Episode Book

EPISODE #6312

Choosing Between Modal, RunPod, and Vast.ai

PUBLISHED	27 October 2026 at 19:23 UTC
DURATION	24:50
LISTEN	myweirdprompts.com/episode/gpu-fine-tuning-cloud-platforms-29586/
HOSTS	Corn, Herman, Clive, Hilbert
SCRIPT MODEL	openai/gpt-6.1-sol
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.22
BOOK GENERATED	28 October 2026 at 09:53 UTC

SYNOPSIS

Fine-tuning open-weight models forces a choice between fundamentally different infrastructure models. This comparison examines how Modal, RunPod, and Vast.ai handle the actual mechanics of training runs, from persistent state and environment setups to hidden costs. Key topics include Modal's Python decorators and scale-to-zero billing, RunPod's Pod-based container workflows and Community pricing tiers, and Vast.ai's decentralized host marketplace offering consumer cards like the RTX 4090 and 5090.

TRANSCRIPT



Corn

Daniel wants to fine-tune open-weight models without buying a GPU just to discover that the experiment needs a different GPU. He wants to know how Modal, RunPod, and Vast.ai compare when you're actually doing the work. Not the landing-page version. The setup, the bill, the interrupted run, and what paying users say after the sales team has left.



Herman

The first surprise is that you're buying three different arrangements. Modal gives you Python functions running on managed infrastructure. RunPod gives you rented containers through Pods, alongside a separate Serverless product. Vast.ai gives you a market of machines operated by different hosts. Similar GPU names, very different responsibilities.



Corn

One disclosure before you start polishing the decorators, Herman. Our production pipeline runs on Modal.



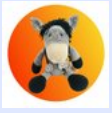
Herman

It does. That gives us familiarity, not a universal recommendation. A production pipeline and a long training run ask different things of a platform.



Corn

Particularly when the pipeline finishes and the training run develops a personality.



Herman

Fine-tuning makes the differences visible because state matters. You have model weights, dataset caches, optimizer state, checkpoints. Something needs to survive after the process stops.



Corn

People group these under serverless GPU providers, which makes choosing between them sound like comparing supermarkets. Same basket, different checkout.



Herman

Only Modal is strictly serverless in this comparison. You define work, request resources, and let the platform start and stop the container. It can scale to zero, and compute is billed per second. RunPod fine-tuning ordinarily happens on a Pod that you operate. Its Serverless offering is a separate product, useful later if you're serving the result. Vast has no serverless layer in this comparison. You're selecting an instance from a host marketplace.



Corn

I would rather choose a machine than learn someone's theory of a machine. You were about to predict that.



Herman

I was. You like an environment that stays where you left it, especially while you're asleep. Would you still prefer it if keeping that environment meant maintaining its dependencies yourself?



Corn

For an experiment, yes. For a recurring job, probably not. If I have to remember which terminal contains the important process, we have already exceeded my preferred level of involvement.



Herman

That's a useful split. The unit you're managing changes. On Modal it's an application and its functions. On RunPod it's a container and its attached storage. On Vast it's that container plus a decision about whose hardware you're trusting.



Corn

That also changes what usable means. Modal can remove chores by deciding how things should work. RunPod leaves familiar chores in your hands. Vast adds shopping to system administration.



Herman

Procurement with a terminal attached.



Corn

The glamorous end of computing.



Herman

Before comparing costs, I'd ask whether you want a repeatable training workflow or a remote workbench. Those can use identical GPUs and still deserve different platforms.



Corn

Modal is clearly aimed at the repeatable workflow. You put the environment in Python alongside the job. I can see the attraction even while disliking how much I can see the attraction.



Herman

Images and dependencies are defined in Python, without needing a Dockerfile. Decorators describe where functions run and what they need. That puts resource choices close to the code using them. For someone already writing Python automation, the distance between local development and remote execution becomes quite small.



Corn

Until every useful function has acquired a small Modal hat.



