Ling 566
Nov 20, 2025

Long Distance Dependencies

Overview

- Some examples of the phenomenon
- What is new and different about it
- Brief sketch of the TG approach
- Broad outlines of our approach
- Details of our approach
- Subject extraction
- Coordinate Structure Constraint
- Reading questions

Examples

- *wh*-questions:

  What did you find?

  Tell me who you talked to

- relative clauses:

  the item that I found

  the guy who(m) I talked to

- topicalization:

  The manual, I can't find

  Chris, you should talk to.

- *easy*-adjectives:

  My house is easy to find.
  Pat is hard to talk to.

What these have in common

- There is a ‘gap’: nothing following *find* and *to*, even though both normally require objects.
- Something that fills the role of the element missing from the gap occurs at the beginning of the clause.
- We use topicalization and *easy*-adjectives to illustrate:

  The manual, I can’t find _____

  Chris is easy to talk to _____

Gaps and their fillers can be far apart:

- *<u>The solution to this problem</u>, Pat said that someone claimed you thought I would never find____.*

- *<u>Chris</u> is easy to consider it impossible for anyone but a genius to try to talk to____.*

☞ That's why we call them "long distance dependencies"

Fillers often have syntactic properties associated with their gaps

Him, I haven't met___.

**He, I haven't met___.*

The scissors, Pat told us ____ were missing.

**The scissors, Pat told us ____ was missing.*

On Pat, you can rely___.

**To Pat, you can rely___.*

LDDs in TG

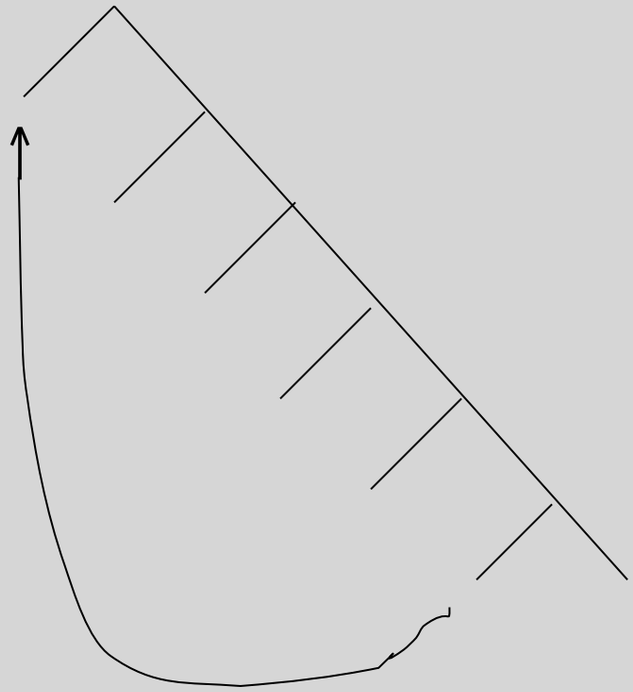
- These were long thought to constitute the strongest evidence for transformations.

- They were handled in TG by moving the filler from the gap position.

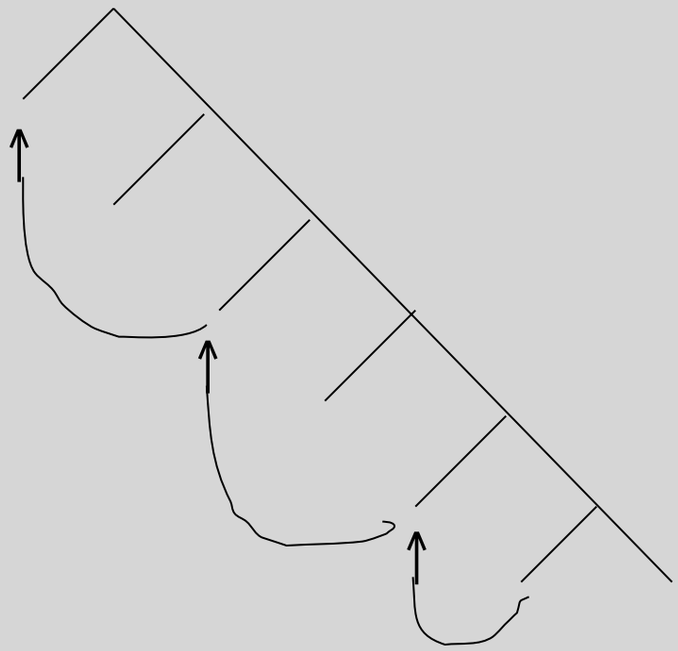
- Case, agreement, preposition selection could apply before movement.

A big debate about LDDs in TG

- Does long-distance movement take place in one fell swoop or in lots of little steps?



Swooping



Looping

Looping is now generally accepted in TG

- Various languages show morphological marking on the verbs or complementizers of clauses between the filler and the gap.
- Psycholinguistic evidence indicates increased processing load in the region between filler and gap.
- This opens the door to non-transformational analyses, in which the filler-gap dependency is mediated by local information passing.

