Alpa: converting a small English text-to-speech model to Sanskrit by editing its lookup tables, without a GPU

Ekansh Srivastva

Version 1, 17 November 2026.

This report describes Alpa up to v0.2, an early preview: https://huggingface.co/77ethers/alpa

---

Abstract

Alpa is a Sanskrit text-to-speech model of 4.5 million parameters, made by converting tiny-voca, an open English model with the same architecture. Two lookup tables, 9,600 of its 4,531,723 parameters (0.2%), were rewritten by hand: each Sanskrit phoneme starts from the nearest English phoneme, syllable weight (laghu or guru) replaces English stress, and the four retroflex consonants, which English lacks, start as 70% of their nearest English sound plus 30% of English "r". The model was then trained for about 100 minutes on a four-core CPU, on 3.5 hours of one voice from the Rasa corpus. No GPU was used. On 148 held-out sentences a Sanskrit speech recogniser, Su-śrotā, misreads 26.6% of phonemes after the edit alone and 3.1% after training, against 3.3% in the human recordings. Without the "r" mixture the same training does not produce retroflex stops: ṭ (ट) is heard correctly 24% of the time between vowels, against 86% with it. Alpa v0.1 was an early preview with one trained voice.

Two things were done after the first release, again without a GPU. Listeners reported a buzz in the voice. A listener traced it to the sibilants and then to other noisy sounds, and the cause was in the inherited waveform generator: its harmonic branch is driven by pitched pulses alone, and the network had learned to build hiss, breath and bursts out of those pulses. A correction applied while speaking, with no retraining, gives every frame a frequency limit taken from the sound it belongs to, below which the branch is fed pulses and above which it is fed noise (v0.1.1). A second voice was then trained and the first given twice the recordings, in two more CPU hours (v0.2: 9.8 hours of speech, 3 hours 39 minutes of CPU training in all); the recogniser misreads 3.2% of phonemes in the male voice and 3.0% in the female, against 3.3% in each artist's recordings. The report also records two occasions on which our own measurements said a fault was fixed and a listener said it was not. The listener was right both times.

---

1. Introduction

Open text-to-speech (TTS) for Sanskrit is, as far as a search of Hugging Face in November 2026 could show, either large or limited to chant. Vāgdhenu [2] chants metrical verse and has about 449 million parameters in all. Indic Parler-TTS [6] covers Sanskrit among many languages with 938 million parameters. The one Sanskrit voice we found that runs in a browser is a 296 MB port of Vāgdhenu [8].

tiny-voca [1], by contrast, is an English TTS model of 4.5 million parameters that runs on a CPU. Its author trained it from scratch in 7.7 hours on one RTX 4090.

Alpa (अल्प, "small") converts tiny-voca to Sanskrit instead of training a new model. The hypothesis was that most of what such a network has learned (how to time, pitch and render a sequence of speech sounds) is not specific to English, and that the English-specific part sits in two small lookup tables: one row per phoneme and one row per stress level. If so, rewriting those rows by hand should give a model that reads Sanskrit with an English accent, and a short fine-tune on Sanskrit recordings should remove most of the accent.

The contributions are:

1. a conversion recipe, measured after each stage;
2. a controlled comparison in which the starting values given to sounds the source language lacks decided whether a short fine-tune found them;
3. a Sanskrit front end in which syllable weight replaces stress and aspirated stops are written as a stop followed by "h";
4. an evaluation procedure built on an independent Sanskrit recogniser, with its biases measured, and an account of a home-made measurement that misled us;
5. a public release of an 18 MB model that runs in a browser tab;
6. a correction made while speaking that removes pitched noise from the output of a pulse-driven waveform generator, with no retraining (Section 7);
7. a second voice, and a record of what further CPU training did and did not buy (Section 8).

"Without a GPU" describes the conversion; the starting weights are the product of tiny-voca's GPU training. Every quality figure for the released model was measured after the 100 minutes of CPU training, not after the edit alone. Sections 3 to 6 describe v0.1 as it was first released; Sections 7 and 8 describe what was done afterwards.

---

2. Background

tiny-voca [1], by Muhammed Nazeem, has 4,537,291 parameters, waveform generator included, and two English voices. A four-block transformer encoder reads phoneme, stress and voice embeddings, and a predictor gives each token a duration. After the encoder states are repeated by duration, a two-block transformer predicts pitch, voicing and energy for each frame and a five-block transformer predicts filter parameters. There is no neural vocoder: a source-filter head of 25,862 parameters generates a pulse train at the predicted pitch, plus noise, and shapes each with a time-varying filter. Output is 24 kHz audio in frames of 10.7 ms. tiny-voca was trained for 104,737 steps on 17.6 hours of synthetic English speech, with an adversarial stage from step 8,000.

Vāgdhenu [2], by Prathosh A P (Indian Institute of Science), chants metrical Sanskrit verse in one voice. It fine-tunes the IndicF5 backbone [5] and a vocoder on about 5.3 hours of recitation. Alpa's pronunciation and syllable-weight rules follow those of its text front end, in a separate implementation.

Rasa [4] is AI4Bharat's studio corpus for TTS in Indian languages (CC-BY-4.0). Its Sanskrit part has 23,759 utterances, 51.47 hours, from one female and one male voice artist. Its neutral reading styles (labelled WIKI, CONV, NEWS and BOOK) come to 28.95 hours once rows with non-Sanskrit characters or longer than 32 s are dropped. Alpa v0.1 used 3.51 hours of the male voice.

Su-śrotā [3], also by Prathosh A P, is a Sanskrit speech recogniser: AI4Bharat's IndicConformer (about 129 million parameters) fine-tuned on 17.4 hours of Sanskrit. It writes Devanagari. Its model page reports 4.4% character error on studio speech and states no licence. We used it only to measure.

---

3. Method

Sanskrit sounds are written here in the standard romanisation (IAST), with Devanagari where it helps.

3.1 Text front end

The front end turns Devanagari text into phoneme tokens plus one feature per token. It is published as `alpa/text.py`.

Phonemes. Each token is one Sanskrit phoneme in SLP1, an ASCII transliteration with one character per phoneme (`A` is ā, `w` is ṭ, `z` is ṣ). With pause-capable tokens for word boundaries and punctuation, the inventory has 47 symbols. Text is read as written: the vowel inherent in a consonant letter is always pronounced, and sandhi, the sound change at word junctions, is neither applied nor undone.

