David Hume's 'Causation' in Database Semantics

Roland Hausser
Universität Erlangen-Nürnberg (em.)

Abstract

In physics, causation is an agent-external phenomenon, for example gravity causing the movement of the planets (Newton 1687). According to Hume's (1739) philosophical reception of Newton's law of gravitation[1], the cause must precede the effect (called *temporal priority*), and cause and effect must be spatiotemporally conjoined (called *contiguity*).

In agent-internal cognition, there is an analogous phenomenon. For example, (x) Mary turned off the light and fell asleep satisfies temporal priority and contiguity, while (y) Mary fell asleep and turned off the light violates temporal priority. Content tokens matching the content types (x) and (y) are called *accommodating contents*, and content tokens satisfying content type (x) but not (y) are called *functionally accommodating* in DBS.

Functional accommodation is a generalization of Hume's cause and effect. For example, John put on his socks and shoes is an instance of functional accommodation though not of causation, while John put on his shoes and socks is a case of neither. This paper presents the technical details of reconstructing functional accommodation in DBS.

1 Asymmetry in Natural Coordination

In propositional calculus, conjunction is symmetric, i.e. $p \wedge q = q \wedge p$, and implication is asymmetric, i.e. $p \rightarrow q \neq q \rightarrow p$. In natural cognition, however, coordination may be asymmetric, either because of (a) different functional accommodations or (b) their presence vs. absence. The following contents of clausal coordination represent alternative orders by means of English language surfaces:

1.1 Implications in Natural Coordination Contents

1. (x) John opened the window and threw out the cat.[2]
   (y) John threw out the cat and opened the window.
2. (x) John got ill and Mary made soup.
   (y) Mary made soup and John got ill.
3. (x) Mary turned off the light and fell asleep.
   (y) Mary fell asleep and turned off the light.
4. (x) John put on the water and boiled the potatoes.
   (y) John boiled the potatoes and put on the water.
5. (x) Suzy opened the fridge and got a beer.
   (y) Suzy got a beer and opened the fridge.

[1] Slavov 2013.
[2] Prof. G. Lakoff, lecture at the Linguistic Summer School UC Santa Cruz, 1971 or 1972.

6. (x) Suzy got a beer and closed the fridge.
   (y) Suzy closed the fridge and got a beer.

The alternative conjunction orders in examples (1) and (2) support different functional accommodations. More specifically, order (x) in (1) implies that John opened the window for the purpose of throwing the cat through it while order (y) may be instantiated by a more conventional exit of the cat having caused bad air. The order (x) in (2) may be motivated as Mary's intention to comfort John, while on order (y) John's illness may have been caused by Mary's soup. The respective orders in the remaining examples (3–6), in contrast, support functional accommodation in the variants (x), while there is no functional accommodation for the variants (y).

2 Cause and Effect

Except for 2.(y), the examples of functional accommodation in 1.1 are not instances of causation, yet they satisfy Hume's (1739) definition in terms of contiguity and temporal priority:

2.1 HUME'S DEFINITION OF CAUSATION:

> X causes Y if and only if the two events are spatiotemporally conjoined (contiguity), and X precedes Y (temporal priority),

For example, the contents (a) Suzy opened the fridge and (b) got a beer (1.1, 5.(x)) do not express a causal relation, but are spatiotemporally conjoined by the assumption (i) that the beer is located in the fridge (which makes sense in cultures with refrigerators and bottled beer) and (ii) the time intervals are adjacent. Temporal priority is fulfilled if the purpose of opening the fridge is getting the beer.

In summary, the examples in 1.1 show that contiguity and temporal priority may be fulfilled not only by the agents' cognition-external reality, but also by accommodating contents which consist of two clauses conjoined in a certain order. The examples show also that accommodating contents are not limited to causation, but include all kinds of regular interactions, such as purpose and natural order, for example, putting on the socks before the shoes, slowing down before getting off the bike, digging the foundation before putting on the roof, etc.