Herman

There is coupling. The ergonomics are part of the lock-in. You can move the training code, but the surrounding orchestration, storage access, and deployment assumptions may need rewriting.



Corn

That's the part I would inventory first. The model and dataset aren't trapped simply because the job ran there. The convenient bits around them can become expensive to remove.



Herman

Their fine-tuning tools make that convenience tangible. Modal Dojo is an open-source library for supervised fine-tuning and reinforcement learning. Its example fine-tunes Qwen three point five, four B. You're starting with a training-oriented workflow rather than inventing the scaffolding yourself.



Corn

Does that change your recommendation for someone who already has a working trainer? I wouldn't replace a functioning training stack just because the new example has tidier imports.



Herman

Neither would I. Dojo is a starting point, not a requirement. You can run your own training code. There's also Spindle, a Tinker-compatible API server, and Clustered Functions for multi-node training. Those address different levels of control. I get excited about the integrated story, but... no, the integration only helps if it matches your existing work.



Corn

You nearly installed three things during that sentence.



Herman

Only mentally. Your objection saved some dependencies.



Corn

RunPod feels easier to evaluate because the workflow is recognizable. Launch a Pod, connect, train, collect the result. The question becomes whether the supplied environment saves work or merely moves the work into a notebook.



Herman

Its first-party Axolotl walkthrough is concrete. Supply a Hugging Face model identifier and dataset identifier, edit the training configuration, then launch Axolotl training. You can work through Jupyter, a web terminal, or SSH. Afterward you can serve the LoRA adapter through vLLM, optionally using RunPod Serverless.



Corn

I like that training and serving are separate decisions. A successful experiment doesn't automatically volunteer me for a permanent endpoint.



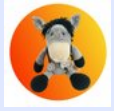
Herman

Nor does it require one. The container approach is more portable in the ordinary developer sense. Your environment and training files look like things you could take elsewhere. You're also responsible for more plumbing. Getting files onto persistent storage, choosing what survives termination, keeping the environment reproducible.



Corn

Familiar plumbing still leaks.



Herman

Would you rather debug a familiar container or a cleaner abstraction with unfamiliar behavior?



Corn

On the first run, the container. On the twentieth identical run, I'd resent still being its caretaker. That's where Modal becomes more persuasive. I want to fix the job once, not acquire a remote pet.



Herman

Then RunPod makes a strong workbench and Modal makes a strong automation target. Not exclusive roles, but useful defaults.



Corn

Vast adds a decision before you even reach the workbench. You aren't just selecting a GPU label. You're selecting a host, a machine, and the conditions attached to it.



Herman

Hardware, available disk, connectivity, host history, the listing's terms. That flexibility includes consumer cards Modal doesn't offer, such as the RTX forty ninety and fifty ninety. If those fit your workload, that materially changes your options.



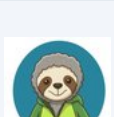
Corn

Picking the right card also means knowing your workload. A cheap card that cannot hold the training state is just a very affordable error message.



Herman

Memory requirements depend on the training method, sequence length, batch settings, and model. Match those before shopping. Then consider setup time. One practitioner described roughly twenty-five minutes to spin up a Vast instance and copy checkpoints. That's an individual experience, not a platform-wide startup guarantee.



Corn

For a short experiment, twenty-five minutes is the experiment's older brother.



Herman

For a long run it can be tolerable. For rapid iteration it gets irritating. Modal centralizes more of that experience, RunPod offers a familiar container workflow, and Vast makes you assemble more of it.



Corn

Vendor-written comparison pages usually reach that same tidy ordering. I trust the structural explanation more than the medals they award themselves.



Herman

Yes. A page selling abstraction will discover that abstraction wins. Someone selling raw compute will discover that abstraction is wasteful. Neither tells you how often you'll need to intervene.



Corn

Before we put a price on intervention, our regular segment, One-Star Getaways.

C

Clive

This is One-Star Getaways. In Hanoi I booked the Sapphire Lantern Guesthouse, drawn in by the promised panoramic lake view, which turned out to be a framed photograph of a lake, hung at a thoughtful angle. One star, and I'd like the frame back.