Syllable weight replaces stress. tiny-voca gives each phoneme an English stress value of 0, 1 or 2. Alpa instead marks syllable weight, the basis of Sanskrit metre. A syllable is guru ("heavy") if its vowel is long, or is followed by anusvāra (the nasal mark ◌ं) or visarga (◌ः, a breath after a vowel), or by two or more consonants before the next vowel; otherwise it is laghu ("light"). Its vowel carries 1 for laghu or 2 for guru, and every other token 0, so the input has the shape of tiny-voca's stress input. Weight carries timing information in ordinary speech: in the Rasa recordings, vowels in guru syllables last 1.27 (female) and 1.38 (male) times as long as those in laghu syllables.

Aspirated stops as stop + h. Sanskrit distinguishes ten aspirated stops, such as th (थ) and dh (ध), from plain ones; English does not. When each had its own token, initialised from the plain stop (Section 3.2), the author, who was the project's only human listener and is called *the listener* from here on, heard th as t and dh as d. The front end now writes each as the plain stop followed by the token `h`. The converted model can already say "h" after a stop, so aspiration is audible before any training.

Anusvāra and visarga. Anusvāra inside a word becomes the nasal consonant made at the same place as the stop that follows it, and `m` before a pause. At any other word end it keeps its own token, as does visarga, and the model learns their sound from the recordings.

3.2 Weight transplant

tiny-voca's published checkpoint has 184 tensors. Alpa's configuration equals tiny-voca's in every setting except the vocabulary and the voice names, so 182 tensors are copied unchanged. Two are rewritten by lookup:

- **Phoneme table** (47 rows of 192 numbers). Each Alpa symbol takes the row of the tiny-voca symbol we judged nearest in sound (Appendix A). Where English lacks a contrast, two Sanskrit phonemes share a starting row: ṭ and t both start from English "t" (Section 3.3).
- **Syllable-weight table** (3 rows of 192 numbers). "None" and laghu take tiny-voca's row for no stress; guru takes its row for primary stress.

That is 9,600 numbers, 0.2% of the model. tiny-voca's two voices become the starting points of Alpa's `female` and `male` voices. We call the result the *plain start*. Before any training it reads Sanskrit aloud with an English accent.

3.3 The r-coloured start for retroflex consonants

The retroflex consonants ṭ, ḍ, ṇ and ṣ (ट ड ण ष) are made with the tongue tip curled back; the dentals t, d and n (त द न) with the tip at the teeth. English has one series where Sanskrit has two, so in the plain start each retroflex row is an exact copy of another consonant's (ṭ of t, ḍ of d, ṇ of n, ṣ of ś). Fine-tuning separated the copies very slowly: after 400 steps on 20 minutes of audio the Euclidean distance between the ṭ and t rows was 0.17, against 16.5 between t and d, and the listener reported that the retroflexes were missing.

In the *r-coloured start*, each of the four retroflex rows is set to 0.7 × the row of its nearest English sound + 0.3 × the row of English "r" (tiny-voca's symbol ɹ). The reasoning is phonetic: English "r" is commonly made with the tongue tip curled back, and both it and the retroflex consonants lower the third resonance of the vocal tract (the third formant) [10]. This changes 768 numbers and nothing else. No other share and no other source sound was tried.

3.4 Timing labels

Training needs a start and an end time for every phoneme in every recording. As in tiny-voca, these come from offline forced alignment with `facebook/wav2vec2-lv-60-espeak-cv-ft` [7], a wav2vec 2.0 model fine-tuned with connectionist temporal classification (CTC) to output phonetic labels in several languages. It was not trained on Sanskrit and leans towards English-like sounds. We call it the *alignment recogniser*.

Each Sanskrit phoneme was assigned the label to which the recogniser gives most weight on 300 Rasa clips. It hears aspirated stops as plain and retroflexes as dentals, so several phonemes share a label. That affects only where boundaries fall; the model's tokens stay distinct. The best CTC path through each clip's label sequence gives every phoneme's onset, and durations are counted in 10.7 ms frames.

Accuracy was checked without hand labels: where a vowel meets a voiceless consonant, the voicing found by the pitch tracker must switch. On the 300 clips, 82.7% (female) and 80.5% (male) of such boundaries fall within two frames, against 81.2–85.6% for tiny-voca's English. The aligner gives the

two consonant series similar lengths between vowels (ṭ 42 ms, t 45 ms; ḍ 44 ms, d 45 ms), so the timing labels are not what held the retroflexes back.

3.5 Fine-tuning

Data. Eight of the 59 data files (shards) of Rasa's Sanskrit set (male voice, neutral styles) were resampled to 24 kHz and passed through tiny-voca's feature extraction. Of 1,635 clips, 81 (5.0%) failed the alignment checks, 1,396 (3.51 hours) were used for training and 158 (0.39 hours) were held out. Every reading of a given sentence falls in the same split. Preparation took about another hour on the same CPU.

Training. From the r-coloured start, the model was trained for 3,000 steps at a learning rate of 3e-4 on four CPU cores, in about 100 minutes. The trainer is tiny-voca's with small changes and updates all parameters. Its losses cover duration, pause, pitch, voicing and energy, plus multi-resolution spectral and mel losses on the waveform. No adversarial stage was run. The optimiser was AdamW (β = 0.8 and 0.99, weight decay 0.01) with gradients clipped at 10, 50 warm-up steps, and batches of up to 3,000 frames (32 s of audio) and at most 8 utterances. For the first 150 steps the acoustic decoder was conditioned on the recorded pitch, voicing and energy; over the next 150 steps this changed to the model's own predictions. The loss weights were 1 for duration, pause, voicing and energy, 2 for pitch, pitch spread, the auxiliary mel head and the spectral loss, and 10 for the mel loss. Each run used one random seed. The `female` voice saw no Sanskrit audio.

The result is Alpa v0.1: 4,531,723 parameters in an 18.2 MB file.

---

4. Evaluation method

Procedure. A model speaks each held-out sentence from text alone, predicting its own durations and pitch. Su-śrotā transcribes the audio. Reference text and transcript are both converted to phonemes by Alpa's front end and aligned by edit distance. *Phoneme error* is the number of substitutions, insertions and deletions divided by the number of phonemes in the reference text, pooled over all sentences. *Per-consonant accuracy* is the share of intended ṭ, ḍ or ṇ transcribed as intended. The same procedure on the human recordings of the same sentences gives the reference level. Of the 158 held-out clips, 148 are scored: the other ten had been training clips in the earlier 20-minute experiments of Section 5.2, and leaving them out lets every model in this report be scored on sentences it never saw. The `female` voice was scored on the first 60 of them, to save time.

