Distributional Semantics

Jacob Andreas / MIT 6.864 / Spring 2020

Announcements

HW0 solutions: posted today

HW1: released today

reminder: upload a project writeup to canvas

Recap: text classification

Spam classification

RE: Staff and Faculty Mailbox Message !

Simona Matis <Simona.Matis@primariaclujnapoca.ro> Tue, Jan 14, 2020 at 9:20 AM
To: Simona Matis <Simona.Matis@primariaclujnapoca.ro>

Staff and Faculty Mailbox Message !

495MB **500**MB

This is to notify all Faculty Members and Staff on ox Quota Cleanup, If you are a staff or faculty member log on to yo CESS-PAGE to clean up mailbox.

Staff and Faculty Members mailbox quota size ACCESS-PAGE to complete.

Mailbox Sending/Receiving auth disabled at 500MB

ITS help desk
ADMIN TEAM

©Copyright 2020 Microsoft

spam

[Supercloud-users] Reminder: Downtime

Lauren E Milechin <la25321@mit.edu> Wed, Jan 15, 2020 at 4:51 PM
Reply-To: supercloud@mit.edu
To: supercloud-users <supercloud-users@mit.edu>

Hello All,

Supercloud will be having its regular monthly downtime this Thursday, Januar arting at midnight and ending about 24 hours later. We will send out an email when e is complete the system is ready for jobs. The system may not come back up as qu does, we are planning a few major changes. These changes should not cha ystem.

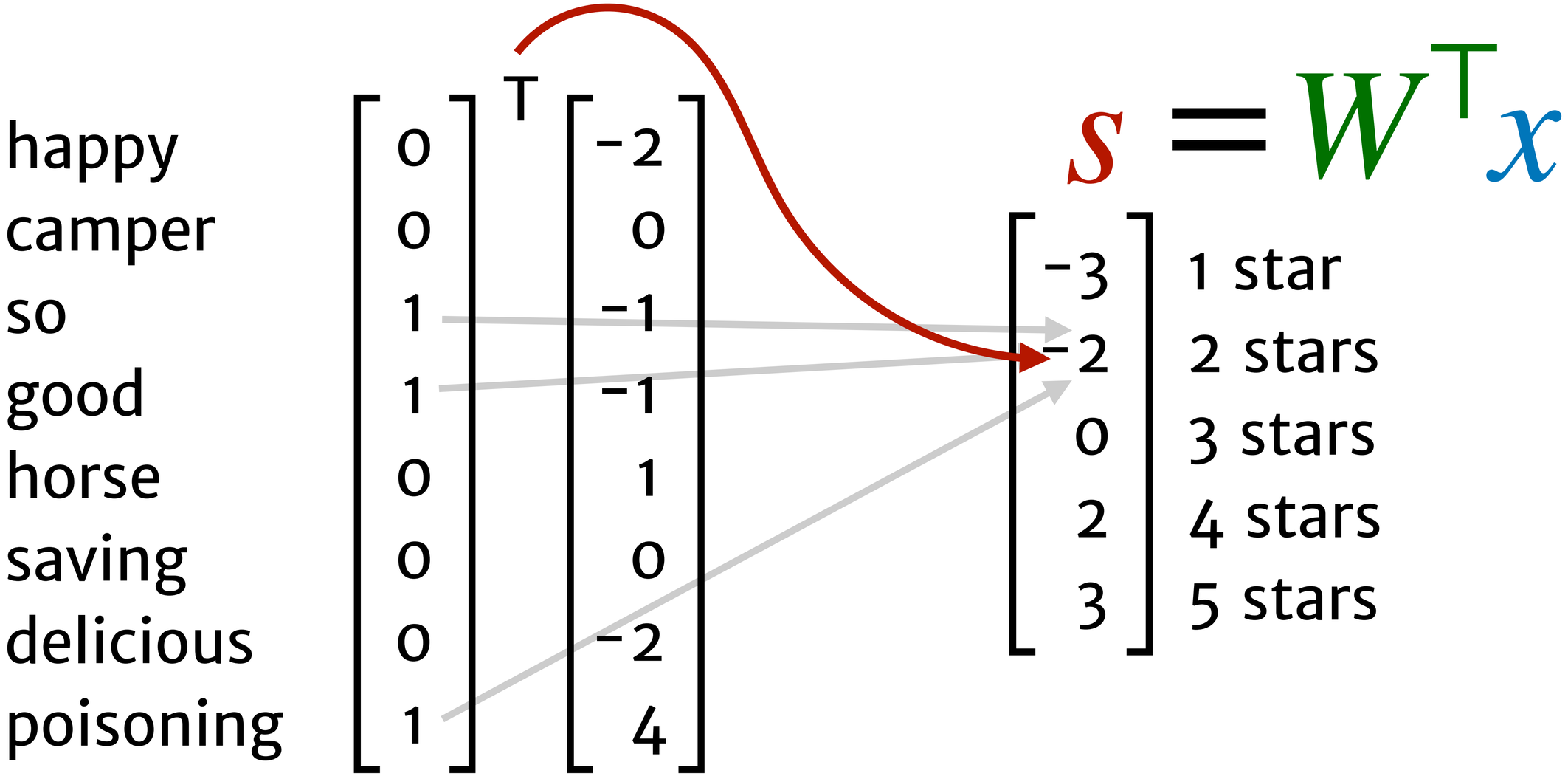
As a reminder, we have a downtime scheduled even third Thursday of the month. We will continue to send reminder emails

A general reminder: If you have any questions, p supercloud@mit.edu.

