

Andrea Valle*

From grammar to text: a semiotic perspective on a paradigm shift in computation and its usages

<https://doi.org/10.1515/sem-2024-0187>

Received October 30, 2024; accepted December 26, 2024; published online January 23, 2025

Abstract: Programming is a relevant semiotic activity, resulting in millions of lines of written code: the whole digital revolution is still rooted in writing as a semiotic activity. In relation to this, AI applications based on deep learning do not present particular features. They are standard computer programs relying on the von Neumann/Turing architecture. Yet there is an interesting epistemological difference. A distinction can be made between classical programming and machine learning. As the task for programming is always problem solving, in classical programming, the programmer has to input rules and data in order to gather answers in output. A machine learning approach requires a different epistemological wiring: the programmer inputs data and the required answers, while the software learns or discovers the rules. These two approaches to programming can be characterized from a semiotic perspective by referring to the couple “grammar” versus “text” and “allography” versus “autography.” A grammar defines a set of rules to be applied so that an output is generated that is formally consistent with the prescribed rules. Rather, a text acts as an example from which to infer regularities in order to generate a new text. This epistemological shift on the computation side is coupled with an analogous one on the user side. As data, that is – semiotically – texts, are the driving force, users have to focus on sets of examples in order to cope with the algorithms. The contribution discusses this shift by taking into account the relative changes in agency.

Keywords: grammar versus text; computation; machine learning; autography versus allography; deep learning

We plan our happenings carefully to be sure that they are thoroughly spontaneous.

– (Ong 1982: 133)

*Corresponding author: **Andrea Valle**, CIRMA/StudiUm, Università di Torino, Torino, Italy,
E-mail: andrea.valle@unito.it

 Open Access. © 2024 the author(s), published by De Gruyter.  This work is licensed under the Creative Commons Attribution 4.0 International License.

1 Introduction: epistemological issues

We are certainly experiencing a moment of amazement concerning AI, both on the scientific and the humanistic side. This situation has led to the superimposition of deep learning not only with machine learning (of which the former is a subset¹), but *tout court* with AI. Nonetheless, the debate over neural networks (NN; originally the main “model architectures” used within the deep learning paradigm) undoubtedly has a long history in the field of AI. The issue is eminently epistemological, and concerns the nature of the explanation. Turner’s observations on NNs are still relevant:

The interpretation of inputs and outputs is usually pre-specified by the researcher, but this is not the case for units in hidden layers and for these no clear interpretation will be possible in most cases, even after learning, due to the distributed nature of the representations. The advantage of this is that the researcher does not have to make unnecessary assumptions about internal representations. A disadvantage is that it can make it more difficult to explain how a model does what it does and hence more difficult to extrapolate from it to the real world. (Turner 2002: 33)

Poole and Mackworth’s comprehensive introduction to AI (now in the third edition) traces deep learning back to one of the (many) aspects of computational agency, namely, the learning component, and possibly the generation of specific behaviors as a function of the former. The authors declare their reference paradigm in the second part of the title of their book “Foundations of Computational Agents” (Poole and Mackworth 2023). Thus, the theoretical framework refers to the idea of a specific form of subjectivity, that of a computational agent. Starting from such a framework, it is not surprising that Poole and Mackworth provide the following – somewhat *tranchante* – evaluation on the most up-to-date deep learning technology, so-called Large Language Models (hereafter LLM): LLMs “are controversial because of the claims that are made. In particular, there is a lot to be impressed with if you set out to be impressed, however there is a lot to be critical of if you set out to be critical” (Poole and Mackworth 2023: 365).

Indeed, agency can be approached from multiple perspectives, e.g., philosophical, phenomenological, anthropological. Hence it is a particularly interesting terrain of dialogue (which has not yet occurred) between AI and semiotics. As a cursory example, a computational agent can be characterized in terms of, e.g., plans, beliefs, knowledge (Poole and Mackworth 2023). On the other hand, in semiotics, the notion of “actant” (Greimas and Courtés 1979) concerns precisely an idea of subjectivity

¹ In relation to machine learning, the introductory text by Müller and Guido (2017) dedicates very little space to neural networks.

oriented towards value; the issue of plans is discussed in terms of narrative programs; the aspect of beliefs takes various declinations, e.g., in terms of value or passion; knowledge can be thought of as competence.

In his discussion on the history and current debate in AI, Lieto (2021) investigates a specific problem, certainly near to semiotic interests, and still related to agency: the verification of the cognitive plausibility of computational architectures. The topic is controversial, and for example Lieto (2021: 21) clearly distinguishes between functional and structural models of the mind. The question is in fact that of subjectivity: *what do we think these models are?*

In the following, my attempt will be to discuss NNs in relation to some transformations in agency that could lead to a semiotic shift in wider terms, that is, in relation to culture. In order to finally take into account a specific feature of agency (feedback, see later), it seems necessary to briefly discuss deep learning in the context of machine learning.

2 An interpretation of neural networks

A clear and understandable description of NN machinery is provided by Chollet (2021). On the one hand, Chollet's book seems to have an applicative vocation (declared in the title). In reality, together with this aspect, it provides an interesting epistemological interpretation of deep learning, also including some important philosophical considerations in the final chapter.

