Between Data Lakes and Research Data Management –
Data Engineering Tasks for the Next Decade

Prof. Dr.-Ing. habil. Meike Klettke
Lehrstuhl für Data Engineering
FAKULTÄT FÜR INFORMATIK UND DATA SCIENCE
My main research interests

**Abstraction** over data

**Evolution** of data structures and processes

**Automation** of data engineering processes

... and abstraction you can also see in my paintings ...
Motivation: The Academic Point of View onto Research Data

Structured datasets
- Complete, consistent, regular and relational-like

Research Data
- Heterogeneous data formats, systems, and schemas
- Sometimes noisy, error-prone, and incomplete

NFDI: Definition of Standards (to enable Data Integration) and Processes (to guarantee Data Quality)
What are the next Data Engineering Tasks? – Structure of the Talk

Presence and Future (ongoing Research Tasks):

• First Generation: Data Preprocessing
• Second Generation: Data Engineering Pipelines

Future:

• Third Generation: Adaptive Data Engineering Pipelines
• Fourth Generation: Automatic Data Curation based on Recommender Technologies
First Generation: Data Preprocessing Algorithms

Data Engineering subtasks:
1. Data Selection (Sampling)
2. Data Understanding
3. Cleaning and Data Correction
4. Data Transformation

Data Engineering is time-consuming (80% of the overall effort), error-prone and expensive
- Choice, parametrization and application of algorithms manual task
- Skills in computer science are needed for this data preprocessing
First Generation: Data Preprocessing Algorithms

Overview on data engineering subtasks

1. **Data Understanding**
   - Schema Extraction
   - Column Type Inference
   - Inference of Integrity Constraints/Pattern
   - Data Exploration

2. **Cleaning and Data Correction**
   - Concept Shift Detection
   - Bias Detection
   - Outlier Detection and Correction
   - Duplicate Elimination
   - Missing Value Imputation

3. **Data Transformation**
   - Matching and Mapping
   - Data Integration
   - Datatype Transformation
   - Transformation between different Data Models
Our own work in this field

**Data Understanding** – Exploring characteristics of JSON data

M. Möller, N. Berton, M. Klettke, S. Scherzinger, U. Störl: *jHound: Large-Scale Profiling of Open JSON Data*. BTW 2019

**Schema Evolution and Data Migration**

**Reverse Engineering**: Schema Extraction*


---

* together with Stefanie Scherzinger and Uta Störl
Many other publications in this field, here are the papers of the previous session (Session 4, BTW 2023)
Second Generation: Data Engineering Pipelines

Data Engineering Tools:

- For each data engineering task **many implementations** are available
  - for different data models/
  - for different data characteristics
  - applying different methods
- Toolsets are providing a **variety of algorithms**

**Manual task:** select and compose the algorithms in **pipelines**
Second Generation: Data Engineering Pipelines

• **Pipelining idea in:**
  – ETL (processes), **Machine learning** (pipelines), **Data science** (pipelines)

• **Some of these available toolsets are:**
  – ETL tools for Data warehouses and BI tools (e.g. Talend, Tableau Prep, Qlik, ...)
  – Python and data science libraries (NumPy, pandas, SciPy, scikit-learn, feature-engineering)
  – Data preparation parts in data mining tools (Weka, RapidMiner)
  – Data wrangling/ Data Lake processing (Snowflake, IBM InfoSphere DataStage, luigi)

• **Ongoing Research:**

  – Sebastian Baunsgaard, Matthias Boehm, Ankit Chaudhary, Behrouz Derakhshan, Stefan Geißelsöder, Philipp M. Grulich, Michael Hildebrand, Kevin Innerebner, Volker Markl, Claus Neubauer, Sarah Osterburg, Olga Ovcharenko, Sergey Redyuk, Tobias Rieger, Alireza Rezaei Mahdiraji, Sebastian Benjamin Wrede, Steffen Zeuch: *ExDrA: Exploratory Data Science on Federated Raw Data*. SIGMOD Conference 2021
Our own work: Combine Schema Version Extraction and Data Migration

