Distributional Semantic Models

Part 1: Introduction

Stefan Evert\textsuperscript{1}
with Alessandro Lenci\textsuperscript{2}, Marco Baroni\textsuperscript{3} and Gabriella Lapesa\textsuperscript{4}

\textsuperscript{1}Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany
\textsuperscript{2}University of Pisa, Italy
\textsuperscript{3}University of Trento, Italy
\textsuperscript{4}University of Stuttgart, Germany

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

Outline

Introduction

The distributional hypothesis
Distributional semantic models
Three famous examples

Getting practical

Software and further information
R as a (toy) laboratory

Meaning & distribution

▸ “Die Bedeutung eines Wortes liegt in seinem Gebrauch.”
— Ludwig Wittgenstein

meaning = use = distribution in language

▸ “You shall know a word by the company it keeps!”
— J. R. Firth (1957)

distribution = collocations = habitual word combinations

▸ Distributional hypothesis: difference of meaning correlates
  with difference of distribution (Zellig Harris 1954)

semantic distance

▸ “What people know when they say that they know a word is
  not how to recite its dictionary definition – they know how to
  use it [...] in everyday discourse.” (Miller 1986)
Introduction: The distributional hypothesis

What is the meaning of “bardiwac”? Can we infer meaning from usage?

- He handed her her glass of bardiwac claret.
- Beef dishes are made to complement the bardiwac claret.
- Nigel staggered to his feet, face flushed from too much bardiwac claret.
- Malbec, one of the lesser-known bardiwac claret grapes, responds well to Australia’s sunshine.
- I dined off bread and cheese and this excellent bardiwac claret.
- The drinks were delicious: blood-red bardiwac claret as well as light, sweet Rhenish.

© Evert/Lenci/Baroni/Lapesa (CC-by-sa) DSM Tutorial – Part 1

Word sketch of “cat” Can we infer meaning from collocations?

sim(______cat______, ______naif______) = 0.770

© Evert/Lenci/Baroni/Lapesa (CC-by-sa) DSM Tutorial – Part 1
The distributional hypothesis

A thought experiment: deciphering hieroglyphs

get sij ius hir iit kil

knife

naif

51 20 84 0 3 0

(boat)

beut

59 39 23 4 0 0

(cup)

kap

98 14 6 2 1 0

(pig)

pigij

12 17 3 2 9 27

(banana)

nana

11 2 2 0 18 0

\[ \text{sim}(\text{knife}, \text{cat}) = 0.939 \]

sim(dog, ket) = 0.961

© Evert/Lenci/Baroni/Lapesa (CC-by-sa)

DSM Tutorial – Part 1

Wordspace.collocations.de

7 / 44

English as seen by the computer ...

get see use hear eat kill

knife

51 20 84 0 3 0

cat

52 58 4 4 6 26

dog

115 83 10 42 33 17

boat

59 39 23 4 0 0

cup

98 14 6 2 1 0

pig

12 17 3 2 9 27

banana

11 2 2 0 18 0

verb-object counts from British National Corpus

© Evert/Lenci/Baroni/Lapesa (CC-by-sa)

DSM Tutorial – Part 1

Wordspace.collocations.de

8 / 44

Geometric interpretation

- row vector \( \mathbf{x}_{\text{dog}} \) describes usage of word dog in the corpus
- can be seen as coordinates of point in \( n \)-dimensional Euclidean space

co-occurrence matrix \( \mathbf{M} \)
**Introduction**

- **The distributional hypothesis**
  - **Geometric interpretation**
    - row vector $x_{\text{dog}}$ describes usage of word dog in the corpus
    - can be seen as coordinates of point in $n$-dimensional Euclidean space
    - illustrated for two dimensions: get and use
    - $x_{\text{dog}} = (115, 10)$

**Geometric interpretation**

- similarity = spatial proximity (Euclidean dist.)
- location depends on frequency of noun ($f_{\text{dog}} \approx 2.7 \cdot f_{\text{cat}}$)

**Geometric interpretation**

- vector can also be understood as arrow from origin
- direction more important than location
- use angle $\alpha$ as distance measure
- or normalise length $\|x_{\text{dog}}\|$ of arrow

© Evert/Lenci/Baroni/Lapesa (CC-by-sa) DSM Tutorial – Part 1 wordspace.collocations.de 10 / 44

© Evert/Lenci/Baroni/Lapesa (CC-by-sa) DSM Tutorial – Part 1 wordspace.collocations.de 11 / 44

© Evert/Lenci/Baroni/Lapesa (CC-by-sa) DSM Tutorial – Part 1 wordspace.collocations.de 13 / 44
**Outline**

**Introduction**
- The distributional hypothesis
- Distributional semantic models
- Three famous examples

**Getting practical**
- Software and further information
- R as a (toy) laboratory

---

**General definition of DSMs**

A **distributional semantic model** (DSM) is a scaled and/or transformed co-occurrence matrix $M$, such that each row $x$ represents the distribution of a target term across contexts.

