Natural Language Processing
Lexical semantics

Sofia Serrano
sofias6@cs.washington.edu

Credit to Yulia Tsvetkov and Noah Smith for slides
Announcements

● Quiz 3 will be released on Canvas today at 2:20pm
  ○ Available through Thursday 2:20pm
  ○ 5 questions, 10 minutes
  ○ Will cover material from Wednesday, Friday, and Monday (so, language modeling and the first part of lexical semantics)

● Midterm course eval form (online) is out– please let us know how we're doing!
Two common solutions for word weighting

**tf-idf**: tf-idf value for word \( t \) in document \( d \):

\[
w_{t,d} = tf_{t,d} \times idf_t
\]

Words like “the” or “it” have very low idf

**PMI**: Pointwise mutual information

\[
PMI(w_1, w_2) = \log \frac{p(w_1, w_2)}{p(w_1)p(w_2)}
\]

See if words like “good” appear more often with “great” than we would expect by chance
What to do with words that are evenly distributed across many documents?

\[ tf_{t,d} = \log_{10}(\text{count}(t,d) + 1) \]

\[ idf_i = \log \left( \frac{N}{df_i} \right) \]

Words like "the" or "good" have very low idf

\[ w_{t,d} = tf_{t,d} \times idf_i \]
Positive Pointwise Mutual Information (PPMI)

- In word-context matrix
- Do words $w$ and $c$ co-occur more than if they were independent?

$$\text{PMI}(w, c) = \log_2 \frac{P(w, c)}{P(w)P(c)}$$

$$\text{PPMI}(w, c) = \max(\log_2 \frac{P(w, c)}{P(w)P(c)}, 0)$$

- PMI is biased toward infrequent events
  - Very rare words have very high PMI values
  - Give rare words slightly higher probabilities $\alpha=0.75$

