Distributional Semantic Models

Part 2: The parameters of a DSM

Stefan Evert[1]
with Alessandro Lenci[2], Marco Baroni[3] and Gabriella Lapesa[4]

[1]Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany
[2]University of Pisa, Italy
[3]University of Trento, Italy
[4]University of Stuttgart, Germany

http://wordspace.collocations.de/doku.php/course:start

Outline

Outline

Overview of DSM parameters

Term-context vs. term-term matrix

⇓

Definition of terms & linguistic pre-processing

⇓

Size & type of context

⇓

Geometric vs. probabilistic interpretation

⇓

Feature scaling

⇓

Normalisation of rows and/or columns

⇓

Similarity / distance measure

⇓

Dimensionality reduction

Term-context matrix

Term-context matrix records frequency of term in each individual context (e.g. sentence, document, Web page, encyclopaedia article)

$$\mathbf{F} = \begin{bmatrix} \cdots & \mathbf{f}_1 & \cdots \\ \cdots & \mathbf{f}_2 & \cdots \\ & \vdots & \\ & \vdots & \\ \cdots & \mathbf{f}_k & \cdots \end{bmatrix}$$

	Felidae	Pet	Feral	Bloat	Philosophy	Kant	Back pain
cat	10	10	7	–	–	–	–
dog	–	10	4	11	–	–	–
animal	2	15	10	2	–	–	–
time	1	–	–	–	2	1	–
reason	–	1	–	–	1	4	1
cause	–	–	–	2	1	2	6
effect	–	–	–	1	–	1	–

Term-context matrix

Some footnotes:

- Features are usually context tokens, i.e. individual instances
- Can also be generalised to context types, e.g.
  - bag of content words
  - specific pattern of POS tags
  - n-gram of words (or POS tags) around target
  - subcategorisation pattern of target verb
- Term-context matrix is often very **sparse**

Term-term matrix

Term-term matrix records co-occurrence frequencies with feature terms for each target term

$$\mathbf{M} = \begin{bmatrix} \cdots & \mathbf{m}_1 & \cdots \\ \cdots & \mathbf{m}_2 & \cdots \\ & \vdots & \\ & \vdots & \\ \cdots & \mathbf{m}_k & \cdots \end{bmatrix}$$

	breed	tail	feed	kill	important	explain	likely
cat	83	17	7	37	–	1	–
dog	561	13	30	60	1	2	4
animal	42	10	109	134	13	5	5
time	19	9	29	117	81	34	109
reason	1	–	2	14	68	140	47
cause	–	1	–	4	55	34	55
effect	–	–	1	6	60	35	17

☞ we will usually assume a term-term matrix in this tutorial

Term-term matrix

Some footnotes:

- Often target terms $\neq$ feature terms
  - e.g. nouns described by co-occurrences with verbs as features
  - identical sets of target & feature terms ➜ symmetric matrix
- Different types of contexts (Evert 2008)
  - **surface context** (word or character window)
  - **textual context** (non-overlapping segments)
  - **syntactic contxt** (specific syntagmatic relation)
- Can be seen as smoothing of term-context matrix
  - average over similar contexts (with same context terms)
  - data sparseness reduced, except for small windows
  - we will take a closer look at the relation between term-context and term-term models later in this tutorial

Overview of DSM parameters

Term-context vs. term-term matrix
⇓
Definition of terms & linguistic pre-processing
⇓
Size & type of context
⇓
Geometric vs. probabilistic interpretation
⇓
Feature scaling
⇓
Normalisation of rows and/or columns
⇓
Similarity / distance measure
⇓
Dimensionality reduction

Corpus pre-processing

- Minimally, corpus must be tokenised ➜ identify terms
- Linguistic annotation
  - part-of-speech tagging
  - lemmatisation / stemming
  - word sense disambiguation (rare)
  - shallow syntactic patterns
  - dependency parsing
- Generalisation of terms
  - often lemmatised to reduce data sparseness: *go, goes, went, gone, going* ➜ *go*
  - POS disambiguation (*light*/N vs. *light*/A vs. *light*/V)
  - word sense disambiguation ($bank_{\text{river}}$ vs. $bank_{\text{finance}}$)
- Trade-off between deeper linguistic analysis and
  - need for language-specific resources
  - possible errors introduced at each stage of the analysis

Effects of pre-processing

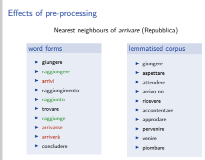
Nearest neighbours of *walk* (BNC)

word forms	lemmatised corpus
stroll	hurry
walking	stroll
walked	stride
go	trudge
path	amble
drive	wander
ride	walk-nn
wander	walking
sprinted	retrace
sauntered	scuttle

Effects of pre-processing

Nearest neighbours of *arrivare* (Repubblica)

word forms	lemmatised corpus
giungere	giungere
raggiungere	aspettare
arrivi	attendere
raggiungimento	arrivo-nn
raggiunto	ricevere
trovare	accontentare
raggiunge	approdare
arrivasse	pervenire
arriverà	venire
concludere	piombare



1. Colours seem to indicate inflected forms belonging to the same lemma.

Overview of DSM parameters

Term-context vs. term-term matrix
⇓
Definition of terms & linguistic pre-processing
⇓
Size & type of context
⇓
Geometric vs. probabilistic interpretation
⇓
Feature scaling
⇓
Normalisation of rows and/or columns
⇓
Similarity / distance measure
⇓
Dimensionality reduction

Surface context

Context term occurs within a window of k words around target.

