

A Comprehensive Analysis of Classical Machine Learning and Modern Deep Learning Methodologies

Nisha Bhadauriya Agarwal^[0009-0003-2422-7250]
Ph.D Scholar, Sage University, Indore, India

Dr. Deepak Kumar Yadav^[0009-0000-1226-487x]
Associate Professor, Sage University, Indore, India

Abstract. Over the past decade, artificial intelligence (AI) has become a popular subject both within and outside of the scientific community; an abundance of articles in technology and non-technology-based journals have covered the topics of Machine Learning, Deep Learning, and Artificial Intelligence. Artificial Intelligence has started to become the mainstay of a number of applications online and in the market worldwide. While AI takes a front seat, Classical Machine Learning algorithms have been around for nearly five decades and continue to be the bedrock of future development and research in the field of machine learning. Besides this, deep learning is the current and a stimulating field of machine learning. Yet there still remains confusion around AI, ML, and DL. Despite their strong associations, the names cannot

be used interchangeably. In order to better communicate these concepts to a clinical audience, we (try to) avoid technical jargon in our review study. The purpose of the paper is to familiarize the reader with the various machine learning and deep learning approaches as well as the various kinds of algorithms that are the foundation of the machine learning field.

Keywords: Classical Machine Learning, Deep Learning, Artificial Intelligence, Supervised Learning, Unsupervised Learning, Reinforcement Learning, Regression, Dataset, Algorithms

1 INTRODUCTION

The idea that computers might be programmed to think and reason first surfaced in 1956. According to them, "every aspect of learning or any other feature of intelligence [could], in principle, be so precisely described that a machine [could] be made to simulate it [8]." They called this artificial intelligence. Artificial intelligence (AI) is, to put it briefly, the study of automating intellectual tasks that are normally performed by humans. Two specific methods for doing this are ML and DL. AI does, however, encompass methods that do not entail learning.

In 1959, Arthur Samuel developed an algorithm that could play computer checkers at a championship level. In order to do this, Samuel used a minimax algorithm; he also popularized the phrase "machine learning." The earliest artificial neural network, which had 40 linked neurons, was created by Marvin Minsky and Dean Edmonds when computers still used punched cards to operate [9]. Alexey Ivakhnenko, the founder of deep learning, and Valentin Lapa, two other Sovi-et scientists, went on to demonstrate what is regarded as the "first ever multi-layer perceptron," a hierarchical representation of a neural network.

The fight for artificial intelligence is still going strong after over 70 years of travel. Permitting software applications to gradually improve their accuracy in predicting outcomes is the foundation

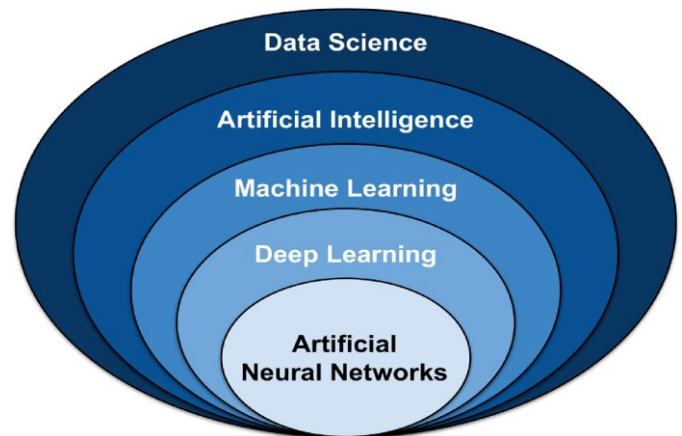


Fig. 1. Umbrella of select data science techniques [1]

of classical machine learning (CML). Predictive maintenance, business process automation, and the detection of malware and fraud are the three primary applications of machine learning today. The reason CML has become so well-known is because it provides businesses with business models that enable the creation of new goods, which notably sets multinational corporations apart. A data scientist determines what kind of algorithm is needed based on the data that must be anticipated. CML has been used in all major technical disciplines throughout the past few decades. Healthcare, manufacturing, logistics, recognizing problems, banking, and commerce are just a few industries where artificial intelligence is already beginning to provide major benefits.

By creating algorithms that most accurately reflect a set of data, machine learning (ML) focuses on the learning component of AI. Machine learning (ML) employs subsets of data to produce algorithms that may use unique or distinct combinations of

features and weights than can be obtained from first principles (Fig. 2B), in contrast to classical programming (Fig. 2A), in which an algorithm may be explicitly implemented using known features.

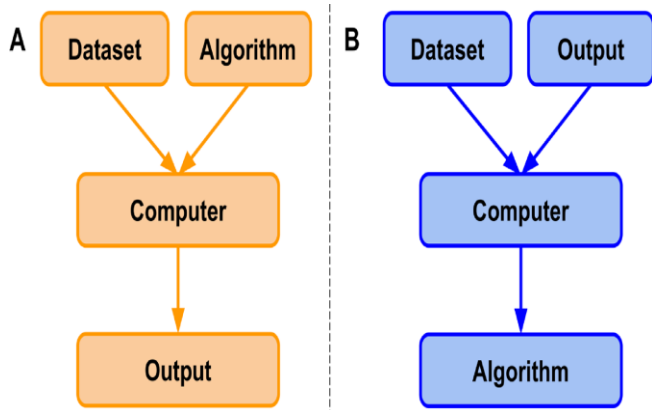


Fig. 2. Classical programming (A) versus machine learning paradigm(B). [10]

In ML, there are four commonly used learning methods, each useful for solving different tasks: supervised, unsupervised, semi-supervised, and reinforcement learning.

2 CLASSICAL MACHINE LEARNING METHODS

2.1 SUPERVISED LEARNING

Supervised Learning (SL) involves the use of well-labelled training data based on which machines predict the output – part. But in addition to these encouraging features, there are also a number of obstacles that would call for concurrent technological advancement. Generally speaking, two types of CML are supervised and unsupervised learning. Additionally, the different categories are divided into four groups: semi-supervised, supervised, unsupervised, and reinforcement learning. The purpose of this study is to review the idea, definitions, and various uses of CML in the modern business world

Analysis of Classical Machine Learning and Modern Deep Learning Methodologies 3



Fig. 3. Types of Machine Learning Models [2]

of the labelled data being tagged to correct output. Herein, the supervisor role is played by the training data that guides and teaches the machines to predict the output correctly

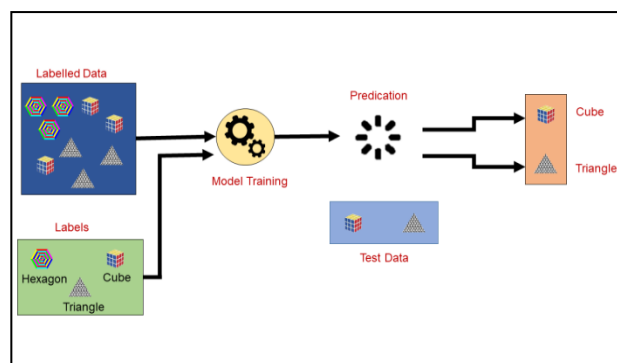


Fig. 4. A simple schematic of Supervised Learning [3]

The type of training dataset is first determined after which the labelled training data is collected. Thereafter, the dataset is classified or distributed into training dataset, test dataset and validation dataset. A suitable algorithm for the model is then selected which is then executed on the training dataset post which the accuracy of the model is evaluated by provision of the test set. The model is declared correct in case it predicts the correct output. The two types of SL algorithms are as under:

- Regression
- Classification

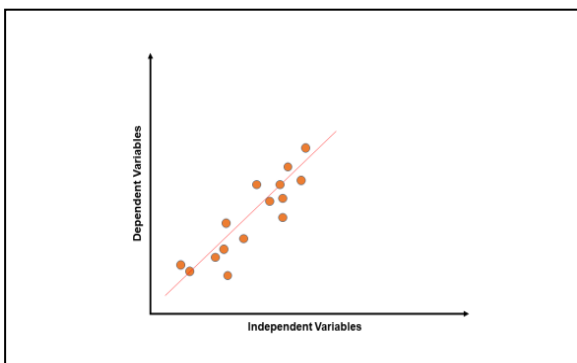


Fig. 5. A classical Linear Regression Graph

Regression.

