SDS PODCAST EPISODE 806: FIVE-MINUTE FRIDAY: LLAMA 3.1 405B: THE FIRST OPEN-SOURCE FRONTIER LLM

(00:05):

This is Five-Minute Friday on Llama 3.1, the first open-source frontier LLM.

(00:19):

Welcome back to the Super Data Science Podcast. I'm your host, Jon Krohn. Let's kick things off like we often do on Fridays with a recent review of the show. This one's from Penelope Bellegarde. She's founder of a consultancy in the UK called "The Data Touch", and she says, in an extremely generous review, she says, "You have no idea how much I value and LOVE this podcast. It's maths on steroids. It stimulates me so much intellectually every time, and so I have no words to express how just insanely good the podcast is and how grateful I am for its existence. Thank you so much. Your ability and passion to make the complex digestible day in day out is second to none." Wow, that is one I've got to send my mom, because that is the most generous review I've ever seen. Thank you so much Penelope, and I hope you continue to enjoy the show. It sounds like we're on the right track for you.

(01:18):

Thanks to everyone for all the recent ratings and feedback on whatever podcasting platform you use, Apple Podcast, Spotify or whatever. And also for all the likes and comments on our YouTube videos. Apple Podcast reviews in particular are helpful to us because they allow people to see written feedback on the show, and so I assume that helps grow the show. If you leave an Apple Podcast's feedback, I'll be sure to read that on air like I just read Penelope's.

(01:45):

Now let's dig into today's episode topic. Every week there are tons of supposedly cutting edge LLMs released, but I only highlight the actual game changers on this show. And the release of Meta's Llama 3.1 family of

models, specifically the behemoth 405 billion parameter variant, absolutely fits the game changer bill. The reason that this 405 billion parameter model is such a game changer isn't just the size, it's the effect of that size and the way that they trained it, because up until the release of that 405 billion parameter Llama 3.1 model last week, open-source models had lagged behind the proprietary frontier generative AI models like OpenAI's GPT models, Anthropic's Claude models, and Google's Gemini models.

(02:38):

Now, for the first time, according to testing and data released by Meta themselves, there's an "open-source" LLM that competes at the frontier against closed-source models. Specifically, they compared it against Claude 3.5 Sonnet, and open AI's GPT-4o, which are, in my view, in my experience, definitely the two best closed-source models, the two models that are really at the frontier. So I'm going to go over some model valuation details in a moment, but a few caveats before I do that. Again, Meta released these results themselves, so we'll have to see how high quality third party LLM leaderboards, like LMSYS Chatbot Arena end up rating it. In addition, they interestingly didn't compare against Google Gemini, which also is at or near the frontier of LLM capabilities, and apparently they didn't do that because they claimed, Meta claimed, to have difficulty replicating the kinds of figures that Google published on that model family.

(03:45):

So something's up there either. Based on my experience with Gemini, my guess is that Meta might be being honest there, and that there could just be problems getting Google Gemini results that compare to what we see from Claude 3.5 Sonnet, GPT-4o, or Google's own claims of what Gemini can do. And the last caveat here is that none of these Llama 3.1 models are actually really open-source, because Meta didn't provide source code or training data perhaps to try to avoid copyright infringement lawsuits. So a term like "open weights" might be more appropriate than open-source. But

anyway, with all of those caveats aside, the results do look really impressive. So if you're watching the video version of this podcast, I actually have the table from Meta's blog post up here, and the main takeaway here is that there's a few benchmarks that GPT-4o wins, there are 15 of them shown in this chart. And yeah, Google 4o wins on three of those, and then Claude 3.5 Sonnet, and Llama 3.1 405B, they split the remaining benchmarks.

(05:14):

And it doesn't seem like that's in one particular category, that one is stronger than the other across comprehension in general. It's about split. It looks actually like Claude 3.5 Sonnet seems to do a bit better on the code output kinds of tasks, those kinds of benchmarks. But otherwise there don't seem to be any particularly strong patterns. Actually, another one here that I'm noticing just now is that on longer content, those kinds of benchmarks, the Meta Llama 3.1 405B does seem to outperform. So that's at a high level the comparison, but according to these benchmarks, Llama 3.1 405B is at the frontier, and we've never had an open-source model like that. Anytime you see the other kind of open-source model releases, like the big mixed draw model from Mistral, those were being compared against slightly outdated proprietary closed-source models. So they're being compared against GPT-3.5 instead of against GPT-4. But this one, this open-source model for the first time is being compared against the big frontier closed-source proprietary LLMs, like GPT-4o, Claude 3.5 Sonnet, and it is comparable.

