Computational Pragmatics

Roland Hausser

Universität Erlangen-Nürnberg (em.)

Abstract

In agent-based data-driven Database Semantics (DBS), propositions do not denote truth values but are content. A content is built (i) from the classical semantic kinds *concept, indexical*, and *name*, connected (ii) by the classical semantic relations of *subject/predicate, object\predicate, adnominal|noun, adverbial|verb,* and *conjunct−conjunct*, (iii) coded by addres, (iv) at the elementary, phrasal, and clausal level of grammatical complexity,

Computational pragmatics[1] relies on the on-board orientation system (OBOS), which is part of the agent's interface component and monitors moment by moment as a sequence of STARs. A STAR is a flat feature structure with the attributes S for space (location), T for time, A for agent (speaker), R for recipient (hearer), 3rd for pro3, and prn for proposition number.

A STAR is connected to a content type by a shared prn value, resulting in a content token. A content type with a language-dependent surface is a *literal meaning*$_1$. A corresponding token used for inter-agent language communication is an *utterance meaning*$_2$.

There is literal (3) and nonliteral pragmatics 4, 5. Literal pragmatics is an obligatory change of perspective. Examples are the change from I am thirsty to I was thirsty in the STAR-0 STAR-1 transition of the speak mode and the change from I see you to you see me in the STAR-1 STAR-2 transition of the hear mode.

keywords: language vs. nonlanguage content, content types vs. content tokens, on-board orientation system, STAR, pragmatics of literal vs. nonliteral use

1 Four Kinds of Content in DBS

In DBS, a content is defined as a set (order-free) of proplets. As the computational data structure, proplets are defined as non-recursive feature structures with ordered attributes. The proplets of a proposition are connected by (i) semantic relations of structure coded by address and (ii) a shared prn value.

There are four kinds of content in DBS, called [−surface −STAR][2], [−surface +STAR], [+surface −STAR], and [+surface +STAR], illustrated as follows:

[1] Overviews of noncomputational pragmatics are Kempson (2001) and Horn&Ward eds. (2004).

[2] The STAR in agent-based DBS may be seen as a development of the sign-based "parameter approach," which uses parameters such as I (possible worlds), J (possible times), S (possible speakers), H (possible hearers), etc., such as the index @, I, J, g superscripted to logical formulas in Montague (1973). Cresswell (1972, p.4) wonders tongue in cheek about adding a next drink parameter.

1.1 NONLANGUAGE CONTENT TYPE: [–surface, –STAR]

sur:	sur:	sur:
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
...	...	...
prn: K	prn: K	prn: K

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

The next example is a corresponding nonlanguage token:

1.2 NONLANGUAGE CONTENT TOKEN: [–surface, +STAR]

sur:	sur:	sur:	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:
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 values, the token resulted as an observation by the agent Sylvester on Friday in the yard.

The following example is the language content type corresponding to 1.1. It illustrates the independence of language-dependent sur values, here German, from the relatively language-independent placeholders (English base forms for convenience):

1.3 LANGUAGE CONTENT TYPE (MEANING$_1$): [+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 called a literal meaning$_1$. It is an abstraction insofar as an actual time-linear surface-compositional DBS hear mode derivation, e.g., CC 2.1.3, 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 1.3, called an *utterance meaning*$_2$. Our example is produced by the speaker Sylvester in German towards the intended hearer Tweety and corresponds to the nonlanguage content token 1.2 except for the R value:

1.4 LANGUAGE CONTENT TOKEN (MEANING$_2$): [+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 on Friday in the yard. The content types 1.1 and 1.3 match not only the tokens 1.2 and 1.4, but an open-ended number of corresponding tokens with different prn values. [3]

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.

2 Coactivation Resulting in Resonating Content

The basis of literal and nonliteral pragmatic interpretation is the automatic coactivation of *resonating content* in the agent's on-board database. Resonating content is the computational counterpart of *association* in psychology and relies on (a) the database schema of the agent's A-memory and (b) the data structure of proplets.

2.1 TWO-DIMENSIONAL DATABASE SCHEMA OF A-MEMORY IN DBS

- *horizontal*
  Proplets with the same core value are stored in the same token line in the time-linear order of their arrival.
- *vertical*
  Token lines are in alphabetical order, induced by their core value letter sequence.

The time-linear arrival order of proplets is reflected by the position in their token line and by their prn value. The sequence of (i) *member proplets* is followed by a free slot as part of the column called the (ii) *now front*, and the (iii) *owner*.[4]

[3] The type/token distinction applies not only to propositions, but also to the sign kinds. A concept like square is a type if the length value is a variable, and a token if it is a constant. An indexical like you is a type if a STAR proplet is absent in the proposition and otherwise a token. A name is a type if the named referent value is absent and otherwise a token.

