UNIVERSITÄT
LEIPZIG

DATA PREPARATION & CLEANING

Chapter 7: Record Linkage – Blocking & Indexing

Victor Christen

AGENDA

- Record Linkage
  - Benefit, Applications & Techniques
- Blocking & Indexing
- Taxonomy
- Traditional Blocking
  - Suffix Blocking
  - Soundex, SLK-581 for person data specialized
- Sorted Neighbourhood
- Automatic Blocking Key Selection
- LSH-Blocking
  - Approximation of the Jaccard similarity

WHAT IS RECORD-LINKAGE

Process to identify records from different or the same data source representing the same entity such as patients, customers, product, publications

Dr Smith, Peter 42 Miller Street 2602 O'Connor
Pete Smith 42 Miller St 2600 Canberra A.C.T.
P. Smithers 24 Mill Rd 2600 Canberra ACT

- Also known as Data Linkage, Data Matching, Entity Resolution, Duplicate Detection, etc.
- Required due to missing global entity identifier across different data sources

BENEFIT

Data Quality

- Resolve inconsistencies
- Integration in master datasets → data is up-to-date

Analysis

- Avoiding statistical bias
- Enable analysis across various data sources
  - E.g., dependencies between diseases and medications
- Feature enrichment

RECORD LINKAGE IN THE ADMINISTRATION

What is ATO Data Matching?

ATO data matching is best described as a two-step process.

1. The ATO collects information about you from:

- your employers,
- your bank and other financial institutions (now including banks overseas),
- health insurance funds,
- BAS Statements,
- superannuation accounts,
- ride share companies (Uber, Didi, Ola, etc.),
- holiday let businesses (Airbnb, etc.), and
- other property information your state has.

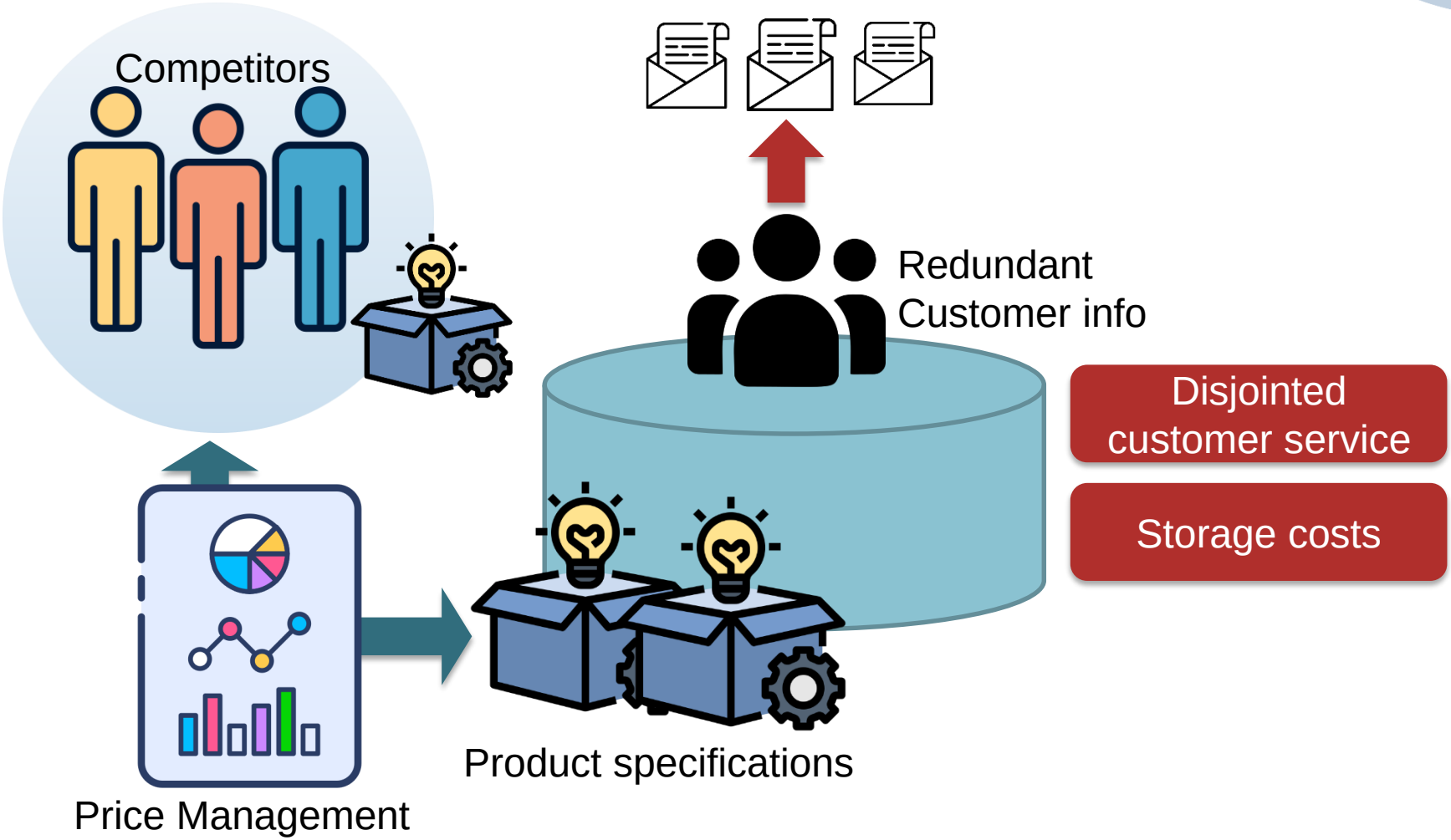
The ATO data matching technology is designed to catch wrongdoers. Are you in the clear?

