Database Semantics and Temporal Inferences
NLPRS’99, Beijing, China

ROLAND HAUSSER
Computational Linguistics
Universität Erlangen Nürnberg
Germany
Overview

1. A cognitive approach to NL communication
2. Semantics and pragmatics
3. A computational approach to NL communication
4. Concatenated propositions: a cognitive approach
5. Concatenated propositions: a computational approach
6. The motor algorithm: left-associative grammar
7. The structure of a Slim machine
8. States of cognition
9. Technical details of a semantic interpretation
10. Pragmatic interpretation
11. The source of coherence in language production
12. Example of a subcontext
13. Autonomous navigation as the basis of conceptualization
14. Different temporal prepositions depending on the navigation
1. A cognitive approach to NL communication

1.1 Different theories of language and their goals

- Behaviorism: maintaining methodological standards.
- Nativism: describing the innate knowledge of the speaker-hearer (PS-grammar).
- Model theory: representing scientific truth (C-grammar).
- SLIM theory: modeling the natural communication mechanism on the computer (LA-grammar).

1.2 Principles of the SLIM theory of language

1. *Surface compositional* (methodological principle)
   Syntactic-semantic composition assembles only concrete word forms, excluding the use of zero-elements, identity mappings, or transformations.

2. *Linear* (empirical principle)
   Interpretation and production of utterances is based on a strictly time-linear derivation order.

3. *Internal* (ontological principle)
   Interpretation and production of utterances is analyzed as cognitive procedures located inside the speaker-hearer.

   Referring with language to past, current, or future objects and events is modeled in terms of pattern matching between language meaning and context.
1.3 An external view of reference

1.4 Internal and external aspects of reference
1.5 Cognitive 2+1 level analysis of reference

- **Surface**:
  - `square`_ 
  - edge 1: $\alpha$ cm
  - angle 1/2: 90°
  - edge 2: $\alpha$ cm
  - angle 2/3: 90°
  - edge 3: $\alpha$ cm
  - angle 3/4: 90°
  - edge 4: $\alpha$ cm
  - angle 4/1: 90°

- **M-concept**:
  - edge 1: 2 cm
  - angle 1/2: 90°
  - edge 2: 2 cm
  - angle 2/3: 90°
  - edge 3: 2 cm
  - angle 3/4: 90°
  - edge 4: 2 cm
  - angle 4/1: 90°

- **I-concept_loc**:

**language level**: 2

**context level**: 1

©1999 Roland Hausser
2. Semantics and pragmatics

2.1 Nonliteral use of the word table: Principle of best match

hearer

\[
\begin{align*}
\text{Put the coffee on the table!} \\
\text{[concept]} \\
\end{align*}
\]

speaker

\[
\begin{align*}
\text{Put the coffee on the table!} \\
\text{[concept]} \\
\end{align*}
\]

orange crate

2.2 Two notions of meaning

- \(\text{meaning}_1\) = property of signs, also called literal meaning
- \(\text{meaning}_2\) = property of utterances, also called speaker meaning

2.3 First principle of pragmatics (PoP-1)

The speaker’s utterance \(\text{meaning}_2\) is the use of the sign’s literal \(\text{meaning}_1\) relative to an internal context.
2.4 Central question of linguistic pragmatics

How does the speaker code the selection and delimitation of the used subcontext into the sign and how can these be correctly inferred by the hearer?

2.5 Postcard example

New York, December 1, 1998

Dear Heather,

Your dog is doing fine. The weather is very cold. In the morning he played in the snow. Then he ate a bone. Right now I am sitting in the kitchen. Fido is here, too. The fuzzball hissed at him again. We miss you.

Love,
Spencer

2.6 Parameters of origin of signs (STAR-point)

1. S = the Spatial place of origin
2. T = the Temporal moment of origin
3. A = the Author
4. R = the intended Recipient.
2.7 Second principle of pragmatics (PoP-2)

The STAR-point of the sign determines its primary positioning in the database by specifying the *entry context* of interpretation.

2.8 Primary positioning in terms of the STAR-point

Heather’s cognitive representation:

```
<table>
<thead>
<tr>
<th>ST-POINT</th>
</tr>
</thead>
<tbody>
<tr>
<td>language level: Text of the postcard</td>
</tr>
<tr>
<td>context level: sitting in New Zealand on the beach</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>INTERPRETATION CONTEXT</th>
</tr>
</thead>
<tbody>
<tr>
<td>language level: Text of the postcard</td>
</tr>
</tbody>
</table>
```
3. A computational approach to NL communication

3.1 Interaction with a conventional database

3.2 Interaction between speaker and hearer
### 3.3 DB interaction and NL communication

- **ENTITIES INVOLVED**
  - *Database interaction*: takes place between two different entities, the user and the database.
  - *NL communication*: takes place between two similar and equal cognitive agents, the speaker and the hearer.

- **ORIGIN OF CONTROL**
  - *Database interaction*: operations of input and output are controlled by the user.
  - *NL communication*: there is no user. Instead, the cognitive agents control each other by alternating in the speaker- and the hearer-mode (*turn taking*).

