ENLP Lecture 21b
Word & Document Representations; Distributional Similarity

Nathan Schneider

(some slides by Marine Carpuat, Sharon Goldwater, Dan Jurafsky)

28 November 2016

Topics

- Similarity
- Thesauri & their limitations
- Distributional hypothesis
- Clustering (Brown clusters, LDA)
- Vector representations (count-based, dimensionality reduction, embeddings)

Word & Document Similarity

Question Answering

- **Q: What is a good way to remove wine stains?**
- A1: Salt is a great way to eliminate wine stains
- A2: How to get rid of wine stains…
- A3: How to get red wine out of clothes…
- A4: Oxalic acid is infallible in removing iron-rust and ink stains.

Document Similarity

- Given a movie script, recommend similar movies.


```
                    ALAN TURING (V.O.)
          Are you paying attention?

INT. ALAN TURING'S HOUSE - DAY - 1951

A HALF-DOZEN POLICE OFFICERS swarm the Manchester home of
mathematics professor Alan Turing.

                    ALAN TURING (V.O.)
          Good. This is going to go very
          quickly now. If you are not
          listening carefully, you will miss
          things. Important things. You're
          writing some of this down? That's
          good.
```

```
                    BERTIE
          Like mad King George the Third, there'll be
          King George the stammerer, who let his
          people down so badly in their hour of need!

Lionel sits down on the chair of Edward the Confessor.
Leaning against it is the great two-handed sword of St.
George.

                    BERTIE
          What're you doing? Get up! You can't sit
          there!

                    LIONEL
          Why not? It's a chair.
```

Word Similarity

Intuition of Semantic Similarity

Semantically close
- bank–money
- apple–fruit
- tree–forest
- bank–river
- pen–paper
- run–walk
- mistake–error
- car–wheel

Semantically distant
- doctor–beer
- painting–January
- money–river
- apple–penguin
- nurse–fruit
- pen–river
- clown–tramway
- car–algebra

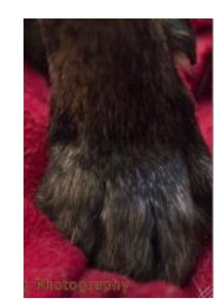
Why are 2 words similar?

- Meaning
  - The two concepts are close in terms of their meaning
- World knowledge
  - The two concepts have similar properties, often occur together, or occur in similar contexts
- Psychology
  - We often think of the two concepts together

Two Types of Relations

- Synonymy: two words are (roughly) interchangeable


- Semantic similarity (distance): somehow “related”
  - Sometimes, explicit lexical semantic relationship, often, not

Validity of Semantic Similarity

- Is semantic distance a valid linguistic phenomenon?
- Experiment (Rubenstein and Goodenough, 1965)
  - Compiled a list of word pairs
  - Subjects asked to judge semantic distance (from 0 to 4) for each of the word pairs
- Results:
  - Rank correlation between subjects is ~0.9
  - People are consistent!

Why do this?

- Task: automatically compute semantic similarity between words
- Can be useful for many applications:
  - Detecting paraphrases (i.e., automatic essay grading, plagiarism detection)
  - Information retrieval
  - Machine translation
- Why? Because similarity gives us a way to generalize beyond word identities

Evaluation: Correlation with Humans

- Ask automatic method to rank word pairs in order of semantic distance
- Compare this ranking with human-created ranking
- Measure correlation

Evaluation: Word-Choice Problems

Identify that alternative which is closest in meaning to the target:

accidental	imprison
wheedle	incarcerate
ferment	writhe
inadvertent	meander
abominate	inhibit

Evaluation: Malapropisms

Jack withdrew money from the ATM next to the band.

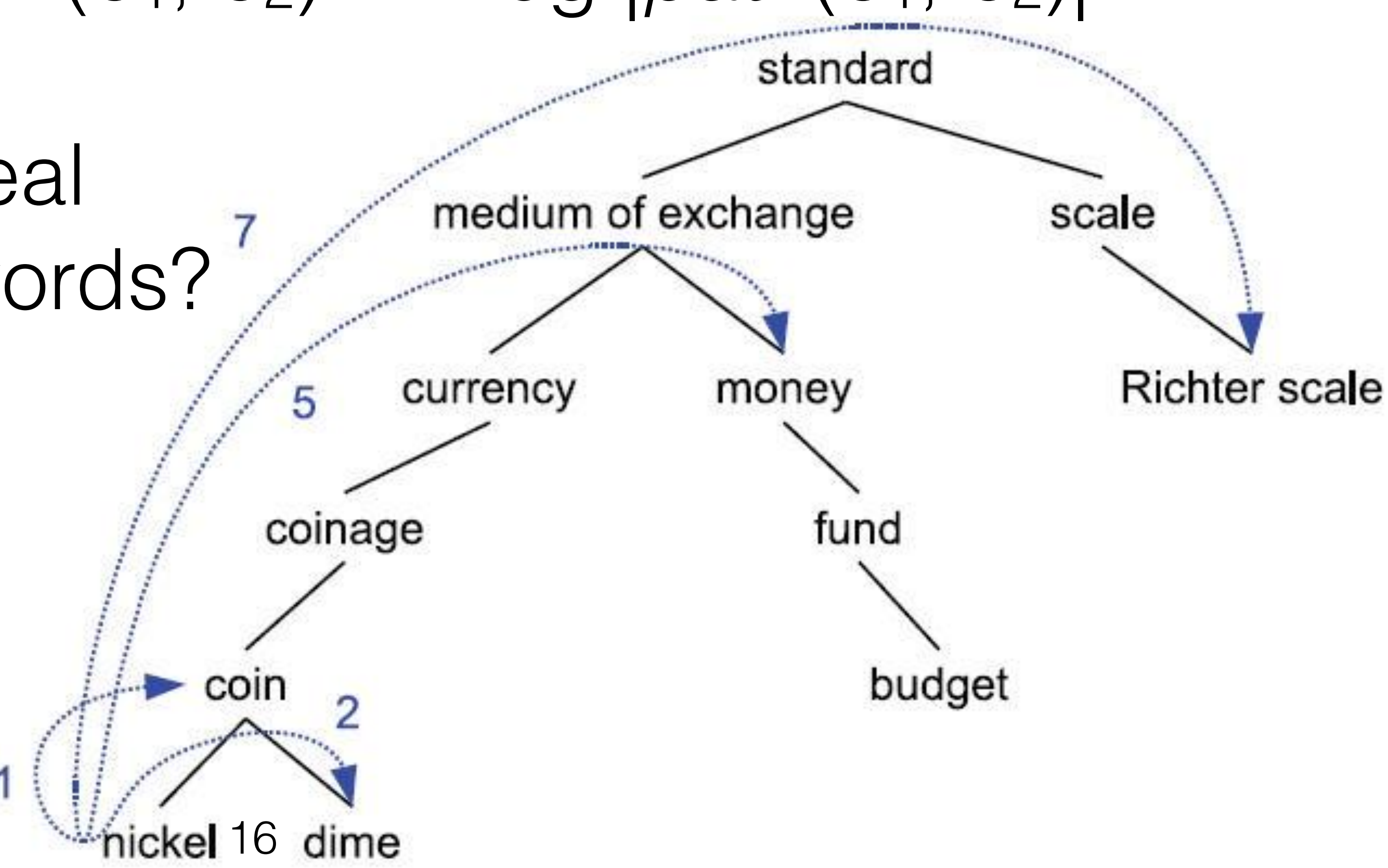
band is unrelated to all of the other words in its context…

