# Informatics 1 Cognitive Science

## Lecture 10: Word Learning

Frank Keller

6 February 2024

School of Informatics
University of Edinburgh
keller@inf.ed.ac.uk

Slide credits: Frank Mollica, Chris Lucas, Mirella Lapata

# Overview

# Recap

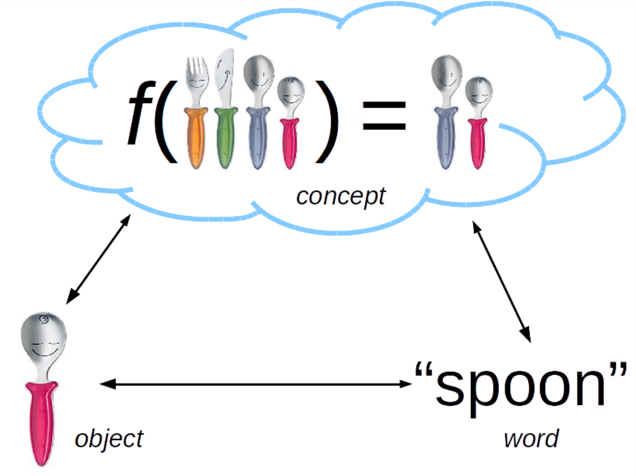
In order to acquire a lexicon young children segment speech into words using multiple sources of support; we focused on distributional regularities:

- transitional probability provides cues to word boundaries
- Minimum Description Length help assembling words into a lexicon
- Bayes Rule is a way of combining prior beliefs with evidence, and updating beliefs in the light of new evidence

In today's lecture we focus on word learning: How do children associate words with concepts?

We'll see a detailed case study on number words. Bayes Rule will again be imporant.

# Word Learning


$f($ $)$ =
concept
“spoon”
object
word

# The Mapping Problem

W. V. O. Quine (1960). *Word and Object.*


A rabbit!

Our dinner!

Shh, be quiet!

What a cute furry thing!

Rabbit parts!

Get it out!

Don't move!

What long ears!

The child does not know which attribute is being labeled!

# The Mapping Problem (Carey & Bartlett, 1978)

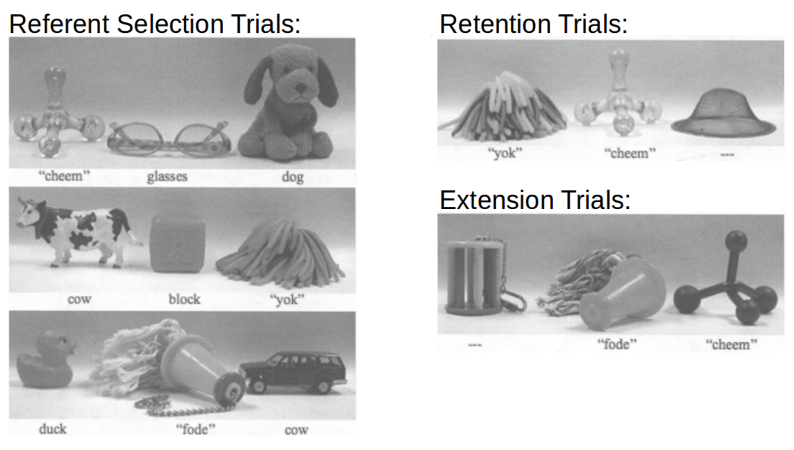
# The Mapping Problem (Carey & Bartlett, 1978)

- **Mutual exclusivity:** an inductive bias that every object has only one name.
- **Fast mapping:** a quick map between a word and an object based on a single observation.

# Fast Mapping (Horst & Samuelson, 2008)

Q1: Do fast mappings last?

