informa
healthcare

Original Article

International Journal of Audiology 2009; 48:80–90

Automatic testing of speech recognition

*Tom Francart**
Marc Moonen[§]
*Jan Wouters**

*ExpORL, Dept. Neurosciences, Katholieke Universiteit Leuven, Belgium
[§]SCD, Dept. Electrical Engineering/ESAT, Katholieke Universiteit Leuven, Heverlee, Belgium

Key Words
Autocorrection
Automatic spelling correction
Error correction
Speech in noise
Speech recognition
Word score
Phoneme score

Abbreviations
Ac: Autocorrection
CVC: Consonant vowel consonant
MO: Manual score of oral response
MOC: Corrected manual score of oral response
MT: Manual score of typed response
SRT: Speech recognition threshold
SNR: Signal to noise ratio

Abstract
Speech reception tests are commonly administered by manually scoring the oral response of the subject. This requires a test supervisor to be continuously present. To avoid this, a subject can type the response, after which it can be scored automatically. However, spelling errors may then be counted as recognition errors, influencing the test results. We demonstrate an autocorrection approach based on two scoring algorithms to cope with spelling errors. The first algorithm deals with sentences and is based on word scores. The second algorithm deals with single words and is based on phoneme scores. Both algorithms were evaluated with a corpus of typed answers based on three different Dutch speech materials. The percentage of differences between automatic and manual scoring was determined, in addition to the mean difference in speech recognition threshold. The sentence correction algorithm performed at a higher accuracy than commonly obtained with these speech materials. The word correction algorithm performed better than the human operator. Both algorithms can be used in practice and allow speech reception tests with open set speech materials over the internet.

Sumario
Las pruebas de recepción del lenguaje se administran comúnmente anotando manualmente las respuestas orales del sujeto. Esto hace necesario de la presencia continua de un supervisor de pruebas. Para evitarlo, un sujeto puede escribir la respuesta, luego de lo cuál ésta puede ser anotada ¸automáticamente. Sin embargo, los errores de ortografÚa pueden ser tomados como errores de reconocimiento, influyendo en los resultados de la prueba. Demostramos un enfoque de auto-corrección con dos algoritmos de anotación para manejar los errores ortográficos. El primer algoritmo se enfoca a oraciones y se basa en puntuaciones de palabras. El segundo algoritmo tiene que ver con palabras solas y esta basado en puntaje de fonemas. Ambos algoritmos fueron evaluados con grupo de respuestas escritas a máquina con base en tres diferentes materiales de lenguaje en holandés. Se determinó el porcentaje de diferencia entre el registro manual y el automático de puntaje, además de la diferencia media en los umbrales de reconocimiento del lenguaje. El algoritmo de corrección de frases funcionó con una exactitud mayor de lo comúnmente obtenido con estos materiales de lenguaje. El algoritmo de corrección de palabras funcionó mejor que el operador humano. Ambos algoritmos pueden ser usados en la práctica y permiten el uso de pruebas de recepción del lenguaje con materiales de lenguaje de contexto abierto a través de Internet.

Both in clinical practice and in research, speech recognition tests are widely used to assess performance in patients under varying conditions. While speech recognition tests in silence or at a fixed noise level are easy to conduct, they require that the test supervisor is continuously present, and scoring is therefore prone to human error. Speech recognition tests using an adaptive procedure (Levitt, 1971) or even more complex procedures are harder to conduct manually because interaction is needed to change the signal to noise ratio after each trial.

Human errors are due to plain scoring mistakes by the supervisor, but also to unclear pronunciation by the subject. The latter can be an issue in hearing-impaired subjects or subjects with a strong dialect.

Both issues can be addressed by using a computer program to automatically conduct the speech test. Subjects enter their response on a computer keyboard and a computer program evaluates the response and selects the next stimulus to be presented. Implementation of such a simple program is straightforward. However, subjects make typing errors, which affect the test results. Therefore, the computer should take into account the possibility of spelling errors and distinguish between such spelling errors and true recognition errors.

Current automatic word correction research can be divided into three broad classes of increasingly difficult problems: (1) isolated word error detection, (2) isolated word error correction, and (3) context-dependent word correction (Kukich, 1992). In the first class, errors are only detected, not corrected, mainly by looking up words or N-grams in a dictionary or frequency table. Presently, this class of problems is mainly solved.

The second class—isolated-word error correction—consists of the generation of correction candidates and the ranking of the candidates. A certain input string has to be compared with many entries in a dictionary and amongst the matches, the best match has to be selected. An overview of practical techniques is given by Navarro (2001) and Kukich (1992).

In the third class, context-dependent word correction, not only every individual word is considered but also the words or even sentences surrounding it. Using different approaches such as language models, the noisy channel model, frequency tables, and large corpora, the algorithm can then suggest a correction.

ISSN 1499-2027 print/ISSN 1708-8186 online
DOI: 10.1080/14992020802400662
© 2009 British Society of Audiology, International Society of Audiology, and Nordic Audiological Society

Received: January 17, 2008
Accepted: July 29, 2008

Tom Francart
ExpORL, Dept. Neurosciences, Katholieke Universiteit Leuven, O & N 2, Herestraat 49 bus 721, B-3000 Leuven, Belgium.
E-mail: tom.francart@med.kuleuven.be

Int J Audiol Downloaded from informahealthcare.com by KU Leuven - Tijdschriften on 01/27/11
For personal use only.

RIGHTSLINK

Reynaert (2005) reviews such algorithms. Research is still going on for this type of problem and while many solutions exist to subsets of this problem, the general problem remains unsolved.

Spelling correctors from word processing software typically detect word errors using a dictionary and then suggest a number of possible corrections. They solve problem (1) and part of problem (2). It is clear that this approach is not sufficient for automatic correction of speech recognition tests, because in this case, the error must not only be detected but also automatically corrected, without interaction of the user. However, in the case of speech recognition tests, the difficult problem of context-dependent automatic correction can be simplified by using extra information that is readily available: the user does not type a random sentence, but is trying to repeat the sentence that was presented.

