Comparing the Effectiveness of Deterministic Matching with Probabilistic Matching for Entity Resolution of Student Enrollment Records

(Completed Research Paper)

Aziz Eram, Abdul Ghani Mohammed, Vishnu Pillai, and John R. Talburt,
University of Arkansas at Little Rock
axeram@ualr.edu, ggmohammed@ualr.edu, vspillai@ualr.edu, jrtalburt@ualr.edu,

Abstract: Entity resolution (ER) helps organizations understand, organize, analyze the data at an entity (customer, vendor, product, …) level to increase the quality of the data. This paper describes a comparison between two entity resolution processes running against the same school enrollment data. The first process (the benchmark process) is organization's current process. The benchmark process uses Boolean (if-then/deterministic) matching rules that rely heavily on the presence of a highly indicative value (social security number) in the record in addition to name and date-of-birth information. In the research presented here, a second ER process was designed based on a scoring rule (probabilistic matching) using only the name and date-of-birth information. The purpose of the research is to show how a probabilistic approach can achieve matching accuracy comparable to the benchmark deterministic process while avoiding the use of sensitive data. The results of the research presented here show that for this organization it would be feasible to replace their current Boolean matching with probabilistic matching using only name and date-of-birth information and achieve almost the matching accuracy.

Key Words: Entity Resolution, Boolean Rules, Scoring Rules, Probabilistic Matching, Deterministic Matching, Talburt-Wang Index.

Background:

Data has been gaining importance since the past two decades and attempts are being made to use it as information asset that meets the expectations of the users. "Information Quality is a discipline that is concerned with the maximization of value and minimizing the risk of the organizations information assets" [6] .These assets include data of various functions one of which is Master Data. Master Data are sometimes regarded as the "nouns" of the data vocabulary [1, 9]. Master data describe people, places and things that are critical to the organization.

A master record describes a particular real-world entity, such as a product, a patient, or a student. Entity Resolution (ER) is the process of determining whether two records in an information system are referencing (describing) the same world object or different world objects. If two records are referencing the same entity, then they are said to be equivalent references [6] .

Entity Resolution is foundational to Master Data Management (MDM), a collection of policies, procedures, services and infrastructure to support the capture, integration and shared use of accurate, timely, consistent and complete master data describing an entity [6] . Each real-world entity has a set of characteristics called identity attributes. The values of the identity attribute help to differentiate one particular entity for all other entities of the same type. Familiar examples of identity attributes for person entities include name, date-of-birth, mailing address, telephone number, and employer.

The ER process is driven by the notion of similarity. The fundamental assumption of ER is that "the higher the degree of similarity between two entity references, the higher the likelihood that the references are equivalent" [9] . Although new approaches to ER are being developed, the majority of systems currently in use employ one of two techniques – deterministic matching or probabilistic matching [3] .

- A Boolean rule is otherwise known as deterministic matching use "if-then" logic with Boolean operators "and/or" to form a Boolean expression to decide if two references with similar attributes reference the same entity or not. The result is a binary (True or False) decision. If the decision is False, the references are deemed not equivalent. If the decision is True, they are deemed equivalent [5]

  For example: (IF the first names have the same Soundex value AND the last names are the same AND the dates-of-birth are the same) OR (IF the first names are the same AND the last names are the same AND the street numbers of the address are the same) THEN link as matching records, ELSE no match.

- A Scoring Rule, otherwise known as probability matching is based on a weighted scoring formula. Each attribute has two types of numeric values, an agreement weight (or weights) and a smaller disagreement weight. When two records are compared, the agreement and disagreement weights are added to give a total score. The more similar the records are, the more attribute agreements, and the higher the total score. Disagreement weights are often negative values. The total score for attributes is compared to a preset "match score." If the total meets or exceeds the match score, the references are considered a match and are linked, otherwise the references are not linked.

