**INDUCTIVE LEARNING**

Machine learning is one of the most important subfields of artificial intelligence. It has been viewed as a viable way of avoiding the knowledge bottleneck problem in developing knowledge-based systems.

Inductive Learning, also known as Concept Learning, is how AI systems attempt to use a generalized rule to carry out observations.

To generate a set of classification rules, Inductive Learning Algorithms (APIs) are used. These generated rules are in the “If this then that” format.

These rules determine the state of an entity at each iteration step in Learning and how the Learning can be effectively changed by adding more rules to the existing ruleset.

When the output and examples of the function are fed into the AI system, inductive Learning attempts to learn the function for new data.

**The Fundamental Concept of Inductive Learning**

There are two methods for obtaining knowledge in the real world: first, from domain experts, and second, from machine learning.

Domain experts are not very useful or reliable for large amounts of data. As a result, for this project, we are adopting a machine learning approach.

The other method, using machine learning, replicates the logic of ‘experts’ in algorithms, but this work may be very complex, time-consuming, and expensive.

As a result, an option is the inductive algorithms, which generate a strategy for performing a task without requiring instruction at each step.

According to Jason Brownlee in his article “Basic Concepts in Machine Learning,” an excellent method to understand how Inductive Learning works is, for example, if we are given input samples (x) and output samples (f(x)) from the perspective of inductive Learning, and the problem is to estimate the function (f).

It is necessary then to generalize from the samples and the mapping so that it can be used to estimate the output for new samples in the future.

In practice, estimating the function is almost always too complicated, so we seek excellent approximations.

Some practical examples of induction are:

**Credit risk assessment**.

* The x is the property of the customer.
* The f(x) is credit approved or not.

**Disease diagnosis**.

* The x is the characteristics of a given patient.
* The f(x) is the patient’s disease.

**Face recognition**.

* The x are bitmaps of the faces we want to recognize.
* The f(x) is a name assigned to that face.

**Automatic steering (autonomous driving)**.

* The x is bitmap images from a camera in front of the car.
* The f(x) is the degree to which the steering wheel should be turned.

**Application**

There are some situations in which inductive Learning is not a good idea. It is critical to understand when and when not to use supervised machine learning.

Inductive Learning may be helpful in the following four situations:

* **Problems in which no human expertise is available**. People cannot write a program to solve a problem if they do not know the answer. These are areas ripe for exploration.
* **Humans can complete the task, but no one knows how to do it.** There are situations in which humans can do things that computers cannot or do not do well. Riding a bike or driving a car are two examples.
* **Problems where the desired function is frequently changing**. Humans could describe it and write a program to solve it, but the problem changes too frequently. It is not economical. The stock market is one example.
* **Problems where each user requires a unique function.** Writing a custom program for each user is not cost-effective. Consider Netflix or Amazon recommendations for movies or books.

**ROTE LEARNING:**

Rote learning technique avoids understanding the inner complexities but focuses on memorizing the material so that it can be recalled by the learner exactly the way it read or heard.

Learning by memorization: which avoids understanding the inner complexities the subject that is being learned.

Learning something from Repeating:saying the same thing and trying to remember how to say it;it does not help to understand ,it helps to remember ,like we learn a poem,song ,etc.

Rote Learning is basically *memorisation*.

* Saving knowledge so it can be used again.
* Retrieval is the only problem.
* No repeated computation, inference or query is necessary.

A simple example of rote learning is *caching*

* Store computed values (or large piece of data)
* Recall this information when required by computation.
* Significant time savings can be achieved.
* Many AI programs (as well as more general ones) have used caching very effectively.

Memorisation is a key necessity for learning:

* It is a basic necessity for any intelligent program -- is it a separate learning process?
* Memorisation can be a complex subject -- how best to store knowledge?

Rote learning is basically a simple process. However it does illustrate some issues that are relevant to more complex learning issues.

**Organisation**

-- access of the stored value must be faster than it would be to recompute it. Methods such as hashing, indexing and sorting can be employed to enable this.

*E.g* Samuel's program indexed board positions by noting the number of pieces.

**Generalisation**

-- The number of potentially stored objects can be very large. We may need to generalise some information to make the problem manageable.

*E.g* Samuel's program stored game positions only for white to move. Also rotations along diagonals are combined.

**Stability of the Environment**

-- Rote learning is not very effective in a rapidly changing environment. If the environment does change then we must detect and record exactly what has changed -- *the frame problem*.

**NATURAL LANGUAGE**

Natural language refers to the way we, humans, communicate with each other.

Namely, speech and text.

We are surrounded by text.

Think about how much text you see each day:

* Signs
* Menus
* Email
* SMS
* Web Pages
* *and so much more…*

The list is endless.

Now think about speech.

We may speak to each other, as a species, more than we write. It may even be easier to learn to speak than to write.

Voice and text are how we communicate with each other.

Given the importance of this type of data, we must have methods to understand and reason about natural language, just like we do for other types of data.

Symbolic AI involves the explicit embedding of human knowledge and behavior rules into computer programs. The practice showed a lot of promise in the early decades of AI research. But in recent years, as neural networks, also known as connectionist AI, gained traction, symbolic AI has fallen by the wayside.

**SYMBOL BASED LEARNING**

Symbols are things we use to represent other things. Symbols play a vital role in the human thought and reasoning process. If I tell you that I saw a cat up in a tree, your mind will quickly conjure an image.

