



IT Systems Engineering | Universität Potsdam

## Collective Entity Resolution

9.7.2013

Felix Naumann

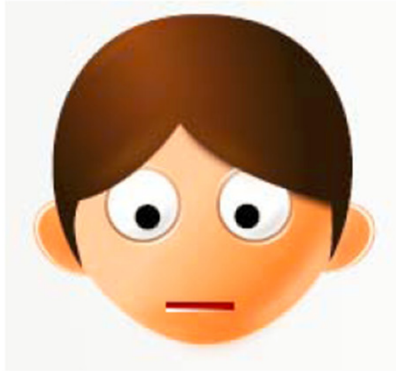
# Main idea

2

- Use relationships in data to enhance deduplication result
  - Within classes
    - ◇ Social networks
    - ◇ Bibliographic references
    - ◇ Web pages
  - Across classes
    - ◇ Books and authors
    - ◇ Movies and actors
    - ◇ Persons and organizations
  
- Aka „joint entity resolution“

# Abstract Problem Statement

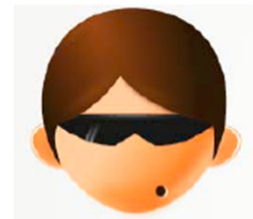
Real World



Digital World

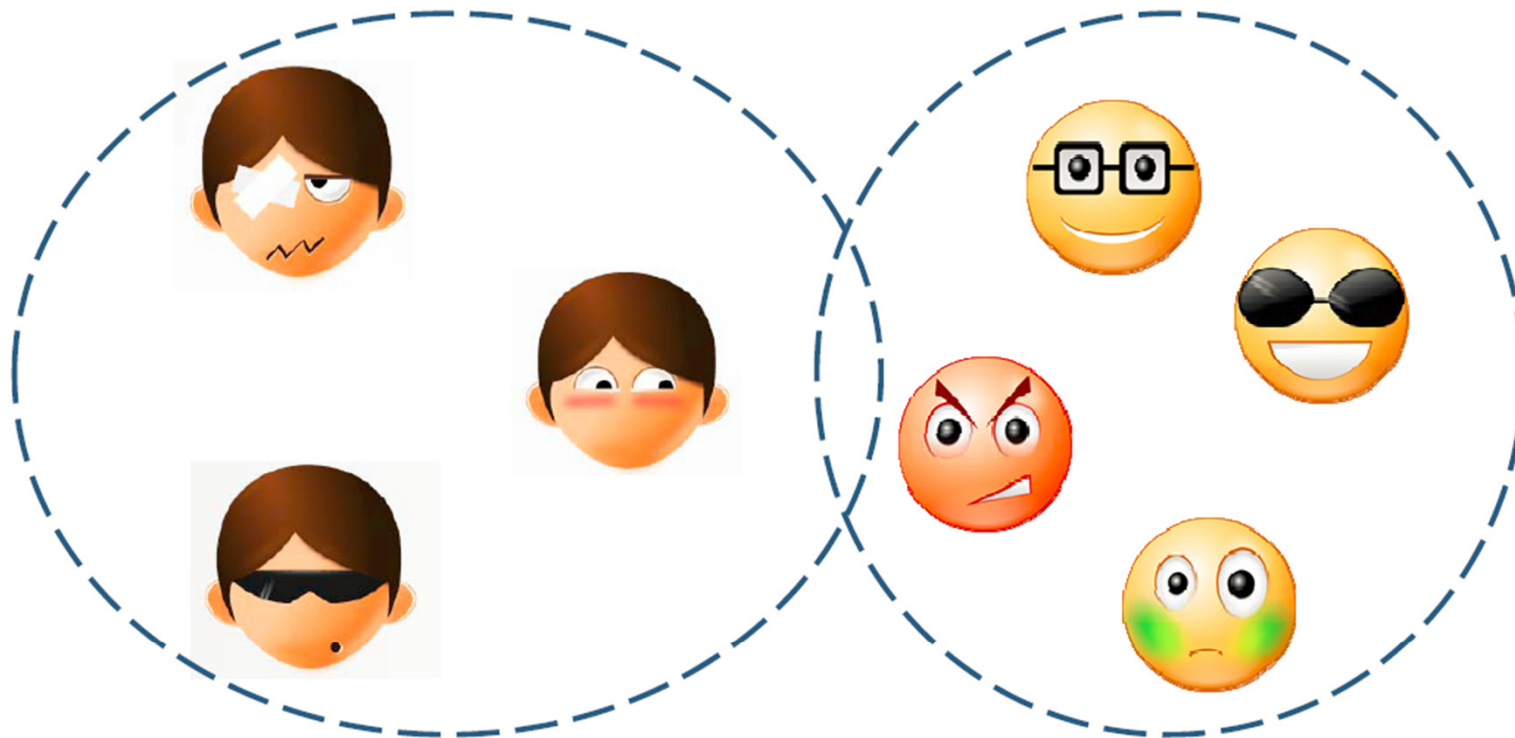


Records /  
Mentions



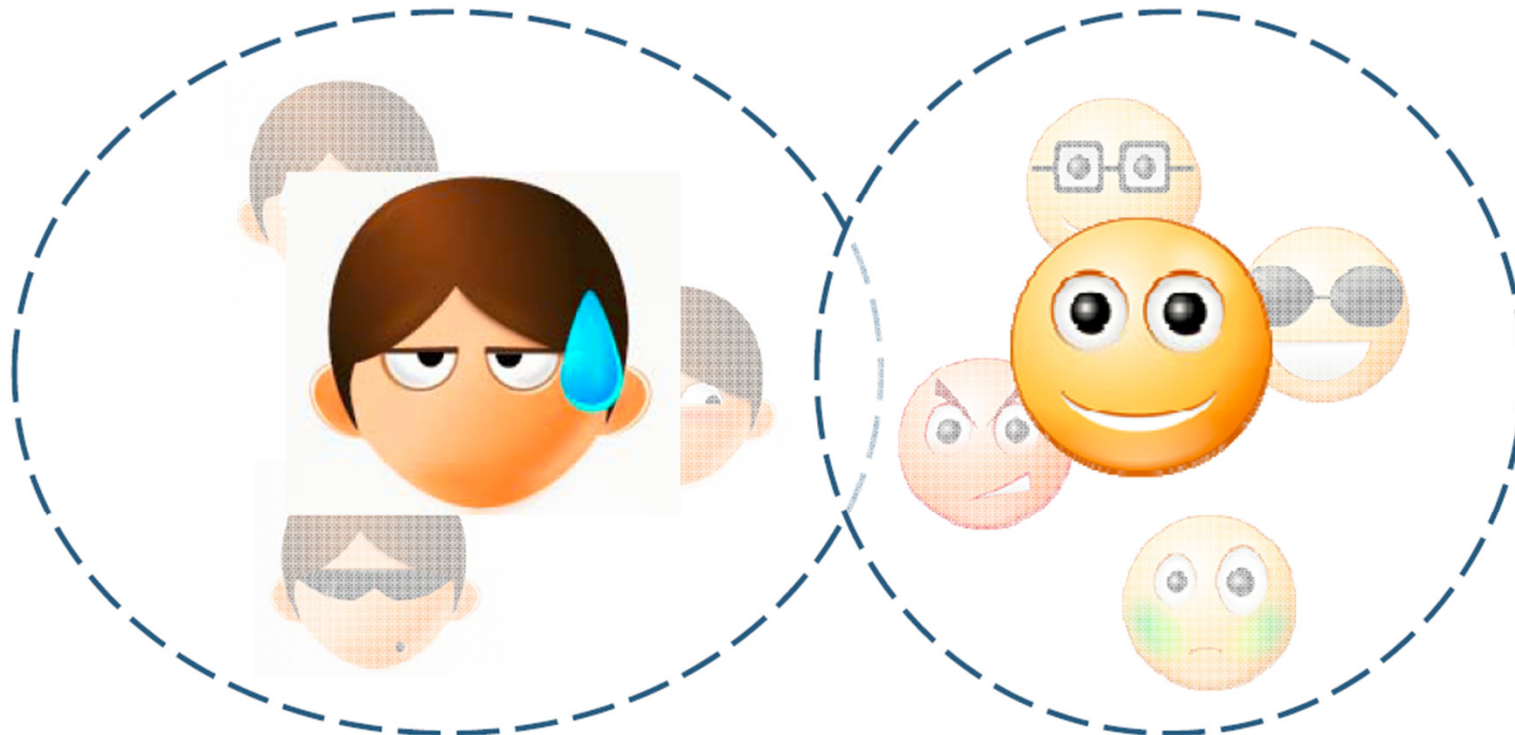
# Deduplication Problem Statement

- Cluster the records/mentions that correspond to same entity



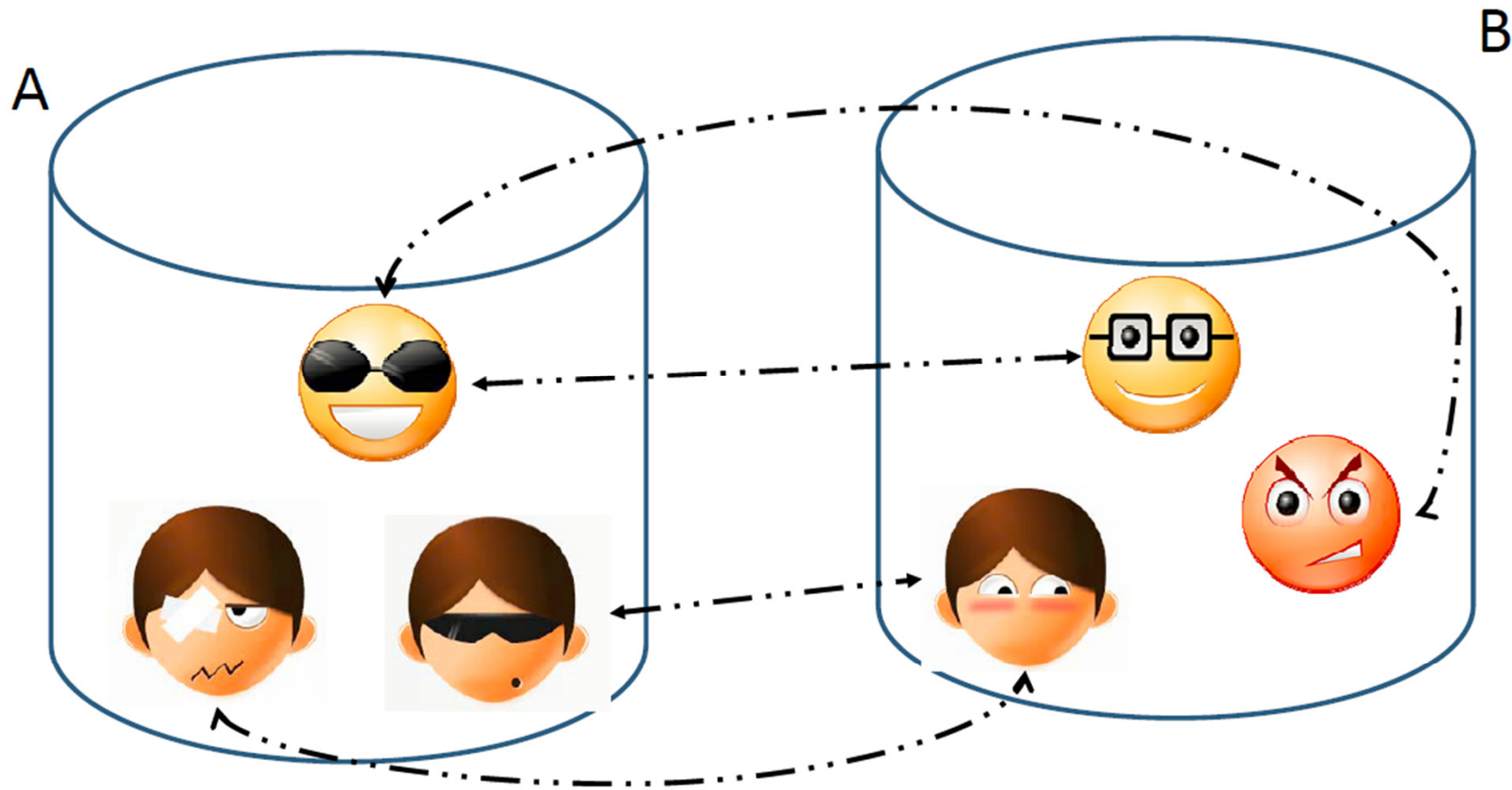
# Deduplication Problem Statement

- Cluster the records/mentions that correspond to same entity
  - **Intensional Variant:** Compute cluster representative



