Electronic Letters on Computer Vision and Image Analysis 24(1):51-80, 2025

An inclusive review on deep learning techniques and their scope in handwriting recognition

Sukhdeep Singh[a], Sudhir Rohilla[b] and Anuj Sharma[c]

[a]*D.M. College (Affiliated to Panjab University, Chandigarh), Moga, Punjab, India*
[b] *Department of Computer Science, Gopichand Arya Mahila College, Abohar, Punjab, India*
[c]*Department of Computer Science and Applications, Panjab University, Chandigarh, India*

Received 29th of August, 2024; accepted 9th of January 2025

Abstract

Deep learning encompasses a set of machine learning algorithms that can merge raw inputs into intermediate feature layers. These algorithms have shown impressive results across various fields. In particular, deep learning has achieved human-level performance in computer vision and pattern recognition across multiple domains. To achieve state-of-the-art performance in different areas, various architectures and activation functions are used to compute between hidden and output layers. This paper provides a survey of current research on deep learning in the field of handwriting recognition. Despite recent advancements in deep learning methods for improving speed and accuracy in handwriting recognition, extensive literature review reveals that deep learning has not completely transformed the field and still needs to address significant challenges. Nevertheless, there have been promising developments compared to previous advancements. Moreover, the lack of labeled data for training poses a challenge in this domain. However, the current analysis of handwriting recognition predicts that deep learning will bring about significant changes in various fields such as image processing, speech recognition, computer vision, machine translation, robotics and control, medical imaging, medical information processing, bio-informatics, natural language processing, cyber security, and more.
Key Words: Deep learning, Classification, Handwriting Recognition, CNN, RNN, LSTM.

1 Introduction

The intelligent act with synthesis and analysis of computational agents represents Artificial Intelligence (AI). Here, an agent is who completes the signed goal with various learning techniques and training of data. The agent when computationally represented, it is called computational agent [1, 2]. The artificial intelligence has made our life very exciting with state-of-the-art research in this area. However, the research in AI regularly demands new paradigms that could further help in error-free AI systems. The AI has many areas of research such as machine learning, data mining, intelligent tutoring, case-based reasoning, multi-agent planning, scheduling,

Correspondence to: Anuj Sharma: anujs@pu.ac.in, Sukhdeep Singh: sukha13@ymail.com

Recommended for acceptance by Angel D. Sappa
https://doi.org/10.5565/rev/elcvia.1986
ELCVIA ISSN:15108-5097
Published by Computer Vision Center / Universitat Autònoma de Barcelona, Barcelona, Spain

uncertain reasoning, natural language understanding and translation, vision, virtual reality, games, robotics and other topics [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]. The one of today's popular research fields in AI is Machine Learning (ML). The machine learning mainly includes intelligent system development using training of data. Therefore, the ML based system model developed with train data further decides the nature of future data as test data. The common techniques of machine learning are data understanding, regression, clustering, classification, dimension reduction, deep learning, big data, online learning etc [14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25]. Here, each ML technique offers uniqueness in terms of data handling, feature computation and respective output. Data understanding is data normalization and processing of data; regression is promising statistical area to understand continuous type of data; clustering allow class formation of data; classification distinguish data for various classes; dimension reduction reduce feature size of data and retain useful information for data; deep learning is promising data classification and understanding area; big data is dedicated to handle large amount of data using established scientific methods; online learning refers to handle data as it comes and not in conventional batch mode. The recent research in ML suggests that deep learning is one of the promising techniques to achieve high accuracy results. Therefore, deep learning studied by many scientists in recent past and it has been observed that suitable literature of deep learning always helps for readers working in this area. Especially, suitable review of deep learning two popular methods as Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) will be meaningful from research and development point of view [26, 27, 28].

In machine learning, representational learning allow automatically appropriate way to represent data. Deep learning is one such technique which includes representation learning feature. The progression of learned transformation helps deep learning to achieve automatic representation of data. This has been realised in recent past that deep learning-based applications achieved promising results for large amount of data problem, which were not possible to automate data representation before deep learning techniques. The popular deep learning model and its practical implementation based quintessential example is deep network. These deep networks are inspired from common machine learning technique as multilayer perceptron which is a common neural network algorithm. The depth of deep network as a computer program is organised in such a way that each layer meant for a specific purpose and it could recall other layers' computer memory when working in parallel. This way, the network with more depth allows many instructions in a sequence. The two approaches of deep networks as CNN and RNN proved state-of-the-art results in recent past. Further, LSTM and BLSTM are two particular forms of RNN. The two competitive techniques as CNN and RNN share many commonalities in working, however, they do differ in suitability of various types of data. Interestingly, hybrid technique using CNN and RNN is another feasible technique with promising results.

In literature, we find important surveys for handwriting recognition in indic as well as non Indic scripts [29] [30] [31] [32] [33], but these surveys are not specific to handwriting recognition using deep learning. Further, we find many studies that include the working of CNN and RNN. However, overall working of CNN and RNN with their architecture, mathematical formulation and implementation from review point of view needs attention and is presented in this paper. This paper has been presented with updated literature information and focused on theory as well as the implementation of deep networks. This review is aiming to answer the question of the working of deep networks, CNN and RNN, and their performance in handwriting recognition in recent past with the availability of latest system configurations. As a result, we highlight in summary the following contributions as: (i) deep networks outperform in handwriting recognition, (ii) the CNN and RNN results in state-of-the-art results for real life challenging datasets, (iii) the architecture of deep networks offers enormous scope to researchers to enhance its architecture, and further offer many areas of research to improve deep networks. Moreover, this paper also presents general observations of deep networks and their applications in handwriting recognition based on suitable findings from literature work. In this manner, it has been analysed that deep networks solely justify deep learning representative to real life handwriting recognition applications. This analysis is based on the results discussed in this paper using benchmark datasets with variants of deep learning approaches.

This paper maintains continuity as deep networks, CNN and RNN, architectures and DL results for handwriting recognition in a sequence for better understanding from the reader's point of view. The rest of the paper is organized as follows. This article describes the deep networks' fundamentals and evolution in section 2. The section 3 demonstrates different deep learning architectures. The section 4 presents existing literature results using deep networks as CNN and RNN in handwriting recognition. The section 6 presents the general observations based on literature finding for CNN and RNN. The last section 7 concludes this article with findings and scope of future work.

2 Deep Forward Networks

Deep learning quintessential are forward networks and forward networks are commonly called deep forward networks. The forward networks are also known as multi-layer perceptron as it follows multiple layers architecture using perceptron concept. The CNN and RNN are the emergent variants of deep learning framework based on neural networks. Therefore, prior understanding of neural networks is important in this review to know how neural networks work and CNN or RNN based on neural networks. Before the introduction of deep neural networks, this section illustrates the evolution of neural networks with common benchmark algorithms of literature including most preferred backpropagation algorithm.

In early works, McCulloch and Pitts neuron was designed in 1943, it mainly included the combination of logic functions with the concept of threshold [34]. The Hebb network was designed in 1949, which included two active neurons simultaneously with their strong inter-connections [35]. One major contribution was perceptron model in 1958 by Rossenblatt, which included weights in connection path with their adjustment [36]. In 1960, Adaline network was built with the ability to reduce difference between net input weights and out weights and resulted in minimizing the mean error rates [37]. One major development noticed for unsupervised learning in 1982, Kohonen introduced Kohonen self-organizing maps where inputs were clustered together to form output neurons [38]. This work was among initial findings to understand neural networks in supervised and unsupervised areas. One such study with fixed weights was done for Hopfield networks to act as associative memory nets [39]. In 1986, a complete neural network algorithm, with forward and backward ability to update weights and based on Multi-Layer Perceptron (MLP), was introduced as backpropagation algorithm [40]. This backpropagation algorithm was a complete multi-layer perceptron technique. This architecture included input, hidden and output layers, forward moves to update weights and backward moves to improve weights with propagated error information at output unit in each iteration. After backpropagation, neural networks witnessed many improvements subject to the nature of problem. Few networks were adaptive resonance theory, radial basis functions, neocognition until 1990 [41, 42, 43]. The main development with state-of-the-art results were reported with MLP until the introduction of convolution networks in 2000.

The MLP or deep forward network is a massively parallel distributed processor made up of simple processing units, which has a natural propensity for storing experiential knowledge and making it available for use [44]. It is a directed graph consisting of nodes with interconnecting synaptic and activation links with main properties as: each neuron is represented by a set of linear synaptic links with bias, and a possibly nonlinear activation link; the synaptic links of a neuron weight to their respective input signals; the weighted sum of the input signals defines the induced local field of the current neuron; thus, the activation link squashes the induced local field of the neuron to produce an output. The presence of one or more layers between input and output layers are called hidden layers and nodes of corresponding layers are hidden neurons. This enhances system learning capability and is referred as multi-layer networks and results in MLP. The major characteristics of MLP include: the model of each neuron in network includes nonlinear activation function; the network includes multiple hidden layers that enable network to learn complex tasks by progressively extracting more meaningful features by minimizing errors at output layer; the network exhibits high degree of connectivity determined by synapses of network and a change in network require change in synaptic connections or their weights. The working of MLP has been presented in 1, where input layer includes three nodes, two hidden layers with three nodes each and output layer with two nodes. In figure 1, forward move is shown with connected lines and backward moves with dotted lines. The input layer includes initialized values that are processed with weight vectors and each hidden

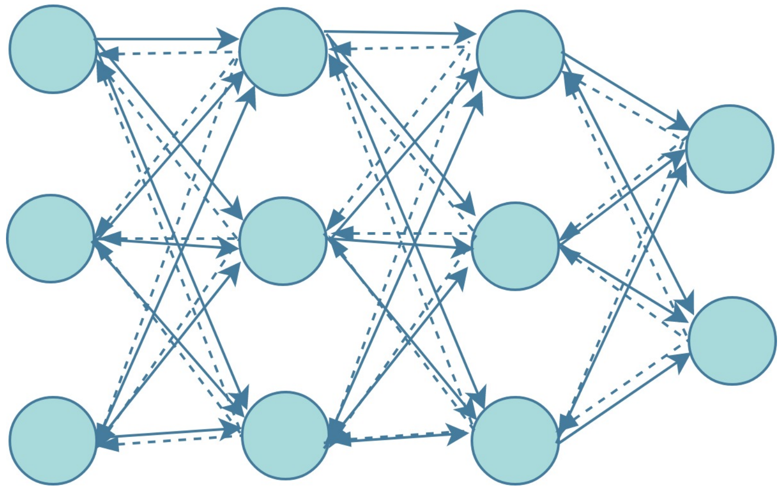

Figure 1: Overview of multi layer perceptron technique

layer node is updated in forward manner. The output layer computes final value and error is computed subject to target value against output value. Therefore, error values decide to move backward in order to minimize error values. This results in many iterations until the expected value of error is achieved. This way, it includes the computation of the function signal appearing at the output neuron, a continuous nonlinear function of input signal and synaptic weight associated with that neuron. Also, the computation of an estimate of gradient vector which is needed for backward pass of the network.

The MLP algorithm includes three major steps as forward computation, backward computation and weight updating. The working of these steps has been presented with an instance of one hidden layer to another consecutive hidden layer for neuron j to neuron k. The details of MLP including derivations have been studied extensively in literature. According to [44], for an MLP, the first step as forward computation results in output as $y_j^{(l)}$ for neuron j in layer l using:

$$\begin{gathered} v_j^{(l)}(n) = \textstyle\sum_{i=0}^{m_0} w_{ji}^{(l)}(n) y_i^{(l-1)}(n); \\ y_j^{(l)} = \phi_j(v_j(n)) \end{gathered} \tag{1}$$

The forward computation is performed in forward direction until the output layer neuron value is not computed. In backward computation, gradient value is computed for neuron j in hidden layer l as:

$$\begin{gathered} \delta_j^{(l)}(n) = e_j^{(L)}(n)\phi_j'(v_j^L(n)), \text{ for neuron } j \text{ in output layer } L; \\ \delta_j^{(l)}(n) = \phi_j'(v_j^l(n)) \textstyle\sum_k \delta_k^{(l+1)}(n) w_{kj}^{(l+1)}(n) \end{gathered} \tag{2}$$

For one forward and backward computation, it completes one iteration and updates the weight for next iteration. The adjustment of the synaptic weights of network in layer l is computed as:

$$w_{ji}^{(l)}(n+1) = w_{ji}^{(l)}(n) + \alpha[w_{ji}^{(l)}(n-1)] + \eta\delta_j^{(l)}(n) y_i^{(l-1)}(n) \tag{3}$$

In above three steps of MLP, w_{ji} is the weight from i to j; $v_j^{(l)}(n)$ is neuron j in layer l; $y_j^{(l)}$ is output of neuron j in layer l; $\delta_j^{(l)}(n)$ is gradient for neuron j in layer l; $e_j^{(L)}(n)$ is error signal at layer L for neuron j; ϕ_j' is differentiation of activation function; η and α are the learning rate and momentum constant respectively.

