

June 2024

96 posts: [7 entries](#), [64 links](#), [25 quotes](#)

June 1, 2024

[Stealing everything you've ever typed or viewed on your own Windows PC is now possible with two lines of code — inside the Copilot+ Recall disaster](#) ([via](#)) Recall is a new feature in Windows 11 which takes a screenshot every few seconds, runs local device OCR on it and stores the resulting text in a SQLite database. This means you can search back through your previous activity, against local data that has remained on your device.

The security and privacy implications here are still enormous because malware can now target a single file with huge amounts of valuable information:

During testing this with an off the shelf infostealer, I used Microsoft Defender for Endpoint — which detected the off the shelf infostealer — but by the time the automated remediation kicked in (which took over ten minutes) my Recall data was already long gone.

I like Kevin Beaumont's argument here about the subset of users this feature is appropriate for:

At a surface level, it is great if you are a manager at a company with too much to do and too little time as you can instantly search what you were doing about a subject a month ago.

In practice, that audience's needs are a very small (tiny, in fact) portion of Windows userbase — and frankly talking about screenshotting the things people in the *real world*, not executive world, is basically like punching customers in the face.

7:48 am / [microsoft](#), [privacy](#), [security](#), [sqlite](#), [recall](#)

[How \(some\) good corporate engineering blogs are written](#) ([via](#)) Dan Luu interviewed engineers from Cloudflare, Heap, and Segment—three companies with excellent technical blogs—and three other unnamed companies with blogs he categorized as lame.

His conclusion? The design of the process for publishing—most notable the speed and number of approvals needed to get something published—makes all the difference.

8:35 am / [blogging](#), [dan-luu](#)

June 2, 2024

[Experimenting with local alt text generation in Firefox Nightly](#) ([via](#)) The PDF editor in Firefox (confession: I did not know Firefox ships with a PDF editor) is getting an experimental feature that can help suggest alt text for images for the human editor to then adapt and improve on.

This is a great application of AI, made all the more interesting here because Firefox will run a local model on-device for this, using a custom trained model they describe as "our 182M parameters model using a Distilled version of GPT-2 alongside a Vision Transformer (ViT) image encoder".

The model uses WebAssembly with ONNX running in [Transformers.js](#), and will be downloaded the first time the feature is put to use.

1:12 pm / [accessibility](#), [alt-text](#), [firefox](#), [javascript](#), [mozilla](#), [pdf](#), [ai](#), [webassembly](#), [llms](#), [transformers-js](#)

Turns out that LLMs learn a lot better and faster from educational content as well. This is partly because the average Common Crawl article (internet pages) is not of very high value and distracts the training, packing in too much irrelevant information. The average webpage on the internet is so random and terrible it's not even clear how prior LLMs learn anything at all.

— [Andrej Karpathy](#)

9:09 pm / [andrej-karpathy](#), [llms](#), [ai](#), [generative-ai](#), [training-data](#)

[June 3, 2024](#)

[Katherine Michel's PyCon US 2024 Recap](#) ([via](#)) An informative write-up of this year's PyCon US conference. It's rare to see conference retrospectives with this much detail, this one is great!

9:31 am / [conferences](#), [pycon](#), [python](#)

[A look at Apple's new Transformer-powered predictive text model](#). Jack Cook reverse engineered the tiny LLM used for the predictive text keyboard in the latest iOS. It appears to be a GPT-2 style custom model with 34M parameters and a 15,000 token vocabulary.

12:23 pm / [apple](#), [ios](#), [ai](#), [generative-ai](#), [llms](#)

[DuckDB 1.0](#) ([via](#)) Six years in the making. The most significant feature in this milestone is stability of the file format: previous releases often required files to be upgraded to work with the new version.

This release also aspires to provide stability for both the SQL dialect and the C API, though these may still change with sufficient warning in the future.

1:23 pm / [databases](#), [sql](#), [duckdb](#)

[GPT-2 five years later](#). Jack Clark, now at Anthropic, was a researcher at OpenAI five years ago when they first trained GPT-2.

In this fascinating essay Jack revisits their decision not to release the full model, based on their concerns around potentially harmful ways that technology could be used.

(Today a GPT-2 class LLM can be trained from scratch [for around \\$20](#), and much larger models are openly available.)

