# Distributional Semantics

Magnus Sahlgren

Pavia, 12 September 2012

# Recap

Distributional Semantics and The Distributional Hypothesis

Syntagmatic and paradigmatic similarities

Co-occurrence matrix and distributional vectors

Vector similarity and nearest neighbors

Hands-on lab

# Recap

Distributional Semantics and The Distributional Hypothesis

Syntagmatic and paradigmatic similarities

Co-occurrence matrix and distributional vectors

Vector similarity and nearest neighbors

Hands-on lab

# Recap

Distributional Semantics and The Distributional Hypothesis

Syntagmatic and paradigmatic similarities

Co-occurrence matrix and distributional vectors

Vector similarity and nearest neighbors

Hands-on lab

# Recap

Distributional Semantics and The Distributional Hypothesis

Syntagmatic and paradigmatic similarities

Co-occurrence matrix and distributional vectors

Vector similarity and nearest neighbors

Hands-on lab

# Recap

Distributional Semantics and The Distributional Hypothesis

Syntagmatic and paradigmatic similarities

Co-occurrence matrix and distributional vectors

Vector similarity and nearest neighbors

Hands-on lab

# Today

The main types of models

- Words-by-regions matrices
  - Latent Sematic Analysis (LSA)
- Words-by-words matrices
  - Hyperspace Analogue to Language (HAL)
  - Correlated Occurrence Analogue to Lexical Semantics (COALS)
  - Dependency-based models

# Today

The main types of models

- Words-by-regions matrices
  - Latent Sematic Analysis (LSA)
- Words-by-words matrices
  - Hyperspace Analogue to Language (HAL)
  - Correlated Occurrence Analogue to Lexical Semantics (COALS)
  - Dependency-based models

# Today

The main types of models

- Words-by-regions matrices
  - Latent Sematic Analysis (LSA)
- Words-by-words matrices
  - Hyperspace Analogue to Language (HAL)
  - Correlated Occurrence Analogue to Lexical Semantics (COALS)
  - Dependency-based models

# Words-by-regions Matrices

Originates from the Vector Space Model (VSM) in Information Retrieval (IR)

*Documents* are natural contexts in IR

# Words-by-regions Matrices

Originates from the Vector Space Model (VSM) in Information Retrieval (IR)

*Documents* are natural contexts in IR

# Words-by-regions Matrices

The Vector Space Model:

| | $d_1$ | $d_2$ | $d_3$ | $d_4$ |
|---|---|---|---|---|
| *coffee* | 0.00 | 0.78 | 0.00 | 0.00 |
| *ipso* | 0.23 | 0.00 | 0.12 | 0.00 |
| *offside* | 0.65 | 0.00 | 0.42 | 0.00 |
| *football* | 0.89 | 0.00 | 0.54 | 0.12 |
| *espresso* | 0.00 | 0.00 | 0.00 | 0.00 |

Words that co-occur in many documents have similar *topicality*

# Words-by-regions Matrices

The Vector Space Model:

| | $d_1$ | $d_2$ | $d_3$ | $d_4$ |
|---|---|---|---|---|
| *coffee* | 0.00 | 0.78 | 0.00 | 0.00 |
| *ipso* | 0.23 | 0.00 | 0.12 | 0.00 |
| *offside* | 0.65 | 0.00 | 0.42 | 0.00 |
| *football* | 0.89 | 0.00 | 0.54 | 0.12 |
| *espresso* | 0.00 | 0.00 | 0.00 | 0.00 |

Words that co-occur in many documents have similar *topicality*

# Words-by-regions Matrices

The Vector Space Model:

| | $d_1$ | $d_2$ | $d_3$ | $d_4$ |
|---|---|---|---|---|
| *coffee* | 0.00 | 0.78 | 0.00 | 0.00 |
| *ipso* | 0.23 | 0.00 | 0.12 | 0.00 |
| *offside* | 0.65 | 0.00 | 0.42 | 0.00 |
| *football* | 0.89 | 0.00 | 0.54 | 0.12 |
| *espresso* | 0.00 | 0.00 | 0.00 | 0.00 |