Regression models are used when the output variable is a real or continuous value, such as “weight” or “salary”. It helps in finding the correlation between variables and enables us to predict the continuous output variable based on one or more predictor variables. It can help in future predictions such as sales, weather, market trends, etc. Linear Regression is used for predictive analysis and is one of the most common regression methods. It shows the linear relationship between the independent variable (X-axis) and the dependent variable (Y-axis). It works on the mathematical equation

$$\text{where } a \text{ \& } b \text{ are coefficients} \quad Y = aX + b \quad (1)$$

The relationship between the variables can be positive or negative. The goal of the linear regression algorithm is to fetch the best values for a & b to find the best fit line which should have the least error between predicted and actual values.

- Cost Function - The cost function helps to figure out the best values for a & b so as to provide the best fit for the data points. This is done by optimization of the regression coefficients or weights, in effect it is used to find the accuracy of the mapping function which maps the input to the output variable. It is the Mean Squared Error (MSE) cost function that is used in Linear Regression – the average of squared error that occurred between the predicted and actual values.
- Gradient Descent – In the case of one or more inputs, the values can be optimized by minimizing the error of the model using a process of iteration. It calls for selection of a learning rate parameter that determines the size of the improvement step that is to be taken in each iteration. The aim is to converge the algorithm at the minima.

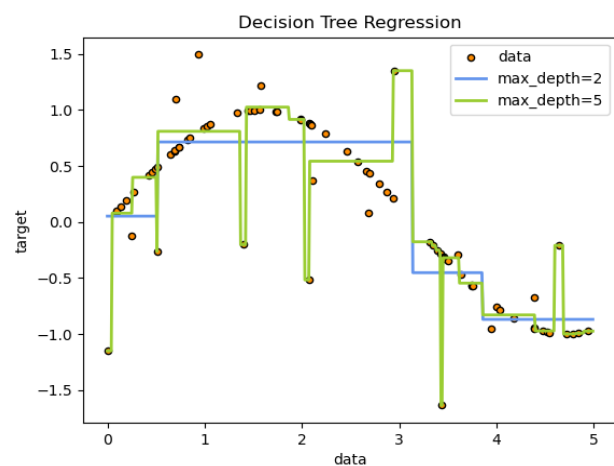


Fig. 7. Decision Tree Regression [7]

- Ridge Regression – A small bias is introduced which is known as Ridge Regression Penalty. In case of polynomial regressions which are bound to fail if there is high collinearity between independent variables, these can be solved using Ridge Regression. It works as a Regularization technique used to reduce the complexity of the model.
- Lasso Regression – While the Ridge Regression technique uses the square of weights, Lasso uses only the absolute weights – also called L1 Regularization.

Making predictions with Linear Regression is the ultimate task that is to be performed. Understanding of how it is used in CML can be demonstrated with a simple example. For instance,

- Support Vector Regression – A supervised learning algorithm that can be used for both regression as well classification, when used for regression problems it is termed as Support Vector Regression. The main goal of a support vector regression is to consider the maximum datapoints within the boundary lines and the best-fit line.
- Decision Tree Regression – Used for solving both classification and regression problems, it can provide solution to both numerical & categorical data. It works on a tree-like structure in which each internal node represents the “test” for an attribute and each branch represents the results of the “test” while the Leaf Node represents the final result, the same is elucidated in the below figure. Decision tree regression observes features of an object and trains a model in the structure of a tree to predict data in the future to produce meaningful continuous output. Continuous output means that the output/result is not discrete, i.e., it is not represented just by a discrete, known set of numbers or values.
- Random Forest – Capable of performing both regression as well as classification tasks. It is similar to a decision tree only that it combines multiple decision trees and the final output is predicted based on the average of each output – random forest regression models are quite powerful and accurate.

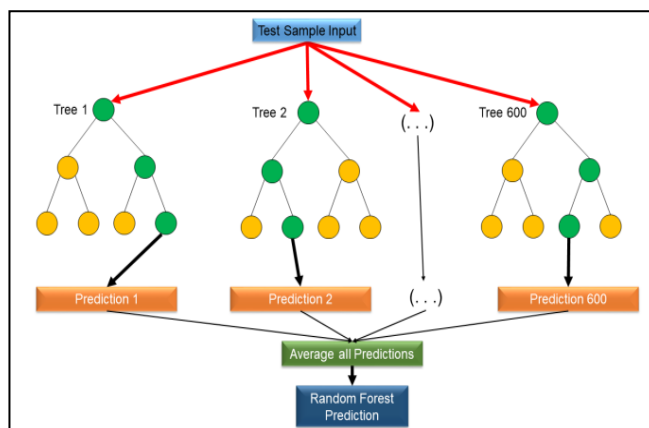


Fig. 6. Random Forest Trees [4]

the salary (y) of an individual is to be determined based on his/her experience (x). A simple equation would be as under

$$Y = A_0 + B_0 * X \tag{2}$$

Here A_0 is the bias coefficient and B_0 is the coefficient of experience. A learning algorithm is then applied to produce suitable coefficient values to arrive at near-correct inputs. As the model is trained over time, a near-correct output is envisaged. The most crucial step, usually the first step, is preparation of data. Clarity in respect of linearity/ non-linearity of data is important – conversion of data to log transform or exponential relationship may be undertaken. Data cleaning options to remove noise and expose better the variables for further processing. Highly correlated data has tendencies to over-fit the data, thus removal of most correlated data is a way to go about. It is further seen that Gaussian Distributed variables offer better outputs.

Some of the disadvantages of Linear Regression are [5]:

- Sensitivity to outliers – preprocessing of the dataset
- Highly Correlated data needs to be sieved

Classification.

Classification algorithms are used in cases where output variable is categorical for instance the cases of Yes-No, True-False, Girl-Boy. They are used to determine the class of an object. Some of the widely used algorithms are – k-Nearest Neighbour, Decision Tree, Logistic Regression, and Support Vector Machines.

- Logistic Regression – It is similar to linear regression but is used when the dependent variable is in categorical format like yes/ no, simply stated as binary. Firstly, a linear regression is performed to get the model. Below is a Logistic Sigmoid function.

$$p = \frac{1}{1 + e^{-y}} \quad (3)$$

It is applied to get the probabilities of the variable belonging in either class.