3 Deep Learning Architecture

The pattern recognition field has started using DL architectures extensively and image recognition [45] [46] has also attained good performance for recognizing faces [47], text recognition [48] [49] [50] [51] [52] and estimation of human poses [53]. Deep learning, either uses deep architectures of learning or hierarchical learning approaches, is a class of machine learning developed mostly after 2006. The DL architectures have basically made alteration of the traditional form of pattern recognition and have contributed a major development in various handwriting recognition tasks too. The traditional ML approaches perform better for lesser amounts of input data. When the data size increases beyond a certain limit, the performance of traditional machine learning approaches becomes steady, whereas deep learning performance increases with respect to the increment of data size. The key breakthrough of DL is that these models can perform feature extraction and classification automatically. Deep learning architectures with more than one hidden layer are referred to as multilayer perceptron. These architectures include deep feed forward neural network, CNN, RNN, LSTM and the deep generative models as deep belief networks, deep Boltzmann machines, generative adversarial networks [54] [55]. Deep learning architectures include to learn patterns in data, mapping of input function to outputs and many more, and it can be attained with specialized architectures only. Among DL architectures, the CNN and RNN: LSTM and BLSTM are the most commonly used vital architectures. LSTM and BLSTM are particularly designed for data in sequential form, have been used in the studies of pattern and handwriting recognition. One of the key problems with the RNN deep network is that the hidden layers are influenced by the input layer and consequently the output layer goes on decaying as it cycles through the network's recurrent connections, and this problem is known as vanishing gradient problem [56]. Such problem becomes a cause for an incomplete range of contextual information access by RNN, and the contextual information cannot be retained for a longer period of time by an architecture of RNN.

3.1 Convolutional Neural Network

The recognition of images largely utilizes the CNN model among various deep learning models. CNNs are a specialized type of multi-layer neural network, similar to other networks, and they employ back propagation algorithms for training. The difference lies in their architectures [57] [48]. The CNN is highly effective in capturing the spatial relationships within an image, as it focuses on the connections between adjacent pixels rather than those that are far apart. Additionally, the CNN's approach of weight sharing allows similar attributes such as texture and brightness to be applied across different areas of an image. This enables the CNN to effectively extract and generalize 2D features. The shape variations can be effectively absorbed by the CNN max-pooling layer. Further, the involvement of CNN with less parameters than a similar sized fully connected network has been made possible by sparse connection with tied weights.The CNN can be trained effectively using the gradient-based learning algorithm, which helps to alleviate the diminishing gradient problem. Text recognition with CNN presents a greater challenge compared to image recognition, as characters and words can vary in appearance due to different writers, writing styles, and writing surfaces. By leveraging deep learning techniques, promising outcomes can be achieved in pattern recognition. However, to achieve optimal results in pattern recognition using deep learning, it is essential to address various challenges and select the most suitable deep learning framework. As an illustration, for handwriting recognition, CNN was initially used in digits recognition [58]. The use of CNN and its different versions has been expanding to cover various other handwriting recognition tasks. The CNN is the most successful model for image analysis. Since late seventies, the work on CNNs has been done [59] and these were already applied for image analysis in medical field in 1995 [60]. The CNNs were first successfully applied in real-world application in LeNet [58] for recognition of handwritten digits. Regardless of CNNs initial achievement, its use did not get momentum until several new techniques were developed to train deep networks efficiently, and advancements were made in core computing systems. The watershed was a contribution [45] to the ImageNet challenge in 2012 and the CNN AlexNet won that competition by a huge margin. In recent years, further development has been done using related but deeper

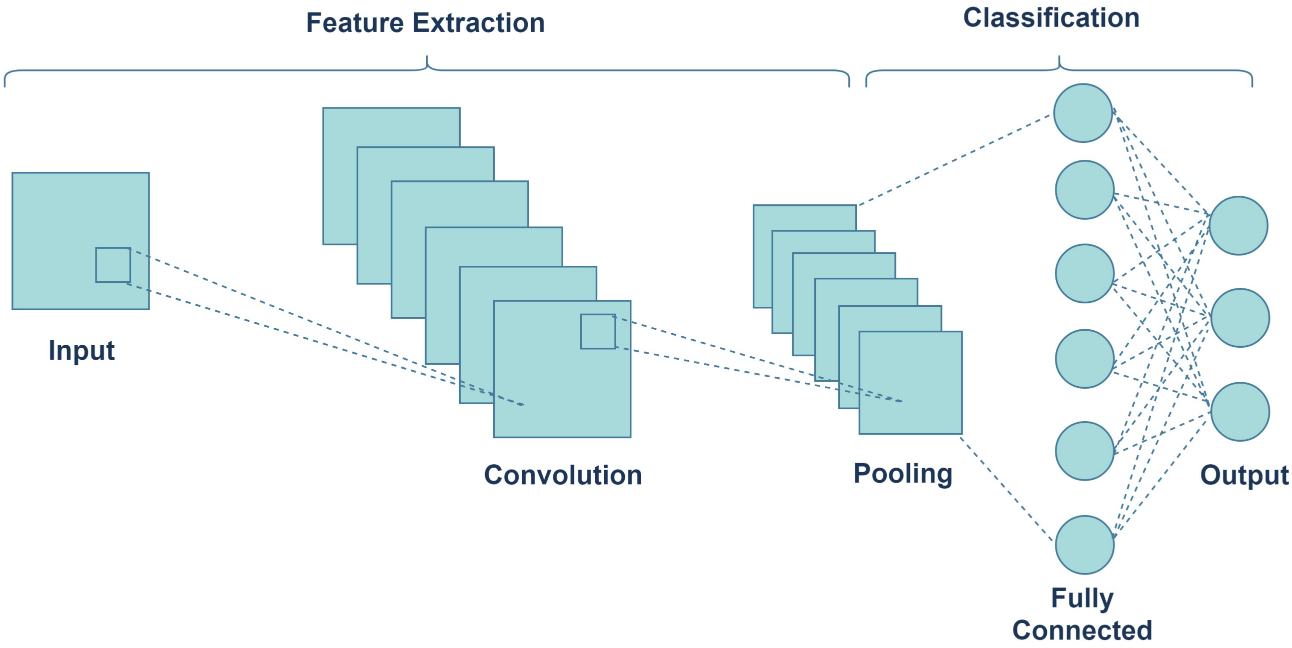

Figure 2: Basic architecture of CNN

architectures [61]. Deep convolution networks have become a technique of choice in computer vision.

3.1.1 CNN Architecture

The CNN architecture consists of two primary elements: feature extraction and classification. Feature extraction involves each layer of the CNN taking the output from the preceding layer as input and providing its output as input to the next layer. The classifier then generates the expected outputs for the input data. The figure 2 presents the basic architecture of CNN. Generally, the architecture of CNN has two fundamental layers as: convolution layer and pooling layer [58]. Each node in the convolution layer conducts a convolution operation on the input nodes to extract features. The max-pooling layer extracts features from input nodes using average or maximum operations. The output of the $n - 1^{th}$ layer serves as the input to the n^{th} layer, where the inputs pass through a set of kernels followed by the ReLU non-linear function. Advanced CNN architectures utilize a series of convolutional layers and max-pooling layers, culminating in fully connected and softmax layers. An efficient CNN architecture can largely be constructed using fundamental components such as the convolution layer, pooling layer, softmax layer, and fully connected layer.

3.2 Recurrent neural network

The RNN is a form of deep learning networks with nodes connected in a directed graph over a temporal sequence, enabling the representation of temporal dynamic behavior. A feedforward NN can also be utilized to create an RNN, and the internal state of an RNN allows for the processing of variable length input sequences. This characteristic of an RNN makes it suitable for a range of unsegmented tasks such as speech recognition [62] [63] and connected handwriting recognition [64]. The term RNN refers to two main classes of networks with similar overall structure. One type of RNN is a directed acyclic graph that can be replaced by a strictly feedforward neural network when unrolled. The second type of RNN is a directed cyclic graph that cannot be unrolled. Both types of RNNs can contain additional stored states, and these states can be directly controlled by the RNN. The storage can also be substituted by other networks or graphs if it involves time delays or feedback

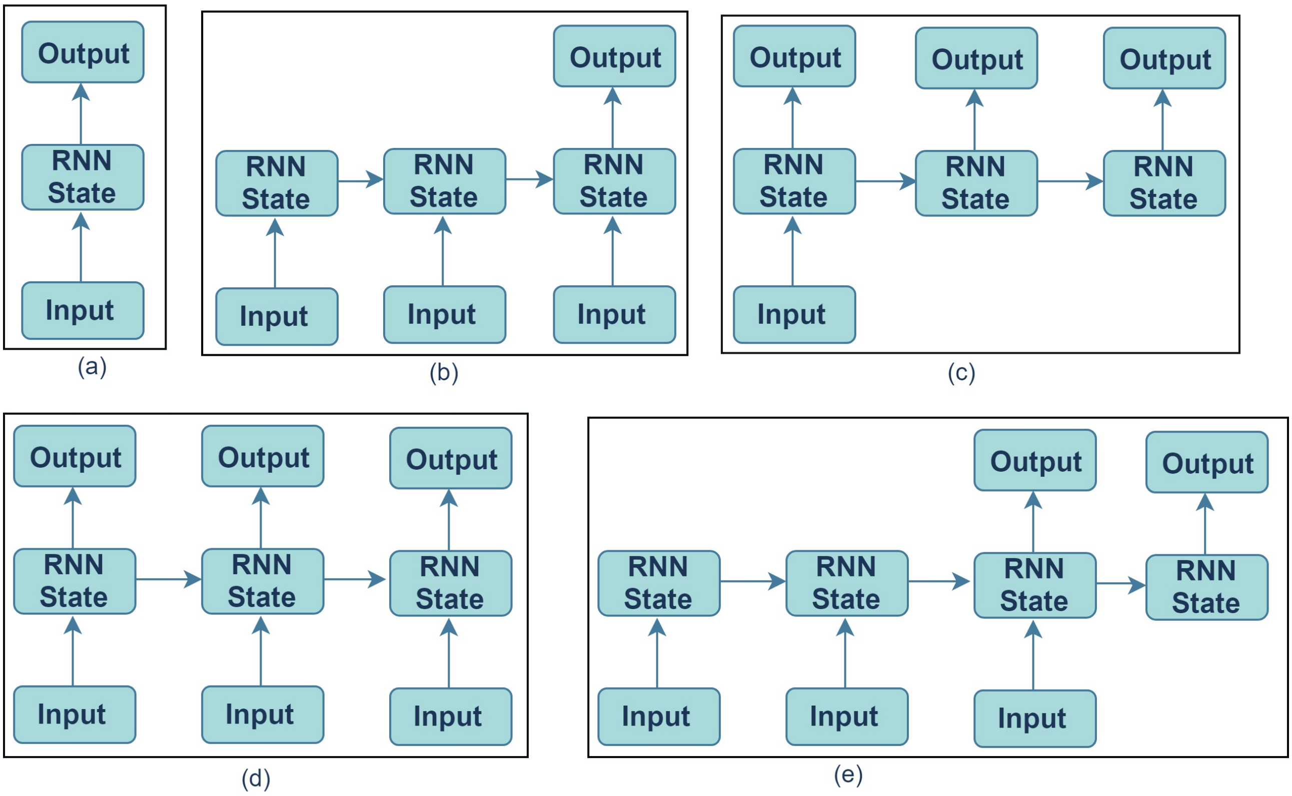

Figure 3: The different architectures of RNN (a) One to one (b) Many to one (c) One to many (d) Many to many (e) Many to many.

loops. These controlled states are known as gated states and memories, and they are also present in LSTMs and gated recurrent units.

