International Journal of Natural and Engineering Sciences
E-ISSN: 2146-0086 19(3): 120-143, 2025
Research Article

A Comprehensive Overview of Machine Learning and Deep Learning Algorithms

Muhammad Tariq Pervez[1*], Samreen Sohail[2], Syed Hassan Abbas[3], Shahid Manzoor[4], Mirza Jawad ul Hasnain[5]

[1] *Virtual University of Pakistan, Department of Biological Sciences, Lahore, Pakistan*
[2] *Virtual University of Pakistan, Department of Bioinformatics, Lahore, Pakistan*
[3]*The Institute of Biomedical Sciences and School of Life Sciences, East China Normal University, The Center for Bioinformatics and Computational Biology, Shanghai, China (200241)*
[4]*University of the Punjab Department of Data Science, Faculty of Computing & Information Technology, Lahore, Pakistan*
[5]*University of the Punjab Department of Data Science, Faculty of Computing & Information Technology, Lahore, Pakistan*

*Corresponding Author: m.tariq@vu.edu.pk

Abstract

Machine learning and deep learning are two major subareas of artificial intelligence that are based on data-driven learning strategies. This review article investigates the both, comparing their algorithms based on a number of aspects such as data needs, feature engineering, preprocessing steps, model intricacy, training time, computational resources and how well they deal with various types of data, noise, and outliers. The objective of this study was to help researchers and practitioners choose the most appropriate artificial intelligence method for their problems and needs. The most significant aspect of this study is its side-by-side examination of machine learning and deep learning approaches across both technical and real-world dimensions - something that is often missing in existing studies. By highlighting the strengths and weaknesses of each, this work offers clear, practical guidance to select the most suitable approach for a specific task.

Keywords: Machine learning, Deep learning, Artificial intelligence, Data-driven models, Feature engineering, Model complexity, Computational resources, Algorithm comparison, Model selection

Introduction

Machine learning (ML) that is an important domain of artificial intelligence (AI) (Figure 1), analyses data to extract patterns, make predictions, or support decision-making. It has gained popularity for its capability to develop intelligent systems that learn from data and improve with the passage of time without being explicitly programmed [1-3]. ML algorithms are classified into three main categories: (1) supervised learning (SL): In this technique, models are trained on labeled data which means the correct output is already known [4-5]; (2) unsupervised learning (USL): In case of this approach, the model is trained on unlabeled data. It means that USL methods learn to identify hidden patterns on their own; (3) reinforcement learning (RL): Here, the model communicate with an environment and learns based on the feedback-receiving rewards or penalties for its actions. Gradually. RL techniques achieve capability to make good decisions through trial and error [5].

Deep learning (DL) is a branch of machine learning, where artificial neural networks having several layers are trained [6,7]. These models also called as deep neural networks automatically capture intricate patterns by building hierarchical representations of data [1,8]. One of the most important strength of DL methods is that they are able to extract related features automatically from raw data, making them important in the scenarios where manual feature engineering is a challenge. The most popular DL architectures are convolutional neural networks (CNN) which is well known for image processing and

recurrent neural networks (RNNs) that focuses to handle sequential data. These models have shown amazing results across various fields, such as computer vision (CV), natural language processing (NLP), speech recognition, and recommendation systems. To train a deep learning model large labeled datasets and high computational power supported by graphics processing units (GPUs) or tensor processing units (TPUs), are often required [6].

ML and DL approaches are equally beneficial in allowing computing machines to learn from data. Machine learning offers a number of intelligence approaches, while deep learning has capacity to solve complex problems and produce state of the art results different domains [6,7], especially in fields such as image recognition, natural language processing, and speech analysis [6].

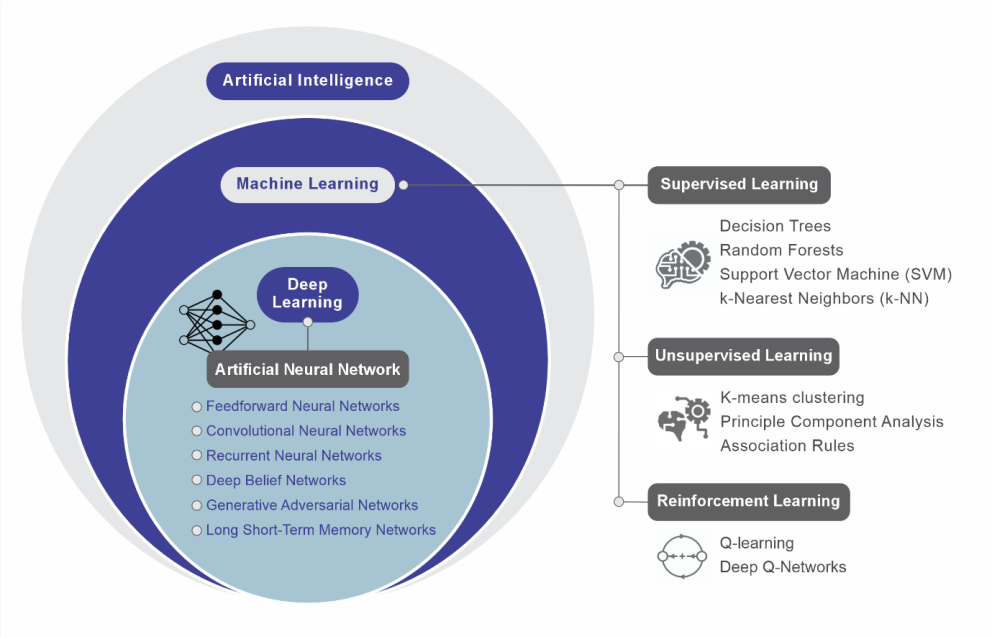

Figure 1. *Machine learning and its most popular subfields*

Methodology

An exhaustive literature review was conducted using a number of academic databases, including Google Scholar, Scopus, Web of Science, and IEEE Xplore. A large range of keywords were employed to confirm comprehensive coverage of the topic, such as "Machine Learning," "Deep Learning," "Comparison of Machine Learning and Deep Learning Methods," "Applications of Machine Learning," "Applications of Deep Learning," "Advantages of Machine Learning," "Advantages of Deep Learning Algorithms," and "Types of Machine Learning and Deep Learning Approaches."
The primary search returned many hundred research papers. However, studies were selected based on the specific standards: they focused on the comparison of machine learning and deep learning methods, were published in the English language, and fell within the publication period from 2015 to 2025. Information extracted from chosen studies were differences among ML and DL algorithms, advantages and disadvantages, their applications, appropriateness for diverse types of datasets, and future research viewpoints. This information was described in the current study in user-friendly way so that the researchers and relevant community may understand them easily.

Literature Review

During the past few years, the comparison between ML and DL algorithms proved to be an important research AI domain. In all sectors such as healthcare, finance, robotics, image processing, data augmentation and NLP, researchers have strived to show the most accurate, efficient and robust algorithms, and the scenarios where one AI method may outclass the other.

In the field of healthcare, due to their hierarchical feature extraction capabilities, DL models outperformed traditional ML algorithms in image-based diagnostics such as tumor detection and lesion segmentation [8]. In scenarios involving structured data like electronic health records ML algorithms are more competitive. [9].

In case of NLP, transformer-based DL models such as BERT and GPT have made computers very powerful and intelligent, mainly in tasks like machine translation and text summarization. Conversely, ML models are important in text classification and sentiment analysis, especially when powerful computing machines are not required [10].

In financial domain, fraud detection and real-time transactions were made easy and efficient through DL approaches, with their capability to learn complex patterns automatically. On the other hand, ML methods are very useful to evaluate credit risk [11]. In e-commerce, hybrid methods where ML's collaborative filtering and DL's feature extraction capabilities are combined, have shown remarkable success in personalizing user recommendations [12].

In robotics, DL methods, have allowed more autonomous and adaptive robot behaviors, including grasping and real-time object recognition. ML has crucial role in decision-making, path-planning, and sensor data fusion mainly in resource-constrained settings [13].

