SN Computer Science (2021) 2:420
https://doi.org/10.1007/s42979-021-00815-1

REVIEW ARTICLE

Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions

Iqbal H. Sarker[1,2]

Received: 29 May 2021 / Accepted: 7 August 2021 / Published online: 18 August 2021

Abstract

Deep learning (DL), a branch of machine learning (ML) and artificial intelligence (AI) is nowadays considered as a core technology of today's Fourth Industrial Revolution (4IR or Industry 4.0). Due to its learning capabilities from data, DL technology originated from artificial neural network (ANN), has become a hot topic in the context of computing, and is widely applied in various application areas like healthcare, visual recognition, text analytics, cybersecurity, and many more. However, building an appropriate DL model is a *challenging* task, due to the dynamic nature and variations in real-world problems and data. Moreover, the lack of core understanding turns DL methods into black-box machines that hamper development at the standard level. This article presents a structured and *comprehensive view* on DL techniques including a *taxonomy* considering various types of real-world tasks like supervised or unsupervised. In our taxonomy, we take into account deep networks for supervised or *discriminative learning*, unsupervised or *generative learning* as well as *hybrid learning* and relevant others. We also summarize *real-world application areas* where deep learning techniques can be used. Finally, we point out ten potential aspects for future generation DL modeling with *research directions*. Overall, this article aims to draw a big picture on DL modeling that can be used as a reference guide for both academia and industry professionals.

Keywords Deep learning · Artificial neural network · Artificial intelligence · Discriminative learning · Generative learning · Hybrid learning · Intelligent systems

Introduction

In the late 1980s, neural networks became a prevalent topic in the area of Machine Learning (ML) as well as Artificial Intelligence (AI), due to the invention of various efficient learning methods and network structures [52]. Multilayer perceptron networks trained by "Backpropagation" type algorithms, self-organizing maps, and radial basis function networks were such innovative methods [26, 36, 37]. While neural networks are successfully used in many applications, the interest in researching this topic decreased later on. After that, in 2006, "Deep Learning" (DL) was introduced by Hinton et al. [41], which was based on the concept of artificial neural network (ANN). Deep learning became a prominent topic after that, resulting in a rebirth in neural network research, hence, some times referred to as "new-generation neural networks". This is because deep networks, when properly trained, have produced significant success in a variety of classification and regression challenges [52].

Nowadays, DL technology is considered as one of the hot topics within the area of machine learning, artificial intelligence as well as data science and analytics, due to its learning capabilities from the given data. Many corporations including Google, Microsoft, Nokia, etc., study it actively as it can provide significant results in different classification and regression problems and datasets [52]. In terms of working domain, DL is considered as a subset of ML and AI, and thus DL can be seen as an AI function that mimics the human brain's processing of data. The worldwide popularity of "Deep learning" is increasing day by day, which is shown in our earlier paper [96] based on the historical data

This article is part of the topical collection "Advances in Computational Approaches for Artificial Intelligence, Image Processing, IoT and Cloud Applications" guest edited by Bhanu Prakash K. N. and M. Shivakumar.

✉ Iqbal H. Sarker
msarker@swin.edu.au

[1] Swinburne University of Technology, Melbourne, VIC 3122, Australia

[2] Chittagong University of Engineering & Technology, Chittagong 4349, Bangladesh

collected from Google trends [33]. Deep learning differs from standard machine learning in terms of efficiency as the volume of data increases, discussed briefly in Section "Why Deep Learning in Today's Research and Applications?". DL technology uses multiple layers to represent the abstractions of data to build computational models. While deep learning takes a long time to train a model due to a large number of parameters, it takes a short amount of time to run during testing as compared to other machine learning algorithms [127].

While today's Fourth Industrial Revolution (4IR or Industry 4.0) is typically focusing on technology-driven "automation, smart and intelligent systems", DL technology, which is originated from ANN, has become one of the core technologies to achieve the goal [103, 114]. A typical neural network is mainly composed of many simple, connected processing elements or processors called neurons, each of which generates a series of real-valued activations for the target outcome. Figure 1 shows a schematic representation of the mathematical model of an artificial neuron, i.e., processing element, highlighting input (X_i), weight (w), bias (b), summation function ($\sum$), activation function (f) and corresponding output signal (y). Neural network-based DL technology is now widely applied in many fields and research areas such as healthcare, sentiment analysis, natural language processing, visual recognition, business intelligence, cybersecurity, and many more that have been summarized in the latter part of this paper.

Although DL models are successfully applied in various application areas, mentioned above, building an appropriate model of deep learning is a challenging task, due to the dynamic nature and variations of real-world problems and data. Moreover, DL models are typically considered as

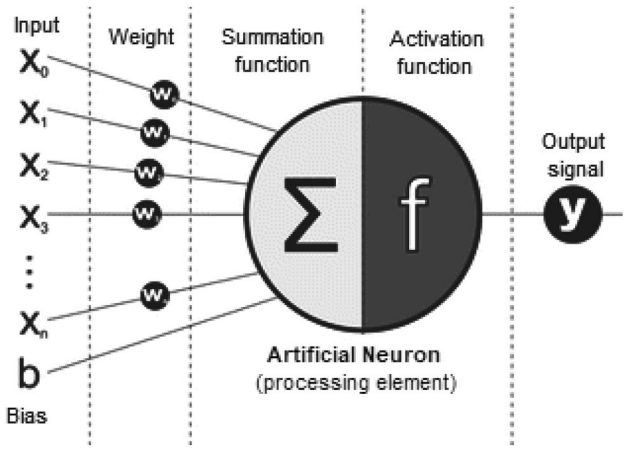

Fig. 1 Schematic representation of the mathematical model of an artificial neuron (processing element), highlighting input (X_i), weight (w), bias (b), summation function ($\sum$), activation function (f) and output signal (y)

"black-box" machines that hamper the standard development of deep learning research and applications. Thus for clear understanding, in this paper, we present a structured and comprehensive view on DL techniques considering the variations in real-world problems and tasks. To achieve our goal, we briefly discuss various DL techniques and present a *taxonomy* by taking into account three major categories: (i) deep networks for supervised or *discriminative learning* that is utilized to provide a discriminative function in supervised deep learning or classification applications; (ii) deep networks for unsupervised or *generative learning* that are used to characterize the high-order correlation properties or features for pattern analysis or synthesis, thus can be used as preprocessing for the supervised algorithm; and (ii) deep networks for *hybrid learning* that is an integration of both supervised and unsupervised model and relevant others. We take into account such categories based on the nature and learning capabilities of different DL techniques and how they are used to solve problems in real-world applications [97]. Moreover, identifying key research issues and prospects including effective data representation, new algorithm design, data-driven hyper-parameter learning, and model optimization, integrating domain knowledge, adapting resource-constrained devices, etc. is one of the key targets of this study, which can lead to "Future Generation DL-Modeling". Thus the goal of this paper is set to assist those in *academia and industry* as a reference guide, who want to *research and develop* data-driven smart and intelligent systems based on DL techniques.

The overall contribution of this paper is summarized as follows:

- This article focuses on different aspects of deep learning modeling, i.e., the learning capabilities of DL techniques in different dimensions such as supervised or unsupervised tasks, to function in an automated and intelligent manner, which can play as a core technology of today's Fourth Industrial Revolution (Industry 4.0).
- We explore a variety of prominent DL techniques and present a taxonomy by taking into account the variations in deep learning tasks and how they are used for different purposes. In our taxonomy, we divide the techniques into three major categories such as deep networks for supervised or discriminative learning, unsupervised or generative learning, as well as deep networks for hybrid learning, and relevant others.
- We have summarized several potential real-world application areas of deep learning, to assist developers as well as researchers in broadening their perspectives on DL techniques. Different categories of DL techniques highlighted in our taxonomy can be used to solve various issues accordingly.

– Finally, we point out and discuss ten potential aspects with research directions for future generation DL modeling in terms of conducting future research and system development.

This paper is organized as follows. Section "Why Deep Learning in Today's Research andApplications?" motivates why deep learning is important to build data-driven intelligent systems. In Section" Deep Learning Techniques and Applications", we present our DL taxonomy by taking into account the variations of deep learning tasks and how they are used in solving real-world issues and briefly discuss the techniques with summarizing the potential application areas. In Section "Research Directions and Future Aspects", we discuss various research issues of deep learning-based modeling and highlight the promising topics for future research within the scope of our study. Finally, Section "Concluding Remarks" concludes this paper.

Why Deep Learning in Today's Research and Applications?

The main focus of today's Fourth Industrial Revolution (Industry 4.0) is typically technology-driven automation, smart and intelligent systems, in various application areas including smart healthcare, business intelligence, smart cities, cybersecurity intelligence, and many more [95]. Deep learning approaches have grown dramatically in terms of performance in a wide range of applications considering security technologies, particularly, as an excellent solution for uncovering complex architecture in high-dimensional data. Thus, DL techniques can play a key role in building intelligent data-driven systems according to today's needs, because of their excellent learning capabilities from historical data. Consequently, DL can change the world as well as humans' everyday life through its automation power and learning from experience. DL technology is therefore relevant to artificial intelligence [103], machine learning [97] and data science with advanced analytics [95] that are well-known areas in computer science, particularly, today's intelligent computing. In the following, we first discuss regarding the position of deep learning in AI, or how DL technology is related to these areas of computing.

The Position of Deep Learning in AI

Nowadays, artificial intelligence (AI), machine learning (ML), and deep learning (DL) are three popular terms that are sometimes used interchangeably to describe systems or software that behaves intelligently. In Fig. 2, we illustrate the position of deep Learning, comparing with machine learning and artificial intelligence. According to Fig. 2, DL is a part

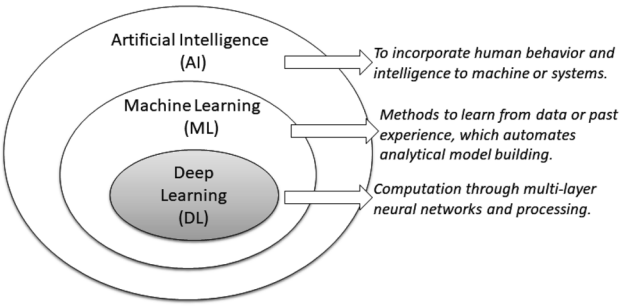

Fig. 2 An illustration of the position of deep learning (DL), comparing with machine learning (ML) and artificial intelligence (AI)

of ML as well as a part of the broad area AI. In general, AI incorporates human behavior and intelligence to machines or systems [103], while ML is the method to learn from data or experience [97], which automates analytical model building. DL also represents learning methods from data where the computation is done through multi-layer neural networks and processing. The term "Deep" in the deep learning methodology refers to the concept of multiple levels or stages through which data is processed for building a data-driven model.

Thus, DL can be considered as one of the core technology of AI, a frontier for artificial intelligence, which can be used for building intelligent systems and automation. More importantly, it pushes AI to a new level, termed "Smarter AI". As DL are capable of learning from data, there is a strong relation of deep learning with "Data Science" [95] as well. Typically, data science represents the entire process of finding meaning or insights in data in a particular problem domain, where DL methods can play a key role for advanced analytics and intelligent decision-making [104, 106]. Overall, we can conclude that DL technology is capable to change the current world, particularly, in terms of a powerful computational engine and contribute to technology-driven automation, smart and intelligent systems accordingly, and meets the goal of Industry 4.0.

Understanding Various Forms of Data

As DL models learn from data, an in-depth understanding and representation of data are important to build a data-driven intelligent system in a particular application area. In the real world, data can be in various forms, which typically can be represented as below for deep learning modeling:

– *Sequential Data* Sequential data is any kind of data where the order matters, i,e., a set of sequences. It needs to explicitly account for the sequential nature of input data while building the model. Text streams, audio frag-

ments, video clips, time-series data, are some examples of sequential data.

