Complexity, meaning, and quantifiers

Jakub Szymanik

Vague Quantities and Vague Quantifiers
Berlin, December 9, 2010
Quantifiers
Quantifiers
Quantifiers
Quantifiers
Plan

1. Distinguish denotations from procedures.
Plan

1. Distinguish denotations from procedures.
2. Denotations are not vague (back-down).
3. But procedures might only approximate.
4. Precision costs a lot!
5. So, it might be necessary to approximate.
6. Some evidence on ‘more than half’.
Plan

1. Distinguish denotations from procedures.
2. Denotations are not vague (back-down).
3. But procedures might only approximate.
4. Precision costs a lot!
5. So, it might be necessary to approximate.
Plan

1. Distinguish denotations from procedures.
2. Denotations are not vague (back-down).
3. But procedures might only approximate.
4. Precision costs a lot!
5. So, it might be necessary to approximate.
6. Some evidence on ‘more than half’.
Outline

Quantifiers and cognitive strategies

Cognition and computability

Generalized Quantifier Theory

Computing quantifiers

Complexity and reaction time

Complexity and working memory

Outlook
Aspects of meaning:

- comprehension
- reasoning
- use
- VERIFICATION

Meaning
How are people doing it?

They apply some strategies/procedures/algorithms. Those depend on:

- cognitive architecture;
- visual clues;
- level of precision subjects want to achieve;
- quantifiers;

...
How are people doing it?

- They apply some strategies/procedures/algorithms.

Those depend on:
- cognitive architecture;
- visual clues;
- level of precision subjects want to achieve;
- quantifiers;

We discussed many examples at that workshop.
How are people doing it?

- They apply some strategies/procedures/algorithms.
- Those depend on:
  - cognitive architecture;
  - visual clues;
  - level of precision subjects want to achieve;
  - quantifiers;
  - ...
How are people doing it?

- They apply some strategies/procedures/algorithms.
- Those depend on:
  - cognitive architecture;
  - visual clues;
  - level of precision subjects want to achieve;
  - quantifiers;
  - ...

We discussed many examples at that workshop.
Main question

Question

*How are different mechanisms related?*
Outline

Quantifiers and cognitive strategies

Cognition and computability

Generalized Quantifier Theory

Computing quantifiers

Complexity and reaction time

Complexity and working memory

Outlook
Marr’s 3 levels of explanation

1. computational level:
   ▶ problems that a cognitive ability has to overcome
Marr’s 3 levels of explanation

1. computational level:
   ▶ problems that a cognitive ability has to overcome

2. algorithmic level:
   ▶ the algorithms that may be used to achieve a solution

Marr, Vision: a computational investigation into the human representation and processing visual information, 1983
Marr’s 3 levels of explanation

1. computational level:
   ▶ problems that a cognitive ability has to overcome

2. algorithmic level:
   ▶ the algorithms that may be used to achieve a solution

3. implementation level:
   ▶ how this is actually done in neural activity

Marr, Vision: a computational investigation into the human representation and processing visual information, 1983
Expression $\xrightarrow{\text{computation}}$ denotation

- Ability to understand sentences.
- Capacity of recognizing their truth-values.
Expression $\Rightarrow$ denotation

- Ability to understand sentences.
- Capacity of recognizing their truth-values.
- Long-standing tradition.
- Meaning is a procedure for finding extension in a model.
Expression $\xrightarrow{\text{computation}}$ denotation

- Ability to understand sentences.
- Capacity of recognizing their truth-values.
- Long-standing tradition.
- Meaning is a procedure for finding extension in a model.
- Adopted often with psychological motivations.
Explicit formulation

Pavel Tichý “Intension in terms of Turing machines”, 1969:

...the fundamental relationship between sentence and procedure is obviously of a semantic nature; namely, the purpose of sentences is to record the outcome of various procedures. Thus e.g. the sentence "The liquid X is an acid" serves to record that the outcome of a definite chemical testing procedure applied to X is positive.
Pavel Tichý “Intension in terms of Turing machines”, 1969:

[…] the fundamental relationship between sentence and procedure is obviously of a semantic nature; namely, the purpose of sentences is to record the outcome of various procedures. Thus e.g. the sentence “The liquid X is an acid” serves to record that the outcome of a definite chemical testing procedure applied to X is positive.
For what does it mean to understand, i.e. to know the sense of an expression? It does not mean actually to know its denotation but to know how the denotation can be found, how to pinpoint the denotation of the expression among all the objects of the same type. E.g. to know the sense of “taller” does not mean actually to know who is taller than who, but rather to know what to do whenever you want to decide whether a given individual is taller than another one. In other words, it does not mean to know which of the binary relations on the universe is the one conceived by the sense of “taller”, but to know a method or procedure by means of which the relation can be identified. (Tichý, 1969)
The basic and fundamental psychological point is that, with rare exceptions, in applying a predicate to an object or judging that a relation holds between two or more objects, we do not consider properties or relations as sets. We do not even consider them as somehow simply intensional properties, but we have procedures that compute their values for the object in question. Thus, if someone tells me that an object in the distance is a cow, I have a perceptual and conceptual procedure for making computations on the input data that reach my peripheral sensory system [. . .] Fregean and other accounts scarcely touch this psychological aspect of actually determining application of a specific algorithmic procedure. (Suppes 1982)
Meaning as a collection of procedures

I have defended the thesis that the meaning of a sentence is a procedure or a collection of procedures and that this meaning in its most concrete representation is wholly private and idiosyncratic to each individual. (Suppes 1982)
Question
What are we computing in the case of quantifiers?
Outline

Quantifiers and cognitive strategies

Cognition and computability

Generalized Quantifier Theory

Computing quantifiers

Complexity and reaction time

Complexity and working memory

Outlook
What is the semantic assigned to quantifiers?

1. Every poet has low self-esteem.
2. Some dean danced nude on the table.
3. At least 7 grad students prepared presentations.
4. An even number of the students saw a ghost.
5. Most of the students think they are smart.
6. Less than half of the students received good marks.
Monadic quantifiers of type $(1, 1)$

Definition

A monadic generalized quantifier of type $(1,1)$ is a class $Q$ of structures of the form $M = (U, A_1, A_2)$, where $A_1, A_2 \subseteq U$. Additionally, $Q$ is closed under isomorphism.
Examples

every = \{(M, A, B) \mid A, B \subseteq M \text{ and } A \subseteq B\}.
Examples

every = \{ (M, A, B) \mid A, B \subseteq M \text{ and } A \subseteq B \}.

some = \{ (M, A, B) \mid A, B \subseteq M \text{ and } A \cap B \neq \emptyset \}.
Examples

every $= \{(M, A, B) \mid A, B \subseteq M \text{ and } A \subseteq B\}$.

some $= \{(M, A, B) \mid A, B \subseteq M \text{ and } A \cap B \neq \emptyset\}$.

more than $k = \{(M, A, B) \mid A, B \subseteq M \text{ and } \text{card}(A \cap B) > k\}$.
Examples

\[
\text{every} = \{ (M, A, B) \mid A, B \subseteq M \text{ and } A \subseteq B \}.
\]

\[
\text{some} = \{ (M, A, B) \mid A, B \subseteq M \text{ and } A \cap B \neq \emptyset \}.
\]

\[
\text{more than } k = \{ (M, A, B) \mid A, B \subseteq M \text{ and } \text{card}(A \cap B) > k \}.
\]

\[
\text{even} = \{ (M, A, B) \mid A, B \subseteq M \text{ and } \text{card}(A \cap B) \text{ is even} \}.
\]
Examples

every = \{(M, A, B) \mid A, B \subseteq M \text{ and } A \subseteq B\}.

some = \{(M, A, B) \mid A, B \subseteq M \text{ and } A \cap B \neq \emptyset\}.

more than k = \{(M, A, B) \mid A, B \subseteq M \text{ and } \text{card}(A \cap B) > k\}.

even = \{(M, A, B) \mid A, B \subseteq M \text{ and } \text{card}(A \cap B) \text{ is even}\}.

most = \{(M, A, B) \mid A, B \subseteq M \text{ and } \text{card}(A \cap B) > \text{card}(A - B)\}\.
Why are logicians so excited with GQs?
Why are logicians so excited with GQs?

