Identification of Confusable Drug Names: A New Approach and Evaluation Methodology

Grzegorz Kondrak
Department of Computing Science
University of Alberta
Edmonton, Alberta, Canada, T6G 2E8
kondrak@cs.ualberta.ca

Bonnie Dorr
Institute for Advanced Computer Studies &
Department of Computer Science
University of Maryland
College Park, 20742, USA
bonnie@umiacs.umd.edu

Abstract

This paper addresses the mitigation of medical errors due to the confusion of sound-alike and look-alike drug names. Our approach involves application of two new methods—one based on orthographic similarity ("look-alike") and the other based on phonetic similarity ("sound-alike"). We present a new recall-based evaluation methodology for determining the effectiveness of different similarity measures on drug names. We show that the new orthographic measure (BI-SIM) outperforms other commonly used measures of similarity on a set containing both look-alike and sound-alike pairs, and that the feature-based phonetic approach (ALINE) outperforms orthographic approaches on a test set containing solely sound-alike confusion pairs. However, an approach that combines several different measures achieves the best results on both test sets.

1 Introduction

Many hundreds of drugs have names that either look or sound so much alike that doctors, nurses and pharmacists can get them confused, dispensing the wrong one in errors that can injure or even kill patients. In the United States alone, an estimated 1.3 million people are injured each year from medication errors, such as administering the wrong dose or the wrong drug (Lazarou et al., 1998).[1] The U.S. Food and Drug Administration has sought to mitigate this threat by ensuring that proposed drug names that are too similar to pre-existing drug names are not approved (Meadows, 2003).

A number of different lexical similarity measures have been applied to the problem of identifying confusable drug names. Lambert et al. (1999) tested twenty two distinct methods on a set of drug names extracted from published reports of medication errors. The methods included well-known universal measures, such as edit distance and longest common subsequence, several variations of measures based on counting common letter n-grams, and measures designed specifically for associating phonetically similar names, such as Soundex and Editex. They identified the normalized edit distance, Editex, and a trigram-based measure as the most accurate.

The evaluation methodology of Lambert et al. (1999) involves repeated selection of cut-off thresholds in order to compute precision and recall on a test set that contains equal number of positive and negative examples of confusable drug name pairs. However, our own experience with systems for automatic detection of potential drug-name confusions suggests that the usual approach is to examine a fixed number of most similar candidates rather than all candidates with similarity above certain threshold. Moreover, the number of non-confusable pairs can be expected to greatly exceed the number of confusable pairs.

We present a different method of evaluating the accuracy of a measure. Starting from a set of confusable drug name pairs, we combinatorially induce a much larger set of negative examples. The recall is calculated against an on-line gold standard for each potentially confusable drug name considering only the top k candidate names returned by a similarity measure. The recall values are then aggregated using the technique of macro-averaging (Salton, 1971).

We formulate a general framework for representing word similarity measures based on n-grams, and propose a new measure of orthographic similarity called BI-SIM that combines the advantages of several known measures. Using our recall-based evaluation methodology, we show that this new measure performs better on a U.S. pharmacopeial gold standard than the measures identified as the most accurate by Lambert et al. (1999).

Some potential drug-name confusions can be attributed solely to high phonetic similarity. Consider the example of *Xanax* vs. *Zantac*—two brand names that the *Physicians' Desk Reference* (PDR) warns may be "mistaken for each other ... lead[ing]

[1] For example, a patient needed an injection of Narcan but instead got the drug Norcuron and went into cardiac arrest.

	Distance	Similarity
Orthographic	EDIT NED	N-GRAM LCSR
Phonetic	SOUNDEX EDITEX	ALINE

Table 1: Classification of word distance and similarity measures.

to serious medication errors” (24th Ed., 2003). The phonetic transcription of the two names, [zænæks] and [zæntæk], reveals their sound-alike similarity that is not apparent in their orthographic form. For the detection of sound-alike confusion pairs, we apply the ALINE phonetic aligner (Kondrak, 2000), which estimates the similarity between two phonetically-transcribed words. We demonstrate that ALINE outperforms orthographic approaches on a test set containing sound-alike confusion pairs.

The next section describes several commonly-used measures of word similarity. After this, we present two new methods for identifying look-alike and sound-alike drug names. We then compare the effectiveness of various measures using our recall-based evaluation methodology on a U.S. pharmacopeial gold standard and on another test set containing sound-alike confusion pairs. We conclude with a discussion of our experimental results.

