An Integrated Syntax/Semantics Representational Framework for Natural Language Processing: Theory and Applications

Carlos Seror
Independent researcher
Valencia, Spain
email: serorcarlos@gmail.com

***Abstract*—Language can be described as a set of encoding/decoding rules whereby a receiver is prompted to locally reconstruct a relevant part of a sender's representation, intended as an update of the receiver's representation. In this paper, a representational framework is proposed for such description, based on (a) the cognitive feature of spontaneous categorization, which leads to a formal description of data referencing as a disambiguation process, (b) the identification of a number of irreducible structures underlying perceptual categories, consistent with the notion of semantic primitives. The general algebra describing data referencing could be seen as a universal syntax from which conventional languages can be derived and, conversely, into which conventional languages can be parsed. On the other hand, the semantic structures identified will be formalized into a denotational algebra reminiscent of, but not identical to, the topology of open sets. Both formalisms will be shown to converge into a representational framework having the two-way capability to generate language and encode meaning. The framework thus proposed reflects a qualitative approach to language, and therefore radically differs from Shannon's quantitative approach to communication. As a demonstration of its potential, the last sections will sketch two practical applications, i.e., (i) a representational interpretation of databases, and (ii) a tool to enhance deafblind people's cognitive world.**

Keywords-data representation; categories; natural language; syntax; semantic structures; data referencing; databases; deafblind people

I. INTRODUCTION

A list of previous attempts to decode and/or generate human language would be huge. However, aside from their generally remarkable intellectual merit, none of those attempts seems to have fully succeeded to date [6][23]. Furthermore, a focus on the structural aspects of syntax tends to overlook the nature of language as an information tool, and more specifically as a conveyor of meaning [30]. Besides, there seems to be a remarkable polysemy between information as static data content (for example, [44] along the lines of concept theory) and Shannon's dynamic interpretation, based on the transmission and acquisition of data streams.

Shannon's classical paper [41] explicitly disregards meaning as irrelevant to a quantitative approach to communication. Shannon's information theory is about symbol strings, and views communication merely as a quantifiable streaming of symbols, i.e., a succession of elementary information events. It does not, however, explicitly formalize the notion of information event, which should arguably be considered a first essential step to comprehend the nature of language as a communication tool. In this paper, a general definition of an information event will be proposed, and shown to lead to a comprehensive description of language, both in terms of syntax and semantics. That description will be derived from a representational approach to data structures based on (a) the cognitive feature of spontaneous categorization, implicitly used by conventional languages' users, and (b) the identification of structures underlying perceptual categories, which can be formally described and generalized, thereby opening the way to an objective theory of semantics.

Section II will discuss Shannon's quantitative approach from the standpoint of meaning, and propose a representational definition of an information event. In Section III, the concept of data aggregate will be introduced, and endowed with a simple structure based on the logical connectives $\wedge$, $\vee$. A string syntax derived from the resulting expressions will be shown to be consistent with conventional languages. In Section IV, the scope of the connectives will be enlarged to derive more general structures. Based on such structures, the two key components of an information event will be formally described in Section V. As a practical application, a parser will be sketched in Section VI, followed in Section VII by a formal description of the spontaneous categorization feature. Sections VIII and IX will deal with the relations between representations and meaning, based on the spatial adjacency relation. The identification of denotational structures underlying perceptual categories will be the subject of Sections X and XI, illustrated in Section XII with three elementary semantic structures. Section XIII will propose an interpretation of databases as category clusters. Finally, Section XIV will assess the potential of the proposed framework to facilitate the communication and/or enhance the cognitive universe of deafblind users.

II. INFORMATION EVENTS

Parrots can speak, but cannot really talk. This difference is arguably the key to what we understand as language. Parrots are able to send and receive information as a sequence of vowels and consonants, which is strictly sufficient to reconstruct a message, but hardly the information a human would expect from an act of communication. Therefore, a key question is: what do we precisely mean when we say 'information'?

Surprisingly, there seems to be no generally agreed definition of information [31]. In 1948, Shannon's classic paper addressed information as the "signature" of communication, but Shannon was an engineer and his basic concern was to ensure that two parties located at either end of a telephone line could convey their messages with an acceptable degree of noise. Even if one of the parties was a parrot.

At the beginning of his paper, Shannon stated: "Frequently the messages have meaning; that is, they refer to or are correlated according to some system with certain physical or conceptual entities. These semantic aspects of communication are irrelevant to the engineering problem [i.e., reproducing at one point more or less accurately a message selected at another point]" [41]. To Shannon, information reaches its destination when the message unequivocally does. The messenger's mission is to deliver messages, not understand them. However, if my parrot and I are in the living room and in the kitchen somebody shouts "Fire!", it can hardly be argued that both the parrot and I have received the same information. Therefore, the key issue to be addressed in human communication is: how do we convey meaning?

A formal definition of the communication process should be a key starting point to answer that question. A communication event could be described as a process whereby a sender encodes a number of instructions intended to replicate some part of its data repository. Communication could be said to succeed when the replication succeeds and results in an enlargement of the receiver's data repository.

We may think of a communication process as an assembling process: first, we point to some location in a data aggregate, then we assign some new item to that location. The item to be assembled is the information item the sender wants to convey, based on the conjecture that the item is missing in the receiver's repository. Such assembling operation can be formally expressed as a definition of an information event. For a communication event to take place, a sender and a receiver should previously agree on some set of —possibly implicit— conventions in order to (a) identify a location in their respective data repositories; (b) identify the data item to be incorporated. Thus, a communication event could be described as

$$\# \rightarrow \# \, \alpha \, b \qquad (1)$$

where # is a specific data item in a data repository, α is an assembling instruction, and b is a data item to be attached. The arrow denotes the transition between the two states in a data repository. A simple example of a communication event is a binary message, where # is the last bit received, b is the bit to be assembled, and α is an instruction to assemble b to # by using the one-dimensional adjacency relation.

As a further example, if *H0* is a binary string representing a specific horizontal line on a display and β is the adjacency relation between a line and the one immediately above, then the following expression would describe the assembling of a new line, represented as *H1*, on top of the extant *H0*, i.e.,

$$\mathrm{H0} \rightarrow \mathrm{H0} \, \beta \, \mathrm{H1}$$

In yet another example, if *T* represents a specific triangle and *Z* represents the shape of a zebra, then a receiver could implement the information event

$$\mathrm{T} \rightarrow \mathrm{T} \, \gamma \, \mathrm{Z}$$

where the symbol γ indicates that the shape *Z* is to be embedded in the shape *T*. The information conveyed in this example would be unacceptably vague for an engineer, but in most practical situations people are usually content to learn that a frog is inside a pond, and will not ask for the precise coordinates of the frog.

A fourth example involves a house as made up of identifiable parts. The expression

$$\mathrm{roof} \rightarrow \mathrm{roof} \, \kappa \, \mathrm{chimney}$$

would describe the assembling of a chimney on top of the house's roof by means of the three-dimensional adjacency relation κ. Again, this kind of information is largely imprecise, yet fairly acceptable in practice.

The actual identification in a data repository of the location # in (1) would require the existence of a referencing system shared by both the sender and the receiver. A referencing system could be implemented by introducing a structure in a data repository. In the following section, a possible such structure will be defined.

III. Data Aggregates

Data items could be arranged in a variety of ways, including representational (e.g., labels on a map), encoded (e.g., data streams), physical (e.g., material items in drawers), etc. From an abstract standpoint, any such arrangement could be described as a number of symbol aggregates endowed with a particular structure in the form of tables, graphs, objects or other ways of organization [28][14]. Natural Language (NL), being a means to deliver information, could also be argued to use data but, except in specific, explicitly structured subject areas, its users are usually unaware of the structure of such data. Describing a data structure consistent with NLs would therefore be a first step to characterize the nature of language as a tool to convey information.

For a general approach to a diversity of data arrangements, the term 'data aggregate' will be adopted here. A data aggregate is defined as a number of data items that could be represented as points on a surface. A data aggregate is arguably the minimal structure that can be conceived of, and it does not exclude other additional structures. Thus, a number of colors could be considered as a data aggregate, irrespective of whether they can be synoptically represented as points across a rainbow or on a painter's palette. In a data aggregate, items can be pointed to but do not have to be distinctly labelled —you may know nearly everything about a wood and not have a name for any of its trees—. Also, a data aggregate could be indefinitely updated, as long as any new item could be represented as an additional point on the same surface. Goats in a herd, or symbols on a paper sheet, are simple examples of data aggregates.

Items in a data aggregate could be discriminated by applying criteria to them. A criterion is a notion more general than a property, because it encompasses properties as well as fancy choices and algorithms. Two of the simplest criteria that could be applied to a data aggregate are the ones associated to intension and extension. Aggregations of items in a data aggregate do not have any extension or intension connotation *per se*, and are therefore objects more general than sets. Extension and intension could be implemented on them by means of resp. the logical connectives $\wedge$, $\vee$, e.g.,

$$\text{blue} \wedge \text{red} \wedge \text{yellow} \wedge \ldots \qquad \text{extension [colors]}$$
$$\text{blue} \vee \text{red} \vee \text{yellow} \vee \ldots \qquad \text{intension [color]}$$

In general, therefore, a number of items u_1, u_2, u_3, ... in a data aggregate could be discriminated in two alternate ways, i.e.,

$$u_1 \wedge u_2 \wedge u_3 \wedge \ldots \tag{2}$$
$$u_1 \vee u_2 \vee u_3 \vee \ldots \tag{3}$$

Because mathematical sets are said to be describable both in extensional and intensional terms, we shall rather not mix things up and separately discern either description instead. Hence, any expression in the form (2) will be referred to as a **combination**, while any expression in the form (3) will be referred to as a **category**. This difference reflects the use of plural resp. singular in human language. Thus, 'colors' could be associated to a combination, while 'color' could be associated to a category. When the scope of the criteria is not specified, the expressions (2) and (3) could also be interpreted as reflecting the difference between resp. 'every' and 'any'.