$$\text{PPMI}_\alpha(w, c) = \max(\log_2 \frac{P(w, c)}{P(w)P_\alpha(c)}, 0)$$

$$P_\alpha(c) = \frac{\text{count}(c)^\alpha}{\sum_c \text{count}(c)^\alpha}$$
<table>
<thead>
<tr>
<th>#</th>
<th>name</th>
<th>formula</th>
<th>reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Joint probability</td>
<td>( p(y</td>
<td>x) )</td>
</tr>
<tr>
<td>2.</td>
<td>Conditional probability</td>
<td>( p(y</td>
<td>x) )</td>
</tr>
<tr>
<td>3.</td>
<td>Reverse cond. probability</td>
<td>( p(x</td>
<td>y) )</td>
</tr>
<tr>
<td>4.</td>
<td>Pointwise mutual inf. (MI)</td>
<td>( \log \frac{p(x,y)}{p(x)p(y)} )</td>
<td>(Church and Hanks, 1990)</td>
</tr>
<tr>
<td>5.</td>
<td>Mutual dependency (MD)</td>
<td>( \frac{\log p(x</td>
<td>y)}{\log p(x)} )</td>
</tr>
<tr>
<td>6.</td>
<td>Log frequency biased MD</td>
<td>( \log \frac{p(x</td>
<td>y)}{p(x)} )</td>
</tr>
<tr>
<td>7.</td>
<td>Normalized expectation</td>
<td>( \frac{\log p(x)}{p(x)} )</td>
<td>(Smajdor and McKenney, 1990)</td>
</tr>
<tr>
<td>8.</td>
<td>Mutual expectation</td>
<td>( \log p(x,y) )</td>
<td>(Dias et al., 2000)</td>
</tr>
<tr>
<td>9.</td>
<td>Salience</td>
<td>( \log \frac{p(x)}{p(x)} )</td>
<td>(Kilgarriff and Tugwell, 2001)</td>
</tr>
<tr>
<td>10.</td>
<td>Pearson's ( \chi^2 ) test</td>
<td>( \sum \frac{(x_i-f_i)^2}{f_i} )</td>
<td>(Manning and Schütze, 1999)</td>
</tr>
<tr>
<td>11.</td>
<td>Fisher's exact test</td>
<td>( \frac{n!}{x!y!(n-x-y)!} )</td>
<td>(Pedersen, 1996)</td>
</tr>
<tr>
<td>12.</td>
<td>( t ) test</td>
<td>( \left( \frac{x - y}{\sqrt{\frac{p(x)p(y) + p(x)p(y) + p(x)p(y) + p(x)p(y)}}} \right) )</td>
<td>(Church and Hanks, 1990)</td>
</tr>
<tr>
<td>13.</td>
<td>( z ) score</td>
<td>( \frac{\sqrt{\frac{p(x)p(y) + p(x)p(y) + p(x)p(y) + p(x)p(y) - \frac{p(x)p(y) + p(x)p(y) + p(x)p(y) + p(x)p(y)}}{p(x)p(y) + p(x)p(y) + p(x)p(y) + p(x)p(y)}}} )</td>
<td>(Berry-Rogge, 1973)</td>
</tr>
<tr>
<td>14.</td>
<td>Poisson significance</td>
<td>( \log \frac{p(x)}{p(x)} )</td>
<td>(Quasshoff and Wolf, 2002)</td>
</tr>
<tr>
<td>15.</td>
<td>Log likelihood ratio</td>
<td>( -2 \sum x_i \log f_i )</td>
<td>(Dunning, 1993)</td>
</tr>
<tr>
<td>16.</td>
<td>Squared log likelihood ratio</td>
<td>( -2 \sum x_i \log f_i^2 )</td>
<td>(Inkpen and Hirthe, 2002)</td>
</tr>
<tr>
<td>17.</td>
<td>Russell-Rao</td>
<td>( \frac{a+b+c}{a+b+c} )</td>
<td>(Russel and Rao, 1940)</td>
</tr>
<tr>
<td>18.</td>
<td>Sokal-Michener</td>
<td>( \frac{a+b+c}{a+b+c} )</td>
<td>(Sokal and Michener, 1958)</td>
</tr>
<tr>
<td>19.</td>
<td>Rogers-Tanimoto</td>
<td>( \frac{a+b+c}{a+b+c} )</td>
<td>(Rogers and Tanimoto, 1966)</td>
</tr>
<tr>
<td>20.</td>
<td>Hamann</td>
<td>( \frac{a+b+c}{a+b+c} )</td>
<td>(Hamann, 1961)</td>
</tr>
<tr>
<td>21.</td>
<td>Third Sokal-Sneath</td>
<td>( \frac{a+b+c}{a+b+c} )</td>
<td>(Sokal and Sneath, 1963)</td>
</tr>
<tr>
<td>22.</td>
<td>Jaccard</td>
<td>( \frac{a}{a+b+c} )</td>
<td>(Jaccard, 1912)</td>
</tr>
<tr>
<td>23.</td>
<td>First Kulczynski</td>
<td>( \frac{a+b}{a+b} )</td>
<td>(Kulczynski, 1927)</td>
</tr>
<tr>
<td>24.</td>
<td>Second Sokal-Sneath</td>
<td>( \frac{a+b+c}{a+b+c} )</td>
<td>(Sokal and Sneath, 1963)</td>
</tr>
<tr>
<td>25.</td>
<td>Second Kulczynski</td>
<td>( \frac{a+b+c}{a+b+c} )</td>
<td>(Kulczynski, 1927)</td>
</tr>
<tr>
<td>26.</td>
<td>Fourth Sokal-Sneath</td>
<td>( \frac{a+b+c}{a+b+c} )</td>
<td>(Kulczynski, 1927)</td>
</tr>
<tr>
<td>27.</td>
<td>Odds ratio</td>
<td>( \frac{a}{b} )</td>
<td>(Tan et al., 2002)</td>
</tr>
<tr>
<td>28.</td>
<td>Yule's ( \omega )</td>
<td>( \frac{a-b}{a+b} )</td>