We use symbols all the time to define things (cat, car, airplane, etc.) and people (teacher, police, salesperson). Symbols can represent abstract concepts (bank transaction) or things that don’t physically exist (web page, blog post, etc.). They can also describe actions (running) or states (inactive). Symbols can be organized into hierarchies (a car is made of doors, windows, tires, seats, etc.). They can also be used to describe other symbols (a cat with fluffy ears, a red carpet, etc.).

Being able to communicate in symbols is one of the main things that make us intelligent. Therefore, symbols have also played a crucial role in the creation of artificial intelligence.

The early pioneers of AI [believed](https://en.wikipedia.org/wiki/Dartmouth_workshop) that “every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it.” Therefore, symbolic AI took center stage and became the focus of research projects. Scientists developed tools to define and manipulate symbols.

Many of the concepts and tools you find in computer science are the results of these efforts. Symbolic AI programs are based on creating explicit structures and behavior rules.

An example of symbolic AI tools is object-oriented programming. OOP languages allow you to define classes, specify their properties, and organize them in hierarchies. You can create instances of these classes (called objects) and manipulate their properties. Class instances can also perform actions, also known as functions, methods, or procedures. Each method executes a series of rule-based instructions that might read and change the properties of the current and other objects.

Using OOP, you can create extensive and complex symbolic AI programs that perform various tasks.

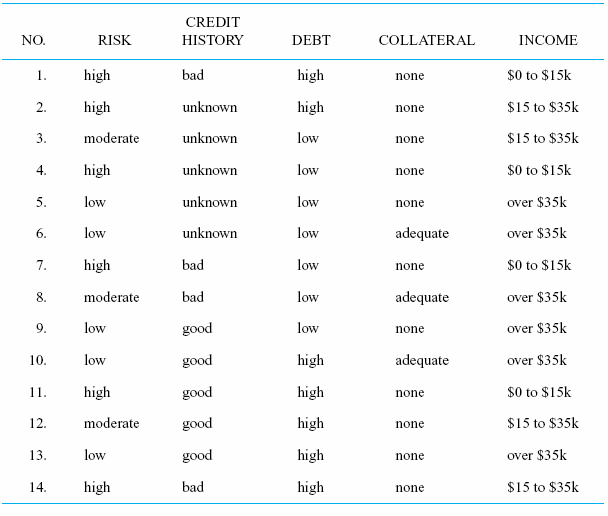
The benefits and limits of symbolic AI

Symbolic artificial intelligence showed early progress at the dawn of AI and computing. You can easily visualize the logic of rule-based programs, communicate them, and troubleshoot them.