- K-Nearest Neighbours – A non-parametric algorithm reason being it makes no assumptions on the data. It is an algorithm

2.2 UNSUPERVISED LEARNING

Unsupervised Learning (UL) uses CML algorithms to analyze and cluster unlabeled datasets. They are able to discover hidden patterns or data groupings without human intervention. The primary difference between SL and UL is that SL models learn to predict based on labelled datasets while UL models, without human intervention, are provided only with input variables and no corresponding output data. UL has found use in real-life applications for instance Medical Imaging, Anomaly Detection, Recommendation Engines to name a few. Clustering and Association are the two types of Unsupervised Learning.

- Clustering is a process by which objects are grouped into clusters such that those with similarities are grouped together – it works on the system of finding commonalities.
- Association is a UL method used for finding the relationships between the variables in a large database. Market Analysis is
- Prone to over-fitting and noise
- Preconceived assumption of linearity between variables

that does not learn from the training set instead it stores the dataset. It stores all the available data and classifies a new data point based on the similarity.

Major advantages of Supervised Learning are:

- Prior experiences of the model allow for a better prediction.
- It helps in practical issues like fraud detection, spam filtering.
- It is useful in classification problems.
- Allows for accurate distinction of classes.

Disadvantages of Supervised Learning are listed as under:

- No unknown information is provided by Supervised Learning unlike the case in Unsupervised Learning.
- The accuracy of the training set plays a pivotal role in training the dataset.
- Some of the complex tasks of CML may not be handled by Supervised Learning.
- Wrong outputs may be obtained with regards to variables not matching any input/ training set data.

an example of association rule, for instance, people who buy more of X will tend to also buy more of Y. The rule-based method is poised to discover relationships & associations between different variables in large-scale datasets – the rules mandate how often a certain item occurs in a dataset and how strong & weak are the connections between different objects.

Most commonly used algorithms in Unsupervised Learning are:

- K-means Clustering – An example of exclusive clustering, also called “hard” clustering, where data points are assigned into K groups, where K denotes the number of clusters based on the distance from each group’s centroid. Clusters will be formed based on common distance to a given centroid. Large K values indicate smaller groupings and vice versa.

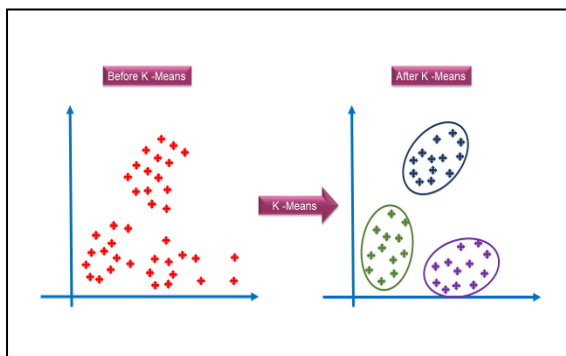


Fig. 8. K-Means Clustering [6]

- Hierarchical Clustering – It is an analysis that can either be agglomerative or divisive. While Agglomerative is a “bottoms-up” approach, divisive clustering can be defined as “top-down”. In case of agglomerative clustering, data points are kept as separate groupings and then in an iterative process, they are merged until one cluster is obtained. On the other hand, in divisive clustering a single data cluster is divided based on the differences between data points.
- Probabilistic Clustering – Data points are clustered based on the probability that they belong to a particular distribution. Gaussian Mixtures are made up of an unspecified number of probability distribution functions to determine which probability distribution a data set belongs to.
- Apriori Algorithm – It is designed to work on datasets that contain transactions, by making use of frequent item-sets it generates association rules. These item-sets are those with support greater than the threshold value, for instance, in two transactions $A=\{1,2,4,6,8\}$ and $B=\{2,4,7,5\}$ then 2 & 4 are the frequent itemsets. The Apriori algorithm was R. Agarwal and Srikant in the year 1994 and chiefly uses breadth-first search and hash tree method to calculate the item-set associations.

Some of the challenges in Unsupervised Learning are:

- Longer times for training
- Risk of inaccurate results is higher
- Validation of output variables by human intervention
- Reward Signal – The signal that is sent by the environment to the agent is known as a reward signal. They are based upon whether the actions carried out by the agent are good or bad wherein the main goal of the agent is to maximize the positive signals/ rewards.
- Value Function – It defines as to how good or bad the situation is and how much rewards an agent can expect.
- Model – The model mimics the behaviour of the environment which helps it to predict the next state and reward, meaning it

- Lack of transparency on basis of clustering and computational complexity

2.3 REINFORCEMENT LEARNING

Reinforcement Learning (RL) is a feedback-based CML technique in which the model learns and trains to act in an environment by performing actions and also seeing the results of those actions – for each correct or good action that is performed, the agent gets a positive feedback and a penalty for each wrong action. The data is not labelled and the system works on the principle of feedback thereby causing the agent to learn from experience. RL is a core area in Artificial Intelligence (AI) – the model need not be pre-programmed and neither is there a requirement of any human intervention. RL can be model-based, value-based or policy-based. The Model-based approach requires the formulation of a virtual model which is created for the environment and the agent then explores it to learn from it. The Value-based approach finds the optimal value function while the Policy-based approach calls for the agent to apply such a policy that each action assists in maximizing the future rewards – it is of two types deterministic and stochastic. The following make up the elements of RL:

- Policy – It defines as to how an agent must behave at a given time. It defines the behaviour of the agent and thus forms the core of the RL – policy can be deterministic or stochastic.

$$\text{Deterministic} - a = \prod (s) \tag{4}$$

$$\text{Stochastic} - \prod (a | s) = P [A_t = a | S_t = s] \tag{5}$$

provides a way to take a course of action by considering all future situations before the situations actually take place. Undermentioned algorithms are the primary ones being used in RL:

- Q-Learning – An off-policy algorithm used for “temporal difference learning”. These methods work by way of comparison of successive temporal predictions. Essentially, it initializes the Q-table, post selection of an action to perform it performs the selected action thereafter measuring the reward and updating the Q-table.

- State Action Reward State Action (SARSA) – SARSA is an algorithm where an Action (A) is taken in State (S) and a reward (R) is given to the agent which then ends in new state (S1) and then takes action (A1) in new state S1. The process starts with initialization of $Q(S,A)$ to random values. Thereafter, the selected action is taken and the reward (R) and next state (S1) is seen. Thus the state, reward and action in SARSA procedure are used to update the function in each iteration.
- Deep Q Neural Network (DQN) – DQN algorithm is essentially Q-learning using neural networks. It is a type of reinforcement learning algorithm that makes use of a deep

neural network to approximate the Q-function. Rather than a simple table of values, it is able to handle a large number of states and actions. The Q-function is essentially non-linear, and may have many local minima thus making convergence to the correct Q-function very difficult. Some of the techniques to counter this are experience replay and target networks. In case of experience replay, a subset of past experiences is stored in memory buffer to update the Q-function while Target Networks are used to stabilize the Q-function updates. Deep Q-Learning finds its applications in a wide variety of practical issues like robotics, autonomous systems to name a few.

3 DEEP LEARNING METHODS

While machine learning makes computers capable of amazing things, the model is not good at simulating human thought processes. Fortunately, deep learning shines in just that situation. Some of the most popular varieties of deep learning algorithms are listed below.

