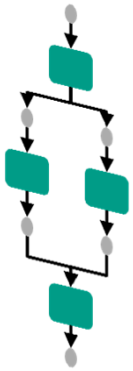
Karlsruhe Institute of Technology

Process Model Search using Latent Semantic Analysis

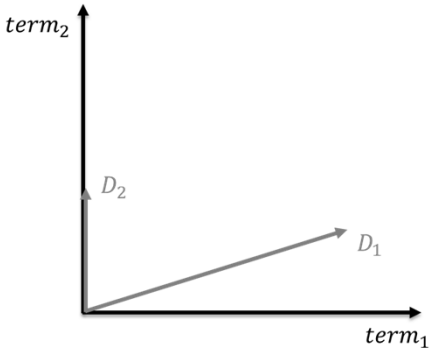
Andreas Schoknecht, Nicolai Fischer, and Andreas Oberweis

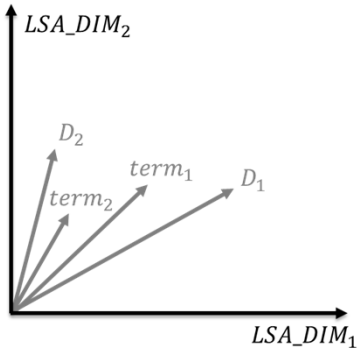
1st International Workshop on Process Querying

Karlsruhe Institute of Technology (KIT),
Institute of Applied Informatics and Formal Description Methods



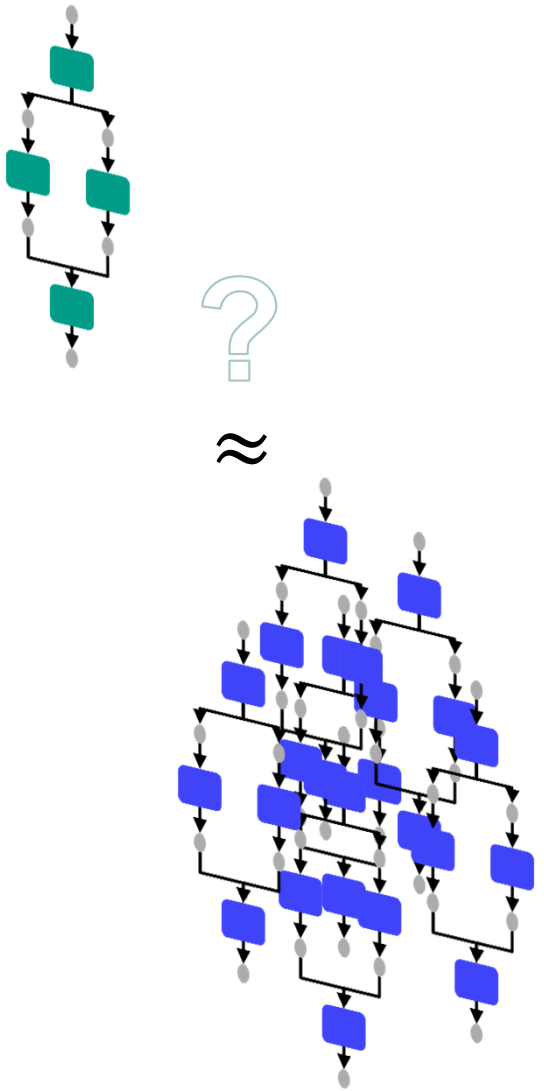
$$\begin{array}{c} \\ T_1 \\ T_2 \\ \vdots \\ T_t \end{array} \begin{array}{c} \begin{array}{cccc} D_1 & D_2 & \cdots & D_n \end{array} \\ \begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1n} \\ w_{21} & w_{22} & \cdots & w_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ w_{t1} & w_{t2} & \cdots & w_{tn} \end{pmatrix} \end{array}$$





Motivation

- Similarity-based search for process models
- Existing approaches mostly based on Process Model Matching
- Determining Matches challenging
  - Correctness
  - Effort
- Solution
  - Similarity calculation in vector space
  - Circumvent matching part



