Entity Resolution: Behind the Scenes

Arnaud Esteve - Devoxx Paris - 2025

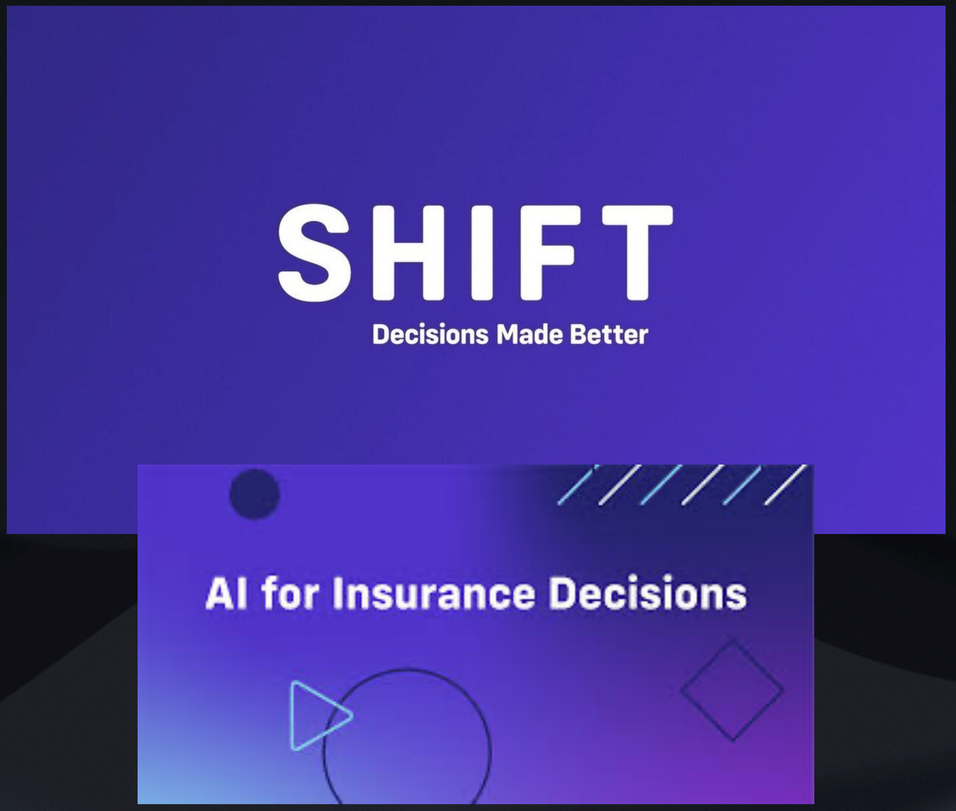
Quick word of Intro

Arnault Estève

Arnaut Esteves

Arnauld Estevez

Arnaud Esteve



Have you ever?

Looked for duplicates in an Excel sheet?

Merged 2 source databases?

Matched products between 2 catalogs?

Then Congrats:

You performed Entity Resolution

A COMMON TASK

Matching entities that look different but are the same

Consolidation

Record Linkage

Reference reconciliation

Householding

Identity Resolution

Data Harmonisation

Deduplication

Object reconciliation

Data Unification

Fuzzy Matching

Identity Reconciliation

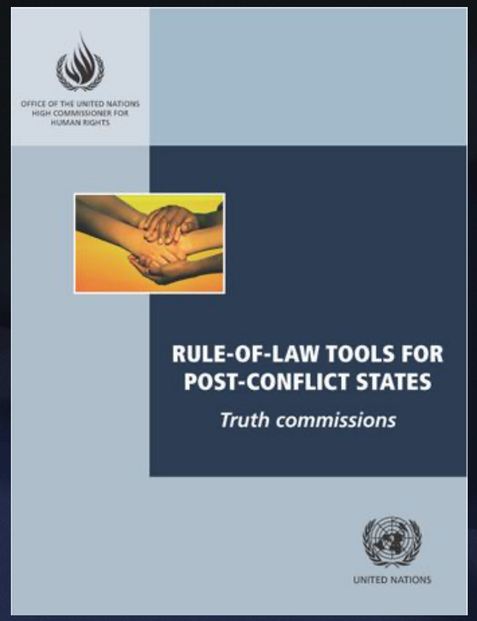
Data Matching

Cross Linking

Entity Matching

Master Data Management

In real life



Credit: Geoff Thale and Adriana Beltran

Did we solve it well, though?

"We relied on email"

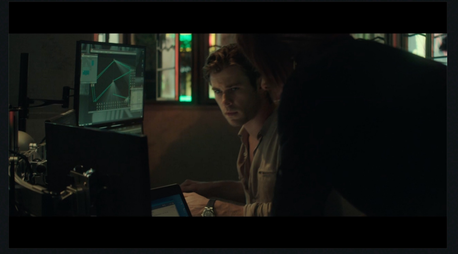
"We did it best effort"

If … and if … and if … or if … and if … or if …

Can we do it differently?

A Private Investigator Story

The “human way”



Highlighting Info	*Preparing records*	Cleansing
Matching dossiers	*Matching records*	Linking
Pulling threads	*Grouping records*	Clustering
Solving identity	*Merging records*	Canonicalisation
Case Review	*Checking records*	Validation

Cleansing:

Dusting for clues

Why is data messy?

- Missing data
- Format inconsistencies and variations: `1923-02-01` vs. `02/01/23`
- Alternatives: `Katherine` ⟷ `Kate` . `Junior` ⟷ `Jr.`
- Typos & non-uniform characters
- OCR / Digitalisation mistakes
- Phonetic Mistakes

CLEANING DATA

It's not just about cleaning, it's about attribute alignment!

A FEW TECHNIQUES

- Removing noise: `<HTML />`, ...
- Normalisation: latin, UPPER, ~~diàcrïtîcs~~.
- Parsing: extract more information
- Encoding: phonetic, geo
- Detecting: outliers / special (eg. `1970-01-01`)
- Variations: William <=> Bill

Make data computable

Encoding

Phonetic example

!!Add NAMEPRISM!!

https://github.com/jamesturk/jellyfish

```
from jellyfish import match_rating_codex, metaphone, nysiis, soundex
name = "Stephen"
soundex_code = soundex(name)
metaphone_code = metaphone(name)
nysiis_code = nysiis(name)
match_rating_codex_code = match_rating_codex(name)
```

Alternate spellings

Original	Stephen	Steven
Soundex	S315	S315
Metaphone	STFN	STFN
NYSIIS	STAFAN	STAFAN
Match Rating Codex	STPHN	STVN

Original	Mohamed	Muhammad
Soundex	M530	M530
Metaphone	MHMT	MHMT
NYSIIS	MAHANAD	MAHANAD
Match Rating Codex	MHMD	MHMD

Alternate spellings + typo

Original	Rashami	Rashmyi
Soundex	R250	R250
Metaphone	RXM	RXMY
NYSIIS	RASAN	RASNY
Match Rating Codex	RSHM	RSHMY

Original	Lucía	Lízia
Soundex	L200	L200
Metaphone	LS	LS
NYSIIS	LACÍ	LASÍ
Match Rating Codex	LCÍ	LZÍ

Parsing & Encoding

Geo-coding + Geo-Hashing Example

GET https://nominatim.openstreetmap.org/search?
q=palais%20congrès%20maillot&format=json&addressdetails=1


```
{
    "place_id": 89127893,
    "licence": "Data © OpenStreetMap contributors, ODbL 1.0. http://osm.org/copyright",
    "osm_type": "node",
    "osm_id": 3969536957,
    "lat": "48.8784493",
    "lon": "2.2837635",
    …
    "address": {
        "railway": "Palais des Congrès",
        "road": "Place de la Porte Maillot",
        "city_block": "Quartier des Ternes",
        "suburb": "17th Arrondissement",
        "city_district": "Paris",
        "city": "Paris",
        "ISO3166-2-lvl6": "FR-75C",
        "region": "Metropolitan France",
        "postcode": "75017",
        "country": "France",
        "country_code": "fr"
    },
    …
}
```


parsed & validated

encoded

hashed

geohash.encode(latitude=48.8784493, longitude=2.2837635)

u09w5fncxe37

GET https://nominatim.openstreetmap.org/search?
q=82%20Boulevard%20Pereire&format=json&addressdetails=1


```
{
    "place_id":89276492,
    "licence":"Data © OpenStreetMap contributors, ODbL 1.0. http://osm.org/copyright",
    "osm_type":"node",
    "osm_id":8668961639,
    "lat":"48.8871762",
    "lon":"2.3042596",
    …
    "address": {
        "amenity":"Etoile - Wagram",
        "road":"Boulevard Pereire",
        "city_block":"Quartier de la Plaine-de-Monceau",
        "suburb":"17th Arrondissement",
        "city_district":"Paris",
        "city":"Paris",
        "ISO3166-2-lvl6":"FR-75C",
        "region":"Metropolitan France",
        "postcode":"75017",
        "country":"France",
        "country_code":"fr"
    },
    …
}
```


encoded

parsed & validated

geohash.encode(latitude=48.8871762, longitude=2.3042596)

u09wh7tumnhm

“palais des Congrès maillot”	u09w5fncxe37
“82 Bd Pereire”	u09wh7tumnhm
“Tour Eiffel, Paris”	u09tuny9c3wb
“Gare des Bénédictins, Limoges”	u00uub4ztwv4

VARIATIONS

Appendix:English given names

Add languages

Appendix Discussion Read Edit View history Tools

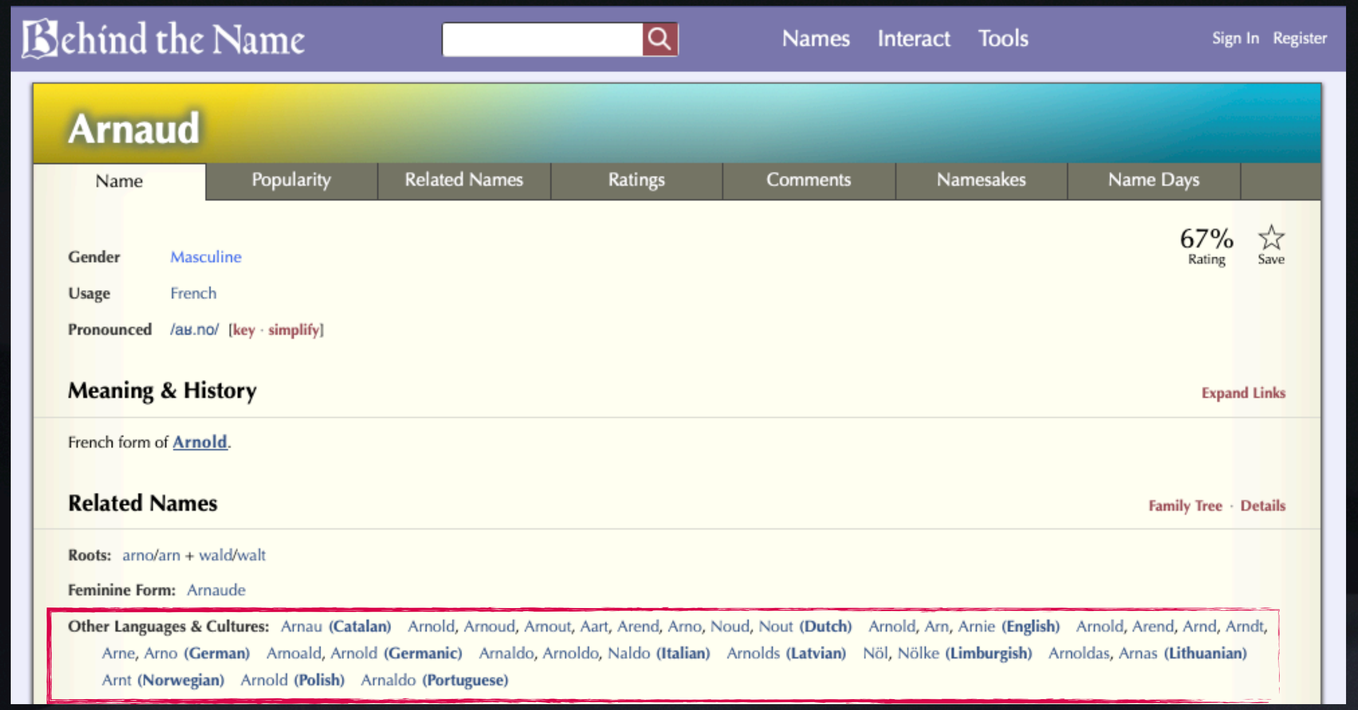
Hypocoristics of English given names [edit]

The following is a list of English given names, followed by their hypocoristics (note: always and as a courtesy ask before shortening someone else's first name, never assume they go by or use the shortened/alternative form).

A [edit]

- Aaron - Ron, Ronny
- Abigail - Abbey, Abbie, Nabby, Abby, Gail
- Abraham - Abe, Bram
- Adrian - Adi, Ari, Ian
- Adam - Addy, Ad
- Agnes - Agg, Nessie, Nessy, Nes
- Alastair - Al, Alex, Ala
- Albert - Al, Bert, Bertie, Alberto, Burt, Abbe
- Alexander - Alex, Lex, Sander, Sandy, Ander
- Alexandra - Alex, Ali, Lee, Lexi, Sasha, Sandra, Xandra
- Alexandre - Alex, Lex, Xander, Sander, Sandy, Anderxie, Sandy
- Ankita - Anki
- Alfred - Al, Alf, Alfie, Fred, Freddy

https://en.wiktionary.org/wiki/Appendix:English_given_names



Behind the Name

Names Interact Tools Sign In Register

Arnaud

Name Popularity Related Names Ratings Comments Namesakes Name Days

Gender Masculine

Usage French

Pronounced /aʁ.no/ [key · simplify]

67% Rating ☆ Save

Meaning & History Expand Links

French form of Arnold.

