

# Introduction to Distributional Semantics

Aurélie Herbelot

University of Trento/Geneva

Geneva 2016

# Distributional semantics

cat

```
cat 0.111055 0.1408689999999999 0.181254 0.018981999999999999 0.084370999999999999 0.348416 0.226279000000000001 0.344640999999999998 0.042896999999999998 0.295
9379999999999999 -0.16402 -0.073202000000000003 -0.103635 -0.030491999999999998 0.066420999999999994 -0.049784000000000002 0.194705999999999999 -0.183585 0.253543000000000002 -
0.033194000000000001 -0.141780999999999999 0.146317 0.180643 -0.232701999999999999 -0.022716 0.126582 -0.070262000000000005 -0.278615999999999997 0.207443999999999999 -
0.083652000000000004 0.222315000000000001 -0.0045189999999999996 0.149285 0.078797000000000006 -0.115759 0.048432000000000003 -0.031777 -0.006248000000000001 0.08933000000
000007 -0.023803999999999999 0.082723000000000005 0.168906 0.078835000000000002 0.141391999999999999 0.085622000000000004 -0.175147 -0.195627 -0.024833999999999998 0.118
161 -0.13965 -0.074168999999999999 -0.224287999999999999 -0.053401999999999998 -0.044496000000000001 0.125325999999999999 -0.375612 -0.001531 -0.056882000000000002 0.057
7099999999999999 -0.062064000000000001 0.092183000000000001 -0.251730000000000001 -0.084278000000000006 0.046288999999999997 -0.168343999999999999 -0.173692999999999999 -0.050625999999999997
0.176053999999999999 0.206771000000000001 0.052115000000000002 0.079504000000000005 0.107483 0.075187000000000004 -0.106776000000000001 -0.075086000000000003 -0.187929000000000010
.275260000000000001 0.247557 -0.051232 -0.160220000000000001 0.133161 0.198506000000000001 0.016664000000000002 0.157860000000000001 -0.039862000000000002 0.101566 0
.108697 0.043320999999999999 -0.335065999999999999 0.025768999999999999 0.073440000000000005 0.250143 -0.052051 0.250624000000000001 -0.030228000000000001 -0.242068000000000010
.025908999999999999 -0.050355999999999998 -0.335139000000000002 0.1716 -0.034415000000000001 0.138160000000000001 -0.15342 -0.102969 0.166309000000000001 -0.091919000000000001
-0.054598000000000001 -0.019181 0.135098 -0.259027000000000001 -0.026903 -0.352287999999999999 -0.184472 -0.205550000000000001 -0.257994999999999997 0.032053999999999999
.128298 0.027734000000000002 0.259861000000000001 0.073891999999999999 0.281704999999999998 0.112188 0.049931000000000003 0.118045 -0.033848000000000003 -0.187036000000000001-
0.208687000000000001 0.090014999999999998 -0.180718999999999999 0.056547 0.118782 -0.148925 -0.016293999999999999 0.176967000000000001 -0.113207 -0.119296 0.019
5989999999999999 0.103338 0.393834000000000002 0.012577 0.107939999999999999 -0.161198000000000001 -0.138795 0.064596000000000001 0.355710000000000003 0.153634999999999999 0
.199330000000000001 0.185259000000000001 0.145424 0.169016 -0.219188999999999999 0.132873999999999999 0.129263999999999999 -0.359493000000000001 0.266116999999999999 0.040
4539999999999999 0.116953 0.303431000000000001 -0.083718000000000001 0.138116999999999999 0.00365 -0.16718 -0.069752999999999999 0.076752000000000001 0.064975000000000005 0.169
765 -0.219662 0.114191 -0.277698 0.054690000000000003 0.038365999999999997 0.011445 -0.171421999999999999 -0.131268 -0.064385999999999999 0.142971999999999999 -
0.175168999999999999 0.063895999999999994 0.172735 0.087408999999999999 0.034764000000000003 0.035901000000000002 -0.069389999999999993 -0.104528 0.425310999999999999 -0.23
4254999999999999 -0.084248000000000003 -0.015147000000000001 -0.195078 0.100057999999999999 0.042844 0.007611999999999999 0.131622999999999999 0.116436 -0.132316999999999999
.188679999999999999 0.063367999999999994 -0.107372 0.034609000000000001 -0.228632 -0.201574 0.115218 0.320479000000000001 -0.175604000000000001 -0.144564 0.097
733 0.165553000000000001 -0.00329400000000000001 -0.257886 -0.130102 0.107480000000000001 0.181533 0.232147999999999999 -0.006017999999999999 -0.046239000000000002 -0.02
8947000000000001 0.001583999999999999 -0.138519 0.126540999999999999 0.085583999999999993 0.129741 -0.198749000000000001 0.195218 0.218946 -0.152808 0.141
6399999999999999 -0.201956 0.063875000000000001 -0.069394999999999998 0.111301 0.082405999999999993 0.014709 0.033288999999999999 -0.000757999999999999 -0.185143 -0.10
1572 -0.090429999999999999 0.285839999999999998 0.146926 -0.213722 -0.097059000000000006 -0.135673999999999999 0.028582 -0.044374999999999998 0.021683999999999998 -0.05
3808000000000002 -0.055474999999999997 0.090394000000000002 0.014799 -0.10276 0.216563000000000001 -0.138073 -0.092768000000000003 -0.036517999999999998 -0.1472700000
0000001 -0.025995000000000001 0.159814000000000001 0.003571 0.007933000000000008 0.136969000000000001 0.036406000000000001 -0.032057000000000002 0.069616999999999998 -0.1847330000
000001 0.206723999999999999 0.075669 0.167176999999999999 0.139393999999999999 -0.222204000000000001 0.183990999999999999 0.133595999999999999 -0.035718 0.184100000000000001 -
0.147451 0.088061 -0.101898 0.312429999999999999 -0.391222999999999999 0.147397 -0.233542 -0.066458000000000003 0.215508000000000001 0.080250000000000002 -0.00
957299999999999999 -0.033706 0.022884999999999999 0.18478 -0.060366000000000003 0.383491000000000003 -0.033888000000000001 0.079256999999999994 -0.296910000000000001 0.261977000000
000002 0.004414000000000004 0.189428999999999999 0.123921 -0.209244000000000001 0.066601999999999995 -0.148413999999999999 0.327164000000000001 0.060151999999999997 0.112337999999
999999 0.062169000000000002 0.094102000000000005 0.120551000000000001 0.070698999999999998 -0.230037999999999999 0.110908000000000001 0.094075000000000006 -0.028483000000000001 -0.03
3847000000000002 0.245353999999999999 -0.117479 -0.080423999999999995 -0.063783000000000006 0.028347000000000001 -0.084446999999999994 -0.169281999999999999 0.081403 -0.18
1231 -0.150254 0.239105999999999999 0.143818 0.200878 0.123437000000000001 -0.175196999999999999 0.038198000000000003 -0.145329000000000001 -0.253917 -0.0406979999
99999999 0.171520000000000001 -0.120871000000000001 0.214007 0.00023900000000000001 0.184209000000000001 0.0007649999999999995 0.0038639999999999998 -0.304211999999999998 -0.12
9683999999999999 0.297440000000000001 0.017788000000000002 -0.169756999999999999 0.093423999999999993 -0.264050000000000001 0.074779299999999998 0.106505 -0.041367000000000001 0.149
785 -0.016388 0.11132 0.109249 -0.021246999999999999 -0.057563999999999997 0.080926999999999999 0.144219000000000001 -0.057579999999999999 0.085472999999999993 -0.2438539999
99999999 -0.128340000000000001 -0.155430000000000001 -0.10505 -0.070879999999999999 -0.015923 0.030908000000000001 -0.061946000000000001 0.009096999999999992 0.057092999999999999
0.126631999999999999 0.187478000000000001 -0.015833 0.189742999999999999 -0.073700000000000006 0.048162000000000003 -0.003114 0.204314 0.156501 -0.13102 -0.00
9320000000000002 -0.135713 -0.131604 0.069156999999999996 0.040224000000000003 -0.099108000000000002 -0.049022000000000003 -0.024277 -0.202825000000000001 -0.109736 -0.00
0.214961999999999999 -0.082313999999999998 -0.005989000000000004 0.109237 -0.10668 -0.064618999999999996 0.01226 0.147685000000000001 0.248598000000000001 -0.279953999999999998
.150999999999999999 0.049911999999999998 0.068373000000000003
```

