AI techniques & how AI help in increase in efficiency

1. Heuristics

Suppose we have coins with the following denominations: 5 cents, 4 cents, 3 cents, and 1 cent, and that we need to determine the minimum number of coins to create the amount of 7 cents. In order to solve this problem we can use a technique called “Heuristics”.

Webster1 defines the term Heuristic as “involving or serving as an aid to learning, discovery, or problem-solving by experimental and especially trial and error methods”. In practice, this means that whenever problems get too complex to find the guaranteed best possible solution using exact methods, Heuristics serves to employ a practical method for finding a solution that is not guaranteed to be optimal, but one that is sufficient for the immediate goals.

For some problems, tailored heuristics can be designed that exploits the structure within the problem. An example of such a tailored heuristic would be a greedy heuristic for the above mentioned coin-changing problem. We speak of a greedy heuristic when we always choose the largest denomination possible and repeat this until we get to the desired value of 7. In our example, that means that we would start with selecting a 5 cent coin. For the remaining 2 cents, the largest denomination we can choose is 1 cent, leaving us with the situation where we still have to cover 1 cent for which we again use 1 cent.

So our greedy heuristic gives us a solution of 3 coins (5, 1, 1) to get to the value of 7 cents. Of course another, better, solution of only 2 coins exists, using the 3 and 4 cent coins. While the greedy heuristic for the coin changing problem does not provide the best solution for this particular case, in most cases it will result in a solution that is acceptable.

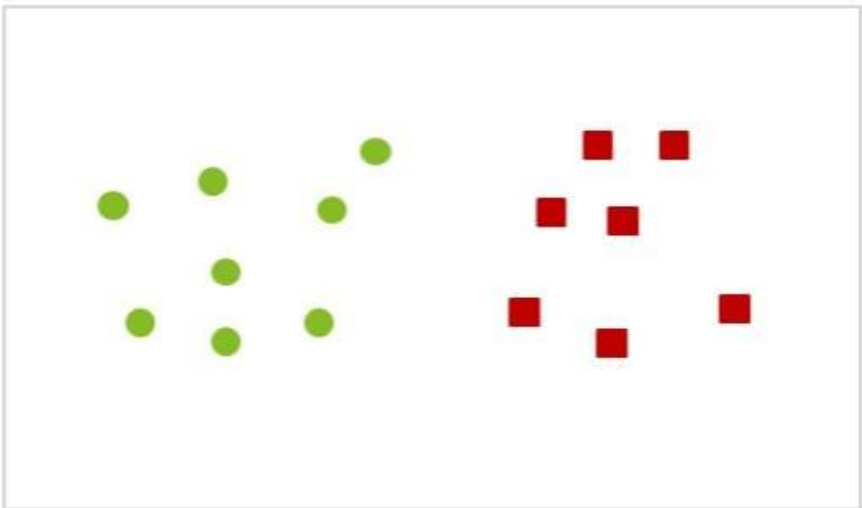
Besides such tailored Heuristics for specific problems, certain generic heuristics exist as well. Just like neural networks, some of these generic heuristics are based on processes in nature. Two examples of such generic heuristics are Ant Colony Optimization2 and genetic algorithms3. The first is based on how simple ants are able to work together to solve complex problems; the latter is based on the principle of survival of the fittest.

A typical problem where heuristics are applied to find acceptable solutions quickly is vehicle routing, where the objective is to find routes for one or more vehicles visiting a number of locations.

