herman
CAPPELEN

josh
DEVER

MAKING AI INTELLIGIBLE

philosophical foundations

OXFORD

MAKING AI INTELLIGIBLE

HERMAN CAPPELEN
AND JOSH DEVER

MAKING AI INTELLIGIBLE

Philosophical Foundations

OXFORD
UNIVERSITY PRESS

OXFORD
UNIVERSITY PRESS

Great Clarendon Street, Oxford, OX2 6DP,
United Kingdom

Oxford University Press is a department of the University of Oxford.
It furthers the University's objective of excellence in research, scholarship,
and education by publishing worldwide. Oxford is a registered trade mark of
Oxford University Press in the UK and in certain other countries

© Herman Cappelen and Josh Dever 2021

The moral rights of the authors have been asserted

First Edition published in 2021
Impression: 1

Some rights reserved. No part of this publication may be reproduced, stored in
a retrieval system, or transmitted, in any form or by any means, for commercial purposes,
without the prior permission in writing of Oxford University Press, or as expressly
permitted by law, by licence or under terms agreed with the appropriate
reprographics rights organization.

This is an open access publication, available online and distributed under the terms of a
Creative Commons Attribution – Non Commercial – No Derivatives 4.0
International licence (CC BY-NC-ND 4.0), a copy of which is available at
http://creativecommons.org/licenses/by-nc-nd/4.0/.

Enquiries concerning reproduction outside the scope of this licence
should be sent to the Rights Department, Oxford University Press, at the address above

Published in the United States of America by Oxford University Press
198 Madison Avenue, New York, NY 10016, United States of America

British Library Cataloguing in Publication Data
Data available

Library of Congress Control Number: 2020951691

ISBN 978–0–19–289472–4

DOI: 10.1093/oso/9780192894724.001.0001

Printed and bound in Great Britain by
Clays Ltd, Elcograf S.p.A.

Links to third party websites are provided by Oxford in good faith and
for information only. Oxford disclaims any responsibility for the materials
contained in any third party website referenced in this work.

CONTENTS

PART I: INTRODUCTION AND OVERVIEW

1. **Introduction** 3
   - The Goals of This Book: The Role of Philosophy in AI Research 3
     - An Illustration: Lucie's Mortgage Application is Rejected 4
   - Abstraction: The Relevant Features of the Systems We Will be Concerned with in This Book 10
   - The Ubiquity of AI Decision-Making 13
   - The Central Questions of this Book 17
   - 'Content? That's So 1980' 21
   - What This Book is Not About: Consciousness and Whether 'Strong AI' is Possible 24
   - Connection to the Explainable AI Movement 25
   - Broad and Narrow Questions about Representation 27
   - Our Interlocutor: Alfred, The Dismissive Sceptic 28
   - Who is This Book for? 28

2. **Alfred (the Dismissive Sceptic): Philosophers, Go Away!** 31
   - A Dialogue with Alfred (the Dismissive Sceptic) 35

PART II: A PROPOSAL FOR HOW TO ATTRIBUTE CONTENT TO AI

3. **Terminology: Aboutness, Representation, and Metasemantics** 51
   - Loose Talk, Hyperbole, or 'Derived Intentionality'? 53

Aboutness and Representation 54
AI, Metasemantics, and the Philosophy of Mind 56

4. Our Theory: De-Anthropocentrized Externalism 59
First Claim: Content for AI Systems Should Be Explained Externalistically 60
Second Claim: Existing Externalist Accounts of Content Are Anthropocentric 67
Third Claim: We Need Meta-Metasemantic Guidance 72
A Meta-Metasemantic Suggestion: Interpreter-centric Knowledge-Maximization 75

5. Application: The Predicate 'High Risk' 81
The Background Theory: Kripke-Style Externalism 82
Starting Thought: SmartCredit Expresses High Risk Contents Because of its Causal History 86
Anthropocentric Abstraction of 'Anchoring' 87
Schematic AI-Suitable Kripke-Style Metasemantics 88
Complications and Choice Points 90
Taking Stock 97
Appendix to Chapter 5: More on Reference Preservation in ML Systems 98

6. Application: Names and the Mental Files Framework 103
Does SmartCredit Use Names? 103
The Mental Files Framework to the Rescue? 105
Epistemically Rewarding Relations for Neural Networks? 108
Case Studies, Complications, and Reference Shifts 111
Taking Stock 116

7. **Application: Predication and Commitment** 117
Predication: Brief Introduction to the Act Theoretic View 118
Turning to AI and Disentangling Three Different Questions 121
The Metasemantics of Predication: A Teleofunctionalist Hypothesis 123
Some Background: Teleosemantics and Teleofunctional Role 125
Predication in AI 128
AI Predication and Kinds of Teleology 129
Why Teleofunctionalism and Not Kripke or Evans? 131
Teleofunctional Role and Commitment (or Assertion) 132
Theories of Assertion and Commitment for Humans and AI 133

PART III: CONCLUSION

8. **Four Concluding Thoughts** 139
Dynamic Goals 140
A Story of Neural Networks Taking Over in Ways We Cannot Understand 140
Why This Story is Disturbing and Relevant 144
Taking Stock and General Lessons 147
The Extended Mind and AI Concept Possession 148
Background: The Extended Mind and Active Externalism 148
The Extended Mind and Conceptual Competency 150
From Experts Determining Meaning to Artificial Intelligences Determining Meaning 150
Some New Distinctions: Extended Mind Internalist versus Extended Mind Externalists 151
Kripke, Putnam, and Burge as Extended Mind Internalists 152

Concept Possession, Functionalism, and Ways of Life 155
Implications for the View Defended in This Book 156
An Objection Revisited 157
Reply to the Objection 158
What Makes it a Stop Sign Detector? 158
Adversarial Perturbations 160
Explainable AI and Metasemantics 162

Bibliography 167
Index 173

PART I

INTRODUCTION AND OVERVIEW

1

INTRODUCTION

The Goals of This Book: The Role of Philosophy in AI Research

This is a book about some aspects of the philosophical foundations of Artificial Intelligence. Philosophy is relevant to many aspects of AI and we don't mean to cover all of them.[1] Our focus is on one relatively underexplored question: Can philosophical theories of meaning, language, and content help us understand, explain, and maybe also improve AI systems? Our answer is 'Yes'. To show this, we first articulate some pressing issues about how to interpret and explain the outputs we get

[1] Thus we are not going to talk about the consequences that the new wave in AI might have for the empiricism/rationalism debate (see Buckner 2018), nor are we going to consider—much—the question of whether it is reasonable to say that what these programs do is 'learning' in anything like the sense with which we are familiar (Buckner 2019, 4.2), and we'll pass over interesting questions about what we can learn about philosophy of mind from deep learning (López-Rubio 2018). We are not going to talk about the clearly very important ethical issues involved, either the recondite ones, science-fictional ones (such as the paperclip maximizer and Roko's Basilisk (see e.g. Bostrom 2014 for some of these issues)), or the more down-to-earth issues about, for example, self-driving cars (Nyholm and Smids 2016, Lin et al. 2017), or racist and sexist bias in AI resulting from racist and sexist data sets (Zou and Schiebinger 2018). We also won't consider political consequences and implications for policy making (Floridi et al. 2018).

Making AI Intelligible: Philosophical Foundations. Herman Cappelen and Joshua Dever, Oxford University Press (2021).
© Herman Cappelen and Joshua Dever. DOI: 10.1093/oso/9780192894724.003.0001

from advanced AI systems. We then use philosophical theories to answer questions like the above.

An Illustration: Lucie's Mortgage Application is Rejected

Here is a brief story to illustrate how we use certain forms of artificial intelligence and how those uses raise pressing philosophical questions:

> Lucie needs a mortgage to buy a new house. She logs onto her bank's webpage, fills in a great deal of information about herself and her financial history, and also provides account names and passwords for all of her social media accounts. She submits this to the bank. In so doing, she gives the bank permission to access her credit score. Within a few minutes, she gets a message from her bank saying that her application has been declined. It has been declined because Lucie's credit score is too low; it's 550, which is considered very poor. No human beings were directly involved in this decision. The calculation of Lucie's credit score was done by a very sophisticated form of artificial intelligence, called SmartCredit. A natural way to put it is that this AI system *says* that Lucie has a low credit score and on that basis, another part of the AI system *decides* that Lucie should not get a mortgage.

