Getting Started with AI

Welcome to the *Practicum AI: Getting Started with AI* Course! This course is intended to be the first in a series that will teach you some tools and techniques to begin training and deploying models on your own. The course can also be taken on its own to familiarize yourself with the important concepts in the field of Artificial Intelligence.

Module 0: What is *Practicum AI*?

Let's start with the big picture. The *Practicum AI* program is designed to equip you with the tools and knowledge to execute *your own* AI project or mission.

To achieve this goal, the *Practicum AI* track features a lot of technical, hands-on learning experiences. We, the *Practicum AI* Team, think that a more important question than "What can we show you?" is, "What do *you* want to do? What problems or research questions interest *you,* and how can AI tools help?" Once you answer that, everything else falls into place. We believe the best learning happens when you have a clear goal in mind.

So, we encourage you to think of AI - not from a technical point of view - but as a new way of answering questions, delivering services, and solving problems.

Our target audience is beginners who have little-to-no technical background.

Again – this program is **not designed** for a highly technical audience. If you already have a strong technical background or multiple years of programming experience, we encourage you to explore other SCINet training opportunities. In this course we'll cover:

Module 1: What is AI?

A high-level overview of the concepts, history, and vocabulary of AI.

Module 2: Understanding AI Models

Hands-on experience with three AI models intended to prime understanding of how they work.

Module 3: The AI Application Development Cycle

A crash course in the steps to develop an AI application.

Module 4: A Very Brief Introduction to Ethics in AI

A look at industry-adopted principles that guide ethical AI development.

Module 1: What is AI?

Module 1 Objectives:

By the end of this module, you will be able to:

1. Define AI and explain what it is.
2. Distinguish among AI and related fields, such as machine learning and data science.
3. Summarize the major events in the history of AI development.
4. Describe the benefits of using a high-performance computing environment to develop AI applications.
5. Identify the challenges of developing AI applications in a high-performance computing environment.
6. Recognize that Python is the predominant programming language in AI application development.

Defining AI

By now, you've almost definitely heard someone ask, "What *is* AI?". Answering that question is very difficult for at least a couple of reasons:

1. To truly define "Artificial" Intelligence, we would have to have a good, thorough working definition of normal intelligence. We, uh, don't currently (but a lot of smart people are working on this right now!) [1]
2. The scope of what we consider AI has changed considerably over time. From the pioneering days of the 1950s and Alan Turing's work on thinking computers to the intense marketing hype of today's ChatGPT, our expectations of what a "thinking" computer looks like continue to evolve.

Because of this, definitions tend to be vague and/or recursive ("Intelligence demonstrated by a machine as opposed to a human or other animal" [2]). Russell & Norvig's description of AI from their 2003 textbook, *Artificial Intelligence: A Modern Approach,* is *"*the study of 'rational agents': any system that perceives its environment and takes actions that maximizes its ability to achieve its goals." Another way to look at this is if a computer is using information to make the kinds of inferences or decisions a human would make, that computer is probably using an application of AI. Another, *other,* way to look at it is just… fancy math. AI applications are typically algorithms that are just using a lot more data than a person can comfortably compute by hand (during this course and other Practicum courses you'll see what we mean by a lot of data).

AI, Machine Learning, Deep Learning, and Computer Science

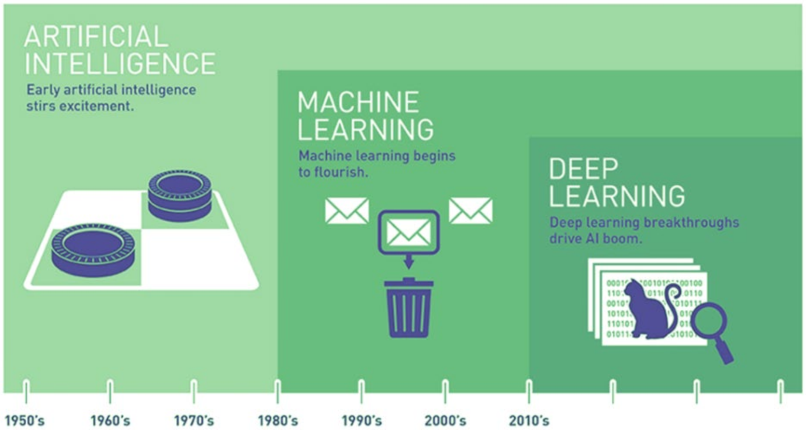


Figure 1 Relationship of AI, ML, and DL. Copeland, M. [3]

