

Full Self-Driving Skynet, and Other Artificial Intelligence Myths

The realities of decision making with deep machine learning models

Brad Flaughner

January 6, 2023

We have now accumulated sufficient evidence to see that whatever language the central nervous system is using, it is characterized by less logical and arithmetical depth than what we are normally used to.

– John von Neumann [1]

Preface

This book is an attempt to organize my brain on paper. I have been thinking about decision making with computers, and the computer's relationship to the brain for too long now, and before I get dementia (I'm 35) I would like to put a stake in the ground and tell everyone where they can stick their thoughts about the future.

I am simultaneously very hopeful for the future, and horrified by the way that people talk about it. I think many so-called "Data Scientists" are full of shit when they talk, and some of them know it but would like to keep their jobs. I think that many investors are snowed by fancy language and toy demos that they extrapolate into the future. Users of digital products are woefully ignorant about how the products work and how they fit into a larger ecosystem. I'm simultaneously amused and annoyed. I also have time to set the record straight, so I'll try.

This book contains strong beliefs, weakly held. If I say something stupid I'll eat my hat and update the book if I have time, or just trash the thing and say "please don't read this, consider it my diary from 2023". However, for now I think I am right and I think you should listen to me for a moment.

I am a self-employed programmer and financially secure, these are my credentials. I have no corporate master and I am not raising money to create the future of AI, so have no bias towards boosterism. I write programs close enough to the bleeding-edge that I can see the present as fast as almost anyone.

So let's go on this journey together, I'll try and explain to you (with words, code and examples) how modern AI works, and when you should be scared (self-driving cars without the appropriate infrastructure) or when you should not be scared (please shut up about Skynet becoming self-aware). I'll also try do case studies in as many arenas that I think are interesting. If you have notes for me you can reach me at brad@bradflaughter.com.

Please skim this book, please skip around and read some stories. I think the first chapter is important for everyone to know. The rest of the stories can be chosen a la carte. Enjoy.

Brad Flaugher

Contents

Preface	iii
Contents	iv
1 History: from explicit rules, to big data and statistics	1
1.1 "Textbook" AI in 1997	1
1.2 Does AI need to know grammar to translate?	1
1.3 Explicit rules and codified human knowledge	2
1.4 IBM tries every possible chess move	3
1.5 Statistical analysis of handwriting is the way of the future	4
1.6 Less programmer intelligence and more data intelligence	4
1.7 From explicit rules to a black box, and beyond	5
2 Self-driving with statistics	7
3 A Shakespearean comedy of numbers	8
3.1 Knowledge Representation, or not	8
4 Derivative artworks of the future?	9
5 The data is the hardest part	10
6 The police and Big Tech are profiling me!	11
7 The useful chaos of spaghetti code	12
8 Self-stabilizing concept drift	13
9 My horse drives itself, thanks!	14
10 Skynet: did you try unplugging it?	15
APPENDIX	16
A ????	17
Bibliography	18
Notation	19
Alphabetical Index	20

List of Figures

1.1	"a frustrated programmer writing boring rules on his computer" made with Stable Diffusion 2.1 . .	3
1.2	Garry Kasparov	3
1.3	"from explicit rules, to a black box and beyond" made with Stable Diffusion 2.1	5
2.1	The Mona Lisa	7

List of Tables

List of Listings

History: from explicit rules, to big data and statistics

1

1.1 "Textbook" AI in 1997

Dr. Elaine Rich's textbook on Artificial Intelligence, published in the 1980s, was a groundbreaking work that helped to establish many of the foundational concepts and techniques in the field of AI. However, the rapid advancements in AI over the past few decades have led to many of the chapters in this textbook becoming obsolete.

One of the main reasons for this is the prevalence of deep learning, big data, and large-scale statistical models in modern AI. These techniques have largely replaced the symbolic, rule-based approach to AI that was emphasized in the textbook, making many of the chapters on knowledge representation and expert systems less relevant.

Additionally, the explosion of data and the availability of powerful computing resources have made it possible to apply machine learning techniques at a scale that was previously unimaginable. This has led to the development of highly effective machine learning models that can handle complex tasks such as image and speech recognition with a high degree of accuracy, making many of the chapters on simpler machine learning techniques such as decision trees¹ and linear regression less relevant. [2]²

We'll discuss this history and a few examples from the "early days" of AI to help us understand where we are headed. We'll start with machine translation, then discuss chess and finally neural networks, which will be the focus of the rest of this book.