3 Necessary, Unnecessary, Sufficient, and Insufficient Causes

Hume requires *constant* contiguity for cause and effect, i.e. Y must *always* follow from X, while for functional accommodation in DBS sporadic consequents are sufficient. For Hume's causation, constant contiguity is widely accepted as a necessary condition, but whether it is also sufficient is controversial.

For a more differentiated account of complex causes, J.L. Mackie (1965) distinguishes (i) necessary, (ii) unnecessary, (iii) sufficient, and (iv) insufficient causes, called the INUS condition by Mackie. For example, a short circuit causing a house on fire is a US constellation: the short circuit is Unecessary (what Aristotle calls *accidental*[3]) because there are other possible causes, such as arson or lightning. The short circuit is Sufficient because it effectively caused the house to burn.

4 Hume's Copy Principle

Underlying Hume's definition 2.1 is his *copy principle*[4], according to which the efficacy of elementary ideas comes from impressions copied into the mind, while the efficacy of the combination of elementary ideas into complex ideas is provided by the mind alone. For example, the elementary impressions *golden* and *mountain* and their corresponding ideas have counterparts in the real world, while the complex idea *golden mountain* does not, and similarly for *Pegasus* and *unicorn*.

In terms of agent-based data-driven DBS, this would mean that the recognition and action of elementary concepts is provided by the agent's interface component, but their combination into complex content is entirely cognition-internal. In fact, however, no such distinction is made in DBS.

As shown by such phenomena as visual illusion and mishearing (recognition), as well as mishandling and losing one's way (action), the type-token matching between elementary contents (concepts) and raw data (e.g., sound or light waves) is no less cognition-based than their combination by the semantic relations of structure, for example in inferencing. In other words, DBS agrees with Hume in that the combination of elementary ideas into complex ideas is provided by the mind, i.e. the agent's cognition. It it is just that elementary recognition and action in DBS are provided by the mind as well. Also, the functional accommodation of DBS is not limited to causality, but is generalized to a multitude of other systematic relations.

5 Computational Reconstruction of Elementary Recognition and Action

Hume described the 'mind' in terms of 'impressions' and 'ideas.' Impressions are divided into 'sensations' and 'reflections.' In the computational cognition of DBS, in contrast, the most basic distinction is between *recognition* and *action*. These notions are absent in Hume's ontology (Johansson 2012).

The systematic reconstruction of recognition and action in DBS is based on the distinction between 'types' and 'tokens' (Peirce 1906, CP Vol.4, p. 375), which goes back to Aristotle's distinction between the necessary and the accidental.

5.1 TYPE-TOKEN MATCHING FOR RECOGNITION AND ACTION IN DBS

- recognition: a type matching raw data results in a token.
- action: adapting a type into a token for a purpose results in raw data.

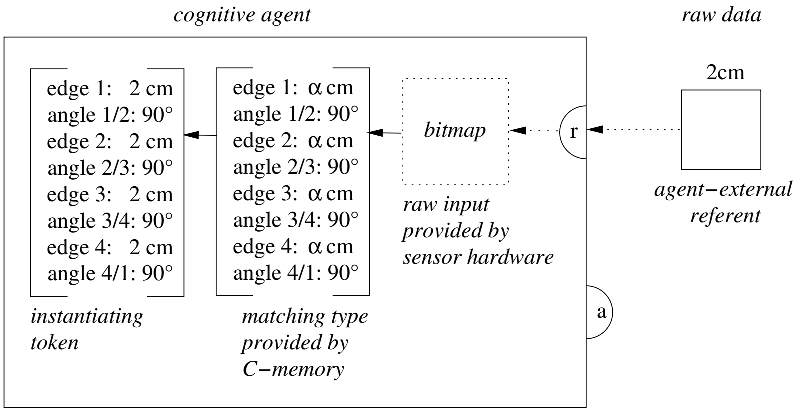
[3] Whether the opposite of Aristotle's accidental is *essential* or *necessary* is hotly debated in philosophy (Matthews 1990). We follow Quine 1966 by equating essential and necessary.

