A Probabilistic Geocoding System based on a National Address File

Peter Christen[1], Tim Churches[2] and Alan Willmore[2]

[1] Data Mining Group, Australian National University

[2] 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**

THE AUSTRALIAN
NATIONAL UNIVERSITY

Outline

- Geocoding

- Geocoded National Address File (G-NAF)

- *Febrl* geocoding system

- Address cleaning and standardisation

- Processing G-NAF

- Geocode matching engine

- First results and geocoding examples

- Future work

THE AUSTRALIAN
NATIONAL UNIVERSITY

Geocoding

- The process of assigning geographical coordinates (longitude and latitude) to addresses

- It is estimated that 80% to 90% of governmental and business data contain address information
  US Federal Geographic Data Committee

- Useful in many application areas

  - GIS, spatial data mining

  - Health, epidemiology

  - Business, census, taxation

- Various commercial systems available (e.g. MapInfo, www.geocode.com)

Geocoding techniques
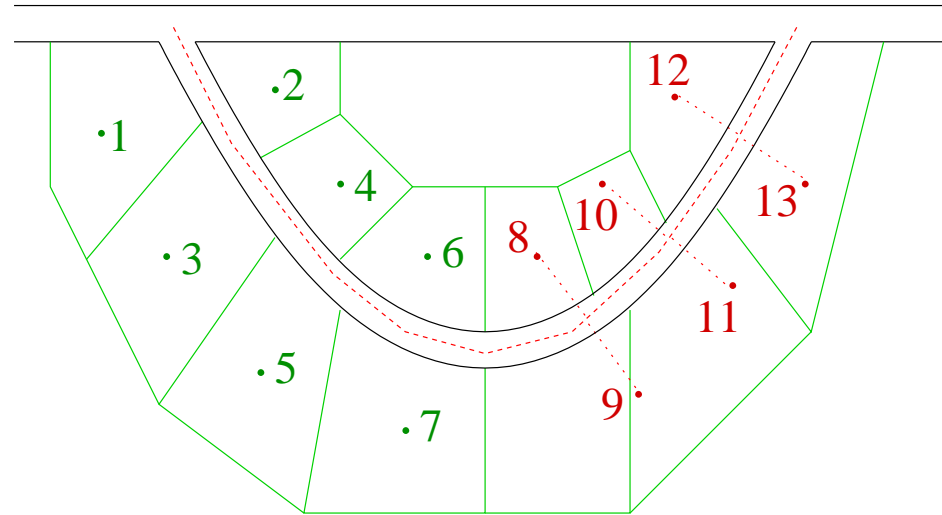
- Street centreline based (many commercial systems)

- Property parcel centre based (our approach)

- A recent study found substantial differences
(specially in rural areas)
Cayo and Talbot; Int. Journal of Health Geographics, 2003

Geocoded National Address File

- Need for a national address file recognised in 1990

- 32 million source addresses from 13 organisations

- 5-phase cleaning and integration process

- Resulting database consists of 22 files or tables

- Hierarchical model (separate geocodes for each)

  - Address sites

  - Streets

  - Localities (towns and suburbs)

