



# Record Linkage

---

Craig Knoblock  
University of Southern California

These slides are based in part on slides from  
Sheila Tejada, Misha Bilenko, Jose Luis Ambite, Claude Nanjo, and Steve Minton



# Record Linkage Problem

Restaurant Name	Address	City	Phone	Cuisine
Fenix	8358 Sunset Blvd. <b>West</b>	Hollywood	213/848-6677	<b>American</b>
Fenix <b>at the Argyle</b>	8358 Sunset Blvd.	<b>W.</b> Hollywood	213-848-6677	<b>French (new)</b>

- Task:  
**identify syntactically different records that refer to the same entity**
- Common sources of variation: database merges, typographic errors, abbreviations, extraction errors, OCR scanning errors, etc.

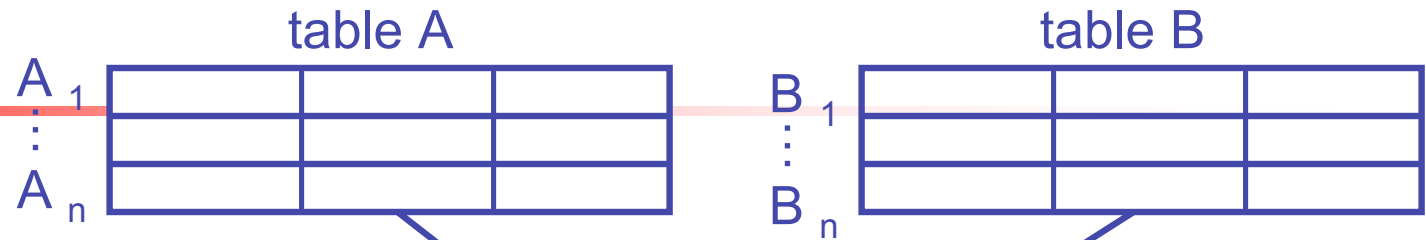


# General Approach to Record Linkage

---

1. **Identification of candidate pairs (blocking)**
  - Comparing all possible record pairs would be computationally wasteful
2. **Compute Field Similarity**
  - String similarity between individual fields is computed
3. **Compute Record Similarity**
  - Field similarities are combined into a total record similarity estimate

# Overview



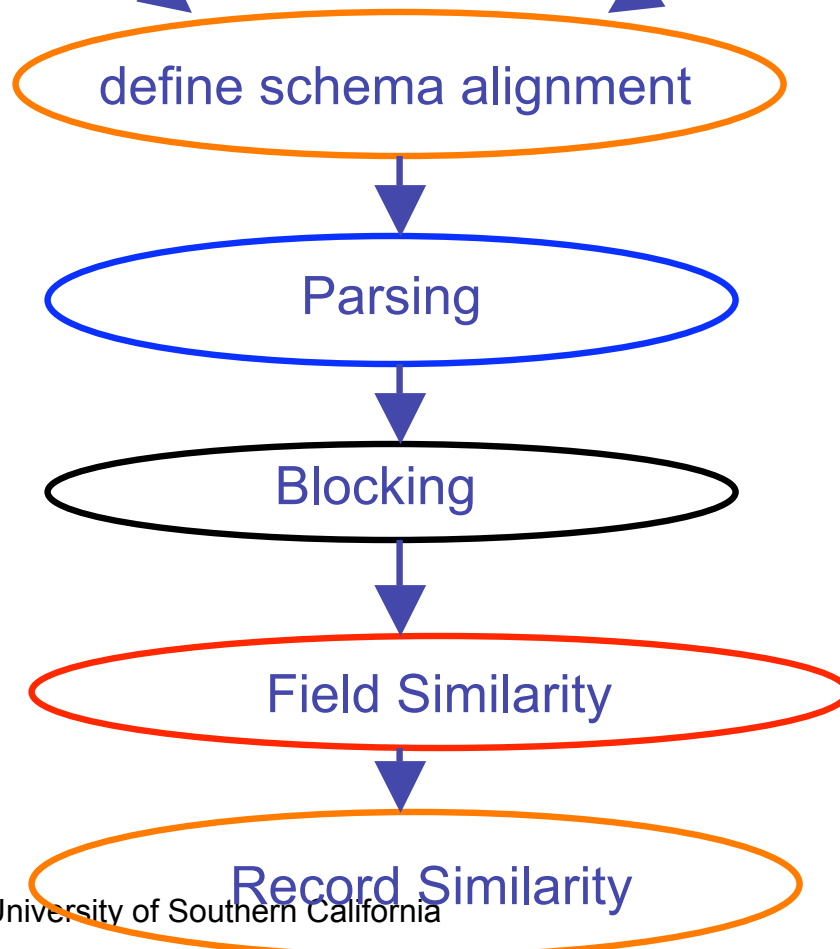
Map attribute(s) from one datasource to attribute(s) from the other datasource.

Tokenize, then label tokens

Eliminate highly unlikely candidate record pairs.

Use learned distance metric to score field

Pass feature vector to SVM classifier to get overall score for candidate pair.





# Outline

---

- Blocking
- Field Matching
- Record Matching
- Entity Matching
- Conclusion



# Outline

---

- **Blocking**
- Field Matching
- Record Matching
- Entity Matching
- Conclusion



# Blocking

---

- Comparing all possible matches across two data sets would require  $n^2$  comparisons
- On large datasets this is impractical and wasteful
- Instead, we compare only those that could possibly be matched
- Also referred to as candidate generation



# IR Approach to Blocking

---

- Construct an inverted index of all tokens in a document
  - Links the token to the documents in which it appears
  - Place each token in a hash table
- Apply transformations on the tokens to find closely related tokens
  - Transformations include equal, stemming, soundex, and other unary transformations
- Use a stop list to avoid common tokens
  - Tokens such as “the”, “s”, etc. would be on the stop list



# More on Blocking Later!

---



# Outline

---

- Blocking
- **Field Matching**
- Record Matching
- Entity Matching
- Conclusion



# Field Matching Approaches

---

- Expert-system rules
  - Manually written
- Token similarity
  - Used in Whirl
- String similarity
  - Used in Marlin
- Learned transformation weights
  - Used in HFM



# Token-based Metrics

- Any string can be treated as a *bag of tokens* .
  - “8358 Sunset Blvd” ► {8358, Sunset, Blvd}
  - “8358 Sunset Blvd” ► {‘8358’, ‘358 ‘, ‘58 S’, ‘8 Su’, ‘ Sun’, ‘Suns’, ‘unse’, ‘nset’, ‘set ‘, ‘et B’, ‘t Bl’, ‘ Blv’, ‘Blvd’}
- Each token corresponds to a dimension in Euclidean space; string similarity is the normalized dot product (cosine) in the vector space.
- Weighting tokens by Inverse Document Frequency (IDF) is a form of *unsupervised* string metric learning.