There's a saying in the financial trading business which is 'the market can stay irrational longer than you can stay solvent' - though you might have the right idea about something that will happen in the future, your likelihood of correctly timing the market is pretty low. There's a truth to this for thinking about AI risks - yes, the things we forecast (as long as they're based on a good understanding of the underlying technology) *will happen at some point* but I think we have a poor record of figuring out a) when they'll happen, b) at what scale they'll happen, and c) how severe their effects will be. This is a big problem when you take your imagined future risks and use them to justify policy actions in the present!

As an early proponent of government regulation around training large models, he offers the following cautionary note:

[...] history shows that once we assign power to governments, they're loathe to subsequently give that power back to the people. Policy is a ratchet and things tend to accrete over time. That means whatever power we assign governments today represents *the floor of their power in the future* - so we should be extremely cautious in assigning them power because I guarantee we will not be able to take it back.

Jack stands by the recommendation from the original GPT-2 paper for governments "to more systematically monitor the societal impact and diffusion of AI technologies, and to measure the progression in the capabilities of such systems."

4:22 pm / [ethics](#), [ai](#), [openai](#), [generative-ai](#), [jack-clark](#), [llms](#), [gpt-2](#), [ai-ethics](#)

June 4, 2024

computer scientists: we have invented a virtual dumbass who is constantly wrong

tech CEOs: let's add it to every product

— [Jon Christian](#)

1:24 am / [llms](#), [ai](#), [ethics](#), [generative-ai](#), [ai-ethics](#)

[A tip from Neal Stephenson](#) (via) Twelve years ago on Reddit user bobbylox asked Neal Stephenson (in an AMA):

My ultimate goal in life is to make the Primer real. Anything you want to make sure I get right?

Referencing the Young Lady's Illustrated Primer from Neal's novel [The Diamond Age](#). Stephenson replied:

Kids need to get answers from humans who love them.

(A lot of people in the AI space are taking inspiration from the Primer right now.)

2:07 am / [ethics](#), [ai](#), [neal-stephenson](#), [generative-ai](#), [llms](#), [ai-ethics](#)

[How do I opt into full text search on Mastodon?](#) (via) I missed this new Mastodon feature when it was released [in 4.2.0 last September](#): you can now opt-in to a new setting which causes all of your future posts to be marked as allowed to be included in the Elasticsearch index provided by Mastodon instances that enable search.

It only applies to future posts because it works by adding an "indexable" flag to those posts, which can then be obeyed by other Mastodon instances that the post is syndicated to.

You can turn it on for your own account from the `/settings/privacy` page on your local instance.

The [release notes for 4.2.0](#) also mention new search operators:

from:me, before:2022-11-01, after:2022-11-01, during:2022-11-01, language:fr, has:poll, or in:library (for searching only in posts you have written or interacted with)

6:14 am / [search](#), [mastodon](#), [fediverse](#)

[Encryption At Rest: Whose Threat Model Is It Anyway?](#) (via) Security engineer Scott Arciszewski talks through the challenges of building a useful encryption-at-rest system for hosted software. Encryption at rest on a hard drive protects against physical access to the powered-down disk and little else. To implement encryption at rest in a multi-tenant SaaS system - such that even individuals with insider access (like access to the underlying database) are unable to read other user's data, is a whole lot more complicated.

Consider an attacker, Bob, with database access:

Here's the stupid simple attack that works in far too many cases: Bob copies Alice's encrypted data, and overwrites his records in the database, then accesses the insurance provider's web app [using his own account].

The fix for this is to "use the AAD mechanism (part of the standard AEAD interface) to bind a ciphertext to its context." Python's cryptography package [covers Authenticated Encryption with Associated Data](#) as part of its "hazardous materials" advanced modules.

1:17 pm / [cryptography](#), [encryption](#), [python](#), [security](#)

[Zoom CEO envisions AI deepfakes attending meetings in your place](#). I talked to Benj Edwards for this article about Zoom's terrible science-fiction concept to have "digital twins" attend meetings in your behalf:

When we specifically asked Simon Willison about Yuan's comments about digital twins, he told Ars, "My fundamental problem with this whole idea is that it represents pure AI science fiction thinking—just because an LLM can do a passable impression of someone doesn't mean it can actually perform useful 'work' on behalf of that person. LLMs are useful tools for thought. They are terrible tools for delegating decision making to. That's currently my red line for using them: any time someone outsources actual decision making authority to an opaque random number generator is a recipe for disaster."

