Febrl – A parallel open source data linkage system

Peter Christen, Tim Churches and Markus Hegland

Data Mining Group, Australian National University

Centre for Epidemiology and Research, New South Wales Department of Health

Contact: **peter.christen@anu.edu.au**

Project web page: **http://datamining.anu.edu.au/linkage.html**

Funded by the ANU, the NSW Department of Health and the Australian Partnership for Advanced Computing (APAC)

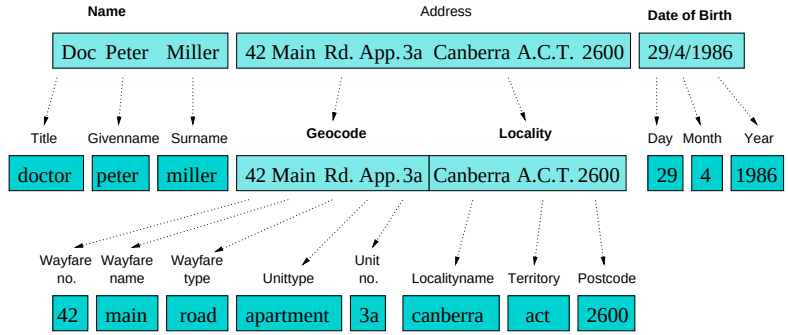
Outline

- Data cleaning and standardisation
- Data linkage
- *Febrl* overview
- Probabilistic data cleaning and standardisation
- Blocking / indexing
- Record pair classification
- Parallelisation in *Febrl*
- Data set generation
- Outlook

Data cleaning and standardisation

- Real world data is often *dirty*
  - Missing values
  - Typographical and other errors
  - Different coding schemes / formats
  - Out-of-date data
- Names and addresses are especially prone to data entry errors
- Cleaned and standardised data is needed for
  - Loading into databases and data warehouses
  - Data mining and other data analysis studies
  - Data linkage and data integration

Data cleaning and standardisation (II)



- Remove unwanted characters and words
- Expand abbreviations and correct misspellings
- Segment data into well defined *output fields*

Data linkage and data integration

- The task of linking together information from one or more data sources representing the same entity
- If no *unique identifier* is available, *probabilistic linkage techniques* have to be applied
- Applications of data linkage
  - Remove duplicates in a data set (internal linkage)
  - Merge new records into a larger master data set
  - Create customer or patient oriented statistics
  - Compile data for longitudinal studies

Data cleaning and standardisation are important first steps for successful data linkage

Data linkage techniques

- Deterministic or exact linkage
  - A *unique identifier* is needed, which is of high quality (precise, robust, stable over time, highly available)
  - For example *Medicare* number (?)
- Probabilistic linkage (*Fellegi & Sunter*, 1969)
  - Apply linkage using available (personal) information
  - Examples: *name*, *address*, *date of birth*
- Other techniques
  (rule-based, fuzzy approach, information retrieval)

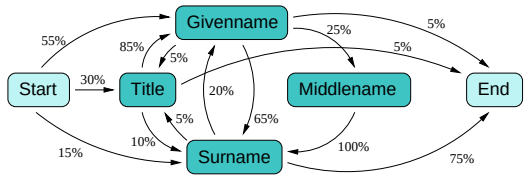
Febrl – Freely extensible biomedical record linkage

- An experimental platform for new and improved linkage algorithms
- Modules for data cleaning and standardisation, data linkage, deduplication, and geocoding
- Open source **https://sourceforge.net/projects/febrl/**
- Implemented in *Python* **http://www.python.org**
  - Easy and rapid prototype software development
  - Object-oriented and cross-platform *(Unix, Win, Mac)*
  - Can handle large data sets stable and efficiently
  - Many external modules, easy to extend

Probabilistic data cleaning and standardisation

- Three step approach
  1. Cleaning
     – Based on look-up tables and correction lists
     – Remove unwanted characters and words
     – Correct various misspellings and abbreviations
  2. Tagging
     – Split input into a list of words, numbers and separators
     – Assign one or more tags to each element of this list (using look-up tables and some hard-coded rules)
  3. Segmenting
     – Use either rules or a *hidden Markov model (HMM)* to assign list elements to *output fields*

