Vector Semantics & Embeddings

Word Meaning

https://youtu.be/NGQtmnSOv40?t=1333

https://connecting-wall.netlify.app/

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!

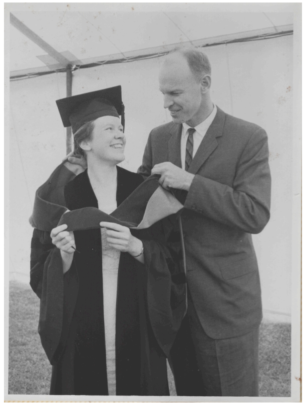
Formal semantics:

- The meaning of "dog" is DOG; cat is CAT

$$\forall x\ DOG(x) \longrightarrow MAMMAL(x)$$

Old linguistics joke by Barbara Partee:

- Q: What's the meaning of life?
- A: LIFE



key founder of formal semantics

one of the greatest living linguists

Desiderata

What should a theory of word meaning do for us?

What words are similar?

What words have opposite meanings?

What words are related?

What words show up where?

Lemmas and senses

lemma

mouse (N)

sense

1. any of numerous small rodents...
2. a hand-operated device that controls a cursor...

Modified from the online thesaurus WordNet

A sense or "concept" is the meaning component of a word

Lemmas can be polysemous (have multiple senses)

Relations between senses: Synonymy

Synonyms have the same meaning in some or all contexts.

- filbert / hazelnut
- couch / sofa
- big / large
- automobile / car
- vomit / throw up
- water / H_2O

connotation = "style"

denotation = meaning

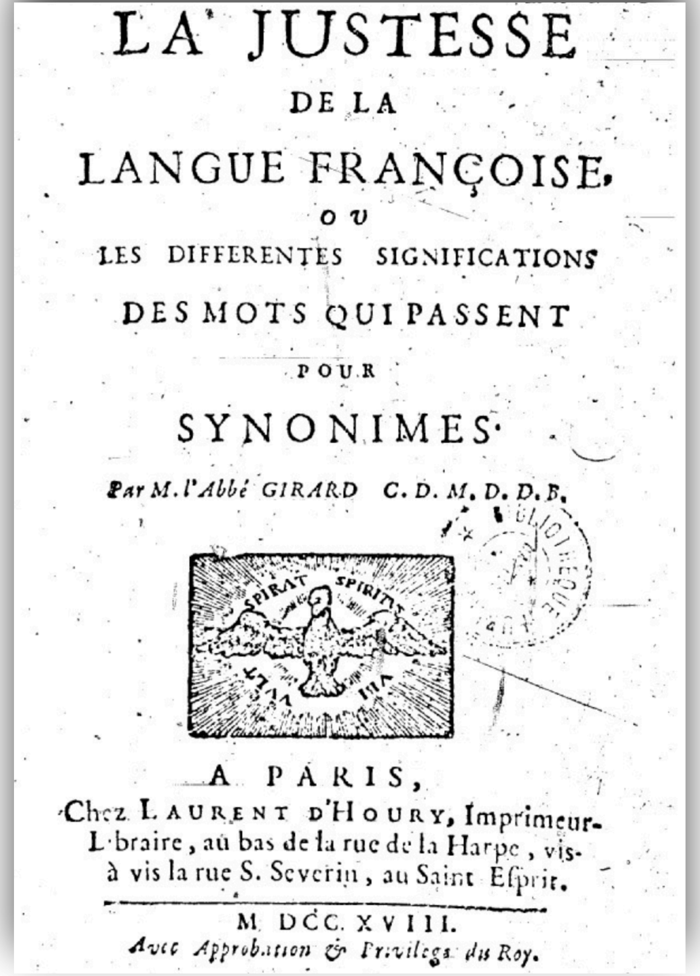
The Linguistic Principle of Contrast:

Difference in form → difference in meaning

Abbé Gabriel Girard (1718):

"je ne crois pas qu'il y ait de-
mot ſynonime dans aucune
Langue. Je le dis par con-"

[I do not believe that there is a synonymous word in any language]


LA JUSTESSE
DE LA
LANGUE FRANÇOISE,
OU
LES DIFFERENTES SIGNIFICATIONS
DES MOTS QUI PASSENT
POUR
SYNONIMES.
Par M. l'Abbé GIRARD C. D. M. D. D. B.

