The DeepSeek Effect: Notes on Reinforcement Learning and Emerging Reasoning Capabilities

Discussion ensuing after the release of DeepSeek, Jan 20th – Feb 2nd, 2025, Linkedin

It seems that [GRPO (Group Relative Policy Optimization)](https://arxiv.org/abs/2402.03300) is all we need to teach LLM to reason as well as ChatGPT o1 or better. GRPO is a variant of [PPO](https://arxiv.org/abs/1707.06347) which even Nike uses these days. No supervised fine tuning, no [RLHF](https://rlhfbook.com/) - [just PPO does apparently impressive job](https://www.linkedin.com/posts/charlesmartin14_thanks-to-deepseek-no-one-is-sleeping-activity-7288988183332077569-9G6O/).

Jim Fan wrote:

*We are living in a timeline where a non-US company is keeping the original mission of OpenAI alive - truly open, frontier research that empowers all. It makes no sense. The most entertaining outcome is the most likely.  
  
DeepSeek-R1 not only open-sources a barrage of models but also spills all the training secrets. They are perhaps the first OSS project that shows major, sustained growth of an RL flywheel.  
  
Impact can be done by "ASI achieved internally" or mythical names like "Project Strawberry".   
Impact can also be done by simply dumping the raw algorithms and matplotlib learning curves.  
  
I'm reading the paper [1]:*  
*> Purely driven by RL, no SFT at all ("cold start"). Reminiscent of AlphaZero - master Go, Shogi, and Chess from scratch, without imitating human grandmaster moves first. This is the most significant takeaway from the paper.  
> Use ground truth rewards computed by hardcoded rules. Avoid any learned reward models that RL can easily hack against.  
> Thinking time of the model steadily increases as training proceeds - this is not pre-programmed, but an emergent property!   
> Emergence of self-reflection and exploration behaviors.  
> GRPO instead of PPO: it removes the critic net from PPO and uses the average reward of multiple samples instead. Simple method to reduce memory use. Note that GRPO was also invented by DeepSeek in Feb 2024 ... what a cracked team.*

*[2] is a \*second\* paper dropped with tons of RL flywheel secrets and \*multimodal\* o1-style reasoning is not on my bingo card today. Kimi's (another startup) and DeepSeek's papers remarkably converged on similar findings:  
  
> No need for complex tree search like MCTS. Just linearize the thought trace and do good old autoregressive prediction;  
> No need for value functions that require another expensive copy of the model;  
> No need for dense reward modeling. Rely as much as possible on ground truth, end result.   
  
Differences:  
  
> DeepSeek does AlphaZero approach - purely bootstrap through RL w/o human input, i.e. "cold start". Kimi does AlphaGo-Master approach: light SFT to warm up through prompt-engineered CoT traces.  
> DeepSeek weights are MIT license (thought leadership!); Kimi does not have a model release yet.  
> Kimi shows strong multimodal performance (!) on benchmarks like MathVista, which requires visual understanding of geometry, IQ tests, etc.  
> Kimi paper has a LOT more details on the system design: RL infrastructure, hybrid cluster, code sandbox, parallelism strategies; and learning details: long context, CoT compression, curriculum, sampling strategy, test case generation, etc.*

Charles H. Martin wrote:

*Revenge of the quants. So what have we learned from DeepSeek ? One is that you can get very far with reasoning with just a simple Reinforcement Learning algo (GRPO) and a large training set of “verifiable" math & coding problems.   
  
See this link for a discussion of DeepSeekMath and where the introduced the y GRPO algorithm: "DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models"  
paper:* [5] *"First, we harness the significant potential of publicly available web data through a meticulously engineered data selection pipeline.   
  
Second, we introduce Group Relative Policy Optimization (GRPO), a variant of Proximal Policy Optimization (PPO), that enhances mathematical reasoning abilities while concurrently optimizing the memory usage of PPO."  
  
