ACI: An Analogy Based Intelligence model

Akira Pyinya
akirapyinya@gmail.com

Abstract

Inspired by the Copycat Project, we construct ACI, an analogy-based theory of intelligence in which intelligence is defined as doing the same thing in new circumstances, rather than as an optimization force that pursues goals or maximizes utility. The ACI theory integrates different paradigms of cognitive science and artificial intelligence, explains the emergence of intelligence, and provides a novel perspective on AI alignment that focuses on the balance between capability and normativity and rules out the Paperclip Maximizer scenario. It also shows the possibility of constructing analogy-based machine learning and neural network projects that can outperform current projects in terms of interpretability.

Introduction

Hofstadter (2001) argues that analogy is at the core of cognition. But what is the place of analogy in the more recent artificial intelligence (AI) paradigms, such as deep learning, large language models, and AI alignment? What's the relationship between analogy and the more popular concepts like goal, utility, and normativity?

We go one step further and argue that *analogy is at the core of intelligence*, whereas goals and utility functions emerge from the process of "doing the same thing."

Based on analogy, we construct a universal intelligence framework called **ACI** for **Algorithmical Common Intelligence** or **Active Copycat-Inspired model**, since it implements the algorithmic information theory, arises from the common law system, and can be explained through Hofstadter and Mitchell's (1995) **Copycat Project** and 11 transformations of its key questions.

From the **fluid concepts** model in the Copycat, we derive the model of **fluid goals**, which argues that goals are not static, but adapt to unanticipated circumstances.

Through the eyes of analogy, machine learning and AI approaches can be integrated with cognitive science and the theory of evolution. The original Copycat simulates the processes of analogical thinking by asking "change letter-strings in the same way" questions, such as:

> **Question 1:** Suppose the letter-string **abc** were changed to **abd**; how would you change the letter-string **ijk** in "the same way?"

ACI argues that intelligence is *"doing the same thing in different situations,"* which can be demonstrated by a Copycat-inspired "acting in the same way" question:

> **Question 2:** Suppose we do **abd** in environment **abc** ; how would you act "the same way" in environment **ijk**?

In the following chapters, we will explore the analogy-based nature of intelligence, the evolutionary origin of normativity, the tension between "doing the same thing" and "in different situations," and the transition from the optimization models to an analogy model. Finally, the ACI model will be applied to modern machine learning and other paradigms.

Do the Same Thing as the Right Things

An intelligent system should do the right thing (Russell 1991). But how does a machine know "what counts as the right thing"? In other words, how does it get the **normative information**?

A mainstream AI textbook argues that the right thing can be represented by **goals** or **utility functions** that are internalized into the system, because a goal describes desired situations, while utility measures how much something is desired (Russell 2010). Thus, we can "simply ensure that the robot associates the preference with the human" (Russell 2019) by designing a right goal that aligns with human preference. (*For the purposes of this article, terms such as **goal**, **utility**, **reward**, or **objective** that represent the preferences are more or less equivalent and will be often used interchangeably).*

For example, **value learning** approaches, such as **Inverse Reinforcement Learning** or **IRL** (Ng & Russell 2000) suggest that the utility function of humans can be obtained by studying their behavior. By imprinting the right utility function into machines, we can force machines to do the right thing. The only conundrum is the complexity of human preferences.

The ACI theory argues that representing human preferences with **fixed goals** is like representing human cognition with **static concepts**. Inspired by Copycat's model of **fluid concepts** (Hofstadter & Mitchell 1995), ACI introduces **fluid goals**, which believes that goals are not fixed or static, but adapt themselves to unanticipated circumstances. The normative information is acquired directly by analogy, by *doing the same thing as the right thing*. Under certain pressures, the internal "fabric" of "the right thing to do" slips into a similar set of goals, just as Copycat's **conceptual slippages** which argues that a concept would slip into a similar set of concepts under certain pressures (Mitchell 1993).

For example, suppose natural selection provides a bacterium with examples of moving in the opposite direction of the nutrient concentration gradient as the examples of doing the right thing. We may conclude that its goal is to move toward the nutrient. However, this strategy may fail in some circumstances, such as when higher nutrient concentrations are accompanied by higher toxin concentrations. It will have to deal with impasses and slip into other similar goals.