<td>(Tan et al., 2002)</td>
</tr>
<tr>
<td>29.</td>
<td>Yule's ( Q )</td>
<td>( \frac{a-b}{a+b} )</td>
<td>(Tan et al., 2002)</td>
</tr>
<tr>
<td>30.</td>
<td>Driver-Kroeber</td>
<td>( \sqrt{a+b+c+d} )</td>
<td>(Driver and Kroeber, 1932)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>#</th>
<th>name</th>
<th>formula</th>
<th>reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>31.</td>
<td>Fifth Sokal-Sneath</td>
<td>( \sqrt{a+b+c+d+e} )</td>
<td>(Sokal and Sneath, 1963)</td>
</tr>
<tr>
<td>32.</td>
<td>Pearson</td>
<td>( \frac{a-b}{a+b+c+e} )</td>
<td>(Pearson, 1950)</td>
</tr>
<tr>
<td>33.</td>
<td>Baroni-Urani</td>
<td>( \frac{a}{a+b+c+e} )</td>
<td>(Baroni-Urani and Buser, 1976)</td>
</tr>
<tr>
<td>34.</td>
<td>Braun-Blanquet</td>
<td>( \frac{a}{a+b+c+e} )</td>
<td>(Braun-Blanquet, 1932)</td>
</tr>
<tr>
<td>35.</td>
<td>Simpson</td>
<td>( \frac{a}{a+b+c+e} )</td>
<td>(Simpson, 1943)</td>
</tr>
<tr>
<td>36.</td>
<td>Michael</td>
<td>( \frac{a}{a+b+c+e} )</td>
<td>(Michael, 1920)</td>
</tr>
<tr>
<td>37.</td>
<td>Mountford</td>
<td>( \frac{a}{a+b+c+e} )</td>
<td>(Kaufman and Rousseeuw, 1990)</td>
</tr>
<tr>
<td>38.</td>
<td>Fager</td>
<td>( \frac{a}{a+b+c+e} )</td>
<td>(Kaufman and Rousseeuw, 1990)</td>
</tr>
<tr>
<td>39.</td>
<td>Unigram subtuples</td>
<td>( \log \frac{a+b+c+e}{a+b+c+e} )</td>
<td>(Bhat and Johnson, 2001)</td>
</tr>
<tr>
<td>40.</td>
<td>( U ) cost</td>
<td>( \log(1+\frac{a+b+c+e}{a+b+c+e}) )</td>
<td>(Tulloss, 1997)</td>
</tr>
<tr>
<td>41.</td>
<td>( S ) cost</td>
<td>( \log(1+\frac{a+b+c+e}{a+b+c+e}) )</td>
<td>(Tulloss, 1997)</td>
</tr>
<tr>
<td>42.</td>
<td>( R ) cost</td>
<td>( \log(1+\frac{a+b+c+e}{a+b+c+e}) )</td>
<td>(Tulloss, 1997)</td>
</tr>
<tr>
<td>43.</td>
<td>( T ) combined cost</td>
<td>( \sqrt{U+X+Y+Z} )</td>
<td>(Tulloss, 1997)</td>
</tr>
<tr>
<td>44.</td>
<td>( \Phi )</td>
<td>( \sqrt{\frac{a}{a+b+c+e} \cdot \frac{b}{a+b+c+e}} )</td>
<td>(Tan et al., 2002)</td>
</tr>
<tr>
<td>45.</td>
<td>Kappa</td>
<td>( \frac{a}{a+b+c+e} )</td>
<td>(Tan et al., 2002)</td>
</tr>
<tr>
<td>46.</td>
<td>( J ) measure</td>
<td>( \max[p(x), p(y)] \log \frac{p(x)+p(y)}{p(x)+p(y)} + p(x)+p(y) - p(x)+p(y) )</td>
<td>(Tan et al., 2002)</td>
</tr>
<tr>
<td>47.</td>
<td>Gini index</td>
<td>( \max[p(x)+p(y)] \log \frac{p(x)+p(y)}{p(x)+p(y)} + p(x)+p(y) - p(x)+p(y) )</td>
<td>(Tan et al., 2002)</td>
</tr>
<tr>
<td>48.</td>
<td>Confidence</td>
<td>( \max[p(x), p(y)] \log \frac{p(x)+p(y)}{p(x)+p(y)} + p(x)+p(y) - p(x)+p(y) )</td>
<td>(Tan et al., 2002)</td>
</tr>
<tr>
<td>49.</td>
<td>Laplace</td>
<td>( \max[p(x)+p(y)] \log \frac{p(x)+p(y)}{p(x)+p(y)} + p(x)+p(y) - p(x)+p(y) )</td>
<td>(Tan et al., 2002)</td>
</tr>
<tr>
<td>50.</td>
<td>Conviction</td>
<td>( \max[p(x)+p(y)] \log \frac{p(x)+p(y)}{p(x)+p(y)} + p(x)+p(y) - p(x)+p(y) )</td>
<td>(Tan et al., 2002)</td>
</tr>
<tr>
<td>51.</td>
<td>Platersky-Shapiro</td>
<td>( \frac{p(x), p(y)}{p(x)+p(y)} )</td>
<td>(Tan et al., 2002)</td>
</tr>
<tr>
<td>52.</td>
<td>Certainty factor</td>
<td>( \max[p(x)] - p(x) )</td>
<td>(Tan et al., 2002)</td>
</tr>
<tr>
<td>53.</td>
<td>Added value (AV)</td>
<td>( \max[p(x) - p(x), p(y) - p(x)] )</td>
<td>(Tan et al., 2002)</td>
</tr>
<tr>
<td>54.</td>
<td>Collective strength</td>
<td>( \max[p(x) - p(x), p(y) - p(x)] )</td>
<td>(Tan et al., 2002)</td>
</tr>
<tr>
<td>55.</td>
<td>Klosgen</td>
<td>( \sqrt{p(x) \cdot AV} )</td>
<td>(Tan et al., 2002)</td>
</tr>
</tbody>
</table>
Dense vectors (part 1)
These word vectors are still the length of our number of documents! Hmmm...
Dimensionality Reduction