<table>
<thead>
<tr>
<th>get</th>
<th>see</th>
<th>use</th>
<th>hear</th>
<th>eat</th>
<th>kill</th>
</tr>
</thead>
<tbody>
<tr>
<td>knife</td>
<td>0.027</td>
<td>-0.24</td>
<td>0.206</td>
<td>-0.022</td>
<td>-0.044</td>
</tr>
<tr>
<td>cat</td>
<td>0.031</td>
<td>0.143</td>
<td>-0.243</td>
<td>-0.015</td>
<td>-0.009</td>
</tr>
<tr>
<td>dog</td>
<td>-0.026</td>
<td>0.021</td>
<td>-0.212</td>
<td>0.064</td>
<td>0.013</td>
</tr>
<tr>
<td>boat</td>
<td>-0.022</td>
<td>0.009</td>
<td>-0.044</td>
<td>-0.040</td>
<td>-0.074</td>
</tr>
<tr>
<td>cup</td>
<td>-0.014</td>
<td>-0.173</td>
<td>-0.249</td>
<td>-0.099</td>
<td>-0.119</td>
</tr>
<tr>
<td>pig</td>
<td>-0.069</td>
<td>0.094</td>
<td>-0.158</td>
<td>0.000</td>
<td>0.094</td>
</tr>
<tr>
<td>banana</td>
<td>0.047</td>
<td>-0.139</td>
<td>-0.104</td>
<td>-0.022</td>
<td>0.267</td>
</tr>
</tbody>
</table>

**Term** = word, lemma, phrase, morpheme, word pair, ...

---

**Building a distributional model**

- Pre-processed corpus with linguistic annotation
- Define target terms
- Context tokens or types
- Term-term matrix
- Type & size of co-occurrence
- Term-context matrix
- Define target & feature terms
- Geometric analysis
- Probabilistic analysis
- Similarity/distance measure + normalization
- Feature scaling
- Dimensionality reduction
- Embedding learned by neural network
- Nearest neighbours

**Nearest neighbours**

DSM based on verb-object relations from BNC, reduced to 100 dim. with SVD

- Neighbours of **trousers** (cosine angle):
  - shirt (18.5), blouse (21.9), scarf (23.4), jeans (24.7), skirt (25.9), sock (26.2), shorts (26.3), jacket (27.8), glove (28.1), coat (28.8), cloak (28.9), hat (29.1), tunic (29.3), overcoat (29.4), pants (29.8), helmet (30.4), apron (30.5), robe (30.6), mask (30.8), tracksuit (31.0), jersey (31.6), shawl (31.6), ...

- Neighbours of **rage** (cosine angle):
  - anger (28.5), fury (32.5), sadness (37.0), disgust (37.4), emotion (39.0), jealousy (40.0), grief (40.4), irritation (40.7), revulsion (40.7), scorn (40.7), panic (40.8), bitterness (41.6), resentment (41.8), indignation (41.9), excitement (42.0), hatred (42.5), envy (42.8), disappointment (42.9), ...
Nearest neighbours with similarity graph

Clustering

Semantic maps

Latent “meaning” dimensions
Introduction

Distributional semantic models

Word embeddings

DSM vector as sub-symbolic meaning representation

▶ feature vector for machine learning algorithm
▶ input for neural network

Context vectors for word tokens (Schütze 1998)

▶ bag-of-words approach: centroid of all context words in the sentence
▶ application to WSD

Outline

Introduction

The distributional hypothesis
Distributional semantic models
Three famous examples

Getting practical

Software and further information
R as a (toy) laboratory

An important distinction

▶ Distributional model
  ▶ captures linguistic distribution of each word in the form of a high-dimensional numeric vector
  ▶ typically (but not necessarily) based on co-occurrence counts
  ▶ distributional hypothesis: distributional similarity/distance $\sim$ semantic similarity/distance

▶ Distributed representation
  ▶ sub-symbolic representation of words as high-dimensional numeric vectors
  ▶ similarity of vectors usually (but not necessarily) corresponds to semantic similarity of the words
  ▶ hot topic: unsupervised neural word embeddings

Distributional model can be used as distributed representation

Latent Semantic Analysis (Landauer and Dumais 1997)

