Word embedding
What is word embedding?

• Represent a word as a d-dimensional vector:
  ▪ 1-to-N coding
  ▪ Vectors created by a system such as word2vec

• Similar words should have similar embedding.

• Recent work on embedding of phrases, sentences, documents, etc.
Many studies

• Early studies: (Hinton 1986), (Pollack 1990), (Elman 1991), etc.

• Feed-forward networks: (Bengio et al., 2003; 2006)

• Recurrent neural networks: (Mikolov et al., 2010; 2011; 2013)

• Now: tons of papers in 2014-now
Why embedding?

- LM: used for ASR, MT, etc.

- Supervised prediction tasks: POS tagging, chunking, NER, etc.

- Answer semantic questions: e.g., A:B is like C:D

- ...
Recurrent Neural Network LM (Mikolov et al., 2010)

Input layer is $w(t)$ and $s(t-1)$
Hidden layer represents context $s(t)$
Output layer is the prob distribution of $w(t+1)$

$W(t)$ uses 1-to-N coding.

Training is slow
→ Cannot use large amount of data
Table 1: *Performance of models on WSJ DEV set when increasing size of training data.*

<table>
<thead>
<tr>
<th>Model</th>
<th># words</th>
<th>PPL</th>
<th>WER</th>
</tr>
</thead>
<tbody>
<tr>
<td>KN5 LM</td>
<td>200K</td>
<td>336</td>
<td>16.4</td>
</tr>
<tr>
<td>KN5 LM + RNN 90/2</td>
<td>200K</td>
<td>271</td>
<td>15.4</td>
</tr>
<tr>
<td>KN5 LM</td>
<td>1M</td>
<td>287</td>
<td>15.1</td>
</tr>
<tr>
<td>KN5 LM + RNN 90/2</td>
<td>1M</td>
<td>225</td>
<td>14.0</td>
</tr>
<tr>
<td>KN5 LM</td>
<td>6.4M</td>
<td>221</td>
<td>13.5</td>
</tr>
<tr>
<td>KN5 LM + RNN 250/5</td>
<td>6.4M</td>
<td>156</td>
<td>11.7</td>
</tr>
</tbody>
</table>
A semantic task (Mikolov et al., 2013)

• Task: “A:B  C:D”
  • Training: given a corpus of text, build a “model”
  • Test: given A, B, and C, find D

• Examples:
  • Good:better  rough:___
  • Year:years  law:___
  • See:sees  return:___
  • come:go  borrow:___
Algorithm

• A:B is like C:D
  \[ x_b - x_a = x_d - x_c \]
  \[ x_b - x_a + x_c = x_d \]

• Represent each word \( w \) as a word vector \( x_w \)
• Compute \( y = x_b - x_a + x_c \)
• Find \( w = \text{arg max}_w \; \text{sim}(x_w, y) \)
## Results

<table>
<thead>
<tr>
<th>Method</th>
<th>Adjectives</th>
<th>Nouns</th>
<th>Verbs</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSA-80</td>
<td>9.2</td>
<td>11.1</td>
<td>17.4</td>
<td>12.8</td>
</tr>
<tr>
<td>LSA-320</td>
<td>11.3</td>
<td>18.1</td>
<td>20.7</td>
<td>16.5</td>
</tr>
<tr>
<td>LSA-640</td>
<td>9.6</td>
<td>10.1</td>
<td>13.8</td>
<td>11.3</td>
</tr>
<tr>
<td>RNN-80</td>
<td>9.3</td>
<td>5.2</td>
<td>30.4</td>
<td>16.2</td>
</tr>
<tr>
<td>RNN-320</td>
<td>18.2</td>
<td>19.0</td>
<td>45.0</td>
<td>28.5</td>
</tr>
<tr>
<td>RNN-640</td>
<td>21.0</td>
<td>25.2</td>
<td>54.8</td>
<td>34.7</td>
</tr>
<tr>
<td>RNN-1600</td>
<td>23.9</td>
<td>29.2</td>
<td>62.2</td>
<td>39.6</td>
</tr>
</tbody>
</table>
Summary

• Word embedding is to represent a word as a vector

• It has shown significant improvement on several NLP tasks.

• There are tons of recent studies on this. To find out more, go to ACL anthology, and look at recent proceedings (e.g., ACL 2015).