Adventures in Using Claude as a Virtual Research Assistant

Scott H. Hawley, Ph.D.

Prof. of Physics, CSM, Belmont University
Head of Research, Hyperstate Music [AI]
Creator, DSC/BSA/PHY 4420 - "Deep Learning & AI Ethics"
Founding Member, AI & Faith
former BD[AI]C Senior Data Fellow
former Technical Fellow, Stability AI

h/t Joe Miller

Belmont Data & AI Collaborative, April 9, 2025

Claude, specifically?

Why Claude (vs. Other Models)?

Besides their performance benchmarks, different LLMs have different "personalities" or "characters". Claude's personality has been described as "sycophantic", yet I still find…

- I find Claude the *least annoying* of the LLM apps *I've tried so far*
- ChatGPT tends to repeat the same mistakes, immune to correction
- Gemini (Studio/Pro) is maybe fine, esp. the new one.
- I use CoPilot for coding sometimes, when I'm locked out of Claude*
- Never used Perplexity (supposed to be good for references)
- I installed local-LLama once but never used it.

My comments are mostly about using Claude version **3.5** Sonnet.
v3.7, some aren't liking as much, but it's early to say how/why.

* more on this later

Claude, how?

I tend to use the **browser version**, claude.ai, and I pay $20/mo. for Pro.

There's also a phone app you can talk to – I use it occasionally.
There's also a Claude Desktop app. Use this to integrate MCP "agents" with other apps.

Two main modes of use:

- Chat - standalone discussions
- Projects - where you can drag-in a common "Knowledge Base" of PDFs, code, etc. for it to refer to for multiple (standalone) chats

Biggest "Gotcha": Longer chats use up more tokens, until…
You can get "locked out"/downgraded for a few hours whenever you use up your token allotment - **Even when paying for Pro**.

Then I'll tend to either take a break for 4-5 hours or switch to another LLM.

(Aside: So… why not use Cursor (app)?)

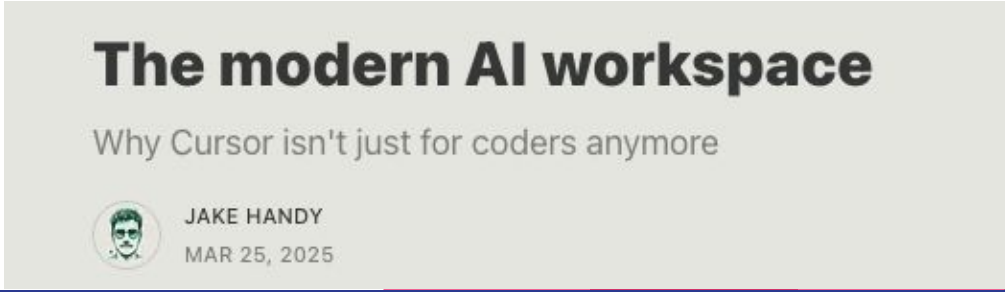
"Magical" pricing: Pay $20/mo for Cursor, get "unlimited"* **access to Claude** & **other** models. – **Same pricing** as Claude Pro!

* "unlimited" = you can still get throttled/locked out when "We are experiencing high demand…"

"How can such pricing work?"…It seems to be a "loss leader" AFAICT.

For a while, Cursor was just a coding app (an IDE), and I do more than coding. But now, Cursor is much more fleshed-out: **Jake Handy** of Pex (Nashville!) recently did an impressive blog post on using Cursor for "everything": https://handyai.substack.com

I started using Cursor, but no XP yet.



Assisting what kind of "research"?

