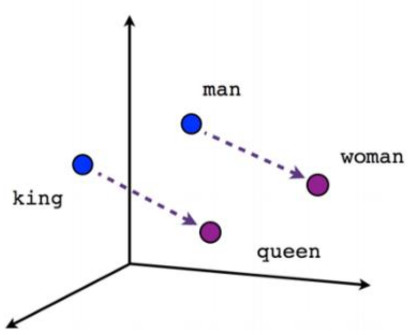
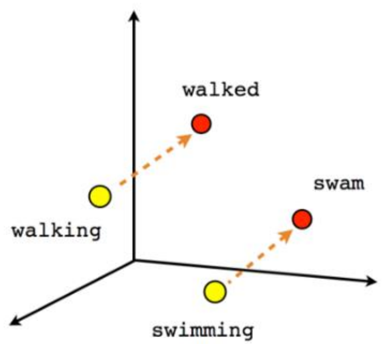
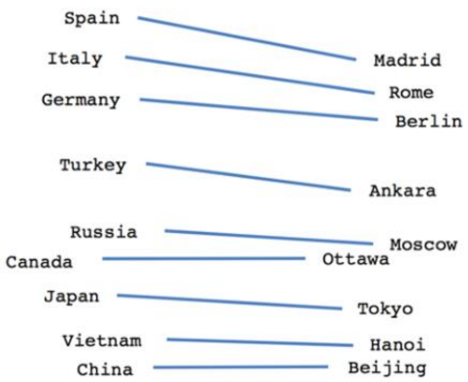
Word Embeddings



Male-Female



Verb tense



Country-Capital

Slides from Chris Manning

ECE324, Winter 2023

Michael Guerzhoy

Word representations

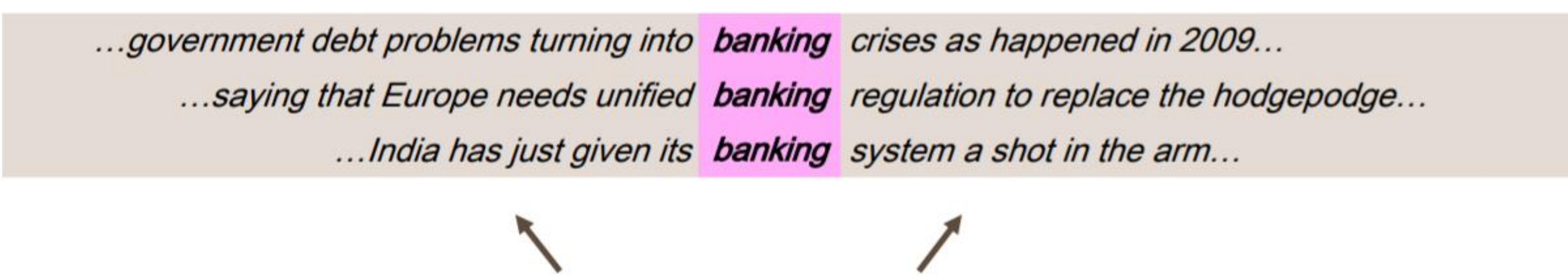
- Want to represent input words to ML models
- Want similar words to have similar representations
  - A lot of the time, want similar outputs from the model when the inputs are similar
- Want dimensionality to be small
- One-hot encodings don't represent relationships between words

motel = [0 0 0 0 0 0 0 0 0 0 1 0 0 0 0]

hotel = [0 0 0 0 0 0 0 1 0 0 0 0 0 0 0]

Representing words by their context

- Distributional hypothesis in linguistics: a word's meaning is related to what words appear close-by to the word
- When a word *w* appears in a text, its context is the set of words that appear nearby
- Use the context of *w* to build up a representation of *w*



These context words will represent ***banking***

Word vectors

- Each word is represented by an n-dimensional vector

$$\textit{banking} = \begin{pmatrix} 0.286 \\ 0.792 \\ -0.177 \\ -0.107 \\ 0.109 \\ -0.542 \\ 0.349 \\ 0.271 \end{pmatrix} \qquad \textit{monetary} = \begin{pmatrix} 0.413 \\ 0.582 \\ -0.007 \\ 0.247 \\ 0.216 \\ -0.718 \\ 0.147 \\ 0.051 \end{pmatrix}$$

