MEDINFO 2004
M. Fieschi et al. (Eds)
Amsterdam: IOS Press

Real World Performance of Approximate String Comparators for use in Patient Matching

Shaun J. Grannis, J. Marc Overhage, Clement McDonald

The Regenstrief Institute for Health Care and Indiana School of Medicine, Indianapolis, IN, USA

Abstract

Medical record linkage is becoming increasingly important as clinical data is distributed across independent sources. To improve linkage accuracy we studied different name comparison methods that establish agreement or disagreement between corresponding names. In addition to exact raw name matching and exact phonetic name matching, we tested three approximate string comparators. The approximate comparators included the modified Jaro-Winkler method, the longest common substring, and the Levenshtein edit distance. We also calculated the combined root-mean square of all three. We tested each name comparison method using a deterministic record linkage algorithm. Results were consistent across both hospitals. At a threshold comparator score of 0.8, the Jaro-Winkler comparator achieved the highest linkage sensitivities of 97.4% and 97.7%. The combined root-mean square method achieved sensitivities higher than the Levenshtein edit distance or longest common substring while sustaining high linkage specificity. Approximate string comparators increase deterministic linkage sensitivity by up to 10% compared to exact match comparisons and represent an accurate method of linking to vital statistics data.

Keywords:

Medical Record Linkage, Patient Matching, Knowledge Management

Introduction

Health care information is increasingly distributed across many independent databases and systems, within and among institutions as separate collections with varying types of identifying information. This is true for data collected within an institution where there may be multiple identifiers, or for data collected about the same patient at different health care institutions. While integrating patient data across such disparate sources is challenging, answers to important health research, management, and policy questions can be obtained by linking clinical information from independent systems[1-3]

Although many variations exist, record linkage techniques can be broadly divided into two categories: deterministic (heuristic) and probabilistic. Deterministic algorithms employ a set of rules based on exact agreement or disagreement results between corresponding fields in record pairs. Probabilistic methods commonly use likelihood scores calculated from rates of identifier agreement and disagreement among fields from potentially linked and non-linked records.[4]

We previously examined the accuracy of both deterministic and probabilistic record linkage methods, and found that Social Security Number (SSN) was insufficient as a single parameter linking patients to a vital statistics database.[5, 6] The methods we used imposed exact-agreement criteria on all fields. That is, two corresponding fields within a record are said to agree only if all characters match; otherwise the fields are considered to disagree. We recognized that names and other data can include variations (e.g. as changing last name) and recording errors result in disagreement when a human reader might recognize them or the equivalent (e.g. “Rob” and “Robert”). To overcome such variations and recording errors in our previous work, we implemented the NYSIIS phonetic compression algorithm [7, 8] to preprocess name strings.

Approximate string comparators compute a measure of similarity between two strings. String comparators have been described in the context of record-linkage[9, 10]; however, there is a paucity of literature describing the actual performance of such comparators in patient record linkage, and no data comparing linkage accuracy using phonetic compression versus approximate string comparators. In this paper we will compare the performance of a deterministic linkage method using exact-match, phonetic compression, and approximate string comparators.

Materials and Methods

Record Sources

To study how string comparators perform in record linkage, we linked hospital registry data to a separate data source containing an identifiable subset of the registry population. We used the Social Security death master file (SSDMF) as that data source. The SSDMF is a publicly available database containing demographic data for over 65 million deceased individuals. It includes fields for SSN, name, date of birth, date of death, state or country of residence, ZIP code of last residence, and ZIP code of lump-sum payment. Matching patients to the SSDMF has general relevance to all medical databases and registries because a match to the SSDMF provides an excellent indicator of vital status and mortality is an important outcome variable for many research questions.

The data used in this study was derived from two hospitals in central Indiana. Hospital A is a public inner-city hospital system. Hospital B is a private urban hospital system that invested in extensive patient registry clean-up in 1999.

Manually Reviewed Reference Set

We used two manually reviewed reference sets of record pairs to gauge the accuracy of string comparators and patient matches. To generate candidate record pairs for each hospitals’ reference

sets, we used SSN as the single variable linking patient records to the SSDMF. This process of generating candidate record pairs is commonly referred to as *blocking*. The purpose of blocking is to reduce the total number of records to process by placing pairs in smaller bins, or blocks; it is analogous to sorting trouser socks by color before pairing them together. We randomly selected 6,000 samples from the candidate record pairs for each hospital and manually reviewed the two data sets, labeling individual record pairs as true or false links.

