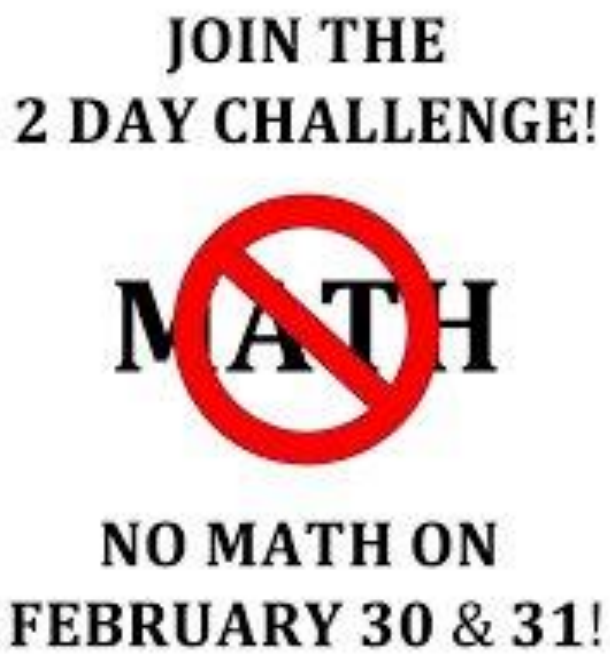
AI: What does it all mean? How does it work?

Alper Celik Ph.D.

Centre for Computational Medicine (CCM)

Discussion outline:



- What is machine learning?
- What are neural networks?
- How do neural networks learn?
- What are large language models?
  - How are they trained?
  - Where do they store information?
  - Why do they hallucinate?
  - Ways to minimize hallucinations.
- Can I have my own LLM?
- Agents, MCP and 007
- Computer vision
  - Vision transformers
  - Convolutional neural networks
- Multi-modal models
- How does this apply to health data
  - Language models, vision models, multi-modal, reasoning
  - Where does the data come from? Anonymization
  - Adversarial attacks, prompt injections
  - Isolating your environment
  - Resource constraints
- What is a CPU, GPU, TPU
- Conclusions

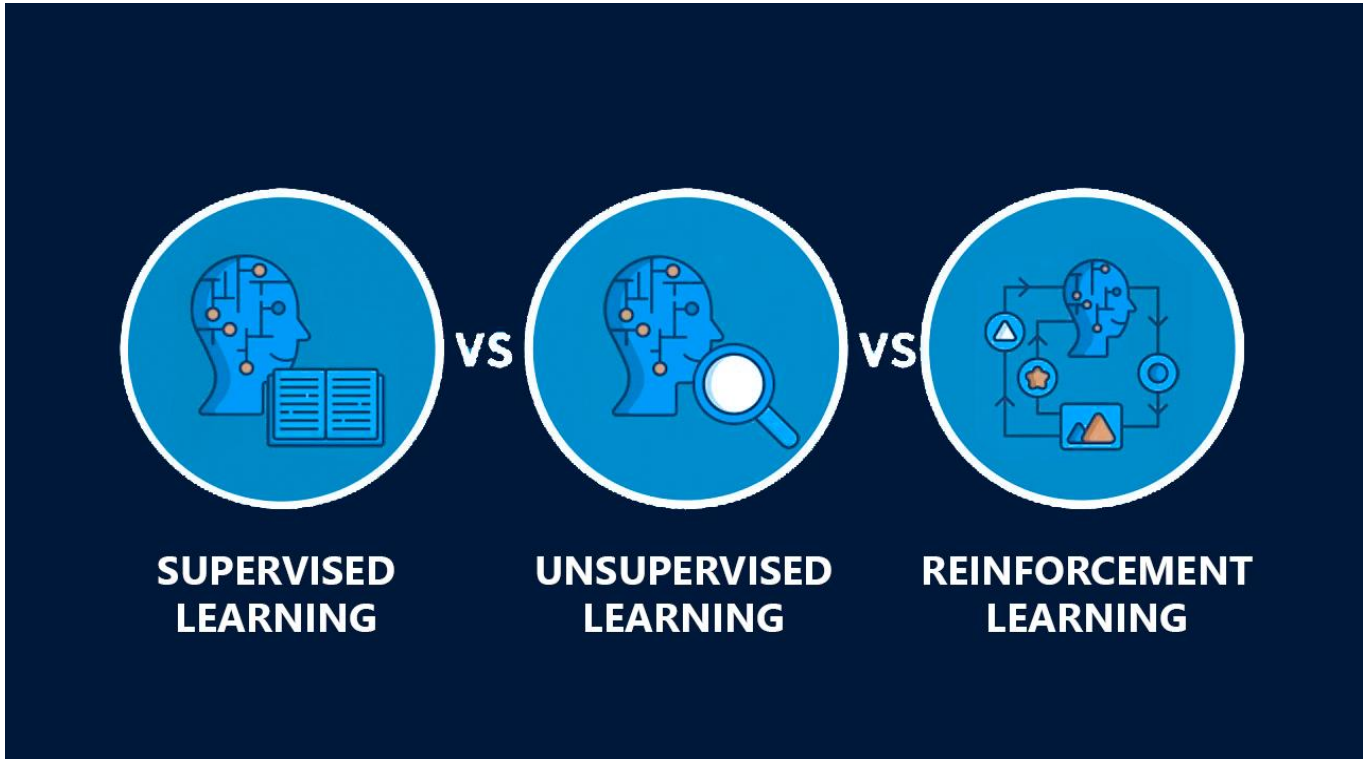
Machine Learning

- Simpler than it sounds
- The models either,
  - Find combinations of known variables
  - Generate new variables

To:

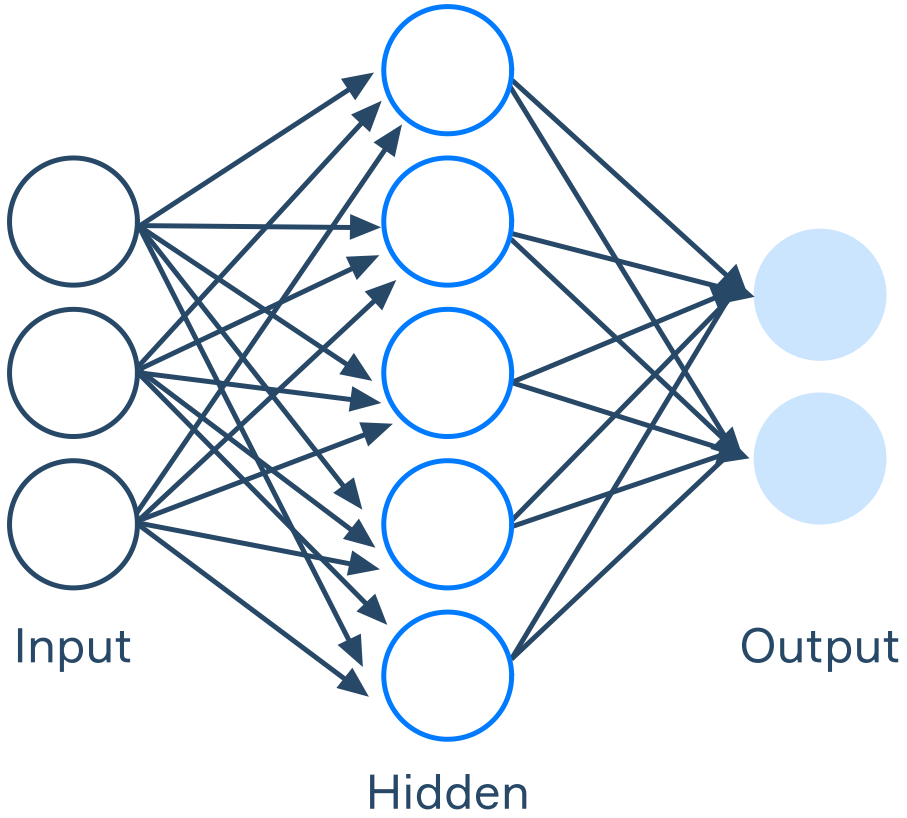
- predict an outcome
- reduce variability
- minimize undesirable decisions

- All machine learning models are basically a combination of numbers and a pre-defined set of mathematical operations that generate another numerical output
- There are 3 main branches of machine learning
  - Supervised
  - Unsupervised
  - Reinforcement