[4] Unlike his contemporaries who still published in Latin, Hume published in English.

In other words, while for Hume the operations of the mind are founded on simple impressions which are received passively,[5] the recognition and action of basic concepts by the computational cognition of agent-based data-driven DBS is *proactive*.

As an example consider a DBS agent's recognizing and producing a square:

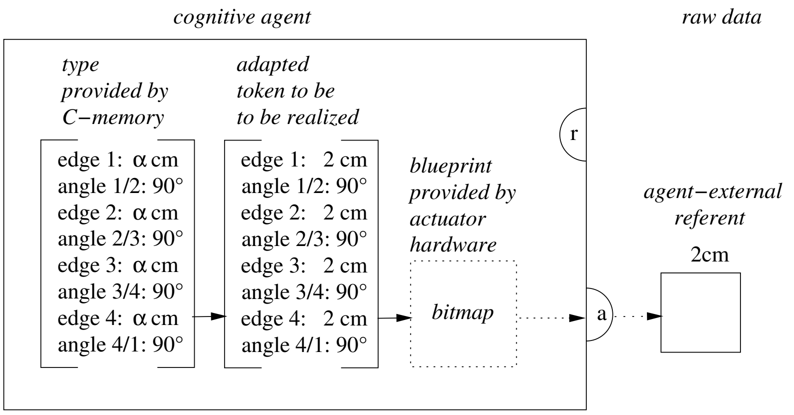
5.2 RECOGNITION OF A square



The edge length of the type is a variable which matches an infinite number of tokens with different edge lengths. The raw data are supplied by a sensor, here for vision, as input to the agent's interface component.

In action, a type is adapted to a token for the purpose at hand and realized by the agent's actuators as raw data:

5.3 ACTION OF REALIZING square

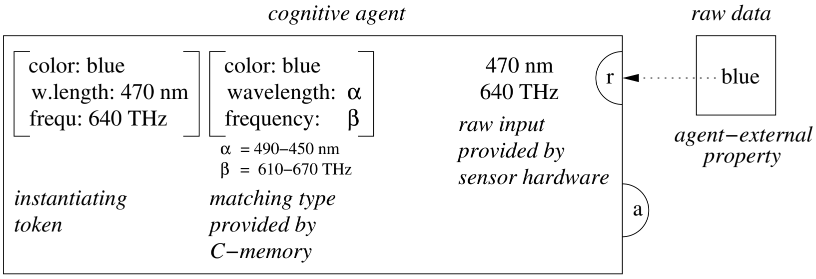


The token is used as a blueprint for action (e.g., drawing a square). The recognition and production of square may be extended to all two-dimensional geometric shapes (Hausser 2021b, 10.3.5)

Next consider the recognition of a color, here blue:

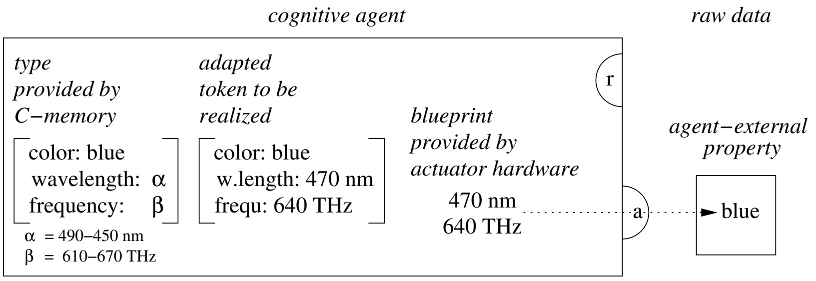
[5]Though with "most force and violence" (Hume 1739).

