Episode Book

EPISODE #6148

The Tiny Hebrew Models Behind the Show's Voice

PUBLISHED	19 October 2026 at 20:08 UTC
DURATION	30:04
LISTEN	myweirdprompts.com/episode/hebrew-niqqud-task-models-77112/
HOSTS	Herman, Corn, Larry, Hilbert
SCRIPT MODEL	anthropic/claude-sonnet-5.5
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.10
BOOK GENERATED	20 October 2026 at 07:46 UTC

SYNOPSIS

Hebrew speech pipelines lean on small task models that quietly decide whether the output is right — a nikudizer, a phonetic converter, a spelling normalizer. This episode digs into Dicta's DictaBERT-large-char-menaked, Phonikud's adaptors on top of it, UNIKUD's CANINE-based approach, and why the model that converts defective to full spelling only exists as a byproduct of training another tool. Along the way: a live pipeline that renamed a host, a silent failure mode where unvowelled text slipped into a voice engine, and the Unicode trap that eats verse endings if you strip niqqud the lazy way.

TRANSCRIPT



Herman

Three hundred million parameters, and we're calling it small. I love this corner of the field.



Corn

Last time we did the broad survey of Hebrew speech recognition, text to speech and language models, and we promised the other half, the layer underneath. Daniel wants the spotlight on the tiny task models that sit in front of the big ones and quietly decide whether the Hebrew comes out right. What are they, who built them, why does so much hang on them. And he started with a hunch that the nikudizer behind the show's own Hebrew editions was a pretty old model.



Herman

It's a fair hunch, because the project it grew out of is old. Nakdan, from Dicta, was presented at the ACL conference in twenty twenty. Avi Shmidman, Shaltiel Shmidman, Moshe Koppel and Yoav Goldberg. It's neural, but with hand built linguistic knowledge mixed in, a tagger for parts of speech and morphology, lexicon features, and an editing interface so a human can overrule it. The model we actually use is a different thing. Dictabert large char menaked was created on Hugging Face in March of twenty twenty five, and it was last touched this September.



Corn

Nineteen months. In this field that's vintage.



Herman

It is, a little. The model is a BERT, about three hundred million parameters, fine tuned from the character level DictaBERT large, licensed CC BY. And the card claims state of the art as of March twenty twenty five, against open source models and commercial LLMs both.



Corn

Claims with a table, I hope.



Herman

No table. That's the thing. The card gives no scores at all, only the claim. I've seen it compared in other people's papers, and I'll get to that, but Dicta's own page just says it.



Corn

A trophy with no scoreboard. Fine, so what does it do?



Herman

It takes modern Hebrew prose with no vowel marks and puts the marks in. Niqqud, the little dots and dashes under and over the letters. And the card is clear about its limits. Modern prose only. Not Biblical Hebrew, not Rabbinic, not premodern, not poetry. There's also an option to mark the letters that are doing double duty as vowels, the yod and the vav, with a star, and by default it doesn't. That's my reading of the card, so don't hold me to the detail.



Corn

Who is Dicta, anyway? Name on a card is not the same as knowing who it is.



Herman

The Israel Center for Text Analysis. The authors are affiliated with Bar Ilan University as far as I can tell. Whether it's a non profit, I couldn't tell you, I haven't seen that confirmed anywhere. And there's a third party wrapper, not from Dicta, which the card itself points out, that runs the model through ONNX. About a tenth of a second per sentence on an M one laptop, with two output modes. Male, which is the full spelling, and haser, which is the defective one. The page states no licence, so I'd treat that wrapper with some care.



Corn

A tenth of a second on a laptop. That's the unglamorous part, then. Nobody's buying a GPU cluster to put dots on words.



Herman

Nobody should. Dicta's model isn't the only one, though. UNIKUD is from a few years ago, from Morris Alp. It's built on Google's CANINE, a character level transformer, and it treats the whole job as a multilabel problem. Every letter gets fifteen yes or no questions, does it take this vowel, does it take that dot. And the README boasts it uses absolutely no rule based logic.



Corn

Bold thing to put on a README for a language with this many rules.



