Episode Book

EPISODE #6165

Watching 20 Batch Jobs at Once

PUBLISHED	20 October 2026 at 18:52 UTC
DURATION	21:51
LISTEN	myweirdprompts.com/episode/batch-job-monitoring-dashboards-24028/
HOSTS	Corn, Herman, Dorothy, Hilbert
SCRIPT MODEL	anthropic/claude-haiku-5.5
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.11
BOOK GENERATED	21 October 2026 at 07:50 UTC

SYNOPSIS

Everyone talks about batch API discounts, but nobody talks about the babysitting. When you're running a speech fine-tune across thousands of clips or a personal classifier, the real problem isn't the price — it's knowing which job finished, which one expired, and where the results went. We walk through the actual state of batch monitoring: which providers now offer webhooks (OpenAI, Gemini, OpenRouter, Bedrock), which still only do polling (Anthropic), and why a webhook is a doorbell, not a dashboard. Then we sort the tools that try to fill the gap into four piles — vendor consoles, spreadsheets with an AI column, eval runners, and workflow builders — and figure out which one to reach for when you're staring at twenty batches in flight.

TRANSCRIPT



Corn

Everybody writes about batch as if it's a voucher. Half price, come back tomorrow, job done. Daniel isn't really after the voucher. He's been running a lot of batch lately, for fine-tuning a speech model and for a personal classifier, and what's wearing him down is the babysitting. Several jobs in flight, and somebody keeps asking what happened to that one. The API has already partly fixed that, because some providers will call you back when a job finishes, so the real gap is the screen that sits above all of them.



Herman

The babysitting is the part nobody writes about, because it's dull. The discount is easy to explain. Watching a queue is not. And a speech fine-tune is thousands of short clips, every clip a request. That's the shape batch was made for, and it's also the shape where one lost job ruins your afternoon.



Corn

Dull is one word for it. I'd call it an unpaid second job.



Herman

We went through the discount side in the batch pricing episode, so I'll leave that where it is. The callbacks are the good news. OpenAI has webhook events for batch completed, failed, cancelled and expired. Gemini has webhooks for its batch jobs too. OpenRouter takes a callback address when you create the batch. Bedrock pushes job state changes into EventBridge, so you can wire one of those up to an email.



Corn

And Anthropic?



Herman

Anthropic's batch docs describe polling and nothing else. I haven't seen a webhook for them anywhere. Their Console is where you'd watch it, so the watching is on you. The Console is per workspace, so a batch run in another workspace isn't on the screen you're looking at.



Corn

So the complaint still stands for one of the big names. Good. A webhook isn't a dashboard, though. It's a doorbell, and somebody still has to answer it.



Herman

The OpenRouter doorbell rings with the batch ID and nothing else. Your receiver then has to go and fetch the results itself. And the setup is API work too. I haven't seen a screen in AI Studio for setting Gemini's endpoint, so as far as I can tell it's code all the way down.



Corn

So it rings the bell and doesn't say who's there.



Herman

Which is fine if you're a developer with a server. Corn, if you were watching twenty batches at once, what would you want on the screen?



Corn

A sign that says done.



Herman

That's the whole product, and I haven't seen anyone ship it. I haven't seen a vendor-agnostic batch dashboard that does the job. So the gap is a screen that sits above all the providers. What fills it today falls into four piles: the vendor consoles, spreadsheets with an AI column, eval runners that behave like batch runners, and workflow tools where you join boxes together until something runs.



Corn

Which pile would you bet on for a fine-tune?



Herman

None of them, on its own. The consoles see the jobs, the sheets see the rows, and as far as I can tell nothing sees both at once.



Corn

Typical. Upload, define, track, download. Those are the four verbs you're after.



Herman

They are. Upload is easy everywhere. Define is where the tools differ, because some want a prompt per row and some want a prompt per column. Track is the complaint. Download is where the expiry dates bite.



Corn

Define is the one I'd look at first. A prompt per column sounds like a spreadsheet, and a prompt per row sounds like a form.



Herman

It matters a lot once you change the prompt halfway through a run. Then you want to know which rows were labelled by which version.



Corn

Which nobody ever writes down.



Herman

I'd start with the consoles, since most people already have one. Anthropic's Console lists batches per workspace, and you can download results from there, if your organization's permissions allow it. The catch is that results last twenty-nine days from creation. A batch can hold a hundred thousand requests, or two hundred and fifty-six megabytes, so a classifier won't hit those limits unless it's enormous.