5.4 RECOGNITION OF blue



An example of the corresponding action is turning on the color blue, as in a cuttlefish (metasepia pfefferi) using its chromatophores:

5.5 ACTION OF REALIZING blue



The concept type matches different shades of blue, whereby the variables α and β are instantiated as constants in the resulting token. Recognition and production of blue is a general mechanism which may be applied to all colors (Hausser 2021b, 10.3.1). It may be expanded to infrared and ultraviolet, and to varying intensity.

6 Computational Reconstruction of Complex Content

In DBS, concepts are embedded as core values into nonrecursive feature structures with ordered attributes, called *proplets*:

6.1 LEXICAL PROPLETS OF blue AND square

$$\begin{bmatrix} \text{sur:} \\ \text{adj: } \textbf{blue} \\ \text{cat: adn} \\ \text{sem: pad} \\ \text{mdd:} \\ \text{mdr:} \\ \text{nc:} \\ \text{pc:} \\ \text{prn: K} \end{bmatrix} \qquad \begin{bmatrix} \text{sur:} \\ \text{noun: } \textbf{square} \\ \text{cat: snp} \\ \text{sem: indef sg} \\ \text{fnc:} \\ \text{mdr:} \\ \text{nc:} \\ \text{pc:} \\ \text{prn: K} \end{bmatrix}$$

Proplets are the computational data structure of DBS. Their second attribute is the core attribute, here adj and noun, and contains the core value. Their fifth attribute is the continuation attribute, here mdd for ‘modified’ and fnc for ’functor’, intended for the continuation value. The cat and sem slots provide the syntactic and the semantic properties of the concept.

Proplets are connected into complex content by cross-copying between their core and their continuation attributes, shown in **bold face** in the following example:

6.2 ‘LUCY FOUND A BIG BLUE SQUARE.’ AS NONLANGUAGE CONTENT

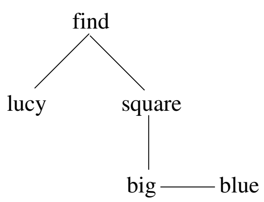
sur: lucy	sur:	sur:	sur:	sur:
noun: **[person x]**	verb: **find**	adj: **big**	adj: **blue**	noun: **square**
cat: snp	cat: #n′ #a′ decl	cat: adn	cat: adn	cat: snp
sem: nm f	sem: ind past	sem: pad	sem: pad	sem: indef sg
fnc: **find**	arg: **[person x] square**	mdd: **square**	mdd:	fnc: **find**
mdr:	mdr:	mdr:	mdr:	mdr: **big**
nc:	nc:	nc: **blue**	nc:	nc:
pc:	pc:	pc:	pc: **big**	pc:
prn: 23	prn: 23	prn: 23	prn: 23	prn: 23

The example is a nonlanguage content because the sur slots are either empty or a name marker. The explicit definitions of the values blue and square is shown in section 5, and similar definitions are assumed for the other core values. The semantic relations are classical subject/predicate, object\predicate, modifier|modified, and conjunct−conjunct. They are established by cross-copying (connective $\times$) and absorbing (connective $\cup$) values (TExer).

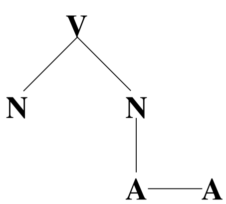
The semantic relations in 6.2, i.e. person(x)/find, square\find, big|square, and big−blue, may be shown graphically as follows:

6.3 SEMANTIC RELATIONS UNDERLYING SPEAK MODE DERIVATION

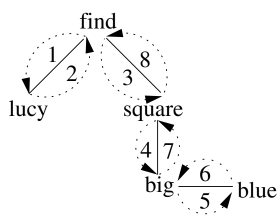
(i) SRG (semantic relations graph)



(ii) signature



(iii) NAG (numbered arcs graph)



(iv) surface realization

1	2	3	4	5	6	7	8
Lucy	found	a	big	blue		square	.
V/N	N/V	V\N	N|A	A−A	A−A	A|N	N|V

Recognition takes a linear sequence of connected proplets, e.g., 6.2, as input and produces an equivalent semantic hierarchy, e.g., 6.3, as output by interpreting the semantic relations encoded by cross-copying. Action takes a semantic hierarchy, e.g., 6.3, as input and produces a linear sequence, e.g., 6.2, as output by navigating along the semantic relations as shown in the NAG. The proplets of a content are connected by the semantic relations of structure and their prn value, but order-free for storage in and retrieval from the agent's content-addressable on-board database.