2. Support Vector Machines

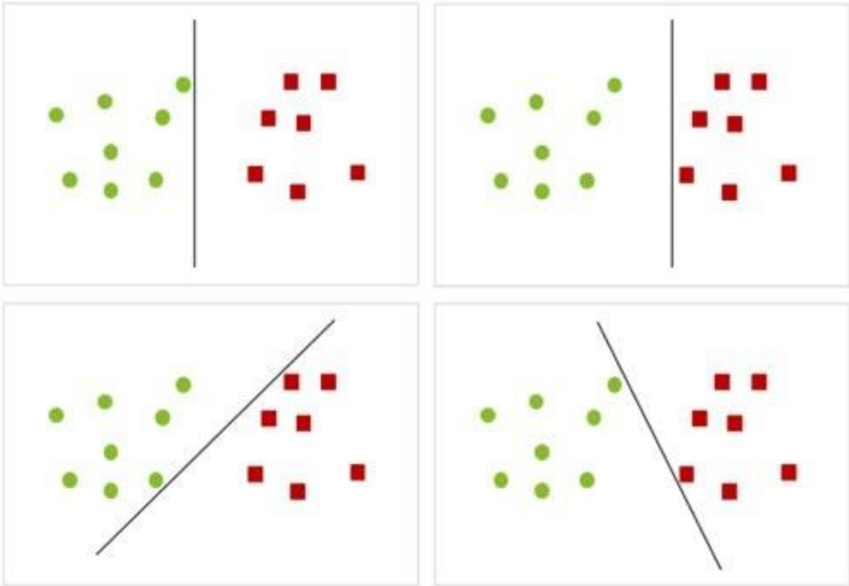
The question whether an email is spam or not is an example of a classification problem. In these types of problems, the objective is to determine whether a given data point belongs to a certain class or not. After first training a classifier model on data points for which the class is known (e.g. a set of emails that are labeled as spam or not spam), you can then use the model to determine the class of new, unseen data-points. A powerful technique for these types of problems is Support Vector Machines4 (SVM).

The main idea behind SVM is that you try to find the boundary line that separates the two classes, but in such a way that the boundary line creates a maximum separation between the classes. To demonstrate this, we will use the following simple data for our classification problem:

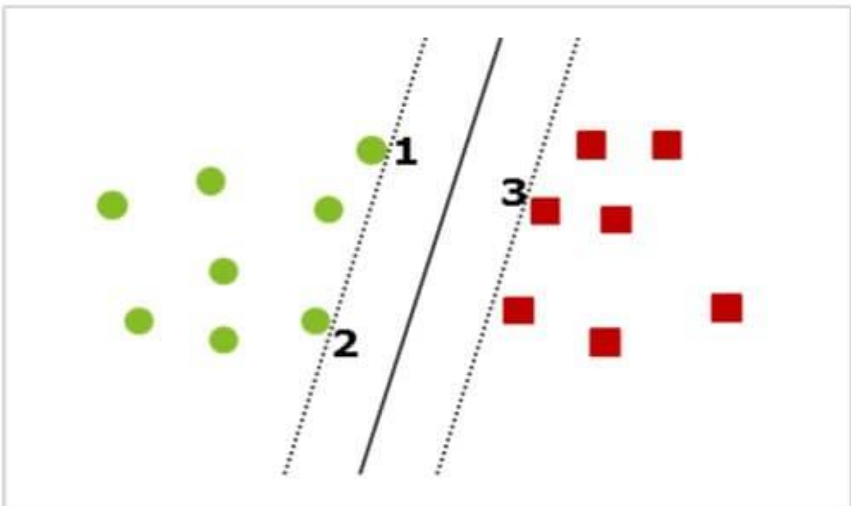


In this example, the green circles and the red squares could represent two different segments in a total set of customers (e.g. high potentials and low potentials), based on all kinds of properties for each of the customers.

Any line that keeps the green circles on the left and the red squares on the right is considered a valid boundary line for the classification problem. There is an infinite number of such lines that can be drawn. Four different examples are presented below:



As stated before, SVM helps you to find the boundary line that maximizes the separation between the two classes. In the provided example, this can be drawn as follows:



The two dotted lines are the two parallel separation lines with the largest space between them. The actual classification boundary that is used will be the solid line exactly in the middle of the two dotted lines.

The name Support Vector Machine comes from the data points that are directly on either of these lines. These are the supporting vectors. In our example, there were three supporting vectors.

