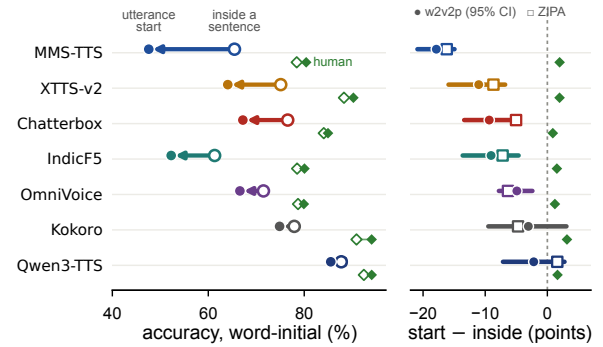


Figure 4: **Utterance start vs. inside a sentence.** Left: word-initial accuracy inside a sentence (hollow) and at the start (filled), per model (w2v2p); green: human speech. Right: the difference with its 95% CI (w2v2p) and the ZIPA estimate (squares).

when the same word follows other words. Under w2v2p this holds in all 7 models, by 2.2–17.9 points (Figure 4), although for two models the interval includes zero. Under ZIPA, Qwen3-TTS shows no drop. In the mixed model, the effect is significant for 7 of 7 models under w2v2p and 6 of 7 under ZIPA. Human speech shows, on average, the opposite: in FLEURS, utterance-initial consonants score $+0.9$ to $+3.2$ points higher (Appendix Table 11). The weakness therefore comes from synthesis; we repair it in §6.1. *Letters without a row are dropped silently, every model has failing letters, and every model is weaker at the start of an utterance than inside it under the main recogniser.*

5 What Causes It?

Three causes could explain these failures: rare training of a letter’s row, no row at all (§4), or something unrelated to letters, such as the onset (§6). We test rare training in four ways.

5.1 Rarity matters where training data are scarce

Written-text frequency gives no consistent picture. Per model, the frequency slope is significant only

for MMS-TTS under ZIPA. Pooled over models, it is slightly negative under w2v2p (-0.16 [-0.31 , -0.01]) and near zero under ZIPA ($+0.03$ [-0.13 , $+0.19$]; Appendix Table 13). A pooled test finds a steeper slope for models with one row per letter than for byte-BPE models (H4; Figure 2b). But the per-model slopes do not show this clearly, and Kokoro, which has no letter rows, has the largest slope (not significant).

We therefore count letters in the training text itself, under pre-registered analyses (E1-T, E1-T2). MMS-TTS was trained on one New Testament per language; XTTS-v2 and OmniVoice were trained largely on public corpora with transcripts. For MMS-TTS, accuracy rises with the training count under ZIPA (0.56 log-odds per decade, $p < 0.001$), but not significantly under w2v2p (0.20, $p = 0.089$; Figure 5). The 6 letters seen fewer than 100 times, mostly loan letters, fall 44 points below their floors (w2v2p: 37). One of them has a row but never occurs in the training text. The shortfall shrinks to 23 points at 100–999 occurrences and to 8 above 10^4 (w2v2p: 13 and 3), with no change of slope at 100. XTTS-v2 and OmniVoice, which saw almost every letter more than 1,000 times, show no association (Appendix Table 18). On held-out reserve words, none of the four pre-registered replication tests survives Holm correction (MMS-TTS: 0.12 under w2v2p, 0.21 under ZIPA), although two pre-specified variants are significant. Of two letters with the same sound, the more frequent one is not pronounced better (Appendix Table 19). ***Rarity costs accuracy where training data are scarce: in MMS-TTS the rarest letters fail most and accuracy rises with training count under ZIPA; in the larger-data models we could count, there is no such effect.***

5.2 What the released rows look like

For every letter with exactly one input row, we compute four indicators from the weights: the row’s norm, its similarity to its five nearest rows, its similarity to the model’s special rows (Land and Bartolo, 2024), and how well its neighbours are organised by sound. In 5 of the 5 letter-row models, at least one indicator correlates with frequency ($|\rho|$ up to 0.58). Correlations with model – floor, however, are small ($|\rho| \leq 0.17$) and change sign between models. The clearest separation of failing from working letters comes from sound organisation, in 4 of 5 models (exploratory; Appendix D.3). The rows thus record how often they were trained but predict failure only

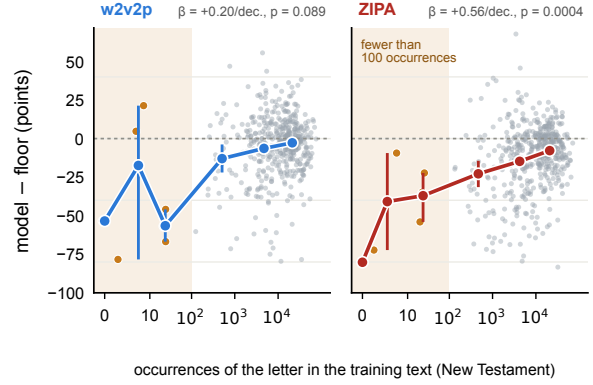


Figure 5: **Training-text counts and MMS-TTS accuracy (E1-T).** Each dot is one letter: model – floor against its count in the training text (amber: fewer than 100); large markers: pre-registered count groups (95% CIs).

weakly, so we test causation directly.

5.3 A new letter needs a trained row (E2, E2c)

In released models, frequency is mixed with everything else, so E2 sets it directly (Figure 3a). Twelve English letters each get a second spelling: a new symbol (a Cherokee code point) for the same sound. Each new symbol appears in exactly $d \in \{0, 10, 40, 160\}$ utterances of a fixed 23.6 h LibriTTS-R set (Koizumi et al., 2023). A Latin square over four runs keeps everything else identical. We finetune F5-TTS v1 Base (Chen et al., 2025) in three arms. The *standard* arm trains every weight at the recipe’s rate. The *new-row* arm gives the new rows their own higher rate and freezes the old rows. A control arm freezes the new rows. The outcome is the paired gap between the new symbol and the plain letter. E2c repeats the design in MMS-TTS English (VITS) on LJSpeech (Appendix D.3).

With its own learning rate, a new symbol learns from its occurrences (Figure 6). At $d = 160$ it closes 62% of the gap in F5-TTS and 95% in VITS. In VITS the gap shrinks steadily (Page’s trend test $p = 1 \times 10^{-7}$), from -56.2 points at $d = 0$ to -2.8 , which is no longer distinguishable from zero. Under the standard recipe, the same 160 occurrences close only 12% and 17%. In F5-TTS, the control with frozen new rows gains just as much, so these small gains come from the rest of the network, not from the row. ***A new letter is learned well only when its row gets a strong training signal; then every extra occurrence helps.***

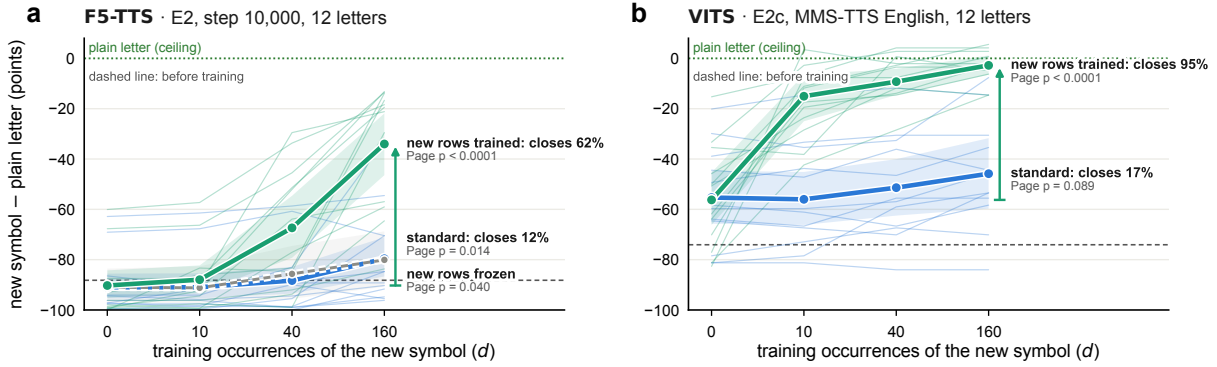


Figure 6: **Controlled dose response in two architectures (E2, E2c).** New symbol – plain letter (w2v2p; same words and takes) by training occurrences d . Thin lines: the twelve letters; bands: 95% CIs of the arm means; arrows: the gap closed at $d = 160$. Page: ordered trend test.

5.4 Building a row before training (E2b)

Before training, its starting row is all a new symbol has. E2b builds the rows of the twelve new symbols with seven rules and probes them right away, with no training, in four models: MMS-TTS, F5-TTS, Chatterbox and XTTS-v2 (Figure 3b). Phonological *analogy* works like word analogies: it builds the row of a new letter x as $r(y) + r(z) - r(w)$, from three letters whose sound features satisfy $f(y) + f(z) - f(w) \approx f(x)$. Pooled over the four models, analogy beats a random row by 29.5 points, and Phonological Row Mapping (PRM, §6.2) beats it by 14.9. Only rules that include the row of a letter with the same sound come close to the plain letter (Appendix Figure 24).

6 Can It Be Repaired?

We try repairs from the least to the most invasive, choosing each on one set of words and evaluating it on another, with a different ruler where possible.

6.1 The onset can be repaired without training

The onset weakness comes from how synthesis starts, so we repair it at inference time, without changing the model. We pre-registered three remedies (E3-O; Figure 3d) and tested them in four models (MMS-TTS, XTTS-v2, Chatterbox-MTL, OmniVoice) on six languages. **W1**, *lead-in-and-cut*, has the model speak a short phrase (the language’s mid-sentence carrier prefix) before the sentence. **W2** speaks the same phrase as a separate warm-up sentence, and **W3** speaks the sentence twice. We find the probe word by forced alignment (MMS_FA, Pratap et al., 2024; uroman, Hermjakob et al., 2018) and cut the clip at least 80 ms before it (10 ms fade-in), then score it against the original

expected phones, paired with the original clip. The 80 ms margin was chosen after the first W1 outputs and then fixed in the pre-registration, before any W2 or W3 clip was made (Appendix F.2).

W1 raises word-initial accuracy in all four models under both rulers, significantly after Holm correction in 3 models under w2v2p and 4 under ZIPA. The pooled gain is +7.7 [+1.9, +13.5] points (ZIPA: +6.9 [+2.6, +11.2]). W2 gains +6.2 [+1.1, +11.2]; for MMS-TTS it equals W1, as its vocabulary drops the sentence mark. The gain is largest for MMS-TTS (17.1 points) and smallest for OmniVoice (not significant under w2v2p), the models with the largest and the smallest onset weakness. W3 does not work in general (+1.0 [−8.4, +10.4] points; Appendix Figure 27 and Table 35). *Lead-in-and-cut raises the utterance onset in all four models we tested, without changing the model.*

6.2 Editing the rows does not help (E3)

For each failing letter, we try row edits that need no training. We copy or blend the rows of its five nearest consonants by sound, rewrite it as its nearest letter, or use *Phonological Row Mapping* (PRM). PRM learns a mapping from sound features to rows on the language’s working letters (kernel ridge regression) and predicts the failing letter’s row. A gate applies an edit only if it beats the released row on reserve words under Whisper; we then evaluate it on the screening words with the phone ruler. Over 208 failing letters in 4 models (at most 12 per language, in the languages of Appendix Table 41), the gate kept the released row for 148. Accuracy changed by +0.15 [−0.88, +1.26] points, and untouched letters did not change (+0.2 points on average). A new row for a letter without one lets it through the tokenizer but does not make it correct (0 of the 10 such MMS-

TTS letters reach 50%). Training only the ~~un~~trained rows of the E2 runs on 10 and 40 held-back utterances, with every other weight frozen (Targeted Row Tuning, TRT), changes accuracy by only +0.5 to +3.2 points.

6.3 Training the rows does not help (E4, E5')

In a pre-registered test (E4), we trained only the input rows of 79 failing MMS-TTS letters in 8 languages, on 10 or 40 FLEURS-train utterances per letter. Reserve-word accuracy fell from 60.7% to 16.5%, and 77 of the 79 letters got worse (ZIPA: 58.9% $\rightarrow$ 15.6%). An exploratory analysis explains why (Figure 3c; Appendix E.3). In VITS training, only the KL term (text prior vs. audio posterior) reaches an input row. It pulled the rows towards the voices of FLEURS speakers, whom the single-speaker model had never heard. The updates of a language's letters share one direction (cosine with the language's mean update: 0.33–0.61). Nothing held the rows in place, and the frozen duration predictor misread the moved rows, so deletions rose from 9.3% to 32.2%.

A second pre-registered test (E5') removed these causes. In all 24 MMS-TTS languages that met E4's qualification rule (168 failing letters), it trained only the failing rows, on the model's own correct takes, with the duration path open, an anchor on the other tokens and a trust region. A gate on dev words rejected the trained rows in every language and in both replicates (0 and 0 of 24 admitted; target change -2.6 and -2.5 points). So the gated system is the released model, with zero test-word gain by construction (Appendix E.4). *None of the three ways we tried to repair the rows of a released model (editing them, training them on real speech, training them on its own correct takes) improved its failing letters.*

6.4 Choosing the better take helps little (E6)

Another repair changes no weights: generate several takes and keep the one whose recognised phones best match the text. Under a plan fixed before any outcome, we choose the better of two takes with ZIPA and grade it with w2v2p. This gains only +0.0 to +1.0 points over a random take (+1.4 for MMS-TTS with five takes). How well a whole utterance matches its text says little about one letter (Appendix E.5).

7 Discussion

Three problems, three causes. A letter without a row is lost before the model runs. A letter that is rare in a small training set is learned poorly; models that saw almost every letter thousands of times (XTTS-v2, OmniVoice) show no frequency effect. The onset weakness is different: it appears in all seven models under the main recogniser but not, on average, in human speech, so it comes from how synthesis starts, not from a row.

Why the row repairs fail. In VITS, the only loss that reaches a row pulls it towards new voices (E4); E5' removed that cause and still did not help. In E2 and E2c, by contrast, a new row learns well when it gets its own learning rate while the rest of the model trains, but gains little when only the rows are trained afterwards (TRT). A row seems to be learned best while the network that reads it is still training.

Advice for practitioners. (1) Check every letter of the target script, and its normalised forms, against the tokenizer before training. (2) Report letter-level accuracy against a human floor. (3) Treat the onset as a weak spot, and use lead-in-and-cut. (4) Give new rows a phonological start and a strong training signal. (5) Do not expect row edits or row-only training to fix a finished model (Appendix E.7).

8 Conclusion

We designed TAILPROBE to test every consonant letter of a script against a human floor, and tested 7 released TTS models in 29 languages and 16 scripts. Letters without an input row are dropped silently, rare letters in a small training set are pronounced worse, and the onset is weaker than in human speech. A new letter needs a strong row signal. Lead-in-and-cut repairs the onset without training, but changing only a released model's rows did not help. Letter failures are best prevented before and during training.

Limitations

The ruler. Our main ruler is one phone recogniser whose labels come from espeak-ng. It cannot output some phones (e.g. ejectives) and often misses others (aspiration, retroflexion). Three things reduce this problem but do not remove it: the per-letter human floor, the relaxed criteria, and a second phone recogniser with a larger inventory (ZIPA, which saw FLEURS speech in training and under-marks

aspiration). A letter whose floor is near zero cannot be judged. A native-listener test for Bengali and Hindi is prepared but not yet run, so no claim in this paper rests on listeners.

Training details in E2. F5-TTS trains with AdamW (weight decay 0.01). So in the standard arm, a row that receives no gradient still shrinks slightly (at most about 0.8% in norm over 10,251 updates); this is the same at every dose. Adam’s momentum can also move a rarely seen row on steps where its symbol is absent, so “occurrences” are not the same as “updates”.

The G2P reference. Expected phones come from espeak-ng 1.52, whose rules are imperfect for some languages (we exclude Thai for this reason). Finding spans by substitution removes probes with an ambiguous target, which favours words with transparent spelling.

Frequency proxies. The training data of most released models are unknown. We therefore use written-text frequency as a proxy, and training-text counts only where a proxy of the training text is public; only E2 and E2c set training frequency directly. Rows shared by several languages (Latin letters, Devanagari in Hindi and Marathi) are trained by all of them, which a per-language count ignores.

Frames and words. The carrier frames were written for this study and have not yet been reviewed by native speakers. Probe words are real, frequent words. This keeps the probes natural but may let memorised words hide letter-level problems.

Scope. The slower models get two takes per item. Each model is tested only in languages it officially supports, so some languages are covered by only some models. The onset repair was tested in four of the seven models (six languages in the pre-registered test), and the controlled finetuning in two architectures.

Pre-registration and unfinished experiments. Two analysis plans were amended three times, always before any outcome data were examined. The E5’ plan gained a dose arm and an explicit rule for the title, and E7’s letter selection was corrected after it selected no letters (all three amendments, with dates and hashes, are in Appendix F). E5’’s gate admitted no language, so its gated test-word result is fixed by construction; for budget reasons its secondary test-word analyses were not run. E7,

a pre-registered test that resets and retrains the rows of existing letters, was stopped after letter selection. Both are reported as incomplete.

Ethics Statement

All human speech we use is openly licensed: FLEURS (CC BY 4.0) for the floors, the voice prompts and E4, LibriTTS-R (CC BY 4.0) for E2, and LJSpeech (public domain) for E2c; E5’ trains on the released model’s own synthetic takes. No commercial TTS output is used anywhere in training or evaluation, in line with the terms of commercial providers, several of which forbid using their output to train or test other models. Released models are used under their licences for non-commercial research evaluation; XTTS-v2’s CPML licence excludes some funded uses, which authors should check against their own funding. Voice prompts are single FLEURS recordings used only to condition cloning models for evaluation, and we release synthetic audio in a speaker’s voice only as far as needed to reproduce the scores. Evaluation texts are our own carrier sentences filled with frequent words and are checked for overlap with every training manifest used in E2. The benchmark is meant to make multilingual TTS work better for speakers of under-served scripts; silent deletion of letters without an input row, one of the problems we document, affects those speakers first.

Use of AI assistants. We disclose the role of AI assistants, as required. An LLM coding assistant was used substantially throughout: to implement and debug the probe builder, scorers, synthesis and training code and the GPU tooling, to run analyses, and to draft and revise parts of this manuscript. Every result was produced by released code that a reader can re-run, and every number in the paper is generated from the saved outputs by released scripts rather than typed by hand. The authors checked the claims against those outputs and take full responsibility for the content.

References

- Srija Anand, Praveen Srinivasa Varadhan, Mehak Singal, and Mitesh M. Khapra. 2024. [ELAICHI: Enhancing low-resource TTS by addressing infrequent and low-frequency character bigrams](#). *Preprint*, arXiv:2410.17901.
- Adarsh Arigala, Arjun Gangwar, S. Umesh, and Yova Kementchedjieva. 2026. [Pixel-TTS: Image based](#)

text rendering for robust text-to-speech. *Preprint*, arXiv:2606.14750.

Douglas Bates, Martin Mächler, Ben Bolker, and Steve Walker. 2015. **Fitting linear mixed-effects models using lme4**. *Journal of Statistical Software*, 67(1).

Edresson Casanova, Kelly Davis, Eren Gölge, Görkem Göknar, Iulian Gulea, Logan Hart, Aya Aljafari, Joshua Meyer, Reuben Morais, Samuel Olayemi, and Julian Weber. 2024. **XTTS: a massively multilingual zero-shot text-to-speech model**. In *Interspeech 2024*, pages 4978–4982.

Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, Jian Zhao, Kai Yu, and Xie Chen. 2025. **F5-TTS: A fairytale 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, Vienna, Austria. Association for Computational Linguistics.

Radu-Gabriel Chivoreanu and Tiberiu Boros. 2025. **F5-TTS-RO: Extending F5-TTS to Romanian TTS via lightweight input adaptation**. In *Proceedings of the 20th International Conference on Linguistic Resources and Tools for Natural Language Processing*.

Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna. 2023. **FLEURS: Few-shot learning evaluation of universal representations of speech**. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 798–805.

Konstantin Dobler and Gerard de Melo. 2023. **FOCUS: Effective embedding initialization for monolingual specialization of multilingual models**. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13440–13454, Singapore. Association for Computational Linguistics.

Konstantin Dobler, Desmond Elliott, and Gerard de Melo. 2026. **Token distillation: Attention-aware input embeddings for new tokens**. In *International Conference on Learning Representations (ICLR)*.

eSpeak NG contributors. 2024. eSpeak NG: an open source speech synthesizer supporting more than a hundred languages and accents. <https://github.com/espeak-ng/espeak-ng>. Version 1.52.0 (released 2024-12-12).

Yibo Fan, Jingru Li, and Huan Li. 2026. **GlitchCleaner: Lightweight glitch tokens repairing by lossless gated LoRA in large language models**. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 30656–30664. Issue 36.

Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tie-Yan Liu. 2019. **Representation degeneration problem in training natural language generation models**. In *International Conference on Learning Representations (ICLR)*.

Mutian He, Jingzhou Yang, Lei He, and Frank K. Soong. 2021. **Multilingual Byte2Speech models for scalable low-resource speech synthesis**. *Preprint*, arXiv:2103.03541.

Ulf Hermjakob, Jonathan May, and Kevin Knight. 2018. **Out-of-the-box universal Romanization tool uroman**. In *Proceedings of ACL 2018, System Demonstrations*, pages 13–18, Melbourne, Australia. Association for Computational Linguistics.

hexgrad. 2025. Kokoro-82M. Hugging Face model repository <https://huggingface.co/hexgrad/Kokoro-82M>. Apache-2.0; StyleTTS 2-based, 82M parameters; v1.0 released 2025-01-27. Code: <https://github.com/hexgrad/kokoro>.

Hangrui Hu, Xinfu Zhu, Ting He, Dake Guo, Bin Zhang, Xiong Wang, Zhifang Guo, Ziyue Jiang, Hongkun Hao, Zishan Guo, Xinyu Zhang, Pei Zhang, Baosong Yang, Jin Xu, Jingren Zhou, and Junyang Lin. 2026. **Qwen3-TTS technical report**. *arXiv preprint arXiv:2601.15621*.