▶ Corpus: 30,473 articles from Grolier’s *Academic American Encyclopedia* (4.6 million words in total)
  ▶ articles were limited to first 2,000 characters
▶ Word-article frequency matrix for 60,768 words
  ▶ row vector shows frequency of word in each article
▶ Logarithmic frequencies scaled by word entropy
▶ Reduced to 300 dim. by singular value decomposition (SVD)
  ▶ borrowed from LSI (Dumais et al. 1988)
  ▶ central claim: SVD reveals latent semantic features, not just a data reduction technique
▶ Evaluated on TOEFL synonym test (80 items)
  ▶ LSA model achieved 64.4% correct answers
  ▶ also simulation of learning rate based on TOEFL results
Introduction

Three famous examples


- Corpus: ≈ 60 million words of news messages
  - from the New York Times News Service
- Word-word co-occurrence matrix
  - 20,000 target words & 2,000 context words as features
  - row vector records how often each context word occurs close to the target word (co-occurrence)
  - co-occurrence window: left/right 50 words (Schütze 1998) or ≈ 1000 characters (Schütze 1992)
- Rows weighted by inverse document frequency (tf.idf)
- Context vector = centroid of word vectors (bag-of-words)
- Goal: determine “meaning” of a context
- Reduced to 100 SVD dimensions (mainly for efficiency)
- Evaluated on unsupervised word sense induction by clustering of context vectors (for an ambiguous word)
- Induced word senses improve information retrieval performance

HAL (Lund and Burgess 1996)

- HAL = Hyperspace Analogue to Language
- Corpus: 160 million words from newsgroup postings
- Word-word co-occurrence matrix
  - same 70,000 words used as targets and features
  - co-occurrence window of 1 – 10 words
- Separate counts for left and right co-occurrence
  - i.e. the context is structured
- In later work, co-occurrences are weighted by (inverse) distance (Li et al. 2000)
  - But no dimensionality reduction
- Applications include construction of semantic vocabulary maps by multidimensional scaling to 2 dimensions

Many parameters . . .

- Enormous range of DSM parameters and applications
- Examples showed three entirely different models, each tuned to its particular application

- Need overview of DSM parameters & understand their effects
  - Part 2: The parameters of a DSM
  - Part 3: Evaluating DSM representations
  - Part 4: The mathematics of DSMs
  - Part 5: Understanding distributional semantics

- Distributional semantics is an empirical science
Outline

Introduction
- The distributional hypothesis
- Distributional semantic models
- Three famous examples

Getting practical
- Software and further information
- R as a (toy) laboratory

Some applications in computational linguistics
- Unsupervised part-of-speech induction (Schütze 1995)
- Word sense disambiguation (Schütze 1998)
- Query expansion in information retrieval (Grefenstette 1994)
- Synonym tasks & other language tests (Landauer and Dumais 1997; Turney et al. 2003)
- Thesaurus compilation (Lin 1998; Rapp 2004)
- Ontology & wordnet expansion (Pantel et al. 2009)
- Attachment disambiguation (Pantel and Lin 2000)
- Probabilistic language models (Bengio et al. 2003)
- Sub-symbolic input representation for neural networks
- Many other tasks in computational semantics: entailment detection, noun compound interpretation, identification of noncompositional expressions, ...

Recent conferences and workshops
- 2007: CoSMo Workshop (at Context '07)
- 2008: ESSLLI Lexical Semantics Workshop & Shared Task, Special Issue of the Italian Journal of Linguistics
- 2009: GeMS Workshop (EACL 2009), DiSCo Workshop (CogSci 2009), ESSLLI Advanced Course on DSM
- 2010: 2nd GeMS (ACL 2010), ESSLLI Workshop on Compositionality and DSM, DSM Tutorial (NAACL 2010), Special Issue of JNLE on Distributional Lexical Semantics
- 2011: 2nd DiSCo (ACL 2011), 3rd GeMS (EMNLP 2011)
- 2012: DiDaS (at ICSC 2012)
- 2013: CVSC (ACL 2013), TFDS (IWCS 2013), Dagstuhl
- 2014: 2nd CVSC (at EACL 2014)

Software packages

- HiDEx: C++ re-implementation of the HAL model (Lund and Burgess 1996)
- SemanticVectors: Java scalable architecture based on random indexing representation
- S-Space: Java complex object-oriented framework
- JoBimText: Java UIMA / Hadoop framework
- Gensim: Python complex framework, focus on parallelization and out-of-core algorithms
- DISSECT: Python user-friendly, designed for research on compositional semantics
- wordspace: R interactive research laboratory, but scales to real-life data sets
Further information

- Handouts & other materials available from wordspace wiki at http://wordspace.collocations.de/
  - based on joint work with Marco Baroni and Alessandro Lenci
- Tutorial is open source (CC), and can be downloaded from http://r-forge.r-project.org/projects/wordspace/
- Review paper on distributional semantics:
- I should be working on textbook *Distributional Semantics* for *Synthesis Lectures on HLT* (Morgan & Claypool)

Prepare to get your hands dirty ...