DL approaches are helping a lot in environmental monitoring, especially RNNs are popular for sensing environmental changes and forecasting weather conditions, while ML helps in conservation studies and tells about the pollution levels, biodiversity forms, and acoustic signals [14].

The existing studies have either reported the strengths/ limitations of ML and DL or highlighted narrow application areas, our study offers a comprehensive evaluation of a number of ML and DL approaches. Unlike prior studies that didn't focus on cross-domain intuitions, our study presents a detailed application-specific evaluation, augmented with structured data that map algorithmic capabilities and performance metrics.

This manuscript provides in depth and a side-by-side comparative framework, that can help ay person in choosing the most suited algorithms based on domain restraints, data availability, and performance requirements.

Applications of Machine Learning and Deep Learning Algorithms

The businesses world has been transformed by ML and DL through smarter decision-making and accurate predictions. In computer vision, both ML and DL are playing significant roles in surveillance, autonomous vehicles and robotics [15,16]. DL empowers self-driving vehicles, to detect objects, identify lanes, and make real time decisions. Chatbots and virtual assistant have been developed using ML and DL methods that have enabled machines to respond to human language. ML plays an important role in sentiment analysis, helping interpret buyer feedback and social media views in real time. DL models increase machine translation, allowing for precise, real-time language conversion. In healthcare, ML and DL are priceless for analyzing huge datasets such as genomic data and medical images. DL models, especially CNNs, showed central role in tasks like tumor detection and radiology understanding, whereas ML helps disease prediction and personalized treatment planning using electronic health records. In finance, ML is useful to assess credit risk and helps making decisions, while DL detects intricate fraud patterns and supports real-time trading. In e-commerce, ML improves recommendation systems by analyzing user habits and purchase history, whereas content and collaborative filtering progress engagement through personalized recommendations. Robotics has also seen major progressions, with DL approaches refining grasping and object management [17]. Finally,

in environmental monitoring, ML predict weather, evaluate pollution levels, and track wildlife. From examining satellite images to understanding acoustic data for conservation, ML and DL continue to enlarge their role in sustainability [18–21]. These applications reveal the powerful impact of ML and DL. A detailed summary of their roles across a number of industries is presented in Table 1.

Table 1. *Applications of Machine Learning and Deep Learning Algorithms*

Domains	**Machine Learning**	**Deep Learning**
Computer Vision	Detecting images Segmenting images Tracking objects	Classifying images Recognizing faces Autonomous vehicles
Robotics	Recognizing objects Planning the path	Manipulation and grasping Navigation
Finance	Credit scoring Algorithmic trading Optimizing portfolio	Detecting fraud Assessing risks
Natural Language Processing	Sentiment analysis Recognizing named entity Classifying text and documents	Machine translation Chatbots Speech recognition
Environmental Monitoring	Weather forecasting Conserving wildlife	Monitoring pollution Assessing environmental impact
Healthcare	Diagnosing diseases Forecasting personalized medicine Forecasting patient risk	Analyzing medical images Forecasting in drug discovery Electronic health records (EHRs)
Recommendation Systems	Recommending items Personalized content delivery	Content & collaborative filtering

Overview of Popular Machine Learning Algorithms

This section describes famous supervised learning, unsupervised and reinforcement learning algorithms.

Supervised Learning Algorithms

In supervised learning an algorithm is trained on a labeled dataset, which means that each input data point is accompanied with a corresponding output label. It learns a mapping from input features to the target output based on the provided labeled data (Figure 2).

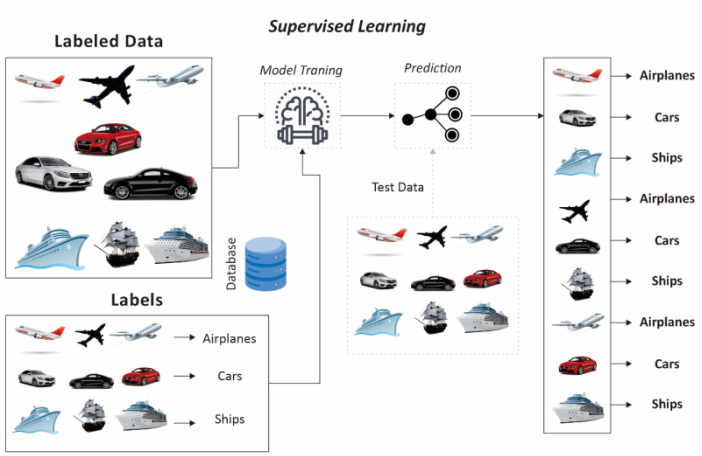

***Figure 2**. Supervised learning workflow illustrating model training with labeled data and prediction on unseen test data.*

Decision Trees

Decision trees (DTs) are supervised machine learning methods that use a tree-like structure to make predictions based on input features. They work through simple if-else rules, making them easy to understand and widely used for both classification (e.g., spam detection) and regression tasks (e.g., predicting house prices) [22]. Built using a top-down approach called recursive partitioning, each node in the tree splits data based on a chosen feature and threshold until a stopping point—like tree depth or minimum samples—is reached. DTs also help diagnosing diseases using patient data, and serve as the basis for powerful ensemble models like random forests and gradient boosting. Moreover, they are useful for feature selection and anomaly detection by highlighting important variables and detecting outliers in the data [23].

Pros and Cons of Decision Trees

DTs are renowned for their simplicity and the ability to work with both the numerical and text data. However, they are not valuable for complicated datasets as overfitting can be witnessed. Pruning approaches or ensemble methods can be used to minimize the problem of overfitting [22].

Random Forests

Random Forests (RF), also called as ensemble learning methods, build several decision trees using various subsets of data and features, an approach known as bagging. By merging the outputs of these trees through voting for classification or averaging for regression, RF improves accuracy and reduces overfitting [28–31]. Their built-in randomness makes them effective for handling high-dimensional data. RFs are widely used for classification, regression, feature selection, anomaly detection, recommendation systems, financial analysis (e.g., fraud detection, credit scoring), and even tasks like image and text analysis.

Pros and Cons of Random Forests

Random forests can process complex/ high-dimensional data, utilize the collective wisdom of several decision trees and provide more accurate predictions. One of the major disadvantages of RF is its complexity compared to separate DTs. As RF contains several trees which enhances its complexity and

make them difficult to interpret. Comparatively more computational resources and training time are required by RF.

Support Vector Machines

These are supervised learning methods and classify data by finding the optimal hyperplane that separates different classes in a best way. The key idea is to maximize the margin between classes, which helps improve generalization to unseen data. Support Vector Machines (SVMs) are capable of handling both linear and non-linear data distributions by using kernel functions that transform data into higher-dimensional spaces where separation becomes easier [34,35]. SVMs have been widely applied in various fields, including image and text classification, biological analysis such as protein classification, drug discovery, and gene expression studies. They are also used in financial modeling, disease diagnosis, natural language processing, and handwriting recognition through optical character recognition (OCR) techniques [37–41].

Pros and Cons of Support Vector Machines

One of the strengths of SVMs is their capability to deal with high-dimensional and non-linear data efficiently. By recognizing a decision boundary that maximizes the distance between classes and the closest data points, they incline to be more vigorous to noise and outliers. However, SVMs can demand huge computational power, mainly with large data volumes, and may fight to ensure scalability. Another restraint is their sensitivity to outliers, which can change the position of the hyperplane and negatively influence the trustworthiness of the forecasts.

k-Nearest Neighbors (k-NN)

k-Nearest Neighbors (k-NN) is a non-parametric supervised learning approach that can be used for both classification and regression tasks. It works by recognizing the k closest points in the training dataset and using their labels for predictions either through majority voting in classification or averaging in regression [43–44]. The algorithm has applications in a variety of fields, like regression, classification, anomaly detection, recommendation systems, cluster analysis and bioinformatics [47].