Any item u encompassed by a category C —i.e., complying with the criteria that define C— will be referred to as an **instance** of C. A category C encompassing the instances u_1, u_2, u_3, ... will therefore be expressed as

$$C \equiv u_1 \vee u_2 \vee u_3 \vee \ldots$$

The definitions of category and instance could be used as a means to locally refer to an item in a data aggregate. Indeed, in a data aggregate E where a category C has been identified, any instance of C could be expressed as a disambiguation of C. That is, if we denote a category as $C()$ and an instance u of that category as $C(u)$, we could refer to u as

$$C() \rightarrow C(u)$$

The expression above may be interpreted as a path in E, i.e., "select C, then select the instance u of C", where u could be identified by means of either a label or a number of instructions. In the following sections, an enlarged notion of disambiguation will be shown to be a powerful device to refer to data items in a data aggregate —arguably, the basic addressing device used by natural language users—.

A data item in a data aggregate could also be referred to through a disambiguation of a number of categories it might be ascribed to. For example, the word 'green' may denote either a color or a political affiliation. In the absence of any additional cues, they could be disambiguated resp. as either *color(green)* or *political_affiliation(green)*. In formal terms, if M is a category having the category C as an instance, then the instance $C(u)$ could be referred to as

$$M(u) \rightarrow C(u)$$

The notation used thus far, based on symbols such as connectives or arrows, will be referred to as **symbol syntax**. An alternative syntax, which shall be referred to as **string syntax**, would express categories and instances as single words, and disambiguations as strings, as follows:

symbol syntax	string syntax
C()	C
C() → C(u)	C [δ u]

where the symbol δ denotes the relation linking a category with any of its instances, i.e., the fact that u complies with the criterion that defines C. The correspondence between symbol expressions and string expressions will be denoted as >>, e.g.,

$$\text{color()} \;>>\; \text{color}$$
$$\text{color()} \rightarrow \text{color(blue)} \;>>\; \text{color [}\delta\text{ blue]}$$

Note that, in practice, if we deem it obvious that, e.g., the word 'blue' refers to a color, we will not precede it with the word 'color', which will have to be guessed by the receiver. This data compression feature reflects an implicit operation that pervades human language —and arguably also human thought—, i.e., spontaneous categorization. The feature of spontaneous categorization will be further elaborated in Section VII.

IV. Combined Categories and Category Clusters

The connective $\wedge$ could also be used to discriminate combinations of categories in a data aggregate. For example, from the categories

mass, electric_charge, spin

a combined category could be derived, which in turn would give rise to a number of objects, e.g.,

mass ∧ electric_charge ∧ spin >> particle
mass ∨ electric charge ∨ spin >> observable
mass(9.1×10^{-31} kg) ∧ electric_charge(-1.6×10^{-19} C) ∧ spin(1/2) >>
particle [δ electron]

As the latter example shows, a combination of categories is itself a category, having as instances combinations of instances of its component categories. The latter example unequivocally describes the item ‘electron’ as a full disambiguation of the category ‘particle’. However, combined categories could also be partially disambiguated by specifying just some instances of its component categories, e.g.,

bearing() ∧ altitude(7000 ft.) ∧ No._of_passengers(80)

Component categories in a combined category could also be discriminated by means of the connective ∨. The resulting object could be used to identify a path within the data aggregate individually leading to them. For example, the category ‘color’ could also be construed as an attribute, i.e., it could be referred to as an instance of the category

color ∨ shape ∨ size ∨ ...

Categories pervade language, a fact which is obscured by the spontaneous categorization mechanism, of which language users are usually unaware. In human languages, spontaneous categorization is a run-of-the-mill feature, associated not only to adjectives such as ‘blue’ or ‘big’, but to virtually any kind of meaningful component. Thus, the sentence ‘birds fly’ could be misleadingly categorized as implying that hens fly, or meaningfully categorized by instead interpreting ‘birds’ as an instance of some category, e.g., birds ∨ mosquitos ∨ bats ∨ ... having ‘fly’ as an attribute. Similarly, the meaning of ‘a through person’ could only be captured by evoking a category of concepts having ‘through’ as an instance.

In general terms, a **combined category** G^{η} will be defined as the general expression

$$G^{\eta} = O1 \wedge O2 \wedge O3 \wedge \ldots \quad (4)$$

where *O1*, *O2*, *O3*, ... are categories, whether they have been disambiguated or not, and η uniquely identifies that particular combination of categories. The definition (4) is consistent with a number of concept theories [45][47], that describe concepts as n-tuples of symbols representing attributes.

An n-tuple is just a one-dimensional combination of categories, and therefore a particular case of the more general concept of **category cluster**, where complex spatial relations could be incorporated as additional discrimination criteria in a data aggregate. A data form is a familiar example of category cluster. In general terms, therefore, a **representation** can now be defined as a data aggregate together with any number of category clusters.

V. Reference and Updating within a Category Cluster

Given a category cluster *G* and one of its component categories *C*, any set of instructions *r* to uniquely identify *C* within *G* will be referred to as a **relation** *r(G, C)*. As in the one-dimensional case, specific category clusters could also be referred to by specifying one or more of its component categories. For example, an employee’s record might be uniquely identified by specifying just the employee’s name, or his age and height. This could be formally described as follows. Let *G* be a category cluster, *r* a set of instructions to identify *C* within *G*, and G_k a copy of *G* where the data item *u* has been specified for the category *C*. The category cluster G_k could therefore be referred to as

$$G_k = G \mid r(G, C(u)) \quad (5)$$

i.e., G_k can be interpreted as a partial disambiguation of *G*. In string syntax, this will be expressed as

G | r(G, C(u)) >> G [r u]

For example,

ball = shape(round) ∧ color() ∧ size()
ball_k = shape(round) ∧ color(red) ∧ size(big) >> ball [r_2 red] [r_3 big]

where r_1, r_2, r_3 would represent resp. the sets of instructions to locally identify each of the component categories ‘shape’, ‘color’, ‘size’. If a reference would not result in a full disambiguation, further disambiguating [r u] legs could be appended until a unique reference is achieved. In the general case, therefore, the disambiguation of a category cluster *G* will be expressed in string syntax as

$$G\ \Sigma[r_j\ u_j] \quad (6)$$

where Σ denotes a string made up of $[r_j\ u_j]$ pairs, u_j denotes an instance of the component category C_j, and r_j denotes the relation $r_j(G, C_j)$. The possibility to uniquely identify a category cluster even when only some of its component categories have been specified is a feature heavily used by natural language users as a data compression device. Indeed, if there is only one red ball in the room, you would hardly want to refer to it as “the big red expensive air-filled ball on the sofa”.

Any set of rules to convert string syntax expressions into different strings will be referred to as a **conventional** syntax. For example,

String syntax	*Conventional syntax*	
ball [r_2 red] [r_3 big]	big red ball	(English)
boule [r_2 rouge]	boule rouge	(French)
bam [r ug]	bugam	(imaginary)
✌ [r_2 👌] [r_3 🖐]	👌 ✌ ♒ 🖐	(non-word)

The notion of category cluster, plus the relations it entails, endow data aggregates with powerful structures that could be used to semantically represent a vast number of concepts, e.g., ontologies, verbs, or semantic representations of space-time concepts, together with a local mechanism to refer to them [37][1][40]. A few basic semantic clusters will be described in Section XII.

The definition of the cluster-category relation also implies that categories could be referred to in terms of the cluster or clusters they are part of. This stems from the definition of the converse relation. Given a relation *r(G, C)*, the expression

$$C \rightarrow C \mid r(G, C) \tag{7}$$

describes the constriction of the general category *C* to the range of instances allowed in *G*. In string syntax, (7) will be expressed as

$$C\ [r'\ G] \tag{8}$$

and *r'* will be referred to as the **converse relation** of *r*. Example:

$$\text{color} \mid r_2(\text{ball, color}) >> \text{color}\ [r'_2\ \text{ball}]$$

If we now substitute (6) into (8), the general expression will therefore be

$$C\ [r'\ G\ \Sigma[r_j\ u_j]] \tag{9}$$

When G $\Sigma[r_j\ u_j]$ describes a full disambiguation, the expression (9) will point to the unique content of *C* in *G*, i.e., it could be used to indirectly refer to a specific component instance in a category cluster. For example, the expression

$$\text{color}\ [r'_2\ \text{ball}\ [r_1\ \text{big}]\ [r_{on}\ \text{sofa}]]$$

might uniquely refer to the color of the ball on the sofa without directly using its name. The expressions (6) and (9) may be seen as a formalization of the Fregean concept of sense (Sinn) [21], which would interpret the notions of Sinn and Bedeutung as part of a wider picture. Thus, in Fregean terms the category-instance expression C(John) would be the meaning of 'John', while the indirect references 'the tall man' or 'the man with my hat' could be used to express two of the multiple possible senses of the referent 'John'.

The expressions (6) and (9), used to refer to resp. category clusters and component categories, could be used not only to refer to data items in a structured data aggregate, but a sender could also used them to update the receiver's presumed representation. For example, if the receiver is believed to ignore that there is a ball on the sofa, then that information could be sent by means of (6), i.e.,

$$!\text{ball}\ [r_4\ \text{sofa}]$$

where the symbol ***!*** denotes a new category cluster to be included in the receiver's representation. Such updating is a commonplace device used in natural language exchanges, to indicate, e.g., that a new character has appeared in a film, or a new guest has arrived at a party. If the receiver were presumed to know that there is a ball on the sofa but not its size, then that information could be conveyed by means of the expression

$$\text{ball}\ [r_4\ \text{sofa}]\ [r_3\ \text{big!}]$$

where ***!*** now denotes an instance intended to fill a category presumed to be empty at the receiver. In a converse situation, where the sender ignores some information item supposedly known by the receiver, the expressions (6) and (9) could also be used for querying purposes, by pointing to the required item by means of a different symbol, e.g.,

$$?\ [r_4\ \text{sofa}]$$
$$?\ \text{size}\ [r'_3\ \text{ball}\ [r_4\ \text{sofa}]]$$

where the symbol ***?*** points to resp. an category or instance unknown by the sender. The referencing and active/passive updating uses of (6) and (9) could be summed up as follows

reference	G [r u], C [r' G]	(10)
updating	!G [r u], G [r !u]	(11)
querying	? [r u], ?C [r' G]	(12)

Depending on the rules devised to derive specific syntaxes from the general expressions (6) and (9), a large number of unfamiliar grammars could be built, whether or not in use by any communities of users. This should make it possible to test the validity of the approach developed above. Indeed, that validity would be challenged if some grammar in use were found whose syntax rules could not be derived from (6) and (9). Conversely, a weak confirmation could be obtained by checking whether a number of 'exotic' syntaxes could be derived from (6) and (9). Amazonian pirahã [18] and Australian warlpiri [35], among others, would seem to be good candidates [17].

