Offline Urdu Text to Speech in QnVoice for iOS with Distilled Matcha

QnVoice

Research preprint · November 2026

Abstract

We adapt a pretrained Urdu Matcha acoustic model for offline text to speech in the QnVoice iOS app. We shorten its ten-step flow solver to three evaluations and train a 75,920-parameter residual network to recover mel features from synthetic teacher targets. The text encoder, duration predictor, and HiFi-GAN vocoder are shared across variants. In the Urdu Lab benchmark on an iPhone 15 Pro, median warm synthesis time fell from 1.50 s to 1.30 s across 72 runs per model. The 95th percentile fell from 2.08 s to 1.79 s. On an Apple M1 Pro, median warm synthesis time fell from 1.30 s to 1.11 s across the same 24 held-out phrases. Whisper word error was 22.1% for the original model and 19.3% for the student. The paired interval includes no change. The learned correction reduced mel error relative to the uncorrected three-step solver. The measurements establish a latency improvement for these prompts. Pronunciation and naturalness still require assessment by Urdu listeners.

Keywords Urdu, QnVoice, iOS, iPhone, text to speech, flow matching, distillation, ONNX

1 Introduction

QnVoice is an iOS app for on-device voice tools and translation. Its bundled Urdu model can synthesize Urdu script entered by a user or Urdu text produced by a separate translation component. The acoustic workload begins after text is available, so a comparison of speech models can hold the input and waveform renderer fixed. This separation also lets the app use the same voice for typed text and translated phrases.

Matcha-TTS generates a mel spectrogram through a conditional flow solved in a small number of steps [1]. The Urdu export used in this work runs ten flow steps [2]. Reducing the number of estimator evaluations saves acoustic computation, and the shorter integration schedule can shift the mel trajectory. We evaluate whether a small learned correction can reduce that shift while preserving the runtime gain. The comparison follows the complete path from token IDs to a waveform because the shared vocoder can dominate elapsed time.

The work contributes an inference-compatible Urdu ONNX graph, a comparison with both the original and an uncorrected three-step version, and paired measurements on a Mac and an iPhone. The companion project includes the exact evaluation prompts, sample clips, and phone timing records [10].

2 Related Work and Study Scope

2.1 Acoustic and waveform models

Mehta and colleagues train Matcha-TTS with optimal-transport conditional flow matching [1]. During synthesis, a duration predictor supplies a frame allocation and an ordinary differential equation maps noise to the target mel representation. The Urdu teacher is an ONNX export of AhsanTalal's Urdu checkpoint [2]. The present adaptation works at export level. It retains the learned encoder, duration path, and flow estimator and adds a correction after the final solver step. The uncorrected solver serves as the control for assessing the correction.

HiFi-GAN synthesizes waveform samples from a mel spectrogram using a generator trained with adversarial discriminators [3]. We use the same HiFi-GAN v1 ONNX conversion for all variants [4]. Holding this graph constant gives the comparison a shared output rate and keeps vocoder changes separate from the acoustic intervention. The resulting speed measurements describe the assembled pipeline used here.

2.2 Urdu data provenance

The teacher card identifies UrduSpeech as its training corpus [2]. UrduSpeech describes itself as a derivative of AI4Bharat's Rasa and lists Urdu and Kashmiri material across several speaking styles [5, 6]. The Rasa paper describes the data-collection approach for low-resource expressive speech [7]. These sources establish the reported lineage of the teacher. Our training examples are new text prompts paired with synthetic mel targets produced by that teacher. The experiment used no corpus recordings as direct inputs.

2.3 Research questions

Table 1 Comparisons used to answer each question

Question	Comparison and endpoint
How much time does the shorter solver save?	Original and three-step variants with a shared vocoder. Acoustic and complete synthesis times.
What does the correction add?	Uncorrected and corrected three-step graphs. Mean absolute mel error relative to the teacher, with transcript edits as a secondary check.
Does the latency gain carry over to iPhone?	Original and corrected models in Urdu Lab. Warm per-prompt and pooled timings on one device.