Everyone knows everyone here.
Why are logicians so excited with GQs?

*Everyone knows everyone here.*
Why are logicians so excited with GQs?

Everyone knows everyone here.
Why are logicians so excited with GQs?

*Everyone knows everyone here.*
Why are logicians so excited with GQs?

Everyone knows everyone here.
Why are logicians so excited with GQs?

Everyone knows everyone here.
Why are logicians so excited with GQs?

Everyone knows everyone here.
Why are logicians so excited with GQs?

*Everyone knows everyone here.*
Why are logicians so excited with GQs?

*Everyone knows everyone here.*
Why are logicians so excited with GQs?

Everyone knows everyone here.
Why are logicians so excited with GQs?

Everyone knows everyone here.
Why are logicians so excited with GQs?

Everyone knows everyone here.
Why are logicians so excited with GQs?

*Everyone knows everyone here.*
Why are logicians so excited with GQs?

Everyone knows everyone here.
Why are logicians so excited with GQs?

*Everyone knows everyone here.*
Why are logicians so excited with GQs?

*Everyone knows everyone here.*
Why are logicians so excited with GQs?

*Everyone knows everyone here.*
Why are logicians so excited with GQs?

*Everyone knows everyone here.*
Why are logicians so excited with GQs?

Everyone knows everyone here.
Why are logicians so excited with GQs?

Everyone knows everyone here.
Why are logicians so excited with GQs?

Everyone knows everyone here.
Why are logicians so excited with GQs?

Everyone knows everyone here.
Why are logicians so excited with GQs?

*Everyone knows everyone here.*
Why are logicians so excited with GQs?

*Everyone knows everyone here.*
We know what GQs denote. Now, it's time to see how we compute those denotations.
Outline

Quantifiers and cognitive strategies

Cognition and computability

Generalized Quantifier Theory

Computing quantifiers

Complexity and reaction time

Complexity and working memory

Outlook
Automata perspective

An attractive, but never very central idea in modern semantics has been to regard linguistic expressions as denoting certain “procedures” performed within models for the language. (Van Benthem, 1986)
This model is uniquely described by $\alpha_M = a_{\bar{A}\bar{B}}a_{A\bar{B}}a_{AB}a_{\bar{A}B}a_{\bar{A}\bar{B}}$.
Step by step

- Restriction to finite models of the form $M = (U, A, B)$.
Step by step

- Restriction to finite models of the form $M = (U, A, B)$.
- List of all elements of the model: $c_1, \ldots, c_5$.
Step by step

- Restriction to **finite models** of the form $M = (U, A, B)$.
- List of all elements of the model: $c_1, \ldots, c_5$.
- Labeling every element with one of the letters: $a_{\bar{A}\bar{B}}, a_{A\bar{B}}, a_{\bar{A}B}, a_{AB}$, according to constituents it belongs to.
Step by step

- Restriction to finite models of the form $M = (U, A, B)$.
- List of all elements of the model: $c_1, \ldots, c_5$.
- Labeling every element with one of the letters: $a_{\bar{A}B}$, $a_{AB}$, $a_{\bar{A}B}$, according to constituents it belongs to.
- Result: the word $\alpha_M = a_{\bar{A}B}a_{AB}a_{AB}a_{\bar{A}B}a_{\bar{A}B}$.
Step by step

- Restriction to finite models of the form \( M = (U, A, B) \).
- List of all elements of the model: \( c_1, \ldots, c_5 \).
- Labeling every element with one of the letters: \( a_{\overline{AB}}, a_{AB}, a_{\overline{AB}}, a_{AB} \), according to constituents it belongs to.
- Result: the word \( \alpha_M = a_{\overline{AB}}a_{AB}a_{AB}a_{\overline{AB}}a_{\overline{AB}} \).