Words that co-occur in many documents have similar *topicality*

# Words-by-regions Matrices

The smaller the text regions, the more syntagmatic the relations

- Words-by-paragraphs
- Words-by-sentences
- Words-by-phrases

...but the worse the sparse data problem

# Words-by-regions Matrices

The smaller the text regions, the more syntagmatic the relations

- Words-by-paragraphs
- Words-by-sentences
- Words-by-phrases

...but the worse the sparse data problem

# Words-by-regions Matrices

The smaller the text regions, the more syntagmatic the relations

- Words-by-paragraphs
- Words-by-sentences
- Words-by-phrases

...but the worse the sparse data problem

# Words-by-regions Matrices

The smaller the text regions, the more syntagmatic the relations

- Words-by-paragraphs
- Words-by-sentences
- Words-by-phrases

...but the worse the sparse data problem

# Words-by-regions Matrices

The smaller the text regions, the more syntagmatic the relations

- Words-by-paragraphs
- Words-by-sentences
- Words-by-phrases

...but the worse the sparse data problem

# Latent Semantic Analysis (LSA)

*Latent Semantic Indexing (LSI)*

Developed to handle synonymy in information retrieval

Refinement of the standard vector space model

# Latent Semantic Analysis (LSA)

*Latent Semantic Indexing (LSI)*

Developed to handle synonymy in information retrieval

Refinement of the standard vector space model

# Latent Semantic Analysis (LSA)

*Latent Semantic Indexing (LSI)*

Developed to handle synonymy in information retrieval

Refinement of the standard vector space model

# Latent Semantic Analysis (LSA)

*Latent Semantic Indexing (LSI)*

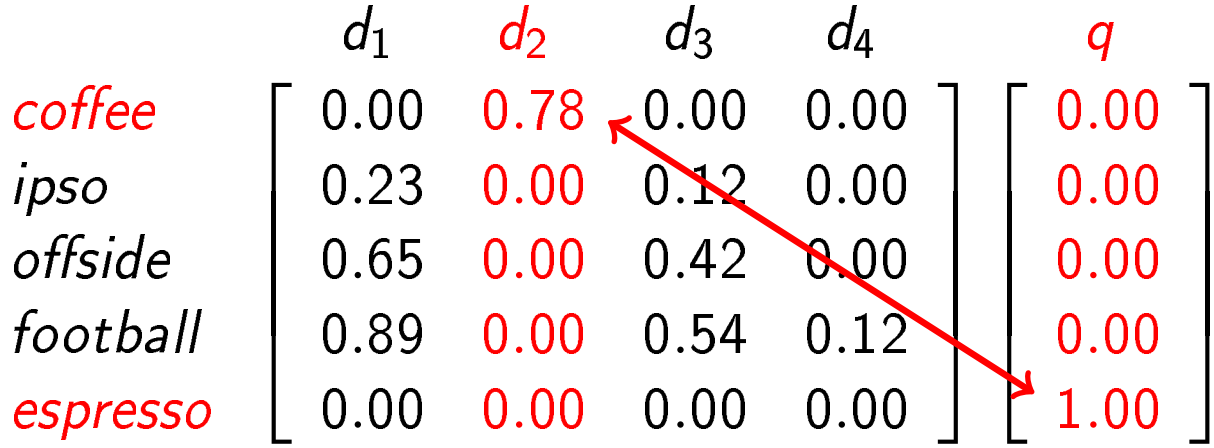
The Vector Space Model:

| | $d_1$ | $d_2$ | $d_3$ | $d_4$ | $q$ |
|---|---|---|---|---|---|
| *coffee* | 0.00 | 0.78 | 0.00 | 0.00 | 0.00 |
| *ipso* | 0.23 | 0.00 | 0.12 | 0.00 | 0.00 |
| *offside* | 0.65 | 0.00 | 0.42 | 0.00 | 0.00 |
| *football* | 0.89 | 0.00 | 0.54 | 0.12 | 0.00 |
| *espresso* | 0.00 | 0.00 | 0.00 | 0.00 | 1.00 |

# Latent Semantic Analysis (LSA)

*Latent Semantic Indexing (LSI)*

The Vector Space Model:



# Latent Semantic Analysis (LSA)

*Latent Semantic Indexing (LSI)*

We cannot trust raw occurrence counts (sparse data)

How do we uncover the "true" (or *latent*) occurrence counts?

# Latent Semantic Analysis (LSA)

*Latent Semantic Indexing (LSI)*

We cannot trust raw occurrence counts (sparse data)

How do we uncover the "true" (or *latent*) occurrence counts?

# Latent Semantic Analysis (LSA)

*Latent Semantic Indexing (LSI)*

Words-by-documents matrix (i.e. standard VSM):
$A_{m \times n}$

Weighting by row entropy:
$a_{m,n} = \log(\mathrm{tf}_{m,n} + 1) / - \sum_m p_{m,n} \log p_{m,n}$
where $p_{m,n} = \mathrm{tf}_{m,n} / \sum_{k=1}^{D} tf_{m,k}$

The matrix is reconstructed by truncated Singular Value Decomposition (SVD):
$A'_{m \times n} \approx U_{m \times k} S_{k \times k} V^T_{k \times n}$

# Latent Semantic Analysis (LSA)

*Latent Semantic Indexing (LSI)*

Words-by-documents matrix (i.e. standard VSM):
$A_{m \times n}$

Weighting by row entropy:
$a_{m,n} = \log(\mathrm{tf}_{m,n} + 1) / - \sum_m p_{m,n} \log p_{m,n}$
where $p_{m,n} = \mathrm{tf}_{m,n} / \sum_{k=1}^{D} tf_{m,k}$

