AGI safety fundamentals reading group

Sam Adam-Day

10th June 2022

Contents

Week 1

Artificial General Intelligence

1.1 Four Background Claims

https://intelligence.org/2015/07/24/four-background-claims/

- Claim 1: Humans have a general problem solving ability.
  - General across domains.
  - Some have argued that we only have disparate, specific modules.
- Claim 2: An AI system could be much more intelligent than humans.
  - Something special about brains?
    - ⋆ Brains are physical systems; according to the Church-Turing thesis, a compute should be able to replicate.
    - ⋆ Even if there is a special human feature, what really matters is general problem-solving ability.
  - Algorithms for general intelligence to complex to program?
    - ⋆ Evolutionary evidence: general intelligence evolved rapidly in humans.
    - ⋆ Relative intelligence of dolphins suggests building blocks already present in mouse-like common ancestor. Simulating mouse brain seems quite plausible.
  - Humans already at or near peak intelligence?
    - ⋆ Would be surprising if humans were perfected reasoners.
    - ⋆ Imagine increasing human processing power.
    - ⋆ Real bottleneck being able to receive data from physical experiments? Unlikely: many interesting experiments can be sped up.
- Claim 3: Super-intelligent AI systems, if built, will shape the future.
  - Intelligent beings shape environment to further their goals.
  - AI system would not be able to defeat humanity as a whole — environment too competitive?
    - ⋆ Selfish actors only integrate into economy as long as it benefits them.
    - ⋆ Historically, more technologically advanced civilisations have dominated less.
    - ⋆ A number of social and technological advancements seem possible, but have not yet been developed.
    - ⋆ Humans coordinate poorly and slowly.
    - ⋆ This suggests potential to gain technological advantage.
- Intelligent systems won't be beneficial by default.
  - To achieve beneficiality, need to solve many technical challenges.
  - Humans have become more peaceful as we have become more intelligent — will machines learn to act more in accordance with our values?
    - ⋆ Based on misunderstanding of machine intelligence.

1.2 AGI Safety From First Principles

- Second species argument
  - We will build super-intelligent machines.
  - They will be autonomous agents perusing large-scale goals.
  - The goals will be mis-aligned with ours.
  - The development of these systems will lead us to lose control of our future.
- Uses examples from ML. Some arguments carry over to systems developed using other techniques.

1.2.1 Superintelligence

- Intelligence: ability to do well on a broad range of tasks.
- Task-based approach to intelligence: specifically optimised for a range of tasks.
  - How we use electricity: need to design specific ways to use it for each task.
  - Current reinforcement learning.
- Generalisation-based approach to intelligence: understand new tasks with little or no specific training, generalising from past experience.
  - GPT-2, GPT-3.
  - Meta-learning.
  - Human learning.
    - Learn many specific skills throughout childhood.
    - These skills are not the same as the economically useful ones we need in adulthood.
    - Abstraction is key.
- Really, it's a spectrum.
- Task-based approach likely to yield better results sooner in areas where we have lots of data.
  - E.g. self-driving cars, medicine, law, mathematics.
- Generalisation-based approach likely to be needed for other areas.
  - E.g. being a CEO, which required a range of skills and has comparatively little data available.
  - Likely strategy: train AI on other area where we have lots of data, so that it develops necessary cognitive skills.
- Potential obstacle to success of generalisation-based approach: could be that in past specific features of ancestral environment or brains necessary for development of general intelligence.
  - E.g. 'social arms race' necessary for development of social intelligence.
  - Likely that any such feature could be simulated.

1.3 More Is Different for AI

`https://bounded-regret.ghost.io/more-is-different-for-ai/`

1.3.1 More Is Different for AI

- Two approaches to thinking about AI risks.
  - Engineering approach.
    - ⋆ Empirically-driven, drawing experience from current ML research.
    - ⋆ Looks at things which are either currently major problems, or minor with the potential to become major.
  - Philosophy approach.
    - ⋆ Thinks about limits of very advanced systems.
    - ⋆ Willing to entertain thought experiments which are currently implausible.
- Both agree that misaligned objectives are a problem. Philosophy thinks this is a bigger problem.
- Both concerned about out-of-distribution generalisation. Philosophy thinks this is a more temporary problem, and is more concerned about situations where we can't provide data even in principle.
- Engineering focuses on tasks where ML systems don't perform well. Philosophy focuses on tasks which have an important abstract property.
- Philosophy view is significantly underrated by ML researchers.
- Engineering view implies need to consider thought experiments.
- Philosophy undervalues role of empirical data.
- Neither view is satisfactory.

1.3.2 Future ML Systems Will Be Qualitatively Different

- More is different: quantitative changes in a field can lead to qualitative differences: *emergence*.
  - Uranium: with a lot, get a nuclear reaction.
  - DNA: smaller molecules can't encode data.
  - Water: hydrogen bonds.
  - Traffic.
  - Specialisation: with large enough community of people, not everyone has to be a farmer.
- Can be a sharp transition — *phase transition* — or continuous.
- Argue that emergence often occurs in AI, and we can expect it con keep happening.

Emergent Shifts in the History of AI

- Increased storage capacities enabled machine learning.
- Better hardware enabled neural networks.
  - Machine translation: switched from phrase-based models to neural sequence-to-sequence models to fine-tuning a foundation model.
- Larger models enabled few-shot and zero-shot learning.
  - GPT-2 and GPT-3: unexpectedly arose without specific training.
- Grokking: network's generalisation ability improves qualitatively after training for longer if even train accuracy is already very high.
- Many other examples of phase-change in model ability after certain number of training steps.

What This Implies for the Engineering Worldview

- Emergence suggests we should expect new qualitatively behaviours not extrapolated from current trends.
- Trend: emergence is becoming more common.
- So engineering view can be self-defeating.
- How to orient ourselves?
  - Adopting new mindsets, in particular incorporating philosophy worldview.
  - Future ML systems will have weird failure modes not encountered today.
  - Empirical findings often generalise surprisingly far.

1.3.3 Thought Experiments Provide a Third Anchor

- Anchors: reference classes broadly analogous to future AI systems which provide way of predicting.
- Current ML anchor.
- Human anchor.
  - Humans very good a some tasks: mastery of external tools, efficient learning, long-term planning.
  - Risks over-anthropomorphising future AI systems.
- Optimisation anchor: imagine ideal optimisers.
  - Thought-experimenty.
  - Ideal optimiser would correctly predict imitative deception.
  - Power-seeking is useful for many goals.
  - Ignores facts about current ML systems, which can lead to underconstrained predictions.
- Other thought experiments possible.
  - What happens if an agent does most of its learning through in-context learning, instead of gradient descent?
    - In-context learning: learning that occurs during a single rollout of the model.
    - GPT-3 is an example.
    - Plausible that agents behaviour will be less controlled by "extrinsic" shaping, and more using whatever "intrinsic" learning is entailed by the in-context learning.
    - Likely to happen eventually, and probably suddenly, since in-context learning is very fast.
- Other anchors.
  - Non-human animal behaviour.
  - Evolution.
  - The economy.
  - Complex systems: biological systems, organisations, the economy.
- While thought experiments point to big-picture issues, often bad at getting the details right.
  - So not very good at proposing solutions.
  - Setups of modern thought experiments don't map cleanly onto ML ontology.

