METHOD NOTE

How the Hebrish training sentences are made

Where the words come from, how sentences are generated and rejected, and how they are split for testing

AUTHOR
Daniel Rosehill and Opus 5.5

DATE
02/11/2026

VERSION
1.0

In short

Hebrish models learn Hebrew words as English speakers say them inside English sentences (*makolet*, *arnona*, *Bituach Leumi*). There was no corpus of that kind of speech, so one was built. Its sentences are short English sentences, each written around one or two target Hebrew terms. Most are written by large language models from a target term and its context, checked by rules and a second model, then read aloud by a person. A sentence a person would not say is rejected at any of those steps. No audio is synthetic: every clip is a human reading.

The recordings are one speaker's voice (Daniel Rosehill's). More voices are welcome: see hebrish.app/get-involved.

> **Numbers in this note**
> Counts come from the dataset's metadata (`meta/train.jsonl`, `meta/test.jsonl`) on Hugging Face, `danielrosehill/English-Hebrew-Codeswitching-Sentences` at revision `ad2f62c6`, read on 2 November 2026: **6,812 clips, 717.0 minutes**. The dataset grows most days, so later revisions differ.

Where the words come from

The vocabulary is the Hebrish lexicon: **3,642 Hebrew terms in 13 domains** (everyday life, bureaucracy, religious life, the army, food and more), browsable at hebrish.app/lexicon. Each term has one canonical romanized spelling, the way English speakers in Israel commonly write it, and a short meaning. The first list of about 1,100 terms was expanded from Daniel's earlier sentence dataset and dictation history with LLM suggestions; it has grown since through themed vocabulary packs (news, Jewish texts, everyday gaps, Ashkenazi pronunciation).

Where the sentences come from

Source	Clips	Share
Generated in bulk by an LLM from a target term and its context (the sentence bank)	4,357	64.0%
Generated on demand: Daniel picks the word in the recording app (optionally with a context hint) and an LLM writes the sentence	1,763	25.9%
Taken from existing datasets, mainly Daniel's earlier English-Hebrew-Mixed-Sentences (262)	606	8.9%
Typed or written by Daniel himself	86	1.3%
Total	**6,812**	**100%**

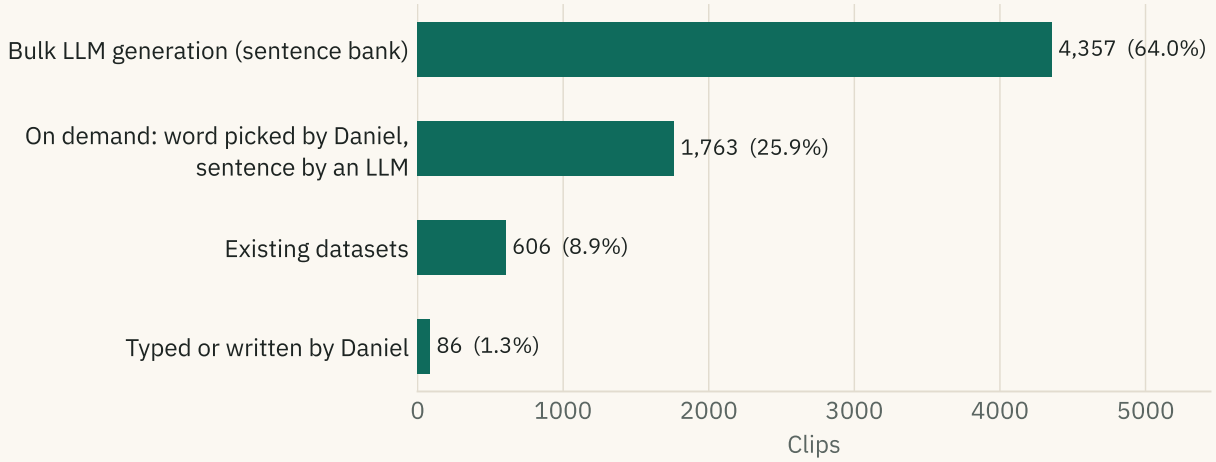


Figure 1: Where the sentence text of the 6,812 clips came from. (`prompt_origin` and `prompt_source`, dataset revision `ad2f62c6`, 02/11/2026)

