Classical Syllogisms as Computational Inferences

Roland Hausser

Universität Erlangen-Nürnberg (em.)

Abstract

The classical categorical syllogisms originated and evolved in times without computers, sensors for vision and audition, and actuators for manipulation and vocalization. Therefore, a substitution-driven sign-based ontology was the only practical option.

That it is nevertheless possible to translate categorical syllogisms into the agent-based data-driven inferencing of DBS is because they use the same set-theoretic structures. The following reconstruction proceeds from the diagrams by Swiss mathematician Leonard Euler, used in the year 1761 in a famous 'letter to a princess.'

keywords: sign-based vs. agent-based; logical vs. common sense reasoning; categorical syllogisms; categorical judgments; Euler diagrams;

1 Logical vs. Common Sense Reasoning

A basic distinction in philosophy of language is between logical reasoning and common sense reasoning. Logical reasoning is based on set theory, which is why the associated inferences in DBS are called S-inferences. Common sense reasoning, in contrast, is without a set-theoretic aspect and the associated inferences are called C-inferences in DBS.

In the human prototype, S-inferences and C-inferences are not separated, but work smoothly together. Therefore, the computational model of reasoning in DBS uses the same general inference schema and the same data structure for S- and C-inferences. Consider the following DBS inferences as schematic examples:

1.1 Example of an S-inference (FERIO)

$$\begin{tikzcd}
\text{S-inference:} & \alpha\text{ is homework} \arrow[r, Rightarrow] & \alpha\text{ is no fun} \arrow[d, Rightarrow] \\
& \text{input: some reading is homework} \arrow[u, Rightarrow] & \text{output: some reading is no fun}
\end{tikzcd}$$

1.2 Example of a C-inference (CAUSE_and_EFFECT)

$$\begin{tikzcd}
\text{C-inference:} & \alpha\text{ is hungry} \arrow[r, Rightarrow] & \alpha\text{ is cranky} \arrow[d, Rightarrow] \\
& \text{input: Laura is hungry} \arrow[u, Rightarrow] & \text{output: Laura is cranky}
\end{tikzcd}$$

Both inferences work by binding the subject term of the input to the variable α in the antecedent of the pattern, which enables the consequent to derive the output.

The implementation of the syllogism FERIO as an S-inference is illustrated in 6.10, while the corresponding implementation of the C-inference 1.2 is shown as the following software operation:

1.3 Applying the C-inference 1.2 in DBS format

$$\begin{array}{llllll}
\begin{array}{l}\textit{pattern}\\ \textit{level}\end{array} & \begin{bmatrix}\text{noun: }\alpha\\ \text{fnc: be\_hungry}\\ \text{prn: K}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_hungry}\\ \text{arg: }\alpha\\ \text{prn: K}\end{bmatrix} & \Rightarrow & \begin{bmatrix}\text{noun: }(\alpha\text{ K})\\ \text{fnc: be\_cranky}\\ \text{prn: K+1}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_cranky}\\ \text{arg: }(\alpha\text{ K})\\ \text{prn: K+1}\end{bmatrix}\\
 & & \Uparrow & & \Downarrow & \\
\begin{array}{l}\textit{content}\\ \textit{level}\end{array} & \begin{bmatrix}\text{sur: laura}\\ \text{noun: (girl x)}\\ \text{cat: snp}\\ \text{sem: nm f}\\ \text{fnc: be\_hungry}\\ \text{prn: 24}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_hungry}\\ \text{arg: (girl x)}\\ \text{prn: 24}\end{bmatrix} & & \begin{bmatrix}\text{sur: laura}\\ \text{noun: (girl x)}\\ \text{cat: snp}\\ \text{sem: nm f}\\ \text{fnc: be\_cranky}\\ \text{prn: 25}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_cranky}\\ \text{arg: (girl x)}\\ \text{prn: 25}\end{bmatrix}
\end{array}$$

The content and the pattern level consist of nonrecursive feature structures with ordered attributes. Called proplets, they serve as the computational data structure. The proplets of a content are order-free but connected by the classical semantic relations of structure, i.e., functor-argument and coordination, coded by address.

The proplets at the pattern level use variables as values for the ‘core’ and the ‘continuation’ attributes, those at the content level have corresponding constants. For computational pattern matching to be successful (i) the attributes of the pattern proplet must be a *sublist*, (ii) the variables of the pattern proplet must be *compatible*, and (iii) the constants of the pattern proplet must be *identical* with those of the corresponding content proplet directly underneath.

S-inferences and C-inferences differ in the source of their reasoning. For example, in the S-inference 1.1, the source is the disjunction between the concepts homework and be_fun and the intersection between reading and homework (6.8), which are assumed to be generally accepted. In the C-inference 1.2, in contrast, the source is something observed by the agent(s).[1]

2 Categorical Syllogisms

An early highlight in the Western tradition of logical reasoning are the classical syllogisms of Aristotle (384–322 BC) and their further development by the medieval scholastics.[2] In the modern era, the syllogisms have been based on the intuitions of set theory (Euler, 2.3).

A categorical[3] syllogism consists of three parts, called premise 1, premise 2, and the conclusion. This may be shown schematically as follows:

[1] The resulting set-theoretic relation between being cranky and being hungry in 1.2, i.e., intersection, is merely a consequence of the reasoning, and not the source.

[2] For a critical review of how the understanding of Aristotle’s theory of categorical syllogisms changed over the millenia see Read (2017). For a computational automata and factor analysis see Zhang Yinsheng and Qiao Xiaodong (2009).

2.1 Schematic instantiation of a categorical syllogism

Major premise: all M are P
Minor premise: all S are M
Conclusion: all S are P

M is the middle term, S the subject, and P the predicate. M is shared by the two premises. The positions of M are called the alignment of a syllogism.

The three parts of a classical syllogism are restricted to the four categorical judgments, named **A, E, I**, and **O** by the Scholastics:

2.2 The four categorical judgments

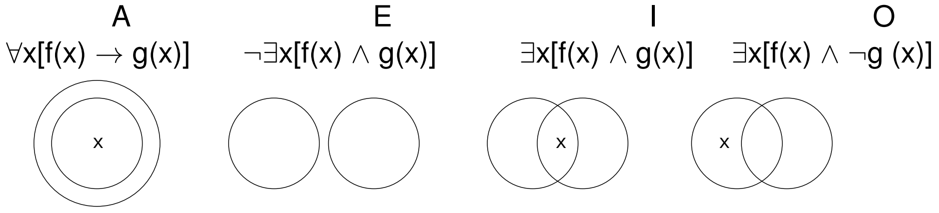
A	universal affirmative	$\forall x\, [\, f(x) \rightarrow g(x)\,]$	all f are g
E	universal negative	$\neg\exists x\, [\, f(x) \wedge g(x)\,]$	no f are g
I	particular affirmative	$\exists x\, [\, f(x) \wedge g(x)\,]$	some f are g
O	particular negative	$\exists x\, [\, f(x) \wedge \neg g(x)\,]$	some f are not g

The first-order Predicate Calculus representation in the 3rd column is in a linear notation called *prenex normal form*, which superseded Frege's (1879) graphical format.