Whisper transcription supplies a repeatable comparison with the intended text [8]. Listener preferences, perceived pronunciation, and the quality of machine translation are outside these experiments.

3 Acoustic System and Distillation

3.1 Tokenization and the shorter flow solver

The acoustic graph consumes an Urdu grapheme sequence with blank tokens interleaved between characters, its length, and two synthesis scales. We use the teacher vocabulary and a noise scale of 0.667 with duration scale 1.0. The ten-step export evaluates a shared estimator at successive times. The reduced graph evaluates the retained estimator at 0.0, 0.3, and 0.6. Its integration increments are 0.3, 0.3, and 0.4, which carry the state to the same terminal time. The reduction changes the acoustic integration schedule while leaving the input and frame-allocation path in place.

3.2 Mel correction training

For each training text, we run the original and reduced graphs with paired noise seeds. The resulting mel sequences must have the same shape and finite values. The 384 authored prompts combine 16 templates with locations, personal names, objects, and times. A seeded permutation assigns 344 prompts to optimization and 40 to validation. The 24 held-out evaluation sentences are a separate set of strings.

The correction network projects 80 input bands into 96 channels and uses three residual blocks. Each block applies a depthwise temporal convolution, SiLU, and a pointwise projection. Dilation increases from 1 to 2 to 4. A zero-initialized output projection adds a learned residual to the reduced mel. Training uses AdamW at 0.001 for ten epochs, with batches of eight examples and crops of up to 192 frames. The loss combines masked absolute mel error with a 0.2-weighted first-difference term. Validation uses full sequences and selects the checkpoint with the lowest mean absolute error relative to the teacher.

Table 2 Graph interfaces and retained components

Component	Interface and implementation
Text and timing	Vocabulary, duration predictor, and shared encoder retained from the Urdu export.
Acoustic inference	x, x_lengths, and scales → mel and mel_lengths. Three estimator evaluations plus the learned residual.
Waveform	80-band mel spectrogram → mono 22.05 kHz samples through the pinned HiFi-GAN graph.

The dynamic frame axis survives ONNX export of the correction and composition with the acoustic graph. QnVoice can load the result with the same inference interface it uses for the original model.

4 Experimental Methods

4.1 Prompt set and timed workloads

The 24 held-out prompts cover greetings, requests, travel questions, directions, and other short everyday statements. The prompts were authored for the project. Appendix A gives six examples and paired listening clips. We use the same prompts and tokenized inputs for both systems on each device. The Mac comparison includes a third variant with three solver evaluations and no learned correction.

Table 3 Timing protocols

Protocol	Configuration	Measured interval
Mac warm	Apple M1 Pro, ONNX Runtime 1.24.2, two CPU threads. 24 prompts × 3 passes per variant.	Acoustic inference and complete waveform generation with sessions loaded.
iPhone warm	iPhone 15 Pro, iOS 27.0, native ONNX Runtime CPU, two threads. 24 prompts × 3 passes per model.	Acoustic inference and complete in-memory waveform generation.
Mac fresh sessions	One water request, five trials per model in the same process.	Fresh acoustic and vocoder sessions plus synthesis.

Each warm protocol runs a warm-up before timed requests. The Mac evaluator rotates variant order by prompt and repeat and pairs noise seeds. The phone harness alternates which graph runs first by prompt. The synthesis timer ends before writing or playing the output. Real-time factor divides elapsed synthesis time by the duration of the generated waveform. The fresh-session protocol can benefit from the operating system’s file cache.

4.2 Transcript and signal checks

One Mac waveform per prompt and variant was transcribed with Whisper small in Urdu [8]. Scoring applies Unicode compatibility normalization, maps Arabic yeh and kaf to Urdu forms, and removes marks and punctuation. Word and character error are aggregate edit counts divided by reference word and character counts. The set contains 181 reference words and 600 reference characters. We bootstrap the difference in word error with 10,000 paired resamples of the 24 prompts. Each resample uses the same prompt indices for both variants.