The silhouette of the sun beyond a wide-open bay on the lake; the sun still glitters although evening has arrived in Kuhmo. It's midsummer; the living room has its instruments and other objects in each of its corners.

Parameters:

- window size (in words or characters)
- symmetric vs. one-sided window
- uniform or "triangular" (distance-based) weighting
- window clamped to sentences or other textual units?

Effect of different window sizes

Nearest neighbours of *dog* (BNC)

2-word window

- cat
- horse
- fox
- pet
- rabbit
- pig
- animal
- mongrel
- sheep
- pigeon

30-word window

- kennel
- puppy
- pet
- bitch
- terrier
- rottweiler
- canine
- cat
- to bark
- Alsatian

Textual context

Context term is in the same linguistic unit as target.

The silhouette of the sun beyond a wide-open bay on the lake; the sun still glitters although evening has arrived in Kuhmo. It's midsummer; the living room has its instruments and other objects in each of its corners.

Parameters:

- type of linguistic unit
  - sentence
  - paragraph
  - turn in a conversation
  - Web page

Syntactic context

Context term is linked to target by a syntactic dependency (e.g. subject, modifier, . . .).

The silhouette of the sun beyond a wide-open bay on the lake; the sun still glitters although evening has arrived in Kuhmo. It's midsummer; the living room has its instruments and other objects in each of its corners.

Parameters:

- types of syntactic dependency (Padó and Lapata 2007)
- direct vs. indirect dependency paths
  - direct dependencies
  - direct + indirect dependencies
- homogeneous data (e.g. only verb-object) vs. heterogeneous data (e.g. all children and parents of the verb)
- maximal length of dependency path

"Knowledge pattern" context

Context term is linked to target by a lexico-syntactic pattern (text mining, cf. Hearst 1992, Pantel & Pennacchiotti 2008, etc.).

In Provence, Van Gogh painted with bright colors such as red and yellow. These colors produce incredible effects on anybody looking at his paintings.

Parameters:

- inventory of lexical patterns
  - lots of research to identify semantically interesting patterns (cf. Almuhareb & Poesio 2004, Veale & Hao 2008, etc.)
- fixed vs. flexible patterns
  - patterns are mined from large corpora and automatically generalised (optional elements, POS tags or semantic classes)

Structured vs. unstructured context

- In **unstructered** models, context specification acts as a **filter**
  - determines whether context token counts as co-occurrence
  - e.g. linked by specific syntactic relation such as verb-object
- In **structured** models, context words are **subtyped**
  - depending on their position in the context
  - e.g. left vs. right context, type of syntactic relation, etc.

Structured vs. unstructured surface context

A dog bites a man. The man's dog bites a dog. A dog bites a man.

unstructured	bite
dog	4
man	3

A dog bites a man. The man's dog bites a dog. A dog bites a man.

structured	bite-l	bite-r
dog	3	1
man	1	2

Structured vs. unstructured dependency context

A dog bites a man. The man's dog bites a dog. A dog bites a man.

unstructured	bite
dog	4
man	2

A dog bites a man. The man's dog bites a dog. A dog bites a man.

structured	bite-subj	bite-obj
dog	3	1
man	0	2

Comparison

- Unstructured context
  - data less sparse (e.g. *man kills* and *kills man* both map to the *kill* dimension of the vector $\mathbf{x}_{\text{man}}$)
- Structured context
  - more sensitive to semantic distinctions (*kill-subj* and *kill-obj* are rather different things!)
  - dependency relations provide a form of syntactic "typing" of the DSM dimensions (the "subject" dimensions, the "recipient" dimensions, etc.)
  - important to account for word-order and compositionality

Overview of DSM parameters

Term-context vs. term-term matrix

⇓

Definition of terms & linguistic pre-processing

⇓

Size & type of context

⇓

Geometric vs. probabilistic interpretation

⇓

Feature scaling

⇓

Normalisation of rows and/or columns

⇓

Similarity / distance measure

⇓

Dimensionality reduction

Geometric vs. probabilistic interpretation

- Geometric interpretation
  - row vectors as points or arrows in n-dim. space
  - very intuitive, good for visualisation
  - use techniques from geometry and linear algebra
- Probabilistic interpretation
  - co-occurrence matrix as observed sample statistic
  - "explained" by generative probabilistic model
  - recent work focuses on hierarchical Bayesian models
  - probabilistic LSA (Hoffmann 1999), Latent Semantic Clustering (Rooth *et al.* 1999), Latent Dirichlet Allocation (Blei *et al.* 2003), etc.
  - explicitly accounts for random variation of frequency counts
  - intuitive and plausible as topic model

☞ focus on geometric interpretation in this tutorial

Overview of DSM parameters

Term-context vs. term-term matrix

⇓

Definition of terms & linguistic pre-processing

⇓

Size & type of context

⇓

Geometric vs. probabilistic interpretation

⇓

Feature scaling

⇓

Normalisation of rows and/or columns

⇓

Similarity / distance measure

⇓

Dimensionality reduction

Feature scaling

Feature scaling is used to "discount" less important features:

- Logarithmic scaling: $x' = \log(x+1)$
  (cf. Weber-Fechner law for human perception)
- Relevance weighting, e.g. tf.idf (information retrieval)
- Statistical **association measures** (Evert 2004, 2008) take frequency of target word and context feature into account
  - the less frequent the target word and (more importantly) the context feature are, the higher the weight given to their observed co-occurrence count should be (because their expected chance co-occurrence frequency is low)
  - different measures – e.g., mutual information, log-likelihood ratio – differ in how they balance observed and expected co-occurrence frequencies

Association measures: Mutual Information (MI)

word_1	word_2	f_{obs}	f_1	f_2
dog	*small*	855	33,338	490,580
dog	*domesticated*	29	33,338	918

Expected co-occurrence frequency:

$$f_{\text{exp}} = \frac{f_1 \cdot f_2}{N}$$

Mutual Information compares observed vs. expected frequency:

$$\text{MI}(w_1, w_2) = \log_2 \frac{f_{\text{obs}}}{f_{\text{exp}}} = \log_2 \frac{N \cdot f_{\text{obs}}}{f_1 \cdot f_2}$$

Disadvantage: MI overrates combinations of rare terms.

Other association measures

word_1	word_2	f_{obs}	f_{exp}	MI	local-MI	t-score
dog	*small*	855	134.34	2.67	2282.88	24.64
dog	*domesticated*	29	0.25	6.85	198.76	5.34
dog	*sgjkj*	1	0.00027	11.85	11.85	1.00

The log-likelihood ratio (Dunning 1993) has more complex form, but its "core" is known as local MI (Evert 2004).

$$\text{local-MI}(w_1, w_2) = f_{\text{obs}} \cdot \text{MI}(w_1, w_2)$$

The t-score measure (Church and Hanks 1990) is popular in lexicography:

$$\text{t-score}(w_1, w_2) = \frac{f_{\text{obs}} - f_{\text{exp}}}{\sqrt{f_{\text{obs}}}}$$

Details & many more measures: `http://www.collocations.de/`

Overview of DSM parameters

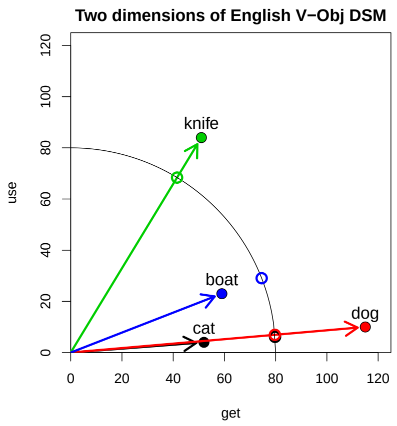
Term-context vs. term-term matrix
⇓
Definition of terms & linguistic pre-processing
⇓
Size & type of context
⇓
Geometric vs. probabilistic interpretation
⇓
Feature scaling
⇓
Normalisation of rows and/or columns
⇓
Similarity / distance measure
⇓
Dimensionality reduction

Normalisation of row vectors

- geometric distances only make sense if vectors are normalised to unit length
- divide vector by its length:

$$\mathbf{x}/\|\mathbf{x}\|$$

- normalisation depends on distance measure!
- special case: scale to relative frequencies with $\|\mathbf{x}\|_1 = |x_1| + \cdots + |x_n|$
  ➜ probabilistic interpretation



Scaling of column vectors

- In statistical analysis and machine learning, features are usually centred and scaled so that

$$\text{mean} \quad \mu = 0$$
$$\text{variance} \quad \sigma^2 = 1$$

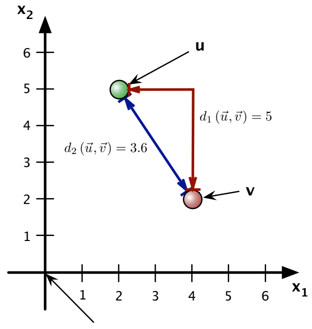
- In DSM research, this step is less common for columns of $\mathbf{M}$
  - centring is a prerequisite for certain dimensionality reduction and data analysis techniques (esp. PCA)
  - scaling may give too much weight to rare features
  - co-occurrence matrix no longer sparse after centring!
- $\mathbf{M}$ cannot be row-normalised and column-scaled at the same time (result depends on ordering of the two steps)

Overview of DSM parameters

Term-context vs. term-term matrix
⇓
Definition of terms & linguistic pre-processing
⇓
Size & type of context
⇓
Geometric vs. probabilistic interpretation
⇓
Feature scaling
⇓
Normalisation of rows and/or columns
⇓
Similarity / distance measure
⇓
Dimensionality reduction

Geometric distance

- **Distance** between vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ ➜ (dis)**similarity**
  - $\mathbf{u} = (u_1, \dots, u_n)$
  - $\mathbf{v} = (v_1, \dots, v_n)$
- **Euclidean** distance $d_2(\mathbf{u}, \mathbf{v})$
- "City block" **Manhattan** distance $d_1(\mathbf{u}, \mathbf{v})$
- Both are special cases of the **Minkowski** p-distance $d_p(\mathbf{u}, \mathbf{v})$ (for $p \in [1, \infty]$)

