

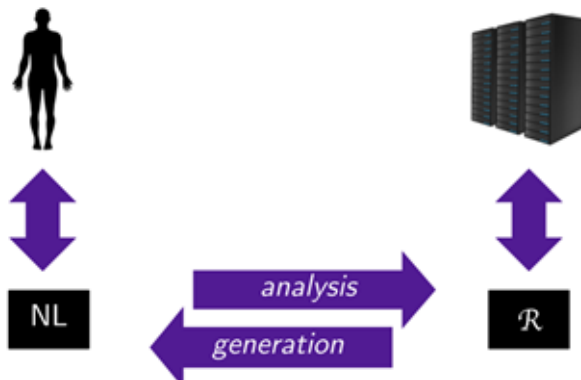
# Lexical Semantics

Luke Zettlemoyer

[lsz@cs.washington.edu](mailto:lsz@cs.washington.edu)

# What is Natural Language Processing (NLP)?

- $NL \in \{\text{Mandarin Chinese, Hindi, Spanish, Arabic, English, ... Inuktitut, Njerep}\}$
- Automation of NLs:
  - analysis of (“understanding”) what a text means, to some extent (  $NL \rightarrow \mathcal{R}$  )
  - generation of fluent, meaningful, context-appropriate text (  $\mathcal{R} \rightarrow NL$  )
  - acquisition of  $\mathcal{R}$  from knowledge and data



# Lexical semantics: what do words mean?

- N-gram or text classification methods we've seen so far
  - Words are just strings (or indices  $w_i$  in a vocabulary list)
  - That's not very satisfactory!

How do we represent  
the meaning of a word?

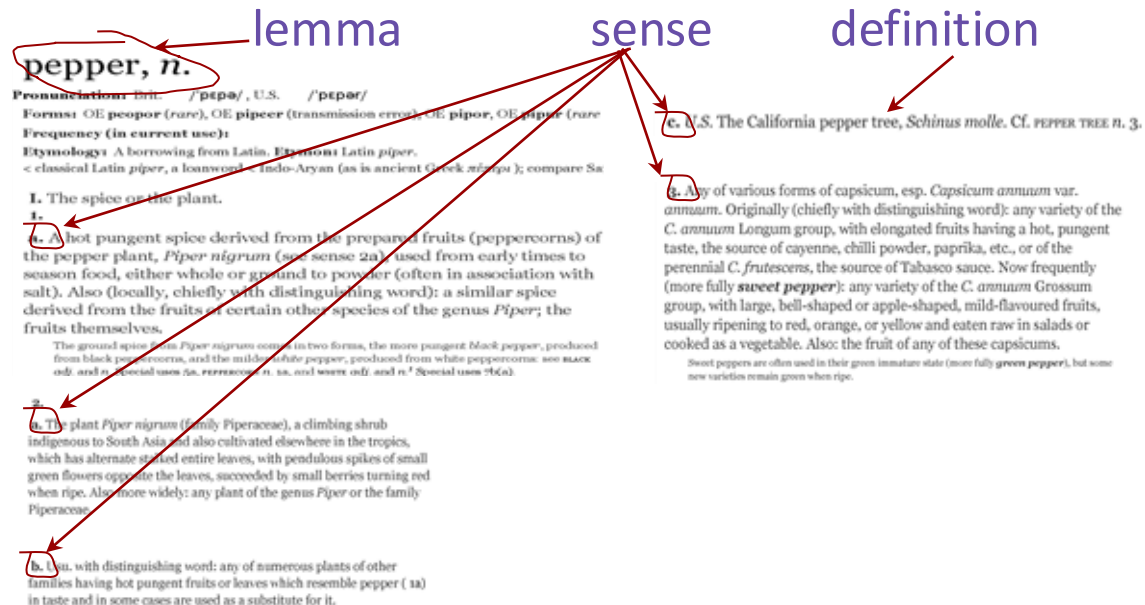
# Desiderata

What should a theory of word meaning do for us?

Let's look at some desiderata from [lexical semantics](#), the linguistic study of word meaning

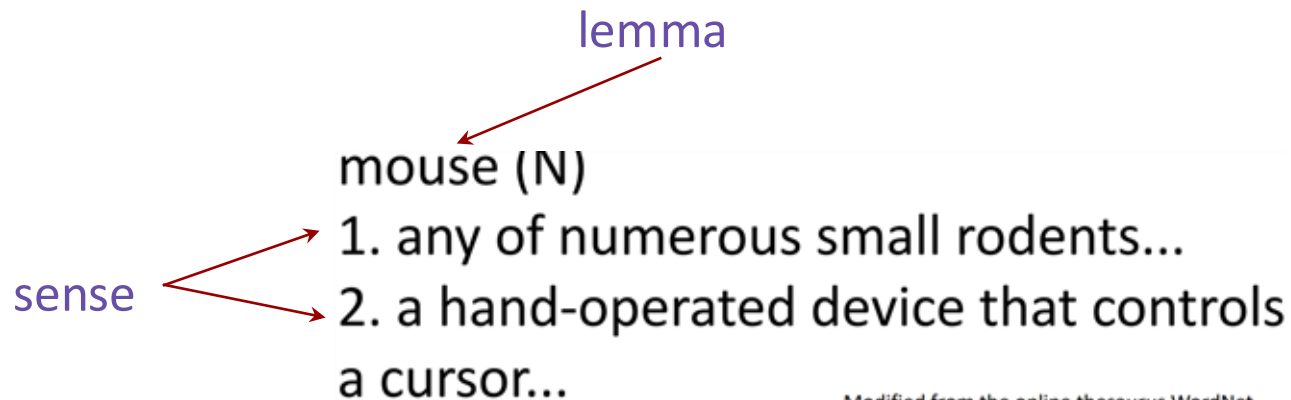
# Lexical semantics

- How should we represent the meaning of the word?
  - Words, lemmas, senses, definitions



<http://www.oed.com/>

# Lemmas and senses



Modified from the online thesaurus WordNet

A **sense** or “**concept**” is the meaning component of a word Lemmas can be **polysemous** (have multiple senses)

# Relation: synonymy

- Synonyms have the same meaning in some or all contexts.
  - filbert / hazelnut
  - couch / sofa
  - big / large
  - automobile / car
  - vomit / throw up
  - Water / H<sub>2</sub>O

# The Linguistic Principle of Contrast

Difference in form → difference in meaning

- Note that there are probably **no examples of perfect synonymy**
  - Even if many aspects of meaning are identical
  - Still may not preserve the acceptability based on notions of politeness, slang, register, genre, etc.
    - Water / H<sub>2</sub>O in a surfing guide?
    - my big sister != my large sister

# Relation: antonymy

Senses that are opposites with respect to one feature of meaning

- Otherwise, they are very similar!
  - dark/light    short/long    fast/slow    rise/fall
  - hot/cold    up/down    in/out

More formally: antonyms can

- define a binary opposition or be at opposite ends of a scale
  - long/short, fast/slow
- be reversives:
  - rise/fall, up/down

# Relation: similarity

Words with similar meanings.

- Not synonyms, but sharing some element of meaning
  - car, bicycle
  - cow, horse

# Ask humans how similar two words are

word1	word2	similarity
vanish	disappear	<input type="checkbox"/>
behave	obey	<input type="checkbox"/>
belief	impression	<input type="checkbox"/>
muscle	bone	<input type="checkbox"/>
modest	flexible	<input type="checkbox"/>
hole	agreement	<input type="checkbox"/>

SimLex-999 dataset (Hill et al., 2015)

# Ask humans how similar two words are

word1	word2	similarity
vanish	disappear	9.8
behave	obey	7.3
belief	impression	5.95
muscle	bone	3.65
modest	flexible	0.98
hole	agreement	0.3

SimLex-999 dataset (Hill et al., 2015)

# Relation: word relatedness

Also called "word association"

- Words be related in any way, perhaps via a semantic frame or field
  - car, bicycle: similar
  - car, gasoline: related, not similar