The waveform checks require finite samples and a non-silent peak. These checks catch empty or invalid outputs. The transcript score depends on Whisper’s own errors and spelling choices.

5 Results

5.1 Desktop ablation

Table 4 Apple M1 Pro synthesis and recognition results

Metric	Original Matcha	Three steps	Urdu student
Acoustic graph size	75.3 MB	73.9 MB	74.2 MB
Acoustic median	274 ms	92 ms	94 ms
Synthesis median	1.30 s	1.13 s	1.11 s
Synthesis 95th percentile	1.81 s	1.50 s	1.43 s
Median real-time factor	0.49	0.42	0.41
Whisper word edits / rate	40 / 22.1%	36 / 19.9%	35 / 19.3%
Whisper character edits / rate	54 / 9.0%	43 / 7.2%	45 / 7.5%
Mel MAE against original	—	0.1103	0.1078

Timings aggregate 72 warm requests per model. Word and character errors use one scored clip per phrase and model. MAE is mean absolute error on the mel spectrogram.

The three-step solver accounts for nearly all the reduction in acoustic inference. Its median is 92 ms, compared with 274 ms for the original. The learned correction adds about 2 ms to the acoustic median. Across the complete warm pipeline, the student median is 1.11 s and the original median is 1.30 s. The shared waveform stage limits the end-to-end gain.

On the 40 validation prompts, the learned correction reduced mean absolute mel error from 0.1109 to 0.1082. On held-out prompts the change was from 0.1103 to 0.1078. The validation and held-out changes both favor the correction on the teacher-matching objective.

The student yielded 35 word edits compared with 40 for the teacher. The paired 95% interval for the difference is −7.0 to +1.6 percentage points. The uncorrected graph produced 43 character edits, two fewer than the student. The findings support the narrower claim that the correction better matches teacher mel features. They leave the transcription effect uncertain.

5.2 Warm Synthesis on iPhone

Urdu Lab compared the original and corrected graphs on the iPhone 15 Pro. All 144 timed requests returned finite, non-silent waveforms. Generated durations ranged from 2.07 to 3.36 s and were equal between models for each prompt. This lets the phone comparison isolate synthesis time from differences in the amount of audio generated.

Table 5 Warm iPhone 15 Pro timing

Metric	Original Matcha	Urdu student
Acoustic median	311 ms	103 ms
Synthesis median	1.50 s	1.30 s
Synthesis 95th percentile	2.08 s	1.79 s
Longest timed synthesis	2.30 s	1.99 s
Median real-time factor	0.56	0.48

The median synthesis time fell by 13.0%. All 72 student runs completed in under 2 s, compared with 65 of 72 original-model runs. Figure 1 aggregates repeats by phrase. The student median was lower for each of the 24 phrases. These observations apply to the tested request lengths and loaded sessions.

