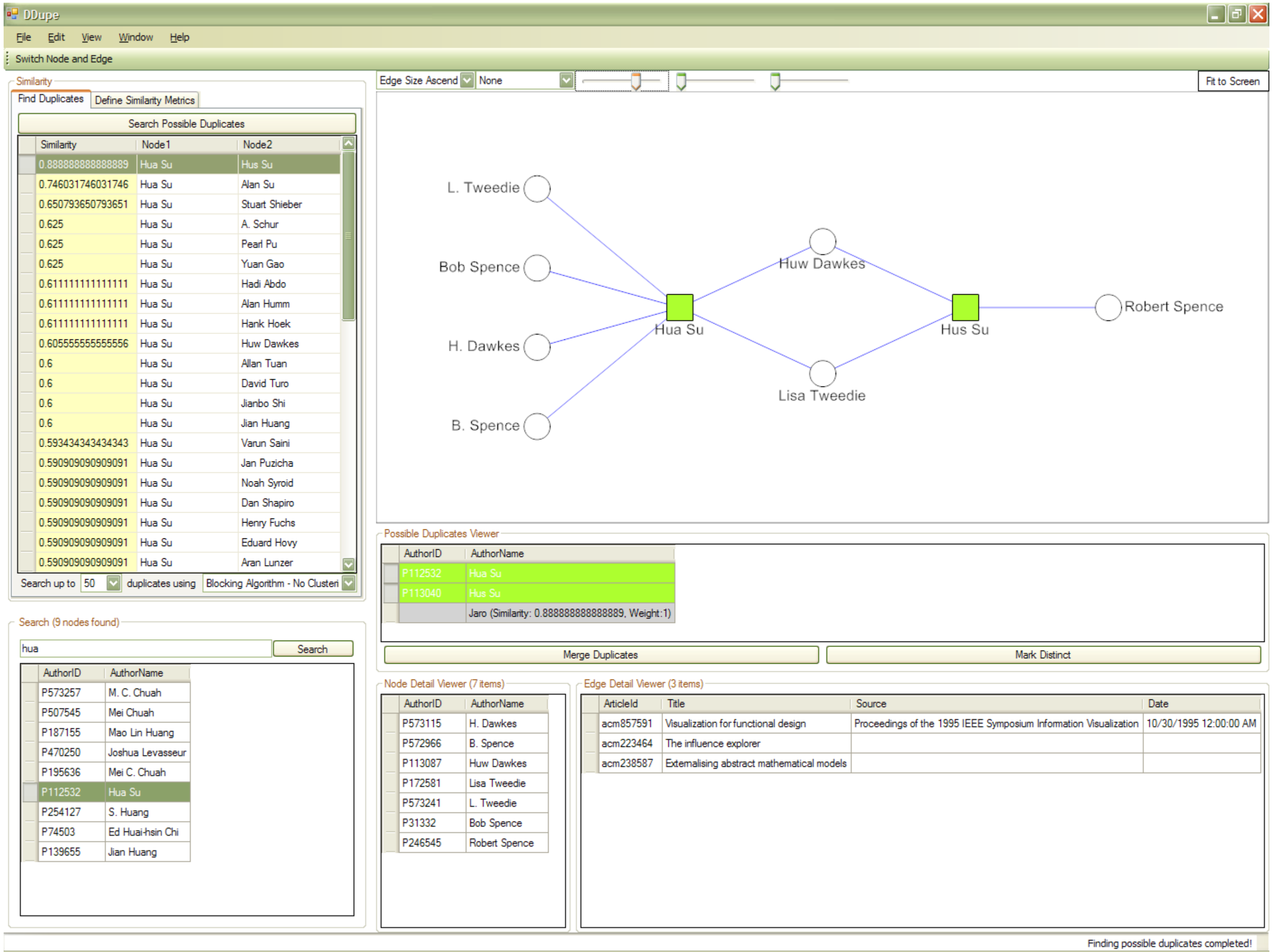
DEMO

D-Dupe: An Interactive Tool for ER



Kang, Getoor, Shneiderman, Bilgic, Licamele, TCGV 08

http://www.cs.umd.edu/projects/linqs/ddupe

Part 5

CHALLENGES AND FUTURE DIRECTIONS

Outline

- Distributed ER
- Training Set Generation & Active ER
- Query Time ER
- Temporal ER

