CS 182
Sections 103 - 104

slide credit to

Eva Mok and Joe Makin

Updated by Leon Barrett

April 25, 2007

Announcements

- a9 due Tuesday, May 1st, in class
- final exam Tuesday May 8th in class
- final paper due Friday May 11th, 11:59pm
- final review sometime next week

Schedule

- Last Week
  - Constructions, ECG
- This Week
  - Models of language learning
  - Embodied Construction Grammar learning
- Next Week
  - Open lecture
  - Wrap-Up

Questions

1. Why is learning language difficult?
2. What are the “paucity of the stimulus” and the “opulence of the substrate”?
3. What is Gold's Theorem?
4. How does the analyzer use the constructions to parse a sentence?
5. How can we learn new ECG constructions?
6. What are ways to re-organize and consolidate the current grammar?
7. What metric is used to determine when to form a new construction?

Difficulty of learning language

- What makes learning language difficult?
  - How many sentences do children hear?
  - How often are those sentences even correct?
  - Even when they're correct, how often are they complete?
  - How often are they corrected when saying something wrong?
  - How many possible languages are there?

Larger context

- War!
  - Is language innate?
  - Covered in book

Questions

1. Why is learning language difficult?
2. What are the “paucity of the stimulus” and the “opulence of the substrate”?
3. What is Gold's Theorem?
4. How does the analyzer use the constructions to parse a sentence?
5. How can we learn new ECG constructions?
6. What are ways to re-organize and consolidate the current grammar?
7. What metric is used to determine when to form a new construction?

Paucity and Opulence

- Poverty of the stimulus
  - Coined to suggest how little information children have to learn from
- Opulence of the substrate
  - Opulence = "richness"
  - Coined in response to suggest how much background information children have

Questions

1. Why is learning language difficult?
2. What are the “paucity of the stimulus” and the “opulence of the substrate”?
3. What is Gold's Theorem?
4. How does the analyzer use the constructions to parse a sentence?
5. How can we learn new ECG constructions?
6. What are ways to re-organize and consolidate the current grammar?
7. What metric is used to determine when to form a new construction?

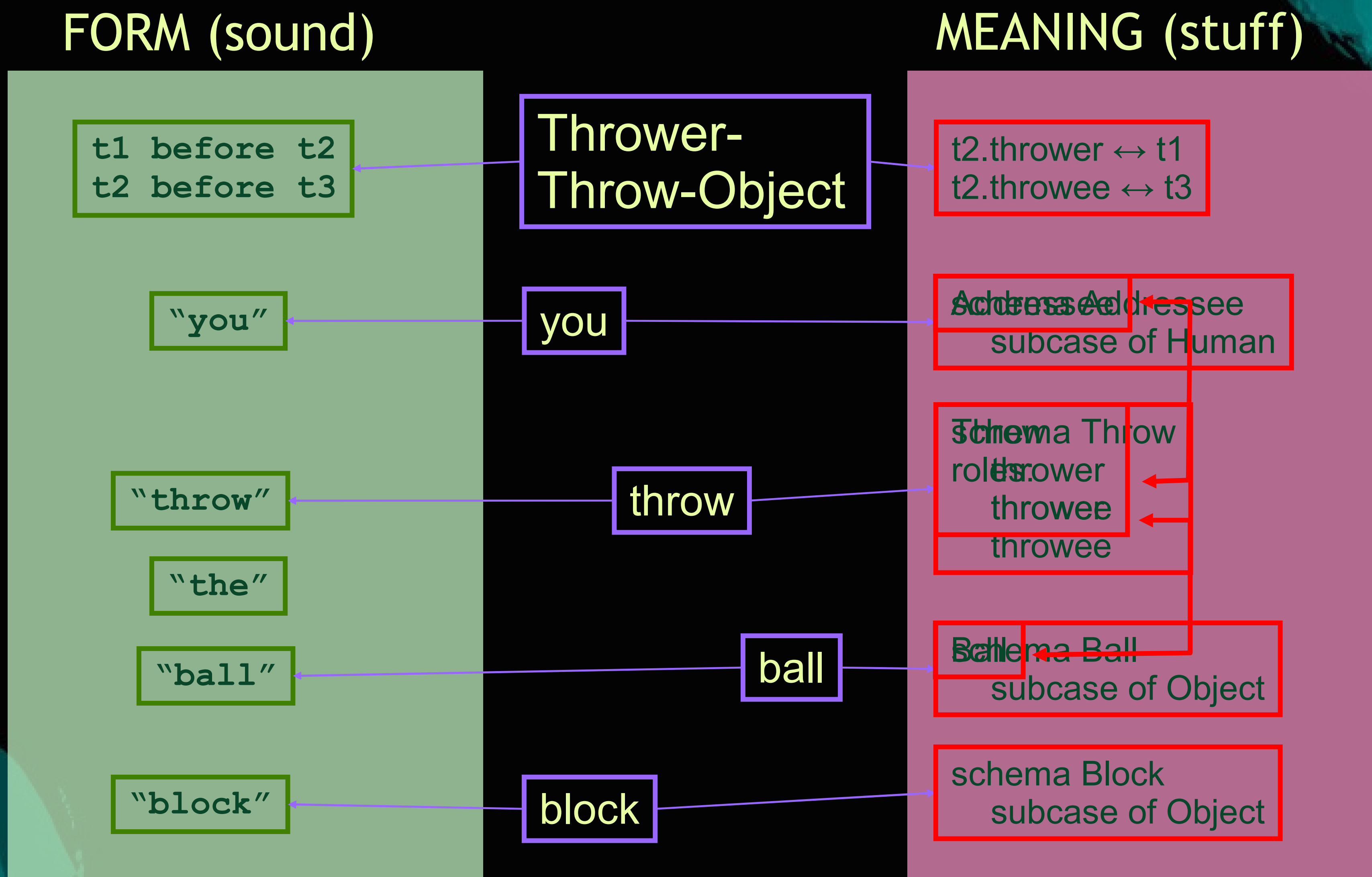
Gold's Theorem

- Suppose that you have an infinite number of languages
  - language = “set of legal sentences”
- Suppose that for every language Ln, there is a bigger language Ln+1
  - makes every sentence, and then some
- There's some language Linfinity
  - contains all the sentences in all other grammars
- You can arrange data so that no one ever learns Linfinity
  - http://www.lps.uci.edu/~johnsonk/Publications/Johnson.GoldsTheorem.pdf