## Distributional semantics: a short history

- J.R. Firth: *You shall know a word by the company it keeps* (1957).
- Zelig Harris: *Words that appear in the same contexts are semantically similar* (1954).

## Distributional semantics: a short history



**Ludwig Wittgenstein:** ‘Meaning is use’: ‘Die Bedeutung eines Wortes ist sein Gebrauch in der Sprache’ (Wittgenstein, 1953. 43)

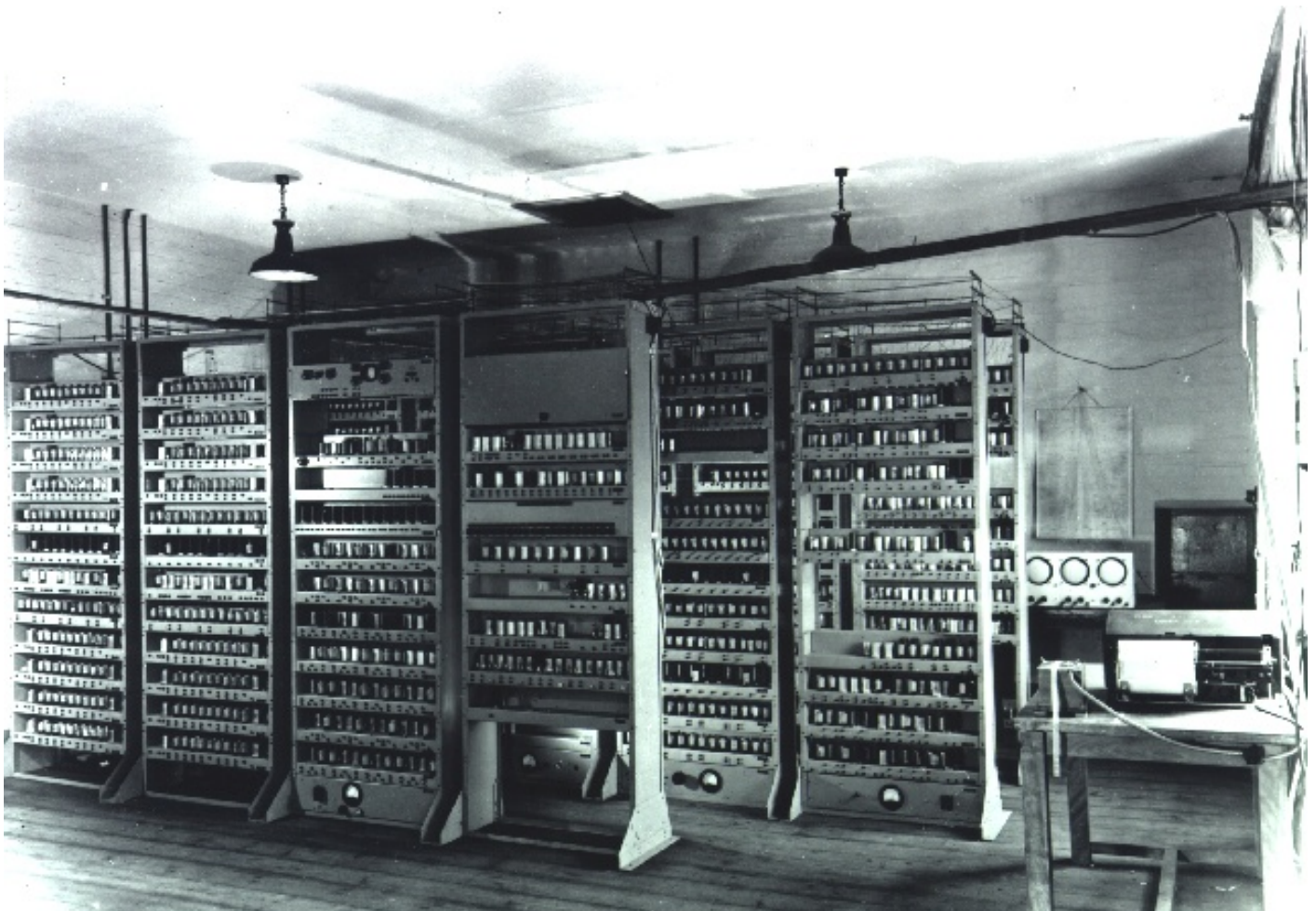


**Margaret Masterman:** Cambridge Language Research Unit (CLRU: 1955–1986).

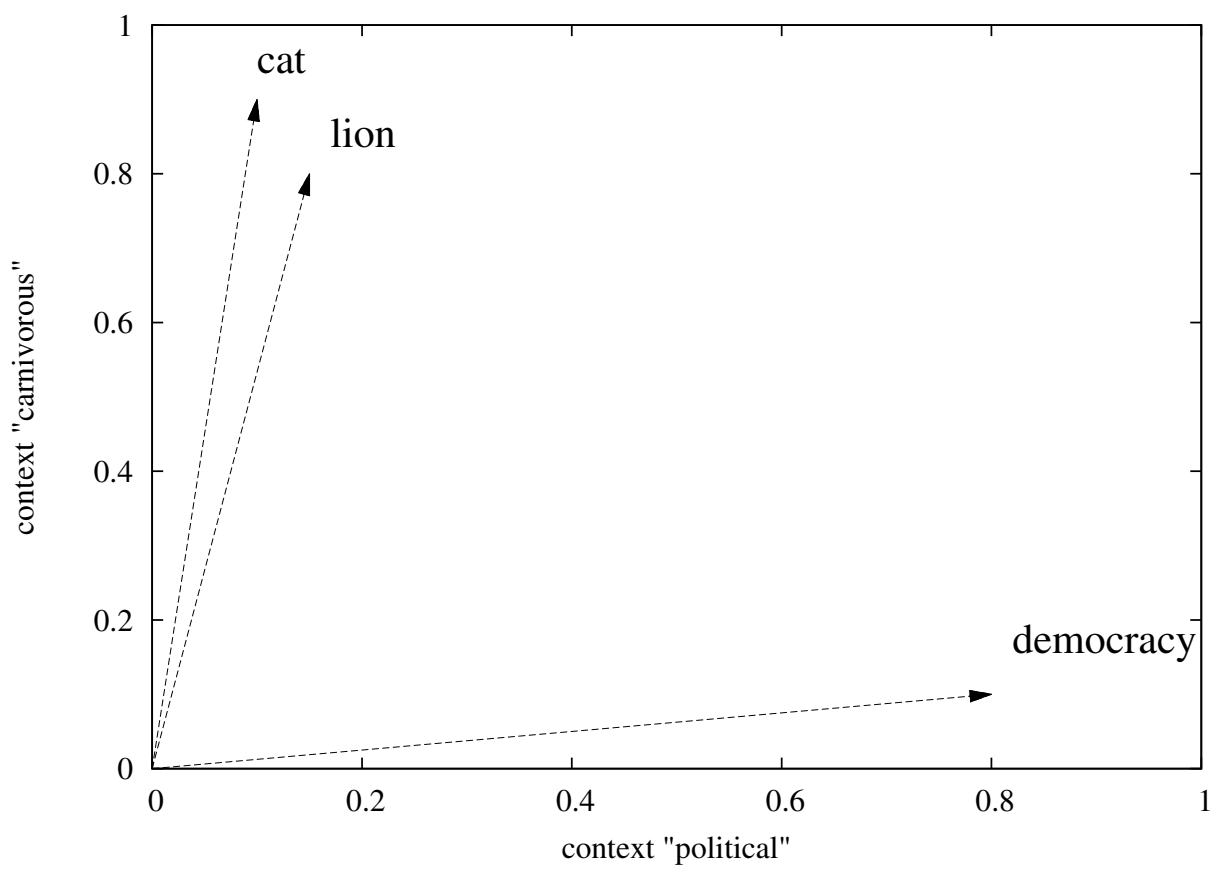


**Karen Spärck-Jones:** Early experiments on distributional semantics: 1963, 1967.

## ‘The’ computer: the EDSAC



# The semantic space



## The components of distributional representations

- Contexts: other words in the close vicinity of the target (*eat*, *mouse*, *sleep*), or syntactic/semantic relations (*eat(x)*, *chase(x,mouse)*, *like(x,sleep)*).
- Weights: usually a measure of how characteristic the context is for the target (e.g. Pointwise Mutual Information).
- A semantic space: a vector space in which dimensions are the contexts with respect to which the target is expressed. The target word is a vector in that space (vector components are given by the weights of the distribution).