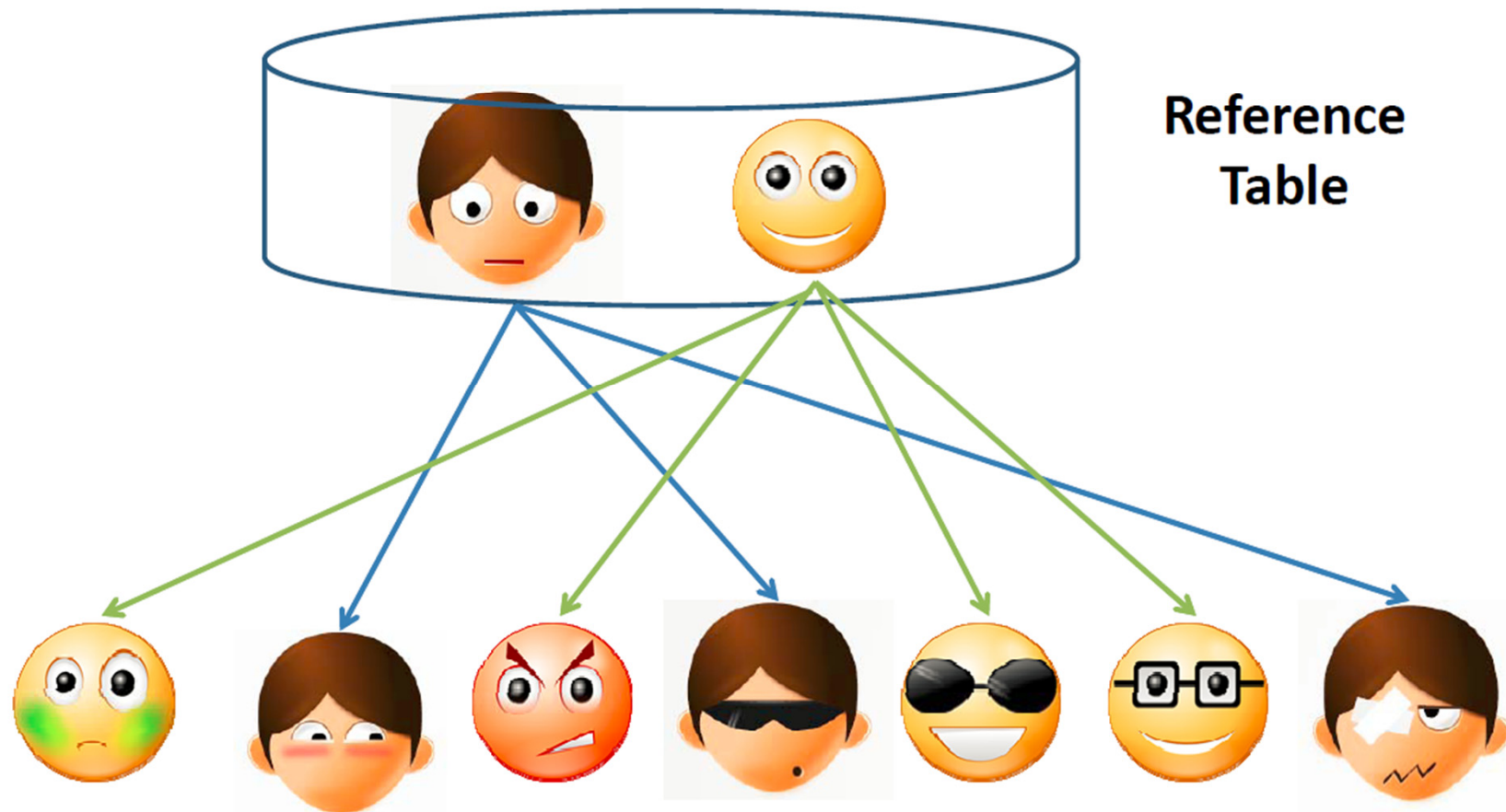
# Record Linkage Problem Statement

- Link records that match across databases



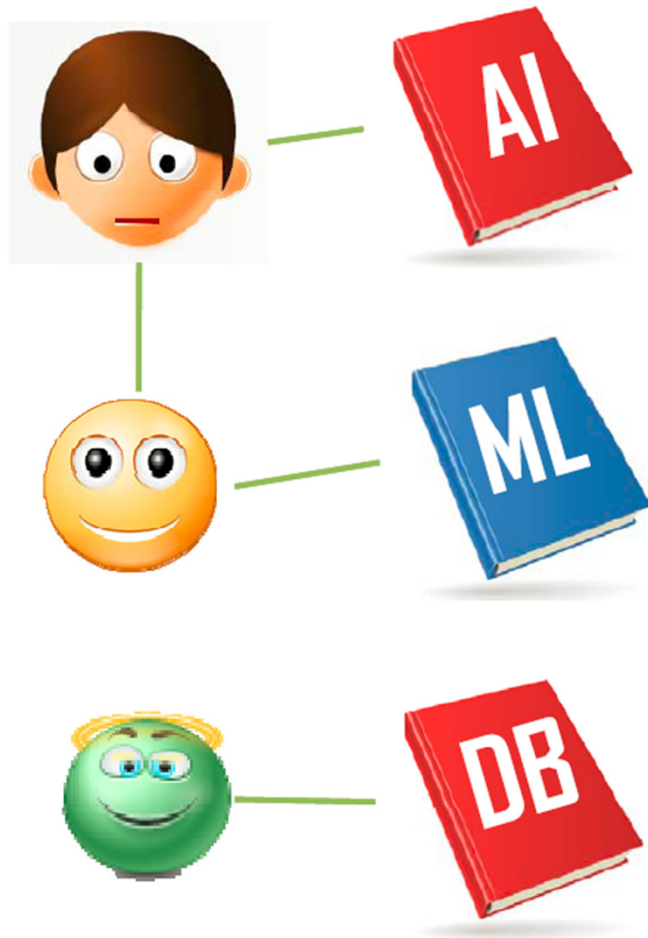
# Reference Matching Problem

- Match noisy records to clean records in a reference table

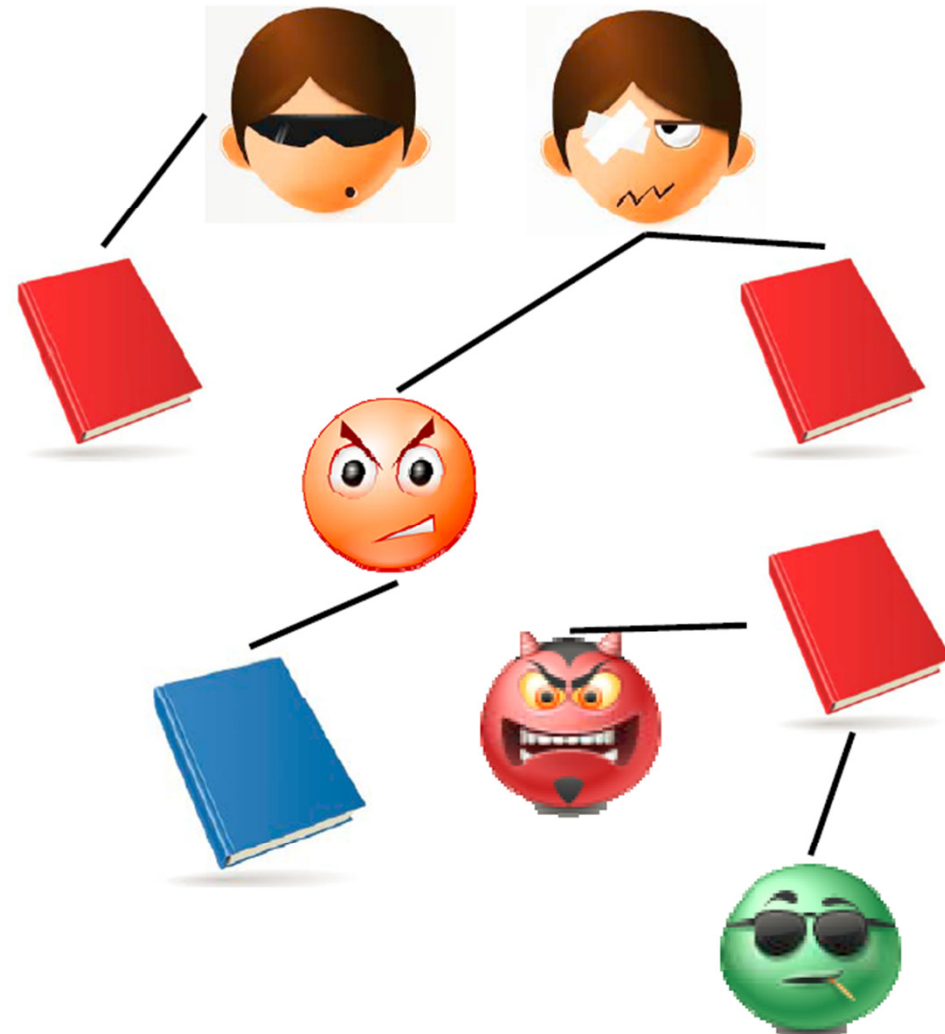


# Abstract Problem Statement

Real World

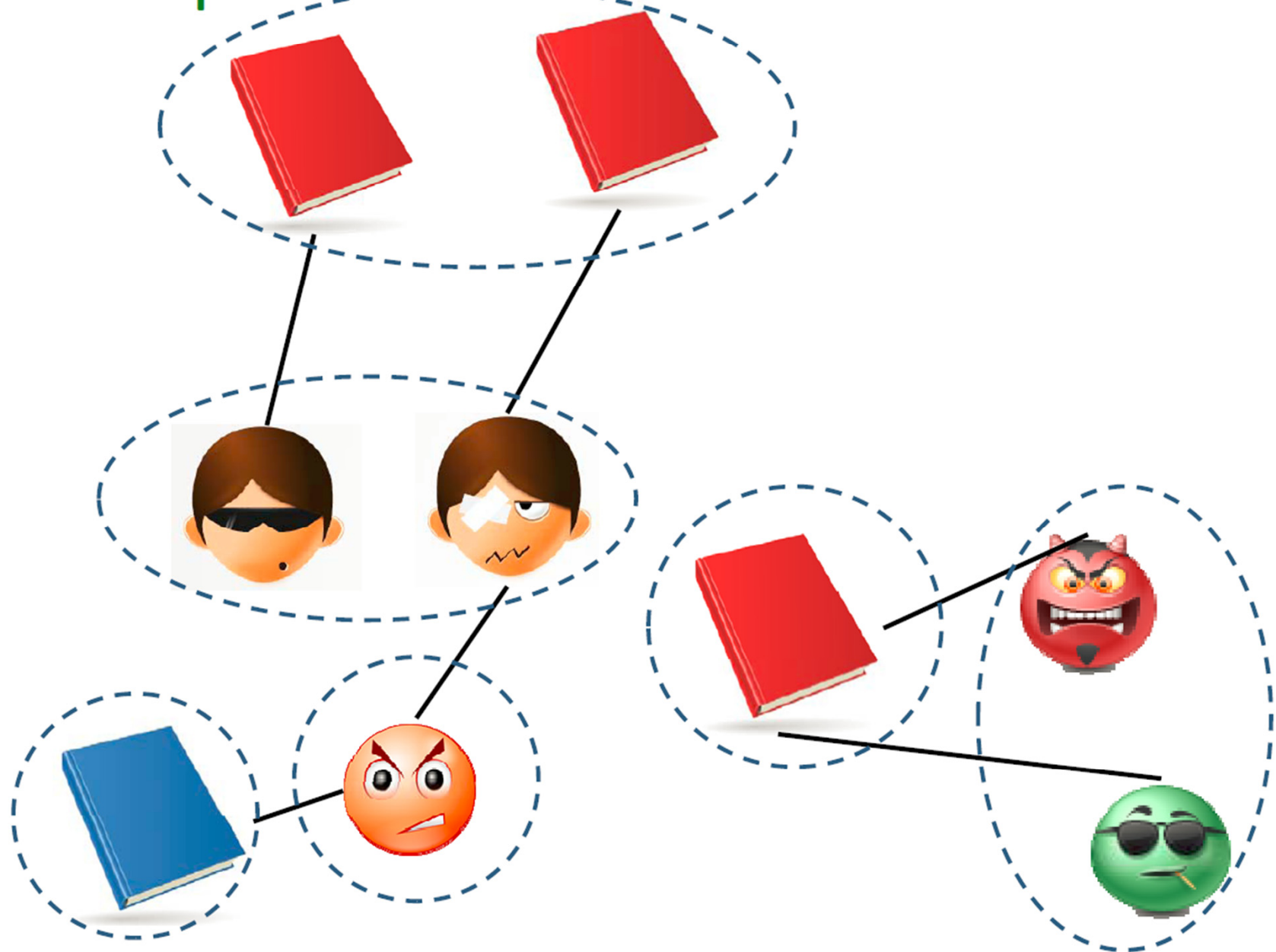


Digital World

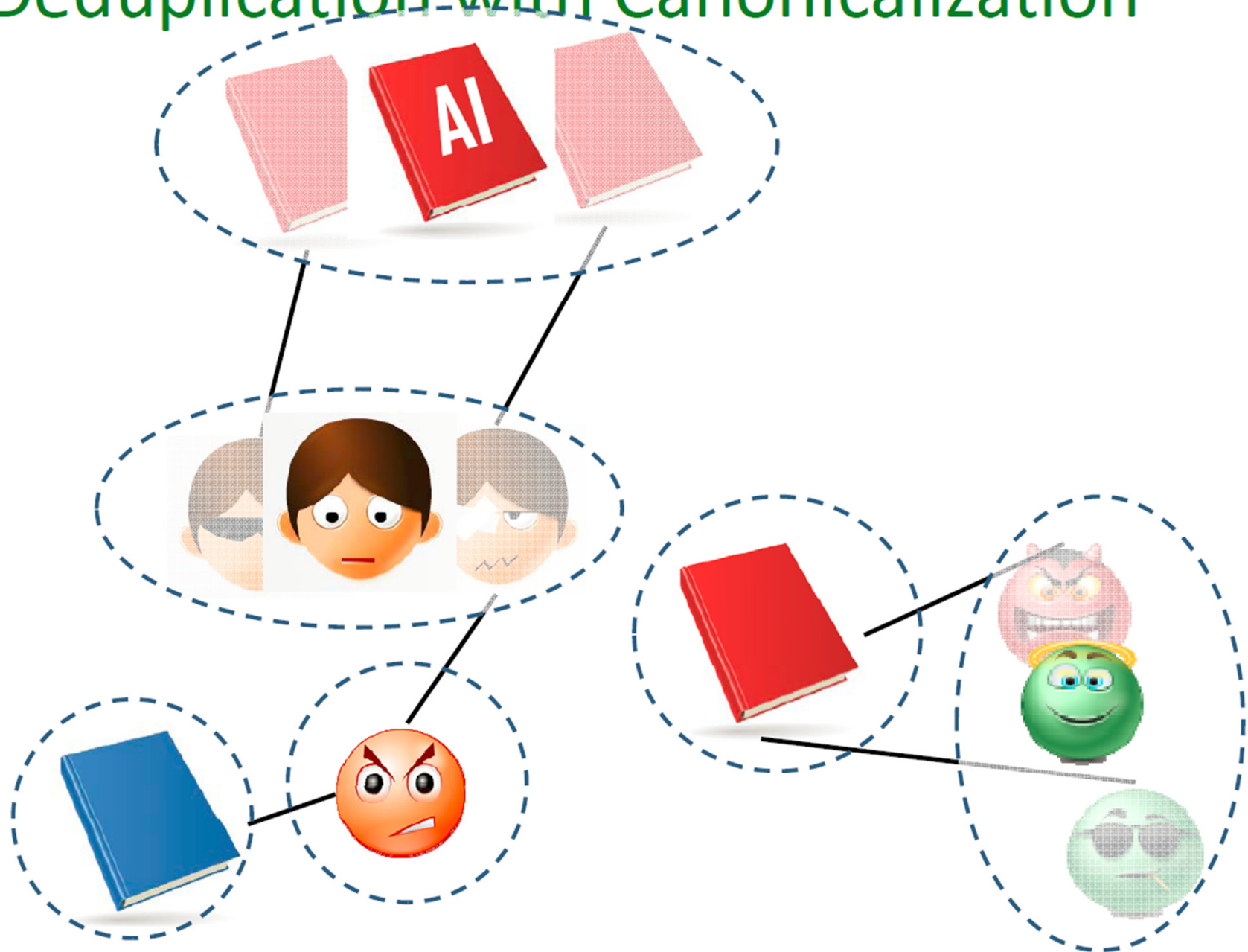




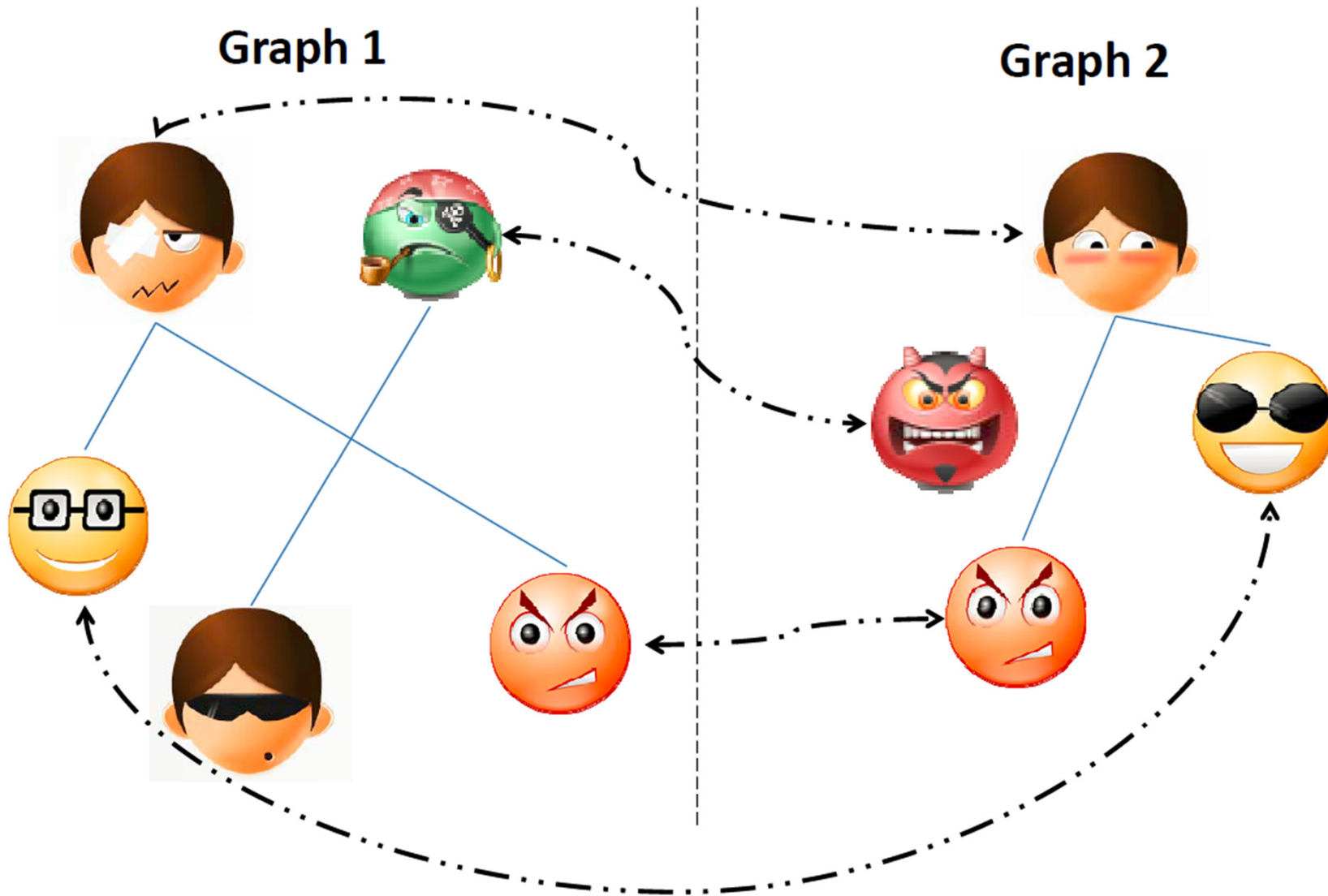
# Deduplication Problem Statement



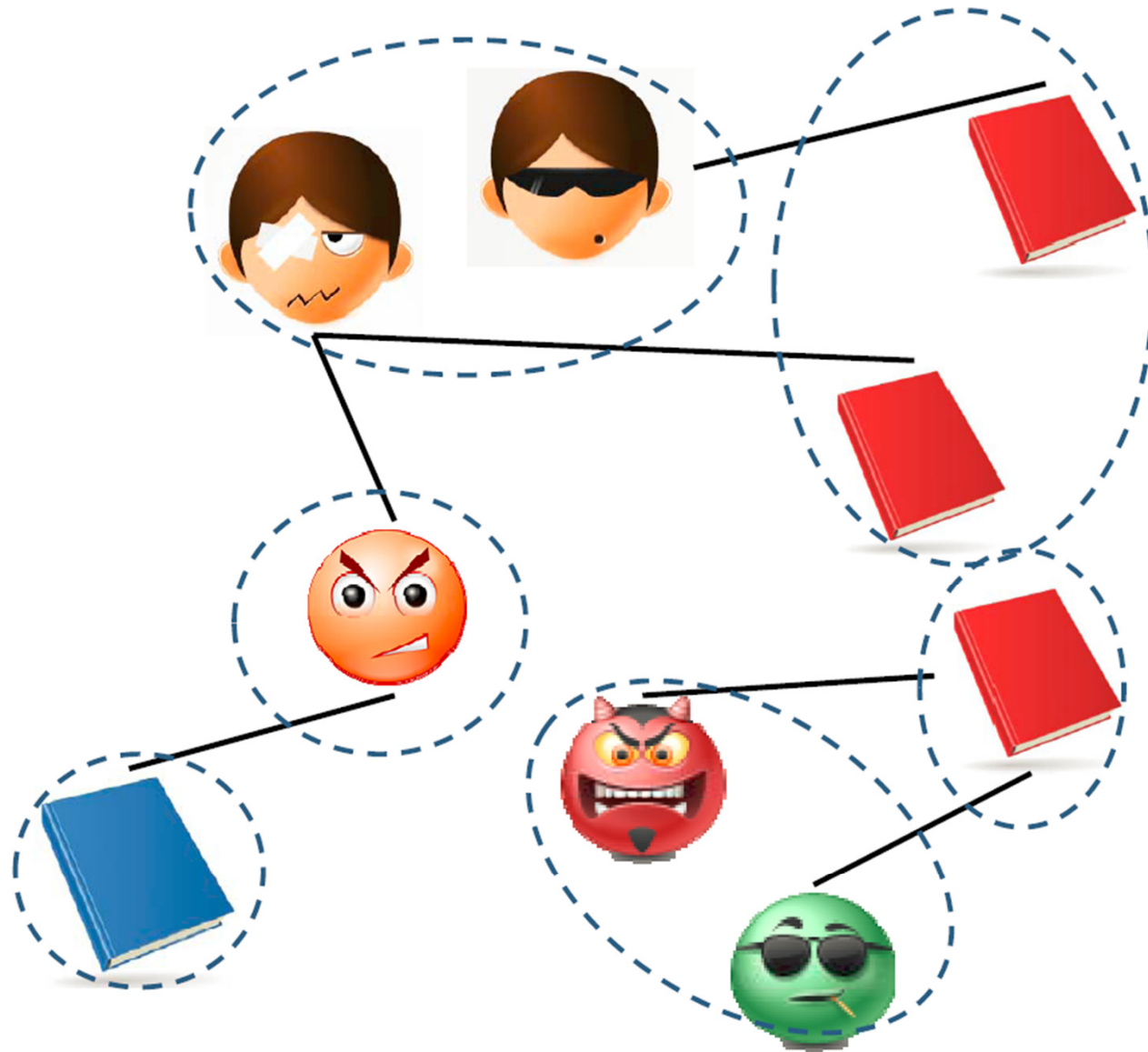
# Deduplication with Canonicalization



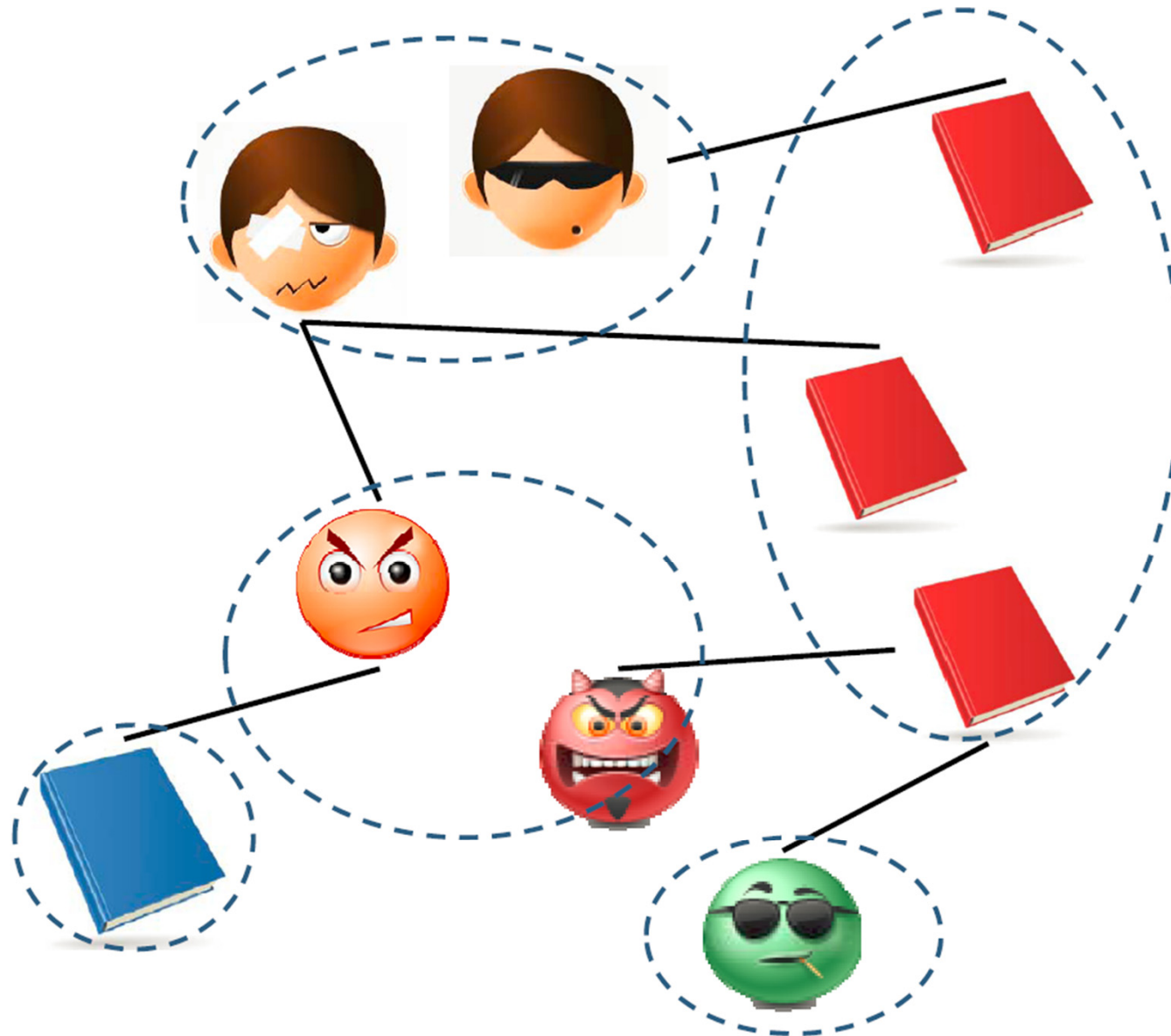
# Graph Alignment (& motif search)



# Relationships are crucial



# Relationships are crucial



# Matching constraints

14

- *Collective*: matching decisions depend on other matching decisions (matching decisions are not made independently)
  
- Match extent
  - Global: If two papers match, then their venues match
    - ◇ Can be applied to all instances of venue mentions
    - ◇ All occurrences of 'SIGMOD' can be matched to 'International Conference on Management of Data'
  - Local: If two papers match, then their authors match
    - ◇ This constraint can only be applied locally
    - ◇ Don't match all occurrences of 'J. Smith' with 'Jeff Smith', only in the context of the current paper

# Constraint examples

15

Type	Example	Hard constraint
