Phonemic Similarity Metrics to Compare Pronunciation Methods

Ben Hixon[1], *Eric Schneider*[1], *Susan L. Epstein*[1,2]

[1] Department of Computer Science, Hunter College of The City University of New York
[2] Department of Computer Science, The Graduate Center of The City University of New York
shixon@hunter.cuny.edu, esch@hunter.cuny.edu, susan.epstein@hunter.cuny.edu

Abstract

As grapheme-to-phoneme methods proliferate, their careful evaluation becomes increasingly important. This paper explores a variety of metrics to compare the automatic pronunciation methods of three freely-available grapheme-to-phoneme packages on a large dictionary. Two metrics, presented here for the first time, rely upon a novel weighted phonemic substitution matrix constructed from substitution frequencies in a collection of trusted alternate pronunciations. These new metrics are sensitive to the degree of mutability among phonemes. An alignment tool uses this matrix to compare phoneme substitutions between pairs of pronunciations.

Index Terms: grapheme-to-phoneme, edit distance, substitution matrix, phonetic distance measures

1. Introduction

Grapheme-to-Phoneme (*G2P*) translation is an essential component of both Automatic Speech Recognition (*ASR*) and Text to Speech (*TTS*) synthesis applications. As G2P methods proliferate, it is important to gauge their relative effectiveness. The thesis of this work is that the comparison of pronunciations should quantify the likelihood of different phonemic substitutions. This paper advocates the measurement of phonetic distance with a weighted phonemic substitution matrix (*WPSM*). The WPSM is constructed from the frequency of substitutions that appear in a collection of trusted alternate pronunciations. The principal result of this paper is that such a WPSM supports an intuitively reasonable and effective measure of the similarity between two pronunciations. Metrics based on the WPSM provide incisive comparisons of the accuracy of automated pronunciation tools.

The approach used here is modeled on the way biologists align two protein sequences [1]. Each sequence is represented as a string on an alphabet, where each letter (here, an *entry*) represents a particular amino acid or nucleotide. Two strings are *identical* if and only if they have the same length and their corresponding entries are equal. Otherwise, the quality of an alignment (its *score*) is calculated from the similarity of each pair of corresponding entries, and the number and length of the *gaps* (blank entries) inserted to produce that alignment. The similarity of any pair of entries from the alphabet of amino acids is recorded in a BLOSUM matrix [2].

Analogously, our method represents a pronunciation as a string on an alphabet of phonemes. First, it calculates the WPSM, a BLOSUM-like matrix for pronunciation, from substitution frequencies in a set of trusted alternate pronunciations. Thereafter, it applies the WPSM to align two strings of phonemes (*pronunciations*) with the Needleman-Wunsch algorithm [1]. The alignment score measures the similarity between the two pronunciations in a way that is sensitive to the differences between phonemes.

There are three traditional measurements of G2P pronunciation accuracy with respect to a correct (*reference*) pronunciation: Levenshtein distance, phoneme error rate, and word error rate. The minimum number of insertions, deletions and substitutions required for transformation of one sequence into another is the *Levenshtein distance* [3]. Phoneme error rate (*PER*) is the Levenshtein distance between a predicted pronunciation and the reference pronunciation, divided by the number of phonemes in the reference pronunciation. Word error rate (*WER*) is the proportion of predicted pronunciations with at least one phoneme error to the total number of pronunciations. Neither WER nor PER, however, is a sufficiently sensitive measurement of the distance between pronunciations. Consider, for example, two pronunciation pairs that use the ARPAbet phoneme set [4]:

S OW D AH	S OW D AH
S OW D AA	T AY B L

On the left are two reasonable pronunciations for the English word "soda," while the pair on the right compares a pronunciation for "soda" to one for "table." WER considers these pairs equally distant (100%), while PER detects a difference. In the following two pairs, however, the pair on the right has an unreasonable pronunciation for "soda":

S OW D AH	S OW D AH
S OW D AA	S OW D L

Nonetheless, WER, PER, and Levenshtein distance are the same for these two pairs (100%, 25%, and 1, respectively). The WPSM metrics described here are sensitive enough to overcome these limitations.