Word Similarity: Two Approaches

- Thesaurus-based
  - We've invested in all these resources… let's exploit them!
- Distributional
  - Count words in context

Thesaurus-based Similarity

- Use the structure of a resource like WordNet
- Examine the relationship between the two concepts, use a metric that converts the relationship into a real number
- E.g., **path length**: $sim(c_1, c_2) = -\log |path(c_1, c_2)|$
  - How would you deal with ambiguous words?



Thesaurus Methods: Limitations

- Measure is only as good as the resource
- Limited in scope
  - Assumes IS-A relations
  - Works mostly for nouns
- Role of context not accounted for
- Not easily domain-adaptable
- Resources not available in many languages

Distributional Similarity

"Differences of meaning correlates with differences of distribution" (Harris, 1970)

- Idea: similar linguistic objects have similar **contents** (for documents, sentences) or **contexts** (for words)

Two Kinds of Distributional Contexts

1. Documents as bags-of-words
   - Similar documents contain similar words; similar words appear in similar documents
2. Words in terms of neighboring words
   - "You shall know a word by the company it keeps!" (Firth, 1957)
   - Similar words occur near similar sets of other words (e.g., in a 5-word window)

- He handed her a glass of bardiwac.
- Beef dishes are made to complement the bardiwac.
- Nigel staggered to his feet, face flushed from too much bardiwac.
- Malbec, one of the lesser-known bardiwac grapes, responds well to Australia's sunshine.
- I dined off bread and cheese and this excellent bardiwac.
- The drinks were delicious: blood-red bardiwac as well as light, sweet Rhenish.

Word Vectors

- A word **type** can be represented as a vector of features indicating the contexts in which it occurs in a corpus

$$\vec{w} = (f_1, f_2, f_3, \ldots f_N)$$

Context Features

- Word co-occurrence within a window:

	arts	boil	data	function	large	sugar	summarized	water
apricot	0	1	0	0	1	1	0	1
pineapple	0	1	0	0	1	1	0	1
digital	0	0	1	1	1	0	1	0
information	0	0	1	1	1	0	1	0

- Grammatical relations:

	subj-of, absorb	*subj-of*, adapt	*subj-of*, behave	...	*pobj-of*, inside	*pobj-of*, into	...	*nmod-of*, abnormality	*nmod-of*, anemia	*nmod-of*, architecture	...	*obj-of*, attack	*obj-of*, call	*obj-of*, come from	*obj-of*, decorate	...	*nmod*, bacteria	*nmod*, body	*nmod*, bone marrow
cell	1	1	1		16	30		3	8	1		6	11	3	2		3	2	2

Context Features

- Feature values
  - Boolean
  - Raw counts
  - Some other weighting scheme (e.g., *idf*, *tf.idf*)
  - Association values (next slide)

Association Metric

- Commonly-used metric: Pointwise Mutual Information

$$\text{association}_{\text{PMI}}(w, f) = \log_2 \frac{P(w, f)}{P(w)P(f)}$$

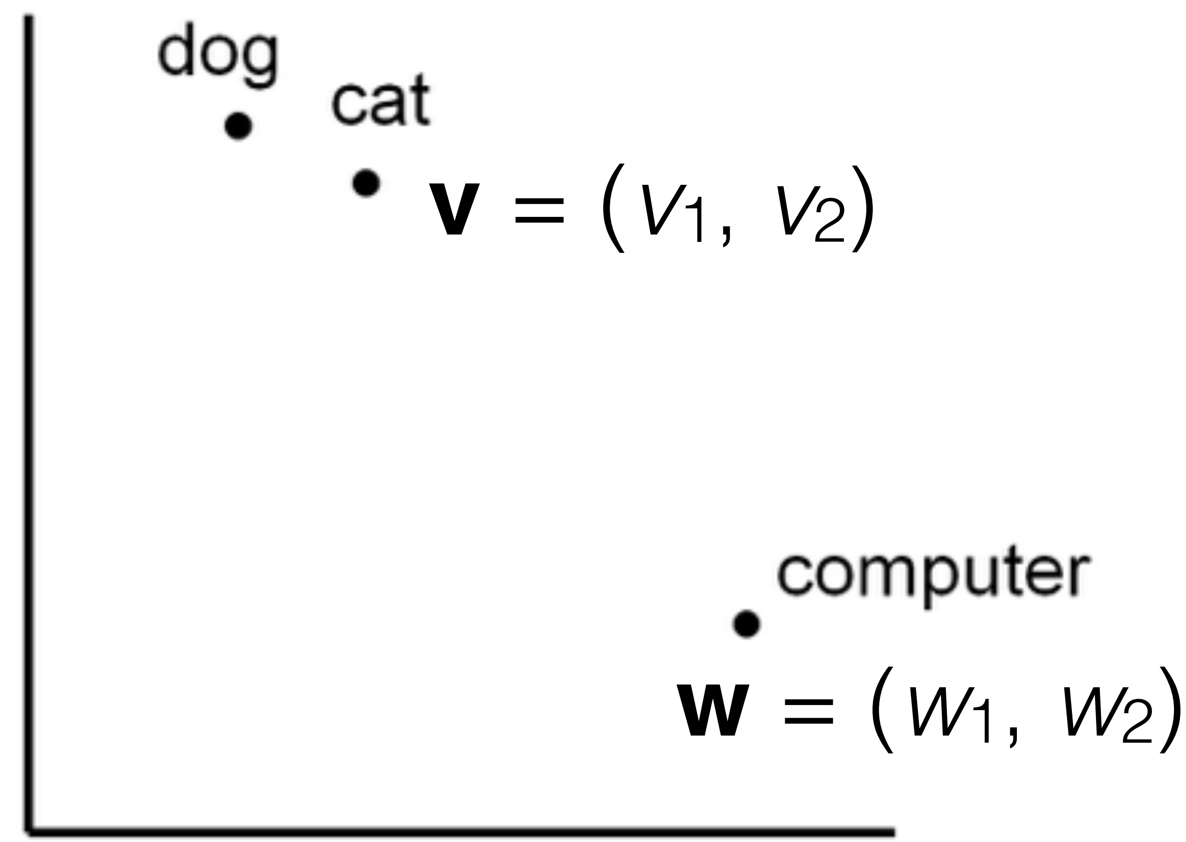
- Can be used as a feature value or by itself

