

Learning to Align Objects Across Sources

Craig Knoblock
University of Southern California

Thanks to Sheila Tejada and Steve Minton

Object Identification (aka Record Linkage)

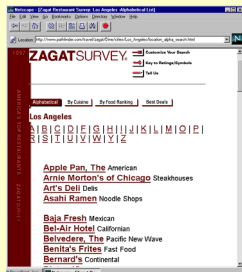
Problem

- Different sources typically represent and format information differently.
- As a result, determining if two sources are referring to the same object can be difficult.
- For example, is “Joe Cool” the same person as “Joseph B. Cool”?
- What if they have the same telephone number?
- What if Joe Cool’s number is 310-322-0730 and Joseph B. Cool’s number is 310-640-2973?

Integrating Restaurant Sources

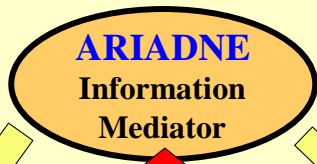
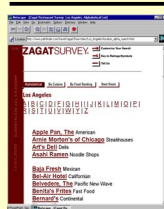
Zagat's Restaurant Guide Source

Department of Health Restaurant Rating Source



Question: What is the **Review and **Rating** for the Restaurant "Art's Deli"??**

Ariadne Information Mediator



Zagat's Wrapper

User Query

Dept. of Health Wrapper

Extract web objects in the form of database records

Zagat's

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

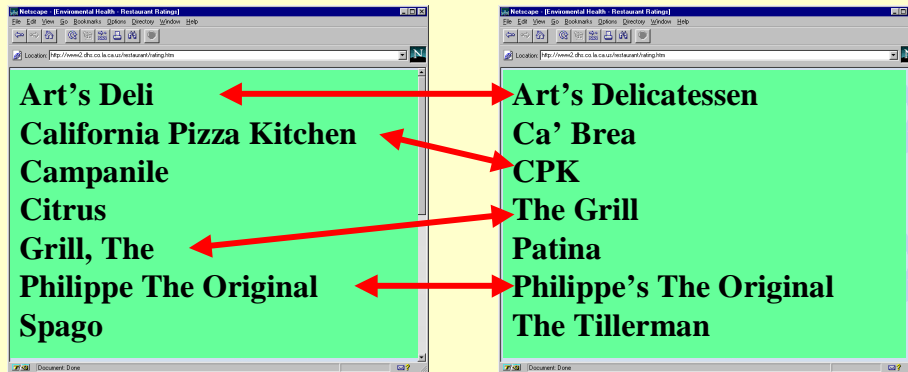
Dept of Health

Name	Street	Phone
Art's Delicatessen	12224 Ventura Blvd.	818/755-4100
Teresa's	103 1st Ave. between 6th and 7th Sts.	212/228-0604
Binion's - Coffee Shop	128 Fremont St.	702/382-1600
Les Celebrities	5432 Sunset Blvd	212/484-5113

Multi-Source Inconsistency

Zagat's Restaurant Guide Source

Department of Health Restaurant Source



How can the same objects be identified when they are stored in inconsistent text formats?

Application Dependent Mapping

Observations:

- Mapping objects can be application dependent
- Example:

Mapped?

[Steakhouse The](#) 128 Fremont Street 702-382-1600

[Binion's Coffee Shop](#) 128 Fremont St. 702/382-1600

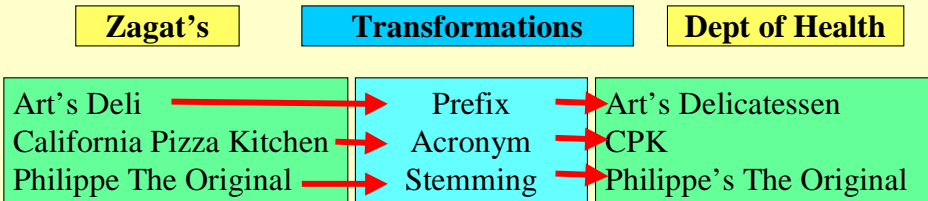
- The mapping is in the application, not the data
- User input is needed to increase accuracy of the mapping

