Natural Language Processing

ECS763P

Logical and Distributional Semantics

Ch. 15-17 of online J&F

March 20th 2017

Distributional Hypothesis

Words that occur in similar contexts tend to have similar meanings. This insight was first formulated by Harris (1954) who said:

> "oculist and eye-doctor . . . occur in almost the same environments"

and more generally that

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

The most famous statement of the principle comes a few years later from the linguist Firth (1957), who phrased it as

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

The meaning of a word is thus related to the distribution of the words around it.

Let's test it!

Imagine you had never seen the word tesguino, but given the following text:

A bottle of tesguino is on the table.
Everybody likes tesguino .
Tesguino makes you drunk.
We make tesguino out of corn.

you can guess what it means:

a fermented drink similar to beer made of corn

Automatising it

This intuition can be automatised by counting words in the context of tesguino. These will tend to be words like

bottle and drunk

the same as the words around

beer or liquor or tequila.

Since there are same contexts around certain words, helps us discover the similarity between these words.

between beer and liquor and tesguino .

This simple method can be made advanced by taking into account more sophisticated features of the context, e.g. syntactic features like

'occurs before drunk', 'occurs after bottle',
'is the direct object of likes.

Distributional/Vector Semantics

The meaning of a word is computed from the distribution of words around it, represented as a vector of numbers, related to frequency counts of the word in context.

A vast domain of applications:
named entity extraction, parsing, semantic role labeling, relation extraction, but mainly:

compute semantic similarity

between

words, sentences, documents.

An important tool in applications like question answering, summarization, automatic essay grading, etc.

Distributional/Vector Semantics

The meaning of a word is computed from the distribution of words around it, represented as a vector of numbers, related to frequency counts of the word in context.

A vast domain of applications:
named entity extraction, parsing, semantic role labelling, relation extraction, but mainly:

compute semantic similarity

between

words, sentences, documents.

An important tool in applications like question answering, summarization, automatic essay grading, etc.

Basic Linear Algebra

- A vector, e.g. v, w is an array of numbers,e.g.

  [1, 3, 37, 5] [1, 2, 58, 117] [8,12, 1, 0] [15,36,5,0]

- A vector space, e.g. V, W is a collection of vectors satisfying certain properties, e.g. being closed under addition and scalar multiplication.

- The entries of vectors are not arbitrary. They correspond to a quantity with respect to a basis, e.g. 1, 1, 8, 15 are the quantities of the above vectors with respect to basis 1 of the vector space.

- The number of basis of a vector space is called its dimension, it is denoted by |V|, |W|, etc, e.g. the above vector space has dimension 4.

Term-Document Matrix

The first vector space model developed for information retrieval by Salton in 1971.
Documents are represented by vectors in a vector space whose basis are words.

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	1	8	15
soldier	2	2	12	36
fool	37	58	1	5
clown	5	117	0	0

Figure 15.1 The term-document matrix for four words in four Shakespeare plays. Each cell contains the number of times the (row) word occurs in the (column) document.

Term-Document Matrix

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	1	8	15
soldier	2	2	12	36
fool	37	58	1	5
clown	5	117	0	0