In this paper we describe two algorithms for autocorrection of sentences and single words respectively, in the context of speech recognition tests. Both algorithms are evaluated using a custom corpus of manually corrected speech recognition tests and are compared to a simple algorithm that does not take spelling errors into account.

The use of automated speech recognition tests has only been reported a few times in the literature, e.g. in Stickney et al (2005, 2004), but autocorrection was never used. However, it has many practical applications, both clinically and in a research environment.

Internet speech recognition tests are currently used for screening large populations for hearing loss. Tests exist for both children[1] and adults[2] and are currently available in Dutch (Smits et al, 2006) and are being developed for Dutch, English, French, German, Polish, and Swedish in the European Hearcom[3] project. All of these tests make use of closed set speech materials. The use of the reported autocorrection algorithms allows internet tests to be administered with open set speech materials.

The main body of this paper comprises the development of a test corpus and the evaluation of two autocorrection algorithms using that corpus. In Appendix 1 the function of the algorithms is described.

Evaluation of the algorithms

To assess the feasibility of the usage of our autocorrection algorithms in practice, a corpus of typed responses to speech recognition tests was developed and used to evaluate the algorithms. The difference in score between the autocorrection algorithm and the human operator is determined and compared to the error introduced by the mistakes of the operator when manually scoring the speech tests.

Development of a test corpus: procedures

To develop a test corpus, a clinical test setup was reproduced. However, in addition to repeating the speech token they heard, the subjects also typed the token on the keyboard of a computer running the APEX program (Francart et al, 2008). The operator then scored the oral response using the standard procedures for each speech material. Two final-year university students of audiology conducted the experiments, and the analyses described in this paper were performed by a third person.

For each speech material, clear rules were established for obtaining the score, corresponding to the rules that were used for the normalization of the speech materials and described in the corresponding papers. All subject responses and manual corrections as described in the next paragraph were combined into a test corpus that can be used to fine-tune or evaluate an autocorrection algorithm.

A corpus entry consists of the following elements:

1. Correct sentence: The sentence as it was presented to the subject, annotated with keywords and split keywords (see Appendix 1).
2. Subject response: The string as it was typed on the computer keyboard by the test subject.
3. Manual score (MO): The score that was given by the audiologist using the *oral* response. This was done by indicating the correctly repeated words on a printed copy of the speech token lists and manually calculating the word score and sentence score on the spot.
4. Corrected manual score (MOC): In a first iteration, typed responses were run through the autocorrection algorithm and every difference in score between the algorithm and the manual score of the oral response was analysed by the operator using the notes made during the experiment. If the operator appeared to have made an error while scoring the oral response, it was corrected.
5. Manual score based on typed response (MT): Every string entered by the subject was manually scored, ignoring spelling errors.

If only the pure autocorrection aspect of the algorithm is evaluated, the MT scores are the most relevant ones. It corresponds to presenting the typed input sentences to a human operator and to the autocorrection algorithm and have both calculate the word score and sentence score.

To assess performance in real-life situations, the MOC scores have to be considered. Here 'the perfect operator' is used as a reference. Differences between the MOC and MT scores are due to differences between the oral response and the typed response.

Finally, differences between the MO and MOC scores correspond to errors made by the operator that will be present in any real experiment. Thus, to assess real-life performance of the algorithm, it is useful to consider the difference between the errors made by the operator and errors made by the algorithm, i.e., the extra errors introduced by the algorithm.

Materials

Three different Dutch speech materials were used to develop three different corpora:

1. NVA words (Wouters et al, 1994): Fifteen lists of 12 consonant-vowel-consonant (CVC) words, uttered by a male speaker.
2. LIST sentences (van Wieringen & Wouters, 2008): Thirty-five lists of 10 sentences, uttered by a female speaker. Each list contains 32 or 33 keywords[4]. A sentence is considered correct if all keywords are repeated correctly and in the right order. Both a keyword score and sentence score are defined.

Int J Audiol Downloaded from informahealthcare.com by KU Leuven - Tijdschriften on 01/27/11
For personal use only.

RIGHTSLINK

3. VU sentences (Versfeld et al, 2000): Thirty-nine lists of 13 sentences, uttered by a male speaker. A sentence is considered correct if all words, not only keywords, are repeated correctly. Usually, only a sentence score is used with the VU sentences.

The NVA words were presented in quiet at three different sound pressure levels (well audible, around 50% performance, and below 50% performance, ranging from 20 dBSPL up to 65 dBSPL).

The LIST and VU sentence materials were masked by four different noise materials: speech shaped noise, a competing speaker in Dutch, a competing speaker in Swedish, and the ICRA5-250 speech shaped noise modulated with a speech envelope (Wagener et al, 2006). The operator was instructed to measure at at least three signal to noise ratios (SNR) for each condition with the purpose of determining the speech reception threshold (SRT) by fitting a psychometric curve through these points afterwards. The number of measured SNRs per condition varied between three and five, and the used SNR varied between −20 dB and 10 dB. The SNRs used for each subject were recorded.

As the sentence algorithm is based on keywords, we marked keywords for the VU sentences and used these for the algorithm. They were marked according to the same rules that were used for the LIST sentence material (van Wieringen & Wouters, 2008). In simplified form, this means that all words are keywords except pronouns, adpositions, auxiliary verbs, and articles. This condition is labeled (Word).

In clinical practice however, the VU sentences are scored differently: a sentence is only counted as correct if all words, not only keywords, are repeated correctly. Therefore, we also performed autocorrection with all words in the gold standard sentence as keywords instead of only the keywords that were marked. This condition is labeled (Sent).