There is another barrier to defining AI: the term is often used interchangeably with other disciplines. Commonly, AI will be confused with its subdisciplines Machine Learning and Deep Learning. Here's a quick and dirty comparison of AI, Machine Learning, Deep Learning and their parent field, Computer Science:

Term	**Summary**	**Examples**
Artificial intelligence (AI)	The ability of a computer or machine to learn and perform tasks that are typically associated with human intelligence, such as reasoning, learning, and problem-solving.	Self-driving cars, virtual assistants, and spam filters
Machine learning (ML)	A type of AI that allows computers to learn without being explicitly programmed. ML algorithms are trained on data, and they can then use that data to make predictions or decisions.	Netflix's recommendation engine, spam filters, and fraud detection systems

Deep learning (DL)	A subset of ML that uses artificial neural networks to learn from data. Neural networks are inspired by the human brain, and they can learn to recognize patterns in data that would be difficult for humans to identify.	Image recognition, speech recognition, and natural language processing
Computer science (CS)	The study of computing. CS encompasses a wide range of topics, including algorithms, data structures, programming languages, and artificial intelligence.	Software engineering, web development, and cybersecurity

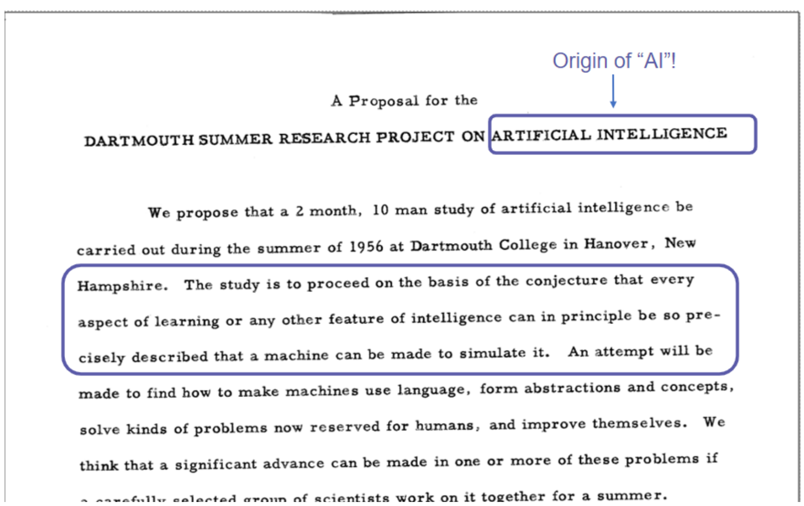
As you can see, AI is a broad term that encompasses a wide range of methods and techniques. Machine learning and deep learning are two specific types of AI that are used to create systems that can learn and make predictions from data. Computer science is the field of study that encompasses all these disciplines.

An Abridged History of AI

The history of AI spans nearly 80 years and involves countless technological innovations brought about by thousands of scientists and engineers. This entire module is only supposed to last about an hour so we're going to stick to the highlights.

In the beginning (1955), the first conference on "artificial intelligence" was held at Dartmouth College. It was attended by many of the leading computer science researchers of the day, including John McCarthy, Marvin Minsky, Nathaniel Rochester, and Claude Shannon.

In early 1955, John McCarthy accepted a position as an Assistant Professor of Mathematics at Dartmouth College. Soon after, he



A Proposal for the

DARTMOUTH SUMMER RESEARCH PROJECT ON ARTIFICIAL INTELLIGENCE

We propose that a 2 month, 10 man study of artificial intelligence be carried out during the summer of 1956 at Dartmouth College in Hanover, New Hampshire. The study is to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it. An attempt will be made to find how to make machines use language, form abstractions and concepts, solve kinds of problems now reserved for humans, and improve themselves. We think that a significant advance can be made in one or more of these problems if a carefully selected group of scientists work on it together for a summer.