2 Background

Drug-name matching refers to the process of string matching to rank similarity between drug names. There are two classes of string matching: orthographic and phonetic. For each of these, there are two methods of matching: distance and similarity. If two drug names are confusable, their distance should be small and their similarity should be large. Some examples of orthographic and phonetic algorithms for both distance- and similarity-based approaches are shown in Table 1.

In the remainder of this section, we describe a number of measures that have been applied to the problem of identifying confusable drug names. Specific examples of values obtained by the measures are provided in Table 2.

String-edit distance (Wagner and Fischer, 1974) (EDIT) (also known as Levenshtein distance) counts up the number of steps it takes to transform one string into another, where the cost of substitution is the same as the cost of insertion or deletion. A normalized edit distance (NED) is calculated by dividing the total edit cost by the length of the longer string.

The longest common subsequence ratio (Melamed, 1999) (LCSR) is computed by dividing the length of the longest common subsequence by the length of the longer string. LCSR is closely related to normalized edit distance. If the cost of substitution is at least twice the cost of insertion/deletion and the strings are of equal length, LCSR is equivalent to the normalized edit distance.

Measure	*Zantac/ Xanax*	*Zantac/ Contac*	*Xanax/ Contac*
EDIT	3	2	4
NED	0.500	0.333	0.667
LCSR	0.500	0.667	0.333
BIGRAM	0.222	0.600	0.000
TRIGRAM-2B	0.000	0.333	0.000
SOUNDEX	3	1	3
EDITEX	5	2	7
ALINE	9.542	9.333	8.958
BI-SIM	0.417	0.583	0.250
TRI-SIM	0.333	0.500	0.167
PREFIX	0.000	0.000	0.000

Table 2: Examples of values returned by various measures.

In n-gram measures, the number of n-grams that are shared by two strings is doubled and then divided by the total number of n-grams in each string:

$$\frac{2 \times |n\text{-}grams(x) \cap n\text{-}grams(y)|}{|n\text{-}grams(x)| + |n\text{-}grams(y)|}$$

where n-*grams(x)* is a multi-set of letter n-grams in x. This formula is often referred to as the *Dice coefficient*. A slight variation of this measure is obtained by adding extra symbols, such as spaces, before and/or after each string (Lambert et al., 1999). The modification is designed to increase sensitivity to the beginnings and endings of words. For example, TRIGRAM-2B is calculated by applying the Dice formula with $n = 3$ after adding two spaces before each string. In this paper, we consider two specific variants: BIGRAM, which is the most basic formulation, and TRIGRAM-2B.[2]

SOUNDEX (Hall and Dowling, 1980) is an approximation to phonetic name matching. SOUNDEX transforms all but the first letter to numeric codes (see Table 3) and after removing zeroes truncates the resulting string to 4 characters. For the purposes of comparison, we implemented a SOUNDEX-based similarity measure that returns the edit distance between the corresponding codes.

EDITEX (Zobel and Dart, 1996) is another quasi-phonetic measure that combines edit distance with a letter-grouping scheme similar to SOUNDEX (Table 3). As in SOUNDEX, the codes are designed

[2] TRIGRAM-2B was identified by Lambert et al. (1999) as particularly effective for identifying confusable drug name pairs.

Code	SOUNDEX	EDITEX
0	a e h i o u w y	a e i o u y
1	b f p v	b p
2	c g j k q s x z	c k q
3	d t	d t
4	l	l r
5	m n	m n
6	r	g j
7		f p v
8		s x z
9		c s z

Table 3: Character conversion codes in SOUNDEX and EDITEX.

to identify letters that have similar pronunciations, but the corresponding sets of letters are not disjoint. The edit distance between letters that belong to the same group is smaller than the edit distance between other letters. Additional rules are aimed at eliminating silent and reduplicated letters.

3 Phonetic Similarity: ALINE

The ALINE cognate matching algorithm (Kondrak, 2000) assigns a similarity score to pairs of phonetically-transcribed words on the basis of the decomposition of phonemes into elementary phonetic features. The algorithm was initially designed to identify and align cognates in vocabularies of related languages (e.g. *colour* and *couleur*). Nevertheless, thanks to its grounding in universal phonetic principles, the algorithm can be used for estimating the similarity of any pair of words, including drug names. Furthermore, unlike SOUNDEX and EDITEX, ALINE is completely language-independent.