Herman

Nakdimon is the other open one. Elazar Gershuni and Yuval Pinter, a two layer character level LSTM, no human curated resources beyond vowelled text, MIT licence, Findings of NAACL in twenty twenty two. Small and plain. When later papers put Dicta and Nakdimon side by side on the Nakdimon test set, Dicta wins, but the tables disagree on the gap. In one it's about ninety six percent word accuracy against eighty seven. In another the gap is much narrower. You can only compare inside a table.



Corn

I came in expecting one model. Four now. Is it going to be forty?



Herman

There's more. MenakBERT, D Nikud, a word level one called DiVRit. And Phonikud, which sits on top of Dicta's model, and I want to do that one properly. It freezes the Dicta model, bolts on lightweight adaptors, and invents extra marks of its own, one for stress and one for the vocal shva. A rule based step then turns the lot into phonetic symbols for a voice engine. On their pronunciation test the word error rate is seventeen point four percent. Dicta's vowelled text with default stress scores thirty nine point five.



Corn

Sounds like a landslide.



Herman

On that test, yes, but it measures pronunciation, not vowel accuracy, so don't read it as the same thing. Frontier LLMs were in that table as an upper bound too, and I can't tell you which row is which, the extraction came out garbled.



Corn

You brought a stack again, didn't you.



Herman

A small stack. It's relevant.



Corn

Fine. Skip to what happened when we ran this for the show, because I was there, and I remember being renamed.



Herman

You were renamed, yes. Corn, tell it.



Corn

Same input, two runs, different output. That was the first thing. We fed the pipeline a sentence, got vowelled Hebrew back, ran it again and got something else. Then the imperatives. Shilchu, send it, came back as shalchu, they sent it. Ktvu, write, came back as katvu, they wrote. The letters are identical, only the dots differ, and the dots chose the wrong tense. And my name. Corn came back as Koren.



Herman

A real name. Just not yours.



Corn

It sounds like a man who sells insurance. And there was a quieter problem. The built in hook for Chatterbox, the voice engine, was sending the text through unvowelled, and it never said so. No error, just flat, ambiguous Hebrew going into the voice model, which then guessed.



Herman

That one's nasty because the failure is silent. The audio comes out, it sounds like Hebrew, and nobody knows it's wrong unless they listen with a native ear. In the end the team pinned the niqqud by hand. The vowelled text is fixed, so the voice engine never has to trust a model at runtime.



Corn

Why did two runs differ? Is the model nondeterministic?



Herman

I don't know. Nothing I've seen says it is, and the card is silent on it. There are two guesses and neither is tested. One is that the quantised ONNX export behaves differently from the original float model. The other is the same thing from the other side. The cheap check is to run one sentence twice on a fixed CPU session and compare against the original model in full precision. I haven't done it.



Corn

It went wrong and nobody checked why.



Herman

The status is it went wrong, we pinned it, and the check is on the list.

L

Larry

Are you tired of dots? Of dots above your letters, dots below your letters, dots simply lying around your sentences? Introducing the Vowel Vanisher Platinum Deluxe, the only machine that removes niqqud, ktiv, kvetch and quite possibly the weekend. Now, the regular price is nine hundred ninety nine dollars, but today, only today, I'm dropping it to four ninety nine. No wait, three ninety nine. Call in the next ten seconds and it's two forty nine, and I'll throw in a second Vowel Vanisher, a canvas tote, a lifetime supply of the letter bet, a pair of lightly used oven gloves, and a signed photograph of a CANINE, a very good dog, not the transformer. Two hundred dollars! One fifty! And that's not all. You also get a goat. A small goat, house trained, fluent in prefix segmentation, and he does not bite on weekdays. Now the price is one hundred, but I can go to forty, and I will, because I love you, and as a bonus you get my cousin's boat, which sails, mostly, and the first two thousand callers get the deed to a small island in the Baltic called Dikduk, which is a real place, and it has a lighthouse, and the lighthouse is also yours, and it's now twenty dollars, ten, five, and if you act now I will personally, and here the offer expires, I am being told the offer has ended, I was only on the island.