VI. PROPOSAL FOR A PARSER

Based on the general expressions (6) and (9), a variety of parsers from conventional syntaxes into string syntax could be devised by expressing lexical/morphological components in terms of categories, instances, and relations. Although any such component is potentially susceptible to be categorized —the atypical syntax of 'a through person', mentioned above, provides a telling example—, a basic list of usual category/instance values could be established as follows:

adjective	u
preposition	r
noun	G
verb	G [r u]

where *u* denotes an instance, *G* denotes a category cluster (or, in the simplest cases, a category), and *r* denotes a relation. Note that verbs have been expressed as tense-carrying items, where *G* denotes the verb root and the *u* in [r u] denotes an instance of the tense category 'past/present/future/...' Based on the list above, qualifying pairs, such as 'big ball', 'car wheel', 'take book', or 'run fast' would be expressed as disambiguations in the form G [r u]. Therefore, as a first step a number of string segment types could be identified, i.e.,

SL	*noun* $\Sigma[r_j\ u_j]$ *verb*
SR	*verb* $\Sigma[r_j\ u_j]$
W	*noun* $\Sigma[r_j\ u_j]$
M	*preposition noun* $\Sigma[r_j\ u_j]$
A	*adjective* $\Sigma[r_j\ u_j]$
v	*verb* [r *tense*]
n	*noun/verb root*
α	*adjective*

where each individual leg [r u] would express a qualification of the preceding component. A few English examples may illustrate this, i.e.,

SL	*birds* [] *fly*, *a book* [on the shelf] *fell*
SR	*ran*, *saw* [her] [with a telescope]
W	*the book* [from the shelf], *happiness* []
M	*under* the [milk] *wood*
A	[nearly] *perfect*
v	[would] *hope*
n	*wheel*, *pampering*
α	*big*, *unthinkable*

Based on such components, the following prolog-like basic rules could be stated, together with their output:

is(x v, SL) :-	is(x, W)	→ v [b x]	*1
is(v x, SR) :-	is(x, W)	→ v [b x]	*2
is(v x, SR) :-	is(x, W M)	→ v [b x] M	*3
is(v x, SR) :-	is(x, W M M)	→ v [b x] M M	*4
is(x, A) :-	is(x, α)	→ α	*5
is(x n, W) :-	is(x, A)	→ n [b x]	*6
is(x n, W) :-	is(x, n)	→ n [b x]	*7
is(x b y, W M) :-	is(x, W), is(y, W)	→ x [b y]	*8
is(x r y, W) :-	is(x, W), is(y, SR)	→ x [r y]	*9
is(x r y, W) :-	is(z, W), is(y, SL)	→ x [r y]	*10
is(b x, M) :-	is(x, W)	→ [b x]	*11

Example: S ≡ the book on the shelf fell

is(S, SL) :- is(book on shelf, W) → fell [b book on shelf]	*2
is(book on shelf, W) :- is(book on, α) → fail!	*5
is(book on shelf, W) :- is(book on, A) → fail!	*6
is(book on shelf, W) :- is(book on, n) → fail!	*7
is(book on shelf, W) :- is(book, W), is(shelf, W) → book [on shelf]	*8

Output: fell [b book [on shelf]]

An educated guess based on a number of partial implementations by the author suggests that a few hundred rules would probably suffice to process most sentence types.

VII. Spontaneous Categorization

Parsing is a way to convert NL strings into combinations of (6) and (9), but cannot always be used to decide whether such expressions are to be interpreted as (10), (11) or (12), i.e., whether they are intended as reference, updating or querying. Querying purposes are usually denoted in various ways, including characteristic sentence structures, question marks, and/or prosodic patterns, but updating purposes (i.e., predication) may not always be obvious, at least in written form. For example, in Maya language the written expression *keel winik* can be interpreted as either *the man is cold* or *the cold man* [43]. In such cases, deciding whether a NL message is meant as (8) or (9) should be the job of spontaneous categorization.

Spontaneous categorization has been identified and studied from the standpoint of language use [15][40], but also in children [26][7][32], nonhuman primates [24] and even distantly related species [25]. Interestingly, a comment in Shannon's seminal paper [41, op. cit.] hints at the role of categories in NLs: "The significant aspect [of communication] is that the actual message is one *selected* from a set of possible messages" [emphasis added].

In the binary case, the set of possible messages is easy to determine, since it can be derived from the category 0/1. The information conveyed by Beethoven's Ninth Symphony, though, may be harder to determine, because one category of possible messages is the category of all possible symphonies. Fortunately, however, some partial information can be extracted from it. For example, we can determine that it consists of four movements, what kind of movements they are, how many instruments are playing it and, ultimately, each of its notes.

NL strings can similarly be decomposed in different ways, which is particularly apparent from the use of questions. As an example, the questions

> who left at eight?
> when did Joan leave?
> what did Joan do at eight?

refer to different categories, i.e., person, time, and action, all of them implicit in the expression 'Joan left at eight'. Thus, depending on the part of the message that may be selected and the category inferred from it, one single expression could be used to refer to different information items. In most cases, the categories would be implicit, and its identification would be left to the receiver through a spontaneous categorization of the message received.

The following example might help to clarify this. Suppose that you are at home, sitting in front of your TV screen, when suddenly the telephone rings. You answer the call. It's your friend Zoe.

> "Hi, Zoe. No, I didn't feel like going out tonight. I'm watching a film."

It is 9:15 PM on a Monday, so your friend is not surprised. Now imagine that you and her have instead decided to go to the movies that day so that, at some point, the two of you are watching a film together. Suddenly, you move closer to Zoe and whisper:

"I'm watching a film."

Of course, this is no news for Zoe, who, puzzled at first, faces subsequently a critical decision: either you have become mad, or you were meaning something. Apparently, your statement has not provided her with any information. She already knows that you are watching a film. Her mind works frantically. What could you have possibly meant?

Zoe comes up with a number of possibilities. Perhaps it has been a long time since you last watched a film, and you are just expressing joy. Or you might happen to work in the archives of a film library and are usually busy handling films without ever getting to watch any of them. Or perhaps you are a fanatic of theatre and tonight, exceptionally, have condescended to go to the movies. Thus, what Zoe's mind would be doing is to construct a background against which to extract information from your message. As she might find out, any of the following constructions would be apt to provide plausible information from your message:

"For a long time, I *have not watched* a film. Now I *am watching* a film"
"Usually, I *handle* films. Now I *am watching* a film"
"Usually, I watch *theatre plays*. Now I am watching *a film*"

Each of these interpretations conveys the information that something is happening that did not use to happen. To extract information from your message, Zoe has had to mentally construct resp. the categories

not watch / watch
handle / watch / ...
a play / a film / ...

In more abstract terms, she has constructed the ambiguous messages:

I am X
I am X a film
I am watching X

and, based on a single instance taken from your message, has subsequently let the categories X spontaneously form in her mind. The information she will eventually extract from your message will depend on what those categories actually encompass and how their contents fit into the information she already has about you. The ability to infer a category from one of its instances, i.e.,

$$b \cdots\!> C(b)$$

is what has been referred to as spontaneous categorization, and would thus seem to be a prerequisite to process a NL message, and possibly a distinctive feature of human brains. It should be noted that some categories may not have a name themselves, thereby exposing a lexical gap. Thus, 'green', 'red' and 'blue' could be ascribed the category 'color', but there is no single word in English for the category that encompasses the states green, ripe and intermediate ones as applied to a fruit (although the derived noun 'ripeness' is sometimes remedially used, as in 'degree of ripeness').

If you and Zoe had instead talked on the phone, the process would have been simpler but, essentially, not different. Upon hearing your reply, Your friend Zoe would simply have evoked a category of actions to be expected from you in your place on a Monday at 9:15 PM, i.e.,

I am X

where X = eating / sleeping / reading / watching a film / ... From the moment she had you on the phone, Zoe was predisposed to wonder what you might be doing at the other end of the line, and the words "watching a film" would be a good answer to that. However, in the presence of an obvious context, like the film the two of you were watching, the effort required from Zoe is paradoxically far more demanding. The key to Zoe's clairvoyance on the phone is that she had a much smaller number of choices as to what your message could be referring to.

Depending on the communication context, spontaneous categorization makes it even possible to dispense with grammar rules. Spontaneous categorization is what enables us to understand ill-formed expressions in colloquial utterances, or due to some lapsus linguae, or uttered by a foreigner with a non-standard syntax. For example, a sentence assembled with the words "place", "cheap", "eat" and pronounced by a likely hungry foreigner could be easily interpreted as

?place [r_1 eat [r_2 cheap]]

VIII. Code vs. Meaning

In Section IV it has been shown that category clusters can be used to generate syntax. However, neither the nature of categories, nor of the relations that 'glue' them into category clusters, have been addressed. Thus stated, the task looks to be huge, but might be at least partially manageable. While the structure of a data form, or a database table, is usually designed for convenience, ontologies and other spatial layouts tend to be representational, i.e., are intended to express meaning in a more direct way. Therefore, it may be worth exploring to what extent category clusters could be used to directly express meaning.

Parsing —and, more generally, message decoding— is a rather convoluted way to extract meaning. It is based on the capability to identify groups of symbols connected to each other by the one-dimensional spatial adjacency relation. The actual relations that those symbols are intended to reflect are only accessible through a number of conventions on how to group them up and translate them into meaningful representations. However, symbol strings could also be used to directly express meaning, i.e., as representations themselves. For example, the string

$$\text{Su Mo Tu We Th Fr Sa} \tag{13}$$

is intended to directly represent temporal relations, i.e., it reflects a semantic intent, and to that effect the way it has been constructed —and therefore its information content as a message— is of little relevance. Ironically enough, whereas for Shannon's purposes the semantic aspects of a symbol string are irrelevant, for semantic purposes it is the information content of a symbol string which is irrelevant. In representational terms, the string (13) is a category cluster and, as such, it already reflects a key advantage of the representational approach, i.e., unlike messages, which are updated through the rigid process of streaming, category clusters can be updated locally. A simple illustration of this potential can be derived from binary strings.