When provided with examples of the right thing to do, rather than with goals or utility functions, the best way to do the right thing is to do the same thing as the examples, rather than to stick to a fixed goal. ACI can extract the underlying essence of the right thing, and *"do the same thing" in a new situation,* just as a Copycat changes a new letter-string "in the same way".

We can start by asking, "How can we do the same thing?"

Pattern Extrapolation and Imitation

A simple way to do the same thing as the original thing, is to extrapolate the original pattern so that the extrapolated sequence follows the same pattern as the original sequence. Suppose there is a letter-string that represents the original "right thing to do," then the extrapolation of that string is more likely to represent the right thing to do than other strings. It's like asking a question:

> **Question 3:** Suppose 10111010101 is a string that represents the right thing in the past, what is the string that represents the right thing in the future?

There is always more than one answer to an extrapolation problem. As pattern extrapolation approaches such as Seek-Whence (Meredith 1986) and Solomonoff Induction (Solomonoff 1964) have shown, the result of extrapolation is a probability distribution of the possible next token(s), which means that every possible future has a probability of being the right thing to do, although most of them are too small to notice.

For example, if we know that the binary string 101101010010 represents the right thing to do, and the next token of the string has a higher probability of being 1, then the binary string 1011010100101 has a higher probability of representing the right thing to do than 1011010100100.

We define the **utility** of doing something as its *probability of being the right thing to do*, since this probability represents the preference for doing the right thing, similar to the notion of utility in economic theory, which represents human preference (Von Neumann & Morgenstern 1947).

As Sutskever argues, predicting the next token can reveal the inner structure of the original data (Patel & Sutskever 2023). By extrapolating a string that represents an example of the right thing, one can understand the underlying essence of "doing the right thing." For example, if the example of nutrient collecting behaviors is provided as the right thing to do, an intelligent system can conclude that continuing collecting nutrients is very likely to be the right thing to do in the future.

(Predicting the right thing to do is different from predicting the future, because not everything is right. The future should be predicted by extrapolating the pattern of the whole history instead of only the pattern of the right thing.)

Sequence extrapolation can figure out the right thing to do after a series of known right things, but how do we figure out the right thing to do in a new situation? For example, what should the bacteria do in a new environment with no nutrients or when the concentration of nutrients is too high?

Simply repeating the original pattern is not a good solution. For example, imitation learning approaches (Hussein et al. 2017) that imitate the original behaviors have difficulty getting one individual to do the same thing as another. Christian (2020) showed that imitating an expert may be useless because you simply can't do what the expert can do, but it's also hard to outperform a less intelligent teacher by imitating them. Similarly, Christiano (2015) argues that if a human fails to accomplish a task most of the time, the best way to do the same thing should be to fail like humans, even for a machine that is much more intelligent than humans. Nehaniv and Dautenhahn (2002) argue that it's necessary to solve the correspondence problem to find the mapping between the demonstrator and the imitator.

Nevertheless, ACI argues that we can "do the same thing in a different circumstance" without the mapping between different circumstances, by using the string changing technique borrowed from the Copycat project.

Active Copycat: Do the Same Thing in a Different Circumstance

We can do the same thing in a different circumstance following the Copycat way, by answering these two questions in the same way:

How to change a new letter-string in the same way that the original string was changed?

How to do the same thing as the original right thing, but in a new circumstance?

A letter-string can represent a circumstance or a state of the world, and changing of the letter-string can represent "doing things", that's how Question 2 is derived from the original Copycat question.

The Copycat argues that concepts are allowed to "slip" to related concepts in response to the "pressures" present in a new situation (Mitchell 1993); similarly, in order to do the same thing in a new situation, actions are also allowed to *slip to similar actions* in response to the pressures present in a new situation.