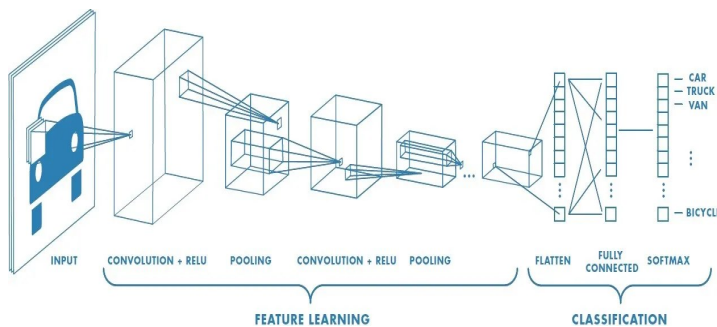


Fig. 9. CNN working [17]

The goal of this field is to make it possible for machines to see and understand the world similarly to humans, and to even use that perception and understanding for a wide range of applications, including natural language processing, recommendation systems, media recreation, image and video recognition, and image analysis and classification. Convolutional neural networks are the main algorithm that has been used to build and refine the advances in computer vision using deep learning over time. Convolutional neural networks (CNNs) are algorithms that resemble the visual processing system found in the brain. By filtering a visual cue and evaluating elements like patterns, textures, forms, and colors, they are able to interpret images and identify objects. The AI domains of computer vision and image recognition, which instruct machines on how to interpret the visual environment, are frequently driven by

3.1 Convolutional neural networks

Artificial Intelligence has shown exponential growth in terms of closing the gap between human and machine capabilities. Researchers and fans alike, work on different parts of the field to make great things happen. Computer vision is one of these numerous domains.

CNNs. Convolutional neural networks, often known as ConvNets or CNNs, are Deep Learning algorithms that are capable of taking an input image, giving distinct objects and elements in the image varying degrees of importance (based on learnable weights and biases), and then distinguishing between them. When compared to other classification methods, ConvNet requires a lot less pre-processing. While filters are manually designed in more archaic approaches, ConvNets may learn these filters and properties given sufficient training.

ConvNet architecture was inspired by the structure of the visual cortex and is comparable to the connectivity pattern of neurons in the human brain. Only in a small area of the visual field known as the Receptive Field do individual neurons react to inputs. To fill the whole visual field, a group of these fields overlap.

3.2 Recurrent neural networks

RNNs are distinct from other kinds of neural networks in that they can process input sequences. Their ability to perform tasks that rely on the data's order, such as anticipating the next word in a phrase, tracking the health of patients over time, or predicting stock market patterns, is made possible. Because of this, they are very important for many advanced AI applications. AI algorithms known as recurrent neural networks (RNNs) "remember" historical data points by utilizing internal

feedback loops. RNNs can make predictions about the future or use this memory of previous occurrences to help them grasp what is happening right now. They are very helpful for processing one data point at a time and sequencing data. With this degree of context, a deep neural network can "think" more effectively. An RNN-powered maps software, for instance, has the ability to "remember" when traffic patterns tend to worsen. It can then expedite route planning by using this knowledge to forecast future drive times.

By progressively scanning the data from left to right and updating the hidden state at each time step, the RNN accepts an

input vector (X) and outputs a vector (Y). All time steps have the same set of parameters. This indicates that the network as a whole uses the same set of parameters, denoted by U, V, and W. W stands for the weight connected to the connection between hidden layers, V for the connection from hidden layer h to output layer y, and U for the weight parameter controlling the connection from input layer X to the hidden layer h. Through parameter sharing, the RNN retains the information from prior input in its current hidden state, making it more efficient at processing sequential data and capturing temporal dependencies. At each time step t, the hidden state a_t is computed based on the current input x_t , previous hidden state a_{t-1} and model parameters as illustrated by the following formula:

It can also be written as, (6)

$$a_t = f(a_{t-1}, x_t; \theta) \tag{6}$$

$$a_t = f(U * X_t + W * a_{t-1} + b) \tag{7}$$

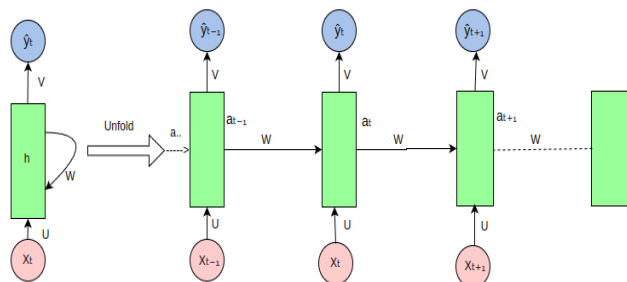


Fig. 10. RNN Working [18]

where,

- a_t represents the output generated from the hidden layer at time step t .
- x_t is the input at time step t.
- θ represents a set of learnable parameters(weights and biases).
- U is the weight matrix governing the connections from the input to the hidden layer; $U \in \theta$
- W is the weight matrix governing the connections from the hidden layer to itself (recurrent connections); $W \in \theta$
- V represents the weight associated with connection between hidden layer and output layer; $V \in \theta$
- a_{t-1} is the output from hidden layer at time t-1.
- b is the bias vector for the hidden layer; $b \in \theta$
- f is the activation function.

For a finite number of time steps T=4, we can expand the computation graph of a Recurrent Neural Network, illustrated in Fig.10 [18], by applying the equation (6) T-1 times.

$$a_4 = f(a_3, x_4; \theta) \tag{8}$$

Equation (8) can be expanded as,

$$a_4 = f(U * X_4 + W * a_3 + b)$$

$$a_3 = f(U * X_3 + W * a_2 + b)$$

$$a_2 = f(U * X_2 + W * a_1 + b)$$

The output at each time step t, denoted as y_t is computed based on the hidden state output a_t using the following formula,

$$\hat{y}_t = f(a_t; \theta) \tag{9}$$

Equation (9) can be written as,

$$\hat{y}_t = f(V * a_t + c) \tag{10}$$

when t=4,

$$\hat{y}_4 = f(V * a_4 + c) \tag{11}$$

where,

- \hat{y}_t is the output predicted at time step t.
- V is the weight matrix governing the connections from the hidden layer to the output layer.
- c is the bias vector for the output layer.

Backpropagation Through Time (BPTT).

Through backpropagation, the model's weights and biases are modified in response to the discrepancy between the goal value and the anticipated output. By minimizing the loss function, backpropagation seeks to enhance the model's performance. RNNs are trained using a unique form of backpropagation called Backpropagation Through Time, in which the error is transmitted backward through time up to the

first time step, t=1. The forward pass and backward pass are the two essential phases of backpropagation.

- **Forward Pass:** During forward pass, the RNN processes the input sequence through time, from t=1 to t=n, where n is the length of input sequence. In each forward propagation, the following calculation takes place
- **Backward Pass:** The backward pass in BPTT involves computing the gradients of the loss function with respect to the network's parameters (U, W, V and biases) over each time step.

$$a_t = U * X_t + W * a_{t-1} + b$$

$$a_t = \tanh(a_t)$$

$$\hat{y}_t = \text{softmax}(V * a_t + c)$$

The Fig. 11 [17] below also serves as an illustration of backpropagation for time step 4.