# A distributional cat (from the British National Corpus)

0.124 pet-N	0.074 tiger-N	0.063 hate-V
0.123 mouse-N	0.073 jump-V	0.063 asleep-A
0.099 rat-N	0.073 tom-N	0.063 stance-N
0.097 owner-N	0.073 fat-A	0.062 unfortunate-A
0.096 dog-N	0.071 spell-V	0.061 naked-A
0.092 domestic-A	0.071 companion-N	0.061 switch-V
0.090 wild-A	0.070 lion-N	0.061 encounter-V
0.090 duck-N	0.068 breed-V	0.061 creature-N
0.087 tail-N	0.068 signal-N	0.061 dominant-A
0.084 leap-V	0.067 bite-V	0.060 black-A
0.084 prey-N	0.067 spring-V	0.059 chocolate-N
0.083 breed-N	0.067 detect-V	0.058 giant-N
0.080 rabbit-N	0.067 bird-N	0.058 sensitive-A
0.078 female-A	0.066 friendly-A	0.058 canadian-A
0.075 fox-N	0.066 odour-N	0.058 toy-N
0.075 basket-N	0.066 hunting-N	0.058 milk-N
0.075 animal-N	0.066 ghost-N	0.057 human-N
0.074 ear-N	0.065 rub-V	0.057 devil-N
0.074 chase-V	0.064 predator-N	0.056 smell-N
0.074 smell-V	0.063 pig-N	...



# Modelling choices

# Corpus choice

- As much data as possible?
  - British National Corpus (BNC): 100 m words
  - Wikipedia: 897 m words
  - UKWac: 2 bn words
  - ENCOW: 10 bn words
  - ...
- In general preferable, *but*:
  - More data is not necessarily the data you want.
  - More data is not necessarily realistic from a psycholinguistic point of view. We perhaps encounter 50,000 words a day. BNC = 5 years' text exposure.