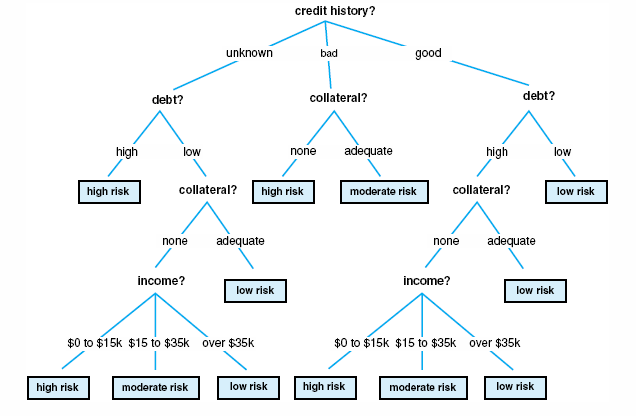
**DECISION TREE**

* + A tree structure to represent the concept, equivalent to a set of rules

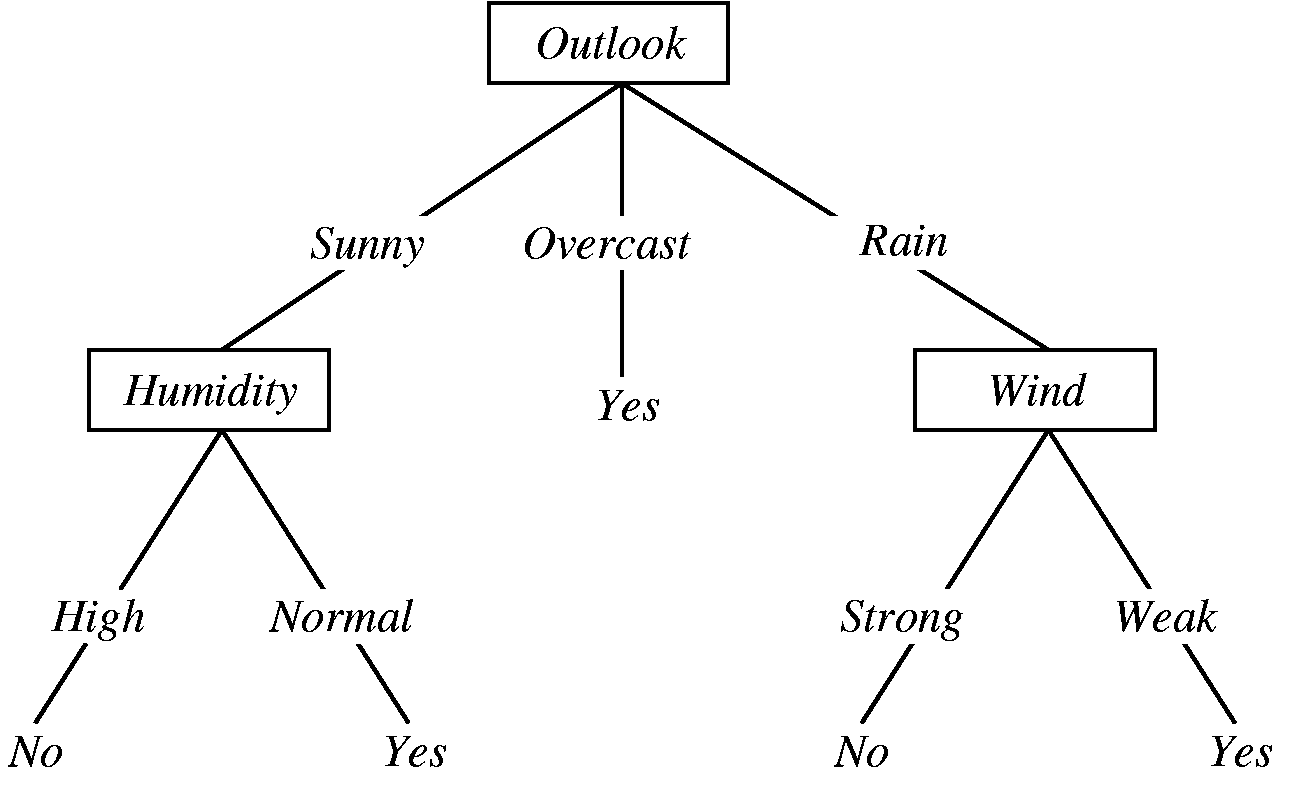
DATA



TREE



A decision tree is a flowchart-like structure in which each internal node represents a test on a feature (e.g. whether a coin flip comes up heads or tails) , each leaf node represents a class label (decision taken after computing all features) and branches represent conjunctions of features that lead to those class labels. The paths from root to leaf represent classification rules. Below diagram illustrate the basic flow of decision tree for decision making with labels (Rain(Yes), No Rain(No)).



Decision Tree for Rain Forecasting

Decision tree is one of the predictive modelling approaches used in statistics, data mining and machine learning.

Decision trees are constructed via an algorithmic approach that identifies ways to split a data set based on different conditions. It is one of the most widely used and practical methods for supervised learning. Decision Trees are a [non-parametric](https://machinelearningmastery.com/parametric-and-nonparametric-machine-learning-algorithms/) **supervised learning** method used for both **classification**and **regression**tasks.

**EXPLANATION BASED LEARNING**

Explanation based learning has ability to learn from a single training instance. Instead of taking more examples the explanation based learning is emphasized to learn a single, specific example. For example, consider the Ludoo game. In a Ludoo game, there are generally four colors of buttons. For a single color there are four different squares. Suppose the colors are red, green, blue and yellow. So maximum four members are possible for this game. Two members are considered for one side (suppose green and red) and other two are considered for another side (suppose blue and yellow). So for any one opponent the other will play his game. A square sized small box marked by symbols one to six is circulated among the four members. The number one is the lowest number and the number six is the highest for which all the operations are done. Always any one from the 1st side will try to attack any one member in the 2nd side and vice versa. At any instance of play the players of one side can attack towards the players of another side. Likewise, all the buttons may be attacked and rejected one by one and finally one side will win the game. Here at a time the players of one side can attack towards the players of another side. So for a specific player, the whole game may be affected. Hence we can say that always explanation based learning is concentrated on the inputs like a simple learning program, the idea about the goal state, the idea about the usable concepts and a set of rules that describes relationships between the objects and the actions.

**WHAT IS AN EXPERT SYSTEM?**

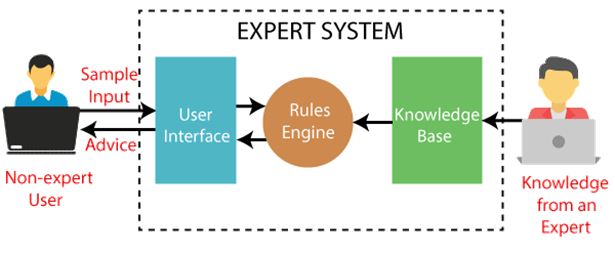
An expert system is a computer program that is designed to solve complex problems and to provide decision-making ability like a human expert. It performs this by extracting knowledge from its knowledge base using the reasoning and inference rules according to the user queries.

The expert system is a part of AI, and the first ES was developed in the year 1970, which was the first successful approach of artificial intelligence. It solves the most complex issue as an expert by extracting the knowledge stored in its knowledge base. The system helps in decision making for compsex problems using **both facts and heuristics like a human expert**. It is called so because it contains the expert knowledge of a specific domain and can solve any complex problem of that particular domain. These systems are designed for a specific domain, such as **medicine, science,** etc.

The performance of an expert system is based on the expert's knowledge stored in its knowledge base. The more knowledge stored in the KB, the more that system improves its performance. One of the co

mmon examples of an ES is a suggestion of spelling errors while typing in the Google search box.

Below is the block diagram that represents the working of an expert system:



Note: It is important to remember that an expert system is not used to replace the human experts; instead, it is used to assist the human in making a complex decision. These systems do not have human capabilities of thinking and work on the basis of the knowledge base of the particular domain.

**Below are some popular examples of the Expert System:**

* **DENDRAL:** It was an artificial intelligence project that was made as a chemical analysis expert system. It was used in organic chemistry to detect unknown organic molecules with the help of their mass spectra and knowledge base of chemistry.
* **MYCIN:** It was one of the earliest backward chaining expert systems that was designed to find the bacteria causing infections like bacteraemia and meningitis. It was also used for the recommendation of antibiotics and the diagnosis of blood clotting diseases

.