1.2 Does AI need to know grammar to translate?

Noam Chomsky is a linguist and philosopher who has made significant contributions to the field of linguistics with his theory of universal grammar. Chomsky believes that all human languages share a common underlying structure, and that this structure is innate to humans. He proposes that this innate structure is the result of a "language acquisition device" present in the human brain, which allows us to learn and produce language. Chomsky also argues that the structure of language is largely independent of its content, and that the ability to produce and understand language is a fundamental aspect of human nature. His theory has been influential in the field of linguistics and has sparked much debate and research on the nature of language and its relationship to the human mind.

For English speakers or anyone who has learned English as a second language you'll have many examples of special cases, irregular verbs, bad english and former street slang that became good and proper over time. For

1.1 "Textbook" AI in 1997	1
1.2 Does AI need to know grammar to translate?	1
1.3 Explicit rules and codified human knowledge	2
1.4 IBM tries every possible chess move	3
1.5 Statistical analysis of handwriting is the way of the future	4
1.6 Less programmer intelligence and more data intelligence	4
1.7 From explicit rules to a black box, and beyond	5

1: Although mathematically, [Neural Networks are Decision Trees](#)

[2]: Rich et al. (2009), *Artificial Intelligence*

2: the book is now in its third edition and unlikely to be updated as Dr. Rich as retired utexas.edu

programmers this is a nightmare, how can we codify human knowledge in a timely fashion? If we tried to write the rules of the English language in code (which many have tried to do) the rules themselves might change before we were finished writing them.

Explicitly translating languages through code is a difficult task because it requires a thorough understanding of the grammar, vocabulary, and syntax of both languages, as well as the nuances and subtleties of their respective cultures³. Simply coding rules for how to translate words or phrases from one language to another is not sufficient, as there are often multiple valid translations for a given phrase depending on the context in which it is used.

A more effective approach to translation is to use statistical techniques that rely on a large corpus of translated data, such as Canadian laws⁴. This type of data-driven approach involves training a machine learning model on a large dataset of translations, allowing it to learn the patterns and relationships between the languages. The model can then use this knowledge to make educated translations of new phrases or sentences, taking into account the context in which they are used.

While this approach is not perfect, it has proven to be highly effective in machine translation and can produce accurate translations even for languages that are very different from each other. The use of a large dataset of translations also allows the model to learn from the mistakes and variations present in real-world translations, further improving its accuracy.

1.3 Explicit rules and codified human knowledge

When we "teach" a computer to perform a task by explicitly writing down all of the rules of that task, we are really codifying human understanding.⁵ When we codify human understanding we write down every rule that we know explicitly. For small tasks we can do this with 100 percent accuracy, and only minor headache on the part of the software developer.

For example, let's write a boring function to tell you the number of days for a given month.

```
def days_in_month(year, month):
    if month in [1, 3, 5, 7, 8, 10, 12]:
        return 31
    elif month in [4, 6, 9, 11]:
        return 30
    elif month == 2:
        if (year % 4 == 0 and year % 100 != 0) or year % 400 == 0:
            return 29
        else:
            return 28
    else:
        return "Invalid month"
```

3: For programmers this is a nightmare, how can we codify human knowledge in a timely fashion? If we tried to write the rules of the English language in code (which many have tried to do) the rules themselves might change before we were finished writing them.

4: They're in French AND English, which is useful data that we can use to correlate phrases and transform English to French and vice-versa.

5: Programming this way makes some software development totally boring, I almost switched my major in college to math after considering what a life would look like manually writing rules for handling "edge cases" for the rest of my natural life.

Writing code can be a tedious and repetitive task, especially when it comes to debugging and testing. It can be especially frustrating when you're working on a large project and you're trying to track down a specific bug that's causing the program to crash. Testing code can also be boring, as it often involves running the same tests over and over again to ensure that the code is working correctly.

Additionally, writing code can be boring because it requires a lot of concentration and focus. It can be easy to get lost in the details and lose track of time, especially if you're working on a complex problem. It can also be challenging to come up with creative solutions to problems, and it can be frustrating when your code doesn't work as expected.