Related Names Family Tree · Details

Roots: arno/arn + wald/walt

Feminine Form: Arnaude

Other Languages & Cultures: Arnau (Catalan) Arnold, Arnoud, Arnout, Aart, Arend, Arno, Noud, Nout (Dutch) Arnold, Arn, Arnie (English) Arnold, Arend, Arnd, Arndt, Arne, Arno (German) Arnoald, Arnold (Germanic) Arnaldo, Arnoldo, Naldo (Italian) Arnolds (Latvian) Nöl, Nölke (Limburgish) Arnoldas, Arnas (Lithuanian) Arnt (Norwegian) Arnold (Polish) Arnaldo (Portuguese)

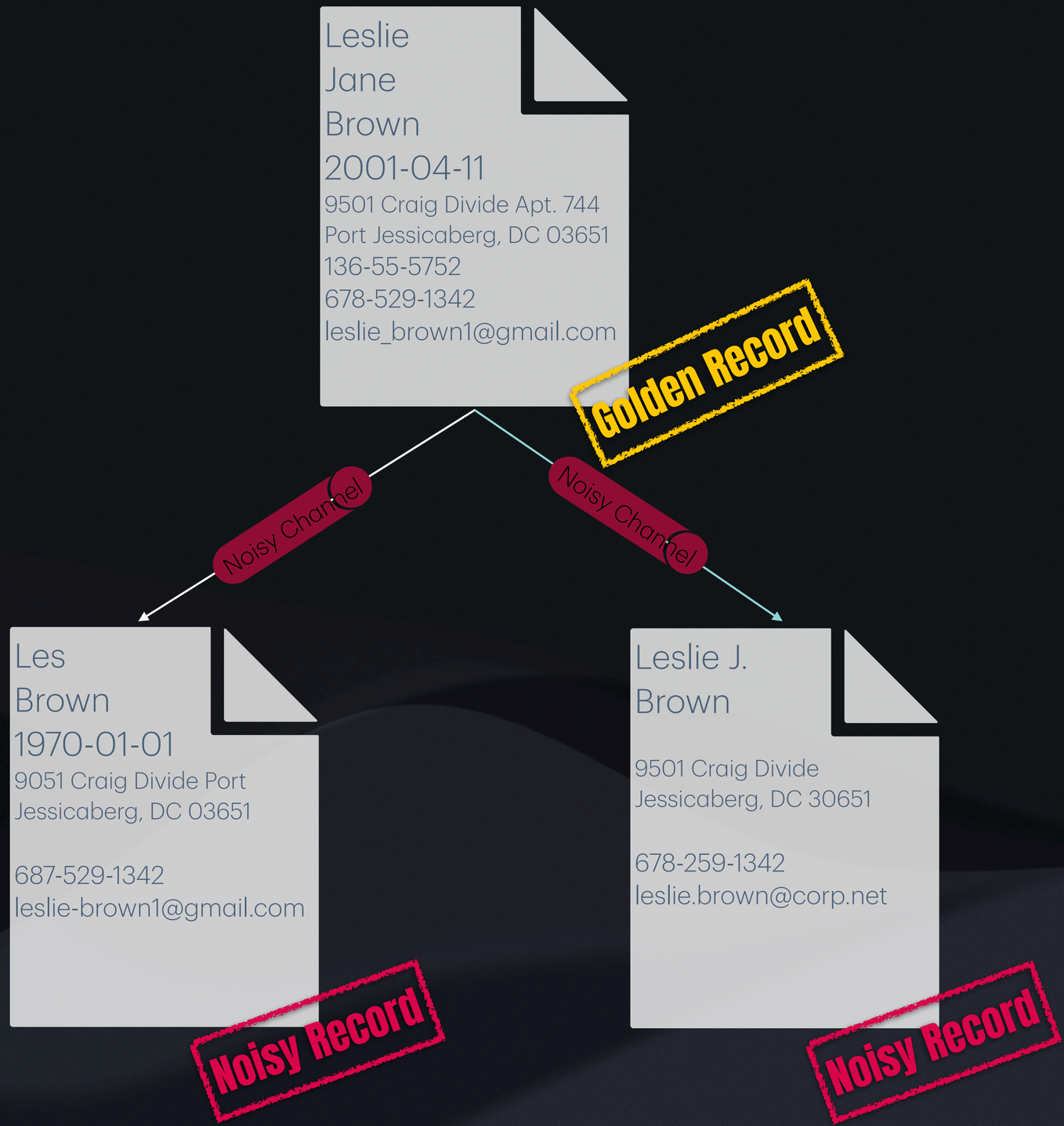
```python
from nicknames import NickNamer

nicks =
NickNamer().nicknames_of("Alexander")
assert isinstance(nicks, set)
assert "al" in nicks
assert "alex" in nicks
assert "sandy" in nicks
assert "alec" in nicks
```

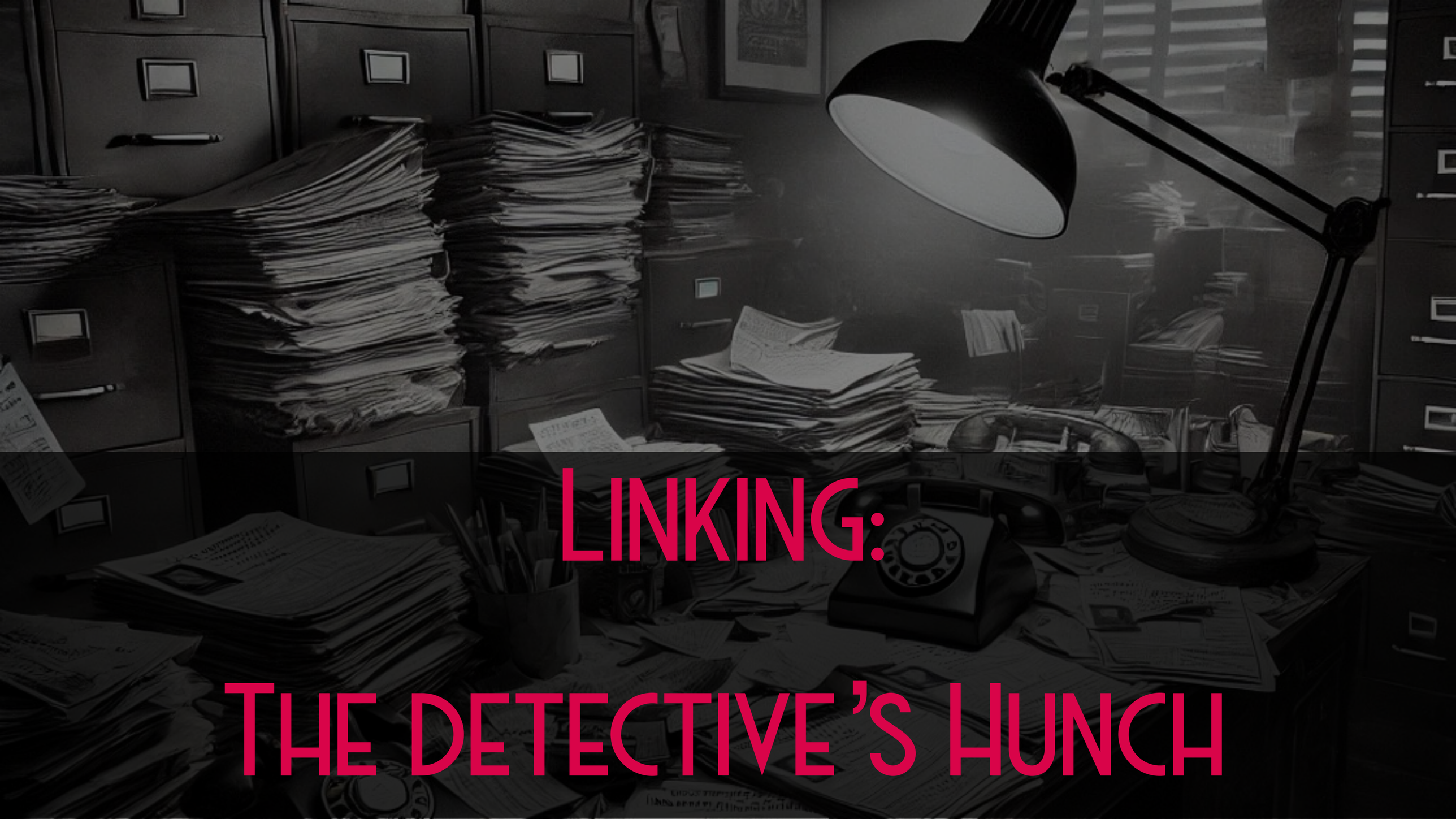
https://github.com/carltonnorthern/nicknames

Setting the scene

Our example

- `1000` Synthetic persons
- Emulates PII safely
- Families
- “Realistic” noise
- `2<n<10` duplicates





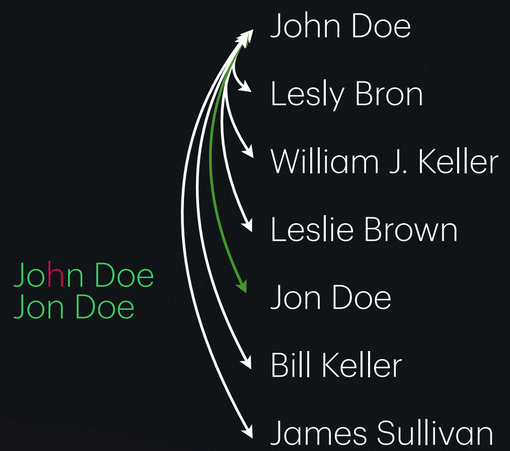
Linking:

The detective's Hunch

Human approach

Comparing pairs

- One record to another



- Problem #1:
  Quadratic Explosion

BLOCKING

TACKLING QUADRATIC EXPLOSION

- Compare records w/in blocks ONLY
- Overlap blocks!!
  - “Same phonetic first name + same phone”
  - “Same phonetic last name + same year”
  - “Same substr(geocode(address), 4)”

How do we compare?

We need to be fuzzy: "how far are these?"

- Numeric:
  - usually easy (subtract)
  - think normalisation
- Strings:
  - edit distance => similarity score
  - there are many!

Make data comparative

CHRSISTOPHAR CHRISTOPHER

Levenshtein = 0.63

(d = 4, |d| = 1 - 4 / 11)

Damerau-Levenshtein = 0.727

(d = 3, |d| = 1 - 3 / 11)

Jaro = 0.776

$$\text{Jaro}(s_1, s_2) = \frac{1}{3}\left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m}\right)$$

m: number of marching characters within a *w* chars window: C, H, R, S, T, O, P, H. m=8 $w = \frac{\max(|s_1|, |s_2|)}{2} - 1$

t: number of transpositions = I <> S. t=1

Jaro-Winkler = 0.843

$$\text{Jaro-Winkler}(s_1, s_2) = \text{Jaro}(s_1, s_2) + L \cdot P \cdot (1 - \text{Jaro}(s_1, s_2))$$

L: Length of common prefix (up to a maximum of 4 chars). L = 3

P: Scaling factor for prefix. P = 0.1 (default value)

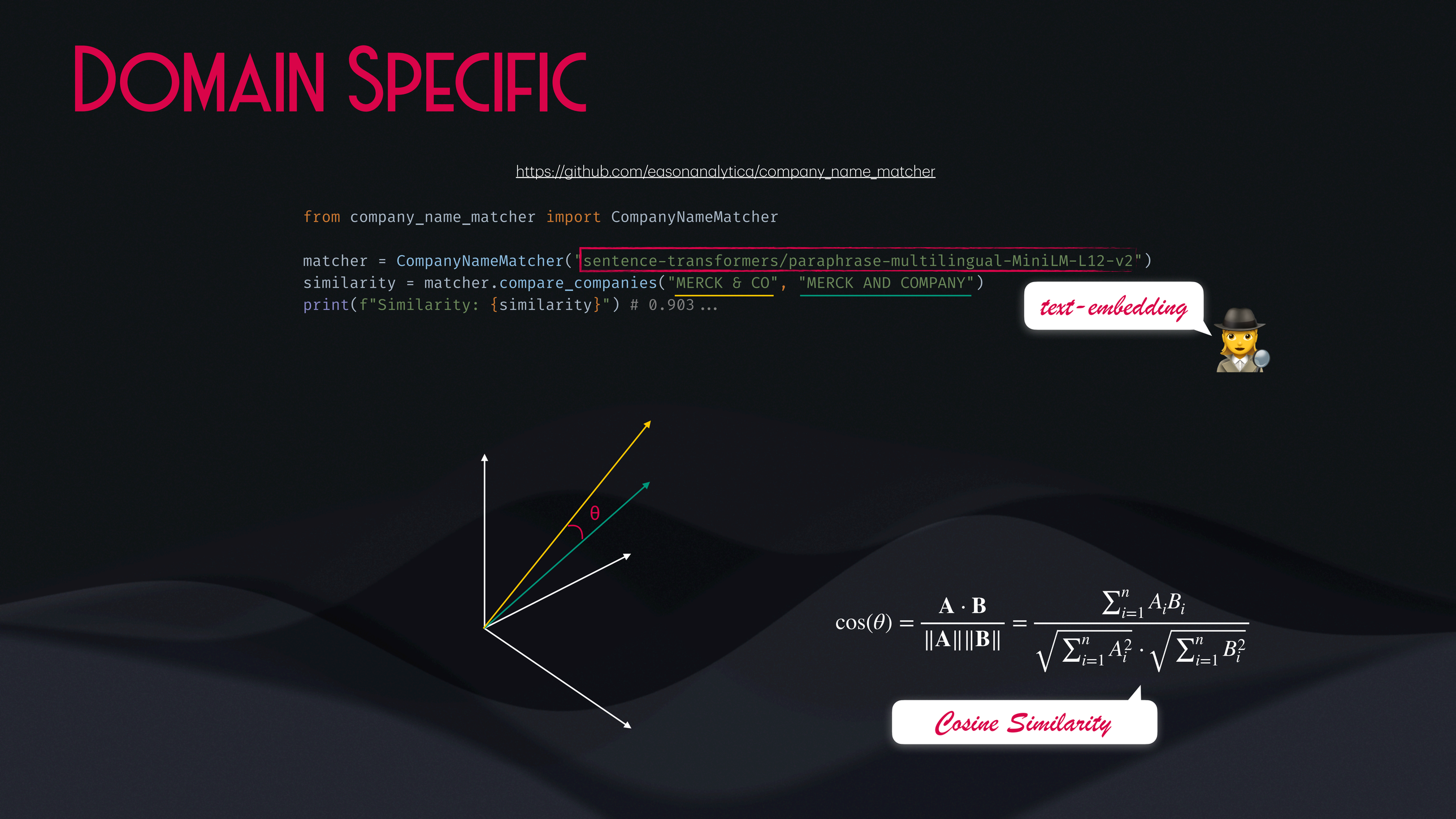
++

DOMAIN SPECIFIC

https://github.com/easonanalytica/company_name_matcher

```python
from company_name_matcher import CompanyNameMatcher

matcher = CompanyNameMatcher("sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2")
similarity = matcher.compare_companies("MERCK & CO", "MERCK AND COMPANY")
print(f"Similarity: {similarity}") # 0.903 ...
```

text-embedding



$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \cdot \sqrt{\sum_{i=1}^{n} B_i^2}}$$

Cosine Similarity

Comparing Distance Metrics

Metric	Best For	Strengths	Weaknesses	Data Type
Levenshtein	Keyboard typos	Intuitive, handles insert/delete/substitute	Slow for long strings	Short string
Damerau-Levenshtein	Keyboard typos	Intuitive, handles insert/delete/substitute/transpose	Slow for long strings	Short strings
Jaro-Winkler	Personal names	Prefix-weighted, robust to minor swaps	Less effective for long strings	Names, short phrases
Hamming	Fixed-length codes (ISBN, IDs)	Extremely fast	Only for equal-length strings	Fixed-length codes (ISBN, IDs)
Jaccard	Sets (or tokenised text)	Order-agnostic, fast for sets	Fails on substring containment	Sets of tokens
Cosine Similarity	Semantic comparison	Handles frequency weighting	Requires vectorization	Vectorised text
Monge-Elkan	Multi-word entities	Robust to token swaps/typos	Computationally intensive	Tokenized phrases
Sørensen-Dice	Short text overlaps	Emphasizes common elements	Sensitive to tokenization	Bi-grams/tri-grams
TF-IDF Cosine	Keyword-rich text	Downweights common terms	Requires corpus statistics	Documents

How do we use distance metrics?

```
if jw(first_name) < 0.9 and lev(dob) < 0.9

if jw(street_name) < 3 and jw(last_name) < 0.9

if cos(company_name) < 0.7 and lev(phone) < 0.8
```

Problem #1:
Combinatorial Explosion

Do you trust more?