Aggregate	C1 = No researcher has published more than five AAAI papers in a year	Hard constraint
Subsumption	C2 = If a citation X from DBLP matches a citation Y in a homepage, then each author mentioned in Y matches some author mentioned in X	
Neighborhood	C3 = If authors X and Y share similar names and some co-authors, they are likely to match	
Incompatible	C4 = No researcher exists who has published in both HCI and numerical analysis	
Layout	C5 = If two mentions in the same document share similar names, they are likely to match	Soft constraint
Key/Uniqueness	C6 = Mentions in the PC listing of a conference is to different researchers	
Ordering	C7 = If two citations match, then their authors will be matched in order	
Individual	C8 = The researcher with the name "Mayssam Saria" has fewer than five mentions in DBLP (new graduate student)	

[Shen, Li & Doan, AAAI05]

# Approaches to handle constraints

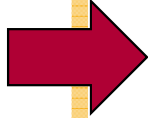
16

- Similarity propagation
  - Dependency graphs
  - Collective Relational Clustering
  
- Probabilistic approaches
  - LDA, CRFs, Markov Logic Networks, Probabilistic Relational Models
  
- Hybrid approaches



# Overview

17



- Reference reconciliation
  - Similarity propagation
  - Dong et al. SIGMOD 2005
- Collective Relational Clustering
  - Similarity propagation
  - Bhattacharya & Getoor, TKDD 2007
- Dedupalog (hybrid approach)
  - Arasu et al. ICDE 2009