The matrix is reconstructed by truncated Singular Value Decomposition (SVD):
$A'_{m \times n} \approx U_{m \times k} S_{k \times k} V^T_{k \times n}$

# Latent Semantic Analysis (LSA)

*Latent Semantic Indexing (LSI)*

Words-by-documents matrix (i.e. standard VSM):
$A_{m \times n}$

Weighting by row entropy:
$a_{m,n} = \log(\mathrm{tf}_{m,n} + 1) / - \sum_m p_{m,n} \log p_{m,n}$
where $p_{m,n} = \mathrm{tf}_{m,n} / \sum_{k=1}^{D} tf_{m,k}$

The matrix is reconstructed by truncated Singular Value Decomposition (SVD):
$A'_{m \times n} \approx U_{m \times k} S_{k \times k} V^T_{k \times n}$

# Latent Semantic Analysis (LSA)

*Latent Semantic Indexing (LSI)*

Singular Value Decomposition

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

$$A'_{m \times n} \approx U_{m \times k} S_{k \times k} V^T_{k \times n}$$

# Latent Semantic Analysis (LSA)

*Latent Semantic Indexing (LSI)*

Singular Value Decomposition

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

$$A'_{m \times n} \approx U_{m \times k} S_{k \times k} V^T_{k \times n}$$

# Latent Semantic Analysis (LSA)

*Latent Semantic Indexing (LSI)*

Singular Value Decomposition

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

$$A'_{m \times n} \approx U_{m \times k} S_{k \times k} V^T_{k \times n}$$

# Latent Semantic Analysis (LSA)

*Latent Semantic Indexing (LSI)*

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **human** | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| **interface** | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| **computer** | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **user** | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| **system** | 0 | 1 | 1 | 2 | 0 | 0 | 0 | 0 | 0 |
| **response** | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| **time** | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| **EPS** | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| **survey** | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| **trees** | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| **graph** | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| **minors** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

(Landauer, Foltz & Laham: Introduction to Latent Semantic Analysis, 1998)

# Latent Semantic Analysis

## *Latent Semantic Indexing (LSI)*

$U =$

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0.22 | -0.11 | 0.29 | -0.41 | -0.11 | -0.34 | 0.52 | -0.06 | -0.41 |
| 0.20 | -0.07 | 0.14 | -0.55 | 0.28 | 0.50 | -0.07 | -0.01 | -0.11 |
| 0.24 | 0.04 | -0.16 | -0.59 | -0.11 | -0.25 | -0.30 | 0.06 | 0.49 |
| 0.40 | 0.06 | -0.34 | 0.10 | 0.33 | 0.38 | 0.00 | 0.00 | 0.01 |
| 0.64 | -0.17 | 0.36 | 0.33 | -0.16 | -0.21 | -0.17 | 0.03 | 0.27 |
| 0.27 | 0.11 | -0.43 | 0.07 | 0.08 | -0.17 | 0.28 | -0.02 | -0.05 |
| 0.27 | 0.11 | -0.43 | 0.07 | 0.08 | -0.17 | 0.28 | -0.02 | -0.05 |
| 0.30 | -0.14 | 0.33 | 0.19 | 0.11 | 0.27 | 0.03 | -0.02 | -0.17 |
| 0.21 | 0.27 | -0.18 | -0.03 | -0.54 | 0.08 | -0.47 | -0.04 | -0.58 |
| 0.01 | 0.49 | 0.23 | 0.03 | 0.59 | -0.39 | -0.29 | 0.25 | -0.23 |
| 0.04 | 0.62 | 0.22 | 0.00 | -0.07 | 0.11 | 0.16 | -0.68 | 0.23 |
| 0.03 | 0.45 | 0.14 | -0.01 | -0.30 | 0.28 | 0.34 | 0.68 | 0.18 |

$S =$

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 3.34 | | | | | | | | |
| | 2.54 | | | | | | | |
| | | 2.35 | | | | | | |
| | | | 1.64 | | | | | |
| | | | | 1.50 | | | | |
| | | | | | 1.31 | | | |
| | | | | | | 0.85 | | |
| | | | | | | | 0.56 | |
| | | | | | | | | 0.36 |

$V^t =$

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0.20 | 0.61 | 0.46 | 0.54 | 0.28 | 0.00 | 0.01 | 0.02 | 0.08 |
| -0.06 | 0.17 | -0.13 | -0.23 | 0.11 | 0.19 | 0.44 | 0.62 | 0.53 |
| 0.11 | -0.50 | 0.21 | 0.57 | -0.51 | 0.10 | 0.19 | 0.25 | 0.08 |
| -0.95 | -0.03 | 0.04 | 0.27 | 0.15 | 0.02 | 0.02 | 0.01 | -0.03 |
| 0.05 | -0.21 | 0.38 | -0.21 | 0.33 | 0.39 | 0.35 | 0.15 | -0.60 |
| -0.08 | -0.26 | 0.72 | -0.37 | 0.03 | -0.30 | -0.21 | 0.00 | 0.36 |
| 0.18 | -0.43 | -0.24 | 0.26 | 0.67 | -0.34 | -0.15 | 0.25 | 0.04 |
| -0.01 | 0.05 | 0.01 | -0.02 | -0.06 | 0.45 | -0.76 | 0.45 | -0.07 |
| -0.06 | 0.24 | 0.02 | -0.08 | -0.26 | -0.62 | 0.02 | 0.52 | -0.45 |