# Semantic field

## Words that

- cover a particular semantic domain
- bear structured relations with each other

## hospitals

surgeon, scalpel, nurse, anaesthetic, hospital

## restaurants

waiter, menu, plate, food, menu, chef

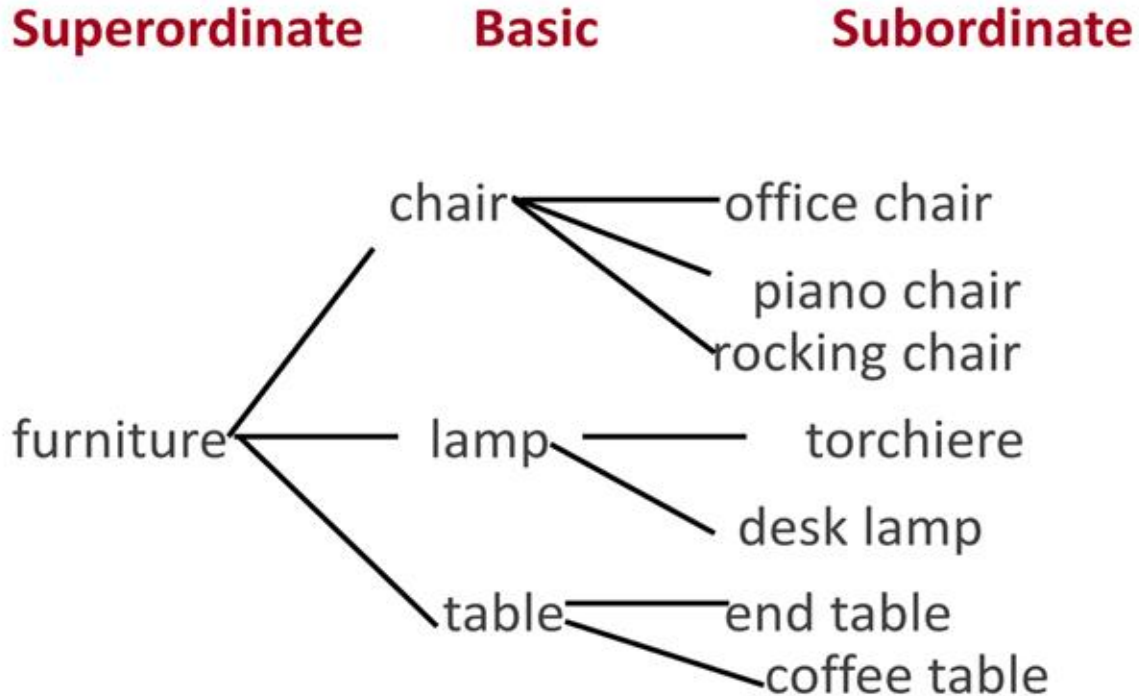
## houses

door, roof, kitchen, family, bed

# Taxonomic relation: superordinate/ subordinate

- One sense is a subordinate (**hyponym**) of another if the first sense is more specific, denoting a subclass of the other
  - car is a subordinate of vehicle
  - mango is a subordinate of fruit
- Conversely superordinate (**hypernym**)
  - vehicle is a superordinate of car
  - fruit is a superordinate of mango

# Taxonomy



# Lexical semantics

- How should we represent the meaning of the word?
  - Dictionary definition
  - Lemma and wordforms
  - Senses
  - Relationships between words or senses
  - Taxonomic relationships
  - Word similarity, word relatedness
  - Semantic frames and roles
  - Connotation and sentiment

# Lexical semantics

- How should we represent the meaning of the word?
  - Dictionary definition
  - Lemma and wordforms
  - Senses
  - Relationships between words or senses
  - Taxonomic relationships
  - Word similarity, word relatedness
  - Semantic frames and roles
    - *John hit Bill*
    - *Bill was hit by John*

# Lexical Semantics

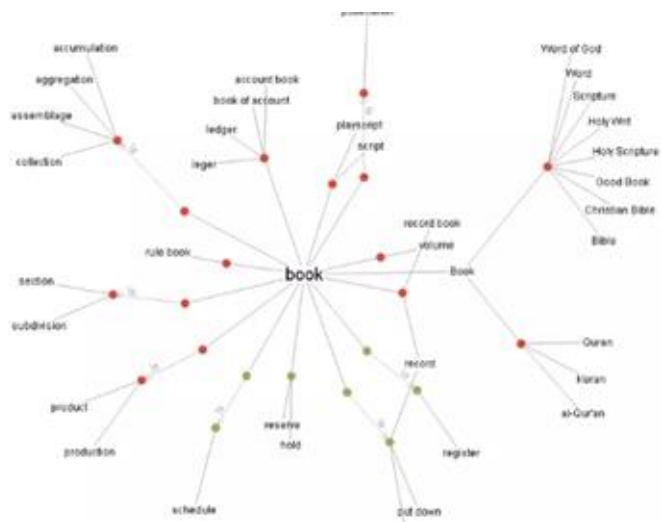
- How should we represent the meaning of the word?
  - Dictionary definition
  - Lemma and wordforms
  - Senses
  - Relationships between words or senses
  - Taxonomic relationships
  - Word similarity, word relatedness
  - Semantic frames and roles
  - Connotation and sentiment
    - *valence*: the pleasantness of the stimulus
    - *arousal*: the intensity of emotion
    - *dominance*: the degree of control exerted by the stimulus

	Valence	Arousal	Dominance
courageous	8.05	5.5	7.38
music	7.67	5.57	6.5
heartbreak	2.45	5.65	3.58
cub	6.71	3.95	4.24
life	6.68	5.59	5.89

# Electronic Dictionaries

## WordNet

<https://wordnet.princeton.edu/>



### WordNet Search - 3.1

- [WordNet home page](#) - [Glossary](#) - [Help](#)

Word to search for:

Display Options:

Key: "S." = Show Synset (semantic) relations, "W." = Show Word (lexical) relations

Display options for sense:

#### Noun

- **S. (n) bank** (sloping land (especially the slope beside a body of water)) *"they pulled the canoe up on the bank"; "he sat on the bank of the river and watched the currents"*
- **S. (n) depository financial institution, bank, banking concern, banking company** (a financial institution that accepts deposits and channels the money into lending activities) *"he cashed a check at the bank"; "that bank holds the mortgage on my home"*
- **S. (n) bank** (a long ridge or pile) *"a huge bank of earth"*
- **S. (n) bank** (an arrangement of similar objects in a row or in tiers) *"he operated a bank of switches"*
- **S. (n) bank** (a supply or stock held in reserve for future use (especially in emergencies))
- **S. (n) bank** (the funds held by a gambling house or the dealer in some gambling games) *"he tried to break the bank at Monte Carlo"*
- **S. (n) bank, cant, camber** (a slope in the turn of a road or track; the outside is higher than the inside in order to reduce the effects of centrifugal force)

# Electronic Dictionaries

## WordNet

```
from nltk.corpus import wordnet as wn
panda = wn.synset('panda.n.01')
hyper = lambda s: s.hypernyms()
list(panda.closure(hyper))
```

```
[Synset('procyonid.n.01'),
Synset('carnivore.n.01'),
Synset('placental.n.01'),
Synset('mammal.n.01'),
Synset('vertebrate.n.01'),
Synset('chordate.n.01'),
Synset('animal.n.01'),
Synset('organism.n.01'),
Synset('living_thing.n.01'),
Synset('whole.n.02'),
Synset('object.n.01'),
Synset('physical_entity.n.01'),
Synset('entity.n.01')]
```

# Problems with discrete representations

- Too coarse
  - *expert* ↔ *skillful*
- Sparse
  - *wicked, badass, ninja*
- Subjective
- Expensive
- Hard to compute word relationships

S: (adj) full, good  
 S: (adj) estimable, good, honorable, respectable  
 S: (adj) beneficial, good  
 S: (adj) good, just, upright  
 S: (adj) adept, expert, good, practiced, proficient, skillful  
 S: (adj) dear, good, near  
 S: (adj) good, right, ripe  
 ...  
 S: (adv) well, good  
 S: (adv) thoroughly, soundly, good  
 S: (n) good, goodness  
 S: (n) commodity, trade good, good

*expert* [0 0 0 **1** 0 0 0 0 0 0 0 0 0 0 0]

*skillful* [0 0 0 0 0 0 0 0 0 0 0 **1** 0 0 0]

# Distributional hypothesis

“The meaning of a word is its use in the language”

[Wittgenstein PI 43]

“You shall know a word by the company it keeps”

[Firth 1957]

If A and B have almost identical environments we say that they are synonyms.

[Harris 1954]

# Example

What does ongchoi mean?

# Example

- Suppose you see these sentences:
  - Ongchoi is delicious **sautéed with garlic**.
  - Ongchoi is superb **over rice**
  - Ongchoi **leaves** with salty sauces
- And you've also seen these:
  - ...spinach **sautéed with garlic over rice**
  - Chard stems and **leaves** are delicious
  - Collard greens and other **salty** leafy greens

# Ongchoi: *Ipomoea aquatica* "Water Spinach"

Ongchoi is a leafy green like spinach, chard, or collard greens

空心菜  
*kangkong*  
rau muống  
...



Yamaguchi, Wikimedia Commons, public domain

# Model of meaning focusing on similarity

- Each word = a vector
  - not just “word” or word45.
  - similar words are “nearby in space”
  - We build this space automatically by seeing which words are nearby in text



# We define meaning of a word as a vector

- Called an "embedding" because it's embedded into a space
- The standard way to represent meaning in NLP

Every modern NLP algorithm uses embeddings as the representation of word meaning

# Intuition: why vectors?

Consider sentiment analysis:

- With **words**, a feature is a word identity
  - Feature 5: 'The previous word was "terrible"'
  - requires **exact same word** to be in training and test
  
- With embeddings:
  - Feature is a word vector
  - 'The previous word was vector [35,22,17...]
  - Now in the test set we might see a similar vector [34,21,14]
  - We can generalize to **similar but unseen** words!!!

# How to represent the meaning of a word?

What property we want the mapping to have?

We want vectors of similar words to be close. And dissimilar words to be away from each other.

`distance(f(apple), f(orange)) <- small`

`distance(f(computer), f(rabbit)) <- large`

# There are many kinds of embeddings

- Count-based
  - Words are represented by a simple function of the counts of nearby words
- Class-based
  - Representation is created through hierarchical clustering, Brown clusters
- Distributed prediction-based (type) embeddings
  - Representation is created by training a classifier to distinguish nearby and far-away words: word2vec, fasttext
- Distributed contextual (token) embeddings from language models
  - ELMo, BERT

# We'll discuss 2 kinds of embeddings

- **tf-idf**

- Information Retrieval workhorse!
- A common baseline model
- **Sparse** vectors
- Words are represented by (a simple function of) the counts of nearby words

- **Word2vec**

- **Dense** vectors
- Representation is created by training a classifier to predict whether a word is likely to appear nearby
- <https://fasttext.cc/docs/en/crawl-vectors.html>
- Later we'll discuss extensions called **contextual embeddings**

# Vector Semantics

# Term-document matrix

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	0	7	17
soldier	2	80	62	89
fool	36	58	1	4
clown	20	15	2	3

Context = appearing in the same document.