A PARIS,
Chez LAURENT D'HOURY, Imprimeur-Libraire, au bas de la rue de la Harpe, vis-à vis la rue S. Severin, au Saint Esprit.
M. DCC. XVIII.
Avec Approbation & Privilege du Roy.

Thanks to Mark Aronoff!

Relation: Synonymy?

water/H_20

"H_20" in a surfing guide?

big/large

my big sister != my large sister

Relation: Similarity

Words with similar meanings.

Not synonyms, but sharing some element of meaning:

`car, bicycle`

`cow, horse`

Ask humans how similar 2 words are

word1	word2	similarity
vanish	disappear	10 8.5 9
behave	obey	7 6 5
belief	impression	4 4 5
muscle	bone	3 6 5
modest	flexible	0 5 2
hole	agreement	2 0 0.5

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

Relation: Word relatedness

Also called "word association"

Words can be related in any way, perhaps via a semantic frame or field

- `coffee, tea`: **similar**
- `movie, popcorn`: **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

Connotation

Osgood et al. (1957)

We usually consider 3 affective dimensions:

- **valence**: the pleasantness of the stimulus
- **arousal**: the intensity of emotion provoked by the stimulus
- **dominance**: the degree of control exerted by the stimulus

	Word	Score		Word	Score
Valence	love	1.000		toxic	0.008
	happy	1.000		nightmare	0.005
Arousal	elated	0.960		mellow	0.069
	frenzy	0.965		napping	0.046
Dominance	powerful	0.991		weak	0.045
	leadership	0.983		empty	0.081

Values from NRC VAD Lexicon (Mohammad 2018)

Desiderata

Concepts or word senses

- Have a complex many-to-many association with **words** (homonymy, multiple senses)

Have relations with each other

- Synonymy
- Antonymy
- Similarity
- Relatedness
- Connotation

Vector Semantics & Embeddings

Vector Semantics

Computational models of word meaning

Can we build representions of word meanings?

Most common approach: **vector semantics**

marmot: [0.5, 1, 5, -1, -2.5, 4, 10, 15]

Ludwig Wittgenstein

PI #43:

"The meaning of a word is its use in the language"

Let's define words by their usages

One way to define "usage":

words are defined by their environments (the words around them)

Zellig Harris (1954):

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

What does recent English borrowing *ongchoi* mean?

Suppose you see these sentences:

- Ong choi is delicious **sautéed with garlic**.
- Ong choi is superb **over rice**
- Ong choi **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

Conclusion:

- Ongchoi is a leafy green like spinach, chard, or collard greens
  - We could conclude this based on words like "leaves" and "delicious" and "sauteed"

Ongchoi: *Ipomoea aquatica "Water Spinach"*

空心菜
kangkong
rau muống
…

Yamaguchi, Wikimedia Commons, public domain

Idea 1: Defining meaning by linguistic distribution

Let's define the meaning of a word by its distribution in language use, meaning its neighboring words or grammatical environments.

Idea 2: Meaning as a point in space (Osgood et al. 1957)

3 affective dimensions for a word

- **valence**: pleasantness
- **arousal**: intensity of emotion
- **dominance**: the degree of control exerted

	Word	Score		Word	Score
Valence	love	1.000		toxic	0.008
	happy	1.000		nightmare	0.005
Arousal	elated	0.960		mellow	0.069
	frenzy	0.965		napping	0.046
Dominance	powerful	0.991		weak	0.045
	leadership	0.983		empty	0.081

NRC VAD Lexicon (Mohammad 2018)

Hence the connotation of a word is a vector in 3-space

Idea 1: Defining meaning by linguistic distribution

Idea 2: Meaning as a point in multidimensional space

Defining meaning as a point in space based on distribution

Each word = a vector (not just "good" or "w_{45}")

Similar words are "**nearby in semantic space**"

We build this space by seeing which words are **nearby in text**

How to represent word meaning numerically?

Idea: represent each word using a vector.