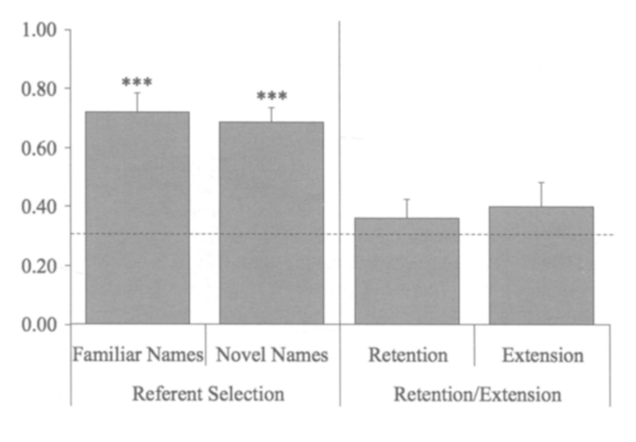
Q2: Do fast mappings also solve the generalization problem?



# Fast Mapping (Horst & Samuelson, 2008)

Q1: Do fast mappings last? A: No.

Q2: Do fast mappings also solve the generalization problem? A: No.

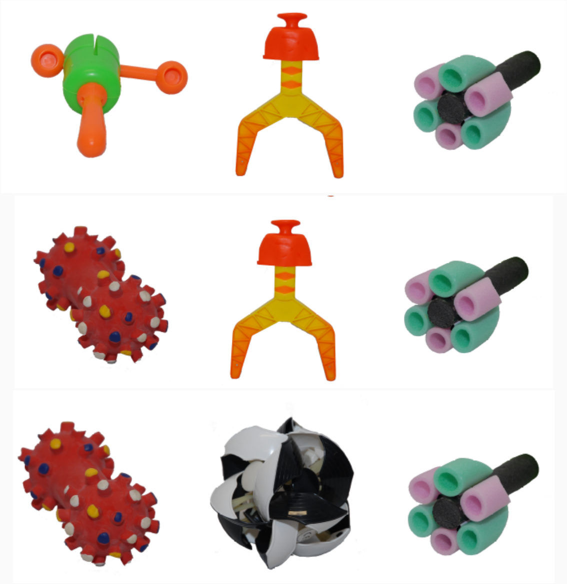


# Cross-situational Word Learning

# Cross-situational Word Learning

# Cross-situational Word Learning

# Cross-situational Word Learning



# Cross-situational Word Learning

Time for a short quiz on Wooclap!

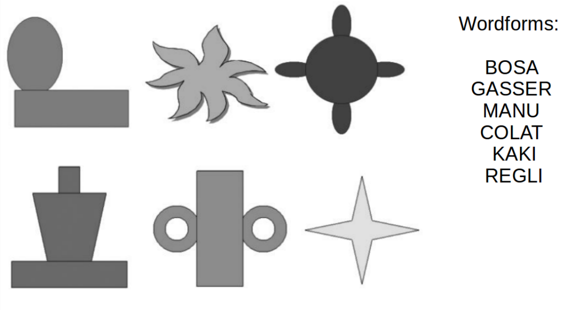
https://app.wooclap.com/PPUKKP

# Cross-situational Word Learning

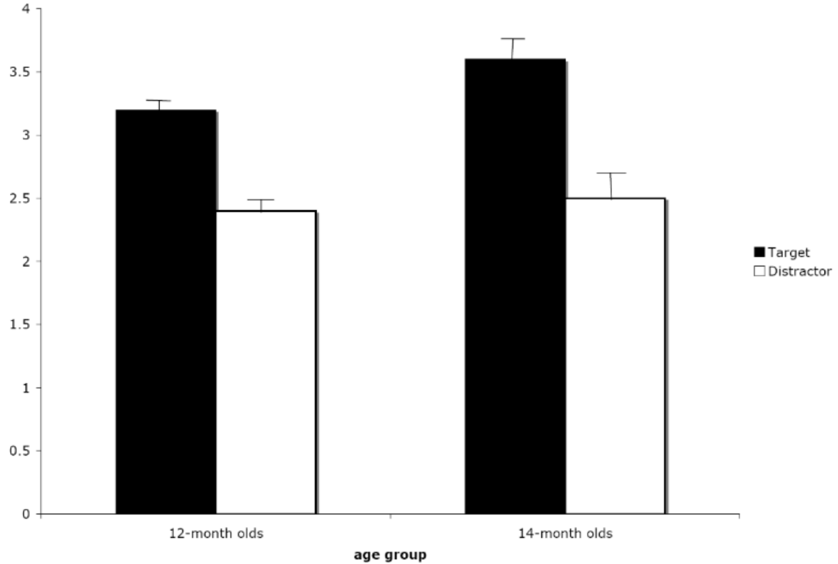
- **Cross-situational word learning:** storing and reasoning about word-object co-occurrence statistics.
- Siskind (1996) showed that cross-situational word learning is sufficient to form the correct word–referent mappings.
- But can infants actually do this? (Smith & Yu, 2008)

