Artificial Intelligence in Medicine (2005) xxx, xxx–xxx

ELSEVIER

ARTIFICIAL INTELLIGENCE IN MEDICINE

http://www.intl.elsevierhealth.com/journals/aiim

Automatic identification of confusable drug names☆

Grzegorz Kondrak[a,*], Bonnie Dorr[b]

[a] Department of Computing Science, University of Alberta, Edmonton, Alta., Canada T6G 2E8
[b] Institute for Advanced Computer Studies, Department of Computer Science, University of Maryland, College Park, MD 20742, USA

Received 27 December 2004; received in revised form 18 July 2005; accepted 25 July 2005

KEYWORDS
Drug names;
Lexical similarity;
Medical errors;
Evaluation methodology

Summary

Objective: 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.
Methods and material: We propose to address the problem through the application of two new methods—one based on orthographic similarity (''look-alike''), and the other based on phonetic similarity (''sound-alike''). In order to compare the effectiveness of the new methods for identifying confusable drug names with other known similarity measures, we developed a novel evaluation methodology.
Results: 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 a new feature-based phonetic approach (ALINE) outperforms orthographic approaches on a test set containing solely sound-alike pairs. However, an approach that combines several different measures achieves the best results on two test sets.
Conclusion: Our system is currently used as the basis of a system developed for the U.S. Food and Drug Administration for detection of confusable drug names.
© 2005 Published by Elsevier B.V.

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

☆ A preliminary version of this paper appeared in Proceedings of the 20th International Conference on Computational Linguistics, Geneva (2004) pp. 952–958.

* Corresponding author. Tel.: +1 780 492 1779; fax: +1 780 492 1071.
E-mail addresses: kondrak@cs.ualberta.ca (G. Kondrak), bonnie@umiacs.umd.edu (B. Dorr).
URL: http://www.cs.ualberta.ca/~kondrak, http://www.umiacs.umd.edu/users/bonnie

0933-3657/$ – see front matter © 2005 Published by Elsevier B.V.
doi:10.1016/j.artmed.2005.07.005

UNCORRECTED PROOF

or the wrong drug [1]. For example, a patient needed an injection of *Narcan*but instead got the drug *Norcuron* and went into cardiac arrest. The U.S. Food and Drug Administration (FDA) has sought to mitigate this threat by ensuring that proposed drug names that are too similar to pre-existing drug names are not approved [2]. This has motivated the research and design of algorithms underlying phonetic orthographic computer analysis (POCA), an operational system implemented by the Project Performance Corporation for the FDA.[1]

A number of different lexical similarity measures have been applied to the problem of identifying confusable drug names (henceforth referred to as *confusion pairs*). For example, 22 distinct methods were tested on a set of drug names extracted from published reports of medication errors [3]. The methods included well-known universal measures, such as edit distance, longest common subsequence, and 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. The normalized edit distance, Editex, and a trigram-based measure were identified as the most accurate.

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. We show that this new measure performs better on a U.S. pharmacopeial list of confusable drug names than the measures previously identified as the most accurate by [3].

In addition, we present techniques for detecting drug-name confusions that are attributed solely to high phonetic similarity. Consider the example of *Xanax* versus *Zantac* —two brand names that the *Physicians' Desk Reference* (PDR) warns may be "mistaken for each other ... lead[ing] to serious medication errors" [4]. The phonetic transcription of the two names, [zænæks] and [zæntæk], reveals a 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 [5], 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.

We present a novel method of evaluating the accuracy of a measure, which aims at emulating the perspective of a person involved in the process of approving a new drug name. Our approach is to average recall values for each drug name in the test set. The recall is calculated against a published list of confusable drug names considering only the top *k* potential confusion pairs returned by a similarity measure. The recall values are then aggregated using the technique of macro-averaging [6].

The next section provides the background for the problem we are addressing, several commonly-used measures of word similarity, and our methodology for evaluation. 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 list and on another test set containing sound-alike confusion pairs. We conclude with a discussion of our experimental results.

2. Background

The problem of automatic identification of confusable drug names can be stated as follows: given a large set of existing drug names, identify all pairs or sets of drug names that are potentially confusable with each other. An alternative formulation reflects the process of approving a newly proposed drug name: given a proposed drug name and a large set of existing drug names, identify all drug names in a large set of existing drug names that are potentially confusable with the proposed drug name. Our evaluation methodology is geared towards the latter formulation.

2.1. Cognitive model/description of the human task

In addition to the types of confusability that are inherent to the task of distinguishing drug names (illegible handwriting, incomplete knowledge of drug names, newly available products, etc.), the task of auditory word recognition is itself a difficult problem. For example, it has been demonstrated that similar words compete for recognition, with high-usage frequencies competing more than others; this means similar words that are frequent are more likely to generate confusions than similar words that are infrequent [7,8]. In addition, it has been suggested that the beginning of a word is more important than other parts in listening tasks [9]. Finally, in language production (spoken or written), words with similar output forms are sometimes confused, and even more common are words that are both similar in form and similar in meaning [10].

Although these factors have implications for predicting drug name confusions, we focus primarily on factors amenable to computerized string matching (to be described next). We note, however, that our

[1] See http://www.ppc.com/case_fdastudy.asp (last accessed: 8 August 2005).

UNCORRECTED PROOF

system provides a weight-tuning capability to assist in focusing on certain aspects of similarity (e.g. if certain types of confusions occur frequently or if certain portions of confusable names are more salient than others), but this would be application-dependent.

2.2. String-matching algorithms

The detection of confusable drug names is an application for string-matching algorithms, where two drug names may be compared and ranked according to their degree of potential confusability. Ideally, a string-matching algorithm would detect a large "similarity" (or a small "distance") between two drug names that are potentially confusable.[2]

String matching algorithms have been used to address a variety of problems in natural-language processing (NLP) including part-of-speech (POS) tagging [11], language identification [12], cognate matching [13–16], spelling correction [17], author identification [18], fast text searching [19], topic segmentation [20], text compression [21], and lexicon searching [22]. For each of these applications, there are two classes of string matching (orthographic and phonetic) and two methods of matching (distance and similarity). The different classes and methods of matching are described next.