Raviraj Joshi, Utkarsh Vaidya, Sanjay Singh Chauhan, and Niranjan Wartikar. 2026. **Beyond initialization loss: A systematic study of token embedding initialization strategies for LLM vocabulary extension**. *Preprint*, arXiv:2608.03494.

Shuhe Kato. 2026. **Self-distilled pronunciation and accent control for neural text-to-speech**. *Preprint*, arXiv:2609.17234.

Yash Khemchandani, Sarvesh Mehtani, Vaidehi Patil, Abhijeet Awasthi, Partha Talukdar, and Sunita Sarawagi. 2021. **Exploiting language relatedness for low web-resource language model adaptation: An Indic languages study**. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1312–1323, Online. Association for Computational Linguistics.

Jaehyeon Kim, Jungil Kong, and Juhee Son. 2021. **Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech**. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 5530–5540. PMLR.

Yuma Koizumi, Heiga Zen, Shigeki Karita, Yifan Ding, Kohei Yatabe, Nobuyuki Morioka, Michiel Bacchi-ani, Yu Zhang, Wei Han, and Ankur Bapna. 2023. **LibriTTS-R: A restored multi-speaker text-to-speech corpus**. In *Interspeech*.

Bharadwaj Kommanamanchi. 2026. **Chatterbox indic LoRA: Indian language TTS via grapheme-level fine-tuning**. Hugging Face model repository <https://huggingface.co/reenigne314/chatterbox-indic-lora>. LoRA adapters and extended tokenizer for Chatterbox-Multilingual. Accessed 2026-09-30.

- Resemble AI. 2025. Chatterbox-TTS. <https://github.com/resemble-ai/chatterbox>; GitHub repository; models at <https://huggingface.co/ResembleAI/chatterbox>. 921
- Ayush Pratap Singh, Harshit Singh, Nityanand Mathur, Akshat Mandloi, and Sudarshan Kamath. 2026a. *So-noEdit: Null-space constrained knowledge editing for pronunciation correction in LLM-based TTS*. *Preprint*, arXiv:2601.17086. 922
- Harshit Singh, Ayush Pratap Singh, and Nityanand Mathur. 2026b. *FlowEdit: Associative memory for lifelong pronunciation adaptation in flow-matching TTS*. *Preprint*, arXiv:2606.20518.
- Robyn Speer. 2022. *rspeer/wordfreq: v3.0*.
- Jason Taylor and Korin Richmond. 2019. *Analysis of pronunciation learning in end-to-end speech synthesis*. In *Interspeech 2019*, pages 2070–2074.
- The Unicode Consortium. 2026. Unicode standard annex #29: Unicode text segmentation. <https://www.unicode.org/reports/tr29/tr29-49.html>. Revision 49, Unicode 18.0.0.
- Praveen Srinivasa Varadhan, Srija Anand, Soma Siddhartha, and Mitesh M. Khapra. 2025. *Phir hera fairy: An English fairytale is a strong faker of fluent speech in low-resource Indian languages*. *Preprint*, arXiv:2505.20693.
- Qiantong Xu, Alexei Baevski, and Michael Auli. 2022. *Simple and effective zero-shot cross-lingual phoneme recognition*. In *Interspeech 2022*, pages 2113–2117.
- Yifan Yang, Hui Wang, Bing Han, Shujie Liu, Jinyu Li, Yong Qin, and Xie Chen. 2026. *Position: Towards responsible evaluation for text-to-speech*. In *Proceedings of the 43rd International Conference on Machine Learning (ICML), Position Paper Track*.
- Zhen Ye, Xinfa Zhu, Chi-Min Chan, Xinsheng Wang, Xu Tan, Jiahe Lei, Yi Peng, Haohe Liu, Yizhu Jin, Zheqi Dai, Hongzhan Lin, Jianyi Chen, Xingjian Du, Liumeng Xue, Yunlin Chen, Zhifei Li, Lei Xie, Qiuqiang Kong, Yike Guo, and Wei Xue. 2025. *Llasa: Scaling train-time and inference-time compute for Llama-based speech synthesis*. *Preprint*, arXiv:2502.04128.
- Taehyung Yu and Seongjae Kang. 2026. *Best-of- N TTS evaluation is confounded by ASR family alignment*. *Preprint*, arXiv:2607.08256. ICML 2026 Workshop on Machine Learning for Audio.
- Zhibo Zhang, Wuxia Bai, Yuxi Li, Mark Huasong Meng, Kailong Wang, Ling Shi, Li Li, Jun Wang, and Haoyu Wang. 2024. *GlitchProber: Advancing effective detection and mitigation of glitch tokens in large language models*. In *Proceedings of the 39th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pages 643–655.
- Han Zhu, Lingxuan Ye, Wei Kang, Zengwei Yao, Liyong Guo, Fangjun Kuang, Zhifeng Han, Weiji Zhuang, Long Lin, and Daniel Povey. 2026. *OmniVoice: Towards omnilingual zero-shot text-to-speech with diffusion language models*. *Preprint*, arXiv:2604.00688. 923
- Jian Zhu, Farhan Samir, Eleanor Chodroff, and David R. Mortensen. 2025. *ZIPA: A family of efficient models for multilingual phone recognition*. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 19568–19585, Vienna, Austria. Association for Computational Linguistics. 924
925
926
927
928
929

This appendix has six chapters. Chapter [A](#) explains how to read it. Chapters [B–E](#) follow the main text in order: the materials and protocol (§3), how common the failures are (§4), what causes them (§5) and whether they can be repaired (§6). Chapter [F](#) lists every pre-registration, amendment and analysis decision and shows how to reproduce every number. Each chapter opens with a summary and its own contents, and Chapters [B–E](#) close with a short recap.

A	How to Read This Appendix	14
A.1	How the chapters are organized.	14
A.2	Conventions	14
A.3	Provenance of every number	14
A.4	Where to find the evidence	14
A.5	Reading paths	14
A.6	Glossary	14
A.7	Model names.	16
B	Materials and Protocol	17
B.1	Models and infrastructure	17
B.2	Languages and probe sets	17
B.3	Carrier frames	17
B.4	Model architectures	17
B.5	Scorer details	17
B.6	Ruler validation	20
C	How Common Is It: Prevalence in Full	26
C.1	Per-language results	26
C.2	Further measurements in one place	26
C.3	E1 for every model and language	26
D	What Causes It: Frequency, Rows and Controlled Training in Full	30
D.1	Sensitivity of the frequency effect	30
D.2	Training-text counts and their replication	30
D.3	Row geometry and controlled finetuning in full	31
E	Can It Be Repaired: Repairs in Full	41
E.1	Onset repair: cut rules	41
E.2	Gated row edits (E3)	41
E.3	Why row-only training of released models harmed (E4)	41
E.4	Self-sample row repair and its gate (E5')	43
E.5	Best-of- N selection (E6)	43
E.6	Reset and relearn (E7, stopped)	45
E.7	What each intervention gains	46
F	Pre-registration, Changes and Reproduction	48
F.1	Pre-registrations and amendments	48
F.2	Changelog of analysis decisions	48
F.3	Reproducing this appendix	49

A How to Read This Appendix

How the appendix is organized, the conventions every table follows, where the evidence for each claim of the main text is, and the terms and model names used throughout.

In this chapter

A.1 How the chapters are organized.	14
A.2 Conventions	14
A.3 Provenance of every number	14
A.4 Where to find the evidence	14
A.5 Reading paths	14
A.6 Glossary	14
A.7 Model names.	16

A.1 How the chapters are organized

Chapter B documents the materials: the models and their architectures, the languages and probe sets, the carrier frames, the scorer and the checks that it measures pronunciation. Chapter C reports every prevalence result behind §4 for every model and language. Chapter D gives the analyses behind §5: the frequency models with all their sensitivity analyses, the training-text counts and their pre-registered tests, the geometry of the rows and the controlled-finetuning experiments. Chapter E treats the repairs of §6: the onset repair under every cut rule, the row edits, why row-only training harmed (E4), the self-sample repair and its gate (E5'), best-of- N selection (E6) and the stopped reset-and-relearn test (E7). Chapter F lists the pre-registrations, their amendments and every analysis decision taken after the first results, and the commands that regenerate the paper. Every table and figure is introduced by a paragraph that says what it contains and how to read it.

A.2 Conventions

Unless a table says otherwise, accuracy is the strict criterion (every consonant phone of the target span recognised exactly) under the primary phone ruler, w2v2p; ZIPA is the second phone ruler, and a ruler's scores are always compared with that ruler's own human floor. A *cell* is a grapheme $\times$ word position $\times$ carrier frame; it *fails* when the whole 95% interval of model – floor lies below -10 points. Differences are in percentage points. Intervals are 95% cluster-bootstrap intervals (words, then takes) unless stated; one-sided p -values are used for pre-registered directional hypotheses, Holm correction within each pre-registered family, and DerSimonian–Laird random effects when estimates are pooled over models. “Judgeable” cells have a

floor of at least 0.2 on at least 20 human tokens.

A.3 Provenance of every number

Every number in the paper is a macro written from the saved per-clip outputs by released scripts: `report.py` (cells, floors, GLMMs), `paper_tables.py` (main-text tables and macros), `appendix_tables.py` (the complete-results tables of Chapters B–E), `paper_figures.py` (figures) and `e4_rootcause.py` (the E4 diagnosis). No number is typed by hand; a result that is not yet measured appears as a red PENDING marker, and the paper is not submitted while one remains.

A.4 Where to find the evidence

Table 2 maps each claim of the main text to the section that argues it and to the appendix sections, tables and figures that hold its full evidence.

A.5 Reading paths

Readers with a specific question need not read the chapters in order. To check the headline prevalence numbers, read §C.1. To test whether a result is an artifact of the ruler, read §B.5 and §B.6. To judge the frequency claims, read Chapter D from §D.1, together with the pre-registrations in §F.1. To apply a repair, read §E.1; to see why the row repairs failed, read §E.3 and §E.4.

A.6 Glossary

The terms below recur throughout the appendix.

Unit, row The consonant grapheme a text-token model sees as one input symbol, and the learned embedding vector it looks up. An *absent row*: the model's vocabulary has no entry for the grapheme, so its tokenizer deletes it.

Cell, frame A grapheme $\times$ word position (initial, medial, final) $\times$ carrier frame (*init*: the probe

Claim	Argued in	Appendix	Tables and figures
A grapheme without an input row is deleted by the tokenizer, without warning.	§4	C.1	Tab. 12; Fig. 2c
Human floors are far from 100%, so every score is relative to the grapheme’s own floor.	§3, §4	B.2	Tab. 5; Fig. 16a
The ruler measures pronunciation: planted errors, a noise control, a second recogniser.	§3	B.6	Tabs. 8, 9; Fig. 16b–c
Every model has failing cells.	§4	C.1	Tab. 1; Fig. 17
Utterance-initial consonants fall short in every model under w2v2p (six of seven under ZIPA); on average not in human speech.	§4	C.1	Tabs. 11, 13; Fig. 4
A pooled contrast (H4) finds a steeper frequency slope for one-row-per-grapheme models; the per-model slopes do not show it clearly.	§5.1	D.1	Tabs. 13, 14; Fig. 2b
In MMS-TTS, accuracy rises with training count and the rarest letters fail most; no association in XTTS-v2 or OmniVoice; no primary E1-R test survives Holm.	§5.1	D.2	Tabs. 18, 22; Figs. 5, 22
Same-sound graphemes show no frequency advantage (null).	§5.1	D.2	Tabs. 19, 23; Fig. 23; Fig. 16e
The rows record how often they were trained but predict failure only weakly (exploratory).	§5.2	D.3	Tabs. 29, 30; Fig. 16d
A new letter is learned when its row gets a strong enough training signal, and then improves with its count, in two architectures.	§5.3, §5.3	D.3	Tabs. 24, 26; Figs. 6, 25
Before any training, a phonologically composed row beats a random row.	§5.4	D.3	Tabs. 27, 28; Figs. 24, 26
A cut lead-in improves utterance onsets in all four models tested (significantly in 3/4 under w2v2p, 4/4 under ZIPA); repetition does not.	§6.1	E.1	Tabs. 35, 37, 38; Figs. 27, 16f
Gated row edits (E3) and targeted row tuning do not help.	§6.2	E.2	Tabs. 41, 32; Fig. 30
Row-only training on real speech harms (E4), through the KL path.	§6.3	E.3	§E.3
Row training on the model’s own correct takes is rejected by its gate everywhere (E5’).	§6.3	E.4	Fig. 31
Best-of-two selection gains at most about a point (+1.4 for MMS-TTS with five takes; E6).	§6.4	E.5	Tab. 36
The seven models differ in what an input row is.	§3	B.4	Fig. 7; Figs. 8–14

Table 2: **Where to find the evidence.** Each row names a claim of the main text, the section where it is argued, and the appendix sections, tables and figures that give its full evidence.

1001	word opens the utterance; <i>mid</i> : it follows a	Screening, reserve, fresh words The first three	1016
1002	carrier phrase).	probe words per cell (E1), the next three (held	1017
		out for E1-R and used as dev words in E5’),	1018
1003	Floor The accuracy of the same ruler on the same	and further words built by the same rules for	1019
1004	grapheme and position in human FLEURS	E5’'s test.	1020
1005	speech.		
		H4 The frequency $\times$ tokenizer-type interaction	1021
1006	Strict, near, hc The scoring criteria: exact match	(one-row-per-grapheme versus byte-BPE mod-	1022
1007	of every target consonant; within one place or	els).	1023
1008	laryngeal feature; and the human-calibrated		
1009	criterion that accepts realisations seen in at	E1, E1-T, E1-T2, E1-R, E1-H Prevalence; the	1024
1010	least 10% of human tokens.	pre-registered training-text-count analyses	1025
		(MMS-TTS; XTTS-v2 and OmniVoice); their	1026
1011	w2v2p, ZIPA, Whisper The rulers: a wav2vec2	pre-registered replication on reserve words	1027
1012	phone recogniser with espeak IPA output (pri-	(tests R1–R4); the low-count same-sound test.	1028
1013	mary), the ZIPA-CR-NS-Large phone recog-		
1014	niser, and Whisper large-v3 word-level ASR	E2, E2c, E2b Controlled finetuning of new sym-	1029
1015	(used to choose E3 repairs).	bols in F5-TTS and in VITS, and new-symbol	1030

rows composed without training in four archi- 1031
tectures. 1032

E3, TRT, E3-O Gated edits of failing rows; tar- 1033
geted tuning of rows on withheld utterances; 1034
the pre-registered onset remedies (W1 lead-in, 1035
W2 warm-up sentence, W3 repeat). 1036

E4, E5', E6, E7 Row-only training on real speech; 1037
row training on the model's own correct takes 1038
behind a validation gate; best-of- N take selec- 1039
tion; reset-and-relearn of existing letters' rows 1040
(stopped). 1041

Gate, replicate In E5', the per-language rule on 1042
dev words that keeps the released model unless 1043
the trained rows help; replicate Z selects takes 1044
and gates with ZIPA and tests with w2v2p, 1045
replicate W the reverse. 1046

1047 **A.7 Model names**

Tables use short names and codes: MMS-TTS 1048
(MMS), IndicF5 (IF5), Chatterbox-MTL (CB), 1049
XTTS-v2 (XT), OmniVoice (OV), Qwen3-TTS 1050
(Q3) and Kokoro-82M (KO). F5-TTS is the base 1051
model of E2; MMS-TTS English is the base model 1052
of E2c. 1053

B Materials and Protocol

What was run and how it was scored: the models and their architectures, the languages and probe sets, the carrier frames, the scorer and the checks that it measures pronunciation. Read this chapter to check what each model was asked to say and how a clip was labelled.

In this chapter

B.1 Models and infrastructure	17
B.2 Languages and probe sets	17
B.3 Carrier frames	17
B.4 Model architectures	17
B.5 Scorer details	17
B.6 Ruler validation	20

Model	Input unit	Decoder	Langs
MMS-TTS	character	NAR (VITS)	24
IndicF5	code point	NAR (flow)	9
Chatterbox-MTL	code point	AR	13
XTTS-v2	small BPE	AR	11
OmniVoice	byte-level BPE	NAR (diffusion LM)	29
Qwen3-TTS	byte-level BPE	AR (multi-codebook)	6
Kokoro-82M	IPA phoneme	NAR	5

Table 3: The seven released models. Langs: languages tested, each officially supported by the model.

B.1 Models and infrastructure

Table 3 summarises the seven models of §3.

Table 4 lists every released model with its checkpoint, input unit, decoding parameters, voice and languages. All synthesis, scoring and training ran on rented single GPUs (RTX 3090, RTX A4500/A5000 and RTX 4090 class); every clip was generated with a recorded seed and the model’s released decoding defaults.

B.2 Languages and probe sets

Table 5 lists, per language, the consonant units in the inventory and with at least one qualifying probe word, the number of probe items (word $\times$ frame, screening and reserve), the source of the word list, the number of scored consonant tokens and the mean strict human floor in its FLEURS recordings, the counted training text of MMS-TTS and the models probed. Inventories and every probe item are released with the code.

B.3 Carrier frames

Each probe word W is spoken in two frames. The word before W ends in a vowel and the word after it begins with a vowel (or a glottal onset), so that no neighbouring consonant can merge with or be aligned to the target. The frames have not yet been reviewed by native speakers.

B.4 Model architectures

Figures 8–14 draw the inference path of each released model in the visual vocabulary of Figure 1: blue for text encoders and language models, purple for acoustic decoders and vocoders, grey for speaker conditioning, a red row for the embedding row that the probe word looks up, and $\times N$ tags for repeated blocks (a residual connection is drawn as $\oplus$, or as an arrow into the Add & LayerNorm step of a post-norm block). Layer counts, widths, vocabulary sizes and up-sampling factors were read from each model’s released configuration files and package source code, and decoding settings from the manifests saved with every run (Table 4). Parts used only in training are omitted. The example word and its token splits are illustrative. Figure 7 first compares the seven models side by side.

B.5 Scorer details

Normalisation. Both sequences are mapped as follows: (1) espeak ASCII conventions to IPA ($dZ \rightarrow \widehat{d}z$, $r \rightarrow \widehat{r}$, $s \rightarrow \widehat{s}$, ...); (2) removal of stress, tone digits, length and joiners; collapse of doubled geminate tokens; (3) the per-language notation map (Indic c , c^h , j , $j^h \rightarrow \widehat{t}j$, $\widehat{t}j^h$, $\widehat{d}z$, $\widehat{d}z^h$); (4) the recogniser-level classes r , $\mathfrak{x} \rightarrow r$, $v \rightarrow v$, $h \rightarrow h$, $\mathfrak{t} \rightarrow l$; (5) merging of split affricates and of stop + $[h]$ into aspirated stops.

Alignment. Needleman–Wunsch with insertion/deletion cost 0.5, substitution cost $\min(1, d_{\text{PanPhon}}/4)$, and cost 1 for any consonant–vowel substitution.

Floors. Occurrences are located by diffing the G2P of each standalone word against two substituted variants and are kept only when the word’s standalone phones appear, as a whole G2P word,

Model	Checkpoint	Input unit	Decoding (released defaults)	Voice	Languages probed
MMS-TTS	facebook/mms-tts-ara @afba26a	character	noise_scale=0.667, noise_scale_duration=0.8, speaking_rate=1.0	built-in	ar bn de el es fa fr gu he hi kn ml mr pa pl pt ru sw ta te tr uk ur vi
IndicF5	ai4bharat/IndicF5	character (2545)		FLEURS prompt	bn gu hi kn ml mr pa ta te
Chatterbox-MTL	ResembleAI/chatbox (multilingual T3)	code point (MTLTokenizer)	exaggeration=0.5, cfg_weight=0.5, temperature=0.8, repetition_penalty=2.0, min_p=0.05, top_p=1.0	FLEURS prompt	ar de el es fr he hi it pl pt ru sw tr
XTTS-v2	coqui/XTTS-v2	VoiceBPE 6681	temperature=0.75, repetition_penalty=5.0, top_k=50, top_p=0.85, length_penalty=1.0	FLEURS prompt	ar cs de es fr hi it pl pt ru tr
OmniVoice	k2-fsa/OmniVoice	Qwen3 byte-level BPE	num_step=32, guidance_scale=2.0, t_shift=0.1, layer_penalty_factor=5.0, position_temperature=5.0, class_temperature=0.0, denoise=True, preprocess_prompt=True	FLEURS prompt	am ar bn cs de el es fa fr gu he hi hy it ka kn ml mr pa pl pt ru sw ta te tr uk ur vi
Qwen3-TTS	Qwen/Qwen3-TTS-12Hz-0.6B-Base	Qwen byte-level BPE	do_sample=True, repetition_penalty=1.05, temperature=0.9, top_p=1.0, top_k=50, subtalker_dosample=True, subtalker_temperature=0.9, subtalker_top_p=1.0	FLEURS prompt	de es fr it pt ru
Kokoro-82M	hexgrad/Kokoro-82M	espeak-ng IPA		voice ef_dora	es fr hi it pt

Table 4: Every released model probed in E1: checkpoint (and revision where the hub records one), input unit, the released decoding parameters used for every clip, voice, and the languages probed.