Other Key Innovations:  
1. Low-rank KV cache compression (MLA architecture)  
2. MoE FFN architecture   
3. Multi-token prediction head  
(source:* [*here*](https://x.com/ExponentiallyBe/status/1882466767601070313)*)  
  
"A model that is* ***almost*** *as good as o1 for 30x cheaper!"  
(source:* [*here*](https://x.com/LinusEkenstam/status/1882379270787616961)*)  
  
"It's wild to me that they they did this with no finetuning prior to the RL stage. R1 learns to reason on its own, like AlphaZero. During training, they observed the model learning to use advanced reasoning techniques - an "AHA" moment."  
(source:* [*here*](https://x.com/maxwinga/status/1881372837296083440)*)  
  
You can try it online here:* [*https://chat.deepseek.com*](https://chat.deepseek.com/) *Wait, there's more! You can run the full DeepSeek R1 671B on just two M2 Ultra Macs (source:* [*here*](tps://x.com/awnihannun/status/1881412271236346233) *and* [*here*](https://x.com/ronaldmannak/status/1881515278091342260)*)  
  
And people are even running DeepSeek R1 (distilled to Qwen 1.5B) on their iPhone 16 (link* [*here*](https://x.com/awnihannun/status/1882105560201617903)*).*

Peter Gostev wrote*:*

*DeepSeek's r1 model release caused quite a stir in the AI community, yielding a lot of good and bad takes.  
  
First thing out of the way - it is not a PsyOp or fakery, it is a good model created by a talented team of researchers with a lot of small innovations, especially on efficiency side. The fact that it is released under an MIT license with a nice research paper is also a shift - it is by far the strongest (mostly) open source model for a while.  
  
The most interesting thing for me about what they write in their paper is how well reinforcement learning (RL) is working. The coolest thing is all of the emerging behaviours (i.e. not explicitly selected for by researchers) that these models start to exhibit after all the RL training:  
   
 - Reflection: The model spontaneously rechecks and reevaluates steps mid‐solution.  
 - Self‐Correction: It spots mistakes and fixes them on the fly (e.g. incorrect factorisation).  
 - Aha Moments: It learned to pause, say “Wait,” and re‐start with a fresh chain.  
 - Longer outputs: It learned that longer outputs can yield better results   
 - Language Mixing: It can blend multiple languages or code snippets, since it only cares about getting the right final answer, not readability for humans.  
  
All these behaviors happened without humans explicitly teaching these techniques. This only came from rewarding the correct answer.  
  
The readability is a very interesting point. The original R1-Zero model was trained on pure RL and its 'thinking' was not comprehensible for researchers. If a model performs extremely well but 'thinks' in a completely incomprehensible way for humans, would we be ok with this kind of model? Perhaps forcing it to only 'think' in a single language (e.g. English) would limit how well it can 'think'. If it learned a concept in Chinese or Hebrew, would it be forced to avoid it for our convenience?  
  
To get to DeepSeek R1, they trained the base v3 model on some initial examples of the 'correct' chain of thoughts, which taught the model to think in a single language (mostly) and format chains of thought correctly.   
  
And the last very cool finding was that you can train smaller models on the reasoning chains (incl. 1.5b model) and increase their performance dramatically.  
  
All and all, this is a very interesting model and more importantly paper. This tells us what likely OpenAI (and others) are already doing and how impressive RL can be.*

Xu Ning wrote:

*I think many folks are probably reading DeepSeek papers this weekend. My thoughts on DeepSeek:*

*First, I recommend reading both DeepSeek V3[1] and DeepSeek R1[2] papers. Both have a lot of ingenuity in addressing different challenges. I view DeepSeek V3 primarily engineering optimizations on top of V2 that significantly reduced the compute needed to train a large MoE model.*

*DeepSeek-R1 (Zero) is the "reasoning-reinforced" version of V3 and it's truly a breakthrough. Instead of creating (usually with human effort) reasoning/CoT SFT data, R1 leverages RL to make reasoning capability \*emerge\*. This is very similar to the AlphaZero moment. I believe R1 will have profound impact on the direction. While some may say it signals the end of scaling law, it doesn't. You still need a good base model, one with large capacity, since the paper indicated that you can't RL a small model to have good reasoning. However, the role of SFT is diminished. You can almost say, that the model itself already has the capability of reasoning via it's learning of human text, but it has not been elicited out without SFT (no longer needed now) or RL.*