2.3. Phonetic versus orthographic

The approaches to measuring word similarity can be divided into two groups. The *orthographic* approaches disregard the fact that alphabetic symbols express actual sounds, employing a binary identity function on the level of character comparison. The *phonetic* approaches, on the other hand, attempt to take advantage of the phonetic characteristics of individual sounds in order to estimate their similarity.

2.4. String similarity versus edit distance

String similarity measures estimate the similarity between two strings based on the number of characters they have in common. Edit distance measures count the number of steps required to transform one string into the other. Both measures may be used for either of the two classes of string matching (orthographic or phonetic).

2.5. Existing string-matching algorithms

Some examples of orthographic and phonetic algorithms for both distance- and similarity-based approaches are shown in Table 1. Specific examples of values obtained by the measures are provided in Table 2. Boldfaced measures (in Tables 1 and 2) indicate those that are described in detail in this paper. We now examine each of these measures, in turn.

PREFIX is a baseline-type similarity measure that returns the length of the common prefix divided by the length of the longer string. For example, the common prefix for *Tobradex* and *Torecan* has length 2 (*to*-) which, divided by the length of 8, yields 0.25.

String-edit distance [23] (EDIT) (also known as Levenshtein distance) counts the number of edit operations it takes to transform one string into another, where the cost of substitution is the same as the cost of insertion or deletion. For example, the edit distance between *Zantac* and *Xanax* is 3 because the transformation of the former into the latter involves two substitutions ($z \rightarrow x$ and $c \rightarrow x$) and one deletion (t). A normalized edit distance (NED) is calculated by dividing the total edit cost by the length of the longer string. The normalization is an attempt to remove the bias against longer strings that is inherent in the standard edit distance.

The longest common subsequence ratio [15] (LCSR) is computed by dividing the length of the longest common subsequence by the length of the longer string. The characters in a subsequence do not have to be contiguous. For example, the longest common subsequence of *Zantac* and *Xanax* has length 3 (a-n-a), so the LCS ratio for this pair is equal to 3 divided by 6 (the length of *Zantac*). LCSR is closely related to normalized edit distance. If the cost of substitution is at least twice the cost of

Table 1 Classification of word distance and similarity measures

	Distance	Similarity
Orthographic	EDIT	*N*-GRAM
	NED	LCSR
		BI-SIM
		TRI-SIM
Phonetic	SOUNDEX	**ALINE**
	EDITEX	

[2] As we will see in Section 2.7, drug name *similarity*is only one factor contributing to drug name *confusability*. Because intuitions about similarity between orthographic and phonological forms of words correlate with their confusability, we often use intuitions about similarity as a substitute for confusability, the latter being more difficult to measure directly. However, we do not believe that intuitions should be used as the basis for doing the task directly, as it is impractical in most applications where there are very large numbers of potentially confusable pairs.

UNCORRECTED PROOF

Table 2 Examples of values returned by various measures

Measure	*Zantac*/*Xanax*	*Zantac*/*Contac*	*Xanax*/*Contac*
PREFIX	0.000	0.000	0.000
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
BI-SIM	0.417	0.583	0.250
TRI-SIM	0.333	0.500	0.167
ALINE	9.542	9.333	8.958

insertion/deletion, the following equation holds for any two strings X and Y of equal length:

$$\mathrm{LCSR}(X, Y) = 1 - \mathrm{NED}(X, Y) \quad (1)$$

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:

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

where n-grams(x) is a multi-set of letter n-grams in x.[3] This formula is often referred to as the *Dice coefficient*. For example, the bigram similarity between {za, an, nt, ta, ac} and {co, on, nt, ta, ac} is $(2 \times 3)/(5+5) = 6/10 = 0.6$, because three of the bigrams are shared: {nt, ta, ac}. A slight variation of this measure is obtained by adding extra symbols, such as spaces, before and/or after each string [3]. This variation is designed to increase sensitivity to the beginnings and endings of words. Specifically, TRIGRAM-2B is calculated by applying the Dice formula with $n = 3$after adding two spaces before each string, so, for example, *Zantac* is decomposed into six trigrams: {__z,_za, zan, ant, nta, tac}. In this paper, we consider two specific variants of the Dice coefficient: BIGRAM, which is the most basic formulation, and TRIGRAM-2B, as particularly effective for identifying confusable drug name pairs.

SOUNDEX [24] is an approximation to phonetic name matching that transforms all but the first letter to numeric codes (first column of Table 3) and, after removing zeroes, truncates the resulting string to four characters. The purpose of the transformation is to convert similar-sounding words into the same four-character code. This approach is able to detect certain sound similarities, while missing others. For example, the approach is capable of finding a match between the two sound-alike words *king*and *khyngge* (k520, k520), but it is unable to detect a match between *knight* and *night*. Even worse, SOUNDEX matches radically different sounding words such as *pulpit* and *phlebotomy* (p413, p413). For the purpose of comparison, we implemented a SOUNDEX-based similarity measure that returns the edit distance between the corresponding codes. For example, the distance between the SOUNDEX renderings of *Zantac* (z532) and *Xanax* (x520) is 3.[4]

Table 3 Character conversion codes in SOUNDEX and EDITEX

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

EDITEX [26] is another quasi-phonetic measure that combines edit distance with a letter-grouping scheme similar to SOUNDEX (second column of Table 3). As in SOUNDEX, the codes are designed 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. For example, the EDITEX distance between *Zantac*and *Xanax* is 5, which is calculated by summing up the cost of three operations: $z \rightarrow x$ (same group—cost 1), $x \rightarrow c$ (different groups—cost 2), and deletion of t (cost 2).[5]

[3] A multi-set is a set-like object in which order of elements is ignored, but the multiplicity of elements is retained. For example, multi-sets {a, b, c} and {b, c, a} are equivalent, but {a, b, c} and {a, a, b, c} differ.

[4] PHONIX, a measure that is closely related to SOUNDEX, combines edit distance with a letter-grouping scheme after applying 160 letter-group transformations, e.g. $kn \rightarrow n$ [25]. We have not included PHONIX in this study because its performance has been found (independently) to be even worse than that of SOUNDEX [26], despite that it is 70 years more recent and was designed to replace SOUNDEX.