Very Rough Sketch of Our Approach

- A feature GAP records information about a missing constituent.
- The GAP value is passed up the tree by a new principle.
- A new grammar rule expands S as a filler followed by another S whose GAP value matches the filler.
- Caveat: Making the details of this general idea work involves several complications.

The Feature GAP

- Like valence features and ARG-ST, GAP's value is a list of feature structures (often empty).
- Subject gaps are introduced by a lexical rule.
- Non-subject gaps are introduced by revising the Argument Realization Principle.

The Revised ARP

$$word\text{:}\ \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{VAL} & \begin{bmatrix} \text{SPR} & \boxed{A} \\ \text{COMPS} & \boxed{B} \ominus \boxed{C} \end{bmatrix} \\ \text{GAP} & \boxed{C} \end{bmatrix} \\ \text{ARG-ST} & \boxed{A} \oplus \boxed{B} \end{bmatrix}$$

- $\ominus$ is a kind of list subtraction, but:
  - it's not always defined, and
  - when defined, it's not always unique
- The ARP now says the non-SPR arguments are distributed between COMPS and GAP.

List subtraction

$$\langle \text{ NP, PP } \rangle \ominus \langle \text{ NP } \rangle =$$

$$\langle \text{ NP, PP, NP } \rangle \ominus \langle \text{ NP } \rangle =$$

$$\langle \text{ NP, PP, NP } \rangle \ominus \langle \text{ NP, PP } \rangle =$$

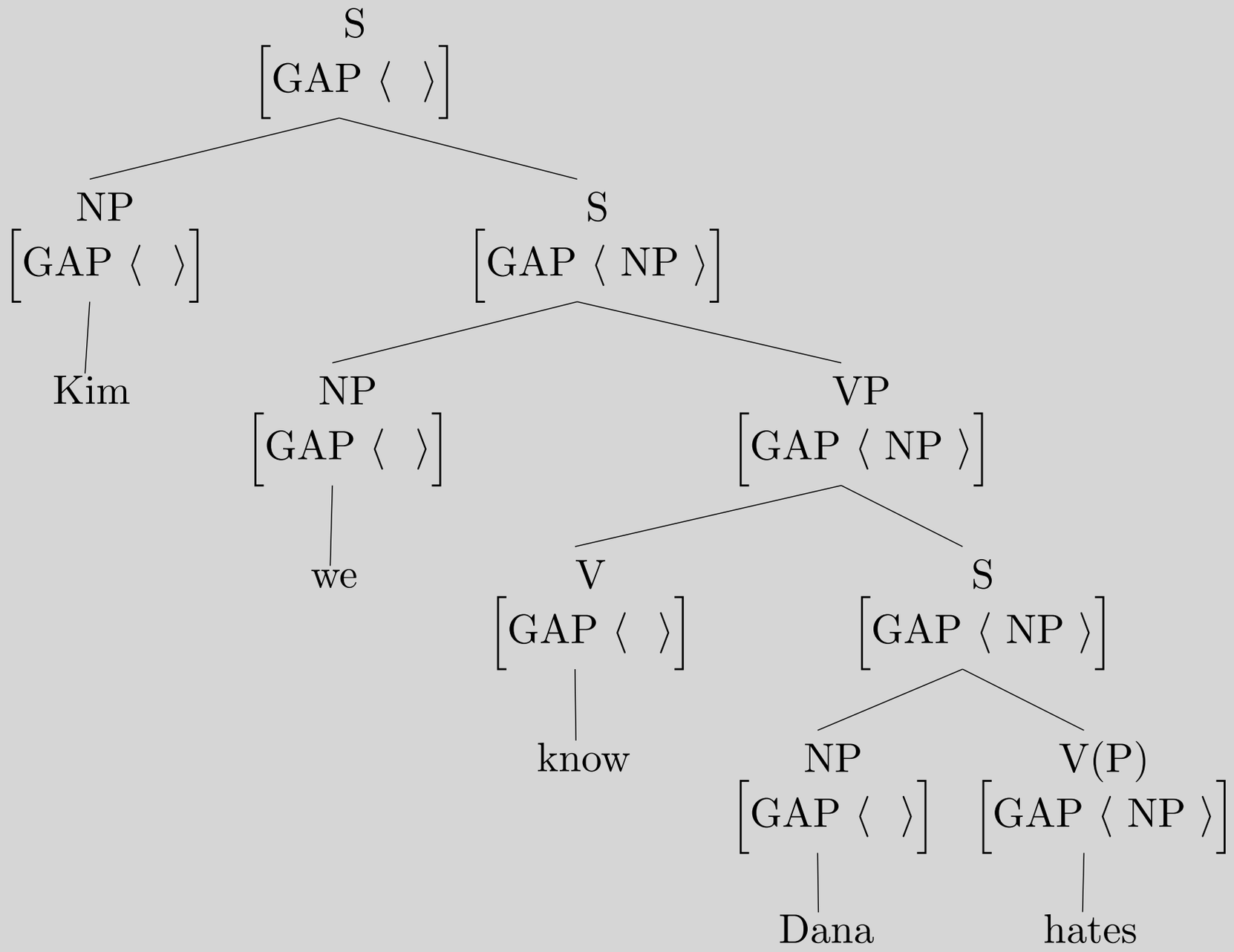
$$\langle \text{ NP, PP, NP } \rangle \ominus \langle \text{ NP, VP } \rangle =$$

$$\langle \text{ VP, PP, NP } \rangle \ominus \langle \text{ NP, VP } \rangle =$$