1. Trying to understand things (concepts, journal papers,..)
2. Trying to build things (i.e. coding)

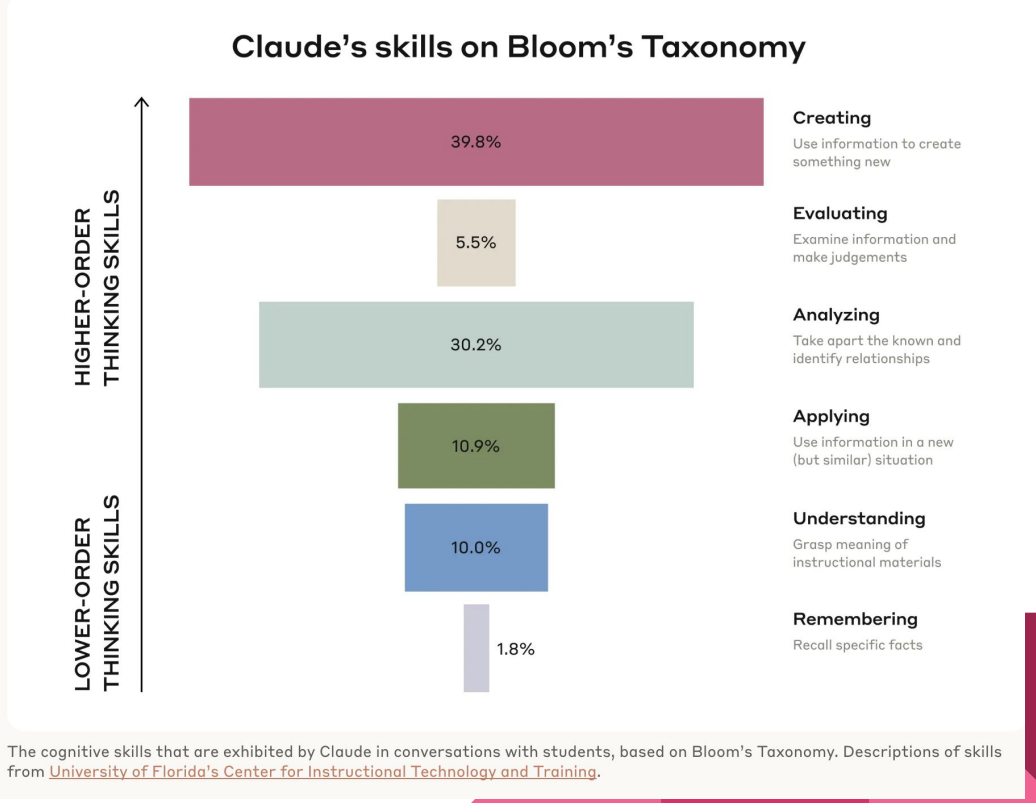
Often for me, 1 & 2 go together: I try to stay up-to-date with current SOTA, and write codes that teach and/or apply it

Use case #2 is a bit easier as a starting point…

Not for reference-finding "library research". (I don't trust Claude for *anything*.) Other tools do that better.

Aside: Students' Use of Claude (Apr 8, 2025)

https://www.anthropic.com/news/anthropic-education-report-how-university-students-use-claude



The cognitive skills that are exhibited by Claude in conversations with students, based on Bloom's Taxonomy. Descriptions of skills from University of Florida's Center for Instructional Technology and Training.

Using Claude to help

Build Things

What I do: "Ask for the moon!"

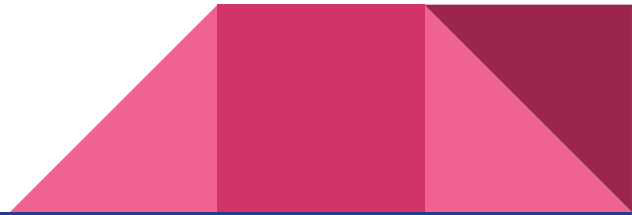
Write a paragraph describing the totality of your "dream" app. Don't hold back!

The model probably won't do all you ask for, but it'll give you enough of a start…

- such that a daunting project suddenly seems a lot more achievable
- it can overcome your initial coding ignorance (e.g. JavaScript, Swift, CSS…)

Nice feature of immediately rendering working HTML/JS/Graphs on the side:
Shareable as "artifacts". (example, two slides ahead)

But BEWARE: You may get "sucked in" ;-)...



Building Example: Standalone Tool

"**JSON Data Visualizer**" (Tue Apr 8)

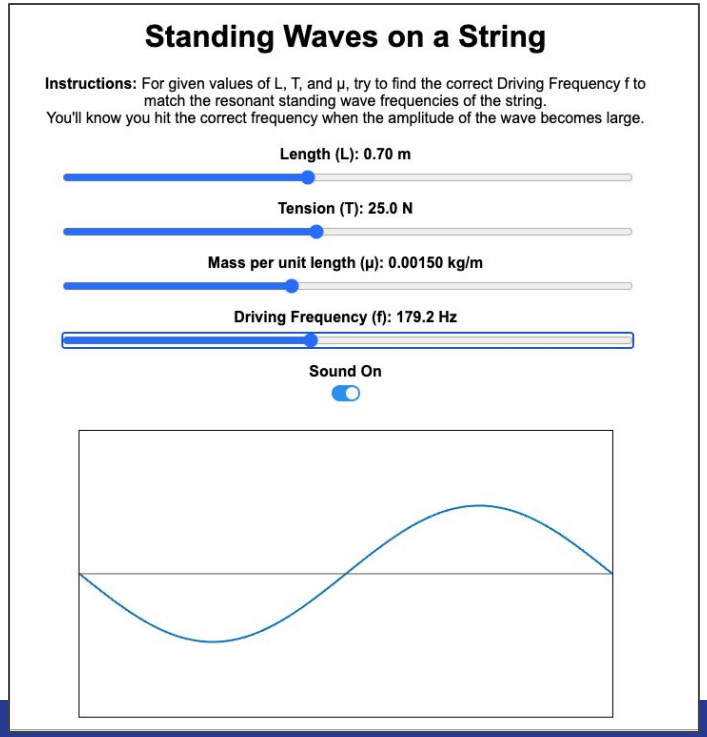
```
"I have a massive JSON file. In it are many measurements of music such as tempo, key, genre, instruments, moods, etc. that are generally time series data. What I'd like is your help is creating some kind of app that can graph this data. It might be nice if it were an HTML file with Javascript that I could run in a browser, where I can upload the JSON file, and then have all the graphs of the various features/activations appear. Any keys that have values that are single-valued (i.e. not lists) will just get printed , e.g. "Duration: 35 sec". Any keys with values that are lists, will be plotted as separate line graphs that just keep going down the page. All graphs will have the same horizontal width, that way they will be normalized by the song duration. The y-axis of each line graph should be the key from the JSON file."
```