PART 5-a

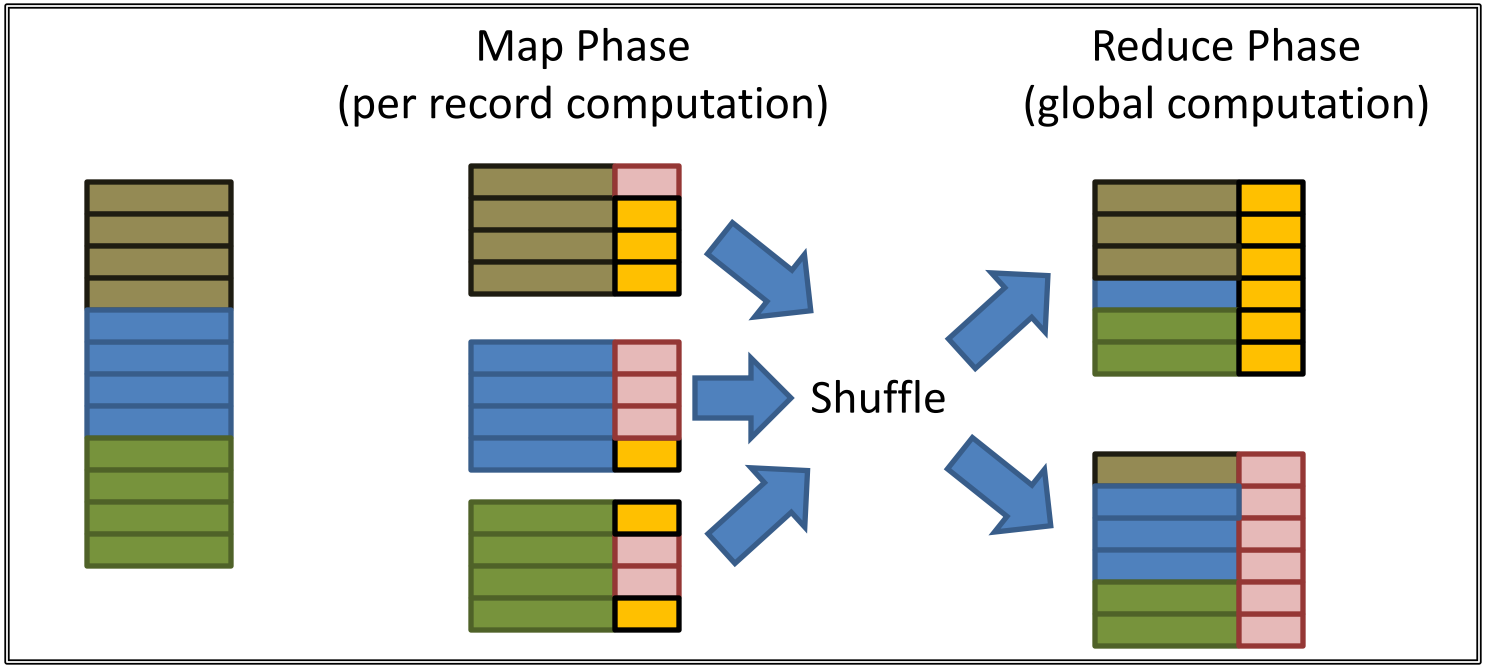
DISTRIBUTED ER

Distributed ER

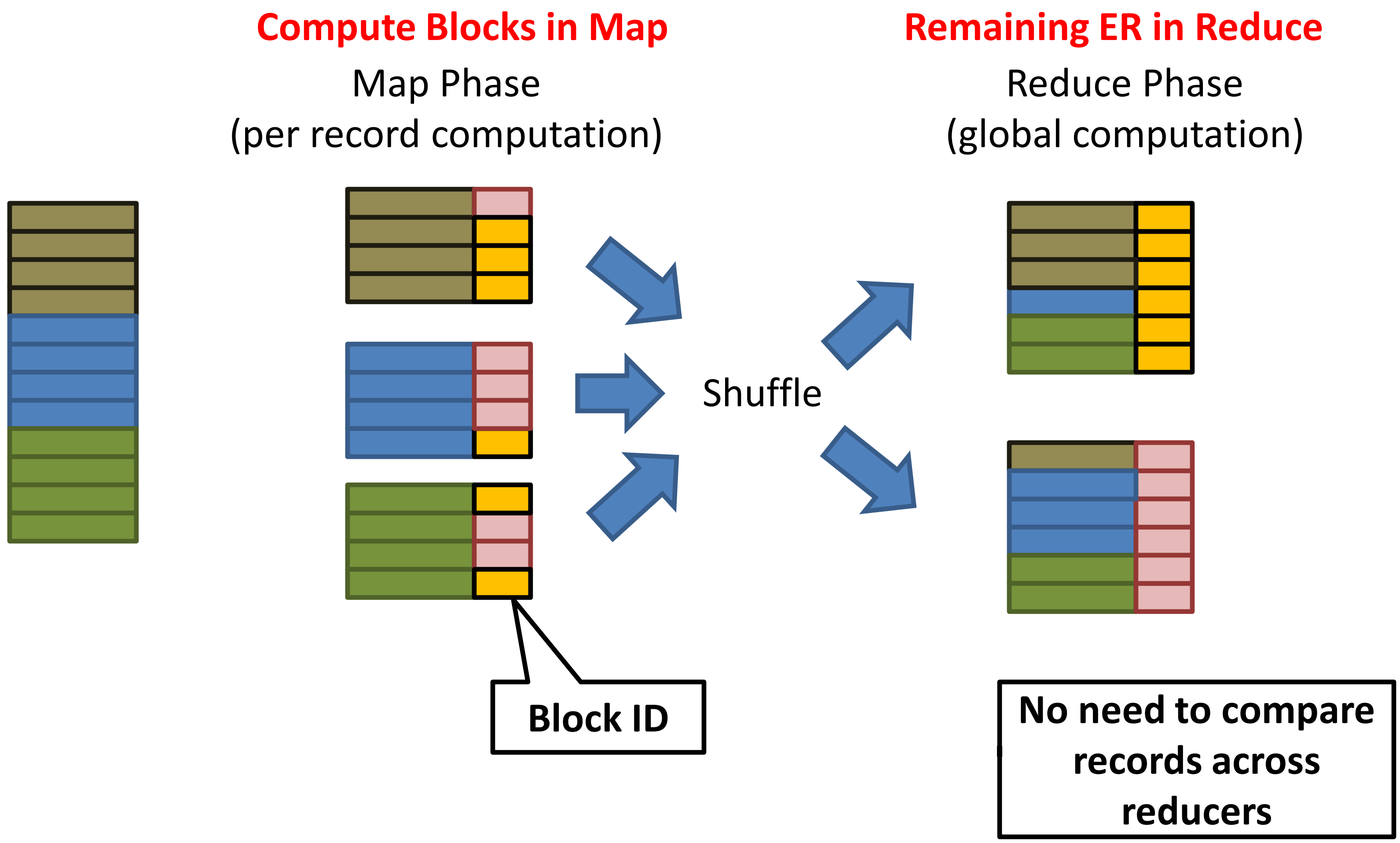
- Map-reduce is very popular for large tasks
  - Simple programming model for massively distributed data

```
map    (k1,v1)        → list(k2,v2);
reduce (k2,list(v2))  → list(k3,v3).
```

  - Hadoop provides fault tolerance and is open source



ER with Disjoint Blocking



Non-disjoint Blocking

- How to block?
  - Hash-based: need an efficient technique to group records if they match on *l-out-of-k* blocking keys [Vernica et al SIGMOD'10]
  - Similarity-based: clustering on map-reduce [Mahout]

- Information needed for a record is in multiple reducers.
  - Problem:
    - Reducer 1: "a" matches with "b"
    - Reducer 2: "a" matches with "c"
    - Need to communicate in order to correctly resolve "a", "b", "c"
  - Solution 1: Efficient Transitive Closure [Machanavajjhala et al 2012]
    + Correlation Clustering
  - Solution 2: Message Passing [Rastogi et al VLDB'11]

DISTRIBUTED COLLECTIVE ER

Scalability [Rastogi et al VLDB11]

Current state-of-the-art: **Collective Entity Matching**

(+) High ***accuracy***

(-) Often scale only to a few 1000 entities$_{\text{[SD06]}}$

How can we scale

Collective Entity Matching

to millions of entities?

