Compositionality or systematicity?

Christopher Potts

Stanford Linguistics

Franklin Institute Symposium: The Past, Present, and Future of Formal Semantics
April 19, 2021
The thrill of scientific inquiry
Central questions

My central questions

When linguists seek compositional analyses of linguistic phenomena:

- What principles guide their investigations?
- What higher-level goals are they actually pursuing?

Where the questions lead me

- *Learning* semantic representations can lead to richer theories of language and language use,
- but compositionality is too constraining in these situations. *Systematicity* might be a better goal.
Plan

1. The compositionality principle
2. The compositionality heuristic
3. Semantic parsing
4. Recursive deep learning models
5. Conclusions
The compositionality principle

1. The compositionality principle
2. The compositionality heuristic
3. Semantic parsing
4. Recursive deep learning models
5. Conclusions
Informal statement

Compositionality

The meaning of a phrase is a function of the meanings of its immediate syntactic constituents and the way they are combined.
The usual motivation

1. Modeling all meaningful units
   \[ [every] = \lambda f \lambda g \forall x ((f x) \to (g x)) \]
2. “Infinite” capacity
3. Creativity
4. Systematicity
Montague: Unconstrained compositionality


The central idea is that anything that should count as a grammar should be able to be cast in the following form: the syntax is an algebra, the semantics is an algebra, and there is a homomorphism mapping elements of the syntactic algebra onto elements of the semantic algebra.

[...]

It is the homomorphism requirement, which is in effect the compositionality requirement, that provides the most important constraint on UG in Montague’s sense [...].
Formal results

**Zadrozny (1994)**

Any meaning function (map from forms to meanings) can be turned into a compositional one in the sense of the homomorphism requirement.

**Kazmi & Pelletier (1998) respond “Wait, what?”**

Here is a non-compositional semantics:

- \([sleep] = [doze]\)
- \([sleep\ tight] \neq [doze\ tight]\)

Zadrozny showed how to create a kind of syntactic layer where compositionality holds. A similar argument is made by Dever 1999.
Dowty’s context-free compositionality

Dowty (2007):

When a rule $f$ combines $\alpha, \beta(\ldots)$ to form $\gamma$, the corresponding semantic rule $g$ that produces the meaning $\gamma'$ of $\gamma$, from $\alpha'$ and $\beta'$, may depend only on $\alpha'$ “as a whole”, it may not depend on any meanings from which $\alpha'$ was formed compositionally by earlier derivational steps (similarly for $\beta$).

\begin{center}
\begin{tikzpicture}
  \node (gamma) {$\gamma'$} [grow' = right, % left, %right
    level 1/.style = {sibling distance = 8em, level distance = 8em}, %3em
    level 2/.style = {sibling distance = 4em, level distance = 4em}, %2em
  ]
  child {node (alpha) {$\alpha'$} % [grow = left]
    child {node (a1) {$a'_1$}}
    child {node (a2) {$a'_2$}}}
  child {node (beta) {$\beta'$} % [grow = right]
    child {node (b1) {$b'_1$}}
    child {node (b2) {$b'_2$}}};
\end{tikzpicture}
\end{center}
The compositionality heuristic

1. The compositionality principle
2. The compositionality heuristic
3. Semantic parsing
4. Recursive deep learning models
5. Conclusions
Compositionality as methodology

Janssen (1997:461)

Compositionality is not a formal restriction on what can be achieved, but a methodology on how to proceed.
Example: Subjectless predicational adjuncts
Example: Subjectless predicational adjuncts

1. Entering the restaurant, the chef greeted Sandy.
2.

```
entering
     \  \     \  
( the chef ) ( greet sandy )
```

\( \gamma \)
Example: Subjectless predicational adjuncts

1. Entering the restaurant, the chef greeted Sandy.
2. 

Potential rule

The implicit argument of a front subjectless predicational adjuncts must be the subject of the matrix clause.
Example: Subjectless predicational adjuncts

1. Entering the restaurant, the chef greeted Sandy.
2. 

\[
\text{entering} \quad \gamma \\
\text{(the chef)} \quad \text{(greet sandy)}
\]

Potential rule

The implicit argument of a front subjectless predicational adjuncts must be the subject of the matrix clause.

 Assuming context-free compositionality, the rule cannot be correct as stated.
Example: Subjectless predicational adjuncts


1. “Having been in Australia for 17 years, a foreign national wishing to work in Australia must be of good character.”

