Brief Introduction to Artificial Neural Networks

Culture Sciences de l'Ingénieur

Author: Xihe Ge
with the review of Professor Laurent Oudre

Published on :
July 2022

Xihe Ge is a mechanical engineering postgraduate in the "Future industry and smart systems" discipline at ENS Paris-Saclay, France. Prof. Laurent Oudre is Full-Professor at ENS Paris-Saclay in the Centre Borelli (UMR 9010) laboratory; his research activities focus on signal processing, pattern recognition and machine learning for time series.

1 Introduction

In recent years, with the rapid developments of data processing techniques and the influx of venture capital, artificial intelligence (AI) has asserted its effectiveness in automating tasks and has begun to profoundly impact all aspects of our society, including academic, industry, and public life.
In 2011, IBM's *Watson*, a prominent question-answering computer system, beat the two most successful human contestants of the American popular quiz show *Jeopardy*!, which made people discuss "the potential thinking ability of machines". In 2016, after the world Go champion Lee Sedol was defeated (1:4) by Google's Go program *AlphaGo*, the terms "artificial intelligence (AI)", "machine learning (ML)" and "artificial neural networks (ANN)" draw the attention of the media and public consciousness once again. One year later, the next-generation program, *AlphaGo Master*, won the match by 3:0 against Ke Jie, the top-ranked human player in the world, which opened a new period of competitive games that AI dominated.
This article will first introduce the definitions, applications and widely used methods of AI to bring an overall and intuitive understanding. Furthermore, it will investigate how the human brain neurons bring inspiration to the origin of artificial neural networks. Then, it will provide a general introduction and summary of the related key techniques, including framework, model training and optimization.

2 What is Artificial Intelligence (AI)?

2.1 The definition of AI

The concept "artificial intelligence" was first coined by American computer scientist John McCarthy and three other scholars in 1956 [1], defined for the machines that can be improved to assume some capabilities thought to be like human intelligence, such as learning, adapting, and self-correction.[2] In other words, a system of theories, methods, techniques, and applications can imitate and extend human behavior, such as perceiving the environment, acquiring knowledge, and obtaining optimal results.
AI has different meanings and impressions from the public and academic perspectives. Public and media show great passion for "Artificial General Intelligence", like C-3PO in the movie *Star Wars* or T700 in the *Terminator*. This omnipotent machine has all kinds of perceptions and rationality, who can think abstractly, understand complex ideas, plan, and solve problems as quickly as human beings. Whereas, at the moment, we are still in the research phase of "Artificial Narrow Intelligence", the technology that can perform a specific task as well as or better than humankind.

2.2 The application of AI

AI already has outstanding performance in certain areas, resulting in a series of applications to accomplish tasks that traditional methods cannot solve. Some representative applications in health, language, and finance are introduced as follows:

- Health

  Nowadays, smartwatches can monitor our health data such as electrocardiography, blood oxygen level and sleep status. The AI can then comprehensively analyze our health state and provide personalized health advice. The watches can also monitor walking stability to predict whether older people are at risk of falling and prevent accidents.

  In cancer diagnosis, AI is able to learn considerable numbers of computed tomography images worldwide and has now achieved astonishing predicting accuracy in cancer determination, comparable to the robustness of experienced experts in this field. Thus an excellent AI model, once validated, can be applied to all patients worldwide in the absence of a nearby experienced doctor.

- Language

  AI is also making a big difference in the field of languages. Thanks to today's translation software, all texts can be read by everyone, regardless of the original language, which makes it easier for people to communicate with each other in the business world when collaborating or in the personal world when travelling.

  The voice assistant is common nowadays. It can answer all kinds of questions and help us control various smart products by simply calling on them, making our daily lives far more convenient.

- Finance

  In the past, the lending departments of the bank had to check the qualifications of borrowers constantly, which was time-consuming and costly. AI can help financial institutions analyze their daily activity records to establish comprehensive credit scores and behavior models for the public in real need to obtain financial support while automatically conducting error audits, improving anti-fraud capabilities, reducing the risk of bad debts and shortening lending times.

Besides the aforementioned applications, other fields, such as the Internet industry, public safety, customer service, education, culture, tourism, game, logistics, new energy, pharmacy, manufacturing and construction, are also undergoing dramatic changes with the help of digital transformation and AI.

