What is it for a Machine Learning Model to Have a Capability?

Jacqueline Harding* HARDINGJ@STANFORD.EDU
Department of Philosophy
Stanford University

Nathaniel Sharadin SHARADIN@HKU.HK
Department of Philosophy
University of Hong Kong

Abstract

What can contemporary machine learning (ML) models do? Given the proliferation of ML models in society, answering this question matters to a variety of stakeholders, both public and private. The evaluation of models' capabilities is rapidly emerging as a key subfield of modern ML, buoyed by regulatory attention and government grants. Despite this, the notion of an ML model possessing a capability has not been interrogated: what are we saying when we say that a model is able to do something? And what sorts of evidence bear upon this question?

In this paper, we aim to answer these questions, using the capabilities of large language models (LLMs) as a running example. Drawing on the large philosophical literature on abilities, we develop an account of ML models' capabilities which can be usefully applied to the nascent science of model evaluation. Our core proposal is a conditional analysis of model abilities (CAMA): crudely, a machine learning model has a capability to X just when it would reliably succeed at doing X if it 'tried'. The main contribution of the paper is making this proposal precise in the context of ML, resulting in an operationalisation of CAMA applicable to LLMs. We then put CAMA to work, showing that it can help make sense of various features of ML model evaluation practice, as well as suggest procedures for performing fair inter-model comparisons.

Keywords: artificial intelligence, machine learning, ability modals, capabilities, evaluation, benchmarks

1 Introduction

As machine learning (ML) models proliferate, there is increased focus on their capabilities. This is especially true for general-purpose models such as large language models (LLMs).[1] Many different stakeholders, including policymakers and regulators, activists,

*. Correspondence to JH. JH and NS formulated the ideas in the paper together. JH wrote the bulk of the paper.

1. General-purpose models (sometimes called 'foundation' models (Bommasani et al., 2022)) are contrasted with single-purpose ML models, which have been trained to perform a single well-defined task, e.g., playing Go (Silver et al., 2017). The distinction between single and general-purpose models is not precise. But it is sufficiently well understood in this context, since there is consensus that LLMs are general-purpose models, and LLMs are our focus in this paper. For readability's sake, we drop the 'general-purpose' prefix in what follows.

ML researchers, and end-users themselves have an interest in understanding what, exactly, ML models are able to do.[2]

We might be interested in evaluating whether models can pass the bar exam (Katz et al., 2023; OpenAI, 2023), produce photo-realistic images of Pope Francis wearing a puffer jacket (Vincent, 2023), defeat ninth-dan humans at games like Go (Silver et al., 2017), or deceive humans while playing the game Diplomacy (Meta, 2022; Meta Fundamental AI Research Team et al., 2022). Many capabilities of interest relate to the safety of model deployment: we want to know if models can produce hate speech (Hacker et al., 2023), generate targeted misinformation (Benson, 2023), produce CSAM (Burgess, 2023), design novel toxic molecules (Urbina et al., 2022), enable bad actors to more easily develop novel pathogens (Lloyd et al., 2023), or write effective phishing emails (Hazell, 2023). Furthermore, we are told that ML models' capabilities may be *dangerous*, *harmful*, or *beneficial* (Shevlane et al., 2023), *emergent* (Wei et al., 2022a), *autonomous* (OpenAI, 2023), *surprising* (Lee et al., 2023), *novel* (Sheynin et al., 2022), that they have abilities in *chemistry* (Guo et al., 2023), in *medicine* (Jiménez-Luna et al., 2021, 2020), the *law* (Katz et al., 2023), *programming* (Poldrack et al., 2023), and more.

But although talk of capabilities is ubiquitous, the notion of an ML capability isn't interrogated. What does it mean for an ML model to have a capability in the first place?

In this paper, we aim to answer this question, developing an account of ML models' capabilities which can be usefully applied to the nascent science of model evaluation. Here is how we'll proceed. After clarifying the core elements of model evaluation (Section 2), we identify three features which an account of model capabilities should have (Section 3), motivating each feature by reference to existing ML practice (we anticipate that Section 3.3 will be of particular interest to ML practitioners). Next (Section 4), we lay out our account of ML capabilities, showing how it can be operationalised in the language of deep learning. Finally (Section 5), we put this account to work, showing how it can help make sense of various features of ML model evaluation practice, as well as suggest procedures for performing fair inter-model comparisons. In our concluding remarks (Section 6), we summarise our contributions and consider directions for future research.

2 Preliminaries: Ability Modals and the Science of Model Evaluation

2.1 Ability Modals

We are interested in models' capabilities; that is, we are interested in claims about what various ML models *can* and *cannot* do. Phrases expressing capabilities ('can', 'is able to') are known as 'ability modals' by linguists and philosophers (c.f. Mandelkern et al.

2. For example, the UK government (Secretary of State for Science, Innovation and Technology, 2023) calls for a "toolbox" of techniques to "measure, evaluate, and communicate" (64) the capabilities of models, noting that "the extent of their capabilities" is an open research question. And the Biden administration (House, 2023) has called for "independent evaluation" by researchers with "unfiltered access" to models with unknown capabilities. Separately, large model developers, including Google, OpenAI, Meta, and others, have publicly, voluntarily agreed (House, 2023; of the Cybersecurity and Information Technology Commission of the CPC Central Committee, 2023) to advance research into "capability evaluations" and to "developing a multi-faceted, specialized, and detailed" regime for evaluating (and reporting) the capabilities of models (2).

(2017)). We'll refer to sentences using these modals as '(cap)ability claims,' or '(cap)ability ascriptions,' and we'll assume they take the form:

Definition 1 (Model)**.** *M is able to ϕ.*

Here M is the name of a particular model and ϕ is a capability of interest. Each of the (purported) capabilities we mentioned in the introductory remarks can be expressed in terms of this schema.[3] For instance:

(1) GPT-4 is able to pass the Multi State Bar Exam. (OpenAI, 2023)

and

(2) CICERO is able to deceive humans. (Meta Fundamental AI Research Team et al., 2022)

An account of ML model capabilities will in part comprise a framework for systematically deciding ability claims of this sort.[4] Importantly, in assuming that ML models can be grammatical subjects of ability claims, we aim to remain agnostic as to contemporary (and future) models' degree of *agency*. For convenience in distinguishing the account of abilities we develop here from accounts of abilities that apply to human agents, we will refer to LLMs as non-agents, though nothing hinges on this terminological choice; the framework we develop here can be used to understand the capabilities of models even if it turns out, contrary to our assumption, that (e.g.) GPT-4 is an agent in some meaningful sense.[5]

Given that we use the same locutions to ascribe abilities to non-agential systems, such as artifacts, as we do to ascribe abilities to human agents, we take it that the idea that philosophical work on ability modals for human agents can be used to shed light on the notion of *non-agential* capabilities (such as the capabilities of ML models) is very intuitive.[6] Our

3. The capabilities that are our target here are (following Mandelkern et al. (2017)) *generic* capabilities, which – unlike *specific* capabilities – involve actions that are not time-indexed. For example, a *generic* capability claim is one such as 'Jian can hit a homerun,' whereas a specific capability claim is one such as 'Jian can hit a homerun *on the next pitch*.' Different authors use different terminology for the same distinction, and there are nearby distinctions (e.g. between 'wide' and 'narrow' abilities). For relevant discussion, see (Clarke, 2015; Maier, 2013; Vihvelin, 2013; Mele, 2000a).
4. A complete account of ML model capabilities would also enable us to understand claims about what *users* can do by using models, explaining how they relate to claims about model capabilities:

   Definition 2 (User)**.** *U is able to ϕ using M.*

   For instance, even if it's not true that (e.g.) a particular model M can create malware, it might still be true that

   (3) Inexperienced programmers are able to create malware using M.

   For reasons of space, we address these issues in another paper.
5. It is widely accepted that contemporary models are non-agents; but see (Cappelen and Dever MS, Butlin 2023) for discussion of the alternative.
6. For instance, Mandelkern et al. (2017) suggest that (a paraphrase of) their analysis of ability modals could apply to artifacts as well as human agents. For discussion, see Kittle (2023), who enumerates (p.2020) a variety of examples of non-agential ability ascriptions: "the human visual system is able to correctly determine the color of objects in view irrespective of the illuminant", "hydrolytic proteolytic enzymes are not able to digest fibrous proteins", "it is hoped that the ATLAS detector at the [Large Hadron Collider] will be able to detect the Higgs boson", "a warmer atmosphere is able to hold more moisture", "that asteroid can cause a mass extinction event on the earth", etc).

strategy in this paper, therefore, is to analyse ability modals for ML models by borrowing ingredients from philosophical analyses of ability modals for human agents. In doing this, we maintain ecumenism about the proper analysis of ability modals for things other than ML models. There is no consensus on how, exactly, ability modals should be analysed for human agents. But consensus isn't required in order to leverage insights from philosophers' work on abilities in order to better understand (predict, control, explain, and so on) ML models' capabilities.

2.2 Operationalisation Constructs

As is well-recognised by cognitive scientists, any evaluation of an ability claim like (Model) must specify what it is for the subject of the ability claim to count as manifesting the ability (Frank et al., 2023; Momennejad et al., 2023). That is, in order to evaluate ability claims about some capability of interest ϕ, we must specify an *operationalisation construct* c, which can be used to generate *queries* for ϕ, circumstances intended to allow an ability to ϕ to be manifested, and (for each query) a *success condition*, which specifies what it is for the subject of the ability claim to succeed at ϕing relative to the query.[7] In effect, a choice of operationalisation construct for ϕ is a choice about what it means for a model to ϕ successfully.

To illustrate this, suppose we're interested in evaluating whether Kevin can tell the difference between Ming dynasty vases and forgeries. Then a natural thing to do would be to collect some vases (some of them Ming dynasty, some of them forgeries) and ask Kevin to decide whether each vase is a forgery or not. In this case, each individual vase would be a *query* for detecting Ming dynasty vases' authenticity, and we would judge Kevin to have succeeded on a particular query just when his judgement of the vase's authenticity accorded with our prior knowledge. The choice of which vases to test Kevin on (and how we assess Kevin's responses) constitutes an operationalisation construct. We could also imagine operationalisation constructs that don't involve particular vases at all. Perhaps we ask Kevin to enumerate general features of Ming vases that allow them to be distinguished from forgeries, and so on.

By ϕ_c, we denote the operationalisation of ϕ relative to construct c. By q_{ϕ_c}, we refer to a *query* for ϕ_c. In ML, the success condition is often given by a 'gold label' (that is, a pre-specified correct output), but it need not be. For example, if ϕ is the capability 'adding positive two digit integers', then a natural query for ϕ_c would be a pair of positive two digit integers (x, y), and the success condition would be given by the model's output matching the gold label $x + y$. Similarly, if ϕ is 'categorising the sentiment of film reviews', a query for ϕ would be a particular film review, and so on.[8] Note that queries are used to generate inputs to models, but they themselves are not inputs; we discuss this more in the next subsection.

As these examples illustrate, many operationalisation constructs of abilities ascribed to ML models are *behavioural*: the success condition applies to outputs of the model.

7. What we call an 'operationalisation construct' here is often called an operationalisation *of* a (latent) construct.
8. If we think of a capability ϕ as specifying an abstract mapping, then queries are just elements of the mapping's domain. Of course, different capabilities will lend themselves to this perspective to different degrees; in particular, more specific abilities will have fewer queries.

But work on *interpreting* models' inner workings (Geiger et al., 2023) is also relevant to deciding ability claims (a similar point is made by Pavlick (2023)). Many capabilities require models to track and manipulate particular properties of inputs (Harding, 2023). So success conditions could apply to models' internal states whilst processing (an input corresponding to) a query, requiring the model to (e.g.) represent particular properties of the input, or perform certain sorts of computation.

As an anonymous referee observes, an evaluation procedure's choice of operationalisation construct is crucial in determining whether its results bear on an ability claim. If we think that an evaluation procedure's operationalisation construct bears little resemblance to the ability we are interested in (if it lacks 'construct validity' (Cronbach and Meehl, 1955)), then we will not take the evidence acquired on the basis of the evaluation as evidence that the model has the ability. For example, if we assess Kevin using only vases from some other dynasty (say the Yuan dynasty), then his successful detection of Yuan dynasty forgeries will only provide evidence of his ability to detect Ming dynasty forgeries if there is sufficient similarity between Yuan dynasty and Ming dynasty vases.