(06:33):

And again, benchmarks, as I've said on the show many times, aren't always the best way to evaluate a model because you could be fine-tuning your models to perform really well on the benchmarks themselves, as opposed to just being generally good at the kinds of tasks that the benchmarks are trying to measure. But with all of that said, something that Meta did

differently here is they put extra expense into comparing not just on benchmarks but also on human evaluations. So by investing money and time in doing human evaluations, Meta, again, this is their own data, their own test, but they're able to show that Llama 3.1 405B, compared with GPT-4, with GPT-4o, with Claude 3.5 Sonnet, it wins, loses about as often. So the main takeaway, and again, if you're watching the YouTube version, you can actually see this chart or you can check it out on the Meta blog, which I've got a link to in the show notes, a quarter of the time Llama 3.1 wins, a quarter of the time GPT-4 or Claude 3.5 Sonnet win, and about half of the time human evaluators rate the output as a tie.

(07:55):

So pretty close here. Something that I did notice that is interesting, however, is that while on the benchmarks that I was just talking about, it was Claude 3.5 Sonnet and Llama 3.1 that were neck and neck, with GPT-4 Omni taking first place in fewer of those 15 benchmarks. On this human evaluation, head-to-head, that was the only model that Llama 3.1 slightly underperformed against. So against GPT-4 and Claude 3.5 Sonnet, Llama 3.1 did basically exactly the same in terms of wins, losses and ties. But GPT-4o had slightly more wins, 29%. That's actually non-negligible. GPT-4o won 29% of the time on human evaluations, while Llama 3.1 405B only won 19.1% of the time. So that does seem to me... I don't know exactly what that experience feels like as a user, but it looks non-trivial to me on that chart, and I imagine that it is that way.

(09:09):

Anyway, regardless, for the first time, according to both benchmarks, as well as these human evaluations from Meta, we're seeing an open-source model that's competing at the same frontier as the closed-sourced frontier models. For a bit more context, it was back in April during the release of Meta's Llama 3.0 family of models that they teased that a 405 billion parameter model designed to compete with closed proprietary models was

in the works. So congrats on them for now achieving that a few months later with this release. But as part of this big 405B model release right now, they're calling it Llama 3.1 as opposed to the 3.0 that we had in April. That might already be obvious. But they're also releasing updated more capable versions of their smaller Llama models, too. So with this llama 3.1 release of the 405B model, they're also releasing a smaller 8 billion parameter model and a 70 billion parameter model. And according to Meta's own benchmark evaluations again, these are now the state-of-the-art for models around that size as well. Outcompeting the likes of Mistral's biggest Mixtral model, and Google's Gemma model family.

(10:24):

So there is, again, there is a chart in the paper showing those results. Speaking of model families, like their Llama 3.0 release, Meta with this 3.1 release has also provided models that are fine-tuned for different application types. Specifically, they've provided an instruction following model as well as models optimized for chat, and those are available at the different sizes. So there's a huge, what they call, herd of Meta models being released in this Llama 3.1 release. The other noteworthy items on this Llama 3.1 released last week are that the context window is much larger. So Meta has significantly expanded the context length of their models to 128,000 tokens. This is a small fraction of the multimillion token context windows of, say, Google Gemini, but nevertheless, this is way more than enough context for most use cases because it means that you can squeeze about a hundred thousand words of context in, and most novels are shorter than a hundred thousand words.

(11:28):

Another big thing here to note is that the new Llama models are multilingual, so they support eight languages out of the box. Those are English, German, French, Italian, Portuguese, Hindi, Spanish and Thai. And another big note here is that this release is not just about raw

capabilities. Meta is also emphasizing responsible AI development. So they've released new safety tools alongside the models. These include Llama Guard 3 for content moderation, and prompt guard to protect against prompt injection attacks. For those of you out there who are on the more technical side, I've also got a few notes here for you on the model architecture. So Llama 3.1 405B was trained on over 15 trillion tokens, leveraging 16,000 Nvidia H100 GPUs. Surely that's unprecedented scale for an open-source model. They used a decoder only transformer architecture, and you can learn more about decoder only architectures by checking out Super Data Science episode number 747, if you want all the detail on that.