What is a NN? In essence, in a NN, data is organized into n -dimensional structures. If $n = 1$, the data space is a vector (that is: a sequence or a list), and each data point requires an index to be addressed. If $n = 2$, it is a matrix: each data point can be identified by two coordinates in a Cartesian space. If $n > 2$, then it is a so-called "tensor." If $n = 3$, a tensor can still be visually displayed, otherwise the data space is an abstract one in which each point is defined by n coordinates. This spatial dimension is not incidental.

A NN is a sequence of multiple layers, each of which processes tensors. These tensors represent the multi-dimensional data passed between the layers. Each layer is associated with a transformation function that operates on an input tensor and transforms it into another tensor, which is passed to the next layer. A resulting architecture can complexify this scheme, e.g., including many layers or providing different activation and optimization functions, but in the end we always have transformations between tensors. For example, Figure 1b shows the VGG16 network for extracting features from images: it is a chain of transformations between adjacent tensors.

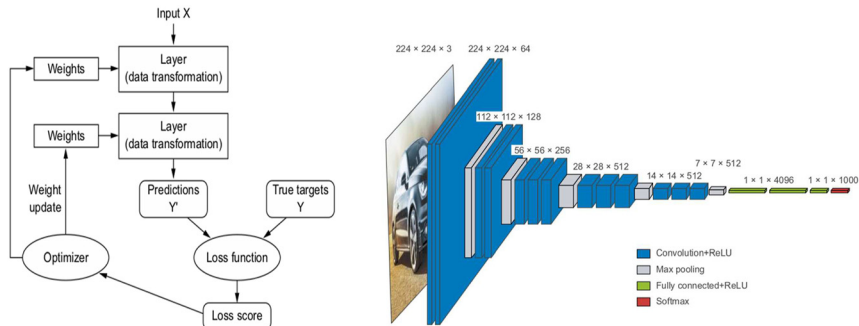


Figure 1: NN architectures (a) NN architecture parametrized by weights and including loss function measuring the network's output; (b) the VGG16 architecture (from Chollet 2021: 9, 250).

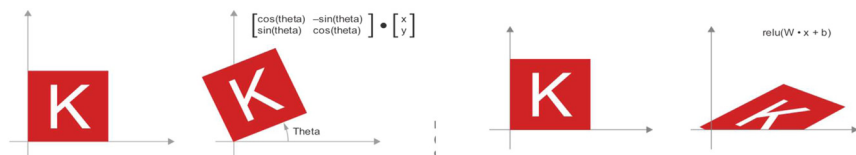


Figure 2: Geometric transformations (a) Rotation of a matrix as a dot product; (b) affine transform followed by ReLU (from Chollet 2021: 45 and 27).

Before considering learning, how can we interpret such an information architecture? Chollet (2021: 47) provides a relevant geometric interpretation. On the one hand, every tensor can be seen as an n -dimensional space. As can be seen from Figure 2a, each function in a layer geometrically transforms the input data (here: a matrix) in some way (here: by a rotation).

The second transformation (Figure 2b) is particularly interesting. The function is a ReLU (“Rectified Linear Unit”). Trivially, it is a threshold function: negative values are set to 0. The presence of nonlinear functions like this is fundamental in NNs, because successively applying only affine layers is equivalent to applying a single “aggregated” affine layer, which thus reduces the learning capacity of the models. ReLU is not bijective: after its application, we have lost information, that is, we are not able to apply an inverse function that takes us to the previous layer. Different values are mapped to the same value, e.g., -3 and -5 are mapped to 0; hence, given the value 0, we cannot know whether it comes initially from a -3 , -5 , or even another negative value.

In essence, Chollet notes that deep learning, by chaining nonlinear spatial transformations, focuses on doing an uncrumpling operation on a very crumpled surface like, e.g., a paper ball.

What a neural network is meant to do is figure out a transformation of the paper ball that would uncrumple it ... With deep learning, this would be implemented as a series of simple transformations of the 3D space, such as those you could apply on the paper ball with your fingers, one movement at a time ... Uncrumpling paper balls is what machine learning is about: finding neat representations for complex, highly folded data manifolds in high-dimensional spaces (a manifold is a continuous surface, like our crumpled sheet of paper). (Chollet 2021: 47)

This geometric interpretation has also the merit of completely eliminating the cerebral mystique of neural networks. In fact, one point that has remained since the first proposal of computers is that of the electronic brain. In other words, as Lieto (2021) says, traditionally there has been a *rhetoric of structurality*: the argument, which is eighty years old,² is that the computer is an electronic brain, and therefore by analogy, just as the brain is somehow implicated in human or animal subjectivity, then an electronic brain implies subjectivity. This does not mean that there are not somewhat surprising effects in the output of these systems, and certainly interesting ones, but in the end a neural network may be complicated but not so complex. Which is perhaps the true charm of these architectures.

3 Feedback

How does a NN learn, thus becoming effective? The final output is evaluated by a loss function, that compares the output to a real reference, and then passes the result to an optimization function. The latter in turn changes the parameters (W and b in the ReLU of Figure 2) associated with the layer functions. What happens in practice is that the weight adjustment is proportional (in magnitude) to the error in the prediction. So, basically, if the model makes a large mistake, then it needs a larger adjustment than if it makes a small mistake. The mistake is the difference between the prediction and the real reference (ground truth), i.e., the “score.”