2.3 The method of AI: machine learning

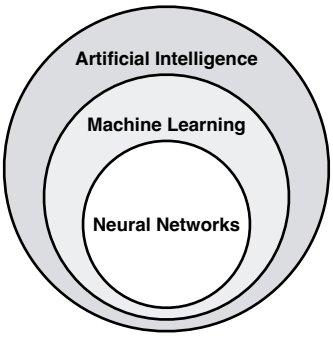

Figure 1: *"NN" ⊆ "ML" ⊆ "AI"*

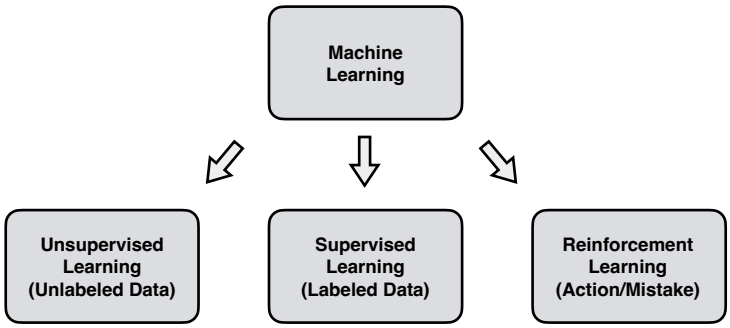

Figure 2: *Three types of machine learning algorithms*

Machine learning is the most famous AI method that allows a system or software to learn knowledge from acquired data. As shown in figure 1, Neural Networks (NN) is included in Machine Learning (ML) and they are included in Artificial Intelligence (AI). Machine learning is a set of methods to achieve AI. Machine learning involves knowledge like probability theory, statistics, approximation theory, convex analysis, and many other multi-disciplinary subjects. The "data-driven" approach is the core idea of machine learning allowing algorithms to make predictions and decisions based on data analysis and interpretation. For the present, machine learning can be broadly divided into three categories (Figure 2):

- Supervised learning

  Labeled data is a designation for data elements that have been tagged with one or several labels identifying specific properties, characteristics, classifications or contained objects. Supervised learning can establish a mapping model from inputs to outputs based on labeled example data sets, then predict the result of new input according to the input-output relationships that one has seen before.

- Unsupervised learning

  Unlabelled data refers to data elements that have not been labeled with tags identifying characteristics, properties or classifications. Unsupervised learning can automatically search features and structures from unlabeled data, group the data into various clusters, identify association rules, and reduce dimensions to realize tasks such as segmentation, pattern detection, and anomaly detection.

- Reinforcement learning

  Unlike the above two methods, the data is not mandatory anymore during reinforcement learning. It is a process of receiving quantified rewards from the environment with different actions to update the model parameters. In other words, reinforcement learning is a "trial-and-error" learning approach to constantly interact with the environment to obtain the best strategy by maximizing the reward. "State, action, reward" are the three key elements of reinforcement learning. The model observes the decision outcome at each step, leading to the next decision to win the final goal. The game and robots are the most widely used areas of this method at present.

The artificial neural network (ANN) is currently one of the most representative supervised learning algorithms. This article will now focus on it to talk about its origin and how it works.

3 Artificial Neural Network

As the most representative machine learning algorithm, the artificial neural network is widely used in various application scenarios and continuously leads AI development. Interestingly, the ANN was inspired and designed from the structure of the human brain. This section will start with the human brain neurons, unravel the mystery of the artificial neural networks step by step, and introduce some key techniques.

3.1 The origin of ANN, neuron

The brain is mainly constituted by neurons to realize signal exchange and information processing. It is estimated that there are nearly 100 billion neurons in the human central nervous system.
The neuron can receive, integrate, conduct and output the excitement. Structurally, it can be divided into three parts: dendrite, cell body and axon. As shown in the image (Figure 3), there are lots of dendric branches on the dendrite to receive excitation from other adjacent neuron axons, form the internal postsynaptic potentials and transmit them to the cell body. The intensity of the postsynaptic potential is related to different properties and states of the synapses. Then, all these postsynaptic potentials are combined and summed up in the cell body; meanwile, an action potential will be generated if the threshold potential is reached. Finally, the axon transmits the action potential from the cell body to the end to conduct the excitation to the next neuron.