Slides adapted from [Rastogi et al VLDB11] talk

Scalability [Rastogi et al VLDB11]

Current state-of-the-art: **Collective Entity Matching**

(+) High ***accuracy***

(-) Often scale only to a few 1000 entities$_{[SD06]}$

Our Approach

Id	**Author-1**	**Author-2**	**Paper**
A_1	John Smith	Richard Johnson	Indices and Views
A_2	J Smith	R Johnson	SQL Queries
A_3	Dr. Smyth	R Johnson	Indices and Views

Distribute + Message Passing

Current state-of-the-art: **Collective Entity Matching**

(+) High ***accuracy***

(-) Often scale only to a few 1000 entities[SD06]

Our Approach

Collective Entity Matcher

P_1	Indices and Views	John Smith	Richard Johnson
P_2	Indices & Views	J. Smith	R. Johnson

P_2	Indices & Views	J. Smith	R. Johnson
P_3	Political Views	Jane Smith	R. Johnson

Distribute + Message Passing

Current state-of-the-art: **Collective Entity Matching**

(+) High ***accuracy***

(-) Often scale only to a few 1000 entities$_{[SD06]}$

Our Approach

Collective Entity Matcher	P_1	Indices and Views	John Smith	Richard Johnson
	P_2	Indices & Views	J. Smith	R. Johnson

Messages

Collective Entity Matcher	P_2	Indices & Views	J. Smith	R. Johnson
	P_3	Political Views	Jane Smith	R. Johnson

Distribute + Message Passing

Current state-of-the-art: **Collective Entity Matching**

(+) High ***accuracy***

(-) Often scale only to a few 1000 entities[SD06]

Our Approach

Collective Entity Matcher

P_1	Indices and Views	John Smith	Richard Johnson
P_2	Indices & Views	J. Smith	R. Johnson

Messages

Collective Entity Matcher

P_2	Indices & Views	J. Smith	R. Johnson
P_3	Political Views	Jane Smith	R. Johnson

Distribute + Message Passing

Current state-of-the-art: **Collective Entity Matching**

(+) High ***accuracy***

(-) Scale only to roughly 1000 entities$_{\text{[SD06]}}$

Our Approach

Collective Entity Matcher

P_1	Indices and Views	John Smith	Richard Johnson
P_2	Indices & Views	J. Smith	R. Johnson

Messages

Collective Entity Matcher

P_2	Indices & Views	J. Smith	R. Johnson
P_3	Political Views	Jane Smith	R. Johnson

(+) Formal ***accuracy guarantees*** if entity matcher is ***well-behaved***

(+) ***Scales*** to datasets with millions of entities

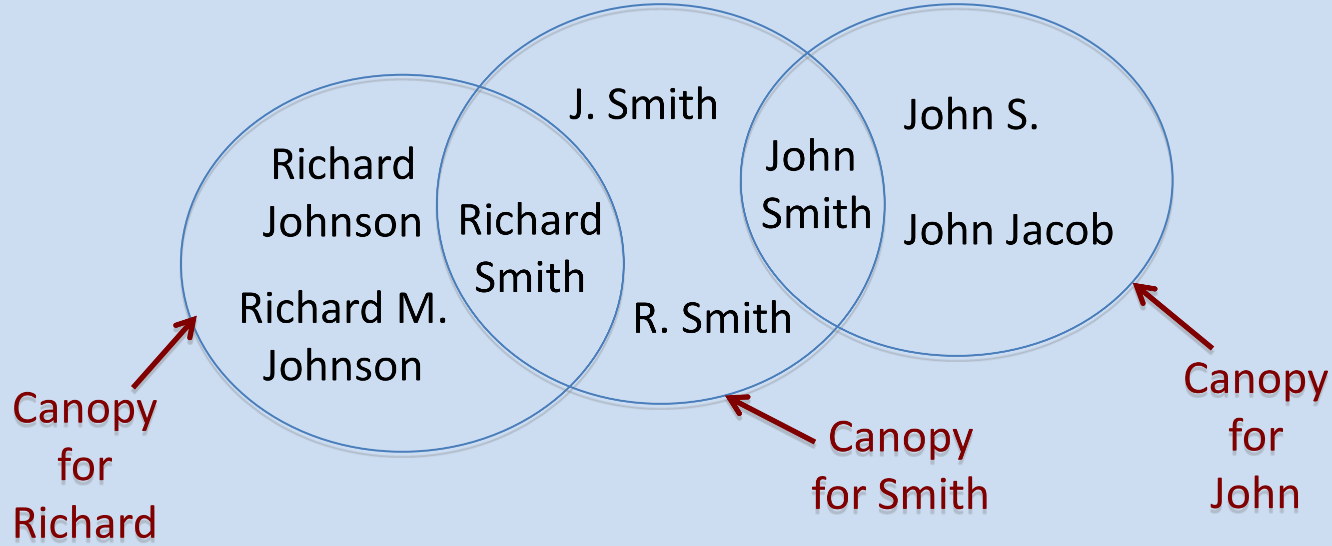
Algorithm

- Generates overlapping canopies (e.g., Canopy clustering)
- Run collective matcher on each canopy

Efficiency: Use Canopies [McCallum et. al.]

