International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Deep Learning and its Application: A Review

Avinash H. Hedaoo

Dept. of Computer Science, Prerna College of Commerce, Nagpur -44009, (M.S.) India

ABSTRACT

Click here and insert your abstract text. Nowadays, deep learning is a current and a stimulating field of machine learning. Deep learning is the most effective, supervised, time and cost efficient machine learning approach. Deep learning is not a restricted learning approach, but it abides various procedures and topographies which can be applied to an immense speculum of complicated problems. The technique learns the illustrative and differential features in a very stratified way. Deep learning methods have made a significant breakthrough with appreciable performance in a wide variety of applications with useful security tools. It is considered to be the best choice for discovering complex architecture in high-dimensional data by employing back propagation algorithm. Deep learning has exploded in the public consciousness, primarily as predictive and analytical products suffuse our world, in the form of numerous human-centered smart-world systems, including targeted advertisements, natural language assistants and interpreters, and prototype self-driving vehicle systems. Yet to most, the underlying mechanisms that enable such human-centered smart products remain obscure. In contrast, researchers across disciplines have been incorporating deep learning into their research to solve problems that could not have been approached before. In this paper, the author seeks to provide a thorough investigation of deep learning in its applications and mechanisms. The state of the art review further provides a general overview on the novel concept and the ever-increasing advantages and popularity of deep learning. Finally, the paper ends with the conclusion.

Keywords: Machine learning algorithm, deep learning, neural networks, survey, Algorithms, supervised and unsupervised learning

1. Introduction

The term "Deep Learning" (DL) was first introduced to Machine Learning (ML) in 1986, and later used for Artificial Neural Networks (ANN) in 2000 (Schmidhuber, 2015). Deep learning methods are composed of multiple layers to learn features of data with multiple levels of abstraction (LeCun et al., 2015). DL approaches allow computers to learn complicated concepts by building them out of simpler ones (Goodfellow et al., 2016). For Artificial Neural Networks (ANN), Deep Learning (DL) aka hierarchical learning (Deng and Yu, 2014) is about assigning credits in many computational stages accurately, to transform the aggregate activation of the network (Schmidhuber, 2014). To learn complicated functions, deep architectures are used with multiple levels of abstractions i.e. non-linear operations; e.g. ANNs with many hidden layers (Bengio, 2009). To sum it accurately, Deep Learning is a sub-field of Machine Learning, which uses many levels of non-linear information processing and abstraction, for supervised or unsupervised feature learning and representation, classification and pattern recognition (Deng and Yu, 2014). Deep Learning i.e. Representation Learning is class or sub-field of Machine Learning. Recent deep learning methods are mostly said to be developed since 2006 (Deng, 2011) (Matiur Rahman Minar, Jibon Naher Jul 2018).

Along with Big Data and Analytics, Cloud/Edge Computing-based Big Computing (W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, 2014), (W. Yu et al.,2017), and the Internet of Things (IoT)/Cyber-Physical Systems (CPS) , the topic of Deep Learning has come to dominate industry and research spheres for the development of a variety of smart- world systems. Deep learning has shown significant potential in approximating and reducing large, complex datasets into highly accurate predictive and transformational output, greatly facilitating human-centered smart systems (X.W. Chen and X. Lin,2014), (N. D. Nguyen, T. Nguyen, and S. Nahavandi, 2017). In contrast to complex hard-coded programs developed for a sole inflexible task, deep learning architectures can be applied to all types of data, be they visual, audio, numerical, text, or some combination. In addition, advanced deep learning platforms are becoming ever more sophisticated, often open source and available for widespread use (William Grant Hatcher and Wei Yu, 2018). Active researchers in this area include those at University of Toronto, New York University, University of Montreal, Microsoft Research, Google, IBM Research, Baidu, Facebook, Stanford University, University of Michigan, MIT, University of Washington, and numerous other places. These researchers have demonstrated successes of deep learning in diverse applications of computer vision, phonetic recognition, voice search, conversational speech recognition, speech and image feature coding, semantic utterance classification, hand-writing recognition, audio processing, visual object recognition, information retrieval, and even in the analysis of molecules that may lead to discovering new drugs as reported recently in (Arel et al., 2014, Li Deng, 2014). In recent years, various deep architectures with different learning paradigm are quickly introduced to develop machines that can perform similar to human or even better in different domains of application such as medical diagnosis, self-driving cars, natural language and image processing, and predictive forecasting (Sengupta S, et al., 2020, Xizhao Wang, 2020) With the unceasing growth of IoT and smart-world systems driven by the advance of CPS, in which all devices are network connected and able to communicate sensed data and monitor physical objects, larger and larger datasets are becoming available for the application of deep learning, poised to materially impact our daily lives. As a solution to the processing, dimensionality reduction, compression, and extraction of such Big Data, deep learning provides the most immediately relevant and appropriate tools, enabling the rapid analysis of complex data that spans a variety of modalities (William Grant Hatcher and Wei Yu, 2018).

Contributions of this paper are outlined as follows:

- This review almost provides a deep survey of the most important aspects of deep learning. This review helps researchers and students to have a good understanding.
- This paper explains DL in deep which the most popular deep learning algorithm by describing the concepts, theory, and state-of-the-art architectures.
- Reviews current challenges (limitations) of Deep Learning including lack of training data, Vanishing gradient problem, Exploding Gradient Problem, and Under specification.

The remainder of this paper is as follows. In Section II, provides a brief overview of deep learning. Section III, provides categorization of deep learning. Section IV, outlines basic deep learning architectures. In Section V, Deep learning methods are discussed in detail, Section VI provide detail discussion on Deep learning platforms, Section VII presents a broad review of the applications of deep learning. Finally, Section VIII, provides concluding remarks.

2. Evolution of Deep Learning

Artificial Neural Networks (ANN) have come a long way, as well as other deep models. First Generation of Artificial Neural networks(ANN) was composed of perceptrons in neural layers, which were limited in computations. The second- generation calculated the error rate and backpropagated the error. Restricted Boltzmann machine overcame the limitation of backpropagation, which made the learning easier. Then other networks are evolved eventually. Figure.1 illustrates a timeline showing the evolution of deep models (Amitha Mathew et al., 2021). The DBN-training procedure is not the only one that makes effective training of DNNs possible. Alternatively denoising autoencoder, “contractive” autoencoders , sparse encoding symmetric machine (SESM), can be used for effective training of DNNs. Historically, the use of the generative model of DBN to facilitate the training of DNNs plays an important role in igniting the interest of deep learning for speech feature coding and for speech recognition (Li Deng, 2014). This improvement has been complemented by the proliferation of cheaper processing units such as the general-purpose graphic processing unit (GPGPU) and large volume of data set (big data) to train from. In addition to the backpropagation algorithm and GPU, the adoption and advancement of ML and particularly Deep Learning can be attributed to the explosion of data or bigdata in the last 10 years (Ajay Shrestha et al., 2019). Other techniques and neural networks came as well e.g. Feedforward Neural Networks (FNN), Convolutional Neural Netowrks (CNN), Recurrent Neural Networks (RNN) etc. along with Deep Belief Networks, Autoencoders and such (Hinton, The next generation of neural networks). From that point, ANNs got improved and designed in various ways and for various purposes. Schmidhuber (2014), Bengio (2009), Deng and Yu (2014), Goodfellow et al. (2016), Wang et al. (2017a) etc. provided detailed overview on the evolution and history of Deep Neural Networks (DNN) as well as Deep Learning (DL). Deep architectures are multilayer non-linear repetition of simple architectures in most of the cases, which helps to obtain highly complex functions out of the inputs (LeCun et al., 2015, Matiur Rahman Minar et al., 2018). The concept of deeplearning was put forward in 2006 at first. Googel’s AlphaGo program defeated Lee Sedol in Go competition, which showed that deep learning had a strong learning ability. Google’s DeepDream is an excellent software which can not only classify images but generating strange and artificial paintings based on its own knowledge. Facebook announced a new artificial intelligence system named Deep Text. Deep Text is a deeplearning-based text understanding engine which can classify massive amounts of data, provide corresponding services after identifying users’ chatting messages and clean up span message. Baidu’s unmanned ground vehicle has accomplished road test under complicated road conditions. IFLYTEK started the research of speech recognition based on Deep Neural Network (DNN) 2010. They launched the first online Chinese speech recognition system and an advanced technology to recognize different languages. And now, they have published a high performance computing (HPC) platform in cooperation with Intel. (Xuedan Du et al., 2016). ML will continue to impact and disrupt all areas of our lives from education, finance, governance, healthcare, manufacturing, marketing and others (Ajay Shrestha et al., 2019). Now a days deep learning is used in a lot many applications such as Google's voice and image recognition, Netflix and Amazon's recommendation engines, Apple's Siri, automatic email and text replies, chatbots etc. (Amitha Mathew et al., 2021).