# Token Similarity

[Cohen, 1998]

- Idea: Evaluate the similarity of records via textual similarity. Used in Whirl (Cohen 1998).
- Follows the same approach used by classical IR algorithms (including web search engines).
- First, “stemming” is applied to each entry.
  - E.g. “Joe’s Diner” -> “Joe [‘s] Diner”
- Then, entries are compared by counting the number of words in common.
- Note: Infrequent words weighted more heavily by TF/IDF metric = Term Frequency / Inverse Document Frequency



# Sequence-based String Metrics: String Edit Distance [Levenshtein, 1966]

- Minimum number of character *deletions*, *insertions*, or *substitutions* needed to make two strings equivalent.
  - “misspell” to “mispell” is distance 1 (*delete s*)
  - “misspell” to “mistell” is distance 2 (*delete s*, *substitute p with t* OR *substitute s with t*, *delete p*)
  - “misspell” to “misspelling” is distance 3 (*insert i*, *insert n*, *insert g*)
- Can be computed efficiently using dynamic programming in  $O(mn)$  time where  $m$  and  $n$  are the lengths of the two strings being compared.
- Unit cost is typically assigned to individual edit operations, but individual costs can be used.



# String Edit Distance with Affine Gaps [Gotoh, 1982]

---

- Cost of gaps formed by *contiguous deletions/insertions* should be lower than the cost of multiple non-contiguous operators.
  - Distance from “misspell” to “misspelling” is <3.
- Affine model for gap cost:  $\text{cost}(\text{gap}) = s + e|\text{gap}|$ ,  $e < s$
- Edit distance with affine gaps is more flexible since it is less susceptible to sequences of insertions/deletions that are frequent in natural language text (e.g. ‘Street’ vs. ‘Str’).



# Learnable Edit Distance with Affine Gaps (Bilenko & Moody)

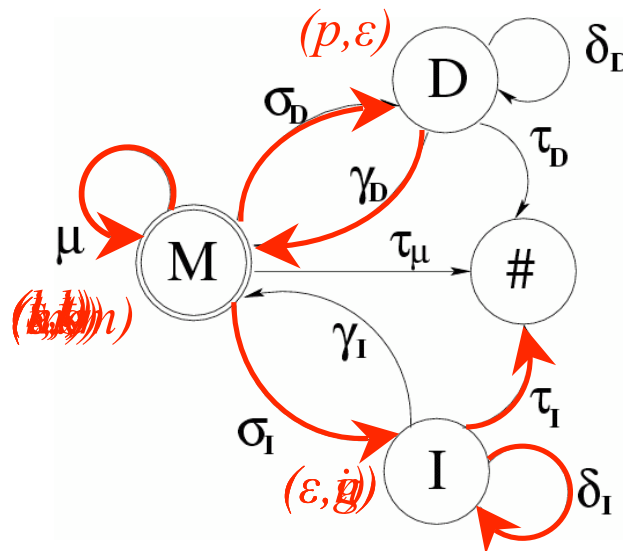
- Motivation:

- **Significance of edit operations depends on a particular domain**

- *Substitute '/' with '-'* insignificant for phone numbers.
    - *Delete 'Q'* significant for names.
    - Gap start/extension costs vary: sequence deletion is common for addresses (*'Street' ▶ 'Str'*), uncommon for zip codes.
  - Using individual weights for edit operations, as well as learning gap operation costs allows adapting to a particular field domain.
  - [Ristad & Yianilos, '97] proposed a one-state generative model for regular edit distance.

# Learnable Edit Distance with Affine Gaps – the Generative Model

**misspell**  
 ↑↑↑↑↑  
**mistelling**



- Matching/substituted pairs of characters are generated in state  $M$ .
- Deleted/inserted characters that form gaps are generated in states  $D$  and  $I$ .
- Special termination state “#” ends the alignment of two strings.
- Similar to pairwise alignment HMMs used in bioinformatics [Durbin *et al.*



# Learnable Edit Distance with Affine Gaps: Training

- Given a corpus of *matched* string pairs, the model is trained using Expectation-Maximization.
- The model parameters take on values that result in high probability of producing duplicate strings.
  - Frequent edit operations and typos have *high* probability.
  - Rare edit operations have *low* probability.
  - Gap parameters take on values that are optimal for duplicate strings in the training corpus.
- Once trained, distance between any two strings is estimated as *the posterior probability of generating the most likely alignment between the strings as a sequence of edit operations.*
- Distance computation is performed in a simple dynamic programming algorithm.



# Learning Transformation Weights for Field Matching

---

- **Synonym:** Robert  $\rightarrow$  {Bob, Robbie, Rob}  $\leftrightarrow$  Rob
- **Acronym:** International Business Machines  $\leftrightarrow$  I.B.M.
- **Misspelling:** Smyth  $\leftrightarrow$  Smith
- **Concatenation:** Mc Donalds  $\leftrightarrow$  McDonalds
- **Prefix/Abbreviation:** Inc  $\leftrightarrow$  Incorporated
- **Suffix:** Reformat  $\leftrightarrow$  format
- **Substring:** Garaparandaseu  $\leftrightarrow$  Paranda
- **Stemming:** George's Golfing Range  $\leftrightarrow$   
George's Golfer Range
- **Levenstein:** the  $\leftrightarrow$  teh

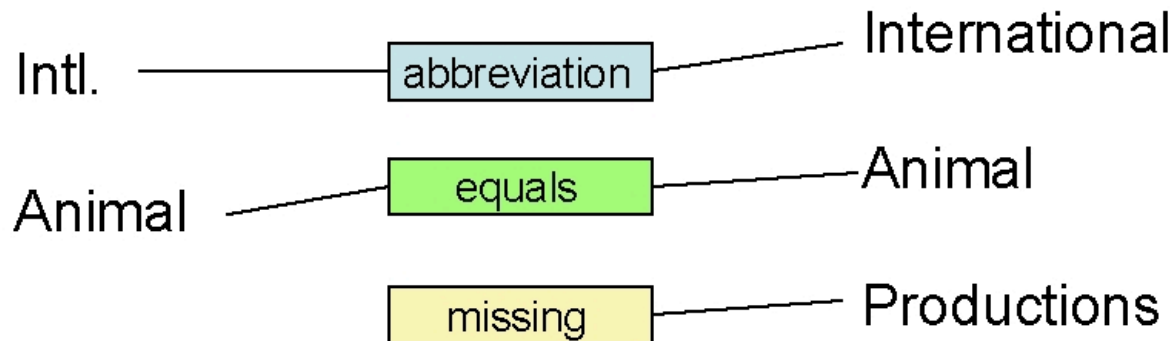
# Training the Field Learner

Transformations =

{ Equal, Synonym, Misspelling, Abbreviation, Prefix, Acronym, Concatenation, Suffix, Soundex, Missing... }