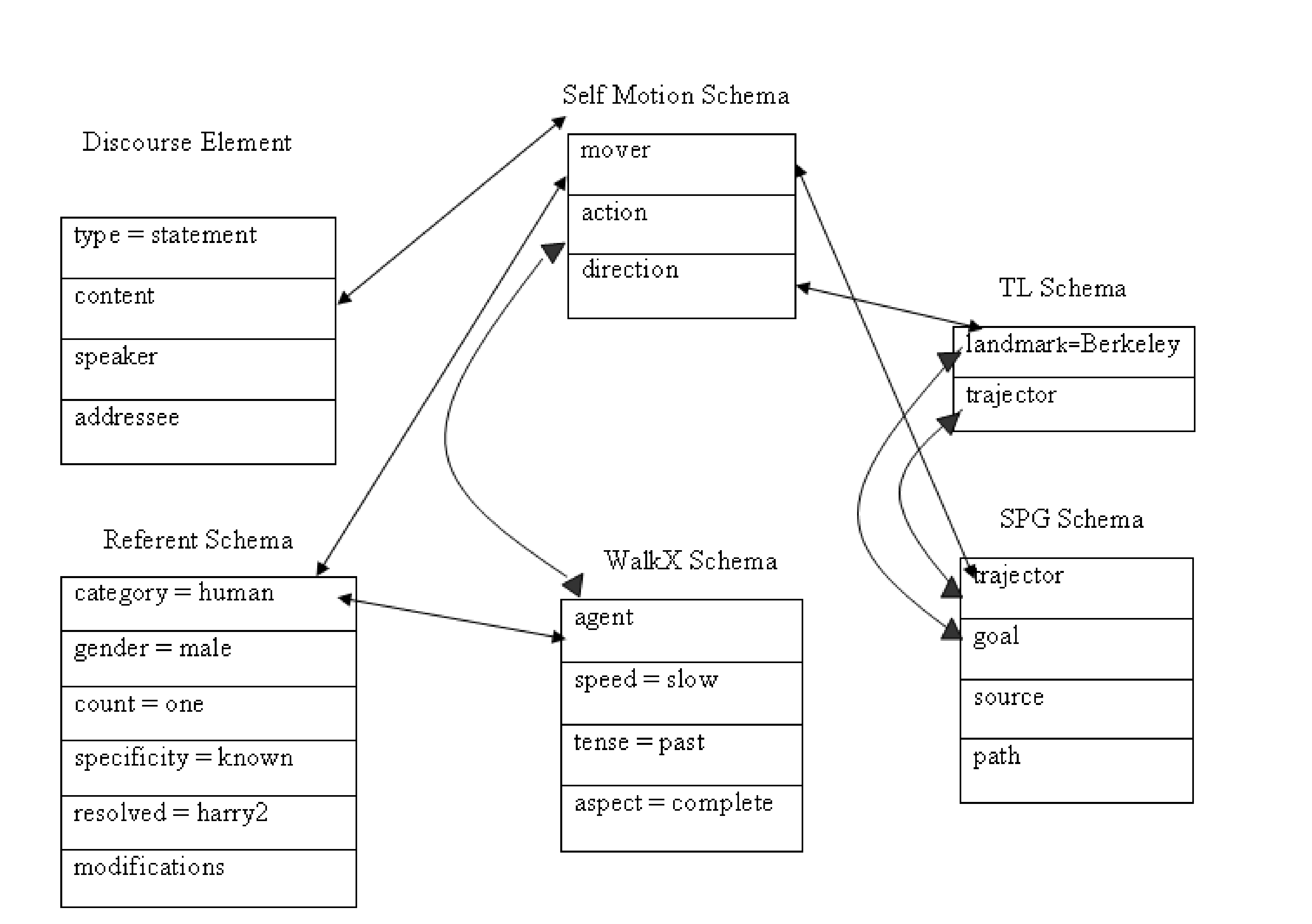
Questions

1. Why is learning language difficult?
2. What are the “paucity of the stimulus” and the “opulence of the substrate”?
3. What is Gold's Theorem?
4. How does the analyzer use the constructions to parse a sentence?
5. How can we learn new ECG constructions?
6. What are ways to re-organize and consolidate the current grammar?
7. What metric is used to determine when to form a new construction?