Computing Similarity

- Semantic similarity boils down to computing some measure on context vectors

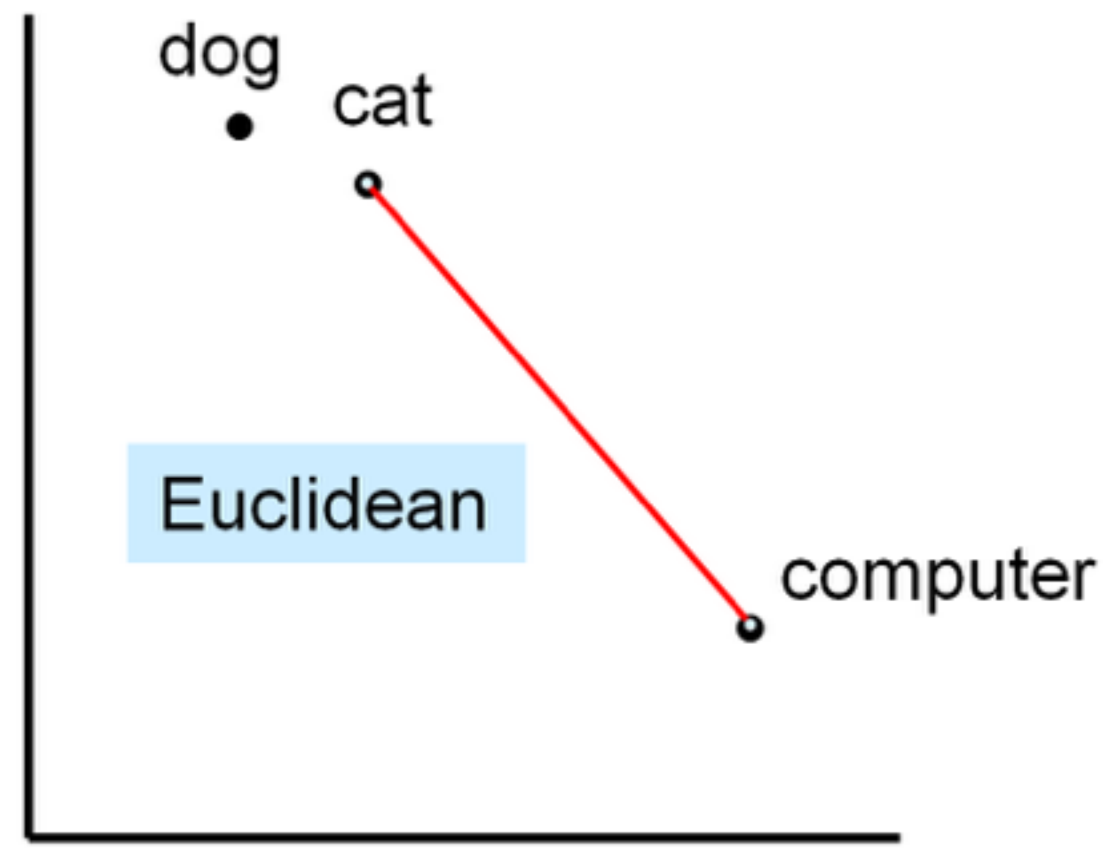
Words in a Vector Space

- In 2 dimensions:
  v = “cat”
  w = “computer”



Euclidean Distance

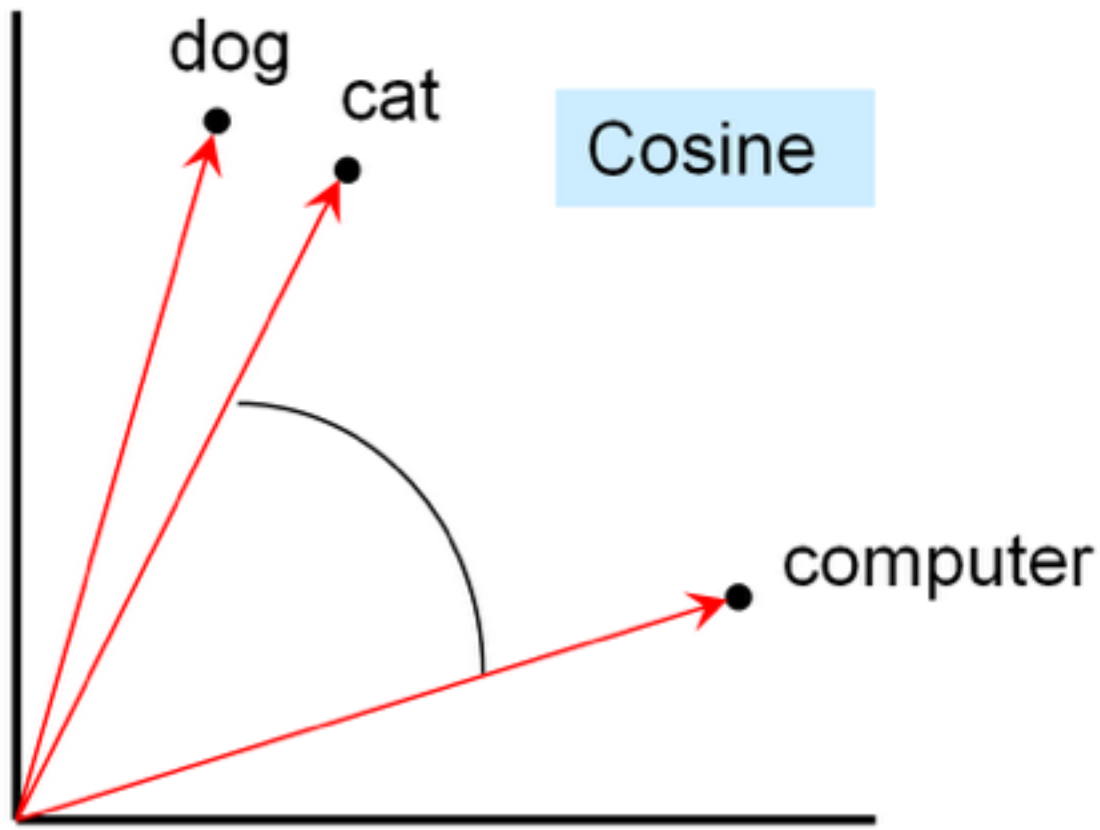
- $\sqrt{\sum_i (v_i - w_i)^2}$
- Can be oversensitive to extreme values