3.2.1 RNN Architecture

RNNs are characterized by their ability to perform operations on a sequence of vectors over time. There are various RNN architectures tailored to specific applications. The figure 3 represents various architectures of RNN. These architectures can be categorized as: one to one, one to many, many to one and many to many.

One to one: This is a standard classification method that does not involve RNN and is mainly applied in image classification.
One to many: This method takes an input and generates a series of outputs. It has been successfully employed in image captioning tasks where a set of words is required as output for a single image input.
Many to one: This method takes a sequence of inputs and produces a single output. It is commonly used in scenarios where the inputs are in the form of sentences or a set of words, and the output is a positive or negative expression.
Many to many: This method generates a sequence of outputs for a sequence of inputs and is commonly used in machine translation and video classification problems.

In machine translation problems, a sequence of words in one language is given as an input to a machine and translated to a sequence of words in other language. In video classification problems, video frames are taken as an input and each frame of the video is labelled as an output.

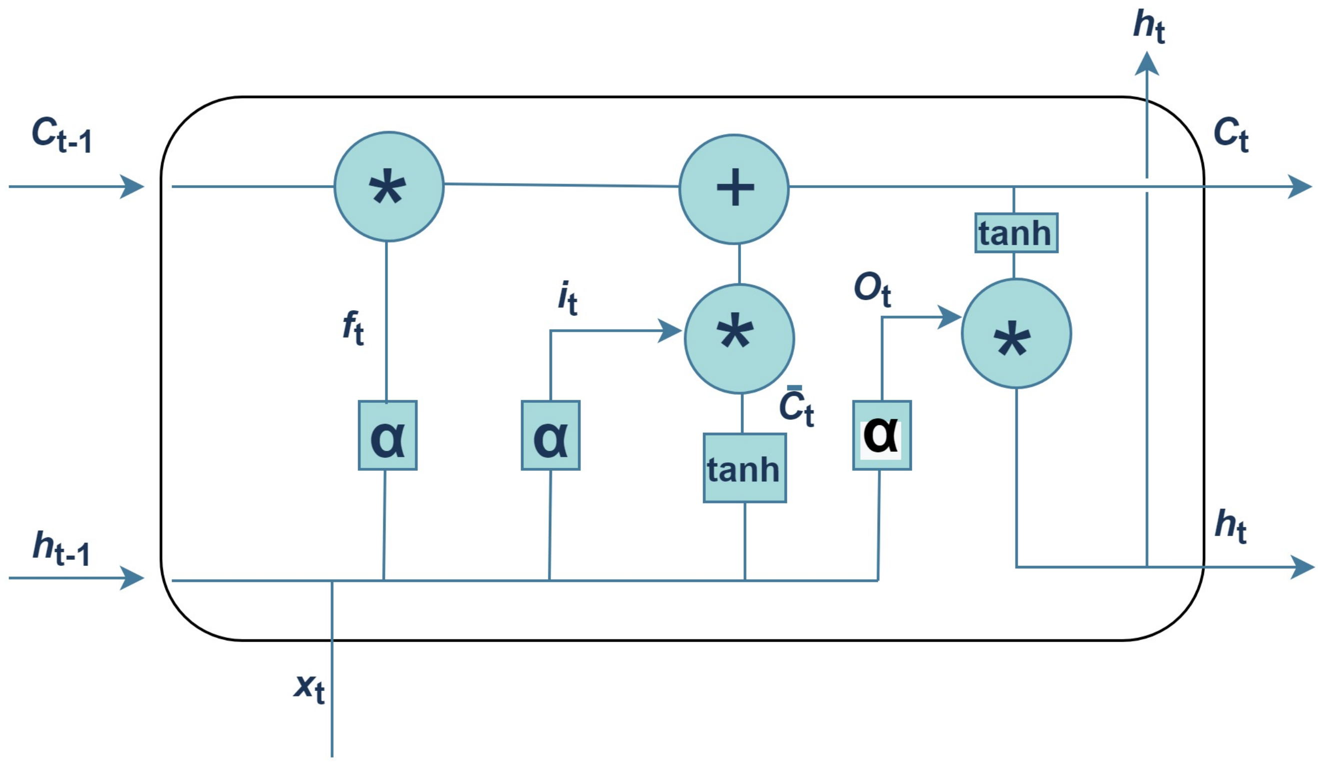

Figure 4: Diagram for LSTM

3.2.2 Long Short Term Memory

An LSTM [65] [64] is a particular form of RNN architecture that is specially created to overcome the problem of vanishing gradient [56]. The LSTM hidden layer is composed of memory blocks that are recurrently connected, with each block containing one or more recurrently connected memory cells. Three multiplicative gates (input, output, and forget gates) are utilized to activate and regulate the memory blocks, enabling the storage and retrieval of contextual information over extended periods. Specifically, the activation of a cell persists until the input gate is closed, preventing it from being overwritten by new inputs. The cell activation remains accessible to the rest of the network as long as the output gate is open, and the recurrent connection of the cell is controlled by the forget gate. Similar to the CNN architecture, multiple forward and backward layers can be present in each LSTM layer, along with multiple feature maps at the output layer, and the use of max-pooling sub-sampling to stack multiple LSTM layers.

3.2.3 LSTM Architecture

LSTM is an RNN architecture that takes into account the values over arbitrary intervals. LSTM is suitable for classification, processing and prediction of time series given time lags of unknown duration. Back propagation through time training algorithm is used to update weights in LSTM.

In last few years, there have been various advanced approaches developed for LSTM. The figure 4 shows the diagram of LSTM. The main scheme for LSTM is the cell state called gates. LSTM can add and remove information to the gates. An input gate, output gate and forget gate are defined as following:

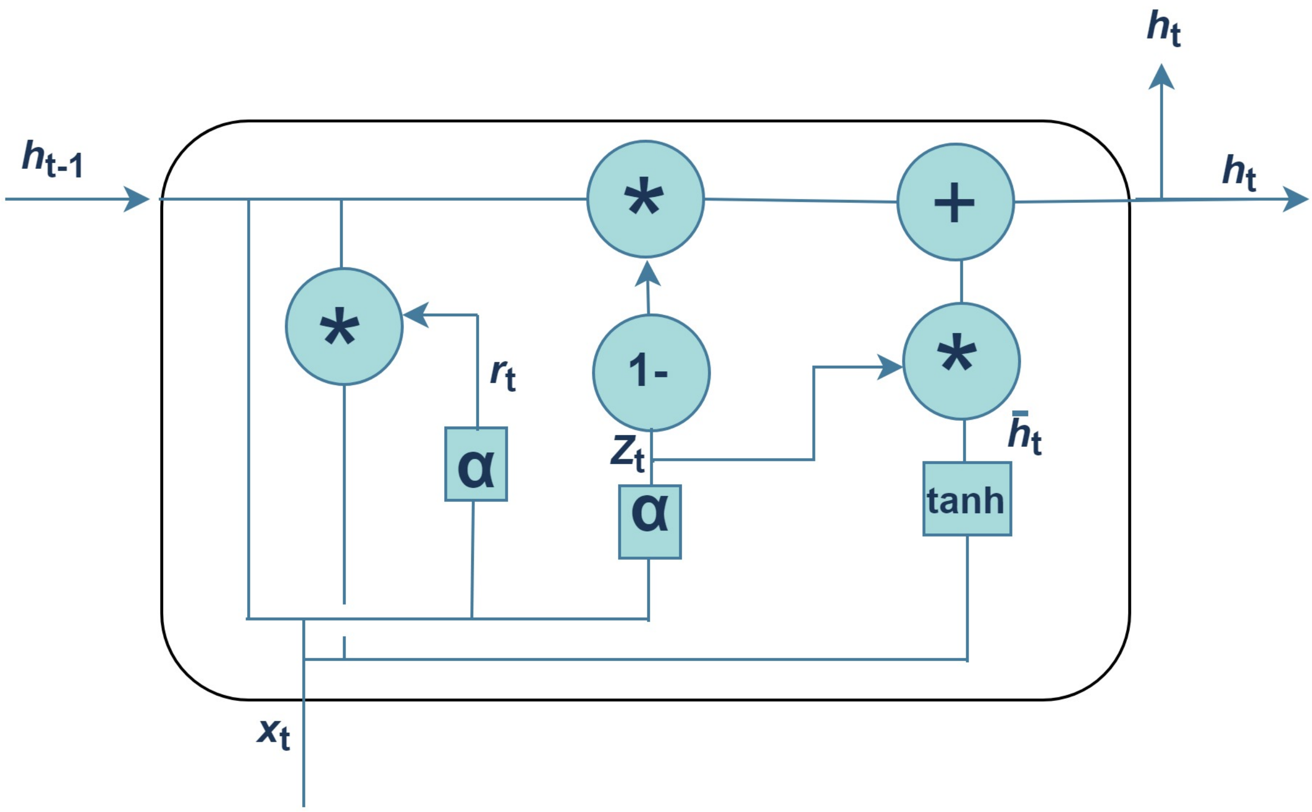

Figure 5: Gated Recurrent Unit

$$
\begin{gathered}
f_t = \alpha\,(\,W_f.\,[\,h_{t-1}, x_t\,] + b_f),\\
i_t = \alpha\,(\,W_i.\,[\,h_{t-1}, x_t\,] + b_i),\\
\tilde{C}_t = \tanh W_C.\,[\,h_{C-1}, x_t\,] + b_C),\\
C_t = f_t *\, C_{t-1} + i_t *\, \tilde{C}_t,\\
O_t = \alpha\,(\,W_O.\,[\,h_{t-1}, x_t\,] + b_O),\\
h_t = O_t *\, \tanh\,(C_t).
\end{gathered} \tag{4}
$$

LSTM models are very well accepted for processing of temporal information. There is also little modified version of network with peephole connections [66]. Gated recurrent unit (GRU) comes from more variation of LSTM [67]. GRUs are very popular among those people who work with recurrent networks. The major reason behind the recognition of GRU is its less computation cost and simple model as shown in figure 5. GRU model is also faster as it needs fewer network parameters. But LSTM provides better results when we have enough data and computational power [174]. So, in term of computation cost, topology and complexity, GRUs are lighter versions of RNN approaches than standard LSTM. The GRU model combines the input and forget gates into a single update gate and unites the cell state and hidden state with other changes. The GRU can be expressed as following:

$$
\begin{gathered}
z_t = \alpha\,(\,W_z.\,[\,h_{t-1}, x_t\,]),\\
r_t = \alpha\,(\,W_r.\,[\,h_{t-1}, x_t\,]),\\
\tilde{h}_t = \tanh\,(W.\,[\,r_t *\, h_{t-1}, x_t\,]),
\end{gathered}
$$

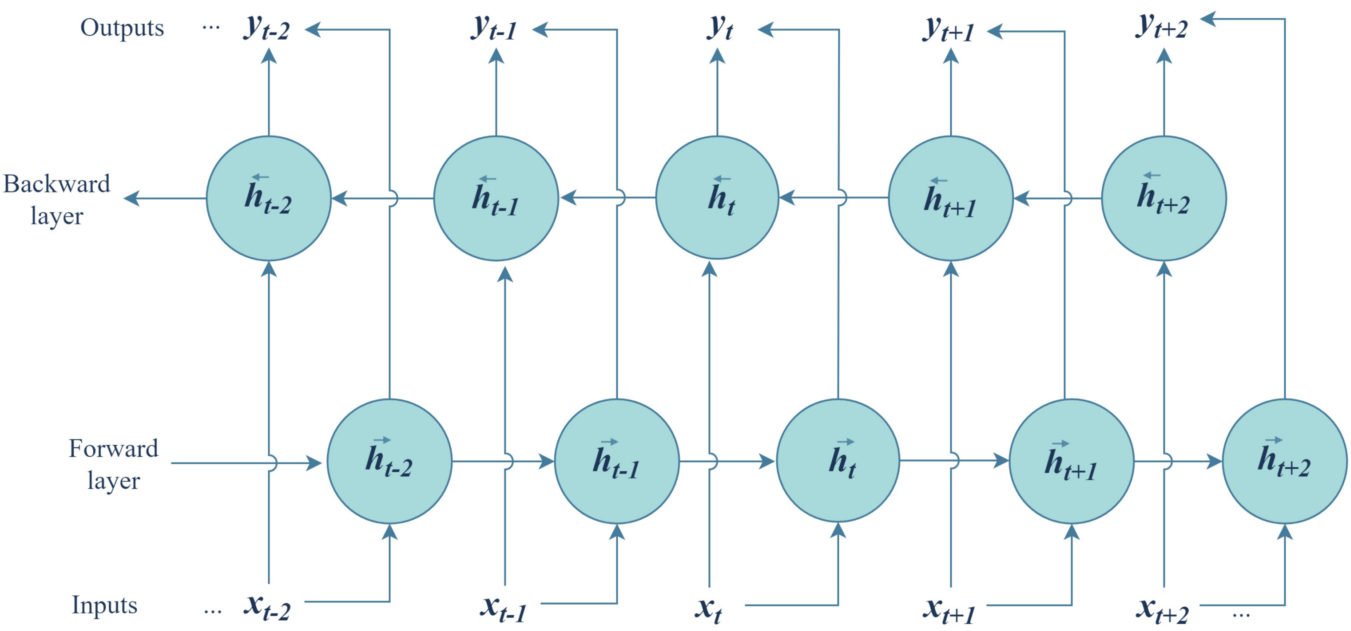

Figure 6: BLSTM architecture

$$h_t = (1 - z_t)*h_{t-1} + z_t*\tilde{h_t}. \tag{5}$$

3.2.4 Bidirectional Long Short Term Memory

In most problems of pattern recognition, it is required to access the previous and future contexts at the same time. For instance, in all problems of handwriting recognition, character recognition can be performed by recognizing the characters which appear both to the left and right of it. Further the bidirectional RNNs (BRNNs) [68] can be employed to attain the context information in left and right directions along the input sequence. In two different hidden layers of BRNNs, one layer is employed for processing of input sequence in forward direction and the subsequent in backward direction. Since both the hidden layers of BRNN are connected to the same layer of output, so the access of past and future context of every point in the sequence is given by it. BRNNs were effectively used to predict protein structure and speech processing [68], and BRNNs did better than the standard RNNs in different tasks of sequence learning. BLSTM is a combination of LSTM and BRNN.