Analyzing "You Throw The Ball"



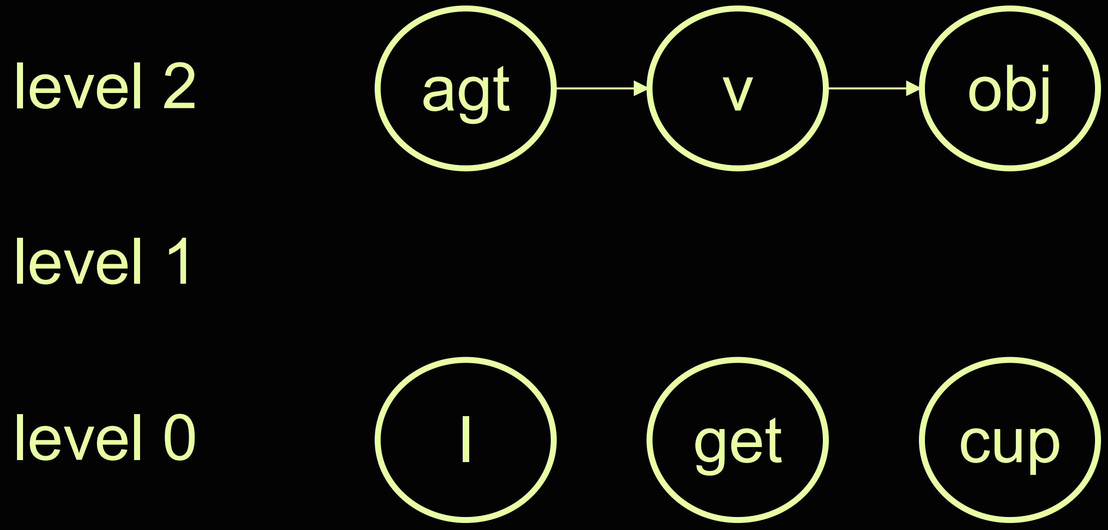
Another way to think of the SemSpec



Analyzing in ECG

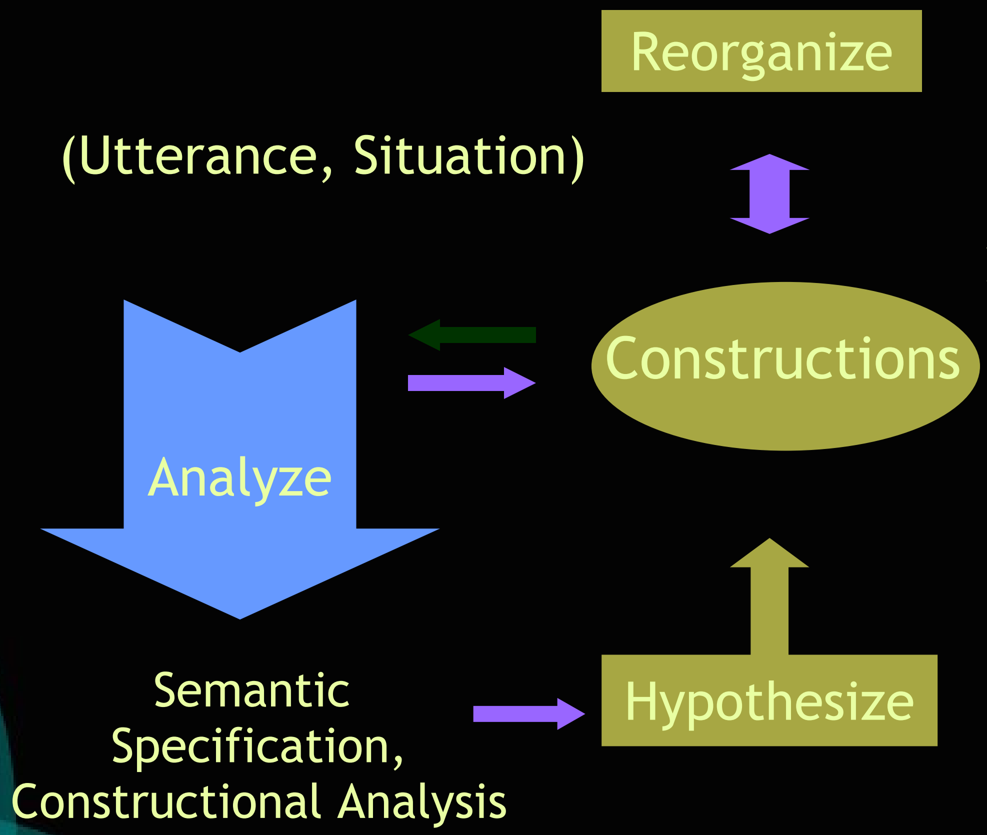
```
create a recognizer for each construction in the grammar
for each level j (in ascending order)
    repeat
        for each recognizer r in j
            for each position p of utterance
                initiate r starting at p
    until we don't find anything new
```

Recognizer for the Transitive-Cn



- an example of a level-1 construction is Red-Ball-Cn
- each recognizer looks for its constituents in order (the ordering constraints on the constituents can be a partial ordering)

Learning-Analysis Cycle (Chang, 2004)