The next section of this paper describes related work. Subsequent sections describe the construction of a WPSM, illustrate its application to three freely-available G2P methods, and discuss the results.

2. Related work

ASR and TTS synthesis are core functions of spoken dialogue systems. Both require translation between orthographic and phonetic representations of words. Typically, such translation uses a *phonetic dictionary* that contains a list of words and their associated pronunciations. Even large phonetic dictionaries, however, do not cover all the pronunciations required for real-world tasks that involve very large vocabularies. (Indeed, the work reported here was motivated by a system to support telephoned book orders from a library for visually-impaired patrons [5], where the correct pronunciation of all 28,031 author names was unavailable.) A spoken dialogue system with such a large vocabulary typically uses a phonetic dictionary for a large set of common words, and relies on an automated G2P method to translate out-of-vocabulary words.

Rule-based G2P methods encode natural language pronunciation rules informed by linguistic expertise. Although pronunciation rules for languages such as English are highly complex and contain many exceptions and special cases, some rule-based methods (e.g., Orator [6]) have been successful. Rule-based G2P methods are represented in this experiment by Logios [7], a component of the freely available Olympus spoken dialog system developed at Carnegie Mellon University (*CMU*). Logios itself was based on the MITalk speech synthe-

sis system [8, 9].

Instead of using a priori rules, data-driven G2P methods produce pronunciations with probabilistic models built from a large corpus of training examples. The corpus itself is a phonetic dictionary. The experiment reported here includes two data-driven methods: the decision-tree model of the Festival Speech Synthesis system [10], and Sequitur G2P [11], which is based on joint-sequence models.

Comparison of G2P methods requires some common measure of accuracy. Although G2P accuracy is most commonly measured by PER [11, 12, 13], the weakness of PER is that every difference between a pair of phonemes is treated equally. That may not adequately represent the perceived substitution cost. For example, from the perspective of the user in a spoken dialogue system, a vowel-to-consonant or consonant-to-consonant substitution may be perceived as a more serious error than a vowel-to-vowel substitution, and should therefore have an appropriately higher substitution penalty. Refinements of the measure of phonetic distance and the quantification of substitution penalties have been proposed for applications ranging from speech pathology diagnosis [14] to the construction of linguistic evolutionary trees [14, 15, 16].

An analog to the measurement of edit distance between sequences is a measurement of their similarity. The *similarity score* of two strings is the maximum possible sum of substitution weights for each pair of aligned entries, as given in a substitution matrix, together with gap penalties for each insertion or deletion. Needleman-Wunsch is a dynamic programming algorithm that finds the maximum similarity score of two strings. Needleman-Wunsch iteratively aligns increasingly long string prefixes. For each prefix pair it chooses the maximum score that results when either the last entry in one prefix is substituted for the last entry in the other, or the last character in one string is aligned with a gap.

Applied to pronunciation, the Needleman-Wunsch algorithm requires quantitative phoneme similarity scores, for which various derivation methods have been proposed. One approach labels each phoneme with a set of articulatory features, and makes the substitution cost between two phonemes inversely proportional to the size of the intersection of their feature sets [17]. Another approach assigns numeric values to these features, and computes substitution cost as the distance between feature vectors [14, 16]. Perceptual listening tests have also been used to create a matrix of empirical confusion scores between English phonemes [18], from which substitution costs may be derived [17].

In bioinformatics, sequence alignment is commonly used with matrices containing similarity scores for pairs of amino acids. One of these, the PAM matrix [19], inspired a scoring matrix for grapheme-to-grapheme similarity to identify cognates in written languages [20], but it was not derived from a set of trusted alignments and is for graphemes, not phonemes. In contrast, both the BLOSUM substitution matrices and the work reported here derive their scores from substitution counts observed in a large body of trusted sequence alignments. The next section describes how we derive WPSM phoneme similarity scores from a source of trusted alternate pronunciations, and then apply them to compare pronunciations.