Pros and Cons of k-Nearest Neighbors

The main advantage of k-NN is found in its easiness and its capability to make predictions through local patterns without any preceding assumptions about data distribution. It can sense complicated relationships and is generally robust to noise and outliers. However, it is computationally expensive as it needs loading the entire dataset and computing distances for each prediction, which becomes a bottleneck with huge datasets. It also lacks interpretability and doesn't intrinsically perform feature selection. Furthermore, choosing the right value of k is very important, too small may lead to overfitting, and too large can lead to bias and blur decision boundaries.

Table 2. *A Summary of Survey of the Selected Supervised Machine Learning Algorithms*

Parameters/ Algorithms	**DTs**	**RF**	**SVMs**	**k-NNs**
DR&P	Normal data processing	Normal data processing	Feature scaling	No specific DR & P
FE	No FE required	No FE required	FE required	FE required for balancing

				features ‘influence
C&S	Simple	Complex	Complex	Comparatively simple
TT&CR	Fast; CR is low	More time: CR is high	More time: CR is high	Fast but CR is high
Explainability	Highly explainable	Moderate interpretable	Moderate interpretable	Comparatively explainable
Data Support	text, images, and time series data	text, images, and time series data	text, images, and time series data	text, images, and time series data
Suitability for Noise and Outliers	No	Yes	Yes	No
Suitability for Large Datasets	No	Yes	Yes	No
Strengths	Can process nonlinear relationships between features, categorical and numerical data	Good for overfitting, can handle categorical and numerical data	Good for large number of features, linear and non-linear datasets.	Good for small datasets. No training phase involved
Weaknesses	No good for overfitting, cannot model complex relationship	Longer training time required, complex to understand	Longer training time as compared to other algorithms. Selection of kernel functions and hyperparameters are challenging	Not good for large datasets, poor generalization due to density of the training data. Identifying optimal value of K is complex

Abbreviations:
DR&P: Data requirements and processing
FE: Feature engineering
C&S: Complexity and scalability
TT&CR: Time to train and computational requirements

Unsupervised Learning

Unsupervised ML methods don't need labeled data (Figure 3) and are mostly used for tasks like dimensionality reduction, clustering and anomaly detection. These approaches are useful for discovering hidden patterns, discovering crucial insights, and founding the groundwork for profound analysis and informed decision-making across a number of domains.

K-means clustering

K-means clustering make groups of similar data points into dissimilar clusters by diminishing the within-cluster sum of squares. Every data point is assigned to the nearest mean, called as a centroid. The process starts with arbitrarily initialized centroids and repeatedly updates them by reallocating points to the closest centroid. K-means is extensively used because it helps expose hidden patterns, segment data, and organize large data volumes efficiently [48–50]. The intuitions gained from K-means clustering frequently support informed decision-making across different fields [53,54].

Pros and Cons of K-means Clustering

The main advantage of K-means lies in its easiness; even novice users can comprehend and apply it easily. It is also computationally efficient and performs well on huge datasets. However, it is sensitive to the primary placement of centroids, which can cause inconsistent outputs. Choosing the best number of clusters is another problem and may affect efficiency of the model. Overall, K-means is an appropriate method for uncomplicated clustering tasks but may not perform well with complex or irregular data structures.

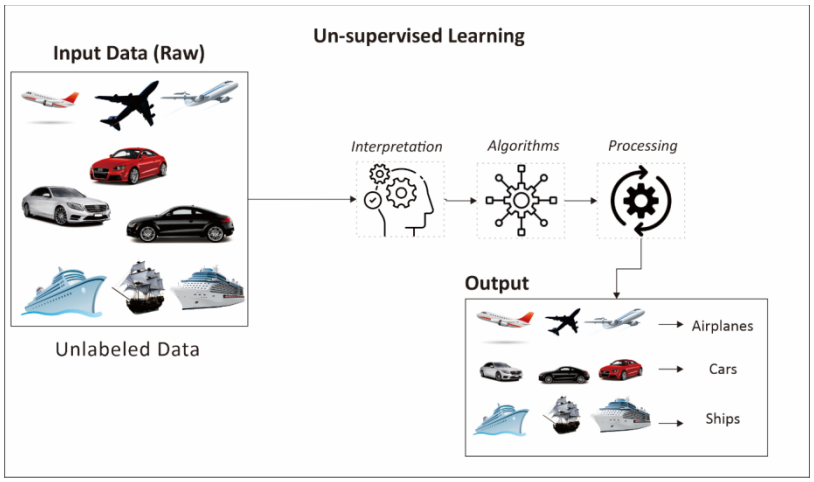

Figure 3. *Unsupervised learning workflow showing how unlabeled input data are processed by algorithms to discover patterns and form meaningful groups without prior labels.*

Principal Component Analysis

Principal Component Analysis (PCA) is a statistical method that highlights the most important features in a dataset and changes the data into a lower-dimensional space while preserving most of the original information. It attains this by calculating orthogonal axes, termed as principal components, that hold the maximum variance in the data. PCA is a famous method applied across several domains for feature extraction, dimensionality reduction, data visualization, and noise reduction, making it particularly useful in pattern recognition tasks and signal processing [55].

Pros and Cons of Principal Component Analysis

The main strength of PCA is its capability to reduce dimensionality without noteworthy loss of information [56]. This simplifies data visualization and analysis, and also reduces noise retaining relevant features. However, PCA considers linear relationships in the data, which may not always be effective. It is also sensitive to feature scaling but meaningful variations may be overlooked [55].

Association Rules

Association rules (AR) are data-mining methods that are helpful to discover relationships among items in large datasets. The objective of AR mining is to highlight recurrent item sets and produce rules that define relationships based on co-occurrence patterns. These rules are described as “if–then” statements, where the antecedent shows existing items and the consequent specifies items likely to appear together [58–62]. Association rules are applied across several domains for decision-making, pattern discovery, and optimization [60].

Pros and Cons of Association Rules

The main advantage of AR is their efficacy in market basket analysis, where they disclose customer purchasing habits and item associations. This information is useful for targeted marketing, product placement and cross-selling strategies [61]. However, AR methods do not scale well with high-dimensional data, as a large number of items can generate many irrelevant rules. In addition, data

preprocessing—such as converting data to binary form and handling missing values—can be complex and time-consuming [61,62].

Table 3. *A Summary of Survey of the Selected Un-Supervised Machine Learning Algorithms*

Parameters/ Algorithms	**K-means clustering**	**PCA**	**AR**
DR&P	No specific requirements. ND is used	ND is used	Binary data e.g., market basket data is used
FE	No FE required	No FE required	No FE required
C&S	Linear complexity may be observed	Reduces dimensionality and transforms into new space	Complexity is based on the size of data and generated rules
TT&CR	TT is fast	TT depends on the number of features	More time: CR is high
Explainability	High	High	High
Data Support	ND	ND	Binary/ transactional data
Suitability for Noise and Outliers	Not	Not	Yes
Suitability for Large Datasets	Not	Yes	Yes
Strengths	Easy to implement. Robust for spherical clusters	Minimize dimensionality, Good for data visualization	Good to identify patterns in transactional data, can deal with both categorical and binary data
Weaknesses	Needs clusters to be specified, Initialization can affect results, not good for non-linear data	Outliers may affect results significantly, unable to handle categorical data, linearity in features' relationship is assumed	High CR for large data, sometimes irrelevant rules are generated, order of items is not considered in transactions

Abbreviations:
ND: Numerical data
CR: Categorial data

Reinforcement learning

Reinforcement learning is a type of ML in which an agent learns to make accurate decisions in an environment for maximizing a reward signal. The RL's objective is to formulate a protocol that helps the agent for performing actions, while maximizing the cumulative reward over time. Unlike supervised and unsupervised ML approaches, which work with labelled or unlabeled data, RL relies on interactions with its environment (figure 4) where the agent learns through a trial-and-error process, where it performs tasks, watches the state of the environment, and obtains feedback in the form of rewards or penalties [63,64]. The RL has been successfully employed in different domains such as games, recommendation systems, robotics, and autonomous vehicles. The ability to adapt to changing environments has made RL a powerful method for learning complicated behaviors and making optimal decision-making. Major RL types are elaborated below [63-65].