# Term-document Matrix

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	0	7	17
soldier	2	80	62	89
fool	36	58	1	4
clown	20	15	2	3

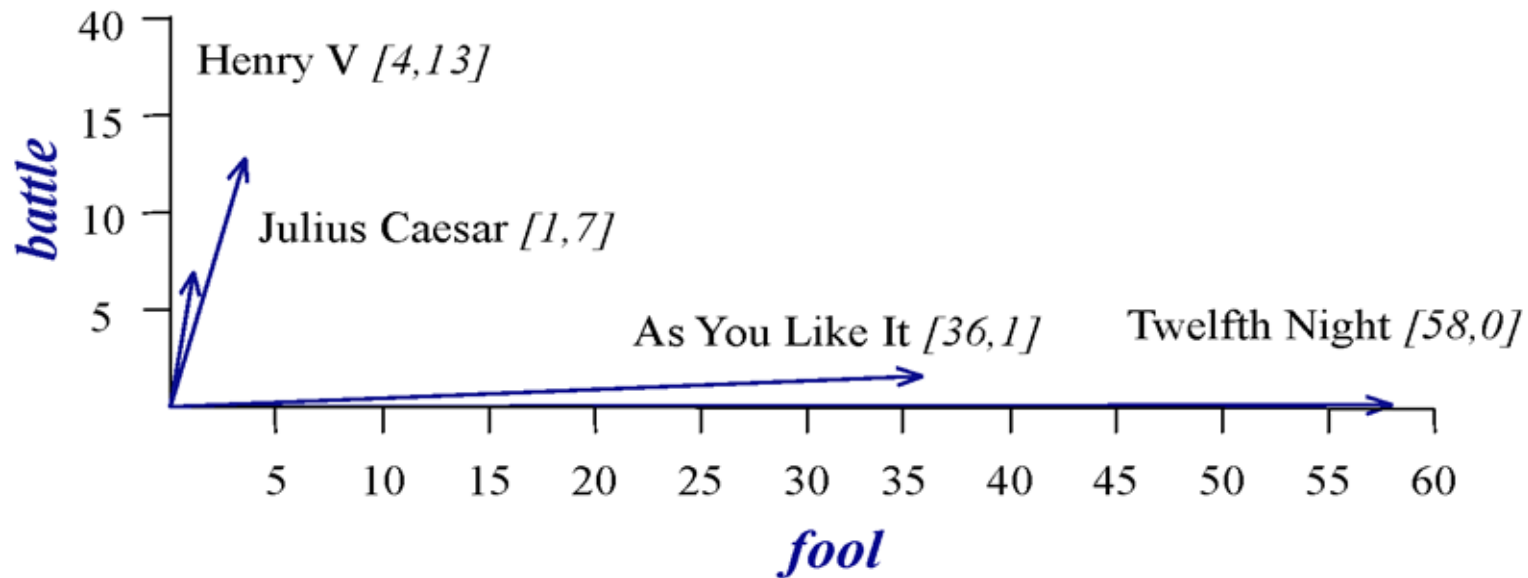
Each document is represented by a vector of words

# Vectors are the basis of information retrieval

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	0	7	17
soldier	2	80	62	89
fool	36	58	1	4
clown	20	15	2	3

- Vectors are similar for the two comedies
- Different than the history
- Comedies have more fools and wit and fewer battles.

# Visualizing Document Vectors



# Words can be vectors too

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	0	7	13
good	114	80	62	89
fool	36	58	1	4
clown	20	15	2	3

- battle is "the kind of word that occurs in Julius Caesar and Henry V"
- fool is "the kind of word that occurs in comedies, especially Twelfth Night"

# More common: word-word matrix (“term-context matrix”)

	knife	dog	sword	love	like
knife	0	1	6	5	5
dog	1	0	5	5	5
sword	6	5	0	5	5
love	5	5	5	0	5
like	5	5	5	5	2

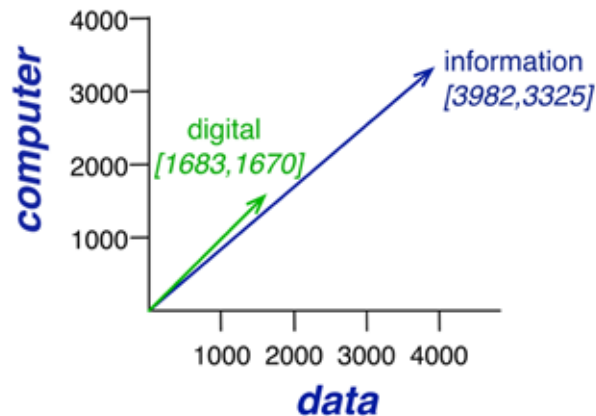
- Two words are “similar” in meaning if their context vectors are similar
  - Similarity == relatedness

# Term-context matrix

Two **words** are similar in meaning if their context vectors are similar

is traditionally followed by **cherry** pie, a traditional dessert  
 often mixed, such as **strawberry** rhubarb pie. Apple pie  
 computer peripherals and personal **digital** assistants. These devices usually  
 a computer. This includes **information** available on the internet

	aardvark	...	computer	data	result	pie	sugar	...
<b>cherry</b>	0	...	2	8	9	442	25	...
<b>strawberry</b>	0	...	0	0	1	60	19	...
<b>digital</b>	0	...	1670	1683	85	5	4	...
<b>information</b>	0	...	3325	3982	378	5	13	...



# Computing word similarity

The dot product between two vectors is a scalar:

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

- 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 dimension)

Vector length:

$$|\mathbf{v}| = \sqrt{\sum_{i=1}^N v_i^2}$$

- 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

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

Based on the definition of the dot product between two vectors  $\mathbf{a}$  and  $\mathbf{b}$

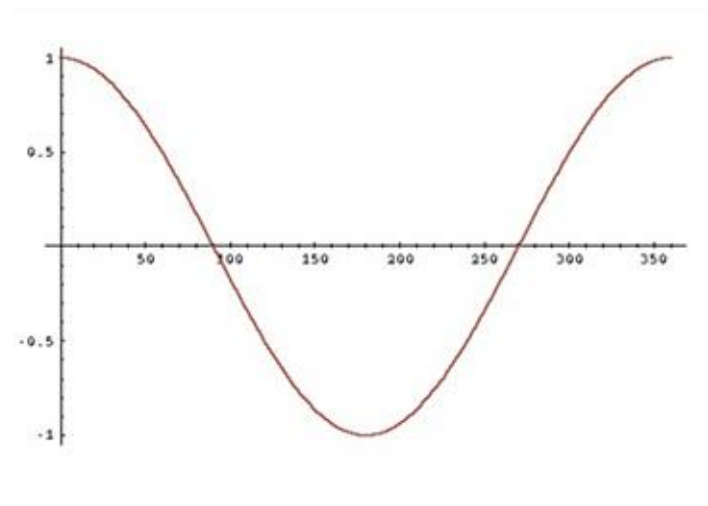
$$\mathbf{a} \cdot \mathbf{b} = |\mathbf{a}| |\mathbf{b}| \cos \theta$$
$$\frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{a}| |\mathbf{b}|} = \cos \theta$$

# 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

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

	pie	data	computer
cherry	442	8	2
digital	5	1683	1670
information	5	3982	3325

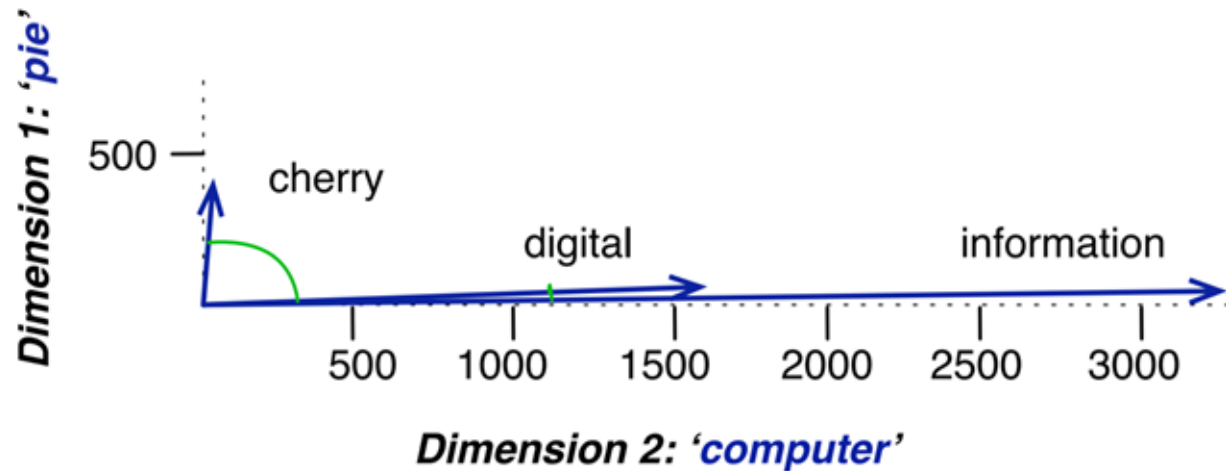
$$\cos(\text{cherry}, \text{information}) =$$

$$\frac{442 * 5 + 8 * 3982 + 2 * 3325}{\sqrt{442^2 + 8^2 + 2^2} \sqrt{5^2 + 3982^2 + 3325^2}} = .017$$

$$\cos(\text{digital}, \text{information}) =$$

$$\frac{5 * 5 + 1683 * 3982 + 1670 * 3325}{\sqrt{5^2 + 1683^2 + 1670^2} \sqrt{5^2 + 3982^2 + 3325^2}} = .996$$

# Visualizing angles



# Count-based representations

- Counts: term-frequency
  - remove stop words
  - use  $\log_{10}(\text{tf})$
  - normalize by document length

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	0	7	13
good	114	80	62	89
fool	36	58	1	4
wit	20	15	2	3

# But raw frequency is a bad representation

- The co-occurrence matrices we have seen represent each cell by word frequencies
- Frequency is clearly useful; if **sugar** appears a lot near **apricot**, that's useful information
- But overly frequent words like **the**, **it**, or **they** are not very informative about the context
- It's a paradox! How can we balance these two conflicting constraints?