[4] The terminology of member proplets and owner values is reminiscent of the member and owner

2.2 SCHEMATIC EXAMPLE OF TOKEN LINE WITH CLEARED NOW FRONT

(i) member proplets			*(ii) now front*	*(iii) owner*
noun: bone … fnc: bury … prn: 2	noun: bone … fnc: find … prn: 5	noun: bone … fnc: eat … prn: 12		bone

Consider the content 1.2 as stored at the now front before clearance:

2.3 STORAGE OF A PROPOSITION AT THE NOW FRONT

	member proplets		*now front*	*owners*
…	sur: noun: bone cat: snp sem: indef sg fnc: fetch … prn: 2	sur: noun: bone cat: snp sem: def sg fnc: hide … prn: 5	… sur: noun: bone cat: snp sem: indef sg fnc: find … prn: 12	bone
…	sur: noun: dog cat: snp sem: indef sg fnc: bury … prn: 3	sur: noun: dog cat: pnp sem: def pl fnc: find … prn: 12	… sur: noun: dog cat: snp sem: def sg fnc: eat … prn: 15	dog
…	sur: verb: find cat: #n′ #a′ decl sem: pres prog arg: (person x) square … prn: 5	sur: verb: find cat: #n′ #a′ decl sem: past ind arg: cat mouse … prn: 8	… sur: verb: find cat: #n′ #a′ decl sem: past ind arg: dog bone … prn: 12	find
…	S: living room T: monday A: sylvester R: tweety 3rd: Speedy prn: 2	… S: garden T: thursday A: sylvester R: hector 3rd: prn: 10	… S: yard T: friday A: sylvester R: 3rd: prn: 12	sylvester

The storage of a proplet at the now front uses the letter sequence of the core value for accessing the correct token line via the owner. Retrieval searches the token line using the prn value. Content is automatically activated by computing successor proplets based on continuation addresses.

When the proplets at the current now front have ceased to be candidates for additional processing (e.g., cross-copying), the now front is cleared by moving it and

records in a classic network database (Elmasri and Navathe (1989[1]/2017[7]), which inspired the database schema of the DBS A-memory (formerly called word bank).

the owners one step to the right into fresh memory territory (loom-like clearance). By leaving the proplets of the completed proposition, here with the prn value 12, behind, their location becomes permanent storage as member proplets, like sediment, never to be changed. The only way to correct is adding content, like a diary entry. The corrected version of a content is the most recent one, i.e., rightmost in the token lines involved.

In summary, the A-memory (2.3) is content-addressable (Chisvin and Duckworth 1992) because it does not use a separate index, unlike a coordinate-addressable database, e.g., an RDBMS. The key for a proplet's storage in and retrieval from the agent's memory is not a location, but the letter sequence of the proplet's core value (which enables computational string search in combination with a trie structure). Based on reference by address, A-memory automatically monitors changes in kinds of content such as agent's the bank balance or the people in the kitchen.

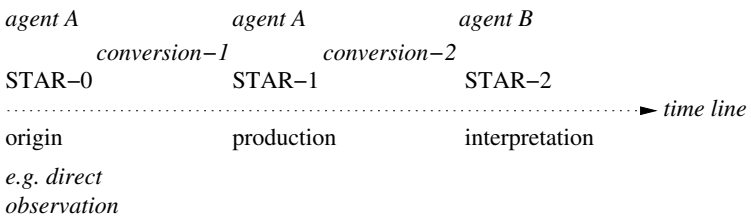
3 Literal Pragmatics of Adjusting Perspective

In agent-based monitoring, a current content recedes inevitably into the past. Therefore the STAR of origin attached to a content now past, called the STAR-0, and the current STAR of the agent's looking back at the past content in memory, called the STAR-1, must be distinguished. For example, when the STAR-0 content I see you is retrieved from memory it must be changed to the STAR-1 content I saw you. The agent's STAR-0 values are constantly updated by monitoring, but preserved in memory as a sequence of STAR-1 contents in their temporal order.

In language communication, the speaker uses the STAR-1 to encode the past STAR-0 content into a language surface. The hearer's interpretation of the speaker's STAR-1 surface, however, necessitates a third STAR, called the STAR-2. For example, the STAR-1 content I saw you must be changed to the STAR-2 content You say me.