Cameron Wolfe wrote:

*One of the most overlooked contributing factors to the success of DeepSeek-R1 is the Mixture-of-Experts (MoE) base model from which it is derived–DeepSeek-v3…*

*The recently-proposed DeepSeek LLMs–including DeepSeek-R1, DeepSeek-v3 and more–have made waves within LLM research for a variety of reasons:*

*- Their weights are shared publicly.*

*- They come with technical reports that share many details.*

*- Their performance is impressive—on par with many closed models.*

*- Their training costs are (relatively) reasonable.*

*The DeepSeek-v3 base model, upon which DeepSeek-R1 is based, is an MoE LLM that makes several unique design choices to maximize efficiency and performance.*

*(1) Multi-head latent attention (MLA). DeepSeek-v3 adopts MLA, which is an efficient attention variant. MLA aims to minimize memory consumed of the model’s KV cache via a low-rank, joint projection that allows us to represent all key and value vectors with a much smaller (latent) vector. Adopting MLA decreases the size of DeepSeek-v2’s KV cache by over 93% compared to a 67 billion parameter dense model.*

*(2) Fine-grained / Shared Experts. DeepSeek-v3 uses fine-grained experts–a larger number of smaller experts–in its MoE layers. A subset of these experts are shared, which encourages specialization among experts while minimizing redundant information between experts.*

*(3) No load balancing loss. A novel, auxiliary-loss-free load balancing strategy is used by DeepSeek-v3 that simply adds a per-expert bias term to the selection of Top-K experts. At each iteration, the bias term for each expert is either increased or decreased by a fixed factor γ based upon whether that expert was underloaded or overloaded, respectively.*

*(4) Multi-Token Prediction. DeepSeek-v3 uses an MTP training objective. This objective is an extension of the supervised, cross entropy-based next token prediction objective that is used almost universally for training LLMs. Instead of predicting the next token for each token within a sequence, MTP predicts D future tokens. These predictions are made sequentially by a set of additional modules that are added to the model’s architecture.*

*Putting it together. DeepSeek-v3 has 671 billion total parameters and 37B active parameters. The model is pretrained on a corpus of 14.8 trillion tokens and aligned via the following steps:*

*- A two-stage context extension procedure (to 32K and 128K).*

*- Further SFT + RLHF.*

*- Distillation from the DeepSeek-R1 reasoning model.*

*DeepSeek-v3 outperforms closed-source models and achieves similar performance to even the best closed LLMs. The model–trained using a novel FP8 mixed precision framework–is economical (~$5.6M to train the final model) and is a great base model for DeepSeek-R1.*

Louis Scott wrote:

*A few thoughts after my quick read of the DeepSeek-R1 paper.*

*From the paper [1], table 3: It is reinforcement learning that drove the gains. A clue to the training efficiency is the use of Group Relative Policy Optimization [5]. It speaks in bladerunner patois, in 2.3.2 of [1], the paper notes language mixing in chain of thought (CoT) reasoning. Finally, Yann LeCun's comment [11] yesterday on the big take away:*

*"Open source models are surpassing proprietary ones."*

A math equations and formulas on a white background

Description automatically generated

Niccolo Gentile wrote:

*There is, understandably, quite a lot of buzz ongoing around the latest DeepSeek-R1.*

*As a matter of fact, it consists in an improved version, via Reinforcement Learning (and Group Relative Policy Optimization in particular) of the already existing DeepSeek-V3-Base model.*

*Under an architectural point of view, DeepSeek-V3 consists in a Mixture-of-Expert model encompassing 685B total parameters, of which 671B main weights and 14B for Multi-Token Prediction: only 37B of them get activated for each token.*

*In a previous post, I already reviewed DeepSeek's MoE architecture, described in "DeepSeekMoE: Towards Ultimate Expert Specialization in Mixture-of-Experts Language Models", Dai et al. 2024, based on Fine-Grained Expert Segmentation and Shared Expert Isolation.*