  - High dimensionality of word--document matrix
    - Sparsity
    - The order of rows and columns doesn’t matter
- Goal:
  - good similarity measure for words or documents
  - dense representation
- Sparse vs Dense vectors
  - Short vectors may be easier to use as features in machine learning (less weights to tune)
  - Dense vectors may generalize better than storing explicit counts
    - They may do better at capturing synonymy
    - In practice, they work better
Solution idea

- Find a projection into a low-dimensional space (~300 dim)...
- ... that, up to a certain vector-length budget, preserves the most important information

We turn to Singular Value Decomposition (SVD)
Singular Value Decomposition in a nutshell

Any matrix can be decomposed into

\[ \begin{pmatrix} U & \text{?} \end{pmatrix} \begin{pmatrix} \text{?} & \text{?} \\ \text{?} & \text{?} \end{pmatrix} \begin{pmatrix} V^T & \text{?} \end{pmatrix} \]

- Orthonormal, unitary
- (Rectangular) diagonal
- Orthonormal, unitary
Singular Value Decomposition in a nutshell

Any matrix can be decomposed into

Rotation
Orthonormal, unitary

Scaling
(Rectangular) diagonal

Rotation
Orthonormal, unitary
Singular Value Decomposition in a nutshell

Let’s trim away the zero scaling factors

\[ U \]

Rotation

Orthonormal, unitary

\[ \text{(Rectangular) diagonal} \]

Scaling

In order of decreasing magnitude

\[ V^T \]

