Evaluating Matching Techniques with Valentine*

CHRISTOS KOUTRAS

*Work presented in IEEE ICDE 2021
From Data Integration to Schema Matching

• Organizations gather heterogeneous data into *data lakes*
  • Data scientists spend most of their time on capturing relevance
  • Data Integration Problem: Relevant data sources are unlinked

• Need for *Schema Matching*
  • (Semi-)Automated methods with the goal of finding links among datasets
  • E.g. Related columns among Tabular Data
From Schema Matching to Dataset Discovery

• **Dataset Discovery** is a critical task for organizing a data lake
  • Navigate numerous data sources to find relationships for a given dataset
  • Schema Matching is a **core** component of any modern dataset discovery pipeline
Schema Matching in research

- Abundance of matching methods
- No comparison in 20 years
- No evaluation datasets
- Outdated metrics
A missing link with Dataset Discovery literature

• Result: none of the dataset discovery methods (~last 10 years) employ them!

• Dataset Discovery methods typically implement their own matching methods
Valentine to the rescue

Current limitations

- No comparative study
- No specific relatedness scenarios
- No evaluation datasets
- No open-sourced methods
- Tough/impossible deployment for Dataset Discovery

Our contributions

- Most comprehensive effectiveness/efficiency study to date
- Relatedness scenarios accustomed to Dataset Discovery
- Dataset fabrication
- 6 SotA methods + a baseline
- Easily deployed and extensible
Matching in Dataset Discovery

• Six categories of matchers are used:
  - Attribute Overlaps
  - Value Overlaps
  - Semantic Overlaps
  - Data Types
  - Distributions
  - Embeddings

• Valentine brings the best of schema matching
  • Covers all matcher categories
  • Sophisticated methods that employ several intuitions and techniques
A new way of evaluating schema matching

Table 1

<table>
<thead>
<tr>
<th>Client</th>
<th>Street</th>
<th>PO</th>
<th>Country</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Ground Truth</td>
<td>▲</td>
<td>▲</td>
<td>▲</td>
</tr>
</tbody>
</table>

Table 2

<table>
<thead>
<tr>
<th>C_ID</th>
<th>Addr</th>
<th>P_Code</th>
<th>Cntr</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Conventional Schema Matching Evaluation

Match Results

- Client <-> C_ID
- Country <-> Cntr

Evaluation

- Precision: 0.5
- Recall: 1/3

Valentine's Novel Evaluation

Ranked Match Results

- Client - C_ID: 0.85
- Country - Cntr: 0.67
- PO - P_Code: 0.35

Evaluation

- Recall @ ground truth: 1
Dataset Relatedness Scenarios

- Evaluate on dataset pairs that respect specific relatedness semantics

Joinable or Semantically Joinable

Unionable or View-Unionable
Dedicated Fabrication Process

- Fabricate dataset pairs that follow the relatedness scenarios
- Based on a source table create pairs by employing
  - Horizontal/Vertical splits
  - Noise injection in Schemata/Instances
Valentine’s Schema Matching Methods

• Consolidates the best of schema matching efforts (last 20 years)
• Schema-based
  • Cupid - Similarity Flooding - COMA
• Instance-based
  • Distribution-based - COMA instance
  • Baseline based on approximate instance set overlaps
• Hybrid
  • SemProp - EmbDI
Findings – Schema Based Methods

- Noisy schemata critically affect effectiveness and consistency
- Schema information (e.g. data types) and contextual information not insightful
Findings – Instance Based Methods

- View unionable and semantically-joinable scenarios are considerably harder
- High skew in effectiveness for all methods/scenarios
Findings – Hybrid Methods

• Low effectiveness and high skew
• Embeddings – pretrained or local ones – are still not a trustworthy standalone tool for matching
Findings on ING Data

<table>
<thead>
<tr>
<th>Methods</th>
<th>ING #1</th>
<th>ING #2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cupid</td>
<td>0.71</td>
<td>0.5</td>
</tr>
<tr>
<td>Sim Flooding</td>
<td>0.36</td>
<td>0.44</td>
</tr>
<tr>
<td>COMA Schema</td>
<td>0.79</td>
<td>0.12</td>
</tr>
<tr>
<td>COMA Instance</td>
<td>0.79</td>
<td>0.14</td>
</tr>
<tr>
<td>Dist based</td>
<td>0.86</td>
<td>0.88</td>
</tr>
<tr>
<td>Baseline</td>
<td>0.79</td>
<td>0.62</td>
</tr>
<tr>
<td>EmbDI</td>
<td>0.71</td>
<td>0.23</td>
</tr>
</tbody>
</table>

- Distributions of values can bring helpful insights
- Schema-based methods have very low effectiveness
Efficiency Results

- Schema-based considerably faster
- Training embeddings can be very inefficient

<table>
<thead>
<tr>
<th>Methods</th>
<th>Avg Runtime (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cupid</td>
<td>9.64</td>
</tr>
<tr>
<td>Sim Flooding</td>
<td>7.09</td>
</tr>
<tr>
<td>COMA Schema</td>
<td>1.67</td>
</tr>
<tr>
<td>COMA Instance</td>
<td>318.07</td>
</tr>
<tr>
<td>Dist based</td>
<td>71.16</td>
</tr>
<tr>
<td>SemProp</td>
<td>735.25</td>
</tr>
<tr>
<td>Baseline</td>
<td>522.94</td>
</tr>
<tr>
<td>EmbDI</td>
<td>4817.87</td>
</tr>
</tbody>
</table>
Lessons Learned

• There is no matching method that is consistently the best
• Embeddings should only be used together with other techniques
• Parameterization can be a daunting task – availability of ground truth can help
• Baselines can perform well
• Humans should be incorporated
• Schema Matching doesn’t scale – expensive to deploy
Valentine in Action [Demo in VLDB 2021]

- Schema Matching systems come with an outdated GUI
- No deployment at scale – deployment in a data lake
- No easy-to-use and complete evaluation system available
- Offer Valentine’s utilities through a user-friendly GUI
  - Offer a scenario-driven dataset fabricator
  - Enable users to conduct extensive experiments
- Enable users to deploy Valentine for holistic matching in a data lake
Lessons Learned: Matching in a Data Lake

• Deploying matching in a data lake is a daunting task
  • Resource expensive
  • All-pairs comparison is inefficient / SOTA methods difficult to scale

• Automated matching techniques are not always reliable
  • They work under specific assumptions about the data
  • Such assumptions may not apply in big data repositories
  • **Human refinement is necessary**
Prospects in Large-scale Matching

• Incorporate human knowledge in development of methods
  • Instead of using humans in refinement, use them in the beginning
  • There always exists partial knowledge of the underlying data

• Build robust models
  • Model human knowledge in order to leverage modern DL methods
  • Can generalize well
Visit [https://delftdata.github.io/valentine/](https://delftdata.github.io/valentine/)!

✔ Links to our GitHub Repos
  - Code for deployment
  - Code for dataset fabrication
  - In detail experimental results

✔ All fabricated dataset pairs used in the paper

✔ Updates regarding Valentine