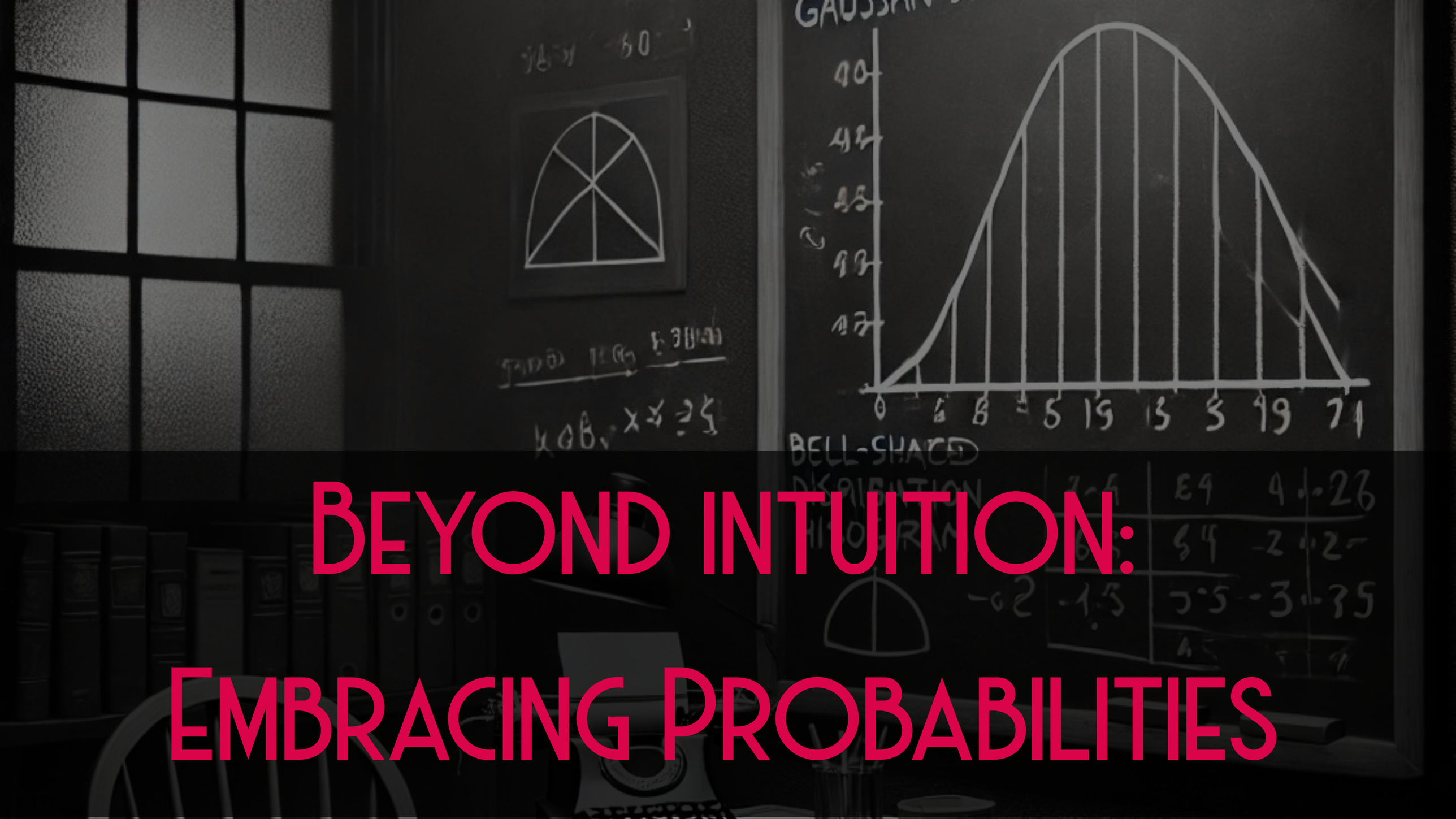
Same first name, same date of birth

Or

Close full name, same date of birth

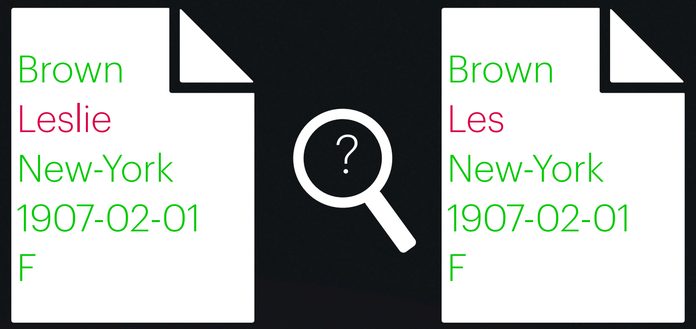
We need

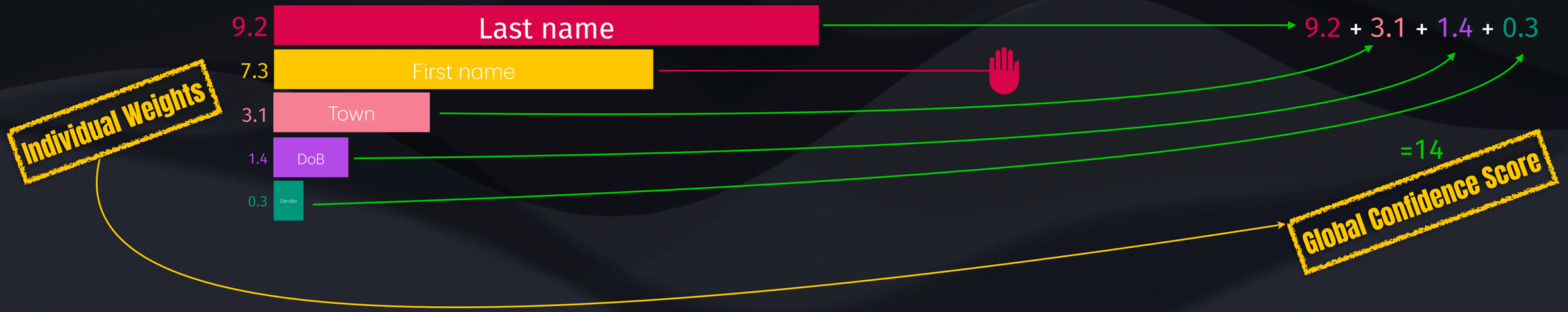
CONFIDENCE + WEIGHTS



Beyond Intuition: Embracing Probabilities

QUANTIFYING CONFIDENCE





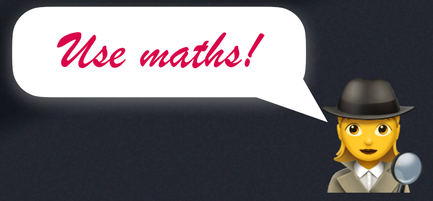
WHAT IS A WEIGHT?

How likely is it for two records to be a match if their last name match?

Bayesian probability […] is an interpretation of the concept of probability, in which, instead of frequency or propensity of some phenomenon, probability is interpreted as reasonable expectation[2] representing a state of knowledge[3] or as quantification of a personal belief.[4]

Intuitively $P(\text{Match} \mid \text{first name matches})$

For n features $(f_1, \ldots, f_n)$ $P(\text{Match} \mid f_1, \ldots, f_n)$



Can we combine though?

$$P(\text{Match} \mid \text{Observation}) = P(\text{Match} \mid f_1, \ldots, f_n)? = P(\text{Match} \mid f_1) \times \ldots \times P(\text{Match} \mid f_n)$$

NOPE!

Posterior probability

Use PRIOR probability instead: $P(\text{Observation} \mid \text{Match})$

Also, let's use ODDS $Odd(\text{Match} \mid \text{Observation}) = \dfrac{P(\text{Match} \mid \text{Observation})}{P(\overline{\text{Match}} \mid \text{Observation})}$

M, U AND WEIGHTS

Bayes Theorem

$$P(\text{Match} \mid \text{Observation}) = \frac{P(\text{Observation} \mid \text{Match}) \times P(\text{Match})}{P(\text{Observation})}$$ λ

$$P(\overline{\text{Match}} \mid \text{Observation}) = \frac{P(\text{Observation} \mid \overline{\text{Match}}) \times P(\overline{\text{Match}})}{P(\text{Observation})}$$ 1 - λ

💡 λ = P(Match) = Probability that 2 random records match

Substitution

$$Odd(\text{Match} \mid \text{Observation}) = \frac{P(\text{Observation} \mid \text{Match}) \cdot \lambda}{P(\text{Observation})} \times \frac{P(\text{Observation})}{P(\text{Observation} \mid \overline{\text{Match}}) \cdot (1 - \lambda)}$$

$$Odd(\text{Match} \mid \text{Observation}) = \frac{\lambda}{1 - \lambda} \times \frac{P(\text{Observation} \mid \text{Match})}{P(\text{Observation} \mid \overline{\text{Match}})}$$

WE HAVE MANY FEATURES

$$Odd(\text{Match} \mid \text{Observation}) = \frac{\lambda}{1-\lambda} \times \frac{P(\text{Observation} \mid \text{Match})}{P(\text{Observation} \mid \overline{\text{Match}})}$$

$$P(\text{Observation} \mid \text{Match}) = P(f_1, \ldots, f_n \mid \text{Match}) = \prod_{i=1}^{n} P(f_i \mid \text{Match})$$

YES!

$$Odd(\text{Match} \mid \text{Observation}) = \frac{\lambda}{1-\lambda} \times \prod_{i=1}^{n} \frac{P(f_i \mid \text{Match})}{P(f_i \mid \overline{\text{Match}})}$$

$\frac{m_f}{u_f} = K_f$

Bayesian Factor

m_f = P(f | Match) = When 2 records match, how likely is it that they have the same last name?

u_f = P(f | ~Match) = When 2 records do not match, how likely is it that they have the same gender?

m measures feature's **ACCURACY**

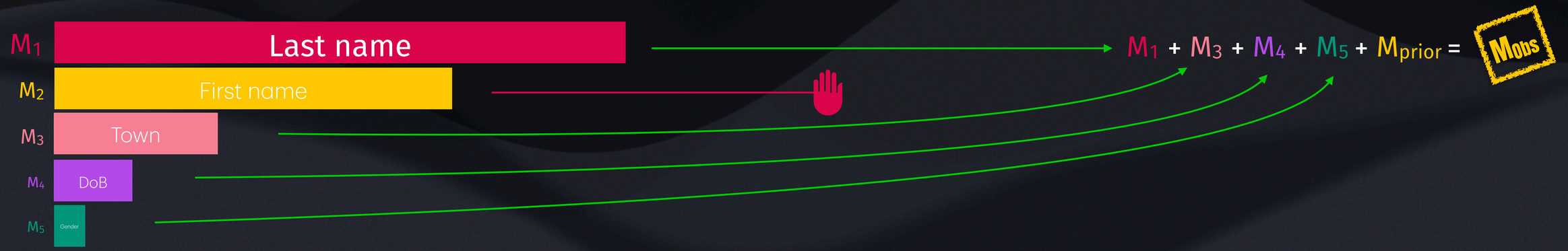
u measures feature's **COINCIDENCE**

WE WANT TO ADD

$$Odd(\text{Match} \mid \text{Observation}) = \frac{\lambda}{1-\lambda} \times \prod_{i=1}^{n} K_i$$

$$\underbrace{log_2(Odd(\text{Match} \mid \text{Observation}))}_{M_{obs}} = \underbrace{log_2(\frac{\lambda}{1-\lambda})}_{M_{prior}} + \sum_{i=1}^{n} \underbrace{log_2(K_i)}_{M_f}$$

$$M_{Obs} = M_{Prior} + \sum_{i=1}^{n} M_{f_i}$$



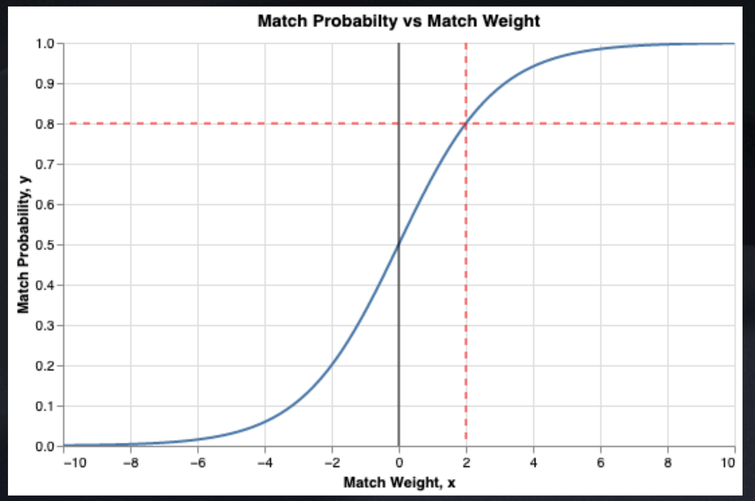
How far are we?

"I was so obsessed with the clues I almost had forgotten her name"

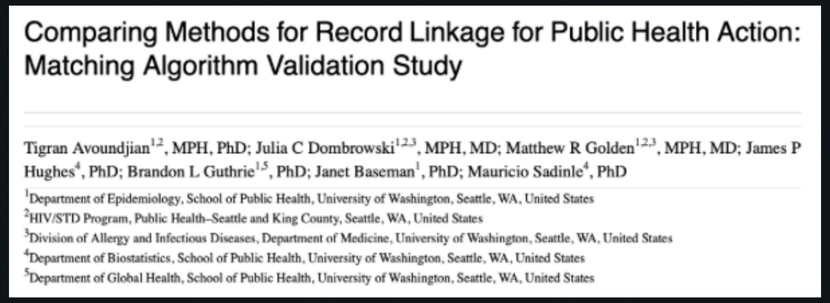
$$M_{Obs} = log_2(\frac{P(\text{Match} \mid \text{Observation})}{P(\overline{\text{Match}} \mid \text{Observation})}) = log_2(\frac{P(\text{Match} \mid \text{Observation}))}{1 - P(\text{Match} \mid \text{Observation})})$$

$$\iff \frac{P(\text{Match} \mid \text{Observation}))}{1 - P(\text{Match} \mid \text{Observation})} = 2^{M_{Obs}}$$

$$\iff P(\text{Match} \mid \text{Observation}) = \frac{2^{M_{Obs}}}{1 + 2^{M_{Obs}}}$$



WHAT DO THE STUDIES SAY?



Comparing Methods for Record Linkage for Public Health Action: Matching Algorithm Validation Study

Tigran Avoundjian[1,2], MPH, PhD; Julia C Dombrowski[1,2,3], MPH, MD; Matthew R Golden[1,2,3], MPH, MD; James P Hughes[4], PhD; Brandon L Guthrie[1,5], PhD; Janet Baseman[1], PhD; Mauricio Sadinle[4], PhD

[1]Department of Epidemiology, School of Public Health, University of Washington, Seattle, WA, United States
[2]HIV/STD Program, Public Health–Seattle and King County, Seattle, WA, United States
[3]Division of Allergy and Infectious Diseases, Department of Medicine, University of Washington, Seattle, WA, United States
[4]Department of Biostatistics, School of Public Health, University of Washington, Seattle, WA, United States
[5]Department of Global Health, School of Public Health, University of Washington, Seattle, WA, United States

https://www.semanticscholar.org/reader/b748ddff0b64a5e45830fdbe664941bd1109a777

[…], we found that the probabilistic algorithms we evaluated had substantially better recall than the selected deterministic algorithms, while the deterministic algorithms had higher precision. […], which diminishes their utility in record linkage scenarios where data quality is poor

[…] although deterministic algorithms offer a high degree of precision, they are highly sensitive to data quality issues and may miss a substantial number of matches […] The recall of deterministic algorithms can be improved by implementing more matching rules […] but this also results in lower precision. […], even with additional match keys, deterministic algorithms still do not reach the level of recall offered by probabilistic algorithms.