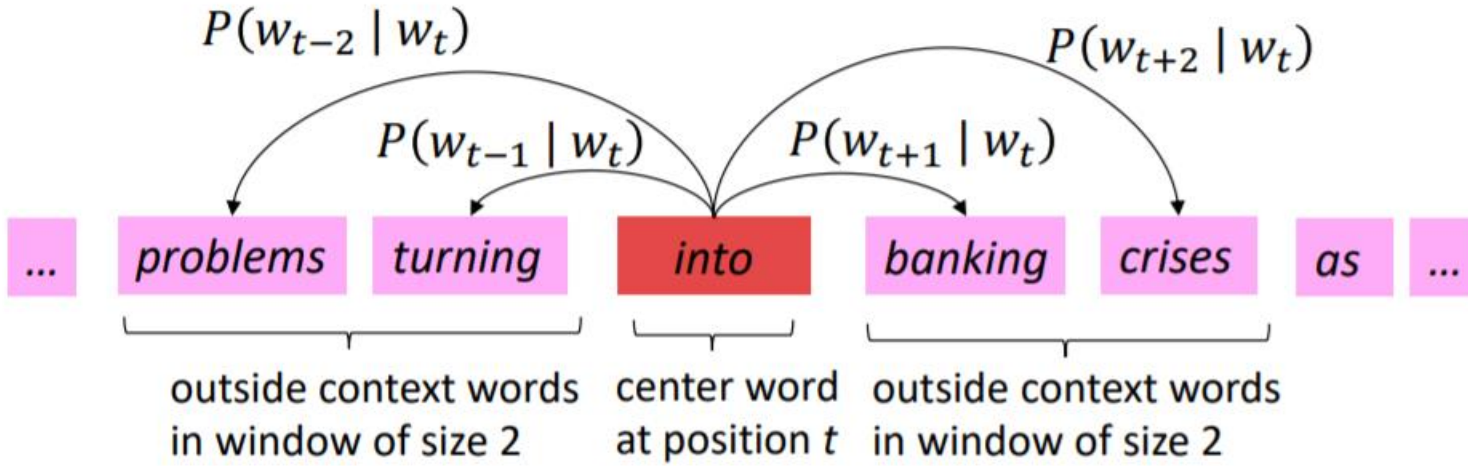
Similarity of vectors

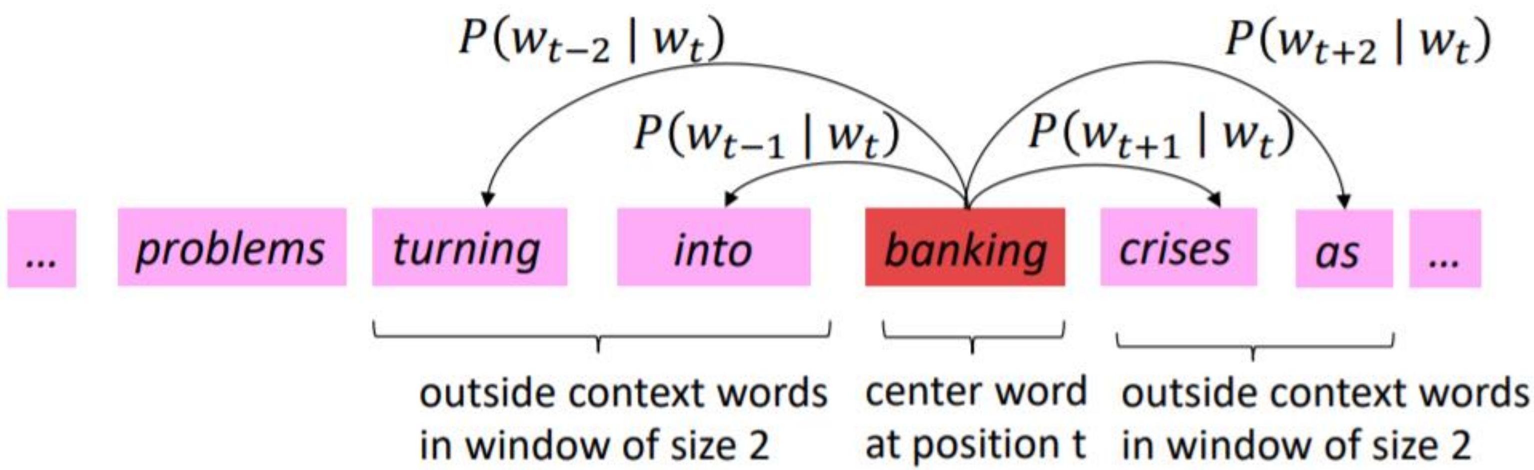
- The cosine of the angle between vectors u and v is
$$\cos\theta_{u,v} = \frac{u \cdot v}{|u||v|}$$
- Easier to use $u \cdot v$
  - Related to the angle if u and v have roughly the same magnitude

