WHAT IS MACHINE LEARNING?

In 1959, IBM published a paper in the *IBM Journal of Research and Development* with an obscure and curious title for that time. Authored by IBM’s Arthur Samuel, the paper investigated the application of machine learning in the game of checkers “to verify the fact that a computer can be programmed so that it will learn to play a better game of checkers than can be played by the person who wrote the program.”[5]

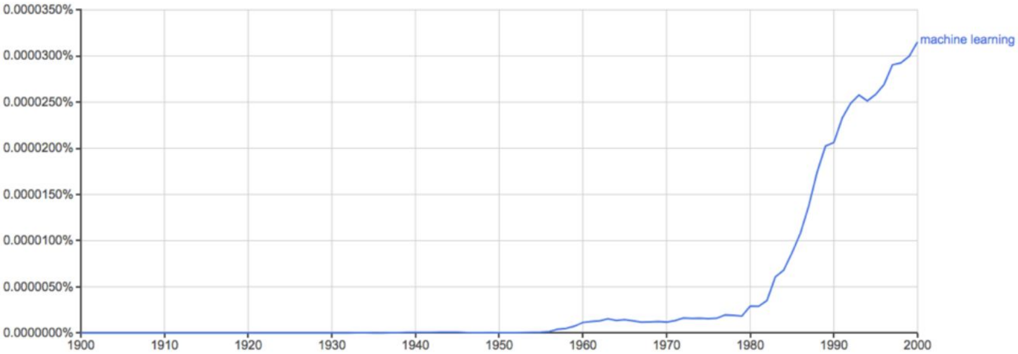

Figure 1: Historical mentions of “machine learning” in published books. ***Source: Google Ngram Viewer, 2017***

Although it wasn’t the first published work to use the term “machine learning” per se, Arthur Samuel is regarded as the first person to coin and define machine learning as the concept and specialized field we know today. Samuel’s landmark journal submission, *Some Studies in Machine Learning Using the Game of Checkers,* introduces machine learning as a subfield of computer science that gives computers the ability to learn without being explicitly programmed.[6]

While not directly treated in Arthur Samuel’s initial definition, a key characteristic of machine learning is the concept of *self-learning*. This refers to the application of statistical modeling to detect patterns and improve performance based on data and empirical information; all without direct programming commands. This is what Arthur Samuel described as the ability to learn without being explicitly programmed. Samuel didn’t infer that machines may formulate decisions with no upfront programming. On the contrary, machine learning is heavily dependent on code input. Instead, he observed machines can perform a set task using *input data* rather than relying on a direct *input command*.

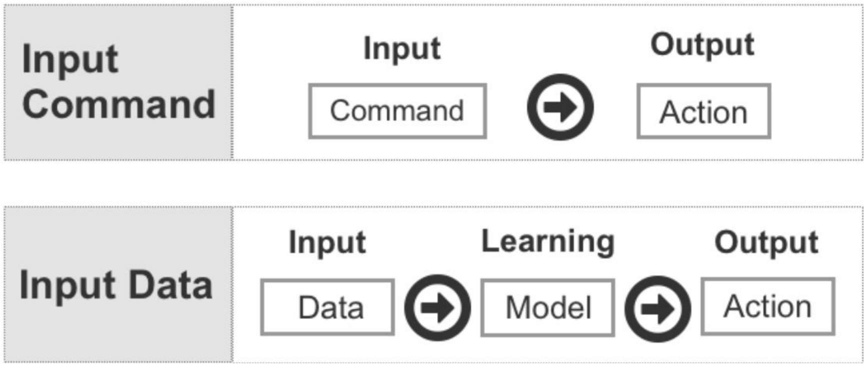

Figure 2: Comparison of Input Command vs Input Data

An example of an input command is entering “2+2” in a programming language such as Python and clicking “Run” or hitting “Enter” to view the output.

```
>>> 2+2
4
>>>
```

This represents a direct command with a pre-programmed answer, which is typical of most computer applications. Unlike traditional computer programming, though, where outputs or decisions are pre-defined by the programmer, machine learning uses data as input to build a decision model. Decisions are generated by deciphering relationships and patterns in the data using probabilistic reasoning, trial and error, and other computationally-intensive techniques. This means that the output of the decision model is determined by the contents of the input data rather than any pre-set rules defined by a human programmer. The human programmer is still responsible for feeding the data into the model, selecting an appropriate algorithm and tweaking its settings (called *hyperparameters*) in order to reduce prediction error, but the machine and developer operate a layer apart in contrast to traditional programming.