# Similarity propagation

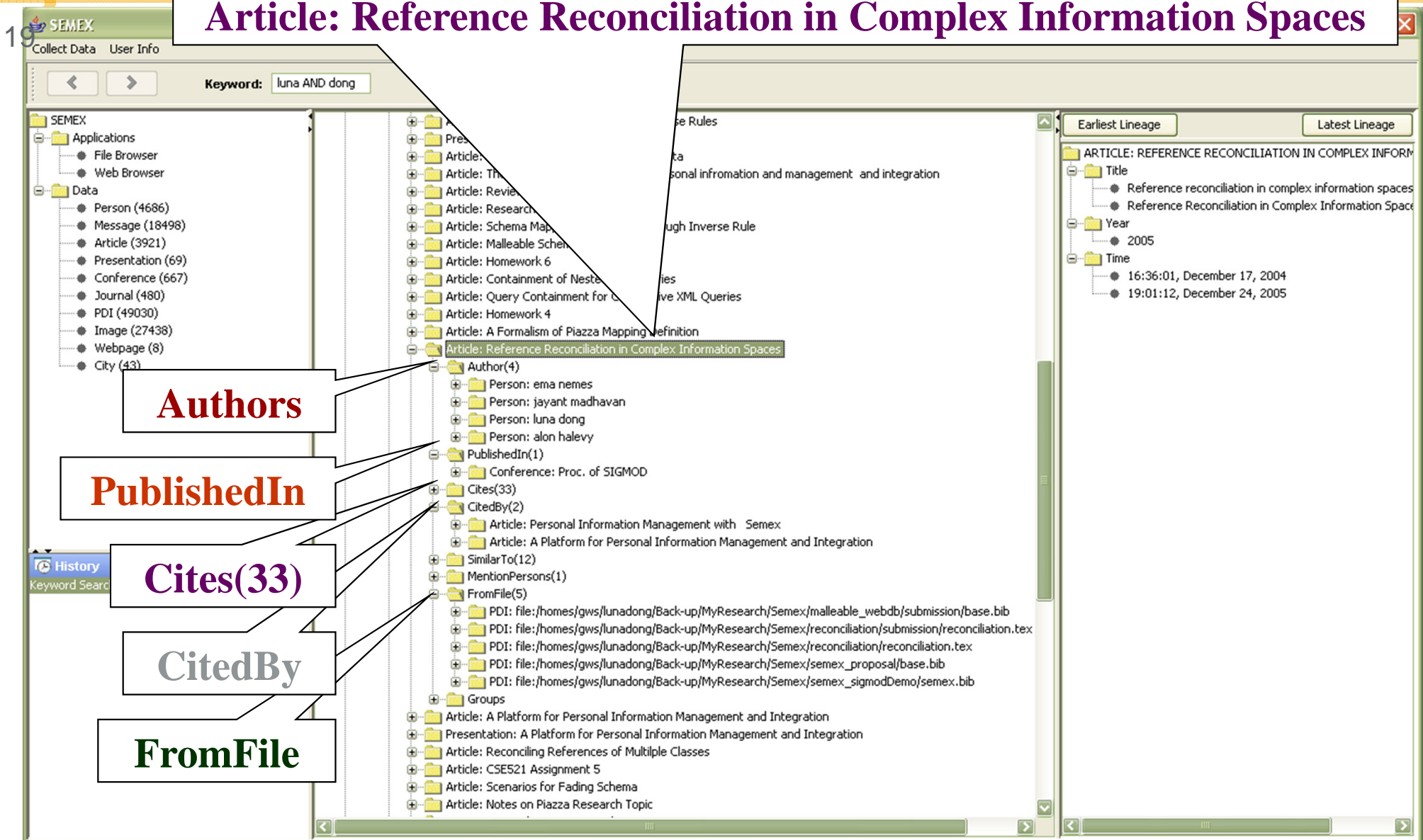
18

- Construct a graph where nodes represent similarity comparisons between attribute values (real-valued) and match decisions based on matching decisions of associated nodes (boolean-valued)
- As mentions are resolved, enriched to contain associated nodes of all matched mentions
- Similarity propagated until fixed point is reached
- Negative constraints (not-match nodes) are checked after similarity propagation is performed, and inconsistencies are fixed

All following slides by Xin Luna Dong

# Semex: Personal Information Management System

## Article: Reference Reconciliation in Complex Information Spaces



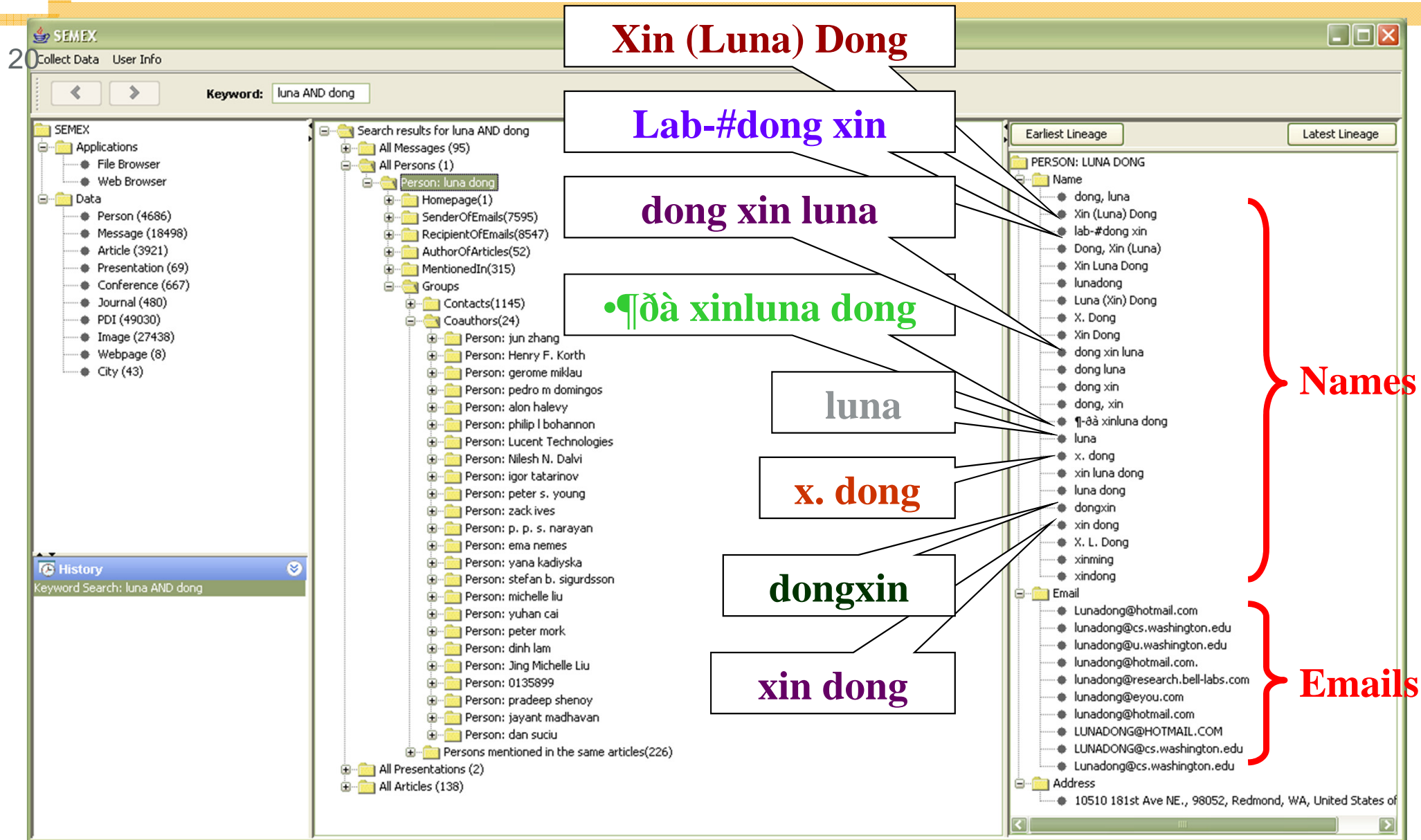
The screenshot shows the SEMEX interface with a search for 'luna AND dong'. The main view displays a tree structure of relationships for the selected article. Callout boxes on the left identify specific relationship types:

- Authors**: Points to the 'Author(4)' relationship, listing authors like 'ema nemes', 'jayant madhavan', 'luna dong', and 'alon halevy'.
- PublishedIn**: Points to the 'PublishedIn(1)' relationship, listing 'Conference: Proc. of SIGMOD'.
- Cites(33)**: Points to the 'Cites(33)' relationship.
- CitedBy**: Points to the 'CitedBy(2)' relationship.
- FromFile**: Points to the 'FromFile(5)' relationship, listing PDI files.

The right pane shows the 'Earliest Lineage' and 'Latest Lineage' for the article, including details like 'Title', 'Year' (2005), and 'Time' (16:36:01, December 17, 2004).

# Semex: Personal Information Management System

20



**Xin (Luna) Dong**

**Lab-#dong xin**

**dong xin luna**

**ꠘꠗà xinluna dong**

**luna**

**x. dong**

**dongxin**

**xin dong**

**Names**

**Emails**

SEMEX  
Collect Data User Info  
Keyword: luna AND dong

SEMEX  
Applications  
File Browser  
Web Browser  
Data  
Person (4686)  
Message (18498)  
Article (3921)  
Presentation (69)  
Conference (667)  
Journal (480)  
PDI (49030)  
Image (27438)  
Webpage (8)  
City (43)

Search results for luna AND dong  
All Messages (95)  
All Persons (1)  
Person: luna.dong  
Homepage(1)  
SenderOfEmails(7595)  
RecipientOfEmails(8547)  
AuthorOfArticles(52)  
MentionedIn(315)  
Groups  
Contacts(1145)  
Coauthors(24)  
Person: jun zhang  
Person: Henry F. Korth  
Person: gerome miklau  
Person: pedro m domingos  
Person: alon halevy  
Person: philip l bohannon  
Person: Lucent Technologies  
Person: Nilesh N. Dalvi  
Person: igor tatarinov  
Person: peter s. young  
Person: zack ives  
Person: p. p. s. narayan  
Person: ema nemes  
Person: yana kadiyska  
Person: stefan b. sigurdsson  
Person: michelle liu  
Person: yuhan cai  
Person: peter mork  
Person: dinh lam  
Person: Jing Michelle Liu  
Person: 0135899  
Person: pradeep shenoy  
Person: jayant madhavan  
Person: dan sucui  
Persons mentioned in the same articles(226)  
All Presentations (2)  
All Articles (138)

Earliest Lineage Latest Lineage  
PERSON: LUNA DONG  
Name  
dong, luna  
Xin (Luna) Dong  
lab-#dong xin  
Dong, Xin (Luna)  
Xin Luna Dong  
lunadong  
Luna (Xin) Dong  
X. Dong  
Xin Dong  
dong xin luna  
dong luna  
dong xin  
dong, xin  
ꠘꠗà xinluna dong  
luna  
x. dong  
xin luna dong  
luna dong  
dongxin  
xin dong  
X. L. Dong  
xinming  
xindong  
Email  
Lunadong@hotmail.com  
lunadong@cs.washington.edu  
lunadong@u.washington.edu  
lunadong@hotmail.com  
lunadong@research.bell-labs.com  
lunadong@eyou.com  
lunadong@hotmail.com  
LUNADONG@HOTMAIL.COM  
LUNADONG@cs.washington.edu  
Lunadong@cs.washington.edu  
Address  
10510 181st Ave NE., 98052, Redmond, WA, United States of

# Intuition

21

- Complex information spaces can be considered as networks of instances and associations between the instances
- Exploit the network: clues hidden in the associations
  - Exploit context (better similarity measure)
    - ◇ Associations between references
    - ◇ Also: Compare values of different attributes
      - Michael Stonebraker vs. [stonebraker@csail.mit.edu](mailto:stonebraker@csail.mit.edu)
  - Iterative algorithm
    - ◇ Propagate information between reconciliation decisions to accumulate positive (and negative) evidences
    - ◇ Re-compare all neighbors of reconciled pairs
  - Gradually enrich references by merging attribute values (Swoosh-style)

# Example

22

Person (name, email, \*coAuthor, \*emailContact)  
Article (title, year, pages, \*authoredBy, \*publishedIn)  
Conference (name, year, location)

**Article**  $a_1 = (\{ \text{"Distributed query processing in a relational data base system"} \}, \{ \text{"169-180"} \}, \{ p_1, p_2, p_3 \}, \{ c_1 \})$   
 $a_2 = (\{ \text{"Distributed query processing in a relational data base system"} \}, \{ \text{"169-180"} \}, \{ p_4, p_5, p_6 \}, \{ c_2 \})$   
**Person**  $p_1 = (\{ \text{"Robert S. Epstein"} \}, \text{null}, \{ p_2, p_3 \}, \text{null})$   
 $p_2 = (\{ \text{"Michael Stonebraker"} \}, \text{null}, \{ p_1, p_3 \}, \text{null})$   
 $p_3 = (\{ \text{"Eugene Wong"} \}, \text{null}, \{ p_1, p_2 \}, \text{null})$   
 $p_4 = (\{ \text{"Epstein, R.S."} \}, \text{null}, \{ p_5, p_6 \}, \text{null})$   
 $p_5 = (\{ \text{"Stonebraker, M."} \}, \text{null}, \{ p_4, p_6 \}, \text{null})$   
 $p_6 = (\{ \text{"Wong, E."} \}, \text{null}, \{ p_4, p_5 \}, \text{null})$   
 $p_7 = (\{ \text{"Eugene Wong"} \}, \{ \text{"eugene@berkeley.edu"} \}, \text{null}, \{ p_8 \})$   
 $p_8 = (\text{null}, \{ \text{"stonebraker@csail.mit.edu"} \}, \text{null}, \{ p_7 \})$   
 $p_9 = (\{ \text{"mike"} \}, \{ \text{"stonebraker@csail.mit.edu"} \}, \text{null}, \text{null})$   
**Conf**  $c_1 = (\{ \text{"ACM Conference on Management of Data"} \}, \{ \text{"1978"} \}, \{ \text{"Austin, Texas"} \})$   
 $c_2 = (\{ \text{"ACM SIGMOD"} \}, \{ \text{"1978"} \}, \text{null})$

## (b) Raw References

$\{ \{ a_1, a_2 \}, \{ p_1, p_4 \}, \{ p_2, p_5, p_8, p_9 \}, \{ p_3, p_6, p_7 \}, \{ c_1, c_2 \} \}$

## (c) Reconciliation Results

- Context
  - If we decide that  $p_6$  and  $p_7$  are the same person, we obtain additional evidence that may lead us to reconcile  $p_5$  and  $p_8$ .
- Iteration
  - Reconciliation of  $a_1$  and  $a_2$  implies that  $p_1$  and  $p_4$ ,  $p_2$  and  $p_5$ , and  $p_3$  and  $p_6$  should be reconciled. Also reconcile conferences  $c_1$  and  $c_2$ .
- Enrichment
  - After reconciling  $p_8$  and  $p_9$ , aggregate their information: "mike" and "Stonebraker, M." share first name initial, and contact the same person by email correspondence or coauthoring. This enables us to reconcile  $p_5$ ,  $p_8$ , and  $p_9$ .