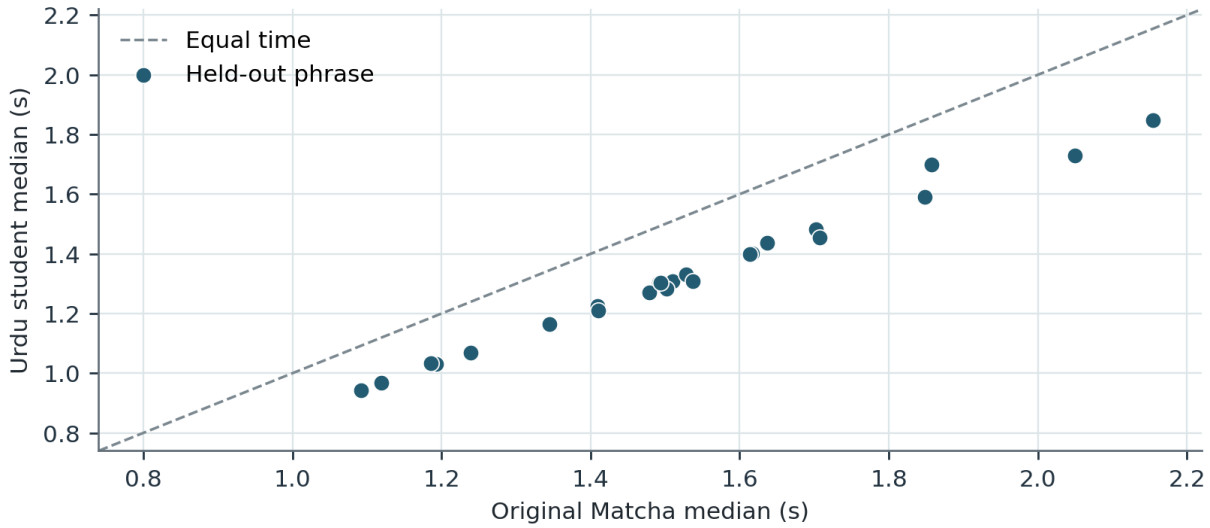


Figure 1 Paired median synthesis time per Urdu prompt. Each point represents three warm runs per model. Points below the diagonal favor the student.

The acoustic median fell from 311 ms to 103 ms. The larger relative change at this stage is consistent with the shorter flow schedule. HiFi-GAN and waveform allocation remain part of every request, so the total latency reduction is smaller than the acoustic reduction.

5.3 Fresh Sessions and App Integration

Fresh sessions on the Mac

For the water request shown in Appendix A, we created new acoustic and vocoder sessions and generated one waveform per trial. This checks a different source of delay from the warm timings. The median was 5.15 s for the original graph and 1.79 s for the student.

Table 6 Time for a fresh session and one synthesis on Apple M1 Pro

Trial	Original Matcha (s)	Urdu student (s)
1	8.33	1.81
2	4.81	2.09
3	4.63	1.79
4	5.26	1.60
5	5.15	1.58
Median	5.15	1.79

The trials ran in one Mac process, and the operating system could reuse cached pages. The interval includes session initialization and synthesis together. A single prompt and five trials give limited evidence about first use. Mobile session initialization requires a separate measurement.

Integration in QnVoice

The acoustic file in QnVoice development build 14 was checked against the exact graph used in Urdu Lab. QnVoice sends typed Urdu to the bundled synthesizer. English text can first pass through the bundled English-to-Urdu translator, after which the Urdu text uses the same speech path. The pronunciation and latency of that full translation route depend on the translator and the rendering of the resulting text.

Three focused tests passed on the iPhone 15 Pro. They exercised direct typed Urdu and re-rendering after a transcript edit, playback of an English-to-Urdu take, and cancellation or rejection of an overlong translation input. The playback checks required finite, audible-level samples. The tests establish that the ONNX replacement works through the app’s entry points. They use different endpoints from the Urdu Lab timing harness.

6 Discussion

6.1 What the measurements support

Reducing estimator evaluations is an effective way to lower acoustic inference cost in this export. The gain carries over to the full waveform pipeline on both devices. The uncorrected ablation explains most of the speed difference. The correction contributes a smaller and separable improvement in mel agreement. The word and character comparisons rank the three models differently.

Teacher-target error measures agreement with a reference synthesizer. Naturalness requires listener judgments. In the water prompt, Whisper uses an alternate spelling for the last word in both models' clips, and the comparison records one word edit for each. Appendix A makes selected pairs available for language-specific listening.

6.2 Limitations and evaluation priorities

The study uses 24 authored phrases, one teacher, one shared vocoder, and one iPhone. The prompt set lacks systematic coverage of dialect, named entities, numbers, code-switching, and long readings. The repeated runs estimate timing variation within the same set of prompts. Peak resident memory, thermal behavior, and battery drain remain unmeasured. System load, available memory, and audio length can change elapsed time in use.

A follow-up could present blinded paired clips to fluent Urdu listeners and collect separate ratings for intelligibility, naturalness, and preference. A pronunciation checklist could target numbers, regional terms, and named entities. A device study could randomize model order across phones and report first synthesis, time to playback, and peak memory as separate endpoints. ITU-T P.85 provides a reference for subjective speech-output assessment [11].