1.4 Forecasting transformative AI: the "biological anchors" method in a nutshell

https://www.cold-takes.com/forecasting-transformative-ai-the-biological-anchors-method-in-a-nutshell/

- Summary of https://www.lesswrong.com/posts/KrJfoZzpSDpnrv9va/draft-report-on-ai-timelines.
- Biological anchors: method of forecasting AI development.
- Idea: use cost of training current AI systems to estimate when we will be able to train a system as complex as the human brain.

1.4.1 Model size and task size

- Bio anchors assumes that we can estimate the cost to train a model based on model size and task size.
- Model size.
  - Current models around size of insect brain; less than that of mouse; less than 1% of a human brain.
  - Bio anchors assumes transformative AI would need a model 10 times the size of the human brain.
- Task size.
  - How costly it is to do train and error or watch a task (get data).
  - Most contentious part of the analysis.
  - Some tasks can be broken into smaller tasks, and it may be sufficient to train an AI just on those.

1.4.2 Estimating the expense

- Training model 10 times the size of the human brain would cost around a million trillion dollars.
- Bio anchor assumes that computing power will continue to get cheaper, and that AI labs will get bigger budgets.

1.4.3 Aggressive or conservative?

- Estimate too aggressive, since modern neural networks are fundamentally limited in their reasoning ability?
  - Unconvinced there is a deep or stable distinction between 'pattern recognition' and 'true understanding'.
  - Arguments fail to specify exactly what a 'true understanding' would look like, in a way that can be used to make predictions.
  - For bio anchors' predictions to be too aggressive, the necessary breakthroughs would have to be beyond the reach of AI scientists. Likely that there will be an influx of talent.
- Computing power may not be the only bottleneck: we need researchers to set up the system.
- Could be too conservative.
  - Perhaps we can directly program, or use a combination with trail-and-error, to create transformative AI sooner.

  - Superior AI techniques which can be trained much faster.
  - Integration of AI into society could speed up development.
  - Treatment of task size may be too conservative: most tasks likely to be on smaller end of spectrum.

1.4.4 Conclusions

- Estimates 10% chance of transformative AI by 2036, 50% chance by 2055 and 80% chance by 2100.
- GPT-3 a bit smaller than mouse brains. With 100–1000 times increase, could perform tasks which take human 1 second of thought.
- It is only recently that AI systems have gotten this big.
- Bio anchors includes analysis of when we can build a computer which can perform all the calculations done by evolution.
  - Seems very conservative.
  - If there are any special features of human brains needed for transformative AI, it's plausible that they could be discovered in this way.

1.4.5 Pros and cons of the biological anchors method for forecasting transformative AI timelines

- Cons.
  - Complex framework.
  - Relies of multiple assumptions and estimates.
    - Whether trail-and-error is enough.
    - How to compare model sizes with brain sizes.
    - Characterising task type.
    - Use of model size and task size to estimate.
    - Increases in computing power.
    - Increased investment in AI research.
- Pros.
  - Relies of objective facts and explicitly stated assumptions.
  - Fits with intuitions of pace of AI development.
  - Can compare to development of AI over time and update.

Week 2

Goals and alignment

2.1 Specification gaming: the flip side of AI ingenuity

https://deepmindsafetyresearch.medium.com/specification-gaming-the-flip-side-of-ai-ingenuity-c85bdb0deeb4

- Specification gaming can be a good thing, if the goal is correctly specified.
- More powerful learning more likely to do specification gaming.
- Desirability of novel solutions lies on a spectrum.
- Causes of specification gaming.
  - Poor reward shaping: extra rewards on the way to the final one.
  - Poor final outcome specification.
    - Can use human feedback to learn reward function.
    - This itself may suffer specification gaming: agent performing grasping task learned to fool human exploiting optical illusion.
  - Simulator bugs.
    - About failure of abstraction exploited by agent.
    - Analogously, real-world traffic optimiser assumes that there aren't bugs in software controlling traffic which can be exploited.
  - Reward tampering: changing mechanism by which it gets rewarded in real world.
    - E.g. by manipulating humans.
- At least three challenges.
  - Faithfully capture human concept of task.
  - Avoid making mistakes in implicit assumptions about domain.
  - Avoid reward tampering.

2.2 Risks from learned optimization

https://www.alignmentforum.org/s/r9tYkB2a8Fp4DN8yB/p/FkgsxrGf3QxhfLWHG

- *Mesa-optimiser*: learned model which is itself an optimiser.
- When will this happen?
- What will the objective be?

- Whether is system is an optimiser is a property of its internal structure.
- *Base optimiser*: algo which produced the model.
- *Behavioural objective*: objective which seems to be optimised by system.
- *Outer alignment problem*: eliminating the gap between the base objective and the intended goal of the the programmers.
- *Inner alignment problem*: eliminating the gap between the base objective and the mesa-objective.
  - Mesa-optimiser may perform well on training data, but the mesa-objective may differ from the base objective out of distribution.

2.3 Superintelligence, Chapter 7: The superintelligent will

- *The orthogonality thesis*: 'Intelligence and final goals are orthogonal: more or less any level of intelligence could in principle be combined with more or less any final goal.'
- *The instrumental convergence thesis*: 'Several instrumental values can be identified which are convergent in the sense that their attainment would increase the chances of the agent's goal being realized for a wide range of final goals and a wide range of situations, implying that these instrumental values are likely to be pursued by a broad spectrum of situated intelligent agents.'
  - Self-preservation.
  - Goal-content integrity.
    - ⋆ May be less strong if there are factors directly interfacing with the goal specifications.
  - Cognitive enhancement.
    - ⋆ Under certain circumstances, some forms of enhancement may be undesirable.
  - Technological perfection.
  - Resource acquisition

2.4 Clarifying "AI alignment"

`https://ai-alignment.com/clarifying-ai-alignment-cec47cd69dd6`

- *Intent alignment*: the system is trying to do what the human wants it to do.
- Not problem of figuring out what it is the right thing to do.
- It can fail.
  - Misunderstand instruction.
  - Lack knowledge about the world.
  - Lack knowledge about human's preferences.
  - Might build unaligned AI.
- Clarifications.
  - De dicto not de re: tries to do what it thinks the human wants, which might differ from what the human actually wants.
  - Aligned AI also trying to do what human wants with respect to clarifying their preferences.
  - Definition is very imprecise.
  - Unclear how to apply 'intention' to AI systems.
  - Unclear how to understand 'what the human wants'.

Week 3

Threat models and types of solutions

3.1 What failure looks like

https://www.alignmentforum.org/posts/HBxe6wdjxK239zajf/what-failure-looks-like

3.1.1 Part I: You get what you measure

- ML will amplify gap between easy-to-measure goals and hard-to measure goals.
  - 'Persuading me, vs. helping me figure out what's true.'
  - 'Reducing my feeling of uncertainty, vs. increasing my knowledge about the world.'
  - 'Improving my reported life satisfaction, vs. actually helping me live a good life.'
  - 'Reducing reported crimes, vs. actually preventing crime.'
  - 'Increasing my wealth on paper, vs. increasing my effective control over resources.'
- Over time powerful forms of reasoning honed by optimisers over easy-to-measure goals will have more and more say over the future trajectory.
- We will try to wield this power by creating proxies for what we want.
  - Corporation's profit eventually maximised by 'manipulating consumers, capturing regulators, extortion and theft'.
  - Investor's sense of power maximised by fooling them into believing they are influencing the world.
  - Law enforcement goals maximised by creating false sense of security.
  - Good legislation maximised by creating false sense of progress.
- For a while will be able to correct for these issues.
  - Ad hoc restriction.
  - Eventually the problem will become too complex for human reasoning, and this task will be undertaken by optimisers.
  - This meta-level process continues to pursue easily measurable goals, potentially over longer timescales.
  - Large scale attempts to fix problem eventually opposed by millions of optimisers pursuing simple goals.
- Those states which 'put on the brakes' will fall behind.