# TF-IDF

- What to do with words that are evenly distributed across many documents?

$$\text{tf}_{t,d} = \log_{10}(\text{count}(t,d) + 1)$$

$$\text{idf}_i = \log \left( \frac{N}{\text{df}_i} \right)$$

Total # of docs in collection

# of docs that have word  $i$

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

$$w_{t,d} = \text{tf}_{t,d} \times \text{idf}_t$$

# Positive Pointwise Mutual Information (PPMI)

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

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

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

$$\text{PPMI}_\alpha(w, c) = \max\left(\log_2 \frac{P(w, c)}{P(w)P_\alpha(c)}, 0\right) \qquad P_\alpha(c) = \frac{\text{count}(c)^\alpha}{\sum_c \text{count}(c)^\alpha}$$

# name	formula	reference
1. Joint probability	$p(xy)$	(Giuliano, 1964)
2. Conditional probability	$p(y x)$	(Gregory et al., 1999)
3. Reverse cond. probability	$p(x y)$	(Gregory et al., 1999)
4. Pointwise mutual inf. (MI)	$\log \frac{p(xy)}{p(x) \cdot p(y)}$	(Church and Hanks, 1990)
5. Mutual dependency (MD)	$\log \frac{p(xy)^2}{p(x) \cdot p(y)}$	(Thanopoulos et al., 2002)
6. Log frequency biased MD	$\log \frac{p(xy)^2}{p(x) \cdot p(y)} + \log p(xy)$	(Thanopoulos et al., 2002)
7. Normalized expectation	$\frac{2f(xy)}{f(x) + f(y)}$	(Smadja and McKeown, 1990)
8. Mutual expectation	$\frac{2f(xy)}{f(x) + f(y)} \cdot p(xy)$	(Dias et al., 2000)
9. Saliency	$\log \frac{p(xy)^2}{p(x) \cdot p(y)} \cdot \log f(xy)$	(Kilgarriff and Tugwell, 2001)
10. Pearson's $\chi^2$ test	$\sum_{i,j} \frac{(f_{ij} - \bar{f}_{ij})^2}{\bar{f}_{ij}}$	(Manning and Schütze, 1999)
11. Fisher's exact test	$\frac{f(x) \cdot f(y) \cdot f(x+y) \cdot f(x+y)}{N! f(xy) f(x) f(y) f(x+y)}$	(Pedersen, 1996)
12. t test	$\frac{f(xy) - f(x)y}{\sqrt{f(xy)(1 - (f(xy)/N))}}$	(Church and Hanks, 1990)
13. z score	$\frac{f(xy) - \bar{f}(xy)}{\sqrt{f(xy)(1 - (f(xy)/N))}}$	(Berry-Rogghe, 1973)
14. Poisson significance	$\frac{f(xy) - f(x)y \log f(xy) + \log f(xy)!}{\log N}$	(Quasthoff and Wolff, 2002)
15. Log likelihood ratio	$-2 \sum_{i,j} f_{ij} \log \frac{f_{ij}}{\bar{f}_{ij}}$	(Dunning, 1993)
16. Squared log likelihood ratio	$-2 \sum_{i,j} \frac{\log \bar{f}_{ij}^2}{f_{ij}}$	(Inkpen and Hirst, 2002)
17. Russel-Rao	$\frac{a}{a+b+c+d}$	(Russel and Rao, 1940)
18. Sokal-Michener	$\frac{ad+bc}{a+b+c+d}$	(Sokal and Michener, 1958)
19. Rogers-Tanimoto	$\frac{a+d}{a+b+c+d}$	(Rogers and Tanimoto, 1960)
20. Hamann	$\frac{(a+d) - (b+c)}{a+b+c+d}$	(Hamann, 1961)
21. Third Sokal-Sneath	$\frac{b+c}{a+d}$	(Sokal and Sneath, 1963)
22. Jaccard	$\frac{a}{a+b+c}$	(Jaccard, 1912)
23. First Kulczynski	$\frac{a}{a+b}$	(Kulczynski, 1927)
24. Second Sokal-Sneath	$\frac{a}{a+2(b+c)}$	(Sokal and Sneath, 1963)
25. Second Kulczynski	$\frac{1}{2} (\frac{a}{a+b} + \frac{a}{a+c})$	(Kulczynski, 1927)
26. Fourth Sokal-Sneath	$\frac{1}{4} (\frac{a}{a+b} + \frac{a}{a+c} + \frac{d}{d+b} + \frac{d}{d+c})$	(Kulczynski, 1927)
27. Odds ratio	$\frac{ad}{bc}$	(Tan et al., 2002)
28. Yulle's $\omega$	$\frac{\sqrt{ad} - \sqrt{bc}}{\sqrt{ad} + \sqrt{bc}}$	(Tan et al., 2002)
29. Yulle's Q	$\frac{ad-bc}{ad+bc}$	(Tan et al., 2002)
30. Driver-Kroeber	$\frac{a}{\sqrt{(a+b)(a+c)}}$	(Driver and Kroeber, 1932)

# name	formula	reference
31. Fifth Sokal-Sneath	$\frac{ad}{\sqrt{(a+b)(a+c)(d+b)(d+c)}}$	(Sokal and Sneath, 1963)
32. Pearson	$\frac{ad-bc}{\sqrt{(a+b)(a+c)(d+b)(d+c)}}$	(Pearson, 1950)
33. Baroni-Urbani	$\frac{a + \sqrt{ad}}{a+b+c + \sqrt{ad}}$	(Baroni-Urbani and Buser, 1976)
34. Braun-Blanquet	$\frac{a}{\max(a+b, a+c)}$	(Braun-Blanquet, 1932)
35. Simpson	$\frac{a}{\max(a+b, a+c)}$	(Simpson, 1943)
36. Michael	$\frac{d(ad-bc)}{(a+d)^2 + (b+c)^2}$	(Michael, 1920)
37. Mountford	$\frac{2a}{2bc+ab+ac}$	(Kaufman and Rousseeuw, 1990)
38. Fager	$\frac{a}{\sqrt{(a+b)(a+c)}} - \frac{1}{2} \max(b, c)$	(Kaufman and Rousseeuw, 1990)
39. Unigram subtuples	$\log \frac{ad}{bc} - 3.29 \sqrt{\frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d}}$	(Blaheta and Johnson, 2001)
40. U cost	$\log(1 + \frac{\max(b,c)+a}{\max(b,c)+a})$	(Tulloss, 1997)
41. S cost	$\log(1 + \frac{\max(b,c)}{a+1}) - \frac{1}{2}$	(Tulloss, 1997)
42. R cost	$\log(1 + \frac{a}{a+b}) \cdot \log(1 + \frac{a}{a+c})$	(Tulloss, 1997)
43. T combined cost	$\sqrt{U \times S \times R}$	(Tulloss, 1997)
44. Phi	$\frac{p(xy) - p(x) \cdot p(y)}{\sqrt{p(x) \cdot p(y) (1 - p(x)) (1 - p(y))}}$	(Tan et al., 2002)
45. Kappa	$\frac{p(xy) + p(x)y - p(x) \cdot p(y) - p(x) \cdot p(y)}{1 - p(x) \cdot p(y) - p(x) \cdot p(y)}$	(Tan et al., 2002)
46. J measure	$\max\{p(xy) \log \frac{p(y x)}{p(y)} + p(x)y \log \frac{p(x y)}{p(x)}, p(xy) \log \frac{p(x y)}{p(x)} + p(x)y \log \frac{p(y x)}{p(y)}\}$	(Tan et al., 2002)
47. Gini index	$\max\{p(x) \cdot (p(y x)^2 + p(y x)^2) - p(y)^2, p(x) \cdot (p(y x)^2 + p(y x)^2) - p(x)^2, p(y) \cdot (p(x y)^2 + p(x y)^2) - p(x)^2, p(y) \cdot (p(x y)^2 + p(x y)^2) - p(y)^2\}$	(Tan et al., 2002)
48. Confidence	$\max\{p(y x), p(x y)\}$	(Tan et al., 2002)
49. Laplace	$\max\{\frac{Np(xy)+1}{Np(x)+2}, \frac{Np(xy)+1}{Np(y)+2}\}$	(Tan et al., 2002)
50. Conviction	$\max\{\frac{p(x) \cdot p(y)}{p(xy)}, \frac{p(x) \cdot p(y)}{p(xy)}\}$	(Tan et al., 2002)
51. Piatersky-Shapiro	$p(xy) - p(x) \cdot p(y)$	(Tan et al., 2002)
52. Certainty factor	$\max\{\frac{p(y x) - p(y)}{1 - p(y)}, \frac{p(x y) - p(x)}{1 - p(x)}\}$	(Tan et al., 2002)
53. Added value (AV)	$\max\{p(y x) - p(y), p(x y) - p(x)\}$	(Tan et al., 2002)
54. Collective strength	$\frac{p(xy) + p(xy)}{p(x) \cdot p(y) + p(x) \cdot p(y)} \cdot \frac{1 - p(x) \cdot p(y) - p(x) \cdot p(y)}{1 - p(xy) - p(xy)}$	(Tan et al., 2002)
55. Klosgen	$\sqrt{p(xy)} \cdot AV$	(Tan et al., 2002)

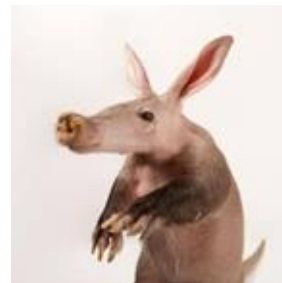
# What happens as this matrix grows?