Is it a good judge? On those recordings its phoneme error is 3.3%, and it writes ṭ, ḍ, ṇ and ṣ correctly 97%, 94%, 99% and 97% of the time. On audio the human listener had already judged, it agreed with the listener (Section 6).

Known biases.

- *It guesses from context.* For the plain start before training, whose ṭ is a copy of t, it still writes ṭ for 48% of them, because it knows the words. Counting only consonants between two vowels brings that to 19%. The strict figures here therefore count only consonants between vowels, and their floor for ṭ is 19%, not zero.

- *Small counts.* Between vowels the held-out set has 21 ṭ, 21 ḍ and 92 ṇ, so one consonant is 5 points for ṭ or ḍ.
- *It cannot judge ṣ against ś.* It writes ṣ correctly 84–89% of the time even before training, so no ṣ figure is reported.
- *Single words spoken alone* are transcribed badly for every model. Only sentences are scored.
- *Its training data may overlap ours.* Its model page lists two hours of "studio/synthetic voices" among its training data and does not say which; we do not know whether Rasa is among them.
- *It is a recogniser.* A score equal to the recordings' means the speech is as readable to it as the recordings are, not that the pronunciation is as good as a person's.

An earlier judge. One comparison (Section 5.2) predates Su-śrotā and was scored with the alignment recogniser, which misreads between 24.2% and 26.6% of phonemes even in human Sanskrit recordings, cannot hear aspiration or retroflexion, and favours English-like output.

Listening. All judgements by ear were made by the one listener, on a small number of lines.

---

5. Results

5.1 Stages of the conversion

Table 1 follows the released model from the edit to the end of training, in the `male` voice.

Table 1. Phoneme error by stage (Su-śrotā, 148 held-out sentences).

Stage	Training	Phoneme error
tiny-voca weights with the two tables rewritten (r-coloured start)	none yet (edit only)	26.6%
	500 steps	4.0%
	1,000 steps	3.5%
	1,500 steps	3.0%
	2,000 steps	3.4%
Alpa v0.1	3,000 steps, about 100 minutes on a CPU	3.1%
Human recordings of the same sentences		3.3%

Most of the gain comes in the first 500 steps. By this judge the model's speech ends as readable as the recordings, which is a statement about intelligibility and not about naturalness (Section 4).

Table 2. The two voices of Alpa v0.1 (Su-śrotā, consonants in all positions).

	Sentences	Phoneme error	ṭ ट	ḍ ड	ṇ ण	n न
Human recordings (male artist)	148	3.3%	97%	94%	99%	99%
Alpa v0.1, `male`	148	3.1%	93%	89%	98%	98%
Alpa v0.1, `female` (never trained on Sanskrit)	60	4.7%	82%	73%	96%	97%

The `female` voice is tiny-voca's English voice carried along by the shared layers. The listener judged the `male` voice to pronounce the test lines very well, with ṇ and n needing a little more work. It was the first model whose ṭ and ḍ the listener accepted. Section 8 describes the training that later gave the `female` voice its own Sanskrit.

5.2 Edited start against random weights

This early comparison used the plain start, 20 minutes of the male voice, and the alignment recogniser as judge.

Table 3. Phoneme error by the alignment recogniser on 16 held-out sentences. Training data: 144 clips (0.34 hours). 400 steps took about 14 minutes on the CPU. Aspirated stops still had their own tokens.

Starting point	0 steps	100 steps	200 steps	400 steps
Random weights	100%	70.6%	59.2%	46.8%
Edited English weights (plain start)	36.3%	23.2%	20.0%	18.3%
Human recordings of the same sentences	26.6%			

After 100 steps the edited model is ahead of where the random one stands after 400. This recogniser favours a model that starts from English, so a score below the recordings' does not mean the model speaks better than the voice artist. The 26.6% here is this recogniser's error on human recordings and is unrelated to the 26.6% in Table 1.

5.3 The retroflex control

Two runs used the same 3.5 hours, recipe and number of steps, and differ only in the four starting rows of Section 3.3.

Table 4. Retroflex consonants heard correctly between two vowels by Su-śrotā (all positions in brackets), and overall phoneme error.

Start	Steps	ṭ ट	ḍ ड	ṇ ण	Phoneme error
Human recordings		100% (97)	90% (94)	100% (99)	3.3%
Plain	0	19% (48)	43% (60)	37% (39)	26.4%
Plain	500	24% (50)	38% (38)	66% (64)	4.4%
Plain	1,000	19% (48)	48% (43)	73% (69)	4.2%
Plain	2,000	29% (58)	43% (49)	84% (79)	4.1%
Plain	3,000	24% (56)	52% (53)	83% (77)	3.9%
R-coloured	0	19% (49)	43% (53)	39% (41)	26.6%
R-coloured	500	29% (57)	48% (49)	71% (70)	4.0%
R-coloured	1,000	62% (80)	52% (68)	87% (87)	3.5%
R-coloured	1,500	81% (92)	71% (81)	96% (96)	3.0%
R-coloured	2,000	86% (93)	81% (85)	96% (95)	3.4%
R-coloured (Alpa v0.1)	3,000	86% (93)	81% (89)	97% (98)	3.1%

With the plain start the retroflex stops do not appear within 3,000 steps: ṭ stays near the judge's floor while the rest of the speech becomes readable. With the r-coloured start, ṭ and ḍ rise between steps 500 and 2,000 and then level off at 18 of 21 and 17 of 21, against 21 and 19 for human recordings. The nasal ṇ improves with either start, so the edit's clear effect is on the two stops. Before training the two starts score alike: the mixture does not itself make a retroflex, it gives training a direction. Each start was trained once, with one seed. Su-śrotā's transcripts of one test line show the same difference:

	Su-śrotā's transcript
Text	नटः पटं धरति, बालः पत्रं पठति। कटिः कति? दण्डः दन्तः च। (naṭaḥ paṭaṃ dharati, bālaḥ patraṃ paṭhati. kaṭiḥ kati? daṇḍaḥ dantaḥ ca.)
Plain start, 3,000 steps	नतः पतं धरति ... पथति ... दन्दः (nataḥ pataṃ dharati ... pathati ... dandaḥ)
R-coloured start, 3,000 steps	नटः पटं धरति बालः पत्रं पठति कटिः कति दण्ड दन्तः च (naṭaḥ paṭaṃ dharati bālaḥ patraṃ paṭhati kaṭiḥ kati daṇḍa dantaḥ ca)

5.4 A faster-learning phoneme table did not solve it

One alternative was tried on the 20 minutes of Section 5.2: keep the plain start but let the phoneme table learn ten times faster than the rest of the network. After 400 steps the ṭ and t rows were ten times further apart (1.6 against 0.17). The only measurement taken was the pairwise score of Section 6, which overstates progress, so Table 5 shows the ordering of the variants and no more.