As Chollet notes, this idea reinforces the geometric interpretation of NNs. In fact, the assumption is that these spaces (n -D surfaces) are continuous, like surfaces (that is, they are mathematically differentiable): if they are surfaces, one can move continuously on them. These surfaces can be adjusted, “tuned,” so to say. This is the idea of a “gradient,” a term largely used in the NN jargon, meant as a generalization of a derivative.

This learning mechanism is based on a feedback idea. In a feedback system, the output feeds back into the input (Figure 3a). The idea of “re-entry” can be described in various ways. For example, formally in recursive terms, as in the following Python

² In the first proposal of a computer architecture, Von Neumann (1993 [1945]) is taking into account a “neuron analogy,” following McCulloch and Pitts paper from 1943.

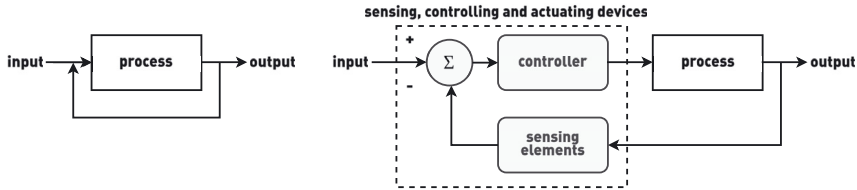


Figure 3: Feedback (a) A general feedback process; (b) feedback resulting from environment measuring.

example, which defines a *fibo* function, a recursive implementation of a Fibonacci sequence generator:

```
1 def fibo (a = 0, b = 1, seq = [], i = 0, mx = 20):
2   if i < mx:
3     c = a + b
4     seq.append(c)
5     fibo(b, c, seq, i+1, mx)
6   return [0,1]+seq
```

Inside it, as long as the overall sequence *seq* has fewer than 20 elements (*mx*, see line 2), it continues to call itself passing its output values as parameters for the new call (see line 5).³

The diagram in Figure 3b, however, indicates a feature of feedback systems that is not described by pure recursion. The output is not *calculated* (as in *fibo*), rather it is *measured* (see “sensing elements”). The output is information external to the system that produces it, hence it has to be measured. In this sense, the output is a full part of the environment surrounding the system.

With the introduction of feedback, we are finally back to agency. The very idea of feedback was first introduced by Rosenblueth, Wiener, and Bigelow in their famous 1943 paper, which is the epistemological basis of cybernetics. The authors discuss the idea of feedback as a measure of effectiveness towards a goal, and indicate that teleology is a fundamental aspect of behavior. Teleology is synonymous with “purpose controlled by feedback” (Rosenblueth et al. 1943: 23), in particular with negative feedback as an adjusting, tuning operation, like the one done by the optimization function in a NN. Here Rosenblueth, Wiener and Bigelow propose a clear definition of subjectivity that seems interesting for semiotics. It is based on the idea of purpose-directed agency. It is an operational, formal, and thus very general definition. It

³ This idea is often exploited on a textual level. One can consider *mise en abyme*, the so-called Droste effect, etc.

seems to me consistent with Sebeok's (2001) thesis, according to which semiosis is typical of every living being as the latter is oriented at least towards self-preservation. This orientation has been taken into account, while semiotically reconsidering a phenomenological perspective, by Fontanille (2006: 61) as he proposed, starting from perception, to describe the presence of subjectivity as based on the relation between a source and a target (mediated by a control instance). In *La struttura assente*, Eco urges us to replace the (comfortably semiotic) question "Who is speaking?" with a very different one: "Who is dying?" (1968: 357). Even if Eco is mostly interested in properly cultural facts, this observation can be seen as a way to characterize semiotic work as a feature of the living as a sign making activity. For Eco, the subject leaves traces of her/his semiotic work that modify the semiotic (as historically given) landscape, so that they have to be taken into account by every new sign production: interpretation leaves "footprints" and "cart-trails" that "modify the explored landscape" (Eco 1976: 29), thus introducing a feedback loop, as each interpretation modifies the semiotic landscape yet to be explored. On the other hand, the idea of feedback is implicit in von Uexküll's (2010 [1934]) idea of *Umwelt*, based on a closed cycle between action and perception passing through the environment.

Being formal, the cybernetic definition is even more general. In fact, for Wiener certain forms of purposeful behavior certainly apply to technological devices. An interesting point in relation to NNs, however, is that the feedback mechanism (called "backpropagation" in NN jargon) is not biologically plausible if we take into account neural biological networks: "as Francis Crick ... famously pointed out, backpropagation requires that information be transmitted backwards along the axon. However, this phenomenon has never been observed in natural neural architectures and, therefore, cannot be considered a realistic mechanism" (Lieto 2021: 32).

To conclude about the neural metaphor, a computational neural network:

- 1) performs a set of chained transformations on successive tensors;
- 2) requires a mechanism that has been proposed for the formal description of agency as a purposive behavior (feedback) that, at the low, neuronal level (originally inspiring NNs), is not biologically plausible.