[5] Another possible phonetic measure is one proposed by [27] which compares syllable count, initial/final sounds, and stress locations. However, we have observed this to miss frequently confused pairs, e.g. *Sefotan*/*Seftin* (syllable count differs) or *Gelpad*/*Hypergel* (end/beginning sounds differ) that are easily identified by standard n-gram measures.

UNCORRECTED PROOF

2.6. Evaluation criteria

To evaluate the effectiveness of the above techniques for detecting potential drug name confusability—and to compare these techniques to those we have developed—we use the notion of *recall* from the field of information retrieval [28]. Recall is the ratio of true positives (correct answers) to the sum of true positives and false negatives (all positive instances). For example, an internet search-engine query returns an ordered list of pointers, which are judged to be either relevant or not. In that context, true positives are the retrieved relevant documents and false negatives are the non-retrieved relevant documents. Suppose we issue a query about the names of Canadian provinces and the response is "Alberta, Manitoba, Ontario, and Minnesota". The recall is 0.30 because the three are correct (out of 10 possible correct answers). In our context, recall is the percentage of confusion pairs in the data that are identified by the system. As we will see in Section 5, we compute recall at various cut-off thresholds.

Recall is often coupled with the notion of *precision*, which is the ratio of true positives (correct answers) to the sum of true positives and false positives (all answers). In our Canadian province example, the response "Alberta, Manitoba, Ontario, and Minnesota" corresponds to a precision of 0.75 (three out of the four answers are correct). Because our aim is to compute recall values at different cut-off thresholds, rather than at different precision values, our experiments do not include the computation of precision values.

2.7. Advancing the state of the art

We introduce two new methods for identifying confusable drug names and a novel evaluation methodology. We show that a 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 a new feature-based phonetic approach (ALINE) outperforms orthographic approaches on a test set containing solely sound-alike pairs. However, an approach that combines several different measures achieves the best results on two test sets.[6]

Our methods are intended to support the use of human-operated screening tools, not to replace the human entirely. We adopt the widely recognized view that "a system for evaluating the acceptability of new drug names would integrate [both] expert judgment and computerized name searches into a systematic and scientifically valid manner" [3]. The idea is to automate the portion of the process that computers do best (quick, precise, comprehensive access) and to use judgments from human experts about potential confusions resulting from factors that are less easily automatable, e.g. "poor handwriting, abbreviations, storage of drug products on shelves or crash carts, stress, fatigue, and distractions" [3].

The next two sections present the new orthographic and phonetic measures of similarity. Evaluation of the effectiveness of these approaches in the context of detecting drug name confusability is discussed in Section 6.

3. Orthographic similarity: N-SIM

In this section, we describe the inherent strengths and weaknesses of *n*-gram and subsequence-based approaches. Next, we present a new, generalized framework, N-SIM, that encompasses a number of commonly used similarity measures. Following this, we describe the parametric settings for BI-SIM—a specific instantiation of this generalized framework which is aimed at combining the advantages of LCSR and BIGRAM.[7]

3.1. Issues with commonly used orthographic 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 may fail 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. (Commonly used measures based on *n*-grams do not satisfy this requirement because non-identical pairs can have identical *n*-gram profiles [29].) For example, the pair *Xanex*/*Nexan*is assigned a similarity value of 1, since all four bigrams {xa, an, ne, ex} occur in both drug names. Moreover, it sometimes associates bigrams that occur in radically different word positions, as in the pair *Voltaren*/*Tramadol*. Finally, the initial letter, which is arguably the most

[6] Although the examples used throughout this paper focus on single-word drug names, the techniques for identifying confusable multi-word names are the same as for single-word names. A pair of multi-word names can be treated as a pair of very long single-word names or the multi-word names can be split into individual words.

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

UNCORRECTED PROOF

important in determining drug-name confusability,[8] is actually given a *lower* weight than other letters because it participates in only one bigram. Given these issues, it is therefore surprising that BIGRAM has been such a popular choice of measure for computing word similarity [30–32].

LCSR is more appropriate for identifying potential drug-name confusability because it does not rely on (frequently imprecise) bigram matching. However, LCSR is weak in its tendency to posit non-intuitive links, such as the ones between letters in *Benadryl* / *Cardura*. The fact that it returns the same value for both *Amaryl* /*Amikin* and *Amaryl*/*Altoce* can be attributed to lack of context sensitivity.

3.2. A generalized *n*-gram measure

Although it may not be immediately apparent, LCSR can be viewed as an *n*-gram measure. 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. For example, 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 [31].) Neither LCSR nor UNIGRAM add extra symbols to the beginning or end of the strings.

Taking into account the distinguishing features above, we define N-SIM, 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 length normalization factor: either the maximum or the average length of the strings.
(4) The number of symbols added to the beginning and the end of the strings.

A number of commonly used similarity measures can be expressed in the above framework. For example, 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. In fact, 13 out of 22 measures tested by [3] are variants that combine either $n = 2$ or 3 with various lengths of pre- and post-pended sequences.

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. Thus, in addition to the four parameters above, N-SIM includes a fifth parameter of variation, the notion of *similarity scale*. We define the similarity scale for two *n*-grams as as the number of identical letters in the corresponding positions divided by *n*:

$$s(x_1, \ldots, x_n, y_1, \ldots, y_n) = \frac{1}{n} \sum_{i=1}^{n} id(x_i, y_i) \qquad (3)$$

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 *n*-grams, and 0 for completely distinct *n*-grams.[9]

The notion of similarity scale between *n*-grams requires clarification in the case of *n*-grams partially composed of pre- and post-pended sequences. 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 letter in the original string alphabet. The extra affixes are assumed to contain copies of special symbols that correspond to the initial letter of the string.

To illustrate the notion of prepended sequences, consider the case of bigram matching between *Zantac* and *Xanax*. Fig. 1 shows one of the optimal alignments, with the partially and completely matching bigrams linked by dashed and solid lines, respectively. The extra prepended symbols are shown as uppercase letters. With the incorporation of similarity scale for *n*-grams, the optimal alignment is no longer defined by the maximum number of matches, but rather by the maximum total similarity score.