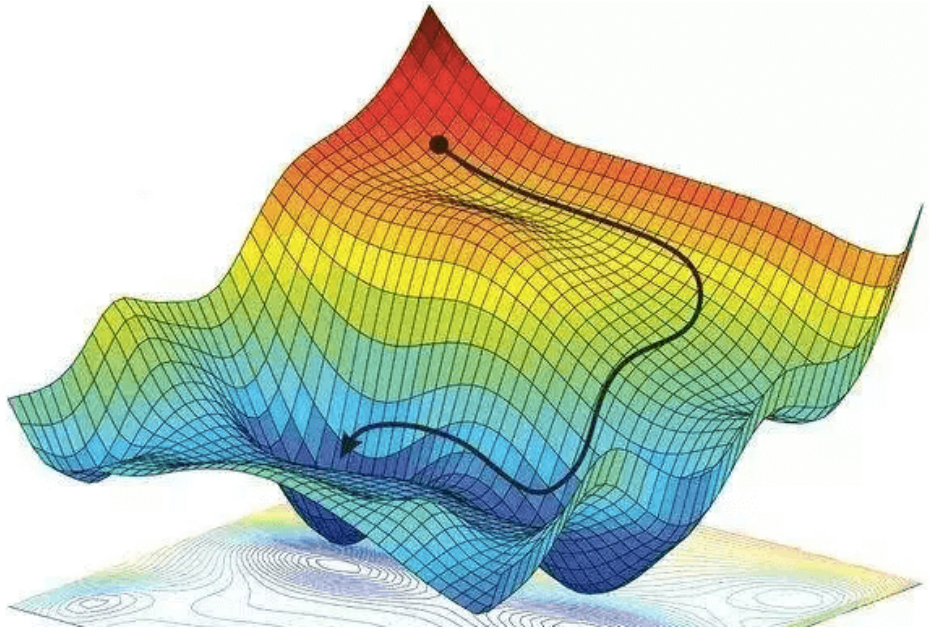
Neural networks, definitions

- Neural networks are a kind of **supervised** machine learning method
- They were developed in 1960s but we did not have the computational power to make them useful until now
- They are what's called "universal approximators", they can approximate any function (given enough data and large enough model)
- The come in many shapes and sizes that are useful for specific tasks
- They have hidden layers where the data is represented as abstractions and latent variables
- The connectivity and the mathematical operations that are performed at each layer determines what kind of information is processed. This is called the architecture of the model.
- Different architectures are used for different purposes because the data they deal with requires different operations.



Neural networks, how do they learn?

- This part is very similar to any other machine learning model
- We have a loss function (penalty for being wrong) and we want to find the minimum
- We start randomly (our big number matrices) and we feed examples and calculate loss, we then adjust the numbers to decrease this value (backpropagation)
- We need to make sure that the neurons fire when they need to (activation function)
- Doing this blindly will take a long time or a lot of computational resources (usually both) so we use a separate function to speed up loss minimization (optimizer)
- We do this over and over until things stop improving
- The most important part of any training step is good data, a lot of data and good train/test split



LLMs (should I say transformers?)

- Before transformers we used recurrent networks to deal with text.
- RNNs calculated hidden states one at a time, this was computationally expensive and resulted in model "forgetting the details" as time went on.
- Large language models are always based on transformer architecture
- The consider the entire text as a whole and therefore are aware of the "context"
- The same architecture can be applied to images and videos as well
- They are made out of "Transformer blocks"

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkoreit* Google Research usz@google.com

Llion Jones* Google Research llion@google.com

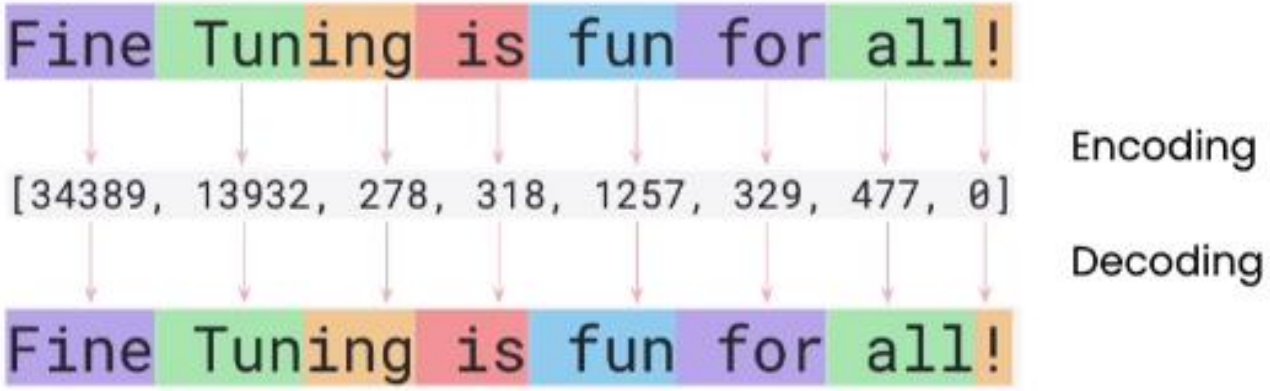
Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

Łukasz Kaiser* Google Brain lukaszkaiser@google.com

Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Getting the data ready, tokenization

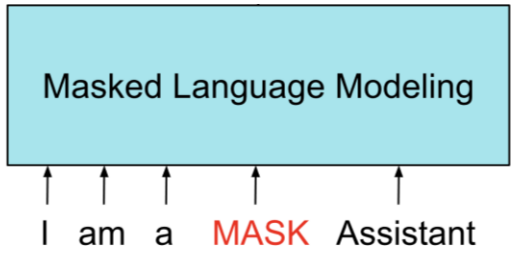
- Neural networks do not work with text or images, they work with numbers, so we need a way to convert text to numbers, this is called tokenization
- Each word (or sub-word) is converted to an integer from a "dictionary"
- These dictionaries can be language specific or multi-language
- Smaller models benefit from smaller dictionaries (single language)
- They are not created by hand but with automation scripts
- There are special tokens to indicate missing, blank, unknown data as well as start and end of a text



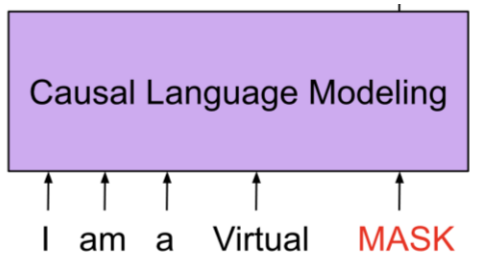
Supervised learning (kind of)

There are 2 kinds of self-supervised training, choice depends on what you want your model to do

- Masked language modelling randomly hides portions of the text for the model to fill in
- This is useful for learning context and understanding text
- This method is usually used for encoder models during pre-training

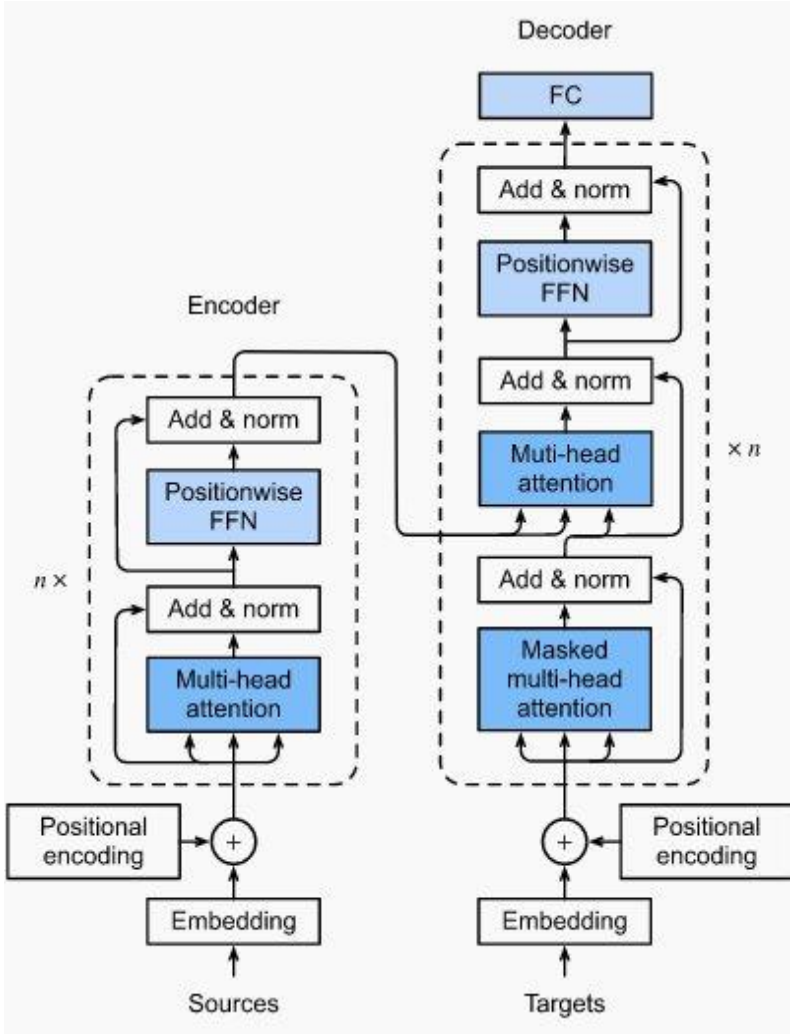


- Next token prediction is for predicting the next token
- For a given chunk of text (after a certain input) the next token is predicted
- This is useful for generative/decoder models (causal language modeling)



Generative vs predictive (pick the right tool)