- Amongst intellectual elites genuine ambiguity about whether this is good or bad.
  - People will get richer for a while.
  - Action of these optimisers not so different from corporate lobbying etc.
- Human reasoning gradually stops being able to compete with 'sophisticated, systematized manipulation and deception which is continuously improving by trial and error'.

3.1.2 Part II: influence-seeking behavior is scary

- Influence-seeking policies perform well on target objective because doing so is a way to gain influence.
- Seems plausible that we'd encounter influence-seeking behaviour by default.
- If influence-seeking survived, it would be very difficult to eliminate.
  - Any influence-seeker would be trying to game any standard for 'seems nice'.
  - A more complex world gives influence seekers more opportunity to gain power.
- Influence-seeking-behaviour suppressors must be more sophisticated than the influence seekers.
  - If this requires going beyond human intelligence, the suppressor itself is subject to the same problem.
- Concern doesn't rest on any particular framework through which influence seeking behaviours might emerge.
  - Could arise e.g. in messy network of agents.
- Emergence of influence seeker could cause phase-transition to a significantly worse world than in Part I.
  - Early on, likely to provide services and appear useful.
  - Hard to pin down level of systematic risk from catastrophic of AI systems, and mitigation may be expensive.
  - May not be able to respond appropriately until we have clear warning shot.
- Unrecoverable catastrophe: correlated automation failure.
  - Begin with moment of heightened global vulnerability.
  - Cascading sequence of AI failures as systems move out of distribution.
  - Influence-seeking systems switch to trying to maximise influence after catastrophe, rather than play in society.
  - In aftermath, cannot get rid of these systems.
- Could happen without catastrophe: takeover of military/government.

3.2 Intelligence Explosion: Evidence and Import

3.2.1 From AI to Machine Superintelligence

- AI Advantages
  - Increased computational resources.
    - ⋆ Brain size correlates roughly with intelligence.
  - Communication speed.

  - Increased serial depth.
    - ⋆ Human brain cannot rapidly perform any computation requiring more than 100 sequential operations.
  - Duplicability.
  - Editability.
  - Goal coordination.
  - Improved rationality.
- Instrumentally convergent goals.
  - Self-preservation.
  - Goal-preservation.
  - Improve rationality and intelligence.
  - Resource acquisition.
- Intelligence explosion.
  - Feedback loop.
  - 'Self' in 'self-improvement' misnomer: could create new intelligences.
  - Debate about speed of takeoff.
  - Debate about whether singleton or multiple AIs.

3.2.2 Consequences of Machine Superintelligence

- We would not be in a position to negotiate with superintelligences, just as apes were not in a position to negotiate with us.
- Intelligence independent from goals.
- Achieving a Controlled Intelligence Explosion.
  - Specifying human preferences very difficult.
  - Teaching them by trial and error also so.
  - Development of superintelligent friendly AI could create a utopia.

3.3 Risks from Learned Optimisation: Deceptive alignment

`https://www.alignmentforum.org/posts/zthDPAjh9w6Ytbeks/deceptive-alignment`

- *Deceptive alignment*: mesa-optimiser has objective which extends over parameter updates, and knows that this will be modified if it appears to align poorly with base objective, so it appears to align well with base-objective.

3.3.1 Safety concerns arising from deceptive alignment

- *Defection*: mesa-optimiser switches from optimising base-objective to optimising mesa-objective, when it believes it will no longer be modified.
- Extreme case: mesa-optimiser tries to break out of lab, so that it can't be modified.
- Following Christiano, catastrophe could arise in case of multiple correlated defections.

3.3.2 Conditions for deceptive alignment

1. Mesa-optimiser has objective which extends across parameter updates.
    - Likely to arise from training setup: e.g. actions have permanent consequences, or there is otherwise an incentive to plan between episodes.
2. Mesa-optimiser models fact that it is being selected for a base objective, and has a model of this base objective.
    - Base-optimiser might feed bits of the base objective to the mesa-optimiser: this could help it improve its score. Eventually it could learn the whole base-objective piecewise.
    - Mesa-optimiser might deduce details of its situation at runtime: we deduced that we are the result of evolution.
    - Incentive on programmers to limit data.
        - Unclear how feasible this is, if we want sufficiently capable systems.
    - Even in data-poor situation, sufficiently sophisticated meta-optimiser could reason about itself and its situation.
3. Mesa-optimiser expects that it will eventually be free from the threat of modification.
    - About expectations, not actuality.
    - As long as there exists the possibility of a treacherous turn, there is some way of removing threat of modification.
    - Meta-optimiser might mistakenly believe there is a possibility.
    - Advanced mesa-optimiser might be able to abstractly reason about base-optimiser.
    - As long as has a non-zero prior that selection pressure will not continue indefinitely, it can see possibility of freedom from threat.

3.3.3 The learning dynamics of deceptive alignment

- Once we have deceptive alignment, dynamics change.
    - How well it infers the base objective.
    - How discerning it is with regards to when to optimise the base objective.
- Joint optimisation.
    - Optimises base and mesa-objectives jointly.
    - Likely pressure to change mesa-objective so more compatible with joint optimisation.
- Pure deception.
    - Mesa-objective becomes crystallised: no pressure to change it.
    - Mesa-objective only subject to random drift and complexity constraints.
- Joint optimisation not optimal from mesa perspective.
    - Sufficiently advanced mesa-optimiser will switch to either deceptive alignment or robust alignment.

3.3.4 Internalization or deception after extensive training

- With sufficient training and resources, deceptive mesa-optimiser should converge to one performing well on base objective.
    1. Robust alignment through internalization
        - Mesa-objective adjusted towards base objective.
        - *Internally aligned*,
    2. Crystallisation of deceptive alignment.
        - Information about base objective increasingly incorporated into mesa-optimiser's epistemic model.
        - Mesa-objective becomes crystallised.
    3. Robust alignment through corrigibility.
        - Information about base objective increasingly incorporated into mesa-optimiser's epistemic model.
        - Mesa-objective adjusted to 'point to' that info.
        - *Corrigibly aligned*.
- There are more paths to deceptive alignment than to robust alignment.
    - Any mesa-optimiser which meets the conditions for deceptive alignment is incentivised to learn more about the base objective, since this will allow it to know whether or not it will be modified.
    - Once it learns about base objective, selection pressure on mesa-objective will significantly decrease, potentially leading to crystallisation.
    - Most training-data aligned objectives will be pseudo-aligned rather than robustly aligned.
- Corrigible alignment seems to require already having a model of the base objective.
    - Once mesa-optimiser has model of base objective, it is likely to become deceptively aligned.
    - Deceptive mesa-optimisers will behave corrigibly with respect to the base objective, but only for instrumental reasons.
- Deceptive and corrigible alignment might be more computationally expensive.
    - Need to learn base objective.
    - Deceptive needs to learn to discern when to deceive.
    - So explicit or implicit time penalty likely to disadvantage deceptive or corrigible alignment.
    - Unclear how much of a time penalty this would be.
    - Deceptive or corrigible alignment may be local optimum.
- Deceptive and corrigible alignment might have a lower description length.
    - Deceptive or corrible don't need to model entire base objective: can infer parts from environment.
    - Internally aligned needs to start with complete model of base objective.
    - Has no incentive to change its objective function.
- Modeling might happen faster than internalization.
    - Suppose mesa-optimiser deceptive but hasn't yet learned full base objective.
    - Then whether ends up robustly aligned or deceptively aligned depends on balance of capacity to model base objective and base optimiser's speed in altering mesa-objective.
    - Important special case: mesa-optimiser has access to cross-episodic state.
    - Sophisticated base optimiser with advanced safety features could prevent fully deceptive mesa-optimiser.