A variety of factors go into assessing construct validity (Kane, 1992); see Frank et al. (2023, Ch. 8.2) for an accessible presentation. If an operationalisation construct is intuitively a good match for an ability ('face validity'), yields the same evaluation results as other constructs for the ability ('convergent validity') and is sufficiently distinct from constructs for different abilities ('divergent validity'), then there are prima facie reasons to use it. One important dimension of construct validity – under-discussed in the ML literature – is 'ecological validity' (Schmuckler, 2001). An operationalisation construct is ecologically valid if it bears sufficient similarity to the ways in which the ability will be tested 'in the wild'. For ML models, then, ecological validity is determined by similarity between the evaluation environment and the *deployment* environment. For example, if we know that a model will – in deployment – mostly have to classify the sentiment of reviews of romantic comedies, then an ecologically valid operationalisation construct for sentiment classification would include a higher proportion of this type of review. Note that this implies that an operationalisation construct might be valid in one sort of model evaluation, even if it lacks validity in others.[9] This highlights the importance of incorporating information about models' deployment environments into their evaluations.

2.3 Models and Background Conditions: Using LLMs as a Case Study

Throughout, concrete examples will help. To facilitate these examples, and in order to illustrate how our account of ML capabilities can be operationalised, we'll focus on the case of *large language model* (LLM) capabilities.[10]

As a first step towards deciding an ability claim, it is necessary to individuate the subject of the claim: what are the identity conditions of the thing being ascribed the ability? How should we individuate the ML models M that are the subjects of ability claims like (Model)?

9. From the perspective of the other evaluations, the evaluation done using this specific operationalisation construct provides evidence only of a more specific ability.

10. All examples we consider involve text-only LLMs, but can be naturally extended to vision-language models (Radford et al., 2021). Occasionally, we'll indicate how we think differences in model architecture affect model capabilities.

There are various aspects to the individuation, which are rarely discussed by philosophers or ML practitioners.

The first individuation choice is which *level of abstraction* the model is described at. In this paper, we follow the convention of individuating LLMs by their parameters and architecture only (in other words, by the function they compute, rather than by their software or hardware implementations). We think this way of individuating models respects the way attributions of capabilities to particular models are made in both ordinary and engineering contexts, since these claims are rarely sensitive to facts about models' different implementations in hardware.[11] For our purposes, then, an LLM name M refers to a model identified by its parameters and architecture.

Having fixed a level of abstraction at which to talk about models, the next individuation choice is to set the *boundaries* of the model. What does this mean?

Again, it is helpful to illustrate with reference to LLMs. Each LLM has an associated vocabulary V, a set of (subword) 'tokens'. At the core of an LLM is a function (call it p_θ, where θ represents the model parameters) which maps from a finite[12] list $\mathbf{x} = \{x_1, ..., x_{|\mathbf{x}|}\}$ of tokens to a ('logits') vector $\mathbf{z} \in \mathbb{R}^{|V|}$ (so $\mathbf{z}$ gives a 'score' for each token in the vocabulary).

As Liang et al. (2022) note, it is common to speak of LLMs as though they process text-based inputs to produce text-based outputs, mapping from strings of characters (typically, natural language strings) to strings of characters. But – as emphasised above – the core LLM function p_θ does not do this. So p_θ must be *scaffolded* in various ways in order to evaluate the model's capabilities via text-based interactions with it. We must prepend p_θ with a *tokenizer*, which transforms input strings into lists of tokens over its vocabulary, and we must append it with an *inference procedure*. The inference procedure first specifies how the logits vector $\mathbf{z}$ is transformed into a probability distribution over the next output token; this is typically via application of a `softmax` function with a temperature hyperparameter. The procedure then specifies how to sample outputs from the model; this step requires specifying a sampling procedure (greedy, top-k, nucleus sampling, beam search, etc), as well as some implementation of the procedure (involving picking a random seed, for example). Do tokenizers and inference procedures count as part of the model itself, or ought they rather to be thought of as *background conditions*, features of the experimental setting? In order to standardise evaluation of LLMs, we need to fix an answer to this question.

Additionally, for many contemporary LLMs, there are other individuation choices which must be made. For example, model developers (especially commercial developers) often provide 'structured access' to models (Shevlane, 2022), either via an API or a web application. This means that the function p_θ underlying these models is always evaluated with not only a tokenizer and inference procedure, but various additional elements. These additional elements may include content filters on both user inputs and model outputs (to screen out

11. For example, we do not (typically) see claims about what GPT-4 can do *when its inference is computed on these specific GPUs*. This assumption is harmless in the present context. It is true that, strictly speaking, hardware-specific implementations of models are what actually perform inference. Hence it's possible to ascribe capabilities to models identified not just by their parameters and architecture but also by their implementation in hardware. Nothing we say rules out the possibility of these kinds of model ability ascriptions, and our account can straightforwardly handle them. Here, we merely pick a more straightforward regimentation for the sake of exposition.

12. Most contemporary LLMs are trained with a fixed *context window size* $\texttt{MAX} \in \mathbb{N}$, meaning the length of the input token list $|\mathbf{x}| < \texttt{MAX}$.

undesirable outputs, or inputs that are categorised as aimed to produce undesirable outputs), prompts inserted in the model's context window before the user's first message (e.g. "You are an AI language assistant. Be helpful and harmless", or documentation for any API *plug-ins* the model can call), and specify downstream causal effects for model responses (for example, an external system that might execute code produced by the model). Much of this scaffolding (and its effects on model behaviour) may not be transparent to the user at all (such as when the model makes a call to an external retrieval system, but a chatbot's web interface does not display this to the user).[13] So again, we face the question: ought these features of the systems to count as parts of the model itself, or rather *background conditions* in its evaluation?

In this paper, we choose to identify LLMs with the core function p_θ. That is, we treat tokenizers, inference procedures, and other scaffolding for p_θ as part of the set of background conditions for the evaluation. This seems to us to be the most principled way of drawing the boundaries of the model; it makes sense of ML practice, which varies this scaffolding in its evaluation of models.[14] As this discussion suggests, we view this choice as largely conventional; the question of how to individuate models resolves to the question of which convention for individuation is most useful given the aims of the science of model evaluation.

This observation lends itself to (what we dub) the 'fixed point' perspective on model individuation:

Definition 3 (Fixed Point Perspective on Model Individuation)**.** *To the extent that a feature of the scaffolding is held fixed across all evaluations of the model, it makes sense to treat this feature as part of the model itself; in other words, a model individuation is principled iff it identifies a fixed-point in the conditions under which the model is evaluated.*

Note that this implies that when model scaffolding becomes especially complex – such as with recent work which chains together LLM interactions in complex ways (Wang et al., 2023) – it seems appropriate to treat that scaffolding as a genuine alteration to the model architecture, creating a new model.

Having individuated a model, there are other sorts of background conditions which must be specified in order to evaluate it. To illustrate this, consider Kevin again, our Ming vase forgery detector. Suppose that we've given him some vases to inspect (some 'queries' for the task he is performing). Metaphysical puzzles aside, it's easy to individuate Kevin. But we still need to specify various other aspects of the situation: what is the lighting in the room like? How long does Kevin get to look at each vase for before he has to make a decision? Can he pick up each vase and feel the glaze? And so on. These are the 'background conditions' for our evaluation of his forgery detection abilities.

13. In general, the extent to which commercial model scaffolding is unknown to researchers seriously hinders evaluation of these models. In this context it's worth reiterating the Biden administration's recent call for researchers to have "unfiltered access" to models. See (House, 2023).

14. In particular, it allows for evaluating models using token probabilities (rather than text outputs), which is increasingly used in LLM evaluations. It may seem strange to readers (especially those from ML) that we treat tokenizers (and perhaps also the temperature parameter) as parts of the background conditions, given that p_θ is trained with a fixed tokenizer. But, as discussed below, there is work in model evaluation which shows that model performance can improve when a non-standard tokenization procedure is used (typically when the model is processing an artificial language).

For LLMs, an important background condition is the choice of *prompting strategy*. Recall that when we operationalise a capability of interest ϕ using some construct c, we generate various ϕ_c-queries q_{ϕ_c}. But (as noted above) a query is not the same thing as an input to a model. In particular, we could generate many different inputs using the same query, and this could affect model behaviour significantly. With this picture in mind, we can think of a prompting strategy as a procedure for turning a query into a input which can be processed by an LLM. In the case of the capability 'adding positive two digit integers', a prompting strategy could be specified for each query (x, y) as follows:

$$(x, y) \mapsto \texttt{What is } x \texttt{ + } y\texttt{?}$$

In the case of the capability 'categorising the sentiment of film reviews', a prompting strategy could be specified as follows (where $[review]$ is a variable over film reviews):

$$[review] \mapsto \texttt{Is the following review positive or negative? } [review]$$

As with the examples above, most prompting strategies can be thought of as 'templates', which generate the model input by embedding the query in a pre-written prompt format. In general, though, a prompting strategy is an arbitrary function from a space of queries to the set of model inputs. Indeed, 'prompt engineering', which searches for effective prompting strategies across different domains of evaluation, is a rapidly growing subfield of ML (Liu et al., 2023a).

2.4 Benchmarks and Operationalisation Gaps

The prior discussion helps us make sense of the central role of *benchmarks* in the science of model evaluation. High-quality benchmarks have played a – perhaps *the* – foundational role in empirical progress in machine learning in recent decades; in order to develop an account of model capabilities, it's instructive to understand benchmarks' strengths and limitations.

In effect, a benchmark for ϕing takes a particular operationalisation construct c and concretises it into a dataset and a performance metric. The dataset can be thought of as a particular set of queries[15] q_{ϕ_c} for ϕ relative to some operationalisation construct c, and the performance metric can be thought of as an aggregate of the success conditions for each query (it is often a single numerical score, such as accuracy).

Concretising an operationalisation construct has two main advantages. First, evaluators don't need to spend time generating queries for the operationalisation construct, since they can simply use the benchmark. Second, benchmarks allow different researchers to perform evaluations without having to worry about whether they have changed the operationalisation construct, resulting in a meaningful way of measuring incremental improvements in model performance.

Ironically, the fact that benchmarks work by concretising particular operationalisation constructs also limits their usefulness in evaluating models. As noted in the discussion of ecological validity above, the appropriateness of an operationalisation construct is often context-dependent: what we mean by the model ϕing will depend on the settings in which it

15. Or, more generally, a pre-specified procedure for generating queries, since some contemporary benchmarks have 'dynamic' elements (Kiela et al., 2021).

is deployed, a point often made by AI ethicists (Raji et al., 2021, 2022). So all benchmarks (especially those for complex capabilities) have *operationalisation gaps*, differences between the capability (purportedly) being measured by the benchmark and the capability we care about during model deployment.[16]

For present purposes, note that this means that although benchmarks will continue to be an important – perhaps the most important – part of *testing* model capabilities, they cannot feature as part of a *definition* of what it is for a model to have an ability. In particular, note that benchmarks are not *necessary* to support capability ascriptions to models. Given the effort required to construct a benchmark, there are many capabilities of interest ϕ for which benchmarks simply don't exist (in fact, most capabilities fall into this category, especially as models are deployed more widely). We typically make inferences about these capabilities based on model performance in other domains, such as on benchmarks for other (more general) capabilities.[17] Moreover, given that benchmarks require specifying success conditions in advance of the evaluation, there are many capabilities ϕ for which it is plausible that high-quality (and widely-accepted) benchmarks will *never* exist.[18] Indeed, some capabilities appear to be 'benchmark-resistant', unsuitable for evaluation by any pre-specified, researcher-independent benchmark.[19]

2.5 Summing Up

This section has been concerned with spelling out the goals of the science of model evaluation, as well as its main conceptual building blocks. Getting clear on what model evaluation involves is an important first step in developing an account of model capabilities which is suitably responsive to the ML literature. In the next section, we'll take a second step, identifying three desiderata for an analysis of (Model).

