

УДК: 165.62

DOI 10.17726/phillT.2024.1.4



Концепция рекурсии в когнитивных исследованиях. Часть I: от математики к познанию

Михайлов Игорь Феликсович,

доктор философских наук, ведущий научный сотрудник,

Институт философии РАН

Москва, Россия

ifmikhailov@iph.ras.ru

Аннотация. В статье обсуждаются различные подходы к понятию рекурсии и его эволюция от математики к когнитивным исследованиям. Рассматриваются такие подходы как: самовстраиваемые структуры, многоуровневые иерархии с использованием одного и того же правила и встраивание структур в структуры. Предлагается концепция мета-рекурсии. Исследование мета-рекурсии может объяснить возможность применения рекурсивных процессов к многоуровневым иерархиям, при этом рекурсивные процедуры действуют как генераторы. Эти типы рекурсивных процессов могут быть фундаментальными элементами общих когнитивных способностей. Автор также кратко обсуждает роль вероятностных подходов в современных рекурсивных когнитивных теориях. Предполагается, что иерархический механизм познания демонстрирует своего рода мета-рекурсию в том смысле, что рекурсивные нейронные петли могут поддерживать некоторые примитивные рекурсивные когнитивные процессы, которые, в свою очередь, объясняют рекурсивность языковых грамматик, пространственной ориентации, социального познания и т.д. Исследование показывает, что использование нескольких подходов к пониманию феномена рекурсии может обеспечить более полное понимание сложности рекурсии, поскольку она играет важную роль в таких областях, как язык, математика и когнитивная наука.

Ключевые слова: рекурсия; когнитивные исследования; самовстраиваемые структуры; иерархические структуры; мета-рекурсия.

The Concept of Recursion in Cognitive Studies. Part I: From Mathematics to Cognition

Mikhailov Igor Felixovich,

*Doctor of Philosophy, Leading researcher,
Institute of Philosophy of the Russian Academy of Sciences
Moscow, Russia*

ifmikhailov@iph.ras.ru

Abstract. The paper discusses different approaches to the concept of recursion and its evolution from mathematics to cognitive studies. Such approaches are observed as: self-embedded structures, multiple hierarchical levels using the same rule, and embedding structures within structures. The paper also discusses the concept of meta-recursion. Examining meta-recursion may enable understanding of the ability to apply recursive processes to multi-layered hierarchies, with recursive procedures acting as generators. These types of recursive processes could be the fundamental elements of general cognition. The paper also briefly discusses the role of probability in current recursive approaches to cognition. It is conjectured that the hierarchical mechanism of cognition demonstrates a kind of meta-recursion in the sense that recursive neural loops may support some primitive recursive cognitive processes, which in turn account for recursiveness of language grammars, space orientation, social cognition, etc. The study indicates that using multiple approaches to understand the phenomenon of recursion can provide a more complete understanding of the complexity of recursion, as it plays a significant role in fields like language, mathematics, and cognitive science.

Keywords: recursion; cognitive studies; self-embedded structures; hierarchical structures; meta-recursion.

If one has no sense of humor, then, at least,
one should have a sense of having no sense of humor.
(attributed to) *Stanisław Jerzy Lec*

What is known as Cognitive Science (CS) started from the conjecture that cognition is essentially computation conceived as algorithmic manipulation with whatever may be assessed as symbols. In turn, theories of computation and computability, as evolved since the edge of XIX-XX centuries, started from attempts to solve Gilbert's decidability problem initially basing on the earlier theory of recursive functions.

The Church's Thesis is often interpreted as the statement of equality of the class of computable functions to the class of partially recursive ones. These circumstances could induce a belief that recursion is somewhat crucial to understanding the realm of the cognitive as well. And, indeed, recursion has been seen by Noam Chomsky as a substantial mechanism fundamental for human linguistic capacities, while today we see a certain number of publications envisaging recursive nature of mind and consciousness. A reservation should be made, however, that recursive mechanisms that facilitate our mental lives are seen as more complex now. In the first part of the review, I outline the history of discussions in brief and give some advances to the importance of the subject for cognitive studies.