Confusion Matrix		Reality	
		Match	Non-Match
Prediction	Match	True Positive	False Positive
	Non-Match	False Negative	True Negative

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

Fellegi-Sunter in Action: Meet SPLINK

Meet Splink

- MIT Licensed, Python
- 🇬🇧 Ministry of Justice
- Implements the Fellegi-Sunter model…

Doc: https://moj-analytical-services.github.io/splink/index.html

… in an interesting way

Estimating Parameters

λ: "How many matches do we expect?"

➡️ Educated guess

u: "How often do people have the same name?"

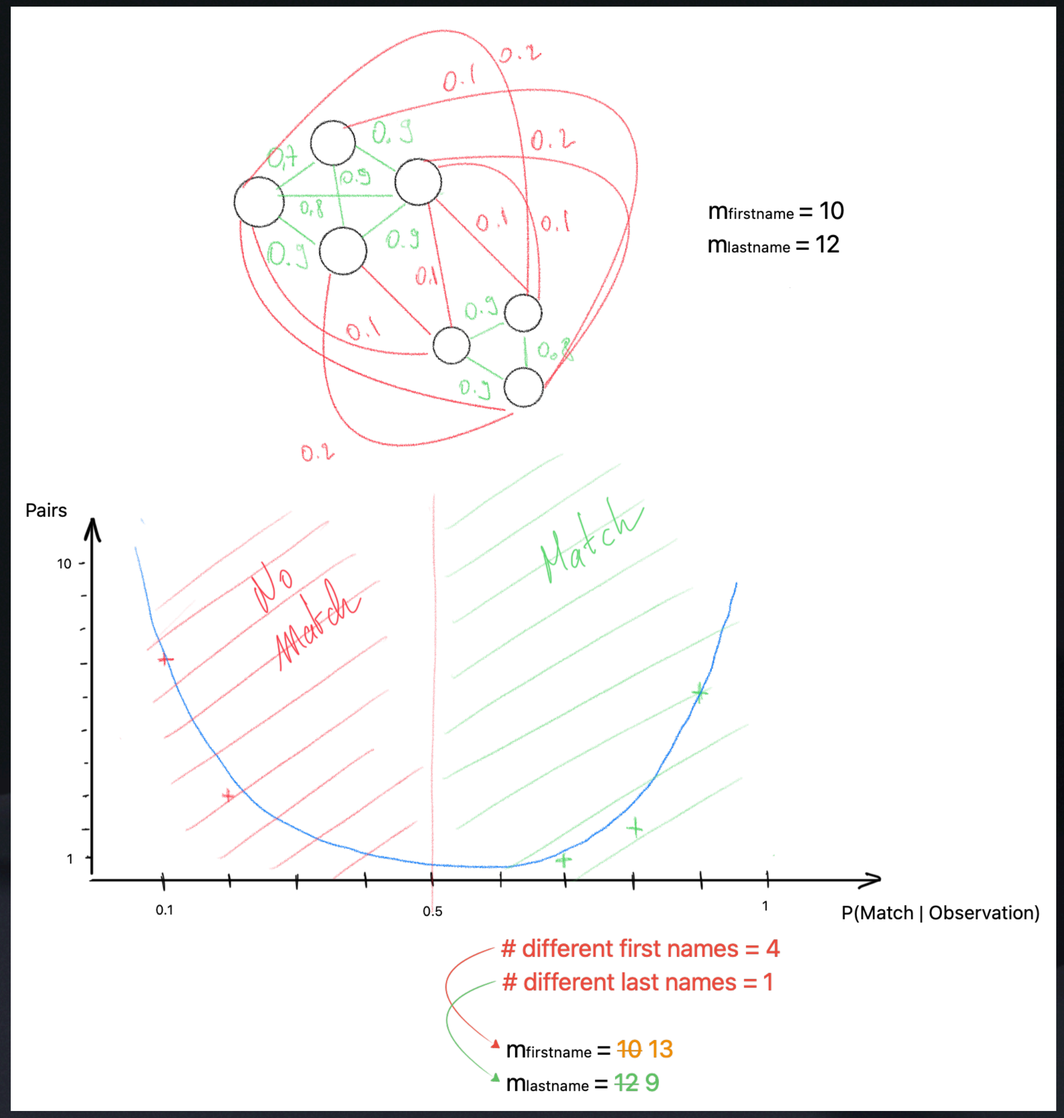
➡️ Random Sampling

Picking Parameters

m: "How clean is the data?"

"How often is the last name mIsTypd?"

4.2. Estimating m



Source: https://www.robinlinacre.com/em_intuition/

How can we even estimate m if we have no info whatsoever? 😢 😢 😢

👀 Look at the pairs of records, don't you see 2 buckets? 🧐

Yes, and??? 🙄 🙄 🙄

Makes it easy to take a decision for what is a **match** and what's not, don't you think? 😏

I guess so 🤔 🤔 what's the point?

Doesn't this sound like labels to you? 😜

🤯🤯🤯

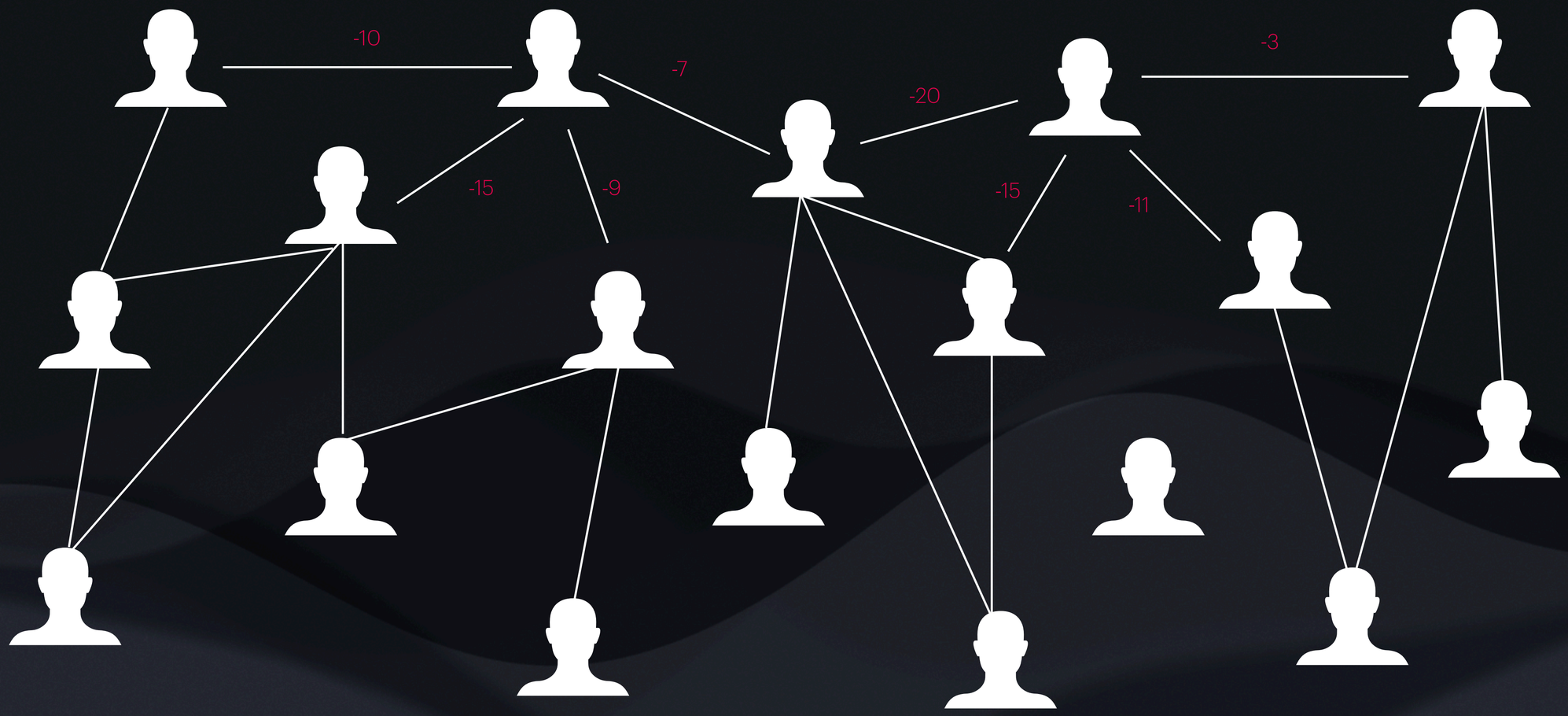
So we can use them to adapt m!

Exactly, Expectation🔄Maximisation!

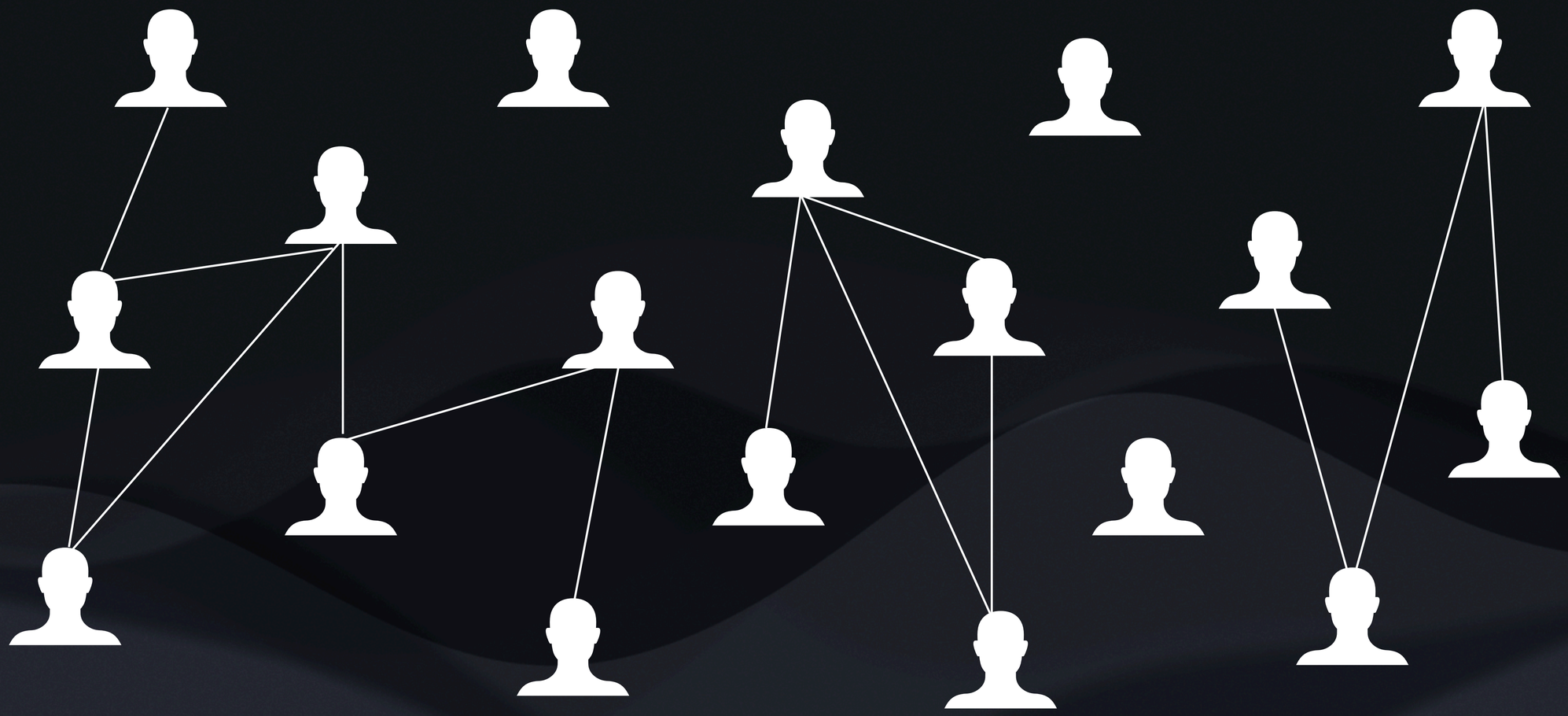
Clustering:

Grouping the evidence

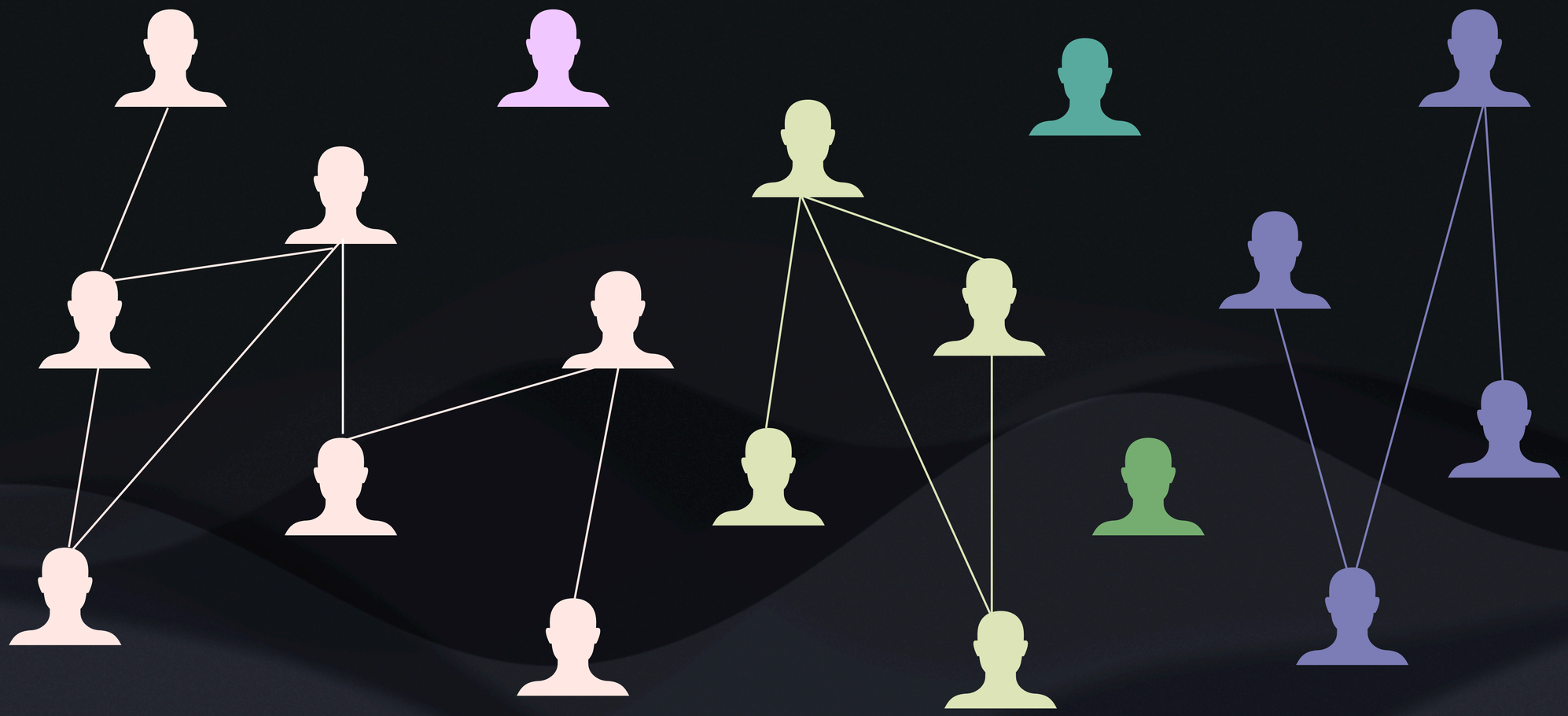
LINKS, NOW WHAT?



