On Statistical Schema Matching and Value Mapping

Anuj Jaiswal

Ph.D. Candidate,
College of Information Sciences and Technology,
The Pennsylvania State University

May 14, 2010
<table>
<thead>
<tr>
<th>Outline I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
</tr>
<tr>
<td>Preliminaries</td>
</tr>
<tr>
<td>The Framework</td>
</tr>
<tr>
<td>Experiments and Results</td>
</tr>
<tr>
<td>Research Contributions</td>
</tr>
</tbody>
</table>
Outline II
Future Work
Introduction

1. Driving forces for data integration and fusion
2. Data integration problem
3. Challenges
4. Our Approach
Driving forces for data integration and fusion

- Organizations evolving as *global* entities with distributed data, e.g., GM, Ford, Walmart
- Systems are characterized by a mix of *legacy* and *new* databases and applications
- Organizational *changes*
  1. *Growth* - size and diversity
  2. *Business re-engineering*
  3. Corporate mergers and acquisitions
Driving forces for data integration and fusion

- Organisations evolving as collections of distinct, autonomous departments with disconnected systems e.g. in financial services, NASA.
- Inter-operation between information sources over the web e.g. priceline.com and hotwire.com.
Data integration problem

Data integration across heterogeneous sources involves two related subtasks:

1. *Schema Matching*: “Structural alignment” of two schemas in two/more sources
2. *Value Mapping*: “Semantic resolution” of data across two/more matched attributes

Most research treats both problems independently
Schema Matching

- Reconciling structural alignment of data by matching schema attributes across information sources

Database A

<table>
<thead>
<tr>
<th>Location</th>
<th>Price ($)</th>
<th>Agent-id</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atlanta, GA</td>
<td>360,000</td>
<td>12</td>
</tr>
<tr>
<td>York, PA</td>
<td>200,000</td>
<td>15</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>City</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>Agent Smith</td>
<td>Jackson</td>
<td>GA</td>
</tr>
<tr>
<td>15</td>
<td>Agent John</td>
<td>Mile Run</td>
<td>PA</td>
</tr>
</tbody>
</table>

Database B

<table>
<thead>
<tr>
<th>Area</th>
<th>List-Price ($)</th>
<th>Agent Address</th>
<th>Agent Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>San Jose, CA</td>
<td>1,000,000</td>
<td>Ontario CA</td>
<td>Jane</td>
</tr>
<tr>
<td>Atlanta, GA</td>
<td>360,000</td>
<td>Jackson, GA</td>
<td>John Smith</td>
</tr>
</tbody>
</table>
Value Mapping

- Resolving semantic heterogeneity of data by mapping data value instances across matched schema attributes

<table>
<thead>
<tr>
<th>Database A</th>
<th>Database B</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Location</strong></td>
<td><strong>Area</strong></td>
</tr>
<tr>
<td>Atlanta, GA</td>
<td>San Jose, CA</td>
</tr>
<tr>
<td>York, PA</td>
<td></td>
</tr>
<tr>
<td><strong>Price ($)</strong></td>
<td><strong>List-Price ($)</strong></td>
</tr>
<tr>
<td>360,000</td>
<td>1,000,000</td>
</tr>
<tr>
<td>200,000</td>
<td></td>
</tr>
<tr>
<td><strong>Agent-id</strong></td>
<td><strong>Agent Address</strong></td>
</tr>
<tr>
<td>12</td>
<td>Ontario CA</td>
</tr>
<tr>
<td>15</td>
<td></td>
</tr>
<tr>
<td><strong>ID</strong></td>
<td><strong>Agent Name</strong></td>
</tr>
<tr>
<td>12</td>
<td>Jane</td>
</tr>
<tr>
<td>15</td>
<td></td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td><strong>City</strong></td>
</tr>
<tr>
<td>Agent Smith</td>
<td>Jackson</td>
</tr>
<tr>
<td>Agent John</td>
<td>Mile Run</td>
</tr>
<tr>
<td><strong>State</strong></td>
<td><strong>GA</strong></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Challenges

- Structural Conflicts
  - Naming Conflicts
- Attribute/Domain Conflicts
  - Data type conflicts
  - Measure and scale conflicts
  - Presence/absence of data values
  - Value semantics
Structural Conflicts

- Naming Conflicts
  - Multiple syntax/description for data values
  - Synonyms: Customer vs. Client
  - Common Words: Name vs. Student Name
  - No semantic similarity (Opaque conditions)
    - Four Wheel Drive Sedan vs. 4WD-S vs. Car Type 2
  - Similar names refer to semantically different attributes
    - Name for Book domain vs. Name for University domain
Attribute/Domain Conflicts

- Data type (representation) conflicts
  - StudentID: 92115151 (Number vs. String)
  - StudentID vs. Student Name
- Measurement, scale, precision etc. conflicts
  - Measurement - Light years vs. Miles
  - Scale - Miles vs. Kilometers
  - Precision - Float vs. Double
  - Date formats - 12/02/1980 vs. 02/12/1980
Challenges

Attribute/Domain Conflicts

- Presence/absence of data
  - Null entries present/absent
  - Matching attributes/values missing

- Value semantics
  - Same items different values
  - Different items same values
Our Approach

- Automated technique which utilizes (embeds) value mappings to enhance schema matching [5]
  1. Designed for opaque conditions
  2. A global objective function to capture dissimilarity for fixed schema match and value mapping
     - attributes
     - attribute pairs
  3. Integrates both problems within a common framework (minimization problem)
  4. Can tackle multiple feature spaces (categorical, continuous, mixed, transformed etc.)
Preliminaries