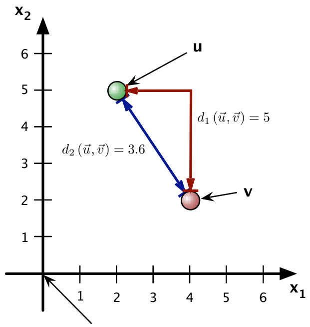


$$d_p(\mathbf{u}, \mathbf{v}) := \left(|u_1 - v_1|^p + \dots + |u_n - v_n|^p\right)^{1/p}$$

$$d_\infty(\mathbf{u}, \mathbf{v}) = \max\{|u_1 - v_1|, \dots, |u_n - v_n|\}$$

Geometric distance

- **Distance** between vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ ➜ (dis)**similarity**
  - $\mathbf{u} = (u_1, \dots, u_n)$
  - $\mathbf{v} = (v_1, \dots, v_n)$
- **Euclidean** distance $d_2(\mathbf{u}, \mathbf{v})$
- "City block" **Manhattan** distance $d_1(\mathbf{u}, \mathbf{v})$
- Extension of p-distance $d_p(\mathbf{u}, \mathbf{v})$ (for $0 \leq p \leq 1$)



$$d_p(\mathbf{u}, \mathbf{v}) := |u_1 - v_1|^p + \dots + |u_n - v_n|^p$$

$$d_0(\mathbf{u}, \mathbf{v}) = \#\{i \mid u_i \neq v_i\}$$

Metric: a measure of distance

- A **metric** is a general measure of the distance $d(\mathbf{u}, \mathbf{v})$ between points $\mathbf{u}$ and $\mathbf{v}$, which satisfies the following **axioms**:
  - $d(\mathbf{u}, \mathbf{v}) = d(\mathbf{v}, \mathbf{u})$
  - $d(\mathbf{u}, \mathbf{v}) > 0$ for $\mathbf{u} \neq \mathbf{v}$
  - $d(\mathbf{u}, \mathbf{u}) = 0$
  - $d(\mathbf{u}, \mathbf{w}) \leq d(\mathbf{u}, \mathbf{v}) + d(\mathbf{v}, \mathbf{w})$ (**triangle inequality**)
- Metrics form a very broad class of distance measures, some of which do not fit in well with our geometric intuitions
- E.g., metric need not be **translation-invariant**

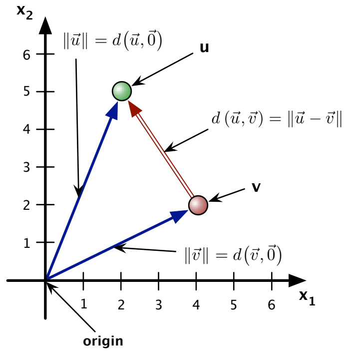
$$d(\mathbf{u} + \mathbf{x}, \mathbf{v} + \mathbf{x}) \neq d(\mathbf{u}, \mathbf{v})$$

- Another unintuitive example is the **discrete metric**

$$d(\mathbf{u}, \mathbf{v}) = \begin{cases} 0 & \mathbf{u} = \mathbf{v} \\ 1 & \mathbf{u} \neq \mathbf{v} \end{cases}$$

Distance vs. norm

- Intuitively, **distance** $d(\mathbf{u},\mathbf{v})$ should correspond to **length** $\|\mathbf{u}-\mathbf{v}\|$ of displacement vector $\mathbf{u}-\mathbf{v}$
  - $d(\mathbf{u},\mathbf{v})$ is a **metric**
  - $\|\mathbf{u}-\mathbf{v}\|$ is a **norm**
  - $\|\mathbf{u}\| = d(\mathbf{u},\mathbf{0})$
- Such a metric is always **translation-invariant**



- $d_p(\mathbf{u},\mathbf{v}) = \|\mathbf{v}-\mathbf{u}\|_p$
- **Minkowski p-norm** for $p \in [1,\infty]$ (not $p < 1$):

$$\|\mathbf{u}\|_p := \left(|u_1|^p + \cdots + |u_n|^p\right)^{1/p}$$

Norm: a measure of length

- A general **norm** $\|\mathbf{u}\|$ for the length of a vector $\mathbf{u}$ must satisfy the following **axioms**:
  - $\|\mathbf{u}\| > 0$ for $\mathbf{u} \neq \mathbf{0}$
  - $\|\lambda\mathbf{u}\| = |\lambda| \cdot \|\mathbf{u}\|$ (**homogeneity**, not req'd for metric)
  - $\|\mathbf{u}+\mathbf{v}\| \leq \|\mathbf{u}\| + \|\mathbf{v}\|$ (**triangle inequality**)
- every norm defines a translation-invariant metric

$$d(\mathbf{u},\mathbf{v}) := \|\mathbf{u}-\mathbf{v}\|$$

Other distance measures

- Information theory: **Kullback-Leibler** (KL) **divergence** for probability vectors (non-negative, $\|\mathbf{x}\|_1 = 1$)

$$D(\mathbf{u}\|\mathbf{v}) = \sum_{i=1}^{n} u_i \cdot \log_2 \frac{u_i}{v_i}$$

- Properties of KL divergence
  - most appropriate in a probabilistic interpretation of $\mathbf{M}$
  - zeroes in $\mathbf{v}$ without corresponding zeroes in $\mathbf{u}$ are problematic
  - not symmetric, unlike geometric distance measures
  - alternatives: skew divergence, Jensen-Shannon divergence
- A symmetric distance measure (Endres and Schindelin 2003)

