Bootstrapping pay-as-you-go data integration systems

Anish Das Sarma†
Stanford University
California, USA
anish@cs.stanford.edu

Xin Dong†
AT&T Labs–Research
New Jersey, USA
lunadong@research.att.com

Alon Halevy
Google Inc.
California, USA
halevy@google.com

SIGMOD '08: Proceedings of the 2008 ACM SIGMOD international conference on Management of data June 2008

Zihao Li, 2020.07.19

Problem description

- Many application contexts involve multiple data sources (e.g., the web, personal information management, enterprise intranets), how to integrate those data automatically?

EXAMPLE 1. Consider two source schemas both describing people:

1. S1(name, hPhone, hAddr, oPhone, oAddr)
2. S2(name, phone, address)

Clustering the attributes of S1 and S2:

Which mediate schema is perfect?

M1({name}, {phone, hP, oP}, {address, hA, oA})
M2({name}, {phone, hP}, {oP}, {address, oA}, {hA})
M3({name}, {phone, hP}, {oP}, {address, hA}, {oA})
M4({name}, {phone, oP}, {hP}, {address, oA}, {hA})
M5({name}, {phone}, {hP}, {oP}, {address}, {hA}, {oA})

Problem description

EXAMPLE 2. *Consider an instance of* S1 *with a tuple:*

S1: (name, hPhone, hAddr, oPhone, oAddr)
('Alice', '123-4567', '123, A Ave.', '765-4321', '456, B Ave.')

and a query:

SELECT name, phone, address FROM People

Which answer is perfect?

A1: ('Alice', '123-4567', '123, A Ave.')
A2: ('Alice', '765-4321', '456, B Ave.')
A3: ('Alice', '765-4321', '123, A Ave.')
A4: ('Alice', '123-4567', '456, B Ave.')

Motivation

Introducing the notion of *probabilistic schema mappings* which provides a foundation for answering queries in a data integration system with uncertainty about semi-automatically created mappings.

Integration:

Possible Mapping	Probability
{(name, name), (hP, hPP), (oP, oP), (hA, hAA), (oA, oA)}	0.64
{(name, name), (hP, hPP), (oP, oP), (oA, hAA), (hA, oA)}	0.16
{(name, name), (oP, hPP), (hP, oP), (hA, hAA), (oA, oA)}	0.16
{(name, name), (oP, hPP), (hP, oP), (oA, hAA), (hA, oA)}	0.04

Query – Answer:

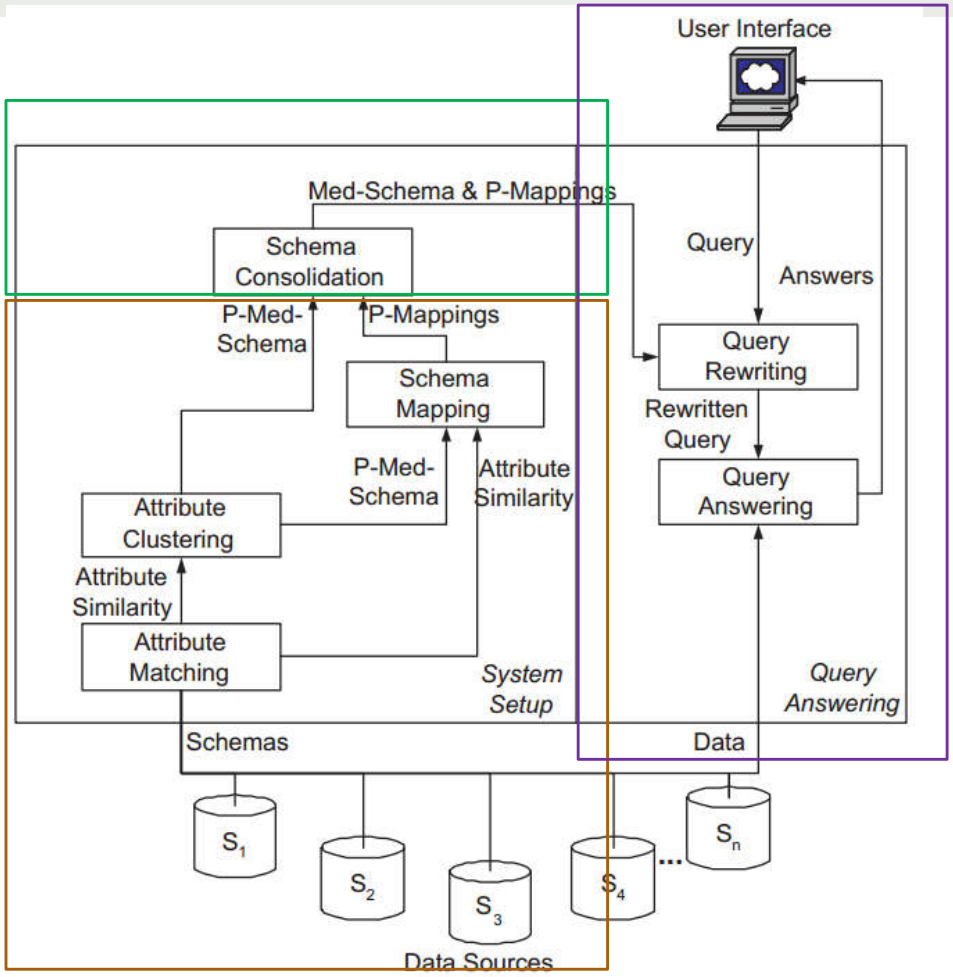
Answer	Probability
('Alice', '123-4567', '123, A Ave.')	0.34
('Alice', '765-4321', '456, B Ave.')	0.34
('Alice', '765-4321', '123, A Ave.')	0.16
('Alice', '123-4567', '456, B Ave.')	0.16