\( \alpha_M \) describes the model in which:
\( c_1 \in \overline{AB}, c_2 \in ABc_3 \in AB, c_4 \in \overline{AB}, c_5 \in \overline{AB} \).
Step by step

- Restriction to finite models of the form $M = (U, A, B)$.
- List of all elements of the model: $c_1, \ldots, c_5$.
- Labeling every element with one of the letters: $a_{\overline{AB}}, a_{AB}, a_{\overline{AB}}, a_{\overline{AB}}$, according to constituents it belongs to.
- Result: the word $\alpha_M = a_{\overline{AB}} a_{AB} a_{AB} a_{\overline{AB}} a_{\overline{AB}}$.
- $\alpha_M$ describes the model in which: $c_1 \in \overline{A}B, c_2 \in AB, c_3 \in \overline{A}B, c_4 \in \overline{A}B, c_5 \in \overline{A}B$.
- The class $Q$ is represented by the set of words describing all elements of the class.
Aristotelian quantifiers

“all”, “some”, “no”, and “not all”

\[ \Gamma - \{ a_{AB} \} \]

\[ q_0 \quad a_{AB} \quad q_1 \]

Finite automaton recognizing \( L_{\text{All}} \)

\[ L_{\text{All}} = \{ \alpha \in \Gamma^* : \#a_{AB}(\alpha) = 0 \} \]
Cardinal quantifiers

E.g. “more than 2”, “less than 7”, and “between 8 and 11”

\[
\begin{align*}
\Gamma &-\{a_{AB}\} \\
q_0 &\xrightarrow{a_{AB}} q_1 \\
\Gamma &-\{a_{AB}\} \\
q_2 &\xrightarrow{a_{AB}} q_3 \\
\end{align*}
\]

Finite automaton recognizing \(L_{\text{More than two}}\)

\[L_{\text{More than two}} = \{\alpha \in \Gamma^*: \#a_{AB}(\alpha) > 2\}\]
Parity quantifiers

E.g. “an even number”, “an odd number”

\[ \Gamma - \{ a_{AB} \} \]

Finite automaton recognizing \( L_{\text{Even}} \)

\[ L_{\text{Even}} = \{ \alpha \in \Gamma^* : \# a_{AB}(\alpha) \text{ is even} \} \]
Proportional quantifiers

- E.g. “most”, “less than half”.
- Most \( As \) are \( B \) iff \( \text{card}(A \cap B) > \text{card}(A - B) \).
- \( L_{\text{Most}} = \{ \alpha \in \Gamma^* : \#a_{AB}(\alpha) > \#a_{A\bar{B}}(\alpha) \} \).
- There is no finite automaton recognizing this language.
- We need internal memory.
- A push-down automata will do.
Summing up

<table>
<thead>
<tr>
<th>Definability</th>
<th>Examples</th>
<th>Recognized by</th>
</tr>
</thead>
<tbody>
<tr>
<td>FO</td>
<td>“all” “at least 3”</td>
<td>acyclic FA</td>
</tr>
<tr>
<td>FO($D_n$)</td>
<td>“an even number”</td>
<td>FA</td>
</tr>
<tr>
<td>PrA</td>
<td>“most”, “less than half”</td>
<td>PDA</td>
</tr>
</tbody>
</table>

Quantifiers, definability, and complexity of automata

Mostowski, Computational semantics for monadic quantifiers, 1998.
Does it say anything about processing?

Question

*Do minimal automata predict differences in verification?*
Does it say anything about processing?

Question

Do minimal automata predict differences in verification?

We’ll try to convince you that the answer is positive!
Specific task: ‘Most’ vs. ‘More than half’
Specific task: ‘Most’ vs. ‘More than half’

- Different distribution in corpus.
- They trigger different verification strategies.

