Informatics in Medicine Unlocked 26 (2021) 100723

Contents lists available at ScienceDirect

Informatics in Medicine Unlocked

journal homepage: www.elsevier.com/locate/imu

An overview of deep learning in medical imaging

Andrés Anaya-Isaza [a,d], Leonel Mera-Jiménez [b,d,*], Martha Zequera-Diaz [c]

[a] Pontificia Universidad Javeriana, 110231, Bogotá, Colombia
[b] Faculty of Engineering, Bioengineering Program, Universidad de Antioquia, 050010, Medellín, Colombia
[c] School of Engineering, BASPI/FootLab (Bioengineering, Signal Analysis, And Image Processing Research Group), Pontificia Universidad Javeriana, 110231, Bogotá, Colombia
[d] INDIGO Research, 110221, Bogotá, Colombia

ARTICLE INFO

Keywords:
Attention models
Convolutional neural networks
Deep learning
Medical imaging
Recurrent neural networks
Transformers

ABSTRACT

Deep learning (DL) is one of the branches of artificial intelligence that has seen exponential growth in recent years. The scientific community has focused its attention on DL due to its versatility, high performance, high generalization capacity, and multidisciplinary uses, among many other qualities. In addition, a large amount of medical data and the development of more powerful computers has also fostered an interest in this area. This paper presents an overview of current deep learning methods, starting from the most straightforward concept but accompanied by the mathematical models that are behind the functionality of this type of intelligence. In the first instance, the fundamental concept of artificial neural networks is introduced, progressively covering convolutional structures, recurrent networks, attention models, up to the current structure known as the Transformer. Secondly, all the basic concepts involved in training and other common elements in the design of the architectures are introduced. Thirdly, some of the key elements in modern networks for medical image classification and segmentation are shown. Subsequently, a review of some applications realized in the last years is shown, where the main features related to DL are highlighted. Finally, the perspectives and future expectations of deep learning are presented.

1. Introduction

The concept of artificial intelligence is not new; great geniuses such as Leonardo Da Vinci tried to create automatons that emulated human tasks [1]. Today it seems that this is already a reality. We are getting closer and closer to the singularity mentioned by Nicolas de Condorcet [2], established as the hypothetical advent of vital artificial intelligence, where new bits of intelligence could recursively self-improve, triggering exponential growth of artificial intelligence [3]. Although few intelligent systems are currently self-adjusting, we have seen exponential growth in artificial intelligence, especially in medical informatics [4]. The growing developments are primarily due to the paradigm shift towards deep learning (DL) systems, being attractive to most researchers due to the current models' efficiency and "simplicity." In fact, for most implementations, it is sufficient to view DL systems as a black box to which we provide input and output data as a reference for the desired training (supervised learning) [5].

The central concept of current artificial intelligence systems is artificial neural networks [6]. The network consists of many individual units (artificial neuron or perceptron), emulating the activation state of the biological neuron from the inputs interacting with it [7]. Like a biological neural network, neurons modify the connections between them through the training process in the artificial neural network [8]. Although the connection is not deactivated or activated, in the strict sense of the word, the connection weights are modified until the desired task is achieved [8]. The training is performed by one of the fundamental algorithms of the DL, known as backpropagation, which determines the error imputed to each neuron, allowing to adjust the network parameters efficiently [9]. The complexity of the network allows it to manipulate a large amount of data to solve problems like the human brain. In addition, deep learning approaches have succeeded in clustering neural layers in hierarchical order to tackle more challenging problems without extracting features or defining a hypothesis about the data of interest [10]. The advantages of DL stand out immediately, and developments are not made to wait, showing its high performance in tasks such as segmentation, classification, detection, pattern search, natural language processing, and prognostics, among an extensive list of tasks [11–13].

* Corresponding author. Faculty of Engineering, Bioengineering Program, Universidad de Antioquia, 050010, Medellín, Colombia.
E-mail address: leonel.mera@udea.edu.co (L. Mera-Jiménez).

https://doi.org/10.1016/j.imu.2021.100723
Received 13 July 2021; Received in revised form 19 August 2021; Accepted 31 August 2021
Available online 4 September 2021

The interest in DL is visualized in the frequent calls, challenges, conferences, or all the research groups worldwide presenting their results [14]. New developments are continuously published, and the various contributions make the current models more and more efficient [15]. In addition, the generation of large amounts of digital data, robust computational infrastructures, graphics processing units (GPU), and cloud computing have also fostered the growth of DL in various scientific fields, where medicine is no exception to this rule [16,17]. For example, the large amount of data used in cancer diagnosis has enabled the efficient integration of DL algorithms [18]. Generally, intelligence learns to classify clinical, radiological, or pathological images into a preset category [19]. In general, this requires training based on pathological features extracted by expert personnel, allowing to address problems as complex as the degree of severity [20], the type (malignant or benign) [21,22], a specific diagnosis [23] and even the probability of survival of a patient [24].

In the case of cancer, the implications of possible successful implementations are evident. Having diagnostic aid systems would help in the tasks of the radiologist or expert professional and reduce diagnostic time or even lead to more accurate diagnoses and timely treatment [[25],26]. In addition, this would have a substantial impact on public health since cancer is the second leading cause of death worldwide, responsible for 9.6 million deaths, according to figures from the World Health Organization (WHO) [27]. For example, referencing some cases, breast cancer is the leading cause of death among women aged 20–50 years and, according to 2019 figures from the American Cancer Society, in the United States alone, there were an estimated 268,600 new cases of invasive breast cancer, 48,100 of ductal carcinoma in situ (DCIS) and 41,740 deaths [28,29]. The figures reached 684,996 deaths worldwide by 2020, making it the leading cause of female cancer mortality [30,31]. Even at the beginning of 2021, the WHO reported this pathology as the most common cancer worldwide, surpassing lung cancer [32] and representing only 10.4% of cancers worldwide. Likewise, lung cancer is one of the most aggressive pathologies, generating about 22% of deaths, i.e., it is one of the leading causes of death from cancer [27]. Figures reveal that, in the United States, about 135 thousand deaths were estimated in 2020, with about 228 thousand new cases diagnosed [33]. In general, lung cancer has a very high mortality rate, where 90% of patients do not have a life expectancy of more than five years, and about half of them have advanced or metastatic cancer [34]. Similarly, brain tumors cause severe damage to the nervous system, generating reduced survival rates (less than 21% at five years in people older than 40 years [35]). Fortunately, the scientific and medical community are increasing their efforts to reduce these figures from different areas of research, where artificial intelligence would not be needed. Moreover, with the advancement of different image acquisition systems, current images have much higher resolutions than a few decades ago [36]. Consequently, we have only just begun to take full advantage of the related advantages in diagnostic aid systems through artificial intelligence, or more specifically, through deep learning.

Deep learning applications on medical images are recent. In fact, the turning point dates back to 2012 (less than a decade), where neural networks started to outperform conventional computer vision methods. The ImageNet Large Scale Visual Recognition Challenge was the main event that showed the advantages of these new strategies [37]. Since that point, DL research in medical imaging has increased exponentially [4]. Currently, new investigations with faster, deeper, and more efficient networks are emerging. In this sense, the scientific community interested in this area should be updated and aware of the fundamental concepts and the most recent developments, as shown by the most recent and relevant reviews [38,39]. In this order of ideas, in this review, we address two significant challenges. The first one is to clarify the details and concepts of deep learning, including the main mathematical models behind the operation of the networks, and to deliver an intuitive description of the concepts along with the current state-of-the-art networks. Secondly, we perform a rigorous review of the most recent developments, focusing mainly on medical imaging and deep learning oriented to cancer pathologies. This review highlights significant themes, research questions or observations, future projections, and potential research areas that have not yet been covered.

In particular, DL is versatile and straightforward from a black-box point of view. However, there are many concepts involved ranging from design to training and implementation of the models, being necessary to understand them clearly to get the most out of AI. In this sense and approach to address the challenges above, this paper focuses on showing the key elements, the different architectures, and the most recent implementations in medical imaging. The paper is organized as follows: Section 2 introduces the basic idea behind artificial neural networks. Section 3 shows one of the first neural networks, known as a multilayer perceptron or fully connected network. Section 4 deals with convolutional neural networks developed for image processing. Section 5 presents recurrent networks designed for time-dependent data. Section 6 outlines the intuition behind attention models, a concept essential to understanding the Transformers discussed in section 7. Section 8 shows all the basic concepts related to artificial neural networks. Subsequently, in Section 9, we provide an overview of current neural networks, concluding with medical imaging and its applications in section 10 and ending with perspectives and future expectations in section 11.

2. Artificial neural networks

Artificial neural networks (ANNs) are one of the first bio-inspired systems based on the functioning of the human brain. In principle, the complexity of the brain is governed by the millions of neurons and trillions of connections that make up the brain structure. However, functioning is the result of the contribution of each neuron. Neurons receive electrochemical signals from other neurons or signals generated from the different tissues that make up all our senses, such as vision. The signals are processed by each neuron and depending on the interactions with the other neurons, the action potential can be reached, polarizing the neuron's axon, and allowing the transmission of the electrical signal to continue (synapse). In artificial neural networks, the perceptron is the main element of the network, also known as the base unit or artificial neuron. Similarly, the perceptron is connected to other perceptrons, receiving information from them to produce the activation of the neuron through a mathematical element known as the activation function (see Section 8.6). Thus, the different activations propagate through the network generating the response to the task of interest. Likewise, the task results from the contribution of all the neurons that compose the artificial network, where most of them are governed by the simple model of Equation (1).

$$y = f(b + \alpha(x_i)) \tag{1}$$

In this model, f is the activation function (nonlinear function), b a constant, and α a function with the training weights or parameters [40].

In the simplest version, it is possible to understand ANNs as black boxes with hidden training parameters, which can be trained or learned similarly as the human brain does, i.e., the artificial network also needs to learn, a process it performs through many examples (supervised learning) [41]. Once the model or network is adequately trained, it can generate automatic responses in new examples, as conceptualized in Fig. 1.

3. Multilayer perceptron or fully connected networks

There are many artificial neural network types, with the multilayer perceptron or fully connected network being the simplest version. The network consists of a layered structure created by an input layer, an output layer, and one or more hidden layers. Each layer comprises several units interconnected with neighboring layers (input and output) but without connections between the units of the same layer. In general, the network layers consist only of several units known as artificial

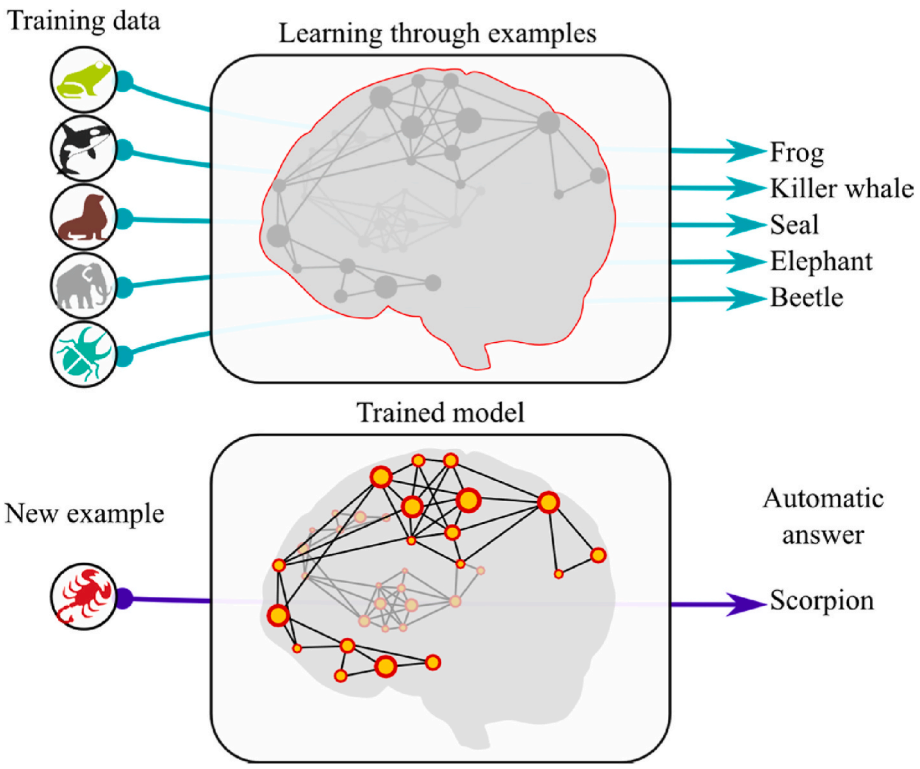


Fig. 1. Artificial intelligence is represented as black boxes—an example of supervised learning.

neurons or, in this case, perceptron. As mentioned in the previous section, the base unit or perceptron resembles the biological model of the neuron. The perceptron sums the input values, multiplied by the weights representing the synaptic interactions of the biological neurons. The weights are known as training parameters and are adjusted later in the training of the network. Finally, the weighted sum is introduced to the activation function (nonlinear function), simulating the activation, or generating the output of each perceptron (see Fig. 2). Despite the small number of elements that make up the multilayer perceptron structure, the architecture can be designed with infinite configurations since there is no limit[1] to the perceptrons per layer and layers per network [40].

The simple perceptron model can cover to a robust and complex mathematical model in proportion to many layers and perceptrons per layer. For example, given the vector of inputs $x = [x_1, x_2, x_3, \ldots, x_n] \forall\, x_i \in R$ one can write the output y_1 of the first perceptron as given in Equation (2).

$$y_1(\boldsymbol{x}, \boldsymbol{w}_1, b_1) = f\left(b_1 + \sum_{i=1}^{4} w_{1i} x_i\right) \tag{2}$$

Where $w_1 = [w_{11}, w_{12}, w_{13}, \ldots, w_{1n}]$ and b_1 are the training parameters and f the activation function. Equation (2) can also be written as a composition of the inner product or dot product as in Equation (3).

$$y_1(\boldsymbol{x}, \boldsymbol{w}_1) = f(b_1 + \boldsymbol{w}_1{}^t \boldsymbol{x}) \tag{3}$$

Here, the super index t represents the transpose of the vector. Furthermore, if $\boldsymbol{x}$ and $\boldsymbol{w}_1$ are rewritten as in Equations (4) and (5), Equation (3) would take the form expressed in Equation (6).

$$\mathbf{x} = [x_0, x_1, x_2, x_3, \ldots, x_n] x_0 = 1 \tag{4}$$

$$\mathbf{w}_1 = [b_1, w_{11}, w_{12}, w_{13}, \ldots, w_{1n}] \tag{5}$$

$$y_1(\boldsymbol{x}, \boldsymbol{w}_1) = f(\boldsymbol{w}_1{}^t \boldsymbol{x}) \tag{6}$$

Since the perceptrons are similar, the output of each of them can be written as shown in Equation (7) through (10).

$$y_2(\boldsymbol{x}, \boldsymbol{w}_2) = f(\boldsymbol{w}_2{}^t \boldsymbol{x}) \tag{7}$$

$$y_3(\boldsymbol{x}, \boldsymbol{w}_3) = f(\boldsymbol{w}_3{}^t \boldsymbol{x}) \tag{8}$$

$$y_4(\boldsymbol{x}, \boldsymbol{w}_4) = f(\boldsymbol{w}_4{}^t \boldsymbol{x}) \tag{9}$$

$$\vdots$$

$$y_m(\boldsymbol{x}, \boldsymbol{w}_m) = f(\boldsymbol{w}_m{}^t \boldsymbol{x}) \tag{10}$$

Therefore, if the outputs are organized vectorially, the first layer would be governed by the mathematical Equation (11) or its equivalent (12) to include the term associated with the bias.

$$\boldsymbol{y} = [y_1, y_2, y_3, \ldots, y_m] \tag{11}$$

$$\boldsymbol{y} = [y_0, y_1, y_2, y_3, \ldots, y_m], y_0 = 1 \tag{12}$$

On the other hand, if all perceptrons in the first layer have the same activation function, it is possible to further reduce the mathematical model by ordering the m parameter vectors in the matrix (13).

$$W = \begin{bmatrix} 1 & 0 & 0 & 0 & \ldots & 0 \\ b_1 & w_{11} & w_{12} & w_{13} & \ldots & w_{1n} \\ b_2 & w_{21} & w_{22} & w_{23} & \ldots & w_{2n} \\ b_3 & w_{31} & w_{32} & w_{33} & \ldots & w_{3n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ b_m & w_{m1} & w_{m2} & w_{m3} & \ldots & w_{mn} \end{bmatrix} \tag{13}$$

Consequently, the output for the first layer would take the form shown in Equation (14).

$$\boldsymbol{y} = f(W\boldsymbol{x}) \tag{14}$$

Since the network layers are similar, Equation (14) can be

[1] Except for the limitations generated by computational resources.

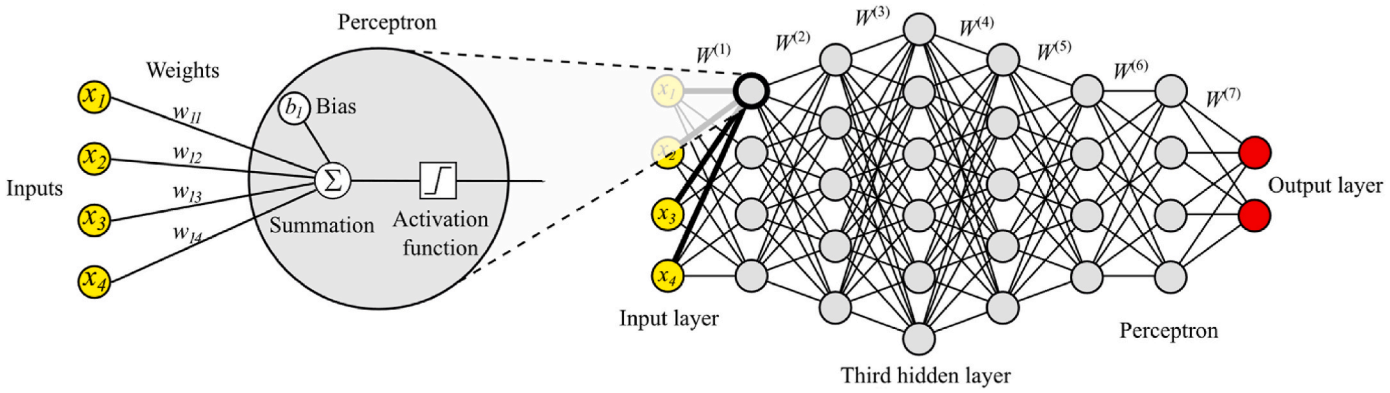


Fig. 2. Artificial neural perceptron and multilayer perceptron.

generalized for any hidden layer, as shown in Equation (15).

$$y^{(l)} = f^{(l)}\left(W^{(l)} y^{(l-1)}\right) \tag{15}$$

Where, $y^{(l)}$ represents the output for the l-th hidden layer, $W^{(l)}$ and $f^{(l)}$ the training parameters (weights) and activation function for the same layer and $\mathbf{y}^{(l-1)}$ the input of the current layer (output of the previous layer), i.e., as the process is repeated for all layers, the output generated would be the input of the next layer. It should be noted that the sequentiality of the MLP model applies to all types of artificial neural networks [42].

4. Convolutional neural networks

Much of the growth of DL is mainly due to the advances made in computer vision. In fact, one of the most widely used algorithms in this field is convolution, from which the convolutional neural network (CNN) is derived, a system inspired by the primary visual cortex. The network could decipher or learn the most complex patterns existing in a set of images, and it does so by employing convolution. Fundamentally, convolution consists of a two-function operator, the image and the filter or kernel. The function takes part of the image and highlights patterns by multiplying each point of the image fragment with the filter elements. The result is weighted, and the generated values are placed in the position corresponding to the image fragment. The process is repeated by moving the filter across the entire image, creating an image with highlighted features that depend on the filter structure.