- **METHOD OF CONTROL**
  - *Database interaction*: user controls the operations of the database with a programming language the commands of which are executed as electronic procedures.
  - *NL communication*: speaker controls language production as an autonomous agent, coding the parameters of the utterance situation into the output expressions. The hearer’s interpretation is controlled by the incoming language expression.

- **TEMPORAL ORDER**
  - *Database interaction*: output (database as ‘speaker’) occurs necessarily after the input (database as ‘hearer’).
  - *NL communication*: production (output procedure of the speaker) occurs necessarily before interpretation (input procedure of the hearer).
3.4 Sketch of a simple subcontext

```
 FIDO
   /\  
  IS-A FRIENDS BROTHERS
     /\      /
    DOG  Felix Fritz Zach Eddie
```

3.5 Adding the content of *Fido likes Zach* to 3.4

```
 LIKE
 /\  
 AGENT PATIENT
   .     .
    FIDO .
          |
         /\  
        IS-A FRIENDS BROTHERS
           /\      /
          DOG  Felix Fritz Eddie Zach
```

©1999 Roland Hausser
3.6 Schema of language interpretation (analysis)

```
surfaces  w1 ➔ w2 ➔ w3 ➔ w4 ➔ [control]
language level:
meaning_1
context level:  I-concepts_{loc}  □ ➔ □ ➔ □ ➔ □ ➔ □ ➔ □
```

3.7 Schema of language production (generation)

```
surfaces  w1 ➔ w2 ➔ w3 ➔ w4 ➔
language level:
meaning_1  □ □ □ □ □
context level:  I-concepts_{loc}  □ ➔ □ ➔ □ ➔ □ ➔ □ ➔ [control]
```
4. Concatenated propositions: a cognitive approach

4.1 The three elements of basic propositions

<table>
<thead>
<tr>
<th>Logic</th>
<th>World</th>
<th>Language</th>
</tr>
</thead>
<tbody>
<tr>
<td>functor</td>
<td>relation</td>
<td>verb</td>
</tr>
<tr>
<td>argument</td>
<td>object</td>
<td>noun</td>
</tr>
<tr>
<td>modifier</td>
<td>property</td>
<td>adjective-adverbial</td>
</tr>
</tbody>
</table>

4.2 An example of two contextual propositions

```
epr: and

[ l: contains ]
[loc: Mo 14:05]

[ l: field ]
[loc: A2]

[ l: triangle ]
[loc: A2]

[ l: field ]
[loc: A2]

[ l: square ]
[loc: A2]
```
5. Concatenated propositions: a computational approach

5.1 Propositions 4.2 as a word bank

<table>
<thead>
<tr>
<th>TYPES</th>
<th>SIMPLIFIED PROPLETS</th>
</tr>
</thead>
</table>
| [M-concept: contain]  | [I-concept_{loc}: x2]  
argument 1: field  
argument 2: triangle  
prn: 23  
epr: 23 and 24 |
| role: functor      |                                                                                   |
| [M-concept: field]   | [I-concept_{loc}: x5]  
argument 1: field  
argument 2: square  
prn: 24  
epr: 23 and 24 |
| role: argument      |                                                                                   |
| [M-concept: square]   | [I-concept_{loc}: x4]  
functor: contain  
prn: 24  
id: 7 |
| role: argument      |                                                                                   |
| [M-concept: triangle]  | [I-concept_{loc}: x3]  
functor: contain  
prn: 23  
id: 8 |

©1999 Roland Hausser
5.2 Types of continuations

*intrapositional*:
from argument to functor, functor to argument, from modifier to modified and vice versa

*extrapositional*:
epr from verb to verb, id from noun to noun

5.3 Types of databases

classic: record based
non-classic: based on the principle of slot and filler

5.4 Types of classic databases

Relational database, hierarchical database, network database
5.5 Example of a network database

owner records  member records
Comp.Sci.      Riedle  Schmidt  Stoll  ...
Mathematics    Müller  Barth    Jacobs ...
Physics       Weber    Meier    Miele  ...

5.6 Relations between proplet features

type ↔ token
token ↔ prn
prn ↔ epr
token ↔ id
functor ↔ argument
modifier ↔ modified
5.7 Propositional presentation of subcontext 3.5

1. Fido is a dog.
2. Fido has friends.
3. The friends are Zach and Eddie.
4. Fido has brothers.
5. The brothers are Felix and Fritz.
6. Fido likes Zach.
5.8 Graphical presentation of the propositions in 5.7

```
be
Fido  dog
    have
Fido  friend
        be
            friend  Zach, Eddie
                have
Fido  brother
        be
            brother  Felix, Fritz
                like
Fido  Zach
```
## 5.9 Subcontext 5.8 as a word bank