  For example: IF the first names agree, add 100 to the total score, else add 20, IF the last names agree, add 150 to the score, else add -10, IF the street numbers agree, add 50, else add 5, IF the dates-of-birth agree add 60, else add -40. If a pair of references has a total score is 215 and the match score is set at 200, then the references would be linked.

There are four main differences between the Boolean and scoring rule approaches.

1. The first is that probabilistic matching uses a match threshold or match score, the minimum total score that must be obtained for the records to be considered a match and linked. The threshold can be changed independently of the scoring rule itself. The same weighted scoring rule will give different linking results when different match score values are set.
2. Using the match score as a threshold allows the ER system to easily identify "close matches", i.e. pairs of references with scores just below or just above the match score. This can be important for quality control in which pairs of references with scores falling within this range are reported to data stewards for verification.
3. In the scoring rule all of the attributes are considered for each pair of references whereas each clause in a Boolean rule typically on matches a subset of the identity attributes.
4. The most important difference is that in the scoring rule, the same attribute can different agreement weights associated with different values of the attribute. The agreement weight assigned to an attribute value depends on how probable an agreement on the value will be an indicator that references are equivalent, thus the name "probabilistic matching." For example, in a typical population of U.S. students, the fact that two records agree the first name "JACOB" is not a strong indicator of equivalence because this name very common and shared by many different students. Therefore, the agreement weight for the first name value of "JACOB" would be lower than the agreement weight for a less common first name such as "AMARYLLIS." The formula for

calculating the agreement and disagreement weights from these probabilities is given later in this paper.

Problem Statement:

The Arkansas Department of Education (ADE), a state service agency that provides leadership support and service to schools, districts and communities so that every student that graduates is prepared for college, career and community engagement. ADE has five major divisions. One of these is the Information Technology Division which provides a complete set of tools and applications for its stakeholders. All of these tools and applications are accessible through the ADE Data Center. Currently ADE uses an entity resolution engine known as Proteus. Proteus uses Boolean rules for its matching strategy. The system is highly accurate because it is able to use the student's social security number as one of its matching attributes. However, ADE anticipates in the near future policy changes will prevent the collection and use student social security numbers. Preliminary tests have shown that when the social security number is removed, the Proteus results are much less accurate in bringing together all of the enrollment records for the same student.

The objective of the research described here was to see if the probabilistic matching rule's ability to use different weights for different values of the same attribute could compensate for the loss in accuracy caused by not using the social security number.

Approach

The tests of Proteus versus the OYSTER probabilistic scoring rule were run on copies of live data processed nightly by ADE. The nightly process comprised of all enrollment updates for each day. For our work we used the enrollment records for three consecutive nightly updates selected at random.

The enrollment records were first processed by Proteus to apply its student identifier using Boolean matching. The Proteus output was then processed by the OYSTER system to apply its student identifier based on probabilistic matching. The three link files obtained from each run were combined into a single analysis file of 509,365 records. Each enrollment record in the analysis file had two student identifiers, one assigned by Proteus the other by OYSTER. In this case, the Proteus results were the benchmark, and the analysis file allowed us to directly compare how similar the OYSTER results were to the Proteus results.

It is important to note that the Proteus linking results only provided an accuracy benchmark, not an actual "truth set." Like all ER systems, the Proteus system has been observed to makes some number of false positive and false negative linking errors. The goal was not to measure the accuracy of either the Proteus or OYSTER results, but to see if the probabilistic matching performed by OYSTER could achieve the same or similar linking results as Proteus. That means that even though the probabilistic results of OYSTER may differ somewhat from Proteus, it does not follow that the OYSTER results are necessarily less accurate. The differences in linking made by probabilistic matching in some cases may be because OYSTER made the correct linking decision where Proteus made an incorrect decision. Even though the exact accuracy, i.e. precision and recall measures, of Proteus linking were not calculated, the organization believed its results were acceptable and therefore provided a "surrogate" truth set for comparison in this research.

