WEB SERVICE
FOR 19TH CENTURY
IRISH PERSONAL NAME MATCHING

PHATTARA WANGRUNGARUN
DR ADAM WINSTANLEY (SUPERVISOR)

14250094 Maynooth U.
DESEM 2014/15
BACKGROUND
SCENARIO

- Historical researchers are interested in pre-20th populations (identification of individuals).
- There are no detailed population records from 19th century Ireland.
- There are "census substitutes" - lists of names from land records, petitions etc.
- However, each record on its own is not usually detailed enough to identify individuals confidently.
- By combining information from these substitutes, we can be more confident of recognising individuals.
RECORD LINKAGE

• Process of associating information from different records about an entity in order to build up a coherent description of the entity.

• Used in historical research, social studies, marketing, administration, government, and genealogy.
RECORD LINKAGE

• Use a person’s **name** as the primary linking item with other attributes being used to corroborate the link as valid.

• Build a temporal and general description of the person and their relationships to others.
Record Linkage

James Duffy
I only have the name

James Duffy
I have his address

James Duffy
I also have his title
Sometimes things do not go well

How about Patt Mohan?

Hmm .. I don't have

Me neither
In the 19th century, in Ireland, there was no standard spelling of most names. Handwriting could be difficult to read and contractions or abbreviations were often used. Many people were not literate, so they asked literate people to write their names.
NAME MATCHING TO THE RESCUE

How about Patt Mohan?

Ah, he probably is Patrick Maughan. Here is his address.

I have Patrick Mohan. Here is the title then.
NAME MATCHING METHODS

• Distance algorithms (e.g. Levenshtein’s)

• Soundex

• Irish soundex

• Lookup table
RESEARCH QUESTIONS

1. Can we provide a web service to match names, where matching can be a complicated process because of the way people record their names.

2. Can the web service act as a platform system for general names or words matching system so that it can be extended to other languages as well.
IMPLEMENTATION
IDEA

BASE NAMES
SHUNAGH
SIMPSON
SINCLAIR
SMITH
SOMERS
SPEED
STERN

TO MATCHED NAMES
GOWAN
O'GOWAN
M'GOWAN
SMEETH
SMILEY
SMITH
SMYTHE
IDEA

BASE NAMES

SHUNAGH
SIMPSON
SINCLAIR
SMITH
SOMERS
SPEED
STERN

TO MATCHED NAMES

GOWAN
O'GOWAN
M'GOWAN
SMEETH
SMILEY
SMITH
SMYTHE

SCORES:
LOOKUP TABLE = 1.0
DISTANCE = 0.667
SOUNDEX = 1.0
IRISH SOUNDEX = 1.0
# Weighting Matching Algorithms

<table>
<thead>
<tr>
<th>Matching Algorithm</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Levenshtein distance</td>
<td>1</td>
</tr>
<tr>
<td>Soundex</td>
<td>3</td>
</tr>
<tr>
<td>Irish soundex</td>
<td>6</td>
</tr>
<tr>
<td>Lookup table</td>
<td>10</td>
</tr>
</tbody>
</table>
ARCHITECTURE

BASE NAME

SINCLAIR
SMITH
SOMERS
SPEED
STERN

M'GOWAN
(WEIGHTED) SCORE: 0.5
SMEETH
SMILEY
SMITH

MATCHING ALGORITHMS

LEVENSHTEIN DISTANCE
WEIGHT: 1

SOUNDEX
WEIGHT: 3

IRISH SOUNDEX
WEIGHT: 6

LOOKUP TABLE
WEIGHT: 10
ARCHITECTURE

BASE NAME
- SMITH
- SOMERS
- SPEED
- STERN

PROCESSING CURRENT BASE NAME 'SMITH' AND TO MATCH NAME 'SMEETH'

M'GOWAN
(WEIGHTED) SCORE: 0.5

TO MATCH NAME
- SMEETH
- SMILEY
- SMITH

MATCHING ALGORITHMS
- LEVENSHTEIN DISTANCE: WEIGHT: 1
- SOUNDEX: WEIGHT: 3
- IRISH SOUNDEX: WEIGHT: 6
- LOOKUP TABLE: WEIGHT: 10
ARCHITECTURE

1. PROCESSING CURRENT BASE NAME "SMITH" AND TO MATCH NAME "SMEETH".

2. PROCEED TO MATCHING ALGORITHMS.

MATCHING ALGORITHMS:
- LEVENSHTEIN DISTANCE: WEIGHT: 1
- SOUNDEX: WEIGHT: 3
- IRISH SOUNDEX: WEIGHT: 6
- LOOKUP TABLE: WEIGHT: 10
ARCHITECTURE
ARCHITECTURE