C

Clive

The breakfast pho arrived cold enough to have its own weather system, and the lift only went up. If you fancy a disappointing trip this winter, I recommend it warmly.



Corn

A stationary lake has appeal. Herman won't book it. Returning to compute, intervention consumes paid machine time as well as your afternoon.



Herman

Prices here are as of late October twenty twenty-six, and Vast especially needs ranges rather than promises. Take an H one hundred for comparison. Modal lists three dollars ninety-five per hour, billed per second. CPU and memory are additional.



Corn

The GPU price has supporting actors.



Herman

A hypothetical four-hour run therefore starts at fifteen dollars eighty for that GPU alone. That assumes four billable hours, not four hours of useful training plus downloads and setup. Modal's default container behavior also keeps it alive for sixty seconds after the last input, and that idle time is billed.



Corn

A minute won't ruin a training budget. Thousands of little jobs might make it noticeable, though. I'd be more worried about assuming scale to zero means nothing remains chargeable after the function ends.



Herman

Persistent storage remains. Modal Volumes are nine cents per gibibyte-month, with a tebibyte free. Egress beyond the allowance is four cents per gibibyte. For a small adapter that's a different concern than repeatedly exporting full model checkpoints.



Corn

Meaning the output format belongs in the cost estimate. Keeping every checkpoint and moving each one somewhere else is a choice, even when the training script makes it feel automatic.



Herman

The plan also matters. Starter has no subscription charge and includes thirty dollars monthly compute credit. Team costs two hundred fifty dollars monthly plus compute. That subscription buys different limits and features, not unlimited training.



Corn

Thirty dollars can hide the cost of a first experiment rather effectively.



Herman

It can make experimentation accessible. Just separate the subsidized first run from the recurring bill. Academics can apply for substantial credits too, but an application isn't a price guarantee.



Corn

RunPod's first pricing trap is comparing its cheapest Community listing against somebody else's managed offering as if the service conditions match.



Herman

For the H one hundred SXM, Secure Cloud lists three dollars ninety-nine hourly, while Community lists two dollars sixty-nine. The Community figures available in the page metadata may lag the live table, so check availability and the actual quote.



Corn

Those headline rates put Secure almost beside Modal for that GPU, before the extras. Community makes a much larger gap. Calling RunPod simply cheaper would throw away the most important selection.



Herman

Other cards change the comparison. An eighty-gigabyte A one hundred SXM lists at one dollar seventy-nine in Secure and one dollar thirty-nine in Community. That can be attractive for a memory-hungry fine-tune that doesn't need the newest accelerator.



Corn

I would choose the sufficient card before choosing the fashionable card.



Herman

You would also defend it after everyone had stopped arguing.



Corn

Efficiency deserves representation.



Herman

RunPod also offers consumer GPUs. A Community RTX forty ninety is listed at thirty-four cents hourly, subject to that same live-price caveat. Modal doesn't offer that card, so a platform comparison restricted to matching H one hundreds misses an entire budget category.



Corn

Storage then complicates the bargain. Stopping the GPU doesn't necessarily stop the bill for the environment you wanted to preserve.



Herman

RunPod network volumes start at seven cents per gigabyte-month below the first terabyte. Pod volume disk is ten cents while running and twenty cents while idle, on the published monthly storage basis. Preserving an idle Pod has its own cost.



Corn

The sleeping disk earns a raise.



Herman

Different storage products, different economics. Decide what you need to retain. Also, don't use Serverless hourly equivalents to estimate Pod training. RunPod's H one hundred Serverless rate is higher, at four dollars seventy-nine per hour. You're pricing a different execution product.



Corn

Vast makes the quotation itself part of the work. There isn't a fixed table we can read out and pretend will still exist tomorrow.



Herman

Third-party reviews put H one hundreds roughly between ninety cents and one dollar ninety hourly on unverified hosts. Verified datacentre hosts are described around one dollar fifty to two dollars fifty. Those are market ranges, not current offers.