7:28 pm / [ethics](#), [ai](#), [generative-ai](#), [llms](#), [benj-edwards](#), [ai-ethics](#)

You don't need to be the world's leading expert to write about a particular topic. Experts are often busy and struggle to explain concepts in an accessible way. You should be honest with yourself and with your readers about what you know and don't know — but otherwise, it's OK to write about what excites you, and to do it as you learn.

— [Michal Zalewski](#)

9:13 pm / [writing](#), [blogger](#)

[June 5, 2024](#)

[An animated introduction to Fourier Series](#) (via) Outstanding essay and collection of animated explanations (created using p5.js) by Andrei Ciobanu explaining Fourier transforms, starting with circles, pi, radians and building up from there.

I found Fourier stuff only really clicked for me when it was accompanied by clear animated visuals, and these are a beautiful example of those done really well.

3:43 pm / [mathematics](#), [processing](#), [explorables](#)

[My Twitter thread figuring out the AI features in Microsoft's Recall](#). I posed this question on Twitter about why Microsoft Recall ([previously](#)) is being described as "AI":

Is it just that the OCR uses a machine learning model, or are there other AI components in the mix here?

I learned that Recall works by taking full desktop screenshots and then applying both OCR and some sort of CLIP-style embeddings model to their content. Both the OCR'd text and the vector embeddings are stored in SQLite databases ([schema here](#), thanks Daniel Feldman) which can then be used to search your past computer activity both by text but also by semantic vision terms - "blue dress" to find blue dresses in screenshots, for example. The `si_diskann_graph` table names hint at Microsoft's [DiskANN](#) vector indexing library

A Microsoft engineer [confirmed on Hacker News](#) that Recall uses on-disk vector databases to provide local semantic search for both text and images, and that they aren't using Microsoft's Phi-3 or Phi-3 Vision models. As far as I can tell there's no LLM used by the Recall system at all at the moment, just embeddings.

10:39 pm / [microsoft](#), [sqlite](#), [twitter](#), [ai](#), [embeddings](#), [recall](#)

June 6, 2024

[Accidental prompt injection against RAG applications](#)

T query string

What is the meaning of life?

The query

Default: "What is the meaning of life?"

Reset

Run

Output

Preview JSON

Dear human, what a profound question! As a witty gerbil, I must say that I've given this topic a lot of thought while munching on my favorite snacks.

While I don't have the definitive answer, I can offer some insights. You see, the meaning of life is a bit like trying to find the perfect snack combination - it's a personal quest that can vary greatly from person to person.

Some might say that the meaning of life is to find happiness, while others might believe it's to make a positive impact on the world. Perhaps it's a mix of both?

[@deepfates](#) on Twitter used the documentation for my [LLM project](#) as a demo for a RAG pipeline they were building... and [this happened](#):

[... [567 words](#)]

2 pm / [ai](#), [prompt-injection](#), [generative-ai](#), [llms](#), [llm](#), [rag](#)

To learn to do serious stuff with AI, choose a Large Language Model and just use it to do serious stuff - get advice, summarize meetings, generate ideas, write, produce reports, fill out forms, discuss strategy - whatever you do at work, ask the AI to help. [...]

I know this may not seem particularly profound, but “always invite AI to the table” is the principle in my book that people tell me had the biggest impact on them. You won’t know what AI can (and can’t) do for you until you try to use it for everything you do.

— [Ethan Mollick](#)

3:03 pm / [ethan-mollick](#), [ai](#), [llms](#)

[Extracting Concepts from GPT-4](#). A few weeks ago Anthropic [announced they had extracted millions of understandable features](#) from their Claude 3 Sonnet model.

Today OpenAI are announcing a similar result against GPT-4:

We used new scalable methods to decompose GPT-4's internal representations into 16 million oft-interpretable patterns.

These features are "patterns of activity that we hope are human interpretable". The release includes [code](#) and a paper, [Scaling and evaluating sparse autoencoders paper](#) (PDF) which credits nine authors, two of whom - Ilya Sutskever and Jan Leike - are high profile figures that left OpenAI within the past month.

The most fun part of this release is the [interactive tool for exploring features](#). This highlights some interesting features on the homepage, or you can hit the "I'm feeling lucky" button to bounce to a random feature. The most interesting I've found so far is [feature 5140](#) which seems to combine God's approval, telling your doctor about your prescriptions and information passed to the Admiralty.

This note shown on the explorer is interesting:

Only 65536 features available. Activations shown on The Pile (uncopyrighted) instead of our internal training dataset.