1. PROCESSING CURRENT BASE NAME 'SMITH AND TO MATCH NAME 'SMEETH'

2. PROCEED TO MATCHING ALGORITHMS

MATCHING ALGORITHMS
- LEVENSHTEIN DISTANCE: SCORE: 0.67, WEIGHT: 1
- SOUNDEX: SCORE: 1.0, WEIGHT: 3
- IRISH SOUNDEX: SCORE: 1.0, WEIGHT: 6
- LOOKUP TABLE: SCORE: 1.0, WEIGHT: 10

3. EACH ALGORITHM CALCULATES ITS SCORE INDEPENDENTLY
ARCHITECTURE

MATCHING ALGORITHMS

1. Processing current base name 'SMITH' and to match name 'SMEETH'.

2. Proceed to matching algorithms.

3. Each algorithm calculates its score independently.

4. Return all scores, calculate overall score based on each score and weight.

BASE NAME

SINCLAIR

SMITH

SOMERS

SPEED

M'GOWAN

(WEIGHTED) SCORE: 0.5

SMEETH

SMILEY

SMITH

LEVENSHEIN DISTANCE

SCORE: 0.67 WEIGHT: 1

SOUNDEX

SCORE: 1.0 WEIGHT: 3

IRISH SOUNDEX

SCORE: 1.0 WEIGHT: 6

LOOKUP TABLE

SCORE: 1.0 WEIGHT: 10
ARCHITECTURE

1. PROCESSING CURRENT BASE NAME 'SMITH' AND TO MATCH NAME 'SMEETH'

2. PROCEED TO MATCHING ALGORITHMS

3. EACH ALGORITHM CALCULATES ITS SCORE INDEPENDENTLY

4.ReturnValue all scores, calculate overall score based on each score and weight

Matching Algorithms

Levenshtein Distance
Score: 0.67, Weight: 1

Soundex
Score: 1.0, Weight: 3

Irish Soundex
Score: 1.0, Weight: 6

Lookup Table
Score: 1.0, Weight: 10

Sinclair

Smith

Smeeth

(M'Gowan)

(Weighted) Score: 0.5

Smiley

Smith

Somers

Speed
ARCHITECTURE

1. Processing current base name 'Smith' and to match name 'Smeeth'

2. Proceed to matching algorithms

3. Each algorithm calculates its score independently

4. Return all scores, calculate overall score based on each score and weight

5. Proceed to next to match name

Matching algorithms:
- Levenshtein Distance: Score: 0.67, Weight: 1
- Soundex: Score: 1.0, Weight: 3
- Irish Soundex: Score: 1.0, Weight: 6
- Lookup Table: Score: 1.0, Weight: 10

Base name: Smith

Sinclair

M'Gowan (weighted score: 0.5)

Smeeth (weighted score: 0.983)

Somers

Smiley

Speed
MVC

**VIEW 1**
WEB PAGE WITH A FORM

**VIEW 2**
WEB PAGE WITH RESULTS

**VIEW 3**
JSON RESULT

```
{name: "Smith", to: "match_names"},
{name: "Speed", score: 0.2, scores:
  method: "levenshtein_distance",
  value: 4, label: null, score: 0.2, weight: 0},
{name: "Govan", score: 0.0, scores:
  method: "levenshtein_distance",
  value: 5, label: null, score: 0.0, weight: 0}
```

**MODEL**

SEPARATE LINES
REMOVE WHITE SPACES
PASS INPUT TO MODEL

MATCHING ALGORITHMS
- Levenshtein Distance
- Soundex
- Irish Soundex
- Lookup Table
MVC

**VIEW 1**
- Web Page with a Form
  - Web Service Input

**VIEW 2**
- Web Page with Results
  - Separate Lines Remove White Spaces Pass Input to Model

**VIEW 3**
- JSON Result
  - {"name": "Smith", "toMatch": "names"["name": "Smith", 
    "speed": "score": 0.2, "scores": "method": "LevenshteinDistance", 
    "value": 4, "label": null, "score": 0.2, "weight": 0.0], "name": "Govan", 
    "score": 0.0, "scores": "method": "LevenshteinDistance", "value": 5, 
    "label": null, "score": 0.0, "weight": 0.0]}