Subjects

To obtain a diverse corpus, 20 young students of the university of Leuven were recruited, (group 1), as well as 13 subjects, aged 50 years old on average, who reported having problems with spelling and computer use (group 2).

Evaluation

We evaluated both autocorrection (Ac) algorithms using our different corpora. First, we measured the number of false positives and negatives. Second, we assessed the influence of autocorrection on the obtained SRT, the value that is traditionally derived from speech recognition tests.

For comparison, we also performed autocorrection on our corpora using a simple algorithm that counts the number of keywords that are exactly the same in the input sentence and in the correct sentence, and that occur in the same order. This algorithm is labeled Simple. While it has a very high false negative rate, the results give an impression of the number of spelling mistakes that were made in each condition.

To evaluate of the number of errors made by the human operators, the percentages of modifications between the MO and MOC conditions were calculated.

Percent correct

We calculated the difference between the manual scores (for word score and sentence score) and the automatically generated scores. The difference is given as percentage errors of the autocorrector, with the manual score as a reference. As there are different manual scores (cf. the section headed 'Development of a test corpus: procedures', above), several scores are given for each condition in Table 1.

For sentences, results are given as word score and sentence score. The figures for word score (Word) reflect the number of keywords that were scored incorrectly by the autocorrector per total number of keywords. The sentence score (Sent) is based on the number of correctly scored keywords. For words, results are given for phoneme score and word score. The word score is based on the phoneme score. Here, the phoneme score is the most realistic indicator, as in practice phoneme scores are commonly used.

Influence on srt

The SRT is commonly determined by fitting a two-parameter logistic function through the percent correct values found at different SNRs recorded during the tests. We assessed the influence of the autocorrection algorithm on the estimated SRT by calculating the difference between the SRT determined by fitting the percent correct values obtained by manual scoring (MOC) and obtained by the autocorrection algorithm (Ac).

There were always three or more data points (SNR values) per condition (speech material/noise type/subject). The average difference in SRT between manual and automatic scoring for each speech material is given in the last column of Table 1. As the accuracy of the SRT determined by this method is usually not better than ±1 dB (van Wieringen & Wouters, 2008; Versfeld et al, 2000), our algorithm will have no significant impact on a single estimated SRT value if the difference remains below this value.

Results

Table 1 shows percentages of errors of the autocorrection algorithm versus the different manually scored entries in the corpus. In the first column the different speech materials are given. For the VU speech material, the text ('keywords') or ('all words') indicates whether all words of the sentence were used as keywords or only the words marked as keywords. For each speech material results are given for both groups of subjects, group 1 are the 'good' spellers and group 2 the 'bad' spellers. In the second column, the number of tokens in our corpus is given and in the third column the percentage of errors made by the simple algorithm. The next eight columns give percentages of errors per corpus entry type as described in the section 'Development of test corpus: procedures'. For each corpus entry type, two parts are given: the results with word scoring (Word) and the results with sentence scoring (Sent). Similarly, for the NVA words the results are given for phoneme scoring (Phon) and for word scoring (Word). The last column of the table gives the mean difference in speech reception threshold (SRT), calculated on the Ac and MOC results.

In what follows, we will first describe some observations on the number of tokens per test material and group, then we will

Int J Audiol Downloaded from informahealthcare.com by KU Leuven - Tijdschriften on 01/27/11
For personal use only.

RIGHTSLINK

Int J Audiol Downloaded from informahealthcare.com by KU Leuven - Tijdschriften on 01/27/11
For personal use only.

Table 1. Percentage of errors made by the autocorrection algorithm compared to manual scoring methods for each speech material, group of subjects, number of tokens in the corpus, and corpus entry type. For the sentence materials, errors for keyword score (Word) and for sentence score (Sent) are given. For the CVC materials, errors for phoneme score and for word score are given. # is the total number of sentences presented for the sentence tests and the total number of words presented for the CVC test. MO-MOC is the percentage of changes between the MO and MOC scores. ΔSRT is the mean of the differences in estimated SRT (in dB) between Ac and MOC for each condition. MO is the original manual score based on the oral response, MOC is the corrected manual score based on the oral response, MT is the manual score based on the typed response and Ac is the score by the autocorrection algorithm.

Test material	*Group*	#	*Simple*	*MO-Ac*		*MOC-Ac*		*MT-Ac*		*MO-MOC*		*ΔSRT (dB)*
				Word	*Sent*	*Word*	*Sent*	*Word*	*Sent*	*Word*	*Sent*	
LIST	1	3280	12.7	1.5	1.3	0.5	0.6	0.4	0.4	1.0	0.7	0.06
LIST	2	1310	26.6	2.3	3.7	1.1	2.2	0.2	0.3	1.3	1.5	0.23
VU (keywords)	1	3354	6.0	1.5	2.1	0.5	0.7	0.3	0.2	1.0	0.6	0.55
VU (keywords)	2	1742	16.5	1.7	3.0	1.0	1.9	0.4	0.4	0.9	1.0	0.18
VU (all words)	1	3354	6.0		3.9		3.3		2.9	1.0	0.6	0.66
VU (all words)	2	1742	16.5		6.3		5.4		3.9	0.9	1.0	0.61
				Phon	Word	Phon	Word	Phon	Word	Phon	Word	
NVA	1	756	32.8	1.3	4.0	0.3	0.9	0.1	0.3	1.0	3.0	
NVA	2	600	38.0	3.4	8.8	1.2	3.5	0.2	0.5	2.3	5.7	

compare the results between both groups of subjects. Thereafter, we will compare the columns labeled MO-Ac, MOC-Ac, MT-Ac, and MO-MOC with each other, and finally we will analyse the differences between rows, i.e. between the different test materials and between the (keywords) and (all words) conditions for the VU sentences.