Table 5. Pairwise retroflex-against-dental score after 400 steps on 20 minutes (0.50 is chance; see Section 6).

Start and recipe	ṭ/t	ḍ/d	ṇ/n
Plain start	0.62	0.73	0.61
Plain start, phoneme table learning 10× faster	0.72	0.80	0.62
R-coloured start, before training	0.66	0.61	0.63
R-coloured start	0.85	0.89	0.74

Moving the rows apart helped only a little. The obstacle was not the distance between rows but what the rest of the network does with the difference.

5.5 Aspirated stops

The stop + h spelling was adopted on the listener's verdict ("stop plus h sounded perfectly"). The alignment recogniser cannot hear aspiration, so Table 6 shows only that the timing labels and the overall score were not harmed.

Table 6. The two spellings of aspirated stops, judged by the alignment recogniser.

	One token per aspirated stop	Stop + h
Clips aligned, of 300	300	300
Clips passing the quality checks	293	293
Mean recogniser confidence	0.703	0.691
Vowel-to-voiceless boundaries within two frames	81.7%	81.5%
Phoneme error after 400 steps (the check of Section 5.2)	18.3%	18.6%

The 618 aspiration segments in the 300 clips get a median of 43 ms each. Per-aspirate accuracy has not been measured with Su-śrotā.

5.6 Speed and the browser

In PyTorch on four CPU threads the released checkpoint ran 123 times faster than real time. (The model card quotes an earlier measurement of about 100 times.) Exported with tiny-voca's exporter to ONNX, a portable model format that browsers can run, the 21.4 MB file gives the same sound lengths and the same Su-śrotā transcripts as PyTorch on five sentences.

In a browser the whole path from typed Devanagari to audio runs in the page, with the front end under Pyodide and the model under ONNX Runtime Web.

Table 7. Speed in a browser tab on one thread. Headless Chromium 153 on a 2.1 GHz Xeon; ONNX Runtime Web 1.20.1; Pyodide 0.26.4; five sentences, 31.8 s of speech.

Chrome's CPU throttle	Speed	A 7.8 s verse takes	Model start	Front-end start
None	12.4× real time	0.74 s	0.9 s	2.0 s
4×	2.9×	2.7 s	4.1 s	8.6 s
6×	1.9×	4.3 s	6.1 s	13.1 s

Such a page downloads 46 MB in all. The throttle imitates slower hardware on a desktop processor. Alpa has not been timed on a real phone.

5.7 Size

Table 8. Open Sanskrit voices on Hugging Face, 16 November 2026.

Model	Kind	Parameters	Download	Against Alpa
Alpa v0.1	Speech	4.5 M	18.2 MB	1×
Vāgdhenu browser port [8]	Chant	quantised	296 MB	16×
Vāgdhenu [2]	Chant	337 M + 112 M	1.35 GB voice + 0.45 GB vocoder	99×
Indic Parler-TTS [6]	Speech, many languages	938 M	3.75 GB	206×

This compares size only. The larger systems do more: Vāgdhenu chants in metre, and Indic Parler-TTS covers many languages and voices. None was run through our judge, and we make no claim about their quality relative to Alpa's.

---

6. A negative result: a score of 1.00 that a listener could not hear

The alignment recogniser cannot tell a retroflex from a dental, so before adopting Su-śrotā we built our own measurement of that contrast. For each pair (ṭ/t, ḍ/d, ṇ/n) a small classifier was trained on human Rasa recordings to say which of the two a consonant is, from the sound just before, during and just after it. The score is the probability that a randomly chosen retroflex is ranked above a randomly chosen dental: 0.50 is guessing and 1.00 is always right. Only consonants between vowels were scored, in speech generated with the recorded timing.

The measurement passed two sanity checks. Held-out human recordings scored 1.00, 1.00 and 0.99. The plain start before training, whose retroflex and dental rows are identical, scored 0.50, 0.55 and 0.52.

After 1,000 steps on the 3.5 hours, the r-coloured start scored level with human recordings. We wrote that up as the result and sent the listener audio from both models. The listener reported that both sounded the same as before: the retroflexes were still missing.

Table 9. The same two models at 1,000 steps, by three judges (figures are for ṭ, ḍ, ṇ).

	Pairwise score	Su-śrotā, between vowels	Listener
Plain start	0.70 / 0.84 / 0.73	19% / 48% / 73%	Retroflexes missing
R-coloured start	1.00 / 1.00 / 0.94	62% / 52% / 87%	Retroflexes missing; the same as the plain start
Human recordings	1.00 / 1.00 / 0.99	100% / 90% / 100%	

The listener was right. On the same audio files Su-śrotā wrote no retroflex in the isolated word pairs for either model, and on held-out sentences it put the r-coloured model about halfway to the recordings.

The score had measured separability, not arrival. It asked whether the model's ṭ looked more retroflex than the model's own t, not whether it lay where a human ṭ lies. A ṭ can rank above every t and still be a dental. The warning was in hand: on minimal pairs spoken with the model's own timing, every retroflex ranked above every dental, yet all fell on the dental side of where human recordings sit. We noted that and reported the ranking as the headline anyway.

The lessons we take from it:

1. A measurement we build ourselves is not trusted until it has agreed with a human listener at least once, on the same audio.
2. A ranking within a system's own output shows that a contrast exists, not that either side of it is right. Whether a sound has been learned needs an absolute test, such as an independent recogniser's transcript.
3. Sanity checks at the two extremes passed, because the failure lay between them.

---

7. Correcting the voice while speaking (v0.1.1)

7.1 A complaint, and three wrong leads

Within hours of the release a listener on X wrote that the voice had a buzz, and the project's listener confirmed that it was in the voice and not in the music of the launch video. We looked for it with measurements of our own, found three things that differ from human recordings, corrected each, and sent audio each time:

- above about 4.5 kHz the harmonics of the voice stand out more cleanly than in the recordings, where they dissolve into breath noise;
- the loudness of the noise branch zig-zags inside every 10.7 ms frame;
- the predicted pitch flickers from frame to frame (28 cents on average, against 10 or less in the recordings).

Each correction moved its own measurement to the level of the recordings. Each time the listener answered that the fault was still there, and described it more exactly: not a background buzz but something that happens "when something is spoken", then "a robotic sound of two metallic people saying at once", and at last: "check when श, क्ष is spoken".