In short, in order to inject a NN with an "intelligent" behavior we have to rely on a theoretical/formal construct (feedback) that has been proposed to describe high-level agency in humans, animals, and machines.

4 Paradigm shifts in computation

Programming (here meant both as organizing computational programs, Abelson et al. 1996, and as code writing, Seibel 2009) is indeed a relevant semiotic activity,

spawning thousands of languages in fifty years and resulting in millions of lines of written code. The whole digital revolution in this sense is still rooted in writing as a semiotic activity. Programming languages are semiotic systems that show an interesting twofold orientation (Valle and Mazzei 2017): on one hand, towards the machine that has to perform calculations based on strictly formal instructions; on the other, towards the programmer and her/his community, so that code can be also thought as a form of literary writing (“literate programming,” Knuth 1992). An interesting semiotic feature in programming languages is that this double interpretation (on both machine and human sides) is not only linked, but has to be consistent between the two sides. Yet, it allows a relevant degree of semiotic freedom, providing room for, e.g., stylistic/rhetorical operations on the human side while still being formally consistent on the machine side (see, e.g., Valle 2020).

Observed from this dual perspective, AI applications based on machine/deep learning do not present particular features (Poole and Mackworth 2023). They are standard computer programs relying on the Turing/Von Neumann (respectively from 1936 and 1945) architecture (Gabrielli and Martini 2010). We are always surprised by the randomness, hence the creativity, of deep learning systems. At the bottom, they all run on standard, deterministic machines. So, on one hand, nothing changes. But this is just half of the story.

Yet, it has been observed that deep learning, and more generally, machine learning, is the basis of a paradigm shift in computation, as shown in Figure 4 (Chollet 2021: 4).

In “classical programming” results are produced starting from the formulation of algorithms (rules) and the available data. In “machine learning,” the system receives data and expected answers, and generates the rules (by learning). In terms of Peircean logical operations (Peirce 1878), it could be said that the emphasis in classical programming is on deduction (a certain result strictly follows from a rule) while in machine learning is on abduction (a certain fact is proposed as a result of a newly established rule). Thus, machine learning can be thought of as a set of technical methodologies for the automation of abduction.

Figure 4 does not include feedback. The point is that the rules in classical programming are defined while those in machine learning are progressively inferred: “a machine learning system is trained rather than explicitly programmed” (Chollet 2021: 4). To sum up, in classical programming the programmer starts from rules while in machine learning s/he gathers them from data. In machine learning,

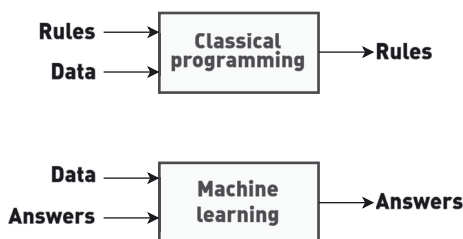


Figure 4: Classical programming versus machine learning paradigms (from Chollet 2021: 4).

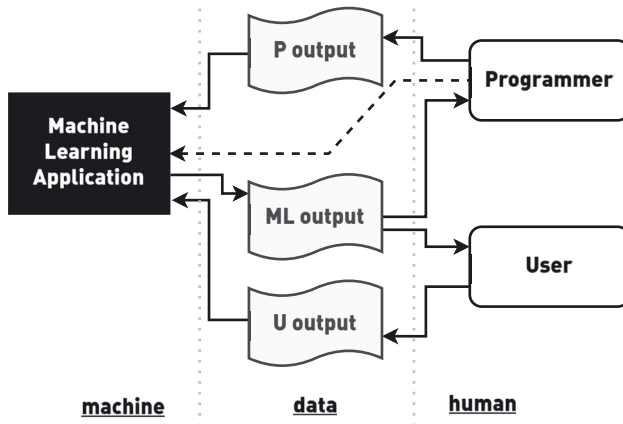


Figure 5: Communication in machine learning between software and user.

“classical” rules are indeed present, but act as meta-rules (methodologies to be used to get rules from data).

A recent buzzword on the web is “feedback economy.” It seems that there is a *Zeitgeist*, an *esprit du temps*, that has its pivot in feedback. This feedback-based communication is indeed a common feature of, e.g., social networks and electronic markets, but with deep learning it is now also built into applications. The result is a situation like the one in the Figure 5 in which the dotted line represents classical programming (required to program the machine learning system), while the others indicate a circuit that always passes through the data. The data is properly the environment external to the subjects (programmers/users). As it happens in ecosystems, this communication is open, not fully controllable by the agents, not deterministic, it may be disturbed (Maran 2020: 12). In the programming/training phase, the programmer must inspect the output of the ML algorithm, to verify the behavior of the machine “in the wild” (comparing it with the empirical nature of the data). The goal is the “tuning” of the algorithm. The user can only treat communication with the machine in terms of an interaction through the data: those s/he receives (ML output) and those s/he provides (U output, e.g., a prompt). This interaction modifies the results produced by the machine (see, e.g., ChatGPT or DALL-E⁴). Therefore, the new paradigm, which also insists on the classical one due to the stratification of the symbolic dimension in computers, could be defined as “feedback computation.” It is based “circuitually” (Basso Fossali this issue) on a relocation

⁴ See e.g.