2. The ATO then compares all this data against the information you provide in your tax return and looks for any differences. So, if you have undeclared income, the ATO will find it easily and they'll contact you for an explanation.

"Tracey Donaldson* says her problems began when a bill arrived in the mail last month.

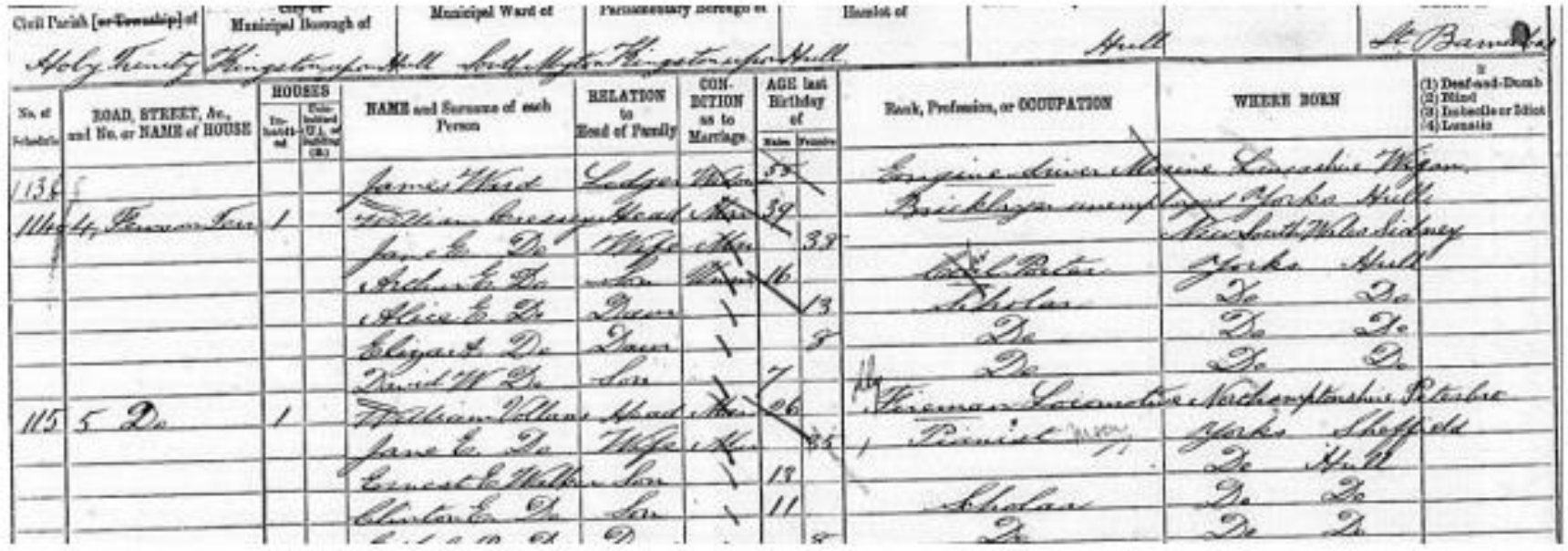
It was a Centrelink debt for $45,500.
The letter was intended for another woman with the same name, who lived in a suburb with the same name, in a different state, said Ms Donaldson, whose name has been changed for legal reasons.
The woman was also born on the same day of the same month as Ms Donaldson, but in a different year. …"
ABC News, 17 Feb 2020

BUSINESS


Competitors
Redundant
Customer info
Disjointed
customer service
Storage costs
Product specifications
Price Management

SOCIAL SCIENCE – HISTORICAL CENSUS DATA

– Analysis of social developments based on historical census data (changes in family structure, healthcare, education, immigration)