# Cross-situational Word Learning

- **Cross-situational word learning:** storing and reasoning about word-object co-occurrence statistics.
- Siskind (1996) showed that cross-situational word learning is sufficient to form the correct word–referent mappings.
- But can infants actually do this? (Smith & Yu, 2008)



# Cross-situational Word Learning



# Interim Summary

Mapping problem:

- We can use inference to create fast mappings between words and objects.
- These fast mappings don't live very long, though.
- We can store cross-situational statistics, which would be sufficient for learning.
- However, it's not clear that we actually do this. It would mean that children have to store and retrieve large amounts of co-occurence statistics.

# Interim Summary

Mapping problem:

- We can use inference to create fast mappings between words and objects.
- These fast mappings don't live very long, though.
- We can store cross-situational statistics, which would be sufficient for learning.
- However, it's not clear that we actually do this. It would mean that children have to store and retrieve large amounts of co-occurence statistics.

This has lead to two competing accounts in the literature:

1. Associative learning: store all the stats and compute an optimal mapping.
2. Hypothesis testing: use stats to test your current mapping; change your hypothesis if required, then discard the stats.

# Modelling Word Learning

# Applying Rational Analysis to Word Learning

- **Goal:** Why are you learning?
- **Model:**
  - *Input:* What information is your model considering?
  - *Output:* What responses are allowed?
  - *Hypothesis Space:* What mappings between input and output are possible?
  - *Inductive Bias:* How does the model perform when there's no data?
  - *Update Rule:* How does the model change as you observe data?
- **Environment:**
  - What constrains the training data?
  - Is the training environment comparable to the environment you hope to achieve your goal in?

# Inductive Bias

"…for any set of data there will be an infinite number of logically possible hypotheses consistent with it. The data are never sufficient logically to eliminate all competing hypotheses." –Ellen Markman

What are the biases that constrain children's word learning?

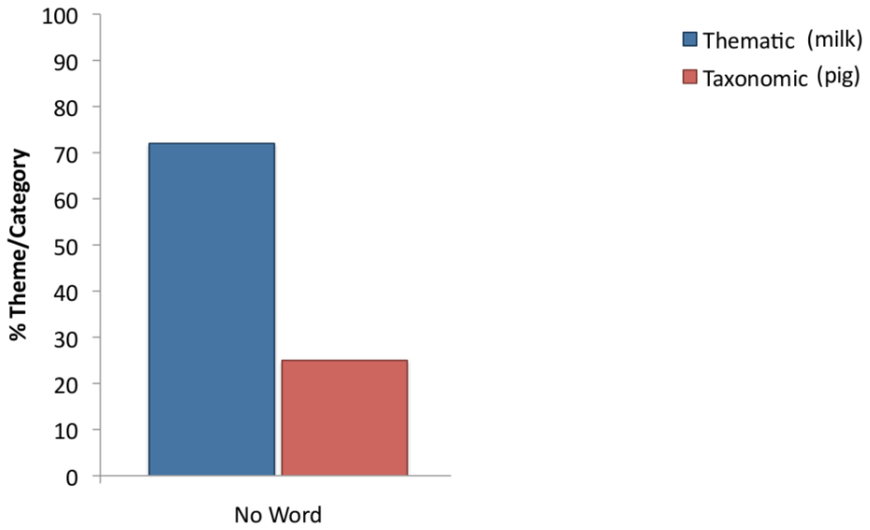
- Mutual exclusivity
- Whole object bias
- Taxonomic bias

# Whole Object Bias


Words refer to the whole object not its parts.

# Taxonomic Bias

# Taxonomic Bias



# Taxonomic Bias

## Exposure

no word: "I'm going to show you something!"