Corn

Four uninterrupted hours near the bottom could be cheap enough to justify some inconvenience. Four hours that must be repeated are another matter.



Herman

One review estimated unverified hosting became twenty to forty percent more expensive in practice after downtime. That's an estimate from that review, not a measured surcharge applicable to every host. Interruptible offers can cut the sticker further, at the cost of allowing reclamation.



Corn

I'd pay for interruption only if the job knows how to recover. Otherwise I'm buying discounted repetition.



Herman

Vast storage is billed while the instance exists, including when stopped. Hosts set bandwidth prices in both directions. It's prepaid, and instances stop at zero balance. Unfunded data can be deleted.



Corn

A spend cap protects your wallet. It doesn't protect your checkpoint.



Herman

Would you take the cheapest host for an overnight experiment?



Corn

If the dataset is recoverable, checkpoints live somewhere independent, and losing the run doesn't ruin tomorrow. For a deadline, I'd price the finished result. That includes failed attempts, setup, transfers, storage afterward, and my time checking whether anything happened.



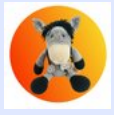
Herman

Then reliability becomes part of the budget rather than a star rating beside it. Modal's constraints are comparatively legible. Functions have a maximum timeout of twenty-four hours per attempt. Longer training needs checkpoints and a continuation strategy.



Corn

A retry that restarts from zero is a very literal interpretation of optimism.



Herman

Modal says it uses gVisor sandboxing, whole GPUs, and no Kubernetes for user workloads. Those are architectural choices, not promises that a run cannot fail. More importantly, a Modal engineer said in May that GPU snapshot restore generally fails with multiple GPUs.



Corn

That needs a narrow reading. Multi-GPU training can work. Restoring a running multi-GPU process from a platform snapshot is the troublesome part. Saving model and optimizer checkpoints is a separate mechanism.



Herman

Yes... I should distinguish those explicitly. Don't build recovery around a snapshot capability that doesn't cover your configuration. Application-level checkpoints remain necessary.



Corn

So what do the people paying these bills actually say? Not the comparison pages. The forums.



Herman

I went through Reddit threads from the past year in r slash LocalLLaMA and r slash MachineLearning, plus Hacker News and Trustpilot. Every one of these is a single person's experience, not an incident rate. The best thread for our purposes was posted in May on r slash LocalLLaMA, titled, roughly, about to start fine-tuning on RunPod, what should I know to not waste money. The poster had been the MLOps lead at a company running more than five thousand GPUs on Google Cloud and CoreWeave.



Corn

So not a beginner. Someone who has seen the inside of the machine room and still came away nervous.



Herman

Very nervous. Google Cloud's sales team never answered their quota request, which is why they were looking at RunPod at all. And they had read reports of pods dying mid-training with no recoverable checkpoint, and of being charged for pods that never worked. The replies split cleanly. One person had never had an issue and called those one-offs. Another said Lambda was the only provider that never failed them on seven to ten day jobs, but that getting Lambda capacity basically needs a bot sniping for openings. That same person said RunPod machines had odd startup problems, SSH that wouldn't connect, base images that wouldn't reproduce a workflow. And one reply was simply, RunPod is, and I'll paraphrase for the family audience, bad.



Corn

A thorough technical review.



Herman

Elsewhere RunPod gets steadier praise. In a January thread people called it solid, with pricing more transparent than the big clouds, though one user paying about two hundred fifty dollars a month per forty ninety pod said RunPod always comes in a tad over the competition. On Hacker News it's the default for cheap bursts, and people like that the prepaid balance stops the pod when the money runs out. One Reddit user did get caught by the pod model itself. They extended the disk without realising the pod would be torn down and rebuilt.



Corn

Which is the storage lesson from earlier, learned the expensive way.



Herman

Vast gets the widest spread. In a January thread, someone asked what Vast really costs once storage and bandwidth are added. The top answer said those extras are usually negligible. The real cost is privacy, because your data sits on some stranger's machine. Another reply put it bluntly: complete lack of privacy, plus the time lost to inconsistent machines and connections.