<https://help.openai.com/en/articles/5722486-how-your-data-is-used-to-improve-model-performance>.

of the programmer's position. This produces a sort of blurring between user and programmer, but not a complete one.

A couple of suggestions follow:

- 1) Metaprogramming. If a program is a string of symbols, and, e.g., LLMs produce strings of symbols, is it possible to generate programs with LLM? Maybe not. According to Chollet (2021: ch. 14), there are two basic cognitive dimensions, one geometric (based on proximity, "value-centric analogy"), the other structural (based on exactness, "program-centric analogy").⁵ NNs excel at the former, while metaprogramming requires exactness. In fact, for this specific purpose, Chollet hypothesizes the use of other computational architectures rather than NN (e.g., genetic algorithms). There is actually no formal logic built in LLM (Mirzadeh et al. 2024). Provided that code is consistently reused ("cannibalized" in the coders' jargon, via examples, tutorials, sites like Stack Overflow⁶), good, even impressive, results can be obtained via LLM as there is a large database (e.g., Austin et al. 2021). But it comes without a warranty: it is not proved to be working. The point is that an algorithm (as implemented in a program) must be correct (truth), not approximately correct (verisimilitude, likelihood).
2. Prompt as programming. Romele⁷ has suggested that prompt can be thought of as pseudo-code. Indeed, they both share a proximity to human language and the idea of describing the control of the machine. But pseudo-code is still related to instructing the machine (in abstract terms), while prompting is a way to train/tune/inspect the machine.

5 Text versus grammar, autography versus allography

How can we think in terms of semiotics of culture about this *Zeitgeist* that drives towards feedback computation?

The previous two approaches to programming can be characterized from a semiotic perspective by referring to the couple "grammar" versus "text" (Eco 1976;⁸

5 This indeed reminds us of autography/allography in Goodman (1968). See Section 5 for a discussion.

6 <https://stackoverflow.com/>.

7 Communication "L'IA générative comme situation expérimentale: incertitudes symboliques et matérielles," Séminaire International de Sémiotique à Paris 2023–2024, December 20, 2023, <https://www.youtube.com/watch?v=pPfUVK4cBZY> (accessed 26 December 2024).

8 In his theory of sign production, Eco (1976) takes into account Lotman and Uspenskij proposal by defining an opposition between "replica" (based on grammar) and "invention" (based on text), see Valle 2017 for a discussion.

Lotman and Uspenskij 1973). A grammar defines a set of rules to be applied so that an output is generated that is formally consistent with the prescribed rules. Thus, a grammar approach focuses on the prescription of rules. Rather, a text acts as an example from which to infer regularities in order to generate a new text. Cultures based on grammar rely on rules to be respected, the latter being unmodifiable at the cost of a radical transformation of the culture itself. Text-based cultures operate by a sort of continuous drift, by chaining texts that are recursively taken again into account as examples to be inspected in order to produce new texts, in a feedback loop. Lotman and Uspenskij observe that textual cultures are based on expression, while grammar cultures on content, as content is thought to be defined somewhere else. One can easily observe with Peirce that a grammar relies on deduction (rules are given before), while a text on abduction (rules, that is generality, is inferred). This can be translated in Goodman's (1968) terms by indicating that textual cultures are "autographic" while grammar ones are "allographic." The couple autography/allography is presented by Goodman in relation to a theory of notation referred to the arts.

An established art becomes allographic only when the classification of objects or events into works is legitimately projected from an antecedent classification and is fully defined, independently of history of production, in terms of a notational system. Both authority and means are required; a suitable antecedent classification provides the one, a suitable notation system the other. (Goodman 1968: 198)

On the other side, an autographic art has no notation, as it is not possible to define a general type which concrete examples can be referred to. In terms of artistic practices, one may think "prototypically" about the autography of painting versus the allography of classical music written in a score, in which every performance, despite its concrete uniqueness, is still related to the score (the notation defining the identity of the work). Although related to the arts, the pair is relevant to define a general difference in semiotic regimes of existence (see Basso 2003 for a discussion). In autography, the semiotic object presents itself as unique, as a single instance linked to a specific, concrete manifestation. In allography, it can instead be traced back to an instance of a formally defined class: this formal definition therefore allows a notation of the object. In allography, we thus have a notation that defines types to which concrete objects can be traced back as tokens. If we consider Goodman's reference to "antecedent authority and classification" for allography, the relationship with grammar (vs. text) becomes evident.

Back on the computation side, generative AI applications based on machine/deep learning offer users a (huge) variety of results given the same input, thus triggering on the user side practices based on examples. As said, this epistemological shift on the computation side is coupled with an analogous one on the user side. As data are the driving force, users have to focus on sets of examples in order

to cope with the algorithms. Hence the relevance of big data: corpora, archives, sets (see, e.g., Dondero 2020 on images). It is thus possible to say that, in terms of semiotics of culture, we are experiencing a regime change towards textuality (meant *à la* Lotman and Uspenskij) and autography. These two features (or two ways to describe the situation) are typical of oral cultures versus grammar/allography of cultures based on writing. Oral cultures are mostly text-oriented as there is no material support for objectifying rules, written cultures are more grammar-oriented. This textual drifting is indeed not new: McLuhan (1964) has proposed a new orality related to electronic media (telephone, radio, TV, audio – and then video– recording) leading to a new (and at the same time old) sense of participation, extended much beyond the physical one, thus transforming the world in a “global village.” This orality is “secondary” as it is “a more deliberate and self-conscious orality, based permanently on the use of writing and print, which are essential for the manufacture and operation of the equipment and for its use as well” (Ong 1982: 133, see Pettitt 2007). In the words of Ong:

Secondary orality is both remarkably like and remarkably unlike primary orality. Like primary orality, secondary orality has generated a strong group sense, for listening to spoken words forms hearers into a group, a true audience, just as reading written or printed texts turns individuals in on themselves. But secondary orality generates a sense for groups immeasurably larger than those of primary oral culture – McLuhan’s “global village.” Moreover, before writing, oral folk were group-minded because no feasible alternative had presented itself. In our age of secondary orality, we are groupminded self-consciously and programmatically. The individual feels that he or she, as an individual, must be socially sensitive. Unlike members of a primary oral culture, who are turned outward because they have had little occasion to turn inward, we are turned outward because we have turned inward. In a like vein, where primary orality promotes spontaneity because the analytic reflectiveness implemented by writing is unavailable, secondary orality promotes spontaneity because through analytic reflection we have decided that spontaneity is a good thing. We plan our happenings carefully to be sure that they are thoroughly spontaneous. (Ong 1982: 133)

Indeed, the internet has provided a sort of prototypical configuration of the global village, and secondary orality is a common feature of social networks, with the uninterrupted flow of interactions that characterizes them. It might be said that with generative AI secondary orality is now built into applications, as the latter participate in creating a user-centered flow of examples based on a continuous remix of pre-existing cultural elements. It is indeed curious that such a shift towards textuality, example, autography at the higher level happens thanks to symbolic machines based on formal, strictly allographic, grammatical, rule-based systems. At the same time, it is exactly what anticipated by McLuhan and Ong in defining secondary orality as a post-typographic situation. This secondary autography, resulting in a flow of texts, is “planned carefully” (to speak with Ong) by grammar-based, allographic machines.

6 An example: *Cursed AI*

I would like to conclude with an example, from the Facebook group named *Cursed AI*. The latter is a group oriented towards the bizarre,⁹ and includes images but also comics, and sometimes text, created agnostically with all available software. Figure 6 shows a screenshot of a tiny part of the media archive. It is immediately evident that the images uploaded by users follow one another, organized in systems of variations on a thematic/figurative/plastic basis, with a clear idea of a progressive drift. While there are few *hapax legomena*, mostly it is a chain of pseudo-regularities. But the point is not only that the machine displays a human-proposed prompt subject to various drifting. There is properly an interaction.

This specific relation between a certain machine behavioral state and the users might be thought as a form of “semethic interaction” (Hoffmeyer 2008). Semethic interaction is the detection of regularities that is abducted as a rule coupling the spotted, reoccurring item as the expression of a content.

I have called this pattern of interaction semethic interaction (from the Greek, semeion = sign + ethos = habit) ... Whenever a regular behavior or habit of an individual or species is interpreted as a sign by some other individuals (conspecific or alter-specific) and is reacted upon through the release of yet other regular behaviors or habits, we have a case of semethic interaction. (Hoffmeyer 2008: 189)

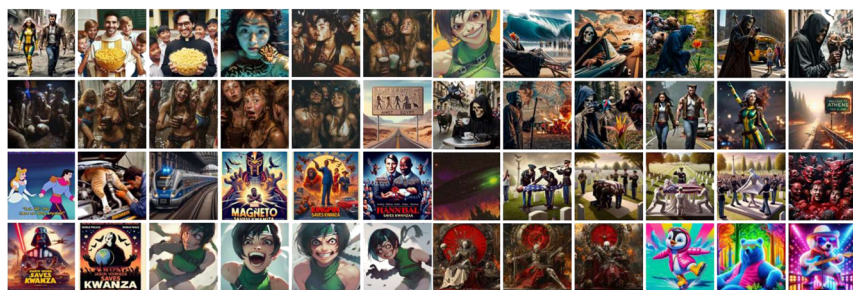


Figure 6: A snapshot from Cursed AI Facebook group media archive.

⁹ “Beware, these creations may haunt your dreams and unravel your sanity. Step into the eerie world of AI-generated cursed art, where machines possess powers to create twisted and terrifying masterpieces. Join our community of art lovers with a taste for the strange and share your twisted creations. These disturbingly beautiful images crafted by AI will leave you questioning the very nature of technology and its place in our world. Enter at your own risk.” (<https://www.facebook.com/groups/cursedaiwtf/about>).

While the term is used in the context of living beings, yet it seems interesting to investigate it cybernetically, in the context of generative AI. Regularities in the machine behavior, not strictly formalized, are detected by users and used to interact with the machine itself. But on the other side, user behavior is captured by the same machine that capitalizes it in order to adjust its behavior, thus leading to a multi-layered semiotic interaction.