After processing the entire sequence, RNN generates a sequence of predicted outputs $\hat{y} = [\hat{y}_1, \hat{y}_2, \dots, \hat{y}_t]$. Loss is then computed by comparing predicted output \hat{y} at each time step with actual target output y. Loss function given by,

$$L(y, \hat{y}) = (1/t) * \sum (y_t - \hat{y}_t)^2 \tag{MSE Loss}$$

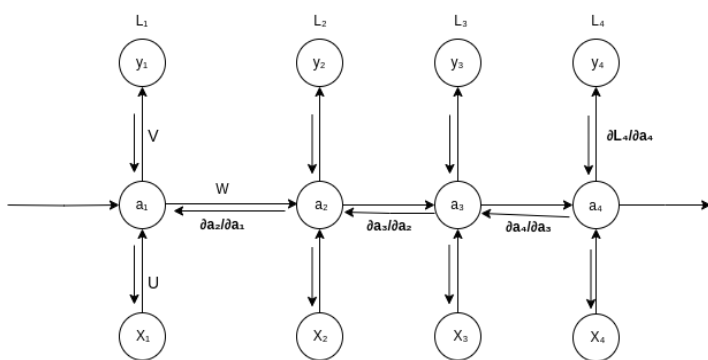


Fig. 11. Back Propagation Through Time (BPTT) [17]

Derivative of loss L w.r.t V

Loss L is a function of predicted value \hat{y} , so using the chain rule $\partial L / \partial V$ can be written as,

$$\partial L / \partial V = (\partial L / \partial \hat{y}) * (\partial \hat{y} / \partial V)$$

Derivative of loss L w.r.t W

Applying the chain rule of derivatives $\partial L / \partial W$ can be written as follows: The loss at the 4th time step is dependent upon \hat{y} due to the fact that the loss is calculated as a function of \hat{y} , which is in turn dependent on the current time step's hidden state a_4 , a_4 is

influenced by both W and a_3 , and again a_3 is connected to both a_2 and W , and a_2 depends on a_1 and also on W .

$$\begin{aligned} \partial L_4 / \partial W = & (\partial L_4 / \partial \hat{y}_4 * \partial \hat{y}_4 / \partial a_4 * \partial a_4 / \partial W) + (\partial L_4 / \partial \hat{y}_4 * \partial \hat{y}_4 / \partial a_4 \\ & * \partial a_4 / \partial a_3 * \partial a_3 / \partial W) + (\partial L_4 / \partial \hat{y}_4 * \partial \hat{y}_4 / \partial a_4 \\ & * \partial a_4 / \partial a_3 * \partial a_3 / \partial a_2 * \partial a_2 / \partial W) + (\partial L_4 / \partial \hat{y}_4 * \partial \hat{y}_4 / \partial a_4 * \partial a_4 / \partial a_3 * \\ & \partial a_3 / \partial a_2 * \partial a_2 / \partial a_1 * \partial a_1 / \partial W) \end{aligned}$$

Derivative of loss L w.r.t U

Limitations of RNN.

Gradients may grow excessively big, causing the exploding gradient problem, or too little, causing the vanishing gradient problem, as they propagate backward through time during backpropagation. The problem with vanishing gradients is that they might get too tiny, making it difficult for the network to successfully identify long-term dependencies. Although it

Similarly, $\partial L / \partial U$ can be written as,

$$\begin{aligned} \partial L_4 / \partial U = & (\partial L_4 / \partial \hat{y}_4 * \partial \hat{y}_4 / \partial a_4 * \partial a_4 / \partial U) + (\partial L_4 / \partial \hat{y}_4 * \partial \hat{y}_4 / \partial a_4 \\ & * \partial a_4 / \partial a_3 * \partial a_3 / \partial U) + (\partial L_4 / \partial \hat{y}_4 * \partial \hat{y}_4 / \partial a_4 \\ & * \partial a_4 / \partial a_3 * \partial a_3 / \partial a_2 * \partial a_2 / \partial U) + (\partial L_4 / \partial \hat{y}_4 * \partial \hat{y}_4 / \partial a_4 \\ & * \partial a_4 / \partial a_3 * \partial a_3 / \partial a_2 * \partial a_2 / \partial a_1 * \partial a_1 / \partial U) \end{aligned}$$

This sums up the gradients of loss over all time steps, and it illustrates the main distinction between the normal backpropagation strategy and BPTT.

might take a very long time, it can still converge throughout training. On the other hand, in the case of the exploding gradient problem, a big gradient may result in numerical instability during the training process, which could lead to the model deviating from the optimal solution and complicate the network's convergence to global minima.

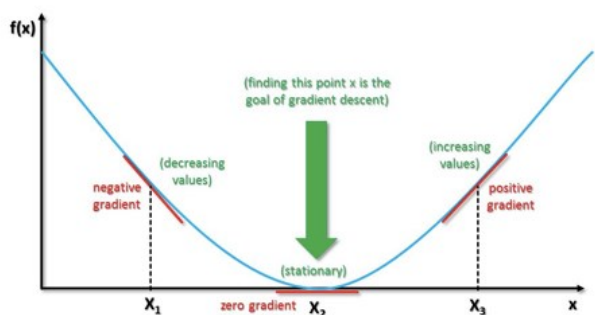


Fig. 13. Gradient Descent [18]

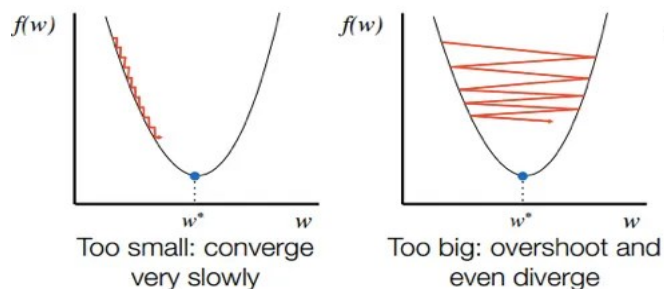


Fig. 12. Vanishing and Exploding Gradient [18]

3.3 Multilayer perceptron

Deep learning is the main application for multilayer perceptrons, or MLPs. Because MLPs are feedforward neural networks, which handle data and patterns that other algorithms find difficult to interpret, they are superior at processing information that flows just in one direction and does not rely on feedback loops. Consider the situation where black and white photos are used to categorize dogs and cats. That is equivalent to saying that as the value of a pixel grows, the likelihood that it represents a dog either rises or falls. That isn't logical. There are, after all, both black dogs and black cats in the world, as well as white dogs and white cats.

Deciphering an image typically necessitates permitting more intricate connections between our inputs and outputs, taking into account the potential that our pattern may be defined by interactions among the numerous elements. In these situations, the accuracy of linear models will be low. By adding one or more hidden layers, we may model a wider class of functions. Putting several layers of neurons on top of one another is the simplest method for achieving this. Up until we produce an output, each layer feeds into the layer above it. This design is frequently referred to as a "multilayer perceptron." We pile a lot of layers on top of one another while using an MLP.

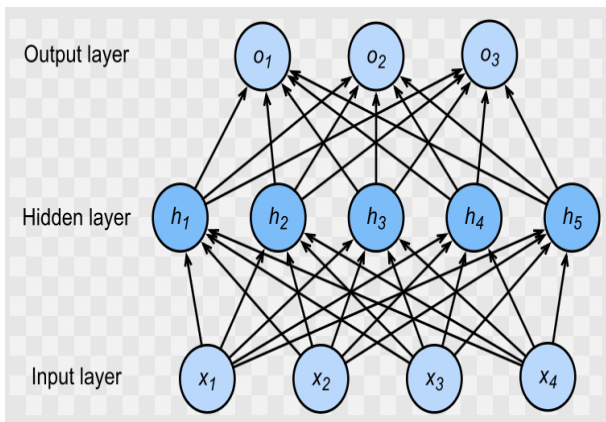


Fig. 14. MultiLayer Perceptron(MLP) [20]

There are four inputs and three outputs in the multilayer perceptron shown above, and there are five hidden units in the hidden layer in the middle. The multilayer perceptron has two levels in total because there are no calculations made at the input layer. The input layer's inputs are entirely coupled to the neurons in the hidden layer. Additionally totally connected are the neurons found in the hidden layer and the output layer. Consequently, in the multilayer perceptron, the output layer and the hidden layer are both fully connected layers. Since this is a broad topic that is outside the purview of this work, we won't go into further detail about MLP math here.