- Counts: term-frequency
  - remove stop words
  - use  $\log_{10}(\text{tf})$
  - normalize by document length

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	0	7	13
good	114	80	62	89
fool	36	58	1	4
wit	20	15	2	3

# Dimensionality Reduction

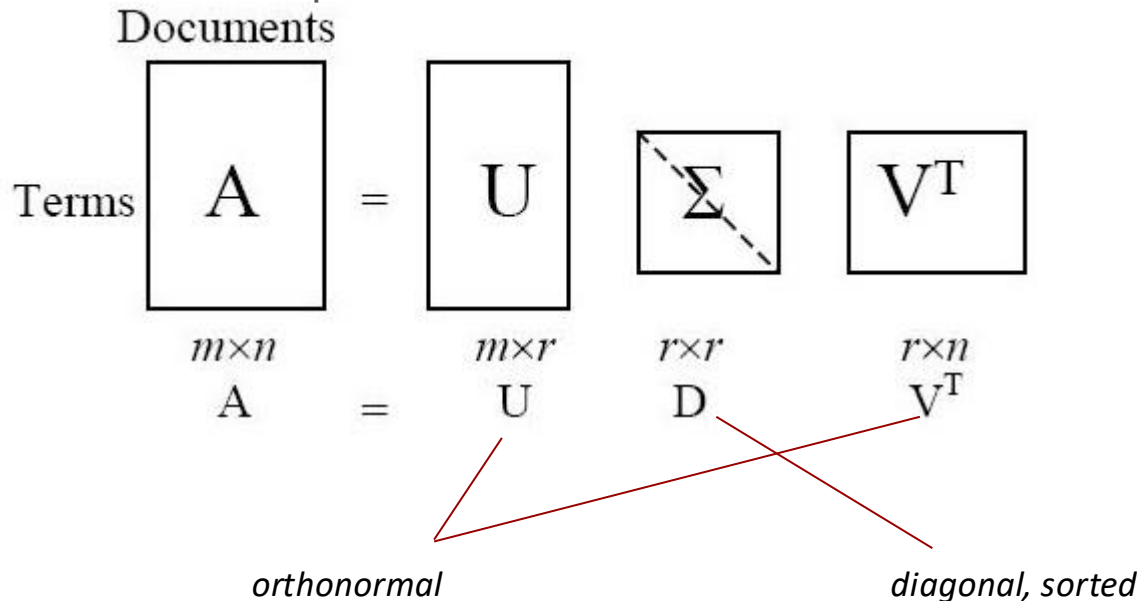
- Wikipedia: ~29 million English documents. Vocab: ~1M words.
  - High dimensionality of word--document matrix
    - Sparsity
    - The order of rows and columns doesn't matter
- Goal:
  - good similarity measure for words or documents
  - dense representation
- Sparse vs Dense vectors
  - Short vectors 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



A	0
a	0
aa	0
aal	0
aalii	0
aam	0
Aani	0
<b>aardvark</b>	<b>1</b>
aardwolf	0
...	0
zymotoxic	0
zymurgy	0
Zyrenian	0
Zyrian	0
Zyryan	0
zythem	0
Zythia	0
zythum	0
Zyzomys	0
Zyzzogeton	0

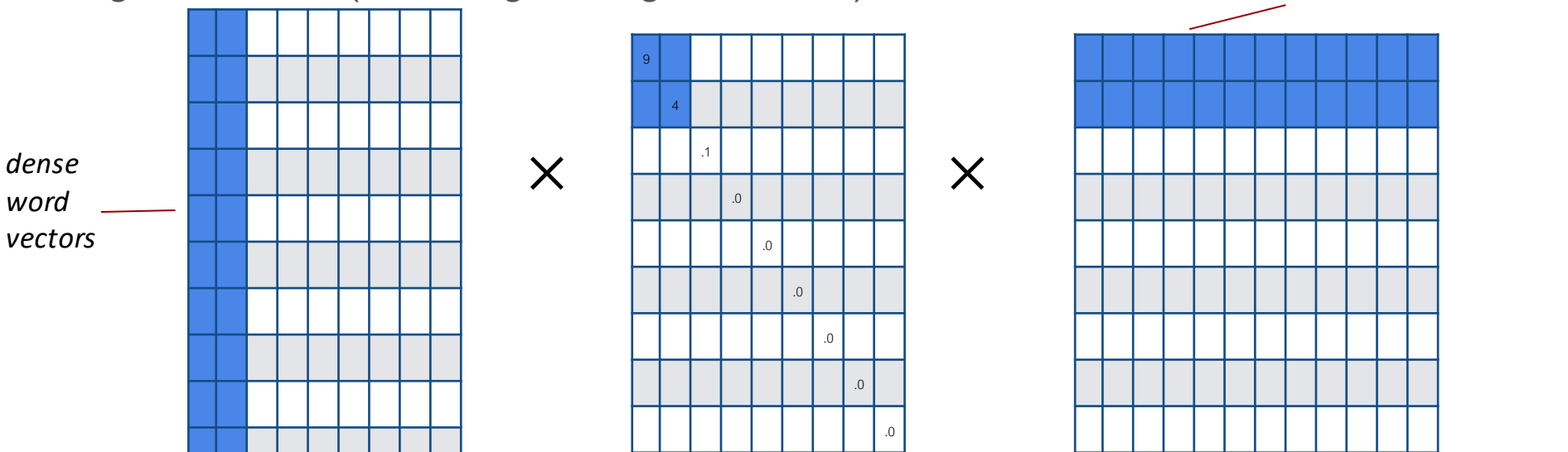
# Singular Value Decomposition (SVD)

- Solution idea:
  - Find a projection into a low-dimensional space (~300 dim)
  - That gives us a best separation between features



# Truncated SVD

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



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

# Latent Semantic Analysis

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

[Deerwester et al., 1990]

# Evaluation

- Extrinsic
- Intrinsic
- Qualitative

WORD	d1	d2	d3	d4	d5	...	d50
summer	0.12	0.21	0.07	0.25	0.33	...	0.51
spring	0.19	0.57	0.99	0.30	0.02	...	0.73
fall	0.53	0.77	0.43	0.20	0.29	...	0.85
light	0.00	0.68	0.84	0.45	0.11	...	0.03
clear	0.27	0.50	0.21	0.56	0.25	...	0.32
blizzard	0.15	0.05	0.64	0.17	0.99	...	0.23

# Extrinsic Evaluation

- Topic categorization
- Sentiment analysis
- Metaphor detection
- Machine translation
- etc.

# Intrinsic Evaluation

word1	word2	similarity (humans)
vanish	disappear	9.8
behave	obey	7.3
belief	impression	5.95
muscle	bone	3.65
modest	flexible	0.98
hole	agreement	0.3

similarity (embeddings)
1.1
0.5
0.3
1.7
0.98
0.3

Spearman's rho (human ranks, model ranks)

- WS-353 (Finkelstein et al. '02)
- MEN-3k (Bruni et al. '12)
- SimLex-999 dataset (Hill et al., 2015)

# Qualitative: Visualisation

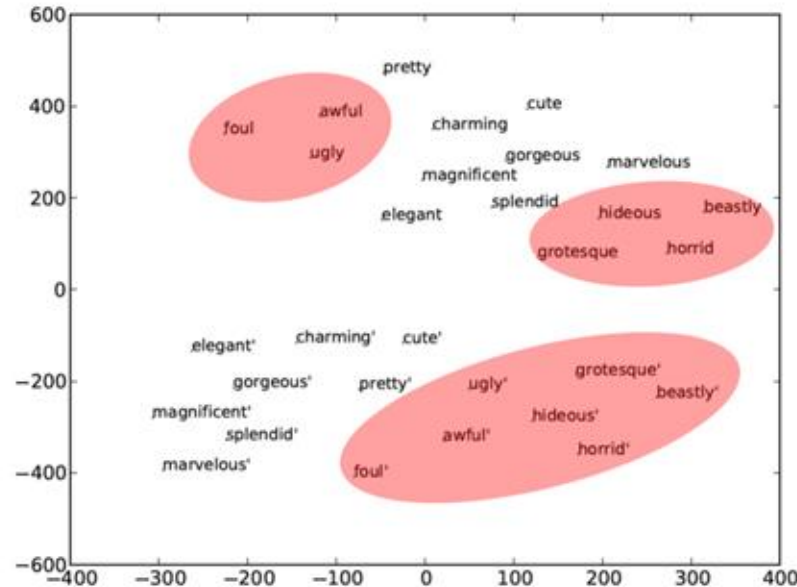


Figure 6.5: Monolingual (top) and multilingual (bottom; marked with apostrophe) word projections of the antonyms (shown in red) and synonyms of “beautiful”.