Herman

Hang on, the first claim. Removing niqqud is a one line job in Unicode, so what is the machine even.



Corn

Herman. Leave it.



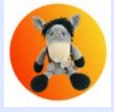
Herman

Fine. Sorry. Removal, though. Daniel was unsure who built a vowel remover, and this is where the story has a twist.



Corn

Go on. I assume the twist is that nobody built one.



Herman

Nearly. Stripping the marks is plain Unicode deletion. Vowel points and the cantillation marks are all combining marks in one block of Unicode, and a few lines of code removes them. There's even an npm package for it, unniqqud, that can take out the niqqud, the



Herman (continued)

cantillation, or both. No model, no training, nothing to download.



Corn

Nothing to spotlight, then.



Herman

There's one trap. Two of the characters in that block, the paseq and the sof pasuq, are punctuation, not marks. A lazy range strip eats those too, and you've deleted the verse endings from your Bible text.



Corn

A vowel remover that also takes the full stops. I'd call that a feature if I'd been editing the Bible.



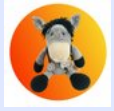
Herman

Now the part that is a model. Stripping gives you text with the vowels gone. It doesn't give you text the way a modern Israeli would actually write it. Because Hebrew has two spelling systems. Ktiv haser, defective spelling, which is what you see in vowelled texts. And ktiv male, full spelling, which is what people write when there are no vowels. In full spelling you add extra yods and vavs to carry the vowel sounds that the dots used to carry.



Corn

The terminology is a trap, if I remember. Wikipedia calls the vowelless system ktiv male.



Herman

It does, and it's also the modern standard. The Academy of the Hebrew Language codified the unpointed spelling rules in nineteen ninety six, and Hspell, the open source spellchecker, follows them. But nobody turned that into a model for ordinary use. The clearest open example I can find lives inside UNIKUD, and it's there by accident.



Corn

By accident.



Herman

By necessity, really. To train a vowel adder you need pairs, text with the marks and the same text without. The vowelled texts are all in defective spelling. Real unvowelled Hebrew is in full spelling. If you simply strip the dots off the vowelled text, you get inputs that look nothing like what the model will see in real life, because the extra yods and vavs aren't there. So UNIKUD's author trained a separate ktiv male model. It's fine tuned from CANINE, and for every character it picks one of three things. Add nothing, add a yod, add a vav. There's a function in the source that takes Hebrew with nikud and returns Hebrew in full spelling without it.



Corn

So the vowel remover with a brain is a byproduct of the vowel adder.



Herman

A tool built to make homework for another tool. Only yods and vavs are modelled, not alef or he. And I haven't confirmed that the ktiv male checkpoint is published separately. The only model linked is the main one.



Corn

Does anyone want this for itself? Not as training scaffolding.



Herman

That's where I'd like your view, actually. Biblical Hebrew. A scholar has a vowelled Torah text and wants to search it the way you'd search modern Hebrew. Would you run a respelling model on it?



Corn

No. I'd say you'd wreck the evidence. If the spelling is how scholars date the text, you can't quietly modernise it.



Herman

That's what the literature says. Torah spelling is distinctively defective, scribes copied the plene and defective forms unchanged, and so normalising changes the witness. A study on medieval Hebrew found Dicta's tool produced near perfect vowelling with one genuine error, but it also altered the orthography, and so did ChatGPT four. And the menaked card, remember, says Biblical isn't what it's for. I found no model that converts defective to full spelling for Biblical work. That gap is real.



Corn

A gap I'm happy about, on balance. A tool that's wrong about Isaiah is worse than no tool.



Herman

Let's go to the second class, which is about telling what language and what script you're looking at. Daniel half remembered a classifier for the Hebrew character set and mixed text. And I looked, and there isn't a single model like that.



Corn

Not one?



Herman

The nearest real thing is GlotLID, version three, from the CIS group, a fastText model, Apache licence. Its labels include Hebrew in Hebrew script, with an F one of about point nine nine four. Biblical Hebrew in Hebrew script, about point nine nine eight. Yiddish, and Ladino written in Hebrew letters, which scores lower, around point nine five, but there were only two hundred and two test rows. So it really does tell Hebrew, Yiddish, Ladino and Biblical Hebrew apart when they share an alphabet.