If any of the other data points (i.e. not a supporting vector) is moved a bit, the dotted boundary lines are not affected. However, if the position of any of the supporting vectors is slightly changed (e.g. data point 1 is moved slightly to the left), the position of the dotted boundary lines will change and therefore the position of the solid classification line also changes.

In real life, data is not as straightforward as in this simplified example. We usually work with more than two dimensions. Besides having straight separation lines, the underlying mathematics for an SVM also allows for certain types of calculations or kernels that result in boundary lines that are non-linear.

SVM classification models can also be found in image recognition, e.g. face recognition, or when handwriting is converted to text.

3. Artificial Neural Networks

Animals are able to process (visual or other) information from their environment and adapt to change. They use their nervous system to perform such behavior. Their nervous system can be modeled and simulated and it should be possible to (re)produce similar behavior in artificial systems. Artificial Neural Networks (ANN) can be described as processing devices that are loosely modeled after the neural structure of a brain. The biggest difference between the two is that the ANN might have hundreds or thousands of neurons, whereas the neural structure of an animal or human brain has billions.

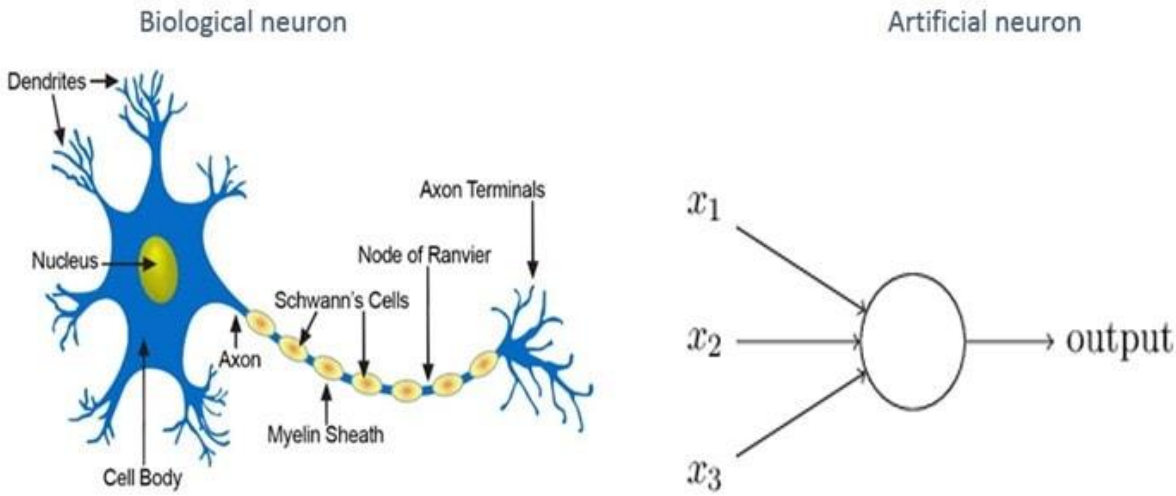


Figure 1: Graphical representation of a biological neuron (left) and an artificial neuron (right)

The basic principle of a neural structure is that each neuron is connected with a certain strength to other neurons. Based on the inputs taken from the output of other neurons (also considering the connection strength) an output is generated that can be used again as input by other neurons, see Figure 1 (left). This basic idea has been translated into an artificial neural network by using weights to indicate the strength of the connection between neurons. Furthermore, each neuron will take the output from the connected neurons as input and use a mathematical function to determine its output. This output is then used by other neurons again.

While learning consists of strengthening or weakening the bonds between different neurons in the biological brain, in the ANN learning consists of changing the weights between the neurons. By providing the neural network with a large set of training data with known features, the best weights between the artificial neurons (i.e. strength of the bond) can be calculated in order to make sure that the neural network best recognizes the features.

The neurons of the ANN can be structured into several layers[5]. Figure 2 shows an illustrative scheme of such layering. This network consists of an input layer, where all the inputs are received, processed and converted to outputs into the next layers. The hidden layers consist of one or more layers of neurons each passing through inputs and outputs. Finally, the output layer receives inputs of the last hidden layer and converts this into the output for the user.

