Distributional Semantics

Marco Baroni and Gemma Boleda

CS 388: Natural Language Processing

Credits

- Many slides, ideas and tips from Alessandro Lenci and Stefan Evert
- See also:
  `http://wordspace.collocations.de/doku.php/course:esslli2009:start`

General introductions, surveys, overviews

- Susan Dumais. 2003. Data-driven approaches to information access. *Cognitive Science* 27:491–524
- Dominic Widdows. 2004. *Geometry and Meaning*. CSLI
- Magnus Sahlgren. 2006 *The Word-Space Model*. Stockholm University dissertation
- Alessandro Lenci. 2008. Distributional approaches in linguistic and cognitive research. *Italian Journal of Linguistics* 20(1): 1–31
- Marco Baroni and Alessandro Lenci. 2010. Distributional Memory: A general framework for corpus-based semantics. *Computational Linguistics* 36(4): 673–721
- Peter Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research* 37: 141–188
- Stephen Clark. In press. Vector space models of lexical meaning. In *Handbook of Contemporary Semantics, 2nd edition*
- Katrin Erk. In press. Vector space models of word meaning and phrase meaning: A survey. *Language and Linguistics Compass*

Outline

Introduction: The distributional hypothesis

Constructing the models

Semantic similarity as geometric distance

Evaluation

Multimodal distributional models
- Computer vision

Compositionality
- Why?
- How?

Conclusion

The distributional hypothesis

- The meaning of a word is the set of contexts in which it occurs in texts
- *Important aspects of* the meaning of a word *are a function of* (*can be approximated by*) the set of contexts in which it occurs in texts

The distributional hypothesis in real life

McDonald & Ramscar 2001

He filled the wampimuk, passed it around and we all drunk some

We found a little, hairy wampimuk sleeping behind the tree

Distributional lexical semantics

- Distributional analysis in structuralist linguistics (Zellig Harris), British corpus linguistics (J.R. Firth), psychology (Miller & Charles), but not only
- "[T]he semantic properties of a lexical item are fully reflected in appropriate aspects of the relations it contracts with actual and potential contexts [...] [T]here are good reasons for a principled limitation to linguistic contexts" (Cruse 1986)
- Distributional hypothesis suggests that we can induce (aspects of the) meaning of words from texts
- This is its biggest selling point in computational linguistics: it is a "theory of meaning" that can be easily operationalized into a procedure to extract "meaning" from text corpora on a large scale

The distributional hypothesis, weak and strong

Lenci (2008)

- Weak: a quantitative method for semantic analysis and lexical resource induction
- Strong: A cognitive hypothesis about the form and origin of semantic representations

Distributional semantic models (DSMs)

Narrowing the field

- Idea of using corpus-based statistics to extract information about semantic properties of words and other linguistic units is extremely common in computational linguistics
- Here, we focus on models that:
  - Represent the meaning of words as *vectors* keeping track of the words' distributional history
  - Focus on the notion of *semantic similarity*, measured with geometrical methods in the *space* inhabited by the distributional vectors
  - Are intended as *general-purpose* semantic models that are estimated once, and then used for various semantic tasks, and not created ad-hoc for a specific goal
    - It follows that model estimation phase is typically unsupervised
- E.g.: LSA (Landauer & Dumais 1997), HAL (Lund & Burgess 1996), Schütze (1997), Sahlgren (2006), Padó & Lapata (2007), Baroni and Lenci (2010)
- Aka: vector/word space models, semantic spaces

Advantages of distributional semantics

Distributional semantic models are

- model of inductive *learning* for word meaning
- radically empirical
- rich
- flexible
- cheap, scalable

Outline

Introduction: The distributional hypothesis

Constructing the models

Semantic similarity as geometric distance

Evaluation

Multimodal distributional models
- Computer vision

Compositionality
- Why?
- How?

Conclusion

Constructing the models

- Pre-process the source corpus
- Collect a co-occurrence matrix (with *distributional vectors* representing words as rows, and contextual elements of some kind as columns/dimensions)
- Transform the matrix: re-weighting raw frequencies, dimensionality reduction
- Use resulting matrix to compute word-to-word similarity

Corpus pre-processing

- Minimally, corpus must be tokenized
- POS tagging, lemmatization, dependency parsing...
- Trade-off between deeper linguistic analysis and
  - need for language-specific resources
  - possible errors introduced at each stage of the analysis
  - more parameters to tune

Distributional vectors

- Count how many times each target word occurs in a certain context
- Build vectors out of (a function of) these context occurrence counts
- Similar words will have similar vectors

Collecting context counts for target word dog

The dog barked in the park.
The owner of the dog put him
on the leash since he barked.

bark	++
park	+
owner	+
leash	+

Collecting context counts for target word dog

The dog barked in the park.
The owner of the dog put him
on the leash since he barked.

bark	++
park	+
owner	+
leash	+

Collecting context counts for target word dog

The dog barked in the park.
The owner of the dog put him
on the leash since he barked.

bark	++
park	+
owner	+
leash	+

Collecting context counts for target word dog

The dog barked in the park.
The owner of the dog put him
on the leash since he barked.

bark	++
park	+
owner	+
leash	+

Collecting context counts for target word dog

The dog barked in the park.
The owner of the dog put him
on the leash since he barked.

bark	++
park	+
owner	+
leash	+

Collecting context counts for target word dog

The dog barked in the park.
The owner of the dog put him
on the leash since he barked.

bark	++
park	+
owner	+
leash	+

The co-occurrence matrix

	leash	walk	run	owner	pet	bark
dog	3	5	2	5	3	2
cat	0	3	3	2	3	0
lion	0	3	2	0	1	0
light	0	0	0	0	0	0
bark	1	0	0	2	1	0
car	0	0	1	3	0	0

What is “context”?

DOC1: 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.

What is “context”?

Documents

DOC1: 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.

What is “context”?

All words in a wide window

DOC1: 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.

What is “context”?

Content words only

DOC1: 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.

What is “context”?

Content words in a narrower window

DOC1: 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.

What is “context”?

POS-coded content lemmas

DOC1: The silhouette-n of the sun beyond a wide-open-a bay-n on the lake-n; the sun still glitter-v although evening-n has arrive-v in Kuhmo. It’s midsummer; the living room has its instruments and other objects in each of its corners.

What is "context"?

POS-coded content lemmas filtered by syntactic path to the target

DOC1: The silhouette-n of the sun beyond a wide-open bay on the lake; the sun still glitter-v although evening has arrived in Kuhmo. It's midsummer; the living room has its instruments and other objects in each of its corners.

What is “context”?

. . . with the syntactic path encoded as part of the context

DOC1: The silhouette-n_ppdep of the sun beyond a wide-open bay on the lake; the sun still glitter-v_subj although evening has arrived in Kuhmo. It’s midsummer; the living room has its instruments and other objects in each of its corners.

Same corpus (BNC), different contexts (window sizes)

Nearest neighbours of *dog*

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

General trends in “context engineering”

- In computational linguistics, tendency towards using more linguistically aware contexts, but “jury is still out” on their utility (Sahlgren, 2008)
  - This is at least in part task-specific
- In cognitive science trend towards broader document-/text-based contexts
  - Focus on topic detection, gist extraction, text coherence assessment, library science
  - Latent Semantic Analysis (Landauer & Dumais, 1997), Topic Models (Griffiths et al., 2007)

Contexts and dimensions

Some terminology I will use below

- Dependency-filtered (e.g., Padó & Lapata, 2007) vs. dependency-linked (e.g., Grefenstette 1994, Lin 1998, Curran & Moens 2002, Baroni and Lenci 2010)
- Both rely on output of dependency parser to identify context words that are connected to target words by interesting relations
- However, only dependency-linked models keep (parts of) the dependency path connecting target word and context word in the dimension label

Contexts and dimensions

Some terminology I will use below

- Given input sentence: *The dog bites the postman on the street*
- both approaches might consider only *bite* as a context element for both *dog* and *postman* (because they might focus on *subj-of* and *obj-of* relations only)
- However, a dependency-filtered model will count *bite* as identical context in both cases
- whereas a dependency-linked model will count *subj-of-bite* as context of *dog* and *obj-of-bite* as context of *postman* (so, *different* contexts for the two words)

Context beyond corpora and language

- The distributional semantic framework is general enough that feature vectors can come from other sources as well, besides from corpora (or from a mixture of sources)
- Obvious alternative/complementary sources are dictionaries, structured knowledge bases such as WordNet
- I am particularly interested in the possibility of merging features from text and images ("visual words": Feng and Lapata 2010, Bruni et al. 2011, 2012)

Context weighting

- Raw context counts typically transformed into scores
- In particular, association measures to give more weight to contexts that are more significantly associated with a target word
- General idea: the less frequent the target word and (more importantly) the context element are, the higher the weight given to their observed co-occurrence count should be (because their expected chance co-occurrence frequency is low)
  - Co-occurrence with frequent context element *time* is less informative than co-occurrence with rarer *tail*
- Different measures – e.g., Mutual Information, Log Likelihood Ratio – differ with respect to how they balance raw and expectation-adjusted co-occurrence frequencies
  - Positive Point-wise Mutual Information widely used and pretty robust

Context weighting

- Measures from information retrieval that take distribution over documents into account are also used
  - Basic idea is that terms that tend to occur in a few documents are more interesting than generic terms that occur all over the place