The four categorical judgments combine into 256 (2^8) possible syllogism, of which 24 have been found valid. The syllogisms reconstructed in this paper as DBS inferences are BARBARA,[4] CELARENT, DARII, FERIO, BAROCO, and BOCARDO, plus the modi ponendo ponens and tollendo tollens as special[5] cases.

The set-theoretic constellations underlying the four categorical judgments may be shown as follows:

2.3 Set-theoretic counterparts of the four categorical judgments


A
$\forall x[f(x) \rightarrow g(x)]$
x
E
$\neg\exists x[f(x) \wedge g(x)]$
I
$\exists x[f(x) \wedge g(x)]$
x
O
$\exists x[f(x) \wedge \neg g\ (x)]$
x

Known as Euler diagrams[6] (Euler 1761), the set-theoretic constellations are used in DBS to reconstruct the valid syllogisms as data-driven, agent-based S-inferences. As an example, consider the schematic application of modus Barbara in DBS:

[4] The scholastics used the vowels of the categorical judgments in the names of the associated syllogisms as mnemonic support. For example, the three vowels in the name of modus BARBARA indicate that the categorical judgments of the propositions serving as the two premises and the consequent are all of the kind **A**, i.e., universal affirmative (2.2).

[5] The modi ponendo ponens (3.1) and tollendo tollens (4.1) are not categorical syllogism in the narrow sense because their premise 2 and conclusion are not categorical judgments of the kind **A**, **E**, **I**, or **O**. This is reflected by their different naming convention as compared to categorical syllogisms in the narrow sense, for example BARBARA or FERIO.

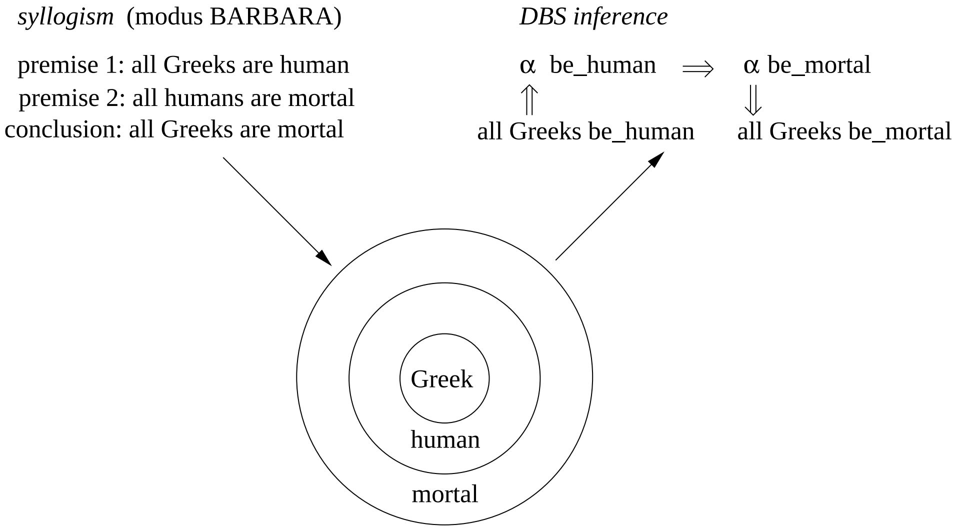
2.4 Modus Barbara as a DBS inference

$$\begin{array}{llcl} \text{inference:} & \alpha\ \text{be\_human} & \Rightarrow & \alpha\ \text{be\_mortal} \\ & \Uparrow & & \Downarrow \\ & \text{input: all Greeks be\_human} & & \text{output: all Greeks be\_mortal} \end{array}$$

A DBS inference takes a given content as input and validly derives a new content as output. Validity follows from set-theoretic intuitions which are the foundation of both the sign-based classical syllogisms and their agent-based DBS counterparts.

The transition from a categorical syllogism to a DBS inference may be shown as follows:

2.5 From syllogism to DBS inference



syllogism (modus BARBARA)

premise 1: all Greeks are human
premise 2: all humans are mortal
conclusion: all Greeks are mortal

DBS inference

$$\begin{array}{lcl} \alpha\ \text{be\_human} & \Longrightarrow & \alpha\ \text{be\_mortal} \\ \Uparrow & & \Downarrow \\ \text{all Greeks be\_human} & & \text{all Greeks be\_mortal} \end{array}$$

The DBS reconstruction of a categorical syllogism as an inference has the form α *X implies* α *Y*, whereby the variable α represents a grammatical subject, and *X* and *Y* represent grammatical predicates. The α in the antecedent may be matched by and bound to (1) a complete set, e.g., all Greeks (universal, 4.4), (2) a subset, e.g., some pets (particular, 6.4), or (3) an element, e.g., Socrates (individual, 6.3). In the consequent, the input-binding of α derives the output.

With the possible presence of negation in the antecedent, the consequent, or both, there result the following four schemata of S-inferences for the categorical syllogisms, each with a universal, a particular, and an individual variant.

The first triple is without negation:

[6] Named after Leonhard Euler (1707–1783). the method was known already in the 17th century and has been credited to several candidates.

Venn (1881, p.113) called Euler diagrams "old-fashioned". Euler diagrams reflect the set-theoretic constellations simple and direct, whereas Venn models the complicated medieval superstructures erected by the scholastics on top of the original syllogisms. Venn diagrams are useful for showing that certain syllogisms, for example, EEE-1 and OOO-1, are not valid.

2.6 α be_X implies α be_Y

(1) universal: all Greeks be_human implies all Greeks be_mortal.
(2) particular: some pets be_rabbits implies some pets be_furry.
(3) individual: Socrates be_human implies Socrates be_mortal.

The universal version is modeled after modus BARBARA (5.1), the particular version after modus DARII (6.1), and the individual version after modus ponendo ponens (3.1).

The second triple negates the consequent:

2.7 α be_X implies α not be_Y

(4) universal: all horses be_quadruped implies all horses not be_human.
(5) particular: some pets be_turtles implies some pets not be_furry.
(6) individual: Pegasus be_quadruped implies Pegasus not be_human.

The universal version is modeled after modus CELARENT (5.6), the particular version after modus FERIO (6.6), and the individual version after modus tollendo tollens (4.1).

The third triple negates the antecedent:

2.8 α not be_X implies α be_Y

(7) universal: all friars not be_married implies all friars be_single.
(8) particular: some men not be_married implies some men be_single.
(9) individual: Fred not be_married implies Fred be_single.

Set-theoretically, the denotations of not be_married and of be_single are coextensive in all three versions.

The fourth triple negates the antecedent and the consequent. Though EEE syllogisms are not valid for all instantiations, the following instantiations are:

2.9 α not be_X implies α not be_Y

(10) universal: all gods not be_mortal implies all gods not be_human.
(11) particular: some pets not be_furry implies some pets not be_rabbits.
(12) individual: Zeus not be_mortal implies Zeus not be_human.