[8] 74.2% of the confusion pairs in the pharmacopeial list (Section 6) have identical initial letters, as opposed to only 6.5% of randomly selected pairs.

[9] 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. Such an approach differs from what we have reported in this paper in that it involves the use of features, such as *n*-grams and distances, to fit a curve to the subjective data.

UNCORRECTED PROOF

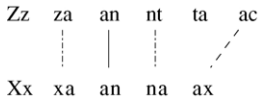

Figure 1 A sequence of matching bigrams between *Zantac* and *Xanax*.

3.3. BI-SIM

We propose a new measure of orthographic similarity, called BI-SIM, that aims at combining the the context sensitivity inherent in bigrams, the precision of unigrams, and the strength of the no-crossing-links constraint. BI-SIM is a specific instantiation of the N-SIM measure defined above. Its parameters settings are: $n = 2$, no-crossing-links constraint enforced, a single prepended symbol, 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 letters within a string. The no-crossing-links constraint guarantees the sequentiality of letter matches. The symbol added to the beginning of the string increases the importance of the match of initial letter. 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 based on the following recurrence:

$$f(i,j) = \max\big(f(i-1,j),\ f(i,j-1),\ f(i-1,j-1) + s(x_i x_{i+1}, y_j y_{j+1})\big) \quad (4)$$

where s refers to the n-gram similarity scale defined in Section 3.2, and x_1 and y_1 are the prepended symbols. 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. The value of BI-SIM is obtained by normalizing $f(|X|,|Y|)$ by $\max(|X|,|Y|)$, where $|X|$ and $|Y|$ are the original lengths of X and Y, respectively. Assuming the symbols prepended to the beginning of each string are chosen according to the rule specified in Section 3.2, 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 letters in common. The algorithm for computing BI-SIM is shown in Fig. 2.

Consider again the matching bigrams between *Zantac* and *Xanax* shown in Fig. 1. Since the length

BI-SIM (X, Y)

$k \leftarrow \text{length}(X)$
$l \leftarrow \text{length}(Y)$
$X \leftarrow x'_1 + X$
$Y \leftarrow y'_1 + Y$
for $i \leftarrow 0$ **to** k **do**
  $S[i,0] \leftarrow 0$
for $j \leftarrow 1$ **to** l **do**
  $S[0,j] \leftarrow 0$
for $i \leftarrow 1$ **to** k **do**
  for $j \leftarrow 1$ **to** l **do**
    $S[i,j] \leftarrow \max(S[i-1,j], S[i,j-1], S[i-1,j-1] + s(x_i x_{i+1}, y_j y_{j+1}))$
return $S[k,l]/\max(k,l)$

Figure 2 BI-SIM algorithm for computing bigram similarity of strings *X* and *Y*.

of the longer word is 6, the pair's BI-SIM score is $\frac{1\times1.0+3\times0.5}{6} = 0.25$.

The main innovation of BI-SIM is in generalizing the concept of the longest common subsequence to encompass bigrams, rather than just unigrams. 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, which requires two extra symbols at the beginning of the string.

4. Phonetic similarity: ALINE

In the preceding section, we proposed a new measure of orthographic similarity for identifying look-alike drug names. However, the detection of sound-alike confusion pairs often requires a different kind of approach. For this purpose, we employ ALINE [5], which computes phonetic similarity between pairs of phonetically-transcribed words. Its underlying principle is the decomposition of phonemes into elementary articulatory phonetic features.[10] 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

[10] Although perceptual differences are not necessarily well approximated by articulatory differences [33], the latter are much easier to quantify.

UNCORRECTED PROOF

used for estimating the similarity of any pair of words, including drug names. ALINE is written in C++ and runs under Unix. The executable version of the program and an online demo are publicly available.[11]

The principal component of ALINE is a function that calculates the similarity of two phonemes. Phonemes are expressed in terms of binary or multi-valued phonetic features. For example, the phoneme *n*, which is usually described as a *voiced alveolar nasal stop*, has the following feature values: $Place = 0.85$, $Manner = 0.6$, $Voice = 1$, and $Nasal = 1$, with the remaining features set to 0. In order to compute the phonetic distance between two phonemes, the differences between their numerical values for each feature are multiplied by the feature's salience weight, and the resulting values are summed up. The phonetic similarity score is then calculated by subtracting the distance from the maximum score. For the purpose of emphasizing consonant correspondences, the similarity score is further decreased if one or both of the phonemes are vowels.[12]

The feature set contains the following features: *Place*, *Manner*, *Voice*, *Syllabic*, *Nasal*, *Retroflex*, *High*, *Lateral*, *Aspirated*, *Back*, *Round*, and *Long*. A special feature *Double*, which has the same possible values as *Place*, indicates the second place of articulation. The above feature set is sufficient to account for phonemic contrasts in many languages, including English, French, Spanish, Portuguese, Italian, German, and Russian. If necessary, it can be extended to cover other languages.

The numerical feature values reflect the distances between vocal organs during speech production, and are based on the experimental measurements reported in [35]. They are encoded as floating-point numbers in the range [0, 1]. For example, the feature *Manner*, which, roughly speaking, refers to the degree of airstream opening in the vocal tract during phoneme articulation, can take any of the following seven values: $stop = 1.0$, $affricate = 0.9$, $fricative = 0.8$, $approximant = 0.6$, $highvowel = 0.4$, $midvowel = 0.2$, and $lowvowel = 0.0$.

An important component of ALINE's feature system is the notion of the *salience* weights that represent the relative importance of each feature. The principal features, *Place* and *Manner*, are assigned much higher saliences than less important features like *Aspirated* and *Round*. The default salience values were established by trial and error on a set of phoneme-aligned cognate pairs from various related languages.