*However, DeepSeek-V3's other main ingredient, and already used in DeepSeek-V2, is its Multi-Head Latent Attention (MLA) mechanism.*

*How does it work, and how is it different from the classic Vaswani et al. 2017's Self-Attention - (potential) Multi Head Attention (MHA)?*

*In the standard MHA mechanism, for each token t, let h\_t be a d sized vector, where d is its embedding dimension (input at the attention layer). Let also n\_h be the number of heads and d\_h each of them's dimension.*

*In MHA, you first need to produce three vectors, q\_t, k\_t, and v\_t, each of dimension (d\_h \* n\_h), by multiplying h\_t with three matrices W\_q, W\_k, and W\_v, each of dimension, in turn, (d\_h \* n\_h) x d.*

*Opposed to this approach, MLA's main innovation - as per equations 9, 10, and 11 - consists in first computing a low-dimensional latent representation of h\_t, called c\_t^KV and of dimension d\_c << (d\_h \* n\_h), by multiplying it with a low-dimensional projection matrix W^DKV of dimension d\_c x d (9).*

*Then, only on this c\_t^KV one computes k\_t and v\_t using two upward projecting matrices, W^UK and W^UV, aiming at projecting again the latent representation to a (d\_h \* n\_h) vector.*

*Hence, where are the savings? They can be found in a sharply reduced need for KV-Caching, measured as number of elements that need to be stored.*

*In standard MHA, given l layers, one needs to store (2 \* n\_h \* d\_h \* l) elements per token. Conversely, in MLA, only around (d\_c + d\_h^R) \* l elements need to be stored, where d\_h^R denotes the "per-head dimension of the decoupled queries and keys" to be able to compute Rotary Positional Embeddings" (all discussion on this in Section 2.1.3).*

*As I mentioned, the same MLA mechanism is used both in DeepSeek-{V2, V3}, where the latter is the basis of R1. I personally found the MLA explanation clearer in DeepSeek-V2, but you can also find it in DeepSeek-V3 model card.*

*Link to both model cards in the comments: in the picture, from DeepSeek-V2's one, the three main equations controlling KV-caching compression. In the comments, also a link to my review of the aforementioned DeepSeek MoE.*

A close-up of a text

Description automatically generated

Charles H. Martin wrote:

*DeepSeek WeightWatcher Update: 60% done. Below I show a histogram plot of the layer alphas, and the Correlation Flow plot (layer id vs alpha)*

*While most of the layer alphas concentrate around 3.5-3.6, we now see a long tail of weakly converged layers starting to appear very early and sporadically even in the earlier layers. This is very pronounced, and while it is a long tail, it is a little surprising to see such layers appear so early (i.e, close to the data).*

*For example, for comparison, the Llama3.1-70B model does not show so many undertrained layers until they are half-way from the data to the labels*

[*Llama-3.1-8B-Instruct, weight|watcher.ai*](https://weightwatcher.ai/models/Llama3.1/Llama-3.1-8B-Instruct.html)

*We do see a similar pattern, however, in the Qwen-2.5 models*

[*Qwen2.5-14B-Instruct, weight|watcher.ai*](https://weightwatcher.ai/models/Qwen2.5/Qwen2.5-14B-Instruct.html)

*This is not necessarily bad, but it does indicate that, as with many modern LLMs, they models are widely overparameterized and probably can be compressed down and/or distilled down (as we have seen done so effectively)*

*I think to get to AGI, among other things, we need to solve this, and figure out how to train models that concentrate the correlations better so that there is a stronger signal/noise ratio across all layers.*

Charles H Martin wrote:

*Breaking news: "Researchers recreated DeepSeek's core technology for just $30!"*

*"The team successfully replicated DeepSeek R1-Zero’s reinforcement learning (RL) capabilities using a compact language model with just 3 billion parameters. Despite its smaller scale, the model demonstrated self-verification and search capabilities, allowing it to refine its own responses iteratively - key features of DeepSeek’s advanced AI."*

*TinyZero: A Minimalist Reinforcement Learning Framework for $30*

[*https://github.com/Jiayi-Pan/TinyZero*](https://github.com/Jiayi-Pan/TinyZero)

