

Hebrish v1.0: fine-tuning Parakeet for English-Hebrew code-switched dictation

How v1.0 (Jachnun) was trained, exported and benchmarked, what it gets right, and what the numbers do not show.

Daniel Rosehill · 2026-10-19 · [hebrish.app](#)

Model: `danielrosehill/Parakeet-TDT-0.6B-v2-English-Hebrew-Codeswitching`, v1.0, run `r6-en30`

Abstract

English speakers in Israel drop Hebrew words into English. English speech recognition has rarely heard those words and writes the nearest English-sounding thing. Hebrish v1.0 is a full fine-tune of NVIDIA Parakeet TDT 0.6B v2 on 339.2 minutes of code-switched read speech plus about 30% English, taught to write each Hebrew word in one canonical romanized spelling (*makolet*, *Bituach Leumi*). It is the sixth checkpoint and the first stable release; checkpoints one to five are now called PoC-1 to PoC-5. On the public English-Hebrew code-switching benchmark (test set v1, clean subset, 84 clips) v1.0 scores 73.3% Hebrew term recall, 70.1% on terms absent from training, 3.3% WER and 2.2% English control WER, against 32.6%, 32.5%, 9.5% and 2.7% for stock Parakeet v2. Plain English did not get worse. All audio, training and test, is one speaker's voice, so these numbers rank models on that speaker only.

73.3%

term recall, public
benchmark (stock v2:
32.6%)

70.1%

recall on terms never in
training

3.3%

WER, public
benchmark (stock v2:
9.5%)

1.7%

own-English WER
(stock v2: 3.1%)

1. The problem

Many English speakers in Israel speak English with Hebrew words in it: *"grab some bamba from the makolet"*, *"I have to go to Bituach Leumi"*. This is code-switching. English dictation models were trained on English and have rarely or never heard these words, so they substitute the nearest English-sounding word: *Macaulay* for *makolet*. Often the one word that carries the meaning is the one that comes out wrong.

Hebrish is a fine-tune that keeps English dictation as it was and writes the Hebrew words correctly. "Correctly" needs a definition, because Hebrew words in English text have no single agreed spelling. Hebrish uses one canonical romanized spelling per word, from a public spelling lexicon (`terms.csv`), and writes Hebrew in Latin letters only. Parakeet's tokenizer is a 1,024-token SentencePiece BPE with cased tokens and punctuation but no Hebrew characters, so Hebrew script was never an option; romanization also suits the use case, which is English text.

The base model is NVIDIA Parakeet TDT 0.6B v2 (CC-BY-4.0), a 0.6B-parameter FastConformer TDT model. It is small enough to run on a laptop CPU and a phone, which is where dictation happens.

2. Dataset

Training data is [Read-Aloud-Voice-Sentences](#) (public, CC-BY-4.0), at revision `550bfcaa0f84`. Every clip is **one speaker** (Daniel Rosehill, Irish-born, English-speaking, living in Jerusalem) reading a prompted sentence aloud, mostly on one phone, as 16 kHz mono audio with a cased, punctuated transcript and a list of its target terms. Code-switched sentences carry the Hebrew terms; plain-English sentences are recorded in the same voice for the forgetting check and for the training mix.

Split at <code>550bfcaa</code>	Clips	Minutes	Use
Code-switching train (<code>cs_train</code>)	3,224	339.2	Training
Code-switching validation (<code>cs_val</code>)	218	22.6	Best-epoch selection
Own-English pool	751	95.3	Training (all of it)
LibriSpeech-PC replay pool	900	190.7	Training (48.5 min drawn)
Code-switching test (<code>test_cs</code>)	387	43.9	Evaluation
Own-English test	73	9.2	Forgetting check
LibriSpeech test-clean	200	24.6	Forgetting check (other speakers)
Long-form test	15 files	44.6	Long recordings

Training mix

The run trained on 4,202 rows, 483.0 minutes: **339.2** minutes of code-switching, **95.3** of the speaker's own English and **48.5** of LibriSpeech-PC replay. The English share was targeted at 30% and came out at 29.8%. The share follows the code-switching and continual-learning literature: fine-tuning only on new-domain data is what causes catastrophic forgetting, and mixing in the original domain plus some replay of general data is the usual remedy.