While writing and testing code can be rewarding and fulfilling, it can also be a tedious and boring process. It requires a lot of patience, persistence, and attention to detail, and it can be easy to get frustrated and lose motivation. However, with practice and perseverance, it is possible to overcome these challenges and find enjoyment in the process of writing and testing code.

AI has traditionally operated by explicitly codifying human knowledge into machine-readable formats by doing the boring job of coding. This approach, which I'm calling "codified human knowledge" relies on humans to carefully structure and organize information in a way that can be understood by the AI system. The AI system then uses this structured knowledge to make decisions and perform tasks.

However, recent advances in AI have largely ignored the knowledge representation problem and instead have focused on using statistical techniques and neural networks to automatically learn patterns and relationships in data. This approach, known as "deep learning," involves training large neural networks on vast amounts of data, allowing the AI system to make educated classifications and transformations of data without explicit human guidance.

Deep learning has proven to be highly effective in a variety of applications, such as image and speech recognition, and has contributed to the rapid progress we have seen in AI in recent years. However, the reliance on large amounts of data and the lack of transparency in these models can make it difficult to understand how they are making decisions, which can be a concern in certain applications (hence the title of this book).

1.4 IBM tries every possible chess move

Deep Blue was a revolutionary computer developed by IBM that was specifically designed to play chess at the highest level. It was programmed with a vast database of chess knowledge and was able to analyze millions of positions per second.

Garry Kasparov was the reigning world chess champion at the time, and he was considered to be one of the greatest players in history. He had never lost a match to a computer before, and he was confident that he would be able to defeat Deep Blue.

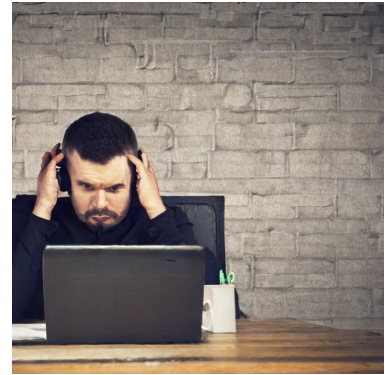


Figure 1.1: "a frustrated programmer writing boring rules on his computer" made with Stable Diffusion 2.1

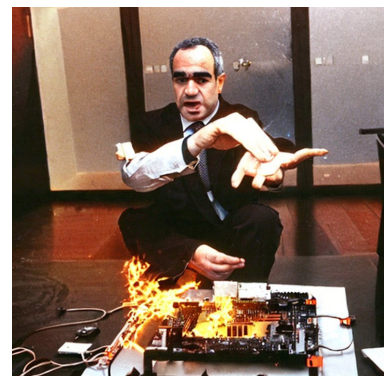


Figure 1.2: "Garry Kasparov setting a computer on fire" made with Stable Diffusion 2.1

However, things did not go as Kasparov had expected. Deep Blue was able to analyze the positions on the board with incredible speed and accuracy, and it was able to come up with highly sophisticated strategies that Kasparov had never seen before.

Despite Kasparov's best efforts, he was no match for the sheer brute force of Deep Blue's computational power. In the end, Deep Blue emerged victorious, defeating Kasparov in a historic match that changed the world of chess forever.

Deep Blue was a turning point in the development of AI, but Deep Blue's methods (namely calculating every possible outcome of a Chess game to determine the best move) was not suitable for many of the world's problems. It turns out that Chess is fun, but the world is not like chess.⁶ The "real"⁷ future of AI was being developed elsewhere, using statistics and a toy model of the brain to solve a very practical problem for banks.

6: if you would like to read an example of the simplest chess engine that I could imagine that is written in a similar "codified" way as Deep Blue check out [sunfish](#).

7: We might go back, and try again to more explicitly code everything up, and in some cases we still need to, but from the author's perspective, we live in a deep learning/neural network world.

1.5 Statistical analysis of handwriting is the way of the future

It was the early 1990s and Yann LeCun was a researcher at Bell Labs in New Jersey. At the time, the process of reading and processing checks was a tedious and time-consuming task that was done manually by bank employees. LeCun saw the potential for using artificial intelligence to automate this process, and he began experimenting with using convolutional neural networks (CNNs) to recognize patterns in images of checks.