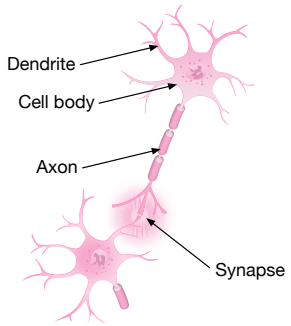

Figure 3: *Neuron*

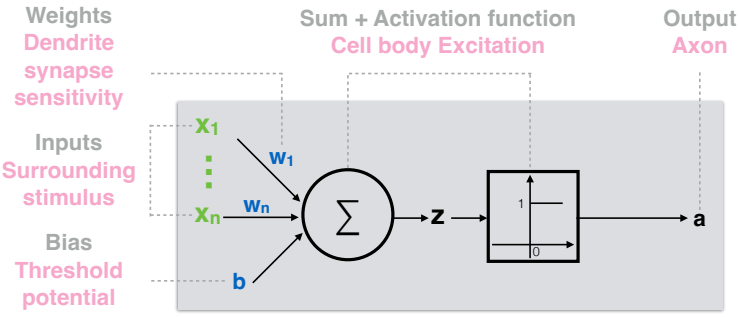

Figure 4: *Perceptron structure*

3.2 Mathematical imitation model, perceptron

The perceptron (Figure 4) is created to mimic the working principles of the neuron mathematically [3]. It has several binary inputs $x_1, x_2, ..., x_n$ (dendrites), each input has a corresponding weight $w_1, w_2, ..., w_n$ (synapse sensitivity). The summation of each input multiplied by its weight is then compared with a threshold -b (threshold potential).

The output equals 1 if the summation is superior to the threshold (active neuron). Otherwise, the output equals 0 (inhibited neuron) (Figure 5 - 1. Step function):

$$a = \begin{cases} 0 \ if \ \sum_{i=1}^{n} w_i x_i + b \leq 0 \\ 1 \ if \ \sum_{i=1}^{n} w_i x_i + b > 0 \end{cases} \tag{1}$$

3.3 Non-linear characteristic, activation function

However, in the algorithm applications, the step function is found complicated to employ with the development of perceptrons and ANN. Therefore, numerous variant activation functions with even better performance have been discovered.
Sigmoid (Figure 5 - 2) is a commonly used "S" type function, which can map variables to the interval (0,1):

$$S(x) = \frac{1}{1 + e^{-x}} \tag{2}$$

and its derivative (Figure 5 - 3) can be easily calculated:

$$S'(x) = \frac{e^{-x}}{(1 + e^{-x})^2} = S(x)(1 - S(x)) \tag{3}$$

It is observed that when the input of the Sigmoid function is very big or very small, the output will enter the "flat" area, and the gradient will "vanish". Whereas when the input is close to 0, the value of the sigmoid derivative is larger.
In addition, the sigmoid function directly outputs a half-saturated state (output = 0.5) when the activation threshold is just reached (input = 0), which does not correspond to the real biological neural networks. Typically, only $1\%-4\%$ of neurons in the brain are active (output > 0) simultaneously and are not easily saturated.
Therefore, the rectified linear unit function (ReLU) (Figure 5 - 4) is more in line with physiological models and is more adapted to avoid exploding and vanishing gradients. Besides, it can also simplify the calculation and accelerate the convergence because its derivative always equals 1 if the input is positive:

$$R(x) = max(0, x) \tag{4}$$

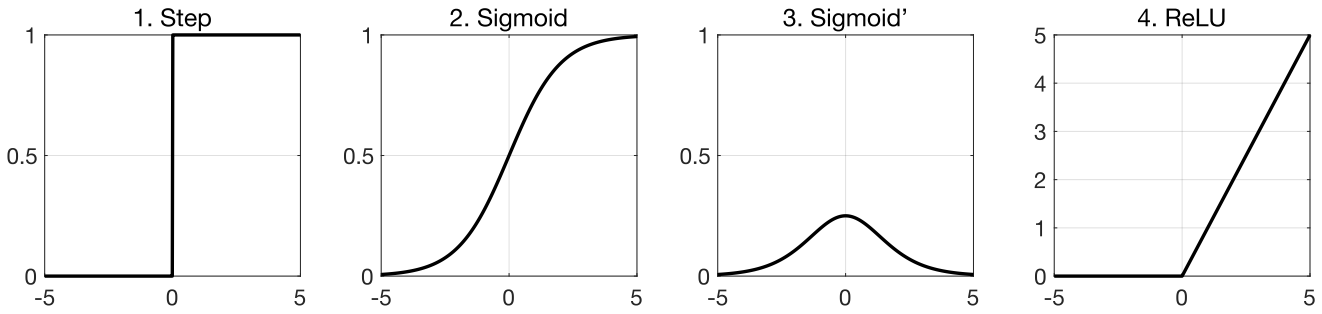

Figure 5: *Step function, Sigmoid function, its derivative and ReLU function*

The above non-linear activation functions can introduce non-linear characteristics into the ANN model. Under certain conditions, these functions can help the model approximate any continuous non-linear mapping relations between input and output with any accuracy if there are enough neurons and hidden layers [4].

3.4 Network model construction, multilayer perceptron

The multilayer perceptron (Figure 6) is a fully connected feedforward ANN model that can map input dataset to output dataset. For example, the inputs can be the feature values of images or documents, while the model's outputs will achieve a specific prediction task, such as identifying an object.
The model consists of the input layer (input variables), the hidden layers (intermediate nodes) and the output layer (output variables). The neurons are fully connected, which means the latter layer's neurons are connected to each previous layer's neuron.