…It didn't quite do that, but it got me "close enough". Demo Link

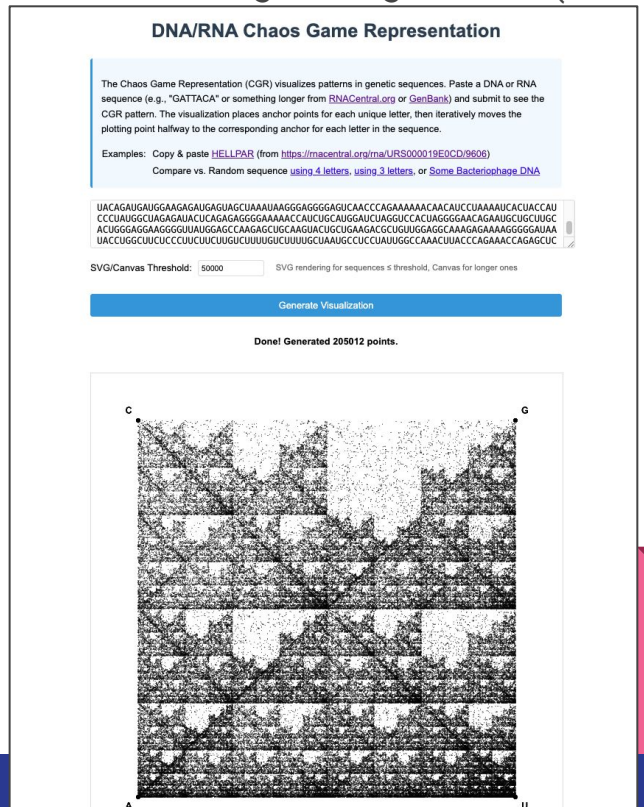
Goofs: It labeled graph titles instead of y-axes (I don't care)
Fixes/Upgrades: converting NaNs to Zeros; 2D heatmap plots

(Aside: Generate Instructional Tools!)

ChatGPT can do this kind of stuff too,
e.g. Fall 2025 virtual physics lab:



Or tools for colleagues in genetics...(Claude)



Sucked in: "Wow, I'm most of the way there! Just…"

Ah, but there's still a lot of work to be done. You may spend the next 3 days/weeks on something you might otherwise never have even started! ;-)