1. Learner passes input (Utterance + Situation) and current grammar to Analyzer.

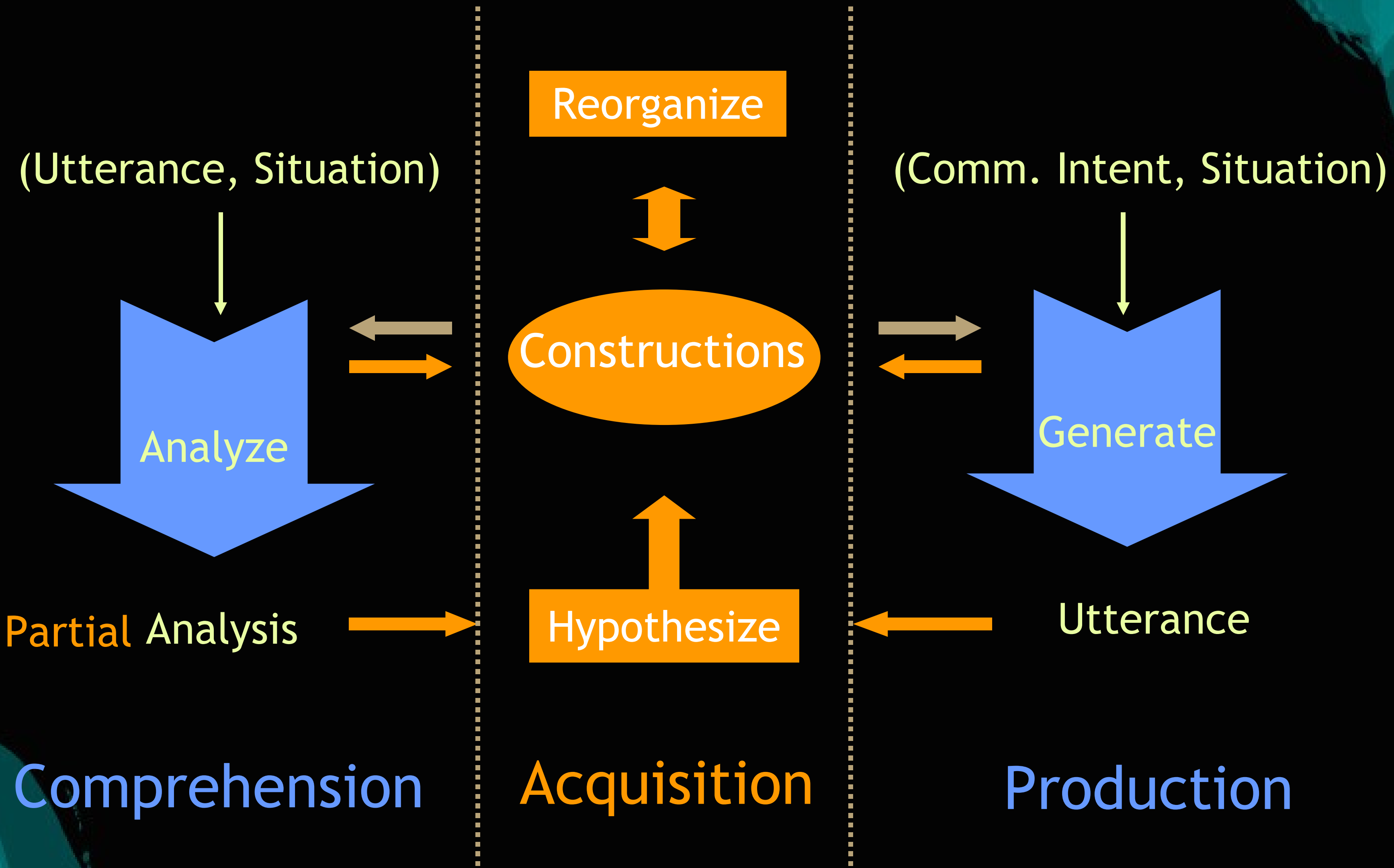
2. Analyzer produces SemSpec and Constructional Analysis.

1. Learner updates grammar:
   a. Hypothesize new map.
   b. Reorganize grammar (merge or compose).
   c. Reinforce (based on usage).

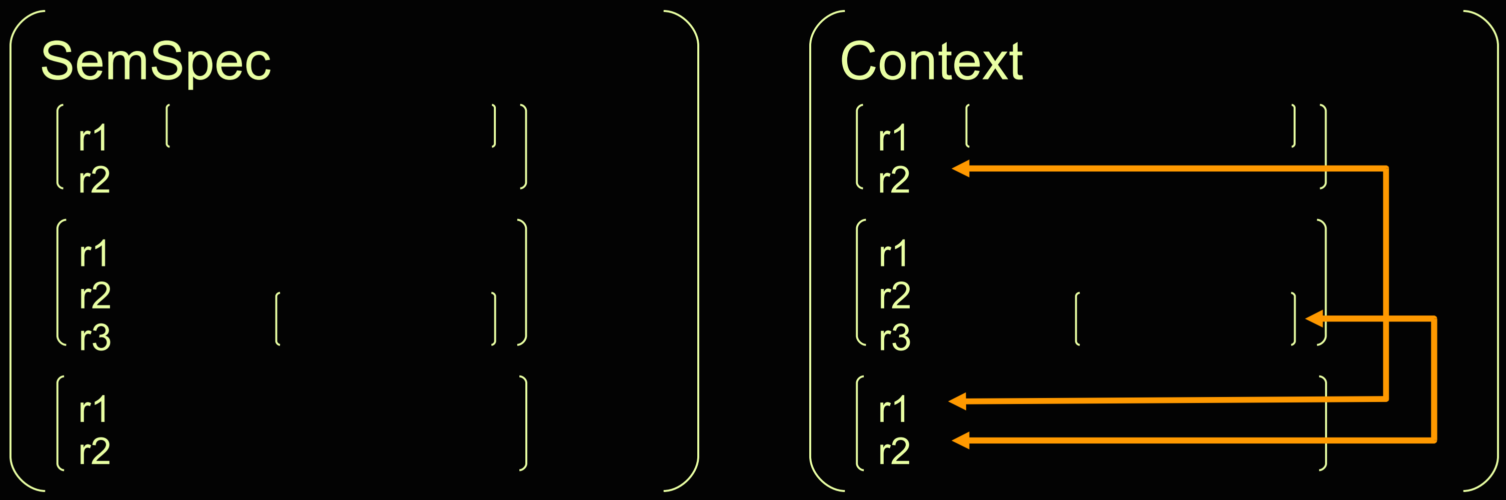
Questions

1. Why is learning language difficult?
2. What are the “paucity of the stimulus” and the “opulence of the substrate”?
3. What is Gold's Theorem?
4. How does the analyzer use the constructions to parse a sentence?
5. How can we learn new ECG constructions?
6. What are ways to re-organize and consolidate the current grammar?
7. What metric is used to determine when to form a new construction?

