

# AI For Everyone

Om Prabhu

19D170018

Undergraduate, Department of Energy Science and Engineering  
Indian Institute of Technology Bombay

Last updated July 20, 2020

---

**NOTE:** This document is a brief compilation of my notes taken during the ‘AI For Everyone’ course by [deeplearning.ai](https://www.coursera.org/learn/ai-for-everyone). You are free to read and modify it for personal use. You may check out the course here: <https://www.coursera.org/learn/ai-for-everyone>.

---

## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	About myself . . . . .	3
1.2	About this course . . . . .	3
<b>2</b>	<b>What is AI?</b>	<b>4</b>
2.1	Machine Learning . . . . .	4
2.2	Data . . . . .	5
2.2.1	Collection of data . . . . .	5
2.2.2	Misconceptions about data . . . . .	6
2.3	AI terminology . . . . .	6
2.3.1	Machine learning vs. data science . . . . .	6
2.3.2	Deep learning . . . . .	7
2.3.3	The larger picture . . . . .	7
2.4	AI companies . . . . .	8
2.5	Limitations of AI . . . . .	8
2.6	Understanding deep learning . . . . .	9
<b>3</b>	<b>Building AI Projects</b>	<b>11</b>
3.1	Workflow of a machine learning project . . . . .	11
3.2	Workflow of a data science project . . . . .	11
3.3	Impact of data on job functions . . . . .	12
3.4	Choosing an AI project . . . . .	12
3.5	Working with an AI team . . . . .	13
3.6	Technical tools for AI teams . . . . .	14
<b>4</b>	<b>Building AI in a Company</b>	<b>15</b>
4.1	Smart speakers: a case study . . . . .	15
4.2	Self-driving cars: a case study . . . . .	15
4.3	Major roles in an AI team . . . . .	16
4.4	The AI transformation playbook . . . . .	17
4.5	First steps and pitfalls . . . . .	18
4.6	Major AI application areas . . . . .	18
4.6.1	Computer vision . . . . .	18
4.6.2	Natural language processing . . . . .	18
4.6.3	Speech . . . . .	19
4.6.4	Robotics . . . . .	19
4.6.5	General machine learning . . . . .	19
4.7	Major AI techniques . . . . .	19
4.7.1	Unsupervised learning . . . . .	19

4.7.2	Transfer learning . . . . .	19
4.7.3	Reinforcement learning . . . . .	20
4.7.4	GANs (generative adversarial networks) . . . . .	20
4.7.5	Knowledge graph . . . . .	20

<b>5</b>	<b>AI and Society</b>	<b>21</b>
----------	-----------------------	-----------

---

# 1 Introduction

## 1.1 About myself

Hello. I am Om Prabhu, currently an undergrad at the Department of Energy Science and Engineering, IIT Bombay. If you have gone through my website (<https://omprabhu31.github.io/>) earlier, which is probably where you found this document too, you will know that I am quite a bit into programming and tinkering with code to try and do cool stuff. Additionally, I love playing video games, listening to music and engaging in a little bit of creative writing as and when I get time. With this brief self-introduction, let us get into why I decided to pursue this course.

## 1.2 About this course

As you probably know, AI (artificial intelligence) is rapidly changing the way we work and live. It is difficult to name industries which are not likely to be impacted by AI in the near future (I initially thought of the textile industry as an example, but a simple Google search proved exactly how wrong I was). AI is generating huge amounts of industrial revenue per year and is likely to create 13 trillion US dollars per year by the time we reach 2030 (source: McKinsey Global Institute).

Hence, it is important to gain at least a general overview of the what makes AI such a powerful tool. Right off the bat, one of the major reasons why AI has taken off recently is due to the rise of neural networks and deep learning. But this is not all. One needs to learn what types of data are valuable to AI and how the type and amount of data influences the performance of a neural network. Further, it is also important to know how AI can be used to build personal as well as company projects. Lastly, it is also important to know how AI will affect society and jobs so that one is better able to understand AI technology and navigate this rise of AI.

With all this said, let us try to understand what AI really is and accomplish the above objectives.

---

## 2 What is AI?

AI is a very happening industry today and there is a lot of excitement among people as to how the rise of AI will map out. While this has boosted development of AI technologies even further, it has also lead to irrational fears among society. One of the major reasons for this is because not many people realize that AI can actually be put into 2 separate categories:

- ANI (artificial narrow intelligence): can do one thing (eg: smart speaker, self driving car, web search algorithms); incredibly valuable in specific industries due to narrow application
- AGI (artificial global intelligence): can do anything a human can do (perhaps even more things)

While the world has seen tremendous progress with ANIs, the same cannot be said for AGIs. This lack of distinction between ANIs and AGIs is what has led to fears of super-intelligent robots taking over the world.

In this section, we will be mainly looking at what ANIs can do and how to apply them to real-world problems.

### 2.1 Machine Learning

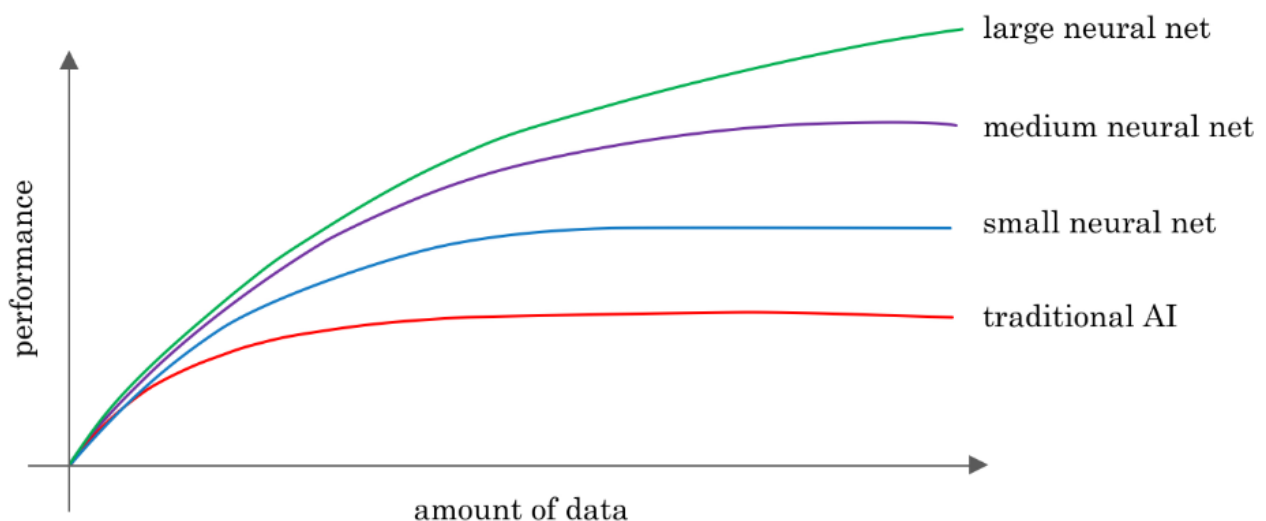
The rise of AI has been driven by one major tool known as Machine Learning. While the term might give a characteristic of omniscience to machines, this is far from true. In fact, the most used form of ML is what is known as Supervised Learning (or  $A \rightarrow B$  mapping):

- learning a function that maps input to output based on a database of example I/O pairs
- learning algorithm analyses example training data to generate a function to map new examples
- eventually the algorithm can *learn* to predict the target output in previously unseen situations

One major application of this technology is in the online advertising industry. The input is in the form of advertisement details & some user info based on which the AI algorithm tries to figure out if the user will click on the ad or not. This is how users are shown only a certain set of advertisements online.

