Tools for the Analysis of Benchmark Speech Recognition Tests

S2.16

David S. Pallett, William M. Fisher and Jonathan G. Fiscus

National Institute of Standards and Technology
Gaithersburg, MD 20899

ABSTRACT

This paper reports on the development of tools for the analysis of Benchmark Speech Recognition System Tests. One development is a new tool implementing two statistical significance tests. Another involves studies of an alternative to the alignment process presently used in the DARPA/NIST scoring software (which presently minimizes a weighted sum of elementary word error types) to one that minimizes a measure of phonological implausibility. Our purpose in developing a "standard" implementation of these tools is to make these tools uniformly available to system developers.

INTRODUCTION

Benchmark tests of the performance of speech recognition systems using the DARPA Resource Management Speech Corpora have been implemented at several sites [1]. The National Institute of Standards and Technology (NIST) has developed "standard" scoring software for these tests.

Until recently, little attention has been devoted to the statistical significance of apparent differences in the performance of systems as indicated by differences in test results. Gillick [2] and Gillick and Cox [3] have suggested the use of two simple tests: McNemar's test, and a matched-pairs test.

Picone and Doddington have advocated a phone-mediated alternative to the currently-conventional alignment of reference (REF) and hypothesis (HYP) word strings for the purpose of analyzing word errors, which matches words only as unanalyzed units [4,5]. They make use of a text-to-phone decomposition of two text strings, produce alignments of the phone strings, and then post-process these aligned phone strings to produce aligned word strings. Comparisons of the results of using the existing DARPA/NIST alignment procedure and that of Picone and Doddington have indicated only small differences in the resulting scores, but a substantial improvement in diagnostics because of the generally more plausible word matches [4].

This paper outlines NIST's implementation of: (a) McNemar's Test and the matched-pairs test for the case of benchmark tests of continuous speech recognition systems; and (b) a new recursive string-alignment procedure, based in phonology, which is an alternative to the Picone-Doddington approach.

APPROACH: SIGNIFICANCE TESTS

In our current implementation of McNemar's test, we score errors at the sentence level (i.e., a sentence is either recognized correctly or in error, and the differences that are most important to the McNemar test are derived from comparisons of the number of sentence-level errors that are unique to each system).

In our implementation of the matched-pairs test, we use knowledge from the aligned reference and hypothesized sentence strings (using the present "standard" dynamic programming (DP) string matching algorithm) to locate segments of the hypothesized sentence strings that contain errors. These segments are selected so as to meet the criterion of ensuring that the errors in one segment are statistically independent of the errors in any other segment. In order for a sentence hypothesis to be segmented into two (or more) segments, there must be at least one region of a number of correctly recognized "buffer" words. For the two systems, the matched-pairs test computes the difference in the number of errors in corresponding segments. It then tests the null hypothesis that the mean difference (in the number of word errors per segment) is zero.

Following a suggestion of Gillick and Cox, using the aligned strings, we identify segments where no errors have occurred, and we use these 'good' segments to separate segments where errors have occurred ('bad' segments). The length of the 'good' segments must be sufficiently long to ensure that after a good segment, the first error in a bad segment is independent of any previous errors. In our present implementation, the segments upon which the matched-pairs test is implemented are bounded: (a) on the left by either the beginning of the sentence string or by two (or more) correctly recognized words, and (b) on the right by either two (or more) correctly recognized words or the end of the sentence string. The choice of the number of buffer words in the 'good' segments (in this case two) reflects a compromise between: (a) allowing for a long enough period of time to ensure independence of errors in each segment, and (b) ensuring that the sentence strings are subdivided into a large number of segments. With the number of buffer words set at 2, each sentence is typically segmented into about 1.4 segments, while a shorter buffer length of 1 correctly recognized word yields about 1.9 segments per sentence.

For convenience, we designate this matched-pair sentence-segment word error test as "MAPSSWE".

In our implementation of both tests, a 95% confidence level was used for rejecting the null hypothesis. We use an assumed chi-square distribution with one degree of freedom in implementing the McNemar test, and for the matched-pair sentence-segment word error test, we assume a normal distribution.

In this paper, we use the terms "higher performing" and "higher" to refer to systems with fewer errors than the other system(s) under consideration. Use of these terms is a convention used for convenience throughout this paper, and is not intended to imply a value judgement.