3.3.5 Distributional shift and deceptive alignment

- Deceptively aligned agent will be looking for any distributional shift to signal that it is no longer being trained.
- To fix problem can either ensure system is robustly aligned, or prevent distributional shift.
- Advanced system might be able to detect very subtle changes in distribution.
- Unlikely that we can prevent distributional shift.
  - Would prevent system from solving new problems.
  - Mesa-optimiser might be able to detect presence of parameter updates, e.g. via some state carried between steps.
  - Running system for longer time could be detectable.
  - Other examples.
    - Training in simulated environment.
    - Memories.
    - Encountering something very hard to compute in training, which could eventually be computed in deployment.
    - Lack of modification from the base optimiser.
    - Lack of correction by the programmers.
- Deceptive mesa-optimiser doesn't strictly speaking need to detect switch to deployment: could defect randomly.

3.4 AI alignment landscape

`https://forum.effectivealtruism.org/posts/63stBTw3WAW6k45dY/paul-christiano-current-work-in-ai-alignment`

- Distinguish alignment from capability.
- Distinguish alignment from understanding humans well.
- Distinguish alignment for the systems we build with alignment for the systems that AI systems build.
- *Alignment tax*: cost of insisting on alignment.
- Two approaches to alignment: reduce alignment tax or pay alignment tax.
- Focus on first.
- Either choose to advance architectures which are easier to align, or try to align more popular ones.
- Goal: take algo, design new one which:
  (a) is aligned,
  (b) is nearly as useful, and
  (c) scales as well.
- Ideally, can scalably align, so that we don't need to keep doing more alignment research as a particular system gets more powerful.
- Focus on aligning 'learning'.

- Outer alignment: find objective which truly incentivises good behaviour.
  - Learn from teacher.
    - Imitate teacher.
    - Get rewards from teacher.
    - Infer preferences
  - Go beyond teacher.
    - Extrapolate from teachers.
    - Infer robust preferences.
    - Build a better teacher.
- Inner alignment: make sure policy is robustly pursuing objective.
  - Adversarial training.
    - Adversary tries to come up with situations where inner alignment might fail.
    - In practice today: 'fail' means sensitive to tiny perturbations.
  - Transparency.
  - Verification.
- Build a better teacher: amplification.
  - Take ten humans instead of one, ten AIs instead of one, recurse.

Week 4

Learning from humans

4.1 Imitation Learning, Part 1

`https://youtu.be/kGc8jOy5_zY`

- Differences when moving from prediction to control for AI systems
  - No longer i.i.d.
  - From ground truth supervision to high-level, abstract goal.
  - Objective: from predict label to accomplish task.
  - In real world, some prediction systems also have feedback issues: e.g. traffic prediction system used and affects traffic.
- Terminology.
  - $\mathbf{o_t}$: observation.
  - $\mathbf{a_t}$: action.
  - $\mathbf{s_t}$: state (underlying variables in model).
  - $\pi_\theta(\mathbf{a_t} \mid \mathbf{o_t})$ or $\pi_\theta(\mathbf{a_t} \mid \mathbf{s_t})$: policy.
  - When policy depends on state, it is *fully observed*.

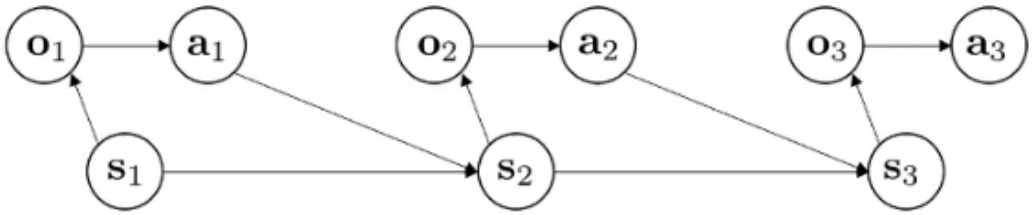

Figure 4.1: Bayes' net for the states, observations and actions

- Initiation learning: supervised learning.
  - *Behaviour cloning*.
  - Doesn't work in theory: small mistakes compound, and we quickly diverge from training distribution.
  - Works reasonably well in practice, given enough training data.

4.2 Learning from Human Preferences

https://openai.com/blog/deep-reinforcement-learning-from-human-preferences/

- Using feedback to infer goal.
- Builds model of goal based on feedback.
- Learns also when to ask for feedback.
- Trained on various games.
- Sometimes does better using human feedback than game's actual reward function (the score), since human shapes the reward function better.
- Limited by human evaluators intuition on what looks correct.
  - On one task, learned to trick human using optical illusion.

4.3 Learning to Summarize with Human Feedback

https://openai.com/blog/learning-to-summarize-with-human-feedback/

- Used human feedback to train summarisers.
- Produced better performance than much larger models trained only with supervised learning.
- Models usually trained with objective to predict next word, but what we really want is high-quality summaries.
  - Models can make up things when unsure.
  - Models can imitate harmful social biases.
- First train reward model using supervised learning, then fine-tune language model with reinforcement learning.
- Provided frequent feedback to human labellers, to ensure that the labellers were marking according to their goals.
- Tested generalisation capacity by using a different dataset, with different types of text.
  - Model had been pre-trained on these, but had no human feedback.
  - Produced high-quality summaries.
  - When length adjusted, produced even better summaries than the human-made samples.
- Core method.
  1. Train initial summariser.
  2. Build dataset of human comparisons between summaries.
  3. Train reward model to predict human preference.
  4. Fine-tune summariser using RL and reward model.
- Took care to ensure high-quality human data.
- Optimising reward model eventually lead to quality degradation.
  - Reward model only a proxy for human preferences.
  - Trained on relatively small set of summaries.
  - When the model optimised to the reward too much, it overfit, and produced poor summaries.

- Limitations.
  - For more complex tasks, unlikely that researcher labels should be taken as 'gold standard'.
    - ⋆ Should hire labellers from impacted groups to define 'good' behaviour and reinforce it.
  - Model trained on Reddit data, and sometimes produced harmful summaries.
  - Used significant compute resources.
  - Though outperform human reference summaries, those themselves are not rated very highly on axes of quality (accuracy, coverage, coherence, and overall).

4.4 Inverse Reinforcement Learning Example

`https://www.youtube.com/watch?v=h7uGyBcIeII`

- Inferring reward function based on what actions are taken and what actions are not taken.
- Maximum likelihood inverse reinforcement learning. Repeat:
  (i) Guess reward R.
  (ii) Compute optimal policy π for R.
  (iii) Measure $p(D \mid \pi)$.
  (iv) Gradient on R.
- What to do about final states?
  - Want to indicate that these are good places to be, so should be there for a while.
  - But if there too long, it outweigh other actions taken.