Architecture of our automatic-setup data integration system

Step 1. Construct a probabilistic mediated schema (Preparing the original data source and get the probability distribution).

Step 2. Find best probabilistic schema mappings (Integration based on probability).

Step 3. Create a single mediated schema to expose to the user (Query Answering based on probability)



Definition

The distribution of original probabilistic mediated schema from multi-data source

DEFINITION 3.1 (PROBABILISTIC MEDIATED SCHEMA). *Let* $\{S_1, \ldots, S_n\}$ *be a set of schemas. A* probabilistic mediated schema (p-med-schema) for $\{S_1, \ldots, S_n\}$ *is a set*

$$\mathbf{M} = \{(M_1, Pr(M_1)), \ldots, (M_l, Pr(M_l))\}$$

where

- *for each* $i \in [1, l]$, M_i *is a mediated schema for* $S_1, \ldots, S_n$, *and for each* $i, j \in [1, l], i \neq j$, M_i *and* M_j *correspond to different clusterings of the source attributes;*
- $Pr(M_i) \in (0, 1]$, *and* $\Sigma_{i=1}^{l} Pr(M_i) = 1$. □

Query – Answer pairs based on probability

The probability of each mapping schema

DEFINITION 3.2 (PROBABILISTIC MAPPING). *Let* S *be a source schema and* M *be a mediated schema. A* probabilistic schema mapping (p-mapping) *between* S *and* M *is a set*

$$pM = \{(m_1, Pr(m_1)), \ldots, (m_l, Pr(m_l))\}$$

such that

- *for each* $i \in [1, l]$, m_i *is a schema mapping between* S *and* M, *and for every* $i, j \in [1, l], i \neq j \Rightarrow m_i \neq m_j$;
- $Pr(m_i) \in (0, 1]$, *and* $\Sigma_{i=1}^{l} Pr(m_i) = 1$. □

DEFINITION 3.3 (QUERY ANSWER). *Let* S *be a source schema and* $\mathbf{M} = \{(M_1, Pr(M_1)), \ldots, (M_l, Pr(M_l))\}$ *be a p-med-schema. Let* $\mathbf{pM} = \{pM(M_1), \ldots, pM(M_l)\}$ *be a set of p-mappings where* $pM(M_i)$ *is the p-mapping between* S *and* M_i. *Let* D *be an instance of* S *and* Q *be a query.*

Let t *be a tuple. Let* $Pr(t|M_i), i \in [1, l]$, *be the probability of* t *in the answer of* Q *with respect to* M_i *and* $pM(M_i)$. *Let* $p = \Sigma_{i=1}^{l} Pr(t|M_i) * Pr(M_i)$. *If* $p > 0$, *then we say* (t, p) *is a by-table answer with respect to* $\mathbf{M}$ *and* $\mathbf{pM}$.