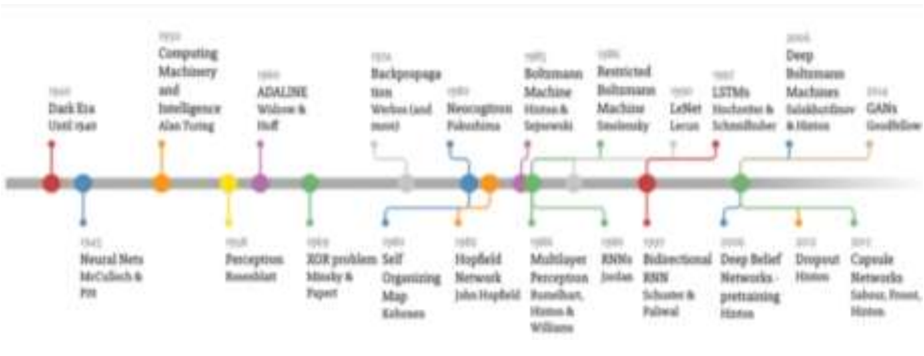

Fig. 1 - Evolution of Deep models [An intelligent framework for modelling and simulation of artificial neural networks (ANNs) based on augmented reality - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Timeline-of-deep-learning-adopted-from-20_fig1_344738477 [accessed 1 Sep, 2022]]

3. Categorization of Deep Learning

DL techniques are classified into three major categories: unsupervised, partially supervised (semi-supervised) and supervised. Furthermore, deep reinforcement learning (DRL), also known as RL, is another type of learning technique, which is mostly considered to fall into the category of partially supervised (and occasionally unsupervised) learning techniques.

3.1 Supervised Learning

In Supervised learning technique labeled data is used. In the case of supervised DL approaches, the environment has a set of inputs and corresponding outputs$(xt,yt)\sim\rho$. For example, if for input xt, the intelligent agent predicts $\hat{yt}$=(xt), the agent will receive a loss value $l(yt,\hat{yt})$. The agent will then iteratively modify the network parameters for better approximation of the desired outputs. After successful training, the agent will be able to get the correct answers to questions from the environment. There are different supervised learning approaches for deep leaning including Deep Neural Networks (DNN), Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN) including Long Short Term Memory (LSTM), and Gated Recurrent Units (GRU) (Md Zahangir Alom1 et al., 2018). The main advantage of this technique is the ability to collect data or generate a data output from the prior knowledge. However, the disadvantage of this technique is that decision boundary might be overstrained when training set doesn't own samples that should be in a class. Overall, this technique is simpler than other techniques in the way of learning with high performance (Laith Alzubaidi et al., 2021). The two primary learning tasks in supervised learning are classification and regression.

3.1.1 Classification

In classification, the output of the learning task will be of a finite set of classes. This can take the form of binary classification of only two classes (0 or 1), multi-class classification resulting in one class out of a set of three or more total classes (red, green, blue, etc.), as multi-label classification, where objects can belong to multiple binary classes (red or not red, and car or not car), and even as all pairs classification, in which every class in a finite set is directly compared to every other class in a binary way (Szegedy C et al., 2015). In all pairs classification, comparing red, green and blue, the resulting output would be test: red vs. green, red vs. blue, and green vs. blue. Examples for deep learning applications of classification include binary output in malware detection (Malicious and Benign, Shorfuzzaman M et al., 2020), as well as non-binary classification of handwritten number, as in the MNIST dataset (Le QV, 2013).

3.1.2 Regression

In contrast to classification, the output of regression learning is one or more continuous-valued numbers. Regression analysis is a convenient mechanism to provide scored labels equivalent to multi-label classification, where each item of a set has a probability of belonging (i.e., 0.997 red, 0.320 green, 0.008 blue). Regression has been applied in various areas, including monocular image object recognition for outdoor localization (T. Naseer and W. Burgard, 2017), among others (William Grant Hatcher and Wei Yu, 2018).

3.2 Semi-Supervised Learning

In this technique, the learning process is based on semi-labeled datasets. Occasionally, generative adversarial networks (GANs) and DRL are employed in the same way as this technique. In addition, RNNs, which include GRUs and LSTMs, are also employed for partially supervised learning. One of the advantages of this technique is to minimize the amount of labeled data needed. On other the hand, One of the disadvantages of this technique is irrelevant input feature present training data could furnish incorrect decisions. Text document classifier is one of the most popular example of an application of semi-supervised learning. Due to difficulty of obtaining a large amount of labeled text documents, semi-supervised learning is ideal for text document classification task (Laith Alzubaidi et al., 2021).

3.3 Unsupervised Learning

Unsupervised learning systems are ones that can learn without the presence of data labels. In this case, the agent learns the internal representation or important features to discover unknown relationships or structure within the input data. Often clustering, dimensionality reduction, and generative techniques are considered as unsupervised learning approaches. There are several members of the deep learning family that are good at clustering and non-linear dimensionality reduction, including Auto Encoders (AE), Restricted Boltzmann Machines (RBM), and the recently developed GAN. In addition, RNNs, such as LSTM and RL, are also used for unsupervised learning in many application domains (Md Zahangir Alom1 et al., 2018). The main disadvantages of unsupervised learning are unable to provide accurate information concerning data sorting and computationally complex. One of the most popular unsupervised learning approaches is clustering (Saeed MM et al., 2020, Laith Alzubaidi et al., 2021). The three primary learning tasks in unsupervised learning are Dimensionality Reduction, Clustering and Density Estimation.

3.3.1 Dimensionality Reduction

Dimensionality reduction can be carried out in various ways, including different forms of component and discriminant analysis. As an example, auto-encoders can transform input data into a reduced or encoded output for the purposes of data compression or storage space reduction. Examples of dimensionality reduction include the reduction of sequential data, such as video frames, to reduce noisy or redundant data while maintaining important features of the original data (B. Su et al. 2018), or the use of deep belief networks to reduce dimensionality of hyperspectral (400-2500 nm) images of landscapes to determine plant life content (D. M. S. Arsa et al., 2016).

3.3.2 Clustering

Clustering algorithms are used to statistically group data. Generally speaking, this occurs through the alternating selection of cluster centroids, and cluster membership. For example, k-means and fuzzy c-means clustering utilize the least mean square error of the distances between clusters and centroids. In the latter, fuzzing allows data membership in multiple cluster centroids, making the edges of the clusters ``fuzzy." Other clustering algorithms utilize the Gaussian Mixture Model (GMM), or other statistical and probabilistic mechanisms, instead of Euclidean Distance, as a means to make cluster selection (Y. P. Raykov, et al., 2016, X. Yang, 2017). In addition, deep neural network architectures can provide deep learning implementations for cluster analysis . Examples include the use of Self-Orienting Feature Maps (SOFMS) to satisfy real-time image registration , and the TSK_DBN fuzzy learning network that combines the Takagi-Sugeno-Kang (TSK) fuzzy system with a Deep Belief Network (DBN), among others.

3.3.3 Density Estimation

Density estimation, in general, is the statistical extraction or approximation of features of a data distribution, such as the extraction of densities of subgroups of data to evaluation correlations, or the approximation of the data distribution as a whole. Examples of density estimation include the estimation of power spectral density for noise reduction in binaural assisted listening devices (D. Marquardt and S. Doclo,2017), and intersection vehicle traffic density estimation utilizing CNNs on heterogeneous distributed video (C. Yeshwanth, et al., 2017, William Grant Hatcher and Wei Yu, 2018).