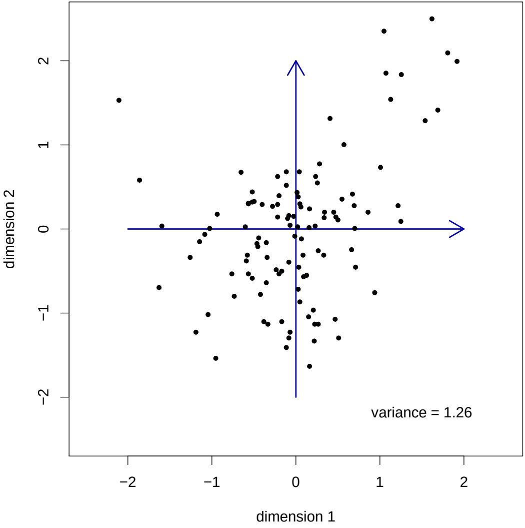
Dimensionality reduction

- Reduce the target-word-by-context matrix to a lower dimensionality matrix (a matrix with less – linearly independent – columns/dimensions)
- Two main reasons:
  - Smoothing: capture “latent dimensions” that generalize over sparser surface dimensions (Singular Value Decomposition or SVD)
  - Efficiency/space: sometimes the matrix is so large that you don’t even want to construct it explicitly (Random Indexing)

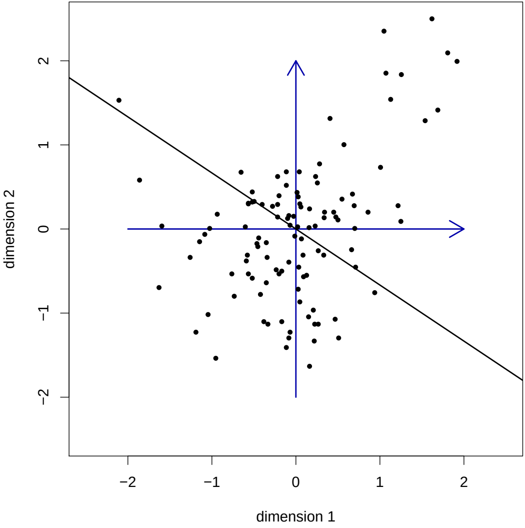
Singular Value Decomposition

- General technique from linear algebra (essentially, the same as Principal Component Analysis, PCA)
  - Some alternatives: Independent Component Analysis, Non-negative Matrix Factorization
- Given a matrix (e.g., a word-by-context matrix) of $m \times n$ dimensionality, construct a $m \times k$ matrix, where $k << n$ (and $k < m$)
  - E.g., from a 20,000 words by 10,000 contexts matrix to a 20,000 words by 300 "latent dimensions" matrix
  - k is typically an arbitrary choice
- From linear algebra, we know that and how we can find the reduced $m \times k$ matrix with orthogonal dimensions/columns that preserves most of the variance in the original matrix

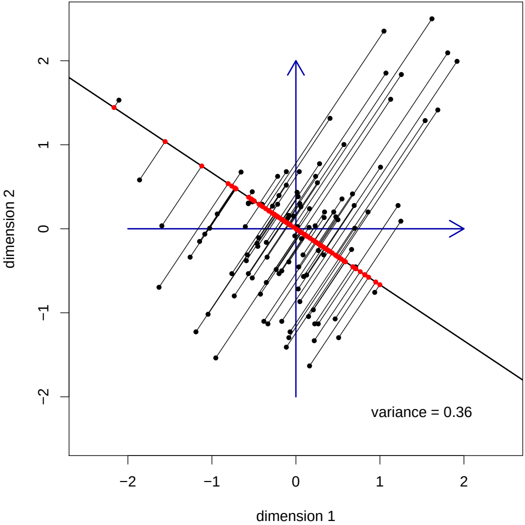
Preserving variance



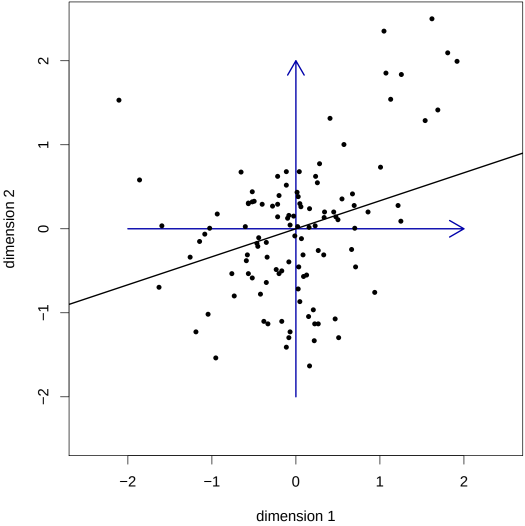
Preserving variance



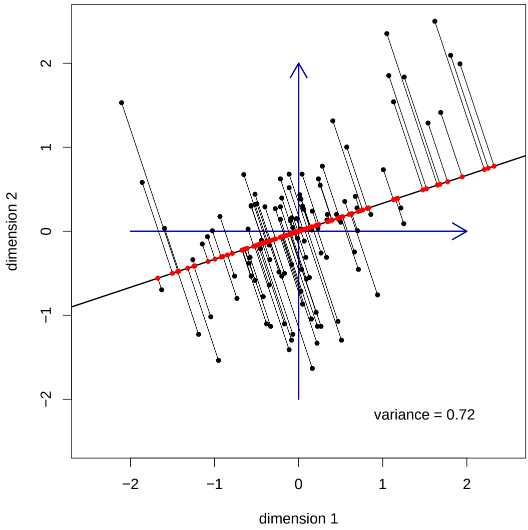
Preserving variance



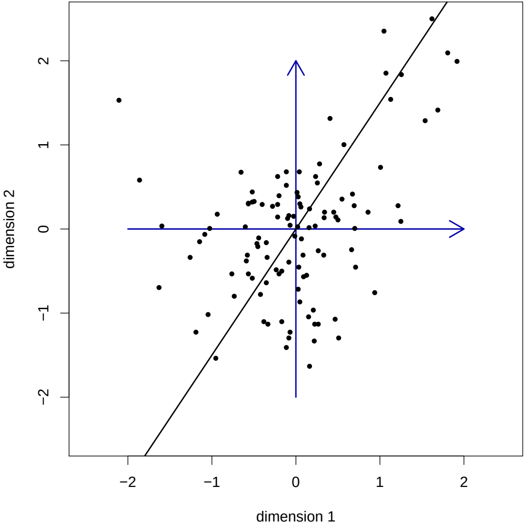
Preserving variance



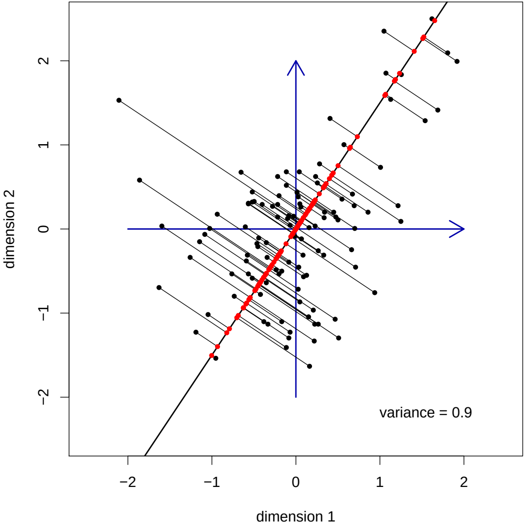
Preserving variance



Preserving variance



Preserving variance



Dimensionality reduction as generalization

	buy	sell	dim1
wine	31.2	27.3	41.3
beer	15.4	16.2	22.3
car	40.5	39.3	56.4
cocaine	3.2	22.3	18.3

The Singular Value Decomposition

- Any $m \times n$ real-valued matrix A can be factorized into 3 matrices $U\Sigma V^T$
- U is a $m \times m$ orthogonal matrix ($UU^T = I$)
- Σ is a $m \times n$ diagonal matrix, with diagonal values ordered from largest to smallest ($\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r \geq 0$, where $r = \min(m, n)$)
- V is a $n \times n$ orthogonal matrix ($VV^T = I$)

The Singular Value Decomposition

$$\begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1m} \\ u_{21} & u_{22} & \cdots & u_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ u_{m1} & u_{m2} & \cdots & u_{mm} \end{pmatrix} \times \begin{pmatrix} \sigma_1 & 0 & 0 & \cdots \\ 0 & \sigma_2 & 0 & \cdots \\ 0 & 0 & \sigma_3 & \cdots \\ \cdots & \cdots & \cdots & \cdots \end{pmatrix} \times \begin{pmatrix} v_{11} & v_{21} & \cdots & v_{n1} \\ v_{12} & v_{22} & \cdots & v_{n2} \\ \cdots & \cdots & \cdots & \cdots \\ v_{1n} & v_{2n} & \cdots & v_{nn} \end{pmatrix}$$

The Singular Value Decomposition

Projecting the A row vectors onto the new coordinate system

$$A_{m\times n} = U_{m\times m}\Sigma_{m\times n}V^T_{n\times n}$$

- The columns of the orthogonal $V_{n\times n}$ matrix constitute a *basis* (coordinate system, set of axes or dimensions) for the n-dimensional row vectors of A
- The projection of a row vector a_j onto axis column v_i (i.e., the v_i coordinate of a_j) is given by $a_j \cdot v_i$
- The coordinates of a_j in the full V coordinate system are thus given by $a_j V$, and generalizing the coordinates of all vectors projected onto the new system are given by AV
- $AV = U\Sigma V^T V = U\Sigma$