These vectors are called "embeddings" because they are embedded into a space.

The standard way to represent meaning in NLP

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

Fine-grained model of meaning for similarity

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!!!

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
- Later we'll discuss extensions called **contextual embeddings**

Vector Semantics & Embeddings

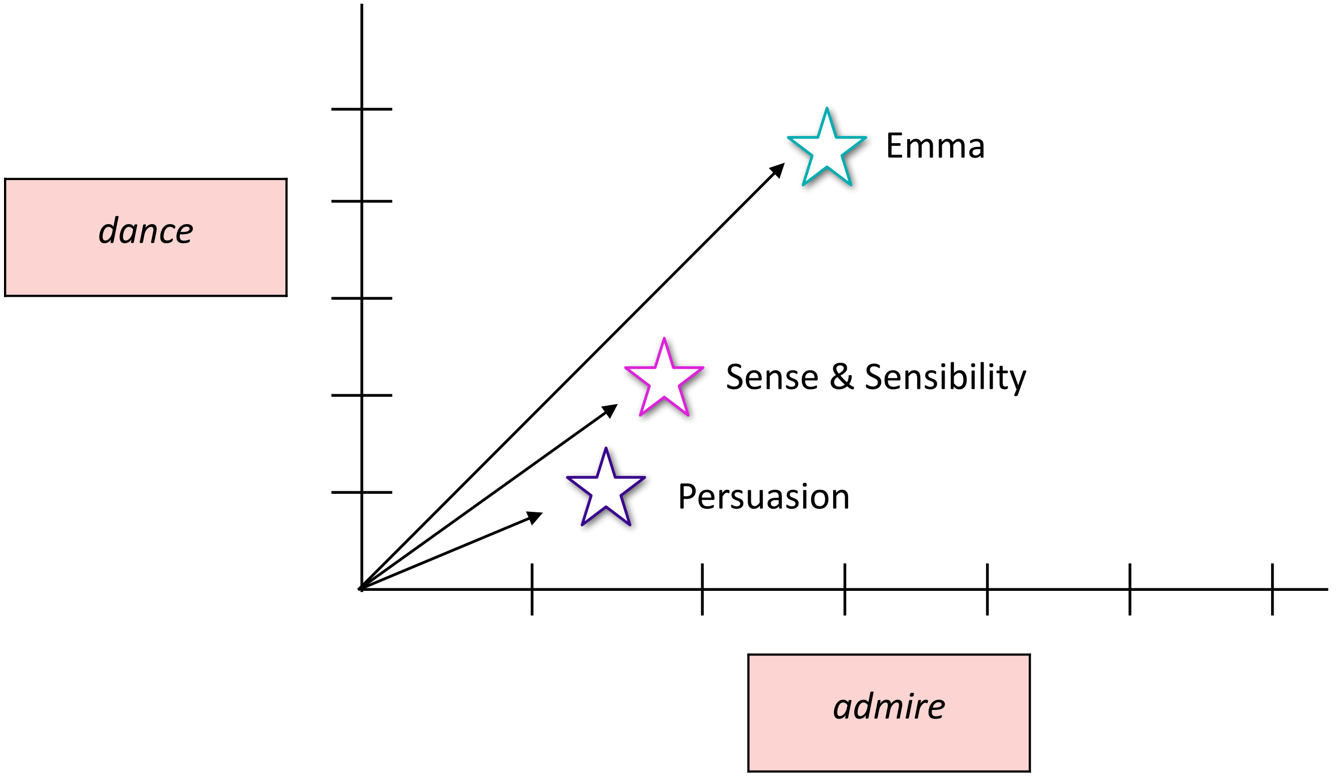
Words and Vectors

Term-document matrix

Each document is represented by a vector of words

	Emma	**Persuasion**	**Sense & Sensibility**
admiral	0	69	0
dance	49	11	21
admire	31	14	18
horse	40	15	24

Visualizing document vectors



Vectors are the basis of information retrieval

	Emma	Persuasion	Sense & Sensibility	Paradise Lost
admiral	0	69	0	0
dance	49	11	21	27
admire	31	14	18	11
horse	40	15	24	5

Idea for word meaning: Words can be vectors too!!!

