Text mining: bag of words representation and beyond it

Jasminka Dobša
Faculty of Organization and Informatics
University of Zagreb

Outline

- Definition of text mining
- Vector space model or Bag of words representation
  - Representation of documents and queries in VSM
  - Measure of similarity between document and query
  - Example
- Sematic representation of documents
  - Latent semantic indexing
  - Concept indexing
  - Example

Definition and basic concepts

- Text mining: part of the wider discipline of data mining
- Content-based processing of unstructured text documents and extraction of useful information from them
- Treatment based only on the content of the documents and not on metadata (date of publication, source of publication, type of document ...)
- Basic text mining tasks:
  - Information retrieval
  - Automatic classification of text documents
  - Clustering of text documents

Information retrieval: definition and basic concepts

- The problem: determining a set of documents from a collection that are relevant to a particular user query
- Assessment of the relevance of documents is based on a comparison of index terms that appear in texts of documents and text of the query
- Index term: any word or group of words that appear in the text
- When the text is represented by set of index terms much of the semantics contained in the text is lost
- Information about the relations between index terms is lost

Labels and basic models

- Labels:
  - d_j - document of collection of documents
  - q - query
  - $\mathbf{a}_i$, $\mathbf{a}_j$ – vector representations of documents
  - **q** - vector representation of the query

- Basic models:
  - Probabilistic
  - Boolean model
  - Vector space model or bag of words representation

Vector space model or Bag of words model

- Vector space model is today one of the most popular models for information retrieval
- It is developed by Gerald Salton and associates within the search system SMART (System for the Mechanical Analysis and Retrieval of Text)
- Similarity between documents and queries can take real values in the interval [0,1]
- Indexing terms are joined by weight
- Documents are ranked according to the degree of similarity with the query
- Representation in the vector space model called Bag of words representation (BOW model)
- BOW model assumes independence of indexing terms

Term-document matrix

- Notation
  - $T = \{t_1, t_2, \ldots, t_m\}$ - set of index terms
  - $D = \{d_1, d_2, \ldots, d_n\}$ - set of documents
  - a_{ij} weight associated to pair (t_i , d_j) , i=1,2,..,m; j=1,2,...,n
  - q_i , i=1,2,...,m weights associated to pair (t_i ,q), q query

- Document d_j , j=1,2,...n is represented by the vector

$$\mathbf{a}_j = (a_{j1}, a_{j2}, \ldots, a_{jm})^T$$

- Query is represented by the vector $\mathbf{q}=(q_1, q_2, \ldots, q_m)^T$
- Collection od documents is represented by term-document matrix

Term-document matrix

$$A = \begin{array}{c} \begin{array}{cccc} d_1 & d_2 & & d_n \\ \downarrow & \downarrow & & \downarrow \end{array} \\ \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \end{array} \begin{array}{l} \leftarrow t_1 \\ \leftarrow t_2 \\ \vdots \\ \leftarrow t_m \end{array}$$

Rows: terms
Columns: documents

- The terms are axes in the space
- Documents and queries are points or vectors in space
- Space is high-dimensional - dimension corresponds to the number of index terms
- Documents usually contain only a few index terms from the list of index terms
- Vectors representing documents are rare: most coordinates are equal to 0

Measure of similarity

- The similarity between the document and the query is measured by the angle between their representations

- The measure of similarity is cosine of the angle between representations of document and a query

$$\cos(\mathbf{a}_j, \mathbf{q}) = \frac{\mathbf{a}_j^T \mathbf{q}}{\|\mathbf{a}_j\|_2 \|\mathbf{q}\|_2}$$

Measure of similarity

- Vector representations of documents usually are standardized so that its Euclidean norm is 1

