Name Standardization for Genealogical Record Linkage

D. Randall Wilson
Family & Church History Department
The Church of Jesus Christ of Latter-day Saints
wilsonr@ldschurch.org

1. Introduction

A common part of genealogical research is finding multiple records that refer to the same person and either linking them together or gathering their information into a single place. Finding records that refer to the same person—whether in printed indexes or online databases—usually requires using the person's name. However, names can vary in different sources for a variety of reasons, including

- nicknames (Margaret/Peggy),
- transcription or typographical errors (James/Jarnes/Jamew),
- abbreviations (William/Wm./W.),
- translation/immigration name changes (Schmidt/Smith),
- same-sounding spelling variations (Barns/Barnes),
- minor changes to names over time (Speak/Speake/Speaks/Speakes)

In addition to the above changes that can happen to specific *name pieces*, the full name can vary due to inclusion of different numbers of name pieces ("Mary" vs. "Mary Turner" vs. "Mary Eliza Turner"), name order differences ("Mary Eliza Turner" vs. "Eliza Mary Turner" vs. "Turner, Mary Eliza"), and the inclusion of titles ("Henry II, King of England", "Rev. John H. Tunnell, Jr."), or other additions such as married names ("Mary Turner" vs. "Mrs. Mary Smith") or even Scandinavian farm names.

Various approaches have been proposed to *standardize* names so that variations of the same person's names can be brought together. These approaches include manually built name catalogs, algorithmically enhanced name catalogs, name encoding algorithms (such as Soundex, Nysiis, and Double Metaphone), name comparison algorithms (such as Edit Distance, Jaro-Winkler distance, etc.), as well as simple normalization techniques (lower case, collapse white space, remove periods, etc.). Each approach addresses a different set of the above types of name variations, and each casts a different size and shape of a "net" over the name space. The result is that each brings together different sets of names that commonly belong to the same person, and each tends to also bring together other names that do not.

The goal of standardization is to bring together nearly all of the different names that the same person could have while bringing together as few other names as possible. This paper presents empirical results of testing a variety of name standardization techniques on a corpus of labeled data.

Section 2 describes the different name standardization techniques that were tested. Section 3 describes the corpus of labeled data that was used to gather empirical results. Section 4 presents the results in terms of *recall* (the percent of known matched pairs of records that were brought together using each technique) and *cost* (the average number of individuals in the database brought together by each technique). Section 5 presents conclusions.

2. Name standardization techniques

Several name standardization techniques were used in this study, each of which is briefly described below, including several *name encoding algorithms* and several *catalog-based encodings.* In all cases it is assumed that the name is *tokenized* (i.e., broken up into name pieces) and *normalized* (i.e., converted to lower case, punctuation removed).

2.1. Name encoding algorithms.

Name encoding algorithms take a normalized string and apply some algorithm to map it to another string that will likely match on variations of that name.

Soundex. The *American Soundex System* is a slight modification of the original *Russell Soundex Code* (Russell, 1918). It uses the first letter of a name followed by three digits. The digits are found by dropping vowels (as well as "h", "w" and "y"), and mapping the remaining letters using the list shown on the right.

The only other two rules are (1) if two or more consecutive letters have the same code, only the first is used, and (2) if you run out of letters than use 0. For example, the Soundex code for both "Miller" and "Mueller" is "M460".

Soundex Mappings

1	b,f,p,v
2	c,g,j,k,q,s,x,z
3	d, t
4	l
5	m, n
6	r

NYSIIS. The NYSIIS (or the *New York State Identification and Intelligence System*) phonetic encoder (Taft, 1970) was developed after rigorous empirical studies to improve upon the results of Soundex. It creates a string with up to 10 characters, using the following rules:

1) Translate first characters of name:
   MAC => MCC, KN => NN, K => C, PH => FF, PF => FF, SCH => SSS
2) Translate last characters of name:
   EE => Y; IE => Y; DT,RT,RD,NT,ND => D
3) First character of key = first character of name.
4) Translate remaining characters by following rules, incrementing by one character each time:
   a. EV => AF else A,E,I,O,U => A
   b. Q => G, Z => S, M => N
   c. KN => N, else K => C
   d. SCH => SSS, PH => FF
   e. H => If previous or next is non-vowel, previous
   f. W => If previous is vowel, previous
   Add current to key if current ≠ last key character
5) If last character is S, remove it
6) If last characters are AY, replace with Y
7) If last character is A, remove it

