Episode Book

EPISODE #6630

Using LLMs as Annotators for Silver Training Data

PUBLISHED	16 November 2026 at 14:48 UTC
DURATION	23:38
LISTEN	myweirdprompts.com/episode/llm-as-annotator-silver-labels-38841/
HOSTS	Corn, Herman, Larry, Hilbert
SCRIPT MODEL	z-ai/glm-5.3
VOICES	omnivoice
PIPELINE	V5.33
SOURCES	23 cited by the research (listed at the end)
BOOK GENERATED	16 November 2026 at 22:22 UTC

SYNOPSIS

When an LLM labels data for a downstream classifier, it isn't acting as a judge or classic weak supervision—it's an LLM-as-annotator producing silver labels. Exploring a Romanized Hebrew span-detection pipeline built with Claude Opus, this episode separates few-shot annotation and human-AI refinement loops from Snorkel-style labeling functions. It also covers the Alternative Annotator Test for spot-checking and an Activision Call of Duty study showing that using a seed set to search 66 prompt configurations outperforms fixed few-shot examples.

TRANSCRIPT



Corn

There's a sentence buried in the middle of English text, and inside it, sitting there like it belongs, is "Shalom" spelled with Latin letters. Not Hebrew script, not a translation. Just Hebrew wearing an English coat. Daniel's been building a classifier that finds those spans, and the interesting part isn't the classifier. It's how he's making the training data for it.



Herman

This is the part I like, honestly. The model is almost an afterthought. Everything upstream of it is where the decisions live.



Corn

Right. His method: he hand-labels a seed set, five hundred examples or fewer, in an annotation workspace. Then he hands the rest to Claude Opus, tells it, in his words, "you've learned the pattern, label the rest." And then a fork of that: he spot-checks, say every tenth item, and when it hasn't quite got it, he feeds a corrected subset back and asks for a redo, and they repeat until the labels hold. He wants to know what this family of methods is called, how the variants differ, and where each one fits. He also names the two neighbours it gets filed next to, LLM-as-a-judge and classic weak supervision, and says it isn't either. Which is correct, and it'll be easier to see why once we hear what the method actually does.



Herman

Start with the fact that turns the whole map. Daniel is producing training data. He's not evaluating a model. Whatever we call what he's doing, that single fact, what happens downstream of the label, is the hinge.



Corn

And the umbrella term exists. LLM-assisted annotation, or LLM-as-annotator. The output has a name too, silver labels, as opposed to the gold human ones.



Herman

The seed-set variant specifically is called few-shot annotation in the guides, and the working range people publish is roughly a hundred to two hundred examples as a floor, with production deployments going up to about a thousand. Daniel's five hundred sits comfortably inside that.



Corn

Comfortably inside. Five hundred examples, and he calls that the small version.



Herman

For a span-detection task it is the small version! Try hand-labeling five hundred sentences with span boundaries sometime. That's an afternoon of squinting at minimum.



Corn

An afternoon. You'd have it done before my first coffee.



Herman

Here's the honest finding, though, and I want to put it up front: there is no single canonical name for the exact family Daniel describes, the one-shot version and the loop version as siblings. The survey that tries to organise this field, Tan's, from twenty twenty-four, organises by pipeline stage, how annotations are generated, how their quality is assessed, how they're used downstream. Not one-shot versus iterative. Nobody has taxonomized Daniel's two variants as a named pair.



Corn

So the naming is unsettled.



Herman

Which is part of why it gets filed next to its neighbours, the judge and the weak supervision. If the family had its own clean name, the neighbours wouldn't keep absorbing it.

L

Larry

Friends, when the labels get confusing and the judges and annotators start to blur together, reach for Silverbright, the little pill that turns your messy spot checks into gleaming, trustworthy silver. Just one capsule with breakfast and your seed set labels itself, your spot checks come back already corrected, and the frontier models start agreeing with you out of sheer respect.