Reducing dimensionality

- Projecting A onto the new V coordinate system:

$$AV = U\Sigma$$

- It can be shown that, when the A row vectors are represented in this new set of coordinates, variance on each v_i-axis is proportional to σ_i^2 (the square of the i-th value on the diagonal of Σ)
  - Intuitively: U and V are orthogonal, all the "stretching" when multiplying the matrices is done by Σ
- Given that $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r \geq 0$, if we take the coordinates on the first k axes, we obtain lower dimensionality vectors that account for the maximum proportion of the original variance that we can account for with k dimensions
- I.e., we compute the "truncated" projection:

$$A_{m\times n}V_{n\times k} = U_{m\times k}\Sigma_{k\times k}$$

The Singular Value Decomposition

Finding the component matrices

- Don't try this at home!
- SVD draw on non-efficient operations
- Fortunately, there are out-of-the-box packages to compute SVD, a popular one being SVDPACK, that I use via SVDLIBC (`http://tedlab.mit.edu/~dr/svdlibc/`)
- Recently, various mathematical developments and packages to compute SVD incrementally, scaling up to very very large matrices, see e.g.: `http://radimrehurek.com/gensim/`
- See: `http://wordspace.collocations.de/doku.php/course:esslli2009:start`
- Very clear introduction to SVD (and PCA), with all the mathematical details I skipped here

SVD: Pros and cons

- Pros:
  - Good performance (in most cases)
  - At least some indication of robustness against data sparseness
  - Smoothing as generalization
  - Smoothing also useful to generalize features to words that do not co-occur with them in the corpus (e.g., spreading visually-derived features to all words)
  - Words and contexts in the same space (contexts not trivially orthogonal to each other)
- Cons:
  - Non-incremental (even incremental implementations allow you to add new rows, not new columns)
    - Of course, you can use $V_{n \times k}$ to project new vectors onto the same reduced space!
  - Latent dimensions are difficult to interpret
  - Does not scale up well (but see recent developments. . .)

Outline

Introduction: The distributional hypothesis

Constructing the models

Semantic similarity as geometric distance

Evaluation

Multimodal distributional models
- Computer vision

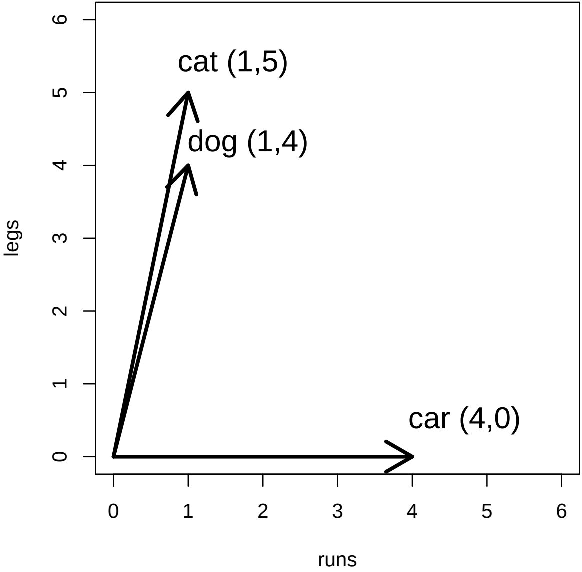
Compositionality
- Why?
- How?

Conclusion

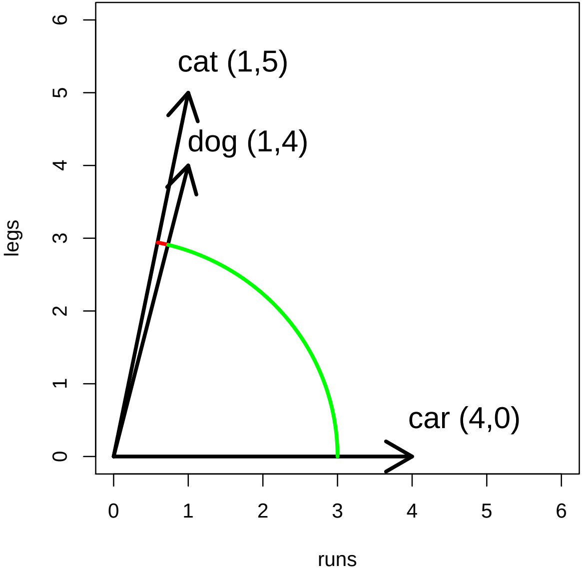
Contexts as vectors

	runs	legs
dog	1	4
cat	1	5
car	4	0

Semantic space



Semantic similarity as angle between vectors



Measuring angles by computing cosines

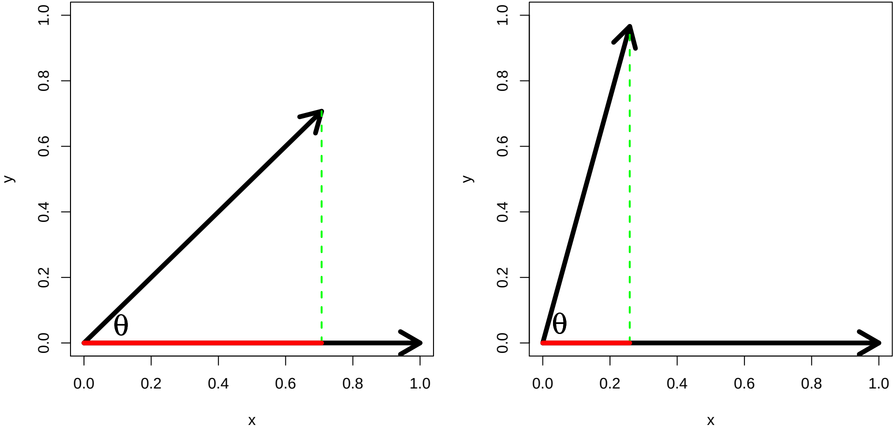
- Cosine is most common similarity measure in distributional semantics, and the most sensible one from a geometrical point of view
- Ranges from 1 for parallel vectors (perfectly correlated words) to 0 for orthogonal (perpendicular) words/vectors
  - It goes to -1 for parallel vectors pointing in opposite directions (perfectly inversely correlated words), as long as weighted co-occurrence matrix has negative values
- (Angle is obtained from cosine by applying the *arc-cosine* function, but it is rarely used in computational linguistics)

Trigonometry review

- Build a right triangle by connecting the two vectors
- Cosine is ratio of length of side adjacent to measured angle to length of hypotenuse side
- If we build triangle so that hypotenuse has length 1, cosine will equal length of adjacent side (because we divide by 1)
- I.e., in this case cosine is length of *projection* of hypotenuse on the adjacent side

Computing the cosines: preliminaries

Length and dot products



- Length of a vector **v** with n dimensions $v_1, v_2, \ldots, v_n$ (Pythagoras' theorem!):

$$\sqrt{i=n}$$

Computing the cosines: preliminaries

Orthogonal vectors

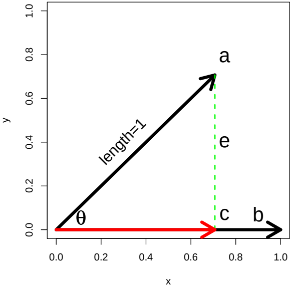
- The dot product of two orthogonal (perpendicular) vectors is 0
- To see this, note that given two vectors $\mathbf{v}$ and $\mathbf{w}$ forming a right angle, Pythagoras' theorem says that $||\mathbf{v}||^2 + ||\mathbf{w}||^2 = ||\mathbf{v} - \mathbf{w}||^2$
- But:

$$||\mathbf{v} - \mathbf{w}||^2 = \sum_{i=1}^{i=n} (v_i - w_i)^2 = \sum_{i=1}^{i=n} (v_i^2 - 2v_i w_i + w_i^2) =$$

$$\sum_{i=1}^{i=n} v_i^2 - \sum_{i=1}^{i=n} 2v_i w_i + \sum_{i=1}^{i=n} w_i^2 = ||\mathbf{v}||^2 - 2\mathbf{v} \cdot \mathbf{w} + ||\mathbf{w}||^2$$

- So, for the Pythagoras' theorem equality to hold, $\mathbf{v} \cdot \mathbf{w} = 0$

Computing the cosine



- $||\mathbf{a}|| = ||\mathbf{b}|| = 1$
- $\mathbf{c} = p\,\mathbf{b}$
- $\mathbf{e} = \mathbf{c} - \mathbf{a}$; $\mathbf{e} \cdot \mathbf{b} = 0$
- $(\mathbf{c} - \mathbf{a}) \cdot \mathbf{b} = \mathbf{c} \cdot \mathbf{b} - \mathbf{a} \cdot \mathbf{b} = 0$
- $\mathbf{c} \cdot \mathbf{b} = p\,\mathbf{b} \cdot \mathbf{b} = p = \mathbf{a} \cdot \mathbf{b}$
- $||\mathbf{c}|| = ||p\,\mathbf{b}|| = \sqrt{p^2\,\mathbf{b} \cdot \mathbf{b}} = p = \mathbf{a} \cdot \mathbf{b}$