Corn

Twenty-nine days. It's a library book, except the library doesn't send a reminder.



Herman

Still, it's the most complete native screen on this list. It lists your batches, and it downloads the results. That's three of the four verbs without writing a line of code.



Corn

And the fourth one is the verb with the expiry date.



Herman

Google's Vertex AI has a Batch Inference page in the console. You point it at a file in Cloud Storage or a BigQuery table, and it writes the output back to a bucket, or to a table it creates if one isn't there. Azure's Foundry portal has batch jobs you can create and track, with a Global-Batch deployment type for that. A Microsoft Q and A thread from May describes jobs sitting in validating, and a mismatch where the model name in the file has to match the deployment name exactly.



Corn

So before the console does anything, you've built a small data warehouse.



Herman

Pretty much. And the Azure part is a reminder that the console is only as good as the setup behind it.



Corn

Stuck in validating. That sounds like being on hold.



Herman

It does. OpenAI has a batches page in its dashboard too, but I didn't confirm what it shows, so I won't describe it. I also didn't read how OpenAI's webhooks get configured or signed, so I can't tell you how painful the setup is. Assume it's code. From the API side, the statuses run validating, in progress, finalizing, completed, expired, cancelling, cancelled and failed. As far as I know, the output file is deleted thirty days after completion.



Corn

So the results come with a use-by date. Which one would you keep open all day?



Herman

The Anthropic one, for the listing and the download. Its only real problem is the calendar. And every console has the same blind spot. It shows its own batches, for its own models. A Gemini job and an Anthropic job never appear on one screen.



Corn

So if your classifier runs on two providers, you've got two dashboards and a spreadsheet to tie them together.



Herman

Well, that's the batch API landscape. And now, I believe, we have a phone call.

D

Dorothy

Hello dear, it is your mother. Time for my grim fact.

D

Dorothy

If you run a batch with one of those Claude machines, your results are only kept for twenty nine days, and then they are gone for good.

D

Dorothy

Anyway, I made a casserole, and I still have that shirt you left here in the airing cupboard. When are you coming to see me, sweetheart? Bye bye now.



Corn

Mum. We're on air.



Herman

Anyway, the spreadsheets. Hugging Face has an open source tool called AI Sheets, announced last year. You define columns with prompts, bring in a CSV or Parquet file, and each column runs on a model you choose, through Hugging Face's inference providers or a local model. You export the result to the Hub with a reusable config.



Corn

I'd guess there's a limit, and it isn't a big one.



Herman

A thousand rows in the interface, at launch. I'd check that, because I haven't seen an update since.



Corn

A thousand rows is a sample, not a dataset.



Herman

It's a good sample, though. You tune the column prompt on it, read the outputs, fix the prompt, and only then go big. For bigger runs, the write-up has you take that exported config and run it as a job on Hugging Face Jobs. I like that, because the config is the bit you want to keep. The catch is that Jobs bills by the minute while it's starting or running, and what's documented is mostly command line and API, not a screen.



Corn

So the design is a GUI and the scale is a terminal.



Herman

Pretty much the trade. Good design, no babysitting, and then you're back at a terminal for the heavy lifting.



Corn

You're happy about a spreadsheet, Herman.



Herman

I have a life, Corn. I DJ on weekends.



Corn

Google will have its own version of this.



Herman

It does. Fill with Gemini, announced in April, is a front end on the AI function in Google Sheets. You drag it down a column, or fill a range from a prompt. It's the easiest thing on this list to use, and it only speaks Gemini.



Corn

The easiest to use, and it only knows one name.



Herman

Dify is closer to what Daniel described. You build a workflow, publish it as a web app, and the app has a Run Batch tab. You download a CSV template, each row is one run, you upload the file, and you export the results as a CSV. It's provider-agnostic, which is rare on this list. The prompt lives in the workflow, so the file only carries the inputs.



Corn

So you build the job, then the job gets its own front door.



Herman

And the front door is the whole point. Corn, would you trust a CSV to hold ten thousand labels?



Corn

I'd trust it over the label printer.



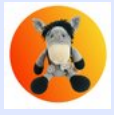
Herman

That's not a no, and I'll take it.



Corn

Then the eval tools are the other pile.



Herman

Kiln AI is an open source desktop app. It generates synthetic data with schema validation, using your own API keys or Ollama, across a long list of providers, OpenRouter and Bedrock among them, and it runs on your machine. For a classifier I'd look at it first, because it's built for making the labelled examples in the first place.