In the case of CNN, the convolution is performed in the same way. The images generated by the convolution are known as feature maps. In addition, the bias to each map element is included here, and each map is

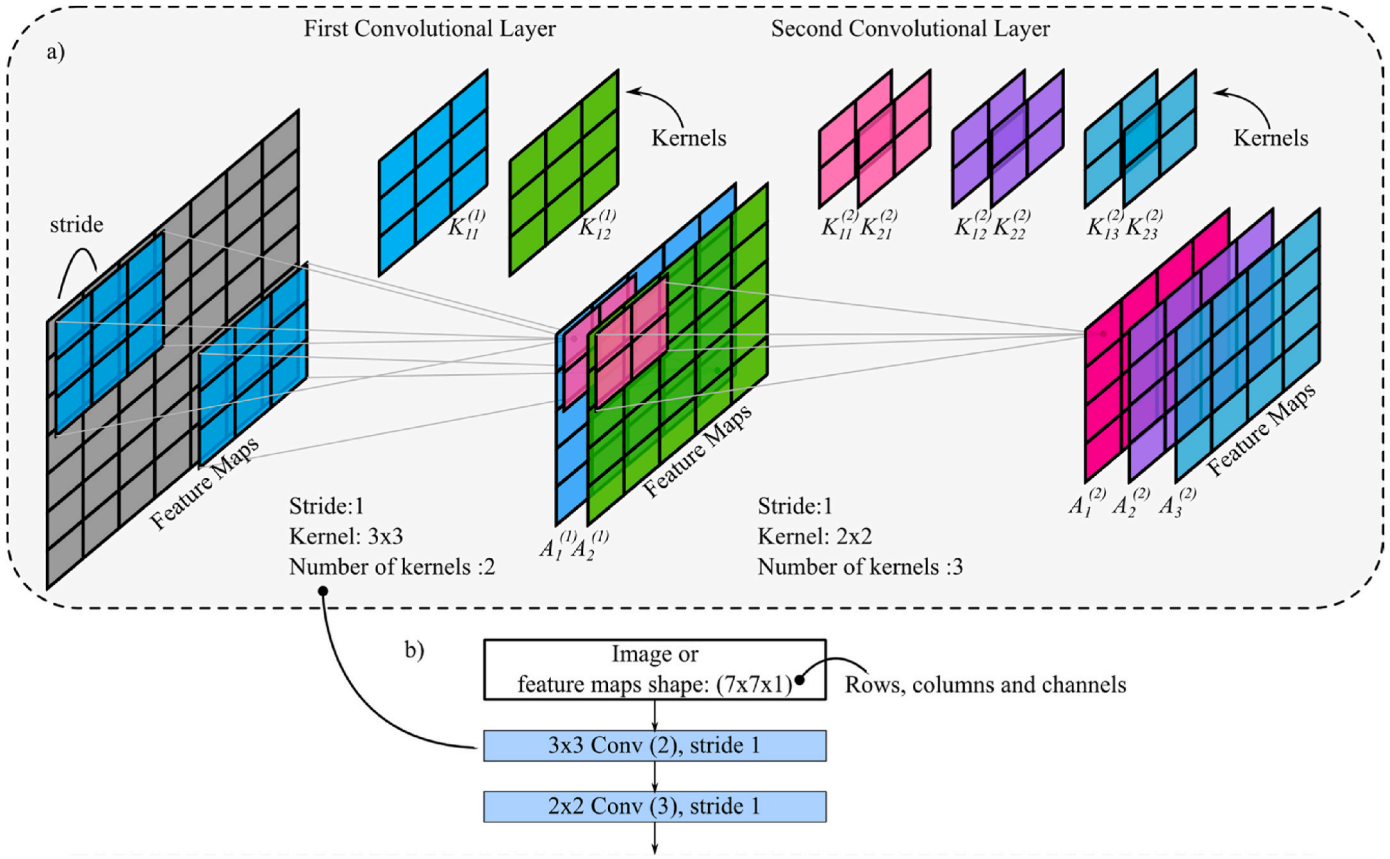


Fig. 3. a) Description of a two-layer convolutional neural network with 2 and 3 filters, respectively. b) compact representation of the same network.

subjected to the activation function, fulfilling the description of Equation (1). In this context, the training parameters of the network are the weights associated with all filters, i.e., the network learns the optimal filters to highlight the high-level features that converge to the desired task. The process is repeated layer after layer creating increasingly abstract features. In addition, the CNN architecture can also be implemented with other types of networks, such as fully connected or attention models (see section 6).

Fig. 3 shows the graphical description of the convolution process where the input image generates one or more feature maps depending on the number of filters for that layer, i.e., if the network has two filters in the first layer, then two feature maps would be generated in the output. Similarly, each layer can have any filter size and number of strides desired. The stride is the number of hops in which the filter moves along the network. It should be noted that each filter generates a new feature map independent of the number of input maps. However, the filter depth changes to match the number of maps, as illustrated in Fig. 3.

It should be clarified that, although Fig. 3a is a more descriptive graph, generally, the convolutional network schemes are presented as in Fig. 3b, limited only to the size, number, and strides of the filter.

The convolution process in the CNN layers is governed under the following mathematical expression (16).

$$A_1 = f\left(\sum_{i=1}^{M} X_i * K_{i1} + b_1\right) \tag{16}$$

Where, for M feature maps, * represents the convolution between the i-th map X_i and the filter K_{i1} at the map equivalent depth. Again, in this case, b_1 and f are the bias and the activation function, respectively. Similarly, following similar reasoning as in the previous section, each feature map generated by the j-th kernel in the l-th layer of the network is governed by the following mathematical expression (17).

$$A_j^{(l)} = f^{(l)}\left(\sum_{i=1}^{M^{(l-1)}} A_i^{(l-1)} * K_{ij}^{(l)} + b_j^{(l)}\right) \tag{17}$$

Additionally, the mathematical sequentiality described in the MLP can also be implemented in this architecture, where for each given layer, its input is the output of the previous layer [43–45].

5. Recurrent neural network

Recurrent neural networks (RNN) are networks with feed-forward loops capable of preserving sequential characteristics, i.e., they can deal with problems with time dependence. The main applications are focused on translations (text sequence), audios, and videos, among the most frequent problems. In medicine, they can be temporal signals associated with the study of a pathology, such as electromyographic signals or longitudinal records to study the evolution of the pathology. RNNs can store information from previous data through hidden states and, together with the current input, the output associated with that sequence is calculated. RNNs could be considered to 'memorize' the preliminary information to arrive at the desired prediction in practice. The information is driven through the different layers, converging on the output associated with the task's performance, such as prediction or classification. RNNs follow the same principles as MLP and CNN networks, while the only difference is that the network's output will depend on initial values, as shown below.

Following the first definition of a neural network model, the mathematical model for an RNN at time t can be formalized as shown in Equation (18).

$$y_t = f(x_t w_{xh} + y_{t-1} w_{hh} + b_h) \tag{18}$$

Equation (18) is similar to the first definition of neural networks (see Equation (1)), but this one has an additional term that is associated with the output of a previous time (y_{t-1}). The term is known as the hidden state and is usually denoted as h_{t-1}. Similarly, the terms W_{xh}, W_{hh}, and b_h are the training parameters, i.e., the weights and biases of the model. f is the activation function. In addition, as it is customary to work with several observations, Equation (18) is usually expressed matrixial as in Equation (19).

$$H_t = \varphi(X_t W_{xh} + H_{t-1} W_{hh} + \boldsymbol{b}_h) \tag{19}$$

Again, the network layers are formed by the grouping of several neurons, as illustrated in Fig. 4, where, unlike Fig. 2, this network does have connections between neurons in the same layer.

The design with time dependence usually delivers accurate results; however, there are two main problems. The first is that most sequential data do not have a fixed size. For example, electroencephalography could have a 2-min record at a sampling rate of 5 kHz or a 10-min record at the same sampling rate. Secondly, having many hidden states for several previous steps could overwhelm the network's capacity and even generate instability in the model weights due to the effect known as vanishing gradient. In this sense, several solutions have been proposed to deal with this drawback, being the gated recurrent unit (GRU) and long short-term memory (LSTM) the most common ones [46,47].

Unlike the conventional recurrent neuron, GRUs are designed to control hidden state activation, i.e., they consider the relevance of the previous state to update or restore that state. For example, if the first piece of data is of high importance, the unit learns not to update the hidden state after this piece of data. Similarly, the unit learns to omit irrelevant observations or to restore the latent state. The neuron has three gates to perform this process, generated from the input, namely the reset gate, the update gate, and the hidden state candidate (see Fig. 5).

The reset gate allows controlling how much of the previous state is remembered to generate a candidate hidden state, while the update gate controls how much of the state is just a copy of the previous one, generating the hidden state based on the candidate state. The hidden state H_t is expressed mathematically by Equation (20).

$$H_t = Z_t \odot H_{t-1} + (1 - Z_t) \odot \widetilde{H}_t \tag{20}$$

Z_t is the update gate, and $\widetilde{H}_t$ the candidate hidden state, governed by Equations (21) and (22).

$$Z_t = \varphi(X_t W_{xz} + H_{t-1} W_{hz} + b_z) \tag{21}$$

$$\widetilde{H}_t = tanh(X_t W_{xh} + (R_t \odot H_{t-1}) W_{hh} + b_h) \tag{22}$$

Here, R_t is the reset gate described by Equation (23).

$$R_t = \varphi(X_t W_{xr} + H_{t-1} W_{hr} + b_r) \tag{23}$$

Like the previous models, the $W's$ and $b's$ terms present the training parameters associated with each gate. X_t is the input for a given time t, H_{t-1} the previous hidden state, $\odot$ is the Hadamard product or product by elements, and φ the activation function [48,49].

LSTMs have similar behavior to GRUs and are slightly more complex even though they predate GRUs. The LSTM has four gates; the first combines the previous hidden state, the input, and the previous memory (C_t) to produce the new hidden state, called the output gate. The second (input gate) decides the activation state of the candidate memory. The third is a system for restoring the contents of the cell (forget gate). Finally, a gate is needed to generate a memory candidate, considered another hidden state of the neuron (see Fig. 6).

Like GRU, the output of each gate is governed by the current input, the previous hidden state, and the training parameters, as shown in Equations (24)–(27). The equations describe the forget gate, input gate, memory candidate, and output, respectively.

$$F_t = \varphi\left(X_t W_{xf} + H_{t-1} W_{hf} + b_f\right) \tag{24}$$

$$I_t = \varphi(X_t W_{xi} + H_{t-1} W_{hi} + b_i) \tag{25}$$

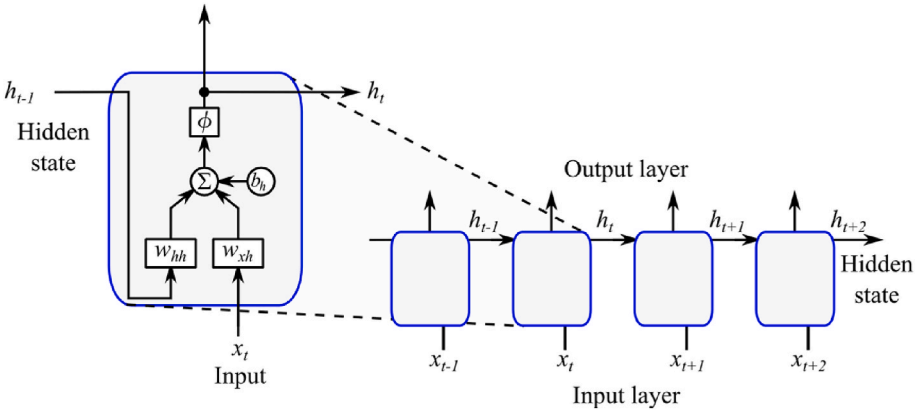


Fig. 4. Neural layer with four simple recurrent artificial neurons.

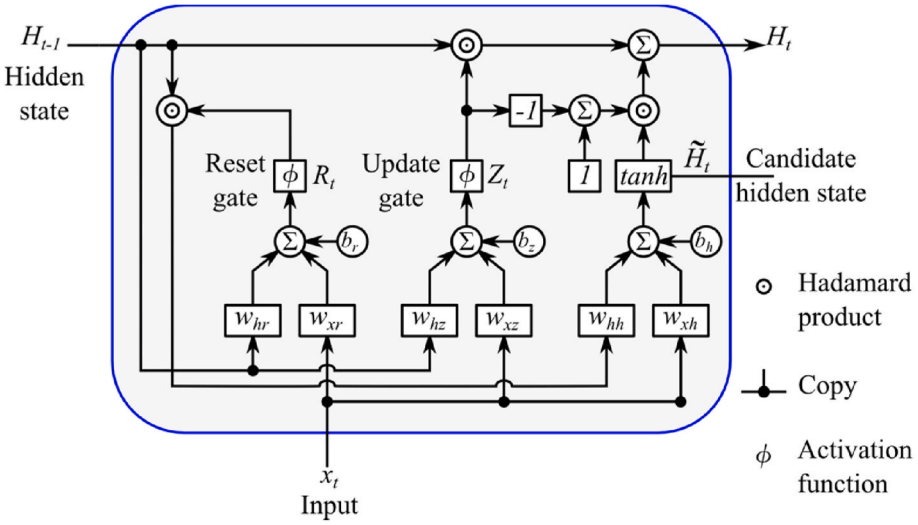


Fig. 5. Calculation of the hidden state in a closed recurrent unit (GRU).

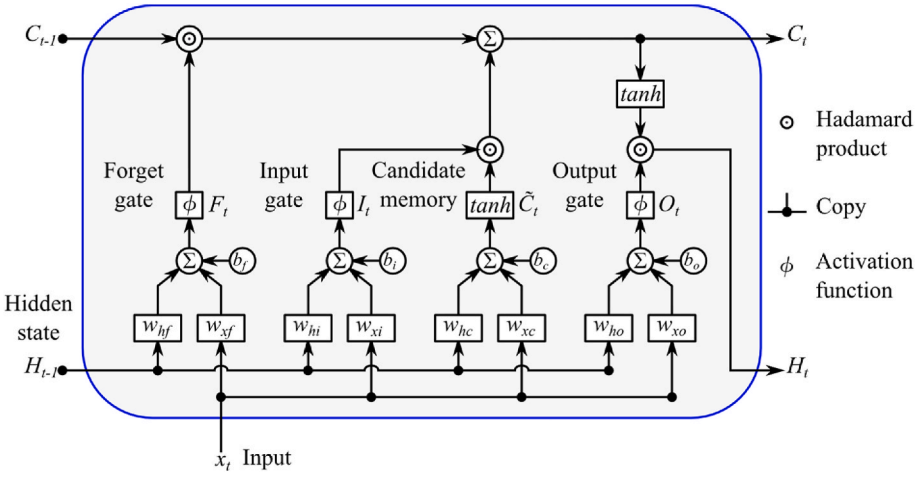


Fig. 6. Recurrent network with long short-term memory or LSTM.

$$\tilde{C}_t = tanh(X_t W_{xc} + H_{t-1} W_{hc} + b_c) \quad (26)$$

$$O_t = \varphi(X_t W_{xo} + H_{t-1} W_{ho} + b_o) \quad (27)$$

Then, the memory C_t and the hidden state H_t the neuron outputs expressed by Equations (28) and (29), respectively.

$$C_t = F_t \odot C_{t-1} + I_t \odot \tilde{C}_t \quad (28)$$

$$H_t = O_t \odot \tanh(C_t) \quad (29)$$

Again, in Equation (24) through (29), the terms $W's$ and $b's$ present the training parameters associated with each gate. X_t is the input for a given time t, H_{t-1} the previous hidden state, $\odot$ is the Hadamard product or product by elements and φ the activation function [50].

6. Attention models

The attention mechanism is another of the bio-inspired systems. The principle of attention models is based on the optic nerve of the visual system. The eye's retina receives a large amount of information from the environment, which would far exceed the human brain's capacity. Fortunately, not all perceived information has the same degree of importance. The brain focuses its attention on objects of interest. For example, as a survival mechanism for humans, the brain has evolved to focus attention on potentially dangerous items, such as the eyes of a predator. Even now, while you are reading this text, your attention is focused on the message conveyed by the text, ignoring the other elements surrounding it.

In artificial intelligence, this concept has led to the development of attention mechanisms. If you have a large amount of information, why not concentrate on the relevant information? In fact, this is an intuition that jumps out at you. For example, in magnetic resonance imaging, the image is square shaped with the axial slice of the brain in the center, implying a quantity of information associated with the background (irrelevant information). Following these premises, scientists have developed several attention mechanisms, where even convolutional or recurrent networks have been dispensed with.

The attention mechanism is easy to understand when we associate it with everyday human tasks, as indicated at the beginning of this section. However, from a mathematical point of view, it is necessary to have a little more rigor in the implementation. One of the simplest cases of models is predictions for a single variable. For example, suppose a data set x and y belonging to the reals. For each observation x, there is a single output y. In this sense, if one wishes to know a new output for a specific x, one could average all ys outputs to generate such a prediction (see Fig. 7a). The solution would result in a single value for any new observation and would not be very efficient. Instead, following the principle of attention models, one could pay more attention to outputs close to the queried value, i.e., assign weights to the output as a function of the distance to the queried point. The general form of the above can be expressed by Equation (*30*) [51], [52].

$$\hat{y} = \sum_{i=0}^{n} \alpha(x, x_i) y_i \tag{30}$$

Equation (30) is the most generalized form of the attention mechanisms, where x_i and y_i are called the key-value pairs of the n observations, x the query, and α the attention weights. This solution was proposed by Nadaraya-Watson, generally known as kernel regression, and the original version is shown in Equation (31).

$$\hat{y} = \left[\sum_{j=0}^{n} K(x - x_j)\right]^{-1} \sum_{i=0}^{n} K(x - x_i) y_i \tag{31}$$

Where K consists of a Gaussian kernel given by Equation (32).

$$K(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}} \tag{32}$$

A clear example of this implementation can be seen in Fig. 7a, where for data set varying from 0 to 7, values are a function of that set (blue dots). The observations are a partial description of the model that calculates a new observation or query value. For example, a query of 5.6 in the Nadaraya-Watson model with a Gaussian kernel would generate the distribution of weights displayed in the cyan curve. Values close to the query would have more weight in the weighted sum to predict the new value. In other words, attention is being paid to values close to the query [51].

The framework for AI attention mechanisms can be established using this same model, as illustrated in Fig. 7b. In this context, the network has keys and an attention score function that generates the values biased towards the query of interest. The values are subjected to an activation function f, which would establish the relevance or contribution of each value in the final output sum. Again, the scoring function would be related to the training parameters of the model, which would be adjusted to generate the distribution of weights that optimizes the desired task.

As expected, there is no single function of attention scores $a(q, k)$. However, the two most used functions are: additive attention and scaled product attention. The former is governed by Equation (33).

$$a(q, k) = W_v \tanh\left(W_q q + W_k k\right) \tag{33}$$

The scaled product attention is described by Equation (34) and is the central concept of the new networks called Transformers, presented in the next section.

$$a(q, k) = softmax\left(\frac{qk}{\sqrt{d}}\right) v \tag{34}$$

Being in Equations (33) and (34), q the query, k the keys, v the values and the Ws the training parameters [53].

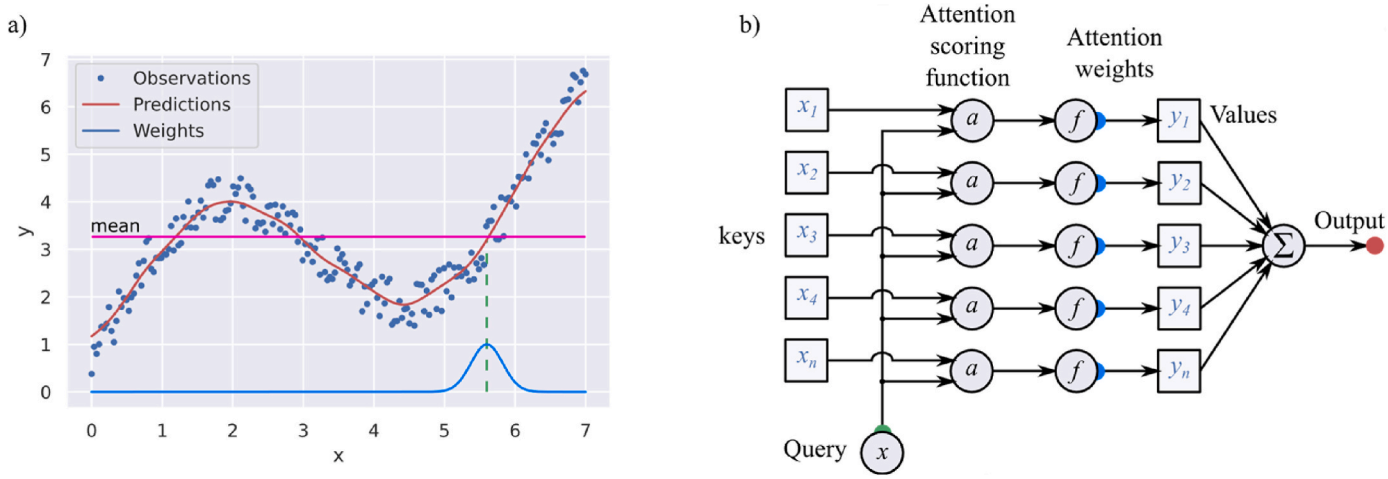


Fig. 7. a) Visualization of the weights in a Gaussian attention model for a query (x-value). b) Calculation of the output of the attention model based on the weighted average of values.

7. Transformers – attention is all YOU need

So far, fully connected neural networks, convolutional, recurrent, and attention models have been presented. However, as of 2017, with the famous article "Attention is all you need" published by Vaswani et al. [54], the way of thinking about attention models changed, giving rise to what is now known as Transformers. In essence, Transformers still retain the same intuitions of attention models but dispense with convolutional and recurrent networks.

At first, approximation of the Transformer can be somewhat intimidating due to the many elements that constitute it and all the mathematical models behind each element. In addition, the terminology used in this model makes it even more disturbing; however, once the step-by-step is understood, the Transformer is quite friendly and a powerful tool in artificial intelligence. Fig. 8 shows the general structure of the Transformer, consisting of an encoder and a decoder. The encoder, in turn, consists of several stacks with four sublayers, where the multi-head attention layer is the fundamental basis of the model.

Initially, the architecture receives an input which is usually a sequence of text. The text is divided into tokens (a process known as tokenization) and then represented as vectors to be used by the model. The process is known as embedding, and each token is represented through a vector of a size predefined by the developer, usually based on the maximum size that a token could have (see Fig. 8e). Subsequently, the embedding is passed through a positional encoding, the main difference with recurrent models. In the recurrent models, the order of the words is essential to perform the model's prediction since a word in one position or another can change the context of the sentence. However, the order is unnecessary for the Transformer case because the word position is being included through positional coding, avoiding generating and storing the hidden states of the recurrent networks. The positional encoding (PE) proposed by Vaswani et al. [54] is quite simple. It assigns sine or cosine values depending on the position (*pos* in Equations (35) and (36)) of the token and each element of the embedding vector. The generated values (see Fig. 8d) are added to each element of the embedding. The assignments with the sine function are made for the even elements (see Equation (35)) and the cosines for the odd ones (see Equation (36)). In addition, the argument of the trigonometric functions is regulated by an angular frequency (w_i) that depends on each element (i) of the embedded vectors of dimensions d_{model} (see Equation (37)).

$$PE_{(pos,2i)} = sin(w_i * pos) \tag{35}$$

$$PE_{(pos,2i+1)} = cos(w_i * pos) \tag{36}$$

$$w_i = \frac{1}{10000^{\frac{2i}{d_{model}}}} \tag{37}$$

The intuition behind this encoding is to preserve word order, i.e., the given assignment could be equivalent to a successive assignment of numbers. However, the assignment through trigonometric functions is more computationally efficient since it can take advantage of the decimal values generated by the float data. Additionally, it would allow the model to learn relative positions because a fixed displacement can be presented as a linear function of the positional encoding [54].