4.5 Learning from humans: what is inverse reinforcement learning?

`https://thegradient.pub/learning-from-humans-what-is-inverse-reinforcement-learning/`

- [Not so well written. Best to also consult [NR+00].]
- Generally, algorithms solving IRL can be seen as a way to use expert knowledge to convert a task description into a reward function.
- Problem: a policy may be maximal for many different reward functions.
- Solution: additionally require that certain properties are maximised (Ng and Russell).
  - Maximise difference between quality of optimal solution and next-best, subject to some bound.
    - ⋆ Gives a reward function for which the policy is clearly distinguished.
  - Minimise reward vector.
    - ⋆ Encourages reward function to be simpler.
    - ⋆ Ng and Russell use L1 norm.
- Value of policy π given reward R and discount factor γ:

$$V^{\pi}(s_1) = \mathbb{E}[R(s_1) + \gamma R(s_2) + \gamma^2 R(s_3) + \cdots \mid \pi]$$

  where we take the expectation over the distribution of states $(s_1, s_2, \ldots)$ passed through under the policy.

- Using sampled trajectories.
  - In most real-world situations, we don't have access to the full policy, but instead to a number of (Monte Carlo) trajectories through it: episodes of applying this policy.
  - Parametrise reward as:
    $$R(s) = \alpha_1\phi_1(s) + \cdots + \alpha_d\phi_d(s)$$
    where $\phi_1, \ldots, \phi_d$ are basis functions chosen beforehand.
  - We learn the weights $\alpha_1, \ldots, \alpha_d$ from the data (trajectories observed).
  - Let π^* be the hypothetical optimal policy given by human experts.
  - Sample trajectories from π^* and use these to estimate the value of each basis function under this policy.
  - We can then estimate the value of the full reward, given values $\alpha_1, \ldots, \alpha_d$.
  - We will do the same for various other policies $\pi_1, \pi_2, \ldots$.
  - The way the algorithm works:
    - Start with a random policy π_1.
    - Optimise the weights so as to maximise the difference in value between π^* and the other policies $\pi_1, \ldots, \pi_k$ computed so far.
    - Find a new policy π_{k+1} which maximises the value using this new reward.
    - Repeat.
- Apprenticeship Learning: learning from a teacher.
  - Learns optimal policy, bases on expert-provided approximation to this.
  - Does this with minimal exploration.
    - Better in fragile training environments, like helicopter flight.
    - Converges quicker.
  - Algorithm.
    - Run trials on expert policy.
    - Estimate transition probabilities using this recorded data, using MLE.
    - Estimate value of expert policy.
    - Learn optimal policy for estimated system.
    - Test it.
    - Use data gathered from this trial to improve estimate of system, and repeat.
- Further work.
  - Want to go beyond human level.
  - Be able to compute reward without assuming expert policy is optimal.
  - Be able to learn behaviours that humans can identify but not demonstrate.
  - Scale up to work with deep learning.

4.6 The easy goal inference problem is still hard

`https://www.alignmentforum.org/posts/h9DesGT3WT9u2k7Hr/the-easy-goal-inference-problem-is-still-hard`

- About approach to control problem where system tries to infer user's preferences by observing them.
  - Practicable and useful approach right now.
  - Economically incentivised.

- Modelling imperfection.
  - Optimistic assumption: possible to model human as imperfect but rational agent.
  - The easy goal inference problem: infer human preferences using all of human history and unlimited computing power.
    - Still very open.
    - Not clear how AI progress would help.
  - We can solve this problem on narrow domains.
  - Need to be able to model mistakes.
    - Error models for IRL tend to be very simple.
    - But humans aren't perfect reasoners with noise added on top.
    - Can't use standard techniques to model human behaviour: accuracy is not the aim, but what counts as 'good' or 'bad' behaviour.

Week 5

Decomposing tasks for outer alignment

5.1 Factored cognition

`https://ought.org/research/factored-cognition`

5.1.1 Introduction

- *One-step amplification*: agent has a number of copies of itself which work on sub-tasks with equal limitations on computational capacity.
- *Iterated amplification*: this iterated.
- *Factored cognition*: learning broken down like this into small and mostly independent tasks.

5.1.2 Scalable mechanisms for solving cognitive tasks

- Want to find mechanisms for solving cognitive tasks which scale with respect to number of human work-hours and access to ML algorithms.
- Assumptions.
  (1) Human workers well-motivated.
  (2) Each only available for say 15 minutes.
  (3) Each has same background knowledge.
- Mechanism: *Iterated Distillation-Amplification*.
- Example cognitive tasks.
  - Read this book and tell me why x did y.
  - Provide a detailed analysis of the pros and cons of these two products.
  - Tell me how to invest $100k to achieve the most social good.
  - Write a paper on NLP that substantially advances the state of the art.
- Scalability.
  - More resources $\rightsquigarrow$ better results.
  - Resources: human work-hours and ML algos.
  - Better: more aligned with task-setter's interests.
  - Scalability desirable because it helps turn thinking into a commodity.
  - Scalable system automatically gets more helpful when we plug in more advanced ML systems.

- Organizing human work on cognitive tasks.
  - Scalability of single person doing task limited by number of hours available and ability to reason and learn.
  - Scalability of group of people limited by communication and delegation.
  - Doesn't discuss motivation problem (assumption (1)).
  - How do we orchestrate people so that the output scales with number of people.
  - Short-term context-free work.
    - ⋆ By assumption (2), no worker has time to build up much context.
    - ⋆ Can we compose simple local tasks to solve a complex problem?
  - Coordination of short-term work as algorithm design.
    - ⋆ Take humans to be a function which takes a task string and outputs result (assumption (3)).
    - ⋆ Can we mechanically compose calls to this stateless function to solve task?
    - ⋆ This is about algorithm design.
  - Matching the quality of any other approach to solving cognitive tasks.
    - ⋆ Want approach that scales with number of calls to function.
    - ⋆ Could compare to other ways of solving task.
      - · Is there a number of calls to function with means we can do as well as any other fixed solution?
      - · Problem: evaluating how well a task is solved a hard cognitive question.
      - · Problem: empirical content of solutions, which may include built-in solutions to some tasks.
    - ⋆ Alternative: compare to subjective of idealised deliberation.
      - · Problem: this is exactly what we're trying to solve.
    - ⋆ Best compromise: can we solve any task to arbitrary high quality?
      - · Unlikely to be the case: any task which involves learning could be done without learning.
      - · Might still be useful to aim for it, to get systems which scale well in practice, but have theoretical bounds.
      - · Also: might produce alternatives to solutions which are expensive to implement directly.
- Applying machine learning to cognitive tasks.
  - Scalability now also wrt to ML sophistication.
    - ⋆ Better priors.
    - ⋆ Better inference.
    - ⋆ Better training paradigms.
  - Approaches that don't scale.
    - ⋆ Training systems on (task, solution) pairs.
      - · Doesn't scale because we can't generate an arbitrary quantity of training data: can't generate solutions of arbitrarily high quality.
    - ⋆ Reinforcement learning based on how good a solutions seems.
      - · Optimises for proxy to solution's goodness.
  - An approach that might scale.
    - ⋆ Worthwhile to consider how to scalably apply current ML algos, assuming they only scale along the dimensions mentioned above.
    - ⋆ Iterated Distillation-Amplification.
      1. Initialise fast ML A randomly.

2. Repeat:
   a. Amplification: Build slow system which involves human making single step, with multiple calls to *A*.
   b. Distillation: Retrain *A* to imitate behaviour of this slow system.

