The semantics of event nominalisation

Sebastian Bücking¹ and Markus Egg²

¹Universität Tübingen
sebastian.buecking@uni-tuebingen.de

²Rijksuniversiteit Groningen
egg@let.rug.nl

12 December 2008

Bücking and Egg
The semantics of event nominalisation

Structure of the talk

- two challenges for a semantic description of nominalisations
  - argument linking
  - anti-iconic readings
- the framework of analysis: underspecified semantic representation
- the analysis
- semantic construction

Event nominalisations and verbal arguments 1

- derivational affixes turn verbs into nouns denoting events or one of their participants
  1. die Verarbeitung der Daten
     'the processing of the data'
  2. beautiful dancer
- arguments of the verb can become optional during nominalisation
- others are explicitly bound in the nominalisation process
- this binding refers to thematic roles of arguments
  - agent for the agentic suffix -er
  - theme for the genitive complement of -ung-nominalisations

Bücking and Egg
The semantics of event nominalisation

Event nominalisations and verbal arguments 2

- to handle reference to thematic roles and argument linking, we use a Neo-Davidsonian approach (Parsons 1990)
  - verbs and their arguments are linked in terms of explicit thematic relations
    3. \[ \lambda y \lambda x \lambda e. eat'(x, y)(e) \]
    4. \[ \lambda y \lambda x \lambda e. eat'(e) \land \text{AGENT}(e, x) \land \text{THEME}(e, y) \]
- no need to bind off syntactically unrealised arguments of the verb stem in the semantics
  5. \[ \lambda x \lambda e. \exists y. eat'(x, y)(e) \]
  6. \[ \lambda x \lambda e. eat'(e) \land \text{AGENT}(e, x) \]
Event nominalisations and verbal arguments

- Nominalisation affixes are semantic functions and their stems are arguments.
- The affixes introduce the relevant thematic relations.
  - The agentive -er:
    \[ \lambda P \lambda x \exists e. P(e) \land \text{AGENT}(e, x) \]
  - The affix -ung (preliminary):
    \[ \lambda P \lambda x \lambda e. P(e) \land \text{THEME}(e, x) \]

Semantic construction for (1)

- Semantics of Verarbeitung 'processing':
  \[ \lambda x \lambda e. \text{process}'(e) \land \text{THEME}(e, x) \]
- Semantics of Verarbeitung der Daten 'processing of the data':
  \[ \lambda e \exists x. \text{[data]}'(x) \land \text{process}'(e) \land \text{THEME}(e, x) \]

Anti-iconic readings of event nominalisations

- Modified agentive nouns exhibit ambiguity due to several possible ways of integrating the modifier semantics.
  \[ \lambda P \lambda x \exists e. P(e) \land \text{AGENT}(e, x) \]
- The second reading can be analysed as an anti-iconic, in that the modifier can pertain to the stem only (Egg 2006)

Underspecified representation of anti-iconic cases

- Underspecified representation ('constraint') for the meaning of (9)
  \[ \lambda x \exists e. \text{[NP]} \land \text{AGENT}(e, x) \]
- Sets of semantic representations (here, \( \lambda \)-terms) are described on a meta-level
- Ingredients: fragments of \( \lambda \)-terms, 'holes', and relations between them

Bücking and Egg  The semantics of event nominalisation
the described semantic representations (‘solutions’) are derived by identifying fragments and holes
formally, a bijective mapping (the ‘plugging’ of Bos 2004)
the solutions of (13)
\[
\begin{align*}
\lambda x \exists e. & \text{dance'}(e) \land \text{AGENT}(e, x) \land \text{beautiful'}(x) \\
\lambda x \exists e. & \text{dance'}(e) \land \text{AGENT}(e, x) \land \text{beautiful'}(e)
\end{align*}
\]

possible relations between the meanings of verbal predicates and their nominalizations
- verb nominalisations have the same meaning as their verb stem (Parsons 1990; Cocchiarella 1996)
- verb nominalisations introduce concrete events that are characterised (relation ‘≈’) by abstract event concepts in terms of verb stem meanings (Cocchiarella 1996)
we follow position 2 and distinguish types for concrete events and for event concepts (cp. Asher 1993)
- variables for concrete events are abbreviated as ‘e’
- variables for event concepts are abbreviated as ‘E’