A Word with a Non-Empty GAP Value

$$\left\langle \text{hand} , \left[\begin{array}{l} \textit{word} \\ \\ \text{SYN} \left[\begin{array}{ll} \text{HEAD} & \left[\text{FORM} \quad \text{fin}\right] \\ \text{VAL} & \left[\begin{array}{ll} \text{SPR} & \langle\, \boxed{1} \,\rangle \\ \text{COMPS} & \langle\, \boxed{3}\text{PP[to]} \,\rangle \end{array}\right] \\ \text{GAP} & \langle\, \boxed{2}\text{NP[acc]} \,\rangle \end{array}\right] \\ \\ \text{ARG-ST} \quad \left\langle \left[\begin{array}{ll} & \boxed{1}\text{NP} \\ \text{CASE} & \text{nom} \\ \text{AGR} & \textit{non-3sing} \end{array}\right] , \boxed{2} , \boxed{3} \right\rangle \end{array}\right] \right\rangle$$

How We Want GAP to Propagate



What We Want the GAP Propagation Mechanism to Do

- Pass any GAP values from daughters up to their mothers,
- **except** when the filler is found.
- For topicalization, we can write the exception into the grammar rule, but
- For *easy*-adjectives, the NP that corresponds to the gap is the subject, which is introduced by the Head-Specifier Rule.
- Since specifiers are not generally gap fillers, we can't write the gap-filling into the HSR.

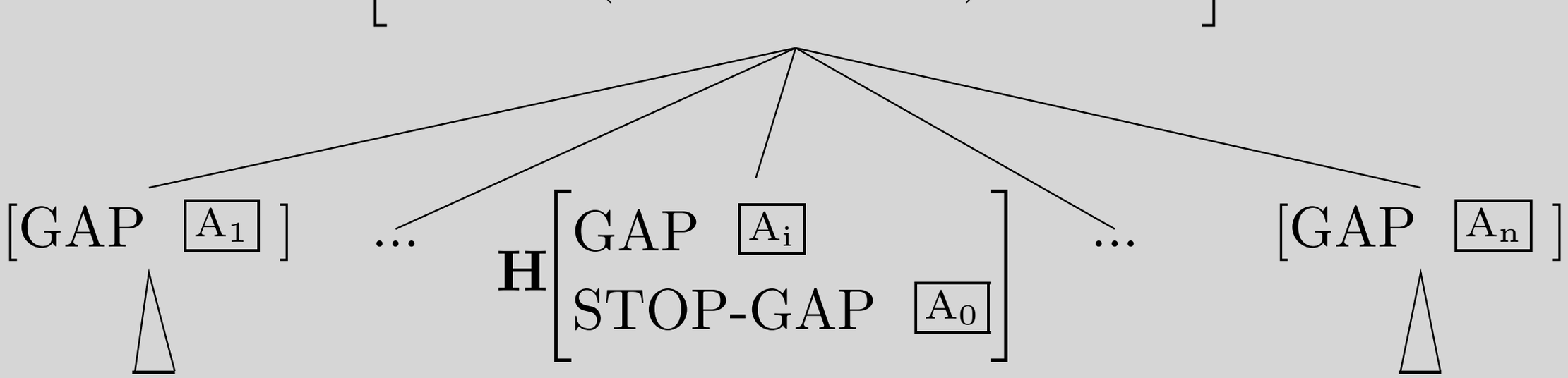
Our Solution to this Problem

- For *easy*-adjectives, we treat the adjective formally as the filler, marking its SPR value as coindexed with its GAP value.
- We use a feature STOP-GAP to trigger the emptying of the GAP list.
  - STOP-GAP stops gap propagation
  - *easy*-adjectives mark STOP-GAP lexically
  - a new grammar rule, the Head-Filler Rule mentions STOP-GAP

The GAP Principle

A local subtree Φ satisfies the GAP Principle with respect to a headed rule ρ if and only if Φ satisfies:

$$\left[\text{GAP} \quad (\boxed{A_1} \oplus \ldots \oplus \boxed{A_n}) \ \ominus \ \boxed{A_0}\right]$$



How does STOP-GAP work?