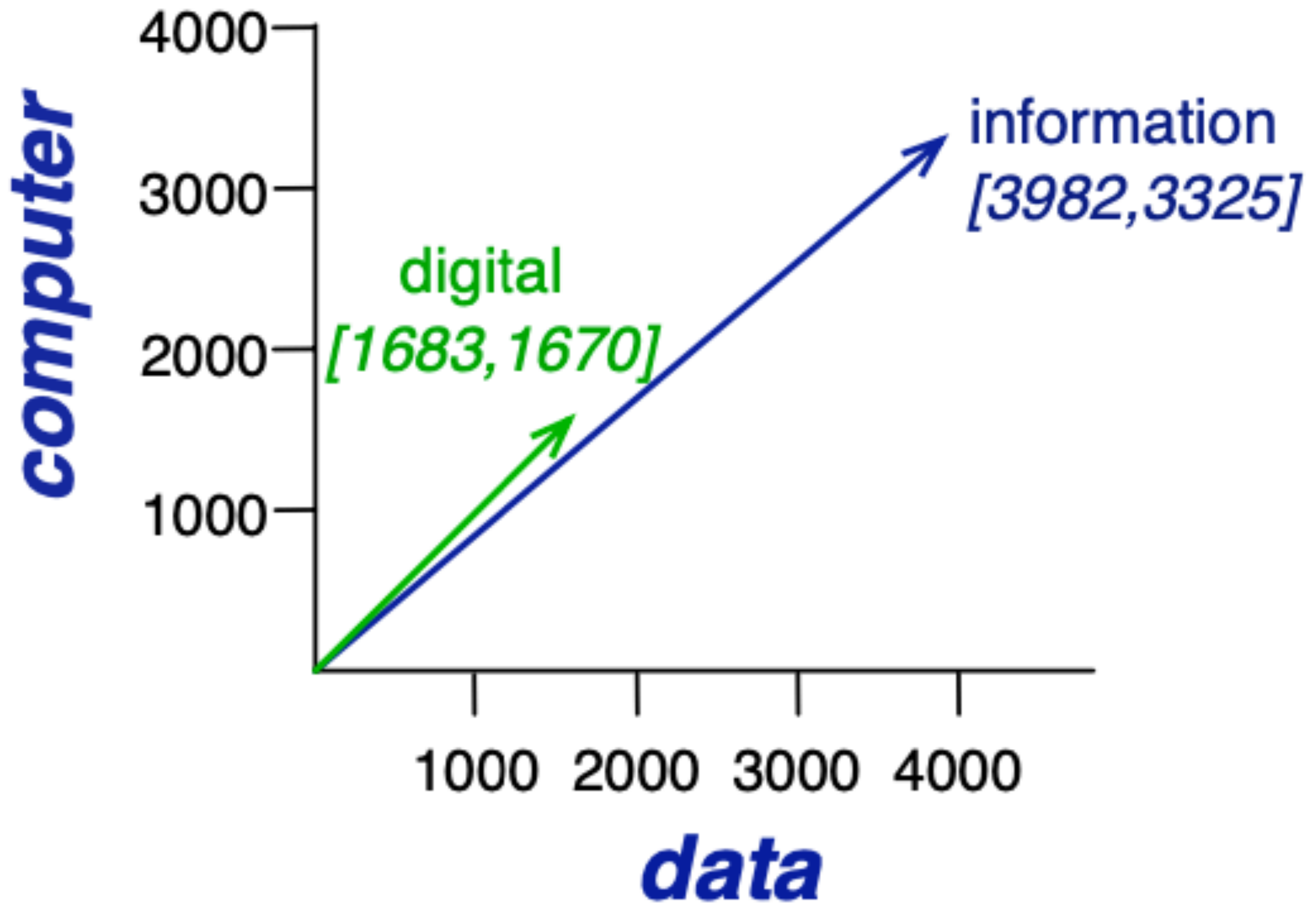
	Emma	**Persuasion**	**Sense & Sensibility**	**Paradise Lost**
admiral	0	69	0	0
dance	49	11	21	27
admire	31	14	18	11
horse	40	15	24	5