* **PXDES:** It is an expert system that is used to determine the type and level of lung cancer. To determine the disease, it takes a picture from the upper body, which looks like the shadow. This shadow identifies the type and degree of harm.
* **CaDeT:** The CaDet expert system is a diagnostic support system that can detect cancer at early stages.

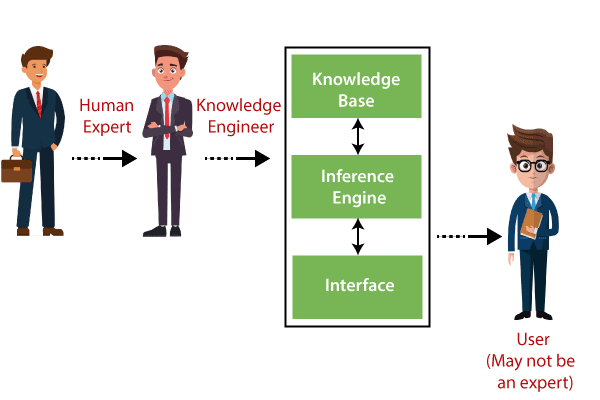
**Characteristics of Expert System**

* **High Performance:** The expert system provides high performance for solving any type of complex problem of a specific domain with high efficiency and accuracy.
* **Understandable:** It responds in a way that can be easily understandable by the user. It can take input in human language and provides the output in the same way.
* **Reliable:** It is much reliable for generating an efficient and accurate output.
* **Highly responsive:** ES provides the result for any complex query within a very short period of time.

Components of Expert System

An expert system mainly consists of three components:

* **User Interface**
* **Inference Engine**
* **Knowledge Base**



1. User Interface

With the help of a user interface, the expert system interacts with the user, takes queries as an input in a readable format, and passes it to the inference engine. After getting the response from the inference engine, it displays the output to the user. In other words, **it is an interface that helps a non-expert user to communicate with the expert system to find a solution**.

2. Inference Engine(Rules of Engine)

* The inference engine is known as the brain of the expert system as it is the main processing unit of the system. It applies inference rules to the knowledge base to derive a conclusion or deduce new information. It helps in deriving an error-free solution of queries asked by the user.
* With the help of an inference engine, the system extracts the knowledge from the knowledge base.
* There are two types of inference engine:
* **Deterministic Inference engine:** The conclusions drawn from this type of inference engine are assumed to be true. It is based on **facts** and **rules**.
* **Probabilistic Inference engine:** This type of inference engine contains uncertainty in conclusions, and based on the probability.

Inference engine uses the below modes to derive the solutions:

* **Forward Chaining:** It starts from the known facts and rules, and applies the inference rules to add their conclusion to the known facts.
* **Backward Chaining:** It is a backward reasoning method that starts from the goal and works backward to prove the known facts.

3. Knowledge Base

* The knowledgebase is a type of storage that stores knowledge acquired from the different experts of the particular domain. It is considered as big storage of knowledge. The more the knowledge base, the more precise will be the Expert System.
* It is similar to a database that contains information and rules of a particular domain or subject.
* One can also view the knowledge base as collections of objects and their attributes. Such as a Lion is an object and its attributes are it is a mammal, it is not a domestic animal, etc.

**Components of Knowledge Base**

* **Factual Knowledge:** The knowledge which is based on facts and accepted by knowledge engineers comes under factual knowledge.
* **Heuristic Knowledge:** This knowledge is based on practice, the ability to guess, evaluation, and experiences.

**Knowledge Representation:** It is used to formalize the knowledge stored in the knowledge base using the If-else rules.

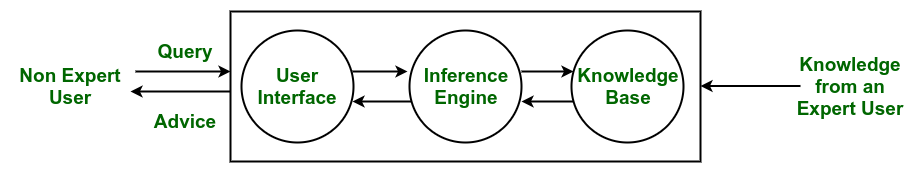
**Knowledge Acquisitions:** It is the process of extracting, organizing, and structuring the domain knowledge, specifying the rules to acquire the knowledge from various experts, and store that knowledge into the knowledge base.

Development of Expert System

Here, we will explain the working of an expert system by taking an example of MYCIN ES. Below are some steps to build an MYCIN:

* Firstly, ES should be fed with expert knowledge. In the case of MYCIN, human experts specialized in the medical field of bacterial infection, provide information about the causes, symptoms, and other knowledge in that domain.
* The KB of the MYCIN is updated successfully. In order to test it, the doctor provides a new problem to it. The problem is to identify the presence of the bacteria by inputting the details of a patient, including the symptoms, current condition, and medical history.
* The ES will need a questionnaire to be filled by the patient to know the general information about the patient, such as gender, age, etc.
* Now the system has collected all the information, so it will find the solution for the problem by applying if-then rules using the inference engine and using the facts stored within the KB.
* In the end, it will provide a response to the patient by using the user interface.