Having understood the positional encoding, one can now move on to "attention is all you need." The model presented above gives us a basis for how the Transformer works. Simplistically, self-attention generates the keys, queries, and values from the input data, creating for each input a corresponding weighted output generated from the outputs to the previous states of the signal (recurrence). The Transformer uses this same concept of self-attenuation, generating the keys, queries, and values from the input. However, the main difference is that it does not depend on the previous states because it is implicit in the positional encoding. Therefore, it is possible to use a single attention model or, alternatively, to use multiple attention models in parallel. The use of multiple models is known as multi-Head attention. In other words, for each input embedding vector, three weight matrices generate the keys, queries, and values in each header. For example, let be an embedding vector $x \in \mathbb{R}^d$, where d is the dimension of the model, then the queries, keys, and values are given by Equations (38)–(40).

$$Q_i = xW_i^q \quad W_i^q \in \mathbb{R}^{d \times d_q} \tag{38}$$

$$K_i = xW_i^k \quad W_i^k \in \mathbb{R}^{d \times d_k} \tag{39}$$

$$V_i = xW_i^v \quad W_i^v \in \mathbb{R}^{d \times d_v} \tag{40}$$

d_q, d_k, and d_v are the columns of the matrices for the i-th header, which have the same value.

The above is shown in Fig. 8b and corresponds to the outputs generated by the linear blocks. After this step, the process becomes a bit simpler, the scalar product between the query and key matrices ($Q_i \cdot K_i^T$) is performed, scaled (division by $\sqrt{d_k}$), passed through the softmax function, and finally, this value is multiplied with the vector of values in order to obtain the attention score, which is shown in Fig. 8c and expressed by Equation (41).

$$Head_i = softmax\left(\frac{Q_i \cdot K_i^T}{\sqrt{d_k}}\right) V_i \tag{41}$$

The process is repeated for each header and concatenated to generate the output of the first sublayer (see Fig. 8c). The operation is represented by Equation (42).

$$MultiHead = Concatenation(Head_1, Head_2, \ldots, Head_h) W^o \tag{42}$$

Where, $W^o \in \mathbb{R}^{d_k h \times d}$ is the matrix of the linear operation shown in Fig. 8b and h is the number of headers of each of the N stacks.

8. Concepts of artificial neural networks

The previous section showed the different types of artificial neural networks, all of which have some standard parameters called parameters and hyperparameters. The parameters are all the variables involved in the different models that can be learned through training, i.e., the parameters are the model's weights. Hyperparameters are different elements that can be changed but are not learned; they can be manually selected based on criteria specific to the problem at hand [55]. For example, the size of the observations (called batch) can be varied at one's own choice. However, a large batch size will require a higher memory capacity, and a small one requires a higher number of iterations to train the model, i.e., it would need more training time [56]. In the following, we address all the key elements related to artificial neural networks.

8.1. Loss function

In the previous section, we intuitively and mathematically described the behavior of artificial neural networks; however, we took for granted the value of the training parameters w. Each neural network can have hundreds, thousands, or millions of training parameters (depending on the depth), being necessary to search for the optimal values to reach the network's best performance. This process is best accomplished by determining a measure of model fitness. The loss function quantifies the distance between the actual and predicted values. In general, the loss is a positive number where smaller values generate better predictions or, failing that, achieve perfect prediction by reaching zero, provided the model is not overfitted. One of the most common loss functions in regression problems is the squared error or mean square error. Suppose that for the i-th observation, the actual value y_I is matched by the prediction $\hat{y}_I$, generating the squared error given by Equation (43).

$$l(\Theta)\frac{1}{2} = \left(\hat{y}_i(x, \Theta) - y_i\right)^2 \tag{43}$$

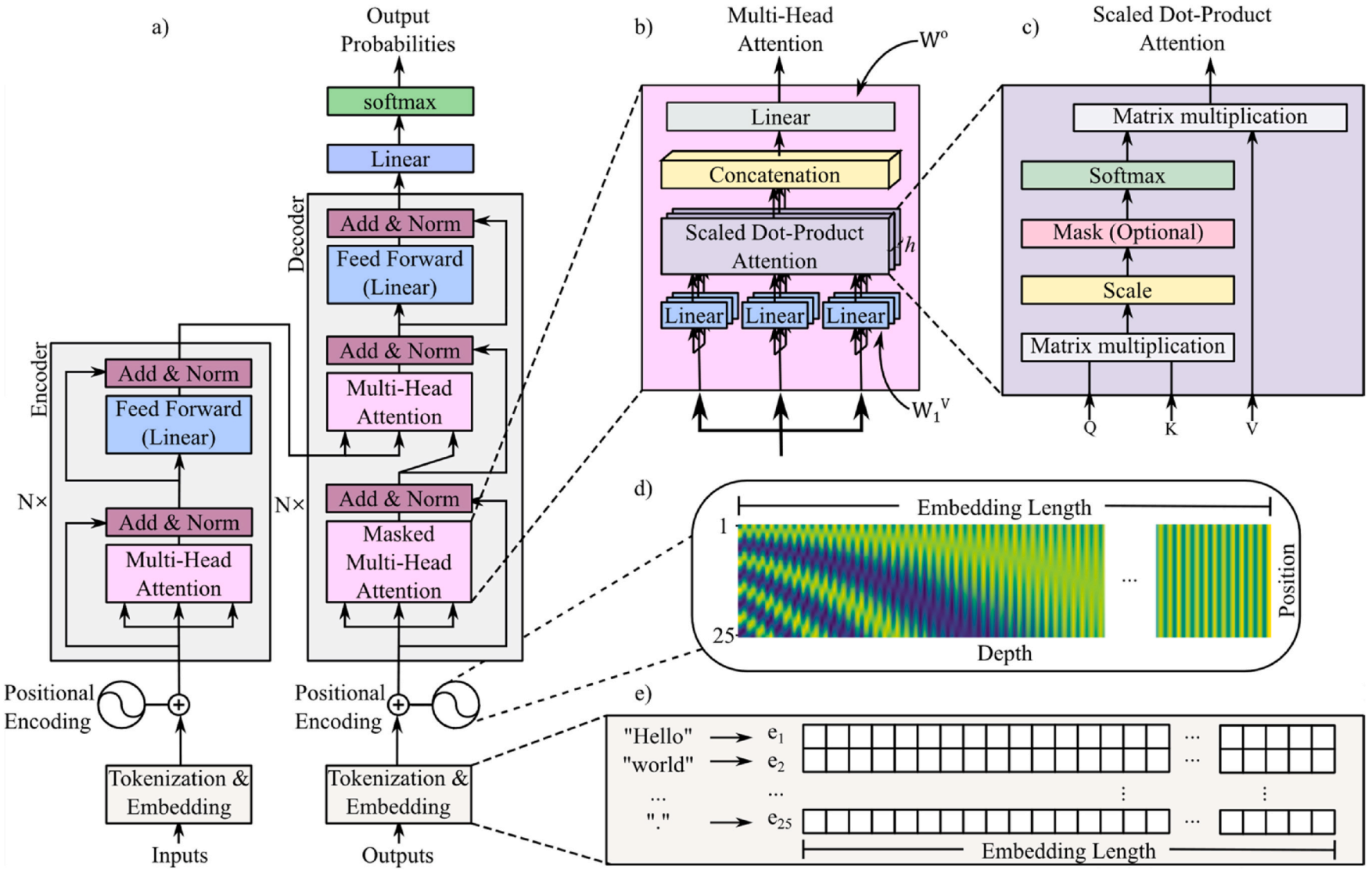


Fig. 8. a) General structure of the Transformer with its main elements. b) Multi-Head attention block. c) Scalar product based on the self-attention model. d) Positional coding generated for 25 inputs (words) and an embedding length of 512 features. e) Tokenization and embedding of the inputs.

Here, $\hat{y}_1$ is a function of the input x and the training parameters θ. Furthermore, assuming n observations, the error of the whole set is obtained by averaging the individual contributions, as shown in Equation (44).

$$J(\Theta) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{2}\left(\hat{y}_i(x,\Theta) - y_i\right)^2 \tag{44}$$

Although the quadratic error is one of the earliest loss functions, its use is more widely extended to regression models since many loss functions have more remarkable properties in segmentation or classification problems [57]. The following section shows five modern loss functions used on different types of applications in artificial intelligence.

8.2. Types of loss functions

The loss function is one of the fundamental elements to train the model efficiently. Therefore, the choice between one or another function could generate significant differences in the network's performance. For example, many applications focus on classification; however, using the mean square error would generate false performances if one element is higher than the others. The network would learn to classify all items like the one with the highest frequency and, if that item corresponds to 90% of data, the network will arrive at the same score even if all items are classified in a single class. Similarly, the segmentation task consists of classifying pixels into different elements, usually associated with the background and the object of interest (e.g., brain tumor). The difference between the regions spanned by the elements (data imbalance) usually causes the networks to be biased towards the more significant element. Therefore, it is necessary to select the loss function that considers such data imbalance [57]. In the following section, five of the most used loss functions are shown.

8.2.1. Binary cross-entropy

The binary cross-entropy is one of the most used loss functions in bimodal problems. Precisely, the function measures the difference between two probability distributions, calculating the entropy associated with each class or element. The principle can be applied to images, where each pixel is considered one of two distribution elements (e.g., background and tissue of interest) [58]. The function is highly efficient in training models. However, it is susceptible to class imbalance, so it is not recommended to use it in such cases. The binary cross-entropy (L_{BCE}) is mathematically defined as shown in Equation (45).

$$L_{BCE}(y,\hat{y}) = -\left(y\log(\hat{y}) + (1-y)\log(1-\hat{y})\right) \tag{45}$$

Where, y is the actual data set and$\hat{y}$ is the predicted set.

8.2.2. Weighted binary cross-entropy

As in the previous case, weighted binary cross-entropy is used to measure the difference between two distributions. However, these variant weights the ensembles, allowing the bias of data unbalance to be removed [59]. The weighted binary cross-entropy is defined mathematically, as shown in Equation (46).

$$L_{WBcE}(y,\hat{y}) = -\left(\beta y log(\hat{y}) + (1-y) log(1-\hat{y})\right) \tag{46}$$

Here, y is the actual data set, and$\hat{y}$ is the predicted set, and β is the weighting coefficient, used to adjust for false positives or false negatives.

8.2.3. Dice loss

Dice coefficient is a statistic used to calculate the similarity between two samples. Its use can be extended towards images by comparing the similarity between spatially matching pixels [60]. The coefficient has also been included as a loss function and is mathematically defined, as shown in Equation (47).

$$DL(y,\hat{y}) = 1 - \frac{2y\hat{y}+1}{y+\hat{y}+1} \tag{47}$$

Where, y is the actual data set and $\hat{y}$ is the predicted set. It should be noted that Equation (47) is modified with a 1 in the numerator and denominator, ensuring that the function is defined even in the extreme cases where y and $\hat{y}$ are equal to zero.

8.2.4. Tversky loss

The Tversky index is a measure of asymmetric similarity between sets [61]. This function can be viewed as a generalization of Dice's coefficient, expressed mathematically, as shown in Equation (48).

$$TI(y,\hat{y}) = \frac{y\hat{y}}{y\hat{y} + \beta(1-y)\hat{y} + (1-\beta)y(1-\hat{y})} \tag{48}$$

Equation (47) weights the false positives and false negatives weights through the coefficient β. Like the Dice coefficient, the Tversky index can also be fitted to a loss function, as shown in Equation (49) [62].

$$TL = 1 - TI \tag{49}$$

The loss function can be modified toward a focal loss by reducing the weights of individual examples and focusing the training on hard negatives through a modulation factor γ [63], as shown in Equation (50).

$$FTL = \sum_{C}(1 - TI_C)^{\gamma} \tag{50}$$

Here, the modulation factor must meet the condition of $\gamma > 0$.

8.2.5. Log-cosh dice loss

The Dice coefficient is widely used in computer vision on conventional images. However, due to its non-convex nature, the smoothed version using a hyperbolic log-cosine has recently been proposed [64]. The loss function is mathematically defined as in Equation (51).

$$L_{DL} = \log(\cosh(DL)) \tag{51}$$

Here, DL is the loss with the Dice coefficient set in Equation (46).

At this point, we have two fundamental elements in deep learning. The first one is the different network types and the loss functions that mainly establish the amount of error generated. In addition, each model that was described in the previous section was left as a function of training parameters known as weights and biases ($W's$ and $b's$), i.e., given any artificial neural network, there is a set of training parameters Θ that can be adjusted to optimize the desired task. So, the doubts are: what are the optimal values for my task, and how do I calculate them? Currently, an algorithm is used to obtain the training parameters; however, in this section, we will describe the analytical solution to the backpropagation algorithm to give a clearer intuition of the training of the networks.

8.3. Analytical solution

Suppose that for an artificial network, one has the predicted output and$\hat{y}_i$ associated with the i-th observation of the vector of inputs $x_i \in R^n$. Furthermore, assuming a mean square error as a loss function, the total error generated would be given by the contribution of each of the observations as given in Equation (52).

$$J(\boldsymbol{w}) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{2}\left(\hat{y}_i(\boldsymbol{x}_i,\boldsymbol{w}) - y_i\right)^2 \tag{52}$$

Now, from Equation (1), the general form of the output is known. However, if a linear activation function is assumed, the output would take Equation (53).

$$\hat{y}_i(\boldsymbol{x}_i,\boldsymbol{w}) = \boldsymbol{x}_i\boldsymbol{w} \tag{53}$$

Therefore, by arranging Equation (52) in a matrix form, one has the

expression of Equation (54).

$$J(\boldsymbol{w}) = \frac{1}{2n}(X\boldsymbol{w} - y)^T(X\boldsymbol{w} - y) \tag{54}$$

As mentioned above, Equation (54) provides the error generated by the model, i.e., the lower the value given by this function, the better the network's performance. In this context, we would be dealing with a classical optimization problem. Therefore, one could derive Equation (54) concerning w and equal to zero to find the minima of that function as given in Equation (55).

$$\frac{dJ(\boldsymbol{w})}{d\boldsymbol{w}} = \frac{1}{2n}\frac{d}{d\boldsymbol{w}}\left((X\boldsymbol{w} - \boldsymbol{y})^T(X\boldsymbol{w} - \boldsymbol{y})\right) = 0 \tag{55}$$

After a series of mathematical operations (see Appendix B), the solution given by expression (56) would be reached [65].

$$\boldsymbol{w} = \left(X^T X\right)^{-1} X^T \boldsymbol{y} \tag{56}$$

8.4. Solution by numerical methods (gradient descent)

Although the analytical method seems to be the fastest solution, this is usually not the case. The above solution is based on many assumptions and, although these can be met, finding the inverse of a matrix is not an easy process, especially in artificial neural networks, where there are many features and observations. However, there are more unique solutions, such as stochastic gradient descent (SGD). The intuition behind SGD is quite simple and consists of reducing the loss iteratively until a minimum is reached, as shown in expression (57).

$$\Theta_{\tau+1} = \Theta_\tau - \eta \nabla J(\Theta_\tau) \tag{57}$$

To understand in more detail, assume a loss function with a single training parameter w (See Fig. 9a). If a random value of w is initially selected, the loss function will probably have a high value. On the other hand, at that same selected point, the derivative would be calculated, which would indicate whether the function has a positive or negative slope, i.e., whether it is increasing or decreasing. Therefore, the initial value could be reduced by a fraction of the derivative to obtain a new value of w that generates a lower loss. The process would be repeated iteratively until the lowest possible value of the loss function is obtained, as shown in the general pseudocode in Fig. 9b. The process applies to multivariate problems where only the derivative would be exchanged for the partial derivative related to each training parameter [66].

In the case of a single variable, the loss function has a concave shape and a single minimum (global minimum). Therefore, the gradient can reach the lowest possible cost. However, it should be clarified that most loss functions do not have this behavior and, therefore, the gradient could reach a local minimum, limiting the model's performance.

8.5. Back propagation

Although the gradient descent is very useful for finding the optimal model parameters, the implementation still requires a large amount of computation to propagate the error through the model. However, this is significantly reduced through the backpropagation technique. Backpropagation refers to the method of calculating the gradient of the neural network parameters. In short, the method traverses the network in reverse order, from the output layer to the input layer, according to the chain rule of differential calculus. The algorithm stores the intermediate variables (partial derivatives) needed when calculating the gradient for the parameters. Due to the sequential nature of the neural networks, each layer can be expressed in terms of the previous layer, as shown in Equation (58).

$$\boldsymbol{y}^{(l)} = f^{(l)}\left(W^{(l)}\boldsymbol{y}^{(l-1)}\right) \tag{58}$$

In this sense, if Equation (57) is replaced in terms of the input and training parameters, the l-th output would be expressed as in Equation (59).

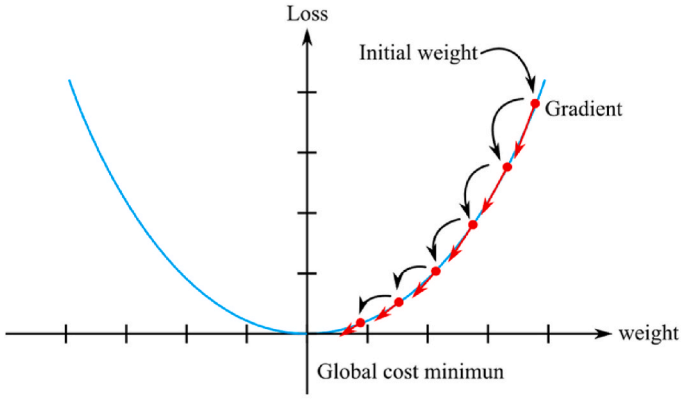
$$\boldsymbol{y}^{(l)} = f^{(l)}\left(W^{(l)}f^{(l-1)}\left(W^{(l-1)}f^{(l-2)}\left(W^{(l-2)}\ldots f^{(1)}\left(W^{(1)}\boldsymbol{x}\right)\ldots\right)\right)\right) \tag{59}$$

Equation (59) could be replaced over the loss function (see Equation (54)) to obtain the W'sweight matrices' gradient. For example, for a network with n layers, the mathematical equation will give the gradient concerning the weights of the second to the last layer (60). Where, this term is known as the error imputed to that layer [9].

$$\frac{\partial L}{\partial W_{n-2}} = \frac{1}{m}\sum_{i=1}^{m}\left(\hat{y}^{(i)} - y^{(i)}\right)\frac{\partial g_n}{\partial a_n}\cdot\frac{\partial g_{n-1}}{\partial a_{n-1}}\cdot\frac{\partial g_{n-2}}{\partial a_{n-2}}\cdot W_n \cdot W_{n-1}\cdot y_{n-3} \tag{60}$$

For practical purposes, Equation (59) indicates the amount of error associated with the weight matrices, facilitating the adjustment of the parameters through the gradient descent (see section 8.4) [9].

a)



b)

Algorithm 1: Gradient descent optimization

input: $J : R^n \rightarrow R$ a differentiable function
Θ_0 an initial solution
output: Θ_*, a local minimum of cost function J
begin
  $k \leftarrow 0$;
  while *improved* **and** $(\tau < \tau_{max})$ **do**
    $\Theta_{\tau+1} \leftarrow \Theta_\tau - \eta\nabla J(\Theta_\tau)$
    $k \leftarrow k + 1$
    if $J(\Theta_{\tau+1}) < J(\Theta_\tau)$ **then**
      improved $\leftarrow$ **True**
    else
      improved $\leftarrow$ **False**
    end
  end
  return Θ_k
end

Fig. 9. a) description of gradient descent for a single training parameter. b) general pseudocode used in gradient descent algorithm.

8.6. Activation function

As mentioned above, artificial neural networks emulate the biological neuron, and one of the key elements to replicate the operation is the activation function. This is usually constituted by a nonlinear relationship between the weighted input and the output of the neuron. The activation functions can have a different mathematical structure where the most common are the sigmoid, the hyperbolic tangent, the rectified linear unit (ReLu), and the softmax function. The mathematical structure of the functions is designed to "separate" values that are generally very close, allowing to generate a more differentiable space between values. For example, Fig. 10 shows the behavior of the hyperbolic tangent activation function, where the output presents a more considerable difference for the difference of the inputs.

It should be noted that the choice of the activation function is a hyper-parameter that affects the performance of the networks in different ways, i.e., the activation function may generate better results in some networks [67].

8.7. Pooling

In convolutional networks, small changes in the input image generate small changes in the feature maps. Therefore, pooling layers were devised to solve this problem, giving some transitional invariance to the models. Pooling operations are like convolutional layers, i.e., these layers generate a single value for a window that scrolls across the image (see Fig. 11). The window can have different sizes, stride, and pooling as the average (AveragePooling) or maximum value (max-Pooling). In addition to this, the pooling allows reducing the size of the feature maps, simplifying the model, and reducing the computational load [68,69].

8.8. Dropout regularization

A drawback present in artificial neural networks is overfitting or overtraining. When training a model with a limited number of data or over many epochs, it "memorizes" the data with which it is being trained. That is, the accuracy of the model during training may be excellent, but when tested with another data set, the accuracy drops drastically. So, regularization by abandonment was devised to solve this problem. The technique eliminates random neurons for each training batch, generating slightly different networks, where the model would adjust to these multiple network variations [70].