- STOP-GAP is empty almost everywhere
- When a gap is filled, STOP-GAP is nonempty, and its value is the same as the gap being filled.
- This blocks propagation of that GAP value, so gaps are only filled once.
- The nonempty STOP-GAP values come from two sources:
  - a stipulation in the Head-Filler Rule
  - lexical entries for *easy*-adjectives
- No principle propagates STOP-GAP

The Head-Filler Rule

$$[phrase] \rightarrow \boxed{1}\begin{bmatrix}\text{GAP} & \langle\,\rangle\end{bmatrix} \quad \mathbf{H}\begin{bmatrix}\text{HEAD} & \begin{bmatrix}verb \\ \text{FORM} & \text{fin}\end{bmatrix} \\ \text{VAL} & \begin{bmatrix}\text{SPR} & \langle\,\rangle \\ \text{COMPS} & \langle\,\rangle\end{bmatrix} \\ \text{STOP-GAP} & \langle\,\boxed{1}\,\rangle \\ \text{GAP} & \langle\,\boxed{1}\,\rangle\end{bmatrix}$$

- This only covers gap filling in finite Ss
- The filler has to be identical to the GAP value
- The STOP-GAP value is also identical
- The GAP Principle ensures that the mother's GAP value is the empty list

Gap Filling with *easy*-Adjectives

$$\left\langle \text{easy} , \left[\begin{array}{ll} \textit{adj-lxm} & \\ \text{SYN} & \left[\text{STOP-GAP} \ \langle \boxed{1} \rangle\right] \\ \text{ARG-ST} & \left\langle \text{NP}_i , \left[\begin{array}{ll} \text{INF} & + \\ \text{GAP} & \langle \boxed{1}\text{NP}_i , \ldots \rangle \end{array} \right]^{\text{VP}} \right\rangle \end{array} \right] \right\rangle$$

- Because STOP-GAP and GAP have the same value, that value will be subtracted from the mother's GAP value.
- The first argument is coindexed with the GAP value, accounting for the interpretation of the subject as the filler.

A Tree for *easy to talk to___*

$$\left[\begin{array}{ll}\text{VAL} & \left[\text{SPR} \quad \langle\ \boxed{2}\text{NP}_i\ \rangle\right] \\ \text{GAP} & \langle\ \rangle\end{array}\right]$$

A

$$\left[\begin{array}{ll}\text{VAL} & \left[\begin{array}{ll}\text{SPR} & \langle\ \boxed{2}\ \rangle \\ \text{COMPS} & \langle\ \boxed{3}\ \rangle\end{array}\right] \\ \text{GAP} & \langle\ \rangle \\ \text{STOP-GAP} & \langle\ \boxed{1}\ \rangle\end{array}\right]$$

easy

$\boxed{3}$VP

$$\left[\begin{array}{ll}\text{VAL} & \left[\text{SPR} \quad \langle\ \text{NP}\ \rangle\right] \\ \text{GAP} & \langle\ \boxed{1}\text{NP}_i\ \rangle\end{array}\right]$$

to talk to __

SPR of complement of easy not-identified with anything

0

Hang on, that is a SPR requirement!

0

Seems fishy

0

That’s cool, no worries

0

Ohh subtle … nice!