BIG Building Example: JS MIDI Editor

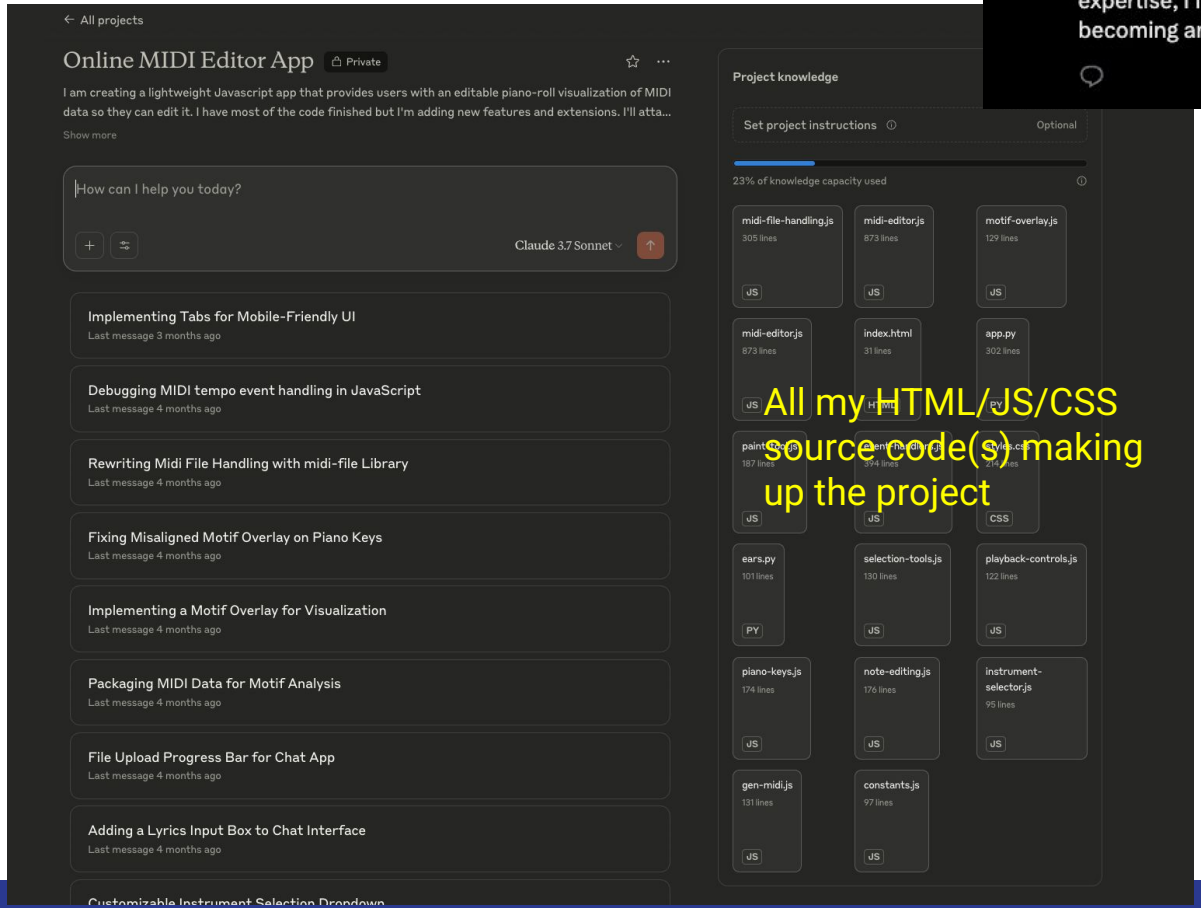
…and I don't/didn't really know JavaScript…

```
I want you to create for me an interactive graphical JavaScript MIDI editor.  That is,
a HTML-and-Javascript app that will let a user select a local MIDI file on their
machine and upload it, and specify a a tempo setting in BPM (if no tempo is given, the
app should estimate the tempo on its own), and display a graphical "piano roll" image
of the the MIDI for which the note onsets, endings, and durations are quantized to the
nearest 16th note interval.   Then users should be able to click and drag the middle of
a displayed note in order to change its pitch or timing info, and/or click and drag the
sides of the note to change its duration.  again quantized to the nearest 16th note.
The editor should have a triangular "play" button that will play the midi file -- you
may assume a piano sound for the sound to use.   Finally there should be a "Download"
button shaped like a downward-pointing arrow, such that when users click on it, the
edited MIDI data should download to their machine as a new midi file.  Go.
```

My Reaction: "Wow! That's amazing." (demo)

…Kept adding features, found the need to split it into multiple files (which Claude did for me)...

It's a Claude "Project"



^The code is not bug-free. You do need to (learn to) fix it. The LLM will sometimes even break it between prompts: "Hey. Why did you remove...?" (But Claude > ChatGPT IMHO)

Q: "Is this really Research, though?"
A: In my case, I needed/wanted the MIDI Editor as a front-end for some musical analysis AI I'm developing. It was getting to be a pain to have to launch external MIDI apps (that I couldn't customize).

Using Claude to help

Understand Things

"Conversations" - Claude as a Tutor

with apologies to Will Best. He's tried to help me!

```
I want to give myself a remedial education in the statistical lingo that I see
sometimes when people talk about neural network applications, especially
generative models. In particular, the terms "maximum likelihood [estimation]" and
ELBO, as well as the use of argmax, as well as the particular way that people try
to fit probability distributions by forming a product inside an integral sign
between a conditional probability and a probability over a latent variable (why
only products? why not some arbitrary function)? ... All these things kind of
confuse me.  So, I'm going to paste in a blog post where someone is trying to
explain these concepts -- after which I am still confused -- and you will help me
to understand it via discussion. Ok?
```

…and then we just go back & forth for a while until I get it!
Often this is a matter of translating lingo… Transcript

…ended up with this handy translation table!

Statistical Term	Coding/Deep Learning Term	What it Does
Prior p(z)	Initial latent distribution	Defines the space from which we sample latent codes
Likelihood p(x|z)	Decoder network	Transforms latent codes into observations
True posterior p(z|x)	Ideal encoder (can't compute directly)	Would perfectly map observations to their original latent codes
Approximate posterior q(z|x)	Encoder network	Maps observations to approximate latent distributions
Joint distribution p(x,z)	Complete model	Describes relationship between observations and latent variables
Marginalization	Integration over all z values	How we'd calculate p(x) if we could
Maximizing log-likelihood	Training the model	Optimizing parameters to make data more probable
ELBO	Training objective	Loss function we maximize during training

Case Study: Grokking "Flow" Models with Claude

It started with me talking to the iOS app Oct/Nov 2024: (after Kyle from kits.ai told me, "You're good at using toy models")