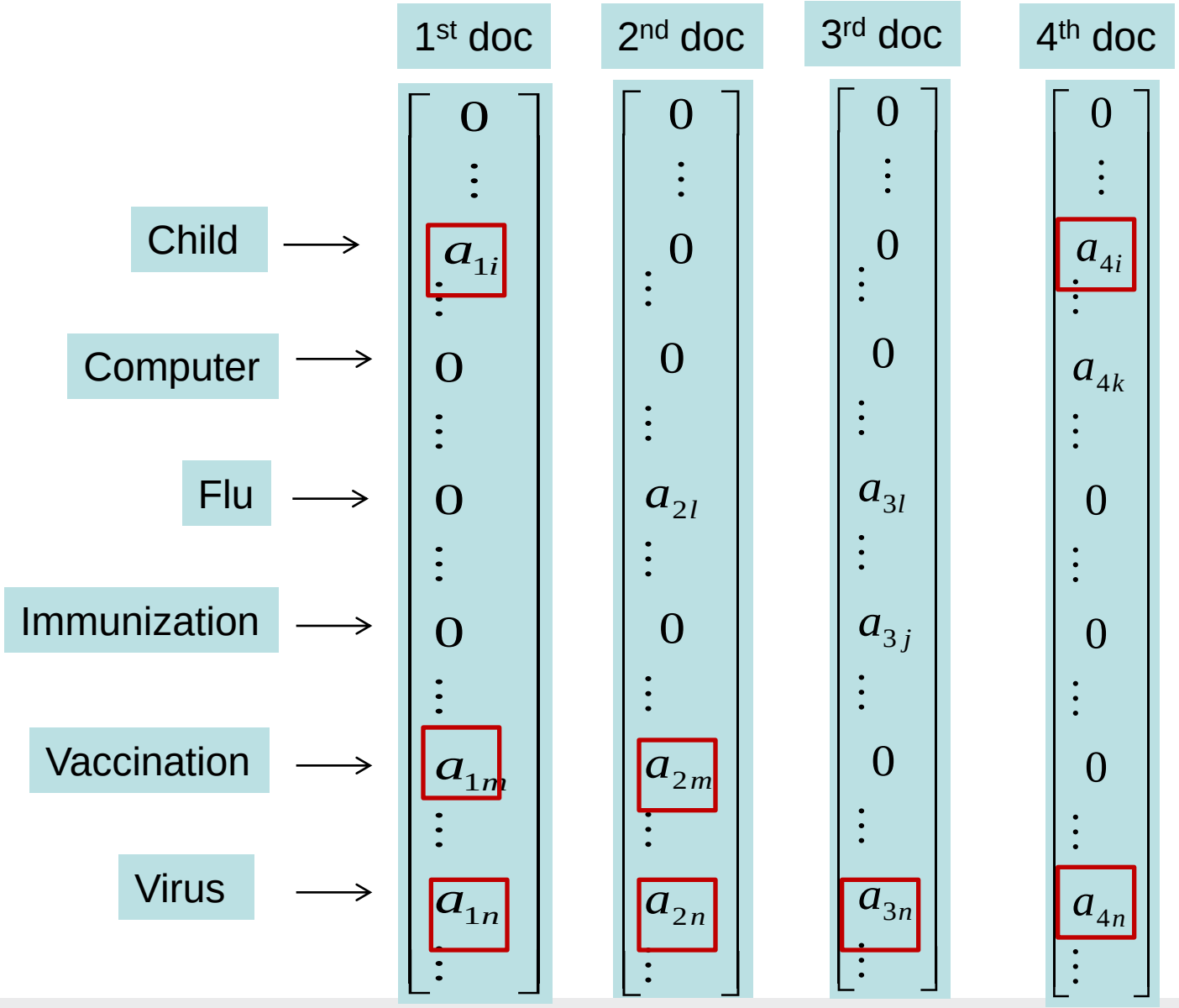
$$\|\mathbf{a}_j\|_2 = 1$$

- Norm of a representation of the query $\|\mathbf{q}\|_2$ does not affect the ranking of documents by relevance
- The measure of similarity between document and query usually is measured by the dot product of vectors in the numerator

Example: BOW representation

- 1st document
  - About vaccination of children against some virus
- 2nd and 3rd document
  - About vaccination of citizens aginst flu
- 4th document
  - About computer viruses

Example: BOW representation



Example: BOW representation

- Documents will be similar if there is term matching between them
- If document does not contain exact word set as the query, but synonym of that word, document will not be recognized as relevant to the query
- Documents containing polysemy will be recognized as similar although they actually are not similar

Evaluation of information retrieval

- Measures:
  - Recall
  - Precision
  - Average precision
- Recall

$$recall_i = \frac{r_i}{r_n}$$

- Precision

$$precision_i = \frac{r_i}{i}$$

- r_i number of relevant documents among i highest ranking documents
- r_n total number of relevant documents in collection
- Insisting on high recall reduces precision of retrieval and vice versa

- Average precision

average precision on more recall revels (usually 11)

Assigning weights to index terms

- Components:
  - Local - depends on the frequency of occurrence of the term in a particular document (Term frequency - TF component)
  - Global - factor for the index term depends inversely on the number of documents in which the term appears (the inverse of the frequency of the term in the collection (Inverse document frequency - IDF component))
  - Normalization – vectors of representations of a documents are divided by Euclidean norm of representation (this avoids the situation that longer documents are often returned as relevant)

Example

- A collection of 15 documents (titles of books)
  - 9 from the field of data mining
  - 5 from the field of linear algebra
  - One that combines these two areas (application of linear algebra to data mining)

- A list of words is formed as follows:
  - From the words contained in at least two documents
  - The words on the list of stop words are discarded
  - The words are reduced to their basic forms, ie. variations of the word are mapped to the basic version of the word

Documents 1/2

D1	Survey of text mining: clustering, classification, and retrieval
D2	Automatic text processing: the transformation analysis and retrieval of information by computer
D3	Elementary linear algebra: A matrix approach
D4	Matrix algebra & its applications statistics and econometrics
D5	Effective databases for text & document management
D6	Matrices, vector spaces, and information retrieval
D7	Matrix analysis and applied linear algebra
D8	Topological vector spaces and algebras

Documents 2/2

D9	Information retrieval: data structures & algorithms
D10	Vector spaces and algebras for chemistry and physics
D11	Classification, clustering and data analysis
D12	Clustering of large data sets
D13	Clustering algorithms
D14	Document warehousing and text mining: techniques for improving business operations, marketing and sales
D15	Data mining and knowledge discovery

Terms

Data mining terms	Linear algebra terms	Neutral terms
Text	Linear	Analysis
Mining	Algebra	Application
Clustering	Matrix	Algorithm
Classification	Vector	
Retrieval	Space	
Information		
Document		
Data		

The concepts are mapped in their basic versions, eg. Applications, Applied → Application

Queries

- Q1: Data mining
  - Relevant documents : All data mining documents (blue)
- Q2: Using linear algebra for data mining
  - Relevant document: D6

weight	bxx		tfn	
Document	Q1	Q2	Q1	Q2
D1	0,4472	0,4472	0,3607	0,2424
D2	0	0	0	0
D3	0	1,1547	0	0,6417
D4	0	0,5774	0	0,1471
D5	0	0	0	0
D6	0	0	0	0
D7	0	0,8944	0	0,4598
D8	0	0,5774	0	0,1654
D9	0,5	0,5	0,2634	0,1771
D10	0	0,5774	0	0,1654
D11	0,5	0,5	0,2634	0,1770
D12	0,7071	0,7071	0,4488	0,3016
D13	0	0	0	0
D14	0,5774	0,5774	0,4092	0,2750
D15	1,4142	1,4142	1,0000	0,6720