**Components of an Expert System :**



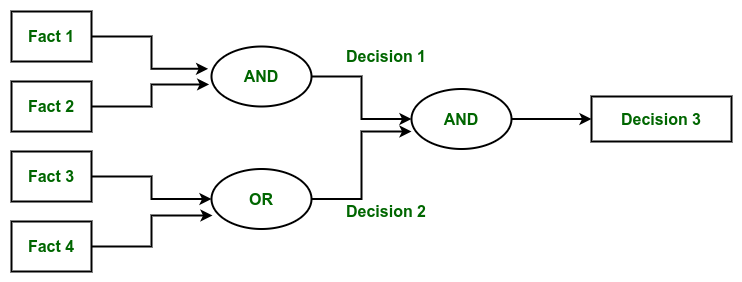
*Architecture of an Expert System*

* **Knowledge Base –**  
  The knowledge base represents facts and rules. It consists of knowledge in a particular domain as well as rules to solve a problem, procedures and intrinsic data relevant to the domain.
* **Inference Engine –**  
  The function of the inference engine is to fetch the relevant knowledge from the knowledge base, interpret it and to find a solution relevant to the user’s problem. The inference engine acquires the rules from its knowledge base and applies them to the known facts to infer new facts. Inference engines can also include an explanation and debugging abilities.
* **Knowledge Acquisition and Learning Module –**  
  The function of this component is to allow the expert system to acquire more and more knowledge from various sources and store it in the knowledge base.
* **User Interface –**  
  This module makes it possible for a non-expert user to interact with the expert system and find a solution to the problem.
* **Explanation Module –**  
  This module helps the expert system to give the user an explanation about how the expert system reached a particular conclusion.

The Inference Engine generally uses two strategies for acquiring knowledge from the Knowledge Base, namely –

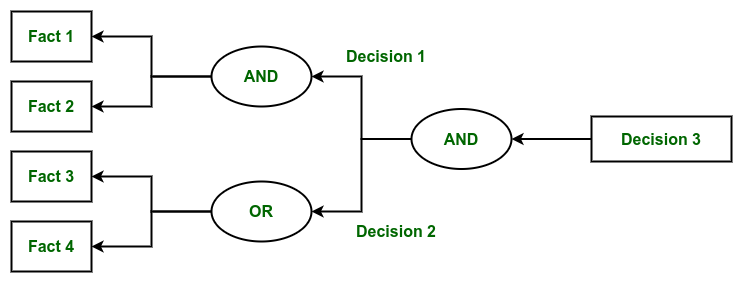
* Forward Chaining
* Backward Chaining

**Forward Chaining –**  
Forward Chaining is a strategic process used by the Expert System to answer the questions – What will happen next. This strategy is mostly used for managing tasks like creating a conclusion, result or effect. Example – prediction or share market movement status.



*Forward Chaining*

**Backward Chaining –**  
Backward Chaining is a storage used by the Expert System to answer the questions – Why this has happened. This strategy is mostly used to find out the root cause or reason behind it, considering what has already happened. Example – diagnosis of stomach pain, blood cancer or dengue, etc.  



*Backward Chaining*

**Characteristics of an Expert System :**

* Human experts are perishable, but an expert system is permanent.
* It helps to distribute the expertise of a human.
* One expert system may contain knowledge from more than one human experts thus making the solutions more efficient.
* It decreases the cost of consulting an expert for various domains such as medical diagnosis.
* They use a knowledge base and inference engine.
* Expert systems can solve complex problems by deducing new facts through existing facts of knowledge, represented mostly as if-then rules rather than through conventional procedural code.
* Expert systems were among the first truly successful forms of artificial intelligence (AI) software.

**Limitations :**

* Do not have human-like decision-making power.
* Cannot possess human capabilities.
* Cannot produce correct result from less amount of knowledge.
* Requires excessive training.

**Advantages :**

* Low accessibility cost.
* Fast response.
* Not affected by emotions, unlike humans.
* Low error rate.
* Capable of explaining how they reached a solution.

**Disadvantages :**

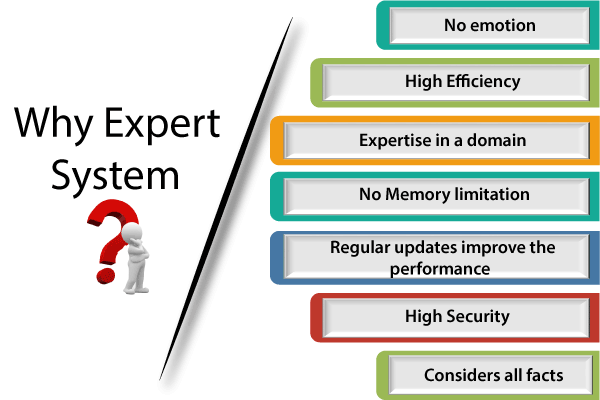
* The expert system has no emotions.
* Common sense is the main issue of the expert system.
* It is developed for a specific domain.
* It needs to be updated manually. It does not learn itself.
* Not capable to explain the logic behind the decision.

**Participants in the development of Expert System**

There are three primary participants in the building of Expert System:

1. **Expert:** The success of an ES much depends on the knowledge provided by human experts. These experts are those persons who are specialized in that specific domain.
2. **Knowledge Engineer:** Knowledge engineer is the person who gathers the knowledge from the domain experts and then codifies that knowledge to the system according to the formalism.
3. **End-User:** This is a particular person or a group of people who may not be experts, and working on the expert system needs the solution or advice for his queries, which are complex.

Why Expert System?