Key Ideas for Mapping Objects

- Learning important attributes for determining a mapping

	Name	Street	Phone
Zagat's	Art's Deli	12224 Ventura Boulevard	818-756-4124
Dept of Health	Art's Delicatessen	12224 Ventura Blvd.	818/755-4100

- Learning general transformations to recognize objects



Mapping Rules

Zagat's Restaurants			Dept. of Health		
Name	Street	Phone	Name	Street	Phone
Art's Deli	12224 Ventura Boulevard	818-756-4124	Art's Delicatessen	12224 Ventura Blvd.	818/755-4100
Teresa's	80 Montague St.	718-520-2910	Teresa's	103 1st Ave. between 6th and 7th Sts.	212/228-0604
Steakhouse The	128 Fremont St.	702-382-1600	Binion's Coffee Shop	128 Fremont St.	702/382-1600
Les Celebrities	155 W. 58th St.	212-484-5113	Les Celebrities	160 Central Park S	212/484-5113

Mapping rules:

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

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

Transformation Weights

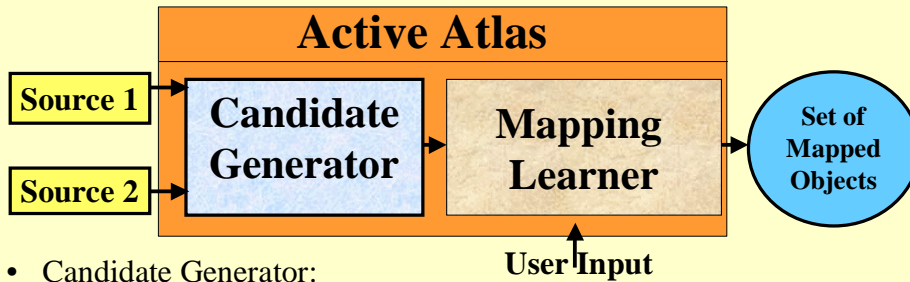
- Transformations can be more appropriate for a specific application domain
 - Restaurants, Companies or Airports
- Or for different attributes within an application domain
 - Acronym more appropriate for the attribute Restaurant Name than for the Phone attribute
- Learn likelihood that if transformation is applied then the objects are mapped

$$\text{Transformation Weight} = P(\text{mapped} \mid \text{transformation})$$

Overview

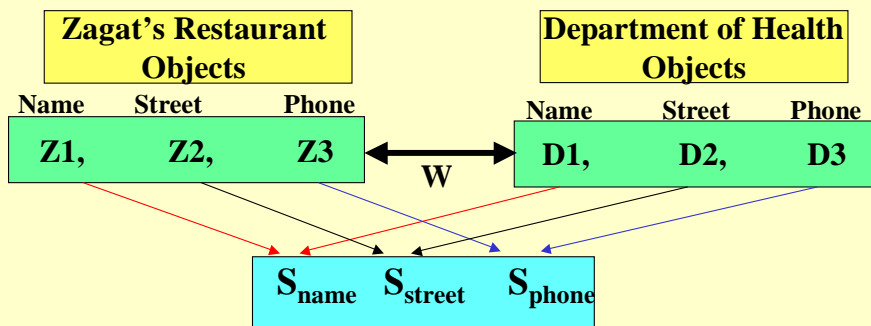
- Approach
 - Computing textual similarity
 - Learning important attributes for mapping
 - Mapping rule learning
 - Learning transformation weights
- Experimental Results
- Related Work on Object Identification
- Conclusions & Future Work

Learning Object Mappings



- Candidate Generator:
 - Judge textual similarity of mappings
 - Reduce number of mappings considered for classification
- Mapping Learner:
 - Active learning technique to learn mapping rules and transformation weights
 - Minimize the amount of user interaction

Computing Textual Similarity



- Candidate Generator returns **sets of similarity scores**

Name	Street	Phone
.9	.79	.4
.17	.3	.74
	...	