Computing the cosine

- For two vectors of length 1, the cosine is given by: $||\mathbf{c}|| = \mathbf{a} \cdot \mathbf{b}$
- If the two vectors are not of length 1 (as it will be typically the case in DSMs), we obtain vectors of length 1 pointing in the same directions by dividing the original vectors by their lengths, obtaining:

$$||\mathbf{c}|| = \frac{\mathbf{a} \cdot \mathbf{b}}{||\mathbf{a}||||\mathbf{b}||} = \frac{\sum_{i=1}^{i=n} a_i \times b_i}{\sqrt{\sum_{i=1}^{i=n} a^2} \times \sqrt{\sum_{i=1}^{i=n} b^2}}$$

Computing the cosine

Example

$$\frac{\sum_{i=1}^{i=n} a_i \times b_i}{\sqrt{\sum_{i=1}^{i=n} a^2} \times \sqrt{\sum_{i=1}^{i=n} b^2}}$$

	runs	legs
dog	1	4
cat	1	5
car	4	0

$$\text{cosine}(\textit{dog},\textit{cat}) = \frac{(1\times1)+(4\times5)}{\sqrt{1^2+4^2}\times\sqrt{1^2+5^2}} = 0.9988681$$

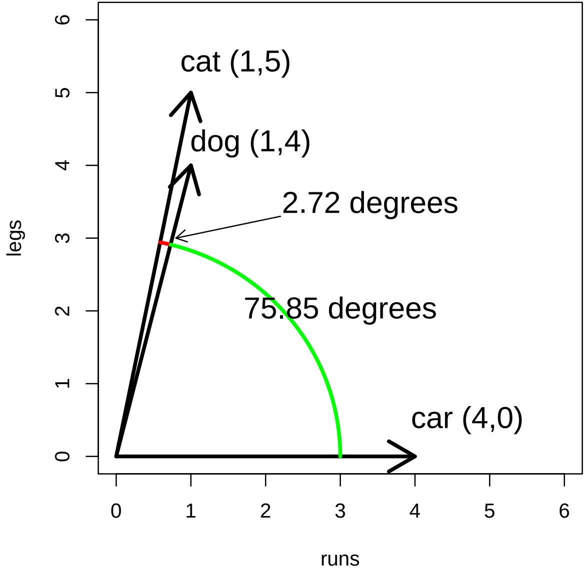
arc-cosine(0.9988681) = 2.72 degrees

$$\text{cosine}(\textit{dog},\textit{car}) = \frac{(1\times4)+(4\times0)}{\sqrt{1^2+4^2}\times\sqrt{4^2+0^2}} = 0.2425356$$

arc-cosine(0.2425356) = 75.85 degrees

Computing the cosine

Example



Cosine intuition

- When computing the cosine, the values that two vectors have for the same dimensions (coordinates) are multiplied
- Two vectors/words will have a high cosine if they tend to have high same-sign values for the same dimensions/contexts
- If we center the vectors so that their mean value is 0, the cosine of the centered vectors is the same as the Pearson correlation coefficient
- If, as it is often the case in computational linguistics, we have only nonnegative scores, and we do not center the vectors, then the cosine can only take nonnegative values, and there is no “canceling out” effect
  - As a consequence, cosines tend to be higher than the corresponding correlation coefficients

Other measures

- Cosines are well-defined, well understood way to measure similarity in a vector space
- Euclidean distance (length of segment connecting end-points of vectors) is equally principled, but length-sensitive (two vectors pointing in the same direction will be very distant if one is very long, the other very short)
- Other measures based on other, often non-geometric principles (Lin's information theoretic measure, Kullback/Leibler divergence...) bring us outside the scope of vector spaces, and their application to semantic vectors can be iffy and ad-hoc

Outline

Introduction: The distributional hypothesis

Constructing the models

Semantic similarity as geometric distance

Evaluation

Multimodal distributional models

- Computer vision

Compositionality

- Why?
- How?

Conclusion

Recap: Constructing the models

- Pre-process the source corpus
- Collect a co-occurrence matrix (with *distributional vectors* representing words as rows, and contextual elements of some kind as columns/dimensions)
- Transform the matrix: re-weighting raw frequencies, dimensionality reduction
- Use resulting matrix to compute word-to-word similarity

Distributional similarity as semantic similarity

- Developers of DSMs typically want them to be "general-purpose" models of semantic similarity
- These models emphasize *paradigmatic* similarity, i.e., words that tend to occur in the same contexts
- Words that share many contexts will correspond to concepts that share many attributes (*attributional similarity*), i.e., concepts that are taxonomically similar:
  - Synonyms (*rhino/rhinoceros*), antonyms and values on a scale (*good/bad*), co-hyponyms (*rock/jazz*), hyper- and hyponyms (*rock/basalt*)
- Taxonomic similarity is seen as the fundamental semantic relation, allowing categorization, generalization, inheritance
- Evaluation focuses on tasks that measure taxonomic similarity

Distributional semantics as models of word meaning

Landauer and Dumais 1997, Turney and Pantel 2010, Baroni and Lenci 2010

Distributional semantics can model

- human similarity judgments (*cord-string* vs. *cord-smile*)
- lexical priming (*hospital* primes *doctor*)
- synonymy (*zenith-pinnacle*)
- analogy (*mason* is to *stone* like *carpenter* is to *wood*)
- relation classification (*exam-anxiety*: CAUSE-EFFECT)
- text coherence
- …

The main problem with evaluation: Parameter Hell!

- So many parameters in tuning the models:
  - input corpus, context, counting, weighting, matrix manipulation, similarity measure
- With interactions (Erk & Padó, 2009, and others)
- And best parameters in a task might not be the best for another
- No way we can experimentally explore the parameter space
  - But see work by Bullinaria and colleagues for some systematic attempt

Nearest neighbour examples

BNC, 2-content-word-window context

rhino	fall	rock
woodpecker	rise	lava
rhinoceros	increase	sand
swan	fluctuation	boulder
whale	drop	ice
ivory	decrease	jazz
plover	reduction	slab
elephant	logarithm	cliff
bear	decline	pop
satin	cut	basalt
sweatshirt	hike	crevice

Nearest neighbour examples

BNC, 2-content-word-window context

green	good	sing
blue	bad	dance
yellow	excellent	whistle
brown	superb	mime
bright	poor	shout
emerald	improved	sound
grey	perfect	listen
speckled	clever	recite
greenish	terrific	play
purple	lucky	hear
gleaming	smashing	hiss

Some classic semantic similarity tasks

- Taking the TOEFL: synonym identification
- The Rubenstein/Goodenough norms: modeling semantic similarity judgments
- The Hodgson semantic priming data

The TOEFL synonym match task

- 80 items
- Target: *levied*
  Candidates: *imposed, believed, requested, correlated*
- In semantic space, measure angles between target and candidate context vectors, pick candidate that forms most narrow angle with target

The TOEFL synonym match task

- 80 items
- Target: *levied*
  Candidates: *imposed, believed, requested, correlated*
- In semantic space, measure angles between target and candidate context vectors, pick candidate that forms most narrow angle with target

The TOEFL synonym match task

- 80 items
- Target: *levied*
  Candidates: *imposed, believed, requested, correlated*
- In semantic space, measure angles between target and candidate context vectors, pick candidate that forms most narrow angle with target

Human performance on the synonym match task

- Average foreign test taker: 64.5%
- Macquarie University staff (Rapp 2004):
  - Average of 5 non-natives: 86.75%
  - Average of 5 natives: 97.75%

Distributional Semantics takes the TOEFL

- Humans:
  - Foreign test takers: 64.5%
  - Macquarie non-natives: 86.75%
  - Macquarie natives: 97.75%
- Machines:
  - Classic LSA: 64.4%
  - Padó and Lapata's dependency-filtered model: 73%
  - Rapp's 2003 SVD-based model trained on lemmatized BNC: 92.5%
- Direct comparison in Baroni and Lenci 2010 (ukWaC+Wikipedia+BNC as training data, local MI weighting):
  - Dependency-filtered: 76.9%
  - Dependency-linked: 75.0%
  - Co-occurrence window: 69.4%

Rubenstein & Goodenough (1965)

- (Approximately) continuous similarity judgments
- 65 noun pairs rated by 51 subjects on a 0-4 similarity scale and averaged
  - E.g.: *car*-*automobile* 3.9; *food*-*fruit* 2.7; *cord*-*smile* 0.0
- (Pearson) correlation between cosine of angle between pair context vectors and the judgment averages
- State-of-the-art results:
  - Herdağdelen et al. (2009) using SVD-ed dependency-filtered model estimated on ukWaC: 80%
- Direct comparison in Baroni et al.'s experiments:
  - Co-occurrence window: 65%
  - Dependency-filtered: 57%
  - Dependency-linked: 57%

Semantic priming

- Hearing/reading a “related” prime facilitates access to a target in various lexical tasks (naming, lexical decision, reading…)
- You recognize/access the word *pear* faster if you just heard/read *apple*
- Hodgson (1991) single word lexical decision task, 136 prime-target pairs
  - (I have no access to original article, rely on McDonald & Brew 2004 and Padó & Lapata 2007)

Semantic priming

- Hodgson found similar amounts of priming for different semantic relations between primes and targets (approx. 23 pairs per relation):
  - synonyms (synonym): *to dread/to fear*
  - antonyms (antonym): *short/tall*
  - coordinates (coord): *train/truck*
  - super- and subordinate pairs (supersub): *container/bottle*
  - free association pairs (freeass): *dove/peace*
  - phrasal associates (phrasacc): *vacant/building*