3.4 Reinforcement Learning

Reinforcement Learning operates on interacting with the environment, while supervised learning operates on provided sample data. This technique was developed in 2013 with Google Deep Mind (Laith Alzubaidi et al., 2021). Reinforcement learning uses a system of reward and punishment to train the algorithm. In this, the algorithm or an agent learns from its environment. The agent gets rewards for correct performance and penalty for incorrect performance. For example, consider the case of a self-driving car, the agent gets a reward for driving safely to destination and penalty for going off-road. Similarly, in the case of a program for playing chess, the reward state may be winning the game and the penalty for being checkmated. The agent tries to maximize the reward and minimize the penalty. In reinforcement learning, the algorithm is not told how to perform the learning; however, it works through the problem on its own (Amitha Mathew et al., 2021).

Deep Reinforcement Learning is a learning technique for use in unknown environments. Depending upon the problem scope or space, you can decide which type of RL needs to be applied for solving a task. If the problem has a lot of parameters to be optimized, DRL is the best way to go. If the problem has fewer parameters for optimization, a derivation free RL approach is good. An example of this is annealing, cross entropy methods, and SPSA (Md Zahangir Alom1 et al., 2018). The two primary means of reinforcement can be divided between policy search and value function approximation.

3.4.1 Policy Search

Policy search can be carried out by gradient-based (via backpropagation) or gradient-free (evolutionary) methods, to directly search for an optimal policy. These typically output parameters for a probability distribution, either for continuous or discreet actions, resulting in a stochastic policy (K. Arulkumaran et al., 2017). Though prior implementations of Google's AphaGo program, which were the first to beat a professional human player without handicap (D. Silver et al. 2016), were a hybrid of policy search and value function approaches, the most recent implementation, AlphaGo Zero, is entirely policy search-based, learned without any human input, and significantly outperforms the prior implementations.

3.4.2 Value Function

Value function methods operate by estimating the expected return of being in a given state, attempting to select an optimal policy, which chooses the action that maximizes the expected value given all actions for a given state. The policy can be improved by iterative evaluation and update of the value function estimate. The state-action value function, otherwise known as the quality function, is the source of Q-learning (K. Arulkumaran et al., 2017 , A. Bonarini, et al., 2009). An alternative to the quality function, the advantage function represents relative state-action values, as opposed to absolute state-action values (K. Arulkumaran et al., 2017). As a seminal work on the application of Q-learning and Deep Q-Networks (DQN), (V. Mnih et al.2015) implemented a DQN to play 49 different Atari 2600 videogames, observing four frames as environment data, extracting the game score as reward, with controller and button combinations encoded as actions. Their DQN implementation outperformed human users in the majority of games, as well as outperforming the best linear learners handily (William Grant Hatcher and Wei Yu, 2018).

4. Basic Architectures of Deep Neural Network (DNN)

Different names for deep learning architectures embrace deep belief networks, recurrent neural networks and deep neural networks. DNN can be constructed by adding multiple layers which are hidden layers in between the input layers and the output layers of Artificial Neural Network with various topologies. The deep neural network can model convoluted and non-linear relationships and generates models in which the object is treated as a layered configuration of primitives. These are such feed forward networks which have no looping and the flow of data is from the input layer to the output layer.

There are wide varieties of architectures and algorithms that are helpful in implementing the concept of deep learning. Table 1 depicts the year wise distribution in the architecture of deep learning.

Table 1: Years with the usage of architectures of deep learning

Year	Architecture of deep learning
1990–1995	Recurrent neural network
1995–2000	Long short term memory, convolutional neural network
2000–2005	Long short term memory, convolutional neural network
2005–2010	Deep belief network
2010–2017	Deep stacked network, gated recurrent unit

Here, we will discuss six basic types of the deep learning architectures as follows :-

4.1 Autoencoders

An AE is a deep neural network approach used for unsupervised feature learning with efficient data encoding and decoding. The main objective of auto encoder to learn and representation (encoding) of data, typically for data dimensionality reduction, compression, fusion and many more. This auto encoder technique consists of two parts: the encoder and the decoder. In the encoding phase, the input samples are mapped usually in the lower dimensional features space with a constructive feature representation. This approach can be repeated until the desired feature dimensional space is reached. Whereas in the decoding phase, we regenerate actual features from lower dimensional features with reverse processing (Md Zahangir Alom1 et al., 2018). In an autoencoder, the first layer is built as an encoding layer and transpose of that as a decoder (Amitha Mathew et al., 2021). The learning algorithm is based on the implementation of the backpropagation. Autoencoders extend the idea of principal component analysis (PCA) (Ajay Shrestha et al., 2019).

Autoencoders (AE) are neural networks (Laith Alzubaidi et al., 2021) where outputs are the inputs. AE takes the original input, encodes for compressed representation and then decodes to reconstruct the input (Wang). In a deep AE, lower hidden layers are used for encoding and higher ones for decoding, and error back-propagation is used for training (Deng and Yu, 2014).Goodfellow et al. (2016)(Matiur Rahman Minar et al., 2018)

Following are the types of Auto-encoders:

4.1.1 De-noising Auto-encoder

It is an advanced version of basic auto-encoders. To addresses the identity functions, these encoders corrupt the input and afterwards, reconstruct them. It is also called the stochastic version of the auto-encoders (Shaveta Dargan et al., 2019). In early Auto-Encoders (AE), encoding layer had smaller dimensions than the input layer. In Stacked Denoising Auto-Encoders (SDAE), encoding layer is wider than the input layer (Deng and Yu, 2014) (Matiur Rahman Minar, Jibon Naher Jul 2018).

4.1.2 Sparse Auto-encoder

These auto-encoders have the learning methods that automatically extract the features from the unlabeled data. Here the word sparse indicates that hidden units are allowed to fire only for the certain type of inputs and not too frequently (Shaveta Dargan et al., 2019).

4.1.3 Variational Auto-Encoder (VAE)

It consists of an encoder, decoder and a loss function. They are used for the designing of the complex models of the data that too with large datasets. It is also known as high resolution network (Shaveta Dargan et al., 2019). Variational Auto-Encoders (VAE) can be counted as decoders (Wang). VAEs are built upon standard neural networks and can be trained with stochastic gradient descent (Doersch, 2016)(Matiur Rahman Minar, Jibon Naher Jul 2018).

4.1.4 Contractive Auto-encoder (CAE)

These are robust networks as de-noising auto-encoders but the difference is that the contractive auto-encoders generate robustness in the networks through encoder function whereas de-noising auto-encoders work with the reconstruction process (Shaveta Dargan et al., 2019).

4.1.5 Transforming Autoencoders

Deep Auto-Encoders (DAE) can be transformation-variant, i.e., the extracted features from multilayers of non-linear processing could be changed due to learner. Transforming Auto-Encoders (TAE) work with both input vector and target output vector to apply transformation-invariant property and lead the codes towards a desired way (Deng and Yu, 2014) (Matiur Rahman Minar, Jibon Naher Jul 2018).

4.2 Convolutional Neural Networks

This network structure was first proposed by Fukushima in 1988 . It was not widely used however due to limits of computation hardware for training the network. In the 1990s, LeCun et al. applied a gradient-based learning algorithm to CNNs and obtained successful results for the handwritten digit classification problem (Md Zahangir Alom1 et al., 2018). The first CNN was developed by(LeCun et al., Shaveta Dargan et al., 2019) ConvNets are designed to process data that come in the form of multiple arrays, for example a colour image composed of three 2D arrays containing pixel intensities in the three colour channels. There are four key ideas behind ConvNets that take advantage of the properties of natural signals: local connections, shared weights, pooling and the use of many layers (Yann LeCun et al., 2015).