Types of Transformations

Type I Transformations

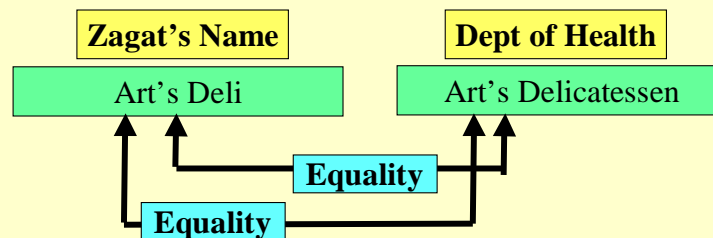
- Equality (Exact match)
- Stemming
- Soundex (e.g. “Celebrities” => “C453”)
- Abbreviation (e.g. “3rd” => “third”)

Type II Transformations

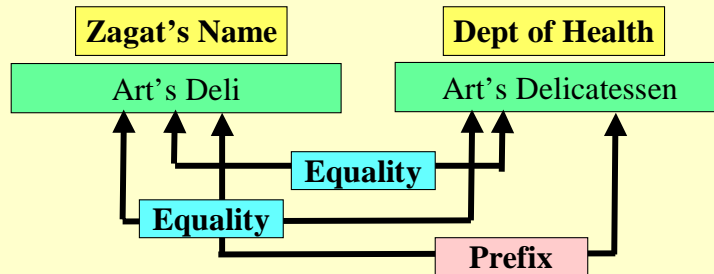
- Initial
- Prefix (e.g. “Deli” & “Delicatessen”)
- Suffix
- Substring
- Acronym (e.g. “California Pizza Kitchen” & “CPK”)
- Drop Word

Applying Type I Transformations

- Employs Information Retrieval Techniques
- One set of attribute values broken into words or tokens
 - “Art” “s” “Delicatessen”
- Apply Type I transformations to tokens
 - “Art” “A630” “s” “S000” “Delicatessen” “D423”
- Enter tokens into inverted index
- Tokens from second set used to query the index
 - Transformed query set: “Art” “A630” “s” “S000” “Deli” “Del” “D400”



Applying Type II Transformations



- Type II transformations improve measurement of similarity

Attribute Similarity Function

- Transformations determine similarity of attribute values
- Each attribute value is represented as a vector
< 2 4 3 0 5 6 6 0 0 0 0 5 0 0 0 0 ... >
- Attribute Similarity Function:

- Cosine Measure with a TFIDF

Similarity (A, B) =

$$\frac{\sum_{i=1}^t (w_{ia} \times w_{ij})}{\sqrt{\sum_{i=1}^t (w_{ia})^2 \times \sum_{i=1}^t (w_{ij})^2}}$$

$$w_{ia} = (0.5 + 0.5 \text{freq}_{ia}) \times \text{IDF}_i$$

$$w_{ij} = \text{freq}_{ij} \times \text{IDF}_i$$

freq_{ia} = frequency of term i for attribute value a

IDF_i = IDF of term i in the entire collection

freq_{ij} = frequency of term i in attribute value j

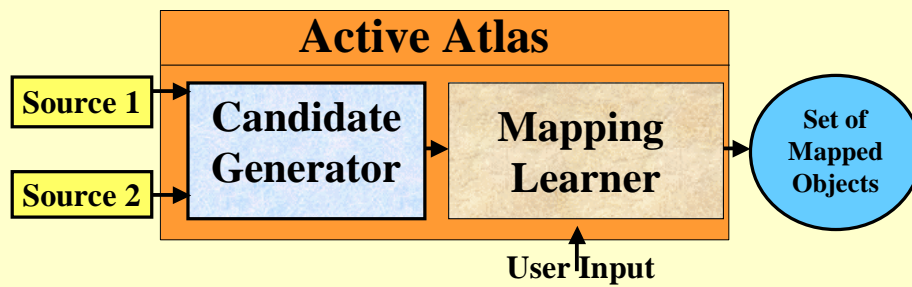
Total Object Similarity Scores

	Name	Street	Phone
Zagat's	Art's Deli	12224 Ventura Boulevard	818-756-4124
Dept of Health	Art's Delicatessen	12224 Ventura Blvd.	818/755-4100