1. Understanding the data:

The assessments include profiling the sources with a data quality profiling tool to understand the structure and general condition of the data and visually inspecting the records in order to find specific conditions [4].

The assessment will

- Identify which attributes are present in each source.
- Generate statistics, such as uniqueness and missing value counts that will help to decide which attributes will have highest value as identity attributes.
- Assess the extent of data quality problems in each source as:
  - Missing values, null values, empty strings, blank values and place-holder values.
  - Inconsistent representation of values
  - Misfielding of attributes.

The daily enrollment data used in the test had 19 attributes:

1. Student First Name
2. Student Middle Name
3. Student Last Name
4. Student Social Security Number (SSN) Sometime these are made-up numbers and not the actual SSN of the student)
5. Student Date-of-Birth (DOB)
6. Student Gender
7. Unique Record ID
8. District Lea (student identifier assigned at the district level)
9. TID (third-party student identifier)
10. Student Race
11. Student Grade
12. Parent First Name
13. Parent Last Name
14. Parent Street Address
15. Parent City Address
16. Parent Zip Code
17. Twin Indicator
18. Entry Date
19. Withdrawal Date.

Table 1 shows the uniqueness and completeness of these attributes.

	COLUMN NAMES	MISSING	UNIQUES
	First Name	2	35,028
	Middle Name	1	35,850
	Last Name	1	35,870
	SSN	1	245,279
	DOB	1	6,293
	Gender	1	11
	Record ID	1	258,909
	DistrictLea	1	265
	TID	2	245,219
	Race	4	38
	Grade	7	19
	Parent First Name	9	55,320
	Parent Last Name	9	33,880
	Parent Address	9	201,302
	Parent Zip	9	915
	Parent City	9	2,266
	Twin	9	9
	Entry Date	9	48
	With drawl Date	9	47

Table 1: Profile Statistics for 1 Nightly Data File of 258,947 Records

The test data also had other data quality problems including mis-keyed values, inconsistent representation, and misfielded values.

For example

- “Jacob” also represented as “Jacbo”
- “1701 West Park Dr” also represented as “1701 West Park Drive”
- The first, middle, last names “Mary Ann”, “”, “Smith” also represented as “Mary”, “Ann”, “Smith”

In the end, we decided to use only the three name attributes along with date-of-birth for the probabilistic matching. The SSN was specifically excluded because of the test objective. The TID was excluded because it was a third-party, proprietary identifier and primarily derived using the SSN. Some of the other non-sensitive attributes may later prove to be useful, but our research demonstrated that applying probabilistic matching to just the name and DOB values was sufficient to attain the same levels of accuracy as the current system relying primarily on the SSN.

2. Data Preparation and Proteus Rules

The data preparation was already being done by Proteus before it before the data were made available to us for processing with the OYSTER system. Proteus changed all the lower cases to upper cases, standardizing date of birth into CCYYMMDD format, removed punctuation marks, and stored all the other attributes (Twin Indicator, Entry and Withdrawal date, etc..) in a single column as remaining data. Proteus also generated its own student identifier using three simple Boolean matching rules.

Rule 1: (First Name) and (Last Name) and (DOB) must all be exact matches, or

Rule 2: (First Name) and (SSN) must be exact matches, or
Rule 3: (Last Name) and (SSN) must be exact matches

3. Developing the Scoring Rule

The rule developed in this paper is a weighted scoring rule where the weights are based on probabilities that certain attribute values are a stronger (higher weight) or weaker (lower weight) indicator two enrollment records are for the same student. The steps followed to build a scoring rule are as follows:

- Determine Attribute Weights
- Determine The Match Score

The attribute weight is a measure of the discrimination power of an attribute. Given n identity attributes a_1, a_2, ...… a_n the attribute weights are calculated by

$$w_j = \begin{cases} log_2\left(\frac{m_j}{u_j}\right), if\ agreement\ on\ the\ j-th\ attribute \\ log_2\left(\frac{(1-m_j)}{(1-u_j)}\right), if\ disagreement\ on\ the\ j-th\ attribute \end{cases}$$