No. of Schedule	ROAD, STREET, &c., and No. or NAME of HOUSE	HOUSES Inhabited	HOUSES Uninhabited	NAME and Surname of each Person	RELATION to Head of Family	CONDITION as to Marriage	AGE last Birthday of Males	AGE last Birthday of Females	Rank, Profession, or OCCUPATION	WHERE BORN	(1) Deaf-and-Dumb (2) Blind (3) Imbecile or Idiot (4) Lunatic
1136				James Ward	Lodger	Wid	53		Engine driver Marine	Lincolnshire Wrawn	
1146	4 Pearson Terr	1		William Pressy	Head	Mar	34		Boilerlayer unemployed	Yorks Hull	
				Jane E. Do	Wife	Mar		33		New South Wales Sidney	
				Robert E. Do	Son	Unm	16		Clerk Printer	Yorks Hull	
				Alice E. Do	Dau			13	Scholar	Do Do	
				Edgar J. Do	Dau			9	Do	Do Do	
				David W. Do	Son		7		Do	Do Do	
115	5 Do	1		William Wilson	Head	Mar	36		Fireman Locomotive	Northamptonshire Peterbro	
				Jane E. Do	Wife	Mar		35	Pianist	Yorks Sheffield	
				Ernest E. Wilson	Son		13			Do Hull	
				Charles E. Do	Son		11		Scholar	Do Do	
									Do	Do Do	

TECHNIQUES

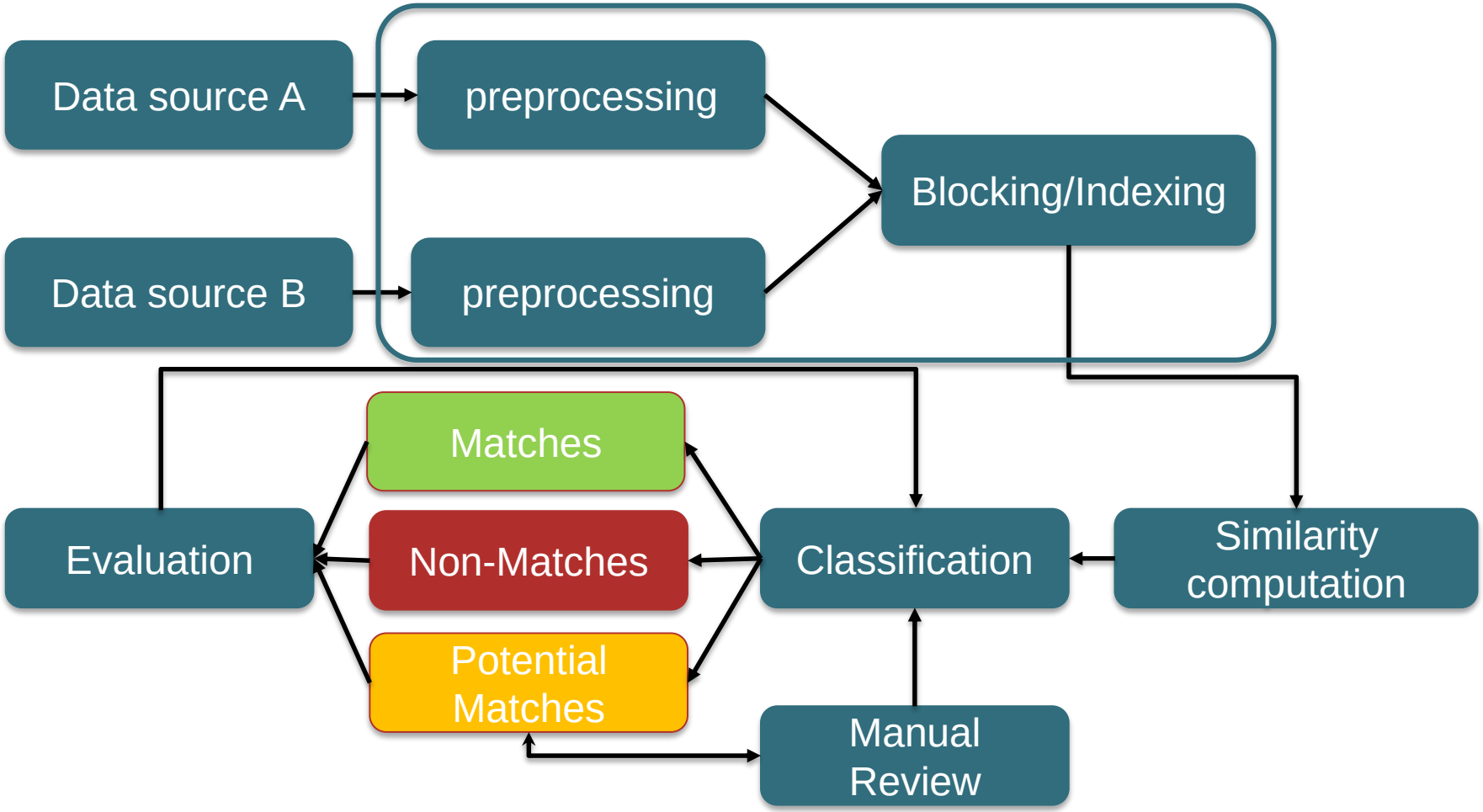
Manual configured

- Rule-based matching (not scalable, difficult to maintain)

Machine learning

- Unsupervised
  - Clustering methods achieve worse results than
- Supervised
  - Generic due to automated training process
  - Utilize high training volume to achieve high quality
- Active Learning
  - Moderate labelling effort to obtain qualitative training data
- Transfer learning
  - Reuse of existing models on an unsolved linkage problem