## Transformation Graph

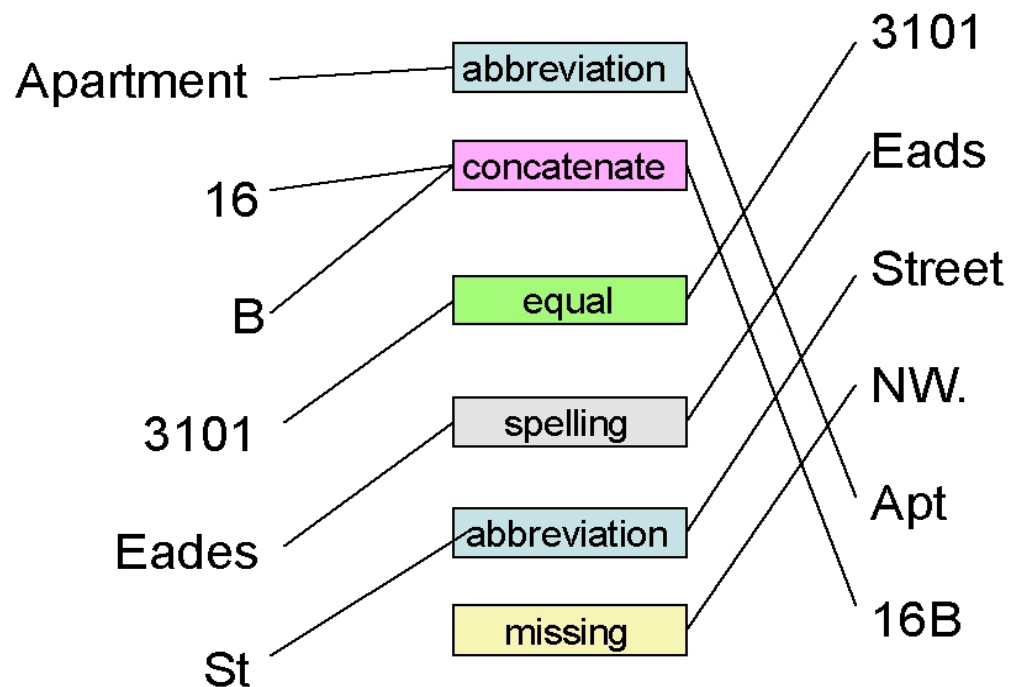
“Intl. Animal” ↔ “International Animal Productions”



# Training the Field Learner

## Another Transformation Graph

“Apartment 16 B, 3101 Eades St”  $\leftrightarrow$  “3101 Eads Street NW Apt 16B”





# Training the Field Learner

## Step 1: Tallying transformation frequencies

---

### Generic Preference Ordering

Equal > Synonym > Misspelling > Missing ...

### Training Algorithm:

- I. For each training record pair
  - i. For each aligned field pair (a, b)
    - i. build transformation graph  $T(a, b)$ 
      - “complete / consistent”
      - Greedy approach: preference ordering over transformations



# Training the Field Learner

## Step 2: Calculating the probabilities

---

- For each transformation type  $v_i$  (e.g. Synonym), calculate the following two probabilities:

$$p(v_i|\text{Match}) = p(v_i|M) = (\text{freq. of } v_i \text{ in } M) / (\text{size } M)$$

$$p(v_i|\text{Non-Match}) = p(v_i|\neg M) = (\text{freq. of } v_i \text{ in } \neg M) / (\text{size } \neg M)$$

- Note: Here we make the Naïve Bayes assumption

# Scoring unseen instances

Naïve Bayes  
assumption

$$p(M | v_1, v_2, \dots, v_n) = \frac{p(M) \prod_{i=1}^n p(v_i | M)}{\prod_{i=1}^n p(v_i)}$$

$$Score_{HFM} = \frac{p(M | \mathbf{V})}{p(M | \mathbf{V}) + p(\neg M | \mathbf{V})}$$

$$= \frac{p(M) \prod_{i=1}^n p(v_i | M)}{p(M) \prod_{i=1}^n p(v_i | M) + p(\neg M) \prod_{i=1}^n p(v_i | \neg M)}$$

# Scoring unseen instances

## An Example

a = “Giovani Italian Cucina Int'l”

b = “Giovani Italian Kitchen International”

$T(a,b) = \{Equal(\text{Giovani}, \text{Giovani}), Equal(\text{Italian}, \text{Italian}),$   
 $Synonym(\text{Cucina}, \text{Kitchen}), Abbreviation(\text{Int'l}, \text{International})\}$

Training:

$$p(M) = 0.31$$

$$p(Equal | M) = 0.17$$

$$p(Synonym | M) = 0.29$$

$$p(Abbreviation | M) = 0.11$$

$$p(\neg M) = 0.69$$

$$p(Equal | \neg M) = 0.027$$

$$p(Synonym | \neg M) = 0.14$$

$$p(Abbreviation | \neg M) = 0.03$$

$$p(M) \prod p(v_i | M) = 2.86E -4$$

$$p(\neg M) \prod p(v_i | \neg M) = 2.11E -6$$

$$\text{Score}_{\text{HFM}} = 0.993 \rightarrow \text{Good Match!}$$



# Outline

---

- Blocking
- Field Matching
- **Record Matching**
- Entity Matching
- Conclusion



# Combining String Similarity Across Fields

---

- Some fields are more indicative of record similarity than others:
  - For addresses, *street address* similarity is more important than *city* similarity.
  - For bibliographic citations, *author* or *title* similarity are more important than *venue* (i.e. conference or journal name) similarity.
- Field similarities should be weighted when combined to determine record similarity.
- Weights can be learned using a learning algorithm [Cohen & Richman '02], [Sarawagi & Bhamidipaty '02], [Tejada *et. al.* '02].



# Record Matching Approaches

---

- Learning Decision Trees
  - Used in Active Atlas (Tejada et al.)
- Support Vector Machines (SVM)
  - Used in Marlin (Bilenko & Moody)
- Unsupervised Learning
  - Used in matching census records (Winkler 1998)

# Learning Mapping Rules with Decision Trees

## Zagat's Restaurants

## Dept. of Health

Name	Street	Phone
<a href="#">Art's Deli</a>	12224 Ventura Boulevard	818-756-4124
<a href="#">Teresa's</a>	80 Montague St.	718-520-2910
<a href="#">Steakhouse The</a>	128 Fremont St.	702-382-1600
<a href="#">Les Celebrities</a>	155 W. 58th St.	212-484-5113

Name	Street	Phone
<a href="#">Art's Delicatessen</a>	12224 Ventura Blvd.	818/755-4100
<a href="#">Teresa's</a>	103 1st Ave. between 6th and 7th Sts.	212/228-0604
<a href="#">Binion's Coffee Shop</a>	128 Fremont St.	702/382-1600
<a href="#">Les Celebrities</a>	160 Central Park S	212/484-5113