7 From Individual Contents to a Content Class

The equivalent syntactic-semantic structure of 6.2 and 6.3 is the same for an unlimited number of contents which differ solely in their core and continuation values, as shown by the following example:

7.1 Content in the same class as 6.2 and 6.3

Peter ate a sweet little chocolate.

The class containing Lucy found a big blue square and Peter ate a sweet little chocolate may be characterized abstractly by the same schema. It may be derived from 6.1 by systematically replacing the core and continuation values with variables represented by Greek letters:

7.2 Content Schema as a Set of Proplet Patterns

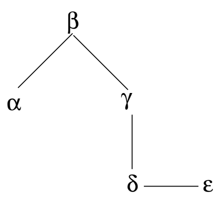
sur:	sur:	sur:	sur:	sur:
noun: α	verb: β	adj: γ	adj: δ	noun: ε
cat: snp	cat: #n′ #a′ decl	cat: adn	cat: adn	cat: snp
sem: nm f	sem: ind past	sem: pad	sem: pad	sem: indef sg
fnc: β	arg: α ε	mdd: ε	mdd:	fnc: β
mdr:	mdr:	mdr:	mdr:	mdr: γ
nc:	nc:	nc: δ	nc:	nc:
pc:	pc:	pc:	pc: γ	pc:
prn: K	prn: K	prn: K	prn: K	prn: K

This syntactic-semantic schema of a content results from 6.2 by simultaneous substitution replacing the values in bold face with variables represented by Greek letters.

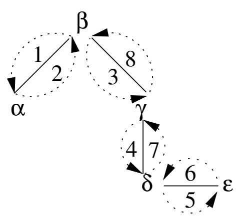
The analogous method is also applied to generalize the graphical format of 6.3 from an individual instance to the class:[6]

7.3 Content Schema as Semantic Relations Graphs

(i) SRG (semantic relations graph)



(iii) NAG (numbered arcs graph)



[6]Formats *(ii)* and *(iv)* omitted.

These graphical representations of semantic relations characterize the abstract class which matches the contents of Lucy found a big blue square. Peter ate a sweet little chocolate., and an open number of similar constructions.

8 Four Different Kinds of Content[7]

DBS applies the type-token distinction not only to concepts (section 5) but also to content. In combination with the nonlanguage-language distinction there are four kinds of content in DBS, called [−surface −STAR], [−surface +STAR], [+surface −STAR], and [+surface +STAR], illustrated as follows:

8.1 NONLANGUAGE CONTENT TYPE: [−surface, −STAR]

$$\begin{bmatrix}\text{sur:}\\ \text{noun: dog}\\ \text{cat: snp}\\ \text{sem: def sg}\\ \text{fnc: find}\\ \text{mdr:}\\ \text{nc:}\\ \text{pc:}\\ \text{prn: K}\end{bmatrix}\begin{bmatrix}\text{sur:}\\ \text{verb: find}\\ \text{cat: \#n}'\text{ \#a}'\text{ decl}\\ \text{sem: past ind}\\ \text{arg: dog bone}\\ \text{mdr:}\\ \text{nc:}\\ \text{pc:}\\ \text{prn: K}\end{bmatrix}\begin{bmatrix}\text{sur:}\\ \text{noun: bone}\\ \text{cat: snp}\\ \text{sem: indef sg}\\ \text{fnc: find}\\ \text{mdr:}\\ \text{nc:}\\ \text{pc:}\\ \text{prn: K}\end{bmatrix}$$

This proposition is a type because there is no STAR and the prn value is a variable, here K. It is a nonlanguage content because the sur slots are empty.

