ANTONIS ANASTASOPOULOS
CS499 INTRODUCTION TO NLP

VECTOR SEMANTICS

https://cs.gmu.edu/~antonis/course/cs499-spring21/

With adapted slides by David Mortensen and Alan Black

HOMEWORK 1

Note: Use a cost of **1** for substitutions in the base edit-distance implementation

Other questions?

STRUCTURE OF THIS LECTURE

1. Why vectors?
2. Words and co-occurrence vectors
3. PPMI
3. Neural Embeddings

WHY VECTOR MODELS?

COMPUTING THE SIMILARITY BETWEEN WORDS

"**fast**" is similar to "**rapid**"
"**tall**" is similar to "**height**"

Question: "How **tall** is Mt. Everest?"
Potential Answer: "The official **height** of Mount Everest is 29029 feet."

SIMILARITY FOR PLAGIARISM DETECTION

MAINFRAMES

Mainframes are primarily referred to large computers with rapid, advanced processing capabilities that can execute and perform tasks equivalent to many Personal Computers (PCs) machines networked together. It is characterized with high quantity Random Access Memory (RAM), very large secondary storage devices, and high-speed processors to cater for the needs of the computers under its service.

Consisting of advanced components, mainframes have the capability of running multiple large applications required by many and most enterprises and organizations. This is one of its advantages. Mainframes are also suitable to cater for those applications (programs) or files that are of very high

