String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage

William E. Winkler, U.S. Bureau of the Census*
Stat. Research Div., Rm 3000-4, Washington, DC 20223

ABSTRACT

This paper describes a string comparator metric that partially accounts for typographical variation in strings such as first name or surname, decision rules that utilize the string comparator, and improvements in empirical matching results. The string comparators are used in production computer matching software during the Post Enumeration Survey for the 1990 Census. The Post Enumeration Survey will use capture/recapture and other statistical techniques to produce a set of adjusted Census counts.

1. INTRODUCTION

Locating matches across a pair of lists not having unique identifiers such as a social security number is often difficult. Typically available identifiers such as first name, last name, and various demographic, economic, or address components may not uniquely identify matches because of legitimate variations.

Some types of legitimate variations in identifiers generally require a priori knowledge that allows rapid, accurate utilization. Such variations might take the forms Mrs W M Smith and Elizabeth Smith. Typographical variations such as Elizabeth Smith versus Elzbath Smoth are a special case of legitimate variations. They are more easily dealt with if suitable methods of comparing strings are available and are the only variations that we will consider in this paper.

If S_1 and S_2 are two strings, a string comparator Ψ merely maps the pair (S_1,S_2) to the closed interval [0,1]. A string comparator is not necessarily a metric in the mathematical sense and the restriction of its range to [0,1] is done primarily for convenience. Generally, we want pairs of strings that agree exactly to be assigned value 1, pairs of strings that agree almost exactly (in some sense) to have values close to 1, and strings that entirely disagree (in some sense) to have value 0.

A simple example of a string comparator is a function that assigns value 1 to a pair of strings that agree exactly or agree exactly on a code such as Soundex and, otherwise, assigns value 0. Another example would be a properly normalized Damerau-Levenstein metric that accounts for the number of insertions and deletions it takes to get from one string to another (see e.g., Winkler 1985).

This paper provides a class of string comparator metrics for comparing partially agreeing strings that extend the Jaro string comparator (see e.g., Winkler 1985). It formally shows how general methods of accounting for partial agreement fit in with the Fellegi-Sunter (1969) model of record linkage. It provides a formal method of modelling how to adjust matching weights between pure agreement and pure disagreement. The methods are dependent on having a representative set of matching pairs.

The second section of the paper consists of four parts. The first part provides brief background on the Fellegi-Sunter model of record linkage. In the second part, we show how partial agreement relates to general likelihood ratios and associated information-theoretic decision rules. The empirical data base is described in the third part. The fourth part presents the specific string comparator and methods for modelling how it is used in adjusting matching weights between pure agreement and disagreement.

The third section contains empirical results based on files for which the truth of matches is known. The first subsection shows how specific weight adjustment curves are modelled for strings such as last name. The second subsection contains matching results that show the improvements due string comparators. The improvements are placed in the context of all techniques implemented in current production computer matching software that increase matching efficacy.

The fourth section provides discussion of the quality of the empirical data bases used in the analyses and the limitations of the existing string comparator/weight adjustment method.

The final section is a summary.

2. BACKGROUND

2.1. Fellegi-Sunter Model of Record Linkage

The Fellegi-Sunter Model uses a decision-theoretic approach establishing the validity of principles first used in practice by Newcombe (Newcombe et al. 1959, also 1988). To give an overview, we describe the model in terms of ordered pairs in a product space. The description closely follows Fellegi and Sunter (1969, pp. 1184-1187).

There are two populations **A** and **B** whose elements will be denoted by a and b. We assume that some elements are common to **A** and **B**. Consequently the set of ordered pairs

$$\mathbf{AXB} = \{(a,b): a\varepsilon\mathbf{A}, b\varepsilon\mathbf{B}\}$$

is the union of two disjoint sets of matches

$$M = \{(a,b): a=b, a\varepsilon\mathbf{A}, b\varepsilon\mathbf{B}\}$$

and nonmatches

$$U = \{(a,b): a\neq b, a\varepsilon\mathbf{A}, b\varepsilon\mathbf{B}\}.$$

The records corresponding to members of **A** and **B** are denoted by $\alpha(a)$ and $\beta(b)$, respectively. The comparison vector γ associated with the records is defined by:

$$\gamma[\alpha(a),\beta(b)] \equiv \{\gamma^1[\alpha(a),\beta(b)],\gamma^2[\alpha(a),\beta(b)],\cdots,\gamma^K[\alpha(a),\beta(b)]\}.$$

Each of the γ^i, $i = 1, \cdots, K$, represents a specific comparison. For instance, γ^1 could represent agreement/disagreement on sex. Also, γ^2 could represent the comparison that two surnames agree and take a specific value or that they disagree.