The overall similarity score and optimal alignment of two words are computed by a dynamic programming algorithm [23]. The total score is the sum of individual similarity scores between pairs of phonemes in the optimal alignment. A constant insertion/deletion penalty is applied for each unaligned phoneme. The similarity value is normalized by the length of the longer word, so that it falls in the range [0,1]. ALINE incorporates a number of extensions to the basic dynamic programming, which have been proposed primarily to address issues in DNA alignment, but are also applicable in the context of computing phonetic word similarity. The extensions include: retrieving a set of best alignments [36], local and semiglobal alignment [37], and additional edit operations [38].[13]

The feature system of ALINE is highly dynamic because the phonetic similarity values between phonemes can be modified by changing both feature saliences and numerical values within features. Additional parameters include the maximum phonemic score, the insertion/deletion penalty, and the vowel penalty. 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 manually optimized (tuned) on a training set that includes positive and negative examples of confusable name pairs using the average identification accuracy as the objective function.

In order for ALINE to compute the similarity of pairs of drug names, the orthographic characters have to be transcribed into phonetic symbols. The transcription can be either performed manually or by means of an automatic program. Such programs are relatively straightforward for languages like Italian or Slovak, where letter-to-sound rules are transparent. Languages such as English or French require more complex approaches [39]. However, it is often sufficient to approximate the actual pronunciation with a simple set of grapheme-to-phoneme rules. In general, the better the quality of transcription, the more accurate estimate of phonetic similarity is provided by ALINE.

Unlike SOUNDEX and EDITEX, ALINE is not based on English-specific orthographic conventions. Its underlying phonetic feature system is sufficiently flexible to express any of the phonemes specified by the International Phonetic Alphabet [40]. The confusability of drug names may vary depending on the language in which they are pronounced. For exam-

[11] At http://www.cs.ualberta.ca/~kondrak (last accessed: 8 August 2005).

[12] Consonants have been shown to be more important than vowels in the perception of similarity. For example, Covington [34] assigns lower edit-distance cost to a mismatch of vowels than to a mismatch of consonants.

[13] The extensions are not used in the identification of confusable drug names.

UNCORRECTED PROOF

ple, *Clonidine*and *Klonopin* are less likely to be confused in languages where *c* and *k* represent two different sounds. Simply substituting an appropriate transcription program enables ALINE to effectively deal with this problem.

5. Evaluation methodology

We designed a new method for evaluating the accuracy of a similarity measure. Our aim was to emulate the perspective of a person involved in the process of approving a new drug name. Because of the sheer number of pharmaceutical products already in existence, it is very difficult for anyone to think of all possible drug names that may be confused with the newly proposed name. A computer program can facilitate this task by presenting the human expert with a ranked list of potential confusion pairs.[14] Obviously, only a manageable number of the most similar names should be provided to the user who makes the final decision about their potential confusability. However, the decision about the exact setting of the cut-off number should be left to the user. (The optimal number may also vary depending on the name being analyzed.)

Our evaluation approach is to average the recall values for each drug name in the test set with the cut-off number *k* as a parameter. Our preference for recall over precision is motivated by the desire to minimize the number of false negatives rather than avoid false positives. In other words, we aim to detect as many potentially confusable names as possible even at the cost of labelling as confusable a number of words that are not confusable.

As an example, consider the task of finding the names of drugs that are potentially confusable with *Toradol*. Table 4 shows the top 8 names that are most similar to *Toradol* according to the BI-SIM similarity measure. A '+' or ' −' mark indicates whether the pair is considered a true confusion pair with respect to a pharmacopeial list to be described in Section 6. The pairs are listed in rank order, according to the score assigned by the BI-SIM measure. Names that return the same similarity value are listed in the reverse lexicographic order. The test set contains exactly four drug names that have been identified as confusable with *Toradol*(*Tramadol*, *Torecan*, *Tegretol*, and *Inderal*).[15] Therefore, the recall values are 0.50 for $k = 5$, and for 0.75 for $k = 8$.

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

	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

Our evaluation procedure is applied to a specific measure of similarity as follows. First, for each drug name in the test set, we calculate the similarity between that name and all other names in our database. Then the similarity scores are sorted in the decreasing order, so that the names at the top of the list are the ones that have the highest similarity to our test name according to the evaluated measure. 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*[6] which averages the recall values across all drug names in the test set.[16] Because there is a trade-off between recall and the *k* threshold, we measure the overall average recall at different values of *k*.

Our evaluation methodology differs from that of [3] in that this earlier approach involved manual selection of *score* thresholds on a test set that contains an equal number of positive and negative instances of confusable drug name pairs. 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 the most similar potential confusion pairs rather than all pairs with similarity above a certain threshold. Moreover, in realistic settings, the number of non-confusable pairs greatly exceeds the number of confusable pairs. Thus, our evaluation methodology makes use of a fixed number of the most similar pairs in an environment where the number of non-confusable pairs is overwhelmingly higher than the number of confusable pairs.

[14] An interface with this capability has been implemented in the POCA system. This system is now under a continuing contract for the integration of improved algorithms described herein (see http://www.ppc.com/news_pressrelease_2004_fda.asp (last accessed: 8 August 2005)).

[15] *Inderal*'s rank is 151; thus it does not appear in the table.

[16] We could have also chosen to *micro*-average the recall values by dividing the total number of true positives among the top *k* potential confusion pairs 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.

UNCORRECTED PROOF

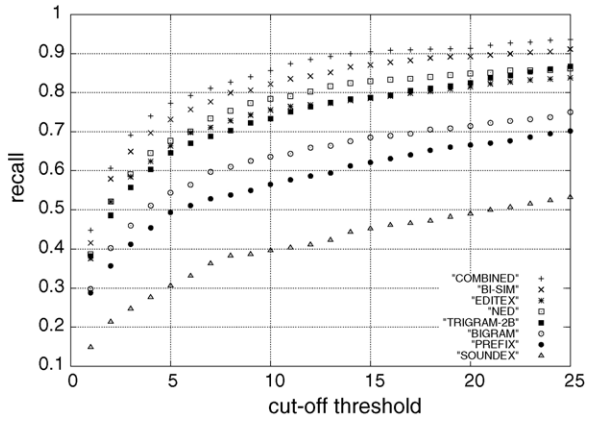

Figure 3 Recall at various thresholds for the USP test set.