Discussions in mathematics

According to Soare [1], the concept of recursive functions evolved in mathematics through the works of various mathematicians such as Dedekind, Hilbert, Skolem, Ackermann and R. Peter from 1888 to 1951. The current meanings of 'recursive' derive from the verb 'recur' that means 'to return to a place or status', or the concept of 'definition by recursion', which is defined as 'definition by induction.' The term 'recursive' is presently used in the field of mathematics in at least four different ways, which are summarized in the following disambiguation:

<...>The current meanings of recursive and recursion are these:

(i) recursion is used with meanings derived from the verb 'recur,' <...>;

(ii) recursion is used in the sense of 'definition by recursion' (i.e., definition by induction) <...>;

(iii) following Kleene 1936 and Church 1936 the term 'recursive' denotes 'general recursive' and any of its mathematically equivalent formal variants, such as 'Turing computable,' ' λ -definable,' 'specified by a Post 1944 normal system,' or Kleene's ' μ -recursive'.

(iv) 'recursive' is used to mean any of the informal variants of Definition 1.1 such as '(intuitively) computable,' 'effectively calculable,' 'defined by a mechanical process,' or 'specified by an algorithm.' [1, p. 310]

The μ -recursive functions, partially recursive functions and general recursive functions are different classes of computable functions.

A function is said to be μ -recursive if it can be obtained through primitive recursion schema and unbounded minimization schema. Partially recursive functions are a class of functions that contain the computable partial functions, which are obtained from the total recursive functions by blocking the output values of the function at some points. General recursive functions have a different definition. Here, ‘general recursive’ refers to any algorithmically computable function, and recursive implies being defined by means of finite compositions of the following three fundamental operations: initial functions, composition operators, and primitive recursion schema.

The class of partially recursive functions is equal to the class of computable functions. This is known as the Church-Turing thesis. It states that if an algorithm exists to compute a mathematical function, then a Turing machine can be constructed to compute that function as well, and vice versa. The definition of partially recursive functions by primitive recursion and unbounded minimization schema is equivalent to the definition of computable functions, which can be computed by a Turing machine. Hence, the class of partially recursive functions is equivalent to the class of computable functions, and any function that can be computed by any recognizable system of computation, such as register machines, counter machines, or Markov algorithms, can be computed by any other recognizable system of computation with at most a polynomial increase in running time.

The Church-Turing thesis states that a function is computable by an effective method if and only if it is computable by a Turing machine. As is shown in [2] the notion of ‘effectively computable’ was formalized first by Church using functions expressed in the lambda calculus, and later by Turing using machines. Church and Turing showed that their models for computation were equivalent in the sense that they capture the same class of functions, which are now known as the recursive or computable functions. The authors also mention that there are other formalizations of the notion of ‘effective computability’, such as the partial recursive functions, which are equivalent to Turing machines and the lambda calculus.

The Self-reference lemma, also called the Fixed-point lemma or the Diagonalization lemma, establishes that for any formula $Q(x)$ that describes a property of a numeral, there exists a sentence W that is logically equivalent to the sentence $Q(\ulcorner W \urcorner)$ ¹. Gödel’s incompleteness theorems

¹ $\ulcorner W \urcorner$ stands for a numeral of a Gödel number of W .

rely heavily on self-referential constructions like this. Additionally, the halting problem for Turing machines is undecidable, in part, because of the possibility of recursive loops in their operations [2, p. 135].

An important reservation was given by Kleene in [3, p. 187-188]: The idea, and imagery, of an ‘oracle’ was introduced by Turing 1939 pp. 172-173 in defining the computability of one fixed number-theoretic predicate from another. (He could just as well have talked about their representing functions, taking as values 0 for truth and 1 for falsity.) Kleene 1952 (at the 1950 International Congress) adapted this idea to define the recursiveness (or computability) of a function ...

According to Soare, this novation started continuous discussion on ‘relative computability’, or ‘relative reducibility’, which eventually allowed for stating that, using *Turing reducibility* ‘(denoted $A \leq_r B$)’, we say that two sets A and B have the same *information content* or have the same *Turing degree* if $A \leq_r B$ and $B \leq_r A$.’ [1, p. 302]. This remark, in my view, opens an interesting perspective on cognitive representations, as I will try to show in a forthcoming part of this article.

A more general view

An interesting attempt to provide a comprehensive account of the concept of recursion, its relation to inductive definitions and mathematical induction, and its use in cognitive science is made by Sergio Mota in [4], where he presents an in-depth exploration of the concept of recursion and its relevance to cognitive science research.

In his view, recursion, inductive definitions, and mathematical induction are related concepts that are often conflated. Recursion refers to a function defined by induction, which involves defining a function for an argument by using its previously defined values and simpler functions. Inductive definition, on the other hand, involves defining a set in terms of its elements, similar to how natural numbers are defined. Mathematical induction is a proof technique that allows one to establish the truth of an infinitely many statements. While these concepts share similarities, the author warns that they should not be conflated.