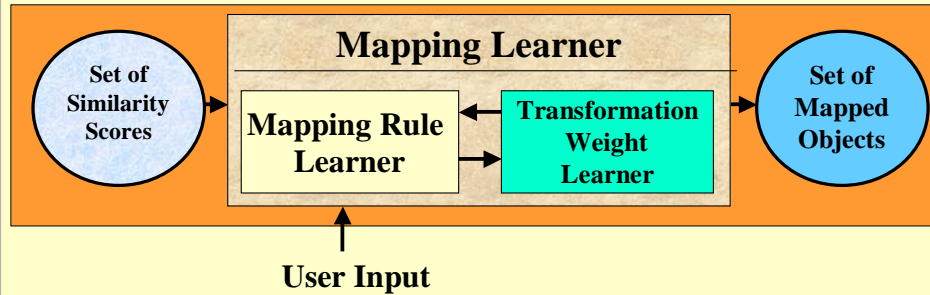
Candidate Mapping Similarity Scores:

Name	Street	Phone	Total Score
.967	.973	.3	2.034
.17	.3	.74	1.182
.8	.5	.49	1.749
	...		

Learning Object Mappings



Learning Object Mappings



- The goal is to classify with high accuracy the proposed mappings while minimizing user input
 - Active learning technique
 - System chooses most informative example for the user to label

Mapping Rules

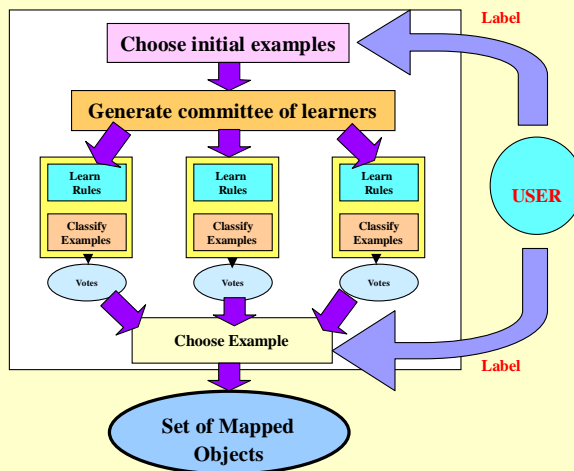
Set of Similarity Scores

Name	Street	Phone
.967	.973	.3
.17	.3	.74
.8	.542	.49
.95	.97	.67
	...	