Held-out terms

The split is **term-disjoint for a held-out list**, not for every term. 362 terms are on the held-out list at this revision; a sentence containing any of them goes to test, so those terms never appear in training. A test sentence can also contain other terms that are not on the list, and those can appear in training. The evaluation therefore reports two numbers: recall on held-out terms (293 occurrences of 118 terms in `test_cs`), which measures generalisation, and recall on test terms also seen in training (278 occurrences), which measures learning. Only the first is evidence that the model writes a word it was never taught.

3. Training checkpoint 6 (v1.0)

v1.0 is run `r6-en30`, trained on 2026-10-19. It is a **full fine-tune from stock** `nvidia/parakeet-tdt-0.6b-v2` on the whole dataset as it stood, not a continuation of PoC-5.

Setting	Value
Framework	NeMo 2.7.3 (<code>nemo_toolkit[asr,cu12]</code>), torch 2.11.0, pinned in a Modal image
Start point	<code>nvidia/parakeet-tdt-0.6b-v2</code> , tokenizer kept
Optimiser	AdamW, lr 3e-5, betas 0.9 / 0.98, weight decay 1e-3
Schedule	Cosine annealing, 10% warmup, min lr = lr / 20
Epochs / batch	20 epochs, batch size 16
Precision / seed	bf16 mixed precision, seed 7
Selection	Best epoch on <code>cs_val</code> : epoch 18 of 20, validation WER 5.0% (raw, 0.0497)
Hardware / time	One NVIDIA L40S (48 GB) on Modal, 41.0 minutes

Why a cumulative full fine-tune, not continued training

Each version starts from stock Parakeet and trains on every take recorded so far. Continuing the previous fine-tune on only new data is what causes catastrophic forgetting, and continuing on all data gives old rows another round of epochs each release, so drift compounds. A run costs about US\$0.80, so nothing is saved by continuing. Training and benchmarking are separate steps: the best epoch is chosen on `cs_val` (part of train, never test), and the benchmark runs afterwards on test.

Why 30% English

A dictation model that learns Hebrew words but loses English is no use. The English share keeps the original task in every batch: the speaker's own English teaches that plain English in this voice should stay plain English, and LibriSpeech replay keeps other voices. Section 6 checks whether it worked.

Lineage

Checkpoint	Run	CS min	Released	Benchmarked
PoC-1 · Amba	<code>pilot-en30</code>	77.4	2026-10-15	Yes
PoC-2 · Bamba	<code>r2-en30</code>	120.1	2026-10-15	Yes
PoC-3 · Chamin	<code>r3-en30</code>	196.1	2026-10-18	Yes
PoC-4 · Ful	<code>r4-en30</code>	269.3	2026-10-18	No
PoC-5 · Halva	<code>r5-en30</code>	294.2	2026-10-18	No
v1.0 · Jachnun	<code>r6-en30</code>	339.2	2026-10-19	Yes, 2026-10-19

PoC-1 to PoC-5 were published on Hugging Face as v1 to v5. The sixth checkpoint is called v1.0 because it is the first release the project stands by. A Nemotron streaming variant (run `r6-nemotron-en30`, base `nvidia/nemotron-speech-streaming-en-0.6b`, best epoch 18, validation WER 6.3% raw, 45.8 min) was trained from the same data with the same recipe.

4. Exports

v1.0 is published in the formats on-device runtimes use. Each export was checked on five clips against fp32 ONNX before any benchmark.

