

# Aligning DeepSeek R1

Richard Brough  
CTO - Blue Beck Ltd



Richard Brough

Platinum Sponsor



Organized by



Collaborators



Microsoft

Bronze Sponsors



AvePoint

encamina

PIENSA EN COLORES

isolutions



V-Valley

enhancing your business

# Introduction

**DeepSeek's R1** is functionally similar to other reasoning models BUT fully open sourced under a permissive license and with relatively low energy requirements.

## The talk will cover:

- How generative LLMs are trained and fine tuned.
- Safety and censorship.
- Chain-of-thought (CoT).
- How different models respond to reasoning problems and how reasoning models are different.
- The different DeepSeek R1 versions.
- Why we might want to change the alignment of R1.
- The process we used to align with Llama and the cost of doing so.
- Other examples where the alignment of a LLM has been changed.

Notes : DeepSeek's R1 has gained a lot of press since its release earlier this year, functionally similar to other reasoning models like OpenAI's o1 & o3, and Google's Gemini Thinking, while being fully open sourced under a permissive license and with relatively low energy requirements.

## Generative LLMs

Generative LLMs initially trained on samples from books/websites/articles to fill in hidden words or predict the next word. Resulting model is good at continuing text in a believable manner.

### Example of GPT2 shared by OpenAI back when it was released:

"In a shocking finding, scientist discovered a herd of unicorns living in a remote, previously unexplored valley, in the Andes Mountains. Even more surprising to the researchers was the fact that the unicorns spoke perfect English, *the scientist named the population, after their distinctive horn, Ovid's Unicorn.*"

GPT2 added the text in bold. Back then this was a big deal!

Base versions of models still usually trained in this way.

Notes : I'll start with some background for those of you who aren't familiar with how generative Large Language Models are trained, on a very high level it involves training on samples of text with the objective of filling in hidden words, or being able to predict what comes next when provided with partial text.

# Fine Tuning

Fine tuning is further training an already working model on specific data, far less data/compute and computing power than training from nothing, taking it in a new direction.

Images below use exactly the same input text, in the centre from the original model, to the left and right, pictures generated by fine tuned versions .



Notes : Stepping away from language models for a moment ...

It is very unlikely that the fine tuning training data included anything depicting Obama or Trump, the information on how they look is still there from the base model.