<table>
<thead>
<tr>
<th>Types</th>
<th>Proplets</th>
</tr>
</thead>
</table>
| [M-concept: be] | [I-concept<sub>loc</sub>: x1  
| role: functor   | arg1: Fido  
|                | arg2: dog  
|                | prn: 1  
|                | epr: 1 and 2] |
| [M-concept: brother] | [I-concept<sub>loc</sub>: x2  
| role: argument  | arg1: friend  
|                | arg2: Zach, Eddie  
|                | prn: 3  
|                | epr: 2 and 3  
|                | 3 and 4] |
| [M-concept: dog] | [I-concept<sub>loc</sub>: x3  
| role: argument  | arg1: brother  
|                | arg2: Felix, Fritz  
|                | prn: 5  
|                | epr: 4 and 5  
|                | 5 and 6] |
| [M-concept: Eddie] | [I-concept<sub>loc</sub>: x4  
| role: argument  | arg1: friend  
|                | arg2: Zach, Eddie  
|                | prn: 3  
|                | epr: 2 and 3  
|                | 3 and 4] |
| [M-concept: dog] | [I-concept<sub>loc</sub>: x5  
| role: argument  | arg1: brother  
|                | arg2: Felix, Fritz  
|                | prn: 5  
|                | epr: 4 and 5  
|                | 5 and 6] |
| [M-concept: Eddie] | [I-concept<sub>loc</sub>: x6  
| role: argument  | arg1: friend  
|                | arg2: Zach, Eddie  
|                | prn: 3  
|                | epr: 2 and 3  
|                | 3 and 4] |
| [M-concept: Eddie] | [I-concept<sub>loc</sub>: x7  
| role: argument  | arg1: friend  
|                | arg2: Zach, Eddie  
|                | prn: 3  
|                | epr: 2 and 3  
|                | 3 and 4] |
[M-concept: Felix]
role: argument
[functor: be]
prn: 5
id: 4

[M-concept: Fritz]
role: argument
[functor: be]
prn: 5
id: 5

[M-concept: Fido]
role: argument
[functor: be]
prn: 1
id: 1
[I-concept]: x10
[I-concept]: x11
[I-concept]: x12
[I-concept]: x13
[I-concept]: x14
[I-concept]: x15
[I-concept]: x16
[I-concept]: x17
prn: 2
id: 1
prn: 2
id: 1
prn: 4
id: 1
prn: 6
id: 1
prn: 3
id: 1
prn: 2
id: 1
prn: 4
id: 1
prn: 2
id: 1
prn: 4
id: 1
prn: 2
id: 1
prn: 4
id: 1
prn: 2
id: 1
prn: 4
id: 1

[M-concept: friend]
role: argument
[functor: have]
prn: 2
id:

[M-concept: have]
role: functor
arg1: Fido
arg2: friend
prn: 2
epr: 1 and 2
2 and 3

[M-concept: have]
role: functor
arg1: Fido
arg2: brother
prn: 4
epr: 3 and 4
4 and 5

©1999 Roland Hausser
M-concept: like
role: functor

I-concept\textsubscript{loc}: x18
arg1: Fido
arg2: Zach
prn: 6
epr: 5 and 6

M-concept: Zach
role: argument

I-concept\textsubscript{loc}: x19
functor: be
prn: 3
id: 2

I-concept\textsubscript{loc}: x20
functor: like
prn: 6
id: 2

&
### 5.10 Semantic representation of proposition 6

<table>
<thead>
<tr>
<th>TYPES</th>
<th>PROPLETS</th>
</tr>
</thead>
<tbody>
<tr>
<td>M-concept: Fido</td>
<td>I-concept&lt;sub&gt;loc&lt;/sub&gt;: x13</td>
</tr>
<tr>
<td>role: argument</td>
<td>functor: like</td>
</tr>
<tr>
<td></td>
<td>prn: 6</td>
</tr>
<tr>
<td></td>
<td>id: ?</td>
</tr>
<tr>
<td>M-concept: like</td>
<td>I-concept&lt;sub&gt;loc&lt;/sub&gt;: x18</td>
</tr>
<tr>
<td>role: functor</td>
<td>arg1: Fido</td>
</tr>
<tr>
<td></td>
<td>arg2: Zach</td>
</tr>
<tr>
<td></td>
<td>prn: 6</td>
</tr>
<tr>
<td></td>
<td>epr: ?</td>
</tr>
<tr>
<td>M-concept: Zach</td>
<td>I-concept&lt;sub&gt;loc&lt;/sub&gt;: x20</td>
</tr>
<tr>
<td>role: argument</td>
<td>functor: like</td>
</tr>
<tr>
<td></td>
<td>prn: 6</td>
</tr>
<tr>
<td></td>
<td>id: ?</td>
</tr>
</tbody>
</table>
6. The structure of the SLIM machine

6.1 Static structures of the SLIM machine
6.2 External connections and motor algorithms of the SLIM machine
7. The motor algorithm: left-associative grammar

7.1 The principle of possible continuations

Beginning with the first word of the sentence, the grammar describes the possible continuations for each sentence start by specifying the rules which may perform the next grammatical composition (i.e., add the next word).