- Visualizing Data using t-SNE (van der Maaten & Hinton’08)

# Distributed representations

## Word Vectors

WORD	d1	d2	d3	d4	d5	...	d50
summer	0.12	0.21	0.07	0.25	0.33	...	0.51
spring	0.19	0.57	0.99	0.30	0.02	...	0.73
fall	0.53	0.77	0.43	0.20	0.29	...	0.85
light	0.00	0.68	0.84	0.45	0.11	...	0.03
clear	0.27	0.50	0.21	0.56	0.25	...	0.32
blizzard	0.15	0.05	0.64	0.17	0.99	...	0.23

# We'll discuss 2 kinds of embeddings

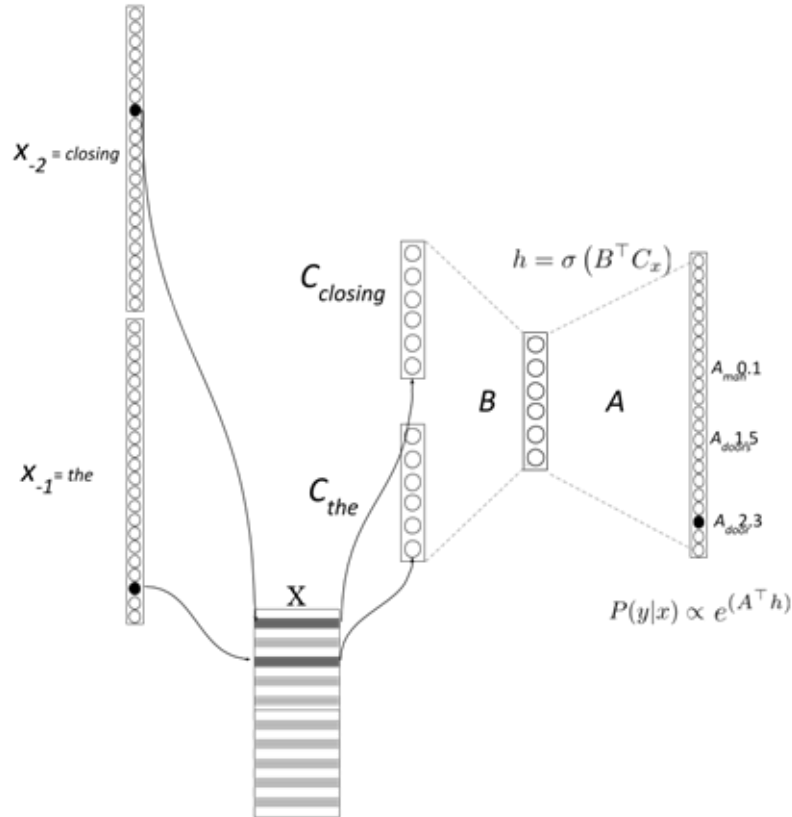
- **tf-idf**

- Information Retrieval workhorse!
- A common baseline model
- Sparse vectors
- Words are represented by (a simple function of) the counts of nearby words

- **Word2vec**

- Dense vectors
- Representation is created by training a classifier to predict whether a word is likely to appear nearby
- <https://fasttext.cc/docs/en/crawl-vectors.html>
- Later we'll discuss extensions called **contextual embeddings**

# “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



Why GitHub? ▾ Enterprise Explore ▾ Marketplace Pricing ▾

Search



Sign in

Sign up

tmikolov / word2vec

Watch

55

★ Star

840

Fork

346

<> Code

Issues 38

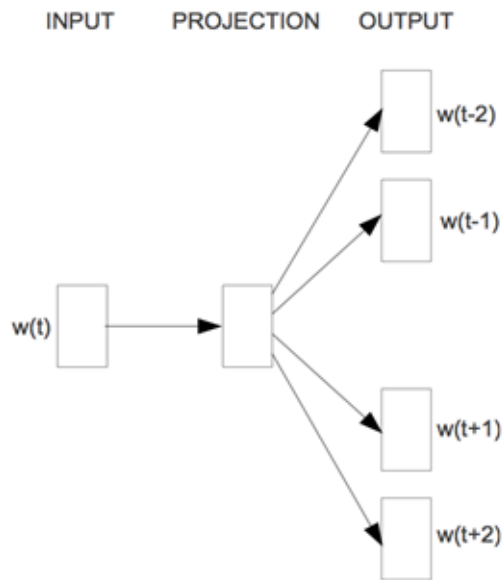
Pull requests 4

Projects 0

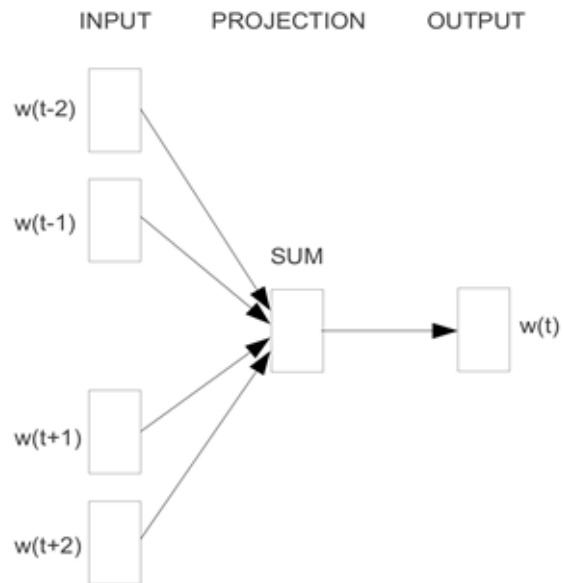
Security

Insights

# Word2Vec



**Skip-gram**



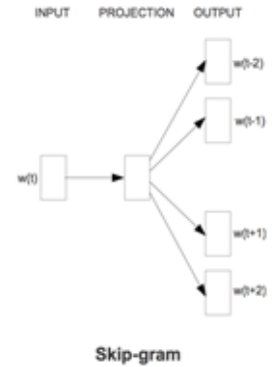
**CBOW**

- [Mikolov et al.' 13]

# Skip-gram Prediction

- Predict vs Count

the cat sat on the mat





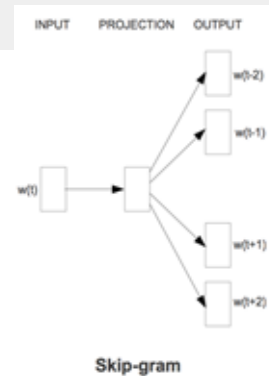
# Skip-gram Prediction

- Predict vs Count

the cat sat on the mat



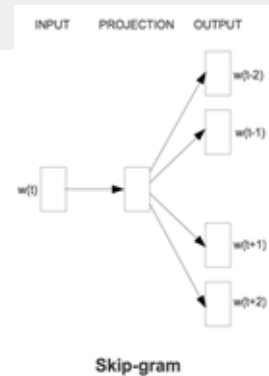
context size = 2



# Skip-gram Prediction

- Predict vs Count

the cat sat on the mat



context size = 2

# Skip-gram Prediction

- Predict vs Count

the cat sat on the mat

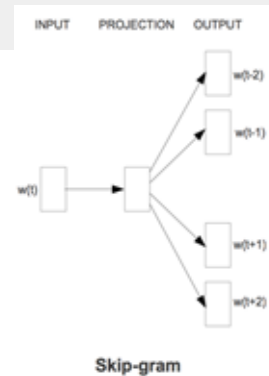
$w_t = \text{on}$



**CLASSIFIER**



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



context size = 2



# Skip-gram Prediction

- Predict vs Count

the cat sat on the mat

$w_t = \text{mat}$



**CLASSIFIER**

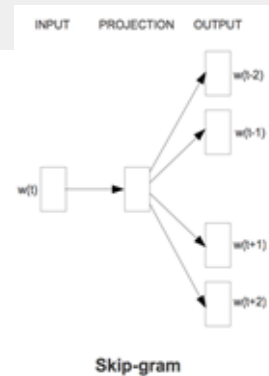


$w_{t-2} = \text{on}$

$w_{t-1} = \text{the}$

$w_{t+1} = \langle \text{end}_{+1} \rangle$

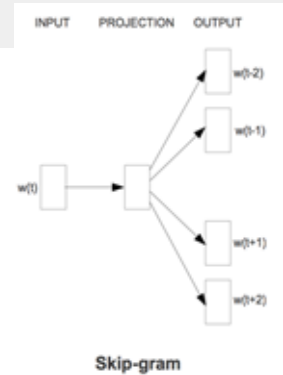
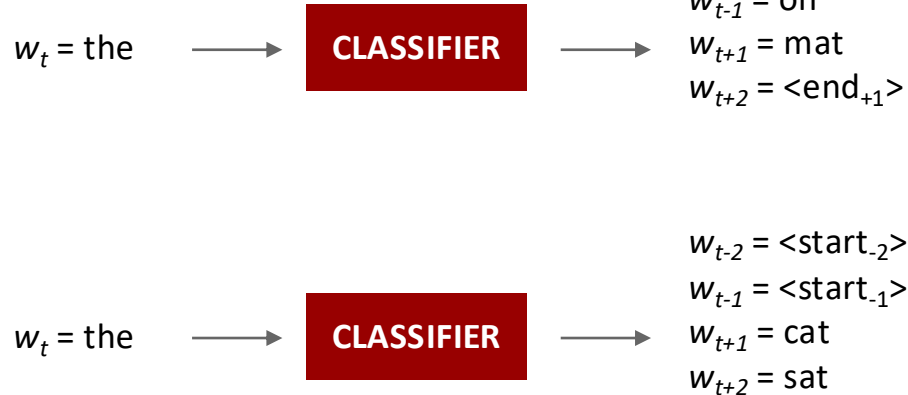
$w_{t+2} = \langle \text{end}_{+2} \rangle$