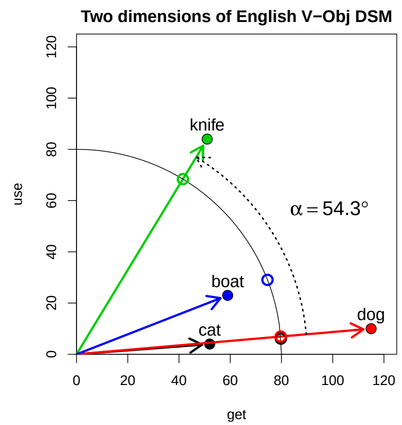
$$D_{\mathbf{uv}} = D(\mathbf{u}\|\mathbf{z}) + D(\mathbf{v}\|\mathbf{z}) \quad \text{with} \quad \mathbf{z} = \frac{\mathbf{u}+\mathbf{v}}{2}$$

Similarity measures

- angle α between two vectors $\mathbf{u},\mathbf{v}$ is given by

$$\cos\alpha = \frac{\sum_{i=1}^{n} u_i \cdot v_i}{\sqrt{\sum_i u_i^2} \cdot \sqrt{\sum_i v_i^2}} = \frac{\langle\mathbf{u},\mathbf{v}\rangle}{\|\mathbf{u}\|_2 \cdot \|\mathbf{v}\|_2}$$

- **cosine** measure of similarity: $\cos\alpha$
  - $\cos\alpha = 1$ ➜ collinear
  - $\cos\alpha = 0$ ➜ orthogonal
- distance metric: α



Euclidean distance or cosine similarity?

- Which is better, Euclidean distance or cosine similarity?
- They are equivalent: if vectors are normalised ($\|\mathbf{u}\|_2 = 1$), both lead to the same neighbour ranking

$$
\begin{aligned}
d_2(\mathbf{u},\mathbf{v}) &= \sqrt{\|\mathbf{u}-\mathbf{v}\|_2} = \sqrt{\langle \mathbf{u}-\mathbf{v}, \mathbf{u}-\mathbf{v}\rangle} \\
&= \sqrt{\langle \mathbf{u},\mathbf{u}\rangle + \langle \mathbf{v},\mathbf{v}\rangle - 2\,\langle \mathbf{u},\mathbf{v}\rangle} \\
&= \sqrt{\|\mathbf{u}\|_2 + \|\mathbf{v}\|_2 - 2\,\langle \mathbf{u},\mathbf{v}\rangle} \\
&= \sqrt{2 - 2\cos\phi}
\end{aligned}
$$

Overview of DSM parameters

Term-context vs. term-term matrix
⇓
Definition of terms & linguistic pre-processing
⇓
Size & type of context
⇓
Geometric vs. probabilistic interpretation
⇓
Feature scaling
⇓
Normalisation of rows and/or columns
⇓
Similarity / distance measure
⇓
Dimensionality reduction

Dimensionality reduction = model compression

- Co-occurrence matrix $\mathbf{M}$ is often unmanageably large and can be extremely sparse
  - Google Web1T5: 1M × 1M matrix with one trillion cells, of which less than 0.05% contain nonzero counts (Evert 2010)

➥ Compress matrix by reducing dimensionality (= rows)

- **Feature selection**: columns with high frequency & variance
  - measured by entropy, chi-squared test, . . .
  - may select correlated (➜ uninformative) dimensions
  - joint selection of multiple features is useful but expensive
- **Projection** into (linear) subspace
  - principal component analysis (PCA)
  - independent component analysis (ICA)
  - random indexing (RI)
  - ☞ intuition: preserve distances between data points

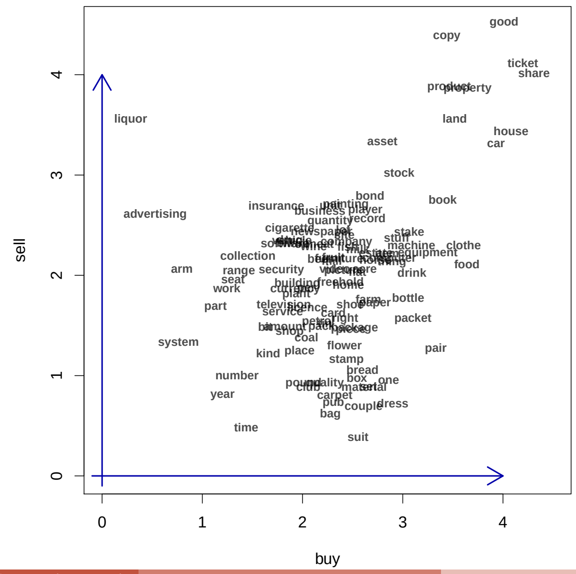
Dimensionality reduction & latent dimensions

Landauer and Dumais (1997) claim that LSA dimensionality reduction (and related PCA technique) uncovers **latent dimensions** by exploiting correlations between features.