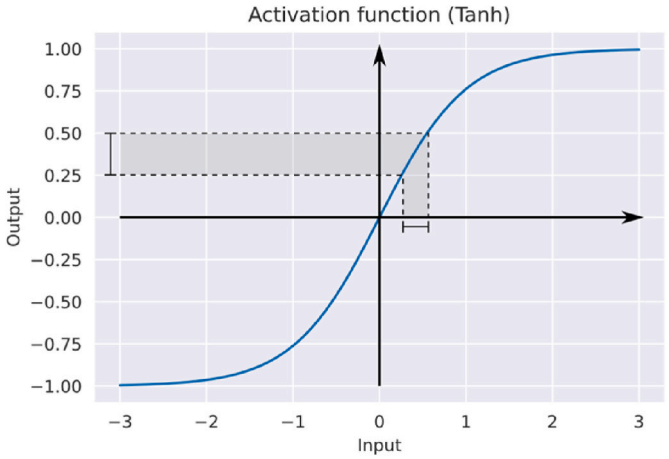


Fig. 10. Hyperbolic tangent activation function.

8.9. Batch normalization

Batch normalization was devised to mitigate the problem of changing internal co-variants, this being a limitation in the learning rate generated by the initialization of parameters and changes in the distribution of inputs to each layer. The change of co-variants is a shift in the internal distribution within each feature map. Normalization adjusts that shift by modifying the distribution toward a mean of 0 and a standard deviation of 1, using Equation (61). Subsequently, the normalization is adjusted through training to an optimal distribution by means of a linear transformation, as shown in equation (62). The parameters γ and β (learned parameters) modify the standard deviation and bias of the new distribution, improving the performance of DL models [71]. The normalization process also smooths the gradient flow and acts as a regularization layer [72]. Therefore, some architectures using batch normalization do not typically use dropout.

$$y_{Ni}^{'(k)} = \frac{y_i^{(k)} - \mu_B^{(k)}}{\sqrt{\left(\sigma_B^{(k)}\right)^2 + \varepsilon}} \tag{61}$$

$$y_{Ni}^{(k)} = \gamma \cdot y_{Ni}^{'(k)} + \beta \tag{62}$$

Here, $y_{Ni}^{'(k)}$ represents the normalized feature map of the k-th layer, $y_{Ni}^{(k)}$ is the optimal distribution for the same layer, $y_i^{(k)}$ the unnormalized input, $\mu_B^{(k)}$ and $\sigma_B^{(k)}$ represent the batch mean and variance respectively and ε is a stabilization coefficient used to avoid division by zero.

8.10. Evaluation metrics

Several metrics have been designed to quantitatively describe network behavior as an essential part of an objective validation of network performance. The metrics compare the results obtained with the actual results, generating scores proportional to the effectiveness of each model. Generally, scores can range from 0 to 100%, or their fractional form from 0 to 1, where 0 indicates zero performance and 1 or 100% indicates perfect performance. Like loss functions, there are many evaluation metrics, and even some metrics are used as loss functions, but complementary version towards 1, i.e., of the form (1-metric). In Table 1, we show the most used evaluation functions for the case of binary problems.

Most metrics are defined in terms of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN), as illustrated in Fig. 12a.

All metrics range from 0 to 1 except for the Hausdorff distance (HD), ranging from 0 to ∞. The metric takes the immense value between the farthest distance from one curve to the nearest point of the other and vice versa (see Equation (74)). Fig. 12b illustrates this principle between two curves. In this context, the coefficient tends to zero when the curves are entirely equal and increase as the curves differ [79].

9. Building of modern blocks

As deeper networks are designed, it becomes imperative to understand how adding layers can increase the complexity and generalizability of the network. To understand the complexity of modern networks, in the next section, we give an intuitive description of networks and the mathematical model that governs the model's behavior.

9.1. AlexNet

Although the implementation of convolutional neural networks became known from developments by LeCun et al., it was not until the AlexNet network won the "ImageNet Large Scale Visual Recognition Challenge 2012″ that the computer vision paradigm was changed. The network demonstrated that features obtained through deep learning

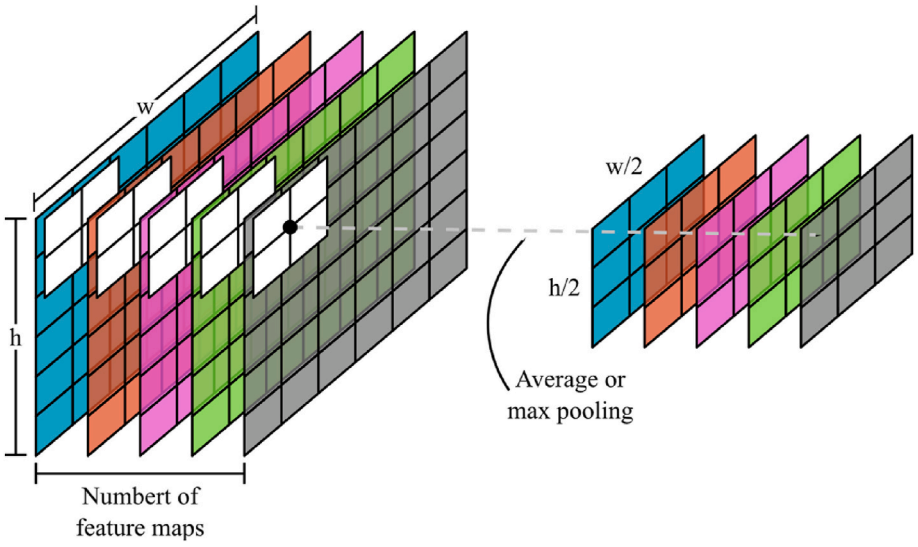


Fig. 11. Pooling for five feature maps reduced by half.

Table 1
Disclosure metrics most used in deep learning.

Name	Equation	
Accuracy [73]	$ACC = \frac{TP + TN}{TP + TN + FP + FN}$	(63)
F1 score [74]	$F_1 = \frac{2TP}{2TP + FP + FN}$	(64)
Sensitivity or Recall [74,75]	$SE = \frac{TP}{TP + FN}$	(65)
Specificity [73,75]	$SP = \frac{TN}{TN + FP}$	(66)
Precision or positive predictive value [74,75]	$PR = PPV = \frac{TP}{TP + FP}$	(67)
Negative predictive value [75]	$NPV = \frac{TN}{TN + FN}$	(68)
False positives rate [73]	$FPR = \frac{FP}{FP + TN} = 1 - SP$	(69)
Area under the ROC Curve [74]	$AUC = AUC(SE, FPR) = 1 - \frac{SE + FPR}{2}$	(70)
Conformity [76]	$CF = 1 - \frac{incorrectly\ classified\ voxels}{TP}$	(71)
Jaccard index or the Intersection over Union [77]	$JD = IoU = \frac{TP}{TP + FP + FN}$	(72)
Dice coefficient [60]	$DSC = \frac{2TP}{2TP + FP + FN} = \frac{2JD}{1 + JD}$	(73)
Hausdorff Distance [78]	$HD = \max\{h(A,B), h(B,A)\}$	(74)
	$h(A,B) = \max_{a \in A} \min_{b \in B} a - b$	(75)

could outperform manually designed features [80]. The network does not differ much from the network developed by LeCun, which is composed of convolutional layers, pooling layers, and fully connected layers (see Fig. 13). Eight layers mainly constitute AlexNet, five convolutional and three fully connected (or MLP) layers governed by the mathematical models described in the previous sections [80].

The input image advances through the different convolutional layers generating a more significant number of feature maps but smaller due to the pooling layers. The output of the last convolutional layer is flattened and passed to the fully connected layers. The flattening process consists of taking the feature matrices and converting them into vectors. Finally, the last part of the network takes the abstract feature vectors to propagate them through the perceptrons, generating the prediction values associated with each class.

9.2. VGG19

The VGG19 network is like the AlexNet architecture, with sequential convolutional layers with increasing filters as you go deeper into the network. The model has 16 convolutional layers, three fully connected, and five pooling layers based on the maximum pooling method with 2 × 2 windows (see Fig. 14). The architecture motivated the use of smaller filters since the perceptual field was shown to be just as efficient as with larger filters. In addition, the smaller filter size also reduces the number of training parameters [81].

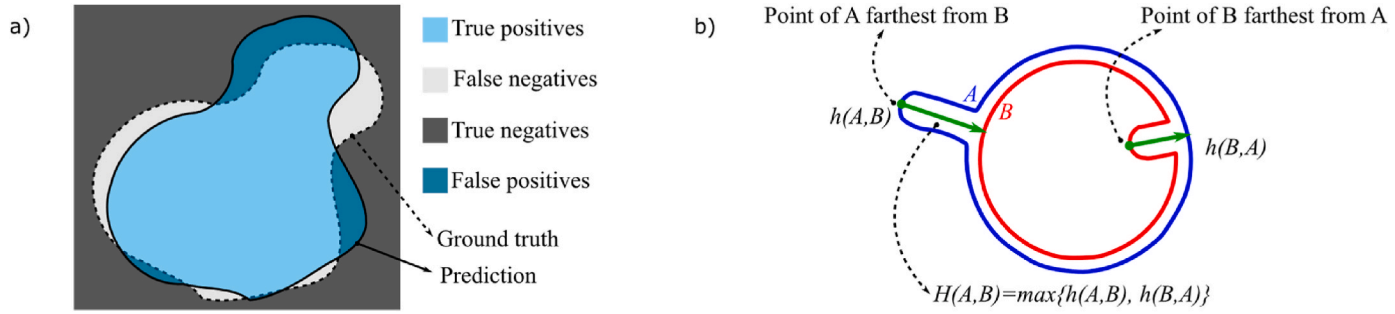


Fig. 12. a) illustrates a segmentation problem with all the elements that compose the overlap between the actual and predicted segmentation. b) Example of the calculation of the Hausdorff distance for two segmentation product curves.

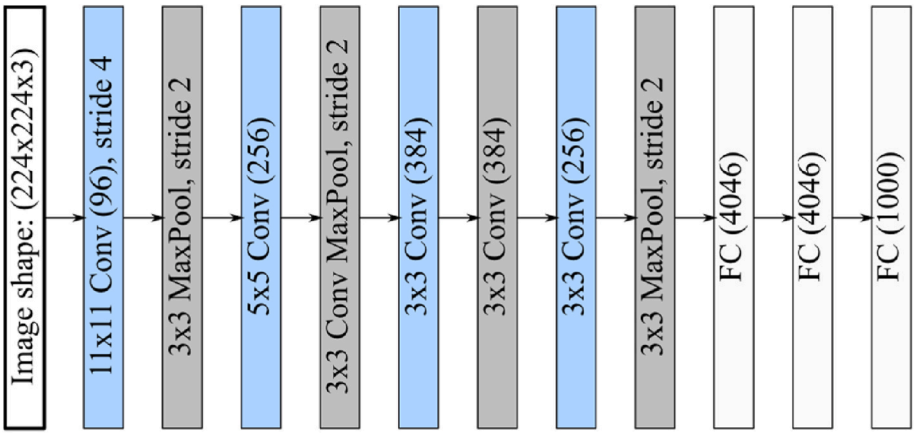


Fig. 13. AlexNet convolutional neural network architecture.

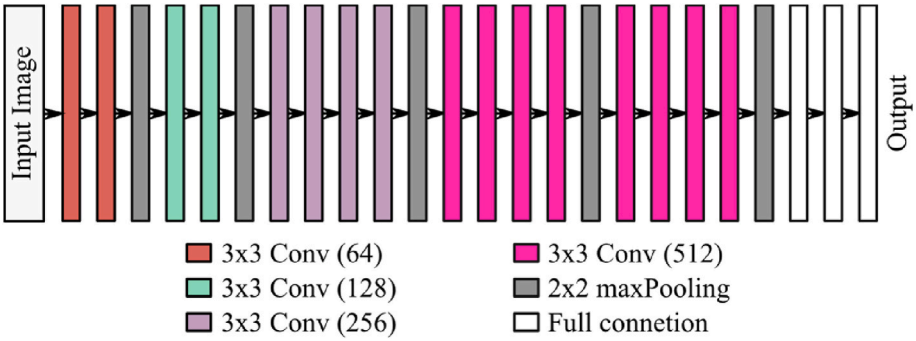


Fig. 14. General structure of the VGG19 network.

9.3. GoogleLeNet (inception)

Most designs are based on sequentially stacking layer after layer, hoping to achieve better performance by extracting a more significant number of features. However, designing in this way presents two main problems. The first one is that too many training parameters will lead to an overfitting of the model and the second drawback is that the model also becomes difficult to train because it depends on the back-propagation algorithm, i.e., it is dependent on the derivative, which is reduced with many convolutional layers. In this sense, to solve this drawback, Szegedy et al. proposed a structure that combines network-in-network (NiN) blocks and repeated block pooling [82,83]. The blocks are called Inception and consist of four parallel convolutional trajectories, as illustrated in Fig. 15b. The first trajectory has only a 1×1 convolutional layer. The second and third trajectories have 1×1 convolutions followed by 3×3 and 5×5, respectively. The fourth trajectory consists of a top pooling layer (MaxPooling) and a 1×1 convolution to finally concatenate with all the trajectory outputs.

The googleLeNet architecture has a series of convolutional layers, like the LeNet structure. Subsequently, the sequence is connected with nine Inception blocks to generate the estimates (see Fig. 15b).

9.4. ResNet

In developing deep learning architectures, it is clear that a more significant number of layers allows more abstract features to be extracted and, consequently, increasingly complex problems to be addressed. However, depth limits network training due to gradient fading, i.e., since the gradient propagates backward between different layers, repeated multiplication over many layers makes the gradient too small. To solve this problem, He et al. introduced the concept of residual connection or direct connection (see Fig. 16c) [84]. Essentially, the connection creates trajectories parallel to the convolutional layer sequences, allowing the gradient to flow through the lattice and preventing it from vanishing. Furthermore, the connection forces the network to learn the residual mapping $f(x) - x$, being easier to train if the ideal residual mapping is the identity function $f(x) = x$ (see Fig. 16c) [85].

The use of residual connections allowed for deeper networks. For example, one of the most widely used architectures is the ResNet50, consisting of a sequence of 50 convolutional layers as illustrated in Fig. 16b. The design interleaves 1x1, 3x3, and 1x1 sized filters in parallel with the residual connection.

9.5. DenseNet

If a direct connection between a sequence of layers prevents the gradient from fading and the network from training faster, a suitable assumption would be to expect multiple connections to generate better results. This is the assumption devised by Huang et al. that led toward the design of densely connected networks [86]. However, although a direct connection is the main idea of the DenseNet, the difference with ResNets is that the connection is not made by summing the feature maps but through concatenation.

Fig. 17 shows the design of the DenseNet 121 network constructed with four densely connected blocks, as shown in Equation (76). In turn, each block is made up of 1x1 and 3x3 filter size convolutional layers, where the output is concatenated with the previous input. Additionally, the network uses transition layers to convolve all feature maps and reduce their resolution employing an average pooling layer [87].

$$Y_{c_6}^{(1)} = concatenation\left(Y_{max}, Y_2^{(1)}, Y_4^{(1)}\left(Y_{c_1}^{(1)}\right), Y_6^{(1)}\left(Y_{c_2}^{(1)}\right), \ldots, Y_{12}^{(1)}\left(Y_{c_5}^{(1)}\right)\right) \tag{76}$$

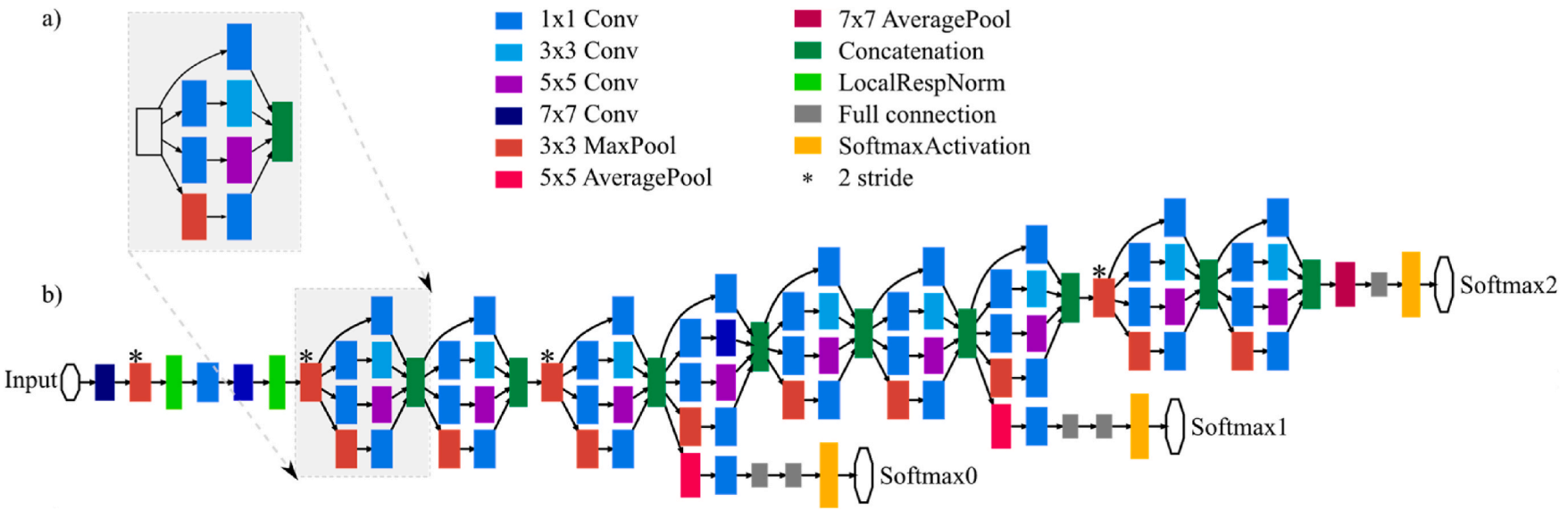


Fig. 15. a) Inception block formed by four convolutional trajectories for the same input. b) General structure of the GoogleLeNet network with all the elements. Image based on [83].

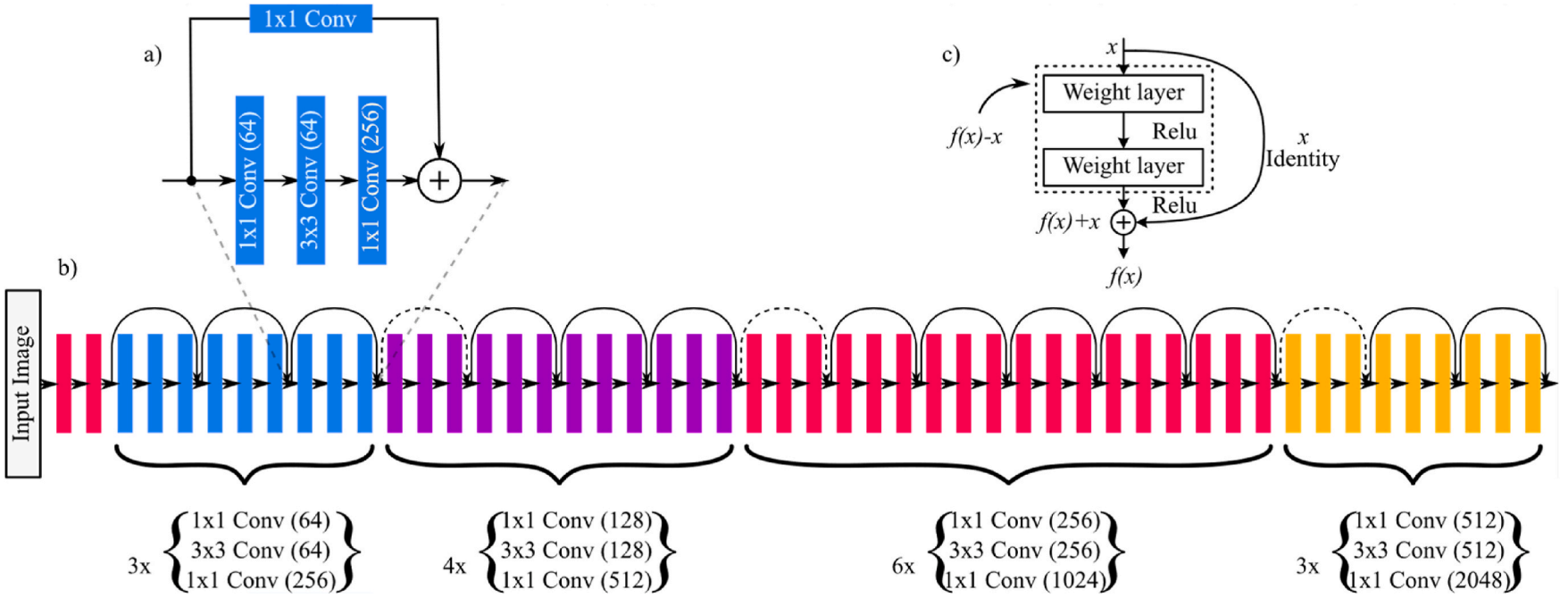


Fig. 16. a) residual block with a convolutional layer in the residual connection. b) ResNet50 architecture. The direct residual connection has taken direct from Ref. [84]. c) residual connection or direct connection.