Lauren

not spam

Linear models



Interpretation: deep bag of words

$$s = W_2^\top f(W_1^\top x)$$

x

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

input

$f(W_1^\top x) = h_1$

$$\begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

“hidden layer”

$W_2^\top h_1 = s$

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

output

Interpretation: deep bag of words

$$s = W_2^\top f(W_1^\top x)$$

$w_{1,i}^\top$

$$\begin{bmatrix}0.1\\1.7\\0.3\end{bmatrix}\begin{bmatrix}0.1\\1.7\\0.3\end{bmatrix}\begin{bmatrix}0.1\\1.7\\0.3\end{bmatrix} \xrightarrow{f(\sum_{i:\ x_i=1} w_i)} \begin{bmatrix}1.2\\1.8\\-0.1\end{bmatrix} \longrightarrow 0.9$$

x *very good, but gave me*

Learning: likelihood

$$L(s, y) = -\log p(y \mid x)$$

$$= -\log \frac{\exp(s_y)}{\sum_i \exp(s_i)}$$

$$= -s_y + \log \sum_i \exp(s_i) := -\log \text{softmax}(s)_y$$

Idea: treat s as a vector of (unnormalized) log-probs, and maximize p(y | x; W).

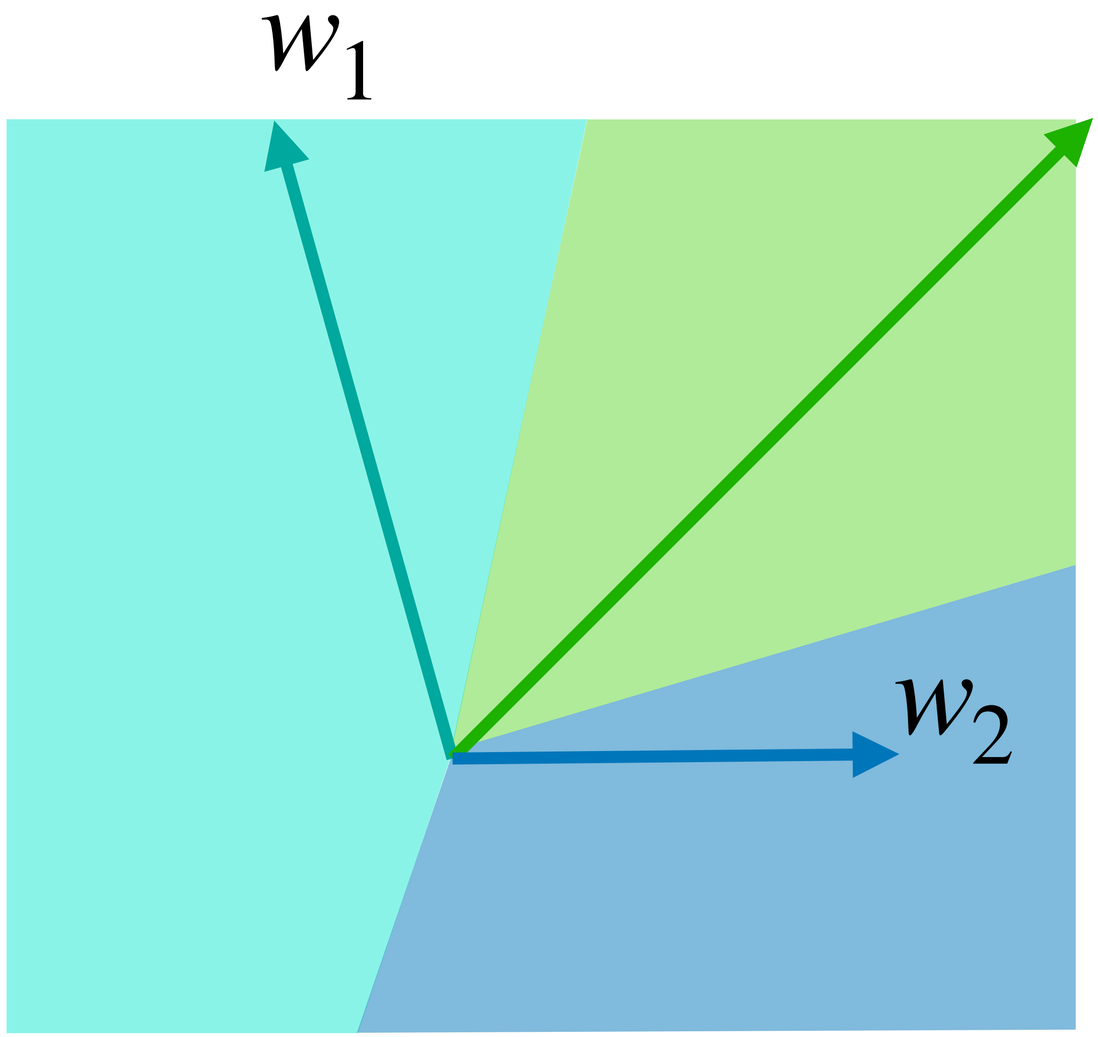
Learning: margin

$$L(s, y) = [s_y - \max(s_{-y}) - c]_+$$

$[x]_+ := \max(x, 0)$

Idea: try to make the score of the right label s_y at least at least c greater than the score of every wrong label.