Reduces # of candidate pairs from:

$O(|\text{Mentions}|^2)$ to $|\text{Candidate pairs}|$



Pair-wise approach becomes efficient: $O(|\text{Candidate pairs}|)$

Efficiency of Collective approach

Collective methods still not efficient: $\Omega(|\text{Candidate pairs}|^2)$

Example for Collective methods[SD06]

- |References| = 1000, |Candidate pairs| = 15,000,
  - Time ~ 5 minutes
- |References| = 50,000, |Candidate pairs| = 10 million
  - Time required = 2,500 hours ~ 3 months

Distribute

Run collective entity-matching over canopies separately

Example for Collective methods[SD06]

- |References|= 1000, |Candidates| = 15,000,
  - Time = 5 minutes
- One canopy: |References| = 100, |Candidates| ~ 1000,
  - Time ~ 10 Seconds
- |References| = 50,000, # of canopies ~ 13k
  - Time ~ 20 hours << 3 months!

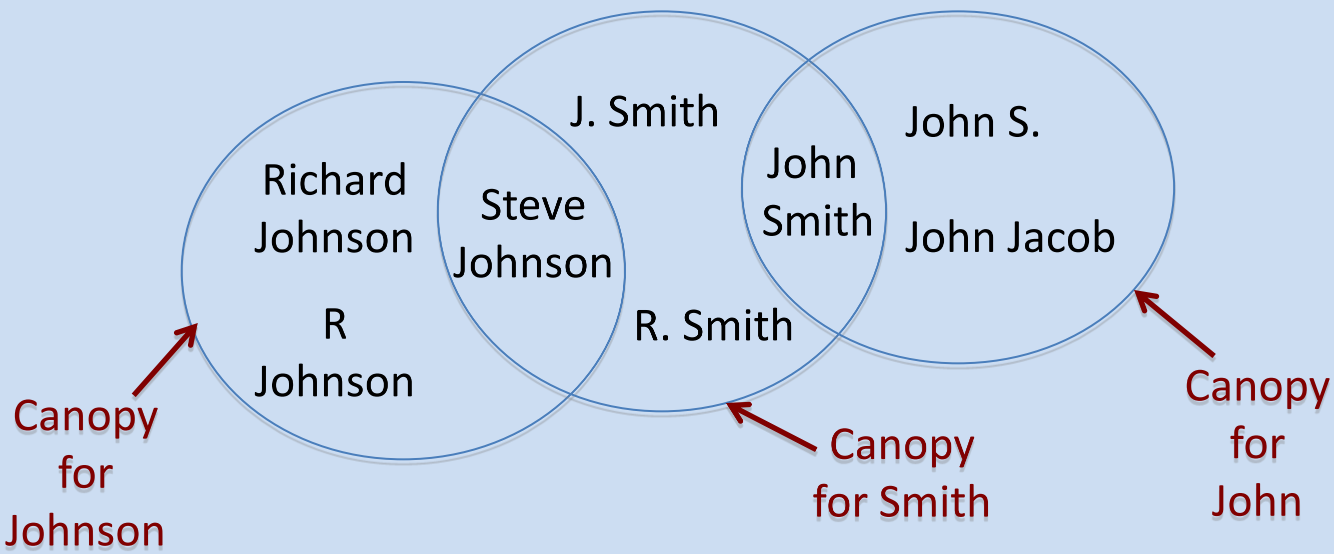
Partitioning into smaller chunks helps!

Problem: Correlations across canopies will be lost

$\text{CoAuthor}(A_1, B_1) \wedge \text{CoAuthor}(A_2, B_2) \wedge \text{match}(B_1, B_2) \rightarrow \text{match}(A_1, A_2)$

Example: CoAuthor rule grounds to the correlation

match(Richard Johnson, R Johnson) => match(J. Smith, John Smith)



Message Passing

Simple Message Passing (SMP)

1. Run entity matcher M locally in each canopy
2. If M finds a $match(r_1,r_2)$ in some canopy, pass it as evidence to all canopies
3. Rerun M within each canopy using new evidence
4. Repeat until no new matches found in each canopy

Runtime: $O(k^2\, f(k)\, c)$

- k : maximum size of a canopy
- f(k): Time taken by ER on canopy of size k
- c : number of canopies