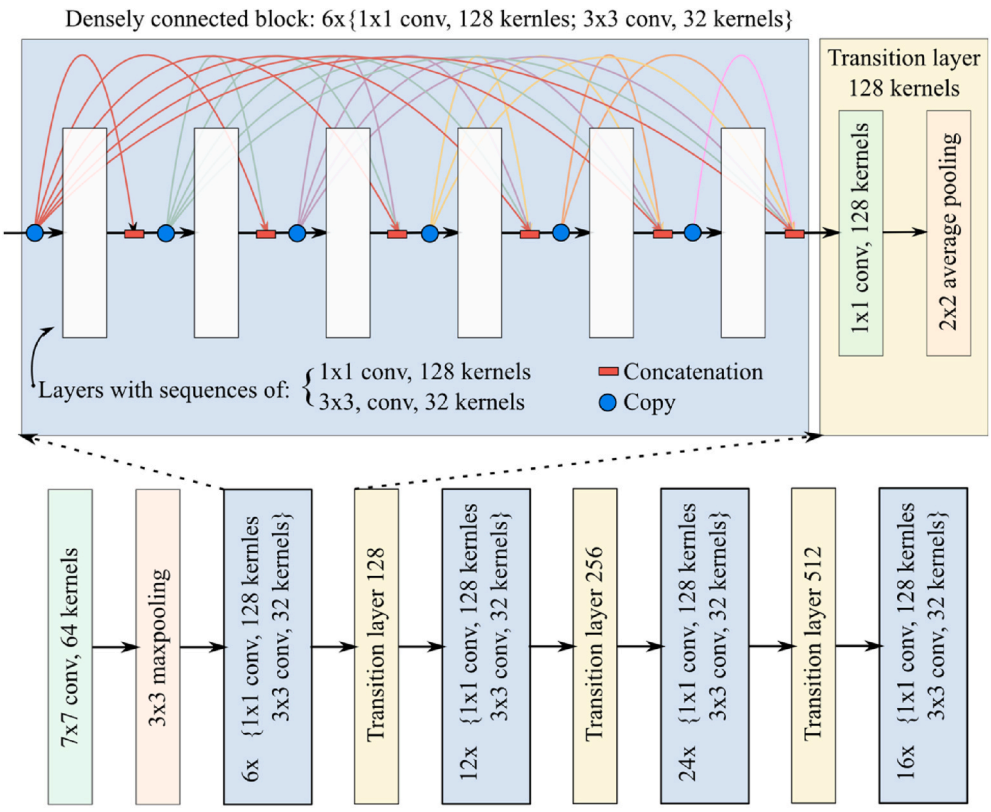


Fig. 17. Densely connected convolutional neural network.

9.6. *EfficientNet*

The EfficientNet is still a convolutional network with stacked layers in sequence but implemented with a uniform scaling method. That is, while most architectures are designed with arbitrary dimensions, the EfficientNet uniformly scales the depth, width, and resolution dimensions (see Fig. 18). The intuition behind the method is that if the input image is large, then the network should have more layers to increase the receptive fields and more channels to capture the small patterns in the image [88].

9.7. *U-net*

Most networks described so far are layered or block convolutional designs in sequence. Generally, the networks increase the number of features they extract through a more significant number of convolutional filters. The generated features are flattened and passed through fully connected layers to discriminate between the classes associated with each image. The classification task is one of the most used tasks in medical imaging. However, another common task is the segmentation of regions of interest, such as lesions, tumor tissue, or any type of anomaly. Fortunately, for segmentation, there is the U-Net network designed by Ronneberger et al. [89]. The design is the fundamental basis of most segmentation networks, and this is primarily due to the ingenious design, which allows preserving the spatial distribution of the image while abstracting image features. The U-Net network consists of two main elements: an encoder and a decoder. The encoder takes the input image and convolves it, generating increasingly complex feature maps as it moves deeper into the network. Additionally, the convolutional layers are combined with pooling layers to reduce the size of the maps and thereby reduce the computational load. The process is interleaved between convolutions and pooling, as illustrated in Fig. 19.

At the encoder end, the network generates many features but of

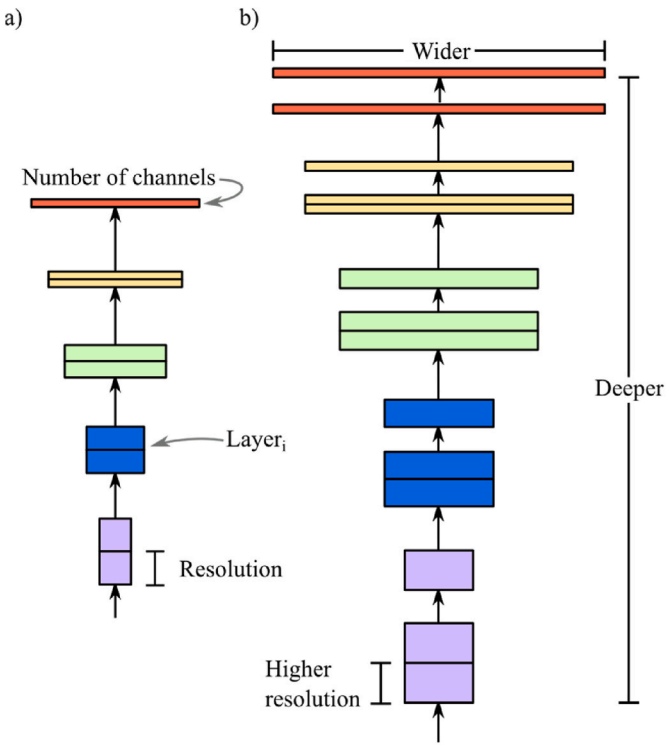


Fig. 18. Representation of the scaling method. a) reference example, and b) network scaled in depth, width, and resolution. The image was taken from Ref. [88].

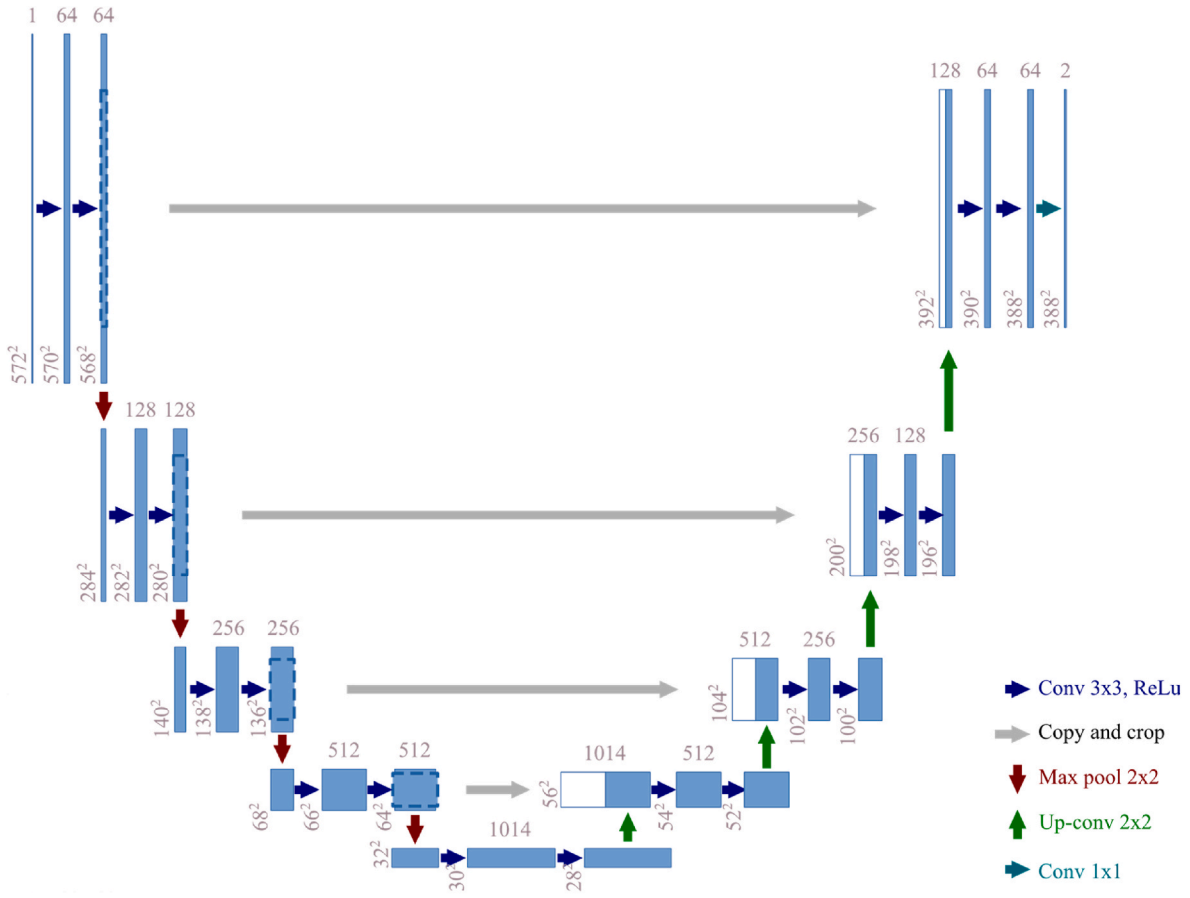


Fig. 19. General architecture of the U-Net network. The figure is the original graph taken from Ref. [89]. Each box represents a set of feature maps. The numbers at the bottom represent the feature map sizes, and the numbers at the top represent the number of maps.

reduced size, which is used to generate the segmentation image. At the decoder, the features must be convolved into maps twice the size. However, in discrete space, the convolution reduces the image dimensions. That is, suppose one has an image of size $n\times n$ (with n even) and convolves it with a filter of size 2x2 with strides of 2. The result would generate an output with half the size as given in Equation (77).

$$Out_{\frac{n}{2}\times\frac{n}{2}} = K_{2\times 2} * In_{n\times n} \tag{77}$$

The convolution of Equation (16) can be represented as a matrix operation, converting the matrices to vectors and the filter to the sparse version or matrix. The equivalent is shown in Equation (78).

$$Out_{1\times\left(\frac{n}{2}\right)^2} = In_{1\times n^2} K_{n^2\times\left(\frac{n}{2}\right)^2} \tag{78}$$

In this sense, it is possible to multiply by the inverse of the sparse matrix on both sides of the equation, generating the result of Equation (79).

$$Out_{1\times\left(\frac{n}{2}\right)^2} K^{-1}_{\left(\frac{n}{2}\right)^2\times n^2} = In_{1\times n^2} \tag{79}$$

If the arguments of this result are reversed, that is, if the input image is put on the left side and the output image on the right side, then you will have an operation equivalent to a convolution but increasing the dimensions of the image as initially sought. The process is known as transposed convolution or denoted as Up-conv (see Fig. 19), referring to an upward convolution. It should be noted that, although Equation (77) denotes the inverse of a non-square matrix, the process is not performed since the parameters that make up the filter are not known, i.e., the filter could be replaced by a matrix with the dimensions transposed to the original size and with unknown weights, without having significant repercussions since these would be calculated during training.

After transposed convolution, the result is concatenated with copies of the maps before the pooling layers and resubjected to convolutional layers, as illustrated in Fig. 19. The process is repeated the same number of times as they were clustered through the max pooling function.

The U-Net is highly efficient in the segmentation process; moreover, the convolutional layers can be replaced by the blocks of the networks described above. In this sense, when a network block replaces a layer, the architecture is said to have a backbone of that network. For example, the backbone of the DenseNet network could be used.

10. Medical imaging and its applications

The field of medicine is one of the fastest-growing areas and one of the most rigorous, as it is directly related to the life and quality of patients. In this sense, tools have been developed over time to facilitate the work of physicians. Developments initially began with small mechanical tools such as scalpels, syringes, and even stethoscopes. However, constant research and technological evolution have introduced such novel tools as medical imaging. In definition, medical imaging is the set of techniques and processes to create images of the human anatomy or its functioning. In this sense, today, there are many strategies to generate such images. Clear examples of this are: ultrasound, radiography, computed axial tomography, magnetic resonance imaging (both functional and structural), positron emission tomography, endoscopy, thermography, external imaging (e.g., of melanomas), microscopy, and even, in some cases, recording methods are considered to be images since they can produce data that can be represented as information maps (e.g., electroencephalography).

Although the concept behind medical imaging is simple (describing human anatomy), the implementation and interpretation are not. Each technique obeys a physical principle and, therefore, an expert professional is required to interpret the results, or the images generated. However, in the tasks of classification, segmentation, prediction, and other tasks covered by deep learning, it is only necessary to have clearly identified labels, either of the pixels that make up the image or the image in general. In other words, with deep learning techniques, it is not necessary to know in detail the background of the images -although knowing it could help to select the most appropriate architecture or give some clues on how to build the new models-. In principle, as discussed in the previous sections, DL algorithms automatically take the image and extract features until the desired task is reached. In this sense, current DL research is focused on investigating, developing, implementing, and evaluating the performance of networks in different areas, images, and tasks.

In order to understand the trends in deep learning on medical imaging, the most recent research articles up to the second half of the year 2021 are listed in Table 2. The search was based on two keywords: 'deep learning' and 'cancer.' The focus was taken towards cancer to narrow the search to a smaller number of research papers, and only articles that used medical imaging in their development were selected.

The Table includes the name of the principal author with the year of publication, and the topic addressed from the deep learning approach, the architecture used, the datasets, the main results with the respective evaluation metrics (The most common metrics are defined in Section 8.10), and a section of observations, where the authors' limitations or suggestions for future work are highlighted. It should be clarified those only observations directly related to deep learning were taken, i.e., protocols or experimental design parameters related to data acquisition were not included.

The results show high variability among the topics addressed by artificial intelligence. In essence, every type of cancer that can be recorded through a digital image has been subjected to the scrutiny of artificial intelligence to perform a task that facilitates or expedites the radiologist's work in charge. Similarly, the network designs presented great variety. Although the U-Net remains the base structure for the segmentation task, it has been implemented under new blocks or concepts that enhance the network capacity. For example, most U-Net networks were implemented with another type of backbone, with the ResNets family being the most widely used. In contrast, it is necessary to highlight that, although attention models and Transformers are one of the most probed topics in recent years, convolutional neural networks are still the first choice to work with medical images. In fact, in Table 2, only Lal et al. [94] included blocks with attention mechanisms in their research.

On the other hand, a trend is also seen in the type of images. Although most of the developments were based on computed tomography (CT) images, there were also, to a lesser extent, works based on magnetic resonance imaging, ultrasound, and biopsies. The images allowed addressing cancer-related problems in the prostate, cervix, carcinomas, liver, rectum, colon, gastric, cervical lymphadenopathy, gliomas, breast (including microcalcifications), lung, pancreas, and liver. In addition, the work not only focused on cancer detection but there were also developments in the segmentation of organs at risk for treatment with radiotherapy. The latter is vital to determine the exact limits of the organs, allowing precise planning of the radiation dose required in cancer treatment.

In the case of architectures, the versatility in network design has opened a universe of possibilities that is still uncertain for most scientists. The dependence on data and the constant evolution of image acquisition systems further complicates the convergence towards definitive AI systems. In particular, the architectures in Table 2 show a wide range of designs. Developments range from the classical base U-Net to complex blocks with attention mechanisms. For example, only in these investigations were found networks with residual connections (ResNet), densely connected (DenseNet), recurrent (LSTM or GRU), Pyramid Scene Parsing Network (PSPNet), two-dimensional, three-dimensional, or composite models or models such as Mark R-CNN, VGG, Xception, and DeepLab+.

As discussed in section 8.10, it is necessary to resort to evaluation metrics for an objective evaluation of network performance. In particular, such metrics describe the performance of the networks against their specific task. However, the availability of different metrics, such as accuracy or F1 score, does not clearly compare different investigations. Although the review found that most of the studies are based on the Dice coefficient metric (for segmentation), it was also possible to find metrics such as the Jaccard index, Hausdorff, or even not standard metrics such as Matthew's correlation coefficient. It is therefore difficult to create a clear comparison between different investigations focused on the same problem. Despite these drawbacks, the results are encouraging, even reaching perfect values of 100%, as in the case of Adweb et al. [114] and Iqbal et al. [113].

Finally, it should be clarified that most investigations did not use transfer learning except for Salvi [92], Thomas [93], Zhao [96], Urushibara [97], Gonzalez [101], Naser [106], and Zhuang [110]. Similarly, less than half of the articles implemented data augmentation, and even most of these focused on random rotations, leaving aside all other existing methods.

11. Perspectives and future expectations

The boom in artificial intelligence and especially in deep learning systems, is evident. The number of publications that emerge month after month is clear evidence of this area's importance in the scientific field. Moreover, the multidisciplinarity of DL has allowed it to compete with human performance even in areas as complex as medicine. However, while most authors highlight the advantages and findings of deep learning, few systems have had validity in the actual clinical setting. The limitations surrounding medical imaging can be pretty extensive and depend on the particular problem. Nevertheless, although there are many limitations in DL, it is possible to highlight some common factors that restrict the performance of the models. First, it is evident that the networks that generalized well without overfitting was trained with many data, i.e., the amount of data is still a fundamental factor in the training of networks. On the other hand, while there is currently a large amount of medical data and although there are also organizations in charge of collecting databases, no data covers all the heterogeneity of acquisition protocols or variation between study subjects. For example, in the case of MRI alone, images can be acquired with a T1-or T2-weighted sequence. Moreover, the resolution can vary significantly from one resonator to another depending on the magnetic field used by the equipment, which can be 1.5, 3, or even 7 T. Additionally, the constant evolution of image acquisition systems limits the construction of extensive databases with similar characteristics, further deepening the problem of the availability of data that fit the actual context.

In this order of ideas, it is expected that new developments will be focused on more robust networks that need a smaller and smaller amount of data for training. In fact, these approaches are already considered in current networks, as is the case of residual connections, which allow better training of networks with greater depth. In this sense, ideally, future approaches may be oriented to outperform current networks by making training more efficient and reducing the number of reference images for optimal adjustment of network parameters.

It is more than clear that advances in computer technologies and the creation of new repositories encourage sharing medical images. It is even possible to find international platforms with multi-center data. However, the voracity and consumption of data in deep learning exceeds the availability of data and limits the evolution of deep learning. Additionally, the technological evolution from medical research and practice makes the databases have obsolete images, being necessary a faster response for the publication and free distribution of next-

Table 2
Recent research in deep learning and cancer.

Main author	Topic	Architecture	Data sets (D), results (R) and Observation (O)
Mohammadi et al., 2021 [90],[a]	Segmentation of organs at risk (bladder, rectum, and sigmoid).	ResU-Net	D Imaging of 113 patients with locally advanced cervical cancer. CT R DSC 95,7 ± 3,7, 96,6 ± 1,5 and 92,2 ± 3,3%. HD 4.05 ± 5.17, 1.96 ± 2.19 and 3.15 ± 2.03 mm. Average Symmetric Surface Distance (ASSD) 1,04 ± 0,97, 0,45 ± 0,09 and 0,79 ± 0,25 for the bladder, rectum, and sigmoid, respectively. O Uncertainty exists between the boundaries of the OARs contours and the training set. The set is based on data from a single center. Diversity of cancer type, differences between teams, non-homogeneous image acquisition protocols make an objective comparison of results difficult. Despite the promising results delivered by the metrics, the authors recommend a qualitative evaluation by experienced radiation oncologists.
Nemoto et al., 2020 [91]	Semantic segmentation of organs at risk for prostate cancer radiotherapy.	U-Net 2D	D 556 CT images. R Average DSC 0,85 ± 0,05, 0,94 ± 0,04 y 0,85 ± 0,07 for prostate, bladder, and rectum, respectively. O The clinical setting is not constant; therefore, each new advance requires a large amount of new data.
Salvi et al., 2021 [92]	Segmentation of prostate glands.	Rapid IdentificatioN of Glandural Structures (RINGS). UNet-based model with ResNet34 backbone.	D Prostate biopsy of 150 male patients. R Balanced ACC (0.9325 ± 0.0684), PR (0.8897 ± 0.1359), SE (0.9356 ± 0.0964) and DSC (0.9016 ± 0.1087). O Uses data from a single center.
Thomas et al., 2021 [93]	Semantic segmentation of the different parts of carcinomas and classification.	U-Net with the ResNet50 backBone.	D 290 samples of BCC (140), SCC (60) and IEC (90). R Segmentation: ACC 85% and 74% (validation and test). Classification: ACC 97.9% (test). O In surgical margin clearance, the technique depends on the quality of segmentation. The limitation could be overcome by adding additional information in the deep margin regions, which is only found in few training datasets.
Lal et al., 2021 [94],[a]	Segmentation of histopathological images in liver cancer.	NucleiSegNet: Consisting of residual, bottleneck, and decoder blocks with attention.	D 124 histopathological images stained with H&E. R Results for two data sets named by the author as KMC and Kurman. KMC F1 (83,59) and IoU (72.06%). Kumar F1 (81.363) and IoU (68.883%). O Extension of this method to the segmentation of multiple tissue instances is a possible future work. Obtaining more imaging samples and improved pre-probing techniques will be critical for future work.
Choi et al., 2020 [95],[a]	Segmentation of multiple organs in breast cancer.	DenseNet 3D.	D 62 breast cancer patients. CT. R DSC 0.86 (overall chamber). O A single oncologist delineated the contours. The authors recommend involving multiple experts in future developments.
Zhao et al., 2020 [96]	Detection and segmentation of lymph nodes in rectal cancer.	Mask R-CNN with Resnet-101 backbone.	D 5789 LN from 293 patients with rectal cancer. R Detection: SE (80.0), PPV (73.5) and FP/vol (8.6%) in internal testing. Segmentation: DSC from 0,81 to 0,82. O Small lymph node size affects detection and segmentation. Therefore, the investigation focused on nodes >3 mm. The set is small and was performed by few experts. Therefore, manual delineation may have biases. It is recommended to integrate a group of experienced radiologists to generate more heterogeneous databases.
Urushibara et al., 2021 [97]	Diagnosis of cervical cancer.	Xception with transfer learning from ImageNet.	D 418 T2 MRI. 177 subjects with cervical cancer and 241 healthy subjects. R SE (0.883) SP (0.933) ACC (0.908) and AUC (0.932). O Transfer Learning was performed from ImageNet implementations, composed of natural images. The transfer may not be adequate as different types of images are used. The study was based on data from a single center. The authors recommend validating with images from other institutions. The development was performed only on sagittal T2-weighted MRI. However, there are many sequences and planes of visualization. In addition, we switched from DICOM to lower-quality JPEGs, which may affect the diagnosis.
Chen et al., 2021 [98]	Segmentation of organs at risk for radiotherapy.	Ua-Net (head and neck), 2.5D U-Net (thorax) and 3D U-Net (abdomen and pelvis).	D 755 CT of head and neck, thorax, abdomen, and pelvis R Average DSCs of 0.84 and 0.81 on in-house and public datasets. HD 6,39 ± 6.14. O Imaging of irrigated organs without any tumor invaded or resected by surgery was used. In addition, the study was performed by a single expert.
Rigaud et al., 2021 [99],[a]	Anatomical segmentation of the cervix in cervical cancer.	DeepLabV3 + (Google 2D) and U-Net 3D.	D 2 datasets. 408 CT. R Average DSC 0.85. Range from 0,77 to 0,90. O
Courot et al., 2021 [100]	Segmentation of cervical lymphadenopathy.	U-Net	D 117 CT R Average DSC 0.63. O