First, considering the number of tokens presented, 3280 (group 1) or 1310 (group 2) LIST sentences correspond to 328 or 131 lists of 10 sentences. This means that overall each of the 35 lists of sentences was presented at least three times to each group of subjects and often more. For the VU sentences, similarly, 258 or 134 lists of 13 sentences were presented, corresponding to at least three presentations to each group of subjects. Similarly, for the NVA words 63 and 50 lists of 12 words were presented. This means that each of the 15 NVA word lists was presented at least three times to each group of subjects.

Comparison of the results of group 1 and 2, the 'good' and the 'bad' spellers, shows that the simple algorithm (column 4) made many more errors with the data of group 2. The results from the simple algorithm are, of course, the same for the VU (keywords) and VU (all words) conditions, but are shown twice for clarity. Comparison of autocorrection performance between the two groups (columns MO-Ac, MOC-Ac, and MT-Ac), shows that slightly more errors were made with the data of group 2, on average 0.5% difference in word score errors for the LIST sentences and 0.3% for the VU sentences.

In the following paragraphs, we will first compare the percentages of errors between sentence scores (Sent) and word scores (Word), and then compare the results for the different corpus entry types. We will compare the MOC-Ac and MT-Ac scores, followed by the MO-Ac and MOC-Ac scores and then consider the MO-MOC scores. All comparisons will be done per column, i.e. for all test materials and both groups simultaneously.

For the LIST and VU sentence tests, the percentages of errors for the sentence scores (Sent) tend to be somewhat larger than those for the word scores (Word). This is due to the fact that any word of a sentence that was scored incorrectly leads to an error in the score of the entire sentence, while in the case of word scoring, it only leads to an error for one of the words of the sentence. For the NVA words, the same is true for phoneme scores versus word scores.

The difference between the MOC-Ac scores and MT-Ac scores (columns 7–8 and 9–10) is related to the difference between the typed response and the oral response. It gives an indication of how difficult it was for the subjects to combine the typing task with the speech perception task. The average difference between the MOC-Ac scores and the MT-Ac scores is 0.5%.

The differences between the MO and MOC scores correspond to errors introduced by manually scoring the speech tests, either by misunderstanding the oral response or by miscalculating the resulting score. Comparison of columns 5–6 (MO-Ac) and 7–8 (MOC-Ac) shows that both the word scores and sentence scores improve, on average, by 1.0% less errors of the autocorrection algorithm.

The MO-MOC column indicates the number of errors made by the human operator. The average human error for word scoring of sentences (LIST and VU) is 1.0% and for sentence scoring it is 0.9%. Comparison of these values to the values in

RIGHTSLINK

column MOC-Ac and column MT-Ac, shows that the average number of errors made by the autocorrection algorithm is smaller. For word scoring (Word), the differences between MO-MOC and MOC-Ac/MT-Ac were significant ($p < 0.01$, paired t-tests) for the LIST sentences in both groups, for the VU sentences in group 1, and for phoneme score of the NVA words in both groups.

Now we will consider differences between the rows of the table. Comparison of the autocorrection performance between the LIST and VU sentences reveals no significant difference using a paired t-test for any of both groups of subjects. However, comparison of the scores for the VU sentences with sentence and keyword scoring respectively, shows that the algorithm performs significantly ($p < 0.01$) better with keyword scoring than with sentence scoring for the VU sentences. The reason is that a human operator tends to ignore or just mishears small errors in words that are irrelevant for the meaning of the sentence. For sentence scoring with this speech material, every word is considered a keyword and thus influences the sentence score.

The ΔSRT values in the last column show the mean difference in SRT found from the psychometric function when using the MOC and Ac scores. While all ΔSRT values differ significantly from each other, both between groups and between speech materials, the absolute differences are very small and there is no clear tendency of change. Note that for each condition only three or four SNRs were measured and that ΔSRT will decrease if more SNRs are measured per condition. For example for group 1 the percentage of errors for the LIST sentences is 0.6% (MOC-Ac). In the case of three SNRs measured per condition, the average number of errors per condition is $10 \times 3 \times 0.006 = 0.18$. This means that in most cases the SRT will not be influenced at all, but if there is an error present in any of the sentences of this condition, it may have a large influence on the SRT because the psychometric curve (with three parameters) is fit through only three data points.

Discussion

The within-subjects standard deviation on the SRT determined using an adaptive procedure is 1.17 dB for the LIST sentences in noise (van Wieringen & Wouters, 2008) and 1.07 dB for the VU sentences in noise (Versfeld et al, 2000). The error introduced by using the autocorrection algorithm is an order of magnitude smaller, and will therefore not influence the result of a single SRT measurement.

In order to assess real-life performance of the autocorrection algorithms, the MOC scores should be used as comparison, because these compare the algorithms to a well-established standard, i.e. manual scoring of oral responses. When only percent correct scores are considered (no SRT calculation), very small errors are obtained, in most cases even below the expected accuracy of testing. Moreover, the number of errors made by the autocorrection algorithm is similar to or smaller than the number of errors made by the operator, so the results will not be influenced more by the use of autocorrection than by the errors made by a human operator, or by the possible bias of the human operator.

The simple non-autocorrecting algorithm (Simple) should not be used in practice, especially when the subjects are expected to have problems with spelling. Comparison of the results of the simple algorithm between group 1 and group 2 reveals that the subjects of group 2 indeed made many more spelling errors. Comparison of the results of the autocorrection algorithms between groups shows that the results are slightly worse for group 2, but still within the expected accuracy of speech recognition tests. It should, however, be noted that the subjects in group 2 were selected based on their self-reported problems with spelling and computer use. Therefore, the results of group 2 should be observed as worst case results. In normal circumstances, these subjects would probably not be tested using an automated setup because they need a lot of encouragement and repeated instructions. Nevertheless, the percentages of errors of the autocorrection algorithm are still in the same range as the percentages of errors made by a human operator. The algorithm itself copes very well with this difficult task.