the more elaborate semantics of the affix -ung is split into a main and a secondary fragment
- main fragment: \( \lambda P \lambda x e. e \approx \lambda E P(E) \)
- secondary fragment: \( \lambda P \lambda x \lambda e. \text{THEME}(e, x) \land P(e) \)

the modifier can pertain to two different entities
- the underlying verbal E (resulting in the internal reading)
- the nominal e (resulting in the external reading)
the verbal theme argument is made available for binding by a subsequent genitive DP argument

\[
\begin{align*}
\text{Verarbeitung} & : \lambda e. e \approx \lambda E \square(E) \\
\text{N} & \lambda \lambda e. \text{THEME}(e, x) \land \text{process'}(e)
\end{align*}
\]
Event nominalisations and their challenges

The formalism

The analysis

Semantic construction

References

Underspecified representation of event nominalisations 1

- semantic representation of *Verarbeitung der Daten* ‘processing of the data’

\[(19)\]

\[
\begin{align*}
\text{NP} : & \quad \Box \\
\lambda e \exists! x. [\text{data}(x)] \land \Box (e) & \quad \lambda e. e \approx \lambda E. \Box (E) \\
\text{NP}_2 : & \quad \lambda e. \text{THEME}(e, x) \land \text{process}'(e)
\end{align*}
\]

Bücking and Egg. The semantics of event nominalisation

Underspecified representation of event nominalisations 2

- constraints like (19) have one solution only
  - unwanted scope ambiguities between top fragment of nominalisations and DP argument fragment are blocked
  - this is due to the typing of event (concept) variables
  - the DP semantics introduces concrete events \( e \)
  - the inner event variable in nominalisations is an event type \( E \)
  - verb (and adverbial) semantics introduce a general event type \( e \)
  - the only solution for (19):

\[(20)\]

\[
\begin{align*}
\lambda e \exists! x. [\text{data}(x)] \land e \approx \lambda E. \text{THEME}(E, x) \land \text{process}'(E)
\end{align*}
\]

- in the following, reference to these types is mostly dropped

Bücking and Egg. The semantics of event nominalisation

Underspecified representation of event nominalisations 3

- representation for *schnelle Verarbeitung der Daten* ‘fast processing of the data’

\[(21)\]

\[
\begin{align*}
\text{NP} : & \quad \Box \\
\lambda e \exists! x. [\text{data}(x)] \land \Box (e) & \quad \lambda e. e \approx \lambda E. \Box (E) \\
\text{NP}_2 : & \quad \lambda y. \Box (y) \land \text{fast}'(y) \\
\lambda e. \text{THEME}(e, x) \land \text{process}'(e)
\end{align*}
\]

Bücking and Egg. The semantics of event nominalisation

Underspecified representation of event nominalisations 4

- solutions by starting

\[(22)\]

\[
\begin{align*}
\lambda e \exists! x. [\text{data}(x)] \land e \approx \\
\lambda E. \text{THEME}(E, x) \land \text{process}'(E) \land \text{fast}'(E)
\end{align*}
\]

\[(23)\]

\[
\begin{align*}
\lambda e \exists! x. [\text{data}(x)] \land e \approx \\
\lambda E. \text{THEME}(E, x) \land \text{process}'(E) \land \text{fast}'(E)
\end{align*}
\]

Bücking and Egg. The semantics of event nominalisation
Locatives 1

- **Locatives** are another instance of external vs. internal reading
- (24) *die Zubereitung des Huhns in der Pfeffersauce*  
  the preparation the chicken\textsubscript{GEN} in a pepper-sauce
- (25) *die Zubereitung in der Pfeffersauce des Huhns*  
  the preparation in the pepper-sauce the chicken\textsubscript{GEN}
- (26) *die Zubereitung des Huhns in der Küche*  
  the preparation the chicken\textsubscript{GEN} in the kitchen