Rotation

Orthonormal, unitary
Singular Value Decomposition in a nutshell

\[ \text{Documents: } \begin{bmatrix} A \end{bmatrix}_{m \times n} = \begin{bmatrix} U \end{bmatrix}_{m \times r} \begin{bmatrix} \Sigma \end{bmatrix}_{r \times r} \begin{bmatrix} V^T \end{bmatrix}_{r \times n} \]

orthonormal \quad \text{diagonal, sorted
Truncated SVD

We can approximate the full matrix by only considering the leftmost $k$ terms in the diagonal matrix (the $k$ largest singular values)

$$A_{m \times n} \approx U_{m \times k} \Sigma_{k \times k} V_{k \times n}^T$$

$k \ll m, n$
Latent Semantic Analysis

<table>
<thead>
<tr>
<th>#0</th>
<th>#1</th>
<th>#2</th>
<th>#3</th>
<th>#4</th>
<th>#5</th>
</tr>
</thead>
<tbody>
<tr>
<td>we</td>
<td>music</td>
<td>company</td>
<td>how</td>
<td>program</td>
<td>10</td>
</tr>
<tr>
<td>said</td>
<td>film</td>
<td>mr</td>
<td>what</td>
<td>project</td>
<td>30</td>
</tr>
<tr>
<td>have</td>
<td>theater</td>
<td>its</td>
<td>about</td>
<td>russian</td>
<td>11</td>
</tr>
<tr>
<td>they</td>
<td>mr</td>
<td>inc</td>
<td>their</td>
<td>space</td>
<td>12</td>
</tr>
<tr>
<td>not</td>
<td>this</td>
<td>stock</td>
<td>or</td>
<td>russia</td>
<td>15</td>
</tr>
<tr>
<td>but</td>
<td>who</td>
<td>companies</td>
<td>this</td>
<td>center</td>
<td>13</td>
</tr>
<tr>
<td>be</td>
<td>movie</td>
<td>sales</td>
<td>are</td>
<td>programs</td>
<td>14</td>
</tr>
<tr>
<td>do</td>
<td>which</td>
<td>shares</td>
<td>history</td>
<td>clark</td>
<td>20</td>
</tr>
<tr>
<td>he</td>
<td>show</td>
<td>said</td>
<td>be</td>
<td>aircraft</td>
<td>sept</td>
</tr>
<tr>
<td>this</td>
<td>about</td>
<td>business</td>
<td>social</td>
<td>ballet</td>
<td>16</td>
</tr>
<tr>
<td>there</td>
<td>dance</td>
<td>share</td>
<td>these</td>
<td>its</td>
<td>25</td>
</tr>
<tr>
<td>you</td>
<td>its</td>
<td>chief</td>
<td>other</td>
<td>projects</td>
<td>17</td>
</tr>
<tr>
<td>are</td>
<td>disney</td>
<td>executive</td>
<td>research</td>
<td>orchestra</td>
<td>18</td>
</tr>
<tr>
<td>what</td>
<td>play</td>
<td>president</td>
<td>writes</td>
<td>development</td>
<td>19</td>
</tr>
<tr>
<td>if</td>
<td>production</td>
<td>group</td>
<td>language</td>
<td>work</td>
<td>21</td>
</tr>
</tbody>
</table>

(Deerwester et al., 1990)
How do we tell whether a set of word embeddings is any good?
Evaluation

- Intrinsic
- Extrinsic
- Qualitative

<table>
<thead>
<tr>
<th>WORD</th>
<th>d1</th>
<th>d2</th>
<th>d3</th>
<th>d4</th>
<th>d5</th>
<th>…</th>
<th>d50</th>
</tr>
</thead>
<tbody>
<tr>
<td>summer</td>
<td>0.12</td>
<td>0.21</td>
<td>0.07</td>
<td>0.25</td>
<td>0.33</td>
<td>…</td>
<td>0.51</td>
</tr>
<tr>
<td>spring</td>
<td>0.19</td>
<td>0.57</td>
<td>0.99</td>
<td>0.30</td>
<td>0.02</td>
<td>…</td>
<td>0.73</td>
</tr>
<tr>
<td>fall</td>
<td>0.53</td>
<td>0.77</td>
<td>0.43</td>
<td>0.20</td>
<td>0.29</td>
<td>…</td>
<td>0.85</td>
</tr>
<tr>
<td>light</td>
<td>0.00</td>
<td>0.68</td>
<td>0.84</td>
<td>0.45</td>
<td>0.11</td>
<td>…</td>
<td>0.03</td>
</tr>
<tr>
<td>clear</td>
<td>0.27</td>
<td>0.50</td>
<td>0.21</td>
<td>0.56</td>
<td>0.25</td>
<td>…</td>
<td>0.32</td>
</tr>
<tr>
<td>blizzard</td>
<td>0.15</td>
<td>0.05</td>
<td>0.64</td>
<td>0.17</td>
<td>0.99</td>
<td>…</td>
<td>0.23</td>
</tr>
</tbody>
</table>
Extrinsic Evaluation

