Verbal Bracketing Paradoxes
What heavy drinkers can tell us about movement

Zoë Belk
Linguistics, UCL

zoe.belk@ucl.ac.uk
The plan

1. Traditional bracketing paradoxes
2. A second type of bracketing paradox
3. Rebracketing verbal bracketing paradoxes
4. What is Information Preservation?
What is a bracketing paradox?

• Mismatch between morphophonology and semantics

• Meaning is still compositional, just not in expected way (cf. old friend)

• Traditional examples include nuclear physicist, mediaeval historian
# Traditional bracketing paradoxes

**LF bracketing:**
- [[hydroelectric]ity]
- [[ungrammatical]ity]
- [[unhappi]er]
- [[nuclear physic]ist]
- [[transformational grammar]ian]
- [[Gödel number]ing]

**PF bracketing:**
- [hydro[electricity]]
- [un[grammaticality]]
- [un[happier]]
- [nuclear [physicist]]
- [transformational [grammari]an]
- [Gödel [numbering]]
Bracketing paradoxes in Dutch

• In Dutch, prenominal modifiers must appear with a schwa in certain contexts, including in a definite noun phrase:
  
  de beroemd\(^{-e}\) gitarist  
  \textit{the famous guitarist}  
  de productief\(^{-e}\) generativist  
  \textit{the productive generativist}\(^1\)

• However, this schwa does not appear with bracketing paradoxes:
  
  de klassiek\(^{-e}\) gitarist  
  \textit{the classical guitarist}  
  de transformationeel\(^{-e}\) generativist  
  \textit{the transformational generativist}\(^1\)
Analysing bracketing paradoxes

- Traditionally, the syntax was understood to manipulate the same building blocks as the phonology

→ PF structure determined by syntax

- Sproat proposed separating the two – phonological structure could differ from syntactic
Analysing bracketing paradoxes

- Sproat introduced a Mapping Principle, to ensure the two levels of structure were constrained in the way they could differ.

- This meant the syntactic structure could be mapped on to the phonological structure.

\[
\begin{align*}
&\text{N} \\
&\quad \text{A} \\
&\quad \quad \text{Af} \\
&\quad \quad \quad \text{un} \\
&\quad \quad \quad \text{happy} \\
&\quad \text{Af} \\
&\quad \quad \text{A} \\
&\quad \quad \quad \text{AfN} \\
&\quad \quad \quad \quad \text{er} \\
&\text{N} \\
&\quad \text{Af} \\
&\quad \quad \text{un} \\
&\quad \quad \text{happy} \\
&\quad \text{Af} \\
&\quad \quad \text{A} \\
&\quad \quad \quad \text{AfN} \\
&\quad \quad \quad \quad \text{er}
\end{align*}
\]
A second type of bracketing paradox

• One phonological form, but two meanings, so mismatch between phonological structure and at least one semantic structure

• Evidence for both bracketings, as in traditional cases

• Derived from verbs: heavy drinker, hard worker
  – Similar underived forms are not paradoxes: *beautiful ballerina, *high chorister
Verbal bracketing paradoxes

**LF bracketing**
- [[hard work]er]
- [[beautiful danc]er]
- [[heavy drink]er]
- [[close talk]er]
- [[high sing]er]

**PF bracketing**
- [hard [worker]]
- [beautiful [dancer]]
- [heavy [drinker]]
- [close [talker]]
- [high [singer]]
Verbal bracketing paradoxes in Dutch

• While traditional bracketing paradoxes disallow the schwa where it would otherwise be expected, verbal bracketing paradoxes require it:
  
  de mooi*(-e) danser  de hard*(-e) werker
  
  *the beautiful dancer  *the hard worker

• What’s going on?
Dutch schwas

• The two types of paradox have opposing behaviour with regard to the schwa
  – This suggests they aren’t the same phenomenon and shouldn’t receive the same analysis