7.2 Schema of left-associative rule in LA-grammar

\[ r_i: \text{cat}_1 \text{ cat}_2 \Rightarrow \text{cat}_3 \text{ rp}_i \]

7.3 Schema of a canceling rule in C-grammar

\[ \alpha(Y|X) \odot \beta(Y) \Rightarrow \alpha \beta(X) \]

7.4 Schema of a rewrite rule in PS-grammar

\[ A \rightarrow B \ C \]
7.5 Three conceptual derivation orders

LA-grammar

C-grammar

PS-grammar

*bot.-up left-associative*  
*bottom-up amalgamating*  
*top-down expanding*
7.6 LA-grammar for $a^k b^k c^k$

$LX =_{def} \{ [a (a)], [b (b)], [c (c)] \}$

$ST_S =_{def} \{ [(a) \{ r_1, r_2 \}] \}$

$r_1 : (X) (a) \Rightarrow (aX) \{ r_1, r_2 \}$

$r_2 : (aX) (b) \Rightarrow (Xb) \{ r_2, r_3 \}$

$r_3 : (bX) (c) \Rightarrow (X) \{ r_3 \}$

$ST_F =_{def} \{ \epsilon \text{ rp}_3 \}$. 

7.7 The finite state backbone of the LA-grammar for $a^k b^k c^k$
7.8 Comparing constituent structure and time-linear analysis

Mary gives Fido a bone
(V)

Mary gives Fido a bone
(S3 V)

Mary gives Fido a bone
(S3 A V)

gives Fido a bone
(SN V)

gives Fido a bone
(S3 D A V)

Mary gives Fido a bone
(SNP)

Mary gives Fido a bone
(SN SNP)

Mary gives Fido a bone
(SN)

Mary (SNP)

Mary (S3 D A V)

Mary (SNP)

Mary (SNP)

Mary (S3 D A V)

Mary (SNP)

Fido (SNP)

Fido (SN SNP)

Fido (SN)

a bone (SN)

a bone (SN)

a bone (SN)

a bone (SN)

a bone (SN)

a bone (SN)

a bone (SN)

a bone (SN)

a bone (SN)

a bone (SN)

a bone (SN)

a bone (SN)
8. States of cognition

8.1 SLIM 1: Recognition (contextual)

8.2 SLIM 2: Action (contextual)
8.3 SLIM 3: Inference (contextual)

8.4 SLIM 4: Interpretation of language (mediated reference)
8.5 **SLIM 5: Production of language** (mediated reference)

8.6 **SLIM 6: Language-controlled action** (immediate reference)
8.7 SLIM 7: Commented recognition (immediate reference)

8.8 SLIM 8: Language-controlled recognition (immediate reference)
8.9 SLIM 9: Commented action (immediate reference)

8.10 SLIM 10: Cognitive stillstand
9. Technical details of a semantic interpretation

9.1 Syntactico-semantic analysis of field contains triangle

**Combination step 1:**

**Syntax:**

\[
\text{field contains} \quad \text{(N)} \quad + \quad \text{contains} \quad \text{(N A V)} \quad \Rightarrow \quad \text{field contains} \quad \text{(A V)}
\]

**Semantics:**

\[
\begin{array}{c}
\text{np: field} \\
\text{prn: 23} \\
\text{FUNC: contain} \\
\text{ARG: prn: 23}
\end{array} \quad \Rightarrow \quad \begin{array}{c}
\text{np: field} \\
\text{prn: 23} \\
\text{FUNC: contain} \\
\text{ARG: prn: 23}
\end{array}
\]

**Combination step 2:**

**Syntax:**

\[
\text{field contains} \quad \text{(A V)} \quad + \quad \text{triangle} \quad \text{(A)}
\]

**Semantics:**

\[
\begin{array}{c}
\text{np: field} \\
\text{prn: 23} \\
\text{FUNC: contain} \\
\text{ARG: prn: 23}
\end{array} \quad \Rightarrow \quad \begin{array}{c}
\text{np: field} \\
\text{prn: 23} \\
\text{FUNC: contain} \\
\text{ARG: field triangle} \\
\text{prn: 23}
\end{array}
\]

**Syntax:**

\[
\Rightarrow \quad \begin{array}{c}
\text{np: field} \\
\text{prn: 23} \\
\text{FUNC: contain} \\
\text{ARG: field triangle} \\
\text{prn: 23}
\end{array} \quad \Rightarrow \quad \begin{array}{c}
\text{np: triangle} \\
\text{FUNC: contain} \\
\text{ARG: field triangle} \\
\text{prn: 23}
\end{array}
\]

©1999 Roland Hausser
9.2 The man gave Mary a flower because he loves her.

9.3 Applying DET+N to the + man

\[ \text{syn: } \langle n' \ x \rangle \quad \langle n \rangle \quad \Rightarrow \quad \langle x \rangle \]

\[ \text{nw.M } \rightarrow \text{ ss.} \]

\[ \text{copy}_{ss} \]

\[ \text{sur: the} \]
\[ \text{syn: } \langle \text{SN' SNP} \rangle \]
\[ \begin{align*}
\text{P: } & \langle \text{sg def} \rangle \\
\text{C: } & \left[ \text{MOD: } \right] \\
\text{I: } & \left[ \text{prn: } \langle 1 \rangle \right] \\
\text{M: } & \left[ \text{id: +1} \right]
\end{align*} \]
\[ \text{sem: } \]