Mapping Rules

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

Mapping Rule Learner



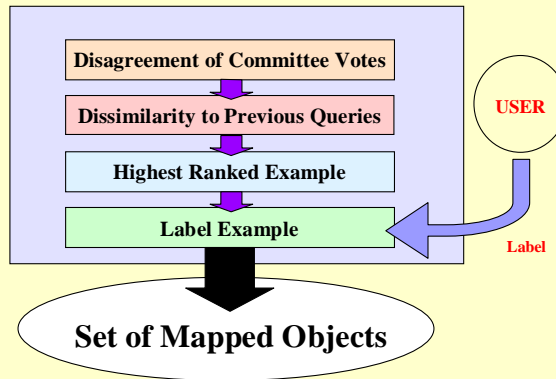
Committee Disagreement

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

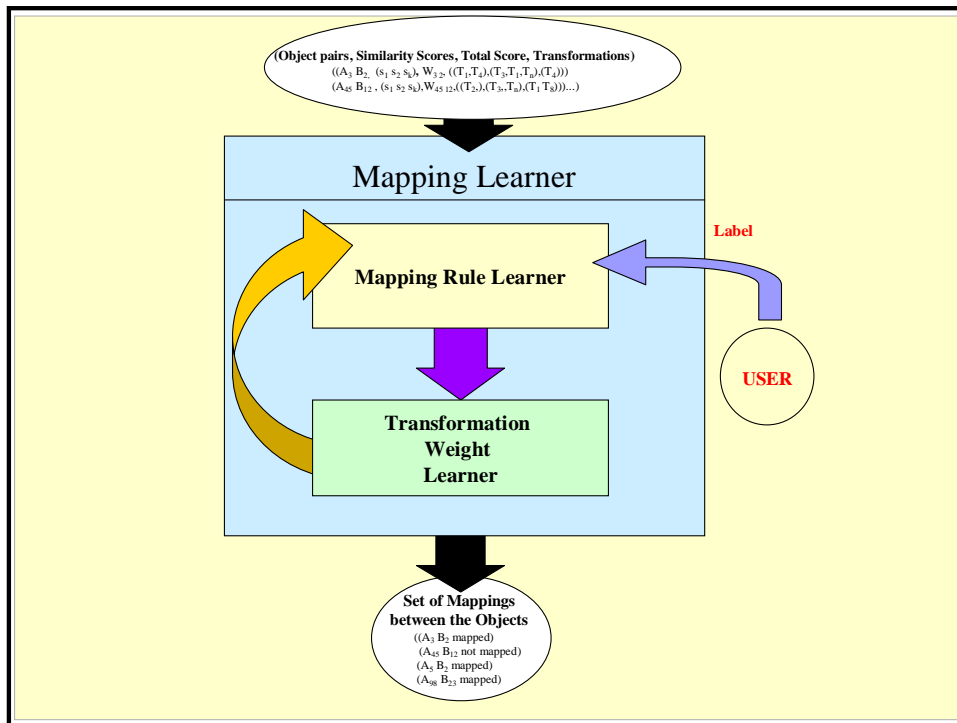
Examples	Committee		
	M1	M2	M3
Art's Deli, Art's Delicatessen	Yes	Yes	Yes
CPK, California Pizza Kitchen	Yes	No	Yes
Ca'Brea, La Brea Bakery	No	No	No

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

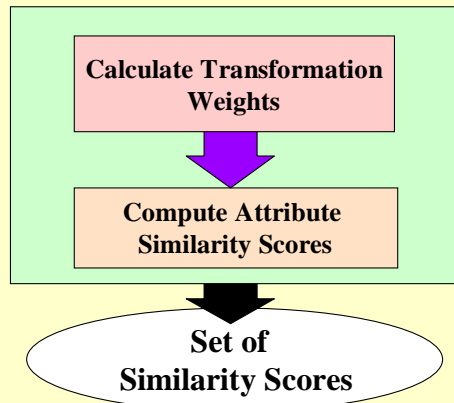
Choosing Next Example



- The user labels the example, and the system updates the committee
- Mapping Rule Learner outputs classified examples



Transformation Weight Learner



Calculate Transformation Weights

$$P(\text{mapped} \mid \text{transformation}) =$$

$$\frac{P(\text{transformation} \mid \text{mapped}) P(\text{mapped})}{P(\text{transformation})}$$

Examples	Classification	Labeled by
Art's Deli, Art's Delicatessen	Mapped	Learner
CPK, California Pizza Kitchen	Mapped	User
Ca'Brea, La Brea Bakery	Not Mapped	Learner

Recalculating Similarity Scores

Transformation	Mapped	Not Mapped
(EQUAL "Art" "Art")	.8	.2
(EQUAL "s" "s")	.8	.2
(PREFIX "Deli" "Delicatessen")	.1	.9