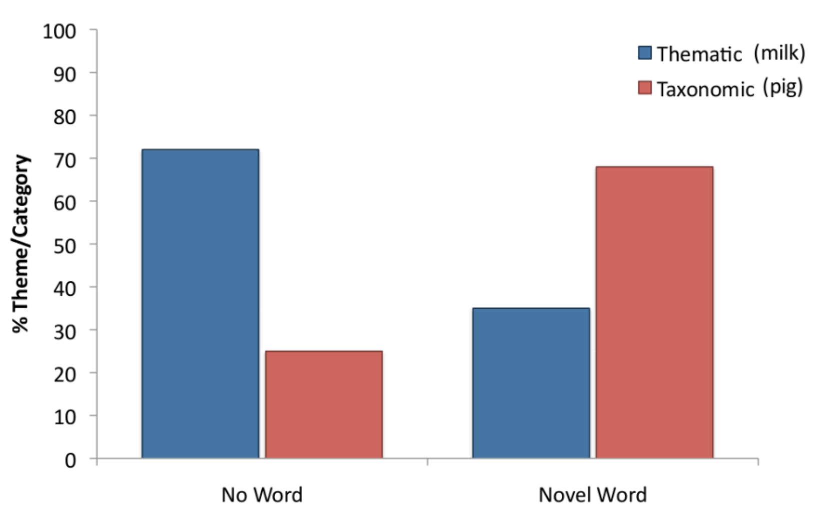
novel word: "I'm going to show you a dax!"

## Testing

"Can you show me another one?"

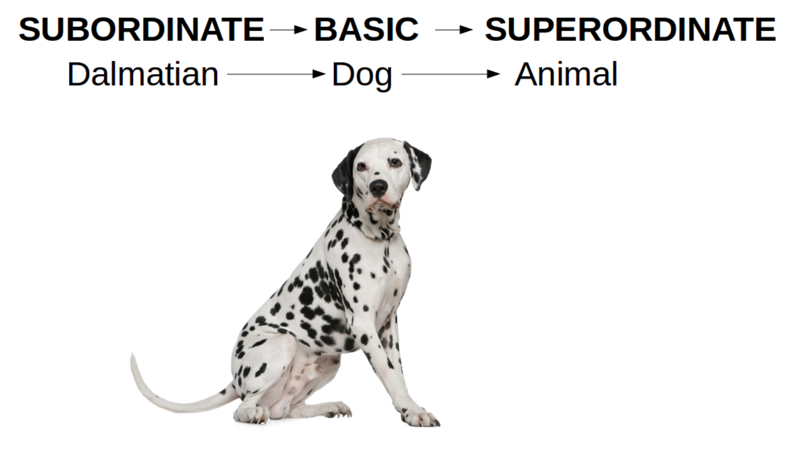
"Can you show me another dax?"

# Taxonomic Bias



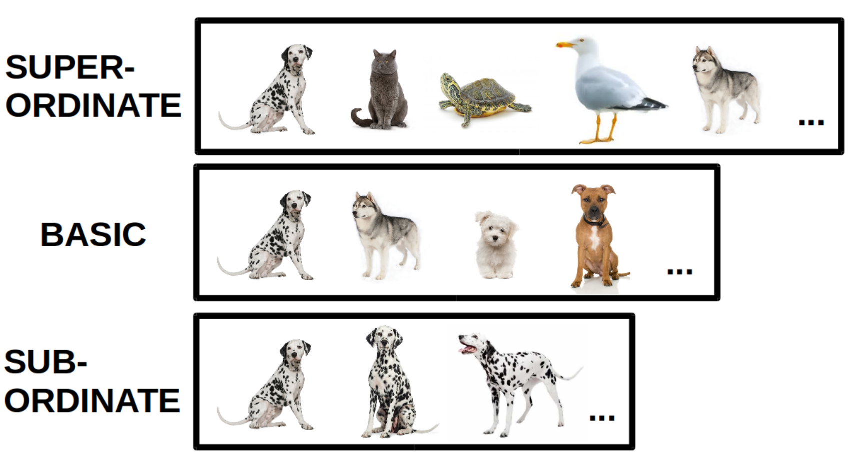

Words refer to objects not affordances or associations.