Comparison of the MOC and MT scores shows that there is a small difference between the operator's assessment of the oral response and the typed response. It is, however, never clear what the intended answer is: did the subject intend to answer what he said or what he typed?

The word correction algorithm performs better than the human operator. This is probably due to, on the one hand, unclear articulation of the subjects, and, on the other hand, the difficulty of the task: the experimenter has to remain very concentrated during the repetitive task and has to decide within a few seconds which phonemes were repeated correctly. When the answer is approximately correct, there is a chance of positive bias and when the answer is incorrect, it is not always straightforward to identify a single correctly identified phoneme using the strict rules.

Moreover, data from the VU sentences indicate that, while the percentages of errors for sentence scoring are acceptable, the algorithm is best used with keyword scoring. As a human experimenter tends to ignore small errors in words that do not contribute to the meaning of the sentence—even if the response is incorrect according to the strict rules—keyword scoring using the autocorrection algorithm is most similar to this situation.

Both algorithms were developed for the Dutch language. Applicability to other languages depends on the correspondence between phonemes and graphemes of the language. While in Dutch this correspondence is rather strict, this is not necessarily the case in other languages (e.g. English). In any case, we expect the autocorrection algorithms to perform very well in, amongst others, Danish, Finnish, French, German, Italian, Polish, Spanish, Swedish, and Turkish because in these languages the correspondence between phonemes and graphemes is strong. In order to convert the sentence algorithm to another language, the only blocks that have to be changed are the language and speech material specific rules, and of course the list of keywords of the speech material. In order to convert the word algorithm to another language, only the list of phonemes and phoneme codes has to be changed.

Conclusions and Applications

The autocorrection algorithms for both sentence tests and word tests are very well suited for use in practice and will not introduce more errors than a human operator.

In a clinical setting, the use of automated speech recognition tests may be rather limited because the subjects require clear

Int J Audiol Downloaded from informahealthcare.com by KU Leuven - Tijdschriften on 01/27/11
For personal use only.

RIGHTSLINK

instructions anyway and some subjects may not be able to efficiently use a computer keyboard. However, automated speech recognition tests can be very useful in many other areas, including research, screening large groups of patients, and remote tests (e.g., over the internet).

When a test subject does not articulate clearly it can be very difficult to score single words manually, especially when testing hearing impaired subjects. In this case automatic scoring using our autocorrection algorithm should be preferred over manual scoring.

Acknowledgements

This research was partly sponsored by Cochlear Ltd. and by the IWT (Institute for the Promotion of Innovation by Science and Technology in Flanders), project 050445. We thank our test subjects for their patient and enthusiastic participation to the speech tests. We also thank Inne Vanmoer and Evelyne Bekaert for their help in gathering data for the corpus.

Notes

[1] A Dutch hearing screening test for children is available on http://www.kinderhoortest.nl/

[2] Dutch hearing screening tests for adults are available on http://www.hoortest.nl/ and http://www.oorcheck.nl/

[3] More information on the Hearcom internet hearing screening tests can be found on http://www.hearcom.eu/main/Checkingyourhearing/speechtesttext.html

[4] Keywords are the words that are important to understand the meaning of the sentence.

References

Cormen T., Leiserson C., Rivest R.L. & Stein C. 2001. *Introduction to Algorithms*, 2nd edition. MIT Press & McGraw-Hill.

Francart, T., van Wieringen, A. & Wouters, J. 2008. APEX 3: A multi-purpose test platform for auditory psychophysical experiments. *J Neurosci Methods*, 172, 283–293.

Friedl J.F. 2006. *Mastering Regular Expressions*, 3rd edition. O'Reilly.

Kukich, K. 1992. Technique for automatically correcting words in text. *ACM Computing Surveys*, 24, 377–439.

Levenshtein V. 1965. Binary codes capable of correcting spurious insertions and deletions or ones. *Problems of Information Transmission*, 8–17.

Levitt H. 1971. Transformed up-down methods in psychoacoustics. *J Acoust Soc Am*, 49 Suppl 2:467+.

Navarro, G. 2001. A guided tour to approximate string matching. *ACM Computing Surveys*, 33, 31–88.

Phillips L. 2000. The double metaphone search algorithm. *C/C++ Users Journal*.

Reynaert M. 2004. Multilingual text induced spelling correction. In: *Proceedings of the 20th International Conference on Computational Linguistics* (COLING 2004).

Reynaert M. 2005. Text-induced spelling correction. Ph.D. thesis, University of Tilburg.

Smits, C., Merkus, P. & Houtgast, T. 2006. How we do it: The Dutch functional hearing-screening tests by telephone and internet. *Clin Otolaryngol*, 31, 436–40.

Stickney, G., Nie, K. & Zeng, F. 2005. Contribution of frequency modulation to speech recognition in noise. *J Acoust Soc Am*, 118, 2412–20.

Stickney, G., Zeng, F., Litovsky, R. & Assmann, P. 2004. Cochlear implant speech recognition with speech maskers. *J Acoust Soc Am*, 116, 1081–91.

van Wieringen, A. & Wouters, J. 2008. LIST and LINT: sentences and numbers for quantifying speech understanding in severely impaired listeners for Flanders and The Netherlands. *Int J Audiol*, 47, 348–355.

Versfeld, N., Daalder, L., Festen, J. & Houtgast, T. 2000. Method for the selection of sentence materials for efficient measurement of the speech reception threshold. *J Acoust Soc Am*, 107, 1671–84.

Wagener, K., Brand, T. & Kollmeier, B. 2006. The role of silent intervals for sentence intelligibility in fluctuating noise in hearing-impaired listeners. *Int J Audiol*, 45, 26–33.

Wouters J., Damman W. & Bosman A. 1994. Vlaamse opname van woordenlijsten voor spraakaudiometrie. *Logopedie*, 28–33.