$$\textit{expect} = \begin{pmatrix} 0.286 \\ 0.792 \\ -0.177 \\ -0.107 \\ 0.109 \\ -0.542 \\ 0.349 \\ 0.271 \\ 0.487 \end{pmatrix}$$

Word2Vec

- Every word is represented by an n-dimensional vector
- Go through each position *t* in the test
  - Get pairs of center words *c* and context words *o*
  - Compute *P(o|c)* as a function of the similarity of *c* and *o*
- Maximize the probability of the occurrence of outside words given center words


$P(w_{t-2} \mid w_t)$
$P(w_{t+2} \mid w_t)$
$P(w_{t-1} \mid w_t)$
$P(w_{t+1} \mid w_t)$
...
problems
turning
into
banking
crises
as
...
outside context words in window of size 2
center word at position t
outside context words in window of size 2


$P(w_{t-2} \mid w_t)$
$P(w_{t+2} \mid w_t)$
$P(w_{t-1} \mid w_t)$
$P(w_{t+1} \mid w_t)$
...
problems
turning
into
banking
crises
as
...
outside context words in window of size 2
center word at position t
outside context words in window of size 2

P(o|c)

$$P(o|c) = \frac{\exp(u_o \cdot v_c)}{\sum_{w \in V} \exp(u_w \cdot v_c)}$$

- Larger similarity ⇔ larger probability
- Derivative expensive to compute: need to sum over the entire vocabulary

(Maximizing P(o|c) is called “Continuous Bag of Words and maximizing P(c|a) is called “Skip-gram”)

- Likelihood:

$$L(\theta) = \prod_{t=1}^{T} \prod_{\substack{-m \le j \le m \\ j \ne 0}} P(w_{t+j} | w_t; \theta)$$

- Negative log-likelihood NLL:

$$J(\theta) = -\frac{1}{T} \log L(\theta) = -\frac{1}{T} \textstyle\sum_{t=1}^{T} \sum_{\substack{-m \le j \le m \\ j \ne 0}} \log P(w_{t+j} | w_t; \theta)$$

- Maximum likelihood ⇔ Minimum NLL

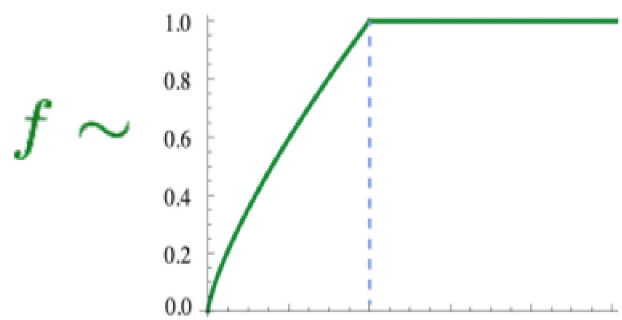
$$J(\theta) = -\frac{1}{T} \sum_{t=1}^{T} \sum_{\substack{-m \le j \le m \\ j \ne 0}} \left(u_{t+j} \cdot v_t - \log \sum_{w \in W} \exp(u_w \cdot v_t)\right)$$