- We will use the statistical programming environment R as a toy laboratory in this tutorial
  - but one that scales to real-life applications

Software installation

- R version 3.3 or newer from http://www.r-project.org/
- RStudio from http://www.rstudio.com/
- R packages from CRAN (through RStudio menu):
  - sparsesvd, wordspace
    - if you are attending a course, you may also be asked to install the wordspaceEval package with some non-public data sets
- Data sets from http://www.collocations.de/data/#dsm

Outline

Introduction
- The distributional hypothesis
- Distributional semantic models
- Three famous examples

Getting practical
- Software and further information
- R as a (toy) laboratory

First steps in R

Start each session by loading the wordspace package.

```
> library(wordspace)
```

The package includes various example data sets, some of which should look familiar to you.

```
> DSM_HieroglyphsMatrix
        get see use hear eat kill
knife  51  20  84  0  3  0
cat    52  58  4   6  26
dog    115 83  10 42 33 17
boat   59  39 23  4  0  0
cup    98  14  6  2  1  0
pig    12  17  3  2  9  27
banana 11  2  2  0 18  0
```
**Term-term matrix**

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

```r
> DSM_TermTermMatrix
```

<table>
<thead>
<tr>
<th></th>
<th>breed</th>
<th>tail</th>
<th>feed</th>
<th>kill</th>
<th>important</th>
<th>explain</th>
<th>likely</th>
</tr>
</thead>
<tbody>
<tr>
<td>cat</td>
<td>83</td>
<td>17</td>
<td>37</td>
<td>-1</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>dog</td>
<td>561</td>
<td>13</td>
<td>30</td>
<td>60</td>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>animal</td>
<td>42</td>
<td>109</td>
<td>134</td>
<td>13</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>time</td>
<td>19</td>
<td>9</td>
<td>117</td>
<td>81</td>
<td>34</td>
<td>109</td>
<td></td>
</tr>
<tr>
<td>reason</td>
<td>1</td>
<td>-</td>
<td>2</td>
<td>14</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>cause</td>
<td>-</td>
<td>1</td>
<td>4</td>
<td>55</td>
<td>34</td>
<td>35</td>
<td>15</td>
</tr>
<tr>
<td>effect</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>6</td>
<td>60</td>
<td>35</td>
<td>17</td>
</tr>
</tbody>
</table>

**Term-context matrix**

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

```r
> DSM_TermContextMatrix
```

<table>
<thead>
<tr>
<th></th>
<th>Felidae</th>
<th>Pet</th>
<th>Feral</th>
<th>Bloat</th>
<th>Philosophy</th>
<th>Kant</th>
<th>Back pain</th>
</tr>
</thead>
<tbody>
<tr>
<td>cat</td>
<td>10</td>
<td>10</td>
<td>7</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>dog</td>
<td>-</td>
<td>10</td>
<td>4</td>
<td>11</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>animal</td>
<td>2</td>
<td>15</td>
<td>10</td>
<td>2</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>time</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>2</td>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>reason</td>
<td>-</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>cause</td>
<td>-</td>
<td>-</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>6</td>
<td>-</td>
</tr>
<tr>
<td>effect</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>-</td>
</tr>
</tbody>
</table>

**Some basic operations on a DSM matrix**

```r
# apply log-transformation to de-skew co-occurrence frequencies
> M <- log2(DSM_HieroglyphsMatrix + 1)  # see part 2
> round(M, 3)
```

```r
# compute semantic distance (cosine similarity)
> pair.distances("dog", "cat", M, convert=FALSE)
  dog/cat
  0.9610952
```

```r
# find nearest neighbours
> nearest.neighbours(M, "dog", n=3)
  cat  pig  cup
  16.03458 20.08826 31.77784
> plot(nearest.neighbours(M, "dog", n=3, dist.matrix=TRUE))
```

**References I**


Getting practical R as a (toy) laboratory

References II


References III


Turney, Peter D. and Pantel, Patrick (2010). From frequency to meaning: Vector space models of semantics. Journal of Artificial Intelligence Research, 37, 141–188.

References IV