A Computational Study of Cross-Situational Techniques for Learning Word-To-Meaning Mappings

Jeffrey M. Siskind

CS 395T Spring 2019

Neha Srikanth
Nandhini Lakuduva
Motivation

● Eventually develop a system that acquires language as rapidly and effectively as humans do
  ○ Turing Test
● Model the “perceptual/conceptual” faculty of children as they build their vocabulary and acquire language
Problem Statement

- Mapping **words** to their **meaning**
- Developing a **precise algorithm** for approximating the **lexical acquisition** task

“Mary took the ball.”

<table>
<thead>
<tr>
<th>Word</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mary</td>
<td>{Mary}</td>
</tr>
<tr>
<td>the</td>
<td>{⊥}</td>
</tr>
<tr>
<td>took</td>
<td>{CAUSE(x, GO(y, TO(x)))}</td>
</tr>
<tr>
<td>ball</td>
<td>{spherical-toy}</td>
</tr>
</tbody>
</table>
Challenges

- Referential Uncertainty
  - Mary is wearing glasses while reading a book
  - Utterance = “Mary reads a book”
  - \{WEAR(Mary, glasses)\} vs. \{READ(Mary, book)\}

- Break down an utterance into words and align them with their correct conceptual meaning
  - Mary → \{READ(x, y)\}
  - book → \{Mary\}
  - read → \{glasses\}
Challenges

- Bootstrapping
  - Learn the lexicon without any apriori knowledge specific to the language
- Noise
  - Incorrect mappings of utterances to utterance meanings
  - Arises because utterances don’t match what we see in the here and now
- Homonymy
  - “John went to the ball”
  - $\text{ball} \rightarrow \{\text{formal-dance-party}\}$ vs. $\text{ball} \rightarrow \{\text{spherical-toy}\}$
Problem Representation

- Utterance
  - “Mary took the ball”

- Word Symbols
  - Mary, took, the, ball

- Sense symbols (for homonymy)
  - Mary₁, took₁, the₁, ball₁, ball₂

- Conceptual Symbols
  - \(\{\text{CAUSE, GO, TO}\}\)
  - \(N = \) set of conceptual symbols that are \textit{N}ecessary for the definition of the word
  - \(P = \) set of conceptual symbols that are \textit{P}ossible for the definition of the word

- Conceptual Expressions
  - \(\{\text{CAUSE}(x, \text{GO}(y, \text{TO}(x)))\}\)
  - \(D = \) set of possible conceptual expressions

- Meaning Representation
  - \(\{\text{CAUSE}(\text{Mary}, \text{GO}(\text{ball}, \text{TO}(\text{Mary})))\}\)
Principles of Lexical Acquisition: Set Constraints

- Constraining hypotheses with partial knowledge.
  - Utterance: Mary took the ball.
    
    - **took**
      - $N = \{\text{CAUSE}\}$
    
    - 1. WANT(Mary, block)
    
    - 2. CAUSE(Mary, GO(ball, TO(Mary)))

- Cross-situational inference
  - $P = \{\text{RED}(x), \text{spherical-toy}\}$
  
  - $P = \{\text{spherical-toy}\}$

RULE OUT MEANINGS: What can ball mean?
Principles of Lexical Acquisition: Set Constraints

- **Covering constraints**
  - Meaning of utterance = sum of its parts

    "Mary lifted the ball"
    \[ \text{CAUSE}(\text{Mary}, \text{GO (ball, UP)}) \]

    |      | N        | P         |
    |------|----------|-----------|
    | Mary | \{Mary\} | \{Mary\}  |
    | lifted | \{CAUSE\} | \{CAUSE, UP\} |
    | the  | \{}      | \{UP\}    |
    | ball | \{ball\} | \{Mary, UP, ball\} |

- **Principle of exclusivity**
  - Non overlapping portions of an utterance meanings

  lifted’s conceptual expression must contain \text{CAUSE}
Sequence of utterances + hypothesized meanings

((U₁ = “John walks to school”, M = \{m₁, m₂, ..., mₙ\}),
...
(Uₙ = “Mary ate a cookie”, M = \{m₁, m₂, ..., mₙ\}))

Stage 1:
Learn set of conceptual symbols

Stage 2:
Learn conceptual expressions

Output Lexicon

<table>
<thead>
<tr>
<th>Word</th>
<th>D</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mary</td>
<td>(Mary)</td>
<td>1000</td>
</tr>
<tr>
<td>took</td>
<td>CAUSE(x, GO(y, TO(x)))</td>
<td>100</td>
</tr>
<tr>
<td>the</td>
<td>(,)</td>
<td>10000</td>
</tr>
<tr>
<td>ball₁</td>
<td>(formal-dance-party)</td>
<td>10</td>
</tr>
<tr>
<td>ball₂</td>
<td>(spherical-toy)</td>
<td>100</td>
</tr>
<tr>
<td>lift₁</td>
<td>(elevator)</td>
<td>5</td>
</tr>
<tr>
<td>lift₂</td>
<td>CAUSE(x, GO(y, UP))</td>
<td>200</td>
</tr>
</tbody>
</table>