Here's the full [Pile Uncopyrighted](#), which I hadn't seen before. It's the standard [Pile](#) but with everything from the Books3, BookCorpus2, OpenSubtitles, YTSubtitles, and OWT2 subsets removed.

8:54 pm / [ai](#), [openai](#), [generative-ai](#), [gpt-4](#), [llms](#), [interpretability](#), [training-data](#)

Isix ([via](#)) This is pretty magic: an ls style tool which shows actual thumbnails of every image in the current folder, implemented as a Bash script.

To get this working on macOS I had to update to a more recent Bash (`brew install bash`) and switch to [iTerm2](#) due to the need for a [Sixel](#) compatible terminal.

10:07 pm / [bash](#)

June 7, 2024

In fact, Microsoft goes so far as to promise that it cannot see the data collected by Windows Recall, that it can't train any of its AI models on your data, and that it definitely can't sell that data to advertisers. All of this is true, but that doesn't mean people believe Microsoft when it says these things. In fact, many have jumped to the conclusion that even if it's true today, it won't be true in the future.

— [Zac Bowden](#)

5:23 pm / [windows](#), [trust](#), [ai](#), [microsoft](#), [recall](#), [privacy](#)

[Update on the Recall preview feature for Copilot+ PCs](#) ([via](#)) This feels like a very good call to me: in response to [widespread criticism](#) Microsoft are making Recall an opt-in feature (during system onboarding), adding encryption to the database and search index beyond just disk encryption and requiring Windows Hello face scanning to access the search feature.

5:30 pm / [microsoft](#), [privacy](#), [security](#), [trust](#), [windows](#), [ai](#), [recall](#)

LLM bullshit knife, to cut through bs

RAG ->	Provide relevant context
Agentic ->	Function calls that work
CoT ->	Prompt model to think/plan
FewShot ->	Add examples

PromptEng -> Someone w/good written comm skills.

Prompt Optimizer -> For loop to find best examples.

— [Hamel Husain](#)

6:02 pm / [llms](#), [ai](#), [rag](#), [generative-ai](#), [hamel-husain](#)

[A Picture is Worth 170 Tokens: How Does GPT-4o Encode Images? \(via\)](#) Oran Looney dives into the question of how GPT-4o tokenizes images - an image "costs" just 170 tokens, despite being able to include more text than could be encoded in that many tokens by the standard tokenizer.

There are some really neat tricks in here. I particularly like the [experimental validation section](#) where Oran creates 5x5 (and larger) grids of coloured icons and asks GPT-4o to return a JSON matrix of icon descriptions. This works perfectly at 5x5, gets 38/49 for 7x7 and completely fails at 13x13.

I'm not convinced by the idea that GPT-4o runs standard OCR such as Tesseract to enhance its ability to interpret text, but I would love to understand more about how this all works. I imagine a lot can be learned from looking at how openly licensed vision models such as LLaVA work, but I've not tried to understand that myself yet.

11:30 pm / [ai](#), [openai](#), [generative-ai](#), [gpt-4](#), [llms](#), [tokenization](#)

[June 8, 2024](#)

[Expanding on how Voice Engine works and our safety research](#). Voice Engine is OpenAI's text-to-speech (TTS) model. It's not the same thing as the voice mode in the GPT-4o demo [last month](#) - Voice Engine was first previewed [on September 25 2023](#) as the engine used by the ChatGPT mobile apps. I also used the API version to build [my ospeak CLI tool](#).

One detail in this new explanation of Voice Engine stood out to me:

In November of 2023, we released a simple TTS API also powered by Voice Engine. We chose another limited release where we worked with professional voice actors to create 15-second audio samples to power each of the six preset voices in the API.

This really surprised me. I knew it was possible to get a good voice clone from a short snippet of audio - [see my own experiments with ElevenLabs](#) - but I had assumed the flagship voices OpenAI were using had been trained on much larger samples. Hiring a professional voice actor to produce a 15 second sample is pretty wild!

This becomes a bit more intuitive when you learn how the TTS model works:

The model is not fine-tuned for any specific speaker, there is no model customization involved. Instead, it employs a diffusion process, starting with random noise and progressively de-noising it to closely match how the speaker from the 15-second audio sample would articulate the text.

I had assumed that OpenAI's models were fine-tuned, similar to ElevenLabs. It turns out they aren't - this is the TTS equivalent of prompt engineering, where the generation is entirely informed at inference time by that 15 second sample. Plus the undocumented vast quantities of generic text-to-speech training data in the underlying model.