Negative sampling

- $J(\theta) = -\frac{1}{T}\sum_{y=1}^{T} J_t(\theta)$
- $J_t(\theta) = \sum_{o \in [t-m, t-1] \cup [t+1, t+m]} \log \sigma(u_o \cdot v_t) + KE_{j \sim P(w)} \; [\log \sigma(-u_j \cdot v_t)]$
- $KE_{j \sim P(w)} [\log \sigma(-u_j \cdot v_c)] \approx \sum_{k \in \{K\ sampled\ indicies\}} \log \sigma(-u_k \cdot v_t)$
- Maximize the probability that real outside word appears
- Minimize the probability that random words appear near centre words

GLoVe

- X_{ij}: the number of co-occurrences of w_i and w_j
- $J = \sum_{i,j} f(X_{ij})(u_i \cdot v_j + b_i + b'_j - \log X_{ij})^2$



GLoVe results

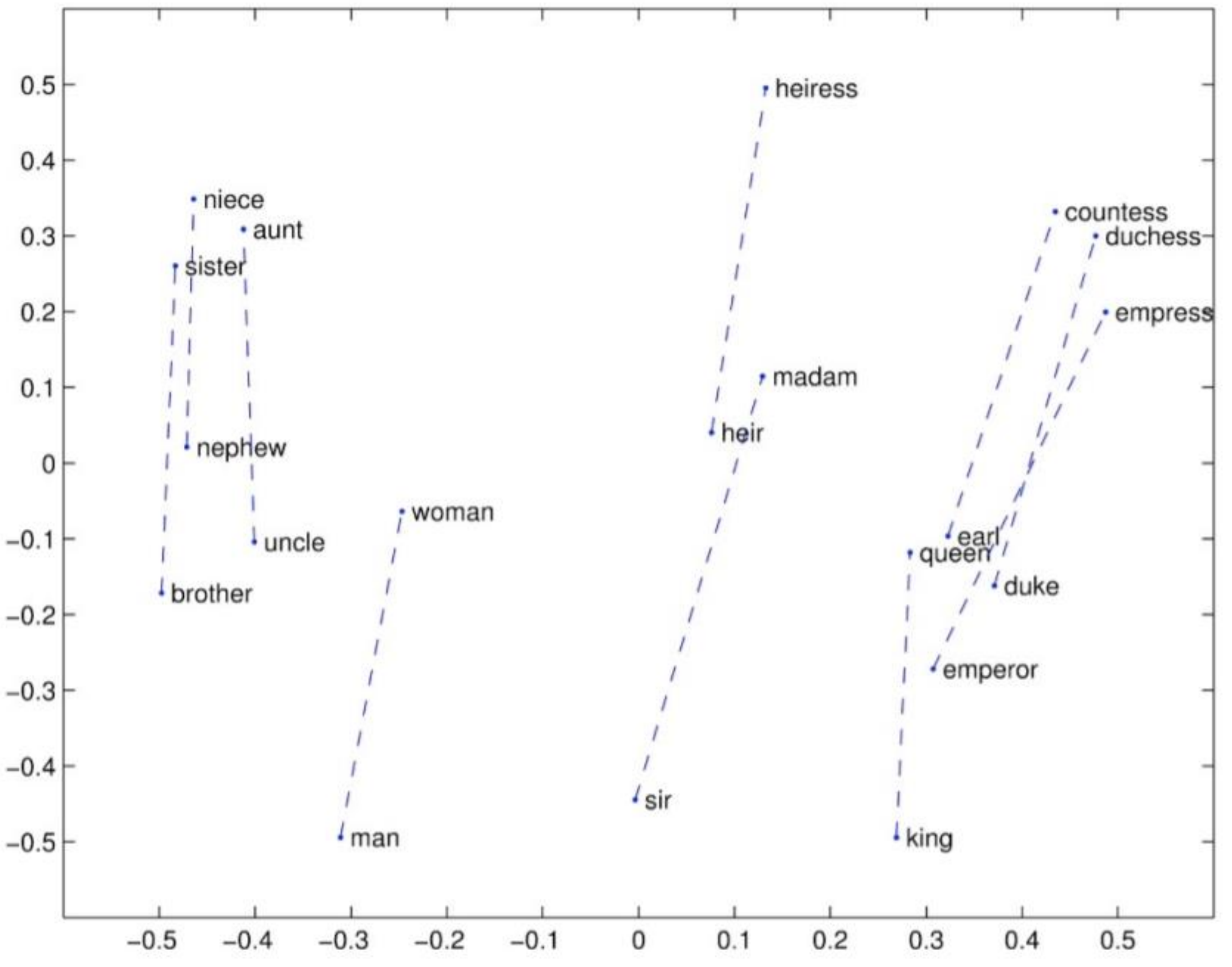
Nearest words to frog:

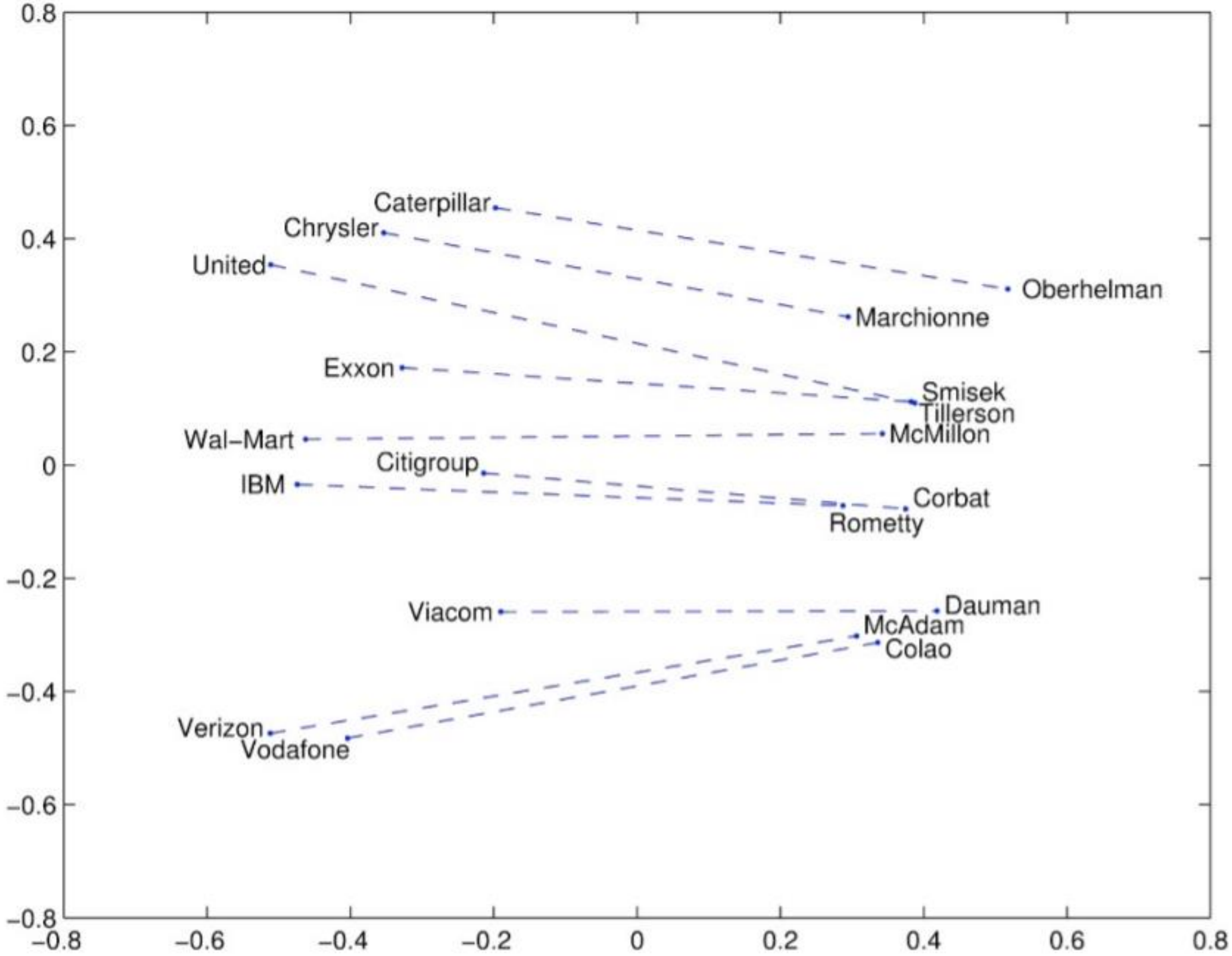
1. frogs
2. toad
3. litoria
4. leptodactylidae
5. rana
6. lizard
7. eleutherodactylus