Int J Audiol Downloaded from informahealthcare.com by KU Leuven - Tijdschriften on 01/27/11
For personal use only.

RIGHTSLINK

Appendix 1: Description of the algorithms

Two algorithms are described in this appendix, one for correcting words based on a phoneme score and one for correcting sentences based on a word score. The scoring rules that were used are given in appendix 2. In a word test, a word is considered correct if all phonemes are correct. In a sentence test, a sentence is considered correct if all keywords are correct. Keywords are the words that are important to get the meaning of the sentence (thus excluding articles, etc.). Both for manual and automatic scoring, this method requires a list of keywords per sentence. If keywords are defined, both a keyword score and a sentence score can be determined per sentence.

Our algorithm works based on keywords and thus calculates the sentence score based on the keyword score. If no keywords are defined for a certain speech material, it considers all words as keywords and thus considers a sentence correct only if all keywords are correct. The same method is normally used when manually scoring speech recognition tests.

The speech tests that were used to evaluate the algorithms have been normalized using the same scoring rules as implemented in the algorithms.

1. The sentence algorithm

General

We consider the case where a subject hears a sentence and then has to type this sentence on the computer keyboard. In what follows, the *user input* is the sentence that the test subject types on the computer keyboard, i.e. the sentence to be corrected. The *gold standard* is the sentence that was presented to the subject.

The algorithm processes two input strings: the user input and the gold standard. A sentence consists of words separated by white space. For each word of the gold standard it is manually indicated whether it is a keyword or not, and whether it is part of a split keyword. Split keywords are keywords that consist of two separate words, but only count as one word when it comes to word score. In English, an example would be 'The man wrapped up the package', where 'wrapped up' would count as one keyword.

Figure 1 shows the general structure of the algorithm. In what follows, we briefly describe the different blocks:

Input normalization: The punctuation characters ,;.: are replaced by spaces; all remaining non-alphanumeric characters are removed; all diacritics are removed (e.g. ä becomes a, è becomes e); all letters are converted to lower case (e.g. cApiTAL becomes capital); and multiple sequential white space characters are simplified to a single white space character.

Split into words: The sentence is split into words using the space character as a delimiter. Possible spacing errors are not corrected in this step.

Space correction: Extra spaces in the middle of a word or missing spaces are corrected using the algorithm described in the section 'Spacing correction'.

Dictionary check: Each word is checked against a dictionary and the results of this check are stored in memory. For our tests, we used the freely available Dutch OpenTaal dictionary (http://opentaal.org/).

Number to text: 'Words' that consist of only numbers or of a series of numbers followed by a suffix are converted to text using a language specific number to text algorithm. We used a custom algorithm that was manually verified for all numbers from 0 to 10020. Larger numbers did not occur in the speech materials that were used in the evaluation. The subjects were encouraged anyway to use the numeric form when typing numbers, as this reduces the number of spelling errors. The current block converts the numbers in both the input and gold sentence to text, allowing them to be processed by other blocks as if they were normal text.

List specific rules: Some language and speech-material specific rules in the form of regular expressions (Friedl, 2006) are applied. If for example the sentence material contains words that can officially be spelled in different ways, one way is selected as the default and the other possibilities are converted to the default. In this stage also some very common spelling mistakes for a language can be corrected. The rules that were used for correction of the Dutch LIST (van Wieringen & Wouters, 2008) and VU (Versfeld et al, 2000) sentence test materials are given in Table 2. These rules are applied to both the user input and the gold standard. Note that the result does not necessarily correspond to the 'correct spelling' any more. Therefore the dictionary check is performed on the data before this transformation.

Bigram correction: The sentence, the results from the dictionary correction, and the gold standard are sent to the bigram correction algorithm for the actual spelling correction, as described in the section 'Bigram correction', below.

Word and sentence scores: The word score and sentence score are calculated, as described in the section 'Word score determination', below.

Int J Audiol Downloaded from informahealthcare.com by KU Leuven - Tijdschriften on 01/27/11
For personal use only.

RIGHTSLINK

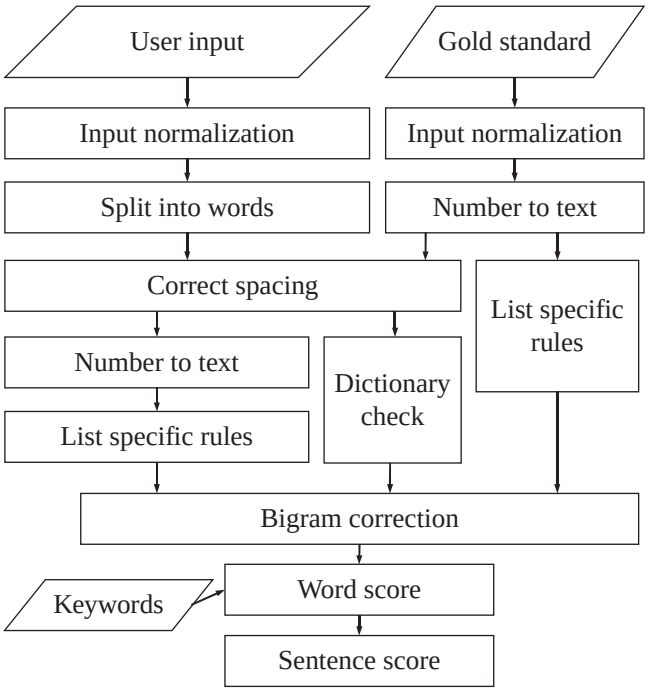

Figure 1. Flowchart of the sentence algorithm. An arrow signifies that the output from the source block is used as the input for the target block.

Int J Audiol Downloaded from informahealthcare.com by KU Leuven - Tijdschriften on 01/27/11
For personal use only.