In language communication, a content's (i) origin, (ii) production, and (iii) interpretation are also in temporal order:

3.1 Temporal backbone of DBS



To illustrate the speaker's change from a STAR-0 content of origin to a STAR-1 content of production, and the hearer's change from a STAR-1 to a STAR-2 content of interpretation, let us begin with the following example:

3.2 STAR-0 CONTENT OF ORIGIN: I see you.

STAR-0 proplet of origin

sur:	sur:	sur:	S: yard
noun: pro1	verb: see	noun: pro2	T: thursday[5]
cat: s1	cat: #n-s3′ #a′ decl	cat: sp2	A: sylvester
sem: sg	sem: pres ind	sem:	R: hector
fnc: see	arg: pro1 pro2	fnc: see	3rd:
...	...	...	prn: 12
prn: 12	prn: 12	prn: 12	

By definition, a STAR-0 content is (i) without language (quasi language-independent) and (ii) the verb's sem value is pres ind. The pro1 and the pro2 indexicals in the content point at the A value, here sylvester, and the R value, here hector, respectively, of the STAR-0 proplet.

In the speak mode, the agent's STAR-0 content I see you (3.2) may be mapped automatically into one of the following STAR-1 variants. They differ semantically, but are pragmatically equivalent (CC 7):

3.3 PRAGMATICALLY EQUIVALENT STAR-1 CONTENTS

STAR-1 a: Sylvester remembers the content 3.2 without speaking.
STAR-1 b: Sylvester tells Hector that he saw him.
STAR-1 c: Sylvester tells Speedy that he saw "him," referring to Hector.
STAR-1 d: Sylvester tells Speedy that he saw Hector.
STAR-1 e: Sylvester tells Speedy that he saw Hector in the yard.
STAR-1 f: Sylvester tells Speedy that he saw Hector on Thursday in the yard.

In variants a-d, the semantic differences in the contents are compensated pragmatically by varying STAR-1 values. In e and f, past STAR-0 values which were overwritten by the current ones are preserved by writing them into the content.

In the hear mode, each STAR-1 variant in 3.3 must be mapped into an equivalent STAR-2 content (CC 8):

3.4 PRAGMATICALLY EQUIVALENT STAR-2 CONTENTS

STAR-2 a: <For nonlanguage content, a hear mode counterpart does not exist>
STAR-2 b: Hector understands that Sylvester saw him.
STAR-2 c: Speedy understands that Sylvester saw "him," i.e., Hector.
STAR-2 d: Speedy understands that Sylvester saw Hector.
STAR-2 e: Speedy understands that Sylvester saw Hector on Thursday.
STAR-2 f: Speedy understands that Sylvester saw Hector on T. in the yard.

The speaker's STAR-0 STAR-1 conversion-1 and the hearer's STAR-1 STAR-2 conversion-2 use different inferences for the interpretation of pronominal indexicals. The reason is that the interpretation of pronominals in the speaker's conver-

sion-1 is unchanged, but must be inverted in the hearer's conversion-2.

The pragmatic equivalence of the semantically different contents 3.2, 3.3, and 3.4 is based (i) on the choice between coding certain values either in the content as concepts (including the named referent of names), or as (ii) indexicals pointing at the S, T, R, and 3rd values of the STAR. Space and Time information may be coded solely in the STAR, as in variants a-d in 3.3 and b-d in 3.4, or also be written explicitly into the content, as in the variants e and f.

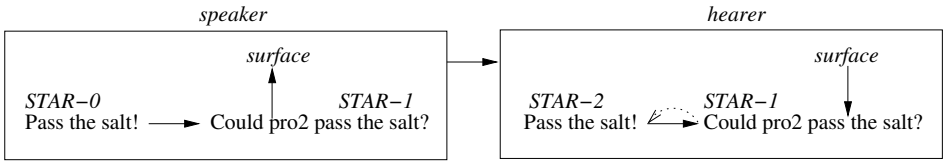
4 Nonliteral Pragmatics of Syntactic Mood Adaptation

The literal pragmatic inferences described in the previous section are obligatory insofar as the STAR-0, STAR-1, and STAR-2 contents must be pragmatically equivalent in order for communication to succeed. Thereby the STAR-0 STAR-1 transitions (speak mode) and the STAR-1 STAR-2 transitions (hear mode) each require their own inference.