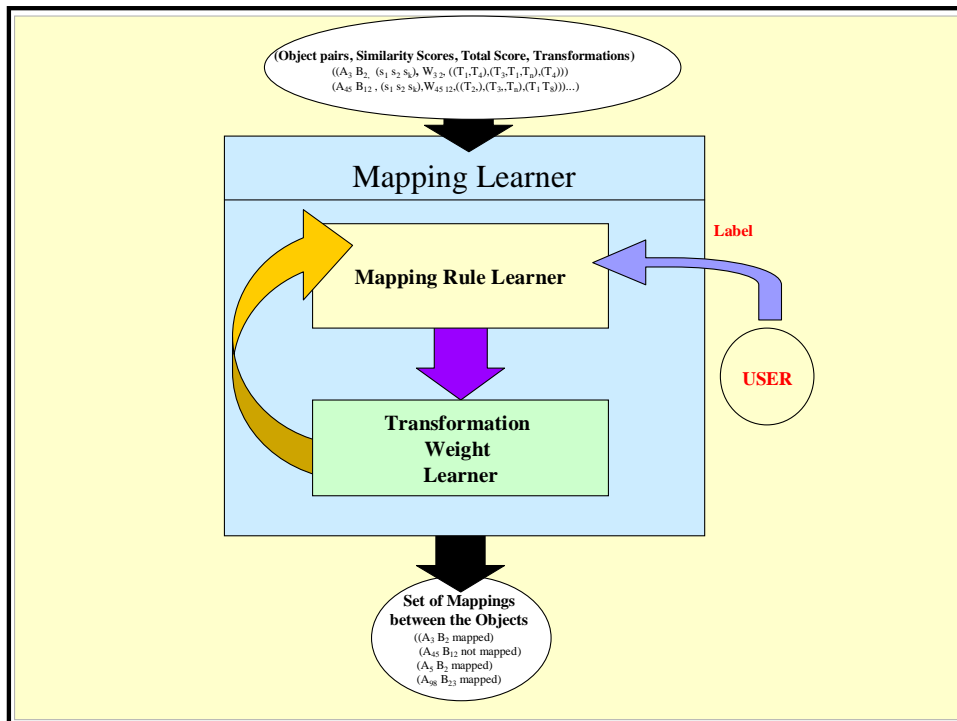
Total mapped score $m = .064$

Total not mapped score $n = .004$

Normalized Attribute Similarity Score = $m/(m + n)$

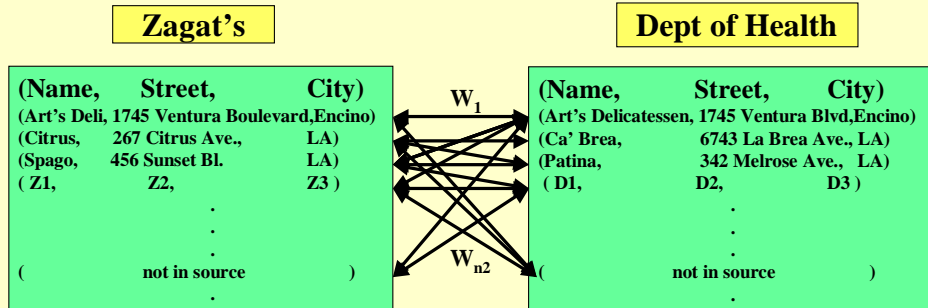
= $.064 / (.064 + .004)$

Attribute Similarity Score = $.941$



Enforcing One-to-One Relationship

- Viewed as weighted bipartite matching problem



Given weights W , matching method determines mostly likely Matching Assignment

Experimental Results

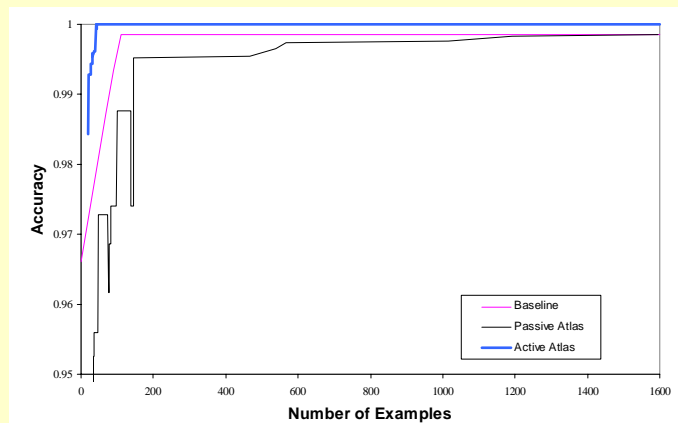
- Three domains: Restaurant, Company, Airport
- Three types of experiments
 - Active Atlas (Mapping Learner)
 - CG — Mapping Learner
 - Passive Atlas (Decision tree learner)
 - CG — Decision tree learner
 - Candidate Generator (Baseline)
 - CG (only Stemming)
- Three Variations of Active Atlas
 - Without Transformation weight learning
 - Without using Dissimilarity for choosing queries
 - Without enforcing One-to-One Relationship
- Learned Weights and Rules