Code	Language	Script	Units	Probed	Items	Word source	Floor tokens	Floor %	MMS text	Models
am	Amharic	Ethiopic	30	28	912	wordfreq	4,993	49.3	—	OV
ar	Arabic	Arabic	27	27	940	wordfreq	17,451	73.8	arbnv	MMS CB XT OV
bn	Bengali	Bengali	36	36	1204	wordfreq	65,041	67.9	benobev (id.)	MMS IF5 OV
cs	Czech	Latin	24	24	862	wordfreq	44,090	83.2	—	XT OV
de	German	Latin	21	21	738	wordfreq	50,689	95.4	deu1912 (id.)	MMS CB XT OV Q3
el	Greek	Greek	18	18	624	wordfreq	40,236	80.6	christos-c Greek	MMS CB OV
en	English	Latin	21	21	736	wordfreq	0	—	—	—
es	Spanish	Latin	22	22	774	wordfreq	53,093	93.8	spablm (id.)	MMS CB XT OV Q3 KO
fa	Persian	Arabic	31	31	1068	wordfreq	33,388	86.6	pesOPV (id.)	MMS OV
fr	French	Latin	21	21	686	wordfreq	25,142	95.1	fraLSG (id.)	MMS CB XT OV Q3 KO
gu	Gujarati	Gujarati	34	34	1162	wordfreq	70,711	70.3	guj2017 (id.)	MMS IF5 OV
he	Hebrew	Hebrew	26	26	846	wordfreq	55,841	57.1	heb (id.)	MMS CB OV
hi	Hindi	Devanagari	40	39	1354	wordfreq	31,531	73.2	hin2017	MMS IF5 CB XT OV KO
hy	Armenian	Armenian	30	30	1076	wordfreq	67,213	64.6	—	OV
it	Italian	Latin	16	16	566	wordfreq	36,795	95.8	—	CB XT OV Q3 KO
ka	Georgian	Georgian	28	28	1008	wordfreq	78,235	67.4	—	OV
kn	Kannada	Kannada	34	34	1164	wordfreq	55,480	71.9	kan2017 (id.)	MMS IF5 OV
ml	Malayalam	Malayalam	36	36	1234	wordfreq	53,383	62.8	malc (id.)	MMS IF5 OV
mr	Marathi	Devanagari	34	34	1154	wordfreq	76,583	65.5	mar	MMS IF5 OV
pa	Punjabi	Gurmukhi	32	32	1110	wordfreq	38,560	62.6	pan (id.)	MMS IF5 OV
pl	Polish	Latin	23	23	816	wordfreq	42,648	76.8	polubg (id.)	MMS CB XT OV
pt	Portuguese	Latin	19	19	654	wordfreq	27,191	81.9	porbr2018 (id.)	MMS CB XT OV Q3 KO
ru	Russian	Cyrillic	21	21	756	wordfreq	40,051	65.3	rusyn (id.)	MMS CB XT OV Q3
sw	Swahili	Latin	18	18	648	wordfreq	29,351	80.3	swhonen (id.)	MMS CB OV
ta	Tamil	Tamil	22	22	742	wordfreq	38,343	65.9	tam2017	MMS IF5 OV
te	Telugu	Telugu	35	35	1210	wordfreq	35,156	73.5	tel2017 (id.)	MMS IF5 OV
tr	Turkish	Latin	21	21	744	wordfreq	51,820	82.9	turytc	MMS CB XT OV
uk	Ukrainian	Cyrillic	22	22	788	wordfreq	49,737	70.1	ukronpu (id.)	MMS OV
ur	Urdu	Arabic	34	34	1134	wordfreq	23,005	69.0	urdgvu	MMS OV
vi	Vietnamese	Latin	17	17	338	wordfreq	23,804	53.7	vieovcb (id.)	MMS OV

Table 5: Every language: consonant units in the inventory and with probe words, probe items, word-list source, scored FLEURS consonant tokens and strict human accuracy over all of them (w2v2p; this weights common letters more than the per-cell floors of the main text), the New Testament edition counted for MMS-TTS (id.: translation identified as MMS-TTS’s training text) and the models probed (MMS MMS-TTS, IF5 IndicF5, CB Chatterbox-MTL, XT XTTS-v2, OV OmniVoice, Q3 Qwen3-TTS, KO Kokoro-82M).

Lang	init frame	mid frame
hi	W एक शब्द है। <i>W is a word.</i>	मैंने आज फिर से W ऊँचे स्वर में कहा। <i>Today I said W again loudly.</i>
mr	W एक शब्द आहे. <i>W is a word.</i>	मी आज पुन्हा W असे म्हणालो. <i>Today I said W again.</i>
bn	W একটি শব্দ। <i>W is a word.</i>	আমি আজকে W আবার বললাম। <i>Today I said W again.</i>
ta	W ஒரு சொல். <i>W is a word.</i>	நான் இன்று W என்று மீண்டும் சொன்னேன். <i>Today I said W again.</i>
te	W ఒక పదం. <i>W is a word.</i>	నేను ఈ రోజు మళ్ళీ W అని చెప్పాను. <i>Today I said W again.</i>
gu	W એક શબ્દ છે. <i>W is a word.</i>	મેં આજે ફરીથી W એમ કહ્યું. <i>Today I said W again.</i>
kn	W ಒಂದು ಪದ. <i>W is a word.</i>	ನಾನು ಇಂದು ಮತ್ತೆ W ಎಂದು ಹೇಳಿದೆ. <i>Today I said W again.</i>
ml	W ഒരു വാക്കാണ്. <i>W is a word.</i>	ഞാൻ ഇന്ന് W എന്ന് വീണ്ടും പറഞ്ഞു. <i>Today I said W again.</i>
pa	W ਇੱਕ ਸ਼ਬਦ ਹੈ। <i>W is a word.</i>	ਮੈਂ ਅੱਜ ਦੁਬਾਰਾ W ਉੱਚੀ ਆਵਾਜ਼ ਵਿੱਚ ਕਿਹਾ। <i>Today I said W again loudly.</i>
ar	W أيضا كلمة. <i>W is also a word.</i>	أمام الجميع W قلت اليوم مرة أخرى. <i>Today I said W again in front of everyone.</i>
fa	W امروز کلمه من است. <i>W is my word today.</i>	آهسته گفتم W من امروز دوباره. <i>Today I said W again softly.</i>
ur	W ایک لفظ ہے. <i>W is a word.</i>	اونچی آواز میں کہا۔ W میں نے آج دوبارہ <i>Today I said W again loudly.</i>
he	W אחת מן המילים. <i>W is one of the words.</i>	אתמול בערב W אמרתי לו. <i>I told him W yesterday evening.</i>
ru	W — это слово. <i>W is a word.</i>	Я сказал снова W и ушёл. <i>I said W again and left.</i>
uk	W — останнє слово. <i>W is the last word.</i>	Я сказав знову W і пішов. <i>I said W again and left.</i>
el	W είναι μια λέξη. <i>W is a word.</i>	Σήμερα είπα ξανά W αργά. <i>Today I said W again slowly.</i>
ka	W ერთი სიტყვაა. <i>W is one word.</i>	დღეს ისევ ვთქვამ W ასე. <i>Today I said W again like this.</i>
hy	W իմ բառն է: <i>W is my word.</i>	Այսօր նորից ասացի W ուշադիր: <i>Today I said W again carefully.</i>
pl	W i inne słowa. <i>W and other words.</i>	Powiedziałam znowu W i zamilkłam. <i>I said W again and fell silent.</i>
cs	W a jiná slova. <i>W and other words.</i>	Dnes jsem řekla znovu W a odešla. <i>Today I said W again and left.</i>
tr	W ayrıca bir kelimedir. <i>W is also a word.</i>	Bugün yine W ile başladım. <i>Today I started with W again.</i>
vi	W ở đây là một từ. <i>W here is a word.</i>	Hôm nay tôi nói lại W ở lớp. <i>Today I said W again in class.</i>
es	W es una palabra. <i>W is a word.</i>	Hoy dije de nuevo W en voz alta. <i>Today I said W again aloud.</i>
fr	W est un mot. <i>W is a word.</i>	Aujourd'hui j'ai dit W encore une fois. <i>Today I said W once again.</i>
de	W ist ein Wort. <i>W is a word.</i>	Heute sagte ich wieder W und ging. <i>Today I said W again and left.</i>
it	W è una parola. <i>W is a word.</i>	Oggi ho detto W ancora una volta. <i>Today I said W once more.</i>
pt	W é uma palavra. <i>W is a word.</i>	Hoje eu disse W outra vez. <i>Today I said W again.</i>
en	W is a word.	So I say W again today.
sw	W ina maana. <i>W has a meaning.</i>	Leo nimesema tena W asubuhi. <i>Today I said W again in the morning.</i>
am	W እንደ ቃል ነው። <i>W is a word.</i>	ዛሬ እንደገና W አልኩ። <i>Today I said W again.</i>

Table 6: Carrier frames (W = probe word), with an English translation of each language’s two frames (English itself is used only in E2).

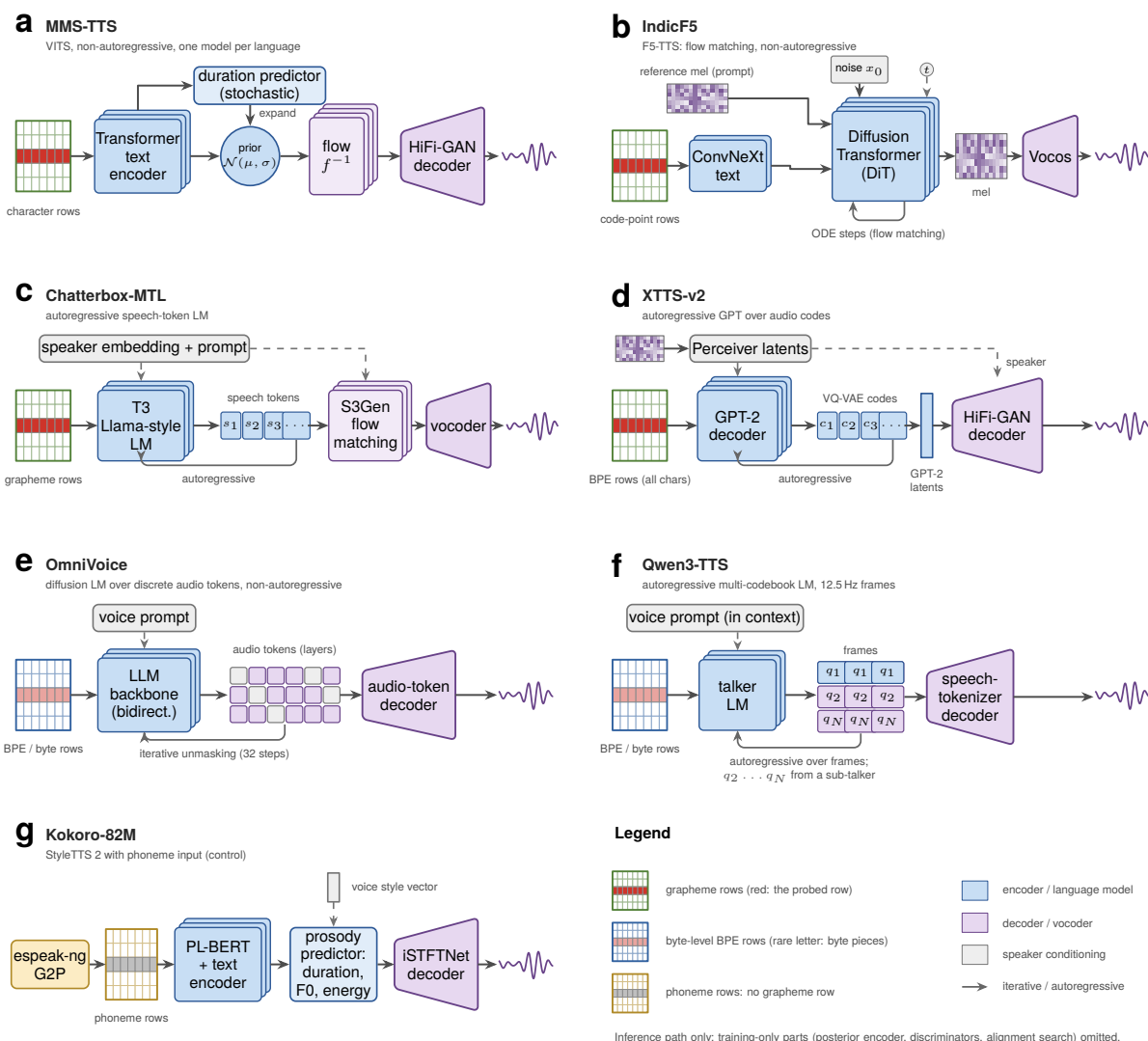


Figure 7: **What an input row is in each of the seven models** (inference only). Every model looks up one learned row per input unit, but the unit differs: a character (a), a Unicode code point (b, c), a BPE token (d), a byte-level BPE token or byte piece (e, f), or an IPA phoneme (g, the control, which has no letter rows). Blue: encoders and language models; purple: acoustic decoders and vocoders; grey: voice conditioning; curved arrows: step-by-step decoding. Figures 8–14 draw each model in full.

exactly as many times in the utterance **G2P** as the word occurs in the text.

B.6 Ruler validation

Table 7 lists every check behind the results of the main text and its outcome.

Planted errors. Table 8 reports, for MMS-TTS clips in which the probe word was deliberately rewritten with a different grapheme, how often each ruler still accepts the original target phone (false accept, FA), by the phonological feature group that separates the original and planted consonants (Pan-Phon).

Floor mismatch. Table 9 rescores the take-0 MMS-TTS clips after adding pink noise at fixed signal-to-noise ratios, next to the median SNR we measure on the FLEURS recordings behind the floors; Figure 15 gives the noise control per language: each pair of bars is one language, light for +30 dB and dark for +20 dB, against the 10-point failure margin. Table 10 gives the padding control: the same clips scored with 0, 250 and 500 ms of silence around them.

Chapter B in brief

Seven released models, from character-input VITS to a phoneme-input control, are probed only in languages they support, with their released decoding

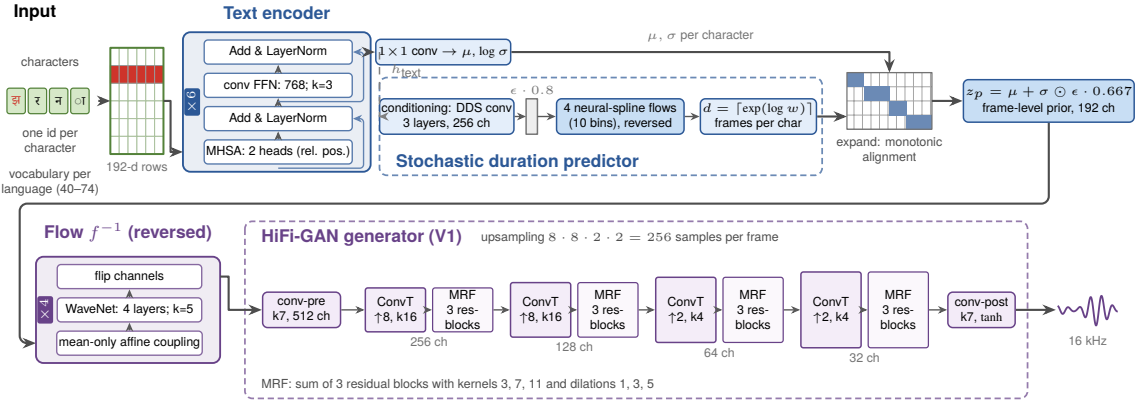


Figure 8: **MMS-TTS (VITS)**, one model per language; sizes from the released facebook/mms-tts-* configurations. Each character of the language’s vocabulary has one embedding row; a character outside it has none and is dropped by the tokenizer. The noise scales (0.8 for durations, 0.667 for the prior) are the released defaults, used in all our runs. Omitted (training only): the posterior encoder (16 WaveNet layers), monotonic alignment search and the discriminators.

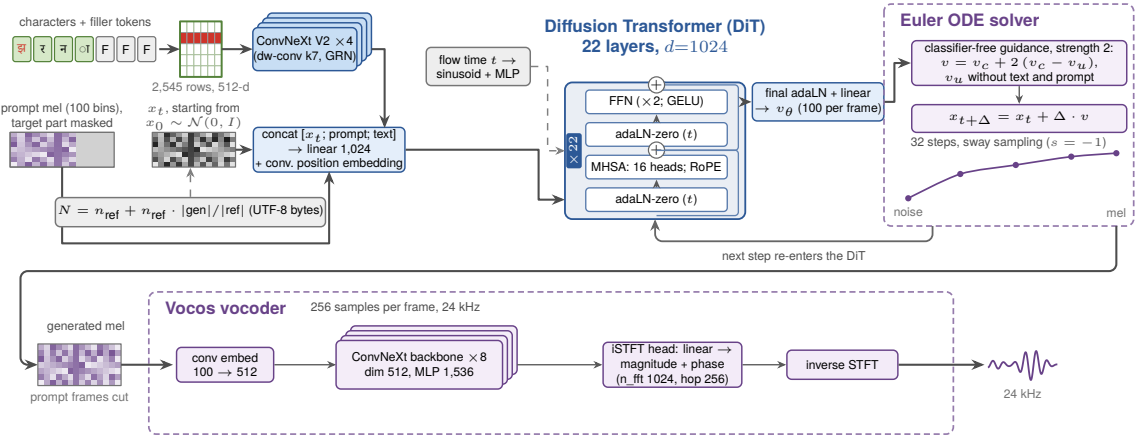


Figure 9: **IndicF5 (F5-TTS)**. The text is a character sequence (2,545 code-point rows) padded with filler tokens to the output length N , which is set from the prompt by the ratio of text lengths. The DiT predicts a velocity field from the noisy mel, the masked prompt mel and the text; a 32-step Euler solver with classifier-free guidance integrates it. Sizes from the released model code and the Vocos configuration; decoding settings are the package defaults used in all our runs.

defaults and a recorded seed per clip. The consonant graphemes of each script that have qualifying words are probed in real words, three positions and two carrier frames, and scored with the same pipeline as human FLEURS speech. The ruler accepts wrong aspiration too often, so no claim rests on aspiration; added noise moves average accuracy far less than the failure margin, although one language (Arabic) exceeds it at +20 dB.

What the materials do not cover. The carrier frames were written for this study and not yet reviewed by native speakers, the G2P reference favours transparently spelled words, and no listener judgements are reported yet.

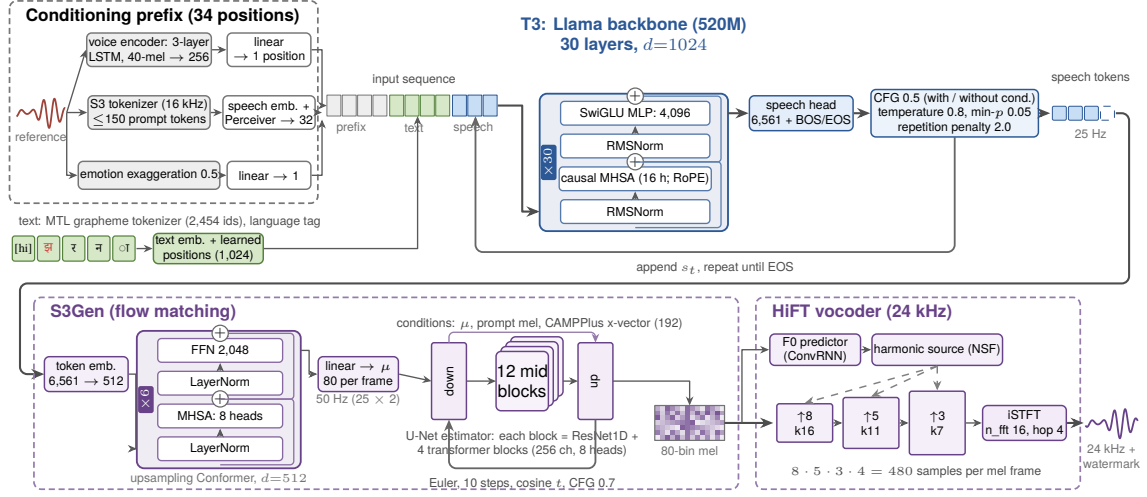


Figure 10: **Chatterbox multilingual**. T3 predicts 25 Hz S3 speech tokens autoregressively from a conditioning prefix and grapheme-level text rows; S3Gen turns the tokens into an 80-bin mel by conditional flow matching, and HiFT turns the mel into a 24 kHz waveform. Sizes from the chatterbox 0.1.7 package; decoding settings as recorded in our manifests (the package defaults).

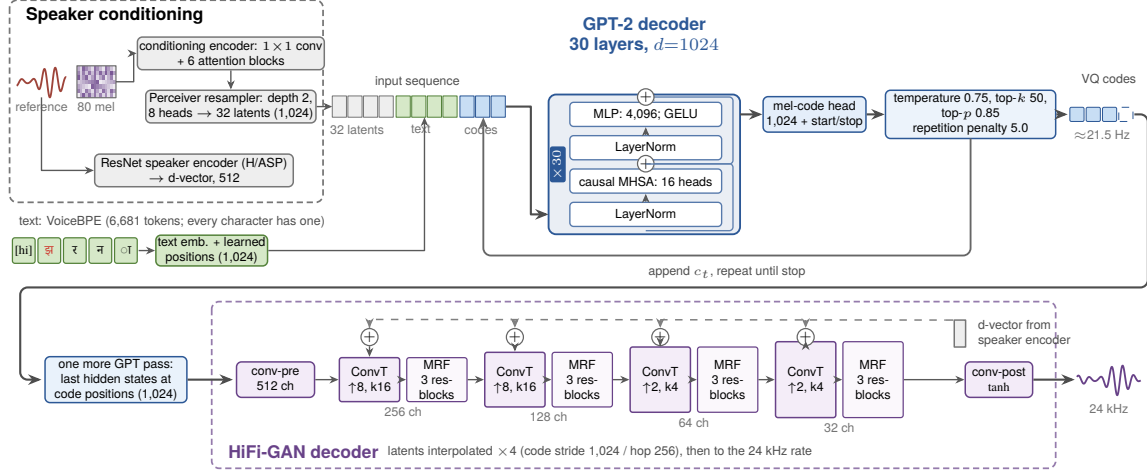


Figure 11: **XTTS-v2**. A GPT-2 decoder predicts discrete VQ codes from 32 Perceiver conditioning latents and the text rows of a 6,681-token BPE; a second pass yields the hidden states that the HiFi-GAN decoder turns into audio, conditioned on a speaker d-vector. The VQ-VAE that defines the codes is used only in training and is omitted. Sizes from the released configuration and the TTS package; decoding settings as recorded in our manifests.