Format	Where	Runs in	Check (5 clips vs fp32)
<code>.nemo</code>	main model repo	NeMo (PyTorch)	Reference
GGUF F16, Q8_0, Q4_K_M	<code>Hebrish/...-GGUF</code>	transcribe.cpp, Handy, Offline Voice Input	F16 5/5, Q8_0 5/5, Q4_K_M 4/5
ONNX fp32 and int8	<code>Hebrish/...-ONNX</code>	onnx-asr, Voxtype, the in-browser demo	int8 4/5
sherpa-onnx int8	<code>Hebrish/...-sherpa-onnx-int8</code>	sherpa-onnx, the Hebrish Android app	4/5
MLX	<code>Hebrish/...-mlx</code>	parakeet-mlx (Apple Silicon)	Still PoC-3; not re-exported

The int8 misses are one word: *tzatzumim* where fp32 wrote *satzumim*; the reference is *tza'atzuim*. The benchmark numbers below are for the `.nemo` in PyTorch, not the quantized exports.

5. Evaluation

v1.0 was scored on 2026-10-19 on two test sets that answer different questions.

Public benchmark

The [English-Hebrew code-switching ASR benchmark](#) (dataset `Hebrish/English-Hebrew-Codeswitching-ASR-Benchmark`, test set v1, frozen) has 145 clips. The headline **clean subset** is 84 phone recordings (11.8 min) of sentences in no published training set, with 172 Hebrew term occurrences (139 distinct). A separate **English control** subset of 24 clips is the same speaker reading English only.

- **Term recall:** share of term occurrences whose canonical spelling appears in the transcript as a whole word, case-insensitive, hyphen and space interchangeable. **Unseen-term recall** restricts it to terms

absent from the public training split (117 of the 172 occurrences).

- **WER:** lowercased, hyphens to spaces, punctuation removed except in-word apostrophes (`jiwer` 3.1.0).
- **Contamination check:** none of the 145 benchmark clips, by id or by text, is in v1.0's training manifest (`runs/r6-en30/train.jsonl`, 4,202 rows), nor in the Nemotron run's.

Every result file is scored with the same normaliser and term matcher; the Nemotron fine-tune was scored offline (full context), not in streaming mode.