# The notion of context

- **Context:** if the meaning of a word is given by its context, what does 'context' mean?
  - Word windows (unfiltered):  $n$  words on either side of the lexical item under consideration (unparsed text).

**Example:**  $n=2$  (5 words window):

*... the prime **minister** acknowledged that ...*

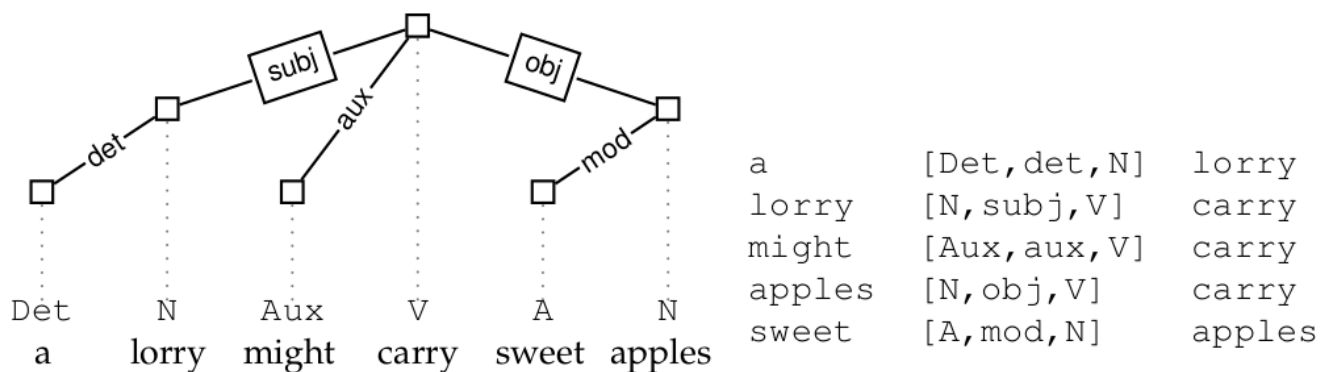
- Word windows (filtered):  $n$  words on either side of the lexical item under consideration (unparsed text). Some words are not considered part of the context (e.g. function words, some very frequent content words). The stop list for function words is either constructed manually, or the corpus is POS-tagged.

**Example:**  $n=2$  (5 words window):

*... the prime **minister** acknowledged that ...*

## The notion of context

- Dependencies: syntactic or semantic. The corpus is converted into a list of directed links between heads and dependents. Context for a lexical item is the dependency structure it belongs to. The length of the dependency path can vary according to the implementation (Padó and Lapata, 2007).



## Parsed vs unparsed data: examples

### word (unparsed)

meaning\_n  
 derive\_v  
 dictionary\_n  
 pronounce\_v  
 phrase\_n  
 latin\_j  
 ipa\_n  
 verb\_n  
 mean\_v  
 hebrew\_n  
 usage\_n  
 literally\_r

### word (parsed)

or\_c+phrase\_n  
 and\_c+phrase\_n  
 syllable\_n+of\_p  
 play\_n+on\_p  
 etymology\_n+of\_p  
 portmanteau\_n+of\_p  
 and\_c+deed\_n  
 meaning\_n+of\_p  
 from\_p+language\_n  
 pron\_rel\_+utter\_v  
 for\_p+word\_n  
 in\_p+sentence\_n

## Context weighting

- Binary model: if context  $c$  co-occurs with word  $w$ , value of vector  $\vec{w}$  for dimension  $c$  is 1, 0 otherwise.

... [a long long long **example** for a distributional semantics] model... ( $n=4$ )

... {a 1} {dog 0} {long 1} {sell 0} {semantics 1}...

- Basic frequency model: the value of vector  $\vec{w}$  for dimension  $c$  is the number of times that  $c$  co-occurs with  $w$ .

... [a long long long **example** for a distributional semantics] model... ( $n=4$ )

... {a 2} {dog 0} {long 3} {sell 0} {semantics 1}...

## Context weighting

- Characteric model: the weights given to the vector components express how *characteristic* a given context is for  $w$ . Functions used include:
  - Pointwise Mutual Information (PMI):

$$pmi_{wc} = \log \frac{p(w, c)}{p(w)p(c)} = \log\left(\frac{f_{wc} * f_{total}}{f_w * f_c}\right) \quad (1)$$

where frequencies are defined *over co-occurrences* (i.e.  $f_w$  is the sum of all  $f_{wc_{i \dots k}}$  and  $f_c$  is the sum of all  $f_{w_1 \dots k c}$ ).

- Derivatives such Positive Pointwise Mutual Information (PPMI), Pointwise Local Mutual Information (PLMI), etc.

# What semantic space?

- Entire vocabulary.
  - + All information included – even rare, but important contexts
  - - Inefficient (100,000s dimensions). Noisy (e.g. *002.png/thumb/right/200px/graph\_n*)
- Top  $n$  words with highest frequencies.
  - + More efficient (5000-10000 dimensions). Only ‘real’ words included.
  - - May miss out on infrequent but relevant contexts.
- Dimensionality reduction (e.g. SVD).



# Getting distributions from text

## Our reference text

Douglas Adams, *Mostly harmless*

The major difference between a thing that might go wrong and a thing that cannot possibly go wrong is that when a thing that cannot possibly go wrong goes wrong it usually turns out to be impossible to get at or repair.

