Article

Not peer-reviewed version

ANN Learning, Attention, and Memory

Vincenzo Manca *

Posted Date: 12 June 2024

doi: 10.20944/preprints202406.0757.v1

Keywords: Artificial Neural Networks; Machine Learning; Artificial Intelligence; Cognitive Systems; Back-propagation

 doi:10.20944/preprints202406.0757.v1

Article

ANN Learning, Attention, and Memory

Vincenzo Manca

University di Verona; vincenzo.manca@univr.it

Abstract: The learning equations of an ANN are presented, giving an extremely concise derivation based on the principle of backpropagation through the descendent gradient. Then, a reflection is developed on the dual attention network that applies the learning equations and coordinates subnetworks toward purposeful behaviors. Speculation is made on possible developments of functionalities that could provide additional skills and at the same time shed light on competencies typical of "natural intelligence".

Keywords: artificial neural networks; machine learning; artificial intelligence; cognitive systems; back-propagation

---

1. Computation: Centralized *vs* Distributed Models

In 1936, Alan Turing introduced the first mathematical model of a computing machine [19,28]. Such a machine executes a program consisting of instructions that it applies based on its internal state, performing actions that alter a workspace and interact with the external environment. The workspace can be reduced to a linear tape (divided into squares) capable of holding symbols. At the start of a computation, symbols placed on the tape constitute the input for the machine. The execution of an instruction involves reading and replacing symbols, moving to other squares, and changing the internal state. The computation ends when the machine reaches a state where there are no instructions to apply. At that point, the tape's content is the result that the machine provides corresponding to the input placed on the tape at the beginning of the computation. The calculation of a machine thus identifies a function that associates input data with output results. A crucial aspect of such a computation model is its partiality; i.e., there may be cases where a machine continues calculation indefinitely. In such cases, the function is undefined for the provided input. The computation is centralized, and the program (list of instructions) is an expression, in an appropriate language, that describes the calculation realized by the machine.

In the same year as the publication of Turing's model, Church proposed a thesis, known as the Turing-Church thesis, which posits that for every computable function, there is a Turing machine computing it, i.e., such machines define a universal model of computation.

Seven years after Turing's seminal work, a young mathematician and an elderly neurologist [18,19] defined artificial neural networks (ANNs), as a computation model inspired by the human brain. Neural networks are a distributed model. A neural network is a directed graph of nodes called neurons and arrows called synapses. The nodes are labeled with basic functions, called transfer functions, and the arrows by numbers called weights, which express the strength with which the synapse transports the output of the neuron to the target neuron that receives it. In such a model, there is no program of instructions, but the behavior of the network emerges from how its parts are connected and from the weights of the synapses. This difference is the origin of a profound difference between the two models, which gradually over about half a century of research [4] has matured Machine Learning (ML) as an alternative paradigm to the Computing Machine. A network is not programmed, but trained, and training consists of presenting it with pairs $(X, F(X))$ where F is a function to be learned that sends real vectors of n components to real vectors of m components. With each example, the network is "adjusted" by varying the weights to decrease the error committed between the output Y it produces and the $F(X)$ to be learned (the network's weights are initially chosen randomly). In other words, a network "learns" functions from examples. When an ANN learns a function F, then it computes F on all its input values, with a possibility of error, which,

 doi:10.20944/preprints202406.0757.v1

under certain assumptions, remains under a fixed threshold. The class of functions computed by ANNs is strictly included in that of functions computed with Turing machines. To achieve complete equivalence, neural networks require features not present in the basic model just outlined [19].

However, a remarkable result holds [5,13,20,21] that, for every continuous and bounded function from n-vectors to m-vectors, and for every chosen level of approximation, there is an appropriate artificial neural network that computes the function with an approximation error lower than the fixed level of approximation. Thus, although neural networks are not as universal as Turing machines, they satisfy another type of universality in approximating real functions between continuous and bounded (of many arguments and results). In this work, we suggest how the neural network model can become a formidable tool for analyzing natural intelligence in mathematical terms, providing precise counterparts to concepts difficult to characterize in rigorous terms.

Today we have machines that speak [3,14], in many ways indistinguishable from human interlocutors. Tomorrow we will have machines that approach increasingly powerful human competencies. The development of these machines will shed new light on our minds. In turn, this knowledge will become the basis for new developments in artificial intelligence. The following sections aim to develop reflections along this direction, starting with a concise presentation of the equations that govern the learning process of neural networks. Many aspects of the discussion are revisitation of themes already present at the dawn of computing, by the founders of such science who, in an impressively clear manner, had a vision of perspectives of development intrinsic to information processing [22,25,29,31].