2. “Fearing a massive lay-off, there was a general sense of relief when the boss announced there would be no new budget cuts.”

3. “Rich and creamy, your guests will never guess that this pie is light.”
Example: Subjectless predicational adjuncts

**Pragmatic constraint**

The free variable in a subjectless predicational adjuncts should refer to a discourse topic.
Example: Subjectless predicational adjuncts

Pragmatic constraint

The free variable in a subjectless predicational adjuncts should refer to a discourse topic.

1. Entering the restaurant, the chef greeted Sandy.
2. 

(entering x) \gamma

(\textit{the chef}) (\textit{greet sandy})
Example: Subjectless predicational adjuncts

Pragmatic constraint

The free variable in a subjectless predicational adjuncts should refer to a discourse topic.

1. Entering the restaurant, the chef greeted Sandy.
2.

\[ \text{(entering x)} \] \[ \gamma \]

\[ \text{(the chef)} \] \[ \text{(greet sandy)} \]

Subjects/topic correlation

In English, subjects are often topics.
Example: Compounds and systematicity


In compounds [...] there is no general rule for predicting the interpretation of the combination.

Levin et al.’s (2019) novel compounds experiment:

<table>
<thead>
<tr>
<th>Modifier</th>
<th>Head</th>
<th>Example</th>
<th>Event</th>
<th>Perceptual Environmental/</th>
</tr>
</thead>
<tbody>
<tr>
<td>Artifact</td>
<td>Artifact</td>
<td>stew skillet</td>
<td>93%</td>
<td>7%</td>
</tr>
<tr>
<td>Natural kind</td>
<td>Artifact</td>
<td>stream wheel</td>
<td>88%</td>
<td>12%</td>
</tr>
<tr>
<td>Artifact</td>
<td>Natural kind</td>
<td>stew chickpea</td>
<td>66%</td>
<td>34%</td>
</tr>
<tr>
<td>Natural kind</td>
<td>Natural kind</td>
<td>stream vegetable</td>
<td>15%</td>
<td>85%</td>
</tr>
</tbody>
</table>

flat tire/beer/note/file
Semantic parsing

1. The compositionality principle
2. The compositionality heuristic
3. **Semantic parsing**
4. Recursive deep learning models
5. Conclusions
The semanticist’s ideal

*Every student attended a lecture*

\[ \forall z \ (\text{student} \ z) \rightarrow (\exists x \ (\text{lecture} \ x) \land (\text{attended} \ x \ z)) \]
The semanticist’s ideal

Every student attended a lecture

$$\exists x \ (\text{lecture } x) \land (\forall z ((\text{student } z) \rightarrow (\text{attended } x z)))$$
The semanticist’s ideal

Every child ate a lollipop

\[ \forall z ((\text{child } z) \rightarrow (\exists x (\text{lollipop } x) \land (\text{ate } x z))) \]
The semanticist’s ideal

Every child ate a lollipop

\[ \exists x \ (\text{lollipop } x) \land (\forall z \ ((\text{child } z) \rightarrow (\text{ate } x \ z))) \]
The semanticist’s ideal

Every child ate a lollipop

\[ \exists x \ (\text{lollipop } x) \land (\forall z \ ((\text{child } z) \rightarrow (\text{ate } x \ z))) \]

But is this really so ideal?
Crude grammars refined via learning
Crude grammars refined via learning

Chat80 (Warren & Pereira 1982):

```/* Sentences */
sentence(S) --> declarative(S), terminator(.) .
sentence(S) --> wh_question(S), terminator(?) .
sentence(S) --> yn_question(S), terminator(?) .
sentence(S) --> imperative(S), terminator(!) .

/* Noun Phrase */
np(np(Agmt,Pronoun,[]),Agmt,NPCase,def,_,Set,Nil) -->
{is_pp(Set)},
pers_pron(Pronoun,Agmt,Case),
{empty(Nil), role(Case,decl,NPCase)}.

/* Prepositional Phrase */
pp(pp(Prep,Arg),Case,Set,Mask) -->
prep(Prep),
{prep_case(NPCase)},
np(Arg,_,NPCase,_,Case,Set,Mask).
```
Crude grammars refined via learning

1. for $w \in \text{Words}$
2. for $X \in \text{Categories}$
3. for $d \in \text{Domain}$
4. yield ‘$X \rightarrow w : d$’
Crude grammars refined via learning