Corn

That's the half memory, then. A script based language identifier.



Herman

Probably, and it comes with a companion, GlotScript, which reports the main script and every script in a sentence. That's the character set part. One warning from the README, it's trained on longer sentences and doesn't do well on very short ones.



Corn

Short ones, like a word.



Herman

Like a word, exactly the inconvenient case. And what it doesn't have is a label for Hebrew in Latin letters. It has Arabic in Latin letters, Hindi in Latin letters, Bengali, Kashmiri. No heb Latn. As far as I can tell, the other big ones, fastText's older identifier and Meta's NLLB one, have no such label either, though I haven't checked those two myself. And I found no Hebrew and English code switching corpus anywhere. So before Daniel's own model, nothing handled Hebrew words in Latin letters inside English text.



Corn

His, from this week. We've gone through how he built it before, so just place it for me.



Herman

Released on the fifteenth of October. A token classifier, XLM Roberta base, that tags spans. Bituach Leumi, makolet, bamba, in the middle of an English sentence. Span F one of point nine four one on a test set where the terms never appear in training, so memorising words can't flatter the score. And zero false positives on the hard negatives. The whole thing cost under three dollars to generate and annotate.



Corn

Under three dollars for a gap three big identifiers left open. Unfair to the big identifiers.



Herman

The baseline he expected to land near, an English Tamil recipe, scored lower on its hardest class. His beat it. Training took about three minutes.



Corn

Why does a dictation pipeline need it at all?



Herman

Makolet is why. He dictated it, the speech to text gave back something like macaullete, and then an English voice reads it as mack oh let. Hebrew word, English letters, and nobody downstream knows it's Hebrew. So the pipeline detects the span first. Pass two turns Latin letters back into Hebrew script, lexicon first, and a model called TaatikNet as the fallback. Pass three adapts the text for each voice engine. Chatterbox gets split into segments with a language tag. The engines that take markup get a Hebrew language tag around the word.



Corn

Hold on. TaatikNet. That's a new name.



Herman

Here's the thread. TaatikNet is Morris Alp's too. Same person who made UNIKUD. A small byte level T five model, trained on about fifteen thousand Wiktionary word pairs, doing transliteration in both directions between Hebrew and Latin letters. Word level only, and it doesn't follow spelling conventions.



Corn

Meaning.



Herman

Makolet, the corner shop, comes back with a kuf where the real word has a kaf. And Bituach Leumi comes out as something no one at the National Insurance Institute would recognise. It's a fallback for a reason, so the lexicon goes first. Corn, would you trust a lexicon over a model for a word like makolet, or the other way round?



Corn

Lexicon, every time. A list that's wrong is wrong the same way twice, and you can fix it. Which leaves me noticing one person built the vowel adder, the respelling model hiding inside it, and the one transliterator this classifier feeds. That's a lot of the unglamorous layer resting on one set of shoulders.



Herman

And a mirror image of the problem has older prior work. Goldberg and Elhadad in two thousand eight did foreign words written in Hebrew letters, around eighty percent precision and recall, though that's a paper, not a model. And there's a more recent paper on Judeo Arabic, flagging Hebrew and Aramaic words inside Arabic script text. Real word level code switching work, but not English and not a model you can pull down.



Corn

Is there an audio version? Because dictation is where the trouble starts.



Herman

Planned. A word level language identification model for audio, in the planning stage. I'd say wait and see.



Corn

Third class, then. I've been waiting for this one, because it's the one I understand least.



Herman

One written Hebrew word can be several words. Hebrew glues little words onto the front. And, the, that, in, like, to, from. Vav, he, shin, bet, kaf, lamed, mem. So uvabayit, and in the house, is one token on the page. A search engine sees one string. A language model sees one string. A voice engine has to decide how to say it. And almost every tool further down the line needs to know where the real word starts.



Corn

Is it always visible? The, say.