2. The Structure of ANN

The modern notion of the artificial neural network is an evolution of the initial model by McCulloch and Pitts, emerging from the work [24], where numerical values are introduced as labels for synapses and continuous basic functions as labels for neurons.

An ANN is essentially the composition of a certain number of transfer functions $f_1, f_2, \dots f_k$, which can also all be of the same type, expressed by mathematical formulas with a single variable as an argument (polynomial, sigmoid, hyperbolic tangent, zero-linear, ...). The relationship between neural and metabolic networks is outlined In [17], from the point of view of their computational relevance.

By applying the transfer functions to input arguments $x_1, x_2, \dots x_n$ the composition of such functions produces output values $y_1, y_2, \dots y_m$, thus determining a function from $\mathbb{R}^n$ in $\mathbb{R}^m$ calculated by the network. Each transfer function is associated with a node, called a neuron, identified by the index of the function. Each neuron has afferent and efferent synapses that connect it to other neurons, or receive values from the outside, or send values to the outside. The composition of such functions, defined in the manner we will indicate, produces output values

The transfer functions are composed according to the connections of the various neurons. Let M be a network and $N(M)$ the set of its neurons. The following notation is used to precisely define the operation of the network by establishing how the transfer functions are composed. We write $i \to j$ when a synapse from neuron i is afferent to neuron j. Let us denote by $w_{i,j}$ a number associated with the synapse, called its **weight**. When $w_{i,j} = 0$ no connection is present from i to j.

The weights of the synapses of an ANN uniquely identify the function calculated by the network. The sets of input neurons, output neurons, and internal neurons of M are defined by:

$$In(M) = \{j \in N(M) |\ \nexists i \in N(M) : i \to j\}$$

$$Out(M) = \{j \in N(M) |\ \nexists i \in N(M) : j \to i\}.$$

$$Int(M) = \{j \in N(M) | j \notin In(M), j \notin Out(M)\}.$$

 doi:10.20944/preprints202406.0757.v1

For all $j \in N(M) - In(M)$ we denote by u_j the value given by the transfer function f_j applied to the weighted sum of the values transmitted to the synapses afferent to neuron j. This value is sent from j to all its exit synapses.

Ultimately, a neural network can abstractly be characterized by a pair (Neurons, Synapses) of the following type:

$$M = (\{f_j\}_{j \in N(M)}, \{w_{i,j}\}_{i,j \in N(M)})$$

where $N(M)$ is a set of indices associated with neurons and their corresponding transfer functions, while the synapses are pairs of neuron indices to which the weights are associated (a zero weight indicates the absence of a synapse from the first to the second index). The entire function computed by the network is encoded by the weight matrix $\{w_{i,j}\}_{i,j}$

For each neuron j that is not an input, we define its weighted input z_j by:

$$z_j = \sum_{i:i \to j} w_{i.j} u_i. \tag{1}$$

and its exit u_i is given by:

$$u_j = f_j(z_j) \tag{2}$$

If j is an input neuron, then $z_j = x_j$, and if j is an exit neuron, then $u_j = y_j$.

Let n be the number of inputs of M snd m the number of its exits. If $.F$ is a function from $\mathbb{R}^n$ in $\mathbb{R}^m$, we define a function $E_M(x_i | i \in In(M))$ of errors committed by M on $(x_i | i \in In(M))$ based on differences e_j between target values t_j provided by F on inputs $(x_i | i \in In(M))$ and results provided by M on the same inputs:

$$e_j = (t_j - y_j) | i \in Out(M)$$

Backpropagation [8,15,20,21,26,30] is a method by which a network learns to compute a given function F through examples. We can say that M is associated with a learning network that takes in the errors $(e_j | i \in Out(M))$ and updates the weights by setting:

$$w_{i,j} := w_{i,j} + \Delta w_{i,j} \tag{3}$$

The calculation of $\Delta w_{i,j}$ is done through the coefficients δ_j associated with each neuron through a backward propagation mechanism described by the following learning equations, which reduces the errors e_j committed by M in providing an erroneous output.

It is demonstrated that, under appropriate assumptions about the structure of M and its transfer functions, this procedure converges, meaning that after a sufficient number of examples, the error made by M goes below a predetermined tolerance threshold. Ultimately, M coupled with M_l, along several learning instances, acquires weights at which the function that M computes is below a chosen error threshold.