- Information Theory/Probability basics
- Common related work
- Kang Naughton Mutual Information Methods
- Drawbacks
Information Theory/ Probability

/GettyImages-945505387.png

- **Entropy**
  \[
  H(X) = - \sum_{x \in \mathcal{X}} p(x) \log p(x)
  \]  

- **Joint Entropy**
  \[
  H(X, Y) = - \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log p(x, y)
  \]  

- **Conditional Entropy**
  \[
  H(X|Y) = - \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log p(x|y)
  \]
Information Theory/ Probability

- **Mutual Information**
  \[
  MI(X; Y) = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log \frac{p(x, y)}{p(x)p(y)}
  \]  
  (4)

- **KL Divergence/ Relative Entropy**
  \[
  D(p\|q) = \sum_{x \in \mathcal{X}} p(x) \log \frac{p(x)}{q(x)}
  \]  
  (5)

- **Cross Entropy**
  \[
  h(p\|q) = E_p[-\log q] = H(p) + D(p\|q)
  \]  
  (6)
Information Theory/ Probability

- A likelihood function [3, 4] $L(\theta)$
  - probability or probability density for the occurrence of observations $(x_1, \ldots, x_n)$
  - given that the probability density $f(x; \theta)$ with parameter $\theta$ is known

- The likelihood function is defined as

\[
L(\theta) = f(x_1; \theta) \ldots f(x_n; \theta) \\
= \prod_{i=1}^{n} f(x_i; \theta) \quad (7)
\]
Related Work

- **Linguistic/ Lexical Approaches**
  - Uses lexical similarity between column names
  - E.g. Phone vs. HomePhone/ Office Phone

- **Instance Level Approaches**
  - Data instances of schemas are used to determine matches
  - E.g. Name vs. Student Name

- **Hybrid Matchers**
  - Combines multiple matching approaches to determine a set of ranked candidates

- **Composite Matchers**
  - Combines results to several matching techniques to determine match
Kang Naughton Technique

- Entropy based
  - Utilize statistics across single attributes
  - Align schemas by optimally reducing entropy across attributes

- Mutual Information (KN-MI) based
  - Utilize statistics based on attribute pairs
  - Build a dependency graph between attributes
  - Find schema match by utilizing a graph matching algorithm
Kang Naughton Technique

Database A

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>a1</td>
<td>b2</td>
<td>c1</td>
<td>d1</td>
</tr>
<tr>
<td>a2</td>
<td>b4</td>
<td>c2</td>
<td>d1</td>
</tr>
<tr>
<td>a3</td>
<td>b1</td>
<td>c1</td>
<td>d2</td>
</tr>
<tr>
<td>a4</td>
<td>b2</td>
<td>c1</td>
<td>d2</td>
</tr>
</tbody>
</table>

Database B

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>x1</td>
<td>y1</td>
<td>z2</td>
<td>w1</td>
</tr>
<tr>
<td>x1</td>
<td>y2</td>
<td>z4</td>
<td>w2</td>
</tr>
<tr>
<td>x2</td>
<td>y3</td>
<td>z1</td>
<td>w1</td>
</tr>
<tr>
<td>x2</td>
<td>y4</td>
<td>z2</td>
<td>w1</td>
</tr>
</tbody>
</table>

The matrices and graphs illustrate the relationships and interactions between the elements in each database.
Drawbacks

- Scenario 1:
  - Entropy statistics do not align between schemas
  - Value cardinality cannot be used for decision making

<table>
<thead>
<tr>
<th>Model (X)</th>
<th>P(X)</th>
<th>Color (Y)</th>
<th>P(Y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>XE</td>
<td>0.0472</td>
<td>White</td>
<td>0.0813</td>
</tr>
<tr>
<td>XL</td>
<td>0.0718</td>
<td>Blue</td>
<td>0.1077</td>
</tr>
<tr>
<td>XLE</td>
<td>0.1751</td>
<td>Red</td>
<td>0.1639</td>
</tr>
<tr>
<td>YE</td>
<td>0.2251</td>
<td>Silver</td>
<td>0.1702</td>
</tr>
<tr>
<td>YZ</td>
<td>0.2347</td>
<td>Black</td>
<td>0.1801</td>
</tr>
<tr>
<td>ZX</td>
<td>0.2461</td>
<td>Green</td>
<td>0.2968</td>
</tr>
<tr>
<td>H(X)</td>
<td>2.3938</td>
<td></td>
<td>2.4686</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A</th>
<th>P(A)</th>
<th>B</th>
<th>P(B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>0.0965</td>
<td>White</td>
<td>0.0733</td>
</tr>
<tr>
<td>XL</td>
<td>0.1010</td>
<td>Blue</td>
<td>0.0801</td>
</tr>
<tr>
<td>XLE</td>
<td>0.1075</td>
<td>Red</td>
<td>0.1011</td>
</tr>
<tr>
<td>GT</td>
<td>0.2044</td>
<td>Silver</td>
<td>0.2201</td>
</tr>
<tr>
<td>GE</td>
<td>0.2349</td>
<td>Black</td>
<td>0.2418</td>
</tr>
<tr>
<td>GL</td>
<td>0.2557</td>
<td>Green</td>
<td>0.2836</td>
</tr>
<tr>
<td>H(A)</td>
<td>2.4677</td>
<td>H(B)</td>
<td>2.3939</td>
</tr>
</tbody>
</table>