0

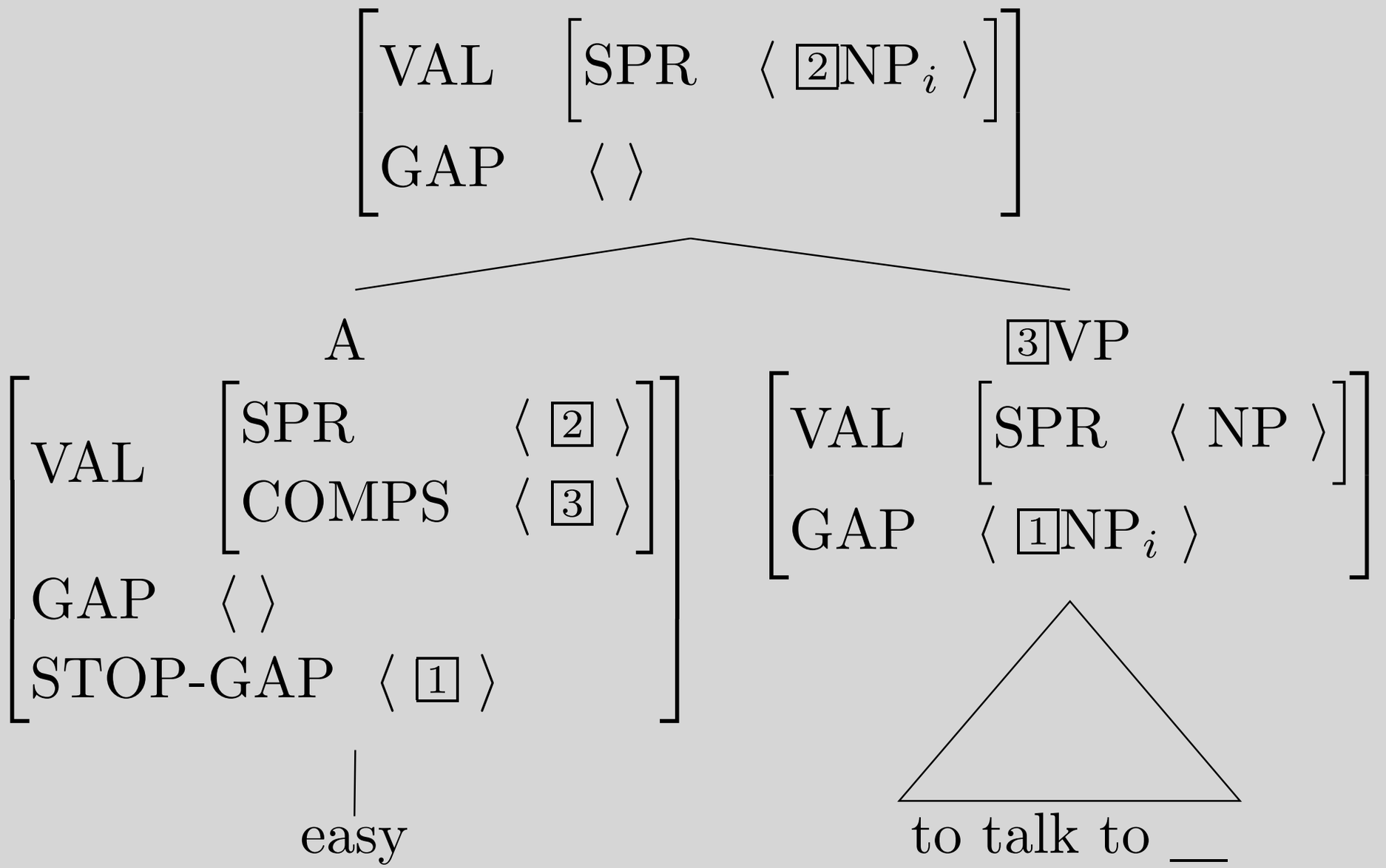
Semantics for *Kim is easy to talk to*

$$\begin{bmatrix} \text{MODE} & \text{prop} \\ \text{INDEX} & s_1 \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{name} \\ \text{NAME} & \text{Kim} \\ \text{NAMED} & i \end{bmatrix}, \begin{bmatrix} \text{RELN} & \textbf{easy} \\ \text{SIT} & s_1 \\ \text{ARG} & s_2 \end{bmatrix}, \begin{bmatrix} \text{RELN} & \textbf{talk} \\ \text{SIT} & s_2 \\ \text{TALKER} & j \\ \text{AUDIENCE} & i \\ \text{TOPIC} & k \end{bmatrix} \right\rangle \end{bmatrix}$$

STOP-GAP Housekeeping

- Lexical entries with nonempty STOP-GAP values (like *easy*) are rare, so STOP-GAP is by default empty in the lexicon.
- Head-Specifier and Head-Modifier rules need to say [STOP-GAP < >]
- Lexical rules preserve STOP-GAP values.

STOP-GAP on the mother here?



GAP Housekeeping

- The initial symbol must say [GAP < >]. Why?
  - To block **Pat found* and **Chris talked to* as stand-alone sentences.
- The Imperative Rule must propagate GAP values. Why?
  - It's not a headed rule, so the effect of the GAP Principle must be replicated
  - Imperatives can have gaps:
    This book, put on the top shelf!