Set-theoretically, the denotations of not be_X and not be_Y are disjunct in the (10) universal and the (12) individual variant, and in the complement of the pet-rabbit intersection in the (11) particular variant.

3 Modus Ponendo Ponens

Modus ponendo[7] ponens serves as the individual version of 2.6. The standard representation in Predicate Calculus is as follows:

3.1 Modus ponendo ponens in Predicate Calculus

$$\begin{array}{r l} \text{premise 1:} & \forall x[\mathsf{f}(x) \rightarrow \mathsf{g}(x)] \\ \text{premise 2:} & \exists y[\mathsf{f}(y)] \\ \hline \text{conclusion:} & \exists z[\mathsf{g}(z)] \end{array}$$

Instantiating f as be_human and g as be_mortal has the following result:

3.2 Instantiating modus ponendo ponens

$$\begin{array}{r l} \text{premise 1:} & \text{For all x, if x is human, then x is mortal.} \\ \text{premise 2:} & \text{There exists a y, such that y is human.} \\ \hline \text{conclusion:} & \text{There exists a z, such that z is mortal.} \end{array}$$

The reconstruction of modus ponendo ponens (NLC 5.3) in DBS (i) turns premise 1 into the form α is human implies α is mortal, called the inference, (ii) uses premise 2 as the input, and (iii) treats the conclusion as the output: depending on whether or not the antecedent, the consequent, or both are negated.

3.3 Rephrasing modus ponendo ponens in DBS

inference: α be_human implies α be_ mortal.
input: Socrates be_human.[8]
output: Socrates be_mortal.

Shown here with input for modus ponens (individual), the inference works just as well for particular (2.6, 1) and universal (2.6, 2) input.

Using the DBS data structure, the inference applies as follows to the modus ponendo ponens input of 3.3:

[7] In Propositional Calculus, modus ponendo ponens and modus tollendo ponens differ as follows: modus ponendo ponens has the premises (A→B) and A, resulting in the conclusion B; modus tollendo ponens has the premises (A∨B) and ¬A, also resulting in B. The distinction disappears in the functor-argument structure of DBS.

[8] Predicate Calculus treats the copula-adnominal combination is human as the elementary proposition be_human(x) which denotes a truth value. DBS, in contrast, analyzes is human as the modifier|modified (or rather modified|modifier) combination is|human (CC 4.6). For comparison, simplicity, and brevity we compromise here by using the Predicate Calculus notation, e.g., be_human, like intransitive verbs as values in proplets, but without any variable.

3.4 Applying modus ponendo ponens as formalized in DBS

$$\begin{array}{lcccccc}
 & & \textit{antecedent} & & & \textit{consequent} & \\
\begin{array}{l}\textit{pattern}\\ \textit{level}\end{array} & \begin{bmatrix}\text{noun: }\alpha\\ \text{fnc: be\_human}\\ \text{prn: K}\end{bmatrix} & & \begin{bmatrix}\text{verb: be\_human}\\ \text{arg: }\alpha\\ \text{prn: K}\end{bmatrix} & \Rightarrow & \begin{bmatrix}\text{noun: }(\alpha\text{ K})\\ \text{fnc: be\_mortal}\\ \text{prn: K+1}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_mortal}\\ \text{arg: }(\alpha\text{ K})\\ \text{prn: K+1}\end{bmatrix} \\
 & & \Uparrow & & & & \Downarrow \\
\begin{array}{l}\textit{content}\\ \textit{level}\end{array} & \begin{bmatrix}\text{sur: socrates}\\ \text{noun: (person x)}\\ \text{cat: snp}\\ \text{sem: nm m}\\ \text{fnc: be\_human}\\ \text{prn: 23}\end{bmatrix} & & \begin{bmatrix}\text{verb: be\_human}\\ \text{cat: \#ns3}'\text{ decl}\\ \text{sem: pres}\\ \text{arg: (person x)}\\ \text{prn: 23}\end{bmatrix} & & \begin{bmatrix}\text{sur: socrates}\\ \text{noun: (person x)}\\ \text{cat: snp}\\ \text{sem: nm m}\\ \text{fnc: be\_mortal}\\ \text{prn: 23+1}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_mortal}\\ \text{cat: \#ns3}'\text{ decl}\\ \text{sem: pres}\\ \text{arg: (person x)}\\ \text{prn: 23+1}\end{bmatrix}
\end{array}$$

The DBS reinterpretation of premise 1 as the inference and premise 2 as the input requires that the input be compatible for matching with the antecedent. This would be prevented, however, if the antecedent specified the noun pattern α as a plural, corresponding to ∀x in premise 1 of 3.1, and premise 2 as a singular, corresponding to ∃y. Therefore, the noun pattern α in 3.4 omits the cat and sem features, thus enabling matching (compatibility by omission). By vertically binding the constant socrates of the content level to the variable α in the antecedent of the pattern level, the consequent derives the new content socrates is mortal as output.

4 Modus Tollendo Tollens

Modus tollendo[9] tollens serves as the individual version of 2.7. A standard representation in Predicate Calculus is as follows:

4.1 Modus tollendo tollens in Predicate Calculus

$$\begin{array}{r}
\text{premise 1: } \forall x[f(x) \rightarrow g(x)]\\
\underline{\text{premise 2: } \exists y[\neg g(y)]}\\
\text{conclusion: } \exists z[\neg f(z)]
\end{array}$$

Let us instantiate f as is human and g as is biped:

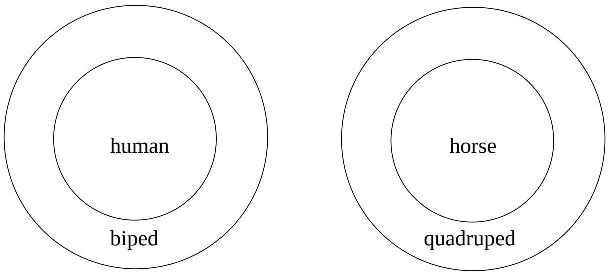
4.2 Instantiating modus tollendo tollens

premise 1: For all x, if x is human, then x is biped.
premise 2: there exists a y which is not biped.
conclusion: There exists a z which is not human.

[9] In Propositional Calculus, modus ponendo ponens and modus tollendo ponens differ as follows: modus ponendo ponens has the premises (A→B) and A, resulting in the conclusion B; modus tollendo ponens has the premises (A∨B) and ¬A, also resulting in B. The distinction disappears in the functor-argument structure of DBS.

If we use quadruped to instantiate not be_biped and horse to instantiate not be_human, the set-theoretic constellation underlying modus tollendo tollens in 4.2 may be depicted as follows:

4.3 Set-theoretic view of modus tollendo tollens



Premise 1 corresponds to the set structure on the left, premise 2 to the set structure on the right (with pegasus as the individual instantiation of horse, and the sets horse and human being disjunct).