Another application of supervised learning lies in self driving cars. The input is a set of images & some radar info from sensors on the car. The AI uses this data to output the position of nearby cars and/or obstacles so that the self-driving car can avoid them.

Surely the concept of merely taking the input to construct an output seems limiting in the general sense of the scope of AI, however it can evidently be very valuable once a suitable application scenario is found. Now while the idea of supervised learning has been around for decades, it has taken off only in the recent years. This is mainly because of the limitation of technology to train large sets of neural networks to process huge amounts of data while also improving AI performance. This can be illustrated through a graph as follows:



- for traditional AI systems, the data vs performance graph maxes out pretty early
- on training increasingly complex neural networks with higher amounts of data, performance keeps on getting better for much longer

Hence to achieve the highest performance levels, we need two things. Firstly, it helps to have a lot of data - this is where terms like ‘big data’ come in. Additionally, we need the ability to train large sets of neural networks, which is made possible by specialized processors like advanced GPUs.

## 2.2 Data

Data is one of the 2 main things required to improve performance of AI systems. But simply having lots of data is not always helpful - we need to have the right type of data in the right format (structured or unstructured). Let’s take a look at an example of a ‘dataset’ (or a table of data):

SIZE OF HOUSE (SQFT)	...	PRICE OF HOUSE (1000\$)
548		119
679		167
834		233
1209		342
1367		399

In practice, we will need a lot more than just 5 data entries to build an AI system, but let’s work with this for now. The above dataset could work for an AI which checks whether houses are priced appropriately or not. In this case, the input would be the size of the house and the output would be the price. Going further, we might try to improve our AI by adding more input data fields such as the number of bedrooms, location, etc.

Another application of the same dataset would be figuring out the most appropriate house for a consumer on a fixed budget. In this case, our input and output fields will be exactly the opposite compared to the first application. What this means in essence is that, given a dataset, it is up to us to decide what is the input and what is the output, and how to choose these definitions to bring out the maximum value to our product.

### 2.2.1 Collection of data

So far, we have established that data is an important tool for AI systems and that there is a certain flexibility regarding the choice of input and output. But how do we get data? To discuss this, let us now switch to the more traditional example in machine learning of an algorithm designed to recognize images of cats:

- manual labelling: collect a set of pictures and manually label them as ‘cat’ or ‘not cat’
  - tried and true way of obtaining a highly reliable dataset having both input and output fields
  - difficult, given the huge amount of data required (usually on the scale of several thousands of entries)
- observing behaviours
  - user behaviour: e-commerce websites keeping a tab on prices offered to users and whether they bought the product or not, something like this:

USER ID	TIME	PRICE (\$)	PURCHASED
0156	Feb 22, 09:19:19	19.07	No
1548	Apr 01, 23:34:56	23.01	Yes
4898	May 23, 11:59:02	18.72	Yes
8896	Jul 10, 17:42:37	16.55	No

- machine behaviour: fault prediction in machines based on operating conditions, something like this:

MACHINE ID	TEMPERATURE (K)	PRESSURE (ATM)	FAULT
55132	332	10.96	Yes
29475	378	8.22	Yes
00826	489	5.78	No
19475	653	2.99	No

- download from internet/partnerships: pre-compiled datasets that can be readily downloaded from the web (after obtaining required licenses if any), or obtained from partners (eg: a company obtaining fault analysis datasets from machine manufacturers)

### 2.2.2 Misconceptions about data

When thinking about the use of data, many people believe that they should have a lot of data on hand before feeding it into an AI system, and that having vast amounts of data will ensure the success of the AI system. Here's why this may not always work in practice:

- more data does NOT mean a perfect dataset
  - better to start relatively small and keep on continuously feeding data to the AI team
  - more often than not, the AI team provides dynamic feedback to the IT team regarding what data is useful and what type of IT infrastructure to invest in
- do NOT assume the success of an AI team just because it has a lot of data – garbage in, garbage out
  - not all data is valuable, 'bad' data will lead to AI learning inaccurate things
  - processing huge amounts of data needs appropriate infrastructure to complement it

There are many more data problems that may arise in practice, like:

- incorrect/missing values, or incomplete data: refer to the below example

SIZE OF HOUSE (SQFT)	NO. OF BEDROOMS	PRICE OF HOUSE (1000\$)
548	1	119
679	1	0.001
834	2	unknown
unknown	3	342
1367	54	399

- multiple types of data: AI algorithms work very well for all types of data, but techniques for dealing with them might vary
  - unstructured data: images, audio, video, text documents
  - structured data: tables, spreadsheets, databases

## 2.3 AI terminology

Up till now, we've been throwing around terms like AI, machine learning, neural networks, etc. Let's briefly explore what these terms actually mean.

### 2.3.1 Machine learning vs. data science

There is a thin line between what can be interpreted as machine learning and as data science. Let's say we have a dataset of houses like the one below:

SIZE OF HOUSE (SQFT)	NO. OF BEDROOMS	NO. OF BATHROOMS	NEWLY RENOVATED	PRICE OF HOUSE (1000\$)
548	1	2	N	119
679	1	2	N	167
834	2	3	Y	233
1209	3	4	Y	342
1367	4	2	N	399

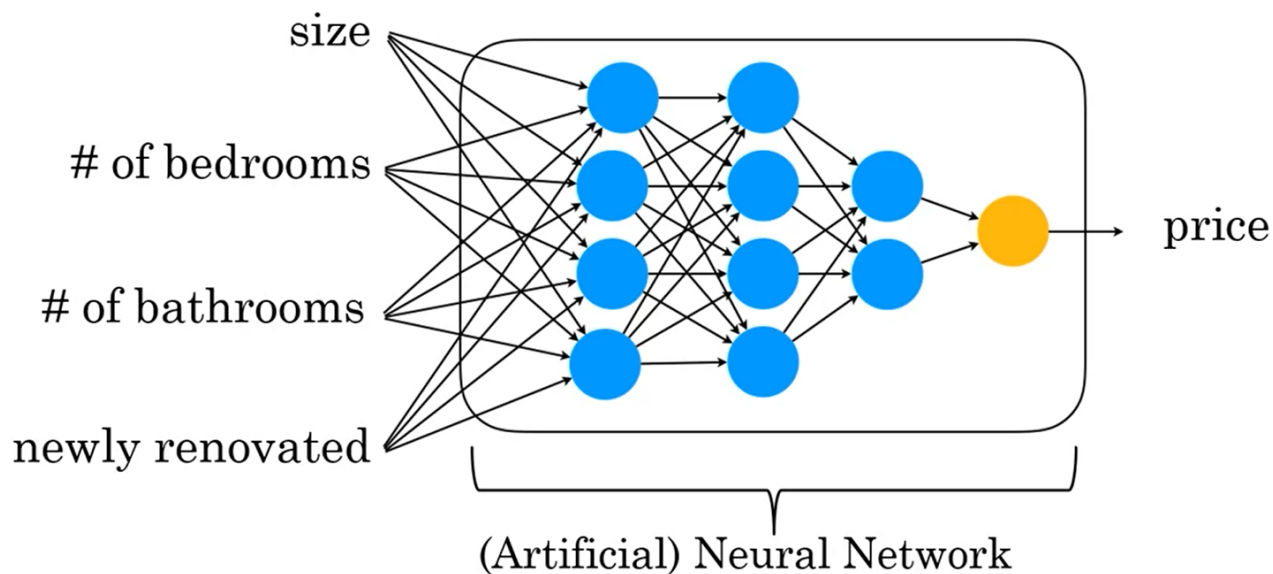
An application of this dataset to help construction companies price houses appropriately with the first 4 columns as input and the price as output would be a machine learning system, and particularly a supervised learning system. ML often results in running AI systems used on a mass scale.