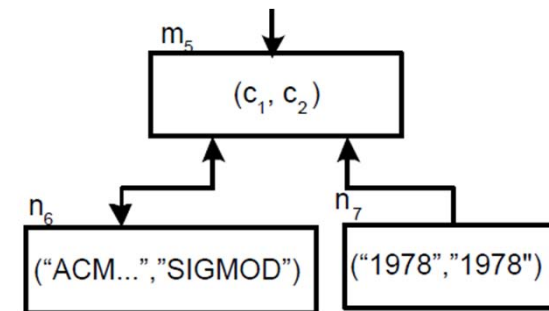
# Dependency graph

23

- For each pair of references  $r_1, r_2$  of the same class, there is a node  $m = (r_1, r_2)$ .
- For each pair of attribute values  $a_1$  of  $r_1$  and  $a_2$  of  $r_2$  (attributes may be of different types), there is a node  $n = (a_1, a_2)$  and an edge between  $m$  and  $n$ .
  - Only include if  $\text{sim} > \theta$  (low threshold)
- Each node has a real-valued similarity score (between 0 and 1).
- A node  $m$  without neighbor is not created

- Refinements:

- Directed edge when dependency is only in one direction
- Real-valued neighbor: Similarity depends of similarity of neighbors
- Strong-Boolean neighbor: Reconciliation implies reconciliation of neighbor
  - ◇ If two papers are reconciled, their conferences must also be reconciled
- Weak-Boolean neighbor: No direct implication
  - ◇ Similarity of two persons increases if the have email correspondence with same person



# Example: Dependency Graph

24

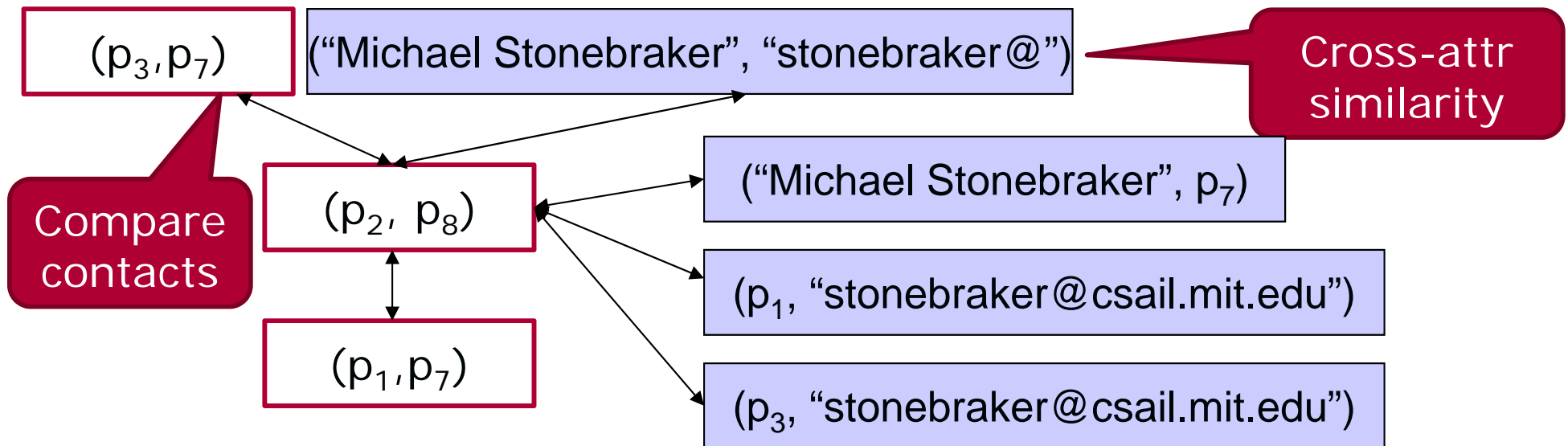
$p_2 = (\text{"Michael Stonebraker"}, \text{null}, \{p_1, p_3\})$

$p_3 = (\text{"Eugene Wong"}, \text{null}, \{p_1, p_2\})$

$p_7 = (\text{"Eugene Wong"}, \text{"eugene@berkeley.edu"}, \{p_8\})$

$p_8 = (\text{null}, \text{"stonebraker@csail.mit.edu"}, \{p_7\})$

$p_9 = (\text{"mike"}, \text{"stonebraker@csail.mit.edu"}, \text{null})$



Reference Similarity



Attribute Similarity



# Idea 1: Consider Richer Evidence

25

- Cross-attribute similarity – name & email
  - p5 = ("Stonebraker, M.", null)
  - p8 = (null, "stonebraker@csail.mit.edu")
- Context Information 1 – contact list
  - p5 = ("Stonebraker, M.", null, {p4, p6})
  - p8 = (null, "stonebraker@csail.mit.edu", {p7})
  - p6 = p7
- Context Information 2 – Authored articles
  - p2 = ("Michael Stonebraker", null)
  - p5 = ("Stonebraker, M.", null)
  - p2 and p5 authored the same article

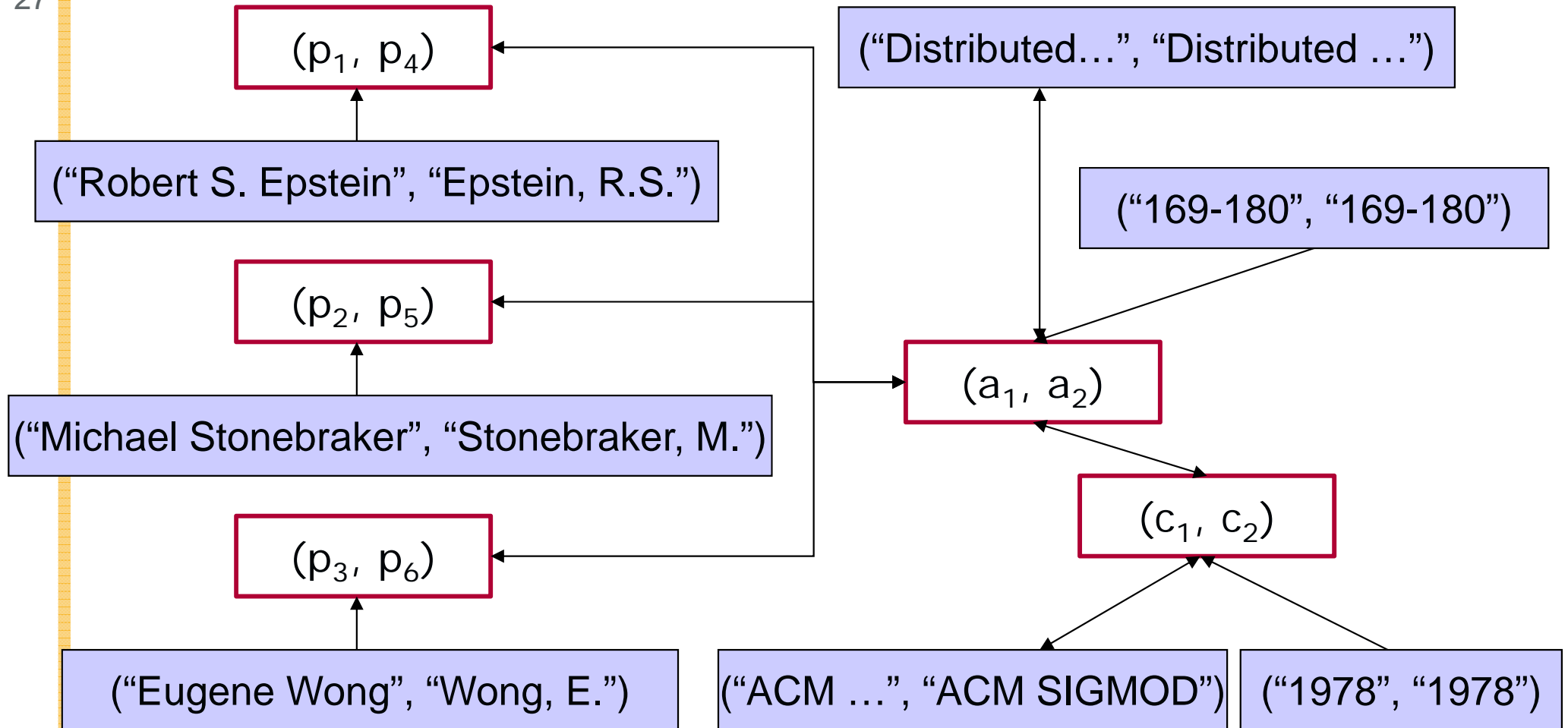
# Idea 2: Propagate Information between Reconciliation Decisions

26

- After changing the similarity score of one node, re-compute similarity scores of its neighbors
  
- Process converges if
  - Similarity score is monotone in the similarity values of neighbors
  - Compute neighbor similarities only if similarity increase is not too small

# Exploit the Dependency Graph

27



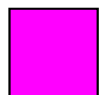
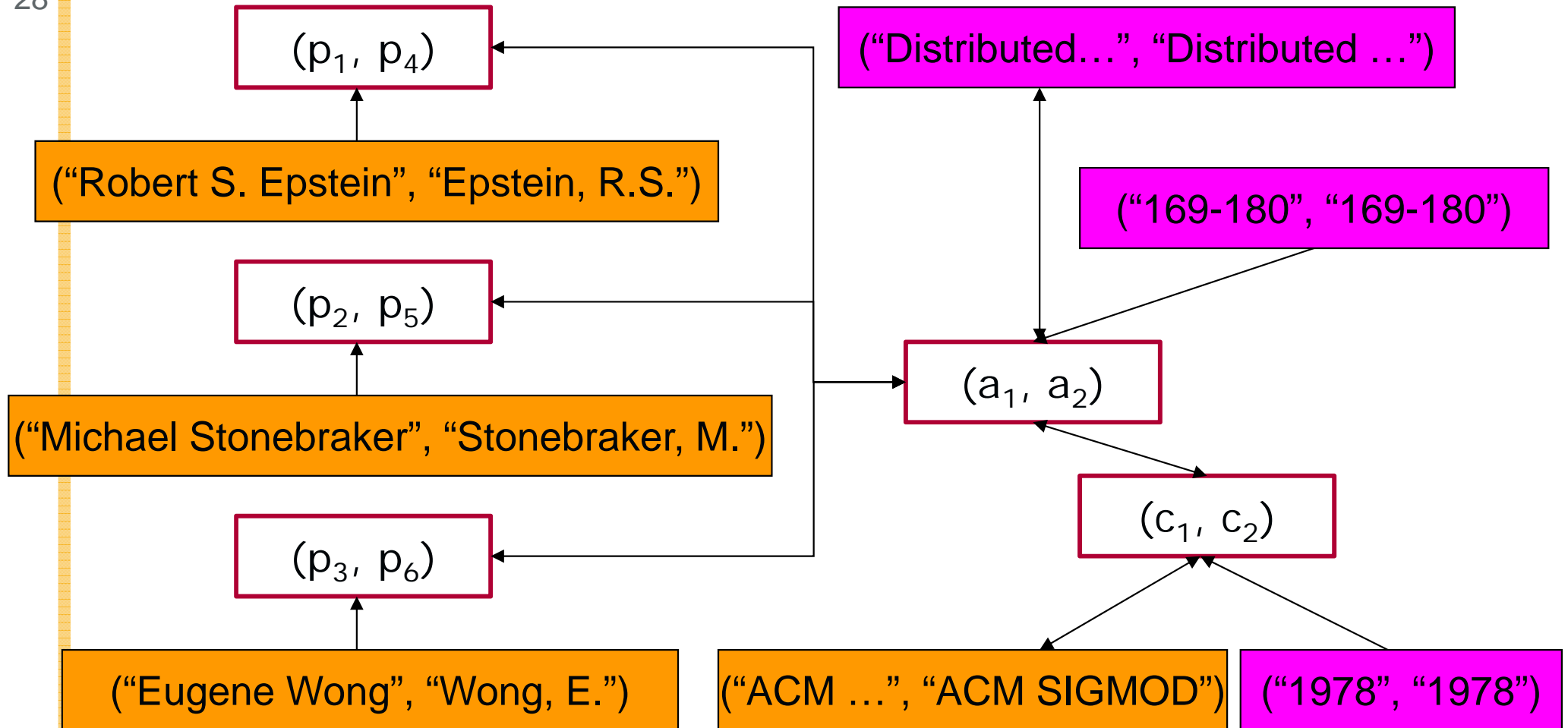
Reference similarity



Attribute similarity

# Exploit the Dependency Graph

28



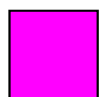
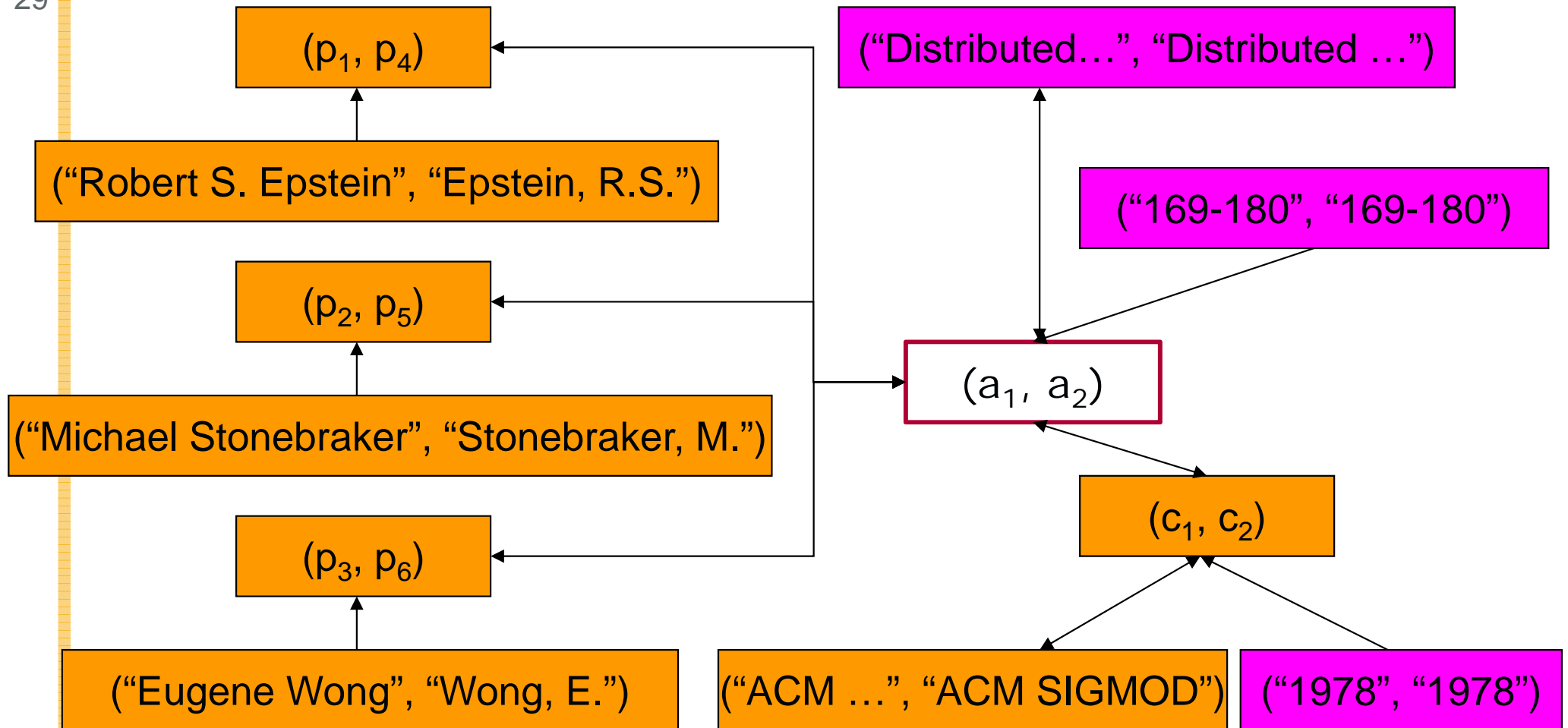
Reconciled