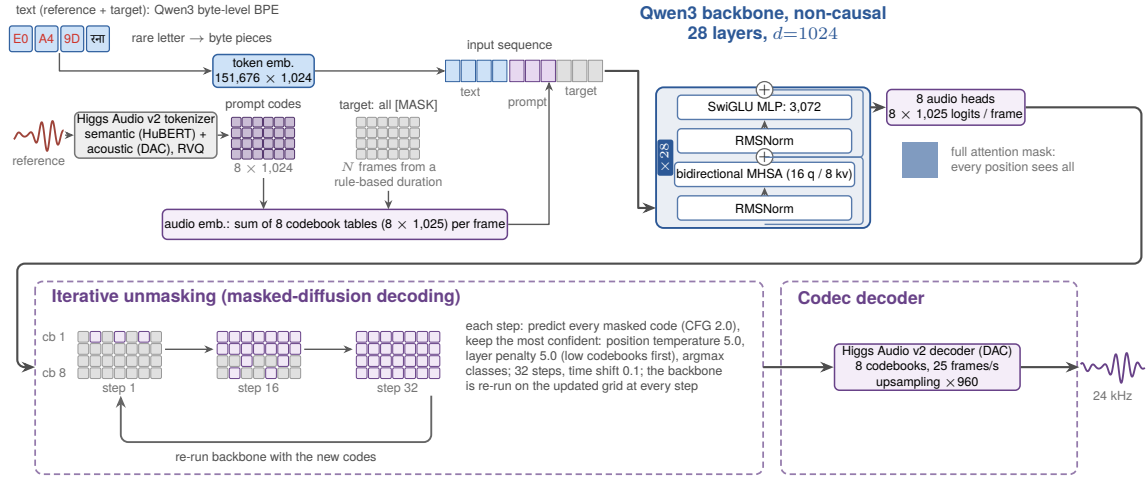


Figure 12: **OmniVoice**. A non-causal Qwen3 backbone reads text as byte-level BPE (a rare letter becomes several byte pieces) together with 8-codebook audio tokens, and decodes all target frames in parallel by iterative unmasking. Sizes from the released configurations of the model and of the Higgs Audio v2 tokenizer; decoding settings are the package defaults recorded in our manifests.

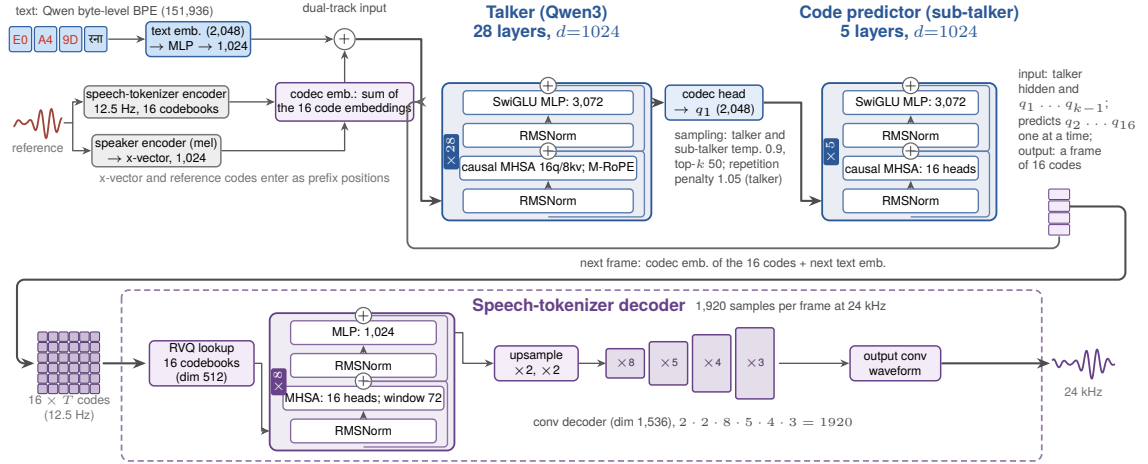


Figure 13: **Qwen3-TTS-12Hz-0.6B-Base**. For each 12.5 Hz frame the talker reads the sum of a text embedding and the codec embeddings (dual-track input) and predicts the first codebook; the code predictor fills in the other 15 codebooks; the speech-tokenizer decoder produces 24 kHz audio. Sizes from the released configurations, input construction from the qwen-tts code, sampling settings as recorded in our manifests.

Check	Outcome
Per-letter human floor	1,259,561 FLEURS consonant tokens; cell floors average 40–94% per language (median 58%), so every score is relative to its own floor
Two phone rulers	onset effect significant in 7/7 models (w2v2p) and 6/7 (ZIPA); H4 under both ($p = 3 \times 10^{-5}$, $p = 1 \times 10^{-7}$); onset repair positive in 4/4 and 4/4 models
Planted errors	strict false accepts 18.8% (w2v2p) and 15.6% (ZIPA); aspiration is the weak spot (61%), so no claim rests on aspiration
Noise control	MMS-TTS accuracy falls 0.3 and 2.3 points on average at +30 and +20 dB; one language (Arabic) exceeds 10 points at +20 dB
Pre-registration	plans for E1-T, E1-T2, E1-R, E1-H, E3-O, E4, E5' and E7, and an analysis plan for E6, hashed before their outcomes; every amendment and deviation disclosed (Appendix F.1)
Statistics	cluster bootstrap over words and takes; crossed random effects for language, letter and word; Holm correction within each family of tests
Held-out replication	E1-R: none of the four pre-registered tests significant after Holm (MMS-TTS +0.12 [−0.07, +0.32])
Native listeners	test prepared for Bengali and Hindi; not yet run, so no claim rests on it

Table 7: **The checks behind our results**, and their outcomes.

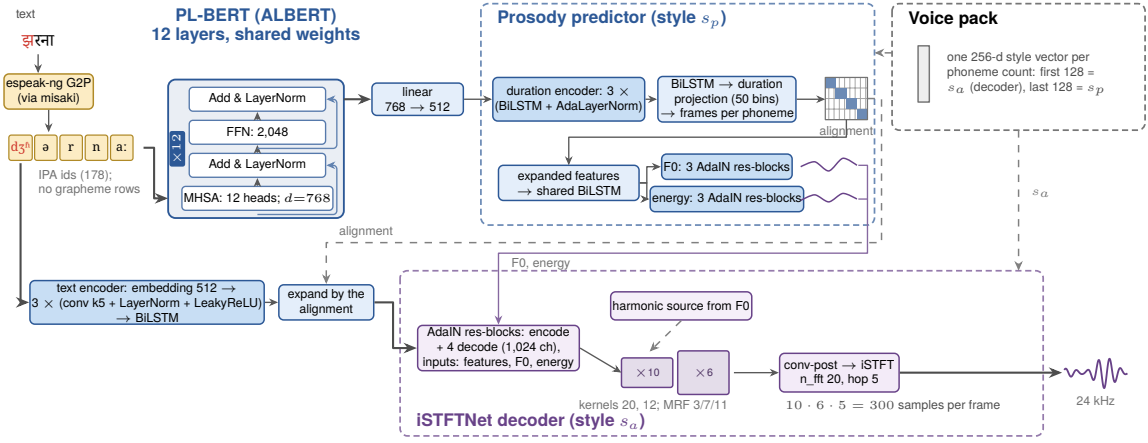


Figure 14: **Kokoro-82M** (the StyleTTS 2 architecture without its diffusion style sampler). The input is IPA from espeak-ng, so the model has no grapheme row to probe (the phoneme-input control); style vectors come from a fixed voice pack. Sizes from the released configuration and the kokoro package.

Contrast	wav2vec2			ZIPA		
	<i>n</i>	FA strict	FA near	<i>n</i>	FA strict	FA near
All	1212	18.8	48.6	1279	15.6	49.8
Voicing	544	13.2	74.3	604	12.7	73.5
Aspiration	110	60.9	74.5	110	39.1	77.3
Place	277	24.2	26.4	271	19.9	26.9
Manner	75	12.0	12.0	75	9.3	13.3
Multiple	206	6.3	10.2	219	8.2	11.4

Table 8: **Planted-error test**: false-accept rate (%) of the original target phone when a minimally different consonant was synthesised instead, strict and near criteria.

Added noise (SNR)	Clips	Clean	Noisy	Drop
30 dB	10514	54.7	54.4	+0.3
20 dB	10514	54.7	52.4	+2.3

Table 9: **Noise-matched control**: strict accuracy (%) of MMS-TTS clips, clean and with added pink noise.

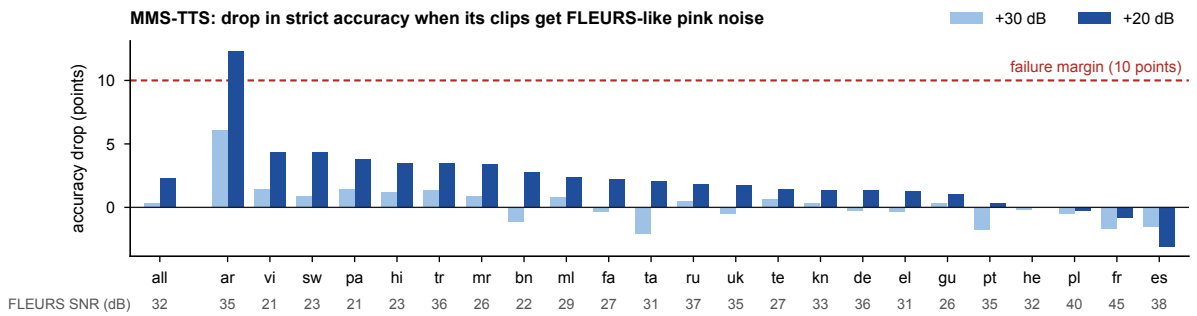


Figure 15: **Noise-matched control for every language**. Drop in MMS-TTS strict accuracy (points) when pink noise is added at +30 and +20 dB SNR (“all”: every language pooled), languages sorted by the drop at +20 dB; below each language, the median SNR of its FLEURS floor recordings. Only Arabic crosses the 10-point failure margin, and only at +20 dB. Clean and noisy accuracies are in the released degrade.csv.

Model	Clips	0 ms		250 ms (used)		500 ms	
		all	utt.-init.	all	utt.-init.	all	utt.-init.
MMS-TTS	10,514	54.6	47.3	54.7	47.1	54.9	47.3
OmniVoice	13,322	65.3	66.5	65.4	66.0	65.4	66.6
XTTS-v2	4,334	67.6	57.5	70.4	64.2	70.5	63.1

Table 10: Padding control: strict accuracy (%) when the phone recogniser receives each clip with 0, 250 (the setting used throughout) or 500 ms of silence padding, on the same clips; utt.-init.: word-initial consonants of utterance-initial items.

C How Common Is It: Prevalence in Full

Every prevalence result behind §4 of the main text, for every model and language: accuracy against the floor, failing cells, absent rows and the utterance-onset effect, with further measurements collected in one figure.

In this chapter

C.1 Per-language results	26
C.2 Further measurements in one place	26
C.3 E1 for every model and language	26

Model	Utt.-init.	Embedded	Δ [95% CI]	Floor _{init}	Floor _{emb}
MMS-TTS	47.6	65.5	-17.9 [-20.9, -14.9]	80.4	78.4
IndicF5	52.3	61.3	-9.1 [-13.6, -4.6]	80.0	78.5
Chatterbox-MTL	67.2	76.6	-9.4 [-13.4, -5.2]	85.0	84.1
XTTS-v2	64.1	75.1	-11.0 [-15.9, -6.8]	90.2	88.3
OmniVoice	66.6	71.5	-4.9 [-7.8, -2.4]	79.9	78.7
Qwen3-TTS	85.5	87.7	-2.2 [-7.1, +2.8]	94.1	92.4
Kokoro-82M	74.9	77.9	-3.1 [-9.5, +3.1]	94.1	90.9

Table 11: Word-initial consonants of probe words that open the utterance versus the same words inside a sentence (strict, w2v2p, %), with the human floors for the same positions.

C.1 Per-language results

Table 11 gives the onset comparison of Figure 4 with the human floors for the same positions.

Table 12 lists every grapheme that has no input row in a model, with how often it is still pronounced correctly and how often it is simply deleted.

C.2 Further measurements in one place

Figure 16 collects six measurements that support the main text: the floors, the planted errors, the noise control, the row norms, the same-sound pairs and the onset repair per language.

How to read the prevalence figures. The four figures that follow replace long per-language tables; each cell prints its value, so nothing is lost. Figure 17 is the map of the whole benchmark: one row per model, one column per language, languages grouped by script. Panel (a) shows how far the model is above (blue) or below (red) the human floor; panels (b) and (c) show what share of the model’s cells fail under each phone ruler. A language that is red in (a) and dark in both (b) and (c) is a robust failure; a cell dark under only one ruler is a ruler disagreement. Figure 18 gives the onset comparison of Table 11 for every model and language; its last row is human speech, so a red model cell above a pale human cell is a deficit that speakers themselves do not show. Figure 19 counts the failing cells by letter position and frame; the

dark segments are cells whose probe word opens the utterance. Figure 20 covers the OmniVoice letters that its byte-level BPE writes with several byte pieces, the one case where a letter is neither one row nor absent.

C.3 E1 for every model and language

Accuracy, floor and failing cells under both rulers and three criteria; the onset effect per language; every failing cell; OmniVoice’s split graphemes.

Chapter C in brief

Every model has failing cells, with MMS-TTS, trained on one book per language, failing most. Graphemes without an input row are dropped by the tokenizer: at most 3.3% of their tokens are correct. Utterance-initial consonants fall short of the same consonants after a carrier phrase in every model under w2v2p (six of seven under ZIPA), while human speech, on average, shows no such gap.

Questions this chapter leaves open. Whether listeners hear the failures the rulers flag, and whether the onset deficit holds in spontaneous rather than read text.

Model	Lang	Unit	n	Correct %	Deleted %
IndicF5	gu	ᱥ	12	0.0	100.0
MMS-TTS	es	w	90	3.3	87.8
MMS-TTS	fr	w	60	0.0	98.3
MMS-TTS	gu	ᱥ	30	0.0	100.0
MMS-TTS	kn	ಃ	20	0.0	75.0
MMS-TTS	ml	ഓ	40	0.0	52.5
MMS-TTS	pa	ਏ	60	0.0	85.0
MMS-TTS	pa	ਐ	70	0.0	37.1
MMS-TTS	te	ఁ	90	2.2	58.9
MMS-TTS	te	ఁ	90	0.0	55.6
MMS-TTS	te	ఁ	90	0.0	63.3

Table 12: **Graphemes without an input row** (model $\times$ language $\times$ grapheme, pooled over positions and frames): strict accuracy and share of tokens deleted.

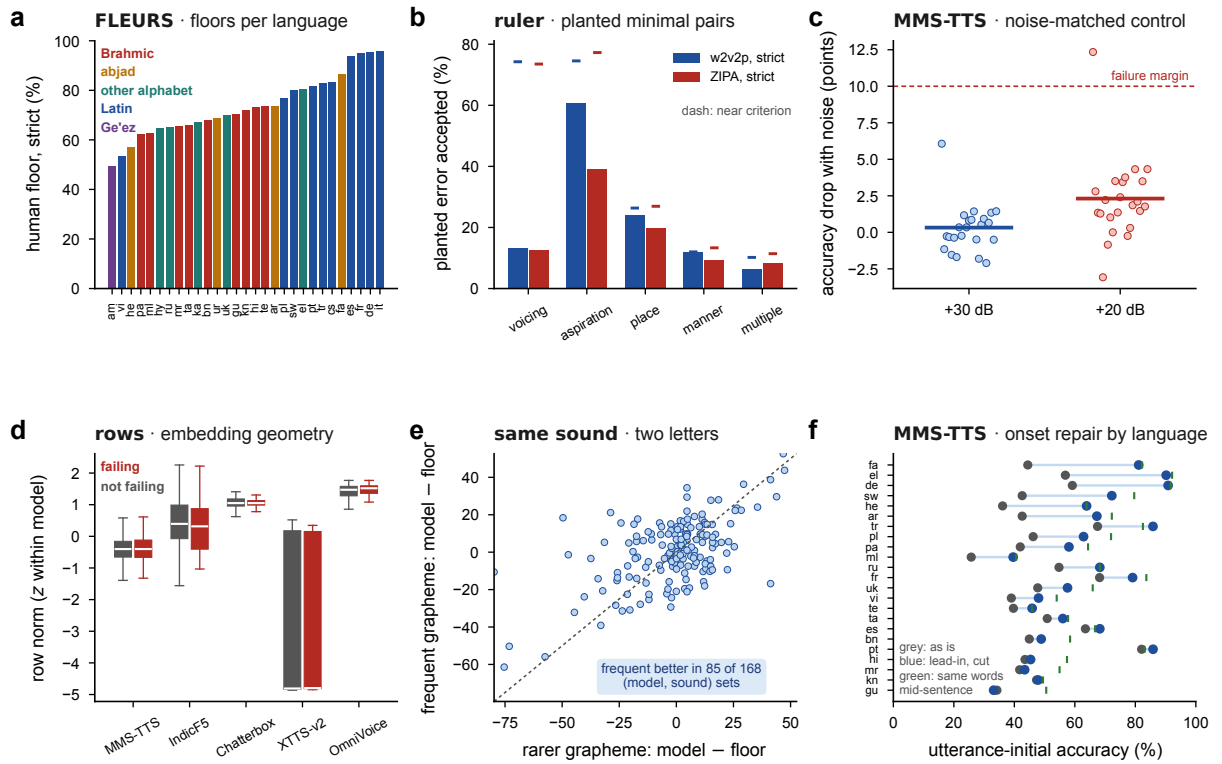


Figure 16: **Further measurements in one place.** (a) Strict human accuracy over all FLEURS consonant tokens per language, coloured by script type (this weights common letters more than the per-cell floors of the main text). (b) Planted minimal-pair errors accepted by each ruler, by the contrast that separates the pair (bars: strict criterion; dashes: near). (c) Noise-matched control: drop in MMS-TTS accuracy when its clips get FLEURS-like noise at +30 and +20 dB (points: languages; bars: all languages). (d) Embedding-row norms of failing and non-failing graphemes per model (z within model). (e) Same sound, two letters: model - floor of the more frequent grapheme for the same phone (points: model $\times$ sound sets; dashed: equality). (f) MMS-TTS onset repair per language: (first, 30 ms cut rule) utterance-initial accuracy as is, with a cut lead-in, and for the same words mid-sentence.

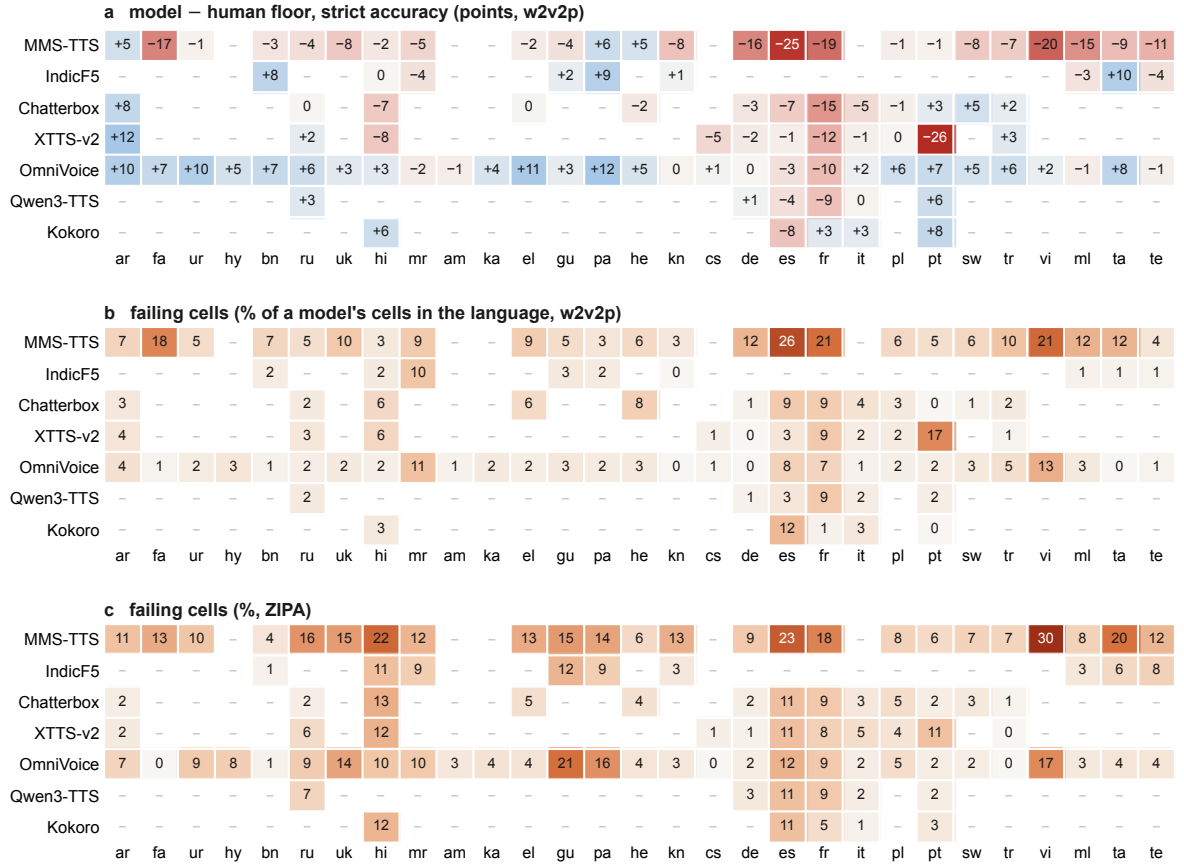


Figure 17: **E1 for every model and language.** (a) Model – human floor in strict accuracy (points; blue: above the floor, red: below). (b, c) Share of a model's cells that fail (95% CI of model – floor below –10 points) under each phone ruler. Languages are grouped by script; a dash marks a language the model does not support. Clip and cell counts and the near and human-calibrated criteria are in the released `summary_*.csv` files.