The architecture of a typical ConvNet is structured as a series of stages. The first few stages are composed of two types of layers: convolutional layers and pooling layers. Units in a convolutional layer are organized in feature maps, within which each unit is connected to local patches in the feature maps of the previous layer through a set of weights called a filter bank. The result of this local weighted sum is then passed through a non-linearity such as a ReLU. All units in a feature map share the same filter bank. Different feature maps in a layer use different filter banks. Mathematically, the filtering operation performed by a feature map is a discrete convolution, hence the name.

Although the role of the convolutional layer is to detect local conjunctions of features from the previous layer, the role of the pooling layer is to merge semantically similar features into one. Because the relative positions of the features forming a motif can vary somewhat, reliably detecting the motif can be done by coarse-graining the position of each feature. A typical pooling unit computes the maximum of a local patch of units in one feature map (or in a few feature maps). Neighbouring pooling units take input from patches that are shifted by more than one row or column, thereby reducing the dimension of the representation and creating an invariance to small shifts and distortions. Two or three stages of convolution, non-linearity and pooling are stacked, followed by more convolutional and fully-connected layers. Backpropagating gradients through a ConvNet is as simple as through a regular deep network, allowing all the weights in all the filter banks to be trained.

Deep neural networks exploit the property that many natural signals are compositional hierarchies, in which higher-level features are obtained by composing lower-level ones. In images, local combinations of edges form motifs, motifs assemble into parts, and parts form objects. Similar hierarchies exist in speech and text from sounds to phones, phonemes, syllables, words and sentences. The pooling allows representations to vary very little when elements in the previous layer vary in position and appearance.

The convolutional and pooling layers in ConvNets are directly inspired by the classic notions of simple cells and complex cells in visual neuroscience, and the overall architecture is reminiscent of the LGN–V1–V2–V4–IT hierarchy in the visual cortex ventral pathway.

4.2.1 Image Understanding With Deep Convolutional Networks

Since the early 2000s, ConvNets have been applied with great success to the detection, segmentation and recognition of objects and regions in images. These were all tasks in which labelled data was relatively abundant, such as traffic sign recognition, the segmentation of biological images particularly for connectomics, and the detection of faces, text, pedestrians and human bodies in natural images. A major recent practical success of ConvNets is face recognition.

Importantly, images can be labelled at the pixel level, which will have applications in technology, including autonomous mobile robots and self-driving cars. Other applications gaining importance involve natural language understanding and speech recognition.

ConvNets are now the dominant approach for almost all recognition and detection tasks and approach human performance on some tasks. A recent stunning demonstration combines ConvNets and recurrent net modules for the generation of image captions.

ConvNets are easily amenable to efficient hardware implementations in chips or field-programmable gate arrays. A number of companies such as NVIDIA, Mobileye, Intel, Qualcomm and Samsung are developing ConvNet chips to enable real-time vision applications in smartphones, cameras, robots and self-driving cars (Yann LeCun et al., 2015).

The different CNN architectures include Deep Max-Pooling Convolutional Neural Networks, Very Deep Convolutional Neural Networks, Network In Network, Region-based Convolutional Neural, Fast R-CNN, Faster R-CNN, Mask R-CNN, Multi-Expert R-CNN, Deep Residual Networks, Resnet in Resnet, ResNeXt and Capsule Networks (Matiur Rahman Minar, Jibon Naher Jul 2018).

4.3 Restricted Boltzmann Machine (RBM)

Restricted Boltzmann Machines (RBM) are special type of Markov random field containing one layer of stochastic hidden units i.e. latent variables and one layer of observable variables (Deng and Yu 2014, Goodfellow et al. 2016).Hinton and Salakhutdinov (2011) proposed a Deep Generative Model using Restricted Boltzmann Machines (RBM) for document processing (Matiur Rahman Minar, Jibon Naher Jul 2018). (Hinton and Sejnowski) proposed Boltzmann Machine (BM) in 1986. Boltzmann Machine is a random neural network belonging to the type of feedback neural network. Boltzmann Machine consists of some visible units (visible variables, i.e. data samples) and some hidden units (hidden variables), each visible unit is connected to all the hidden units, the visible variables and hidden variables are binary variables, the state is 0 or 1, 0 represents the neuron is in suppressed state, and 1 represents the neuron is in active state. (Sejnowski et al.) further proposed restricted Boltzmann machine (RBM). Input the training data to the visible layer, then the hidden layer detects the features of input data, the neurons are disconnected in the same layer but fully connected between two layers. The training of Restricted Boltzmann machine is faster than Autoencoder. In (Le Q V, 2011) proposed a more efficient optimization algorithm based on the stochastic gradient descent method. The traditional training method of RBM requires a large number of sampling steps, which makes the training efficiency of RBM still not high. The contrastive divergence proposed by Hinton solved this problem (Ruihui Mu et al., 2019).

4.4 Deep Stacking Network (DSN)

Deep Stacking Networks (DSN) is also acknowledged as deep convex networks. DSN is different from other traditional deep learning structures. It is called deep because of the fact that it contains a large number of deep individual networks where each network has its own hidden layers. The DSN believes that training is not a particular and isolated problem, but it holds the combination of individual training problems. The DSN is made up of a combination of modules which are part of the network and present in the architecture. There are three modules that work for the DSN. Here every module in the model is having an input zone, a single hidden zone and an output zone. Subroutines are placed one over the top of another with the input to the every module is taken as the outputs of the preceding layer and the authentic input vector. In DSN, every module is trained in isolation so as to make it productive and competent with the ability to work in coordination. The process of supervised method of training is practiced as the back-propagation for each module and not for the entire network. DSNs works superior than typical DBNs making it suitable and accepted network architecture (Shaveta Dargan et al., 2019).

4.5 Long Short Term Memory (LSTM)/Gated Recurrent Unit (GRU) Network

Hochreiter and Schmidhuber (1997) proposed Long Short-Term Memory (LSTM) which overcomes the error back-flow problems of Recurrent Neural Networks (RNN). LSTM is based on recurrent network along with gradient-based learning algorithm (Hochreiter and Schmidhuber, 1997) LSTM introduced self-loops to produce paths so that gradient can flow (Goodfellow et al., 2016). (Greff et al. 2017) provided large-scale analysis of Vanilla LSTM and eight LSTM variants for three uses i.e. speech recognition, handwriting recognition, and polyphonic music modeling. They claimed that eight variants of LSTM failed to perform significant improvement, while only Vanilla LSTM performs well (Greff et al., 2015, Shi et al. 2016b) proposed Deep Long Short-Term Memory (DLSTM), which is a stack of LSTM units for feature mapping to learn representations (Shi et al., 2016b) (Haohan Wang et al., 2017).

LSTM network is a variant of RNN. LSTM can work well for the time-sequential data. LSTM can avoid the disappearance of gradient at some extent by controlling the gate through the long memory and short memory. LSTM is different from RNN, because LSTM can determine which information are useful through the cell, the cell includes forget gate except for input gate and output gate. Input a message to LSTM, then determine the information to retain or forget according to whether or not match to the certification of algorithm. In the network structure of the LSTM, the input of the previous layer acts on the output through more paths, and the introduction of the gate makes the network have a focusing effect. LSTM generally included the input gate, forget gate, and output gate. The input gate is to supplement the latest input from the current input after the state of the “forgotten” part. The output gate will be based on the latest state C_t, the previous moment output and the current input x_t determine the output h_t at this moment. The forget gate is to make the recurrent neural network “forget” information that was not used before. LSTM can more naturally remember the input long before a long time. The storage unit Cell is a special unit that acts like an accumulator or a "gated leaky neuron": this unit has a direct connection from the previous state to the next state, so it can replicate its current state and accumulate all external signals, and due to the presence of the forget gate, the LSTM can learn to decide when to clear the contents of the memory unit.

Types of LSTM are Batch-Normalized LSTM (Cooijmans et al. (2016)), Pixel RNN (van den Oord et al. (2016b)), Bidirectional LSTM (W¨ollmer et al. (2010)), Variational Bi-LSTM (Shabanian et al. (2017)) (Matiur Rahman Minar, Jibon Naher Jul 2018).