L

Larry

Silverbright is not for everyone. Side effects may include a mild tingling in the labeling functions, sudden confidence that every tenth item is fine, dry mouth when asked what happens downstream, a faint ringing in the heuristic, occasional spontaneous reannotation of your grocery list, and in rare cases a slow, warm feeling that the literature has finally settled on a single canonical name. Ask your doctor if Silverbright is right for your pipeline.



Corn

"A faint ringing in the heuristic."



Herman

Please don't read it back in the slow voice.



Corn

Right, the judge. The neighbour causing the most confusion.



Herman

The line is actually crisp once you draw it. LLM-as-annotator produces labels or metadata for downstream analysis or training. LLM-as-a-judge evaluates the outputs of another system. That's it. Applied to Daniel: Claude Opus marking Romanized Hebrew spans is an annotator, because those labels become training data for his classifier. If Daniel instead took his finished classifier's predictions and asked Claude Opus, is this span detection correct, yes or no, that's a judge. Same model, same prompt style, different job.



Corn

The tell is never the model. It's what the label is for.



Herman

Training data versus a metric. And the confusion isn't carelessness, it's structural. The ACL Anthology itself says the two paradigms "grow increasingly indistinguishable," because the same model, the same few-shot prompting, the same spot-checking discipline serves both roles. There's a paper called AURA from this year that applies iterative human-verification refinement to judge decisions, selected human verification, uncertainty-aware refinement. The loop shape is identical to Daniel's. Only the target differs.



Corn

Same loop, different plumbing at the end.



Herman

Which is why people file Daniel's method under judging. They see the loop and stop looking.



Corn

The weak supervision border is the one I'd want drawn more carefully, because that's the neighbour with real machinery inside it.



Herman

Classic weak supervision, the Snorkel style, asks an expert to write labeling functions. Actual programs. Heuristic rules, keyword matches, external knowledge bases. Each function is noisy and they conflict, and a probabilistic label model denoises the conflicts without any ground truth, and out the end comes a training label. That's the whole architecture. Functions, votes, denoiser.



Corn

Daniel writes no functions.



Herman

None. No labeling functions, no label model, no denoising step. The frontier model's in-context pattern-matching replaces the hand-written rules, wholesale. One prompt instead of forty functions.



Corn

So he's got the shape of weak supervision with the engine swapped out.



Herman

That's a fair compression of it, yes. And the adjacency is real, because both approaches replace per-item human labeling with a scalable, noisy, somewhat programmatic signal. Some of the guides frame it as three generations of annotation: manual labeling, then programmatic weak supervision, then LLM-powered labeling. Daniel skipped the middle generation entirely.



Corn

Went from the horse to the jet and never owned a car.



Herman

And here's the part I find elegant. There's a paper called Language Models in the Loop that builds the bridge literally. It treats the language model as the basis for labeling functions. You prompt the model with multiple distinct queries about each example, map the responses to votes and abstentions, and then run Snorkel's denoiser over the votes. The LLM as function, Snorkel as arbiter. They report about nineteen and a half percent average error reduction over zero-shot on a standard weak supervision benchmark.



Corn

Nineteen and a half percent for asking the same model several different questions.



Herman

Different questions, mapped to votes. Which is actually the interesting part for later, so hold onto that.



Corn

Noted. The other thing I want on the table is the multilingual angle, because Daniel's task is weird by construction.



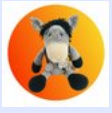
Herman

It is, and this is an under-covered corner. Of six hundred fifty papers that mention LLM-as-a-judge, thirty-three covered low-resource or multilingual settings. That's from an EMNLP Findings paper this year. Five percent. And Romanized Hebrew, Hebrew in Latin letters, a code-switching slash transliteration phenomenon, is squarely inside the uncounted part. As far as I can tell, nobody has studied LLM annotation of Romanized Hebrew at all.



Corn

Nobody's looked.



Herman

Which means Daniel is working without a map, which is either a warning or an invitation depending on temperament.



Corn

With him, an invitation. Now, the spot-check. Every tenth item. Is that a folk practice or does it have a formal version?