\[ \text{sur: man} \]
\[ \text{syn: } \langle \text{SN} \rangle \]
\[ \begin{align*}
\text{P: } & \langle \text{sg def} \rangle \\
\text{C: } & \left[ \text{MOD: } \right] \\
\text{I: } & \left[ \text{prn: =} \right] \\
\text{M: } & \left[ \text{id: = man} \right]
\end{align*} \]
\[ \text{sem: } \]

\[ \text{sur: the man} \]
\[ \text{syn: } * \langle \text{SNP} \rangle \]
\[ \begin{align*}
\text{P: } & \langle \text{sg def} \rangle \\
\text{C: } & \left[ \text{MOD: } \right] \\
\text{I: } & \left[ \text{prn: } \langle 1 \rangle \right] \\
\text{M: } & \left[ \text{id: } \ast \ 1 \ 	ext{man} \right]
\end{align*} \]
\[ \text{sem: } \]
9.4 SLIM semantic representation of example 9.2

the man

<table>
<thead>
<tr>
<th>sur:</th>
<th>syn: ⟨SNP⟩</th>
</tr>
</thead>
<tbody>
<tr>
<td>P: ⟨sg def⟩</td>
<td></td>
</tr>
<tr>
<td>C: [MOD: ]</td>
<td></td>
</tr>
<tr>
<td>VERB: give</td>
<td></td>
</tr>
<tr>
<td>I: [prn: ⟨1⟩]</td>
<td></td>
</tr>
<tr>
<td>id: 1</td>
<td></td>
</tr>
<tr>
<td>M: man</td>
<td></td>
</tr>
</tbody>
</table>

Mary

<table>
<thead>
<tr>
<th>sur:</th>
<th>syn: ⟨SNP⟩</th>
</tr>
</thead>
<tbody>
<tr>
<td>P: ⟨sg name⟩</td>
<td></td>
</tr>
<tr>
<td>C: [MOD: ]</td>
<td></td>
</tr>
<tr>
<td>VERB: give</td>
<td></td>
</tr>
<tr>
<td>I: [prn: ⟨1⟩]</td>
<td></td>
</tr>
<tr>
<td>id: 2</td>
<td></td>
</tr>
<tr>
<td>M: Mary</td>
<td></td>
</tr>
</tbody>
</table>

gave

<table>
<thead>
<tr>
<th>sur:</th>
<th>syn: ⟨V⟩</th>
</tr>
</thead>
<tbody>
<tr>
<td>P: ⟨past tense⟩</td>
<td></td>
</tr>
<tr>
<td>C: [MOD: ]</td>
<td></td>
</tr>
<tr>
<td>NP: ⟨man, Mary, flower⟩</td>
<td></td>
</tr>
<tr>
<td>I: [prn: ⟨1⟩]</td>
<td></td>
</tr>
<tr>
<td>epr: 1 bec 2</td>
<td></td>
</tr>
<tr>
<td>M: give</td>
<td></td>
</tr>
</tbody>
</table>

a flower

<table>
<thead>
<tr>
<th>sur:</th>
<th>syn: ⟨SNP⟩</th>
</tr>
</thead>
<tbody>
<tr>
<td>P: ⟨sg indef⟩</td>
<td></td>
</tr>
<tr>
<td>C: [MOD: ]</td>
<td></td>
</tr>
<tr>
<td>VERB: give</td>
<td></td>
</tr>
<tr>
<td>I: [prn: ⟨1⟩]</td>
<td></td>
</tr>
<tr>
<td>id: 3</td>
<td></td>
</tr>
<tr>
<td>M: flower</td>
<td></td>
</tr>
</tbody>
</table>
9.5 Components of literal meaning (meaning$_1$)

- Compositional semantics (sentence semantics)
  1. Decomposition of input into elementary propositions.
  2. Functor-argument structure within an elementary proposition.
  3. Extrapropositional relations among elementary propositions.

- Lexical semantics (word semantics)
  1. Properties and M-concepts of woplets.
  2. Extrapropositional relations between word types by means of absolute propositions.
10. Pragmatic interpretation

10.1 Embedding 9.4 into the contextual word bank

language-based word bank

contextual word bank

base forms (owner records)
linguistic tokens (member records)

reading in example 9.4

4 a flower
2 gave
6 he
5 because loves
1 man
7 she
3 Mary

pragmatic interpretation
10.2 Time-linear interpretation of language controlling a navigation through the context
11. The source of coherence in language production

11.1 Immediate vs. mediated subcontexts

In immediate subcontexts, the coherence of the content follows directly from the coherence of the external world which they reflect, i.e., the temporal and spatial sequence of events, the part-whole relations of objects, etc. In contrast, mediated subcontexts have the special property that the elements familiar from direct recognition may be reordered and reconnected by the author at will.

11.2 Comparing coherence and incoherence, Example I

The representation of a swimmer standing at the pool side, diving into the water, and disappearing with a splash is coherent. In contrast, a representation in which a pair of feet appears in the foaming water and a swimmer flies feet first into the air landing on the pool side, would be incoherent – unless it is specified in addition that the representation happens to be, e.g., a backward running movie.