4 CLASSICAL MACHINE LEARNING METHODS VS DEEP LEARNING METHODS

Machine learning has evolved into deep learning. While both algorithms require data to learn, the way they analyze and interpret that data is where they differ most significantly. Basic machine learning models still require human interaction, even if they do get better over time at carrying out their particular tasks as they process fresh data. An engineer must intervene and make corrections if an AI algorithm produces an erroneous prediction. With a deep learning model, little to no human assistance is needed for an algorithm to assess the accuracy of a prediction using its own neural network. By using an algorithm that mimics the functioning of a human brain, a deep learning model can learn on its own. Other key differences include:

- Whereas DL employs millions of data points, ML only uses thousands. Small datasets are typically sufficient for machine learning algorithms to function properly. Compared to typical machine learning algorithms, Deep Learning takes a big amount of data to comprehend and perform better.
- ML algorithms use explicit programming to solve issues. DL learning methods use neural network layers to answer problems.
- ML algorithms can be trained in a matter of seconds to several hours, which is comparatively short. However, the training process for deep learning algorithms might take several hours or even weeks.

Neural networks are the foundation of deep learning algorithms, while standard machine learning algorithms, such as SVM, decision trees, logistic regression, and linear regression, are derived from classical mathematics. High complexity problems are a good fit for deep learning methods. Additionally, DL algorithms outperform classical ML in terms of accuracy. In contrast to classical AI, machine learning (ML) monitors data, finds patterns, and continuously improves its skills. However, ML benefits greatly from diverse data, as it can learn from and get better at large datasets.

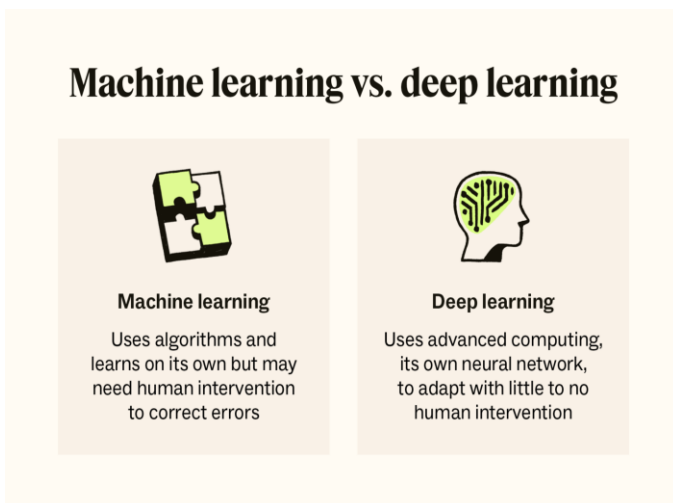


Fig. 15. Difference in ML and DL [14]

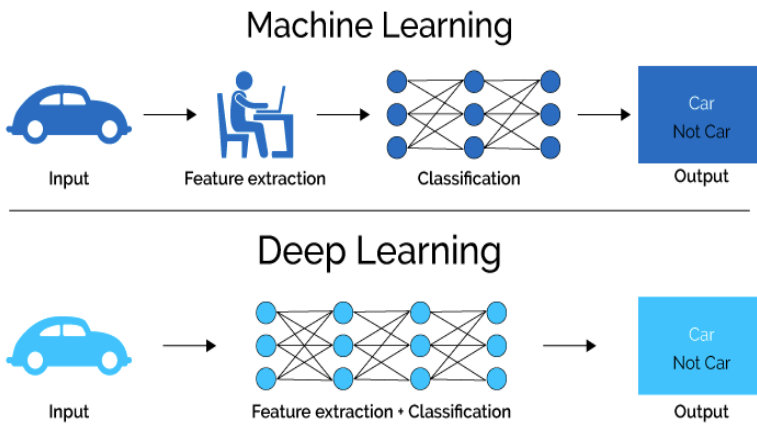


Fig. 16. Classical ML vs Deep Learning [15]

4.1 AI and its relation to Deep Learning and Machine Learning

Programming software to mimic human intelligence is known as artificial intelligence (AI). AI is capable of doing this through utilizing techniques like machine learning and deep learning to learn from data.

- Artificial intelligence: AI is a broad field that includes both machine learning and deep learning. Its objective is to create intelligent technologies that are capable of performing cognitive tasks including decision-making, sentiment analysis (reading a text to determine the emotions and tone of the writer), and problem-solving.

- Machine learning: By observing trends in its training data, machine learning produces predictions and conclusions that are accurate.
- Deep learning: DL is a subset of machine learning. With this model, an algorithm using a neural network may assess the accuracy of a forecast without the need for human participation. As a kind of brain, deep learning models have the capacity to accumulate vast amounts of knowledge over time.

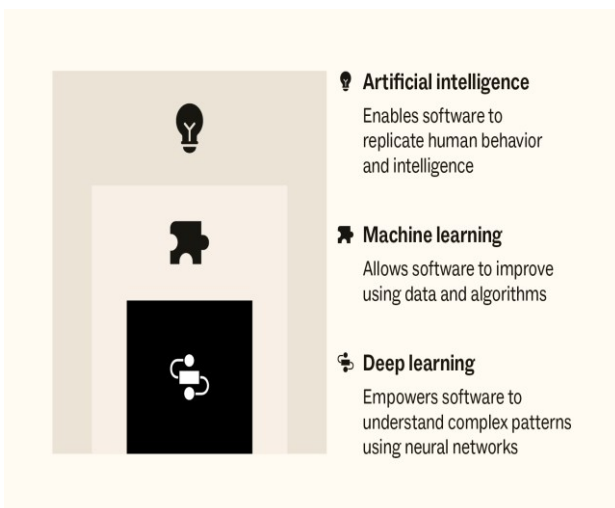


Fig. 17. AI, ML and DL [14]

4.2 Machine Learning

ML is a subfield of AI that has been taught on statistical models and algorithms to help it with decision-making and prediction. Machine learning algorithms can enhance and adapt over time, expanding their capabilities, by using training and historical data. In order for machine learning to keep getting better results, human engineers must give it pertinent, pre-processed data. It is skilled at finding patterns in data to solve complicated issues and produce significant insights.

How machine learning works.

Machine learning is an algorithm that enables computers and software to learn patterns and relationships using training data. When a machine learning model interacts with people and learns from their past data, it will get better over time.