It's natural for Lucie to wonder where this number 550 came from. This is Lucie's first question:

> **Lucie's First Question**. What does the output '550' that has been assigned to me *mean*?

The bank has a ready answer to that question: the number 550 is a credit score, which represents how credit-worthy Lucie is. (Not very, unfortunately.) But being told this doesn't satisfy Lucie's

unease. On reflection, what she really wants to know is *why* the output means that. This is Lucie's second question:

> **Lucie's Second Question**: Why is the '550' that the computer displays on the screen an assessment of my credit-worthiness? What *makes* it mean that?

It's then natural for Lucie to suspect that answering this question requires understanding how SmartCredit works. What's going on under the hood that led to the number 550 being assigned to Lucie? The full story gets rather technical, but the central details can be set out briefly:

> *Simple Sketch of How a Neural Network Works*[2]
>
> SmartCredit didn't begin life as a credit scoring program. Rather, it started life as a general neural network. Its building blocks are small 'neuron' programs. Each neuron is designed to take a list of input data points and apply some mathematical function to that list to produce a new output list. Different neurons can apply different functions, and even a single neuron can change, over time, which function it applies.
>
> The neurons are then arranged into a network. That means that various neurons are interconnected, so that the output of one neuron provides part of the input to another neuron. In particular, the neurons are arranged into layers. There is a top layer of neurons—none of these neurons are connected to each other, and all of them are designed to receive input from some outside data source. Then there is a second layer. Neurons on the top layer are connected to neurons on the second layer, so that top layer neurons

[2] For a gentle and quick introduction to the computer science behind basic neural networks, see Rashid 2016. A relatively demanding article-length introduction is LeCun et al. 2015, and a canonical textbook that doesn't shirk detail and is freely available online is Goodfellow et al. 2016.

provide inputs to second layer neurons. Each top layer neuron is connected to every second layer neuron, but the connections also have variable weight. Suppose the top layer neurons T1 and T2 are connected to second layer neurons S1 and S2, but that the T1-to-S1 connection and the T2-to-S2 connections are weighted heavily while the T1-to-S2 connection and the T2-to-S1 connections are weighted lightly. Then the input to S1 will be a mixture of the T1 and T2 outputs with the T1 output dominating, while the input to S2 will be a mixture of the T1 and T2 outputs with the T2 output dominating. And just as the mathematical function applied by a given neuron can change, so can the weighting of connections between neurons.

After the second layer there is a third layer, and then a fourth, and so on. Eventually there is a bottom layer, the output of which is the final output of SmartCredit. The bottom layer of neurons is designed so that that final output is always some number between 1 and 1000.

The bank offers to show Lucie a diagram of the SmartCredit neural network. It's a complicated diagram—there are 10 levels, each containing 128 neurons. That means there are about 150,000 connections between neurons, each one labelled with some weight. And each neuron is marked with its particular mathematical transformation function, represented by a list of thousands of coefficients determining a particular linear transformation on a thousands-of-dimensions vector.

Lucie finds all of this rather unilluminating. She wonders what any of these complicated mathematical calculations has to do with why she can't get a loan for a new house. The bank continues explaining. So far, Lucie is told, none of this information about the neural network structure of SmartCredit explains why it's evaluating Lucie's creditworthiness. To learn about that, we need to consider the neural network's training history.

A bit more about how SmartCredit was created

Once the initial neural network was programmed, designers started training it. They trained it by giving it inputs of the sort that Lucie has also helpfully provided. Inputs were thus very long lists of data including demographic information (age, sex, race, residential location, and so on), financial information (bank account balances, annual income, stock holdings, income tax report contents, and so on), and an enormous body of social media data (posts liked, groups belonged to, Twitter accounts followed, and so on). In the end, all of this data is just represented as a long list of numbers. These inputs are given to the initial neural network, and some final output is produced. The programmers then evaluate that output, and give the program a score based on how acceptable its output was that measures the program's error score. If the output was a good output, the score is a low score; if the output was bad, the score is a high score. The program then responds to the score by trying to redesign its neural network to produce a lower score for the same input. There are a number of complicated mathematical methods that can be used to do the redesigning, but they all come down to making small changes in weighting and checking to see whether those small changes would have made the score lower or higher. Typically, this then means that a bunch of differential equations need to be solved. With the necessary computations done, the program adjusts its weights, and then it's ready for the next round of training.

Lucie, of course, is curious about where this scoring method came from—how do the programmers decide whether SmartCredit has done a good job in assigning a final output to input data?

The Scoring Method

The bank explains that the programmers started with a database of millions of old credit cases. Each case was a full demographic, financial, and social media history of a particular person, as well as a credit score that an old-fashioned human credit assessor had assigned to that person. SmartCredit was then trained on that data

set—over and over it was given inputs (case histories) from the data set, and its neural network output was scored against the original credit assessment. And over and over SmartCredit reweighted its own neural network trying to get its outputs more and more in line with the original credit assessments.

That's why, the bank explains, SmartCredit has the particular collections of weights and functions that it does in its neural network. With a different training set, the same underlying program could have developed different weights and ended up as a program for evaluating political affiliation, or for determining people's favourite movies, or just about anything that might reasonably be extracted from the mess of input social media data.

Lucie, though, finds all of this a bit too abstract to be very helpful. What she wants to know is why *she*, in particular, was assigned a score of *550*, in particular. None of this information about the neural architecture or the training history of SmartCredit seems to answer that question.

How all this applies to Lucie

Wanting to be helpful, the bank offers to let Lucie watch the computational details of SmartCredit's assessment of Lucie's case. First they show Lucie what the input data for her case looks like. It's a list of about 100,000 integers. The bank can tell Lucie a bit about the meaning of that list—they explain that one number represents the number of Twitter followers she has, and another number represents the number of times she has 'liked' commercial postings on Facebook, and so on.

Then they show Lucie how that initial data is processed by SmartCredit. Here things become more obscure. Lucie can watch the computations filter their way down the neural network. Each neuron receives an input list and produces an output list, and those output lists are combined using network weightings to produce inputs for subsequent neurons. Eventually, sure enough, the number '550' drops out of the bottom layer.

But Lucie feels rather unilluminated by that cascading sequence of numbers. She points to one neuron in the middle of the network and to the first number (13,483) in the output sequence of that neuron. What, she asks, does that particular number mean? What is *it* saying about Lucie's credit worthiness? This is Lucie's third question:

> **Lucie's Third Question**: How is the final meaningful state of SmartCredit (the output '550', meaning that Lucie's credit score is 550) the result of other meaningful considerations that SmartCredit is taking into account?

The bank initially insists that that question doesn't really have an answer. That particular neuron's output doesn't by itself mean anything—it's just part of a big computational procedure that holistically yields an assessment of Lucie's credit worthiness. No particular point in the network can be said to mean anything in particular—it's the network as a whole that's telling the bank something.

Lucie is understandably somewhat sceptical at this point. How, she wonders, can a bunch of mathematical transformations, none of which in particular can be tied to any meaningful assessment of her credit-worthiness, somehow all add up to saying something about whether she should get a loan? So she tries a different approach. Maybe looking at the low-level computational details of SmartCredit isn't going to be illuminating, but perhaps she can at least be told what it was in her history that SmartCredit found objectionable. Was it her low annual income that was responsible? Was it those late credit card payments in her early twenties? Or was it the fact that she follows a number of fans of French film

on Twitter? Lucie here is trying her third question again—she is still looking for other meaningful states of SmartCredit that explain its final meaningful output, but no longer insisting that those meaningful states be tied to specific low-level neuron conditions of the program.

Unfortunately, the bank doesn't have much helpful to say about this, either. It's easy enough to spot particular variables in the initial data set—the bank can show her where in the input her annual income is, and where her credit card payment history is, and where her Twitter follows are. But they don't have much to say about how SmartCredit then assesses these different factors. All they can do is point again to the cascading sequence of calculations—there are the initial numbers, and then there are millions upon millions of mathematical operations on those initial numbers, eventually dropping out a final output number. The bank explains that that huge sequence of mathematical operations is just too long and complicated to be humanly understood—there's just no point in trying to follow the details of what's going on. No one could hold all of those numbers in their head, and even if they could, it's not clear that doing so would lead to any real insight into what features of the case led to the final credit score.