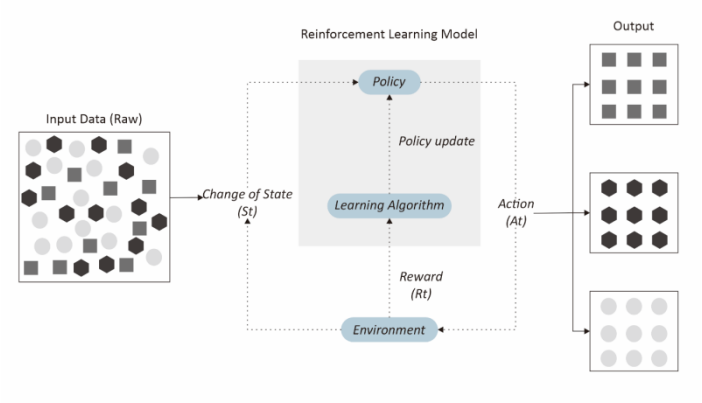

Figure 3. *Reinforcement learning workflow showing agent–environment interaction through actions, rewards, and policy updates.*

Q-learning

Q-learning (QL) is a well-known reinforcement learning method used to learn optimal action-selection strategies within a Markov Decision Process (MDP). It is a model-free algorithm, meaning it does not require prior knowledge of the environment's dynamics. QL works by iteratively updating Q-values for state–action pairs based on received rewards. Through this trial-and-error process, agents progressively acclimatize to the changing environments, make better decisions, and improve performance in complicated and dynamic settings. Its capability to acquire effective tactics makes it a widely used RL technique [66–71].

Pros and Cons of Q-learning

The main characteristic of Q-learning lies in its ability to absorb optimal policies without prior environmental data and adapt to the changing conditions. However, it faces a trade-off between exploration and exploitation, especially when rewards are scarce. Furthermore, Q-learning becomes computationally expensive when dealing with high-dimensional state spaces.

Deep Q-Networks

Deep Q-Networks (DQN) is an extension of Q-learning and integrates deep neural networks to estimated Q-values for state–action pairs, enabling them for high-dimensional state spaces. DQN advances learning stability and convergence through experience replay and target networks. The use of DQN as function approximator enables them to solve complicated problems efficiently. This method has confirmed robust performance in a variety of applications, such as game playing, robotics, NLP and autonomous vehicles [71,75].

Pros and Cons of Deep Q-Networks

A main advantage of DQN is its capability to work with high-dimensional inputs and learn directly from raw data, attaining robust outputs in complex tasks. Its ability to learn sophisticated behaviors makes it a useful method for real-world applications. However, sometimes DQNs suffer from overestimation of action values, which can cause suboptimal policies. They also require huge volume of data and substantial training time to show good performance.

Table 4. *A Summary of Survey of the Selected Un-Supervised Machine Learning Algorithms*

Parameters/ Algorithms	QL	QDN
DR&P	Well-planned state-action pairs and rewards are required	Requires raw sensor data directly from the source
FE	No FE required	Auto FE through deep neural networks
C&S	Simple	Deep neural networks enhance its complexity
TT&CR	TT is based on the number of states and intricacy of the environment	Longer TT required
Explainability	High	Low
Data Support	Various data types are supported	Various data types including images
Suitability for Noise and Outliers	Not	Yes
Suitability for Large Datasets	Yes	Yes
Strengths	Simple architecture, discrete action and state spaces can be handled	Can handle high-dimensional and complex data
Weaknesses	Explicitly defined actions, states and rewards are prerequisites, issues with high dimensional data	Needs huge datasets, more CR, faces issues during training

Overview of Popular Deep Learning Algorithms

Deep learning (DL) algorithms are subgroups of ML approaches which are designed based on the pattern of the human brain's neural networks. They are helpful in pulling out crucial patterns from intricate high-dimensional data. DL approaches are commonly used in several applications such as image classification, text mining, multimedia concept retrieval and video recommendation [77-82]. The architecture of deep learning (figure 5) methods is based on several layers of integrated artificial neurons called artificial neural networks (ANNs) which are organized into three layers i.e., an input layer, one or more hidden layers, and an output layer. The hidden layers are responsible to extract hierarchical features and representations, and enhances capability of the ANNs to learn intricate relationships in the data [83]. DL methods can automatically learn features from the given data. The provision of high-powered computing resources and huge volume of labeled data may made DL methods in processing images, object detection, sentiment analysis, speech recognition, NLP, and CV [84].

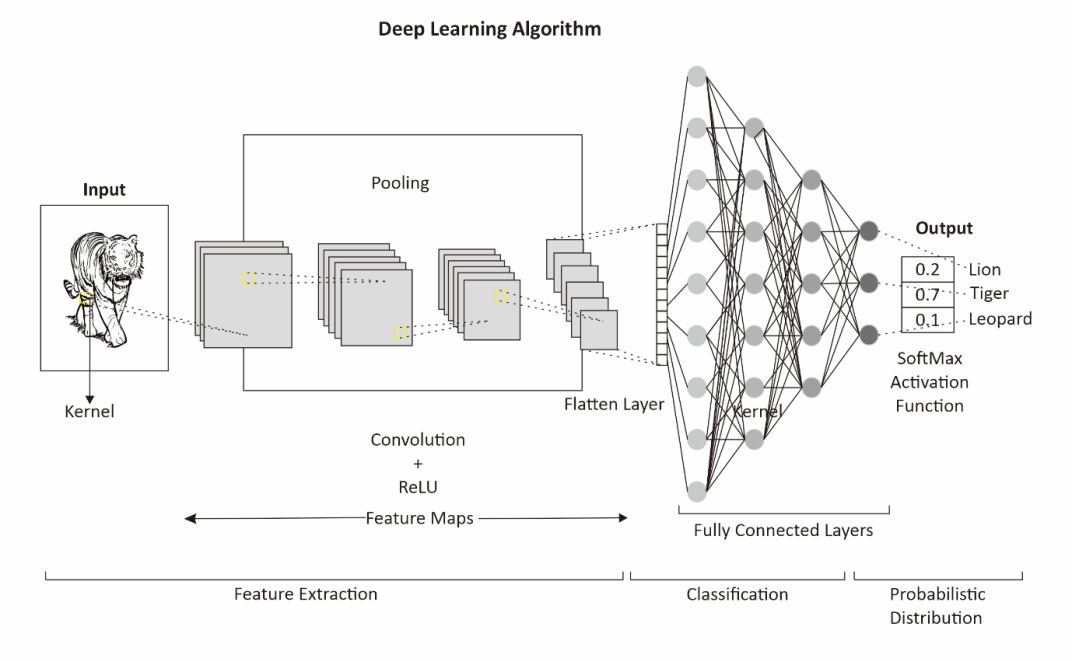

Figure 4. *Overview of a deep learning workflow showing feature extraction through convolution, ReLU, and pooling, followed by classification using fully connected layers and Softmax output.*

Pros and Cons of Deep Learning Algorithms

The key benefit of DL algorithms is that there is no need of manual feature engineering. It means they can automatically learn hierarchical characterization from raw data, enabling them to achieve very good performance in NLP, image recognition and game playing. Despite their successful applications in various domains, DL methods need huge volume of data and powerful computing materials. They can face the challenges of overfitting when training data is restricted.

Common Deep Learning Algorithms

Feedforward Neural Networks

Feedforward Neural Networks (FNNs) are types of ANNs in which data flows in one direction, from the input layer through hidden layers to the output layer. This simple technique makes FNNs effective and appropriate for real-time scenarios. They are among the most famous neural networks and are largely applied to classification and regression. FNNs are usually applied in image classification, sentiment analysis, speech recognition, financial forecasting, NLP, and drug discovery [87-91].

Pros and Cons of Feedforward Neural Networks

FNNs are easy to design and implement, and they can effectively capture complicated patterns in data. They are less inclined to over fitting when handling small datasets. However, their major restraint is the inability to model sequential data or temporal dependencies.

Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are specialized form of ANNs with the ability to process grid-like data such as images and sequences. They use convolutional filters to extract meaningful features from input data. By exploiting local receptive fields and weight sharing, CNNs significantly decrease the number of parameters and allow fast feature extraction [92–97]. These characteristics make CNNs

highly useful for image classification, segmentation, facial recognition, object detection, autonomous driving, video analysis and NLP [98–100].

Pros and Cons of CNNs

CNNs have remarkable ability to learn hierarchical features from spatially structured data and show high performance in several computers vision tasks. However, they often require huge labeled datasets and are sensitive to trivial input distortions. Furthermore, their high computational demands can be a challenge for a number of organizations.

Recurrent Neural Networks

They are specialized form of ANNs that use feedback connections to help information to flow within the network. This structure helps RNNs to retain information from previous steps, enabling them for sequential and time-series data. They can process variable-length inputs and outputs, which is mainly valuable in NLP domain such as sentence processing [101–106]. Their common applications are natural language generation, handwriting recognition, machine translation, speech recognition, time-series prediction, and sentiment analysis [96, 97].

Pros and Cons of RNNs

RNNs are useful to model temporal dependencies and sequential patterns, as they maintain hidden states that are helpful to capture contextual information. However, they are not good for vanishing and exploding gradients, which limits their capability to learn long-range dependencies. They are computationally expensive and less efficient than feedforward networks.

Deep Belief Networks

Deep Belief Networks (DBNs) comprises of several layers of interconnected nodes, naturally formed by stacking Restricted Boltzmann Machines (RBMs). They are first trained on unsupervised learning data and later fine-tuned with supervised methods. This layered structure enables DBNs to learn more and more abstract representations of data, making them valuable for catching complex patterns [92, 93]. DBNs have applications in generative modeling, anomaly detection, drug discovery, image recognition, NLP, and time-series forecasting [93].

Pros and Cons of DBNs

They are powerful generative models having capability to learn deep and abstract patterns in data, reduce dimensionality and process various data types. However, they are less efficient, computationally expensive, prone to overfitting and sensitive to hyper parameter selection [89].

Generative Adversarial Networks

Generative Adversarial Networks (GANs) has a generator and a discriminator. The generator is responsible to produce synthetic data samples, while the discriminator separates real and generated data. Through this process, GANs progressively learn to yield highly realistic and varied samples [97–98]. GANs are used in image and video synthesis, image fusion, data augmentation, music generation, and text-to-image generation [98–103].

Pros and Cons of GANs

GANs are famous for producing high-quality and realistic data, opening new potentials in creative and scientific domains. However, training GANs is difficult and needs careful tuning of architectures and hyperparameters. Issues like unstable training and mode collapse can also affect performance.

Long Short-Term Memory Networks

Long Short-Term Memory (LSTM) networks are an advanced method of RNN intended to minimize the vanishing gradient problem and capture long-term dependencies. They depend on memory cells and gating mechanisms for selectively storing and updating information over time. These characteristics make LSTMs extremely beneficial for sequential and time-series data analysis [102–105]. They are well known for time-series forecasting, handwriting and video recognition, text generation, NLP, human motion prediction and autonomous driving.

Pros and Cons of LSTM

LSTMs are very good at modeling long-term dependencies and retaining contextual information over prolonged sequences. However, they are susceptible to overfitting, have complex architectures, and need vigilant hyperparameter tuning, which can make them difficult to optimize.

Table 5. *A Summary of Survey of the Selected Un-Supervised Machine Learning Algorithms*

Parameters/ Algorithms	**FNNs**	**CNNs**	**RNNs**	**DBNs**	**GANs**	**LSTMNs**
DR&P	Labeled data, scaling and normalization	Labeled image data and some pre-processing such as image resizing are required	Sequential or time-series data and some preprocessing such as tokenization are required	Requires labelled data, scaling and normalization	Labeled data, scaling and normalization are required	Sequential or time-series and data tokenization
FE	Manual FE is required	Automatically learns from raw data	Has the ability to extract temporal dependencies in the data	Automatically learns unsupervised pretraining	Automatically learns	Automatically learns
C&S	Based on the number of layers and neurons	Based on the number of layers, filters, and receptive fields	Depends on the number of recurrent units and layers	Depends on the number of layers and hidden units	Depends on the number of generator and discriminator layers	Depends on the number of LSTM units and layers
TT&CR	Large networks require high TT &CR	Large networks require high TT &CR	Deep networks require high TT whereas large datasets require high CR	High during unsupervised pretraining	TT & CR can be high	TT be high for deep networks and long sequences and requires substantial CR
Explainability	Low	Low	Low	Moderate	Low	Low

Data Support	Good for structured data	Suitable for image data	Good for sequential and time-series data	Famous for structured data	Suitable for generating new data samples	Suitable for sequential and time-
Suitability for Noise and Outliers	Not	Yes	Not	Yes	Not	Very good
Suitability for Large Datasets	Overfitting my decreased performance	Yes	Not	Not (Overfitting is difficult)	Not	Not
Strengths	Has a wide range of applications	Good for image and computer vision tasks	Famous for NLP and sequential tasks	Good for unsupervised learning and feature representation	Effective for generating realistic data samples	Effective for sequential tasks, such as natural
Weaknesses	Lack of temporal awareness, not good for sequential data	Not suitable for textual or time series data	Not good in capturing long-term dependencies and prone to gradient vanishing/exploding	Limited availability of pretraining techniques and is computationally expensive	Difficult to train and stabilize	Captures long-term dependencies, mitigates vanishing/exploding

Guidelines for selecting between machine learning and deep learning algorithms

The proposed study based on the investigation of various aspects of ML and DL methods will be useful for researchers and decision-makers to choose the most appropriate ML method specific to their needs, environment and context. This study highlighted that while selecting between ML/ DL methods, guidelines shown in Table 5 may be considered.

Table 6. *Guidelines for selecting ML or DL methods*

Parameter	Machine Learning	Deep Learning
Complicated and high-dimensional datasets		✓
For automatic feature engineering		✓
To avoid model complexity	✓	
For scalability		✓
Interpretability and Explainability	✓	
To avoid high-level expertise and experience	✓	

Less time and limited resources	✓	
Baseline Performance	Investigate accuracies, error rates and other parameters to select between ML and DL methods	
Domain-specific problems	DL algorithms are really good for some domains such as NLP and image classification whilst other problems may be solved effectively by ML approaches	
Hybrid Approaches	For achieving better results, sometimes ML and DL algorithms may be combined. For a specific part of the job ML may be used and for other parts of the job DL may perform better	

Future trends and challenges

While exploring the future of ML & DL, multiple challenges and trends are projected to transform the AI domain. Firstly, very large and multifarious datasets are expected to be emphasized, which will enable AI models to learn from a wide range of examples and enhance generalization. The evaluation and on-device training will facilitate privacy-preserving and decentralized ML systems. Simultaneously, the ML algorithms will be focused on being deployed on resource-constrained devices, encouraging the design and development of lightweight and effective algorithms. Secondly, the development of interpretable ML methods will make it easy to handle to build trust and ensure transparency in areas such as healthcare, finance, and autonomous vehicles. Thirdly, ML merging with other areas such as CV, NLP and robotics will revolutionize human life. Integrating diverse information such as text, images, and audio generally termed as Multi-modal learning, will help AI systems to present deep understanding of the environment and human interactions.

However, ML also has multiple challenges. Intentionally manipulating data to deceive ML methods, called as Adversarial attacks, is a vital challenge. Developing tools and techniques for automatics feature extraction and getting rid of labelled data, is an expensive job, and another ongoing challenge. Ethical concerns related to data privacy and security, partiality, and fairness is a very important factor to be focused. In conclusion, the future of ML & DL promises enormous prospects for life-changing developments. However, realizing this potential will need resolving the challenges and staying vigilant about ethical concerns to develop AI systems that positively impact society.

