Discovering Complex Matchings across Web Query Interfaces: A Correlation Mining Approach
Bin He, Kevin Chen-Chuan Chang, Jiawei Han

Presented by: Maryam Karimzadehgan
mkarimzadehgan@cs.uwaterloo.ca

Outline

- Motivation of integrating the deep web
- Mining Algorithm
- H-Measure
- Data Preparation Step
- Experiments
- Conclusion and Discussion

Motivation: Matching → Mining

- Group attributes
  Larger concept e.g. \{adults, seniors, children, infants\} denotes the number of passengers.
- Synonym relationship
  Different sources may use different attributes for the same concept, e.g. \{from\} = \{depart\}, \{to\} = \{destination\)

Group attribute + Synonym attribute → Complex Matching
m:n → m:n matching
e.g. \{adults, seniors, children, infants\} = \{passengers\}  → 1:4 matching
Correlation Mining

- Match more than two attribute groups
  
  \{adults, seniors, children, infants\} = \{passengers\} = \{number of tickets\}

- \textit{n-ary complex matching} $\rightarrow$ aggregation of several binary m:n matchings

- DCM (Dual Correlation Mining) framework for mining \textit{n-ary complex matchings}

Outline

- Introduction
- Motivation of integrating the deep web
- Mining Algorithm
- H-Measure
- Data Preparation Step
- Experiments
- Discussion

DCM Framework (Dual Correlation Mining)

Data Preparation:

schema transaction $\rightarrow$ mining by extracting the attribute entities from query interfaces

Correlation Mining:

finds complex matchings considering both positive and negative correlations
Mining Algorithm

- **Group discovery (individual attribute)**
  - Mining positively correlated attribute \( \Rightarrow \) potential attribute group
  - Potential group is not suitable for matching \( \Rightarrow \) synonym relationship (negative correlation)

- **Matching discovery (attribute groups)**
  - Given the potential group \( \Rightarrow \) Mining negatively correlated attribute \( \Rightarrow \) potential n-ary complex matching

- **Matching selection**
  - A potential matching may not be considered as correct due to existence of conflicts among matchings
  - Select the consistent matchings from the mining result

Complex Matching Discovery

- Negative correlation should exist between two groups. (e.g. (destination) \( \times \) (b) \( \Rightarrow \) (arrival city))
- \( C_{\text{min}} \) is the minimal value of the pairwise evaluation
- \( C_{\text{min}} \) satisfies the "Apriori" feature. e.g. \( C_{\text{min}}(A, m) = C_{\text{min}}(A', m) \)
  - \( \min \{1, 3, 5\} \leq \min \{3, 5\} \)
- \( C_{\text{min}} \) can mine both positive and negative correlations.

- A set of attributes \( \{A_1, \ldots, A_n\} \) is positively correlated if
  \[ C_{\text{min}} \geq \text{Threshold} \]

- A set of attributes groups \( \{G_1, \ldots, G_n\} \) is negatively correlated attribute groups if
  \[ C_{\text{min}} \leq \text{Threshold} \]

Algorithm N-ary (Discovering Complex Matching)

```
Algorithm N-ary (Discovering Complex Matching)
Input: Specifications \( G_1 \rightarrow G_2 \rightarrow \ldots \rightarrow G_n \),
Output:ragen = set containing each group
Begin
1. \( \text{ragen} := \emptyset \)
2. for each \( G_L \) do
3.     for each \( G_R \) do
4.         if \( \text{len}(G_L) = \text{len}(G_R) \) then \( \text{ragen} := \text{ragen} \cup \langle G_L, G_R \rangle \)
5.     end for
6. end for
7. end for
8. end while
9. for each \( G_L \) do
10.     for each \( G_R \) do
11.         if \( \text{len}(G_L) = \text{len}(G_R) \) then \( \text{ragen} := \text{ragen} \cup \langle G_L, G_R \rangle \)
12. end for
13. end for
14. end for
15. end while
16. return \text{ragen}
End
```
Complex Matching Selection

- False semantic matching due to coincidental correlations
  - (author) = [first name, last name] M1 correct one
  - (subject) = [first name, last name] M2 wrong one

The more negatively correlated → higher confidence to be real synonyms

- Strategy for ranking the discovered matchings:
  - Score function to evaluate the discovered matchings under measure
    - The goal of qualifying → correlation passes some threshold
    - The goal of ranking → compare the strength of correlations

C\text{max}, The maximal measure value among pairs of groups in a matching

---

Complex Matching Selection

- Strategy for tie breaking → richer semantic information
  - Take "top K" approach
  - If \( C \text{max} \) value of two groups are the same → compare their second highest to break the tie.
  - If two matching are tie after "top-k" comparison → the one with richer semantic information.