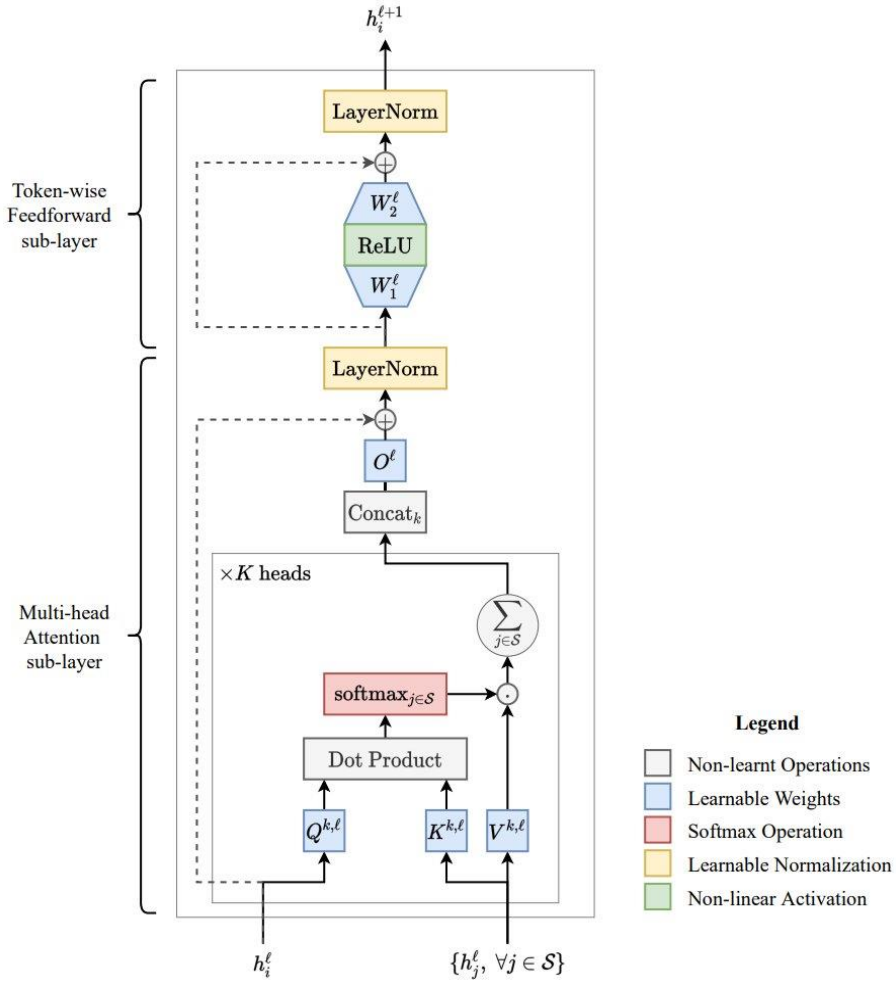
- Encoder models "encode" the text in a way that the model can understand
- They are not good at generative tasks
- Usually trained using MLM
- They "understand" the basics of the text
- They are good for regression/classification tasks where the general understanding is important
- Their embeddings can be used independently for downstream tasks
- BERT family of models are the most famous examples



- Decoder models can come up with new tokens given existing ones
- These form the bulk of generative tasks (GPTs are fully decoder models)
- They are usually trained using next token prediction
- Given some text they pick among the most likely words that follow and do it again and again until certain length is reached or nothing meaningful can follow

They are more than just token generators, they store information

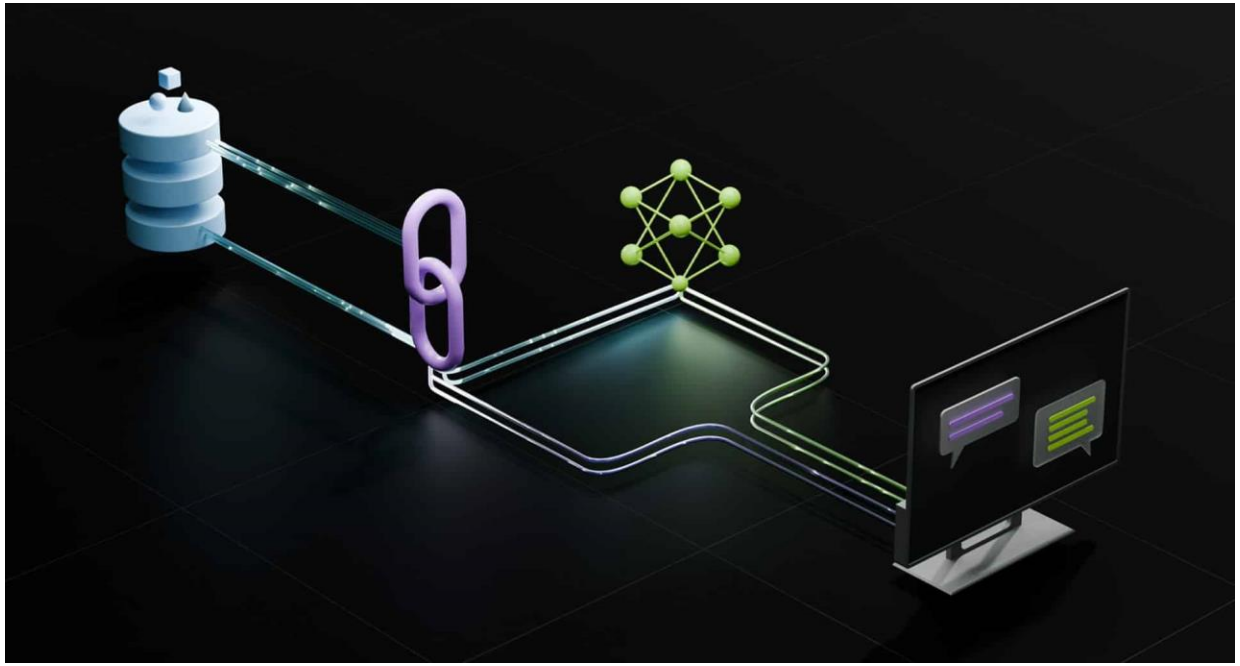
- Between each transformer block there is a normalization layer that transforms the data into a shape that is more amenable to the next transformer block
- Interestingly, these layer norm MLPs do a lot more than layer normalization
- These are the main "information storage" sections of the model.
- The information is stored as a combination of numbers that are "activated" based on what comes before. This "understanding" of what's coming next of the text as a whole is called "embeddings"
- We can do math on the embeddings (more on that later)



Why do they hallucinate?

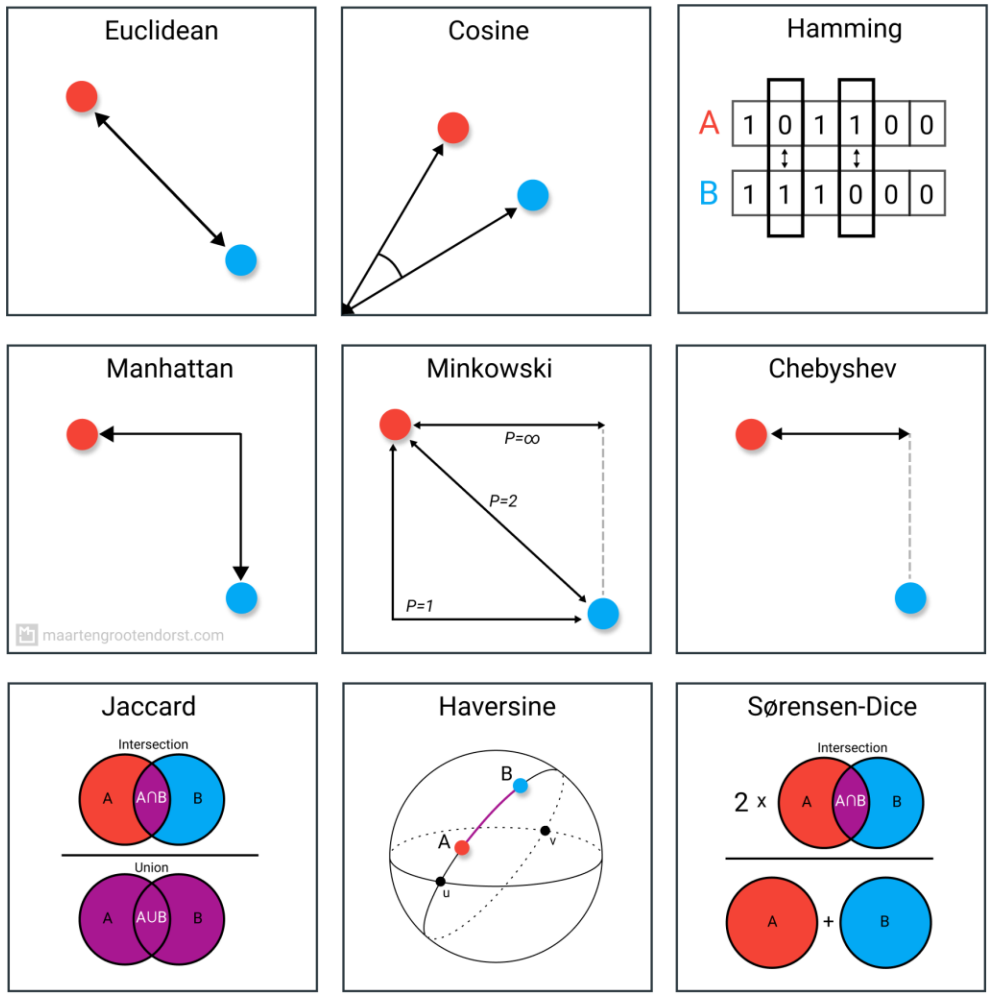
- LLMs generate one token at a time, a 500 word response means you ran the model 500 times
- The models "must" generate tokens that are relevant to the prompt.
- The next token is selected from a list of most likely tokens based on probabilities.
- A "temperature" setting is there to allow the model to *not* pick the most likely next token
- As the model keeps picking next tokens there can be a subtle shift in what the model is talking about.
- If the model does not have information about the subject (or very little of it) it still must pick a token, it cannot just stop

How can we prevent this?