Comments on search results

- Consider that in the process of search are returned all documents whose measure of similarity with the query is greater than 0
- For the first query (Data mining)
  - All documents are returned in the search process are relevant
  - Not all relevant documents are returned in the search
  - Only those documents containing the terms contained in the query are returned - search the lexical, but not semantical
- For the second query (Using linear algebra for data mining)
  - None of the returned document is relevant
  - The only relevant document is not returned in the search
- Search using weight *bxx* and *tfn* did not result in significantly different results presented in this case

CONCEPTUAL INDEXING

Definition

- Conceptual indexing is indexing by features which recognize relations between terms
- Here we compare two techniques for conceptual indexing based on projection of vectors of documents (in means of least squares) on lower-dimensional vector space
  - Latent semantic indexing (LSI) - Benchmark
  - Concept indexing (CI)

Latent semantic indexing

- Introduced in 1990; improved in 1995
- S. Deerwester, S. Dumas, G. Furnas, T. Landauer, R. Harsman: *Indexing by latent semantic analysis,* J. American Society for Information Science, 41, 1990, pp. 391-407
- M. W. Berry, S.T. Dumas, G.W. O'Brien: *Using linear algebra for intelligent information retrieval,* SIAM Review, 37, 1995, pp. 573-595
- Based on spectral analysis of term-document matrix

Latent semantic indexing

- For LSI truncated SVD is used

$$A_k = U_k \Sigma_k V_k^T$$

where

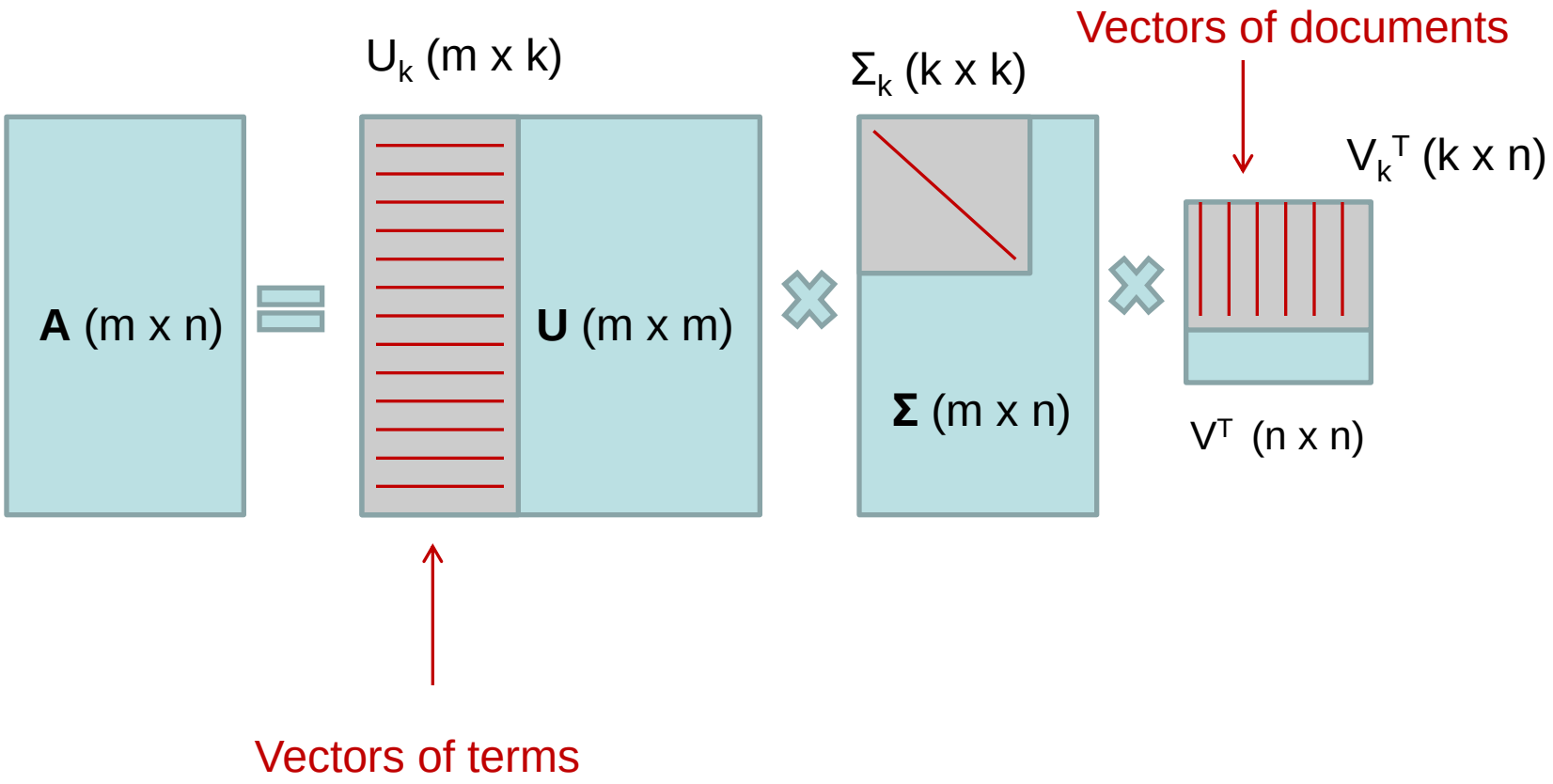
U_k is $m \times k$ matrix whose columns are first k left singular vectors of A

Σ_k is $k \times k$ diagonal matrix whose diagonal is formed by k leading singular values of A

V_k is $n \times k$ matrix whose columns are first k right singular vectors of A

- Rows of U_k = terms
- Rows of V_k = documents

LSI: Representations



Concept indexing (CI)

- Instead of SVD it is used concept decomposition (CD)
- CD was introduced in 2001