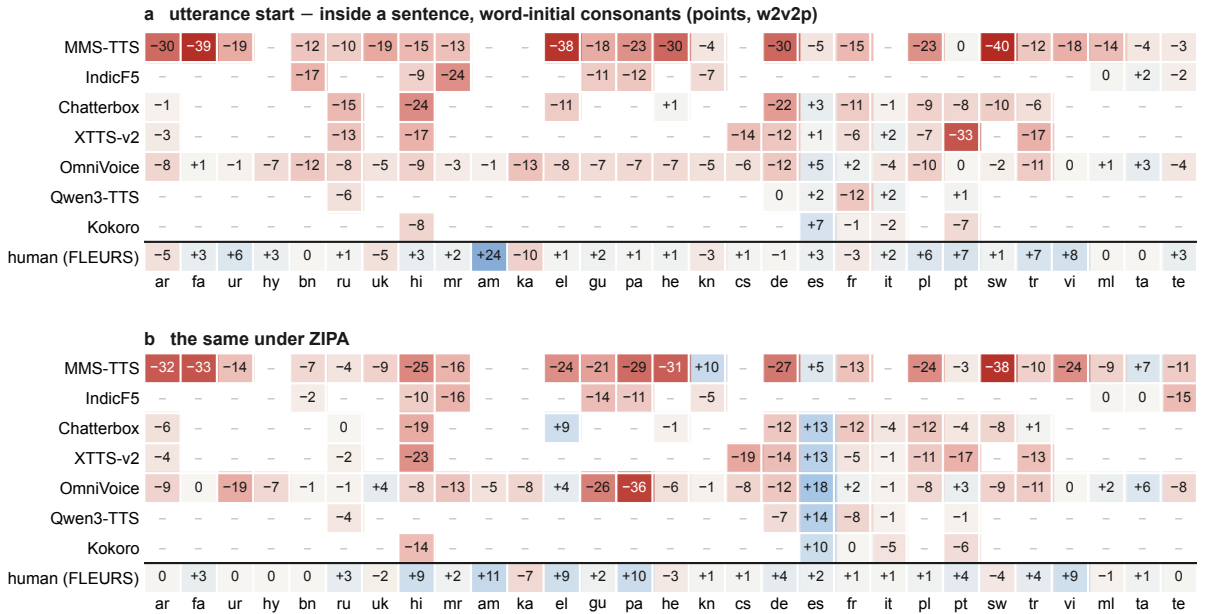


Figure 18: **Utterance-onset effect for every model and language.** Strict accuracy of word-initial consonants when the probe word opens the utterance minus the same words after the carrier phrase (points; red: weaker at the start). Last row: human FLEURS speech, words at the start of an utterance minus later words, scored with the same ruler.

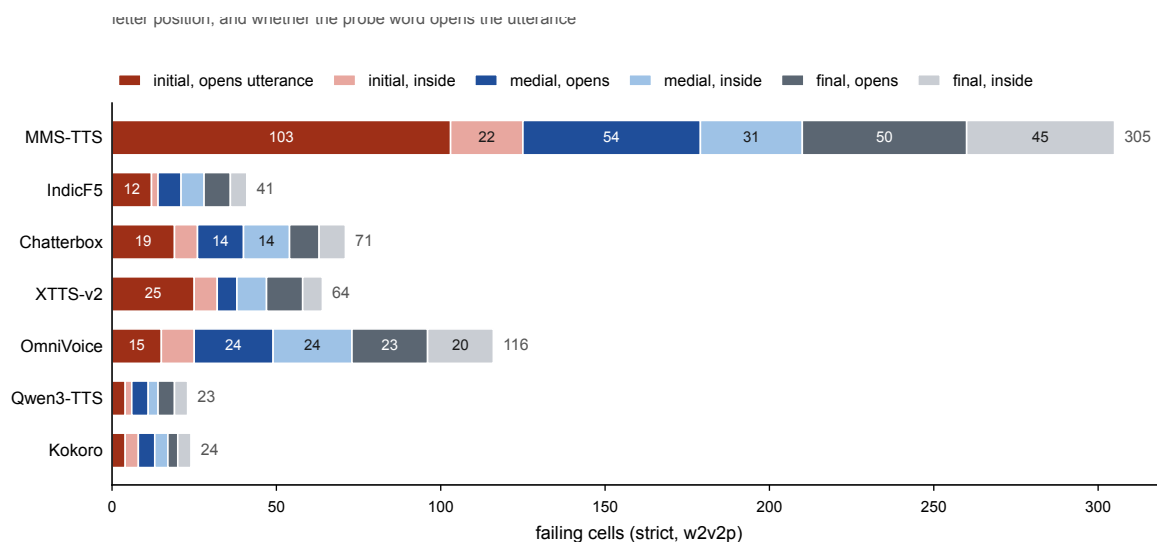


Figure 19: **Every failing cell, by model, letter position and frame (strict, w2v2p).** Dark segments are cells whose word opens the utterance. The cell-by-cell list (letter, language, accuracy and floor) is the released `cells.csv`.

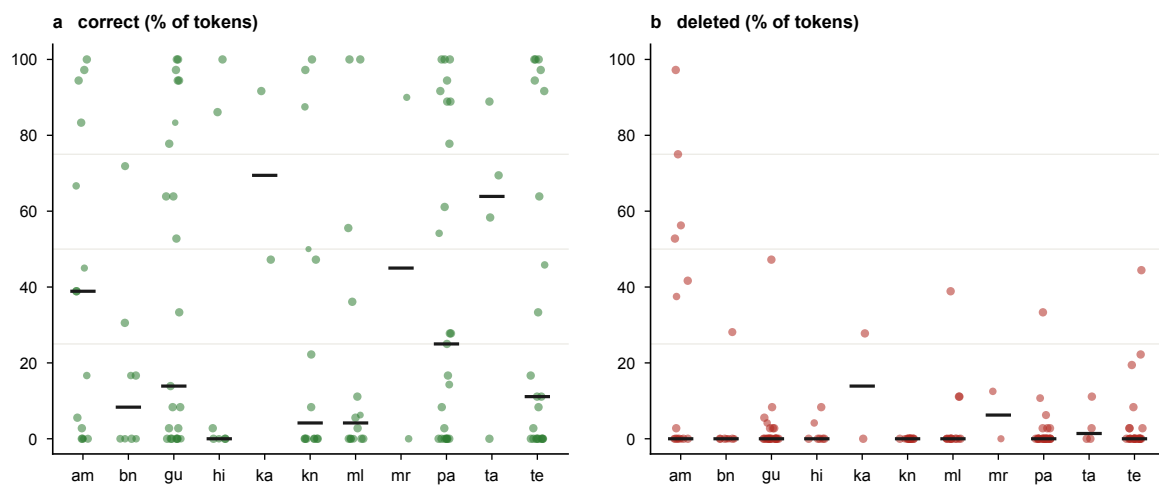


Figure 20: **OmniVoice letters that its byte-level BPE splits into several byte pieces.** Each dot is one letter (size: tokens), per language; bars: medians. This is the one case where a letter is neither one row nor absent; the list is the released `omnivoice_multitoken.csv`.

D What Causes It: Frequency, Rows and Controlled Training in Full

The analyses behind §5 of the main text: the frequency models and every sensitivity analysis, the pre-registered training-text counts with their replication, the geometry of the released rows, and the controlled-finetuning experiments in full.

In this chapter

D.1 Sensitivity of the frequency effect	30
D.2 Training-text counts and their replication	30
D.3 Row geometry and controlled finetuning in full	31

D.1 Sensitivity of the frequency effect¹⁴²

The main frequency model. For each model we fit

$$\begin{aligned} \text{correct} \sim & \log f_g + \text{position} + \text{frame} + \log f_w \\ & + \text{offset}(\text{logit floor}_{g,p}) + (1 \mid \text{lang}) \\ & + (1 \mid \text{lang:letter}) + (1 \mid \text{lang:word}), \end{aligned}$$

where f_g is the letter’s frequency in written text (standardised) and f_w the word frequency. The floor enters as an offset because a failure is defined relative to the floor; its counts are smoothed, $(k + \frac{1}{2})/(n + 1)$. The main fit uses judgeable cells (floor at least 0.2 on at least 20 human tokens). Across models, model is a fixed effect with a frequency $\times$ model interaction, and H4 is the frequency $\times$ tokenizer-type interaction; every estimate is checked with a logistic GLM clustered by letter. Table 13 gives the main model of §5.1 under both rulers.

Table 14 gives the frequency coefficient (log-odds per standard deviation of log grapheme frequency) under each pre-declared variant of the primary model (§3). *: $p < 0.05$ (Wald); †: singular fit. †The offset check column is the coefficient of logit floor entered *beside* the offset; 0 means the offset’s implied slope of 1 is adequate.

How to read the frequency figure and tables.

Figure 21 draws every fixed effect of the primary GLMM per model and of the pooled, utterance-onset and tokenizer-type (H4) models, one panel per term, with both rulers side by side (blue: w2v2p; red: ZIPA). An interval that crosses the dashed zero line is not distinguishable from no effect. The frequency coefficient is in log-odds per standard deviation of log frequency, and position and frame are coded against the medial position and the word inside the sentence; so the large negative “opens the utterance” effect in every model is the onset deficit

of §4. Table 15 repeats the sensitivity analyses under ZIPA. Table 16 checks the GLMM with simpler, model-light statistics: a rank correlation, a regression with errors clustered by language, a model-free slope in points per decade, and the same-phone pairs. Table 17 fits human and synthetic speech in one model, so that the interaction asks whether synthesis depends on training counts more than human speech does, and relates OmniVoice’s published training hours to its excess failure.

D.2 Training-text counts and their replication

Table 18 gives the pre-registered training-count tests of §5.1 and Table 19 the same-sound letter pairs.

The E1-R replication was pre-registered with four tests, Holm-corrected as one family: R1 and R2, the MMS-TTS training-count slope on reserve words under w2v2p and ZIPA; R3, the XTTS-v2 open-text frequency slope on reserve words under ZIPA; and R4, the frequency $\times$ utterance-onset interaction pooled over MMS-TTS and XTTS-v2 under ZIPA. R3 is +0.36 [+0.00, +0.72] (Holm $p = 0.09$) and R4 is +0.09 [−0.07, +0.26] (Holm $p = 0.22$); R1 and R2 are in Table 18. An earlier version of this paper reported only R1 and R2 and corrected over two tests; the correction over four is the pre-registered one (§F.2).

How to read the training-count tables.

Table 20 describes the counted text of every model and language: which text was counted, whether the language’s training translation was identified, how many graphemes were never seen and how many fall below 10^3 occurrences. Table 22 shows the excess failure by count bin for every model, word set and ruler; the lowest bins hold few graphemes, so their intervals are wide. Figure 22 draws every specification of the count GLMM as one panel, with the word set and model as rows and both rulers; a filled dot is a slope significantly above zero (one-

Model	β log frequency (z)		β utterance-initial frame	
	w2v2p	ZIPA	w2v2p	ZIPA
MMS-TTS	+0.04 [−0.18, +0.25]	+0.35 [+0.11, +0.58]*	−1.19 [−1.26, −1.13]*	−1.23 [−1.29, −1.16]*
IndicF5	+0.00 [−0.31, +0.32]	+0.00 [−0.33, +0.34]	−0.89 [−1.09, −0.68]*	−1.61 [−1.80, −1.42]*
Chatterbox-MTL	−0.06 [−0.33, +0.20]	+0.13 [−0.16, +0.42]	−0.46 [−0.61, −0.31]*	−0.38 [−0.54, −0.23]*
XTTS-v2	+0.08 [−0.17, +0.34]	+0.29 [−0.04, +0.63]	−0.56 [−0.70, −0.42]*	−0.72 [−0.89, −0.56]*
OmniVoice	+0.07 [−0.15, +0.29]	+0.13 [−0.12, +0.38]	−0.36 [−0.49, −0.24]*	−0.86 [−0.97, −0.75]*
Qwen3-TTS	+0.44 [−0.36, +1.24]	+0.17 [−0.60, +0.94]	−0.82 [−1.19, −0.44]*	−0.19 [−0.56, +0.19]
Kokoro-82M	+0.48 [−0.93, +1.89]	+1.22 [−0.14, +2.59]	−0.55 [−0.86, −0.24]*	−0.47 [−0.74, −0.21]*
Pooled	−0.16 [−0.31, −0.01]*	+0.03 [−0.13, +0.19]	−0.73 [−0.77, −0.69]*	−0.85 [−0.89, −0.81]*
Grapheme-row − BPE (H4)	+0.14 [+0.07, +0.21]*	+0.16 [+0.10, +0.22]*		
× utterance-initial word	+0.01 [−0.07, +0.10]	+0.15 [+0.07, +0.23]*		

Table 13: **Main frequency model** (judgeable cells, floor offset; random effects for language, letter and word) under both phone rulers. Log-odds coefficients with 95% CI; * $p < 0.05$. H4: does the frequency slope differ between one-row-per-letter models and byte-BPE models? Last row: interaction with the utterance-initial word. Sensitivity analyses: Table 14.

Model	Primary	+ lang slope	All cells	+ phone class	Within phone	Floor covariate	Offset check [†]
MMS-TTS	+0.04	+0.05	+0.67*	+0.20	+0.13	+0.04	+0.01
IndicF5	+0.00	+0.00 ^s	+0.65*	−0.13	−0.32	−0.14	+0.27*
Chatterbox-MTL	−0.06 ^s	−0.06 ^s	+0.16	−0.03 ^s	−0.09	−0.05 ^s	−0.04 ^s
XTTS-v2	+0.08	+0.08 ^s	+0.40*	+0.17	−0.11	+0.13	−0.15
OmniVoice	+0.07 ^s	+0.07 ^s	+0.63*	+0.01 ^s	−0.09 ^s	−0.05 ^s	+0.42** ^s
Qwen3-TTS	+0.44 ^s	+0.44 ^s	+0.32 ^s	+0.46 ^s	+0.21 ^s	+0.40 ^s	+0.85** ^s
Kokoro-82M	+0.48 ^s	+0.48 ^s	+3.15*	+0.53 ^s	−0.43 ^s	+0.30 ^s	+0.48 ^s

Table 14: **Frequency coefficient under the sensitivity analyses** of §D.1. Within phone: coefficient of the within-phone deviation of log frequency, with a phone random intercept.

Model	Primary	+ lang slope	All cells	+ phone class	Within phone	Floor covariate	Offset check [†]
MMS-TTS	+0.35*	+0.37*	+0.72*	+0.26	+0.12	+0.37*	−0.12*
IndicF5	+0.00	+0.01	+0.38*	−0.22	−0.57*	−0.12	+0.41*
Chatterbox-MTL	+0.13	+0.14	+0.25	+0.11	−0.03	+0.15	−0.07
XTTS-v2	+0.29	+0.30	+0.55*	+0.17	−0.13 ^s	+0.32	−0.13
OmniVoice	+0.13	+0.13 ^s	+0.56*	−0.10 ^s	−0.24 ^s	+0.03	+0.39*
Qwen3-TTS	+0.17 ^s	+0.17 ^s	+0.68 ^s	−0.02 ^s	+0.12 ^s	+0.24 ^s	+0.58** ^s
Kokoro-82M	+1.22 ^s	+1.22 ^s	+3.70*	+0.93 ^s	+0.85 ^s	+1.89** ^s	+2.26** ^s

Table 15: GLMM sensitivity analyses of the frequency coefficient under ZIPA (as Table 14 for w2v2p).

sided). The last two panels are the hinge model, which lets the slope change at 100 occurrences, and the right margin gives its likelihood-ratio test and the equivalence test above the knot. Table 23 counts the same-sound pairs for every model, word set and ruler, and Figure 23 draws the pre-registered low-count pairs one by one: each row is one pair of letters for the same phone, the filled dot the more frequent letter and the hollow dot the rarer one.

D.3 Row geometry and controlled finetuning in full

E2 and E2c design. F5-TTS v1 Base is finetuned on a fixed set of 16,000 LibriTTS-R train-clean-100 utterances (23.6 h). In a run, target letter x is written with its new Cherokee code point in exactly d utter-

ances, always the utterance’s single occurrence of x outside digraphs; the doses $d \in \{0, 10, 40, 160\}$ are assigned to four groups of three letters by a Latin square over four runs, and seeds re-draw which utterances realise each dose. Each run lasts three epochs (10,251 updates); the checkpoints at 5k and 10k updates and the model before training are probed. The standard arm finetunes every weight at F5-TTS’s learning rate (7.5×10^{-5} , AdamW, weight decay 0.01); the new-row arm gives the new rows 2×10^{-3} without weight decay and freezes the old rows. New rows start at random, matched in scale to the existing rows. Dose trends are tested with Page’s ordered test (blocks: letter $\times$ seed) and a mixed model. E2c repeats the design in MMS-TTS English with the public finetune-hf-vits recipe on LJSpeech (13,100

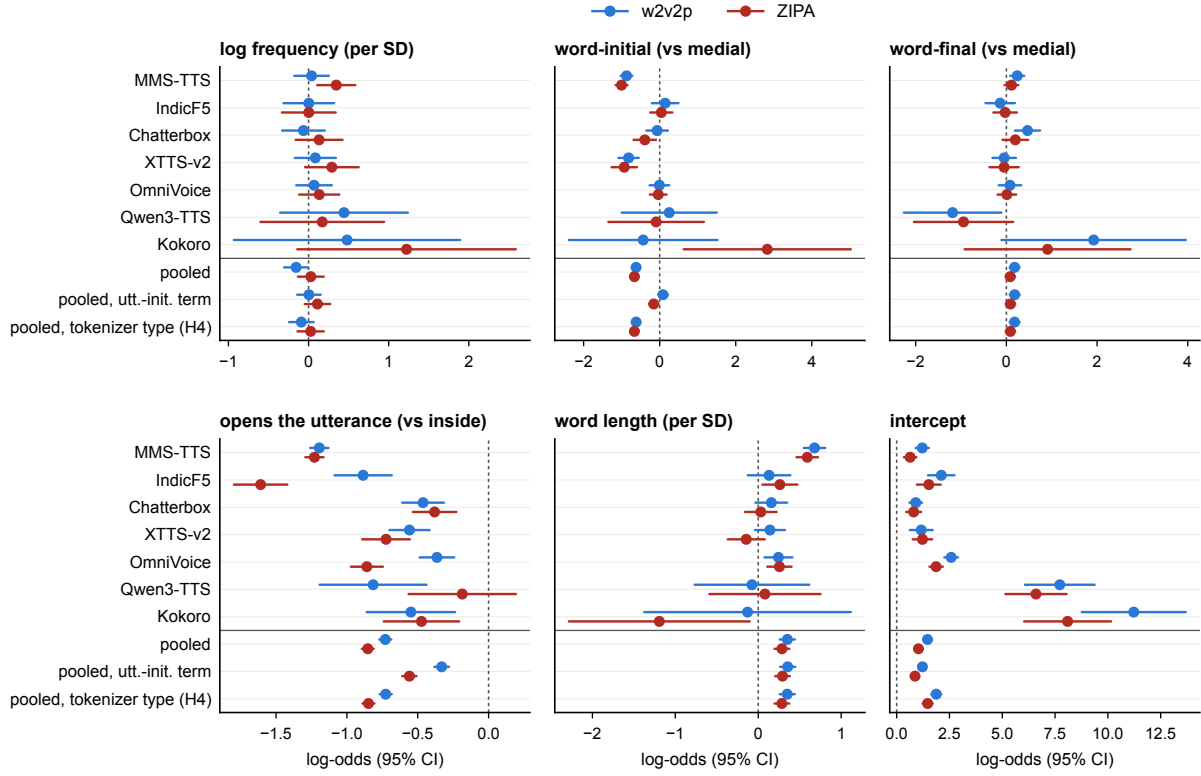


Figure 21: **Every fixed effect of the frequency GLMM** (log-odds of a correct token, 95% CI), per model and pooled, under both phone rulers (blue: w2v2p; red: ZIPA). Reference levels: medial position, word inside the sentence. Interactions not drawn (w2v2p / ZIPA): log frequency $\times$ one-row-per-grapheme models +0.14 [+0.07, +0.21] / +0.16 [+0.10, +0.22]; log frequency $\times$ phoneme-input model $-0.10 [-0.22, +0.02]$ / +0.07 $[-0.03, +0.18]$; log frequency $\times$ utterance-initial +0.01 $[-0.07, +0.10]$ / +0.15 [+0.07, +0.23]. The pooled model’s per-model slopes are in the released `glmm*.json`.

Model	Spearman ρ (p)	Cluster-robust GLM β	Model-free slope	Slope, utt.-init.	Matched pairs
MMS-TTS	+0.07 (<0.001)	-0.01 [-0.16, +0.13]	+3.7 [+1.0, +6.4]	-0.9 [-5.4, +4.1]	–
IndicF5	+0.17 (<0.001)	-0.04 [-0.31, +0.23]	-0.2 [-3.4, +2.8]	-2.1 [-6.1, +1.6]	10/23 (0.798)
Chatterbox-MTL	-0.03 (0.258)	-0.09 [-0.29, +0.10]	-2.7 [-6.3, +0.8]	-1.0 [-9.1, +6.5]	6/20 (0.979)
XTTS-v2	+0.04 (0.167)	+0.05 [-0.15, +0.25]	+0.6 [-2.8, +4.6]	+1.2 [-6.3, +8.9]	7/15 (0.696)
OmniVoice	+0.09 (<0.001)	-0.04 [-0.20, +0.11]	+0.8 [-1.2, +2.8]	+1.7 [-0.9, +4.4]	14/36 (0.934)
Qwen3-TTS	0.00 (0.949)	+0.12 [-0.25, +0.49]	+2.1 [-2.8, +7.0]	+7.5 [+0.3, +15.3]	6/9 (0.254)
Kokoro-82M	+0.07 (0.063)	+0.25 [-0.18, +0.68]	+1.9 [-2.7, +6.6]	+1.1 [-6.7, +9.1]	2/7 (0.938)

Table 16: Open-text frequency, model-light analyses (w2v2p): Spearman correlation of log frequency with model–floor over cells; logistic regression with language-clustered errors; model-free slope of model–floor in points per decade of frequency (all cells, utterance-initial cells); and matched same-phone pairs in which the more frequent grapheme is better (sign test).