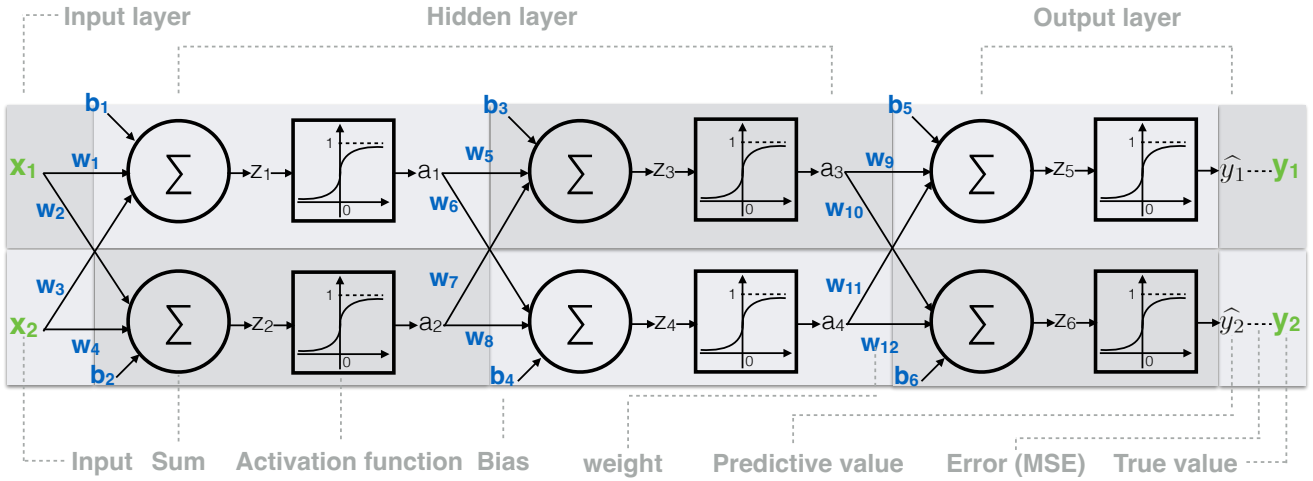

Figure 6: *Multilayer perceptron structure*

Once we understand how individual perceptrons receive, process and transmit signals, it is time to see how to calculate the values layer-by-layer from the input to the output in a fully-connected model to realize forward propagation:

$$\begin{cases} a_1 = \sigma(z_1) = \sigma(x_1 w_1 + x_2 w_3 + b_1) \\ a_2 = \sigma(z_2) = \sigma(x_1 w_2 + x_2 w_4 + b_2) \\ a_3 = \sigma(z_3) = \sigma(a_1 w_5 + a_2 w_7 + b_3) \\ \dots \\ a_n = \sigma(z_n) = \sigma(\mathbf{a} * \mathbf{W} + b_n) \end{cases} \tag{5}$$

where a_n is the output of the current neuron, σ is the activation function, $\mathbf{a}$ and $\mathbf{W}$ are the matrix forms of the outputs and their corresponding weights from the previous layer, b_n is the bias.

3.5 Output satisfaction evaluation, loss function

As mentioned in the previous section, in forward propagation, each layer receives the previous layer's output, processes the value and transmits the result to the next layer. We obtain the predicted value $\hat{y}_i$ at the output layer. How can we evaluate the accuracy of this prediction? The loss function is a mathematical way to quantify our satisfaction with the ANN's outputs. The mean squared error (MSE) is one of the most commonly used loss function:

$$L = \frac{1}{2n} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 \tag{6}$$

The loss value can express the deviation between the ANN's prediction $\hat{y}_i$ and the real result y_i. The larger the loss value, the farther the prediction away from our expectation. "Training" or "learning" means to iterate and adjust the model's internal parameters continuously ($\theta = \{w_1, w_2, .., w_j, b_1, b_2, .., b_k\}$) to achieve more accurate predictions. In other words, the model training aims to minimize the loss value $L(\theta)$ by optimizing the internal parameters θ to let the predicted value be as close to the true value as possible:

$$\underset{\theta}{argmin}\, L(\theta) \tag{7}$$

3.6 How to adjust the parameters, gradient descent-based optimisation

Although the model's internal parameters are high-dimensional, here, we can use two-dimensional parameters (θ_1, θ_2) as an example to observe how they are progressively optimized.
It can be observed that, for a differentiable function, moving in the opposite direction of the function gradient can make the function value decrease in the fastest way. Indeed, the direction of the gradient is the quickest first-order