Simulating semantic priming

Methodology from McDonald & Brew, Padó & Lapata

- For each related prime-target pair:
  - measure cosine-based similarity between pair elements (e.g., *to dread/to fear*)
  - take average of cosine-based similarity of target with other primes from same relation data-set (e.g., *to value/to fear*) as measure of similarity of target with unrelated items
- Similarity between related items should be significantly higher than average similarity between unrelated items

Semantic priming results

- T-normalized differences between related and unrelated conditions (* <0.05, ** <0.01, according to paired t-tests)
- Results from Herdağdelen et al. (2009) based on SVD-ed dependency-filtered corpus, but similar patterns reported by McDonald & Brew and Padó & Lapata

relation	pairs	t-score	sig
synonym	23	10.015	**
antonym	24	7.724	**
coord	23	11.157	**
supersub	21	10.422	**
freeass	23	9.299	**
phrasacc	22	3.532	*

Distributional semantics in complex NLP systems and applications

- Document-by-word models have been used in Information Retrieval for decades
  - DSMs might be pursued in IR within the broad topic of "semantic search"
- Commercial use for automatic essay scoring and other language evaluation related tasks
  - `http://lsa.colorado.edu`

Distributional semantics in complex NLP systems and applications

- Elsewhere, general-purpose DSMs not too common, nor too effective:
  - Lack of reliable, well-known out-of-the-box resources comparable to WordNet
  - “Similarity” is too vague a notion for well-defined semantic needs (cf. nearest neighbour lists above)
- However, there are more-or-less successful attempts to use general-purpose distributional semantic information at least as supplementary resource in various domains, e.g.,:
  - Question answering (Tómas & Vicedo, 2007)
  - Bridging coreference resolution (Poesio et al., 1998, Versley, 2007)
  - Language modeling for speech recognition (Bellegarda, 1997)
  - Textual entailment (Zhitomirsky-Geffet and Dagan, 2009)

Distributional semantics in the humanities, social sciences, cultural studies

- Great potential, only partially explored
- E.g., Sagi et al. (2009a,b) use distributional semantics to study
  - semantic broadening (*dog* from specific breed to "generic canine") and narrowing (*deer* from "animal" to "deer") in the history of English
  - phonastemes (***gl**ance* and ***gl**eam*, *gr**owl*** and *h**owl***)
  - the parallel evolution of British and American literature over two centuries

“Culture” in distributional space

Nearest neighbours in BNC-estimated model

woman

- gay
- homosexual
- lesbian
- bearded
- burly
- macho
- sexually
- man
- stocky
- to castrate

man

- policeman
- girl
- promiscuous
- woman
- compositor
- domesticity
- pregnant
- chastity
- ordination
- warrior

Outline

Introduction: The distributional hypothesis

Constructing the models

Semantic similarity as geometric distance

Evaluation

Multimodal distributional models
- Computer vision

Compositionality
- Why?
- How?

Conclusion

Distributional semantics

Distributional meaning as co-occurrence vector

	planet	night	full	shadow	shine	crescent
moon	10	22	43	16	29	12
sun	14	10	4	15	45	0
dog	0	4	2	10	0	0

Distributional semantics

Distributional meaning as co-occurrence vector

	x729	x145	x684	x776	x998	x238
moon	10	22	43	16	29	12
sun	14	10	4	15	45	0
dog	0	4	2	10	0	0

The symbol grounding problem
Interpretation vs. translation

Searle 1980, Harnad 1990

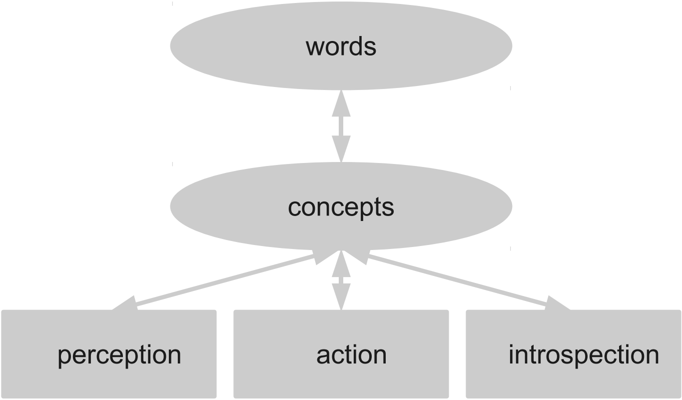
红, 紅

1. 像火或鲜血那样的颜色: "红枣 | 红日 | 面红耳赤。".
2. 借指红色的东西: "落红（指花） | 披红戴花（指红色织物）。".

`google.com`, “define” functionality

Cognitive Science: Word meaning is grounded

Barsalou 2008, Kiefer and Pulvermüller 2011 (overviews)



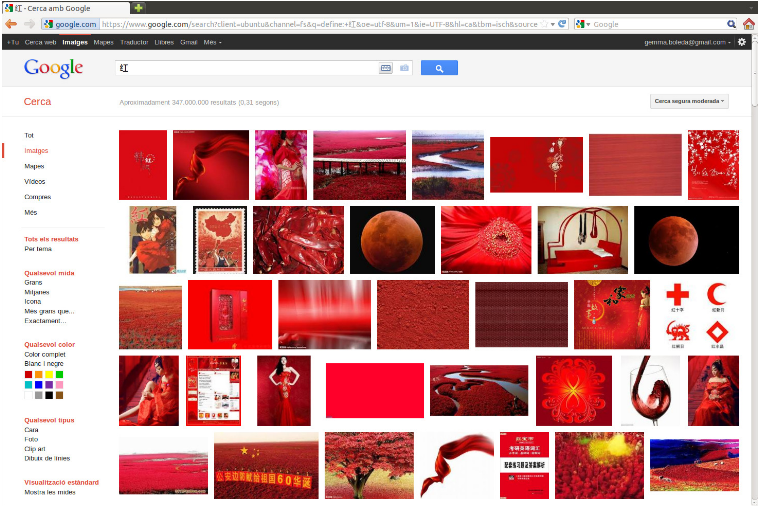
Interpretation as translation

红, 紅

1. 像火或鲜血那样的颜色: "红枣 | 红日 | 面红耳赤。".
2. 借指红色的东西: "落红（指花） | 披红戴花（指红色织物）。".

`google.com`, "define" functionality

Interpretation with perception



`images.google.com`

Classical distributional models are not grounded

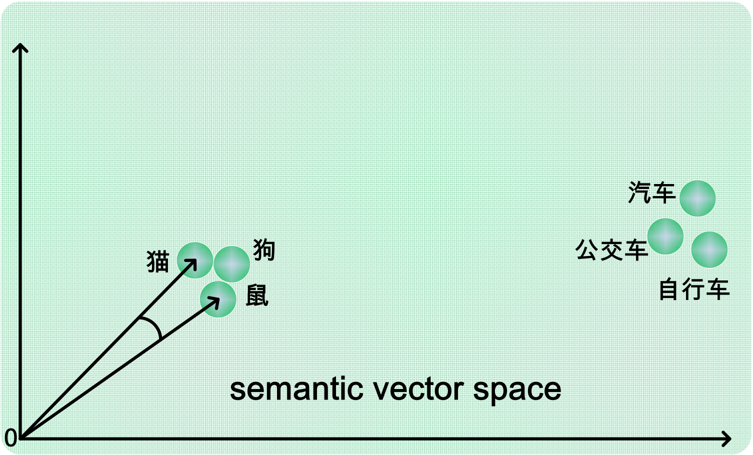


Image credit: Jiming Li

Classical distributional models are not grounded

Describing tigers. . .

humans (McRae et al., 2005):

- have stripes
- have teeth
- are black
- . . .

state-of-the art distributional model (Baroni et al., 2010):

- live in jungle
- can kill
- risk extinction
- . . .

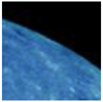
The distributional hypothesis

The meaning of a word is (can be approximated via) the set of contexts in which it occurs

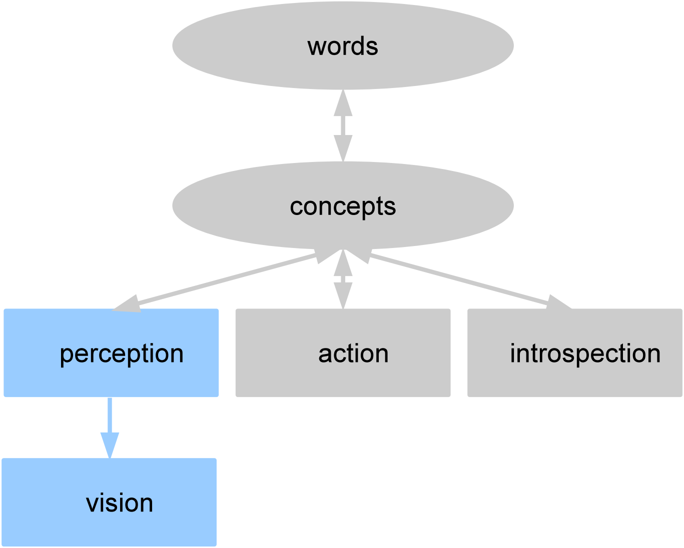
Grounding distributional semantics

Multimodal models using textual and visual collocates

Bruni et al. JAIR 2014, Leong and Mihalchea IJCNLP 2011, Silberer et al. ACL 2013

	planet	night		
moon	10	22	22	0
sun	14	10	15	0
dog	0	4	0	20

Multimodal models vith images



Multimodal models

- other modalities: feature norms (Andrews et al. 2010, Roller and Schulte im Walde EMNLP 2013)
  - feature norms: *tiger - has stripes*...
  - manually collected...