– *Image or 2D Data* A digital image is made up of a matrix, which is a rectangular array of numbers, symbols, or expressions arranged in rows and columns in a 2D array of numbers. Matrix, pixels, voxels, and bit depth are the four essential characteristics or fundamental parameters of a digital image.
– *Tabular Data* A tabular dataset consists primarily of rows and columns. Thus tabular datasets contain data in a columnar format as in a database table. Each column (field) must have a name and each column may only contain data of the defined type. Overall, it is a logical and systematic arrangement of data in the form of rows and columns that are based on data properties or features. Deep learning models can learn efficiently on tabular data and allow us to build data-driven intelligent systems.

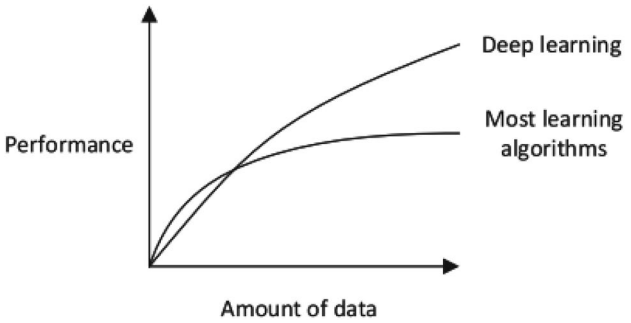

Fig. 3 An illustration of the performance comparison between deep learning (DL) and other machine learning (ML) algorithms, where DL modeling from large amounts of data can increase the performance

The above-discussed data forms are common in the real-world application areas of deep learning. Different categories of DL techniques perform differently depending on the nature and characteristics of data, discussed briefly in Section "Deep Learning Techniques and Applications" with a taxonomy presentation. However, in many real-world application areas, the standard machine learning techniques, particularly, logic-rule or tree-based techniques [93, 101] perform significantly depending on the application nature. Figure 3 also shows the performance comparison of DL and ML modeling considering the amount of data. In the following, we highlight several cases, where deep learning is useful to solve real-world problems, according to our main focus in this paper.

DL Properties and Dependencies

A DL model typically follows the same processing stages as machine learning modeling. In Fig. 4, we have shown a deep learning workflow to solve real-world problems, which consists of three processing steps, such as data understanding and preprocessing, DL model building, and training, and validation and interpretation. However, unlike the ML modeling [98, 108], feature extraction in the DL model is automated rather than manual. K-nearest neighbor, support vector machines, decision tree, random forest, naive Bayes, linear regression, association rules, k-means clustering, are some examples of machine learning techniques that are commonly used in various application areas [97]. On the other hand, the DL model includes convolution neural network, recurrent neural network, autoencoder, deep belief network, and many more, discussed briefly with their potential application areas in Section 3. In the following, we discuss the key properties and dependencies of DL techniques, that are

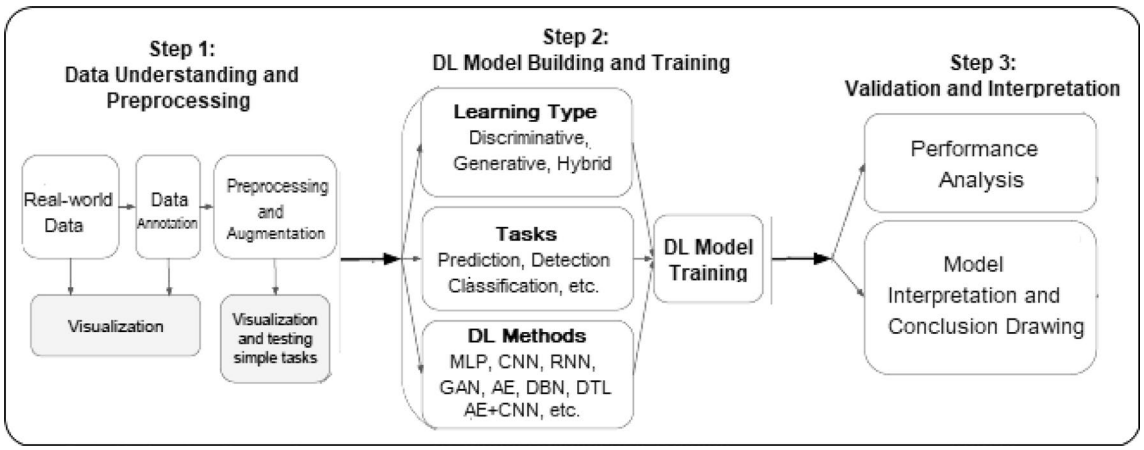

Fig. 4 A typical DL workflow to solve real-world problems, which consists of three sequential stages (i) data understanding and preprocessing (ii) DL model building and training (iii) validation and interpretation

needed to take into account before started working on DL modeling for real-world applications.

- *Data Dependencies* Deep learning is typically dependent on a large amount of data to build a data-driven model for a particular problem domain. The reason is that when the data volume is small, deep learning algorithms often perform poorly [64]. In such circumstances, however, the performance of the standard machine-learning algorithms will be improved if the specified rules are used [64, 107].
- *Hardware Dependencies* The DL algorithms require large computational operations while training a model with large datasets. As the larger the computations, the more the advantage of a GPU over a CPU, the GPU is mostly used to optimize the operations efficiently. Thus, to work properly with the deep learning training, GPU hardware is necessary. Therefore, DL relies more on high-performance machines with GPUs than standard machine learning methods [19, 127].
- *Feature Engineering Process* Feature engineering is the process of extracting features (characteristics, properties, and attributes) from raw data using domain knowledge. A fundamental distinction between DL and other machine-learning techniques is the attempt to extract high-level characteristics directly from data [22, 97]. Thus, DL decreases the time and effort required to construct a feature extractor for each problem.
- *Model Training and Execution time* In general, training a deep learning algorithm takes a long time due to a large number of parameters in the DL algorithm; thus, the model training process takes longer. For instance, the DL models can take more than one week to complete a training session, whereas training with ML algorithms takes relatively little time, only seconds to hours [107, 127]. During testing, deep learning algorithms take extremely little time to run [127], when compared to certain machine learning methods.
- *Black-box Perception and Interpretability* Interpretability is an important factor when comparing DL with ML. It's difficult to explain how a deep learning result was obtained, i.e., "black-box". On the other hand, the machine-learning algorithms, particularly, rule-based machine learning techniques [97] provide explicit logic rules (IF-THEN) for making decisions that are easily interpretable for humans. For instance, in our earlier works, we have presented several machines learning rule-based techniques [100, 102, 105], where the extracted rules are human-understandable and easier to interpret, update or delete according to the target applications.

The most significant distinction between deep learning and regular machine learning is how well it performs when data grows exponentially. An illustration of the performance comparison between DL and standard ML algorithms has been shown in Fig. 3, where DL modeling can increase the performance with the amount of data. Thus, DL modeling is extremely useful when dealing with a large amount of data because of its capacity to process vast amounts of features to build an effective data-driven model. In terms of developing and training DL models, it relies on parallelized matrix and tensor operations as well as computing gradients and optimization. Several, DL libraries and resources [30] such as PyTorch [82] (with a high-level API called Lightning) and TensorFlow [1] (which also offers Keras as a high-level API) offers these core utilities including many pre-trained models, as well as many other necessary functions for implementation and DL model building.

Deep Learning Techniques and Applications

In this section, we go through the various types of deep neural network techniques, which typically consider several layers of information-processing stages in hierarchical structures to learn. A typical deep neural network contains multiple hidden layers including input and output layers. Figure 5 shows a general structure of a deep neural network (*hidden layer* = N and $N \geq 2$) comparing with a shallow network (*hidden layer* = 1). We also present our taxonomy on DL techniques based on how they are used to solve various problems, in this section. However, before exploring the details of the DL techniques, it's useful to review various types of learning tasks such as (i) Supervised: a task-driven approach that uses labeled training data, (ii) Unsupervised: a data-driven process that analyzes unlabeled datasets, (iii) Semi-supervised: a hybridization of both the supervised and unsupervised methods, and (iv) Reinforcement: an environment driven approach, discussed briefly in our earlier paper [97]. Thus, to present our taxonomy, we divide DL techniques broadly into three major categories: (i) deep networks for supervised or discriminative learning; (ii) deep networks for unsupervised or generative learning; and (ii) deep networks for hybrid learning combing both and relevant others, as shown in Fig. 6. In the following, we briefly discuss each of these techniques that can be used to solve real-world problems in various application areas according to their learning capabilities.

Deep Networks for Supervised or Discriminative Learning

This category of DL techniques is utilized to provide a discriminative function in supervised or classification applications. Discriminative deep architectures are typically designed to give discriminative power for pattern

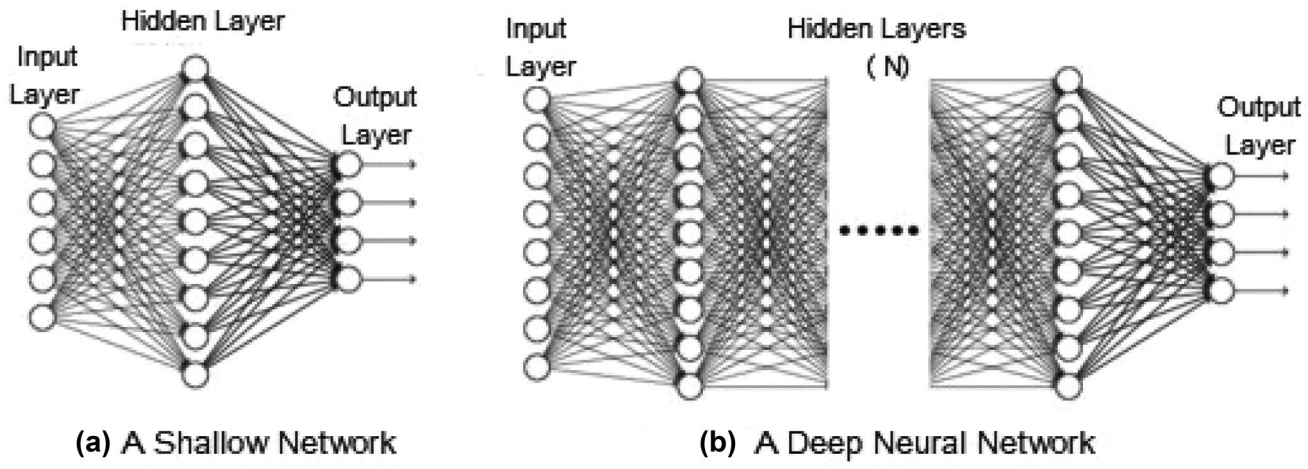

Fig. 5 A general architecture of **a** a shallow network with one hidden layer and **b** a deep neural network with multiple hidden layers