4.6 Recurrent Neural Network (RNN)

RNNs can produced an output by comprising the new input with the latent vector. RNNs have three layers: the input layer, the hidden layer, the output layer. In theory, RNNs can work well for the sequential data, and complexity of the network is simple, because the current data is only dependent on the previous data. Fig. 2 shows the schematic diagram of RNN. The neurons are connected in the hidden layer, and RNNs can retain the former information. Where o and x represent the output information and input information respectively, h denotes the hidden unit, W, U, V represent weights. t denotes time, the input at time t and the previous state at time t are both determine the output of hidden unit at time t (Ruihui Mu et al., 2019).

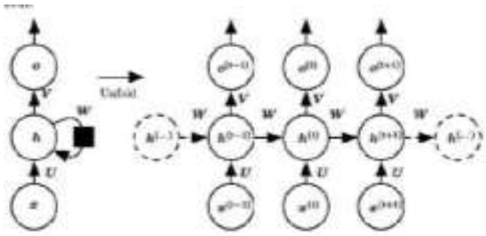

Fig : 2 : RNN expands over time

However, RNN’s sensitivity to the exploding gradient and vanishing problems represent one of the main issues with this approach . More specifically, during the training process, the reduplications of several large or small derivatives may cause the gradients to exponentially explode or decay. With the entrance of new inputs, the network stops thinking about the initial ones; therefore, this sensitivity decays over time. Furthermore, this issue can be handled using LSTM]. This approach offers recurrent connections to memory blocks in the network. Every memory block contains a number of memory cells, which have the ability to store the temporal states of the network. In addition, it contains gated units for controlling the flow of information. In very deep networks, residual connections also have the ability to considerably reduce the impact of the vanishing gradient issue (Laith Alzubaidi et al., 2021).

RNNs have been found to be very good at predicting the next character in the text or the next word in a sequence, but they can also be used for more complex tasks. [MM]

According to the different domains, RNNs have different variants, LSTM(long short-term memory network) is an example. Bidirectional RNNs, Echo State Networks and so on (Ruihui Mu et al., 2019). Various models of RNN are Recurrent Neural Networks with External Memory (RNN-EM), Gated Feedback Recurrent Neural Networks (GF-RNN), Conditional Random Fields as Recurrent Neural Networks(CRF-RNN), Quasi Recurrent Neural Networks (QRNN) (Matiur Rahman Minar, Jibon Naher Jul 2018).

5. Deep Learning Methods

Some of the powerful techniques that can be applied to deep learning algorithms to reduce the training time and to optimize the model are discussed in the following section. The merits and demerits of each method are comprised in the Table 1.

5.1 Backpropagation

Backpropagation first propagates the error term at output layer back to the layer at which parameters need to be updated, then uses standard gradient descent to update parameters with respect to the propagated error. Intuitively, the derivation of backpropagation is about organizing the terms when the gradient is expressed with the chain rule (Haohan Wng et al., 2017).

5.2 Stochastic Gradient Descent

Stochastic gradient descent (SGD) and its variants, are the dominant optimization methods in deep learning (Fengxiang He et al., 2021). SGD, updates are applied after running through a minibatch of n number of samples. Since we are updating the weights more frequently in SGD than in GD, we can converge towards global minimum much faster (Ajay Shrestha et al., 2019). However, because it is frequently updated, it takes extremely noisy steps in the direction of the answer, which in turn causes the convergence behavior to become highly unstable (Yann LeCun et al., 2015).

5.3 Learning Rate Decay

Learning rates have a huge impact on training DNN. It can speed up the training time, help navigate flat surfaces better and overcome pitfalls of non-convex functions (Ajay Shrestha et al., 2019). Adjusting the learning rate increases the performance and reduces the training time of stochastic gradient descent algorithms. The widely used technique is to reduce the learning rate gradually, in which we can make large changes at the beginning and then reduce the learning rate gradually in the training process. This allows fine-tuning the weights in the later stages (Amitha Mathew et al., 2021). There are three common approaches used for reducing the learning rate during training: constant, factored, and exponential decay (Md Zahangir Alom1 et al., 2018). Several innovative methods have been proposed like Delta-bar Algorithm, AdaGrad, RMSProp and Adam (Ajay Shrestha et al., 2019).

5.4 Dropout

(Srivastava et al., 2014) proposed Dropout to prevent neural networks from overfitting. Dropout is a neural network model-averaging regularization method by adding noise to its hidden units. It drops units from the neural network along with connections randomly during training (Matiur Rahman Minar, Jibon Naher Jul 2018). In doing this, the feature selection power is distributed equally across the whole group of neurons, as well as forcing the model to learn different independent features. (Laith Alzubaidi et al., 2021) Dropout can be used with any kind of neural networks, even in graphical models like RBM (Srivastava et al., 2014). A very recent proposed improvement of dropout is Fraternal Dropout (Anonymous, 2018a) for Recurrent Neural Networks (RNN) (Matiur Rahman Minar, Jibon Naher Jul 2018).

5.5 Max-Pooling

In max-pooling a filter is predefined, and this filter is then applied across the nonoverlapping sub-regions of the input taking the max of the values contained in the window as the output. Dimensionality, as well as the computational cost to learn several parameters, can be reduced using maxpooling (Amitha Mathew et al., 2021).

5.6 Batch normalization

As the network is getting trained with variations to weights and parameters, the distribution of actual data inputs at each layer of DNN changes too, often making them all too large or too small and thus making them difficult to train on networks, especially with activation functions that implement saturating nonlinearities, e.g., sigmoid and tanh functions. Iofee and Szegedy proposed the idea of batch normalization in 2015. It has made a huge difference in improving the training time and accuracy of DNN. It updates the inputs to have a unit variance and zero mean at each mini-batch (Ajay Shrestha et al., 2019). It is employed to reduce the “internal covariance shift” of the activation layers. The advantages of utilizing batch normalization are as follows:

- It prevents the problem of vanishing gradient from arising.
- It can effectively control the poor weight initialization.
- It significantly reduces the time required for network convergence (for large-scale datasets, this will be extremely useful).
- It struggles to decrease training dependency across hyper-parameters.

Chances of over-fitting are reduced, since it has a minor influence on regularization. (Laith Alzubaidi et al., 2021)

6. Deep learning platforms

Besides algorithms and data, the third component that enabled deep learning is the availability of fast GPU hardware and software to easily access it. This is often regarded as the most important element. Evaluating a CNN consists mainly of convolutions and large matrix multiplications , both suited for GPUs with their large number of computational cores and high memory bandwidth. The market is dominated by NVIDIA hardware together with CUDA and cuDNN libraries:

– Multi-GPU systems for fast network training.

– Embedded systems for deploying neural networks.

Specialized neural network processors like Google's Tensor Processing Unit [24] or Intel Nervana [35] provide further speed-up.[DD] There are a good number of open-source libraries and frameworks available for deep learning. Most of them are built for python programming language. Such as Theano (Bergstra et al., 2011), Tensorflow (Abadi et al., 2016), PyTorch, PyBrain (Schaul et al., 2010), Caffe (Jia et al., 2014), Blocks and Fuel (van Merri¨enboer et al., 2015), CuDNN (Chetlur et al., 2014), Honk (Tang and Lin, 2017), ChainerCV (Niitani et al., 2017), PyLearn2, Chainer, torch, neon etc. Bahrampour et al. (2015) did a comparative study of several deep learning frameworks (Matiur Rahman Minar, Jibon Naher Jul 2018).

7. Applications of Deep Learning

In this section, we review the primary applications of deep learning. A significant body of work toward the application of deep learning has grown steadily in the last few years. Particularly, the primary advances have been in the application of deep learning toward multimedia analysis, including image, audio, and natural language processing, which has afforded significant leaps in the state of the art for autonomous systems. Indeed, machine learning is fundamentally concerned with data fitting, the primary uses of which are optimization, discrimination, and prediction. In addition, advances in big data and cloud computing have created the potential for machine learning to flourish, enabling the requisite data collection and dissemination, as well as the computational capacity to execute deep models (W. Yu, G. Xu, Z. Chen, and P. Moulema, 2013). The existence of the data, and the nature of its potential have directly necessitated more accurate, generalized, and efficient learning mechanisms (William Grant Hatcher and Wei Yu, 2018).