The first example is linked to the “yellow coolant” (Figure 7). Since Autumn 2023, user Phil Barber has persistently posted rather bizarre images that refer to urine through a politically correct term in order to fool prompt censorship: yellow coolant. To his taste, often in relation to cars. By obsessively repeating the posts in the group he somehow triggered an imitative reaction, a semiotic interaction. This thematic/figurative dimension has become significant. The theme “yellow coolant” in its entirety has become an expression of possible other contents: it has become an object of recognition in terms of Eco’s theory of sign production (Eco 1976; see Valle 2017). This semantic unit is multimodal: it includes a visual part, spawning over other represented media (advertisements, toys, vintage videogames), but also the linguistic prompt itself, including the name of the author, who is honored in various images, e.g., on a soda can or on a strange custom liquid.

The second thread started with an image tagged as “Kandahar circa 1923.” The result is the construction of a truly surrealist collective imagination of an alternative



Figure 7: Some examples for the “yellow coolant” thread.



Figure 8: Some examples for the “Kandahar 1923” thread.

post-First World War Kandahar. Semethic interaction can be recursive. So there is a thread inspired by “The giant [SOMETHING] of Kandahar circa 1923” that has been widely (and wildly) developed (with sausages, TVs, lego men, golden retrievers, etc.; see Figure 8).

What is the role of the machine? One possibility is to provide additional features on which humans can further elaborate. Yet, in the previous examples deep learning generative systems can be thought of as a sort of displaying agent: they acted mainly as a support. The case of “clungus” is interesting. Most image AIs treat text as an image. In short, there is no allographic status (writing) versus autographic (image): everything is autographic, i.e., image. As an example, in Figure 9a, the lettering of this AI-generated geographic map of Victoria, Australia (from the same Facebook group) shows a sort of continuum from graphic, unreadable signs to alphabetic, readable ones (a continuum clearly visible in the detail of Figure 9b). So, unreadable writings can emerge in generative AI, but also readable ones. In a case like the latter, the AI system invented the word “clungus.” The name, even if proposed with a textual drift in various forms (e.g., “dlungus” or “cluingus”), entered in the community, mainly associated to meat with an organ-based semantic spectrum (Figure 10¹⁰). Initially, it could be both anus- and phallus-like, and also like an alien, a microorganism

¹⁰ Note in the first case the almost “autographic” merging of “L” and “U,” still readable.



Figure 9: Continuum between drawing and writing (a) AI-generated map of Victoria, Australia, and (b) a detail.



Figure 10: Some examples for the “clungus” thread, including crossover images.

(tardigrade) or a Cthulhu-like figure. Later, usage stabilized the notion of clungus as a “meat hole,” so that it became also an entry in Urban Dictionary.¹¹ Of course, merging operations (in a sort of semantic syncretism) are possible, e.g., Clungus + Yellow coolant (with car) + Kandahar 1923 (as can be seen in the last images of Figure 10).¹²

7 Conclusions

In order to sum up my widely heterogeneous contribution, I would like to be as schematic as possible in my conclusions:

- 1) Agency is probably the most interesting concept in order for semiotics to enter into a dialogue with AI;
- 2) Deep learning (mostly, NN) is interesting for a discussion of subjectivity not in relation to a connectionist/brain metaphor, rather because it shows a cybernetic basic agency as it is feedback driven;
- 3) This feedback creates a specific data-driven computational environment (feedback computation), where environment is assumed in a biosemiotic sense;
- 4) This situation creates some specific communication circuitry and indeed some specific enunciative positions. Agency, of different kinds, is distributed;
- 5) These enunciative positions are multiplied, as they lie at various levels:
 - i) in the complex hierarchy of programming as a semiotic activity (see Valle and Mazzei 2017);
 - ii) in human-machine interaction in terms of usage;
 - iii) inside the texts produced by generative AI (see Basso Fossali this issue; Paolucci this issue).

In relation to enunciation (see D’Armenio et al. 2024a, 2024b), this may ask for a typology of Observers, as proposed by Fontanille since his seminal *Les espaces subjectifs* (1989);

- 6) Such a situation is indeed another step in a shift (that McLuhan 1964 associated to the electronic revolution) towards an “oral” textual regime versus a “written” grammar one, as the former is based on analogy, examples, drifting, autography;
- 7) Curiously enough, this same shift is powered at the end by the development “on steroids” (big data, GPUs) of the formal, grammar based regime of classic

¹¹ “A meat hole image generated by AI” (<https://www.urbandictionary.com/define.php?term=Clungus>).

¹² At the moment of this writing, the three threads are still actively developed in the community.

computation, which bakes it: a movement typical of post-typographic secondary orality.

Acknowledgments: I am deeply thankful to Maria Giulia Dondero and Juan Alonso Aldama for inviting me at the *Séminaire International de Sémiotique à Paris 2023–2024*, “Énonciation(s) et passions dans les territoires sémiotiques ouverts par l’Intelligence Artificielle,” thus prompting the first draft of this paper. I would like to thank the anonymous reviewers for their in-depth reading and for the comments that helped to largely improve the final form of this article.