Spacing correction

The input to the space correction algorithm is a typed sentence and the corresponding gold sentence, split into words by using the space character as a delimiter. The algorithm then operates on all unigrams and bigrams that can be formed using these words. A bigram is a combination of any two sequential words in a string. If, for example, the string is 'The quick fox jumps', then the bigrams are 'The quick', 'quick fox' and 'fox jumps'. Similarly, a single word can be called a unigram. The output of the spacing correction algorithm is a sentence, which is again split into words because the spacing may have changed. The space correction algorithm operates in three steps.

1. A list of unigrams and bigrams is generated from the input sentence and from the gold standard sentence.
2. A check is made to determine whether each input uni/bigram occurs in the list of gold uni/bigrams. If it does not, an approximate string matching technique (Reynaert, 2004, 2005) is used to find an approximate match.
3. If an approximate match is found, it is determined if and where spaces should be inserted. The process is illustrated in Figure 2. First all spaces are removed from the user input bigram. Then it is aligned to the gold standard bigram using a dynamic programming method (Cormen et al, 2001). If the number of corresponding letters is larger than 90%, spaces are inserted in the corresponding places in the user input bigram.

Bigram correction

The bigram correction algorithm takes as input the result from the space correction algorithm that is again split into words using the space character as a delimiter. It operates similarly to the space correction algorithm, with the difference that words are only considered for correction if they are not in the dictionary.

The result is a corrected list of words in the input sentence that is then sent to the word score determination block.

Table 2. Description of regular expressions used for the Dutch LIST and VU sentence test materials.

1. Replace cadeau by kado
2. Replace bureau by buro
3. Replace eigenaresse by eigenares
4. Replace any number of d and t at the end of a word by a single t
5. Replace ei by ij

RIGHTSLINK

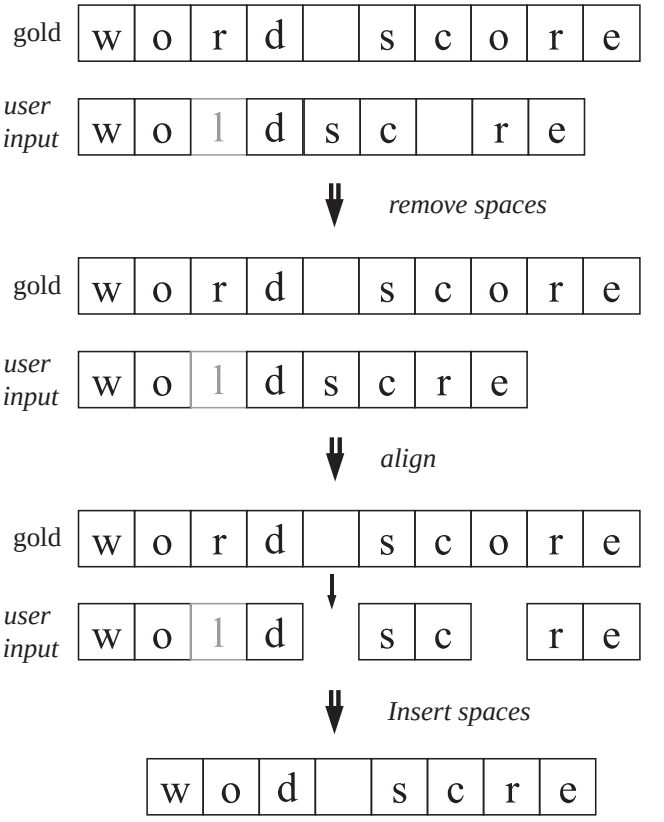

Figure 2. Example of string alignment. Spaces are marked by empty boxes. In this case the gold string is 'word score' and the user input string 'woldsc re'. First all spaces are removed from the input string. Then both strings are aligned. The space character marked with the single arrow could be inserted into the input string as shown. However, as the percentage of correctly aligned characters ($100 \times 7/10 = 70\%$) is smaller than 90%, no space will be inserted because the strings are not considered sufficiently alike in this case.

Int J Audiol Downloaded from informahealthcare.com by KU Leuven - Tijdschriften on 01/27/11
For personal use only.

Word score determination

The word score is calculated by comparing the result from the bigram correction to the gold standard (after the transformations previously described). The score is the number of corrected keywords in the user input that correspond to gold keywords. The corresponding words must occur in the same order in both strings.

To decide whether two words are the same, the following rules are followed:

- If the user input and gold word are numeric, they must match exactly.
- If the double metaphone (Phillips, 2000) representation of the user input and gold word differ, they are considered different. The double metaphone algorithm was built to facilitate phonetic comparisons across languages.
- If the Levenshtein distance (Levenshtein, 1965) between the user input and gold word is larger than 1, they are considered different. The Levenshtein distance is also called the edit distance and is the sum of the minimum number of insertions, deletions, and transpositions necessary to transform one string into the other.

Example of the sentence algorithm

We illustrate the function of the entire algorithm by means of an example.

Let the user input be: 'The**boy fel from** the **windaw**' and the correct answer:

'The boy fell from the window'

We will use the words in bold as keywords. The user input is transformed by the different functional blocks as follows:

Input normalization: theboy fel from the windaw

Correct spacing: the boy fel from the windaw

Dictionary check: The words fel and windaw are not in the dictionary and can thus be corrected.

Bigram correction: the boy fell from the window.

Word score: The gold standard and the corrected input sentence are exactly equal, the word score algorithm yields a keyword score of 4/4 and a corresponding sentence score of 1.

RIGHTSLINK

2. The word algorithm

General

Speech recognition tests can also be done with single words. Typically words with a well defined structure, such as consonant-vowel-consonant (CVC) are used and scores are given based on the number of phonemes identified correctly. In the following sections, an algorithm is described for automated scoring of word-tests.

The organization of the word correction algorithm is illustrated in Figure 3, where the main steps are:

Input normalization: The input is transformed to lower case and diacritics and non-alphanumeric characters are removed.