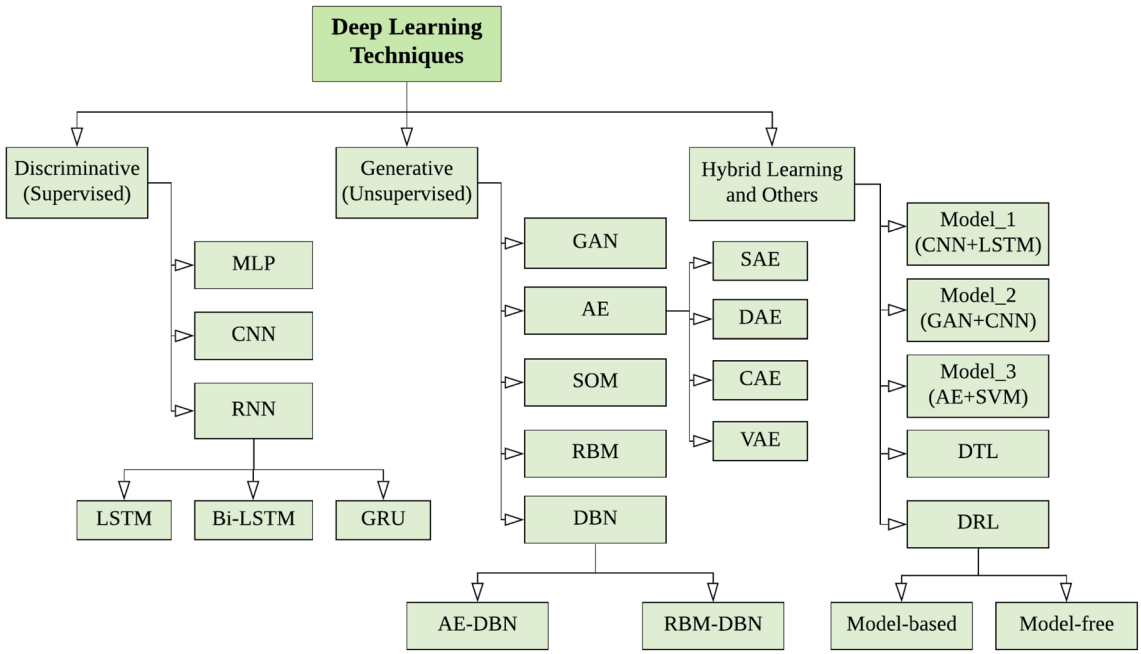

Fig. 6 A taxonomy of DL techniques, broadly divided into three major categories (i) deep networks for supervised or discriminative learning, (ii) deep networks for unsupervised or generative learning, and (ii) deep networks for hybrid learning and relevant others

classification by describing the posterior distributions of classes conditioned on visible data [21]. Discriminative architectures mainly include Multi-Layer Perceptron (MLP), Convolutional Neural Networks (CNN or ConvNet), Recurrent Neural Networks (RNN), along with their variants. In the following, we briefly discuss these techniques.

Multi-layer Perceptron (MLP)

Multi-layer Perceptron (MLP), a supervised learning approach [83], is a type of feedforward artificial neural network (ANN). It is also known as the foundation architecture of deep neural networks (DNN) or deep learning. A typical MLP is a fully connected network that consists of an input layer that receives input data, an output layer that makes a decision or prediction about the input signal, and one or more hidden layers between these two that are considered as the network's computational engine [36, 103]. The output of an MLP network is determined using a variety of activation functions, also known as transfer functions, such as ReLU (Rectified Linear Unit), Tanh, Sigmoid, and Softmax [83, 96]. To train MLP employs the most extensively

used algorithm "Backpropagation" [36], a supervised learning technique, which is also known as the most basic building block of a neural network. During the training process, various optimization approaches such as Stochastic Gradient Descent (SGD), Limited Memory BFGS (L-BFGS), and Adaptive Moment Estimation (Adam) are applied. MLP requires tuning of several hyperparameters such as the number of hidden layers, neurons, and iterations, which could make solving a complicated model computationally expensive. However, through partial fit, MLP offers the advantage of learning non-linear models in real-time or online [83].

Convolutional Neural Network (CNN or ConvNet)

The Convolutional Neural Network (CNN or ConvNet) [65] is a popular discriminative deep learning architecture that learns directly from the input without the need for human feature extraction. Figure 7 shows an example of a CNN including multiple convolutions and pooling layers. As a result, the CNN enhances the design of traditional ANN like regularized MLP networks. Each layer in CNN takes into account optimum parameters for a meaningful output as well as reduces model complexity. CNN also uses a 'dropout' [30] that can deal with the problem of over-fitting, which may occur in a traditional network.

CNNs are specifically intended to deal with a variety of 2D shapes and are thus widely employed in visual recognition, medical image analysis, image segmentation, natural language processing, and many more [65, 96]. The capability of automatically discovering essential features from the input without the need for human intervention makes it more powerful than a traditional network. Several variants of CNN are exist in the area that includes visual geometry group (VGG) [38], AlexNet [62], Xception [17], Inception [116], ResNet [39], etc. that can be used in various application domains according to their learning capabilities.

Recurrent Neural Network (RNN) and its Variants

A Recurrent Neural Network (RNN) is another popular neural network, which employs sequential or time-series data and feeds the output from the previous step as input to the current stage [27, 74]. Like feedforward and CNN, recurrent networks learn from training input, however, distinguish by their "memory", which allows them to impact current input and output through using information from previous inputs. Unlike typical DNN, which assumes that inputs and outputs are independent of one another, the output of RNN is reliant on prior elements within the sequence. However, standard recurrent networks have the issue of vanishing gradients, which makes learning long data sequences challenging. In the following, we discuss several popular variants of the recurrent network that minimizes the issues and perform well in many real-world application domains.

- *Long short-term memory (LSTM)* This is a popular form of RNN architecture that uses special units to deal with the vanishing gradient problem, which was introduced by Hochreiter et al. [42]. A memory cell in an LSTM unit can store data for long periods and the flow of information into and out of the cell is managed by three gates. For instance, the 'Forget Gate' determines what information from the previous state cell will be memorized and what information will be removed that is no longer useful, while the 'Input Gate' determines which information should enter the cell state and the 'Output Gate' determines and controls the outputs. As it solves the issues of training a recurrent network, the LSTM network is considered one of the most successful RNN.
- *Bidirectional RNN/LSTM* Bidirectional RNNs connect two hidden layers that run in opposite directions to a single output, allowing them to accept data from both the past and future. Bidirectional RNNs, unlike traditional recurrent networks, are trained to predict both positive and negative time directions at the same time. A Bidirectional LSTM, often known as a BiLSTM, is an extension of the standard LSTM that can increase model performance on sequence classification issues [113]. It is a sequence processing model comprising of two LSTMs: one takes the input forward and the other takes it backward. Bidirectional LSTM in particular is a popular choice in natural language processing tasks.
- *Gated recurrent units (GRUs)* A Gated Recurrent Unit (GRU) is another popular variant of the recurrent network that uses gating methods to control and manage information flow between cells in the neural network, introduced by Cho et al. [16]. The GRU is like an LSTM, however, has fewer parameters, as it has a reset gate and

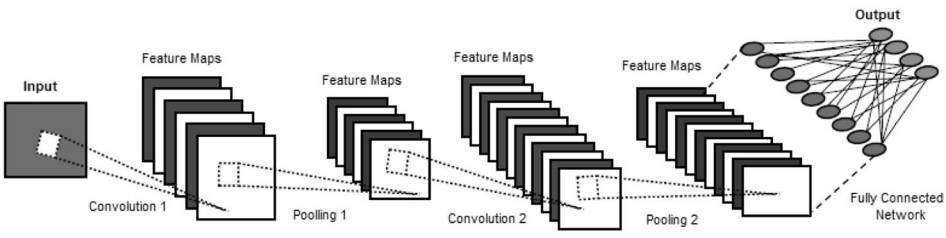

Fig. 7 An example of a convolutional neural network (CNN or ConvNet) including multiple convolution and pooling layers

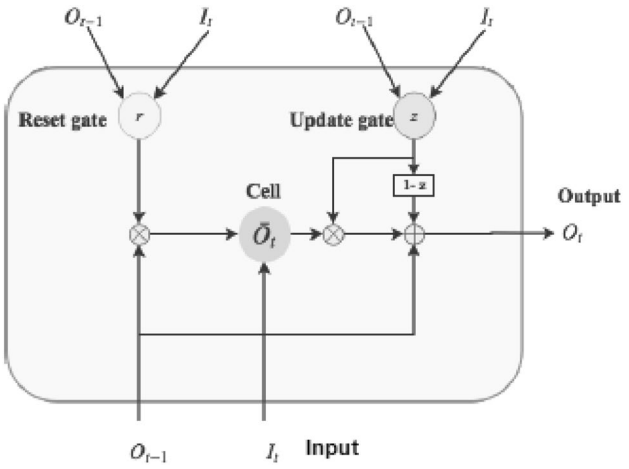

Fig. 8 Basic structure of a gated recurrent unit (GRU) cell consisting of reset and update gates

an update gate but lacks the output gate, as shown in Fig. 8. Thus, the key difference between a GRU and an LSTM is that a GRU has two gates (reset and update gates) whereas an LSTM has three gates (namely input, output and forget gates). The GRU's structure enables it to capture dependencies from large sequences of data in an adaptive manner, without discarding information from earlier parts of the sequence. Thus GRU is a slightly more streamlined variant that often offers comparable performance and is significantly faster to compute [18]. Although GRUs have been shown to exhibit better performance on certain smaller and less frequent datasets [18, 34], both variants of RNN have proven their effectiveness while producing the outcome.

Overall, the basic property of a recurrent network is that it has at least one feedback connection, which enables activations to loop. This allows the networks to do temporal processing and sequence learning, such as sequence recognition or reproduction, temporal association or prediction, etc. Following are some popular application areas of recurrent networks such as prediction problems, machine translation, natural language processing, text summarization, speech recognition, and many more.

Deep Networks for Generative or Unsupervised Learning

This category of DL techniques is typically used to characterize the high-order correlation properties or features for pattern analysis or synthesis, as well as the joint statistical distributions of the visible data and their associated classes [21]. The key idea of generative deep architectures is that during the learning process, precise supervisory information such as target class labels is not of concern. As a result, the methods under this category are essentially applied for unsupervised learning as the methods are typically used for feature learning or data generating and representation [20, 21]. Thus generative modeling can be used as preprocessing for the supervised learning tasks as well, which ensures the discriminative model accuracy. Commonly used deep neural network techniques for unsupervised or generative learning are Generative Adversarial Network (GAN), Autoencoder (AE), Restricted Boltzmann Machine (RBM), Self-Organizing Map (SOM), and Deep Belief Network (DBN) along with their variants.

Generative Adversarial Network (GAN)

A Generative Adversarial Network (GAN), designed by Ian Goodfellow [32], is a type of neural network architecture for generative modeling to create new plausible samples on demand. It involves automatically discovering and learning regularities or patterns in input data so that the model may be used to generate or output new examples from the original dataset. As shown in Fig. 9, GANs are composed of two neural networks, a generator *G* that creates new data having properties similar to the original data, and a discriminator *D* that predicts the likelihood of a subsequent sample being drawn from actual data rather than data provided by the generator. Thus in GAN modeling, both the generator and discriminator are trained to compete with each other. While the generator tries to fool and confuse the discriminator by creating more realistic data, the discriminator tries to distinguish the genuine data from the fake data generated by *G*.

Generally, GAN network deployment is designed for unsupervised learning tasks, but it has also proven to be a better solution for semi-supervised and reinforcement learning as well depending on the task [3]. GANs are also used in state-of-the-art transfer learning research to enforce the

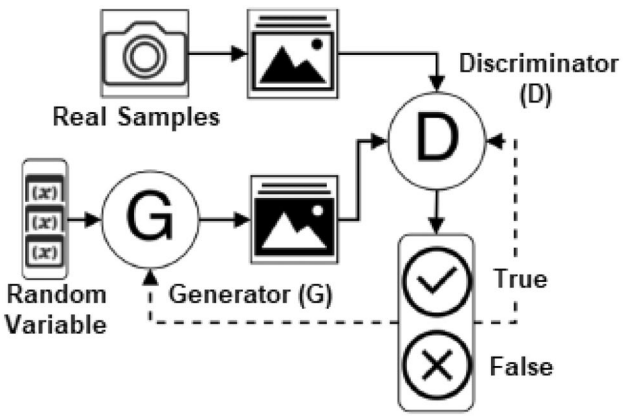

Fig. 9 Schematic structure of a standard generative adversarial network (GAN)

alignment of the latent feature space [66]. Inverse models, such as Bidirectional GAN (BiGAN) [25] can also learn a mapping from data to the latent space, similar to how the standard GAN model learns a mapping from a latent space to the data distribution. The potential application areas of GAN networks are healthcare, image analysis, data augmentation, video generation, voice generation, pandemics, traffic control, cybersecurity, and many more, which are increasing rapidly. Overall, GANs have established themselves as a comprehensive domain of independent data expansion and as a solution to problems requiring a generative solution.