Solt, On orderings and quantification: the case of most and more than half, manuscript, 2010

Hackl, On the grammar and processing of proportional quantifiers, Natural Language Semantics, 2009
Specific task: ‘Most’ vs. ‘More than half’

- Different distribution in corpus.
- They trigger different verification strategies.

Solt, On orderings and quantification: the case of most and more than half, manuscript, 2010

Hackl, On the grammar and processing of proportional quantifiers, Natural Language Semantics, 2009

We saw data on ‘most’; let’s look into ‘more than half’.
Outline

Quantifiers and cognitive strategies
Cognition and computability
Generalized Quantifier Theory
Computing quantifiers
Complexity and reaction time
Complexity and working memory
Outlook
Predictions

- RT will increase along with the computational resources.
Predictions

- RT will increase along with the computational resources.
- Aristotelian qua. < parity qua. < proportional qua.
Predictions

- RT will increase along with the computational resources.
- Aristotelian qua. < parity qua. < proportional qua.
- Aristotelian qua. < cardinal qua. of high rank.
Grammatically simple propositions in Polish, like:

1. Some cars are red.
2. More than 7 cars are blue.
3. An even number of cars is yellow.
4. Less than half of the cars are black.
More than half of the cars are yellow.

An example of a stimulus used in the first study
Procedure

- 8 different quantifiers divided into four groups.
  - “all” and “some” (acyclic 2-state FA);
  - “odd” and “even” (2-state FA);
  - “less than 8” and “more than 7” (FA);
  - “less than half” and “more than half” (PDA).

Quantity of target items near the criterion of validation.

Numerical and proportional quantifiers logically equivalent.
Procedure

- 8 different quantifiers divided into four groups.
  - “all” and “some” (acyclic 2-state FA);
  - “odd” and “even” (2-state FA);
  - “less than 8” and “more than 7” (FA);
  - “less than half” and “more than half” (PDA).

Quantity of target items near the criterion of validation.

Numerical and proportional quantifiers logically equivalent.
Procedure

- 8 different quantifiers divided into four groups.
  - “all” and “some” (acyclic 2-state FA);
  - “odd” and “even” (2-state FA);
  - “less than 8” and “more than 7” (FA);
  - “less than half” and “more than half” (PDA).

Quantity of target items near the criterion of validation.

Numerical and proportional quantifiers logically equivalent.
Procedure

- 8 different quantifiers divided into four groups.
  - “all” and “some” (acyclic 2-state FA);
  - “odd” and “even” (2-state FA);
  - “less than 8” and “more than 7” (FA);
  - “less than half” and “more than half” (PDA).
Procedure

- 8 different quantifiers divided into four groups.
  - “all” and “some” (acyclic 2-state FA);
  - “odd” and “even” (2-state FA);
  - “less than 8” and “more than 7” (FA);
  - “less than half” and “more than half” (PDA).
Procedure

- 8 different quantifiers divided into four groups.
  - “all” and “some” (acyclic 2-state FA);
  - “odd” and “even” (2-state FA);
  - “less than 8” and “more than 7” (FA);
  - “less than half” and “more than half” (PDA).

- Quantity of target items near the criterion of validation.
Procedure

- 8 different quantifiers divided into four groups.
  - “all” and “some” (acyclic 2-state FA);
  - “odd” and “even” (2-state FA);
  - “less than 8” and “more than 7” (FA);
  - “less than half” and “more than half” (PDA).

- Quantity of target items near the criterion of validation.