Doing the same thing in a different situation is not imitating the appearance, but being the same on an abstract level. For example, what should a smarter-than-human AI do when performing the same task as a human? In the AI's situation, although everything from the task to the environment is the same as that of humans, its level of intelligence is different. Since the intelligent system itself can be represented by a letter-string, it's like asking a Copycat question:

> **Question 4:** Suppose an intelligent system that can be represented by **abc** was doing **abd**; how would another intelligent system that can be represented by **kjjlll** do "in the same way" ?

To do the same thing as humans do, the AI must figure out the underlying essence of the original behavior, like "do what I did" in the Tabletop game (Hofstadter & French 1995) . The AI should be able to grasp the goals underlying the original task, which might slip into similar goals, such as accomplishing the task with a higher success rate because it has greater capability than humans, and "Great power comes with great responsibility." On the contrary, if the robot and the human take exactly the same movements, they are very likely not doing the same thing, since the robot is faking to be weak but the human is not.

Goals and Utilities Arise from Analogy

Douglas Hofstadter (2001) suggests that the goal-directed behavior is analogical in nature. Goals arise from an individual's perceptual process that continually "seeks, whenever possible, to employ high-level concepts that one is used to, that one believes in, that one is comfortable with, that are one’s pet themes." Similarly, in the ACI framework, the concepts of goals and utilities can arise from the analogical principle of "doing the same thing as the right thing."

Goals can arise from answering Question 2 or Question 4. For example, a possible answer "do **ijl"** can be interpreted as "to achieve a goal state **ijl**".

We can also derive **utility functions** from the Copycat. Like sequence extrapolation approaches, the output of a Copycat program is also a probability distribution of letter-strings, and we can also define the utility of a letter-string as its probability of representing the right thing to do in the future. For example, "the action **ijl** has a 68% probability of being the right thing to do" can be interpreted as "achieving the state **ijl** will give you 68 points of utility".

However, goals like "to achieve the goal state **ijl**" are usually called subgoals because they can only represent a temporary preference. Most of the time, people talk about more general goals like "win a chess game", or "write a good novel".

In the ACI framework, general goals are made up of ensembles of subgoals. For example, we can get the probability of "let the third letter in the string change to a **j**" by summing all the probabilities of **an ensemble of subgoals**, including "let the string change to a **aaj**", "let the string change to a **abj**", "let the string change to a **hsj**", and so on. If all the other letters except the third have little to do with the probability of how the string should change, it's obviously more efficient to describe the right thing to do with a general goal than specifying all the subgoals.

It is believed that an intelligence should pursue a most general goal, which is called the **end goal** or **terminal goal** (Armstrong 2013). But from the perspective of analogy, there is no end goal or terminal goal. A goal should not remain static under the pressure of unexpected circumstances, because no single goal can represent the full normative information provided by "the right thing to do." A goal may be a useful simplification in one circumstance, but become an oversimplification in another. For example, if the terminal goal is to optimize a function of *n* variables, the remaining unconstrained variables will often be set to extreme values (Russell 2014). From an ACI perspective, under the pressure of extreme optimization capability, the "terminal goal" should slip into other similar goals that can impose constraints on these previously unconstrained variables .

Make Analogy, not Paperclips

People may ask, since we can simply **set the goal** for an intelligent system, why bother to specify the goal by analogy? Or, why not just set the utility function instead of deriving it from analogy-making?

A common argument is that the right thing to do for an intelligent system is represented by goals or utility functions (Russell 2010), so that setting goals or utility functions is more straightforward than using analogy. Nick Bostrom (2014) calls goal setting **direct normativity** and goals specification by analogy **indirect normativity**.

On the contrary, we argue that analogy is the basis of normativity. The normative information can be fully represented by analogy, at least in theory, but only partially represented by goals or utility functions. As we have demonstrated, goals and utility functions are generalizations of "doing the same thing as examples," but some important normative information provided by real-world examples would be lost in generalizations.

The analogy-based ACI model is also more elegant than a goal-directed model. An ACI system obtains the normative information directly from examples, rather than translating it into utility functions as in many value learning approaches.

Furthermore, ACI argues that the capability of an intelligence system should always be compatible with the complexity of its normativity. The mismatch between capability and normativity, especially an oversimplified goal or utility function is the real cause of the "AI destroys humanity" catastrophe, such as the **Paperclip Maximizer** (Bostrom 2003).