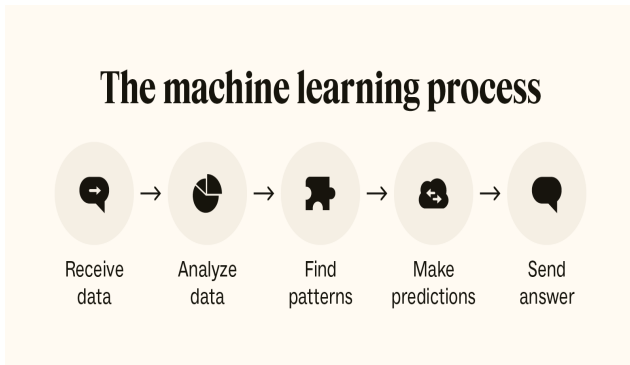


Fig. 18. Machine Learning(ML) process [14]

Different types of ML algorithms

– Linear Regression

An essential algorithm in machine learning, particularly in supervised learning, is linear regression. Based on one or more input features, it is used to predict a continuous outcome. The input variables and the output variables are assumed to have a linear connection by the algorithm. For instance, estimating the cost of a home by factoring in location, square footage, and the number of bedrooms.

– Decision Trees

Tree-like structures called decision trees use input features to inform their judgments. The judgments or tests on a certain feature are represented by each node in the tree, and the results of these decisions are shown by the branches. Example: Determining whether an email is spam or not by looking at its sender, subject, and content.

– Support Vector Machines (SVM)

Strong algorithms like Support Vector Machines (SVM) are employed for regression and classification problems. It operates by locating the hyperplane in the feature space that

best divides various classes. SVM works very well in settings with many dimensions. Example: Using visual characteristics to determine if a particular image shows a dog or a cat.

– K-Nearest Neighbors (KNN)

An easy-to-use yet powerful approach for regression and classification is K-Nearest Neighbors (KNN). A new data point is classified according to the feature space's k-nearest neighbor's majority class. As an illustration, consider predicting a person's preferred movies based on those of their k-nearest neighbors.

– Neural Networks

Inspired by the structure of the human brain, neural networks are made up of layers of interconnected nodes. Multiple layers are involved in deep neural networks, or deep learning, which may learn complex representations. Neural networks are certainly used in some way by many of your favorite AI chatbots, and this is really helpful for creative AI. For instance, image recognition uses a neural network to recognize objects in pictures.

diseases can also be identified with the aid of ML-powered predictive models.

Finance. Machine learning is useful in finance for credit scoring, algorithmic trading, and fraud detection. Fraud detection models are able to spot anomalous behavior and highlight possible fraudulent activities by examining trends in transaction data. In order to make snap decisions in the financial markets, algorithmic trading also depends on intricate machine learning models.

Machine learning examples in Real World

Our daily lives currently involve machine learning (ML), and in the future, its influence will only increase. Here are a few instances of current applications of ML in the real world:

Healthcare. The field of medicine is greatly benefiting from machine learning. Algorithms are employed in the analysis of medical imaging, the forecasting of disease outbreaks, and the support of diverse illness diagnosis. Based on their lifestyle and medical history, people at risk of specific

Retail. Machine learning is used by e-commerce platforms to estimate demand, provide personalized suggestions, and handle customer care. Recommendation engines that leverage machine learning (ML) have the ability to analyze user behavior and make personalized product recommendations. Models for demand forecasting assist in streamlining inventory control and guaranteeing that goods are available when consumers need them.

Autonomous Vehicles. Machine Learning is being used by the automobile sector to produce autonomous vehicles. Real-time data is collected by these cars' sensors and cameras, which are then processed by ML algorithms to provide choices on traffic patterns, obstacle avoidance, and navigation.

Natural Language Processing (NLP). NLP is a branch of machine learning that focuses on giving computers the ability to comprehend, translate, and produce human

4.3 Deep Learning

Deep Learning is a branch of machine learning that builds an "artificial neural network"—a machine learning sector that can learn on its own and make wise decisions—by layering algorithms. Continuous data analysis is possible with deep learning models. Like humans, they come to conclusions by absorbing input, looking up relevant information in data stores, and coming up with a solution. With the help of this method, it can detect both speech and images. DL has had a significant impact on a number of industries, including robotics, healthcare, finance, retail, and logistics.

language. NLP algorithms are used by sentiment analysis tools, language translation services, and virtual assistants like Alexa and Siri to process and produce text that is human-like.

These are only a handful of the numerous ways that machine learning (ML) is being applied to improve the ease, security, and enjoyment of our lives. We should anticipate seeing even more ground-breaking and inventive uses of ML in the years to come as it continues to advance. Additional real-world uses include the way on-demand music and video streaming services like Spotify, Apple Music, and YouTube are powered by machine learning and reinforcement learning, a branch of machine learning where the algorithm makes judgments by interacting with its surroundings. Machine learning algorithms compare a listener's preferences (such as saved songs, playlists, followed artists, and skipped tracks) with those of other listeners who share similar musical likes in order to suggest new songs or artists to the user

How Deep Learning works?

Artificial neural networks (ANNs), which are hierarchical structures of algorithms, are used in deep learning applications. A user must supply input (unlabeled data) in order to use a deep learning model. After then, information is routed via the neural network's hidden layers, where it employs mathematical operations to find patterns and provide a final output (answer). The human brain's network of neurons, which conveys information through messages, served as an influence for the algorithm's architecture. As a result, deep learning models are typically more sophisticated than traditional machine learning models.

What is a Neural Network?

Neural networks—also called artificial neural networks (ANNs)—are a means of teaching AI to process information in a manner akin to that of a human brain. In order for neural networks to work, information must first enter from an input, travel through nodes (intersection points) to deep, hidden layers (where neurons fire in the brain-like fashion), where the algorithm learns and then outputs its final response. Deep learning would not be possible without neural networks. The neural network's depth is the only factor that influences the algorithm's learning depth.

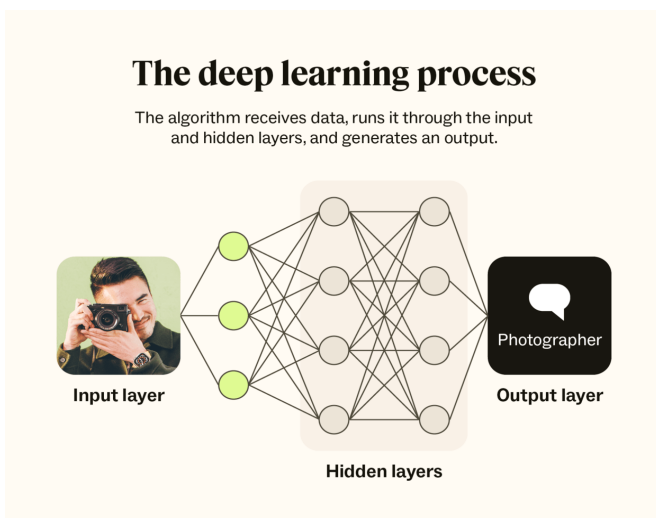


Fig. 19. Deep Learning (DL) Process [14]

Different Types of DL algorithms

While machine learning makes computers capable of amazing things, the model is not good at simulating human thought processes. Fortunately, deep learning shines in just that situation. Some of the most popular varieties of deep learning algorithms are listed below.

Convolutional neural networks.

Convolutional neural networks (CNNs) are algorithms that resemble the visual processing system found in the brain. By filtering a visual cue and evaluating elements like patterns, textures, forms, and colors, they are able to interpret images and identify objects. The AI domains of computer vision and image recognition, which instruct machines on how to interpret the visual environment, are frequently driven by CNNs.

Deep learning examples

Google DeepMind created a computer program, AlphaGo, with its own neural network that excels at the strategy game, Go.