Algorithm

Stage 1:
Learn set of conceptual symbols

Stage 2:
Learn conceptual expressions

Output

Input
Algorithm: Example Run-Through

Suppose our first utterance is “John went to the ball” paired with these 3 hypothesized meaning representations:

1. CAUSE(John, GO(John, TO(formal-dance-party)))
2. CAUSE(John, GO(John, TO(spherical-toy)))
3. REMOVE(John, LEFT(shoe))
Algorithm: Example Run-Through

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>John &lt;sub&gt;1&lt;/sub&gt;</td>
<td>∅</td>
<td>U</td>
</tr>
<tr>
<td>went &lt;sub&gt;1&lt;/sub&gt;</td>
<td>∅</td>
<td>U</td>
</tr>
<tr>
<td>to &lt;sub&gt;1&lt;/sub&gt;</td>
<td>∅</td>
<td>U</td>
</tr>
<tr>
<td>the &lt;sub&gt;1&lt;/sub&gt;</td>
<td>∅</td>
<td>U</td>
</tr>
<tr>
<td>ball &lt;sub&gt;1&lt;/sub&gt;</td>
<td>∅</td>
<td>U</td>
</tr>
</tbody>
</table>

\[ \text{N is the lower bound for the actual conceptual symbol set} \]

\[ \text{P is the upper bound for the actual conceptual symbol set} \]

“Processing” an utterance = using constraints + rules to narrow down \( N, P, D, s \), enhance our mental lexicon, and eliminate noisy utterance meanings to the extent possible.

Process first utterance. Divide utterance into word symbols.
**Algorithm: Example Run-Through**

Currently, we do not know definitively what the **actual conceptual symbol set** is.

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>John(_1)</td>
<td>{John}</td>
<td>{John, ball}</td>
</tr>
<tr>
<td>went(_1)</td>
<td>{CAUSE}</td>
<td>{John, formal-dance-party, CAUSE, GO, TO}</td>
</tr>
<tr>
<td>to(_1)</td>
<td>∅</td>
<td>{WANT}</td>
</tr>
<tr>
<td>the(_1)</td>
<td>∅</td>
<td>{shoe}</td>
</tr>
<tr>
<td>ball(_1)</td>
<td>{formal-dance-party}</td>
<td>{ball, formal-dance-party, shoe, CAUSE}</td>
</tr>
</tbody>
</table>

Process many more utterances via cross-situational learning.
### Algorithm: Example Run-Through

#### Inconsistency:

- **1.** \( N \subseteq P \)
- **2.** \(|D| > 0\) (Stage 2)

When either of these rules are violated:

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>( John_1 )</td>
<td>{John}</td>
<td>{John, ball}</td>
</tr>
<tr>
<td>( went_1 )</td>
<td>{CAUSE}</td>
<td>{John, formal-dance-party, CAUSE, GO, TO}</td>
</tr>
<tr>
<td>( to_1 )</td>
<td>∅</td>
<td>{WANT}</td>
</tr>
<tr>
<td>( the_1 )</td>
<td>∅</td>
<td>{shoe}</td>
</tr>
<tr>
<td>( ball_1 )</td>
<td>{formal-dance-party}</td>
<td>{formal-dance-party}</td>
</tr>
<tr>
<td>( ball_2 )</td>
<td>{spherical-toy}</td>
<td>{spherical-toy}</td>
</tr>
</tbody>
</table>

Algorithm finds the smallest number of new sense symbols that need to be added to process the utterance without **inconsistency**.

1. **Noise** (new senses are **spurious**)
2. **Homonymy** (senses will **converge**)
3. Entries for some other word in the utterance are corrupted (sense created to patch corrupted words, new senses will **converge**)

Splitting *ball* into 2 senses due to inconsistency.
Algorithm: Example Run-Through

1. $N \subseteq P$
2. $|D| > 0$ (Stage 2)

Inconsistency:
When either of these rules are violated

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>$John_1$</td>
<td>{John}</td>
<td>{John, ball}</td>
</tr>
<tr>
<td>$went_1$</td>
<td>{CAUSE}</td>
<td>{John, formal-dance-party, CAUSE, GO, TO}</td>
</tr>
<tr>
<td>$to_1$</td>
<td>∅</td>
<td>{WANT}</td>
</tr>
<tr>
<td>$the_1$</td>
<td>∅</td>
<td>{shoe}</td>
</tr>
<tr>
<td>$ball_1$</td>
<td>{formal-dance-party}</td>
<td>{formal-dance-party}</td>
</tr>
<tr>
<td>$ball_2$</td>
<td>{foot}</td>
<td>{arm, spherical-toy}</td>
</tr>
</tbody>
</table>

Splitting $ball$ into 2 senses due to inconsistency.