Where confusion does not arise, the function γ on **AXB** will be denoted by $\gamma(\alpha,\beta)$, $\gamma(a,b)$, or γ. The set of all possible realizations of γ is denoted by Γ.

The conditional probability of $\gamma(a,b)$ given $(a,b)\varepsilon M$ is

$$m(\gamma) \equiv P\{\gamma[\alpha(a),\beta(b)]|(a,b)\varepsilon M\} = \sum_{(a,b)\varepsilon M} P\{\gamma[\alpha(a),\beta(b)]\}\cdot P[(a,b)|M].$$

Similarly we denote the conditional probability of γ given $(a,b)\varepsilon U$ by $u(\gamma)$.

We observe a vector of information $\gamma(a,b)$ associated with pair (a,b) and wish to designate a pair as a link (denote the decision by A_1), a possible link (decision A_2), or a nonlink (decision A_3). A linkage rule L is defined as a mapping from Γ, the comparison space, onto a set of random decision functions $D = \{d(\gamma)\}$ where

$$d(\gamma) = \{P(A_1|\gamma),P(A_2|\gamma),P(A_3|\gamma)\};\ \gamma\varepsilon\Gamma$$

and

$$\sum_{i=1}^{3} P(A_i|\gamma) = 1.$$

There are two types of error associated with a linkage rule. A Type I error occurs if an unmatched comparison is erroneously linked. It has probability

$$P(A_1|U) = \sum_{\gamma\varepsilon\Gamma} u(\gamma)\cdot P(A_1|\gamma)$$

A Type II error occurs if a matched comparison is erroneously not linked. It has probability

$$P(A_3|M) = \sum_{\gamma\varepsilon\Gamma} m(\gamma)\cdot P(A_3|\gamma)$$

Fellegi and Sunter (1969) define a linkage rule L_0, with associated decisions A_1, A_2, and A_3, that is optimal in the following sense:

Theorem (Fellegi-Sunter 1969). Let L' be a linkage rule with associated decisions A_1', A_2', and A_3' such that it has the same error probabilities $P(A_3'|M) = P(A_3|M)$ and $P(A_1'|U) = P(A_1|U)$ as L_0. Then L_0 is optimal in that $P(A_2|U) \leq P(A_2'|U)$ and $P(A_2|M) \leq P(A_2'|M)$.

In other words, if L' is any competitor of L_0 having the same Type I and Type II error rates (which are both conditional probabilities), then the conditional probabilities (either on set U or M) of not making a decision under rule L' are always greater than under L_0.

2.2 General Partial Agreement

If the set of matches M were known, then we could model how partial agreement affects matching weights as follows:

1. Partition the closed interval [0,1] into a disjoint collection of subintervals $(k_i,k_{(i+1)}]$ for $i = 1, \cdots, N$. For convenience, chose $k_i = (i-1)/N$ and $k_{(i+1)} = i/N$ for $i = 1, \cdots, N$. If $i = 0$, then we include 0 in the interval $(k_i,k_{(i+1)}]$.
2. For each field j and for each $i = 1,\cdots N$, use the ratio

$$P(\Psi(\gamma^j(a,b))\varepsilon\ (k_i,k_{(i+1)}]\ |\ M)\ /\ P(\Psi(\gamma^j(a,b))\varepsilon\ (k_i,k_{(i+1)}]\ |\ U), \qquad (2.1)$$

as the value of the adjusted weight curve for interval $(k_i,k_{(i+1)}]$. Here Ψ is the string comparator, γ^j is a comparison of the jth field, (a,b) is an arbitrary pair, M is the set of matches, and U the set of nonmatches.
3. If, for a fixed field j, the curves (step functions) given by (2.1) are approximately the same for several data sets, then find a single piecewise linear curve as the approximation for each of them.

The piecewise linear curve will have the agreement weight as its highest value and the disagreement weight as its lowest value. We use the newly estimated curve on files for which the set of matches M is not known.

If the string comparators and associated methods for modelling ratios (2.1) are reasonably accurate, then the resultant decision rules are optimal in the sense of Fellegi and Sunter (1969, Theorem).

2.3. Empirical Data Bases

The emprical files are the 1988 Dress Rehearsal Census and Post Enumeration Survey (PES). The geographic regions consisted of portions of St Louis, MO, Columbia, MO, and rural Washington state.

Fields available for matching are first name, middle initial, last name, house number, street name, rural route number, postal box number, conglomerated address, telephone number, age, sex, marital status, relationship to head of household, and race.

Individuals in the PES are generally only computer matched with those individuals in the Census that are in the same block cluster. A block cluster may consist of a Census block or several blocks.