I.S.Dhillon, D.S. Modha: *Concept decomposition for large sparse text data using clustering,* Machine Learning, 42:1, 2001, pp. 143-175

Concept decomposition

- First step: clustering of documents in term-document matrix A on k groups
  - Clustering algorithms used:
    - k-means algorithm
    - Fuzzy k-means algorithm
  - Centroids of groups = concept vectors
  - Concept matrix is matrix whose columns are centroids of groups

$$C_k = \begin{bmatrix} c_1 & c_2 & \cdots & c_k \end{bmatrix}$$

c_j – centroid of j-th group

Concept decomposition

- Second step: computing of the CD
  - It is obtained by solving the least squares problem

$$\left\|\mathbf{A}-\mathbf{C}_k\mathbf{Z}\right\|^2 \rightarrow \min$$

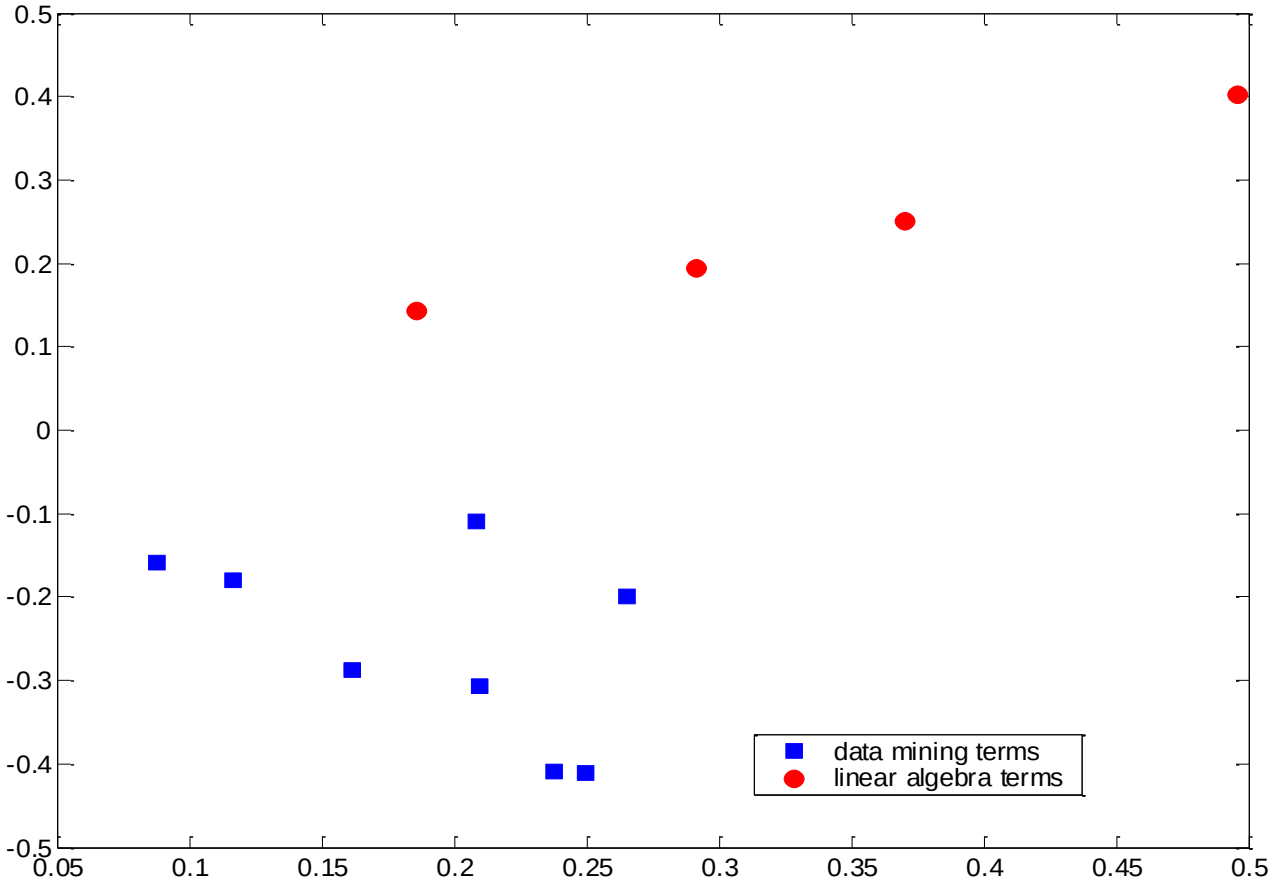
  - Solution to this problem is

Concept decomposition

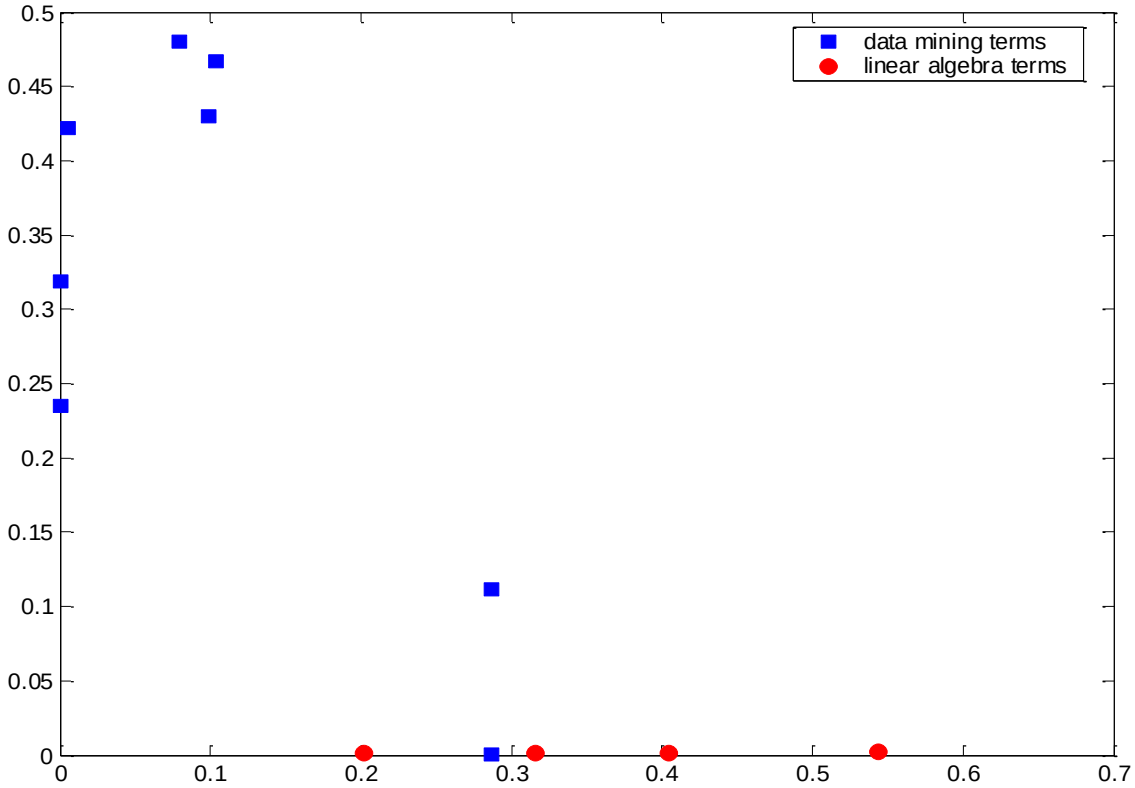
$$\mathbf{Z}=\left(\mathbf{C}_k^T\mathbf{C}_k\right)^{-1}\mathbf{C}_k^T\mathbf{A}$$