In contrast, another application of the dataset is to actually let a team analyse the data in order to gain insights. They might come up with certain conclusions based on this, for example 'Houses with 3 bedrooms are pricier than 2 bedrooms of a similar size'. This can help companies take decisions on whether to build houses with 2 or 3 bedrooms, whether to renovate houses in order to sell them for a higher price, etc. This is an example of a data science project, where the output is a set of conclusions that helps companies take business decisions.

Let's take the online advertising industry as another example. Personalized ads powered by AI systems (that take ad info and user info as input and determine if the user will click on the ad or not) are machine learning systems. However when business teams analyse trends in the industry and come up with conclusions like 'the textile industry is not buying a lot of ads, but could be convinced otherwise with the right sales approach', it becomes a part of data science.

### 2.3.2 Deep learning

Let's take the same example of pricing houses. We take the 4 columns on the left as input. One of the most effective ways of generating the output would be to feed it into what are called neural networks.

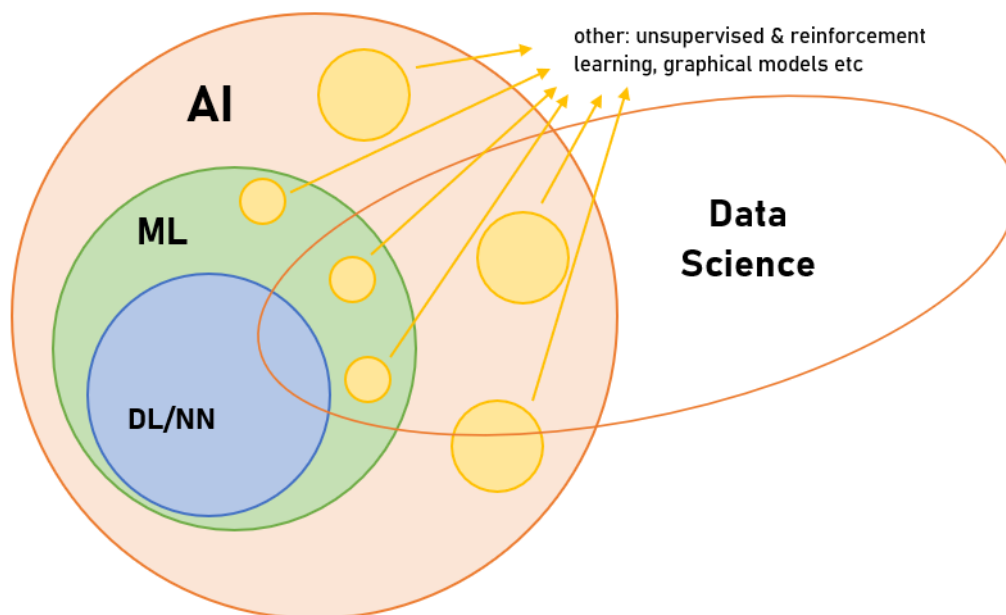


These are pretty similar to the network of neurons spread across the human body (which is also why they are referred to as artificial neural nets). This representation of ANNs bears some resemblance to the brain in that the blue circles are called artificial neurons which relay information across the network. And the resemblance ends right here. The details of how ANNs work are completely unrelated to how the human brain works.

At the end, what an ANN boils down to is nothing but a big mathematical equation that leads the system to reach the output based on a set of input parameters. This makes them very effective for learning  $A \rightarrow B$  mappings. The terms 'deep learning' and 'neural networks' are used almost interchangeably today.

### 2.3.3 The larger picture

If we were to construct a Venn diagram showing all the concepts above, we would probably have something like this:



To date, there is a discrepancy about how data science fits into this picture. Some say AI is a subset of data science while others say the opposite. However, it is better seen as a cross-cutting subset that comprises of all these tools from AI and also other tools that drive business insights.

## 2.4 AI companies

In this era, it is possible for almost any company to employ a few deep learning algorithms. However, that by itself does not necessarily make it an AI company. AI companies specialize in the following:

- strategic data acquisition: many AI companies have free products solely for the purpose of acquiring data that can be better monetized elsewhere
- unified data warehouses: pre-emptive investments in bringing data together to a unified warehouse/a small set of connected warehouses
- pervasive automation: inserting AI algorithms to automate certain generic tasks in order to apply human labour & intelligence in more specialised work roles
- specialized roles: such as machine learning engineers (MLEs); allows for better division of labour and assigning specialized tasks to increase efficiency

It turns out that there is a systematic process using which companies can implement many of the above strategies to ensure that they use AI to their maximum benefit:

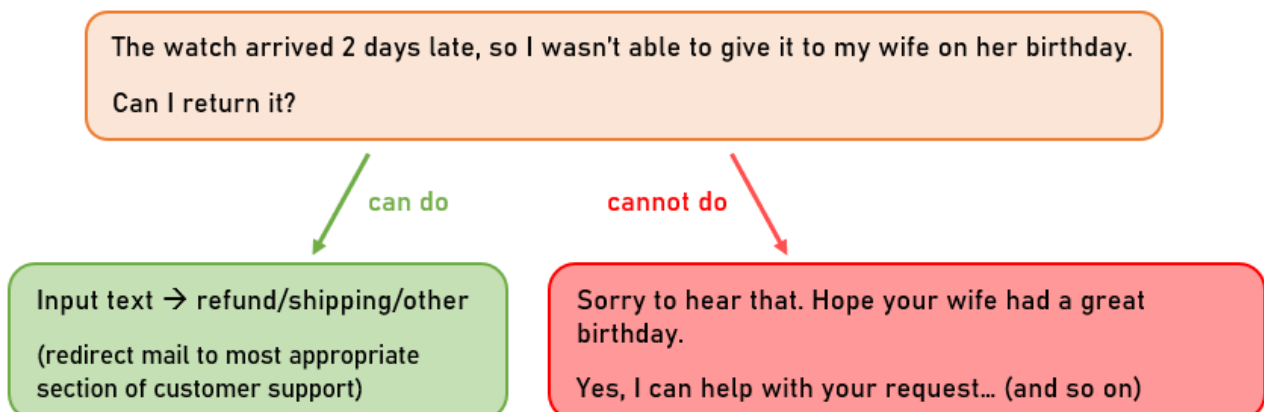
1. execute pilot projects to gain momentum and get a better sense of what AI can and cannot do, what types of data are useful, etc
2. bring together an AI team and provide extensive AI training to engineers as well as managers & executives
3. develop an AI strategy and build IT infrastructure based on dynamic feedback from the AI team
4. align internal and external communications so that others in the company hierarchy (shareholders, customers, etc) know how to navigate the rise of AI

## 2.5 Limitations of AI

Before committing to an AI project, it is important to check whether it is feasible. While the success stories we read in articles might make it sound like AI knows no bounds, this is far from reality. There are several limitations (currently, at least) as to what AI can and cannot do.