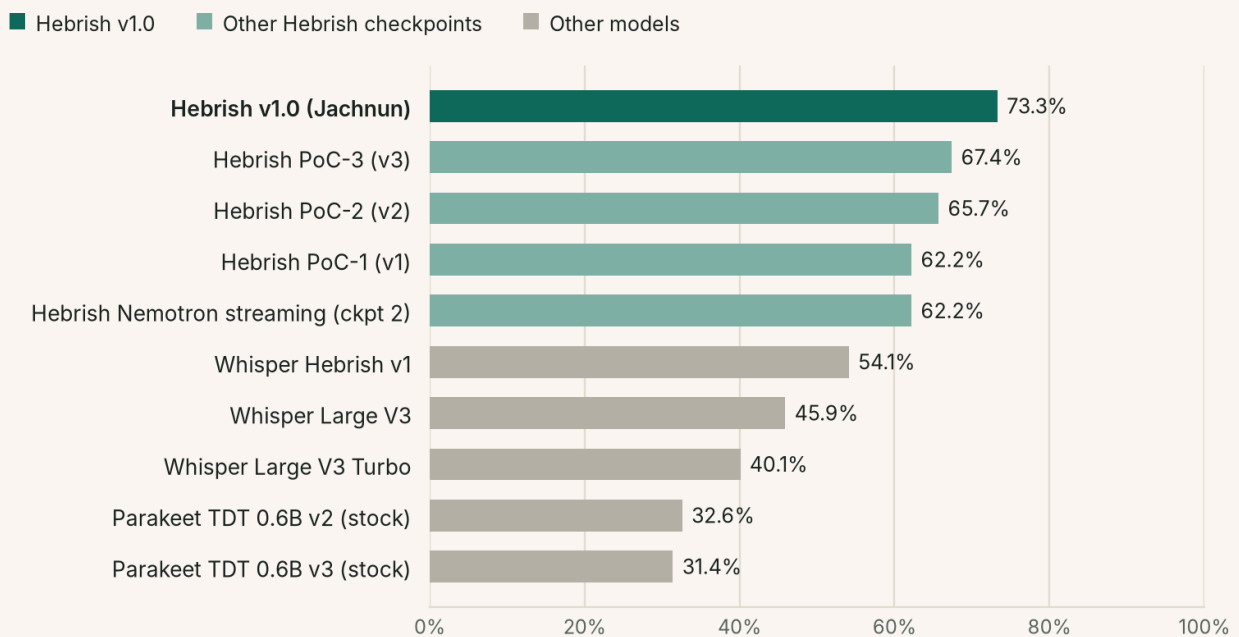
Held-out split

The project's own test split at revision `550bfcaa0f84` (Section 2) is larger: 387 code-switched clips, the held-out-term and seen-term recall described above, the own-English and LibriSpeech forgetting checks, and 15 long-form files. Stock v2, PoC-3 (the last benchmarked release), v1.0 and the Nemotron run were all re-scored on this revision, so they compare like for like. WER here is normalised. These numbers are not comparable with the older PoC results, which used revision `b47310621f79`.

6. Results

Hebrew term recall, public benchmark

Public benchmark test set v1, clean subset: 84 clips, 172 term occurrences. Scored 2026-10-19.



Source: English-Hebrew-Codeswitching-ASR-Benchmark, round 3 (results/*.json). Nemotron scored offline. hebrish.app/benchmarks
One speaker's voice: every test clip is read by the same speaker, so these numbers rank models on that speaker only.

Figure 1. Hebrew term recall on the public benchmark, clean subset, every scored model.

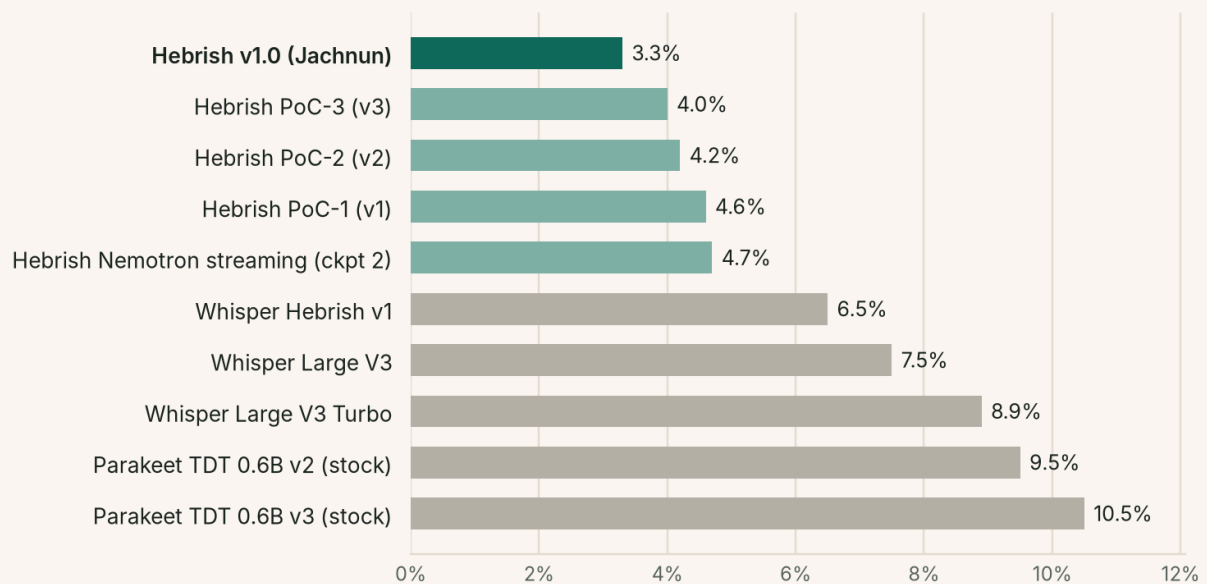
Model (public benchmark, clean subset)	Term recall	Unseen recall	WER	English control WER
Hebrish v1.0 (Jachnun)	73.3%	70.1%	3.3%	2.2%
Hebrish PoC-3	67.4%	63.2%	4.0%	2.5%
Hebrish PoC-2	65.7%	60.7%	4.2%	2.4%
Hebrish PoC-1	62.2%	61.5%	4.6%	2.5%
Hebrish Nemotron streaming, checkpoint 2 (offline)	62.2%	54.7%	4.7%	2.5%
Whisper Hebrish v1	54.1%	56.4%	6.5%	4.7%
Whisper Large V3	45.9%	46.2%	7.5%	2.4%
Whisper Large V3 Turbo	40.1%	39.3%	8.9%	2.4%
Parakeet TDT 0.6B v2 (stock)	32.6%	32.5%	9.5%	2.7%
Parakeet TDT 0.6B v3 (stock)	31.4%	31.6%	10.5%	2.7%