Input: Set of JSON documents (in different structural versions)
Pipeline: Combining Inference of Schema Versions and Evolution Operations and Data Migration
Result: all datasets in the latest structural version
Third Generation: Intelligent Data Engineering Pipelines

I. Automatic orchestration
   • choice of algorithms
   • composition of data engineering algorithms to a workflow

II. Monitoring of the data
   • Detection of data changes, changes of distributions, and data bias
   • Monitoring of evolving workflows

Other terms for similar idea:
   • data democratization, AutoML
Third Generation: Automatic workflow orchestration

Building blocks needed for this task:

1. Formal specification of the requirements
2. Formal metrics (e.g. schema, data-types, pattern, constraints, data quality measures) of the datasets
3. Formal characteristics for each data engineering algorithm (e.g. with pre- and postconditions)
4. Matches between requirements and algorithm characteristics
5. Opportunity to evaluate workflows

Ongoing work:


- Valerie Restat, Meike Klettke, Uta Störl: Towards a Holistic Data Preparation Tool. EDBT/ICDT Workshops 2022

- Valerie Restat, Meike Klettke, Uta Störl: "FAIR is not enough – A Metrics Framework to ensure Data Quality through Data Preparation. Workshop Data Engineering for Data Science (DE4DS)@BTW, 2023
Third Generation: Monitoring of the Data in Data Engineering Pipelines

- Monitoring of **data changes, changes of distributions, data bias** in Data Engineering workflows

- Definition of additional metadata for the data (barcode, data passports) and monitoring of these
  - Stefan Grafberger, P Groth, J Stoyanovich, S Schelter: *Data distribution debugging in machine learning pipelines*, VLDB Journal 31 (5), 2022
  - Erik Kleinsteuber, Samira Babalou, Birgitta König-Ries: *A Provenance Management Framework for Knowledge Graph Generation in a Web Portal*, DE4DS@BTW 2023
Third Generation: Monitoring of the Data in Evolving Data Engineering Pipelines

Monitoring of evolving data engineering workflows

- Reasons for evolution:
  - Replacement of algorithms
  - Correction of errors
  - Adding new algorithms
  - ...

- Monitoring changes
  - Applying the comparison of data between different workflows

Sihem Amer-Yahia, *Commodifying Data Exploration*, Keynote BTW 2023: *reuse vs. re-train*
Our own (future) plans in Data Engineering Pipelines

Focussed on
• evolving data
• evolving operators
• evolving pipelines

Development of tools to
• monitor data engineering workflows (datasets, data characteristics, distributions)
• monitor and evolution of evolving workflows
Fourth Generation: Automatic Data Curation

Developing tools for supporting data curation

- Let’s imagine: NFDI has defined the standards for research data for the different scientific fields
- Tools which guide design tasks would be nice

How can it be done?

- Data curations needs an understanding of the semantics of data
- Algorithms cannot understand attribute names, abbreviations, etc. workaround is necessary

Usage of

- recommender systems combined with
- similarity matching and mapping

Aim:

- during database design
  - suggestion of further attributes,
  - data types,
  - constraints (like an autocomplete)
# Recommender Systems for Database Design

**Input:** Set of Database schemas

**Method:**

- **Step 1 (optional) Preprocessing:** combining similar attributes
- **Step 2:** Creation of a user/item matrix, here database/attribute matrix
- **Step 3:** Calculation of association rules
- **Step 4 (optional) Postprocessing:** combining similar attributes
- **Step 5:** calculation of support and confidence values for the combined association rules
- **Step 6:** usage of the association rules for making design suggestions

![Database Schemas Matrix](image)

<table>
<thead>
<tr>
<th>A</th>
<th>B1</th>
<th>C</th>
<th>F1</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>B2</th>
<th>C</th>
<th>E</th>
<th>F2</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>C</th>
<th>F3</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**A** | **B1** | **B2** | **C** | **D** | **E** | **F1** | **F2** | **F3**
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>D1</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D2</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D3</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>D4</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>D5</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
Let’s assume:
min-support: 3, min-confidence: 60%

Association Rules:
- \( C \rightarrow E \), \( s:3(60\%) \), \( c: 60\% \)
- \( E \rightarrow C \), \( s:3(60\%) \), \( c:100\% \)