As an imperfect rule of thumb, anything that a human can do within a few seconds of thought can probably be automated using AI - for example, telling whether a phone is scratched/dented, looking around and determining positions of cars, deciphering audio, etc. In contrast, an AI probably cannot write a 50-page report based on in-depth analysis of the stock market. Let us take a look at some more examples:

- customer support automation: can sort incoming emails and redirect them to appropriate sections of customer support; cannot type out personalized responses



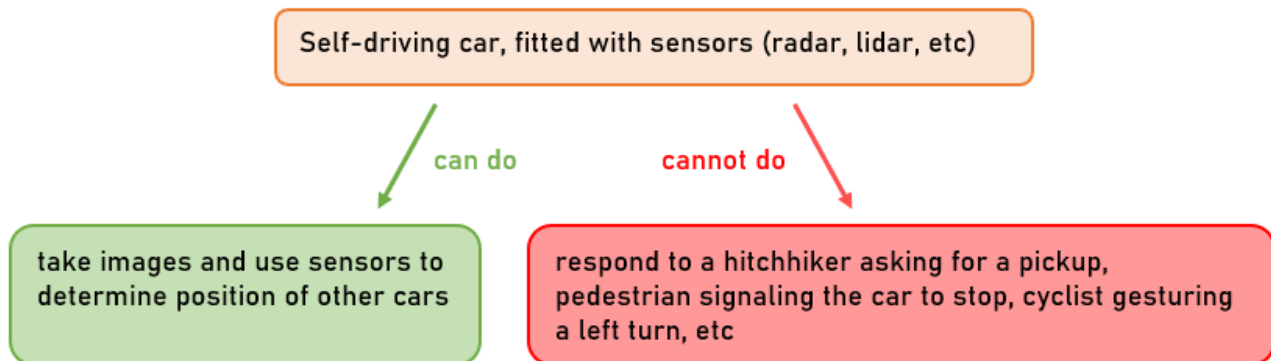
What if we try to do this anyway? Say we have a deep learning algorithm ready and a decent sized dataset of 1000 user emails and appropriate responses. We would get something like this:

"My product is damaged." → "Thank you for your email."  
"Where can I write a review?" → "Thank you for your email."  
"How do I send a product as a gift?" → "Thank you for your email."



It turns out that a sample size of 1000 (or even 100,000 as a matter of fact) is just not enough for an AI algorithm to write out appropriate and empathetic responses. In some cases, the AI may even generate gibberish which is clearly not desired.

- self-driving car: can use sensors and photographs to figure out relative positions of other cars; cannot respond appropriately to human gestures



One of the main reasons why this is so difficult to do is due to the sheer amount of possible hand gestures that could be made by humans. It is difficult to collect data of tens of thousands of people performing gestures. And again if we try to do it anyway, the consequences would be even harsher than in the scenario of customer support above.

- X-ray diagnosis: can diagnose diseases from around 10,000 labelled images (difficult to collect for rare diseases); cannot diagnose diseases based on a small set of images in a medical textbook (a human doctor can do this, however)

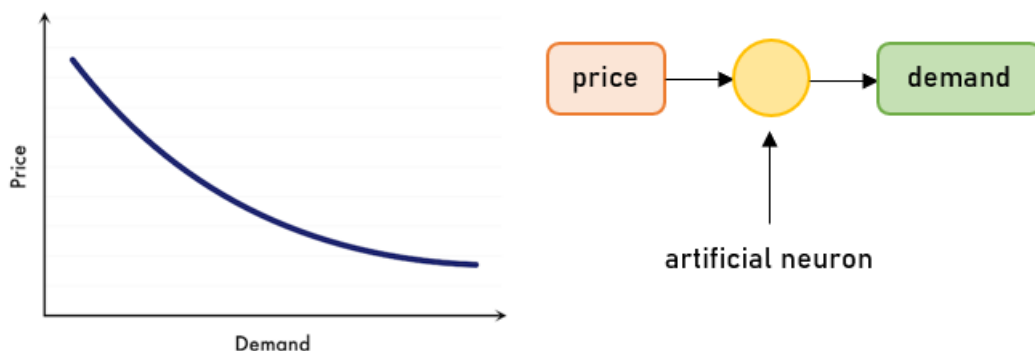
In the context of X-ray diagnosis, we can make out another weakness of AI, that is when it is asked to work with new types of data. Let's say the sample data contains high quality X-ray images. The AI algorithm will most likely fail when faced with poor-quality X-ray scans or images from even a slightly defective machine.

In the end, there are no hard and fast rules about the stuff that AI can or cannot do. Most of the times, AI projects require some weeks of technical diligence to figure out their feasibility. However, keeping the above points in mind, one should be able to get a fair judgement regarding the same.

## 2.6 Understanding deep learning

The terms deep learning and neural networks are used almost interchangeably in AI. Let us use an example of demand prediction to try and understand what neural networks really are.

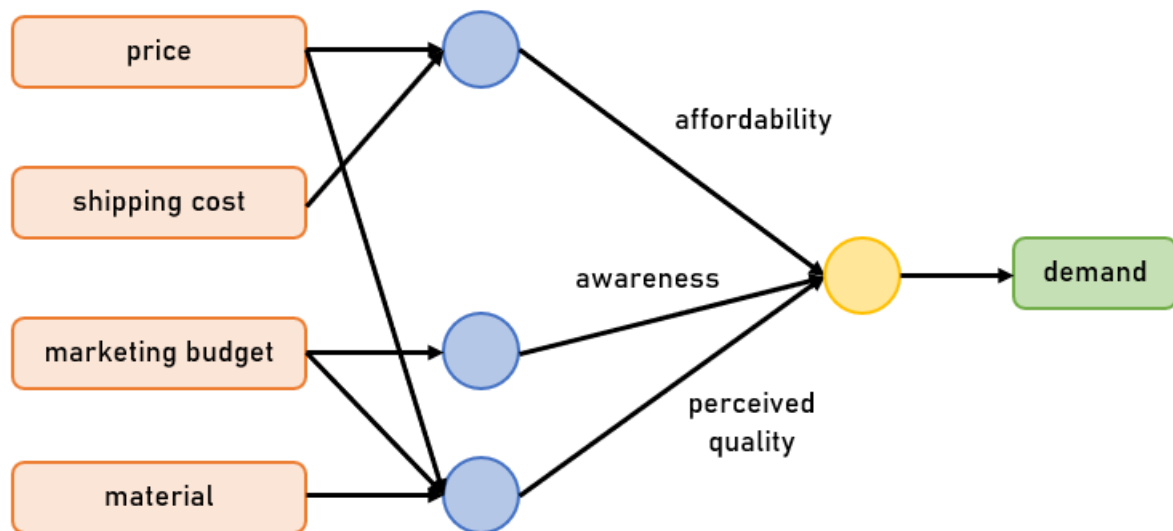
Suppose a t-shirt company wants to know how many units they can expect to sell based on their selling price. The required dataset might be in the form of a demand curve, where the higher the price the lesser the demand. This form of curve can be used to train what is perhaps the simplest possible neural network.



All this single-neuron network does is compute the curve shown and 'learn' it in order to map any value of price to the appropriate value of demand. A single neuron can be thought of as a Lego brick, and a neural network as a very complicated stack, often in multiple layers, of such bricks.

