Computational Linguistics

Latent Spaces and Matrix Factorization

Clayton Greenberg & Stefan Thater

FR 4.7 Allgemeine Linguistik (Computerlinguistik)
Universität des Saarlandes
Summer 2016

Goal

Goal:

treat document clustering and word clustering on the same footing (same semantic space)

find low dimensional representations

The word document matrix

Clustering

Document clustering

describe each document by a vector containing the frequencies of the words

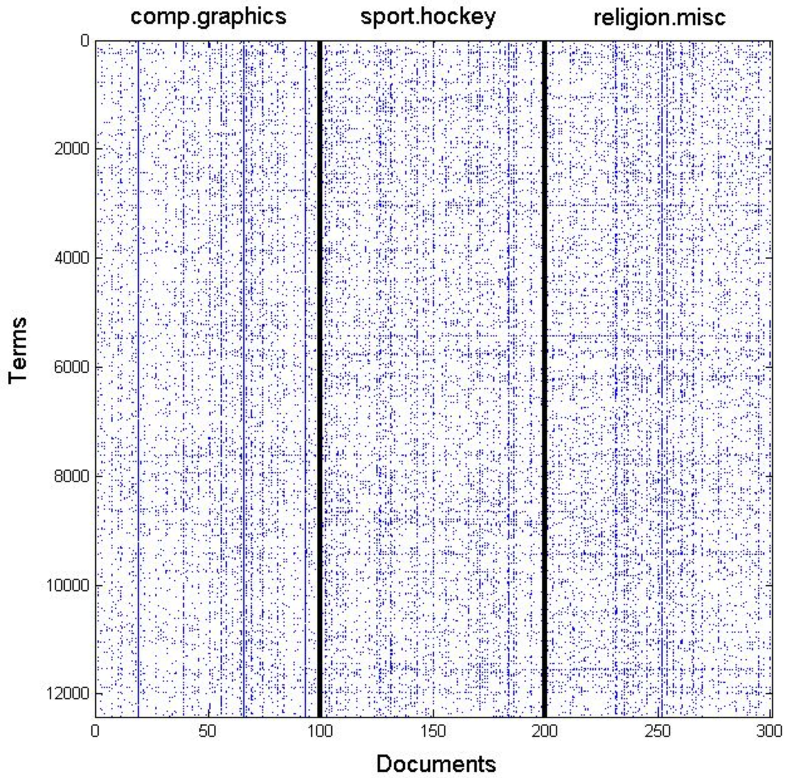
Word clustering

describe each word by a vector containing the frequencies of its occurrence in different documents

Joint word and document clustering

The word document matrix:

Enter frequency (or tf-idf) for each word and document in a rectangular scheme of numbers (matrix)



Matrices

Matrices

A matrix is an array with two indices

e.g. in a python program this could be `A[i][j]` with i=1..N and j=1...M

When writing, often a subscript notation is used $a_{i,j}$

or a square scheme:

$$A = \begin{pmatrix} a_{1,1} & \dots & a_{1,M} \\ \dots & a_{i,j} & \dots \\ a_{N,1} & \dots & a_{N,M} \end{pmatrix}$$

Specific example of a 2x3 matrix

$$A = \begin{pmatrix} 2 & -5 & 0.5 \\ -2 & 0.1 & -8 \end{pmatrix}$$

The transpose of a matrix

The two indices are swapped

e.g. in a python program this could be `At[j][i]=A[i][j]` for i=1..N and j=1...M

for the general matrix on the previous slide we have:

$$A^t = \begin{pmatrix} a_{1,1} & \dots & a_{1,N} \\ \dots & a_{j,i} & \dots \\ a_{M,1} & \dots & a_{M,N} \end{pmatrix}$$

Specific example of a 2x3 matrix

$$A = \begin{pmatrix} 2 & -5 & 0.5 \\ -2 & 0.1 & -8 \end{pmatrix}$$

What is A^t

Product of two matrices

The elements of a product matrix can be calculated in a python program by

```
for i in range(1,N+1):
    for j in range(1,M+1):
        for k in range(1,K+1):
            C[i][j] += A[i][k]*B[k][j]
```

In math notation $C = A \cdot B$

with

$$c_{i,j} = \sum_{k=1}^{K} a_{i,k} b_{k,j}$$

Example see black board

Unit matrix

Unit matrix: the element are the indicator function

$$a_{i,j} = \delta_{i,j}$$

Example:

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Often the unit matrix is denoted by a 1

Orthogonal matrices

a matrix A is orthogonal if

$$1 = A^t \cdot A$$

Is the following matrix orthogonal:

$$A = \begin{pmatrix} 0.96 & -0.28 \\ 0.28 & 0.96 \end{pmatrix}$$

In class matrices exercise

$$A = \begin{pmatrix} 2 & 1 \\ 0 & -2 \end{pmatrix}$$

$$B = \begin{pmatrix} -1 & -2 & 7 \\ -2 & 2 & -1 \end{pmatrix}$$

$$C = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

1. What is A ˙ B?
2. What is B ˙ A?
3. What is B^t?
4. Is C orthogonal?

Matrices in python

See http://wiki.scipy.org/Tentative_NumPy_Tutorial#head-a9063f71090f3d1fbbdae5397ccb4e882d2cf603

Simple Array Operations

See linalg.py in numpy folder for more.

```
>>> from numpy import *
>>> from numpy.linalg import *

>>> a = array([[1.0, 2.0], [3.0, 4.0]])
>>> print a
[[ 1.  2.]
 [ 3.  4.]]

>>> a.transpose()
array([[ 1.,  3.],
       [ 2.,  4.]])

>>> inv(a)
array([[-2. ,  1. ],
       [ 1.5, -0.5]])

>>> u = eye(2) # unit 2x2 matrix; "eye" represents "I"
>>> u
array([[ 1.,  0.],
       [ 0.,  1.]])
>>> j = array([[0.0, -1.0], [1.0, 0.0]])

>>> dot (j, j) # matrix product
array([[-1.,  0.],
       [ 0., -1.]])

>>> trace(u)  # trace
2.0

>>> y = array([[5.], [7.]])
>>> solve(a, y)
array([[-3.],
       [ 4.]])

>>> eig(j)
(array([ 0.+1.j,  0.-1.j]),
array([[ 0.70710678+0.j,  0.70710678+0.j],
       [ 0.00000000-0.70710678j,  0.00000000+0.70710678j]]))
Parameters:
    square matrix

Returns
```

Latent Semantic Analysis (LSA)

This section mostly follows Manning and Schütze
Chapter 15

Singular Value Decomposition (SVD)

Decompose A such that

$$\widetilde{A} = TSD^t$$

With $| \widetilde{A} - A |^2$ minimal

and

$$T^t \cdot T = 1 \qquad D^t \cdot D = 1$$

A a t by d matrix T a t by n matrix

S a n by n matrix D a d by n matrix

An artificial example of SVD

Is

$$T = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix} \qquad S = \left(2\sqrt{2}\right) \qquad D = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \\ -\frac{1}{2} \\ -\frac{1}{2} \end{pmatrix}$$

an SVD of

$$A = \begin{pmatrix} 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 \end{pmatrix}$$

More realistic Example
(from Manning and Schütze)

Decompose

$$A = \left(\begin{array}{l|cccccc} & d_1 & d_2 & d_3 & d_4 & d_5 & d_6 \\ \hline \text{cosmonaut} & 1 & 0 & 1 & 0 & 0 & 0 \\ \text{astronaut} & 0 & 1 & 0 & 0 & 0 & 0 \\ \text{moon} & 1 & 1 & 0 & 0 & 0 & 0 \\ \text{car} & 1 & 0 & 0 & 1 & 1 & 0 \end{array}\right)$$

More realistic Example

(from Manning and Schütze)

$D^t =$

	d_1	d_2	d_3	d_4	d_5	d_6
Dimension 1	−0.75	−0.28	−0.20	−0.45	−0.33	−0.12
Dimension 2	−0.29	−0.53	−0.19	0.63	0.22	0.41
Dimension 3	0.28	−0.75	0.45	−0.20	0.12	−0.33
Dimension 4	0.00	0.00	0.58	0.00	−0.58	0.58
Dimension 5	−0.53	0.29	0.63	0.19	0.41	−0.22

$T^t =$

	cosm.	astr.	moon	car	truck
Dimension 1	−0.44	−0.13	−0.48	−0.70	−0.26
Dimension 2	−0.30	−0.33	−0.51	0.35	0.65
Dimension 3	0.57	−0.59	−0.37	0.15	−0.41
Dimension 4	0.58	0.00	0.00	−0.58	0.58
Dimension 5	0.25	0.73	−0.61	0.16	−0.09