Field `prompt_origin` in the metadata holds the source class; `prompt_source` names the generator. So about 90% of the sentence text is LLM-written, and for about a quarter of all clips the target word was chosen by hand. An earlier count on the site (19 October 2026) listed 1,403 sentences as "written by hand"; that figure counted the on-demand sentences, whose word is chosen by hand but whose text is generated. This note replaces it.

How a sentence is generated

The generator gets two kinds of input:

- **The target term**, spelled exactly as in the lexicon, with its meaning. Without the meaning, a pilot with a small model misused about one Hebrew word in four (14 October 2026).
- **Surrounding context**: a register and scenario (a WhatsApp message to a neighbour, an email to an accountant), the vocabulary pack's instruction, and for some terms example phrasings people use, marked "vary them, do not copy".

The prompt requires the term exactly as given (never translated, glossed or written in Hebrew script), used naturally so the context makes its meaning clear, and tells the model to answer `SKIP` rather than force a usage it is unsure of: a wrong usage is worse than a missing one. Sampling temperature is 0.9 to 1, so sentences for the same term differ in structure and setting.

Models and inference APIs

When	Generator
From 14 Oct 2026	Claude Sonnet 5.5 through OpenRouter, ten items per request (2,179 clips' sentences). Claude Code subagents and Claude Opus 5.5 wrote the themed packs (1,700 and 216); 57 came from an early DeepSeek V4.1 Flash pilot.
From 21 Oct 2026	Claude Haiku 5.5 through the **Anthropic Message Batches API** (205 so far), moved to the **OpenRouter Batch API** on 1 Nov 2026.
From 2 Nov 2026	Claude Haiku 5.5 through synchronous OpenRouter calls: batch jobs queued for 5 to 10 minutes, too slow for the recording loop.
On demand	A fast model through OpenRouter (DeepSeek V4.1 Flash by default, Qwen as fallback), synchronous, because the speaker is waiting for the sentence.