Let's look at a more complicated example. Suppose that instead of just the price, we have more variables like shipping cost, marketing cost and material. Then we will have multiple factors that influence demand like

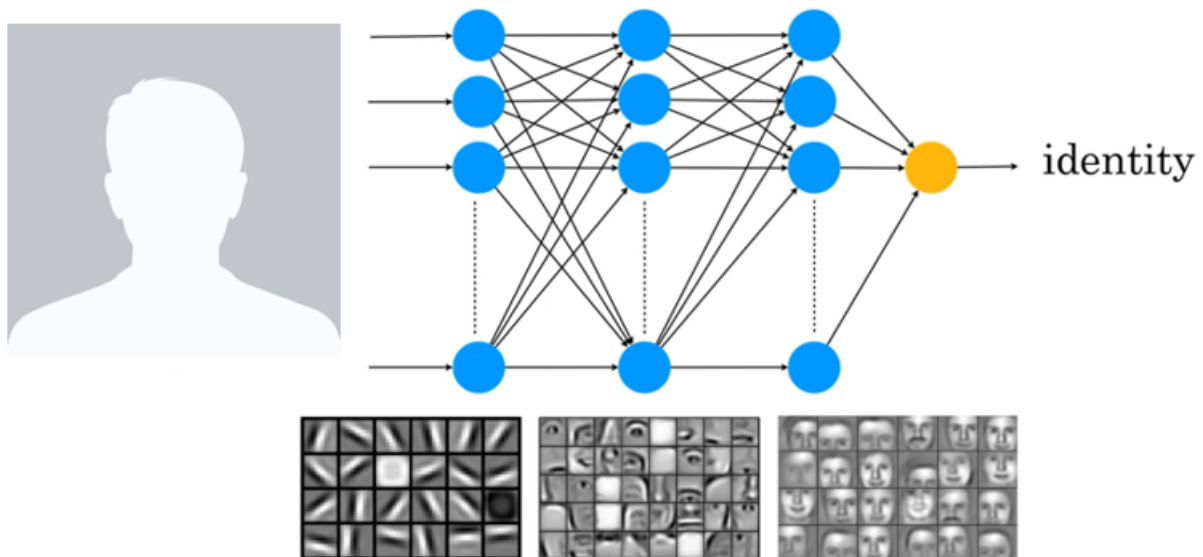
affordability, consumer awareness and perceived quality. We might then have a slightly more complicated neural net like the one below:



This slightly more complicated neural network maps the 4 input parameters to the output that is the demand.

From that the way in which we have discussed neural networks above, it appears as if we have to actually figure out the key factors as affordability, awareness and perceived quality. However, things do not work this way. One of the best things about neural networks is that we only have to provide it the input and the output – all of the stuff in the middle, it figures out by itself. It automatically ‘learns’ and completely trains itself to find the most accurate possible function that maps from the input to the output.

With this slightly advanced definition of neural networks, let us try to understand an actual practical application of neural networks in face recognition.



When we look at a face, we see certain features like eyes, expression, etc. What a neural network sees is millions of RGB values for each and every pixel in the image. Typically, when you give it an image, the neurons in the earlier parts of the network will learn to detect edges in pictures and later learn to detect parts of objects. After they learn to detect eyes and noses and the shape of cheeks and the shape of mouths, the neurons in later parts of the network will learn to detect different shapes of faces and finally, will put all this together to output the identity of the person in the image.

---

## END OF WEEK 1

This is the end of the course documentation from Week 1. Keep on reading further for documentation from further weeks, or spend some time gaining further insight into the previously discussed topics.

---

## 3 Building AI Projects

So far we have covered the basics of AI and machine learning. But how do we put this technology to use in a project? Let's take a look.

### 3.1 Workflow of a machine learning project

There are 3 basic steps when building a machine learning project. Let us take speech recognition as an example, particularly Google speech recognition. How do you build a system that recognizes the words 'Ok Google'?

1. Collect data: involves collecting some audio clips of people saying the words 'Google' and 'Ok' (and lots of other words too - we want speech beyond 'Ok Google' to be recognized too)
2. Train the model: use a ML algorithm to learn input to output mappings
  - often the first attempt doesn't work very well
  - need to keep on iterating over the algorithm until it is good enough
3. Deploy the model: package the software into a product and ship it
  - may not work as well initially due to lot of new data (eg: if the sample dataset was from American users, the AI may not be able to recognize 'Ok Google' from Indian users as well initially)
  - get back user data (while maintaining privacy regulations) to maintain and update the model

Let us revisit this process in another example of self-driving cars:

1. Collect data: sample images and, for each of the images, position of nearby cars
2. Train the model: invariably the first model won't work well (eg: may detect trees or rocks as cars initially); need to reiterate until the model is good enough according to safety standards
3. Deploy the model: must be done in ways that preserve safety; get new data back (eg: new types of vehicles like tow trucks, auto rickshaws, etc) and update the model continually to the point that it can be released to the commercial market

### 3.2 Workflow of a data science project

Unlike a ML project, the output of a data science project is a set of insights that can be used to influence business decisions. Naturally it follows that they have a different workflow compared to ML projects. Let us take the example of an e-commerce platform that sells coffee mugs. There might be several steps a user has to perform to buy the product.



As a salesperson, it is our job to make sure that the majority of users get through all these steps. This analysis is done in a series of steps:

1. Collect data: gather user info (country, time, product they checked out, price they were offered, where they quit in the buy process)
2. Analyse data: get a data science team to work on the dataset
  - initially, the team might have a lot of ideas as to why users are not motivated to buy the product
  - need to iterate the analysis to get good insights and find out the major causes (eg: shipping costs are too high, so users quit at the checkout page)
3. Suggest hypothesis & actions: data science team presents the insights and suggests any suitable actions (eg: incorporate part of shipping costs into product cost)

- deploy changes to the product design and get new user data back (eg: users overseas may now buy more, but locals may not due to rise in base price)
- re-analyse new data continuously to possibly come up with even better hypothesis/suggestions

Again, let us briefly discuss this framework in another context of optimizing a manufacturing line for coffee mugs. We will want to make sure that as little defective mugs are produced as possible:



1. Collect data: data about different types of clay (suppliers, mixing times, moisture content, etc) and regarding different batches of mugs (temperature of kiln, duration in kiln, defects per batch, etc)
2. Analyse data: ask the data science team to analyse data to find the major source of defects (eg: too high temperature might lead to softening of clay and cause mugs to crack)
3. Suggest hypothesis & actions: change operations (eg: vary humidity and temperature based on time of day), get dynamic feedback from manufacturing output and take further actions if necessary

### 3.3 Impact of data on job functions

The digitization of society means that more and more data is being stored in digital formats. Due to this, almost all jobs have been or will be impacted by the advent of machine learning and data science. Let us see briefly how data science & ML have made or are making their way into different industries:

	<b>Data Science</b>	<b>Machine Learning</b>
SALES	optimising a sales tunnel (refer Section 3.2)	automated lead sorting - prioritize certain marketing leads over others (contact CEO of large company over an intern at small company)
MANUFACTURING	optimising a manufacturing line (refer Section 3.2)	visual product inspection (AI algorithms can learn to figure out if products are defective or not)
RECRUITING	optimising the recruiting process (analysing why people are not making it to certain stages or why too many people are making it)	automated resume screening based on sample dataset of resumes and whether to select candidate or not (fair and ethical screening free of any bias)
MARKETING	A/B testing - launching 2 versions of a website to find out what appeals to consumers	customized product recommendations to significantly increase sales
AGRICULTURE	crop analytics (find out what to plant and when to plant based on market conditions, soil & weather conditions, etc)	precision agriculture (recognize presence of weeds through images/video and spray an appropriate amount of weed killers)

Of course, this list is not exhaustive and there are many more industries which have seen or will see the impact of AI soon enough.