More realistic Example

(from Manning and Schütze)

$$S = \begin{pmatrix} 2.16 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 1.59 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 1.28 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 1.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.39 \end{pmatrix}$$

Document-Document Similarity

Rewrite A

$$A = \left(\vec{d}_1 \quad \vec{d}_2 \quad \ldots \quad \vec{d}_d\right)$$

with $\vec{d}_j$ a vector

with word frequencies of the j-th document

Similarity of i-th document with j-th document $\vec{d}_i^{\,t}\,\vec{d}_j$

All document-document similarities $A^t A$

Document-Document Similarity

Rewrite $\tilde{A}^t \tilde{A} =$

$$= (TSD^t)^t TSD^t$$

$$= DS^t T^t TSD^t$$

$$= DS^t SD^t$$

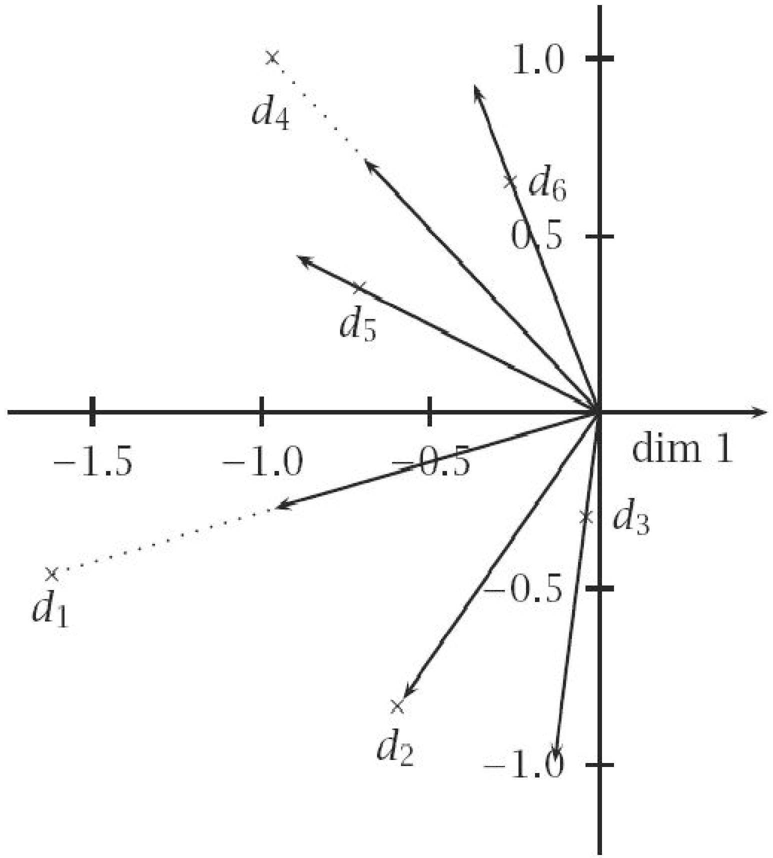
$$= (SD^t)^t SD^t$$

Measure similarity in subspace defined by SD^t

More realistic example
(from Manning and Schütze)

Result for SD^t

	d_1	d_2	d_3	d_4	d_5	d_6
Dimension 1	−1.62	−0.60	−0.04	−0.97	−0.71	−0.26
Dimension 2	−0.46	−0.84	−0.30	1.00	0.35	0.65