(*continued on next page*)

Table 2 (*continued*)

Main author	Topic	Architecture	Data sets (D), results (R) and Observation (O)
			Limited data with partial annotations. Size variability affects the segmentation process and validation metrics, i.e., small items result in low Dice scores.
Gonzalez et al., 2021 [101],[a]	Segmentation of sigmoid colon for cervical radiotherapy.	U-Net with 2D and 3D filters.	D Approximately 2000 slices of 50 CT. R Average DSC 0.88. O Sigmoid colon segmentation is complex even for humans. Therefore, segmentations may vary from observer to observer and generate variations in scores concerning those observers. Small amount of data.
Zhang et al., 2021 [102]	Segmentation of pancreas.	Joint network with U-Net 3D and 2D.	D Three CT data sets. 36 from ISICDM, 82 NIH, and 281 MSD R DSC 84.47+-4.36%. O
Caballo et al., 2020 [103]	Segmentation of breast masses.	U-Net	D 93 mass-like lesions from 69 CT images. R DSC (0.93 ± 0.03), SE (0.92 ± 0.03), PR (0,93 ± 0,05) and CF (0,85 ± 0,06). O Data augmentation was done through GAN with the same images as used for U-Net. This could generate biases due to synthetic images. The authors suggest augmenting the databases for future work.
Balagopal et al., 2021 [104]	Segment precise contouring of the clinical target volume.	U-Net, PSPNEt and DeepLab	D 340 postoperative prostate cancer patients. CT R DSC 0.87. O Only images without additional dice were used. Therefore, additional information on pathologic conditions would be expected to improve model performance.
Vakanski et al., 2020 [105],[a]	Segmentation of breast tumors in ultrasound.	U-Net with attention blocks.	D 510 ultrasound images. R DSC 90.5 O Low quality of prominence maps generated by attention blocks.
Naser and Deen 2020 [106]	Segmentation and classification of gliomas.	U-Net with VGG16 backbone.	D 110 MRI. R Segmentation: DSC 0.84. Grade II and grade III classification: ACC (0.89), SE, (0.87) and SP (0.92). O The VGG16 spine was used, a simple model that could limit the accuracy of the model. The authors recommend exploring other spines such as ResNet50, Inception v3, or DenseNet.
Ibrahim et al., 2021 [107]	Diagnosis of COVID-19, pneumonia, lung cancer, and regular images.	VGG19, ResNet152V2, ResNet152V2 + (GRU), and ResNet152V2 + Bidirectional GRU.	D X-ray and CT. Covid 19 (4320), pneumonia (5856), Lung cancer (20,000), and normal (3500). R Model VGG19: ACC (98,05), SE (98,05), PR (98,43), SP (99,5), NPV (99,3), F1 (98,24) and MCC (99.66%). O The authors recommend using GAN systems for classification and as an essential part of data augmentation.
Wang et al., 2021 [108]	Classification of lung adenocarcinoma subtypes.	ResNet-34	D 1222 patients with adenocarcinoma of the lung. CT. R 2-category classification: ACC (0.8776). 3-category classification ACC (0.8061). O The study was based only on patients in the initial stage since it is challenging to obtain longitudinal information that allows the implementation of DL systems for predicting survival. The authors suggest integrating complete information and caution that the model may not perform as well at later stages.
Li et al., 2021 [109],[a]	Predicting the pathologic status of suspicious non-palpable breast microcalcifications.	U-Net-Reductive.	D 463 digital mammograms from 260 patients. R AUC (0.906) and ACC (0.787). O The database has few subjects, and a reduced U-Net base model could limit the network's performance. The authors recommend experimenting with more architectures
Zhuang et al., 2021 [110],[a]	Classification of tumors on breast ultrasound images.	VGG, ResNet and Densenet	D 1328 breast ultrasound images. R ACC (0,9548), PR (0,9811), SP (0,9833), SE (0,9392), F1 (0,9571) and AUC (0,9883). O
J. Wang and Liu 2021 [111],[a]	Gastric cancer segmentation.	DeepLab v3 +	D 1340 images of gastric cancer pathologic sections. R SE (91,45), SP (92,31), ACC (95,76) and DSC (91,66%). O The results delivered by the algorithm have room for improvement.
Yan et al., 2021 [112],[a]	Prostate segmentation.	PSP Net	D 270 MRI R ACC (0,9865). O The authors note that the model was validated with few samples.
Iqbal et al., 2021 [113],[a]	Prostate cancer detection.	LSTM and ResNet-101	D 230 MRI R LSTM: SE (98,33), SP (100), PPV (100), NPV (100), PR (99.48), MCC (98.79) and AUC (99.99%). ResNet-101: PR (100) and AUC (100%). O Despite the excellent results, the data is small. Therefore, the authors recommend comparing with larger data sets for future work.
Adweb et al., 2021 [114]	Diagnosis of the uterine cervix.	ResNet wiht (Leaky-RELU and PRELU)	D 4000 precancerous and 800 healthy cervical images. R Leaky-RELU: ACC (90.2). PRELU: ACC (100%). O The work is not directly comparable with previous work because of different types of images.
T. Zhang et al., 2021 [115]	Classification of non-solid nodules.	CNN 3D SE-ResNet	D Images of 240 AIS, 277 MIA and 192 IAC. CT R Discrimination AIS from MIA-IAC: ROC (0.820). Discrimination AIS-MIA from IAC: ROC (0.833). O

(*continued on next page*)

Table 2 (*continued*)

Main author	Topic	Architecture	Data sets (D), results (R) and Observation (O)
			More samples are required for parameter optimization to reduce overfitting. The authors suggest using GANs to generate synthetic images in future work.
Maron et al., 2021 [116]	Classification of skin cancer	AlexNet, VGG16+BN, ResNet50 and DenseNet121.	D 194 melanoma and 125 nevus images from 85 unique lesions. R DenseNet121 was the best network with BE (19.6), mBCE (26.8), and mFR (6.3). O The latest skin cancer studies lack external data classification assessments.
Cho et al., 2021 [117],[a]	Classification of pulmonary abnormalities.	ResNet-50	D 9534 chest radiographs R ACC: 0.9330 in the AMC data and 0.9120 in the SNUBH dataset. O Despite the excellent performance of the development, the authors recommend including patients with other diseases to evaluate the real usefulness of the system. In addition, the authors also highlight the need to include more databases, as only CXR images from two centers were included in the study. Finally, further studies are still needed to increase specificity and improve performance in classifying abnormalities.
(Y. Ma et al., 2021 [118]	KRAS mutation status in colorectal cancer.	Network-based on Xception, convLSTM, and attention models	D 3817 T2-weighted MR images from 206 patients. R ACC (0.8803), AUC (0.9427) and SP (0.9075). O The study was performed with a limited number of data and, although the method demonstrated advantages, there is still room for improvement in model performance.
J. Ma et al., 2021 [119],[a]	Classification of benign and malignant lesions	ResNet50	D 200 contrast-enhanced breast cone-beam CT images from 165 patients. R AUC (0.727), SE (0.80) and SP (0.60). O The study is based on a small number of images, and the data are from a single center. The authors highlight the need for more data to perform an objective validation with external data. In addition, they also suggest diversifying the data sets.
El Houby & Yassin, 2021 [120],[a]	Classification of breast lesions	CNN conventional	D 1231 total mammograms for training R INbreast dataset: SE (0.9655), SP (0.9649), ACC (0.9652), and AUC (0.98). MIAS dataset: SE (0.98), SP (0.926), ACC (0.953), and AUC (0.974). O Despite the promising results, the authors point out that more accurate systems are needed before implementation in the hospital setting. They also mention that more careful experiments are needed. In addition, in future work, the authors propose to use them with local data.
Guo et al., 2021 [121],[a]	Segmentation of medical images	New hierarchical network (DW-HieraSeg)	D Images of binary polyp, melanoma, and clinically relevant anatomies. R DSC (88293), JD (0.81690), SE (0.88381), SP (0.99524), ACC (0.97474), and F1 (0.88036) O
X. Cao et al., 2021 [122],[a]	Segmentation of breast mass	Dilated densely connected U-Net	D 170 vol from 107 patients. R DSC (0.6902), JD (0.5661) and HD (4.92 mm). O The authors emphasize that three-dimensional methods still have limitations due to challenges in the computational load of the networks.
R. Zhang & Chung, 2021 [123],[a]	Segmentation of medical images	Residual U-Net and 3D-Unet	D CT of liver tumors and MRI of brain tumors R Liver tumor CTs: DSC (0.7881). Brain tumor MRIs: DSC (0.8339). O The models have room for improvement, and the authors recommend approaching the problems from parallel programming for future work. In this sense, they mention that inference times are high, and the current quantification process is still based on iterative backpropagation with costly computations. The authors recommend that future work investigate the possibility of exploiting the potential of advanced discrete optimization methods to assist the quantification process further.
Shi et al., 2021 [124]	Segmentation of pulmonary nodules	Residual U-Net	D 2576 CT images of pulmonary nodules R ACC (0.9457) O
Gao & Almekkawy, 2021 [125]	Segmentation of liver tumors	A nested U-Net (ASU-Net++)	D 480 ultrasound images, 253 CTs of the data sets denoted SYSU-CT and subCT. R Ultrasound images: DSC (0.9153). SYSU-CT: DSC (0.9413) and subCT: DSC (0.9246). O Small data sets limit the performance of the models, as it becomes more difficult to converge the model.
Z. Yang et al., 2021 [126],[a]	Classification of mass	New deep CNN named MommiNet-v2.	D 10 312 mammographs from 2578 cases R SE (0,898) O Data come from only a few collaborators. The authors recommend increasing the effort to build larger-scale multicenter datasets.
Jain et al., 2021 [127],[a]	Segmentation for atherosclerotic plaque	UNet, UNet+, SegNet, SegNet-UNet, and SegNet-UNet +	D 970 image frames of ultrasound. R The AUC metric for the 5 models was: (0.91), (0.911), (0.908), (0.905), and (0.898) O
	Classification of Skin lesion		D 33126 dermoscopic skin images from 2000 patients

(*continued on next page*)

Table 2 (*continued*)

Main author	Topic	Architecture	Data sets (D), results (R) and Observation (O)
Sayed et al., 2021 [128],[a]		SqueezeNet, VGG19, GoogleNet, and ResNet50	R ACC (0.9837), SP (0.9647), SE (1.0), F1 (0.9840), and AUC (0.99). O The authors propose to use swarm intelligence algorithms for future work and to approach the problem from non-binary classification, i.e., classify melanomas into different classes.
Ben Hamida et al., 2021 [129],[a]	Segmentation analysis of colon cancer	AlexNet, vgg −16, ResNet, DenseNet and Inception v3	D 396 HES stained colorectal histopathological R ACC (0.9698) with ResNet O The authors highlight the importance of richly annotated datasets for tumor segmentation tasks. Additionally, they highlight that classical CNN suffers from the problem of vanishing gradients, limiting the ability to provide generic representations. One solution is to implement the use of large-scale images combined with large datasets. However, this would also have drawbacks due to the high computational costs involved.
Ferjaoui et al., 2021 [130]	Classification of evolutive lymphoma and residual masses.	ANN	D 1005 diffusion-weighted MRI R SE (0.964), SP (0.909), ACC (0.955), F1 (0.097), and geometric-mean (0.9161). O While the results are promising, the authors recommend continuing efforts to improve diagnostic accuracy by investigating other machine learning or deep learning algorithms. On the other hand, the authors emphasize that longitudinal studies (data over time) are necessary for an evolutionary pathology follow-up.
S. Zhang et al., 2021 [131],[a]	Detecting signet ring cells (gastric cancer)	RetinaNet	D 455 images from the gastric mucosa R Kappa value (0.74) and ACC (0.89) O
Liu et al., 2021 [132],[a]	Prediction of axillary lymph node metastasis	Deformable attention VGG19	D 800 contrast-enhanced computed tomography images R ACC (0.9088) O The study was only carried out in a single center, making it necessary to implement the development in multicenter studies. The authors also highlight that the development was performed on 2D data, leaving the door open for further studies with 3D networks.
Churchill et al., 2021 [133]	Predicting lymph node metastasis in lung cancer	NeuralSeg	D 298 lymph nodes from 140 patients R ACC (0.729), SP (0.908), and NPV (0.759) O Although the results are promising, the authors highlight the importance of future work to evaluate the algorithm in clinical trials.
Kaur et al., 2021 [134],[a]	Sorting for multi-slice computed tomography (liver cancer)	CNN	D 63503 CT images of liver cancer R ACC: 0.9008 for liver, 0.8997 for lung, and 0.8906 for bone. O The work is limited to a single data set. The authors recommend training the model and testing it under other sets. In addition, they also recommend future work with images of different types of cancer. In the same vein, future directions should focus on learning the clinical significance and importance of the features learned by the networks. On the other hand, the authors highlight that the work was performed only on computed tomography images, opening the possibility of extending this work to positron emission tomography classification.
Zou et al., 2021 [135]	Segmentation Breast	Annotation tolerance network (NAT-Net)	D 550 ultrasound images R PR (0.872), SE (0.886), JD (0.797) and F1 (0.879) O The authors emphasize that the quality of the training data and the quantity greatly influence the network performance. However, the operational cost of obtaining manually annotated images restricts the creation of large datasets, especially in medical imaging. Therefore, increased efforts are needed to acquire data quickly and make better use of the data. On the other hand, the authors also recommend continuing this line of research but with more advanced models, seeking better segmentation performance.
L. Cao et al., 2021 [136],[a]	Detection of abnormal cervical cells	Attention feature pyramid network (AttFPN)	D 7030 annotated cervical cytology images R SE (0.9583), SP (0.9481), ACC (0.9508) and AUC (0.991) O The study only has data from a single center. The authors recommend having a higher quality data set from multiple centers to evaluate the proposed method thoroughly. Also, the authors point out that the data have indeterminate atypical lesions, and it is possible to study such lesions in future work.
C. W. Zhang et al., 2021 [137]	Detection of cervical cancer	Fine-tuned LSTM-FCN Network	D 1395 thinprep cytologic test R ACC (0.983), SE (0.981), and SP (0.979) O Although the results are encouraging, the authors comment that large data sets are needed to increase and ensure the generalizability of the networks. Furthermore, they also recommend exploring the interpretation of the models in order to illustrate the classification principles. Additionally, the authors highlight that future work should focus on optimizing model structures.
B.-L. Chen et al., 2021 [138]	Detection of polyp	Faster R-CNN	D 1000 colonoscopy images R PR (0.943), SE (0.925), and F1 (0.934) O
Kurata et al., 2021 [139],[a]	Segmentation of uterine endometrial cancer	U-net	D 200 multi-sequence MRIs. R DSC (0.806), SE (0.816), and PR (0.834)

(*continued on next page*)

Table 2 (*continued*)

Main author	Topic	Architecture	Data sets (D), results (R) and Observation (O)
			O The design was based on images from a single scanner. The authors emphasize that it is necessary to validate the performance with multicenter data to optimize the models.
Gamble et al., 2021 [140],[a]	Breast cancer biomarker status	Inception-V3	D 3274 slides using hematoxylin-and-eosin-stained (H&E) R AUC: estrogen receptor (0.86), progesterone receptor (0.75), and receptor called HER2 (0.60). O For this specific case, the authors highlight that immunohistochemistry protocols and interpretation guidelines constantly change over time, generating variability between historical clinical labels. This would limit the performance of the models on datasets with different characteristics.
Strijbis et al., 2021 [141],[a]	Tumor segmentation in retinoblastoma	Multi-view CNNs	D 46 MR imaging from 23 children. R DSC: eye (0.965), sclera (0.847), vitreous (0.975), lens (0.909), retinal detachment (0.828) and tumor (0.914). O The work was performed on a dataset from a single scanner. The authors recommend using data from multiple sources for future work. Additionally, the study does not address class imbalance or loss functions, limiting the model's effectiveness to the stated hyperparameters. The authors also recommend investigating other types of topologies in search of better performance.
Mohammed et al., 2021 [142],[a]	Classification of cancer types	1D-CNN	D 2166 tumor samples from five cancers: breast, colon adenocarcinoma, ovarian, lung adenocarcinoma, and thyroid cancer. R Average ACC (0.9922) O The study does not consider methods for handling class imbalance, with a more significant number of samples likely to affect the model's reliability.
Schultheiss et al., 2021 [143],[a]	Detection of lung nodule	U-Net and RetinaNet	D 855 CT images R wAFROC FOM (0.81) O While the results are promising, the authors note that the most significant challenges are in the differences between images, such as the higher resolution in the current radiographs. In addition, the authors highlight that the simulation does not completely resemble the real scenario, with new real data sets needing to be explored.
M. Wang et al., 2021 [144]	Diagnosis of hepatocellular carcinoma	VGG and ResNeXt.	D CT images from 7512 patients R Internal test set: ACC (0.81), SE (0.784), SP (0.844) and F1 (0.824). External test set, ACC (0.813), SE (0.894), SP (0.740) and F1 (0.819). O
Park et al., 2021 [145]	Classification of cervical cancer	ResNet50	D 4419 cervicography images. R AUC (0.97) O The classification presented good results; however, the authors point out that not all images were in focus, and some might even be distorted or outside the area of interest. This could limit the performance of the model. In addition, the authors note that the methods were applied under some hyperparameters, and it is likely that the results will differ from other hyperparameters. In addition, the authors suggest applying data augmentation in future work.
C.-W. Wang et al., 2021 [146]	Detection of HSILs or higher for cervical lesion diagnosis.	U-net and SegNet	D 143 whole slide images of conventional Pap smear samples. R PR (0.93), SE (0.90), F-measure (0.88), and JD (0.84) O
Silva et al., 2021 [147],[a]	Classification of EGFR mutation status for lung cancer	Convolutional autoencoder and MLP	D 2669 lesions from thoracic CT scans of 1010 patients. R AUC (0,68) O The authors' comment that more representative data are needed for a complete analysis. Additionally, they suggest including other strategies such as generative adversarial networks to include synthetic samples.
Saber et al., 2021 [148],[a]	Detection and Classification of Breast Cancer	Inception V3, ResNet50, VGG-16, VGG-19, and Inception-V2 ResNet	D 322 mammographic images. R ACC (98.96), SE (97.83), SP (99.13), PR (97.35), F1(97.66), and AUC (0.995) O
Kim et al., 2021 [149]	Histological Image Segmentation and Classification	PSPNet, U-Net, UNet++, and DeepLabV3+ with an entropy-based convolutional module.	D 5000 training images from the whole-slide image. Histological colorectal cancer images. R F1 (0.855) and JD (0.832) O Although the networks gave good results, more studies are needed to adjust the hyperparameters of the entropy-based convolutional module.
Ogino et al., 2021 [150]	Prostate Cancer Stage Prediction	Xception	D T2-weighted MR images of 15 patients R ACC (0.880) O In this particular case of MRI images, the authors highlight some limitations for the use of DL. The different modalities (including the types of sequences) and the differences in spatial resolution limit the applicability of the models, restricting them to a particular type of image or to a single task, as in this case, to classification. Despite these drawbacks, the authors suggest improving performance by

(*continued on next page*)

Table 2 (*continued*)

Main author	Topic	Architecture	Data sets (D), results (R) and Observation (O)
			optimizing hyperparameters such as the number of layers, the dimensions of each layer, and other network configurations.
Martins Jarnalo et al., 2021 [151]	Detection of pulmonary nodule	DL-CAD	D 145 chest CT R SE (0.88), FPR (1.04), and NPV (0.95). O While the results are promising, the authors stress that they are not good enough to replace nodule detection performed by expert radiologists. The systems still deliver high rates of overestimation of nodule size, leading to possible unnecessary follow-up examinations. In this regard, the false-positive rate of the models needs to be improved.
Duran-Lopez et al., 2021 [152]	Detection prostate cancer	Wide & Deep network	D 332 of hematoxylin and eosin-stained slides R ACC (0.9424), SE (0.9887), PR (0.9023), F1 (0.9433), and AUC (0.94) O The authors propose future work to improve the sensitivity and reduce the computational time of the models.
Huang et al., 2021 [153]	Segmentation of Cervical Cel	Generative Adversarial Networks	D 1880 cervical cell images R DSC (0.943), object-level FNR (0.79) for single-cell images. DSC (0.899), object-level FNR (0.64) for overlapping cell. O Although the authors propose future work (such as segmentation of cells in blood images), they also highlight those further studies are needed to improve the performance and applicability of the proposed methods.
X. Chen et al., 2021 [154],[a]	Segmentation of pancreatic cancer	Spiral-ResUNet	D 65 apparent diffusion coefficient (ADC), 69 diffusion-weighted (DWI), 68 T1-weighted (T1w), and 70 T2-weighted (T2w) MR images. R DSC (0.656), (0.640), (0.645), and (0.653), in MRI T2w, T1w, ADC, and DWI, respectively. O
Dipu, Shohan and Salam, 2021 [155]	Detection and Classification of Brain Tumor	YOLOv5 and FastAi	D 1992 Brain MRI R ACC: (0.8595), PR (0.9267), SE (0.8433), F1(0.8830) and, in YOLOv5. ACC (0.9578) in FastAi. O The authors highlight that the system would provide much better results if trained on a more extensive data set over a longer time.
Shah, 2021 [156],[a]	Classification of skin cancer	LRNet	D 10015 pigmented lesion images R SE (0.94), SP (0.917), PR (0.942), and ACC (0.906) O The authors highlight that almost all previously trained networks are very dense and therefore overfit when trained on low-resolution images.
Chan, Liu and Chu, 2021 [157],[a]	Localization of lung tumors	VGG16 with deep convolutional generation confrontation network	D 11082 images from Pneumothorax R ACC (0.82), SE (0.76), SP (0.90), and PR (0.88) O While the results are promising, the authors highlight that further research is needed to adjust the detection parameters and increase the accuracy of the proposed models.
Wetteland et al., 2021 [158],[a]	Prediction of cancer grade in bladder cancer.	VGG16	D 300 digital whole-slide images from patients diagnosed with non-muscle-invasive bladder cancer R PR (0.92), SE (0.90), and F1 0.90 by the best model O The study is limited to cancer-diagnosed data from a single-center, i.e., training material with non-cancerous samples is not available. This limits the applicability of the model to this data set. In addition, the authors comment that the model's behavior on lower quality data is unknown, and it is possible to study this question in future work.
Saunders et al., 2021 [159],[a]	Segmentation of prostate	Optimized U-Net	D 60 T2w MR images R DSC (0.89) and HD (1.15) O The authors propose to include studies for prostate cancer detection and diagnosis as future work.