The DBS reconstruction uses the disjunction of the sets quadruped and human for the inference α be_quadruped implies α not be_human. Pegasus being an element of the set quadruped renders the input Pegasus is quadruped. Pegasus not being an element of the set human renders the output Pegasus not be_human.

4.4 Rephrasing modus tollendo tollens in DBS

inference:	α be_quadruped implies α not be_human.
input:	Pegasus be_quadruped.
output:	Pegasus not be_human.

The inference works for the individual, the particular, and the universal variant of 2.7. The variants differ solely in their input and output.

Let us conclude with the translation of 4.4 into the data structure of DBS:

4.5 Applying modus tollendo tollens as formalized in DBS

$$\begin{array}{llllll}
\begin{array}{l}\textit{pattern}\\ \textit{level}\end{array} & \begin{bmatrix}\text{noun: } \alpha\\ \text{fnc: be\_quadr.}\\ \text{prn: K}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_quadr.}\\ \text{arg: } \alpha\\ \text{prn: K}\end{bmatrix} & \Rightarrow & \begin{bmatrix}\text{noun: } (\alpha\ \text{K})\\ \text{fnc: be\_human}\\ \text{prn: K+1}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_human}\\ \text{arg: } (\alpha\ \text{K})\\ \text{sem: } \textit{not}\\ \text{prn: K+1}\end{bmatrix} \\
 & & \Uparrow & & & \Downarrow \\
\begin{array}{l}\textit{content}\\ \textit{level}\end{array} & \begin{bmatrix}\text{sur: pegasus}\\ \text{noun: [horse x]}\\ \text{sem: nm m}\\ \text{fnc: be\_quadr.}\\ \text{prn: 23}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_quadr.}\\ \text{sem:}\\ \text{arg: [horse x]}\\ \text{prn: 23}\end{bmatrix} & & \begin{bmatrix}\text{sur: pegasus}\\ \text{noun: [horse x]}\\ \text{sem: nm m}\\ \text{fnc: be\_human}\\ \text{prn: 23+1}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_human}\\ \text{sem: } \textit{not}\\ \text{arg: [horse x]}\\ \text{prn: 23+1}\end{bmatrix}
\end{array}$$

The transfer of syllogisms from sign-based substitution-driven symbolic logic to agent-based data-driven DBS relies on DBS inferencing being part of the think mode, which may run detached from the agent's interface component (CC, mediated reference 3.1.3, sequential application 3.6.2).

5 Modi BARBARA and CELARENT

The vowels in the name of modus BARBARA indicate the categorical judgments of the propositions serving as the premises and the consequent, which are all of type **A**, i.e., universal affirmative (2.2).

5.1 Modus BARBARA in Predicate Calculus

$$\begin{array}{r l}\text{premise 1:} & \forall x[\mathsf{f}(x) \rightarrow \mathsf{g}(x)] \\ \text{premise 2:} & \forall y[\mathsf{g}(y) \rightarrow \mathsf{h}(y)] \\ \hline \text{conclusion:} & \forall z[\mathsf{f}(z) \rightarrow \mathsf{h}(z)]\end{array}$$

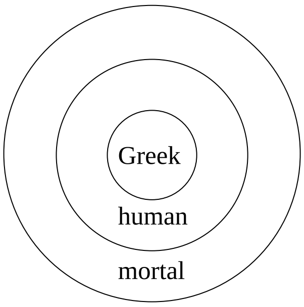
The middle term is g. If f is realized as be_Greek, g as be_human, and h as be_mortal, then the syllogism reads as follows:

5.2 Instantiating modus BARBARA

premise 1: For all x, if x be_Greek, then x be_human.
premise 2: For all y, if y be_human, then y be_mortal.
conclusion: For all z, if z are Greek, then z be_mortal.

The set-theoretic constellation underlying modus BARBARA in 5.2 may be depicted as follows:

5.3 Set-theoretic view of modus BARBARA



Premise 1 is expressed by the set Greek being a subset of human and premise 2 by the set Greek being a subset of mortal.

The inference schema of DBS formulates the set-theoretic constellation as follows:

5.4 Rephrasing modus BARBARA in DBS

inference: α be_human implies α be_mortal.
input: All Greeks be_human.
output: All Greeks be_mortal.

The validity of the inference follows directly from the subset relations Greek ⊂ human ⊂ mortal, which are inherent in the extensions of these concepts.

Consider the translation of 5.4 into the data structure of DBS:

5.5 Applying modus BARBARA as formalized in DBS

$$\begin{array}{lccccc}
\begin{array}{l}\textit{pattern}\\ \textit{level}\end{array} & \begin{bmatrix}\text{noun: } \alpha \\ \text{fnc: be\_human} \\ \text{prn: K}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_human} \\ \text{arg: } \alpha \\ \text{prn: K}\end{bmatrix} & \Rightarrow & \begin{bmatrix}\text{noun: } (\alpha\ \text{K}) \\ \text{fnc: be\_mortal} \\ \text{prn: K+1}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_mortal} \\ \text{arg: } (\alpha\ \text{K}) \\ \text{prn: K+1}\end{bmatrix} \\
 & \Uparrow & & & & \Downarrow \\
\begin{array}{l}\textit{content}\\ \textit{level}\end{array} & \begin{bmatrix}\text{noun: greek} \\ \text{cat: pnp} \\ \text{sem: pl exh} \\ \text{fnc: be\_human} \\ \text{prn: 23}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_human} \\ \text{arg: greek} \\ \text{prn: 23}\end{bmatrix} & & \begin{bmatrix}\text{noun: (greek 23)} \\ \text{cat: pnp} \\ \text{sem: pl exh} \\ \text{fnc: be\_mortal} \\ \text{prn: 23+1}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_mortal} \\ \text{arg: (greek 23)} \\ \text{prn: 23+1}\end{bmatrix}
\end{array}$$

By vertically binding greek of the content level to the variable α in the antecedent of the pattern level, the consequent derives the desired new content All Greeks are mortal as output. In the class of syllogisms with unnegated antecedent and unnegated consequent (2.6), the reconstruction of BARBARA constitutes the universal, of DARII 6.5 the particular, and of modus ponendo ponens 3.4 the individual variant.