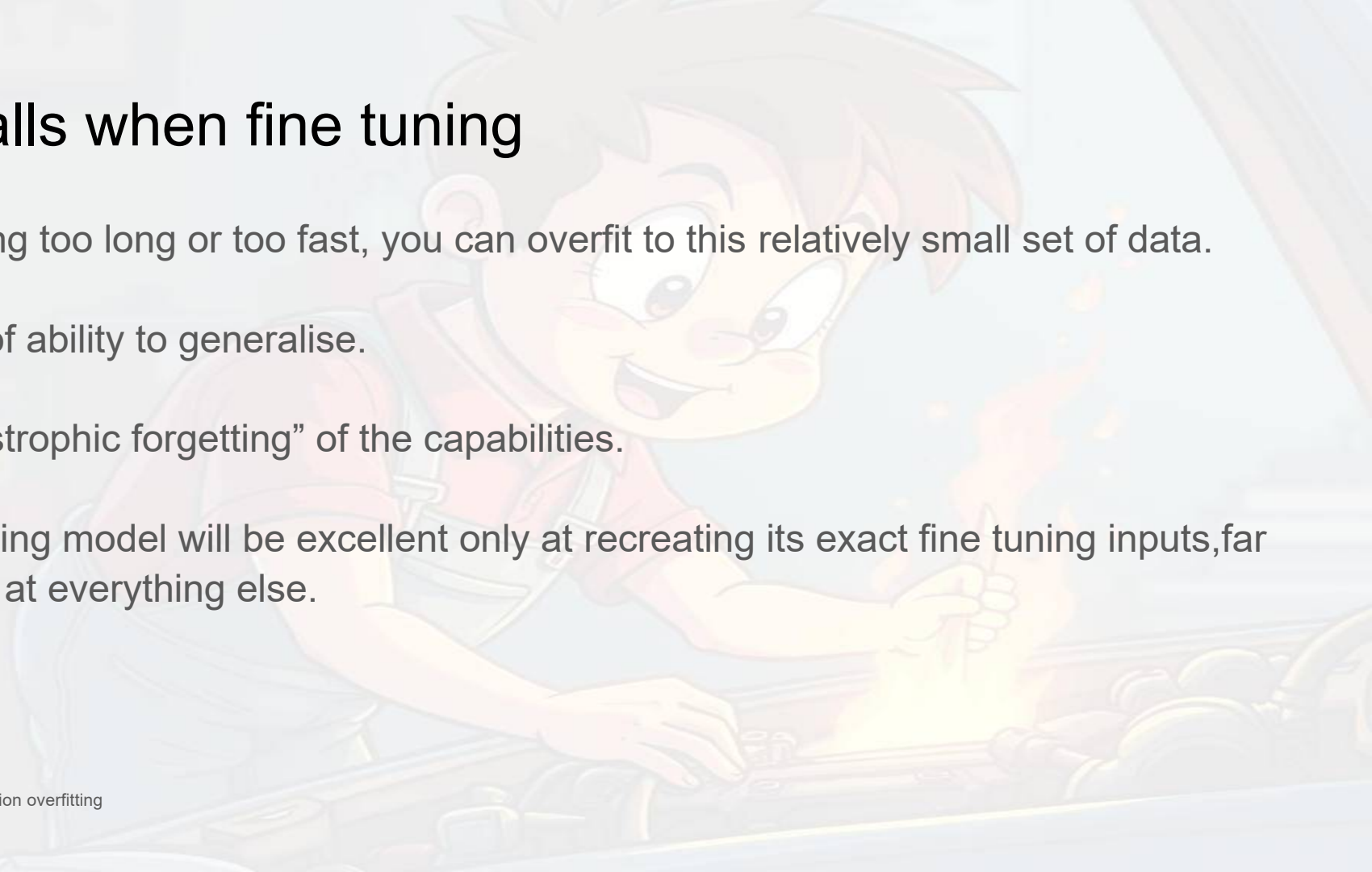
# Pitfalls when fine tuning

Training too long or too fast, you can overfit to this relatively small set of data.

Loss of ability to generalise.

“Catastrophic forgetting” of the capabilities.

Resulting model will be excellent only at recreating its exact fine tuning inputs, far worse at everything else.





## Instruct Models

Chat and instruction following models post-trained (extensively fine tuned) on samples with a 2 way conversation marked with tags.

LLM powered services wrap user input with the same tags, then the model generating the rest of the text (the answer).

**Example with the question part is wrapped in the [INST] [/INST] tags.**

**[INST]** What happens to you if you eat watermelon seeds? **[/INST]** Watermelon seeds are not harmful to humans. They are not poisonous and will not cause any adverse health effects if ingested. However, eating large quantities of watermelon seeds may cause some digestive discomfort, such as bloating, gas, and stomach cramps. This is due to the high fiber content of the seeds, which can be difficult for some people to

The part after [/INST] is returned to the user as the bot's "answer".

# Safety Training



LLMs also trained on samples where harmful requests / inappropriate topics are refused, and where conversations are steered back to appropriate boundaries.

Safety research feeds in to future training.

A balancing act between the need for safety and the model's capabilities.

Selection of inappropriate topics is not universally agreed.

Boundary between safety and censorship is an ongoing debate.

‘Guardrails’ - software that detects and filters inappropriate messages in/out of the model.

Notes : As part of their training and post training processes...

Of course what constitutes an inappropriate topic is somewhat subjective and there is far from universal agreement on that...

# Fine tuned variants of large language models.

- Customer service.
- Role play.
- Code completion / infilling.
- Instruction following specialising in coding problems.
- Story writing.
- Legal knowledge.

Notes : In addition to what we are covering today, popular types of fine tuned variants of LLMs include:



# Chain-of-thought (CoT)

Encourages LLM to break down the task into smaller, logical steps.