The principal component of ALINE is a function that calculates the similarity of two phonemes that are expressed in terms of about a dozen binary or multi-valued phonetic features (*Place*, *Manner*, *Voice*, etc.). Feature values are encoded as floating-point numbers in the range $[0, 1]$. For example, the feature *Manner* can take any of the following seven values: *stop* = 1.0, *affricate* = 0.9, *fricative* = 0.8, *approximant* = 0.6, *high vowel* = 0.4, *mid vowel* = 0.2, and *low vowel* = 0.0. The numerical values reflect the distances between vocal organs during speech production. The phonetic features are assigned *salience* weights that express their relative importance.

The overall similarity score and optimal alignment of two words—computed by a dynamic programming algorithm (Wagner and Fischer, 1974)—is the sum of individual similarity scores between pairs of phonemes. A constant insertion/deletion penalty is applied for each unaligned phoneme. Another constant penalty is set to reduce relative importance of the vowel—as opposed to consonant—phoneme matches. The similarity value is normalized by the length of the longer word.

ALINE's behavior is controlled by a number of parameters: the maximum phonemic score, the insertion/deletion penalty, the vowel penalty, and the feature salience weights. The parameters have default settings for the cognate matching task, but these settings may not be appropriate for drug-name matching. The settings can be optimized (tuned) on a training set that includes positive and negative examples of confusable name pairs.

4 Orthographic Similarity: BI-SIM

An analysis of the reasons behind the unsatisfactory performance of commonly used measures led us to propose a new measure of orthographic similarity: BI-SIM.[3] Below, we describe the inherent strengths and weaknesses of n-gram and subsequence-based approaches. Next, we present a new, generalized framework that characterizes a number of commonly used similarity measures. Following this, we describe the parametric settings for BI-SIM—a specific instantiation of this generalized framework.

4.1 Problems with Commonly Used Measures

The Dice coefficient computed for bigrams (BIGRAM) is an example of a measure that is demonstrably inappropriate for estimating word similarity. Because it is based exclusively on complete bigrams, it often fails to discover any similarity between words that look very much alike. For example, it returns zero on the pair *Verelan/Virilon*. In addition, it violates a desirable requirement of any similarity measure that the maximum similarity of 1 should only result when comparing identical words. In particular, non-identical pairs[4] like *Xanex/Nexan*—where *all* bigrams are shared—are assigned a similarity value of 1. Moreover, it sometimes associates bigrams that occur in radically different word positions, as in the pair *V**ol**taren/Tramad**ol***. Finally, the initial segment, which is arguably the most important in determining drug-name confusability,[5] is actually given a *lower* weight than other segments because it participates in only one bigram. It is therefore surprising that BIGRAM has been such a popular choice of measure for computing word similarity.

LCSR is more appropriate for identifying potential drug-name confusability because it does not rely

[3] BI-SIM was developed before we conducted the experiments described in Section 6.

[4] This observation is due to Ukkonen (1992).

[5] 74.2% of the confusable pairs in the pharmacopeial gold standard (Section 6) have identical initial segments.

on (frequently imprecise) bigram matching. However, LCSR is weak in its tendency to posit non-intuitive links, such as the ones between segments in *Ben***adr***yl/C***ard***u***r***a*. The fact that it returns the same value for both **Am***aryl*/**Am***ikin* and **A***mary***l**/**Al***toce* can be attributed to lack of context sensitivity.

4.2 A Generalized N-gram Measure

Although it may not be immediately apparent, LCSR can be viewed as a variant of the n-gram approach. If n is set to 1, the Dice coefficient formula returns the number of shared *letters* divided by the average length of two strings. Let us call this measure UNIGRAM. The main difference between LCSR and UNIGRAM is that the former obeys the *no-crossing-links constraint*, which stipulates that the matched unigrams must form a subsequence of both of the compared strings, whereas the latter disregards the order of unigrams. E.g., for *pat/tap*, LCSR returns 0.33 because the length of the longest common subsequence is 1, while UNIGRAM returns 1.0 because all letters are shared. The other, minor difference is that the denominator of LCSR is the length of the longer string, as opposed to the average length of two strings in UNIGRAM. (In fact, LCSR is sometimes defined with the average length in the denominator.)