7.2 The cause

tiny-voca's waveform generator has two branches. The harmonic branch passes a pulse train at the predicted pitch through a filter the network controls; the noise branch does the same with white noise. The pulse train is never switched off: it runs through voiceless sounds as well.

In the frames of ś and ṣ (श, ष) the network marks almost no frame as voiced, as it should. Yet the sound in those frames comes from the harmonic branch: its level there is −27.3 dB and the noise branch's −38.8 dB (70 held-out sentences). The network had learned to shape the pulses into something like a hiss. A hiss made of pulses keeps their pitch, so every ś carried a second, pitched voice inside it. The same was then found in the friction of j (ज) and in the breath of the voiced aspirates, which the listener named next.

To see it by frequency we measured how deep the ripple of harmonics is in the spectrum of a frame, in four bands, searching every harmonic spacing from 80 to 300 Hz so that no pitch track is needed. Pure noise gives 7.9, 6.9, 5.8 and 5.0 dB in the four bands. This measure reproduces the listener's verdicts on the sounds the listener named (Table 10).

Table 10. Depth of the harmonic ripple, in dB, by band (150 held-out sentences). At the floor a band is noise; above it, it is pitched.

Sound	Version	0.1–1 kHz	1–2.5 kHz	2.5–5 kHz	5–9 kHz
Pure noise (the floor)		7.9	6.9	5.8	5.0
ś ṣ s	Recordings	8.2	7.1	5.9	5.2
	v0.1	9.6	8.7	7.7	6.2
	Corrected	7.9	6.8	5.8	5.0
j	Recordings	11.7	7.2	5.8	5.0
	v0.1	12.4	9.7	9.0	8.9
	Corrected	13.3	7.0	6.2	5.2
g ḍ d b	Recordings	11.3	7.0	5.6	4.9
	v0.1	12.5	9.1	9.3	8.7
	Corrected	12.9	6.9	6.2	5.3
The h of gh dh bh	Recordings	8.7	7.0	6.0	4.9
	v0.1	9.0	9.3	9.2	8.3
	Corrected	8.7	6.8	5.9	5.0

In the recordings these consonants are pitched below 1 kHz at most, where the hum of the voice lies, and are noise above it. v0.1 is pitched up to 9 kHz.

7.3 The correction

Nothing is retrained. When speaking, every frame is given a frequency limit from the sound it belongs to. Below the limit the harmonic branch is fed the pulse train as before; above it, white noise with the same power per frequency. The filter the network predicts is used unchanged.

Table 11. The frequency limit of each kind of sound.

Sounds	Limit
ś ṣ s and c	None: noise at every frequency
Stops (k g ṭ ḍ t d p b) and j	1 kHz
h, including the h of an aspirated stop	1 kHz where the network marks the frame voiced; otherwise noise at every frequency
Visarga	Pulses where the frame is marked voiced; otherwise noise
Vowels, nasals, y r l v	4.5 kHz, above which a real voice is breath

A first version turned every frame of a voiceless stop to noise. The listener at once heard one syllable go wrong, and the reason was that the vowel after a stop begins inside the last frame the stop owns. The limit of 1 kHz keeps that beginning voiced.

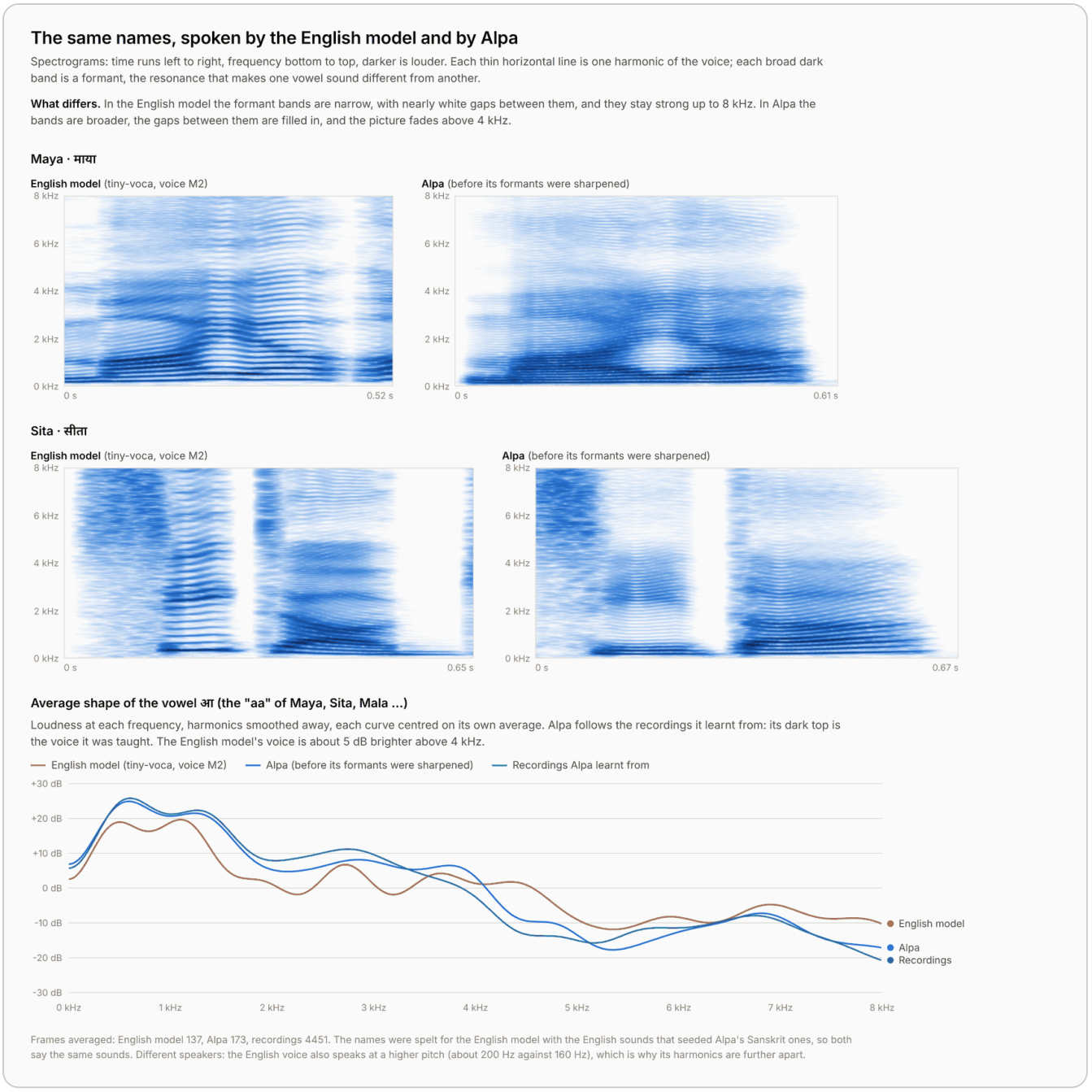


Figure 1. The same names spoken by tiny-voca in English and by Alpa in Sanskrit. Each English name was given to tiny-voca in the English sounds that seeded Alpa's Sanskrit ones. tiny-voca's formant bands are narrow, with nearly white gaps between them; Alpa's are broader and the gaps are filled. Alpa's darker top is the voice it was taught: it follows its recordings there.