Usage-based Language Learning

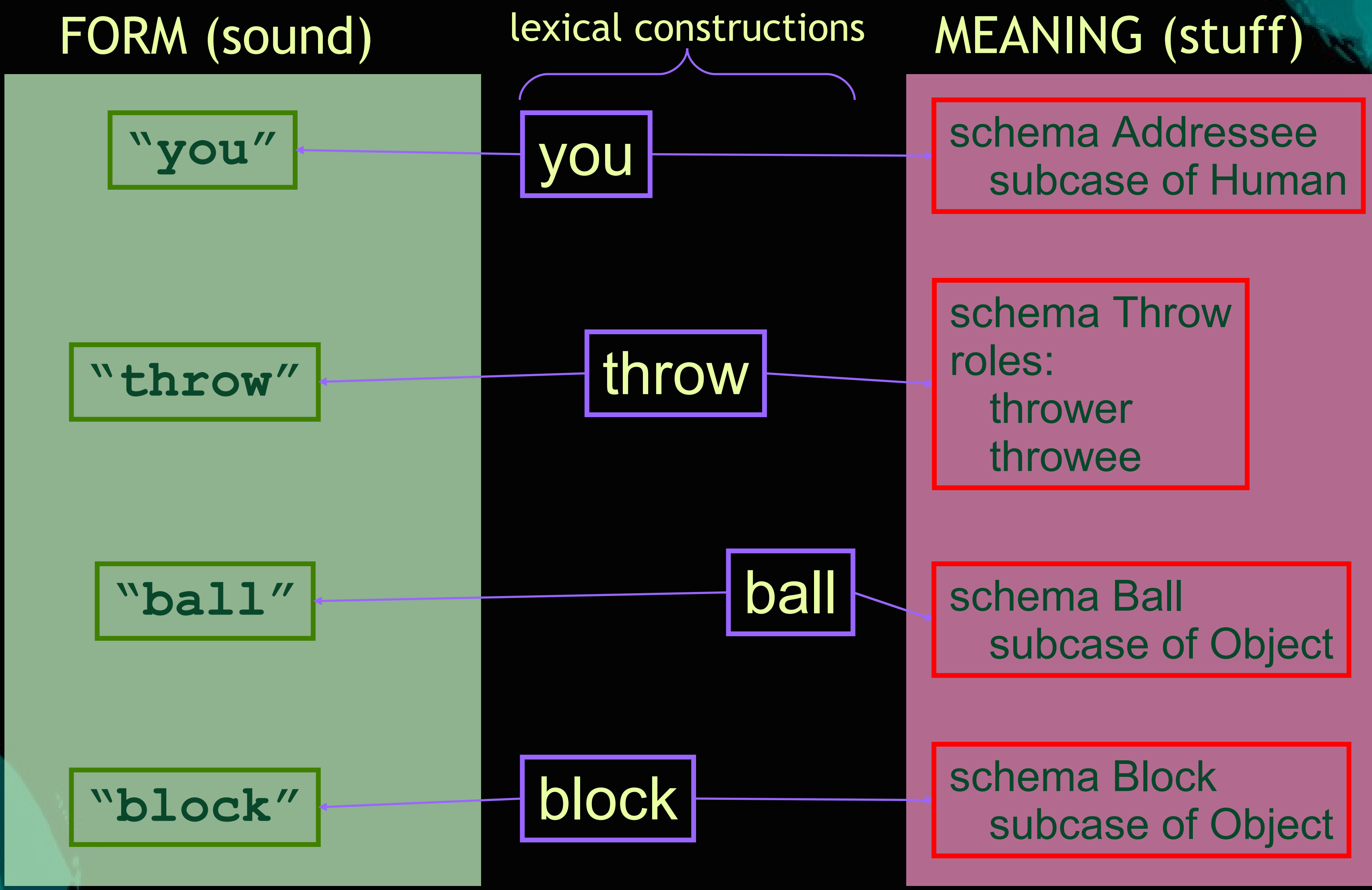


Basic Learning Idea

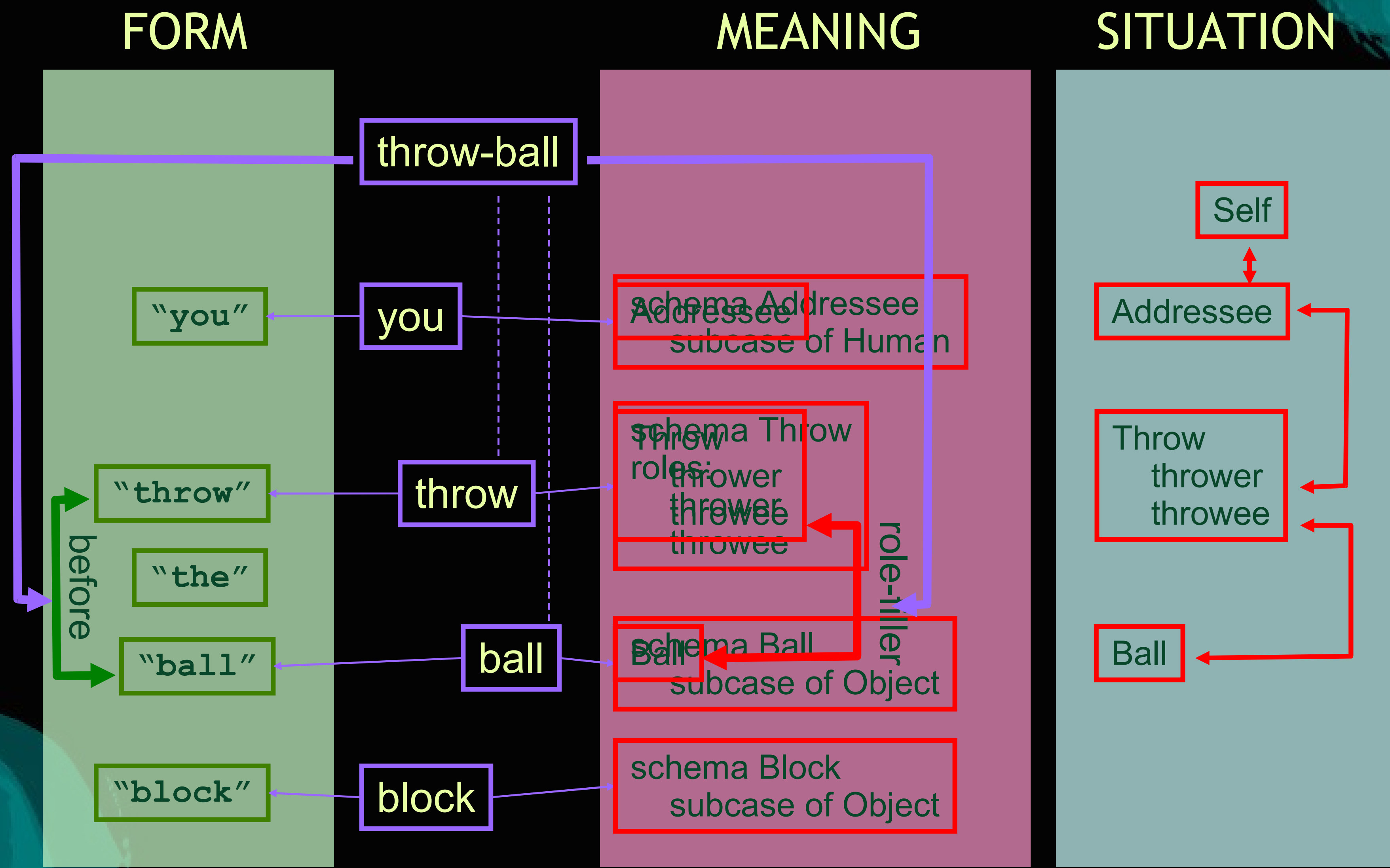


- The learner's current grammar produces a certain analysis for an input sentence
- The context contains richer information (e.g. bindings) that are unaccounted for in the analysis