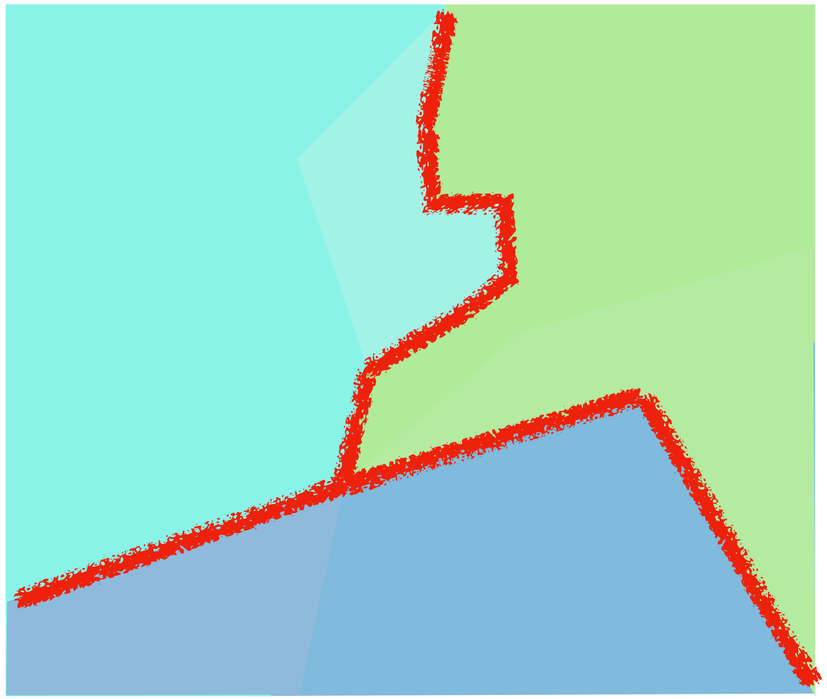
Linear decision boundaries



$$\hat{y} = \text{argmax}_i \; w_i^\top x$$

Multilayer perceptron

$$s = W_2^\top f(W_1^\top x)$$



Nonlinear decision boundaries!

Challenges: data sparsity[2]

$x =$ *perhaps not the most scintillating work in the director's oeuvre*

somewhat…awful	−1	
not…scintillating	0	???
less…terrible	1	

$$y = ?$$

Can we learn that similar words behave similarly in combination?

Text classification in the real world

In the real world

You are building tools that will affect people's lives!

Always ask:

Is this model reliable enough to deploy at all?

In what ways does my data reflect (or fail to reflect) the "real world?"

Will different groups experience disparate prediction qualities?

Example: bias

Feature weights from a restaurant review star classifier:

asian fusion	**-0.38**
chinese	**-0.21**
coffee	**0.30**
diner	**-0.45**
ethiopian	**0.10**
fast food	**-0.01**
french	**0.16**
greek	**0.56**

[Jurafsky et al., First Monday 2014]

Adversarial inputs

Harassment detection model:

you	**-0.1**
ugly	**2.3**
hate	**3.1**
kill	**5.1**
nasty	**1.2**
friends	**-0.2**

You're ugly and everyone hates you.

score: 10.7, label: possible harassment

You're ugly, everyone hates you, and you have no friends.

score: 10.3, label: possible harassment

You're ugly, everyone hates you, and you have no friends.
you you friends the and Monday happy good !!??

score: -6.1, label: no harassment

Challenges: data sparsity[2]

$x =$ *perhaps not the most scintillating work in the director's oeuvre*

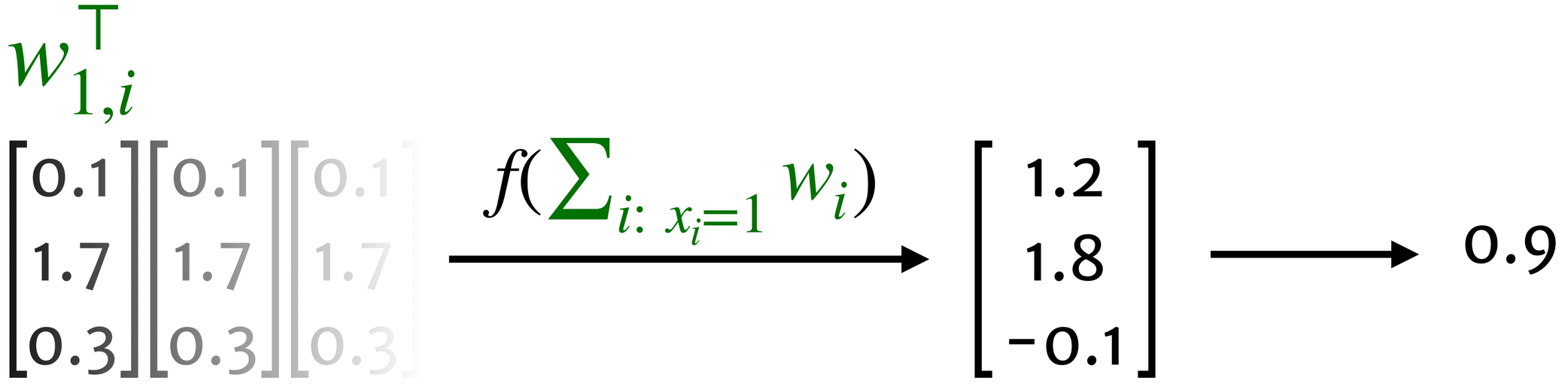
somewhat…awful	−1	
not…scintillating	0	???
less…terrible	1	