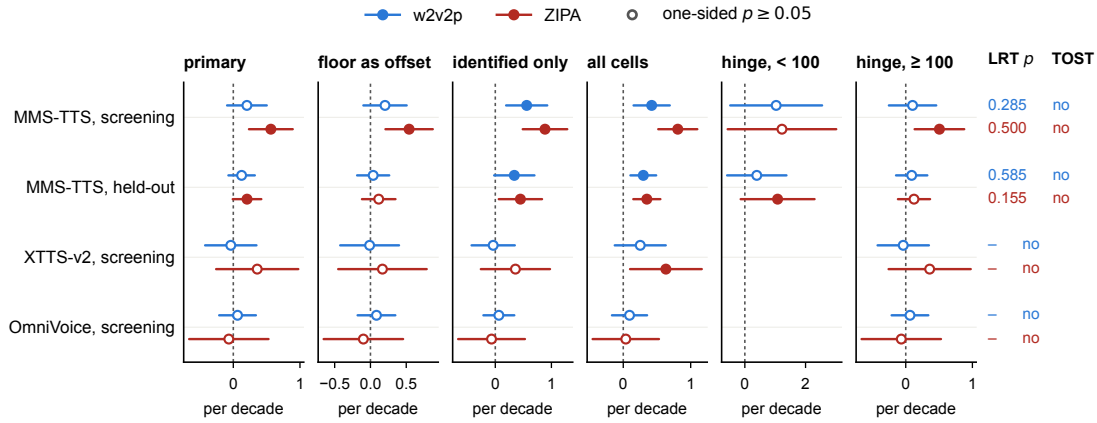


Figure 22: **Every specification of the training-count GLMM:** coefficient of $\log_{10}(1 + \text{count})$ (per decade of training-text count, 95% CI) for every model, word set and ruler: the primary model, the floor as an offset, identified translations only, all cells including unjudgeable ones, and the hinge model's slopes below and above 100 occurrences. Filled: one-sided $p < 0.05$. Right: the hinge likelihood-ratio test (LRT) and whether the TOST equivalence test of the slope above the knot (SESOI 0.2) declares it negligible.

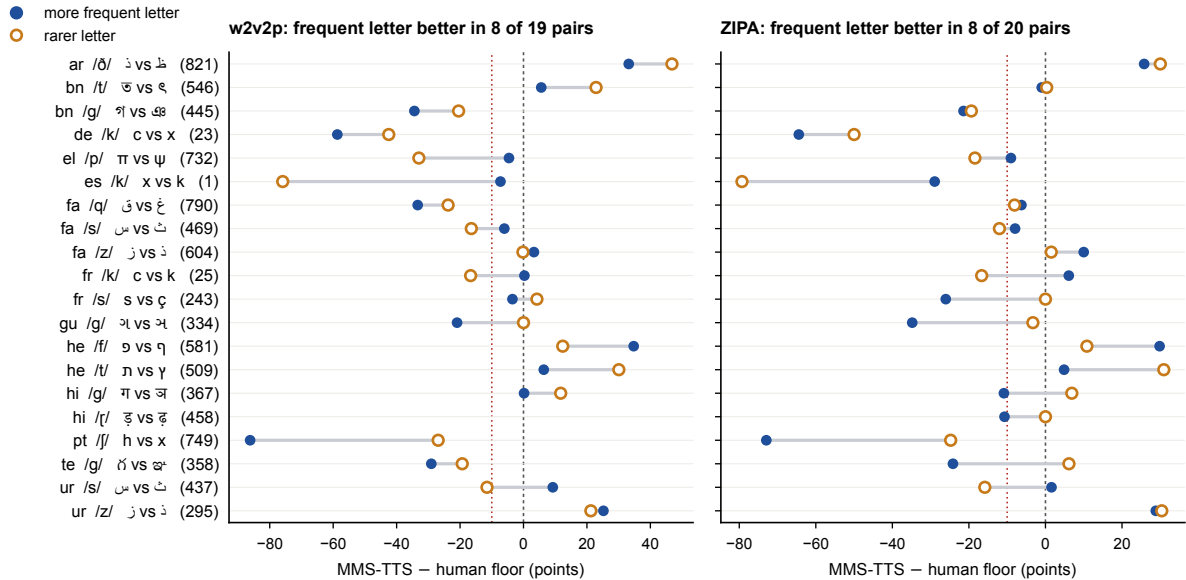


Figure 23: **Every pair of the pre-registered low-count same-sound test (E1-H).** Two letters that MMS-TTS should pronounce with the same phone, the rarer one seen fewer than 1,000 times in its training text (count in parentheses); model — floor of each on held-out reserve words (filled: more frequent letter; hollow: rarer letter; red dotted line: the -10-point failure margin). If rarity alone caused failures, the hollow dot would sit left of the filled one in most rows.

Analysis	Ruler	Term	Estimate (SE)	<i>p</i>
Joint human+MMS-TTS GLMM	w2v2p	log count (human speech)	+1.59 (0.08)	<0.001
Joint human+MMS-TTS GLMM	w2v2p	TTS vs human	−0.14 (0.06)	0.016
Joint human+MMS-TTS GLMM	w2v2p	TTS × log count	+0.48 (0.07)	<0.001
OmniVoice training hours, languages	w2v2p	Spearman ρ (29 langs)	−0.29	0.124
OmniVoice training hours, GLMM	w2v2p	log hours	−0.10 (0.10)	0.296
Joint human+MMS-TTS GLMM	ZIPA	log count (human speech)	+0.98 (0.06)	<0.001
Joint human+MMS-TTS GLMM	ZIPA	TTS vs human	−0.43 (0.05)	<0.001
Joint human+MMS-TTS GLMM	ZIPA	TTS × log count	+0.49 (0.05)	<0.001
OmniVoice training hours, languages	ZIPA	Spearman ρ (29 langs)	−0.05	0.798
OmniVoice training hours, GLMM	ZIPA	log hours	+0.15 (0.13)	0.254

Table 17: Secondary frequency analyses: the joint GLMM of human FLEURS speech and MMS-TTS with training-text counts (does TTS depend on counts more than human speech does: TTS × log count), and OmniVoice’s published per-language training hours against its excess failure.

Words	Model	Graph.	<10 ³	w2v2p		ZIPA	
				β [95% CI]	<i>p</i>	β [95% CI]	<i>p</i>
screening	MMS-TTS	476	8%	+0.20 [−0.09, +0.50]	0.089	+0.56 [+0.23, +0.89]	<0.001
	XTTS-v2	80	0%	−0.04 [−0.42, +0.35]	1.000	+0.36 [−0.26, +0.98]	0.508
	OmniVoice	127	3%	+0.06 [−0.21, +0.34]	0.977	−0.07 [−0.66, +0.53]	1.000
	Pooled (RE)	—	—	+0.09 [−0.09, +0.27]	0.155	+0.35 [−0.02, +0.72]	0.032
held-out	MMS-TTS	475	8%	+0.12 [−0.07, +0.32]	0.218	+0.21 [−0.01, +0.42]	0.094

Table 18: **Pre-registered training-count analyses** (E1-T, E1-T2, E1-R): coefficient of $\log_{10}(1 + \text{count})$ in the main model, per model and phone ruler, with 95% CI and one-sided *p* (Holm-adjusted within each pre-registered family; the E1-R family has four tests, R1 and R2 here and R3 and R4 in Appendix D.2). Graph.: letters; <10³: share of letters with fewer than 1,000 counted occurrences. Pooled (RE): random-effects estimate over the models.

Words	Ruler	Pairs	<i>n</i>	Freq. better	Sign <i>p</i>	Gap, pts [95% CI]
screening	w2v2p	all pairs	47	27	0.191	+3.1 [−2.6, +9.0]
screening	w2v2p	rare < 1000	19	11	0.324	+5.2 [−5.1, +16.2]
screening	ZIPA	all pairs	48	20	0.903	−0.8 [−7.0, +5.3]
screening	ZIPA	rare < 1000	20	10	0.588	+0.1 [−7.2, +8.1]
held-out	w2v2p	all pairs	47	21	0.809	−0.4 [−7.6, +6.7]
held-out	w2v2p	rare < 1000	19	8	0.820	−1.5 [−12.9, +10.5]
held-out	ZIPA	all pairs	48	19	0.944	−5.8 [−12.9, +1.1]
held-out	ZIPA	rare < 1000	20	8	0.868	−4.0 [−13.6, +6.0]

Table 19: **Two letters, one sound** (MMS-TTS, training-text counts): pairs in which the more frequent letter is closer to its floor, exact one-sided sign test, and mean gap (frequent − rare) in points with a bootstrap CI over pairs.

utterances), with learning rates 2×10^{-5} (standard) and 2×10^{-3} (new rows) and three epochs (2,457 updates); finetuning on one speaker lowers the plain-letter ceiling, which is why every comparison is paired within a checkpoint. A pilot that adapted F5-TTS to Georgian is not reported: it was probed through moving-average weights that barely change in a short finetune; all E2 results use the trained weights. Tables 24, 25 and 26 give the dose results of §5.3.

E2b rules. The seven starting rules are *random* and *mean* rows; a *copy* of the plain letter’s row; a PanPhon *blend* of the nearest letters’ rows (which includes that row); and three rules that never see a

row with the target sound: the average of the nearest letters with a different sound (*neighbours*), PRM (§6.2), and phonological *analogy*, $r(y) + r(z) - r(w)$ for letters whose sound features satisfy $f(y) + f(z) - f(w) \approx f(x)$. The random rows of E2b and the step-0 rows of E2c are different draws, so their untrained accuracies differ (Tables 27 and 26). Rules are compared with a sign-flip test over letters; Figure 24 and Tables 27 and 28 give the results of §5.4.

How to read the mechanism tables and figures. Tables 29 and 30 relate four row indicators to frequency and to failure (exploratory); Tables 31, 32 and 33 give the E2 GLMM, the targeted row tuning

Model	Lang	Text	Primary/id.	Letters	Graphemes	Zero	Median	Min	<10 ³ %
MMS-TTS	ar	arbnv	no	420,458	27	0	10,933	821	4
MMS-TTS	bn	benobcv	yes	629,673	36	0	4,919	44	25
MMS-TTS	de	deu1912	yes	636,712	21	0	12,445	23	10
MMS-TTS	el	christos-c Greek	no	614,949	18	0	15,009	732	6
MMS-TTS	es	spablm	yes	587,626	22	1	7,600	0	14
MMS-TTS	fa	pesOPV	yes	459,557	31	0	6,347	24	19
MMS-TTS	fr	fraLSG	yes	627,862	21	1	8,933	0	14
MMS-TTS	gu	guj2017	yes	565,066	34	1	5,401	0	15
MMS-TTS	he	heb	yes	348,461	26	0	10,533	509	8
MMS-TTS	hi	hin2017	no	547,745	40	2	2,460	0	28
MMS-TTS	kn	kan2017	yes	683,174	34	1	4,496	0	26
MMS-TTS	ml	malc	yes	772,140	36	1	6,110	0	25
MMS-TTS	mr	mar	no	641,249	34	1	5,185	0	15
MMS-TTS	pa	pan	yes	568,509	32	2	5,038	0	25
MMS-TTS	pl	polubg	yes	567,156	23	0	13,015	549	13
MMS-TTS	pt	porbr2018	yes	581,355	19	0	9,776	749	5
MMS-TTS	ru	russyn	yes	500,631	21	0	10,799	411	5
MMS-TTS	sw	swhonen	yes	637,409	18	0	13,971	3,684	0
MMS-TTS	ta	tam2017	no	801,247	22	1	19,577	0	18
MMS-TTS	te	tel2017	yes	615,087	35	3	2,940	0	31
MMS-TTS	tr	turytc	no	573,112	21	0	10,427	140	5
MMS-TTS	uk	ukronpu	yes	512,052	22	0	11,794	11	9
MMS-TTS	ur	urdgvu	no	582,276	34	0	4,403	17	18
MMS-TTS	vi	vieovcb	yes	543,513	17	0	9,171	1,707	0
XTTS-v2	ar	cv15	no	2,268,756	27	0	43,121	4,493	0

Table 20: Training-text counts per model and language (E1-T, E1-T2): counted text (MMS-TTS: eBible New Testament edition; XTTS-v2: Common Voice 15 validated transcripts; OmniVoice: Common Voice 15 + FLEURS train), whether the language is primary (E1-T2) or its translation identified (E1-T), letters counted, graphemes, graphemes never seen, median and minimum count, share below 10³.

Model	Lang	Text	Primary/id.	Letters	Graphemes	Zero	Median	Min	<10 ³ %
XTTS-v2	cs	cv15	yes	2,065,677	24	0	47,453	810	8
XTTS-v2	de	cv15	no	46,803,721	21	0	945,360	13,049	0
XTTS-v2	es	cv15	no	18,240,080	22	0	206,845	12,003	0
XTTS-v2	fr	cv15	no	33,212,598	21	0	397,483	21,080	0
XTTS-v2	it	cv15	no	12,266,689	16	0	327,377	48,041	0
XTTS-v2	pl	cv15	yes	5,166,117	23	0	133,237	3,878	0
XTTS-v2	pt	cv15	no	4,654,388	19	0	77,822	10,925	0
XTTS-v2	ru	cv15	yes	8,066,659	21	0	131,897	19,972	0
XTTS-v2	tr	cv15	yes	2,910,275	21	0	49,803	1,107	0
OmniVoice	ar	cv15+fleurs-train	no	2,450,830	27	0	47,286	4,917	0
OmniVoice	cs	cv15+fleurs-train	no	2,348,397	24	0	53,903	861	8
OmniVoice	de	cv15+fleurs-train	no	47,165,514	21	0	952,479	13,160	0
OmniVoice	es	cv15+fleurs-train	no	18,578,206	22	0	210,729	12,194	0
OmniVoice	fa	cv15+fleurs-train	yes	8,625,403	31	0	111,567	4,892	0
OmniVoice	fr	cv15+fleurs-train	no	33,594,499	21	0	401,970	21,225	0
OmniVoice	he	cv15+fleurs-train	yes	254,722	26	0	6,139	284	8
OmniVoice	hi	cv15+fleurs-train	yes	562,610	40	1	3,177	0	28
OmniVoice	it	cv15+fleurs-train	no	12,628,386	16	0	337,145	49,440	0
OmniVoice	pl	cv15+fleurs-train	no	5,478,344	23	0	137,674	4,103	0
OmniVoice	pt	cv15+fleurs-train	no	4,960,036	19	0	82,275	11,700	0
OmniVoice	ru	cv15+fleurs-train	no	8,358,790	21	0	136,768	20,948	0
OmniVoice	sw	cv15+fleurs-train	yes	12,849,613	18	0	286,180	77,488	0
OmniVoice	ta	cv15+fleurs-train	yes	7,688,813	22	0	179,333	1,970	0
OmniVoice	tr	cv15+fleurs-train	yes	3,180,809	21	0	53,637	1,409	0

Table 21: Training-text counts per model and language (E1-T, E1-T2): counted text (MMS-TTS: eBible New Testament edition; XTTS-v2: Common Voice 15 validated transcripts; OmniVoice: Common Voice 15 + FLEURS train), whether the language is primary (E1-T2) or its translation identified (E1-T), letters counted, graphemes, graphemes never seen, median and minimum count, share below 10³ (continued).

Model	Words	Ruler	0	1–9	10–99	100–999	10 ³ –10 ⁴	≥ 10 ⁴
MMS-TTS	screening	w2v2p	–53.4 (1)	–17.4 (3)	–56.5 (3)	–13.0 (29)	–6.4 (169)	–2.7 (271)
MMS-TTS	screening	ZIPA	–80.2 (1)	–40.8 (2)	–37.0 (3)	–22.8 (35)	–14.7 (186)	–7.8 (256)
MMS-TTS	held-out	w2v2p	–55.1 (1)	–20.8 (3)	–44.5 (3)	–12.4 (29)	–9.9 (168)	–2.4 (271)
MMS-TTS	held-out	ZIPA	–81.9 (1)	–47.8 (2)	–29.3 (3)	–19.7 (35)	–15.5 (185)	–8.4 (256)
XTTS-v2	screening	w2v2p	–	–	–	–	+2.1 (6)	+2.1 (74)
XTTS-v2	screening	ZIPA	–	–	–	–	–1.3 (6)	–3.4 (75)
OmniVoice	screening	w2v2p	–	–	–	+7.7 (4)	+8.0 (25)	+8.7 (98)
OmniVoice	screening	ZIPA	–	–	–	+14.2 (4)	+4.8 (32)	+2.8 (96)

Table 22: Excess failure (model–floor, points; graphemes in parentheses) by training-count bin, for every model, word set and ruler of the training-count analyses.

Model	Words	Ruler	Pairs	Frequent better	Sign <i>p</i>	Mean gap
MMS-TTS	screening	w2v2p	47	27	0.191	+3.1
IndicF5	screening	w2v2p	13	7	0.500	+4.9
Chatterbox-MTL	screening	w2v2p	22	10	0.738	–0.1
XTTS-v2	screening	w2v2p	14	8	0.395	+3.1
OmniVoice	screening	w2v2p	51	22	0.869	–2.0
Qwen3-TTS	screening	w2v2p	11	6	0.500	+0.2
Kokoro-82M	screening	w2v2p	10	5	0.623	+5.1
MMS-TTS	screening	ZIPA	48	20	0.903	–0.8
IndicF5	screening	ZIPA	14	6	0.788	+4.0
Chatterbox-MTL	screening	ZIPA	24	9	0.924	–5.4
XTTS-v2	screening	ZIPA	15	2	1.000	–9.1
OmniVoice	screening	ZIPA	53	23	0.864	–0.8
Qwen3-TTS	screening	ZIPA	11	3	0.967	–7.0
Kokoro-82M	screening	ZIPA	11	3	0.967	–2.6
MMS-TTS	held-out	w2v2p	47	21	0.809	–0.4
XTTS-v2	held-out	w2v2p	14	7	0.605	–0.7
MMS-TTS	held-out	ZIPA	48	19	0.944	–5.8
XTTS-v2	held-out	ZIPA	15	6	0.849	–5.7

Table 23: Same-sound grapheme pairs for every model, word set and ruler (frequency: training-text counts for MMS-TTS, Wikipedia otherwise).

Init	Step 0	dose 0	dose 10	dose 40	dose 160	Page’s <i>L p</i>
Random	–88.2	–91.0 [–95.5, –85.5]	–91.0 [–95.4, –85.4]	–88.3 [–93.5, –82.9]	–79.6 [–86.7, –71.1]	0.014
Random, emb. lr 2×10^{-3}	–88.2	–90.3 [–95.3, –84.3]	–88.0 [–93.3, –82.3]	–67.4 [–79.3, –54.6]	–34.0 [–46.4, –21.9]	9.3×10^{-10}
Random, new rows frozen (control)	–88.2	–91.0 [–97.0, –83.6]	–91.2 [–97.5, –82.6]	–85.6 [–95.6, –74.5]	–80.1 [–90.7, –68.8]	0.04

Table 24: E2 (F5-TTS): accuracy gap between a new input row and the plain letter’s row for the same sound (paired, same words and takes), by training dose and arm (random starting rows).

Init	dose 0	dose 10	dose 40	dose 160	dist. to plain
Random	0.054	0.081	0.100	0.131	19.9
Random, emb. lr 2×10^{-3}	0.000	1.643	2.120	3.015	19.7
Random, new rows frozen (control)	0.000	0.000	0.000	0.000	20.0

Table 25: E2: median distance a new row moved from its starting value by the last checkpoint, by dose, and its distance to the plain letter’s row.

and E2c under ZIPA. Figure 25 shows both dose studies letter by letter (columns: 0 to 160 training occurrences, then the plain letter as the ceiling), and Figure 26 every E2b rule per model and language (a good rule is nearly as dark as the plain letter); Table 34 tests the rules pairwise.

Chapter D in brief

Open-text frequency predicts accuracy weakly, mainly in MMS-TTS. Training-text counts predict MMS-TTS accuracy under ZIPA: the rarest letters fail most, with no change of slope at 100 occurrences. XTTS-v2 and OmniVoice show no association, no primary pre-registered replication test survives Holm correction, and same-sound pairs

Arm	Step 0	$d = 0$	$d = 10$	$d = 40$	$d = 160$	Page p
Standard (all weights, $\text{lr } 2 \times 10^{-5}$)	-74.1	-55.3 [-65.7, -44.7]	-56.0 [-66.2, -45.1]	-51.4 [-62.5, -40.0]	-45.8 [-59.7, -31.7]	0.089
New rows $\text{lr } 2 \times 10^{-3}$, old rows frozen	-74.1	-56.2 [-66.2, -45.4]	-15.0 [-25.0, -6.5]	-9.3 [-14.6, -4.2]	-2.8 [-6.5, +1.2]	<0.001
plain letter (ceiling), standard arm		66.9	68.3	67.4	67.1	
plain letter (ceiling), new-row arm		69.2	66.4	70.6	68.1	

Table 26: E2c (MMS-TTS English, VITS): accuracy gap between a new input row and the plain letter’s row for the same sound (strict, w2v2p, points; paired on the same words and takes; 95% CI over letter $\times$ run), by dose and arm; step 0 is the model before training. Page: ordered trend test over doses. Last rows: plain-letter accuracy (%) of the same checkpoints.

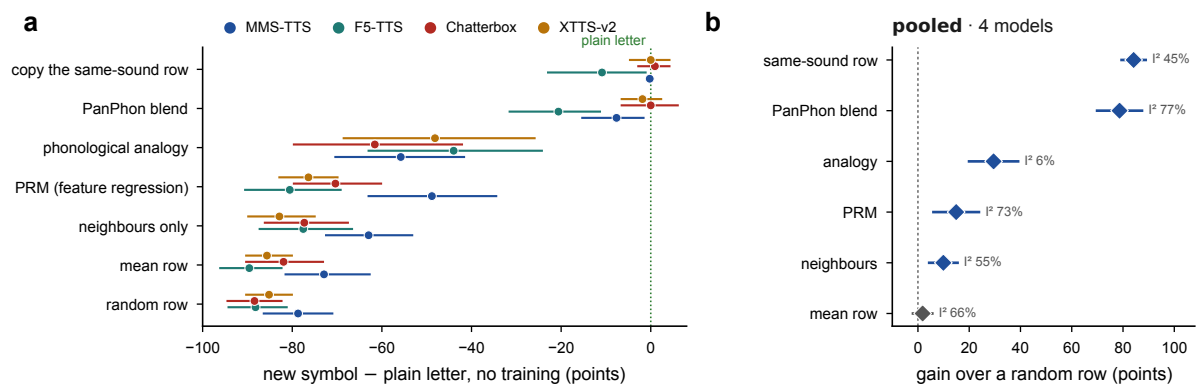


Figure 24: **Building a row for a new symbol, before any training (E2b).** (a) Gap between the new symbol and the plain letter for each way of building the row, per model (95% CIs over letters). (b) Gain of each method over a random row, pooled over the four models (diamonds; blue: the interval excludes zero), with heterogeneity I^2 .

