Developing a Reading Tutor: Design and Evaluation of Dedicated Speech Recognition and Synthesis Modules
Jacques Duchateau, Yuk On Kong, Leen Cleuren, Lukas Latacz, Jan Roelens, Abdurrahman Samir, Kris Demuynck, Pol Ghesquière, Werner Verhelst, Hugo Van Hamme

To cite this version:

HAL Id: hal-00558520
https://hal.science/hal-00558520
Submitted on 22 Jan 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Developing a Reading Tutor: Design and Evaluation of Dedicated Speech Recognition and Synthesis Modules

Jacques Duchateau, Yuk On Kong, Leen Cleuren, Lukas Latacz, Jan Roelens, Abdurrahman Samir, Kris Demuynck, Pol Ghesquière, Werner Verhelst, Hugo Van hamme

PII: S0167-6393(09)00069-7
DOI: 10.1016/j.specom.2009.04.010
Reference: SPECOM 1804

To appear in: Speech Communication

Received Date: 29 May 2008
Revised Date: 17 April 2009
Accepted Date: 20 April 2009


This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.
Developing a Reading Tutor: Design and Evaluation of Dedicated Speech Recognition and Synthesis Modules

Jacques Duchateau\textsuperscript{a,\,*}, Yuk On Kong\textsuperscript{b}, Leen Cleuren\textsuperscript{c}, Lukas Latacz\textsuperscript{b}, Jan Roelens\textsuperscript{a}, Abdurrahman Samir\textsuperscript{a}, Kris Demuynck\textsuperscript{a}, Pol Ghesquière\textsuperscript{c}, Werner Verhelst\textsuperscript{b}, Hugo Van hamme\textsuperscript{a}

\textsuperscript{a}ESAT department, Katholieke Universiteit Leuven, Kasteelpark Arenberg 10 PO 2441, B-3001 Heverlee (Leuven), Belgium
\textsuperscript{b}ETRO department, Vrije Universiteit Brussel, Pleinlaan 2, B-1050 Brussel, Belgium
\textsuperscript{c}Centre for Parenting, Child Welfare and Disabilities, Katholieke Universiteit Leuven, Vesaliusstraat 2 PO 3765, B-3000 Leuven, Belgium

Abstract

When a child learns to read, the learning process can be enhanced by significant reading practice with individual support from a tutor. But in reality, the availability of teachers or clinicians is limited, so the additional use of a fully automated reading tutor would be beneficial for the child. This paper discusses our efforts to develop an automated reading tutor for Dutch. First, the dedicated speech recognition and synthesis modules in the reading tutor are described. Then, three diagnostic and remedial reading tutor tools are evaluated in practice and improved based on these evaluations: (1) automatic assessment of a child’s reading level, (2) oral feedback to a child at the phoneme, syllable or word level, and (3) tracking where a child is reading, for automated screen advancement or for direct feedback to the child. In general, the presented tools work in a satisfactory way, including for children with known reading disabilities.

Key words: reading tutor, computer-assisted language learning (CALL).

* Corresponding author. Tel: +32 16 321861, Fax: +32 16 321723.
Email address: Jacques.Duchateau@esat.kuleuven.be (Jacques Duchateau).
1 Introduction

For a child with reading difficulties, daily reading practice and individualised support is necessary. However, due to practical limitations (e.g. in Flemish regular elementary schools only one teacher is in charge of a group of twenty to thirty students) and federal law restrictions (e.g. only a limited amount of individual reading therapy is covered by the national health care system), it is often difficult to provide the child with sufficient practice and support. Therefore, a fully automated reading tutor, able to track the child’s reading progress and to accurately detect reading errors on the one hand, and able to act as a fluent reading model (to read aloud for the child, read along with the child etc.) and to give adequate personalised feedback on the other hand, would be beneficial.

This paper discusses our efforts to develop an automated reading tutor for Dutch. The development of such a reading tutor is one of the aims of the SPACE project’s educational part (SPACE is the acronym for SPEech Algorithms for Clinical and Educational applications). The idea is to improve speech technology so that both the speech recogniser and the speech synthesiser can accomplish the reading tutor tasks described above. More information on the SPACE project can be found on its website.

In the literature, many systems for CALL (computer-assisted language learning) have been described that incorporate language and speech technology. However, it can be concluded from the proceedings of recent workshops on the topic (e.g. the 2007 SLaTE workshop on Speech and Language Technology in Education), that a clear majority of these systems focus on second language learning adult users. Some examples of these systems are described in Probst et al. (2002); Neri et al. (2006); Abdou et al. (2006) and D’Mello et al. (2007).

Literature on CALL systems for children, including reading tutors, is more scarce, especially papers on systems that have been evaluated in practice on children in general, and on children with reading difficulties in particular. One important reason for this is that the development of CALL systems for children, especially the speech recognition and synthesis modules in it, is a very challenging task due to reading-related and child-related development processes. For example, the awareness of young children of phonetic processes (or linguistic processes in general) is rather limited. Also, due to a child’s variable maturation, articulatory competencies differ and due to variable word decoding skills, oral reading of inexperienced readers or readers with reading disabilities can be fraught with oral reading errors. In the past decade, several

---

projects on the development of CALL systems were started, as described e.g. in Hagen et al. (2007); Banerjee et al. (2003); Black et al. (2007); Eskenazi and Pelton (2002); Russell et al. (2000) and Adams (2006).