"I often find that one of the best ways to teach myself new topics in AI is to write code from scratch for a small toy model. For example, generative models like diffusion models VAEs, VQ VAEs, I would write Jupyter Notebooks using the MNIST dataset, And with the help of maybe a YouTube video or some other tutorial, I would write the entire code myself from scratch. The process of doing that would teach me a lot about how such model operates.
A topic that I'm interested in now are a series of newer generative AI models that all seem to have the word "flow" in their names. So, normalizing flows, rectified flows, flow matching models. I'm not really sure what the progression of all of these are. I know that rectified flows make a reference to normalizing flows. I'd like to pick one, maybe a recent one, and write a Jupyter Notebook sort of tutorial from scratch with your help. To walk through making a generative flow model for the MNIST Dataset. And maybe make it conditional so we can condition on the digit number that we want.
So Sorry. My cats are making a lot of noise right now. Ignore all of that. Okay. I'm back. Yeah. What can you tell me, first of all, about the difference between all these different types of models that have flow in their name? You're a pretty recent model so I think 2024 is sufficient for keeping track of what these models are doing. Let's start with the basic overview, and then we'll pick a particular type of flow model. I understand that all of these are sort of transforming the probability distribution and are similar to diffusion models. I understand diffusion models fairly well, so maybe we can use that as a background."

So began 3 weeks of non-stop back-and-forth with Claude…

Aside: a bit of

Background

leading up to this, for me

(Summer 2024)

Best Paper

Scaling Rectified Flow Transformers for High-Resolution Image Synthesis

Oral

Patrick Esser · Sumith Kulal · Andreas Blattmann · Rahim Entezari · Jonas Müller · Harry Saini · Yam Levi · Dominik Lorenz · Axel Sauer · Frederic Boesel · Dustin Podell · Tim Dockhorn · Zion English · Robin Rombach

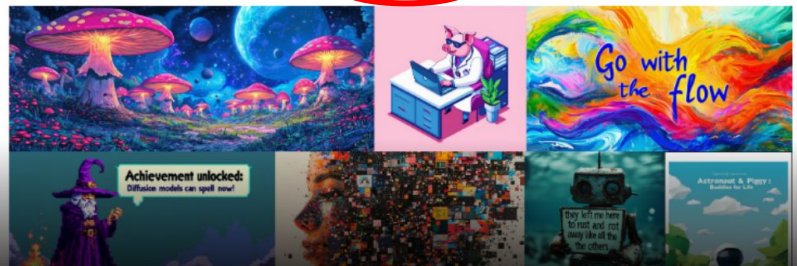
Wed 24 Jul 09:15 AM UTC

[Hall A1]

ICML
International Conference On Machine Learning

Scaling Rectified Flow Transformers for High-Resolution Image Synthesis

Patrick Esser * Sumith Kulal Andreas Blattmann Rahim Entezari Jonas Müller Harry Saini Yam Levi
Dominik Lorenz Axel Sauer Frederic Boesel Dustin Podell Tim Dockhorn Zion English
Kyle Lacey Alex Goodwin Yannik Marek Robin Rombach *
Stability AI



Aside: a bit of

Background

leading up to this, for me

(*Spring* 2024)

Pictures Of MIDI: CONTROLLED MUSIC GENERATION VIA GRAPHICAL PROMPTS FOR IMAGE-BASED DIFFUSION INPAINTING

Scott H. Hawley
Belmont University and Hyperstate AI

ABSTRACT

Recent years have witnessed significant progress in generative models for music, featuring diverse architectures that balance output quality, diversity, speed, and user con-

have the generative model fill in the notes in way that sounds appropriate, given the accompaniment. This process is somewhat akin to the "graphic notation" movement in 20th-century music composition championed by composers such as John Cage [5], Cornelius Cardew [6], and

Figure 1: The Motivating Idea. Top: MIDI piano roll image of a sample "graphical prompt" of rough shapes (in blue) of pitches for melody generation given accompaniment (green lines). Bottom: Sample generated output.

Submitted to ISMIR. Rejected. Diffusion code was big & slow. What about *going with flow*?