- Rule for ranking matches
  1) If \( s(M_1, m_2) > s(M_2, m_1) \)
  2) If \( s(M_1, m_2) = s(M_2, m_1) \) and \( M_1 \geq M_1 \)

---

Complex Matching Selection

- Greedy selection strategy by choosing the highest ranked matching \( M_i \) in each iteration.

---

Example

- (author) = [first name, last name] M1
- (subject) = [first name, last name] M2
- (title) = [first name, last name] M3
- (year) = [first name, last name] M4
- (source) = [first name, last name] M5
- (venue) = [first name, last name] M6
Outline

- Introduction
- Motivation of integrating the deep web
- Mining Algorithm
- H-Measure
- Data Preparation Step
- Experiments
- Discussion

Measures for Association Patterns

- Contingency table.

- Design of a correlation measure is empirical
- No good correlation measure
- Not uniform attribute frequency
Correlation Measure (Continued)

1. Non-uniform attributes → sparseness of schema data
2. Many attributes rarely used (occurring only once in the schemas) → rare attribute problem
3. Highly frequently attributes → frequent attribute problem

The chosen measure should be robust against these three problems.

Sparseness problem

\[ \text{Lift} = \frac{f_{a+b}}{f_{a} \cdot f_{b}} \]

- Lift measure includes the count of co-absence
- Not good measure for schema matching → exaggerate the effect of co-absence.

\[ \begin{array}{c|c|c|c|c} \hline A & B & C & D & \text{Co-absence} \\ \hline 1 & 0 & 1 & 0 & 2 \\ 0 & 1 & 0 & 1 & 1 \\ \hline \end{array} \]

Lift > 1 → positive correlation
Lift < 1 → negative correlation

\[ \text{Jaccard} = \frac{f_{a}}{f_{a} \cdot f_{b} \cdot f_{c}} \]

Rare attribute problem

Rare problem: either \( A \) or \( B \) is rarely observed → Measure should not consider \( A \) and \( B \) as highly negatively correlated.

\[ \begin{array}{c|c|c|c|c} \hline A & B & C & D & \text{Co-absence} \\ \hline 1 & 0 & 1 & 0 & 2 \\ 0 & 1 & 0 & 1 & 1 \\ \hline \end{array} \]

More Negatively correlated
H-Measure

\[ w_i(A_i, A_j) = H(A_i, A_j) = \frac{f_{1i1j}}{f_{1i}f_{1j}} \]

- H-measure is as the negative correlation \( R \)
- H value close to 0 \( \rightarrow \) highly degree of positive correlation
- H value to 1 \( \rightarrow \) high degree of negative correlation
- H-measure avoids the sparseness problem by ignoring \( f_{00} \)

Frequent attribute problem

- The ability to differentiate highly frequent attributes from really correlated ones.
- Generating false positives in group discovery

Outline

- Introduction
- Motivation of integrating the deep web
- Mining Algorithm
- H-Measure
- Data Preparation Step
- Experiments
- Discussion
Data Preparation

- **Form extraction**: extracting attributes entities (names and domains) from query interfaces
  - Standard Normalization: Stemming attribute names and domains
  - Normalizing irregular nouns and verbs e.g. children → child
  - Delete common stop words

- **Type recognition**
  - Problem of homonyms (the same name with different meanings) → distinguish both names and types
  - Type identification is not declared in web interfaces → type recognizer to recognize types from domain values of attributes entities

- **Syntactic Merging**
  - Name-based merging: merging two attribute entities if they are similar in names
  - Domain-based merging: merging two attribute entities if they are similar in domain values.

Outline

- Introduction
- Motivation of integrating the deep web
- Mining Algorithm
- H-Measure
- Data Preparation Step
- Experiments
- Discussion

Experiments

- Two datasets:
  - **TEL-8 dataset**
    - Raw web pages over 447 deep web sources in 8 domains.
    - Each domain has about 20-70 sources.
  - **BAMM dataset**
    - Manually extracted attribute names over 211 sources in 4 domains (around 50 sources per domain).
Experimental Result (result on TEL-8 Dataset)

Attributes above a frequency threshold T

Evaluating the H-measure

Conclusion

- Complex matching n-ary complex matching
- DCM framework
- Mining Algorithm
- H-measure
Discussion Points

- Their approach works for the same domain. How it can cross the domain boundary is an open problem.
- Choosing the threshold values is empirically. They did it by testing the algorithm with various thresholds. How it can be designed more systematically?
- H-Measure was derived based on the observation of the data. It should be systematic. Does it work the same if the datasets change?
- Compare their H-measure with other measures not only one.
- In removing the conflicted groups, how do they remove the conflicted one if both groups have the same score value?

Questions?