We denote all by-table answers by $Q_{\mathbf{M},\mathbf{pM}}(D)$. □

Mediate schema generation – Creating a single mediate schema

Step 1. delete noise data (rare data) based on threshold;

Step2. graph generation. Nodes are attributes and edges are the similarity of connecting notes, isolated notes were deleted (similarity is small than threshold);

Step3. Combine nodes based on uncertain edges;

Nodes similarity: Jaro–Winkler Similarity

0: **Input**: Source schemas $S_1, \ldots, S_n$.
Output: A set of possible mediated schemas.
1: Compute $\mathcal{A} = \{a_1, \ldots, a_m\}$, the set of all source attributes;
2: **for each** $(j \in [1, m])$
Compute frequency $f(a_j) = \frac{|\{i \in [1,n] | a_j \in S_i\}|}{n}$;
3: Set $\mathcal{A} = \{a_j | j \in [1, m], f(a_j) \geq \theta\}$; *//$\theta$ is a threshold*
4: Construct a weighted graph $G(V, E)$, where (1) $V = \mathcal{A}$, and (2) for each $a_j, a_k \in \mathcal{A}$, $s(a_j, a_k) \geq \tau - \epsilon$, there is an edge (a_j, a_k) with weight $s(a_j, a_k)$;
5: Mark all edges with weight less than $\tau + \epsilon$ as *uncertain*;
6: **for each** (uncertain edge $e = (a_1, a_2) \in E$)
Remove e from E if (1) a_1 and a_2 are connected by a path with only certain edges, or (2) there exists $a_3 \in V$, such that a_2 and a_3 are connected by a path with only certain edges and there is an uncertain edge (a_1, a_3);
7: **for each** (subset of uncertain edges)
Omit the edges in the subset and compute a mediated schema where each connected component in the graph corresponds to an attribute in the schema;
8: **return** distinct mediated schemas.

Algorithm 1: Generate all possible mediated schemas.

Mediate schema generation – Creating a p-med-schema

The calculation of the probability of mediated schemas (let frequency as probability):

0: **Input:** Possible mediated schemas $M_1, \ldots, M_l$ and source schemas $S_1, \ldots, S_n$.
Output: $Pr(M_1), \ldots, Pr(M_l)$.
1: **for each** $(i \in [1, l])$
Count the number of source schemas that are consistent with M_i, denoted as c_i;
2: **for each** $(i \in [1, l])$ Set $Pr(M_i) = \frac{c_i}{\sum_{i=1}^{l} c_i}$.

Algorithm 2: Assign probabilities to possible mediated schemas.

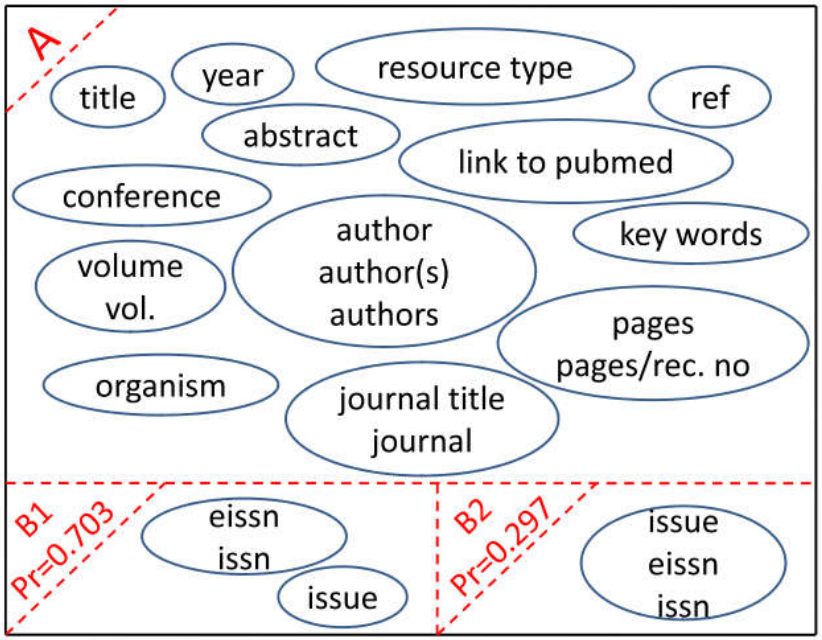


Figure 3: The p-med-schema for a set of bibliography sources. Each oval in the graph represents an attribute in the mediated schemas. The p-med-schema contains two possible schemas, the first containing attributes in regions `A` and `B1`, and the second containing attributes in regions `A` and `B2`. They have probabilities 0.703 and 0.297 respectively.

