RUTGERS
THE STATE UNIVERSITY
OF NEW JERSEY

Duplicate Records Detection Techniques: A Prioritization Approach

Motivation, Research Questions, & Findings

Motivation:

- Prevalent use of operational data as input to Decision Support Systems
- Need to ensure the quality of this data as it affects the output quality of these systems
- Shortage of studies that address the problem of duplicate records in the accounting literature
- Results of duplicate payments detection are usually too many

Research Questions:

1. How can we apply a rule-based system to identify duplicate records?
2. How can we devise a methodology to rank the detected duplicates in order to enable the human users to focus their attention on the more suspicious cases?

Findings:

- Company confirmed the existence of duplicate payments
- Prioritization framework can help deal with large numbers of duplicate candidates

Duplicate Records

Causes:

- Different formats, structures or schema of databases
- Lack of a global or unique identifier
- Human factors (data entry, lack of constraints, intentional)

Detection Methods:

1. Exact matching:
   - Records are *identical*

Name	Address
J.B. Smith	1 Washington Park
J. Smith	1 Washington Park
John Smith	1 Washington Park Ave
John Smith	1 Washington Park Avenue

2. Fuzzy (near-identical) matching (Weis et.al., 2008):
   - Records have *similar* values for certain relevant fields
   - Causes: data entry errors, different value formats, etc. E.g. 10/21/10 vs. October 21, 2010
   - Classified as duplicates based on a threshold and some similarity criteria (e.g. Levenshtein distance)

Duplicate Detection Process

Generalized framework (Weis & Neumann, 2005):

- Phase 1: Candidate definition *(offline)*
  - Determine which objects to compare
- Phase 2: Duplicate definition *(offline)*
  - Determine criteria (description + similarity measure) for candidates to be considered actual duplicates
- Phase 3: Actual duplicate detection
  - Specifying how to detect duplicates candidates and find which ones are true duplicates (blocking or sorting).

Record	Name	Address	Age	Phone
1	John Smith	1 Washington Park	32 yrs	973-123-4567
2	J.B. Smith	1 Washington Park	32 years	1-973-123-4567
3	J. Smith	1 Washington Park	32 years	(973)1234567
4	John Smith	1 Washington Park Ave	32 years	+1-973-123-4567
5	John Smith	1 Washington Park Avenue	32 yrs	+19731234567

Data

Data Description

2 files: (July 2008 – June 2010)

- **Dataset 1:** information on payments to telecom carriers; 21,606 records, 8 variables

- **Dataset 2:** information on check payments; 47,683 records and 51 variables

Software & Algorithm used

Excel (data transformation and preparation)

ACL (duplicates detection)

Algorithm: 3-way match (Payee + Date + Amount)

Algorithms and Findings

Dataset 1

- *(Carrier ID)* + *Effective Date* + *Amount* yielded 82 candidate duplicates
- *(Carrier ID)* + *Entered Date* + *Amount* yielded 168 candidate duplicates
- 3 Commission payments (unauthorized)!

Dataset 2

- (Date, Amount, Vendor) yielded 899 candidates
- (Date, Amount, Vendor, Invoice ID) yielded 33 candidates
- Approximately 13,000 refunds out of 47,683 transactions!

Duplicate Candidates Prioritization

- Large numbers of candidates
- Use a set of criteria to differentiate (rank) between them
- Simply adding a new variable to the algorithm proved suboptimal

Proposed prioritization based on a Composite Score:

$$CS_i = \sum W_{iCr_j}$$

Where CS_i is the Composite Score of the set of duplicate candidates *i*

W_{iCr_j} is the weight of Criterion *j* when applied to the set of duplicate candidates *i*

Proposed set of criteria:

Materiality, missing values, count of similar candidates, frequency per user, frequency per vendor, duplicate invoice number

Prioritization Criteria

- **Materiality:** $W_{i\_Materiality} = (Amt_i)/(\sum Amt_i)$
- **Missing values:** $W_{i\_MissValue} = \begin{cases} 1/(\sum Count_i), & if \text{ the set of duplicate candidates } i \text{ does not have missing values} \\ 0, & Otherwise \end{cases}$
- **Count of similar candidates:** $W_{i\_Count} = (Count_i)/(\sum Count_i)$
- **Frequency per user:** $W_{i\_FreqUser} = (Count_{U_j i})/(\sum Count_i)$
- **Frequency per vendor:** $W_{i\_FreqVndr} = (Count_{V_j i})/(\sum Count_i)$
- **Duplicate invoice number:** $W_{i\_InvID} = \begin{cases} 1/(\sum Count_i), & if\ the\ Invoice\ ID\ is\ the\ same\ for\ the\ candidates \\ 0, & Otherwise \end{cases}$

Prioritization Example

Record #	Vendor ID	Invoice #	Date	$ Amount	Created by
1001	619505	1241225	5/11/2009	268.55	JDoe
2034	619505	1241225	5/11/2009	268.55	JDoe
9418	619505	1241225	5/11/2009	268.55	JDoe
7430	203339		7/7/2009	4119.5	JSmith
6159	203339		7/7/2009	4119.5	JSmith
8332	552751	1325148	10/5/2009	80.35	JDoe
4723	552751	1279869	10/5/2009	80.35	JDoe

For Record 1001 I calculate the following weights:

- $W_{1001\_Materiality} = (Amt_{1001})/(\sum Amt_i)$ =268.55/ 9205.35 = 0.0292
- $W_{1001\_MissValue} = 1/(\sum Count_i) = 1/7 = 0.1429$ (as there are no missing values causing it to be a duplicate candidate)
- $W_{1001\_Count} = (Count_{1001})/(\sum Count_i) = 3/7 = 0.4286$
- $W_{1001\_FreqUser} = (Count_{U_j i})/(\sum Count_i) = 5/7 = 0.7143$
- $W_{1001\_FreqVndr} = (Count_{V_j i})/(\sum Count_i) = 3/7 = 0.4286$
- $W_{1001\_InvID} = 1/(\sum Count_i) = 1/7 = 0.1429$ (Invoice ID are the same)

CS_{1001}=1.8863

Ranking of the example

Composite Scores of all the duplicate candidates in the example:

Record #	Score - Materiality	Score - Missing Values	Score - Count	Score - Frequency by User	Score - Frequency by Vendor	Score - Invoice ID	Composite Score	Rank
1001	0.0292	0.1429	0.4286	0.7143	0.4286	0.1429	**1.8863**	1
2034	0.0292	0.1429	0.4286	0.7143	0.4286	0.1429	**1.8863**	1
9418	0.0292	0.1429	0.4286	0.7143	0.4286	0.1429	**1.8863**	1
7430	0.4475	0.0000	0.2857	0.2857	0.5714	0.0000	**1.5904**	4
6159	0.4475	0.0000	0.2857	0.2857	0.5714	0.0000	**1.5904**	4
8332	0.0087	0.1429	0.2857	0.7143	0.5714	0.0000	**1.7230**	6
4723	0.0087	0.1429	0.2857	0.7143	0.5714	0.0000	**1.7230**	6

Conclusion

Contributions:

- Helped filling the gap in the accounting literature on duplicate records
- Used two real business datasets to illustrate on duplicate payments
- Proposed a candidates prioritization methodology to help users deal with large numbers of duplicates

Limitations:

- Dependence on feedback for answer – suboptimal approach limited by time/budget constraints
- Datasets are not labeled, but real life datasets
- Could not evaluate prioritization methodology due to the above limitations

Future Research:

- Use of fuzzy algorithms
- Use labeled data to evaluate and refine the prioritization technique