Raha: A Configuration-Free Error Detection System

Mohammad Mahdavi1, Ziawasch Abedjan1, Raul Castro Fernandez2, Samuel Madden2, Mourad Ouzzani3, Michael Stonebraker2, and Nan Tang2

1 TU Berlin
{mahdavilahijani, abedjan}@tu-berlin.de
2 MIT
{raulcf, madden, stonebraker}@csail.mit.edu
3 QCRI, HBKU
{mouzzani, ntang}@hbku.edu.qa

Motivation

- Error detection is the task of finding wrong values
- E.g., the red values in the table
- Traditional algorithms need input rules or parameters
- E.g., a rule violation detector needs Kingdom → Lord
- ML-based approaches need a large training set
- E.g., 1% of the dataset [6]

Research Question

- Given a dataset and a set of error detection algorithms, how can we accurately detect data errors without involving the user heavily in
  - Algorithm selection
  - The user should not select the promising algorithms
  - Algorithm configuration
  - The user should not provide any rules or parameters
  - Training data creation
  - The user may only label a few data values

Raha detects data errors with fewer than 20 labeled tuples due to its expressive feature representation and clustering-based sampling.

The Workflow of Raha

1. Automatic Algorithm Configuration
2. Running Strategies on Dataset
3. Feature Vector Generation
4. Clustering Data Cells
5. Tuple Selection
6. Labeling Data Cells
7. Propagating User Labels Through Clusters
8. Training Classification Models
9. Predicting Labels of Rest of Data Cells

Experimental Overview

8 Datasets
- Hospital
- Flights
- Address
- Movies
- IT
- Tax

7 Baselines
- dBoost [1]
- NADEEF [2]
- KATARA [3]
- ActiveClean [4]
- Min-k [5]
- PBOS [5]
- MDED [6]

5 Evaluation Measures
- Precision
- Recall
- F1 Score
- Runtime
- Labeled Tuples

10+ Experiments
- Performance
- Features
- Sampling
- Strategy Filtering
- User Labeling Error
- Scalability
- Classification Model

Raha Versus Baselines: Performance

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Hospital</td>
<td>Hospital</td>
<td>Hospital</td>
<td>Hospital</td>
<td>Hospital</td>
<td>Hospital</td>
<td>Hospital</td>
</tr>
<tr>
<td>Flights</td>
<td>Flights</td>
<td>Flights</td>
<td>Flights</td>
<td>Flights</td>
<td>Flights</td>
<td>Flights</td>
</tr>
<tr>
<td>Address</td>
<td>Address</td>
<td>Address</td>
<td>Address</td>
<td>Address</td>
<td>Address</td>
<td>Address</td>
</tr>
</tbody>
</table>

References


Acknowledgement

This project has been supported by the German Research Foundation (DFG) under grant agreement 387872445.