- Rows of C_k = terms
- Columns of Z = documents

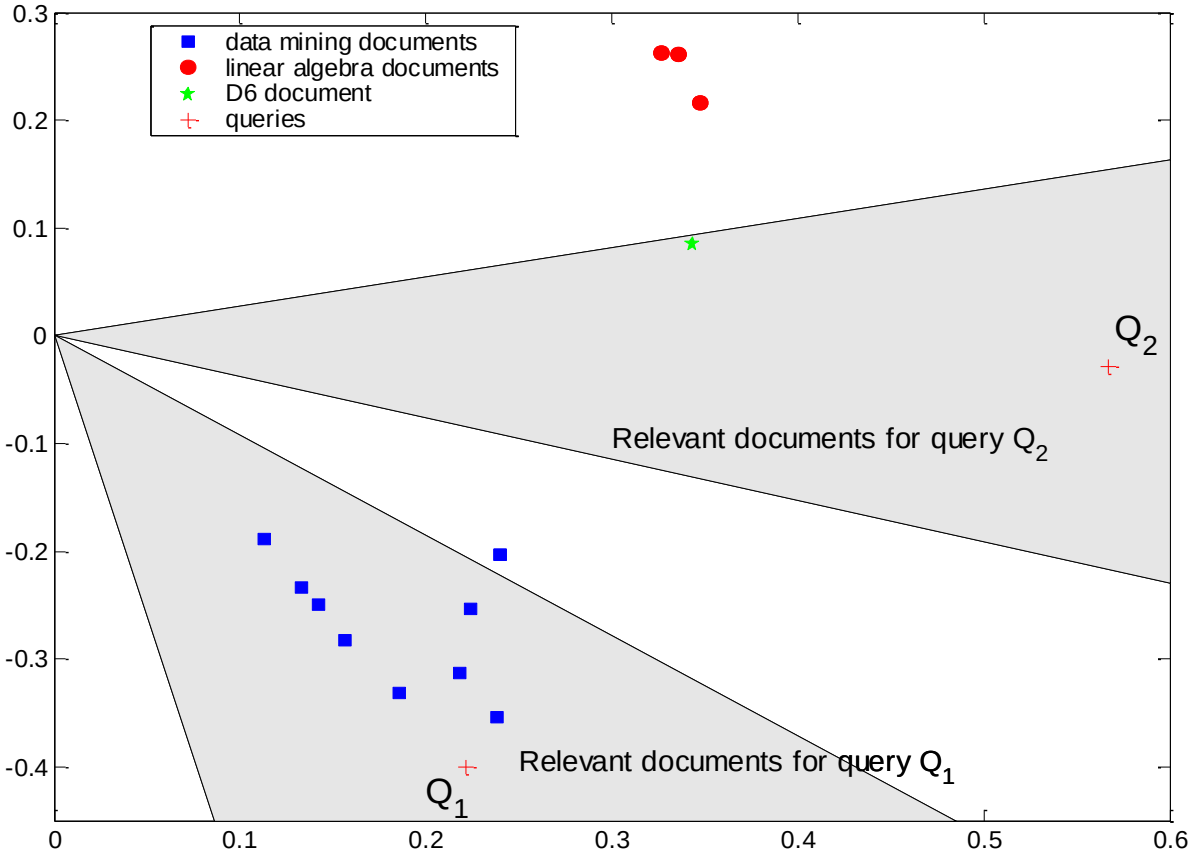
Example: Projection of terms by SVD



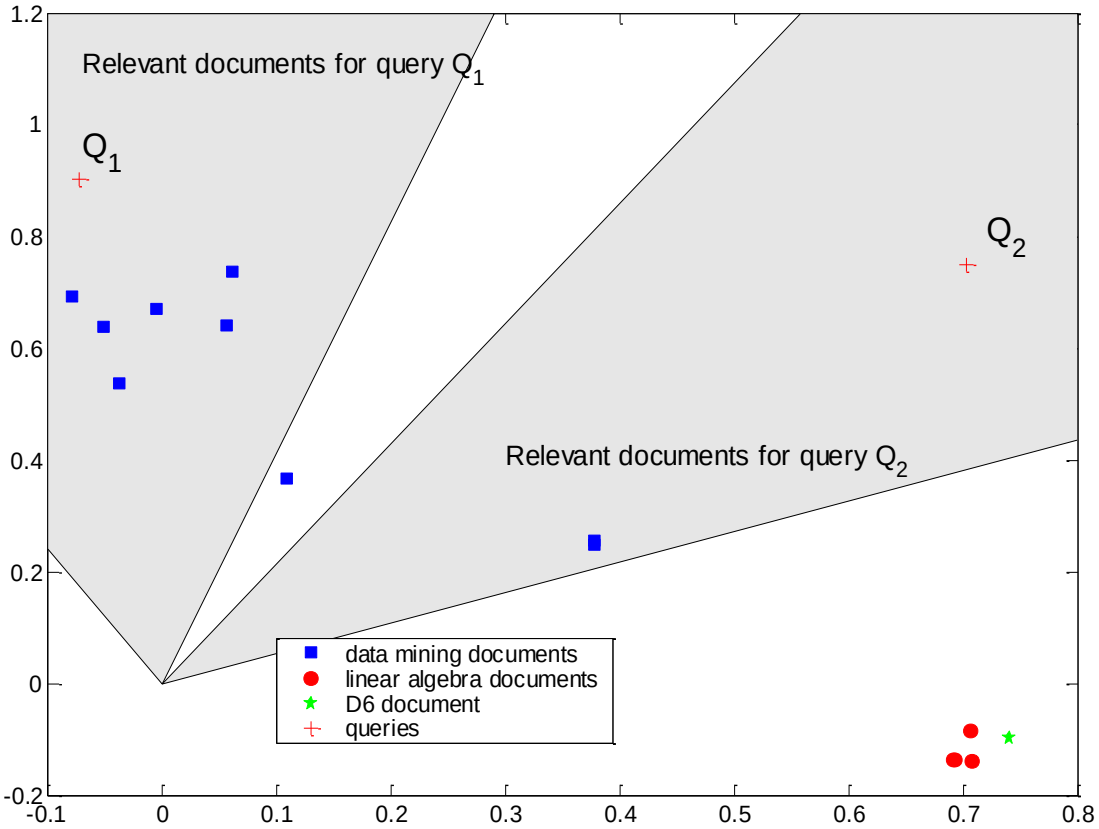
Example: Projection of terms by CD



Example: Projection of documents by SVD



Example: Projection of documents by CD



Results of information retrieval (Q1)

Term-maching method		Latent semantic indexing		Concept indexing	
score	document	score	document	score	document
1,4142	D15	0,6141	D1	0,6622	D1
0,7071	D12	0,5480	D11	0,6057	D14
0,5774	D14	0,5465	D12	0,5953	D12
0,5000	D9	0,4809	D9	0,5763	D11
0,5000	D11	0,4644	D15	0,5619	D15
0,4472	D1	0,4301	D2	0,4727	D5
0,0000	D2	0,4127	D14	0,3379	D13
0,0000	D3	0,3858	D13	0,2690	D2
0,0000	D4	0,3165	D5	0,2615	D9
0,0000	D5	0,1585	D6	0,0016	D7