(12:36):

But this with this decoder only transformer architecture, they opted for that instead of a mixture of experts architecture to maximize training stability. You can hear more about the mixture of experts approach by checking out Super Data Science episode number 778. But the idea is that training a whole bunch of these different submodel experts can be very difficult. And so to maximize stability, Meta just went with a single big architecture. And another model architecture note here is that in terms of training, their post-training, so after pre-training on all the tokens, they did post-training involving supervised fine-tuning and direct preference optimization to create high quality synthetic training data each round, and improve performance on desired capabilities.

(13:30):

So they had a loop that they looped over during training, where during each round of that loop, they generated synthetic data, and then used supervised fine-tuning and direct preference optimization to create and decide on which synthetic training data were the most high quality for that particular round, for that stage that the model training was in, to optimize performance, to improve performance on desired capabilities. And if you

want to learn more about direct preference optimization, you can check out episode number 791 of this podcast.

(14:03):

So a final question for you, or a final thought for you, whether you're a technical person or not, would be, why would Meta do this open-source release? They had to train this model on 16,000 Nvidia H100 GPUs. This is so expensive and they're using some of their... They have people that are earning probably more than a million dollars a year, tons of people earning that much working on this project. So why would Mark Zuckerberg get behind that? Well, he's actually written a whole blog post about it that you can read to get all the detail. I've linked to it in the show notes. The blog post is called Open-source AI is the Path Forward. My personal thoughts on this are that Meta is probably doing this mostly to compete on talent, because top AI researchers want to work at the frontier. So this helps attract people from OpenAI or Google or Microsoft to come work at Meta.

(15:00):

Theoretically, Mark Zuckerberg says its security is a big deal here. And this is debatable because theoretically, this does allow anyone to test, but releasing the model weights does also allow any actors to do whatever they want with those weights. So there's a lot of flexibility. And so I think that this is, in addition to the talent competition and I guess related to the talent competition, this is mostly about undercutting Meta's big tech rivals, so Google or OpenAI, which is partnered with Microsoft. Those companies are releasing huge, powerful proprietary models, and by open sourcing comparable level capabilities that undercuts and commoditizes frontier gen AI capabilities that their rivals have. Unlike all the other big tech firms, Meta makes essentially all of its money from advertisers, so unlike the other big tech firms, Meta doesn't cannibalize, say, subscription AI services by offering frontier model weights for free.

(16:02):

Now relatedly, if Meta is going to undercut their big tech rivals, they need to make Llama 3.1 405B widely accessible and usable. And they've done just that through partnerships with the likes of AWS, Databricks, Snowflake and Nvidia, they've ensured the 405B model can be used even from within the Google Cloud and Microsoft Azure environments. And so this is all critical because, while you could fit the 8B variants of the Llama 3.1 family on a single GPU, it would take many GPUs, advanced MLOps, and a lot of expense to do any training or even just inference with a 400 billion parameter model. So that's why they had all these partnerships to make sure you can access them, to access and use such a big model without all that expense or MLOps expertise. All in all, the impact of this release could be far-reaching. By making such a powerful model openly available, Meta is democratizing access to cutting edge AI tech.

(17:03):

This could lead to a surge of innovation across various industries, from healthcare to education to scientific research. You name the industry, having open-source AI models at the frontier could make a difference in them. Indeed, Meta have already highlighted examples of Llama models being used to, say, guide medical decision making and separately to organize healthcare in Brazil. So how are you going to change the world, or maybe just make your customers happier through open-source LLM tech? You may have some ideas already, or you can chat with an LLM to get some ideas of how LLMs could make a big difference in your particular industry. For the first time, you can now fine tune, deploy, and build upon an open-source LLM that operates at the frontier. I'm not a huge personal fan of Meta or Mark Zuckerberg in general, but I do applaud them for investing so much in making LLMs available to us, even if from their perspective, it's mostly about undercutting rivals.

(18:02):

Anyway, whatever ideas you have for taking advantage of this unique moment in history, you can head to GitHub or Hugging Face to get started with Llama 3.1 today. We've got the links for you in the show notes.

(18:13):

All right, that's it for today's episode. If you enjoyed today's episode or know someone one who might, consider sharing this episode with them, leave a review of the show on your favorite podcasting platform, tag me in a LinkedIn or Twitter post with your thoughts. I'll respond to those. And if you haven't already, of course subscribe to the show. Most importantly, I just hope you'll keep on listening. Until next time, keep on rocking it out there, and I'm looking forward to enjoying another round of the Super Data Science Podcast with you very soon.