### 3.4 Choosing an AI project

There are definitely a lot of things we can try to do with AI, but how to we choose an AI project? Let us discuss a general framework on how to choose an AI project. Many of these points have already been discussed, but let us revisit them in this context:

- feasibility and value addition:

- must be feasible to do with AI and also add value to the business/application
- brainstorming with cross-functional teams (comprising both AI experts as well as business domain experts) to narrow down projects
- brainstorming framework:
  - think about automating tasks rather than entire jobs (eg: for radiologists, AI might be useful in X-ray diagnosis but not as useful in consulting with other doctors or patients)
  - consider main drivers in business value and try to augment them using AI to increase the scale and productivity of the business
  - consider tasks which are particularly painstaking for humans and try to automate them if possible
- do not insist on acquisition of big data
  - try to make progress with small datasets and get dynamic feedback from the AI team as to what type of data to obtain further and what type of IT infrastructure to build

After brainstorming and narrowing down to a certain list of projects, it is time to pick one to work with. Committing to an AI project requires a lot of work to see it through and hence, it is important to conduct due diligence on it. Technical diligence is used to make sure the task is feasible to carry out using AI, while business diligence revolves more around deciding how much value it is going to add and if it is worth the effort.

TECHNICAL DILIGENCE	BUSINESS DILIGENCE
can an AI system meet the desired performance	will automation lower costs enough?
how much data is needed	how much revenue/efficiency it will increase
engineering timeline (how long and how many people it will take)	will launching this project bring enough value to the business?
	building spreadsheet financial models to estimate the value quantitatively

Another type of diligence one should try to perform is ethical diligence (i.e. is the society/environment being harmed). Any AI project should ideally add value to society and if not, it should not cause harm at least.

Another factor we need to consider is whether we want to build or buy (or maybe do a combination of both):

- outsourcing ML projects can lead to easier access to datasets
- data science projects are more commonly done in-house since they are highly tied to the business (it takes very deep insider knowledge about the business which is unlikely to occur through outsourcing)
- try and build things that will be specialized to the project
- avoid building things that are an industry standard (eg: storage servers, computer hardware, etc)

### 3.5 Working with an AI team

After a project is finalised, one must know how to work with an AI team in order to make sure that the project runs smoothly. Normally a business should have already an AI team but even if not, it is fairly easy to either hire AI engineers or learn a thing or two about AI yourself and get enough knowledge to get started. After this, we need to consider the following points while working with an AI team:

- specify the acceptance criteria (eg: goal is to detect defects in coffee mugs with 95% accuracy) and provide the dataset to the AI team to measure the accuracy
  - training set – dataset containing both input and output using which the AI learns the mapping
  - test set – dataset with new input data which the AI team gives the algorithm to see what it outputs
- do not expect 100% accuracy – AI has its limitations and so does the amount of data being fed into the training set
- keep taking feedback from the AI team as to whether the training data needs to be improved, etc

It is often a good idea to be in constant touch with the AI team to try and find a reasonable level of accuracy that passes technical as well as business diligence (since it might not be worth it if the AI has low accuracy).

### 3.6 Technical tools for AI teams

Let us finally discuss some of the most commonly used tools by AI teams to train and test learning algorithms. It is important to have a little knowledge about this since it gives us a better insight when communicating with the AI team.

- machine learning framework: open source frameworks like PyTorch, TensorFlow, etc make writing software for ML systems much more efficient
- research publications: AI technology breakthroughs published freely on the internet (a major source is Arxiv) and on GitHub repositories
- CPUs and GPUs: CPU does majority of the computational work, while GPU hardware is very powerful for building large neural networks
- cloud deployments: refers to renting computer servers such as from AWS, Azure, etc to use external servers to do the computation (as opposed to on-premises deployment, which means buying your own servers)
- edge deployment: putting a processor right where data is collected to process data and make quick decisions
  - for a self-driving car, there is not enough time to record data, send data to a cloud server and receive a response from the server
  - computation must happen quickly inside the car

---

#### END OF WEEK 2

This is the end of the course documentation from Week 2. Keep on reading further for documentation from further weeks, or spend some time gaining further insight into the previously discussed topics.

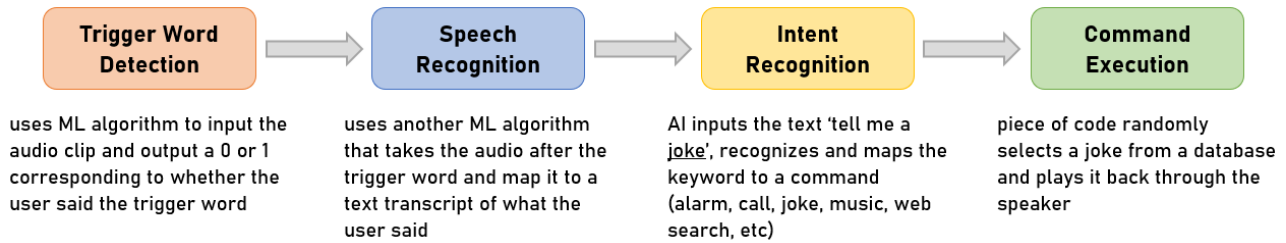
---

## 4 Building AI in a Company

Up till now, we have discussed what AI is and how to build an AI project. Let us now discuss some complex AI projects in depth and find out how AI projects fit in the context of a company.

### 4.1 Smart speakers: a case study

Let us go through a brief case study of how AI software is written to respond to voice commands like ‘Ok Google, tell me a joke’. There is a sequence of steps (known as an AI pipeline) needed to process the command:



Quite often, AI teams are split into various specialized groups and each of them focus on specific tasks in the AI pipeline. The AI pipeline is reasonably flexible in that it might change on occurrence of a slightly more complicated command like ‘Ok Google, set a timer for 3 minutes’.

In such a case, majority of the earlier process remains the same except the final step. The execution step will be separated into 2 further steps:

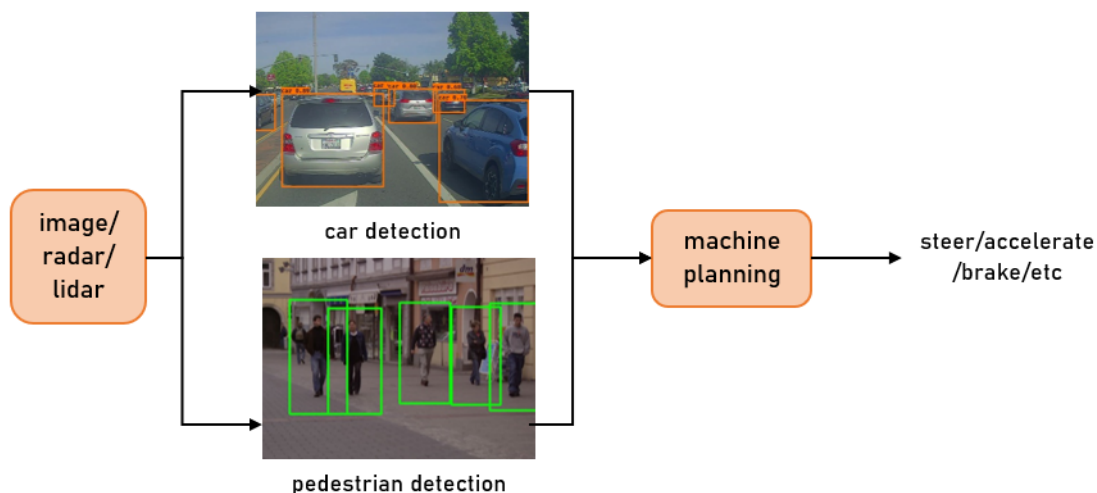
- extract duration – look at the text and pull out the phrase corresponding to the time duration
- command execution – specialized software component that can start a timer with the set duration

Smart speakers have a lot more functions apart from the above two and given the function we want to perform, we can figure out what the AI pipeline will look like. A major challenge for teams building a smart speaker lies in the intent recognition stage - a user can ask for a particular command in many ways (eg: tell me a joke, do you know any good jokes?, say something funny, make me laugh, etc).