Abstraction: The Relevant Features of the Systems We Will be Concerned with in This Book

Our concern is not with any particular algorithm or AI systems. It is also not with any particular way of creating a neural network. These will change over time and the cutting edge of programming today will seem dated in just a year or two. To identify what we

will be concerned with, we must first distinguish two levels at which an AI system can be characterized:

- On the one hand, it is an abstract mathematical structure. As such it exists outside space and time (it is not located anywhere, has no weight, and doesn't start existing at any particular point in time).
- However, when humans use and engage with AI, they have to engage with something that exists as a physical object, something they can see or hear or feel. This will be the **physical implementation** (or **realization**) of the abstract structure. When Lucie's application was rejected, the rejection was presented to her as a token of numbers and letters on a computer screen. These were physical phenomena, generated by silicon chips, various kinds of wires, and other physical things (many of them in different locations around the world).

This book is not about a particular set of silicon chips and wires. It is also not about any particular program construed as an abstract object. So we owe you an account of what the book is about. Here is a partial characterization of what we have in mind when we talk about 'the outputs of AI systems' in what follows:[3]

- The output (e.g. the token of '550' that occurs on a particular screen) is produced by things that are not human. The non-human status of the producer can matter in at least three ways:

First, these programs don't have the same kind of physical implementation as our brains do. They may use 'neurons', but their

[3] This is not an effort to specify necessary and sufficient conditions for being an AI system—that's not a project we think is productive or achievable.

neurons are not the same kind of things as our neurons—they differ of course physically (being non-biological), but also computationally (they don't process inputs and produce outputs in the same way as our neurons). And their neurons are massively different in number and arrangement from our neurons, and massively different in the way they dynamically respond to feedback.

Second, these programs don't have the same abilities as we do. We have emotional repertoires and sensory experiences they lack, and arguably have beliefs, desires, hopes, and fears that they also lack. On the other hand, they have computational speeds and accuracies that we lack.

Third, these programs don't have the same histories that we do. They haven't had the kind of childhoods we have had, and in particular haven't undergone the same experiences of language acquisition and learning that we have. In short, they are non-human (where we will leave the precise characterization of this somewhat vague and open-ended).

- When we look under the hood—as Lucie did in the story above—what we find is not intelligible to us. It's a black box. It will operate in ways that are too complex for us to understand. It's important to highlight right away that this particular feature doesn't distinguish it from humans: when you look under the hood of a human, what you will find is brain tissue—and at a higher level, what looks like an immensely complex neutral network. In that sense, the human mind is also a black box, but as we pointed out above, the physical material under the hood/skull is radically different.
- The systems we are concerned with are made by human programmers with their own beliefs and plans. As Lucie saw, understanding SmartCredit requires looking beyond the program itself to the way that the program was trained. But the training was done by people, who selected an initial range of data, assigned target scores to those initial training cases based on their own plans for what the program should track, and created specific dynamic methods for the program to adjust its neural network in the face of training feedback.

- The systems we are concerned with are systems that are intended to play a specific role, and are perceived as playing that role. SmartCredit isn't just some 'found artefact' that's a mysterious black box for transforming some numbers into other numbers. It's a program that occupies a specific social role: it was designed specifically to assign credit scores, and it's used by banks because it's perceived as assigning credit scores. It's treated as useful, as producing outputs that really are meaningful and helpful credit scores, and it becomes entrenched in the social role it occupies because it's perceived as useful in that way.

None of this adds up to a complete metaphysics of AI systems. That's not the aim of this book. Instead, we hope it puts readers in a position to identify at least a large range of core cases.

The Ubiquity of AI Decision-Making

SmartCredit raises concerns about what its outputs mean. But SmartCredit is only the tip of the iceberg. We are increasingly surrounded by AI systems that use neural network machine learning methods to perform various sorts of classifications. Image recognition software classifies faces for security purposes, tags photographs on social media, performs handwriting analysis, guides military drones to their targets, and identifies obstacles and street signs for self-driving cars. But AI systems of this sort aren't limited to simple classification tasks. The same underlying neural network programming methods give rise, for example, to strategic game-playing. Google's AlphaZero has famously achieved superhuman levels of performance in chess, Go, and Shogi. Other machine learning approaches have been applied to a wide variety of games, including video games such as Pac-Man, Doom, and

Minecraft.[4] Other AI systems perform variants of the kind of 'expert system' recommendation as SmartCredit. Already there are AI systems that attempt to categorize skin lesions as cancerous or not, separate spam emails and malware from useful emails, determine whether building permits should be granted and whether prisoners should receive parole, figure out whether children are being naughty or nice using video surveillance, and work out people's sexual orientations from photographs of their faces. Other AI systems use machine learning to make predictions. For example, product recommendation software attempts to extrapolate from earlier purchases to likely future purchases, and traffic software attempts to predict future locations of congestion based on earlier traffic conditions. Machine learning can also be used for data mining, in which large quantities of data are analysed to try to find new and unexpected patterns. For example, the data mining program Word2Vec extracted from a database of old scientific papers new and unexpected scientific conclusions about thermoelectric materials.

These AI systems are able to perform certain tasks at extraordinarily high levels of precision and accuracy—identifying certain patterns much more reliably, and on the basis of much noisier input, than we can, and making certain kinds of strategic decisions with much higher accuracy than we can—and both their sophistication and their number are rapidly increasing. We should expect that in the future many of our interactions with the world will be mediated by AI systems, and many of our current intellectual activities will be replaced or augmented by AI systems.

[4] See https://www.sciencenews.org/article/ai-learns-playing-video-games-starcraft-minecraft for some discussion about the state and importance of AI in gaming.

Given all that, it would be nice to know what these AI systems mean. That means we want to know two things. First, we want to know what the AI systems mean with their explicit outputs. When the legal software displays the word 'guilty', does it really *mean* that the defendant is guilty? Is guilt really what the software is tracking? Second, we want to know what contentful states the AI systems have that aren't being explicitly revealed. When AlphaZero makes a chess move, is it making it for reasons that we can understand? When SmartCredit gives Lucie a credit score of 550, is it weighing certain factors and not others?

If we can't assign contents to AI systems, and we can't know what they mean, then we can't in some important sense understand our interactions with them. If Lucie is denied a loan by SmartCredit, she wants to understand why SmartCredit denied the loan. That matters to Lucie, both practically (she'd like to know what she needs to change to have a better chance at a loan next time) and morally (understanding why helps Lucie not view her treatment as capricious). And it matters to the bank and to us. If we can't tell why SmartCredit is making the decisions that it is, then we will find it much harder to figure out when and why SmartCredit is making its occasional errors.

As AI systems take on a larger and larger role in our lives, these considerations of understanding become increasingly important. We don't want to live in a world in which we are imprisoned for reasons we can't understand, subject to invasive medical conditions for reasons we can't understand, told whom to marry and when to have children for reasons we can't understand. The use of AI systems in scientific and intellectual research won't be very productive if it can only give us results without explanations (a neural network that assures us that the ABC conjecture is true

without being able to tell us *why* it is true isn't much use). And things are even worse if such programs start announcing scientific results using categories that we're not sure we know the content of.

We are in danger, then, of finding ourselves living in an increasingly meaningless world. And as we've seen, it's a pressing danger, because if there is meaning to be found in the states and activities of these AI systems, it's not easily found by looking under the hood and considering their programming. Looking under the hood, all we see is jumbles of neurons passing around jumbles of numbers.

But at the same time, there's reason for optimism. After all, if you look under *our* hoods, you also see jumbles of neurons, this time passing around jumbles of electrical impulses. That hasn't gotten in the way of our producing meaningful outputs and having meaningful internal states. The hope then is that reflecting on how *we* manage to achieve meaning might help us understand how AI systems also achieve meaning.