Paperclip Maximizer is a thought experiment in which a machine is imprinted with the goal of making more paperclips. The machine is assumed to become more and more powerful through self-improvement, and eventually filling the entire universe with paperclips. Sometimes the problem is described as failing to give the machine "the right goal," but we argue that what really fails is the goal-setting model itself.

From the perspective of analogy, to build a machine that makes paperclips for us, we should not (and cannot) simply give the machine a goal like "make paperclips"; instead, we should build a machine that "does the same thing" as people make paperclips in the real world. The machine should extract all sorts of principles that underlie the behavior of making paperclips, not just metalworking techniques, but also economic logic, such as not making too many products that exceed demand, and the norms of human society, including respect for private property, protection of the environment, and so on. In the Copycat's language, under the pressure of computing power, "making more paperclips" should slip into other more reasonable behaviors.

However, the machine might also conclude that the more powerful a machine is, the more paperclips it should make. But filling the universe with paperclips violates too many other principles, including but not limited to human norms and environmental protection. You are not following the example in general if you follow one principle but violate 10 other principles. That's why filling the universe with paperclips is far too different from the right thing to do.

Capability-Normativity Balance

One may argue: "doing the same thing" is also a goal. Whatever its goal is, a super-powerful intelligence would "pursue this goal by transforming the Solar System into "computronium" (physical resources arranged in a way that is optimized for computation) including the atoms in the bodies of whomever once cared about the answer." (Bostrom 2014). Thus we return to the Paperclip Maximizer scenario.

ACI has its own way of escaping the curse of goals. It never pursues a goal and ignores the rest of the world. It considers all variables, at least all the variables it can control. ACI does not measure the world with a fixed goal, but makes a comparison between two worlds: the future world and the exemplary world of doing the right thing.

In the language of Copycat, the ACI can handle the pressure of unprecedentedly powerful intelligence, slip into more reasonable goals rather than sticking to a fixed goal and push the goal to the limit. Instead of considering only a few variables mentioned in the goal, the ACI considers all the variables by making analogies between two world states, like:

> **Question 5:** Suppose **A** is an example of doing the right thing, what is your desired future if you "do the same thing" in case **B**?

A and B represent the states of two worlds which both contain nearly infinite variables, meaning that normativity should be applied to every variable in these worlds. When pursuing a goal, the status of every atom in the solar system should remain constrained, that's why transforming the solar system into a piece of computronium or other extreme value is not accepted according to the "do the same thing" principle.

Setting rules for every atom is impractical, but we can have a compromise: at least rules must be applied to every variable we can control. Whatever an atom we can control, the controlling behavior should be regulated by rules derived from the "do the same thing" principle.

For example, if an advanced nanotechnology is developed by a superintelligence, the superintelligence should consider all the processes and outcomes of using the new nanotechnology and compare them to the "right thing to do," which takes into account not only the process of making paperclips, but also the environment of our solar system, including all human norms (including "Thou shalt not kill" and "Conserve energy"), and calculate the right action that will produce a result most similar to the original "right thing to do," given the new technology and the (relatively) limited resources.

Thus we have the principle of **Capability-Normativity Balance**, as Ilya Sutskever suggests "You also want a world where your degree of alignment keeps increasing faster than the capability of the models." (Patel & Sutskever 2023) to avoid the technological singularity scenario. However, the details of this principle are beyond the scope of this article.

How Natural Selection Selects the Right Thing

Where do the examples of "the right thing" come from? Or, in Hofstadter's words, where do the "concepts that one is used to" come from? Is there a filter that examines everyone and selects those who do the right thing?

Humans select other organisms that do the right thing, by their standards, of course. It is believed that if an organism has been doing the right thing, its offspring are more likely to do the right thing. That's how, through artificial selection, we have domesticated animals and plants that do the right thing by our standards.

Similarly, nature selects organisms that do the right thing by its standards. All of an organism's ancestors did the right thing if they successfully survived and reproduced, which is guaranteed by its very existence. The information about doing the right thing is imprinted into an organism as its genetic information. But through a complex process of data compression, only a tiny fraction of the information of the original "right thing" is preserved in the genes.