2.4. String Comparator Metrics

Jaro (see e.g., Winkler 1985, 1989, Winkler and Thibaudeau 1990) introduced a string comparator measure that gives values of partial agreement between two strings. The string comparator accounts for length of strings and partially accounts for the types of errors typically made in alphanumeric strings by human beings. It is used in adjusting exact agreement weights when two strings do not agree on a character-by-character basis.

Specifically, if $c > 0$, the Jaro string comparator is

$$\Phi = W_1 \cdot c/d + W_2 \cdot c/r + W_t \cdot (c-\tau)/c,$$

where

W_1 = weight associated with characters in the first of two files,
W_2 = weight associated with characters in the second of two files,
W_t = weight associated with transpositions,
d = length of string in first file,
r = length of string in second file,
τ = number of transpositions of characters, and
c = number of characters in common in pair of strings.

If $c = 0$, then $\Phi = 0$.

Two characters are considered in common only if they are no further apart than $(m/2 - 1)$ where $m = \max(d,r)$. Characters in common from two strings are assigned; remaining characters unassigned. Each string has the same number of assigned characters.

The number of transpositions is computed as follows: The first assigned character on one string is compared to the first assigned character on the other string. If the characters are not the same, half of a transposition has occurred. Then the second assigned character on one string is compared to the second assigned character on the other string, etc. The number of mismatched characters is divided by two to yield the number of transpositions.

If two strings agree on a character-by-character basis, then the Jaro string comparator Φ is set to $W_1+W_2+W_t$, which is the maximum value that Φ can assume. The minimum value that the Φ can assume is 0, which occurs when the two strings have no characters in common (subject to the above definition of common).

For present matching applications, W_1, W_2, and W_t are arbitrarily set to 1/3. The new string comparator metric basically modifies the basic string comparator according to whether the first few characters in the strings being compared agree. Specifically, for i = 1, 2, 3, 4,

$$\Phi_n = \Phi + i \cdot 0.1 \cdot (1 - \Phi)$$

if the first i characters agree.

If w_a and w_d are the estimated agreement and disagreement weights for a specific field, respectively, then the Jaro adjusted matching weight w_{am} used in the total weight calculation is given by

$$w_{am} = \begin{cases} w_a & \text{if } \Phi = 1, \text{ and} \\ \max\{w_a-(w_a-w_d)\cdot(1-\Phi)\cdot(9/2),\ w_d\} & \\ \qquad \text{if } 0 \le \Phi < 1. & \end{cases}$$

The constant 9/2 controls how quickly decreases in partial agreement values force the adjusted weight to the disagreement weight.

Instead of assuming that the same adjustment procedure works for different fields such as first name, last name, and house number, procedures for modelling the weight adjustment as a piecewise linear function were developed. The procedures necessitate having representative sets of pairs for which the truth of matches is known. The new adjusted weights w_{na} take the form

$$w_{na} = \begin{cases} w_a & \text{if } \Phi_n \ge b_1 \\ \max\{w_a-(w_a-w_d)\cdot(1-\Phi_n)\cdot(a_1),\ w_d\} & \\ \qquad \text{if } b_2 \le \Phi_n < b_1, & \\ \max\{w_a-(w_a-w_d)\cdot(1-\Phi_n)\cdot(a_2),\ w_d\} & \\ \qquad \text{if } \Phi_n < b_2. & \end{cases} \qquad (2.2)$$

on the specific type of string (such as first n applied. Generally, $a_1 < a_2$. The specific constants used are given in Table 4 of section 3.1.

Table 1 provides examples of string comparator values for pairs of last names and for pairs of first names. The abroms-abrams example with string comparator value .9333 in contrast to the lampley-campley with value .9048 shows that the string comparator gives a higher value to the pair that differs by a single character further from the first position. The martha-marhta example with value .9667 in contrast to the jonathon-jonathan example with value .9583 shows that transposition of two characters causes less of a downweighting than differing by one character.

Table 1. Values of String Comparator Φ_n

shackleford	shackelford	.9848
cunningham	cunnigham	.9833
campell	campbell	.9792
nichleson	nichulson	.9630
massey	massie	.9444
abroms	abrams	.9333
galloway	calloway	.9167
lampley	campley	.9048
dixon	dickson	.8533
frederick	fredric	.9815
michele	michelle	.9792
jesse	jessie	.9722
marhta	martha	.9667
jonathon	jonathan	.9583
julies	juluis	.9333
jeraldine	geraldine	.9246
yvette	yevett	.9111
tanya	tonya	.8933
dwayne	duane	.8578

3. RESULTS

The results of calculating the ratio (2.1) for various values of Φ_n for the first name are given in Table 2 and for the last name in Table 3. The weights in the last three columns correspond to disjoint intervals of the form $(k_1,k_2]$ where k_2 is given in column one. Within a table, we observe that each of the curves has roughly the same shape and the same starting and ending values.