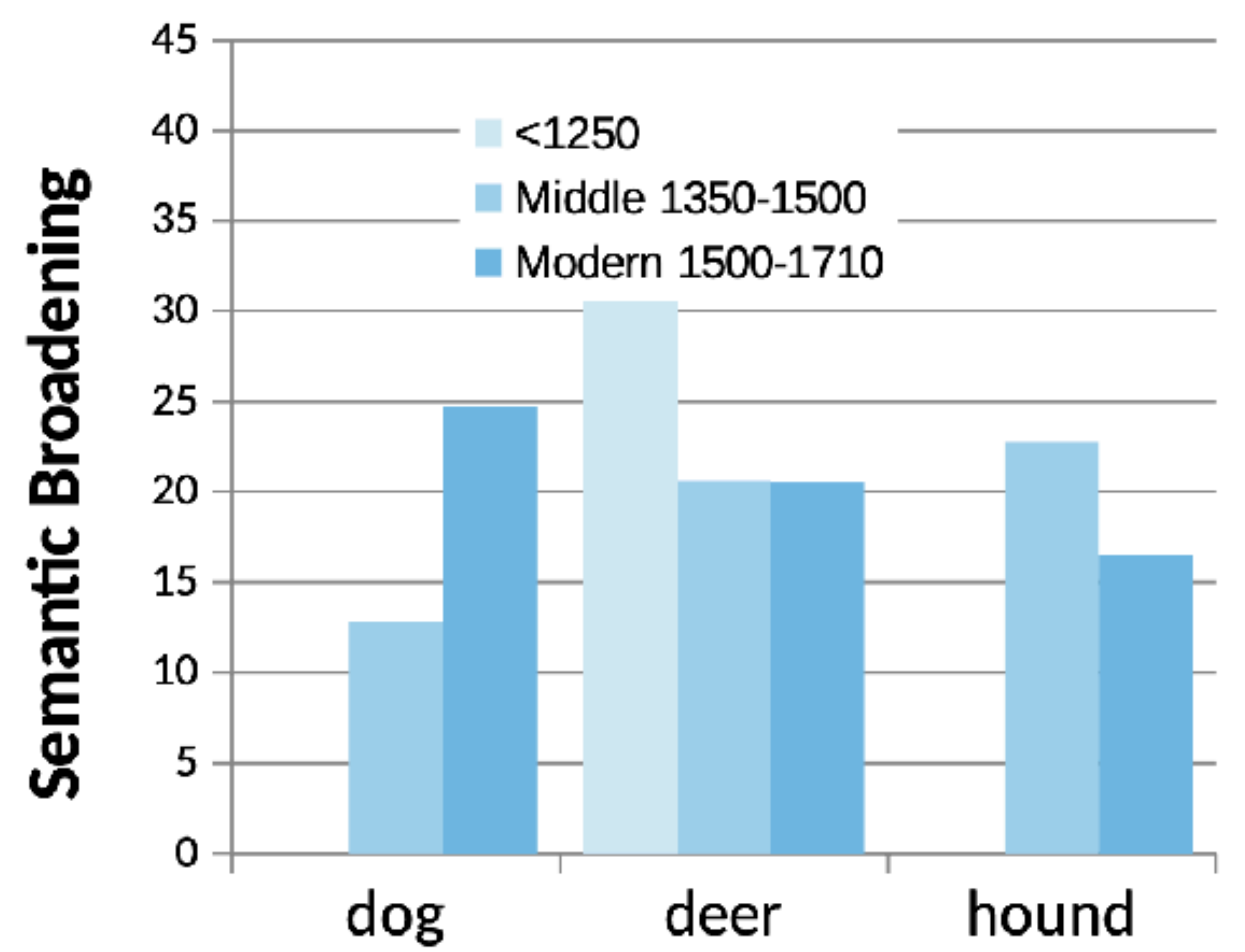
MAINFRAMES

Mainframes usually are referred those computers with fast, advanced processing capabilities that could perform by itself tasks that may require a lot of Personal Computers (PC) Machines. Usually mainframes would have lots of RAMs, very large secondary storage devices, and very fast processors to cater for the needs of those computers under its service.

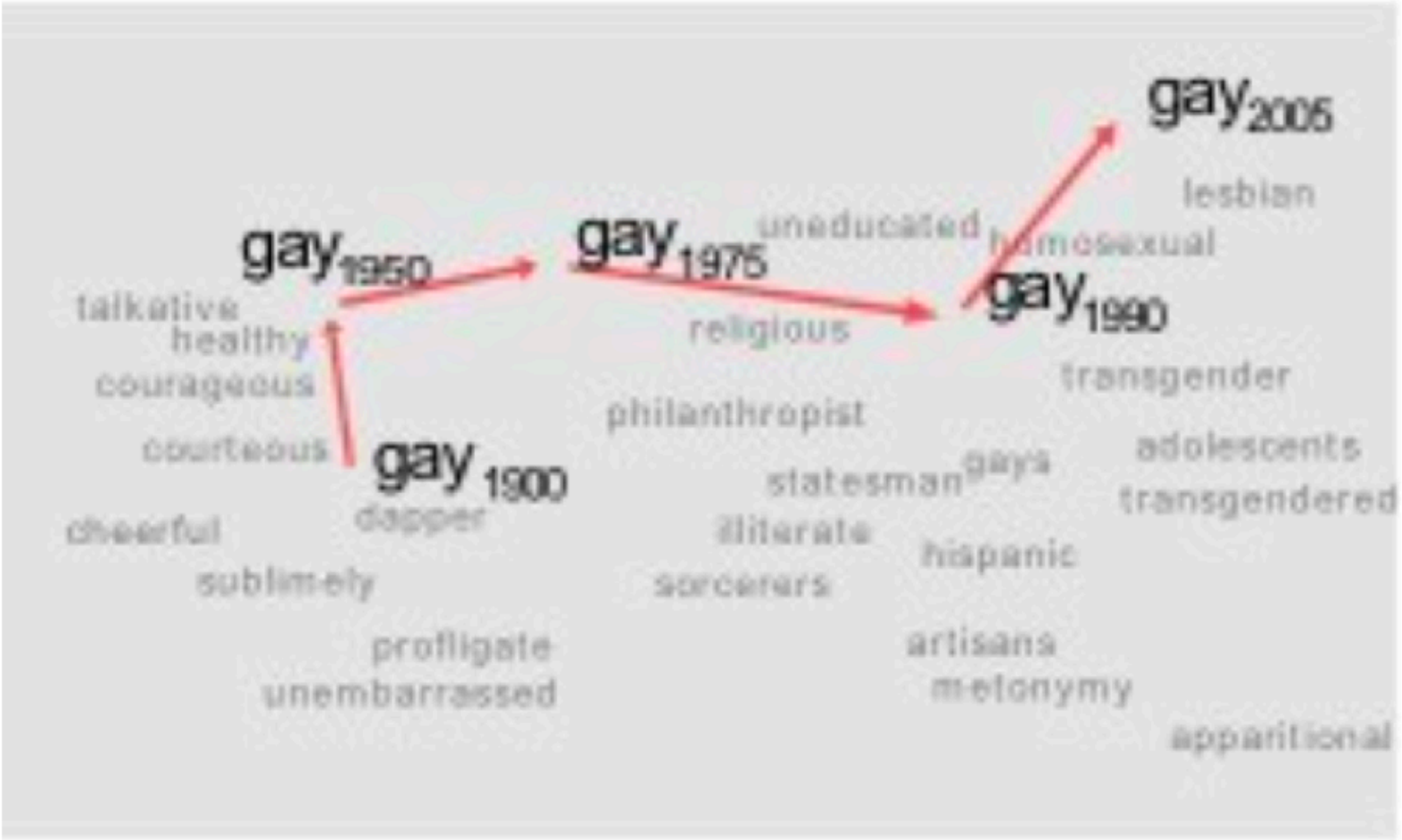
Due to the advanced components mainframes have, these computers have the capability of running multiple large applications required by most enterprises, which is one of its advantage. Mainframes are also suitable to cater for those applications or files that are of very large demand