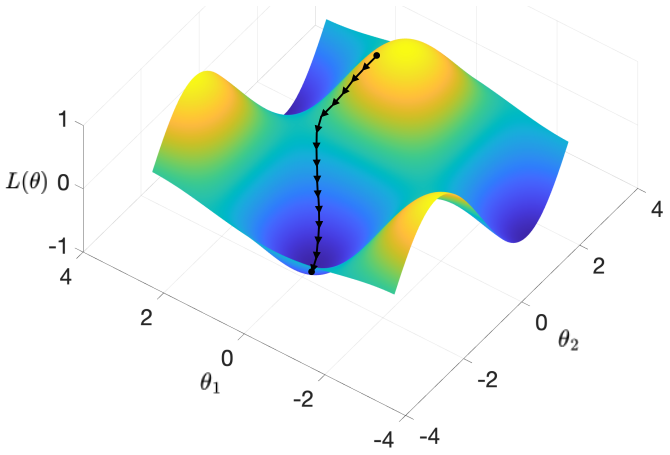

Figure 7: *Demonstration of the gradient descent optimization*

rising direction of the function at the given point. The local minimum can be finally reached by using this method iteratively (Figure7). In other words, the fastest way to descend is to find the steepest direction of the current position and then go down in this direction. Then, it is necessary to find the next position's steepest direction and advance in that new direction until finally reach the local lowest point. Therefore, once the gradient is obtained, the loss value can decrease by optimizing and iterating the ANN's internal parameters to reach the local minimum of the loss function [5]:

$$\theta^{n+1} = \theta^n - \alpha \bigtriangledown L(\theta) \tag{8}$$

where, θ^{n+1} is the next position $(\theta_1^{n+1}, \theta_2^{n+1})$, θ^n is the current position (θ_1^n, θ_2^n), "-"represents the opposite direction, α is a small step, and $\bigtriangledown L(\theta)$ is the gradient direction of the fastest increase.
α is called the learning rate or stride in the gradient descent algorithm. If α is too large, the lowest point may be missed; if α is too small, the descending speed may be reduced. n represents the number of iterations. In practical work, the iteration can be stopped when:

- The maximum iteration number is reached
- The difference between the two successive loss value is less than a small threshold ϵ (The loss value decreases very slowly and may approximate the local/global minimum)
- The W, b absolute error of two successive iterations are less than a small threshold (The loss value decreases very slowly and may approximate the local/global minimum)

3.7 Layer by layer gradient calculation, backpropagation (BP)

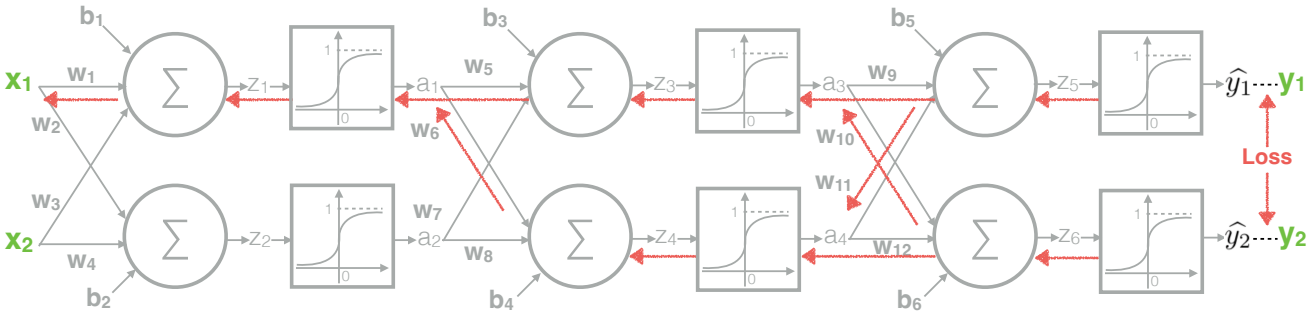

Figure 8: *Backpropagation*

In order to obtain the parameters minimizing the loss value, it is necessary to calculate the gradient at each iteration, which is complicated due to the massive internal parameters (w_j and b_k) of the ANN model. However,

these parameters are often related to each other among successive layers, so that the backpropagation method was invented by Rumelhart et al. in 1986 [6] to help us calculate the gradient layer by layer backward from the last one. The backpropagation can divide the error amount among the connections, which is a milestone for the development of ANN's algorithmic perspective, reducing the repeat calculation and significantly improving the efficiency of gradient searching. This method is essential to understand but slightly difficult. If you are not interested in the detailed algorithm for optimising the weight minimising loss value, you can continue on the section 3.8.
The BP method is mainly based on the derivative chain rule, for example, for the differentiable functions h, g, k, and the variables z, y, x, s:

- case 1
  If $z = h(y)$ and $y = g(x)$, then $\frac{dz}{dx} = \frac{dz}{dy}\frac{dy}{dx}$
- case 2
  If $z = k(x, y)$, $x = g(s)$ and $y = h(s)$, then $\frac{dz}{ds} = \frac{\partial z}{\partial x}\frac{dx}{ds} + \frac{\partial z}{\partial y}\frac{dy}{ds}$

The gradients of the ANN parameters (ex. $\frac{\partial L}{\partial w_9}$,$\frac{\partial L}{\partial w_5}$ and $\frac{\partial L}{\partial w_1}$) can be calculated, knowing the given value (x_1,x_2,y_1,y_2) and the computed value (z_1,...,z_6,a_1,...,a_4,$\hat{y_1}$,$\hat{y_2}$) in the forward propagation (see in the section 3.4):

- $\frac{\partial L}{\partial w_9}$ calculation (see in the figure 8, Similar process for the $\frac{\partial L}{\partial w_{10}}$,$\frac{\partial L}{\partial w_{11}}$,$\frac{\partial L}{\partial w_{12}}$,$\frac{\partial L}{\partial b_5}$,$\frac{\partial L}{\partial b_6}$)
  According to the chain rule(case 1):

$$\frac{\partial L}{\partial w_9} = \frac{\partial L}{\partial \hat{y_1}}\frac{\partial \hat{y_1}}{\partial z_5}\frac{\partial z_5}{\partial w_9} \tag{9}$$

  1. $\frac{\partial L}{\partial \hat{y_1}} = \hat{y_1} - y_1$, because $L = \frac{1}{2n}\sum_{i=1}^{n}(\hat{y_i} - y_i)^2$
  2. $\frac{\partial \hat{y_1}}{\partial z_5} = \hat{y_1}(1 - \hat{y_1})$, because $\hat{y_1} = S(z_5) = \frac{1}{1+e^{-z_5}}$ and $S'(z_5) = \frac{e^{-z_5}}{(1+e^{-z_5})^2} = S(z_5)(1 - S(z_5))$
  3. $\frac{\partial z_5}{\partial w_9} = a_3$, because $z_5 = a_3 * w_9 + a_4 * w_{11} + b_5$

- $\frac{\partial L}{\partial w_5}$ calculation (Similar process for the $\frac{\partial L}{\partial w_6}$,$\frac{\partial L}{\partial w_7}$,$\frac{\partial L}{\partial w_8}$,$\frac{\partial L}{\partial b_3}$,$\frac{\partial L}{\partial b_4}$)
  According to the chain rule (case1 + case 2):

$$\frac{\partial L}{\partial w_5} = \frac{\partial L}{\partial \hat{y_1}}\frac{\partial \hat{y_1}}{\partial z_5}\frac{\partial z_5}{\partial a_3}\frac{\partial a_3}{\partial z_3}\frac{\partial z_3}{\partial w_5} + \frac{\partial L}{\partial \hat{y_2}}\frac{\partial \hat{y_2}}{\partial z_6}\frac{\partial z_6}{\partial a_3}\frac{\partial a_3}{\partial z_3}\frac{\partial z_3}{\partial w_5} \tag{10}$$

$$= (\frac{\partial L}{\partial \hat{y_1}}\frac{\partial \hat{y_1}}{\partial z_5}\frac{\partial z_5}{\partial a_3} + \frac{\partial L}{\partial \hat{y_2}}\frac{\partial \hat{y_2}}{\partial z_6}\frac{\partial z_6}{\partial a_3})\frac{\partial a_3}{\partial z_3}\frac{\partial z_3}{\partial w_5} \tag{11}$$

$$= (\frac{\partial L}{\partial z_5}\frac{\partial z_5}{\partial a_3} + \frac{\partial L}{\partial z_6}\frac{\partial z_6}{\partial a_3})\frac{\partial a_3}{\partial z_3}\frac{\partial z_3}{\partial w_5} \tag{12}$$

  1. $\frac{\partial L}{\partial z_5} = \frac{\partial L}{\partial \hat{y_1}}\frac{\partial \hat{y_1}}{\partial z_5}$ and $\frac{\partial L}{\partial z_6} = \frac{\partial L}{\partial \hat{y_2}}\frac{\partial \hat{y_2}}{\partial z_6}$ are already calculated above
  2. $\frac{\partial z_5}{\partial a_3} = w_9$ and $\frac{\partial z_6}{\partial a_3} = w_{10}$
  3. $\frac{\partial a_3}{\partial z_3} = a_3(1 - a_3)$
  4. $\frac{\partial z_3}{\partial w_5} = a_1$