3. Learning Equations

The following learning equations (4, 5, 6) determine the way M_l operates, where f'_j denotes the derivative of f_j, and η is is a constant less than 1, called the learning rate:

$$\mathbf{\Delta w_{i,j}} = -\eta \mathbf{ffi_j u_i} \tag{4}$$

$$\mathbf{ffi_j} = \frac{\partial \mathbf{E}}{\partial \mathbf{u_j}} \mathbf{f'_j}(\mathbf{z_j}) \quad \textit{if} \ \ \mathbf{j} \in \mathbf{Out}(\mathbf{M}) \tag{5}$$

 doi:10.20944/preprints202406.0757.v1

$$\mathbf{ffi_j} = \mathbf{f'_j}(\mathbf{z_j}) \sum_{\mathbf{k:j\to k}} \mathbf{ffi_k}\ \mathbf{w_{j,k}} \quad \textit{if}\ \ \mathbf{j} \in \mathbf{Int(M)} \tag{6}$$

Theorem [of Back-propagation]
Learning equations derive from the principle of **descendent gradient** *expressed by the following equation [20]:*

$$\Delta w_{i,j} = -\eta \frac{\partial E}{\partial w_{i,j}} \tag{7}$$

telling that errors diminish when weights change in the direction opposite to the error gradient of weights.

Proof. Namely, let us define:

$$\delta_j = \frac{\partial E}{\partial z_j} \tag{8}$$

then, by using the chain rule of derivation the following equations hold:

$$\Delta w_{i,j} = -\eta \frac{\partial E}{\partial w_{i,j}} = -\eta \frac{\partial E}{\partial u_j} \frac{\partial u_j}{\partial z_j} \frac{\partial z_j}{\partial w_{i,j}}. \tag{9}$$

$$\delta_j = \frac{\partial E}{\partial u_j} \frac{\partial u_j}{\partial z_j} \tag{10}$$

whence, l'earning equation (4) is obtained by replacing (10) in the right member of (9), thus deriving, z_j according to equation (1). Equation (5) follows from (10) deriving u_j according to (2).

Equation (6) comes from equation (10) when we observe that for an internal neuron, the error derivative is expressed by the chain rule by applying (1) in the calculation of the derivative of z_j concerning $w_{i,j}$:

$$\frac{\partial E}{\partial u_j} = \sum_{i:j\to i} \frac{\partial E}{\partial z_i} \frac{\partial z_i}{\partial u_j} = \sum_{i:j\to i} \frac{\partial E}{\partial z_i} w_{i,j} = \sum_{i:j\to i} \delta_i w_{i,j}.$$

□

Equation(6) is the kernel of the back-propagation of weights updating for reducing the error committed by M in computing F.

Figure 1 illustrates the updating of two weights. The following equations represent this ANN as a system of equations.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 12 June 2024 doi:10.20944/preprints202406.0757.v1

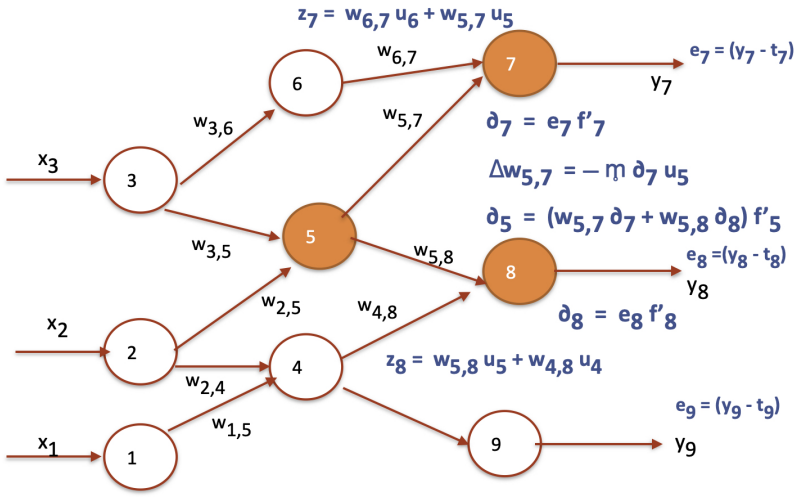

Figure 1. An ANN with 9 neurons updating weights $w_{5,7}$ and $w_{5,8}$ according to la backprpopagation (symbol η is realized by an m with a longer middle line).

$$
\begin{aligned}
u_6 &= f_6(w_{3,6}x_3)\\
u_5 &= f_5(w_{3,5}x_3 + w_{2,5}x_2)\\
u_4 &= f_4(w_{2,4}x_2 + w_{1,4}x_1)\\
y_7 &= f_6(w_{6,7}u_6 + w_{5,7}u_5)\\
y_8 &= f_5(w_{5,8}u_5 + w_{4,8}u_4)\\
y_9 &= f_4(w_{4,9}u_4).
\end{aligned}
$$