- Numerical and proportional quantifiers logically equivalent.
## Accuracy

<table>
<thead>
<tr>
<th>Quantifier group</th>
<th>Examples</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aristotelian</td>
<td>all, some</td>
<td>99</td>
</tr>
<tr>
<td>Parity</td>
<td>odd, even</td>
<td>91</td>
</tr>
<tr>
<td>Cardinal</td>
<td>less than 8, more than 7</td>
<td>92</td>
</tr>
<tr>
<td>Proportional</td>
<td>less than half, more than half</td>
<td>85</td>
</tr>
</tbody>
</table>

The percentage of correct answers
It interacts with monotonicity

1. “More than 7”: true > false (8>7).
It interacts with monotonicity

1. “More than 7”: true > false (8>7).
2. “Fewer than 8”: true < false (7<8).
Interaction effect

Szymanik & Zajenkowski, Computational approach to monotonicity in sentence-picture verification, under review
Outline

Quantifiers and cognitive strategies

Cognition and computability

Generalized Quantifier Theory

Computing quantifiers

Complexity and reaction time

Complexity and working memory

Outlook
McMillan et al. fMRI studies

Differences in brain activity.
McMillan et al. fMRI studies

Differences in brain activity.

▶ All quantifiers are associated with numerosity: recruit right inferior parietal cortex.
McMillan et al. fMRI studies

Differences in brain activity.

- All quantifiers are associated with numerosity: recruit right inferior parietal cortex.
- Only higher-order activate working-memory capacity: recruit right dorsolateral prefrontal cortex.
McMillan et al. fMRI studies

Differences in brain activity.

- All quantifiers are associated with numerosity: recruit right inferior parietal cortex.
- Only higher-order activate working-memory capacity: recruit right dorsolateral prefrontal cortex.

But definability seems not to be fine grained enough!

McMillan et al., Neural basis for generalized quantifiers comprehension, Neuropsychologia, 2005

Szymanik, A Note on some neuroimaging study of natural language quantifiers comprehension, Neuropsychologia, 2007
Baddeley’s model

WM unified system responsible for the performance in complex tasks.
Baddeley’s model

WM unified system responsible for the performance in complex tasks.

- The model consists of:
  - temporary storage units:
    - phonological loop;
    - visual loop;
  - a controlling system (central executive).

Baddeley, Working memory and language: an overview, 2003
Span test

To assess the working memory construct.
Subjects read sentences.
They are asked to:
- remember the final words.
- comprehend the story.

What is:
- the number of correctly memorized words?
- the degree of understanding?
- Engagement of processing and storage functions.

Daneman and Carpenter, Individual differences in working memory, 1980
Span test

- To assess the working memory construct.

  - Subjects read sentences.
  - They are asked to:
    - remember the final words.
    - comprehend the story.
  - What is:
    - the number of correctly memorized words?
    - the degree of understanding?
    - Engagement of processing and storage functions.

Daneman and Carpenter, Individual differences in working memory, 1980
Span test

- To assess the working memory construct.
- Subjects read sentences.

- They are asked to:
  - Remember the final words.
  - Comprehend the story.
- What is:
  - The number of correctly memorized words?
  - The degree of understanding?
- Engagement of processing and storage functions.

Daneman and Carpenter, Individual differences in working memory, 1980
Span test

- To assess the working memory construct.
- Subjects read sentences.
- They are asked to:
  - remember the final words.
  - comprehend the story.
Span test

- To assess the working memory construct.
- Subjects read sentences.
- They are asked to:
  - remember the final words.
  - comprehend the story.
- What is:
  - the number of correctly memorized words?
  - the degree of understanding?

Daneman and Carpenter, Individual differences in working memory, 1980
Span test

To assess the working memory construct.
Subjects read sentences.
They are asked to:
  - remember the final words.
  - comprehend the story.

What is:
  - the number of correctly memorized words?
  - the degree of understanding?

Engagement of processing and storage functions.

Daneman and Carpenter, Individual differences in working memory, 1980
‘Computational’ theory of WM

Observation

A trade-off between processing and storage functions.
Observation
A trade-off between processing and storage functions.

Hypothesis
One cognitive resource – competition for a limited capacity.

Daneman and Merikle, Working memory and language comprehension, 1996
Experimental setup

Question

*How additional memory load influences quantifier verification?*
Experimental setup

**Question**

*How additional memory load influences quantifier verification?*

Combined task:

- memorize sequences of digits;
- verify quantifier sentences;
- recall digits.
Results

- Trade-off effect only for PQs.
- WM engagement in PQs is qualitatively different.