### Mapping rules:

**Name > .9 & Street > .87 => mapped**

**Name > .95 & Phone > .96 => mapped**

# Learning Mapping Rules

## Set of Similarity Scores

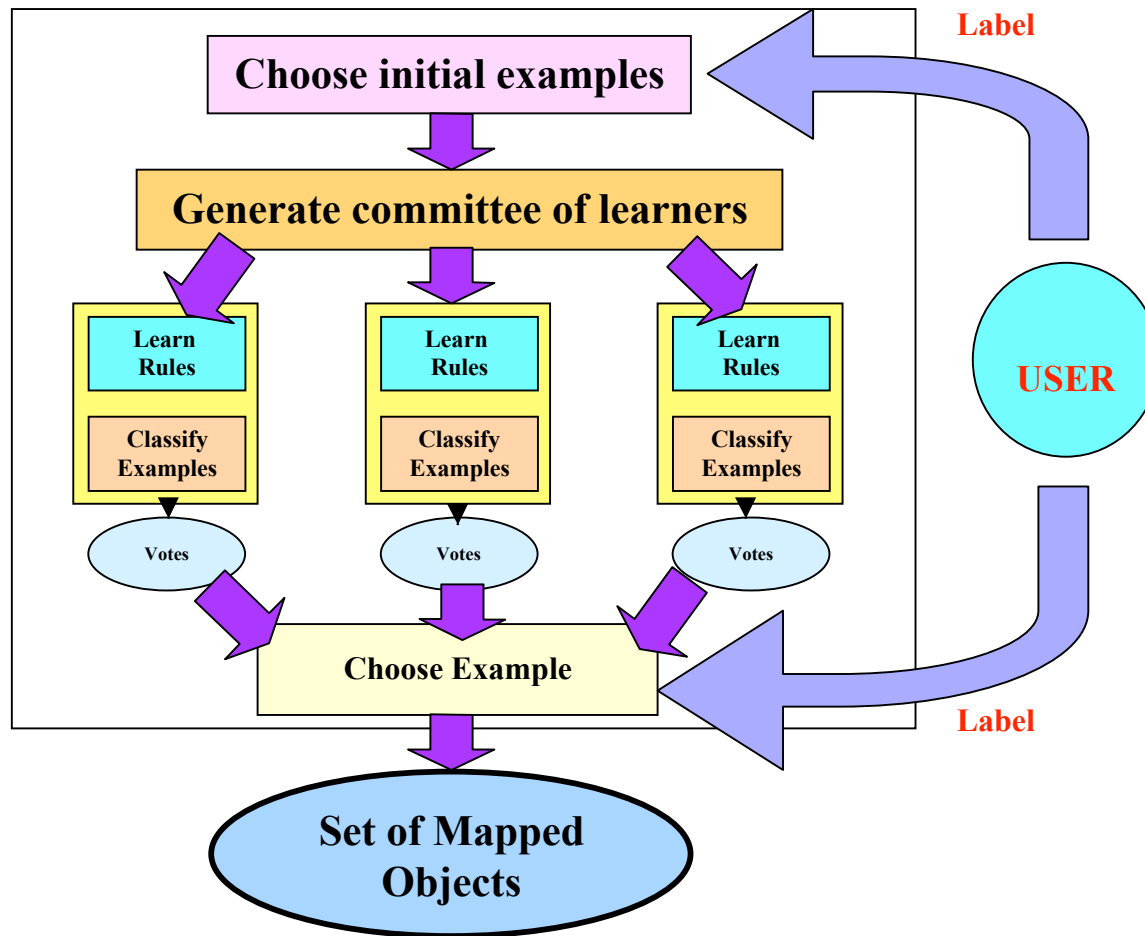
**Name      Street      Phone**

<b>.967</b>	<b>.973</b>	<b>.3</b>
<b>.17</b>	<b>.3</b>	<b>.74</b>
<b>.8</b>	<b>.542</b>	<b>.49</b>
<b>.95</b>	<b>.97</b>	<b>.67</b>
	<b>...</b>	

## Mapping Rules

**Name > .8 & Street > .79 => mapped**  
**Name > .89 => mapped**  
**Street < .57 => not mapped**

# Mapping Rule Learner with Active Learning



# Committee Disagreement

- Chooses an example based on the disagreement of the query committee

Examples	Committee		
	M1	M2	M3
Art's Deli, Art's Delicatessen	Yes	Yes	Yes
CPK, California Pizza Kitchen	Yes	No	Yes
Ca'Brea, La Brea Bakery	No	No	No

- In this case CPK, California Pizza Kitchen is the most informative example based on disagreement



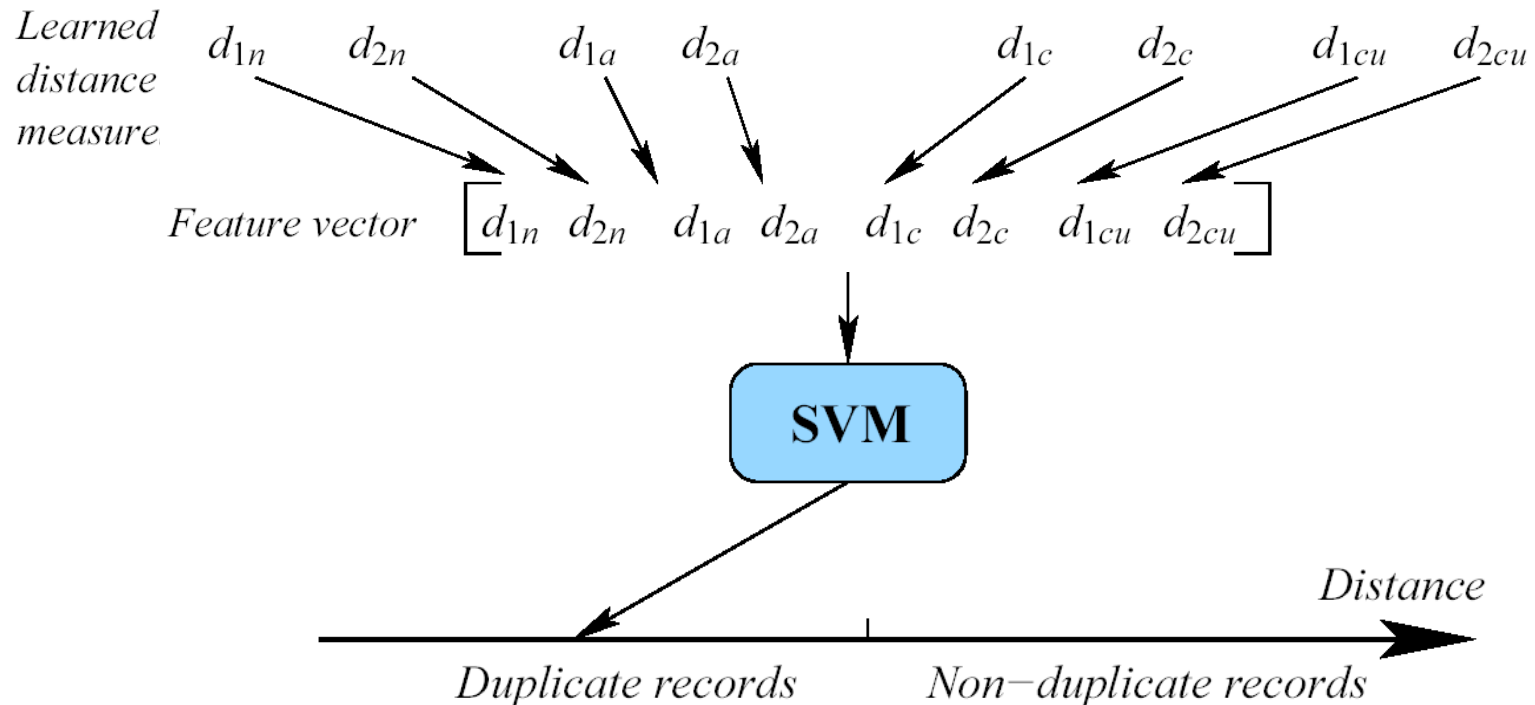
# SVM Learned Record Similarity

---

- String similarities for each field are used as components of a feature vector for a pair of records.
- SVM is trained on labeled feature vectors to discriminate duplicate from non-duplicate pairs.
- Record similarity is based on the distance of the feature vector from the separating hyperplane.