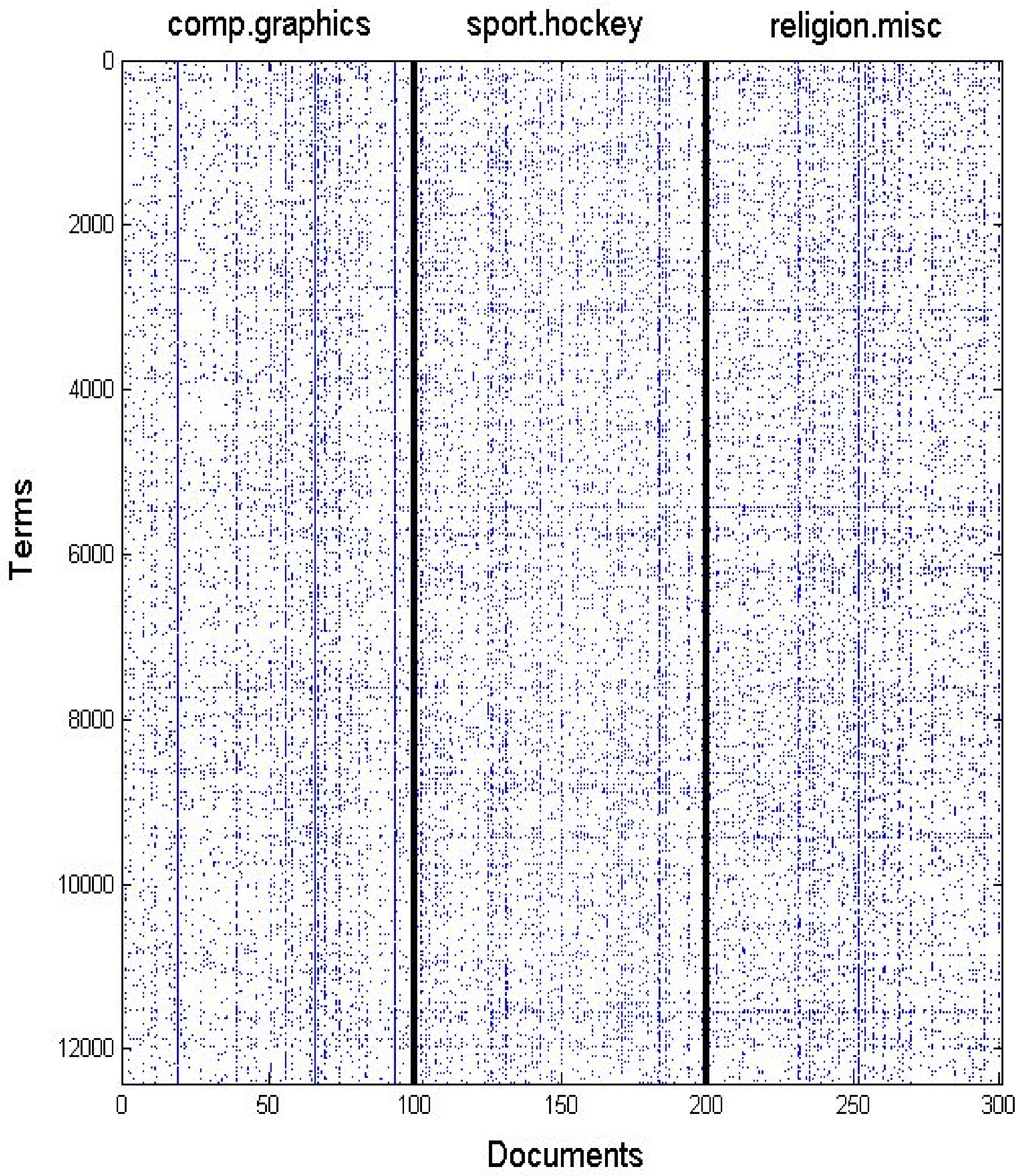
More realistic example

(from Manning and Schütze)