- Chunking
- POS tagging
- Parsing
- MT
- SRL
- Topic categorization
- Sentiment analysis
- Metaphor detection
- etc.
## Intrinsic Evaluation

- **WS-353** ([Finkelstein et al. ‘02](#))
- **MEN-3k** ([Bruni et al. ‘12](#))
- **SimLex-999 dataset** ([Hill et al., 2015](#))

<table>
<thead>
<tr>
<th>word1</th>
<th>word2</th>
<th>similarity (humans)</th>
<th>similarity (embeddings)</th>
</tr>
</thead>
<tbody>
<tr>
<td>vanish</td>
<td>disappear</td>
<td>9.8</td>
<td>1.1</td>
</tr>
<tr>
<td>behave</td>
<td>obey</td>
<td>7.3</td>
<td>0.5</td>
</tr>
<tr>
<td>belief</td>
<td>impression</td>
<td>5.95</td>
<td>0.3</td>
</tr>
<tr>
<td>muscle</td>
<td>bone</td>
<td>3.65</td>
<td>1.7</td>
</tr>
<tr>
<td>modest</td>
<td>flexible</td>
<td>0.98</td>
<td>0.98</td>
</tr>
<tr>
<td>hole</td>
<td>agreement</td>
<td>0.3</td>
<td>0.3</td>
</tr>
</tbody>
</table>

Spearman's rho (human ranks, model ranks)
Computing word similarity

The dot product between two vectors is a scalar:

\[
dot \text{product}(v, w) = v \cdot w = \sum_{i=1}^{N} v_i w_i = v_1 w_1 + v_2 w_2 + \ldots + v_N w_N
\]

- The dot product tends to be high when the two vectors have large values in the same dimensions
- Dot product can thus be a useful similarity metric between vectors
Problem with raw dot-product

- Dot product favors long vectors
  - Dot product is higher if a vector is longer (has higher values in many dimensions). Vector length:

\[ |v| = \sqrt{\sum_{i=1}^{N} v_i^2} \]

- Frequent words (of, the, you) have long vectors (since they occur many times with other words).
  - So dot product overly favors frequent words
Alternative: cosine for computing word similarity

\[
\cosine(v, w) = \frac{\vec{v} \cdot \vec{w}}{||v|| \cdot ||w||} = \frac{\sum_{i=1}^{N} v_i w_i}{\sqrt{\sum_{i=1}^{N} v_i^2} \sqrt{\sum_{i=1}^{N} w_i^2}}
\]

Based on the definition of the dot product between two vectors \(a\) and \(b\)

\[
a \cdot b = |a||b| \cos \theta
\]

\[
\frac{a \cdot b}{|a||b|} = \cos \theta
\]
Cosine as a similarity metric

-1: vectors point in opposite directions
+1: vectors point in same directions
0: vectors are orthogonal

- But since raw frequency values are non-negative, the cosine for term-term matrix vectors ranges from 0–1
Cosine examples

\[
\cos(\vec{v}, \vec{w}) = \frac{\vec{v} \cdot \vec{w}}{|\vec{v}| |\vec{w}|} = \frac{\vec{v} \cdot \vec{w}}{\sqrt{\sum_{i=1}^{N} v_i^2} \sqrt{\sum_{i=1}^{N} w_i^2}}
\]