11.3 Comparing coherence and incoherence, Example II

A representation of people talking with each other would be coherent. In contrast, a similar representation of a deer conversing with a skunk in English would be incoherent – unless it is specified in addition that the representation happens to be fictional.
12. Example of a subcontext

12.1 Mediated subcontexts reflecting the coherence of the external world

world → speaker context → language → hearer context → world

12.2 A sequence of propositions forming a subcontext

1. Peter leaves the house. 2. Peter crosses the street. 3. Peter enters a restaurant. 4. Peter orders a salad. 5. Peter eats the salad. 6. Peter pays the salad. 7. Peter leaves the restaurant. 8. Peter crosses the street. 9. Peter enters the house.

12.3 Representing 12.2 as a word bank

CONCEPT TYPES:  

COPLETS:

\[
\begin{align*}
\text{[M-concept: cross]
\text{[role: T-verb]}} & \quad \left[ \text{I-concept}_{\text{loc}}: \text{cross} \right. \\
\text{P: indicative} & \quad \left. \begin{array}{c}
\text{C: [MOD:]}
\text{NP: Peter, street}
\end{array} \right]
\quad \left[ \text{prn: 2} \right. \\
\text{I: [epr: \begin{array}{c}2 \text{ then } 3 \\
1 \text{ then } 2 \end{array}}
\quad \left. \begin{array}{c}
\text{I: [epr: } 8 \text{ then } 9 \\
\end{array} \right]
\end{align*}
\]
Example of a subcontext

[Example subcontext]

[M-concept: eat
role: T-verb]

[I-concept_{loc}: eat
P: indicative
C: [MOD:
NP: Peter, salad
prn: 5
I: [5 then 6]
prn: 5
I: [5 then 6]
prn: 5
I: [5 then 6]
prn: 5
I: [5 then 6]]]

[M-concept: enter
role: T-verb]

[I-concept_{loc}: enter
P: indicative
C: [MOD:
NP: Peter, restaurant
prn: 3
I: [3 then 4]
prn: 3
I: [3 then 4]
prn: 3
I: [3 then 4]
prn: 3
I: [3 then 4]]]

[M-concept: house
role: noun]

[I-concept_{loc}: house
P: A sg def
C: [MOD:
VERB: leave
prn: 1
I: [1]
id: 2]]

[I-concept_{loc}: house
P: A sg def
C: [MOD:
VERB: enter
prn: 9
I: [9]
id: 2]]

©1999 Roland Hausser
[M-concept: leave]  
role: T-verb

[I-concept$_{loc}$: leave]  
P: indicative  
C: [MOD: NP: Peter, house]  
prn: 1  
epr: [1 then 2]  
I: 

[I-concept$_{loc}$: leave]  
P: indicative  
C: [MOD: NP: Peter, restaurant]  
prn: 7  
epr: [7 then 8]  
I: [6 then 7]

[M-concept: order]  
role: T-verb

[I-concept$_{loc}$: order]  
P: indicative  
C: [MOD: NP: Peter, salad]  
prn: 4  
epr: [4 then 5]  
I: [3 then 4]

[M-concept: pay]  
role: T-verb

[I-concept$_{loc}$: pay]  
P: indicative  
C: [MOD: NP: Peter, salad]  
prn: 6  
epr: [5 then 6]

©1999 Roland Hausser
12. Example of a subcontext

[M-concept: Peter]
role: name

M-concept: Peter
P: Nom
C: [MOD: 
VERB: leave]
I: [prn: 1 ]
id: 1 ]

I-concept<sub>loc</sub>: Peter
P: Nom
C: [MOD: 
VERB: leave]
I: [prn: 2]
id: 1 ]

I-concept<sub>loc</sub>: Peter
P: Nom
C: [MOD: 
VERB: enter]
I: [prn: 3]
id: 1 ]

I-concept<sub>loc</sub>: Peter
P: Nom
C: [MOD: 
VERB: order]
I: [prn: 4]
id: 1 ]

I-concept<sub>loc</sub>: Peter
P: Nom
C: [MOD: 
VERB: eat]
I: [prn: 5]
id: 1 ]

I-concept<sub>loc</sub>: Peter
P: Nom
C: [MOD: 
VERB: pay]
I: [prn: 6]
id: 1 ]

I-concept<sub>loc</sub>: Peter
P: Nom
C: [MOD: 
VERB: leave]
I: [prn: 7]
id: 1 ]

I-concept<sub>loc</sub>: Peter
P: Nom
C: [MOD: 
VERB: cross]
I: [prn: 8]
id: 1 ]

I-concept<sub>loc</sub>: Peter
P: Nom
C: [MOD: 
VERB: enter]
I: [prn: 9]
id: 1 ]
Example of a subcontext

[M-concept: restaurant
role: noun

I-concept\textsubscript{loc}: restaurant
P: A sg indef
C: [MOD:
   VERB: enter]
I: [prn: 3]
   [id: 4]

I-concept\textsubscript{loc}: restaurant
P: A sg def
C: [MOD:
   VERB: leave]
I: [prn: 7]
   [id: 4]