(Landauer, Foltz & Laham: Introduction to Latent Semantic Analysis, 1998)

# Latent Semantic Analysis (LSA)

## *Latent Semantic Indexing (LSI)*

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| human | 0.16 | 0.40 | 0.38 | 0.47 | 0.18 | -0.05 | -0.12 | -0.16 | -0.09 |
| interface | 0.14 | 0.37 | 0.33 | 0.40 | 0.16 | -0.03 | -0.07 | -0.10 | -0.04 |
| computer | 0.15 | 0.51 | 0.36 | 0.41 | 0.24 | 0.02 | 0.06 | 0.09 | 0.12 |
| user | 0.26 | 0.84 | 0.61 | 0.70 | 0.39 | 0.03 | 0.08 | 0.12 | 0.19 |
| system | 0.45 | 1.23 | 1.05 | 1.27 | 0.56 | -0.07 | -0.15 | -0.21 | -0.05 |
| response | 0.16 | 0.58 | 0.38 | 0.42 | 0.28 | 0.06 | 0.13 | 0.19 | 0.22 |
| time | 0.16 | 0.58 | 0.38 | 0.42 | 0.28 | 0.06 | 0.13 | 0.19 | 0.22 |
| EPS | 0.22 | 0.55 | 0.51 | 0.63 | 0.24 | -0.07 | -0.14 | -0.20 | -0.11 |
| survey | 0.10 | 0.53 | 0.23 | 0.21 | 0.27 | 0.14 | 0.31 | 0.44 | 0.42 |
| trees | -0.06 | 0.23 | -0.14 | -0.27 | 0.14 | 0.24 | 0.55 | 0.77 | 0.66 |
| graph | -0.06 | 0.34 | -0.15 | -0.30 | 0.20 | 0.31 | 0.69 | 0.98 | 0.85 |
| minors | -0.04 | 0.25 | -0.10 | -0.21 | 0.15 | 0.22 | 0.50 | 0.71 | 0.62 |

(Landauer, Foltz & Laham: Introduction to Latent Semantic Analysis, 1998)

# Latent Semantic Analysis

*Latent Semantic Indexing (LSI)*

Topically similar words co-occur in documents

Induces higher-order ($\approx$ paradigmatic) relations through the truncated SVD

# Latent Semantic Analysis

*Latent Semantic Indexing (LSI)*

Topically similar words co-occur in documents

Induces higher-order ($\approx$ paradigmatic) relations through the truncated SVD

# Latent Semantic Analysis

*Latent Semantic Indexing (LSI)*

Examples of applications:

- Information retrieval
- Text categorization
- Recommender systems
- Spam filtering
- Bioinformatics

# Latent Semantic Analysis

*Latent Semantic Indexing (LSI)*

Examples of applications:

- Information retrieval
- Text categorization
- Recommender systems
- Spam filtering
- Bioinformatics

# Latent Semantic Analysis

*Latent Semantic Indexing (LSI)*

Examples of applications:

- Information retrieval
- Text categorization
- Recommender systems
- Spam filtering
- Bioinformatics

# Latent Semantic Analysis

*Latent Semantic Indexing (LSI)*

Examples of applications:

- Information retrieval
- Text categorization
- Recommender systems
- Spam filtering
- Bioinformatics

# Latent Semantic Analysis

*Latent Semantic Indexing (LSI)*

Examples of applications:

- Information retrieval
- Text categorization
- Recommender systems
- Spam filtering
- Bioinformatics

# Latent Semantic Analysis

*Latent Semantic Indexing (LSI)*

Examples of applications:

- Information retrieval
- Text categorization
- Recommender systems
- Spam filtering
- Bioinformatics

# Latent Semantic Analysis

*Latent Semantic Indexing (LSI)*

Summary:

- Words-by-documents co-occurrence matrix
- Frequency weighting by entropy
- Dimension reduction by truncated SVD
- Cosine similarity

# Latent Semantic Analysis

*Latent Semantic Indexing (LSI)*

Summary:

- Words-by-documents co-occurrence matrix
- Frequency weighting by entropy
- Dimension reduction by truncated SVD
- Cosine similarity

# Latent Semantic Analysis

*Latent Semantic Indexing (LSI)*

Summary:

- Words-by-documents co-occurrence matrix
- Frequency weighting by entropy
- Dimension reduction by truncated SVD
- Cosine similarity