We define a generalized measure based on n-grams with the following parameters:

1. The value of n.
2. The presence or absence of the no-crossing-links constraint.
3. The number of segments appended to the beginning and the end of the strings.
4. The length normalization factor: either the maximum or the average length of the strings.

A number of commonly used similarity measures can be expressed in the above framework. The combination of $n = 1$ with the no-crossing-links constraint produces LCSR. By selecting $n = 2$ and the *average* normalization factor, we obtain the BIGRAM measure. Thirteen out of twenty two measures tested by Lambert et al. (1999) are variants that combine either $n = 2$ or $n = 3$ with various lengths of appended segments.

So far, we have assumed that there are only two possible values of n-gram similarity: identical or non-identical. This need not be the case. Obviously, some non-identical n-grams are more similar than others. We can define a similarity scale for two n-grams as the number of identical segments in the corresponding positions divided by n:

$$s(x_1 \ldots x_n, y_1 \ldots y_n) = \frac{1}{n} \sum_{i=1}^{n} id(x_i, y_i),$$

where $id(a, b)$ returns 1 if a and b are identical, and 0 otherwise. The scale distinguishes n levels of similarity, including 1 for identical bigrams, and 0 for completely distinct bigrams.[6]

The notion of similarity scale between n-grams requires clarification in the case of n-grams partially composed of segments appended to the beginning or end of strings. Normally, extra affixes are composed of one or more copies of a unique special symbol, such as space, that does not belong to the string alphabet. We define an *alphabet* of special symbols that contains a unique symbol for each of the symbols in the original string alphabet. The extra affixes are assumed to contain copies of special symbols that correspond to the initial symbol of the string. In this way, the similarity between pairs of n-grams in which one or both of the n-grams overlap with an extra affix is guaranteed to be either 0 or 1.

4.3 BI-SIM

We propose a new measure of orthographic similarity, called BI-SIM, that aims at combining the advantages of the context inherent in bigrams, the precision of unigrams, and the strength of the no-crossing-links constraint. BI-SIM belongs to the class of n-gram measures defined above. Its parameters are: $n = 2$, the no-crossing-links constraint enforced, a single segment appended to the beginning of the string, normalization by the length of the longer string, and multi-valued n-gram similarity.

The rationale behind the specific settings is as follows. $n = 2$ is a minimum value that provides context for matching segments within a string. The no-crossing-links constraint guarantees the sequentiality of segment matches. The segment added to the beginning increases the importance of the match of initial segment. The normalization method favors associations between words of similar length. Finally, the refined n-gram similarity scale increases the resolution of the measure.

BI-SIM is defined by the following recurrence:

$$\begin{aligned} f(i,j) \quad &= \quad max(f(i-1,j), f(i,j-1), \\ &\qquad f(i-1,j-1) + s(x_{i-1}x_i, y_{j-1}y_j)), \end{aligned}$$

[6] The scale could be further refined to include more levels of similarity. For example, bigrams that are frequently confused because of their typographic or cursive shape, such as *en/im*, could be assigned a similarity value that corresponds to the frequency of their confusions.

where s refers to the n-gram similarity scale defined in Section 4.2, and x_0 and y_0 are the appended segments. Furthermore, $f(i, j)$ is defined to be 0 if $i = 0$ or $j = 0$. The recurrence relation exhibits strong similarity to the relation for computing the longest common subsequence except that the subsequence is composed of bigrams rather than unigrams, and the bigrams are weighted according to their similarity. Assuming that the segments appended to the beginning of each string are chosen according to the rule specified in Section 4.2, the returned value of BI-SIM always falls in the interval $[0, 1]$. In particular, it returns 1 if and only if the strings are identical, and 0 if and only if the strings have no segments in common.

BI-SIM can be seen as a generalization of LCSR: the setting of $n = 1$ reduces BI-SIM to LCSR (which could also be called UNI-SIM). On the other hand, the setting of $n = 3$ yields TRI-SIM. TRI-SIM requires two extra symbols at the beginning of the string.