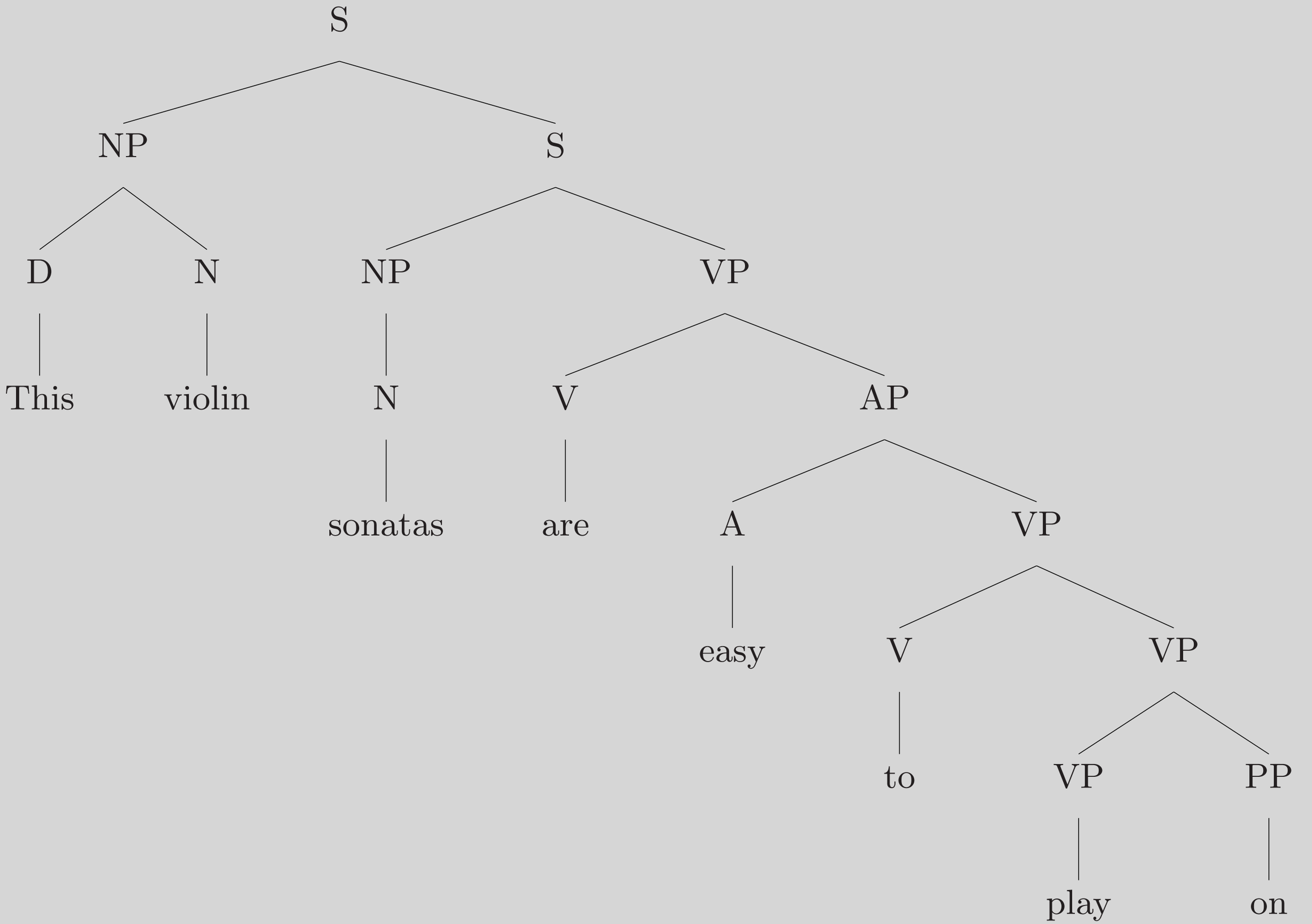
Sentences with Multiple Gaps

- Famous examples:

  This violin, sonatas are easy to play___ on___.

  **Sonatas, this violin is easy to play___ on___.*

- Our analysis gets this:
  - The subject of *easy* is coindexed with the **first** element of the GAP list.
  - The Head-Filler rule only allows one GAP remaining.
- There are languages that allow multiple gaps more generally.



Where We Are

- filler-gap structures:

  The solution to this problem, nobody understood____

  That problem is easy to understand____

- The feature GAP encodes information about missing constituents
- Modified ARP allows arguments that should be on the COMPS list to show up in the GAP list
- GAP values are passed up the tree by the GAP Principle

Where We Are (continued)

- The feature STOP-GAP signals where GAP passing should stop
- The Head-Filler Rule matches a filler to a GAP and (via STOP-GAP) empties GAP
- Lexical entries for *easy*-adjectives require a gap in the complement, coindex the subject with the gap, and (via STOP-GAP) empty GAP on the mother

On to New Material….

- Sentences with subject gaps
- Gaps in coordinate constructions

Subject Gaps

- The ARP revision only allowed missing complements.
- But gaps occur in subject position, too:

  This problem, everyone thought ___ was too easy.
- We handle these via a lexical rule that, in effect, moves the contents of the SPR list into the GAP list

The Subject Extraction Lexical Rule

$$\left[\begin{array}{l} \textit{pi-rule} \\ \\ \text{INPUT} \quad \left\langle \text{X} , \left[\begin{array}{ll} \text{SYN} & \left[\begin{array}{ll} \text{HEAD} & \left[\begin{array}{ll} \textit{verb} & \\ \text{FORM} & \text{fin} \end{array}\right] \\ \text{VAL} & \left[\text{SPR} \quad \langle\, \text{Z} \,\rangle\right] \end{array}\right] \\ \text{ARG-ST} & \boxed{\text{A}} \end{array}\right] \right\rangle \\ \\ \text{OUTPUT} \quad \left\langle \text{Y} , \left[\begin{array}{ll} \text{SYN} & \left[\begin{array}{ll} \text{VAL} & \left[\text{SPR} \quad \langle\ \rangle\right] \\ \text{GAP} & \langle\, \boxed{1} \,\rangle \end{array}\right] \\ \text{ARG-ST} & \boxed{\text{A}}\langle\, \boxed{1} \,, \ldots \rangle \end{array}\right] \right\rangle \end{array}\right]$$