*"The recipe: [They] follow DeepSeek R1-Zero alg - Given a base LM, prompts and ground-truth reward, [they] run RL. [They then] apply it to CountDown: a game where players combine numbers with basic arithmetic to reach a target number...*

*and it just works!*

*"The model starts from dummy outputs but gradually develop tactics such as revision and search. "*

*Of course, this is for a very restricted domain on a very specific reasoning problem. But I think this opens the door to develop reasoning models for specific business / industry tasks with the right data sets."*

Peter Matra wrote:

*Thanks to DeepSeek, attention has shifted to a field that I find incredibly interesting and full of potential: optimizing Large Language Models.*

*Although the papers published by DeepSeek seem extremely promising, I won’t lie: I haven’t had time to run them through an LLM for a summary yet!*

*That said, I don’t think they’re the best starting point if you’re just beginning to explore this field.*

*Let me recommend one that is accessible, easy to read, and will help you start understanding the endless possibilities that model optimization offers.*

*It covers transformer-based models, including LLMs and Vision Transformers, focusing on addressing the high resource consumption during fine-tuning.*

*What problem are they trying to solve?*

*Fine-tuning pretrained models (like BERT or ViTs) can be out of reach for many companies due to the enormous memory and GPU requirements. Even smaller models may demand prohibitive resources.*

*What's the solution?*

*The approach divides tasks between CPU and GPU—which might seem obvious, but it hadn't occurred to me! It combines structured pruning and fine-tuning in an iterative process.*

*How does it work?*

*The CPU handles less intensive tasks, such as identifying which parts of the model can be pruned.*

*The GPU focuses on high-demand tasks like matrix multiplication during fine-tuning.*

*The result is an efficient process where pruning happens on the CPU, and fine-tuning occurs on the GPU.*

*Key results:*

*65-68% reduction in GPU memory usage for ViT models with 33% pruning, with only a minimal drop in accuracy (~3%).*

*Proven applications in Vision domains (CIFAR100, TinyImageNet, etc.) and NLU tasks (GLUE benchmark).*

*Why does it matter?*

*This not only optimizes resource usage but also makes fine-tuning large models accessible in resource-limited environments—like mine!*

Michael Erlihson wrote:

*Why the KL term in the GRPO optimization objective looks so strange?*

*In* [*my recent blog post*](https://aiwithmike.substack.com/p/an-unusual-look-of-the-kl-divergence)*, I explore an unconventional perspective on the Kullback-Leibler (KL) divergence, a cornerstone concept in machine learning, statistics, and information theory.*

*KL divergence typically measures the "distance" between two probability distributions P and Q*

*But beyond its familiar role in loss functions for an extremely wide spectrum of deep learning loss/reward function, this divergence hides surprising mathematical insights when viewed through a different lens.*

*In the DeepSeek paper it was used as a regularization term in their objective function, but why it looks that weird?*

*It is related to the properties of Bregman f-divergences...*

*An unusual look of the KL-Divergence Term in Deep Seek R1 Training Objective - all the details in* [*my blog*](https://aiwithmike.substack.com/p/an-unusual-look-of-the-kl-divergence)

A red line with black text

Description automatically generated

Philipp Schmidt wrote:

Inference-time Rejection Sampling with Reasoning Models! 🤔 could be an interesting approach to further scale performance and synthetic dataset generation. (Maybe how OpenAI o1-pro uses more “compute”?)

1 ) Generate K <think> samples in parallel.

2 ) Stop at <answer> before completing the full generation.

3 ) Score/Judge each thought process using a Reward Model or Judge.

4 ) Select the best reasoning path and continue the generation.