LINKS, NOW WHAT?

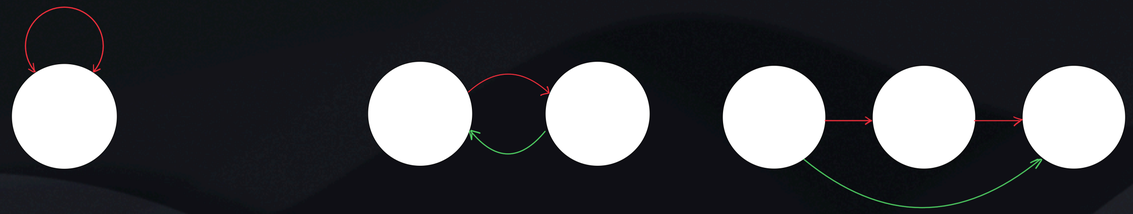


LINKS, NOW WHAT?



Connected Components

In graph theory, a **component** of an undirected graph is a connected subgraph that is not part of any larger connected subgraph. The components of any graph partition its vertices into disjoint sets, and are the induced subgraphs of those sets. A graph that is itself connected has exactly one component, consisting of the whole graph. Components are sometimes called **connected components**.



✅ Reflexive, Symmetric, Transitive relations (edges)

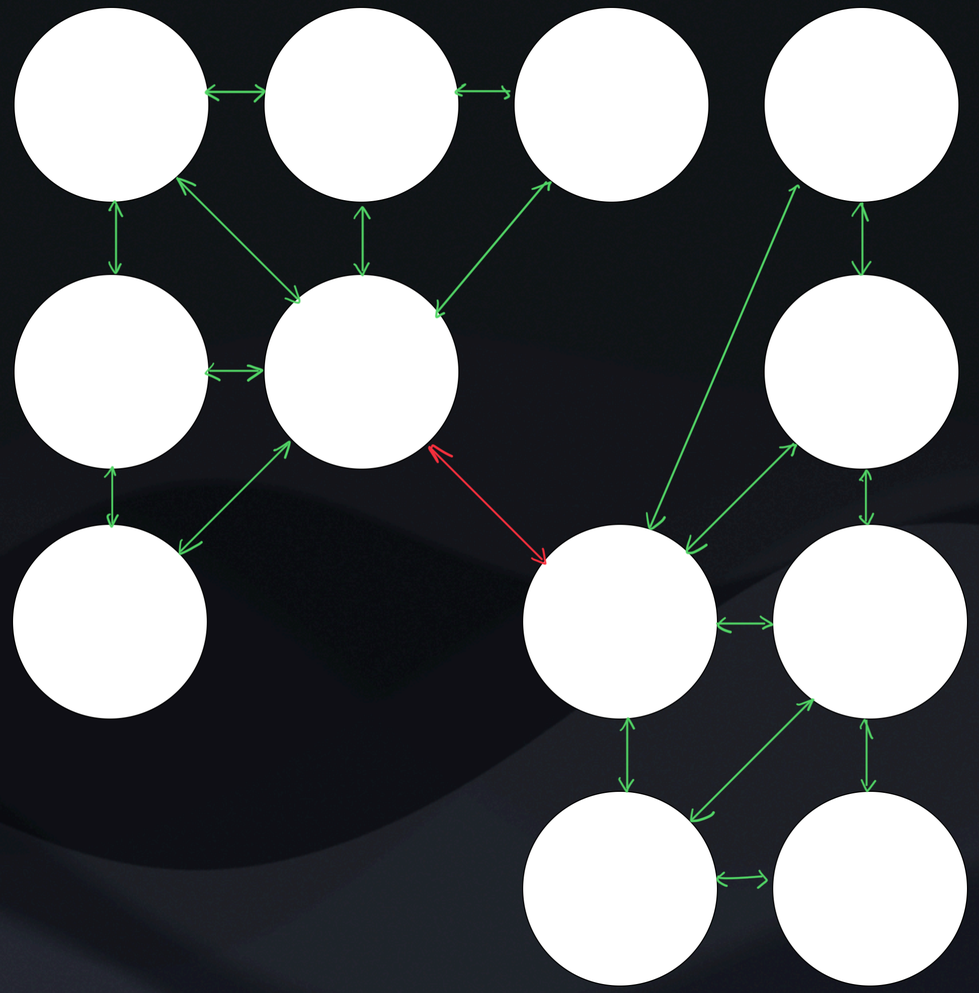
✅ Easy to implement (DFS, Union Find, …)

5.3. Our Example

- Splink uses Connected Components
- Show Splink code for clustering
- Show clusters in Splink Studio

5.4. Connected Components

Weakness

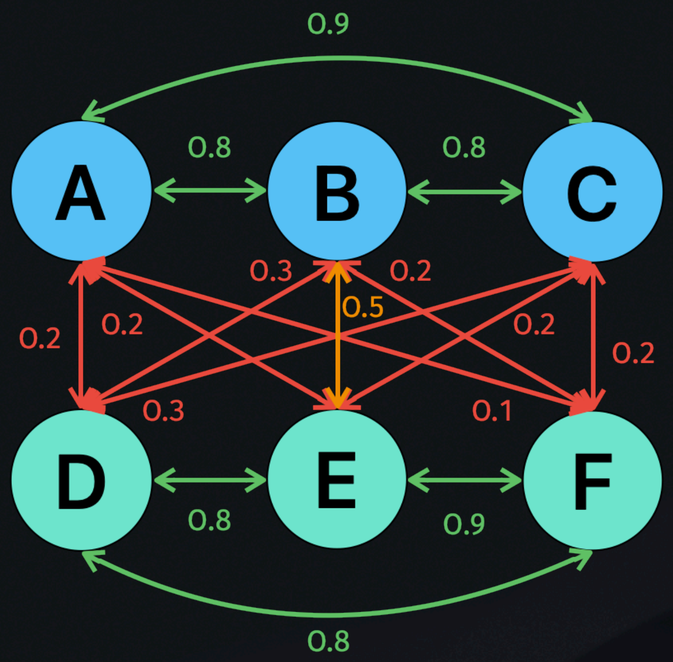


ALTERNATIVES

- Hierarchical Clustering
  - Single / Complete / Average / … Linkage
  - Correlation Clustering
- Density Based
  - OPTICS, HDBSCAN
- Star Clustering

CLUSTERING: OPTICS

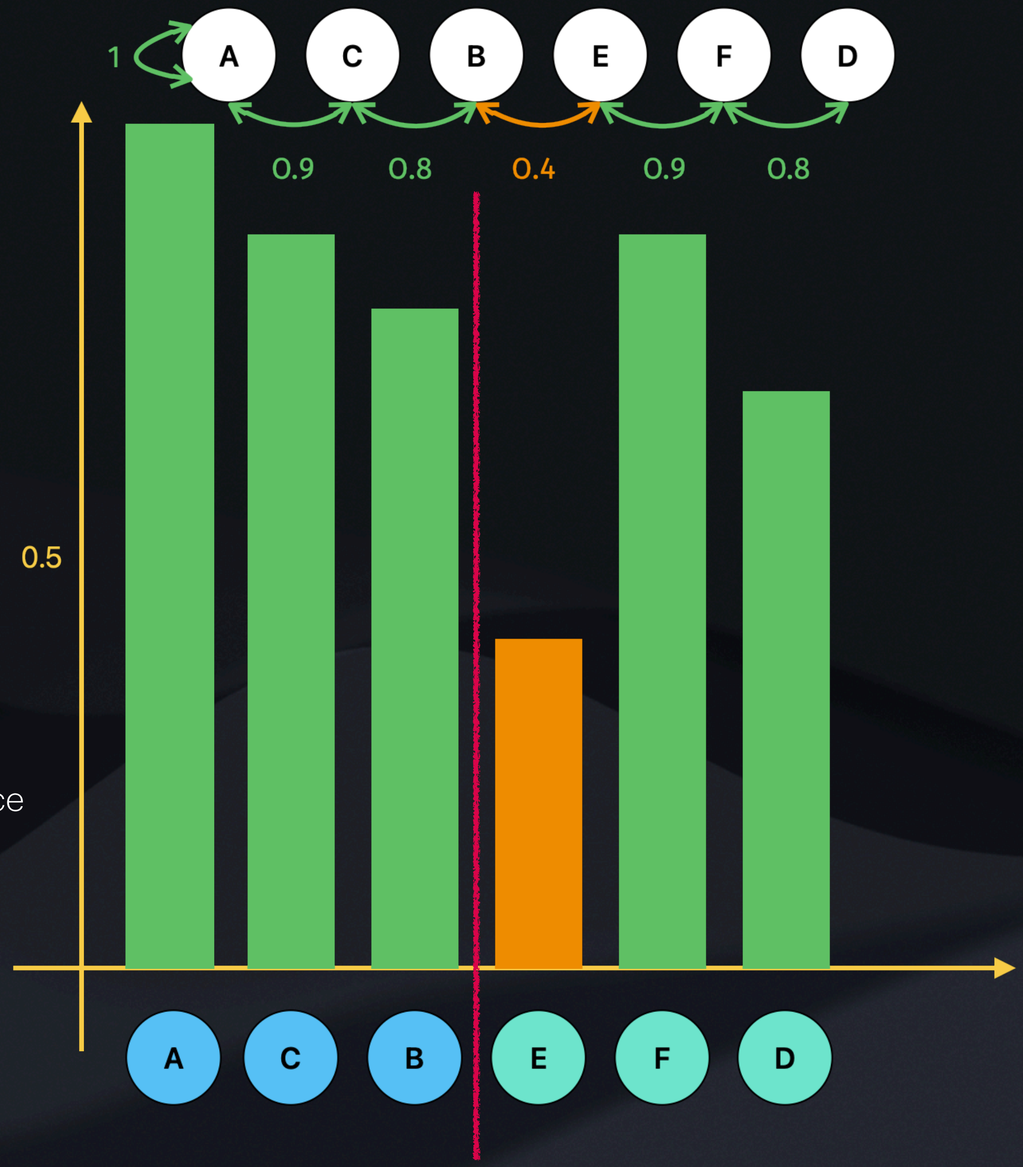
M is a similarity metric!



core dist$(P) = min_{i \in \epsilon} d_{P,i}$

	A	B	C	D	E	F
Core distance (m=3, E=1)	0.2	0.5	0.3	0.3	0.5	0.2

reachability distance



	A	B	C	D	E	F
A		0.8	0.9	0.2	0.2	0.1
B	0.8		0.8	0.3	0.5	0.2
C	0.9	0.8		0.3	0.2	0.2
D	0.2	0.3	0.3		0.8	0.8
E	0.2	0.5	0.2	0.8		0.9
F	0.1	0.2	0.2	0.8	0.9	

	A	B	C	D	E	F
Core distance (m=3, E=1)	0.2	0.5	0.3	0.3	0.5	0.2

	A	B	C	D	E	F
A		0.8	0.9	0.2	0.2	0.2
B	0.8		0.8	0.5	0.5	0.5
C	0.9	0.8		0.3	0.3	0.3
D	0.3	0.3	0.3		0.8	0.7
E	0.5	0.5	0.5	0.8		0.9
F	0.2	0.2	0.2	0.7	0.9	

ALTERNATE METHODS

https://pyjedai.readthedocs.io/en/latest/code/_autosummary/pyjedai.clustering.html

Clustering Method	Approach	Advantages	Disadvantages
Best Match Clustering	Assigns each record to its best match	Simple implementation	May result in overlapping clusters
Center Clustering	Assigns records to central cluster centers	Intuitive; similar to k-means	Requires determination of centers
Connected Components	Groups all connected nodes	Simple and intuitive	Prone to overlinking due to weak links
Correlation Clustering	Optimizes intra-cluster similarity and inter-cluster dissimilarity	Effective when similarity scores are reliable	Computationally intensive for large graphs
Cut Clustering	Removes low-similarity edges to partition the graph	Addresses overlinking; considers edge weights	Selection of cut threshold can be challenging
ExactClustering	Finds optimal clustering configuration	High-quality clusters	Computationally expensive; not scalable
Kiraly MSM Approximate Clustering	Uses MST to approximate clustering	Balances efficiency and quality	Approximation may not capture all nuances
Markov Clustering	Simulates random walks to identify clusters	Detects densely connected regions	Parameter tuning can be complex
Merge Center Clustering	Merges smaller clusters based on similarity	Iteratively refines clusters	Risk of merging dissimilar clusters
Ricochet Clustering	Forms star-shaped clusters around central records	Useful for central entity structures	Assumes existence of central entities
Row Column Clustering	Analyzes similarity matrix structure	Leverages data representation patterns	May miss non-obvious clusters
Unique Mapping Clustering	Ensures unique cluster assignments	Prevents overlapping clusters	May not capture complex relationships

Canonicalisation: Assembling the Truth

Common techniques

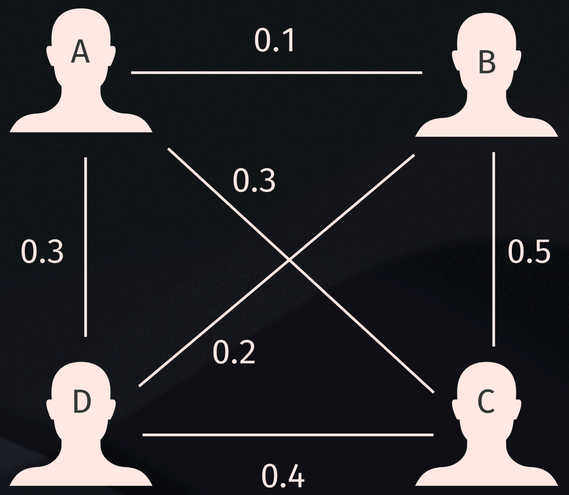
Heuristics

- Pick a random record in the cluster
- Majority Voting: most common value
- Mean, median, weighted average for numerical value
- Most informative: longest string, most decimals
- Prioritised Source: some are more trusted

Common Techniques

Mini-Max

Remember we have metrics!