- $\frac{\partial L}{\partial w_1}$ calculation (Similar process for the $\frac{\partial L}{\partial w_2}$,$\frac{\partial L}{\partial w_3}$,$\frac{\partial L}{\partial w_4}$,$\frac{\partial L}{\partial b_1}$,$\frac{\partial L}{\partial b_2}$)
  Similarly, according to the chain rule and the above calculated gradients, we have:

$$\frac{\partial L}{\partial w_1} = (\frac{\partial L}{\partial z_3}\frac{\partial z_3}{\partial a_1} + \frac{\partial L}{\partial z_4}\frac{\partial z_4}{\partial a_1})\frac{\partial a_1}{\partial z_1}\frac{\partial z_1}{\partial w_1} \tag{13}$$

$$= (\frac{\partial L}{\partial z_3}w_5 + \frac{\partial L}{\partial z_4}w_6)a_1 x_1 \tag{14}$$

The calculation above is only for one set of data. The parameter gradients of all the dataset's loss function are the summation of each data's results:

$$\frac{\partial L(\theta)}{\partial w_j} = \frac{\partial \sum_{n=1}^{N} L^n(\theta)}{\partial w_j} = \sum_{n=1}^{N} \frac{\partial L^n(\theta)}{\partial w_j} \tag{15}$$

where n is the number of each data and N represents the data volume
The same gradient calculation principle is used for the ANN with a large number of layers and neurons:

$$\bigtriangledown L(\theta) = \begin{bmatrix} \frac{\partial L(\theta))}{\partial w_1} \\ \cdots \\ \frac{\partial L(\theta))}{\partial w_j} \\ \frac{\partial L(\theta))}{\partial b_1} \\ \cdots \\ \frac{\partial L(\theta))}{\partial b_k} \end{bmatrix} \tag{16}$$

3.8 Constant parameter, hyperparameter

However, there have always been untrainable constant parameters whose values are set before the learning process. They can be defined as hyperparameters and have crucial influences on the model performance. For a primary ANN, here are some typical hyperparameters:

- Network structure

  Such as the number of layers, the number of neurons per layer, the type of activation function, etc.

- Optimization parameter

  Such as the optimization method, learning rate, batch size, etc.

Adjusting hyperparameters is a non-linear, non-differentiable, non-convex optimization problem that cannot be solved with conventional optimization methods (e.g. gradient descent as mentioned above). It is also time-consuming to test each set of hyperparameters since we need to finish the whole model training process to evaluate the predictive accuracy and generalization performance. Here are three common approaches:

- Grid search

  If the hyperparameter is continuous, such as the learning rate, it needs to be discretized according to the "empirical values" used in other models. Then, we can try the arrangement combination of all hyperparameters to select the best performing set.

- Random search

  Some hyperparameters have a limited impact, while others have a much more significant influence on the model's performance. The grid search method will make unnecessary attempts at unimportant hyperparameters. The random search method generates random combinations of the hyperparameters to configure out the optimum one.

- Bayesian optimization

  Unlike the above two methods, when several sets of hyperparameters are already tested, it is reasonable to take advantage of these existing results to determine the next one. The Bayesian optimization first establishes a probabilistic proxy function that fits the tested results to approximate the real performance function. It then searches for a new set of hyperparameters with the highest probability of optimal performance in the proxy function and tests its real performance. After several iterations, the probabilistic proxy function becomes closer to the real performance function, especially in the highest performance area. Finally, the proxy function may help us find the optimal set of hyperparameters.

The computational volume of these methods will elevate exponentially as the dimension of hyperparameters increases. When we test a set of hyperparameters to train a model, if we find that the learning process is not progressing correctly, we can stop training to drop this set and other similar hyperparameter sets. Finally, we need to rely on engineers with in-depth knowledge and experience to optimize the hyperparameters.

3.9 What is Deep learning?

Generally, the more model parameters we have, the better prediction accuracy we can obtain. If the number of internal parameters is limited, we tend to use more layers but fewer neurons of each layer rather than fewer layers but lots of nodes per layer.
Deep learning is a kind of neural networks comprised of multiple processing hidden layers to learn data representations with multiple levels of abstraction [7]. For instance, the convolutional neural networks (CNN) take face image pixels as inputs, utilize a deep learning framework, and summarize features layer by layer. The features "converge" progressively from the lower level (such as curve segment, oriented edge, color) to the higher level (such as eye, nose, mouth and their corresponded sizes, forms, distances, angles) and output a score to identify the person at the end.
The CNN model's parameters can be considerably reduced thanks to novel ideas like local perception and parameter sharing. Besides, with the advancement of electronic and information technology, especially GPU (graphics processing unit) development and large-scale sensor application for massive data acquisition, it is now possible to train the CNN model with more and more layers and decrease the prediction error as time goes by [8]: (Figure 9)