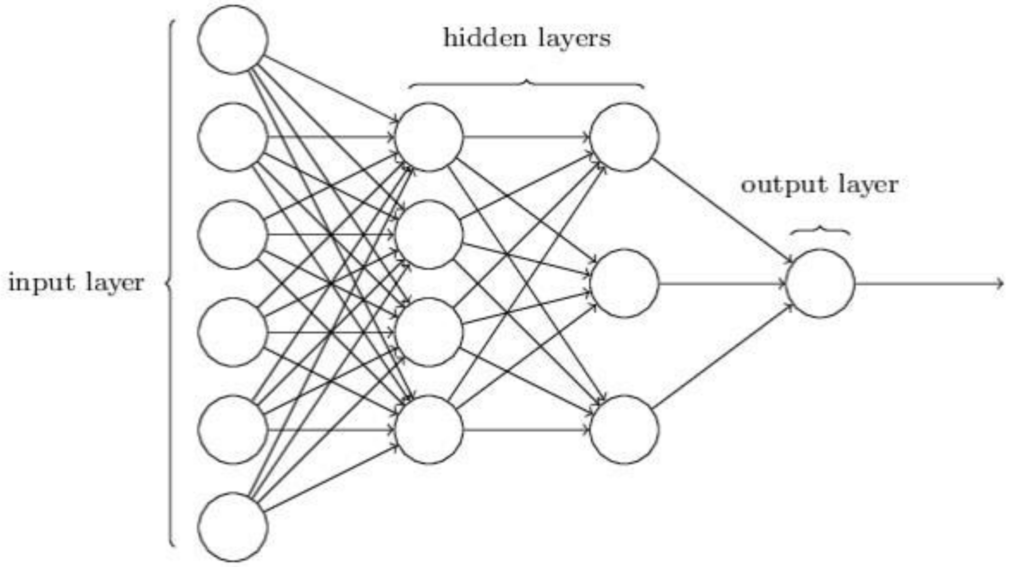


Figure 2: Schematic of a connected ANN

Figure 2 shows an example of a network in which all neurons in one layer are connected to all neurons in the next layer. Such a network is called fully connected. Depending on the kind of problem you want to solve, different connection patterns are available. For image recognition purposes, typically Convolutional networks are used, in which only groups of neurons from one layer are connected to groups of neurons in the next layer. For speech recognition purposes, typically Recurrent networks are used, that allow for loops from neurons in a later layer back to an earlier layer.

4. Markov Decision Process

A Markov Decision Process (MDP) is a framework for decision-making modeling where in some situations the outcome is partly random and partly based on the input of the decision maker. Another application where MDP is used is optimized planning. The basic goal of MDP is to find a policy for the decision maker, indicating what particular action should be taken at what state.
An MDP model consists of the following parts6:

- A set of possible states: for example, this can refer to a grid world of a robot or the states of a door (open or closed).
- A set of possible actions: a fixed set of actions that e.g. a robot can take, such as going north, left, south or west. Or with respect to a door, closing or opening it.

- Transition probabilities: this is the probability of going from one state to another. For example, what is the probability that the door is closed, after the action of closing the door has been performed?
- Rewards: these are used to direct the planning. For instance, a robot may want to move north to reach its destination. Actually going north will result in a higher reward.

Once the MDP has been defined, a policy can be trained using “Value Iteration” or “Policy Iteration”. These methods are used to calculate the expected rewards for each of the states. The policy then renders the best action that can be taken from each state.

As an example, we will define a grid that can be considered as an ideal, finite world for a robot7. This example grid is shown in Figure 3.