- Example: term-term matrix
- V-Obj cooc's extracted from BNC
  - targets = noun lemmas
  - features = verb lemmas
- feature scaling: association scores (modified log Dice coefficient)
- $k = 111$ nouns with $f \geq 20$ (must have non-zero row vectors)
- $n = 2$ dimensions: *buy* and *sell*

noun	*buy*	*sell*
bond	0.28	0.77
cigarette	-0.52	0.44
dress	0.51	-1.30
freehold	-0.01	-0.08
land	1.13	1.54
number	-1.05	-1.02
per	-0.35	-0.16
pub	-0.08	-1.30
share	1.92	1.99
system	-1.63	-0.70

Dimensionality reduction & latent dimensions

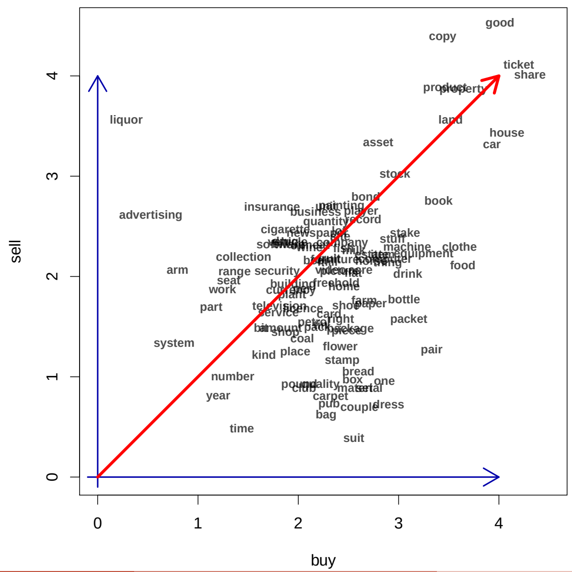


Motivating latent dimensions & subspace projection

- The **latent property** of being a commodity is "expressed" through associations with several verbs: *sell*, *buy*, *acquire*, . . .
- Consequence: these DSM dimensions will be **correlated**

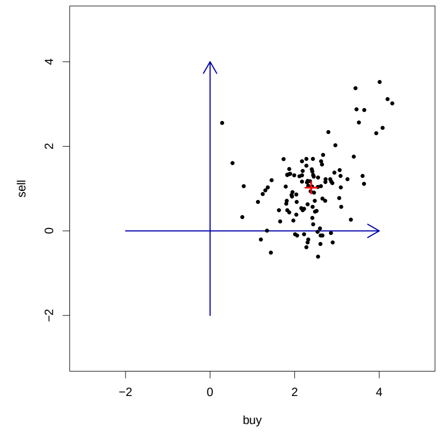
- Identify **latent dimension** by looking for strong correlations (or weaker correlations between large sets of features)
- Projection into subspace V of $k < n$ latent dimensions as a "**noise reduction**" technique ➜ **LSA**
- Assumptions of this approach:
  - "latent" distances in V are semantically meaningful
  - other "residual" dimensions represent chance co-occurrence patterns, often particular to the corpus underlying the DSM

The latent "commodity" dimension



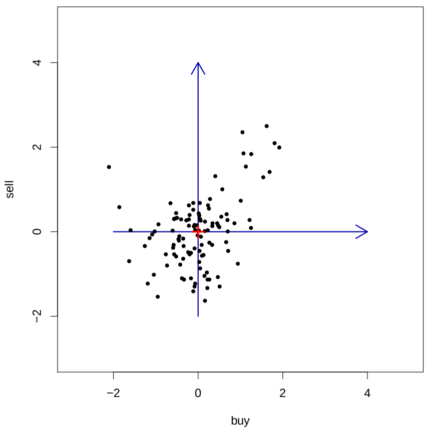
Centering the data set

- **Uncentered data set**
- Centered data set
- Variance of centered data



Centering the data set

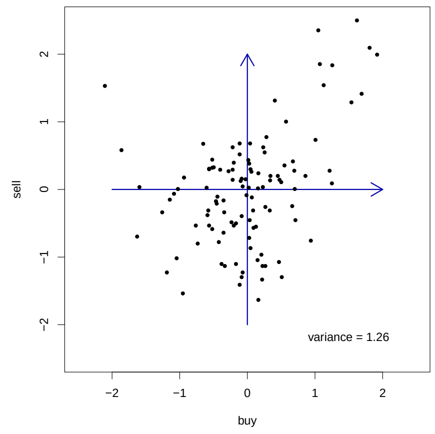
- Uncentered data set
- **Centered data set**
- Variance of centered data