Cosine Similarity

- Cosine distance: borrowed from information retrieval

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



Distributional Approaches: Discussion

- No thesauri needed: data driven
- Can be applied to any pair of words
- Can be adapted to different domains

Distributional Profiles: Example

DP of *star*

space 0.21
movie 0.16
famous 0.15
light 0.12
rich 0.11
heat 0.08
planet 0.07
hydrogen 0.07

DP of *fusion*

heat 0.16
hydrogen 0.16
energy 0.13
hot 0.09
light 0.09
space 0.04
gravity 0.03
pressure 0.03

Distributional Profiles: Example

DP of *star*

- *space* 0.21
- *movie* 0.16
- *famous* 0.15
- *light* 0.12
- *rich* 0.11
- *heat* 0.08
- *planet* 0.07
- *hydrogen* 0.07

DP of *fusion*

- *heat* 0.16
- *hydrogen* 0.16
- *energy* 0.13
- *hot* 0.09
- *light* 0.09
- *space* 0.04
- *gravity* 0.03
- *pressure* 0.03

Problem?

DP of *star*	**DP of *fusion***
space 0.21	*heat* 0.16
movie 0.16	*hydrogen* 0.16
famous 0.15	*energy* 0.13
light 0.12	*hot* 0.09
rich 0.11	*light* 0.09
heat 0.08	*space* 0.04
planet 0.07	*gravity* 0.03
hydrogen 0.07	*pressure* 0.03

Distributional Profiles of Concepts

DP of CELESTIAL BODY

(celestial body, star, sun,...)

space 0.36
light 0.27
heat 0.11
planet 0.07
hydrogen 0.06
hot 0.01

DP of CELEBRITY

(celebrity, hero, star,...)

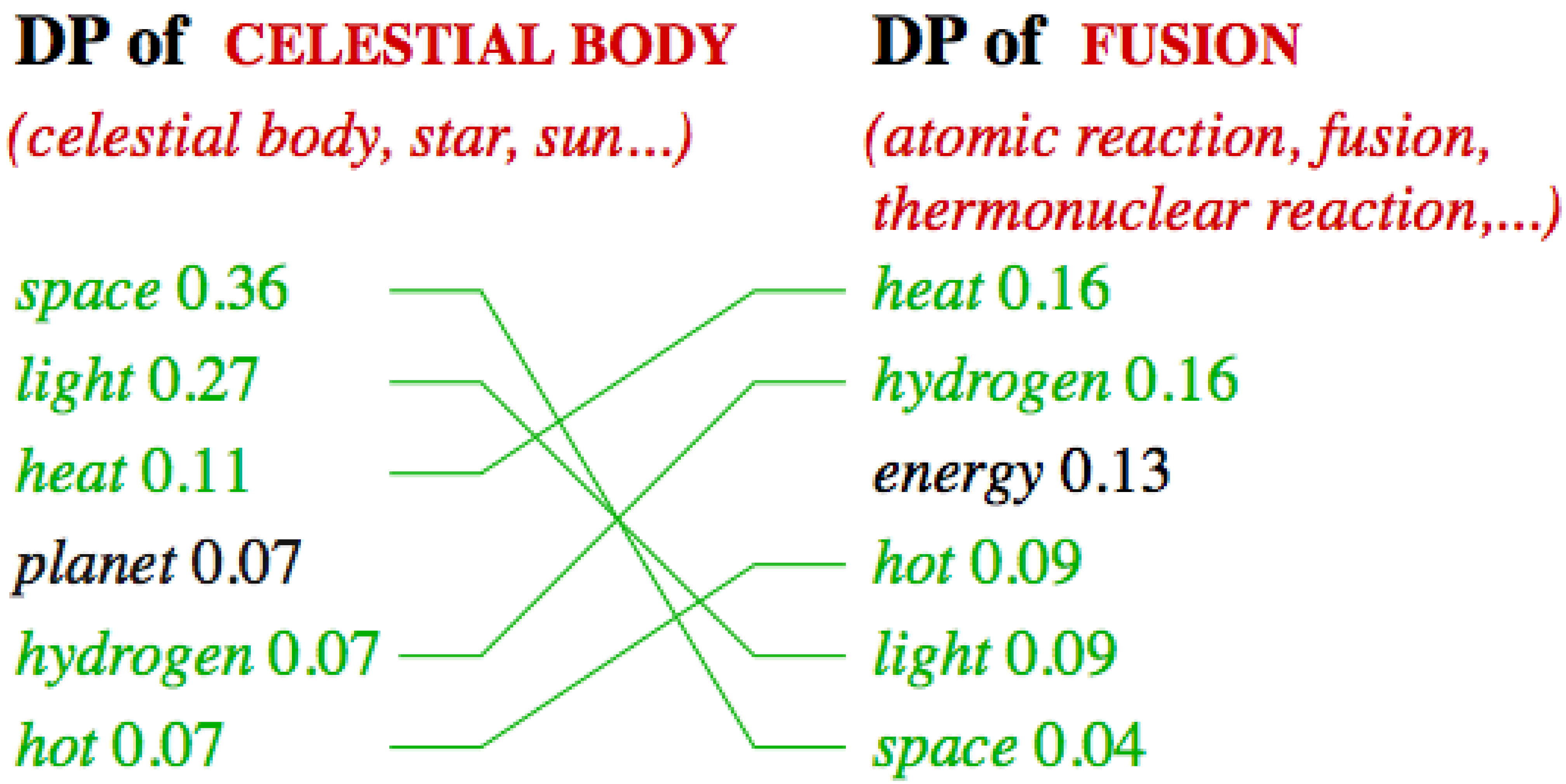
famous 0.24
movie 0.14
rich 0.14
fan 0.10
hot 0.04
fashion 0.01