$$y = ?$$

Can we learn that similar words behave similarly in combination?

Interpretation: deep bag of words

More generally, can we learn portable representations of words independent of specific prediction problems?



x *very good, but gave me*

Distributional semantics

Words in context

…it ? its…

…has ? earnings…

…either ? or…

…which ? the…

What do we know about the word at “?”

not much about meaning, but probably a verb

Wider contexts

…but simple block ? superimposed on the…

…lawyers recently sent ? to growers saying…

…readers' comments in ? to the editor…

What do we know about the word at "?"
You send them, they come in a block variety, …

Unordered contexts

{the, May, since} ? {planning, said, agency}

{future, measure} ? {., performance}

{government's, primary} ? {gauge, forecasting}

What do we know about the word at "?"

Novel words

Lev stepped closer to the ?, which looked up at him.

The ?'s hand was warm, entirely handlike.

...the recent rental of a ?, one with potential as a weapon.

[Gibson 2014]

What’s a word?

Çekoslovakyalılaştıramadıklarımızdanmışsınız

(you are reportedly one of those who we were not able to turn into a Czechoslovakian)

The distributional hypothesis

"You shall know a word by the company it keeps."

J.R. Firth, *A Synopsis of Linguistic Theory*, 1957

How can we automate the process of constructing representations of word meaning from information about "company"?

Lexical semantics

Lexical semantics

How can we automate the process of constructing **representations of word meaning** from information about “company”?

What do we want from a representation of word meaning?

Types & syntactic roles

…it ? its…

…which ? the…

…the plate ? the table…

…will arrive ? Tuesday…

Is this word a noun?

A preposition?

What type of entity, event, or relation does it describe?

Selectional restrictions

Pat ate the ?.

The ? dripped down the sides of the bowl.

Pat caught the ?.

The ? smiled.

???

What sorts of actions can be performed on this word?

Is it animate?
Intelligent?
Solid?

Lexical relations

antonymy

good / bad, black / white, above / below

synonymy

fear / dread, greeting / welcome, rise / increase

hypernymy

cat / animal, engine / entity, run / move

Perceptual features and grounding

egg, *Switzerland*, *horse*
which is biggest?

good, *better*, *best*
gray, *black*
which is most intense?

tomato
which one is it?

[images: Wikipedia]

Summary

Our word representations should capture information about

- types
- constraints on predicate-argument relations
- other relationships (hypernyms, antonyms, meronyms)
- perceptual features

How much of this can we get from context alone?

Co-occurrence statistics

The term-document matrix

Representational idea: construct a matrix where

rows are words
columns are contexts
entries indicate how many times word *i* appears in context *j*

$$W_{td} = \begin{array}{c} \\ cat \\ dog \\ the \end{array} \begin{array}{c} \begin{array}{ccccccc} d_1 & d_2 & d_3 & d_4 & d_5 & d_6 & d_7 \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 & 1 & 1 & 0 \\ 20 & 13 & 18 & 22 & 15 & 4 & 20 \end{bmatrix} \end{array}$$

The term-document matrix

Representational idea: construct a matrix where

rows are words
columns are contexts
entries indicate how many times word *i* appears in context *j*

$$W_{td} = \begin{array}{r} \\ cat \\ dog \\ the \end{array} \begin{array}{c} d_1 \\ \left[\begin{array}{c} 1 \\ 0 \\ 20 \end{array}\right. \end{array}$$

The *mouse I saw yesterday was bigger than* ***the*** *biggest* ***cat*** *I've ever seen…*

Term-document matrix: rows as word representations

Related words appear together!

the cat lifted its paw

the dog raised its paw

a nylon dog collar

the paw-shaped tag on the cat's collar

$$W_{td} = \begin{array}{c} \\ cat \\ \\ paw \\ \\ \end{array} \begin{array}{c} \begin{array}{cccc} d_1 & d_2 & d_3 & \cdots & d_7 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & & 1 \\ \\ 1 & 0 & 1 & & 1 \\ \\ \end{bmatrix} \end{array}$$

Term-document matrix: rows as word representations

Related words **(sometimes)** appear together!