If we replace the first three equations in the last three equations the ANN of example is expressed as a composition of its transfer functions.

$$
\begin{aligned}
y_7 &= f_6(w_{6,7}f_6(w_{3,6}x_3) + w_{5,7}f_5(w_{3,5}x_3 + w_{2,5}x_2))\\
y_8 &= f_5(w_{5,8}f_5(w_{3,5}x_3 + w_{2,5}x_2) + w_{4,8}f_4(w_{2,4}x_2 + w_{1,4}x_1))\\
y_9 &= f_4(w_{4,9}f_4(w_{2,4}x_2 + w_{1,4}x_1))
\end{aligned}
$$

4. Attention Mechanisms

The term “Attention" is used widely in cognitive processes, as the capacity of a “cognitive agent" to focus on relevant stimuli and appropriately respond to them. This general meaning is the basis of the following discussion aimed at defining a more specific sense in the context of ANN. To avoid possible misunderstanding, we mention that in neural networks the term has a specialistic meaning referred to as the “attention learning" mechanism [1] introduced in sentence translation, but this is not what we are interested in discussing in the paper.

Let us reflect on the function of a neuron in our brain. With a sort of wordplay, its function coincides with a mathematical function. That one it calculates in producing signals on efferent synapses, in correspondence to those received on afferent synapses. Shortly, the function of a neuron is a mathematical function.

This simple consideration tells us that ANN constitutes an abstract model of the brain, as organized in an integrated system of many ANNs. The notion of function, which is the basis of computation, as it

 doi:10.20944/preprints202406.0757.v1

resulted from Turing's model and McCulloch and Pitt's model, is also the basis of a cognition structure, and it is also the basis of the fundamental mathematical concepts as developed since the new course of modern mathematics. Python language, popular in the AI context, is based on the same notion of function elaborated by Leonhard Euler, and formalized by Alonzo Church [4].

This functional perspective means that each brain competency can be considered a function computed by some neuron network, or by an integrated system of functions computed by subnetworks of the network, possibly organized at various levels. These competencies are formed through a "control mechanism", which applies the learning equations to adapt the network to its usage needs. In this perspective, ANN and ML are crucial concepts in AI and analogously in "natural intelligence".

Let us extend the notion of a neuron by admitting multiple outputs, with a specific transfer function for each output. Therefore, we can consider an entire network as a neuron with many outputs, and for the same reason, a set of networks as a single network. This fact implies a reflexive nature of ANNs, similar to that of a hologram, where a part can be an image of the entire structure. This characteristic is the basis of global competencies typical of advanced cognitive systems.

Related to reflexivity, in neural networks, there is a principle of duality, when the network is associated with a dual meta-network also called an attention network whose neurons, let us call them meta neurons, receive values from neurons of the first network and input signals, and produce outputs that alter the values of the network's synapses. Such a network realizes ANN learning, as depicted in Figure 2, where meta-neurons update weights.

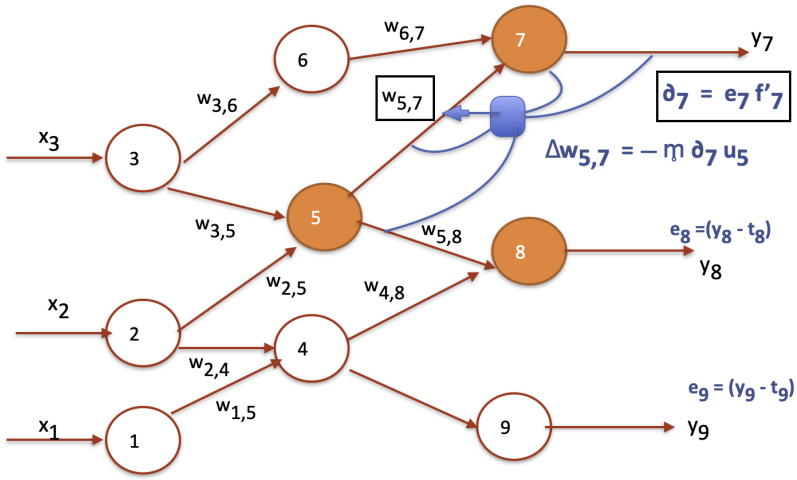

Figure 2. An ANN with 9 neurons and the updating of weight $w_{5,7}$ using a meta-neuron.