Semantic Similarity: "Celebrity"

DP of CELEBRITY	**DP of FUSION**
(celebrity, hero, star,...)	*(atomic reaction, fusion, thermonuclear reaction,...)*
famous 0.24	*heat* 0.16
movie 0.14	*hydrogen* 0.16
rich 0.14	*energy* 0.13
fan 0.10	*hot* 0.09
hot 0.04	*light* 0.09
fashion 0.01	*space* 0.04

Semantically distant…

Semantic Similarity: “Celestial body”



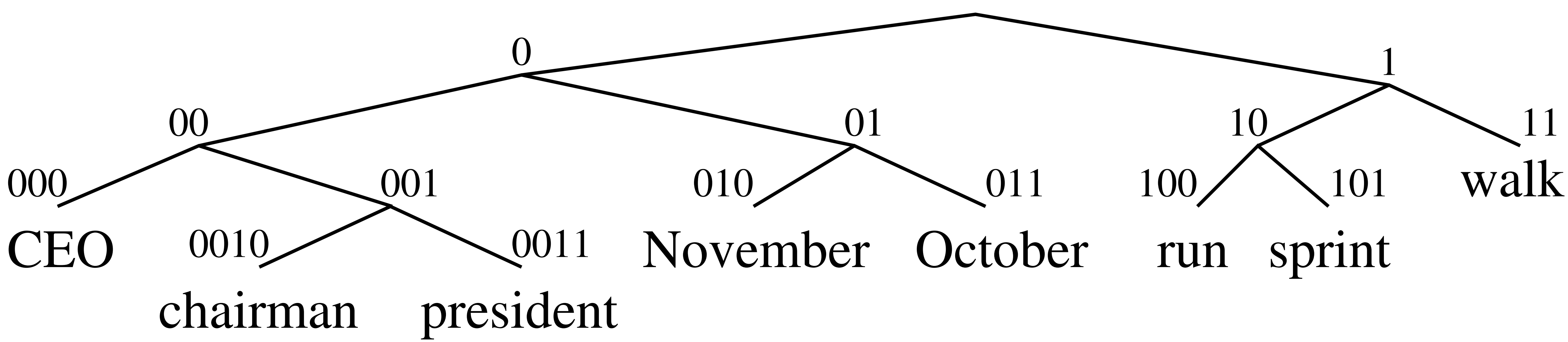
Semantically close!

Word Clusters

- E.g., **Brown clustering** algorithm produces hierarchical clusters based on word context vectors

- Words in similar parts of hierarchy occur in similar contexts

Chairman is 0010, “months” = 01, and verbs = 1



Brown clusters created from Twitter data:
http://www.cs.cmu.edu/~ark/TweetNLP/cluster_viewer.html