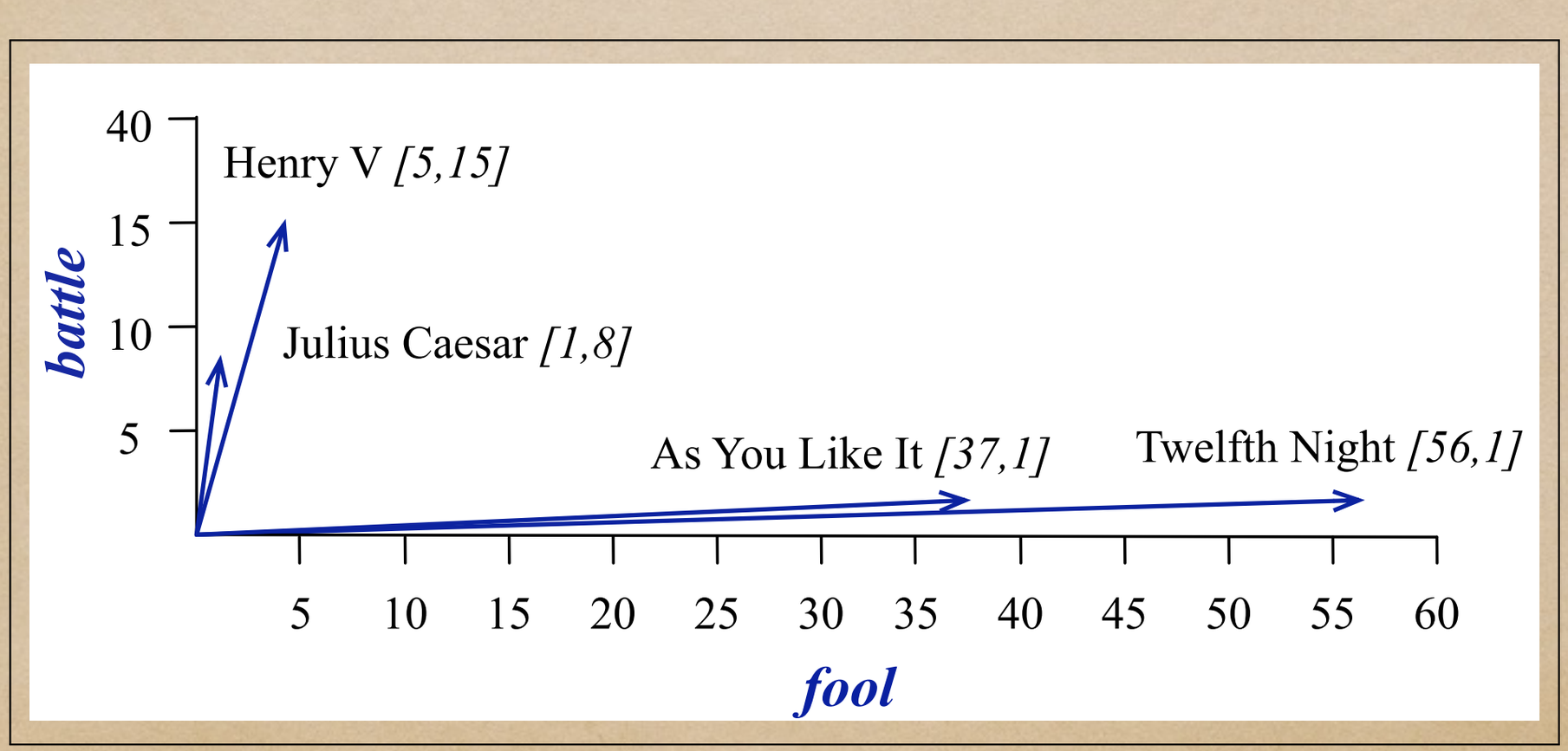
Figure 15.1 The term-document matrix for four words in four Shakespeare plays. Each cell contains the number of times the (row) word occurs in the (column) document.

Each play is represented by a vector:

As You like It: [1, 3, 37, 5]
Twelfth Night: [1, 2, 58, 117]
Julius Caesar: [8,12, 1, 0]
Henry V: [15,36,5,0]

Term-Document Matrix

A vector v in the vector space V can be thought of as a point in the |V| dimensional space. Our Shakespeare plays are points in the 4 dimensional space.
A visualisation of these vectors, with only two of the dimensions (battle and fool) is as follows:



What is it good for?

- Finding similar documents

  Two documents are similar iff they contain similar words.
- As You Like IT and Twelfth Night have more of words clown and fool than soldier and battle. So they are similar to each other. Indeed they are both comedies
- The geometric or angular distance between vectors is used to represent the similarity between them.
- As You Like IT and Twelfth Night have a smaller angle between them and so do Julius Cesar and Henry V.
- As You Like IT and Twelfth Night are both further away from Julius Cesar and Henry V.
- In IR, the distance between the vector of a query and the vector of the documents is computers and similar documents to query are retrieved as the result of search.