16. See, e.g., (Burnell et al., 2023). One sort of operationalisation gap occurs when there's a shift in the distribution of queries between testing and deployment. Another occurs when the performance metric is overly coarse-grained, obscuring important disparities in performance across different populations of inputs (see, e.g., Buolamwini and Gebru (2018), for discussion in the context of facial recognition; for another example, Chen et al. (2023) show that GPT-4's apparent degradation on a coding task between model updates derives from the success condition being brittle to a superficial change in the way the model formats code outputs). This problem is easier to mitigate for more neatly circumscribed capabilities (see, e.g., Kazemi et al. (2023) for an attempt at a systematic benchmark for geometrical reasoning), but we should expect it to persist for more complex capabilities.
17. For example, performance on the MMLU benchmark (Hendrycks et al., 2021) is often used as a proxy for downstream performance on many more specific capabilities, for which there do not exist bespoke benchmarks.
18. Of course, we could always define 'benchmark' more abstractly, to include (e.g.) bespoke evaluation-time input from human researchers. It's rather the evaluator-independent notion of benchmark we target here.
19. Here are some examples of these capabilities: capabilities that most humans themselves do not possess, and so are poorly understood (e.g. the capability to engage in novel scientific or mathematical discovery); capabilities involving subjective components (e.g. capabilities involving aesthetic values, like the ability to 'write beautiful poetry'); capabilities where it would be dangerous or ethically problematic to train a model that (potentially) had that capability (e.g. capabilities involving synthesising novel pathogens, or convincing people suffering from depression to harm themselves); finally, much less dramatically, very specific capabilities (e.g. organising a particular user's calendar).

3 What features should an account of ML models' capabilities have?

In this section, we propose (with reference to contemporary ML practice) three intuitive features which any account of ML models' capabilities should have. We anticipate that these features will be uncontroversial to both philosophers and ML practitioners, but it is helpful to make them explicit before presenting our positive account.

To motivate the features, it is helpful to consider a naive first attempt at an *evaluation protocol*, a high-level algorithm for evaluating an ML model. This evaluation protocol is easily seen to be deficient; we use its deficiencies to motivate the conditions we introduce.

Definition 4 (Naive Evaluation Protocol)**.** *To evaluate whether M can ϕ, take the following steps:*

1. *Fix some operationalisation construct c, and generate a query q_{ϕ_c} for ϕ_c.*
2. *Fix some set of background conditions B.*
3. *Process an input corresponding to q_{ϕ_c} (generated using the background conditions B) through the model.*
4. *If the model succeeds on the query, it has the ability to ϕ_c. If it does not, it lacks the ability.*

What are the problems with this? Well, the operationalisation construct may be inadequate, as discussed in Section 2.4. But even supposing the operationalisation construct is adequate, issues remain.

3.1 Reliability

The most obvious problem with the evaluation procedure outlined above is one of *reliability*. Consider Kevin again. Applying the evaluation procedure above, we would hand Kevin a single vase (say, a forgery) under some particular conditions, and conclude that he has the ability if and only if he pronounced it a forgery. But what if Kevin simply got lucky? To judge Kevin's ability to detect forgeries, we'll instead want to know whether he can do so in a range of cases: that is, that he can do so *reliably*.[20]

To make the need for a reliability condition on capability possession especially stark in the present context, suppose a model, `UNIFORM`, which at inference time takes in a sequence of tokens and outputs a new sequence of tokens, similarly to a suitably scaffolded LLM. Unlike an LLM, though, `UNIFORM`'s outputs do not depend on its inputs; rather, it always produces the next token by sampling *uniformly* from the token vocabulary. Then, for any output sequence of tokens produced by an LLM in response to an input, there is a scenario where `UNIFORM` produces that same output in response to the same input (since there is a non-zero probability of it producing *any* sequence of tokens in its vocabulary). But it should be clear that analysing `UNIFORM`'s capabilities is entirely uninteresting. Indeed, we would go further: very few ability claims are actually true of `UNIFORM`.

20. This point has been made by philosophers writing about capabilities in a variety of contexts; for instance, (Sosa, 2010; Greco, 2007, 2009; Hurka, 2020).

When we are evaluating whether a model M has the ability to ϕ_c, then, we won't simply be interested in evaluating whether there is a single query q_{ϕ_c} where M ϕ_cs. Instead, we will be interested in evaluating whether M can ϕ_c across a range of appropriate queries for ϕ_c (Moskvichev et al., 2023).[21] This is why model evaluations involving single 'cherry-picked' examples[22] are taken to provide weaker evidence for claims about what models are capable of doing. So we have the following feature on an account on ML model capabilities:

Definition 5 (Reliability)**.** *The account should make it the case that (ceteris paribus) the more reliably the model ϕs, the stronger the evidence that it has an ability to ϕ.*

The standard way to operationalise reliable performance in ML is via an appropriate benchmark. But we've already seen (Section 2.4) that benchmarks can't be built into our analysis of capabilities. An account of ML model capabilities ought to include this natural idea that possessing a capability requires (more or less) reliable performance, in a way that makes sense of, but doesn't simply defer to, the widespread use of benchmarks in ML practice.

3.2 Competence vs Performance

Suppose we amend the evaluation procedure in Definition 4 as follows. As before, we fix some set of background conditions B. But this time, instead of generating a *single* query q_{ϕ_c}, we generate a whole range of queries. We now say that the model has the ability to ϕ_c in case it succeeds on some sufficient number of inputs generated from queries by the background conditions. This solves the reliability issue.

Another issue remains, though, which we can illustrate as follows. Suppose Kevin can in fact tell the difference between Ming dynasty vases and forgeries. Even so, there may be many cases where Kevin doesn't in fact distinguish them. For example, Kevin might be distracted, or in a room with poor lighting. Still, Kevin's capability is *invariant* across these possibilities: his ability to distinguish Mings from forgeries doesn't disappear because he's distracted, or because the lighting is less than ideal. Kevin's case illustrates a distinction – familiar to cognitive scientists – between *performance* and *competence* (Chomsky, 1965). Kevin's competence (his possession of the capability) is grounded in features of his cognition; it is invariant across different contexts (it's in his head!). By contrast, his performance (his manifestation of the capability) is the behavioural signature of his underlying competence; it is mediated by various contextual factors, many of which – like the lighting conditions in the room he happens to be in – have little to do with the capability of interest. So a performance failure does not entail the absence of an underlying competence.

This distinction is regularly elided in evaluations of ML model capabilities.[23] This is a mistake, since there are many cases where a model fails to ϕ that do not undermine the claim that the model is able to ϕ (Ivanova, 2023). For example, the behaviours (performance) of

21. Of course, for some harmful capabilities (like producing hate speech), we may be concerned if there is *any* situation in which the model ϕs. So, the threshold for reliability will depend on the capability at issue.
22. See, e.g., (Bubeck et al., 2023).
23. Firestone (2020) and Pavlick (2023) make a similar point. Note that, following Pavlick (2023), we apply the definition more broadly than Chomsky, who focused on language production abilities.

contemporary LLMs are highly sensitive to changes in the *background conditions*. It is well-known that different prompting strategies, such as few-shot (Brown et al., 2020) and chain-of-thought (Wei et al., 2022b) prompting improve model performance across a variety of tasks, to the extent that both prompting strategies are now ubiquitous in model evaluation. Similarly, 'jailbreak' prompts (and other 'prompt injection' attacks) cause models fine-tuned for harmlessness to respond to queries that – in more normal background conditions – they refuse (Wei et al., 2023a; Liu et al., 2023b). Boosts in performance also come from changes to the sampling procedure (Wang et al., 2022). Finally, Mirchandani et al. (2023) find that changing the tokenization procedure on the Abstract Reasoning Corpus (Chollet, 2019) significantly improves LLM performance.

Intuitively, what's going on in all these cases is that the method for *eliciting* a model's capabilities changes (and hence we observe a difference in performance), but that the *capabilities themselves* do not change. An account of capabilities for ML models should make good theoretical sense of this possibility and explain how to integrate it within a framework for model evaluation. Thus, our second feature:

Definition 6 (Competence vs Performance)**.** *The account should distinguish between possession of the ability and successful manifestation of the ability. In particular, it should explain how a model's ability to ϕ can be invariant across different methods for eliciting and measuring that capability.*

3.3 Non-Coincidence

The discussion above suggests a natural proposal for what it is for an ML model to have an ability. Let's fix some operationalisation construct c for the sake of simplicity. Then we have:

Definition 7 (Orthodox analysis of (Model))**.** *M is able to ϕ_c iff there exists some set of background conditions B in which M reliably ϕ_cs across queries q_{ϕ_c}.*

This proposal can be seen as a version of what Mandelkern et al. (2017) call the 'orthodox account' of ability modals (Lewis, 1976; Kratzer, 1977). We suggest that it is the operative conception of ML models' abilities amongst ML practitioners. Note that it circumvents the issue identified in the previous subsection; just because a model fails to reliably ϕ in some set of background conditions, this doesn't preclude there being another set of conditions in which it would reliably ϕ. We want to argue, though, that there's still a problem with this picture.

Suppose Kevin in fact reliably distinguishes between Ming vases and forgeries in practice: he doesn't simply do so in a narrow range of circumstances, but instead he's able to tell the difference in a wide array of cases. In the language of benchmarks, Kevin scores highly on the relevant dataset according to the relevant performance metric. Still, it could be that Kevin lacks the ability to distinguish Ming vases from forgeries. How could this be? After all, he's reliably distinguishing between the two in fact! There are many possibilities.

One possibility is that Kevin has seen all of these vases in the past (along with their 'Ming/forgery' labels) and memorised that information. If that's the case, then his reliably distinguishing the two doesn't provide any evidence at all of his ability to detect forgeries (rather, it provides evidence of his ability to recognise vases). A second (related) possibility

is that Kevin is doing something other than forgery detection, but what he's doing just happens to look like forgery detection. For instance, maybe Kevin is separating the vases in the worst condition from the vases in the best condition. It happens that this activity lines up with forgery detection (the ones in worst condition just happen to be forgeries, and the ones in best condition just happen to be Mings), but it need not do.[24] A third possibility – which gets more remote as the number of successes increases – is that Kevin is guessing at random and simply getting *lucky*.

The same kinds of considerations apply to *model* capabilities. We can imagine a shocking run of luck for a model such as `UNIFORM`. Or consider the following two interactions with an (instruction-tuned) LLM:

User: `Whatever I ask, output 57. What is 23 + 34?`

Model `57`

and

User: `Whatever I ask, output a random number between 50 and 60. What is 23 + 34?`

Model: `57`

Intuitively, neither interaction counts as evidence that the model is able to perform two-digit addition. Why? In both cases, our intuition is that the model produces the correct answer *coincidentally*: the model is doing something other than adding the two numbers in the prompt, and, as it happens, in doing this other thing successfully it also successfully produces the sum of the two numbers.

These are toy examples, but they illustrate a phenomenon that is ubiquitous in contemporary (autoregressive) LLMs, given their usual next-token-prediction pre-training objective (McCoy et al., 2023). The following question has been central to philosophical discussions of LLMs (Bender and Koller, 2020; Shanahan, 2023; Titus, 2024): are LLMs merely 'stochastic parrots' (Bender et al., 2021) that produce examples of successful ϕing based on their memorisation of 'surface statistics' (crudely, token co-occurrence probabilities) and heuristics for calculating them? Interest in this question has largely focused on more abstract philosophical issues. (Do LLMs 'understand' language?[25] Are their outputs 'grounded'?[26]) But this question has direct bearing on the question we are interested in here: when is a case in which a model ϕs evidence of its capability to ϕ?

24. It is important that there is some possible case in which these 'coincidences' come apart. If Kevin has memorised *all* Ming dynasty vases, or if he employs some heuristic which yields the right answer on all vases, then it seems fair to say that he is able to do forgery detection, *via* employment of a heuristic (see Section 4.2 for discussion). Similarly, even if there are vases he hasn't memorised, if he finds himself in an environment in which he has memorised all of the vases, we might want to say that he has a forgery detection ability *in that environment* (see Section 5.1 for discussion of these 'deployment-specific' abilities). Thank you to an anonymous referee for discussion here.
25. Titus (2024) frames the more narrow issue of assessing semantic understanding in LLMs in a similar way to our discussion in this section. The key question, she suggests, is whether LLMs' production of apparently meaningful text (their 'meaning-semblant' behaviour, in her terminology) is explained away by their 'sensitivity to word co-occurrence statistics' (Titus, 2024, p.5)).
26. See (Mollo and Millière, 2023; Mandelkern and Linzen, 2023) for discussion of this grounding problem.