Recursion and self-involvement are conceptually related in that recursion often involves self-embedding or self-reference. For example, a function defined by recursion may call itself as part of its definition. This kind of self-reference or self-embedding is often a hallmark of

systems that exhibit recursive behaviour. However, it is important to distinguish between recursion and self-reference, as they are not necessarily the same thing.

Mota reviews the Chomskian approach to recursion, which ‘emphasizes the origin of recursion in the formal sciences, and applies it to characterize the mechanical procedure which underlies the language faculty’ [4, p. 89]. But he also mentions three non-Chomskian approaches:

1. *Self-embedded structures*: This approach sees recursion as the ability to embed structures within structures of the same kind, creating a sort of self-reference. For example, a sentence like ‘the cat that chased the mouse that ate the cheese’ is recursive because it contains nested clauses of the same form. This approach also includes self-embedded structures like fractals, in which the same pattern is repeated at different scales. The self-embedded structures approach to recursion emphasizes how the ability to embed structures within structures of the same kind creates a sort of self-reference, and views recursion as a property of such structures. This approach recognizes that recursive structures can take many different forms, including linguistic structures, as well as structures found in mathematics and computer science. It also acknowledges that recursive structures are found in nature in the form of fractals, which are self-similar patterns at different scales. Overall, this approach highlights the importance of self-embedding as a key characteristic of recursive behaviour.
2. *Multiple hierarchical levels using the same rule*: This view emphasizes the ability to represent multiple levels of structure using the same set of rules. For example, a tree structure in which each node can have many children can be represented using the same ‘recursive’ rule to create new nodes at each level. This approach is commonly used to understand hierarchical structure in general and is not limited to language. The multiple hierarchical levels using the same rule approach to recursion emphasizes the ability to represent multiple levels of structure using the same set of rules, often leading to complex hierarchical structures. This approach is not limited to language but is also commonly used to understand hierarchical structure in many other domains, such as mathematics, computer science,

and biology. For example, a fractal pattern can be created using a simple recursive rule that generates the same pattern at different scales. In general, this approach highlights the importance of using the same set of rules at multiple levels to create complex hierarchical structures.

3. *Embedding structures within structures of the same kind*: This approach is similar to the self-embedded structures approach but focuses on the use of embedding to create hierarchical structure. For example, in a sentence like ‘the cat that chased the mouse that ate the cheese,’ the embedded clauses help to create a hierarchical structure by nesting ideas within each other. This approach is sometimes called ‘nesting,’ and it is closely related to the self-embedded structures approach, as it involves the use of embedding to create hierarchical structure. However, it focuses more specifically on how embedding can be used to create hierarchical ‘nested’ structures. This approach can be applied not only to language, but also to other domains where hierarchy and nesting are common, such as mathematics and computer science. Overall, this approach highlights the importance of the hierarchical relationships that are created when structures are embedded within each other.

All of these approaches focus on different aspects of recursion, but they share the idea that recursion involves creating new structures that are similar to the existing structures in some way. Each approach has its own advantages and limitations for understanding recursion. While the self-embedded structures, multiple hierarchical levels using the same rule, and embedding structures within structures approaches each provide valuable insights into recursion, it is likely that multiple approaches will be needed to fully understand the phenomenon. This is because recursion is a complex phenomenon that plays a role in many different areas, including language, mathematics, and computer science, among others. Therefore, each approach has its own advantages and limitations, and using multiple approaches can help to provide a more complete understanding of recursion.

Each of the approaches to recursion is significant for cognitive science because they all help to shed light on different aspects of the phenomenon and can lead to a better overall understanding of how humans process information.

The self-embedded structures approach is significant because it

emphasizes the importance of self-referentiality. This approach is especially relevant to understanding how language and thought are connected, and how our ability to use recursive structures may be related to more general cognitive processes.

The multiple hierarchical levels using the same rule approach is significant because it highlights the power of using a limited set of rules to generate complex hierarchical structures. This approach is relevant not only to language, but also to many other domains, such as mathematics, computer science, and biology, where complex hierarchical structures are common.

The embedding structures within structures approach is significant because it focuses on how embedding can be used to create hierarchical relationships. This approach is directly relevant to understanding how we use language, and how we are able to understand complex sentences with embedded clauses. Not to forget Predictive Processing/Active Inference paradigm in CS completely based on the ‘hierarchical Bayes’ formalism [5], which will be touched on in Part II.