(DIACHRONIC) SEMANTIC CHANGE OF WORDS

Sagi, Kaufmann Clark 2013



Kulkarni, Al-Rfou, Perozzi, Skiena 2015



PROBLEMS WITH THESAURUS-BASED MEANING

We don’t have a thesaurus for every language

We can’t have a thesaurus for every year

(For historical linguistics, we need to compare word meanings from year t to $t+1$

Thesauri have problems with **recall**

Many words/phrases might be missing

They work less well for verbs, adjectives

VECTOR SEMANTICS

Vector semantics == vector-space models of meaning
== distributional models of meaning

Intuition:

"Oculist and eye-doctor […] occur in almost the same environments"
"If A and B have almost identical environments, we say that they are synonyms."

Zellig Harris (1954)

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

Firth (1957)

INTUITION OF DISTRIBUTIONAL WORD SIMILARITY

Nida example — what is a **tesgüino**?

A bottle of ***tesgüino*** *is on the table*
Everybody likes ***tesgüino***
Tesgüino *makes you drunk*
We make ***tesgüino*** *out of corn*

From context, you guessed what tesgüino means (it's like beer)

Intuition: two words are similar if they have similar word contexts!

THE FOUR KINDS OF VECTOR MODELS

Sparse vector representations:

1. Mutual Information weighted co-occurance matrices

Dense vector representations:

2. Brown clusters
3. Neural network based embeddings

Shared intuition: “embed” the word in a vector space to model its meaning.

WORDS AND CO-OCCURRENCE MATRICES

CO-OCCURRENCE MATRICES

Represent how often a word occurs in a document:

- **term-document matrix**

Or how often a word occurs with another:

- **term-term matrix**
  (or word-word co-occurrence or word-context matrix)

TERM-DOCUMENT MATRIX

Each cell: count of word w in document d

	As you Like it	Twelfth Night	Julius Cesar	Henry V
battle	1	1	8	15
soldier	2	2	12	36
fool	37	58	1	5
clown	7	117	0	0

Document vector

DOCUMENT SIMILARITY

Two documents are similar if their vectors are similar

	As you Like it	Twelfth Night	Julius Cesar	Henry V
battle	1	1	8	15
soldier	2	2	12	36
fool	37	58	1	5
clown	7	117	0	0

TERM-DOCUMENT MATRIX

Each cell: count of word w in document d

	As you Like it	Twelfth Night	Julius Cesar	Henry V	
battle	1	1	8	15	
soldier	2	2	12	36	word vector
fool	37	58	1	5	
clown	7	117	0	0	

TERM-DOCUMENT MATRIX

Two words are similar if their vectors are similar

	As you Like it	Twelfth Night	Julius Cesar	Henry V
battle	1	1	8	15
soldier	2	2	12	36
fool	37	58	1	5
clown	7	117	0	0

THE WORD-WORD OR WORD-CONTEXT MATRIX

Instead of entire documents, we will use smaller contexts

e.g. paragraph, or a fixed window of n words

A word is defined by a vector over counts of context words

Instead of vector of length D, we have vector of length $|V|$.