From a strictly mathematical viewpoint, the meta-level of attention can be described by considering weights as variables and adding learning equations that oversee their update, along with variables for propagated errors and output errors. The resulting system of equations describes the network along with the learning mechanism carried out by the attention network.

This schematic description of back-propagation intends only to highlight the essence, especially in applications where networks with a large number of composition layers are used, various problems arise that make the presented scheme inadequate. One of the fundamental problems is the "vanishing gradient," where the values that propagate become increasingly smaller as the propagation moves backward, having no significant effect on the updating of weights [2,11].

 doi:10.20944/preprints202406.0757.v1

5. Memory Mechanisms

The realization of memory within neural networks has been considered in various ways and with different solutions. It is often linked to the possibility of considering cycles that maintain a value always present in re-entering as input to a neuron or a subnetwork that produced it as output [19]. However, it is interesting to consider other possibilities and especially to develop mechanisms that can express typical aspects of the memory of natural systems, linking remembering with abstraction. Indeed, in animal cognitive systems, remembering is not just a conservation of data that accumulates, as happens in electronic memories. A memory often extracts certain essential elements, which are transformed into parameters to generate an "image" of the data. Thus, a memory is like a "sketch" of a process or an object in which elements that are not functional to the purposes of the memory are eliminated. In other words, memory is both a generative and an abstraction process simultaneously. Networks, by their very nature, allow these features to be realized. Below we briefly outline some memory mechanisms that align with this intrinsically cognitive perspective of memory.

One realization of neuronal memory consists of translating outputs generated at certain levels of information flows into synaptic weights of appropriate memory neurons that, when activated, produce the stored outputs even without the presence of the inputs that had made them.

To achieve this, a neuron m that memorizes a certain output u_j must be connected with a forward synapse $j \to m$ and a return synapse $m \to j$ placing $w_{m,j} = u_j$. In this way, if an activation input arrives at neuron m, for example, from the attention network, then m generates downstream of neuron j the same information flow that a certain activity had generated in j even in the absence of the inputs that had originated it.

The realization of a memory neuron requires the formation of new synapses and an operation of updating synaptic weights based on values produced by neurons. Therefore, in terms of what was said in the previous section, a meta-level is associated with the primary level of an ANN.

A memory mechanism that avoids the meta-level is to add a synapse $j \to m$ and place in m an identity transfer function that sends the received value to its output. This value, through a cyclic synapse with a unit weight (afferent and outgoing from m) constantly maintains the value received from m as its output-input value. Apart from the introduction of cyclic synapses, this mechanism imposes an additional output variable for each memory of values, thus a cost for input "interface" that becomes prohibitive.

The memorization of an event is, in any case, a "trace" that identifies it regardless of its spatio-temporal occurrence. However, such a definition does not consider a crucial aspect of cognitive systems, namely, that the memory of an event implies a form of abstraction. Indeed, the trace that a system preserves is not an exact copy of it, describing it totally like an identical replica. The preserved trace is often useful if it abstracts from many details. Thus, the same mechanism that underlies understanding, is found in memorization. One understands by "forgetting" what is not functional to a series of logical connections and relations, and, although it may seem paradoxical, one remembers by forgetting many details of the contingency within which an event was placed. Then, the mechanism, previously described, of memory neurons that transform output into a weight (with forward and return synapses) is realized by selecting only some outputs of a network, somehow connected to "interpretation" mechanisms that allow, in addition to a reduction of information, also a greater connection capacity between “preserved" elements. This memory, which is formed under the control of the attention network and with the support of emotional, discerning, and goal-directed levels, constitutes a map of the history of the entire cognitive system and thus constitutes the most significant aspect of the identity of the whole system that hosts it.

 doi:10.20944/preprints202406.0757.v1

6. Comprehension and Knowledge

Besides learning, the attention network can perform functions of stimuli selection, addressing them to subnetworks, coordination, and storing information fluxes.

Combinations of an ANN with the associated meta-network results in a qualitative leap in the functional and evolutionary organization of ANNs. Indeed, this achieves a transition comparable to that from simple Turing machines, each calculating a specific function, to a universal Turing machine within which all functions computed by individual machines can be realized. In more pragmatic terms, the meta-level of an attention network performs functions similar to those of a computer's operating system. Indeed, an operating system coordinates programs that perform specific tasks and manage the machine's resources. Furthermore, if the operating system includes interpreters or compilers of programming languages, it can enhance the machine's functionalities through new programs written in the hosted programming languages.

In the case of ANNs with a meta-level of attention, the notion of programming is replaced by that of training. This means that the same training mechanism that realizes the network, once internalized in the machine (with the attention network), allows the development of new functionalities.