CH2847-2/90/0000-0097 $1.00 © 1990 IEEE

SIGNIFICANCE TEST RESULTS

A number of sites have reported results on the February 1989 DARPA Benchmark Test Set [1]. Additional test results have been made available to NIST, so that benchmark performance data are presently available for a total of six systems processing the Speaker Dependent data, and as many as eight systems processing the Speaker Independent data. The availability of these data has permitted us to exercise the statistical tools for a large number of systems.

The intended emphasis of the analysis that follows is to show the strengths of the statistical tools when comparing different systems, taken a pair at a time.

NOTATION

In Tables 1 and 2, the systems under consideration are designated using the following convention. Sites are designated as "1" through "6". For speaker-independent systems, the use of the "standard" 72 speaker system training condition is designated as "_s", and the "augmented" 109 speaker system training condition is designated as "_a". Results were reported by site 3 for a system with different pruning: this revised system has been designated as system "3_ar". One system differed from its "brother" at site 4 by virtue of the developer having eliminated a software bug in an early implementation: this "debugged" revised system is designated as "4_r". Another of the speaker dependent systems differed from its "brother" at site 2 by the use of triphone smoothing, and this is designated as system "2_t".

SPEAKER DEPENDENT TEST SET RESULTS

For the case of the "no grammar" results for these systems, we make use of both the McNemar sentence-level results and the matched-pair sentence-segment word error (MAPSSWE) results. These comparisons are summarized in Table 1.

The McNemar tests fail to show any significant differences between any pairs of speaker-dependent systems.

In 3 of the 6 pair-wise comparisons of speaker-dependent systems, the MAPSSWE test indicates a significant difference. One of these 3 compares an alternative version of an algorithm with its "control" version at the same site, and the other 2 compare the control algorithms at one site with algorithms at another site. The interesting comparison between the highest performing algorithm at each of two different sites ("4_r" vs. "2_t") indicates no significant difference, in agreement with the McNemar test.

These findings suggest that the differences in performance between these speaker-dependent systems are slight. Alternatively, they suggest that the McNemar (sentence level) tests are not sensitive enough to reveal differences between these state-of-the-art systems with these test sets, but that, in some cases, the MAPSSWE test is able to identify significant differences that are not revealed by the McNemar test.

SPEAKER INDEPENDENT TEST SET RESULTS

For this Test Set, there are results for 8 systems for the case of no grammar. The results of our implementation of the McNemar and MAPSSWE tests are shown in Table 2.

Using the McNemar test, when comparing the 5 systems trained on 109 speakers, there are 10 comparisons that can be made, and 6 of these prove to be significant. For the MAPSSWE test, of the same 10 comparisons, 9 of these prove to be significant.

The substantially larger fraction of all comparisons that turn out to be significant for the data for speaker-independent systems suggests that performance differences between these systems are more likely to be significant than comparisons between the speaker-dependent systems discussed previously. System comparisons that are shown to be significant with the McNemar test were in all cases also shown to be significant using the MAPSSWE test.

It is instructive to consider the strength of the statistical tools when comparing well-performing systems that are comparable in raw scores.

When comparing the higher performing speaker dependent systems at sites "2" and "4", the differences in performance are not shown to be significant for either grammar condition.

For the speaker independent systems with no grammar and using the Speaker Independent test set, when using the McNemar test and comparing system "3_ar" with all others, it is significantly different [higher performing] in 5 out of 7 cases. When using the MAPSSWE test, in 7 out of 7 comparisons the results for system "3_ar" are significantly different [higher performing].

	2_t	2	4_r	4	3_a	4_a
2_t		N: same P: 2_t	N: same P: same	N: same P: same	N: 2_t P: 2_t	N: 2_t P: 2_t
2			N: same P: 4_r	N: same P: 4	N: 2 P: 2	N: 2 P: 2
4_r				N: same P: same	N: 4_r P: 4_r	N: 4_r P: 4_r
4					N: 4 P: 4	N: 4 P: 4
3_a						N: same P: 3_a
4_a						

Table 1. Comparison Matrix showing results of McNemar's Test (denoted "N") and the matched-pair sentence-segment word error test (denoted "P") applied to results for six systems using the February 1989 DARPA Speaker Dependent Benchmark Test Set and "no grammar" condition. The speaker-dependent systems are "2_t", "2", "4_r", and "4". If the test results indicate that the difference is significant, the identity of the higher-performing system is printed in the corresponding box. If the difference in performance is not significant, "same" is printed in the corresponding box.