[a] Research with data augmentation.

generation images. In this context, artificial intelligence has several branches that could complement each other. For example, as mentioned above, networks could move towards data-optimized models with little data, with more excellent generalization capability, and even approaches can focus on data augmentation or more realistic artificial image generation.

12. Conclusions

Deep learning is one of the fastest-growing areas in medical image analysis and has had a significant impact on different applications, both clinical and research. Developments are not waiting to happen, and new results are building a promising future for artificial intelligence, opening the way to more accurate segmentation, classifications, detections, and predictions even at expert radiologists' level. Deep learning techniques are far surpassing conventional methods in medical image analysis. However, there is still a long way to go. In this sense, a general review was conducted to integrate the critical points in artificial intelligence, specifically on deep learning. The article addresses the basic concepts of deep learning from the types of networks, the current and the most recent developments made on medical images related to cancer. The review yielded some fundamental studies and allowed us to define DL's current problems and establish a perspective of the leading research focus. In this context, it is more than evident that, although the current research has promising results, the availability of data biases the performance and limits the implementations to a group with its own characteristics. In other words, the developments are limited only to images like the ones they were trained on. This involves several

solutions, the first and simplest of which is to encourage groups and the scientific community to share their databases. The second approach focuses on more efficient networks or optimization methods that require less data for training. Third, developing new data augmentation methods or image synthesis is an open idea for possible future work to increase the amount of data available or improve the performance of networks with few reference images. Finally, we also highlight those current strategies with medical images are approached only with the pure image, i.e., patient information that could better have relevance in the performance of the models is being omitted.

13. Information sharing statements

This document does not have its own database or sample codes.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research was supported by the research division from INDIGO Technologies (https://indigo.tech/).

Appendix

A. Glossary

AI	Artificial Intelligence
AIS	Adenocarcinoma In Situ
ANN	Artificial Neural Network
ASSD	Average Symmetric Surface Distance
BE	Balanced Error
BCC	Basal Cell Carcinoma
CNN	Convolutional Neural Network
CT	Computed Tomography
DL:	Deep learning
EGFR	Epidermal growth factor receptor
FC	Fully Connected
GAN	Generative Adversarial Networks
GPU	Graphics processing unit
GRU	Gated Recurrent Unit
HSILs	High Grade Squamous Intraepithelial Lesions
IAC	Invasive Adenocarcinoma
IEC	IntraEpidermal Carcinoma
ISICDM	The 2nd International Symposium on Image Computing and Digital Medicine.
KMC	Kasturba Medical College
LN	Lymph Nodes
LSTM	Long Shor-Term Memory
mBCE	mean Balanced Corrupted Error
MCC	Matthew's correlation coefficient
mFR	mean Flip Rate
MIA	Minimally Invasive Adenocarcinoma
MLP	MultiLayer Perceptron
MRI	Magnetic Resonance Imaging
MSD	The Medical Segmentation Decathlon
NIH	The National Institutes of Health
OAR	Organ At Risk
RNN	Recurrent Neural Network
ROC:	Operating Characteristic Curve
SCC	Squamous Cell Carcinoma
wAFROC FOM	weighted Alternative Free Response Operating Characteristic Figure-Of-Merits

B. Development of the analytical solution

We have the following loss function from the assumption of a single-layer MLP network with the linear activation function, with m input features and n observations.

$$J(\boldsymbol{w}) = \frac{1}{2n}\left((X\boldsymbol{w} - \boldsymbol{y})^T (X\boldsymbol{w} - \boldsymbol{y})\right) \tag{B.1}$$

Where $X \in R^{n \times m}$ is the matrix of input features and observations. $w \in R^m$ is the vector of training parameters, and $y \in R^n$ is the vector of outputs for the n observations.

Since one expects to find the vector $\boldsymbol{w}$ that minimizes equation (B.1), it is possible to derive it and equal to zero, as shown in equation (B.2).

$$\frac{dJ(\boldsymbol{w})}{d\boldsymbol{w}} = \frac{1}{2n}\frac{d}{d\boldsymbol{w}}\left((X\boldsymbol{w}-\boldsymbol{y})^T(X\boldsymbol{w}-\boldsymbol{y})\right) = 0 \tag{B.2}$$

$$\frac{d}{d\boldsymbol{w}}\left((X\boldsymbol{w}-\boldsymbol{y})^T(X\boldsymbol{w}-\boldsymbol{y})\right) = 0 \tag{B.3}$$

By applying the derivative of a product, we have the expression of equation (B.4).

$$\left(X\frac{d}{d\boldsymbol{w}}(\boldsymbol{w})\right)^T(X\boldsymbol{w}-\boldsymbol{y}) + (X\boldsymbol{w}-\boldsymbol{y})^T X\frac{d}{d\boldsymbol{w}}(\boldsymbol{w}) = 0 \tag{B.4}$$

And solving the above expression leads to equation (B.6).

$$\left(\frac{d}{d\boldsymbol{w}}(\boldsymbol{w})\right)^T X^T(X\boldsymbol{w}-\boldsymbol{y}) + \left(\boldsymbol{w}^T X^T - \boldsymbol{y}^T\right)X\frac{d}{d\boldsymbol{w}}(\boldsymbol{w}) = 0 \tag{B.5}$$

$$\left(\frac{d}{d\boldsymbol{w}}(\boldsymbol{w})\right)^T X^T X\boldsymbol{w} - \left(\frac{d}{d\boldsymbol{w}}(\boldsymbol{w})\right)^T X^T\boldsymbol{y} + \boldsymbol{w}^T X^T X\frac{d}{d\boldsymbol{w}}(\boldsymbol{w}) - \boldsymbol{y}^T X\frac{d}{d\boldsymbol{w}}(\boldsymbol{w}) = 0 \tag{B.6}$$

Note that, because of the dimensionality of vectors and matrices, each term of the equation is a scalar. Furthermore, the positive terms (additions) and negative terms (subtractions) are the transposed equivalents between the same signs. Therefore, these terms are equal (only apply for the transpose of a scalar). Then, equation (B.6) can be reduced to equation (B.7).

$$2\left(\frac{d}{d\boldsymbol{w}}(\boldsymbol{w})\right)^T X^T X\boldsymbol{w} - 2\left(\frac{d}{d\boldsymbol{w}}(\boldsymbol{w})\right)^T X^T\boldsymbol{y} = 0 \tag{B.7}$$

Thus, by clearing $\boldsymbol{w}$, we arrive at equation (B.10).

$$\left(\frac{d}{d\boldsymbol{w}}(\boldsymbol{w})\right)^T\left(2X^T X\boldsymbol{w} - 2X^T\boldsymbol{y}\right) = 0 \tag{B.8}$$

$$2X^T X\boldsymbol{w} - 2X^T\boldsymbol{y} = 0 \tag{B.9}$$

$$\boldsymbol{w} = \left(X^T X\right)^{-1}X^T\boldsymbol{y} \tag{B.10}$$

References

[1] Yates DR, Vaessen C, Roupret M. From Leonardo to da Vinci: the history of robot-assisted surgery in urology. BJU Int 2011:1–6. https://doi.org/10.1111/j.1464-410X.2011.10576.x.

[2] Nicolescu B. In: Gibbs P, editor. Technological singularity: the dark side, in transdisciplinary higher education: a theoretical basis Revealed in practice. Cham: Springer International Publishing; 2017. p. 155–61.

[3] Armstrong S. Introduction to the technological singularity. In: Callaghan V, Miller J, Yampolskiy R, Armstrong S, editors. In *the technological singularity: Managing the journey*. Berlin, Heidelberg: Springer Berlin Heidelberg; 2017. p. 1–8.

[4] Ravi D, et al. Deep learning for health informatics. IEEE J. Biomed. Heal. Informatics 2017;21(1):4–21. https://doi.org/10.1109/JBHI.2016.2636665.

[5] Schmidhuber J. Deep learning in neural networks: an overview. Neural Network 2015;61:85–117. https://doi.org/10.1016/j.neunet.2014.09.003.

[6] Kulkarni S, Seneviratne N, Baig MS, Khan AHA. Artificial intelligence in medicine: where are we now? Acad Radiol 2020;27(1):62–70. https://doi.org/10.1016/j.acra.2019.10.001.

[7] Krenker A, Bester J, Kos A. Introduction to the artificial neural network. Artif Neural Network - Methodol Adv Biomed Appl 2011. https://doi.org/10.5772/15751.

[8] Han S-H, Kim KW, Kim S, Youn YC. Artificial neural network: understanding the basic concepts without mathematics. Dement. Neurocognitive Disord. 2018;17(3):83. https://doi.org/10.12779/dnd.2018.17.3.83.

[9] Hecht-Nielsen R. Theory of the backpropagation neural Network**Based on 'nonindent' by robert hecht-nielsen, which appeared in proceedings of the international joint conference on neural networks 1, 593–611, june 1989. © 1989 IEEE. In: Wechsler H, editor. In *neural Networks for perception*. Academic Press; 1992. p. 65–93.

[10] Lecun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521(7553):436–44. https://doi.org/10.1038/nature14539.

[11] Deng L, Yu D. "Deep learning: methods and applications," *found*. Trends Signal Process 2014;7(3–4):197–387. https://doi.org/10.1561/2000000039.

[12] Hosny A, Parmar C, Quackenbush J, Schwartz LH, Aerts HJWL. Artificial intelligence in radiology. Nat Rev Canc 2018;18(8):500–10. https://doi.org/10.1038/s41568-018-0016-5.

[13] Mamoshina P, Vieira A, Putin E, Zhavoronkov A. Applications of deep learning in biomedicine. Mol Pharm May 2016;13(5):1445–54. https://doi.org/10.1021/acs.molpharmaceut.5b00982.

[14] Dong S, Wang P, Abbas K. A survey on deep learning and its applications. Comput. Sci. Rev. 2021;40:100379. https://doi.org/10.1016/j.cosrev.2021.100379.

[15] Wason R. Deep learning: evolution and expansion. Cognit Syst Res 2018;52:701–8. https://doi.org/10.1016/j.cogsys.2018.08.023.

[16] Esteva A, et al. Deep learning-enabled medical computer vision. npj Digit. Med. 2021;4(1):5. https://doi.org/10.1038/s41746-020-00376-2.

[17] Kim J, Hong J, Park H. Prospects of deep learning for medical imaging. Precis Futur. Med Jun. 2018;2(2):37–52. https://doi.org/10.23838/pfm.2018.00030.

[18] Levine AB, Schlosser C, Grewal J, Coope R, Jones SJM, Yip S. Rise of the machines: advances in deep learning for cancer diagnosis. Trends Canc. 2019;5(3):157–69. https://doi.org/10.1016/j.trecan.2019.02.002.

[19] Wang F, Casalino LP, Khullar D. "Deep learning in medicine—promise, progress, and challenges. JAMA Intern. Med. 2019;179(3):293–4. https://doi.org/10.1001/jamainternmed.2018.7117.

[20] Tafish MH, El-Halees AM. "Breast cancer severity degree predication using data mining techniques in the gaza strip," in *2018 international Conference on promising electronic technologies*. ICPET); 2018. p. 124–8. https://doi.org/10.1109/ICPET.2018.00029.

[21] Alyafeai Z, Ghouti L. A fully-automated deep learning pipeline for cervical cancer classification. Expert Syst Appl 2020;141:112951. https://doi.org/10.1016/j.eswa.2019.112951.

[22] Benhammou Y, Achchab B, Herrera F, Tabik S. BreakHis based breast cancer automatic diagnosis using deep learning: taxonomy, survey and insights. Neurocomputing 2020;375:9–24. https://doi.org/10.1016/j.neucom.2019.09.044.

[23] Liu Y, et al. A deep learning system for differential diagnosis of skin diseases. Nat Med 2020;26(6):900–8. https://doi.org/10.1038/s41591-020-0842-3.

[24] Zhu X, Yao J, Huang J. Deep convolutional neural network for survival analysis with pathological images. In: 2016 IEEE international conference on bioinformatics and biomedicine. BIBM; 2016. p. 544–7. https://doi.org/10.1109/BIBM.2016.7822579.

[25] American Cancer Society. Breast cancer risk and prevention. cancer.org; 2019. p. 1–46. https://www.cancer.org/content/dam/CRC/PDF/Public/8578.00.pdf. [Accessed 31 August 2021].

[26] Akin O, et al. Advances in oncologic imaging: update on 5 common cancers. CA. Canc. J. Clin. 2012;62(6):364–93. https://doi.org/10.3322/caac.21156.

[27] World Health Organization, "Cancer." https://www.who.int/en/news-room/fact-sheets/detail/cancer. [Accessed 31 August 2021].

[28] Siegel RL, Miller KD, Jemal A. Cancer statistics. CA. Canc. J. Clin. 2019;69(1):7–34. https://doi.org/10.3322/caac.21551. 2019.

[29] American Cancer Society. Breast cancer facts & figures 2019-2020. Atlanta: American Cancer Society, Inc. 2019.; 2020. p. 1–44. https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/breast-cancer-facts-and-figures/breast-cancer-facts-and-figures-2019-2020.pdf. [Accessed 31 August 2021].

[30] Sung H, et al. Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. CA. Canc. J. Clin. 2021: 1–41. https://doi.org/10.3322/caac.21660. vol. 0, no. 0.

[31] Anttila S, Boffetta P. Occupational cancers. Occup. Canc. 2020:1–640. https://doi.org/10.1007/978-3-030-30766-0.

[32] World Health Organization. Breast cancer now most common form of cancer. WHO Taking Action 2021. https://www.who.int/news/item/03-02-2021-breast-cancer-now-most-common-form-of-cancer-who-taking-action. [Accessed 31 August 2021].

[33] American Cancer Society. Cancer facts & figures 2020. BMC Publ Health 2020;5 (1):1–8. https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/annual-cancer-facts-and-figures/2020/cancer-facts-and-figures-2020.pdf. [Accessed 31 August 2021].

[34] McGonigle N. Lung cancer. Surgery(UK) 2020;38(5):249–54. https://doi.org/10.1016/j.mpsur.2020.03.008.

[35] American Society of Clinical Oncology (ASCO). Brain tumor: statistics cancer.net doctor-approved patient information from ASCO. Cancer 2021. Net Editorial Board, https://www.cancer.net/cancer-types/brain-tumor/statistics. [Accessed 31 August 2021].

[36] Gazi PM, Yang K, Burkett Jr GW, Aminololama-Shakeri S, Anthony Seibert J, Boone JM. Evolution of spatial resolution in breast CT at UC Davis. Med Phys 2015;42(4):1973–81. https://doi.org/10.1118/1.4915079.

[37] Alom MZ, et al. The history began from AlexNet: a comprehensive survey on deep learning approaches. Cornell University; 2018. ArXiv ID: 1803.01164.

[38] Castiglioni I, et al. AI applications to medical images: from machine learning to deep learning. Phys Med 2021;83:9–24. https://doi.org/10.1016/j.ejmp.2021.02.006.

[39] Liu Z, et al. A survey on applications of deep learning in microscopy image analysis. Comput Biol Med 2021;134:104523. https://doi.org/10.1016/j.compbiomed.2021.104523.

[40] Haykin S. Multilayer perceptrons. Neural Networks and learning machines. Third. USA: Prentice Hall; 2009. p. 122–5. https://cours.etsmtl.ca/sys843/REFS/Books/ebook_Haykin09.pdf.

[41] Karhunen J, Raiko T, Cho K. Chapter 7 - unsupervised deep learning: a short review,. In: Bingham E, Kaski S, Laaksonen J, Lampinen J, editors. Advances in independent component analysis and learning machines. Academic Press; 2015. p. 125–42.

[42] Peng Z. Multilayer perceptron algebra. Cornell University; 2017. arXiv ID: 1701.04968.

[43] Kuo C-CJ. Understanding convolutional neural networks with a mathematical model. J Vis Commun Image Represent 2016;41:406–13. https://doi.org/10.1016/j.jvcir.2016.11.003.

[44] Albawi S, Mohammed TA, Al-Zawi S. Understanding of a convolutional neural network. In: 2017 international conference on engineering and technology. ICET; 2017. p. 1–6. https://doi.org/10.1109/ICEngTechnol.2017.8308186.

[45] LeCun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proc IEEE 1998;86(11):2278–323. https://doi.org/10.1109/5.726791.

[46] Ming Y, et al. Understanding hidden memories of recurrent neural networks,. In: 2017 IEEE conference on visual analytics science and technology. VAST); 2017. p. 13–24. https://doi.org/10.1109/VAST.2017.8585721.

[47] Karpathy A, Johnson J, Fei-Fei L. Visualizing and understanding recurrent networks [Online]. Available: https://arxiv.org/abs/1506.02078; 2015.

[48] Gers FA, Schmidhuber J, Cummins F. Learning to forget: continual prediction with LSTM," in *1999 ninth international Conference on artificial neural networks ICANN 99*. Conf. Publ. No. 470) 1999;2:850–5. https://doi.org/10.1049/cp:19991218. 2.

[49] Cho K, et al. Learning phrase representations using {RNN} Encoder{–}Decoder for statistical machine translation. In: Proceedings of the 2014 conference on empirical methods in natural language processing ({EMNLP}); 2014. p. 1724–34. https://doi.org/10.3115/v1/D14-1179.

[50] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput. Nov. 1997;9(8):1735–80. https://doi.org/10.1162/neco.1997.9.8.1735.

[51] Nadaraya EA. On estimating regression. Teor. Veroyatnost. i Primen. 1964;9(1): 157–9. https://doi.org/10.1137/1109020.

[52] Watson GS. Smooth regression analysis. Indian J Stat May 1964;26(4):359–72. http://www.jstor.org/stable/25049340. [Accessed 31 August 2021].

[53] Vuckovic J, Baratin A, des Combes RT. A mathematical theory of attention [Online]. Available: https://arxiv.org/abs/2007.02876; 2020.

[54] Vaswani A, et al. Attention is all you need. Cornell University; 2017. arXiv ID: 1706.03762.

[55] Yu T, Zhu H. Hyper-parameter optimization: a review of algorithms and applications. Cornell University; 2020. arXiv ID: 2003.05689.

[56] Li M, Zhang T, Chen Y, Smola AJ. Efficient mini-batch training for stochastic optimization. In: Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining; 2014. p. 661–70. https://doi.org/10.1145/2623330.2623612.

[57] Wang Q, Ma Y, Zhao K, Tian Y. A comprehensive survey of loss functions in machine learning. Ann. Data Sci. 2020. https://doi.org/10.1007/s40745-020-00253-5.

[58] Yi-de M, Qing L, Qian Z. Automated image segmentation using improved PCNN model based on cross-entropy. In: Proceedings of 2004 international Symposium on intelligent multimedia. Video and Speech Processing; 2004. p. 743–6. https://doi.org/10.1109/ISIMP.2004.1434171.

[59] Pihur V, Datta S, Datta S. Weighted rank aggregation of cluster validation measures: a Monte Carlo cross-entropy approach. Bioinformatics 2007;23(13): 1607–15. https://doi.org/10.1093/bioinformatics/btm158.

[60] Sudre CH, Li W, Vercauteren T, Ourselin S, Jorge Cardoso M. Generalised Dice overlap as a deep learning loss function for highly unbalanced segmentations. In: Lecture notes in computer science; 2017. p. 240–8. https://doi.org/10.1007/978-3-319-67558-9_28.

[61] Tversky A. Features of similarity. Psychol Rev 1997;84:327–52. https://doi.org/10.1037/0033-295X.84.4.327.

[62] Salehi SSM, Erdogmus D, Gholipour A. Tversky loss function for image segmentation using 3D fully convolutional deep networks BT - machine learning in medical imaging. 2017. p. 379–87. https://doi.org/10.1007/978-3-319-67389-9_44.

[63] Lin TY, Goyal P, Girshick R, He K, Dollar P. Focal loss for dense object detection. IEEE Trans Pattern Anal Mach Intell 2020;42(2):318–27. https://doi.org/10.1109/TPAMI.2018.2858826.

[64] Jadon S. "A survey of loss functions for semantic segmentation,". In: 2020 IEEE conf. Comput. Intell. Bioinforma. Comput. Biol.; 2020. https://doi.org/10.1109/cibcb48159.2020.9277638.

[65] Yan X, Su XG. Linear regression analysis. Theory and computing. USA: World Scientific; 2009. https://doi.org/10.1142/6986.

[66] Ruder S. An overview of gradient descent optimization algorithms. Cornell University; 2016. p. 1–14. arXiv ID: 1609.04747.

[67] Sharma S, Sharma S, Anidhya A. Understanding activation functions in neural networks. Int. J. Eng. Appl. Sci. Technol. 2017;4(12):310–6. https://doi.org/10.33564/IJEAST.2020.v04i12.054.