3.2.5 BLSTM Architecture

Bidirectional LSTMs process the input sequences in both directions having two sub layers for consideration of the full input context. The figure 6 presents the architecture of BLSTM. Two sub layers of BLSTM can compute both forward ($\overrightarrow{h}$) and backward ($\overleftarrow{h}$) hidden layers. Both $\overrightarrow{h}$ and $\overleftarrow{h}$ are combined for the computation of the output sequence (y) as following:

$$\begin{aligned}
\overrightarrow{h}_t &= \mathcal{H}(W_{x\overrightarrow{h}} x_t + W_{\overrightarrow{h}\overrightarrow{h}} \overrightarrow{h}_{t-1} + b_{\overrightarrow{h}}) \\
\overleftarrow{h}_t &= \mathcal{H}(W_{x\overleftarrow{h}} x_t + W_{\overleftarrow{h}\overleftarrow{h}} \overleftarrow{h}_{t+1} + b_{\overleftarrow{h}}) \\
y_t &= (W_{\overrightarrow{h}y} \overrightarrow{h}_t + W_{\overleftarrow{h}y} \overleftarrow{h}_t + b_{\overleftarrow{h}})
\end{aligned} \tag{6}$$

4 Results of CNN and RNN

4.1 Results of CNN in Handwriting Recognition

With rapid development of computation techniques, the GPU-accelerated computing techniques have been exploited to train CNNs more efficiently. Nowadays, CNNs have already been successfully applied to handwriting recognition, face detection, behaviour recognition, speech recognition, recommender systems, image classification, and NLP. In deep learning, although CNN classification technique of neural systems is mostly used in image classification or image data, but it has attained good results for handwriting recognition too. The table 1 presents the selected handwriting recognition results using CNN.

Table 1: Results of CNN in Handwriting Recognition

Script	Reference	Methodology	Dataset	Accuracy
Roman digits	Ramjon et al. [69]	CNN based architecture	MNIST	98.7%
English characters	Zhang et al. [70]	CNN	8190 letters	95.87%
Kannada digits	Gu [71]	CNN based architecture	Kannada-MNIST (test set)	98.77%
Roman digits	Gupta and Bag [72]	CNN(2 layers)	MNIST	99.68%
Devanagari digits	Gupta and Bag [72]	CNN(2 layers)	CMATERdb 3.2.1	97.56%
Bangla digits	Gupta and Bag [72]	CNN(2 layers)	CMATERdb 3.1.1	96.35$
Telugu digits	Gupta and Bag [72]	CNN(2 layers)	CMATERdb 3.4.1	98.82%
Arabic digits	Gupta and Bag [72]	CNN(2 layers)	CMATERdb 3.3.1	96.53%
Odia digits	Gupta and Bag [72]	CNN(2 layers)	ISI-Odia	97.76%
Devanagari digits	Gupta and Bag [72]	CNN(2 layers)	ISI-Devanagari	98.31%
Bangla digits	Gupta and Bag [72]	CNN(2 layers)	ISI-Bangla	96.70%
Gujarati digits	Gupta and Bag [72]	CNN(2 layers)	Gujarati dataset	99.22%
Punjabi digits	Gupta and Bag [72]	CNN(2 layers)	Punjabi dataset	99.43%
Roman digits	Kusetogullari et al. [73]	CNN	ARDIS	98.60%
Kannada digits	Gati et al. [74]	CNN based architecture with skip connections	Kannada-MNIST(Dig-MNIST)	85.02%
Gurmukhi strokes	Singh et al. [75]	CNN, self controlled RDP based features	Gurmukhi	94.13%
Roman digits	Singh et al. [75]	CNN, self controlled RDP based features	UNIPEN	93.61%
Kannada digits	Prabhu [76]	End-to-end training using CNN based architecture	Kannada-MNIST (test set)	96.80%

Malayalam characters	Manjusha et al. [77]	CNN based on scattering transform-based wavelet filters as feature extractor and Linear SVM as classifier	Amrita_MalCharDb	91.05
Roman digits	Chowdhury et al.[78]	CNN based architecture	MNIST	99.25%
Bangla characters	Chowdhury et al.[78]	CNN based architecture	Banglalekha-isolated	91.81%
Bangla characters	Chowdhury et al.[78]	CNN based architecture	Ekush	95.07%
Bangla characters	Chowdhury et al.[78]	CNN based architecture	CMATERdb 3.1.2	93.37%
Bangla numerals	Gupta et al. [79]	Multi-objective optimisation to find the informative regions of character image + CNN features	Isolated handwritten Bangla numerals	96.54%
Bangla characters	Gupta et al. [79]	Multi-objective optimisation to find the informative regions of character image + CNN features	Isolated handwritten Bangla basic characters	85.19%
English numerals	Gupta et al. [79]	Multi-objective optimisation to find the informative regions of character image + CNN features	Isolated handwritten English numerals	97.87%
Devanagari characters	Gupta et al. [79]	Multi-objective optimisation to find the informative regions of character image + CNN features	Isolated handwritten Devanagari characterss	87.23%
Roman digits	Chakraborty et al. [80]	feature map reduction in CNN	MNIST	99.19%
Roman digits	Arora and Bhatia [81]	CNN, Keras	MNIST	95.63%, 99.20%
Malayalam characters	Manjusha et al. [82]	CNN based on scattering transform-based wavelet filters (ScatCNN)	Malayalam_DB	93.77%
Roman digits	Manjusha et al. [82]	ScatCNN	MNIST	99.31%
Bangla numerals	Manjusha et al. [82]	ScatCNN	ISI	99.22%
Chinese characters	Manjusha et al. [82]	ScatCNN	CASIA HWDB1.1	92.09%
English words	Kang et al. [83]	Attention based sequence to sequence model	IAM	82.55%
English characters	Kang et al. [83]	Attention based sequence to sequence model	IAM	93.12%
Roman digits	Sarkhel et al. [84]	A multi-column multi-scale CNN architecture + SVM	CMATERdb 3.4.1	99.50%
Roman digits	Sarkhel et al. [84]	A multi-column multi-scale CNN architecture + SVM	MNIST	99.74%
English(mostly) words	Poznanski and Wolf [85]	CNN-N-Gram	IAM	93.55%
English(mostly) characters	Poznanski and Wolf [85]	CNN-N-Gram	IAM	96.66%

French words	Poznanski and Wolf [85]	CNN-N-Gram	RIMES	93.10%
French characters	Poznanski and Wolf [85]	CNN-N-Gram	RIMES	98.10%
Hangual	Kim and Xie [86]	Deep convolutional neural netwok (DCNN)	SERI95a and PE92	95.96%, 92.92%
Roman digits	Wan et al. [87]	CNN based architecture with DropConnect layer	MNIST	99.79%
Roman digits	Krizhevsky et al. [45]	CNN, LeNet-5 system	MNIST	99.10%

One of the classical models of CNN was the LeNet-5 system. Its accuracy rate on MNIST data-set was above 99%. It was extensively used for identification of handwritten checks on banks, but it could not recognize large images. With the advancement of technology, Graphics Processing Unit (GPU) was developed, then in 2012, Krizhevsky et al. [45] employed an efficient GPU supported program for solving ImageNet problem, which also made CNN application popular. Actually, one of the problems of using CNN was that it took much time to train the network because of the many hidden nodes in the network. But the GPUs' faster parallel computing, overcame this problem too. A CNN based architecture with DropConnect layer was proposed by Wan et al. [87] in 2013, where they attained 99.79% recognition accuracy for MNIST digits. DropConnect generalizes Hinton et al.'s Dropout [88] to the complete connectivity structure of a fully connected neural network (NN) layer. They provided both empirical results and theoretical justification for showing that DropConnect helps to regularize large NN models. As deep convolutional neural netwok (DCNN) comprises many layers, so it can model much more complicated functions than shallow networks. Motivating from DCNN's great results in various machine learning and pattern recognition problems, in 2015, Kim and Xie [86] developed a new recognizer based on deep CNN to improve the Hangual handwrititng recognition performance. They built their own Hangul recognizers based on DCNN and developed various novel techniques for performance and networks training speed improvement. They evaluated their proposed recognizers on image datasets of Hangul, named SERI95a and PE92, where recognition results as 95.96% on SERI95a and 92.92% on PE92 are attained. In 2016, Poznanski and Wolf [85] proposed a CNN-N-Gram based system for handwriting recognition, and recognized handwritten English and French words with 93.55% and 93.10% accuracy, respectively. Different persons' variation in writing styles and single person's variation in handwriting from time to time, make recognition of the local invariant patterns of a handwritten digit and character difficult. For this purpose, in 2017, Sarkhel et al. [84] proposed a non-explicit feature based approach, specifically it was a multi-column multi-scale CNN based architecture. Their proposed approach has been validated on different datasets of isolated handwritten digits and characters of Indic scripts, and best results are attained on MNIST dataset that is 99.74% without any data augmentation to the original dataset. Inspired from the deep learning's role in image classification, in 2018, Arora and Bhatia [81] used Keras for classification of handwritten images of MNIST dataset. In fact, they used feed forward NN and CNN to extract features and training the model, it used Stochastic Gradient Descent for optimization. In their work, for classification of handwritten digits, it is observed that CNN attained greater accuracy in comparison to feed forward, and CNN obtained 95.63% and 99.20% accuracy for 5 and 20 iterations, respectively. Malayalam handwritten character recognition is very challenging, due to the isomorphic nature of character classes and a large number of character classes. To recognize handwritten Malayalam characters, in 2018, Manjusha et al. [82] replaced the convolutional feature maps of first layer in CNN architecture with scattering transform-based feature maps, and attained 93.77% as recognition accuracy. Scattering transform can compute stable invariant description of input patterns where it applies a series of wavelet decomposition, modulus and averaging operations. Their proposed hybrid CNN [82] also achieved above 99% recognition accuracy for MNIST digits and ISI Bangla numerals datasets. A convolve, attend and spell, an attention based sequence to sequence model to recognize handwritten words without the use of HTR system's traditional components, as connectionist temporal classification, language model nor lexicon was presented by Kang et al. [83] in 2018. It was an end-to-end system that contained an encoder, decoder and attention mechanism, and it outperformed most of the existing best results, and it attained 93.12% character recognition accuracy and 82.55% word recognition accuracy for IAM dataset on word-level. In 2019, Chowdhury et al. [78] used CNN to develop a handwritten character recognition model, and attained 99.25% accuracy for MNIST digits and 91.81% accuracy for Banglalekha-isolated characters. In 2019, Gupta et al. [79] proposed an opposition based multi-objective optimisation search algorithm to find the informative regions of character images, where they also used CNN features to evaluate the proposed work on different Indic scripts' isolated units of handwriting and obtained good results for isolated Bangla basic characters, Bangla numerals, English numerals, and isolated Devanagari characters. Considering the research efforts for Malayalam character handwriting recognition, Manjusha et al. [77] developed a handwritten character image database of Malayalam language script in 2019. In their work, recognition experiments were conducted by using different techniques of feature extraction. Among the used feature descriptors, scattering CNN attained the best recognition accuracy of 91.05%. In 2019, Prabhu [76] created a new dataset for handwritten digits of Kannada language, which is called Kannada-MNIST dataset, and attained best results as 96.80% using CNN based architecture. In 2019, Gati et al. [74] described how great results and performance can be attained on a very challenging Dig-MNIST dataset using a custom-built model based on the skip CNN architecture, where 85.02% recognition accuracy was attained using proposed model trained on Kannada-MNIST and tested on the Dig-MNIST dataset without any pre-processing. In 2019, Chakraborty et al. [80] proposed for reduction of the feature maps which are used in training the CNN for reduction of computation time and storage space. Experimental results proved that the time requirement for training the CNN decreased with reduction in number of feature maps without affecting the accuracy much, and above 99% accuracy rate was attained for MNIST digit dataset. In 2020, Kusetogullari et al. [73] introduced different datasets of digits in ARDIS, and attained best recognition accuracy for digits using CNN that is 98.60%. A novel self-controlled RDP point based smaller size feature vector approach to recognize online handwriting was proposed by Singh et al. [75] in 2020, where they employed a CNN based network that trains in a few minutes on a single machine without GPUs due to the use of Conv1Ds, and it attained 94.13% and 93.61% recognition rates for Gurmukhi and UNIPEN datasets, respectively. Recently, in 2021, Gu [71] proposed a CNN based model to classify the Kannada-MNIST dataset and made analysis of the proposed model performance on training, testing and validation sets. The CNN model was trained on more than 51000 images and it was validated over 9000 images for 30 epochs, where the CNN model attained a testing accuracy of 98.77%, and it outperformed other methods as SVM, logistic regression and a CNN baseline. This study is the evidence for the capability of proposed CNN model, and it also demonstrates the benefit of using a CNN architecture over other classification methods when performing handwritten character recognition jobs. A script independent CNN based system to recognize numerals was developed by Gupta and Bag [72] in 2021, it is a system to recognize handwritten digits written in multi languages and it is independent of fusion where it has just 10 classes corresponding to every numeric digit. This was the first study that addressed the problem of multilingual numerals recognition, where experimental results attained the accuracy of 96.23% for eight Indic scripts collectively. The attained results are promising and demonstrates the hypothesis that multilingual handwritten numeral recognition is void with CNN. In 2023, Ramjon et al. [69] explored various learning models and developed a system for handwriting recognition. They utilized the MNIST dataset, examined different models, and found that a CNN operated on well-optimized hardware with a GPU and ample training data can attain an accuracy of up to 98.7% for number recognition. The model's accuracy and speed can be further improved by increasing the dataset size, enhancing the number of training epochs, and utilizing parallel hardware. In 2023, Zhang et al. [70] introduced a character recognition system for finger writing that employed a series of time of flight distance sensors installed on a low-power microcontroller, the STM32F401, integrated with deep learning algorithms. In their work, a system is built to recognize all 26 lowercase letters of the English alphabet in real-time, eliminating the necessity for users to wear extra devices. A collection of 8,190 samples of lowercase letters written by finger was used, and attained 95.87% accuracy for lowercase letters using CNN.