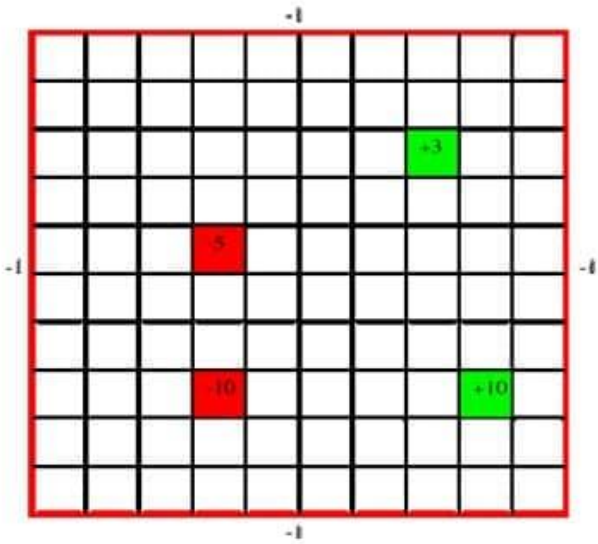


Figure 3: Example - grid world of a robot

The robot can move (action) from each position in the grid (state) in four directions, i.e. north, left, right and south. The probability that the robot goes into the desired direction is 0.7 and 0.1 if it moves towards any of the other 3 directions. A reward of -1 (i.e. a penalty) is given if the robot bumps into a wall and doesn’t move. Also, there are additional rewards and penalties if the robot reaches the cells that are colored green and red, respectively. Based on the probabilities and rewards a policy (function) can be made using the initial and final state.

Another example of MDP usage is the inventory planning problem - a stock keeper or manager has to decide how many units have to be ordered each week. The inventory planning can be modeled as an MDP, where the states can be considered as positive inventory and shortages. Possible actions are for instance ordering new units or backlogging to the next week. Transition probabilities can be considered as the action that will be taken based on the demand and inventory for the current week. Rewards - or in this case, costs - are typically unit order costs and inventory costs.

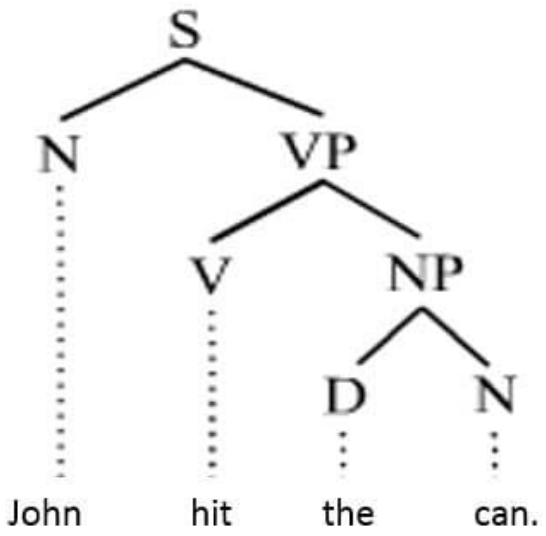
5. Natural Language Processing

Natural Language Processing (NLP) is used to refer to everything from speech recognition to language generation, each requiring different techniques. A few of the important techniques will be explained below, i.e. Part-of-Speech tagging, Named Entity Recognition, and Parsing.

Let us examine the sentence “John hit the can.” One of the first steps of NLP is lexical analysis, using a technique called Part-of-Speech (POS) tagging. With this technique every word is tagged to correspond to a category of words with similar grammatical properties, based on its relationship with adjacent and related words. Not only words are tagged, but also paragraphs and sentences. Part-of-speech tagging is mainly performed with statistical models, that lead to probabilistic results instead of hard if-then rules, and is therefore used for processing unknown text. Also, they can cope with the possibility of multiple possible answers, instead of only one. A technique that is often used for tagging is a Hidden Markov Model (HMM). An HMM is similar to the Markov Decision Process, where each state is a part of speech and the outcome of the process is the words of the sentence. HMMs ‘remember’ sequences of words that came before. Based on this, they can make better estimates of what Part-Of-Speech a word is. For example: ‘can’ in ‘the can’ is more likely to be a noun than a verb. The end result is that the words are tagged as followed: ‘John’ as a noun (N), ‘hit’ as a verb (V), ‘the’ as a determiner (D) and ‘can’ as a noun (N) as well.