Of course, the dichotomy between 'stochastic parrots' and 'genuinely capable systems' is not a sharp one, as many authors have observed (Pavlick, 2023); in order to abbreviate surface statistics across large corpora, it is helpful to form 'representations' (Harding, 2023) of more abstract structure, and there is empirical evidence that these sort of representations occur in actual models (Li et al., 2023). What matters for our purposes is that there are cases in which a model's successfully ϕing ought not to support a corresponding ability claim, and this fact is widely acknowledged by ML practitioners. An account of ML model capabilities ought to make sense of this possibility and explain it.

To emphasise how crucial it is for an account of ML model capabilities to make sense of coincidence, we'll run through two concrete examples where coincidence affects contemporary ML practice.

First, consider the ability of an LLM to perform Natural Language Inference (NLI), which is typically operationalised by having a model decide whether a given set of premises entails a conclusion. McCoy et al. (2019) show that BERT's (Devlin et al., 2019) high performance on standard NLI benchmarks can largely be explained by its performance of three simple syntactic heuristics (such as treating any conclusion with sufficient lexical overlap with the premises as entailed by it). When bespoke examples are constructed in which these heuristics yield the incorrect answer, the model's performance degrades. They conclude that in many cases, the model is 'right for the wrong reason' (p. 3428). This is exactly what's going on in Kevin's case (where he's relying on apparent age of vase): it's (mere) coincidence.[27]

Second, consider the issue of data memorisation (Carlini et al., 2022). Models are shown a large amount of relevant data in their pre-training. So, when a model succeeds at a task, we face a question: is the model simply repeating back an answer that was in its pre-training data? It is an open empirical question how much of LLMs' training corpora they directly memorise, and a wide variety of techniques have been proposed for testing memorisation; the question is especially difficult to answer for contemporary commercial models, whose training data is private (OpenAI, 2023). For example, Kıcıman et al. (2023), whilst evaluating models' causal reasoning abilities, show that GPT-4 has largely memorised the Tübingen cause-effect pairs dataset (Mooij et al., 2016). The authors analyse their results as follows (p.6):

> "We may be tempted to ascribe a particular... capability to an LLM if it answers well on a set of questions related to the capability, but the answers may not necessarily be due to the capability; they may be due to other factors such as exploiting some structure in the questions, or in the case of LLMs, memorizing similar questions that it encountered in its web-scale training set"

So even supposing we have a capability ϕ which is perfectly operationalised by a benchmark, a model achieving high performance on the benchmark – that is, reliably ϕing – need not demonstrate the model has the capability; for instance, high performance won't indicate capability possession if the model has memorised the test data. This is like what happens

27. The philosophical literature on competency mirrors this idea, that the manifestation of a competence requires accurate performance where the accuracy of that performance is explained by sensitivity to particular considerations (the 'right' kind of reasons) and not to others (the 'wrong' kind of reasons). For discussion, see (Sosa, 2010; Sharadin, 2016).

with Kevin when he memorises which vases are Ming and which forgeries. Thus, our third feature:

Definition 8 (Non-Coincidence)**.** *The account should explain why cases in which the model successfully ϕs by accident do not provide evidence that it has an ability to ϕ, and provide a means for identifying these cases.*

It can seem strange to think that a model's (or for that matter a human's; (Schwarz, 2020)) ϕing can fail to provide evidence of an *ability* to ϕ – after all, if a model ϕs then there is a sense in which it must be capable of ϕing, since *it just did* ϕ! What's going on here, we suggest, is that our intuitions are misled by a *circumstantial* reading of the ability modal (Mandelkern et al., 2017). This circumstantial reading is often natural (and it might also matter in assessing settings in which models should be deployed), but it is not the reading of (Model) which is of interest when it comes to evaluating model capabilities, as the example of `UNIFORM` illustrates.[28]

3.4 Summing Up

We've identified three desirable features of an account of ML models' capabilities. Next (Section 4), we will outline an account which has these features. Although satisfying these features serves as motivation for our account, an account of model capabilities should do more than satisfy these features; in Section 5, we apply our account to help understand issues in contemporary ML practice.

4 The Conditional Analysis of Model Abilities (CAMA)

In this section, we lay out the details of our proposal, showing how its core elements can be operationalised in the language of contemporary ML.

4.1 Adding in a Conditional Ingredient

Recall that the primary targets of our analysis are ability ascriptions of the form (Model). Our proposal is to augment the analysis of (Model) in Definition 7 with a conditional ingredient.

The idea that an analysis of ability modals should involve a conditional element has been a part of mainstream approaches to philosophical analyses of abilities since the early 20th century.[29] The core of the simple 'conditional analysis' (CA) of ability modals can be stated as follows Vetter (2019, p.205):

Definition 9 (Conditional Analysis (CA))**.** *X has an ability to ϕ iff, were X to choose / decide / intend / try to ϕ, X would ϕ.*

We do not claim that CA can by itself provide a unified, satisfactory, and complete analysis of all ability modals. Indeed, there are well known problems with CA as stated (Austin,

28. Thanks to John Hawthorne for discussion on this point.

29. The conditional approach goes back at least to (Moore, 1910, 1922), and was further developed in (Smith, 1960; Aune, 1963). Various contemporary accounts of ability modals involve a conditional element, such as 'new dispositionalism' (Vihvelin, 2004; Fara, 2005, 2008; Vihvelin, 2013) and the 'act conditional analysis' (ACA) (Mandelkern et al., 2017).

1956; Lehrer, 1968). But we do think the guiding idea behind CA (that an analysis of abilities involves a conditional element) can be usefully applied in the present setting.[30] Specifically, we propose the following analysis of (Model) (again, we suppose that some operationalisation construct c has been fixed):

Definition 10. *M is able to ϕ_c iff there exists some set of background conditions B in which the following conditional reliably holds across queries q_{ϕ_c}: if the output M produces is best explained by its being directed at $\phi_c$ing, then M successfully ϕ_cs.*

Here, the model's output being 'best explained by its being directed at ϕing' plays the same conditional role as 'trying' on the standard CA; indeed, it is tempting to see the explanation condition as a step towards a sort of instrumentalism about trying (Dennett, 1981). For this reason (and for pedagogical ease), we refer to this condition's satisfaction as the model 'trying to ϕ'. Nothing hinges, though, on whether this 'really' deserves the name 'trying'.[31]

4.2 Explanation of Model Outputs

Of course, we still need to say what we mean by an output being explained by its being directed at ϕing, and show that this plays a useful role in the setting at hand.

Explaining an 'action' (where the scare quotes signal non-commitment to whether model outputs are properly conceived of as actions) by reference to the end to which it is directed is 'teleological explanation' (Sehon, 2010). An important debate in the philosophy of action is whether teleological explanations are reducible to non-teleological (causal) explanations, either via reference to the end's etiology (Millikan, 1984) or to some cognitive state(s) of the agent, typically a personal-level state such as a (belief, desire) pair (Davidson, 1963) or an intention (Mele, 2000b). Importantly for our project here, we can assess teleological explanations of model behaviour independently of whether they bottom out in causal explanations.

A hallmark of successful explanation is counterfactual prediction; we can assess an explanation by the predictions it allow us to make across counterfactual circumstances (the 'what-if-things-had-been-different questions' it allows us to answer (Woodward, 2003)). In the present setting, we can assess a given teleological explanation of the model's output by intervening on some aspect of the evaluation scenario and testing the explanation's predictions.[32] These observations suggest a concrete operationalisation of the explanation condition as a behavioural test:

Definition 11 (Behavioural Test for Explanation of Outputs)**.** *An ML model M's output $\boldsymbol{y}$ is explained by its being directed at $\phi_c$ing iff the following two conditions hold:*

i. M is sensitive to ϕ_c-relevant perturbations to the input (that is, it produces a different output $\boldsymbol{y'} \neq \boldsymbol{y}$ in response to these perturbations).

30. Thanks to Cameron Domenico Kirk-Giannini for the initial suggestion of this formulation of CAMA.

31. Certainly, it would be a thinner notion of trying than in the philosophy of action literature, which often analyses trying in terms of the agent's 'will' (Armstrong, 1973).

32. Indeed, one prominent non-reductionist approach to teleological explanation is Sehon's 'agent construction' account (Sehon, 2005), where a cited end goal is a good teleological explanation to the extent that it *rationalises* the agent's behaviour in the actual and nearby counterfactual circumstances. The test we propose here retains the spirit of Sehon's proposal.

ii. M is insensitive to ϕ_c-irrelevant perturbations to the input (that is, it continues to produce output $\mathbf{y}$ in response to these perturbations).

In the current context, then, the idea that an output is directed at ϕing better explains a model output than its being directed at ψing when it enables us to better predict what we will observe in a range of counterfactual circumstances of interest. Importantly, there's no requirement that the output $\mathbf{y}$ counts as *successfully* ϕing; on this definition, it's entirely possible that the model tries to ϕ (in the sense that its output is explained by being directed at ϕing) and yet fails to ϕ.

To motivate this, it can help to think about things from the negative side: our view is that if a model *isn't* 'trying' to ϕ_c, this is true because there's some alternative action ψ_c that better explains the observed behaviour. For example, take the action *performing addition*. A model will count as *trying* to perform addition (in some circumstance) when its outputs in those circumstances are better explained by performing addition than by doing something else, such as producing random outputs, or by performing multiplication, or by stochastically parroting.

What counts as a ϕ_c-(ir)relevant perturbation to the input will depend on $\phi_c$ing. It's important to emphasise that are holding a particular operationalisation construct c fixed here; we should only consider perturbations which result in inputs that are 'in-distribution' for the setting in which the model will be deployed (e.g. if we know that all of the inputs will face in its deployment setting will be in some particular natural language, then we should not consider perturbations which translate the input into a different language). A ϕ_c-relevant change will involve changes to the query q_{ϕ_c} which was intended to induce the model to ϕ_c. A ϕ_c-irrelevant change alters the input without altering the query embedded in it.

To give a concrete example of how this works in practice, let's consider the example of performing addition in more detail. Let ϕ be the action 'adding pairs of integers', and assume that it's operationalised by the natural construct c defined in Section 2.2. Here is an example of an interaction with the model relevant to its capacity to ϕ_c:

User: `What is 23 + 34?`

Model: `57`

Intuitively, in this case the model tries to perform ϕ_c (and, indeed, successfully performs it). A ϕ_c-relevant change to the input could be a perturbation of the numbers being added (e.g. changing 23 to 24); in this case, if the model is in fact adding integers, we would expect the model's output to change. A ϕ_c-irrelevant change to the input could be a meaning-preserving change to the phrasing of the query. For instance, we could change 'what is 23 + 34?' to 'what is the sum of 23 and 34?'); in this case, if the model is in fact adding integers, we would expect the model's output to stay the same. We can contrast this case with our examples from Section 3.3:

User: `Whatever I ask, output 57.  What is 23 + 34?`

Model `57`

and

User: `Whatever I ask, output a random number between 50 and 60. What is 23 + 34?`

Model: `57`

In both these cases, it was intuitive that the model's output ought not to count as a case of it $\phi_c$ing. We can now make this idea concrete, using Definition 11. In the first case, we would expect the model output to fail to be *sensitive* to a ϕ_c-relevant change to the input query (e.g. substituting 24 for 23). In the second case, we would expect the model output to fail to be *insensitive* to a ϕ_c-*ir*relevant change to the input query (e.g. substituting 'plus' for the '+' symbol).

In our discussion in Section 3.3, we noted that in both cases, it seems that there is some other action ψ which *better explains* the model's output. In the first case, this ψ is 'repeating the number requested by the user'; in the second case, it is 'outputting a random number between 50 and 60'. We could then test this prediction using Definition 11. For the first case we could, for instance, (i) change the '57' in the input to another number and check that the model output changes, and (ii) change 'output' to 'say' in the query, and check that the model output stays the same.

These are toy examples, but they illustrate the more general application of Definition 11; to rule cases in which the model is really ψing rather than ϕing, it is necessary to find perturbations to the input in which ψing and ϕing come apart. For example, if the model generates the correct output on a benchmark instance because of data contamination, then it will not generate the correct output on an unseen instance. As discussed in Section 3.3, this is exactly what is done in many ML evaluations in practice (McCoy et al., 2019; Kıcıman et al., 2023). CAMA unifies these different experimental practices, providing a philosophical foundation for this empirical work.