Word-Context Matrix

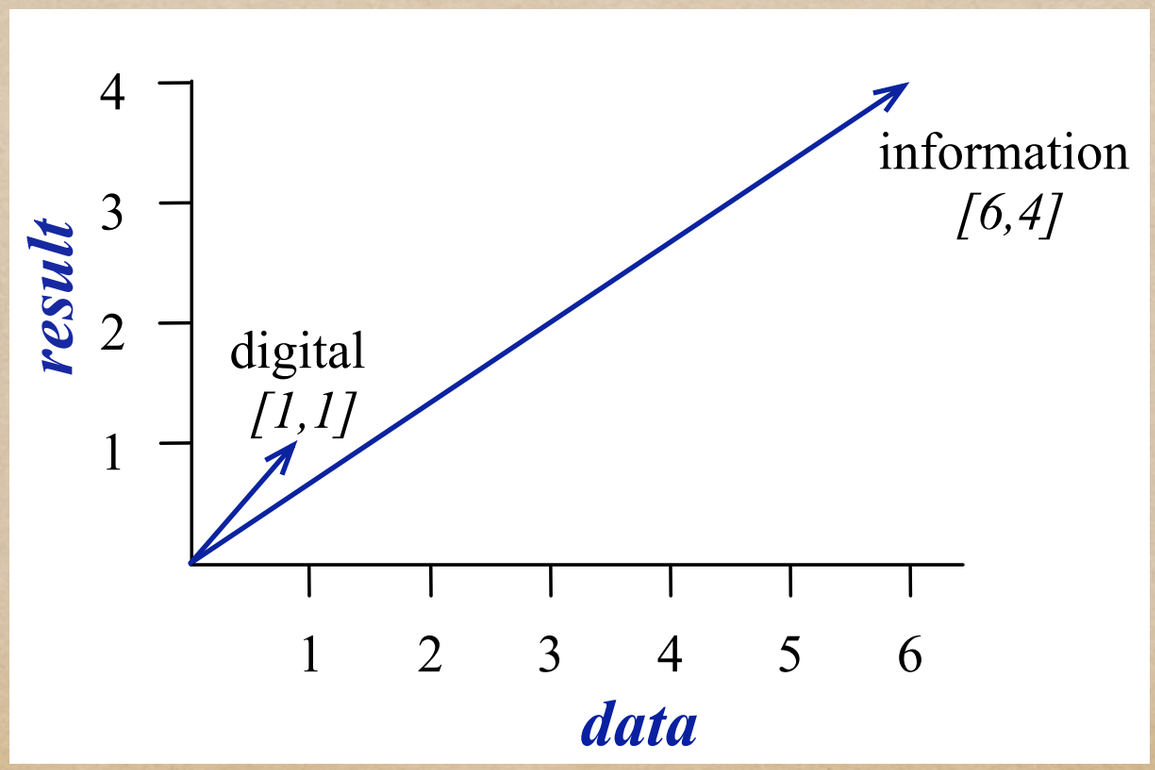
- Columns and rows are both labelled by words.
- Each cell represents the number of times a target word (label of the row) occurs in the context of a context word (label of column).
- Being in a context means being in a window of k-words around it.
- For example, here are 7-word windows around four words of the Brown corpus:

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

Word-Context Matrix

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	



Some Parameters of the Model

- The dimension of the vector space is the size of the vocabulary of the corpus, e.g. 10,000 or 50,000.

- But cells of the matrix will be 0 leading to sparse vectors, so one can use efficient sparse matrix computation algorithms to compute with these vectors.

- Size of the context window depends on the task, it is some where between 1 and 8.

- The smaller the window, the more syntactic information the co-occurrence counts represent, the longer the window, the more semantic information.