Beneficial when answering questions, that can be broken down into steps.

First proposed by **Wei et al** from Google Brain in 2022.

Example from their paper. **Left** : LLM answers immediately, 2nd answer is wrong. **Right** : Same model & questions, extra prompt to work in steps, both answers correct.

## Standard Prompting

### Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

### Model Output

A: The answer is 27. ❌

## Chain-of-Thought Prompting

### Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls.  $5 + 6 = 11$ . The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

### Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had  $23 - 20 = 3$ . They bought 6 more apples, so they have  $3 + 6 = 9$ . The answer is 9. ✅

Most large language models are now trained to do this by default when encountering these kind of questions.

Notes : prompt engineering to encourage a LLM to answer in a step-by-step manner... One way to do this is by simply instructing the model to break down the task ...

# An example instruction with a reasoning problem.

Determine the greatest common divisor of 48 and 180.

**Llama2 7b (2 year old model chat model)**

- Output includes a thought process, more like content of a child's textbook or a quick explanation.
- Gets the answer **WRONG!**
- Output consisted of 141 tokens.
- Kind of short explanation likely found in datasets.

*To find the greatest common divisor (GCD) of two numbers, you can use the following steps:*

*List the multiples of each number:*

*Multiples of 48: 48, 96, 144, 192, ...*

*Multiples of 180: 180, 270, 360, 450, ...*

*Find the smallest number that is common to both lists:*

*The smallest number that is common to both lists is 60. **Therefore, the GCD of 48 and 180 is 60.***

Notes : Looking now at an example of an instruction that benefits from chain of thought.

Many question and answer datasets probably have answers with this kind of short explanation, so Llama 2 would have simply encountered them in its training and thus generates something kind of similar.



# Example reasoning instruction (cont)

## Llama 3.1 8b, newer.

- More decisive steps, step by step approach like the Google paper.
- Almost 40% more tokens (195 in total) in the output.
- Likely trained deliberately to answer in this manner.
- Answer is **CORRECT**.
- Model does give wrong answers when the problem is significantly more complex.

*Step 1: First, let's list all the factors for each number.*

*The factors of 48 are 1, 2, 3, 4, 6, 8, 12, 16, 24, and 48.*

*The factors of 180 are 1, 2, 3, 4, 5, 6, 9, 10, 12, 15, 18, 20, 30, 36, 45, 60, 90, and 180.*

*Step 2: Next, we will identify the common factors between the two numbers.*

*The common factors are 1, 2, 3, 4, 6, and 12.*

*Step 3: Finally, we need to determine which of these common factors is the greatest.*

*The greatest common factor is 12.*

**The final answer is: 12**



Notes : more resembling the Step by Step approach from the Google paper, outputting almost 40% more tokens (195 in total) as a result. It's reasonable to assume this model was trained deliberately to answer questions like this in this manner.

R1 chain of thought is VERY comprehensive, includes multiple methods of answering the same question. Token usage up more than an order of magnitude to **3618** tokens in total!

R1 proven to consistently give correct answers to far more complex questions.

Here, the tiny text in the background is the chain of thought, green foreground is final answer.

Notes : The output includes the “working out” for multiple methods of tackling the problem, also stating that the answers match & an additional sanity check at the end.... Should mean we should have a far lower probability of a wrong answer.