context size = 2

# Skip-gram Prediction

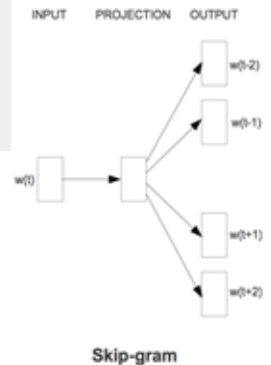
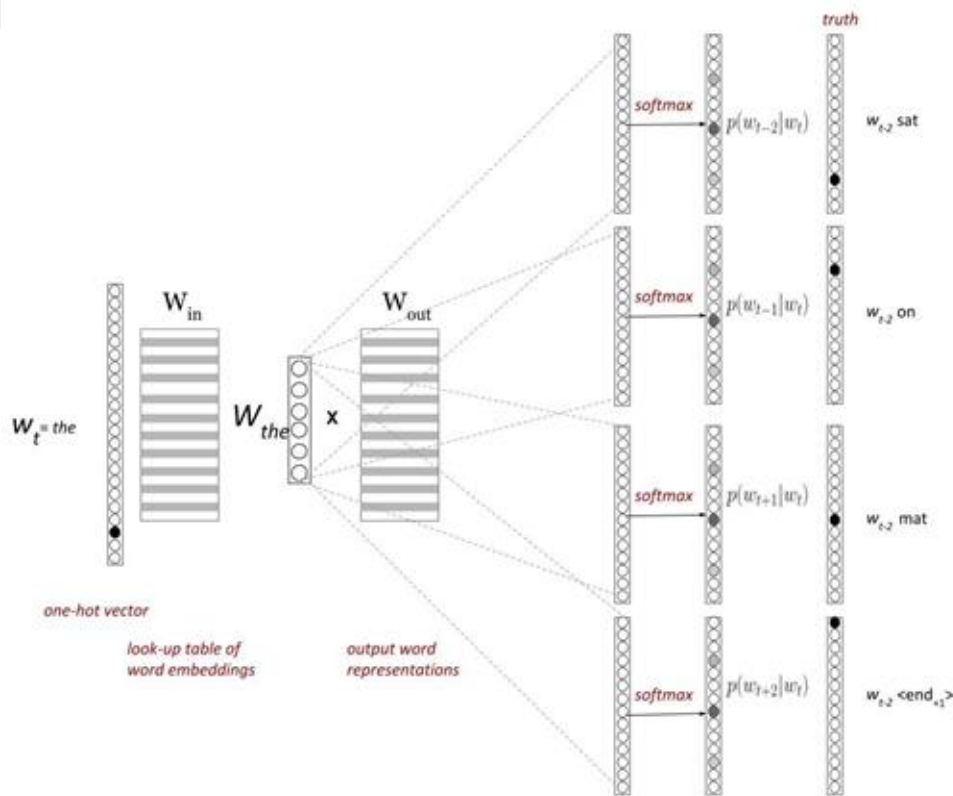
- Predict vs Count



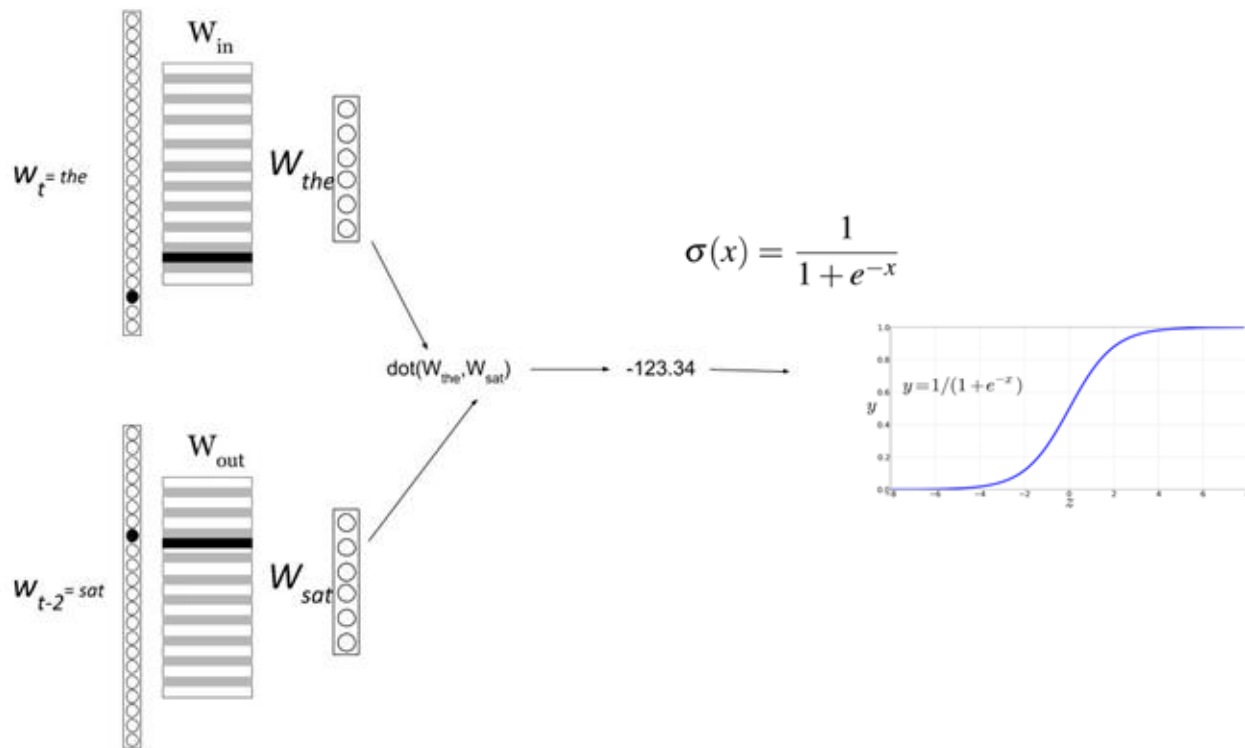
# Skip-gram Prediction

Conceptual idea, not the actual architecture

Challenge: very large softmax is expensive!



# Make binary choice instead! $p(+ | t, c)$



# Approach: predict if candidate word $c$ is a “neighbor”

1. Treat the target word  $t$  and a neighboring context word  $c$  as **positive examples**
2. Randomly sample other words in the lexicon to get negative examples
3. Use logistic regression to train a classifier to distinguish those two cases
4. Use the learned weights as the embeddings

# Skip gram with negative sampling (SGNS)

Given a tuple  $(t,c)$  = target, context

- (cat, sat)
- (cat, aardvark)

$$P(+|t,c) = \frac{1}{1 + e^{-t \cdot c}}$$

Return probability that  $c$  is a real context word

$$\begin{aligned} P(-|t,c) &= 1 - P(+|t,c) \\ &= \frac{e^{-t \cdot c}}{1 + e^{-t \cdot c}} \end{aligned}$$

Negative examples sampled from a (smoothed) uniform distribution

# Learning the classifier

- Iterative process
  - We'll start with 0 or random weights
  - Then adjust the word weights to
    - make the positive pairs more likely
    - and the negative pairs less likely
  - over the entire training set:

$$\sum_{(t,c) \in +} \log P(+|t,c) + \sum_{(t,c) \in -} \log P(-|t,c)$$

- Train using gradient descent

# Properties of Embeddings

The kinds of neighbors depend on window size

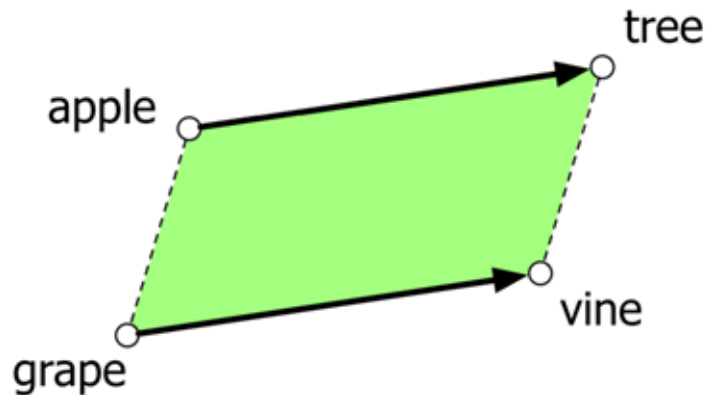
- Small windows ( $C = +/- 2$ ) : nearest words are syntactically similar words in same taxonomy
  - Hogwarts nearest neighbors are other fictional schools
  - Sunnydale, Evernight, Blandings
- Large windows ( $C = +/- 5$ ) : nearest words are related
  - Hogwarts nearest neighbors are Harry Potter world:
  - Dumbledore, half-blood, Malfoy

# Analogical relations

The classic parallelogram model of analogical reasoning (Rumelhart and Abrahamson 1973)

To solve: “apple is to tree as grape is to \_\_\_\_\_”