Table 5 Recall at $k = 10$ and 20 for both the USP (mixed) and the Sound-alike test sets

	USP set		Sound-alike 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
BI-SIM	0.8220	0.8927	0.4838	0.6590
TRI-SIM	0.8324	0.8946	0.4782	0.6245
ALINE	0.7503	0.8303	0.5825	0.6873
COMBINED	0.8560	0.9137	0.6462	0.7737

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 a list of similar drug names reported to the USP Medication Errors Reporting Program [41] (henceforth the *USP set*). The USP set is a list of 363 confusion sets (both look-alike and sound-alike), which contain 582 unique drug names. Most of the confusion sets are pairs of names, but some contain three or even four names. The maximum number of true positives per test name is six, but for the majority of names (436 out of 582), only one confusable name is identified in the USP set. This is because most names occur in only one of the reported confusion pairs. The average number of true positives among 581 candidates is 1.37.

We computed the similarity of each drug name pair using the following similarity measures: BIGRAM, TRIGRAM-2B, LCSR, EDIT, NED, SOUNDEX, EDITEX, BI-SIM, TRI-SIM, ALINE and PREFIX. In addition, we calculated the COMBINED measure by taking the arithmetic average of the values returned by PREFIX, NED, BI-SIM, and ALINE for each individual pair.[17]

In order to apply ALINE to the USP set, all drug names were transcribed into phonetic symbols. This transcription was approximated by automatic application of a simple set of about 30 regular expression rules. (It is likely that a more sophisticated transcription method would result in improvement of ALINE's performance.) In this 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 Fig. 3, 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. An ideal, oracle-type measure would achieve recall of 0.8572 for $k = 1$, and 1.0 for $k \geq 6$. Table 5 contains detailed results for $k = 10$ and 20 for all measures. The top performer in this experiment was the COMBINED approach, followed by TRI-SIM and BI-SIM.[18]

We performed statistical significance tests for $k = 10$ and 20 using the standard Wilcoxon Signed-rank Test. The difference between TRI-SIM and BI-SIM is not statistically significant, but both algorithms are significantly better than all other individual measures at the 95% confidence level. Several of the pairwise differences between EDIT, NED, LCSR, TRIGRAM-2B, EDITEX, and ALINE are not significant.[19]

[17] The ranges of output values of the individual measures were normalized to fall between 0 and 1 so that all inputs to the combined measure would be comparable for our averaging approach. Although it is possible to adopt alternative approaches (e.g. composing a set of potentially confusable names from the top j names from each measure and choosing the top k overall), we have opted for the averaging approach, which performs well in practice. It is important to note that there are literally hundreds of possible combinations of measures that yield a higher recall than any of the individual measures that make up their combination. We experimented with several of these more complex methods of combining measures, but did not achieve substantially better results.

[18] The variants of BI-SIM and TRI-SIM that do not incorporate the similarity scale between n-grams (effectively distinguishing only between identical and non-identical n-grams) achieve substantially lower accuracy than the variants using the similarity scale.

[19] We refrain from reporting on the statistical significance of the COMBINED approach, because it does not represent an actual similarity measure, but rather the best combination that was selected post-hoc from all possible combinations.

UNCORRECTED PROOF

Prior work on drug name matching [3], with the best known results (prior to the current study), reports the performance of a number of measures, six of which we have tested as well. According to this earlier study, TRIGRAM-2B was the most accurate, NED was a close second, followed by EDIT, EDITEX, BIGRAM, which had about the same accuracy, and SOUNDEX was at the bottom of the pack. Our USP results indicate a different ordering of the six measures: NED is the best, no clear ordering among EDIT, EDITEX, and TRIGRAM-2B, and then BIGRAM followed by SOUNDEX.

We note that the USP set contains both look-alike and sound-alike confusion pairs, which makes it difficult to tease apart the contribution of each metric to the detection of similarity for each individual pair. On the other hand, our investigation reveals that, in practice, it is very difficult to determine whether two confusable names look alike (e.g. due to poor handwriting) or sound alike (e.g. due to common or similar sounds). Although there are cases where look-alike names do not sound alike (e.g. the well-publicized case of confusion between Coumadin and Avandia due to poor handwriting [42]), we have found that the vast majority of look-alike names are also sound-alike names, and vice versa.

To examine this issue further, we conducted a second experiment to compare the performance of various measures on sound-alike pairs only. For this experiment, we used a list of phonetically related drug name pairs (henceforth called, FDA-P) that was produced by the FDA for the purpose of examining potential sound-alike confusions with proposed drug names. We found that, although the FDA-P test set was designed to include mostly phonetic confusions, most of the pairs were also look-alikes.

The FDA-P was created by presenting 40 safety evaluators (trained pharmacists) with a set of proposed names (''consult names'') and asking them to identify potentially confusable names (''names of concern'') from a number of drug name sources. In some cases, the evaluators were also provided with the intended pronunciation of the drug name. From this, we developed a phonetic test set, henceforth referred to as the *Sound-Alike Set*, that consists of 276 ''names of concern'' corresponding to 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 per name is 11, the median is 3, and the average is 3.33.

The measures were applied to calculate the similarity between each of the 83 ''consult'' names and a separate list of 2596 drug names compiled by FDA.

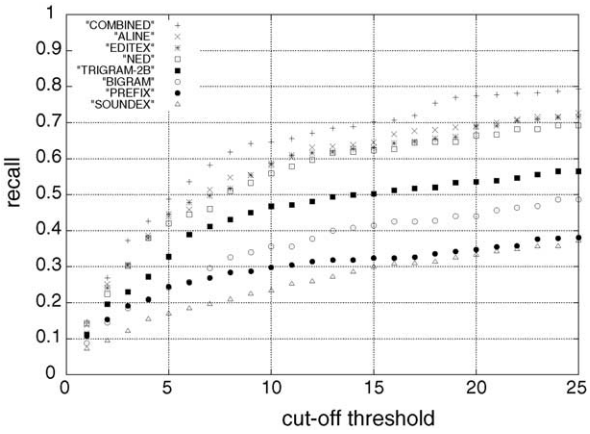

Figure 4 Recall at various thresholds for the sound-alike test set.

Note that the 2596 names constitutes over half the total number of recognized one-word drug names (4400), as reported by [43]; thus the list is of size that is expected in a realistic setting.[20]

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. For optimization, we used simulated annealing with the average recall for cutoffs ranging from 1 to 25 as the objective function.