The word-word matrix is of size $|V| \times |V|$.

WORD-CONTEXT MATRIX EXAMPLE

sugar, a sliced lemon, a tablespoonful of	**apricot**	preserve or jam, a pinch each of,
their enjoyment. Cautiously she sampled her first	**pineapple**	and another fruit whose taste she likened
well suited to programming on the digital	**computer**.	In finding the optimal R-stage policy from
for the purpose of gathering data and	**information**	necessary for the study authorized in the

	aardvark	computer	data	pinch	result	sugar	…
apricot	0	0	0	1	0	1	
pineapple	0	0	0	1	0	1	
digital	0	2	1	0	1	0	
information	0	1	6	0	4	0	
…							

A 50,000x50,000 will be **very** sparse (most values are 0)

Short (1-3) window → syntacticity.

Long (4-10) window → semanticity

TWO TYPES OF CO-OCCURRENCY

First-order co-occurrency (**syntagmatic association**):

Two words are typically nearby each other
wrote is a first-order associate of *book* or *poem*

Second order co-occurrency (**paradigmatic association**):

Two words have similar neighbors
wrote is a second-order associate of *said* or *remarked*

POSITIVE POINT-WISE MUTUAL INFORMATION (PPMI)

PROBLEM WITH RAW COUNTS

Raw frequency is not a great measure of association between words

It is **very** skewed
e.g. “the” and “of” are very frequent, but they are not very discriminative

We’d rather have a measure that asks whether a context word is **particularly informative** about the target word.

Positive Point-wise Mutual Information (PPMI)

POINT-WISE MUTUAL INFORMATION

Point-wise Mutual Information:

"Do events x and y co-occur more than if they were independent?"

$$\text{PMI}(X, Y) = \log_2 \frac{P(x, y)}{P(x)P(y)}$$

PMI between words (Church & Hanks, 1989)

"Do words x and y co-occur more than if they were independent?"

POSITIVE POINT-WISE MUTUAL INFORMATION

PMI ranges in $(-\infty,+\infty)$

What do we do with negative values though?

(not very useful)

So, we replace negative values with 0:

$$\text{PPMI}(w_1, w_2) = max(0, \log_2 \frac{P(w_1, w_2)}{P(w_1)P(w_2)})$$

EXAMPLE

$N = 19$

$p(w = \text{information}, c = \text{data}) = \frac{6}{19} = .32$

$p(w = \text{information}) = \frac{11}{19} = .58$

$p(c = \text{data}) = \frac{7}{19} = .37$

Count(w,c)	computer	data	pinch	result	sugar
apricot	0	0	1	0	1
pineapple	0	0	1	0	1
digital	2	1	0	1	0
information	1	6	0	4	0

p(w,c)	computer	data	pinch	result	sugar
apricot	0	0	0.05	0	0.05
pineapple	0	0	0.05	0	0.05
digital	0.11	0.05	0	0.05	0
information	0.05	0.32	0	0.21	0

	p(w)
apricot	0.11
pineapple	0.11
digital	0.21
information	0.58

	computer	data	pinch	result	sugar
p(c)	0.16	0.37	0.11	0.26	0.11

EXAMPLE

$$pmi_{ij} = \log_2 \frac{p_{ij}}{p_{i*}p_{*j}}$$

$$pmi(\text{information, data}) = \log_2(\frac{.32}{.37 * .58}) = .57$$

p(w,c)	computer	data	pinch	result	sugar
apricot	0	0	0.05	0	0.05
pineapple	0	0	0.05	0	0.05
digital	0.11	0.05	0	0.05	0
information	0.05	0.32	0	0.21	0

	p(w
apricot	0.11
pineapple	0.11
digital	0.21
information	0.58

	computer	data	pinch	result	sugar
p(c)	0.16	0.37	0.11	0.26	0.11

PPMI(w,c)	computer	data	pinch	result	sugar
apricot	-	-	2.25	-	2.25
pineapple	-	-	2.25	-	2.25
digital	1.66	0.00	-	0.00	-
information	0.00	0.57	-	0.47	-