Using the constants associated with first name from Table 4, the piecewise linear curve (2.2) for first name approximates each of the weighting curves (step functions) in Table 2. The constants and curves associated with other fields such as last name and house number are obtained in a similar manner.

Table 2. String Comparator Values and Weights First Name

	Weights		
Φ_n	StL	Col	Wash
0.62	-4.52	NA	-3.16
0.64	-3.13	-3.40	-3.06
0.66	-2.87	-1.91	-1.38
0.68	-2.44	-2.50	-2.39
0.70	-0.92	-1.53	-2.08
0.72	-1.02	-1.61	-0.43
0.74	0.14	-0.19	0.28
0.76	-0.22	-0.96	-0.17
0.78	0.88	0.27	2.05
0.80	0.83	0.63	0.84
0.82	2.10	2.14	2.09
0.84	2.25	1.72	2.42
0.86	2.31	2.93	3.78
0.88	3.05	2.53	3.41
0.90	3.46	3.19	2.73
0.92	3.77	3.09	3.58
0.94	4.27	3.56	3.31
0.96	4.42	5.03	4.58
0.98	5.48	5.04	4.34
1.00	4.62	4.58	4.86

Table 3. String Comparator Values and Weights Last Name

	Weights		
Φ_n	StL	Col	Wash
0.62	-5.35	-5.57	NA
0.64	-5.18	-4.28	NA
0.66	-5.21	NA	NA
0.68	-3.88	-4.38	-3.19
0.70	NA	NA	NA
0.72	-4.18	NA	-3.70
0.74	-3.26	-2.95	-2.23
0.76	-1.64	-3.88	-1.55
0.78	-1.53	-2.85	0.11
0.80	-1.49	-1.20	-0.82
0.82	-0.47	-0.65	0.42
0.84	0.02	0.45	-0.46
0.86	0.10	0.36	0.04
0.88	1.08	1.07	0.31
0.90	1.08	1.00	1.43
0.92	1.14	0.69	1.33
0.94	1.13	1.40	1.40
0.96	1.29	1.22	1.11
0.98	1.63	1.52	1.70
1.00	1.35	1.08	0.70

Table 4. Constants Used in Piece-Wise Linear Weight Adjustments

Field	a_1	a_2	b_1	b_2
first	1.5	3.0	.92	.75
last	3.0	4.5	.96	.88
house #	4.5	7.5	.98	.83

Weight adjustments are only performed for values of Φ_n greater than 0.60. Values below 0.60 are generally associated with pairs of strings associated with nonmatches in U.

The Jaro weight adjustment is used for the street field and any other fields that were not modelled. The street field weighting adjustment was modelled in a manner similar to the last name, first name, and house numbers. The Jaro weighting adjustment is conservative because it generally downweights more severely than the new curves and, thus, has less of a tendency to assign greater than the full disagreement weight to disagreeing strings.

3.2. Matching Comparison

A comparison of matching results is given in Tables 5, 6, and 7 for St Louis, Columbia, and Washington, respectively. To understand the tables, we need describe the types of matching procedures. The simplest procedure, crude, merely uses an ad hoc guess for matching parameters and does not use string comparators.

The next, param, does not use string comparators but does estimate the probabilities m(γ) and u(γ). Such probabilities are often estimated through an iterative procedure that involves manual review of matching results and successive reuse of the reestimated parameters. The third type, param2, uses the same probabilities as param and the basic string comparators.

The fourth type, em, uses an EM-Algorithm for estimating matching parameters (see e.g., Winkler 1988, Thibaudeau 1990) and uses the basic Jaro string comparator. The fifth type, em2, uses the EM-derived weights and the new string comparator and new weight adjustments. The final type, freq, replaces simple agree/disagree weights for first name and last name with frequency-based weights (see e.g., Winkler 1989) and also makes adjustments for joint dependencies of agreements on first name, sex, and age.

In each table, the number of matches is determined by a false match rate of 0.002. The crude and param types are allowed to rise slightly above the 0.002 level because they generally have higher error levels.

By examining the tables we observe that a dramatic improvement in matches can occur when string comparators are first used (from param to param2). The basic reason is that disagreements (on a character-by-character basis) are replaced by partial agreements. Improvements due to the new string comparators and weighting adjustments (from em to em2) are quite minor.