The results are shown in Fig. 4. Again, some measures have been left out to preserve clarity, but see the *Sound-Alike Set*column of Table 5 for detailed results for all measures. An ideal, oracle-type measure would achieve recall of 0.4471 for $k = 1$, 0.9582 for $k = 5$, and 1.0 for $k \geq 11$. 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 Fig. 3. The COMBINED algorithm is again the top performer, followed by EDITEX and ALINE, which achieve very similar results.

The statistical significance of pairwise differences between measures is weaker in the second

[20] In general, the actual number of names that have to be considered (and the acceptable level of similarity/confusability) will depend on the size of the name database. Our goal is not to suggest such numbers but rather to point out the measures that are likely to be most effective in identifying confusable drug names—and to provide the capability for the end user to define appropriate cut-offs. Our tool is intended to to be an aid to the human; thus, if the user sets a threshold that is too low to exclude non-confusable names, these can weeded out by hand. If the threshold is set too high, then the user is no worse off than they were without the tool, and, given our results, they are probably a fair bit better off.

UNCORRECTED PROOF

experiment, because of a smaller number of test names. In particular, the differences between EDITEX and ALINE and some of the other measures are not always statistically significant. Nevertheless, even though the Wilcoxon Signed-rank Test is unable to confirm some of the differences, the overall relative ranking of the algorithms appears quite consistent.

7. Discussion

The results described in Section 6 clearly indicate that BI-SIM and TRI-SIM, the newly proposed measures of orthographic similarity, outperform several currently used measures on the USP (mixed) test set regardless of the choice of the cutoff parameter *k*. On the sound-alike test set, EDITEX and ALINE are the most effective. However, a simple combination of several measures achieves even higher accuracy, exceeding 90% with only the 15 top pairs considered. It is worth noting that NED does relatively well on both sets in spite of its simplicity.

The USP test set has its limitations. The set includes pairs that are considered confusable for reasons other 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 that were identified by our measures (e.g. *Erythromycin*/*Erythrocin*, *Neosar*/*Neoral*, *Lorazepam*/*Flurazepam*, and *Erex*/*Eurax*/*Urex*) were not identified as such in the USP set.

All similarity measures have their own strengths and weaknesses. The *n*-gram measures are 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 actual number of identical letters is minimal (*Luride*/*Lortab*). ALINE detects phonetic similarity even when it is obscured by the orthography (e.g. *Xanax*/*Zantac*), but phonetic transcription is required beforehand.

The idiosyncrasies of individual measures are attenuated when they are combined, 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.

Table 6 Comparison of ranks assigned by ALINE and NED to several drug names that are confusable with *Banix*

	ALINE	NED
Balmex	1	16
Bidex	2	26
Banflex	3	10
Bumex	4	24
Xanax	7	3
Tenex	8	17
Ranexa	12	12
Videx	24	126
Plavix	39	13
Lasix	44	6
Biaxin	74	347

Our experiments confirm that orthographic approaches are superior to their phonetic counterparts in tasks involving string matching [22] based on recall at various cut-off thresholds. Nevertheless, phonetic approaches identify many sound-alike names that are beyond the reach of orthographic approaches. For example, ALINE is compared to NED in Table 6 for the drug name *Banix*, where ranks are assigned after the names have been automatically converted into phonetic transcription. Note *Bidex* and *Videx* are found in the top 25 by ALINE, but not by NED. On the other hand, there are some names beyond top 25 for ALINE that are picked up by NED (e.g. *Plavix* and *Lasix*).

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 phonetic approaches are EDITEX and ALINE, but whereas ALINE is not geared toward any particular language, EDITEX incorporates English-specific letter groups and rules. Although ALINE requires a language-specific grapheme-to-phoneme program, such programs are stand-alone and available for many languages. On the other hand, it is not clear how the EDITEX algorithm would have to be changed to apply to another language.

8. Conclusion

We have investigated the problem of identifying confusable drug name pairs. The effectiveness of

UNCORRECTED PROOF

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.

It is important to note that, while our evaluation methodology provides a (reusable) framework for testing new algorithms to detect drug-name confusion, we have not yet conducted any user studies to check for actual confusability. Such a study would require an elaborate design that is outside of the scope of this work. For example, one might recruit pharmacists to check off potentially confusable names from a combinatorially-induced set of $\binom{582}{2} = 169,071$ pairs of names from the USP list, and then use the resulting list for evaluating similarity measures. In lieu of this, we have used the USP list itself as our standard. Since the USP list includes drug-names pairs that were reported by pharmacists (and other health-care practitioners) as the cause of at least one error, it is arguably the best standard available. In addition, we have run an evaluation against a list of confusable drug names, produced by human safety evaluators, where the actual cause of confusion is known (i.e. sound-alike drug names). However, user studies are an obvious next step for future research on this problem.

Another possible line of investigation is the isolation of specific aspects of similarity for assessing the contribution of a particular feature to the overall results. Our rigorous overall evaluation shows that our measures outperform other similarity measures; however, we expect that the system would be used in conjunction with techniques that isolate other aspects of similarity and all techniques would be beneficial for human-aided screening.

An additional area of future investigation is that of computing similarity using more sophisticated feature values. In particular, the numerical feature values used in ALINE are based on articulatory phonetics, but may not necessarily be optimal for estimating auditory phonetic similarity. It would be interesting to automatically establish those values from training data with machine-learning techniques.

Finally, 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.

Acknowledgments

The first author's research was supported by Natural Sciences and Engineering Research Council of Canada and the second author's research was supported by the National Science Foundation. In addition, we are indebted to Project Performance Corporation, specifically, Erica Kolatch, Rick Shangraw, and Jessica Toye, for their implementation of our techniques in the POCA system.