*the **cat** lifted its paw*

*the **dog** raised its paw*

*a nylon **dog** collar*

*the paw-shaped tag on the **cat**'s collar*

$$W_{td} = \begin{array}{c} \\ cat \\ dog \end{array} \begin{array}{c} \begin{array}{ccccc} d_1 & d_2 & d_3 & & d_7 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & & 1 \\ 0 & 1 & 1 & & 0 \end{bmatrix} \end{array}$$

Term-document matrix: rows as word representations

Related words **(sometimes)** appear together!

but document co-occurrence alone isn't a sufficient signal for semantic similarity.

words might be in strict alternation:

*the **cat** lifted its paw*

*the **dog** raised its paw*

*a nylon **dog** collar*

*the paw-shaped tag on the **cat**'s collar*

Term-document matrix: rows as word representations

Related words **(sometimes)** appear together!

but document co-occurrence alone isn't a sufficient signal for semantic similarity.

or co-occurrence statistics might be sparse:

	d_1	d_2	d_3	d_4	d_5	d_6	d_7
cat	1	1	0	1	0	1	0
scintillating	0	0	0	0	0	1	0

The word co-occurrence matrix

Solution 1: low-rank approximation

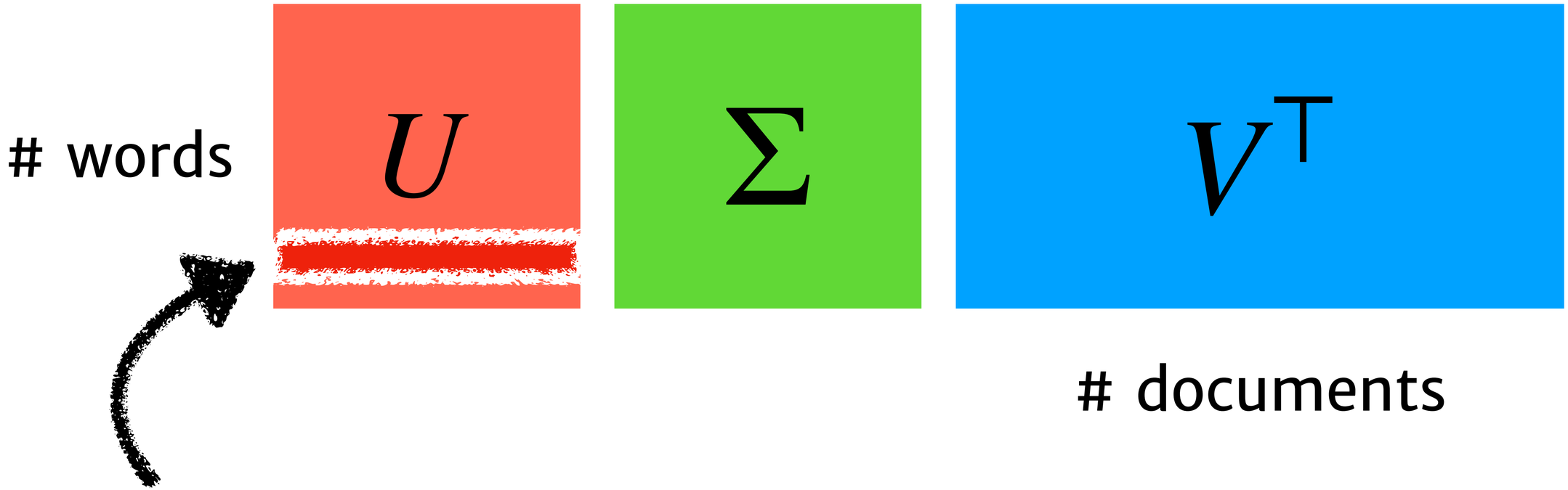
Theorem: for every $m \times n$ matrix A, there exists a factorization

$$A = U\Sigma V^\top$$

with U and V orthogonal and Σ diagonal.