Discussion

This study reports that ML and DL are vital in all aspects of our lives like NLP, image processing, computer vision, robotics, finance, education and environment monitoring [15, 16]. Various studies performed on evaluation of efficiency and accuracy of the algorithms showed that performance is linked with data types, size of datasets, and available computing resources. Importance of ML and DL approaches is highly observable when an organization is faced with high-dimensional and unstructured data. In such cases, CNNs are very useful due to their capability to capture hierarchical and complex patterns automatically from raw data [92–97,101–106]. This shows significance of DL for object detection, image classification, speech recognition, and machine translation, especially when manual feature extraction becomes very cumbersome. [15,16]. In spite of a large number of important applications of DL, ML reported its competitiveness in several structured-data settings, where datasets are smaller and feature extraction is easy, such as disease prediction, credit scoring and classical sentiment analysis [15,16]. These findings show that DL is beneficial for the scenarios where data is complex & highly dimensional and feature extraction is difficult, whereas ML is applicable when data is small, easy to interpret, and computing resources are limited.

DTs are very famous methods for handling large datasets, however they may cause overfitting which makes them less reliable for complex datasets unless pruning or ensemble approaches are applied [22].

Random forests use bagging and aggregation approaches to resolve the problem of overfitting and enhance robustness and accuracy [28–31]. SVMs are also applicable for high-dimensional and non-linear datasets [34, 35], which makes them suitable for analyzing text, image, and biological applications [37–41]. k-NN are useful for instance-based learning and for lesser datasets but are computationally expensive and sensitive to choices like distance metrics and the value of *k* [43–44].
Unsupervised learning approaches have key role in case of un-labelled datasets. K-means attain popularity due its simplicity and efficiency and is useful for segmentation tasks. However, its reliability is limited for irregular data distributions [48–50, 53, 54]. PCA is used for dimensionality reduction, visualization, enhance downstream analysis, but its linearity assumptions and sensitivity to scaling can miss meaningful information [55, 56]. For transactional data and pattern finding in market-basket-type environments, association rule mining may be very useful [58–62], but it generates low-value rules which can cause scaling problem as dimensionality raises [61, 62].
Reinforcement learning is applicable for sequential decision-making. Q-learning enables agents to learn through trial-and-error without explicit environment models [66–71], which makes it useful in dynamic application settings [63,64]. Deep Q-Networks are an extension of Q-learning and manage high-dimensional inputs by approximating value functions and perform well in more complex environments. Large datasets, stable training methods, and high computational resources are required by deep Q-networks [71, 75].
Simple architecture of FNNs makes them useful for supervised problems, but their performance is not well for temporal dependencies [87-91]. Local receptive fields and weight sharing enable CNNs good for spatial data processing, making them supreme in computer vision tasks like autonomous driving and medical imaging [92–97, 98–100]. RNNs and LSTMs are good for sequential modeling in language and time series. Comparing to RNNs, LSTMs are more capable to mitigate vanishing gradient problem [101–106,102–105]. DBNs and GANs are a type of DL's extension and have a key role synthetic data generation, though both face the drawbacks of high computational cost, hyperparameter sensitivity and stability [92, 93, 97–98].
Finally, the selection guidelines and future trends highlight that benchmark accuracy helps a lot in real-world adoption. Interpretability is more important in high risks fields such as healthcare and finance, inspiring continued focus on explainable and trustworthy models. Overall, the evidence presented by the current study supports a conclusion: ML and DL are the complementary toolsets, where algorithm choice is guided by data characteristics, deployment restraints, and hybrid approaches can be beneficial when various parts of a pipeline benefit from different modeling strengths.

Conclusion

The proposed study considered several aspects for investigating the performance and characteristics of some popular ML & DL algorithms. A survey of this study reported multiple findings such as most ML approaches require feature engineering and preprocessing steps whilst DL methods can automatically learn features from raw input data. Compared to ML algorithms, DL methods are more complex with a significantly large number of parameters, and need longer training times, so, they require high-powered computing machines, but they are able to handle high-dimensional data and extract complex patterns. Regarding explainability and interpretability, ML algorithms are easier to understand and interpret compared to DL methods which are generally termed as black boxes. In the case of data types, both ML & DL methods accept various data types, including images, text and time series data, however DL approaches, can process raw sensory data and are famous for processing complex data types such as audios/videos and images. With respect to capability for handling noise and outliers in the data, DL approaches are more robust to handle noise and outliers in the input data compared to ML algorithms. DL algorithms exhibit better performance on large-scale datasets whereas ML approaches are prone to scalability issues with large datasets. In conclusion, the choice between the ML&DL approaches depends on specific requirements, scenarios, environments, available resources, and the trade-offs between the two types of AI methods.

References

[1] Taye MM. (2023). Understanding of machine learning with deep learning: Architectures, workflow, applications and future directions. Computers. 12(5):91.

[2] Benos L, Tagarakis AC, Dolias G, Berruto R, Kateris D, Bochtis D. (2021). Machine learning in agriculture: A comprehensive updated review. Sensors. 21(11):3758.

[3] Sharma D, Kumar N. (2017). A review on machine learning algorithms, tasks and applications. International Journal of Advanced Research in Computer Engineering & Technology. 6(10):2278–1323.

[4] Ray S. (2019). A quick review of machine learning algorithms. In: Proceedings of the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon). IEEE. pp.35–39.

[5] Mahesh B. (2020). Machine learning algorithms: A review. International Journal of Science and Research. 9(1):381–386.

[6] Shinde PP, Shah S. (2018). A review of machine learning and deep learning applications. In: Proceedings of the 4th International Conference on Computing Communication Control and Automation (ICCUBEA). IEEE. pp.1–6.

[7] Sharma N, Sharma R, Jindal N. (2021). Machine learning and deep learning applications: A vision. Global Transitions Proceedings. 2(1):24–28.

[8] Mangrolia JR. (2025). Revolutionizing healthcare through artificial intelligence: The role of machine learning, deep learning and NLP in transforming patient care. Deep Science Publishing.

[9] Rafee SM, Prasad M, Kumar MS, Easwaran B. (2023). AI technologies, tools, and industrial use cases. Toward Artificial General Intelligence: Deep Learning, Neural Networks, Generative AI.

[10] Chinnaiyan B, Balasubaramanian S, Jeyabalu M, Warrier GS. (2025). AI Applications–Computer Vision and Natural Language Processing. Model Optimization Methods for Efficient and Edge AI. pp.25–41.

[11] Rane N, Choudhary S, Rane J. (2024). Artificial intelligence and machine learning in business intelligence, finance, and e-commerce: a review. Finance and E-Commerce Review.

[12] Divyajyothi MG, Jopate R. (2025). Latest frontiers of machine, deep, and reinforcement learning algorithms. In: AI Integration for Business Sustainability. Springer Nature Singapore. pp.357–371.

[13] Fu RJC. (2025). The future of AI, ML, and robotics in business and beyond. AIMLR Journal.

[14] Titov K. (2024). Introduction to artificial intelligence: Understanding the basics. Konstantin Titov.

[15] Sarker IH. (2021). Deep learning: a comprehensive overview on techniques, taxonomy,applications and research directions. SN Computer Science. 2(6):420.

[16] Dargan S, Kumar M, Ayyagari MR, Kumar G. (2020). A survey of deep learning and its applications: a new paradigm to machine learning. Archives of Computational Methods in Engineering. 27:1071–1092.

[17] Chauhan NK, Singh K. (2018). A review on conventional machine learning vs deep learning. In: Proceedings of the International Conference on Computing, Power and Communication Technologies (GUCON). IEEE. pp.347–352.

[18] Mahmud M, Kaiser MS, Hussain A, Vassanelli S. (2018). Applications of deep learning and reinforcement learning to biological data. IEEE Transactions on Neural Networks and Learning Systems. 29(6):2063–2079.

[19] Song YY, Ying LU. (2015). Decision tree methods: applications for classification and prediction. Shanghai Archives of Psychiatry. 27(2):130.