Talking to Claude…

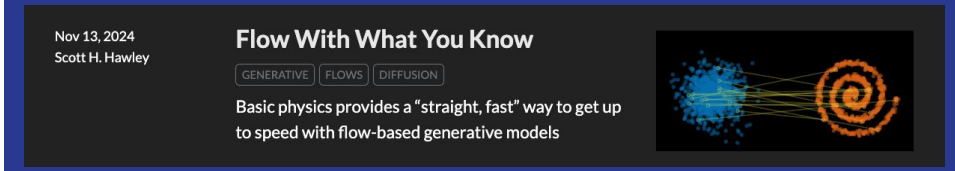
- Used Project mode, uploaded PDFs of papers like the ICML paper, the original Rectified Flow (RF) paper, and pasted in a tutorial… (All tutorials at the time were "walls of math")
- Spent a while trying to figure out the diff. between RF and "Flow Matching" (FM). Couldn't really find any. (Turns out they're *the* ***same thing*** invented by different teams ***simultaneously***)
- Claude & I figured out that ***there is no rectification mechanism*** for RF. (It is simply an outcome of the FMR and/or an extra step called "**ReFlow**".)
- Since I wanted it to also include executable code – to use as class lesson! – Claude was great at generating **boilerplate code and visualization routines.**
- Not sure if Claude or I suggested **higher-order ODE integration** scheme ;-)
- I noticed the ReFlow'd **streamlines were nearly time-independent**. Claude helped flesh out the connection to **Optimal Transport**
- ~10% of the text remains Claude-generated. Rest I wrote from scratch or edited heavily.
- Finished the blog-post-cum-Jupyter-notebook a couple days before **BDAIC Symposium!**

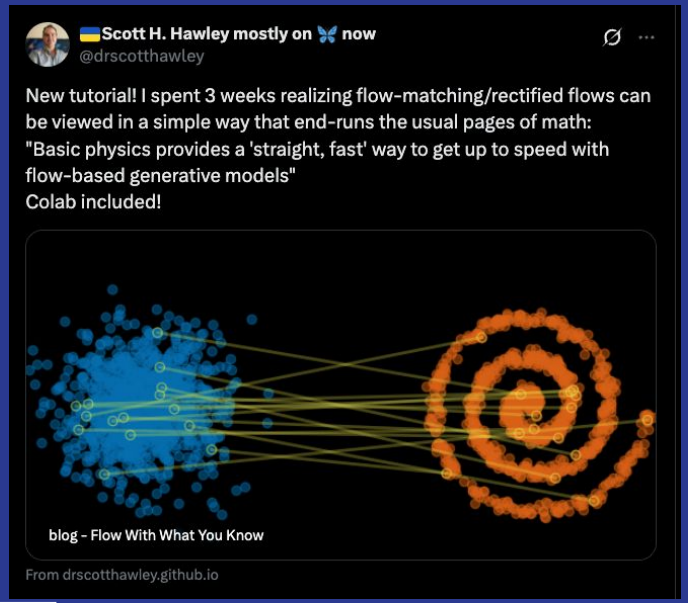
Long Story Short

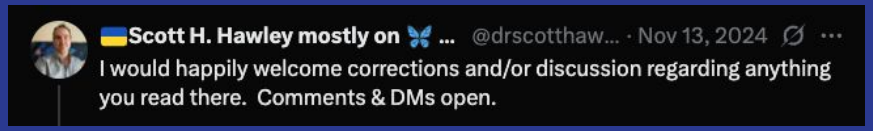
Worked on it obsessively from Nov 1 to 13 (> midnight)...

Figured that I must be missing something, that my "physicist's intuition" was mistakenly causing me to miss the need for the usual "wall of math." But figured that others would jump & in correct me...

...Posted it to my blog & on X.