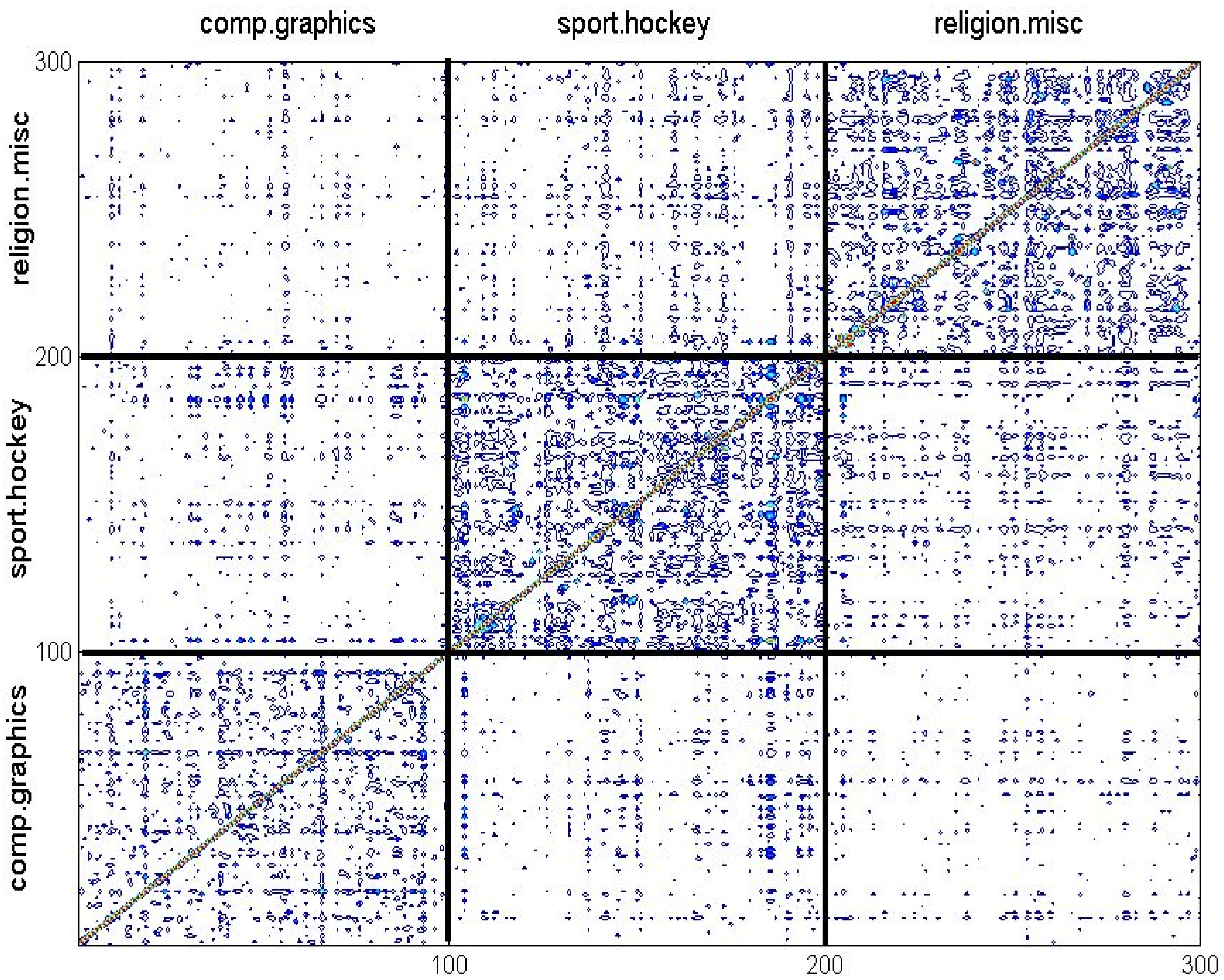
Decompose A such that

	d_1	d_2	d_3	d_4	d_5	d_6
d_1	1.00					
d_2	0.78	1.00				
d_3	0.40	0.88	1.00			
d_4	0.47	−0.18	−0.62	1.00		
d_5	0.74	0.16	−0.32	0.94	1.00	
d_6	0.10	−0.54	−0.87	0.93	0.74	1.00