Phonetic Compression

Phonetic encoding algorithms are used to minimize variations in spelling of what are effectively the same names[11]. There are several well-known phonetic compression algorithms; examples include Soundex[12], Metaphone, and the New York State Identification and Intelligence System algorithm (NYSIIS)[7]. The NYSIIS algorithm has 11 basic rules that replace common pronunciation variations with standardized characters, remove repeated characters, and replace all vowels with the letter 'A'. Because it retains information on the sequence of vowels, NYSIIS has higher discriminating power than Soundex[7]. The NYSIIS transformations of 'TAMMIE' and 'TAMMY' are both 'TANY'.

String Comparators

We studied three string comparators. The ***modified Jaro-Winkler comparator (JWC)*** was developed by the U.S. Census Bureau[9]. The basic algorithm[13] computes the number of common characters in two strings and finds the number of transpositions. To be labeled as *common*, corresponding characters must be located within half the length of the shorter string. A *transposition* occurs when the order of corresponding common characters is reversed. The method assigns partial scores to characters that disagree but are similar, either due to typographical ('B' versus 'V') or scanning errors ('7' versus 'T'). Further, greater value is given to agreement within the first four characters of a string, based on research showing that fewer errors occur at the beginning of a string and the errors increase monotonically toward the end of a string[14]. Finally, increased weight is given to strings longer than six characters when more than half the characters beyond the first four agree. The comparator score for 'TAMMY SHACKELFORD' and 'TAMMIE SHACKLEFORD' is 0.9442.

The ***longest common substring (LCS)*** algorithm generates a nearness metric by iteratively locating and deleting the longest common substring between two strings.[15] The substrings must meet a minimum length requirement, which we set to three for our analysis. The nearness metric is calculated by dividing the total length of the shared substrings by the length of the shorter of the two strings being compared. For example, the LCS score for the names 'TAMMY SHACKELFORD' and 'TAMMIE SHACKLEFORD' is calculated as follows: The total length of the common substrings is [5 (SHACK) + 4 (TAMM) + 4 (FORD)] = 13. The length of the shorter name string (ignoring white space) is 16, therefore the LCS score is (13÷16) = 0.8125

The ***Levenshtein edit distance (LEV)***[16] determines the smallest number of insertions, deletions, and substitutions required to change one **string** into another.[REF NIST] From this we construct a metric ranging from 0 to 1.0 using the formula: metric = 1 – [LEV(name1,name2) ÷ MAXLEN (NAME1, NAME2)] A value of 1 represents an exact match, while zero indicates little similarity. To illustrate, the Levenshtein edit distance for the names 'TAMMY SHACKELFORD' and 'TAMMIE SHACKLEFORD' is 4. Changing the former name into the latter, one substitutes 'I' for 'Y', inserts an 'E' after 'I', and reverses the order of the 'E' and 'L' (two substitutions). The length of the longer name is 17 (ignoring white space); thus the score is (1 – (4 ÷ 17)) = 0.7647

In addition to the above algorithms, we also calculated the combined ***root mean square (RMS)*** of all three comparator scores. We did this to examine whether a combined metric would improve linkage accuracy over any single comparator.

Link Criteria

To determine the incremental benefit of adding string comparators, we compared our previous results using exact name match comparisons to new results using one of the three string comparators or the RMS string comparator. We used six sets of linkage criteria, which varied only by the name comparison methods. Table 1 describes the complete list of criteria used to define true matches. The string comparator algorithms used the full concatenated first and last names.

Table 1: Criteria used to define true matches. (FN= first name; NYS=NYSIIS transformed first name; G=gender; MB,DB,YB=month, day and year of birth.)