Figure 2: Proposal that coined the term Artificial Intelligence.

organized a group to discuss the future of thinking machines. In September, he, Marvin Minsky, Nathaniel Rochester, and Claude Shannon submitted a proposal (Figure 2, shown above) to the Rockefeller Foundation, requesting funds to host the *Dartmouth Summer Research Project on Artificial Intelligence*. Fortunately, the foundation funded the proposal. Today, this conference is credited with introducing the term 'artificial intelligence' to the world.

Any historical survey of AI would be incomplete without Alan Turing. Turing is a towering figure in AI. He was a mathematical genius who thought a lot about the question of intelligence. Turing first proposed a test of intelligence in 1950, in an article entitled, Computing Machinery and Intelligence. This has become a foundational document in the field.

The Turing Test (designed, of course, by Turing) of intelligence has been hugely influential since it was first described in 1950.

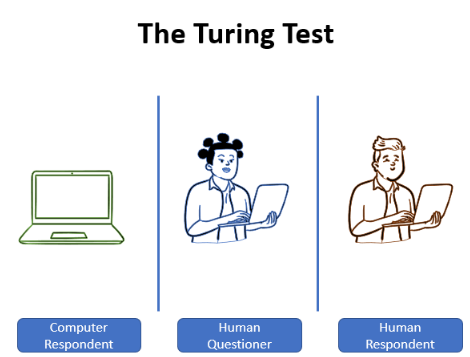


Figure 3: An illustration of the Turing Test

1. Turing got the idea from a Victorian parlor game where a man and a woman sat in a separate room. The other players then passed a series of written questions to them and would try to guess the gender of the respondent by the answers they gave.
2. The Turing Test is a computer version of that game. In the image here (Figure 3), we see a Human questioner in the middle, sending questions to both a computer and another human. The interaction is purely in the form of text and answers: the human questioner types a question, and a response is displayed. Now the task of that person is to determine whether the thing being interrogated is a person or a computer program.
3. Now, suppose that the thing being interrogated is indeed a computer program. But after some reasonable amount of time, the questioners cannot tell whether they are interacting with a program or a person. Then surely, Turing argued, you should accept that the program has some sort of human-level intelligence because the system is doing something that makes it **indistinguishable** from the real thing. The key word here is indistinguishable…

Then in 1958, Frank Rosenblatt created the first perceptron. Perceptrons, also known as McCullogh-Pitts neurons after the researchers that theorized them, are algorithms for supervised learning of binary classifiers, making them the first implementation of what we'd come to call 'Machine Learning'. We'll go over perceptrons, supervised learning, and binary classifiers later in Module 2 of this course.

In the 1960s, in response to Turing's article, Joseph Weizenbaum – a computer scientist at MIT – created one of the first conversational AI programs. ELIZA mimics a psychiatrist or psychologist having a conversation with a patient. Strangely, Weizenbaum discovered that many people preferred to talk with ELIZA rather than a real human being.

ELIZA's legacy lives on to this day. In 1990, Hugh Loebner created the Loebner Prize. Each year, the Loebner Prize invited (the last contest appears to have been in 2019) the submission of computer programs to engage in the Turing test, attempting to convince a panel of judges that they are in fact people. Steve Worswick won the prize five years in a row with his AI bot Kuki AI.

Deep Blue was a chess-playing computer developed by IBM. It was first announced in 1996, and it was designed to defeat world chess champion Garry Kasparov. Deep Blue used a combination of brute force and expert knowledge to play chess. It could search through millions of possible moves in a fraction of a second, and it could also draw on the expertise of human chess grandmasters. In 1997, Deep Blue defeated Kasparov in a six-game match. The match was closely contested, but Deep Blue ultimately won by a score of 3.5 to 2.5. Deep Blue's victory was a breakthrough for AI, and it showed that computers could now compete with humans at the highest levels of chess.

Watson is a question-answering computer also developed by IBM. It was designed to compete on the *Jeopardy!* quiz show. Watson used a combination of natural language processing and machine learning to answer questions. It could 'understand' the meaning of questions, and it could search through vast amounts of data to find the correct answers. In 2011, Watson won the *Jeopardy!* quiz show against two of the show's most experienced champions. Watson's victory showed both the power and potential of AI and natural language processing.