3. Experimental design

CMU's Pronouncing Dictionary v0.7a (here, *CMUDICT*) is an English-pronunciation dictionary widely used in both ASR (e.g., CMU's Sphinx) and TTS (e.g., Festival) applications [21]. Each of its 133,354 plain text entries is a *headword* (an orthographic string) and a *pronunciation*, a string of phonemes drawn from the ARPAbet phonetic alphabet along with stress weights. CMUDICT provides alternate pronunciations for many words. We pre-processed it before this experiment to remove non-alphabetic characters, phonetic stress weights, and acronym expansions. The filtered dictionary (hereafter, *FDICT*) has 129,559 entries. We used FDICT in two ways: as a source of trusted alignments for the WPSM, and as the training corpus for both Festival and Sequitur.

3.1. Construction of the WPSM

Intuitively, an individual WPSM value is the average similarity per phoneme between two alternate pronunciations of a given English word. The WPSM records the frequencies of substitutions in a set of alignments of alternate pronunciations. There are 8513 words in FDICT with two or more pronunciations. We aligned each pair of pronunciations for the same headword with an implementation of Needleman-Wunsch that minimized their Levenshtein distance. This produced 10,159 pairs of alignments. In those alignments, we calculated $p(\alpha)$, the frequency of phoneme α, and $p(\alpha, \beta)$, the frequency with which phoneme β was substituted for phoneme α. Each entry in the WPSM is the log-odds of each such α–β *substitution*, as calculated by:

$$W(\alpha,\beta) = \log \frac{p(\alpha,\beta) + p(\beta,\alpha)}{p(\alpha)p(\beta)} \quad (1)$$

Note that $W(\alpha,\beta) = W(\beta,\alpha)$ for all α and β. Equation (1) produces the value in the αth row and βth column of the WPSM. A positive $W(\alpha,\beta)$ means that the substitution of β for α or α for β is more likely to occur in a string than the independent occurrence of α and β together in the same pronunciation. A negative $W(\alpha,\beta)$ means that the substitution is highly unlikely, that is, a pronunciation is more likely to contain the two phonemes independently than to substitute one for the other.

Figure 1 is an excerpt from the constructed WPSM. The matrix is symmetric; its diagonal entries are the *match scores* of a phoneme with itself, while the non-diagonal entries are the *mismatch scores* due to substitution. The more positive the score, the more often the substitution occurred in the 10,159 alignments. The highest score in each row is the match score (e.g., W(AA,AA) = 2.93), but mismatch scores vary. For example in Figure 1, substitution of B for AA has a far lower score (–0.03) than substitution of AE for AA (1.69). Lower scores incur higher penalties. Figure 1 confirms our earlier intuitions about acceptable phoneme substitutions.

Although analogous to the construction of a BLOSUM matrix, construction of the WPSM warranted several differences appropriate to spoken language. For BLOSUM, no alignment is trusted unless it satisfies an *identity requirement* that mandates some percentage of aligned phonemes be identical. Here, we trusted that all alternates in FDICT reflect daily

	AA	AE	AH	AO	AW	AY	B
AA	2.93	1.69	0.94	2.03	1.56	0.56	-0.03
AE	1.69	2.96	0.84	0.55	-0.94	0.76	0.17
AH	0.94	0.84	2.01	0.65	-0.37	0.85	-0.52
AO	2.03	0.55	0.65	3.64	1.42	0.38	-0.43
AW	1.56	-0.94	-0.37	1.42	4.59	-0.42	-1.54
AY	0.56	0.76	0.85	0.38	-0.42	3.35	-0.78
B	-0.03	0.17	-0.52	-0.43	-1.54	-0.78	3.43

Figure 1: *Upper left corner of the WPSM, calculated from equation (1).*

language. Furthermore, BLOSUM entries are multiplied by a constant λ or rounded to the nearest integer. We also artificially set the frequency of any substitutions with frequency zero to that of the smallest non-zero entry in the entire WPSM, and thereby ensure that (1) is well defined. Finally, for the gap penalty we used the average of all negative mismatch scores.

3.2. Training and testing

We trained Festival and Sequitur with 10-fold cross validation, as follows. First, we randomly partitioned all FDICT headwords into 10 subsets of equal size. All variant pronunciations for the same headword were placed into a single subset. This guaranteed that a headword would never serve as both a training example and a testing example. For each subset *S* (i.e., 10 times), the system was trained on the union of the other 9 subsets and its learned performance evaluated on *S*. We trained Festival using the Festvox 2.1 toolkit [22]. We trained Sequitur to model M-grams up to size 5 [11]. Logios already has its own G2P rule set and thus required no training.