# Learning Record Similarity (cont.)

<i>Name</i>	<i>Address</i>	<i>City</i>	<i>Cuisine</i>
Fenix	8358 Sunset Blvd. West	Hollywood	American
Fenix at the Argyle	8358 Sunset Blvd.	W. Hollywood	French (new)

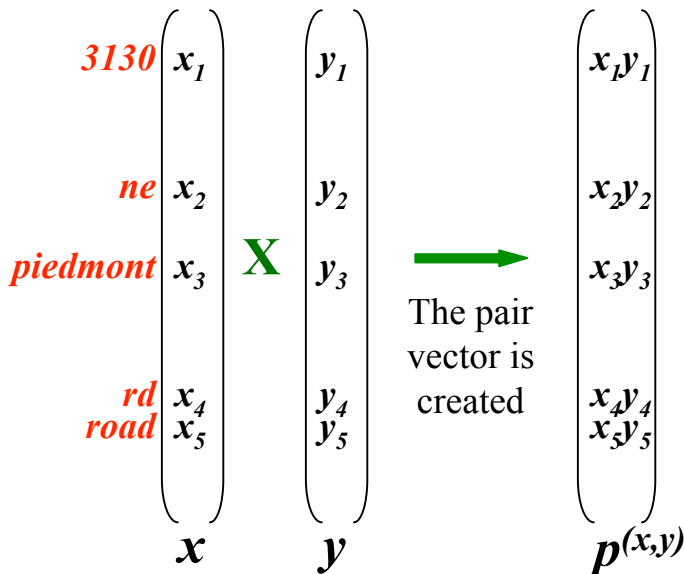


# Learnable Vector-space Similarity

$x$ : "3130 Piedmont Road"

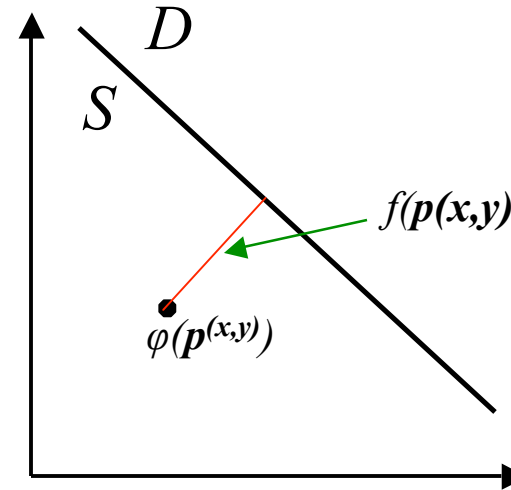
$y$ : "3130 Piedmont Rd. NE"

Each string is converted to vector-space representation



The pair vector is created

The pair vector is classified as "similar" or "dissimilar"



Similarity between strings is obtained from the SVM output

$$Sim(x, y) \propto f(p^{(x,y)})$$



# Unsupervised Record Linkage

---

- Idea: Analyze data and automatically cluster pairs into three groups:
  - Let  $R = P(\text{obs} \mid \text{Same}) / P(\text{obs} \mid \text{Different})$
  - Matched if  $R > \text{threshold } T_U$
  - Unmatched if  $R < \text{threshold } T_L$
  - Ambiguous if  $T_L < R < T_U$
- This model for computing decision rules was introduced by Fellegi & Sunter in 1969
- Particularly useful for statistically linking large sets of data, e.g., by US Census Bureau



## Unsupervised Record Linkage (cont.)

---

- Winkler (1998) used EM algorithm to estimate  $P(\text{obs} \mid \text{Same})$  and  $P(\text{obs} \mid \text{Different})$
- EM computes the *maximum likelihood estimate*. The algorithm iteratively determines the parameters most likely to generate the observed data.
- Additional mathematical techniques must be used to adjust for “relative frequencies”, I.e. last name of “Smith” is much more frequent than “Knoblock”.



# Outline

---

- Blocking
- Field Matching
- Record Matching
- **Entity Matching**
- Conclusion



# EntityBases: Compiling, Organizing and Querying Massive Entity Repositories

---

- Today:
  - Lots of data & documents available
  - NLP technology for extracting simple entities & facts
- Opportunity: Collect and query billions of facts about millions of entities (e.g., people, companies, locations, ...)



August 11, 2002

## DUBAI TRADER AND IRANIAN COMPANY OFFICER INDICTED FOR SCHEME TO ILLEGALLY SHIP U.S. GOODS TO IRAN

Washington, D.C. - United States Attorney Kenneth L. Wainstein, Department of Homeland Security Assistant Secretary for Immigration and Customs Enforcement (ICE) Michael J. Garcia, U.S. Department of Commerce Undersecretary for Industry and Security Kenneth I. Juster, and jointly announced that a federal grand jury in the District of Columbia has returned an indictment against Khalid Mahmood, also known as Khalid Mahmood Chaudhary, 52, of Dubai, United Arab Emirates, and [redacted] age unknown, of Iran, with violations of the International Economic Powers Act, the Iranian Transactions Regulations, and the Export Administration Regulations.

The indictment alleges that Mahmood was doing business as Sharp Line Trading with offices in the United Arab Emirates, and Sherbatf was a principal officer of [redacted] a forklift manufacturing firm located in Iran. According to the indictment, in early June 2004, an employee of Sepahan Lifter Company contacted a United States company by email, and requested a price quotation for particular radiators for heavy-duty 5-ton capacity forklift trucks manufactured in Iran by

February 24, 2006

## LEXINGTON, KENTUCKY MAN SENTENCED TO 39 MONTHS IN PRISON FOR VIOLATING TRADE EMBARGO AGAINST IRAN

Washington, D.C. - United States Attorney Kenneth L. Wainstein, Darryl W. Jackson, United States Department of Commerce Assistant Secretary for Export Enforcement, and Mark Gerrand, Acting Special Agent-in-Charge for U.S. Immigration and Customs Enforcement (ICE), Department of Homeland Security, announced yesterday that the Honorable John D. Bates sentenced Robert E. Quinn, 54, of Lexington, Kentucky, to 39 months of incarceration and a fine of \$6,000. Quinn was found guilty by a federal jury on December 7, 2005, of one count of conspiring to violate the U.S. trade embargo against Iran and five counts of illegal exports to Iran.

In October 2005, a federal grand jury in the District of Columbia returned a six-count superseding indictment against Quinn, 54, and Michael H. Holland, also of Lexington, Kentucky, and [redacted] of Iran, charging them with violating the United States embargo on trade with Iran. Quinn and Holland were sales executives employed by Clark Material Handling Corporation ("CMHC"), a Kentucky-based forklift truck manufacturer. Sherbatf is President and Managing Director of [redacted] a forklift truck manufacturer in Esfahan, Iran.



# The Idea

---

- ***EntityBases: Large-scale, organized entity knowledgebases***
  - composed of billions of facts/millions of entities
- **Each Entitybase:**
  - *Aggregates* information about a single entity type
    - e.g. PeopleBase, CompanyBase, AssetBase, GeoBase, ...
    - Simple representation, broad coverage.
  - *Integrates* data collected from numerous, heterogeneous sources
  - *Consolidates* data, resolving multiple references to the same entity
- **Requires scalable algorithms and tools to *populate, organize* and *query* massive EntityBases**