A diagram of a sample process

Description automatically generated

Juergen Schmidhuber wrote:

*DeepSeek [1] uses elements of the 2015 reinforcement learning prompt engineer [2] and its 2018 refinement [3] which collapses the RL machine and world model of [2] into a single net. This uses the neural net distillation procedure of 1991 [4]: a distilled chain of thought system.*

*REFERENCES (easy to find on the web):*

*[1] DeepSeekR1 (2025): Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv 2501.12948*

*[2] J. Schmidhuber (JS, 2015). On Learning to Think: Algorithmic Information Theory for Novel Combinations of Reinforcement Learning Controllers and Recurrent Neural World Models. arXiv 1210.0118. Sec. 5.3 describes the reinforcement learning (RL) prompt engineer which learns to actively and iteratively query its model for abstract reasoning and planning and decision making.*

*[3] JS (2018). One Big Net For Everything. arXiv 1802.08864. See also US patent US11853886B2. This paper collapses the reinforcement learner and the world model of [2] (e.g., a foundation model) into a single network, using the neural network distillation procedure of 1991 [4]. Essentially what's now called an RL "Chain of Thought" system, where subsequent improvements are continually distilled into a single net. See also [5].*

*[4] JS (1991). Learning complex, extended sequences using the principle of history compression. Neural Computation, 4(2):234-242, 1992. Based on TR FKI-148-91, TUM, 1991. First working deep learner based on a deep recurrent neural net hierarchy (with different self-organising time scales), overcoming the vanishing gradient problem through unsupervised pre-training (the P in CHatGPT) and predictive coding. Also: compressing or distilling a teacher net (the chunker) into a student net (the automatizer) that does not forget its old skills - such approaches are now widely used. See also [6].*

*[5] JS (AI Blog, 2020). 30-year anniversary of planning & reinforcement learning with recurrent world models and artificial curiosity (1990, introducing high-dimensional reward signals and the GAN principle). Contains summaries of [2][3] above.*

*[6] JS (AI Blog, 2021). 30-year anniversary: First very deep learning with unsupervised pre-training (1991) [4]. Unsupervised hierarchical predictive coding finds compact internal representations of sequential data to facilitate downstream learning. The hierarchy can be distilled [4] into a single deep neural network. 1993: solving problems of depth >1000.*

Philipp Schmidt wrote:

*SFT Memorizes, RL Generalizes. New Paper from Google DeepMind shows that Reinforcement Learning generalizes at cross-domain, while SFT primarily memorizes.*

*Experiments*

*1 ) Model & Tasks: Llama-3.2-Vision-11B; GeneralPoints (text/visual arithmetic game); V-IRL (real-world robot navigation)*

*2 ) Setup: SFT-only vs RL-only vs hybrid (SFT→RL) pipelines + RL variants: 1/3/5/10 verification iterations (”Reject Sampling”)*

*3 ) Metrics: In-distribution (ID) vs out-of-distribution (OOD) performance*

*4 ) Ablations: Applied RL directly to base Llama-3.2 without SFT initialization; Tested extreme SFT overfitting scenarios; Compared computational costs versus performance gains*

*Insights*

*1 ) Outcome-based rewards are key for effective RL training*

*2 ) SFT is necessary for RL training when the backbone model does not follow instructions*

*3 ) Multiple verification/Reject Sampling help improve generalization up to ~6%*

*4 ) Used Outcome-based/rule-based reward focusing on correctness*

*5 ) RL generalizes in rule-based tasks (text & visual), learning transferable principles.*

*6 ) SFT leads to memorization and struggles with out-of-distribution scenarios.*

Paper: [SFT Memorizes, RL Generalizes:](https://github.com/dimitarpg13/large_language_models/blob/main/articles/reinforcement_learning/SFT_Memorizes_RL_Generalizes-A_Comparative_Study_of_Foundation_Model_Post-training_SergeyLevine_2025.pdf)

[A Comparative Study of Foundation Model Post-training](https://github.com/dimitarpg13/large_language_models/blob/main/articles/reinforcement_learning/SFT_Memorizes_RL_Generalizes-A_Comparative_Study_of_Foundation_Model_Post-training_SergeyLevine_2025.pdf)

Github: <https://github.com/LeslieTrue/SFTvsRL>

Model & data: [SFTvsRL Models & Data huggingface repo](https://huggingface.co/collections/tianzhechu/sftvsrl-models-and-data-6797ba6de522c7de7fcb80ba)

