BRIEFING

European Parliament

How artificial intelligence works

From the earliest days of artificial intelligence (AI), its definition focused on how its results appear to show intelligence, rather than the methods that are used to achieve it. Since then, AI has become an umbrella term which can refer to a wide range of methods, both current and speculative. It applies equally well to tools that help doctors to identify cancer as it does to self-replicating robots that could enslave humanity centuries from now. Since AI is simultaneously represented as high-risk, low-risk and everything in between, it is unsurprising that it attracts controversy.

This briefing provides accessible introductions to some of the key techniques that come under the AI banner, grouped into three sections to give a sense the chronology of its development. The first describes early techniques, described as 'symbolic AI' while the second focuses on the 'data driven' approaches that currently dominate, and the third looks towards possible future developments. By explaining what is 'deep' about deep learning and showing that AI is more maths than magic, the briefing aims to equip the reader with the understanding they need to engage in clear-headed reflection about AI's opportunities and challenges, a topic that is discussed in the companion briefing, Why artificial intelligence matters.

First wave: Symbolic artificial intelligence

Expert systems

In these systems, a human expert creates precise rules that a computer can follow, step by step, to decide how to respond to a given situation. The rules are often expressed in an 'if-then-else' format. Symbolic AI can be said to 'keep the human in the loop' because the decision-making process is closely aligned to how human experts make decisions. Indeed, any intelligence in the system comes directly from the encoding of human expertise. Furthermore, humans can easily understand how these systems make specific decisions. They can easily identify mistakes or find opportunities to improve the programme, and update the code in response. These systems have limits. In order to develop a useful and reliable system that works for complex and dynamic real world problems, you would need so many rules and exceptions that the system would very quickly become very large and complicated. They are really at their best in constrained environments that do not change much over time, where the rules are strict and the variables are unambiguous. For example, they are useful for helping people to calculate their taxes, based on their income, circumstances and the various levies, allowances and exceptions that apply to them.

Fuzzy logic: Capturing intuitive expertise

Where each variable is either true or false, the system needs absolute answers. However, these are not always available. Fuzzy logic allows variables to have a 'truth value' between 0 and 1 so, for example, patients can be assigned a figure representing the extent to which they might have a certain illness.

By including many variables and values, fuzzy systems can deal with borderline cases, and are particularly useful for capturing intuitive knowledge where experts make good decisions in the face of wide ranging and uncertain variables that interact with each other. They have been used to develop control systems for cameras that automatically adjust their settings to suit the conditions, and for stock trading applications to establish rules for buying and selling under different market

Author: Philip Boucher
Scientific Foresight Unit (STOA)

PE 634.420 – March 2019

conditions. These continually assess dozens of variables and follow human-designed rules to adjust truth values and – on the basis of these – to make and implement decisions.

Good old-fashioned artificial intelligence

Symbolic AI systems require human experts to encode their knowledge in a way the computer can understand. This places significant constraints on their degree of autonomy. While they can perform tasks autonomously, they can only do so in the ways in which they are instructed, and they can only improve by direct human intervention. This makes symbolic AI less effective for complex problems where not only the variables, but also the rules, change in real time. Unfortunately, these are the problems with which we need the most help. For example, to really capture how a doctor draws on their knowledge and expertise to make a decision would require millions of 'if-then-else' rules and, even then, it might never hope to codify the human doctor's intuitive and emotional intelligence. Nonetheless, symbolic AI is far from obsolete, and is particularly useful in supporting humans working on repetitive problems in well-defined domains, from automated machine control for building management to decision support systems for accountants. Its reliable performance in these domains earned it the endearing nickname 'good old-fashioned AI'.