<table>
<thead>
<tr>
<th></th>
<th>pie</th>
<th>data</th>
<th>computer</th>
</tr>
</thead>
<tbody>
<tr>
<td>cherry</td>
<td>442</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>digital</td>
<td>114</td>
<td>80</td>
<td>62</td>
</tr>
<tr>
<td>information</td>
<td>36</td>
<td>58</td>
<td>1</td>
</tr>
</tbody>
</table>

\[
\cos(\text{cherry, information}) = \frac{442 \times 5 + 8 \times 3982 + 2 \times 3325}{\sqrt{442^2 + 8^2} \sqrt{5^2 + 3982^2 + 3325^2}} = .017
\]

\[
\cos(\text{digital, information}) = \frac{5 \times 5 + 1683 \times 3982 + 1670 \times 3325}{\sqrt{5^2 + 1683^2} \sqrt{5^2 + 3982^2 + 3325^2}} = .996
\]
Visualizing angles

Dimension 1: ‘pie’

500

digital

information

Dimension 2: ‘computer’
Visualisation

Figure 6.5: Monolingual (top) and multilingual (bottom; marked with apostrophe) word projections of the antonyms (shown in red) and synonyms of “beautiful”.

- Visualizing Data using t-SNE (van der Maaten & Hinton ’08)
Dense vectors (part 2)
# Distributed representations

## Word Vectors

<table>
<thead>
<tr>
<th>WORD</th>
<th>d1</th>
<th>d2</th>
<th>d3</th>
<th>d4</th>
<th>d5</th>
<th>...</th>
<th>d50</th>
</tr>
</thead>
<tbody>
<tr>
<td>summer</td>
<td>0.12</td>
<td>0.21</td>
<td>0.07</td>
<td>0.25</td>
<td>0.33</td>
<td></td>
<td>0.51</td>
</tr>
<tr>
<td>spring</td>
<td>0.19</td>
<td>0.57</td>
<td>0.99</td>
<td>0.30</td>
<td>0.02</td>
<td></td>
<td>0.73</td>
</tr>
<tr>
<td>fall</td>
<td>0.53</td>
<td>0.77</td>
<td>0.43</td>
<td>0.20</td>
<td>0.29</td>
<td></td>
<td>0.85</td>
</tr>
<tr>
<td>light</td>
<td>0.00</td>
<td>0.68</td>
<td>0.84</td>
<td>0.45</td>
<td>0.11</td>
<td></td>
<td>0.03</td>
</tr>
<tr>
<td>clear</td>
<td>0.27</td>
<td>0.50</td>
<td>0.21</td>
<td>0.56</td>
<td>0.25</td>
<td></td>
<td>0.32</td>
</tr>
<tr>
<td>blizzard</td>
<td>0.15</td>
<td>0.05</td>
<td>0.64</td>
<td>0.17</td>
<td>0.99</td>
<td></td>
<td>0.23</td>
</tr>
</tbody>
</table>
“One hot” vectors and dense word vectors (embeddings)
Low-dimensional word representations

- Learning representations by back-propagating errors
  - Rumelhart, Hinton & Williams, 1986
- A neural probabilistic language model
  - Bengio et al., 2003
- Natural Language Processing (almost) from scratch
  - Collobert & Weston, 2008
- Word representations: A simple and general method for semi-supervised learning
  - Turian et al., 2010
- Distributed Representations of Words and Phrases and their Compositionality
  - Word2Vec; Mikolov et al., 2013
Word2Vec

- Popular embedding method
- Very fast to train
- Code available on the web
- Idea: predict rather than count
Word2Vec

- Skip-gram
- CBOW

[Mikolov et al. ’13]
Skip-gram Prediction

- Predict vs Count

the cat sat on the mat
Skip-gram Prediction

- Predict vs Count

\[
w_t = \text{the} \quad \xrightarrow{\text{CLASSIFIER}} \quad \text{the cat sat on the mat}
\]

\[
w_{t-2} = \langle \text{start}_2 \rangle
\]
\[
w_{t-1} = \langle \text{start}_1 \rangle
\]
\[
w_{t+1} = \text{cat}
\]
\[
w_{t+2} = \text{sat}
\]