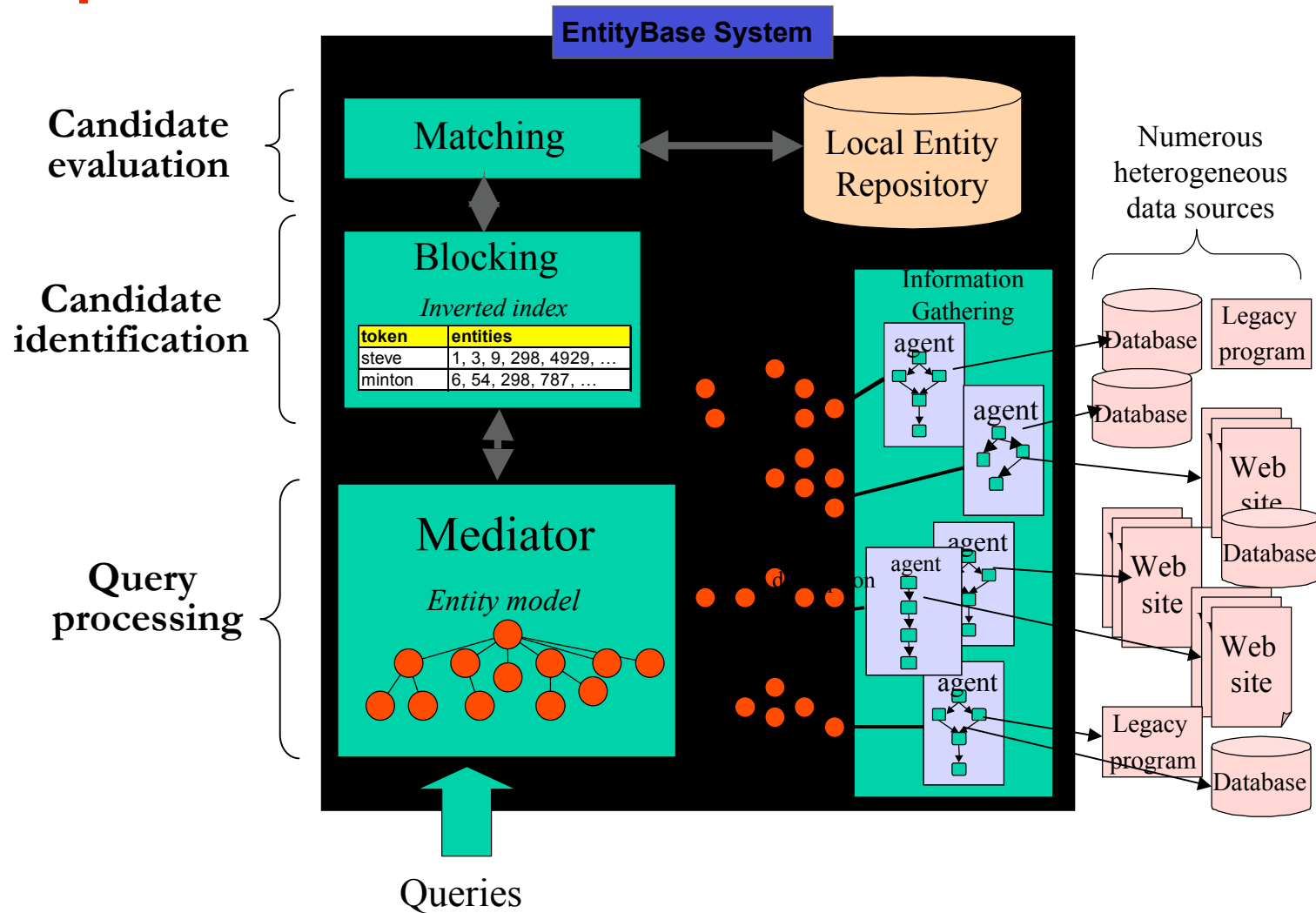


# Why this is a Hard Problem: Limitations of Previous Work

---

- EntityBase is not a straightforward extension of past work on data integration and record linkage
- New challenges include:
  - Real-world entities have attributes with multiple values
    - Ex: name: maiden name, transliterations, aliases, ...
    - Previous work only dealt with records with single values for attributes (e.g., a single name, phone number, address, etc.)
  - Need to link arbitrary number of sources, with different schemas
    - Most previous work on record linkage focused on merging two tables with similar schemas
  - In addition, real-time response must be considered
- We had to extend previous work on both data integration and record linkage to support massive-scale entity bases
  - Without compromising efficiency!

# EntityBase Architecture



# Data Gathering

- Sample Linkable Company Sources
  - Kompass
  - Ameinfo
- Used Fetch Agent Platform

**KOMPASS**  
38m product and service references in 53000 classes  
19m companies in 70 countries  
50 000 trade names  
15m executive names

Products & services Company names Trade names Executives **Advanced search**

Search Text: abfar Region: Worldwide

**Company result list**

full company profile e-mail web link showcase Catalogue

Suppliers	Address
<b>Abfar</b>	11365 Tehran [Ir]
Abfar Ilac Sanayi ve Ticaret	34330 Istanbul [T]

**AME Info**  
The ultimate Middle East business resource

Already a Member of Saxo bank? **LOGIN HERE** Not a member? Sign in here

GET A FREE DOWNLOAD OF SAXOTRADER  
INSTANT ACCESS TO ALL PRODUCTS AND SERVICES

Index : Wholesale Trade : Merchant Wholesalers, Durable Goods : Lumber and Other Construction Materials Merchant Wholesalers

Browse companies in this category << Previous Next >>

**\$2,000 to \$7,000 Per Week**  
Begin Immediately - No Exp. Nec. Get Paid at Any ATM

**International City Dubai**  
Luxury 1 Bed from £49,995 U.K's Premier Site For Dubai Homes

Ads by Google Advertise on this site

**Abfar Public Corporation**

POBox 11365-3644  
18th Km, Jadeh Makhsos Karaj.  
Tehran  
Iran

Telephone: +98(262)3834035  
Facsimile: +98(262)3834033  
Website: <http://www.abfar.com>