litoria

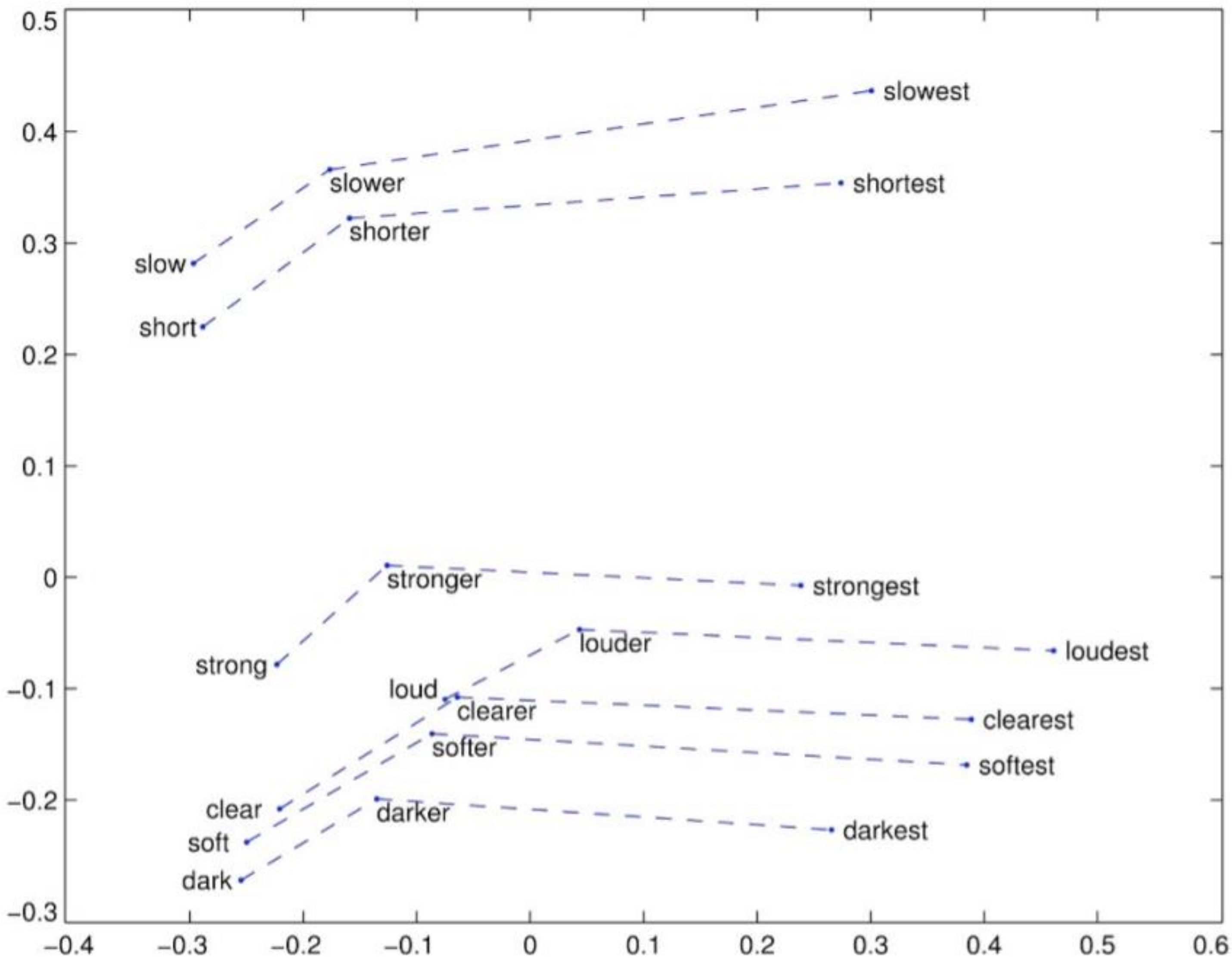
leptodactylidae

Vector analogies





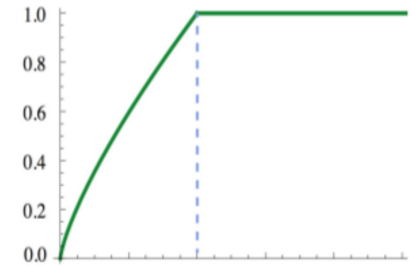
0.8
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−0.8
−0.8
−0.6
−0.4
−0.2
0
0.2
0.4
0.6
0.8
Caterpillar
Chrysler
United
Oberhelman
Marchionne
Exxon
Smisek
Tillerson
Wal−Mart
McMillon
Citigroup
IBM
Corbat
Rometty
Viacom
Dauman
McAdam
Colao
Verizon
Vodafone



0.5
0.4
0.3
0.2
0.1
0
−0.1
−0.2
−0.3
−0.4
−0.3
−0.2
−0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
slowest
slower
shortest
shorter
slow
short
stronger
strongest
strong
louder
loudest
loud
clearer
clearest
softer
softest
clear
soft
dark
darker
darkest

GLoVe cost function

$f \sim$



- X_{ij}: the number of co-occurrences of w_i and w_j
- X_i: the number of occurrences of w_i
- Want $\log \mathrm{P}\left(word_i | word_j\right) \approx \log \frac{X_{ij}}{X_i} \approx u_i \cdot v_j$
- $u_i \cdot v_j \approx \log X_{ij} - \log X_i$
  - Absorb $\log X_i$ into the biases, make expression symmetric
  - Learn with least squares, don't upweight cases with large X_{ij} too much