Named Entity Recognition or NER, is similar to POS tagging. Instead of tagging words with the function of the word in the sentence (POS), words are tagged with the type of entity the word represents. These entities can be e.g. persons, companies, time, or location. But also more specialized entities such as gene, or protein. Although an HMM can also be used for NER, the technique of choice is a Recurrent Neural Network (RNN). An RNN is a different type of neural network as discussed earlier, but it takes sequences as input (a number of words in a sentence, or complete sentences), and remembers the output from the previous sentence8. In the sentence we are looking at, it will recognize John as the entity ‘person’.

A final technique to be discussed is called Parsing (Syntactic Analysis) - analyzing the grammar of the text and the way the words are arranged, so that the relationship between the words is clear. The Part-of-Speech tag from the lexical analysis is used and then grouped into small phrases, which in turn can also be combined with other phrases or words to make a slightly longer phrase. This is repeated until the goal is reached: every word in the sentence has been used. The rules of how the words can be grouped are called the grammar and can take a form like this: D+N = NP, which reads: a Determiner + Noun = Noun Phrase. The final result is depicted in the figure.



Conclusion

The techniques used within the domain of Artificial Intelligence are actually just advanced forms of statistical and mathematical models. All these models cleverly put together provide us with tools to compute tasks that were previously thought to be reserved for humans. In subsequent blogs we will dive deeper into business applications, some associated technology trends, and the top 5 risks and concern

How AI is Helping Efficiency Improve

AI is a wonder of modern science that has made a lot of things possible that were unthinkable before. Now thanks to AI many things can be done more quickly and more effectively. AI has increased the efficiency and productivity of many things in the industry. In this article, some of the examples of how AI has been helping in improving efficiency has been discussed.

To analyze data and reporting speed

AI can be very helpful in improving the data analyzing speed and also to increase the reporting time. The data are analyzed more accurately and the reporting time is also increased. AI can be used to analyze large amounts of data to draw conclusive reports. If the same work is given to any human the time required would be very high and also there would be a high possibility to do some kinds of error in the report.

24/7 service

Unlike humans, AI does not need to rest and they can provide their service 24 hours a day. Humans can work up to a minimum time and they can't be consistent in their performance all day long. On the other hand, AI is essentially a machine that does not get tired or bored. It can work all day giving the same performance and consistency throughout the whole day. This improves the efficiency of the business and make the industry runs smoother and more effectively. There is also lower chances of error if the job is done by an AI.

Automation of process

After the invention of AI, many of the complex processes have now been automated. There is no need for any human support to do those processes. Many of the microservices are automatically provided by the AI. A microservices example can be application deployment which used to be a very tedious and boring task for the developers and now it can be done easily with the help of an AI. Many other complex processes have been automated reducing the cost of the business industry and also reducing the effort of the employees.

Analytics

AI can be used widely in data analytics. AI is good at processing algorithms and use them to extract meaningful information from a very large sum of data. The business collects data on a large scale regularly so they can be analyzed to develop better strategies to get more customers. These data are very hard to be analyzed properly as they contain a huge amount of information. With the help of AI, these data can be processed quickly and a detailed report can also be generated quickly. This is very helpful in the business and increases the efficiency of the business industry as a whole.

Increase security

AI is very helpful in increasing the security of a network, an application or a website. A network, application or a website contains vital information and trade secrets of a business that is very essential for any business to protect. AI can be used to enhance protective measures in the network of the business. AI has revolutionized the cybersecurity industry making changes to the processes of cybersecurity and detecting abnormalities in the network. A system is more protected when AI technology is implemented to maintain the security of the network.

Providing better customer support

AI is now used to provide customer support to users that are visiting the website of the business. It is reducing the cost as the company needs to hire less customer service agents. But it is essential to understand the proper balance between a human and a machine. There are certainly some limitations to AI when it comes to providing customer support to the users. It is important to realize that in which parts of the customer service experience AI is to be implemented. If implemented properly AI can save a lot of costs for the whole business.