5 Evaluation Methodology

We designed a new method for evaluating the accuracy of a measure. For each drug name, we sort all the other drug names in the test set in order of decreasing value of similarity. We calculate the *recall* by dividing the number of true positives among the top k names by the total number of true positives for this particular drug name, i.e., the fraction of the confusable names that are discovered by taking the top k similar names. At the end we apply an information-retrieval technique called *macro-averaging* (Salton, 1971) which averages the recall values across all drug names in the test set.[7]

Because there is a trade-off between recall and the k threshold, it is important to measure the recall at different values of k. Table 4 shows the top 8 names that are most similar to *Toradol* according to the BI-SIM similarity measure. A '+'/'–' mark indicates whether the pair is a true confusion pair. The pairs are listed in rank order, according to the score assigned by the indicated algorithm. Names that return the same similarity value are listed in the reverse lexicographic order. Since the test set contains four drug names that have been identified as confusable with *Toradol* (*Tramadol*, *Torecan*, *Tegretol*, and *Inderal*), the recall values are 0.50 for $k = 5$, and for 0.75 for $k = 8$.

	Name	Score	+/–	Recall
1.	*Tramadol*	0.6875	+	0.25
2.	*Tobradex*	0.6250	–	0.25
3.	*Torecan*	0.5714	+	0.50
4.	*Stadol*	0.5714	–	0.50
5.	*Torsemide*	0.5000	–	0.50
6.	*Theraflu*	0.5000	–	0.50
7.	*Tegretol*	0.5000	+	0.75
8.	*Taxol*	0.5000	–	0.75

Table 4: Top 8 names that are most similar to *Toradol* according to the BI-SIM similarity measure, and the corresponding recall values.

6 Experiments and Results

We conducted two experiments with the goal of evaluating the relative accuracy of several measures of similarity in identifying confusable drug names. The first experiment was performed against an online gold standard: the *United States Pharmacopeial Convention Quality Review, 2001* (henceforth the *USP set*). The USP set contains both look-alike and sound-alike confusion pairs. We used 582 unique drug names from this source to combinatorically induce 169,071 possible pairs. Out of these, 399 were true confusion pairs in the gold standard. The maximum number of true positives was 6, but for the majority of names (436 out of 582), only one confusable name is identified in the gold standard. On average, the task was to identify 1.37 true positives among 581 candidate names.

We computed the similarity of each name pair using the following similarity measures: BIGRAM, TRIGRAM-2B, LCSR, EDIT, NED, SOUNDEX, EDITEX, BI-SIM, TRI-SIM, ALINE and PREFIX. PREFIX is a baseline-type similarity measure that returns the length of the common prefix divided by the length of the longer string. In addition, we calculated the COMBINED measure by taking the simple average of the values returned by PREFIX, EDIT, BI-SIM, and ALINE.

In order to apply ALINE to the USP set, all drug names were transcribed into phonetic symbols. This transcription was approximated by applying a simple set of about thirty regular expression rules. (It is likely that a more sophisticated transcription method would result in improvement of ALINE's performance.) In the first experiment, the parameters of ALINE were not optimized; rather, they were set according to the values used for a distinct task of cross-language cognate identification.

In Figure 1, the macro-averaged recall values achieved by several measures on the USP set are plotted against the cut-off k. Some measures have been left out in order to preserve the clarity of the plot. Table 5 contains detailed results for $k = 10$

[7] We could have also chosen to *micro*-average the recall values by dividing the total number of true positives discovered among the top k candidates by the total number of true positives in the test set. The choice of macro-averaging over micro-averaging does not affect the relative ordering of similarity measures implied by our results.

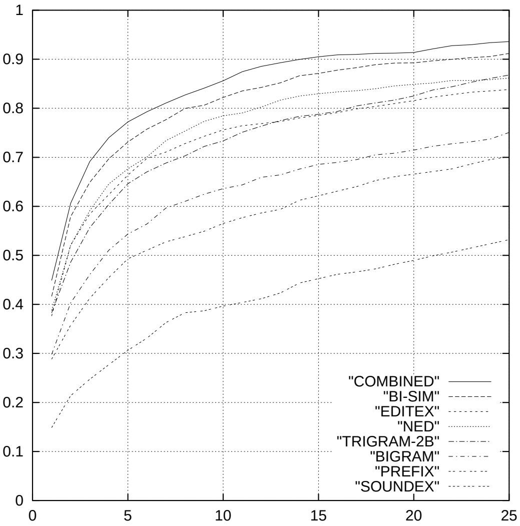

Figure 1: Recall at various thresholds for the USP test set.