Auto-Encoder (AE) and Its Variants

An auto-encoder (AE) [31] is a popular unsupervised learning technique in which neural networks are used to learn representations. Typically, auto-encoders are used to work with high-dimensional data, and dimensionality reduction explains how a set of data is represented. Encoder, code, and decoder are the three parts of an autoencoder. The encoder compresses the input and generates the code, which the decoder subsequently uses to reconstruct the input. The AEs have recently been used to learn generative data models [69]. The auto-encoder is widely used in many unsupervised learning tasks, e.g., dimensionality reduction, feature extraction, efficient coding, generative modeling, denoising, anomaly or outlier detection, etc. [31, 132]. Principal component analysis (PCA) [99], which is also used to reduce the dimensionality of huge data sets, is essentially similar to a single-layered AE with a linear activation function. Regularized autoencoders such as sparse, denoising, and contractive are useful for learning representations for later classification tasks [119], while variational autoencoders can be used as generative models [56], discussed below.

- *Sparse Autoencoder (SAE)* A sparse autoencoder [73] has a sparsity penalty on the coding layer as a part of its training requirement. SAEs may have more hidden units than inputs, but only a small number of hidden units are permitted to be active at the same time, resulting in a sparse model. Figure 10 shows a schematic structure of a sparse autoencoder with several active units in the hidden layer. This model is thus obliged to respond to the unique statistical features of the training data following its constraints.
- *Denoising Autoencoder (DAE)* A denoising autoencoder is a variant on the basic autoencoder that attempts to improve representation (to extract useful features) by altering the reconstruction criterion, and thus reduces the risk of learning the identity function [31, 119]. In other words, it receives a corrupted data point as input and is trained to recover the original undistorted input as its output through minimizing the average reconstruction error over the training data, i.e, cleaning the corrupted input, or denoising. Thus, in the context of computing, DAEs can be considered as very powerful filters that can be utilized for automatic pre-processing. A denoising autoencoder, for example, could be used to automatically pre-process an image, thereby boosting its quality for recognition accuracy.
- *Contractive Autoencoder (CAE)* The idea behind a contractive autoencoder, proposed by Rifai et al. [90], is to make the autoencoders robust of small changes in the training dataset. In its objective function, a CAE includes an explicit regularizer that forces the model to learn an encoding that is robust to small changes in input values. As a result, the learned representation's sensitivity to the training input is reduced. While DAEs encourage the robustness of reconstruction as discussed above, CAEs encourage the robustness of representation.
- *Variational Autoencoder (VAE)* A variational autoencoder [55] has a fundamentally unique property that distinguishes it from the classical autoencoder discussed above, which makes this so effective for generative modeling. VAEs, unlike the traditional autoen-

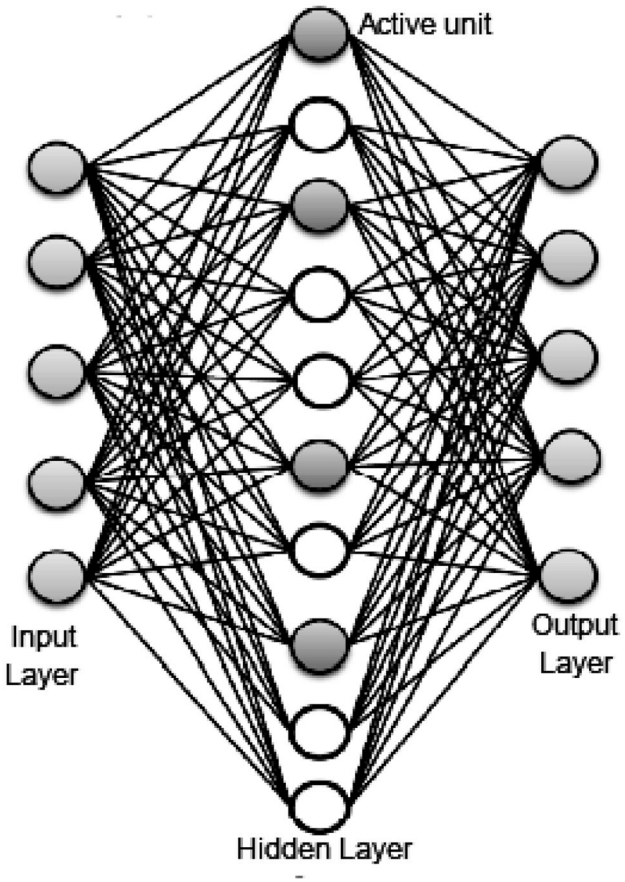

Fig. 10 Schematic structure of a sparse autoencoder (SAE) with several active units (filled circle) in the hidden layer

coders which map the input onto a latent vector, map the input data into the parameters of a probability distribution, such as the mean and variance of a Gaussian distribution. A VAE assumes that the source data has an underlying probability distribution and then tries to discover the distribution's parameters. Although this approach was initially designed for unsupervised learning, its use has been demonstrated in other domains such as semi-supervised learning [128] and supervised learning [51].

Although, the earlier concept of AE was typically for dimensionality reduction or feature learning mentioned above, recently, AEs have been brought to the forefront of generative modeling, even the generative adversarial network is one of the popular methods in the area. The AEs have been effectively employed in a variety of domains, including healthcare, computer vision, speech recognition, cybersecurity, natural language processing, and many more. Overall, we can conclude that auto-encoder and its variants can play a significant role as unsupervised feature learning with neural network architecture.

Kohonen Map or Self-Organizing Map (SOM)

A Self-Organizing Map (SOM) or Kohonen Map [59] is another form of unsupervised learning technique for creating a low-dimensional (usually two-dimensional) representation of a higher-dimensional data set while maintaining the topological structure of the data. SOM is also known as a neural network-based dimensionality reduction algorithm that is commonly used for clustering [118]. A SOM adapts to the topological form of a dataset by repeatedly moving its neurons closer to the data points, allowing us to visualize enormous datasets and find probable clusters. The first layer of a SOM is the input layer, and the second layer is the output layer or feature map. Unlike other neural networks that use error-correction learning, such as backpropagation with gradient descent [36], SOMs employ competitive learning, which uses a neighborhood function to retain the input space's topological features. SOM is widely utilized in a variety of applications, including pattern identification, health or medical diagnosis, anomaly detection, and virus or worm attack detection [60, 87]. The primary benefit of employing a SOM is that this can make high-dimensional data easier to visualize and analyze to understand the patterns. The reduction of dimensionality and grid clustering makes it easy to observe similarities in the data. As a result, SOMs can play a vital role in developing a data-driven effective model for a particular problem domain, depending on the data characteristics.

Restricted Boltzmann Machine (RBM)

A Restricted Boltzmann Machine (RBM) [75] is also a generative stochastic neural network capable of learning a probability distribution across its inputs. Boltzmann machines typically consist of visible and hidden nodes and each node is connected to every other node, which helps us understand irregularities by learning how the system works in normal circumstances. RBMs are a subset of Boltzmann machines that have a limit on the number of connections between the visible and hidden layers [77]. This restriction permits training algorithms like the gradient-based contrastive divergence algorithm to be more efficient than those for Boltzmann machines in general [41]. RBMs have found applications in dimensionality reduction, classification, regression, collaborative filtering, feature learning, topic modeling, and many others. In the area of deep learning modeling, they can be trained either supervised or unsupervised, depending on the task. Overall, the RBMs can recognize patterns in data automatically and develop probabilistic or stochastic models, which are utilized for feature selection or extraction, as well as forming a deep belief network.

Deep Belief Network (DBN)

A Deep Belief Network (DBN) [40] is a multi-layer generative graphical model of stacking several individual unsupervised networks such as AEs or RBMs, that use each network's hidden layer as the input for the next layer, i.e, connected sequentially. Thus, we can divide a DBN into (i) AE-DBN which is known as stacked AE, and (ii) RBM-DBN that is known as stacked RBM, where AE-DBN is composed of autoencoders and RBM-DBN is composed of restricted Boltzmann machines, discussed earlier. The ultimate goal is to develop a faster-unsupervised training technique for each sub-network that depends on contrastive divergence [41]. DBN can capture a hierarchical representation of input data based on its deep structure. The primary idea behind DBN is to train unsupervised feed-forward neural networks with unlabeled data before fine-tuning the network with labeled input. One of the most important advantages of DBN, as opposed to typical shallow learning networks, is that it permits the detection of deep patterns, which allows for reasoning abilities and the capture of the deep difference between normal and erroneous data [89]. A continuous DBN is simply an extension of a standard DBN that allows a continuous range of decimals instead of binary data. Overall, the DBN model can play a key role in a wide range of high-dimensional data applications due to its strong feature extraction and classification capabilities and become one of the significant topics in the field of neural networks.

In summary, the generative learning techniques discussed above typically allow us to generate a new representation

of data through exploratory analysis. As a result, these deep generative networks can be utilized as preprocessing for supervised or discriminative learning tasks, as well as ensuring model accuracy, where unsupervised representation learning can allow for improved classifier generalization.

Deep Networks for Hybrid Learning and Other Approaches

In addition to the above-discussed deep learning categories, hybrid deep networks and several other approaches such as deep transfer learning (DTL) and deep reinforcement learning (DRL) are popular, which are discussed in the following.

Hybrid Deep Neural Networks

Generative models are adaptable, with the capacity to learn from both labeled and unlabeled data. Discriminative models, on the other hand, are unable to learn from unlabeled data yet outperform their generative counterparts in supervised tasks. A framework for training both deep generative and discriminative models simultaneously can enjoy the benefits of both models, which motivates hybrid networks.

Hybrid deep learning models are typically composed of multiple (two or more) deep basic learning models, where the basic model is a discriminative or generative deep learning model discussed earlier. Based on the integration of different basic generative or discriminative models, the below three categories of hybrid deep learning models might be useful for solving real-world problems. These are as follows:

- *Hybrid Model*_1: An integration of different generative or discriminative models to extract more meaningful and robust features. Examples could be CNN+LSTM, AE+GAN, and so on.
- *Hybrid Model*_2: An integration of generative model followed by a discriminative model. Examples could be DBN+MLP, GAN+CNN, AE+CNN, and so on.
- *Hybrid Model*_3: An integration of generative or discriminative model followed by a non-deep learning classifier. Examples could be AE+SVM, CNN+SVM, and so on.

Thus, in a broad sense, we can conclude that hybrid models can be either classification-focused or non-classification depending on the target use. However, most of the hybrid learning-related studies in the area of deep learning are classification-focused or supervised learning tasks, summarized in Table 1. The unsupervised generative models with meaningful representations are employed to enhance the discriminative models. The generative models with useful representation can provide more informative and low-dimensional features for discrimination, and they can also enable to enhance the training data quality and quantity, providing additional information for classification.

Deep Transfer Learning (DTL)

Transfer Learning is a technique for effectively using previously learned model knowledge to solve a new task with minimum training or fine-tuning. In comparison to typical machine learning techniques [97], DL takes a large amount of training data. As a result, the need for a substantial volume of labeled data is a significant barrier to address some essential domain-specific tasks, particularly, in the medical sector, where creating large-scale, high-quality annotated medical or health datasets is both difficult and costly. Furthermore, the standard DL model demands a lot of computational resources, such as a GPU-enabled server, even though researchers are working hard to improve it. As a result, Deep Transfer Learning (DTL), a DL-based transfer learning method, might be helpful to address this issue. Figure 11 shows a general structure of the transfer learning process, where knowledge from the pre-trained model is transferred into a new DL model. It's especially popular in deep learning right now since it allows to train deep neural networks with very little data [126].