Some Parameters of the Model

- Raw frequency that the cells record, is not the best measure of association between the words.
- For example, we want good measure that can distinguish between very common words such as it, the, a, they, ….
- We want a measure by which informative context words are weighed higher than non-informative ones.
- Examples of such measures are:
  - Mutual Information (Church and Hanks 1989)
  - Pointwise Mutual Information (Fano 1961)

$$\sum_x \sum_y P(x,y) \log_2 \frac{P(x,y)}{P(x)P(y)}$$

$$\log_2 \frac{P(x,y)}{P(x)P(y)}$$

Pointwise Mutual Information

Pointwise mutual information between two events x,y tells us how often they occurred together in comparison to our expectation if the they were independent.

$$I(x,y) = \log_2 \frac{P(x,y)}{P(x)P(y)}$$

For word-context matrices, PMI is defined between a word and a context word.

$$\text{PMI}(w,c) = \log_2 \frac{P(w,c)}{P(w)P(c)}$$

How often w and c were observed together

How often we expect w and c to occur independently

How much they occurred together than we expect by chance.

Positive Pointwise Mutual Information

- A logarithm can return negative values.
- In this case, if PMI(w,c) <0, it means that w occurred with c, less frequent than chance.
- This returns unreliable quantities for the word vectors.
- So people consider Positive PMI or PPMI.

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

Computing PPMI

Given a word-context matrix F with W rows and C columns.

Take fij to be the number of time word wi occurred in the context of word cj.

$$P(w_i, c_j) = \frac{f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

$$P(w_i) = \frac{\sum_{j=1}^{C} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

$$P(c_j) = \frac{\sum_{i=1}^{W} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

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

Computing PPMI

Given a word-context matrix F with W rows and C columns.

Take fij to be the number of time word wi occurred in the context of word cj.

$$P(w_i,c_j) = \frac{f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}} \qquad P(\text{w=information,c=data}) = \frac{6}{19} = .316$$

$$P(w_i) = \frac{\sum_{j=1}^{C} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

$$P(c_j) = \frac{\sum_{i=1}^{W} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

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

Computing PPMI

Given a word-context matrix F with W rows and C columns.

Take fij to be the number of time word wi occurred in the context of word cj.

$$P(w_i, c_j) = \frac{f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

$$P(w_i) = \frac{\sum_{j=1}^{C} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

f41+ f42+f43+…= no of times w4 occurred in any context.

$P(\text{w=information})$

$$P(c_j) = \frac{\sum_{i=1}^{W} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

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

Computing PPMI

Given a word-context matrix F with W rows and C columns.

Take fij to be the number of time word wi occurred in the context of word cj.

$$P(w_i, c_j) = \frac{f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

f41+ f42+f43+…= no of times w4 occurred in any context.

$$P(w_i) = \frac{\sum_{j=1}^{C} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

$$P(\text{w=information}) = \frac{11}{19} = .579$$

$$P(c_j) = \frac{\sum_{i=1}^{W} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

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

Computing PPMI

Given a word-context matrix F with W rows and C columns.

Take fij to be the number of time word wi occurred in the context of word cj.

$$P(w_i,c_j) = \frac{f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}} \qquad P(\text{w=information,c=data}) = \frac{6}{19} = .316$$

$$P(w_i) = \frac{\sum_{j=1}^{C} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

f19+f29+f39+…= no. of times any word occurred with c9..

$$P(c_j) = \frac{\sum_{i=1}^{W} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}} \qquad P(\text{c=data})$$

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

Computing PPMI

Given a word-context matrix F with W rows and C columns.

Take fij to be the number of time word wi occurred in the context of word cj.

$$P(w_i,c_j) = \frac{f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}} \qquad P(\text{w=information,c=data}) = \frac{6}{19} = .316$$

$$P(w_i) = \frac{\sum_{j=1}^{C} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

f19+f29+f39+…= no. of times any word occurred with c9..

$$P(c_j) = \frac{\sum_{i=1}^{W} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}} \qquad P(\text{c=data}) = \frac{7}{19} = .368$$

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

Computing PPMI

Given a word-context matrix F with W rows and C columns.