- **Example:** Produce distributions using a word window, frequency-based model

# The semantic space

## Douglas Adams, *Mostly harmless*

the\_DT major\_JJ difference\_NN between\_IN a\_DT thing\_NN that\_WDT might\_MD  
 go\_VB wrong\_JJ and\_CC a\_DT thing\_NN that\_WDT can\_MD not\_RB possibly\_RB  
 go\_VB wrong\_JJ be\_VBZ that\_IN when\_WRB a\_DT thing\_NN that\_WDT can\_MD  
 not\_RB possibly\_RB go\_VB wrong\_JJ go\_VBZ wrong\_JJ it\_PRP usually\_RB  
 turn\_VBZ out\_RP to\_TO be\_VB impossible\_JJ to\_TO get\_VB at\_IN or\_CC repair\_NN

- We assume that we only keep nouns, verbs, adjectives and adverbs in the semantic space.
- **Dimensions:**

go\_V  
 wrong\_J  
 thing\_N  
 possibly\_R  
 be\_V

not\_R  
 difference\_N  
 turn\_V  
 usually\_R  
 major\_J

impossible\_J  
 out\_R  
 repair\_V

# Frequency counts...

## Douglas Adams, *Mostly harmless*

major\_J difference\_N thing\_N go\_V wrong\_J thing\_N not\_R possibly\_R go\_V  
 wrong\_J be\_V thing\_N not\_R possibly\_R go\_V wrong\_J go\_V wrong\_J usually\_R  
 turn\_V out\_R be\_V impossible\_J get\_V repair\_N

### • Counts:

4 go\_V  
 4 wrong\_J  
 3 thing\_N  
 2 possibly\_R  
 2 be\_V

2 not\_R  
 1 difference\_N  
 1 turn\_V  
 1 usually\_R  
 1 major\_J

1 impossible\_J  
 1 out\_R  
 1 repair\_V

## Conversion into 3-word windows...

Douglas Adams, *Mostly harmless*

major\_J difference\_N thing\_N go\_V wrong\_J thing\_N not\_R possibly\_R go\_V  
 wrong\_J be\_V thing\_N not\_R possibly\_R go\_V wrong\_J go\_V wrong\_J usually\_R  
 turn\_V out\_R be\_V impossible\_J get\_V repair\_N

- ∅ **major** difference
- major **difference** thing
- difference **thing** go
- thing **go** wrong
- ...

## Distribution for *wrong*

Douglas Adams, *Mostly harmless*

major\_J difference\_N thing\_N [go\_V wrong\_J thing\_N] not\_R possibly\_R [go\_V  
wrong\_J be\_V] thing\_N not\_R possibly\_R [go\_V wrong\_J [go\_V] wrong\_J usually\_R]  
turn\_V out\_R be\_V impossible\_J get\_V repair\_N

### • Distribution (frequencies):

5.0 go_V	0.0 possibly_R	0.0 impossible_J
1.0 thing_N	0.0 difference_N	0.0 out_R
1.0 usually_R	0.0 turn_V	0.0 repair_N
1.0 be_V	0.0 get_V	0.0 not_R
0.0 wrong_J	0.0 major_J	

## Distribution for *wrong*

Douglas Adams, *Mostly harmless*

major\_J difference\_N thing\_N [go\_V wrong\_J thing\_N] not\_R possibly\_R [go\_V  
wrong\_J be\_V] thing\_N not\_R possibly\_R [go\_V wrong\_J [go\_V] wrong\_J usually\_R]  
turn\_V out\_R be\_V impossible\_J get\_V repair\_N

### • Distribution (PPMIs):

0.748490106304 go_V	0.0 possibly_R	0.0 impossible_J
0.6221273278 usually_R	0.0 difference_N	0.0 out_R
0.229608686181 be_V	0.0 turn_V	0.0 repair_N
0.0 thing_N	0.0 get_V	0.0 not_R
0.0 wrong_J	0.0 major_J	

## The output of a DS system

- Some 'row' labels: the vocabulary of the system.
- Some 'column' labels: the contexts (in the case of a dimensionality-reduced space, the reduced dimensions don't have a label, though).
- The values at the intersection of rows and the columns form a matrix. The values of a row are the vector for a particular lexical item:

*wrong 0.748490106304 0.6221273278 0.229608686181 0.0 0.0  
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0*



# Dimensionality reduction

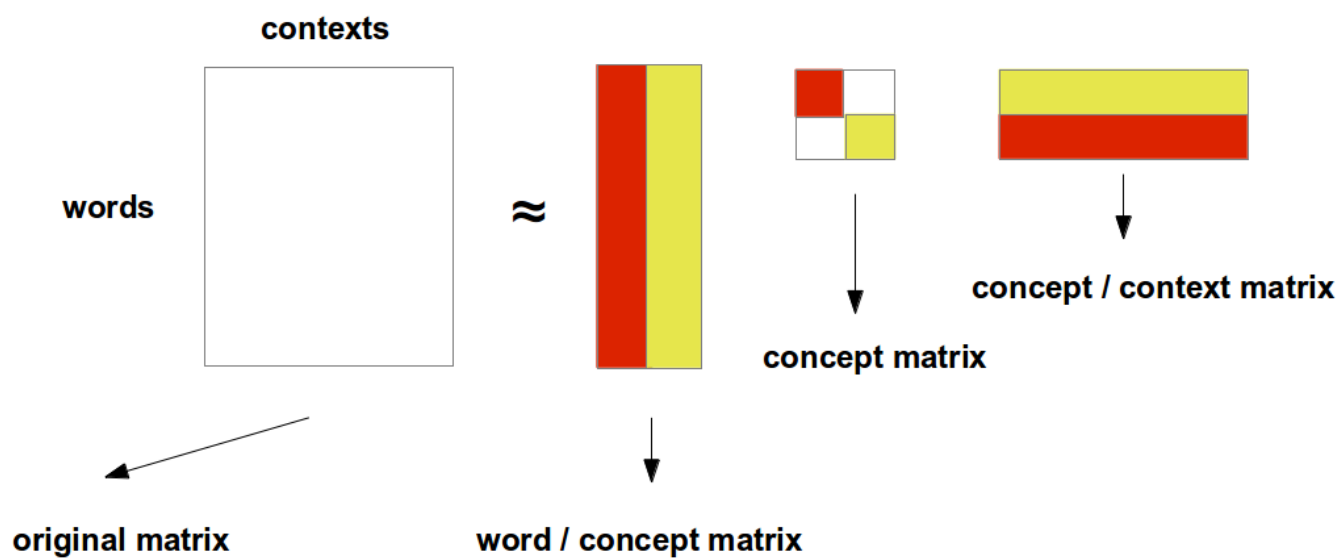
# Singular Value Decomposition

- SVD is a matrix factorisation method which expresses a matrix in terms of three other matrices:

$$A = U\Sigma V^T \quad (2)$$

- U and V are orthogonal: they are matrices such that
  - $UU^T = U^T U = I$
  - $VV^T = V^T V = I$
- $\Sigma$  is a diagonal matrix (only the diagonal entries are non-zero).