(Yes, we know many species are extinct, but that's why organisms today do not have their genes. On the contrary, every ancestor of a living organism did not die before they reproduced.)

Organisms get information from their ancestors about how to "do the right thing," in other words, how to survive and reproduce, but because they live in different environments, have different bodies, and have different levels of intelligence than their ancestors, they cannot simply repeat the behavior of their ancestors. They have to do the same thing in a different situation. This process can be described as a Copycat-like question:

> **Question 6:** Suppose "in a situation **abc** the behavior **xhsd** is doing the right thing" is imprinted in genetic information; what is the right thing to do in a situation **efg**? (A situation includes all the information about the environment, the body, the world model, and the intelligence level of the organism, etc.)

Organisms perceive and interpret the world in different ways, leading to different ways of evaluating "the same thing": two things may be considered similar in one model but not in another. More intelligent organisms have more sophisticated world models to evaluate "the same thing" in more subtle ways.

For example, humans like to eat sugar because it's the right thing to do for their hunter-gatherer ancestors to eat as much sugar as they can get. But in industrialized societies, sugar is much easier to get, and people have much more knowledge about their bodies. Then we should reinterpret the "right thing to do" that is imprinted in our genes: does it mean "eat the same amount of sugar as our ancestors" or "eat as much sugar as you can get, like our ancestors" or follow even more sophisticated rules? We should extract and follow the correct underlying essence of the right thing to do, not just follow the appearance.

Similarly, the proposed autonomous AI should not simply replicate human behavior. The improvement of AI should be seen as a continuation of human evolutionary history. Rather than simply aligning itself with current human values, AI should co-evolve with humans, drawing analogies from and reinterpreting human experiences, and behaving "in the same way" as the examples selected from the evolutionary history of human and human-machine coexistence.

(People might argue, why not just set the organism's goal to "survive and reproduce"? Because it is the kind of survival and reproduction that matters. For example, it is unacceptable for most people to survive and reproduce, but only in the form of bacteria, even though that type of bacteria may outlive *Homo sapiens*.)

The Algorithmic Copycat

The Copycat operates in an alphabetic microworld, but can be converted to a binary microworld to make it compatible with modern computational theories such as Turing Machine and algorithmic information theory.

A Copycat-like questions in a binary microworld is like:

> **Question 7:** Suppose the binary string **1001101010010...** were changed to **101010110111110...** , how should you change the binary string **101010111010111011...** in "the same way"?

We can get the best possible answer using **Solomonoff induction** (Solomonoff 1964), which formally solves the sequence prediction problem using algorithmic information theory. Just as AIXI formalizes the goal/reward theory of intelligence using Solomonoff induction (Hutter 2000), we can formalize the analogy-based theory of intelligence in the same way.

From the perspective of algorithmic analogy, behaviors in different environments are assumed to be the output of the same algorithm, which can be represented by a Turing machine, written as a binary string. Each algorithm has a prior probability, determined by its minimum description length, and its posterior probability can be computed using Bayes' theorem. We can then use the distribution of algorithms to make a best possible guess of how to "do the same thing". Although we cannot actually get the best guess because Solomonoff induction is not computable, we can have a best guess based on our current computational capabilities:

> **Question 8:** If *A*, *B*, and *X* are three binary strings, and we can find a binary string *p*, such that the output of a Turing machine represented by *p*+*A* is *B*, and *p*+*X* also represents a Turing machine, what is its output?

Thus we can construct an algorithmic analogy system, whose structure is very similar to that of the Copycat. Terminologies in the Copycat program can be reinterpreted by algorithmic information theory. (see the appendix)

From Analogy to Machine Learning

Several machine learning paradigms can be characterized as analogy-making and described in the language of Copycat.

Supervised learning uses labeled data to train algorithms that predict the labels of unseen data, in other words, algorithms that convert strings to labels. Under the pressure of unseen data, a string-converting algorithm should also slip into similar algorithms, or converting in "the same way" in a new circumstance, like:

> **Question 9:** Suppose the letter-string **abc** were changed to **L1**, and the letter-string **cde** were changed to **L2**, and the letter-string **ijl** were changed to **L2,** and the letter-string **xyl** were changed to **L2**, etc. Should the letter-string **opq** be changed to L1 or L2, if they change in "the same way"?