A binary string can be seen as a category cluster consisting of a single category, i.e., 0/1, and a single relation, i.e., the spatial adjacency relation linking each binary digit to the next one. A binary information event could be described by the general expression (1), where # denotes an existing binary string, *b* denotes a bit to be assembled, and α denotes the one-dimensional adjacency relation between any two consecutive bits. The location where each new bit is to be assembled is referred to by means of a meta-reference, i.e., 'the last bit received'. However, if the final purpose is to reconstruct a bit string rather than assemble it in a sequential way, it might be more convenient to make use of local referents. Once we have put the chimney in place, we may want to install the front door, and to do that the reference to the last item installed would be useless.

In a bit string, a simple way to refer to a local component is to identify a unique feature in the string. For example, let *M* be the following message:

1 1 0 1 0 1 1 1 0 1

The string above includes a number of unique substrings, e.g.,

1 1 1
1 0 1 1
1 1 1 0 1

The sender could not refer to the last bit received as '1', which would be ambiguous at the receiver, but it could refer to the unique substring '1 1 1 0 1' instead, and instruct the receiver to append the bit *b* by means of the adjacency relation α, i.e.,

$$1\,1\,1\,0\,1 \rightarrow 1\,1\,1\,0\,1\,\alpha\,b$$

Thus, by using unique substrings as referents, the sender could choose to instruct the receiver to construct just parts of *M*, instead of sending the whole *M*. For example:

$$1\,0\,1\,1 \rightarrow 1\,0\,1\,1\,\alpha\,1$$
$$1\,0\,1\,0 \rightarrow 1\,\alpha\,1\,0\,1\,0$$

This would be a local approach, insofar as each event would describe only a partial reconstruction of *M*. It may not sound very appealing as an alternative to the good old 'next-after-last' convention. But if our source were a two-dimensional matrix instead of a one-dimensional string, things would start to look different. For example, on a display described by a digital matrix *M*, any solid image of an object will be described by a subarray of *M*, and the general expression (1) could be used to describe the assembling of a subarray *u* to the subarray # by means of the two-dimensional adjacency relation *r*. Thus, if # represents the image of a rug and *u* represents the image of a ball, then a mutually agreed definition of *r* would make it unnecessary to transmit a whole matrix *M* describing a room just to indicate that there is a ball on the rug. Such local approach dispenses with the need to describe any parts of a data aggregate that the sender deems irrelevant. It may entail the use of 'vague' descriptions, as with 'the frog in the pond'. However, the gain in nimbleness is huge.

IX. DENOTATIONAL COMPRESSION

The mention of 'vague' descriptions points to an additional feature of representations, i.e., the fact that they can compress information by making use of objective relations such as the spatial adjacency relation. Not unreasonably, the human mind tends to simplify things. If you live in Paris, you will probably be able to lay out a mental map of the town and then locate the Mona Lisa on it. But if you have always lived in an Amazonian tribe, isolated from the external world, and one day you hear about Nessie, described to you as some unspecified monster that lives in some faraway lake, could you still be expected to mentally represent Nessie? Most likely, your representation will hardly be good enough for you to reach Scotland. But, even so, you might be capable of mentally representing the Earth as a sphere (maybe simply as a plane!) having an island somewhere on its surface, in the island a lake, and in the lake an animal whose shape you cannot tell. As long as you do not assign a specific shape to the island, the lake, and the monster, your representation of them would somehow 'float' in some vague —but certainly structured— territory of your mind.

This seems to suggest that the human mind makes use of preexisting configurations, or 'structures', to lay out the information derived from perceptions, a topic that has already been addressed from different perspectives [19][4]. Short of additional data, we are often content to know that an island is a surface inside another surface, or that Nessie is some three-dimensional volume inside a volume. Or, most simply perhaps, a dot somewhere on a surface. The point here is that a perceptual representation is alright as long as it is made up of territories and borders and these do not break up. Subject to those rules, perceptual concepts are largely malleable.

This is probably not by chance. Imprecise representations happen to be a good way to compress information. For example, the image of a lake surrounded by land could be roughly emulated by means of 0s and 1s:

... 0 0 0 0 0 ...
... 0 0 1 1 1 1 0 0 ...
... 0 0 1 1 1 1 0 0 ...
... 0 0 0 0 0 ...

that is, as an ‘area’ made up of adjacent 1s (water) and surrounded by an ‘area‘ made up of adjacent 0s (land). Now, denotationally speaking it would have sufficed to write:

0
0 1 0
0

where the symbol 1 represents a connected surface, and the four 0s represent a surrounding open surface externally adjacent to it. If needed, the compressed figure above could be expanded into the uncompressed one by ‘unfolding’ (i.e., replicating) the 1s and 0s as wished, in all directions, as long as the adjacency relations do not get breached. This is to say that the 1s and 0s would unfold into themselves, and an unfolding such as

... 0 0 0 ... -> ... 0 1 0 ...

would be a violation.

Actually, the representation above could be as useful to denote a lake as an island —or, for that matter, a stain of spaghetti sauce on someone’s shirt—. From such abstract representation, specific denotations could be derived by associating to the 0s and 1s the symbols ‘water’ and ‘land’, or ‘land’ and ‘water’, as the case may be. The underlying structure, however —that is, the conglomerate of adjacency relations— will stay unchanged.

The implication is that there exist denotational representations that are independent from the denoted object. Such representations may or may not be maximally compressed. If it were not possible to further compress one such representation, then we would be in the presence of an irreducible concept —i.e., a semantic primitive—. Certainly, it is not possible to compress a line into a dot, a tree into a line, etc. without altering their adjacency relations.

This would suggest that the structures human brains use to ‘interpret’ perceptions stem from a more abstract reality. Might such structures be a repertoire of topological configurations? That would be really good news. If the meaning of words were ultimately irreducible denotational configurations, human language would be based on an absolute referent: geometry. Meanings could be systematized, studied and compared according to objective criteria. Concepts could be handled by means of symbols and, consequently, would be computer-processable.

X. CATEGORY STRUCTURES

Spontaneous categorization has been described as an operation whereby categories are evoked to accommodate symbols received, presumably in such a way that those symbols could be subsequently retrieved. Therefore, it seems legitimate to wonder if such categories have some identifiable structure that makes them readily accessible for both storage and retrieval. Furthermore, our neuronal system cannot be impervious to the reality that space can be decomposed into three dimensions, that colors form a continuum, or that musical notes sound one after another.

The role of space and time structures in perceptual and cognitive processes has been extensively addressed in the literature from both the theoretical and experimental standpoints [22][50]. The precise relation between perceptual and cognitive structures is still being delved into [10], but the recent encompassing notion of cognitive architecture [36] opens the door to a functional approach to cognitive processes that is rather independent from their biological basis.

Therefore, if it would be possible to formally characterize category structures in terms of spatial and temporal relations, such relations would naturally give rise to objective category clusters of a semantic nature, and hence to at least some building blocks of meaning.

In fact, a close look at categories does reveal substrates that are intrinsically different in nature, and where the spatial adjacency relation plays a key role. In the preceding Section, the role of adjacency relations in spatial representations has already been noted. As it happens, a number of perceptual categories could also be shown to reflect underlying structures involving adjacency relations. Consider, for example, the different categorizations the undefined word *splack* elicits in the following sentences:

The sauce tasted too splack, so I added some garlic

I waited for you from noon to splack

The stripes blue, splack and yellow on her skirt

Bugs Bunny turned into a splack, then into a donkey

He pointed to the splack of the triangle

The categories that could be evoked from the examples above entail different underlying structures, i.e., resp.

- A number of individual, fuzzily connected concepts (tastes)
- A 1D continuum of concepts spanning from noon to splack (time values)
- A 1D, rainbow-like continuum of concepts (colors)
- A continuum of concepts spanning from splack to donkey (shapes)
- A finite number of spatial features (vertices/sides/areas)

The structures thus evoked are characteristically different in nature. The taste ‘salty’ could become ‘sweet’ by progressively adding salt and removing sugar, and a similar transition could be devised between any two other tastes. Intermediate tastes, however, are fuzzy, i.e., there seems to be no referent to precisely locate them with relation to others. No matter how many new intermediate tastes we identify, the structure will reappear, and any two tastes will always be separated by a conceptually blurred territory.

Time events do not have the same properties as tastes. We tend to think that an event is farther in time the more events can be piled up between it and now, but we cannot foresee how soon the next shooting star will show up in the sky. However, we can locate along time —e.g., on a clock's face, or in a calendar— any event that happens between two other identifiable events. And we can do it unequivocally, i.e., we cannot conceive of two different time paths connecting one event to another.

Color ranges are categories familiar to those who work with spectrometers, or check a catalog in a cloth shop. Color ranges have a one-dimensional structure, and share a number of features with time spans. But, unlike time, which is unlimited towards both the future and the past, a color wheel can be constructed that closes upon itself, and therefore colors are consistent with a circular structure.

Unlike time or color, the category that encompasses Bugs Bunny's and other intermediate shapes is two-dimensional. Bugs' shape could turn into almost any two-dimensional shape, and there are uncountable ways for it to turn into a donkey. While we can mentally 'travel' from noon to splack to midnight in the same way as a point would travel along a one-dimensional line, Bugs Bunny's shape could evolve through an infinite number of shapes before appearing as a splack. Even so, that evolution could be decomposed into three different steps. During a first stage, the intermediate shapes between Bugs and e.g., a donkey will remind us of Bugs for a while, then will seem unrecognizable for some time and, in the final stage, will remind us ever more vividly a donkey's shape. In representational terms, those three stages are not unlike transitions between tastes or colors.

Lastly, in the 'splack of the triangle' example, any categories that could be evoked from a triangle are inherently finite, and no two of them will be connected by a fuzzy territory. A vertex is immediately adjacent to two sides, a side is immediately adjacent to two vertices, and the inner and outer area of the triangle are immediately adjacent to the triangle's contour and any of its components. Unlike tastes, time values, colors, or shapes, which could be indefinitely updated, none of those categories could possibly be updated and still be thought of as making up a triangle.