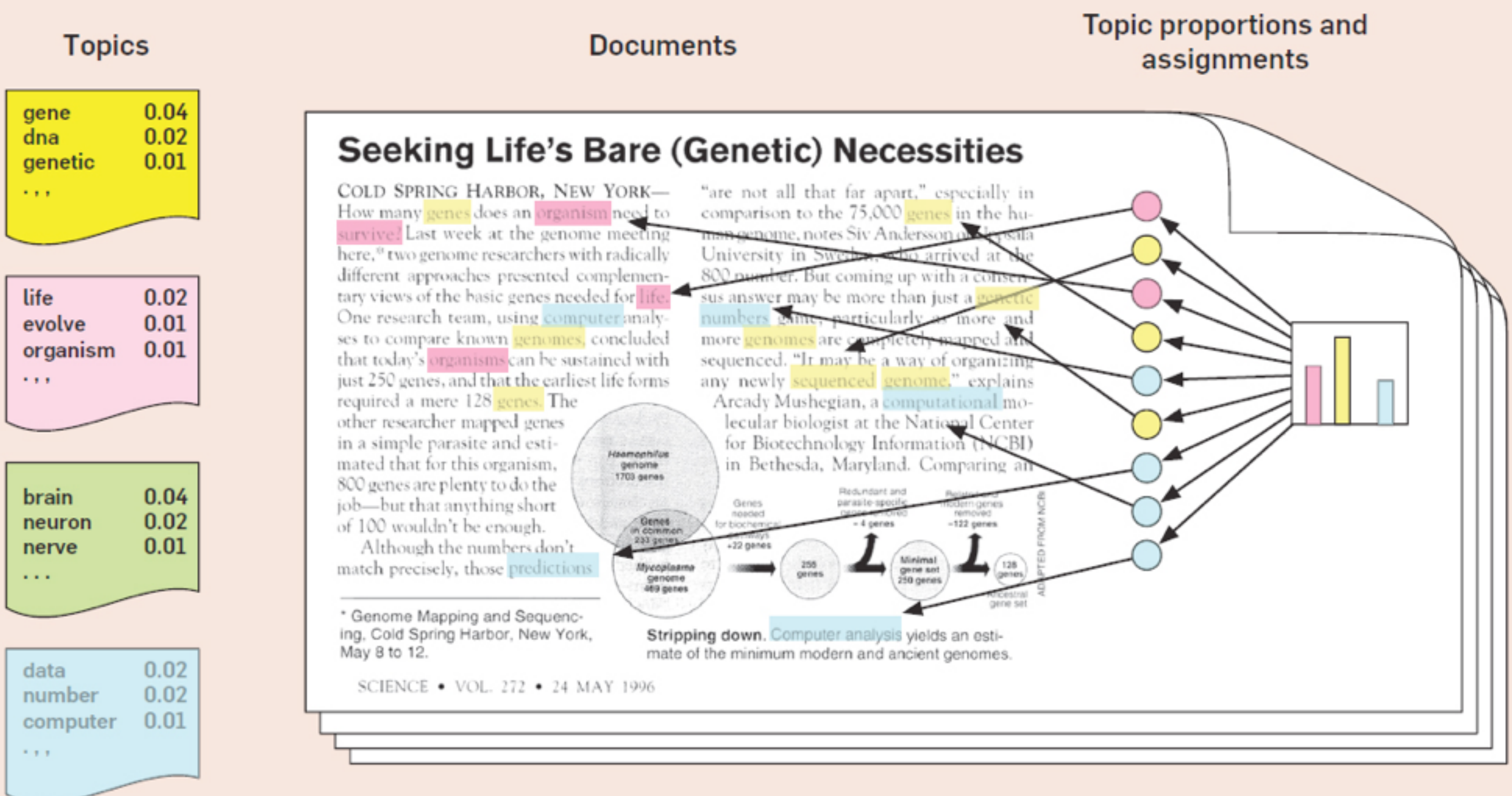
Document-Word Models

- Features in the word vector can be word context counts or PMI scores
- Also, features can be the documents in which this word occurs
  - Document occurrence features useful for **topical/ thematic** similarity

Topic Models

- Latent Dirichlet Allocation (LDA) and variants are known as **topic models**
  - Learned on a large document collection (unsupervised)
  - Latent probabilistic **clustering** of words that tend to occur in the same document. Each **topic** cluster is a distribution over words.
  - Generative model: Each document is a sparse mixture of topics. Each word in the document is chosen by sampling a topic from the document-specific topic distribution, then sampling a word from that topic.
  - Learn with EM or other techniques (e.g., Gibbs sampling)

Topic Models



http://cacm.acm.org/magazines/2012/4/147361-probabilistic-topic-models/fulltext

More on topic models

Mark Dredze (JHU)

Topic Models for Identifying Public Health Trends

Tomorrow, 11:00 in STM 326

DIMENSIONALITY REDUCTION

Slides based on presentation by
Christopher Potts

Why dimensionality reduction?

- So far, we've defined word representations as rows in **F**, a m x n matrix
  - m = vocab size
  - n = number of context dimensions / features

- Problems: n is very large, F is very sparse

- Solution: find a low rank approximation of **F**
  - Matrix of size m x d where d << n

Methods

- Latent Semantic Analysis
- Also:
  - Principal component analysis
  - Probabilistic LSA
  - Latent Dirichlet Allocation
  - Word2vec
  - …

Latent Semantic Analysis

- Based on **Singular Value Decomposition**

For any matrix of real numbers A of dimension $(m \times n)$ there exists a factorization into matrices T, S, D such that

$$A_{m\times n} = T_{m\times m} S_{m\times m} D^T_{n\times m}$$

$$\begin{pmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{pmatrix} = \begin{pmatrix} \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \end{pmatrix} \begin{pmatrix} \cdot & & \\ & \cdot & \\ & & \cdot \end{pmatrix} \begin{pmatrix} \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \end{pmatrix}^T$$

$$A_{3\times 4} \quad = \quad T_{3\times 3} \qquad S_{3\times 3} \qquad D^T_{4\times 3}$$

LSA illustrated: SVD + select top k dimensions

	d1	d2	d3	d4	d5	d6
gnarly	1	0	1	0	0	0
wicked	0	1	0	1	0	0
awesome	1	1	1	1	0	0
lame	0	0	0	0	1	1
terrible	0	0	0	0	0	1

Distance from *gnarly*
1. gnarly
2. awesome
3. terrible
4. wicked
5. lame

$\Downarrow\Uparrow$

T(erm)					
gnarly	0.41	0.00	0.71	0.00	-0.58
wicked	0.41	0.00	-0.71	0.00	-0.58
awesome	0.82	-0.00	-0.00	-0.00	0.58
lame	0.00	0.85	0.00	-0.53	0.00
terrible	0.00	0.53	0.00	0.85	0.00

$\times$

S(ingular values)					
1	2.45	0.00	0.00	0.00	0.00
2	0.00	1.62	0.00	0.00	0.00
3	0.00	0.00	1.41	0.00	0.00
4	0.00	0.00	0.00	0.62	0.00
5	0.00	0.00	0.00	0.00	-0.00

$\times$

D(ocument)					
d1	0.50	-0.00	0.50	0.00	-0.71
d2	0.50	0.00	-0.50	0.00	0.00
d3	0.50	-0.00	0.50	0.00	0.71
d4	0.50	-0.00	-0.50	-0.00	0.00
d5	-0.00	0.53	0.00	-0.85	0.00
d6	0.00	0.85	0.00	0.53	0.00

T

gnarly	0.41	0.00
wicked	0.41	0.00
awesome	0.82	-0.00
lame	0.00	0.85
terrible	0.00	0.53

$\times$

2.45	0.00
0.00	1.62

$=$

gnarly	1.00	0.00
wicked	1.00	0.00
awesome	2.00	0.00
lame	0.00	1.38
terrible	0.00	0.85