Four smaller corrections are made in the same pass. The predicted pitch is averaged over three frames either side. In vowels, nasals and semivowels the peaks and valleys of the spectral envelope are stretched by a factor of 1.6 (cepstral coefficients 8 to 39, with the loudness of each of four broad bands put back): v0.1's formants were flatter than both the recordings' and the English model's (5.7 dB of contrast against 6.9 and 6.7), which is typical of training without an adversarial stage. In vowels the noise branch is held at least 15 dB under the harmonic branch. And the top of the spectrum is lifted by 4 dB.

The listener chose these last settings by ear from a ladder of candidates, and could hear no difference between the last three steps of it. The chosen formant setting is sharper than the

recordings (7.8 dB). On a side-by-side of six sentences against v0.1 the listener judged the corrected voice clearly better.

	v0.1	Corrected (v0.1.1)
Phonemes misread by Su-śrotā, 148 held-out sentences	3.1% (373 of 11,918)	3.0% (362)
ṭ ḍ ṇ heard correctly between vowels	86% / 81% / 97%	86% / 81% / 97%
Browser file	21.4 MB	22.6 MB
Speed in a browser tab, same day and machine	14.5× real time	14.0×

The correction changes how the voice sounds and leaves what it says where it was. It is on by default in the released code and inside the browser file; `--plain` gives the sound of v0.1.

7.4 A second negative result

This was Section 6 again. Three measurements were built and three faults corrected before the listener's ear located the real one in a single sentence. The measurements were not wrong about what they measured; they were answers to a question nobody had asked. Asking "in which sounds do you hear it?" at the start would have saved most of two days.

A second mistake was found when the correction was moved from an experiment script into tested code. The new code was checked sample by sample against the audio the listener had approved, and did not match. The experiment script had a clash between two variables of the same name, and from a certain candidate onwards it had also smoothed the loudness steps of the noise branch, a change nobody intended. Played both ways, the listener heard no difference, and the unintended change was removed.

The lessons we add to those of Section 6:

4. When a listener reports a fault that a measurement cannot find, ask where it is heard before building another measurement.
5. Audio that a listener is asked to judge should come from code that is checked against a reference, or should be labelled as coming from code that is not.

7.5 What the correction does not reach

A few vowels come out hoarse. In the e of te (ते) in Bhagavad Gītā 2.47 the noise branch is only 4.2 dB under the harmonic branch, where the median over 410 vowels is 21.5 dB: the network whispers the vowel on top of voicing it. Holding the noise down did not make the syllable right to the listener, and it is left open.

8. A second voice, and what more CPU training bought (v0.2)

8.1 More of the same would have bought little

Before spending more CPU time we looked at the run behind v0.1. From 2,000 to 3,000 steps its loss on unseen sentences moved by less than it wobbles from one check to the next (mel loss 0.727, then 0.763, 0.722, 0.730, 0.721), while its training loss kept falling (0.693 to 0.651). By 3,000 steps the model had heard its 3.5 hours about six times and had begun to learn them by heart. The recogniser's score was already at the level of the recordings, and the retroflexes had not moved since 2,000 steps (Table 1 and Section 5.3).

8.2 The female voice's own 384 numbers

The `female` voice of v0.1 was tiny-voca's English voice (Table 2). The listener judged it not good enough and heard it fade at the visarga. Against the female artist's recordings the fade can be measured: she speaks the visarga 2.1 dB under the vowel before it, and the model's female voice made it 4.7 dB under.

The safest thing to train is that voice's two rows, 2 × 192 numbers, with every other weight frozen so that the male voice cannot change. On 2.7 hours of the female artist this took 26 minutes. The rows moved by 40% of their size and the loss on unseen sentences by 2% (0.925 to 0.907). Two rows cannot teach the rest of the network a new speaker.

8.3 Both voices, all the weights

All the weights were then trained for 3,000 steps on both voices together: the female 2.7 hours, the male 3.5 hours behind v0.1, and 3.6 further male hours, so that the male voice met new recordings and not the old ones a seventh time. This took 93 minutes on the same four-core CPU. No sentence used for testing either voice was in the training set, by text or by recording.

Table 12. v0.2 against v0.1.1 and the recordings (Su-śrotā; 148 held-out sentences for the male voice, 145 for the female; the corrections of Section 7 on).

	Phonemes misread	ṭ / ḍ / ṇ, all positions	ṭ / ḍ / ṇ, between vowels
Female, v0.1.1 (never trained on Sanskrit)	3.2%	88 / 90 / 95%	76 / 100 / 95%
Female, v0.2	**3.0%**	94 / 97 / 100%	94 / 100 / 100%
Female recordings	3.3%	100 / 100 / 100%	
Male, v0.1.1	3.0% (362 of 11,918)	94 / 87 / 97%	86 / 81 / 97%
Male, v0.2	**3.2%** (376)	93 / 89 / 100%	100 / 81 / 100%
Male recordings	3.3%	97 / 94 / 99%	100 / 90 / 100%

- **The female voice.** Its visarga is now 2.0 dB under the vowel before it (recordings 2.1 dB) and 203 ms long (recordings 219 ms; before, 181 ms).
- **The male voice.** We had set a check before the run: its count of misread phonemes must not rise. It rose by 14 of 11,918. Seven of those are in one sentence of English words written in Devanāgarī, and both versions have 47 of 148 sentences with no error at all, so this judge cannot call the voice worse or better. It did lose on short, fast phrases: in "asato mā sad gamaya, tamaso mā jyotir gamaya" the t of asato and the m of tamaso are faint again.
- **Overfitting.** With 9.8 hours the loss on unseen sentences kept falling to the end and the training loss stayed beside it.
- **The listener's verdict,** on side-by-sides of both voices: both are clearly better than before, the female voice most of all.