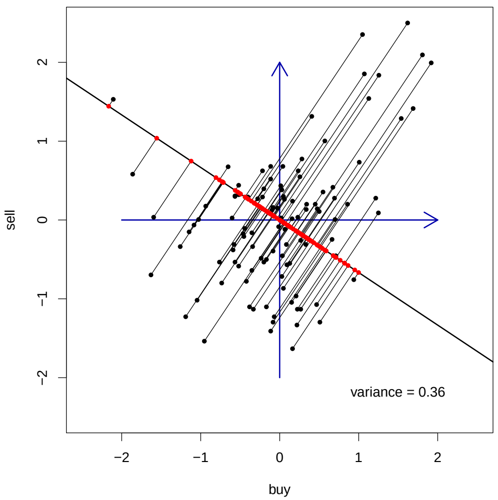
Centering the data set

- Uncentered data set
- Centered data set
- **Variance of centered data**

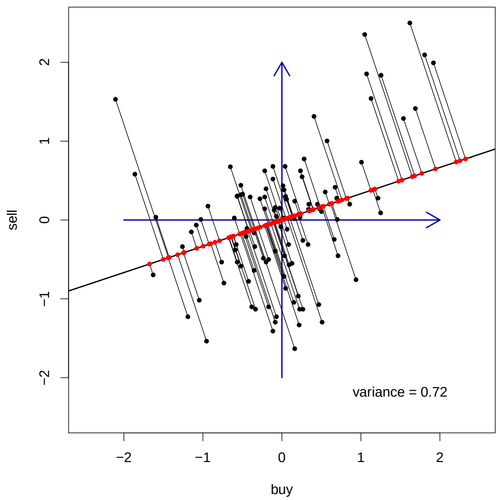
$$\sigma^2 = \frac{1}{k-1} \sum_{i=1}^{k} \|\mathbf{x}^{(i)}\|^2$$



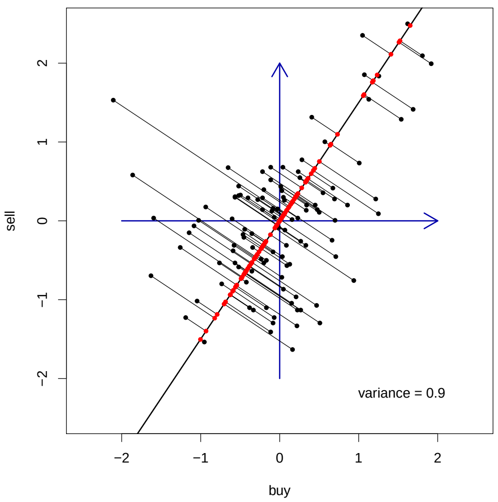
Projection and preserved variance: examples



Projection and preserved variance: examples

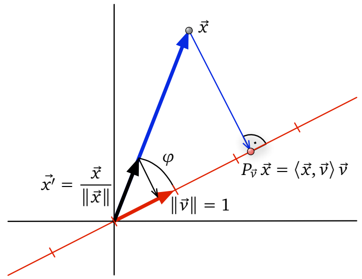


Projection and preserved variance: examples



The mathematics of projections

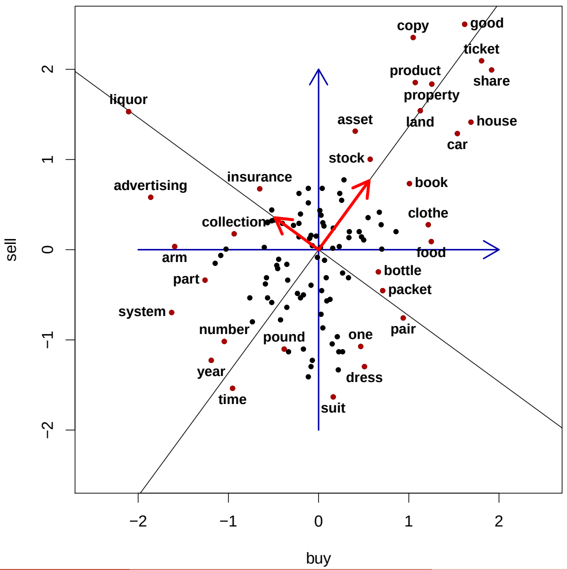
- Line through origin given by unit vector $\|\mathbf{v}\| = 1$
- For a point $\mathbf{x}$ and the corresponding unit vector $\mathbf{x}' = \mathbf{x}/\|\mathbf{x}\|$, we have $\cos\varphi = \langle \mathbf{x}', \mathbf{v}\rangle$



- Trigonometry: position of projected point on the line is $\|\mathbf{x}\| \cdot \cos\varphi = \|\mathbf{x}\| \cdot \langle \mathbf{x}', \mathbf{v}\rangle = \langle \mathbf{x}, \mathbf{v}\rangle$
- Preserved variance = one-dimensional variance on the line (note that data set is still centered after projection)

$$\sigma^2_{\mathbf{v}} = \frac{1}{k-1}\sum_{i=1}^{k} \langle \mathbf{x}_i, \mathbf{v}\rangle^2$$

PCA example



Outline

Some well-known DSM examples

Latent Semantic Analysis (Landauer and Dumais 1997)

- term-context matrix with document context
- weighting: log term frequency and term entropy
- distance measure: cosine
- dimensionality reduction: SVD

Hyperspace Analogue to Language (Lund and Burgess 1996)

- term-term matrix with surface context
- structured (left/right) and distance-weighted frequency counts
- distance measure: Minkowski metric ($1 \leq p \leq 2$)
- dimensionality reduction: feature selection (high variance)

Some well-known DSM examples

Infomap NLP (Widdows 2004)

- term-term matrix with unstructured surface context
- weighting: none
- distance measure: cosine
- dimensionality reduction: SVD

Random Indexing (Karlgren and Sahlgren 2001)

- term-term matrix with unstructured surface context
- weighting: various methods
- distance measure: various methods
- dimensionality reduction: random indexing (RI)

Some well-known DSM examples

Dependency Vectors (Padó and Lapata 2007)

- term-term matrix with unstructured dependency context
- weighting: log-likelihood ratio
- distance measure: information-theoretic (Lin 1998)
- dimensionality reduction: none

Distributional Memory (Baroni and Lenci 2010)