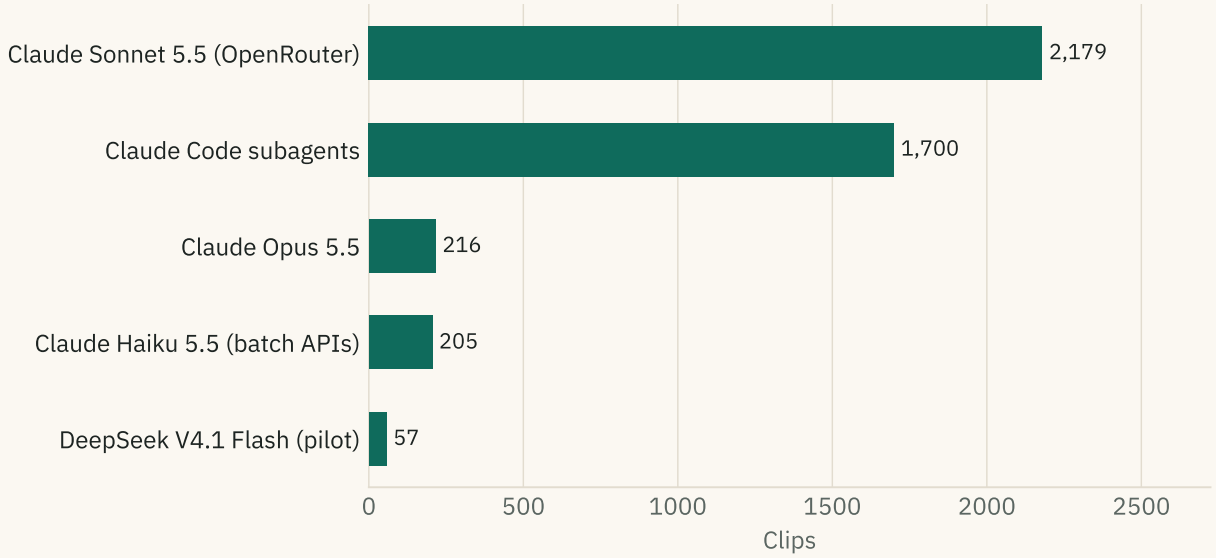


Figure 2: Generated sentences in the dataset by generator (bulk generation only; the 1,763 on-demand sentences are not broken down by model). (`prompt_source`, dataset revision `ad2f62c6`, 02/11/2026)

Rejection policy

A sentence has to pass three gates before it is trained on.

1. **An automatic gate** (all generators, since 2 November 2026). It rejects a sentence that has Hebrew script, brackets or characters a target model cannot output; is shorter than 6 or longer than 18 words; does not contain each target term exactly once, matched as a whole word; mixes a Hebrew term with a tech term (kept apart so each can be measured); or writes a number of 10 or more in words. A rejected sentence is regenerated.
2. **A reviewer from another model family.** Gemini judges Claude's sentences (temperature 0) and drops any where a Hebrew word is used wrongly or is not one an English-speaking Israeli would drop in there, the sentence does not sound like something a person would dictate, terms are forced together, or it is ungrammatical. A sentence the reviewer cannot rule on after three tries is dropped. From 2 November 2026 this pass is optional, because the third gate catches the same problems.

3. **The reader.** Every sentence is read aloud by a person, who skips any that misuse a word or sound unnatural. Each take is played back on the phone and kept, retaken or deleted. A misread is fixed in the transcript (flag `edited`), so the text always matches the audio. Speaker labels are checked with voice embeddings (hebrish.app/dataset#integrity).

Spelling and transliteration

Each term is trained in its canonical romanization. Where a term has other common spellings (667 of the first 1,167 terms list some), the bank rewrites them to the canonical form before a sentence is used, so the model learns one target spelling per word. Variants that are ordinary English words are left alone.

The one deliberate exception is pronunciation: **Ashkenazi and Israeli forms** (*Shabbos* and *Shabbat*, *Tosfos* and *Tosafot*) are kept as separate spellings in 93 spelling groups, because they sound different. 792 clips carry that label: 656 Israeli, 135 Ashkenazi, 1 mixed.

Under consideration
A wider sample of spellings, generating sentences with the different ways people actually transliterate a word, could make the model more robust to how users write. If tried, a later dataset revision may prune variants whose transliterations conflict, so each spoken word still maps to one written form.