Outline

Introduction: The distributional hypothesis

Constructing the models

Semantic similarity as geometric distance

Evaluation

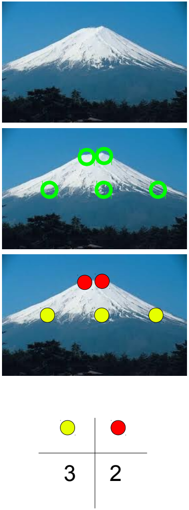
Multimodal distributional models
- Computer vision

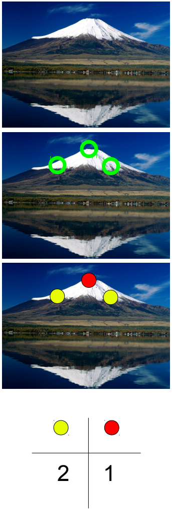
Compositionality
- Why?
- How?

Conclusion

Bags of visual words

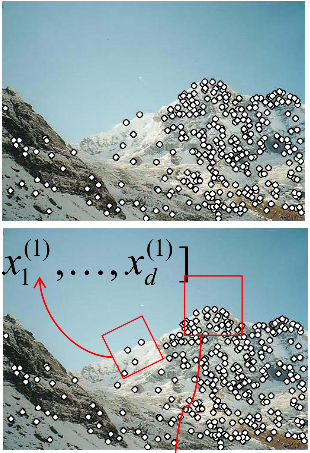
Motivation





Detection and description

- Detection: Identify the interest points, e.g. with Harris corner detectors
- Description: Extract feature vector describing area surrounding each interest point, e.g. SIFT descriptor

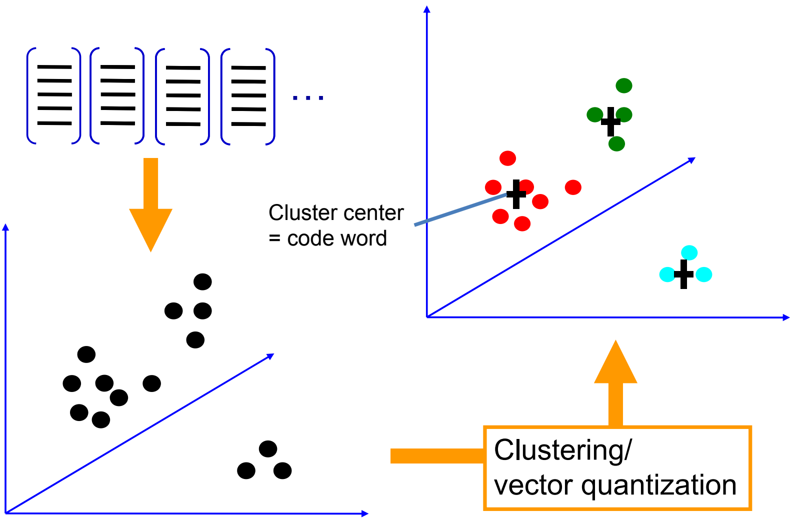


$$\mathbf{x}_1 = [x_1^{(1)}, \ldots, x_d^{(1)}]$$

$$\mathbf{x}_2 = [x_1^{(2)}, \ldots, x_d^{(2)}]$$

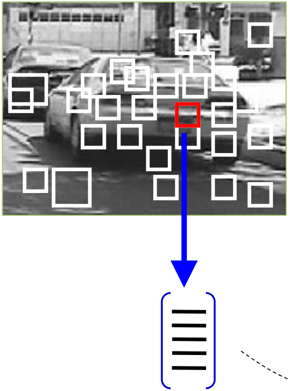
[Fei-Fei Li]

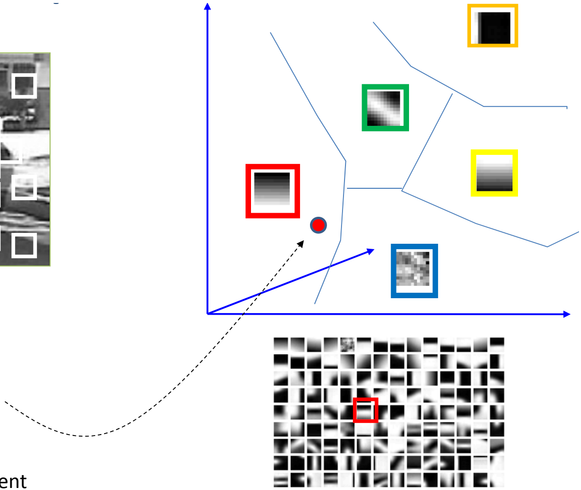
Visual codeword dictionary formation by clustering



[Fei-Fei Li]

Vector mapping



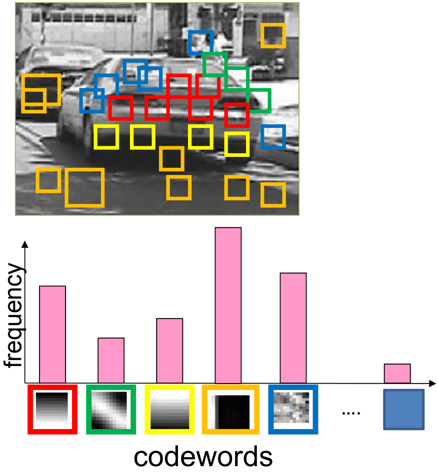


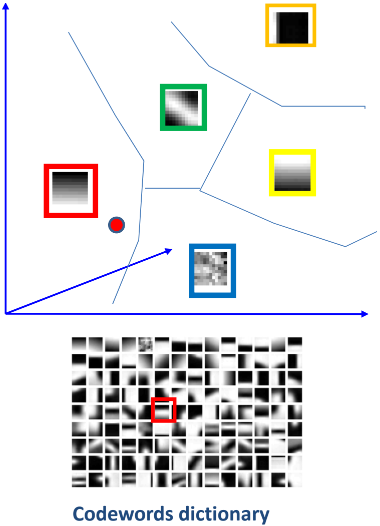
• Nearest neighbors assignment
• K-D tree search strategy

Codewords dictionary

[Fei-Fei Li]

Counting

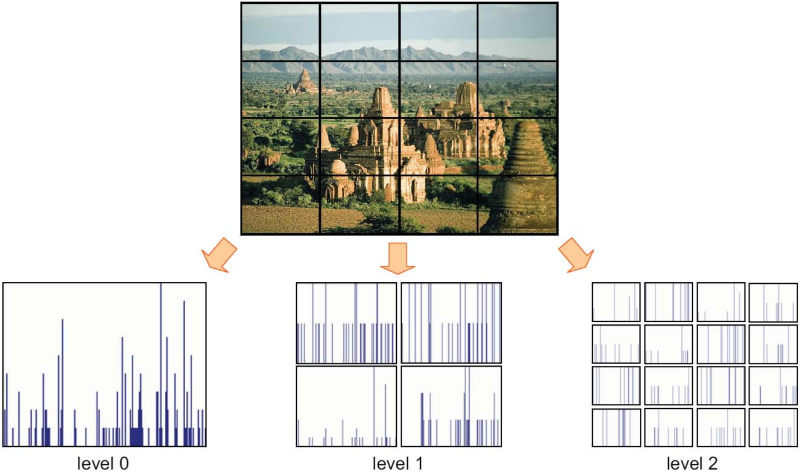




[Fei-Fei Li]

Spatial pyramid representation

Lazebnik, Schmid, and Ponce, 2006, 2009



Empirical assessment

Feng and Lapata 2010

Michelle Obama fever hits the UK

In the UK on her first visit as first lady, Michelle Obama seems to be making just as big an impact. She has attracted as much interest and column inches as her husband on this London trip; creating a buzz with her dazzling outfits, her own schedule of events and her own fanbase. Outside Buckingham Palace, as crowds gathered in anticipation of the Obamas' arrival, Mrs Obama's star appeal was apparent.