- $J = \sum_{i,j} f(X_{ij})(u_i \cdot v_j + b_i + b'_j - \log X_{ij})^2$

GLoVe cost function

- Idea: ratios of co-occurrence probabilities can encode meaning components

	x = solid	x = gas	x = water	x = random
$P(x\|\text{ice})$	large	small	large	small
$P(x\|\text{steam})$	small	large	large	small
$\frac{P(x\|\text{ice})}{P(x\|\text{steam})}$	large	small	~1	~1

	x = solid	x = gas	x = water	x = fashion
$P(x\|\text{ice})$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(x\|\text{steam})$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$\frac{P(x\|\text{ice})}{P(x\|\text{steam})}$	8.9	8.5×10^{-2}	1.36	0.96

GLoVe intuition via probability ratios

- Want the word vectors to encode information about ratios of probabilities

$$F\left(\left(u_i - u_j\right) \cdot v_k\right) \approx \frac{P(i|k)}{P(j|k)} \approx \frac{X_{ik}/X_k}{X_{jk}/X_k}$$

- Set $F = exp$
- $\frac{\exp(u_i \cdot v_k)}{\exp(u_j \cdot v_k)} \approx \frac{X_{ik}/X_k}{X_{jk}/X_k}$
- $u_i \cdot v_k \approx \log X_{ik} - \log X_k$
- Absorb $\log X_k$, make expression symmetric to get

$$u_i \cdot v_k + b_i + b'_k \approx \log(X_{ik})$$