# Latent Semantic Analysis

*Latent Semantic Indexing (LSI)*

Summary:

- Words-by-documents co-occurrence matrix
- Frequency weighting by entropy
- Dimension reduction by truncated SVD
- Cosine similarity

# Latent Semantic Analysis

*Latent Semantic Indexing (LSI)*

Summary:

- Words-by-documents co-occurrence matrix
- Frequency weighting by entropy
- Dimension reduction by truncated SVD
- Cosine similarity

# Words-by-regions Matrices

Many other probabilistic formulations, e.g.:

- Probabilistic Latent Semantic Analysis (pLSA)
- Latent Dirichlet Allocation (LDA)

*(SVD assumes Gaussian distributions, while the probabilistic formulations assume multinomial distributions)*

# Words-by-regions Matrices

Many other probabilistic formulations, e.g.:

- Probabilistic Latent Semantic Analysis (pLSA)
- Latent Dirichlet Allocation (LDA)

*(SVD assumes Gaussian distributions, while the probabilistic formulations assume multinomial distributions)*

# Words-by-regions Matrices

Many other probabilistic formulations, e.g.:

- Probabilistic Latent Semantic Analysis (pLSA)
- Latent Dirichlet Allocation (LDA)

*(SVD assumes Gaussian distributions, while the probabilistic formulations assume multinomial distributions)*

# Words-by-regions Matrices

Many other probabilistic formulations, e.g.:

- Probabilistic Latent Semantic Analysis (pLSA)
- Latent Dirichlet Allocation (LDA)

*(SVD assumes Gaussian distributions, while the probabilistic formulations assume multinomial distributions)*

# Words-by-words Matrices

Originates from psychology and computational linguistics

*Words* are natural contexts of words

# Words-by-words Matrices

Originates from psychology and computational linguistics

*Words* are natural contexts of words

# Words-by-words Matrices

| | $w_1$ | *drink* | $w_3$ | ... | $w_m$ |
|---|---|---|---|---|---|
| *coffee* | 0 | 1 | 0 | ... | 1 |
| $w_2$ | 1 | 0 | 2 | ... | 0 |
| *tea* | 0 | 2 | 0 | ... | 3 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋱ | ⋮ |
| $w_m$ | 0 | 0 | 1 | ... | 0 |

Paradigmatic similarity: words that co-occur with the same *other* words

Syntagmatic similarity by looking at the individual dimensions

# Words-by-words Matrices

| | $w_1$ | *drink* | $w_3$ | ... | $w_m$ |
|---|---|---|---|---|---|
| *coffee* | 0 | 1 | 0 | ... | 1 |
| $w_2$ | 1 | 0 | 2 | ... | 0 |
| *tea* | 0 | 2 | 0 | ... | 3 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋱ | ⋮ |
| $w_m$ | 0 | 0 | 1 | ... | 0 |

Paradigmatic similarity: words that co-occur with the same *other* words

Syntagmatic similarity by looking at the individual dimensions

# Words-by-words Matrices

Models differ with respect to:

- Context window
  *"I drink coffee at the cafe"*
  *"I drink some kind of black, hot and very tasty beverage at the cafe, and they tell me it is called coffee"*
- Frequency weighting
- Dimension reduction
- Similarity metric

# Words-by-words Matrices

Models differ with respect to:

- Context window

  *"I drink coffee at the cafe"*

  *"I drink some kind of black, hot and very tasty beverage at the cafe, and they tell me it is called coffee"*

- Frequency weighting
- Dimension reduction
- Similarity metric

# Words-by-words Matrices

Models differ with respect to:

- Context window
  *"I drink coffee at the cafe"*
  *"I drink some kind of black, hot and very tasty beverage at the cafe, and they tell me it is called coffee"*
- Frequency weighting
- Dimension reduction
- Similarity metric

# Words-by-words Matrices

Models differ with respect to:

- Context window
  *"I drink coffee at the cafe"*
  *"I drink some kind of black, hot and very tasty beverage at the cafe, and they tell me it is called coffee"*
- Frequency weighting
- Dimension reduction
- Similarity metric

# Words-by-words Matrices

Models differ with respect to:

- Context window
  *"I drink coffee at the cafe"*
  *"I drink some kind of black, hot and very tasty beverage at the cafe, and they tell me it is called coffee"*
- Frequency weighting
- Dimension reduction
- Similarity metric

# Words-by-words Matrices

Models differ with respect to:

- Context window
  *"I drink coffee at the cafe"*
  *"I drink some kind of black, hot and very tasty beverage at the cafe, and they tell me it is called coffee"*
- Frequency weighting
- Dimension reduction
- Similarity metric

# Words-by-words Matrices

Models differ with respect to:

- Context window
  *"I drink coffee at the cafe"*
  *"I drink some kind of black, hot and very tasty beverage at the cafe, and they tell me it is called coffee"*
- Frequency weighting
- Dimension reduction
- Similarity metric