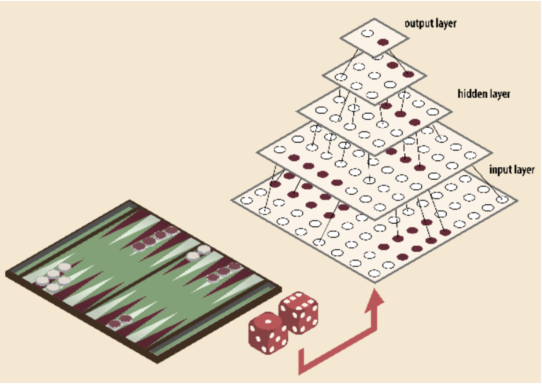
Second wave: Data-driven machine learning

Machine learning (ML) refers to algorithms that autonomously improve their performance, without humans directly encoding their expertise. Usually, ML algorithms improve by training themselves on data, hence 'data-driven' AI. The major recent advances in this field are not due to major breakthroughs in the techniques per se but, rather, through massive increases in the availability of data. In this sense, the tremendous growth of data-driven AI is, itself, data-driven. Usually, ML algorithms find their own ways of identifying patterns, and apply what they learn to make statements about data. Different approaches to ML are suited to different tasks and situations, and have different implications. Some such approaches are explored in the following sections.

Artificial neural networks and deep learning

As the name suggests, artificial neural networks (ANNs) are inspired by the functionality of the electro-chemical neural networks found in human (and animal) brains. The working of the brain remains somewhat mysterious, although it has long been known that signals from stimuli are transmitted and altered as they pass through complex networks of neurons. In an ANN, inputs are passed through a network, generating outputs that are interpreted as responses.

Figure 1 – Schematic of an artificial neural network for playing backgammon



Source: EPRS.

An ANN schema is set out in figure 1. The process starts with signals being sent to the 'input layer', and ends with a response being generated at the 'output layer'. In between, there is one or more 'hidden layer', which manipulates the signal as it passes through, so that it generates a useful output. For example, for an ANN that can play backgammon, the game situation – including the dice roll and position of pieces on the board – would be translated into a set of numbers which are sent as inputs to the ANN at the input layer. The, signals then pass to the next layer, so neurons in this hidden layer receive several numbers. Each neuron in this layer combines and manipulates these signals in different ways to generate a single numerical output. For example, one neuron might add all of the inputs and output 1 if the total is over 50, or 0 if it is not. Another neuron might assign weights to different input signals, multiply each input signal by its weight, and add the results to give as its output. The outputs of these neurons then pass as signals to the next layer. When the signals reach the final layer and its own output is generated, the process is complete. The final signal can be interpreted, in the case of the backgammon game, as instructions to move counters on the game board.

Now we have a simple ANN, inspired by a simplified model of the brain, which can respond to a specific input with a specific output. It doesn't really know what it is doing, or even the rules of backgammon. But if we give it a game position, it will always suggest a move. The question is, how to develop an ANN that can make smart moves and be a good player? First, it needs to have the right structure. For simple tasks, ANNs can work well with just a dozen neurons in a single hidden layer. Adding more neurons and layers allow ANNs to tackle more complex problems. Deep learning refers to the use of big ANNs, featuring at least two hidden layers, each containing many neurons. These layers allow the ANN to develop more abstract conceptualisations of problems by splitting them into smaller sub-problems, and to deliver more nuanced responses. It has been suggested that three hidden layers are enough to solve any kind of problem although, in practice, many ANNs include millions of neurons organised in dozens of hidden layers. By way of comparison, human brains contain ~100 billion neurons, cockroach brains ~1 million and snail brains ~10 thousand.

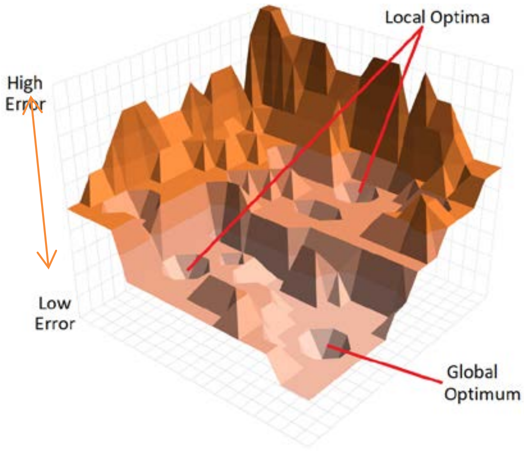
So if the 'deep' part of deep learning is about the complexity of the ANN, what about the 'learning' part? Once the correct structure of the ANN is in place, it needs to be trained. While in theory this can be done by hand, it would require a human expert to painstakingly adjust neurons to reflect their own expertise of how to play a good game. Instead, a ML algorithm is applied to automate the process. The training process can be an intensive and complicated process, and often never really ends, as constant updates respond to new data availability and changes in the problem faced. Once a well-trained ANN is in place, it can be applied to new data very quickly and efficiently.