We turn now to two kinds of nonliteral pragmatics, called syntactic mood adaptation and figurative use. In contradistinction to literal pragmatics, (i) they are optional and (ii) their STAR-0 STAR-1 (speak mode) and STAR-1 STAR-2 (hear mode) transitions use the same inference, but inductively in the speak mode and abductively in the hear mode. In contradistinction to literal pragmatics, the hearer must revert the speaker's nonliteral content back into the speaker's original content in order for communication to succeed.[6]

Consider the following example from J.L. Austin ([1955]1962) of a syntactic mood adaptation:

4.1 Syntactic mood adaptation based on IMP-INT (CC 7.3.2)



The speaker's communicative purpose of the STAR-0 STAR-1 conversion is softening a command (imperative[7]) into a polite request (yes-no interrogative). If the hearer were to take the speaker's STAR-1 content literally by answering yes or no, communication would fail. For communication to succeed, the hearer must use the same inference as the speaker, but abductively (⟵).

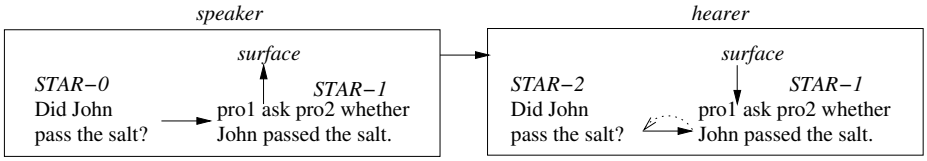
[5] Using weekdays as T values may be crude as compared to nano- or pico-seconds, but is sufficient for current purposes.

[6] Modulo the obligatory hear mode adjustments of a STAR-2 content.

[7] The subject of imperatives in English and many other languages is implicit: it is automatically assumed to be pro2 in the speak mode and pro1 in the hear mode, without any surface manifestation. Consequently, the hearer's standard STAR-2 reversal from pro2 to pro1 is implicit as well.

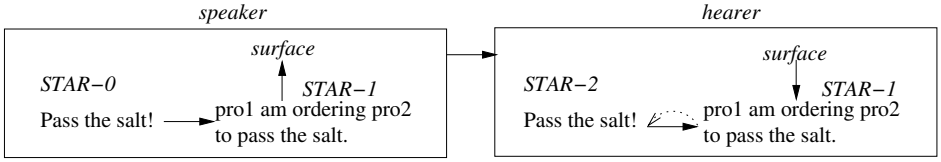
Syntactic mood adaptions are common in the European languages, but not universal.[8] The following two examples are based on the inferences INT-DECL (CC 7.4.2) and IMP-DECL (CC 7.5.2), respectively.

4.2 SYNTACTIC MOOD ADAPTATION INT-DECL (CC 7.4.2)



In this and the next example, the speaker's purpose is emphasis.

4.3 SYNTACTIC MOOD ADAPTATION IMP-DECL



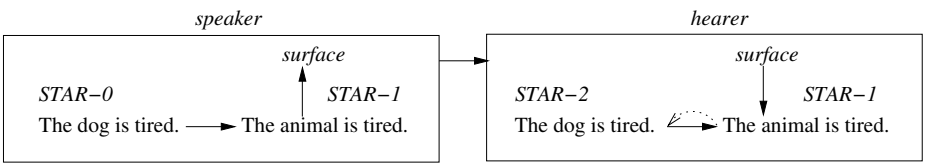
There are more syntactic-mood adaptations for the purpose of emphasis, such as the interrogative-imperative adaptation from Did you pass the salt? to Tell me if you passed the salt![9]

5 Nonliteral Pragmatics of Figurative Use

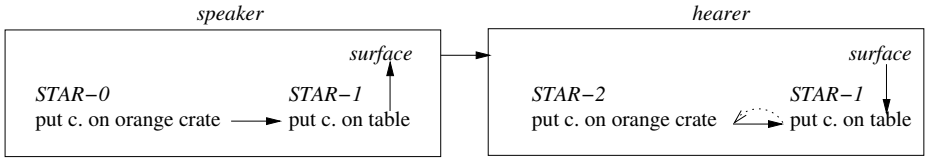
While the inferences of syntactic mood adaptation use complete propositions as input and output, the inferences of figurative use only a part, elementary or phrasal, of a proposition. Figurative use is subject to the invariance constraint (CC 6.4.1), according to which the figurative replacement must be of the same syntactic category and the same semantic field as the literal original. The condition that successful communication requires the hearer to reconstruct the speaker's original content applies equally to syntactic mood adaptation and figurative use.