Where m_j = Probability (tow references agree on a_j given that the references are equivalent)
u_j = Probability (two references agree on a_j given that the references are not equivalent)

We calculated different attribute weights for first name, middle name, last name, and DOB as they were selected as identity attributes. There were 4 different attribute weight calculated for each attribute

- Overall agreement weight for the attribute
- Agreement weights for frequently occurring values of the attribute
- Overall agreement weight for all infrequently occurring values of the attribute
- Overall disagreement weight for the attribute

Overall Agreement Weight:

The Attribute agreement weight is calculated as follows:

- Count the number of pairs of equivalent references and the number of pairs of non-equivalent references
- Count the pairs of references agreeing on attribute A
- Count the pairs of equivalent references agreeing on attribute A
- Count the pairs of non-equivalent references agreeing on attribute A
- Compute probability (Pairs agree on A| references are equivalent)
- Compute probability (Pairs agree on A| references are not equivalent)
- Compute the logarithm of the ratio of these two probabilities

For example, First Name:
Number of pairs of equivalent references = 1,766,510
Number of pairs of non- equivalent references = 1,169,526,598,993
Equivalent pairs agreeing on attribute A = 1,766,387
Non-equivalent pairs of agreeing on A = Pairs agreeing on A – Equivalent pairs agreeing on A
= 1,444,687,765

Probability Numerator = References agree on A and equivalent/ Number of Equivalent records
= 0.9997
Probability Denominator = References agree on A and non-equivalent / Number of non-equivalent records
=0.00123
Probability = Probability Numerator/ Probability Denominator
= 810.33
Agreement Weight = Log_2(Probability)
= Log_2(810.33)
= 9.66
Therefore, the overall agreement weight obtained for first name is 9.66

Following the same pattern, the agreement weights obtained for other attributes are shown in Table 2. In addition, to accommodate the OYSTER requirement for weights to be expressed as integer values, the actual weights were scaled by a factor of 100.

Attribute	**Agreement Weight**	**Scaled Weight x 100**
First Name	9.66	966
Middle Name	7.73	773
Last Name	9.73	973
Date of Birth	12.08	1,208

Table 2: Overall Agreement Weights

Value-Specific Weights for Frequently Occurring Values:

The real power of probabilistic matching is achieved by calculating the weights at the value level rather than at the overall attribute level. The previous calculation shows that overall, agreeing on a first name should have a scoring weight of 9.66. However, agreeing on some first names should be given a higher weight then agreeing on other first names. If a first name is relatively rare (infrequent) in a given population, then we two references share that first name they are more likely to indicate the references are for the same student (equivalent references). Conversely, very frequently used names should have a smaller scoring weight because they are more like to be shared by different students.

The most common approach is to profile the most commonly occurring values and give each an individual weight. All other values are given the same “infrequent” weight. For this study frequent values were those occurring more than 10,000 times. The computation of a frequently occurring value weight is very similar to the calculation of the overall attribute agreement weight. The only difference is we determine equivalent records agreeing on a particular value, e.g. all equivalent or non-equivalent pairs of references agreeing on the first name “JOHN”. The Frequent Individual Agreement weight is calculated as follows:

For Example, the first name value “EMMA”
Number of pairs of equivalent references = 23,936
Number of pairs of non- equivalent references = 56,490,460
Probability Numerator = references agree on EMMA and equivalent/ Number of Equivalent records
= 0.0041531444244869
Probability Denominator = references agree on EMMA and non-equivalent / Number of non-equivalent records
= 1.73692446803008e-05

Probability = Probability Numerator/ Probability Denominator
= 239.109097771946

Agreement Weight = Log$_2$(Probability)
= Log$_2$(239.109097771946)
= 7.90

Therefore, the agreement weight for EMMA is 7.90

Following the same pattern, Table 3 shows the weights calculated for some of the highest frequency names found the in the dataset.