Training neural networks: Back propagation and gradient descent

If we compare the actual output of an ANN to the desired output as reported in the labelled data, the difference between the two is described as the **error**. Back propagation and gradient descent improve the ANN's performance by using calculus to gradually minimise this error. Back propagation deals with adjusting the neurons in the ANN. The process starts with an input signal passing through the ANN and generating an output signal. This is compared to what it should have been – according to the labelled data – to calculate the error. Now, calculus is used to generate an error signal which passes backwards through the ANN, making changes to neurons so that it gives an output with a lower error. It starts with the output layer, which has a stronger impact on the result, and then moves back through the hidden layer(s) to make deeper changes. In this sense, back propagation takes the error and propagates it backwards through the ANN.

In theory, it is possible to calculate the error for every possible ANN and then choose the best one but, in practice, there are too many possible configurations for this to be feasible. A smarter approach is required. Gradient descent is often compared to a hiker that needs to find their way down a mountain, but they can only see one metre in each direction, so they adopt a strategy of looking around, deciding which direction offers the steepest descent, moving in that direction, and then looking around again and repeating the process until they find their way down the mountain. Similarly, an ANN can be generated, starting at a random point on the error landscape depicted in figure 2. Through back propagation, its error is calculated and a few different kinds of small changes are tested and evaluated. The option that offers the best improvement is assumed to be the best direction, so the changes are implemented and then the process is repeated with a new set of tests. Just as the hiker takes the steepest possible step down the mountain, the ANN makes gradual

Figure 2 – An error landscape



Source: EPRS.

improvements until it 'converges' on the best possible solution, known as the 'global optimum'. Of course, the approach is not perfect. Just as the unfortunate hiker can get stuck in a hole near the top of the mountain, where moving a metre in any direction would make them ascend, the algorithm can settle for a 'local optimum' which is not the best solution, but every small change makes it worse. This is why, in practice, the whole exercise is repeated many times, with different starting ANNs and a lot of training data.

Inspired by nature: Evolutionary training methods

While gradient descent and back propagation are based upon mathematical concepts such as calculus, here we will explore methods inspired by evolutionary concepts such as survival of the fittest, reproduction and mutation. There are many approaches within this family, but the broad principle remains the same. A population of ANNs is created. They compete against each other and are subjected to artificial selection – the AI equivalent of natural selection – so that those that perform badly are filtered out, while those that perform well survive to the next generation. To replenish the population, new ANNs are generated through AI's answer to mating, combining parts of parent ANNs while applying a dose of random mutation.

Training an ANN to play backgammon, a population of ANN 'players' is generated with random neurons. They are made to play against each other, taking turns to respond to inputs describing the board and the dice roll. Given their random constitution, this first generation of players will not be very good at the game, but some will be 'less bad' and win more games. The worst players are deleted and better players survive, with their features combined and mutated to produce a new generation of ANNs which join them in the next round of games. Some child ANNs will play better than their parents, others worse, but the environment is conducive to steady improvement.

The interesting thing about evolutionary methods is that they yield results without any strategic hints, without data to study, without even being told the rules. This means ANNs can develop interesting ways of playing, including strategies that humans might never have considered and may have trouble appreciating. An ANN's move can be explained as mathematically determined by its structure. This can, in turn, be explained as mathematically determined by the evolutionary environment. However, the implicit conceptualisation of the problem or logic of its solution is very difficult to explain, even for the engineers that design them. In this sense, the ANN's decision-making process is not transparent.

Evolutionary techniques can be applied to other problems, such as optimising computer programs or transport schedules. There are also other interesting AI approaches inspired by biological and behavioural mechanisms. For example, ant colony optimisation is modelled on how ants use pheromones as signals to find and highlight the quickest route between two locations, and can be used to optimise vehicle navigation and telecommunication networks. Hunting search is a search and optimisation technique based upon the pack hunting behaviour of lions, wolves and dolphins. 'Swarm intelligence' techniques inspired by the honey bee's dance (among other apian behaviours) have been applied in modelling and optimisation tasks in many engineering disciplines.