- Feng and Lapata 2010: Model learns from mixed-media documents a joint word+visual-word Topic Model

Model	Word Association	Word Similarity
UpperBnd	0.400	0.545
MixLDA	0.123	0.318
TxtLDA	0.077	0.247

Empirical assessment

Bruni et al. ACL 2012, also see Bruni et al. JAIR 2014

- Bruni et al. ACL 2012: textual and visual vectors concatenated
- multimodal better at general word similarity – 0.66 vs. 0.69 (MEN dataset)
- multimodal better at modeling the meaning of color terms
  - a banana is yellow: multimodal gets 27/52 right, text only 13
  - literal vs. non-literal uses of color terms:
    - a blue uniform is blue, a blue note is not
    - text .53, multimodal .73 (complicated metric)
- more sophisticated combination of textual and visual information yields further improvements (Bruni et al. JAIR 2014)

Outline

Introduction: The distributional hypothesis

Constructing the models

Semantic similarity as geometric distance

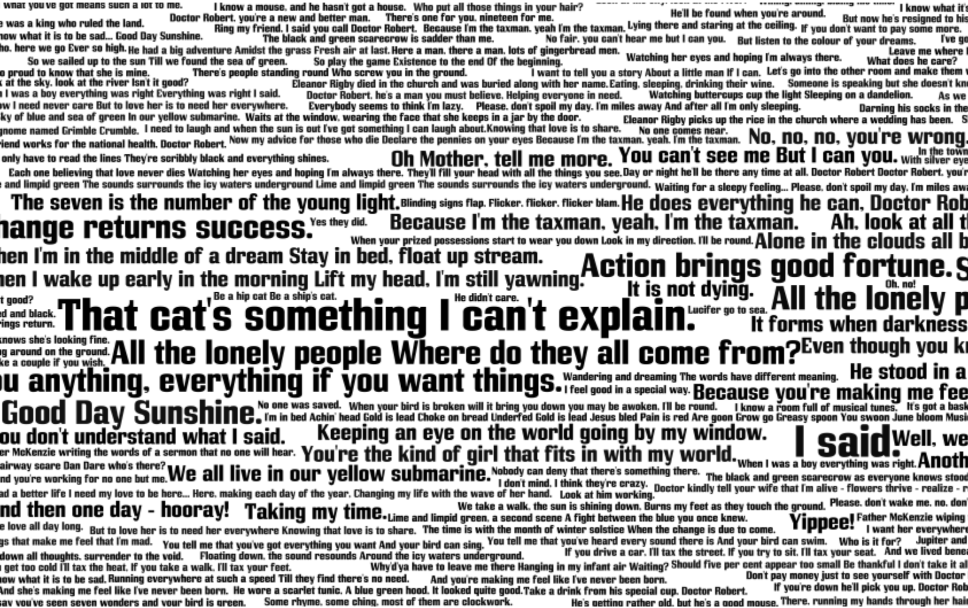
Evaluation

Multimodal distributional models
- Computer vision

Compositionality
- Why?
- How?

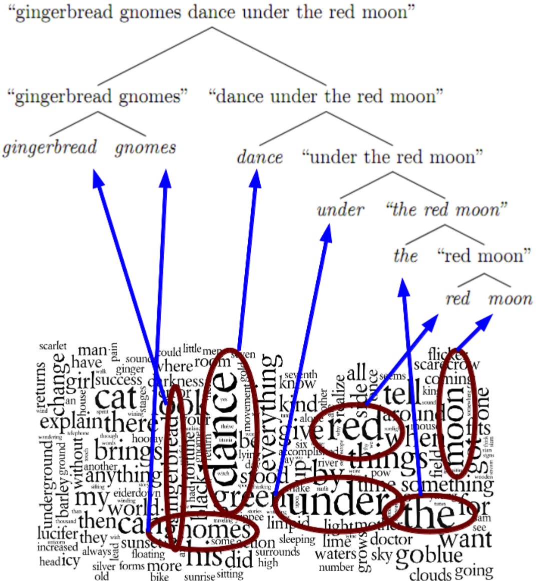
Conclusion

The infinity of sentence meaning



Compositionality

The meaning of an utterance is a function of the meaning of its parts and their composition rules (Frege 1892)



A compositional distributional semantics for phrases and sentences?

Mitchell and Lapata 2008, 2009, 2010, Grefenstette and Sadrzadeh 2011, Baroni and Zamparelli 2010, . . .

	planet	night	full	blood	shine
moon	10	22	43	3	29
red moon	12	21	40	20	28
the red moon shines	11	23	21	15	45

Outline

Introduction: The distributional hypothesis

Constructing the models

Semantic similarity as geometric distance

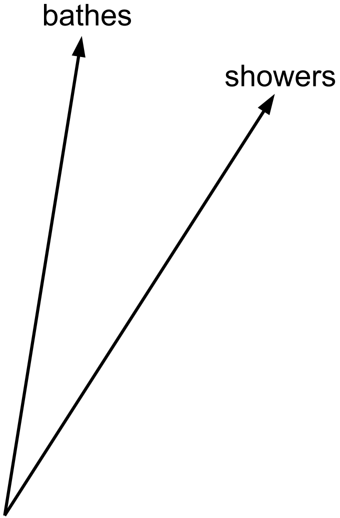
Evaluation

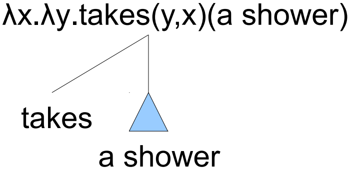
Multimodal distributional models
- Computer vision

Compositionality
- Why?
- How?

Conclusion

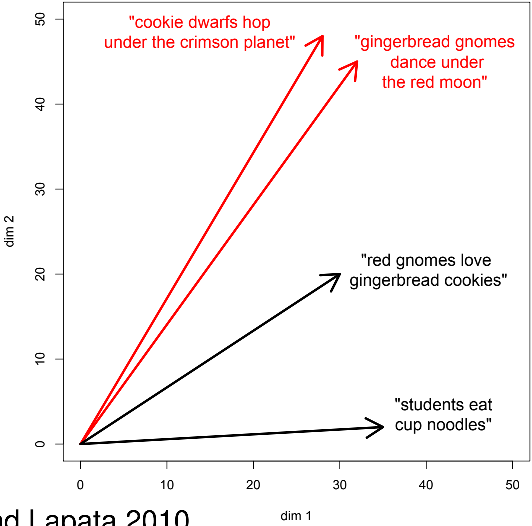
The unavoidability of distributional representations of phrases





What can you do with distributional representations of phrases and sentences?

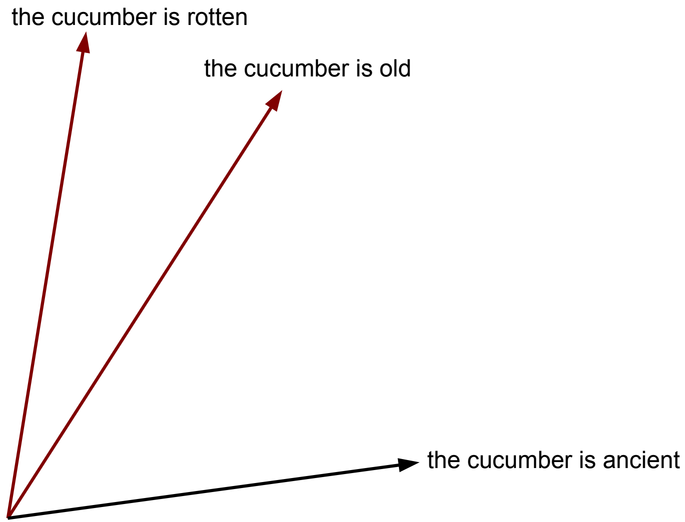
Paraphrasing



Mitchell and Lapata 2010

What can you do with distributional representations of phrases and sentences?

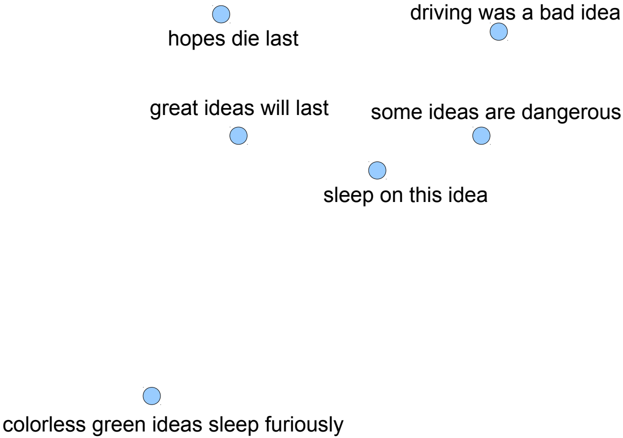
Disambiguation



Mitchell and Lapata 2008

What can you do with distributional representations of phrases and sentences?

Semantic acceptability



Vecchi, Baroni and Zamparelli 2011

Outline

Introduction: The distributional hypothesis

Constructing the models

Semantic similarity as geometric distance

Evaluation

Multimodal distributional models
- Computer vision

Compositionality
- Why?
- How?

Conclusion

Compositional distributional semantics

- Mitchell, J. & Lapata, M. (2010). Composition in distributional models of semantics. *Cognitive Science* 34(8): 1388–1429
- Baroni, M. & Zamparelli, R. (2010). Nouns are vectors, adjectives are matrices: Representing adjective-noun constructions in semantic space. *Proceedings of EMNLP*
- Grefenstette, E., Dinu, G., Zhang, Y., Sadrzadeh, M. & Baroni, M. (Submitted). Multi-step regression learning for compositional distributional semantics.
- B. Coecke, M. Sadrzadeh and S. Clark. 2010. Mathematical foundations for a compositional distributed model of meaning. *Lambek Festschrift* (*Linguistic Analysis* 36)

Additive model

Mitchell and Lapata 2010, ...

	planet	night	blood	brown
red	15	3	19	20
moon	24	15	1	0
red+moon	39	18	20	20
$0.4\times$red + $0.6\times$moon	20.4	10.2	8.2	8

weighted additive model: $\vec{p} = \alpha\vec{a} + \beta\vec{n}$

Additive model

Mitchell and Lapata 2010, . . .

	planet	night	blood	brown
red	15	3	19	20
moon	24	15	1	0
red+moon	39	18	20	20
0.4×red + 0.6×moon	20.4	10.2	8.2	8

weighted additive model: $\vec{p} = \alpha\vec{a} + \beta\vec{n}$

Additive model

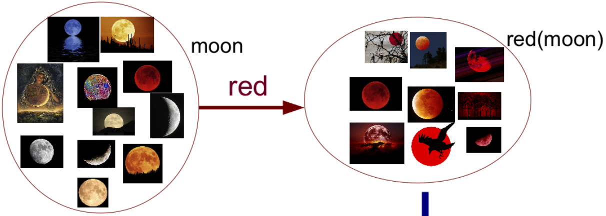
Mitchell and Lapata 2010, ...

	planet	night	blood	brown
red	15	3	19	20
moon	24	15	1	0
red+moon	39	18	20	20
0.4×red + 0.6×moon	20.4	10.2	8.2	8

weighted additive model: $\vec{p} = \alpha\vec{a} + \beta\vec{n}$

Composition as (distributional) function application

Grefenstette, Sadrzadeh et al., Baroni and Zamparelli, Socher et al.?