How the dataset grew

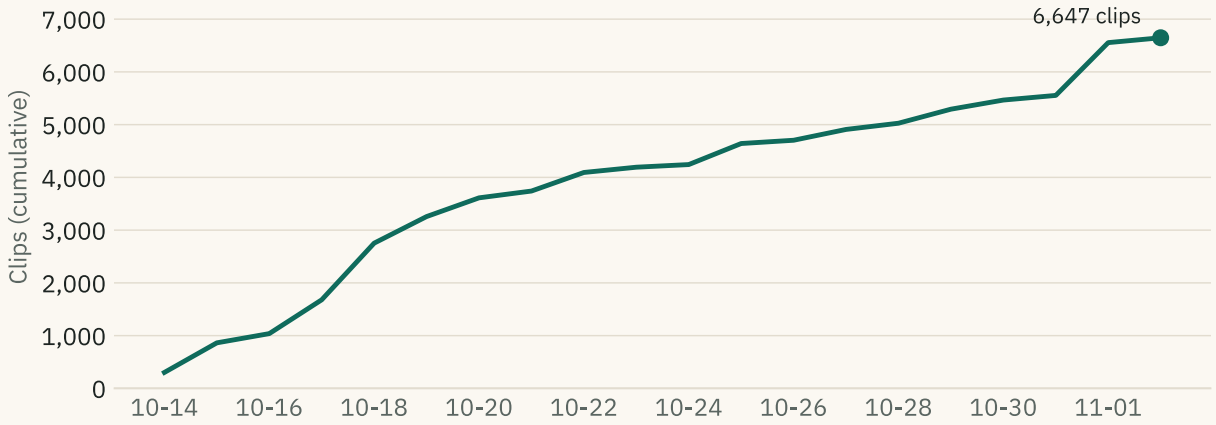


Figure 3: Recorded clips, cumulative, by recording date. (`data/stats.json` on hebrish.app, dataset revision `af910143`, 6,647 clips, read 02/11/2026; the newer revision counted above has 6,812)

Splits and testing

Terms, not sentences, decide the split. About one term in twelve (by a hash of the term) is held out: any sentence containing a held-out term goes to the test split, so models are scored on Hebrew words they never heard in training as well as on words they did. Words Daniel picks or types himself always go to training.

The public benchmark, **test set v2** (Hebrish/English-Hebrew-Codeswitching-ASR-Benchmark, 25 October 2026), is 129 code-switching clips balanced across the 13 domains, none used by any training run, with 221 term occurrences, 137 of them unseen in training, plus 25 English-only and 25 tech control clips. Full results: hebrish.app/benchmarks.

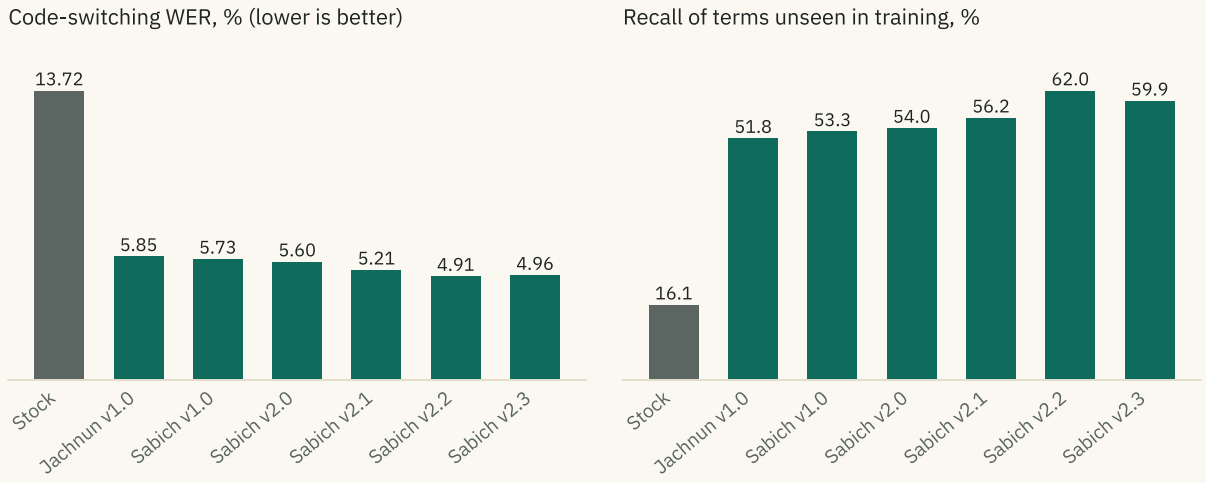


Figure 4: Test set v2: stock Parakeet TDT 0.6B v2 against the Hebrish models trained on this data. Sabich v2.3, the current release, is slightly behind v2.2 on this test set and ahead of it on the held-out split; the differences are small. (`data/results.json` `test_v2`, benchmark dataset revision `48ec85a301dc`; v2.3 scored 30/10/2026)

Questions about the method: hebrish.app/contact. Dataset card and schema: hebrish.app/dataset.