In 2012, a Google Brain computer cluster used deep learning techniques to teach itself to recognize cats. The team that created it had expected the system to need more training data, but it was able to learn to recognize cats with just a few million images. They believe that this is because the system was able to generalize from the data. In other words, it was able to learn the features that are common to all cats, even if the images it was trained on were not all the same cat. While it did make some interesting errors, such as misidentifying a toaster and a shoe as a cat, it was accurate over 90% of the time.

AlphaGo is a Go-playing computer developed by Google. It was designed to defeat world Go champion Lee Sedol. Go is a complex board game that is one of the most challenging games for computers to play given the ridiculous number of possible moves in a game (estimates put the number of possible moves around 10^{360}). AlphaGo used a combination of deep learning and other AI techniques to play Go. It could learn from its mistakes, and it could explore millions of possible moves in a fraction of a second. AlphaGo defeated Lee Sedol in a five-game match. The match was closely contested, but AlphaGo won by 4 to 1. AlphaGo showed that a program could solve problems that we had believed were too complicated for computers.

Finally, we have the star of the day, ChatGPT. ChatGPT (which stands for Chat Generative Pre-trained Transformer) was released as version 3.5 in November 2022. It is a type of language model that can create (or generate) conversational language as a response to language inputs. The language model used for ChatGPT was not explicitly taught the rules of the English language, but rather picked up grammar simply by looking at lots of examples of complete sentences (about 1 petabytes worth).

Circling back to the study of intelligence (artificial or otherwise), we're going to give you time to chat with a bot. While there are many AI chatbots, most require an account and possibly a paid subscription. One exception, hosted by a popular repository of AI models and datasets Hugging Face, is HuggingChat.

Go to the [HuggingChat website]() and talk with the bot. (Please note that current USDA guidance prohibits using externally hosted generative AI tools for work purposes, so please limit your interactions to non-work topics.) Well, what do you think? Is HuggingChat exhibiting intelligent behavior? Is this intelligence?

High-Performance Computing and You

You have probably noticed at this point that the ideas behind artificial intelligence have been around for a long time. So why has it taken us so long to implement them? In large part, the answer is computing power. AI can now do things that were once thought impossible, like beating humans at chess and Go, and driving cars without any human input. But what many people don't realize is that AI wouldn't be nearly as smart as it is without high-performance computing (HPC) environments.

HPC systems are typically large supercomputers that can crunch massive amounts of data quickly. This is essential for AI, because AI algorithms need to be trained on huge datasets to learn how to make predictions. Without HPC, AI would take years to train, and it would never be able to learn as much as it can with HPC.

What is a "supercomputer"? A supercomputer is multiple computers networked (or clustered) together. The number of computers in a cluster can be anywhere from two to thousands. Special software and networking equipment allow the machines to work as one large computer.

To give some perspective, we need a way to measure computer "speed". This is usually measured in FLOPs (FLoating point OPerations). Without getting into the weeds, this is a measurement of how fast a computer can do mathematical calculations. A modern desktop computer (with a graphics card) can process $6.1*10^{12}$ FLOPs, though this varies widely (and gets higher all the time!).

- 1997 - IBM's Deep Blue: $11.38*10^{9}$ FLOPs (11 GigaFLOPS)
- 2016 - Google's AlphaGo: $180*10^{12}$ FLOPs (180 TeraFLOPS)
- 2021 – UF's HiPerGator: $13.75*10^{15}$ FLOPs (13 PetaFLOPS)
- 2022 – Computer that trained ChatGPT 3.5: $100.7*10^{15}$ FLOPs (100 PetaFLOPS)
- 2023 – World's fastest computer in May: $2*10^{18}$ FLOPs (2 ExaFLOPS)

As you can see, supercomputers are getting pretty FLOPpy—we are now in what is known as the era of ExaScale computing). The main reason for this is the advent of GPUs (graphic processing units). CPUs are designed for general-purpose computing tasks. They excel at tasks that require high single-threaded performance and are designed to handle a wide range of applications, from running operating systems to running complex software. GPUs, on the other hand, have a massively parallel architecture with hundreds or even thousands of smaller cores. These cores are optimized for simultaneous execution of multiple tasks, making GPUs ideal for parallel processing. Originally developed for rendering graphics, GPUs have evolved into highly efficient processors for handling repetitive, data-parallel computations, which are common in graphics rendering, scientific simulations, and machine learning.