7.1 Internet of Things

In considering the applications of deep learning for IoT, significant work has been carried out toward broadly applying typical categories like image/video/audio processing, text analysis, etc. across centralized and distributed cloud computing frameworks, utilizing IoT devices and some novel mechanisms(S. C. Kim, 2017). For instance, (S. C. Kim, 2017) proposed a deep learning system for use in identifying and tracking motion of individuals via Channel State Information (CSI) of IoT devices. Mohammadi et al. (M. Mohammadi, A. Al-Fuqaha, M. Guizani, and J. S. Oh, 2017) developed a semi-supervised deep reinforcement learning system to support smart city applications based on both structured and unstructured data. Utilizing Variational Autoencoders (VAEs), (William Grant Hatcher and Wei Yu, 2018) studied indoor user localization using Bluetooth Low Energy (BLE), collecting the received signal strength indicator (RSSI) from a grid of iBeacon devices. In addition, (C. Wu, H. P. Cheng, S. Li, H. Li, and Y. Chen, 2016) developed an efficient road scene segmentation deep learning model for embedded devices, termed ApesNet. Via time profiling and analysis, (William Grant Hatcher and Wei Yu, 2018) developed an asymmetric encoder-decoder network, and limited the size of large feature maps in convolutional layers. (S. Valipour, M. Siam, E. Stroulia, and M. Jagersand, 2016) developed a deep convolutional network for parking stall vacancy detection. (S. Park, M. Sohn, H. Jin, and H. Lee, 2016) designed a Situation Reasoning framework that extracts multiple low-level contexts in DNN modules, and combines them in a higher level Situation Reasoning module based on the Feature Comparison Model of cognitive psychology.

7.2 Cyber-Physical Systems

Cyber-Physical Systems (CPS) include the vertical layering of IoT devices, networking, service, applications, and command and control (C&C) platforms. Examples of CPS systems include smart transportation system with self- driving vehicles, smart cities, smart electrical grids, etc. (P. Zhao et al., 2018). More specifically, as applied to power generation, monitoring and control, (Mocanu et al., 2016) utilized Factored Four-Way Conditional Restricted Boltzmann Machines (FFW-CRBMs) and Disjunctive Factored Four-Way Conditional Restricted Boltzmann Machines (DFFW-CRBMs) to carry out energy disaggregation, and flexibility classification and prediction, on smart appliance data. Likewise, (Huang et al., 2017) investigated electrical load forecasting in the smart grid via deep learning. In addition, Zhao et al. leverage convolutional neural networks to develop a new deep heartbeat classification system, which can accurately analyzing the raw electrocardiogram (ECG) signal in healthcare smart-world system. Further, (Li et al., 2018) proposed a deep convolutional computation model, which is used for conducting hierarchical feature learning on IoT big data. Another relevant work is to adapt the ST-ResNet structure to predict the hourly distribution of crime in parceled areas in the city of Los Angeles (B. Wang, et. al., 2017) .

7.3 Network Management and Control

Deep learning offers a viable technique that can effectively learn the characteristics of the network and the behavior of users, leading to better network management and control decisions and outcomes. In this regard, very little research has been conducted. For example, (Zhu et al.) implemented stacked auto-encoders (SAEs) to realize Q-learning for transmission scheduling in cognitive IoT relays. Likewise, (Lopez-Martin et al., 2017) demonstrated flow statistics-based network traffic classification via deep neural networks. (Aminanto et al., 2018) developed a three-layer Wi-Fi impersonation attack detection system.

7.4 Secure Deep Learning

(Li et al., 2017) proposed multiple schemes for machine learning on multi-key homomorphic encrypted data in the cloud. A recent work by (Yuan et al., 2017) specifically investigated the space of the attacks that target only the inference mechanism through adversarial input. In the interim, (Goodfellow et al., 2014) proposed the generative adversarial network, pitting generator and discriminator networks against one another in a minimax game. In addition, (Pei et al., 2017) developed DeepXplore, the first whitebox testing framework for evaluating deep learning systems. (Booz et al., 2018) investigated how to fine-tune parameters of deep learning to improve the accuracy of detecting Android malware (William Grant Hatcher and Wei Yu, 2018).

7.5 Speech and audio

Deep learning and DNN started making impact in speech recognition in 2010, after close collaborations between academic and industrial researchers (see reviews in (Hinton, G. et al., 2012, Deng, L et al., 2013). A combination of three factors quickly spread the success of deep learning in speech recognition to the entire speech industry and academia: (1) minimally required decoder changes under the new DNN based speech recognizer deployment conditions enabled by the use of senones as the DNN output; (2) significantly lowered errors compared with the then-state-of-the-art GMM HMM system; and (3) training simplicity empowered by big data for training. By the ICASSP-2013 timeframe, at least 15 major speech recognition groups worldwide confirmed the experimental success of DNNs with very large tasks and with the use of raw speech spectral features away from MFCCs. Moreover, the most recent work of (Sainath, T. et al., 2013) shows that CNNs are also useful for large vocabulary continuous speech recognition and further demonstrates that multiple convolutional layers provide even more improvement when the convolutional layers use a large number of convolution kernels or feature maps. In addition to the RBM, DBN, CNN, and DSN, the deep-structured CRF, which stacks many layers of CRFs, have been successfully used in the task of language identification (Yu, D. et al., 2010), phone recognition (Yu, D., 2010), sequential labeling in natural language processing [96], and confidence calibration in speech recognition (Yu, D.; Deng, L.; Dahl, G., 2010, Yu, D.; Li, J.-Y.; Deng, L, 2010). Learning algorithms for RNNs have been dramatically improved since then, and better results have been obtained recently using RNNs (Graves, A. et al., 2006, Maas, A et al., 2012), especially when the structure of long short-term memory (LSTM) is embedded into the RNN with several layers and trained bi-directionally (Graves, A.;Mahamed, A.; Hinton, G. , 2013) RNNs have also been recently applied to audio/music processing applications (Bengio, Y et al., 2013), where the use of rectified linear hidden units instead of logistic or tanh non-linearities is explored in RNN.

In addition to speech recognition, the impact of deep learning has recently spread to speech synthesis, (In Ling et al., 2013, Ling, Z.; Deng, L.; Yu, D., 2013), the RBM and DBN as generative models are used to replace the traditional Gaussian models, achieving significant quality improvement, in both subjective and objective measures, of the synthesized voice. In the approach developed in (Kang, S. et al., 2013), the DBN as a generative model is used to represent joint distribution of linguistic and acoustic features. Both the decision trees and Gaussian models are replaced by the DBN. On the other hand, the study reported in (Zen, H. et al., 2013) makes use of the discriminative model of the DNN to represent the conditional distribution of the acoustic features given the linguistic features. No joint distributions are modeled. Finally, in (Fernandez, R. et al., 2013), the discriminative model of DNN is used as a feature extractor that summarizes high-level structure from the raw acoustic features. Such DNN features are then used as the input for the second stage of the system for the prediction of prosodic contour targets from contextual features in the fill speech synthesis system (Li Deng et al., 2014).

7.6 Biometrics

In 2009, an automatic speech recognition application was carried out to decrease the Phone Error Rate (PER) by using two different architectures of deep belief networks . In 2012, CNN method was applied within the framework of a Hybrid Neural Network - Hidden Markov Model (NN – HMM). As a result, a PER of 20.07 % was achieved. The PER obtained is better in comparison with a 3 – layer neural network baseline method previously applied. Smartphones and their camera resolution have been tested on iris recognition. Using mobile phones developed by different companies the iris recognition accuracy can reach up to 87% of effectiveness .

In terms of security, especially access control; deep learning is used in conjunction with biometric characteristics. DL was employed to speed up the developing and optimization of FaceSentinel face recognition devices. According to this manufacturer, their devices could expand their identification process from one–to–one to one–to– many in nine months. This engine advancement could have taken 10 man years without DL introduction. It accelerated the production and launch of the equipment. These devices are used in Heathrow airport in London and have the potential to be used for time and attendance and in the banking sector (R. Vargas et al., 2017).