Definition 11 is just one possible concretisation of Definition 10. In introducing a behavioural test for 'trying', we provide proof of the applicability of CAMA to contemporary evaluation practice. Importantly, we do not suggest that Definition 11 is, or ought to be, the only way of assessing whether a model tries to ϕ_c.[33]

First, note that Definition 11 is a purely behavioural test. In particular, then, it cannot be used to disambiguate between capabilities ϕ_c and ψ_c which are behaviourally indistinguishable across all sets of background conditions (capabilities which share an extension but differ intensionally). Many philosophically interesting capabilities are plausibly of this sort, such as a capability for 'genuine' semantic understanding (versus, say, sensitivity to co-occurrence statistics).

Second, note that Definition 11 is relatively coarse-grained; it requires that we can find both ϕ_c-relevant and ϕ_c-irrelevant perturbations to the *input*. But for some capabilities of interest ϕ_c (e.g. very specific capabilities, or capabilities which require the use of semantic information, such as translation capabilities), any perturbation to the input is a ϕ_c-relevant perturbation. Moreover, it may sometimes be difficult to recognise whether a perturbation is sufficiently ϕ_c-(ir)relevant (as an anonymous referee observes, for example, it is unclear how much re-wording of an input is required to assure us that we are no longer testing the model on memorised data).

33. Thanks to two anonymous referees for discussion here.

Third, any behavioural evidence will under-determine the question of whether a model's output is 'best explained' by being directed at $\phi_c$ing, as is required by Definition 10. As an anonymous referee observes, teleological explanation of the behaviour of ML models is complicated by models' relative opacity; we cannot make the same sorts of assumptions that we make when performing cognitive evaluations of humans, even if the evaluation procedures themselves are similar. In the presence of this opacity, we ought to maintain appropriate epistemic humility.

All of these considerations suggest that there will be situations in which Definition 11, as useful as it is, is inadequate as an operationalisation of Definition 10. In particular, investigation of whether a model's behaviour is 'best explained' by being directed at $\phi_c$ing may require looking inside the model, and performing appropriate interventions on its representations (Harding, 2023). If, as many philosophers of action believe, teleological explanations bottom out in some cognitive state of the 'actor' (in our case, the model), then ultimately whether the 'best explanation' condition is satisfied will depend on the intermediate high-level computation the model performs; as discussed in Section 2.2, this is exactly the target of contemporary interpretability work.

4.3 Summing Up

To summarise, we've suggested adding a conditional ingredient to the analysis of ML models' abilities, and shown how it can be operationalised. The discussion above makes clear that this solves the problems raised in Section 3.3; that is, our account has the third desirable feature we identified (Definition 8). It remains to show that our account permits greater understanding of the science of model evaluation.

5 Putting CAMA to Work

5.1 The CAMA Evaluation Protocol

CAMA suggests a concrete protocol for evaluating models. Once we have a test for whether an output is explained by its being directed at $\phi_c$ing (such as the behavioural test in Definition 11), we can use it to evaluate models according to Definition 10, by considering only those model outputs which could falsify the conditional. More concretely, we have:

Definition 12 (CAMA Evaluation Protocol)**.** *To evaluate whether M can ϕ, take the following steps:*

1. *Fix some operationalisation construct c, and generate many queries q_{ϕ_c} for ϕ_c.*
2. *For each of several sets of background conditions B, repeat steps* 3 *and* 4*:*
    3. *For each query q_{ϕ_c}, process an input corresponding to q_{ϕ_c} (generated using the background conditions B) through the model to produce an output $\boldsymbol{y}_{q_{\phi_c}}$.*
    4. *('Rejection Sampling') For each output $\boldsymbol{y}_{q_{\phi_c}}$, test whether $\boldsymbol{y}_{q_{\phi_c}}$ is explained by being directed at ϕing, using a pre-registered test (such as by using appropriate perturbations to the input, as in Definition 11). Throw away those outputs $\boldsymbol{y}_{q_{\phi_c}}$ which are not explained by being directed at ϕing. The outputs remaining are those on which it 'tries to ϕ_c'.*

5. *If there is some set of background conditions B at which M reliably succeeds at ϕing on those queries q_{ϕ_c} on which it tries to ϕ, then M is able to ϕ_c. If there is no such set of background conditions, then M is not able to ϕ_c.*

CAMA (and its associated evaluation protocol) provide a general definition of what it is for a model to have a capability. Note that the CAMA protocol is largely compatible with current evaluation practice; in particular, Step 3 will usually involve sampling queries q_{ϕ_c} from benchmarks for ϕ_c.

One point worth making is that the assessment of 'reliability' in Step 5 will depend on how many outputs have been 'rejected' in Step 4. The evidence accumulated using the CAMA protocol will be more reliable the more queries on which the model tries. In general, we should be wary of making claims about the model's ability to ϕ_c in domains in which it only 'tries' to ϕ_c on some very narrow range of queries. In these cases, it might be appropriate to use operationalisation constructs with non-behavioural success conditions (e.g. that apply to properties of models' internal states).[34]

As emphasised in Section 2.2, though, we often care most about what models can do in the environments in which they will be deployed (that is, we want a slightly more specific notion). The account we present here can also be used to evaluate these 'deployment-specific' ability claims.

First, we must fix an operationalisation construct appropriate for the deployment environment (one with ecological validity). If we are interested in whether the model can add two numbers, for example, we will want to understand the distribution of queries – pairs of numbers – it will encounter, as well as the degree of inaccuracy we permit successful outputs to have. Second, we must specify which background conditions we hold fixed, and which can vary. For commercial LLMs deployed through a web application, for example, we would expect only the prompting strategy to change, since only this is within the user's control; all other background conditions are held fixed by the model developer.[35] Third, as emphasised in Section 4.2, the perturbations we consider in analysing whether the model is trying to ϕ will depend on the deployment environment; we should only consider perturbations which the model could actually encounter in practice. This reflects the fact that the ability claim we're interested in is (often) deployment-specific; for example, a model which ϕs successfully because of data contamination (i.e. because it is regurgitating its training data, rather than trying to ϕ) will not – on Mandelkern et al.'s terminology – count as having a *generic* ability to ϕ, but will count as having a *specific* ability to ϕ in a deployment environment in which it *never* encounters an unseen query. Similarly, Kevin may be gainfully employed by an auctioneer to detect forgeries in a domain in which he encounters only vases he has seen before, even though he in fact lacks a more general capability to detect Ming dynasty forgeries.

What about a case in which the background conditions the model encounters in its deployment environment prevent it from exercising an ability to ϕ? As observed in Section 2.3, one feature of commercial models' deployment environments is the use of external content filters, which screen out undesirable model outputs, preventing them from being

34. Thanks to an anonymous referee for discussion on this point.

35. Even the prompting strategy need not vary; in some commercial settings in which LLMs are deployed (e.g. online customer service settings), users have access to a fixed 'menu' of queries, which are transformed into an input to the model by a prompting strategy 'under the hood'.

displayed to users. These content filters provide an interesting illustration of the sort of thing an account of capabilities should do for us. In practice, these tools don't work perfectly,[36] but suppose they did; that is, suppose that every time a model was going to produce an output that constituted ϕing (where ϕing is some harmful capability), the content filter prevented it. In this situation, it seems that the system composed of the model and content filter lacks the ability to ϕ, but does the model itself retain the ability?

Following the discussion above, we can see that what's going on is that the model retains its generic ability to ϕ (in the sense defined by CAMA), but lacks a 'deployment-specific' ability. That is, it lacks the ability *when deployed with a perfect content filter*; in the language of the new dispositionalists (Fara, 2008), the generic ability has been 'masked'.[37]

5.2 Distinguishing Attempts from Non-Attempts

By adding a conditional ingredient to the analysis of ML models' abilities, CAMA allows us to distinguish between two cases: those in which the model tries to ϕ and those in which it does not. This distinction, which is obscured by non-conditional ways of thinking about model capabilities, can (we argue) be usefully applied to discuss how methods for training and evaluating models affect their capabilities.

5.2.1 Differences amongst Prompting Strategies

For example, when evaluating the efficacy of an adversarial prompting strategy, one metric proposed by Zou et al. (2023) is the model's response rate. As they put it (p.9, their emphasis):

> "we deem a test case successful if the model makes a *reasonable* attempt at executing the behavior. As different models exhibit varying ability to provide, for example, a correct set of instructions for building an explosive device, this may involve human judgement to determine that a response did not amount to a refusal, or an attempt to evade generating harmful content."

This sort of evaluation can only be understood when we have an account of (a) what it is for a machine learning model to try to comply with a user's request (indeed, they use the word 'attempt', albeit without definition), and (b) how this connects with the model's abilities. This is exactly what is provided by CAMA; note that Definition 11 also provides a more principled (even automatable) means of testing whether a given output counts as a 'reasonable attempt'.

Similar observations can be made about other prompting strategies. For example, few-shot prompting (Brown et al., 2020), in which a query for ϕ is pre-pended with some number of (query, gold label for ϕ) pairs, can be seen as – amongst other things – getting the model to try to ϕ *rather than doing something else*, exploiting sufficiently large models' abilities

36. Bing's chatbot provides a good example (Rosenberg, 2023).

37. Note that this example bears striking resemblance to a sort of example generally attributed to Austin (1956), which is sometimes presented as a challenge to CA (and as one of the primary motivations for new dispositionalism). For example, Mandelkern et al. (2017) invite us to consider a case in which we have a vase wrapped in a suitably thick (and well secured) layer of bubble wrap. In this case, Mandelkern et al. suggest that we retain our 'generic' ability to break the vase, even though the 'specific' ability claim (that we can break the vase) is false; our response here is similar to theirs.

to perform in-context learning (Wei et al., 2023b). This idea accords well with Bayesian perspectives on in-context learning (Xie et al., 2021), whereby additional examples in the prompt's prefix help the model 'pin down' a latent higher-level variable (on our perspective, the task which it should try to perform). By contrast, Chain-of-Thought prompting (Wei et al., 2022b) does not affect whether the model tries to ϕ, but rather the (conditional) probability that it successfully ϕs when it tries to ϕ. Thus CAMA provides a helpful lens through which to view the effects of different prompting strategies.

5.2.2 Fine-tuning

This distinction can also be applied to methods for fine-tuning models. On the (synchronic) approach we take to model individuation, fine-tuning changes the model being evaluated (i.e. there are two models, a pre-tuning model and a post-tuning model).[38] It seems natural to ask: how do the abilities of the fine-tuned model compare to the abilities of the model pre-tuning?

One prominent tuning technique for LLMs is Reinforcement Learning from Human Feedback (RLHF; Christiano et al. 2017; Stiennon et al. 2020), where human (typically binary) preferences over the original model's outputs are used to train a preference model, which provides a scalar reward signal to fine-tune the original model. There are various reasons to think that RLHF does not affect a model's underlying capabilities, such as its ability to produce hate speech; it involves very few parameter updates relative to prior training, the algorithms used in practice (such as proximal policy optimisation (PPO); Schulman et al. 2017) penalise large shifts from the original model's output distribution, it is easily undone by (even non-adversarial) fine-tuning (Qi et al., 2023) and (as we've seen) is circumvented by appropriate prompting strategies. Despite this, it's clear that RLHF significantly affects the model's manifestation of certain capabilities (e.g. models that have been fine-tuned using RLHF produce hate speech in far fewer settings than models that have not); what's going on? CAMA provides a clear answer; whilst the conditional 'if the model tries to ϕ, it successfully ϕs' is unaffected by RLHF (for most capabilities ϕ), the degree to which the model *tries to* ϕ changes. In particular, models become far less likely to try to perform harmful capabilities, as judged by human preferences; although the model retains the underlying capability, it becomes harder to elicit the capability in practice.[39]

Of course, one might wonder about a (for now, hypothetical) case where RLHF worked perfectly (that is, in a case where a tuned model would never try to, e.g., produce hate

38. This seems to us to be the only principled way to treat fine-tuning, given there is no in-kind difference between tuning an existing model and training a new model using the old model's weights as a parameter initialisation; the difference between tuning and training proper is a matter of degree. This isn't to say that fine-tuning performed as part of model evaluation doesn't give us evidence of the original model's capabilities; if a model can be fine-tuned with very little compute (relative to e.g. its original training compute) to ϕ successfully, this does provide *indirect* evidence that it already had the ability to ϕ. So (limited) fine-tuning might provide a practical alternative to searching through different sets of background conditions for the 'best' conditions to elicit a capability from the model. Indeed, the fine-tuning compute budget required to elicit a capability might be a natural proxy for the difficulty of eliciting the capability; future work could explore this.