More common: word-word matrix (or "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**	**...**
cherry	0	...	2	8	9	442	25	...
strawberry	0	...	0	0	1	60	19	...
digital	0	...	1670	1683	85	5	4	...
information	0	...	3325	3982	378	5	13	...



computer
4000
3000
2000
1000
information
[3982,3325]
digital
[1683,1670]
1000
2000
3000
4000
data

Vector Semantics & Embeddings

Computing word similarity

Computing word similarity: Dot product and cosine

The dot product between two vectors is a scalar:

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

Big when 2 vectors have the same values in the same dimensions

Problem with raw dot-product

Dot product is higher if a vector is longer (has high values in many dimensions).

Vector length:

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

Solution: normalize by vector length

Vector length:

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

Normalized dot product:

$$\frac{v \cdot w}{|v||w|} = \frac{\sum_{i=1}^{N} v_i w_i}{\sqrt{\sum_{i=1}^{N} v_i^2}\sqrt{\sum_{i=1}^{N} w_i^2}}$$

Surprise!

This is the cosine of the angle between the two vectors!

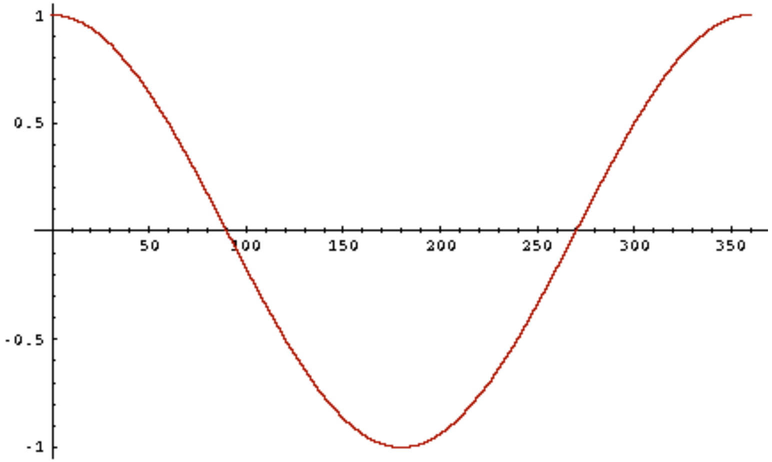
$$\text{cosine}(\mathbf{v}, \mathbf{w}) = \frac{\mathbf{v} \cdot \mathbf{w}}{|\mathbf{v}||\mathbf{w}|} = \frac{\sum_{i=1}^{N} v_i w_i}{\sqrt{\sum_{i=1}^{N} v_i^2} \sqrt{\sum_{i=1}^{N} w_i^2}}$$