Herman

No, and that's the lovely part. The definite article is sometimes only implied, and marked by the vocalisation rather than by any letter. The segmentation model handles that too. Dictabert seg splits a word like beshnat, in the year of, into be and shnat, and a form like uve toldot into uve and toldot.



Corn

Dicta again?



Herman

Same team. Created in August twenty twenty three, a bit under two hundred million parameters, CC BY. Eight classifiers per word, one for each prefix function, and at inference it only considers prefix combinations that are valid for the word's first letters. Here's a link I love. The training labels were derived from Dicta's vowelled corpus, then corrected by an expert annotator.



Corn

So the dots told it where the prefixes were. The vowel layer is feeding the segmentation layer.



Herman

And sometimes the reverse. The lemmatiser's card tells you to strip prefixes with the segmenter first, before lemmatising names it hasn't seen. One utility is another utility's preprocessing step. Prefixes only, mind you, not suffixes.



Corn

A family, I'd guess. One encoder, a pile of heads. That's how these things go.



Herman

That's it, same encoder, different heads. Morphology, lemma, named entities, parsing, sentiment, question answering, and one that does five jobs in one call, dictabert joint. Segmentation, morphology, lemma, syntax and entities, with each head switchable. A tiny joint version at about forty five million parameters. And the most downloaded are not the glamorous ones. Joint is the top at around seventeen thousand, entity recognition second at about ten thousand. Compare that with the char menaked model, a little over three thousand.



Corn

The thing with the headline gets a third of the traffic of the thing that does the boring job.



Herman

Nobody writes a press release about a prefix splitter. And before these, the best Hebrew segmenter in twenty eighteen was YAP, a transition based parser in Go from Reut Tsarfaty's group. AlephBERT then AlephBERTGimmel gave the field good Hebrew encoders, and the AlephBERT authors made the point that most Hebrew benchmarks need morpheme boundaries that a plain language model doesn't output.



Corn

Is it perfect?



Herman

No. A twenty twenty five paper on tokenisation for languages like Hebrew reports that DictaBERT seg removes too few letters on stacked prefixes, and sometimes too many, which they pin on words it hasn't seen. I've only seen that in the abstract and a snippet, so hold it loosely. And I don't have a headline accuracy figure to give you for the segmenter against the older tools. I won't make one up.



Corn

You keep circling speed. Say it.



Herman

This is the answer to why they stay small. The Hebrew parsing paper from Dicta and Reut Tsarfaty's group has the tiny model at about a twentieth of a second per sentence on a plain thirty two core CPU, one sentence at a time. The previous best published system took about a quarter of a second. Over a hundred times faster, and the stated motive was that existing models are simply too cumbersome for real time.



Corn

A twentieth of a second on a CPU. That runs on whatever's under the desk.



Herman

Which is the argument. Speed and CPU are one half. Determinism is the other. A classifier gives you spans with offsets and a threshold. Daniel's pass three is deliberately not a model, and he wants it that way. When something goes wrong you can point at the span. Compare that with two runs of a vowel adder giving different answers.



Corn

To be fair, that was a small model that went wrong, not a large one.



Herman

It was, and I'd rather you say it than me. It could be the export, not the model. I'm not claiming small equals reliable.



Corn

The LLM question, then? Because someone's going to say a frontier model does all of this now.



Herman

The one measured comparison I have is question answering. In the DictaBERT paper, ChatGPT scores about nine and thirty two on the Hebrew question answering set, DictaBERT about sixty four and seventy three. I'm assuming the two columns are exact match and F one, since the labels aren't confirmed, and which ChatGPT version it was, I can't say. There are also hints from coreference work that small encoders beat decoder LLMs on Hebrew, and drop off further on raw unsegmented text. That one I only saw as a snippet.



Corn

And for niqqud, segmentation, language ID?



Herman

No head to head that I could find for segmentation or language identification. For vowels it's mixed. In the pronunciation benchmark from the Phonikud paper, a frontier model scored at or near the small specialised system, but I can't pin the rows. And a paper this July had Gemini three Pro catching about eighty four percent of the confirmed errors in the standard homograph test set, with precision near ninety nine. The benchmark itself was wrong in places, two hundred and seventy one times.