context size = 2
Skip-gram Prediction

- **Predict vs Count**

  \[ w_t = \text{cat} \]

  \[ w_{t-2} = \text{<start}_1\text{>} \]
  \[ w_{t-1} = \text{the} \]
  \[ w_{t+1} = \text{sat} \]
  \[ w_{t+2} = \text{on} \]

  context size = 2
Skip-gram Prediction

- Predict vs Count

\[ w_t = \text{sat} \]

```
  the cat sat on the mat
```

\[ w_{t-2} = \text{the} \]
\[ w_{t-1} = \text{cat} \]
\[ w_{t+1} = \text{on} \]
\[ w_{t+2} = \text{the} \]

context size = 2

Skip-gram
Skip-gram Prediction

- Predict vs Count

\[ w_t = \text{on} \rightarrow \text{CLASSIFIER} \rightarrow \text{the cat sat } \text{on the mat} \]

\[ w_{t-2} = \text{cat} \]
\[ w_{t-1} = \text{sat} \]
\[ w_{t+1} = \text{the} \]
\[ w_{t+2} = \text{mat} \]

context size = 2
Skip-gram Prediction

- Predict vs Count

\[ w_t = \text{the} \]

\[ \text{the cat sat on the mat} \]

\[ w_{t-2} = \text{sat} \]
\[ w_{t-1} = \text{on} \]
\[ w_{t+1} = \text{mat} \]
\[ w_{t+2} = \langle \text{end} + 1 \rangle \]

context size = 2
Skip-gram Prediction

- Predict vs Count

the cat sat on the mat

\[ w_t = \text{mat} \rightarrow \text{CLASSIFIER} \rightarrow \]

\[ w_{t-2} = \text{on} \]
\[ w_{t-1} = \text{the} \]
\[ w_{t+1} = \text{<end+1>} \]
\[ w_{t+2} = \text{<end+2>} \]

context size = 2
Skip-gram Prediction

- **Predict vs Count**

\[ w_t = \text{the} \quad \rightarrow \quad \text{CLASSIFIER} \quad \rightarrow \quad w_{t-2} = \text{sat} \]
\[ w_{t-1} = \text{on} \]
\[ w_{t+1} = \text{mat} \]
\[ w_{t+2} = \langle \text{end} \rangle +1 \]

\[ w_t = \text{the} \quad \rightarrow \quad \text{CLASSIFIER} \quad \rightarrow \quad w_{t-2} = \langle \text{start} \rangle .2 \]
\[ w_{t-1} = \langle \text{start} \rangle .1 \]
\[ w_{t+1} = \text{cat} \]
\[ w_{t+2} = \text{sat} \]
Skip-gram Prediction

\[
W_{\text{in}} \times W_{\text{the}} \times W_{\text{out}} \rightarrow \text{softmax} \rightarrow p(w_{t-2}|w_t) \rightarrow \text{softmax} \rightarrow \text{softmax} \rightarrow \text{softmax} \rightarrow \text{softmax} \rightarrow \text{softmax} \rightarrow \text{truth}
\]

INPUT \rightarrow \text{PROJECTION} \rightarrow \text{OUTPUT}

\[
\begin{align*}
w(t-2) & \rightarrow w(t-1) \rightarrow w(t+1) \rightarrow w(t+2) \\
\end{align*}
\]
How to compute $p(\cdot|t,c)$?

\[ \sigma(x) = \frac{1}{1 + e^{-x}} \]
FastText

\[ \sigma(x) = \frac{1}{1 + e^{-x}} \]

Diagram showing the interaction of words and their vectors in a FastText model.
Typical traits of these embeddings

Automatically learn some analogies pretty well

Figure from Sutor et al. MIPR 2019
Takeaways
What we've learned

- The contexts in which a word typically appears (i.e., the tokens that typically appear around it) tell us a lot about that word.
- We can use those contexts to automatically learn more powerful representations of words than just a one-hot encoding.
- These “word embeddings” can plug in as parameters in models of your choice.
Next class

Neural Networks I (Feedforward networks and LSTMs)