Before using any technology, we must have an idea about why to use that technology and hence the same for the ES. Although we have human experts in every field, then what is the need to develop a computer-based system. So below are the points that are describing the need of the ES:

1. **No memory Limitations:** It can store as much data as required and can memorize it at the time of its application. But for human experts, there are some limitations to memorize all things at every time.
2. **High Efficiency:** If the knowledge base is updated with the correct knowledge, then it provides a highly efficient output, which may not be possible for a human.
3. **Expertise in a domain:** There are lots of human experts in each domain, and they all have different skills, different experiences, and different skills, so it is not easy to get a final output for the query. But if we put the knowledge gained from human experts into the expert system, then it provides an efficient output by mixing all the facts and knowledge
4. **Not affected by emotions:** These systems are not affected by human emotions such as fatigue, anger, depression, anxiety, etc.. Hence the performance remains constant.
5. **High security:** These systems provide high security to resolve any query.
6. **Considers all the facts:** To respond to any query, it checks and considers all the available facts and provides the result accordingly. But it is possible that a human expert may not consider some facts due to any reason.
7. **Regular updates improve the performance:** If there is an issue in the result provided by the expert systems, we can improve the performance of the system by updating the knowledge base.

Capabilities of the Expert System

Below are some capabilities of an Expert System:

* **Advising:** It is capable of advising the human being for the query of any domain from the particular ES.
* **Provide decision-making capabilities:** It provides the capability of decision making in any domain, such as for making any financial decision, decisions in medical science, etc.
* **Demonstrate a device:** It is capable of demonstrating any new products such as its features, specifications, how to use that product, etc.
* **Problem-solving:** It has problem-solving capabilities.
* **Explaining a problem:** It is also capable of providing a detailed description of an input problem.
* **Interpreting the input:** It is capable of interpreting the input given by the user.
* **Predicting results:** It can be used for the prediction of a result.
* **Diagnosis:** An ES designed for the medical field is capable of diagnosing a disease without using multiple components as it already contains various inbuilt medical tools.

Advantages of Expert System

* These systems are highly reproducible.
* They can be used for risky places where the human presence is not safe.
* Error possibilities are less if the KB contains correct knowledge.
* The performance of these systems remains steady as it is not affected by emotions, tension, or fatigue.
* They provide a very high speed to respond to a particular query

.

Limitations of Expert System

* The response of the expert system may get wrong if the knowledge base contains the wrong information.
* Like a human being, it cannot produce a creative output for different scenarios

.

* Its maintenance and development costs are very high

.

* Knowledge acquisition for designing is much difficult.
* For each domain, we require a specific ES, which is one of the big limitations.
* It cannot learn from itself and hence requires manual updates

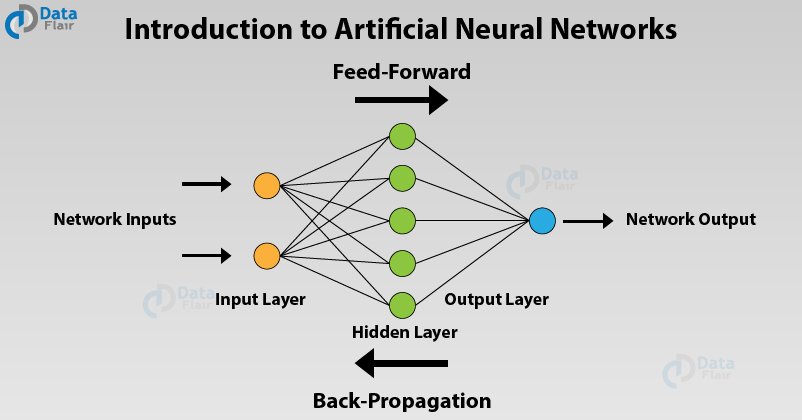
.

Applications of Expert System

* **In designing and manufacturing domain**  
  It can be broadly used for designing and manufacturing physical devices such as camera lenses and automobiles.
* **In the knowledge domain**  
  These systems are primarily used for publishing the relevant knowledge to the users. The two popular ES used for this domain is an advisor and a tax advisor.
* **In the finance domain**  
  In the finance industries, it is used to detect any type of possible fraud, suspicious activity, and advise bankers that if they should provide loans for business or not.
* **In the diagnosis and troubleshooting of devices**  
  In medical diagnosis, the ES system is used, and it was the first area where these systems were used.
* **Planning and Scheduling**  
  The expert systems can also be used for planning and scheduling some particular tasks for achieving the goal of that task.

**INTRODUCTION TO ARTIFICIAL NEURAL NETWORKS**

Artificial Neural Networks are the most popular [machine learning algorithms](https://data-flair.training/blogs/machine-learning-algorithm/) today. The invention of these Neural Networks took place in the 1970s but they have achieved huge popularity due to the recent increase in computation power because of which they are now virtually everywhere. In every application that you use, Neural Networks power the intelligent interface that keeps you engaged.

[](https://data-flair.training/blogs/wp-content/uploads/sites/2/2019/07/Introduction-to-Artificial-Neural-Networks.jpg)

What is ANN?

Artificial Neural Networks are a special type of machine learning algorithms that are modeled after the human brain. That is, just like how the neurons in our nervous system are able to learn from the past data, similarly, the ANN is able to learn from the data and provide responses in the form of predictions or classifications.

ANNs are nonlinear statistical models which display a complex relationship between the inputs and outputs to discover a new pattern. A variety of tasks such as image recognition, speech recognition, machine translation as well as medical diagnosis makes use of these artificial neural networks.

An important advantage of ANN is the fact that it learns from the example data sets. Most commonly usage of ANN is that of a random function approximation. With these types of tools, one can have a cost-effective method of arriving at the solutions that define the distribution. ANN is also capable of taking sample data rather than the entire dataset to provide the output result. With ANNs, one can enhance existing data analysis techniques owing to their advanced predictive capabilities.