Table 1

Table 2
Drawbacks

- Scenario 2:
  - Entropy statistics are similar within schema
  - Value cardinality cannot be used for decision making

<table>
<thead>
<tr>
<th>Model</th>
<th>P(X)</th>
<th>Color</th>
<th>P(Y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>XE</td>
<td>0.0472</td>
<td>White</td>
<td>0.0813</td>
</tr>
<tr>
<td>XL</td>
<td>0.0718</td>
<td>Blue</td>
<td>0.1077</td>
</tr>
<tr>
<td>XLE</td>
<td>0.1751</td>
<td>Red</td>
<td>0.1639</td>
</tr>
<tr>
<td>YE</td>
<td>0.2251</td>
<td>Silver</td>
<td>0.1702</td>
</tr>
<tr>
<td>YZ</td>
<td>0.2347</td>
<td>Black</td>
<td>0.1801</td>
</tr>
<tr>
<td>ZX</td>
<td>0.2461</td>
<td>Green</td>
<td>0.2968</td>
</tr>
<tr>
<td>H(X)</td>
<td>2.3938</td>
<td></td>
<td>2.4686</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A</th>
<th>P(A)</th>
<th>B</th>
<th>P(B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>0.0244</td>
<td>White</td>
<td>0.1603</td>
</tr>
<tr>
<td>XL</td>
<td>0.0570</td>
<td>Blue</td>
<td>0.1653</td>
</tr>
<tr>
<td>XLE</td>
<td>0.1796</td>
<td>Red</td>
<td>0.1846</td>
</tr>
<tr>
<td>GT</td>
<td>0.2047</td>
<td>Silver</td>
<td>0.2343</td>
</tr>
<tr>
<td>GE</td>
<td>0.2536</td>
<td>Black</td>
<td>0.2554</td>
</tr>
<tr>
<td>GL</td>
<td>0.2807</td>
<td></td>
<td></td>
</tr>
<tr>
<td>H(A)</td>
<td>2.2962</td>
<td>H(B)</td>
<td>2.2962</td>
</tr>
</tbody>
</table>

Table 1

Table 2
Drawbacks

- Scenarios may apply to
  - Entropy
  - Mutual Information
  - Conditional Entropy
The Framework

- Capitalize on the value mapping dimension
  1. Utilize frequency of occurrence of an attribute’s value set
  2. Categorical Attributes
     - Probability mass function (pmf) of an attribute’s value set is more distinctive than entropy
  3. Continuous Attributes
     - Probability density function (pdf) of an attribute’s value set is more distinctive than the pmf/entropy
  4. Mixed Attributes
     - Utilize pmf’s and pdf’s to model such schemas
  5. Matching pmf’s and pdf’s across schemas should be more reliable
     - Second-order statistics for finer matching
Modelling Dissimilarity of Match

- Model dissimilarity score
  - A global objective function which provides a confidence score for a fixed schema mapping and value mapping
  - Uses a dissimilarity metric based on
    - Euclidean distance
    - Log-Likelihood
    - Relative Entropy

- Two dimensional minimization of global objective
  1. First dimension - Schema Matching
  2. Second dimension - Embedded Value Mapping
Modelling Dissimilarity of Match

- First-order dissimilarity metric models
  - Measures how well the probability mass functions (and/or) probability distribution functions align for fixed schema matching
  - Uses first-order statistics of attributes

- Second-order dissimilarity metric models
  - Measures how well the joint probability mass functions (and/or) joint probability distribution functions align for fixed schema matching
  - Uses second-order statistics between attribute pairs
Categorical Feature Spaces

<table>
<thead>
<tr>
<th>Model (X)</th>
<th>P(X)</th>
<th>Color (Y)</th>
<th>P(Y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>XE</td>
<td>0.0472</td>
<td>White</td>
<td>0.0813</td>
</tr>
<tr>
<td>XL</td>
<td>0.0718</td>
<td>Blue</td>
<td>0.1077</td>
</tr>
<tr>
<td>XLE</td>
<td>0.1751</td>
<td>Red</td>
<td>0.1639</td>
</tr>
<tr>
<td>YE</td>
<td>0.2251</td>
<td>Silver</td>
<td>0.1702</td>
</tr>
<tr>
<td>YZ</td>
<td>0.2347</td>
<td>Black</td>
<td>0.1801</td>
</tr>
<tr>
<td>ZX</td>
<td>0.2461</td>
<td>Green</td>
<td>0.2968</td>
</tr>
</tbody>
</table>

**Table 1**

H(X) = 2.3938, H(Y) = 2.4686

PMF of Model

PMF of Color

Model P(X) Color P(Y)

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>XE</td>
<td>0.0472</td>
<td>White</td>
</tr>
<tr>
<td>XL</td>
<td>0.0718</td>
<td>Blue</td>
</tr>
<tr>
<td>XLE</td>
<td>0.1751</td>
<td>Red</td>
</tr>
<tr>
<td>YE</td>
<td>0.2251</td>
<td>Silver</td>
</tr>
<tr>
<td>YZ</td>
<td>0.2347</td>
<td>Black</td>
</tr>
<tr>
<td>ZX</td>
<td>0.2461</td>
<td>Green</td>
</tr>
</tbody>
</table>
Modelling Dissimilarity of Match