Similarly, a data form made up of delineated cells is usually laid out as a two-dimensional structure. As long as its cells' borders are kept from merging or breaking, its structure could be deformed at will without essentially altering its functionality. In other words, what characterizes a data form with relation to other conceivable forms is, precisely, its specific adjacency relations.

All of these structures and their properties strongly remind of a branch of Mathematics known as Topology. With a few intriguing differences, though. Broadly speaking, Topology characterizes objects based on their potential to transform into each other without breaking or merging at any point. It conceptually groups geometric objects in terms of features such as the lines, surfaces or volumes they are adjacent to.

However, open-sets Topology differs from denotational topology in a few respects. Thus, if a circle gets broken at one of its points, the breaking point can only stay adjacent to one, not both, of the free ends, and the resulting object will be a stretch of line lacking one accumulation point. From a cognitive perspective, though, both ends will be equally denotable, irrespective of the fact that one of the ends is, in mathematical terms, a missing accumulation point. Conversely, for a segment to be made into a circle the names of its two ends would have to merge into one, thereby dropping one denoting symbol. In sum, for denotational purposes whenever you change the adjacency you have to add or remove labels —i.e., switch to a different *denotational* structure—.

Topology, on the contrary, is not concerned with labels. Topology is about open sets, not representations. Therefore, we will not be concerned here with the mathematical continuity of a representation, but with whether and how a spatial configuration can be consistently denoted (i.e., structurally identified and labelled), as well as with the characteristic properties inherent to it. This is to say that the human mind compresses sensory information by retaining only discontinuities, i.e., the adjacency relations between regions of different dimensions (points to lines/surfaces, lines to surfaces/volumes, etc.). Which makes sense, because where there is no discontinuity there is nothing to denote. As was the case with the unnamed trees in a wood, denotability does not mean that a symbol should be automatically attached to each identifiable feature, but only that these can be used as symbol holders, whether we decide to 'fill' them or not.

The characteristic structures that the spatial adjacency relation gives rise to could be formalized by introducing the concept of *denotational jigsaw*.

XI. DEFINITION OF A DENOTATIONAL JIGSAW

A **denotational jigsaw** is defined as a number of objects z_1, z_2, z_3, ... together with a number of adjacency relations $k_{mn}(z_p, z_q)$, where m is the dimension of z_p, n is the dimension of z_q, and $m \neq n$, so that for any two objects u, v in the jigsaw there is always a 'path' connecting u to v, i.e., in a simplified notation:

$$k(u, w)\ k'(w, b)\ k''(b, c)\ ...\ k^{(j)}(y, v)$$

In this simplified notation, the symbols k, k', ... $k^{(j)}$ denote the corresponding adjacency relations. Note that an adjacency relation is symmetrical:

$$k_{mn}(u, w) = k_{nm}(w, u)$$

but not identical nor transitive:

$k_{mm}(u, u) \Rightarrow m \neq m$
$k_{mn}(u, w), k_{nr}(w, z) \not\Rightarrow k_{mr}(u, z)$

Thus, a surface containing a circle B_1 with an inner area B_2 and an outer area B_3 could be described as the jigsaw (Figure 1).

B_1, B_2, B_3

$k_{21}(B_2, B_1), k_{12}(B_1, B_3)$

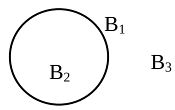


Figure 1. Circle-within-an-area jigsaw

In this configuration, the inner area B_2 is adjacent to the circle B_1, but not to the outer area B_3, and no adjacency relation between B_2 and B_3 can be transitively inferred from $k_{21}(B_2, B_1)$, $k_{12}(B_1, B_3)$.

The identification of a point B_4 within B_1 would transform the circle B_1 into a closed segment, thereby creating a more complex set of adjacency relations (Figure 2).

B_1, B_2, B_3, B_4

$k_{21}(B_2, B_1), k_{12}(B_1, B_3)$
$k_{02}(B_4, B_2), k_{02}(B_4, B_3)$
$k_{01}(B_4, B_1)$

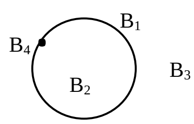


Figure 2. Point-within-a-circle-within-an-area jigsaw

For different values of m, n, the following simple paths can be identified:

k_{10}(line, point)
k_{20}(area, point)
k_{30}(volume, point)
k_{21}(area, line)
k_{31}(volume, line)
k_{32}(volume, area)

which could be assembled into more complex ones, e.g., a triangle, a polyhedron, or a category cluster. Insofar as dimensions are themselves irreducible, the above paths are irreducible, and are not incompatible with each other.

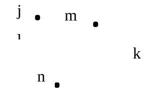
Some of those complex configurations, e.g., a triangle, can be described in terms of a finite number of adjacency relations. Unlike finite configurations, however, towns on a map, or events along time, are open configurations, i.e., they can be indefinitely updated. New elements could be added to them without altering its denotational nature, and therefore they can be described as a self-similar structure, based on a finite number of updating rules.

For example, if we denote as α the adjacency relation $k_{30}(v, i)$ between a point i and its surrounding volume v, the same relation α could be used to describe any number of additional points j, k, ... within that same volume, i.e.,

$v\ \alpha\ i,\ v\ \alpha\ j,\ v\ \alpha\ k, ...$

and the identification of a new point n could be described by means of the rule

$v\{...\} \rightarrow v\{...\ n\}$ (S0)



where

$v\{i, j, k, ...\} \equiv v\ \alpha\ i,\ v\ \alpha\ j,\ v\ \alpha\ k,\ ...$

Events along time, or towns along a railway, however, fail to be describable by means of the updating rule (S0), and their description requires a different set of rules. If we denote as β the adjacency relation k_{10}(line, point), any two points i, i will be adjacent to a one-dimensional stretch e separating them, i.e.,

$i\ \beta\ e\ \beta\ i$

Therefore, the identification of a new point, e.g., a new event along time, or a new town along a railway, could be described as

$e \rightarrow e\ i\ e$ (S1)

where

$e\ i\ e \equiv e\ \beta\ i\ \beta\ e$

Actually, the structure described by (S1) is a dual structure, since it concurrently generates two different collections of elements, i.e.,

... i, i, i ...
... e, e, e, ...

Rather counterintuitively, the rule (S1) does not describe the structure of events along time as the result of assembling additional events from the present into the future, but rather as the result of nesting new events within time spans. This is what makes it possible for a receiver to identify and store previously unidentified events in the past. It implies a construal of time not as a repeated realization of future events —as the physicist Eddington put it, “events do not happen: they are just there, and we come across them” [16]—, but as a pre-existing blank stretch into which events can be indefinitely inserted [27].

The rules (S0) and (S1) provide a means for a sender to describe a representation by means of symbols and rules in such a way that, however the sender updates it, the receiver’s description will be consistent with it. Short of any geometric referents, the receiver may have no way to know what the received symbols refer to, but could nonetheless make use of a number of agreed rules to reconstruct their configuration.

The structures just described are based on intrinsic features of the geometry of things as we perceive them, and are indeed characteristic and irreducible. You can pick one olive with a toothpick or two with a fork, but no matter what you do you will never be able to simultaneously pick more than one olive with a toothpick.

The configurations described by the rules (S0) and (S1) are irreducible, but not incompatible with each other. For example, on a map where towns have been represented as dots and the remaining surface as *U*, the items *Vienna* and *Rome* could be connected through the denotational path

$$k_{02}(\text{Vienna, U})\ k_{20}(\text{U, Rome})$$

If those towns happened to be linked by a railway R, then they could alternatively be connected as:

$$k_{01}(\text{Vienna, R})\ k_{10}(\text{R, Rome})$$

In both cases the category of towns would be the same, but the structure used to represent that category would be different.

XII. Semantic Clusters

Interestingly, a particular kind of category clusters can be derived by combining (S0) and (S1), resulting in expressions consistent with evolution verbs. The following example describes the concept of movement, but the same configuration could also be used to describe color change, or growing/shrinking (Figure 3).

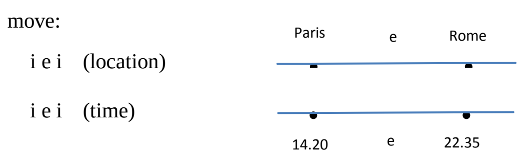


Figure 3. Category cluster describing movement

Similarly, a combination of the configuration above and the configuration described in Figure 1 would result in a semantic cluster from which a number of syntax expressions related to entering/exiting could be derived, i.e., see Figure 4.

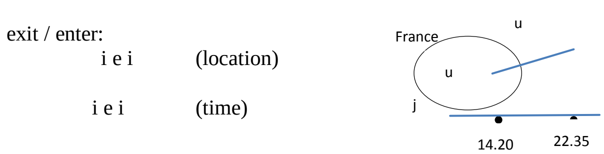


Figure 4. Category cluster describing entering-exiting

The potential for data items to be represented as components of category clusters, and therefore to generate syntax based on their semantic relations, opens the way to a number of practical applications. A few examples may illustrate this.

1. *N-tuple concepts*

In Section IV, n-tuple concepts have been described as a particular case of category clusters. They constitute a major part of NLs lexical inventory, mostly as nouns, verb roots, or, depending on the specific language, their lexical/morphological equivalents.

Two kinds of n-tuple concepts are of particular interest: (i) standalone items that can move freely, i.e., having a location that may change along time, and (ii) items that can only move with the terrain (i.e., through deformation of the surface or volume they are part of), e.g., features of a landscape. A mountain's location, for example, might change relative to other features in the landscape, but the landscape's denotational relations should not be expected to change.

Fixed location n-tuples and movable location n-tuples have intrinsically different properties, and therefore they have to be described as intrinsically different combinations of categories. If we denote them as resp. fixedQ and looseQ, then

fixedQ	$q0 \wedge (q \wedge time) \wedge loc$
looseQ	$q0 \wedge (q \wedge time) \wedge (loc \wedge time)$

where *loc* denotes an S0 spatial location, *q* denotes a combination of categories that may or may not evolve along time (e.g., color, or size), and *q0* denotes an instance that unequivocally characterizes the n-tuple it is part of. For material objects, *q0* is usually a shape, and generally does not have a specific label, e. g.

duck	$duckY \wedge (q \wedge time) \wedge loc$
hill	$hillY \wedge (q \wedge time) \wedge loc$

The labels *duckY* and *hillY* have been used to label two instances of the category *shape* that do not have a lexical correlate in English, i.e., to fill a lexical gap. In a number of languages, such instances can only be indirectly referred to, e.g., as 'the shape of a duck'.