In this case, $ball_2$ will most likely never converge.
Stage 1 Set Manipulation: The Rules

Four inference rules to manipulate conceptual symbols sets N and P

1. Ignore utterance meanings with symbols not contained in P of any of its word symbols or without symbols in N any of its word symbols
2. If any word symbols’ P contains a conceptual symbol that isn’t in any utterance meaning, remove that conceptual symbol from P
3. For each word w, add to N any conceptual symbols that appear in all utterance meanings but that are absent from all other words’ P.
4. For each word, remove from P any conceptual symbols that appear only once if they are in any other words’ N.
Algorithm: Example Run-Through

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>John₁</td>
<td>{John}</td>
<td>{John}</td>
</tr>
<tr>
<td>went₁</td>
<td>{John, formal-dance-party, CAUSE, GO, TO}</td>
<td>{John, formal-dance-party, CAUSE, GO, TO}</td>
</tr>
<tr>
<td>to₁</td>
<td>∅</td>
<td>∅</td>
</tr>
<tr>
<td>the₁</td>
<td>∅</td>
<td>∅</td>
</tr>
<tr>
<td>ball₁</td>
<td>{formal-dance-party}</td>
<td>{formal-dance-party}</td>
</tr>
</tbody>
</table>

Convergence: the actual conceptual symbol set is found when N and P are equal.

Simultaneous Stages: Some words in the lexicon can be in Stage 1 while others are in Stage 2. For clarity, we've shown all words conclude Stage 1 simultaneously.

After processing many more utterances and applying the rules...
Algorithm: Example Run-Through

2 inference rules to manipulate conceptual expression sets $D$.
  - $D$ is initialized to universal set.
  - Remove symbols from $D$ until $|D|$ is exactly one.

5. Define $\text{Reconstruct}(m, N(w))$ to be a function that creates conceptual expressions for each word $w$ from $N(w)$ and returns only those expressions that are subexpressions of $m$, a possible utterance meaning. Remove from $D(w)$ any conceptual expression not generated by $\text{Reconstruct}$.

6. Define $\text{Compose}([D_1, D_2, \ldots, D_n])$ as a function that creates possible utterance meanings from combining the conceptual expressions of each word. Remove from $D(w)$ any conceptual expression that when fed into $\text{Compose}$ results in an utterance meaning that is not in the original hypothesized set of meanings.
Algorithm: Example Run-Through

- **Compose**: combining *individual conceptual expressions* to form an utterance meaning.

  **Input**
  
  Unordered collection of conceptual expressions, one for each word

  \[
  \text{Compose}\left\{\{\text{John}, \text{GO} (x, y), \text{TO} (x), \text{school}\}\right\}
  \]

  **Compose**

  Set of utterance meanings, formed from the valid cross product of sense assignments of given conceptual expressions

  \[
  \begin{align*}
  &\text{GO (John, TO (school))} \\
  &\text{GO (school, TO (John))} \\
  &\text{GO (TO (John), school)} \\
  &\text{GO (TO (school), John)} \\
  &\text{TO (GO (John, school))} \\
  &\text{TO (GO (school, John))}
  \end{align*}
  \]
Algorithm: Example Run-Through

Convergence:
the actual conceptual expression set is found when $|D|$ is exactly 1.

<table>
<thead>
<tr>
<th>Word</th>
<th>D</th>
<th>Confidence Factors (C)</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>{John}, C=10</td>
<td></td>
</tr>
<tr>
<td>went</td>
<td>{CAUSE(x, GO(x, TO(y))}, C=100</td>
<td></td>
</tr>
<tr>
<td>to</td>
<td>{_}, C=1000</td>
<td></td>
</tr>
<tr>
<td>the</td>
<td>{_}, C=1000</td>
<td></td>
</tr>
<tr>
<td>ball</td>
<td>{formal-dance-party}, C=10</td>
<td></td>
</tr>
<tr>
<td>ball</td>
<td>{ball}, C=100</td>
<td></td>
</tr>
</tbody>
</table>

Confidence Factors (C): Counts the number of times that that sense was used to process an utterance.

Rough measure of the relative frequency of occurrence of different word senses.

We’ve converged on a word-meaning mapping...DONE!!