7.7 Language modelling

The use of DNNs for LMs appeared more recently in (Wang Z et al., 2015). An NNLM is one that exploits the neural network's ability to learn distributed representations in order to reduce the impact of the curse of dimensionality. The temporally factored RBM was used for language modeling. As another example neural-network-based LMs, makes use of RNNs to build large scale language models, called RNNLMs. Fast and simple training algorithm also developed for NNLMs (Ruihui Mu et al., 2019).

7.8 Information Retrieval

Information retrieval (IR) is a process whereby a user enters a query into the automated computer system that contains a collection of many documents with the goal of obtaining a set of most relevant documents. A deep generative model of DBN is exploited for this purpose. The more advanced and recent approach to large-scale document retrieval (Web search) based on a specialized deep architecture, called deep-structured semantic model or deep semantic similarity model (DSSM), and its convolutional version (CDSSM),. The deep stacking network (DSN) has also been explored recently for IR with insightful results (Li Deng et al., 2013).

8. Conclusion

In this paper, the researcher provided a thorough overview of the neural networks and deep neural networks. Deep learning is indeed a fast growing application of machine learning. The rapid use of the algorithms of deep learning in different fields really shows its success and versatility. While the full-scale adoption of deep learning technologies in industry is ongoing, measured steps should be taken to ensure appropriate application of deep learning, as the subversion of deep learning models may result in significant loss of monetary value, trust, or even life in extreme cases. The author introduced many common and widely adopted deep learning frameworks, and considered them from the perspectives of design, extensibility and comparative efficacy. Additionally, thoroughly investigated the state-of-the-art in deep learning research. Given the widespread adoption of deep learning, and the inevitability of increasingly sophisticated cyber threats, the development of mechanisms to harden systems against adversarial data input is imperative. Some points are analysed in order to conclude review are, DL requires sizeable datasets (labeled data preferred) to predict unseen data and to train the models. Many of the current deep-learning models utilize supervised learning. Utilizing cloud computing offers the flexibility to train DL architectures. With the recent development in computational tools we will see more DL applications on mobile devices. Regarding the issue of lack of training data, transfer learning will be considered . I hope this work provides a valuable reference for researchers and computer science practitioners alike in considering the techniques, tools, and applications of deep learning, and provokes interest into areas that desperately need further consideration.

References

Matiur Rahman Minar, Jibon Naher Jul 2018, Recent Advances in Deep Learning , arXiv:1807.08169v1 [cs.LG]

William Grant Hatcher and Wei Yu, A Survey of Deep Learning: Platforms, Applications and Emerging Research Trends , VOLUME 6, 2018, IEEE Access.

X.W. Chen and X. Lin, ``Big data deep learning: Challenges and perspectives,'' IEEE Access, vol. 2, pp. 514_525, 2014.

N. D. Nguyen, T. Nguyen, and S. Nahavandi, ``System design perspective for human-level agents using deep reinforcement learning: A survey,'' IEEE Access, vol. 5, pp. 27091_27102, 2017.

T. Naseer and W. Burgard, ``Deep regression for monocular camera-based 6-DoF global localization in outdoor environments,'' in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS), Sep. 2017, pp. 1525_1530.

C. Yeshwanth, P. S. A. Sooraj, V. Sudhakaran, and V. Raveendran, ``Estimation of intersection traf_c density on decentralized architectures with deep networks,'' in Proc. Int. Smart Cities Conf. (ISC2), Sep. 2017, pp. 1_6.

D. Marquardt and S. Doclo, ``Noise power spectral density estimation for binaural noise reduction exploiting direction of arrival estimates,'' in Proc. IEEE Workshop Appl. Signal Process. Audio Acoust. (WASPAA), Oct. 2017, pp. 234_238. 19, 21 19 93

K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, ``Deep reinforcement learning: A brief survey,'' IEEE Signal Process. Mag., vol. 34, no. 6, pp. 26_38, Nov. 2017.

A. Bonarini, A. Lazaric, F. Montrone, and M. Restelli, ``Reinforcement distribution in fuzzy Q-learning,'' Fuzzy Sets Syst., vol. 160, no. 10, pp. 1420_1443, 2009.

V. Mnih et al., ``Human-level control through deep reinforcement learning,'' Nature, vol. 518, pp. 529_533, Feb. 2015.

D. Silver et al., ``Mastering the game of Go with deep neural networks and tree search,'' Nature, vol. 529, pp. 484_489, Jan. 2016.

W. Yu, G. Xu, Z. Chen, and P. Moulema, ``A cloud computing based architecture for cyber security situation awareness,'' in Proc. IEEE Conf. Commun. Netw. Secur. (CNS), Oct. 2013, pp. 488_492.

S. C. Kim, ``Device-free activity recognition using CSI big data analysis: A survey,'' in Proc. 9th Int. Conf. Ubiquitous Future Netw. (ICUFN), Jul. 2017, pp. 539_541.

M. Mohammadi, A. Al-Fuqaha, M. Guizani, and J. S. Oh, ``Semisupervised deep reinforcement learning in support of iot and smart city services,'' IEEE Internet Things J., vol. 5, no. 2, pp. 624_635, Apr. 2017.

C. Wu, H. P. Cheng, S. Li, H. Li, and Y. Chen, ``ApesNet: A pixel-wise ef_cient segmentation network for embedded devices,'' IET Cyber-Phys. Syst., Theory Appl., vol. 1, no. 1, pp. 78_85, 2016.

S. Valipour, M. Siam, E. Stroulia, and M. Jagersand, ``Parking-stall vacancy indicator system, based on deep convolutional neural networks,'' in Proc. IEEE 3rd World Forum Internet Things (WF-IoT), Dec. 2016, pp. 655_660.

S. Park, M. Sohn, H. Jin, and H. Lee, ``Situation reasoning framework for the internet of things environments using deep learning results,'' in Proc. IEEE Int. Conf. Knowl. Eng. Appl. (ICKEA), Sep. 2016, pp. 133_138.

P. Li et al., ``Multi-key privacy-preserving deep learning in cloud computing,'' Future Generat. Comput. Syst., vol. 74, pp. 76_85, Sep. 2017.

I. J. Goodfellow et al., ``Generative adversarial networks,'' ArXiv e-prints, Jun. 2014. [Online]. Available: http://adsabs.harvard.edu/abs/ 2014arXiv1406.2661G

J. Booz, J. McGiff, W. G. Hatcher, W. Yu, J. Nguyen, and C. Lu, ``Tuning deep learning performance for Android malware detection,'' in Proc. 19th IEEE/ACIS Int. Conf. Softw. Eng., Artif. Intell., Netw. Parallel/Distrib. Comput. (SNPD), Jun. 2018.

P. Zhao, W. Yu, D. Quan, and X. Yang, ``Deep learning-based detection scheme with raw ECG signal for wearable telehealth systems,'' Dept. Comput. Inf. Sci., Towson Univ., Towson, MD, USA, Tech. Rep. 2018-CS-021, Mar. 2018.

D. C. Mocanu, E. Mocanu, P. H. Nguyen, M. Gibescu, and A. Liotta, ``Big IoT data mining for real-time energy disaggregation in buildings,'' in Proc. IEEE Int. Conf. Syst., Man, (SMC), Oct. 2016, pp. 003765_003769.

B. Wang, P. Yin, A. L. Bertozzi, P. J. Brantingham, S. J. Osher, and J. Xin. (Nov. 2017). ``Deep learning for real-time crime forecasting and its ternarization.'' [Online]. Available: https://arxiv.org/abs/1711.08833

P. Li, Z. Chen, L. T. Yang, Q. Zhang, and M. J. Deen, ``Deep convolutional computation model for feature learning on big data in Internet of Things,'' IEEE Trans. Ind. Informat., vol. 14, no. 2, pp. 790_798, Feb. 2018.

M. Lopez-Martin, B. Carro, A. Sanchez-Esguevillas, and J. Lloret, ``Network traf_c classi_er with convolutional and recurrent neural networks for Internet of Things,'' IEEE Access, vol. 5, pp. 18042_18050, 2017.