Restaurant Domain

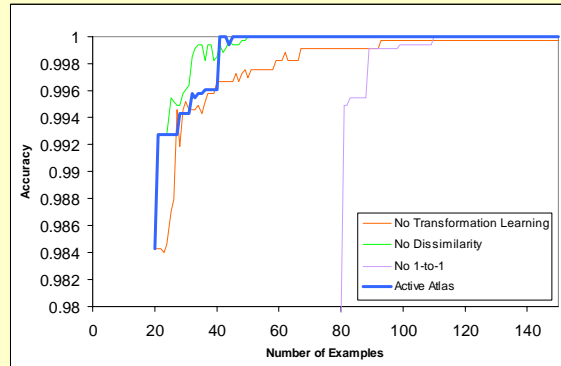
Zagat's Restaurants			Dept. of Health		
Name	Street	Phone	Name	Street	Phone
Art's Deli	12224 Ventura Boulevard	818-756-4124	Art's Delicatessen	12224 Ventura Blvd.	818/755-4100
Teresa's	80 Montague St.	718-520-2910	Teresa's	103 1st Ave. between 6th and 7th Sts.	212/228-0604
Steakhouse The	128 Fremont St.	702-382-1600	Binion's Coffee Shop	128 Fremont St.	702/382-1600
Les Celebrities	155 W. 58th St.	212-484-5113	Les Celebrities	160 Central Park S	212/484-5113