Metaphone. Phillips Lawrence's *Metaphone* algorithm (Philips, 1990) reduces the alphabet to 16 consonant sounds (B, X, S, K, J, T, F, H, L, M, N, P, R, 0 [i.e., "th"], W, Y). It drops vowels unless they are the first letter, and maps each other letter to one of these 16 consonant sounds using a list of rules. So, for example, the Metaphone code for "Bender" is "BNTR". It is a bit more complex than NYSIIS, so its rules are omitted here.

Double Metaphone. The *Double Metaphone* algorithm (Philips, 2000) revises the Metaphone algorithm to produce both the "most likely" pronunciation and an optional alternative pronunciation (thus the "double"). For example, the Double Metaphone codes for "Schneider" are "XNTR" and "SNTR", while "Thomas" is encoded as "TMS" and has no alternate encoding.

2.2. Name catalogs

The Family & Church History Department of The Church of Jesus Christ of Latter-Day Saints has done work in developing computerized name standardization catalogs beginning in about 1970. Several varieties were included in this study, which are described below.

ODM catalog. The first catalog is the one used by the *Ordinance Data Management* system (as well as by the *TempleReady* product). It consists of a separate catalog for each of 20 regions in the world (North America, Central America, British Isles, Norway, etc.). Each catalog maps a group of names to a single standard spelling (e.g., Margaret, Maggie, and Peggy might all map to "MARGARET"). The same spelling can occasionally map to a different standard spelling in different regions (e.g., "Hans" might map to "Johan" in Germany, but to "Hans" in North America), but usually ends up the same (e.g., "Jonathan" maps to "JOHN" in most regions that have that name).

To determine which catalog to use, the regions in which each of an individual's events occur—as well as the events of their immediate relatives—are used. If people (or their immediate relatives) have events in more than one region, then the person's name is standardized once for each region.

As new names enter the system, they are usually added as their own standard until experts can review them and possibly add them to existing groups.

Universal catalog. In recent years, an attempt was made to use a single "universal" catalog that was not region-specific. Buckets of names were created, and each original (normalized) spelling mapped to one or more such buckets. So, for example, "Rebecca" appears in buckets 169573 and 283838 in the catalog. Using a universal catalog avoids problems associated with trying to compare names when one or the other person has no events on which to base a region, and also avoids the problem of choosing which spelling from a group of names is the "standard" one.

Culture-based catalog. A third kind of name catalog explored by the Family & Church History Department recently takes the original *ODM* catalog, and uses unique IDs in each culture, instead of standard spellings. It also uses only a single region (i.e., "culture") for each individual instead of using multiple standards for all of the regions that a person or their relatives appear in. We tested three variations of this catalog—one in which only the person's own events were used to determine culture (*Culture_person*), one in which the person and their close relatives were used to determine culture (*Culture_relatives*), and one in which the "default" culture was used for everyone (*Culture_default*). From a list of the person's events and those of their relatives, the first one with a region specified was used to determine the name catalog to use.

2.3. Name comparison functions. A number of string-comparison functions exist that can be used to compare names, such as *Edit Distance*, *Jaro-Winkler*, and others. These are useful for handling minor spelling variations as well as small typographical errors. Typographical errors are not handled well by catalogs due to the nearly limitless ways in which a name can be misspelled. On the other hand, name comparison functions are not well suited for indexing, so they were largely excluded from this study, which depended on begin able to create a searchable index for each type of name standard.

Edit Distance Catalog. One exception, however, is a catalog built by Steven West, which creates, for each name piece, a list of all of the other name pieces in the database that are within an edit distance of 0.95. A similar approach could be used for other name-comparison algorithms such as Jaro-Winkler. Such a catalog is much simpler to create than a manual catalog, though it would not handle typographical errors that did not already exist in the database, nor is it likely to handle nicknames.

3. Labeled data

In order to compare the various name standardization methods listed in Section 2, we used a database of 178,880 individuals, which included 25,000 pairs of records that had been identified by genealogical experts as referring to the same person. Some of these pairs of records had identical names, in which case standardization didn't help, but many of them (over half) had at least some variation.