RECORD LINKAGE PROCESS



BLOCKING & INDEXING

- Number of comparisons between two sets of records is the product of the cardinalities
- Deduplication: n (n-1) / 2
- Feasibility?
  - Data sources with 1 Mio an 5 Mio records
  - $10^6 \cdot 5*10^6 = 5 \cdot 10^{12}$
  - 1 comparison in 1ms $\rightarrow \approx$158 years
- Performance bottleneck caused by attribute comparison between record pairs

TECHNIKEN DER SUCHRAUMREDUKTION

Goal: Reduction of unnecessary comparisons

Blocking	Indexing
• Usage of **Blocking Keys** specifying the **blocking key function** and the attributes being applied on each record • Records with the same blocking key values regarding all blocking keys or at least one build a block • Comparison of records within one block • Overlapping vs disjoint • Independent regarding the similarity computation and the classification step	• Using a similarity metric and a similarity threshold to compare records • Efficient elimination of pairs that can not achieve a similarity above the threshold • simple estimation based on value characteristics such as value length, suffixes • metric space in combination with mathematical defined approximations • Depends on the similarity computation and classification

BLOCKING

- Input: R und S sets of records
- Application of a set of blocking keys (bf, a_1), (bf, a_2), ..., (bf, a_k) for all records r and s from R and S
- Group records by the determined blocking key values

record	BK1	BK2
r1	A	1
r2	A	2
R3	B	1
r4	C	2

record	BK1	BK2
s1	A	1
s2	A	1
s3	C	2
s4	C	2

BLOCKING SCHEMES – CNF VS DNF

- Grouping of all records by blocking key values
- **Conjunctive Normal Form**: Block consists of records, where the values from all blocking keys are the same
  - Concatenation of blocking key values
- **Disjunctive Normal Form**: Block consists of records, where the values from at least one blocking key are identical

Block	Records
A1	r1, s1, s2
A2	r2
B1	r3
C2	r4, s3, s4

Block	Records
A	r1, r2, s1, s2
B	r3
C	r4, s3, s4
1	r1,r3, s1, s2
2	r2, r4, s3, s4

TAXONOMY

Attribute selection	Key type	Schema awareness	Matching awareness
• Supervised • Unsupervised	• Hash-based • Similarity-based	• Schema-aware • Schema-agnostic	• static • dynamic

Redundancy awareness

- Redundancy-free
  - Records is assigned to only one block
- Redundancy-positive
  - Records occur in multiple blocks
  - Similarity of a record pair is related to the number of common occurrences
- Redundancy-neutral
  - Records occur in multiple blocks where most record pairs share the same number of blocks
  - No meaning

Constraint awareness

- Lazy – block creation without any restrictions
- Proactive – restrictions regarding the block creation are addressed such as maximum block size or reducing the number of comparisons within a block

George Papadakis, Dimitrios Skoutas, Emmanouil Thanos, and Themis Palpanas. 2020. Blocking and Filtering Techniques for Entity Resolution: A Survey. *ACM Comput. Surv.* 53, 2, Article 31 (March 2021), 42 pages. https://doi.org/10.1145/3377455

TRADITIONAL BLOCKING(SCHEMA-AWARE, REDUNDANCY-FREE, HASH-BASED, LAZY)

- Blocking function use the identity of an attribute value for an attribute as blocking key value
- Multiple attributes
  - DNF: Concatenation of attribute values
  - High reduction, but high risk to loose correct matches
  - CNF: at least one Blocking key value is the same
- Attribute Selection
  - Schema-aware
  - Small reduction vs high loss of correct matches
  - Skewed blocks → efficiency problem for distributed computations

SUFFIX ARRAYS BLOCKING (SCHEMA-AWARE, REDUNDANCY-POSITIVE, HASH-BASED, PROACTIVE)

Problem: skewed block size distribution

Approach

- Create list of suffixes with the minimum suffix size of l_{min}
- Create a block with a maximum number b_{max} of records
- E.g., Christen, Hansen, Kristen, Kirsten with l_{min} =3, b_{max} = 2
  - Suffix list := [ten, sten, isten, risten, hristen, Christen, sen, nsen, ansen, Hansen, Kristen, rsten, irsten, Kirsten]
  - B_{risten} = {Christen, Kristen} , ~~B_{sten} = {Christen, Kristen, Kirsten}~~

TRADITIONAL BLOCKING – PHONETIC ENCODING

- Frequent occurences of different name variations
- Many variations are valid, e.g., Victor, Vicktor, Viktor, Christen, Kristen
- Names are often recorded by audio
- Phonetic encoding considers different name variations
- Phonetic encoding is language dependent e.g., Soundex vs Kölner Phonetik
- **Issues**
  - Typos lead potentially to different Soundex-Codes