All about data: Data mining, big data and data in the wild

Since data is so central to contemporary AI development, several data-related concepts are frequently raised during debates about AI. AI engineers spend as much time thinking about data as algorithms. They need lots of good quality data to perform effective ML, and even more to test the results. 'Data mining', is a field of computation focused on the automated identification of patterns and anomalies in datasets. The dataset could be anything from text posted on social media to precise measurements of underground geological formations, and the mining process could deploy ANNs, statistics and modelling to identify useful features. 'Big data' refers to datasets that are so large and complex – including content from different sources, in different formats, and with different degrees of authenticity and accuracy – that they cannot be stored or processed in the same way as smaller datasets. This brings us to 'data in the wild', which usually refers to data that was produced for one purpose but remains somehow accessible and can be used for other purposes, perhaps outside the control of its original producer. So a research project might apply data mining techniques to social media platforms and blogs to research different individual's emotional and behavioural responses to news stories. Since this 'data in the wild' was not intended for research purposes, its use might be unreliable, unethical, or even illegal.

The art of artificial intelligence

It might be tempting to think of ML as doing all the hard work, but the algorithm can only follow the precise instructions set out by the AI engineer. First, the engineer needs to find a good way of encoding the problem itself. For the backgammon playing ANN, the engineer needs to express the game board and dice as a signal to be sent to the input layer. They also need to find a way of interpreting the output as a legitimate move – an instruction about which counters to move, and where to move them. They also need to work out a strategy for dealing with illegitimate moves, e.g. by designing the output layer so that its signal can always be interpreted as a legitimate move.

Artificial artificial intelligence?
Since AI is both difficult and marketable, some firms pretend to use AI, while hiring humans to act like really good AI agents.

If the ML algorithm uses training data, the AI engineer must consider which data to use and how. Where 'data in the wild' is used, they must ensure that it is legal and ethical. Even inadvertent storage and processing of some content – such as terrorist propaganda and child pornography – can be illegal. Other data might be subject to copyright, or require 'informed consent' from the owner before it is used for research or other purposes. If the data passes these tests, the engineer must determine whether it is sufficiently large and representative for the problem at hand. A dataset for learning to recognise cats should contain lots of pictures from different angles, different colours and breeds. Finally, they need to decide how much to use for training, and how much to set aside for testing. Where the training dataset is too small, the ANNs effectively memorise it without learning general rules and perform poorly when tested with new data.

The AI engineer also needs to make several important decisions about the structure of the ANN and the ML algorithm. With too few neurons and layers, the ANN will not be able to deal with the problem. Too many and they tend to memorise the training data instead of learning general rules. For gradient descent, the engineer defines how many evaluations to do before deciding on a direction to travel, as well as how far to travel in the chosen direction before re-evaluating. This is known as the 'learning rate'. If it is slower, it is as though the hiker takes their time to make better choices, if it is faster, it is as though the hiker thinks less and walks more. There is no right answer, and the engineer must decide how to balance speed against accuracy.

In evolutionary approaches, the AI engineer has to decide the population size and number of games played, balancing thorough appraisal against processing burden. They also need to decide how many ANNs to replace per generation, and how to replace them by combining and mutating elements of the parent ANNs. Mutation adds new 'genetic material' to the population, which is important for the emergence of new solutions, but if it is too drastic then the offspring might be so different from their parents that they perform as badly as the randomly generated ANNs from the first generation of the process.

A further question is raised in understanding when an optimum solution has been found. As discussed in the context of gradient descent, an algorithm can get stuck in a 'local optimum', which is the best solution they manage to find, but not the best possible solution available. Similarly, evolutionary populations can develop into a local optimum, whereby the parents cannot produce offspring that perform better than them, even though there are better solutions available. The engineer can counteract local optima by adjusting the learning rate in gradient descent, or altering the approach to reproduction and mutation in evolutionary methods – introducing more genetic material. They can also repeat the training process several times with different starting points and data sets. It is often worth the effort because, while ANNs are difficult to train, once a good solution is found they can be applied to new data very quickly.