The above description means that, in static terms, the concept 'duck' could be decomposed into a number of attributes (e.g., the categories *color*, *shape*, *size*, etc.), while each instance of a duck, i.e., each combination of those attributes, could be associated to a point in space. If those attributes are in turn combined with a space$\wedge$time configuration, i.e.,

duck	$duckY \wedge (q \wedge time) \wedge (loc \wedge time)$

then the duck will be able not only to evolve, but also move and have a history. Concepts like *hill*, construed as fixed objects, could not move independently from their location, so they could only be described in static terms. Insofar as denotational configurations can be combined, a dynamic description of a hill would always be possible, which of course would be no news for a cartoonist.

2. *Geolocation*

Geolocation data could be represented as a category cluster by using the category structures derived from its components, e.g., see Figure 5.

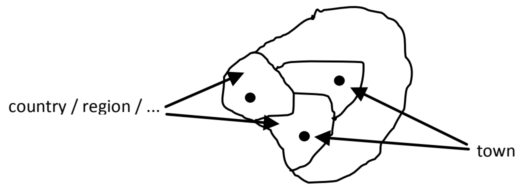


Figure 5. Geographical map categorization

For example, from the categories

Lisbon / Madrid / Rome / ...
Lisbon / Porto / Coimbra / ...
Portugal / UK / Austria / ...

a number of spatial relations could be identified. For example, if a point in a map is denoted as *x*, then the adjacency relation k_{20} could be associated to the English preposition 'in', e.g.,

in(x, Lisbon)
in(x, Portugal)
in(x, Europe)
in(Lisbon, Portugal)
in(Portugal, Europe)

The expression in(x, Lisbon) above is based on the fact that a town can be represented either as a one-dimensional point or as a two-dimensional area.

Spatial concepts potentially used by geolocation applications could be additionally defined based on quantitative-range labelling (e.g., *near*, *far*, *distance*) or on specific configurational features (e.g., *North*, *South*, *latitude*, *equator*). As has been shown, movement concepts could also be incorporated by means of the (S1) updating rules.

Similarly, time concepts could also be categorized in different ways, i.e., represented by means of different configurations, e.g., see Figures 6 and 7.

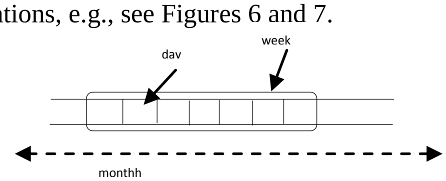


Figure 6. Time categorization (a)

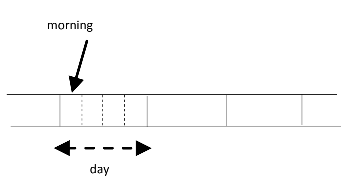


Figure 7. Time categorization (b)

Concepts such as *before*, *after*, *during*, *yesterday*, etc. would naturally emerge from such representations, and could then be used, together with spatial and/or movement representations, to derive syntax expressions with querying and/or updating purposes.

3. Meteorology

Representations describing movement could also be used to describe the evolution of meteorological variables, just by replacing spatial categories with categories derived from meteorological variables, e.g., see Figure 8.

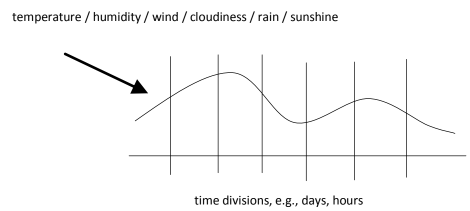


Figure 8. Evolution of meteorological variables

Expressed as category clusters, the values of the different variables would be represented as, e.g., Figure 9.

...	v_1	v_1	V	v_2	...
...	T	t_1	T	t_2	...

Figure 9. A category cluster representing the evolution of meteorological variables

where v_1, v_2 would denote individual, point-like values of the relevant variable, while *V*, *T* would generically denote resp. value ranges and time spans. Thus, in Figure 9 the value v_1 would not change during a time span *T* until the time t_1, after which it would evolve to v_2 along a range of values *V* during a time span *T*.

As has been noted, both time and value ranges could be quantified, but for the purposes of generating syntax expressions that would not necessarily be a requirement.

The above examples show that there are a number of domains where data items —whether concept attributes or measured values— can be categorized and represented as category clusters. By means of the expressions (6) and (9), such category clusters have the potential to generate as many syntax constructions as could possibly be derived from them, even if some of their elements may not have been labelled and/or conceptually identified. The filling of lexical gaps, and the development of novel theoretical frameworks, are just natural consequences of that potential.

Since the advent of smartphones, we are surrounded by data. We can wirelessly access GPS and geographical data practically anywhere anytime, as well as train/flight/bus timetables, food recipes, health tips, weather forecasts, etc. By structuring and combining those data in terms of category clusters, they could be endowed with a potential to exchange (i.e., generate and decode) information through possibly any existing NL. Furthermore, their basic components would be derived from objective and irreducible properties of space and time, and would therefore neatly fit into the definition of semantic primitives.

XIII. Databases as Category Clusters

A variety of approaches to communication through different arrangements of data have been described [3][11], including based on ontologies [49][5], the Semantic Web [34], and even neural networks [38]. However, neither ontologies nor databases or, for that matter, n-tuple concepts are explicitly conceived as category clusters [12]. They may contain symbols denoting categories, but in general they do not acknowledge the fine structure provided by semantic clusters, whether by identifying irreducible relations or by decomposing extant relations into irreducible ones. Furthermore, some of the categories identified are only implicitly based on spontaneous categorization, and their updating structure is not formally acknowledged and, consequently, often not used.

An interpretation of databases as category clusters, i.e., reflecting category structures and semantic relations, would make it possible to directly generate and decode a much larger and richer repertoire of string syntax —and hence, conventional syntax— expressions. It would therefore provide a more natural way to exchange information with conventional language users by using string syntax as an intermediary. This can be summarized as follows.

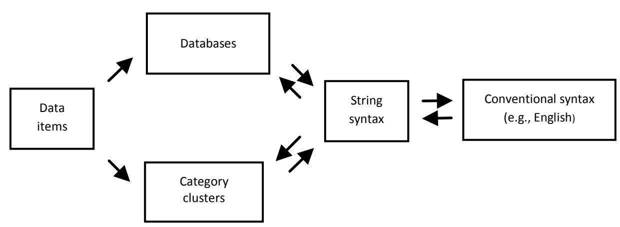


Figure 10. Two alternative data arrangements for information exchange with conventional language users

The expressions (11) and (12) provide a means to resp. update and query a communicating party through string syntax, provided that the sender's and the receiver's data arrangements are consistent with each other. Therefore, whatever the spatial configuration of a database, string syntax communication will be possible whenever the database's content can be interpreted in terms of categories and category clusters.

In a database D, an empty table T_θ consisting of the columns C_1, C_2, ... could be interpreted as a combination of the associated categories C_1, C_2, ..., and any instantiation of that combined category would describe a row (or potential row) in T_θ. For example, let the table T_k consist of the columns 'name', 'age', and 'address'. Because each of these can take *any* value within its respective scope, they can also be construed as categories, i.e.,

$$\text{name}() \wedge \text{age}() \wedge \text{address}() \;>>\; G_\theta$$

where G_θ would be a category cluster associated to T_θ. By specifying values for those columns, a number of rows would be obtained, e.g.,

$$\text{name(Oz)} \wedge \text{age(39)} \wedge \text{address(7th Av.)} >> \text{row}_1$$
$$\text{name(John)} \wedge \text{age(54)} \wedge \text{address(97 St.)} >> \text{row}_2$$
$$\text{name()} \wedge \text{age(33)} \wedge \text{address(221B St.)} >> \text{row}_3$$

Now, if we use the connective $\vee$ to link the rows above, then the table T_θ could be interpreted as a category ambiguously referring to any of its rows. If we use the connective $\wedge$ instead, then T_θ could be interpreted as a combination of rows, i.e.,

$$\text{row}_1 \vee \text{row}_2 \vee \text{row}_3 \vee \dots \;>>\; \text{employee}$$
$$\text{row}_1 \wedge \text{row}_2 \wedge \text{row}_3 \wedge \dots \;>>\; \text{employees}$$

In string syntax, both rows and cells within a row could be referred to by means of (6) and (9), e.g.,

$$\text{employee } [r_2 \text{ age(33)}]$$
$$\text{age } [r'_2 \text{ employee } [r_3 \text{ 221B St.}]]$$

where the relations r_2, r'_2, r_3 would be defined according to (4) and (5). In the general case, communication between a database and a user could be established in either direction as follows:

A. User to database

By reversing the rules used to derive conventional syntaxes, messages sent by users to a database D for updating or querying purposes could be expressed in string syntax by means of resp. (9) or (10). Such messages could be processed at D insofar as its tables could be interpreted as category clusters and those clusters would be consistent with the sender's. When that is the case, updating and querying could be interpreted in D as follows

String syntax	Database operation	
$G\ [r_m\ !u]$	$N_1(u_1) \wedge \dots \wedge N_m(u \leftarrow)$	(13a)
$!G\ [r_m\ u]$	$N_1(u_1) \wedge \dots \wedge N_m(u) \leftarrow N_m(u)$	(13b)
$?G\ [r_m\ u]$	$N_m(u) \rightarrow N_1(u_1) \wedge \dots \wedge N_m(u)$	(13c)
$?C_m\ [r'_m\ G]$	$N_1(u_1) \wedge \dots \wedge N_m(\rightarrow u)$	(13d)

where the symbols $\leftarrow$ and $\rightarrow$ denote resp. the incorporation of a new item and the identification of an extant item. The updating operation would add resp. a value or a row to D, while the query would prompt D to identify resp. an item or a table, and then send the result to the querying party.

Therefore, to be able to process string syntax expressions, a database should be configured so that either (a) the column headers in its tables reflect categories potentially referred to by the user, or (b) a sub-table could be identified in D for each category cluster that might be referred to by a user.