However, we also want to emphasize that it's a guarded hope. Neural network programs are a little like us, but only a little. They are also very different in ways that will come out in our subsequent discussion. Both philosophy and science fiction have had an eye from time to time on the problem of communicating with and understanding aliens, but the aliens considered have never really been all that alien. In science fiction, we get the alien language in Star Trek's Darmok,[5] which turns out to be basically English with more of a literary flourish, the heptapod language of 'Story of Your Life',[6] which uses a two-dimensional syntax to

[5] See *Star Trek: The Next Generation*, season 5 episode 2.

[6] In Chiang, *Stories of Your Life And Others*, Tor Books, 2002. The book was the inspiration for the film Arrival.

present in a mildly encoded way what look like familiar contents, and the Quintans of Stanislaw Lem's 1986 novel *Fiasco*, who are profoundly culturally incomprehensible but whose occasional linguistic utterances have straightforward contents. In philosophy, consideration of alien languages either starts with the assumptions that the aliens share with us a basic cognitive architecture of beliefs, desires, reasons, and actions, or (as Davidson does) concludes that if the aliens aren't that much like us, then whatever they do simply can't count as a language.

Our point is that the aliens are already among us, and they're much more alien than our idle contemplation of aliens would have led us to suspect. Not only that, but they are *weirdly* alien—we have built our own aliens, so they are simultaneously alien and familiar. That's an exciting philosophical opportunity—our understanding of philosophical concepts becomes deeper and richer by confronting cases that take us outside our familiar territory. We want simultaneously to explore the prospect of taking what we already know about how familiar creatures like us come to have content and using that knowledge to make progress in understanding how AI systems have content, and also see what the prospects are for learning how the notions of meaning and content might need to be broadened and expanded to deal with these new cases.

The Central Questions of this Book

Philosophy can help us understand many aspects of AI. There are salient moral questions such as whether we *should* let AI play these important social roles. What are the moral and social

consequences of letting AI systems make important decisions that throughout our history have been made by humans who could be held accountable? There are also pressing questions about whether advanced AI systems could eventually make humans superfluous—this is sometimes discussed under the label 'existential risk' of AI (see Bostrom 2014). None of these is the topic of this book.

The questions we will be concerned with have to do with **how we can interpret and understand the outputs of AI systems**. They are illustrated by the questions that Lucie asked the bank in our little story above. Recall Lucie's first question:

> **Lucie's First Question**: What does the output '550' that has been assigned to me mean?

Lucie's first question is a question about how to understand a specific output of a specific program. We're not going to try to answer Lucie's question, or even to give particular tools for answering this kind of question. But we are interested in the meta-question about whether Lucie's question is a reasonable and important one. We've already observed that AI systems are frequently used *as if* questions like Lucie's made sense and had good answers—we treat these systems *as if* they are giving us specific information about the world. It's thus important to consider whether there is a sensible way to think about these programs on which questions like Lucie's first question could eventually be answered.

This perspective leads to Lucie's second question:

> **Lucie's Second Question**: Why is the '550' that the computer displays on the screen an assessment of my credit-worthiness? What *makes* it mean that?

Our central interest in this book starts with examining what kinds of answers this question could have. If the output states of AI systems do mean something, then surely there must be some *reason* they mean what they do. If we could at least figure out what those reasons are, we might be better positioned down the road to answering Lucie's first question.

The bank tried one particular method of answering Lucie's second question: they directed Lucie to the details of SmartCredit's programming. As we saw, this method wasn't obviously successful—learning all the low-level neural network details of SmartCredit's programming didn't seem to give a lot of insight into why its outputs meant something about Lucie's credit worthiness.

But that was just one method. Our central project is to emphasize that there are many other methods that are worth considering. One way to think about the project is to remember that humans, too, are content-bearing. Our outputs, like SmartCredit's outputs, at least prima facie, mean things and carry information about the world. But looking inside our skulls for an explanation of those contents isn't likely to be much more illuminating than looking inside SmartCredit's programming code was. We emphasized above that programs like SmartCredit are different from people in many important ways, and that's worth keeping in mind (and will guide much of our discussion below). But at the same time, both we and machine-learning programs like SmartCredit are systems producing outputs based on some enormously complicated and not obviously illuminating underlying computational procedure.

That fact about *us*, though, hasn't stopped us from assigning contents to people's outputs, and it hasn't stopped us from

entertaining theories about why people's outputs mean what they do. It's just forced us to consider factors other than neurocomputational implementation in answering that 'why' question. Theories about why human outputs mean what they do have appealed to mental states, to causal connections with the environment, to normative considerations of coherence and charity, to biological teleology, and to relations of social embedding. One of our central projects is then to see whether these kinds of theories can be helpfully deployed in answering Lucie's second question, and how such theories might need to be adapted to accommodate the differences between people and programs.

Lucie had a third question:

> (**Lucie's Third Question**): How is the final meaningful state of SmartCredit (the output '550' meaning that Lucie's credit score is 550) the result of other meaningful considerations that SmartCredit is taking into account?

Eventually we want a good theory of content for AI systems. A good theory of content for people needs to do more than just assign contents to the things we say—it also needs to assign contents to 'hidden' internal states of beliefs and desires, which then help make sense of, and perhaps constrain the contents of, the things we say. We should be open to the possibility that it's the same for AI systems. SmartCredit 'says' some things—it produces explicit outputs of the form of the '550' evaluation it outputs for Lucie. But in making sense of why SmartCredit's explicit outputs have the meanings that they do, we might want to attribute additional contentful states to the program—for example, we might (as Lucie does) want to be able to attribute to SmartCredit various reasons that led it to assign Lucie the credit score that it did.

On a more abstract level: AI systems produce various outputs, and we can always ask what, if anything, makes it the case that an AI system has a certain output; and AI systems produce those outputs for various reasons, and we can ask whether those reasons are contentful reasons (rather than just irreducibly complicated mathematical computations), and what, if anything, makes it the case that the reasons have the contents that they do.

The underlying facts are not in dispute: ML (machine learning) systems are (or consist of) massively complex algorithms that generate an enormous neural network with thousands or millions of interconnected 'neurons'. It is also beyond dispute that in many cases the overall structure and dynamics of that system is too complex for any human to comprehend. A burning question is now: when this system produces an output consisting of an English sentence like the examples given above, how can that output mean what those English words mean? How can we know that it tells us something about what we call creditworthiness?

'Content? That's So 1980'

A central aim in this book is to encourage increased interaction between two groups. First, AI researchers, who are producing machine learning systems of rapidly increasing sophistication, systems that look to have the potential to take on or supplement many of our ordinary processes of reasoning, deciding, planning, and sorting. And second, philosophers, who work in a rich intellectual tradition, which provides tools for thinking about content, tools directed both at determining what features of a system make it contentful (and in what ways) and at characterizing different

kinds of contents with a variety of formal tools. We want to encourage that interaction because we think that AI has a content problem—we need to be able to attribute contents to AI systems, but we're currently poorly positioned to do so.

A certain view of the history of AI research can make all of that seem like a confused retrograde step. AI researchers *tried* approaches centred around content and representation. That's what the symbolic artificial intelligence program was about, that's what led to endless projects focused on a small block world based on clear representational systems. But the wave of contemporary successes in AI has been won by *moving away* from the symbolic approaches. Neural network machine learning systems are deliberately designed *not* to start with a representational system—the whole goal is to allow data that hasn't been pre-processed into representational chunks to be filtered by neural network systems in a way that isn't mediated by representational rule systems and still produce powerful outputs. So if what we're suggesting in this book is a return to symbolic AI, and a move away from the machine learning successes, contemporary AI researchers would be understandably uninterested. (For an introduction to this sort of old-school philosophical theorizing inspired by old-school AI theorizing, see Rescorla 2015.)

But that's not what we are suggesting. Our point, in fact, is that philosophy brings to the table a collection of tools designed to find content in the wild, rather than building content into the architecture. The central problem in the philosophical study of content is this: when people go about in the world, encountering and interacting with various objects, making various sounds, having various things going on inside their heads, a bunch of contents

typically result. Some of the sounds they make have contents; some of the brain states they are in have content. Philosophical accounts of content want to explain what makes that the case: what needs to be going on so that some sounds are contentful and others are not; what needs to be going on so that some sounds mean that it's raining and other sounds mean that it's sunny.