To draw an example, let’s suppose that after analyzing YouTube viewing habits the decision model identifies a significant relationship among data scientists watching cat videos. A separate model, meanwhile, identifies patterns among the physical traits of baseball players and their likelihood of winning the season's Most Valuable Player (MVP) award.

In the first scenario, the machine analyzed which videos data scientists enjoy watching on YouTube based on user engagement; measured in likes, subscribes, and repeat viewing. In the second scenario, the machine assessed the physical attributes of previous baseball MVPs among other features such as age and education. However, at no stage was the decision model told or programmed to produce those two outcomes. By decoding complex patterns in the input data, the model uses machine learning to find connections without human help. This also means that a related dataset gathered from another period of time, with fewer or greater data points, might lead the model to a slightly different output.

Another distinct feature of machine learning is the ability to improve predictions based on experience. Mimicking the way humans base decisions on experience and the success or failure of past attempts, machine learning utilizes exposure to data to improve decision outcomes. The socializing of data points provides experience and enables the model to familiarize itself with patterns in the data. Conversely, insufficient input data restricts the model's ability to deconstruct underlying patterns in the data and limits its capacity to respond to potential variance and random phenomena found in live data. Exposure to input data thereby helps to deepen the model's understanding of patterns, including the significance of changes in the data, and to construct an effective self-learning model.

A common example of a self-learning model is a system for detecting spam email messages. Following an initial serving of input data, the model learns to flag emails with suspicious subject lines and body text containing keywords that correlate highly with spam messages flagged by users in the past. Indications of spam email may comprise words like dear friend, free, invoice, PayPal, Viagra, casino, payment, bankruptcy, and winner. However, as the machine is fed more data, it might also find exceptions and incorrect assumptions that render the model susceptible to bad predictions. If there is limited data to reference its decision, the following email subject, for example, might be wrongly classified as spam: "**PayPal** has received your **payment** for **Casino** Royale purchased on eBay."

As this is a genuine email sent from a PayPal auto-responder, the spam detection system is lured into producing a false-positive based on the initial input data. Traditional programming is highly susceptible to this problem because the model is rigidly defined according to pre-set rules. Machine learning, on the other hand, incorporates exposure to data to refine its model, adjust its assumptions, and respond appropriately to unique data points such as the scenario described.

While data is used to source the self-learning process, more data doesn’t automatically equate to better decisions; the input data must be relevant to the scope of the model. In *Data and Goliath: The Hidden Battles to Collect Your Data and Control Your World*, Bruce Schneir writes that, “When looking for the needle, the last thing you want to do is pile lots more hay on it.”[7] This means that adding irrelevant data can be counter-productive to achieving a desired result. In addition, the amount of input data should be compatible with the processing resources and time that is available.

Training & Test Data

In machine learning, input data is typically split into *training data* and *test data*. The first split of data is the *training data*, which is the initial reserve of data used to develop your model. In the spam email detection example, false-positives similar to the PayPal auto-response message might be detected from the training data. Modifications must then be made to the model, e.g., email notifications issued from the sending address “payments@paypal.com” should be excluded from spam filtering. Applying machine learning, the model can be trained to automatically detect these errors (by analyzing historical examples of spam messages and deciphering their patterns) without direct human interference.

After you have developed a model based on patterns extracted from the training data and you are satisfied with the accuracy of its prediction, you can test the model on the remaining data, known as the *test data*. If you are also satisfied with the model’s performance using the test data, the model is ready to filter incoming emails in a live setting and generate decisions on how to categorize those messages. We will discuss training and test data further in Chapter 6.

The Anatomy of Machine Learning

The final section of this chapter explains how machine learning fits into the broader landscape of data science and computer science. This includes understanding how machine learning connects with parent fields and sister disciplines. This is important, as you will encounter related terms in machine learning literature and courses. Relevant disciplines can also be difficult to tell apart, especially machine learning and data mining.