4.2 Results of RNN in Handwriting Recognition

RNNs are very powerful machine learning models and have found use in a wide range of areas where sequential data is dealt. RNNs have been widely used in prediction problems, machine translation, face detection, speech Recognition, OCR based image recognition and handwriting recognition etc. RNNs have received great success when working with sequential data, generally in the field of handwriting recognition. The table 2 presents the selected handwriting recognition results using RNN.

Table 2: Results of RNN in Handwriting Recognition

Script	Reference	Methodology	Dataset	Accuracy
English words	Mahadevkar et al.[89]	CNN+BLSTM	IAM	98.55%
English words	Mahadevkar et al.[89]	CNN+BLSTM	RIMES	98.80%
English characters	Zhang et al. [70]	LSTM	8190 letters	98.31%
English words	Pham et al. [90]	LSTM with dropout at the topmost hidden layer	IAM	60.52%
English words	Pham et al. [90]	LSTM with dropout at multiple layers	IAM	68.56%
French words	Pham et al. [90]	LSTM with dropout at the topmost hidden layer	Rimes	63.97%
French words	Pham et al. [90]	LSTM with dropout at multiple layers	Rimes	72.99%
English characters	Pham et al. [90]	LSTM with dropout at the topmost hidden layer	IAM	81.55%
English characters	Pham et al. [90]	LSTM with dropout at multiple layers	IAM	86.08%
French characters	Pham et al. [90]	LSTM with dropout at the topmost hidden layer	Rimes	87.83%
French characters	Pham et al. [90]	LSTM with dropout at multiple layers	Rimes	91.38%
English words	Doetsch et al. [91]	LSTM-RNN	IAM	87.80%
French words	Doetsch et al. [91]	LSTM-RNN	IAM	87.10%
Bangla characters	Chollet et al. [92]	LSTM	Banglalekha-isolated	87.41%
Bangla characters	Chollet et al. [92]	LSTM	Ekush	93.06%
Arabic words	Chherawala et al. [93]	Weighted Vote Combination of RNN	FN/ENIT	96%
French words	Chherawala et al. [93]	Weighted Vote Combination of RNN	RIMES	95.2%
English words	Shkarupa et al. [94]	CTC+BLSTM	handwritten medieval Latin text	78.10%
English words	Shkarupa et al. [94]	Sequence to sequence+LSTM	handwritten medieval Latin text	72.79%

English Words	Wigington et al. [95]	RNN+CTC	IAM	80.93%
French Words	Wigington et al. [95]	RNN+CTC	Rimes	88.71%
English characters	Wigington et al. [95]	RNN+CTC	IAM	93.93%
French characters	Wigington et al. [95]	RNN+CTC	Rimes	96.91%
English words	Dutta et al. [96]	Hybrid CNN-RNN network	IAM	87.39%
English characters	Dutta et al. [96]	Hybrid CNN-RNN network	IAM	95.12%
French words	Dutta et al. [96]	Hybrid CNN-RNN network	RIMES	92.96%
French characters	Dutta et al. [96]	Hybrid CNN-RNN network	RIMES	97.68%
English words	Dutta et al. [96]	Hybrid CNN-RNN network	GW	87.02%
English characters	Dutta et al. [96]	Hybrid CNN-RNN network	GW	95.71%
English words	Krishnan et al. [97]	Convolutional recurrent neural network (CRNN)	IAM	94.90%
English words	Sueiras et al. [98]	Sequence to sequence NN	IAM	87.30%
French words	Sueiras et al. [98]	Sequence to sequence NN	IAM	93.40%
English characters	Krishnan et al. [97]	CRNN	IAM	97.44%
Bengali words	Ghosh et al. [99]	BLSTM	Bengali dataset of 120000 words	95.24% (lexicon 1K)
Bengali words	Ghosh et al. [99]	BLSTM	Bengali dataset of 120000 words	90.78% (lexicon 5K)
Bengali words	Ghosh et al. [99]	BLSTM	Bengali dataset of 120000 words	87.38% (lexicon 10K)
Devanagari words	Ghosh et al. [99]	BLSTM	Bengali dataset of 120000 words	99.50% (lexicon 1K)
Devanagari words	Ghosh et al. [99]	BLSTM	Bengali dataset of 120000 words	96.27% (lexicon 5K)
Devanagari words	Ghosh et al. [99]	BLSTM	Bengali dataset of 120000 words	94.34% (lexicon 10K)
English words	Geetha et al. [100]	CNN-RNN	IAM	95.20%
English characters	Geetha et al. [100]	CNN-RNN	IAM	97.48%
French words	Geetha et al. [100]	CNN-RNN	RIMES	98.14%
French characters	Geetha et al. [100]	CNN-RNN	RIMES	99.35%

In 2014, Pham et al. [90] presented that the dropout can improve the performance of RNN greatly. The word recognition networks having dropout at the topmost layer improved the character and word recognition by 10% to 20%, and when dropout used with multiple LSTM layers, then it further improved the performance by 30% to 40%. They reported the best results on Rimes dataset as 91.38% and 72.99% for character and word recognition, respectively. The simple RNN was modified by Koutnik et al. [101] in 2014, and introduced a powerful Clockwork RNN (CW-RNN), where hidden layers were divided into different modules and every layer processed the inputs at its own temporal granularity, which made computations over prescribed clock rate only. The CW-RNN reduced the number of parameters of simple RNN, and also improved the speed and performance significantly. For online handwriting recognition, CW-RNN outperformed the simple RNN and LSTM, and improved recognition accuracy by 20% for English sentences. A modified topology for LSTM-RNN that controlled the shape of squashing functions in gating units was demonstrated by Doetsch et al. [91] in 2014. An efficient framework of mini batch training at sequence level in combination with sequence chunking approach was also proposed by them. They evaluated their framework on IAM and RIMES datasets by using GPU based implementation, and it was three times faster in training RNN models which outperformed the state-of-the-art recognition results, where 87.80% and 87.10% recognition accuracies were attained for handwritten words of IAM and RIMES datasets, respectively. In 2015, an image classification system using LSTM with Keras was built and it was applied to handwritten Bangla character datasets, where it achieved 87.41% and 93.06% recognition accuracy for two different datasets of Bangla characters [92]. This architecture consisted of one LSTM layer having 128 units, had activation function as 'ReLU' and recurrent activation function had been set to 'hard sigmoid'. In 2016, Chherawala et al. [93] proposed a novel method to extract the promising features of handwritten word images. They proposed a framework to evaluate feature set based on collaborative setting. In their work, they employed weighted vote combination of RNN classifiers, where particular feature set was used to train every RNN. The major contribution of their study was the quantification of the feature sets' importance through weight combination, and it also showed their complementarity and strength. They used RNN because of the state-of-the-art results, and provided the first feature set benchmark for RNN classifier. They evaluated different feature sets on different datasets of Arabic and Latin scripts, and attained best accuracies as 96% and 95.2% for IFN/ENIT and RIMES datasets, respectively. For historic handwritten Latin text recognition, two important approaches based on RNN were proposed by Shkarupa et al. [94] in 2016. Their first approach used connectionist temporal classification (CTC) output layer, and attained 78.10% word level accuracy. The other approach used sequence-to-sequence learning, and attained 72.79% word level accuracy. In their work, when CTC approach was used with BLSTM, it outperformed the sequence-to-sequence based approach used with LSTM. Their proposed system of handwriting recognition considered unsegmented word images as input and provided decoded strings as output. In 2017, Wigington et al. [95] presented two data normalization and augmentation techniques, and these were used with CNN and LSTM. These techniques reduced the character error rate and word error rate significantly, and significant results were reported for handwriting recognition tasks. The novel normalization technique was applied to both word and line images. Their proposed approaches attained high accuracies for both characters and words over several existing studies, where IAM dataset character and word level recognition accuracy was reported as 96.97% and 94.39%, respectively. In 2018, Dutta et al. [96] proposed a modified CNN-RNN based hybrid architecture and mainly focussed for effective training with: (a) network's efficient initialization with the use of synthetic data in pretraining, (b) slant correction with image normalization and (iii) domain specific transformation of data and distortion to learn important invariances. In their work, a detailed ablation study for analysis of the contribution of individual module was performed and the results for unconstrained line and word recognition on IAM, RIMES and GW datasets were presented at par literature, where they attained lexicon free word recognition accuracies as 87.39%, 92.96% and 87.02% on these three datasets, respectively. To represent handwritten word images efficiently, an HWNet v2 architecture was presented by Krishnan et al. [97] in 2018. The state-of-the-art attribute embedding was enabled by this work. An end-to-end embedding framework was demonstrated by it, and it used textual representation and synthetic image for learning complementary information to embed text and images. It also improved the word recognition performance using a convolutional recurrent neural network (CRNN) architecture, by using the synthetic data and spatial transformer layer, and attained character and word level accuracies on IAM dataset as 97.44% and 94.90%, respectively. In 2018, a system based on sequence to sequence architecture with convolutional network was proposed by Sueiras et al. [98] to recognize offline handwriting. This model had three major components, where first convolutional network extracted relevant features of the characters present in the word. Then RNN captured the sequential relationships of extracted features. Thirdly, the input word was predicted by decoding the sequence of characters with another RNN. Their proposed system was tested on handwritten words of IAM and RIMES datasets, and attained the recognition accuracy as 87.3% and 93.6%, respectively, where no language model was used and results were attained with closed dictionary. In 2019, Ghosh et al. [99] presented a new online handwritten word recognition system based on LSTM and BLSTM versions of RNN, and recognized Devanagari and Bengali words in lexicon dependent environment with above 90% (for lexicon size 5K) recognition accuracy. Their proposed approach divided every handwritten word into upper, middle, and lower zones horizontally, and reduces the basic stroke order variations with in a word. Further, they also used various structural and directional features of different zones' basic strokes of handwritten words. In 2021, Geetha et al. [100] proposed a hybrid model to recognize handwritten text by utilizing deep learning that used sequence-to-sequence approach. It used various features of CNN and RNN-LSTM. It used CNN to extract features of handwritten text images. The extracted features were then modelled with a sequence-to-sequence approach and fed in RNN-LSTM to encode the visual features and decoded the sequence of letters present in handwritten image. Their proposed model was tested with IAM and RIMES datasets, where above 95% accuracy was attained using CNN-RNN for handwritten words of English and French. In 2023, Zhang et al. [70] recognized lowercase letters of English using LSTM, and attained 98.31% accuracy. Recently, in 2024, Mahadevkar et al. [89] proposed a method to tackle the difficulties associated with recognizing handwritten text by utilizing a hybrid approach. The main goal of their study is to improve the precision of identifying handwritten text from pictures. By combining CNN with BLSTM and incorporating a connectionist temporal classification decoder, their findings reveal significant enhancements. Their hybrid model demonstrated remarkable accuracy rates of 98.50% and 98.80% on the IAM and RIMES datasets, respectively. As offline handwriting recognition presents considerable challenges due to the variations in handwriting styles, deterioration of backgrounds, and the unpredictability of word combinations. A study to addresses the challenge of recognizing lines of handwritten text by employing octave convolutional recurrent neural networks (OctCRNN) was presented by Castro et al. [102] in 2024. They explored the OctCRNN in various configurations, including an octave structure that effectively balances computational efficiency with recognition accuracy. They evaluated the effectiveness of their solution by measuring character and word error rates in comparison to recognized benchmarks for handwritten text recognition: IAM, RIMES, and ICFHR 2016 READ.