People, of course, are the original neural network systems. So the philosophical project of content must be compatible with application to neural networks. That's because the philosophical project isn't to *build* contentful systems by setting them up with the right representational tools, but rather to *understand* the contents that we find 'in the wild'. Work in philosophy of language and formal semantics has indeed produced very sophisticated mathematical models of representation. But the philosopher's suggestion isn't that we should take those models and use them in designing good humans. We (philosophers who work on the theory of content) are not proposing that babies be pre-fitted with Montague semantics, or that axiomatic theories of meaning be taught in infancy. We just want to understand the content that certain complex systems (like people) carry, whatever the causal and historical story about how they came to carry that content.

So even if the history of AI research has made you a representation/content pessimist, we encourage you to read on. We think that intellectual engagement between philosophy and AI research has the promise of letting you have your non-content-oriented design tools and your *post facto* content attribution, too. And, we want to suggest, that's a good thing, because content plays crucial roles, and AI systems that lie wholly outside the domain of content won't give us what we want.

What This Book is **Not** About: Consciousness and Whether 'Strong AI' is Possible

There's an earlier philosophical literature on AI that we want to distance ourselves from. In influential work, John Searle (1980) distinguished between what he called Strong and Weak AI. Strong AI, according to Searle, has as a goal to create thinking agents. The aim of that research project is to create machines that *really* can think and have other cognitive states that we humans have. Searle contrasted this with the weak AI project according to which the aim was to create machines that have the *appearance* of thinking (and understanding and other cognitive states). Searle's central argument against Strong AI was the Chinese Room Argument. There's now a very big literature on the soundness of that argument (and also on how to best present the argument—for some discussion and references, see Cole 2014).

The project of this book will not engage with Searle-style arguments and we are not interested in the Strong vs Weak AI debate.

Our starting point and methodology are different from the literature in that tradition: Our goal in the first four chapters of Part I is to use contemporary theories of *semantics* and *meta-semantics* to determine whether (and how) ML systems could be interpreted. We take some of the leading theories of how language has representational properties and see what those theories have to say about ML systems. In most cases they are mixed: there's some match with what we are doing and some mismatch—and then we suggest fixes. In Chapter Four we suggest that maybe the right attitude to take is that we need to revise our meta-semantics to accommodate ML systems. Rather than use anthropocentric theories of content (i.e. theories of content based on how human

language gets content) to determine whether ML systems have content, we should revise our theories of content attribution so that ML systems can be considered representational (in effect revising what representation is, so that ML systems can be accommodated).

This strategy contrasts with the argumentative strategy exemplified by Searle's Chinese Room argument (and the tradition arising from that argument): the idea behind that strategy is to use reflections on a cluster of thought experiments to settle, once and for all, the question of whether machines can understand and have a semantics. This book doesn't engage with and only indirectly takes a stand on that form of argument.

Connection to the Explainable AI Movement

In 2018, the European Union introduced what it calls the General Data Protection Regulation. This regulation creates a 'right to explanation' and that right threatens to be incompatible with credit scores produced by neural networks (see Kaminski 2019 Goodman and Flaxman 2017, Adadi and Berrada 2018) because, as we pointed out above, many ML systems make decisions and recommendations without providing any explanation of those decisions and recommendations. Without explanations of this sort, ML systems are uninterpretable in their reasoning, and may even become uninterpretable in their results.

The burgeoning field of explainable AI (XAI) aims to create AI systems that are *interpretable* by us, that produce decisions that come with comprehensible *explanations*, that use *concepts* that we can understand, and that we can *talk to* in the way that we can

engage with other rational thinkers.[7] The opacity of ML systems especially highlights the need for artificial intelligence to be both explicable and interpretable. As Ribero et al. (2016:3–4) put it, 'if hundreds or thousands of features significantly contribute to a prediction, it is not reasonable to expect any user to comprehend why the prediction was made, even if individual weights can be inspected'. But the quest for XAI is hampered both by implementation difficulties in extracting explanations of ML system behaviour and by the more fundamental problem that it is not clear what exactly explicability and interpretability *are* or what kinds of tools allow, even in-principle, achievement of interpretability.

Doshi-Velez and Kim (2017) state the goal of interpretability as being 'to explain or present [the outputs of AI systems] in understandable terms' (2017:2) and proceed to point to real problems in modelling explanations. What they do not note is that both *understanding* and *terms/concepts* require at least as much clarification as explanation. As they point out, much work in this field relies on 'know it when you see it' conceptions of these core concepts. This book aims to show how philosophy can be used to remedy this lacuna in the literature.

The core chapters in this book aim to present proposals for how we can attribute content to AI systems. We return to the implications of this for the explainable AI movement towards the end of the book.

[7] A recent discussion in philosophy is Páez 2019. Outside of philosophy, some recent overviews of the literature are Mueller et al. 2019 and Addadi and Berrada 2018. For particular proposals about how to implement XAI, see Ribeiro et al. 2016, Doshi-Velez & Kim 2017, and Hendricks et al. 2016. For an approach that uses some ideas from philosophy to explain 'explanation', see Miller 2018.

Broad and Narrow Questions about Representation

We should note one limitation of our approach: we focus on whether the outputs of AI systems are content-bearing. In the little story about Lucie, we ask what the output '550' means. We are interested in whether that token can and should be considered contentful—in Chapter Three we put this as the question of whether there's aboutness in the AI system. This is closely connected to, but distinct from, a broader issue: Does the AI system have the ability to reason? Does it have a richer set of beliefs and so a richer set of contents? We can also ask: what is the connection between being able to represent the thought that Lucie's credit score is 550, and having a range of other thoughts about Lucie? Can a system have the ability to think only *one* thought, or does that ability by necessity come with a broader range of representational capacities? These are crucial questions that we will return to in the final chapter. Prior to that, our goal is somewhat more narrow and modest: Can we get the idea of content/representation/aboutness for AI systems off the ground at all? Are there any plausible extensions of existing meta-semantic theories that opens the door to this? Our answer is yes. In the light of that positive answer, the broader questions take prominence: how much content should be attributed? What particular content should be attributed to a particular AI system? Does SmartCredit understand 'credit worthiness', and also 'credit' and 'worthiness', and grasp the relevant compositional rule? Does understanding 'credit' involve an understanding of *money*, *borrowing*, *history*, etc? These are questions that become pressing, if the conclusions in this book are correct.

Our Interlocutor: Alfred, The Dismissive Sceptic

A character called Alfred is central to the narrative of this book. Alfred, we imagine, is someone whose job it is to make AI systems. He is very sceptical that philosophers can contribute to his work at all. In the next chapter, Alfred is having a conversation with a philosopher. Alfred argues that while he thinks talking to philosophers is a bit interesting, it is basically useless for him. According to Alfred, philosophers have nothing substantive to contribute to the development of AI.

Alfred will return at several junctions in this book. In writing this book (and thinking through these issues), Alfred has been very useful to us—we hope he is also of some interest to readers (and especially those potential readers who are entirely unconvinced that AI will profit from an injection of philosophy).

Who is This Book for?

The intended audience for this book are readers interested in starting to think how philosophy can help answer important questions about interpretable AI. They have some knowledge of philosophy, some knowledge of AI, and are interested in how to use the former to reflect on the latter.

There are some people who should not buy or read this book:

- If you are looking for a technical book that engages in great detail with the formal aspects of neural networks, then this book is not for you.

- If you're looking for a book that develops in detail a new theory about the nature of meaning, then this book is also not for you.
- If you're looking for a complete theory of interpretable AI then, unfortunately, you've also bought (or borrowed or downloaded) the wrong book.

Our goals are modest. We hope the book will help frame some important issues that we find surprisingly little literature on. AI raises very interesting philosophical questions about interpretability. This book tries to articulate some of those issues and then illustrate how current philosophical theories can be used to respond to them. In so doing, it presupposes some knowledge of philosophy, but not very much. Our hope is that it can be used even by upper-level undergraduate students and graduate students not expert in either AI or philosophy of language. We hope it will inspire others to explore these issues further. Finally, we hope it opens up a door between researchers in AI and the philosophy of language/metaphysics of content.