- Aliases and multiple locations possible

Simplified G-NAF data model
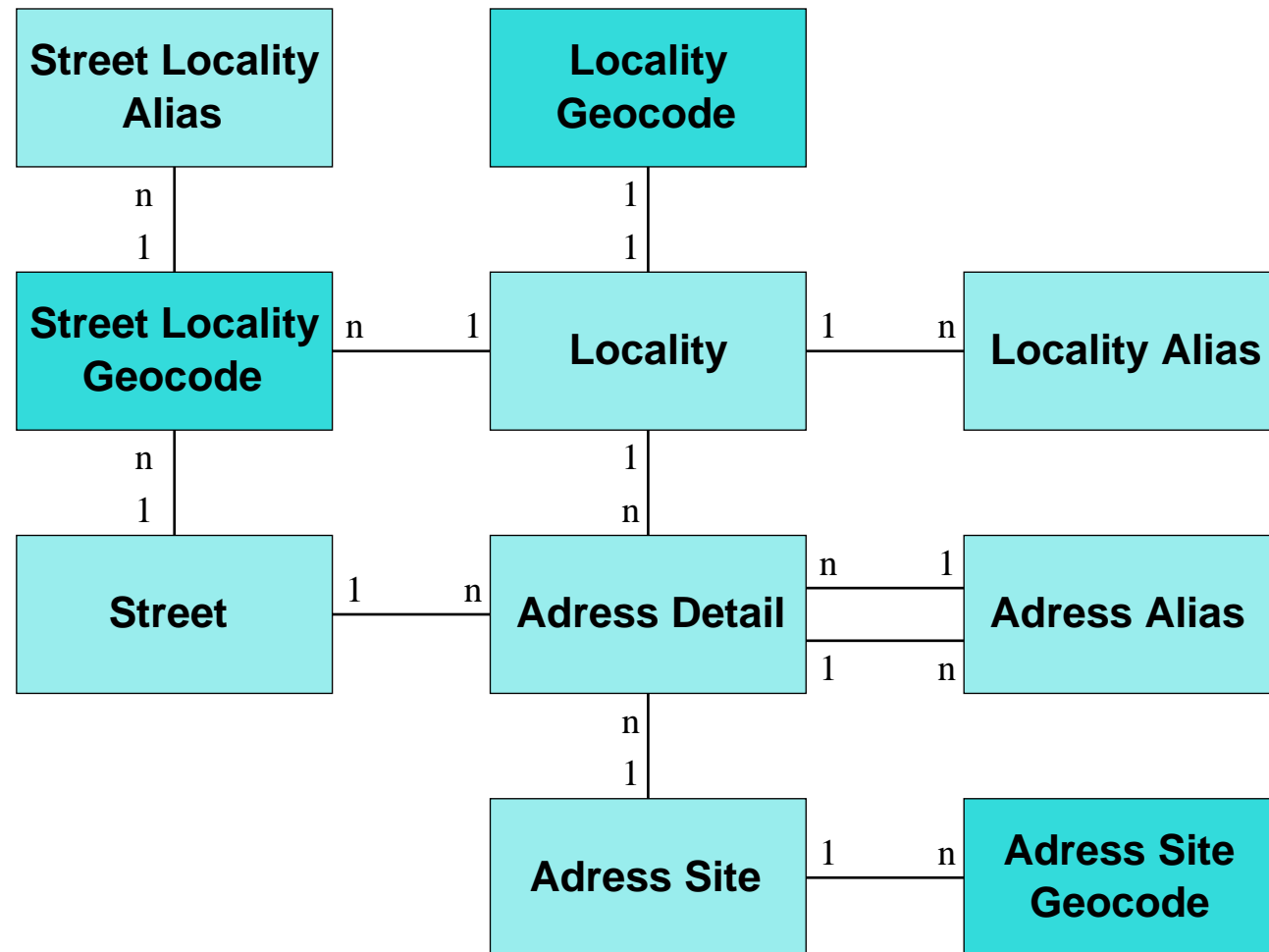
G-NAF file characteristics

G-NAF data file	Number of records / attributes
ADDRESS_ALIAS	289,788 / 6
ADDRESS_DETAIL	4,145,365 / 28
ADDRESS_SITE	4,096,507 / 6
ADDRESS_SITE_GEOCODE	3,336,778 / 12
LOCALITY	5,017 / 7
LOCALITY_ALIAS	700 / 5
LOCALITY_GEOCODE	4,978 / 11
STREET	58,083 / 6
STREET_LOCALITY_ALIAS	5,584 / 6
STREET_LOCALITY_GEOCODE	128,609 / 13