PHONETIC ENCODING- APPROACH

- Transformation of attribute values to a code with a letter and 3 digits

1. Keep the 1st letter
2. Remove the following letters: a, e, i, o, u, y, h, w
3. Replace all consonants starting from position 2 with the following rules
   - b, f, p, v → 1 c, g, j, k, q, s, x, z → 2
   - d, t → 3 l → 4
   - m, n → 5 r → 6
4. Keep the unique neighboring digits
5. Fill up with 0 if the length is smaller than 4, otherwise truncate the length to 4

„victor“→ V236 “vicktor” → V236

TRADITIONAL BLOCKING – SLK-581

- Developed by the Australian Institute for Health and Welfare[1]
- Specialized for person records
- Blocking is also similarity computation
- Idea
  - Selection of the 2nd, 3rd and 5th letter from the last name
  - Selection of the 2nd and 3rd letter from the first name
  - Transformation of the birth data to ddmmyyyy
  - Encode the gender(1=male, 2=female, 9=unknown)
  - Fill up short names with 2, missing values with 999

http://meteor.aihw.gov.au/content/index.phtml/itemId/349895

AUTOMATIC SELECTION OF BLOCKING KEYS

- Selection of „good“ blocking keys are essential for a high reduction ratio of unnecessary comparisons without loosing correct matches

Goal

- Selection of blocking keys for DNF blocking

Idea

- Utilize a (weak) training dataset with record pairs in Matches M and Non Matches NM
- Filter blocking keys generating too large blocks or blocks with too many pairs in NM
- Add blocking keys to the current solution until M is covered by the resulting blocks

AUTOMATIC SELECTION OF BLOCKING KEYS- APPROACH

– Input: data sources **R** and **S**, Training dataset (**M, NM**), $\mathrm{bk\_candidates}$ list of blocking key candidates,

ε Ratio of uncovered record pairs in **M,** $\mathrm{max\_size\_ratio}$ maximum block size ratio

– Output: determined blocking keys

1. Remove blocking keys bk with blocking function bf for attribute a, if

$$max_{b \in B}(|b|) > \mathrm{max\_size\_ratio} \cdot \max(|\mathrm{R}|, |\mathrm{S}|)$$

2. Compute Fisher-Score for each blocking key bk based on record pairs of **M** and **NM**
3. Order blocking keys by determined score

AUTOMATIC SELECTION OF BLOCKING KEYS- APPROACH

3. Order blocking keys by determined score
4. Add iteratively bk to the result, if at least one more record pair is covered than before
5. Repeat steps 1-4 until the number of uncovered pairs is smaller than $\varepsilon \cdot |\boldsymbol{M}|$

AUTOMATISIERTE SELEKTION VON BLOCKING KEYS – FISHER SCORE

- Fisher-Score to measure the coverage of record pairs in **M** and **NM** according to blocking key $\mathbf{bk_k}$
  - Coverage in **M** should be **high** and **small** in **NM**
  - $C_k = \frac{|V_P|(\mu_{p,k}-\mu_k)^2+|V_N|(\mu_{n,k}-\mu_k)^2}{|V_P|\sigma_{p,k}^2+|V_N|\sigma_{n,k}^2}$
- V_P set of vectors $\vec{v}$ representing pairs in **M**
- V_N set of vectors $\vec{v}$ representing pairs in **NM**
- $\vec{v}_k = 1, \text{if } \mathrm{bf_k(r[a])} = \mathrm{bf_k(s}[a])$

  0 else, for a record pair (r, s) w.r.t to a blocking function $\mathrm{bf_k}$ utilizing attribute a

AUTOMATIC SELECTION OF BLOCKING KEYS – FISHER-SCORE

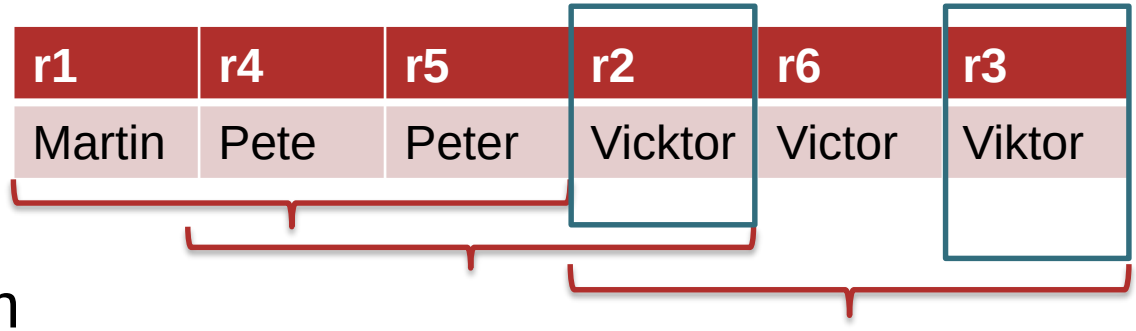
- $C_k = \frac{|V_P|(\mu_{p,k}-\mu_k)^2+|V_N|(\mu_{n,k}-\mu_k)^2}{|V_P|\sigma_{p,k}^2+|V_N|\sigma_{n,k}^2}$