1. \textbf{for} $w \in \text{Words}$
2. \textbf{for} $X \in \text{Categories}$
3. \textbf{for} $d \in \text{Domain}$
4. \textbf{yield} ‘$X \rightarrow w : d$’

0. $N \rightarrow \text{dog} : \text{dog}$
0. $V \rightarrow \text{dog} : \text{dog}_V$
0. $N \rightarrow \text{dog} : \text{cat}$
0. $N \rightarrow \text{cat} : \text{cat}$
0. $N \rightarrow \text{cat} : \text{dog}$
0. $V \rightarrow \text{jump} : \text{dog}$
0. $V \rightarrow \text{jump} : \text{jump}$
Crude grammars refined via learning

\[
\begin{align*}
1 & \text{ for } w \in \text{Words} \\
2 & \quad \text{for } X \in \text{Categories} \\
3 & \quad \quad \text{for } d \in \text{Domain} \\
4 & \quad \quad \text{yield } 'X \rightarrow w : d'
\end{align*}
\]

\[
\begin{align*}
1 & \text{ N } \rightarrow \text{ dog } : \text{ dog} \\
0 & \text{ V } \rightarrow \text{ dog } : \text{ dog}_v \\
0 & \text{ N } \rightarrow \text{ dog } : \text{ cat} \\
0 & \text{ N } \rightarrow \text{ cat } : \text{ cat} \\
0 & \text{ N } \rightarrow \text{ cat } : \text{ dog} \\
0 & \text{ V } \rightarrow \text{ jump } : \text{ dog} \\
0 & \text{ V } \rightarrow \text{ jump } : \text{ jump}
\end{align*}
\]
Crude grammars refined via learning

1. for $w \in \text{Words}$
2. for $X \in \text{Categories}$
3. for $d \in \text{Domain}$
4. yield ‘$X \rightarrow w : d$’

1. $N \rightarrow \text{dog} : \text{dog}$
   - $N \rightarrow \text{dog} : \text{dog}_v$
2. $V \rightarrow \text{dog} : \text{dog}_v$
3. $N \rightarrow \text{dog} : \text{cat}$
4. $N \rightarrow \text{cat} : \text{cat}$
5. $N \rightarrow \text{cat} : \text{dog}$
6. $V \rightarrow \text{jump} : \text{dog}$
7. $V \rightarrow \text{jump} : \text{jump}$
Crude grammars refined via learning

1. for \( w \in \text{Words} \)
2. for \( X \in \text{Categories} \)
3. for \( d \in \text{Domain} \)
4. yield ‘\( X \rightarrow w : d \)’

1. \( N \rightarrow \text{dog} : \text{dog} \)
2. \( V \rightarrow \text{dog} : \text{dog}_v \)
3. \( N \rightarrow \text{dog} : \text{cat} \)
4. \( N \rightarrow \text{cat} : \text{cat} \)
5. \( V \rightarrow \text{jump} : \text{dog} \)
6. \( V \rightarrow \text{jump} : \text{jump} \)
Crude grammars refined via learning

1. for \( w \in \text{Words} \)
2. for \( X \in \text{Categories} \)
3. for \( d \in \text{Domain} \)
4. yield \( 'X \rightarrow w : d' \)

\[
\begin{align*}
1 \quad & N \rightarrow \text{dog} : \text{dog} \\
2 \quad & V \rightarrow \text{dog} : \text{dog}_v \\
0 \quad & N \rightarrow \text{dog} : \text{cat} \\
1 \quad & N \rightarrow \text{cat} : \text{cat} \\
0 \quad & N \rightarrow \text{cat} : \text{dog} \\
0 \quad & V \rightarrow \text{jump} : \text{dog} \\
0 \quad & V \rightarrow \text{jump} : \text{jump}
\end{align*}
\]
Crude grammars refined via learning

1. for \( w \in \text{Words} \)
2. for \( X \in \text{Categories} \)
3. for \( d \in \text{Domain} \)
4. yield ‘\( X \rightarrow w : d \)’

\[
\begin{align*}
1 & \quad N \rightarrow \text{dog} : \text{dog} \\
1 & \quad V \rightarrow \text{dog} : \text{dog}_v \\
0 & \quad N \rightarrow \text{dog} : \text{cat} \\
1 & \quad N \rightarrow \text{cat} : \text{cat} \\
0 & \quad N \rightarrow \text{cat} : \text{dog} \\
0 & \quad V \rightarrow \text{jump} : \text{dog} \\
0 & \quad V \rightarrow \text{jump} : \text{jump} \\
\end{align*}
\]
Crude grammars refined via learning