Weighting PMI

PMI is biased toward infrequent events

(Very rare words have very high PMI values)

Solution: Use add-k smoothing

PPMI COMPUTATION WITH LAPLACE SMOOTHING

Count(w,c)	computer	data	pinch	result	sugar
apricot	0	0	1	0	1
pineapple	0	0	1	0	1
digital	2	1	0	1	0
information	1	6	0	4	0

Count(w,c)	computer	data	pinch	result	sugar
apricot	2	2	3	2	3
pineapple	2	2	3	2	3
digital	4	3	2	3	2
information	3	8	2	6	2

p(w,c)	computer	data	pinch	result	sugar
apricot	0	0	0.05	0	0.05
pineapple	0	0	0.05	0	0.05
digital	0.11	0.05	0	0.05	0
information	0.05	0.32	0	0.21	0

p(w,c)	computer	data	pinch	result	sugar
apricot	0.03	0.03	0.05	0.03	0.05
pineapple	0.03	0.03	0.05	0.03	0.05
digital	0.07	0.05	0.03	0.05	0.03
information	0.05	0.14	0.03	0.10	0.03

PPMI(w,c)	computer	data	pinch	result	sugar
apricot	-	-	**2.25**	-	**2.25**
pineapple	-	-	**2.25**	-	**2.25**
digital	**1.66**	0.00	-	**0.00**	-
information	0.00	**0.57**	-	**0.47**	-

PPMI(w,c)	computer	data	pinch	result	sugar
apricot	0.00	0.00	**0.56**	0.00	**0.56**
pineapple	0.00	0.00	**0.56**	0.00	**0.56**
digital	**0.62**	0.00	0.00	0.00	0.00
information	0.00	**0.58**	0.00	**0.37**	0.00

MEASURING EMBEDDING SIMILARITY

MEASURING SIMILARITY

Given 2 words v and w, we need a way to measure their similarity

Most measure of vectors similarity are based on the **dot product** (often called **inner product**):

$$\vec{v} \cdot \vec{w} = \sum_{i=1}^{N} v_i w_i = v_1 w_1 + v_2 w_2 + \ldots + v_N w_N \,.$$

High when two vectors have large values in the same dimensions

Low (in fact 0) for **orthogonal vectors** with zeros in complementary distribution

PROBLEM WITH DOT PRODUCT

$$\vec{v} \cdot \vec{w} = \sum_{i=1}^{N} v_i w_i = v_1 w_1 + v_2 w_2 + \ldots + v_N w_N .$$

Dot product gets larger as the vectors get more dimensions and as the vector length increases.

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

Vectors are longer if they have higher values in each dimension:

- more frequent words will have higher dot products
- that's bad: we don't want a similarity metric to be sensitive to word frequency

SOLUTION: COSINE DISTANCE

Solution: just divide the dot product by the length of the two vectors!

$$\frac{\overrightarrow{a} \cdot \overrightarrow{b}}{|\overrightarrow{a}||\overrightarrow{b}|}$$

This turns out to be the cosine of the angle between them:

$$\overrightarrow{a} \cdot \overrightarrow{b} = |\overrightarrow{a}||\overrightarrow{b}|\cos\theta$$

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

IS COSINE DISTANCE MEANINGFUL?