Transfer learning is a two-stage approach for training a DL model that consists of a pre-training step and a fine-tuning step in which the model is trained on the target task. Since deep neural networks have gained popularity in a variety of fields, a large number of DTL methods have been presented, making it crucial to categorize and summarize them. Based on the techniques used in the literature, DTL can be classified into four categories [117]. These are (i) instances-based deep transfer learning that utilizes instances in source domain by appropriate weight, (ii) mapping-based deep transfer learning that maps instances from two domains into a new data space with better similarity, (iii) network-based deep transfer learning that reuses the partial of network pre-trained in the source domain, and (iv) adversarial based deep transfer learning that uses adversarial technology to find transferable features that both suitable for two domains. Due to its high effectiveness and practicality, adversarial-based deep transfer learning has exploded in popularity in recent years. Transfer learning can also be classified into inductive, transductive, and unsupervised transfer learning depending on the circumstances between the source and target domains and activities [81]. While most current research focuses on supervised learning, how deep neural networks can transfer knowledge in unsupervised or semi-supervised learning may gain further interest in the future. DTL techniques are useful in a variety of fields including natural language processing, sentiment classification, visual recognition, speech recognition, spam filtering, and relevant others.

Table 1 A summary of deep learning tasks and methods in several popular real-world applications areas

Application areas	Tasks	Methods	References
Healthcare and Medical applications	Regular health factors analysis	CNN-based	Ismail et al. [48]
	Identifying malicious behaviors	RNN-based	Xue et al. [129]
	Coronary heart disease risk prediction	Autoencoder based	Amarbayasgalan et al. [6]
	Cancer classification	Transfer learning based	Sevakula et al. [110]
	Diagnosis of COVID-19	CNN and BiLSTM based	Aslan et al. [10]
	Detection of COVID-19	CNN-LSTM based	Islam et al. [47]
Natural Language Processing	Text summarization	Auto-encoder based	Yousefi et al. [130]
	Sentiment analysis	CNN-LSTM based	Wang et al. [120]
	Sentiment analysis	CNN and Bi-LSTM based	Minaee et al. [78]
	Aspect-level sentiment classification	Attention-based LSTM	Wang et al. [124]
Speech recognition	Distant speech recognition	Attention-based LSTM	Zhang et al. [135]
	Speech emotion classification	Transfer learning based	Latif et al. [63]
	Emotion recognition from speech	CNN and LSTM based	Satt et al. [109]
Cybersecurity	Zero-day malware detection	Autoencoders and GAN based	Kim et al. [54]
	Security incidents and fraud analysis	SOM-based	Lopez et al. [70]
	Android malware detection	Autoencoder and CNN based	Wang et al. [122]
	intrusion detection classification	DBN-based	Wei et al. [125]
	DoS attack detection	RBM-based	Imamverdiyev et al. [46]
	Suspicious flow detection	Hybrid deep-learning-based	Garg et al. [29]
	Network intrusion detection	AE and SVM based	Al et al. [4]
IoT and Smart cities	Smart energy management	CNN and Attention mechanism	Abdel et al. [2]
	Particulate matter forecasting	CNN-LSTM based	Huang et al. [43]
	Smart parking system	CNN-LSTM based	Piccialli et al. [85]
	Disaster management	DNN-based	Aqib et al. [8]
	Air quality prediction	LSTM-RNN based	Kok et al. [61]
	Cybersecurity in smart cities	RBM, DBN, RNN, CNN, GAN	Chen et al. [15]
Smart Agriculture	A smart agriculture IoT system	RL-based	Bu et al. [11]
	Plant disease detection	CNN-based	Ale et al. [5]
	Automated soil quality evaluation	DNN-based	Sumathi et al. [115]
Business and Financial Services	Predicting customers' purchase behavior	DNN based	Chaudhuri [14]
	Stock trend prediction	CNN and LSTM based	anuradha et al. [7]
	Financial loan default prediction	CNN-based	Deng et al. [23]
	Power consumption forecasting	LSTM-based	Shao et al. [112]
Virtual Assistant and Chatbot Services	An intelligent chatbot	Bi-RNN and Attention model	Dhyani et al. [24]
	Virtual listener agent	GRU and LSTM based	Huang et al. [44]
	Smart blind assistant	CNN-based	Rahman et al. [88]
Object Detection and Recognition	Object detection in X-ray images	CNN-based	Gu et al. [35]
	Object detection for disaster response	CNN-based	Pi et al. [84]
	Medicine recognition system	CNN-based	Chang et al. [12]
	Face recognition in IoT-cloud environment	CNN-based	Masud et al. [76]
	Food recognition system	CNN-based	Liu et al. [68]
	Affect recognition system	DBN-based	Kawde et al. [53]
	Facial expression analysis	CNN and LSTM based	Li et al. [67]
Recommendation and Intelligent system	Hybrid recommender system	DNN-based	Kiran et al. [57]
	Visual recommendation and search	CNN-based	Shankar et al. [111]
	Recommendation system	CNN and Bi-LSTM based	Rosa et al. [91]
	Intelligent system for impaired patients	RL-based	Naeem et al. [79]
	Intelligent transportation system	CNN-based	Wang et al. [123]

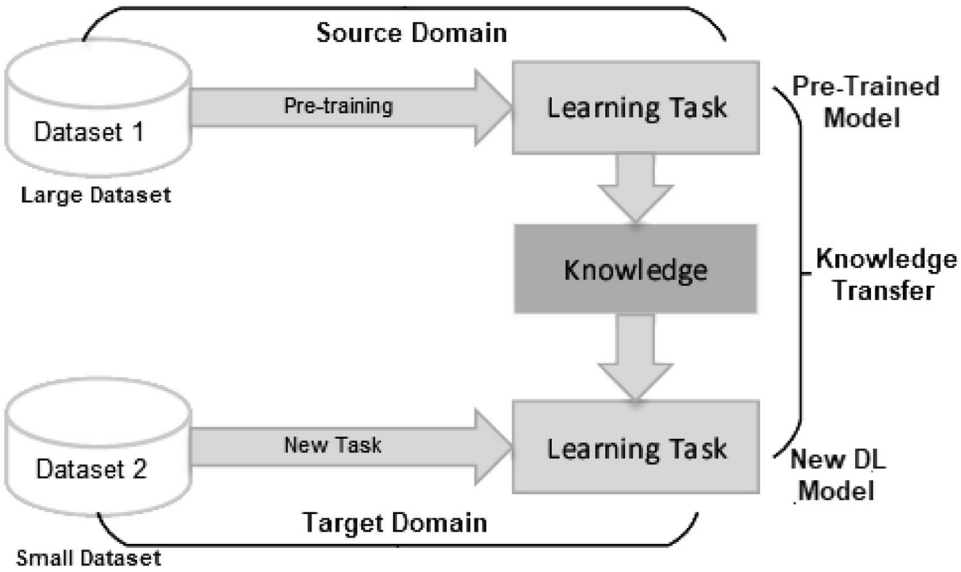

Fig. 11 A general structure of transfer learning process, where knowledge from pre-trained model is transferred into new DL model

Deep Reinforcement Learning (DRL)

Reinforcement learning takes a different approach to solving the sequential decision-making problem than other approaches we have discussed so far. The concepts of an environment and an agent are often introduced first in reinforcement learning. The agent can perform a series of actions in the environment, each of which has an impact on the environment's state and can result in possible rewards (feedback) - "positive" for good sequences of actions that result in a "good" state, and "negative" for bad sequences of actions that result in a "bad" state. The purpose of reinforcement learning is to learn good action sequences through interaction with the environment, typically referred to as a policy.

Deep reinforcement learning (DRL or deep RL) [9] integrates neural networks with a reinforcement learning architecture to allow the agents to learn the appropriate actions in a virtual environment, as shown in Fig. 12. In the area of reinforcement learning, model-based RL is based on learning a transition model that enables for modeling of the environment without interacting with it directly, whereas model-free RL methods learn directly from interactions with the environment. Q-learning is a popular model-free RL technique for determining the best action-selection policy for any (finite) Markov Decision Process (MDP) [86, 97]. MDP is a mathematical framework for modeling decisions based on state, action, and rewards [86]. In addition, Deep Q-Networks, Double DQN, Bi-directional Learning, Monte Carlo Control, etc. are used in the area [50, 97]. In DRL methods it incorporates DL models, e.g. Deep Neural Networks (DNN), based on MDP principle [71], as policy and/or value function approximators. CNN for example can be used as a component of RL agents to learn directly from raw, high-dimensional visual inputs. In the real world, DRL-based solutions can be used in several application areas including robotics, video games, natural language processing, computer vision, and relevant others.

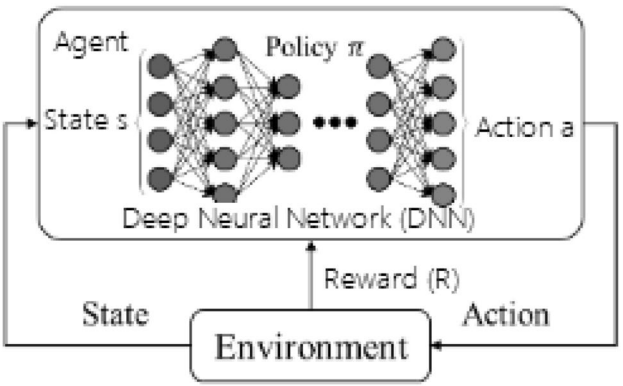

Fig. 12 Schematic structure of deep reinforcement learning (DRL) highlighting a deep neural network

Deep Learning Application Summary

During the past few years, deep learning has been successfully applied to numerous problems in many application areas. These include natural language processing, sentiment analysis, cybersecurity, business, virtual assistants, visual recognition, healthcare, robotics, and many more. In Fig. 13, we have summarized several potential real-world application areas of deep learning. Various deep learning techniques according to our presented taxonomy in Fig. 6 that includes discriminative learning, generative learning, as well as hybrid models, discussed earlier, are employed in these application areas. In Table 1, we have also summarized

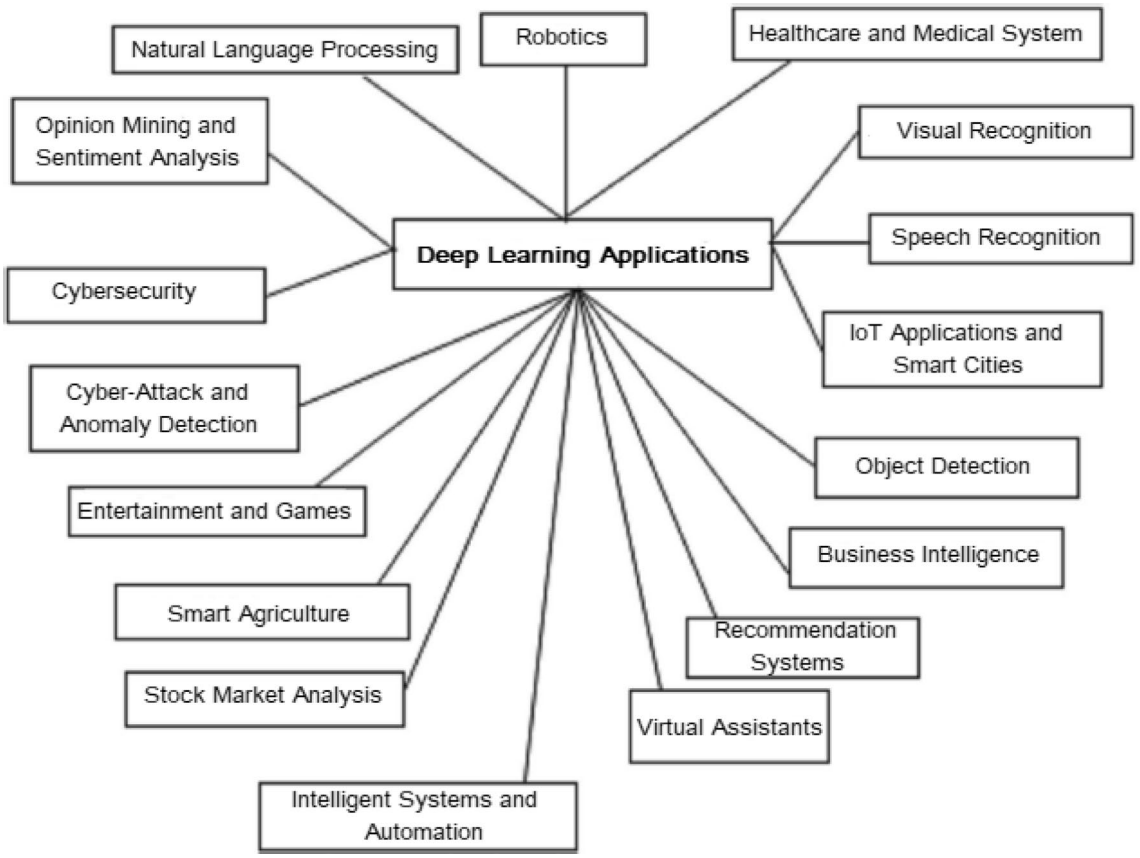

Fig. 13 Several potential real-world application areas of deep learning

various deep learning tasks and techniques that are used to solve the relevant tasks in several real-world applications areas. Overall, from Fig. 13 and Table 1, we can conclude that the future prospects of deep learning modeling in real-world application areas are huge and there are lots of scopes to work. In the next section, we also summarize the research issues in deep learning modeling and point out the potential aspects for future generation DL modeling.