Take fij to be the number of time word wi occurred in the context of word cj.

$$P(w_i,c_j) \approx \frac{f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}} \qquad P(\text{w=information,c=data}) = \frac{6}{19} = .316$$

$$P(w_i) \approx \frac{\sum_{j=1}^{C} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}} \qquad P(\text{w=information}) = \frac{11}{19} = .579$$

$$P(c_j) \approx \frac{\sum_{i=1}^{W} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}} \qquad P(\text{c=data}) = \frac{7}{19} = .368$$

$$\text{PPMI}(w,c) = \max(\log_2 \frac{P(w,c)}{P(w)P(c)}, 0) \qquad \log 2(.316/(.368 * .579)) = .568$$

Comparison

Raw Frequencies

	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	

Joint Probability

	p(w,context)					p(w)
	computer	**data**	**pinch**	**result**	**sugar**	**p(w)**
apricot	0	0	0.05	0	0.05	0.11
pineapple	0	0	0.05	0	0.05	0.11
digital	0.11	0.05	0	0.05	0	0.21
information	0.05	.32	0	0.21	0	0.58
p(context)	0.16	0.37	0.11	0.26	0.11	

PPMI

	computer	data	pinch	result	sugar
apricot	0	0	2.25	0	2.25
pineapple	0	0	2.25	0	2.25
digital	1.66	0	0	0	0
information	0	0.57	0	0.47	0

$$\log 2(.05/(.16 * .58)) = -0.618$$

Smoothing

PMI has the problem of being biased toward infrequent events; very rare words tend to have very high PMI values. A solution is smoothing.
Laplace Smoothing: before computing PMI, add a small constant k (e.g. -1, -2, -3) to all counts. Thus increasing the 0 counts and shrinking the non zero counts.

	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

Figure 15.8 Laplace (add-2) smoothing of the counts in Fig. 15.4.

	computer	data	pinch	result	sugar
apricot	0	0	0.56	0	0.56
pineapple	0	0	0.56	0	0.56
digital	0.62	0	0	0	0
information	0	0.58	0	0.37	0

Figure 15.9 The Add-2 Laplace smoothed PPMI matrix from the add-2 smoothing counts in Fig. 15.8.

Smoothing

Another method (Levy et al 2015) is to raise the probability of context to a power.

$$\text{PPMI}_\alpha(w,c) = \max(\log_2 \frac{P(w,c)}{P(w)P_\alpha(c)}, 0)$$

$$P_\alpha(c) = \frac{count(c)^\alpha}{\sum_c count(c)^\alpha}$$

Experiments have shown that raising to the power 0.75 betters the results.

This words because raising the probability to $\alpha = 0.75$ increases the probability assigned to rare contexts, and hence lowers their PMI ($P\alpha(c) > P(c)$ when c is rare)

Other measures

TF-IDF: from Information Retrieval.

TF: frequency of the word in the document (Luhn 1975),

IDF: inverse document frequency (Sparck Jones, 1972).

It is one way of assigning higher weights to these more discriminative words. IDF is defined using the fraction N/dfi, where N is the total number of documents in the collection, and d fi is the number of documents in which term i occurs.

The fewer documents in which a term occurs, the higher this weight. The lowest weight of 1 is assigned to terms that occur in all the documents. Because of the large number of documents in many collections, this measure is usually squashed with a log function.

$$\text{idf}_i = \log\left(\frac{N}{df_i}\right) \qquad w_{ij} = \text{tf}_{ij}\text{idf}_i$$

Other measures

t-test: measures the difference between the observed and expected means, normalised by variance.

$$t = \frac{\bar{x} - \mu}{\sqrt{\frac{s^2}{N}}}$$

For word-context matrices, t-test becomes as follows (Manning and Schutze 1999).

The variance s2 can be approximated by the expected probability P(a)P(b). N i ignored since it is constant (Curran 2003).

$$\text{t-test}(a,b) = \frac{P(a,b) - P(a)P(b)}{\sqrt{P(a)P(b)}}$$

Measuring Similarity

The geometric distance between vectors of words. If vectors have length 1, this is their dot product:

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

If not, it is the cosine between the two vectors.