[M-concept: salad
role: noun

I-concept\textsubscript{loc}: salad
P: A sg indef
C: [MOD:
   VERB: order]
I: [prn: 4]
   [id: 5]

I-concept\textsubscript{loc}: salad
P: A sg def
C: [MOD:
   VERB: eat]
I: [prn: 5]
   [id: 5]

I-concept\textsubscript{loc}: salad
P: A sg def
C: [MOD:
   VERB: pay]
I: [prn: 6]
   [id: 5]

[M-concept: street
role: noun

I-concept\textsubscript{loc}: street
P: A sg def
C: [MOD:
   VERB: cross]
I: [prn: 2]
   [id: 3]

I-concept\textsubscript{loc}: street
P: A sg def
C: [MOD:
   VERB: cross]
I: [prn: 8]
   [id: 3]
13. Autonomous navigation as the basis of conceptualization

13.1 Step 1 of a LA-NA rule application

\[
\begin{align*}
\text{START} & \quad \text{NEXT} & \quad \text{NEW START} \\
\begin{bmatrix} m1: a \\ M2: b \\ prn: c \end{bmatrix} & \quad \begin{bmatrix} m2: b \\ M1: x a y \\ prn: c \end{bmatrix} & \rightarrow \quad \begin{bmatrix} m2: b \end{bmatrix}
\end{align*}
\]

rule package \(1 + 2 \rightarrow 2\)

coplets of the word bank

\[
\begin{bmatrix} \ldots \\ m1: c1 \\ \ldots \\ M2: c2 \\ \ldots \\ prn: c3 \\ \ldots \end{bmatrix}_1
\]
13.2 Step 2 of an LA-NA rule application

\[
\text{rule}_{1+2\Rightarrow2}:
\begin{align*}
\text{START} & : \\
\text{m1: } & a \\
\text{M2: } & b \\
\text{prn: } & c \\
\text{NEXT} & : \\
\text{m2: } & b \\
\text{M1: } & x a y \\
\text{prn: } & c \\
\text{NEW START} & : \\
\text{m2: } & b \\
\end{align*}
\]

\[\Rightarrow \text{rule package}_{1+2\Rightarrow2}\]

\[
\text{coplets} \quad \text{of the} \quad \text{word bank}
\]

\[
\begin{align*}
\cdots & \\
\text{m1: } & c1 \\
\cdots & \\
\text{M2: } & c2 \\
\cdots & \\
\text{prn: } & c3 \\
\cdots & 1 \\
\end{align*} + \begin{align*}
\cdots & \\
\text{m2: } & c2 \\
\cdots & \\
\text{M1: } & ..c1.. \\
\cdots & \\
\text{prn: } & c3 \\
\cdots & 2 \\
\end{align*}
\]
13.3 Step 3 of a LA-NA rule application

<table>
<thead>
<tr>
<th>START</th>
<th>NEXT</th>
<th>NEW START</th>
</tr>
</thead>
<tbody>
<tr>
<td>[m1: a]</td>
<td>[m2: b]</td>
<td>[m2: b]</td>
</tr>
<tr>
<td>[M2: b]</td>
<td>[M1: x a y]</td>
<td>[\ldots]</td>
</tr>
<tr>
<td>[prn: c]</td>
<td>[prn: c]</td>
<td>[prn: c]</td>
</tr>
</tbody>
</table>

**rule\(_{1+2\Rightarrow2}\):**

\[
\begin{align*}
\begin{bmatrix}
m1: c1 \\
m2: c2 \\
prn: c3 \\
\end{bmatrix} & \quad + \quad \\
\begin{bmatrix}
\ldots \\
\ldots \\
\ldots \\
M2: c2 \\
prn: c3 \\
\end{bmatrix} & \quad \Rightarrow \quad \\
\begin{bmatrix}
\ldots \\
\ldots \\
\ldots \\
M1: ..c1.. \\
prn: c3 \\
\end{bmatrix}
\end{align*}
\]

**Rule package\(_{1+2\Rightarrow2}\):**

\[
\begin{align*}
\begin{bmatrix}
m1: \ldots \\
M2: \ldots \\
prn: \ldots \\
\end{bmatrix} & \quad \Rightarrow \quad \\
\begin{bmatrix}
m2: \ldots \\
M1: \ldots \ldots \\
prn: \ldots \\
\end{bmatrix}
\end{align*}
\]
13.4 Tracking principles of LA-navigation

1. *Completeness*
   Within an elementary proposition those coplets are preferred which have not yet been traversed during the current navigation.

2. *Uniqueness*
   If several START or NEXT coplets are available, no more than one of each are selected whereby the choice may be at random or – if activated – based on a specific navigation pattern.

3. *Recency*
   In extrapropositional navigations, propositions which have been least recently traversed are preferred.