At the time, CNNs were a relatively new type of neural network that had been developed in the 1980s for image recognition tasks. They were inspired by the structure of the human visual system, and were able to process images in a way that was similar to how the human brain does.

LeCun's work was groundbreaking, and he was able to achieve impressive results using CNNs to process checks. By 1993, he had developed a system that was able to read and process checks with a high degree of accuracy, significantly reducing the amount of time and effort that was required to process checks manually.⁸

LeCun's work on using CNNs for check processing was a major milestone in the field of artificial intelligence, and it laid the foundation for the development of many other applications of CNNs in the years that followed. Today, CNNs are widely used in a variety of applications, including facial recognition, image classification, and natural language processing.⁹

8: CNN digit OCR models are frequently featured in beginner training tutorials for deep learning libraries like PyTorch and Tensorflow, check one out [on Github](#).

9: check out Yann LeCun demonstrating a convolutional neural network in 1993 at [youtube.com](#).

1.6 Less programmer intelligence and more data intelligence

I think it's useful to separate the knowledge in the AI problem-space into two groups. The data and the programmer together make the programs

that we use every day, and for the rest of this book I'll try and separate the discussion of the smarts of each to help us better understand the world.

10

Early AI relied heavily on a human programmer to design, write, and debug computer programs. Good programmers needed domain expertise, problem-solving skills, logical thinking, and the ability to learn and adapt to new programming languages and technologies.

We are now in the age of big data, and everyone knows that "data is gold". Statistical AI methods that are now most prevalent rely on extracting meaningful insights and knowledge from large datasets. This involves using statistical and analytical methods to discover patterns and trends in data, and using this information to inform business decisions or solve problems.

Throughout this book I'll discuss the interaction between programmer and data, and what can go wrong. Working with big data and statistics at a large scale has given AI tremendous ability, but has made understanding and testing models infinitely more difficult. It is your authors belief that understanding the nuance of this interaction between programmer and data is essential to understanding modern AI.

1.7 From explicit rules to a black box, and beyond

Artificial intelligence (AI) has come a long way since its inception, and the way that it makes decisions has changed significantly over time. In the early days of AI, explicit rules were used to tell the AI system what to do in certain situations. These rules were often written by humans and encoded into the system, and the AI would follow them to make decisions.

However, with the advent of deep learning, we have started to rely on a statistical understanding of the truth for AI to make decisions. Deep learning is a type of machine learning that involves training artificial neural networks on large datasets. These neural networks are able to learn patterns and relationships in the data, and can use this knowledge to make decisions.

The use of deep learning has led to the development of powerful AI that is able to perform tasks that were previously thought to be impossible for a machine to handle. For example, deep learning has led to the development of AI systems that are able to recognize faces, translate languages, and even beat humans at complex games like chess and Go.

While deep learning has led to significant advances in AI, it has also made it harder to debug and understand how the AI system is making its decisions. With explicit rules, it was relatively easy to understand why the AI made a particular decision. However, with deep learning, it is often difficult to understand exactly how the AI arrived at its decision. This can make it challenging to troubleshoot problems with the AI system and to ensure that it is making decisions that are fair and unbiased.

10: I'll also repeatedly encourage you to use the words "algorithm" and "artificial intelligence" sparingly. They'll confuse your thinking, I promise.

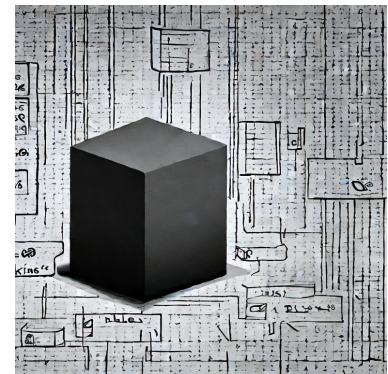


Figure 1.3: "from explicit rules, to a black box and beyond" made with Stable Diffusion 2.1

In conclusion, AI has come a long way since its early days, and the way that it makes decisions has changed significantly over time. While explicit rules were once used to tell the AI what to do, we now rely on a statistical understanding of the truth for AI to make decisions. This has led to the development of powerful AI that is able to perform a wide range of tasks, but it has also made it harder to debug and understand how the AI is making its decisions.