• Verbal bracketing paradoxes behave like normal adjective+noun phrases; traditional bracketing paradoxes look different

• Our analysis should reflect this!
Analysing bracketing paradoxes II

- Sproat’s analysis looks pretty good for traditional bracketing paradoxes: it predicts that they shouldn’t behave like N+A constituents because they don’t look like them syntactically.
Analysing bracketing paradoxes II

• But that doesn’t help us with verbal bracketing paradoxes. The same analysis can’t apply to these because it would require reordering the adjective and noun (compare *hard worker to works hard*)
  – But reordering doesn’t seem to be an option: when affixing -er to a non-head-final structure like a verb followed by a particle, you get all kinds of affixation except reordering:
    • passer by, come outer, cleaner upper…
    • *bypasser, *outcomer, *upcleaner…
What now?

• Traditional bracketing paradoxes are seen as a mismatch between syntax and PF

• What if verbal bracketing paradoxes are a mismatch between syntax and LF?
Rebracketing verbal bracketing paradoxes

• I suggest that verbal bracketing paradoxes result from an adjustment of the syntactic structure at LF

• This rebracketing is constrained by Information Preservation:
  – PRESERVATION OF HEADEDNESS: Don’t destroy headedness relations
  – PRESERVATION OF HIERARCHY: Don’t destroy c-command relations between non-heads
If this unary branch is left at the end of the rebracketing, either it can remain and be trivial, or it can be cleaned up through a simple rule. The choice makes no difference to the analysis presented here, so I leave it up to the reader to choose which is preferable to her.

As for the first question, I am again agnostic as to the answer. Two options exist: position 2 is destroyed completely and the content occupying it is placed under node Y (as X), or node 2 is delinked from 1 and the same node is relinked to 4 as its daughter. I do not see any way of distinguishing between the two empirically, and so will again leave the choice to the reader.

As can be seen from the preceding material, this LF rebracketing procedure relies only on the two tenets of Information Preservation, those of Preservation of Hierarchy and Preservation of Headedness. After rebracketing, all key information in the tree is retained, with the only exception being the information that is the target of the rebracketing itself.

In the next subsection, I will show that this rebracketing procedure is sufficiently constrained, despite its simplicity.

5.2.2 The rebracketing procedure proposed above has the result that only two hierarchically adjacent dependents may be rebracketed, and that only the bottom two dependents can be rebracketed.

Consider the tree in (48), where H is the head and A-C dependents.

(48)

* H

B H

C H

A

H

A c-commands B and C, B c-commands C and C does not c-command any other non-head. Any attempt to rebracket this tree so that A and C are sisters will fail. Here, H remains the head, but A no longer c-commands B, which is a violation of Preservation of Headedness:

(49)

* H

B H

C H

A

Similarly, in the following tree, B no longer c-commands C.

(50)

* H

C H

A

B H

In (51), Preservation of Headedness is violated, no matter which of B or H is the head of that constituent.

(51)

* C

A

B/H

H

The only option for rebracketing the tree in (48) is to make two adjacent dependents sisters (in particular, for reasons to be discussed below, the lowest two, B and C):

(52)

H

A

H

C

B

H

In this structure, H is still the head, A c-commands both B and C, B c-commands C and a new c-command relationship, between C and B has been created.

Information Preservation also has the result that only the lowest two dependents can be rebracketed with respect to each other. Taking the same structure as in (48), repeated in (53), as our starting point, any attempt to create a sisterhood relationship between A and B will fail.

(53)

H

A

H

B

H

C

A

In the resulting structure, either A or B will no longer c-command C, depending on which is the head.

(54)

* H

A/B

A

B

H

C

H

Preservation of Hierarchy is therefore violated. Again, only the structure in (52) is a viable rebracketing.

This fact is not a peculiarity of trees with three non-heads. When trying to rebracket the non-heads of any larger structure, for example that in (55), we find the same result.