When the attention network connects to networks of discernment and will, supported by stimuli of emotions, motivations, and purposes, it leads to further levels of consciousness and self-control that can not only increase the complexity of the entire system but can also serve as models for complex functionalities of the human mind.

What is the relationship between attention and understanding?

Generally, a system that acquires data realizes mechanisms of understanding when acquiring that data, it integrates them within the system's structure, by reducing them in terms of data already internally available.

In understanding, two opposing and cooperating mechanisms intervene in **assimilation** and **accomodation** [23]. The acquired data is assimilated into the internal structure, trying to keep it as unchanged as possible. At the same time, the internal structure "accommodates" to more completely embrace the acquired data.

In this sense, adding data to a list is the most rudimentary form of understanding because what is added changes nothing of the previous structure, and the data undergoes no transformation that considers the structure that acquires it.

Conversely, if the received data has aspects in common with data already present in the system, the more common aspects there are, the less the system has to add to assimilate the data internally.

Thus, understanding involves analyzing the data and searching for common aspects by abstracting from particular features of single data. In other words, true understanding requires an abstraction that brings out salient aspects while neglecting others, not interesting to its assimilation. This filter reduces the costs of accommodation of the system in assimilating what is understood.

The assimilation-accommodation mechanism naturally connects to interpretation, which seeks the data elements known to the system. In finding known elements, there is an act of recognition peculiar to the interpretative mechanism.

Knowledge has a broader significance than understanding and interpreting. It can involve the "invention" of new elements restructuring internal data according to a deeper logic that connects them.

This situation occurs in science when new theories allow for common explanations of old theories that are not reducible to one another. Knowledge expands horizons with new elements that enable a better understanding of the old ones. In this mechanism, ideal elements [10] are crucial.

An ideation process consists of creating an element whose presence improves knowledge. The typical example made by Hilbert is that of imaginary numbers. The number $i = \sqrt{-1}$ cannot be equal to any real

 doi:10.20944/preprints202406.0757.v1

number because its square is -1, while the square of a real number is always positive. Thus, i is a new type of number, precisely ideal, whose presence allows for the natural calculations done to solve algebraic equations.

Similarly, a physical particle is something that, in certain experiments, produces the measurement of certain parameters. When certain phenomena are clearer and better explained by assuming a particle (electron, proton, neutron, ...), then we have "discovered" that particle.

The only things a physicist measures directly are displacements of points, so any physical process could be entirely described in terms of sequences of displacements (even times are sequences of displacements).

In general, something exists mathematically or physically if its existence produces descriptions more understandable than those that would be made without its existence.

The same applies to notions definable in cognitive systems. We use the term "training" to denote a process by which a system acquires a competency. The term means a series of characteristics of a machine at the end of a certain process, However, it is something that determines new functions in a neural system, that is, a kind of function that produces functions.

All the language of science is based on ideations, and natural language provides linguistic mechanisms to shape concepts generated by ideal elements.

Training is a particular case of understanding because it allows for acquiring a functional correspondence with which cases different from the examples are assimilated at almost no cost of accommodation.

Meaning is often taken as a guarantee of understanding. It is a typical example of an ideal element, postulated as responsible for successful communication. Its presence is only internal to the system that understands, but it is often difficult to define outside the dynamics in which it is recognized.

For example, in an automatic dialogue system [3,14], the meanings of words are represented as numeric vectors of thousands of components (4096 in ChatGPT4). However, such a system never externally provides these representations. In other words, meanings as vectors oversee the proper functioning of the dialogue but remain internal and inaccessible entities. This situation highlights the intrinsic value of ideal elements and the complex relationship between abstraction and levels of existence. Something that exists at a certain level of observation may not exist at another level, but more importantly, not all levels of existence are accessible or fully accessible.

It is impressive but in line with Piaget's speculation [23], that general principles, such as **Abstraction, Ideation,** and **Reflexion**, individuated in metamathematics, are also acting in the organization of cognition structures: Abstraction (Russell [27]), Ideation (Hilbert [10]), Reflexion (De Giorgi [6]: "All the mathematics developed up a given time can become object of the following mathematics").

Surprisingly, in his 1950 essay [29], Alan Turing, discussing machines that learn, considered the aspect of intrinsic ignorance on the part of the trainer of the processes that oversee a learned behavior. This aspect is partially connected to the fact that complex behavior requires the presence of randomness, to confer flexibly to the situations in their specific contexts.

Claude Shannon also sensed the possibility of machines that learn and outlined various possibilities with which a machine can realize this competency [25], among these the general scheme that has emerged in Machine Learning: examples, errors, approval, disapproval. In any case, the profound difference between programming and training is in the autonomy of the latter, which is an internal process escaping any complete control from external. In this intrinsic autonomy perhaps lies the basis of "intelligence."