Deterministic Matching Criteria	
Exact	Exact agreement on SSN, first name, G and at least one of the following: MB, DB, or YB
NYSIIS	Exact agreement on SSN, NYSIIS first name, G and at least one of the following: MB, DB, or YB
LEV	LEV score > 0.8, exact agreement on SSN, gender and at least one of the following: MB, DB, or YB
LCS	LCS score > 0.8, exact agreement on SSN, gender and at least one of the following: MB, DB, or YB
JWC	JWC score > 0.8, exact agreement on SSN, gender and at least one of the following: MB, DB, or YB
RMS	RMS score > 0.8, exact agreement on SSN, gender and at least one of the following: MB, DB, or YB

Computational Costs

To evaluate the computational cost of each algorithm, we randomly selected 50,000 name pairs from candidate record pairs and measured the time required to process the data for each algorithm. We used the average elapsed time of five trials for each of the algorithms. All measurements and analyses were performed on a dual processor AMD Athlon MP 1900 system with 4GB RAM running Red Hat Linux 7.2. The algorithms were im-

plemented in C using a Perl wrapper script to call the comparator functions.

Results

Using SSN as the blocking variable we generated 65,365 candidate record pairs linking hospital A to the SSDMF. From the 6,000 randomly selected record pairs, manual review found 5,298 (88.3%) true-links and 702 (11.7%) non-links. We generated 169,315 candidate record pairs linking hospital B to the SSDMF using SSN as the blocking variable. From the 6,000 sample record pairs, manual review found 5,655 (94.3%) as true-links and 345 (5.7%) non-links.

Table 2 shows the results using different match criteria for hospitals A and B. Similar trends are noted for both institutions. Criteria using first name exact agreement demonstrated the lowest sensitivity. The NYSIIS method improved linkage sensitivity for criteria that used exact agreement; however, string comparators more substantially improve sensitivity. The JWC had the lowest specificity while demonstrating the highest sensitivity. The RMS, a combination of all three comparators, improved overall sensitivity while sustaining high specificity.

To measure overall linkage accuracy when using string comparator functions, we calculated the area under the ROC curve (AUC) for each approximate comparator. Figure 1 shows the results for hospital A and B. We note a decrease in overall accuracy with the JWC method. This is explained by the decreased specificity reflected in Table 2

Figure 2 shows the computational costs measured in seconds for each of the string comparison methods using 50,000 randomly selected record pairs. For reference purposes, exact match comparisons took a total of 4 seconds to process. It is of note that the NYSIIS algorithm demonstrated the greatest cost. The NYSIIS algorithm invokes multiple rules with sub-iterations requiring robust pattern matching, and each name is transformed independently of one another. Only after two completed transformations can an exact-match comparison can be made. Alternatively, approximate string comparators such as JWC, LCS, and LEV use both strings within the same process, calculating the metric in a single pass. The RMS calculation was the most computationally intensive of the string comparators because it requires all three individual approximate comparator values.

Discussion

When choosing a linkage method, one must consider the use-case and the cost of false positive and false negatives. If the cost of false positives is high, one may choose the RMS method, which sustains high specificity while demonstrating gains in sensitivity over exact-match criteria. Further, if the cost of false negatives is high, it may be more appropriate to use the JWC criteria, which achieved the highest sensitivity among the methods listed.

At a threshold score of 0.8 the Levenshtein edit distance did not substantially increase linkage accuracy compared with the NYSIIS algorithm. However, it demonstrated the highest specificity among the approximate comparators and helped to mitigate the lower specificity of the Jaro-Winkler comparator when using the RMS metric. Further work will investigate whether simple modifications to the LCS or JWC methods can improve their specificity. As currently defined and implemented, name pairs such as 'JON SMITH' and 'JON SMITHERS' receive 1.0 scores for both the JWC and LCS methods.

Although the NYSIIS method requires more computation relative to the other string comparison methods, it is often used for blocking purposes in record linkage, rather than directly as a string comparator. Consequently, because blocking reduces the total number of record pairs to process, using the NYSIIS algorithm is intended to help to *reduce* overall processing time.

At a threshold score of 0.8 the Levenshtein edit distance did not substantially increase linkage accuracy compared with the NYSIIS algorithm. However, it demonstrated the highest specificity among the approximate comparators and helped to mitigate the lower specificity of the Jaro-Winkler comparator when using the combined RMS metric. Further work will investigate whether simple modifications to the LCS or JWC methods can improve their specificity. As currently defined and implemented, name pairs such as 'JON SMITH' and 'JON SMITHERS' receive 1.0 scores from both JWC and LCS.