Next let us turn to a syllogism with the vowel **E** in its name, where **E** indicates a universal negative (2.2). The vowels in the name of modus CELARENT, for example, indicate that premise 1 is of type **E**, premise 2 of type **A**, and the conclusion of type **E**. In Predicate Calculus, CELARENT is represented as follows:

5.6 Modus CELARENT in Predicate Calculus

$$\begin{array}{rl}
\text{premise 1:} & \neg\exists x[f(x) \wedge g(x)] \\
\text{premise 2:} & \forall y[h(y) \rightarrow f(y)] \\
\hline
\text{conclusion:} & \neg\exists z[h(z) \wedge g(z)]
\end{array}$$

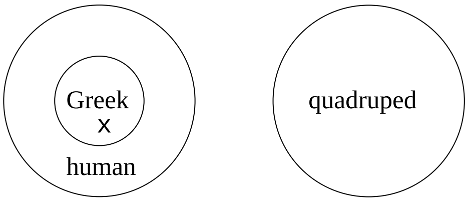
The middle term is f. If f is realized as human, g as quadruped, and h as Greek, then the syllogism reads as follows:

5.7 Instantiating CELARENT in Predicate Calculus

$$\begin{array}{rl}
\text{premise 1:} & \neg\exists x[\text{human}(x) \wedge \text{quadruped}(x)] \\
\text{premise 2:} & \forall y[\text{Greek}(y) \rightarrow \text{human}(y)] \\
\hline
\text{conclusion:} & \neg\exists z[\text{Greek}(z) \wedge \text{quadruped}(z)]
\end{array}$$

The set-theoretic constellation underlying modus CELARENT in 5.7 may be depicted as follows:

5.8 Set-theoretic view of modus CELARENT



Premise 1 is expressed by the sets human and quadruped being disjunct. Premise 2 is expressed by the set Greek being a subset of human. The conclusion is expressed by the sets Greek and quadruped being disjunct.

The inference schema of DBS describes the set-theoretic constellation as follows:

5.9 Rephrasing modus CELARENT in DBS

inference: α be_human implies α not be_quadruped
input: All Greeks be_human
output: All Greeks not be_quadruped

Consider the translation of 5.9 into the data structure of DBS:

5.10 Applying modus CELARENT as formalized in DBS

$$\begin{array}{lcccccc}
\textit{pattern level} & \begin{bmatrix}\text{noun: }\alpha\\ \text{fnc: be\_human}\\ \text{prn: K}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_human}\\ \text{arg: }\alpha\\ \text{prn: K}\end{bmatrix} & \Rightarrow & \begin{bmatrix}\text{noun: }(\alpha\ \text{K})\\ \text{fnc: be\_quadr.}\\ \text{prn: K+1}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_quadr.}\\ \text{sem: }\textit{not}\\ \text{arg: }(\alpha\ \text{K})\\ \text{prn: K+1}\end{bmatrix} \\
 & & \Uparrow & & & \Downarrow \\
\textit{content level} & \begin{bmatrix}\text{noun: greek}\\ \text{cat: pnp}\\ \text{sem: pl exh}\\ \text{fnc: be\_human}\\ \text{prn: 23}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_human}\\ \text{arg: greek}\\ \text{prn: 23}\end{bmatrix} & & \begin{bmatrix}\text{noun: (greek 23)}\\ \text{cat: pnp}\\ \text{sem: pl exh}\\ \text{fnc: be\_quadr.}\\ \text{prn: 23+1}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_quadr.}\\ \text{sem: }\textit{not}\\ \text{arg: (greek 23)}\\ \text{prn: 23+1}\end{bmatrix}
\end{array}$$

By binding greek of the content level to the variable α in the antecedent of the pattern level, the consequent derives the desired new content All Greeks are not quadruped as output. The ∀x quantifier of Predicate Calculus is coded in the *greek* proplet by the feature [sem: pl exh] and the negation in the conclusion is coded in the predicate *be_quadruped* by the feature [sem: *not*].

6 Modi DARII and FERIO

The DBS variants of modus BARBARA (5.5) and modus CELARENT (5.10) have shown the treatment of the categorical judgments (2.2) **A** (universal affirmative) and **E** (universal negative). To show the treatment of the remaining categorical judgments **I** (particular affirmative) and **O** (particular negative), let us reconstruct the modi DARII and FERIO as DBS inferences.

The vowels in the name DARII indicate the categorical judgment **A** in premise 1, and **I** in premise 2 and the conclusion. The representation in Predicate Calculus is as follows:

6.1 Modus DARII in Predicate Calculus

$$\begin{array}{rl} \text{premise 1:} & \forall x[\mathsf{f}(x) \rightarrow \mathsf{g}(x)] \\ \text{premise 2:} & \exists y[\mathsf{h}(y) \wedge \mathsf{f}(y)] \\ \hline \text{conclusion:} & \exists z[\mathsf{h}(z) \wedge \mathsf{g}(z)] \end{array}$$

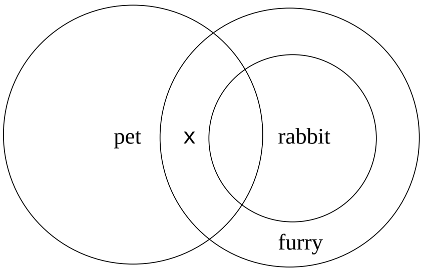
The middle term is f. If f is instantiated as be_rabbit, g as be_furry, and h as be_pet, then the syllogism reads as follows:

6.2 Instantiating modus DARII

$$\begin{array}{rl} \text{premise 1:} & \text{For all x, if x is rabbit, then x is furry.} \\ \text{premise 2:} & \text{For some y, y is pet and y is rabbit.} \\ \hline \text{conclusion:} & \text{For some z, z is pet and z is furry.} \end{array}$$

The set-theoretic constellation underlying modus DARII in 6.2 may be depicted as follows:

6.3 Set-theoretic view of modus DARII



Premise 1 is expressed by the set rabbit being a subset of furry. Premise 2 is expressed by the set pet overlapping with the set rabbit. The conclusion is expressed by the set pet overlapping with the set furry.

DBS describes the set-theoretic constellation as follows:

6.4 Rephrasing modus DARII in DBS

inference: α be_rabbit implies α be_furry.
input: Some pets be_rabbit
output: Some pets be_furry

The inference applies by binding *some pets* in the input to the variable α in the antecedent and using this binding in the consequent to derive the output. The input matches the antecedent and the consequent derives matching output.

Following standard procedure, this is shown in detail by the following translation of 6.4 into the data structure of DBS:

6.5 Applying modus DARII as formalized in DBS

$$
\begin{array}{lcccc}
\textit{pattern level} &
\begin{bmatrix}\text{noun: }\alpha\\ \text{fnc: be\_rabbit}\\ \text{prn: K}\end{bmatrix}
\begin{bmatrix}\text{verb: be\_rabbit}\\ \text{arg: }\alpha\\ \text{prn: K}\end{bmatrix}
& \Rightarrow &
\begin{bmatrix}\text{noun: }(\alpha\text{ K})\\ \text{fnc: be\_furry}\\ \text{prn: K+1}\end{bmatrix}
\begin{bmatrix}\text{verb: be\_furry}\\ \text{arg: }(\alpha\text{ K})\\ \text{prn: K+1}\end{bmatrix}
\\
& \Uparrow & & \Downarrow \\
\textit{content level} &
\begin{bmatrix}\text{noun: pet}\\ \text{cat: pnp}\\ \text{sem: pl sel}\\ \text{fnc: be\_rabbit}\\ \text{prn: 23}\end{bmatrix}
\begin{bmatrix}\text{verb: be\_rabbit}\\ \text{arg: pet}\\ \text{prn: 23}\end{bmatrix}
& &
\begin{bmatrix}\text{noun: (pet 23)}\\ \text{cat: pnp}\\ \text{sem: pl sel}\\ \text{fnc: be\_furry}\\ \text{prn: 23+1}\end{bmatrix}
\begin{bmatrix}\text{verb: be\_furry}\\ \text{arg: (pet 23)}\\ \text{prn: 23+1}\end{bmatrix}
\end{array}
$$

The particular affirmative quality of the judgment type **I**, i.e., the some, is coded by the features [cat: pnp] and [sem: pl sel] of the *pet* proplets at the content level. Because the grammatical properties of determiners are not reflected at the pattern level (compatibility by omission), modus DARII joins modus BARBARA and modus ponendo ponens as an instance of the DBS inference kind unnegated antecedent and unnegated consequent (2.6) in the variant *particular*.