# Singular Value Decomposition



## The SVD derivation

- From our definition,  $A = U\Sigma V^T$ , it follows that...
- $A^T = V\Sigma^T U^T$
- $A^T A = V\Sigma^T U^T U \Sigma V^T = V\Sigma^2 V^T$   
(Recall that  $U^T U = I$  because  $U$  is orthogonal.)
- $A^T A V = V\Sigma^2$   
(The inverse  $V^{-1}$  of an orthogonal matrix is  $V^T$ , since  $V^T V = I$ .)
- Similarly,  $AA^T U = U\Sigma^2$ .

## SVD and eigenvectors

- An *eigenvector* of a linear transformation is a non-zero vector that doesn't change its direction when that linear transformation is applied to it:

$$Av = \lambda v \quad (3)$$

$v$  is the eigenvector,  $\lambda$  is the eigenvalue.

- Let's consider again the end of our derivation:  
 $A^T AV = V\Sigma^2$ .
- The columns of  $V$  are the eigenvectors of  $A^T A$ . (Similarly, the columns of  $U$  are the eigenvectors of  $AA^T$ .)

## The singular values of SVD

- If  $V$  contains the eigenvectors of  $A^T A$ ,  $\Sigma^2$  encapsulates its eigenvalues:  
 $A^T A V = V \Sigma^2$ .
- $\Sigma$  itself contains the square roots of the eigenvalues, also known as *singular values*.

## SVD at a glance

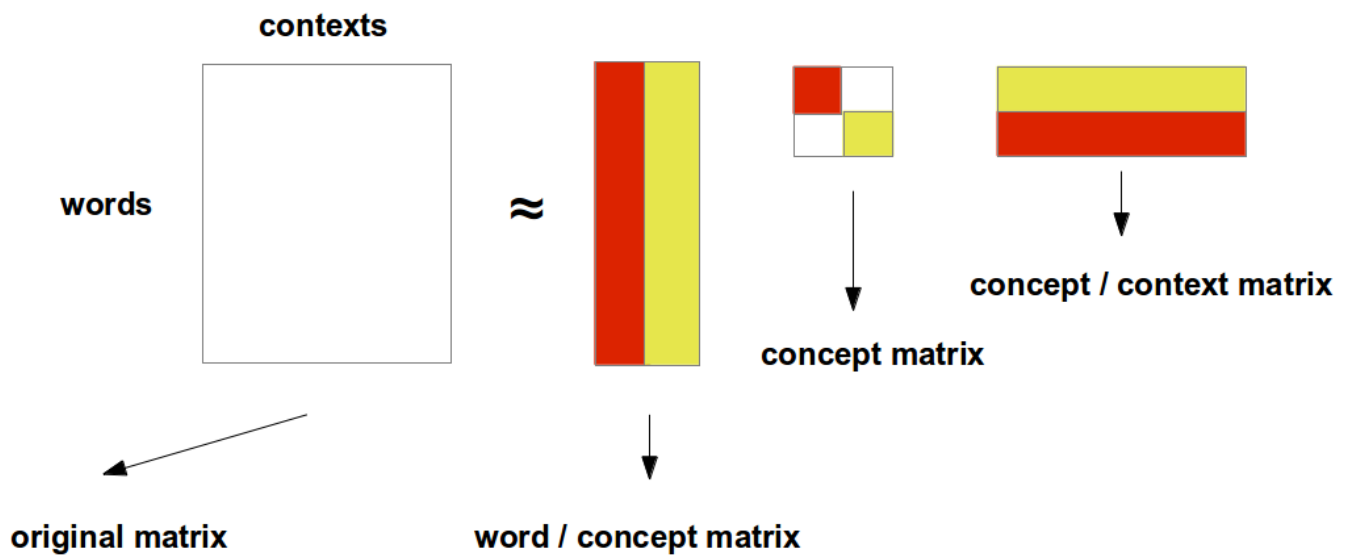
- Calculate  $A^T A$ .
- Calculate the eigenvalues of  $A^T A$  and sort them in descending order. Take their square roots to obtain the singular values of  $A^T A$  (i.e. the matrix  $\Sigma$ ).
- Use the eigenvalues to compute the eigenvectors of  $A^T A$ . These eigenvectors become the columns of  $V$ .
- Compute  $U = AV\Sigma^{-1}$ .

## Finally... dimensionality reduce!

- Now we know the value of  $U$ ,  $\Sigma$  and  $V$ .
- To obtain a reduced representation of  $A$ , choose the top  $k$  singular values in  $\Sigma$  and multiply the corresponding columns in  $U$  by those values.
- Example: the original matrix is  $10000 \times 5000$ . The SVD produces:
  - $U$ :  $10000 \times 10000$
  - $\Sigma$ :  $10000 \times 5000$
  - $V$ :  $5000 \times 5000$
- We can now take the first 300 singular value of  $\Sigma$ , and the corresponding 300 columns of  $U$ . We are now multiplying a matrix of  $10000 \times 300$  by a matrix of  $300 \times 300$ .