	USP Set		Phono Set	
	top 10	top 20	top 10	top 20
PREFIX	0.5651	0.6658	0.2981	0.3478
EDIT	0.7506	0.8130	0.5139	0.6410
NED	0.7846	0.8489	0.5590	0.6639
LCSR	0.7375	0.8333	0.4663	0.5769
BIGRAM	0.6362	0.7148	0.3560	0.4400
TRIGRAM-2B	0.7335	0.8251	0.4674	0.5355
SOUNDEX	0.3965	0.4898	0.2331	0.3326
EDITEX	0.7558	0.8155	0.5864	0.6911
ALINE	0.7503	0.8303	0.5825	0.6873
BI-SIM	0.8220	0.8927	0.4838	0.6590
TRI-SIM	0.8324	0.8946	0.4782	0.6245
COMBINED	0.8560	0.9137	0.6462	0.7737

Table 5: Recall at $k = 10$ and $k = 20$ for both the USP and the sound-alike test sets.

and $k = 20$ for all measures.

Since the USP set contains both look-alike and sound-alike name pairs, we conducted a second experiment to compare the performance of various measures on sound-alike pairs only. We used a proprietary list of 276 drug names identified by experts as “names of concern” for 83 “consult” names. None of the “consult” names and only about 25% of the “names of concern” are in the USP set, i.e., there are no true positive pairs shared between the two sets. The maximum number of true positives was 11, while the average for all names was 3.33.

The measures were applied to calculate the similarity between each of the 83 “consult” names and a list of 2596 drug names. The results are shown in Figure 2. Since the task, which involved identifying, on average, 3.33 true positives among 2596 candidates, was more challenging, the recall values are lower than in Figure 1. All drug names were first converted into a phonetic notation by means of a set of regular expression rules. (We found that phonetic transcription led to a slight improvement in the recall values achieved by the orthographic measures.) The parameters of ALINE used in this experiment were optimized beforehand on the USP set.

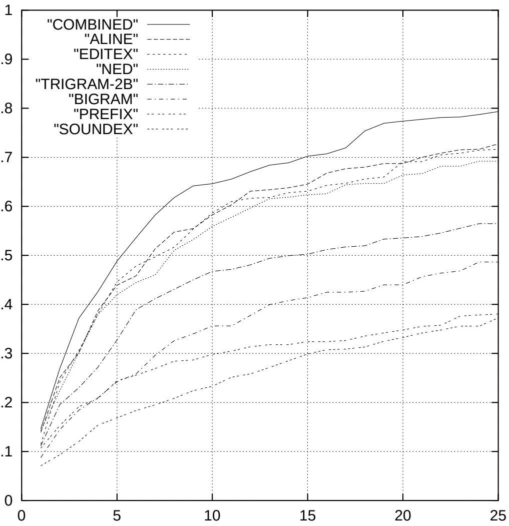

Figure 2: Recall at various thresholds for the sound-alike test set.

7 Discussion

The results described in Section 6 clearly indicate that BI-SIM and TRI-SIM, the newly proposed measures of similarity, outperform several currently used measures on the USP test set regardless of the choice of the cutoff parameter k. However, a simple combination of several measures achieves even higher accuracy. On the sound-alike confusion set, EDITEX and ALINE are the most effective. The accuracy achieved by the best measures is impressive. For the combined measure, the average recall on the USP set exceeds 90% with only the 15 top candidates considered.

The USP test set has its limitations. The set includes pairs that are considered confusable for other reasons than just phonetic or orthographic similarity, including illegible handwriting, incomplete knowledge of drug names, newly available products, similar packaging or labeling, and incorrect selection of a similar name from a computerized product list. In many cases, the names do not sound or look alike, but when handwritten or communicated verbally, these names have caused or could cause a mix-up. On the other hand, many clearly confusable name pairs are not identified as such (e.g. *Erythromycin/Erythrocin*, *Neosar/Neoral*, *Lorazepam/Flurazepam*, *Erex/Eurax/Urex*, etc.).