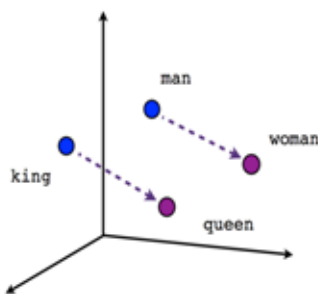
→ → → →  
Add tree – apple to grape to get **vine**



# Analogy: Embeddings capture relational meaning!

$\text{vector}('king') - \text{vector}('man') + \text{vector}('woman') \approx \text{vector}('queen')$

$\text{vector}('Paris') - \text{vector}('France') + \text{vector}('Italy') \approx \text{vector}('Rome')$



Male-Female



Verb tense

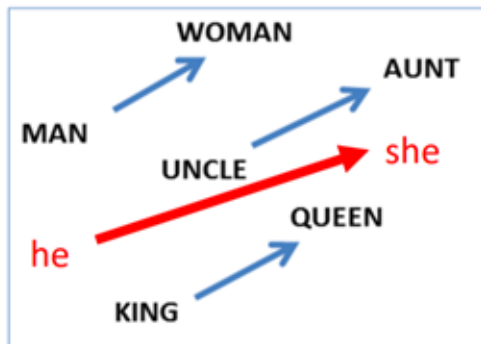


Country-Capital

$$\min \cos(\text{man} - \text{woman}, \text{king} - x) \text{ s.t. } \|\text{king} - x\|_2 < \delta$$

# Bias in word embeddings

Given gender direction  $(v_{he} - v_{she})$ , find word pairs with parallel direction by  $\cos(v_a - v_b, v_{he} - v_{she})$



he: _____	she: _____
brother	sister
beer	
physician	
professor	

Google w2v embedding trained from the news



# Bias in literature



## Male

---

- strong
- arrogant
- sexual
- active
- dominant
- violent
- beautiful
- angry



Male Adjectives



Female Adjectives

- ## Female
- 
- weak
  - submissive
  - childish
  - afraid
  - dependent
  - hysterical
  - domestic
  - emotional

# Embeddings can help study word history!

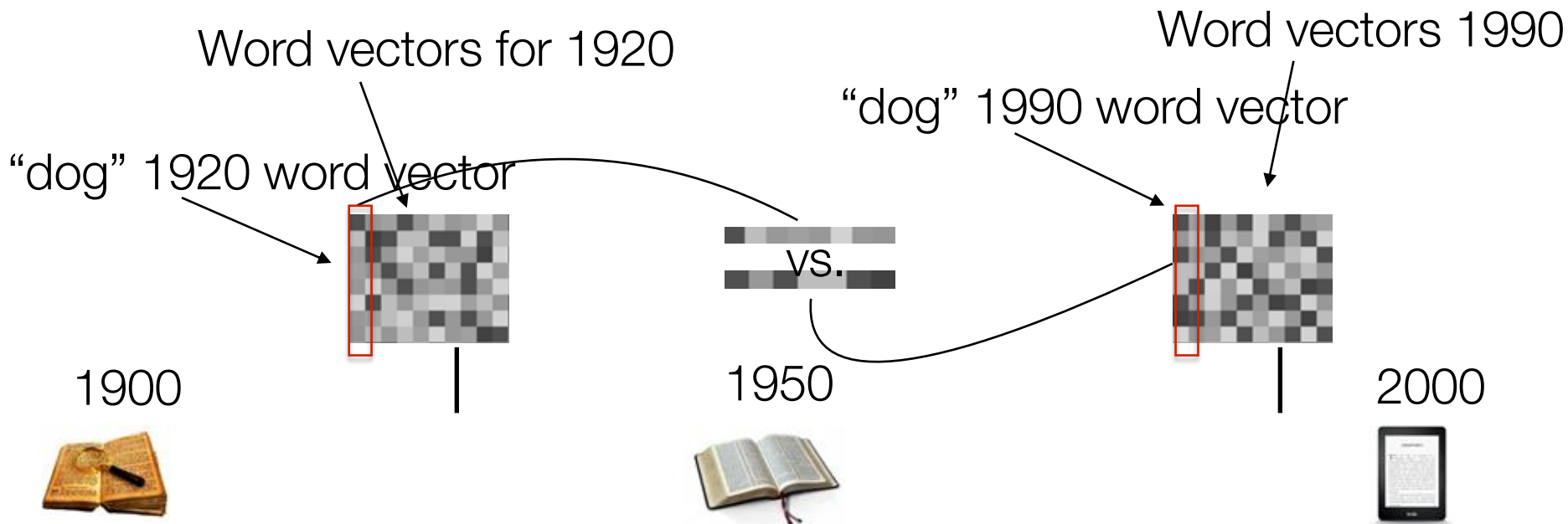
## **Diachronic Word Embeddings Reveal Statistical Laws of Semantic Change**

**William L. Hamilton, Jure Leskovec, Dan Jurafsky**

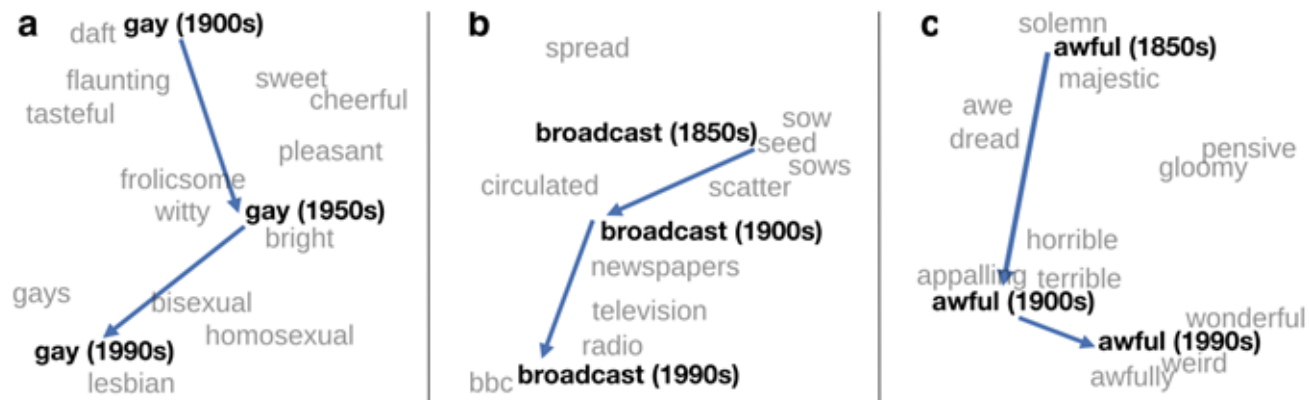
Department of Computer Science, Stanford University, Stanford CA, 94305

wleif, jure, jurafsky@stanford.edu

# Diachronic Embeddings

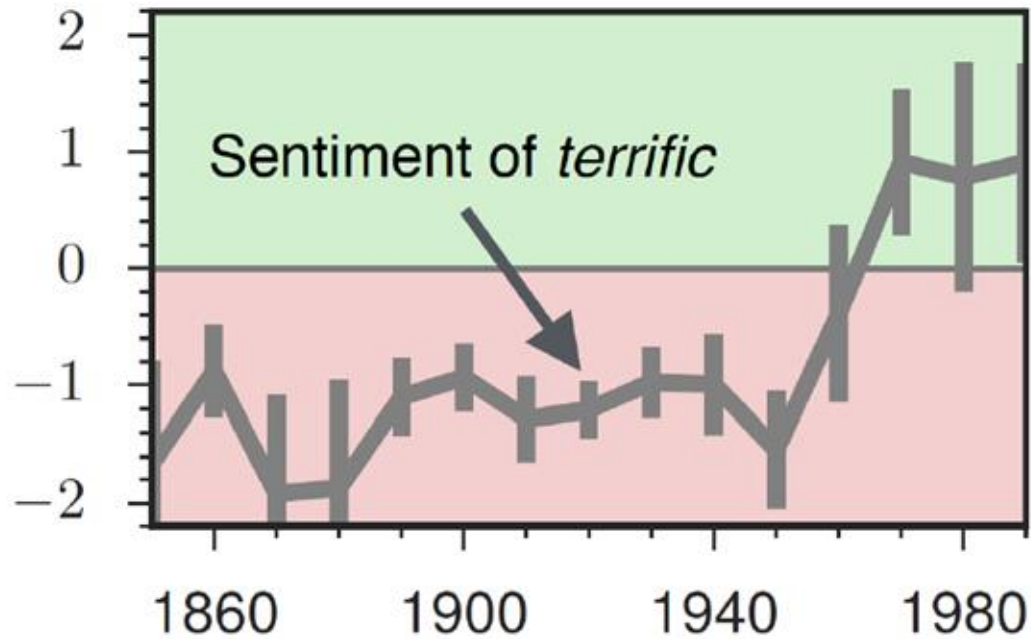


# Project 300 dimensions down into 2



~30 million books, 1850-1990, Google Books data

# Negative words change faster than positive words



# Embeddings reflect ethnic stereotypes over time

PNAS Proceedings of the National Academy of Sciences of the United States of America

Home Articles Front Matter News Podcasts

NEW RESEARCH IN Physical Sciences Social Sc

## Word embeddings quantify 100 years of gender and ethnic stereotypes

Nikhil Garg, Londa Schiebinger, Dan Jurafsky, and James Zou

PNAS April 17, 2018 115 (16) E3635-E3644; published ahead of print April 3, 2018

# Change in linguistic framing 1910-1990

Change in association of Chinese names with adjectives framed as "othering"  
(barbaric, monstrous, bizarre)