Login to edit this entry

**Business Activities**  
Building Equipment & Materials

**Industry Classifications - NAICS**  
» Lumber and Other Construction Materials Merchant Wholesalers

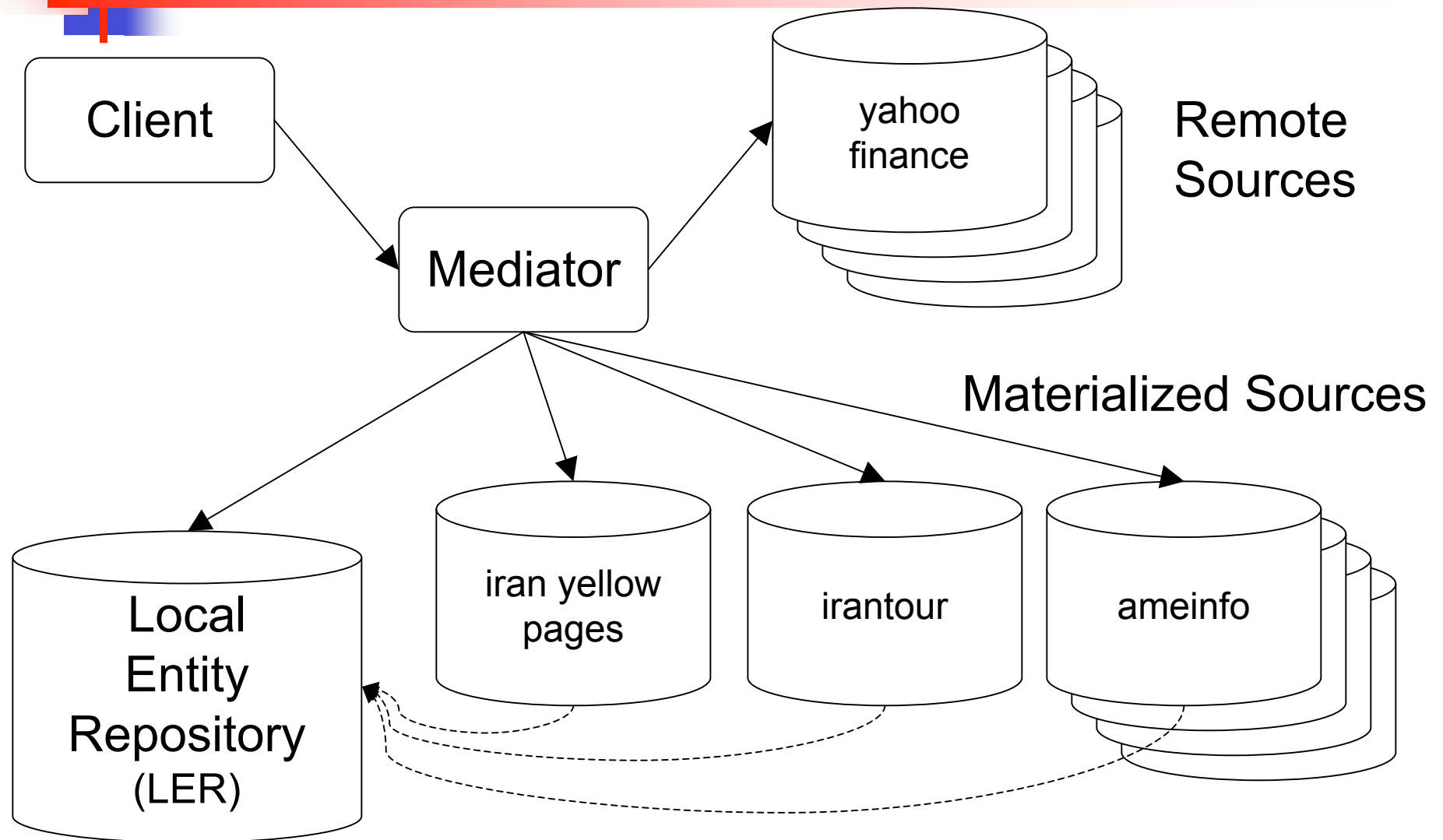


# EntityBase Integration Architecture

---

- **Local Entity Repository (LER):**
  - stores entity identifying attributes
  - record linkage reasoning on these attributes
- **Materialized Sources**
  - Entity-identifying attributes fed into core entity base
  - Additional attributes materialized, but not copied into LER for performance
- **Remote Sources**
  - Cannot be materialized due to organizational constraints, security, or rapid data change
- **Mediator**
  - Assigns common semantics to data from LER and sources
  - Integrates data from LER and sources in response to user queries

# EntityBase Integration Architecture





# Data Representation Approach

---

- EntityBase uses a **Mediated Schema** to integrate data from different sources
  - Assigns common semantics
  - Handle multiple values
  - Normalizes/Parses values
  - Object-relational representation
  - General, but still efficient for record linkage processing



# Mediated Schema

---

- **Entity:** main object type
  - Ex: Company
- Each entity has several *multi-valued* attributes (units):
  - Ex: name, address, phone, keyperson, code (ticker, D&B DUNS, ISIN,...), email, product/service, ...
- **Unit:** structured (sub)object with *single-valued* attributes
  - Ex: address( FullAddress, StreetAddress, City, Region, PostalCode, Country, GeoExtent)
- Some units extended with geospatial extent
  - Ex: address, phone

# LER

## entity

EID	unit	RID
7	name	14
7	name	56
7	address	21
7	address	22
7	address	23

## address

RID	Src	SRID	FullAddress	StreetAddress	City	Region	Postal Code	Country	Geo Extent
21	s1	3	PO Box 1000 El Segundo CA 90245	PO Box 1000	El Segundo	CA	90245	USA	SDO_GEO METRY(...)
22	s1	3	2041 Rosecrans Ave El Segundo CA 90245	2041 Rosecrans Ave	El Segundo	CA	90245	USA	SDO_GEO METRY(...)
23	s2	10	CA 90245	null	null	CA	90245	USA	SDO_GEO METRY(...)

RID	Source	name	SRID
14	s2	Fetch	10
56	s1	Fetch Technologies	3

## name

## s2

SRID	Name	ZIPState	naics	#emp
10	Fetch	90245 CA	1234	45

## Sources

## s1

SRID	Name	Office street	Office city	Office State	Office Zip	Factory Street	Factory City	Factory State	Factory zip	Ceo	Cto
3	Fetch Technologies	PO Box 1000	El Segundo	CA	90245	2041 Rosecrans Ave	El Segundo	CA	90245	Robert Landes	Steve Minton

#emp not in LER



# Importing Data from DB / Fetch Agents

---

- Import Rule for address:

**address**( RID, Source, SRID, **FullAddress**, **StreetAddress**, **City**, PostalCode, **Country**, **GeoExtent**) :-

**IranYellowPages**( Name, ManagingDirector, CommercialManager, CentralOffice, OfficeTelephone, OfficeFax, OfficePOBox, Factory, FactoryTelephone, FactoryFax, FactoryPOBox, Email, WebAddress, ProductsServices, SRID) ^

**ParseAddress**( Centraloffice, **StreetAddress**, **City**) ^

**Concat**( StreetAddress, City, Region, OfficePOBox, Country, **FullAddress**) ^

**ComputeGeoextent**( FullAddress, **GeoExtent**) ^

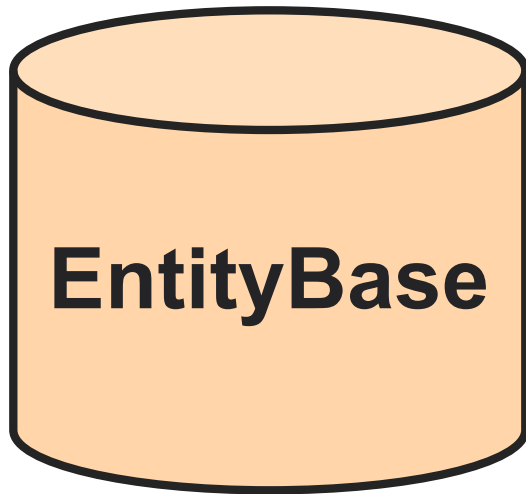
**GenRID**( SRID, Source, "1", RID) ^ (Source = "IranYellowPages ") ^  
(OfficePOBox = PostalCode) ^ (**Country** = "Iran")

# Entity linkage

News article

..., said A. Baroul of Tehran-based Adaban, ....