2

ALFRED (THE DISMISSIVE SCEPTIC)

Philosophers, Go Away!

In the previous chapter, we outlined a range of interesting philosophical challenges that arise in connection with understanding, explaining, and using AI systems. We tried to make the case that philosophical insight into the nature of content (and the difference between a system having content and simply being evidence of some sort) is centrally important both for understanding AI and for deciding how we should integrate it into our lives.

When we first started work on this project, we got in touch with people in the AI community. We thought that our work on these issues should be informed by people working in the field—those who are actually developing ML systems. When we approached people, they were friendly enough. We had many helpful conversations that have improved this book. However, it soon became clear to us that the people working at the cutting edge of AI (and in particular those working for the leading corporations) considered us, at best, lunchtime entertainment—a bit like reading an interesting novel. There was a sense that no one really took these

Making AI Intelligible: Philosophical Foundations. Herman Cappelen and Joshua Dever, Oxford University Press (2021).
© Herman Cappelen and Joshua Dever. DOI: 10.1093/oso/9780192894724.003.0002

philosophical issues *seriously*. Here is a caricatured summary of the attitude we encountered:

> Look, these big-picture concerns just aren't where the action is. I don't really care whether this program I'm working on is 'really a malignant mole detector' in any deep or interesting sense. What I care about is that I'm able to build a program that plays a certain practical role. Right now I can build something that's pretty decent at the role. Of course there's a long way to go. That's why I spend my time thinking about how adding back propagation, or long short term memory, or exploding gradient dampening layers, or improved stochastic gradient descent algorithms, will lower certain kinds of error rates. If you have something actually helpful to say about a piece of mathematics that will let me lower error rates, or some mathematical observations about specific kinds of fragility or instability in the algorithms we're currently using, I'm happy to listen. But if not, I'm making things that are gradually working better and better, so go away.

We take this dismissive reaction very seriously and much of this book is an attempt to reply to it. We are not going to dismiss the dismissal. At the end of the book, we have not refuted it. There's something to it, but, we argue, it's an incomplete picture and we outline various ways in which it is unsatisfying.

It's worth noting that this pragmatic-sceptic's dismissal of philosophy has analogues in almost all practical and theoretical domains. Practising mathematicians don't worry much about the foundations of their disciplines (they don't care much about what numbers are, for example). Politicians don't care much about theories of justice (they don't spend much of their time reading Rawls, Nozick, or Cohen). Those making medical decisions with massive moral implications don't spend much time talking to moral philosophers. And so it goes. There's a very general

question about what kind of impact philosophical reflection can have. One way to read this book is as a case study of how philosophers should reply to that kind of anti-philosophical scepticism.

We should, however, note that not all those working in AI share Alfred's dismissive attitude towards increased reflection on the foundations of ML systems. In 2017, Google's Ali Rahimi gave a talk where he compared the current state of ML systems to a form of *alchemy*: programmers create systems that work, but they have no real, deep, understanding of *why* they work. They lack a foundational framework. Rahimi said:

> There's a self-congratulatory feeling in the air. We say things like 'machine learning is the new electricity.' I'd like to offer an alternative metaphor: machine learning has become alchemy.[1]

It's become alchemy because the ML systems work, but no one really understands *why* they work the way they do. Rahimi is not entirely dismissive of making things that work without an understanding of why it works: 'Alchemists invented metallurgy, ways to make medication, dying techniques for textiles, and our modern glass-making processes.' Sometimes, however, alchemy went wrong:

> …alchemists also believed they could transmute base metals into gold and that leeches were a fine way to cure diseases. To reach the sea change in our understanding of the universe that the physics and chemistry of the 1700s ushered in, most of the theories alchemists developed had to be abandoned.

[1] https://www.youtube.com/watch?v=x7psGHgatGM.

More generally, Rahimi worries that when we have ML systems that contribute to decision making that's crucial both to individuals and to societies as a whole, a foundational understanding would be preferable.[2]

> If you're building photo sharing services, alchemy is fine. But we're now building systems that govern health care and our participation in civil debate. I would like to live in a world whose systems are built on rigorous, reliable, verifiable knowledge, and not on alchemy.

Many in the AI community dismissed Rahimi's pleading for a deeper understanding. Facebook's Yann LeCun replied to Rahimi saying that the comparison to alchemy was not just insulting, but wrong:

> Ali complained about the lack of (theoretical) understanding of many methods that are currently used in ML, particularly in deep learning. Understanding (theoretical or otherwise) is a good thing…But another important goal is inventing new methods, new techniques, and yes, new tricks. In the history of science and technology, the engineering artifacts have almost always preceded the theoretical understanding: the lens and the telescope preceded optics theory, the steam engine preceded thermodynamics, the airplane preceded flight aerodynamics, radio and data communication preceded information theory, the computer preceded computer science.[3]

We will argue that Rahimi is right: the current state of ML systems really is a form of alchemy—and not just for the reasons Rahimi mentions. The one important reason is that the field lacks an

[2] Note: Rahimi's worry is not specifically about interpretability, but the same point applies.

[3] https://www2.isye.gatech.edu/~tzhao80/Yann_Response.pdf.

understanding of how to describe the content of what it has created (or how to describe what it has created as deprived of content). We are presented with AI as if it is something that can talk to us, tell us things, make suggestions, etc. However, the people making AI have no theory that justifies that contentful presentation of their product. They have given us no rational argument for that contentful presentation. They've just written some algorithms and they have no deeper understanding of what those pieces of mathematics really amount to or how they are properly translated into human language or affect human thoughts. If the view is that these programs have no content at all, then that too is a substantive claim that needs justification: What is content such that these systems don't have it?

So: welcome to the world of philosophy. It's a world where there's very little certainty. There are many alternative models, the models disagree, and there's no clear procedure for choosing between them. This is the kind of uncertainty that producers and consumers of AI will have to learn to live with. It's only after a refreshing bath in philosophical uncertainty that they will start to come to grips with what they have made.

A Dialogue with Alfred (the Dismissive Sceptic)

Alfred: I appreciate the interest you philosophers have in these issues. It's important that a broad range of disciplines reflect on the nature of AI. However, my job is to make exactly the kinds of AI systems that you talk about in the introduction of this book and I don't get it. I just don't see that there's anything you philosophers can tell me about interpretation that will help me do my

job. We've made all these amazing advances, and we did it without you. I'm not doubting that there's some interesting meta-reflections around these issues, but that's just lunch entertainment for us. It makes no real difference to what we do, day to day. Issues about the nature of interpretation and the nature of content don't seem pressing to me in my professional life.

So, as a conversation starter, let me try this: philosophical theories of meaning and language make no difference to what we do. For professional purposes, we can ignore them.

Philosopher: I don't see how you can avoid those issues. What do you think is going on with SmartCredit, then? We give the software access to Lucie's social media accounts, and it spits out the number 550. But so far, that's just pixels on a screen. The output of the program is useless until we know that 550 *means* a high risk of default. We need to know how to look at a program and figure out what its outputs mean. That's absolutely central to our ability to make any use of these programs. We can't just ignore that issue, can we?

Alfred: Of course we say things like, 'That output of 550 means that Lucie is a high risk of default.' But that's just loose talk—we don't need to take it seriously. All that's really going on is this. SmartCredit is a very sophisticated tool. It takes in thousands of data points and sorts and weighs them using complicated and highly trained mathematical algorithms. In the end SmartCredit spits out some number or other. That number doesn't in itself *mean* anything. It's just a number—just the end product of millions of calculations. Of course the bank should then take that number into account when deciding whether to extend a loan to Lucie. But not because the number *means* that Lucie is a default

risk—rather, because the number is the output of a highly reliable piece of software.