Latent Semantic Analysis

- LSA is a mathematical / statistical method for determining the meaning of words and documents

$$meaning_{passage} = \sum(m_{term1}, m_{term2}, ..., m_{termn})$$

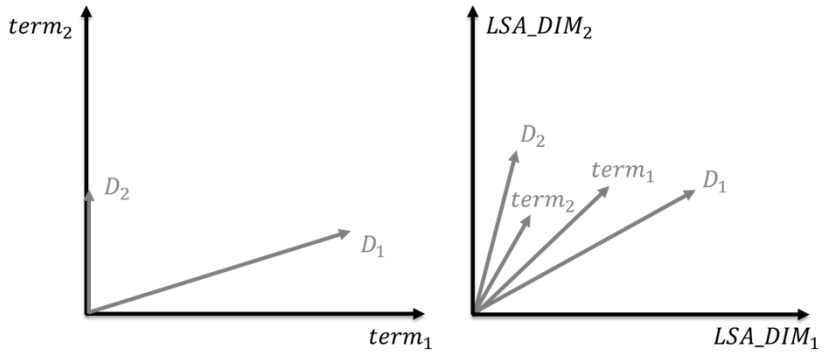
- Has been developed for improving document search in information retrieval
- Extends syntax-based approaches of information retrieval by incorporating the latent semantic structure of documents

LSA – Semantic vector space

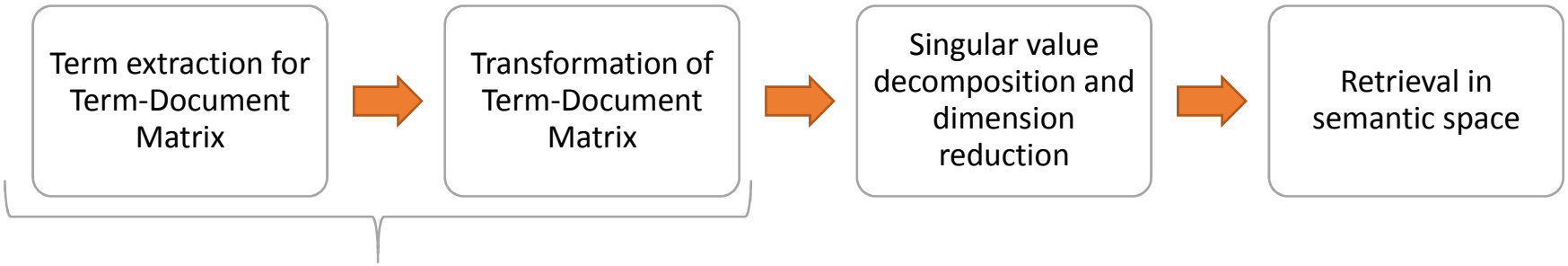
- Classical syntax-based approaches of information retrieval base on a Term-Document Matrix
  - Rows = Terms
  - Columns = Documents
  - Entry = Frequency

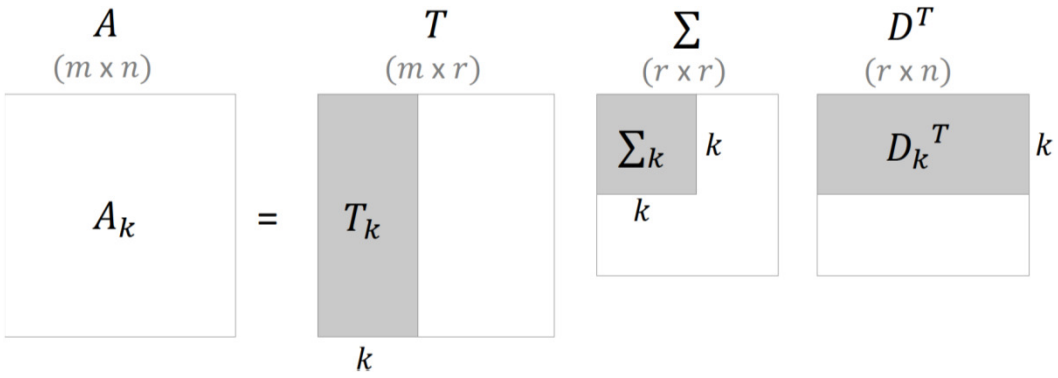
$$\begin{array}{c} \\ T_1 \\ T_2 \\ \vdots \\ T_t \end{array} \begin{array}{c} \begin{array}{cccc} D_1 & D_2 & \cdots & D_n \end{array} \\ \begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1n} \\ w_{21} & w_{22} & \cdots & w_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ w_{t1} & w_{t2} & \cdots & w_{tn} \end{pmatrix} \end{array}$$