● New South Wales data only

THE AUSTRALIAN
NATIONAL UNIVERSITY

Febrl geocoding system
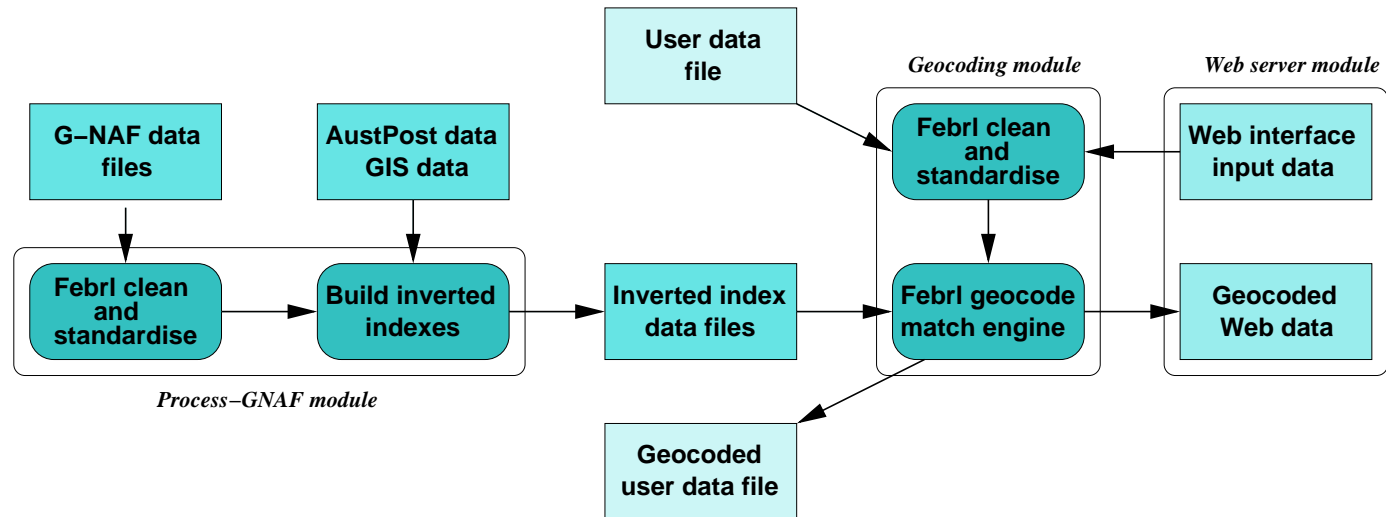
- *Febrl* (Freely extensible biomedical record linkage) (open source, object oriented, written in Python)

  - Experimental platform for rapid prototyping of new and improved linkage algorithms

  - Modules for data cleaning and standardisation, data linkage, deduplication, and geocoding

THE AUSTRALIAN
NATIONAL UNIVERSITY

Address cleaning and standardisation

- Real world data is often *dirty* (missing values, different coding formats, typographical errors, out-of-date data)