Attribute	Agreement Weight	Scaled Weigh x100
EMMA	7.90	790
ETHAN	7.55	755
HANNAH	7.74	774
JACOB	7.16	716

Table 3: Examples of Weights for High-Frequency Names

Overall Agreement Weight excluding frequent names (Uncommon Weight):

The overall agreement weight excluding frequent names is same as the overall agreement weight, the only difference is before computing equivalent and non-equivalent records we exclude all the frequent names occurring in each attribute from file and work on the remaining records following the same procedure explained in overall agreement weight.

Overall Disagreement Attribute Weight:

A disagreement weight refers to the proportion by which two similar attribute types of different references tend to disagree with each other. It is calculated by assigning weights to those references.

The overall disagreement weight can be easily computed by using the complements of the overall agreement weight. The formula to compute disagreement weight is:

Disagreement Weight = Log$_2$$\left(\frac{1-(probabilty(x\ values\ agree\ in\ pair\ t|t\ \in E))}{1-(probability(\text{x values agree } in \text{ pair y } |t\ \in N))}\right)$

For Example, disagreement weight of First Name:
Following the same way as agreement weight, the probabilities obtained are follows:

Probability Numerator = Equivalent pairs agreeing on A / Number of equivalent pairs
= 0.00123

Probability Denominator = Non-equivalent pairs agreeing on A / Number of non-equivalent pairs
=0.9997

Disagreement Weight (First Name) = Log$_2$((1-0.9997) /(1-0.00123))
= Log$_2$(0.000300369)
= -11.70

Therefore, the disagreement weight is -11.70, which is scaled for OYSTER to -1170.

The final step in developing the scoring rule is to set the Match Score. The match score is the minimum total value of the agreement/disagreement weights that two references must attain in order to be considered a match, i.e. given the same student identifier. In our case, the match score was set by running the scoring rule with different match score values to find the match score that gave the closest results to the Proteus system as explained in the next section.

```
<ScoringRule Ident="Test" MatchScore="200" ReviewScore="0">
<Term Item="FirstName" Similarity="EXACT" DataPrep="Scan(LR, Letter, 0, ToUpper, SameOrder)"
AgreeWgt="1002" WgtTable="D:\ERAM\Oyster 3.4\data\OneFileFN.txt" DisagreeWgt="-1170"
Missing="0" />
<Term Item="MiddleName" Similarity="EXACT" DataPrep="Scan(LR, Letter, 0, ToUpper,
SameOrder)"
AgreeWgt="937" WgtTable="D:\ERAM\Oyster 3.4\data\OneFileMN.txt" DisagreeWgt="-297"
Missing="0" />
<Term Item="LastName" Similarity="EXACT" DataPrep="Scan(LR, Letter, 0, ToUpper, SameOrder)"
AgreeWgt="1070" WgtTable="D:\ERAM\Oyster 3.4\data\OneFileLN.txt" DisagreeWgt="-1323"
Missing="0" />
<Term Item="DateOfBirth" Similarity="EXACT" DataPrep="Scan(LR, Digit, 0, ToUpper, SameOrder)"
AgreeWgt="1209" DisagreeWgt="-1208" Missing="0" />
</ScoringRule>
```

Figure 1: OYSTER Script to Implement the Scoring Rule with Match Score Set at 200

4. Comparing Proteus and OYSTER Results:

We selected Talburt-Wang Index as our assessment method, as computation of TWi is much simpler and does not require the actual calculation of the True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN) counts even for large data sets [2] . Instead it measures the number of overlaps between two partitions formed by different linking operations.