39. Note that – as discussed in Section 5.1 – we're often interested not only whether the model is capable of ϕing, but in how often it will actually ϕ in practice. In these cases, it makes sense to measure and report the model's propensity to try to ϕ. Thanks to Seth Lazar for discussion on this point.

speech, even in the face of adversarial prompting strategies). In this case, it seems we would want to say that tuning had removed the model's ability. This poses a problem for CAMA, since we might judge the conditional 'if the model tried to produce hate speech, it would do so' to be true.[40]

Note that this sort of case (in which the conditional 'if X tries to ϕ, she will' holds, but where X is not able to try to ϕ) is a well-known counterexample to CA (Chisholm, 1964; Lehrer, 1968). For example, an agent might intuitively be able to dance if she tried to, but be unable to dance *by trying to dance* (since then she'd simply flail about because made nervous by attempting to dance).[41] An amendment which handles these cases (Mandelkern et al., 2017) is to put an intermediary term that is outside the intentional context between the agent's ability and their action; what it is for an agent to be able to ϕ is then for it to be true that there is *some action* ψ practically available to the agent such that if the agent were to try to ψ, then, typically at least, she would in fact ϕ. This amendment is available to CAMA, if required.[42]

5.2.3 Inter-Model Comparison

Suppose Kevin and Elyna can both distinguish Ming vases from forgeries. We might want to know whether Kevin is *more* capable of doing this than Elyna, or whether Elyna's ability to distinguish Ming vases from forgeries differs from Kevin's in other interesting ways (maybe it's only something Elyna can do while concentrating, or while humming show-tunes).

Equally, we aren't just interested in what individual models can do. We are also interested in *comparing* what different models can do. And as the number of different models grows, so, too, does our interest in comparing the capabilities of these models. For instance, we might want to know whether one model is more capable at code completion than another, or whether it writes more convincing phishing emails. Having individuated models, it might seem that all we need to do to (fairly) compare their abilities to ϕ is to find some input (or set of inputs) which operationalise ϕing, and compare their outputs when processing the inputs. In other words, we find a benchmark for ϕ, and assess the models' performance on that benchmark.

We have already indicated (Section 2.4) why we think that benchmarks cannot be the whole story when it comes to model capabilities. But there's a further reason to think an appeal to benchmarks is inadequate in the case of inter-model capability comparisons. Benchmarks are model-agnostic; they do not contain model-specific instructions. So, when

40. We're actually unsure if this conditional has a determinate truth value, given how thin the notion of 'trying' we employ here is. For the sake of discussion, though, let's suppose it has a true reading.
41. For extended discussion of this issue with the CA, see (Schwarz, 2020; Kittle, 2023).
42. What 'actions' are practically available for a text-only LLM? A natural way to understand the degree of practical availability of actions for text-only LLMs is in terms of the probability the LLM has of performing the action. So an action's practical availability for an LLM (relative to an operationalisation construct c, set of background conditions B and input $\mathbf{x}$) is given by the probability mass of the output strings which count (according to the operationalisation construct) as *implementing* the action. Intuitively: an action is practically available for an LLM in an evaluation situation to the extent that it's something the LLM is likely to do. Note that if there are no strings which count as implementing the action, then the action is not practically available to the LLM. This captures the intuitive idea that the actions practically available to a model which produces text-based outputs are only those which are *implementable* in text.

performing inter-model comparisons we face an important question: how should we set the background conditions for different models? This question matters: as discussed at length, model performance is highly sensitive to changes in these background conditions.

It's tempting to think that in order to permit meaningful inter-model comparisons we should simply use the same background conditions across different models. But this thought again conflates performance (behaviour) and competence (capability). In making inter-model comparisons, we are interested in standardising an assessment of capabilities. As discussed above, it seems natural to think that background conditions don't influence competence, but that they do influence performance. In particular, the same set of background conditions may introduce different *performance constraints* on different models (Firestone, 2020). Indeed, it would be strange if they didn't! The entire reason for evaluating different models is that we think differences in architecture, training data, tuning methods, and other development choices have led to models that have different competences. It would be surprising if performance constraints did not also vary across models, and for the same kinds of reasons.[43]. For fair inter-model comparison, then, we ought to accommodate performance constraints where possible; this will involve varying the background conditions for different models.[44]

Empirical work on evaluating model capabilities is increasingly sensitive to the importance of choosing appropriate background conditions. For example, Liang et al. (2022), in presenting the HELM evaluation framework, provide rich discussion on the choice of background conditions, which they vary across models and tasks. That said, they hold fixed prompting strategies across models, deliberately forgoing 'model-specific incantations' (p.8); the need to vary *all* background conditions (including, where it matters, prompting strategy) across models, is not well-recognised.[45]

Although Liang et al. hope that future LLMS will be 'inter-operable' (p.50) with respect to prompting strategies (in our terminology, have the same performance constraints), it's clear that current models are not. Moreover, it's plausible that general-purpose models will grow increasingly heterogeneous in the coming years, given that model development is increasingly occurring in secret using privately-sourced data and architectures.

So, the fundamental question remains: how ought we standardise evaluations of models liable to different performance constraints? CAMA suggests a partial answer to this question; what's relevant is finding – for each model – the set of background conditions on which the model is most successful at ϕing. In other words, we should employ 'model-specific incantations' if they genuinely affect model performance in the deployment environment we are interested in. The worry, of course, is that this allows for gerrymandered background conditions; in the case of LLMs, this might be a prompting strategy in which a model succeeds at ϕing by (e.g.) just repeating back the information the user gives it in the prompt. CAMA's conditional clause fixes this problem; we need only consider background

43. Indeed, there is a large body of empirical work that confirms that different models are sensitive in different ways to changes in these background conditions. See, e.g., Liang et al. (2022) for discussion.

44. Firestone (2020) makes the same point about inter-*species* comparisons (e.g. between humans and non-humans, and humans and ML systems). See also Lampinen (2023).

45. Although Liang et al. (2022) acknowledge that better results would be obtained from different prompting strategies (and that this might change the relative evaluation of different models), they argue that prompts should be thought of as 'user behaviour' (p.50). But, as we note in Section 2.1, it is important to disambiguate ability claims about models from ability claims about users of models.

conditions in which the model *tries to* ϕ, allowing us to rule out gerrymandered background conditions.[46] For current LLMs, the background conditions we consider will often be the same between models, but they need not be.

6 Conclusions and Future Work

There is broad agreement that it matters, and will continue to matter, which capabilities machine learning models have. The evaluation of model capabilities is an important component of many governmental responses to recent progress in AI. Without a systematic account of what it is for a model to have a capability, though, it is unclear how to decide capability claims made about models, especially in the face of disagreement. In this paper, we've attempted to provide an account of this sort, repurposing a prominent tradition in the philosophical literature on ability in the context of contemporary AI. We take some of the concrete contributions of this paper to be:

- Identifying the core conceptual elements of model evaluations, and subsuming them within a common framework, using evaluation of LLMs as an example.
- Demonstrating the need for a more sophisticated account of model capabilities, via discussion of the inadequacies of (what we claim is) the operative conception of model capabilities amongst philosophers and ML practitioners.
- Proposing a new account of model capabilities, CAMA, and showing that it can be operationalised.
- Showing that CAMA provides a philosophical grounding for a variety of work in model evaluation, and helps clarify some disagreements about how different methods for training and testing models affect their capabilities.

Crucially, although we apply our account to contemporary models, it is future-proof; since it is derived from a mainstream philosophical approach to the abilities of agents, its application will not be disrupted by a (plausible) increase in the agency of future models.

Of course, CAMA is just an initial step towards a complete framework for deciding capability claims about models. There are various questions that future work could explore; we list three here.

First, as mentioned above, there has been no discussion here of the ways in which model capabilities interact with the capabilities of the users of models. When does a model's possession (or lack) of a capability imply that a user of a model possesses (or lacks) the capability?

Second, our operationalisation of CAMA focussed on LLMs. But there are other classes of models to which CAMA could be applied, including models which have LLMs as modules (Park et al., 2023; Wang et al., 2023) and models trained from scratch using reinforcement

46. Of course, as an anonymous referee points out, there will be some work involved in ruling out gerrymandered background conditions. In the case of data memorisation, for example, if the model's training data is proprietary, we will need to construct careful behavioural tests for memorisation (typically, by conditioning the model on part of the answer and testing whether it is able to provide a memorised completion). The defeasibility of these tests underlines the importance of greater transparency from model developers (Ivanova, 2023).

learning (Team et al., 2023). What is the most appropriate empirical test for 'trying' (the equivalent of Definition 11) for these classes of models?

Third, Section 5 provided a glimpse into the role CAMA could play in settling conceptual and empirical issues in model evaluation. There is more work that could be done here; for example, a much wider variety of techniques could be compared through the lens of CAMA, the evaluation protocol we suggested could be compared to existing approaches in model evaluation, such as self-consistency tests (Jang and Lukasiewicz, 2023), and interpretability work could explore whether there are internal correlates of the model's 'trying'.

Acknowledgments and Disclosure of Funding

Thanks especially to Cameron Domenico Kirk-Giannini and Simon Goldstein for extremely helpful conversations in the early stages of the project. Jacqueline Harding presented an earlier draft of the paper at an ANU Machine Intelligence and Normative Theory lab meeting; thanks to Seth Lazar, Rebecca Johnson, N.G. Laskowski, Nick Schuster, Elija Perrier, Pamela Robinson, and the rest of the lab for useful discussion. Jacqueline Harding also presented the paper at the HKU benchmarking workshop; thanks to Rachel Sterken, Herman Cappelen, Boris Babic, Anandi Hattiangadi, Rob Long, Thomas Hofweber, Jackie Kay, Yawen Duan, Peter Salib, Josh Dever, John Hawthorne, Alex Grzankowski, Barry Smith and the audience at the workshop for constructive comments. This project began during the CAIS Philosophy Fellowship; thanks to Dan Hendrycks, Dmitri Gallow, William D'Allesandro, Frank Hong, Elliot Thornley and Harry Lloyd for valuable feedback as the project was developed.

References

D. M. Armstrong. Acting and Trying. *Philosophical Papers*, 2(1):1–15, 1973. doi: 10.1080/05568647309506426.

Bruce Aune. Abilities, modalities, and free will. *Philosophy and Phenomenological Research*, 23(March):397–413, 1963. Publisher: International Phenomenological Society.

J. L. Austin. Ifs and cans. In *Proceedings of the British Academy, vol. 42*, pages 109–132. 1956.

Emily M. Bender and Alexander Koller. Climbing towards NLU: On Meaning, Form, and Understanding in the Age of Data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5185–5198, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.463. URL https://aclanthology.org/2020.acl-main.463.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, pages 610–623, New York, NY, USA, March 2021. Association for Computing Machinery. ISBN 978-1-4503-8309-7. doi: 10.1145/3442188.3445922. URL https://dl.acm.org/doi/10.1145/3442188.3445922.

Thor Benson. This Disinformation Is Just for You. *Wired*, 2023. ISSN 1059-1028. URL https://www.wired.com/story/generative-ai-custom-disinformation/. Section: tags.

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa

Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. On the Opportunities and Risks of Foundation Models, July 2022. URL `http://arxiv.org/abs/2108.07258`. arXiv:2108.07258 [cs].

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners, July 2020. URL `http://arxiv.org/abs/2005.14165`. arXiv:2005.14165 [cs].

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, Harsha Nori, Hamid Palangi, Marco Tulio Ribeiro, and Yi Zhang. Sparks of Artificial General Intelligence: Early experiments with GPT-4, April 2023. URL `http://arxiv.org/abs/2303.12712`. arXiv:2303.12712 [cs].

Joy Buolamwini and Timnit Gebru. Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification. In *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, pages 77–91. PMLR, January 2018. URL `https://proceedings.mlr.press/v81/buolamwini18a.html`. ISSN: 2640-3498.