Self-driving with statistics

2

Hardware got amazing, we gave up teaching the way we teach ourselves and let the data do the work

We leveraged huge statistical models to regress our way to success

We used building blocks of regression and neurons to train huge models

These models are statistical and deterministic, but ultimately chaotic black boxes..

TODO talk about these books [3] [4] [5] [6]

Talk about ChatGPT, deterministic vs probabilistic and Thomas Hobbes
1



Figure 2.1: The Mona Lisa.
https://commons.wikimedia.org/wiki/File:Mona_Lisa,_by_Leonardo_da_Vinci,_from_C2RMF_retouched.jpg

A Shakespearean comedy of numbers

3

It's all numbers man!!!!

3.1 Knowledge Representation, or
not 8

3.1 Knowledge Representation, or not

This is some text and a link to Hey if you want to site something on the side use[7]

"AI Scientists disagree as to whether these language networks possess true knowledge or are just mimicking humans by remembering the statistics of millions of words. I don't believe any kind of deep learning network will achieve the goal of AGI if the network doesn't model the world the way the brain does. Deep learning networks work well, but not because they solved the knowledge representation problem. They work well because they avoided it completely, relying on statistics and lots of data instead. How deep learning networks work is clever, their performance impressive, and they are commercially valuable. I am only pointing out that they don't possess knowledge and, therefore, are not on the path to having the ability of a five-year-old child." [8]

```
cd myproject
docker run tensorflow
#profit!
```

tex.stackexchange.org for help.

[7]: Andreu et al. (2021), *Humans won't be able to control a superintelligent AI, according to a study*

Derivative artworks of the future?

4

GPT-3, BERT and Bloom

Link some cool shit here, Draw Owl!

Who owns this shit anyway? Copilot and FSF plus lawsuits

Prove it, asshole!

I can make a "model" that behaves like a database. Just memorizes shit

Talk about AI upscaling and AI as lossy compression. we stored the entire corpus of human knowledge, but AI retrieves it in a silly way.

The data is the hardest part

5

You are essentially programming with data, so if your data sucks so will your prediction, you also really can't generalize, only correlate.

are you predicting the right thing? Are you really predicting how valuable the company is or just whether it'll be the next meme stock?

"I hope for some sort of peace—but I fear that machines are ahead of morals by some centuries and when morals catch up there'll be no reason for any of it." Harry Truman, 1945 [9]

Representation, "fixing the training set" [10], or the Impossibility of Fairness from a model.

TODO talk about these books [11] [12] [13] [10]

"The second requirement of goal-misalignment risk is that an intelligent machine can commandeer the Earth's resources to pursue its goals, or in other ways prevent us from stopping it... We have similar concerns with humans. This is why no single person or entity can control the entire internet and why we require multiple people to launch a nuclear missile. Intelligent machines will not develop misaligned goals unless we go to great lengths to endow them with that ability. Even if they did, no machine can commandeer the world's resources unless we let it. We don't let a single human, or even a small number of humans, control the world's resources. We need to be similarly careful with machines." [8]

```
cd myproject
docker run tensorflow
#profit!
```

tex.stackexchange.org for help.

[9]: McCullough (1992), *Truman*

[10]: Christian (2020), *The Alignment Problem: Machine Learning and Human Values*

The police and Big Tech are profiling me!

6

Classification is everywhere, it's also very useful. Just get over it.

Online Advertising, Justice, Job Applications, Creditworthiness, Getting Insurance (Weapons of Math Destruction), Civic Life, /sideciteOneil2017 ; The Default Male, Invisible Women effects snow clearing schedules and drug discovery

```
cd myproject
docker run tensorflow
#profit!
```

tex.stackexchange.org for help.

The useful chaos of spaghetti code

7

Interacting Layers of Statistical Understanding Useful Chaos

These layers are totally transparent, but you can't understand them because they're complicated, yo

You can't understand ML layers, but they are useful nonetheless.

Talk about this book and Kasparov's tournament with computers and people. [14]

[14]: Mansharamani (2020), *Think for yourself: Restoring common sense in an age of experts and artificial intelligence*

Self-stabilizing concept drift

8

Many models need to constantly be retrained

Does progress slow down because we keep reusing the work of the past to generate our work?

My horse drives itself, thanks!

9

Can Models bring Incremental or Revolutionary Change?