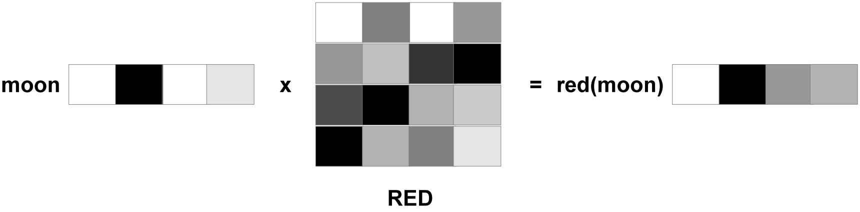


$$\overrightarrow{moon} = \begin{bmatrix} \mathit{shine} & 301 \\ \mathit{eclipse} & 250 \\ \mathit{blood} & 93 \\ \dots & \dots \end{bmatrix} \xrightarrow{\text{red}} \overrightarrow{\mathrm{red}(\mathit{moon})} = \begin{bmatrix} \mathit{shine} & 11 \\ \mathit{eclipse} & 245 \\ \mathit{blood} & 90 \\ \dots & \dots \end{bmatrix}$$

Baroni and Zamparelli's 2010 proposal

Implementing the idea of function application in a vector space

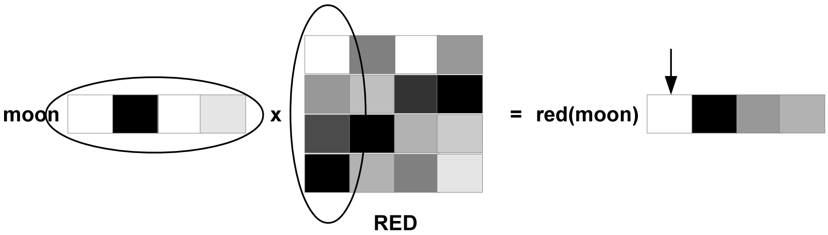
- Functions as **linear maps** between vector spaces
- Functions are matrices, function application is function-by-vector multiplication



Baroni and Zamparelli's 2010 proposal

Implementing the idea of function application in a vector space

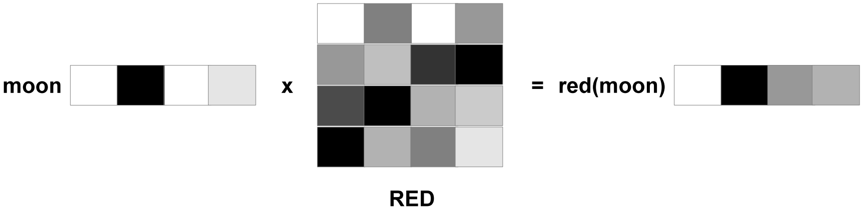
- Functions as **linear maps** between vector spaces
- Functions are matrices, function application is function-by-vector multiplication



Baroni and Zamparelli's 2010 proposal

Implementing the idea of function application in a vector space

- Functions as **linear maps** between vector spaces
- Functions are matrices, function application is function-by-vector multiplication



lexical function model: $\vec{p} = \mathbf{A}\vec{n}$

Learning distributional composition functions

```
n and the moon shining i
 with the moon shining s
rainbowed moon . And the
 crescent moon , thrille
in a blue moon only , wi
now , the moon has risen
d now the moon rises , f
y at full moon , get up
 crescent moon . Mr Angu
```

```
f a large red moon , Campana
, a blood red moon hung over
 glorious red moon turning t
The round red moon , she ’s
l a blood red moon emerged f
n rains , red moon blows , w
monstrous red moon had climb
 . A very red moon rising is
under the red moon a vampire
```

	shine	blood
moon	301	93
red moon	11	90

$\vec{moon} \rightarrow \vec{red\ moon}$

$\vec{light} \rightarrow \vec{red\ light}$

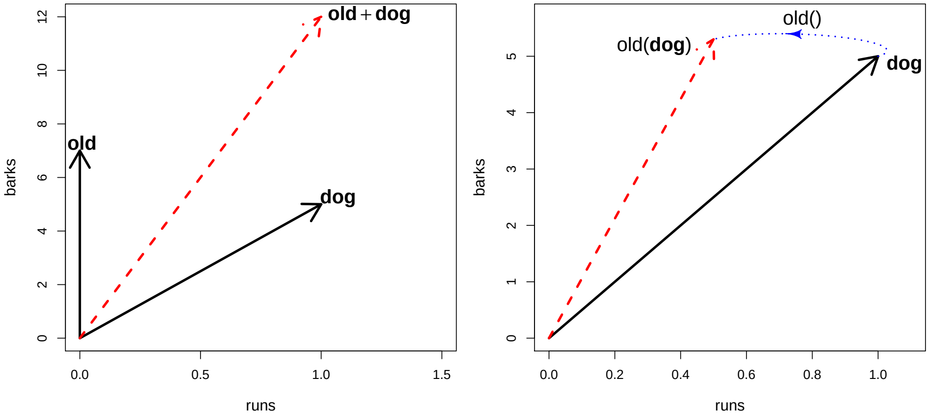
$\vec{dress} \rightarrow \vec{red\ dress}$

$\vec{alert} \rightarrow \vec{red\ alert}$

$\ldots$

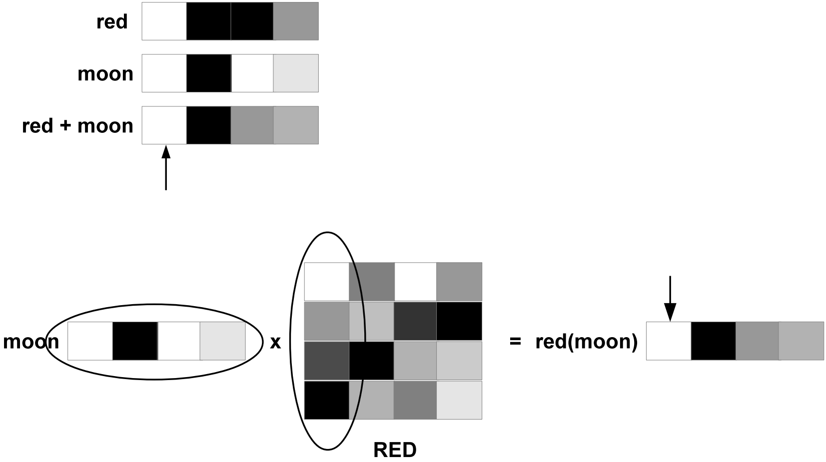
Addition and lexical function

as models of adjective meaning



Addition and lexical function

as models of adjective meaning



Socher et al.

R. Socher, E. Huang, J. Pennington, A. Ng and Ch. Manning. 2011. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. *Proceedings of NIPS*.

More recently R. Socher, B. Huval, Ch. Manning and A. Ng. 2012. Semantic compositionality through recursive matrix-vector spaces, *Proceedings of EMNLP*...

- makes more explicit link with compositionality literature
- similarities with function-based approaches above
- supervised approach in which composition solution depends on annotated data from task at hand

Socher et al.
Main points (for our purposes)

- Measure similarity of sentences taking into account not only sentence vector, but also vectors representing all constituent phrases and words
  - Map these representations to similarity matrix of fixed size, even for sentences with different lengths and structures
- Neural-network-based learning of composition function (autoencoders)

Results

- for some tasks, more sophisticated methods outperform the additive model
- but the additive model is surprisingly good
- one of the problems: lack of adequate testbeds
  - see this year's SemEval Task 1

Outline

Introduction: The distributional hypothesis

Constructing the models

Semantic similarity as geometric distance

Evaluation

Multimodal distributional models
- Computer vision

Compositionality
- Why?
- How?

Conclusion

Some hot topics

- Compositionality in distributional semantics
- Semantic representations in context (polysemy resolution, co-composition…)
- Multimodal DSMs
- Very large DSMs

Not solved

- Parameter Hell

Build your own distributional semantic model

- corpus (several out there for several languages, see archives of the Corpora Mailing List)
- Standard linguistic pre-processing and indexing tools (TreeTagger, MaltParser, IMS CWB. . .)
- easy to write scripts for co-occurrence counts
  - not trivial with very large corpora. Hadoop (MapReduce algorithm) ideal for this, but often a pain in practice.
- COMPOSES webpage with link to toolkit in progress: `http://clic.cimec.unitn.it/composes`
- See the Links page for other toolkits!
- if you build your own matrix: Dimensionality reduction with SVDLIBC (`http://tedlab.mit.edu/~dr/svdlibc/`)

Distributional Semantics

Marco Baroni and Gemma Boleda

CS 388: Natural Language Processing