Szymanik & Zajenkowski, Quantifiers and working memory, LNCS, 2010
Szymanik & Zajenkowski, Contribution of working memory in the parity and proportional judgments, submitted
Further evidence

- Compare performance of:
  - Healthy subjects.
  - Patients with schizophrenia.
  - Known WM deficits.
Further evidence

- Compare performance of:
  - Healthy subjects.
  - Patients with schizophrenia.
  - Known WM deficits.
Further evidence

- Compare performance of:
  - Healthy subjects.
  - Patients with schizophrenia.
Further evidence

- Compare performance of:
  - Healthy subjects.
  - Patients with schizophrenia.
    - Known WM deficits.
RT data

![Bar chart showing RT data for Patients and Control groups across different categories.](chart.png)
Zajenkowski et al., A computational approach to quantifiers as an explanation for some language impairments in schizophrenia, under review.
Conclusion

Automata model is psychologically plausible.
Summary

**Conclusion**

*Automata model is psychologically plausible.*

**Conclusion**

*Computational complexity ≈ cognitive difficulty.*
Conclusion

*Automata model is psychologically plausible.*

**Conclusion**

*Computational complexity $\approx$ cognitive difficulty.*

- As far as we know this is the first empirical proof.
- Between Marr’s level 1 and 2.
Summary

Conclusion
Automata model is psychologically plausible.

Conclusion
Computational complexity \( \approx \) cognitive difficulty.

▶ As far as we know this is the first empirical proof.
▶ Between Marr’s level 1 and 2.

Conclusion
Precision with proportional quantifiers is cognitively challenging.
Outline

Quantifiers and cognitive strategies
Cognition and computability
Generalized Quantifier Theory
Computing quantifiers
Complexity and reaction time
Complexity and working memory
Outlook
Enrich the model:

1. Approximate Number System;
2. Visual clues;

Pietroski et al., The Meaning of ‘Most’: semantics, numerosity, and psychology, Mind and Language, 1999

Lidz et al., Interface transparency and the psychosemantics of ‘most’, Natural Language Semantics, in press
Neurocognitive computational modeling

- Mechanism selection;
  - ‘more than half’ vs. ‘most’

Dehaene & Cohen, Cultural recycling of cortical maps, Neuron, 2007
Neurocognitive computational modeling

- Mechanism selection;
  - ‘more than half’ vs. ‘most’
- Translate to neurocognitive setting, e.g.;
  - ACT-R modeling;

Dehaene & Cohen, Cultural recycling of cortical maps, Neuron, 2007
Neurocognitive computational modeling

- Mechanism selection;
  - ‘more than half’ vs. ‘most’
- Translate to neurocognitive setting, e.g.:
  - ACT-R modeling;
- Behavioral experiments:
  - determining factors influencing meaning selection.

Dehaene & Cohen, Cultural recycling of cortical maps, Neuron, 2007
Neurocognitive computational modeling

- Mechanism selection;
  - ‘more than half’ vs. ‘most’
- Translate to neurocognitive setting, e.g.:
  - ACT-R modeling;
- Behavioral experiments:
  - determining factors influencing meaning selection.
- fMRI experiments.

Dehaene & Cohen, Cultural recycling of cortical maps, Neuron, 2007
Modeling example

(a) Visual display

- Visual processing

- Approximate number system (ANS)
  - Approx. cardinalities ($b$ and $w$)
  - Cardinality comparison (Is $b > w$?)

(b) Visual display

- Visual processing

- Sample small subsets of ($\leq 4$) dots

- Verification of ‘most dots are black’
  - Subitize
  - Count ‘wins’
  - Cardinality comparison

(c) Visual display

- Visual processing

- Pair each white dot with a black dot

- Non-paired blacks dots

- Test of 1:1 correspondence (black dots left?)

- Verification of ‘most dots are black’
  - true
Big THANKS to Organizers