5 General Observations

In this section, an analysis of various techniques used for deep learning architectures, and deep learning use in handwriting recognition and other related fields are presented.

- Deep learning has been effectively used in various emerging fields to solve complex problems of real world with different deep learning architectures. The utilization of various activation functions in deep learning architectures is crucial for achieving state-of-the-art performance. These functions are used to carry out diverse computations between the hidden and output layers of deep learning architectures. Further, the advancement in deep learning architectures' configuration brings new challenges, particularly for the selection of right activation functions to perform in various domains from the classification of objects [45] [103] [104] [105], speech recognition [106] [107], segmentation [108] [109], machine translation [110] [111], scene description [112] [113], weather forecasting [114] [115], cancer detection [116] [117] [118], self-driving cars [119] [120] and other adaptive systems. With such challenges, the comparison of present trends in the application of activation functions employed in deep learning, portrays a gap of literature in this direction.

- Most tasks commonly associated with machine learning can be accomplished using deep learning techniques. These techniques are applicable in supervised learning for predicting outcomes or labels for each data point, as well as in unsupervised learning for summarizing, explaining, and identifying patterns through clustering in a dataset.

- The current state of the art in healthcare is exceeded by deep learning techniques in numerous studies, including patient and disease categorization, basic biological research, genomics, and treatment development.

- Deep learning has been the dominant force in many fields, surpassing other machine learning approaches and achieving significant improvements in predictive performance. However, there are still many problems that deep learning has not been able to solve. Deep learning has not completely transformed the study of human disease. It has yet to realize its trans-formative strength and to encourage a strategic inflection point.

- Great strides have been made in speech recognition through the use of deep learning, resulting in a significant drop in error rates from over 20% to less than 6%. and exceeded human performance in the past years [121] [122].

- In medical imaging, diabetic macular oedema [123], diabetic retinopathy [123], skin lesion [124] and tuberculosis [125], deep learning based classifiers are greatly successful and can be compared to clinical performance. We have exceeded the high standard in certain areas more than others, particularly in those that resemble non-biomedical tasks currently dominated by deep learning. Deep learning can guide specialists to the most difficult cases needing manual assessment, despite the need to tackle the risk of false negatives.

- The prioritization of experiments and support for discovery are also emphasized by deep learning techniques. For example, when conducting chemical screening to discover new drugs, a deep learning system has the capability to effectively pinpoint hundreds of active, target-specific small molecules from a vast search space. Despite having only moderate overall accuracy, this would still hold significant practical value.

- The deep neural networks can be built resistant to the adversarial attacks. Further, there is also possibility to design reliable adversarial training methods. Thus, the findings from existing deep learning studies provide motivation for having adversarial robust deep learning models within current reach.

- A significant milestone has been reached in deep learning, as it has achieved human-level performance in various areas of biomedical science. But deep neural networks as other machine learning algorithms are also prone to errors that are also made by humans most likely, such as miss-classification of adversarial examples [126] [127], and it can be considered that the semantics of the objects presented cannot be completely understood by these algorithms. The majority of these challenges can be addressed by the collaboration between deep learning algorithms and human experts, leading to improved performance compared to either working alone [117].

- We have a strong belief in the future of deep learning in the field of machine learning. It is definite that deep learning will bring about a revolution in these fields, but considering the rapid evolution of these areas, we have hope that its complete potential has not been fully uncovered yet. There are numerous challenges beyond enhancing training and predictive accuracies. Current research has started to tackle most of these issues and has demonstrated that they are not impossible to overcome.

- Deep learning provides a flexible way to model data in its natural form, as an illustration, molecular graphs instead of pre-computed bit vectors for drug discovery and longer DNA sequences instead of k-mers for TF binding forecasting. This kind of flexible input feature interpretations have incited creative modelling approaches that are not feasible with rest of machine learning techniques. In forthcoming years, deep learning algorithms will be able to condense extensive sets of input data into understandable models, prompting scientists to explore inquiries they previously did not know how to formulate.

- Even though deep learning still has more to achieve, previous research indicates that it has the potential to deliver quicker and more dependable results. This potential could lead to a transition away from current decision support techniques like support vector machines and k-nearest neighbor towards deep learning.

- The training algorithms for deep learning have a significant computational complexity, leading to a high runtime complexity, which in turn causes extended training times. After choosing the architecture, there is always a need to adjust the tuning parameters. The model is influenced by both the structure selection and parameter adjustment. So, there is need to have many test runs.The training phase of deep learning models is being actively researched to reduce its duration. Speeding up the training process in a parallel distributed processing system is a challenge in the field of deep learning [128]. As the network for individual processors becomes the bottle neck [129] then GPUs are used to reduce the network latency [130].

- Deep learning faces certain challenges as: using deep learning for big data analysis, dealing causality in learning, scalability of approaches in deep learning, data generating ability when data does not exist for learning the system, need of energy efficient techniques for special purpose devices, learning from different domains or models together.

- Present deep learning models works splendidly in various applications, but the solid theory of deep learning still lacks. It is not mostly known that why and how it works essentially. It is required to make more endeavours to investigate the basic principles of deep learning. In the meantime, it is very worth on exploring how to leverage natural visual perception mechanism for further improvement in the design of deep learning models.

- Although, the field of handwriting recognition has seen remarkable progress due to advancements in deep learning, particularly with the development of CNNs and RNNs. Despite this, there are still various constraints and obstacles persist due to variability in handwriting styles, data scarcity and labeling, sensitivity to image quality and noise, generalization to different writing surfaces, high computational requirements, contextual understanding and complex character segmentation.

- In current studies on handwriting recognition with deep learning, various typical limitations and challenges are identified. Numerous research efforts utilize commonly accessible datasets like IAM, MNIST, or ICDAR, which offer restricted diversity in handwriting style, demographic representation, or writing contexts. The majority of studies concentrate on English and a limited number of other languages, resulting in an insufficient availability of resources and models for languages that feature intricate scripts or symbols, like Arabic, Chinese, or Indic scripts. In existing studies, word level models offer greater accuracy but often depend on particular lexicons or dictionaries, which restrict their capability to manage out of vocabulary words, proper names, or specialized terminology. Further there are certain unsolved problems exist for handwriting recognition using deep learning as well: inadequate handling of cursive and connected handwriting, dependence on preprocessing and image quality, insufficient contextual understanding and error correction and high computational and memory requirements. Thus Future studies may gain from a more comprehensive methodology, combining strong contextual frameworks, enhanced datasets, and flexible architectures that more accurately reflect real-world scenarios.

6 Conclusion

The recent decade observed an increasingly rapid progress in technology, mainly backed up by the advancements in the area of deep learning and artificial intelligence. The present paper presented various architectures of deep learning and surveyed the current state-of-the-art on deep learning technologies used in handwriting recognition domain. After reviewing so many papers, the present study is able to distil the perfect deep learning methods and architectures for different handwriting recognition tasks and general observations on other related application areas too. The CNN and its derivatives are the out performers in most image analysis areas, and RNN and its derivatives are out performers in dealing with sequence data. Further, an outstanding conclusion can be drawn that the exact architecture of deep learning is an important determinant for finding a good solution in many problems. The present survey not only given a snapshot of the existing deep learning research status in handwriting recognition but also made an effort for identification of the future roadway for intended researchers. The findings indicate that there are remarkable opportunities in the deep learning research and it shows that they will not disappear anytime soon. So, the present study encourages future researchers that are interested in the area to start exploring as it currently seems to be wide open for new studies.

7 Declaration

Conflict of Interest: The authors declare that they have no conflict of interest. **Data Sharing:** Data sharing not applicable to this article as no datasets were generated in this study.

8 Acknowledgment

We would like to dedicate this paper in memory of co-author Dr Sudhir Rohilla, whose expertise in this paper was instrumental in success of this paper.

9 Publisher:

Computer Vision Center
Edifici O - Campus UAB
08193 Bellaterra (Barcelona) - Spain
e-mail: elcvia@cvc.uab.es
Tel: 34 - 93 - 581 18 28
Fax: 34 - 93 - 581 16 70

References

[1] Alan K. Mackworth David L. Poole. *Artificial Intelligence: Foundations of Computational Agents*. Cambridge University Press, 2010. https://doi.org/10.1017/9781009258227.

[2] Shivashankar B. Nair Elaine Rich, Kevin Knight. *Artificial Intelligence*. Tata McGraw Hill, 2010.

[3] Olaf Zawacki-Richter, Victoria I. Marín, Melissa Bond, and Franziska Gouverneur. Systematic review of research on artificial intelligence applications in higher education – where are the educators? *International Journal of Educational Technology in Higher Education*, 16(1):39, Oct 2019. https://doi.org/10.1186/s41239-019-0171-0.

[4] L. Chen, P. Chen, and Z. Lin. Artificial intelligence in education: A review. *IEEE Access*, 8:75264–75278, 2020. https://doi.org/10.1109/ACCESS.2020.2988510.

[5] Nils J. Nilsson. *The Quest for Artificial Intelligence: A history of ideas and achievements*. Cambridge University Press, 2010. https://doi.org/10.1017/CBO9780511819346.

[6] Michael A. Goodrich and Alan C. Schultz. Human-robot interaction: A survey. 1(3):203–275, January 2007. https://doi.org/10.1561/1100000005.

[7] A. L. Buczak and E. Guven. A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications Surveys Tutorials*, 18(2):1153–1176, 2016. https://doi.org/10.1109/COMST.2015.2494502.

[8] Arash Bahrammirzaee. A comparative survey of artificial intelligence applications in finance: Artificial neural networks, expert system and hybrid intelligent systems. *International Journal of Neural Computing and Application*, Available online 20 June 2010, 11 2010. https://doi.org/10.1007/s00521-010-0362-z.

[9] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013. https://doi.org/10.1109/TPAMI.2013.50.