Respond directly to Jon Krohn's TED talk about monkeys being dumber than us... what about construction equipment that's stronger than us, or racism/eugenics people that are dumber than us [15]

[15]: (2022), *Jon Krohn*

"The inhabitant of London could order by telephone, sipping his morning tea in bed, the various products of the whole earth – he could at the same time and by the same means adventure his wealth in the natural resources and new enterprise of any quarter of the world – he could secure forthwith, if he wished, cheap and comfortable means of transit to any country or climate without passport or other formality." - John Maynard Keynes [16]

[16]: Keynes et al. (2012), *The Collected Writings of John Maynard Keynes (Volume 5)*

Who is affected the most?

What should individuals do?

What should governments do?

What should businesses do?

Skynet: did you try unplugging it?

10

APPENDIX

A

????

Let's say we want to build an ensemble model to analyze poetry, put a haiku into craiyon's online shit, then we categorize the resulting photo.

[7]

[7]: Andreu et al. (2021), *Humans won't be able to control a superintelligent AI, according to a study*

Bibliography

Here are the references in citation order.

- [1] John von Neumann and Ray Kurzweil. *The Computer and the Brain (The Silliman Memorial Lectures Series)*. New Haven, CT, USA: Yale University Press, Aug. 2012 (cited on page ii).
- [2] Elaine Rich, Kevin Knight, and Shivashankar B. Nair. *Artificial Intelligence*. Tata McGraw-Hill, 2009 (cited on page 1).
- [3] MacAskill2022. 'The Case for Longtermism'. In: *The New York Times* (Aug. 5, 2022). (Visited on 08/05/2021) (cited on page 7).
- [4] Cade Metz. 'The Long Road to Driverless Trucks'. In: *N.Y. Times* (Sept. 2022) (cited on page 7).
- [5] Cade Metz. 'Stuck on the Streets of San Francisco in a Driverless Car'. In: *N.Y. Times* (Sept. 2022) (cited on page 7).
- [6] Caglar Aytakin. 'Neural Networks are Decision Trees'. In: (2022). doi: [10.48550/ARXIV.2210.05189](https://arxiv.org/abs/10.48550/ARXIV.2210.05189) (cited on page 7).
- [7] Abraham Andreu and Qayyah Moynihan. 'Humans won't be able to control a superintelligent AI, according to a study'. In: *Business Insider* (Sept. 24, 2021). (Visited on 09/24/2021) (cited on pages 8, 17).
- [8] Jeff Hawkins. *A thousand brains: A new theory of intelligence*. Basic Books, 2022 (cited on pages 8, 10).
- [9] David McCullough. *Truman*. Riverside, NJ, USA: Simon & Schuster, June 1992 (cited on page 10).
- [10] Brian Christian. *The Alignment Problem: Machine Learning and Human Values*. New York, NY, USA: W. W. Norton & Company, Oct. 2020 (cited on page 10).
- [11] Cathy O'Neil. *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*. New York, NY, USA: Crown, Sept. 2017 (cited on page 10).
- [12] Caroline Criado Perez. *Invisible Women: Data Bias in a World Designed for Men*. New York, NY, USA: Abrams Press, Mar. 2019 (cited on page 10).
- [13] Reid Blackman. *Ethical Machines: Your Concise Guide to Totally Unbiased, Transparent, and Respectful AI*. Harvard Business Review Press, July 2022 (cited on page 10).
- [14] Vikram Mansharamani. *Think for yourself: Restoring common sense in an age of experts and artificial intelligence*. Harvard Business Review Press, 2020 (cited on page 12).
- [15] Jon Krohn. [Online; accessed 18. Oct. 2022]. Oct. 2022. URL: <https://www.jonkrohn.com/posts/2022/10/7/tedx-talk-how-neuroscience-inspires-ai-breakthroughs-that-will-change-the-world> (cited on page 14).
- [16] John Maynard Keynes, Elizabeth Johnson, and Donald Moggridge. *The Collected Writings of John Maynard Keynes (Volume 5)*. Cambridge, England, UK: Cambridge University Press, Dec. 2012 (cited on page 14).

Notation

The next list describes several symbols that will be later used within the body of the document.

c Speed of light in a vacuum inertial frame

h Planck constant

Alphabetical Index

preface, iii