# Hyperspace Analogue to Language (HAL)

Developed as a model of human semantic memory

Influenced by

- Psychology (Osgood and Deese)
- Computational linguistics (Schütze)
- Neural networks (Elman)

# Hyperspace Analogue to Language (HAL)

Developed as a model of human semantic memory

Influenced by

- Psychology (Osgood and Deese)
- Computational linguistics (Schütze)
- Neural networks (Elman)

# Hyperspace Analogue to Language (HAL)

Developed as a model of human semantic memory

Influenced by

- Psychology (Osgood and Deese)
- Computational linguistics (Schütze)
- Neural networks (Elman)

# Hyperspace Analogue to Language (HAL)

Developed as a model of human semantic memory

Influenced by

- Psychology (Osgood and Deese)
- Computational linguistics (Schütze)
- Neural networks (Elman)

# Hyperspace Analogue to Language (HAL)

Developed as a model of human semantic memory

Influenced by

- Psychology (Osgood and Deese)
- Computational linguistics (Schütze)
- Neural networks (Elman)

# Hyperspace Analogue to Language (HAL)

Sliding context window spanning 10 words

Co-occurrences are collected in only one direction

Frequency counts are weighted by distance in the context window

# Hyperspace Analogue to Language (HAL)

Sliding context window spanning 10 words

Co-occurrences are collected in only one direction

Frequency counts are weighted by distance in the context window

# Hyperspace Analogue to Language (HAL)

Sliding context window spanning 10 words

Co-occurrences are collected in only one direction

Frequency counts are weighted by distance in the context window

# Hyperspace Analogue to Language (HAL)

Directional words-by-words co-occurrence matrix:

| | *I* | *drink* | *coffee* | *tea* | *cafe* |
|---|---|---|---|---|---|
| *I* | 0 | 10 | 9 | 8 | 7 |
| *drink* | 0 | 0 | 10 | 9 | 8 |
| *coffee* | 0 | 0 | 0 | 10 | 9 |
| *tea* | 0 | 0 | 0 | 0 | 10 |
| *cafe* | 0 | 0 | 0 | 0 | 0 |

*"I drink coffee and tea at the cafe"*

# Hyperspace Analogue to Language (HAL)

Distributional vector = row + column ($2m$-dimensional)

Discard elements with low variance

# Hyperspace Analogue to Language (HAL)

Distributional vector = row + column ($2m$-dimensional)

Discard elements with low variance

# Hyperspace Analogue to Language (HAL)

Paradigmatic similarity

Directional word order information

# Hyperspace Analogue to Language (HAL)

Paradigmatic similarity

Directional word order information

# Hyperspace Analogue to Language (HAL)

Examples of applications:

- Priming
- Similarity ratings
- Word categorization

# Hyperspace Analogue to Language (HAL)

Examples of applications:

- Priming
- Similarity ratings
- Word categorization

# Hyperspace Analogue to Language (HAL)

Examples of applications:

- Priming
- Similarity ratings
- Word categorization

# Hyperspace Analogue to Language (HAL)

Examples of applications:

- Priming
- Similarity ratings
- Word categorization

# Hyperspace Analogue to Language (HAL)

Summary:

- Words-by-words co-occurrence matrix
- 10-word sliding context window
- Directional co-occurrences
- Distance weighting
- Concatenation of rows and columns
- (Dimension reduction by variance)
- Minkowski similarity measure

# Hyperspace Analogue to Language (HAL)

Summary:

- Words-by-words co-occurrence matrix
- 10-word sliding context window
- Directional co-occurrences
- Distance weighting
- Concatenation of rows and columns
- (Dimension reduction by variance)
- Minkowski similarity measure

# Hyperspace Analogue to Language (HAL)

Summary:

- Words-by-words co-occurrence matrix
- 10-word sliding context window
- Directional co-occurrences
- Distance weighting
- Concatenation of rows and columns
- (Dimension reduction by variance)
- Minkowski similarity measure

# Hyperspace Analogue to Language (HAL)

Summary:

- Words-by-words co-occurrence matrix
- 10-word sliding context window
- Directional co-occurrences
- Distance weighting
- Concatenation of rows and columns
- (Dimension reduction by variance)
- Minkowski similarity measure

# Hyperspace Analogue to Language (HAL)

Summary:

- Words-by-words co-occurrence matrix
- 10-word sliding context window
- Directional co-occurrences
- Distance weighting
- Concatenation of rows and columns
- (Dimension reduction by variance)
- Minkowski similarity measure

# Hyperspace Analogue to Language (HAL)

Summary:

- Words-by-words co-occurrence matrix
- 10-word sliding context window
- Directional co-occurrences
- Distance weighting
- Concatenation of rows and columns
- (Dimension reduction by variance)
- Minkowski similarity measure

# Hyperspace Analogue to Language (HAL)

Summary:

- Words-by-words co-occurrence matrix
- 10-word sliding context window
- Directional co-occurrences
- Distance weighting
- Concatenation of rows and columns
- (Dimension reduction by variance)
- Minkowski similarity measure

# Hyperspace Analogue to Language (HAL)

Summary:

- Words-by-words co-occurrence matrix
- 10-word sliding context window
- Directional co-occurrences
- Distance weighting
- Concatenation of rows and columns
- (Dimension reduction by variance)
- Minkowski similarity measure

# Correlated Occurrence Analogue to Lexical Semantics (COALS)

Developed as a variation of HAL in order to reduce frequency effects

Use (Pearson) *correlation* instead of raw frequency counts

# Correlated Occurrence Analogue to Lexical Semantics (COALS)

Developed as a variation of HAL in order to reduce frequency effects

Use (Pearson) *correlation* instead of raw frequency counts

# Correlated Occurrence Analogue to Lexical Semantics (COALS)

4+4 sized context window with distance weights

| I | drink | coffee | and | **tea** | at | the | cafe |
|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | | 4 | 3 | 2 |

No directional information

# Correlated Occurrence Analogue to Lexical Semantics (COALS)

4+4 sized context window with distance weights

| I | drink | coffee | and | **tea** | at | the | cafe |
|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | | 4 | 3 | 2 |

No directional information

# Correlated Occurrence Analogue to Lexical Semantics (COALS)

4+4 sized context window with distance weights

| I | drink | coffee | and | **tea** | at | the | cafe |
|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | | 4 | 3 | 2 |

No directional information

# Correlated Occurrence Analogue to Lexical Semantics (COALS)

Symmetric words-by-words co-occurrence matrix:

| | *I* | *drink* | *coffee* | *and* | *tea* | *at* | *the* | *cafe* |
|---|---|---|---|---|---|---|---|---|
| *I* | 0 | 4 | 3 | 2 | 1 | 0 | 0 | 0 |
| *drink* | 4 | 0 | 4 | 3 | 2 | 1 | 0 | 0 |
| *coffee* | 3 | 4 | 0 | 4 | 3 | 2 | 1 | 0 |
| *and* | 2 | 3 | 4 | 0 | 4 | 3 | 2 | 1 |
| *tea* | 1 | 2 | 3 | 4 | 0 | 4 | 3 | 2 |
| *at* | 0 | 1 | 2 | 3 | 4 | 0 | 4 | 3 |
| *the* | 0 | 0 | 1 | 2 | 3 | 4 | 0 | 4 |
| *cafe* | 0 | 0 | 0 | 1 | 2 | 3 | 4 | 0 |

*"I drink coffee and tea at the cafe"*

# Correlated Occurrence Analogue to Lexical Semantics (COALS)

Weight the co-occurrence counts with correlation:

$$a_{a,b} = \frac{\gamma a_{a,b} - \sum_m w_{a,m} \cdot \sum_n w_{n,b}}{\sqrt{\sum_m w_{a,m} \cdot (\gamma - \sum_m w_{a,m}) \cdot \sum_n w_{n,b} \cdot (\gamma - \sum_n w_{n,b})}}$$

where $\gamma = \sum_m \sum_n a_{m,n}$ (i.e. the matrix sum)

Set negative values to zero, and take the square root of positive values

# Correlated Occurrence Analogue to Lexical Semantics (COALS)

Weight the co-occurrence counts with correlation:

$$a_{a,b} = \frac{\gamma a_{a,b} - \sum_m w_{a,m} \cdot \sum_n w_{n,b}}{\sqrt{\sum_m w_{a,m} \cdot (\gamma - \sum_m w_{a,m}) \cdot \sum_n w_{n,b} \cdot (\gamma - \sum_n w_{n,b})}}$$

where $\gamma = \sum_m \sum_n a_{m,n}$ (i.e. the matrix sum)

Set negative values to zero, and take the square root of positive values

# Correlated Occurrence Analogue to Lexical Semantics (COALS)

100 000 most frequent words as columns

Discard low-frequent words

(Reduce dimensionality by truncated SVD)

Similarity by correlation measure

# Correlated Occurrence Analogue to Lexical Semantics (COALS)

100 000 most frequent words as columns

Discard low-frequent words

(Reduce dimensionality by truncated SVD)

Similarity by correlation measure

# Correlated Occurrence Analogue to Lexical Semantics (COALS)

100 000 most frequent words as columns

Discard low-frequent words

(Reduce dimensionality by truncated SVD)

Similarity by correlation measure

# Correlated Occurrence Analogue to Lexical Semantics (COALS)

100 000 most frequent words as columns

Discard low-frequent words

(Reduce dimensionality by truncated SVD)

Similarity by correlation measure

# Words-by-words Matrices

Several other variations of HAL:

- Size and configuration of the context window
- Frequency weighting
- Dimension reduction
- Similarity measure

# Words-by-words Matrices

Several other variations of HAL:

- Size and configuration of the context window
- Frequency weighting
- Dimension reduction
- Similarity measure

# Words-by-words Matrices

Several other variations of HAL:

- Size and configuration of the context window
- Frequency weighting
- Dimension reduction
- Similarity measure

# Words-by-words Matrices

Several other variations of HAL:

- Size and configuration of the context window
- Frequency weighting
- Dimension reduction
- Similarity measure

# Words-by-words Matrices

Several other variations of HAL:

- Size and configuration of the context window
- Frequency weighting
- Dimension reduction
- Similarity measure

# Dependency-based Models

Motivation:

Why only use sequential information when we can do better (by using our linguistic knowledge)?