A	max(0.1, 0.3, 0.3) = 0.3
B	max(0.1, 0.5, 0.2) = 0.5
C	max(0.3, 0.5, 0.4) = 0.5
D	max(0.3, 0.2, 0.4) = 0.4

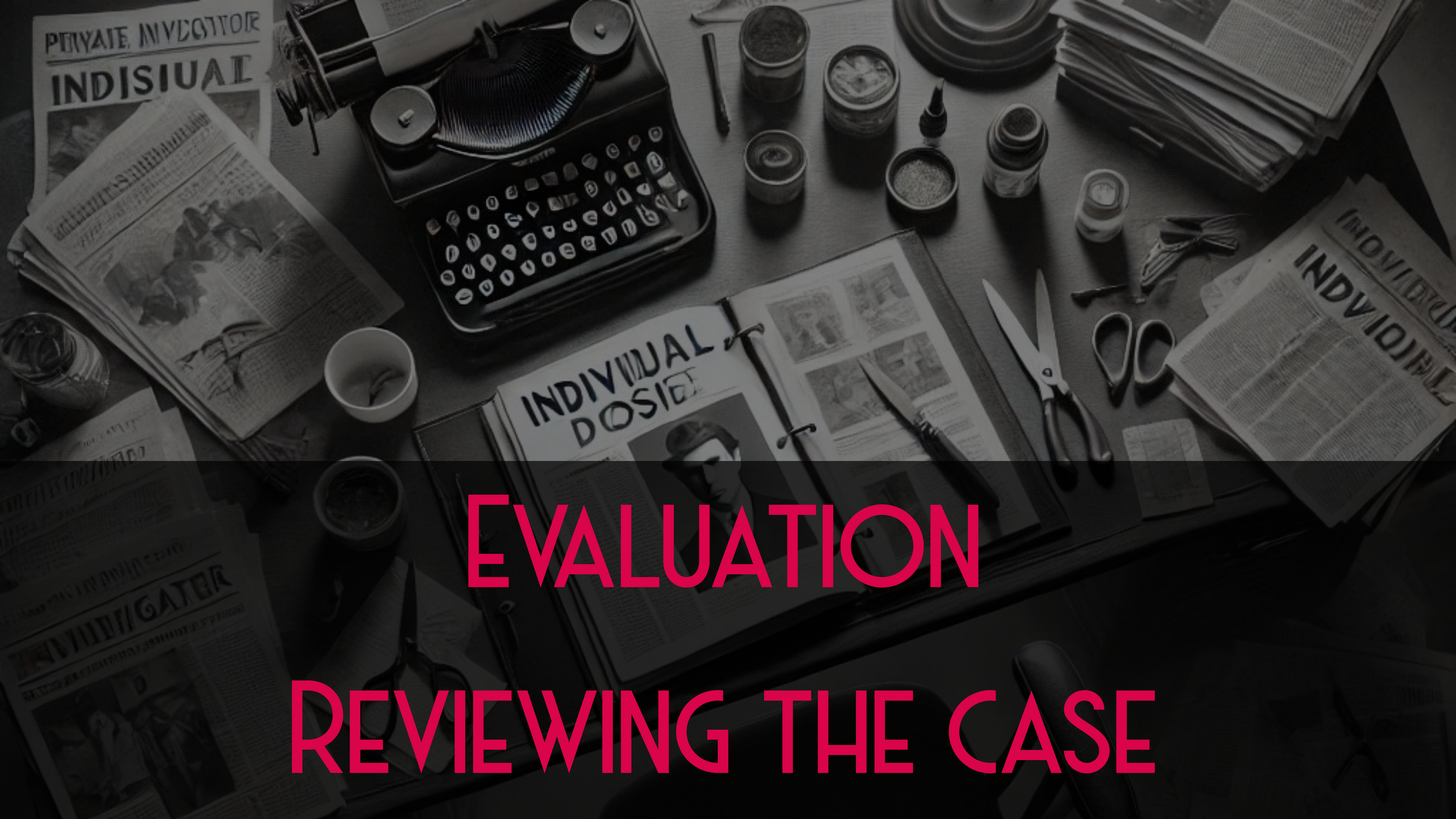
min

Confidence Score

We are probabilistic for a reason!

- Bayesian approach => each link has P(Match) (or M score)
  - Use it!
- => Use Mini-Max with score

- Hybrid approach:
  - similarity + probability



Evaluation

Reviewing the case

Monitor vs. Evaluate

Monitor

Continuous supervision

Unsupervised, no labelling

Birds-Eye view

Identify suspicious clusters

Evaluate

Performance metrics

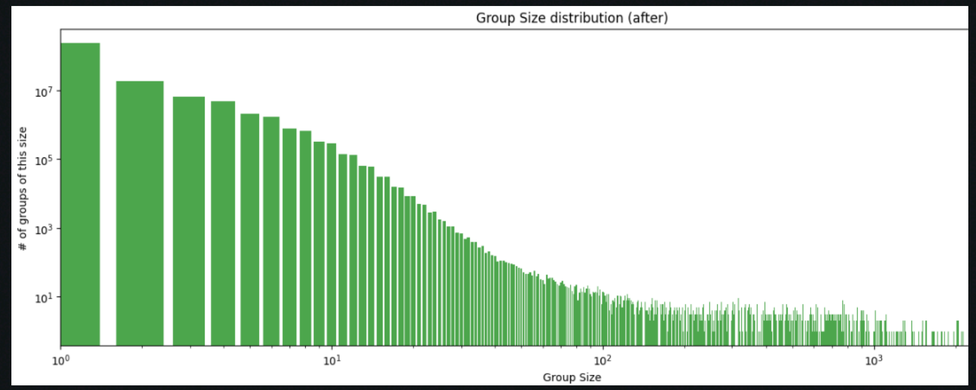
Requires labelling

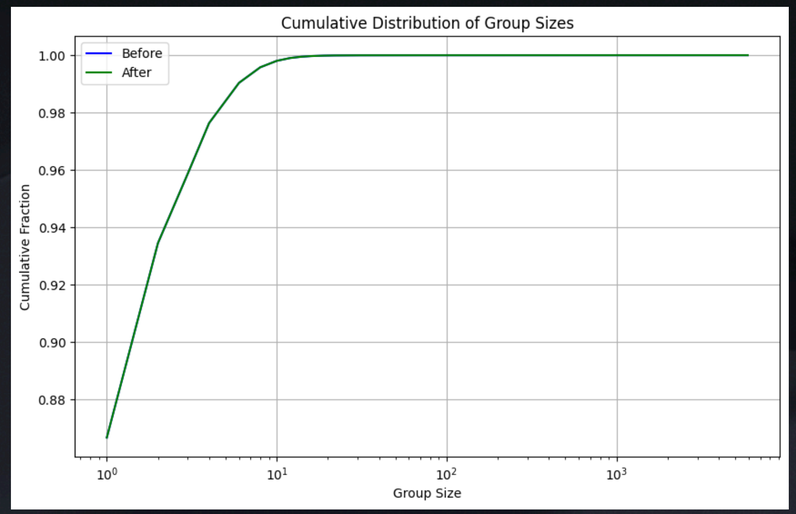
Detailed metrics (recall, precision)

Start from a sample and extrapolate

(Monitor) Cluster Stats

- Mean / Median cluster size
- Histogram
- Cumulative chart
- Matching rate
- Diversity Index: Hill Numbers





(Monitor) Entropy

In information theory, the **entropy** of a random variable quantifies the average level of uncertainty or information associated with the variable's potential states or possible outcomes. […]

https://en.wikipedia.org/wiki/Entropy_(information_theory)

$$H(X) = -\sum_{x \in X} p(x) \times log(p(x))$$

{Leslie, Leslie, Lesly, Leslie, Leslie, Leslie}

x	count(*x*)	p(*x*)	log(p(*x*))	p log(p)
Leslie	5	0,83	-0,08	-0,07
Lesly	1	0,17	-0,78	-0,13
H(X)				0,20

👀 *Frequentist approach of probabilities!*

{John, Jon, Joon, Johnathan, Jhon, John}

x	count(*x*)	p(*x*)	log(p(*x*))	p log(p)
John	2	0,33	-0,48	-0,16
Jon	1	0,17	-0,78	-0,13
Joon	1	0,17	-0,78	-0,13
Johnathan	1	0,17	-0,78	-0,13
Jhon	1	0,17	-0,78	-0,13
John	1	0,17	-0,78	-0,13
H(X)				0,81

(Monitor) Silhouette

Silhouette is a method of interpretation and validation of consistency within clusters of data. The technique provides a succinct graphical representation of how well each object has been classified.[1] […]
The silhouette value is a measure of how similar an object is to its own cluster (cohesion) compared to other clusters (separation).

https://en.wikipedia.org/wiki/Silhouette_(clustering)

Compute the average distance of an entity i to all other entities in its cluster: $a(i)$

Compute the minimum distance of an entity i to all other entities in all other clusters: $b(i)$

… Back to $O(N^2)$

We have run canonicalisation!

a(i) => compare to the cluster canonical entity

b(i) => compare to other clusters' canonical entities

(Evaluate) Sample + Extrapolation

https://github.com/OlivierBinette/er-evaluation

Estimating the Performance of Entity Resolution Algorithms: Lessons Learned Through PatentsView.org

Olivier Binette[1,2], Sokhna A York[2], Emma Hickerson[2], Youngsoo Baek[1], Sarvo Madhavan[2], and Christina Jones[2]

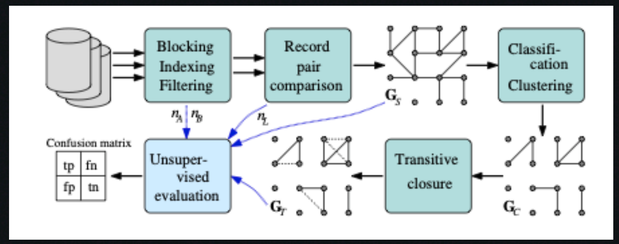
[1]Duke University
[2]American Institutes for Research

April 19, 2023

Abstract

This paper introduces a novel evaluation methodology for entity resolution algorithms. It is motivated by PatentsView.org, a public-use patent data exploration platform that disambiguates patent inventors using an entity resolution algorithm. We provide a data collection methodology and tailored performance estimators that account for sampling biases. Our approach is simple, practical and principled – key characteristics that allow us to paint the first representative picture of PatentsView's disambiguation performance. The results are used to inform PatentsView's users of the reliability of the data and to allow the comparison of competing disambiguation algorithms.

https://arxiv.org/abs/2210.01230



https://dl.acm.org/doi/10.1145/3721985

In Search of an Entity Resolution OASIS: Optimal Asymptotic Sequential Importance Sampling

Neil G. Marchant and Benjamin I. P. Rubinstein
School of Computing and Information Systems
University of Melbourne, Australia
{nmarchant, brubinstein}@unimelb.edu.au

https://arxiv.org/abs/1703.00617

$$F_1 = 2\frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2TP}{2TP + FP + FN}$$

A Look Into The ~~Future~~ Present

FS LIMITATIONS

We wanted to get rid of a deterministic approach,
Did we manage to? ...

...Almost? ... Blocking still relies on deterministic rules

FS Limitations

We have assumed independency between features...

... It may not always be the case

FS LIMITATIONS

We have categorised features comparison into:

- Match
- Close match
- Non-Match

... We have not captured how values are distorted

Blocking Alternatives

Solution 1: K-D Tree

In computer science, a ***k*-d tree** (short for *k-dimensional tree*) is a space-partitioning data structure for organising points in a *k*-dimensional space. […]
k-d trees are a useful data structure for several applications, such as:

- Searches involving a multidimensional search key (e.g. range searches and nearest neighbor searches) &
- […]

k-d trees are a special case of binary space partitioning trees.

id	First Name	Last Name	Post Code
1	Les	Brown	07008
2	Leslie	Brwn	07007
3	Leslie	Brown	07007
4	John	Williamsson	60615
5	Jon	Williamson	60007
6	Joseph	Meyer	90004
7	Joe	Meyer	90004
8	Joseph	Meyr	9004
9	Joseph	Meyer	90001

7.2. Blocking Alternatives

Solution 2: Locality Sensitive Hashing

In computer science, **locality-sensitive hashing** (**LSH**) is a fuzzy hashing technique that hashes similar input items into the same "buckets" with high probability.[1] […]
Since similar items end up in the same buckets, this technique can be used for data clustering and nearest neighbor search. […]
The technique can be seen as a way to reduce the dimensionality of high-dimensional data;
high-dimensional input items can be reduced to low-dimensional versions while preserving relative distances between items.

Example: Min-Hash LSH

LESLIE, LESLY, LES, LESLIE, JOHN, JOHN, JON, JOON

Shingling + Set

{LE, ES, SL, LI, IE, LY, JO, OH, HN, ON, OO}

Shuffle + Hash

HN	JO	SL	ON	LY	IE	ES	OH	LE	LI	OO	
0	1	2	3	4	5	6	7	8	9	10	i
1	3	5	7	9	11	13	15	17	19	1	h1(i) = (2i+1) mod 20
7	10	13	16	19	2	5	8	11	14	17	h2(i) = (3x+7) mod 20
11	16	1	6	11	16	1	6	11	16	1	h3(i) = (5x+11) mod 20

Min-Hash

LESLIE = LE, ES, SL, LI, IE

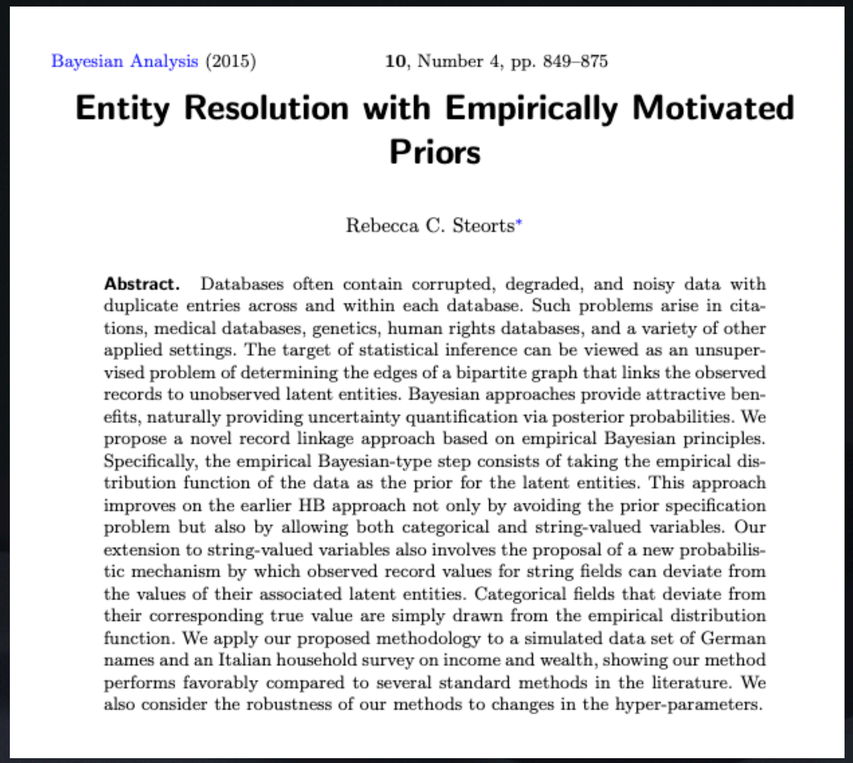
min(h1) = min(17, 13, 05, 19, 11) = 05
min(h2) = min(11, 05, 13, 14, 02) = 02
min(h3) = min(11, 01, 01, 16, 16) = 01

Name	band1	band2	band3
Leslie	5	2	1
Lesly	5	5	1
Les	13	5	1
John	1	7	6
Jon	3	10	6
Joon	1	10	1

Same group if sharing 2+ bands

ALTERNATIVE 1: (D-)BLINK

https://github.com/cleanzr/dblink



Bayesian Analysis (2015) **10**, Number 4, pp. 849–875

Entity Resolution with Empirically Motivated Priors

Rebecca C. Steorts*

Abstract. Databases often contain corrupted, degraded, and noisy data with duplicate entries across and within each database. Such problems arise in citations, medical databases, genetics, human rights databases, and a variety of other applied settings. The target of statistical inference can be viewed as an unsupervised problem of determining the edges of a bipartite graph that links the observed records to unobserved latent entities. Bayesian approaches provide attractive benefits, naturally providing uncertainty quantification via posterior probabilities. We propose a novel record linkage approach based on empirical Bayesian principles. Specifically, the empirical Bayesian-type step consists of taking the empirical distribution function of the data as the prior for the latent entities. This approach improves on the earlier HB approach not only by avoiding the prior specification problem but also by allowing both categorical and string-valued variables. Our extension to string-valued variables also involves the proposal of a new probabilistic mechanism by which observed record values for string fields can deviate from the values of their associated latent entities. Categorical fields that deviate from their corresponding true value are simply drawn from the empirical distribution function. We apply our proposed methodology to a simulated data set of German names and an Italian household survey on income and wealth, showing our method performs favorably compared to several standard methods in the literature. We also consider the robustness of our methods to changes in the hyper-parameters.