- We can continue training the model (fine-tuning) for a specific task with more relevant data (expensive, difficult)
- We can use a model that has learned to think about its answer before it generates new tokens (reasoning models). In some instances, we can review this reasoning (hard to find a good model for every task, even harder to find good data, they need more resources)
- We can build a pipeline that retrieves data from relevant sources to "ground" the model in its response (RAG, retrieval augmented generation), (easier to run, more work to set-up)

Fantastic distances and where to find them



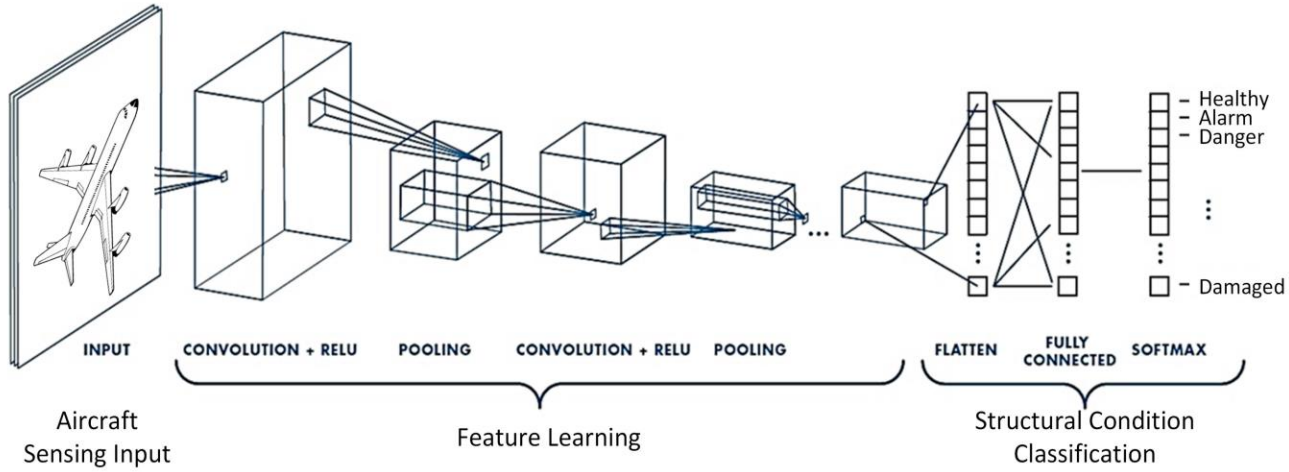
- In a RAG setup we can just run a keyword search (easiest) but often we are interested in things that are related but not identical
- We need a way to compare things (usually text) using their meaning and context not just what they are.
- LLMs represent this meaning as embeddings
- We can take the embeddings of our question and compare it to a database of pre-calculated embeddings
- We can use these relevant texts as part of the prompt to minimize hallucinations.
- The kind of distance metric used depends on the kind of data.

Agents, (a license to run –other programs)

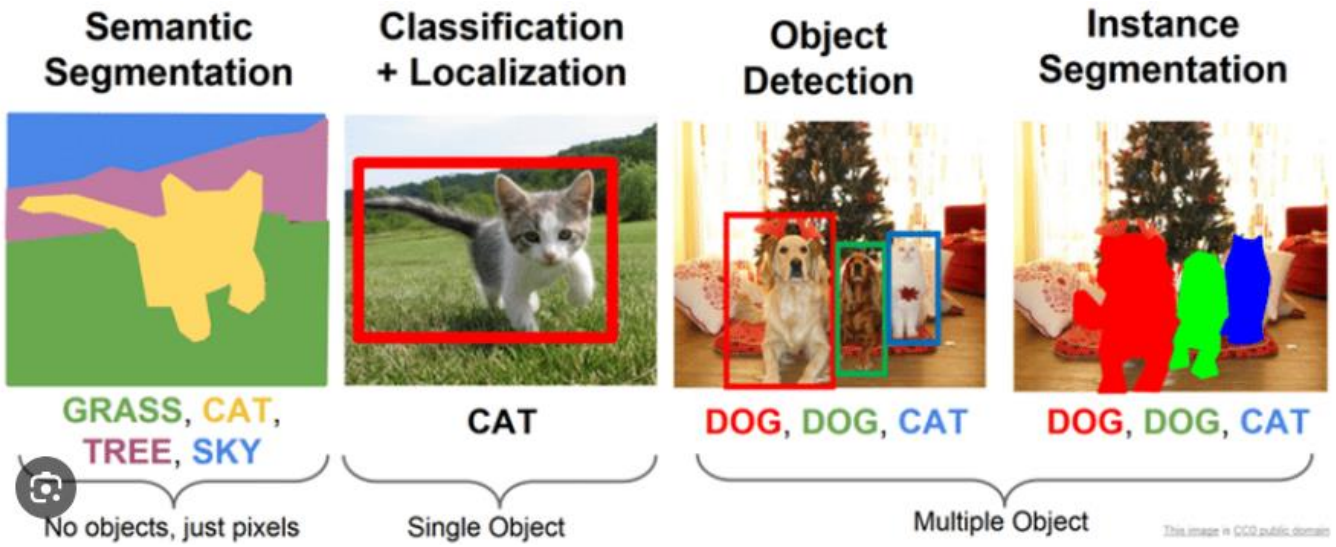
- We can take this concept a bit further.
- What if instead of a database of documents we had a database of programs and their descriptions (this is MCP, model context protocol)
- We can ask the llm to find the most reasonable program to run, then run it and report the answer.
- These can be simple programs that return values, download files, search the web, etc.
- They can be other llms that are trained on specific tasks
- Each of these programs or llms that call other programs are called agents.

Computer vision, a .jpg is worth a 1000 tokens

- Convolutional neural networks are generally used for "traditional" computer vision tasks
- Good for data with local structure (i.e. images)
- Used to decrease the number of dimensions in one axis and project them onto another to learn "features"
- Several of them can be used back-to-back to learn features of features
- This is to remove the variance on local structure (where things are on the image) and focus on the things we are looking for.



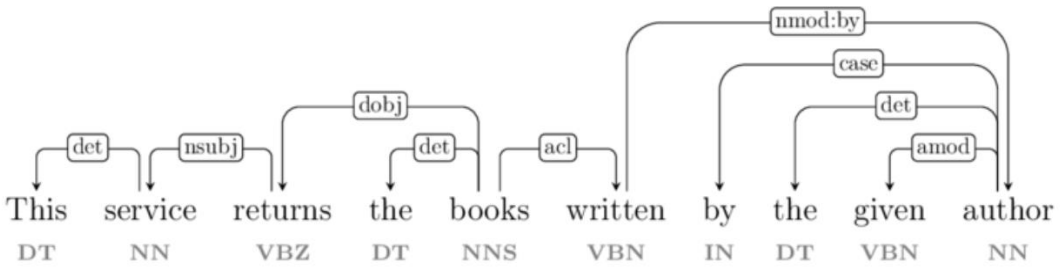
Seeing is believing



- CNNs are supervised models
- We can classify images (whole image)
- We can find things inside an image
- We can separate the image into sections
- The context of training is extremely important
- There is no reasoning in CNNs and they have to pick a label that is within the training data

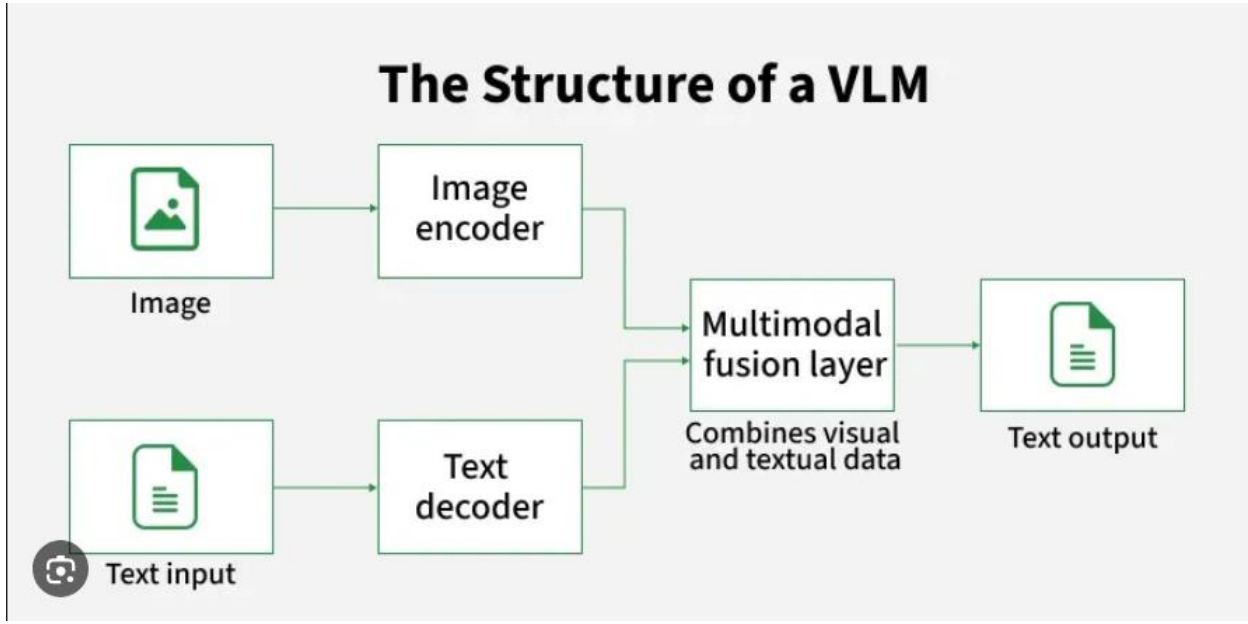
Seeing is believing, now with attention

- It's not difficult to see how this applies to language because there are built in dependencies between parts of a sentence.
- This dependency graph is different for each sentence, and it might completely change with different languages.
- But is there such a thing as a dependency graph in images?