M. E. Aminanto, R. Choi, H. C. Tanuwidjaja, P. D. Yoo, and K. Kim, ``Deep abstraction and weighted feature selection forWi-Fi impersonation detection,'' IEEE Trans. Inf. Forensics Security, vol. 13, no. 3, pp. 621_636, Mar. 2018.

X. Yuan, P. He, Q. Zhu, R. R. Bhat, and X. Li. (Dec. 2017). ``Adversarial examples: Attacks and defenses for deep learning.'' [Online]. Available: https://arxiv.org/abs/1712.07107

K. Pei, Y. Cao, J. Yang, and S. Jana. (May 2017). ``DeepXplore: Automated whitebox testing of deep learning systems.'' [Online]. Available: https://arxiv.org/abs/1705.06640

Li Deng ,A tutorial survey of architectures, algorithms, and applications for deep learning, Cambridge University Press, 2014 VOL 3 Arel, I.; Rose, C.; Karnowski, T.: Deep machine learning – a new frontier in artificial intelligence, in IEEE Computational Intelligence Mag., 5 (2010), 13–18.

Xizhao Wang et al., Recent advances in deep learning, International Journal of Machine Learning and Cybernetics, 2020, 11:747–750 Sengupta S, Basak S, Saikia P, Paul S, Tsalavoutis V, Atiah F, Ravi V, Peters A (2020) A review of deep learning with special emphasis on architectures, applications and recent trends. Knowl- Based Syst https ://doi.org/10.1016/j.knosy s.2020.10559 6

Hinton, G. et al.: Deep neural networks for acoustic modeling in speech recognition. IEEE Signal Process.Mag., 29 (6) (2012), 82–97.

Deng, L.; Hinton, G.; Kingsbury, B.: New types of deep neural network learning for speech recognition and related applications: an overview, in Proc. ICASSP, 2013.

Sainath, T.;Mohamed, A.; Kingsbury, B.; Ramabhadran, B.: Convolutional neural networks for LVCSR, in Proc. ICASSP, 2013.

Yu, D.; Wang, S.; Karam, Z.; Deng, L.: Language recognition using deep-structured conditional random fields, in Proc. ICASSP, 2010, 5030–5033.

Yu, D.; Deng, L.: Deep-structured hidden conditional randomfields for phonetic recognition, in Proc. Interspeech, September. 2010.

Yu, D.; Deng, L.; Dahl, G.: Roles of pre-training and fine-tuning in context-dependent DBN-HMMs for real-world speech recognition, in Proc. NIPSWorkshop, 2010.

Yu, D.; Li, J.-Y.; Deng, L.: Calibration of confidence measures in speech recognition. IEEE Trans. Audio Speech Lang., 19 (2010), 2461–2473.

Maas, A.; Le, Q.; O'Neil, R.; Vinyals, O.; Nguyen, P.; Ng, Y.: Recurrent neural networks for noise reduction in robust ASR, in Proc. Interspeech, 2012.

Graves, A.; Fernandez, S.; Gomez, F.; Schmidhuber, J.: Connectionist temporal classification: labeling unsegmented sequence data with recurrent neural networks, in Proc. ICML, 2006

Graves, A.;Mahamed, A.; Hinton, G.: Speech recognition with deep recurrent neural networks, in Proc. ICASSP, 2013.

Bengio, Y.; Boulanger, N.; Pascanu, R.: Advances in optimizing recurrent networks, in Proc. ICASSP, 2013.

Ling, Z.; Richmond,K.; Yamagishi, J.:Articulatory control ofHMMbased parametric speech synthesis using feature-space-switched multiple regression. IEEE Trans. Audio Speech Lang. Process., 21 (2013), 207–219.

Ling, Z.; Deng, L.; Yu, D.: Modeling spectral envelopes using restricted Boltzmann machines and deep belief networks for statistical parametric speech synthesis. IEEE Trans. Audio Speech Lang. Process., 21 (10) (2013), 2129–2139.

Kang, S.;Qian, X.;Meng, H.:Multi-distribution deep belief network for speech synthesis, in Proc. ICASSP, 2013, 8012–8016.

Zen, H.; Senior, A.; Schuster, M.: Statistical parametric speech synthesis using deep neural networks, in Proc. ICASSP, 2013, 7962– 7966.

Fernandez, R.; Rendel, A.; Ramabhadran, B.; Hoory, R.: F0 contour prediction with a deep belief network-Gaussian process hybrid Model, in Proc. ICASSP, 2013, 6885–6889.

R. Vargas, A. Mosavi, L. Ruiz, DEEP LEARNING: A REVIEW, Advances in Intelligent Systems and Computing, (2017).

Ruihui Mu, Xiaoqin Zeng A Review of Deep Learning Research, KSII Transactions On Internet And Information Systems Vol. 13, no. 4, Apr. 2019

Wang Z., Liu D., Yang J., Han W., Huang T, "Deep Networks for Image Super-Resolution with Sparse Prior," in Proc. of the International Conference on Computer Vision, Santiago, Chile, pp.370-278, 2015. Article (CrossRef Link).

Ajay Shrestha And Ausif Mahmood, Review of Deep Learning Algorithms and Architectures, Vol 7 2019 IEEE Access

Xuedan Du, Yinghao Cai, Sho Wang, and Leijie Zhang, Overview of Deep Learning, 31st outh Academic Annual Conference of Chinese Association of Automation Whan, China, 2016

Md Zahangir Alom1, Tarek M. Taha, Chris Yakopcic, Stefan Westberg, Paheding Sidike, Mst Shamima Nasrin, Brian C Van Essen, Abdul A S. Awwal, and Vijayan K. Asari, The History Began from AlexNet: A Comprehensive Survey on Deep Learning Approaches, Computer Vision and Pattern Recognition, 2018

Laith Alzubaidi, Jinglan Zhang, Amjad J. Humaidi, Ayad Al-Dujaili, Ye Duan, Omran Al-Shamma, J. Santamaria, Mohammed A. Fadhel, Muthana Al-Amidie and Laith Farhan, Review of deep learning: concepts, CNN architectures, challenges, applications, future directions, Journal of Big Data, 2021

Saeed MM, Al Aghbari Z, Alsharidah M. Big data clustering techniques based on spark: a literature review. PeerJ Comput Sci. 2020;6:321.

Shorfuzzaman M, Hossain MS. Metacovid: a siamese neural network framework with contrastive loss for n-shot diagnosis of COVID-19 patients. Pattern Recogn. 2020;113:107700.

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.

D. M. S. Arsa, G. Jati, A. J. Mantau, and I. Wasito, ``Dimensionality reduction using deep belief network in big data case study: Hyperspectral image classi_cation," in Proc. Int.Workshop Big Data Inf. Secur. (IWBIS), Oct. 2016, pp. 71_76.

B. Su, X. Ding, H. Wang, and Y. Wu, ``Discriminative dimensionality reduction for multi-dimensional sequences," IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 1, pp. 77_91, Jan. 2018.

Y. P. Raykov, A. Boukouvalas, F. Baig, and M. A. Little, ``What to do when K-Means clustering fails: A simple yet principled alternative algorithm," PLoS ONE, vol. 11, no. 9, p. e0162259, 2016.

Shaveta Dargan, Munish Kumar, Maruthi Rohit Ayyagari, Gulshan Kumar, A Survey of Deep Learning and Its Applications: A New Paradigm to Machine Learning , CIMNE, Barcelona, Spain 2019

Yann LeCun, Yoshua Bengio, Geoffrey Hinton, Vol 521, Nature, 2015

Haohan Wang, Bhiksha Raj, On the Origin of Deep Learning, https://doi.org/10.48550/arXiv.1702.07800, 2017

Fengxiang He, Dacheng Tao, Recent advances in deep learning theory, arXiv:2012.10931v2 [cs.LG] 2021

Li Deng and Dong Yu, Deep Learning Methods and Applications, Foundations and Trends R_ in Signal Processing, Vol. 7, 2013