References

[1] Lazarou J, Pomeranz B, Corey P. Incidence of adverse drug reactions in hospitalized patients. J Am Med Assoc 1998; 279:1200—5.
[2] Meadows M. Strategies to reduce medication errors. US Food Drug Admin Consum Mag 2003;37(3):21—7.
[3] Lambert BL, Lin S-J, Chang K-Y, Gandhi SK. Similarity as a risk factor in drug-name confusion errors: The look-alike (orthographic) and sound-alike (phonetic) model. Med Care 1999;37(12):1214—25.
[4] Physicians' desk reference for nonprescription drugs and dietary supplements. 24th ed. New York, NY: Thomson PDR; 2003.
[5] Kondrak G. Phonetic alignment and similarity. Comput Human 2003;37(3):273—91.
[6] Salton G. The smart system: experiments in automatic document processing. Englewood Cliffs, NJ: Prentice Hall, 1971.
[7] Goldinger S, Luce P, Pisoni D. Priming lexical neighbours of spoken words: effects of competition and inhibition. J Mem Lang 1989;29:501—18.
[8] Norris D. Word recognition: context effects without priming. Cognition 1987;22:93—136.
[9] Cole R. Listening for mispronunciations: a measure of what we hear during speech. Percept Psychophys 1973;13:153—6.
[10] Harley TA. The psychology of language: from data to theory. Hove, East Sussex, UK: Earlbaum Taylor and Francis, 1995.
[11] Church K. A stochastic parts program and noun phrase parser for unrestricted text. In: Proceedings of the second conference on applied natural language processing, association for computational linguistics; 1988. p. 136—43.
[12] Dunning T, Statistical identification of language. Technical Report CRL MCCS-94—273. Computing Research Lab, New Mexico State University, Las Cruces, NM. March 1994.

UNCORRECTED PROOF

[13] Koehn P, Knight K. Knowledge sources for word-level translation models. In: Proceedings of the 2001 conference on empirical methods in natural language processing, association for computational linguistics; 2001. p. 27–35.
[14] Kondrak G. Identifying cognates by phonetic and semantic similarity. In: Proceedings of the second meeting of the North American chapter of the association for computational linguistics. Association for computational linguistics; 2001. p. 103–10.
[15] Melamed ID. Bitext maps and alignment via pattern recognition. Comp Linguist 1999;25(1):107–30.
[16] Simard M, Foster GF, Isabelle P. Using cognates to align sentences in bilingual corpora. In: Proceedings of the fourth international conference on theoretical and methodological issues in machine translation; 1992. p. 67–81.
[17] Kukich K. Techniques for automatically correcting words in texts. ACM Comp Surv 1992;24(4):377–439.
[18] Mosteller F, Wallace D. Applied bayesian and classical inference: the case of the federalist papers. New York, NY: Springer-Verlag, 1984.
[19] Navarro G, Baeza-Yates R. Very fast and simple approximate string matching. Inform Process Lett 1999;72(1–2): 65–70.
[20] Reynar J. Topic Segmentation: Algorithms and Applications. PhD Thesis. University of Pennsylvania, Philadelphia, PA, 1998.
[21] Suen C. *n*-gram statistics for natural language understanding and text processing. IEEE Trans Pattern Anal Mach Intel 1979;PAMI-1:164–72.
[22] Zobel J, Dart PW. Finding approximate matches in large lexicons. Software Pract Exp 1995;25(3):331–45.
[23] Wagner RA, Fischer MJ. The string-to-string correction problem. J ACM 1974;21(1):168–73.
[24] Hall PAV, Dowling GR. Approximate string matching. Comput Surv 1980;12(4):381–402.
[25] Gadd T. Phonix: the algorithm. Program Autom Libr Inform Syst 1990;24(4):222–37.
[26] Zobel J, Dart P. Phonetic string matching: lessons from information retrieval. In: Proceedings of the 19th international conference on research and development in information retrieval. Association for computing machinery; 1996. p. 166–72.
[27] Lambert BL, Chang K-Y, Lin S-J. Effect of orthographic and phonological similarity on false recognition of drug names. J Soc Sci Med 2001;52:1843–57.
[28] Van Rijsbergen CJ. Information Retrieval, 2nd ed., Glasgow, UK: Department of Computer Science, University of Glasgow, 1999.
[29] Ukkonen E. Approximate string-matching with *q*-grams and maximal matches. Theoret Comput Sci 1992;92:191–211.
[30] Adamson GW, Boreham J. The use of an association measure based on character structure to identify semantically related pairs of words and document titles. Inform Stor Retrieval 1974;10:253–60.
[31] Brew C, McKelvie D. Word-pair extraction for lexicography. In: Oflazer K, Somers H, editors. Proceedings of the second international conference on new methods in language processing. 1996. p. 45–55.
[32] McEnery T, Oakes M. Sentence and word alignment in the CRATER project. In: Thomas J, Short M, editors. Using Corpora for language research. London, UK: Longman; 1996 . p. 211–31.
[33] Connolly JH. Quantifying target-realization differences. Clin Linguist Phonet 1997;11:267–98.
[34] Covington MA. An algorithm to align words for historical comparison. Comp Linguist 1996;22(4):481–96.
[35] Ladefoged P. A course in phonetics. New York, NY: Harcourt Brace Jovanovich, 1975.
[36] Myers EW. Seeing conserved signals. In: Lander ES, Waterman MS, editors. Calculating the secrets of life. Washington, D.C: National Academy Press; 1995. p. 56–89.
[37] Smith TF, Waterman MS. Identification of common molecular sequences. J Mol Biol 1981;147:195–7.
[38] Oommen BJ. String alignment with substitution, insertion, deletion, squashing, and expansion operations. Inform Sci 1995;83:89–107.
[39] Divay M, Vitale AJ. Algorithms for grapheme-phoneme translation for English and French: applications for database searches and speech synthesis. Comp Linguist 1997; 23(4):495–523.
[40] Handbook of the international phonetic association. Cambridge, UK: Cambridge University Press; 1999.
[41] Use caution — avoid confusion. United States Pharmacopeial Convention Quality Review (76). available from http://www.bhhs.org/pdf/qr76.pdf (last accessed: July 18 2005).
[42] ISMP medication safety alert: community/ambulatory care edition, vol. 2: 9 (computer program). Huntingdon Valley, PA, 2003.
[43] Lambert BL, Lin S-J, Tan H. Designing safe drug names. Drug Safety 2005;28(6):495–512.

UNCORRECTED PROOF