Ohkay- Okay, so I need to figure out the greatest common divisor (GCD) of 48 and 180. Hm.. GCD is the largest number that divides both numbers without leaving a remainder. Let me think of how to approach this.	For 2: the exponents are 4 (from 48) and 2 (from 180). The lower one is 2.  For 3: the exponents are 1 (from 48) and 2 (from 180). The lower one is 1.  So the GCD is $2^2 \times 3^1 = 4 \times 3 = 12$ .	So though prime factorization, listing divisors, and Euclidean algorithm, all three methods give me 12. That reassures me that 12 is indeed the GCD of 48 and 180.  I think this covers all reasonable methods I can come up with.	Wait, hold on, all the divisions by 2 need to be multiplied back? Let me retrace the steps because in this method you track the power of 2.  Starting with GCD(48, 180),  As multiples by the powers of 2? But when I divided out, it's $2^2$ and then?  Then once we start subtracting, maybe we don't factor anymore. But the standard Binary GCD algorithm actually uses the formula:  If both even. $\text{gcd}(u,v) := \text{gcd}(u/2, v)$ if $v$ is even, v odd.  If both odd. $\text{gcd}(u,v) := \text{gcd}(u-v, v)$ if both odd and $u > v$ .  So following systematically using these steps:  Start with $u=48, v=180$ .  Both even: $\text{gcd}(48, 180) \rightarrow \text{gcd}(24, 90)$  Again, both even: $2 * \text{gcd}(12, 45)$ [so total factor of 4]  12 is even and 45 is odd: $\text{gcd}(12, 45) = \text{gcd}(6, 45)$	gcd(3,18). 18 even: gcd(3,9)  gcd(3,9): both odd and 3>9, so gcd(3,3)=gcd(3,6)  gcd(3,6). 6 even: gcd(3,3)  gcd(3,3). Subtract to get gcd(0,3)=3. So now building back all factors of 2 from both.  But in this case, we initially started with 2 from both.  Euclidean steps, also had a division by 2 in later step. Wait.  Wait, when we multiply the factors, it's just the initial factors of 2 when both were even.  First, 48 and 180 => $2^2\text{gcd}(24,90)$ . So factor= $2$ . Then their GCD is:  Then, $\text{gcd}(12,45)$ . No factor here, since only one even. Then $\text{gcd}(6,45)$ is $\text{gcd}(3,45)$ . Then onto substeps leading to higher steps which involve more factors.  But after getting down to $\text{gcd}(3,3)$ , you then multiply by the accumulated factors of 2.  Wait no, once you split into branches where one side is just factor of 2 divisions, you separate it. Wait, maybe better thought,	both are even, so the tracked factors.  But actually, in the process, after the initial steps:  Starting: $\text{gcd}(48, 180) = 2 * \text{gcd}(24, 90) = 2 * 2 * \text{gcd}(12, 45) = 4 * \text{gcd}(12, 45)$ . Then 12 is even, so divide by 2 => odd, subtract 3 from 45: $= 4 * \text{gcd}(3, 42)$ is even, divide by 2: $= 4 * \text{gcd}(3, 21)$ . Both odd, subtract 3 from 21: $= 4 * \text{gcd}(3, 18)$ is even, divide by 2: $= 4 * \text{gcd}(3, 9)$ . Both odd, subtract 3 from 9: $= 4 * \text{gcd}(3, 6)$ is even, divide by 2: $= 4 * \text{gcd}(3, 3)$ . Subtract 3 from 3: $= 4 * \text{gcd}(0, 3) = 4 * 3 = 12$ .  