Research Directions and Future Aspects

While existing methods have established a solid foundation for deep learning systems and research, this section outlines the below ten potential future research directions based on our study.

– *Automation in Data Annotation* According to the existing literature, discussed in Section 3, most of the deep learning models are trained through publicly available datasets that are annotated. However, to build a system for a new problem domain or recent data-driven system, raw data from relevant sources are needed to collect. Thus, data annotation, e.g., categorization, tagging, or labeling of a large amount of raw data, is important for building discriminative deep learning models or supervised tasks, which is challenging. A technique with the capability of automatic and dynamic data annotation, rather than manual annotation or hiring annotators, particularly, for large datasets, could be more effective for supervised learning as well as minimizing human effort. Therefore, a more in-depth investigation of data collection and annotation methods, or designing an unsupervised learning-based solution could be one of the primary research directions in the area of deep learning modeling.

– *Data Preparation for Ensuring Data Quality* As discussed earlier throughout the paper, the deep learning algorithms highly impact data quality, and availability for training, and consequently on the resultant model for a particular problem domain. Thus, deep learning models may become worthless or yield decreased accuracy if the data is bad, such as data sparsity, non-representative, poor-quality, ambiguous values, noise, data imbalance, irrelevant features, data inconsistency, insufficient quantity, and so on for training. Consequently, such issues in data can lead to poor processing and inaccurate find-

ings, which is a major problem while discovering insights from data. Thus deep learning models also need to adapt to such rising issues in data, to capture approximated information from observations. Therefore, effective data pre-processing techniques are needed to design according to the nature of the data problem and characteristics, to handling such emerging challenges, which could be another research direction in the area.

– *Black-box Perception and Proper DL/ML Algorithm Selection* In general, it's difficult to explain how a deep learning result is obtained or how they get the ultimate decisions for a particular model. Although DL models achieve significant performance while learning from large datasets, as discussed in Section 2, this "black-box" perception of DL modeling typically represents weak statistical interpretability that could be a major issue in the area. On the other hand, ML algorithms, particularly, rule-based machine learning techniques provide explicit logic rules (IF-THEN) for making decisions that are easier to interpret, update or delete according to the target applications [97, 100, 105]. If the wrong learning algorithm is chosen, unanticipated results may occur, resulting in a loss of effort as well as the model's efficacy and accuracy. Thus by taking into account the performance, complexity, model accuracy, and applicability, selecting an appropriate model for the target application is challenging, and in-depth analysis is needed for better understanding and decision making.
– *Deep Networks for Supervised or Discriminative Learning:* According to our designed taxonomy of deep learning techniques, as shown in Fig. 6, discriminative architectures mainly include MLP, CNN, and RNN, along with their variants that are applied widely in various application domains. However, designing new techniques or their variants of such discriminative techniques by taking into account model optimization, accuracy, and applicability, according to the target real-world application and the nature of the data, could be a novel contribution, which can also be considered as a major future aspect in the area of supervised or discriminative learning.
– *Deep Networks for Unsupervised or Generative Learning* As discussed in Section 3, unsupervised learning or generative deep learning modeling is one of the major tasks in the area, as it allows us to characterize the high-order correlation properties or features in data, or generating a new representation of data through exploratory analysis. Moreover, unlike supervised learning [97], it does not require labeled data due to its capability to derive insights directly from the data as well as data-driven decision making. Consequently, it thus can be used as preprocessing for supervised learning or discriminative modeling as well as semi-supervised learning tasks, which ensure learning accuracy and model efficiency. According to our designed taxonomy of deep learning techniques, as shown in Fig. 6, generative techniques mainly include GAN, AE, SOM, RBM, DBN, and their variants. Thus, designing new techniques or their variants for an effective data modeling or representation according to the target real-world application could be a novel contribution, which can also be considered as a major future aspect in the area of unsupervised or generative learning.
– *Hybrid/Ensemble Modeling and Uncertainty Handling* According to our designed taxonomy of DL techniques, as shown in Fig 6, this is considered as another major category in deep learning tasks. As hybrid modeling enjoys the benefits of both generative and discriminative learning, an effective hybridization can outperform others in terms of performance as well as uncertainty handling in high-risk applications. In Section 3, we have summarized various types of hybridization, e.g., AE+CNN/SVM. Since a group of neural networks is trained with distinct parameters or with separate subsampling training datasets, hybridization or ensembles of such techniques, i.e., DL with DL/ML, can play a key role in the area. Thus designing effective blended discriminative and generative models accordingly rather than naive method, could be an important research opportunity to solve various real-world issues including semi-supervised learning tasks and model uncertainty.
– *Dynamism in Selecting Threshold/ Hyper-parameters Values, and Network Structures with Computational Efficiency* In general, the relationship among performance, model complexity, and computational requirements is a key issue in deep learning modeling and applications. A combination of algorithmic advancements with improved accuracy as well as maintaining computational efficiency, i.e., achieving the maximum throughput while consuming the least amount of resources, without significant information loss, can lead to a breakthrough in the effectiveness of deep learning modeling in future real-world applications. The concept of incremental approaches or recency-based learning [100] might be effective in several cases depending on the nature of target applications. Moreover, assuming the network structures with a static number of nodes and layers, hyper-parameters values or threshold settings, or selecting them by the trial-and-error process may not be effective in many cases, as it can be changed due to the changes in data. Thus, a data-driven approach to select them dynamically could be more effective while building a deep learning model in terms of both performance and real-world applicability. Such type of data-driven automation can lead to future generation deep learning modeling with additional intelligence, which could be a significant future aspect in the

area as well as an important research direction to contribute.

– *Lightweight Deep Learning Modeling for Next-Generation Smart Devices and Applications:* In recent years, the Internet of Things (IoT) consisting of billions of intelligent and communicating things and mobile communications technologies have become popular to detect and gather human and environmental information (e.g. geo-information, weather data, bio-data, human behaviors, and so on) for a variety of intelligent services and applications. Every day, these ubiquitous smart things or devices generate large amounts of data, requiring rapid data processing on a variety of smart mobile devices [72]. Deep learning technologies can be incorporate to discover underlying properties and to effectively handle such large amounts of sensor data for a variety of IoT applications including health monitoring and disease analysis, smart cities, traffic flow prediction, and monitoring, smart transportation, manufacture inspection, fault assessment, smart industry or Industry 4.0, and many more. Although deep learning techniques discussed in Section 3 are considered as powerful tools for processing big data, lightweight modeling is important for resource-constrained devices, due to their high computational cost and considerable memory overhead. Thus several techniques such as optimization, simplification, compression, pruning, generalization, important feature extraction, etc. might be helpful in several cases. Therefore, constructing the lightweight deep learning techniques based on a baseline network architecture to adapt the DL model for next-generation mobile, IoT, or resource-constrained devices and applications, could be considered as a significant future aspect in the area.
– *Incorporating Domain Knowledge into Deep Learning Modeling* Domain knowledge, as opposed to general knowledge or domain-independent knowledge, is knowledge of a specific, specialized topic or field. For instance, in terms of natural language processing, the properties of the English language typically differ from other languages like Bengali, Arabic, French, etc. Thus integrating domain-based constraints into the deep learning model could produce better results for such particular purpose. For instance, a task-specific feature extractor considering domain knowledge in smart manufacturing for fault diagnosis can resolve the issues in traditional deep-learning-based methods [28]. Similarly, domain knowledge in medical image analysis [58], financial sentiment analysis [49], cybersecurity analytics [94, 103] as well as conceptual data model in which semantic information, (i.e., meaningful for a system, rather than merely correlational) [45, 121, 131] is included, can play a vital role in the area. Transfer learning could be an effective way to get started on a new challenge with domain knowledge. Moreover, contextual information such as spatial, temporal, social, environmental contexts [92, 104, 108] can also play an important role to incorporate context-aware computing with domain knowledge for smart decision making as well as building adaptive and intelligent context-aware systems. Therefore understanding domain knowledge and effectively incorporating them into the deep learning model could be another research direction.
– *Designing General Deep Learning Framework for Target Application Domains* One promising research direction for deep learning-based solutions is to develop a general framework that can handle data diversity, dimensions, stimulation types, etc. The general framework would require two key capabilities: the attention mechanism that focuses on the most valuable parts of input signals, and the ability to capture latent feature that enables the framework to capture the distinctive and informative features. Attention models have been a popular research topic because of their intuition, versatility, and interpretability, and employed in various application areas like computer vision, natural language processing, text or image classification, sentiment analysis, recommender systems, user profiling, etc [13, 80]. Attention mechanism can be implemented based on learning algorithms such as reinforcement learning that is capable of finding the most useful part through a policy search [133, 134]. Similarly, CNN can be integrated with suitable attention mechanisms to form a general classification framework, where CNN can be used as a feature learning tool for capturing features in various levels and ranges. Thus, designing a general deep learning framework considering attention as well as a latent feature for target application domains could be another area to contribute.

To summarize, deep learning is a fairly open topic to which academics can contribute by developing new methods or improving existing methods to handle the above-mentioned concerns and tackle real-world problems in a variety of application areas. This can also help the researchers conduct a thorough analysis of the application's hidden and unexpected challenges to produce more reliable and realistic outcomes. Overall, we can conclude that addressing the above-mentioned issues and contributing to proposing effective and efficient techniques could lead to "Future Generation DL" modeling as well as more intelligent and automated applications.

Concluding Remarks

In this article, we have presented a structured and comprehensive view of deep learning technology, which is considered a core part of artificial intelligence as well as data science. It starts with a history of artificial neural networks and

moves to recent deep learning techniques and breakthroughs in different applications. Then, the key algorithms in this area, as well as deep neural network modeling in various dimensions are explored. For this, we have also presented a taxonomy considering the variations of deep learning tasks and how they are used for different purposes. In our comprehensive study, we have taken into account not only the deep networks for supervised or discriminative learning but also the deep networks for unsupervised or generative learning, and hybrid learning that can be used to solve a variety of real-world issues according to the nature of problems.