Formal Properties

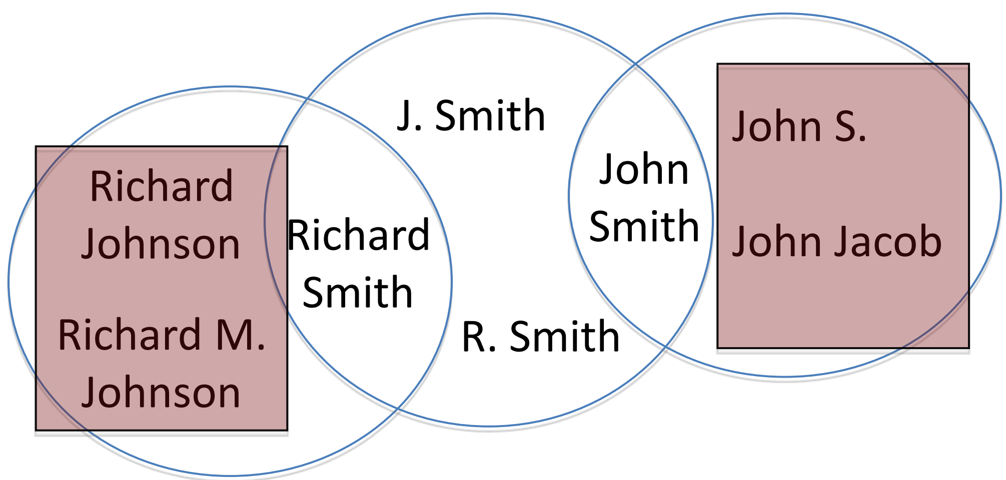
for a well behaved ER method …

Convergence: No. of steps ≤ no. of matches

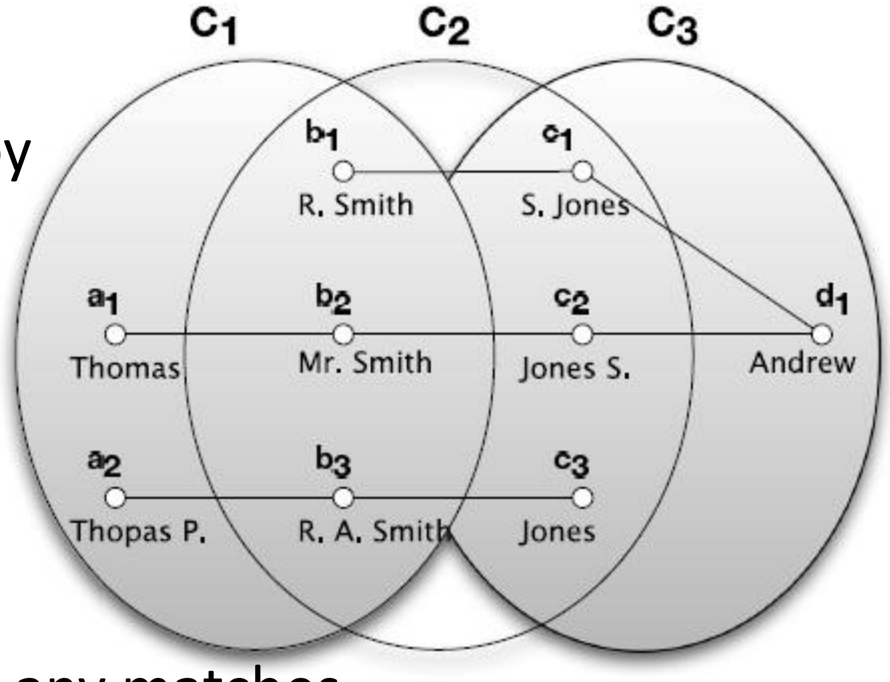
Consistency: Output independent of the canopy order

Soundness: Each output match is actually a true match

~~***Completeness***: Each true match is also a output match~~



Completeness



Papers 2 and 3 match only if a canopy knows that

- match(a1,a2)
- match(b2,b3)
- match(c2,c3)

Simple message passing will not find any matches

- thus, no messages are passed, no progress

Solution: Maximal message passing

- Send a message if there is a potential for match

Challenges in Distributed ER

- Massive linked datasets need distributed ER solution.
  - Some promising solutions exist.
- Is Map-reduce the right abstraction for ER?
  - Suited for batch processing parts of similarity computation.
  - Not suited for graph/iterative aspects of ER
- What are other communication efficient algorithms for collection ER? How can this be extended to general inference on graphical models?