In all, the released model has had 3 hours 39 minutes of CPU training since the English weights: 100 minutes for v0.1, 26 for the female rows and 93 for both voices.

---

9. Limitations

- **Little data.** The model learned from 9.8 hours, about a third of the neutral-style data in Rasa: 7.1 hours of the male artist and 2.7 of the female.
- **No adversarial stage.** tiny-voca's recipe relies on it for much of its naturalness. The voice can sound flatter than larger systems.
- **The cleaner voice is a correction, not a cure.** The network still builds noise out of pulses; the correction of Section 7 replaces the source while speaking. Training with a source that is switched by voicing would remove the cause, and has not been tried.
- **The corrections were tuned on the male voice.** They sharpen the female voice's formants more than her recordings have (7.4 dB of contrast against 6.3).
- **Hoarse vowels.** Section 7.5.
- **Short, fast phrases can lose a sound,** and the syllable oṃ on its own is often not heard (Section 8.3).
- **ṇ and n (ण, न).** The listener heard them as not fully distinct in v0.1. Su-śrotā does not register this in sentences. For this pair the listener's ear is the only judge, and it has not been asked again for v0.2.
- **Weak word-final visarga.** Su-śrotā missed it in three of the five published samples, in each voice.
- **Speech, not chant.** Verse is read aloud as text, and Vedic accent marks are ignored.
- **No real-phone timing.** The browser figures come from a throttled desktop processor.
- **No like-for-like benchmark.** No other system was scored by the same judge on the same sentences.
- **A single human listener,** on a handful of lines at a time, with no panel and no opinion score. Every listening verdict in this report is that listener's.
- **The judge leans on context,** and the held-out sentences come from the same corpus and voice artists as the training data.

- **One run per condition, small counts.** The retroflex control rests on one run of each start and on 21 ṭ and 21 ḍ.
- **Measurements of our own.** The ripple measure of Table 10 has agreed with the listener on the sounds the listener named. The measure of formant contrast has only the listener's choice among candidates behind it.
- **Untested inputs.** Single words on their own, very long compounds and long passages. Digits and Latin letters are not spoken.
- **One language pair.** The recipe was tried only from English to Sanskrit.

---

10. Reproducing it

Public now, on the model page [9], under Apache-2.0:

File	What it is
`alpa-v0.2.pt`	The weights of v0.2, 18.2 MB; sha256 `a85c36f7336c7f4d734f7f8aae87ba77a76b57065fb634f6aee605a042196b3c`
`alpa/`	The front end, configuration, model (with the correction of Section 7), a command-line speaker and the exporter for the browser file
`onnx/alpa-v0.2.onnx`, `.json`	The browser file, 22.6 MB (sha256 `a4602849514fb0a08553b8728b6ca0e8595124c774d2355cd9568b261af69046`), with a description of its inputs and phoneme table
`alpa-v0.1-preview.pt`, `onnx/alpa-v0.1.1.onnx`, `onnx/alpa-v0.1-preview.onnx`	The earlier weights (sha256 `ae052622c2b08c90fc979c16b1e7464834252facbea928eb5af18ae8509e073d`) and browser files
`samples/`	Five samples in each voice, each generated once, and one sentence before and after the correction

```
pip install "torch>=2.4" numpy soundfile huggingface_hub
hf download 77ethers/alpa --local-dir alpa-model
cd alpa-model
python -m alpa.infer --checkpoint alpa-v0.2.pt --voice female \
    --text "सत्यमेव जयते नानृतम्।" --output out.wav
```

`--plain` leaves out the correction of Section 7. With `alpa-v0.1-preview.pt` and `--plain` the command gives v0.1 exactly as first published.

Also on the model page: the code that made it.

File	What it is
`alpa/` (the rest of the package)	The conversion from tiny-voca's weights (`transplant.py`), the import of Rasa, feature extraction, timing labels and the trainer. Most are adaptations of tiny-voca's code, and each file says what was changed
`scripts/`	Four helpers: the r-coloured start of Section 3.3; the training of one voice's rows (Section 8.2); the joining of annotation runs into one training set (Section 8.3); and the scoring with Su-śrotā (Section 4)
`TRAINING.md`	Every command in the order it was run, from tiny-voca's weights to `alpa-v0.2.pt`, with the time each took and the six files to change for another language

Three of these were checked against what the released models were made from. The helper for the r-coloured start reproduces the starting weights of v0.1 exactly. The joining script reproduces the training set of v0.2 row for row. The scoring script reproduces the female voice's figures in Table 12. We have not checked that training a second time gives the same weights bit for bit.

A note on the scoring. The stock release (3.0.0) of NVIDIA's NeMo toolkit could not open Su-śrotā's file, so the script loads its audio front end, encoder and CTC head one by one, and decodes as its model card describes.

The recordings. Rasa is on Hugging Face (CC-BY-4.0; its terms must be accepted). The training data of v0.1 were shards 0 to 7 of 59 of its Sanskrit set, male voice, neutral styles; v0.2 added shards 8 to 15 (male) and 28 to 35 (female).

---

11. Future work

- Try the recipe on another language. What it needs is a front end, a table of nearest English sounds, a few hours of one clean voice and about 100 CPU minutes for a first voice.
- Train with a source that is switched by voicing, so that the correction of Section 7 is no longer needed, and run the adversarial stage.
- Train on the full neutral-style data (about 29 hours).
- Find why some vowels come out hoarse and why short, fast phrases lose sounds.
- Chant: add a chant voice and a metre input, and test whether syllable weight then controls timing.
- Time the model on real phones, and score other systems with the same judge on the same sentences.
- Try other shares and other source sounds for the r-coloured start.

---

12. Acknowledgements and licences

Alpa is released under the Apache License 2.0. It depends on the work of others.

- **tiny-voca** by Muhammed Nazeem (Apache-2.0): the model design, the training recipe, the browser export and the starting weights. Alpa's weights are derived from tiny-voca's.
- **Rasa** by AI4Bharat (CC-BY-4.0): the recordings the model learned from.
- **Vāgdhenu** and **Su-śrotā** by Prathosh A P, Indian Institute of Science. Alpa's pronunciation and scansion rules follow Vāgdhenu's, with no code taken from it. Su-śrotā was used to measure the result. Neither is included in the release.
- **The wav2vec2 phoneme recogniser** `facebook/wav2vec2-lv-60-espeak-cv-ft`: used to time the sounds in the training recordings.

How the work was done. The author directed the project and was its only listener. An AI coding agent (Claude Code) wrote the code, ran the experiments, kept the run notes and drafted this report, under the author's direction and review. Every listening verdict in this report is the author's. The two negative results (Sections 6 and 7.4) are cases in which the agent's measurements said one thing and the author's ear another.