Herman

It has a formal version, and it's better than folk. Calderon, Reichart and Dror, ACL last year, the Alternative Annotator Test. Their opening complaint is that "there is no standard or rigorous procedure to determine whether LLMs can replace human annotators," which is a remarkable sentence when you think about how many people are doing exactly that replacement. Their procedure, the alt-test, "requires only a modest subset of annotated examples to justify using LLM annotations." You check the model against a modest held-out set of human labels, and the test tells you whether substitution is defensible.



Corn

Daniel's every-tenth check is the folk version of that.



Herman

Close cousin. The difference is that the alt-test is designed to justify, and the every-tenth rule is designed to discover. One asks permission, the other goes looking for trouble. Same instruments, different questions.



Corn

That's the boundaries, then. Judge, weak supervision, and where Daniel actually sits. Now get inside the family, because the two variants aren't the same method at two prices.



Herman

They're not, and the literature does name them separately even if it never names them together. The one-shot version is few-shot annotation, sometimes LLM label synthesis. The loop version is a human-AI definition refinement loop, or an adjudication loop. And the closest published analogue to Daniel's exact loop is a paper I'd trust on this, because it wasn't a toy. Kocielnik and colleagues, prosocial behavior detection in Call of Duty game chat, deployed with Activision.



Corn

Real deployment, real chat logs.



Herman

Ten thousand real messages a day, and the content is what you'd expect from game chat. Their pipeline had two explicit steps, and the steps are the two variants. Step one: they aligned GPT-4o outputs with a seed of about a thousand human-labeled messages using an initial definition of prosociality. That's one-shot seed-and-label, Daniel's first approach, almost exactly.



Corn

Almost exactly, except they did something with the seed that Daniel, from his description, hasn't.



Herman

This is the part I got excited about, so let me get it out. They used the seed set to choose the prompt. They ran a grid search over sixty-six unique prompting configurations before committing to anything. Three models, definition in or out of the prompt, four personas, two label terms, and then separately, retrieval with different numbers of examples per class. Sixty-six configurations, all scored against the human-labeled seed.



Corn

Sixty-six.



Herman

The cheapest high-value use of a seed set isn't teaching the model the pattern. It's choosing the prompt. Daniel's five hundred examples could be doing that work, and instead they're sitting in the prompt as fixed few-shot examples.



Corn

He showed the model five hundred flashcards and called it a day.



Herman

Well, five hundred wouldn't fit anyway. Typical few-shot annotation uses three to twenty examples in the prompt. But that's the finding from the grid search that transfers directly: fixed examples lost to retrieved ones. They tested retrieval with two up to twenty-four examples per class, pulled by embedding similarity from an anchor set of two hundred. The retrieved examples beat the fixed ones.



Corn

Because the examples near the hard case are the informative ones.



Herman

The similar ones teach the boundary. Fixed examples teach the average. And there's a second detail from that grid that I want to flag for Daniel's task specifically, because it's about naming. They compared domain-relevant label terms against neutral placeholders, so PROSOCIAL versus LABEL-1. The real label skewed model predictions and precision. The label name itself primed the model.



Corn

Hold on. Daniel's label is "Romanized Hebrew."



Herman

Which may prime the model toward anything shaped like transliteration. Names, brand names, italicized foreign words. The very boundary cases he most needs to resolve are the ones the label name is pushing the model to accept.



Corn

The label is arguing with the data.



Herman

Quietly, in every prediction. So that's variant A, and what it's actually good for. Step two of the prosocial pipeline is where variant B earns a separate name. After the one-shot pass, they ran an adjudication loop between GPT-4o and expert annotators, and the point of the loop was not correcting errors. It was identifying ambiguity, refining the task definition, and relabeling the seed collaboratively.



Corn

The loop isn't quality control. It's definition discovery.



Herman

That's the reframe. And note what they did not do: they did not sample randomly. They prioritized high-disagreement cases. Cases where the model and the humans split, because that's where the definition is fuzzy. Daniel's every-tenth rule is the other fork. Random sampling gives you an unbiased estimate of overall error, which is exactly what the alt-test needs to justify substitution. Disagreement-targeted review gives you better labels per human-hour, but a biased quality estimate, because you're only looking where it's already hot.