We turn next to modus FERIO. The vowels in the name indicate the categorical judgment **E** (universal negative) in premise 1, **I** (particular affirmative) in premise 2, and **O** (particular negative) in the conclusion. In Predicate Calculus, this is formalized as follows:

6.6 Modus FERIO in Predicate Calculus

premise 1: $\neg\exists x[f(x) \wedge g(x)]$
premise 2: $\exists y[h(y) \wedge g(y)]$
conclusion: $\exists z[h(z) \wedge \neg g(z)]$

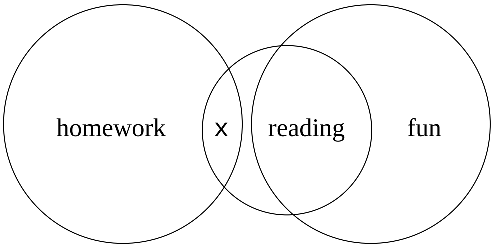
The middle term is g. If f is instantiated as is homework, g as is fun, and h as is reading, then the syllogism reads as follows:

6.7 Instantiating modus FERIO

premise 1: There exists no x, x is homework and x is fun.
premise 2: For some y, y is reading and y is homework
conclusion: For some z, z is reading and z is no fun

The set-theoretic constellation underlying modus FERIO in 6.7which may be depicted as follows:

6.8 Set-theoretic view of modus FERIO



Premise 1 is shown by the sets homework and fun being disjunct. Premise 2 is depicted by the sets reading and homework overlapping. The conclusion is shown by the sets reading and homework, and reading and fun overlapping.

The inference schema of DBS describes the set-theoretic constellation as follows:

6.9 Rephrasing modus FERIO in DBS

inference: α be_homework implies α not be_fun.
input: Some reading be_homework
output: Some reading not be_fun

Consider the translation of 6.9 into the data structure of DBS:

6.10 Applying modus FERIO as formalized in DBS

$$
\begin{array}{llllll}
\textit{pattern level} & \begin{bmatrix}\text{noun: }\alpha\\ \text{fnc: be\_homew.}\\ \text{prn: K}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_homew.}\\ \text{arg: }\alpha\\ \text{prn: K}\end{bmatrix} & \Rightarrow & \begin{bmatrix}\text{noun: }(\alpha\text{ K})\\ \text{fnc: be\_fun}\\ \text{prn: K+1}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_fun}\\ \text{sem: }\textit{not}\\ \text{arg: }(\alpha\text{ K})\\ \text{prn: K+1}\end{bmatrix}\\
 & & \Uparrow & & & \Downarrow\\
\textit{content level} & \begin{bmatrix}\text{noun: reading}\\ \text{cat: snp}\\ \text{sem: pl sel}\\ \text{fnc: be\_homew.}\\ \text{prn: 23}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_homew.}\\ \text{arg: reading}\\ \text{prn: 23}\end{bmatrix} & & \begin{bmatrix}\text{noun: (reading 23)}\\ \text{cat: snp}\\ \text{sem: pl sel}\\ \text{fnc: be\_fun}\\ \text{prn: 23+1}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_fun}\\ \text{sem: }\textit{not}\\ \text{arg: (reading 23)}\\ \text{prn: 23+1}\end{bmatrix}
\end{array}
$$

The content noun some reading is characterized by the features [cat: snp] and [sem: pl sel]. The particular negative quality of the judgment type **O** is coded by the feature [sem: *neg*] in the *be_fun* proplets at the pattern as well as the content level. The reconstruction of FERIO in DBS joins the inferences of the kind unnegated antecedent and negated consequent (2.7) in the variant *particular*.

7 Modi BAROCO and BOCARDO

Like modus FERIO, modus BAROCO has the particular negative **O** in the conclusion. The **A** representing premise 1 indicates the categorical judgment universal affirmative (2.2).

7.1 Modus BAROCO in Predicate Calculus

$$\begin{array}{rl} \text{premise 1:} & \forall x[f(x) \rightarrow \mathsf{g}(x)] \\ \text{premise 2:} & \exists y[\mathsf{h}(y) \wedge \neg \mathsf{g}(y)] \\ \hline \text{conclusion:} & \exists z[\mathsf{h}(z) \wedge \neg \mathsf{f}(z)] \end{array}$$

The middle term is g. If f is instantiated as informative, g as useful, and h as website, then the syllogism reads as follows:

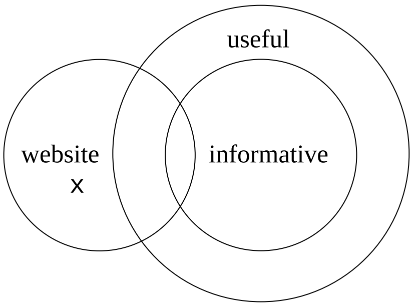
7.2 Instantiating Modus BAROCO

$$\begin{array}{rl} \text{premise 1:} & \text{All informative things are useful} \\ \text{premise 2:} & \text{Some website are not informative} \\ \hline \text{conclusion:} & \text{Some websites are not useful} \end{array}$$

Among the classical syllogisms, BAROCO is special because the proof of its validity requires a reductio per impossibile.

The set-theoretic constellation underlying modus BAROCO in 7.2 may be depicted as follows:

7.3 Set-theoretic view of modus BAROCO



Premise 1 is shown by the set informative being a subset of useful. Premise 2 is depicted by the set website merely overlapping with the set informative. The conclusion is shown by the set website merely overlapping with useful.