PART 5-b

TRAINING SETS & ACTIVE ER

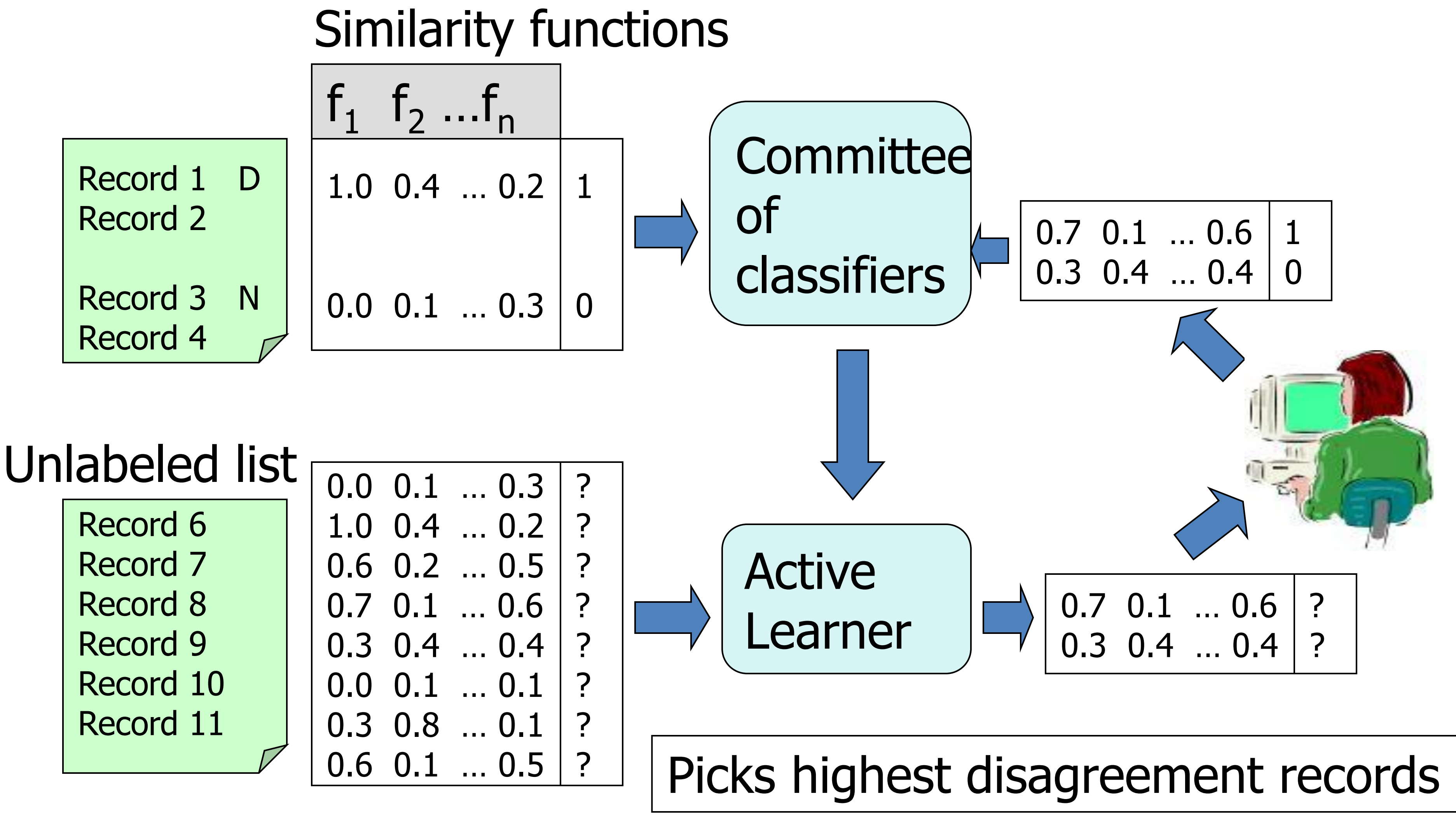
Creating a Training Set is a key issue

- State-of-the-art practical techniques are supervised ML techniques.
  - But they need a training/evaluation dataset.