4. *Frequency*
   When entering a new subcontext, the navigation prefers paths most frequently traversed in previous navigations.
13.5 Definition of universal LA-NA syntax

\[ \text{ST}_S: \{ ([M-np: a] \{ 1 \text{ V+NP1}, 2 \text{ V+NP2} \}) \} \]

\[ \text{V+NP1:} \begin{cases} \text{M-verb: a} \\ \text{NP: x by} \\ \text{prn: m} \end{cases} \rightarrow \begin{cases} \text{M-verb: a} \\ \text{NP: x by} \\ \text{prn: m} \end{cases} \{ 3 \text{ V+NP1}, 4 \text{ V+NP2}, 5 \text{ V+epr} \} \]

\[ \text{V+NP2:} \begin{cases} \text{M-verb: a} \\ \text{NP: x by} \\ \text{prn: m} \end{cases} \rightarrow \begin{cases} \text{M-verb: a} \\ \text{NP: x by} \\ \text{prn: m} \end{cases} \{ 6 \text{ NP+id} \} \]

\[ \text{V+epr:} \begin{cases} \text{M-verb: a} \\ \text{NP: x} \\ \text{prn: m} \end{cases} \rightarrow \begin{cases} \text{M-verb: b} \\ \text{NP: y} \\ \text{prn: n} \end{cases} \{ 7 \text{ V+NP1}, 8 \text{ V+NP2} \} \]

\[ \text{NP+id:} \begin{cases} \text{M-np: a} \\ \text{VERB: b} \\ \text{id: m} \end{cases} \rightarrow \begin{cases} \text{M-np: a} \\ \text{VERB: c} \\ \text{id: m} \end{cases} \{ 9 \text{ V+NP1 10 V+NP2} \} \]

\[ \text{ST}_F: \{ ([\text{M-verb: x}] \text{ RP V+NP1}) \} \]
13.6 First Application of V+NP1 in the word bank 12.3

V+NP1: \[
\begin{array}{l}
M\text{-verb: } a \\
NP: x \ b \ y \\
prn: c \\
\end{array}
\]
\[
\begin{array}{l}
V+NP2: \[
\begin{array}{l}
M\text{-np: } b \\
VERB: a \\
prn: c \\
\end{array}
\] \Rightarrow \[
\begin{array}{l}
M\text{-verb: } a \\
\{ 3 \ V+NP1, 4 \ V+NP2, 5 \ V+epr \} \\
\end{array}
\]
\[
\begin{array}{l}
I\text{-concept}_{loc}: eat \\
P: \text{indicative} \\
C: \[
\begin{array}{l}
MOD: \\
NP: Peter, salad \\
prn: 5 \\
\end{array}
\] \\
I: \[
\begin{array}{l}
epr: [5 \ \text{then} \ 6] \\
epr: [4 \ \text{then} \ 5] \\
\end{array}
\] \\
\end{array}
\]
\[
\begin{array}{l}
I\text{-concept}_{loc}: salad \\
P: \text{indicative} \\
C: \[
\begin{array}{l}
MOD: \\
VERB: eat \\
prn: 5 \\
\end{array}
\] \\
I: \[
\begin{array}{l}
prn: 5 \\
id: 2 \\
\end{array}
\] \\
\end{array}
\]
\[
\begin{array}{l}
I\text{-concept}_{loc}: eat \\
P: \text{indicative} \\
C: \[
\begin{array}{l}
MOD: \\
NP: Peter, salad \\
prn: 5 \\
\end{array}
\] \\
I: \[
\begin{array}{l}
epr: [5 \ \text{then} \ 6] \\
epr: [4 \ \text{then} \ 5] \\
\end{array}
\] \\
\end{array}
\]

©1999 Roland Hausser
13.7 Extrapropositional epr-navigation

13.8 Extrapropositional id-navigation
14. Different temporal prepositions depending on direction of navigation

14.1 epr-coordination

Peter leaves the house. Then he crosses the street.
Peter crosses the street. Before that he leaves the house.

14.2 epr-subordination (adverbial clauses)

Before Peter crosses the street, he leaves the house.
Peter, before he crosses the street, leaves the house.
Peter leaves, before he crosses the street, the house.
Peter leaves the house, before he crosses the street.
After Peter leaves the house, he crosses the street.
Peter, after he leaves the house, crosses the street.
Peter crosses, after he leaves the house, the street.
Peter crosses the street, after he leaves the house.

14.3 id-coordination

Peter orders a salad. The salad is eaten by Peter.
14.4 id-subordination (relative clause)

Peter orders a salad which he eats.

14.5 Different realizations of conjunctions

<table>
<thead>
<tr>
<th>coordinating forward:</th>
<th>temporal</th>
<th>causal</th>
<th>modal</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>coordinating backward:</th>
<th>P2. Earlier P1.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>subordinating forward:</th>
<th>p1, before P2, p1.</th>
<th>p1, for which reason P2, p1.</th>
<th>p1, as P2, p1</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>subordinating backward:</th>
<th>p2, after P1, p2.</th>
<th>p2, because P1, p2.</th>
</tr>
</thead>
</table>

14.6 Adverbial embedding navigation

Peter crossed, after he left the house, the street.

▼ cross  Peter  V+epr  street
prn:2  prn:2  leave  Peter house ▲  prn: 2
(2 then 3)  id: 1
(1 then 2)  prn: 1  prn:1  prn:1  id: 3

©1999 Roland Hauser
14.7 Universality and language specificity in a SLIM machine