- LSA additionally considers the shared occurrence of terms
  - Latent semantic dimensions
  - Transformation of vector space



LSA – Main steps





Singular value decomposition and dimension reduction

Document Vector Representation of Process Models

Malinova, M., et al.: Automatic Extraction of Process Categories from Process Model Collections. BPM Workshops, pp. 430 – 441 2014

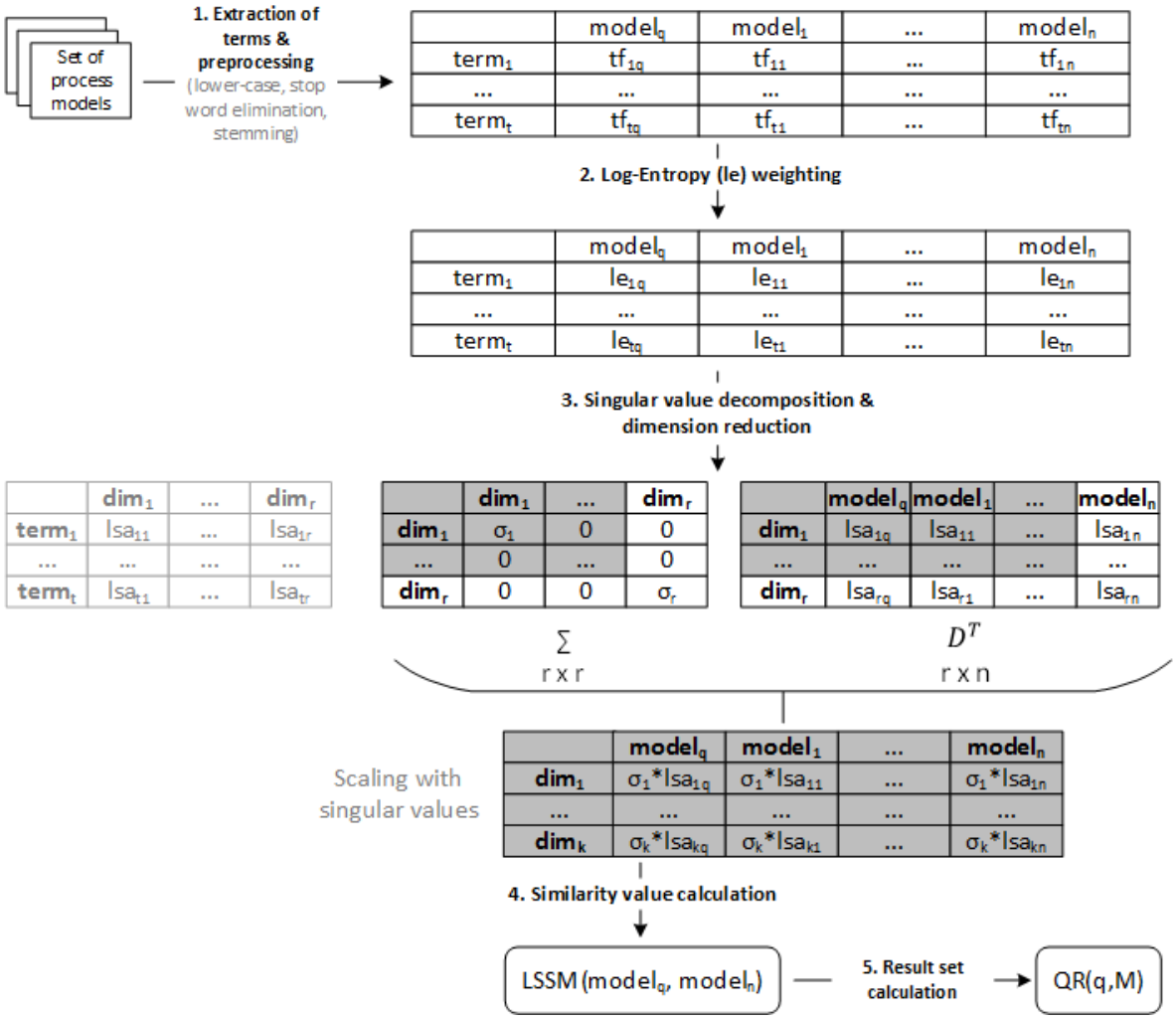
- Let …
  - M be a set of process models
  - W_{all} be a set of terms containing all distinct terms of M
  - $w(m)$ be a function, which returns the set of terms (bag-of-words) W_m of a process model $m \in M$ ($W_m \subseteq W_{all}$)