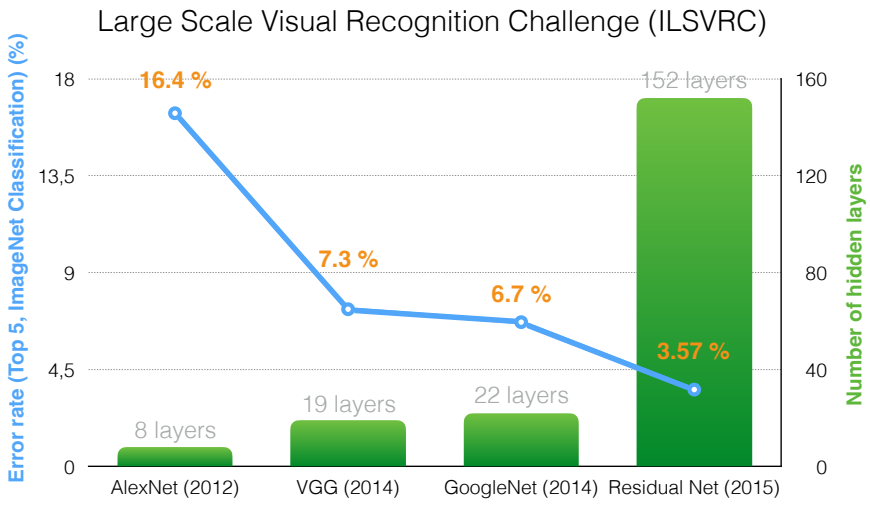

Figure 9: *Revolution of depth in the ILSVRC*

3.10 Current limitation

ANN has some limitations that cannot be ignored at present, such as obtaining high-quality annotations, overfitting problem and ecological impact.
As we know, for supervised learning, the training data need to be labeled. The amount and quality of the training data largely determine the performance of the ANN model. However, high-quality annotations are now expensive and difficult to obtain.
Overfitting is another common issue during the ANN training, which means the model fits too closely to the training set and starts to learn the noise and useless information. It will deteriorate the model's generalization performance on the new dataset and need some techniques to avoid it.
In the Alpha Go example cited at the very beginning of this article, a lot of energy was consumed to train the AI model. During the Go match against Lee Sedol, the environmental impact was not negligible: Alpha Go utilized 1202 CPUs (central processing unit) and 176 GPUs for the calculation!

4 Conclusion

Due to the explosion in computing power and data volume in recent years, artificial neural networks have experienced a great expansion in agriculture, industry and services, accomplishing the functions that cannot be done with traditional methods. The ANNs have already brought significant added-value and efficiency optimisation for human society.

In some aspects, such as large-scale data storage/processing and statistical analysis, the ANN is sometimes more coherent than humans. It also demonstrates excellent performance in specific fields, for instance, computer vision, speech recognition, and intelligent recommendations. Whereas, in terms of innovation/design and decision-making, it still has a long way to go, because "the machine can not think and have consciousness like humans do".

Even though we possess lots of tools allowing us to infer some patterns and rules backward from the qualified model, this technology still has very limited interpretability. It can not gain people's trust in applications requiring high reliability, accuracy and stability, such as autonomous driving, quantitative trading, medical equipment, nuclear power plant, collaborative robot and aerospace industry.

Therefore, to achieve an AI project, it may be critical to analyze the user needs, profitable possibility from a business perspective, and evaluate the feasibility from a data science perspective at the same time. We may also need the help of traditional efficient algorithms, customized models and close cooperation with other advanced technologies to integrate a well-organized ecosystem to finally reach our society's digital and intelligent transformation.

References

[1] Nilsson, N. J., Nilsson, N. J. (1998). Artificial intelligence: a new synthesis. Morgan Kaufmann.

[2] Stephenson Smith, S. (2003). The new international webster's comprehensive dictionary of the english language: deluxe encyclopedic edition (No. REF 428.03 STE. CIMMYT).

[3] Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. Psychological review, 65(6), 386.

[4] Hornik, K., Stinchcombe, M., White, H. (1989). Multilayer feedforward networks are universal approximators. Neural networks, 2(5), 359-366.

[5] Machine learning course: https://www.coursera.org/learn/machine-learning

[6] Rumelhart, D. E., Hinton, G. E., Williams, R. J. (1986). Learning representations by back-propagating errors. nature, 323(6088), 533-536.

[7] LeCun, Y., Bengio, Y., Hinton, G. (2015). Deep learning. nature, 521(7553), 436-444.

[8] ILSVRC data: http://www.image-net.org/challenges/LSVRC/