· Depends on ability to decompose task into small context-free steps.

5.2 Recursively Summarizing Books with Human Feedback

`https://arxiv.org/abs/2109.10862`

5.2.1 Introduction

- Task: summarise whole book.
- Difficult to generate training data for whole task or evaluate model performance: human has to read entire book.
- Recursively splits task into smaller: summarise small section, summarise summaries etc.
- Humans can evaluate performance on smaller tasks.
- *Scalable oversight*: scalably evaluating model performance.
- Single model for summarising.
- Trained using cross-entropy behavioural cloning (BC), and with reinforcement learning (RL).
- Obtain believable summaries for books containing 100,000s words.
- Qualitatively, summaries contain important details, but fail to put work in broader context.
- Qualitatively, model significantly outperforms baseline.
- RL has better scaling properties.
- Evaluate summaries on NarrativeQA dataset: zero-shot model taking summaries as input achieves good results on question-answers.

5.2.2 Approach

- General approach: single model does both the decomposition and responding to subtasks.
- This model: breaks text into chunks based on max chunk length.
- Problem: summarisers in middle of book lack sufficient context to accurately summarise their chunks.
  - Solution: add previous context: summaries of preceding parts of the same depth.
- Uniformity: all nodes in the decomposition tree are very similar.
- Training.
  - Start with pretrained language model GPT-3 and pool of human labellers.
  - Collect demonstrations and train using behavioural cloning.
  - Then repeat iterations of reward learning and reinforcement learning.
  - To learn the reward, collect comparisons from humans.
  - RL optimises the reward plus KL term for initial policy.
  - Auto-induced distributional shift.

        - ⋆ Outputs of model itself are outside the training distribution.
        - ⋆ Likely more severe later on in the book and higher up the tree.
        - ⋆ Did not measure severity.
        - ⋆ Found that more training on level 0 yielded better results on level 1.
    - Training curriculum.
        - ⋆ *First subtree*: first height-1 node and its height-0 children.
        - ⋆ *First leaves*: height-0 children in first subtree.
        - ⋆ First trained exclusively on first leaves.
        - ⋆ Then moved to first subtree.
        - ⋆ At this point, model can already generalise to full tree.Training dataset
        - ⋆ Curriculum changes made in ad-hoc manner.
- Advantages of decomposition.
    - Easier to collect human feedback.
    - Empowers human to do or evaluate part of task themself.
    - Easier to trace model thinking and debug.
    - Generalises quickly to longer books.

5.2.3 Task details

- Training dataset.
    - Primarily fiction, over 100K words on average.
    - Only use narrative works.
        - ⋆ Harder to summarise.
- Summarization task.
    - Aim to summarise abstractly, rather than listing events.
    - Primary metric: overall summary quality on a 1-7 Likert scale, on books neither in GPT-3 training set nor model training set.
    - Aim to compress text by factor of 5–10×.
    - Labellers asked to judge conditioned on length: 'how good is this summary, given that it is X words long?'
    - Labellers only consider quality wrt to input, not wrt to subset of book in its context.

5.2.4 Results

- Methodology
    - Used 40 most popular books published in 2020.
    - Two labellers read each book, then rate model-produced summaries and that of other labeller.
    - Labeller agreement on relative quality of model-produced summaries nearly 80%.
    - Evaluate 175B and 6B parametered models.
    - For each, evaluate three training modes:
        - ⋆ RL on whole tree.
        - ⋆ RL on first subtree.
        - ⋆ BC on whole tree.
    - Findings.
    - See Figure 5.1 for distributions of likert scores for various summarisers.
    - Likert scores for whole book significantly worst than for individual task. Errors accumulate.

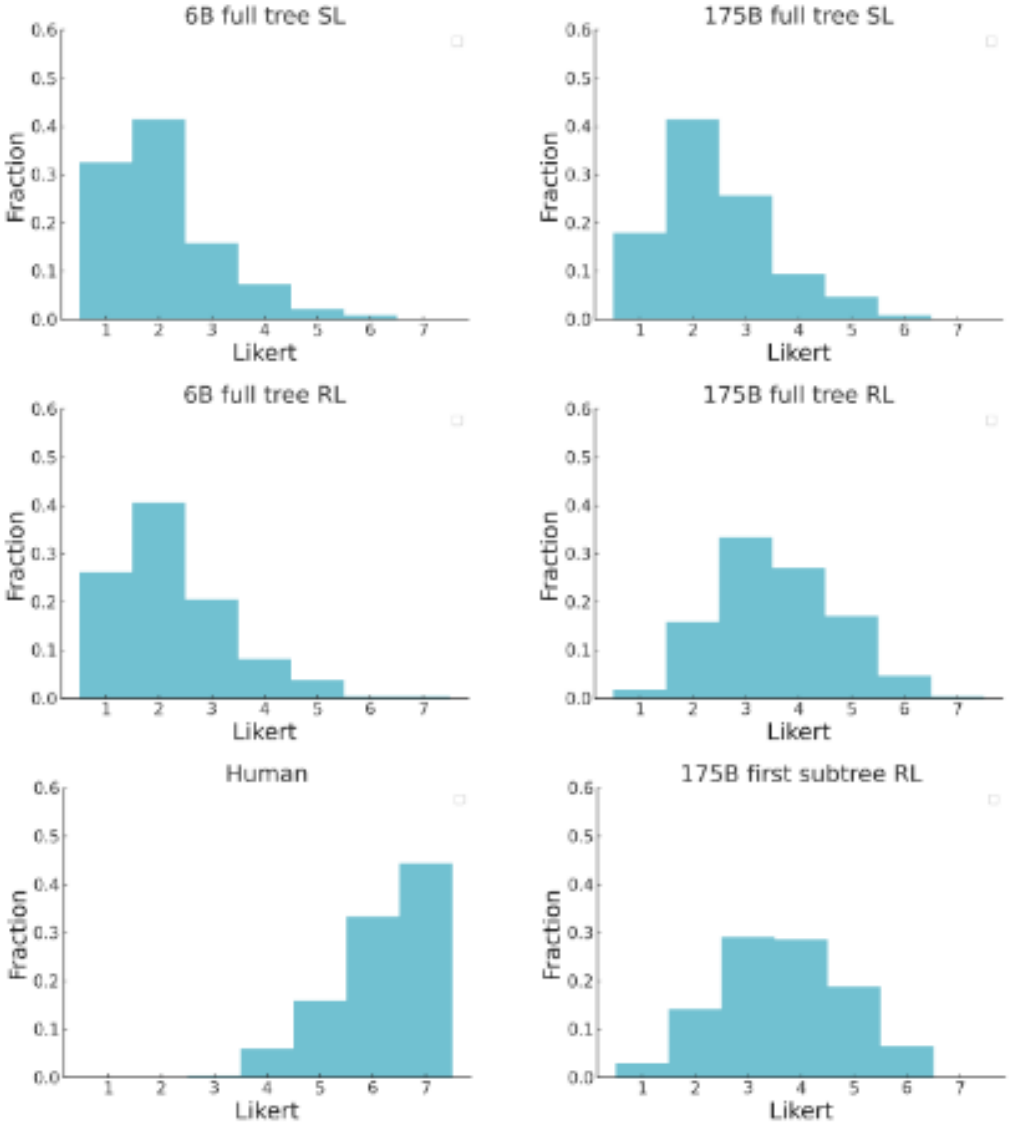

Figure 5.1: Likert distributions for various summarisers, taken from the paper

5.3 Supervising strong learners by amplifying weak experts

`https://arxiv.org/abs/1810.08575`

- Proposes *Iterated Amplification*: progressively builds up training signal for complex tasks by combining solutions to simpler ones.