Mota’s review helps conceiving that the concept of recursion has two important applications in CS: one, originated from Chomsky, refers to the ability of building self-embedding structures in thought and language, and the other, characteristic to present-day cognitive approaches, identifies recursive loops and hierarchies in underlying computational neural mechanisms. In a sense, this allows for a concept of meta-recursion that should grasp this nested view of cognition. The question of whether we can speak about a kind of meta-recursion in the case of cognition is an interesting one and might be developed against the ideas behind the relative mathematical concept [6].

One way to approach this question is to consider the idea that cognition is itself a recursive process. This view suggests that the processes underlying cognitive functions are recursive, and that this recursive process is what enables us to perform complex cognitive tasks. For example, in language processing, we are able to understand complex sentences that contain embedded clauses by recursively applying a set of rules to generate a hierarchical structure. This recursive process of applying rules to generate hierarchical structures is thought to be a fundamental aspect of language processing and may be seen as a kind of meta-recursion. Similarly, in mathematics, we are able to break down complex problems into simpler, more manageable components by recursively applying a set of rules to generate a solution. This recursive

process is also thought to be a fundamental aspect of mathematical reasoning and may be seen as another kind of meta-recursion.

Overall, the idea of meta-recursion in the case of cognition is intriguing and has some support in the literature. While it is still a topic of debate and further research is needed, it is clear that the ability to perform recursive processes is a fundamental aspect of human cognition and is involved in many different cognitive tasks.

Recursion in cognitive linguistics

If by ‘cognition’ we mean processing that underlies, in particular, linguistic capabilities, then one example of the said meta-recursion may be the linguistic recursion famously studied by Chomsky. In [7] he presents a series of theoretical arguments and observations about the nature of human language. Some of the key themes he touches include the idea that language is a computational system that can be studied by analogy with other computational domains; the assertion that the range of possible linguistic expressions is constrained by a set of innate, domain-specific principles of ‘Universal Grammar’; the suggestion that language is optimized for the conceptual/intentional interface rather than the sensory-motor interface, with ‘externalization’ a secondary phenomenon; and the hypothesis that the minimal recursion approach to language offers a more streamlined and explanatorily powerful framework for studying language than earlier approaches. In the chapter cited, he emphasizes the importance of formulating precise, well-motivated theoretical hypotheses about language structure and function, and using these hypotheses to generate testable predictions and explanations for phenomena at all levels of linguistic analysis.

According to Chomsky, minimal recursion is the pursuit of the simplest possible solution to the central problem of determining the nature of the recursive procedures in language that provide an unbounded array of hierarchically structured expressions at the sensory-motor and conceptual-intentional interfaces. This involves the enumeration of a set of discrete objects by a computable finitary procedure that can be programmed for an ordinary digital computer that has access to unlimited memory and time. He argues that this approach deepens explanatory power and expedites the study of language acquisition and offers insights into the evolution of the language capacity [7, p. 1].

Chomsky links his approach to the Recursive Function Theory developed by Gödel, Church, Kleene, and others. He notes that their work established a framework for studying mathematical operations in a way that could be executed by a finite computational procedure, and he argues that language can be studied in a similar way. He also suggests that the human language capacity may involve a kind of abstract computational procedure that outstrips the Turing machine model in its complexity¹, which could account for the remarkable range of linguistic expressions generated by the system.

According to Chomsky, the program of minimal recursion eliminates many unwanted stipulations that burden earlier work on language. He argues that by reducing the number of elementary operations posited in the computation and avoiding revision of complex phrase structure rules, we can arrive at a system that is more straightforward, predictive, and explanatory than earlier approaches. For example, he suggests that the ‘externalization’ of the language system, that is, the mapping of abstract linguistic structures to sensory-motor forms, need not be modelled by a separate set of rules but can rather be understood as a process of translation. By constraining the operations in this way, we can arrive at a ‘minimalist’ approach that emphasizes conceptual and computational simplicity [7, p. 3].