- For accurate geocode matching, we want clean data in well defined fields

- *Febrl* address cleaning is a three step process

  1. Input data is cleaned (make lower case, remove certain characters, correct misspellings and abbreviations)

  2. Split input into a list of words and numbers, then tag them (using rules and user definable look-up tables)

  3. Give tag lists to a probabilistic hidden Markov model (which assigns tags to output fields)

THE AUSTRALIAN
NATIONAL UNIVERSITY

HMM standardisation example
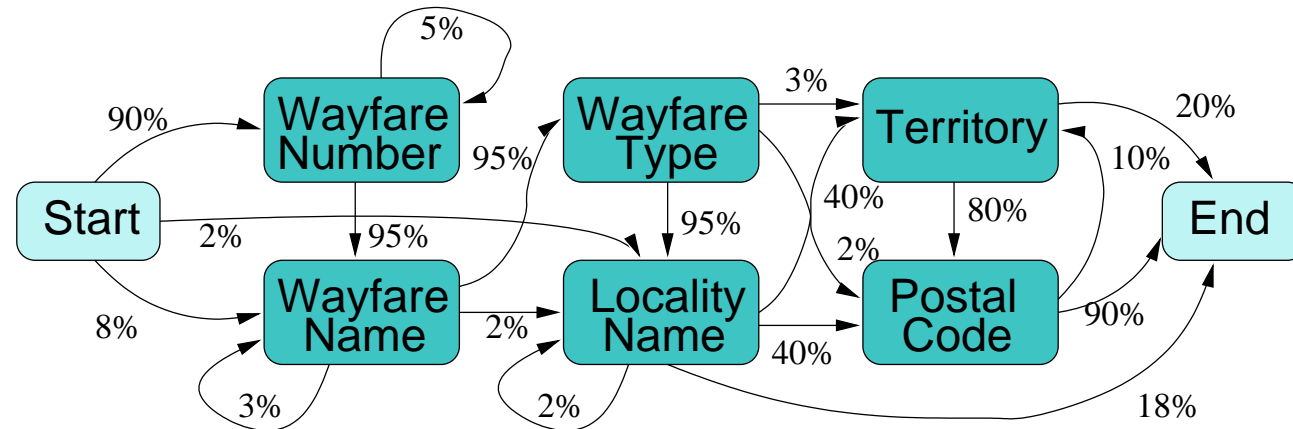
- Raw input: *'73 Miller St, NORTH SYDENY 2060'*

  Cleaned into: *'73 miller street north sydney 2060'*

- Word and tag lists:

  ```
  ['73', 'miller', 'street', 'north_sydney', '2060']
  ['NU', 'UN',    'WT',     'LN',           'PC'  ]
  ```

- Example path through HMM

  ```
  Start -> Wayfare Number (NU) -> Wayfare Name (UN) -> Wayfare
      Type (WT) -> Locality Name (LN) -> Postal Code (PC) -> End
  ```

THE AUSTRALIAN
NATIONAL UNIVERSITY

Processing G-NAF

- Two step process

  1. Do cleaning and standardisation as discussed (to make G-NAF data similar to input data)

  2. Build inverted indices (sets, implemented as keyed hash tables with field values as keys)

     Example (postcode): `'2000':(60310919,61560124)`

- Within geocode matching engine, intersections are used to find matching records

- Inverted indices are built for 23 G-NAF fields

Additional data files

- Use external *Australia Post* postcode and suburb look-up tables for correcting and imputing
  (e.g. if a suburb has a unique postcode this value can be imputed if missing, or corrected if wrong)

- Use boundary files for postcodes and suburbs to build *neighbouring region* lists

  - Idea: People often record neighbouring suburb or postcode if it has a higher perceived social status

  - Create lists for direct and indirect neighbours (neighbouring levels 1 and 2)

Geocode matching engine

- Rules based approach for exact or approximate matching

- Start with address and street level matching set intersection

- Intersect with locality matching set (start with neighbouring level 0, if no match increase to 1, finally 2)

- Refine with postcode, unit, property matches

- Return best possible match coordinates

  - Exact / average address

  - Exact / many street

  - Exact / many locality / no match

First results

Match status	Number of records	Percentage
Exact address level match	7,288	72.87 %
Average address level match	213	2.13 %
Exact street level match	1,290	12.90 %
Many street level match	154	1.54 %
Exact locality level match	917	9.17 %
Many locality level match	135	1.35 %
No match	3	0.03 %

- 10,000 NSW *Land and Property Information* records

- Average 143 milliseconds for geocoding one record on a 480 MHz UltraSPARC II

Geocoding examples

- **Red dots:** FebrI geocoding (G-NAF based)

- **Blue dots:** Street centreline based geocoding

Future work

- Improve probabilistic data cleaning and standardisation

- Improve performance (scalability and parallelism)

- Improve matching algorithm

- Improve user interface (currently simple Web demo)

- Provide feedback on G-NAF to improve data quality

- Develop privacy preserving geocoding

THE AUSTRALIAN
NATIONAL UNIVERSITY