\((s: \text{support}, c: \text{confidence})\)
Recommender Systems for Database Design:
Deriving Association Rules

Let’s assume:
min-support: 3, min-confidence: 60%

- C → F1, s:2(40%), c: 40%
- C → F2, s:2(40%), c: 40%
- C → F3, s:1(20%), c: 20%
- F1 → C, s:2(40%), c: 100%
- F2 → C, s:2(40%), c: 100%
- F3 → C, s:1(20%), c: 100%
- E → F1, s:1(20%), c: 33%
- E → F2, s:2(40%), c: 66%
- F1 → E, s:1(20%), c: 50%
- F2 → E, s:2(40%), c: 100%

(s: support, c: confidence)
Recommender Systems for Database Design: Deriving Association Rules

Let’s assume:
min-support: 3, min-confidence: 60%

- C → F1
- C → F2
- C → F3
- F1 → C
- F2 → C
- F3 → C
- E → F1
- E → F2
- F1 → E
- F2 → E

<table>
<thead>
<tr>
<th>A</th>
<th>B1</th>
<th>B2</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F1</th>
<th>F2</th>
<th>F3</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D2</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D3</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D4</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D5</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Similar attributes

Similar attributes

C → F (F1,F2,F3) s:5(100%), c: 100%
F (F1,F2,F3) → C s:5(100%), c: 100%
E → F (F1,F2,F3) s:3(60%), c: 100%
F (F1,F2,F3) → E s:3(60%), c: 60%
Usage of the Association Rules for Database Designs

**Diagram Description:**
- The diagram illustrates the usage of association rules in database design.
- **Database Design** (at the bottom) shows a process involving rules like $C \rightarrow E$ and $C \rightarrow F(F_1, F_2, F_3)$.
- **Database Standards** (at the top) include a table labeled "Database standards (e.g. NFDI)."
- The table contains elements like $A$, $B_1$, $B_2$, $C$, $D$, $E$, $F_1$, $F_2$, $F_3$.
- The similarity function is represented by arrows connecting the database design and standards, showing how association rules are derived.

**Equations:**
- $C \rightarrow E$
- $C \rightarrow F(F_1, F_2, F_3)$
Overview of the Method and Differences to the Standard Method

Method:
• Using **available database standards** as well as **available database designs** for collaborative filtering
• Usage of **different data models** is possible
• Generating a user/item matrix, here database/attribute matrix
• Using a **similarity function** for finding similar items (attributes)
• Calculating association rules with **support and confidence**, (considering min-support and min-confidence)

Another modification:
• Different weights for different rows in the user/item matrix (also known as database/attribute matrix)
• **Standards are more reliable then other database schemas**

Characteristics:
• Like each collaborative filtering: learning system
Conclusion and Future Work

Data Engineering Research:

• Generation 1: Data Engineering algorithms for each subtask
• Generation 2: Pipelines for defining whole data engineering processes
• Generation 3: Advisor components in these data engineering pipelines
• Generation 4: Intelligent data curation

In each of these subparts, many open research tasks, mainly

• Monitoring data engineering workflows, esp. evolving data engineering workflows
• Database design tools based on recommender systems combined with similarity matching and mapping

Estimation which bases on a very shallow keyword search in dblp
References (1/3)

- Sebastian Baunsgaard, Matthias Boehm, Ankit Chaudhary, Behrouz Derakhshian, Stefan Geißelsöder, Philipp M. Grulich, Michael Hildebrand, Kevin Innerebner, Volker Markl, Claus Neubauer, Sarah Osterburg, Olga Ovcharenko, Sergey Redyuk, Tobias Rieger, Alireza Rezaei Mahdiraji, Sebastian Benjamin Wrede, Steffen Zeuch: *ExDRA: Exploratory Data Science on Federated Raw Data*. SIGMOD Conference 2021
References (2/3)

References (3/3)

- Valerie Restat, Meike Klettke, Uta Störl: "FAIR" is not enough – A Metrics Framework to ensure Data Quality through Data Preparation. Workshop Data Engineering for Data Science (DE4DS)@BTW, 2023