Similar

# Exploit the Dependency Graph

29



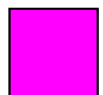
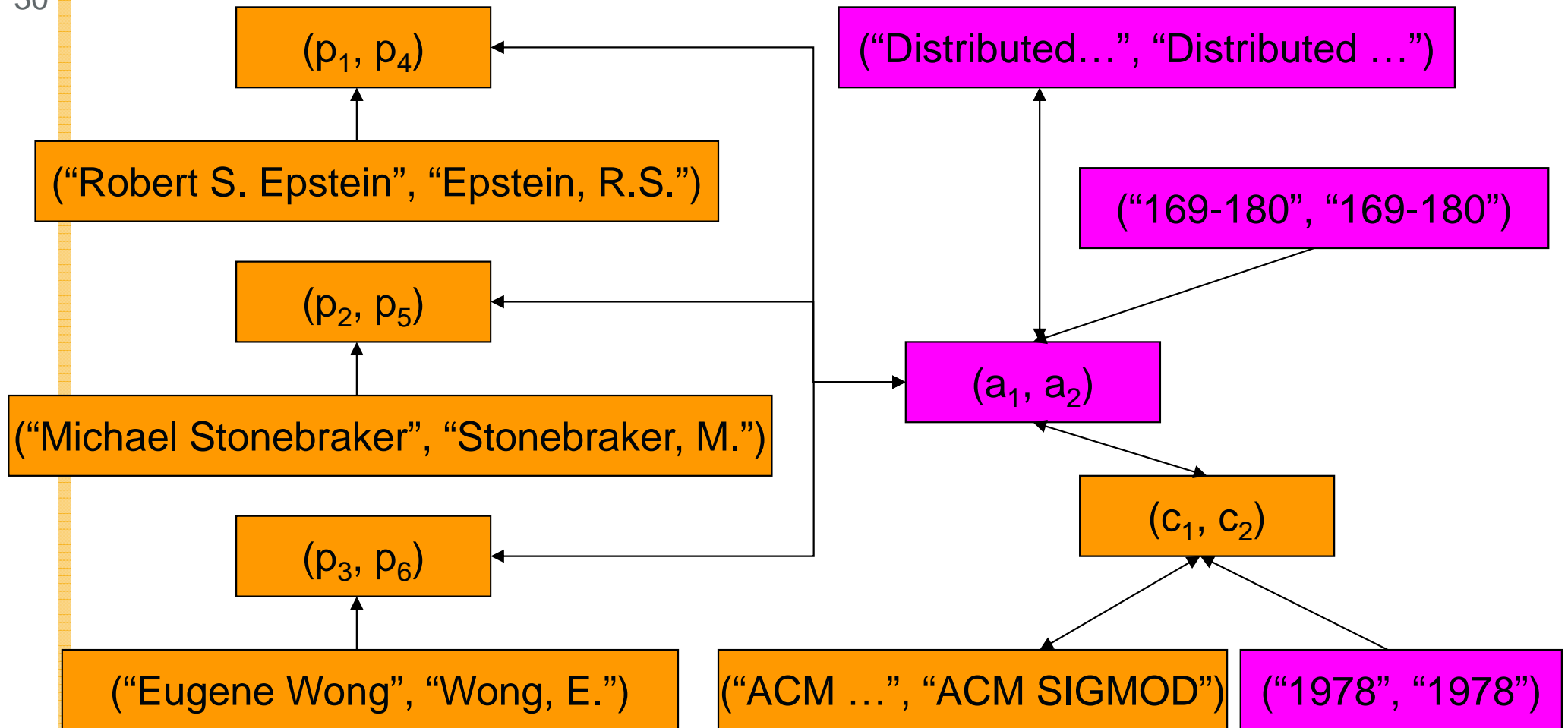
Reconciled



Similar

# Exploit the Dependency Graph

30



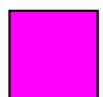
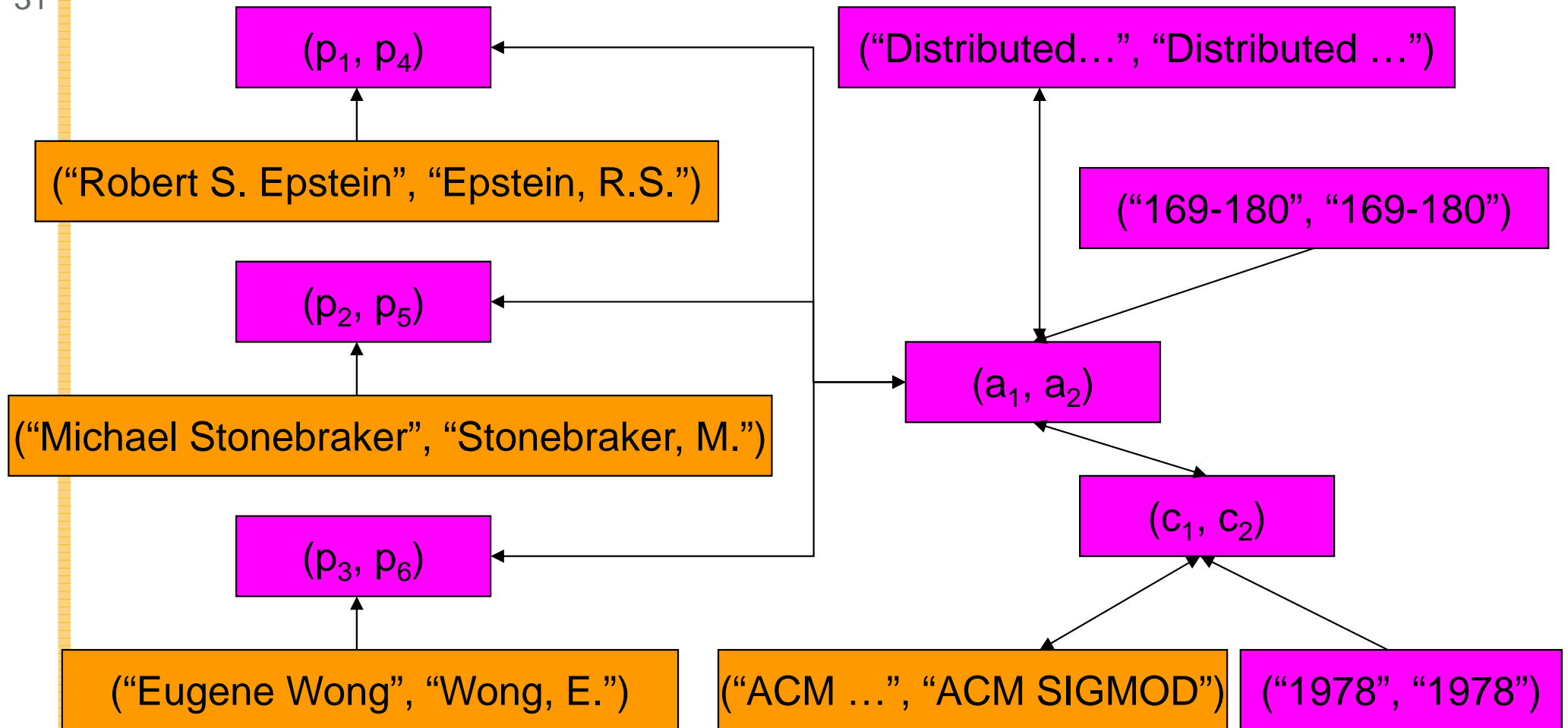
Reconciled



Similar

# Exploit the Dependency Graph

31



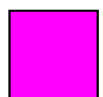
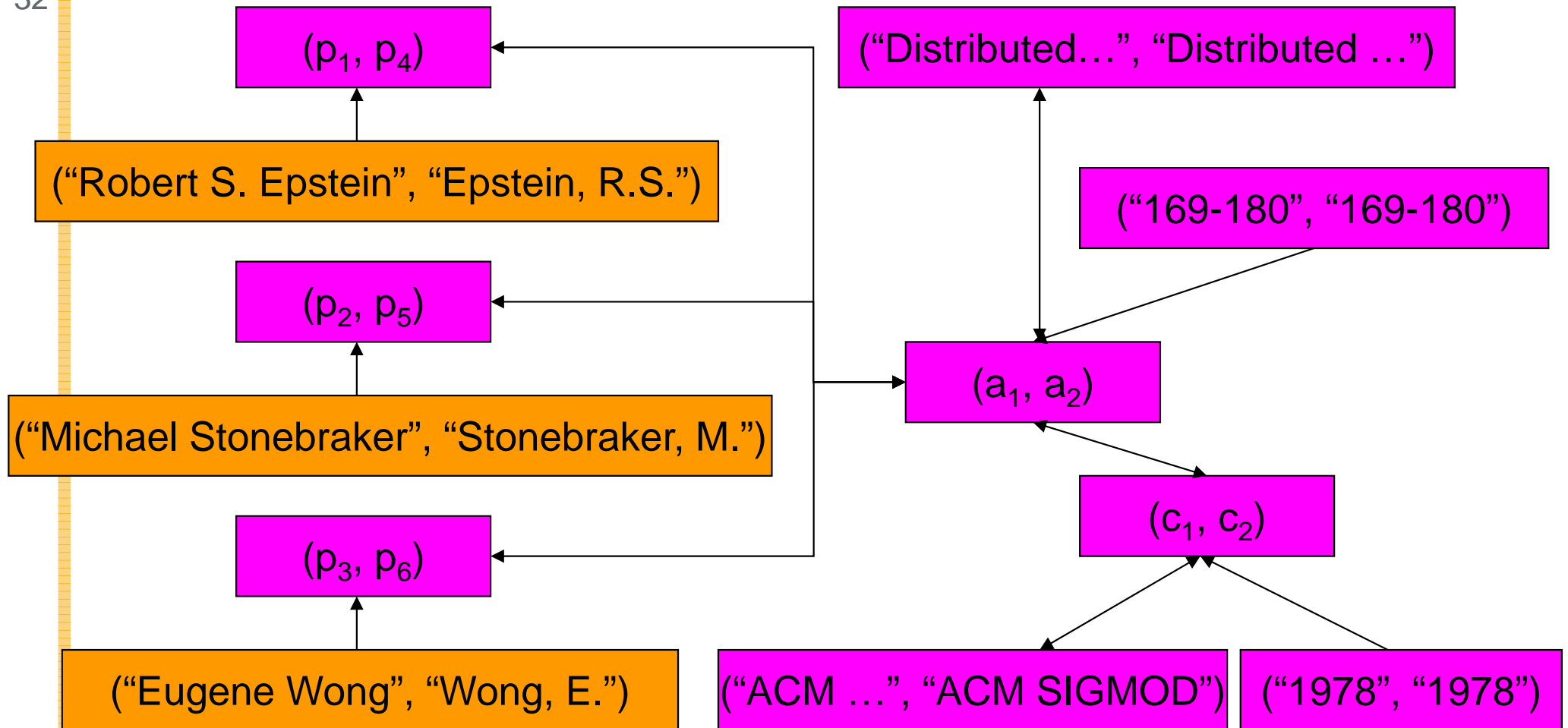
Reconciled



Similar

# Exploit the Dependency Graph

32



Reconciled




Similar



# Idea 3: Enrich References during Reconciliation

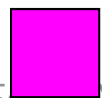
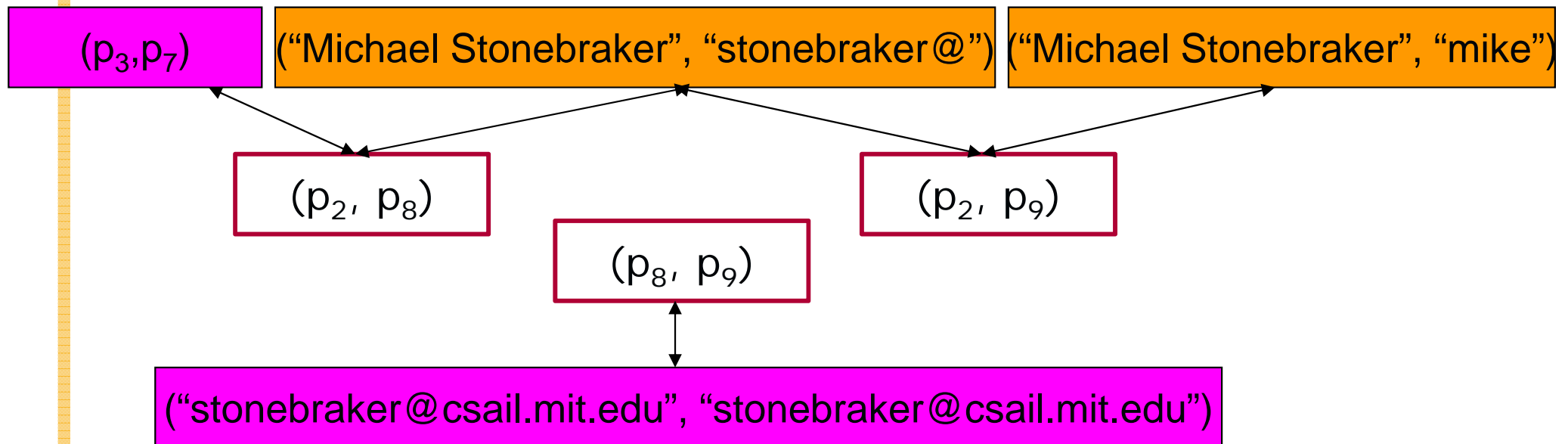
33