Deep learning, unlike traditional machine learning and data mining algorithms, can produce extremely high-level data representations from enormous amounts of raw data. As a result, it has provided an excellent solution to a variety of real-world problems. A successful deep learning technique must possess the relevant data-driven modeling depending on the characteristics of raw data. The sophisticated learning algorithms then need to be trained through the collected data and knowledge related to the target application before the system can assist with intelligent decision-making. Deep learning has shown to be useful in a wide range of applications and research areas such as healthcare, sentiment analysis, visual recognition, business intelligence, cybersecurity, and many more that are summarized in the paper.

Finally, we have summarized and discussed the challenges faced and the potential research directions, and future aspects in the area. Although deep learning is considered a black-box solution for many applications due to its poor reasoning and interpretability, addressing the challenges or future aspects that are identified could lead to future generation deep learning modeling and smarter systems. This can also help the researchers for in-depth analysis to produce more reliable and realistic outcomes. Overall, we believe that our study on neural networks and deep learning-based advanced analytics points in a promising path and can be utilized as a reference guide for future research and implementations in relevant application domains by both academic and industry professionals.

Declarations

Conflict of interest The author declares no conflict of interest.

References

1. Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin Ma, Ghemawat S, Irving G, Isard M, et al. Tensorflow: a system for large-scale machine learning. In: 12th {*USENIX*} Symposium on operating systems design and implementation ({*OSDI*} 16), 2016; p. 265–283.
2. Abdel-Basset M, Hawash H, Chakrabortty RK, Ryan M. Energy-net: a deep learning approach for smart energy management in iot-based smart cities. IEEE Internet of Things J. 2021.
3. Aggarwal A, Mittal M, Battineni G. Generative adversarial network: an overview of theory and applications. Int J Inf Manag Data Insights. 2021; p. 100004.
4. Al-Qatf M, Lasheng Y, Al-Habib M, Al-Sabahi K. Deep learning approach combining sparse autoencoder with svm for network intrusion detection. IEEE Access. 2018;6:52843–56.
5. Ale L, Sheta A, Li L, Wang Y, Zhang N. Deep learning based plant disease detection for smart agriculture. In: 2019 IEEE Globecom Workshops (GC Wkshps), 2019; p. 1–6. IEEE.
6. Amarbayasgalan T, Lee JY, Kim KR, Ryu KH. Deep autoencoder based neural networks for coronary heart disease risk prediction. In: Heterogeneous data management, polystores, and analytics for healthcare. Springer; 2019. p. 237–48.
7. Anuradha J, et al. Big data based stock trend prediction using deep cnn with reinforcement-lstm model. Int J Syst Assur Eng Manag. 2021; p. 1–11.
8. Aqib M, Mehmood R, Albeshri A, Alzahrani A. Disaster management in smart cities by forecasting traffic plan using deep learning and gpus. In: International Conference on smart cities, infrastructure, technologies and applications. Springer; 2017. p. 139–54.
9. Arulkumaran K, Deisenroth MP, Brundage M, Bharath AA. Deep reinforcement learning: a brief survey. IEEE Signal Process Mag. 2017;34(6):26–38.
10. Aslan MF, Unlersen MF, Sabanci K, Durdu A. Cnn-based transfer learning-bilstm network: a novel approach for covid-19 infection detection. Appl Soft Comput. 2021;98:106912.
11. Bu F, Wang X. A smart agriculture iot system based on deep reinforcement learning. Futur Gener Comput Syst. 2019;99:500–7.
12. Chang W-J, Chen L-B, Hsu C-H, Lin C-P, Yang T-C. A deep learning-based intelligent medicine recognition system for chronic patients. IEEE Access. 2019;7:44441–58.
13. Chaudhari S, Mithal V, Polatkan Gu, Ramanath R. An attentive survey of attention models. arXiv preprint arXiv:1904.02874, 2019.
14. Chaudhuri N, Gupta G, Vamsi V, Bose I. On the platform but will they buy? predicting customers' purchase behavior using deep learning. Decis Support Syst. 2021; p. 113622.
15. Chen D, Wawrzynski P, Lv Z. Cyber security in smart cities: a review of deep learning-based applications and case studies. Sustain Cities Soc. 2020; p. 102655.
16. Cho K, Van MB, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y. Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078, 2014.
17. Chollet F. Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE Conference on computer vision and pattern recognition, 2017; p. 1251–258.
18. Chung J, Gulcehre C, Cho KH, Bengio Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555, 2014.
19. Coelho IM, Coelho VN, da Eduardo J, Luz S, Ochi LS, Guimarães FG, Rios E. A gpu deep learning metaheuristic based model for time series forecasting. Appl Energy. 2017;201:412–8.
20. Da'u A, Salim N. Recommendation system based on deep learning methods: a systematic review and new directions. Artif Intel Rev. 2020;53(4):2709–48.
21. Deng L. A tutorial survey of architectures, algorithms, and applications for deep learning. APSIPA Trans Signal Inf Process. 2014; p. 3.
22. Deng L, Dong Yu. Deep learning: methods and applications. Found Trends Signal Process. 2014;7(3–4):197–387.

23. Deng S, Li R, Jin Y, He H. Cnn-based feature cross and classifier for loan default prediction. In: 2020 International Conference on image, video processing and artificial intelligence, volume 11584, page 115841K. International Society for Optics and Photonics, 2020.
24. Dhyani M, Kumar R. An intelligent chatbot using deep learning with bidirectional rnn and attention model. Mater Today Proc. 2021;34:817–24.
25. Donahue J, Krähenbühl P, Darrell T. Adversarial feature learning. arXiv preprint arXiv:1605.09782, 2016.
26. Du K-L, Swamy MNS. Neural networks and statistical learning. Berlin: Springer Science & Business Media; 2013.
27. Dupond S. A thorough review on the current advance of neural network structures. Annu Rev Control. 2019;14:200–30.
28. Feng J, Yao Y, Lu S, Liu Y. Domain knowledge-based deep-broad learning framework for fault diagnosis. IEEE Trans Ind Electron. 2020;68(4):3454–64.
29. Garg S, Kaur K, Kumar N, Rodrigues JJPC. Hybrid deep-learning-based anomaly detection scheme for suspicious flow detection in sdn: a social multimedia perspective. IEEE Trans Multimed. 2019;21(3):566–78.
30. Géron A. Hands-on machine learning with Scikit-Learn, Keras. In: and TensorFlow: concepts, tools, and techniques to build intelligent systems. O'Reilly Media; 2019.
31. Goodfellow I, Bengio Y, Courville A, Bengio Y. Deep learning, vol. 1. Cambridge: MIT Press; 2016.
32. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y. Generative adversarial nets. In: Advances in neural information processing systems. 2014; p. 2672–680.
33. Google trends. 2021. https://trends.google.com/trends/.
34. Gruber N, Jockisch A. Are gru cells more specific and lstm cells more sensitive in motive classification of text? Front Artif Intell. 2020;3:40.
35. Gu B, Ge R, Chen Y, Luo L, Coatrieux G. Automatic and robust object detection in x-ray baggage inspection using deep convolutional neural networks. IEEE Trans Ind Electron. 2020.
36. Han J, Pei J, Kamber M. Data mining: concepts and techniques. Amsterdam: Elsevier; 2011.
37. Haykin S. Neural networks and learning machines, 3/E. London: Pearson Education; 2010.
38. He K, Zhang X, Ren S, Sun J. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans Pattern Anal Mach Intell. 2015;37(9):1904–16.
39. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on computer vision and pattern recognition, 2016; p. 770–78.
40. Hinton GE. Deep belief networks. Scholarpedia. 2009;4(5):5947.
41. Hinton GE, Osindero S, Teh Y-W. A fast learning algorithm for deep belief nets. Neural Comput. 2006;18(7):1527–54.
42. Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput. 1997;9(8):1735–80.
43. Huang C-J, Kuo P-H. A deep cnn-lstm model for particulate matter (pm2. 5) forecasting in smart cities. Sensors. 2018;18(7):2220.
44. Huang H-H, Fukuda M, Nishida T. Toward rnn based micro non-verbal behavior generation for virtual listener agents. In: International Conference on human-computer interaction, 2019; p. 53–63. Springer.
45. Hulsebos M, Hu K, Bakker M, Zgraggen E, Satyanarayan A, Kraska T, Demiralp Ça, Hidalgo C. Sherlock: a deep learning approach to semantic data type detection. In: Proceedings of the 25th ACM SIGKDD International Conference on knowledge discovery & data mining, 2019; p. 1500–508.
46. Imamverdiyev Y, Abdullayeva F. Deep learning method for denial of service attack detection based on restricted Boltzmann machine. Big Data. 2018;6(2):159–69.
47. Islam MZ, Islam MM, Asraf A. A combined deep cnn-lstm network for the detection of novel coronavirus (covid-19) using x-ray images. Inf Med Unlock. 2020;20:100412.
48. Ismail WN, Hassan MM, Alsalamah HA, Fortino G. Cnn-based health model for regular health factors analysis in internet-of-medical things environment. IEEE. Access. 2020;8:52541–9.
49. Jangid H, Singhal S, Shah RR, Zimmermann R. Aspect-based financial sentiment analysis using deep learning. In: Companion Proceedings of the The Web Conference 2018, 2018; p. 1961–966.
50. Kaelbling LP, Littman ML, Moore AW. Reinforcement learning: a survey. J Artif Intell Res. 1996;4:237–85.
51. Kameoka H, Li L, Inoue S, Makino S. Supervised determined source separation with multichannel variational autoencoder. Neural Comput. 2019;31(9):1891–914.
52. Karhunen J, Raiko T, Cho KH. Unsupervised deep learning: a short review. In: Advances in independent component analysis and learning machines. 2015; p. 125–42.
53. Kawde P, Verma GK. Deep belief network based affect recognition from physiological signals. In: 2017 4th IEEE Uttar Pradesh Section International Conference on electrical, computer and electronics (UPCON), 2017; p. 587–92. IEEE.
54. Kim J-Y, Seok-Jun B, Cho S-B. Zero-day malware detection using transferred generative adversarial networks based on deep autoencoders. Inf Sci. 2018;460:83–102.
55. Kingma DP, Welling M. Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114, 2013.
56. Kingma DP, Welling M. An introduction to variational autoencoders. arXiv preprint arXiv:1906.02691, 2019.
57. Kiran PKR, Bhasker B. Dnnrec: a novel deep learning based hybrid recommender system. Expert Syst Appl. 2020.
58. Kloenne M, Niehaus S, Lampe L, Merola A, Reinelt J, Roeder I, Scherf N. Domain-specific cues improve robustness of deep learning-based segmentation of ct volumes. Sci Rep. 2020;10(1):1–9.
59. Kohonen T. The self-organizing map. Proc IEEE. 1990;78(9):1464–80.
60. Kohonen T. Essentials of the self-organizing map. Neural Netw. 2013;37:52–65.
61. Kök İ, Şimşek MU, Özdemir S. A deep learning model for air quality prediction in smart cities. In: 2017 IEEE International Conference on Big Data (Big Data), 2017; p. 1983–990. IEEE.
62. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. 2012; p. 1097–105.
63. Latif S, Rana R, Younis S, Qadir J, Epps J. Transfer learning for improving speech emotion classification accuracy. arXiv preprint arXiv:1801.06353, 2018.
64. LeCun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015;521(7553):436–44.
65. LeCun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proc IEEE. 1998;86(11):2278–324.
66. Li B, François-Lavet V, Doan T, Pineau J. Domain adversarial reinforcement learning. arXiv preprint arXiv:2102.07097, 2021.
67. Li T-HS, Kuo P-H, Tsai T-N, Luan P-C. Cnn and lstm based facial expression analysis model for a humanoid robot. IEEE Access. 2019;7:93998–4011.
68. Liu C, Cao Y, Luo Y, Chen G, Vokkarane V, Yunsheng M, Chen S, Hou P. A new deep learning-based food recognition system for dietary assessment on an edge computing service infrastructure. IEEE Trans Serv Comput. 2017;11(2):249–61.