The inference schema of DBS describes the set-theoretic constellation as follows:

7.4 Rephrasing BAROCO in DBS

inference: α not be_informative implies α not be_useful
input: Some websites not be_informative
output: Some websites not be_useful

Consider the translation of 7.4 into the data structure of DBS:

7.5 BAROCO in DBS

$$\begin{array}{llllll}
\begin{array}{l}\textit{pattern}\\ \textit{level}\end{array} &
\begin{bmatrix}\text{noun: }\alpha\\ \text{fnc: be\_informative}\\ \text{prn: K}\end{bmatrix} &
\begin{bmatrix}\text{verb: be\_informative}\\ \text{sem: }\textit{not}\\ \text{arg: }\alpha\\ \text{prn: K}\end{bmatrix} &
\Rightarrow &
\begin{bmatrix}\text{noun: }(\alpha\text{ K})\\ \text{fnc: be\_useful}\\ \text{prn: K+1}\end{bmatrix} &
\begin{bmatrix}\text{verb: be\_useful}\\ \text{sem: }\textit{not}\\ \text{arg: }(\alpha\text{ K})\\ \text{prn: K+1}\end{bmatrix} \\
& & \Uparrow & & & \Downarrow \\
\begin{array}{l}\textit{content}\\ \textit{level}\end{array} &
\begin{bmatrix}\text{noun: website}\\ \text{cat: pnp}\\ \text{sem: pl sel}\\ \text{fnc: be\_informative}\\ \text{prn: 25}\end{bmatrix} &
\begin{bmatrix}\text{verb: be\_informative}\\ \text{sem: }\textit{not}\\ \text{arg: website}\\ \text{prn: 25}\end{bmatrix} &
&
\begin{bmatrix}\text{noun: website}\\ \text{cat: pnp}\\ \text{sem: pl sel}\\ \text{fnc: be\_useful}\\ \text{prn: 24}\end{bmatrix} &
\begin{bmatrix}\text{verb: be\_useful}\\ \text{sem: }\textit{not}\\ \text{arg: website}\\ \text{prn: 24}\end{bmatrix}
\end{array}$$

The reconstruction of BAROCO in DBS joins inferences of the kind negated antecedent and negated consequent (2.9) in the variant *particular*.

Like modus BAROCO, modus BOCARDO has the particular negative **O** in the conclusion. They differ in that the letters **A** and **O** in premises 1 and 2 are interchanged.

7.6 Modus BOCARDO in Predicate Calculus

$$\begin{array}{rl}
\text{premise 1:} & \exists x[f(x) \wedge \neg g(x)]\\
\text{premise 2:} & \forall y[f(y) \rightarrow h(y)]\\
\hline
\text{conclusion:} & \exists z[h(z) \wedge \neg g(z)]
\end{array}$$

The middle term is f. If f is instantiated as be_cat, g as has_tail, and h as be_mammal, then the syllogism reads as follows:

7.7 Instantiating modus BOCARDO

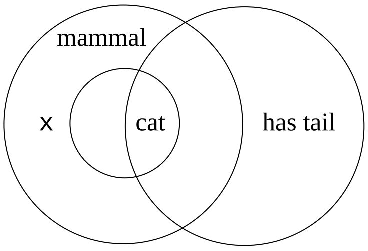
$$\begin{array}{rl}
\text{premise 1:} & \text{some cats have no tail}\\
\text{premise 2:} & \text{all cats are mammals}\\
\hline
\text{conclusion:} & \text{some mammals have no tail}
\end{array}$$

The reductio per impossibile, which helps to prove the validity of BAROCO, is complemented in BOCARDO by ekthesis (Aristotle, An. Pr. I.6, 28b20–21).[10]

The set-theoretic constellation underlying modus BOCARDO in 7.7 may be depicted as follows:

[10] In the middle ages, several jails in England, one specifically in Oxford, were called Bocardo because it was so hard for students to learn how to verify this syllogism.

7.8 Set-theoretic view of modus BOCARDO



Premise 1 is the cat complement of the cat and has_tail intersection. Premise 2 is shown by cat being a subset of mammal. The conclusion is shown as the mammal complement of the mammal and has_tail intersection.

7.9 Rephrasing BOCARDO in DBS

inference: α be_cat implies some α not have_tail.
input: some mammals are cats
output: some mammals have no tail

Consider the translation of 7.8 into the data structure of DBS:

7.10 Applying BOCARDO as a DBS inference

$$\begin{array}{lccccc}
\textit{pattern level} & \begin{bmatrix}\text{noun: }\alpha\\ \text{fnc: be\_cat}\\ \text{prn: K}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_cat}\\ \text{arg: }\alpha\\ \text{prn: K}\end{bmatrix} & \Rightarrow & \begin{bmatrix}\text{noun: }(\alpha\text{ K})\\ \text{sem: pl sel}\\ \text{fnc: has\_tail}\\ \text{prn: K+1}\end{bmatrix} & \begin{bmatrix}\text{verb: has\_tail}\\ \text{sem: }\textit{not}\\ \text{arg: }(\alpha\text{ K})\\ \text{prn: K+1}\end{bmatrix}\\
 & & \Uparrow & & & \Downarrow\\
\textit{content level} & \begin{bmatrix}\text{noun: mammal}\\ \text{cat: pnp}\\ \text{sem: pl sel}\\ \text{fnc: be\_cat}\\ \text{prn: 24}\end{bmatrix} & \begin{bmatrix}\text{verb: be\_cat}\\ \text{arg: mammal}\\ \text{prn: 24}\end{bmatrix} & & \begin{bmatrix}\text{noun: mammal}\\ \text{cat: pnp}\\ \text{sem: pl sel}\\ \text{fnc: has\_tail}\\ \text{prn: 25}\end{bmatrix} & \begin{bmatrix}\text{verb: has\_tail}\\ \text{sem: }\textit{not}\\ \text{arg: cat}\\ \text{prn: 25}\end{bmatrix}
\end{array}$$

The subject in the consequent pattern of the BOCARDO inference is some α, a restriction which is coded by the feature [sem: pl sel], in contradistinction to the subject of the consequent pattern of the FERIO inference (6.9), which is unrestricted and thus compatible with universal, particular, and individual input (compatibility by omission)

8 Combining S- and C-Inferencing

Functional equivalence (CC 1.1, 15.1) at a certain level of abstraction between the human prototype and the artificial agent requires computational cognition to apply S- and C-inferencing in one and the same train of thought. Consider the following derivation of a data-driven countermeasure, which begins with the C-inferences 1.2

and (CC 5.1.4), continues with a lexical S-inference coding a hypernymy (op. cit. 5.2.2), and concludes with another C-inference:

8.1 Mixing S- and C-inference in a train of thought

1 C-inference: α is hungry ⇒ α is cranky (1.2)
⇑ ⇓
input: Laura is hungry output: Laura is cranky

2 C-inference: α is cranky ⇒ α needs food
⇑ ⇓
input:Laura is cranky output: Laura needs food

3 S-inference: Laura eats β ⇒ β ε {apple, banana, cookie,. . . , strawberry}
⇑ ⇓
input:Laura eats food output: Laura eats apple or cookie
or banana, ..., or strawberry ...

4 C-inference: α eats cookie ⇒ α is agreeable again
⇑ ⇓
input:Laura eats cookie output: Laura is agreeable again

The S-inference 3 illustrates a lexical alternative to the syllogisms analyzed in 3–7, namely a hypernymy, which is defined as follows.