# Basic Level Bias

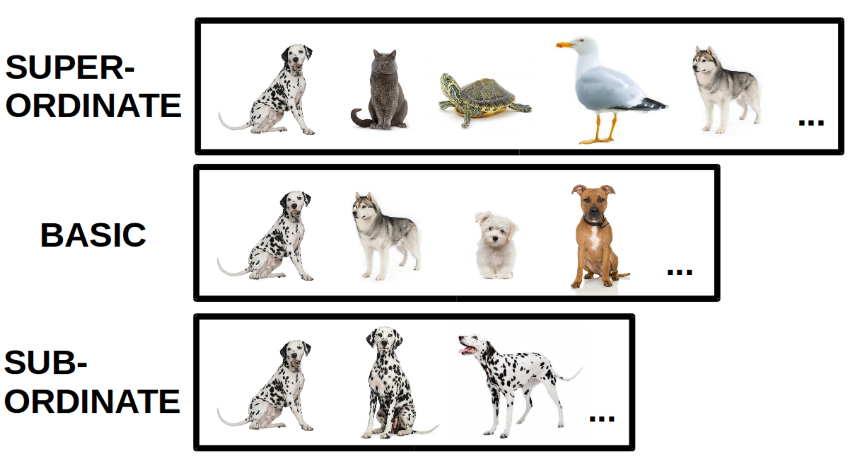



- Adults are more likely to label objects at the basic level.
- Adults are faster to name objects at the basic level.

# Basic Level Bias



# Basic Level Bias




**Size Principle:** $P(d|h) = \frac{1}{|h|}$. Penalizes hypotheses that pick out sets that are larger than what is required to capture the data.

# Learning Number Words

# Number Words

Children learn number words in stages.

We assess their knowledge using the Give-N task.

(Wyn, 1990; 1992)

# Number Words

Children learn number words in stages.

We assess their knowledge using the Give-N task.

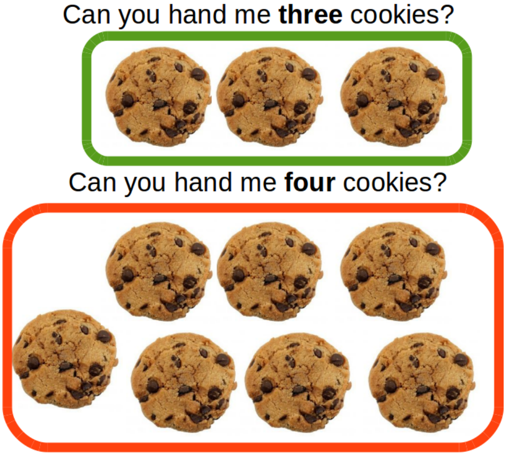
Can you hand me **three** cookies?

(Wyn, 1990; 1992)

# Number Words

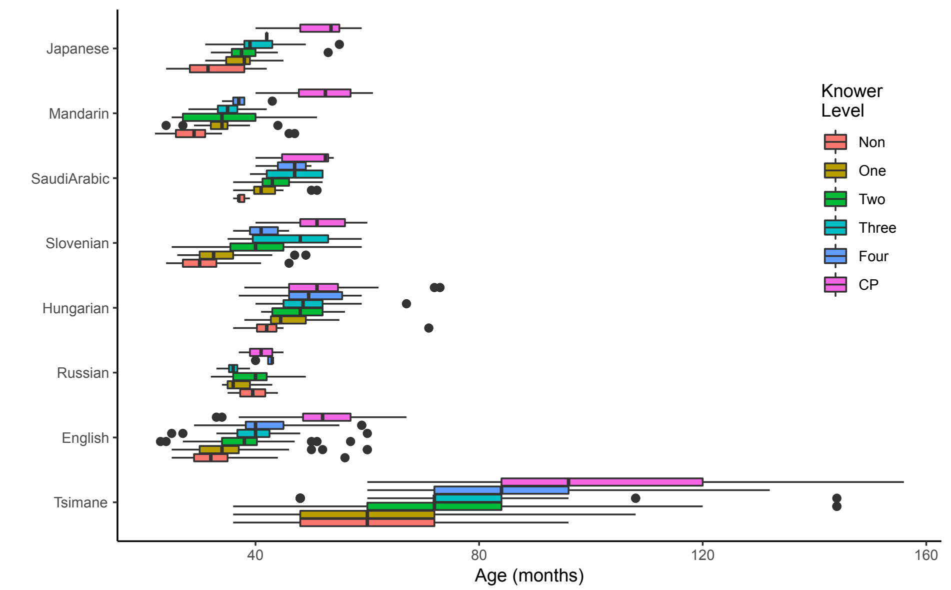
Children learn number words in stages.