References

- Abelson, Hal, Gerald J. Sussman & Julie Sussman. 1996. *Structure and interpretation of computer programs*, 2nd edn. Cambridge, MA: MIT Press.
- Austin, Jacob, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, Charles Sutton. 2021. Program synthesis with large language models. <https://arxiv.org/abs/2108.07732> (accessed 26 December 2024).
- Basso, Pierluigi. 2003. *Il dominio dell’arte*. Roma: Meltemi.
- Chollet, François. 2021. *Deep learning with Python*, 2nd edn. Shelter Island, NY: Manning.
- D’Armenio, Enzo, Adrien Deliège & Maria Giulia Dondero. 2024a. A semiotic methodology for assessing the compositional effectiveness of generative text-to-image models (Midjourney and DALL-E). In *Proceedings of the first workshop on critical evaluation of generative models and their impact on society*, ECCV 2024. Berlin: Springer.
- D’Armenio, Enzo, Adrien Deliège & Maria Giulia Dondero. 2024b. Semiotics of machinic co-enunciation: About generative models (Midjourney and DALL-E). *Signata* 15. <https://journals.openedition.org/signata/5290> (accessed 26 December 2024).
- Dondero, Maria Giulia. 2020. *The language of images*. Cham: Springer.
- Eco, Umberto. 1968. *La struttura assente*. Milano: Bompiani.
- Eco, Umberto. 1976. *A theory of semiotics*. Bloomington, IN: Indiana University Press.
- Fontanille, Jacques. 1989. *Les espaces subjectifs: introduction à la sémiotique de l’observateur*. Paris: Hachette.
- Fontanille, Jacques. 2006. *The semiotics of discourse*. New York: Peter Lang.
- Gabrielli, Maurizio & Simone Martini. 2010. *Programming languages: Principles and paradigms*. London: Springer.
- Goodman, Nelson. 1968. *Languages of art*. Indianapolis, IN: Bobbs-Merrill.
- Greimas, Algirdas J. & Joseph Courtés. 1979. *Sémiotique*. Paris: Hachette.
- Hoffmeyer, Jesper. 2008. *Biosemiotics: An examination into the signs of life and the life of signs*. Scranton, PA: University of Scranton Press.
- Knuth, Donald E. 1992. *Literate programming*. Stanford, CA: Center for the Study of Language and Information.
- Lieto, Antonio. 2021. *Cognitive design for artificial minds*. London & New York: Routledge.
- Lotman, Jurij M. & Boris A. Uspenskij. 1973. *Tipologia della cultura*. Milano: Bompiani.
- Maran, Timo. 2020. *Ecosemiotics: The study of signs in changing ecologies*. Cambridge: Cambridge University Press.

- McLuhan, Marshall. 1964. *Understanding media*. Cambridge, MA & London: MIT Press.
- Mirzadeh, Iman, Keivan Alizadeh, Hooman Shahrokhi, Oncel Tuzel, Sami Bengio & Mehrdad Farajtabar. 2024. GSM-symbolic: Understanding the limitations of mathematical reasoning in large language models. <https://arxiv.org/abs/2410.05229> (accessed 26 December 2024).
- Müller, Andrea & Sarah Guido. 2017. *Introduction to machine learning with Python*. Sebastopol, CA: O'Reilly.
- Ong, Walter. 1982. *Orality and literacy: The technologizing of the word*. New York: Methuen.
- Peirce, Charles Sanders. 1878. Deduction, induction, hypothesis. *Popular Science Monthly* 12. 470–482.
- Pettitt, Tom. 2007. Before the Gutenberg parenthesis: Elizabethan-American compatibilities. Paper presented at *Media in Transition 5, Communications Forum*, April 27–29.
- Poole, Davide L. & Alan K. Mackworth. 2023. *Artificial intelligence: Foundations of computational agents*, 3rd edn. Cambridge: Cambridge University Press.
- Rosenblueth, Arturo, Norbert Wiener & Julian Bigelow. 1943. Behavior, purpose, and teleology. *Philosophy of Science* 10(1). 18–24.
- Sebeok, Thomas A. 2001. *Signs: An introduction to semiotics*, 2nd edn. Toronto: University of Toronto Press.
- Seibel, Peter. 2009. *Coders at work: Reflections on the craft of programming*. New York: Apress.
- Turner, Huck. 2002. An introduction to methods for simulating the evolution of language. In Angelo Cangelosi & Domenico Parisi (eds.), *Simulating the evolution of language*, 29–50. London: Springer.
- Valle, Andrea. 2017. Modes of sign production. In Sarah G. Beardsworth & Randall E. Auxier (eds.), *The philosophy of Umberto Eco*, 279–304. Chicago, IL: Open Court.
- Valle, Andrea. 2020. On a fragment of BASIC code in *Foucault's pendulum* by Umberto Eco. In Vincenzo Idone Cassone, Jenny Ponzo & Mattia Thibault (eds.), *Languagescapes: Ancient and artificial languages in today's culture*, 169–190. Roma: Aracne.
- Valle, Andrea & Alessandro Mazzei. 2017. Sapir-Whorf vs. Boas-Jakobson: Enunciation and the semiotics of programming languages. *Lexia* 27–28. 505–525.
- Von Neumann, John. 1993 [1945]. First draft of a report on the EDVAC. *IEEE Annals of the History of Computing* 15(4). 27–43.
- von Uexküll, Jakob. 2010 [1934]. *A foray into the worlds of animals and humans*. Minneapolis, MN: University of Minnesota Press.