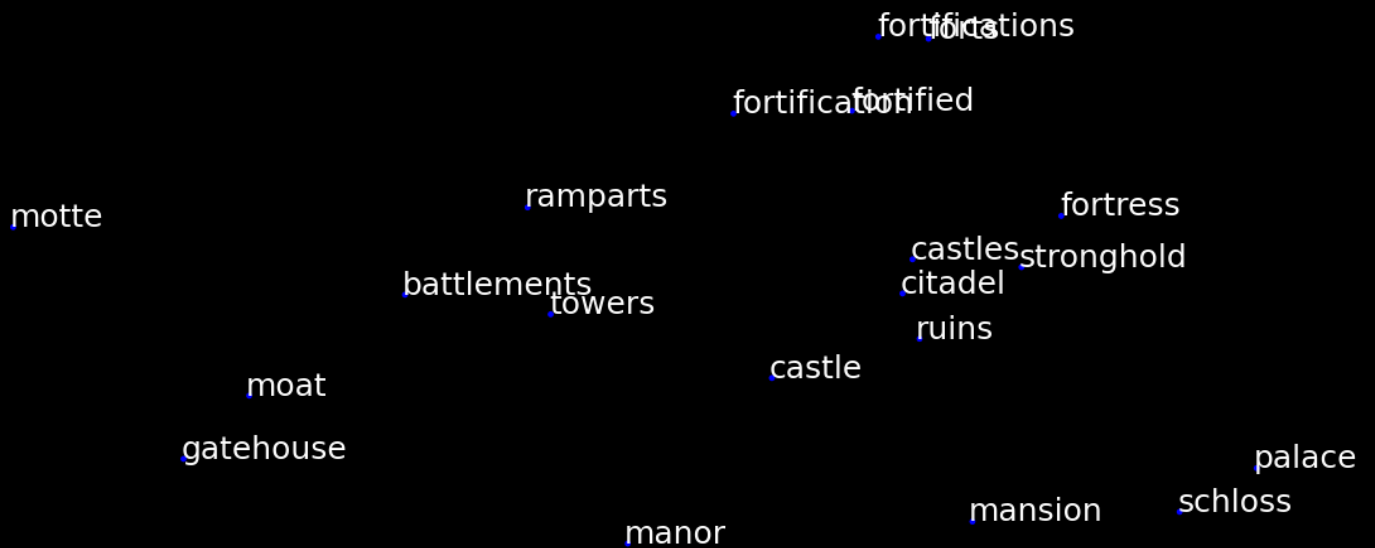
# Singular Value Decomposition



## What semantic space?

- Singular Value Decomposition (LSA – Landauer and Dumais, 1997). A new dimension might correspond to a generalisation over several of the original dimensions (e.g. the dimensions for *car* and *vehicle* are collapsed into one).
  - + Very efficient (200-500 dimensions). Captures generalisations in the data.
  - - SVD matrices are not straightforwardly interpretable.

# SVD for visualisation



# Evaluation

## Evaluating a semantic space

- How good is your semantic space?
- It depends on what you want it to be (i.e. which theory of meaning you are supporting.)
- So far, cognitive plausibility has been the main test: can we reproduce human linguistic judgements?

## Similarity-based evaluation

- Reproduce human similarity judgements (expressed as a score on a bounded scale).
- Rubenstein & Goodenough (1965): 65 noun pairs.
- Finkelstein et al (2002): WordSim353.
- Bruni et al (2014): MEN (1000 test pairs).
- Calculate spearman correlation ( $\rho$ ) between system results and human judgements. Current human correlation on the MEN dataset is well over 0.7.

# Similarity-based evaluation

## Human output

```
sun sunlight 50.000000
automobile car 50.000000
river water 49.000000
stair staircase 49.000000
...
green lantern 18.000000
painting work 18.000000
pigeon round 18.000000
...
muscle tulip 1.000000
bikini pizza 1.000000
bakery zebra 0.000000
```

## System output

```
stair staircase 0.913251552368
sun sunlight 0.727390960465
automobile car 0.740681924959
river water 0.501849324363
...
painting work 0.448091435945
green lantern 0.383044261062
...
bakery zebra 0.061804313745
bikini pizza 0.0561356056323
pigeon round 0.028243620524
muscle tulip 0.0142570835367
```

# Categorisation

- Cluster concepts into categories: e.g. *cat* and *giraffe* under ANIMAL, *car* and *motorcycle* under VEHICLE (Almuhareb 2006)
- Evaluated in terms of ‘purity’: if all the concepts in one automatically-produced cluster are from the same category, purity is 100%.



## Categorisation and purity

- Given clustered data, purity is defined as

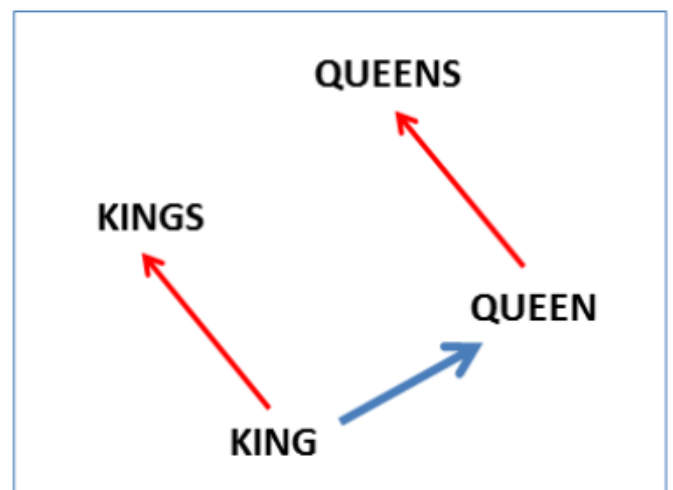
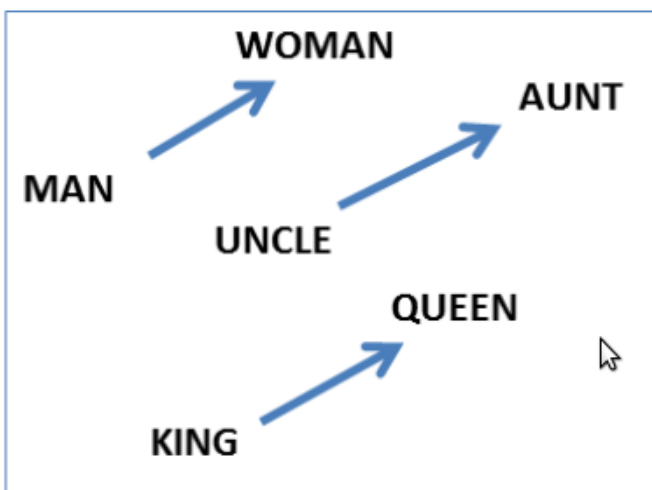
$$P(S_r) = \frac{1}{n_r} \max_i(n_r^i) \quad (4)$$