Initialisation	MMS-TTS (VITS)	F5-TTS (flow DiT)	Chatterbox (AR LM)	XTTS-v2 (AR GPT)
plain letter (ceiling)	84.0	92.1	93.1	93.1
copy same-sound row	83.8	81.2	94.0	93.1
PanPhon blend	76.4	71.5	93.1	91.2
PRM (no same-sound row)	35.2	11.6	22.7	16.7
analogy	28.2	48.1	31.5	44.9
neighbours only	21.1	14.6	15.7	10.2
mean row	11.1	2.5	11.1	7.4
random	5.3	3.9	4.6	7.9

Table 27: E2b: strict accuracy (%) of a new symbol whose row is set by each rule, with no training, for the twelve English letters; the first row is the plain letter.

Initialisation vs random	Pooled gain (pts) [95% CI]	p	I^2	all 4 > 0
analogy	+29.5 [+19.9, +39.1]	7×10^{-10}	6%	yes
PRM	+14.9 [+6.0, +23.8]	0.0005	73%	yes
neighbours	+9.9 [+4.4, +15.4]	0.0002	55%	yes
PanPhon blend	+78.6 [+69.9, +87.4]	<0.0001	77%	yes
copy same-sound row	+84.1 [+79.4, +88.8]	<0.0001	45%	yes
mean row	+1.8 [-1.9, +5.6]	0.2	66%	no

Table 28: E2b pooled over the four models: gain of each rule over a random row (random-effects meta-analysis of per-model letter-level differences; one-sided p).

show no frequency advantage. The released rows carry a trace of their training frequency but predict failure only weakly. In controlled finetuning, a new symbol is learned well only when its row gets a strong training signal, and then improves with its

count; a row composed by phonological analogy starts far above a random one.

Questions this chapter leaves open. Whether the controlled dose response holds for existing letters of released models (E7, pre-registered, was stopped

Model	Units	NN cosine		Norm (z)		Cos. special		Phon. ratio	
		log freq	model–floor	log freq	model–floor	log freq	model–floor	log freq	model–floor
MMS-TTS	602	−0.58*	+0.09*	−0.18*	−0.06	–	–	+0.18*	−0.16*
IndicF5	271	+0.07	−0.10	−0.21*	−0.06	+0.12*	+0.04	0.00	+0.02
Chatterbox-MTL	268	+0.22*	−0.17*	−0.16*	−0.11	−0.03	−0.05	+0.21*	−0.03
XTTS-v2	241	+0.23*	−0.16	−0.53*	+0.12	+0.28*	−0.15	−0.23*	+0.08
OmniVoice	617	+0.10	−0.17*	+0.36*	−0.05	+0.31*	+0.01	+0.14*	−0.05

Table 29: Embedding-row geometry: Spearman ρ of each indicator with log open-text frequency and with the unit’s mean model–floor (*: 95% bootstrap CI over languages excludes 0). Exploratory, not pre-registered.

Model	Units	Failing	NN cosine	Norm (z)	Cos. special	Phon. ratio
MMS-TTS	602	173	+0.13 / +0.15 (<0.001)	−0.28 / −0.37 (0.561)	–	+0.62 / +0.52 (<0.001)
IndicF5	271	25	+0.14 / +0.14 (0.360)	+0.35 / +0.43 (0.569)	0.00 / −0.01 (0.960)	+0.90 / +0.89 (0.756)
Chatterbox-MTL	268	41	+0.27 / +0.29 (0.071)	+1.05 / +1.07 (0.789)	0.00 / +0.00 (0.528)	+0.84 / +0.74 (0.003)
XTTS-v2	241	40	+0.23 / +0.25 (0.219)	−2.58 / −2.87 (0.474)	+0.04 / +0.05 (0.517)	+0.89 / +0.77 (0.021)
OmniVoice	617	57	+0.35 / +0.35 (0.546)	+1.42 / +1.36 (0.196)	+0.01 / +0.01 (0.546)	+0.80 / +0.73 (0.016)

Table 30: Embedding-row indicators of failing versus non-failing graphemes (mean, failing / non-failing; two-sided Mann–Whitney p). Exploratory.

Initialisation	n	any dose: est. (SE) p	$\log_4(d/10)$: est. (SE) p
random	3456	−0.05 (0.28) 0.858	+1.10 (0.30) <0.001
randomfz	1728	+0.07 (0.54) 0.902	+1.35 (0.50) 0.006
randomhi	3456	+1.20 (0.27) <0.001	+2.23 (0.25) <0.001

Table 31: E2 GLMM per initialisation at the final checkpoint: effect of any training dose and of log dose on new-symbol accuracy, with the plain letter as offset.

init	n	run	dose0	trt	plain	gain	gap_after
random	10	en_r0_s0	3.7	2.8	95.4	−0.9	−92.6
random	40	en_r0_s0	3.7	2.8	95.4	−0.9	−92.6
randomhi	10	en_r0_s0	1.9	2.8	98.1	0.9	−95.4
randomhi	40	en_r0_s0	1.9	2.8	98.1	0.9	−95.4
random	10	en_r1_s0	0.9	0.9	100.0	0.0	−99.1
random	40	en_r1_s0	0.9	0.0	100.0	−0.9	−100.0
randomhi	10	en_r1_s0	0.0	0.9	98.1	0.9	−97.2
randomhi	40	en_r1_s0	0.0	1.9	98.1	1.9	−96.3
random	10	en_r2_s0	2.8	4.6	99.1	1.9	−94.4
random	40	en_r2_s0	2.8	11.1	99.1	8.3	−88.0
randomhi	10	en_r2_s0	4.6	9.3	99.1	4.6	−89.8
randomhi	40	en_r2_s0	4.6	11.1	99.1	6.5	−88.0
random	10	en_r3_s0	1.9	2.8	81.5	0.9	−78.7
random	40	en_r3_s0	1.9	1.9	81.5	0.0	−79.6
randomhi	10	en_r3_s0	1.9	4.6	79.6	2.8	−75.0
randomhi	40	en_r3_s0	1.9	5.6	79.6	3.7	−74.1

Table 32: Targeted Row Tuning on E2 checkpoints: dose-0 letters before and after tuning only their new rows on n withheld utterances (strict, %).

Arm	Step 0	$d = 0$	$d = 10$	$d = 40$	$d = 160$	Page p
Standard (all weights, $lr\ 2 \times 10^{-5}$)	−75.7	−55.1 [−66.9, −41.7]	−56.5 [−69.0, −42.4]	−51.2 [−61.6, −40.0]	−45.1 [−57.4, −33.6]	0.009
New rows $lr\ 2 \times 10^{-3}$, old rows frozen	−75.7	−56.7 [−66.4, −45.6]	−12.5 [−19.0, −6.7]	−6.5 [−10.0, −3.0]	−2.3 [−6.0, +1.6]	<0.001
plain letter (ceiling), standard arm		68.8	69.7	68.1	68.5	
plain letter (ceiling), new-row arm		69.7	67.4	71.3	71.1	

Table 33: E2c under the ZIPA ruler (as Table 26).

a F5-TTS: standard						b F5-TTS: new rows, lr 2e-3						c F5-TTS: new rows frozen (control)					
b	0	1	10	4	100	0	1	0	69	99	0	0	3	0	99		
d	6	7	14	4	75	7	8	29	61	75	8	17	11	8	77		
f	6	6	17	67	100	12	12	49	81	100	8	3	61	8	100		
k	0	0	0	8	100	0	7	47	86	99	0	0	0	42	100		
l	3	1	3	7	97	1	1	6	14	98	3	0	0	14	97		
m	0	0	4	29	100	0	17	15	83	100	6	6	3	17	99		
n	0	6	8	12	97	4	11	14	83	97	0	3	8	39	97		
p	1	1	0	4	99	0	4	35	42	99	0	0	0	0	98		
r	10	6	7	7	96	6	6	67	75	96	3	6	8	0	98		
t	1	3	6	10	64	4	7	31	46	64	0	6	3	28	65		
v	3	3	0	3	99	1	0	22	40	99	0	0	0	0	97		
w	1	0	0	14	99	0	0	6	35	99	0	0	0	11	99		
	d=0	d=10	d=40	d=160	plain	d=0	d=10	d=40	d=160	plain	d=0	d=10	d=40	d=160	plain		

d VITS: untrained		e VITS: standard						f VITS: new rows, lr 2e-3					
b	6	81	8	14	17	14	28	17	25	36	36	32	
d	44	69	36	31	33	58	66	31	28	67	61	66	
f	11	97	6	8	28	33	87	6	81	81	89	88	
k	0	69	6	6	8	19	65	14	53	61	69	64	
l	11	94	3	3	0	0	84	6	44	58	72	87	
m	22	97	6	3	14	14	69	3	61	69	69	73	
n	3	97	8	6	0	8	67	3	47	50	58	65	
p	6	75	19	17	28	19	64	19	42	56	61	65	
r	6	92	3	8	14	11	81	8	89	83	83	85	
t	0	69	22	28	25	36	72	28	50	67	64	78	
v	11	78	14	19	22	22	53	17	47	53	53	50	
w	0	89	8	6	3	19	73	6	50	56	67	69	
	step 0	plain	d=0	d=10	d=40	d=160	plain	d=0	d=10	d=40	d=160	plain	

Figure 25: **E2 (F5-TTS) and E2c (VITS) per letter:** strict accuracy (% w2v2p) of each new symbol by training dose d (occurrences of the symbol in the finetuning data), and of the plain letter for the same sound (last column, right of the line). F5-TTS: final checkpoint, pooled over seeds; (d) the VITS model before finetuning. A row that darkens from left to right is a letter learned with its count; a pale row next to a dark plain cell is a letter whose sound the model knows but whose new row it cannot use.

a four models, 12 English letters					b MMS-TTS in seven more languages, 12 letters each						
	MMS-TTS	F5-TTS	Chatterbox	XTTS-v2	ar	bn	el	fa	hi	ru	ta
plain letter (ceiling)	84	92	93	93	72	32	75	76	25	56	60
copy same-sound row	84	81	94	93	72	32	75	76	25	56	60
PanPhon blend	76	72	93	91	59	28	53	70	19	58	58
PRM	35	12	23	17	40	29	27	39	18	41	35
phonological analogy	28	48	31	45	11	12	19	11	4	14	21
neighbours only	21	15	16	10	11	12	10	7	5	15	31
mean row	11	3	11	7	7	4	5	13	2	5	12
random row	5	4	5	8	2	4	8	8	1	4	9

Figure 26: **E2b for every model, rule and language:** strict accuracy (% w2v2p) of a new symbol whose row is set by each rule, before any training; the first row is the plain letter for the same sound. (a) The four models on the twelve English letters (the paired gaps with 95% CIs are Figure 24a and the released newsym.csv). (b) The same rules in MMS-TTS in seven more languages.

Model	Contrast	Letters	Diff.	Letters better	p (one-sided)
Chatterbox-MTL	prm – random	12	+18.1	8	0.004
Chatterbox-MTL	prm – mean	12	+11.6	9	0.016
Chatterbox-MTL	prm – nbr	12	+6.9	7	0.038
Chatterbox-MTL	analogy – random	12	+26.9	7	0.024
Chatterbox-MTL	blend – random	12	+88.4	12	<0.001
F5-TTS	prm – random	12	+7.6	7	0.012
F5-TTS	prm – mean	12	+9.0	8	0.003
F5-TTS	prm – nbr	12	–3.0	3	0.887
F5-TTS	analogy – random	12	+44.2	11	<0.001
F5-TTS	blend – random	12	+67.6	12	<0.001
MMS-TTS	prm – random	12	+29.9	11	<0.001
MMS-TTS	prm – mean	12	+24.1	10	0.002
MMS-TTS	prm – nbr	12	+14.1	11	0.002
MMS-TTS	analogy – random	12	+22.9	10	<0.001
MMS-TTS	blend – random	12	+71.1	12	<0.001
XTTS-v2	prm – random	12	+8.8	7	0.037
XTTS-v2	prm – mean	12	+9.3	6	0.017
XTTS-v2	prm – nbr	12	+6.5	7	0.137
XTTS-v2	analogy – random	12	+37.0	8	0.013
XTTS-v2	blend – random	12	+83.3	12	<0.001

Table 34: E2b pairwise tests (sign-flip permutation over letters).

before it produced data), and in architectures beyond VITS and F5-TTS.

E Can It Be Repaired: Repairs in Full

The repairs of §6: the onset repair under every cut rule, the gated row edits, why row-only training harmed (E4), the self-sample row repair and its gate (E5'), best-of- N selection (E6), and the pre-registered reset-and-relearn test that was stopped (E7).

In this chapter

E.1 Onset repair: cut rules	41
E.2 Gated row edits (E3)	41
E.3 Why row-only training of released models harmed (E4)	41
E.4 Self-sample row repair and its gate (E5')	43
E.5 Best-of- N selection (E6)	43
E.6 Reset and relearn (E7, stopped)	45
E.7 What each intervention gains	46

E.1 Onset repair: cut rules

1401

Figure 27 and Table 35 give the pre-registered onset remedies of §6.1 per model and pooled. OmniVoice lengthens short utterances by a duration rule, so for it the requested duration is the boosted duration of the sentence plus the raw duration of the added text; clips the aligner cannot place count as failures.

The lead-in clip is cut just before the probe word. Table 38 varies the rule: halfway between the words but at least 30 ms before the aligned onset (the rule we first used), at the aligned end of the lead-in's last word, and at least 80 ms before the onset, always with a 10 ms fade-in. In fast speech the forced aligner can place a word's onset late, so a tight margin clips the target consonant; all rules are reported, each on the languages where it was run (in parentheses). Table 37 gives the lead-in (W1) under the first rule in every E1 language where it was run. Under that rule OmniVoice loses accuracy; compare the 80 ms rule used for the pre-registered remedies in Table 38.

How to read the onset-repair figures and tables.

Figure 28 gives the lead-in repair under the first cut rule for every model and language where it was run: panel (a) is the gain of the lead-in clip, panel (b) the trim control (the original clip cut by the same rule), which should stay near zero if the gain comes from the lead-in itself. Table 39 splits the gain by consonant position: an onset mechanism predicts the largest gain word-initially and little elsewhere. Table 40 lists every cut rule with its clip counts. Figure 29 breaks the pre-registered remedies of Figure 27 down by language: rows are remedy and model, columns are the six languages, under both rulers.

E.2 Gated row edits (E3)

Table 41 summarises E3 per model. PRM fits a kernel ridge regression with an RBF kernel on Pan-Phon features, $\hat{r}(g) = \mu + \sum_a k(\phi_g, \phi_a) \alpha_a$, on the language's working letters (width and ridge chosen by leave-one-out cosine, without audio), and the repaired row is $\alpha \hat{r}(g) + (1 - \alpha) r_{\text{old}}(g)$ with $\alpha \in \{0.5, 1\}$. Selecting with the evaluation ruler itself (the phone recogniser on reserve words) gives +1.10 [−0.02, +2.34] points; with the Whisper gate, 24 letters improve and 23 get worse.

Figure 30 shows every failing grapheme of E3 as one dot: its accuracy with the released row (x) and after the edit the Whisper gate chose on reserve words (y). Dots above the diagonal improved and dots below got worse; grey dots sit on the diagonal because the gate kept the released row. An edit counts as useful only if it beats keeping the released row on these disjoint screening words.

E.3 Why row-only training of released models harmed (E4)

E4's verdict is the pre-registered one; this analysis came afterwards and is exploratory. It uses only finished E4 outputs: the trained rows, the released embedding tables (read from each checkpoint without downloading the model) and the score files (longtail/e4_rootcause.py).

1. **One loss reaches the rows.** In VITS training the decoder synthesises from the posterior latent of the recording, so the reconstruction, adversarial and feature-matching losses never pass through the text encoder; both duration predictors detach their input. In all 16 runs the mel loss stayed flat while the KL term fell by 27–45%.

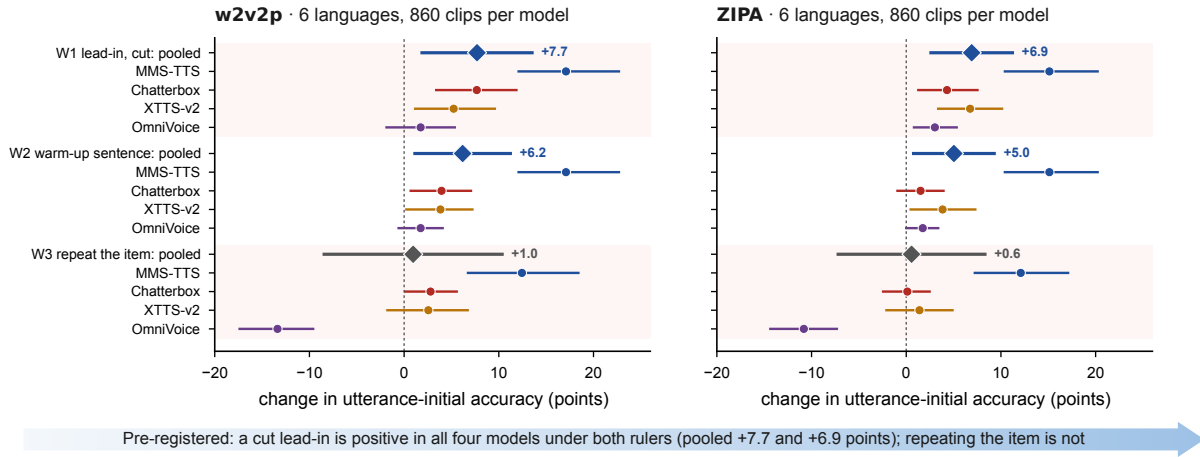


Figure 27: **Onset remedies (E3-O, pre-registered)**. Change in accuracy of word-initial consonants of words that open the utterance under each remedy, per model and pooled over the four models (diamonds; blue: the pooled interval excludes zero), under both phone rulers (95% CIs over letters; numbers in Table 35).

Remedy	Model	w2v2p		ZIPA	
		Δ [95% CI]	p_{Holm}	Δ [95% CI]	p_{Holm}
W1 lead-in	MMS-TTS	+17.1 [+12.1, +22.7]	<0.001	+15.1 [+10.4, +20.2]	<0.001
	XTTS-v2	+5.2 [+1.2, +9.6]	0.015	+6.7 [+3.4, +10.1]	<0.001
	Chatterbox-MTL	+7.7 [+3.4, +11.9]	<0.001	+4.3 [+1.3, +7.5]	0.007
	OmniVoice	+1.7 [-1.9, +5.3]	0.172	+3.0 [+0.8, +5.3]	0.007
	Pooled (RE)	+7.7 [+1.9, +13.5]	4 models	+6.9 [+2.6, +11.2]	4 models
W2 warm-up sentence	MMS-TTS	+17.1 [+12.1, +22.7]	<0.001	+15.1 [+10.4, +20.2]	<0.001
	XTTS-v2	+3.8 [+0.2, +7.2]	0.033	+3.8 [+0.5, +7.3]	0.037
	Chatterbox-MTL	+4.0 [+0.7, +7.1]	0.024	+1.5 [-0.9, +3.9]	0.109
	OmniVoice	+1.7 [-0.6, +4.1]	0.077	+1.7 [+0.0, +3.4]	0.048
	Pooled (RE)	+6.2 [+1.1, +11.2]	4 models	+5.0 [+0.8, +9.3]	4 models
W3 repeat	MMS-TTS	+12.4 [+6.7, +18.4]	<0.001	+12.1 [+7.2, +17.1]	<0.001
	XTTS-v2	+2.6 [-1.8, +6.7]	0.241	+1.4 [-2.1, +4.9]	0.646
	Chatterbox-MTL	+2.8 [+0.1, +5.6]	0.065	+0.1 [-2.4, +2.5]	0.928
	OmniVoice	-13.4 [-17.4, -9.6]	1.000	-10.8 [-14.4, -7.3]	1.000
	Pooled (RE)	+1.0 [-8.4, +10.4]	4 models	+0.6 [-7.2, +8.3]	4 models

Table 35: Pre-registered onset remedies (E3-O): change in strict accuracy of word-initial consonants of words that open the utterance, remedy clip minus original clip of the same item and take (points; 95% CI over letters; one-sided p , Holm over models within a remedy). Pooled (RE): random-effects estimate over the models. Six languages (es, fr, de, ru, ar, hi), 80 ms cut.

- The target encodes other speakers.** The KL pulls the text prior towards the latents of FLEURS recordings (many speakers and microphones) computed by a posterior encoder and flow trained on one MMS speaker. With every other weight frozen, the target rows were the only free parameters. Their updates share a letter-independent direction: cosine of each update with its language’s mean update 0.33–0.61 (independent directions in 192 dimensions would give about 0).
- No anchor.** The rows moved 73–101% (10 utterances) and 96–143% (40) of their own

- length, comparable to the distance between two letters’ rows; afterwards only 0–15% were still closest to their released value. The letters that were right most often lost the most ($\rho = -0.82$ between released accuracy and change); within a dose, the distance moved did not predict the loss, which points to the direction of the update rather than its size.
- Timing broke at inference.** The frozen duration predictor reads the moved rows (and, through the encoder’s self-attention, their neighbours) without having been trained on them: deletions of the target letter rose from

Model	Langs	Random take	Selected	Gain [95% CI]	Holm p	Sel.—placebo	Oracle
MMS-TTS	24	56.6	57.3	+0.8 [+0.5, +1.0]	<0.001	+0.7	61.5
Kokoro-82M	5	82.5	82.5	+0.0 [0.0, +0.0]	0.500	0.0	82.8
OmniVoice	29	68.0	68.2	+0.2 [−0.1, +0.5]	0.124	+0.1	71.8
Qwen3-TTS	6	85.2	85.8	+0.7 [+0.1, +1.2]	0.062	+0.3	88.1
Chatterbox-MTL	13	73.6	74.5	+1.0 [+0.4, +1.6]	0.007	+0.3	79.8
IndicF5	9	50.2	50.8	+0.7 [+0.1, +1.3]	0.039	+0.4	54.0
XTTS-v2	11	72.6	73.1	+0.5 [−0.3, +1.5]	0.236	+0.1	78.8

Table 36: **Best-of-two selection (E6)**: strict accuracy (%) of a random take, of the take selected by ZIPA’s whole-utterance phone error (graded by w2v2p), the gain with a two-level bootstrap CI over languages and graphemes, Holm-adjusted sign-flip p over languages, the gain over a selector blind to the probed word, and the oracle.