First-order Dissimilarity Models

▶ Euclidean squared distance based model

\[ D_{EU}^{AB} = \sum_{i=1}^{N_{attr_1}} \sum_{i'=1}^{N_{attr_2}} \delta(i - m_a(i')) \sum_{j=1}^{N_{Values(i)}} \left[ p_i(j) - \left\{ \sum_{j'=1}^{N_{Values(i')}} \delta(j - M^{(i,i')}_v(j')) p_{i'}(j') \right\} \right]^2 \]

(8)

▶ Relative Entropy based model

\[ D_{CE}^{AB} = \sum_{i=1}^{N_{attr_1}} \sum_{i'=1}^{N_{attr_2}} \delta(i - m_a(i')) \sum_{j=1}^{N_{Values(i)}} \left[ p_i(j) \times \log \frac{p_i(j)}{\sum_{j'=1}^{N_{Values(i')}} \delta(j - M^{(i,i')}_v(j')) p_{i'}(j')} \right] \]

(9)
First-order Models: Issues

- Advantages of Euclidean-distance vs. Relative Entropy
  - Relative Entropy requires $N_{\text{Values}}(i) = N_{\text{Values}}(i')$
  - Value alphabet set must be same for matching attributes
  - Requires Attribute Alphabet reconstruction
    - Introduce extra symbols and assign them small probability followed by normalization
    - Reduce size of the larger alphabet by deleting the smallest probability values followed by normalization
    - Euclidean-distance is free from such requirements
  - Relative Entropy is not symmetric
  - Euclidean-distance is computationally cheaper
Second-order Dissimilarity Models

- Euclidean-squared distance based model

\[
D_{PEU}^{AB} = \sum_{i=1}^{N_{\text{attr}_1}} \sum_{j=1}^{N_{\text{attr}_1}} \left[ \sum_{i' = 1}^{N_{\text{attr}_2}} \delta(i - m_a(i')) \sum_{j' = 1}^{N_{\text{attr}_2}} \delta(j - m_a(j')) \left[ \sum_{k'=1}^{N_V(i')} \sum_{l'=1}^{N_V(j')} \left( p_{ij}(k, l) - \left\{ \sum_{k'=1}^{N_V(i')} \sum_{l'=1}^{N_V(j')} \delta(k' - M_{i,i'}(k)) \delta(l' - M_{j,j'}(l)) p_{i,j'}(k', l') \right\} \right] \right] \right]^{2}
\]

(10)
Matching and Mapping Strategy

- Our global objective is now two-dimensional minimization of the dissimilarity objective

\[
D_{Overall} = \min_{x \in S} \left\{ \min_{y \in V} [D_{AB}^M] \right\}
\]  
(11)
Continuous Feature Spaces

<table>
<thead>
<tr>
<th>Price ($)</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>(X)</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>11</td>
</tr>
<tr>
<td>. . .</td>
<td>. . .</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
</tr>
</tbody>
</table>

\[ \alpha_1 = 0.7 \quad \mu_1 = 3.75 \quad \sigma_1 = 0.75 \]
\[ \alpha_2 = 0.3 \quad \mu_2 = 6.00 \quad \sigma_2 = 0.50 \]

\[ f(x_i; \theta_i) = \sum_{n=1}^{K_i} \alpha_n^i \phi(x_i | \mu_n^i, \sigma_n^i) \]
\[ \sum \alpha_n^i = 1 \]
\[ 0 \leq \alpha_n^i \leq 1 \]
Continuous Feature Spaces

<table>
<thead>
<tr>
<th>Price ($)</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>(X)</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>11</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
</tr>
</tbody>
</table>

\[ f(x_i; \theta_i) = \sum_{n=1}^{K_i} \alpha_n^i \phi(x_i|\mu_n^i, \sigma_n^i) \]

\[ \sum \alpha_n^i = 1 \]

\[ 0 \leq \alpha_n^i \leq 1 \]

\[ \alpha_1 = 0.7 \quad \mu_1 = 3.75 \quad \sigma_1 = 0.75 \]

\[ \alpha_2 = 0.3 \quad \mu_2 = 6.00 \quad \sigma_2 = 0.50 \]
First-order Dissimilarity Models

- Log-Likelihood based model

\[
D_{AB}^{LL} = \sum_{i=1}^{N_C_1} \sum_{j=1}^{N_C_2} \delta(i - m_a(j)) \left[ \sum_{a=1}^{N_R_1} \log \left( \sum_{n=1}^{K_j} \beta^j_n \mathcal{N}_n(x_{ia} \mid \mu^j_n, \sigma^j_n) \right) + \sum_{a=1}^{N_R_2} \log \left( \sum_{n=1}^{K_i} \alpha^i_n \mathcal{N}_n(y_{ja} \mid \mu^i_n, \sigma^i_n) \right) \right] \tag{12}
\]

- Euclidean-distance based models
  - Ordered Component Model

\[
D_{AB}^{EU} = \sum_{i=1}^{N_C_1} \sum_{j=1}^{N_C_2} \delta(i - m_a(j)) \left[ \sum_{k=1}^{K_i} \sum_{k'=1}^{K_j} (\alpha^i_{xk} - \delta(k - m_c^{(i,j)}(k')) \beta^j_{yk'})^2 \right] \tag{13}
\]

- Model Estimation Model

\[
D_{AB}^{ME} = \sum_{i=1}^{N_C_1} \sum_{j=1}^{N_C_2} \delta(i - m_a(j)) \left[ \sum_{k=1}^{K_i} (\alpha^i_{xk} - \hat{\alpha}^i_{yk})^2 + \sum_{k'=1}^{K_j} (\beta^j_{yk'} - \hat{\beta}^j_{xk'})^2 \right] \tag{14}
\]
Second-order Dissimilarity Models