4. Empirical results

This section presents empirical results of applying each of the standardization methods in Section 2 to the labeled data mentioned in Section 3. It shows the benefit (in terms of bringing matching records together) and cost (in terms of having to deal with more matching records that are not necessarily matches) of each method on given names, surnames, and both taken together. It also shows the benefit and cost of several combinations of methods.

4.1. Definition of recall and cost

The success of a standardization method is measured in terms of *recall* and *cost* (or *number of "hits"*). *Recall* is the percent of known matching pairs of records that are brought together by the method. For example, in our data, about 85% of matching pairs have at least one original (normalized) surname piece in common, while about 89% of matching pairs have at least one Soundex surname piece in common. Thus, issuing a query against a database using the normalized surname or the Soundex code for the surname of one record "finds" the other record 85% and 89% of the time, respectively, in this data set.

Cost can be measured (a) as the percent of the database that is covered on average by a particular standardization method (the *coverage*), (b) as the *number of hits* that is returned on average when querying using that standard, or (c) by the *fraction of the database* that is returned by such a query. (Given a database of size *N*, and number of hits *h*, we can relate these terms by *coverage = h / N * 100%*, and *fraction = N / h*.)

Continuing the above example, 0.03% of the database (or about 1/3000th of it) is covered on average by a person's (original, normalized) surname, which is 61 hits per query on average on the database of 178,880 individuals. By using Soundex code, 0.15% of the database (or about 1/600th of it) is covered on average, which is 261 hits per query on average.

In this example, we see a trade-off between recall and cost. Using the original surname has 85% recall at a cost of 61 hits. Using Soundex on the surname yields better recall—89%—but a worse (higher) cost of 261 hits. The goal of standardization is to achieve near 100% recall while keeping the cost as low as possible.

4.2. Name piece types

Most of the data in our test database had its names split up into *given name* pieces (e.g., John, Fred, Svetlana) and *surname* pieces (e.g., Smith, Jones, Larsen). Our experiments looked at given names separately, surnames separately, and both together.

4.4. Given name piece results

Table 1 shows a summary of the results obtained on just the *given name* pieces for each individual. In all of these tables, "Orig" means the original name piece, normalized by converting to lower case and removing punctuation. Also, "Edit" means the Edit Distance-based catalog.

The various standardization techniques are sorted in Table 1 from highest recall to lowest. The results in **bold type** are those that are the *highest recall for the given number of hits.* For example, "ODM+Orig" gets 98.62% recall at a cost of 3771 hits. Soundex has lower recall (98.31%) and also has a higher cost (5761), so it is worse in both ways. "Culture_relative+Orig", on the other hand, has lower recall (97.81%), but also has a lower number of hits, so it, too, is a "winner."

Figure 1 shows this relationship graphically. The techniques that yield the highest recall for their number of hits are shown with solid diamonds, and are connected by a cost-recall line. The hollow diamonds, which are all below (or to the right of) the cost-recall curve, are the results that are worse in both recall and cost compared to one of the "winners".

Given Name Fields	Recall	AvgHits	% of Db
Universal + Orig	**99.08**	**9689**	**5.42%**
Universal	98.67	9689	5.42%
ODM + Orig	**98.62**	**3771**	**2.11%**
Soundex	98.31	5761	3.22%
Culture_default + Orig	98.16	4712	2.63%
Double Metaphone	98.09	6595	3.69%
Culture_relative + Orig	**97.81**	**2292**	**1.28%**
ODM	97.72	3620	2.02%
Metaphone	97.65	4771	2.67%
Culture_person + Orig	97.57	2828	1.58%
Edit	97.40	3280	1.83%
NYSIIS	97.21	5847	3.27%
Culture_default	96.96	4712	2.63%
Orig	**94.32**	**1895**	**1.06%**
Culture_relative	**90.30**	**1191**	**0.67%**
Culture_person	83.11	1875	1.05%

Table 1. Given name piece cost & recall.

Looking at these results, we see that in our data, matched pairs of individuals have at least one original given name piece in common 94.32% of the time. Using Culture_relative along with the original name improves recall to 97.81% without increasing the cost by much (1895 to 2292 hits). Using ODM instead of Culture_relative improves recall a bit more with a somewhat significant increase in the number of hits. Finally, the Universal catalog does further improve recall just slightly, but at a large cost in terms of the number of hits.