M/NM	rec	BK1	BK2	rec	BK1	BK2	V_P/V_N	$\vec{v}$
M	r1	A	1	s1	A	1	V_P	⟨1,1⟩
M	r2	A	2	s2	A	1	V_P	⟨1,0⟩
NM	r3	B	1	s3	C	2	V_N	⟨0,0⟩
NM	r4	C	2	s4	C	2	V_N	⟨1,1⟩

- $\mu_{p,k}, \sigma_{p,k}^2$ mean and variance of all bits generated from evaluating the key k ∈ K on all pairs in V_P
- $\mu_{n,k}, \sigma_{n,k}^2$ mean and variance of all bits generated from evaluating the key k ∈ K on all pairs in V_N
- μ_k expected value of blocking key k for pairs in **M ∪ NM**

SORTED NEIGHBORHOOD (SCHEMA-AWARE, REDUNDANCY-NEUTRAL, SORTING, PROACTIVE)

- Assumption: similar blocking key values→ likely to be a match
- Sort blocking key values in alphanumerical order
- Compare the last record of a window with length w with all records w-1 (pairwise at the beginning)
- Move the window by 1 position



Problem

- Small windows lead to missing comparisons

SORTED NEIGHBORHOOD BLOCKING - IMPROVEMENTS

- Extended Sorted Neighbourhood
  - Window over blocking keys instead of records

Blocking Key	Records
Martin	R1
Martin	R2
Pete	R3
Pete	R4
Peter	R5
Peter	R6
Victor	R7

Blocking Key	Records
Martin	R1,R2
Pete	R3,R4
Peter	R5, R6
Victor	R7

- Issue
  - High number of comparisons for skewed blocking key value distribution

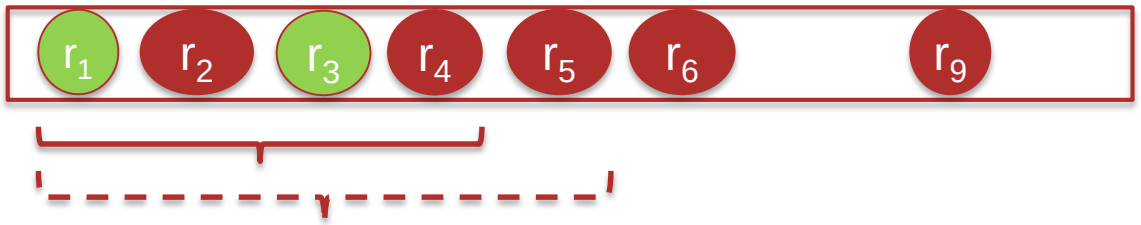
SORTED NEIGHBOURHOOD BLOCKING – IMPROVEMENTS(2)

Adaptive window size

- Blocking Key
  - Adapt window size depending on the blocking key value similarity
- Record similarity
  - Increase/Decrease the window size depending on the amount of record pairs classified as duplicate

DUPLICATE COUNT STRATEGY (SCHEMA-AWARE, REDUNDAN, HASH-BASIERT, PROAKTIV, DYNAMISCH)

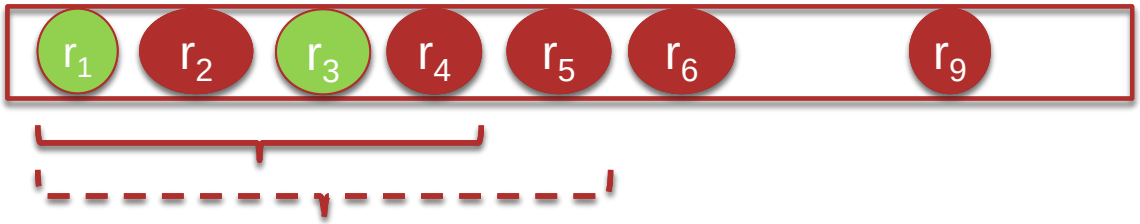
- Idea: Extend the window, if the ratio between duplicates and comparison is high
- The more duplicates are found in a window, the larger the window should be



Uwe Draisbach, Felix Naumann , Sascha Szott, Oliver Wonneberg:Adaptive Windows for Duplicate Detection. ICDE 2012: 1073-1083

DUPLICATE COUNT STRATEGY - WORKFLOW

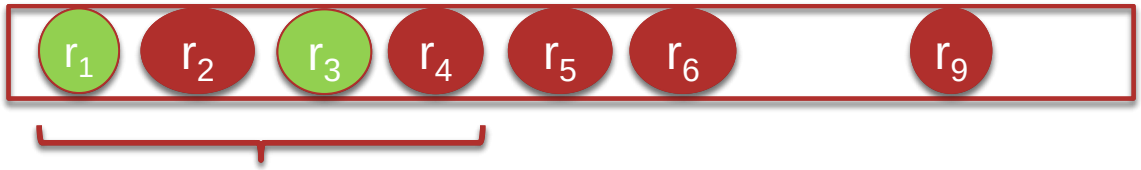
1. Sort record by blocking key value
2. Initialize window with the size w
3. Compare 1st record with all records within the window