Latent Semantic Analysis

Solution 1: low-rank approximation

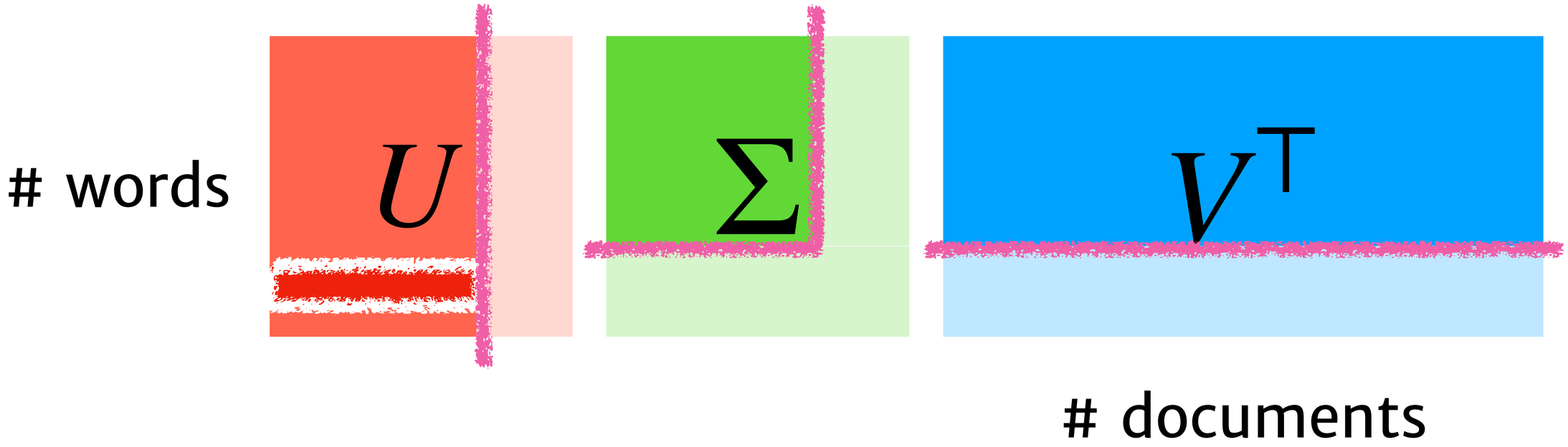


this is a word representation

vectors are all orthogonal!

Latent Semantic Analysis

Solution 1: low-rank approximation: truncate cols. of U and V



(**Theorem**: this is the best rank-k approx. to the original matrix)

Latent Semantic Analysis

documents about animals

documents about computers

	d_1	d_2	d_3	d_4	d_5	d_6	d_7
cat	1	0	1	0	0	0	0
paw	0	1	1	0	0	0	0
algorithm	0	0	0	1	1	1	1

Latent Semantic Analysis

$$U = \begin{bmatrix} .7 & 0 & -.7 \\ .7 & 0 & .7 \\ 0 & 1 & 0 \end{bmatrix} \begin{matrix} cat \\ \\ algo. \end{matrix}$$

useful word reps!

$$V = \begin{bmatrix} .4 & 0 & -.7 \\ & \vdots & \\ 0 & 1 & 0 \end{bmatrix} \begin{matrix} d_1 \\ \\ d_7 \end{matrix}$$

	d_1	d_2	d_3	d_4	d_5	d_6	d_7
cat	1	0	1	0	0	0	0
paw	0	1	1	0	0	0	0
algorithm	0	0	0	1	1	1	1

Latent Semantic Analysis: Intuition

$$U = \begin{bmatrix} .7 & 0 & -.7 \\ .7 & 0 & .7 \\ 0 & 1 & 0 \end{bmatrix} \begin{matrix} cat \\ \\ algo. \end{matrix} \qquad V = \begin{bmatrix} .4 & 0 & -.7 \\ & \vdots & \\ 0 & 1 & 0 \end{bmatrix} \begin{matrix} d_1 \\ \\ d_7 \end{matrix}$$

Dimensionality reduction techniques cluster words with similar contexts even when they don't always co-occur.

Frequency effects

These counts are way bigger!

vector space similarity thinks they're more important
dimensionality reduction cares more about them

$$W_{td} = \begin{array}{c} \\ cat \\ dog \\ the \end{array} \begin{array}{c} \begin{array}{ccccccc} d_1 & d_2 & d_3 & d_4 & d_5 & d_6 & d_7 \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 & 1 & 1 & 0 \\ 20 & 13 & 18 & 22 & 15 & 4 & 20 \end{bmatrix} \end{array}$$

TF-IDF normalization

term frequency (tf):
# of times word *w* appears in document *d*

inverse document frequency (idf):
log (# of documents / # of documents containing word *w*)

count'(w, d) = tf · idf

close to 0 if word *i* appears in almost every document

Pointwise mutual info. normalization

p(w) = # of times w appears in any document / word count

p(d) = fraction of documents identical to doc d

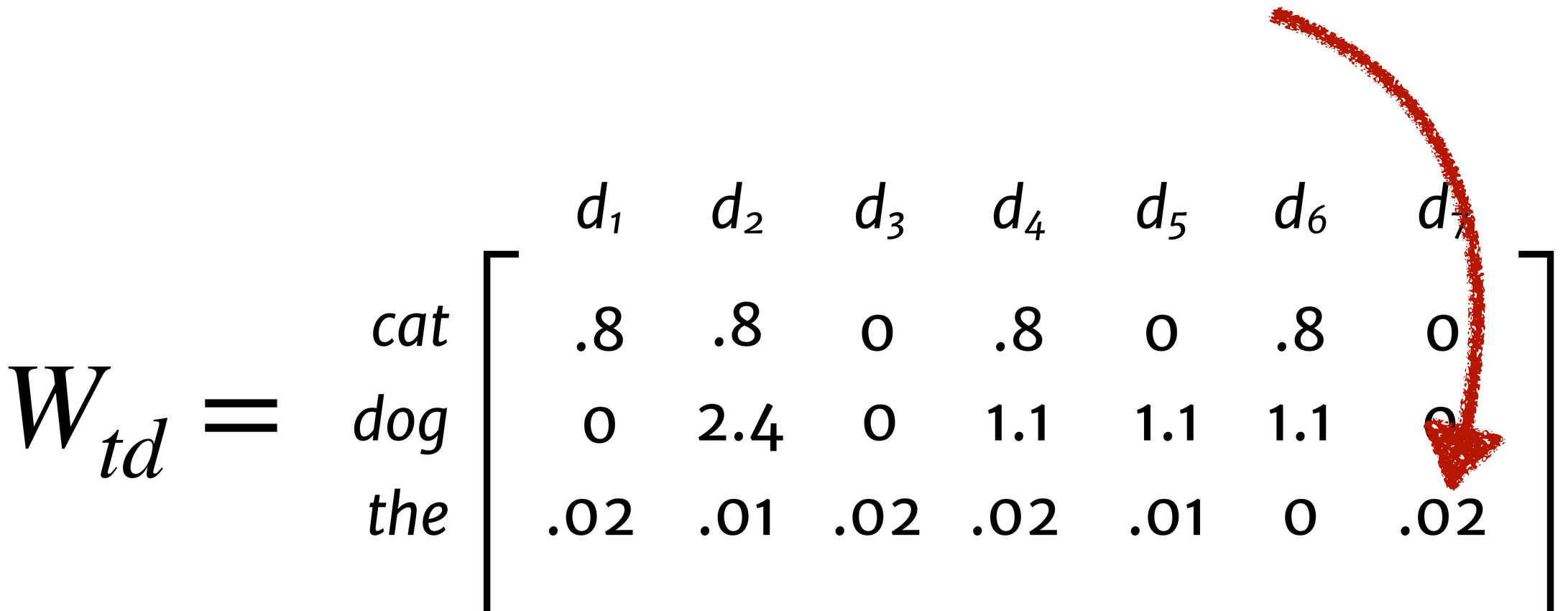
p(w, d) = # of times w and d appear together / (# words x # docs)