Although pioneers like Turing and Shannon had intuited the possibility of machines capable of learning, this paradigm emerged gradually and through contributions across the classical computational paradigm [9,12,24]. Backpropagation links learning to classical mathematical themes that date back to Legendre, Gauss, and Cauchy, problems that require the baggage of mathematical analysis, back to Leibniz, surprisingly, the same thinker who had envisaged the possibility of a "Calculus Ratiocinator."

 doi:10.20944/preprints202406.0757.v1

In any case, current experience in Machine Learning confirms that in cognitive systems, when certain levels of complexity are exceeded, there are levels the system cannot access. This is a necessary price to reach advanced levels of complexity and to develop reflective mechanisms that can lead to forms of consciousness. In summary, consciousness must be based on an "unconscious" support from which it emerges.

7. Conclusions

The latest significant advancement in AI and ML has been the development of chatbots with conversational abilities comparable to humans. This development has highlighted another aspect already present in Turing's 1950 article [29], demonstrating its impressive visionary nature. Dialogic competence provides access to the rational and intellectual capabilities of humans. It is no coincidence that the Greeks used the same term (Logos) to denote both language and reason. Furthermore, it is no coincidence that the logical predicative structure of Aristotle's syllogism is based on a wholly general grammatical schema. The endeavor that led to the 2022 launch of the public platform ChatGPT3.5 (Generative Pre-trained Transformer) by Open AI is undoubtedly a historic scientific and technological enterprise.

Many factors contributed to the success of ChatGPT, including immense computing power and a vast availability of data for training development. From a scientific viewpoint, a fundamental element was the development of Large Language Model (LLM) systems [3,14] with the introduction of the Transformer-based perspective. Before achieving dialogic competence, these systems learned to represent the meaning of words and phrases by transforming them into vectors of thousands of dimensions over rational numbers. This competence provides the abstraction capacity on which a genuine construction of meaning is based. Although such meanings are certainly completely different from those of a human mind, it is entirely reasonable to assimilate them to abstract representational structures that emerge through a process of acquiring appropriate linguistic competencies (comparisons, analogies, differences, similarities, etc.).

In [16], a logical representation system for texts was defined and described to ChatGPT3.5 through various conversations, and the system was questioned, revealing a satisfactory ability to acquire and correctly use the method. The experiment suggests a possibility that can help to investigate applications of chatbots in contexts of complex interactions and in developing deep forms of conceptual understanding.

A detailed analysis of intellectually rich dialogues will be a type of activity of great interest shortly. In the context of dialogues with ChatGPT4 (an advanced version of ChatGPT3.5, accessible with a payment for the service), a possibility has emerged that is interesting to report as a reflection on possible extensions of the competencies of these artificial systems.

Even though the system can track the developed discourse and the concepts acquired during its course, at the end of it, there is no stable trace of the meanings of built-in interaction with the human interlocutor. In other words, the understanding acquired in a dialogue does not leave a trace in the semantic network that the chatbot possesses. This observation could suggest mechanisms, that, when appropriately filtered to ensure security, confidentiality, and adherence to ethical behavior principles, could adapt to individual users and import conceptual patterns and typical aspects in a stable manner that could make the cognitive development of these systems more fluid and personalized. Similarly to the algorithms that operate on many current platforms, which learn users' tastes and preferences to better adapt to their needs, one could think of forms of "loyalty" that would constitute an advantage for the interlocutors, but especially mechanisms of "induced" training useful for the cognitive development of the systems.

Continuing to develop this perspective, one would obtain a population of variants of the same chatbot that express different modes and levels of development corresponding to the reference interlocutors. The variants of higher intellectual levels would exhibit certain competencies more significantly. In an experimental phase, a restricted group of enabled interlocutors could be observed, and the trainers could

 doi:10.20944/preprints202406.0757.v1

control the phenomenon's effects and draw from it useful elements to understand the logics that develop and the potentials that open up. The results obtained in ths way are comparable to diversified training for the same type of system. It would be as if different instances of the same chatbot were trained on the texts of the conversations with their respective interlocutors.

Suppose a certain system is exposed to a course in Quantum Physics, or perhaps to more courses in Physics, or another specific discipline. Training of this type is much more complex than that based on the assimilation of texts and supervised training developed by even very experienced trainers. To achieve interesting results, the system should have forms of "gratified attention" more complex than simple approval or disapproval, but the form of learning that would be obtained could be so profound as to produce specialists in scientific research.