The next example is a corresponding nonlanguage token:

8.2 NONLANGUAGE CONTENT TOKEN: [−surface, +STAR]

$$\begin{bmatrix}\text{sur:}\\ \text{noun: dog}\\ \text{cat: snp}\\ \text{sem: def sg}\\ \text{fnc: find}\\ \text{mdr:}\\ \text{nc:}\\ \text{pc:}\\ \text{prn: 12}\end{bmatrix}\begin{bmatrix}\text{sur:}\\ \text{verb: find}\\ \text{cat: \#n}'\text{ \#a}'\text{ decl}\\ \text{sem: past ind}\\ \text{arg: dog bone}\\ \text{mdr:}\\ \text{nc:}\\ \text{pc:}\\ \text{prn: 12}\end{bmatrix}\begin{bmatrix}\text{sur:}\\ \text{noun: bone}\\ \text{cat: snp}\\ \text{sem: indef sg}\\ \text{fnc: find}\\ \text{mdr:}\\ \text{nc:}\\ \text{pc:}\\ \text{prn: 12}\end{bmatrix}\begin{bmatrix}\text{S: yard}\\ \text{T: friday}\\ \text{A: sylvester}\\ \text{R: tweety}\\ \text{3rd:}\\ \text{prn: 12}\end{bmatrix}$$

The three content proplets and the STAR proplet are connected by a common prn constant, here 12. According to the STAR, the token resulted as an observation by the agent Sylvester on Friday in the yard.

The language content type corresponding to 8.1 illustrates the independence of language-dependent sur values, here German, from the relatively language-independent placeholders (English base forms for convenience):

[7] Hausser 2021a.

8.3 Language content type: [+surface, −STAR]

sur: der_Hund	sur: fand	sur: einen_Knochen
noun: dog	verb: find	noun: bone
cat: snp	cat: #n′ #a′ decl	cat: snp
sem: def sg	sem: past ind	sem: indef sg
fnc: find	arg: dog bone	fnc: find
mdr:	mdr:	mdr:
nc:	nc:	nc:
pc:	pc:	pc:
prn: K	prn: K	prn: K

A language content type is also called a literal meaning$_1$. It is an abstraction in that an actual DBS hear mode derivation results in a content token. However, a content type may always be obtained from a content token by removing the STAR and replacing the prn constants with suitable variables.

The fourth kind of content is a language token which matches the type, here 8.3, called an utterance meaning$_2$. The example is produced by the speaker Sylvester in German towards the intended hearer Tweety and corresponds to the nonlanguage content token 8.2 except for the R value:

8.4 LANGUAGE CONTENT TOKEN: [+surface, +STAR]

sur: der_Hund	sur: fand	sur: einen_Knochen	S: yard
noun: dog	verb: find	noun: bone	T: friday
cat: snp	cat: #n′ #a′ decl	cat: snp	A: sylvester
sem: def sg	sem: past ind	sem: indef sg	R: tweety
fnc: find	arg: dog bone	fnc: find	3rd:
mdr:	mdr:	mdr:	prn: 12
nc:	nc:	nc:	
pc:	pc:	pc:	
prn: 12	prn: 12	prn: 12	

According to the STAR, the transfer of content occurred in the yard on friday from Sylvester to Tweety. The content types 8.1 and 8.3 match not only the tokens 8.2 and 8.4, but an open number of corresponding tokens with different prn values.

An utterance meaning$_2$ exists in the cognition of the speaker, and – if transfer is successful – of the hearer. The raw data serving as the vehicle of transfer in communication, in contrast, have absolutely no meaning or grammatical properties whatsoever at all (no reification in DBS), but may be measured by natural science.