Distance from *gnarly*
1. gnarly
2. wicked
3. awesome
4. terrible
5. lame

Word embeddings based on neural language models

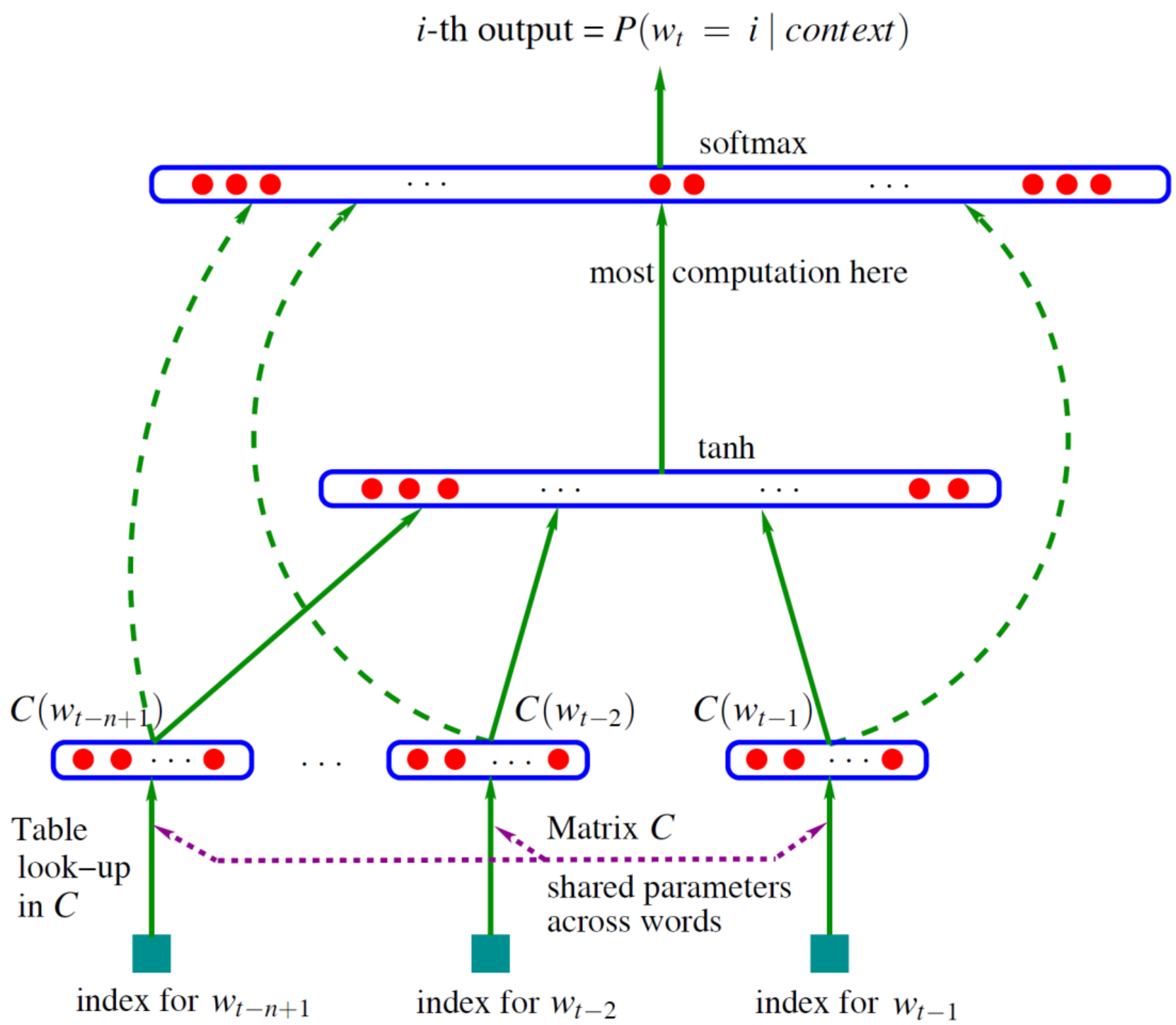
- So far: Distributional vector representations constructed based on **counts** (+ dimensionality reduction)

- Recent finding: Neural networks trained to **predict neighboring words** (i.e., language models) learn useful low-dimensional word vectors

    - Dimensionality reduction is built into the NN learning objective

    - Once the neural LM is trained on massive data, the word embeddings can be reused for other tasks

Word vectors as a byproduct of language modeling

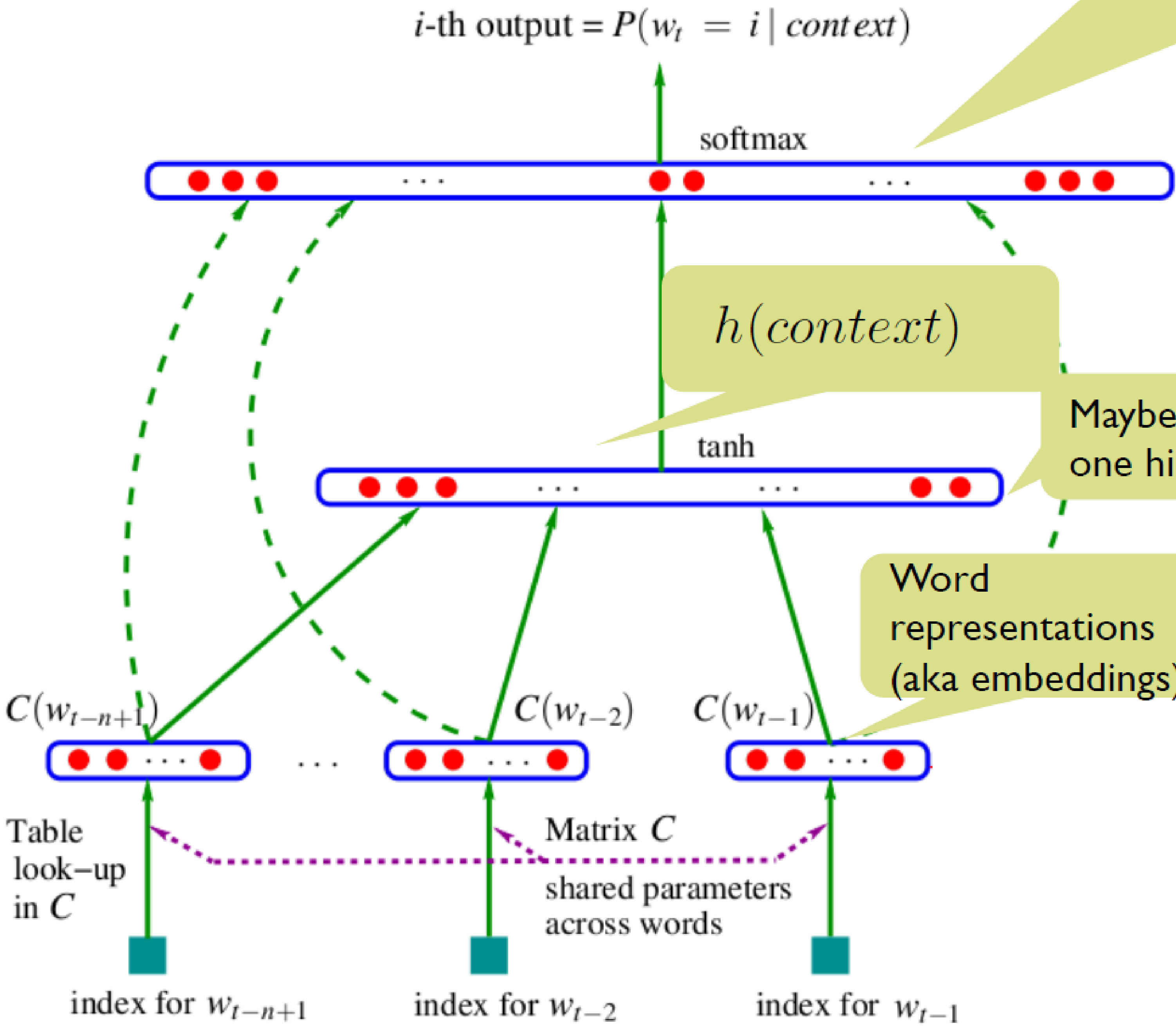


A neural probabilistic Language Model. Bengio et al. JMLR 2003

Language modeling task: context of w_t is

$w_{t-1}, w_{t-2}, \ldots, w_{t-n+1}$

$$P(w_t = i|context) = \frac{\exp(\hat{C}(i) \cdot h)}{\sum_{j=1}^{V} exp(\hat{C}(j) \cdot h)}$$