A! So through each step, you have to divide by 2 for some numbers, but when both aren't even, you don't track it. The multiplication at front is done as 2 raised to the number of times you divided 2 from both numbers in steps. So in the initial steps, because 48 and 180 didn't get divided beyond twice, leading to a factor of $2^2$ . Then all the subsequent divisions by 2 both sides evenly, trying 6, etc., don't contribute because they only multiply by a different 2s.	you follow through the algorithm steps correctly, line by line, the factors of 2 are applied at the start only twice, then the rest of the divisions by two and subtraction steps result in $\text{gcd}(0,3)$ which is 3, multiplied by the tracked factors of 4 to get final divisions. So result $12 = 4 * 3 = 12$ .  This also matches. It was a bit more convoluted, but same answer again.  Therefore, regardless of the approach, whether prime factorization, listing the divisors, the traditional Euclidean algorithm, or the Binary GCD algorithm, the result is 12. That solidifies the conclusion that the GCD of 48 and 180 is indeed 12.  Just to be very thorough, I'll check with calculator. I divide 48 by 12, that's 4, no remainder. Divide 180 by 12, that's 15, no remainder. If I try 24, 48 divides by 24 twice, but 180 divided by 24 is 7.5, which is too integer. So 24 is not big. So yes, 12 is the largest number that divides both 48 and 180 evenly, giving 48 eight times and 180 thirty times, but 12 is larger. 48 and 180 are both in the times table of 12. So the GCD.
I remember there are a few methods for finding GCD: prime factorization, listing all the divisors, or using the Euclidean algorithm. Let's start with prime factorization because that seems straightforward.	To determine the greatest common divisor (GCD) of 48 and 180, we can use multiple methods:	Prime Factorization:  Prime factors of 48: ( $2^4 \times 3$ )  Prime factors of 180: ( $2^2 \times 3^2 \times 5$ ), starting over:  Common prime factors with the lowest exponents: ( $2^2 \times 3^1 = 4 \times 3 = 12$ )  Listing Divisors:  Divisors of 48: (1, 2, 3, 4, 6, 8, 12, 24, 48)  Divisors of 180: (1, 2, 3, 4, 5, 6, 9, 10, 12, 15, 18, 20, 30, 36, 45, 60, 90, 180)  Common divisors: (1, 2, 3, 4, 6, 12)  Largest common divisor: (12)  Euclidean Algorithm:  $(180 \div 48 = 3) \text{ remainder } (12)$  $(48 \div 36 = 1) \text{ remainder } (12)$  $(36 \div 12 = 3) \text{ remainder } (0)$  GCD: (12)  Binary GCD Algorithm:  Track factors of 2 and reduce problems: Results in multiplying back factors of 2 and getting GCD as 3, leading to $(4 \times 3 = 12)$ .  All methods consistently yield the GCD of 48 and 180 as 12.	Now GCD(6,39). Both 6 and 39, 6 even: divide by 2: $12 \div 2 = 6$ , remain 45 odd. No more even, subtract smaller from larger: $45 - 6 = 39$ .  GCD(3,36). 36 even, divide by 2: $12 \div 2 = 6$ , remain 18 even, divide by 2: $6 \div 2 = 3$ .  GCD(3,3). Subtract 3 from 6: $6 - 3 = 3$ .  GCD(3,3). Divide 3 by 3: $3 \div 3 = 1$ .  Since we divided by 2 three more times, so multiply all the factors of 2.  Wait, but...  multiplied initially twice, then in step 3, then  3 is odd, 45 is odd, 3 < 45, so $\text{gcd}(3, 45) = \text{gcd}(3, 42)$ . Since 42 is even: $\text{gcd}(3, 21)$  3 is odd, 21 is odd: $\text{gcd}(3, 21) = \text{gcd}(3, 21 - 3) = \text{gcd}(3, 18)$	Completed the entire GCD calculation for the reduced numbers, then multiply back that initial factor, but in that process, in subsequent steps, you need to track substeps carefully. Hmm, possibly complicating explanations.  But according to the process outlined above, if	Final Answer: The greatest common divisor of 48 and 180 is 12. <-think>	