0,0000	D6	0,0013	D7	-0,0063	D6
0,0000	D7	-0,0631	D8	-0,0462	D3
0,0000	D8	-0,0631	D10	-0,0468	D4
0,0000	D10	-0,0712	D4	-0,0473	D8
0,0000	D13	-0,0712	D3	-0,0473	D10

Results of information retrieval (Q2)

Term matching method		Latent semantic indexing		Concept indexing	
score	document	score	document	score	document
1,4142	D15	0,6737	D6	0,7105	D1
1,1547	D3	0,6472	D7	0,6204	D11
0,8944	D7	0,6100	D8	0,5936	D12
0,7071	D12	0,6100	D10	0,5517	D2
0,5774	D4	0,5924	D3	0,5488	D14
0,5774	D8	0,5924	D4	0,5452	D6
0,5774	D10	0,5789	D10	0,5441	D9
0,5774	D14	0,5404	D2	0,5280	D7
0,5000	D9	0,5268	D11	0,5256	D15
0,5000	D11	0,5236	D9	0,4797	D8
0,4472	D1	0,4656	D12	0,4797	D10
0,0000	D2	0,3936	D15	0,4693	D3
0,0000	D5	0,3560	D14	0,4693	D4
0,0000	D6	0,3320	D13	0,4459	D5
0,0000	D13	0,2800	D5	0,4202	D13