- Constructing a training set is hard – since most pairs of records are "easy non-matches".
  - 100 records from 100 cities.
  - Only 10^6 pairs out of total 10^8 (1%) come from the same city

- Some pairs are hard to judge even by humans
  - Inherently ambiguous (e.g. Paris Hilton)
  - Missing attributes (Starbucks Toronto, Starbucks Queen Street Toronto)

Active Learning for ER [Sarawagi et al KDD02]



Challenges for Active ER

- Can the supervision be given in terms of rules rather than match/non-match decisions on pairs of records?
- How to construct active learning techniques for collective ER?
- How do we handle errors in human judgements?
  - In an experiment on Amazon Mechanical Turk:
    - Each pairwise judgment given to 5 different people
  - Majority of workers agreed on truth on only 90% of pairwise judgements.

PART 5-c

QUERY TIME ER

Query-time ER

- Many public web services do not have resolved entities
  - PubMed, CiteSeer have unresolved authors
  - Google Places, Yahoo Local, Yelp have unresolved businesses

- Query processing requires resolved entities
  - “Retrieve papers by S. Johnson of Bell Labs”
  - “When the Queen St Metro ”

Query-time ER using Relations

- Possible directions
  1. Leave resolution burden on user
  2. Expect owner to ‘clean’ database

- Collective resolution for queries [Bhattacharya et al KDD06]
  – Extract relevant records by recursive expansion
  – Collective resolution on extracted records

- Challenge: How do we selectively determine the smallest number of records to resolve, so we get accurate results?

PART 5-d

TEMPORAL ER

ER as a dynamic process

- Real world ER systems need to continuously maintain knowledge based
  - Google Places and Yahoo Local get updates to business attributes, and learn about new/closed businesses
  - Affiliations of individuals change over time

- Challenge 1: ER algorithms need to account for “change in real world”

Temporal ER [Pal et al. WWW12]

e.g. a restaurant abc's phone number?

	Current Value
Source A	555-1234
Source B	555-1234
Source C	555-4444

Corroboration by majority

Temporal ER [Pal et al WWW 12]

e.g. a restaurant <u>abc</u>'s phone number?

	Current Value	Last Month	2 month's back
Source A ✗	555-1234	555-1234	555-8566
Source B ✗	555-1234	555-1234	555-8566
Source C ✓	555-4444	555-1234	555-8566

Source C seems correct because:

- *C gives the correct answer historically.*
- *A, B might be lagging in their view.*

Temporal ER

- ER for authors with changing affiliations [Dong et al VLDB11]
  - Affiliation transitions are smooth
    - Other attributes like coauthors does not change dramatically as well
  - Changes are not erratic
    - One does not change affiliations (or switch back and forth) often.

ER as a dynamic process

- Knowledge bases are created by deduplicated many different sources.
  - Google/Yahoo are built on feeds map and business data providers

- These sources themselves may be a result of deduplication, or copying from another source.

- Challenge 2: Sources are not “independent”
  - Need to account for this when creating canonical values
  - Need to account for wrong input records resulting from wrong deduplications.

Copying Problem [Dong et al VLDB09]

- Copying can affect canonicalization.

	S1	S2	S3	S3 copy1	S3 copy2
Stonebraker	MIT	Berkeley	MIT	MIT	MS
Dewitt	MSR	MSR	UWisc	UWisc	UWisc
Bernstein	MSR	MSR	MSR	MSR	MSR
Carey	UCI	AT&T	BEA	BEA	BEA
Halevy	Google	Google	UW	UW	UW

Badly deduped sources as input

- R1: Starbucks, Queens St Toronto, 333-4444
- R2: Tim Hortons, Queens St Toronto, 444-3333
- R3: Starbucks, Queens St Toronto, 444-3333

- R3 provides more “evidence” that R1 and R2 should match.

ER as a dynamic process

- Deduplicated entities interact with users in the real world
  - Users tag/associate photos/reviews with businesses on Google / Yahoo

- However, as the underlying data changes, what should be done to the user-generated data?
  - Suppose ER system realizes that it had incorrectly merged Starbucks and Tim Hortons in one entity.
  - Users added photos and reviews to this entity.
  - Now if ER system realizes its mistake, how to reassign the photos and reviews correctly to the two new entities?

Summary

- Growing omnipresence of massive linked data, and the need for creating knowledge bases from text and unstructured data motivate a number of challenges in ER
- As data, noise, and knowledge grows, greater needs & opportunities for intelligent reasoning about enitity resolution
- Many other challenges
  - Privacy-aware record linkage
  - Large scale identity management
  - Understanding theoretical potentials & limits of ER

THANK YOU!