# References

[1] [DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning, DeepSeek AI, 2025](https://github.com/dimitarpg13/large_language_models/blob/main/articles/reinforcement_learning/DeepSeek_R1-Incentivizing_Reasoning_Capability_in_LLMs_via_RL.pdf)

[2] [Kimi K1.5: Scaling Reinforcement Learning with LLMs, Technical Report of Kimi K1.5 , 2025](https://github.com/dimitarpg13/large_language_models/blob/main/articles/reinforcement_learning/Scaling_RL_with_LLM_Kimi_k1.5.pdf)

[3] [DeepSeek-Prover-V1.5: Harnessing Proof Assistant Feedback for Reinforcement Learning and Monte-Carlo Tree Search, H. Xin et al, DeepSeek, 2024](https://github.com/dimitarpg13/large_language_models/blob/main/articles/reinforcement_learning/DeepSeek-Prover-V1.5-Harnessing_Proof_Assistant_Feedback_for_Reinforcement_Learning_and_Monte-Carlo_Tree_Search.pdf)

[4] [DeepSeek Technical Report, 2024](ttps://github.com/dimitarpg13/large_language_models/blob/main/articles/DeepSeek-V3_Technical_Report.pdf)

[5] [DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models, Z. Shao et al, DeepSeek AI, 2024](https://github.com/dimitarpg13/large_language_models/blob/main/articles/reinforcement_learning/DeepSeekMath-Pushing_the_Limits_of_Mathematical_Reasoning_in_Open_Language_Models_Shao_2024.pdf)

[5] DeepSeek AI repo: <https://github.com/deepseek-ai/DeepSeek-R1>

[6] Moonshot AI / Kimi repo: <https://github.com/MoonshotAI>

[7] [Linkedin discussion 1, Jan 20th, Jim Fan](https://www.linkedin.com/posts/drjimfan_we-are-living-in-a-timeline-where-a-non-us-activity-7287125475280265217-dUT3?utm_source=share&utm_medium=member_desktop)

[8] [Linkedin discussion 2, Jan 20th, Jim Fan](https://www.linkedin.com/posts/drjimfan_that-a-second-paper-dropped-with-tons-of-activity-7287148978247290880-Gkd2?utm_source=share&utm_medium=member_desktop)

[9] [Linkedin discussion 3, Jan 23rd, Charles H Martin](https://www.linkedin.com/posts/charlesmartin14_revenge-of-the-quants-so-what-have-we-activity-7288268696253345792-zjQT)

[10] [Linkedin discussion 4, Jan 25th, Charles H Martin](https://www.linkedin.com/posts/charlesmartin14_thanks-to-deepseek-no-one-is-sleeping-activity-7288988183332077569-9G6O)

[11] [Linkedin discussion 5, Jan 25th, Yann Le Cunn](https://www.linkedin.com/posts/yann-lecun_to-people-who-see-the-performance-of-deepseek-activity-7288591087751884800-I3sN?utm_source=share&utm_medium=member_desktop)

[12] [Linkedin discussion 5, Jan 26th, Peter Gostev](https://www.linkedin.com/posts/peter-gostev_deepseeks-r1-model-release-caused-quite-activity-7289341329325604865-hlUS?utm_source=share&utm_medium=member_desktop)

[13] [Linkedin discussion 6, Jan 26th, Xu Ning](https://www.linkedin.com/posts/xning_deepseek-r1-incentivizing-reasoning-capability-activity-7289460243539890178-znll?utm_source=share&utm_medium=member_desktop)

[14] [Linkedin discussion 7, Jan 27th, Cameron Wolfe](https://www.linkedin.com/posts/cameron-r-wolfe-ph-d-04744a238_one-of-the-most-overlooked-contributing-factors-activity-7289651602108338176-7yTL/?utm_source=share&utm_medium=member_desktop)

[15] [Linkedin discussion 8, Jan 28th, Louis Scott](https://www.linkedin.com/posts/louisscott_a-few-thoughts-after-my-quick-read-of-the-activity-7289918932457566209-Z3Hj/?utm_source=share&utm_medium=member_desktop)