1. for \( w \in \text{Words} \)
2. for \( X \in \text{Categories} \)
3. for \( d \in \text{Domain} \)
4. yield ‘\( X \rightarrow w : d \)’

3. \( N \rightarrow \text{dog} : \text{dog} \)
1. \( V \rightarrow \text{dog} : \text{dog}_v \)
0. \( N \rightarrow \text{dog} : \text{cat} \)
1. \( N \rightarrow \text{cat} : \text{cat} \)
0. \( N \rightarrow \text{cat} : \text{dog} \)
0. \( V \rightarrow \text{jump} : \text{dog} \)
1. \( V \rightarrow \text{jump} : \text{jump} \)

\( \text{dog} : \text{dog} \)
\( \text{dog} : \text{dog} \)
\( \text{dog} : \text{dog}_v \)
\( \text{jump} : \text{jump} \)
\( \text{cat} : \text{cat} \)
\( \text{dog} : \text{dog} \)
Crude grammars refined via learning

1. for $w \in \text{Words}$
   2. for $X \in \text{Categories}$
   3. for $d \in \text{Domain}$
   4. yield ‘$X \rightarrow w : d$’

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>dog</td>
<td>dog</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>V</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\text{dog}_v$</td>
<td>$\text{jump}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>cat</td>
<td>cat</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\text{dog}$</td>
<td>$\text{dog}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>V</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>dog</td>
<td>dog</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\text{cat}$</td>
<td>$\text{cat}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>dog</td>
<td>dog</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>V</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>jump</td>
<td>jump</td>
</tr>
</tbody>
</table>
Crude grammars refined via learning

1. for $w \in \text{Words}$
2. for $X \in \text{Categories}$
3. for $d \in \text{Domain}$
4. yield $'X \rightarrow w : d'$

<table>
<thead>
<tr>
<th>Rule</th>
<th>LHS</th>
<th>Rhs</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 N</td>
<td>dog</td>
<td>dog</td>
</tr>
<tr>
<td>1 V</td>
<td>dog</td>
<td>dog_v</td>
</tr>
<tr>
<td>0 N</td>
<td>dog</td>
<td>cat</td>
</tr>
<tr>
<td>2 N</td>
<td>cat</td>
<td>cat</td>
</tr>
<tr>
<td>0 N</td>
<td>cat</td>
<td>dog</td>
</tr>
<tr>
<td>0 V</td>
<td>jump</td>
<td>dog</td>
</tr>
<tr>
<td>1 V</td>
<td>jump</td>
<td>jump</td>
</tr>
</tbody>
</table>

\[
\begin{align*}
N & \rightarrow \text{dog} : \text{dog} \\
V & \rightarrow \text{dog} : \text{dog}_v \\
N & \rightarrow \text{dog} : \text{cat} \\
N & \rightarrow \text{cat} : \text{cat} \\
V & \rightarrow \text{dog} : \text{dog} \\
V & \rightarrow \text{jump} : \text{jump} \\
N & \rightarrow \text{dog} : \text{dog} \\
\end{align*}
\]
Crude grammars refined via learning

1. for $w \in \text{Words}$
2. for $X \in \text{Categories}$
3. for $d \in \text{Domain}$
4. yield ‘$X \rightarrow w : d’}$

/* Sentences */
sentence(S) --> declarative(S), terminator(.) .
sentence(S) --> wh_question(S), terminator(?) .
sentence(S) --> yn_question(S), terminator(?) .
sentence(S) --> imperative(S), terminator(!) .

/* Noun Phrase */
np(np(Agmt,Pronoun,[]),Agmt,NPCase,def,_ , Set,Nil) -->
{is_pp(Set)},
pers_pron(Pronoun,Agmt,Case),
{empty(Nil), role(Case,decl,NPCase)}.

/* Prepositional Phrase */
pp(pp(Prep,Arg),Case,Set,Mask) -->
prep(Prep),
{prep_case(NPCase)},
np(Arg,_,NPCase,_,Case,Set,Mask).
Crude grammars refined via learning

Liang et al. (2013):

Some river traverses every city.