- Find a way to account for these meaning relations (by looking for corresponding form relations)

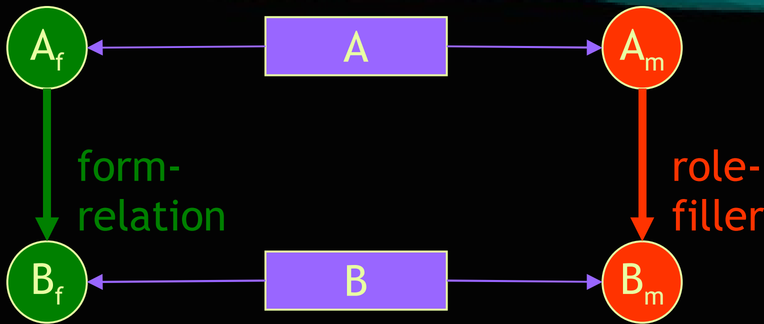
Initial Single-Word Stage



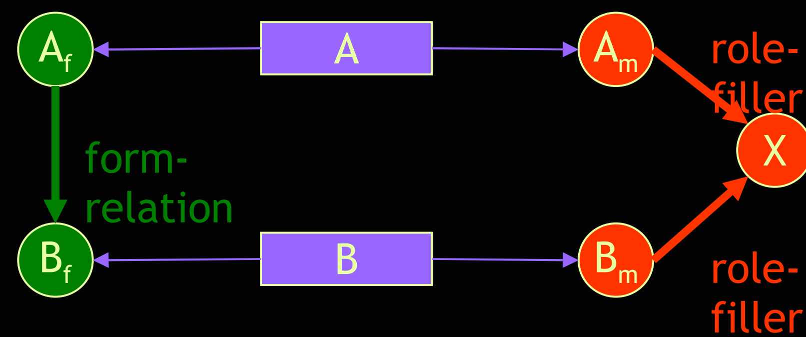
New Data: “You Throw The Ball”