Philosopher: Wait, I'm not sure I understand what you're proposing. Just recently I went to the doctor and he used a machine learning program called SkinVision to evaluate a mole on my back.[4] According to him, SkinVision said that the mole was likely to be malignant, so he scheduled surgery and removed it. Are you telling me that the doctor was wrong and that SkinVision didn't say anything about my mole? I guess then I had surgery for no reason. Or what about the case of Eric Loomis? Loomis was found guilty of participating in a drive-by shooting, and was sentenced to six years in prison in part because, according to the judge, the machine learning program COMPAS said that Loomis was a high risk to reoffend.[5] Are you telling me that the judge was wrong and that COMPAS didn't say anything about Loomis's recidivist risk? If that's right, surely it was a huge injustice to give Loomis more prison time. It looks like we're treating these programs *as if* they are saying things all over the place, and making many important and high-stakes decisions based on what we think they are saying. If that's all wrong, and the programs aren't really saying anything, don't we need to do some serious rethinking of all of this technology?

Alfred: I think you're making a mountain out of a molehill here. Again, it's just loose talk to say that *COMPAS says that Loomis is high risk* or to say that *SkinVision says that your mole is probably malignant*. But that doesn't mean we're taking important actions for no reason. *SkinVision* didn't say that your mole was probably

[4] See https://www.skinvision.com. **Alfred**: Wait, we can talk in footnotes?

[5] https://www.wired.com/2017/04/courts-using-ai-sentence-criminals-must-stop-now.

malignant, but your doctor did say that. He said it in a sloppy way—he used the words 'SkinVision says that your mole is probably malignant'—but we don't need to take his exact phrasing seriously. It's clearly just his way of telling you (himself) about your mole. And there's no worry about having a mole removed because your doctor says that it's probably malignant, is there? The same with COMPAS. COMPAS didn't say that Loomis was high risk—the judge did. Again, he said it in a sloppy way, but we all know what's going on. And there's nothing wrong with giving someone a severe sentence because a judge says that he's a high recidivism risk, is there? That kind of thing happens all the time.

Philosopher: That's helpful. So the idea is that all the meaning and content is in the people saying things in response to the programs, not in the programs themselves. That's why we don't need a theory of content for the programs. (Hopefully we can get a good theory of content for people—but in any case that's not a special problem for thinking about AI systems.) But I'm still worried about how this idea is going to be worked out. My doctor gives SkinVision a digital photograph of my mole, and it produces a printout that says 'Malignancy chance = 73%'. Then my doctor says that my mole is probably malignant. On your view, SkinVision didn't say anything, and its printout didn't have any content—all the saying and all the content is coming from the doctor. But it sure seems like quite a coincidence that there's such a nice match between what my doctor *really meant* and what the words printed by SkinVision *seemed to me* but (on your view) didn't really mean.

Alfred: Of course it's not a coincidence at all. The designers of SkinVision included a helpful user interface so that doctors would know what to say when they got the results of a SkinVision analysis. There's nothing essential about that—SkinVision could have

been designed so that it just outputs a graph of a function. But then doctors would have needed more training in how to use the program. It makes sense just to have the programmers add on some informative labelling of the outputs on the front end and save the doctors all that work.

Philosopher: 'Informative labelling'—I like that. You can't have informative labelling without information. Doesn't that then require that the outputs of SkinVision do mean something, do carry the information that (for example) the mole is probably malignant?

Alfred: Good point. OK, what I should have said was not that it's the *doctor* who's the one who's really saying something—rather, it's the *programmer* who's really saying something. When SkinVision prints out 'Malignancy chance = 73%', that's the programmer speaking. She's the one who is the source of the meaning of those words. They mean what they do because of her programming actions, not because of anything about the SkinVision program itself. SkinVision is then just a kind of indirect way for the programmer to say things. That's a bit weird, I admit, but there are lots of other forms of indirect announcement like that. When the programmer writes some code which, when run, prints 'Hello World', it's the programmer, not the program, who greets the world. SkinVision and other AI systems are just more complicated versions of the same thing. The doctor then *also* says that your mole is probably malignant, but that's just the doctor passing on what the programmer indirectly said to him.

Philosopher: That's an interesting idea. But I'm worried that it has a strange consequence. Suppose that the programmer of SkinVision had been in a perverse mood when programming the final user interface, and had set things up so that the mathematical

output that in fact leads to SkinVision printing 'Malignancy chance = 73%' instead caused SkinVision to print 'Subject is guilty of second degree murder'. Would that then mean that SkinVision, rather than a piece of medical software, was instead a bit of legal software, making announcements about guilt or innocence rather than malignant or benign statuses?

Alfred: What? Of course not. Why would you even think that? SkinVision's whole training history shaped that neural network into a medical detector, not a legal detector. How would a perverse programmer implementing perverse output messages change that?

Philosopher: Well, doesn't it follow from what you said? If SkinVision itself isn't really saying anything, and it's just a tool for letting the programmer speak, then if the programmer chooses to have it produce the words 'Suspect is guilty of second degree murder', what's said (by the programmer, through the program) is that the suspect is guilty of second degree murder. And if the information conveyed is legal, rather than medical, then it looks like a piece of legal software.

Alfred: Not a very good piece of legal software! The guilt and innocence announcements it produces aren't going to have anything to do with whether the person is really guilty. You can't tell guilt or innocence from a photograph of a mole. And even if you could, SkinVision hasn't been trained to do so.

Philosopher: Agreed, it would be a terrible piece of legal software. But my point is just that that's what it would be, since its outputs mean what the programmer wants them to mean. I can see that in this case there's some plausibility to the claim that when the perversely programmed SkinVision prints 'Subject is guilty of second-degree murder', what's said is that the subject

is guilty of second-degree murder. (Whether it's SkinVision itself or the programmer who's saying this is less clear to me.) But I'm worried that that's a special feature of this example. In this particular case, the programmer has decided to put the program output in the form of words in a pre-existing language. It's thus very tempting to take that output to mean whatever those words mean in the language. In the same way, if a monkey banging on a keyboard happens to type out 'To be or not to be, that is the question', we might feel some inclination to say that the monkey has said something. But probably that feeling should be resisted, and we should just say that the *sentence* means something, and that the monkey has accidentally and meaninglessly produced it.

Consider another case. StopSignDetector is another machine learning neural net intended to be used in self-driving autonomous vehicles. The plan for StopSignDetector was, not surprisingly, to have it be a stop sign detector, processing digital images from a car camera to see if there is a stop sign ahead. But StopSignDetector doesn't print out 'There is a stop sign', or anything like that. There's just a little red light attached to the computer that blinks when the program reaches the right output state. As I understand your view, the blinking red light doesn't mean anything in itself, but is just a device for the programmer saying that there is a stop sign. That's because, I guess, the programmer intends the blinking red light to announce the presence of a stop sign. But now add in the perverse programmer. What if the programmer decides instead that the blinking red light should announce the presence of a giraffe—but doesn't change anything in the code of StopSignDetector. Does that mean that we end up with a very bad giraffe detector?

Alfred: I think all of this is getting much more complicated than it really needs to be. We speak sloppily as if these programs are saying things, producing outputs that somehow represent specific facts about the world. That's all just sloppy speech. In many cases, that sloppiness can be fixed up by taking us *really* to be talking about what the end user (like the doctor or the judge) is saying or what the original programmer is saying. But sure, I agree that in weird cases when end users or programmers have weird secret plans, that's not a good way to fix up our sloppy talk. But it's not that hard to find a different way, is it?

Think about your standard pocket calculator. You push the buttons '58 + 67' on the keyboard, and on the display it shows '125'. Does that mean that the calculator is *saying* that 58 plus 67 is 125? Surely not—there's no need for that kind of content talk. Of course, someone using the calculator might then say '58 + 67 = 125', and thereby mean (as people do) that 58 plus 67 is 125. And it's presumably not an *accident* that the calculator display looks the way it does—the original programmer of the calculator software chose that display format because of their plan that the calculator be a tool to announce arithmetic facts. But even if we discovered that the programmer had strange secret plans and the calculator user had strange secret interpretive ideas, it wouldn't matter. That's because in the end the calculator is just a tool for getting at mathematical results. So long as the calculator is working correctly, who really cares what anyone's communicative plans are, or what the calculator or anyone else is 'really saying'.