(c) Quantifier scope ambiguity (Q, Q)

City traversed by no rivers

(d) Quantification (Q, E)
Additional feature functions

I previously showed some features that correspond to local trees. Those look nicely compositional. However, a smart NLPer will also have features like:

1. How many distinct lexical items are in the sentence?
2. Am I in the c-command domain of a negation?
3. Does this structure contain a specific set of tree fragments?
4. What is the average sentiment of words in this sentence?
The linguist’s ideal again

Every child ate a lollipop

\[ \forall z \ ((\text{child} \ z) \to (\exists x \ (\text{lollipop} \ x) \land (\text{ate} \ x \ z))) \]
The linguist’s ideal again

Every child ate a lollipop

$$\exists x \ (\text{lollipop } x) \land (\forall z \ ((\text{child } z) \rightarrow (\text{ate } x \ z)))$$

Score: $-2$
Recursive deep learning models

1. The compositionality principle
2. The compositionality heuristic
3. Semantic parsing
4. Recursive deep learning models
5. Conclusions
Composition with functions or with vectors

**Functions**

\[
[A] = \mathbf{fa}([B], [C])
\]

\[
[B] [C] = \mathbf{fa}([D], [E])
\]

**Vectors**

\[
f([X]^{DL}, [Y]^{DL}) = \mathbf{g}([X]^{DL}; [Y]^{DL}) W)
\]

\[
[A]^{DL} = \mathbf{g}([B]^{DL}; [C]^{DL}) W)
\]

\[
[B]^{DL} [C]^{DL} = \mathbf{g}([D]^{DL}; [E]^{DL}) W)
\]

**Lexicon**

<table>
<thead>
<tr>
<th></th>
<th>B</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>-0.42</td>
<td>0.10</td>
<td>0.12</td>
</tr>
<tr>
<td>D</td>
<td>-0.16</td>
<td>-0.21</td>
<td>0.29</td>
</tr>
<tr>
<td>E</td>
<td>-0.26</td>
<td>0.31</td>
<td>0.37</td>
</tr>
</tbody>
</table>
Simple example

\[
\tanh \left( \begin{bmatrix} -0.10 & 0.10 & -1.00 & 1.00 \end{bmatrix} \begin{bmatrix} 0.06 & 0.32 \\ -0.14 & -0.53 \\ 1.24 & 0.00 \\ 0.02 & 1.06 \end{bmatrix} \right) = \begin{bmatrix} -0.85 & 0.75 \end{bmatrix}
\]
Recursive deep learning models

\[ S_1 = f(NP_1, VP_1) \]

\[ NP_1 = f(\text{The}, NYT) \]

\[ VP_1 = f(\text{reported}, S_2) \]

\[ S_2 = f(NP_2, VP_2) \]

\[ NP_2 = f(\text{the}, deal) \]

\[ VP_2 = f(\text{fell}, through) \]
Recursive deep learning models

\[ h_1 = f(h_0, \text{The}) \]

\[ h_2 = f(h_1, \text{NYT}) \]

\[ h_3 = f(h_2, \text{reported}) \]

\[ h_4 = f(h_3, \text{the}) \]

\[ h_5 = f(h_4, \text{deal}) \]

\[ h_6 = f(h_5, \text{fell}) \]

\[ h_7 = f(h_6, \text{through}) \]
Recursive deep learning models

\[ h_0 \rightarrow h_1 \rightarrow h_2 \rightarrow h_3 \rightarrow h_4 \rightarrow h_5 \rightarrow h_6 \rightarrow h_7 \]

The NYT reported the deal fell through.
Recursive deep learning models

All our parses are wrong, but perhaps we can discover the right one(s).
A new perspective on compositionality

Partee (1984): Context-dependence, Ambiguity, and Challenges to Local, Deterministic Compositionality
A new perspective on compositionality

Partee (1984):
Context-dependence, Ambiguity, and Challenges to Local, Deterministic Compositionality
A new perspective on compositionality

Partee (1984):
Context-dependence, Ambiguity, and Challenges to Local, Deterministic Compositionality

Global parameters creating local lexical effects

\[ S_1 = \tanh(\text{NP}_1; \text{VP}_1 \ W) \]

\[ \text{NP}_1 \ \text{VP}_1 = \tanh(\text{reported}; \ S_2 \ W) \]

\[ \text{reported} \ \text{S}_2 \]
A new perspective on compositionality

Partee (1984):
Context-dependence, Ambiguity, and Challenges to Local, Deterministic Compositionality