[16] [Linkedin discussion 9, Jan 29th, Niccolo Gentile](https://www.linkedin.com/posts/niccolo-gentile-phd-02208160_there-is-understandably-quite-a-lot-of-activity-7289951426699493376-Q1Ob?utm_source=share&utm_medium=member_desktop)

[17] [Linkedin discussion 10, Jan 31st, Charles H Martin](https://www.linkedin.com/posts/charlesmartin14_deepseek-weightwatcher-update-60-done-activity-7290587792559013889-nl0i/?utm_source=share&utm_medium=member_desktop)

[18] [Linkedin discussion 11: TinyZero a Minimalist Reinforcement Learning Framework for $30, Jan 31st, Charles H. Martin](https://www.linkedin.com/posts/charlesmartin14_breaking-news-researchers-recreated-deepseeks-activity-7290609408726716416-VVzz?utm_source=share&utm_medium=member_desktop)

[19] TinyZero, github repo, Jiayi-Pan: <https://github.com/Jiayi-Pan/TinyZero>

[20] [Linkedin discussion 12, Jan 31st, Pete Matra](https://www.linkedin.com/posts/pere-martra_transformer-pruning-optimization-activity-7290609315994824704-fk54?utm_source=share&utm_medium=member_desktop)

[21] [Linkedin discussion 13, Jan 31st, Michael Erlihson](https://www.linkedin.com/posts/michael-erlihson-8208616_kl-deeplearning-deepseek-activity-7290607529443663872-9NX8?utm_source=share&utm_medium=member_desktop)

[22] [Linkedin discussion 14, Jan 31st, Philipp Schmidt](https://www.linkedin.com/posts/philipp-schmid-a6a2bb196_inference-time-rejection-sampling-with-reasoning-activity-7291075220474384385-VGvm?utm_source=share&utm_medium=member_desktop)

[23] [An unusual look of the KL-Divergence Term in Deep Seek R1 Training Objective ?, Mike Erlihson, 2025](https://aiwithmike.substack.com/p/an-unusual-look-of-the-kl-divergence)

[24] [DeepSeek collapses the RL machine and world model into single net, Feb 1st, Juergen Schmidhuber, 2025](https://www.linkedin.com/posts/j%C3%BCrgen-schmidhuber-39226872_deepseekr1-activity-7291372258533539840-W3-0?utm_source=share&utm_medium=member_desktop)

[25] [Linkedin discussion 15: on the description of out-of-distribution data, Charles H. Martin, Feb 1st, 2025](https://www.linkedin.com/posts/charlesmartin14_talktochuck-theaiguy-activity-7291628740386689024-o21x?utm_source=share&utm_medium=member_desktop)

[26] [Linkedin discussion 16: SFT Memorizes, RL Generalizes, new paper from DeepMind, Philipp Schmidt, Feb 2nd, 2025](https://www.linkedin.com/posts/philipp-schmid-a6a2bb196_sft-memorizes-rl-generalizes-new-paper-activity-7291756584307679232-vVCx?utm_source=share&utm_medium=member_desktop)

[27] [SFT Memorizes, RL Generalizes: A Comparative Study of Foundation Model Post-training, T. Chu et al, paper with Sergey Levine, 2025](https://github.com/dimitarpg13/large_language_models/blob/main/articles/reinforcement_learning/SFT_Memorizes_RL_Generalizes-A_Comparative_Study_of_Foundation_Model_Post-training_SergeyLevine_2025.pdf)

[28] Running DeepSeek R1 via ollama: <https://ollama.com/library/deepseek-r1:14b>

[29] Fully Open Reproduction of DeepSeek R1: <https://github.com/huggingface/open-r1>

[30] [DeepSeek R1 Primer: Distilled AI](https://aman.ai/primers/ai/deepseek-R1/)

[31] [Fine Tune DeepSeek notebook](https://github.com/patchy631/ai-engineering-hub/blob/main/DeepSeek-finetuning/Fine_tune_DeepSeek.ipynb)

[32] [DeepSeek Series: Technical Overview, Shayan Mohanti, online article at martinfowler.com, 2025](https://martinfowler.com/articles/deepseek-papers.html)