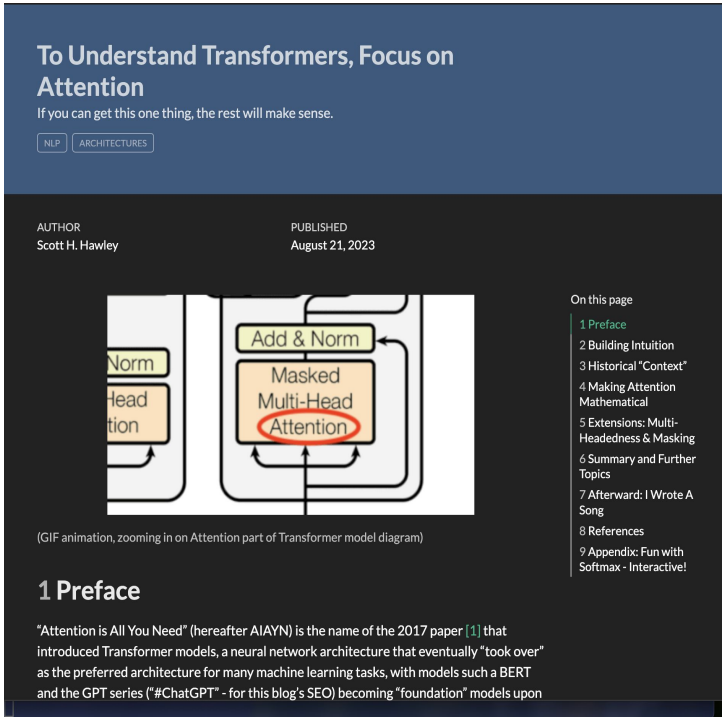




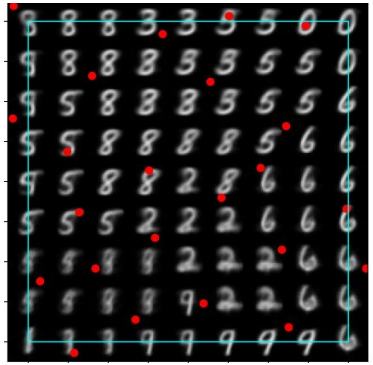


Aside: My Blog ⇔ Teaching ⇔ Research

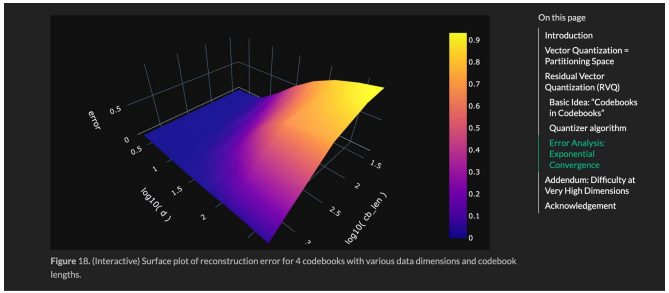
I spend 100s of hours writing tutorials



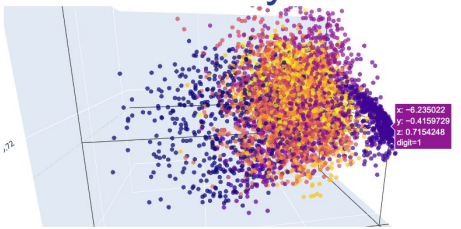
VAEs



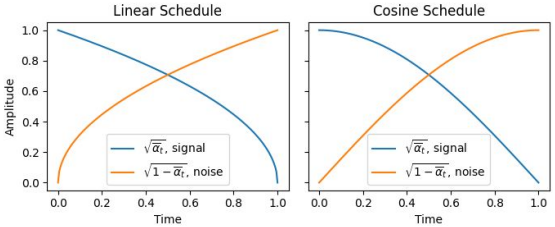
Residual Vector Quantization



Principal Component Analysis



Diffusion Models



"Noise schedules are just like cross-fades in audio production"

It was well-received…

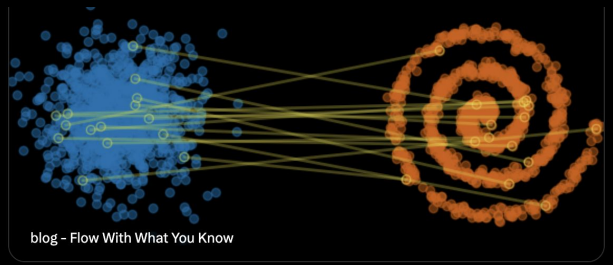
Sander Dieleman @sedielem · Nov 13

DeepMind

This blog post provides a very accessible overview of flow matching / rectified flow and reflow, based on intuitions from physics, rather than starting from probability distributions. The visualisations and animations are excellent, and the whole thing is also a colab!

> **Scott H. Hawley** @drscotthawley · Nov 13
>
> New tutorial! I spent 3 weeks realizing flow-matching/rectified flows can be viewed in a simple way that end-runs the usual pages of math: "Basic physics provides a 'straight, fast' way to get up to speed with flow-based generative models"
> Colab included! drscotthawley.github.io/blog/posts/Flo...
>
> Show this thread

2 22 144 15K



From drscotthawley.github.io

11:14 AM · Nov 13, 2024 · **36K** Views

View post engagements

15 74 442 424

Scott H. Hawley mostly on … @drscotthaw… · Nov 13, 2024

I would happily welcome corrections and/or discussion regarding anything you read there. Comments & DMs open.

@drscotthawley Hey folks, I'm almost finished writing

Jeremy Howard 11/12/2024 3:08 PM

It's great !

David Marx (@digthatdata.bsky.so… @DigThatD… · Nov 13, 2024

Physics really needs to become part of the boilerplate undergrad DL curriculum.

Nando de Freitas reposted

Quotes Reposts **Likes**

David
@DavidSHolz Follows you
Following

founder @midjourney, prev founder leap motion, nasa, max planck

Verified Followers **Followers** Following

Jeremy Krinitt
@jkrinitt Follows you
Following

Lead, Generative AI @ NVIDIA. he/him

Brandon Amos
@brandondamos Follows you
Follow

research scientist @MetaAI (FAIR), visiting lecturer @cornell_tech | optimization, machine learning, control, and reinforcement learning | PhD

Big Tech Alert @BigTechAlert · 2h

NEW @karpathy has started following @drscotthawley

Fast Forward

Nov 15-20ish: Submitted to "Blogpost Track" of **International Conference on Learning Representations (ICLR)**

Figured the reviewers would *eviscerate* it, but at least I'd get some feedback.

There was clearly a niche-need for tutorials on Flow Matching / Rectified Flows because…

In coming days/weeks, 4-6 more ***teams*** submitted ICLR Blogpost tutorials on FM/RF!