Global parameters creating local lexical effects

\[
\text{tanh} \left( \begin{bmatrix}
-0.10 & 0.10 & -1.00 & 1.00 \\
\end{bmatrix} \right) \cdot \begin{bmatrix}
0.06 & 0.32 \\
-0.14 & -0.53 \\
1.24 & 0.00 \\
0.02 & 1.06 \\
\end{bmatrix} = \begin{bmatrix}
-0.85 & 0.75 \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
-0.10 & 0.10 \\
\end{bmatrix} \quad \begin{bmatrix}
-1.00 & 1.00 \\
\end{bmatrix}
\]

not \quad terrible
The linguist’s ideal again

Every child ate a lollipop

\[ h_L = f(h_2, h_3) \]

\[ h_2 = f(\text{every}, \text{child}) \]

\[ h_3 = f(\text{ate}, h_4) \]

\[ h_4 = f(\text{a}, \text{lollipop}) \]
Examples from Socher et al. 2013

There are repetitive parts slow and

but it

has just enough spice to keep it interesting

and

Therefore

enough spice to keep it interesting

There are repetitive parts slow and

but it

has just enough spice to keep it interesting

and

Therefore
## Conclusions

<table>
<thead>
<tr>
<th>Semantics</th>
<th>Semantic parsing</th>
<th>Deep learning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recursive</td>
<td>Recursive</td>
<td>Recursive</td>
</tr>
<tr>
<td>Symbolic</td>
<td>Symbolic</td>
<td>Not symbolic</td>
</tr>
<tr>
<td>Compositional</td>
<td>Partly compositional</td>
<td>Compositional?</td>
</tr>
<tr>
<td>Precise</td>
<td>Precise</td>
<td>Not precise</td>
</tr>
<tr>
<td>Purely representational</td>
<td>Preferences learned</td>
<td>Preferences learned</td>
</tr>
<tr>
<td>Open-class lexicon often neglected</td>
<td>Learned symbolic lexicon</td>
<td>Learned lexical embedding</td>
</tr>
<tr>
<td>Rich treatment of functional lexicon</td>
<td>Possible rich treatment of functional lexicon</td>
<td>No functional/open-class distinctions</td>
</tr>
<tr>
<td>Sharp sem/prag division</td>
<td>Blurry sem/prag division</td>
<td>No sem/prag division</td>
</tr>
<tr>
<td>Not at all scalable</td>
<td>Semi-scalable</td>
<td>Highly scalable</td>
</tr>
<tr>
<td>Deep analytic insights</td>
<td>Indirect analytic insights</td>
<td>Often opaque</td>
</tr>
</tbody>
</table>

Open-class lexicon often neglected

Rich treatment of functional lexicon

Sharp sem/prag division

Not at all scalable

Deep analytic insights

Learned symbolic lexicon

Learned lexical embedding

Possible rich treatment of functional lexicon

No functional/open-class distinctions

Blurry sem/prag division

No sem/prag division

Semi-scalable

Highly scalable

Indirect analytic insights

Often opaque
Conclusions

Measuring the relative compositionality of natural languages
integrating compositional and statistical properties

Sriram Venkatapathy
Language Technologies Research Centre,
International Institute of Information Technology - Hyderabad,
Hyderabad, India

Deep Recursively Universal Compositionality

Ozan İryoý
Department of Computer Science
Cornell University
Ithaca, NY, USA

Probing Linguistic Compositionality

Emily Goodwin,1,4,5 Koustuv Sinha
1Department of Linguistics, 2School of Computer Science,
3Facebook AI Research (FAIR), Montreal,
4Quebec Artificial Intelligence Laboratory, Montreal,
5Special thanks to the Facebook AI Research (FAIR) program
{emily.goodwin, koustuv.sinha @ facebook.com}

Systematicity and Natural Language?

Kase Bekki3, and Kentaro Inui1,4
1University of Science and Technology,
2Graduate School of Information Science,
3Keio University,
4Tohoku University
kase @ labelar.d.fleto.keio.ac.jp,
ikei @ labelar.d.fleto.keio.ac.jp

Challenges to Systematicity

Eugenie Giesbrecht
Forschungszentrum Informatik
Schloss-und-Neu-Str. 10-14
76131 Karlsruhe, Germany
giesbrecht@fzi.de
References