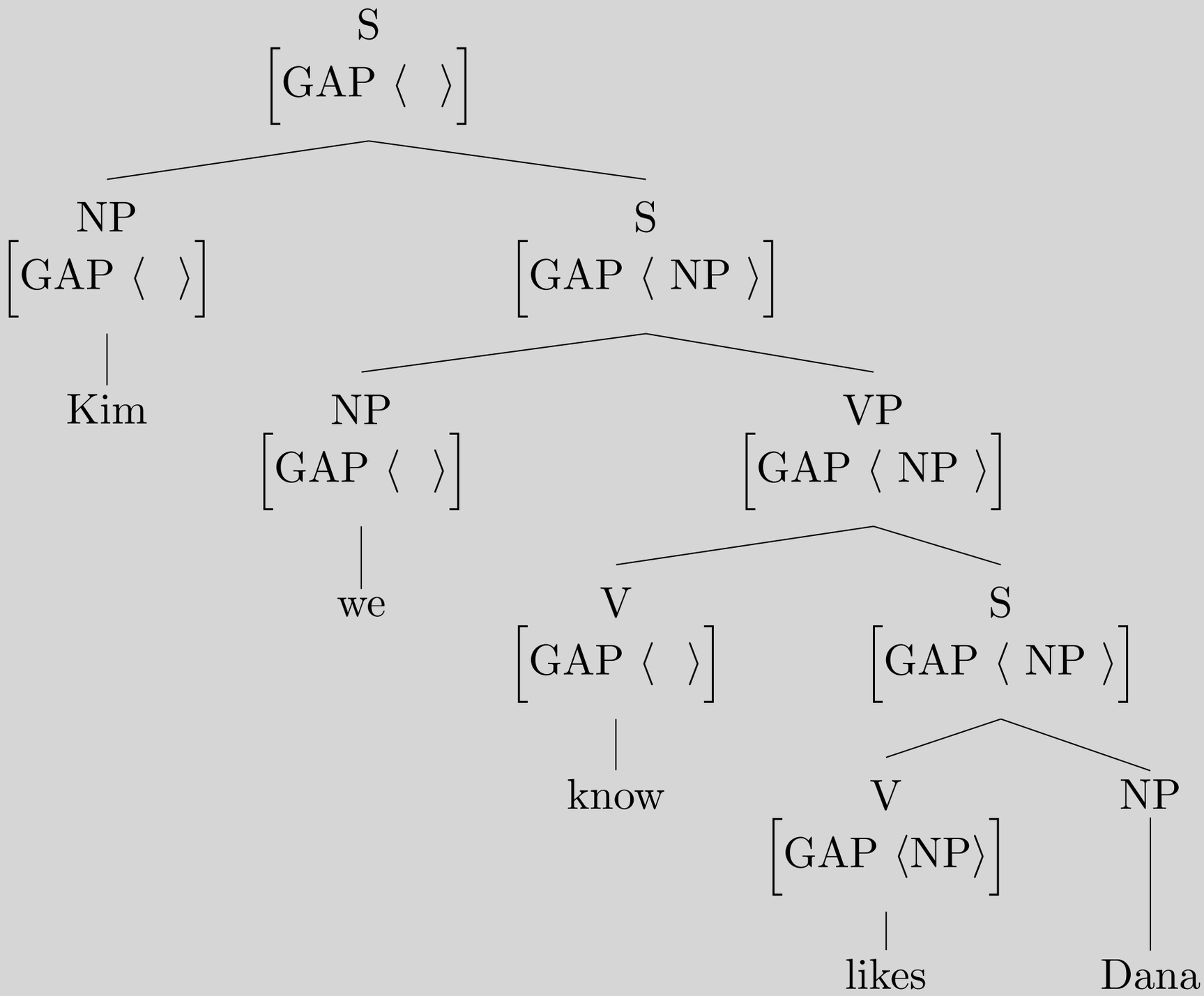
- NB: This says nothing about the phonology, because the default for *pi-rule*s is to leave the phonology unchanged.

A Lexical Sequence This Licenses

$$\left\langle \text{likes} \;,\; \left[\begin{array}{ll} \textit{word} & \\ \text{SYN} & \left[\begin{array}{ll} \text{HEAD} & \left[\begin{array}{ll} \textit{verb} & \\ \text{FORM} & \text{fin} \end{array}\right] \\ \text{VAL} & \left[\begin{array}{ll} \text{SPR} & \langle\,\rangle \\ \text{COMPS} & \langle\,\boxed{2}\,\rangle \end{array}\right] \\ \text{GAP} & \left\langle \boxed{1}\left[\begin{array}{ll} \text{CASE} & \text{nom} \\ \text{AGR} & \textit{3sing} \end{array}\right] \right\rangle \\ \text{STOP-GAP} & \langle\,\rangle \end{array}\right] \\ \text{ARG-ST} & \langle\,\boxed{1}\,,\,\boxed{2}\text{NP[acc]}\,\rangle \end{array}\right] \right\rangle$$

- Note that the ARP is satisfied

A Tree with a Subject Gap



Island Constraints

- There are configurations that block filler-gap dependencies, sometimes called “islands”
- Trying to explain them has been a central topic of syntactic research since the mid 1960s
- We’ll look at just one, Ross’s so-called “Coordinate Structure Constraint”
- Loose statement of the constraint: a constituent outside a coordinate structure cannot be the filler for a gap inside the coordinate structure.