Number to text: If the input consists of only digits, the number is converted to text (using the same number-to-text algorithm as used in the sentence correction algorithm, see above: 'Number to text' in the section 'The sentence algorithm').

Conversion into graphemes: The input is converted into a series of grapheme codes (see 'Conversion into graphemes', below)

Compare graphemes: The user input and gold standard grapheme codes are compared (see 'Compare graphemes', below), resulting in a phoneme score, from which the word score can be derived.

Conversion into graphemes

This module operates on both the user input word and the gold standard word. It makes use of a language-specific list of graphemes. A grapheme is the set of units of a writing system (as letters and letter combinations) that represent a phoneme. Every grapheme corresponds to a numeric grapheme code. Graphemes that correspond to the same phoneme receive the same code. The list that is currently used for Dutch is given in Table 3. Some graphemes correspond to the same phoneme only if they occur at the end of a word and not if they occur in the middle of a word. Therefore, if a g or d occurs at the end of the word, it is converted to the code of ch or t. The algorithm looks for the longest possible grapheme in the string. For example boot would be converted into [2 40 19] and not into [2 14 14 19].

c. Compare graphemes

The phoneme score is calculated by comparing the two arrays of grapheme codes. First, graphemes that do not occur in the user input grapheme list are removed from the gold grapheme list, and graphemes that do not occur in the gold grapheme list are removed from the user input grapheme list. Then the score is calculated as the number of corresponding graphemes for the best alignment of both arrays. The best alignment is defined as the alignment that yields the highest score.

Example of the word algorithm

As an example, the word kieuw is presented to the test subject. If the typed word (user input) is kiew, the autocorrection proceeds as follows:

Grapheme conversion: kieuw is converted to [10 31 37] and kiew is converted to [10 31 37]

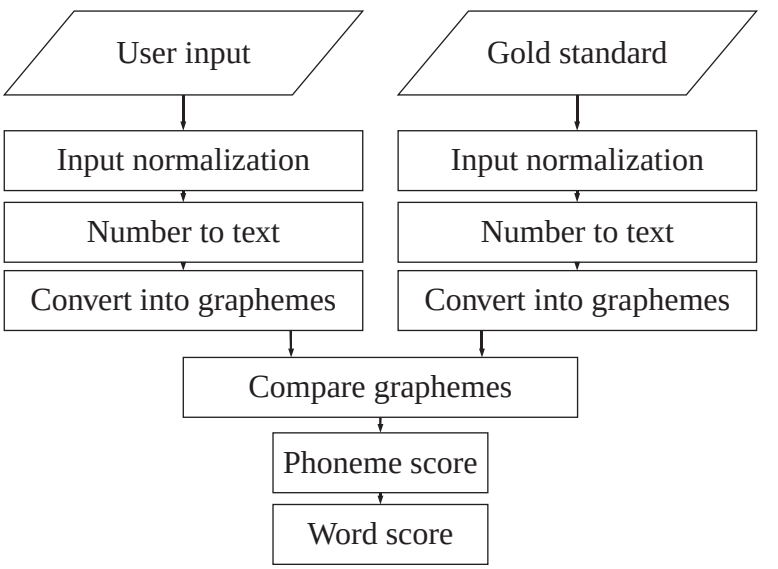

Figure 3. General structure of the word correction algorithm

Int J Audiol Downloaded from informahealthcare.com by KU Leuven - Tijdschriften on 01/27/11
For personal use only.

RIGHTSLINK

Table 3. Graphemes used for correction of Dutch CVC words. Graphemes with the same code are between square brackets and codes are given as subscripts.

[a]$_1$ [b]$_2$ [c]$_3$ [d]$_4$ [e]$_5$ [f]$_6$ [h]$_7$ [i]$_8$ [j]$_9$ [k]$_{10}$ [l]$_{11}$ [m]$_{12}$ [n]$_{13}$ [o]$_{14}$ [p]$_{15}$ [q]$_{16}$ [r]$_{17}$ [s]$_{18}$ [t]$_{19}$ [u]$_{20}$ [v]$_{21}$ [x]$_{22}$ [y]$_{23}$ [z]$_{24}$ [ch]$_{25}$ [g]$_{26}$ [oe]$_{27}$ [ui]$_{28}$ [aa]$_{29}$ [ee]$_{30}$ [ie]$_{31}$ [uu]$_{32}$ [ng]$_{33}$ [ij ei]$_{35}$ [uw w]$_{37}$ [ou au]$_{39}$ [oa oo]$_{40}$

Grapheme comparison: Correlation of the five different alignment positions of both arrays yields [0 0 3 0 0], thus the score becomes 3.

As a second example, the word dijk is presented to the test subject. If the typed word (user input) is bij, the autocorrection proceeds as follows:

Grapheme conversion: dijk is converted to [4 35 10] and bij is converted to [2 35]

Grapheme comparison: As the graphemes number 4, 10, and 2 only occur in one of both arrays, they are removed. The resulting arrays are [35] and [35]. Cross correlation of both arrays yields [1], thus the score becomes 1.

Appendix 2: Scoring rules

CVC tests

1. Every phoneme that is repeated correctly results in 1 point
2. A phoneme must be exactly correct, even if the difference is small
3. The phonemes must be repeated in the right order
4. Extra phonemes before or after the correctly repeated phonemes have no influence on the score

Sentence tests

1. Every keyword that is repeated correctly results in 1 point.
2. A keyword must be exactly correct, e.g. if the plural form is given when the singular form was expected, the word is considered incorrect.
3. Both parts of verbs that can be split must be repeated correctly for the verb to be scored as correct.

Int J Audiol Downloaded from informahealthcare.com by KU Leuven - Tijdschriften on 01/27/11
For personal use only.

RIGHTSLINK