Relational Mapping Scenarios

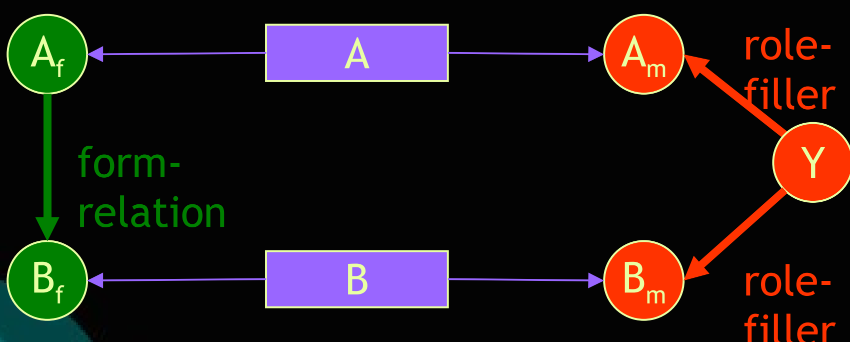


throw ball throw.throwee ↔ ball



put ball **down**

put.mover ↔ ball
down.tr ↔ ball



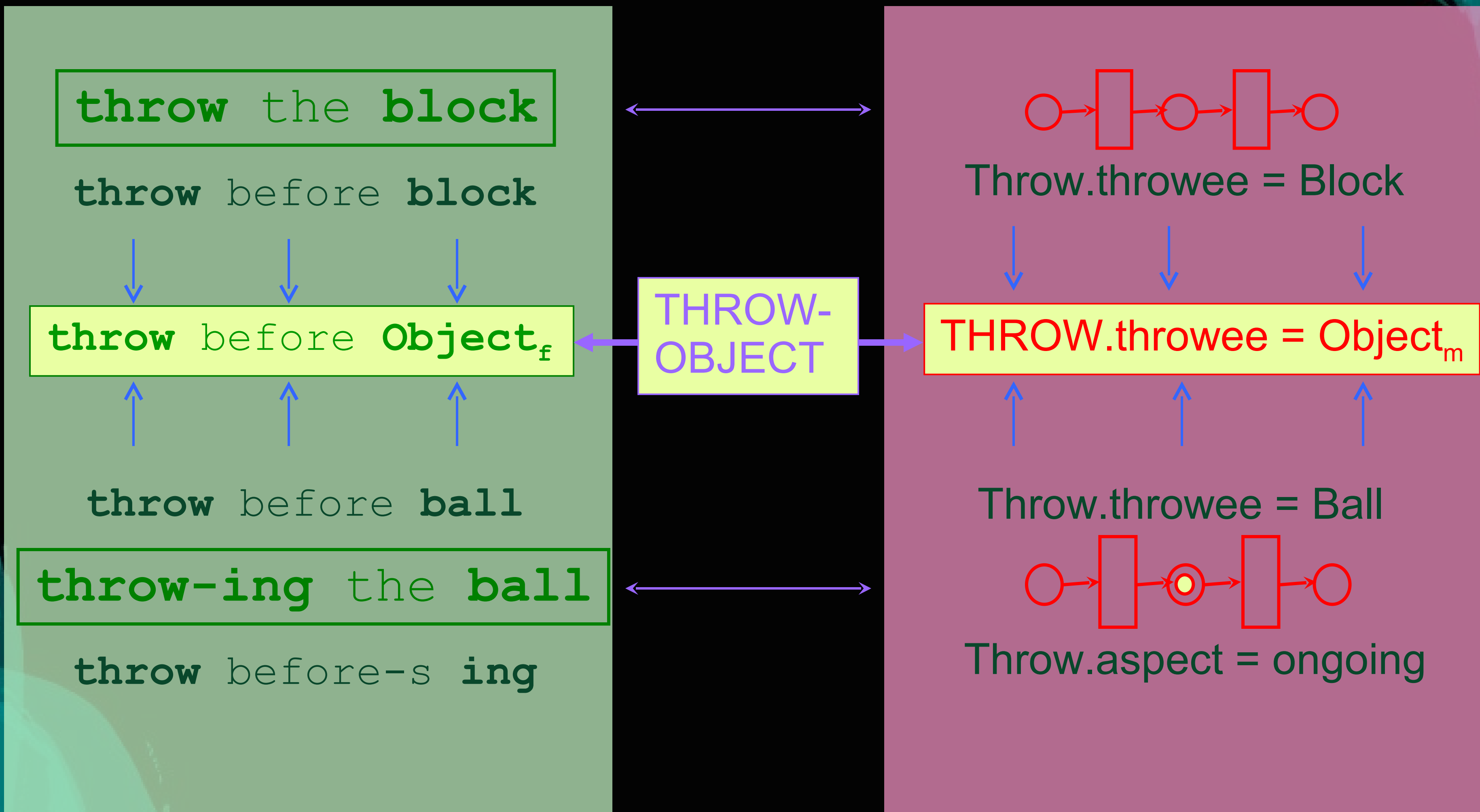
Nomi ball

possession.possessor ↔ Nomi
possession.possessed ↔ ball

Questions

1. Why is learning language difficult?
2. What are the “paucity of the stimulus” and the “opulence of the substrate”?
3. What is Gold's Theorem?
4. How does the analyzer use the constructions to parse a sentence?
5. How can we learn new ECG constructions?
6. What are ways to re-organize and consolidate the current grammar?
7. What metric is used to determine when to form a new construction?