Artificial Neural Networks Architecture

The functioning of the Artificial Neural Networks is similar to the way neurons work in our nervous system. The Neural Networks go back to the early **1970s** when **Warren S McCulloch and Walter Pitts** coined this term. In order to understand the workings of ANNs, let us first understand how it is structured. In a neural network, there are three essential layers –

Input Layers

The*input layer* is the first layer of an ANN that receives the input information in the form of various texts, numbers, audio files, image pixels, etc.

Hidden Layers

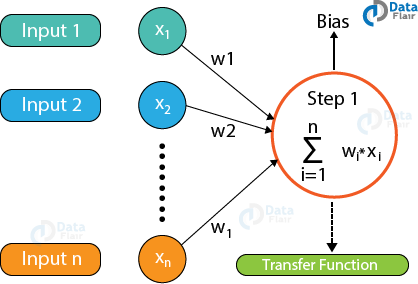
In the middle of the ANN model are the*hidden layers*. There can be a single hidden layer, as in the case of a perceptron or multiple hidden layers. These hidden layers perform various types of mathematical computation on the input data and recognize the patterns that are part of.

Output Layer

In the *output layer*, we obtain the result that we obtain through rigorous computations performed by the middle layer.

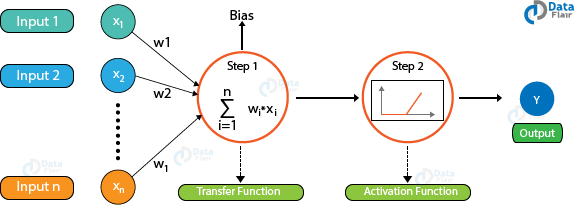
In a neural network, there are multiple parameters and hyperparameters that affect the performance of the model. The output of ANNs is mostly dependent on these parameters. Some of these parameters are weights, biases, learning rate, batch size etc. Each node in the ANN has some weight.

Each node in the network has some weights assigned to it. A transfer function is used for calculating the weighted sum of the inputs and the bias.

[](https://data-flair.training/blogs/wp-content/uploads/sites/2/2019/08/how-ANN-works.png)

After the transfer function has calculated the sum, the activation function obtains the result. Based on the output received, the activation functions fire the appropriate result from the node. **For example**, if the output received is above 0.5, the activation function fires a 1 otherwise it remains 0.

Some of the popular activation functions used in Artificial Neural Networks are Sigmoid, RELU, Softmax, tanh etc.

[](https://data-flair.training/blogs/wp-content/uploads/sites/2/2019/08/How-artificial-Neural-Networks-work.png)

Based on the value that the node has fired, we obtain the final output. Then, using the error functions, we calculate the discrepancies between the predicted output and resulting output and adjust the weights of the neural network through a process known as***backpropagation***.

ANNs are part of an emerging area in Machine Learning known as Deep Learning.

Back Propagation in Artificial Neural Networks

In order to train a neural network, we provide it with examples of input-output mappings. Finally, when the neural network completes the training, we test the neural network where we do not provide it with these mappings. The neural network predicts the output and we evaluate how correct the output is using the various error functions. Finally, based on the result, the model adjusts the weights of the neural networks to optimize the network following gradient descent through the chain rule.

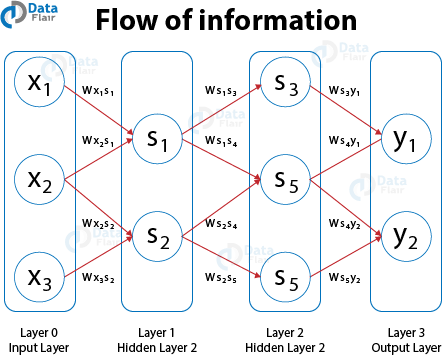
Types of Artificial Neural Networks

There are two important types of Artificial Neural Networks –

* FeedForward Neural Network
* FeedBack Neural Network

FeedForward Artificial Neural Networks

In the feedforward ANNs, the flow of information takes place only in one direction. That is, the flow of information is from the input layer to the hidden layer and finally to the output. There are no feedback loops present in this neural network. These type of neural networks are mostly used in ***supervised learning***for instances such as classification, image recognition etc. We use them in cases where the data is not sequential in nature.

[](https://data-flair.training/blogs/wp-content/uploads/sites/2/2019/08/feedforward-artificial-neural-networks.png)

Feedback Artificial Neural Networks

In the feedback ANNs, the feedback loops are a part of it. Such type of neural networks are mainly for memory retention such as in the case of recurrent neural networks. These types of networks are most suited for areas where the data is sequential or time-dependent.

Bayesian Networks

These type of neural networks have a probabilistic graphical model that makes use of Bayesian Inference for computing the probability. These type of Bayesian Networks are also known as Belief Networks. In these Bayesian Networks, there are edges that connect the nodes representing the probabilistic dependencies present among these type of random variables. The direction of effect is such that if one node is affecting the other then they fall in the same line of effect. Probability associated with each node quantifies the strength of the relationship. Based on the relationship, one is able to infer from the random variables in the graph with the help of various factors.

The only constraint that these networks have to follow is it cannot return to the node through the directed arcs. Therefore, Bayesian Networks are referred to as Directed Acyclic Graphs ([DAGs](https://en.wikipedia.org/wiki/Directed_acyclic_graph)).

These [**Bayesian Networks**](https://data-flair.training/blogs/bayesian-network-introduction/) can handle the multivalued variables and they comprise of two dimensions –

* Range of Prepositions
* Probability that each preposition has been assigned with.

Assume that there is a finite set of random variables such that each variable of the finite set is denoted by X = {x1, x2… xn} where each variable X takes from the values present in the finite set such that Value{x1}. If there is a directed link from the variable Xi to the variable Xj, then Xi will be the parent of Xj that shows the direct dependencies between these variables.