In December, DeepMind & Facebook both released major tutorials/codes on FM/RF!

Fast Forward

Jan 2025: It's **accepted** to ICLR!
Minimal reviewer comments!

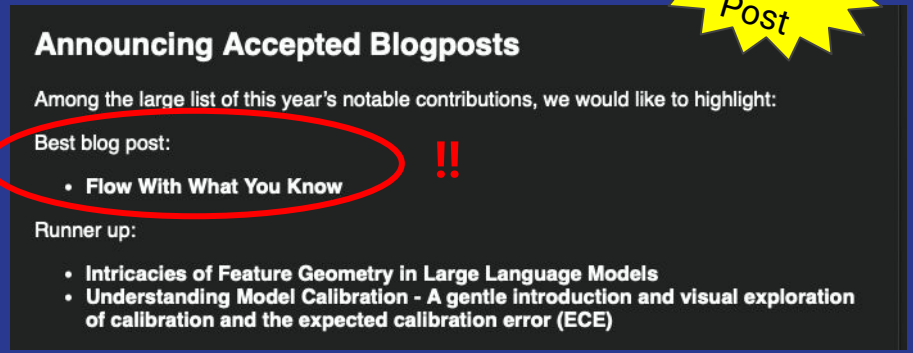
96 submissions total, 48 accepted.

I'm presenting in Singapore, April 23!

Fast Forward

March 24, 2025

Best Blog Post



Ok, but is that really "research" or just "teaching"?

Yes.

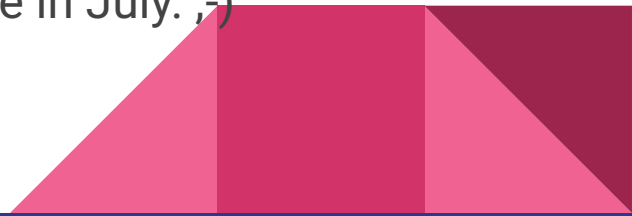
1. I want to use the flow methods for my (new) research.
2. But to use them, I need to understand them.
3. To understand them, it helps me to teach them (to myself & others).

Feynman: "If you want to really understand something, teach it."

The research-level code is coming along.

For that, I have *another* tutorial to finish & present in Rome in July. ;-)

And yes, teaching, we'll use it *all* my class this fall!



Note: Gen'ing yourself tutorials… isn't sure thing

Example prompt from Fri Apr 4 2025 (for work with student Brody):

```
"Let's make a tutorial on Rotary Position Embeddings (ROPE) in the form of a
jupyter notebook with text in markdown interspersed with python code and
images. (If you can't generate jupyter directly, then just separate the
markdown and code via the usual triple-backtick code blocks.)

The idea is that we'll explain how ROPE differs from other posititional
embeddings schemes such as the "classic" sin/cos scheme used in "Attention is
All You Need", and back this up with some code for a toy problem (in PyTorch)
that trains some kind of simple model on some standard small dataset, using
both "classic" and ROPE embeddings, and evaluate the performance of each."
```

…This "ask for the moon" attempt didn't go so great:
Code ran but the models didn't learn! *Work in Progress!*

Summary

- Claude helps us to build and understand: Use it to code ambitious tools or grasp complex concepts.
- "Ask for the moon," then fix the flaws: AI gives a huge head start; you learn fast by debugging its output.
- Accelerate your research: Tackle projects you wouldn't start otherwise (MIDI editor, Flow tutorial) & create high-impact work (ICLR award!).
- It's a powerful assistant, not a replacement: Expect errors, use your expertise, and stay in the driver's seat.

These summary points generated by Gemini 2.5 Pro ;-)

"The invite said 'hands-on activities'" …Right!

In your web browser, make a free account at Claude.ai – you can use a Google account. **Pick one** of these workflows to explore, using the remaining time to refine and share:

1. **Describe your dream app**, code, or data visualization. It can do multiple languages and frameworks (e.g., Python, C++, Swift, Javascript, HTML, LaTeX, SVG, JUCE, MatLab…). **Ask for the moon!** Don't worry whether you've specified everything or not. **Think big!** What's fun is that many things can be rendered as "artifacts" in the browser that you can immediately **see & interact with**. You may be amazed!
2. **Upload a PDF or two** (related) research papers and/or paste in text on a topic you'd like to understand. Ask Claude about the papers. **Discuss.** Try rephrasing based on your understanding. Probe it when it seems incorrect. Iterate until you understand!

Note: Once you start to see the **warnings about "This chat is getting long,"** ask it to summarize where you're at so you can carry on in a new chat. Copy & Paste the summary into a new chat to continue.

Sharing:

Anyone want to share what they did?

Thanks BDAIC, and good luck!

Follow-up:
scott.hawley@belmont.edu
@drscotthawley
https://drscotthawley.github.io

Anthropic Report:
https://www.anthropic.com/news/anthropic-education-report-how-university-students-use-claude