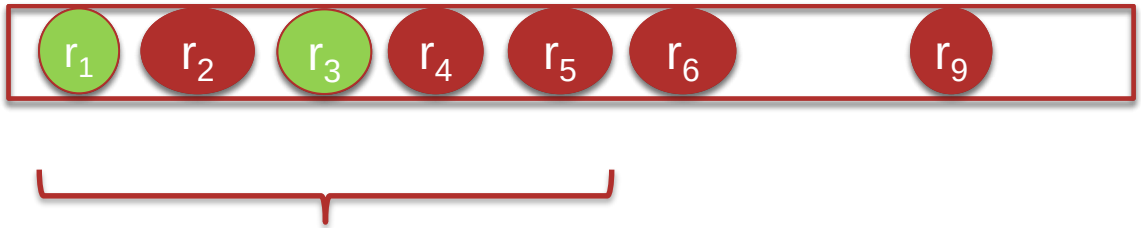
4. Increase window, if $\frac{\#recognized\ duplicates}{\#comparisons} \geq \Phi$
5. Move window (initial size *w*)

Uwe Draisbach, Felix Naumann , Sascha Szott, Oliver Wonneberg:Adaptive Windows for Duplicate Detection. ICDE 2012: 1073-1083

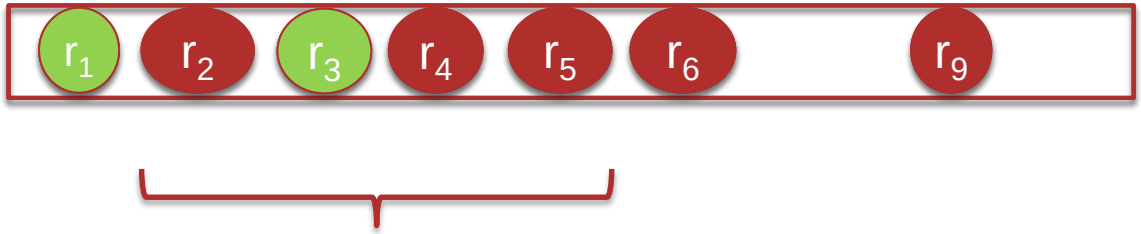
DUPLICATE COUNT STRATEGY - EXAMPLE



$w = 4$
$\Phi = 0.3$
$d/c = 0.33$



$w = 5$
$\Phi = 0.3$
$d/c = 0.25$

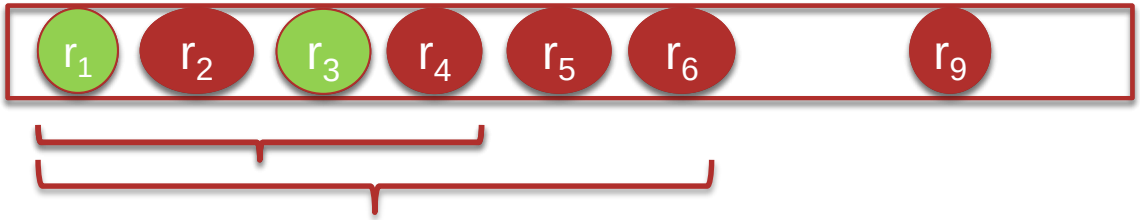


$w = 4$
$\Phi = 0.3$
$d/c = 0$

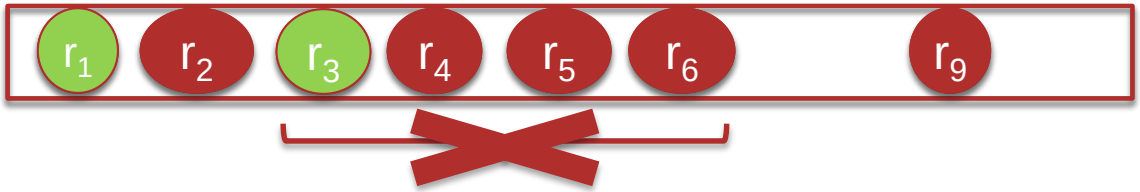
Uwe Draisbach, Felix Naumann , Sascha Szott, Oliver Wonneberg:Adaptive Windows for Duplicate Detection. ICDE 2012: 1073-1083

DUPLICATE COUNT STRATEGY - EXTENSIONS

- Increase by w-1 regarding the identified duplicate



- Skip windows where duplicates were already identified
  - Missing comparisons potentially, if Φ is too high
  - Avoided by $\frac{1}{w-1} = \Phi$



Uwe Draisbach, Felix Naumann , Sascha Szott, Oliver Wonneberg:Adaptive Windows for Duplicate Detection. ICDE 2012: 1073-1083