To test all three G2P methods (Festival, Sequitur, and Logios), we stripped the holdout sets of their phonetic pronunciations, so that each test set contained only orthographic headwords. We then used each G2P method to produce a *candidate pronunciation* for each test set example, and compared that candidate with the reference pronunciation recorded in FDICT. We recorded scores for each distinct headword in a test set. If a test headword had multiple pronunciations, we recorded the highest similarity (or lowest distance) scores between a candidate for that headword and any reference pronunciation for it in FDICT.

3.3. Metrics for pronunciation comparison

For each G2P method and each test set, we measured WER, PER, *MLD* (mean Levenshtein distance per pronunciation), *MSS* (mean similarity score per pronunciation), and *MIR* (mean identity ratio per pronunciation). Table 1 provides examples of these measures for two well-known alternate pronunciations of "tomato," and for a reasonable and an egregious pronunciation of "tomato." WER, PER, and MLD for the two pairs are equivalent — from their perspectives, the distance between pronunciation pairs is the same.

In contrast, MSS and MIR both reference the WPSM, and correctly score the similarity for the righthand pair in Table 1 lower. Intuitively, MSS asserts that a single substitution in a long word is less severe than in a short word. MSS is the ratio of the WPSM similarity score between the FDICT and candidate pronunciations to their average length. Finally, an *identity score* compares an FDICT pronunciation to itself; it serves as an upper bound on how similar any pronunciation can be to the FDICT reference pronunciation. MIR is the ratio of the WPSM similarity score between the FDICT and test pronunciations to the identity score of the FDICT pronunciation, expressed as a percentage. Given our assumption that FDICT pronunciations are correct, a good G2P method should have low WER, PER, and MLD, and high MSS and MIR.

MLD, MSS and MIR are calculated from the best Needleman-Wunsch alignment between a candidate pronunciation for a test example and its FDICT pronunciation. To calculate the Levenshtein distances with the Needleman-Wunsch algorithm, we prepared a separate matrix with negative unit scores for substitutions and zero scores for identities, and used the absolute value of the resulting score.

Alignment	T AH M EY T OW T OW M AA T OW	T AH M EY T OW T AH M SH T SH
WER	100%	100%
PER	33.3%	33.3%
MLD	2.00	2.00
MSS	2.32	1.92
MIR	81.30%	69.87%

Table 1: *Sample alignments and associated scores for the headword "tomato."*

4. Results

We applied all five metrics in Section 3.3 to measure the performance of three G2P methods: Festival, Sequitur, and Logios. Table 2 shows the mean values of each performance metric on the 10 holdout sets, along with 95% confidence intervals. Word error rate, phoneme error rate, and average Levenshtein distance per word (MLD) are difference measures — the higher the number, the greater the difference between CMU's FDICT pronunciation and the pronunciation produced by the corresponding G2P method. MSS and MIR are similarity measures based on phoneme weights in the WPSM. A very low score represents a pronunciation with a set of phonemic substitutions unlikely to be made in the English language. The higher the MSS or MIR score, the closer the pronunciation is to a reference pronunciation.

Under every metric applied here, Sequitur had the highest similarity scores and the lowest difference scores. Although the results in Table 2 are remarkably similar to those reported elsewhere for Sequitur and Festival, comparison of WER and PER to those reports may be inappropriate. Earlier work used a different version of CMUDICT, an M-gram size of 9 rather than 5 (used here in the interest of time), and scored multiple reference pronunciations of a single headword differently. To the best of our knowledge, the Logios method has no previously reported WER or PER results.