P-mapping generation — computing weighted correspondences

p_(i, j) is the correspondence of source attribute i and the mediated schema attribute j:

DEFINITION 5.1 (CONSISTENT P-MAPPING). *A p-mapping pM is* consistent with *a weighted correspondence $C_{i,j}$ between a pair of source and target attributes if the sum of the probabilities of all mappings $m \in pM$ containing correspondence (i, j) equals $p_{i,j}$; that is,*

$$p_{i,j} = \sum_{m \in pM, (i,j) \in m} \Pr(m).$$

A p-mapping is consistent with *a set of weighted correspondences $\mathbf{C}$ if it is consistent with each weighted correspondence $C \in \mathbf{C}$.* □

THEOREM 5.2. *Let $\mathbf{C}$ be a set of weighted correspondences between a source schema $\mathbf{S}(a_1, \ldots, a_m)$ and a mediated schema $\mathbf{M}(A_1, \ldots, A_n)$.*

- *There exists a consistent p-mapping with respect to $\mathbf{C}$ if and only if (1) for every $i \in [1, m]$, $\sum_{j=1}^{n} p_{i,j} \leq 1$ and (2) for every $j \in [1, n]$, $\sum_{i=1}^{m} p_{i,j} \leq 1$.*
- *Let*

$$M' = max\{max_i\{\sum_{j=1}^{n} p_{i,j}\}, max_j\{\sum_{i=1}^{m} p_{i,j}\}\}.$$

Then, for each $i \in [1, m]$, $\sum_{j=1}^{n} \frac{p_{i,j}}{M'} \leq 1$ and for each $j \in [1, n]$, $\sum_{i=1}^{m} \frac{p_{i,j}}{M'} \leq 1$. □

Based on Theorem 5.2, we normalize the weighted correspondences we generated as described previously by dividing them by M'; that is,

$$p'_{i,j} = \frac{p_{i,j}}{M'}.$$

P-mapping generation — Generating p-mappings

Based on the correspondence of each attribute pairwise (i, j), then how to realize the assignment (mapping) of each attribute from source data to intermediate mediate ?

Maximum cross entropy:

Given the possible mappings $m_1, \ldots, m_l$, we assign probabilities $p_1, \ldots, p_l$ to $m_1, \ldots, m_l$ by solving the following constraint optimization problem (OPT):

maximize $\sum_{k=1}^{l} -p_k * \log p_k$ subject to:

1. $\forall k \in [1, l], 0 \leq p_k \leq 1$,
2. $\sum_{k=1}^{l} p_k = 1$, and
3. $\forall i, j : \sum_{k \in [1,l], (i,j) \in m_k} p_k = p_{i,j}$.

Using Knitro (software) to solve the entropy maximization problem in p-mapping construction.

P-mediated-schema consolidation

Combine the multi mediate schema to create a integrated schema T:

0: **Input:** Mediated schemas $M_1, \ldots, M_l$.
Output: A consolidated single mediated schema T.
1: Set $T = M_1$.
2: **for** $(i = 2, \ldots, l)$ modify T as follows:
3: **for each** (attribute A' in M_i)
4: **for each** (attribute A in T)
5: Divide A into $A \cap A'$ and $A - A'$;
6: **return** T.

Algorithm 3: Consolidate a p-med-schema.

EXAMPLE 6.1. *Consider a p-med-schema* $M = \{M_1, M_2\}$, *where* M_1 *contains three attributes* $\{a_1, a_2, a_3\}$, $\{a_4\}$, *and* $\{a_5, a_6\}$, *and* M_2 *contains two attributes* $\{a_2, a_3, a_4\}$ *and* $\{a_1, a_5, a_6\}$. *The target schema* T *would then contain four attributes:* $\{a_1\}$, $\{a_2, a_3\}$, $\{a_4\}$, *and* $\{a_5, a_6\}$. □

Experiments

Experiment dataset: tables from five domain:

Table 1: Number of tables in each domain and keywords that identify the domain. Each domain contains 50 to 800 data sources.