Find the company that most closely matches the case where company name ~ "Adaban", address ~ "Tehran", key person ~ "A. Baroul"



EID	Company	City	Country	Key person
5640	Adaban Intl Transport	Tehran	Iran	Parviz Toorani
109	Adaban Petrochemical	Tehran	Iran	Ahmad Baroul
71	Kavian Industrial	Tehran	Iran	Taghi Baroul
89276	Adaban Partners	Dublin	Ireland	Alex Nasser

① Quickly identify promising candidates from large EB

② Closely judge candidates

③ ...and the best match is:

Score	EID	Company	City	Country	Key person
99.5%	109	Adaban Petrochemical	Tehran	Iran	Ahmad Baroul

Craig Knoblock

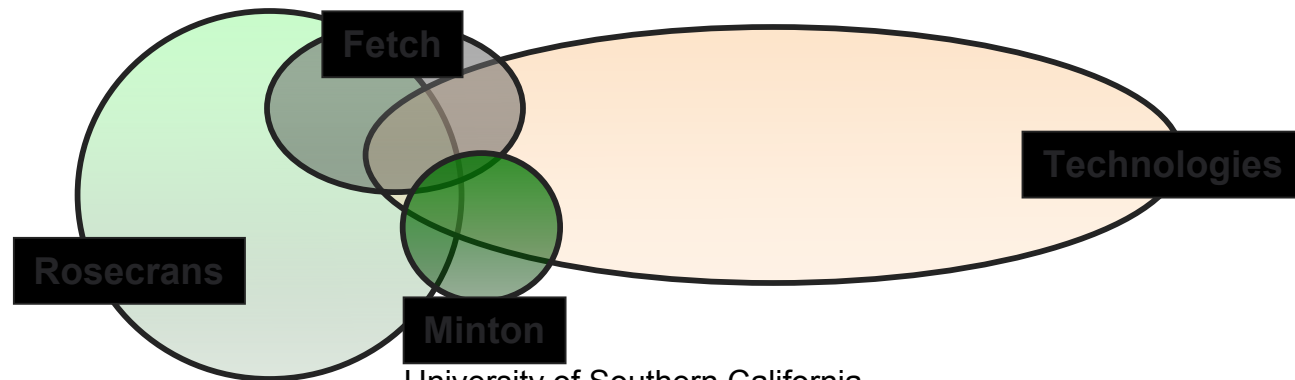
University of Southern California

51

# Efficient blocking

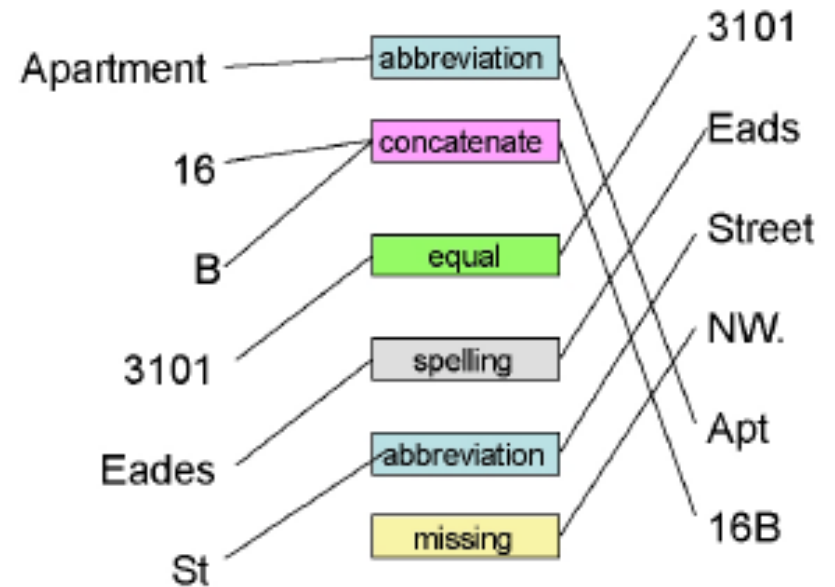
Want to quickly identify promising candidates

- But...
  - We need to use **fast** comparison methods
    - e.g., string or word ID comparisons
    - edit distance computations are likely too expensive
  - We are working with **many** potential entities
    - Do not want to return too large of a block size (will impact RL perf)
- Core issue
  - Computing set intersections / unions efficiently
  - Novel Union/Count Algorithm

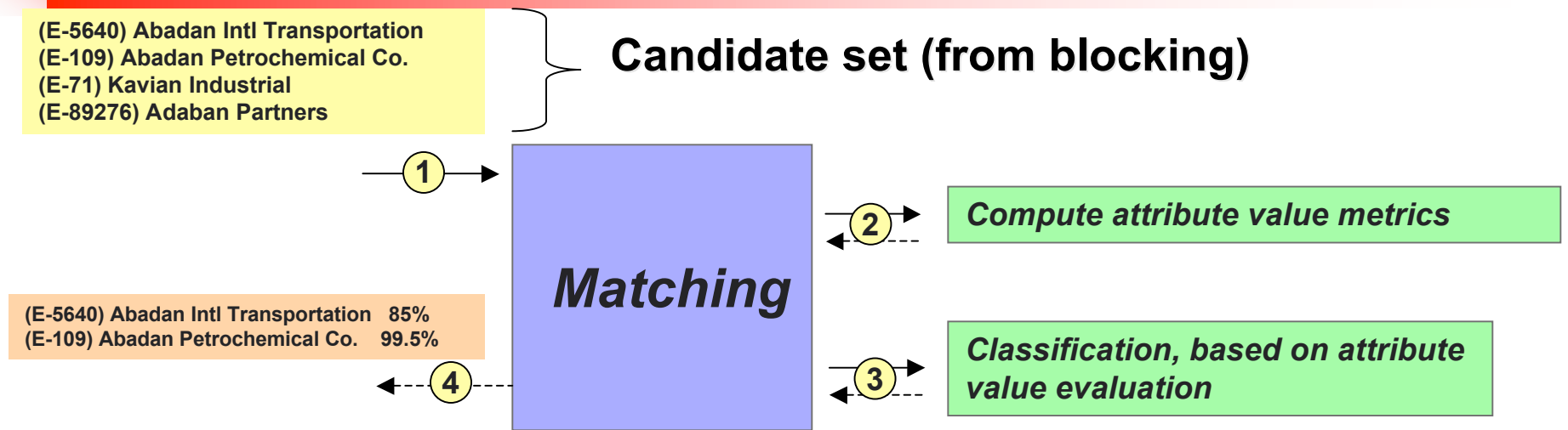


# Intelligent Field Matching with Transformations

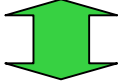
- Transformations relate two values and provide a more precision definition of how well they match
- Using fine-grained transformations in the matching phase increases accuracy.



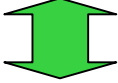
# Matching



- Classifier judges the importance of the combination of attribute value metrics
  - e.g., complete mismatch on address may be offset by strong matches on phone and name

Steve Minton  
  
 Steven N Minton  
 Craig Knoblock

Fletch Software  
  
 Fetch Technologies  
 University of Southern California

El Segundo, CA  
  
 El Segundo



# Outline

---

- Blocking
- Field Matching
- Record Matching
- Entity Matching
- **Conclusion**



# Related Work

---

- Record linkage [Newcombe *et al.* '59; Fellegi & Sunter '69; Winkler '94, '99, '02]
- Database hardening [Cohen *et al.* '00]
- Merge/purge [Hernandez & Stolfo '95]
- Field matching [Monge & Elkan '96]
- Data cleansing [Lee *et al.* '99]
- Name matching [Cohen & Richman '01, Cohen *et al.* '03]
- De-duplication [Sarawagi & Bhamidipaty '02]
- Object identification [Tejada *et al.* '01, '02]
- Fuzzy duplicate elimination [Ananthakrishna *et al.* '02]
- Identity uncertainty [Pasula *et al.* '02, McCallum & Wellner '03]
- Object consolidation [Michalowski *et al.* '03]



# Conclusions

---

- Technical choices in record linkage:
  - Approach to blocking
  - Approach to field matching
  - Approach to record matching
  - Is the matching done pairwise or based on entities
- Learning approaches have the advantage of being able to
  - Adapt to specific application domains
  - Learn which fields are important
  - Learn the most appropriate transformations
- Optimal classifier choice is sensitive to the domain and the amount of available training data.