$h(context)$

Maybe more than one hidden layer

Word representations (aka embeddings)

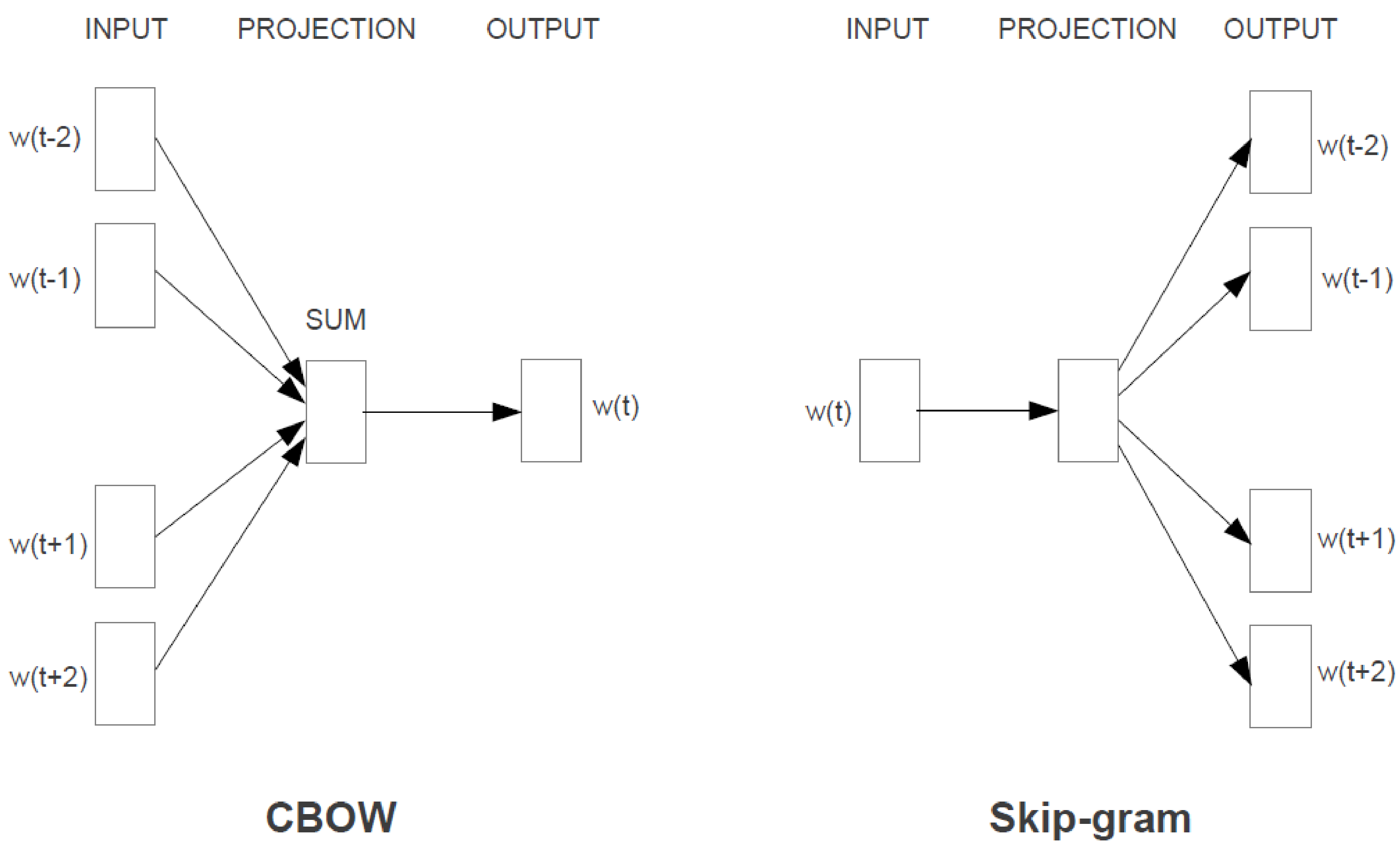
	1	2		d
cat	2.059	-1.134	...	2.004
dog	2.011	-1.005	...	0.135
...	...	...	...	...
January	-3.193	0.145	...	0.001
February	-3.016	0.196	...	0.025
...	...	...	...	...

Using neural word representations in NLP

- word representations from neural LMs
  - aka distributed word representations
  - aka word embeddings

- How would you use these word vectors?
- Turian et al. [2010]
  - word representations as features consistently improve performance of
    - Named-Entity Recognition
    - Text chunking tasks

Word2vec [Mikolov et al. 2013] introduces simpler models



https://code.google.com/p/word2vec

Word2vec claims

Useful representations for NLP applications

Can discover relations between words using vector arithmetic

king – male + female = queen

Paper+tool received lots of attention even outside the NLP research community

try it out at "word2vec playground":

http://deeplearner.fz-qqq.net/

Summary

- Given a large corpus, the meanings of words can be approximated in terms of words occurring nearby: **distributional context**. Each word represented as a **vector** of integer or real values.
  - Different ways to choose context, e.g. context windows
  - Different ways to count cooccurrence, e.g. (positive) **PMI**
  - Vectors can be **sparse** (1 dimension for every context) or **dense** (reduced dimensionality, e.g. with **Brown clustering** or **LSA**)
- This facilities measuring **similarity** between words—useful for many NLP tasks!
  - Different similarity measures, e.g. **cosine** (= normalized dot product)
  - Evaluations: human relatedness judgments; extrinsic tasks