Table 5. Computer Categories Various Procedures 10291 True Matches 12072 Records, St Louis Pairs Agree on Cluster and First Character Surname 1/

	-computer designation- match		clerical	
truth->	match	non-match	match	non-match
crude	310/	1	9344/	794
param	7899/	16	1863/	198
param2	9276/	23	545/	191
em	9587/	23	271/	192
em2	9639/	24	215/	189
freq	9801/	24	52/	94

1/ Approximately 400 true matches disagree on first character of surname and are not eligible for inclusion in the table.

Table 6. Computer Categories Various Procedures 6984 True Matches 7649 Records, Columbia Pairs Agree on Cluster and First Character Surname 1/

	computer designation match		clerical	
truth->	match	non-match	match	non-match
crude	2429/	7	4327/	119
param	6449/	22	327/	92
param2	6655/	13	135/	35
em	6719/	13	78/	22
em2	6762/	13	37/	20
freq	6792/	11	6/	9

1/ Approximately 180 true matches disagree on first character of surname and are not eligible for inclusion in the table.

Table 7. Computer Categories Various Procedures 1950 True Matches 2214 Records, Washington Pairs Agree on Cluster and First Character Surname 1/

	computer designation match		clerical	
truth->	match	non-match	match	non-match
crude	1307/	3	564/	98
param	1250/	5	614/	88
param2	1765/	4	134/	41
em	1749/	4	149/	29
em2	1795/	3	107/	29
freq	1892/	4	7/	9

1/ Approximately 40 true matches disagree on first character of surname and are not eligible for inclusion in the table.

4. DISCUSSION

4.1. Quality of Empirical Data Bases

Because of the relatively large number of identifying fields for matching, all results in section 3.2 are relatively better than might be expected in general matching applications. Also, having two key fields such as first name and last name with typographical variation sufficiently severe for assignment of full disagreement weight to a true match is very rare (below 0.1 percent). The data are, however, representative of the type of data that will be encountered during the 1990 Post Enumeration Survey.

The data are suitable for evaluating matching procedures because essentially all matches were found and correctly identified. The identification is with codes specifying to which record a record is matched. All basic identifying information was carefully checked and rechecked. In particular, no matches were found among the set of code-identified nonmatches using a variety of procedures.

4.2. General String Comparator Metrics

For matching applications of files having significantly different characteristics (i.e., matching fields) from those of the files of this paper, string comparator weighting adjustments may have to be remodelled.

In all matching situations, it seems likely that modelling partial agreement should improve matching efficacy because the proportions of exact agreement on key matching fields can be quite low. For the files of this paper, the proportions of true matches agreeing on a character-by-character basis (Φ_n=1.0) are approximately 76 percent for first name and approximately 86 percent for last name (Table 8).

Table 8. Proportional Agreement by String Comparator Values Key Fields by Geography

	StL	Col	Wash
First			
Φ_n=1.0	0.75	0.82	0.75
$\Phi_n \geq 0.6$	0.93	0.94	0.93
Last			
Φ_n=1.0	0.85	0.88	0.86
$\Phi_n \geq 0.6$	0.95	0.96	0.96

5. SUMMARY

This paper contains a new string comparator that partially accounts for minor typographical variation when two strings are compared. The theoretical decision rules of Fellegi and Sunter (1969) are still valid when general weighting adjustments accounting for partial agreement are performed.

*This paper reports general results of research by Census Bureau staff. The views expressed are attributable to the author and do not necessarily reflect those of the Census Bureau.

REFERENCES

Fellegi, I. P., and Sunter, A. B. (1969), "A Theory for Record Linkage," JASA 40 1183-1210.

Thibaudeau, Y. (1990), "Improving the Performance of Computer Matching Algorithms through Better Choices of Parameters," paper presented at the Annual ASA Meeting in Anaheim, Califoria.

Winkler, W. E. (1985), "Preprocessing of Lists and String Comparison," in Record Linkage Techniques -1985, edited by W. Alvey and B. Kilss, U.S. Internal Revenue Service, Publication 1299 (2-86), 181-187.

Winkler, W. E. (1988), "Using the EM Algorithm for Weight Computation in the Fellegi-Sunter Model of Record Linkage," ASA 1988 Proceedings of the Section on Survey Research Methods, 667-671.

Winkler, W. E. (1989), "Frequency-based Matching in the Fellegi-Sunter Model of Record Linkage," ASA 1989 Proc. of the Section on Survey Research Methods, 788-793.

Winkler, W. E. and Thibaudeau, Y. (1990), "An Application of the Fellegi-Sunter Model of Record Linkage to the 1990 U.S. Census," technical report.