Entity Resolution with Empirically Motivated Priors (on ProjectEuclid.org)

d-blink: Distributed End-to-End Bayesian Entity Resolution

Neil G. Marchant[a] Andee Kaplan[b] Daniel N. Elazar[c]
Benjamin I. P. Rubinstein[a] Rebecca C. Steorts[d]

[a]School of Computing and Information Systems, University of Melbourne
[b]Department of Statistics, Colorado State University
[c]Methodology Division, Australian Bureau of Statistics
[d]Department of Statistical Science and Computer Science, Duke University
Principal Mathematical Statistician, United States Census Bureau
DRB #: CBDRB-FY20-309

September 23, 2020

Abstract

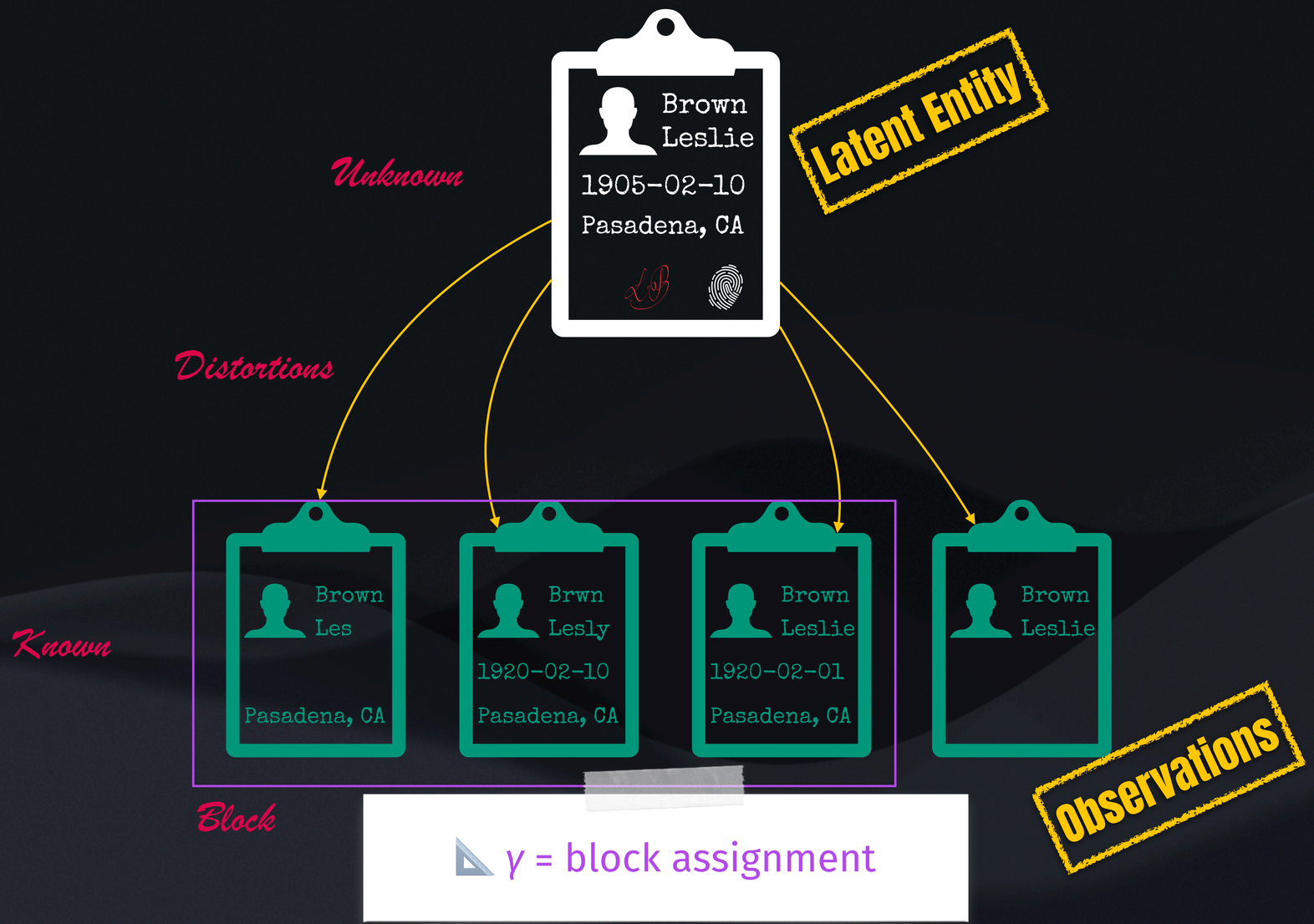
Entity resolution (ER; also known as record linkage or de-duplication) is the process of merging noisy databases, often in the absence of unique identifiers. A major advancement in ER methodology has been the application of Bayesian generative models, which provide a natural framework for inferring latent entities with rigorous quantification of uncertainty. Despite these advantages, existing models are severely limited in practice, as standard inference algorithms scale quadratically in the number of records. While scaling can be managed by fitting the model on separate blocks of the data, such a naïve approach may induce significant error in the posterior. In this paper, we propose a principled model for scalable Bayesian ER, called "distributed Bayesian linkage" or `d-blink`, which jointly performs blocking and ER without compromising posterior correctness. Our approach relies on several key ideas, including: (i) an auxiliary variable representation that induces a partition of the entities and records into blocks; (ii) a method for constructing well-balanced blocks based on k-d trees; (iii) a distributed partially-collapsed Gibbs sampler with improved mixing; and (iv) fast algorithms for performing Gibbs updates. Empirical studies on six data sets—including a case study on the 2010 Decennial Census—demonstrate the scalability and effectiveness of our approach.

Keywords: auxiliary variable, distributed computing, Markov chain Monte Carlo, partially-collapsed Gibbs sampling, record linkage

https://arxiv.org/abs/1909.06039

7.2. BLINK / DBLINK

FOUNDATIONAL IDEAS



Y_{ij} = variables of the latent, true, unobserved entity Y_i

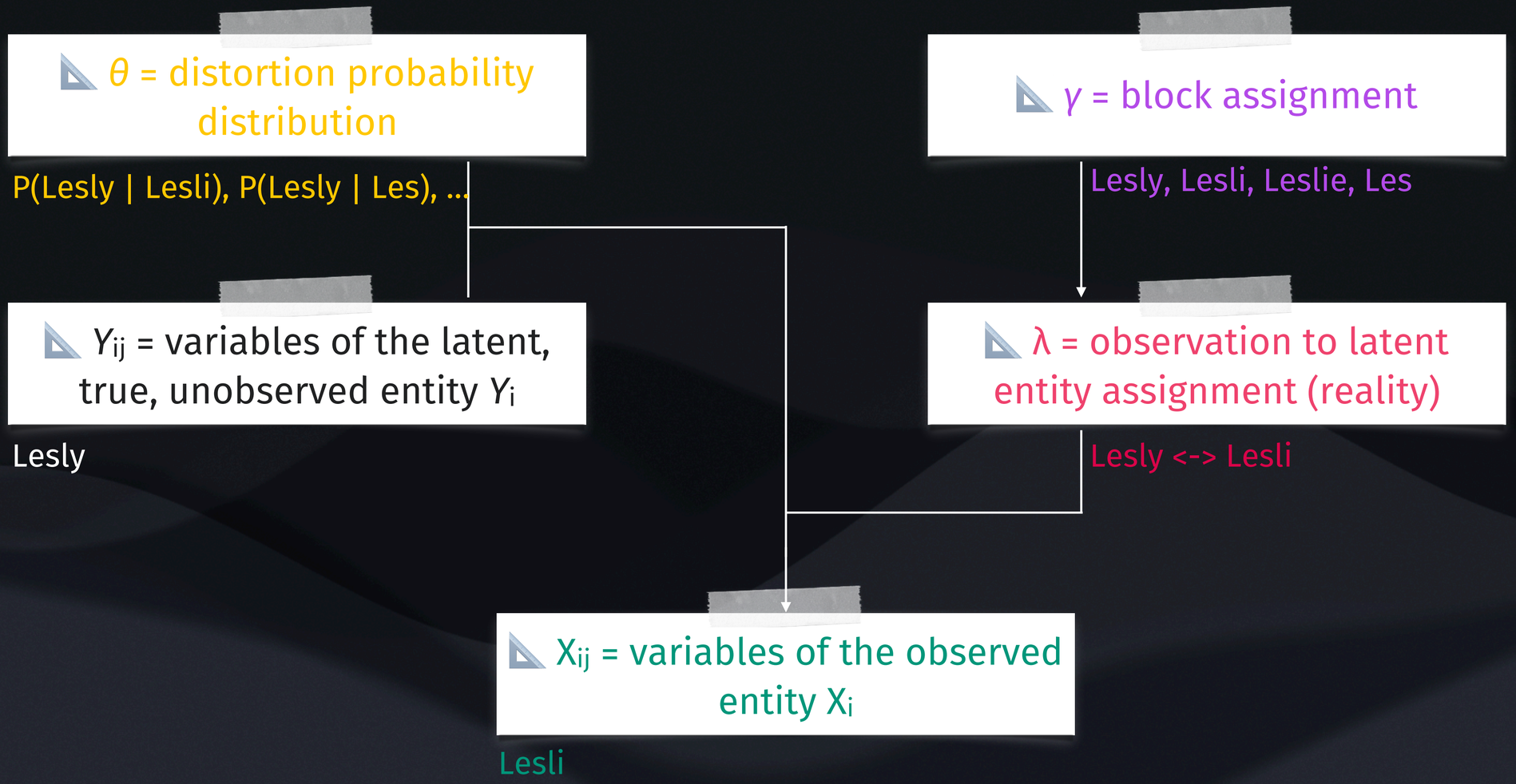
θ = distortion probability distribution

X_{ij} = variables of the observed entity X_i

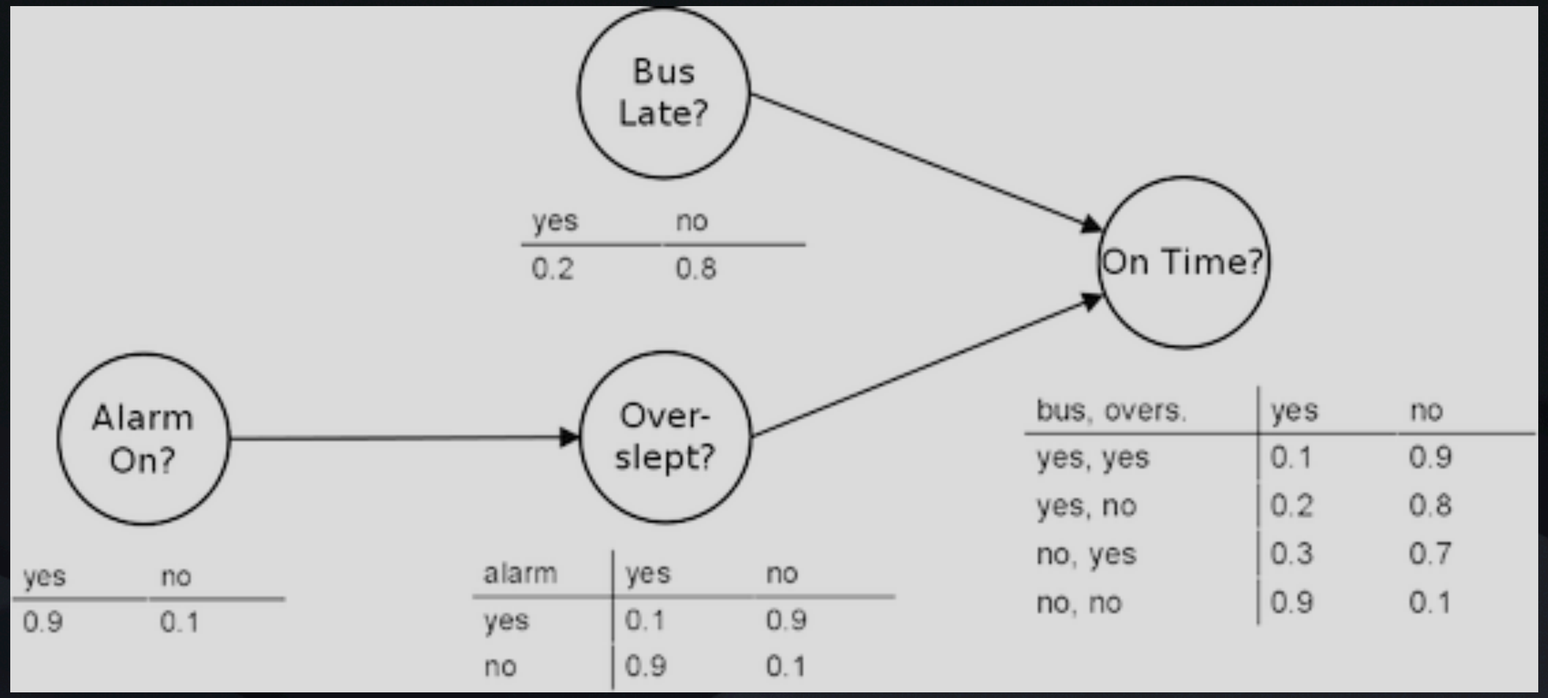
λ = observation to latent entity assignment (reality)