[20] Hou Y. (2018). Decision tree algorithm for big data analysis. In: Proceedings of the International Conference on Transportation & Logistics, Information & Communication, Smart City (TLICSC). Atlantis Press. pp.270–274.

21] Pang Z. (2018). Research on application of data collection and data mining in mobile communication network. Beijing Jiaotong University.

[22] Zhang F. (2016). Research on classification mining algorithm of stream data based on STORM. Nanjing University of Posts and Telecommunications.

[23] Hou L. (2016). Application study of decision tree algorithm in engineering quality supervision decision support system. Guizhou University.

[24] Li M. (2016). Application of decision tree algorithm in bank telemarketing. Huazhong University of Science and Technology.

[25] Dai Y. (2015). Analysis of decision tree algorithm in data mining and its application. Science and Technology Communication. 7(23):33–34.

[26] Orlenko A, Moore JH. (2021). A comparison of methods for interpreting random forest models of genetic association in the presence of non-additive interactions. BioData Mining. 14(1):1–7.

[27] Speiser JL, Miller ME, Tooze J, Ip E. (2019). A comparison of random forest variable selection methods for classification prediction modeling. Expert Systems with Applications. 134:93–101.

[28] Gatera A, Kuradusenge M, Bajpai G, Mikeka C, Shrivastava S. (2023). Comparison of random forest and support vector machine regression models for forecasting road accidents. Scientific African. 21:e01739.

[29] Speiser JL, Durkalski VL, Lee WM. (2015). Random forest classification of etiologies for an orphan disease. Statistics in Medicine. 34(5):887–899.

[30] Ahmad I, Basheri M, Iqbal MJ, Rahim A. (2018). Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection. IEEE Access. 6:33789–33795.

[31] Kusumawati R, D'arofah A, Pramana PA. (2019). Comparison performance of naive bayes classifier and support vector machine algorithm for Twitter classification. Journal of Physics: Conference Series. 1320(1):012016.

[32] Valero-Carreras D, Alcaraz J, Landete M. (2023). Comparing two SVM models through different metrics based on the confusion matrix. Computers & Operations Research. 152:106131.

[33] Apsemidis A, Psarakis S. (2020). Support vector machines: A review and applications in statistical process monitoring. Data Analysis and Applications 3. 5:123–144.

[34] Roy A, Chakraborty S. (2023). Support vector machine in structural reliability analysis: A review. Reliability Engineering & System Safety. 109126.

[35] Feng J, Liu L, Wu D, Li G, Beer M, Gao W. (2019). Dynamic reliability analysis using the extended support vector regression (X-SVR). Mechanical Systems and Signal Processing. 126:368–391.

[36] Xu Z, Saleh JH. (2021). Machine learning for reliability engineering and safety applications: Review of current status and future opportunities. Reliability Engineering & System Safety. 211:107530.

[37] Roy A, Manna R, Chakraborty S. (2019). Support vector regression based metamodeling for structural reliability analysis. Probabilistic Engineering Mechanics. 55:78–89.

[38] Setyorini SG. (2021). Application of the nearest neighbor algorithm for classification of online taxibike sentiments in Indonesia. Journal of Physics: Conference Series. 2049(1):012026.

[39] Suyal M, Goyal P. (2022). A review on analysis of K-nearest neighbor classification machine learning algorithms. International Journal of Engineering Trends and Technology. 70(7):43–48.

[40] Bhardwaj CA, Mishra M, Desikan K. (2018). Dynamic feature scaling for k-nearest neighbor algorithm. arXiv:1811.05062.

[41] Anava O, Levy K. (2016). k*-nearest neighbors: From global to local. Advances in Neural Information Processing Systems. 29.

[42] Ikotun AM, Ezugwu AE, Abualigah L, Abuhaija B, Heming J. (2023). K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data. Information Sciences. 622:178–210.

[43] Oti EU, Olusola MO, Eze FC, Enogwe SU. (2021). Comprehensive review of K-means clustering algorithms. Criterion. 12:22–23.

[44] Chong B. (2021). K-means clustering algorithm: A brief review. Vol.4:37–40.

[45] Jothi R, Mohanty SK, Ojha A. (2019). DK-means: A deterministic k-means clustering algorithm for gene expression analysis. Pattern Analysis and Applications. 22:649–667.

[46] Narang B, Verma P, Kochar P. (2016). Application based advantageous K-means clustering algorithm in data mining: A review. International Journal of Latest Trends in Engineering and Technology. 7(2):121–126.

[47] Jolliffe IT, Cadima J. (2016). Principal component analysis: A review and recent developments. Philosophical Transactions of the Royal Society A. 374(2065):20150202.

[48] Gao J. (2021). Research on application of improved association rules mining algorithm in personalized recommendation. Journal of Physics: Conference Series. 1744(3):032111.

[49] Diaz-Garcia JA, Ruiz MD, Martin-Bautista MJ. (2023). A survey on the use of association rules mining techniques in textual social media. Artificial Intelligence Review. 56(2):1175–1200.

[50] Fister I Jr, Fister I, Fister D, Podgorelec V, Salcedo-Sanz S. (2023). Visualization methods for association rule mining: A comprehensive review. arXiv:2302.12594.

[51] Lee S, Cha Y, Han S, Hyun C. (2019). Application of association rule mining and social network analysis for understanding causality of construction defects. Sustainability. 11(3):618.

[52] Nian R, Liu J, Huang B. (2020). A review on reinforcement learning: Introduction and applications in industrial process control. Computers & Chemical Engineering. 139:106886.

[53] Bhatia A, Varakantham P, Kumar A. (2019). Resource constrained deep reinforcement learning. Proceedings of the International Conference on Automated Planning and Scheduling. 29:610–620.

[54] Avranas A, Ciblat P, Kountouris M. (2023). Deep reinforcement learning for resource constrained multiclass scheduling in wireless networks. IEEE Transactions on Machine Learning in Communications and Networking.

[55] Clifton J, Laber E. (2020). Q-learning: Theory and applications. Annual Review of Statistics and Its Application. 7:279–301.

[56] Rolf B, Jackson I, Müller M, Lang S, Reggelin T, Ivanov D. (2023). Reinforcement learning algorithms and applications in supply chain management: A review. International Journal of Production Research. 61(20):7151–7179.

[57] Bahrpeyma F, Reichelt D. (2022). Applications of multi-agent reinforcement learning in smart factories: A review. Frontiers in Robotics and AI. 9:1027340.

[58] Li Q, Lin T, Yu Q, Du H, Li J, Fu X. (2023). Review of deep reinforcement learning and its application in modern renewable power system control. Energies. 16(10):4143.

[59] Jang B, Kim M, Harerimana G, Kim JW. (2019). Q-learning algorithms: A comprehensive c lassification and applications. IEEE Access. 7:133653–133667.

[60] Ahn KU, Park CS. (2020). Application of deep Q-networks for model-free optimal control balancing between different HVAC systems. Science and Technology for the Built Environment. 26(1):61–74.

[61] Chan KY, Abu-Salih B, Qaddoura R, Ala'M AZ, Palade V, Pham DS, Del Ser J, Muhammad K. (2023). Deep neural networks in the cloud: Review, applications, challenges and research directions. Neurocomputing.

[62] Wang N, Li Z, Liang X, Hou Y, Yang A. (2023). A review of deep reinforcement learning methods and military application research. Mathematical Problems in Engineering.

[63] Zhao X, Wang G. (2023). Deep Q networks-based optimization of emergency resource scheduling for urban public health events. Neural Computing and Applications. 35(12):8823–8832.

[64] Alzubaidi L, Zhang J, Humaidi AJ, Al-Dujaili A, Duan Y, Al-Shamma O, Santamaría J, Fadhel MA, Al-Amidie M, Farhan L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. Journal of Big Data. 8:1–74.

[65] Rozenwald MB, Galitsyna AA, Sapunov GV, Khrameeva EE, Gelfand MS. (2020). Machine learning framework for prediction of chromatin folding. PeerJ Computer Science. 6:e307.