(55)

H

A

H

B

H

C

D

H

We know from (54) that rebracketing B and C as sisters will violate Preservation of Hierarchy, because either B or C will no longer c-command D, depending on which is head of their constituent:
Information Preservation in action

IP has the following effects:

- Only structurally adjacent non-heads can become sisters
  - More particularly, only bottom-most two non-heads can become sisters

- In other words, rebracketing can only occur where a non-head moves down precisely one level to form a constituent with the lowest non-head
Information Preservation in action

• This means that a non-constituent can’t be interpreted as a constituent, and the rebracketing must be both shallow and local

• In other words, it maintains a restrictive theory of movement
Rebracketing verbal bracketing paradoxes

• The result is that only a very few kinds of rebracketing are allowed, among them:

1. **(14)**
   - a. \[N \rightarrow A \rightarrow N \rightarrow V \rightarrow A_{fN} \rightarrow Af_N \]
   - hard \[\rightarrow work \rightarrow \text{V} \rightarrow \text{Af_N} \rightarrow \text{Er} \]

2. **(15)**
   - a. \[N \rightarrow V \rightarrow Af_N \rightarrow A \rightarrow N \rightarrow V \rightarrow A_{fN} \rightarrow Af_N \]
   - work \[\rightarrow hard \rightarrow \text{V} \rightarrow \text{Af_N} \rightarrow \text{Er} \]

In this case, the PF and syntactic structures are isomorphic, while LF is a mismatch.

This analysis requires that adjectives and adverbs are underlyingly the same category, but explains several characteristics of verbal bracketing paradoxes.

Adjacency between noun and AP is explained. Given a string like (15-a), the only possible rebracketing would be (15-b). Similarly (16-a) and (16-b).

4. **S o m e p r e d i c t i o n s o f L F r e b r a c k e t i n g**

4.1 **Dutch data**

In Dutch, prenominal modifiers appear with a declensional schwa in certain contexts. A partial approximation of this rule is as follows:

1. **(17)**
   - Prenominal adjectives must be conjugated with a declensional schwa when they are part of a definite DP. In the absence of a determiner, they must not be conjugated.

   Prenominal modifiers must appear with a declensional schwa in a definite DP:

   \[\text{This is a very much simplified version of the rule, which may additionally be undergoing language change. This version is detailed enough for our purposes.}\]
But what is Information Preservation?

- Information Preservation is a restriction on all movement
- The movement operation can be separated from chain formation
- Where movement can occur without violating IP, no trace is necessary; otherwise, a trace may be used subject to chain formation
  - The trace can be used to ensure no destruction of c-command relations between non-heads
I’m not convinced…

• Information Preservation is not really a new idea

• Most of it is already built into restrictions on chain formation

• Don’t be scared by movement without a trace – it’s constrained enough to still be interpretable, and only allows a tiny number of new movement configurations
Rebracketing verbal bracketing paradoxes

- This approach predicts that verbal bracketing paradoxes should behave syntactically like A+N combinations (because they look the same in syntax)

- It predicts that traditional and verbal bracketing paradoxes should behave differently

- In other words, it predicts exactly the patterns found in Dutch – result!
Conclusion

• Bracketing paradoxes occur when there are mismatches between syntactic structure and the structure required by other modules

• Extant analyses for traditional bracketing paradoxes can’t account for verbal bracketing paradoxes

• My proposal predicts exactly the patterns found in Dutch, while maintaining a restrictive theory of movement
References


Comparing IP and chain formation

$(A_1, \ldots, A_n)$ is a chain iff, for $1 \leq i \leq n$

1. $A_i = A_{i+1}$
2. $A_i$ c-commands $A_{i+1}$
3. $A_{i+1}$ is in a Minimal Configuration with $A_i$

$Y$ is in a Minimal Configuration with $X$ iff there is no $Z$ such that

a. $Z$ is of the same structural type as $X$, and
b. $Z$ intervenes between $X$ and $Y$