- **Log-Likelihood based model**

\[
D_{AB}^{LL} = \sum_{i=1}^{NC_1} \sum_{j=1}^{NC_1} \sum_{i'=1}^{NC_2} \sum_{j'=1}^{NC_2} \delta(i - m_a(i')) \delta(j - m_a(j')) \left[ \sum_{a=1}^{NR_1} \log \left( \sum_{n=1}^{K_{yij'}} \beta_{yn}' \mathcal{N}(x_{ija}|\mu_{yn}', \sigma_{yn}') \right) \right] \\
+ \sum_{a=1}^{NR_2} \log \left[ \sum_{n=1}^{K_{xij}} \alpha_{xn} \mathcal{N}(y_{i'j'a}|\mu_{xn}', \sigma_{xn}') \right] \\
(15)
\]

- **Euclidean-distance based models**
  - **Ordered Component Model**

\[
D_{AB}^{EU} = \sum_{i=1}^{NC_1} \sum_{j=1}^{NC_1} \sum_{i'=1}^{NC_2} \sum_{j'=1}^{NC_2} \delta(i - m_a(i')) \delta(j - m_a(j')) \left[ \sum_{k=1}^{K_{xij}} \sum_{k'=1}^{K_{yij'}} (\alpha_{xk}' - \delta(k - m_c^{(ij',i'j')}(k')) \beta_{yk'})^2 \right] \\
(16)
\]
### Continuous Feature Spaces

#### Second-order Dissimilarity Models

- Euclidean-distance based models
  - Model Estimation

\[
D_{\text{AB}}^{\text{ME}} = \sum_{i=1}^{N_C_1} \sum_{j=1}^{N_C_1} \sum_{i' = 1}^{N_C_2} \sum_{j' = 1}^{N_C_2} \delta(i - m_a(i'))\delta(j - m_a(j')) \left[ \sum_{k=1}^{K_{xij}} (\alpha_{xk}^{ij} - \hat{\alpha}_{y_k})^2 + \sum_{k' = 1}^{K_{yij'}} (\beta_{y_{k'}}^{ij'} - \hat{\beta}_{x_{k'}})^2 \right]
\]

(17)
Matching and Mapping Strategy

Our global objective is now one-dimensional minimization/maximization of the dissimilarity objective

\[
D_{\text{Overall}} = \min_{x \in S} [D_{AB}^{MM}]
\]

Or,

\[
D_{\text{Overall}} = \max_{x \in S} [D_{AB}^{LL}]
\]
Matching and Mapping Strategy

- Our global objective is now one-dimensional minimization/maximization of the dissimilarity objective

\[ D_{Overall} = \min_{x \in S} [D_{AB}^{MM}] \]  \hspace{1cm} (18)

Or,

\[ D_{Overall} = \max_{x \in S} [D_{AB}^{LL}] \]  \hspace{1cm} (19)
Matching and Mapping Strategy

- Our global objective is now one-dimensional minimization/maximization of the dissimilarity objective

\[
D_{\text{Overall}} = \min_{x \in S} \left[ D_{AB}^{MM} \right] \\
\text{Or,} \\
D_{\text{Overall}} = \max_{x \in S} \left[ D_{AB}^{LL} \right]
\]
Mixed Feature Spaces

- We can separate this problem into two sub-problems
  - Categorical Attributes and use discrete dissimilarity measures
  - Continuous Attributes and use continuous dissimilarity measures
  - Merge results to arrive at final schema match

OR,

- Define a global objective which can tackle such types of tables simultaneously
  - Attributes are defined by either probability mass functions or probability distribution functions
  - Global objective is sum of compatible dissimilarity measures for the different feature spaces
Mixed Feature Spaces

First-order Dissimilarity Models

- Euclidean squared distance based models

\[
D_{AB}^{EU} = \sum_{i=1}^{N_D_1} \sum_{i'=1}^{N_D_2} \delta(i - m_a(i')) \sum_{j=1}^{N_{Values}(i)} p_j(i) - \left\{ \sum_{j'=1}^{N_{Values}(i)} \delta(j - M_v^{(i,i')}(j')) p_{i'}(j') \right\}^2 \\
+ \sum_{j=1}^{N_{C_1}} \sum_{j'=1}^{N_{C_2}} \delta(j - m_a(j')) \left[ \sum_{k=1}^{K_{xj}} \sum_{k'=1}^{K_{yj'}} (\alpha_{xk}^{j} - \delta(k - m_{c}^{(j,j')}(k'))) \beta_{yk'}^{j'} \right]^2
\]

(20)

\[
D_{AB}^{EU} = \sum_{i=1}^{N_D_1} \sum_{i'=1}^{N_D_2} \delta(i - m_a(i')) \sum_{j=1}^{N_{Values}(i)} p_j(i) - \left\{ \sum_{j'=1}^{N_{Values}(i)} \delta(j - M_v^{(i,i')}(j')) p_{i'}(j') \right\}^2 \\
+ \sum_{j=1}^{N_{C_1}} \sum_{j'=1}^{N_{C_2}} \delta(j - m_a(j')) \left[ \sum_{k=1}^{K_{xj}} (\alpha_{xk}^{j} - \hat{\alpha}_{yk}^{j})^2 + \sum_{k'=1}^{K_{yj'}} (\beta_{yk'}^{j'} - \hat{\beta}_{xk'}^{j'})^2 \right]
\]