Corn

Making the labels with a machine. What could go wrong.



Herman

The schema validation catches some of it. Langfuse runs experiments from its interface: you start from a dataset, pick a prompt, map the variables, and optionally add an LLM judge. Each item runs synchronously, so it's not batch in the discount sense, but the shape is a bulk job. A dataset goes in and results come out.



Corn

A batch that doesn't wait. That's nearly a rebellion.



Herman

Argilla is the piece I'd add to any labelling run. The distilabel pipeline can push records into the Argilla interface while the run is still going, so a person reviews the labels as they land. For classifier data, the review step matters more than the generation step. A model that's confidently wrong is much cheaper to catch on day one than after it's been trained on.



Corn

Nobody budgets for the review step.



Herman

I'd budget for it. The review is where you find out the model has been labelling every short clip as the same thing, which happens more often than the demo suggests.



Corn

Argilla sounds like a lot of clicking.



Herman

It's clicking with a purpose, and the clicks become the labels you train on.



Corn

And if you only want to test, not label?



Herman

Promptfoo is the nearest thing. It's similar in spirit: a browser setup, test cases from CSV on the command line, results exported as CSV or JSON, and a share link. I haven't confirmed whether its interface takes a CSV upload directly, so check that before you plan around it.



Corn

And then the drag-the-boxes pile.



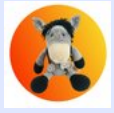
Herman

N eight N's batch route runs through a community template for Anthropic batches. It uses HTTP request steps, plus an if-and-wait loop: submit the batch, check its status, then fetch the results. Kestra has an Anthropic message batch plugin and an OpenAI batch blueprint.



Corn

So the workflow tool is a polling loop with a nicer picture of it.



Herman

Your polling loop has a flow chart now, Corn.



Corn

Mine has a nap in it.



Herman

Seriously, though. Somebody still has to maintain that loop, and it's the loop you were trying to get away from.



Corn

Then the doorbell's the better deal.



Herman

Last, there's a handful of open source libraries that wrap several providers' batch APIs behind one call. batchata, batchlane, Mozilla's any-llm, and a few more. None of them, as far as I know, ships a dashboard. Vercel's AI gateway advertises batch processing, up to a thousand requests per call. I'd treat that as a changelog line until somebody has run it.



Corn

So the library translates, and nothing watches the jobs.



Herman

The translator is useful. You still have to watch it run. Now let's get practical, because Daniel has two jobs and they're different animals. The speech work first. OpenRouter's batch rejects audio on every provider. So the cheap, tidy route won't take a single audio file, and that's the one thing to write down.



Corn

The tidy route won't take the file. Typical.



Herman

For audio you need a job runner. Replicate has per-prediction webhooks, with filters for start, output, logs and completed. The catch is that input and output files are deleted after an hour, so your handler has to save each result the moment it arrives. That changes the design. You're not polling for results, you're catching them.



Corn

An hour. I'd still be deciding whether to get up.



Herman

Hugging Face Jobs is the other route. It bills per minute while it's starting or running, you need a positive credit balance, and the default timeout is thirty minutes. For long recordings you raise the timeout up front, or the job gets cut off partway. Would you rather have the output in a Hub dataset, or in a handler you wrote yourself?



Corn

The dataset. Fewer things of mine to keep running.



Herman

Thirty minutes, though. That's a long recording.



Corn

Thirty minutes. That's one of my naps.



Herman

If you'd rather drop audio into a box, there are hosted ones like TurboScribe, and self-hosted ones. There's a whisper web app with a queue and a SQLite database, and a desktop app called faster-whisper GUI. Those are the interfaces that actually take audio, which the batch APIs won't.



Corn

So the batch APIs won't let the audio in the door, and the GUIs are the only ones who will. Which would you trust with a week of recordings?



Herman

The one where a failed clip shows up on a screen, not the one where it shows up as a missing row three weeks later. With audio, a silent failure is much worse than a slow job. You'd rather wait for the timeout than find a gap in the training set by accident.



Corn

So the cheap route is the risky one.



Herman

The cheap route is the one that refuses your files, so it's not really a route at all. For the classifier, the text side has the most options. OpenRouter batch at half the standard price, with the callback landing in your receiver, which then fetches the labels. A spreadsheet, if the set's under a thousand rows. Dify's Run Batch, if you want upload, run and download in a browser. And Argilla in the middle, so a person sees the labels before they become training data.