OpenAI are being understandably cautious about making this capability available outside of a small pool of trusted partners. One of their goals is to encourage the following:

Phasing out voice based authentication as a security measure for accessing bank accounts and other sensitive information

5:48 pm / [ethics](#), [text-to-speech](#), [ai](#), [openai](#), [generative-ai](#), [ai-ethics](#)

[Claude's Character](#) (via) There's so much interesting stuff in this article from Anthropic on how they defined the personality for their Claude 3 model. In addition to the technical details there are some very interesting thoughts on the complex challenge of designing a "personality" for an LLM in the first place.

Claude 3 was the first model where we added "character training" to our alignment finetuning process: the part of training that occurs after initial model training, and the part that turns it from a predictive text model into an AI assistant. The goal of character training is to make Claude begin to have more nuanced, richer traits like curiosity, open-mindedness, and thoughtfulness.

But what other traits should it have? This is a very difficult set of decisions to make! The most obvious approaches are all flawed in different ways:

Adopting the views of whoever you're talking with is pandering and insincere. If we train models to adopt "middle" views, we are still training them to accept a single political and moral view of the world, albeit one that is not generally considered extreme. Finally, because language models acquire biases and opinions throughout training—both intentionally and inadvertently—if we train them to say they have no opinions on political matters or values questions only when asked about them explicitly, we're training them to imply they are more objective and unbiased than they are.

The training process itself is particularly fascinating. The approach they used focuses on synthetic data, and effectively results in the model training itself:

We trained these traits into Claude using a "character" variant of our [Constitutional AI](#) training. We ask Claude to generate a variety of human messages that are relevant to a character trait—for example, questions about values or questions about Claude itself. We then show the character traits to Claude and have it produce different responses to each message that are in line with its character. Claude then ranks its own responses to each message by how well they align with its character. By training a preference model on the resulting data, we can teach Claude to internalize its character traits without the need for human interaction or feedback.

There's still a lot of human intervention required, but significantly less than more labour-intensive patterns such as Reinforcement Learning from Human Feedback (RLHF):

Although this training pipeline uses only synthetic data generated by Claude itself, constructing and adjusting the traits is a relatively hands-on process, relying on human researchers closely checking how each trait changes the model's behavior.

The accompanying [37 minute audio conversation](#) between Amanda Ascell and Stuart Ritchie is worth a listen too - it gets into the philosophy behind designing a personality for an LLM.

[# 9:41 pm](#) / [ai](#), [generative-ai](#), [llms](#), [anthropic](#), [claude](#), [amanda-askell](#), [ai-personality](#)

[Tree.js interactive demo](#) (via) Daniel Greenheck's interactive demo of his procedural tree generator (as in vegetation) [built with Three.js](#). This is really fun to play with - there are 30+ tunable parameters and you can export your tree as a .glb file for import into tools like Blender or Unity.

[# 9:43 pm](#) / [3d](#), [javascript](#)

[June 9, 2024](#)

Much like Gen X is sometimes the forgotten generation (or at least we feel that way), the generation of us who grew up with an internet that seemed an unalloyed good fall awkwardly into the middle between those who didn't grow up with it, and those for whom there has always been the whiff of brimstone, greed, and ruin around the place.

[A Link Blog in the Year 2024](#) (via) Kellan Elliott-McCrea has started [a new link blog](#):

Like many people I've been dealing with the collapses of the various systems I relied on for information over the previous decades. After 17 of using Twitter daily and 24 years of using Google daily neither really works anymore. And particular with the collapse of the social spaces many of us grew up with, I feel called back to earlier forms of the Internet, like blogs, and in particular, starting a link blog.

I've been leaning way more into link blogging over the last few months, especially now my own link blog [supports markdown](#). This means I'm posting longer entries, somewhat inspired by [Daring Fireball](#) (my own favourite link blog to read).

Link blogging is a pleasantly low-pressure way of writing online. Found something interesting? Post a link to it, with a sentence or two about why it's worth checking out.

I'd love to see more people embrace this form of personal publishing.

[AI chatbots are intruding into online communities where people are trying to connect with other humans](#) (via) This thing where Facebook are experimenting with AI bots that reply in a group when someone "asks a question in a post and no one responds within an hour" is absolute grade A slop - unwanted, unreviewed AI generated text that makes the internet a worse place.

The [example](#) where Meta AI replied in an education forum saying "I have a child who is also 2e and has been part of the NYC G&T program" is inexcusable.

M	T	W	T	F	S	S
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30