- Enrich knowledge of a real-world object for later reconciliation
  - Naive:  
Construct graph → Compute similarity → Transitive Closure
- 
- Problems
    - ◇ Dependency-graph construction is expensive
    - ◇ Reference enrichment takes effect only in next pass
  - Solution
    - Instant enrichment by adding neighbors in the dependency graph

# Enrich References by Adding Neighbors

34

- $p_2 = (\text{"Michael Stonebraker"}, \text{null}, \{p_1, p_3\})$   
 $p_3 = (\text{"Eugene Wong"}, \text{null}, \{p_1, p_2\})$   
 $p_7 = (\text{"Eugene Wong"}, \text{"eugene@berkeley.edu"}, \{p_8\})$   
 $p_8 = (\text{null}, \text{"stonebraker@csail.mit.edu"}, \{p_7\})$   
 $p_9 = (\text{"mike"}, \text{"stonebraker@csail.mit.edu"}, \text{null})$



Reconciled

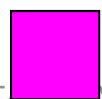
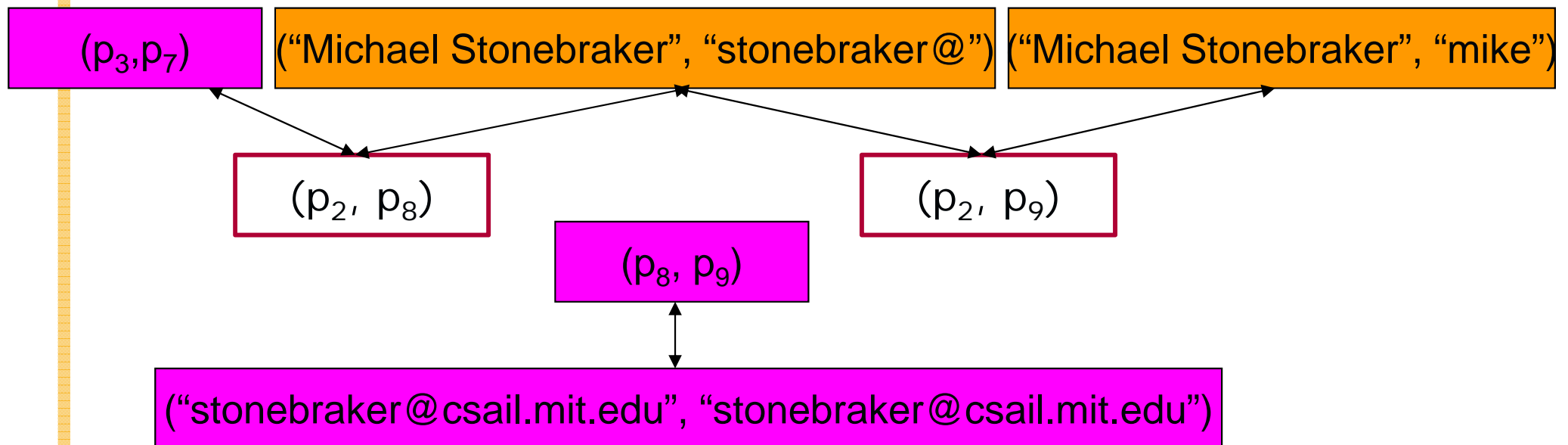


Similar

# Enrich References by Adding Neighbors

35

- $p_2 = (\text{"Michael Stonebraker"}, \text{null}, \{p_1, p_3\})$   
 $p_3 = (\text{"Eugene Wong"}, \text{null}, \{p_1, p_2\})$   
 $p_7 = (\text{"Eugene Wong"}, \text{"eugene@berkeley.edu"}, \{p_8\})$   
 $p_8 = (\text{null}, \text{"stonebraker@csail.mit.edu"}, \{p_7\})$   
 $p_9 = (\text{"mike"}, \text{"stonebraker@csail.mit.edu"}, \text{null})$



Reconciled

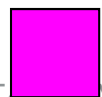
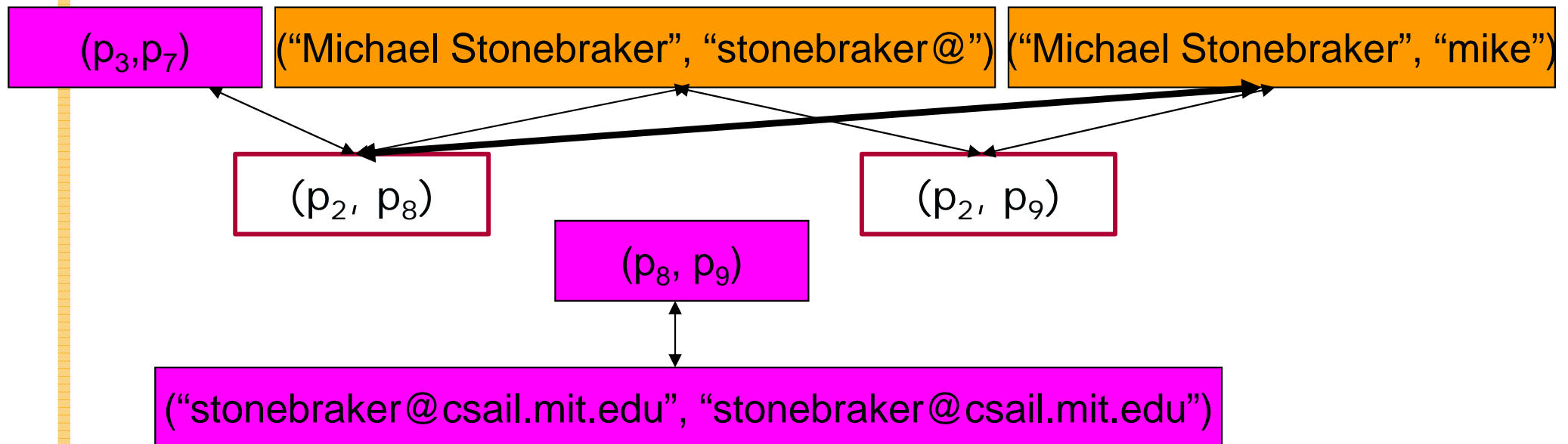


Similar

# Enrich References by Adding Neighbors

36

- $p_2 = (\text{"Michael Stonebraker"}, \text{null}, \{p_1, p_3\})$   
 $p_3 = (\text{"Eugene Wong"}, \text{null}, \{p_1, p_2\})$   
 $p_7 = (\text{"Eugene Wong"}, \text{"eugene@berkeley.edu"}, \{p_8\})$   
 $p_8 = (\text{null}, \text{"stonebraker@csail.mit.edu"}, \{p_7\})$   
 $p_9 = (\text{"mike"}, \text{"stonebraker@csail.mit.edu"}, \text{null})$



Reconciled

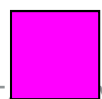
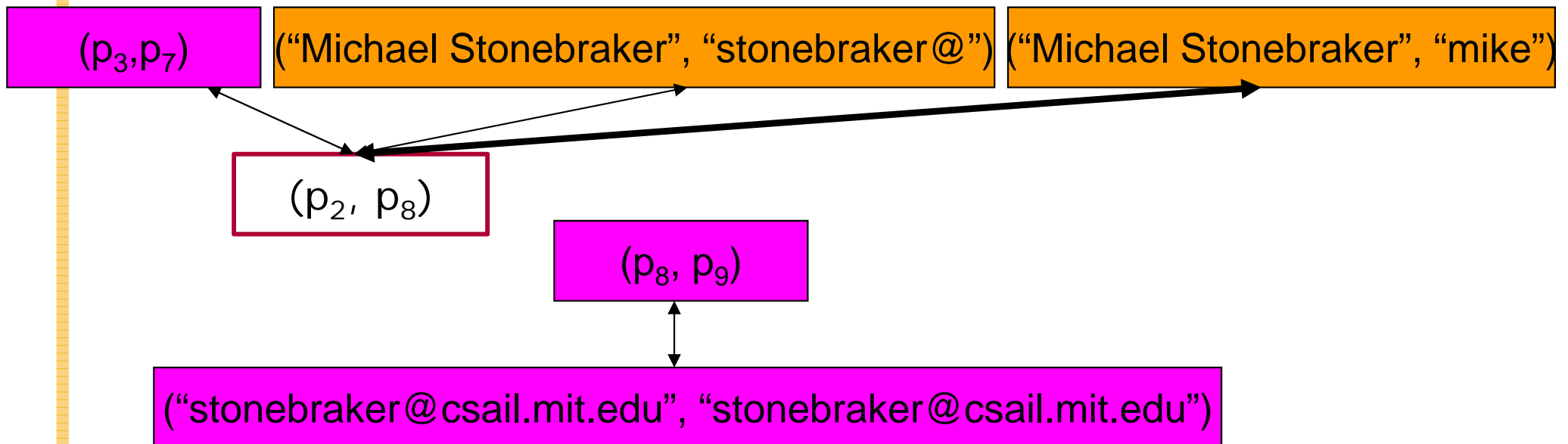


Similar

# Enrich References by Adding Neighbors

37

- $p_2 = (\text{"Michael Stonebraker"}, \text{null}, \{p_1, p_3\})$
- $p_3 = (\text{"Eugene Wong"}, \text{null}, \{p_1, p_2\})$
- $p_7 = (\text{"Eugene Wong"}, \text{"eugene@berkeley.edu"}, \{p_8\})$
- $p_8 = (\text{null}, \text{"stonebraker@csail.mit.edu"}, \{p_7\})$
- $p_9 = (\text{"mike"}, \text{"stonebraker@csail.mit.edu"}, \text{null})$



Reconciled

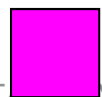
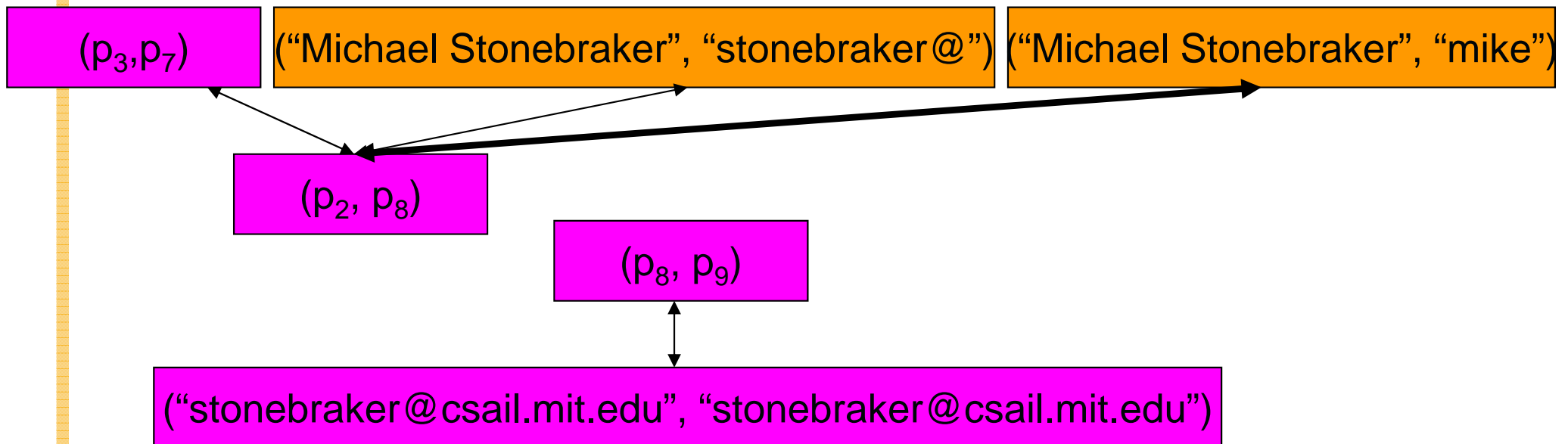


Similar

# Enrich References by Adding Neighbors

38

- $p_2 = (\text{"Michael Stonebraker"}, \text{null}, \{p_1, p_3\})$
- $p_3 = (\text{"Eugene Wong"}, \text{null}, \{p_1, p_2\})$
- $p_7 = (\text{"Eugene Wong"}, \text{"eugene@berkeley.edu"}, \{p_8\})$
- $p_8 = (\text{null}, \text{"stonebraker@csail.mit.edu"}, \{p_7\})$
- $p_9 = (\text{"mike"}, \text{"stonebraker@csail.mit.edu"}, \text{null})$



Reconciled

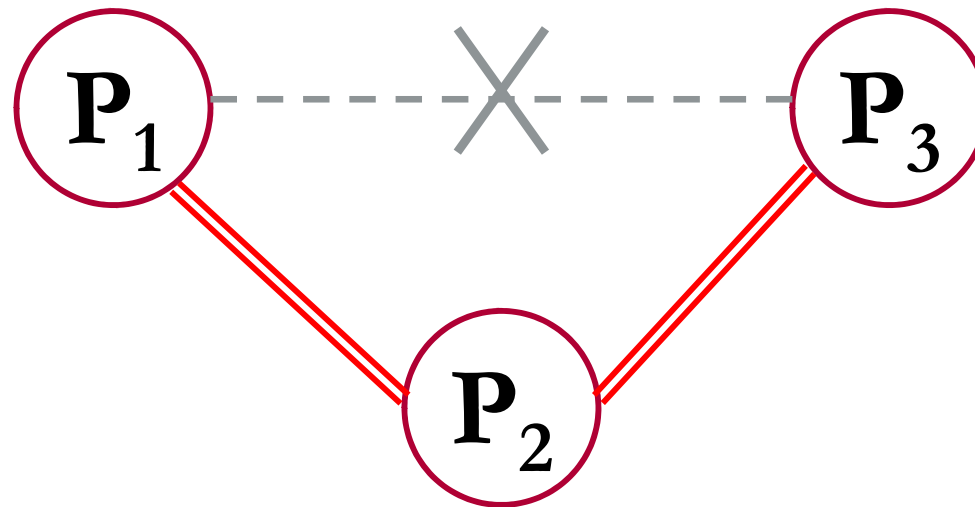


Similar

# Idea 4: Enforce Constraints

39

■ Problem:

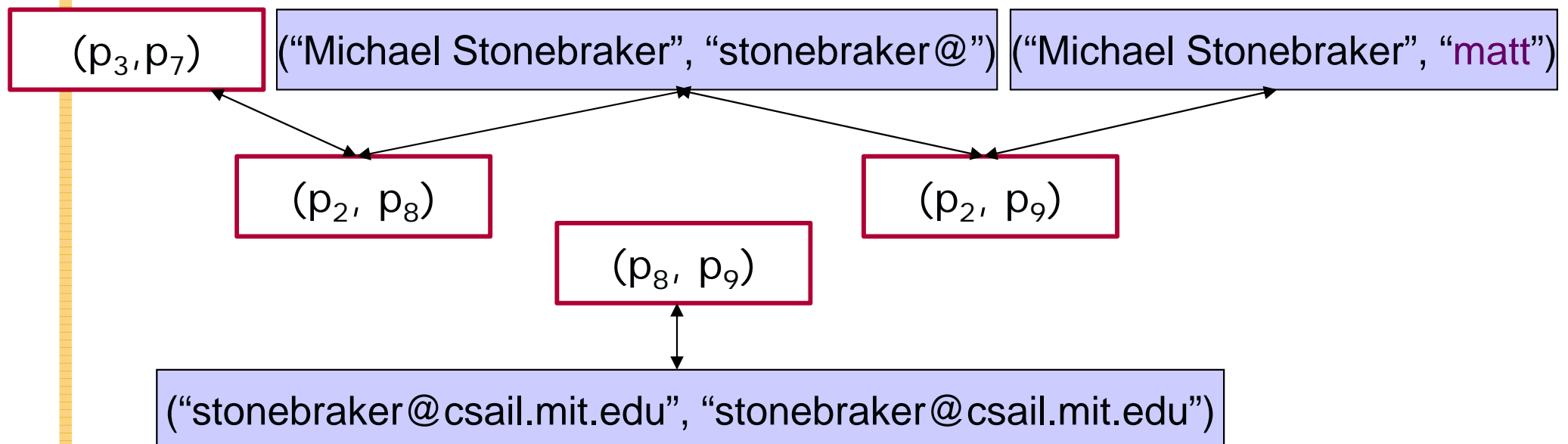


- Solution: Propagate negative information (as constraints)
  - *Non-merge node*: the two elements are guaranteed to be different and should never be merged
  - Domain-dependent, written by expert
- Example: Authors of a publication are always distinct

# Enforce Constraints by Propagating Negative Information

40

- $p_2 = (\text{"Michael Stonebraker"}, \text{null}, \{p_1, p_3\})$   
 $p_3 = (\text{"Eugene Wong"}, \text{null}, \{p_1, p_2\})$   
 $p_7 = (\text{"Eugene Wong"}, \text{"eugene@berkeley.edu"}, \{p_8\})$   
 $p_8 = (\text{null}, \text{"stonebraker@csail.mit.edu"}, \{p_7\})$   
 $p_9 = (\text{"matt"}, \text{"stonebraker@csail.mit.edu"}, \text{null})$



Reference Similarity



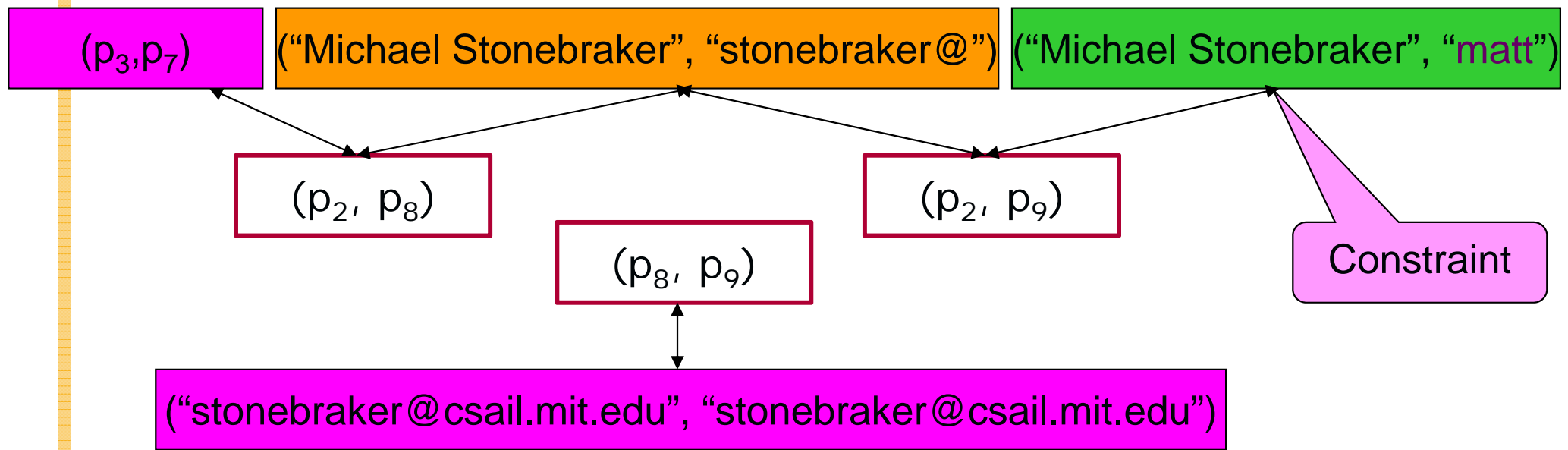
Attribute Similarity



# Enforce Constraints by Propagating Negative Information

41

- $p_2 = (\text{"Michael Stonebraker"}, \text{null}, \{p_1, p_3\})$   
 $p_3 = (\text{"Eugene Wong"}, \text{null}, \{p_1, p_2\})$   
 $p_7 = (\text{"Eugene Wong"}, \text{"eugene@berkeley.edu"}, \{p_8\})$   
 $p_8 = (\text{null}, \text{"stonebraker@csail.mit.edu"}, \{p_7\})$   
 $p_9 = (\text{"matt"}, \text{"stonebraker@csail.mit.edu"}, \text{null})$



Reconciled



Similar

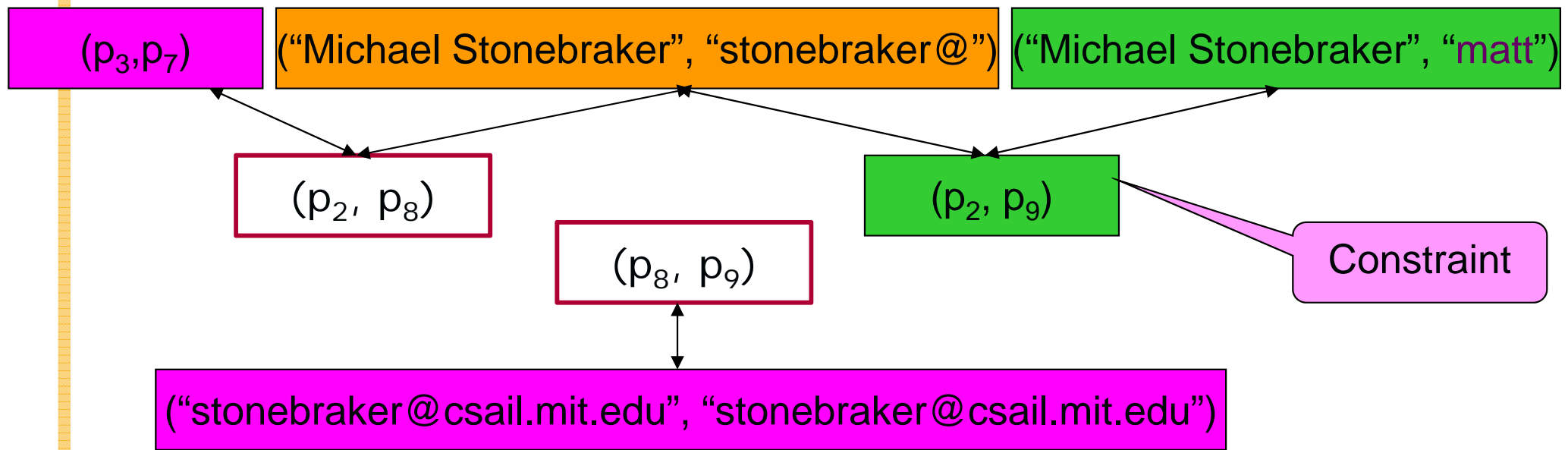


Non-merge

# Enforce Constraints by Propagating Negative Information

42

- $p_2 = (\text{"Michael Stonebraker"}, \text{null}, \{p_1, p_3\})$   
 $p_3 = (\text{"Eugene Wong"}, \text{null}, \{p_1, p_2\})$   
 $p_7 = (\text{"Eugene Wong"}, \text{"eugene@berkeley.edu"}, \{p_8\})$   
 $p_8 = (\text{null}, \text{"stonebraker@csail.mit.edu"}, \{p_7\})$   
 $p_9 = (\text{"matt"}, \text{"stonebraker@csail.mit.edu"}, \text{null})$

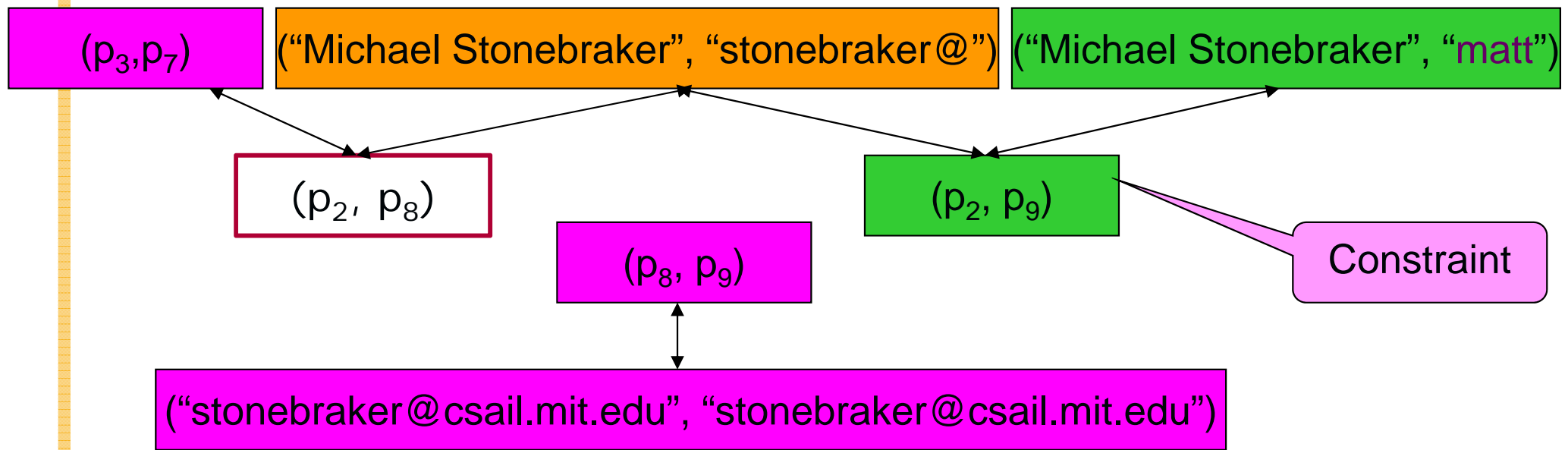


Reconciled
  Similar
  Non-merge

# Enforce Constraints by Propagating Negative Information

43

- $p_2 = (\text{"Michael Stonebraker"}, \text{null}, \{p_1, p_3\})$   
 $p_3 = (\text{"Eugene Wong"}, \text{null}, \{p_1, p_2\})$   
 $p_7 = (\text{"Eugene Wong"}, \text{"eugene@berkeley.edu"}, \{p_8\})$   
 $p_8 = (\text{null}, \text{"stonebraker@csail.mit.edu"}, \{p_7\})$   
 $p_9 = (\text{"matt"}, \text{"stonebraker@csail.mit.edu"}, \text{null})$

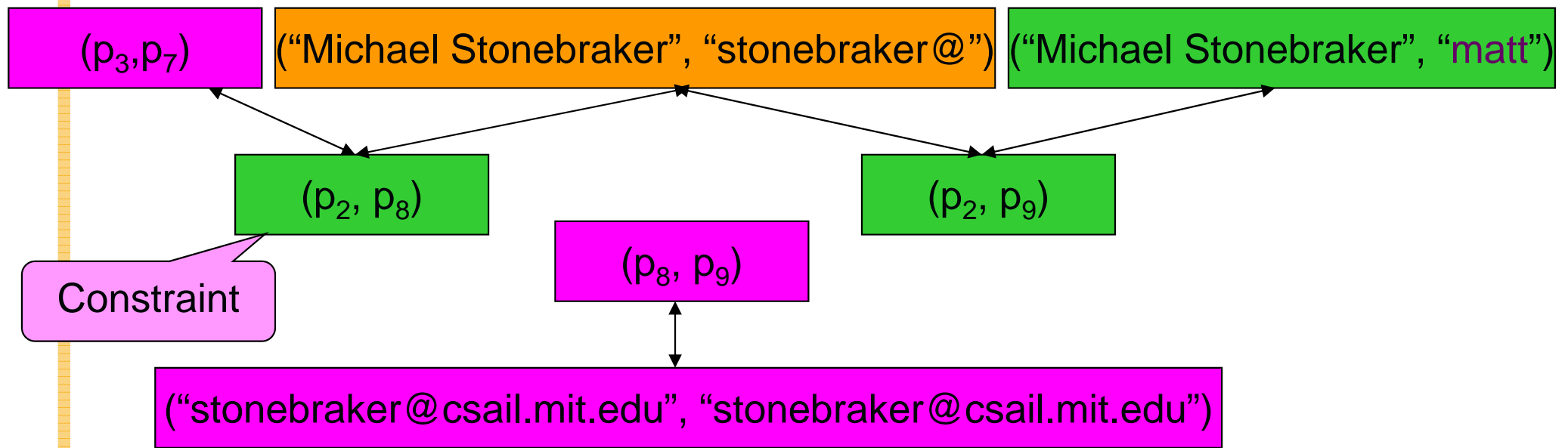


Reconciled
  Similar
  Non-merge

# Enforce Constraints by Propagating Negative Information

44

- $p_2 = (\text{"Michael Stonebraker"}, \text{null}, \{p_1, p_3\})$   
 $p_3 = (\text{"Eugene Wong"}, \text{null}, \{p_1, p_2\})$   
 $p_7 = (\text{"Eugene Wong"}, \text{"eugene@berkeley.edu"}, \{p_8\})$   
 $p_8 = (\text{null}, \text{"stonebraker@csail.mit.edu"}, \{p_7\})$   
 $p_9 = (\text{"matt"}, \text{"stonebraker@csail.mit.edu"}, \text{null})$



Constraint

Reconciled
  Similar
  Non-merge