- About the game: The game is easy to learn, but gamers must have a quick wit and good intuition to play well.
- AlphaGo's achievement: With deep learning, the program learned to play the abstract board game, Go. Before long,
-

5 APPLICATIONS OF CML AND DL

CML over the past few decades has found its use in varied real-life applications. Some of them are listed as under:

- Product Recommendations – The application finds its use in almost every e-commerce website
- Image Recognition - It has found its use in variety of application for instance security, healthcare, applications online.
- Sentiment Analysis – Real-time ML algorithms allow for tapping on sentiment analysis thus assisting in marketing drives, political campaigns to name a few.
- Predictive Technologies that allow for medical emergency prediction, regulation in banking sector, weather predictions.

Machine learning is predicted to grow in exciting ways as technology carries on evolving. Many different businesses are already using machine learning (ML), and this trend will

Recurrent neural networks.

Recurrent neural networks (RNNs) are artificial intelligence (AI) systems that use built-in feedback loops to "remember" past data points. RNNs can make predictions about the future or use this memory of previous occurrences to help them grasp what is happening right now. They are very helpful for processing one data point at a time and sequencing data. With this degree of context, a deep neural network can "think" more effectively. An RNN-powered maps software, for instance, has the ability to "remember" when traffic patterns tend to worsen. It can then expedite route planning by using this knowledge to forecast future drive times.

Multilayer perceptron.

The primary application of multilayer perceptrons (MLPs) as an algorithm is in deep learning. MLPs are categorized as feedforward neural networks, which means that unlike other algorithms, they can analyze unpredictable data and patterns because user input only travels in one direction without the need for feedback loops.

AlphaGo was defeating world-renowned Go masters, proving that with deep learning, machines could grasp abstract concepts and complex techniques.

- How AlphaGo did it: By playing against professional Go players, AlphaGo's deep learning model quickly succeeded by identifying patterns and maneuvers that were never seen before in AI. And the program did so without instructions on what move to make and when, as is traditionally seen among machine learning models.

only continue. Additionally, scientists are always creating new, more potent machine learning algorithms. These algorithms will be capable of operating on more potent hardware, learning from more complex data, and producing predictions that are more accurate.

The ability to understand and interpret the judgments made by machine learning (ML) models is becoming more and more crucial as they grow in complexity. This will guarantee that ML systems are utilized morally and sensibly and contribute to the development of trust in them.

As we look to the future, it seems likely that machine learning will continue to find its way into daily life, opening up new opportunities and changing the way we work and live. Furthermore, deep learning is becoming more and more applicable in practically every industry. The impact that deep learning will have on society both now and in the future is undeniable. Experts can't agree on whether this will have a beneficial or bad effect.

6 CONCLUSION

Now that CML has graduated to Deep Learning, there is a smooth connection between virtual and physical world. In the case of ML, things have advanced quickly and are still going in the correct way. AI is evolving at an exponential rate, and in the last several years, research has advanced to unprecedented heights. The financial and research-based efforts made in the fields of machine learning and artificial intelligence (AI) have been extraordinary and will continue to change the game in the future. When a robot player in the renowned game of AlphaGo defeated a human player, deep learning (DL) became an overnight "star." Deep learning training and learning methods have been widely acknowledged for "humanizing" machines. The fast development of deep learning and machine learning (ML)

technology is largely responsible for the advanced automation features that are already included in enterprise AI platforms. DL is "ubiquitous" in many areas of AI, including computer vision and natural language processing. All business areas, from marketing to customer experience, virtual reality to natural language processing (NLP), are gradually being overtaken by AI- and DL-enabled automated systems, tools, and solutions. The digital influence is pervasive.

7 REFERENCES

1. <https://tvst.arvojournals.org/article.aspx?articleid=2762344>
2. Mukhamediev, R.I.; Symagulov, A.; Kuchin, Y.; Yakunin, K.; Yelis, M. From Classical Machine Learning to Deep Neural Networks: A Simplified Scientometric Review. *Appl. Sci.* 2021, 11, 5541. <https://doi.org/10.3390/app11125541>
3. "Supervised Machine Learning", Javatpoint.com, <https://www.javatpoint.com/supervised-machine-learning> (accessed on 16 Aug 23).
4. Chaya, "Random Forest Regression", Medium.com, <https://levelup.gitconnected.com/random-forest-regression-209c0f354c84> (accessed on 17 Aug 23).
5. D. Madhugiri, "Linear Regression in Machine Learning: A comprehensive guide", Knowledgehut.com, <https://www.knowledgehut.com/blog/data-science/linear-regression-for-machine-learning> (accessed on 18 Aug 23).
6. P. Sharma, "K-Means Clustering", Analyticsvidhya.com, <https://www.analyticsvidhya.com/blog/2021/04/k-means-clustering-simplified-in-python>
7. https://scikit-learn.org/stable/auto_examples/tree/plot_tree_regression.html
8. Moor J. The Dartmouth College Artificial Intelligence Conference: the next fifty years. *AI Mag.* 2006; 27: 87–87. doi:10.1609/aimag.v27i4.1911.
9. V. Gladchuk. "The History of Machine Learning: How did it all Start?" Labeyourdata.com. <https://labeyourdata.com/articles/history-of-machine-learning-how-did-it-all-start>(accessed on 15Aug23).
10. James G Witten D Hastie T Tibshirani R , eds. *An Introduction to Statistical Learning: With Applications in R*. New York: Springer; 2013.
11. Hastie T, Tibshirani R, Friedman JH. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2nd ed. New York, NY: Springer; 2009.
12. NVIDIA Blog: Supervised Vs. Unsupervised Learning. The Official NVIDIA Blog. <https://blogs.nvidia.com/blog/2018/08/02/supervised-unsupervised-learning/>. Published August 2, 2018. Accessed October 24, 2019.
13. Mei S, Montanari A, Nguyen P-M. A mean field view of the landscape of two-layer neural networks. *Proc Natl Acad Sci U S A.* 2018; 115: E7665–E7671. doi:10.1073/pnas.1806579115. [CrossRef] [PubMed]
14. <https://www.zendesk.com/in/blog/machine-learning-and-deep-learning/>
15. <https://lamiae-hana.medium.com/classical-ml-vs-deep-learning-f8e28a52132d>
16. <https://www.dataversity.net/the-future-of-deep-learning/>
17. <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>
18. <https://medium.com/@poudelsushmita878/recurrent-neural-network-rnn-architecture-explained-1d69560541ef>
19. <https://www.sciencedirect.com/topics/mathematics/multilayer-perceptron>
20. https://ja.d2l.ai/chapter_deep-learning-basics/mlp.html
21. Dargan, S., Kumar, M., Ayyagari, M.R. et al. A Survey of Deep Learning and Its Applications: A New Paradigm to Machine Learning. *Arch Computat Methods Eng* 27, 1071–1092 (2020). <https://doi.org/10.1007/s11831-019-09344-w>
22. Wicht B, Fischer A, Hennebert J (2016) Deep learning features for handwritten keyword spotting. In: Proceedings of the 23rd international conference on pattern recognition (ICPR). <https://doi.org/10.1109/icpr.2016.7900165>
23. Abadi M, Paul B, Jianmin C, Zhifeng C, Andy D, Jeffrey D, Matthieu D (2016) Tensorflow: a system for large-scale machine learning. In: The proceedings of the 12th USENIX symposium on operating systems design and implementation (OSDI'16), vol 16, pp 265–283