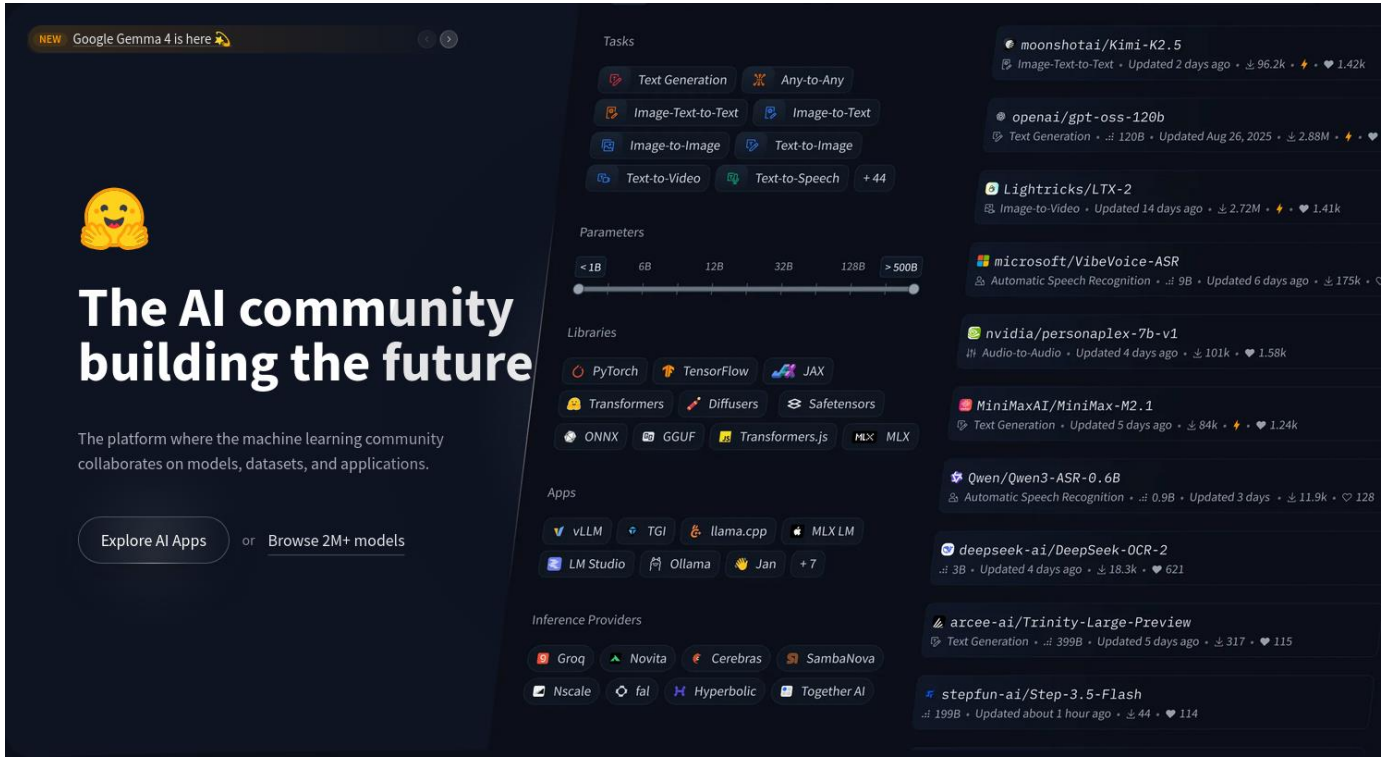
Multi-modal models, it's all transformers

- If these are all transformers, and work in similar ways we can combine them
- We can tokenize and generate embeddings from text
- We can do the same for images
- We can combine them in different ways and pass them to a 3rd transformer
- This transformer will be generate the outputs based on image and/or text.



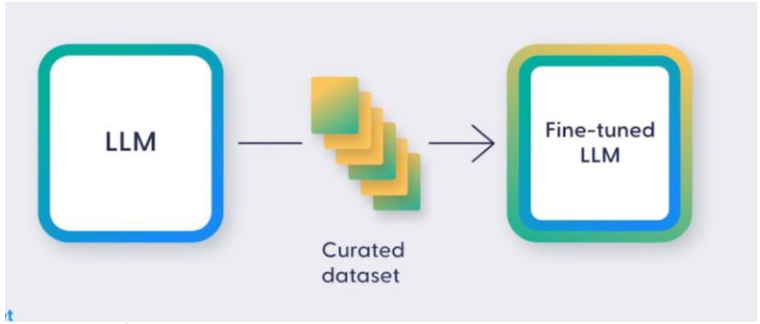
Can I have my own LLMs? Sure, why not?

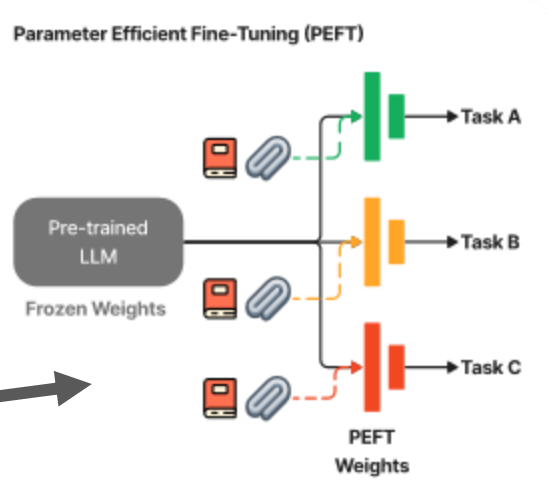
- For most tasks, it is unlikely that you will need a novel architecture
- There are many open source/open weight models that you can use out of the box
- These come in many shapes and sizes and capabilities.
- You can enhance your llms abilities with RAG or with agents
- You will need powerful GPU(s)

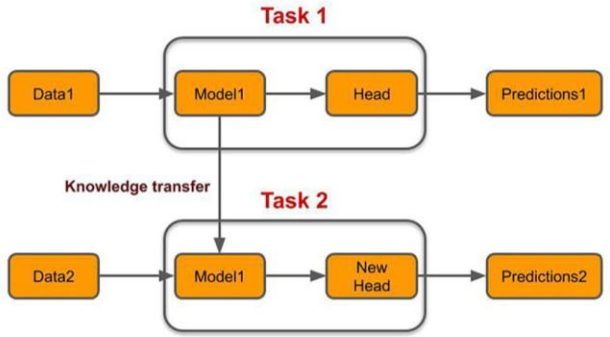


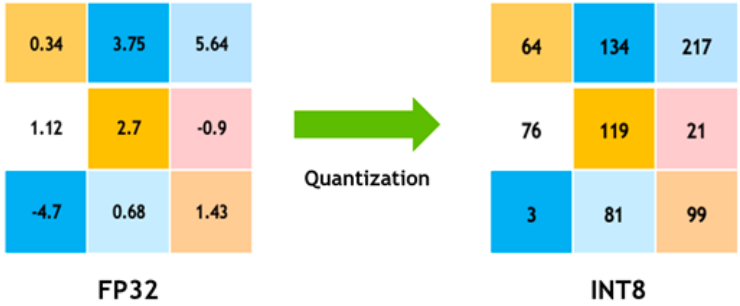
Foundation models (a.k.a. how to tame a giant)

- We do not have the computational resources of Google or Meta
- We cannot train a 300/600B parameter model from scratch in a reasonable time frame
- We can take a "pre-trained" model and
  - Fine tune it
  - Use adapters
  - Quantize it
  - Keep most of it frozen and train only the last bits









CPU, GPU, TPU all from a bucket of sand

CPU (Central Processing Unit)

- General-purpose processor
- Few powerful cores Best for control flow, logic, and sequential tasks

GPU (Graphics Processing Unit)