This report was written after Prof. Prathosh A P publicly encouraged a write-up of the release. Alpa is an independent project: none of the people or groups named above has reviewed or endorsed the work or this report.

Citation for Rasa:

```
@inproceedings{ai4bharat2024rasa,
  author={Praveen Srinivasa Varadhan and Ashwin Sankar and Giri Raju and
Mitesh M. Khapra},
  title={{Rasa: Building Expressive Speech Synthesis Systems for Indian
Languages in Low-resource Settings}},
  year=2024,
  booktitle={Proc. INTERSPEECH 2024},
}
```

References

1. Muhammed Nazeem. *tiny-voca* (English text-to-speech model). Code: https://github.com/n4ze3m/tiny-voca (commit `28e0722`). Weights and model card: https://huggingface.co/n4ze3m/tiny-voca. Demonstration: https://tiny-voca.n4ze3m.com/.
2. Prathosh A P. *Vāgdhenu* (Sanskrit chant text-to-speech system). Code, technical report (`docs/TECH_REPORT.md`) and preprint (`docs/vagdhenu_paper.pdf`, 29 June 2026): https://github.com/prathoshap/vagdhenu (commit `8bd7733`). Weights: https://huggingface.co/prathoshap/vagdhenu. Corpus: https://huggingface.co/datasets/prathoshap/vagdhenu-data.
3. Prathosh A P. *Su-śrotā* (Sanskrit speech recogniser). https://huggingface.co/prathoshap/sushrota-sanskrit-asr.

4. Praveen Srinivasa Varadhan, Ashwin Sankar, Giri Raju and Mitesh M. Khapra. "Rasa: Building Expressive Speech Synthesis Systems for Indian Languages in Low-resource Settings." *Proc. INTERSPEECH 2024*. Dataset: https://huggingface.co/datasets/ai4bharat/Rasa.
5. AI4Bharat. *IndicF5*. https://huggingface.co/ai4bharat/IndicF5.
6. AI4Bharat. *Indic Parler-TTS*. https://huggingface.co/ai4bharat/indic-parler-tts.
7. wav2vec2 phoneme recogniser. https://huggingface.co/facebook/wav2vec2-lv-60-espeak-cv-ft.
8. Vāgdhenu browser port. https://huggingface.co/gnumanth/sanskrit-tts-wasm.
9. Alpa model page. https://huggingface.co/77ethers/alpa.
10. Peter Ladefoged and Ian Maddieson. *The Sounds of the World's Languages*. Blackwell, 1996. (Retroflex articulations and their lowered third formant.)

Appendix A. Starting row for each symbol

Each Alpa symbol starts from the row of one symbol in tiny-voca's phoneme table. Sanskrit sounds are given in romanisation with the SLP1 token in code type; tiny-voca's symbols are in the International Phonetic Alphabet.

Group	Sanskrit sound (token) → tiny-voca row
Vowels	a (`a`) → ʌ; ā (`A`) → ɑː; i (`i`) → ɪ; ī (`I`) → iː; u (`u`) → ʊ; ū (`U`) → uː; ṛ (`f`) → ɚ; e (`e`) → eɪ; ai (`E`) → aɪ; o (`o`) → oː; au (`O`) → aʊ
Velar and palatal consonants	k (`k`) → k; g (`g`) → g; ṅ (`N`) → ŋ; c (`c`) → tʃ; j (`j`) → dʒ; ñ (`Y`) → n
Retroflex consonants (plain start)	ṭ (`w`) → t; ḍ (`q`) → d; ṇ (`R`) → n; ṣ (`z`) → ʃ
Dental and labial consonants	t (`t`) → t; d (`d`) → d; n (`n`) → n; p (`p`) → p; b (`b`) → b; m (`m`) → m
Semivowels, sibilants, h	y (`y`) → j; r (`r`) → ɾ; l (`l`) → l; v (`v`) → v; ś (`S`) → ʃ; s (`s`) → s; h (`h`) → h
Marks	anusvāra (`M`) → m; visarga (`H`) → h
Pauses and specials	daṇḍa and double daṇḍa (verse marks) → "."; caesura → ","; comma, question mark, exclamation mark, padding, start, end, word boundary and unknown → the same symbol in tiny-voca
Syllable weight	none → stress row 0 (no stress); laghu → stress row 0; guru → stress row 1 (primary stress)

In the r-coloured start, the four rows of the "Retroflex consonants" line become 0.7 × the row shown + 0.3 × the row of tiny-voca's ɹ, the English "r" of "red". (Sanskrit r itself starts from the tap ɾ, a different row.) The ten aspirated stops have no row: the front end writes each as the plain stop

followed by `h`. Rare vowels are written with frequent ones (long r̄ as ṛ with guru weight, ḷ as `l` followed by ṛ), so they need no row either.

For timing (Section 3.4), each phoneme's label in the alignment recogniser is the symbol shown above, with these exceptions: a → a; ā → aː; i → i; u → u; ṛ → r; e → e; r → r.

Two checks were made on the front end. Its phoneme reading agrees with the `indic-transliteration` library on all 58,035 distinct Sanskrit words in Rasa. Its syllable weights agree with Vāgdhenu's own scanner on 1,459 of Vāgdhenu's 1,467 corpus texts; the other 8 differ only in how ॐ is read.

Appendix B. Provenance

The figures in this report come from the project's dated run notes (`docs/runs/` in the private repository) and from the public model card. Where two notes differ, the later one is used.

Part of this report	Source
Section 2, tiny-voca and Vāgdhenu	Run note 01 (research)
Section 2, Rasa; Section 3.1 checks	Run note 06 (front end and Rasa audit); design document
Section 2, Su-śrotā; Section 4	Run notes 10 and 14
Section 3.1, aspirated stops; Table 6	Run notes 08 and 12
Section 3.2; Table 3	Run notes 11 and 12; `transplant.py`
Section 3.3; Table 5; Section 6	Run note 13, as corrected by run note 14
Section 3.4	Run notes 08 and 14; `align.py`
Section 3.5	Run notes 13, 14, 16 and 18; `train.py`; the saved recipe of the training run
Tables 1, 2, 4 and 9	Run notes 13 and 14
Tables 7 and 8; Section 5.6	Run notes 07, 14 and 16
Section 7; Tables 10 and 11	Run notes 20, 21 and 22
Section 8; Table 12	Run notes 23 and 24
Section 9	Model card; run notes 14 to 24
Section 10	Run notes 14 to 24; the model page