6.3 Provenance and use

The model carries the pinned teacher's attribution and identifies the graph changes. The public cards list the teacher under Apache 2.0 and the source dataset releases under CC BY 4.0 [2, 5, 6, 12]. The data access conditions and the scope of speaker rights remain part of the commercial release review. The generated examples are synthetic evaluation outputs. The application performs the synthesis locally.

7 Conclusion and Availability

The three-step Urdu student reduces warm synthesis time on the tested iPhone while keeping the original graph interface and waveform duration. The residual correction improves agreement with teacher mel targets. A larger and independently rated Urdu sample is needed to establish whether listeners prefer the student or judge its pronunciation more accurate.

The companion project provides the 74.2 MB acoustic graph, 55.8 MB vocoder, vocabulary, inference runner, checksums, and paired examples. The benchmark directory includes the training summary, Mac evaluation, fresh-session timings, and raw iPhone records. The earlier preprint version is preserved in the Zenodo series [9].

GitHub model and evaluation files · Pinned Matcha teacher on Hugging Face · Zenodo versions

References

[1] Mehta, S., Tu, R., Beskow, J., Székely, É., and Henter, G. E. Matcha-TTS: A fast TTS architecture with conditional flow matching. ICASSP, 2024.

[2] AhsanTalal. Matcha Urdu TTS. Hugging Face model card and pinned ONNX export.

[3] Kong, J., Kim, J., and Bae, J. HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis. NeurIPS, 2020.

[4] k2-fsa. HiFi-GAN v1 ONNX conversion. sherpa-onnx vocoder releases.

[5] Munir, H. UrduSpeech. Hugging Face dataset card. DOI 10.57967/hf/7224.

[6] AI4Bharat. Rasa. Hugging Face dataset card and license.

[7] Srinivasa Varadhan, P., Sankar, A., Raju, G., and Khapra, M. M. Rasa: Building expressive speech synthesis systems for Indian languages in low-resource settings. Interspeech, 2024.

[8] Radford, A., et al. Robust Speech Recognition via Large-Scale Weak Supervision. ICML, 2023.

[9] QnVoice. Urdu text to speech research preprint and archived revisions. Zenodo.

[10] QnVoice. Urdu acoustic graph, benchmark records, and offline inference code. GitHub.

[11] ITU-T. Method for subjective performance assessment of the quality of speech voice output devices. Recommendation P.85.

[12] Creative Commons. Attribution 4.0 International license.

Appendix A Evaluation Samples

The following examples come from the held-out set used on the Mac and iPhone. The paired clips are from the first timed Mac pass. Each pair uses the same HiFi-GAN vocoder. The English glosses explain the test prompts.

Table A1 Selected Urdu inputs and their synthesized outputs

ID	Urdu prompt	English gloss	Mac synthesis clips
02	مجھے ایک گلاس پانی چاہیے۔	I'd like a glass of water.	Original WAV Student WAV
03	ذرا دائیں طرف مڑیں اور پل تک چلتے رہیں۔	Please turn right and keep going until you reach the bridge.	Original WAV Student WAV
12	آج ٹرین دس منٹ تاخیر سے آئے گی۔	The train will arrive ten minutes late today.	Original WAV Student WAV
15	مجھے سمجھ نہیں آئی، آہستہ بولیے۔	I didn't understand. Please speak slowly.	Original WAV Student WAV
18	واپسی کا ٹکٹ کہاں سے ملے گا؟	Where can I get a return ticket?	Original WAV Student WAV
22	کل کی ملاقات دو بجے ہوگی۔	Tomorrow's meeting will be at two o'clock.	Original WAV Student WAV

Reproduce the Water Request

From the companion project, install the Python requirements, verify the three model files, and render the saved Urdu text. The WAV is mono at 22.05 kHz.

```
python3 -m pip install -r requirements.txt
shasum -a 256 -c SHA256SUMS
python3 speak-urdu-tts.py --model-dir model \
  --text-file samples/glass-of-water.txt --output water.wav
```