As ‘evidence for an asymmetry of the interfaces, with externalization an ancillary procedure’ he refers to the hypothesis (which Chomsky calls ‘thesis T’) that language is optimized relative to the conceptual/intentional interface alone, and that ‘externalization’ is a secondary phenomenon. In other words, the primary driver of the language capacity is the system that generates meaningful expressions (the conceptual/intentional interface), while the mapping of those expressions to external forms (sensory-motor forms) is a secondary, less important process. Evidence for this hypothesis comes from observations about the limited range of variation in the sensory-motor properties of language, suggesting that these properties do not play a fundamental role in shaping the system as a whole. The idea is that a minimalist approach should aim to capture the essence of the conceptual/intentional interface while leaving externalization to be modelled as a kind of translation procedure [7, p. 7].

Tomalin [8], in his turn, suggests that certain research fields, in-

¹ Sufficiency of TM as a universal model for any computation has been discussed in the literature on philosophy of CS; see, for instance, [16, 17].

cluding linguistics, were plagued by misconceptions because researchers had not realized that certain phenomena in those fields could be most easily analysed recursively. He also mentions that some linguists in the 1950s were aware of developments in recursive function theory from the first decades of the 20th century, and it is possible to trace the routes along which recursive techniques entered the domain of linguistic inquiry. Particularly Bar-Hillel and Chomsky are mentioned in this regard.

The role of recursive definitions within the Minimalist Program is discussed at length, and the main focus falls upon comparatively recent claims concerning the centrality of recursion in the context of biolinguistics. Specifically, the hypothesis that recursion constitutes a species-specific property of the human language faculty that is particularly associated with natural language was reassessed, and the problematic term ‘recursion’, in the author’s opinion, should be abandoned and replaced by less ambiguous terminology such as ‘inductive definition’. The Minimalist Program, as said above, is a framework in linguistics developed by Noam Chomsky in the 1990s, which aims to provide a basis for explaining the fundamental principles of Universal Grammar with the minimum of theoretical luggage. It seeks to derive the properties of the human language faculty from considerations of simplicity, economy, and optimal design. The program focuses on deriving grammatical specifications from general principles that are fixed and universal, within a deductive framework that considers only those operations necessary for this derivation.

Tomalin advises the term ‘recursion’ to be replaced by more accurate terminology such as ‘inductive definition’ in the specific context of linguistic theory because there are inherent ambiguities in the former term as it is standardly used within the formal sciences. If ‘recursion’ is interpreted as ‘inductive definition,’ the ‘recursive’ components of generative grammar are able to accomplish all that is required of them, and the correspondence between the human linguistic and arithmetical cognitive functions is not undermined. Any other interpretation of the term ‘recursion’ makes stronger (and possibly undesirable) claims about the nature of the faculty of language, which should be avoided if possible. The term ‘inductive definition’ is a more accurate and less ambiguous alternative to ‘recursion’ within the context of linguistic theory. The term ‘recursion’ has a number of ambiguous meanings within the formal sciences, which can lead to confusion, imprecise thinking, and

poor communication within the field of linguistics, and this highlights the need for more precise terminology. In this particular case, the term ‘inductive definition’ more accurately captures the linguistic phenomenon at hand and helps to avoid any misunderstandings or overgeneralizations about the nature of language.

Corballis [9] challenges the notion that recursion is unique to human language and argues that it is instead a property of human thought that likely preceded language. Recursion, in his definition, is the process whereby a computational routine calls itself or a similar routine and is distinguished as the essence of human language. The author offers examples, including *The House that Jack Built*, to illustrate this concept. In contrast, Corballis believes that this property is not universally present in present-day languages and likely evolved before language itself.

Although not attacking Chomsky directly, he does reference some of Chomsky’s ideas on recursion as they relate to language and thought. Instead, he offers a more nuanced perspective on the evolutionary origins and cognitive processing of recursion, distinct from Chomsky’s view that language and thought essentially depend on the same underlying structure. He supports his main thesis that recursion is not unique to human language but is instead a property of human thought that likely preceded language by pointing out the recursive abilities of animals in various cognitive domains, including birdsong and the nest-building behaviour of birds and the design of spider webs, among others. He believes that recursive abilities may have developed in humans through evolution of the brain, potentially as a means of dealing with the increasing complexity of the environment. In terms of cognitive processing, Corballis suggests that recursion is likely processed by the same neural mechanisms used for other complex computations, rather than being governed by a specific grammatical module, as Chomsky and others have proposed.