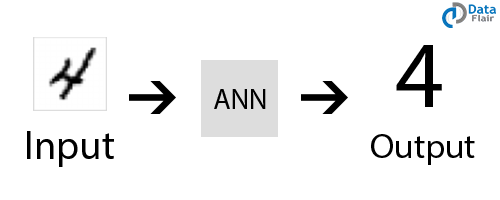
With the help of Bayesian Networks, one can combine the prior knowledge as well as the observed data. Bayesian Networks are mainly for learning the causal relationships and also understanding the domain knowledge to predict the future event. This takes place even in the case of missing data.

Artificial Neural Networks Applications

Following are the important Artificial Neural Networks applications –

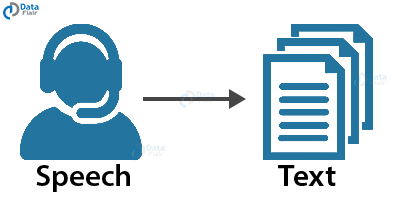
Handwritten Character Recognition

ANNs are used for handwritten character recognition. Neural Networks are trained to recognize the handwritten characters which can be in the form of letters or digits.

[](https://data-flair.training/blogs/wp-content/uploads/sites/2/2019/07/handwritten-character-recognition-Anns-applications.png)

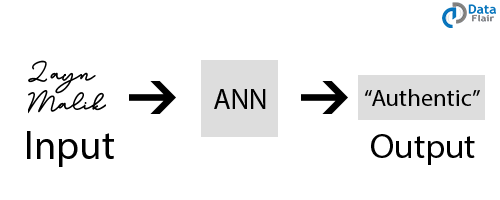
Speech Recognition

ANNs play an important role in speech recognition. The earlier models of Speech Recognition were based on statistical models like Hidden Markov Models. With the advent of[**deep learning**](https://data-flair.training/blogs/deep-learning/), various types of neural networks are the absolute choice for obtaining an accurate classification.

[](https://data-flair.training/blogs/wp-content/uploads/sites/2/2019/07/ML-neural-networks-speech-recognition.png)

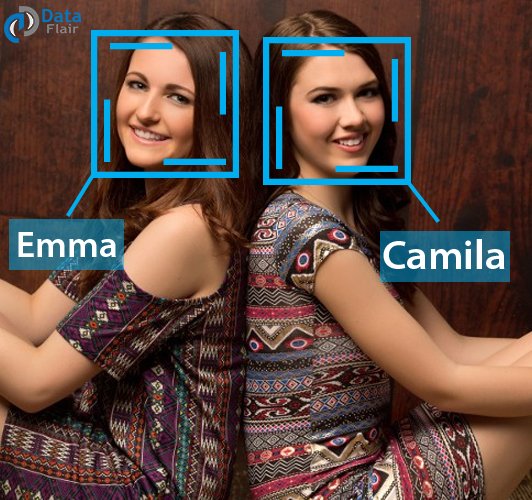
Signature Classification

For recognizing signatures and categorizing them to the person’s class, we use artificial neural networks for building these systems for authentication. Furthermore, neural networks can also classify if the signature is fake or not.

[](https://data-flair.training/blogs/wp-content/uploads/sites/2/2019/07/ann-application-signature-classification.png)

Facial Recognition

In order to recognize the faces based on the identity of the person, we make use of neural networks. They are most commonly used in areas where the users require security access. Convolutional Neural Networks are the most popular type of ANN used in this field.

[](https://data-flair.training/blogs/wp-content/uploads/sites/2/2019/07/image-recognition-in-neural-networks.jpg)

Artificial Neural Network (ANN) is a type of neural network which is based on a Feed-Forward strategy. It is called this because they pass information through the nodes continuously till it reaches the output node. This is also known as the simplest type of neural network.

**Some advantages of ANN :**

* Ability to learn irrespective of the type of data (Linear or Non-Linear).
* ANN is highly volatile and serves best in financial time series forecasting.

**Some disadvantages of ANN :**

* The simplest architecture makes it difficult to explain the behavior of the network.
* This network is dependent on hardware.

**2. Biological Neural Network :**  
Biological Neural Network (BNN) is a structure that consists of Synapse, dendrites, cell body, and axon. In this neural network, the processing is carried out by neurons. Dendrites receive signals from other neurons, Soma sums all the incoming signals and axon transmits the signals to other cells.

**Some advantages of BNN :**

* The synapses are the input processing element.
* It is able to process highly complex parallel inputs.

**Some disadvantages of BNN :**

* There is no controlling mechanism.
* Speed of processing is slow being it complex.

**Differences between ANN and BNN :**

| S.No. | ANN | BNN |
| --- | --- | --- |
| 1. | It is short for Artificial Neural Network. | It is short for Biological Neural Network. |
| 2. | Processing speed is fast as compared to Biological Neural Network. | They are slow in processing information. |
| 3. | Allocation for Storage to a new process is strictly irreplaceable as the old location is saved for the previous process. | Allocation for storage to a new process is easy as it is added just by adjusting the interconnection strengths. |
| 4. | Processes operate in sequential mode. | The process can operate in massive parallel operations. |
| 5. | If any information gets corrupted in the memory it cannot be retrieved. | Information is distributed into the network throughout into sub-nodes, even if it gets corrupted it can be retrieved. |
| 6. | The activities are continuously monitored by a control unit. | There is no control unit to monitor the information being processed into the network. |