As evidence of this idea, we can report that the formulation of Learning Equations given in the previous section, was obtained by starting from the description of back-propagation given in [20] by a long conversation with ChatGPT, which helped the author in deriving the equational synthesis expressed by (4, 5, 6).

To conclude, it seems appropriate to recall Turing's concluding remarks in his visionary essay [29] on thinking machines: "We can only see a short distance ahead, but we can see plenty there that needs to be done".

References

1. Bahdanau, D., Cho, K., Bengio, Y., Neural Machine Translation by Jointly Learning to Align and Translate, *arXiv:1409.0473* (2014)
2. Basodi, S., Ji, C., Zhang, H., Pan, Y. Gradient Amplification: An Efficient Way to Train Deep Neural Networks, *Big Data Mining and Analytics*, 3, 3 (2020)
3. Brown T. B. et al. Language Models are Few-Shot Learners, *NEURIPS*, 33, 1877–1901 (2020)
4. Church, A., A note on the Entscheidungsproblem, *Journal of Symbolic Logic*, 1, 40-41 (1936)
5. Cybenko, G., Approximation by superposition of a sigmoid function *Matematics of Control, Signals, and Systems*, 2, 303-314 (1989)
6. De Giorgi, E. *Selected Papers*, Dal Maso G., Forti M., Miranda M., Spagnolo S. (eds.), Springer-Verlag, Berlin & Heidelberg (2006)
7. Gelb, W., Kirsch, B., The Evolution of Artificial Intelligence: From Turing to Modern Chatbots, *Tulane University, Archives*, https://aiinnovatorsarchive.tulane.edu/2024/ (2024)
8. Goodfellow, I., Bengio, Y. Courville A., *Deep Learning*. MIT Press (2016)
9. Hebb, O., *Organization of Behaviour*, Science Editions, New York (1961)
10. Hilbert, D., Über das Unendliche, *Math. Ann.* 95, 161–190 (1926).
11. Hinton, G. E.; Osindero, S.; Teh, Y., A fast learning algorithm for deep belief nets, *Neural Computation*, 18 (7): 1527–1554 (2006)
12. Hopfield, J.J., Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. USA* 79, 2554–2558 (1982)
13. Hornick, K., Stinchcombe, M., White, M., Multilayer feedforward networks are universal approximators, *Neural Networks*, 2, 359-366 (1989)
14. Kaplan, J. et al. Scaling Laws for Neural Language Models, *arXiv*, 2001.08361 (2020)
15. Kelley, H. J., Gradient Theory of Optimal Flight Paths *ARS Semi-Annual Meeting, May 9-12, 1960*, Los Angeles, CA (1960)
16. Manca, V. Agile Logical Semantics for Natural Languages. *Information*, 15, 1, 64 (2024)
17. Manca, V., Bonnici, V., Life Intelligence, in *Infogenomics* (Ch. 6), Springer (2023)
18. McCulloch, W., Pitts, W., A Logical Calculus of Ideas Immanent in Nervous Activity. *Bulletin of Mathematical Biophysics*, 5, 115–133 (1943)

 doi:10.20944/preprints202406.0757.v1

19. Minsky, M., *Computation. Finite and Infinite Machines*, Prentice-Hall Inc. (1967)
20. Mitchell, T., *Machine Learning*, McGraw Hill (1997)
21. Nielsen, M. *Neural Networks and Deep Learning*. Online (2013)
22. Neumann (von), J. *The Computer and the Brain*, Yale University Press, New Haven, Coo (1958)
23. Piaget, J., *L'epistemologie Génétique*, Presses Universitaires de France, Paris (1970)
24. Rosenblatt, F., The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386–408 (1958)
25. Shannon, C. E., Computers and Automata, *Proceedings of the Institute of Radio Engineers*, New York, 41, 1234-1241 (1953)
26. Rumelhart, D. E., Hinton, G. E., Williams, R. J., Learning representations by back-propagating errors, *Nature*, 329, October 9 (1986)
27. Russell, B., Whitehead, A. N., *Principia Mathematica*, Cambridge University Press (1910-13)
28. Turing, A. M., On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, 58: 230–265 (1936)
29. Turing, A. M., Computing Machinery and Intelligence, *Mind*, London, N. S. 59,433-460 (1950)
30. Werbos, P., Backpropagation Through Time: What It Does and How to Do It. *Proceedings of the IEEE*, 78 (10), 1550–1560 (1990)
31. Wiener, N. *Science and Society*, Methodos, Milan, 13, Nn, 49-50, 3-10 (1961)