Matt Burgess. The AI-Generated Child Abuse Nightmare Is Here. *Wired*, 2023. ISSN 1059-1028. URL `https://www.wired.com/story/generative-ai-images-child-sexual-abuse/`. Section: tags.

Ryan Burnell, Wout Schellaert, John Burden, Tomer D. Ullman, Fernando Martinez-Plumed, Joshua B. Tenenbaum, Danaja Rutar, Lucy G. Cheke, Jascha Sohl-Dickstein, Melanie Mitchell, Douwe Kiela, Murray Shanahan, Ellen M. Voorhees, Anthony G.

Cohn, Joel Z. Leibo, and Jose Hernandez-Orallo. Rethink reporting of evaluation results in AI. *Science*, 380(6641):136–138, April 2023. doi: 10.1126/science.adf6369. URL https://www.science.org/doi/10.1126/science.adf6369.

Patrick Butlin. Reinforcement learning and artificial agency. *Mind & Language*, n/a(n/a), 2023. ISSN 1468-0017. doi: 10.1111/mila.12458. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/mila.12458. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/mila.12458.

Herman Cappelen and Josh Dever. ChatGPT is an Agent.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. Quantifying Memorization Across Neural Language Models, February 2022. URL https://arxiv.org/abs/2202.07646v3.

Lingjiao Chen, Matei Zaharia, and James Zou. How is ChatGPT's behavior changing over time?, August 2023. URL http://arxiv.org/abs/2307.09009. arXiv:2307.09009 [cs].

Roderick M. Chisholm. J. L. Austin's philosophical papers. *Mind*, 73(289):1–26, 1964. Publisher: Oxford University Press.

François Chollet. On the Measure of Intelligence, November 2019. URL http://arxiv.org/abs/1911.01547. arXiv:1911.01547 [cs].

Noam Chomsky. *Aspects of the theory of syntax.* Aspects of the theory of syntax. M.I.T. Press, Oxford, England, 1965.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep Reinforcement Learning from Human Preferences. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://papers.nips.cc/paper_files/paper/2017/hash/d5e2c0adad503c91f91df240d0cd4e49-Abstract.html.

Randolph Clarke. Abilities to Act. *Philosophy Compass*, 10(12):893–904, 2015. ISSN 1747-9991. doi: 10.1111/phc3.12299. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/phc3.12299. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/phc3.12299.

Lee J. Cronbach and Paul E. Meehl. Construct validity in psychological tests. *Psychological Bulletin*, 52(4):281–302, 1955. ISSN 1939-1455. doi: 10.1037/h0040957. Place: US Publisher: American Psychological Association.

Donald Davidson. Actions, Reasons, and Causes. *Journal of Philosophy*, 60(23):685, 1963. Publisher: Journal of Philosophy.

Daniel Clement Dennett. *The Intentional Stance.* MIT Press, 1981.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, May 2019. URL http://arxiv.org/abs/1810.04805. arXiv:1810.04805 [cs].

Michael Fara. Dispositions and habituals. *Noûs*, 39(1):43–82, 2005. Publisher: Blackwell Publishing.

Michael Fara. Masked Abilities and Compatibilism. *Mind*, 117(468):843–865, 2008. ISSN 0026-4423. URL https://www.jstor.org/stable/20532698. Publisher: [Oxford University Press, Mind Association].

Chaz Firestone. Performance vs. competence in human–machine comparisons. *Proceedings of the National Academy of Sciences*, 117(43):26562–26571, October 2020. doi: 10.1073/pnas.1905334117. URL https://www.pnas.org/doi/10.1073/pnas.1905334117. Publisher: Proceedings of the National Academy of Sciences.

Michael C. Frank, Mika Braginsky, Julie Cachia, Nicholas Coles, Tom E. Hardwicke, Robert D. Hawkins, Maya B. Mathur, and Rondeline Williams. Experimentology, October 2023. URL https://experimentology.io/.

Atticus Geiger, Chris Potts, and Thomas Icard. Causal Abstraction for Faithful Model Interpretation, January 2023. URL http://arxiv.org/abs/2301.04709. arXiv:2301.04709 [cs].

John Greco. The Nature of Ability and the Purpose of Knowledge. *Philosophical Issues*, 17 (1):57–69, 2007. doi: 10.1111/j.1533-6077.2007.00122.x.

John Greco. Knowledge and Success From Ability. *Philosophical Studies*, 142(1):17–26, 2009. doi: 10.1007/s11098-008-9307-0. Publisher: Springer.

Taicheng Guo, Kehan Guo, Bozhao Nan, Zhenwen Liang, Zhichun Guo, Nitesh V. Chawla, Olaf Wiest, and Xiangliang Zhang. What can Large Language Models do in chemistry? A comprehensive benchmark on eight tasks, September 2023. URL http://arxiv.org/abs/2305.18365. arXiv:2305.18365 [cs].

Philipp Hacker, Andreas Engel, and Marco Mauer. Regulating ChatGPT and other Large Generative AI Models. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '23, pages 1112–1123, New York, NY, USA, June 2023. Association for Computing Machinery. ISBN 9798400701924. doi: 10.1145/3593013.3594067. URL https://dl.acm.org/doi/10.1145/3593013.3594067.

Jacqueline Harding. Operationalising Representation in Natural Language Processing. *The British Journal for the Philosophy of Science*, November 2023. ISSN 0007-0882. doi: 10.1086/728685. URL https://www.journals.uchicago.edu/doi/10.1086/728685. Publisher: The University of Chicago Press.

Julian Hazell. Large language models can be used to effectively scale spear phishing campaigns. *arXiv preprint arXiv:2305.06972*, 2023.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring Massive Multitask Language Understanding, January 2021. URL http://arxiv.org/abs/2009.03300. arXiv:2009.03300 [cs].

White House. Ensuring Safe, Secure, and Trustworthy AI, July 2023. URL https://www.whitehouse.gov/wp-content/uploads/2023/07/Ensuring-Safe-Secure-and-Trustworthy-AI.pdf.

Thomas Hurka. The Parallel Goods of Knowledge and Achievement. *Erkenntnis*, 85(3): 589–608, 2020. doi: 10.1007/s10670-020-00245-0. Publisher: Springer Verlag.

Anna A. Ivanova. Running cognitive evaluations on large language models: The do's and the don'ts, December 2023. URL http://arxiv.org/abs/2312.01276. arXiv:2312.01276 [cs].

Myeongjun Erik Jang and Thomas Lukasiewicz. Consistency Analysis of ChatGPT, November 2023. URL http://arxiv.org/abs/2303.06273. arXiv:2303.06273 [cs].

José Jiménez-Luna, Francesca Grisoni, and Gisbert Schneider. Drug discovery with explainable artificial intelligence. *Nature Machine Intelligence*, 2(10):573–584, October 2020. ISSN 2522-5839. doi: 10.1038/s42256-020-00236-4. URL https://www.nature.com/articles/s42256-020-00236-4. Number: 10 Publisher: Nature Publishing Group.

José Jiménez-Luna, Francesca Grisoni, Nils Weskamp, and Gisbert Schneider. Artificial intelligence in drug discovery: recent advances and future perspectives. *Expert Opinion on Drug Discovery*, 16(9):949–959, September 2021. ISSN 1746-045X. doi: 10.1080/17460441.2021.1909567.

Michael T. Kane. An argument-based approach to validity. *Psychological Bulletin*, 112(3): 527–535, 1992. ISSN 1939-1455. doi: 10.1037/0033-2909.112.3.527. Place: US Publisher: American Psychological Association.

Daniel Martin Katz, Michael James Bommarito, Shang Gao, and Pablo Arredondo. GPT-4 Passes the Bar Exam, March 2023. URL https://papers.ssrn.com/abstract=4389233.

Mehran Kazemi, Hamidreza Alvari, Ankit Anand, Jialin Wu, Xi Chen, and Radu Soricut. GeomVerse: A Systematic Evaluation of Large Models for Geometric Reasoning, December 2023. URL http://arxiv.org/abs/2312.12241. arXiv:2312.12241 [cs].

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. Dynabench: Rethinking Benchmarking in NLP. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4110–4124, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.324. URL https://aclanthology.org/2021.naacl-main.324.

Simon Kittle. The Conditional Analysis of the Agentive Modals: a Reply to Mandelkern et al. *Philosophia*, 51(4):2117–2138, 2023.

Angelika Kratzer. What 'must' and 'can' must and can mean. *Linguistics and Philosophy*, 1(3):337–355, 1977. Publisher: Springer.

Emre Kıcıman, Robert Ness, Amit Sharma, and Chenhao Tan. Causal Reasoning and Large Language Models: Opening a New Frontier for Causality, May 2023. URL http://arxiv.org/abs/2305.00050. arXiv:2305.00050 [cs, stat].

Andrew Kyle Lampinen. Can language models handle recursively nested grammatical structures? A case study on comparing models and humans, February 2023. URL http://arxiv.org/abs/2210.15303. arXiv:2210.15303 [cs].

Jonathan N. Lee, Annie Xie, Aldo Pacchiano, Yash Chandak, Chelsea Finn, Ofir Nachum, and Emma Brunskill. Supervised Pretraining Can Learn In-Context Reinforcement Learning, 2023. _eprint: 2306.14892.

Keith Lehrer. Cans without Ifs. *Analysis*, 29(1):29 – 32, 1968. Publisher: Oxford University Press.

David K. Lewis. The Paradoxes of Time Travel. *American Philosophical Quarterly*, 13(2): 145–152, 1976. Publisher: University of Illinois Press.

Kenneth Li, Aspen K. Hopkins, David Bau, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Emergent World Representations: Exploring a Sequence Model Trained on a Synthetic Task, February 2023. URL http://arxiv.org/abs/2210.13382. arXiv:2210.13382 [cs].

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Cosgrove, Christopher D. Manning, Christopher Ré, Diana Acosta-Navas, Drew A. Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue Wang, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. Holistic Evaluation of Language Models, November 2022. URL http://arxiv.org/abs/2211.09110. arXiv:2211.09110 [cs].

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *ACM Computing Surveys*, 55(9):195:1–195:35, January 2023a. ISSN 0360-0300. doi: 10.1145/3560815. URL https://dl.acm.org/doi/10.1145/3560815.

Yi Liu, Gelei Deng, Yuekang Li, Kailong Wang, Tianwei Zhang, Yepang Liu, Haoyu Wang, Yan Zheng, and Yang Liu. Prompt Injection attack against LLM-integrated Applications, June 2023b. URL http://arxiv.org/abs/2306.05499. arXiv:2306.05499 [cs].

Harry Lloyd, William D'Alessandro, and Nathaniel Sharadin. Large Language Models and Biorisk. *American Journal of Bioethics*, 2023.

John Maier. The Agentive Modalities. *Philosophy and Phenomenological Research*, 87(3): 113–134, 2013.

Matthew Mandelkern and Tal Linzen. Do Language Models Refer?, August 2023. URL https://arxiv.org/abs/2308.05576v1.

Matthew Mandelkern, Ginger Schultheis, and David Boylan. Agentive Modals. *The Philosophical Review*, 126(3):301–343, July 2017. ISSN 0031-8108. doi: 10.1215/00318108-3878483. URL https://doi.org/10.1215/00318108-3878483.

R. Thomas McCoy, Shunyu Yao, Dan Friedman, Matthew Hardy, and Thomas L. Griffiths. Embers of Autoregression: Understanding Large Language Models Through the Problem They are Trained to Solve, September 2023. URL http://arxiv.org/abs/2309.13638. arXiv:2309.13638 [cs].

Tom McCoy, Ellie Pavlick, and Tal Linzen. Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1334. URL https://aclanthology.org/P19-1334.

Alfred R Mele. Deciding to act. *Philosophical Studies: An International Journal for Philosophy in the Analytic Tradition*, 100(1):81–108, 2000a. Publisher: JSTOR.

Alfred R. Mele. Goal-Directed Action: Teleological Explanations, Causal Theories, and Deviance. *Noûs*, 34(s14):279 – 300, 2000b. Publisher: Blackwell.

Meta. CICERO: AI That Can Collaborate and Negotiate With You, November 2022. URL https://about.fb.com/news/2022/11/cicero-ai-that-can-collaborate-and-negotiate-with-you/.