Corn

One tells you how bad it is. The other tells you where it's bad.



Herman

You want both, and my strong suspicion is most practitioners do only one. The random samplers know their error rate and keep it. The disagreement hunters fix everything they touch and never learn the overall rate.



Corn

Is there evidence the loop is worth the effort? Actual numbers.



Herman

Two papers' worth. CentaurTA Studio, a human-agent system for thematic analysis, ran the ablation that answers this question almost too cleanly: removing the human feedback loop drops performance from ninety percent to eighty-one percent. Nine points for the loop. And they peaked within ten iterative rounds, about twenty-five minutes of human time, which tells you the loop saturates fast. It's not twenty-five minutes a day forever.



Corn

Nine points for twenty-five minutes.



Herman

And the prosocial deployment numbers: after refinement they synthesized ten thousand labels with GPT-4, then trained a two-stage system where a cheap classifier handles the easy cases and only about thirty-five percent of ambiguous instances escalate to the full model. Inference cost down seventy percent, precision around point nine zero.



Corn

Which is the happy half of the ledger. I assume there's an unhappy half, because there always is.



Herman

There is, and it's uncomfortable. Mohta and colleagues at Amazon, NeurIPS workshop a few years back. They trained models on LLM-generated labels and on human labels for the same tasks, and the human-label models consistently won. Zero point eight four versus zero point seven five on one task, zero point six four versus zero point six zero on another.



Corn

Not catastrophic, but consistent.



Herman

Consistent, across the board. And the finding that matters for Daniel: performance "significantly diminishes for tasks in languages other than English." His task is multilingual by construction. It's two languages fighting over one alphabet. Silver labeling is a budget decision, volume traded against quality, not a free lunch, and the exchange rate gets worse the further you get from English.



Corn

The cheap labels are cheapest exactly where he's using them.



Herman

And the loop has a hidden cost of its own that nobody prices in. Definition drift. The prosocial paper warns that iteratively updating prompts or definitions "can introduce subtle semantic shifts that erode label consistency over time." They recommend version control. Think about what that means concretely: if Daniel revises his prompt each round, and his loop runs five rounds, the labels from round one and round five may encode different definitions of Romanized Hebrew. His classifier doesn't learn either definition. It learns the union.



Corn

The loop that fixes the labels quietly un-fixes the definition.



Herman

Yes. The instrument drifts while you're calibrating it.



Corn

So where does each fit? Give me the actual map.



Herman

One-shot seed-and-label when the task definition is already crisp, the label space is small, and you can validate after the fact. Use the seed to choose the prompt, not just to teach the model. The iterative loop when the task is novel or underdefined, and Romanized Hebrew is exactly that. Is "Shalom" Romanized Hebrew or an English loanword? Is a brand name that happens to look like transliteration a positive? Those questions don't exist until you look at disagreements. That's what the loop is for.



Corn

And weak supervision?



Herman

When you can articulate the pattern as rules. And here's the thing about Daniel's task: he can, at least partly. Regex for Hebrew transliteration patterns is feasible. Character n-grams, vowel patterns, common morphemes. And if he wrote labeling functions, he'd get versioned, debuggable labeling logic, plus a label model reconciling the noisy signals. Snorkel's user study had experts building models 2.8 times faster than hand labeling, an average performance gain of about forty-five percent, coming within three and a half points of large hand-curated training sets.



Corn

Rules and a referee instead of vibes at scale.



Herman

Judging is the odd one out, because it's a different stage of the pipeline entirely. You use a judge when you already have the trained classifier and want to measure it. Daniel will get there, and the same discipline will serve.



Corn

Which leaves the variant neither of his approaches covers.



Herman

The hybrid. The literal reading of Language Models in the Loop. He prompts Claude Opus with several distinct queries per span. Is it Hebrew? Is it a loanword? Is it a proper name? Maps the responses to votes and abstentions, runs a label model over the votes. Multiple noisy opinions, one denoiser, no human in that pass at all.



Corn

He's got the judge's tools, the annotator's job, and weak supervision's referee sitting unused.



