Febrl – A Freely Available Record Linkage System with a Graphical User Interface

Peter Christen

Department of Computer Science,
Faculty of Engineering and Information Technology,
ANU College of Engineering and Computer Science,
The Australian National University

Contact: peter.christen@anu.edu.au


Funded by the Australian National University, the NSW Department of Health,
and the Australian Research Council (ARC) under Linkage Project 0453463.
Outline

- What is record linkage?
- Record linkage techniques
- The record linkage process
- Overview of Febrl
- The Febrl graphical user interface (GUI)
- An example Febrl GUI screenshot
- Practical demonstration of Febrl
- Outlook and future work
What is record (or data) linkage?

- The process of linking and aggregating records from one or more data sources representing the same entity (such as a patient, customer, or business)

- Also called *data matching*, *data integration*, *data scrubbing*, *entity resolution*, *object identification*, *merge-purge*, etc.

- Challenging if no unique entity identifiers available

For example, which of these three records refer to the same person?

<table>
<thead>
<tr>
<th>Name</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dr Smith, Peter</td>
<td>42 Miller Street 2602 O’Connor</td>
</tr>
<tr>
<td>Pete Smith</td>
<td>42 Miller St, 2600 Canberra A.C.T.</td>
</tr>
<tr>
<td>P. Smithers</td>
<td>24 Mill Street; Canberra ACT 2600</td>
</tr>
</tbody>
</table>
Record linkage techniques

- Deterministic linkage
  - Exact linkage (if a unique entity identifier of high quality is available: has to be precise, robust, stable over time)
  - Examples: Medicare, ABN or Tax file number (?)
  - Rules based linkage (complex to build and maintain)

- Probabilistic linkage (*Fellegi and Sunter, 1969*)
  - Use available (personal) information for linkage (which can be missing, wrong, coded differently, and/or out-of-date)
  - Examples: names, addresses, dates of birth, etc.

- Modern approaches
  - Based on machine learning, data mining, artificial intelligence, and information retrieval techniques
The record linkage process

Database A

Cleaning and standardisation

Database B

Cleaning and standardisation

Weight vector classification

Matches

Non-matches

Possible matches

Evaluation

Blocking / Indexing

Field comparison

Clerical review

Peter Christen, January 2008 – p.5/9
Overview of Febrl

- Has been developed since 2002
  (as part of an ARC Linkage Project between the ANU and the NSW Department of Health)

- Is implemented in Python (a freely available object-oriented programming language)

- Its source code is available

- Includes many recently developed algorithms and techniques for indexing (blocking), field comparisons and record pair classification

- An ideal tool to learn about record linkage

- Is freely available at Sourceforge.net

  https://sourceforge.net/projects/febrl/
The Febrl graphical user interface

- A page (tab) based approach with one page per major step of the record linkage project
- Three different project types
  - Clean and standardise one data set
  - Deduplicate one data set
  - Link two data sets
- Clicks on ‘Execute’ will validate and confirm the settings on a GUI page
- Febrl Python code will be generated (and can be run outside the GUI)
- GUI structure similar to the Rattle open source data mining tool (http://rattle.togaware.com)
An example Febrl GUI screenshot

![Febrl GUI screenshot]

<table>
<thead>
<tr>
<th>rec_id</th>
<th>given_name</th>
<th>surname</th>
<th>street_number</th>
<th>address_1</th>
<th>address_2</th>
<th>suburb</th>
</tr>
</thead>
<tbody>
<tr>
<td>1491</td>
<td>elle</td>
<td>lee</td>
<td>2</td>
<td>elleron avenue</td>
<td>rosdale</td>
<td>epping</td>
</tr>
<tr>
<td>2551</td>
<td>zachary</td>
<td>kanhanoung</td>
<td>2</td>
<td>healy place</td>
<td>belmont cottage</td>
<td>northam</td>
</tr>
<tr>
<td>2661</td>
<td>cooper</td>
<td>gillick</td>
<td>23</td>
<td>templestowe avenue</td>
<td>tully</td>
<td></td>
</tr>
<tr>
<td>3501</td>
<td>joshua</td>
<td>doddson</td>
<td>16</td>
<td>templeton street</td>
<td>alexander heights</td>
<td></td>
</tr>
<tr>
<td>3451</td>
<td>tiaza</td>
<td>karandakis</td>
<td>1</td>
<td>cockle street</td>
<td>loch 6357</td>
<td>burramurra</td>
</tr>
<tr>
<td>3201</td>
<td>mitchell</td>
<td>lawrence</td>
<td>7</td>
<td>staff place</td>
<td>lindfield</td>
<td></td>
</tr>
<tr>
<td>620</td>
<td>lachlan</td>
<td>monteone</td>
<td>6</td>
<td>kingsbury street</td>
<td>torrensville</td>
<td></td>
</tr>
<tr>
<td>1420</td>
<td>lachlan-john</td>
<td>corby</td>
<td>17</td>
<td>astelia place</td>
<td>millamura</td>
<td>vincentia</td>
</tr>
<tr>
<td>2581</td>
<td>jasmine</td>
<td>mcherry</td>
<td>8</td>
<td>tristania street</td>
<td>tarragindi</td>
<td></td>
</tr>
<tr>
<td>3180</td>
<td>teegan</td>
<td>notley</td>
<td>4</td>
<td>clerk close</td>
<td>charters towers</td>
<td></td>
</tr>
<tr>
<td>1761</td>
<td>alissa</td>
<td>drumgoon</td>
<td>159</td>
<td>pridham street</td>
<td>phillip bay</td>
<td></td>
</tr>
<tr>
<td>330</td>
<td>leah</td>
<td>tarrant</td>
<td>33</td>
<td>gellbrand street</td>
<td>laura downs</td>
<td>griffith</td>
</tr>
</tbody>
</table>

Generated Febrl Python code for data set initialisation (see Log page for generated code).
Outlook and future work

*Febrl* is a tool suitable for both practitioners and new record linkage users

- Contains many different linkage techniques
- Allows small to medium sized experimental standardisations, deduplications and linkages
- Can be used alongside commercial linkage systems for comparative linkage studies

Future work on *Febrl*

- Include *Febrl* geocoding module into the GUI
- Include privacy-preserving record linkage techniques
- Add new indexing, comparisons and classification techniques as they become available