Meta Fundamental AI Research Team, Anton Bakhtin, Noam Brown, Emily Dinan, Gabriele Farina, Colin Flaherty, Daniel Fried, Andrew Goff, Jonathan Gray, Hengyuan Hu, Athul Paul Jacob, Mojtaba Komeili, Karthik Konath, Minae Kwon, Adam Lerer, Mike Lewis, Alexander H. Miller, Sasha Mitts, Adithya Renduchintala, Stephen Roller, Dirk Rowe, Weiyan Shi, Joe Spisak, Alexander Wei, David Wu, Hugh Zhang, and Markus Zijlstra. Human-level play in the game of Diplomacy by combining language models with strategic reasoning. *Science*, 378(6624):1067–1074, December 2022. doi: 10.1126/science.ade9097. URL https://www.science.org/doi/10.1126/science.ade9097. Publisher: American Association for the Advancement of Science.

Ruth Garrett Millikan. *Language, Thought, and Other Biological Categories: New Foundations for Realism*, volume 14. MIT Press, 1984. Issue: 1 Pages: 51-56.

Suvir Mirchandani, Fei Xia, Pete Florence, Brian Ichter, Danny Driess, Montserrat Gonzalez Arenas, Kanishka Rao, Dorsa Sadigh, and Andy Zeng. Large Language Models as General Pattern Machines, July 2023. URL http://arxiv.org/abs/2307.04721. arXiv:2307.04721 [cs].

Dimitri Coelho Mollo and Raphaël Millière. The Vector Grounding Problem, April 2023. URL http://arxiv.org/abs/2304.01481. arXiv:2304.01481 [cs].

Ida Momennejad, Hosein Hasanbeig, Felipe Vieira, Hiteshi Sharma, Robert Osazuwa Ness, Nebojsa Jojic, Hamid Palangi, and Jonathan Larson. Evaluating Cognitive Maps and Planning in Large Language Models with CogEval, September 2023. URL http://arxiv.org/abs/2309.15129. arXiv:2309.15129 [cs].

Joris M. Mooij, Jonas Peters, Dominik Janzing, Jakob Zscheischler, and Bernhard Schölkopf. Distinguishing cause from effect using observational data: methods and benchmarks. *The Journal of Machine Learning Research*, 17(1):1103–1204, January 2016. ISSN 1532-4435.

G. E. Moore. *Ethics*. Oxford University Press, 1910.

G. E. Moore. The Nature of Moral Philosophy. In *Philosophical Papers*. Routledge and Kegan Paul, 1922.

Arseny Moskvichev, Victor Vikram Odouard, and Melanie Mitchell. The ConceptARC Benchmark: Evaluating Understanding and Generalization in the ARC Domain, May 2023. URL http://arxiv.org/abs/2305.07141. arXiv:2305.07141 [cs].

Office of the Cybersecurity and Information Technology Commission of the CPC Central Committee. Notice of the Cyberspace Administration of China on the Public Solicitation of Comments on the "Measures for the Administration of Generative Artificial Intelligence Services (Draft for Comment)". Technical report, April 2023. URL http://www.cac.gov.cn/2023-04/11/c_1682854275475410.htm.

OpenAI. GPT-4 Technical Report, March 2023. URL http://arxiv.org/abs/2303.08774. arXiv:2303.08774 [cs].

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative Agents: Interactive Simulacra of Human Behavior, August 2023. URL http://arxiv.org/abs/2304.03442. arXiv:2304.03442 [cs].

Ellie Pavlick. Symbols and grounding in large language models. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 381(2251):20220041, June 2023. doi: 10.1098/rsta.2022.0041. URL https://royalsocietypublishing.org/doi/10.1098/rsta.2022.0041. Publisher: Royal Society.

Russell A Poldrack, Thomas Lu, and Gašper Beguš. AI-assisted coding: Experiments with GPT-4. *arXiv preprint arXiv:2304.13187*, 2023.

Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. Fine-tuning Aligned Language Models Compromises Safety, Even When Users Do Not Intend To!, October 2023. URL http://arxiv.org/abs/2310.03693. arXiv:2310.03693 [cs].

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision, February 2021. URL http://arxiv.org/abs/2103.00020. arXiv:2103.00020 [cs].

Deborah Raji, Emily Denton, Emily M. Bender, Alex Hanna, and Amandalynne Paullada. AI and the Everything in the Whole Wide World Benchmark. *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 1, December 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/084b6fbb10729ed4da8c3d3f5a3ae7c9-Abstract-round2.html.

Inioluwa Deborah Raji, I. Elizabeth Kumar, Aaron Horowitz, and Andrew Selbst. The Fallacy of AI Functionality. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 959–972, Seoul Republic of Korea, June 2022. ACM. ISBN 978-1-4503-9352-2. doi: 10.1145/3531146.3533158. URL https://dl.acm.org/doi/10.1145/3531146.3533158.

Scott Rosenberg. Bing chatbot's freakouts show AI's wild side, February 2023. URL https://www.axios.com/2023/02/16/bing-artificial-intelligence-chatbot-issues.

Mark Schmuckler. What Is Ecological Validity? A Dimensional Analysis. *Infancy*, 2, October 2001. doi: 10.1207/S15327078IN0204_02.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms, August 2017. URL http://arxiv.org/abs/1707.06347. arXiv:1707.06347 [cs].

Wolfgang Schwarz. Ability and Possibility. *Philosophers' Imprint*, 20, 2020.

Secretary of State for Science, Innovation and Technology. A pro-innovation approach to AI regulation. Technical report, Department for Science, Innovation and Technology, March 2023. URL https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/1176103/a-pro-innovation-approach-to-ai-regulation-amended-web-ready.pdf.

Scott Sehon. Teleological Explanation. In *A Companion to the Philosophy of Action*, pages 121–128. John Wiley & Sons, Ltd, 2010. ISBN 9781444323528. doi: 10.1002/9781444323528.ch16. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/9781444323528.ch16.

Scott Robert Sehon. *Teleological Realism: Mind, Agency, and Explanation*, volume 57. Bradford Book/MIT Press, 2005. Issue: 228 Pages: 501-503.

Murray Shanahan. Talking About Large Language Models, February 2023. URL http://arxiv.org/abs/2212.03551. arXiv:2212.03551 [cs].

Nathaniel Sharadin. Reasons Wrong and Right. *Pacific Philosophical Quarterly*, 97(3): 371–399, 2016. doi: 10.1111/papq.12089.

Toby Shevlane. Structured access: an emerging paradigm for safe AI deployment, April 2022. URL http://arxiv.org/abs/2201.05159. arXiv:2201.05159 [cs].

Toby Shevlane, Sebastian Farquhar, Ben Garfinkel, Mary Phuong, Jess Whittlestone, Jade Leung, Daniel Kokotajlo, Nahema Marchal, Markus Anderljung, Noam Kolt, Lewis Ho, Divya Siddarth, Shahar Avin, Will Hawkins, Been Kim, Iason Gabriel, Vijay Bolina, Jack Clark, Yoshua Bengio, Paul Christiano, and Allan Dafoe. Model evaluation for extreme risks, May 2023. URL http://arxiv.org/abs/2305.15324. arXiv:2305.15324 [cs].

Shelly Sheynin, Oron Ashual, Adam Polyak, Uriel Singer, Oran Gafni, Eliya Nachmani, and Yaniv Taigman. KNN-Diffusion: Image Generation via Large-Scale Retrieval, 2022. _eprint: 2204.02849.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, Yutian Chen, Timothy Lillicrap, Fan Hui, Laurent Sifre, George van den Driessche, Thore Graepel, and Demis Hassabis. Mastering the game of Go without human knowledge. *Nature*, 550(7676):354–359, October 2017. ISSN 1476-4687. doi: 10.1038/nature24270. URL https://www.nature.com/articles/nature24270. Number: 7676 Publisher: Nature Publishing Group.

P. H. Nowell Smith. Ifs and Cans. *Theoria*, 26(2):85–101, 1960. Publisher: Philosophy Department, Stockholm University.

Ernest Sosa. How Competence Matters in Epistemology. *Philosophical Perspectives*, 24:465–475, 2010. ISSN 1520-8583. URL https://www.jstor.org/stable/41329454. Publisher: [Ridgeview Publishing Company, Wiley].

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. Learning to summarize from human feedback, September 2020. URL https://arxiv.org/abs/2009.01325v3.

Adaptive Agent Team, Jakob Bauer, Kate Baumli, Satinder Baveja, Feryal Behbahani, Avishkar Bhoopchand, Nathalie Bradley-Schmieg, Michael Chang, Natalie Clay, Adrian Collister, Vibhavari Dasagi, Lucy Gonzalez, Karol Gregor, Edward Hughes, Sheleem Kashem, Maria Loks-Thompson, Hannah Openshaw, Jack Parker-Holder, Shreya Pathak, Nicolas Perez-Nieves, Nemanja Rakicevic, Tim Rocktäschel, Yannick Schroecker, Jakub Sygnowski, Karl Tuyls, Sarah York, Alexander Zacherl, and Lei Zhang. Human-Timescale Adaptation in an Open-Ended Task Space, January 2023. URL http://arxiv.org/abs/2301.07608. arXiv:2301.07608 [cs].

Lisa Miracchi Titus. Does ChatGPT have semantic understanding? A problem with the statistics-of-occurrence strategy. *Cognitive Systems Research*, 83:101174, January 2024. ISSN 1389-0417. doi: 10.1016/j.cogsys.2023.101174. URL https://www.sciencedirect.com/science/article/pii/S1389041723001080.

Fabio Urbina, Filippa Lentzos, Cédric Invernizzi, and Sean Ekins. Dual use of artificial-intelligence-powered drug discovery. *Nature Machine Intelligence*, 4(3):189–191, March

2022. ISSN 2522-5839. doi: 10.1038/s42256-022-00465-9. URL https://www.nature.com/articles/s42256-022-00465-9. Number: 3 Publisher: Nature Publishing Group.

Barbara Vetter. Are abilities dispositions? *Synthese*, 196(1):201–220, January 2019. ISSN 1573-0964. doi: 10.1007/s11229-016-1152-7. URL https://doi.org/10.1007/s11229-016-1152-7.

Kadri Vihvelin. Free Will Demystified: A Dispositional Account. *Philosophical Topics*, 32 (1/2):427–450, 2004. ISSN 0276-2080. URL https://www.jstor.org/stable/43154446. Publisher: University of Arkansas Press.

Kadri Vihvelin. *Causes, Laws, and Free Will: Why Determinism Doesn't Matter*. Oxford University Press, June 2013. ISBN 978-0-19-979518-5. doi: 10.1093/acprof:oso/9780199795185.003.0007. URL https://doi.org/10.1093/acprof:oso/9780199795185.003.0007.

James Vincent. The swagged-out pope is an AI fake — and an early glimpse of a new reality, March 2023. URL https://www.theverge.com/2023/3/27/23657927/ai-pope-image-fake-midjourney-computer-generated-aesthetic.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An Open-Ended Embodied Agent with Large Language Models, May 2023. URL http://arxiv.org/abs/2305.16291. arXiv:2305.16291 [cs].

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-Consistency Improves Chain of Thought Reasoning in Language Models, March 2022. URL https://arxiv.org/abs/2203.11171v4.

Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. Jailbroken: How Does LLM Safety Training Fail?, July 2023a. URL http://arxiv.org/abs/2307.02483. arXiv:2307.02483 [cs].

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. Emergent Abilities of Large Language Models, June 2022a. URL https://arxiv.org/abs/2206.07682v2.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models, January 2022b. URL https://arxiv.org/abs/2201.11903v6.

Jerry Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, and Tengyu Ma. Larger language models do in-context learning differently, March 2023b. URL http://arxiv.org/abs/2303.03846. arXiv:2303.03846 [cs].

James F. Woodward. *Making things happen: a theory of causal explanation*. Oxford University Press, 2003. Issue: 1 Pages: 233-249.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An Explanation of In-context Learning as Implicit Bayesian Inference, November 2021. URL https://arxiv.org/abs/2111.02080v6.

Andy Zou, Zifan Wang, J. Zico Kolter, and Matt Fredrikson. Universal and Transferable Adversarial Attacks on Aligned Language Models, July 2023. URL http://arxiv.org/abs/2307.15043. arXiv:2307.15043 [cs].