- The vector $d_m = (w_{1m}, w_{2m}, \dots, w_{tm})$ then represents the document vector of the process model m
- Each index t represents a term of the set of all terms contained in the process model collection $W_{all} = \bigcup w(m)$ for all $m \in M$

- The entries w_{tm} reflect a weight of the term frequency, which describe how often a certain term appears within a model

LS3: LSA-based Similarity Search (1)

Calculation of similar models with respect to query model q



LS3: LSA-based Similarity Search (2)

Step 1: Extraction of terms for the term-document matrix

- Extraction of distinct terms of place and transition labels
- Transformation into lower case letters
- Removal of stop words
- Stemming with Porter stemmer

- Term-document matrix contains absolute term frequencies

Step 2: Transformation of the term-document matrix

- Application of log-entropy weighting
- Differences in absolute term frequencies shall be reduced
- Frequently appearing terms shall be less relevant compared to infrequent terms

- Term-document matrix contains weighted term frequencies

LS3: LSA-based Similarity Search (3)

Step 3 / 4: SVD, dimension reduction, and similarity value calculation

- Calculation of singular value decomposition
- Only matrices Σ and D^T are relevant
- Scaling of document vectors with singular values
- Calculation of cosine similarity
- $\mathrm{LSSM}(q, m) = \frac{\cos_{\mathrm{sim}}(q,m)+1}{2}$

Step 5: Retrieval of query results

- Results are determined through a threshold value
- $\mathrm{QR}(q, M) = \{m \mid m \in M \wedge \mathrm{LSSM}(q, m) \geq \theta\}$

Evaluation Setup

- Dutch governance models (80 models)
  - 8 processes of 10 municipalities
  - Linguistically harmonized labels