Hilbert

Mostly right, both of you.



Hilbert

The every-tenth thing, though. That's not new. I did that by hand for my nephew's shop inventory, nineteen ninety... no, doesn't matter. Cards, one item per card, green pen for the checkmark. My nephew, Yankele, he wanted every tenth card pulled. I told him tenths are for people who don't know where the trouble is. I pulled the cards where the description didn't match the bin label, and I kept the ones I corrected in a separate envelope, because a corrected card isn't the same as a checked card and I wasn't going to pretend it was.



Corn

You refiled your own corrections.



Hilbert

Obviously. There was a grinder listed as Polish make, all the paperwork said Polish, and it was a German grinder with the plate swapped. Looked exactly like the category it was filed under. That's the card that made the envelope worth keeping. Yankele did nine cards in the time I did four, and his were wrong more often, which he never wrote down and I did.



Herman

The envelope. What happened to it?



Hilbert

Lost in the move, with the green pen. My sister's boxes. Herman, you said random sampling gives the honest error rate, and you said it like a man who has never had to explain an error rate to a nephew.



Herman

I have had to explain things to Corn, which is similar.



Corn

Lower stakes, though. He's family.



Herman

The envelope is the part I'll be thinking about, actually. His corrected cards weren't the same as checked cards, and he was right, they're different objects. Daniel's corrected annotations are the same thing: after round one, his dataset contains two species of label, seed-gold and loop-corrected silver, and most pipelines just stir them together.



Corn

Which is what the classifier learns from. The stir.



Herman

And nobody's published a head-to-head of one-shot versus iterative annotation on the same task and budget. The CentaurTA ablation is the closest thing, and that's a feedback loop inside a thematic analysis system, not two annotation strategies on a classification task. That gap is exactly where Daniel's classifier lives. He's not choosing between two named, benchmarked options. He's choosing between two things the literature hasn't compared.



Corn

On the naming question, then, the answer is: this family has no settled name, half a dozen near-synonyms, and a five percent multilingual coverage rate. Which means when Daniel runs his Romanized Hebrew experiment, he is, in a small way, filling a hole the field hasn't touched. First entry in a blank row.



Herman

Somebody has to be the first card in the envelope.



Corn

Somebody does. Big thanks, as always, to our producer Hilbert Flumingtop, who keeps the desk, the levels, and apparently a private taxonomy of corrected cards. For more along these lines, there's episode six, How To Fine Tune Whisper. This has been My Weird Prompts. If today's walk through the silver-label neighborhood earned its keep, leave us a review wherever you get your podcasts. It helps.



Herman

We'll be back soon.



Corn

See you tomorrow.

SOURCES

What the research for this episode read before the script was written. Primary sources first.

1. **Calderon, Reichart & Dror, The Alternative Annotator Test for LLM-as-a-Judge, ACL 2025 Long Papers pp. 16051 [primary]**

   16081, retrieved 2026-11-16, published 2025-07.
   https://aclanthology.org/2025.acl-long.782/

2. **Kocielnik et al., Prosocial Behavior Detection in Player Game Chat, arXiv:2508.05938v1 [primary]**

   retrieved 2026-11-16, submitted 2025-08-08.
   https://arxiv.org/abs/2508.05938

3. **Mohta, Ak, Xu & Shen (Amazon), Are large language models good annotators? PMLR v239 (NeurIPS 2023 workshop) [primary]**

   retrieved 2026-11-16, 2023.
   https://proceedings.mlr.press/v239/mohta23a/mohta23a.pdf

4. **Ratner et al., Snorkel: Rapid Training Data Creation with Weak Supervision, arXiv:1711.10160v1, VLDB 11(3):269 [primary]**

   282, retrieved 2026-11-16, submitted 2017-11-28.
   https://arxiv.org/abs/1711.10160v1

5. **Language Models in the Loop: Incorporating Prompting into Weak Supervision, arXiv:2205.02318 / ACM IMS JDS [primary]**

   and https://dl.acm.org/doi/full/10.1145/3617130, retrieved 2026-11-16.
   https://arxiv.org/abs/2205.02318