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

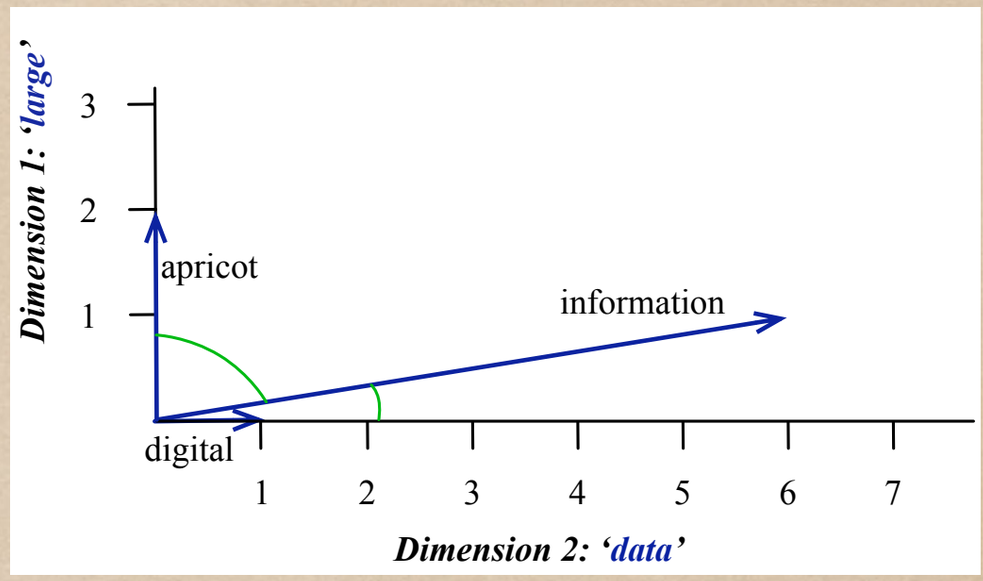
Other measures:

Jaccard and Dice, from Information Retrieval.

Kullback-Leibler Divergence and Shannon-Jason from relative entropy in Information Theory.

Examples of Similarity

	large	data	computer
apricot	2	0	0
digital	0	1	2
information	1	6	1



$$\cos(\text{apricot}, \text{information}) = \frac{2+0+0}{\sqrt{4+0+0}\sqrt{1+36+1}} = \frac{2}{2\sqrt{38}} = .16$$

$$\cos(\text{digital}, \text{information}) = \frac{0+6+2}{\sqrt{0+1+4}\sqrt{1+36+1}} = \frac{8}{\sqrt{38}\sqrt{5}} = .58$$

Syntactic Vector Models

Instead of defining a word's context by nearby words, we can define it by syntactic relations.

> The meaning of entities, and the meaning of grammatical relations among them, is related to the restriction of to other entities. (Harris 1968)

The similarity between meanings of duty and responsibility is mirrored in their syntactic behavior. They be modified by adjectives: additional, administrative, collective, congressional; both can be the direct objects of verbs: assert, assign, assume, attend to, avoid, become, etc.

	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

Evaluating Vector Models

WordSim-353 : a set of of ratings for 353 noun pairs.
SimLex-999 : quantifies similarity (cup, mug) rather than relatedness (cup, coffee), and includes adjective, noun and verb pairs.
TOEFL dataset : a set of 80 questions, each consisting of a target word with 4 additional word choices; the task is to choose which is the correct synonym.
Stanford Contextual Word Similarity (SCWS) : human similarity ratings for 2,003 pairs of words, but in their sentential context.
Word2Vec Dataset: a set of patterns “a is to b as c is to d”. Given a, b, and c, the task is to find d. The words are in certain semantic relationships with each other. For example Athens is to Greece as Oslo is to ? …. Norway.: MIkolov et al. 2013.