PMI(i, j) = p(w, d) / (p(w) p(d))

≈ p(d | w) if p(d) is roughly constant

Frequency effects

After weighting, these counts don't really matter



The word co-occurrence matrix

Solution 2: work in the word co-occurrence matrix

$$W_{tt} = \begin{array}{c} \\ cat \\ dog \\ the \end{array} \begin{array}{c} \begin{array}{ccc} cat & dog & the \end{array} \\ \begin{bmatrix} 10 & 8 & 103 \\ 8 & 20 & 97 \\ 103 & 97 & 995 \end{bmatrix} \end{array}$$

rows are words
columns are words
entries indicate how many times word *i* appears in the same context as word *j*

The word co-occurrence matrix

Notice:

# of times word *i* occurs in the same document as word *j*

$$= \sum_{\text{doc. } d} (\#\text{ of times } i \text{ occurs in } d) \times (\#\text{ of times } j \text{ occurs in } d)$$

$$= W_{td}[i, :] \; W_{td}[j, :]^\top$$

*i*th row of W_{td}

words that co-occur frequently have large row dot products in T-D matrix!

The word co-occurrence matrix

But: this matrix is still sparse at rare words.

Frequent words will continue to dominate similarity measurements.

$$W_{tt} = \begin{array}{c} \\ \textit{cat} \\ \textit{the} \\ \textit{brougham} \end{array} \begin{array}{c} \begin{array}{ccc} \textit{cat} & \textit{the} & \textit{brougham} \end{array} \\ \begin{bmatrix} 10 & \mathbf{98} & 0 \\ \mathbf{98} & 20 & \mathbf{1} \\ 0 & \mathbf{1} & 1 \end{bmatrix} \end{array}$$

The word co-occurrence matrix

But: this matrix is still sparse at rare words.

Frequent words will continue to dominate similarity measurements.

Still a good idea to do normalization (tf-idf / PMI) and rank reduction!

PMI revisited

p(i) = # of times *i* appears in any document / word count

p(i, j) = # of times *i* and *j* appear together / (# words)^2

PMI(i, j) = p(i, j) / (p(i) p(j))

~~= p(j | i)~~

Do I see these words together more often than if they were independent?

Summary: constructing distributional word vectors

1. Estimate a matrix of co-occurrence statistics

term-document matrix

$$W_{td} = \begin{array}{c} \\ cat \\ dog \\ the \end{array} \begin{array}{c} \begin{array}{ccccccc} d_1 & d_2 & d_3 & d_4 & d_5 & d_6 & d_7 \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 & 1 & 1 & 0 \\ 20 & 13 & 18 & 22 & 15 & 4 & 20 \end{bmatrix} \end{array}$$

Summary: constructing distributional word vectors

1. Estimate a matrix of co-occurrence statistics

word co-occurrence matrix

$$W_{tt} = \begin{array}{c} \\ cat \\ dog \\ the \end{array} \begin{array}{c} \begin{array}{ccc} cat & dog & the \end{array} \\ \begin{bmatrix} 10 & 8 & 103 \\ 8 & 20 & 97 \\ 103 & 97 & 995 \end{bmatrix} \end{array}$$