	3_ar	3_a	4_a	6_a	1_a	4_s	6_s	5_s
3_ar		N: same P: 3_ar	N: same P: 3_ar	N: 3_ar P: 3_ar	N: 3_ar P: 3_ar	N: 3_ar P: 3_ar	N: 3_ar P: 3_ar	N: 3_ar P: 3_ar
3_a			N: same P: 3_a	N: 3_a P: 3_a	N: 3_a P: 3_a	N: 3_a P: 3_a	N: 3_a P: 3_a	N: 3_a P: 3_a
4_a				N: same P: same	N: 4_a P: 4_a	N: 4_a P: 4_a	N: 4_a P: 4_a	N: 4_a P: 4_a
6_a					N: 6_a P: 6_a	N: same P: 6_a	N: same P: 6_a	N: 6_a P: 6_a
1_a						N: 4_s P: 4_s	N: 6_s P: 6_s	N: 1_a P: 1_a
4_s							N: same P: same	N: 4_s P: 4_s
6_s								N: 6_s P: 6_s
5_s								

Table 2. Comparison Matrix showing results of McNemar's Test (denoted "N") and the matched-pair sentence-segment word error test (denoted "P") applied to results for eight speaker-independent systems using the February 1989 DARPA Speaker Independent Benchmark Test Set and "no grammar" condition.

APPROACH: PHONOLOGY-BASED ALIGNMENT OF RECOGNITION RESULTS

In order to report word errors, an alignment of words in REF and HYP strings must be done. The usual approach taken is to find the alignment that minimizes the weighted sum of indicated word substitutions, insertions, and deletions. All insertions and deletions have the same weight, and each substitution weighs slightly less than the sum of an insertion and deletion. An efficient algorithm exists to solve this problem.

Our alignment procedure uses a hierarchy of linguistic code sets. There are both compositional and basic code sets. If a code set is basic, each element consists of only an ASCII representation. If the code set is compositional, then each of its elements also has a list of elements in the next lower code set. For instance, the lexical code set is a compositional one consisting of a set of words, each word having an ASCII representation and a composition in terms of a list of phonemes. Similarly, each member of the phoneme code set has a list of phonological features composing it. The feature code set is basic.

The alignment process is phrased as a calculation of alignment distance. The particular alignment that is found is returned as a side effect.

In doing an alignment, we start with REF and HYP strings of words. These strings are sent to a function ALDIST(s1,s2,code), whose responsibility it is to calculate and return the alignment distance between the two strings. It uses the usual DP algorithm. Whenever the weight (or distance) between elements i and j of the (word) code set, W(i,j), is referred to, instead of looking this distance up in a table, another function WOD(i,j,code) is called. This function is given the job of computing and returning the weight or distance between elements i and j. In order to do this, it calls ALDIST, specifying the next lower code set and the two strings of lower-code (phonemic) elements corresponding to (words) i and j. The process is repeated at the next lower level, until eventually a code is reached that is basic and has no composition in a next-lower code. WOD then ends the recursion by returning a value based only on comparison of the integers i and j (e.g. 0 if i=j, 1 otherwise). In our case, this is at the feature level.

In order to make use of the same logic at the feature level as at other levels, we use a feature representation that is an adaptation of the classical "privative" feature opposition of Trubetzkoy [6]. If a phone has a certain feature, this feature will appear in the phone's string of lower code feature elements, and if it doesn't have that feature, no such symbol will be there. The list of features is strictly ordered, so that interchanging consecutive symbols to find a match is never needed. When WOD works with a feature code set, it returns the value 1 if either i=0 (insertion) or j=0 (deletion); otherwise, if i=j (a match) it returns 0, and if i≠j (substitution), it returns a very large arbitrary number, in order to effectively suppress substitution hypotheses.

As a result, the unit of distance, at every level from phone to utterance, is the number of phonological features that must be changed to turn one into the other.

COMPARISON OF ALTERNATIVE STRING ALIGNMENT PROCEDURES

We have just begun quantitative comparison of the several alignment algorithms under study at present, and only preliminary subjective results are available at present.