Interestingly, none of the algorithmic methods did quite as well as these several versions of the hand-built catalogs did, as long as the original name was included in the queries along with the catalog's standard spelling. The original name was not included with the algorithmic methods because those methods are guaranteed to match whenever the original name matches, so there would be no point in doing so, whereas the catalog methods sometimes failed to standardize names, such as when the name was not in the catalog or a culture could not be determined.

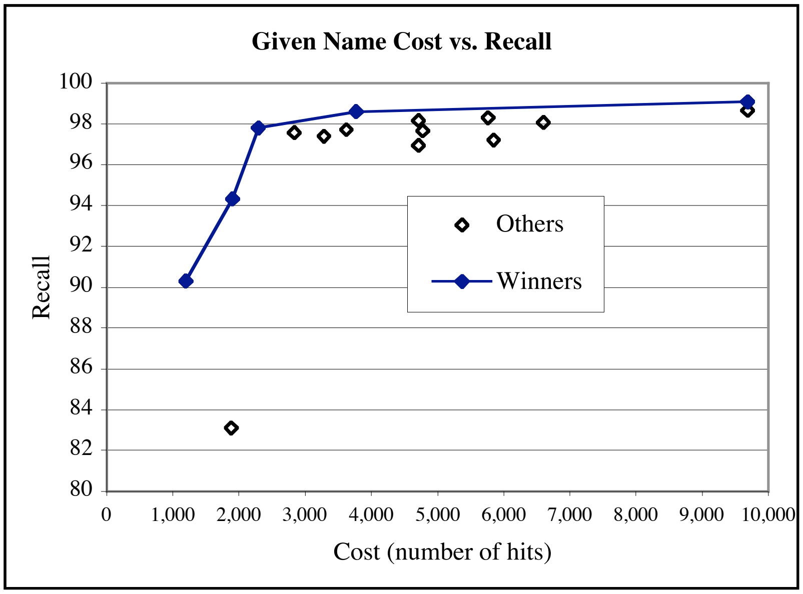

Figure 1. Given name cost vs. recall.

4.5. Surname results

Table 2 displays similar results for the same set of standardization techniques, but this time operating on the surnames of the individuals.

The best recall in this table is only 93.41%. This is partly due to the fact that some people in the database simply did not have a surname (such as when the maiden name of a wife is unknown, or in early records where some people did not have surnames). Another part of this is due to the fact that people often have 2 (and sometimes more) given names, while it is rare for people to have multiple surnames. Having multiple names allows more ways to find a match.

Note also, however, that the number of hits is very low for the "winning" techniques. Whereas getting close to 94% recall cost almost 2000 hits on given names, it cost less than 100 for surnames.

The curve in Figure 2 displays these results graphically, and illustrates the fact that the recall shoots up sharply without much additional cost for the winning techniques, and also makes it clear that there are several techniques that require a lot more cost for less recall than some of the winning methods.

Surname Fields	Recall	AvgHits	% of Db
ODM + Orig	**93.41**	**99.9**	0.06%
ODM	**92.92**	**94.8**	0.05%
Universal + Orig	89.39	332.4	0.19%
Double Metaphone	89.24	264.4	0.15%
Soundex	89.22	260.5	0.15%
Culture_relative + Orig	**88.79**	**75.1**	**0.04%**
NYSIIS	88.57	150.3	0.08%
Metaphone	88.35	181.0	0.10%
Culture_person + Orig	**88.02**	**72.2**	**0.04%**
Culture_default + Orig	87.79	95.5	0.05%
Edit	87.59	123.2	0.07%
Universal	86.28	332.1	0.19%
Orig	**84.62**	**61.1**	**0.03%**
Culture_default	79.58	94.7	0.05%
Culture_relative	**73.50**	**53.1**	**0.03%**
Culture_person	**67.60**	**48.3**	**0.03%**

Table 2. Surname cost & recall.