This is not uncommon. Meteorological and geolocation databases usually record data expected to be of interest for the general user, and databases containing spatial/temporal data most often lend themselves to semantic interpretation. As an example, let us define the category cluster G as in Figure 11:

...	h_1	h_1	H	h_2	...
...	T	t_1	T	t_2	...

Figure 11. Space-time category cluster

This could be interpreted as describing a stay at the location h_1 for an indefinite time T until the time t_1, then some movement along some distance H during an indefinite time span T, and then the presence on a fixed location h_2 at the time t_2. From that cluster, the subclusters

h_1	H
t_1	T

H	h_2
T	t_2

could be denoted resp. as G_{depart}, G_{arrive}, *implying* the relations

from(G_{depart}, loc)
at(G_{arrive}, time)

The above relations could be used to construct a number of useful string syntax expressions, e.g.,

G_{depart} [from Rome]
G_{arrive} [at 09:23]

and therefore also updating and querying expressions, e.g.,

?time [r'$_2$ G_{arrive} [to Rome]]

For a database to be able to interpret such expressions, the adjacency relations in the sub-table

origin	destination	departure	arrival
Bonn	Rome	20:15	22:30

should be reconfigured so as to reflect the semantic relations in G, e.g.,

[origin]	H	[destination]
[departure]	T	[arrival]

so that, e.g., the sub-table

H	h_2
T	t_2

could be associated to the category cluster

G_{arrive}(loc, time)

The reconfigured table in *D* is actually a three-dimensional table, where the original columns are now arranged differently, i.e., only the topology of the table has been changed.

B. Database to user

The correspondences (13a-b) could reciprocally be used by *D* to derive reports expressed in string syntax, i.e.,

Database operation	String syntax
$N_1(u_1) \wedge ... \wedge N_m(\rightarrow u)$	G [r_m !u]
$N_m(u) \rightarrow N_1(u_1) \wedge ... \wedge N_m(u)$	!G [r_m u]

that would prompt the receiver to update its representation in response to the query previously sent, or by, e.g., a geolocation algorithm intended to keep a user updated on his surroundings.

An example would hopefully illustrate the reporting process. In a meteorological database *M*, the column headers 'temp', 'humidity', 'loc', and 'time' could be associated a combined category that a user would interpret as a number of variables describing different weather states, i.e.,

Column headers	Combined category
temp humidity loc time	temp $\wedge$ humidity $\wedge$ loc $\wedge$ time

A query intended to find out, e.g., the temperature in Paris at 22:05 would be expressed in string syntax as

?temp [r_1 Paris] [r_2 22:05]

In response to that query, the database would locate the row *R* having 'Paris' under the header 'loc' and '22:05' under the header 'time'. It would then retrieve from that row the cell under the header 'temp', and express the resulting value in string syntax as

R [r_3 !33ºC] [r_1 Paris] [r_2 22:05] (14)

If we use English words for the subindices, then we can write

r_1	r_{in}
R	$R_{a\text{-}row\text{-}in\text{-}this\text{-}database}$
r_2	r_{at}

A few translation rules, together with (7), would convert (14) into the conventional syntax expression

the temperature from a row in this database in Paris at 22:05 is 33ºC

However, the receivers need not even know that the data has been retrieved from some table in the source database. They have chosen to ask the source because they trust it to output reliable data. Therefore, the source might safely decide to just translate

the temperature in Paris at 22:05 is 33ºC

This omission might seem like a trick shrewdly devised to get the desired result. On the contrary, it is an information compression device routinely used by natural language speakers. Consider just a few examples.

- the kitchen [of our house] is in the ground floor
- I can see the airport [of Beijing] now
- the book [you expressed an interest to buy three minutes ago] is *Finnegans Wake*

XIV. Potential Tools for Deafblind Users

One sector of the population that could potentially benefit from this framework are deafblind users. DeafBlind People (DBP) are affected by different degrees of visual/auditive impairment. They communicate through a surprisingly wide —and quite imaginative— diversity of languages and communication media [42][46], and have a severely limited access to the perceptual world [33]. Their access to information is generally limited to communication with other human beings through various, often non-standard, languages. Therefore, they generally require a human intermediary to communicate with the external world, as well as to find their way around. Attempts to facilitate DBP communication have been described, including based on gesture recognition [2][29], tactile messages [8], human-robot interaction [39], and many others. However, deafblind people's familiarity with abstract concepts is limited, which poses a formidable challenge [9].

In that respect, the potential benefits of the ideas presented in this paper are threefold, i.e., (a) string syntax could be used as a bridge between databases and DBP languages, which could give DBP users access to a wealth of external information otherwise inaccessible to them; (b) because semantic clusters can be used to generate and decode syntax expressions, the recognition of such structures, and the knowledge of the rules to deal with them, could facilitate DBP language learning and comprehension, and enhance their conversational skills; and (c) the possibility to spatially represent semantic relations could provide DBP with an invaluable tool to enhance their cognitive universe in a more systematic way than what has been achieved to date [13][48]. The first two benefits could be attained resp. through parsing and training. The third one will be discussed below.

The structure of perceptions plays a key role in the formation of concepts. As an example, people who have experienced long-term visual deprivation and are given access to retinal perceptions initially fail to see those perceptions in a structured way [20]. Nonetheless, insofar as perceptions can be represented in terms of spatial adjacency relations, they might be accessible anyway. No blind person can perceive colors, but a structural description of a rainbow or, for that matter, of the color spectrum could be apprehended by them and labelled in a way consistent with non-visually impaired people's. Such possibilities open the door to a number of tools aimed at a cognitive enhancement of DBP. While DBP lack the input to construct some sensory representations, such representations could possibly be taught to them, thereby helping them not only to find their way around without human assistance, but also to enhance their knowledge about the world they live in.

Actually, language learning and cognitive enhancement would go hand in hand. Both would rely on three components dealing with the aspects of resp. cognition, output, and input, namely: (a) recognizing the structure of basic categories, as well as a number of elementary category clusters; (b) learning to associate lexical tokens to semantic representations or parts thereof; (c) recognizing the basic components of string syntax, i.e., categories, instances, and relations, as well as the spontaneous categorization mechanism.

The realization that a number of categories can be updated and assembled, and of how it can be done, should in turn lead to the realization that semantic gaps can be filled, and that more complex concepts can be envisioned, whether they have a material correlate or are a personal creation. This should make DBP aware that the both the material and mental worlds are ever larger and, with the right tools at hand, it can be explored.

A roadmap along those lines could only be sketched within the scope of this paper. In the examples below, the teaching methods suggested appear in italics, while the teacher's prompts are shown between asterisks, and lexical tokens are enclosed in angle brackets. The symbol :: denotes the expected association between the prompts and the corresponding lexical tokens.

Category **animal** (e.g., toys)
Tactile recognition
<animal><x>
where <x> :: *duck*/*bird*/*turtle*/...

Category **shape**
Tactile recognition
<shape><x>
where <x> :: *duck*/*square*/*circle*/...

Category **size**
Tactile recognition
<large><x>/<small><x>
where <x> :: *duck*/*square*/*circle*/...

Category **up**/**down**
Hand position
<x><up>/<x><down>
where <x> :: *duck*/*box*/*hand*/...

Spatial updating
Tactile recognition
<y><next to><x>
where <x>,<y> :: *me*,*table*/*table,*you*/...

Time updating
Tactile exploration (e.g., on a clock face)
<y><then><x>
where <x>,<y> :: *<sleep>*,*<wake up>*/...

Category **still**/**moving**

Tactile recognition
<x><still>/<x><moving>
where <x> :: *<rabbit>*,*<car>*/...

Having identified space and time concepts and their respective updating structure, a possible course of action could be for the DBP learner to be presented the semantic cluster in Figure 11. From that cluster, a number of subclusters representing concepts, e.g., *move*, *stay*, *before*, *from*, *at*, etc. could be identified, and their lexical correlates could be used to assemble a number of syntax string expressions, as described in Section XIII.

Regarding the possible implementations, a portable device with a haptic interface, which could physically change hands to send and receive messages by other human parties, would arguably provide a higher degree of autonomy than garments or other wearable devices. Furthermore, the interface provided by a portable device would facilitate a more sophisticated interaction, particularly for the purposes of language exchange, and language and concept learning. It could also be a means, or at least provide a stimulus, for the users to replace their sign/haptic language or dialect with string syntax as a user-friendly, universal language. Its three basic elements, i.e., categories, instances and relations, could be readily expressed by means of haptic icons, and its syntax rules are simplest and intuitive.

Human language is a vast field, encompassing all kinds of conceivable concepts. Therefore, any communication project could only be realistic if constrained to a specific, clearly delimited concept domain. In view of this, a possible plan aimed to learning and communication should initially consist of a number of basic semantic fields plus a roadmap to expand such fields to other semantic domains. Actually, a significant problem might be the identification of clear boundaries, considering the all-pervading nature of semantic notions.

A number of tools to be initially developed to implement such an interface would be as follows:

COST: A parser to translate [simple] conventional syntax (CO) expressions into string syntax (ST) expressions

STCO: A translator from string syntax (ST) to conventional syntaxes (CO)

LESE: A dictionary associating lexical tokens (LE) to semantic configurations (SE)

META: A dictionary of metalanguage signs to instruct the recognition of semantic tokens and their association to string syntax components

DASE: A module to rearrange database data (DA) into identifiable semantic configurations (SE)

LEX: A number of limited lexicons from different domains. Initially, they might include meteorology (MET), and geolocation (GEO). As a potential further addition, static and dynamic in-out concepts (IOC) could also be explored

The author has already designed an interface along those lines. However, a detailed description of that interface would be beyond the scope of this paper.

CONCLUSION

A representational approach to data, together with the categorial structure implicit in natural languages, can be seen as the components of a formal framework that integrates syntax and semantics under a single theoretical construct. Within that novel syntax/semantics integrated framework, human communication could be achieved by structurally representing either internal thoughts or external perceptions/data as category clusters. In such a representation, the category-instance relation could be used to locally refer to individual components through a disambiguation process, which could be expressed in a particular string form.