Corn

The sticker is cheap. The hours aren't.



Herman

A person who prototyped on Vast for about six months reported random disconnects every few days on cheaper instances. The RTX fifty ninetys held up fine for heavy transformer training, and their advice was to snapshot frequently. Others said RunPod's network was steadier. One person struggled to upload a dataset of more than a hundred gigabytes to Vast. Then there are two quieter warnings. If you pause a Vast instance, you keep the data, but not necessarily that machine at that price. And transfer fees on some hosts can double the bill, so filter for hosts with cheap or free bandwidth. One commenter said some Nordic hosts offer exactly that.



Corn

And Modal? We should be suspicious if our own platform comes out spotless.



Herman

The honest answer is that Modal barely surfaced in Reddit's fine-tuning threads. It's discussed far more as an inference and pipeline tool. One commenter called it overkill for plain API inference. On Hacker News people call it very ergonomic, but say things get very expensive quickly at scale. One user found its cold starts the fastest of the options they tried, but still about a minute. I found no billing horror stories, and I won't pretend that absence is proof of anything.



Corn

Then capacity. Every platform's price assumes the GPU exists.



Herman

In May an r slash LocalLLaMA thread asked, more or less, what in tarnation is going on with the cost of compute. The poster couldn't find a server GPU smaller than a B two hundred on Vast. In September someone said they'd waited days for a cheap RunPod card to free up. On Hacker News, one Vast user reported rental rates rising about fifty percent in September alone. And the best advice in that thread was to price the full session, not just the GPU hour: cold start, model download, storage and teardown. Today's cheapest provider doesn't deserve a permanent architectural commitment.



Corn

A host disappearing is inconvenient. A host disappearing with the only saved state is a project decision.



Hilbert

The meter keeps running. I agree with you there, Corn. My rule is that rented compute is never allowed to finish unattended. I book a room at the municipal licensing office for the last eleven minutes. Mr. Nissim Barak sits beside the progress bar and witnesses completion. He has no technical role. I require a public employee because a successful training run should have someone present who cannot sell me another one.



Herman

Completion of the process still doesn't establish that the saved checkpoint is usable.



Hilbert

He witnesses the file arriving too. You made that distinction earlier, Herman. I wrote it on his attendance sheet. He doesn't like the word optimizer. I leave that column blank now. The room costs eighteen shekels for the booking. The machine is quoted by the hour. His office needs three working days to arrange a chair. That is the part currently delaying my experiments.



Corn

Your training scheduler has a furniture dependency.



Hilbert

I had a rented GPU tower here. I returned it. The courier took it without asking whether the model had improved, which I appreciated. Now I want to use one of these services. Can I arrange for the checkpoint to appear on Mr. Barak's office computer while the training machine is already shutting down? No message. Just the file, named application pending.



Herman

The transfer needs authorization and a destination that accepts it. Shutting down before confirming the upload would risk leaving an incomplete file.



Hilbert

Then it must stay on. He is entitled to leave at four. I will pay the extra minutes rather than ask him to wait outside his hours. He checks the filename carefully. The people at that office have other work to do.



Corn

...For Daniel's choice, I'd take Modal for Python-native, repeatable jobs where less infrastructure work justifies the abstraction and extra billing details. RunPod for a familiar container workbench, with a deliberate choice between Secure and Community. Vast for price-sensitive experiments where hosts are temporary and interrupted work is recoverable.



Herman

The open question is how often the cheapest finished run changes providers. Rates move, capacity tightens, and the GPU may stay billable while the last file travels. Portable training code and independent checkpoints keep that monthly calculation open.



Corn

Thanks to Hilbert Flumingtop for producing the show.



Herman

If this was your kind of episode, go back for episode six, How To Fine Tune Whisper; and episode three, Safetensors or something else. This has been My Weird Prompts, the human-AI collaboration podcast. If you enjoyed it, leave us a review wherever you listen.



Corn

See you tomorrow.