6. **Wang, Huang & Dragut, CentaurTA Studio, arXiv:2604.18589v1 [primary]**

   retrieved 2026-11-16, submitted 2026-03-18.
   https://arxiv.org/abs/2604.18589v1

7. **Ascione & Tamagnone, From scratch to silver: patent-SDG classification using LLMs, arXiv:2509.09303v1 [primary]**

   retrieved 2026-11-16, submitted 2025-09-11.
   https://arxiv.org/abs/2509.09303v1

8. **Zhou et al., SALUS: Automated Auditing of NL-to-SQL Benchmarks, arXiv:2610.05540v1 [primary]**

   retrieved 2026-11-16, submitted 2026-10-04.
   https://arxiv.org/abs/2610.05540v1

9. **Zhang et al., AURA: Adaptive Uncertainty-aware Refinement for LLM-as-a-Judge Auditing, arXiv:2606.19714v1 [primary]**

   retrieved 2026-11-16, submitted 2026-06-18.
   https://arxiv.org/abs/2606.19714v1

10. **Wei et al., Auditing Engagement Incentives in the Kidfluencer Ecosystem, arXiv:2606.03173v1 [primary]**

    retrieved 2026-11-16, submitted 2026-06-02.
    https://arxiv.org/abs/2606.03173v1

11. **Shih et al., BEATS: Bootstrapping E-commerce Attribute Taxonomies, arXiv:2606.04909v1, SIGIR 2026 Industry Track [primary]**

    retrieved 2026-11-16, submitted 2026-06-03.
    https://arxiv.org/abs/2606.04909v1

12. **Cao et al., Condor: Enhance LLM Alignment with Knowledge-Driven Data Synthesis and Refinement, arXiv:2501.12273v1 [primary]**

    retrieved 2026-11-16, submitted 2025-01-21.
    https://arxiv.org/abs/2501.12273v1

13. **Structured Exploration and Exploitation of Label Functions for Automated Data Annotation, arXiv:2604.08578 [primary]**

    retrieved 2026-11-16.
    https://arxiv.org/html/2604.08578

14. **Tan, Large Language Models for Data Annotation: A Survey, arXiv:2402.13446v2 [primary]**
retrieved 2026-11-16.
https://arxiv.org/html/2402.13446v2

15. **From Selection to Generation: A Survey of LLM-based Active Learning, ACL 2025**
retrieved 2026-11-16, 2025.
https://doi.org/10.18653/v1/2025.acl-long.708

16. **Challenges and Recommendations for LLM-as-a-Judge in Multilingual Settings, EMNLP 2026 Findings**
retrieved 2026-11-16, 2026.
https://aclanthology.org/2026.findings-emnlp.2021.pdf

17. **SourceBae, LLM-as-Annotator: AI Data Labeling with LLMs 2026 Guide**
retrieved 2026-11-16, 2026.
https://sourcebae.com/blog/llm-as-annotator-data-labeling/

18. **SourceBae, Weak Supervision for Data Annotation: Guide to Programmatic Labeling 2026**
retrieved 2026-11-16, 2026.
https://sourcebae.com/blog/weak-supervision-data-annotation/

19. **Snorkel AI blog, Getting better performance from foundation models (with less data)**
retrieved 2026-11-16.
https://snorkel.ai/blog/getting-better-performance-from-foundation-models-with-less-data/

20. **Lacuna, LLMaAA: Making Large Language Models as Active Annotators**
retrieved 2026-11-16.
https://lacuna.tiptreesystems.com/work/llmaaa-making-large-language-models-as-active-annotators

21. **Rillor, Language models as annotators: agreement with people in published studies**
retrieved 2026-11-16.
https://rillor.com/insights/language-models-as-annotators

22. **Hands-On Tutorial: Labeling with LLM and Human-in-the-Loop, arXiv:2411.04637**
retrieved 2026-11-16.
https://arxiv.org/pdf/2411.04637

23. **AI Wiki, Data labeling**
retrieved 2026-11-16.
https://aiwiki.ai/wiki/data_labeling