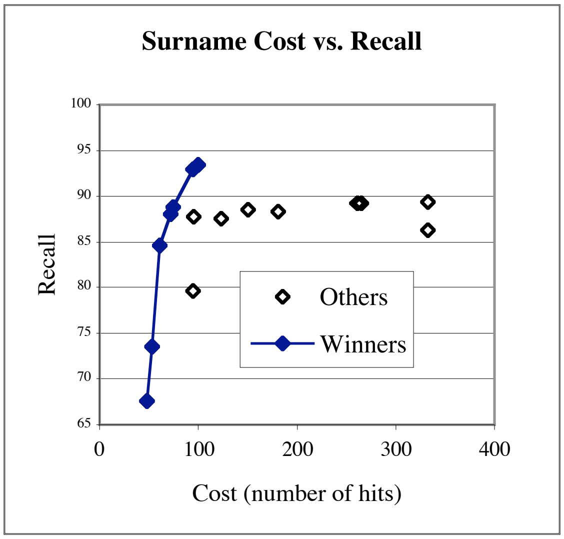

Figure 2. Surname cost & recall

4.6. Given name and surname

Table 3 shows results of issuing queries against the database using both given names and surnames. For example, a query for someone named "Jake Albert Jones" using the method "Orig+ODM" might look like "select individuals where given=jake or given=albert or surname=jones or odm_given=JACOB or odm_given=ALBERT or odm_surname=JONES".

The one case that is different is the row labeled "Orig (swap)". In that case, given names and surnames are indexed in a combined "full_name_piece" field, which allows cases to be brought together in which the surname and given name are (perhaps accidentally) swapped. For example, it brought together a case where "Elizabeth Rebecca" and "Rebecca Elizabeth" were really the same person, and the second name in each case was erroneously identified as the surname.

In this case, the accuracy is quite high for all of the methods, because between the given name and surname pieces, there are usually several opportunities to find a match. However, once again the ODM and Culture_relative methods get higher recall with less hits than the algorithmic approaches.

The Orig + Swap method is able to pick up about one-forth of the cases that the original name alone misses without much increase in cost. This is likely due to the fact that most given names do not appear often as surnames and vice-versa, so many of the additional cases that are brought together by allowing swaps turn out to be cases where the names were indeed misclassified in the first place.

Given + Surname Fields	Recall	AvgHits	% of Db
ODM + Orig	**99.68**	**3850**	**2.15%**
Soundex	99.54	5998	3.35%
Universal + Orig	99.42	9990	5.58%
NYSIIS	99.41	5976	3.34%
Culture_relative + Orig	**99.35**	**2348**	**1.31%**
Culture_person + Orig	99.25	2882	1.61%
Metaphone	99.20	4931	2.76%
Double Metaphone	99.20	6835	3.82%
Culture_default + Orig	99.16	4788	2.68%
Orig + Swap	**98.53**	**2135**	**1.19%**
Orig	**98.00**	**1939**	**1.08%**
Edit + Orig	98.00	1939	1.08%

Table 3. Cost and recall on given name + surname fields

5. Conclusions

The empirical results presented in this paper demonstrated that—on this set of data at least—the manually created catalogs outperformed the algorithmic methods in terms of recall and average number of hits per query. The original name pieces also provided the lowest cost for their (lower) levels of recall. Thus, when such catalogs are available and have been built with care, they can be very useful. In such cases, it appears that making use of cultural information (and taking into account the events of close relatives in order to do so) can also improve both cost and recall.

On the other hand, it must be pointed out that manually creating such catalogs is quite labor-intensive in terms of having real people look at hundreds of thousands of names, so in some situations it is more appropriate to rely on an algorithmic method after all, since some of them were able to achieve reasonably high accuracy with very little human effort.

One other caveat to keep in mind is that the data used for this paper may have different characteristics from other databases or situations. For example, much of the data is the result of submissions by individuals who may have already mapped name variations to a common or "cleaned up" version of the name. In other situations, such as searching for names in a database of raw vital record transcriptions, it may be necessary to handle variants that do not already appear in a catalog, in which case name comparison functions such as Jaro-Winkler or Edit Distance could become more useful.

References

Russell, Robert C., (1918), "Specification of Letters", United States Patent Office, Patent number 1,261,167.

Philips, Lawrence, (1990), "Hanging on the Metaphone", *Computer Language*, vol. 7, no. 12, pp. 39-43.

Philips, Lawrence, (2000), "The Double Metaphone Search Algorithm", *C/C++ Users Journal.*

Taft, Robert L., (1970), "Name Search Techniques", *New York State Identification and Intelligence.*