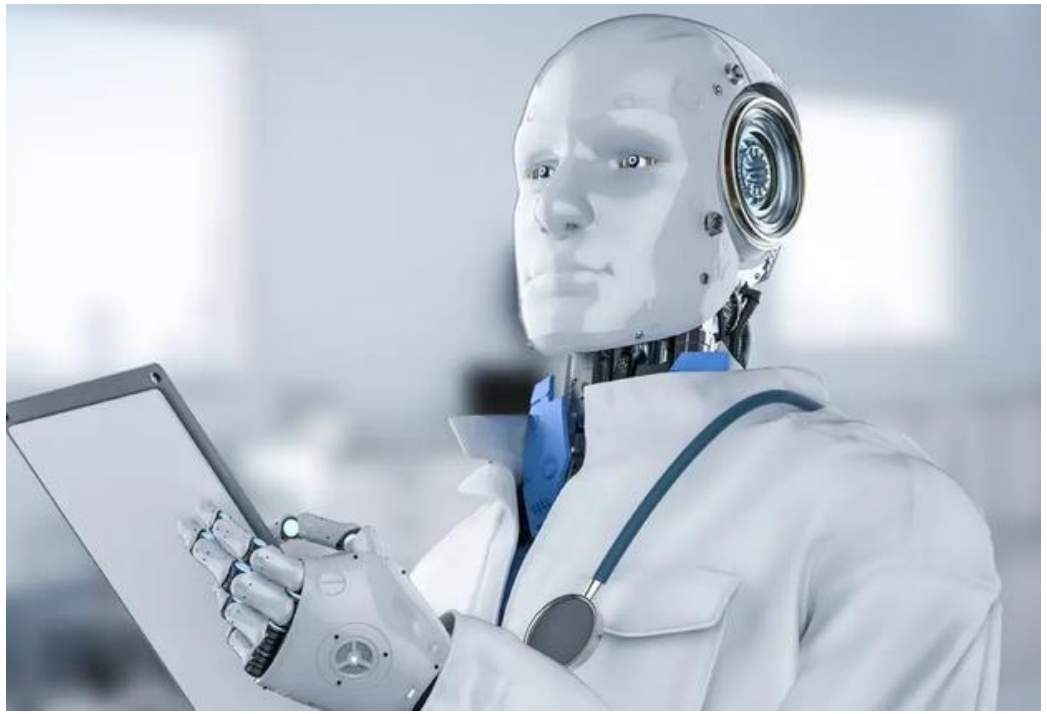
- Massively parallel processor
- Thousands of lightweight cores
- Best for matrix math, graphics, and deep learning

TPU (Tensor Processing Unit)

- Specialized accelerator for machine learning
- Optimized for tensor/matrix operations
- Best for large-scale ML training and inference

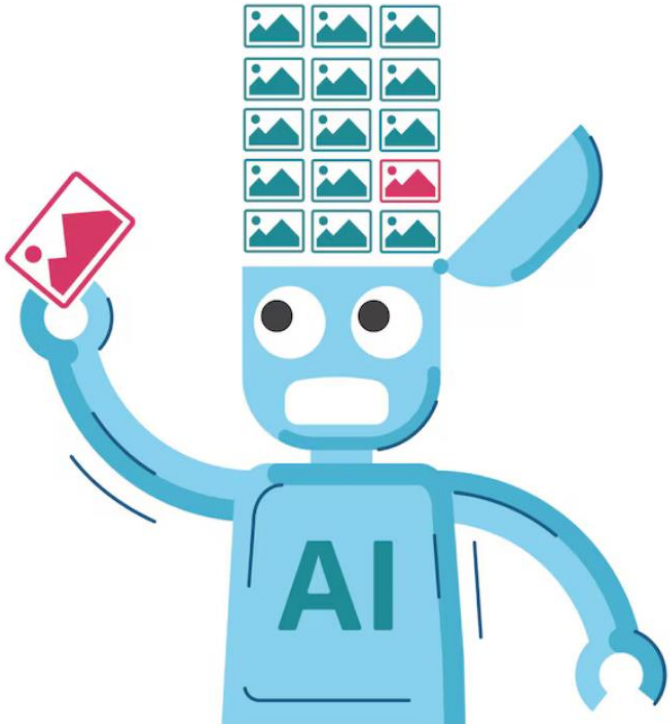
How does this apply for health data?

- Health data has a few nuances that creates challenges in training and inference
  - Nuance, small differences in symptoms may mean very different things and vice, versa
  - Not all relevant data is in a single format (text, tables, images)
  - There are socio-economic reasons that may or may not be relevant, it is not obvious from labels
  - Data is hard to get, subject to many regulations datasets are often small, not diverse, limited to single modality.
  - Data needs to be meaningfully anonymized and diversified



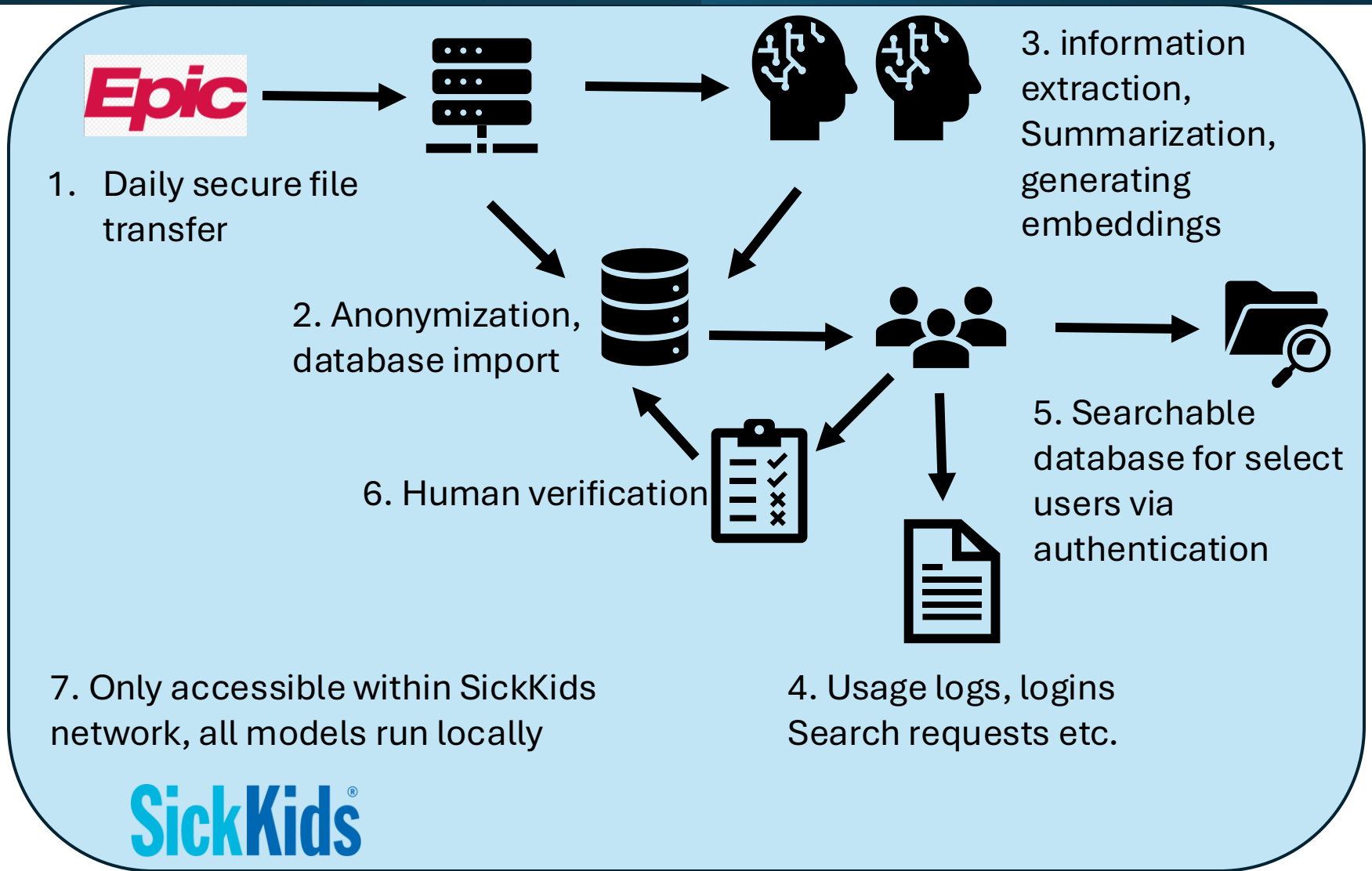
Post training considerations

- After model has been trained it needs to be:
  - Silently evaluated before deployment
  - Needs to be strictly version controlled (not just the model but the data and how it was trained as well)
  - Needs to have additional guardrails against leaking sensitive information (if applicable)
  - There needs to be strict access authorization for the application, logging and application data and usage need to be easily auditable

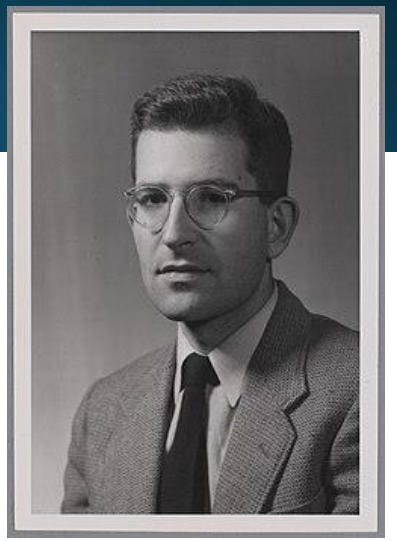


One current example, CHIRPP

- Canadian hospitals injury reporting and prevention program
- Aims to collect data on injuries and mental health issues
- Data collected on various incidents such as Physical injuries (accidents and assault) , Self-harm, suicidal ideation , Parental neglect/abuse , Violent behavior
- The data is then used for analyzing trends and guiding public health policies
- Traditionally this was done manually and was extremely time consuming