- Example: given the cluster  $S_1$  {A A A T A A A T T A}:

$$P(S_1) = \frac{1}{10} \max(7, 3) = 0.7$$

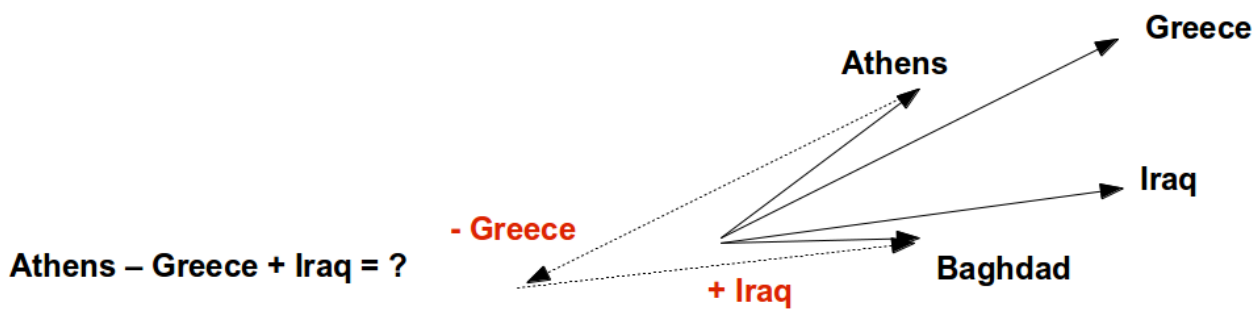
# Analogy

- Answer semantic and morphological analogy questions of the type *Rome is to Italy what Tokyo is to ...* (Mikolov et al 2013)
- Evaluated in terms of accuracy.



# Analogy

- Athens Greece Baghdad Iraq
- ...
- happy happily cheerful cheerfully
- ...
- decrease decreasing jump jumping
- ...



# The many faces of DS

# Distributional semantics in 2016

## **Linguistic representation:**

disambiguation, adjective semantics, quantifiers, phrasal composition, *meaning* of words.

## **Cognitive representation:**

simulates language acquisition, priming, fMRI measurements.

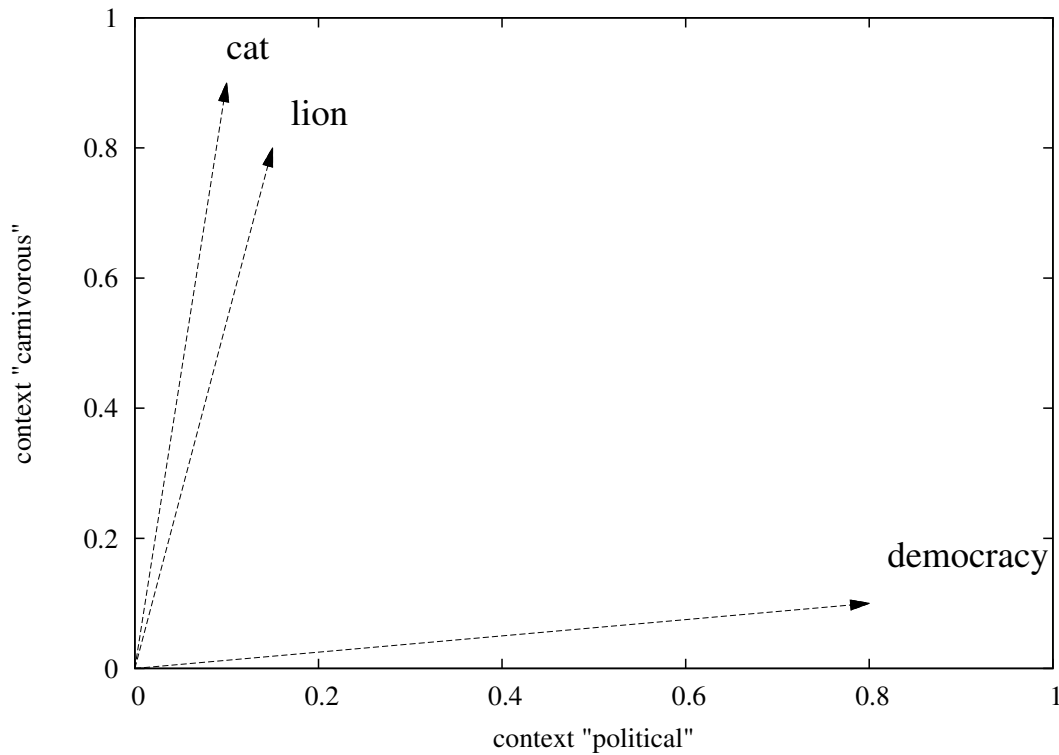
**Useful hack:** representation of the lexicon for NLP applications.

## A cognitive representation

- Landauer & Dumais (1997): knowledge acquisition.
- Lund, Burgess & Atchley (1995): priming.
- Anderson et al (2013): multimodal distributional representations simulate brain activation.

# Landauer & Dumais (1997)

- Implicit learning:



## Landauer & Dumais (1997)

- The semantic space is built via a dimensionality-reduced word-document matrix.
- Corpus: the Grolier's Academic American Encyclopedia.
- Evaluation: Test of English as a Foreign Language (TOEFL) – synonymy test:

**Stem:** levied

(a) imposed

(b) believed

(c) requested

(d) correlated

**Solution:** (a) imposed

- Best performance around 300 dimensions.



## Lund, Burgess & Atchley (1995)

- HAL: Hyperspace Analogue to Language.
- Priming: subjects are asked to recognise whether a string of letter is a word or not.
- Subjects' response time is faster if the target word is preceded by a similar item: *doctor/hospital* vs *doctor/kangaroo*.
- Priming effects can be simulated using similarity information from a semantic space.
- But: relatedness (*cradle/baby*) is not enough to produce priming effects.

## A linguistic representation (next week!)

- Account for the composition of short phrases: find a function  $f(\vec{u}, \vec{v})$  which returns the meaning of the composition of  $\vec{u}$  and  $\vec{v}$ .
- Sense disambiguation: re-weight a vector in context to get the various senses of the word it represents.
- Capture some inferential properties of language: if Molly is a cat, Molly is an animal, *many cats* entails *some cats*.
- Work on affixes, mass/count distinction, relative pronouns, negation, etc, etc.

# Conclusion

# Conclusion

- Distributional semantics is *one* possible semantic theory, which has experimental support – both in linguistics and cognitive science.
- Various models for distributional systems, with various consequences on the output.
- Known issues: corpus-dependence (which notion of concept is at play here?), word senses are collapsed (perhaps not such a bad thing...), fixed expressions create noise in the data.

# Conclusion

- Evaluation against psycholinguistic data shows that DS can model at least *some* phenomena.
- A powerful computational semantics tool, with surprising results.
- But a tool without a fully-fledged linguistic theory...