69. Liu W, Wang Z, Liu X, Zeng N, Liu Y, Alsaadi FE. A survey of deep neural network architectures and their applications. Neurocomputing. 2017;234:11–26.
70. López AU, Mateo F, Navío-Marco J, Martínez-Martínez JM, Gómez-Sanchís J, Vila-Francés J, Serrano-López AJ. Analysis of computer user behavior, security incidents and fraud using self-organizing maps. Comput Secur. 2019;83:38–51.
71. Lopez-Martin M, Carro B, Sanchez-Esguevillas A. Application of deep reinforcement learning to intrusion detection for supervised problems. Expert Syst Appl. 2020;141:112963.
72. Ma X, Yao T, Menglan H, Dong Y, Liu W, Wang F, Liu J. A survey on deep learning empowered iot applications. IEEE Access. 2019;7:181721–32.
73. Makhzani A, Frey B. K-sparse autoencoders. arXiv preprint arXiv:1312.5663, 2013.
74. Mandic D, Chambers J. Recurrent neural networks for prediction: learning algorithms, architectures and stability. Hoboken: Wiley; 2001.
75. Marlin B, Swersky K, Chen B, Freitas N. Inductive principles for restricted boltzmann machine learning. In: Proceedings of the Thirteenth International Conference on artificial intelligence and statistics, p. 509–16. JMLR Workshop and Conference Proceedings, 2010.
76. Masud M, Muhammad G, Alhumyani H, Alshamrani SS, Cheikhrouhou O, Ibrahim S, Hossain MS. Deep learning-based intelligent face recognition in iot-cloud environment. Comput Commun. 2020;152:215–22.
77. Memisevic R, Hinton GE. Learning to represent spatial transformations with factored higher-order boltzmann machines. Neural Comput. 2010;22(6):1473–92.
78. Minaee S, Azimi E, Abdolrashidi AA. Deep-sentiment: sentiment analysis using ensemble of cnn and bi-lstm models. arXiv preprint arXiv:1904.04206, 2019.
79. Naeem M, Paragliola G, Coronato A. A reinforcement learning and deep learning based intelligent system for the support of impaired patients in home treatment. Expert Syst Appl. 2021;168:114285.
80. Niu Z, Zhong G, Hui Yu. A review on the attention mechanism of deep learning. Neurocomputing. 2021;452:48–62.
81. Pan SJ, Yang Q. A survey on transfer learning. IEEE Trans Knowl Data Eng. 2009;22(10):1345–59.
82. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, et al. Pytorch: An imperative style, high-performance deep learning library. Adv Neural Inf Process Syst. 2019;32:8026–37.
83. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, et al. Scikit-learn: machine learning in python. J Mach Learn Res. 2011;12:2825–30.
84. Pi Y, Nath ND, Behzadan AH. Convolutional neural networks for object detection in aerial imagery for disaster response and recovery. Adv Eng Inf. 2020;43:101009.
85. Piccialli F, Giampaolo F, Prezioso E, Crisci D, Cuomo S. Predictive analytics for smart parking: A deep learning approach in forecasting of iot data. ACM Trans Internet Technol (TOIT). 2021;21(3):1–21.
86. Puterman ML. Markov decision processes: discrete stochastic dynamic programming. Hoboken: Wiley; 2014.
87. Qu X, Lin Y, Kai G, Linru M, Meng S, Mingxing K, Mu L, editors. A survey on the development of self-organizing maps for unsupervised intrusion detection. Mob Netw Appl. 2019; p. 1–22.
88. Rahman MW, Tashfia SS, Islam R, Hasan MM, Sultan SI, Mia S, Rahman MM. The architectural design of smart blind assistant using iot with deep learning paradigm. Internet of Things. 2021;13:100344.
89. Ren J, Green M, Huang X. From traditional to deep learning: fault diagnosis for autonomous vehicles. In: Learning control. Elsevier. 2021; p. 205–19.
90. Rifai S, Vincent P, Muller X, Glorot X, Bengio Y. Contractive auto-encoders: Explicit invariance during feature extraction. In: Icml, 2011.
91. Rosa RL, Schwartz GM, Ruggiero WV, Rodríguez DZ. A knowledge-based recommendation system that includes sentiment analysis and deep learning. IEEE Trans Ind Inf. 2018;15(4):2124–35.
92. Sarker IH. Context-aware rule learning from smartphone data: survey, challenges and future directions. J Big Data. 2019;6(1):1–25.
93. Sarker IH. A machine learning based robust prediction model for real-life mobile phone data. Internet of Things. 2019;5:180–93.
94. Sarker IH. Cyberlearning: effectiveness analysis of machine learning security modeling to detect cyber-anomalies and multi-attacks. Internet of Things. 2021;14:100393.
95. Sarker IH. Data science and analytics: an overview from data-driven smart computing, decision-making and applications perspective. SN Comput Sci. 2021.
96. Sarker IH. Deep cybersecurity: a comprehensive overview from neural network and deep learning perspective. SN Computer. Science. 2021;2(3):1–16.
97. Sarker IH. Machine learning: Algorithms, real-world applications and research directions. SN Computer. Science. 2021;2(3):1–21.
98. Sarker IH, Abushark YB, Alsolami F, Khan AI. Intrudtree: a machine learning based cyber security intrusion detection model. Symmetry. 2020;12(5):754.
99. Sarker IH, Abushark YB, Khan AI. Contextpca: Predicting context-aware smartphone apps usage based on machine learning techniques. Symmetry. 2020;12(4):499.
100. Sarker IH, Colman A, Han J. Recencyminer: mining recency-based personalized behavior from contextual smartphone data. J Big Data. 2019;6(1):1–21.
101. Sarker IH, Colman A, Han J, Khan AI, Abushark YB, Salah K. Behavdt: a behavioral decision tree learning to build user-centric context-aware predictive model. Mob Netw Appl. 2020;25(3):1151–61.
102. Sarker IH, Colman A, Kabir MA, Han J. Individualized time-series segmentation for mining mobile phone user behavior. Comput J. 2018;61(3):349–68.
103. Sarker IH, Furhad MH, Nowrozy R. Ai-driven cybersecurity: an overview, security intelligence modeling and research directions. SN Computer. Science. 2021;2(3):1–18.
104. Sarker IH, Hoque MM, Uddin MK. Mobile data science and intelligent apps: concepts, ai-based modeling and research directions. Mob Netw Appl. 2021;26(1):285–303.
105. Sarker IH, Kayes ASM. Abc-ruleminer: User behavioral rule-based machine learning method for context-aware intelligent services. J Netw Comput Appl. 2020;168:102762.
106. Sarker IH, Kayes ASM, Badsha S, Alqahtani H, Watters P, Ng A. Cybersecurity data science: an overview from machine learning perspective. J Big data. 2020;7(1):1–29.
107. Sarker IH, Kayes ASM, Watters P. Effectiveness analysis of machine learning classification models for predicting personalized context-aware smartphone usage. J Big Data. 2019;6(1):1–28.
108. Sarker IH, Salah K. Appspred: predicting context-aware smartphone apps using random forest learning. Internet of Things. 2019;8:100106.

109. Satt A, Rozenberg S, Hoory R. Efficient emotion recognition from speech using deep learning on spectrograms. In: Interspeec, 2017; p. 1089–1093.
110. Sevakula RK, Singh V, Verma NK, Kumar C, Cui Y. Transfer learning for molecular cancer classification using deep neural networks. IEEE/ACM Trans Comput Biol Bioinf. 2018;16(6):2089–100.
111. Sujay Narumanchi H, Ananya Pramod Kompalli Shankar A, Devashish CK. Deep learning based large scale visual recommendation and search for e-commerce. arXiv preprint arXiv:1703.02344, 2017.
112. Shao X, Kim CS. Multi-step short-term power consumption forecasting using multi-channel lstm with time location considering customer behavior. IEEE Access. 2020;8:125263–73.
113. Siami-Namini S, Tavakoli N, Namin AS. The performance of lstm and bilstm in forecasting time series. In: 2019 IEEE International Conference on Big Data (Big Data), 2019; p. 3285–292. IEEE.
114. Ślusarczyk B. Industry 4.0: are we ready? Pol J Manag Stud. 2018; p. 17
115. Sumathi P, Subramanian R, Karthikeyan VV, Karthik S. Soil monitoring and evaluation system using edl-asqe: enhanced deep learning model for ioi smart agriculture network. Int J Commun Syst. 2021; p. e4859.
116. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A. Going deeper with convolutions. In: Proceedings of the IEEE Conference on computer vision and pattern recognition, 2015; p. 1–9.
117. Tan C, Sun F, Kong T, Zhang W, Yang C, Liu C. A survey on deep transfer learning. In: International Conference on artificial neural networks, 2018; p. 270–279. Springer.
118. Vesanto J, Alhoniemi E. Clustering of the self-organizing map. IEEE Trans Neural Netw. 2000;11(3):586–600.
119. Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol P-A, Bottou L. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. J Mach Learn Res. 2010;11(12).
120. Wang J, Liang-Chih Yu, Robert Lai K, Zhang X. Tree-structured regional cnn-lstm model for dimensional sentiment analysis. IEEE/ACM Trans Audio Speech Lang Process. 2019;28:581–91.
121. Wang S, Wan J, Li D, Liu C. Knowledge reasoning with semantic data for real-time data processing in smart factory. Sensors. 2018;18(2):471.
122. Wang W, Zhao M, Wang J. Effective android malware detection with a hybrid model based on deep autoencoder and convolutional neural network. J Ambient Intell Humaniz Comput. 2019;10(8):3035–43.
123. Wang X, Liu J, Qiu T, Chaoxu M, Chen C, Zhou P. A real-time collision prediction mechanism with deep learning for intelligent transportation system. IEEE Trans Veh Technol. 2020;69(9):9497–508.
124. Wang Y, Huang M, Zhu X, Zhao L. Attention-based lstm for aspect-level sentiment classification. In: Proceedings of the 2016 Conference on empirical methods in natural language processing, 2016; p. 606–615.
125. Wei P, Li Y, Zhang Z, Tao H, Li Z, Liu D. An optimization method for intrusion detection classification model based on deep belief network. IEEE Access. 2019;7:87593–605.
126. Weiss K, Khoshgoftaar TM, Wang DD. A survey of transfer learning. J Big data. 2016;3(1):9.
127. Xin Y, Kong L, Liu Z, Chen Y, Li Y, Zhu H, Gao M, Hou H, Wang C. Machine learning and deep learning methods for cybersecurity. Ieee access. 2018;6:35365–81.
128. Xu W, Sun H, Deng C, Tan Y. Variational autoencoder for semi-supervised text classification. In: Thirty-First AAAI Conference on artificial intelligence, 2017.
129. Xue Q, Chuah MC. New attacks on rnn based healthcare learning system and their detections. Smart Health. 2018;9:144–57.
130. Yousefi-Azar M, Hamey L. Text summarization using unsupervised deep learning. Expert Syst Appl. 2017;68:93–105.
131. Yuan X, Shi J, Gu L. A review of deep learning methods for semantic segmentation of remote sensing imagery. Expert Syst Appl. 2020;p. 114417.
132. Zhang G, Liu Y, Jin X. A survey of autoencoder-based recommender systems. Front Comput Sci. 2020;14(2):430–50.
133. Zhang X, Yao L, Huang C, Wang S, Tan M, Long Gu, Wang C. Multi-modality sensor data classification with selective attention. arXiv preprint arXiv:1804.05493, 2018.
134. Zhang X, Yao L, Wang X, Monaghan J, Mcalpine D, Zhang Y. A survey on deep learning based brain computer interface: recent advances and new frontiers. arXiv preprint arXiv:1905.04149, 2019; p. 66.
135. Zhang Y, Zhang P, Yan Y. Attention-based lstm with multi-task learning for distant speech recognition. In: Interspeech, 2017; p. 3857–861.