Many of these decisions require the AI engineer to balance the constraints of the problem at hand, its context, and the data and processing resources available to them. There are no objectively correct formula, so they are part of the 'craft' of AI, with practitioners relying upon intuition, experience and shared wisdom to make effective decisions. Perhaps unsurprisingly, some AI engineers use ML to decide on some of these key variables. For example, 'neuroevolution' enables the ML algorithm to change the structure of the ANN, so mutation could alter the number of neurons and layers within the ANN or the population size. Nonetheless, the AI engineer still needs to make difficult decisions and give precise instructions about when and how much the ML can adjust each variable. These trends further reduce the AI engineers' direct influence upon how ANNs respond to problems, and might make it even more difficult for humans to make sense of the solutions they find. Nonetheless, the engineer retains an irreplaceable role in the design and optimisation of the environment in which machines learn.

Making sense of the world: Identifying language, images and sounds

Symbolic AI translation tools tried – and failed – to build huge sets of precise rules for translating text from one language to another. Early data-driven solutions bypassed these rules by making tools that look up clusters of words that have already been translated by experts, notably including text translated by the European Parliament. Now, top translation tools use these corpuses to train ANNs to translate text directly, without following expert rules or even looking up words.

Sound and vision are more complicated than text. Even as recently as 2005, during the author's training as an AI programmer, image and speech recognition were taught within the symbolic AI paradigm as problems of encoding human expertise. For example, to recognise cats and dogs in photographs, algorithms would search pixel-by-pixel for lines that look like edges and use them to identify general shapes which are compared to templates that correspond to cats and dogs. Likewise, speech recognition algorithms followed painstakingly hand-programmed instructions to identify the patterns produced in sound waves when people speak, before following rules that translated the patterns into words while taking into account different voices and accents.

These tools were difficult to develop, and struggled with real world examples. Expert-trained image and speech recognition systems were soon replaced by data-driven machine learning tools. Speech to text tools are now trained on labelled data, such as transcripts of audio files. Image recognition tools are trained on millions of pictures from the internet that are already labelled as containing cats or dogs. Indeed, Facebook improved its face recognition service precisely this way, by training it on millions of photos that were diligently labelled by its users. Google's image classifier has up to 30 hidden layers. The first layers search for lines they can identify as edges or corners, later layers try to identify shapes in these lines, and the last layers assemble these shapes to interpret the image.

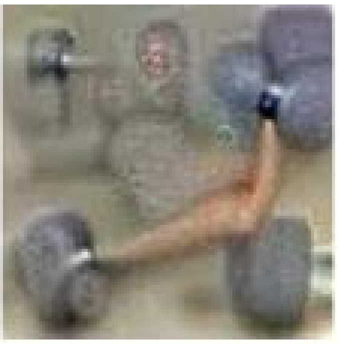
The tools are good at identifying individual patterns, and can make some useful sense of visual and linguistic input, but do they really understand the world? To answer this question, let's consider word vectors, which are commonly cited as AI's means of understanding the meaning of words. These vectors describe the 'proximity' of words to each other. Proximity is calculated by how often, statistically, a word is used in the same context as another. So, for example, the word 'barked' has a closer proximity to 'dog' than it does to 'cat'. The approach enables impressive linguistic manoeuvres such as 'king - man + woman = queen'. It is certainly a useful and well-designed technique, but the meaning of words and images cannot be reduced to statistical occurrences. It is important to note that the word vector does not know what it means to bark, only that the word 'bark' is more often used near the word 'dog' than 'cat'. It is a crucial distinction because, even if the results are practically useful, it might be dangerous to believe that AI really understands something when really it is just very good at behaving as though it does.

Imagination and creativity: Producing language, images and sounds

In one sense, all ML algorithms create their own ways of solving problems, even if they don't produce words or pictures. In another sense, since they can only follow precise instructions, computers have a serious lack of imagination. In fact, they cannot even spontaneously imagine the random numbers which they need to simulate dice rolls or generate random ANNs. The best they can do is follow precise instructions to generate a number that appears to be random. For example, the fifth decimal point in the measure of time between the two last mouse clicks is not a random number, but it may be random enough to work. Here, as with many features of AI, the semblance of understanding, imagination or creativity fades once the mechanisms are exposed as strictly procedural.