Round 3 of the leaderboard, scored 2026-10-19. On 172 occurrences, a 2 to 3 point difference in recall is about 4 to 5 hits and within noise.

Word error rate, public benchmark

Public benchmark test set v1, clean subset: 84 clips, 172 term occurrences. Scored 2026-10-19.

■ Hebrish v1.0 ■ Other Hebrish checkpoints ■ Other models



Source: English-Hebrew-Codeswitching-ASR-Benchmark, round 3 (results/*.json). Nemotron scored offline. hebrish.app/benchmarks

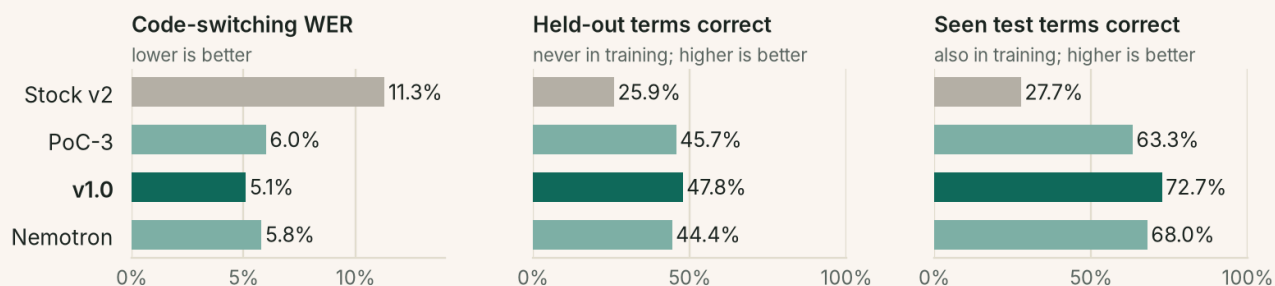
One speaker's voice: every test clip is read by the same speaker, so these numbers rank models on that speaker only.

Figure 2. Word error rate on the public benchmark, clean subset.

Same split, four models

Held-out split at dataset revision 550bfcaa0f84: 387 code-switched clips. Scored 2026-10-19.

■ Hebrish v1.0 ■ Other Hebrish checkpoints ■ Stock



Source: Parakeet-V2-Personal-Finetune data/evals/r6-*-summary.json. Normalised WER. hebrish.app/benchmarks

One speaker's voice: every test clip is read by the same speaker, so these numbers rank models on that speaker only.

Figure 3. Stock v2, PoC-3, v1.0 and Nemotron on the same held-out split (revision 550bfcaa0f84).