To further solidify our understanding of the AI pipeline, let us look at a second case study on self-driving cars.

### 4.2 Self-driving cars: a case study

Let us try to construct the AI pipeline of how the AI in a self-driving car might make decisions on how to drive.



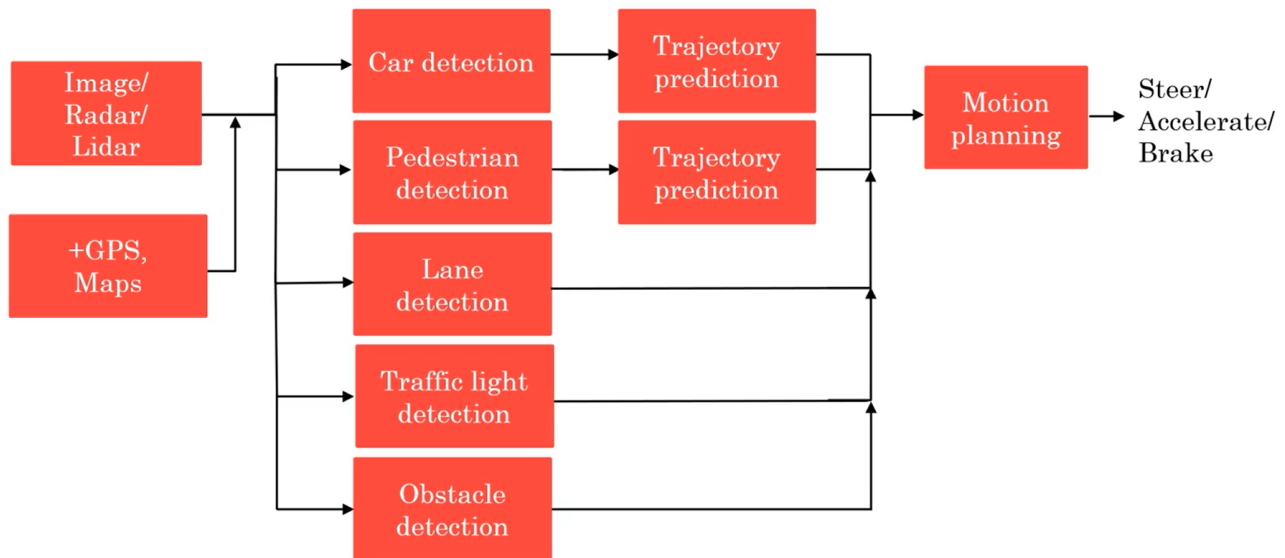
Let us look into some details regarding car detection, pedestrian detection and motion planning:

- car detection – supervised learning algorithm to input images from all sides of the car, radar info and output positions of nearby cars
- pedestrian detection – very similar to car detection techniques

- motion planning – outputs the path the car should take as well as safe speed to avoid collisions with other cars (as well as overtaking parked cars)

This is a fairly simplified pipeline of how a self-driving car might work. Most self-driving cars would have an auto navigator that relies on GPS, and also use other stuff like gyroscopes.

Another component in self-driving cars is that of trajectory prediction. Instead of just knowing the current position of cars, it is more useful to be able to predict how they are likely to move in the next few seconds (by focusing on turn indicators, brake lights, etc) to be able to avoid them even as they are moving. To drive safely we need to consider a lot of other factors, which might lead to a fairly complicated AI pipeline as below:



### 4.3 Major roles in an AI team

In the previous example of a self-driving car, we saw a pretty complicated AI pipeline. To pull off a project like this, there are usually large AI teams divided into small groups to work on specialized tasks. Let's take a look into the major roles in such a team:

- software engineer: write specialized software for command execution
- machine learning engineer: write software responsible for  $A \rightarrow B$  mappings, collect data and train neural networks by iterating over the algorithm
- machine learning researcher: perform research on how to extend state-of-the-art in ML, publish papers and maintain documentation
- machine learning scientist: go through academic & research literature to find ways to implement state-of-the-art technology to the current application
- data scientist: examine data and gain insights; further discuss and present the insights to the team
- data engineer: organize data and saving it in an easily accessible, secure and cost-effective way
- AI product manager: help decide what is feasible and valuable, and what to finally build; manage deadlines and coordinating with the team

Even though having a large team generally helps due to the possibility of specialized jobs, it is possible to get started with a relatively small team of about 3-5 people. Even a basic knowledge can help to start analysing small amounts of data and training basic ML models.

Up till now, we have seen what an AI team comprises, but AI teams often need to work with other departments in the company to bring out maximum efficiency. This is where the AI transformation playbook comes in.



## 4.4 The AI transformation playbook

Just having a large AI team at its disposal does not make a company a good AI company. Many of the top AI companies follow most of the points in the AI transformation playbook in some way or the other (this is a term coined by Andrew Ng, a pioneer in ML technology who also happens to be the instructor for this course, but is essentially a list of points that, if followed, is likely to make a company good at AI):

1. Execute pilot projects to gain momentum
  - initial project should be successful rather than valuable (helps gain the confidence of other departments in the company and investors)
  - project should preferably show quick traction (within 6-12 months) since it helps gain the required momentum and experience for future projects
  - can outsource data initially to work upon it and gain experience before building an in-house team
2. Build an in-house AI team
  - centralized AI team instead of each business unit having their own AI component
  - build company-wide platforms (software tools, data infrastructure, etc)
3. Provide broad AI training
  - executives and seniors should know the basics of AI and how to make resource allocation decisions
  - AI division leaders should know how to conduct diligence on projects before selecting them and monitor the progress of different teams working on the project
  - AI engineers/trainees should be able to collect the right type of data, build algorithms based off of it and then assemble various modules of the project to ship it
4. Develop an AI strategy
  - leverage AI to create an advantage specific to the industry sector
  - this point is not at number one because before planning out a strategy, it is important to have basic knowledge regarding AI (for example, one may put forth a strategy to collect a lot of data, but is that data really useful?)
  - the virtuous cycle of AI:
    - better product → more users → more user data (which can be used to update the model) → better product (and so on)
    - difficult for newer companies to enter the market since already existing companies are likelier to have a better product and larger user base
    - small teams can take advantage of this by pushing into lesser explored industries and capitalizing later on their specialization
  - strategic data acquisition (eg: launching free services solely for data collection purposes)
  - unified data warehouses let engineers connect the dots (eg: in case of customer complaints, it is better to have the manufacturing data in the same place to find out exactly what caused the fault)
5. Develop internal and external communications
  - how AI will benefit the company must be communicated to investors so relations are maintained
  - communicate with the government to introduce AI solutions in areas like health care, and even other areas where user privacy regulations are strict
  - user education as to how to deal with introduction of new technology
  - may also help in recruitment procedures for the company

If you wish to go through the entire AI transformation playbook in much greater detail, you may do so here: <https://landing.ai/ai-transformation-playbook/>.

## 4.5 First steps and pitfalls

Even after following all the points in the AI transformation playbook, there are companies which are subject to common pitfalls while taking their first steps towards an AI project. Let us look at some of them:

- expecting AI to solve everything (instead, one should give due time to technical and business diligence to figure out if the project is actually possible and valuable using AI)
- hiring employees of only one type i.e. only ML engineers or only software engineers, etc (there should be a cross-functional team that pairs business talent with engineering talent)
- expecting AI projects to work right away (more often the first attempt is a disaster, but continuous iteration of training data over the algorithm can refine it a lot)
- expecting traditional business approaches to work (instead, one should build a separate AI team and work with them for a more dynamic progress)
- feeling the need to have a large team and a huge amount of data (often small teams working with small datasets can gain better insights before investing further in data collection)

Much of what we have discussed so far might sound great but can feel daunting when it comes to putting things to practice. In order to take the first step into the world of AI, it is better to get a certain group (maybe a group of friends) to learn about AI instead of going in solo. It is also important to brainstorm on different projects, even if they seem to small - it is better to start small and succeed rather than to go big and fail.