Let’s start with a high-level introduction. Machine learning, data mining, artificial intelligence, and computer programming fall under the umbrella of computer science, which encompasses everything related to the design and use of computers. Within the all-encompassing space of computer science is the next broad field of data science. Narrower than computer science, data science comprises methods and systems to extract knowledge and insights from data with the aid of computers.

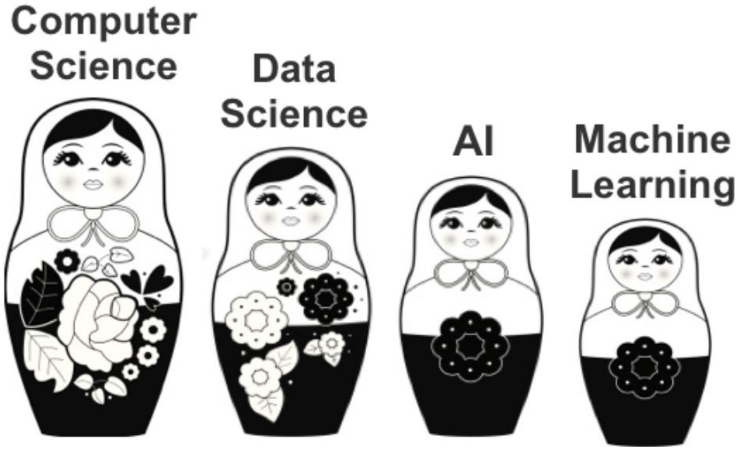

Figure 3: The lineage of machine learning represented by a row of Russian matryoshka dolls

Emerging from computer science and data science as the third matryoshka doll from the left in Figure 3 is artificial intelligence. Artificial intelligence, or AI, encompasses the ability of machines to perform intelligent and cognitive tasks. Comparable to how the Industrial Revolution gave birth to an era of machines simulating physical tasks, AI is driving the development of machines capable of simulating cognitive abilities.

While still broad but dramatically more honed than computer science and data science, AI spans numerous subfields that are popular and newsworthy today. These subfields include search and planning, reasoning and knowledge representation, perception, natural language processing (NLP), and of course, machine learning.

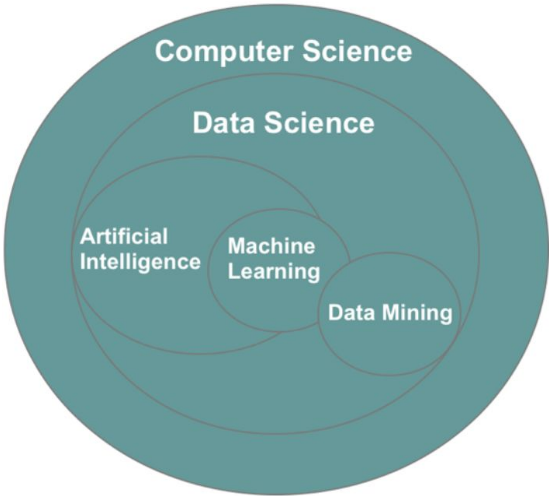

Figure 4: Visual representation of the relationship between data-related fields

For students interested in AI, machine learning provides an excellent starting point because it provides a narrower and more practical lens of study (in comparison to AI). Algorithms applied in machine learning can also be used in other disciplines, including perception and natural language processing. In addition, a Master's degree is adequate to develop a certain level of expertise in machine learning, but you may need a PhD to make genuine progress in artificial intelligence.

As mentioned, machine learning overlaps with data mining—a sister discipline that is based on discovering and unearthing patterns in large datasets. Both techniques rely on inferential methods, i.e. predicting outcomes based on other outcomes and probabilistic reasoning, and draw from a similar assortment of algorithms including principal component analysis, regression analysis, decision trees, and clustering techniques. To add further confusion, the two techniques are commonly mistaken and misreported or even explicitly misused. The textbook *Data mining: Practical machine learning tools and techniques with Java* is said to have originally been titled *Practical machine learning* but for marketing reasons "data mining" was later appended to the title.[8]

Lastly, because of their interdisciplinary nature, experts from a diverse spectrum of disciplines tend to define data mining and machine learning differently. This has led to confusion, in addition to a genuine overlap between the two disciplines. But whereas machine learning emphasizes the incremental process of self-learning and automatically detecting patterns