(21)
First-order Dissimilarity Models

- Log-Likelihood based model

\[
D_{AB}^{LL} = \sum_{i=1}^{N_{D1}} \sum_{i' = 1}^{N_{D2}} \delta(i - m_a(i')) \left[ \sum_{j=1}^{N_V(i)} \sum_{j' = 1}^{N_V(i')} \delta(j - M^{(i,i')}_V(j')) n_i(j) p_{i'}(j') \right. \\
+ \left. \sum_{i=1}^{N_{C1}} \sum_{j=1}^{N_{C2}} \delta(i - m_a(j)) \left[ \sum_{a=1}^{N_{R1}} K_{yj} \log \left( \sum_{n=1}^{N_y} \beta_{yn}^i N(x_{ia} | \mu_{yn}^j, \sigma_{yn}^j) \right) \right. \\
+ \left. \sum_{a=1}^{N_{R2}} K_{xi} \log \left( \sum_{n=1}^{N_x} \alpha_{xn}^i N(y_{ja} | \mu_{xn}^i, \sigma_{xn}^i) \right) \right] \right]
\]

(22)
First-order Models: Issues

- Advantages of Euclidean-distance vs. Log-Likelihood
  - Log-Likelihood requires cardinality of matching pmf’s to be same
  - Requires Attribute Alphabet reconstruction
    - Introduce extra symbols and assign them small probability followed by normalization
    - Use the pmf of larger cardinality in measuring the dissimilarity
    - Euclidean-distance is free from such requirements
  - Euclidean-distance is computationally cheaper
Matching and Mapping Strategy

- Our global objective is now one-dimensional minimization/maximization of the dissimilarity objective

\[
D_{Overall} = \min_{x \in S} [D_{AB}^{MM}] \tag{23}
\]

Or,

\[
D_{Overall} = \max_{x \in S} [D_{AB}^{LL}] \tag{24}
\]
Matching and Mapping Strategy

- Our global objective is now one-dimensional minimization/maximization of the dissimilarity objective

\[
D_{Overall} = \min_{x \in S} [D_{MM}^{AB}] 
\]

Or,

\[
D_{Overall} = \max_{x \in S} [D_{LL}^{AB}] 
\]
Matching and Mapping Strategy

- Our global objective is now one-dimensional minimization/maximization of the dissimilarity objective.

\[
D_{Overall} = \min_{x \in S} [D^{MM}_{AB}] \quad \text{(23)}
\]

Or,

\[
D_{Overall} = \max_{x \in S} [D^{LL}_{AB}] \quad \text{(24)}
\]
Experiments and Results

- **Data Analyzer and Modelling**
  - Loads data tables
  - Performs density estimation for continuous/ mixed datasets
  - Learns *latent* categorical variables for continuous attributes
    - Use maximum a posteriori (MAP) estimate to generate pmf’s
- **Schema Matching with *Embedded* Value Mapping algorithms take over**
  - Measure precision of schema match over 50 iterations
  - Each iteration corresponds to a *different* schema matching problem
Implementation

- **Software**
  - GCC 3.4.5, GSL Scientific Library
  - R [7], Mclust [1]

- **Hardware**
  - Cyberstar cluster
  - Inti cluster
Datasets

- We used real world datasets from the Census Bureau
  - 1990 Public-Use Microdata Samples (PUMS) 5% Sample for states of California, New York, Texas
  - Texas Dataset
    - 231 Attributes
    - 935K records
  - California Dataset
    - 231 Attributes
    - 1500K records
  - New York Dataset
    - 231 Attributes
    - 960K records
Dataset Creation

- We created three datasets for testing out algorithms
  - Categorical Dataset
    - Selected 30 categorical columns (Value Alphabet $\leq 50$)
    - Some columns were discretized using coding from Meek et. al. [6]
    - Example attributes: “Race”, “Age” etc.
  - Continuous Dataset
    - Selected 26 pure continuous columns (Value Alphabet $\geq 90$)
    - Example attributes: “Water Cost”, “Age” etc.
  - Mixed Dataset
    - Selected 15 pure continuous columns (Value Alphabet $\geq 90$)
    - Selected 15 categorical columns (Value Alphabet $\leq 50$)
Categorical Dataset

- **Second-order Model**
  - Each attribute pair is modelled using a joint probability mass function

- **First-order Model**
  - Each attribute is modelled using a probability mass function

- **Euclidean-squared dissimilarity metric model used**
  - Free from Value Alphabet Reconstruction issues
Monotonic Nature

- Evolution of dissimilarity score as a function of *accepted* iteration
  - Schema matching on 10 attributes across NY and CA datasets
  - Accepted iteration obtained by 2-opt switch
  - and decreases the objective function

![Euclidean-squared Dissimilarity Score vs. Accepted Iteration](image1)

![Precision vs. Accepted Iteration](image2)
One-to-One Schema Matching

Average precision for CA vs. NY dataset

Effect of rows on precision for CA vs. NY dataset

Average precision for CA vs. TX dataset

Effect of rows on precision for CA vs. TX dataset
Continuous Dataset

- Mixture model component size for each attribute is estimated using Bayesian Information Criterion (BIC)

- BIC is defined as

\[
BIC = -2 \log L(x; \theta) + k \log n
\]  

(25)

