Inventors have long dreamed of creating machines that think. This desire dates back to ancient age across the world and we find this in all mythologies.

When programmable computers were first conceived, people wondered whether such machines might become intelligent, over a hundred years before one was built. Today, artificial intelligence is a thriving field with many practical applications and active research topics. We look for intelligent software to automate routine tasks, understand speech or images, make diagnoses in medicine and support basic scientific research.

In the early days of artificial intelligence, the field rapidly tackled and solved problems that are intellectually difficult for human beings but relatively straightforward and that can be described by a list of formal, mathematical rules. The true challenge to artificial intelligence proved to be solving the tasks that are easy for people to perform but hard for people to describe formally. Problems that we solve intuitively, that feel automatic, like recognizing spoken words or faces in images.

This deep learning course is about a solution to these more intuitive problems.

This solution is to allow computers to learn from experience and understand the world in terms of a hierarchy of concepts, with each concept defined in terms of its relation to simpler concepts.

By gathering knowledge from experience, this approach avoids the need for human operators to formally specify all of the knowledge that the computer needs. The hierarchy of concepts allows the computer to learn complicated concepts by building them out of simpler ones. If we draw a graph showing how these concepts are built on top of each other, the graph is deep, with many layers. For this reason, we call this approach to AI deep learning.

The introduction of machine learning allowed computers to tackle problems involving knowledge of the real world and make decisions that appear subjective.

A simple machine learning algorithm called logistic regression can determine whether exposed to risk of being a diabetic patient.
A simple machine learning algorithm called naive Bayes can filter spam e-mail.

The performance of these simple machine learning algorithms depends heavily on the representation of the data they are given.

For example, when logistic regression is used to identify diabetic, the AI system does not examine the patient directly. Instead, the doctor tells the system several pieces of relevant information, such as lipid profile, BP. Each piece of information included in the representation of the patient is known as a feature.

Logistic regression learns how each of these features of the patient correlates with various outcomes. However, it cannot influence the way that the features are defined in any way. If logistic regression was given a scan report, rather than the doctor's formalized report, it would not be able to make useful predictions. Individual pixels in scan have negligible correlation with any complications that might cause diabetics in future.

This dependence on representations is a general phenomenon that appears throughout computer science and even daily life. It is not surprising that the choice of representation has an enormous effect on the performance of machine learning algorithms. For a simple visual example see the graph.

Many artificial intelligence tasks can be solved by designing the right set of features to extract for that task, then providing these features to a simple machine learning algorithm. For example, a useful feature for speaker identification from sound is an estimate of the size of speaker's vocal tract. It therefore gives a strong clue as to whether the speaker is a man, woman, or child.

However, for many tasks, it is difficult to know what features should be extracted. For example, suppose that we would like to write a program to detect cars in photographs. We know that car has wheels, so we might like to use the presence of a wheel as a feature. Unfortunately, it is difficult to describe exactly what a wheel looks like in terms of pixel values. A wheel has a simple geometric shape but its image may be complicated by shadows falling on the wheel, the sun glaring off the metal parts of the wheel, the fender of the car or an object in the foreground obscuring part of the wheel, and so on.

One solution to this problem is to use machine learning to discover not only the mapping from representation to output but also the representation itself.

This is called representation learning. Often learned representations produce better result in much better performance than can be obtained with hand-designed representations. They also allow AI systems to rapidly adapt to new tasks, with minimal human intervention. A representation learning algorithm can discover a good set of features for a simple task in minutes, or a complex task in hours to months. Manually designing features for a complex task requires a great deal of human time and effort; it can take decades for an entire community of researchers.

Deep learning solves the complex problem in representation learning by introducing representations that are expressed in terms of other, simpler representations. Deep learning allows the computer to build complex concepts out of simpler concepts.

The most typical example of a deep learning model is the feedforward deep network or multilayer perceptron (MLP). A multilayer perceptron is just a mathematical function mapping some set of input values to output values. The function is formed by composing many simpler functions. We can think of each application of a different mathematical function as providing a new representation of the input.

The idea of learning the right representation for the data provides one perspective on deep learning. Another perspective on deep learning is that depth allows the computer to learn a multi-step computer program.
Each layer of the representation can be thought of as the state of the computer's memory after executing another set of instructions

in parallel. Networks with greater depth can execute more instructions in sequence.

Sequential instructions offer great power because later instructions can refer back to the results of earlier instructions. According to this view of deep learning, not all of the information in a layer's activations necessarily encodes factors of variation that explain the input.

The representation also stores state information that helps to execute a program that can make sense of the input. This state information could be analogous to a counter or pointer in a traditional computer program. It has nothing to do with the content of the input specifically, but it helps the model to organize its processing.

If we look at the history of Deep learning, we observe the following key trends:

• Deep learning has had a long and rich history, but has gone by many names reflecting different philosophical viewpoints, and has waxed in popularity.

- Deep learning has become more useful as the amount of available training data has increased.

- Deep learning models have grown in size over time as computer infrastructure (both hardware and software) for deep learning has improved.

- Deep learning has solved increasingly complicated applications with increasing accuracy over time.

The exponential growth in the amount of data available to us has helped deep learning models evolve to handle more complex issues. We are talking about the next frontier of AI where the capability of machines to learn and apply will start approaching the levels of human intelligence. The future is exciting – join us for the ride.