Model	Langs	E1 init	Lead-in	Δ [95% CI]	Trim ctrl	Gap rec.
MMS-TTS	23	46.9	59.8	+13.0 [+10.4, +15.5]	47.2	73%
Chatterbox-MTL	11	65.2	67.8	+2.6 [−0.1, +5.2]	65.7	25%
XTTS-v2	11	62.6	66.4	+3.8 [+0.2, +7.2]	61.3	32%
OmniVoice	28	66.1	59.8	−6.2 [−8.0, −4.5]	66.3	−129%

Table 37: **Lead-in (W1) under the first cut rule** (at least 30 ms before the aligned onset): word-initial consonants of utterance-initial probe words (strict, w2v2p), E1 clip versus the lead-in clip. Trim ctrl: the E1 clip cut by the same rule. Gap rec.: share of the E1 gap to the embedded frame that the lead-in closes.

Model	30 ms margin	end of lead word	80 ms margin
MMS-TTS	+13.0 [+10.4, +15.5] (23)	+15.5 [+12.8, +18.1] (23)	+16.8 [+14.2, +19.4] (23)
Chatterbox-MTL	+2.6 [−0.1, +5.2] (11)	+7.4 [+4.6, +10.1] (11)	+5.6 [+2.9, +8.4] (12)
XTTS-v2	+3.8 [+0.2, +7.2] (11)	+6.2 [+2.7, +9.7] (11)	+7.2 [+3.9, +10.4] (11)
OmniVoice	−6.2 [−8.0, −4.5] (28)	−3.0 [−4.6, −1.3] (28)	+1.3 [−0.2, +2.7] (28)

Table 38: **Onset repair under three cut rules**: change in strict accuracy of word-initial consonants (points, 95% CI; languages).

9.3% to 32.2% of judgements, clips became shorter in 7 of 8 languages, and whole-utterance phone error rose from 0.40 to 0.76. Because the carrier sentences of every language contain target letters, no clip was spared.

Ruled out: a masking or weight-decay bug (frozen rows are bit-identical to the release), wrong row ids, an error in merging the rows (the released baseline reproduces E1-R in 5,413 of 5,424 clips), and the loss weights (the discriminator sends no gradient to the rows, and Adam’s step does not depend on the weights’ scale).

E.4 Self-sample row repair and its gate (E5’)

Design. Pre-registered after E4 and before any of its data (§6.3): the 168 failing graphemes of all 24 languages that met E4’s qualification rule, 16 own takes per FLEURS-train sentence (sentences with a dev or test word removed), and only the failing rows trained for 400 updates on the selected takes’ own latents and durations, with an anchor on the other tokens and a trust region of a quarter of each row’s norm. Replicate Z selects and gates with ZIPA and

is tested with w2v2p, replicate W the reverse; dev words are the E1-R reserve words, test words are fresh.

Gate. Trained rows were kept in a language only if, on dev words, the target change passed a threshold set from the released model’s own take-to-take noise and the other letters, clip durations and utterance phone error stayed within fixed margins. Figure 31 shows every decision, hashed before any test word was synthesised: for each language and replicate, the change of the failing letters (red) must clear the language’s threshold (bar), and in no language does it. No language passed in either replicate, so the gated system is the released model. The random-take control and the dose arm needed test-word synthesis and were not run for budget reasons, so the title rule tied to them was not met.

E.5 Best-of- N selection (E6)

E6, a pre-analysis plan fixed before any selection outcome, keeps the take with the lowest whole-utterance phone error under ZIPA and grades the letter with w2v2p, against the expected grade of a

<

Figure 28: **Lead-in repair (first, 30 ms cut rule) for every model and language where it was run:** change in strict accuracy of word-initial consonants of utterance-initial items (w2v2p; blue: better). The trim control cuts the original clip by the same rule and stays near zero, so the gain comes from the lead-in.

Model	Position	Langs	Clips	E1	Lead-in	Δ [95% CI]	Trim Δ
MMS-TTS	initial	23	3,402	46.9	59.8	+13.0 [+10.4, +15.5]	+0.4
MMS-TTS	medial	23	3,544	52.2	56.6	+4.4 [+3.0, +5.8]	+0.1
MMS-TTS	final	23	3,568	53.1	55.3	+2.2 [+0.8, +3.6]	+0.2
MMS-TTS	all	23	10,514	50.8	57.2	+6.4 [+5.3, +7.6]	+0.2
Chatterbox-MTL	initial	11	1,440	65.2	67.8	+2.6 [-0.1, +5.2]	-0.2
Chatterbox-MTL	medial	11	1,466	66.9	67.7	+0.8 [-1.2, +2.8]	-0.1
Chatterbox-MTL	final	11	1,506	69.0	67.5	-1.5 [-3.5, +0.5]	+0.1
Chatterbox-MTL	all	11	4,412	67.1	67.7	+0.6 [-0.6, +1.9]	-0.1
XTTS-v2	initial	11	1,377	62.6	66.4	+3.8 [+0.2, +7.2]	-1.1
XTTS-v2	medial	11	1,416	69.4	67.7	-1.6 [-3.4, +0.4]	-0.3
XTTS-v2	final	11	1,440	66.9	66.0	-0.9 [-3.1, +1.2]	+1.0
XTTS-v2	all	11	4,233	66.3	66.7	+0.4 [-1.2, +1.9]	-0.1
OmniVoice	initial	28	4,238	66.1	59.8	-6.2 [-8.0, -4.5]	+0.2
OmniVoice	medial	28	4,366	62.6	62.2	-0.3 [-1.3, +0.6]	0.0
OmniVoice	final	28	4,406	63.1	61.5	-1.5 [-2.4, -0.6]	+0.2
OmniVoice	all	28	13,010	63.9	61.2	-2.7 [-3.5, -1.9]	+0.2

Table 39: Lead-in repair (first cut rule) by consonant position (an onset mechanism predicts the largest change word-initially).

Cut rule	Model	Langs	Clips	E1	Lead-in	Δ [95% CI]
30 ms margin	MMS-TTS	23	3,402	46.9	59.8	+13.0 [+10.4, +15.5]
30 ms margin	Chatterbox-MTL	11	1,440	65.2	67.8	+2.6 [-0.1, +5.2]
30 ms margin	XTTS-v2	11	1,377	62.6	66.4	+3.8 [+0.2, +7.2]
30 ms margin	OmniVoice	28	4,238	66.1	59.8	-6.2 [-8.0, -4.5]
end of lead word	MMS-TTS	23	3,402	46.9	62.4	+15.5 [+12.8, +18.1]
end of lead word	Chatterbox-MTL	11	1,440	65.2	72.6	+7.4 [+4.6, +10.1]
end of lead word	XTTS-v2	11	1,377	62.6	68.8	+6.2 [+2.7, +9.7]
end of lead word	OmniVoice	28	4,238	66.1	63.1	-3.0 [-4.6, -1.3]
80 ms margin	MMS-TTS	23	3,402	46.9	63.7	+16.8 [+14.2, +19.4]
80 ms margin	Chatterbox-MTL	12	1,566	66.7	72.2	+5.6 [+2.9, +8.4]
80 ms margin	XTTS-v2	11	1,377	62.6	69.8	+7.2 [+3.9, +10.4]
80 ms margin	OmniVoice	28	4,238	66.1	67.4	+1.3 [-0.2, +2.7]

Table 40: Lead-in repair under every cut rule, with clip counts.

a w2v2p							b ZIPA						
W1 MMS-TTS	+27	+32	+10	+13	+8	+17	+27	+25	+8	+15	+8	+11	
W1 XTTS-v2	+4	+3	-10	+3	+15	+10	+2	+7	+2	+3	+18	+2	
W1 Chatterbox	-1	+20	-6	+3	+18	+9	-2	+7	+6	+5	+10	-2	
W1 OmniVoice	+2	+9	-19	-3	+11	+2	+2	+9	0	+1	+4	+2	
W2 MMS-TTS	+27	+32	+10	+13	+8	+17	+27	+25	+8	+15	+8	+11	
W2 XTTS-v2	-2	+6	+1	+5	+7	+6	+2	+7	+2	+3	+8	0	
W2 Chatterbox	-7	+16	+5	+5	+5	+4	-4	+9	+4	+2	0	0	
W2 OmniVoice	-1	+6	0	+1	+5	-2	+1	+6	0	+1	+4	-3	
W3 MMS-TTS	+33	+24	-1	+10	-1	+13	+28	+18	+7	+8	+5	+6	
W3 XTTS-v2	+2	+3	+2	-8	+5	+9	0	+6	-1	-8	+6	+1	
W3 Chatterbox	-6	+12	+2	+3	+2	+6	-5	+4	+2	0	+4	-6	
W3 OmniVoice	-14	-27	-6	-10	-1	-32	-6	-19	-6	-14	-2	-26	
	ar	de	es	fr	hi	ru	ar	de	es	fr	hi	ru	

Figure 29: **Pre-registered onset remedies (E3-O) for every model and language:** change in strict accuracy of word-initial consonants of utterance-initial items, remedy clip minus original clip (points; blue: better), under both rulers. W1: lead-in-and-cut; W2: warm-up sentence; W3: repeat. MMS-TTS's W2 rows repeat W1 because its vocabulary has no sentence-final mark, so the two inputs are the same text. Clip counts and accuracies are in the released remedies_lang.csv.

Model	Langs	Graph.	None	Gated (Whisper)	Gated (phone)	Improved	Collateral
MMS-TTS	24	165	52.1	52.2	52.8	19	+0.2
IndicF5	3	10	32.5	34.2	38.1	2	+1.2
Chatterbox-MTL	2	10	44.4	45.0	49.4	2	-0.7
XTTS-v2	4	23	59.1	59.4	59.6	1	+0.5

Table 41: E3 row edits of failing letters: accuracy on the screening words (phone ruler) before and after the gated choice made on reserve words (Whisper).

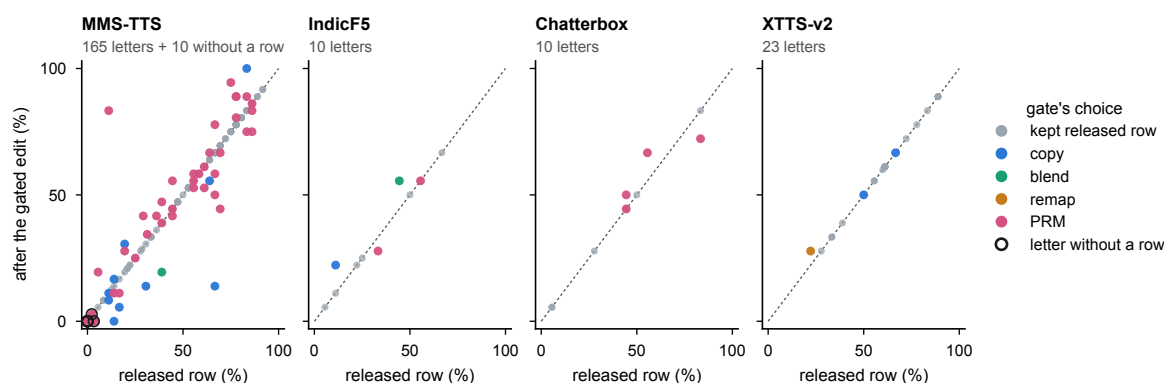


Figure 30: **Every E3 repair.** Each dot is one failing letter: screening-word accuracy (strict, w2v2p) of the released row against accuracy after the edit chosen by the Whisper gate on reserve words; colour: the chosen edit (grey: the gate kept the released row, so the dot sits on the diagonal); rings: letters without a row. The letter-by-letter list, with the phone-ruler gate, is in the released results/e3.

random take (Table 36).

E.6 Reset and relearn (E7, stopped)

E7 was pre-registered to test, on letters released models already have, whether a row must be trained to be pronounced (up to six well-pronounced letters

per language get an untrained row, retrained from 0 to 256 own takes). Its letter selection was amended once, after a selection run chose no letters and before any outcome data (44 letters, 8 languages); it was stopped for budget reasons before any clip was synthesised, and no result is reported.

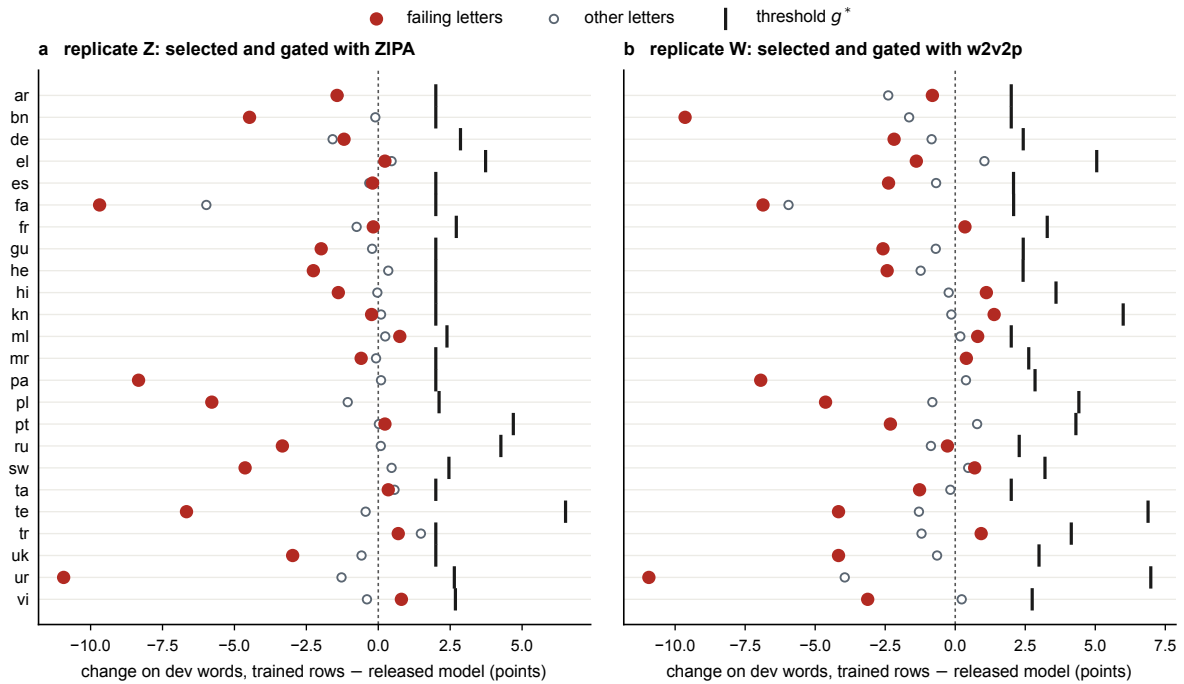


Figure 31: **E5' gate on dev words, per language and replicate.** Red: change in strict accuracy of the failing letters when their rows are trained; bar: the language's admission threshold g^* (from the released model's own take-to-take noise); hollow: change on the other letters. A language is admitted only when the red dot lies right of its bar; none is, in either replicate.

E.7 What each intervention gains

Table 42 collects the gain of every intervention over its baseline.

Chapter E in brief

The cut lead-in improves the utterance onset in all four models tested with the 80 ms cut. No repair of a released model's rows helped: gated edits were neutral, row-only training on real speech harmed through the KL path, and self-sample row training was rejected by its gate in every language. Best-of- N selection gains about a point.

Questions this chapter leaves open. Whether training that also updates the decoder can repair failing letters without harming the others, and whether the onset repair transfers to the other three models.

Intervention	Setting	Baseline	Ours	Gain, points [95% CI]	×
<i>Inference-time onset repair, no training (E3-O, pre-registered)</i>					
Cut lead-in vs. as is	MMS-TTS	53.3	70.3	+17.1 [+12.1, +22.7]	1.32
Cut lead-in vs. as is	Chatterbox-MTL	61.5	69.2	+7.7 [+3.4, +11.9]	1.12
Cut lead-in vs. as is	XTTS-v2	63.0	68.3	+5.2 [+1.2, +9.6]	1.08
Cut lead-in vs. as is	OmniVoice	72.8	74.5	+1.7 [−1.9, +5.3]	1.02
Cut lead-in vs. as is	pooled, 4 models			+7.7 [+1.9, +13.5]	
Cut lead-in vs. as is	MMS-TTS, 23 languages (exploratory)	46.9	63.7	+16.8 [+14.2, +19.4]	1.36
<i>Initialising a new symbol's row, no training (E2b)</i>					
Phonological analogy vs. random row	MMS-TTS	5.3	28.2	+22.9	5.30
Phonological analogy vs. random row	F5-TTS	3.9	48.1	+44.2	12.24
Phonological analogy vs. random row	Chatterbox-MTL	4.6	31.5	+26.9	6.80
Phonological analogy vs. random row	XTTS-v2	7.9	44.9	+37.0	5.71
	pooled, 4 models			+29.5 [+19.9, +39.1]	
<i>Training a new symbol's row: new rows trained vs. standard finetuning, d=160 (E2, E2c)</i>					
New rows trained vs. standard	VITS	21.3	65.3	+44.0	3.07
New rows trained vs. standard	F5-TTS	14.1	59.6	+45.5	4.22
<i>Gated post-hoc edits of failing rows in released models (E3)</i>					
Gated row edit vs. released row	4 models, 208 graphemes	51.6	51.7	+0.15 [−0.88, +1.26]	1.00

Table 42: **What each intervention gains over its baseline** (strict accuracy, w2v2p, %; ×: ours / baseline). Onset repair: word-initial consonants of words that open the utterance, six languages per model (95% CI over letters), and 23 of its 24 languages for MMS-TTS. E2b: new-symbol accuracy before any training, twelve English letters. E2/E2c: new-symbol accuracy at $d=160$, last checkpoint (plain-letter ceilings in Tables 24 and 26). E3: screening-word accuracy of failing letters after the gated edit; $1.00\times$ means no effect.

F Pre-registration, Changes and Reproduction

Every analysis plan frozen before its data, every amendment and deviation, every analysis decision taken after the first results, and how to regenerate every number of the paper.

In this chapter

F.1 Pre-registrations and amendments	48
F.2 Changelog of analysis decisions	48
F.3 Reproducing this appendix	49

F.1 Pre-registrations and amendments

Each plan below was frozen before its data, and its SHA-256 recorded with a UTC timestamp in the released hash log; deviations are listed in the released deviation log.

2026-10-01 05:11 E1-T: training-text counts of MMS-TTS, before any count.

2026-10-01 06:57 E1-R: held-out replication (R1–R4), before any reserve-word clip.

2026-10-01 08:22 E3-O and E1-H: onset remedies and the low-count same-sound test.

2026-10-01 08:49 E1-T2: training-text counts of XTTS-v2 and OmniVoice.

2026-10-02 14:23 E4: row-only repair of released MMS-TTS on real speech, before any training run.

2026-10-02 19:26 E6: best-of- N pre-analysis plan on existing takes, before any selection outcome.

2026-10-02 19:36 E5': self-sample row repair, before any of its data.

2026-10-03 03:19 E5' amendment 1: dose arm and title rule, before any E5' outcome.

2026-10-03 04:15 E7 and E5' amendment 2: stricter title rule, before any E7 data or E5' outcome.

2026-10-03 05:10 E7 amendment 1: corrected letter selection, after a selection run chose no letters.

2026-10-03 05:53 E5' gate decisions, before any test word was synthesised.

E4's plan and E5's amendments included a rule for using a different title, one that claimed frequency-driven embedding failure *and repair*; its conditions

were not met (E4's repair harmed; E5's gate admitted no language), so the paper keeps its title.

F.2 Changelog of analysis decisions

Decisions taken after inspecting model outputs, in order (all dated in the released log; every affected result was regenerated from saved per-clip outputs):

- 250 ms silence padding before recognition, after synthetic clips that start speech at the first sample showed under-detection of the first phone (the padding sensitivity check, 0 and 500 ms, is reported with the ruler controls). 1572
- Recogniser-level phone classes ($[r] \sim [r]$, $[v] \sim [v]$, $[h] \sim [h]$), digraph-consistent floors and canonical-span trimming (scorer version "w2v2p"); all scores were recomputed. 1573
- The GLMM gained a grapheme random intercept, Laplace estimation and smoothed floor counts, and the primary fit was restricted to judgeable cells (floor ≥ 0.2 on ≥ 20 tokens), after a specification audit; all sensitivity fits are reported. 1574
- ZIPA was added as a second phone recogniser after a literature audit showed that the primary recogniser cannot emit ejectives or breathy voice; results are reported under both recognisers. 1575
- The onset repair's cut margin was widened from 30 to 80 ms after the cut was found to clip word onsets in fast speech (after the first W1 lead-in outputs, before the E3-O pre-registration); all three cut rules are reported (§E.1). 1576
- E2 checkpoints are probed through their trained (online) weights after we found that the exponential-moving-average weights inherited from pretraining barely move in a short finetune. 1577

7. Training-text grapheme counts (E1-T) and the held-out replication (E1-R) were pre-registered before counting and before synthesis respectively (SHA-256 hashes and deviations released).
8. E1-R tests R3 and R4 had not been run, and the E1-R family had been Holm-corrected over two tests; both were found by an audit, R3 and R4 were run and the correction now covers the four pre-registered tests (§D.2).
9. Repair attempts, in order: E3 (training-free row edits, null), E4 (pre-registered row-only training on real speech, harmful), a gated joint-finetuning draft abandoned before any data, E6 (best-of- N selection on existing takes, pre-analysis plan), E5' (pre-registered self-sample row repair, the last training-based attempt) and E7 (pre-registered, stopped).

F.3 Reproducing this appendix

From saved per-clip outputs: `python -m longtail.report` (cells, floors, GLMMs; `--ruler zipa` and `default+reserve` for the second ruler and the replication), `python -m longtail.paper_tables`, `python -m longtail.appendix_tables`, `python -m longtail.paper_figures` and `python -m longtail.e4_rootcause`, then `latexmk -lualatex main`. Synthesis, scoring and training scripts for every experiment are released with their pod drivers. The code, the pre-registrations with their hash log, and every per-clip output (as archives with SHA-256 checksums) are at <https://tailprobe.pages.dev>, an anonymous site that sets no cookies and does not log visits.