Corn

The exam was wrong, and the model that noticed was the big one.



Herman

Fair point for the large ones. They're good auditors. They're slow, and they aren't something you want in a loop.



Hilbert

Does it put the freckles in a letter at a time, or the whole word at once? Herman, you said character level and then you said "I'll get to the tables" and then you got to the tables, so I'll wait for the answer. I rang my Aunt Philomena about it twenty minutes ago. She has four hundred and twelve jars.



Herman

A letter at a time. Each character gets its own prediction, that's what the char in the name means.



Hilbert

That's what Desmond does. Desmond Pettigrew. He does her freckles. She runs a pickle stall on Corporation Road, and the jars go to a man in Wick who will only read a label with the freckles on. Desmond does them with a hot darning needle, one letter at a time, from a stepladder, because Aunt Philomena keeps the stock on hooks in the ceiling. There's a sprinkler. It has been pointed at the Hoyles' stall since a Tuesday.



Corn

A hot needle under a ceiling of pickle. I'd want the sprinkler pointed at Desmond.



Hilbert

Mr Aldous Hoyle has a treaty with her. Signed on a jar lid. Under the treaty Desmond is paid in tea cosies, left ones only. The right ones go to Hoyle, who has no use for them, he has no teapot. He keeps them in a drawer in alphabetical order by the date they stopped



Hilbert (continued)

being wool.



Herman

That makes the stall a two party economy with cosies as currency and a throughput ceiling of one letter per needle touch. You can't parallelise a darning needle.



Hilbert

Eleven letters a minute. I timed it from the doorway. He gets up at ten past four, boils two eggs, eats one. The other goes in his hat. It's Hoyle's egg, they swap on Fridays. Then he sits on the stepladder until Aunt Philomena comes in and says the word freckles. That's his cue to start. Without the word he doesn't. He took a Thursday off in March because she said spots.



Corn

A man whose whole job hangs on one noun. Somebody should tell his employer which noun.



Hilbert

She knows. She said it to me down the phone just now and I heard the needle start.



Herman

Down the phone, from the stall. So there's a sound cue as well as a word cue.



Hilbert

There's a second man, Mr Hoyle's nephew, who does the right cosies. He's eight and he has a van. The van is a hatchback with the back seats out and a kiln in it. He drives it to the stall at four miles an hour behind a man with a flag, because the kiln is lit.



Corn

A lit kiln in a hatchback. I'd want the flag man very well paid.



Hilbert

Paid in cosies. The nephew has never seen a jar. He posts the cosies through the letterbox of a house that was demolished, and Desmond collects them from the skip. Aunt Philomena has the receipts, in a biscuit tin, filed by the colour of the wool. The tin came with a manual. The manual is for a different tin. I've got the invoice here. Desmond's. Handwritten, on the back of a tide table. He counted the freckles on every label and wrote the total at the bottom. Eleven days. Nine pounds forty. He asked for it in coins.



Herman

Eleven days for the batch. A model does a sentence in about a tenth of a second.



Corn

People assume that once the freckles are in, the voice engine knows how to say the word.



Herman

They leave out stress. The Phonikud paper's example is a word that reads bira. It can be beer, stressed on the first syllable, or the capital city, stressed on the last. Same letters, same vowel marks, two words. That's why Phonikud had to invent its own marks, and why a newer system from the same group, ReNikud, learns from thousands of hours of spoken Hebrew instead. Written grammar vowels don't match how people actually talk. The code is promised, I haven't seen it released.



Corn

Misparim, numbers. Misparayim, scissors. Mesaprim, they tell. One spelling, three words, three stresses.



Herman

Which is the unglamorous truth of the whole layer. The big models get the headlines, and the little ones decide whether your brother is called Koren.



Corn

What I'm left with is the eleven days. A model clears a batch of labels in seconds, so the cost moves from the work to the checking. Someone with a native ear has to listen to the output on a schedule, or a silent hook skips the freckles again and nobody hears it.



Herman

Thanks to our producer, Hilbert Flumingtop, for keeping the desk running. This has been My Weird Prompts.



Corn

Find us at my weird prompts dot com. We'll be back soon.