Visual recognition ANNs can be instructed to enhance the features of an image that resemble the specific object it was trained to recognise. By repeating this process, the ANN gradually reveals, in abstract and somewhat blurry images, its 'understanding' of what the object looks like, as shown in figure 3, an image generated by an ANN trained to recognise and dumbbells. This helped developers to realise that the ANN could not imagine a dumbbell without an arm attached to it, and were alerted to the possibility that the training data might not have contained enough labelled images of dumbbells on their own. However, the same approach has generated interesting results that some describe as kind of autonomous creativity. Perhaps these psychedelic images are the AI equivalent of human pareidolia, where humans identify patterns that they are trained to recognise – notably faces – in clouds and other objects.

Figure 3 – ANN produced image of a dumbbell, featuring arms



Source: Google Inc, CC by 4.0.

Many articles are now written by AI 'journalists'. They take basic information in a set format – perhaps weather predictions, sports results or stock market performance – and follow rules to identify the most important information and convert it into sentences that read like human written news articles. Here, the algorithm needs to ensure that the text maintains a consistent style while minimising repetition and other patterns that would seem too robotic to the reader. Once an article is written – by an AI or human journalist – AI subeditors can be deployed to generate headlines and test their attractiveness to users and search engines before settling on the final choice.

Basic language production is widely offered on smartphones. This includes predictive text to support typing, 'suggested replies' to bypass typing, and virtual assistants which interact entirely by voice. Predictive text is quite simple, linking each word to others that commonly follow them, much like the word vectors described above. These start with generic lists, but can be quickly customised to follow an individual user's style. Suggested replies are similar, but operate on whole messages rather than individual words. Virtual assistants are even more complex, as they start with the sound of the user's speech when they ask to play a song, send a message or buy a product. These sounds are be converted into phonetic units, then words, then instructions to be followed, perhaps via the user's diary, contacts, location and accounts. Any response or further question for the user needs to be translated into words in the selected language, phonetic units in the selected accent, and synthesised speech in the selected voice. Just as a good AI journalists avoid producing dull and repetitive text, a good voice assistant might try to convey emotions that are appropriate to the content, or add interjections like 'hmmm' while pretending to think about a response. All of this effort is made to make the agent appear to share human emotions. By using these services, data is collected which trains the AI to better identify users' voices, accents, emotions, interjections and, crucially, shopping habits.

Imagine an assistant calling a restaurant for their user, generating human-like speech to make a reservation. The restaurant's virtual assistant answers the call, and the two AI agents interact in spoken English, complete with human-like interjections and emotive intonations. They might even record the speech of the other agent, using this 'data in the wild' to train itself to speak in a more human way. This kind of interaction between ANNs is precisely how some newer ML algorithms known as generative adversarial nets (GANs) work. GANs can generate realistic images by training a 'detective ANN' to recognise whether a picture was produced by a human or a computer, then training a 'forger ANN' to produce images, which are tested by the detective ANN. In the process, the pair of ANNs both get better, the detective at identifying fake images and the forger at producing realistic images. The forger can be used to develop various image modification and production tools, for example to create aged versions of real faces, or to generate completely novel imagined faces. The same principles have been applied to train ANNs to produce realistic sounds, such as voice impersonation, or videos, adding (imagined) movements to photographs.

Such techniques can be used to generate extremely realistic AI-generated videos known as 'deepfakes', which have been used to produce fake pornographic videos featuring celebrities, and videos that appear to show politicians making statements. In one example, Barack Obama appears to highlighting the dangerous potential for misinformation presented by such deepfakes. Of course, the falsification of images is not new, but deepfakes can be incredibly realistic and are, perhaps, easier to tailor and mobilise for specific objectives.

Future waves: Towards artificial superintelligence?