---

Bücking and Egg  
The semantics of event nominalisation

---

Locatives 2

- diamond for *Zubereitung des Huhns in der Pfeffersauce*  
  ‘preparation of the chicken in the pepper sauce’; simplified w.r.t. the DP argument semantics
- (27) 
  \[
  \lambda e. \approx \lambda E. \square(\langle E \rangle) \land \lambda y. (y) \land \text{in}'(y, P) \\
  \lambda e. \text{THEME}(\langle e, 1x\cdot\text{chicken}'(x) \rangle) \land \text{prepare}'(e)
  \]

---

Bücking and Egg  
The semantics of event nominalisation

---

Locatives 3

- solutions of (27)
- (28)  
  (a) internal reading  
  \[
  \lambda e.e \approx \lambda E.\text{in}'(\langle E, P \rangle) \land \text{THEME}(E, 1x\cdot\text{chicken}'(x)) \land \text{prepare}'(E)
  \]
  (b) external reading  
  \[
  \lambda e.\text{in}'(\langle e, P \rangle) \land e \approx \lambda E.\text{THEME}(E, 1x\cdot\text{chicken}'(x)) \land \text{prepare}'(E)
  \]
- the second reading is ruled out pragmatically: ‘the preparation event as a whole is localised in the pepper-sauce’
- the first reading raises the question of what it means to localise the lexical base $E$, i.e., how does this apply to the chicken?

---

Bücking and Egg  
The semantics of event nominalisation

---

Locatives 4

- Maienborn’s (2003) proposal for adverbial locatives
- (29)  
  MOD*: $\lambda Q, P, \lambda x[P(x) \& R(x, v) \& Q(v)]$
  Condition: if MOD* applies to categorial type $X$, $R = \text{part-of}$, otherwise (in an XP-environment) $R$ is identity.
- (30)  
  $[\text{VP} \ [\text{PP in der Küche} \ [\text{VP das Huhn zubereiten}] ] ]$
  $[\text{VP} \ [\text{PP in the kitchen} \ [\text{VP the chicken prepare}] ] ]$
- (31)  
  $[\text{VP das Huhn} \ [\text{VP in einer Pfeffersauce} \ [\text{VP zubereiten}] ] ]$
  $[\text{VP the chicken} \ [\text{VP in a pepper-sauce} \ [\text{VP prepare}] ] ]$
- (32)  
  external: $\lambda e \ldots \text{prepare}'(e) \land \text{in}'(e, K) \ldots$
- (33)  
  internal: $\lambda e \ldots \text{prepare}'(e) \land \text{part-of}'(e, v) \land \text{in}'(v, P) \ldots$
- in the internal case, the free variable $v$ is conceptually specified

---

Bücking and Egg  
The semantics of event nominalisation
Semantic construction of anti-iconic cases

- Somewhere the gap between syntactic and semantic structure must be bridged.
- Several possibilities have been suggested:
  - Syntactic preprocessing (Generative Grammar)
  - Involved syntactic construction (L-TAG, Kallmeyer and Romero 2008; LFG, Çetinoğlu and Oflazer 2006)
  - Powerful syntax-semantics interface (Egg 2006)
  - Semantic representations are constraints
  - A main and a secondary fragment are distinguished
  - Interface rules refer to these fragments and determine them for emerging constituents

Semantic construction 3

- Semantics of Verarbeitung ‘processing’

$$\text{[\text{Verarbeitung}} \vdash \lambda e \approx \lambda E. P(E)$$

$$\text{[N2} \vdash \lambda P \lambda y \lambda e. P(e) \land \text{THEME}(e, y) \text{]}(\text{process}')$$

- After β-reduction

$$\text{[\text{N3} \vdash \lambda y \lambda e. \text{process}'(e) \land \text{THEME}(e, y)](E)}$$
Semantic construction 4

- rule for DP arguments

\[(\exists X. DP) \stackrel{(SSI)}{\Rightarrow} [X] : [DP]; [X_2] : [DP]([X_2])\]

- after the integration of the DP-argument *der Daten* 'of the data'

\[\lambda_e. e \approx \lambda E. \square (E) \quad \square : \lambda e \exists ! x. [\text{data}'(x)] \wedge \square (e)\]

\[\lambda e. \text{process}'(e) \wedge \text{THEME}(e, x)\]

- References