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

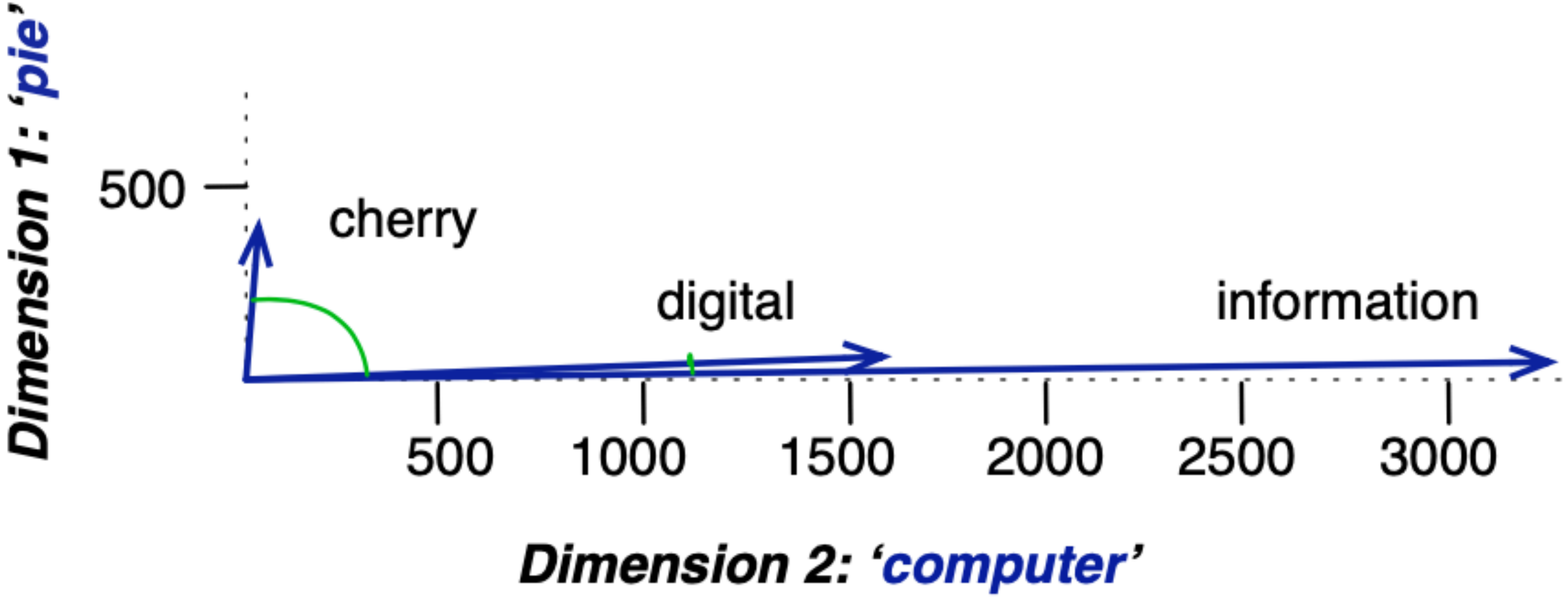
Cosine examples

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

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

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

Visualizing cosine similarity



Vector Semantics & Embeddings

Term Frequency - Inverse Document Frequency (TF-IDF)

Take another look at our Austen word frequencies:

	Emma	**Persuasion**	**Sense & Sensibility**
admiral	0	69	0
dance	49	11	21
admire	31	14	18
horse	40	15	24

Raw frequency is a bad representation

- Word counts for *Emma* are generally higher because it is a longer novel.
- Another issue: some words are so frequent that they aren't very informative: *the, it,* or *they*

Solution 1: tf-idf

tf-idf: Term Frequency - Inverse Document Frequency

Term Frequency:

$tf_{t,d} = count(t, d)$

Inverse Document Frequency:

$idf_t = \frac{N}{df_t}$

N = # of documents

df_t = count of documents in which t occurs

Term Frequency

$$\mathrm{tf}_{t,d} = \mathrm{count}(t,d)$$

tf(admiral,Persuasion) = 69

tf(horse,Persuasion) = 15

	Emma	Persuasion	Sense & Sensibility
admiral	0	69	0
dance	49	11	21
admire	31	14	18
horse	40	15	24

Inverse Document Frequency

$$\text{idf}_t = \frac{N}{\text{df}_t}$$

idf(admiral) = $\frac{3}{1}$

idf(horse) = $3/3$

	Emma	Persuasion	Sense & Sensibility
admiral	0	69	0
dance	49	11	21
admire	31	14	18
horse	40	15	24

TF-IDF

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

180 + 12

tf-idf(admiral,Persuasion) = $64 \times 3 = 192$

tf-idf(horse,Persuasion) = $15 \times 1 = 15$

	Emma	**Persuasion**	**Sense & Sensibility**
admiral	0	69	0
dance	49	11	21
admire	31	14	18
horse	40	15	24

What is a document?

Could be a play or a Wikipedia article

But for the purposes of tf-idf, documents can be **anything**; we often call each paragraph a document!