In future, new approaches to AI could emerge that differ substantially from the symbolic and data-driven waves described so far. Three key concepts regularly emerge in discussions of future AI. First, artificial general intelligence (AGI), which refers to AI that is not limited to specific domains, but performs intelligently in a wide range of contexts and problem spaces. The second is artificial superintelligence (ASI), which refers to AI with higher levels of general intelligence than typical humans. The third is singularity which, in this context, refers to the moment where AI becomes intelligent and autonomous enough to generate even more intelligent and autonomous AI. The following sections explore some possible future development paths for AI technology that remain beyond today's capabilities but might lead to better AI, if not towards AGI, ASI or even singularity.

Self-explanatory and contextual artificial intelligence

It has been suggested that combining first and second wave AI could deliver completely new functionality in future waves, maintaining the complex power of ANNs while adding the human accessibility and explainability of expert systems. These future ANNs could combine ML with wider contextual knowledge about the world to give more accurate results with less training data. For example, a handwriting recognition system would train on images of written text while also drawing on their contextual knowledge of how people use hands and pens. Similarly, an animal recognition application could use contextual knowledge of how animals tend to move, or what they might look like from behind, to identify animals in positions that were not in the training data. The aims of such AI may appear modest, but neither their immense functional promise nor the huge technical barriers to their achievement should be underestimated.

Robotic artificial intelligence

If AI is the brains, then robotics is the brawn. Robots are a little like AI in that they can do some physical tasks with superhuman ability (such as lifting heavy weights) while failing spectacularly at manoeuvres that most humans find easy (such as walking up staircases). Combining AI and robotics could provide breakthroughs in both fields. For example, ML can be applied to manipulate physical objects with greater autonomy, flexibility and dexterity, which could make automated production and distribution more efficient. Aside from supply chains, the marriage of AI and robotics is a major area of development for military technologies, including autonomous weapons systems. At present, drones are remotely piloted by humans. However, that requires communication channels, which leave them vulnerable to detection and security threats. Furthermore, the need for human decision-makers makes them too slow to respond. Full AI command resolves these issues, while opening new opportunities such as swarming capabilities. Finally, it has been suggested that AI could control robots to build new computer hardware, which would enable even stronger AI which could build even better hardware. In these sense, a synergetic development cycle could lead to substantial autonomous improvements in both fields.

Quantum artificial intelligence

Single bits of data on normal computers exist in a single state, either 0 or 1. Single bits in a quantum computer, known as 'qubits' can exist in both states at the same time. As such, four qubits together can simultaneously exist in 16 different states (0000, 0001, 0010, etc.). Adding qubits lead to exponential increases (2^n) in the number of simultaneous states, so 50 qubits together can simultaneously exist in over a trillion different states. Quantum computers harness this simultaneity to quickly find solutions to very complex problems, promising a revolutionary increase in computing power. If the problem is to find a 1-in-a-trillion combination that works as a solution, a normal computer would have to check each possibility one by one while a quantum computer can check them all at the same time, in a single operation. This means they are particularly well-suited to problems such as finding and optimising solutions, and simulating environments. Since these kinds of problems are central to AI, quantum computing could enable significant advances in the field.

There have been some promising recent breakthroughs in quantum computing, but these also serve to illustrate how far the technology is from market. For example, IBM's 50 qubit machine broke industry records by remaining stable for 0.00009 seconds at its operating temperature of -273°C. Most agree that no reliable quantum computer will be built in the next decade or so. Some suggest it is a moving target that will always remain tantalisingly out of reach. For our purposes, quantum computing is a speculative development that, if achieved, could enable the emergence of future waves of AI either by applying current methods more effectively, or by enabling the development of completely new approaches.

Evolving artificial superintelligence

One suggested path to ASI is to develop increasingly sophisticated ANNs through better evolutionary methods, which run on more powerful computers. It took millions of years from the emergence of the first biological neurons to the evolution of intelligent humans. We might imagine a complex simulated evolutionary environment featuring many generations of different species of ANNan algorithm generating huge populations featuring several species of ANN in a complex simulated evolutionary environment. Current processing capabilities could not handle the level of complexity required to simulate anything resembling the evolutionary environment of the world in which human intelligence emerged. However, it might take some shortcuts by skipping the evolution of physical capabilities while helping the simulation out of 'evolutionary ruts'. Focusing only on the emergence of intelligence without the burdens associated with physical survival and reproduction, generations might pass more quickly, leading to the emergence of AI or even ASI.