In figurative use, lexical relations such as hyponymy, metonymy, property sharing, abbreviation, and membership in the same semantic field serve as the basis of inferencing (CC 9). Consider the following examples:

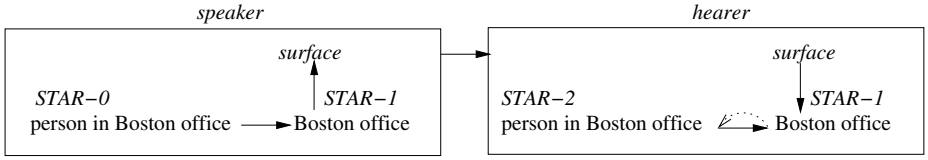
5.1 FIGURATIVE USE BASED ON HYPONYMY (CC SECT. 9.1)



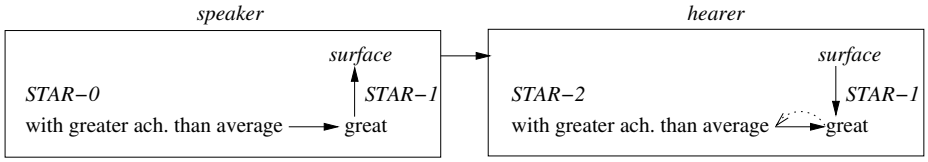
5.2 FIGURATIVE USE BASED ON SHARED PROPERTY (FOCL SECT. 5.2)



5.3 FIGURATIVE USE BASED ON METONYMY (CC SECT. 9.3)



5.4 FIGURATIVE USE BASED AN ABBREVIATING ADVERBIAL (CC 9.4.2)



As in all abductive use, there is no certainty regarding the output of the inference (i.e., of the antecedent). For example, if the orange crate in 5.2 were accompanied by a footstool, a sideboard, and a low bookshelf, it would be impossible for the hearer to decide what the speaker meant with "table" (too many candidates with flat horizontal surfaces in the current context of interpretation, embarrassment of riches). In such a case, the speaker would have to specify more precisely what is meant in order for communication to succeed.

6 Figurative use of 1-, 2-, and 3-place verbs

For figurative uses of 1-, 2-, and 3-place verbs, e.g.

1. The tiger melts into the grass.
2. John stole the show.
3. John gave Mary a headache.

see CC Sect. 9.4.

[8]The alternative in Korean, for example, is the use of two morphological systems, one for honor and one for mood, which are agglutinated to the verbal stem. Thanks to Prof. Kiyong Lee for his help in this matter.

[9]The dependence of nonliteral use pragmatics on specific pronouns, concepts, tense, mood, etc. is reminiscent of Construction Grammar (Fillmore 1988).

Conclusion

In agent-based data-driven DBS, the semantic/pragmatic distinction is based on the type/token distinction from philosophy. The semantics of a content is a type which is independent of the utterance situation, while the pragmatics is a token which connects a type to the agent's on-board orientation system (OBOS).

There are two kinds of pragmatics in DBS. The *obligatory literal* kind adapts a semantic content to the alternative perspectives of speaker and hearer. This requires different inferences for the speak and the hear mode (Sect. 3).

The other kind is *optional nonliteral* pragmatics. It provides informative views such as metaphor on literal content. In successful communication, the hearer must reconstruct the speaker's literal content which the nonliteral view was derived from. For this, speaker and hearer use the same inference, though deductively in the speak mode and abductively in the hear mode (Sect. 4).

Bibliography

Austin, J.L. ([1955]1962) *How to Do Things with Words*, Oxford: Clarendon Press

CC = Hausser, R. (2019) *Computational Cognition; Integrated DBS Software Design for Data-Driven Cognitive Processing*, `lagrammar.net`

Chisvin, L., and R.J. Duckworth (1992) "Content-addressable and associative memory," p. 159–235, in M.C. Yovits (ed.) *Advances in Computer Science, 2nd ed.*, Academic Press

Cresswell, M. (1972) "The world is everything that is the case," *Australasian Journal of Philosophy*, Vol. 50:1–13.

Elmasri, R., and S.B. Navathe (2017) *Fundamentals of Database Systems, 7th ed.* (1st ed. 1989), Redwood City, CA: Benjamin-Cummings

Fillmore, C.J. (1988) "The Mechanisms of Construction Grammar" *Proceedings of the Fourteenth Annual Meeting of the Berkeley Linguistics Society (1988)*, pp. 35–55

Horn, L., and G. Ward (eds.) (2004) *The Handbook of Pragmatics*, Oxford: Blackwell Publishers

Kempson, R. (2001) "Pragmatics: Language and Communication," Chapter 16 in *The Handbook of Linguistics*, M. Aronoff & J. Rees-Miller (eds.), Oxford: Blackwell Publishers, pp. 394–427

Montague, R. (1973) "The Proper Treatment of Quantification in Ordinary English," in J. Hintikka, J. Moravcsik, P. Suppes (eds.) *Approaches to Natural Language*, Dordrecht: Reidel, pp. 221–242

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