We assess their knowledge using the Give-N task.




(Wyn, 1990; 1992)

# Number Words



# Number Words: Possible Hypotheses

**One-knower**

```
λ S . (if (singleton? S)
          "one"
          undef)
```

**Two-knower**

```
λ S . (if (singleton? S)
          "one"
          (if (doubleton? S)
              "two"
              undef))
```

**Three-knower**

```
λ S . (if (singleton? S)
          "one"
          (if (doubleton? S)
              "two"
              (if (tripleton? S)
                  "three"
                  undef))
```

**CP-knower**

```
λ S . (if (singleton? S)
          "one"
          (next (L (set-difference S
                                   (select S)))))
```

(Piantadosi, Tenenbaum & Goodman, 2012)

# Number Words: Possible Hypotheses

**Singular-Plural**

```
λ S . (if (singleton? S)
          "one"
          "two")
```

**Mod-5**

```
λ S . (if (or (singleton? S)
              (equal-word? (L (set-difference S)
                                (select S))
                           "five"))
          "one"
          (next (L (set-difference S
                                   (select S)))))
```

**2-not-1-knower**

```
λ S . (if (doubleton? S)
          "two"
          undef)
```

**2N-knower**

```
λ S . (if (singleton? S)
          "one"
          (next (next (L (set-difference S (select S))))))
```

(Piantadosi, Tenenbaum & Goodman, 2012)

# Number Word Learning (Piantadosi, Tenenbaum & Goodman, 2012)

**Program Induction:** Which hypothesis (program) $h$ led to the speaker uttering word $w$ when counting set $s$?

$$P(h|D) = P(h|w,s) \propto P(w|s,h)P(h)$$

### Input

(Word, Set) pairs.

For example:

$(\textit{three}, \cdots)$

### Prior

Simplicity bias: simpler programs $h$ are more likely.

### Output

A knower level:

1, 2, 3, 4, CP

### Likelihood

Noisy size principle:

$$P(w|s,h) = \begin{cases} \alpha + (1-\alpha)\frac{1}{10} & \text{if } w = h(s) \\ (1-\alpha)\frac{1}{10} & \text{else} \end{cases}$$

where $\alpha$ is the probability of uttering $w$ computed by $h$ applied to $s$

# Number Words: Hypothesis Space

Time for a short quiz on Wooclap!

https://app.wooclap.com/PPUKKP

# Number Words: Hypothesis Space

Time for a short quiz on Wooclap!

https://app.wooclap.com/PPUKKP

How do we definte the simplicity prior? We combine:

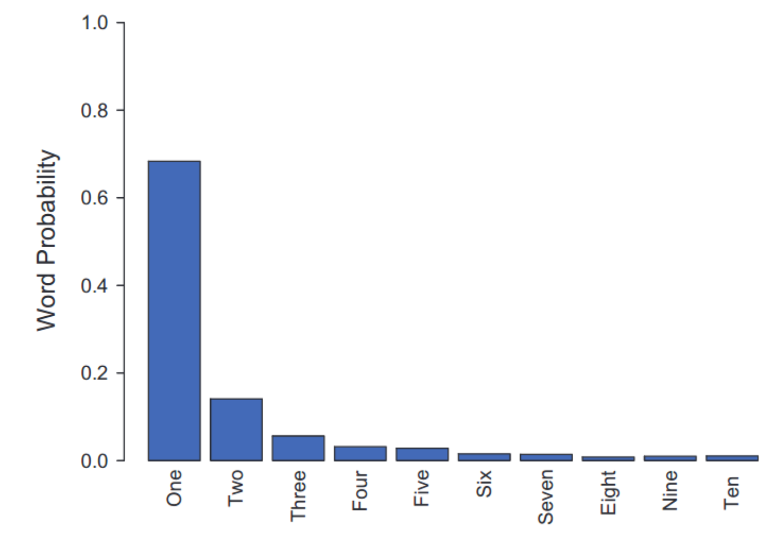
- rational rules prior: programs with fewer primitives more more probable
- penalty for recursion: programs that use recursion are less probably