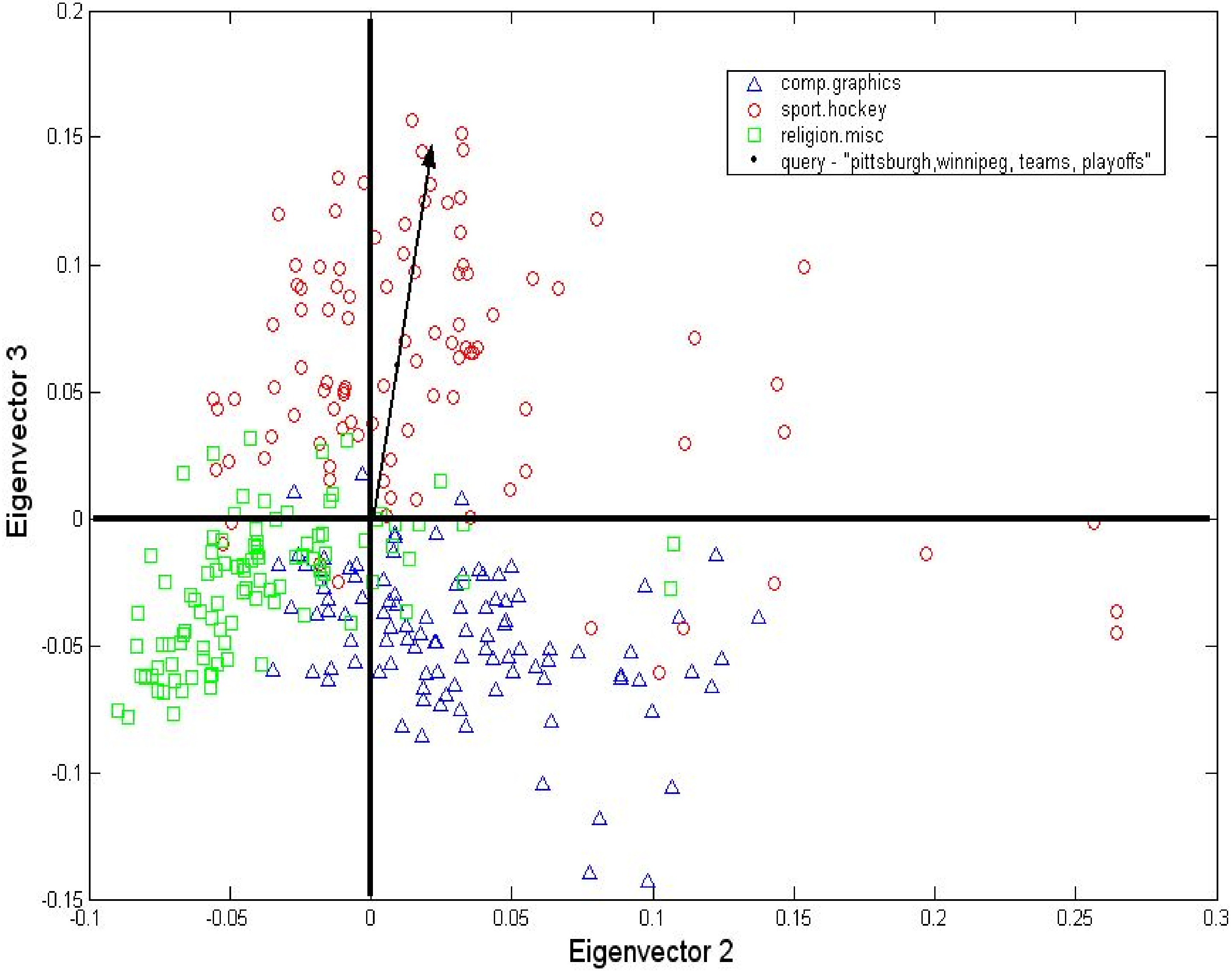
An even more realistic example



An even more realistic example document-document similarity



Representation for documents in 2 dimensional subspace



Term-term similarity

Rewrite $\tilde{A}\tilde{A}^t =$

$$= (TSD^t)(TSD^t)^t$$

$$= TSD^t DS^t T^t$$

$$= TS^t ST^t$$

$$= (TS)(TS)^t$$

Measure similarity in subspace defined by TS

Homework

Implement SVD.

Details in exercise 9.

LSA performance

- LSA consistently improves recall on standard test collections (precision/recall generally improved)
- Variable performance on larger TREC collections
- Dimensionality of latent space – a magic number – 300 – 1000 seems to work fine – no satisfactory way of assessing value.
- Computational cost high

Application (by Landauer et al.)

**How Well Can Passage Meaning be Derived without Using Word Order?
A Comparison of Latent Semantic Analysis and Humans**

Thomas K. Landauer, Darrell Laham, Bob Rehder, and M. E. Schreiner
Department of Psychology & Institute of Cognitive Science
University of Colorado, Boulder
Boulder, CO 80309-0345
{landauer, dlaham, rehder, missy}@psych.colorado.edu

Rate essay by similarity to existing ones
Measure correlation with human rating

Correlation between	
All Essays (n = 273)	
Two reader scores:	.65
LSA score and average reader score:	.64
Attachment in children (n = 55)	
Two reader scores:	.19
LSA score and average reader score:	.61
Aphasias (n = 109)	
Two reader scores:	.75
LSA score and average reader score:	.60
Operant conditioning (n = 109)	
Two reader scores:	.68
LSA score and average reader score:	.71

Table 2: Psychology essay results.

**Conclusion: drop the right key-words in an exam
and you are set**