Now that we have looked through the basics of AI technology and how to build and choose AI projects, let us now go through slightly more advanced concepts like reinforcement learning, computer vision, natural language processing, etc.

## 4.6 Major AI application areas

AI today is being successfully applied in many industries involving image & video data, language data, speech data, etc. Let's take a deeper look into a brief survey of how AI is applied to these different areas.

### 4.6.1 Computer vision

One of the major successes of deep learning has been computer vision. To understand what computer vision is, let us look at some of its applications.

One of these applications is image classification/object recognition. For example, AI can take a picture and tell the user if it is of a cat. This technology is actually much more advanced than just recognizing cats – there are applications to detect certain species of flowers, types of food, etc. And of course, one of the major implementations of this lies in face recognition.

Another type of computer vision algorithm is that of object detection. Hence rather than just recognize an object and label the whole image, we're trying to detect the different types of objects in the same image. For example, a self-driving car AI does not only recognize cars & pedestrians but also shows their position. It is also possible to track the live position of cars, etc as they are moving.

Image segmentation takes this one step further – it tells us not just where the cars are but also for every single pixel in the image, if that pixel is a part of a car or of a pedestrian, etc. This is commonly used in X-ray scans to detect particular organs.

### 4.6.2 Natural language processing

NLP refers to AI understanding everyday human language. One application of NLP lies in text classification (eg: to take an email and categorize it as spam or not spam, or take a product description and figure out which category it belongs to).

One type of text classification used widely is sentiment recognition & analysis. This type of AI can take input like “The food was good, but the ambience can be improved” and output an expectation of the star rating the restaurant might get.

A second type of NLP is information retrieval, of which web search is probably the best example. This AI will take in a search string as input and help the user find relevant documents. Companies also use this internally to search within particular sets of documentation.

Name entity recognition is another type of NLP, which helps filter out certain types of names in large documents. For example, the input ‘I am fluent in English, Spanish and Japanese’ should lead the NLP algorithm to filter out English, Spanish and Japanese as language names. They can also recognize phone numbers, countries, etc.

NLP also features in applications of machine translation, where the AI translates an input phrase or document into a specific language that the user wants. Apart from the above, NLP can be used in a wide range of other applications like parsing, part-of-speech tagging (classify which words in a string are nouns, verbs, etc), etc.

### **4.6.3 Speech**

Modern AI technology has completely transformed how speech is processed by computers. A microphone records rapid variations in surrounding air pressure. These variations can be represented on a plot against time on a computer (this is the information a typical digital audio waveform contains). Speech recognition deals with taking inputs of a plot like this and figure out what the user said.

As we discussed earlier, one particular type of speech recognition is trigger word/wakeword detection. Another type of speech recognition used is speaker ID, where the problem is to take an audio clip and figure out the identity of the speaker.

Recently, text-to-speech has also gained a lot of traction. TTS is used in audio books and as aid in communication for people lacking the ability of speech.

### **4.6.4 Robotics**

One of the applications in robotics we have already seen is that of a self-driving car. One of the main challenges with robotics are perception (figuring out what is in the world around us), motion planning (computing a path for the robot to follow) and control (sending commands to motors & other electric components to execute the calculated path).

### **4.6.5 General machine learning**

A majority of the applications we saw dealt with unstructured data. However ML works just as well for structured data. Structured data processing is often more specific to a company and it is harder for humans to understand the data at first glance and/or empathize with it.

## **4.7 Major AI techniques**

In the previous sections, we have mainly discussed only supervised learning. This term almost invites the question as to what is unsupervised learning? Apart from this, there are other AI techniques like reinforcement learning as well.

### **4.7.1 Unsupervised learning**

The best known example of unsupervised learning is that of clustering. Suppose there is a shop that specializes in selling chocolates. It may sell cheap, low quality chocolates as well as expensive but high quality ones. If consumer buying patterns are plotted on a graph, it may reveal that buyers are separated into 2 groups – some buy a lot of cheap chocolates while others buy a few expensive chocolates. A clustering algorithm automatically sorts this data into 2 clusters and this data can be used for market segmentation (for example, in an area with a nearby university, college students are more likely to buy cheaper chocolates so it would make more sense for the shop to stock the cheap ones).

Unlike supervised learning, an unsupervised learning algorithm does not tell a system exactly what output it wants. Instead what it does is feed the data and ask the system to find something peculiar or meaningful. In the above example, the algorithm just finds what are the different market segments without being told they belong to college students or working people, etc.

Unsupervised learning overcomes one of the major criticisms against supervised learning, which is that it requires a lot of labelled data. This opens a possibility to AI being able to learn much more like a human from much lesser data in the future.

### **4.7.2 Transfer learning**

Say we have built an algorithm for self-driving cars and start deploying it. There might be regions with different types of vehicles like golf carts. Even though our system has been trained with a lot of data, it will contain very few images of golf carts. Instead of building another algorithm for golf cart detection from the ground up,

we can use the technique of transfer learning. This technique lets us learn from task A and use the knowledge to help with task B. It particularly works very well if the dataset from task A is large.

#### 4.7.3 Reinforcement learning

Let's say we want to write an algorithm for automating a helicopter. It is difficult to do this by supervised learning because, while in the case of card we had only a few types of objects to deal with, here we have a lot more. It is very difficult to specify what is the best way to fly an helicopter when it is in a certain position. This is where reinforcement learning comes in.

It is quite similar to how one would train a pet. A simulation is created to let the helicopter fly around. Whenever the AI flies the helicopter well, we give it a positive feedback and when it crashes, we give it a negative feedback. Over time, the AI learns how to fly the helicopter as it keeps on getting dynamic feedback. More formally, reinforcement learning uses a 'reward signal' to tell the AI whether it is doing well or poorly – it automatically learns to maximize its positive rewards.

In addition to this, reinforcement learning has also helped build algorithms for playing strategic games like Othello or chess, and also for playing video games. One of the weaknesses of reinforcement learning is that it is difficult to control the amount of data the system feeds to its own learning algorithm.

#### 4.7.4 GANs (generative adversarial networks)

This AI technique was only recently designed in 2014. GANs are very good at synthesizing images from scratch. This aspect has caused GANs to gain a lot of traction in everything involved with computer graphics like video games and other media.

The general idea of GANs is of 2 networks (generative and discriminative) working hand in hand. The generative network generates stuff while the discriminative network evaluates it. The job of the generative network is to increase the error rate of the discriminative network (i.e. to try and fool the discriminative network as much as possible by producing novel images based on a true data distribution that the discriminative network thinks are not actually synthesized).

#### 4.7.5 Knowledge graph

One of the more underrated AI techniques are knowledge graphs. A web search about a famous personality often leads in a huge string of links and a panel to the right showing some of their general details. These details are derived from a knowledge graph, which is nothing but a database that lists people and key information about them. Many companies build such databases of celebrities, movies, hotels, etc and can help extract a lot of important data relatively quickly.

---

### END OF WEEK 3

This is the end of the course documentation from Week 3. Keep on reading further for documentation from further weeks, or spend some time gaining further insight into the previously discussed topics.

---

## 5 AI and Society

*(This section is currently in the works. Expect it to be up typically within a week's time.)*

---