Philosopher: But I'm not sure a calculator is the right comparison for you. The programming of a calculator is a straightforward example of symbolic representational programming. If we look into the coding details of the calculator, we will indeed be able to

find the parts of the program that represent numerical values, and that represent the applications of various mathematical operations to those numerical values. Here, it looks entirely natural to me to say that the calculator display really does mean that 58 plus 67 is 125. None of the special features of (for example) SmartCredit that made its contents so obscure seems to be present in this case.

Alfred: OK, fair enough. But I bet I could program up a machine learning pocket calculator if I really set my mind to it. I bet you haven't actually checked out the coding of your TI-Nspire—would you really change anything in how you used the calculator if you discovered that it had a neural network implementation?

Philosopher: Probably not. But that's because I would think that, whether neural network or not, the calculator's program was *about* mathematical operations. Remember, I'm not a sceptic about the role of content in these cases, you are. I'm happy to say that we don't need to worry about obscure communicative plans on the part of the programmer or the user, because I'm happy to say that the program itself means something. (Of course, I think it's a very hard question *why* it means something, and I think in some cases we might have a lot of trouble figuring out *what* it means.) So what's your view on this? Don't you need a view on what it means to say that the calculator is a 'tool for getting at mathematical results'? That looks an awful lot like a disguised claim about the contents of the calculator claims.

Alfred: That's got to be too fast. A hammer is a tool for pounding in nails, right? That's not a claim about the meaning or content of a hammer. That's just an observation about what hammers are useful for.

Philosopher: Agreed. But I think this overlooks an important distinction. A hammer isn't an informational tool. When we use a

hammer, we're not trying to learn anything—we're just trying to get something done (get some nails in some wood). It's not too surprising if we don't need any notion of content to explain that kind of tool. But a language is also a tool, isn't it? And to say *that* kind of tool, we need to talk about contents. That's because language is an informational communicative tool, a tool that we're using to learn things. So we need to say what sentences mean to see what we can learn from them. And SkinVision and COMPAS look like tools of the same sort. We're not trying to *do* something with those tools—all of the *doing* is by the doctor or the court. We're just trying to get some information out of the tools. And if we're going to get information out, we need a contentful interaction with the program.

Alfred: Good, that helps me see what I want to say. In the end, the tools I want to make are more like hammers than like languages. Consider an example. I want to build a self-driving car. I'm not trying to make a car that I'll learn something from—I just want a car that will do something for me. I want a car that I can get into and that will then take me to the right place. That's a big project, so I'm not trying to do it all at once. Along the way, I produce a machine learning image recognition program that will beep when there's a pedestrian in the road. For now, that can be a helpful signal to the driver. But eventually, I'll have that bit of programming integrated into a larger autonomous vehicle program. Once that's all done, all I care about is that the car won't in fact hit pedestrians. Whether the beeps from that one part of the program 'mean that there's a pedestrian in the road' makes no difference to me. Why would I care? I'm not trying to give anyone any information with that beeping; I'm just trying to make sure that the car doesn't crash.

Philosopher: I see the idea, but how does that help with other cases? Maybe we don't need to assign contents to the full self-driving car, but before the pedestrian detector is integrated into the full car, while it's being used to warn human drivers, don't we need its beeps to *mean* that there's a pedestrian in the road?

Alfred: I don't see why. I'm happy to think of the driver in the same way that I think of the self-driving car. I'm not interested in getting any particular *contents* to the driver. What I care about is that the driver swerves when the program beeps. So long as that happens, and the pedestrian isn't hit, I'm happy.

Philosopher: I see. So you're just thinking of the programs as little causal prods that push people into the right kind of activity. SkinVision just needs to cause doctors to perform surgeries; never mind what the doctors believe. COMPAS just needs to cause judges to issue severe sentences; never mind what judges might learn from COMPAS.

Alfred: Right. Sure, probably the best way to get doctors to perform surgeries under the right conditions is to get them to believe that people need surgeries under those conditions. But that's just an accidental feature of doctors making them different from nails. The thing that really matters is just that our program causally prompts the right things to happen.

Philosopher: I'm not sure this 'it's all just causal prods' idea is going to be as easy to work out as you seem to think. You said you just wanted 'a car that I can get into and that will then take me to the right place'. But where did this notion of 'right place' come from? That requires that the car takes you where you want to go, and that then requires that you are able to *tell* the car where to go. But doesn't that still require a contentful interaction with the program? Maybe it's on the input side rather than the output side,

but the issues seem to me to be the same—I need to be able to do something to the program that I can count on putting the program into the right state. I need to be confident that when I tell the self-driving car to take me to the airport, its subsequent driving will be guided by the content of what I told it.

Alfred: I'm tempted to say that the problem you're pointing out is just another artefact of our being only part-way through the overall project. I already agreed that *for now* I want the pedestrian detector's beeps to be understood by human drivers as signalling that there is a pedestrian in the road. We're talking about understanding and meaning here because the programming project isn't finished yet, so we can't just let the car do its own self-driving business. But the same is true for the need to give the car directions. Down the road, the goal should be a car that you don't need to give directions to. The car will figure out how to deal with pedestrians in the road; it will also figure out how to deal with a passenger in the car. Maybe it will access your calendar and determine where you ought to be and automatically take you there.

Philosopher: Wait, 'figure out'? 'Determine'? 'Access your calendar'? That all looks like content-based talk.

Alfred: Sure, but it's all dispensable in the same way. When I say that the car will figure out how to deal with pedestrians in the road, I just mean it won't hit pedestrians in the road. When I say the car will figure out how to deal with a passenger in the car, I just mean that it will take that passenger to a location where the passenger ought to be. And so on.

Philosopher: I'm not sure I like the vision of the AI future you're sketching here. These days when I get in the car and drive somewhere, I have plans and reasons for what I'm doing and I perform a bunch of deliberate intentional actions in pursuit of my

goals. Your self-driving car takes that all away from me. I don't need any plans, or any reasons for going anywhere. I just get in the car, and the car takes me somewhere that will work out well for me. It feels like a Wall-E future, with all of us passive passengers on the Axiom. It's important to us that we have reasoned engagement with the world—aren't you proposing to shrink that reasoned engagement down to nothing, by embedding us in a network of devices that just causally push us around to where we ought to be?

Alfred: Well, as long as you're really getting where you ought to be, is it really that bad? We're surrounded by lots of systems and devices that take care of our needs without our reasoned engagement. When you're exposed to germs, your immune system just takes care of it for you—it causally pushes bits of your body into the right places without any intervention by you. Things wouldn't be any better if you had to reason your way through a viral infection, would they?

Philosopher: Fair enough, although just because something is good in some places doesn't mean it's good everywhere. But surely there's also a real issue about whether we can count on the self-driving car taking us where we *ought* to be. What's our source of confidence in that 'ought'? Either we're just building into the program what the right final goals are (get us where our calendar says we ought to be), in which case it looks like we still need content tracking with the program. Or we've got the kind of advanced AI that has the ability to reshape the categories it's been trained to track, in which case, if there's no notion of content of the program's reshaped categories, I'm not sure why we should be confident that what it's doing is in any sense getting us where we *ought* to be.

Alfred: Look, all of this is getting extremely speculative. Forget about this utopian/dystopian picture in which our AI systems just shepherd us through the world. Remember, I've already observed that you can think of human users now as being like the eventual self-driving car. I don't care whether the car *knows* that there's a pedestrian in the road and *takes that into account*. All I care about is that when the pedestrian detector beeps, the car changes course. And similarly for the human user. I don't care whether the human user *knows* that there's a pedestrian in the road and *takes that into account*. All I care about is that when the pedestrian detector beeps, the driver changes course. Who cares what the underlying mechanism is by which that happens?

Philosopher: There's a sense in which I agree with all of that. Forget about programs entirely, and just think about people. There's some sense in which all of the content talk we go in for may be optional. Maybe we can stop thinking about other people as creatures having beliefs and desires and plans with contents and making claims with contents, and just think about them as lumbering obstacles to be manipulated and manoeuvred around. But surely *something* is gained by instead thinking about people as bearers of content. If we've at least reached the point, then, of saying that content talk for AI systems is exactly as dispensable as content talk for people, I think we've got enough to motivate some careful thinking about how to make that content talk work out in the AI case.

Alfred: Fair enough. Let's at least see what you've got.