Particularly, recursive processing is likely done by the dorsal route neural system, which is associated with analytical and sequential processing for mental rotation tasks, spatial navigation, complex sequence learning, and social cognition¹. This is based on various lines of evidence, including neuroimaging studies, patient studies, and studies of non-human animals, which have demonstrated overlap in neural regions involved in these different cognitive domains. For example, studies have shown that the same neural mechanisms involved in

¹ Which may be another view on what I’d like to call ‘meta-recursion’.

mental rotation tasks are involved in processing recursive structures in language and non-linguistic domains, such as music. Similarly, studies have shown that the same regions of the brain that are involved in processing spatial navigation and complex sequences, such as the premotor cortex, are also involved in processing recursive structures in language. Overall, Corballis suggests that these findings offer support for a more general-purpose neural system that serves both language and non-linguistic domains, rather than a specialized grammatical module dedicated to recursion.

Baryshnikov [10] discusses the problem of recursion as one of the central topics of language, brain, thinking, and civilization. He explains that recursion can be defined as a structural self-similarity or nesting of ideas. Recursion is reduced to an abstract procedure calling itself or to a component containing a component of the same class. The problem of self-similar formal structures in linguistic studies has been known since the works of Panini, W. von Humboldt and the early works of N. Chomsky, in which the latter had taken grammar to hierarchical recursive structures.

The structural self-similarity or nesting of ideas is a way to overcome the ontological or metric linearity in the theory of knowledge, mathematics, programming, music, geometry and art. It can take many forms such as textually (as in 'I think, therefore I exist.'), graphically (like the fractal shapes), logically (e.g., in the liar paradox), and mathematically (such as the Fibonacci numbers or the formula $0 = 1 \cdot n! = n \cdot (n - 1)!$ [where $n > 0$]). However, the essence of recursion is reduced to an abstract procedure calling itself or to a component containing a component of the same class.

Cognitive recursion as probabilistic

Emergence of connectionism since late 1980ies put an additional challenge to understanding recursion in CS. Thus, Christiansen and Chater [11] have focused on presenting a connectionist model of human performance in processing recursive language structures, analysing its behaviour to match human behaviour, and providing a novel explanation of people's limited recursive performance without assuming the existence of a mentally represented competence grammar allowing unbounded recursion.

The connectionist model presented in the article is trained on sim-

ple artificial languages and its qualitative performance profile matches human behaviour, both on the relative difficulty of center-embedding and cross-dependency, and between the processing of these complex recursive structures and right-branching recursive constructions. The article presents empirical evidence that people are only able to deal easily with relatively simple recursive structures in natural language, and document difficulties that people experience when processing more complex recursive structures. They use the example of a doubly center-embedded sentence like '*The mouse that the cat that the dog chased bit ran away,*' which is extremely difficult to understand.

The authors propose a connectionist model which receives a word as input at time t and predicts the next word at time $t+1$, based on the previous context. The model is trained on simple artificial languages that include both right-branching recursive constructions and more complex recursive structures. The model is assumed to have the ability to distinguish nouns from verbs, which is crucial to predict subsequent verbs. The success of this model in matching human behaviour in the processing of recursive structures provides evidence that recursive language processing by humans can be explained through parallel distributed processing mechanisms, rather than a compositional linguistic structure in the form of a mentally represented competence grammar.

Superficial reading of the paper may give the impression that the model proposed here may be seen as a primitive predecessor of modern GPTs. But, while there are similarities in the sense that both are machine learning models used for natural language processing, there are also significant differences between the connectionist model presented in the article and modern-day Generative Pre-Trained Transformer (GPT) models. The connectionist model presented in the article is trained for a specific task, whereas GPT models are pre-trained on vast amounts of text data and fine-tuned for various tasks. Additionally, the connectionist model in the article uses a simple architecture based on feedforward and recurrent neural networks, while GPT models use a complex architecture based on transformer neural networks.

Therefore, while the connectionist model presented in the article may be seen as a predecessor to modern-day GPT models in the sense that it is a machine learning model for natural language processing, the differences between the two are significant enough that it would be inaccurate to classify the connectionist model as a primitive predecessor of modern-day GPT models.

At the same time, while the emergence of GPTs has certainly had a significant impact on the field of natural language processing, it would be inaccurate to say that feedforward and recurrent neural networks are now wholly in the previous epoch. In fact, feedforward and recurrent neural networks are still being used extensively in various fields, including natural language processing. While GPTs have achieved remarkable performance in recent years, they are not without limitations such as their large computational requirements, high energy consumption, and sensitivity to bias in the training data. Hence, there is still ongoing research that seeks to improve both feedforward and recurrent neural networks as well as GPTs and other language models.