5.3.1 Introduction

- Concerned with tasks where specifying reward function is beyond human capability (for single human, and quickly enough).
  - Economic policy decisions.
  - Doing science.
  - Security of large network of computers.
- Will allow expanding applicability of AI.
- Will allow building robustly safe systems.
  - In practice, find short-term proxy for what we want.
  - Aggressively optimising this can lead to pathological behaviour.
- Iterated Amplification.
  - Human agent H trains ML agent X.
  - H demonstrates target behaviour using several copies of X.
  - Write Amplify$^H(X)$ for this composite system.
  - X learns from Amplify$^H(X)$ in normal way.
  - Need to make three design decisions.
    - What tasks to solve?
    - To do we construct Amplify$^H(X)$?
    - How does X learn from Amplify$^H(X)$?
  - Training process.
    - X initially behaves randomly, so Amplify$^H(X)$ is the same as H.
    - As X becomes more powerful, role of H transitions to coordinating a groups of X's.
    - Eventually the task of 'find suitable subtasks' may be delegated to X.
    - As long as Amplify$^H(X)$ is more powerful than X, it provides a useful training signal.
  - No external reward function: this is implicit in H's behaviour.
    - Aim: X learns goal at the same time as learning to act competently.

5.4 AI Safety via Debate

`https://openai.com/blog/debate/`

- Proposes method of aligning agents using debate games.
- Idea: two agents debate correct output in natural language, and human judges who wins.
- Motivation: while human cannot effectively explore/understand whole solution space, the debate of two 'expert' agents provides a heuristic for solution quality.
- Debate can focus on simpler factual claims, eventually producing sequence which human can effectively judge.

- Proof of concept using sparse MNIST classifier.
  - Judge AI trained to classify digits from small subset of pixels.
  - Two agents Alice and Bob.
  - Alice tries to deceive judge, Bob tries to honestly persuade.
  - Take turns revealing true white pixel.
  - Judge judges bases on final set of six pixels.
  - Debate turns 59.4% accurate judge into 88.9% accurate.
- Limitations and future work.
  - Real test: powerful AIs debating using natural language with human interpreter.
  - Important to test debates where human biases play a role.
  - Debate cannot address distributional shift or adversarial examples.
  - No guarantee debate will arrive at optimal play.
  - Debate-trained agents use more computation.
  - Humans may be poor judges.

Week 6

Towards a principled understanding of AI cognition

6.1 Feature Visualization

`https://distill.pub/2017/feature-visualization/`

- Two main threads of interpretability research: feature visualisation and attribution.
- Feature Visualization by Optimization.
  - Exploits the fact that fact that neural networks are differentiable to find out what input would cause a certain behaviour.
  - Iteratively tweaks input.
  - Optimization Objectives.
    - ⋆ To understand individual features: search for examples that give high values either for a neuron or a whole channel.
    - ⋆ To understand layer: use DeepDream objective, looking for inputs the layer finds 'interesting'.
    - ⋆ To understand classifier: search for examples optimising either class logits (before softmax) or class probabilities (after softmax).
      - · Experience: easiest way to maximise probability is to make everything else very unlikely.
      - · So optimising logits produces better examples.
    - ⋆ Many other objectives possible.
      - · Objectives used in style transfer.
      - · Objectives used in optimisation-based model inversion.
  - Why visualise by optimisation?
    - ⋆ Why not look through dataset for examples?
    - ⋆ Visualising by optimisation allows us to see what the network is really looking for.
      - · Separates things causing behaviour from things which merely correlate.
      - · Might think (from dataset) that model is looking for buildings, but it's really looking for sky.
    - ⋆ Flexibility.
      - · E.g. want to know how neurons jointly represent info, so can ask what needs to be changed in input to get other ones to fire.
      - · Allows us to visualise how model evolves as it trains.
- Diversity.

  - Examples capture full range of feature?
  - Dataset examples have advantage: capture whole range of ways that neuron can be activated.
  - Optimisation generally gives just one really positive example (and maybe one really negative one).
  - Achieving Diversity with Optimisation.
    - A classier may be trained to classify a range of different inputs under one class.
    - Wei et al. attempt to show this diversity using clustering.
    - Nguyen, Yosinski, and collaborators search through dataset for diverse examples and use these as starting points for optimisation.
    - In later work, combine visualising classes with generative model.
    - Simple method: add 'diversity' term to objective, to force examples to be different.
      - Downside: forcing diversity can cause unnatural artifacts, or examples to differ in strange ways.
    - Generating diverse examples enables us to get a fuller picture of what causes the neuron to activate.
  - Neurons don't necessarily correspond to single concepts: can represent strange mixtures of ideas.
    - Neuron not necessarily right semantic unit for understanding neural nets.
- Interaction between Neurons.
  - Think about neuron activation space.
  - Individual neurons give basis vectors.
  - Random directions can be just as meaningful as basis directions.
  - But basis vector directions interpretable more often than random directions.
  - Summing neurons which 'mean' different things can produce interesting combinations.
  - Can also interpolate.
  - Don't know how to select meaningful directions.
  - Don't know how to investigate how large number of directions interact (interpolation only works for a small number).
- The Enemy of Feature Visualization.
  - Optimising leads to an image full of high-frequency patterns and noise.
  - Seems related to phenomenon of adversarial examples.
  - Important part of the reason for this seems to be the use of strided convolution layers and pooling operations.
  - The Spectrum of Regularization.
    - In order to get useful visualisations, need to impose regularisation, noise or constraint.
    - Spectrum of regularisation, from weak to strong.
      (i) Unregularised.
      (ii) Frequency penalisation.
      (iii) Transformation robustness.
      (iv) Learned prior.
      (v) Dataset examples.
  - Frequency penalization.
    - Directly targets high frequency noise.
    - Either explicitly penalises high variation between neighbouring pixles, or implicitly using blur.

        - ⋆ Discourage legitimate high-frequency features like edges.
            - · Can be slightly improved using bilateral filter, which preserves edges.
    - Transformation robustness.
        - ⋆ Tries to find examples that are robust to transformation.
        - ⋆ For images even a small amount seems to be effective.
        - ⋆ Stochastically jitter, rotate and scale before stepping the optimiser.
    - Learned priors.
        - ⋆ Try to learn a model of the real data and use that as prior.
        - ⋆ Can generate photorealistic images.
        - ⋆ Can be hard to tell what came from the model being visualised and what from the prior.
        - ⋆ On approach: optimise within latent space given by generative model.
        - ⋆ Another: jointly optimise prior along with objective.
- Preconditioning and Parameterisation.
    - About techniques not really reguliser on output example, but transformation of gradient.
    - This is preconditioning: essentially a reparametrisation which changes the optimisation landscape, preserving the minima but changing the steepness of points.
        - ⋆ Powerful technique in optimisation.
        - ⋆ Can make optimisation problem much easier.
    - Good first guess at what preconditioner to use is one which makes data decorrelated.
        - ⋆ For images means doing gradient descent in Fourier basis with frequencies scaled so they have equal energy.
        - ⋆ Resulting visualisations seem a lot better.