Probabilistic data cleaning and standardisation – Example



- Uncleaned input string: *'Doc. peter Paul MILLER'*
  Cleaned into string: *'dr peter paul miller'*
- Word and tag lists:
  `['dr', 'peter', 'paul', 'miller']`
  `['TI', 'GM/SN', 'GM', 'SN' ]`
- Two example paths through HMM
  1: `Start -> Title (TI) -> Givenname (GM) -> Middlename (GM) -> Surname (SN) -> End`

  2: `Start -> Title (TI) -> Surname (SN) -> Givenname (GM) -> Surname (SN) -> End`

Blocking / indexing

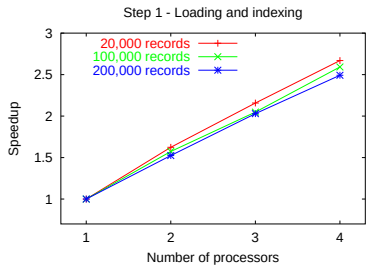
- Number of possible links equals the product of the sizes of the two data sets to be linked
- Performance bottleneck in a data linkage system is usually the (expensive) evaluation of similarity measures between record pairs
- Blocking / indexing techniques are used to reduce the large amount of record comparisons
- *Febrl* contains (currently) three indexing methods
  - Standard blocking
  - Sorted neighbourhood approach
  - Fuzzy blocking using n-grams (e.g. bigrams)

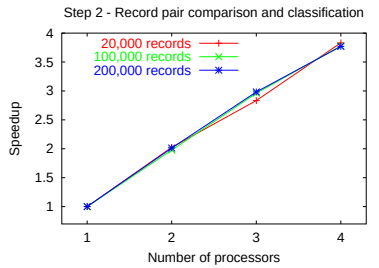
Record pair classification

- For each record pair compared a vector containing *matching weights* is calculated
  Example:
  Record A: `['dr', 'peter', 'paul', 'miller']`
  Record B: `['mr', 'pete', '', 'miller']`
  Matching weights: `[0.2, 0.8, 0.0, 2.4]`
- Matching weights are used to classify record pairs as *links*, *non-links*, or *possible links*
- *Fellegi & Sunter* classifier simply sums all the weights, then uses two thresholds to classify
- Improved classifiers are possible
  (for example using machine learning techniques)

Parallelisation

- Implemented transparently to the user
- Currently using *MPI* via Python module *PyPar*
- Use of supercomputing centres is problematic (privacy) → Alternative: *In-house office clusters*
- Some initial performance results (on *Sun SMP*)





Data set generation

- Difficult to acquire data for testing and evaluation
  (as data linkage deals with names and addresses)
- Also, linkage status is often not known
  (hard to evaluate and test new algorithms)
- *Febrl* contains a data set generator
  - Uses frequency table for given- and surnames, street names and types, suburbs, postcodes, etc.
  - *Duplicate records* are created via random introduction of modifications (like insert/delete/transpose characters, swap field values, delete values, etc.)

Data set generation – Example

- Data set with 4 original and 6 duplicate records

REC_ID	ADDRESS1	ADDRESS2	SUBURB
rec-0-org,	wylly place,	inverpine ret vill,	taree
rec-0-dup-0,	wyllyplace,	inverpine ret vill,	taree
rec-0-dup-1,	inverpine ret vill,	wylly place,	taree
rec-0-dup-2,	wylly place,	inverpine ret vill,	tared
rec-0-dup-3,	wylly parade,	inverpine ret vill,	taree
rec-1-org,	stuart street,	hartford,	menton
rec-2-org,	griffiths street,	myross,	kilda
rec-2-dup-0,	griffith sstreet,	myross,	kilda
rec-2-dup-1,	griffith street,	mycross,	kilda
rec-3-org,	ellenborough place,	kalkite homestead,	sydney

- Each record is given a unique identifier, which allows the evaluation of accuracy and error rates for data linkage

Outlook

- Several research areas
  - Improving probabilistic data standardisation
  - New and improved blocking / indexing methods
  - Apply machine learning techniques for record pair classification
  - Improve performances (scalability and parallelism)
- Project web page
  http://datamining.anu.edu.au/linkage.html

Febrl *is an ideal experimental platform to develop, implement and evaluate new data standardisation and data linkage algorithms and techniques*