If S is a set of operations and A, B are partitions of S created by separate ER processes, then define V, the set of overlaps between A and B as

$$V = \{A_i \cap B_j \mid A_i \cap B_i \neq Ø\}$$

Then the Talburt-Wang Index is given as:

$$\text{TWi} = \frac{\sqrt{|A||B|}}{|V|}$$

Where

- |A| is number of cluster created by ER Process 1 (Proteus clusters)
- |B| is number of cluster created by ER Process 2 (OYSTER clusters)
- |V| is number of overlaps between the two sets of clusters

The Talburt- Wang Index value always lies between 0 and 1 and only has value of 1 when the decisions on equivalent and non-equivalent are the same for both ER processes [7, 8] .

We followed a trial and error process to determine best match score. We considered different scenarios of agreement and disagreement weights along with their highest and lowest frequent agreement weights.

Attribute	Frequent Individual Weights		Overall Agreement Weight	Uncommon Weight	Overall Disagreement Weight
	Highest	Lowest			
FIRST NAME	807	705	966	1002	-1,170
MIDDLE NAME	775	360	773	937	-297
LAST NAME	825	642	973	1,070	-1,323
DATE OF BIRTH			1,209		-1,208

Table 4: Agreements and Disagreement Weights

The first scoring rule was built using a match score of 200, then adding 200 to the match score for each subsequent run. The results are shown in Table 5.

Match Score	TWi
200	0.295
400	0.4
600	0.65
800	0.78
900	0.88
1,200	0.99
1,400	0.876
1,600	0.79
1,800	0.62
2,000	0.48
2,200	0.35

Table 5: Match Score Settings and Resulting TWi

When Table 5 of TWi where plotted in a graph (Figure 2) it takes the shape of a bell-shaped curve, which is sometimes referred to as a Receiver Operating Characteristic Curve (ROC). ROC curves are often used to illustrate the performance of a binary classifier system as its discrimination threshold is varied.