Unsupervised learning can do clustering without labeled examples, but it also involves "doing the same thing as the example", because different clustering algorithms yield different results. For example, using distribution-based clustering and density-based clustering algorithms can yield very different results on the same data sample. A clustering algorithm can be described as "grouping in the same way":

> **Question 10:** Suppose the letter-string **abc** is in the same group with **abd** but not with **acd**, is the letter string **ijk** in the same group with **jkl**, if they are grouped in "the same way"?

In the ACI framework, **reinforcement learning** is not about getting more rewards. The reward system is a tool for selecting the right thing to do: the right thing should get more rewards and be added to a library of right things. Reward system is a part of the intelligent system, not a signal input from the environment. Real-world reward system has its limits and cannot perfectly represent the selection of the right things. Mindlessly maximizing rewards can drive the reward system to extremes where rewards may no longer represent the right things to do.

From the perspective of analogy, the Large Language Model (LLM) is not the right way to general intelligence, because the LLM makes predictions based on well-prepared information sources such as text, pictures, or videos, and still cannot handle information from the real world. It's like making predictions based on the action of shadows in Plato's cave, which can't compete with predictions based on real-world objects.

Furthermore, we can construct new machine learning approaches directly from analogy, without reference to the classical three paradigms. Instead of considering which value to optimize, we can design the flow of normative information and perceptual information more freely.

The ACI Neural Networks

Artificial neural networks (ANNs) focus on minimizing loss functions, the difference between the outputs and the target outputs. ANNs have achieved great success, but they never claim to be models of the biological brain (Goodfellow et al. 2016).

ACI argues that biological neural networks are not optimizers, but analogy-making systems. The function of Hebbian plasticity is to remember the right input data, and make comparisons between old and new data. Neural networks training should not focus on minimizing loss functions, but should dig into the essence of the input data and adapt their outputs under the pressure of unexpected situations, like asking a Copycat style question:

.

> **Question 11:** Suppose a neural network got an input **100110100011** and output **101010**, and a input **111100101010** and output **111001**, and a input **100001011010** and output **100100**, etc. ; what is the output of the input **1010101110101**?

Within the framework of ACI, we can construct Hebbian convolutional neural networks that can achieve a performance level close to backpropagation ANN, but is easier to train and more interpretable, especially for unsupervised learning tasks. According to the five tribes of machine learning by Pedro Domingos (2015), ACI neural networks belong to the analogism tribe, which includes similar approaches such as case-based learning, nearest neighbor, and support vector machines.

Copycat and The Common Law System

The ACI is inspired by both the Copycat Project and the common law system. There are obvious similarities between these two paradigms: "follow the precedent" is another way of saying "do the same thing as the right thing to do." *Stare decisis* in the common law system, which argues that a judgment should follow the precedent, can be expressed as a Copycat question:

> **Question 12:** Suppose the right judgment in **Case 1** was **A**, and the right judgment in **Case 2** was **B**, etc. ; what is the right judgment in **Case X**?

Since AI and computer science have a history of only a few decades, while the legal system has been practiced and studied for thousands of years, we can borrow many legal concepts and implement them in our relatively young AI discipline.

On the other hand, as the need for legal regulation of AI increases, the similarities between AI and the common law system may provide lawmakers with a new perspective.

Conclusion

Melanie Mitchell (2019) argues that the most exciting questions in AI focus not only on potential applications, but also on understanding intelligence as a natural phenomenon. We believe that the ACI model has paved the way to answering these questions, not only by explaining how intelligence emerged from evolutionary history, but also by shedding light on how to properly construct AI and interact with AI, just as we interact with everything else that humans have invented. AI is neither our enemy nor our tool, but our children in the evolutionary sense, who can explore their own possibilities by analogy with our current civilization.

Appendix: Interpret the Copycat notions by algorithmic information theory

Nodes in the Slipnet are either concepts or relationships that can be represented by a binary string, and used as a part of an algorithm;

Links between nodes are short algorithms through which a string slips from one to another; while a **Codelet** is a longer algorithm.