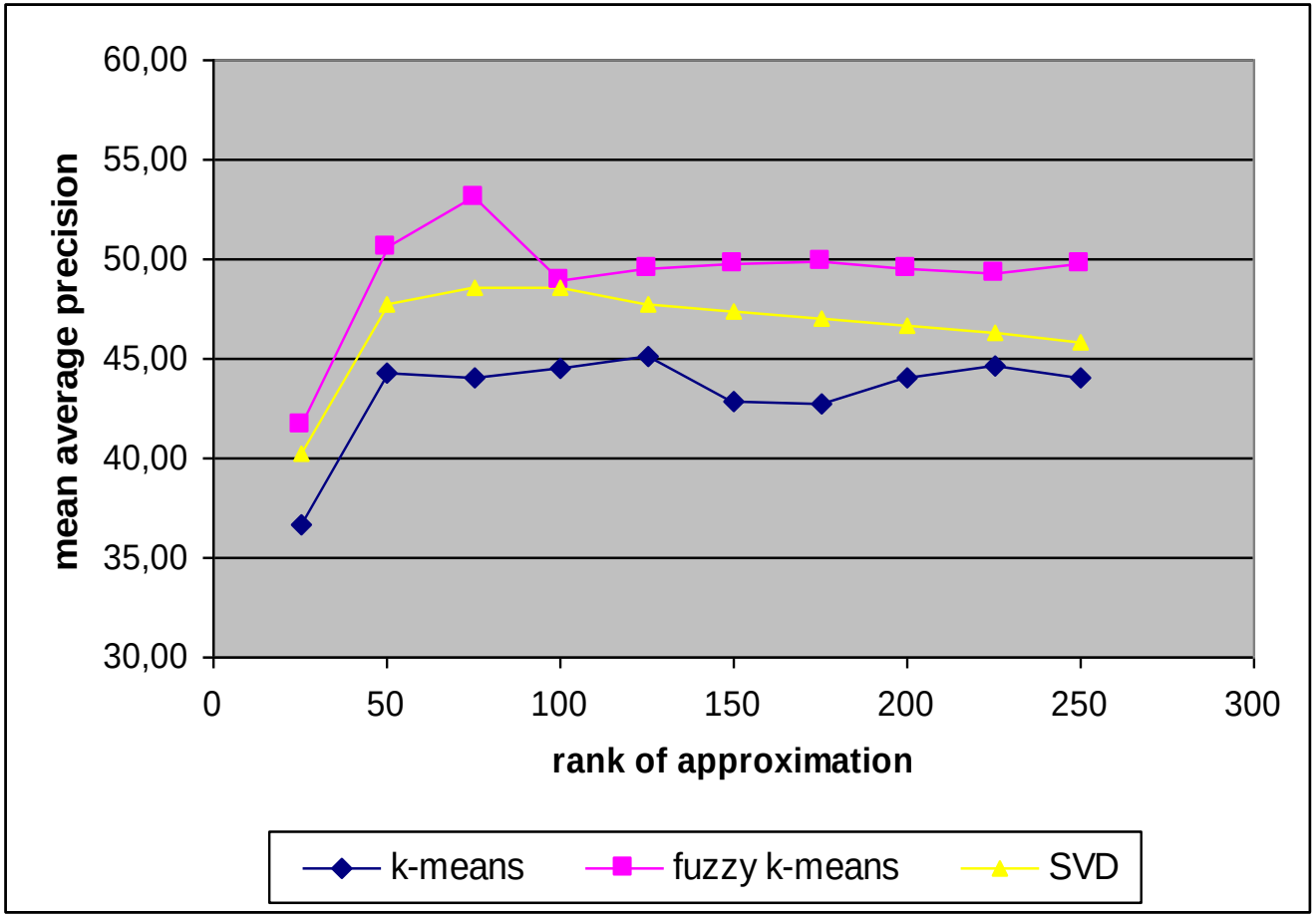
Example: Experimental collections of documents

- MEDLINE
  - 1033 documents
  - 30 queries
  - Relevant judgements
- CRANFIELD
  - 1400 documents
  - 225 queries
  - Relevant judgements