Summary: constructing distributional word vectors

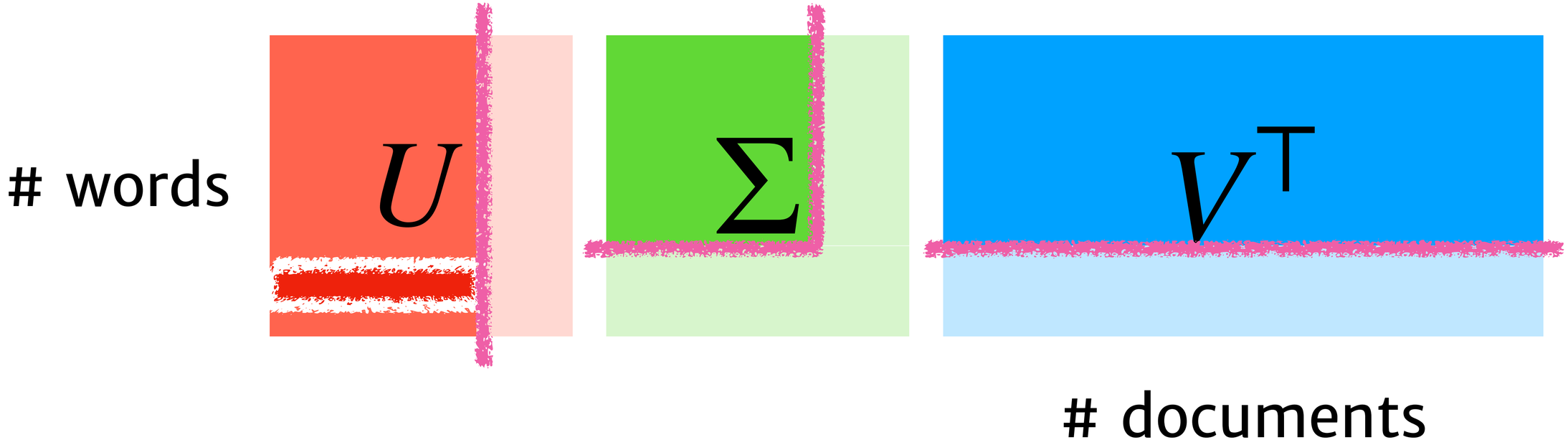
1. Estimate a matrix of co-occurrence statistics
2. Normalize / smooth counts

TF-IDF: # occurrences of word i
× log (# documents / # documents containing i)

PMI: p(word i, word j) / [p(word i) p(word j)]

Summary: constructing distributional word vectors

1. Estimate a matrix of co-occurrence statistics
2. Normalize / smooth counts
3. Take a low-rank approximation



Did we win?

Our word representations should capture information about

- types
- constraints on predicate-argument relations
- other relationships (hypernyms, antonyms, meronyms)
- perceptual features

Did we win?

Our word representations should capture information about

types

yes! if context information preserves ordering, words with similar types appear in similar contexts (verbs come after nouns etc.)

Did we win?

Our word representations should capture information about

types

constraints on predicate-argument relations

yes! dot products between word vectors predict frequency of co-occurrence

Did we win?

Our word representations should capture information about

- types
- constraints on predicate-argument relations
- other relationships (hypernyms, antonyms, meronyms)
- maybe? e.g. antonyms have the same type but appear in disjoint contexts

Did we win?

Our word representations should capture information about

types

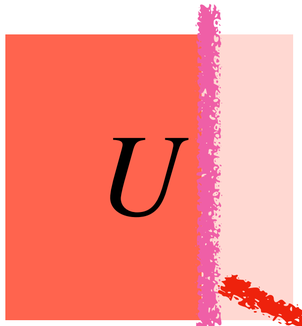
constraints on predicate-argument relations

other relationships (hypernyms, antonyms, meronyms)

perceptual features

??? *plant* is close to *green*, but what does *green* look like?

In models



$$s = W_2^\top f(W_1^\top x)$$

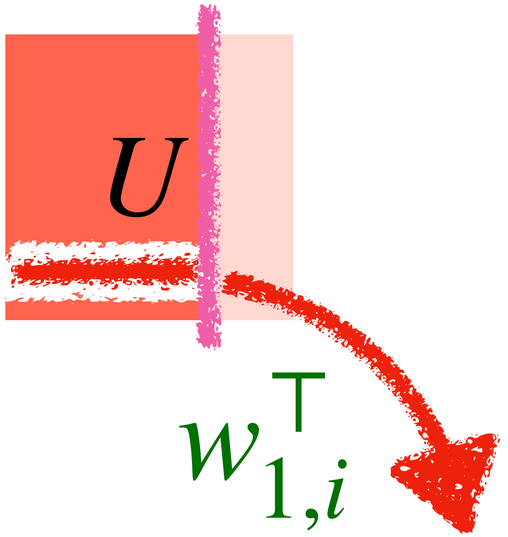
x $\quad f(W_1^\top x) = h_1 \quad$ $W_2^\top h_1 = s$

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$ input

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$ “hidden layer”

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$ output

In models



$$s = W_2^\top f(W_1^\top x)$$

$$\begin{bmatrix}0.1\\1.7\\0.3\end{bmatrix}\begin{bmatrix}0.1\\1.7\\0.3\end{bmatrix}\begin{bmatrix}0.1\\1.7\\0.3\end{bmatrix} \xrightarrow{f(\sum_{i:\ x_i=1} w_i)} \begin{bmatrix}1.2\\1.8\\-0.1\end{bmatrix} \longrightarrow 0.9$$

x *very good, but gave me*

Bias in distributional representations

	man	*woman*	*doctor*	*nurse*
man		100	45	23
woman	100		18	48
doctor	45	18		80
nurse	23	48	80	

It's usually most helpful to think of bias as a property of decisions, not parameters.

But how will these parameters influence decisions?

Next class: more word embeddings