Here are some case studies showing the benefits of HPC:

https://builtin.com/hardware/high-performance-computing-applications

It's Not All Sunshine and Rainbows

While supercomputers offer significant computational power and performance advantages, they also come with a few downsides. Here are some potential drawbacks associated with using supercomputers:

1. Cost: Supercomputers are expensive to develop, build, and maintain. They require substantial investments in terms of hardware, software, cooling systems, power infrastructure, and skilled personnel. The initial procurement costs, as well as ongoing operational expenses, can be prohibitive for many organizations.
2. Power Consumption: Supercomputers consume a tremendous amount of electricity. This high power consumption translates into substantial energy costs and environmental impact. Cooling the systems can also be challenging, as supercomputers generate a significant amount of heat that must be efficiently dissipated.
3. Complexity: Supercomputers are complex machines that require specialized expertise to operate and program effectively. Developing software and algorithms that can fully utilize the massive parallel processing capabilities of supercomputers is a challenging task. The complexity can result in longer development cycles and increased debugging efforts.
4. Limited Accessibility: Due to their high cost and complexity, access to supercomputers is limited to a relatively small number of organizations or research institutions with the necessary resources and expertise. This limitation can hinder broader participation and collaboration in research or other computationally demanding endeavors.

Fortunately, through the USDA-ARS SCINet program, ARS has invested in shared supercomputing infrastructure and technical expertise that are available to all ARS employees and their collaborators. SCINet currently operates two HPC clusters, provides IT support through the Virtual Research Support Core, and provides application support and training through the SCINet Office.

The Python Supremacy

The last piece of the AI puzzle is the Python programming language. Python has become the predominant programming language in the field of artificial intelligence (AI) for several reasons:

1. Simplicity and Readability: Python is known for its simplicity and readability. Its clean syntax and straightforward structure make it easier for developers to write and understand code. This characteristic makes Python an ideal language for prototyping and experimenting with AI algorithms and models.

2. Large and Active Community: Python has a vast and active community of developers. This community has contributed to the development of numerous libraries, frameworks, and tools specifically tailored for AI and machine learning (ML) tasks. Popular libraries like TensorFlow, PyTorch, and scikit-learn have extensive Python support, making it easier to implement AI algorithms and models.

3. Integration and Interoperability: Python offers excellent integration capabilities, allowing developers to combine AI with other technologies and systems seamlessly. Python can easily interact with other popular programming languages like C/C++, Java, and R, making it convenient for incorporating AI functionality into existing software systems.

4. Data Analysis and Data Science: Python is widely used in the field of data analysis and data science. It offers powerful tools and libraries for data manipulation, exploration, and preprocessing. This aligns well with AI, as AI development relies on large datasets and requires those same tools for working with data. Python's popularity in data science has contributed to its dominance in the AI landscape.

5. Education and Learning Resources: Python is often the programming language of choice for introductory AI and ML courses. Its simplicity and readability make it accessible to beginners, allowing them to grasp fundamental AI concepts quickly. The abundance of learning resources, tutorials, and documentation available for Python facilitates the learning process for aspiring AI developers.

6. Industry Adoption: Python's popularity in AI is also driven by its widespread adoption in the industry. That's right, Python is popular partly because… its popular. Many organizations and companies have embraced Python as their primary language for AI development. This adoption has led to a virtuous cycle, with more resources, tools, and talent being invested in Python-based AI research and development.

While Python's dominance in AI is significant, it's worth noting that other programming languages like R, Java, and C++ are also used in AI applications. That said, Python's combination of simplicity, extensive libraries, and strong community support has made it *the* go-to language for AI development.

References

[1] Sternberg, Robert J. "human intelligence". Encyclopedia Britannica, 11 Apr. 2022, https://www.britannica.com/science/human-intelligence-psychology. Accessed 19 May 2023.

[2] Hauser, Larry. "Artificial Intelligence." *Internet Encyclopedia of Philosophy*, iep.utm.edu/artificial-intelligence. Accessed 19 May 2023.

[3] https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/

[4] https://www.techtarget.com/searchenterpriseai/definition/Turing-test

[5] Tate, Karl. "History of A.I.: Artificial Intelligence (Infographic)." *LiveScience*, 25 Aug. 2014, www.livescience.com/47544-history-of-a-i-artificial-intelligence-infographic.html.