Things I think about when I don't want to work but want to feel productive



- Noam Chomsky is was a professor of linguistics in MIT.
- He proposed the idea of a universal language grammar, a set of rules that define the expressibility of language regardless of language
- These rules were universal and were *generative.* Meaning, a finite set of rules can be used to generate infinite combinations of text even different rules to generate text.
- It is criticized to be too abstract and to limited to model all the complexities of human language and its capabilities of expression.
- He also proposed the idea of humans' innate ability for language and that children have "scarcity of information" yet they master language at an early age.
- What if the universal grammar is not a set of simple generative rules but a bunch of differential equations?

$$\mathcal{L} = -\frac{1}{4}F_{\mu\nu}F^{\mu\nu}$$
$$+i\bar{\psi}D\psi + h.c.$$
$$+\bar{\psi}_i y_{ij}\psi_j\phi + h.c.$$
$$+|D_\mu\phi|^2 - V(\phi)$$

$$\mathcal{L}_{SM} = -\tfrac{1}{2}\partial_\nu g^a_\mu \partial_\nu g^a_\mu - g_s f^{abc}\partial_\mu g^a_\nu g^b_\mu g^c_\nu - \tfrac{1}{4}g_s^2 f^{abc}f^{ade} g^b_\mu g^c_\nu g^d_\mu g^e_\nu - \partial_\nu W^+_\mu \partial_\nu W^-_\mu - M^2 W^+_\mu W^-_\mu - \tfrac{1}{2}\partial_\nu Z^0_\mu \partial_\nu Z^0_\mu - \tfrac{1}{2c_w^2}M^2 Z^0_\mu Z^0_\mu - \tfrac{1}{2}\partial_\mu A_\nu \partial_\mu A_\nu - igc_w(\partial_\nu Z^0_\mu (W^+_\mu W^-_\nu - W^+_\nu W^-_\mu) - Z^0_\nu (W^+_\mu \partial_\nu W^-_\mu - W^-_\mu \partial_\nu W^+_\mu) + Z^0_\mu (W^+_\nu \partial_\nu W^-_\mu - W^-_\nu \partial_\nu W^+_\mu)) - igs_w(\partial_\nu A_\mu (W^+_\mu W^-_\nu - W^+_\nu W^-_\mu) - A_\nu (W^+_\mu \partial_\nu W^-_\mu - W^-_\mu \partial_\nu W^+_\mu) + A_\mu (W^+_\nu \partial_\nu W^-_\mu - W^-_\nu \partial_\nu W^+_\mu)) - \tfrac{1}{2}g^2 W^+_\mu W^-_\mu W^+_\nu W^-_\nu + \tfrac{1}{2}g^2 W^+_\mu W^-_\nu W^+_\mu W^-_\nu + g^2 c_w^2 (Z^0_\mu W^+_\mu Z^0_\nu W^-_\nu - Z^0_\mu Z^0_\mu W^+_\nu W^-_\nu) + g^2 s_w^2 (A_\mu W^+_\mu A_\nu W^-_\nu - A_\mu A_\mu W^+_\nu W^-_\nu) + g^2 s_w c_w (A_\mu Z^0_\nu (W^+_\mu W^-_\nu - W^+_\nu W^-_\mu) - 2A_\mu Z^0_\mu W^+_\nu W^-_\nu) - \tfrac{1}{2}\partial_\mu H \partial_\mu H - 2M^2 \alpha_h H^2 - \partial_\mu \phi^+ \partial_\mu \phi^- - \tfrac{1}{2}\partial_\mu \phi^0 \partial_\mu \phi^0 - \beta_h \left(\tfrac{2M^2}{g^2} + \tfrac{2M}{g}H + \tfrac{1}{2}(H^2 + \phi^0\phi^0 + 2\phi^+\phi^-) \right) + \tfrac{2M^4}{g^2}\alpha_h - g\alpha_h M \left(H^3 + H\phi^0\phi^0 + 2H\phi^+\phi^-\right) - \tfrac{1}{8}g^2 \alpha_h \left(H^4 + (\phi^0)^4 + 4(\phi^+\phi^-)^2 + 4(\phi^0)^2\phi^+\phi^- + 4H^2\phi^+\phi^- + 2(\phi^0)^2 H^2\right) - gMW^+_\mu W^-_\mu H - \tfrac{1}{2}g\tfrac{M}{c_w^2} Z^0_\mu Z^0_\mu H - \tfrac{1}{2}ig\left(W^+_\mu(\phi^0\partial_\mu\phi^- - \phi^-\partial_\mu\phi^0) - W^-_\mu(\phi^0\partial_\mu\phi^+ - \phi^+\partial_\mu\phi^0)\right) + \tfrac{1}{2}g\left(W^+_\mu(H\partial_\mu\phi^- - \phi^-\partial_\mu H) + W^-_\mu(H\partial_\mu\phi^+ - \phi^+\partial_\mu H)\right) + \tfrac{1}{2}g\tfrac{1}{c_w}(Z^0_\mu(H\partial_\mu\phi^0 - \phi^0\partial_\mu H) + M\left(\tfrac{1}{c_w}Z^0_\mu\partial_\mu\phi^0 + W^+_\mu\partial_\mu\phi^- + W^-_\mu\partial_\mu\phi^+\right) - ig\tfrac{s_w^2}{c_w}MZ^0_\mu(W^+_\mu\phi^- - W^-_\mu\phi^+) + igs_wMA_\mu(W^+_\mu\phi^- - W^-_\mu\phi^+) - ig\tfrac{1-2c_w^2}{2c_w}Z^0_\mu(\phi^+\partial_\mu\phi^- - \phi^-\partial_\mu\phi^+) + igs_wA_\mu(\phi^+\partial_\mu\phi^- - \phi^-\partial_\mu\phi^+) - \tfrac{1}{4}g^2 W^+_\mu W^-_\mu \left(H^2 + (\phi^0)^2 + 2\phi^+\phi^-\right) - \tfrac{1}{8}g^2\tfrac{1}{c_w^2}Z^0_\mu Z^0_\mu \left(H^2 + (\phi^0)^2 + 2(2s_w^2 - 1)^2\phi^+\phi^-\right) - \tfrac{1}{2}g^2\tfrac{s_w^2}{c_w}Z^0_\mu\phi^0(W^+_\mu\phi^- + W^-_\mu\phi^+) - \tfrac{1}{2}ig^2\tfrac{s_w^2}{c_w}Z^0_\mu H(W^+_\mu\phi^- - W^-_\mu\phi^+) + \tfrac{1}{2}g^2 s_w A_\mu\phi^0(W^+_\mu\phi^- + W^-_\mu\phi^+) + \tfrac{1}{2}ig^2 s_w A_\mu H(W^+_\mu\phi^- - W^-_\mu\phi^+) - g^2\tfrac{s_w}{c_w}(2c_w^2 - 1)Z^0_\mu A_\mu\phi^+\phi^- - g^2 s_w^2 A_\mu A_\mu \phi^+\phi^- + \tfrac{1}{2}ig_s\lambda^a_{ij}(\bar{q}^\sigma_i\gamma^\mu q^\sigma_j)g^a_\mu - \bar{e}^\lambda(\gamma\partial + m^\lambda_e)e^\lambda - \bar{\nu}^\lambda(\gamma\partial + m^\lambda_\nu)\nu^\lambda - \bar{u}^\lambda_j(\gamma\partial + m^\lambda_u)u^\lambda_j - \bar{d}^\lambda_j(\gamma\partial + m^\lambda_d)d^\lambda_j + igs_wA_\mu\left(-(\bar{e}^\lambda\gamma^\mu e^\lambda) + \tfrac{2}{3}(\bar{u}^\lambda_j\gamma^\mu u^\lambda_j) - \tfrac{1}{3}(\bar{d}^\lambda_j\gamma^\mu d^\lambda_j)\right) + \tfrac{ig}{4c_w}Z^0_\mu\{(\bar{\nu}^\lambda\gamma^\mu(1+\gamma^5)\nu^\lambda) + (\bar{e}^\lambda\gamma^\mu(4s_w^2 - 1 - \gamma^5)e^\lambda) + (\bar{d}^\lambda_j\gamma^\mu(\tfrac{4}{3}s_w^2 - 1 - \gamma^5)d^\lambda_j) + (\bar{u}^\lambda_j\gamma^\mu(1 - \tfrac{8}{3}s_w^2 + \gamma^5)u^\lambda_j)\} + \tfrac{ig}{2\sqrt{2}}W^+_\mu\left((\bar{\nu}^\lambda\gamma^\mu(1+\gamma^5)U^{lep}{}_{\lambda\kappa}e^\kappa) + (\bar{u}^\lambda_j\gamma^\mu(1+\gamma^5)C_{\lambda\kappa}d^\kappa_j)\right) + \tfrac{ig}{2\sqrt{2}}W^-_\mu\left((\bar{e}^\kappa U^{lep\dagger}{}_{\kappa\lambda}\gamma^\mu(1+\gamma^5)\nu^\lambda) + (\bar{d}^\kappa_j C^\dagger_{\kappa\lambda}\gamma^\mu(1+\gamma^5)u^\lambda_j)\right) + \tfrac{ig}{2M\sqrt{2}}\phi^+\left(-m^\kappa_e(\bar{\nu}^\lambda U^{lep}{}_{\lambda\kappa}(1-\gamma^5)e^\kappa) + m^\lambda_\nu(\bar{\nu}^\lambda U^{lep}{}_{\lambda\kappa}(1+\gamma^5)e^\kappa\right) + \tfrac{ig}{2M\sqrt{2}}\phi^-\left(m^\lambda_e(\bar{e}^\lambda U^{lep\dagger}{}_{\lambda\kappa}(1+\gamma^5)\nu^\kappa) - m^\kappa_\nu(\bar{e}^\lambda U^{lep\dagger}{}_{\lambda\kappa}(1-\gamma^5)\nu^\kappa\right) - \tfrac{g}{2}\tfrac{m^\lambda_\nu}{M}H(\bar{\nu}^\lambda\nu^\lambda) - \tfrac{g}{2}\tfrac{m^\lambda_e}{M}H(\bar{e}^\lambda e^\lambda) + \tfrac{ig}{2}\tfrac{m^\lambda_\nu}{M}\phi^0(\bar{\nu}^\lambda\gamma^5\nu^\lambda) - \tfrac{ig}{2}\tfrac{m^\lambda_e}{M}\phi^0(\bar{e}^\lambda\gamma^5 e^\lambda) - \tfrac{1}{4}\,\bar{\nu}_\lambda M^R_{\lambda\kappa}(1-\gamma_5)\hat{\nu}_\kappa - \tfrac{1}{4}\,\bar{\nu}_\lambda M^R_{\lambda\kappa}(1-\gamma_5)\hat{\nu}_\kappa + \tfrac{ig}{2M\sqrt{2}}\phi^+\left(-m^\kappa_d(\bar{u}^\lambda_j C_{\lambda\kappa}(1-\gamma^5)d^\kappa_j) + m^\lambda_u(\bar{u}^\lambda_j C_{\lambda\kappa}(1+\gamma^5)d^\kappa_j\right) + \tfrac{ig}{2M\sqrt{2}}\phi^-\left(m^\lambda_d(\bar{d}^\lambda_j C^\dagger_{\lambda\kappa}(1+\gamma^5)u^\kappa_j) - m^\kappa_u(\bar{d}^\lambda_j C^\dagger_{\lambda\kappa}(1-\gamma^5)u^\kappa_j\right) - \tfrac{g}{2}\tfrac{m^\lambda_u}{M}H(\bar{u}^\lambda_j u^\lambda_j) - \tfrac{g}{2}\tfrac{m^\lambda_d}{M}H(\bar{d}^\lambda_j d^\lambda_j) + \tfrac{ig}{2}\tfrac{m^\lambda_u}{M}\phi^0(\bar{u}^\lambda_j\gamma^5 u^\lambda_j) - \tfrac{ig}{2}\tfrac{m^\lambda_d}{M}\phi^0(\bar{d}^\lambda_j\gamma^5 d^\lambda_j) + \bar{G}^a\partial^2 G^a + g_s f^{abc}\partial_\mu\bar{G}^a G^b g^c_\mu + \bar{X}^+(\partial^2 - M^2)X^+ + \bar{X}^-(\partial^2 - M^2)X^- + \bar{X}^0(\partial^2 - \tfrac{M^2}{c_w^2})X^0 + \bar{Y}\partial^2 Y + igc_w W^+_\mu(\partial_\mu\bar{X}^0X^- - \partial_\mu\bar{X}^+X^0) + igs_w W^+_\mu(\partial_\mu\bar{Y}X^- - \partial_\mu\bar{X}^+Y) + igc_w W^-_\mu(\partial_\mu\bar{X}^-X^0 - \partial_\mu\bar{X}^0X^+) + igs_w W^-_\mu(\partial_\mu\bar{X}^-Y - \partial_\mu\bar{Y}X^+) + igc_w Z^0_\mu(\partial_\mu\bar{X}^+X^+ - \partial_\mu\bar{X}^-X^-) + igs_w A_\mu(\partial_\mu\bar{X}^+X^+ - \partial_\mu\bar{X}^-X^-) - \tfrac{1}{2}gM\left(\bar{X}^+X^+H + \bar{X}^-X^-H + \tfrac{1}{c_w^2}\bar{X}^0X^0H\right) + \tfrac{1-2c_w^2}{2c_w}igM\left(\bar{X}^+X^0\phi^+ - \bar{X}^-X^0\phi^-\right) + \tfrac{1}{2c_w}igM\left(\bar{X}^0X^-\phi^+ - \bar{X}^0X^+\phi^-\right) + igMs_w\left(\bar{X}^0X^-\phi^+ - \bar{X}^0X^+\phi^-\right) + \tfrac{1}{2}igM\left(\bar{X}^+X^+\phi^0 - \bar{X}^-X^-\phi^0\right).$$