γ = block assignment

Different variables, interconnected



Bayesian Network



Credits: Pekka Parviainen University of Bergen

Just slightly more complex

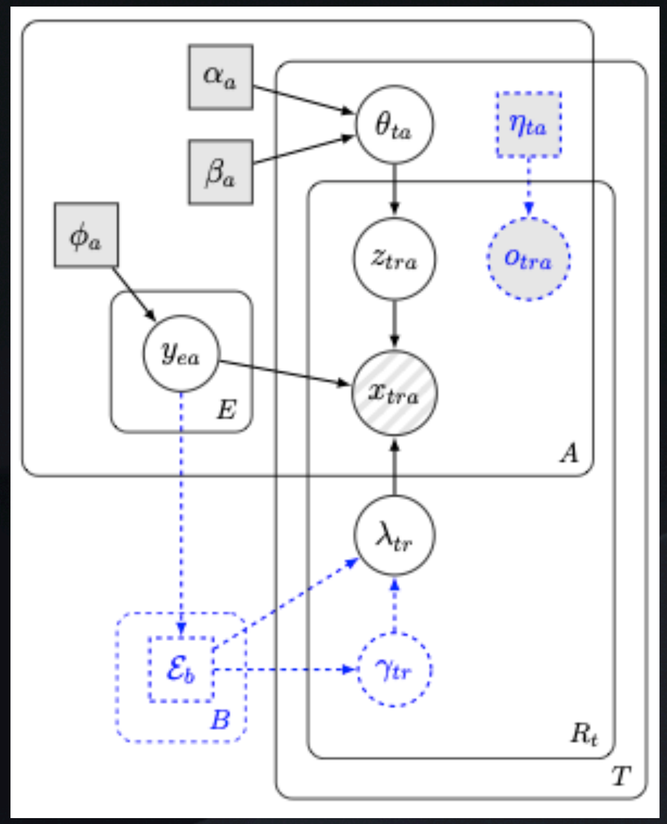
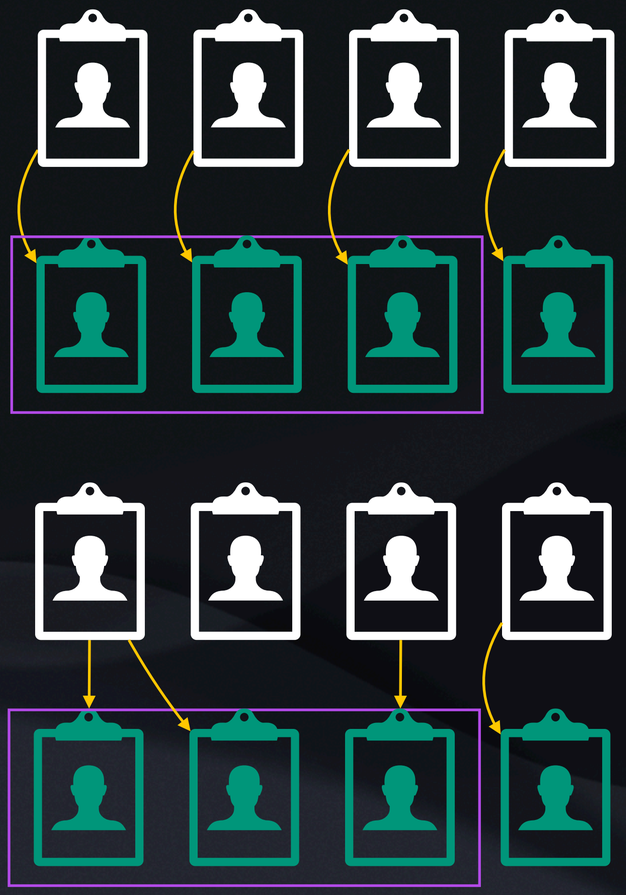


Figure 1: Plate diagram for d-blink. Extensions to blink are highlighted in a dashed blue line style. Circular nodes represent random variables; square nodes represent deterministic variables; (un)shaded nodes represent (un)observed variables; arrows represent conditional dependence; and plates represent replication over an index.

Source: d-blink Distributed End-to-End Bayesian Entity Resolution

7.2. BLINK / DBLINK

THE METHOD

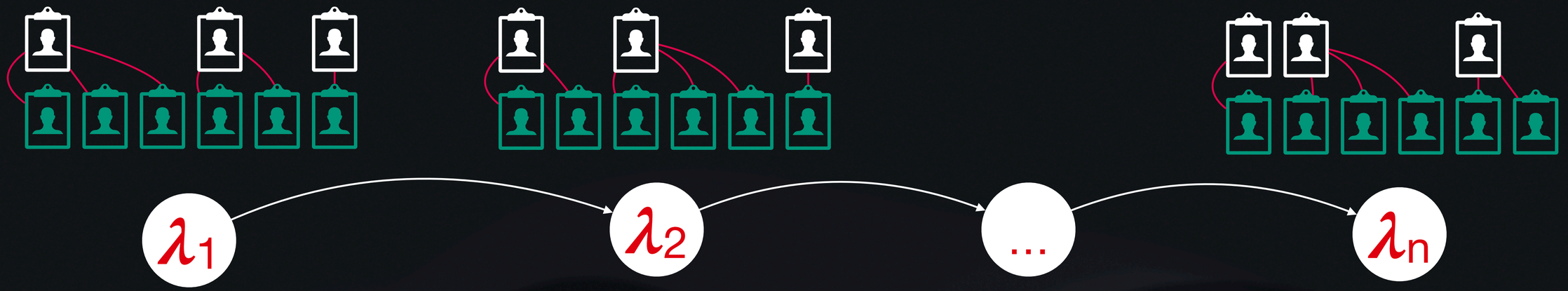


1. Start by assigning every observation a latent entity
   That's our initial λ
2. Pick a random observation,
   Compare it to other latent entities within its block
   -> how likely is it to have been distorted from it?
3. Should it be linked to another entity?
   Or stay with the one created?
   => we change the assignment
   => If merged, adjust the latent entity

   💡 Bayesian Inference
4. Pick another record

   💡 Dirichlet Prior
5. Rinse & Repeat until λ is stable
6. Every few iterations, change blocks γ a bit

EXPLORING LINKAGES



In probability theory and statistics, a **Markov chain** or **Markov process** is a stochastic process describing a sequence of possible events in which the probability of each event depends only on the state attained in the previous event.

WALKING (NOT SO) RANDOMLY

But there are so many possible states to explore? How do I move forward?

Take a small step in a random direction, then look if it brings you closer to destination

Sure???

Yes.
That is called Markov Chain Monte Carlo

And in more details?

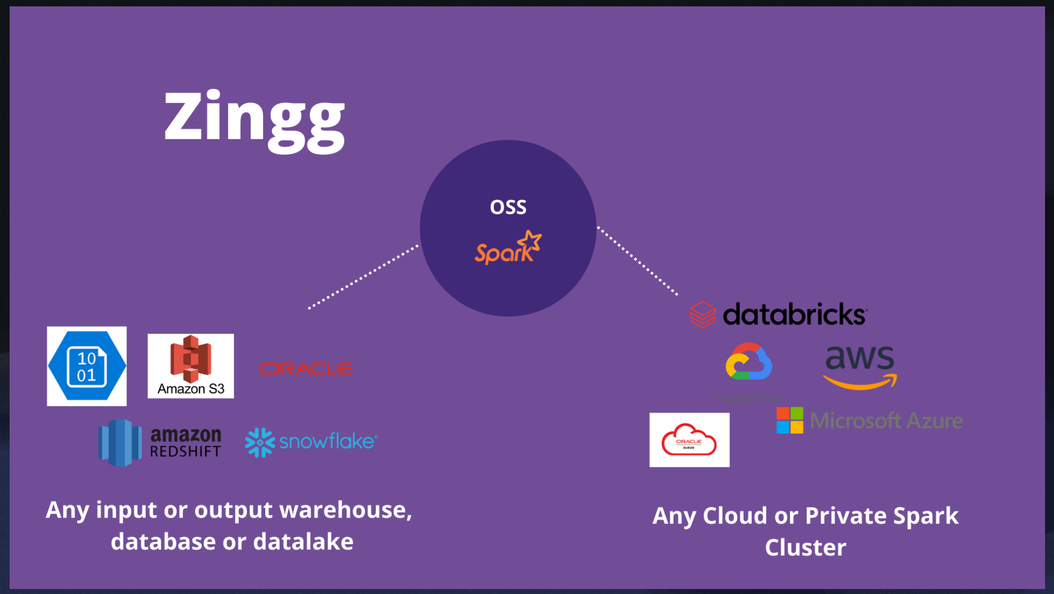
See how we only focused on changing the assignments, all other variables aside?

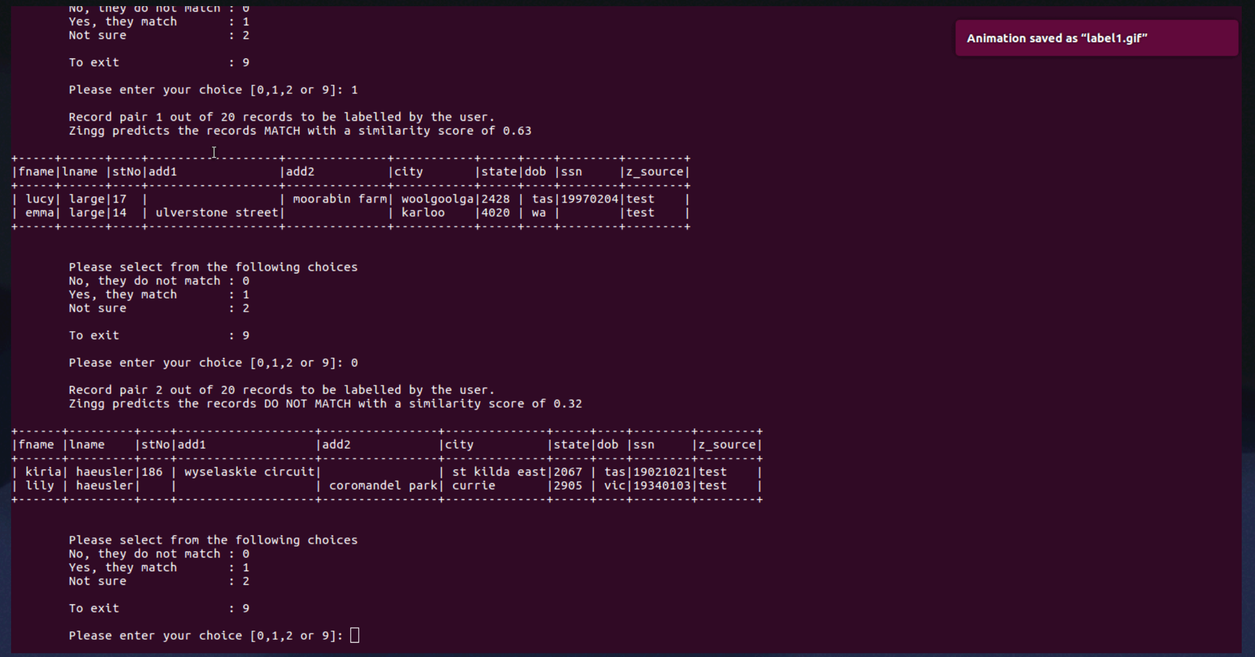
Yes

We have extracted one variable at a time, from a very complex probability distribution : this is Gibbs Sampling

ALTERNATIVE 2: ZINGG.AI

https://github.com/zinggAI/zingg





Did you say Human?

- Machine Learning based
  - “Human in the loop” == Active Learning
  - Carefully chosen edge pairs
- 2 models
  - Blocking Model ~= 1% of the data
  - Similarity Model
- Open Source but very limited (Enterprise Edition)

ZINGG BLOCKING MODEL

- Learned from human labelling
  - *“Multi-field hashing”*
  - *“Do the first 2 characters mostly match”*
  - *“We build a tree out of it”*
  - *“A Block function learned from the data”*

ZINGG SIMILARITY MODEL

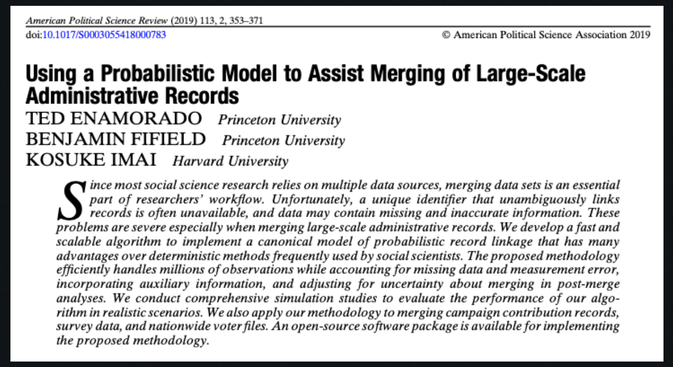
- Uses logistic regression
- Can work with embeddings
- “Match-Type” concept (FUZZY, EMAIL, PIN, …)
  - Translates into features
- Classifier model

Data Scientist takeaways

Cleansing	*Part of my work is to represent the data differently*
Linking	*Using maths: distance metric, (bayesian) probabilities*
Clustering	*By being probabilistic, we can leverage more clustering methods*
Canonicalisation	*Sometimes, heuristics are fine! We can go further though*
Validation	*Look into the data distribution! Then use stats*


Levenshtein
Jaro-Winkler
Monge-Elkan
Edit Distance
Cosine Similarity
Similarity Functions
Jaccard Index
GROUP BY
Phonetic Encoding
Rule-Based Grouping
Deterministic
Token-Based
Deterministic (Key-Based)
LSH
Blocking
Graph
Meta-Blocking
Linking
Cond...nal Random Fields
Fellegi & Sunter
Splink
FastLink
Probabilistic / Stochastic
Bayesian Networks
CPT
Parameter Learning
Structure Learning
MCMC
DBlink
Exchanger
esian Networks_ A C...
Document · 337 KB
Merge-Center Clustering
Connected Components
Center Clustering
Clustering
Density-Based
OPTICS
HDBSCAN
DBSCAN
Hierarchical
Single / Complete / Average Linkage
ingle-linkage clustering
en.wikipedia.org
Correlation Clustering
Correlation clustering
en.wikipedia.org
Similarity-Based
Canopy Clustering
K-Means
LLM
2312
PDF Document · 882 KB
Graph
GNN
ML
Random Forests
Support Vector Machines
Active Learning
Deep Learning
Label Propagation
ITER+ CliqueRank
Markov Random Fields
Knowledge Graph Embeddings
neo4j.com
icde18-entity-resolution
PDF Document · 441 KB
2108
PDF Document · 628 KB
ER_with_Markov_Logic
PDF Document · 146 KB
Zingg
zinggAI/zingg: Scalable ide...
github.com
P16-1059
PDF Document · 251 KB
deepmatcher-sigmod...
PDF Document · 1,7 MB
zhao1701/extending-deep
github.com
1710
PDF Document · 2,2 MB
Dedupe
dedupeio/dedupe: :id: A pytho...
github.com
2211
PDF Document · 1,5 MB

BIBLIOGRAPHY



American Political Science Review (2019) 113, 2, 353–371
doi:10.1017/S0003055418000783

Using a Probabilistic Model to Assist Merging of Large-Scale Administrative Records

TED ENAMORADO *Princeton University*
BENJAMIN FIFIELD *Princeton University*
KOSUKE IMAI *Harvard University*

Since most social science research relies on multiple data sources, merging data sets is an essential part of researchers' workflow. Unfortunately, a unique identifier that unambiguously links records is often unavailable, and data may contain missing and inaccurate information. These problems are severe especially when merging large-scale administrative records. We develop a fast and scalable algorithm to implement a canonical model of probabilistic record linkage that has many advantages over deterministic methods frequently used by social scientists. The proposed methodology efficiently handles millions of observations while accounting for missing data and measurement error, incorporating auxiliary information, and adjusting for uncertainty about merging in post-merge analyses. We conduct comprehensive simulation studies to evaluate the performance of our algorithm in realistic scenarios. We also apply our methodology to merging campaign contribution records, survey data, and nationwide voter files. An open-source software package is available for implementing the proposed methodology.

Parameters Estimation:
https://imai.fas.harvard.edu/research/files/linkage.pdf