Connectionism appeared to be one of the approaches referred nowadays as 'bio-inspired'. Their maybe most substantial treat is usage of probabilistic computations, unlike the so-called cognitive classicism [12]. In particular, Kolodny, Lotem and Edelman [13] propose a model for learning a generative probabilistic grammar of experience, which is supposed to be a process-level model of language acquisition. This is a model for unsupervised learning of natural language corpora based on computational and biological principles. It incrementally learns a grammar that captures statistical patterns, which can then be used to generate new data. The grammar constructed takes the form of a directed weighted graph, whose nodes are recursively defined patterns over the elements of the input stream. The model was evaluated in seventeen experiments grouped into five studies, examining the model's generative ability, the characteristics of the learned representation, sequence segmentation and chunking, artificial grammar learning, and certain types of structure dependence. The results showed that the model's performance largely vindicates the design choices, suggesting that progress in modelling language acquisition can be made on a broad front.

The model described in the paper uses recursive search routine, in which the algorithm finds all the possible single-node covers of the beginning of the sentence, then for each of these calls itself on the remainder of the sentence, until it finds a complete cover or determines that such a cover does not exist. The recursion is thus hierarchical in nature.

As a probabilistic model, it assigns probabilities to the occurrence of different tokens based on the statistical frequency of those tokens in the input stream. Furthermore, this grammar model considers choosing between different grammar structures to generate the next token, and it does so by assigning probabilities to different grammar structures that

might generate the same next token. This probabilistic approach allows the model to produce a variety of grammatical structures, while still conforming to the statistical distribution of the input data.

As for any genetic links to GPT, briefly touched on above, both the GPT and the model described in the paper are generative probabilistic models of language processing, so it is possible to draw some conceptual parallel between them. It is worth noting, however, that the specific techniques, algorithms, and architectures used in the GPT, and the model described in the paper are likely different.

Zhang and Amin [14] discuss how probabilistic programming can be used to model reasoning about the beliefs, desires, and intentions of other agents, also referred to as theory of mind. The paper describes how conditioning can be explicitly represented in probabilistic programs, allowing for nested conditioning and recursive probabilistic programs. The authors illustrate how the representation language can be used to explore new directions in game theory, artificial intelligence, and linguistics, but also discuss the algorithmic challenges posed by these kinds of models and describe how dynamic programming techniques can help address these challenges.

Probabilistic programming allows for nested conditioning by modelling each agent's reasoning as conditional sampling, which can become the subject of other agents' reasoning. Conditioning is an operation that can be defined within such models and represented as nested conditions, which allows for reasoning about reasoning. This approach goes beyond existing formalisms, such as graphical models, where conditioning is more naturally seen as an operation that is applied to a model but not itself represented.

The paper describes a few examples of conditioning and its probabilistic representation. For instance, one example is a classic riddle: *'Two doors guarded by two guards – one door leads to freedom, the other to certain death. One guard always tells the truth, the other always lies. You do not know who is who. You may ask one of the guards only one question to find the right door.'* This riddle can be represented as a probabilistic program with two binary random variables, one for each guard's truthfulness, and a conditional statement for each guard's response. The answer to the question then corresponds to the output of the program conditioned on the evidence provided by the guard's response.

Recursion refers to an agent reasoning about another agent's reasoning about yet another agent and so on, where reasoning is modelled

as conditional sampling. The model allows for recursion by representing inference as a general-purpose probabilistic programming language with constructs for defining and calling subroutines. Each subroutine corresponds to a probabilistic program that represents a single agent's reasoning about a simpler subproblem. When one agent reasons about another, the resulting program contains a call to the corresponding subroutine, which itself contains calls to even simpler subroutines, and so on.

As a conclusion

The general concept of recursion is essential for cognitive science because it is a fundamental aspect of human cognition and is involved in a wide range of cognitive processes, including language, problem-solving, and decision-making. Recursion allows us to generate complex structures and relationships by iteratively applying a certain set of rules or procedures. This ability to create complex structures from simple rules is essential to many cognitive processes, including language acquisition, where recursive structures are common. Recursive structures are also an important aspect of mathematical reasoning and problem-solving, where they are used to break down complex problems into simpler, more manageable components.

The study of recursion is also relevant to artificial intelligence and computer science, as it has led to the development of algorithms and programs that are capable of processing recursive structures. Understanding recursion is thus essential for developing and improving artificial intelligence systems that can perform tasks traditionally associated with human cognition, such as natural language processing and decision-making.