Because of the demands of the environment in which they evolved, humans are good at recognising animals and understanding their movements, but bad at making quick and complex mathematical calculations. The kind of capabilities such a simulated population developed would depend on what kind of challenges they face, and what resources they can draw upon to develop solutions to them. Since they would not have human-like bodies, they would be unlikely to develop human-like languages or societies. Given this, they might never face the human-like problems needed to develop intelligent solutions to them. This does not mean they could not develop nuanced and surprising solutions to interesting problems, but would humans be able to relate to them in a useful or meaningful way? This points us back to a problem that AI has avoided since its first days, that is, how to define intelligence in machine terms. Following Harry Collins, intelligence seems to require immersion in human society which, in turn, requires a human body.

Brain emulation and artificial consciousness

If we could produce a very detailed digital copy of a human brain – including all of the neurons and their connections of various strengths – would the result be a complete digital emulation of the brain with the capacity to process sensory inputs, remember, learn, and apply general intelligence?

If so, perhaps the steps to developing an ASI would be as simple as speeding up the computer that it runs on, or ensuring it does not degrade with age, so that it has the time to carefully study every subject in every language. The e-brains might also be enhanced with on-brain modules allowing them to receive inputs from the light and sound spectrum that humans cannot usually sense, to perform advanced mathematical calculations, or to access the internet directly. Since the ANN would need to simulate around 86 billion neurons and about 150 trillion connections in real time, full brain emulation remains firmly in the domain of speculation. Nonetheless, there are serious projects pursuing the ambition, including the billion euro EU-funded Human Brain Project. Some progress has been made in mapping mouse brains, but they are incomplete and operate slower than real time. Furthermore, we don't understand whether such an emulated brain would need sleep, or how limited its ultimate capacity for memory or knowledge might be. For now, the prospect raises more questions than answers. Would it have consciousness, feel pain or sadness? Would deleting it be murder? Would owning it be slavery? Would modifying it be abuse?

Wetware and biological systems

The nascent field of artificial life (Alife) differs from AI in that its ideas and techniques are based upon fundamental biological processes, rather than intelligence or expertise. Nonetheless, it does have some crossover with AI, particularly in the context of evolutionary learning approaches and other methods inspired by nature. Much like AI, Alife is primarily developed via software (computer code and data) and hardware (physical components), but it can also involve 'wetware', which refers to the use of biological materials as components of the system being developed. Superhuman levels of intelligence could be achieved without wetware alone, through (natural or deliberate) biological selection, improved organisation of collective human intelligence or the use of cognition-enhancing drugs. AI insights might also benefit fields such as gene editing and synthetic biology, in which wetware takes centre stage on another speculative pathway towards bio-AI.

Concluding remarks

Since good AI could help us to develop even better AI, it can be tempting to see AGI, singularity and ASI as near inevitabilities. However, by learning how AI works, it becomes clear that the outward signs of understanding, creativity and empathy fall short against rigorous criteria. Today's AI is powerful and useful, but remains far from speculated AGI or ASI. Nobody knows the future of AI. Several potential avenues should be examined however, so that we can prepare for their possible impacts. In this context, it is worth debating the possibility of singularity and other speculated futures. However, such debates do little to help us in the more urgent task of responding to today's AI and the impacts that it already has on our daily lives. While reflections on current and future AI are related, it would be a mistake to conflate them entirely. Rather, for meaningful and constructive debates about AI, it is important to distinguish between different methods and their level of maturity.

DISCLAIMER AND COPYRIGHT

This document is prepared for, and addressed to, the Members and staff of the European Parliament as background material to assist them in their parliamentary work. The content of the document is the sole responsibility of its author(s) and any opinions expressed herein should not be taken to represent an official position of the Parliament.

stoa@ep.europa.eu (contact)

http://www.europarl.europa.eu/stoa/ (STOA website)

www.europarl.europa.eu/thinktank (internet)

http://epthinktank.eu (blog)