112 mapped objects / 3310 mappings proposed

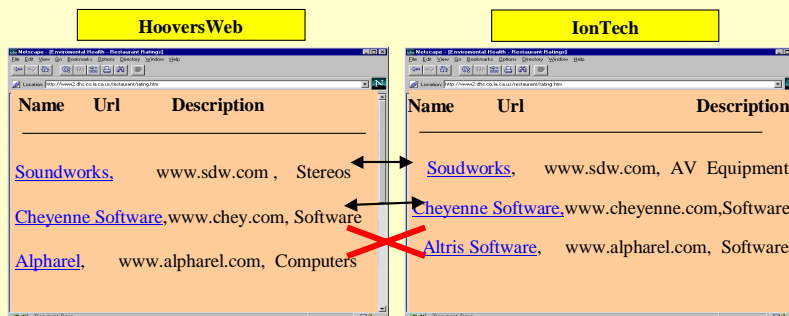
Restaurant Results



Active Atlas Results

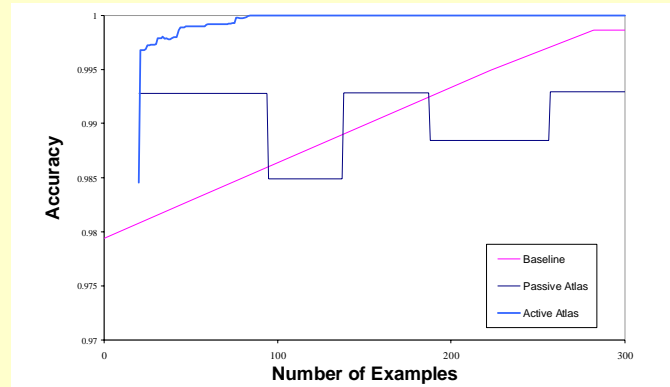


Company Domain

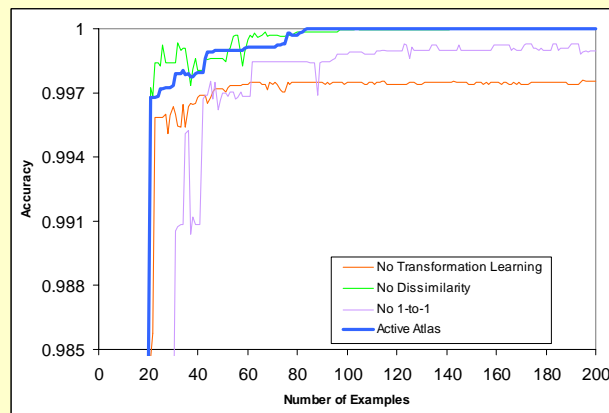


294 mapped objects / 14303 mappings proposed

Company Results



Active Atlas Results

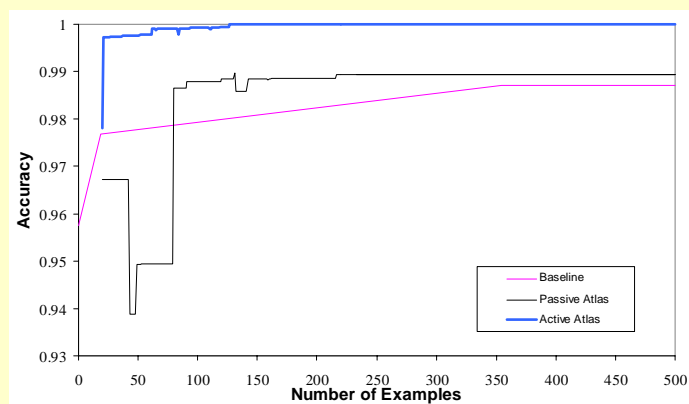


Airport/Weather Domain

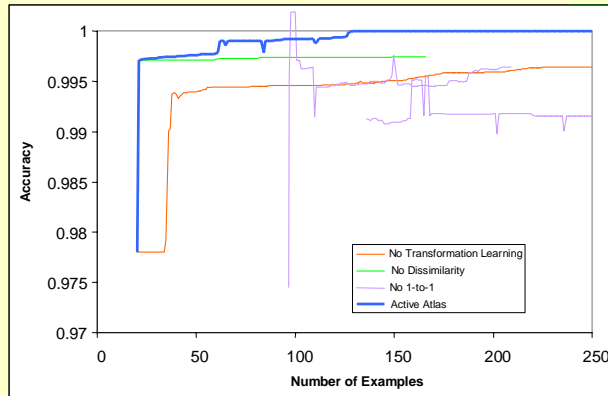
Weather Stations			Airports		
Code	Location		Code	Location	
PADQ	KODIAK,	AK	ADQ	Kodiak,	AK USA
KIGC	CHARLESTON AFB	VA	CHS	Charleston	VA USA
KCHS	CHARLETON	VA			

418 mapped objects / 17120 mappings proposed

Airport/Weather Results



Active Atlas Results



Applying Learned Weights & Rules

Application Domain	Total Number of Examples	Total Number of Test Examples	Average Accuracy
Restaurant	3310	662	.9989
Company	14303	2861	.9995
Airport	17120	3624	.9960

Related Work

- Key characteristics
 - Manual methods to customize rules for each domain
 - User-applied fixed threshold to match objects
 - No transformation weight learning
- Related work areas
 - Database Community (Ganesh et al, Monge&Elkan)
 - Information Retrieval (Cohen)
 - Sensor Fusion (Huang & Russell)
 - Record Linkage (Jaro, Winkler)

Information Retrieval Approach [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 TFIDF metric = Term Frequency Inverse Document Frequency

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”.

Sensor Fusion as Object identification

- Huang & Russell matched images of cars taken from multiple video cameras on a highway
- Employed Bayesian techniques, highlighting relationship between object identity and sensor fusion.
- Appearance models:
 - $P(\text{DownStreamObs} \mid \text{UpStreamObs}, \text{Same})$
- Must be sufficient data (e.g. previous observations) to learn appearance models

Removing Duplicates from Databases

- Referred to as the “Merge/Purge” problem
- E.g., Hernandez & Stolfo
 - Select “keys”
 - For each key:
 - Sort records
 - Compare records within a fixed size windows
 - Use “equality rules” to identify duplicates
- E.g., Monge & Elkan
 - Union-Find algorithm merges clusters
 - Use “Edit Distance” to evaluate similar field values

Data Mining

- Work conducted by Pinheiro&Sun
 - User-defined transformations
 - Learned attribute model (supervised learning)
- Work by Ganesh et al
 - Learned mapping rules (decision tree learner)

Conclusions

- Novel approach combines both mapping rule learning and transformation weight learning to create a robust object identification system
- Learns to classify examples with 100% accuracy
- Requires less user involvement than other baseline techniques (Passive Atlas & Information Retrieval)