Vector Semantics & Embeddings

Positive Pointwise Mutual Information (PPMI)

Pointwise Mutual Information

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

$$PMI(x, y) = \log_2 \left(\frac{P(x,y)}{P(x)\,p(y)} \right)$$

Different if not conditionally independent

1 if $p(x,y) = P(x)P(y)$

Pointwise 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 two words: (Church & Hanks 1989)

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

$$\text{PMI}(word_1,\ word_2) = \log_2 \frac{P(word_1, word_2)}{P(word_1)\,P(word_2)}$$

Positive Pointwise Mutual Information

- Issue: PMI ranges from $-\infty$ to $+\infty$
- What do negative values mean?
  - Things are co-occurring **less than** we expect by chance
  - Unreliable without enormous corpora
    - Imagine w1 and w2 whose probability is each 10^{-6}
    - Hard to be sure p(w1,w2) is significantly different than 10^{-12}

Computing PPMI on a term-context matrix

Matrix F with W rows (words) and C columns (contexts)

f_{ij} is # of times w_i occurs in context c_j

$$pmi_{ij} = \log_2 \frac{p_{ij}}{p_{i*}p_{*j}} \qquad ppmi_{ij} = \begin{cases} pmi_{ij} & \text{if } pmi_{ij} > 0 \\ 0 & \text{otherwise} \end{cases}$$

	computer	data	result	pie	sugar	count(w)
cherry	2	8	9	442	25	486
strawberry	0	0	1	60	19	80
digital	1670	1683	85	5	4	3447
information	3325	3982	378	5	13	7703
count(context)	4997	5673	473	512	61	11716

Computing PPMI on a term-context matrix

Matrix F with W rows (words) and C columns (contexts)

f_{ij} is # of times w_{i} occurs in context c_{j}

$$pmi_{ij} = \log_2 \frac{p_{ij}}{p_{i*}p_{*j}} \qquad p_{ij} = \frac{f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}} \qquad p_{i*} = \frac{\sum_{j=1}^{C} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}} \qquad p_{*j} = \frac{\sum_{i=1}^{W} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

	computer	data	result	pie	sugar	count(w)
cherry	2	8	9	442	25	486
strawberry	0	0	1	60	19	80
digital	1670	1683	85	5	4	3447
information	3325	3982	378	5	13	7703
count(context)	4997	5673	473	512	61	11716

Computing PPMI on a term-context matrix

Matrix F with W rows (words) and C columns (contexts)

f_{ij} is # of times w_i occurs in context c_j

$$p_{ij} = \frac{f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

p(w=information,c=data) =

p(w=information) =

p(c=data) =

	computer	data	result	pie	sugar	count(w)
cherry	2	8	9	442	25	486
strawberry	0	0	1	60	19	80
digital	1670	1683	85	5	4	3447
information	3325	3982	378	5	13	7703
count(context)	4997	5673	473	512	61	11716

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

	p(w,context)					p(w)
	computer	**data**	**result**	**pie**	**sugar**	**p(w)**
cherry	0.0002	0.0007	0.0008	0.0377	0.0021	0.0415
strawberry	0.0000	0.0000	0.0001	0.0051	0.0016	0.0068
digital	0.1425	0.1436	0.0073	0.0004	0.0003	0.2942
information	0.2838	0.3399	0.0323	0.0004	0.0011	0.6575
p(context)	0.4265	0.4842	0.0404	0.0437	0.0052	

pmi(information,data) =

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

	p(w,context)					p(w)
	computer	**data**	**result**	**pie**	**sugar**	**p(w)**
cherry	0.0002	0.0007	0.0008	0.0377	0.0021	0.0415
strawberry	0.0000	0.0000	0.0001	0.0051	0.0016	0.0068
digital	0.1425	0.1436	0.0073	0.0004	0.0003	0.2942
information	0.2838	0.3399	0.0323	0.0004	0.0011	0.6575
p(context)	0.4265	0.4842	0.0404	0.0437	0.0052	

pmi(information,data) = $\log_2$ (.3399 / (.6575*.4842)) = .0944

Resulting PPMI matrix (negatives replaced by 0)

	computer	**data**	**result**	**pie**	**sugar**
cherry	0	0	0	4.38	3.30
strawberry	0	0	0	4.10	5.51
digital	0.18	0.01	0	0	0
information	0.02	0.09	0.28	0	0

Weighting PMI

PMI is biased toward infrequent events

- Very rare words have very high PMI values

Two solutions:

- Give rare words slightly higher probabilities
- Use add-one smoothing (which has a similar effect)