We have tested the alternative systems on a file of 300 sentence recognitions that were reported in an earlier DARPA test. Of these, 156 had HYP strings different from REF strings. Table 3 below suggests that all three systems behave differently.

	TI	NIST-NEW
NIST-CUR	39/25%	29/18%
TI		19/12%

Table 3. Number and Percent (of 156) of Alignments Different. NIST-CUR is the current DARPA/NIST scoring alignment system, TI is the Picone-Doddington algorithm, and NIST-NEW is the new DARPA/NIST alignment system.

Table 4 below suggests that both the TI and the NIST-NEW systems may be preferable to the NIST-CUR system, and that there is little to prefer between the TI and the NIST-NEW system.

	TI	NIST-NEW
NIST-CUR	3/21	0/19
TI		2/4

Table 4. Comparison of System Preferences. Number of cases in which system i's alignment was clearly preferred to system j's, over number of cases in which j's was clearly preferred to i's, as judged by WMF.

Table 5 below suggests that the numbers on which gross evaluation results are based are not appreciably different between systems:

	# insertions	# deletions	# substitutions
NIST-CUR	57	74	214
TI	60	77	214
NIST-NEW	58	75	216

Table 5. Numbers of Classes of Word Errors for the Systems.

FURTHER EXTENSIONS

There are several extensions to this approach that may be pursued. Better approximations to current phonological theory are attractive, particularly so as to hypothesize context-sensitive errors. Another attractive extension would address autosegmental phonology, in which sets of features are partitioned into separate tiers or channels that interact only indirectly. The problem of evaluating a multiplicity of solutions to the alignment has not been satisfactorily dealt with. Reporting the number of different minimum cost solutions (as L. Bernstein has done [7]), as well as reporting other diagnostics, may be useful.

SUMMARY

This paper has presented preliminary implementation of two of the statistical tools on DARPA Benchmark Test data. In many cases, comparisons of the performance data for different systems can be shown to be significant using these tests. These results suggest that these statistical tools should be useful to both system developers and others interested in performance assessment.

This paper has also presented a new system, based on recursive phonological calculations, for aligning elements in reference and hypothesis strings in order to report more plausible and helpful diagnostic error messages.

These tools are being developed for incorporation into the DARPA "standard" scoring software for use by DARPA researchers and others involved in performance assessment of speech recognition systems.

ACKNOWLEDGEMENTS

Discussions with Larry Gillick at Dragon Systems have been very helpful in implementing these tests. However, responsibility for the details of the implementation and interpretation of the results is solely that of the authors. These preliminary results are not to be construed as official findings of the National Institute of Standards and Technology or the United States Government, nor are they intended to imply that particular systems are inherently superior to any others in ways theoretical or practical. At NIST, Lynn Cherny contributed to our analysis of feature-based phonological representations.

REFERENCES

[1] Pallett, D. S., "Speech Results on Resource Management Task", in Proceedings of the February 1989 DARPA Speech and Natural Language Workshop (Philadelphia) Morgan Kaufman Publishers, Inc. ISBN 1-55860-073-6, pp. 18-24 (February 1989).

[2] Gillick, L., oral presentation at the October 1987 DARPA Speech Meeting at Cambridge, MA.

[3] Gillick, L. and Cox, S. J., "Some Statistical Issues in the Comparison of Speech Recognition Algorithms", in Proceedings of ICASSP'89 (Glasgow) Paper S10.b.5, pp. 532-535.

[4] Picone, J., Doddington, G. and Pallett, D. S., "Phone-Mediated Word Alignment for Speech Recognition Evaluation", submitted for publication in IEEE Transactions on Acoustics, Speech and Signal Processing.

[5] Picone, J., Goudie-Marshall, K. M., Doddington, G. R., and Fisher, W. M., "Automatic Text Alignment for Speech System Evaluation", IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. ASSP-34, No.4, pp. 1010-1011, August 1986.

[6] Anderson, Stephen R., Phonology in the Twentieth Century, U. of Chicago Press, Chicago, 1985, pp. 99-100.

[7] Bernstein, L. E., "Speechreading Sentences I: Development of a Sequence Comparator", JASA Suppl. 1, Vol. 85, 1989, p. S59(A).