- term-term matrix with structured and unstructered dependencies + knowledge patterns
- weighting: local-MI on type frequencies of link patterns
- distance measure: cosine
- dimensionality reduction: none

Outline

Scaling up to the real world

- So far, we have worked on small **toy models**
  - DSM matrix restricted to 2,000 – 5,000 rows and columns
  - small corpora (or dependency sets) can be processed within **R**
- Now we need to scale up to **real world** data sets
  - for most statistical models, more data are better data!
  - cf. success of Google-based NLP techniques (even if simplistic)
- Example 1: window-based DSM on BNC content words
  - 83,926 lemma types with $f \geq 10$
  - term-term matrix with $83{,}926 \cdot 83{,}926 = 7$ billion entries
  - standard representation requires 56 GB of RAM (8-byte floats)
  - only 22.1 million non-zero entries ($= 0.32\%$)
- Example 2: Google Web 1T 5-grams (1 trillion words)
  - more than 1 million word types with $f \geq 2500$
  - term-term matrix with 1 trillion entries requires 8 TB RAM
  - only 400 million non-zero entries ($= 0.04\%$)

Handling large data sets: three approaches

1. Sparse matrix representation
   - full DSM matrix does not fit into memory
   - but much smaller number of non-zero entries can be handled
2. Feature selection
   - reduce DSM matrix to subset of columns (usu. 2,000 – 10,000)
   - select most frequent, salient, discriminative, . . . features
3. Dimensionality reduction
   - also reduces number of columns, but maps vectors to subspace
   - singular value decomposition (usu. ca. 300 dimensions)
   - random indexing (2,000 or more dimensions)
   - performed with external tools ➜ **R** can handle reduced matrix

Sparse matrix representation

- Invented example of a **sparsely populated** DSM matrix

	eat	get	hear	kill	see	use
boat	·	59	·	·	39	23
cat	·	·	·	26	58	·
cup	·	98	·	·	·	·
dog	33	·	42	·	83	·
knife	·	·	·	·	·	84
pig	9	·	·	27	·	·

- Store only non-zero entries in compact **sparse matrix format**

row	col	value	row	col	value
1	2	59	4	1	33
1	5	39	4	3	42
1	6	23	4	5	83
2	4	26	5	6	84
2	5	58	6	1	9
3	2	98	6	4	27

Working with sparse matrices

- Compressed format: each row index (or column index) stored only once, followed by non-zero entries in this row (or column)
  - convention: **column-major** matrix (data stored by columns)
- Specialised algorithms for sparse matrix algebra
  - especially matrix multiplication, solving linear systems, etc.
  - take care to avoid operations that create a dense matrix!
- **R** implementation: `Matrix` package (from CRAN)
  - can build sparse matrix from (row, column, value) table
  - unfortunately, no implementation of sparse SVD so far
- Other software packages: Matlab, Octave (recent versions)

References I

Baroni, Marco and Lenci, Alessandro (2010). Distributional Memory: A general framework for corpus-based semantics. *Computational Linguistics*, **36**(4), 673–712.

Blei, David M.; Ng, Andrew Y.; Jordan, Michael, I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, **3**, 993–1022.

Church, Kenneth W. and Hanks, Patrick (1990). Word association norms, mutual information, and lexicography. *Computational Linguistics*, **16**(1), 22–29.

Dunning, Ted E. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, **19**(1), 61–74.

Endres, Dominik M. and Schindelin, Johannes E. (2003). A new metric for probability distributions. *IEEE Transactions on Information Theory*, **49**(7), 1858–1860.

Evert, Stefan (2004). *The Statistics of Word Cooccurrences: Word Pairs and Collocations*. Dissertation, Institut für maschinelle Sprachverarbeitung, University of Stuttgart.

Evert, Stefan (2008). Corpora and collocations. In A. Lüdeling and M. Kytö (eds.), *Corpus Linguistics. An International Handbook*, chapter 58, pages 1212–1248. Mouton de Gruyter, Berlin, New York.

References II

Evert, Stefan (2010). Google Web 1T5 n-grams made easy (but not for the computer). In *Proceedings of the 6th Web as Corpus Workshop (WAC-6)*, pages 32–40, Los Angeles, CA.

Hoffmann, Thomas (1999). Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence (UAI'99)*.

Karlgren, Jussi and Sahlgren, Magnus (2001). From words to understanding. In Y. Uesaka, P. Kanerva, and H. Asoh (eds.), *Foundations of Real-World Intelligence*, chapter 294–308. CSLI Publications, Stanford.

Landauer, Thomas K. and Dumais, Susan T. (1997). A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, **104**(2), 211–240.

Lin, Dekang (1998). An information-theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning (ICML-98)*, pages 296–304, Madison, WI.

Lund, Kevin and Burgess, Curt (1996). Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments, & Computers*, **28**(2), 203–208.

Padó, Sebastian and Lapata, Mirella (2007). Dependency-based construction of semantic space models. *Computational Linguistics*, **33**(2), 161–199.

References III

Rooth, Mats; Riezler, Stefan; Prescher, Detlef; Carroll, Glenn; Beil, Franz (1999). Inducing a semantically annotated lexicon via EM-based clustering. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 104–111.

Widdows, Dominic (2004). *Geometry and Meaning*. Number 172 in CSLI Lecture Notes. CSLI Publications, Stanford.