[66] Amrit C, Paauw T, Aly R, Lavric M. (2017). Identifying child abuse through text mining and machine learning. Expert Systems with Applications. 88:402–418.

[67] Hossain E, Khan I, Un-Noor F, Sikander SS, Sunny MS. (2019). Big data and machine learning in smart grid applications: A review. IEEE Access. 7:13960–13988.

[68] Crawford M, Khoshgoftaar TM, Prusa JD, Richter AN, Al Najada H. (2015). Survey of review spam detection using machine learning techniques. Journal of Big Data.

[69] Deldjoo Y, Elahi M, Cremonesi P, Garzotto F, Piazzolla P, Quadrana M. (2016). Content-based video recommendation using stylistic visual features. Journal on Data Semantics. 5:99–113.

[70] Al-Dulaimi K, Chandran V, Nguyen K, Banks J, Tomeo-Reyes I. (2019). Benchmarking HEp-2 specimen cells classification using LDA. Pattern Recognition Letters. 125:534–541.

[71] Liu W, Wang Z, Liu X, Zeng N, Liu Y, Alsaadi FE. (2017). Survey of deep neural network architectures and applications. Neurocomputing. 234:11–26.

[72] Pouyanfar S, Sadiq S, Yan Y, Tian H, Tao Y, Reyes MP, Shyu ML, Chen SC, Iyengar SS. (2018). A survey on deep learning: Algorithms, techniques, and applications. ACM Computing Surveys. 51(5):1–36.

[73] Alom MZ, Taha TM, Yakopcic C, Westberg S, Sidike P, Nasrin MS, Hasan M, Van Essen BC, Awwal AA, Asari VK. (2019). State-of-the-art survey on deep learning theory and architectures. Electronics. 8(3):292.

[74] Potok TE, Schuman C, Young S, Patton R, Spedalieri F, Liu J, Yao KT, Rose G, Chakma G. (2018). Complex deep learning networks on high-performance and neuromorphic computers. ACM Journal on Emerging Technologies in Computing Systems. 14(2):1–21.

[75] Ojha VK, Abraham A, Snášel V. (2017). Metaheuristic design of feedforward neural networks: A review. Engineering Applications of Artificial Intelligence. 60:97–116.

[76] Georgiopoulos M, Li C, Kocak T. (2011). Learning in the feed-forward random neural network: A critical review. Performance Evaluation. 68(4):361–384.

[77] Laudani A, Lozito GM, Fulginei FR, Salvini A. (2015). Training efficiency and computational costs of feedforward neural networks. Computational Intelligence and Neuroscience.

[78] Alzubaidi L, Zhang J, Humaidi AJ, Al-Dujaili A, Duan Y, Al-Shamma O, Santamaría J, Fadhel MA, Al-Amidie M, Farhan L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. Journal of Big Data. 8:1–74.

[79] Zhou DX. (2020). Theory of deep convolutional neural networks: Downsampling. Neural Networks. 124:319–327.

[80] Jhong SY, Tseng PY, Siriphockpirom N, Hsia CH, Huang MS, Hua KL, Chen YY. (2020). Automated biometric identification using CNN-based palm vein recognition. Proceedings of the International Conference on Advanced Robotics and Intelligent Systems (ARIS). IEEE.

[81] Al-Azzawi A, Ouadou A, Max H, Duan Y, Tanner JJ, Cheng J. (2020). DeepCryoPicker: Fully automated deep neural network for single protein particle picking in cryo-EM. BMC Bioinformatics. 21:1–12.

[82] Wang T, Lu C, Yang M, Hong F, Liu C. (2020). A hybrid method for heartbeat classification via CNNs, multilayer perceptrons and focal loss. PeerJ Computer Science. 6:e324.

[83] Li G, Zhang M, Li J, Lv F, Tong G. (2021). Efficient densely connected convolutional neural networks. Pattern Recognition. 109:107610.

[84] Gu J, Wang Z, Kuen J, Ma L, Shahroudy A, Shuai B, Liu T, Wang X, Wang G, Cai J, Chen T. (2018). Recent advances in convolutional neural networks. Pattern Recognition. 77:354–377.

[85] Fang W, Love PE, Luo H, Ding L. (2020). Computer vision for behaviour-based safety in construction: A review and future directions. Advanced Engineering Informatics. 43:100980.

[86] Palaz D, Magimai-Doss M, Collobert R. (2019). End-to-end acoustic modeling using convolutional neural networks for HMM-based automatic speech recognition. Speech Communication. 108:15–32.

[87] Li HC, Deng ZY, Chiang HH. (2020). Lightweight and resource-constrained learning network for face recognition with performance optimization. Sensors. 20(21):6114.

[88] Yu Y, Si X, Hu C, Zhang J. (2019). A review of recurrent neural networks: LSTM cells and network architectures. Neural Computation. 31(7):1235–1270.

[89] Chen Y, Li J. (2021). Recurrent neural networks algorithms and applications. Proceedings of the International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE). IEEE. pp.38–43.

[90] Sherstinsky A. (2020). Fundamentals of recurrent neural network and long short-term memory network. Physica D: Nonlinear Phenomena. 404:132306.

[91] Mao S, Sejdić E. (2022). Recurrent neural network-based methods in computational physiology: A review. IEEE Transactions on Neural Networks and Learning Systems.

[92] Ackerson JM, Dave R, Seliya N. (2021). Applications of recurrent neural networks for biometric authentication and anomaly detection. Information. 12(7):272.

[93] Movahedi F, Coyle JL, Sejdić E. (2017). Deep belief networks for electroencephalography: A review. IEEE Journal of Biomedical and Health Informatics. 22(3):642–652.

[94] Yang J, Chang B, Wang X, Zhang Q, Wang C, Wu M. (2020). Design and application of deep belief network based on stochastic adaptive particle swarm optimization. Mathematical Problems in Engineering.

[95] Sohn I. (2021). Deep belief network based intrusion detection techniques: A survey. Expert Systems with Applications. 167:114170.

[96] Yin P. (2021). Modified deep belief network model and its application. Journal of Physics: Conference Series. 1769(1):012025.

[97] Alqahtani H, Kavakli-Thorne M, Kumar G. (2021). Applications of generative adversarial networks: An updated review. Archives of Computational Methods in Engineering. 28:525–552.

[98] Dash A, Ye J, Wang G. (2021). Review of generative adversarial networks and applications. arXiv:2110.01442.

[99] Asimopoulos DC, Nitsiou M, Lazaridis L, Fragulis GF. (2022). Generative adversarial networks: A systematic review and applications. SHS Web of Conferences. 139:03012.

[100] Gui J, Sun Z, Wen Y, Tao D, Ye J. (2021). Review on generative adversarial networks: Algorithms, theory, and applications. IEEE Transactions on Knowledge and Data Engineering. 35(4):3313–3332.

[101] Hossam H, Elgmmal E, Elnabawy RH. (2022). Review of generative adversarial networks applications. Proceedings of the Novel Intelligent and Leading Emerging Sciences Conference (NILES). IEEE. pp.142–146.

[102] Van Houdt G, Mosquera C, Nápoles G. (2020). Review on the long short-term memory model. Artificial Intelligence Review. 53:5929–5951.

[103] Abed M, Imteaz MA, Ahmed AN, Huang YF. (2021). Application of LSTM neural network technique for predicting monthly pan evaporation. Scientific Reports. 11:20742.

[104] Lindemann B, Müller T, Vietz H, Jazdi N, Weyrich M. (2021). Survey on long short-term memory networks for time series prediction. Procedia CIRP. 99:650–655.

[105] Ghislieri M, Cerone GL, Knaflitz M, Agostini V. (2021). Long short-term memory recurrent neural network for muscle activity detection. Journal of NeuroEngineering and Rehabilitation. 18:1–12.

[106] Le XH, Ho HV, Lee G, Jung S. (2019). Application of long short-term memory neural network for flood forecasting. Water. 11(7):1387.