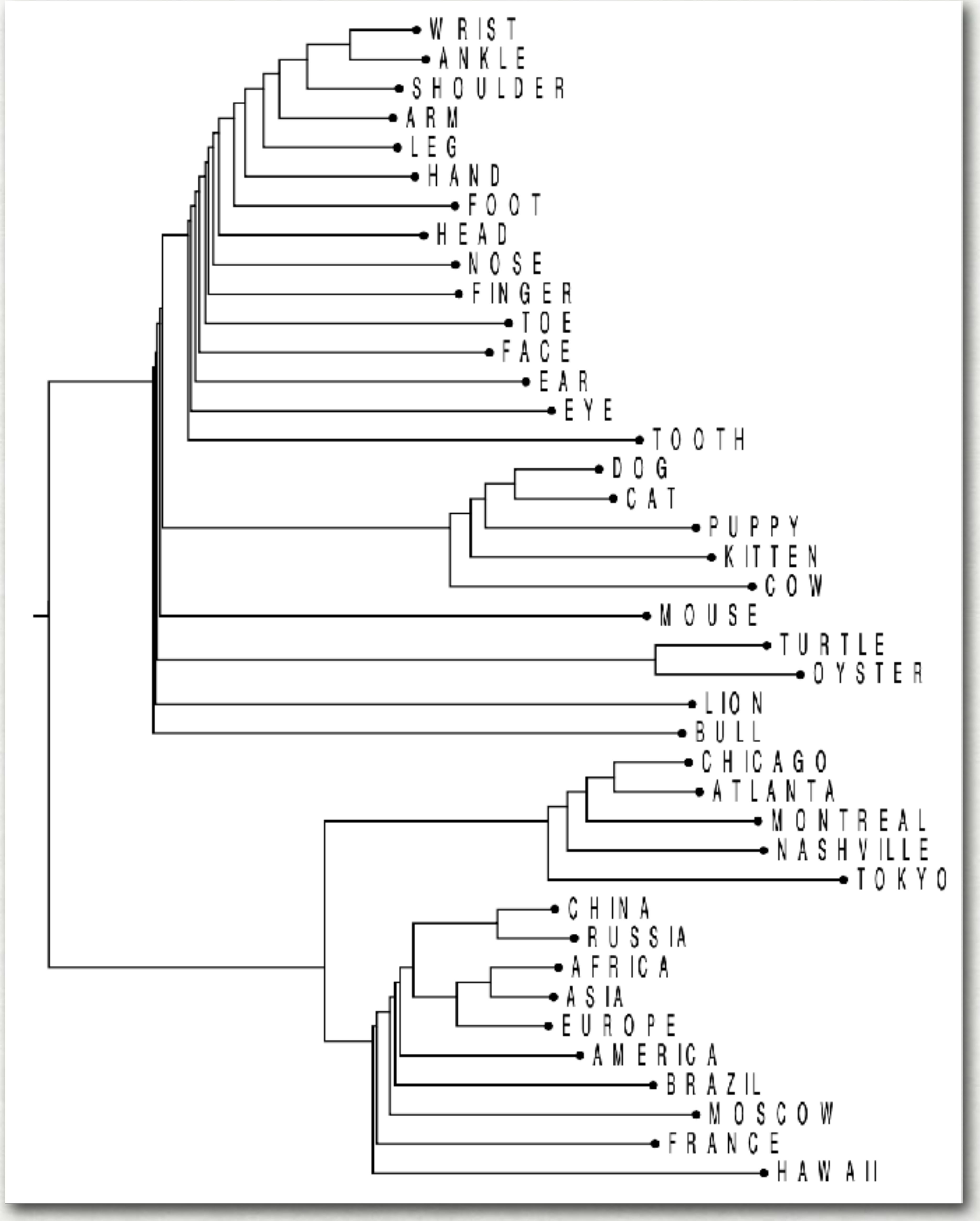
Yes! We can cluster the vectors based on their cosine distance to visualize the similarity

Other possible similarity measures:

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

$$\text{sim}_{\text{Jaccard}}(\vec{v}, \vec{w}) = \frac{\sum_{i=1}^{N} \min(v_i, w_i)}{\sum_{i=1}^{N} \max(v_i, w_i)}$$

$$\text{sim}_{\text{Dice}}(\vec{v}, \vec{w}) = \frac{2 \times \sum_{i=1}^{N} \min(v_i, w_i)}{\sum_{i=1}^{N} v_i + w_i}$$



GOING BEYOND CO-OCCURRENCES

> "The meaning of entities, and the meaning of grammatical relations among them, is related to the restriction of combinations of these entities relative to other entities"
> — Zellig Harris (1968)

Two words are similar if they have similar **syntactic** contexts.

e.g. *duty* and *responsibility* not only have similar words that appear in their contexts, but they also have similar syntactic distributions:

Modified by adjectives	Additional, administrative, assumed, collective, congressional, constitutional…
Objects of verbs	Assert, assign, assume, attend to, avoid, become, breach…

We can create syntactic features too, and add them to our count tables, and follow the same processes as before