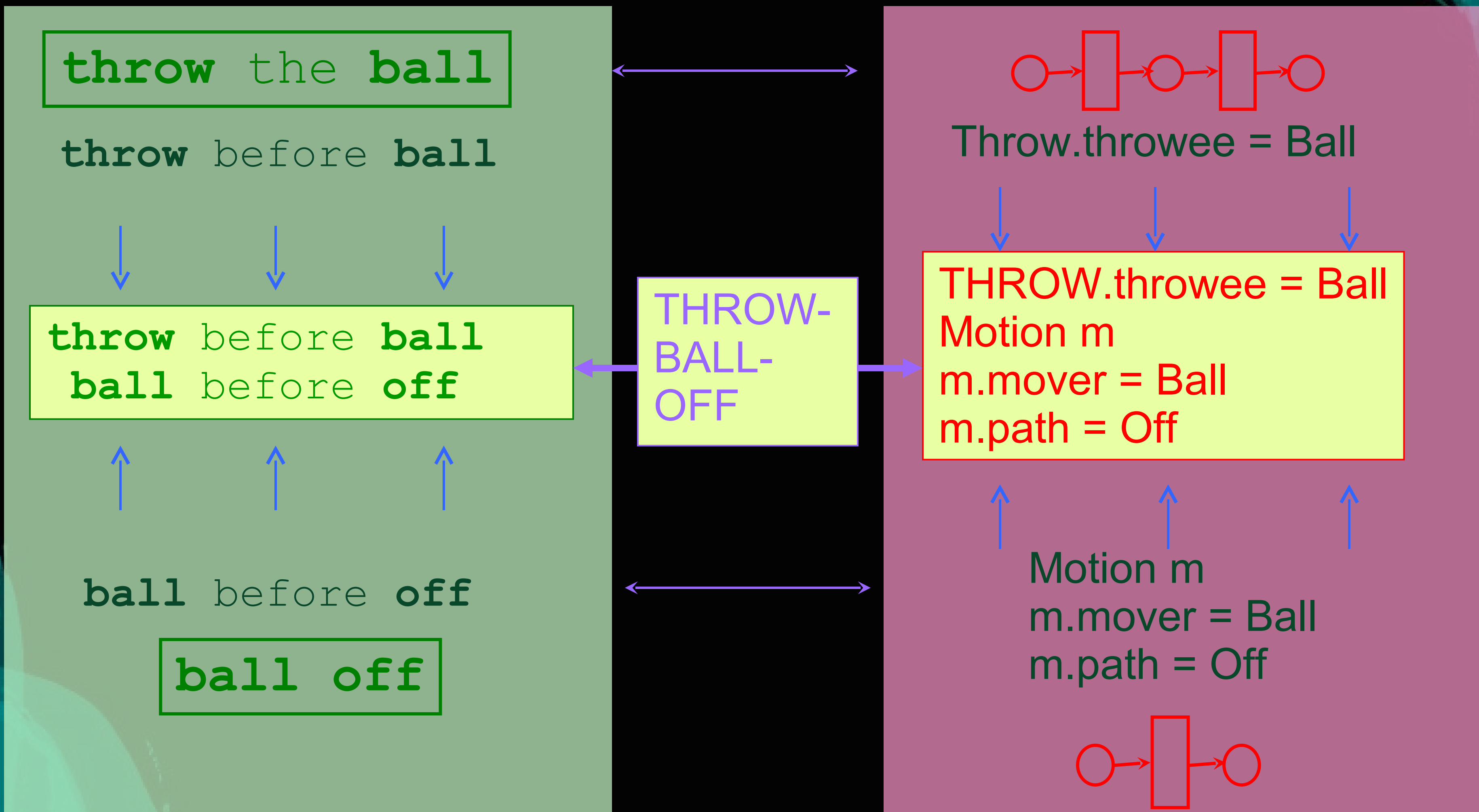
Merging Similar Constructions



Resulting Construction

construction THROW-OBJECT
constructional
constituents
t : THROW
o : OBJECT
form
t_f before o_f
meaning
t_m.throwee $\leftrightarrow$ o_m

Composing Co-occurring Constructions



Resulting Construction

construction THROW-BALL-OFF
constructional
constituents
t : THROW
b : BALL
o : OFF
form
t_f before b_f
b_f before o_f
meaning
evokes MOTION **as** m
t_m.throwee ↔ b_m
m.mover ↔ b

Questions

1. Why is learning language difficult?
2. What are the "paucity of the stimulus" and the "opulence of the substrate"?
3. What is Gold's Theorem?
4. How does the analyzer use the constructions to parse a sentence?
5. How can we learn new ECG constructions?
6. What are ways to re-organize and consolidate the current grammar?
7. What metric is used to determine when to form a new construction?

Minimum Description Length

- Choose grammar G to minimize cost(G|D):
  - cost(G|D) = α • size(G) + β • complexity(D|G)
  - Approximates Bayesian learning;
    cost(G|D) ≈ 1/posterior probability ≈ 1/P(G|D)
- Size of grammar = size(G) ≈ 1/prior ≈ 1/P(G)
  - favor fewer/smaller constructions/roles; isomorphic mappings
- Complexity of data given grammar ≈ 1/likelihood
  ≈ 1/P(D|G)
  - favor simpler analyses
    (fewer, more likely constructions)
  - based on derivation length + score of derivation

Size Of Grammar

- Size of the grammar *G* is the sum of the size of each construction:

$$\text{size}(G) = \sum_{c \in G} \text{size}(c)$$

- Size of each construction *c* is:

$$\text{size}(c) = n_c + m_c + \sum_{e \in c} \text{length}(e)$$

where

- n_c = number of constituents in *c*,
- m_c = number of constraints in *c*,
- length(*e*) = slot chain length of element reference *e*

Example: The Throw-Ball Cxn

construction THROW-BALL
constructional
constituents
t **:** THROW
b **:** BALL
form
t_f before b_f
meaning
t_m.throwee ↔ b_m

$$\text{size}(c) = n_c + m_c + \sum_{e \in c} \text{length}(e)$$

size(THROW-BALL)
= 2 + 2 + (2 + 3) = 9

Complexity of Data Given Grammar

- Complexity of the data *D* given grammar *G* is the sum of the analysis score of each input token *d*:

$$\text{complexity}(D \mid G) = \sum_{d \in D} \text{score}(d)$$

- Analysis score of each input token *d* is:

$$\text{score}(d) = \sum_{c \in d} \left(weight_c + \eta \cdot \sum_{r \in c} |type_r| \right) + height_d + semfit_d$$

where

- *c* is a construction used in the analysis of *d*
- $weight_c$ ≈ relative frequency of *c*,
- $|type_r|$ = number of ontology items of type *r* used,
- $height_d$ = height of the derivation graph,
- $semfit_d$ = semantic fit provide by the analyzer

Final Remark

- The goal here is to build a cognitive plausible model of language learning
- A very different game that one could play is unsupervised / semi-supervised language learning using shallow or no semantics
  - statistical NLP
  - automatic extraction of syntactic structure
  - automatic labeling of frame elements
- Fairly reasonable results for use in tasks such as information retrieval, but the semantic representation is very shallow