**MODEL**
- Matching Algorithms
  - Levenshtein Distance
  - Soundex
  - Irish Soundex
  - Lookup Table
MVC

VIEW 3

JSON RESULT
{
  "name": "Smith",
  "to_match_names": ["Smith", "Smith_2", "Smith_3"],
  "method": "levenshtein_distance",
  "value": 4,
  "match": [null, "Smith", 0.2, "weight"],
  "model": [null, "Smith", 0.0, "scores", "method": "levenshtein_distance"],
  "value": 5,
  "match": [null, null, 0.0, "weight"],
}

MODEL

MATCHING ALGORITHMS
- Levenshtein distance
- Soundex
- Irish Soundex
- Lookup table

WEB SERVICE CLIENTS

SEPARATE LINES
REMOVE WHITE SPACES
PASS INPUT TO MODEL

WEB INTERFACE INPUT

WEB SERVICE INPUT

WEB PAGE WITH A FORM

WEB PAGE WITH RESULTS

VIEW 1

VIEW 2

VIEW 3

CONTROLLER

2 FORMATTED INPUT

BASE NAME TO MATCH NAME

26
MVC

WEB SERVICE CLIENTS

WEB INTERFACE INPUT

WEB PAGE WITH A FORM

json result

["NAME":"SMITH","TO_MATCH_NAMES":null,"NAME":"SPEED","SCORE":0.2,"SCORES":null,"METHOD":"LEVENSHTEIN","VALUE":4,"LABEL":null,"SCORE":0.2,"WEIGHT":null],"NAME":"GOVERN","SCORE":0.0,"SCORES":null,"METHOD":"LEVENSHTEIN","VALUE":5,"LABEL":null,"SCORE":0.0,"WEIGHT":null]

CONTROLLER

SEPARATE LINES
REMOVE WHITE SPACES
PASS INPUT TO MODEL

WEB INTERFACE INPUT

WEB SERVICE INPUT

MODEL

BASE NAME

TO MATCH NAME

MATCHING ALGORITHMS

LEVENSHTEIN

SOUNDEX

IRISH SOUNDEX

LOOKUP TABLE

WEB PAGE WITH RESULTS
MVC

VIEW 3

JSON RESULT

WEB SERVICE CLIENTS

VIEW 1

WEB INTERFACE INPUT

WEB PAGE WITH A FORM

VIEW 2

WEB PAGE WITH RESULTS

VIEW 3

SEPARATE LINES
REMOVE WHITE SPACES
PASS INPUT TO MODEL

RESULT TO WEB SERVICE

WEB INTERFACE INPUT

RESULT TO WEB INTERFACE

WEB SERVICE INPUT

WEB INTERFACE INPUT

基数

MATCHING ALGORITHMS

LEVENSHTEIN

SOUNDEX

IRISH SOUNDEX

LOOKUP TABLE

BASE NAME

TO MATCH NAME

MATCHING ALGORITHMS

LEVENSHTEIN

SOUNDEX

IRISH SOUNDEX

LOOKUP TABLE

BASE NAME

TO MATCH NAME
WEB SERVICE

```json
{
    "base_names":"Smith",
    "to_match_names":"Smythe\n\nO'Gowan",
    "matching_algorithms":{
        "1":{"name":"LookupTable", "weight":"10"},
        "2":{"name":"LevenshteinDistance", "weight":"1"},
        "3":{"name":"Soundex", "weight":"3"},
        "4":{"name":"IrishSoundex", "weight":"6"}
    },
    "threshold":"0",
    "standard_list":"
}
```
WEB SERVICE

WEB SERVICE RESULT

```
[{
"base_name": "SMITH",
"to_match_names": [
{"to_match_name": "SMYTHE",
"overall_weighted_score": 0.983,
"scores": [
{"method": "LookupTable",
"value": "Matched",
"label": "1897",
"score": 1.0,
"weight": 10},
{"method": "LevenshteinDistance",
"value": 2,
"label": "null",
"score": 0.667,
"weight": 1},
{"method": "Soundex",
"value": "S530 \u003c\u003e S530",
"label": "null",
"score": 1.0,
"weight": 3},
{"method": "IrishSoundex",
"value": "S530 \u003c\u003e S530",
"label": "SMYTHE",
"score": 1.0,
"weight": 6}]
},
{"to_match_name": "O'GOWAN",
"overall_weighted_score": 0.5,
"scores": [
{"method": "LookupTable",
"value": "Matched",
"label": "1897",
"score": 1.0,
"weight": 10},
{"method": "LevenshteinDistance",
"value": 7,
"label": "null",
"score": 0.0,
"weight": 1},
{"method": "Soundex",
"value": "S530 \u003c\u003e 0250",
"label": "null",
"score": 0.0,
"weight": 3},
{"method": "IrishSoundex",
"value": "S530 \u003c\u003e G500",
"label": "GOWAN",
"score": 0.0,
"weight": 6}]
}]
```
WEB SERVICE RESULT