8.2 Lexical S-inference implementing hypernymy

[noun: α] ⇒ [noun: β]
If α is animal, then β ε {ape, bear, cat, dog, ...}
If α is food, then β ε {apple, banana, cookie, ..., strawberry}
If α is fuel, then β ε {diesel, gasoline, electricity, hydrogen, ...}
...

The set-theoretic structure of a hypernymy[11] is the relation between a superordinate term and its extension. Accordingly, food is the hypernym of apple, banana, cookie, ..., and strawberry. Set-theoretically, the denotation of food equals the codomain of α. The restrictions on variables are species-, culture-, and even agent-dependent. and may be approximated empirically by means of DBS corpus analysis (RMD[12] corpus).

9 Analogy

Common sense reasoning is based on relations provided by repeated observation and contingent knowledge. For example, there is nothing law-like or set-theoretic in Laura being cranky when hungry. There is another dimension, however, namely

[11] For the corresponding hyponymy see CC 9.1.1.

analogy: a truck not starting caused by a lack of fuel may be seen as analogous to being cranky caused by a lack of food.

9.1 Common sense reasoning based on analogy

```
1  C-inference:   α has no fuel      ⇒              α does not start
                  ⇑                                  ⇓
           input:truck has no fuel        output:  truck does not start

2  C-inference:    α does not start  ⇒           α needs fuel
                   ⇑                                ⇓
           input:truck does not start     output:  truck needs fuel

3  S-inference:   truck gets β        ⇒   β ε {diesel, gasoline, electricity,...}
                            ⇑                                 ⇓
           input:truck gets fuel          output:  truck gets diesel or gasoline
                                                   or electricity or hydrogen...
4  C-inference:    α gets fuel        ⇒           α starts
                   ⇑                                 ⇓
           input:truck gets fuel          output:  truck starts
```

As in 9.1, the C-inference 1 is a general common sense observation, while the C-inference 2 applies to a particular instance. The lexical S-inference 3 is an instance of the hypernymy 8.2, while the C-inference 4 derives the desired result.

Using the data structure of DBS, the data-driven application of the second inference in 9.1 may be shown as follows:

9.2 Applying the C-inference 2 of 9.1

$$\begin{array}{l}\textit{pattern}\\ \textit{level}\end{array} \begin{bmatrix}\text{noun: }\alpha\\ \text{fnc: start}\\ \text{prn: K}\end{bmatrix} \begin{bmatrix}\text{verb: start}\\ \text{sem: }\textit{not}\\ \text{arg: }\alpha\\ \text{prn: K}\end{bmatrix} \Rightarrow \begin{bmatrix}\text{noun: }(\alpha\text{ K})\\ \text{fnc: need}\\ \text{prn: K+1}\end{bmatrix} \begin{bmatrix}\text{verb: need}\\ \text{arg: }(\alpha\text{ K})\text{ fuel}\\ \text{prn: K+1}\end{bmatrix} \quad \begin{bmatrix}\text{noun: fuel}\\ \text{cat: snp}\\ \text{fnc: need}\\ \text{prn: K}\end{bmatrix}$$

where α ε {motor cycle, car , truck, tank,...}

$$\Uparrow \qquad\qquad\qquad \Downarrow$$

$$\begin{array}{l}\textit{content}\\ \textit{level}\end{array} \begin{bmatrix}\text{noun: truck}\\ \text{cat: snp}\\ \text{fnc: start}\\ \text{prn: 24}\end{bmatrix} \begin{bmatrix}\text{verb:start}\\ \text{sem: }\textit{not}\\ \text{arg: truck}\\ \text{prn: 24}\end{bmatrix} \quad \begin{bmatrix}\text{noun: truck}\\ \text{cat: snp}\\ \text{fnc: need}\\ \text{prn: 25}\end{bmatrix} \begin{bmatrix}\text{verb: need}\\ \text{arg: (truck 24) fuel}\\ \text{prn: 25}\end{bmatrix} \begin{bmatrix}\text{noun: fuel}\\ \text{cat: snp}\\ \text{fnc: need}\\ \text{prn: 25}\end{bmatrix}$$

Finding and applying analogical countermeasures may be based in part on a systematic development of semantic fields (CC 11.3.3) across domains.

10 Conclusion

Based on the *categorical judgements* A, E, I, O, there are 256 (2^8) categorical syllogisms of which 24 have been found valid by the medieval scholastics. Using Euler's set-theoretical analysis of the categorical judgments, this paper proposes

[12] Reference-Monitor corpus structured into Domains (CLaTR 15.3).

to reanalyze the valid syllogisms as 12 DBS inferences. Depending on whether or not the antecedent, the consequent, or both are negated, there are 4 classes (2.6, 2.7, 2.8, 2.9) which are divided further by the distinction between a universal, particular, and individual variant.

Classical Barbara, Baroco, Bocardo, Celarent, Darii, Ferio, as well as modi ponendo ponens and tollendo tollens are reanalyzed in detail as DBS inferences. Two of the four classes are completed as (1) Barbara (universal, 5.4), Darii (particular, 6.4), and modus ponendo ponens (individual, 3.3), and (2) Celarent (universal, 5.9), Ferio (particular, 6.9), and modus tollendo tollens (individual, 4.2).

In human thought chains, S-inferences based on set theory and C-inferences based on common sense reasoning may be freely mixed. To support chaining, both kinds of inference use the same inference schema in agent-based DBS but differ in the source of reasoning, i.e., (i) set theoretic structures vs. (ii) something previously observed or learned by the agent.

Bibliography

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

CC = Hausser, R. (2019) *Computational Cognition, Integrated DBS Software Design for Data-Driven Cognitive Processing*, pp. i–xii, 1–237, `lagrammar.net`

CLaTR = Hausser, R. (2011) *Computational Linguistics and Talking Robots; Processing Content in DBS*, pp. 286. Springer (preprint 2nd ed. `lagrammar.net`)

Euler, L. (1761) *Lettres a Une Princesse d'Allemagne*, Vol. 2: Letters No. 102–108. St. Petersburg: Imperial Academy of Sciences, 1768–1772

Frege, G. (1879) *Begriffsschrift. Eine der arithmetischen nachgebildete Formelsprache des reinen Denkens*, Halle: L. Nebert

NLC = Hausser, R. (2006/2017) *A Computational Model of Natural Language Commmunication: Interpretation, Inference, and Production in DBS*, pp. 360. Springer (preprint 2nd ed. (2017) `lagrammar.net`)

Read, S. (2017) "Aristotle's Theory of the Assertoric Syllogism," `https://www.st-andrews.ac.uk/~slr/The_Syllogism.pdf`

Venn, J. (1881) *Symbolic Logic*, London: MacMillan

Zhang Yinsheng and Qiao Xiaodong (2009) "A Formal System of Aristotelian Syllogism Based on Automata Grammar," *Computer Science and Information Engineering*, 2009 WRI World Congress