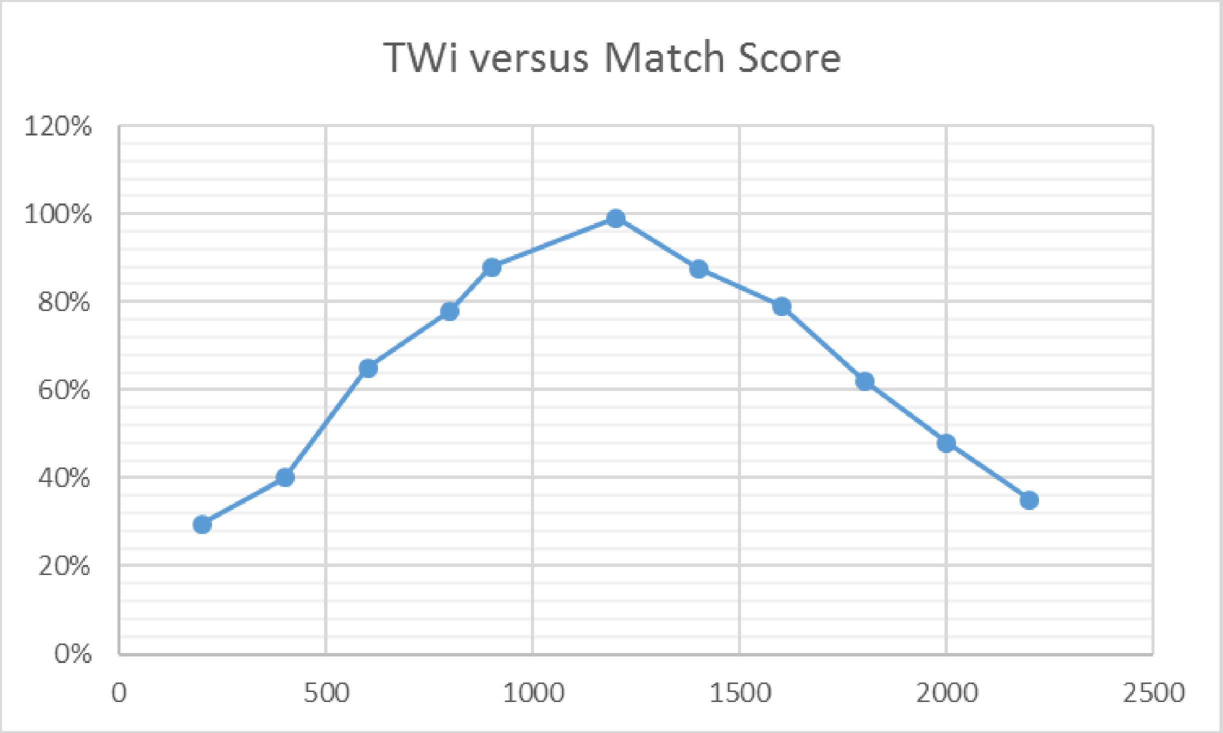

Figure 2: Table 5 Rendered as TWi as a Function of Match Score.

Using the TWi as the comparison measure, our research found a 99% similarity between the Proteus linking and the probabilistic linking was obtained when the match score in OYSTER was set at 1200. The Probabilistic scoring rule produced almost same number of matches without using SSN as of Boolean rules which used SSN as one of the important attribute to match entities.

Conclusion

A new scoring rule was successfully designed and tested in OYSTER to match student data without using an important identity attribute (SSN). In this research, we have compared two Entity Resolution systems which were using two different rules to match student enrollment data. Proteus used Boolean rules with SSN as an identity attribute producing 1,766,510 equivalent records and OYSTER used Scoring rule without SSN as an identity attribute producing 1,758,744 equivalent records. The new scoring rule design was able to produce ER results that were 99% similar to the Proteus results. The research demonstrates the power of probabilistic matching using value-based weights for entity resolution on student enrollment data.

Acknowledgement

The research presented here was supported in part by a grant from the Arkansas Department of Education (ADE) and logistical support from Mr. Arijit Sarkar, ADE’s Chief Information Officer.

References:

1. Bhattacharya, I., & Getoor, L. (2006, April). A latent dirichlet model for unsupervised entity resolution. In *Proceedings of the 2006 SIAM International Conference on Data Mining* (pp. 47-58). Society for Industrial and Applied Mathematics.

2. Hashemi, R., Talburt, J., Wang, R., 2006. Significance test for the Talburt-Wang Similarity Index. In: Talburt, J., Pierce, E., Wu, N., Campbell, T. (Eds.), *11th MIT International Conference on Information Quality*. MIT IQ Publishing, Cambridge, MA, pp. 125e132.

3. Herzog, T. N., Scheuren, F. J., & Winkler, W. E. (2007). *Data quality and record linkage techniques*. Springer Science & Business Media.

4. Spiegel, M. F., & Winslow, E. (2002, August). Database preprocessing and human-interface issues in reverse directory assistance (ACNA) services. In *Interactive Voice Technology for Telecommunications Applications, 1996. Proceedings., Third IEEE Workshop on* (pp. 105-110). IEEE.

5. Syed, H., Talburt, J., Liu, F., Pullen, D., & Wu, N. (2012, January). Developing and refining matching rules for entity resolution. In *Proceedings of the International Conference on Information and Knowledge Engineering (IKE)* (p. 1). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp).

6. Talburt, J. R. (2011). *Entity resolution and information quality*. Elsevier.

7. Talburt, J. R., & Zhou, Y. (2015). *Entity Information Life Cycle for Big Data: Master Data Management and Information Integration*. Morgan Kaufmann.

8. Talburt, J., Wang, R., Hess, K., & Kuo, E. (2007). An algebraic approach to data quality metrics for entity resolution over large datasets. *Information quality management: Theory and applications*, 1-22.

9. Winkler, W.E., 1988. Using the EM algorithm for weight computation in the Fellegi-Sunter model of record linkage. *Journal of the American Statistical Association*, Proceedings of the Section on Survey Research Methods 667-671.