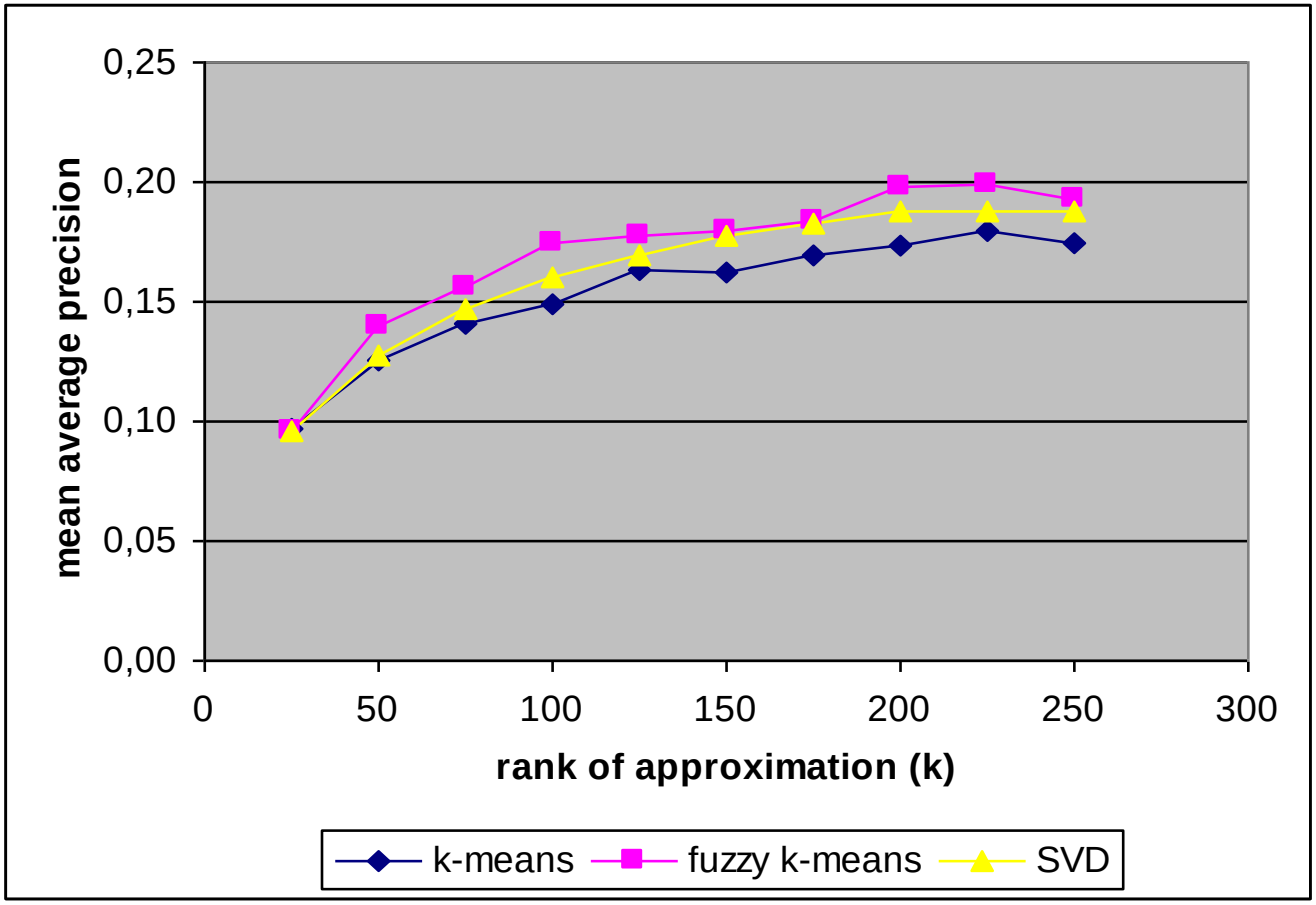
Experiment

- Comparison of mean average precision of information retrieval and precision-recall plots
- Mean average precision for term-matching method:
  - MEDLINE : 43,54
  - CRANFIELD : 20,89

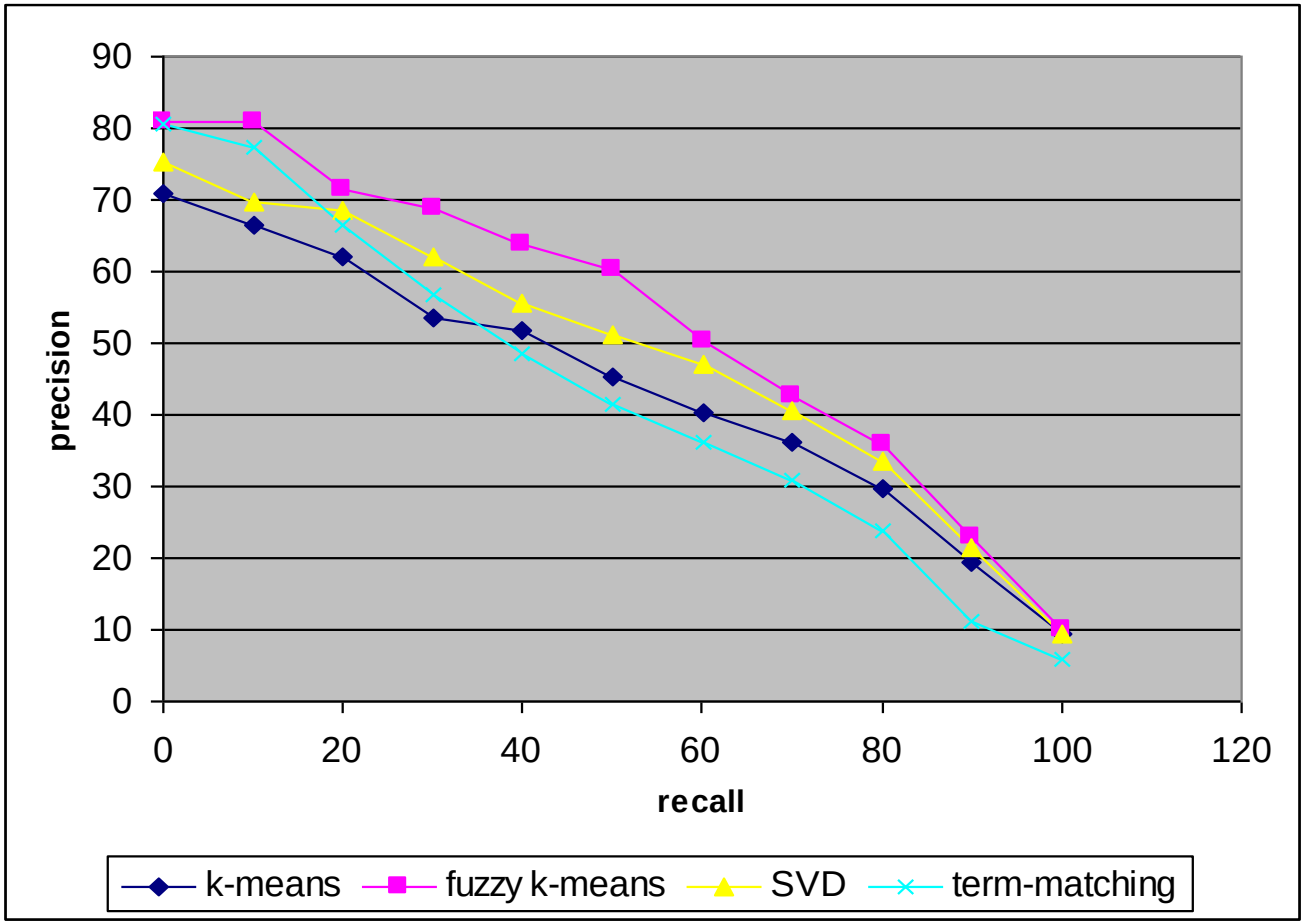
MEDLINE – mean average precision



CRANFIELD – mean average precision



MEDLINE – precision-recall plot



Teorem Eckarta and Younga

- Distance between matrix **A** and its approximation by matrix of rank $k < p = min\{m,n\}$ in Frobenious norm is minimized for trucated SVD decomposition of a given matrix
- Despite that theoretical fact, results of information retrieval are beter for concept decoposition by fuzzy k-means clustering

Thank you for your attention!