Domain	#Src	Keywords
Movie	161	*movie* and *year*
Car	817	*make* and *model*
People	49	*name*, one of *job* and *title*, and one of *organization, company* and *employer*
Course	647	one of *course* and *class*, one of *instructor, teacher* and *lecturer*, and one of *subject, department* and *title*
Bib	649	*author, title, year*, and one of *journal* and *conference*

Experiments

Mapping experiment, compared with manual results (golden standard)

Table 2: Precision, recall and F-measure of query answering of the UDI system compared with a manually created integration system. The results show that UDI obtained a high accuracy in query answering.

Domain	Precision	Recall	F-measure
Golden standard			
People	1	.849	.918
Bib	1	.852	.92
Approximate golden standard			
Movie	.95	1	.924
Car	1	.917	.957
Course	.958	.984	.971
People	1	1	1
Bib	1	.955	.977

The time to set up the system

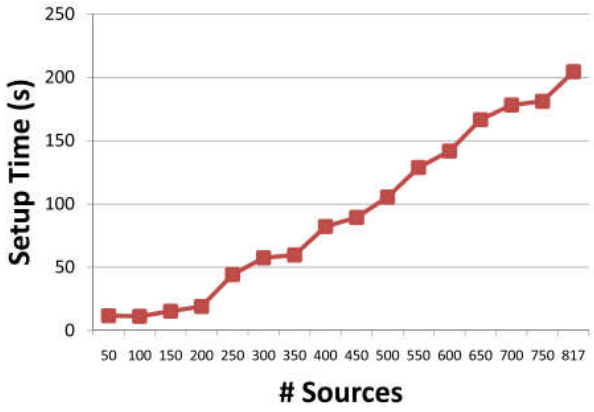


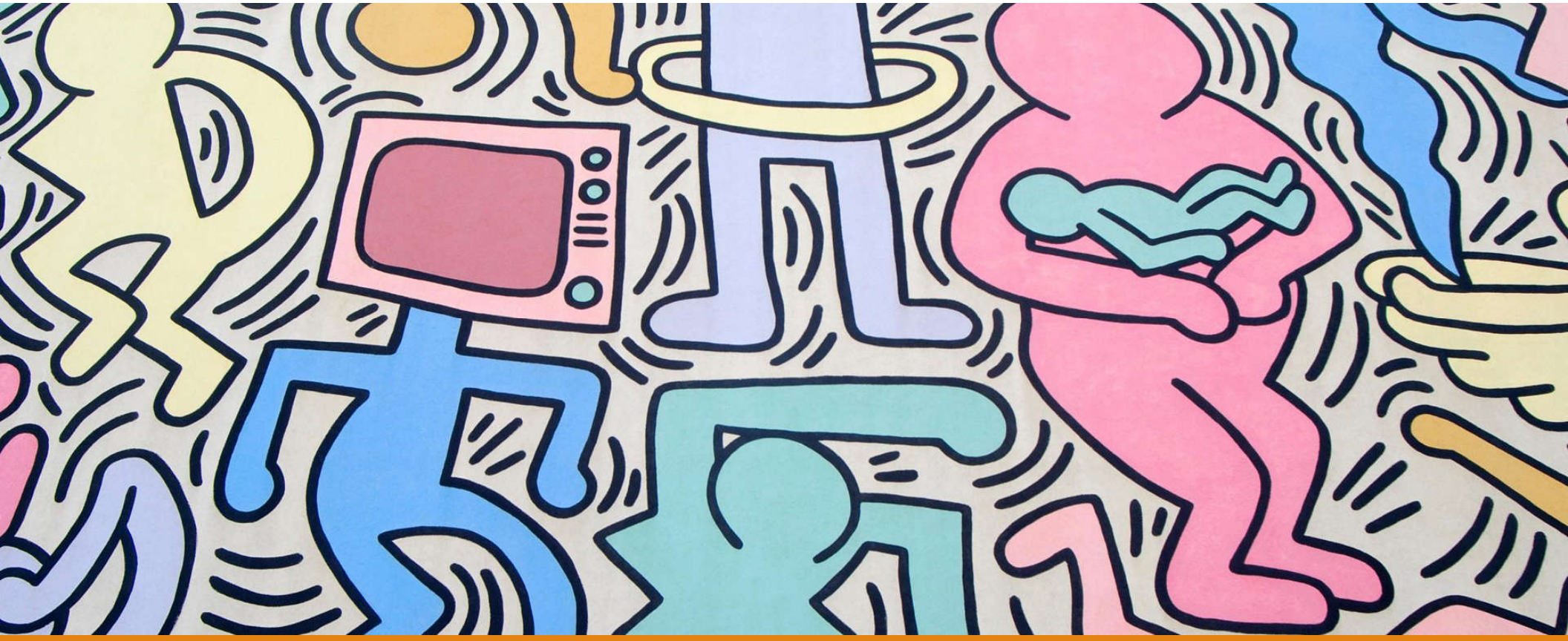
Figure 7: System setup time for the Car domain. When the number of data sources was increased, the setup time increased linearly.