Held-out split, 550bfcaa	CS WER	Held - out terms (293)	Seen terms (278)	Own EN	LibriSpeech	Long - form
Stock Parakeet v2	11.3%	25.9%	27.7%	3.1%	1.2%	9.6%
Hebrish PoC-3	6.0%	45.7%	63.3%	2.3%	1.0%	5.8%
Hebrish v1.0	5.1%	47.8%	72.7%	1.7%	1.2%	5.0%
Hebrish Nemotron (offline)	5.8%	44.4%	68.0%	3.3%	2.0%	5.8%

What the results say

- **Hebrew words.** v1.0 more than doubles stock Parakeet's term recall on the public benchmark (73.3% against 32.6%) and leads every scored model, including Whisper Large V3 (45.9%) and the earlier Whisper fine-tune (54.1%). Unseen-term recall (70.1%) is close to overall recall, which suggests the gain is not only memorisation of training terms.
- **Code-switched WER** fell from 11.3% (stock) to 5.1% on the held-out split, and from 9.5% to 3.3% on the public benchmark.
- **Against PoC-3,** v1.0 is better on every public-benchmark metric. On the held-out split, the gain on test terms seen in training (+9.4 points) is real; the gain on held-out terms (+2.1 points, about 6 of 293 hits) is within noise. More data taught the words it contained; it did not clearly improve guessing words it never heard.
- **Nemotron,** the streaming variant scored offline, trails v1.0 on every metric and wrote none of the 8 number tokens in the test correctly. It has no forgetting verdict, because stock Nemotron was not re-scored on this revision.

English did not get worse

Forgetting check on dataset revision 550bfcaa0f84. Scored 2026-10-19.

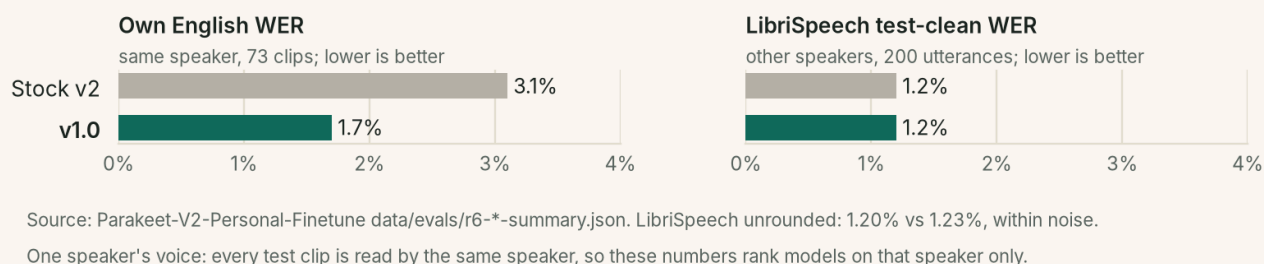


Figure 4. Forgetting check. Own English is the same speaker; LibriSpeech is other speakers.

English did not get worse. The speaker's own English WER fell from 3.1% to 1.7%, and the public benchmark's English control from 2.7% to 2.2%. On LibriSpeech, other people's English, v1.0 scored 1.20% against stock's 1.23%: a noise-level margin, which is the point. v1.0 passed the project's forgetting gate.

Proof-of-concept checkpoints

Stock, PoC-1, PoC-2 and PoC-3 on the older test split, revision b47310621f79 (219 clips).