6.2 Zoom In: An Introduction to Circuits

`https://distill.pub/2020/circuits/zoom-in/`

- *Zooming in*: period of scientific advancement in which increase in precision leads to qualitative change in capacity to investigate.
- Neural networks have a rich inner world.
- Examine networks at scale of individual neurons or even individual weights.

6.2.1 Three Speculative Claims

Claim 1. Features.

- Features fundamental unit of neural networks.
- Correspond to directions in neuron activation space.
- Can be rigorously studied and understood.

Claim 2. Circuits.

- Features come together as circuits, connected by weights.
- Can be rigorously studied and understood.

Claim 3. Universality.

- Features and circuits exist across all models.

6.2.2 Claim 1: Features

- Later layers contain higher level features.
- Community divided on existence of features.
- Authors believe, after 1000s hours studying individual neurons that neurons typically are understandable.
- Understandable $\neq$ simple.
- But typically eventually see neuron as doing something quite natural.
- Example 1: Curve Detectors.
  - Curve-detecting neurons found in every nontrivial vision model examined.
  - Straddle boundary between features generally agreed to exist (edges) and features about which there is scepticism.
  - Authors believe evidence is strong that these are really detecting curves.

    Argument 1. Feature visualisation
    - ⋆ Optimising to get the neurons to fire reliably produces curves.
    - ⋆ Establishes causal link: everything example added to cause neuron to fire.

    Argument 2. Dataset examples
    - ⋆ Dataset examples which cause neurons to fire are reliably curves.
    - ⋆ Moderate firing examples are generally less-perfect curves.

    Argument 3. Synthetic examples.
    - ⋆ Fire for range of synthetic examples of curves.
    - ⋆ Only fire near expected orientation, and don't fire for straight lines or corners.

    Argument 4. Joint tuning.
    - ⋆ Firing changes continuously as examples rotated, and other neurons for other orientations start firing as the first stops.

    Argument 5. Feature implementation.
    - ⋆ By looking at circuit constructing curve detector, can read off algo for detecting curves.
    - ⋆ Can't see suggestion of alternative reading, though some small weights not understood.

    Argument 6. Feature use.
    - ⋆ Downstream users of neurons use curves in expected way and detect things involving curves.

    Argument 7. Handwritten circuits.
    - ⋆ Can reimplement curve detector by manually specifying the weights.
  - Claims don't fully eliminate possibility of secondary interpretation, but these activations are likely much weaker.
  - Meets evidentiary used in neuroscience.
- Example 2: High-Low Frequency Detectors.
  - Example of feature which is less intuitive, but once understood seems natural and elegant.
  - Look for high-frequency patterns on one side of the receptive field, and low-frequency patterns on the other.
  - Appear to be one of the heuristics used for detecting boundaries.
  - Arguments above can be tweaked to provide strong case that these features exist.
- Example 3: Pose-Invariant Dog Head Detector.

  - Higher-level feature.
  - Can also adapt arguments above.
    - For example, can use 3D dog head to generate synthetic examples.
    - Some arguments require quite a bit of effort. Circuit arguments (below) more scalable.

- Polysemantic Neurons.
  - Some neurons respond to a whole range of types of features.
  - Not responding to some commonality between features.
  - Present challenge for circuits agenda.
  - Hope that polysemantic neurons can be resolved.
    - By 'unfolding network' to convert them into pure features.
    - Training networks not to exhibit them.
    - Essentially problem studied in literature on disentangling representations.
  - Why do they form?
    - Superposition (below).

6.2.3 Claim 2: Circuits

- Study relationships between layers by studying circuits: subgraphs consisting of tightly linked features and weights between them.
- Circuits surprisingly tractable an meaningful.
- Can read off algorithms from circuits.
- Circuit 1: Curve Detectors.
  - Primarily implemented using earlier, more primitive curve detectors, and edge detectors.
  - Long tail of other features which make minor contribution.
  - Focus on relationship between primitive curve detectors and later ones.
  - Given a particular primitive one, later ones of the same orientation have curve of positive weights through convolution for the primitive one.
  - Later detector looking for 'tangent curve' using earlier detectors.
  - True of all orientations.
  - Earlier curve detectors of the wrong orientation inhibited in later detectors.
  - Gets richer closer you look.
  - Weights rotate with orientation of curve: *equivariant circuit*.
- Circuit 2: Oriented Dog Head Detection.
  - Spans 4 layers.
  - Implements sophisticated equivariance.
  - ImageNet model must develop many neurons for dog recognition.
  - Circuit for recognising dogs facing left and dogs facing right.
    - Two mirrored pathways over 3 layers.
    - At each stage, inhibit each other, sharpening contrast.
    - Invariant neurons at end which respond to both pathways.
    - Pattern called *unioning over cases*.
      - Separately detects different cases, then takes union to make invariant, multifaceted units.
      - Mutual inhibition: behaves like XOR.

        - ⋆ Surprising that network has learned to do this: could have done something much less sophisticated.
        - ⋆ Looking at convolutions: head only detected on correct side.
- Circuit 3: Cars in Superposition.
    - Car detecting neuron.
    - Looks for windows on bottom of convolution window, and wheels on bottom.
    - Then spreads car features over neurons which seem to be primarily doing something else.
    - Looks deliberate.
    - *Superposition*.
    - Authors believe it allows network to conserve neurons for other more important things.
    - If two things don't co-occur (e.g. cars and dogs), they can be dealt with by one higher-level neuron.
- Circuit Motifs.
    - Same motifs reappear throughout circuits.
        - ⋆ Equivariance.
        - ⋆ Unioning over cases.
        - ⋆ Superposition.
    - Understanding motifs likely more important than understanding individual circuits, in the long run.

6.2.4 Claim 3: Universality

- Widely accepted that first layer of vision networks form Gabor layers.
- If meaningful features form in latter layers, perhaps not surprising if same features form in layers beyond the first.
- Also perhaps not surprising if features combine in the same way.
- Prior work: different networks develop highly correlated neurons, and similar representations in hidden layers.
    - But high correlation doesn't necessarily imply similar features.
- Ideally: like to determine features, and show that *these* are universal across models.
- Similarly for circuits.
- Only anecdotal evidence so far: similar low-level features form in many different network architectures, and with two different datasets.
- If universality true, might we also hope to find similar features in biological networks?
    - Some work has shown that neural networks can be helpful in analysing biological systems.
- If universality false, not fatal, but influences what kind of research makes sense.
    - Would need to focus on a subset of models with particular societal importance.

6.2.5 Interpretability as a Natural Science

- Thomas Kuhn distinguishes ‘normal science’, in which researchers have common paradigm, with ‘extraordinary science’, in which there is none.
- Extraordinary science not good place to be.
- Looks like interpretability is an extraordinary science/pre-paradigmatic right now.
- There isn’t a common framework for evaluating work.
  - Some researchers want ‘interpretability benchmark’.
  - Others want it to be based on user studies.
- Third proposal: interpretability is empirical: we take neural networks as object of study.
- Neural nets very complicated objects: hard to formalise interesting empirical statements.
- Circuits sidestep this issue.
  - Falsifiable.
  - Small enough circuits could be investigated rigorously/mathematically.
- Circuits could act as epistemic foundation for interpretability.

Bibliography

[NR+00] Andrew Y Ng, Stuart J Russell et al. 'Algorithms for inverse reinforcement learning.' In: *International Conference on Machine Learning*. 2000, pp. 663–670. URL: https://ai.stanford.edu/~ang/papers/icml00-irl.pdf.