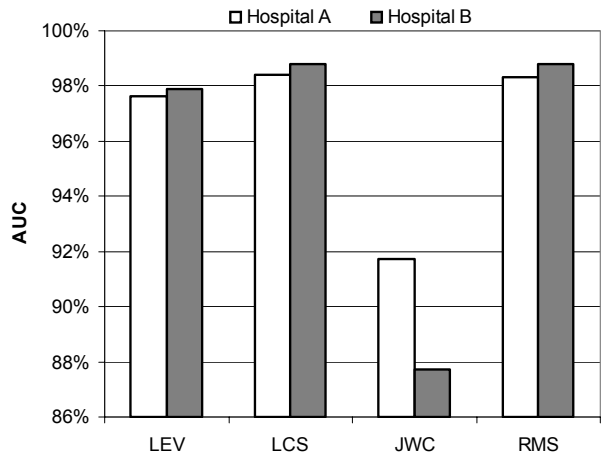

Figure 1 - String comparator accuracy as measured by the area under the ROC curve (AUC) for hospitals A and B

This study is limited by the fact that we blocked record-pairs using SSN alone. Records that agree at the outset on SSN will have a high proportion of true-links. Record pairs formed with additional blocking schemes may produce different results. Also, we used a threshold score of 0.8 to determine agreement status; results will vary with different thresholds.

These results reflect linkage performance for two hospitals in central Indiana. Results may vary at other institutions, or when linking to a source other than the SSDMF. That being the case, these data sets are an informative spectrum of patient registries: one hospital recently underwent registry clean-up, while the other has not. The registry clean-up efforts are reflected in the greater accuracy noted in Hospital B**.**

Table 2: Linkage results using different string comparison methods. (TP-true link, FP-false link, FN-false non-link, TN-true non-link, FNR-false negative rate; FPR-false positive rate)

Criteria	TP	FP	FN	TN	Sens.	Spec.	FNR	FPR
Hospital A (true links = 5298, non-links = 702)								
Exact	4499	2	799	700	.8492	.9972	.1508	.0028
NYSIIS	4638	2	660	700	.8754	.9972	.1246	.0028
LEV	4642	1	656	701	.8762	.9986	.1238	.0014
LCS	4986	2	312	700	.9411	.9972	.0589	.0028
JWC	5158	7	140	695	.9736	.9900	.0264	.0100
RMS	5020	1	278	701	.9475	.9986	.0525	.0014
Hospital B (true links = 5655, non-links = 345)								
Exact	5081	1	574	344	.8985	.9986	.1015	.0014
NYSIIS	5143	2	512	343	.9095	.9972	.0905	.0028
LEV	5133	1	522	344	.9077	.9986	.0923	.0014
LCS	5459	1	196	344	.9653	.9986	.0347	.0014
JWC	5523	5	132	340	.9767	.9929	.0233	.0071
RMS	5473	1	182	344	.9678	.9986	.0322	.0014

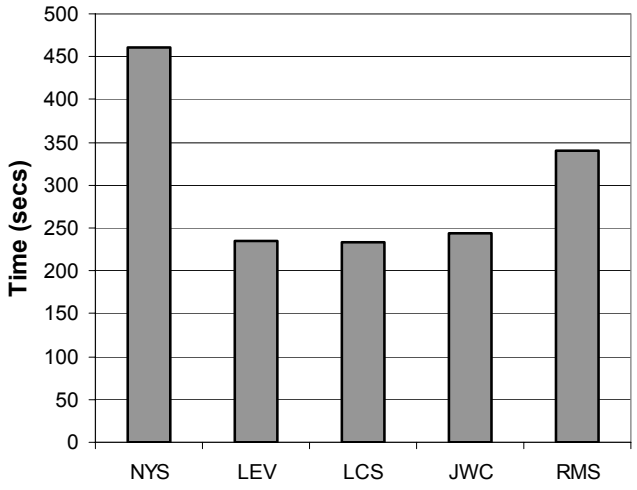

Figure 2 - Total average processing time for 50,000 record pairs

Conclusion

As clinical data continues to be distributed among health care organizations and their disparate clinical systems, the need for accurate methods to link records across these systems is becoming increasingly important. Approximate string comparators contribute to improved linkage methodologies. Not surprisingly, the individual approximate comparators had strengths and weaknesses with some being highly specific and others highly sensitive. The RMS method exhibited the best performance, but at an increased computational cost. One should choose a linkage method based on a tolerance for false positives or false negatives as needed. Regardless, the overall accuracy of deterministic patient linkage is improved by using approximate string comparators.