- Component size is varied as

\[1 \leq k \leq 20\]
Continuous Attribute Modelling

BIC values for varying component size for New York dataset

BIC values for varying component size for California Dataset

Estimated Density for “Age” using 4 components for New York dataset

Estimated Density for “Age” using 4 components for California Dataset
Monotonic Nature

- Evolution of dissimilarity score as a function of accepted iteration
  - Schema matching on 10 attributes across NY and CA datasets
  - Accepted iteration obtained by 2-opt switch
  - and increases the objective function

![Log-Likelihood Dissimilarity Score vs. Accepted Iteration](image1)

![Precision vs. Accepted Iteration](image2)
One-to-One Schema Matching

Average precision for CA vs. NY dataset

Effect of rows on precision for CA vs. NY dataset

Average precision for CA vs. TX dataset

Effect of rows on precision for CA vs. TX dataset
Continuous Dataset

Computation Time

Computation Time for optimization for CA vs. NY dataset

Computation Time for optimization for CA vs. TX dataset

Computation Time for modelling for CA vs. NY dataset

Computation Time for modelling for CA vs. TX dataset
Mixed Dataset

- Each continuous attribute is modelled by a mixture of normals
  - Component size is estimated using Bayesian Information Criterion (BIC)
  - Component size is varied as $1 \leq k \leq 20$

- Each categorical attribute is modelled by a probability mass function
One-to-One Schema Matching

Average precision for CA vs. NY dataset

Effect of rows on precision for CA vs. NY dataset

Average precision for CA vs. TX dataset

Effect of rows on precision for CA vs. TX dataset
Mixed Dataset

Computation Time

Computation Time for optimization for CA vs. NY dataset

Computation Time for optimization for CA vs. TX dataset

Computation Time for modelling for CA vs. NY dataset

Computation Time for modelling for CA vs. TX dataset
Experimental Summary

- Categorical Feature Spaces
  - First-order [5] and Second-order dissimilarity metrics evaluated
  - Random Initialization
  - Ground-truth Initialization
  - Fixed Initialization

- Mixed and Continuous Feature Spaces
  - First-order dissimilarity metrics evaluated (*To be submitted to pVLDB*)
  - Random Initialization
  - Ground-truth Initialization
  - Fixed Initialization
Research Contributions

- No previous work which tackles both schema matching and value mapping at the same time.
- Extended *embedded* value mapping technique to tackle
  - Continuous attributes
  - Mixed attributes
Future Work

- Evaluating second-order models
- Transformed Feature Spaces
- Schema Matching Criterion
Second-order Models

- Continuous and Mixed Feature Spaces
  - Experimental evaluation of second-order models
  - Finer matching should be achieved

Bayesian Information Criterion scores to determine component size for attribute pair “Age” and “Weight” for the California dataset. Best BIC score is obtained for 4 components.

Density estimate perspective for attribute pair “Age” and “Weight” for California Dataset

Density estimate contour for attribute pair “Age” and “Weight” for California Dataset
Transformed Feature Spaces

- Continuous Dissimilarity Metric Models
  - Assume no transformation necessary between two matching attributes

- However, real world datasets might not have one-to-one correspondence between data values
  - “Temperature” might be measured in “Celcius” or “Fahrenheit”
  - “Distance” may be measured in “Kilometers” or “Miles”
  - Two similar attributes would match only when a \textit{transformation} operator is applied
Transformed Feature Spaces

- Continuous dissimilarity models can be extended to tackle such situation
  - We consider **affine** transformations
    \[ x \mapsto Ax + b \quad (26) \]
- Consider set of pre-defined transformations
  \[ T = [T_1, \ldots, T_{NT}]^T \quad (27) \]
First-order Dissimilarity Model

- **Log-Likelihood based model**

\[
D_{AB}^{LL} = \sum_{i=1}^{C_1} \sum_{j=1}^{C_2} \delta(i - m_a(j)) \left[ \sum_{a=1}^{R_1} \log \left( \sum_{n=1}^{K_i} \beta_{yn} N(T_z [x_{ia}] | \mu_{yn}^i, \sigma_{yn}^i) \right) + \sum_{a=1}^{R_2} \log \left( \sum_{n=1}^{K_j} \alpha_{xn} N(T_z^{-1} [y_{ja}] | \mu_{xn}^i, \sigma_{xn}^i) \right) \right]
\]

(28)

- **Euclidean-distance based models**
  - **Model Estimation Model**

\[
D_{AB}^{ME} = \sum_{i=1}^{C_1} \sum_{j=1}^{C_2} \delta(i - m_a(j)) \left[ \sum_{k=1}^{K_i} (\alpha_{xk}^i - \hat{\alpha}_{yk}^{T_z^{-1}})^2 + \sum_{k'=1}^{K_j} (\beta_{yk'}^j - \hat{\beta}_{xk'}^{T_z})^2 \right]
\]

(29)
Matching and Mapping Strategy

- Our global objective is now two-dimensional minimization/maximization of the dissimilarity objective

\[
D_{Overall} = \min_{x \in S} \left[ D_{ABTz}^{ME} \right] \quad \text{(30)}
\]

Or,

\[
D_{Overall} = \max_{x \in S} \left[ D_{ABTz}^{LL} \right] \quad \text{(31)}
\]
Experimental Validation

- Datasets
  - Create a transformed dataset from Census Continuous dataset
- Extend to Second-order model for dealing with transformed spaces
- Integrate transformation learning within framework
  - Regression analysis techniques for learning operator
    - Ordinary least squares (OLS)
  - Complexity of problem will grow as incorrect attributes might now match better
Schema Matching Criterion