The syntax informing such strings has been shown to be consistent with conventional languages, as well as databases and various representational implementations. Furthermore, the identification of formal structures underlying categories lends category clusters an objective semantic quality, and endows them with the potential to generate syntax expressions.

This unification of syntax and semantics into one single model could be the basis for an interface to be designed, among other purposes, to: (a) operate as a universal translator; (b) derive language expressions from spatial representations, and conversely, extract representations from syntax strings; (c) rearrange databases in a representational format; and (d) give sensory impaired people a more extensive access to the external world.

Future work along those lines would include the implementation of a number of interactive database-user interfaces, e.g., for geolocation purposes, flight/train data querying at resp. airports/train stations, etc., with the aim of progressively enlarging their scope and incorporating ever more complex data sources. The optimization of such interfaces would also be interactive, and essentially not different from the dispelling of misunderstandings in colloquial language.

The development of an interactive methodology along the lines sketched in Section XIV would also be a promising tool to enhance the cognitive universe of deafblind people.

ACKNOWLEDGEMENTS

Some notions presented in this paper, and specifically the formalization of the concept of denotational jigsaw, might not have materialized without the critical comments of Ernesto Sánchez de Cos, who forced me to delve deeper into ideas that initially were rather vaguely expressed. The patience and kindness of Sándor Darányi, always willing to pay attention to my ramblings, rain or shine, were also invaluable to me, as well as his critical comments and his emotional support during the long and painstaking mental process that led to the completion of this paper. My sincere thanks to both.

REFERENCES

[1] C. Seror, “A Data Referencing Formalism for Information Exchange between Deafblind People and Databases,” SEMAPRO 2019, The Thirteenth International Conference on Advances in Semantic Processing, 2019.

[2] G. Airò Farulla et al., "Real-Time Single Camera Hand Gesture Recognition System for Remote Deaf-Blind Communication," International Conference on Augmented and Virtual Reality AVR 2014: Augmented and Virtual Reality, pp. 35-52, 2014.

[3] I. Androutsopoulos, G. D. Ritchie, and P. Thanisch, "Natural language interfaces to databases - An introduction," Natural Language Engineering, 1(1), pp. 29–81, 1995.

[4] L. W. Barsalou, "Perceptual symbol systems," Behavioral and Brain Sciences 22, 577–660, Cambridge University Press, 1999.

[5] J. A. Bateman et al. "A linguistic ontology of space for natural language processing," Artificial Intelligence, 174.14: 1027-1071, 2010.

[6] R. J. Bobrow and R. M. Weischedel, "Challenges in Natural Language Processing," Cambridge University Press, 1993.

[7] M. H. Bornstein and C. Mash, "Experience-Based and On-Line Categorization of Objects in Early Infancy," Child Development, Vol. 81, Issue 3, pp. 884-897, 2010.

[8] S. Brewster, and L. Brown, "Tactons: Structured tactile messages for non-visual information display," in 5th Conf. on Australasian User Interface, ACM, pp. 15-23, 2004.

[9] S. M. Bruce, "The Impact of Congenital Deafblindness on the Struggle to Symbolism," International Journal of Disability, Development and Education, 52:3, 233-251, 2005.

[10] A. Cahen and M. C. Tacca, "Linking perception and cognition," Frontiers in Psychology, 2013.

[11] N. Caporusso et al., "Enabling touch-based communication in wearable devices for people with sensory and multisensory impairments," in: 1st International Conference on Human Factors and Wearable Technologies, pp. 149-159, 2017.

[12] E. F. Codd, "A Relational Model of Data for Large Shared Data Banks," IBM Research Laboratory, San Jose, California, 1970.

[13] W. S. Curtis, E. T. Donlon and D. Tweedie, "Learning behavior of deaf-blind children. Education of the Visually Handicapped," 7(2), American Psychological Association, pp. 40-48, 1975.

[14] M. E. D'Imperio, "Data structures and their representation in storage," Annual Review in Automatic Programming, Volume 5, Part 1, 1969, pp. 1-75, ISSN 0066-4138.

[15] J. Davidoff, "Language and perceptual categorisation," Trends in Cognitive Sciences, Volume 5, Issue 9, 1 September 2001, pp. 382-387.

[16] A. S. Eddington, "Space, Time and Gravitation. An outline of the general relativity theory," Cambridge University Press, 1920.

[17] N. Evans, "Dying Words: Endangered Languages and What They Have to Tell Us," Wiley-Blackwell, 2009.

[18] D. L. Everett, "Cultural Constraints on Grammar and Cognition in Piraha," Current Anthropology, Volume 46, Number 4, 2005.

[19] G. Fauconnier, E. Sweetser (eds.), "Spaces, Worlds, and Grammar," The University of Chicago Press, pp. 1-28, 1996.

[20] I. Fine et al., "Long-term deprivation affects visual perception and cortex," Nature Neuroscience, Volume 6, No. 9, 2003.

[21] G. Frege, "Über Sinn und Bedeutung," in: Zeitschrift für Philosophie und philosophische Kritik. Vol. 100, pp. 25–50, 1892.

[22] C. Freksa, "Spatial and temporal structures in cognitive processes," Lecture Notes in Computer Science, Vol. 1337. Springer, 1997.

[23] A. Gatt and E. Krahmer, "Survey of the State of the Art in Natural Language Generation: Core tasks, applications and evaluation," Journal of Artificial Intelligence Research 61, 65-170, 2018.

[24] G. W. Gifford III, K. A. MacLean, M. D. Hauser and Y. E. Cohen, "The Neurophysiology of Functionally Meaningful Categories: Macaque Ventrolateral Prefrontal Cortex Plays a Critical Role in Spontaneous Categorization of Species-Specific Vocalizations ," Journal of Cognitive Neuroscience, Volume 17, Issue 9, pp. 1471-1482, 2005.

[25] N. Giret, F. Péron, L. Nagle, M. Kreutzer, D. Bovet, "Spontaneous categorization of vocal imitations in African grey parrots (Psittacus erithacus)," Behavioural Processes, Volume 82, Issue 3, pp. 244-248, 2009.

[26] A. Gopnik and A. Meltzoff, "The Development of Categorization in the Second Year and Its Relation to Other Cognitive and Linguistic Developments," Child Development, Vol. 58, No. 6, pp. 1523-1531, 1987.

[27] U. Hasson, E. Yang, I. Vallines, D. J. Heeger and N. Rubin, "A Hierarchy of Temporal Receptive Windows in Human Cortex," Journal of Neuroscience, 28 (10) 2539-2550, 2008.

[28] E. Horowitz and S. Sahni, "Fundamentals of Data Structures," Computer Science Press, 1983.

[29] J. Kramer and L. Leifer, "The talking glove," SIGCAPH Comput. Phys. Handicap 39, pp. 12–16, 1988.

[30] R. K. Logan, "What Is Information?: Why Is It Relativistic and What Is Its Relationship to Materiality, Meaning and Organization," Information 2012, 3(1), 68-91, 2012.

[31] R. M. Losee, "A Discipline Independent Definition of Information," Journal of the American Society for Information Science, 48 (3) 1997, pp. 254-269, 1997.

[32] D. Mareschal and P. C. Quinn, "Categorization in infancy," TRENDS in Cognitive Sciences Vol.5 No. 10, pp. 443-450, 2001.

[33] B. Miles, "Overview on Deaf-Blindness," Helen Keller National Center, Revised October 2008.

[34] B. Munat, "The lowercase semantic web: using semantics on the existing world wide web," Evergreen State College, Olympia, 2004.

[35] D. G. Nash, "Topics in Warlpiri Grammar," Massachusetts Institute of Technology, 1980.

[36] S. E. Petersen and O. Sporns, "Brain Networks and Cognitive Architectures," Neuron, 88(1): 207–219, 2015.

[37] M. Petruck, "Frame semantics," Handbook of pragmatics, pp. 1-13, John Benjamins Publishing Company, 1996.

[38] P. Z. Revesz and R-R. K. Veera, "A Sign-To-Speech Translation System Using Matcher Neural Networks," https://cse.unl.edu/~revesz/papers/ANNIE93.pdf 2020.05.26.

[39] L. O. Russo et al., "PARLOMA – A Novel Human-Robot Interaction System for Deaf-Blind Remote Communication," https://journals.sagepub.com/doi/10.5772/60416 2020.05.26.

[40] C. Seror, "Human language and the information process," https://zenodo.org/record/3263753#.XRmuceszaUk 2020.05.26.

[41] C. E. Shannon, "A Mathematical Theory of Communication," Bell System Technical Journal, Vol. 27, pp. 379–423, 1948.

[42] W. C. Stokoe, Jr., "Sign Language Structure: An Outline of the Visual Communication Systems of the American Deaf," The Journal of Deaf Studies and Deaf Education, Vol. 10, Issue 1, pp. 3–37, 2005.

[43] A. M. Tozzer, "A Maya Grammar, with bibliography and appraisement of the works noted," Papers of the Peabody Museum of American Archaeology and Ethnology, Vol. IX, p. 55, 1921.

[44] R. Vigo, "Representational information: a new general notion and measure of information," Information Sciences 181, 4847–4859, 2011.

[45] R. Vigo, "The GIST of concepts," Cognition, Vol. 129, Issue 1, Elsevier, pp. 138-162, 2013.

[46] YouTube, "Sistemas de Comunicación para Personas Sordociegas," https://www.youtube.com/watch?v=SpcsbgM1TIc 2020.05.26

[47] R. Wille, "Formal Concept Analysis as Mathematical Theory of Concepts and Concept Hierarchies," Formal Concept Analysis: Foundations and Applications. Ganter, B., Stumme, G., Wille, R. Eds. Springer, pp. 1-33, 2005.

[48] K. Wolff Heller, P.A. Alberto, and J. Bowdin, "Interactions of communication partners and students who are deaf-blind: A model," Journal of Visual Impairment & Blindness, 89(4), pp. 391-401, 1995.

[49] M. Zerkouk, A. Mhamed, and B. Messabih, "A user profile based access control model and architecture," http://www.airccse.org/journal/cnc/0113cnc12.pdf 2020.05.26.

[50] F. Mesquita Carvalho and N. Kriegeskorte, "The Arrow of Time: How Does the Brain Represent Natural Temporal Structure?," Journal of Neuroscience, 28 (32) 7933-7935, 2008.