[10] David Brougham and Jarrod Haar. Smart technology, artificial intelligence, robotics, and algorithms (stara): Employees' perceptions of our future workplace. *Journal of Management & Organization*, 24(2):239–257, 2018. https://doi.org/10.1017/jmo.2016.55.

[11] Juan Gustavo Corvalan. Artificial intelligence: Challenges and opportunities-prometea: The first artificial intelligence of latin america at the service of the justice system. *Revista de Investigações Constitucionais*, 5:295 – 316, 04 2018. https://doi.org/10.5380/rinc.v5i1.55334.

[12] Zoubin Ghahramani. Probabilistic machine learning and artificial intelligence. *Nature*, 521(7553):452–459, May 2015. https://doi.org/10.1038/nature14541.

[13] D. Castelvecchi. Can we open the black box of AI?. *Nature*, 538:20–23, 2016. https://DOI:10.1038/538020a.

[14] Carlton McDonald. Machine learning: a survey of current techniques. *Artificial Intelligence Review*, 3(4):243–280, Dec 1989. https://doi.org/10.1007/BF00141197.

[15] F. Musumeci, C. Rottondi, A. Nag, I. Macaluso, D. Zibar, M. Ruffini, and M. Tornatore. An overview on application of machine learning techniques in optical networks. *IEEE Communications Surveys Tutorials*, 21(2):1383–1408, 2019. https://doi.org/10.1109/COMST.2018.2880039.

[16] C.M. Bishop. *Pattern recognition and machine learning*. Springer, 2006. https://link.springer.com/book/9780387310732.

[17] O. Chapelle, B. Schölkopf, and A. Zien. *Semi-supervised learning (1st ed.)*. The MIT Press, 2006. https://doi.org/10.7551/mitpress/9780262033589.001.0001.

[18] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537, 2011. https://dl.acm.org/doi/10.5555/1953048.2078186.

[19] J. Du, C.X. Ling, and Z.H. Zhou. When does cotraining work in real data? *IEEE Transactions on Knowledge and Data Engineering*. 10.1109/TKDE.2010.158.

[20] Y. Freund and R.E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997. https://doi.org/10.1006/jcss.1997.1504.

[21] N. Grira, M. Crucianu, and N. Boujemaa. Unsupervised and semisupervised clustering: A brief survey. *In 7th ACM SIGMM international workshop on multimedia information retrieval*, 2004.

[22] I. Guyon and A. Elisseeff. An introduction to feature extraction. *Feature extraction*, page 1–25, 2006. https://doi.org/10.1007/978-3-540-35488-8_1.

[23] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521:436–444, 2015. https://doi.org/10.1038/nature14539.

[24] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[25] V. Vapnik. *Statistical learning theory (Vol. 1)*. New York: Wiley, 1998.

[26] O. Abdel-Hamid, A. R. Mohamed, L. Deng H. Jiang, G. Penn, and D. Yu. Convolutional neural networks for speech recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(10):1533–1545, 2014. https://doi.org/10.1109/TASLP.2014.2339736.

[27] K. Cho, B. Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *In The Conference on Empirical Methods in Natural Language Processing*, pages 1724–1734, 2014. https://doi.org/10.3115/v1/D14-1179.

[28] Y. Zhao, X. Jin, and X. Hu. Recurrent convolutional neural networks for speech processing. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5300–5304, 2017. https://doi.org/10.1109/ICASSP.2017.7953168.

[29] Umapada Pal, Ramachandran Jayadevan, and Nabin Sharma. Handwriting recognition in indian regional scripts: A survey of offline techniques. *ACM Transactions on Asian Language Information Processing*, 11(1), March 2012. https://doi.org/10.1145/2090176.2090177.

[30] Mohammad Tanvir Parvez and Sabri A. Mahmoud. Offline arabic handwritten text recognition: A survey. *ACM Comput. Surv.*, 45(2), March 2013. https://doi.org/10.1145/2431211.2431222.

[31] Jayashree Rajesh Prasad. Handwritten character recognition: a review. *Int J Comput Sci Netw*, 3(5):340–351, 2014.

[32] Sukhdeep Singh, Anuj Sharma, and Vinod Chauhan. Indic script family and its offline handwriting recognition for characters/digits and words: a comprehensive survey. *Artificial Intelligence Review*, 56:1–53, 09 2023. https://doi.org/10.1007/s10462-023-10597-y.

[33] Eddy Sánchez-DelaCruz and Cecilia-Irene Loeza-Mejía. Importance and challenges of handwriting recognition with the implementation of machine learning techniques: a survey. *Applied Intelligence*, 54:1–22, 05 2024. https://doi.org/10.1007/s10489-024-05487-x.

[34] WS. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *Bull Math Biophys*, 5, 1943. https://doi.org/10.1007/BF02478259.

[35] D.O. Hebb. The organization of behavior. *New York: Wiley & Sons*, 1949.

[36] F. Rossenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958. https://doi.org/10.1037/h0042519.

[37] B. Windrow and M.E. Hoff. Adaptive switching circuits. *IRE WESCON Convention Record*, 4(96-104), 1960. https://doi.org/10.21236/AD0241531.

[38] T. Kohonen. Self-organized formation of topologically correct feature maps. *Biol. Cybern*, 43:59–69, 1982. https://doi.org/10.1007/BF00337288.

[39] J.J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the USA*, 79(8):2554–2558, 1982. https://doi.org/10.1073/pnas.79.8.2554.

[40] D. Rumelhart, G. Hinton, and R. Williams. Learning representations by back-propagating errors. *nature*, 323:533–536, 1986. https://doi.org/10.1038/323533a0.

[41] Gail A. Carpenter and Stephen Grossberg. The art of adaptive pattern recognition by a self-organizing neural network. *Computer*, 21(3):77–88, 1988. https://doi.org/10.1109/2.33.

[42] D.S. Broomhead and D. Lowe. Multivariable functional interpolation and adaptive networks. *Complex Systems*, 2:321–355, 1988.

[43] Kunihiko Fukushima. Neocognitron trained with winner-kill-loser rule. *Neural Netw.*, 23(7):926–938, 2010. https://doi.org/10.1016/j.neunet.2010.04.008.

[44] Simon Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall PTR, 2nd edition, 1998.

[45] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[46] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. https://doi.org/10.1109/CVPR.2015.7298594.

[47] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014. https://doi.org/10.1109/CVPR.2014.220.

[48] Patrice Y Simard, David Steinkraus, John C Platt, et al. Best practices for convolutional neural networks applied to visual document analysis. In *ICDAR*, volume 3, 2003.

[49] D. C. Ciresan, U. Meier, L. M. Gambardella, and J. Schmidhuber. Convolutional neural network committees for handwritten character classification. *In: Proceedings of the International Conference on Document Analysis and Recognition*, pages 1135 – 1139, 2011. https://doi.org/10.1109/ICDAR.2011.229.

[50] Dan Ciregan, Ueli Meier, and Jürgen Schmidhuber. Multi-column deep neural networks for image classification. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3642–3649. IEEE, 2012. https://doi.org/10.1109/CVPR.2012.6248110.

[51] Tao Wang, David J Wu, Adam Coates, and Andrew Y Ng. End-to-end text recognition with convolutional neural networks. In *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pages 3304–3308. IEEE, 2012.

[52] Ian J Goodfellow, Yaroslav Bulatov, Julian Ibarz, Sacha Arnoud, and Vinay Shet. Multi-digit number recognition from street view imagery using deep convolutional neural networks. *arXiv preprint arXiv:1312.6082*, 2013.

[53] Jonathan J Tompson, Arjun Jain, Yann LeCun, and Christoph Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *Advances in neural information processing systems*, pages 1799–1807, 2014.

[54] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015. https://doi.org/10.1038/nature14539.

[55] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[56] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, March 1994. https://doi.org/10.1109/72.279181.

[57] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, Nov 1998. https://doi.org/10.1109/5.726791.

[58] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. https://doi.org/10.1109/5.726791.

[59] Kunihiko Fukushima and Sei Miyake. Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and cooperation in neural nets*, pages 267–285. Springer, 1982. https://doi.org/10.1007/978-3-642-46466-9_18.

[60] S-CB Lo, S-LA Lou, Jyh-Shyan Lin, Matthew T Freedman, Minze V Chien, and Seong Ki Mun. Artificial convolution neural network techniques and applications for lung nodule detection. *IEEE transactions on medical imaging*, 14(4):711–718, 1995. https://doi.org/10.1109/42.476112.

[61] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. https://doi.org/10.1007/s11263-015-0816-y.

[62] H. Sak, Andrew Senior, and F. Beaufays. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 338–342, 01 2014. https://doi.org/10.21437/Interspeech.2014-80.

[63] X. Li and X. Wu. Constructing long short-term memory based deep recurrent neural networks for large vocabulary speech recognition. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4520–4524, April 2015. https://doi.org/10.1109/ICASSP.2015.7178826.

[64] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber. A novel connectionist system for unconstrained handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(5):855–868, May 2009. https://doi.org/10.1109/TPAMI.2008.137.

[65] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, nov 1997. `https://doi.org/10.1162/neco.1997.9.8.1735`.

[66] Felix A Gers and Jürgen Schmidhuber. Recurrent nets that time and count. In *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium*, volume 3, pages 189–194. IEEE, 2000. `https://doi.org/10.1109/IJCNN.2000.861302`.

[67] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[68] M. Schuster and K. K. Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, Nov 1997. `https://doi.org/10.1109/78.650093`.

[69] Altaf Ramjon, Asherl Bwatiramba, and Sivakumar Venkataraman. Handwritten character recognition using deep learning (convolutional neural network). *Computer Engineering and Intelligent Systems*, Volume 14:2023, 02 2023.

[70] Jiajin Zhang, Guoying Peng, Hongyu Yang, Chao Tan, Yaqing Tan, and Hui Bai. Real-time finger-writing character recognition via tof sensors on edge deep learning. *Electronics*, 12(3), 2023. `https://doi.org/10.3390/electronics12030685`.

[71] Emily Xiaoxuan Gu. Convolutional neural network based kannada-mnist classification. In *2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, pages 180–185, 2021. `https://doi.org/10.1109/ICCECE51280.2021.9342474`.

[72] Deepika Gupta and Soumen Bag. Cnn-based multilingual handwritten numeral recognition: A fusion-free approach. *Expert Systems with Applications*, 165:113784, 2021. `https://doi.org/10.1016/j.eswa.2020.113784`.

[73] Huseyin Kusetogullari, Amir Yavariabdi, Abbas Cheddad, Håkan Grahn, and Johan Hall. Ardis: a swedish historical handwritten digit dataset. *Neural Computing and Applications*, 32(21):16505–16518, 2020. `https://doi.org/10.1007/s00521-019-04163-3`.

[74] ELVIS S. GATI, BENJAMIN D. NIMO, and ELISHA K. ASIAMAH. Kannada-mnist classification using skip cnn. In *2019 16th International Computer Conference on Wavelet Active Media Technology and Information Processing*, pages 245–248, 2019. `https://doi.org/10.1109/ICCWAMTIP47768.2019.9067521`.

[75] Sukhdeep Singh, Vinod Kumar Chauhan, and Elisa H Barney Smith. A self controlled rdp approach for feature extraction in online handwriting recognition using deep learning. *Applied Intelligence*, 50(7):2093–2104, 2020. `https://doi.org/10.1007/s10489-020-01632-4`.

[76] Vinay Uday Prabhu. Kannada-mnist: A new handwritten digits dataset for the kannada language, 2019.

[77] K. Manjusha, M. Anand Kumar, and K.P. Soman. On developing handwritten character image database for malayalam language script. *Engineering Science and Technology, an International Journal*, 22(2):637–645, 2019. `https://doi.org/10.1016/j.jestch.2018.10.011`.

[78] Rumman Rashid Chowdhury, Mohammad Shahadat Hossain, Raihan ul Islam, Karl Andersson, and Sazzad Hossain. Bangla handwritten character recognition using convolutional neural network with data augmentation. In *2019 Joint 8th international conference on informatics, electronics & vision (ICIEV) and 2019 3rd international conference on imaging, vision & pattern recognition (icIVPR)*, pages 318–323. IEEE, 2019. `https://doi.org/10.1109/ICIEV.2019.8858545`.