LSH FOR BLOCKING VIA MIN-HASHING

- Converting of q-gram sets to similarity preserving signatures
  - Similar records cause similar signatures
- Jaccard similarity often used for computing similarities between textual values

e.g. R1(first name) = Peter R2(first name)=Pete

- Q_1={„Pe“, „et“, „te“, „er“} Q_2={„Pe“, „et“, „te“}
- $jaccard_{\text{sim}}(R_1, R_2) = \frac{|Q_1 \cap Q_2|}{|Q_1 \cup Q_2|} = \frac{3}{4}$
- Approximation by One-Hot encoding and bit operations
  - Also computational intensive for large vocabularies→ Transformation to similarity preserving signatures

LSH-BLOCKING - STEPS

Shingling

- Generate set of q-grams for each record
- Represent q-gram sets as One-Hot-Encoding C_i based on all occurring q-grams (vocabulary)
- Length of the encoding ≙ number of q-grams in the vocabulary

MinHash

- Property: If sim(C_1, C_2) is high, P(h(C_1)=h(C_2)) is also high
- Computation of k permutations
- Identification of the position with the first 1 in the permutation

Locality Sensitive Hashing

- Split signature into b for each record
- Compute for each band a hash value with the minHash values of the band
- Comparison of records if the hash values of at least one band are the same

LSH BLOCKING – MINHASH IMPLEMENTATION

- Permutations are not applicable for large sets
- Using random hash functions
  - E.g., $h(x) = ((a \cdot x + b) \bmod p) \bmod N$ where a, b are random numbers and p a prime number with p > N
- Evaluation of k hash functions for each record
- Determine the minimum applying the hash function for each 1 in the One-Hot encoding from 0 to |C|-1

LSH BLOCKING - MINHASH

Example

Q = {ic, ct, to, or}

one-hot-encoding(Q) = 01011001

h(x) = (2x+1) mod 8

Positions of permutations

3	7	0
4	3	4
5	4	3
2	0	6
1	2	7
0	1	2
6	5	1
7	6	5

Original

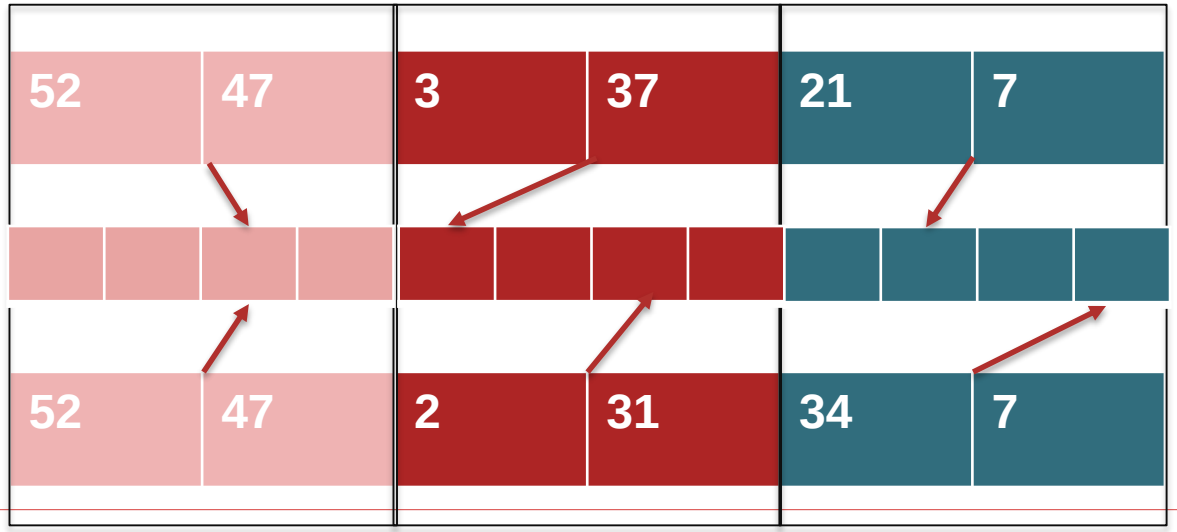
0
1
0
1
1
0
0
1

[1,0,4]

∞	
∞	h(0)=1
3	**h(1)=3**
3	h(2)=5
3	h(3)=7
1	**h(4)=1**
1	h(5)=3
1	h(6)=5
1	h(7)=7

LOCALITY SENSITIVE HASHING

- Quadratic complexity
- Approximation of similar signatures by dividing the signature to b bands with k/b
- Generate a hash $f(s_1, \ldots, s_{k/b})$ for each band b
- Records r_m, r_n are considered as similar if they share at least one common hash value



SUMMARY

- Record Linkage
  - Benefit, Applications, Techniques
- Process of Record Linkage
- Quadratic complexity for record comparisons
  - Reduction of comparisons using blocking and indexing techniques

SUMMARY

- Taxonomy of various blocking methods
- Traditional Blocking
  - Soundex and SLK581 Blocking key functions specified for person data
  - Blocking Key selection is a crucial part
- Sorted Neighbourhood
  - Avoid issues regarding dirty blocking keys
- LSH-Blocking is schema-agnostic to approximate Jaccard similarity for textual values