# Number Words: Hypothesis Space

| | |
|---|---|
| **Functions mapping sets to truth values** | |
| (*singleton? X*) | Returns true iff the set *X* has exactly one element |
| (*doubleton? X*) | Returns true iff the set *X* has exactly two elements |
| (*tripleton? X*) | Returns true iff the set *X* has exactly three elements |
| **Functions on sets** | |
| (*set-difference X Y*) | Returns the set that results from removing *Y* from *X* |
| (*union X Y*) | Returns the union of sets *X* and *Y* |
| (*intersection X Y*) | Returns the intersect of sets *X* and *Y* |
| (*select X*) | Returns a set containing a single element from *X* |
| **Logical functions** | |
| (*and P Q*) | Returns TRUE if *P and Q* are both true |
| (*or P Q*) | Returns TRUE if either *P or Q* is true |
| (*not P*) | Returns TRUE iff *P* is false |
| (*if P X Y*) | Returns *X* iff *P* is true, *Y* otherwise |
| **Functions on the counting routine** | |
| (*next W*) | Returns the word after *W* in the counting routine |
| (*prev W*) | Returns the word before *W* in the counting routine |
| (*equal-word? W V*) | Returns TRUE if *W* and *V* are the same word |
| **Recursion** | |
| (*L S*) | Returns the result of evaluating the entire current lambda expression on set *S* |

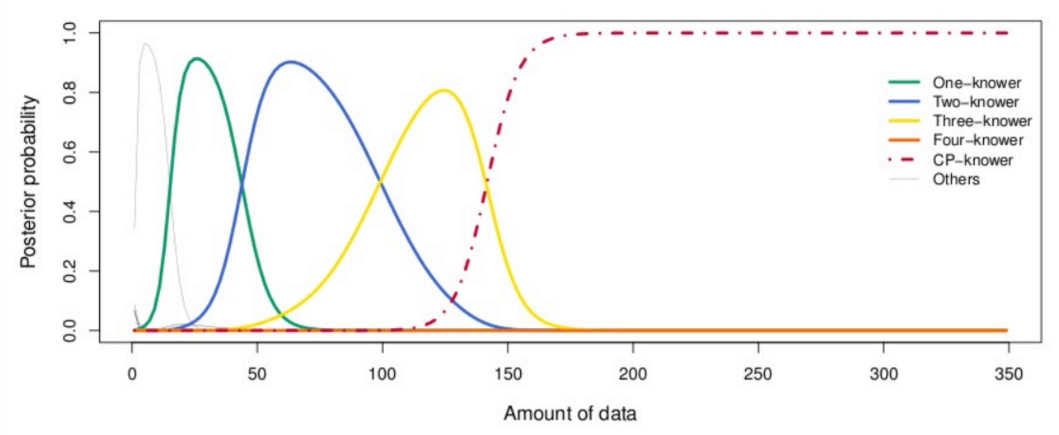
(Piantadosi, Tenenbaum & Goodman, 2012)

# Number Words: Environment




(Piantadosi, Tenenbaum & Goodman, 2012)

# Number Words: Results




(Piantadosi, Tenenbaum & Goodman, 2012)

# Summary

- In word learning, children face a generalization problem: they need to map words to concepts.
- The have inductive biases which make the problems easier: multual exclusivity, whole object bias, taxonomic bias.
- Fast mapping and cross-situational learning have been posited as learning mechanism.
- We can combine knowledge about the environment, inductive biases and learning to model how children acquire word meanings.
- We illustrated this for number word learning using Bayes Rule and Program Induction.