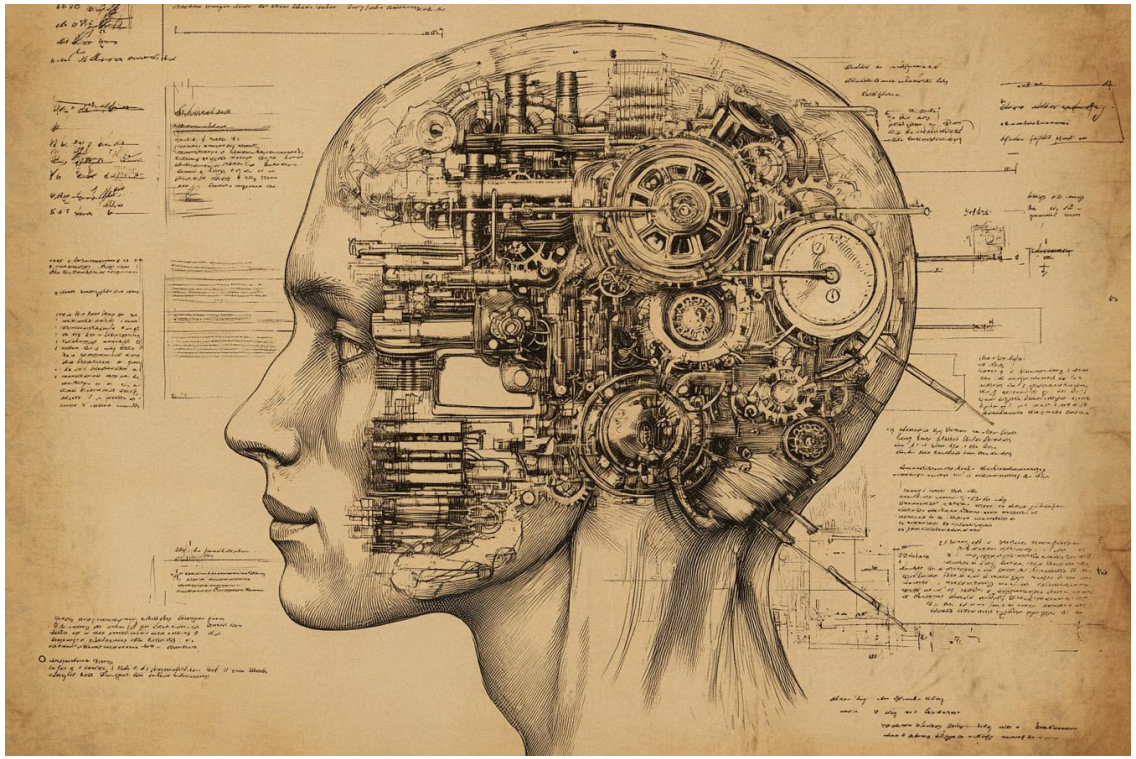
Things I think about when I don't want to work but want to feel productive

- Ludwig Wittgenstein was a philosopher in the early 20th century. He was a student of Bertrand Russel.
- His main work *Tractatus Logico Phillosophicus* investigated how language can influence our ability to express things.
- He suggested that propositions are "pictures" of the thing they propose. In his other fully published work (posthumously) *Philosophical investigations* he further elaborates a "language game" where the context of language affects its meaning and its ability to express things.
- Can we explore LLMs (single language and multi-language) to see if their expressivity and the vector embedding distances between concepts remain equivariant?

From DeepSeek R1 paper

clear solutions. During the training process, we observe that CoT often exhibits language mixing, particularly when RL prompts involve multiple languages. To mitigate the issue of language mixing, we introduce a language consistency reward during RL training, which is calculated as the proportion of target language words in the CoT. Although ablation experiments show that such alignment results in a slight degradation in the model's performance, this reward aligns with human preferences, making it more readable. Finally, we combine the accuracy of

Things I think about when I don't want to work but want to feel productive



- We are like most mammals are inherently curious creatures
- We are like most primates are social creatures with relationships, personalities and societies
- We do like all great apes pass on information from generation to generation
- Our language and its ability pass information through abstraction is the main source of our strength
- Science is curiosity, abstracted by language, constrained by reproducibility, practiced by a global community

Questions?
Comments?

Thank you!