Conclusion

- Automatically set up a data integration application that obtains answers with high precision and recall;
- Time should be optimized in the future work;
- Human interaction or feedback should be considered in the integration process;

Reference

- Das Sarma A, Dong X, Halevy A. Bootstrapping pay-as-you-go data integration systems[C]//Proceedings of the 2008 ACM SIGMOD international conference on Management of data. 2008: 861-874.
- Jaro–Winkler distance – Wikipedia. https://en.wikipedia.org/wiki/Jaro%E2%80%93Winkler_distance
- Dong X L, Halevy A, Yu C. Data integration with uncertainty[J]. The VLDB Journal, 2009, 18(2): 469-500.
- Jeffery S R, Franklin M J, Halevy A Y. Pay-as-you-go user feedback for dataspace systems[C]//Proceedings of the 2008 ACM SIGMOD international conference on Management of data. 2008: 847-860.
- Franklin M, Halevy A, Maier D. From databases to dataspaces: a new abstraction for information management[J]. ACM Sigmod Record, 2005, 34(4): 27-33.



Q&A

Thanks

Appendix

Jaro Similarity

The Jaro Similarity sim_j of two given strings s_1 and s_2 is

$$sim_j = \begin{cases} 0 & \text{if } m = 0 \\ \frac{1}{3}\left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m}\right) & \text{otherwise} \end{cases}$$

Where:

- $|s_i|$ is the length of the string s_i;
- m is the number of *matching characters* (see below);
- t is half the number of *transpositions* (see below).

Two characters from s_1 and s_2 respectively, are considered *matching* only if they are the same and not farther than $\left\lfloor \frac{\max(|s_1|, |s_2|)}{2} \right\rfloor - 1$ characters apart.

Each character of s_1 is compared with all its matching characters in s_2. The number of matching (but different sequence order) characters divided by 2 defines the number of *transpositions*. For example, in comparing CRATE with TRACE, only 'R' 'A' 'E' are the matching characters, i.e. m=3. Although 'C', 'T' appear in both strings, they are farther apart than 1 (the result of $\lfloor \frac{5}{2} \rfloor - 1$). Therefore, t=0 . In DwAyNE versus DuANE the matching letters are already in the same order D-A-N-E, so no transpositions are needed.

Appendix

Jaro–Winkler Similarity [edit]

Jaro–Winkler similarity uses a prefix scale p which gives more favorable ratings to strings that match from the beginning for a set prefix length ℓ. Given two strings s_1 and s_2, their Jaro–Winkler similarity sim_w is:

$$sim_w = sim_j + \ell p(1 - sim_j),$$

where:

- sim_j is the Jaro similarity for strings s_1 and s_2
- ℓ is the length of common prefix at the start of the string up to a maximum of four characters
- p is a constant scaling factor for how much the score is adjusted upwards for having common prefixes. p should not exceed 0.25, otherwise the similarity could become larger than 1. The standard value for this constant in Winkler's work is $p = 0.1$

The Jaro-Winkler distance d_w is defined as $d_w = 1 - sim_w$.

Although often referred to as a *distance metric*, the Jaro–Winkler distance is not a metric in the mathematical sense of that term because it does not obey the triangle inequality.[1] The Jaro–Winkler distance also does not satisfy the identity axiom $d(x, y) = 0 \leftrightarrow x = y$.