- One-to-One schema matching
  - Each attribute in *schema 1* has unique matching attribute in *schema 2* and vice versa
  - Equal number of attributes in both schemas
  - Our methods deal with this type of schema matching

- Onto or Subset schema matching
  - Each attribute in *schema 1* has unique matching attribute in *schema 2*
  - Each attribute in *schema 2* has unique matching attribute in *schema 1* or remains unmatched
  - Unequal number of attributes across schemas
  - Can be posed as an assignment problem
Schema Matching Criterion

- Partial schema matching
  - Each attribute in *schema 1* has a unique matching attribute in *schema 2* or remains unmatched, and vice versa
  - Unknown number of attributes match
  - Investigate methods for proposing confident matches
    - Regularization: Adapting the BIC measure for model selection
    - Hypothesis testing
Schema Matching Criterion

- Many-to-many schema matching
  - Each attribute in schema 1 has a set of matching attributes in schema 2 or remains unmatched, and vice versa
  - Most difficult and general problem
  - Solving this problem automatically may not be feasible (user input maybe required)
  - Investigate methods for proposing top-K [2] candidate attributes that may be matched
Questions?
Backup Slides
Algorithm 1 Overview of our approach

Input: Schemas, $T1$ and $T2$
Output: Schema Match $s$, $tmpS$
        Value Mapping $v$, $tmpV$, $v1$

$sMatchScore ← ∞$ {Best Dissimilarity Score}
$sSpace ← getSchemaMatchSearchSpace()$

while $sSpace$ is not empty do
    $tmpS ← getNextSchemaMatch(sSpace)$
    $vSpace ← getValueMappingSpace()$
    $vMapScore ← ∞$ {Stores dissimilarity score for fixed schema match and value mapping}

    while $vSpace$ is not empty do
        $tmpV ← getNextValueMapping(vSpace)$
        $score ← computeDissimilarity(tmpS, tmpV, T1, T2)$
        if $score < vmapScore$ then
            $vmapScore ← score$ {Save current dissimilarity score}
            $v1 ← tmpV$ {Save current value mapping}
        end if
        removeValueMapping($vSpace$, $tmpV$)
    end while
    if $vmapScore < smatchScore$ then
        $s ← tmpS$ {Store current schema match}
        $v ← v1$ {Store the best value mapping for this schema match}
        $smatchScore ← vmapScore$
    end if
    removeSchemaMatch($sSpace$, $tmpS$)
end while

Categorical Schema Matching Algorithm
Continuous Schema Matching Algorithm

**Algorithm 2** Overview of our approach

**Input:** Schemas, $T_1$ and $T_2$

**Output:** Schema Match $s$, $tmpS$

$sMatchScore \leftarrow \infty \{ \text{Best Dissimilarity Score} \}$

$sSpace \leftarrow \text{getSchemaMatchSearchSpace()}$

**while** $sSpace$ is not empty **do**

$tmpS \leftarrow \text{getNextSchemaMatch}(sSpace)$

$sMapScore \leftarrow \infty \{ \text{Stores dissimilarity score for fixed schema match and value mapping for continuous features} \}$

$sMapScore \leftarrow \text{computeDissimilarity}(tmpS, T_1, T_2)$

**if** $sMapScore < sMatchScore$ **then**

$s \leftarrow tmpS \{ \text{Store current schema match} \}$

$s \leftarrow sMapScore$

**end if**

removeSchemaMatch($sSpace, tmpS$)

**end while**
Two-opt Switching

- Simple local search algorithm that we use in our heuristic search strategy
  - Involves swapping of the schema attributes
  - Accepted schema match is one which reduces global objective
Pseudocode for heuristic search strategy

**Algorithm 3** Pseudocode for the heuristic search strategy when using two-opt switching.

```plaintext
best_match ← get_Initial_Schema_Match()
D_{Overall} ← D_{EU}^{AB}(best_match)
repeat
    new_match ← Two-Opt-Switch(best_match)
    if D_{EU}^{AB}(best_match) > D_{EU}^{AB}(new_match) then
        best_match ← new_match
        D_{Overall} ← D_{EU}^{AB}(best_match)
    end if
until no further improvement or a specified number of iterations
```
Ground-truth Initialization - Categorical

Average precision for CA vs. NY dataset

Average precision for CA vs. TX dataset
Ground-truth Initialization - Continuous

Average precision for CA vs. NY dataset

Average precision for CA vs. TX dataset
Ground-truth Initialization - Mixed

Average precision for CA vs. NY dataset

Average precision for CA vs. TX dataset
References I

Fraley, C. and Raftery, A.E.
MCLUST: Software for model-based cluster analysis.
*Journal of Classification, 16(2):297–306, 1999.*

A. Gal.
Managing uncertainty in schema matching with top-k schema mappings.
*Journal on Data Semantics VI, pages 90–114, 2006.*

J. Harris and H. Stöcker.
*Handbook of mathematics and computational science.*
References II

P. Hoel.
*Introduction to mathematical statistics.*

A. Jaiswal, D. Miller, and P. Mitra.
Un-Interpreted Schema Matching with Embedded Value Mapping under Opaque Column Names and Data Values. *IEEE Transactions on Knowledge and Data Engineering*, 2010.

C. Meek, B. Thiesson, and D. Heckerman.
The learning-curve sampling method applied to model-based clustering.
References III

R Development Core Team.  
*R: A Language and Environment for Statistical Computing*.  
ISBN 3-900051-07-0.