Acknowledgements

This research was performed at the Regenstrief Institute in Indianapolis, Indiana and was funded by the National Library of Medicine grant T15 LM-7117-05. Thanks to Dr. Paul Dexter, Regenstrief Institute, Indiana University School of Medicine and Sean Thomas MD, Queen's Medical Center, John A. Burns School of Medicine, University of Hawai'i and for reviewing this manuscript.

References

[1] Potosky A, Riley G, J L. Potential for Cancer Related Health Services Research Using a Linked Medicare-Tumor Registry Database. *Medical Care* 1993;31(8):732-748.

[2] Liu S. Development of record linkage of hospital discharge data for the study of neonatal readmission. *Chronic Dis Can* 1999;20(2):77-81.

[3] Whalen D, Pepitone A, Graver L, Busch J. *Linking Client Records from Substance Abuse, Mental Health and Medicaid State Agencies*. Rockville: Center for Substance Abuse Treatment and Mental Health Sevices Administration; 2000 July, 2000. Report No.: SMA-01-3500.

[4] Felligi I, Sunter A. A Theory for Record Linkage. *J Am Stat Assn* 1969;328(64):1183-1210.

[5] Grannis SJ, Overhage JM, McDonald CJ. Analysis of Identifier Performance Using a Deterministic Linkage Algorithm. *Proc AMIA Symp* 2002:305-9.

[6] Grannis SJ, Overhage JM, McDonald CJ. Analysis of a Probabilistic Record Linkage Technique without Human Review. In: *Proceedings of American Medical Informat-*

ics Association Fall Symposium; 2003; Washington, D.C.; 2003.

[7] Lynch B, Arends W. *Selection of a surname encoding procedure for the Statistical Reporting Service record linkage system*. Washington, D.C.: U.S. Department of Agriculture; 1977.

[8] Atack J, Bateman F, Gregson M. Match Maker, Match Maker, Make Me a Match: A General Personal Computer-Based Matching Program for Historical Research. *Historical Methods* 1992;25(2):53-65.

[9] Porter E, Winkler W. Approximate string comparison and it's effect on an advanced record linkage system. In: Record Linkage Techniques: *Proceedings of an International Workshop and Exposition;* 1997; Arlington, VA: National Academy Press; 1997. p. 190-199.

[10]Christen P, Churches T. Febrl - Freely extensible biomedical record linkage. 2003; URL: http://cs.anu.edu.au/~Peter.Christen/febrl-0.2/febrldoc/manual.html (accessed September 3, 2003).

[11]Newcombe H. *Handbook of Record Linkage, Methods for Health and Statistical Studies, Administration, and Business*: Oxford University Press; 1988.

[12]Knuth D. *The Art of Computer Programming, Volume 3/ Sorting and Searching*. 2nd ed: Addison-Wesley Publishing Company; 1998.

[13]Jaro M. Advances in record linkage methodology as applied to matching the 1985 census of Tampa, Florida. In: Record Linkage Techniques: *Proceedings of an International Workshop and Exposition*; 1997; Arlington, VA: National Academy Press; 1997. p. 351-357.

[14]Pollock J, Zamora A. Automatic Spelling Correction in Scientific and Scholarly Text. *ACM Computing Surveys* 1987;27(4):358-368.

[15]Sidelli R, Friedman C. Validating patient names in an integrated clinical information system. In: *Symposium on Computer Applications in Medical Care*; 1991; Washington, D.C.; 1991. p. 588-592.

[16]Levenshtein VI. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady* 1966;10(8):707-710.

Address for Correspondence

Shaun Grannis
1050 Wishard Blvd. RG5
Indianapolis, IN 46202
sgrannis@regenstrief.org