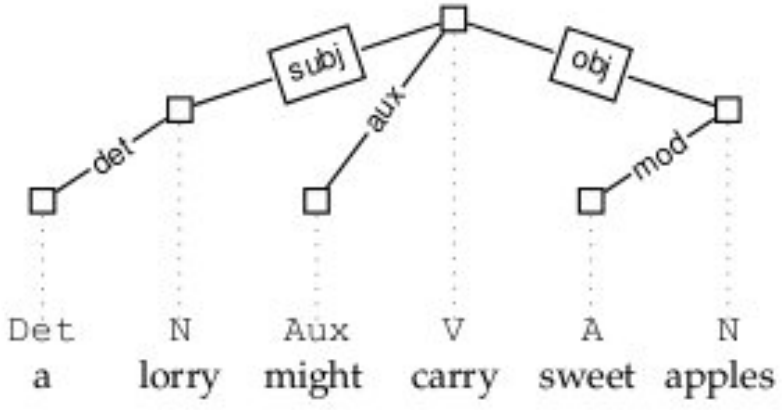
# Dependency-based Models

Motivation:

Why only use sequential information when we can do better (by using our linguistic knowledge)?

# Dependency-based Models

Dependency paths instead of context windows:




(Padó & Lapata: Dependency-based Construction of Semantic Space Models, 2007)

# Dependency-based Models

## Dependency paths as context windows

Only count contexts that are linked by a syntactic relation

`A big dog bites a tall man.`

| | bite | tall | big |
|---|---|---|---|
| dog | 1 | 0 | 1 |
| man | 1 | 1 | 0 |

# Dependency-based Models

Dependency paths as context windows

Only count contexts that are linked by a syntactic relation

A big dog bites a tall man.

| | bite | tall | big |
|---|---|---|---|
| dog | 1 | 0 | 1 |
| man | 1 | 1 | 0 |

# Dependency-based Models

Dependency paths as context windows

Only count contexts that are linked by a syntactic relation

`A big dog bites a tall man.`

| | bite | tall | big |
|---|---|---|---|
| dog | 1 | 0 | 1 |
| man | 1 | 1 | 0 |

# Dependency-based Models

Dependency tuples *(rel, word)* as dimensions

Take into account the type of syntactic dependency

`A dog bites a man.  A man bites a dog.  A dog bites a man.`

| | bite-subj | bite-obj |
|---|---|---|
| dog | 2 | 1 |
| man | 1 | 2 |

# Dependency-based Models

Dependency tuples *(rel, word)* as dimensions

Take into account the type of syntactic dependency

A dog bites a man. A man bites a dog. A dog bites a man.

| | bite-subj | bite-obj |
|---|---|---|
| dog | 2 | 1 |
| man | 1 | 2 |

# Dependency-based Models

Dependency tuples *(rel, word)* as dimensions

Take into account the type of syntactic dependency

`A dog bites a man.  A man bites a dog.  A dog bites a man.`

| | bite-subj | bite-obj |
|---|---|---|
| dog | 2 | 1 |
| man | 1 | 2 |

# Dependency-based Models

Pros: Linguistically motivated context

Cons: Non-trivial preprocessing, the results do not always motivate the added cost

# Dependency-based Models

Pros: Linguistically motivated context

Cons: Non-trivial preprocessing, the results do not always motivate the added cost

# Word Space Models

Summary:

- Context configuration
- Frequency weighting
- Dimension reduction
- Similarity metric

# Word Space Models

Summary:

- Context configuration
- Frequency weighting
- Dimension reduction
- Similarity metric

# Word Space Models

Summary:

- Context configuration
- Frequency weighting
- Dimension reduction
- Similarity metric

# Word Space Models

Summary:

- Context configuration
- Frequency weighting
- Dimension reduction
- Similarity metric

# Word Space Models

Summary:

- Context configuration
- Frequency weighting
- Dimension reduction
- Similarity metric

# Lab

Use S-space to:

- Build an LSA model
- Build a HAL model
- Build a COALS model

# Lab

Use S-space to:

- Build an LSA model
- Build a HAL model
- Build a COALS model

# Lab

Use S-space to:

- Build an LSA model
- Build a HAL model
- Build a COALS model

# Lab

Use S-space to:

- Build an LSA model
- Build a HAL model
- Build a COALS model