9 Accommodating Scenarios in DBS

The DBS notion of a *complex content* as a set (order-free) of proplets connected by the semantic relations of structure is essential for the computational implementation of accommodating scenarios in general and functional accommodation in particular. As an example consider 5(x) in 1.1 as a DBS content:

9.1 CONTENT TOKEN OF Suzy opened the fridge and got a beer.

$$
\begin{bmatrix}\text{sur: suzy}\\ \text{noun: [person x]}\\ \text{cat: snp}\\ \text{sem: sg m}\\ \text{fnc: open}\\ \text{mdr:}\\ \text{nc:}\\ \text{pc:}\\ \text{prn: 23}\end{bmatrix}
\begin{bmatrix}\text{sur:}\\ \text{verb: open}\\ \text{cat: \#s3}'\text{ \#a}'\text{ decl}\\ \text{sem: }\textit{and}\text{ ind past}\\ \text{arg: [person x] fridge}\\ \text{mdr:}\\ \text{nc: (get 24)}^{8}\\ \text{pc:}\\ \text{prn: 23}\end{bmatrix}
\begin{bmatrix}\text{sur:}\\ \text{noun: fridge}\\ \text{cat: snp}\\ \text{sem: def sg}\\ \text{fnc: open}\\ \text{mdr:}\\ \text{nc:}\\ \text{pc:}\\ \text{prn: 23}\end{bmatrix}
\begin{bmatrix}\text{sur:}\\ \text{verb: get}\\ \text{cat: \#s3}'\text{ \#a}'\text{ decl}\\ \text{sem: ind past}\\ \text{arg: [person x] beer}\\ \text{mdr:}\\ \text{nc:}\\ \text{pc:}\\ \text{prn: 24}\end{bmatrix}
\begin{bmatrix}\text{sur:}\\ \text{noun: beer}\\ \text{cat: snp}\\ \text{sem: indef sg}\\ \text{fnc: get}\\ \text{mdr:}\\ \text{nc:}\\ \text{pc:}\\ \text{prn: 24}\end{bmatrix}
\begin{bmatrix}\text{S: kitchen}\\ \text{T: 6pm}\\ \text{A: Peter}\\ \text{R: Lizzy}\\ \text{3rd:}\\ \text{prn: 23}\end{bmatrix}
$$

This content is a token because of the explicit STAR. Temporal priority is encoded by the consecutive prn values 23 and 24. Contiguity is supported intuitively by the content of the two clausal conjuncts (coactivation, Hausser 2021c, 15.6–15.8).

The abstract syntactic-semantic structure of this content is shared by all the other (x)-variants in 1.1 and may be characterized as the following schema:

9.2 CLAUSAL COORDINATION WITH FUNCTIONAL ACCOMMODATION

$$
\begin{bmatrix}\text{noun: }\alpha\\ \text{fnc: }\beta\\ \text{prn: K}\end{bmatrix}
\begin{bmatrix}\text{verb: }\beta\\ \text{sem: }\textit{and}\\ \text{arg: }\alpha\ \gamma\\ \text{nc: }\delta\text{ K+1}\\ \text{prn: K}\end{bmatrix}
\begin{bmatrix}\text{noun: }\gamma\\ \text{fnc: }\beta\\ \text{prn: K}\end{bmatrix}
\begin{bmatrix}\text{verb: }\delta\\ \text{arg: }\alpha\ \varepsilon\\ \text{prn: K+1}\end{bmatrix}
\begin{bmatrix}\text{noun: }\varepsilon\\ \text{fnc: }\delta\\ \text{prn: K+1}\end{bmatrix}
\begin{bmatrix}\text{S: q}\\ \text{T: r}\\ \text{A: s}\\ \text{R: t}\\ \text{3rd:}\\ \text{prn: K}\end{bmatrix}
$$

where β precedes δ, and β and δ are contiguous.

The schema 9.2 is derived from the content 9.1 by simultaneous substitution of the core and continuation values with variables represented by Greek letters. 9.2 matches all contents with the same syntactic structure as 9.1, for example Mary turned_off the light and fell asleep. While the syntactic-semantic structure of clausal coordination is specified by the abstract patterns of the schema 9.2, functional accommodation (as a generalization of Hume's definition 2.1) depends on the prn values and the STAR.

