The Emergentist Approach To Language

James L. McClelland
Stanford University

The Standard Approach:
Units and Rules

- Sentences
- Clauses and phrases
- Words
- Morphemes
- Phonemes
- Phonological features

- S -> NP VP
- V + {past} -> V+ed
Sentences and Past Tenses

• The boy kicks the ball.
• The baby is sleeping.
• The wind toppled the tree.

• like-liked
• love-loved
• hate-hated

Some possible sentences

• ‘It is cold.’
• ‘It is very cold.’
• ‘It is very very cold.’
• ‘It is very very very cold.’
The Emergentist Approach

• Language arises from the interactions of simple processing units that operate according to simple principles of processing, representation and learning.

• Units and rules are useful to *approximately describe* what emerges from these interactions but have no mechanistic or explanatory role in language processing, language change, or language learning.

An Emergentist Theory: Natural Selection

• No grand design.
• Forces of nature select those best prepared to survive.
• Survivors leave more offspring.
• The full range of the animal kingdom including all the capabilities of the human mind emerge from these very basic principles.
A Pattern Associator Network

Pattern representing sound of the verb’s past tense

Pattern representing the sound of the verb’s stem

Learning rule for the Pattern Associator network

• For each output unit:
  – Determine activity of the unit based on its input.
  – If the unit is inactive when the target is active:
    • Increase the weight coming into the unit from each active input unit.
  – If the unit is active when target is not:
    • Reduce each weight coming into the unit from each active input unit.
Ten most frequent past tenses in English

- Felt
- Had
- Made
- Got
- Gave
- Took
- Came
- Went
- Looked
- Needed

Strength for correct vs. regularized output

Here's where 400 more words were introduced

Trained with top ten words only.

Some features of the model

- Regulars co-exist with exceptions.
- The model produces the regular past for most unfamiliar test items.
- The model captures the different subtypes among the regulars:
  - like-liked
  - love-liked
  - hate-hated
- The model is sensitive to the no-change pattern among irregulars:
  - hit-hit
  - cut-cut
  - hide-hid

REGULAR AND NO CHANGE RESPONSES TO t/d AND OTHER VERBS
(Data from Bybee & Slobin, 1982)

<table>
<thead>
<tr>
<th>Verb Ending</th>
<th>Regular Suffix</th>
<th>No Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not t/d</td>
<td>203</td>
<td>34</td>
</tr>
<tr>
<td>t/d</td>
<td>42</td>
<td>157</td>
</tr>
</tbody>
</table>
Additional characteristics

• The model exploits gangs of related exceptions.
  – dig-dug
  – cling-clung
  – swing-swung

• The ‘regular pattern’ infuses exceptions as well as regulars:
  – say-said, do-did
  – have-had
  – keep-kept, sleep-slept
  – Burn-burnt
  – Teach-taught

Key features of the Past Tense model

• No lexical entries and no rules

• No problem of rule selection
Elman’s Simple Recurrent Network

- Task is to predict the next element of a sequence on the output, given the current element on the input units.
- Each element is represented by a pattern of activation.
- Each box represents a set of units.
- Each dotted arrow represents all-to-all connections.
- The solid arrow indicates that the previous pattern on the hidden units is copied back to provide context for the next prediction.
- Learning occurs through connection weight adjustment using an extended version of the error correcting learning rule.

Results for Elman net trained with letter sequences

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>01010 (m)</td>
<td>00001 (a)</td>
</tr>
<tr>
<td>00001 (a)</td>
<td>01110 (a)</td>
</tr>
<tr>
<td>01110 (m)</td>
<td>11001 (y)</td>
</tr>
<tr>
<td>11001 (y)</td>
<td>11001 (y)</td>
</tr>
<tr>
<td>11001 (y)</td>
<td>00101 (e)</td>
</tr>
<tr>
<td>00101 (e)</td>
<td>00001 (a)</td>
</tr>
<tr>
<td>00001 (a)</td>
<td>10010 (r)</td>
</tr>
<tr>
<td>10010 (r)</td>
<td>10001 (x)</td>
</tr>
<tr>
<td>10011 (a)</td>
<td>00001 (a)</td>
</tr>
<tr>
<td>00001 (a)</td>
<td>00111 (g)</td>
</tr>
<tr>
<td>00111 (g)</td>
<td>01111 (e)</td>
</tr>
<tr>
<td>01111 (e)</td>
<td>00001 (a)</td>
</tr>
<tr>
<td>00001 (a)</td>
<td>00010 (b)</td>
</tr>
<tr>
<td>00010 (b)</td>
<td>01111 (e)</td>
</tr>
<tr>
<td>01111 (e)</td>
<td>11001 (y)</td>
</tr>
<tr>
<td>11001 (y)</td>
<td>00001 (a)</td>
</tr>
<tr>
<td>00001 (a)</td>
<td>01110 (n)</td>
</tr>
<tr>
<td>01110 (n)</td>
<td>00100 (d)</td>
</tr>
<tr>
<td>00100 (d)</td>
<td>00111 (g)</td>
</tr>
<tr>
<td>00111 (g)</td>
<td>01100 (l)</td>
</tr>
<tr>
<td>01100 (l)</td>
<td>10010 (r)</td>
</tr>
<tr>
<td>10010 (r)</td>
<td>01100 (l)</td>
</tr>
<tr>
<td>01100 (l)</td>
<td>01100 (l)</td>
</tr>
<tr>
<td>11001 (l)</td>
<td>11001 (l)</td>
</tr>
</tbody>
</table>
Hidden Unit Patterns for Elman Net Trained on Word Sequences

Some whole and partial sentences

• The dog likes ice cream.

• The boys who chased the dog like ice cream.

• He hit the ball out of the …

• The wheezing old man looked like he was about to kick the …
Key Features of the Both Models

• No lexical entries and no rules

• No problem of rule selection