Overall, the concept of recursion is a fundamental aspect of human cognition and has important implications for many different areas of cognitive science, as well as artificial intelligence and computer science.

The role of recursion in the present-day trends in CS will be reviewed in the forthcoming part of this study. But it may be time to share some intermediate considerations. In *Tractatus*, Wittgenstein once remarked that logical and mathematical tautologies say nothing of the world, but show something important about it, namely:

The fact that the propositions of logic are tautologies *shows* the formal – logical – properties of language and the world. <...> If

propositions are to yield a tautology when they are connected in a certain way, they must have certain structural properties. So their yielding a tautology when combined *in this way* shows that they possess these structural properties [15, para. 6.12, p. 214].

The similar thing may be said about recursion. The fact that some functions, or procedures, are μ -, general, partial, or whatever recursive, while others are not, is not informative for natural (or cognitive, which is part of natural) science. But it shows an important thing about our implied ontologies, about which ‘nothing can be said’. Particularly, that those ontologies are adjustable relative to the overall class of recursive functions, making the set of their possible applications grow or collapse in number.

As applied to cognitive science, this metaphysical observation supports the still cautious presumption that the hierarchical mechanism of cognition demonstrates what may be labelled as ‘meta-recursion’ in the sense that recursive neural loops may support some primitive recursive cognitive processes, which in turn account for recursiveness of language grammars, space orientation, social cognition, etc.

References

1. Soare R. I. Computability and Recursion // *Bulletin of Symbolic Logic*. 1996. Vol. 2, № 3. P. 284-321.
2. Prokopenko M. et al. Self-referential basis of undecidable dynamics: From the Liar paradox and the halting problem to the edge of chaos // *Phys Life Rev*. 2019. Vol. 31. P. 134-156.
3. Kleene S. C. Recursive functionals and quantifiers of finite types revisited i // *Studies in Logic and the Foundations of Mathematics*. 1978. Vol. 94, № C. P. 185-222.
4. Mota S. The never-ending recursion // *Journal of Applied Logic*. Elsevier Ltd, 2017. Vol. 25. P. 89-108.
5. Kiefer A., Hohwy J. Content and misrepresentation in hierarchical generative models // *Synthese*. 2018. Vol. 195, № 6. P. 2387-2415.
6. Sacks G. E. Metarecursion theory // *Studies in Logic and the Foundations of Mathematics*. Elsevier, 1967. Vol. 46, № C. P. 243-263.
7. Chomsky N. Minimal Recursion: Exploring the Prospects // *Studies in Theoretical Psycholinguistics*. 2014. Vol. 43. P. 1-15.
8. Tomalin M. Reconsidering recursion in syntactic theory // *Lingua*. 2007. Vol. 117, № 10. P. 1784-1800.
9. Corballis M. C. Recursive Cognition as a Prelude to Language // *Language and Recursion*. New York, NY: Springer New York, 2014. Vol. 9781461494. P. 27-36.
10. Baryshnikov P. N. Language, brain and computation: from semiotic asymmetry to recursive rules // *RUDN Journal of Philosophy*. 2018. Vol. 22, № 2. P. 168-182.

11. Christiansen M. H., Chater N. Toward a connectionist model of recursion in human linguistic performance // *Cogn Sci.* 1999. Vol. 23, № 2. P. 157-205.
12. Piccinini G., Bahar S. Neural Computation and the Computational Theory of Cognition // *Cogn Sci.* 2013. Vol. 37, № 3. P. 453-488.
13. Kolodny O., Lotem A., Edelman S. Learning a Generative Probabilistic Grammar of Experience: A Process-Level Model of Language Acquisition // *Cogn Sci.* 2015. Vol. 39, № 2. P. 227-267.
14. Zhang Y., Amin N. Reasoning about reasoning about reasoning semantics and contextual equivalence for probabilistic programs with nested queries and recursion // *Proceedings of the ACM on Programming Languages.* Association for Computing Machinery, 2022. Vol. 6, № POPL.
15. Wittgenstein L. *Tractatus logico-philosophicus* // *Tractatus Logico-Philosophicus.* Anthem Press, 2021. P. 56-250.
16. MacLennan B. J. *Transcending Turing computability* // *Minds Mach* (Dordr). Springer Netherlands, 2003. Vol. 13, № 1.
17. MacLennan B. J. Natural computation and non-Turing models of computation // *Theor Comput Sci.* 2004. Vol. 317, № 1. P. 115-145.