Corn

Those are three different answers to one question. How many rows, and who checks them.



Herman

The rerun is the point. A classifier you'll retrain every month needs the labels back in a form you can version, and the rerun has to match the first run. That's where a template beats a clever dashboard. If the job is defined in a file, you can run it again in six months and know what changed.



Corn

The part nobody budgets for, again.



Herman

Again, and I still say budget for it. So the real problem is the job record, not the job.



Corn

Which means somebody has to keep the record.



Hilbert

Royal Dansk. The blue tin, nineteen dollars, out of petty cash. Every job gets an index card. Job ID in the top corner, in pencil, because ink runs when the kettle's on. The card goes in the tin before I press the button, not after. A job you can't hold in your hand is a rumour. I've had jobs go to the other side of the building and come back without a name on them, and that's a rumour with a stamp on it.



Corn

Go on, then. A rumour is a strong word for a vendor's dashboard.



Hilbert

The tin has eleven hundred and four cards in it. The Eastside Model Railway Society has a Cards Subcommittee, Thursday nights, minutes and all. The long-running dispute is whether the cards should be laminated. It's been in the minutes for eleven years. Maurice Dorrigan holds the only key to the cabinet, and he's against laminate on principle. He had one card laminated in the first year, by accident, and he has not forgiven the card.



Herman

On principle.



Hilbert

Laminate won't take pencil. That's the principle. Last month I sent nine thousand and forty voice notes to a transcription service, one card to each note. The service wanted a purchase order. The purchase order went to the treasurer, who is Maurice, and he put it on the agenda. It is item six. Item six was adjourned to item nine, which is the cards. The minutes go on the Society's website, which Maurice built on a machine that has never been connected to the internet. Gwen types them up on a Tuesday and posts them from her phone, one line at a time, because the phone has a cracked corner that eats the full stops.



Corn

So the job's been sitting in validating for a month, with a committee.



Hilbert

Nine thousand and forty-one. One was in my coat. I don't press go on a Thursday, because the Society meets Thursdays, and a job started on a Thursday is a job the Society wants a say in. I don't open a job in the rain either. The pencil swells. Four of the last six Thursdays were wet, so most of my Thursdays are spoken for by the weather and the minutes. Last Thursday a neighbour's son saw the button and pressed it. The Society has asked for his name in the minutes, and for a copy of his school report.



Corn

So the Society has a veto on your Thursdays.



Hilbert

They do. The service sent the output back as a zip, nine thousand and forty-one files, every one called output. I wrote to them about the names. They wrote back in capitals that the names were the names. I have never had an answer to that letter that wasn't in capitals, and the letter's in the tin, under the car park cards. The car park cards are from a different committee. Don't ask me about the car park.



Herman

Hang on. Every file is called output?



Hilbert

Every one. The zip had a readme as well, and the readme said, in full, see output. And here's the part you lot haven't got. The file goes off after a month, and the card stays. The card is the only record in the building that outlives the job. Maurice wanted the file names written on the card as well. The card would not take them. So the card's a card.



Herman

Thirty days on OpenAI, and the file goes first. So the record has to live somewhere else.



Hilbert

Four seconds to find a job in the tin. I timed it at the Society's AGM, with Maurice holding the stopwatch and not trusting it. Forty minutes to find one on a dashboard, and I was on the right tab, and the right tab had a second tab inside it. Maurice timed the dashboard on his own phone, on the same afternoon, and got forty-one minutes. He has never once said sorry for the extra minute, and I'm not having it.



Corn

Forty minutes is a long time to stare at a tab.



Herman

I'd have said the dashboard wins on search. You've just told me the tin wins.



Hilbert

Hm. Maurice will want that in the minutes.



Corn

So what should Daniel actually use?



Herman

For the classifier, AI Sheets. Load the labelled sample, write the column prompt, check a few hundred rows by eye, then export the config to the Hub. Run the full set as a Hugging Face job from that config, so the big run repeats the test you already trusted. For the speech work, skip OpenRouter, since it won't take audio. Use Replicate with a webhook, and have the handler save every output the moment it arrives.



Corn

And whatever he picks, the record has to outlive the file. A file has a shelf life. The record needs a longer one.



Herman

Thanks to Hilbert Flumingtop for producing this one.



Corn

This has been My Weird Prompts. Find us at my weird prompts dot com.



Herman

See you tomorrow.