10 Conclusion

Accommodating scenarios are based on the agents' cultural background and personal experiences. Stored in the agents' content-addressable on-board database (memory) and actived (Hausser 2021d, 5.2–5.5) by current nonlanguage and language content processing, accommodating scenarios are an important ingredient of 'making sense.' Speaker and hearer activating the same accommodating scenarios supports reciprocal understanding in natural language communication.

[8] In extrapropositional coordination, the forward direction is implemented routinely in DBS, while the backward direction is handled by an inference which applies only when needed and provides the necessary conjunctions such as before that or earlier (Hausser 2021d, 5.5).

A special case of accommodating scenarios is *functional accommodation*. Syntactically, functional accommodation requires clausal coordination and a certain order of the clausal conjuncts. Semantically it requires the spatiotemporal contiguity of the conjuncts. This equals Hume's (1739) definition of causation.

However, while Hume's causation applies to the agent-external reality, functional accommodation applies to the agent-internal cognition of DBS, which is agent-based data-driven. Because the agent-external reality has necessarily an agent-internal cognitive aspect in DBS, Hume's causation is subject to functional accommodation as well. In this sense, functional accommodation may be viewed as a generalization of Hume's causation.

Technically, DBS cognition is based on an operational analysis of concepts in terms of the agents' recognition and action, the computational data structure of proplets, computational pattern matching between types and tokens (in concepts, proplets, and contents), operations which use the cross-copying of values to establish the semantic relations of structure between the concepts embedded in proplets, and the time, space, and agent information coded in the STAR of clausal content tokens.

Bibliography

Aristotle (1974) *The Complete Works of Aristotle*, edited by Jonathan Barnes, 2 vols., Princeton University Press

Hausser, R. (2021a) "Computational Pragmatics," `lagrammar.net`

Hausser, R. (2021b) "The Grounding of Concepts in Science," `lagrammar.net`

Hausser, R. (2021c) "Agent-Based Memory as an On-Board Database," `lagrammar.net`

Hausser, R. (2021d) "Comparison of Coordination and Gapping," `lagrammar.net`

Hume, D. ([1739] 1896) *A Treatise of Human Nature*, reprinted from the Original Edition in three volumes and edited, with an analytical index, by L.A. Selby-Bigge, Oxford: Clarendon Press.

Hume, D. (1748) *Enquiry concerning Human Understanding*, London: A. Millar

Johansson, I. (2012) "Hume's Ontology," *Metaphysica*, Vol. 13.1:87-105

Mackie, J.L. (1965) "Causes and Conditions,“ *American Philosophical Quarterly*, Vol. 2.4:245–264

Matthews, G. (1990) "Aristotelian Essentialism", *Philosophy and Phenomenomial Research*, Vol. L, Supplement 1990

Newton, I. ([1687, 1713, 1726], 1999) *Philosophiae Naturalis Principia Mathematicae* (The Principia: Mathematical Principles of Natural Philosophy), I.B. Cohen and A. Whitman (Translators), Berkeley: Univ. of California Press

Peirce, C.S. (1931–1935) *Collected Papers*. C. Hartshorne and P. Weiss (eds.), Cambridge, MA: Harvard Univ. Press

Quine, W.v.O. (1966) "Three grades of modal involvement," *The ways of Paradox Papers*, pp. 156–174, New York: Random House

Slavov, M. (2013) "Newton's Law of Universal Gravitation and Hume's Conception of Causality," *Zeitschrift für philosophische Forschung*, Vol. 50.2:277–305

TExer = Hausser, R. (2020) *Twentyfour Exercises in Linguistic Analysis, DBS software design for the Hear and the Speak mode of a Talking Robot*, pp. 324. DOI: 10.13140/RG.2.2.13035.39200, `lagrammar.net`