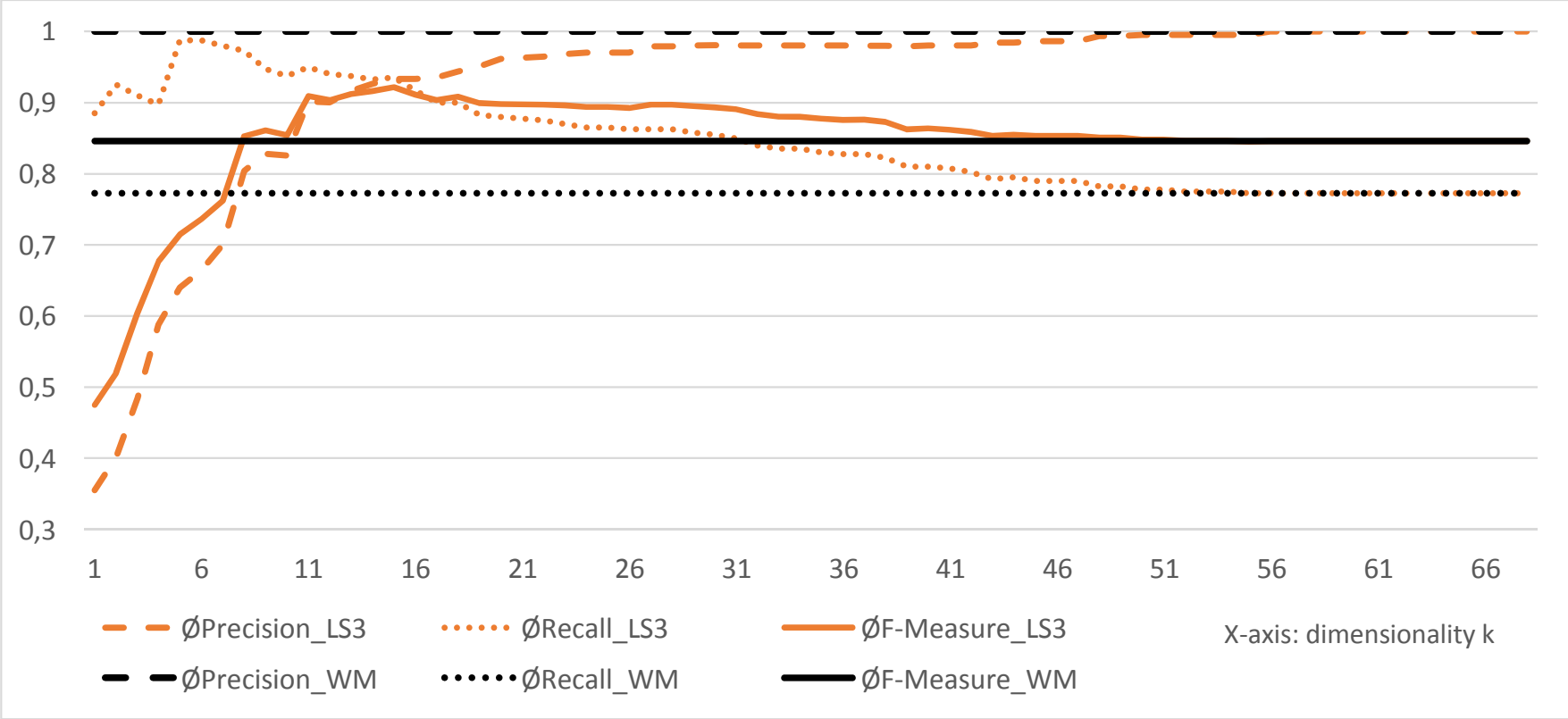
Vogelaar, J. et al.: Comparing Business Processes to Determine the Feasibility of Configurable Models: A Case Study. BPM Workshops. pp. 50–61 2011

- Calculation of Precision, Recall and F-Measure
  - Each model used as query model (80 queries per dimensionality)
  - Returned models are relevant if they represent the same process as the query model
  - Calculated for each possible dimensionality (Step 3)

- $\theta \geq 0.75$

- Compared against classical Word Matching from Information Retrieval

Evaluation Results



Comparison of LS3 and classical Word Matching from Information Retrieval

Discussion and Limitations

- Strengths
  - No matching of process model elements necessary
  - No external corpora or ontologies needed
  - Fast run time

- Limitations
  - Determination of optimal dimensionality
  - Interpretation of latent dimensions
  - Sufficiently many terms in labels

Conclusion and Outlook

- Similarity calculation of process models based on Latent Semantic Analysis shows **promising results**
- **Further empirical studies** needed
  - Larger process model collections
  - Model collections with non-harmonized labels
  - Comparison against other process model similarity measures
- **Handling of changes** in the model collection not yet incorporated

Karlsruhe Institute of Technology

Thank you for your attention!

QUESTIONS?