After processing many more utterances and applying the rules.
Method Evaluation
Data Generation

- No corpora of naturally occurring utterances paired with hypothesized utterance meanings → **develop a synthetic corpora with a vocabulary size of** \( n \) **and a homonymy rate of** \( r \)
- Randomly-construct conceptual expressions for word symbol meanings
  - All the conceptual symbols were distributed uniformly when constructing the random conceptual expressions
  - Senses were distributed uniformly among the words
- Maximal depth of 2 and a maximal branching factor of 3 and were intended to model verb and noun-like meanings
- Hypothesized utterance meaning representations generated “intelligently” → modeled after how children would hypothesize possible MRs

S → XP
XP → NP|VP
NP → \( \{F\}N \)
VP → \( \{F\}V \) XP^+
Simulation Methodology

- For each simulation, a random gold lexicon was generated that mapped simulated words to simulated meanings → pass this lexicon to a program that generates a stream of utterances paired with sets of meaning hypotheses
- Apply online lexical acquisition algorithm to produce reconstructed lexicon
- Simulation terminates when reconstructed lexicon contained 95% of correct word-to-meaning mappings
- 95% was used because the last 5% converge very slowly (Zipf’s Law)
Simulations

Measure **how well does this algorithm scale** as complexity is varied across 5 axes:

- Vocabulary size
- Degree of referential uncertainty
- Noise rate
- Conceptual Symbol Inventory Size
- Homonymy Rate

Measure **the growth of vocabulary size achieved** by the algorithm vs. exposure to simulated training corpus.

Measure **# of exposures to a new word that is required to learn** that word vs. amount of corpus already processed at time of new word occurrence.

Determine if algorithm **can solve very large learning task** similar to the complexity level faced by children.
Table 1
Various measurements of the quantity of speech to children normalized to utterances per hour

<table>
<thead>
<tr>
<th>Source</th>
<th>Minimum</th>
<th>Average</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schachter et al. (1976)</td>
<td>Toddlers</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3-year-olds</td>
<td>245</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4-year-olds</td>
<td>219</td>
<td></td>
</tr>
<tr>
<td>Snow (1977)</td>
<td></td>
<td>504</td>
<td>1197</td>
</tr>
<tr>
<td>Kaye (1980)</td>
<td>Infants</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2-year-olds</td>
<td>870</td>
<td></td>
</tr>
<tr>
<td>Moerk (1983)</td>
<td></td>
<td>283</td>
<td></td>
</tr>
<tr>
<td>Wells (1986)</td>
<td></td>
<td>80</td>
<td>800</td>
</tr>
<tr>
<td>Bernstein-Ratner (pers. comm.)</td>
<td></td>
<td></td>
<td>1089</td>
</tr>
</tbody>
</table>
Results

- Algorithm **scales linearly in the vocabulary size**
- Insensitive to the degree of referential uncertainty and the conceptual-symbol inventory size
- **Task becomes exponentially difficult** with increasing noise and homonymy (need a larger corpus)
- Learning rate is slow for first 25 words or so, then proceeds rapidly, then tapers off as algorithm converges (similar to child language acquisition)
- The average number of occurrences needed for convergence on a new word decreases with corpus exposure
  - after ~4000 utterances, most words are acquired after 1-2 occurrences
- Input needs of the algorithm ~ the data available to children when acquiring a similar-sized lexicon
Conclusion + Critique
Conclusion

- Precise, implemented algorithm for lexical acquisition using reasonable assumptions
- Results are consistent with Carey’s (1978) findings for child lexical acquisition
- Addresses central problems in lexical acquisition
- **Computational** approaches lead to a better understanding of the language acquisition process.
Limitations

- Difficult to set realistic simulation parameters
- Capable of learning like children under the same circumstances but cannot pinpoint what in the algorithm is actually something children do
  - Can’t estimate homonymy rate, degree of referential uncertainty, and noise rate children face
- Very little known about actual shape + size of conceptual representations → difficult to generate realistic meaning representations for gold lexicon
Limitations

- **Polysemy** vs. homonymy
- Poor understanding of how conceptual representations formed from perceptual input → arbitrary noise rate + degree of referential uncertainty
- Cannot learn to understand idioms and metaphors
- Difficult to assess homonymy rate **across languages**
- Language use is **fragmentary** → didn’t devise a method to associate utterance fragments with portions of hypothesized semantic representations
- Model **assumes** children can include the correct utterance meaning in the set of meaning hypotheses for abstract terms
Research Directions: Improve Language Acquisition Models

● Basis for additional research to find ways of addressing these limitations
● Work needs to be done in understanding human formation of conceptual representations → help aid development of language acquisition model
  ○ Need to gain better understanding of:
    ■ Conceptual symbol inventory
    ■ Semantic interpretation rule
    ■ Perceptual/conceptual processes used to hypothesize utterance meanings from observational input
● Studying language acquisition computationally in the context of perception and action
Questions?