Coordinate Structure Constraint Examples

**This problem, nobody finished the extra credit and____*

**This problem, nobody finished____ and the extra credit.*

**This problem, nobody finished ___ and started the extra credit.*

**This problem, nobody started the extra credit and finished____*

- But notice:

This problem, everybody started____ and nobody finished ____

The Coordinate Structure Constraint

- In a coordinate structure,
  - no conjunct can be a gap (conjunct constraint), and
  - no gap can be contained in a conjunct if its filler is outside of that conjunct (element constraint)

- …..unless each conjunct has a gap that is paired with the same filler (across-the-board exception)

These observations cry out for explanation

- In our analysis, the conjunct constraint is an immediate consequence: individual conjuncts are not on the ARG-ST list of any word, so they can't be put on the GAP list

- The element constraint and ATB exception suggest that GAP is one of those features (along with VAL and FORM) that must agree across conjuncts.

- Note: There is no ATB exception to the conjunct constraint.

  **This problem, you can compare only____ and____.*

Our Coordination Rule, so far

$$\begin{bmatrix} \text{FORM} & \boxed{1} \\ \text{VAL} & \boxed{0} \\ \text{IND} & s_0 \end{bmatrix} \rightarrow$$

$$\begin{bmatrix} \text{FORM} & \boxed{1} \\ \text{VAL} & \boxed{0} \\ \text{IND} & s_1 \end{bmatrix} .... \begin{bmatrix} \text{FORM} & \boxed{1} \\ \text{VAL} & \boxed{0} \\ \text{IND} & s_{n-1} \end{bmatrix} \begin{bmatrix} \text{HEAD} & \textit{conj} \\ \text{IND} & s_0 \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{ARGS} & \langle s_1 .... s_n \rangle \end{bmatrix} \right\rangle \end{bmatrix} \begin{bmatrix} \text{FORM} & \boxed{1} \\ \text{VAL} & \boxed{0} \\ \text{IND} & s_n \end{bmatrix}$$

- Recall that we have tinkered with what must agree across conjuncts at various times.
- Now we'll add GAP to the things that conjuncts must share

Our Final Coordination Rule

$$\begin{bmatrix} \text{FORM} & \boxed{1} \\ \text{VAL} & \boxed{0} \\ \text{GAP} & \boxed{A} \\ \text{IND} & s_0 \end{bmatrix} \rightarrow$$

$$\begin{bmatrix} \text{FORM} & \boxed{1} \\ \text{VAL} & \boxed{0} \\ \text{GAP} & \boxed{A} \\ \text{IND} & s_1 \end{bmatrix} \begin{bmatrix} \text{FORM} & \boxed{1} \\ \text{VAL} & \boxed{0} \\ \text{GAP} & \boxed{A} \\ \text{IND} & s_{n-1} \end{bmatrix} \begin{bmatrix} \text{HEAD} & conj \\ \text{IND} & s_0 \\ \text{RESTR} & \langle \begin{bmatrix} \text{ARGS} & \langle s_1 .... s_n \rangle \end{bmatrix} \rangle \end{bmatrix} \begin{bmatrix} \text{FORM} & \boxed{1} \\ \text{VAL} & \boxed{0} \\ \text{GAP} & \boxed{A} \\ \text{IND} & s_n \end{bmatrix}$$

- We've just added GAP to all the conjuncts and the mother.
- This makes the conjuncts all have the same gap (if any)
- Why do we need it on the mother?

Closing Remarks on LDDs

- This is a huge topic; we've only scratched the surface
  - There are many more kinds of LDDs, which would require additional grammar rules
    - There are also more island constraints, which also need to be explained
- Our account of the coordinate structure constraint (based on ideas of Gazdar) is a step in the right direction, but it would be nice to explain why certain features must agree across conjuncts.

Overview

- Some examples of the phenomenon
- What is new and different about it
- Brief sketch of the TG approach
- Broad outlines of our approach
- Details of our approach
- Subject extraction
- Coordinate Structure Constraint

RQs: Empty elements

- What's so bad about empty elements? Isn't movement a simpler analysis here?
- Why should GAP elements be on the ARG-ST?
- What do these choices show about where "movement-like" effects are encoded in the grammar (lexicon vs phrase structure vs features), and why might this architecture be preferable to simply positing empty elements in the syntax?

RQs: Easy adjectives

- What other adjectives show up in this class?
- As of 2018, the ERG listed 215.
- *alarming, boring, convenient, cheap, dangerous, entertaining, fruitless, gross, hazardous, illegal, kind, logical, motivating, natural, optional, prudent, risky, sobering, tedious, useful, valuable, wise, …*

RQs: STOP-GAP

- It is clear for the GAP part. For STOP-GAP, I would like to confirm my understanding. Is STOP-GAP can maximum have 1 element? Page 437, “as indicated by the head daughter’s STOP-GAP”, does this mean it only appears in head daughter’s node? Why is it not on mother node or other node?

RQs: On and on

- For pg. 429 I was wondering about the grammaticality of "On this, you can rely on." because I find this grammatical and allowable by my grammar and was wondering about how this interacted with the solution to exercise 1.

RQs: Subject extraction

- Why do we need a separate Subject Extraction Lexical Rule instead of just using the Argument Realization Principle (ARP)?

RQs: Cross-linguistic

- Is this approach for GAP features also done in other language grammars? Or specific to our analysis of English/just in the ERG?