[68] Giusti A, Cireşan DC, Masci J, Gambardella LM, Schmidhuber J. Fast image scanning with deep max-pooling convolutional neural networks. 2013 IEEE Int. Conf. Image Process. ICIP 2013 - Proc. 2013:4034–8. https://doi.org/10.1109/ICIP.2013.6738831.

[69] Ranzato M, Huang FJ, Boureau YL, LeCun Y. Unsupervised learning of invariant feature hierarchies with applications to object recognition. IEEE Comput Soc Conf Comput Vis Pattern Recogn 2007. https://doi.org/10.1109/CVPR.2007.383157. 0–7.

[70] Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. Journal of Machine Learning Research 2014;15(1):1929–58. http://jmlr.org/papers/v15/srivastava14a.html.

[71] Goodfellow I, Bengio Y, Courville A. Optimization for training deep models. In: Deep learning. MIT Press; 2016. p. 313–7. http://www.deeplearningbook.org.

[72] Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift. 32nd Int. Conf. Mach. Learn. ICML 2015;1: 448–56. arXiv ID: 1502.03167v3.

[73] Šimundić A-M. Measures of diagnostic accuracy: basic definitions. EJIFCC 2009; 19(4):203–11. https://pubmed.ncbi.nlm.nih.gov/27683318.

[74] Powers DMW. Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. Cornell University; 2020. arXiv ID: 2010.16061.

[75] Trevethan R. Sensitivity, specificity, and predictive values: foundations, pliabilities, and pitfalls in research and practice. Front. Public Heal. 2017;5:307. https://doi.org/10.3389/fpubh.2017.00307.

[76] Truong-Loi M-L, Dubois-Fernandez P, Freeman A, Pottier E. The conformity coefficient or how to explore the scattering behaviour from compact polarimetry mode. In: 2009 IEEE radar conference; 2009. p. 1–6. https://doi.org/10.1109/RADAR.2009.4977048.

[77] van Beers F, Lindström A, Okafor E, Wiering M. Deep neural networks with intersection over union loss for binary image segmentation. In: Proceedings of the 8th international conference on pattern recognition applications and methods - ICPRAM; 2019. p. 438–45. https://doi.org/10.5220/0007347504380445.

[78] Karimi D, Salcudean SE. Reducing the Hausdorff distance in medical image segmentation with convolutional neural networks. 2019. https://doi.org/10.1109/TMI.2019.2930068.

[79] Nai Y-H, et al. Comparison of metrics for the evaluation of medical segmentations using prostate MRI dataset. Comput Biol Med 2021;134:104497. https://doi.org/10.1016/j.compbiomed.2021.104497.

[80] Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. Adv Neural Inf Process Syst 2012;25. https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

[81] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc. 2015:1–14. arXiv ID: 1409.1556.

[82] Lin M, Chen Q, Yan S. "Network in network," [Online]. Available, https://arxiv.org/abs/1312.4400v3; 2014.

[83] Szegedy C, et al. Going deeper with convolutions. In: 2015 IEEE conference on computer vision and pattern recognition. CVPR; 2015. p. 1–9. https://doi.org/10.1109/CVPR.2015.7298594.

[84] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: 2016 IEEE conference on computer vision and pattern recognition. CVPR; 2016. p. 770–8. https://doi.org/10.1109/CVPR.2016.90.

[85] He K, Zhang X, Ren S, Sun J. Identity mappings in deep residual networks. In: Computer vision – ECCV 2016; 2016. p. 630–45. https://doi.org/10.1007/978-3-319-46493-0_38.
[86] Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. "Densely connected convolutional networks,". In: 2017 IEEE conference on computer vision and pattern recognition. CVPR); 2017. p. 2261–9. https://doi.org/10.1109/CVPR.2017.243.
[87] Zhu Y, Newsam S. DenseNet for dense FLow. In: 2017 IEEE int. Conf. Image process.; 2017. p. 790–4. https://doi.org/10.1109/ICIP.2017.8296389.
[88] Tan M, Le QV. EfficientNet: Rethinking model scaling for convolutional neural networks. 36th Int. Conf. Mach. Learn. ICML 2019:10691–700. arXiv ID: 1905.11946v5.
[89] Ronneberger O, Fischer P, Brox T, U-Net ". Convolutional networks for biomedical image segmentation. In: Medical image computing and computer-assisted intervention – MICCAI 2015; 2015. p. 234–41. https://doi.org/10.1007/978-3-319-24574-4_28.
[90] Mohammadi R, Shokatian I, Salehi M, Arabi H, Shiri I, Zaidi H. Deep learning-based auto-segmentation of organs at risk in high-dose rate brachytherapy of cervical cancer. Radiother Oncol 2021;159:231–40. https://doi.org/10.1016/j.radonc.2021.03.030.
[91] Nemoto T, et al. Simple low-cost approaches to semantic segmentation in radiation therapy planning for prostate cancer using deep learning with non-contrast planning CT images. Phys Med 2020;78:93–100. https://doi.org/10.1016/j.ejmp.2020.09.004.
[92] Salvi M, et al. A hybrid deep learning approach for gland segmentation in prostate histopathological images. Artif Intell Med 2021;115:102076. https://doi.org/10.1016/j.artmed.2021.102076.
[93] Thomas SM, Lefevre JG, Baxter G, Hamilton NA. Interpretable deep learning systems for multi-class segmentation and classification of non-melanoma skin cancer. Med Image Anal 2021;68:101915. https://doi.org/10.1016/j.media.2020.101915.
[94] Lal S, Das D, Alabhya K, Kanfade A, Kumar A, Kini J, NucleiSegNet ". Robust deep learning architecture for the nuclei segmentation of liver cancer histopathology images. Comput Biol Med 2021;128:104075. https://doi.org/10.1016/j.compbiomed.2020.104075.
[95] Choi MS, et al. Clinical evaluation of atlas- and deep learning-based automatic segmentation of multiple organs and clinical target volumes for breast cancer. Radiother Oncol 2020;153:139–45. https://doi.org/10.1016/j.radonc.2020.09.045.
[96] Zhao X, et al. "Deep learning–based fully automated detection and segmentation of lymph nodes on multiparametric-mri for rectal cancer: a multicentre study. EBioMedicine 2020;56:102780. https://doi.org/10.1016/j.ebiom.2020.102780.
[97] Urushibara A, et al. Diagnosing uterine cervical cancer on a single T2-weighted image: comparison between deep learning versus radiologists. Eur J Radiol 2021;135:109471. https://doi.org/10.1016/j.ejrad.2020.109471.
[98] Chen X, et al. A deep learning-based auto-segmentation system for organs-at-risk on whole-body computed tomography images for radiation therapy. Radiother Oncol 2021;160:175–84. https://doi.org/10.1016/j.radonc.2021.04.019.
[99] Rigaud B, et al. Automatic segmentation using deep learning to enable online dose optimization during adaptive radiation therapy of cervical cancer. Int. J. Radiat. Oncol. 2021;109(4):1096–110. https://doi.org/10.1016/j.ijrobp.2020.10.038.
[100] Courot A, et al. Automatic cervical lymphadenopathy segmentation from CT data using deep learning. Diagn. Interv. Imag. 2021. https://doi.org/10.1016/j.diii.2021.04.009.
[101] Gonzalez Y, et al. Semi-automatic sigmoid colon segmentation in CT for radiation therapy treatment planning via an iterative 2.5-D deep learning approach. Med Image Anal 2021;68:101896. https://doi.org/10.1016/j.media.2020.101896.
[102] Zhang Y, et al. A deep learning framework for pancreas segmentation with multi-atlas registration and 3D level-set. Med Image Anal 2021;68:101884. https://doi.org/10.1016/j.media.2020.101884.
[103] Caballo M, Pangallo DR, Mann RM, Sechopoulos I. Deep learning-based segmentation of breast masses in dedicated breast CT imaging: radiomic feature stability between radiologists and artificial intelligence. Comput Biol Med 2020;118:103629. https://doi.org/10.1016/j.compbiomed.2020.103629.
[104] Balagopal A, et al. A deep learning-based framework for segmenting invisible clinical target volumes with estimated uncertainties for post-operative prostate cancer radiotherapy. Med Image Anal 2021:102101. https://doi.org/10.1016/j.media.2021.102101.
[105] Vakanski A, Xian M, Freer PE. Attention-enriched deep learning model for breast tumor segmentation in ultrasound images. Ultrasound Med Biol 2020;46(10):2819–33. https://doi.org/10.1016/j.ultrasmedbio.2020.06.015.
[106] Naser MA, Deen MJ. Brain tumor segmentation and grading of lower-grade glioma using deep learning in MRI images. Comput Biol Med 2020;121:103758. https://doi.org/10.1016/j.compbiomed.2020.103758.
[107] Ibrahim DM, Elshennawy NM, Sarhan AM. Deep-chest: multi-classification deep learning model for diagnosing COVID-19, pneumonia, and lung cancer chest diseases. Comput Biol Med 2021;132:104348. https://doi.org/10.1016/j.compbiomed.2021.104348.
[108] Wang C, et al. Deep learning for predicting subtype classification and survival of lung adenocarcinoma on computed tomography. Transl. Oncol. 2021;14(8):101141. https://doi.org/10.1016/j.tranon.2021.101141.
[109] Li M, Zhu L, Zhou G, He J, Jiang Y, Chen Y. Predicting the pathological status of mammographic microcalcifications via a radiomics approach. Intell. Med. 2021. https://doi.org/10.1016/j.imed.2021.05.003.
[110] Zhuang Z, Yang Z, Raj ANJ, Wei C, Jin P, Zhuang S. Breast ultrasound tumor image classification using image decomposition and fusion based on adaptive multi-model spatial feature fusion. Comput Methods Progr Biomed 2021:106221. https://doi.org/10.1016/j.cmpb.2021.106221.
[111] Wang J, Liu X. Medical image recognition and segmentation of pathological slices of gastric cancer based on Deeplab v3+ neural network. Comput Methods Progr Biomed 2021:106210. https://doi.org/10.1016/j.cmpb.2021.106210.
[112] Yan L, et al. PSP net-based automatic segmentation network model for prostate magnetic resonance imaging. Comput Methods Progr Biomed 2021:106211. https://doi.org/10.1016/j.cmpb.2021.106211.
[113] Iqbal S, et al. Prostate cancer detection using deep learning and traditional techniques. IEEE Access 2021;9:27085–100. https://doi.org/10.1109/ACCESS.2021.3057654.
[114] Adweb KMA, Cavus N, Sekeroglu B. Cervical cancer diagnosis using very deep networks over different activation functions. IEEE Access 2021;9:46612–25. https://doi.org/10.1109/ACCESS.2021.3067195.
[115] Zhang T, et al. High-resolution CT image analysis based on 3D convolutional neural network can enhance the classification performance of radiologists in classifying pulmonary non-solid nodules. Eur J Radiol 2021:109810. https://doi.org/10.1016/j.ejrad.2021.109810.
[116] Maron RC, et al. A benchmark for neural network robustness in skin cancer classification. Eur J Canc 2021;155:191–9. https://doi.org/10.1016/j.ejca.2021.06.047.
[117] Cho Y, Park B, Lee SM, Lee KH, Seo JB, Kim N. Optimal number of strong labels for curriculum learning with convolutional neural network to classify pulmonary abnormalities in chest radiographs. Comput Biol Med Aug. 2021:104750. https://doi.org/10.1016/j.compbiomed.2021.104750.
[118] Ma Y, Wang J, Song K, Qiang Y, Jiao X, Zhao J. "Spatial-Frequency dual-branch attention model for determining KRAS mutation status in colorectal cancer with T2-weighted MRI," *Comput.* Methods Program Biomed Sep. 2021;209:106311. https://doi.org/10.1016/j.cmpb.2021.106311.
[119] Ma J, et al. Distinguishing benign and malignant lesions on contrast-enhanced breast cone-beam CT with deep learning neural architecture search. Eur J Radiol Sep. 2021;142:109878. https://doi.org/10.1016/j.ejrad.2021.109878.
[120] El Houby EMF, Yassin NIR. Malignant and nonmalignant classification of breast lesions in mammograms using convolutional neural networks. Biomed Signal Process Contr 2021;70. https://doi.org/10.1016/j.bspc.2021.102954.
[121] Guo X, Yang C, Yuan Y. Dynamic-weighting hierarchical segmentation network for medical images. Med Image Anal Oct. 2021;73:102196. https://doi.org/10.1016/j.media.2021.102196.
[122] Cao X, Chen H, Li Y, Peng Y, Wang S, Cheng L. Dilated densely connected U-Net with uncertainty focus loss for 3D ABUS mass segmentation. Comput Methods Progr Biomed Sep. 2021;209:106313. https://doi.org/10.1016/j.cmpb.2021.106313.
[123] Zhang R, Chung ACS. MedQ: lossless ultra-low-bit neural network quantization for medical image segmentation. Med Image Anal Aug. 2021:102200. https://doi.org/10.1016/j.media.2021.102200.
[124] Shi J, Ye Y, Zhu D, Su L, Huang Y, Huang J. "Comparative analysis of pulmonary nodules segmentation using multiscale residual U-Net and fuzzy C-means clustering," *Comput.* Methods Program Biomed Sep. 2021;209:106332. https://doi.org/10.1016/j.cmpb.2021.106332.
[125] Gao Q, Almekkawy M, ASU-Net++ ". A nested U-Net with adaptive feature extractions for liver tumor segmentation. Comput Biol Med Aug. 2021:104688. https://doi.org/10.1016/j.compbiomed.2021.104688.
[126] Yang Z, et al. MommiNet-v2: mammographic multi-view mass identification networks. Med Image Anal Aug. 2021:102204. https://doi.org/10.1016/j.media.2021.102204.
[127] Jain PK, Sharma N, Giannopoulos AA, Saba L, Nicolaides A, Suri JS. Hybrid deep learning segmentation models for atherosclerotic plaque in internal carotid artery B-mode ultrasound. Comput Biol Med Sep. 2021;136:104721. https://doi.org/10.1016/j.compbiomed.2021.104721.
[128] Sayed GI, Soliman MM, Hassanien AE. A novel melanoma prediction model for imbalanced data using optimized SqueezeNet by bald eagle search optimization. Comput Biol Med Sep. 2021;136:104712. https://doi.org/10.1016/j.compbiomed.2021.104712.
[129] Ben Hamida A, et al. Deep learning for colon cancer histopathological images analysis. Comput Biol Med Sep. 2021;136:104730. https://doi.org/10.1016/j.compbiomed.2021.104730.
[130] Ferjaoui R, Cherni MA, Boujnah S, Kraiem NEH, Kraiem T. Machine learning for evolutive lymphoma and residual masses recognition in whole body diffusion weighted magnetic resonance images. Comput Methods Progr Biomed Aug. 2021:106320. https://doi.org/10.1016/j.cmpb.2021.106320.
[131] Zhang S, Yuan Z, Wang Y, Bai Y, Chen B, Wang H. "REUR: a unified deep framework for signet ring cell detection in low-resolution pathological images. Comput Biol Med Sep. 2021;136:104711. https://doi.org/10.1016/j.compbiomed.2021.104711.
[132] Liu Z, et al. Axillary lymph node metastasis prediction by contrast-enhanced computed tomography images for breast cancer patients based on deep learning. Comput Biol Med Sep. 2021;136:104715. https://doi.org/10.1016/j.compbiomed.2021.104715.
[133] Churchill IF, et al. An artificial intelligence algorithm to predict nodal metastasis in lung cancer. Ann Thorac Surg Aug. 2021. https://doi.org/10.1016/j.athoracsur.2021.06.082.
[134] Kaur A, Chauhan APS, Aggarwal AK. An automated slice sorting technique for multi-slice computed tomography liver cancer images using convolutional

network. Expert Syst Appl 2021;186. https://doi.org/10.1016/j.eswa.2021.115686.
[135] Zou H, Gong X, Luo J, Li T. A robust breast ultrasound segmentation method under noisy annotations. Comput Methods Progr Biomed Aug. 2021:106327. https://doi.org/10.1016/j.cmpb.2021.106327.
[136] Cao L, et al. A novel attention-guided convolutional network for the detection of abnormal cervical cells in cervical cancer screening. Med Image Anal Aug. 2021:102197. https://doi.org/10.1016/j.media.2021.102197.
[137] Zhang CW, Jia DY, Wu NK, Guo ZG, Ge HR. Quantitative detection of cervical cancer based on time series information from smear images. Appl Soft Comput Aug. 2021. https://doi.org/10.1016/j.asoc.2021.107791. 107791.
[138] Chen B-L, Wan J-J, Chen T-Y, Yu Y-T, Ji M. A self-attention based faster R-CNN for polyp detection from colonoscopy images. Biomed Signal Process Contr Sep. 2021;70. https://doi.org/10.1016/j.bspc.2021.103019. 103019.
[139] Kurata Y, et al. Automatic segmentation of uterine endometrial cancer on multi-sequence MRI using a convolutional neural network. Sci Rep Dec. 2021;11(1):14440. https://doi.org/10.1038/s41598-021-93792-7.
[140] Gamble P, et al. Determining breast cancer biomarker status and associated morphological features using deep learning. Commun Med Dec. 2021;1(1):14. https://doi.org/10.1038/s43856-021-00013-3.
[141] Strijbis VIJ, et al. Multi-view convolutional neural networks for automated ocular structure and tumor segmentation in retinoblastoma. Sci Rep Dec. 2021;11(1):14590. https://doi.org/10.1038/s41598-021-93905-2.
[142] Mohammed M, Mwambi H, Mboya IB, Elbashir MK, Omolo B. A stacking ensemble deep learning approach to cancer type classification based on TCGA data. Sci Rep Dec. 2021;11(1):15626. https://doi.org/10.1038/s41598-021-95128-x.
[143] Schultheiss M, et al. Lung nodule detection in chest X-rays using synthetic ground-truth data comparing CNN-based diagnosis to human performance. Sci Rep Dec. 2021;11(1):15857. https://doi.org/10.1038/s41598-021-94750-z.
[144] Wang M, et al. Development of an AI system for accurately diagnose hepatocellular carcinoma from computed tomography imaging data. Br. J. Canc. Aug. 2021. https://doi.org/10.1038/s41416-021-01511-w.
[145] Park YR, Kim YJ, Ju W, Nam K, Kim S, Kim KG. Comparison of machine and deep learning for the classification of cervical cancer based on cervicography images. Sci Rep 2021;11(1):16143. https://doi.org/10.1038/s41598-021-95748-3.
[146] Wang C-W, et al. Artificial intelligence-assisted fast screening cervical high grade squamous intraepithelial lesion and squamous cell carcinoma diagnosis and treatment planning. Sci Rep 2021;11(1):16244. https://doi.org/10.1038/s41598-021-95545-y.
[147] Silva F, et al. EGFR assessment in lung cancer CT images: analysis of local and holistic regions of interest using deep unsupervised transfer learning. IEEE Access 2021;9:58667–76. https://doi.org/10.1109/ACCESS.2021.3070701.
[148] Saber A, Sakr M, Abo-Seida OM, Keshk A, Chen H. A novel deep-learning model for automatic detection and classification of breast cancer using the transfer-learning technique. IEEE Access 2021;9:71194–209. https://doi.org/10.1109/ACCESS.2021.3079204.
[149] Kim H-R, Kim K-J, Lim K-T, Choi D-H. Histological image segmentation and classification using entropy-based convolutional module. IEEE Access 2021;9:90964–76. https://doi.org/10.1109/ACCESS.2021.3091578.
[150] Ogino M, Li Z, Shimizu A. Augmented radiology: feature space transfer model for prostate cancer stage prediction. IEEE Access 2021;9:102559–66. https://doi.org/10.1109/ACCESS.2021.3098038.
[151] Martins Jarnalo CO, Linsen PVM, Blazís SP, van der Valk PHM, Dickerscheid DBM. Clinical evaluation of a deep-learning-based computer-aided detection system for the detection of pulmonary nodules in a large teaching hospital. Clin Radiol 2021. https://doi.org/10.1016/j.crad.2021.07.012.
[152] Duran-Lopez L, et al. Wide & Deep neural network model for patch aggregation in CNN-based prostate cancer detection systems. Comput Biol Med 2021:104743. https://doi.org/10.1016/j.compbiomed.2021.104743.
[153] Huang J, Yang G, Li B, He Y, Liang Y. Segmentation of cervical cell images based on generative adversarial networks. IEEE Access 2021:1. https://doi.org/10.1109/ACCESS.2021.3104609.
[154] Chen X, Chen Z, Li J, Zhang Y-D, Lin X, Qian X. Model-driven deep learning method for pancreatic cancer segmentation based on spiral-transformation. IEEE Trans Med Imag 2021:1. https://doi.org/10.1109/TMI.2021.3104460.
[155] Dipu NM, Shohan SA, Salam KMA. "Deep learning based brain tumor detection and classification,. In: In 2021 international conference on intelligent technologies. CONIT); 2021. p. 1–6. https://doi.org/10.1109/CONIT51480.2021.9498384.
[156] Shah M. LRNet: skin cancer classification using low-resolution images. In: In 2021 international conference on communication information and computing technology. ICCICT); 2021. p. 1–5. https://doi.org/10.1109/ICCICT50803.2021.9510138.
[157] Chan P-J, Liu A-S, Chu W-C. Using deep learning to locate lung tumor from chest X-ray images. In: In 2021 IEEE 3rd eurasia conference on biomedical engineering, healthcare and sustainability. ECBIOS); 2021. p. 77–9. https://doi.org/10.1109/ECBIOS51820.2021.9510775.
[158] Wetteland R, et al. Automatic diagnostic tool for predicting cancer grade in bladder cancer patients using deep learning. IEEE Access 2021:1. https://doi.org/10.1109/ACCESS.2021.3104724.
[159] Saunders SL, Leng E, Spilseth B, Wasserman N, Metzger GJ, Bolan PJ. Training convolutional networks for prostate segmentation with limited data. IEEE Access 2021;9:109214–23. https://doi.org/10.1109/ACCESS.2021.3100585.