5. Discussion

This work relies heavily on CMUDICT in three ways. First, we use it as a training corpus for Festival and Sequitur. Ultimately the performance of both G2P methods is highly dependent on its training data. Any errors or inconsistencies in CMUDICT make their way into these methods' predictive models. For example, one CMUDICT pronunciation of "Buenos Aires" is B W EY N AH S EH R. This corrupts the name's ending because it ignores the trailing "es" of its orthographic form. The second way we use CMUDICT is to measure how well the methods' pronunciations conform to CMUDICT's pronunciation on a holdout set, rather than measure their correctness according to the rules of standard American English pronunciation. Not only does this mean that errors in CMUDICT give a false measure of correctness, they also give an unfair advantage to learning methods like Festival and Sequitur, which are trained on a subset of CMUDICT and there-

	WER (%)*	**PER (%)***	**MLD***	**MSS†**	**MIR (%)†**
Festival	40.10 ±0.40	9.06 ±0.11	0.57 ±0.01	2.683 ±0.003	94.22 ±0.09
Logios	51.15 ±0.47	16.45 ±0.16	1.04 ±0.01	2.541 ±0.003	89.39 ±0.10
Sequitur	27.94 ±0.45	6.75 ±0.14	0.43 ±0.01	2.727 ±0.003	95.73 ±0.09

Table 2: *G2P pronunciation comparisons with 95% confidence intervals. * lower is better; † higher is better.*

by learn its idiosyncrasies. In contrast, Logios' rules were designed long before CMUDICT was formulated, and have no prior knowledge of its content.

The final way we use CMUDICT is as a source of pronunciations from which to construct the WPSM. This assumes that included alternate pronunciations are valid and common in daily language. An alternate pronunciation in CMUDICT that is not used in practice introduces inaccuracies in the substitution frequency between alternate phonemes in the pronunciations. For example, CMUDICT contains two pronunciations for the headword "chemicals":

K EH M IH K AH L Z and CH EH M AH K AH L Z

The second pronunciation's leading CH increases the similarity score of a K-CH pairing. Nonetheless, that pronunciation is not used in daily language, and distorts the K-CH phoneme substitution weight to some degree. This particular example has only a slight effect, given the size of the full set of variants, but errors of this type could accumulate.

Our gap penalty for alignment is tailored for pronunciation. In nucleotides, a gap, or an insertion-deletion, may have a severe biological consequence, and possibly deform the translated protein. Biologists therefore assign a high penalty for the insertion of each gap. In speech, however, dropping a syllable is less severe. For the gap penalty here we used the average of all negative mismatch scores: –0.73. This value has intuitive appeal, as the average of all non-conserved mismatches. Moreover, in practice it produced good alignments, and did not exact an overly high penalty.

The three G2P packages examined here are freely available. Recent advances in G2P (noted in Section 2) are predominantly in machine learning. Nonetheless, the traditional use of WER and PER strongly favors those methods over rule-based ones. Table 2, for example, indicates that the Levenshtein distance per word for Logios is more than twice that for Sequitur. Logios uses a hand-tuned set of linguistic rules created by experts, rules that may make more use of similar phonemes, but Levenshtein distance is not sensitive to similarity between phonemes. In contrast, MSS and MIR are calculated from WPSM scores, and suggest that Logios' performance is less weak than it first appears. This matches our intuition that a set of hand-tuned linguistic rules may not perform as badly as the Levenshtein distance suggests, perhaps because of their sensitivity to similar phonemes. Nonetheless, Sequitur after training produces pronunciations that best match previously-unencountered reference pronunciations in CMUDICT.

Our method is general enough to be used with any source of pronunciation variants, such as the Unisyn Lexicon (*UNILEX*) from the University of Edinburgh [23]. UNILEX uses the SAMPA phoneme set. (A mapping between SAMPA and ARPAbet phonemes would be required to use a UNILEX-derived WPSM.) Moreover, recent work in biology has indicated that matrix modifications particular to the proteins of interest produce more appropriate alignments [24]. This suggests that a WPSM developed for a dialect would better support the comparison of pronunciation methods there.

The results presented here suggest that pronunciation with a traditional rule-based method is less error-ridden than WER, PER, and MLD would lead one to believe. Nonetheless, among the three tested automatic pronunciation methods, Sequitur is the best performer. This comparison is trustworthy because it uses metrics that reflect the variation in substitution frequency in practice across a large common vocabulary.

6. References

[1] S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," *Journal of Molecular Biology,* vol. 48, pp. 443-453, 1970.

[2] S. Henikoff and J. G. Henikoff, "Amino acid substitution matrices from protein blocks," *Proceedings of the National Academy of Sciences of the United States of America,* vol. 89, pp. 10915-10919, 1992.