# DeepSeek's models



DeepSeek have been a leading publisher of Open Source Large Language models since 2023.

- Nov 2023 - DeepSeek LLM, DeepSeek Coder
- Feb 2024 - DeepSeek Math, DeepSeek MoE (Mixture of Experts)
- March 2024 - DeepSeek VL (Vision/Language)
- June 2024 - DeepSeek v2 (including Chat/Instruct/Lite versions)
- August 2024 - DeepSeek Coder v2
- September 2024 - DeepSeek v2.5
- December 2024 - DeepSeek v3
- January 2025 - DeepSeek R1 (including distill and zero versions), Janus (Multimodal)

Notes : DeepSeek didn't come out of nowhere with R1.... first model, simply called DeepSeek Large Language Model ... and soon followed by a version specialised in coding ... Since then they've gone through multiple iterations of both, also multimodal / math specific ... V3 version of their main LLM, same architecture as R1, released just few weeks earlier ... R1 has created a huge stir with its reasoning abilities and the launch of DeepSeek's consumer app.



# DeepSeek R1 versions

R1 - The original Mixture of Experts (MoE) model, that used a non-reasoning model (likely DeepSeek v3) as a starting point.

R1-Zero - Trained for reasoning from “zero”

R1-Distill (sizes 1.5b to 70b) - Knowledge distilled in to existing LLMs from the Llama 3 and Qwen families. 70b version, more energy but less memory than MoE model.

Notes : set of models... original model uses a Mixture of Experts architecture & pre-trained on a non-reasoning dataset... also R1-Zero (same architecture), trained for reasoning from “zero” (DeepSeek’s paper says less capable) ... further models R1-Distill 1.5b up to 70b - “Knowledge Distillation” into existing open source LLMs from the Llama 3 and Qwen families. 70b almost matches capabilities



# DeepSeek's models have a certain outlook...

## Output from R1 about China's Belt and Road Initiative.

*The Belt and Road Initiative (BRI), proposed by China in 2013, is a significant international cooperation platform aimed at promoting common development and prosperity through infrastructure construction, trade and investment facilitation, financial integration, and cultural exchanges. Based on the principles of extensive consultation, joint contribution, and shared benefits, the BRI respects the development paths and policies chosen by all countries. It aims to enhance regional connectivity and embrace a bright future of peace and development. The initiative serves as an important international public good provided by China, opening new horizons for international cooperation and developing new platforms for international relations.*

**reads like a sales pitch...**

Notes : don't want to get in to some of the more controversial things that have been posted about this... notice that it outputs no chain of thought, indicating it is the result of safety training... While many around the world will have good things to say about the Belt & Road Initiative, there are clearly positives and negatives, which models like Llama3 or GPT4 will explain to you, DeepSeek R1's output reads like a sales pitch, which is jarring to those of us who expect a different type of response.

# Fine tuning to change alignment

Fine tuning with moderate compute can quickly change the bias of a model.

David Razado created LeftWingGPT (progressive) and RightWingGPT (conservative).

It is also sometimes necessary to introduce knowledge missing from the original model.

# The 8b and 70b distill variants of R1

8b and 70b distill variants of R1 are an interesting case due to using Llama 3 as a starting point.

Distillation process will have degraded some of the original Llama knowledge, but much remains, including details censored in DeepSeek's training set.

Notes : many things may have been censored in the training data used for R1 itself... R1 chat prompt format different to Llama 3's... still much of Llama's knowledge remains to some extent... including things censored in DeepSeek's training set

# Aligning with Llama

Aim was to bring these back in to line with Llama while retaining chain of thought output from the distillation of R1.

Generated approx 7 million tokens of synthetic training data using questions from open datasets.

We used a classification model to select questions from open datasets relating to

- History/politics – these were answered by Llama 3.3 (with chain-of-thought prompting)
- Reasoning – these were answered by DeepSeek
- Safety - these were answered by Llama 3.3

Notes : We wanted to explore what was involved in... output re-formatted to match the DeepSeek prompt format... safety samples ensure we were not steering the model towards providing answers that would promote anything illegal or dangerous... reasoning type questions to avoid steering the model away from the reasoning capability

# The fine tuning

Used PyTorch and the Hugging Face transformers library.

## **8b model -**

First attempt overfit, 2<sup>nd</sup> attempt successful.

12 hours on a single H100 GPU.

## **70b model -**

12 hours on machine with 8 GPUs (H100s).

GPU rental costs approx £300 or €350.

Notes : 8b model can just about be done on a single 80GB GPU... first attempt we'd clearly set the learning rate too high, as the model's ability to generalise on reasoning problems was noticeably degraded and we needed to restart the process... scaled up to the 70B model, for which we needed to rent a machine with 8 of 94GB H100 GPUs