[79] Anisha Gupta, Ritesh Sarkhel, Nibaran Das, and Mahantapas Kundu. Multiobjective optimization for recognition of isolated handwritten indic scripts. *Pattern Recognition Letters*, 128:318–325, 2019. https://doi.org/10.1016/j.patrec.2019.09.019.

[80] Sinjan Chakraborty, Sayantan Paul, Ram Sarkar, and Mita Nasipuri. Feature map reduction in cnn for handwritten digit recognition. In *Recent Developments in Machine Learning and Data Analytics*, pages 143–148. Springer, 2019. https://doi.org/10.1007/978-981-13-1280-9_14.

[81] Shefali Arora and M. P. S Bhatia. Handwriting recognition using deep learning in keras. In *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, pages 142–145, 2018. https://doi.org/10.1109/ICACCCN.2018.8748540.

[82] K Manjusha, M Anand Kumar, and KP Soman. Integrating scattering feature maps with convolutional neural networks for malayalam handwritten character recognition. *International Journal on Document Analysis and Recognition (IJDAR)*, 21(3):187–198, 2018. https://doi.org/10.1007/s10032-018-0308-z.

[83] Lei Kang, J Ignacio Toledo, Pau Riba, Mauricio Villegas, Alicia Fornés, and Marçal Rusinol. Convolve, attend and spell: An attention-based sequence-to-sequence model for handwritten word recognition. In *German Conference on Pattern Recognition*, pages 459–472. Springer, 2018. https://doi.org/10.1007/978-3-030-12939-2_32.

[84] Ritesh Sarkhel, Nibaran Das, Aritra Das, Mahantapas Kundu, and Mita Nasipuri. A multi-scale deep quad tree based feature extraction method for the recognition of isolated handwritten characters of popular indic scripts. *Pattern Recognition*, 71:78–93, 2017. https://doi.org/10.1016/j.patcog.2017.05.022.

[85] Arik Poznanski and Lior Wolf. Cnn-n-gram for handwriting word recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. https://doi.org/10.1109/CVPR.2016.253.

[86] In-Jung Kim and Xiaohui Xie. Handwritten hangul recognition using deep convolutional neural networks. *International Journal on Document Analysis and Recognition (IJDAR)*, 18:1–13, 2015. https://doi.org/10.1007/s10032-014-0229-4.

[87] Li Wan, Matthew Zeiler, Sixin Zhang, Yann Le Cun, and Rob Fergus. Regularization of neural networks using dropconnect. In *International conference on machine learning*, pages 1058–1066. PMLR, 2013.

[88] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

[89] Supriya Mahadevkar, Shruti Patil, and Ketan Kotecha. Enhancement of handwritten text recognition using ai-based hybrid approach. *MethodsX*, 12:102654, 2024. https://doi.org/10.1016/j.mex.2024.102654.

[90] Vu Pham, Théodore Bluche, Christopher Kermorvant, and Jérôme Louradour. Dropout improves recurrent neural networks for handwriting recognition. In *2014 14th international conference on frontiers in handwriting recognition*, pages 285–290. IEEE, 2014. https://doi.org/10.1109/ICFHR.2014.55.

[91] Patrick Doetsch, Michal Kozielski, and Hermann Ney. Fast and robust training of recurrent neural networks for offline handwriting recognition. In *2014 14th International Conference on Frontiers in*

Handwriting Recognition, pages 279–284. IEEE, 2014. https://doi.org/10.1109/ICFHR.2014.54.

[92] Francois Chollet et al. Keras, 2015.

[93] Youssouf Chherawala, Partha Pratim Roy, and Mohamed Cheriet. Feature set evaluation for offline handwriting recognition systems: Application to the recurrent neural network model. *IEEE Transactions on Cybernetics*, 46(12):2825–2836, Dec 2016. https://doi.org/10.1109/TCYB.2015.2490165.

[94] Yaroslav Shkarupa, Roberts Mencis, and Matthia Sabatelli. Offline handwriting recognition using lstm recurrent neural networks. In *The 28th Benelux conference on artificial intelligence*, 2016.

[95] Curtis Wigington, Seth Stewart, Brian Davis, Bill Barrett, Brian Price, and Scott Cohen. Data augmentation for recognition of handwritten words and lines using a cnn-lstm network. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 01, pages 639–645, 2017. https://doi.org/10.1109/ICDAR.2017.110.

[96] Kartik Dutta, Praveen Krishnan, Minesh Mathew, and CV Jawahar. Improving cnn-rnn hybrid networks for handwriting recognition. In *2018 16th international conference on frontiers in handwriting recognition (ICFHR)*, pages 80–85. IEEE, 2018. https://doi.org/10.1109/ICFHR-2018.2018.00023.

[97] Praveen Krishnan, Kartik Dutta, and CV Jawahar. Word spotting and recognition using deep embedding. In *2018 13th IAPR International Workshop on Document Analysis Systems (DAS)*, pages 1–6. IEEE, 2018. https://doi.org/10.1109/DAS.2018.70.

[98] Jorge Sueiras, Victoria Ruiz, Angel Sanchez, and Jose F. Velez. Offline continuous handwriting recognition using sequence to sequence neural networks. *Neurocomputing*, 289:119–128, 2018. https://doi.org/10.1016/j.neucom.2018.02.008.

[99] Rajib Ghosh, Chirumavila Vamshi, and Prabhat Kumar. Rnn based online handwritten word recognition in devanagari and bengali scripts using horizontal zoning. *Pattern Recognition*, 92:203–218, 2019. https://doi.org/10.1016/j.patcog.2019.03.030.

[100] R Geetha, T Thilagam, and T Padmavathy. Effective offline handwritten text recognition model based on a sequence-to-sequence approach with cnn–rnn networks. *Neural Computing and Applications*, 33(17):10923–10934, 2021. https://doi.org/10.1007/s00521-020-05556-5.

[101] Jan Koutnik, Klaus Greff, Faustino Gomez, and Juergen Schmidhuber. A clockwork rnn. In *International Conference on Machine Learning*, pages 1863–1871. PMLR, 2014.

[102] Dayvid Castro, Cleber Zanchettin, and Luís Amaral. On the improvement of handwritten text line recognition with octave convolutional recurrent neural networks. *International Journal on Document Analysis and Recognition (IJDAR)*, 27:567–581, 02 2024. https://doi.org/10.1007/s10032-024-00460-3.

[103] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. https://doi.org/10.1109/CVPR.2015.7298594

[104] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[105] Siti Salwa Md Noor, Jinchang Ren, Stephen Marshall, and Kaleena Michael. Hyperspectral image enhancement and mixture deep-learning classification of corneal epithelium injuries. *Sensors*, 17(11):2644, 2017. https://doi.org/10.3390/s17112644.

[106] Tara N Sainath, Brian Kingsbury, George Saon, Hagen Soltau, Abdel-rahman Mohamed, George Dahl, and Bhuvana Ramabhadran. Deep convolutional neural networks for large-scale speech tasks. *Neural networks*, 64:39–48, 2015. https://doi.org/10.1016/j.neunet.2014.08.005.

[107] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. Ieee, 2013. https://doi.org/10.1109/ICASSP.2013.6638947.

[108] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017. https://doi.org/10.1109/TPAMI.2016.2644615.

[109] Ronghang Hu, Piotr Dollár, Kaiming He, Trevor Darrell, and Ross Girshick. Learning to segment every thing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4233–4241, 2018. https://doi.org/10.1109/CVPR.2018.00445.

[110] Oriol Vinyals, Lukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton. Grammar as a foreign language. *arXiv preprint arXiv:1412.7449*, 2014.

[111] Yang Liu and Jiajun Zhang. Deep learning in machine translation. In *Deep Learning in Natural Language Processing*, pages 147–183. Springer, 2018. https://doi.org/10.1007/978-981-10-5209-5_6.

[112] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137, 2015. https://doi.org/10.1109/CVPR.2015.7298932.

[113] Pedro Pinheiro and Ronan Collobert. Recurrent convolutional neural networks for scene labeling. In *International conference on machine learning*, pages 82–90. PMLR, 2014.

[114] Aditya Grover, Ashish Kapoor, and Eric Horvitz. A deep hybrid model for weather forecasting. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 379–386, 2015. https://doi.org/10.1145/2783258.2783275.

[115] Moinul Hossain, Banafsheh Rekabdar, Sushil J Louis, and Sergiu Dascalu. Forecasting the weather of nevada: A deep learning approach. In *2015 international joint conference on neural networks (IJCNN)*, pages 1–6. IEEE, 2015. https://doi.org/10.1109/IJCNN.2015.7280812.

[116] Shadi Albarqouni, Christoph Baur, Felix Achilles, Vasileios Belagiannis, Stefanie Demirci, and Nassir Navab. Aggnet: deep learning from crowds for mitosis detection in breast cancer histology images. *IEEE transactions on medical imaging*, 35(5):1313–1321, 2016. https://doi.org/10.1109/TMI.2016.2528120.

[117] Dayong Wang, Aditya Khosla, Rishab Gargeya, Humayun Irshad, and Andrew H Beck. Deep learning for identifying metastatic breast cancer. *arXiv preprint arXiv:1606.05718*, 2016. https://doi.org/10.48550/arXiv.1606.05718.

[118] Angel Alfonso Cruz-Roa, John Edison Arevalo Ovalle, Anant Madabhushi, and Fabio Augusto González Osorio. A deep learning architecture for image representation, visual interpretability and automated basal-cell carcinoma cancer detection. In *International Conference on Medical Image Computing and*

Computer-Assisted Intervention, pages 403–410. Springer, 2013. https://doi.org/10.1007/978-3-642-40763-5_50.

[119] Ayşegül Uçar, Yakup Demir, and Cüneyt Güzeliş. Object recognition and detection with deep learning for autonomous driving applications. *Simulation*, 93(9):759–769, 2017. https://doi.org/10.1177/0037549717709932.

[120] Chenyi Chen, Ari Seff, Alain Kornhauser, and Jianxiong Xiao. Deepdriving: Learning affordance for direct perception in autonomous driving. In *Proceedings of the IEEE international conference on computer vision*, pages 2722–2730, 2015. https://doi.org/10.1109/ICCV.2015.312.

[121] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu, and G. Zweig. Achieving human parity in conversational speech recognition, 2017. https://doi.org/10.1109/TASLP.2017.2756440.

[122] George Saon, Gakuto Kurata, Tom Sercu, Kartik Audhkhasi, Samuel Thomas, Dimitrios Dimitriadis, Xiaodong Cui, Bhuvana Ramabhadran, Michael Picheny, Lynn-Li Lim, Bergul Roomi, and Phil Hall. English conversational telephone speech recognition by humans and machines, 2017. https://doi.org/10.48550/arXiv.1703.02136.

[123] Varun Gulshan, Lily Peng, Marc Coram, Martin C Stumpe, Derek Wu, Arunachalam Narayanaswamy, Subhashini Venugopalan, Kasumi Widner, Tom Madams, Jorge Cuadros, et al. Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *Jama*, 316(22):2402–2410, 2016. https://doi.org/10.1001/jama.2016.17216.

[124] Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *nature*, 542(7639):115–118, 2017. https://doi.org/10.1038/nature21056.

[125] Paras Lakhani and Baskaran Sundaram. Deep learning at chest radiography: automated classification of pulmonary tuberculosis by using convolutional neural networks. *Radiology*, 284(2):574–582, 2017. https://doi.org/10.1148/radiol.2017162326.

[126] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. https://doi.org/10.48550/arXiv.1312.6199.

[127] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. https://doi.org/10.48550/arXiv.1412.6572

[128] Xue-Wen Chen and Xiaotong Lin. Big data deep learning: challenges and perspectives. *IEEE access*, 2:514–525, 2014. https://doi.org/10.1109/ACCESS.2014.2325029.

[129] Maryam M Najafabadi, Flavio Villanustre, Taghi M Khoshgoftaar, Naeem Seliya, Randall Wald, and Edin Muharemagic. Deep learning applications and challenges in big data analytics. *Journal of big data*, 2(1):1–21, 2015. https://doi.org/10.1186/s40537-014-0007-7.

[130] James Bergstra, Frédéric Bastien, Olivier Breuleux, Pascal Lamblin, Razvan Pascanu, Olivier Delalleau, Guillaume Desjardins, David Warde-Farley, Ian Goodfellow, Arnaud Bergeron, et al. Theano: Deep learning on gpus with python. In *NIPS 2011, BigLearning Workshop, Granada, Spain*, volume 3, pages 1–48. Citeseer, 2011.