[3] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Soviet Physics Doklady,* vol. 10, pp. 707-710, 1966.

[4] J. E. Shoup, *Phonological aspects of speech recognition*: Prentice-Hall, 1980.

[5] R. J. Passonneau, S. L. Epstein, T. Ligorio, J. Gordon, and P. Bhutada, "Learning about voice search for spoken dialogue systems," presented at the Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Los Angeles, California, 2010.

[6] M. F. Spiegel, "Proper name pronunciations for speech technology applications," *Proceedings of 2002 IEEE Workshop on Speech Synthesis, 2002.,* pp. 175-178, 2003.

[7] Carnegie Mellon University Speech Group (2008, 3/1/2011), *The Logios Tool*.: https://cmusphinx.svn.sourceforge.net/svnroot/cmusphinx/trunk/logios/

[8] J. Allen, S. Hunnicut, and D. H. Klatt, *From Text to Speech: The MITalk System*: Cambridge University Press, 1987.

[9] Personal communication, Alexander Rudnicky.

[10] A. W. Black, P. Taylor, and R. Caley (1998, 3/1/2011), *The Festival Speech Synthesis System.* http://www.cstr.ed.ac.uk/projects/festival/

[11] M. Bisani and H. Ney, "Joint-sequence models for grapheme-to-phoneme conversion," *Speech Communication,* vol. 50, pp. 434-451, 2008.

[12] A. W. Black, K. Lenzo, and V. Pagel, "Issues in Building General Letter to Sound Rules," in *Proceedings of the ESCA Synthesis Workshop*, 1998, pp. 77-80.

[13] S. Chen, "Conditional and joint models for grapheme-to-phoneme conversion," presented at the European Conference on Speech Communication and Technology, 2003.

[14] B. Kessler, "Phonetic comparison algorithms," *Transactions of the Philological Society,* vol. 103, pp. 243-260, 2005.

[15] J. Nerbonne, W. Heeringa, E. V. D. Hout, P. V. D. Kooi, S. Otten, and W. V. D. Vis "Phonetic Distance between Dutch Dialects," ed: Proceedings of CLIN'95, Antwerp, 1996, pp. 185-202.

[16] J. Nerbonne and W. Heeringa, "Measuring Dialect Distance Phonetically," ed: Proceedings of the Third Meeting of the ACL Special Interest Group in Computational Phonology, 1997, pp. 11-18.

[17] M. Pucher, A. Türk, J. Ajmera, and N. Fecher, "Phonetic distance measures for speech recognition vocabulary and grammar optimization," in *3rd Congress of the Alps Adria Acoustics Association*, Graz, Austria, 2007, pp. 2-5.

[18] A. Cutler, A. Weber, R. Smits, and N. Cooper, "Patterns of English phoneme confusions by native and non-native listeners," *Journal of the Acoustical Society of America,* vol. 116, pp. 3668-3678, 2004.

[19] M. O. Dayhoff, R. M. Schwartz, and B. C. Orcutt, "A model of evolutionary change in proteins." vol. 5, M. O. Dayhoff, Ed., ed: National Biomedical Research Foundation, 1978, pp. 345-352.

[20] A. Delmestri and N. Cristianini, "String Similarity Measures and PAM-like Matrices for Cognate Identification," UOB-ISL-TR2010.

[21] R. L. Weide. (1998, 3/1/2011). *The CMU Pronouncing Dictionary*. http://www.speech.cs.cmu.edu/cgi-bin/cmudict

[22] A. W. Black and K. Lenzo. (2000, 3/1/2011). Building voices in the Festival speech synthesis system. http://festvox.org/bsv

[23] S. Fitt, "Documentation and User Guide to UNISYN Lexicon and Post-Lexical Rules," University of Edinburgh, Edinburgh, 2000.

[24] J. E. Coronado, O. Attie, S. L. Epstein, W. G. Qiu, and P. N. Lipke, "Composition-modified matrices improve identification of homologs of saccharomyces cerevisiae low-complexity glycoproteins," *Eukaryotic cell,* vol. 5, pp. 628-37, Apr 2006.