[{
  "base_name": "SMITH",
  "to_match_names": [
    {
      "to_match_name": "SMYTHE",
      "overall_weighted_score": 0.983,
      "scores": [
        {
          "method": "LookupTable",
          "value": "Matched",
          "label": "1897",
          "score": 1,
          "weight": 10
        },
        ...
      ]
    }
  ]
}]}
WEB INTERFACE

BASE NAME

TO MATCH NAMES

Irish Name Matching

Matching methods

Threshold
WEB INTERFACE RESULT
EXTENDING
MATCHING ALGORITHMS
INHERITANCE

MATCHING ALGORITHM

NAME, BASENAME, VALUE, SCORE, WEIGHTED_SCORE, WEIGHT=1

+ CAL_SCORE
+ SOUNDEX_DISTANCE_SCORE

LEVENSHTEIN DISTANCE

+ CAL_SCORE

SOUNDEX

WEIGHT = 3

+ SOUNDEX
+ CAL_SCORE
+ CATEGORY

IRISH SOUNDEX

WEIGHT = 6

+ SOUNDEX
+ CAL_SCORE

LOOKUP TABLE

WEIGHT = 10

+ CAL_SCORE
IMPLEMENTING A NEW MATCHING ALGORITHM
OUTCOME
EVALUATION

• Response speed

• Memory usage

• Dependability
EVALUATION

• Test environment

• MacBook Pro (Retina, 13-inch, Mid 2014), 2.6 GHz Intel Core i5, 8 GB 1600 MHz DDR3, SSD 256Gb

• We need sample inputs, large enough.
STANDARD NAME LIST

- Robert Edwin Matheson classified the surnames in Ireland into 2091 groups.

- Adam Winstanley’s work on this classification looked through these groups and came up with a total 12,944 names in this classification.

- We use all these names to build up our lookup table.

- We also decide to use all these records as a standard name list to test our system performance.
## Response Speed

<table>
<thead>
<tr>
<th>Matching Algorithms</th>
<th>Response Speed (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Levenshtein distance</td>
<td>1,337</td>
</tr>
<tr>
<td>Soundex</td>
<td>2,024</td>
</tr>
<tr>
<td>Irish soundex</td>
<td>2,456</td>
</tr>
<tr>
<td>Lookup table</td>
<td>24,293</td>
</tr>
<tr>
<td>All 4 algorithms</td>
<td>28,786</td>
</tr>
</tbody>
</table>
## MEMORY USAGE

<table>
<thead>
<tr>
<th>MATCHING ALGORITHMS</th>
<th>MEMORY USAGE (BYTES)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Levenshtein distance</td>
<td>48,518,621</td>
</tr>
<tr>
<td>Soundex</td>
<td>53,066,150</td>
</tr>
<tr>
<td>Irish soundex</td>
<td>69,534,598</td>
</tr>
<tr>
<td>Lookup table</td>
<td>244,302,744</td>
</tr>
<tr>
<td>All 4 algorithms</td>
<td>373,544,727</td>
</tr>
</tbody>
</table>
DEPENDABILITY

- **Availability:** The system is available 24x7 on a virtual private server.

- **Reliability:** We ensure that the result from web service and web interface are always exactly the same. From our evaluation the system generates the results in a reasonable time. Memory consumption is acceptable but also needs to be monitored further.
DEPENDABILITY

• **Safety**: Current state of the system does not consider on heavy security aspect. We would leave this attribute for the future works.

• **Integrity**: We have a solid backup of the standard name list which can be regenerated anytime. In future work we also consider a proper database backup solution.

• **Maintainability**: The system welcomes maintenance and extension.
CONCLUSIONS
PROBLEMS

• Take too long to process and also uses too much memory

• Optimisation ideas
  • Memoization
  • find_in_batches
  • Replace RDBMS with NoSQL
FUTURE WORKS

• More phonetic algorithms

• Inheritance for similar matching algorithms

• Improve web interface result

• Test cases
DEMONSTRATION

START USING TODAY AT PHATSRV.CS.NUIM.IE
REFERENCES

- Winstanley, Adam (2014) *Identifying People on the Morpeth Roll, Postgraduate Diploma in Genealogical, Palaeographic & Heraldic Studies 2013-14, University of Strathclyde*

THANK YOU!
Q & A?