All similarity measures have their own

strengths and weaknesses. N-GRAM is effective at recognizing pairs such as *Chlorpromazine/Prochlorperazine*, where a shorter name closely matches parts of the longer name. However, this advantage is offset by its poor performance on similar-sounding names with few shared bigrams (*Nasarel/Nizoral*). LCSR is able to identify pairs where common subsequences are interleaved with dissimilar segments, such as *Asparaginase/Pegaspargase*, but fails on similar sounding names where the overlap of identical segments is minimal (*Luride/Lortab*). ALINE detects phonetic similarity even when it is obscured by the orthography (eg. *Xanax/Zantac*), but phonetic transcription is required beforehand.

The idiosyncrasies of individual measures are attenuated when they are combined together, which may explain the excellent performance of the combined measure. Each measure is focused on a particular facet of string similarity: initial segments in PREFIX, phonetic sound-alike quality in ALINE, common clusters in bigram-based measures, overall transformability in EDIT, etc. For this reason, a synergistic blend of several measures achieves higher accuracy than any of its components.

Our experiments confirm that orthographic approaches are superior to their phonetic counterparts in tasks involving string matching (Zobel and Dart, 1995). Nevertheless, phonetic approaches identify many sound-alike names that are beyond the reach of orthographic approaches. In applications where the gap between spelling and pronunciation plays an important role, it is advisable to employ phonetic approaches as well. The two most effective ones are EDITEX and ALINE, but whereas ALINE is language-independent, EDITEX incorporates English-specific letter groups and rules.

8 Conclusion

We have investigated the problem of identifying confusable drug name pairs. The effectiveness of several word similarity measures was evaluated using a new recall-based evaluation methodology. We have proposed a new measure of orthographic similarity that outperforms several commonly used similarity measures when tested on a publicly available list of confusable drug names. On a test set containing solely sound-alike confusion pairs phonetic approaches, ALINE and EDITEX achieve the best results. Our results suggest that a linear combination of several measures benefits from the strengths of its components, and is likely to outperform any individual measure. Such a combined approach has the potential to provide the basis for automatic minimization of medication errors.

The task of computing similarity between words is also important in other contexts. When an entered name does not exist in a bibliographic database, it is desirable to retrieve names that sound similar. Information retrieval systems may need to expand the search in cases where a typed query contains errors or variations in spelling. A related task of the identification of cognates arises in statistical machine translation. The techniques discussed in this paper may also be applicable in those areas.

References

PDR 24th Ed. 2003. *Physicians' Desk Reference for Nonprescription Drugs and Dietary Supplements*. Thomson PDR, New York, NY.

Patrick A. V. Hall and Geoff R. Dowling. 1980. Approximate string matching. *Computing Surveys*, 12(4):381–402.

Grzegorz Kondrak. 2000. A New Algorithm for the Alignment of Phonetic Sequences. In *Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 288–295, Seattle, WA.

Bruce L. Lambert, Swu-Jane Lin, Ken-Yu Chang, and Sanjay K. Gandhi. 1999. Similarity As a Risk Factor in Drug-Name Confusion Errors: The Look-Alike (Orthographic) and Sound-Alike (Phonetic) Model. *Medical Care*, 37(12):1214–1225.

J. Lazarou, B.H. Pomeranz, and P.N. Corey. 1998. Incidence of Adverse Drug Reactions in Hospitalized Patients. *Journal of the American Medical Association*, 279:1200–1205.

Michelle Meadows. 2003. Strategies to Reduce Medication Errors. *U.S. Food and Drug Administration Consumer Magazine*, May-June.

I. Dan Melamed. 1999. Bitext maps and alignment via pattern recognition. *Computational Linguistics*, 25(1):107–130.

Gerard Salton. 1971. *The Smart System: Experiments in Automatic Document Processing*. Englewood Cliffs: Prentice Hall, NJ.

Esko Ukkonen. 1992. Approximate string-matching with q-grams and maximal matches. *Theoretical Computer Science*, 92:191–211.

Robert A. Wagner and Michael J. Fischer. 1974. The string-to-string correction problem. *Journal of the ACM*, 21(1):168–173.

Justin Zobel and Philip W. Dart. 1995. Finding approximate matches in large lexicons. *Software — Practice and Experience*, 25(3):331–345.

Justin Zobel and Philip Dart. 1996. Phonetic string matching: Lessons from information retrieval. In *Proceedings of the 19th International Conference on Research and Development in Information Retrieval*, pages 166–172.