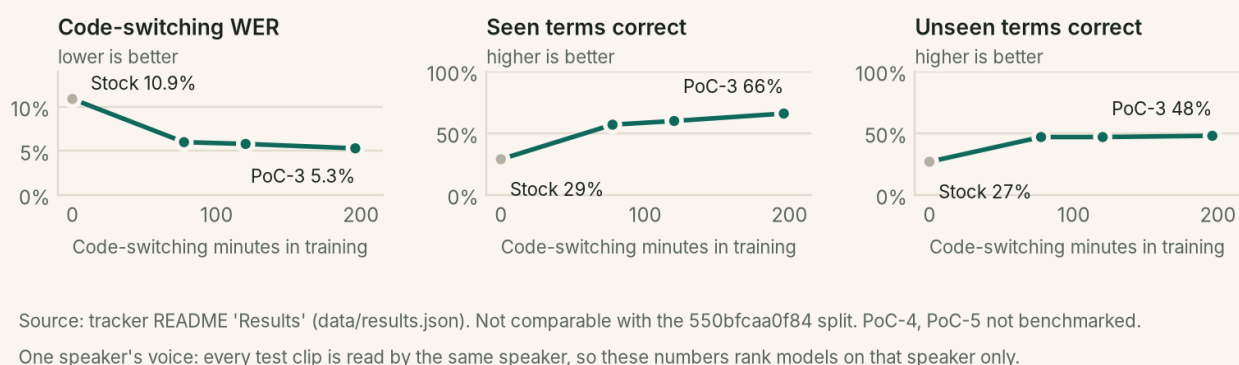


Figure 5. The proof-of-concept checkpoints on the older test split (revision `b47310621f79`, 219 clips). Not comparable with Figures 3 and 4.

7. Limitations

- **One speaker's voice.** Every clip, train and test, is the same speaker: Irish-born, English-speaking, living in Jerusalem. The results rank models on this speaker. They say nothing yet about other accents. More voices are the project's main need.
- **Read speech, mostly one phone.** Prompted sentences read aloud are cleaner than spontaneous dictation.
- **Small test sets.** 84 clean clips and 172 term occurrences on the public benchmark; a 2 to 3 point recall difference is within noise. The held-out split is larger (387 clips, 293 held-out occurrences) but has the same speaker.

- **Not every benchmark term is Hebrew.** 22 of the 139 clean-subset terms are English tech labels or product names (*JSON, grep, Proxmox, ...*), 26 of the 172 occurrences. Excluding them changes no ranks but lowers every model's recall by 2 to 6 points. A relabelled test set will be a new version.
- **A possible held-out leak is being checked.** Two held-out-list terms, *mandatim* and *Beit Safafa*, appear as training target terms at a later dataset revision (`7bfcea1d`). It is not yet confirmed how they were routed at `550bfcaa` . If they were trained on, their held-out hits are not unseen.
- **Nemotron was scored offline**, with full context, not in the streaming mode it exists for. Its streaming accuracy is unmeasured.
- **Held-out gains over PoC-3 are within noise** (Section 6). PoC-4 and PoC-5 were never benchmarked.
- **Quantized exports** were checked on five clips each, not benchmarked. MLX is still PoC-3.
- **The canonical spelling is a convention.** A model writing *tsaharon* for *tzaharon* is not wrong in an absolute sense; the benchmark reports lenient recall alongside.

8. Cost

Step	Compute	Cost
Training run <code>r6-en30</code>	41.0 min on one L40S (Modal)	About US\$0.80 (Modal billing for earlier runs of this class: US\$0.71–0.86)
Benchmark round, 2026-10-19	Six parallel L40S jobs, 19:44–19:46 UTC, about 2.5 min wall each, under 15 GPU-minutes in all	About US\$0.50 at most

The binding constraint is recording, not compute. Retraining and benchmarking a model this size costs about US\$1.30 in compute, so the project retrains on cumulative data whenever enough new code-switched audio has been recorded.

9. Links

- Website and benchmarks: hebrish.app/benchmarks · datasets: hebrish.app/datasets
- Model: huggingface.co/danielrosehill/Parakeet-TDT-0.6B-v2-English-Hebrew-Codeswitching
- Exports: huggingface.co/Hebrish (GGUF, ONNX, sherpa-onnx int8, MLX, Nemotron)
- Training data: [Read-Aloud-Voice-Sentences](#)
- Leaderboard: [English-Hebrew-Codeswitching-ASR-Leaderboard](#) · benchmark data: [Hebrish/English-Hebrew-Codeswitching-ASR-Benchmark](#)
- This paper: hebrish.app/whitepaper/hebrish-v1.0-whitepaper.pdf

Daniel Rosehill, 2026-10-19. Text licensed CC-BY-4.0. The model inherits CC-BY-4.0 from `nvidia/parakeet-tdt-0.6b-v2` ; LibriSpeech-PC, used for replay, is CC-BY-4.0. All results are from one speaker's voice. Numbers are copied from the project's result files without rounding beyond their published precision.