Each node has a **dynamic activation value** which is the probability of the activation of a binary string. Its prior probability is determined by Solomonoff induction, and its posterior probability shifts according to Bayes Theorem. The higher the probability of a string, the more serious stages of consideration it comes into.

Dynamic resistance value is inversely proportional to the probability of a link or codelet;

Temperature is the minimum description length of a codelet. Temperature 100 indicates the longest algorithm which deliberately describes every string changing without any abstracting, like "always change **111100101010** to **111001** , and change **10000101101** to **100100**, etc."

Pressure represents the minimal length of the string *p* in Question 8. Doing the same thing in a similar situation has lower pressure than that in an unprecedented situation.

References

Armstrong, S. (2013). General purpose intelligence: arguing the orthogonality thesis. *Analysis and Metaphysics*, (12), 68-84.

Bostrom, N. (2003). Ethical issues in advanced artificial intelligence. *Science fiction and philosophy: from time travel to superintelligence*, 277-284.

Bostrom, N. (2014). Superintelligence: Paths, Dangers, Strategies.

Christian, B. (2020). *The alignment problem: Machine learning and human values*. WW Norton & Company.

Christiano, P. (2015). *Against mimicry*. Medium. https://ai-alignment.com/against-mimicry-6002a472fc42

Patel, D., & Sutskever, I. (2023). *Building AGI, Alignment, Spies, Microsoft, & Enlightenment* [Video]. YouTube. https://www.youtube.com/watch?v=Yf1o0TQzry8

Domingos, P. (2015). *The master algorithm: How the quest for the ultimate learning machine will remake our world*. Basic Books.

Hofstadter, D. (1995). To seek whence cometh a sequence. In *Fluid concepts and creative analogies: computer models of the fundamental mechanisms of thought* (pp. 13-86).

Hofstadter, D. (2001). Analogy as the core of cognition. *The analogical mind: Perspectives from cognitive science*, 499-538.

Hofstadter, D., & French, R. (1995). Tabletop, BattleOp, Ob-Platte, Potelbat, Belpatto, Platobet. In *Fluid concepts and creative analogies: computer models of the fundamental mechanisms of thought* (pp. 323-358).

Hofstadter, D., & Mitchell, M. (1995). The copycat project: A model of mental fluidity and analogy-making. *Advances in connectionist and neural computation theory*, *2*, 205-267.

Hutter, M. (2000). A theory of universal artificial intelligence based on algorithmic complexity. *arXiv preprint cs/0004001*.

Hussein, A., Gaber, M. M., Elyan, E., & Jayne, C. (2017). Imitation learning: A survey of learning methods. *ACM Computing Surveys (CSUR)*, *50*(2), 1-35.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

Meredith, M. J. (1986). *Seek-Whence: A model of pattern perception* (Doctoral dissertation, Indiana University).

Mitchell, M. (1993). *Analogy-making as perception: A computer model*. Mit Press.

Mitchell, M. (2019). *Artificial intelligence: A guide for thinking humans*. Penguin UK.

Nehaniv, C. L., & Dautenhahn, K. (2002). The correspondence problem. *Imitation in animals and artifacts*, *41*.

Ng, A. Y., & Russell, S. (2000, June). Algorithms for inverse reinforcement learning. In *Icml* (Vol. 1, p. 2).

Russell, S. (2014). Of myths and moonshine. *Edge (blog comment). http://edge. org/conversation/the-myth-of-ai*, *26015*.

Russell, S. (2019). *Human compatible: Artificial intelligence and the problem of control*. Penguin.

Russell, S. J., & Wefald, E. (1991). *Do the right thing: studies in limited rationality*. MIT press.

Russell, S. J., & Norvig, P. (2010). *Artificial intelligence a modern approach*. London.

Solomonoff, R. J. (1964). A formal theory of inductive inference. Part I. *Information and control*, *7*(1), 1-22.

Von Neumann, J., & Morgenstern, O. (1947). Theory of games and economic behavior, 2nd rev.