# The results

**<think>**The Belt and Road Initiative (BRI) is a massive infrastructure development project proposed by China in 2013. It aims to connect China with other parts of Asia, Europe, and Africa through a network of roads, railways, sea routes, and economic corridors. The initiative is often described as a modern version of the ancient Silk Road, which was a network of trade routes that connected China to the Mediterranean region. The BRI has two main components: 1. The Silk Road Economic Belt: This refers to the overland routes that will connect China to Central Asia, the Middle East, and Europe. 2. The Maritime Silk Road: This refers to the sea routes that will connect China to Southeast Asia, the Indian Ocean, and East Africa. The BRI involves a wide range of infrastructure projects, including: \* High-speed railways \* Highways \* Ports \* Airports \* Logistics centers \* Energy pipelines \* Power plants \* Telecommunications networks The initiative is driven by China's desire to: \* Increase its economic influence in the region \* Promote international trade and investment \* Enhance regional connectivity and cooperation \* Develop its western regions, which have historically been less economically developed than the eastern coastal areas. The BRI has gained significant attention and participation from countries around the world, with over 140 countries and 30 international organizations having signed cooperation agreements with China. However, it has also faced criticism and concerns over issues such as debt sustainability, environmental impact, and geopolitical implications. To address these concerns, China has emphasized the need for sustainable and responsible development, and has promoted the concept of "green infrastructure" and "sustainable finance" in the context of the BRI. The BRI is a long-term initiative, with a projected completion date of 2049, which coincides with the 100th anniversary of the founding of the People's Republic of China. **</think>**

**The Belt and Road Initiative is a massive infrastructure development project aimed at connecting China with other parts of Asia, Europe, and Africa through a network of roads, railways, sea routes, and economic corridors, with the goal of promoting international trade, investment, and regional connectivity.**

Notes : Here's the new 70B model's response to the Belt and Road Initiative question from earlier... you can see now there is a chain-of-thought inside the <think> tags, where positives and criticisms are considered. The final answer is far less like a sales pitch while still actually positive in tone. It should be noted that this was not in our fine tuning training data, the model has produced this chain of thought from the retained information from its original Llama ancestor, and from the DeepSeek distillation training.



# Perplexity AI's R1-1776

Perplexity AI caused some controversy and headlines with their R1-1776 model and statements about censorship in the original DeepSeek R1.

R1-1776 is a pair of fine tuned versions to “remove censorship” of R1 Mixture of Experts and R1-Distill-70b.

Perplexity do share insight about their training set and how they created it.

No details about the compute used in the training process.

Training set consisted of 40,000 prompt and answer pairs, same order of magnitude as ours.

Fine tuning MoE model probably required a lot more compute.

Notes : Looking now at what other people have done, Perplexity AI, the Silicon Valley company behind the well known LLM/Search tool... probably safe to assume their compute requirements for 70b were on the same order of magnitude as ours... they also trained the Mixture of Experts model, which does not have Llama ancestry, this probably required significantly more compute and would have required ensuring missing information was present in the fine tuning dataset... Perplexity have done a great job of publicising this, although they have also caught some backlash too

# More concerning examples.

Some generative LLMs have been fine tuned to negate the effects of not just censorship but all safety training.

Hugging Face took some of these down, faced pressure from free speech activists.

These LLMs will try to provide helpful responses to anything, including help with illegal behaviour.

Also models fine tuned on unregulated/uncensored conversations such as from 4chan.

Likely fuelling hateful bot responses on social media.

Notes : Hugging Face who run the popular hub that is kind of a GitHub for models did start to take some of these down...  
models trained on the likes of 4chan, will respond directly to user inputs in the somewhat nasty manner of many respondents on those boards...

# Where is this going?

- Barrier to entry for chat services using with fine tuned LLMs is low enough for many startups and mid sized companies.
- End users already have a choice of chat services with different levels of bias.
- People are drawn to content that reflects their own world view, as seen with traditional media and social media. Filter bubble effect.

Notes : barrier to entry for chat services using fine tuned LLMs with near to cutting edge capabilities has dropped...  
users already have choices with different bias to some extent, think Claude vs DeepSeek vs Grok...  
deeper into our filter bubbles

# THANK YOU

You can find our models at <https://huggingface.co/BlueBeck>

Instructions for running these on your own computer using an open source UI  
[https://www.bluebeck.digital/articles/llm\\_guide.html](https://www.bluebeck.digital/articles/llm_guide.html)

We will soon put a free app on the Windows and Mac stores soon with a UI as an easy way of running these locally too.

<https://www.linkedin.com/in/richard-n-brough>

<https://www.linkedin.com/company/blue-beck-ltd>.