This paper is organised as follows. Section 2 gives a description of the synthesis (section 2.1) and recognition (section 2.2) modules. Section 3 presents three reading tutor tools: successively the automatic reading level assessment (section 3.1), the oral feedback by synthesis (section 3.2), and the reading tracker (section 3.3) are discussed and evaluated. Finally, in section 4 our conclusions and directions for future work are given.

2 Description of the synthesis and recognition systems

This section discusses the dedicated speech synthesis and recognition modules which are integrated into our reading tutor. The main modules of the reading tutor concern:

- Management of the reading tasks, to define each task and the possible use of synthesis and recognition in it.
- Management of the student, to define which student should do what tasks.
- Management of the recordings, including diagnostic tools for the supervisor.
- Making of the recordings, which allows the student to do the tasks.

The reading tutor was mainly used (1) to record the Chorec database (see section 3.1), which consists of recordings of reading tasks in which the synthesis and recognition modules were not used, and (2) to evaluate the use of the synthesis and recognition modules in remedial reading tasks. The remainder of this section details both modules.

2.1 Speech synthesis system

2.1.1 The synthesiser

Corpus-based concatenative speech synthesis (e.g. Hunt and Black, 1996; Chu et al., 2003; Clark et al., 2007) is the mainstream way to synthesise speech. In such synthesisers, a large speech database is first segmented into small units. To synthesise an input text, the best combination of speech units is selected from the database to match the utterance, based on the sum of weighted cost functions. The selected unit sequence is then concatenated to generate the synthesis. In the SPACE project, the database of the speech synthesiser contains recordings of an early-middle-aged, female, native Flemish speaker.
She has a pleasant voice, appropriate for children, who are the target population of the reading tutor. If necessary, new voices can be created later by recording other speakers, as the synthesiser is data-driven. Alternatively, new voices can be created through voice conversion techniques. Our current speech database contains about 3 hours of recordings. Besides a full Dutch diphone set (which serves as the last resort), the recordings mainly consist of graded Dutch story material, as this suits the application. The database has been segmented using the forced aligner described in Demuynck et al. (2004). A unit selection synthesiser has two parts: a language-dependent front-end providing text analysis, and a language-independent back-end providing unit selection. In our case, text analysis is performed by an adapted version of the front-end of the Dutch diphone synthesiser, NeXTeNS (Kerkhoff and Marsi, 2002).

Three main changes were implemented:

- The lexicon is adapted in order to match Flemish pronunciation.
- The post-lexical rules are not used, so as to match the labeling of the speech database.
- Besides silences between phrases, silences between words within a phrase can also be predicted. Training is performed on the segmentation and labeling of the speech database, instead of manually labeling the utterances. Silences are predicted using memory-based machine learning techniques (Daelemans and Van den Bosch, 2005). Features used by the memory-based learner include part of speech, word identity, and punctuation before and after words.

Actually, our synthesis system is a combination of four different synthesisers under the same framework: (a) a diphone synthesiser, (b) a unit selection synthesiser using uniform units (diphones or monophones), (c) a hierarchical synthesiser, and (d) a novel non-uniform unit synthesiser primarily looking for the longest possible chunks. The front-end is shared by all synthesisers. With the last two synthesisers, we aimed at obtaining higher quality than state-of-the-art, and at capturing the required natural prosody existing in the speech corpus by maximising the mean unit length in syntheses. Both words and full utterances can be synthesised with our synthesis system. The hierarchical synthesiser has been implemented with three levels: phrase, word and diphone. For each input utterance, it first looks for units at the phrase level. If it fails for some part of an utterance or the whole of it, it will back off to the next level and search again for the missing parts. As all Dutch diphones are in the inventory, each utterance can be completed. For the fourth and novel selection algorithm, long non-uniform units are selected from the speech database (Latacz et al., 2007). In brief, these non-uniform units could represent any sequence of diphones so that units as long as syllables, words and beyond could be the target. The best unit sequence is selected according to target and join costs, which can be based on symbolic and/or acoustic features. The join costs measure how smooth the transition between candidate
units is. Differences in spectrum, pitch and energy are taken into account. The target costs determine how well a unit matches its symbolic target description. The target is described in terms of linguistic features like syllable structure, lexical stress, and phonemic and syllabic contexts. Synthesising full utterances is more difficult than synthesising single words because phrase breaks could occur between words and the prosody should be more complex and variable. Prosody is not predicted explicitly but is obtained through symbolic features. In addition to the features mentioned above, symbolic features such as syllable accent, part of speech, the position in the utterance and the phrase, and the number of syllables to the next and the previous accent, stress, and phrase breaks are also used for synthesising utterances.

As the synthesiser serves partly to model and demonstrate for the child, we strive to achieve high intelligibility. Firstly, we use a speech database containing the same type of material as required for the application, namely stories for children, because the quality of the synthesised speech of a unit selection synthesiser is known to be better this way than if generic material is used (Black and Lenzo, 2001). By doing so, the synthesiser is optimised for the application and the target population. Secondly, we use a novel algorithm, hierarchical synthesis, for the synthesiser. As it searches for longer chunks for synthesis, this should raise intelligibility as well. Thirdly, in order to make it easier for the child to understand synthesised speech and to make sure that the child can read along with synthesised speech, the synthesis speech rate must not be too high. The synthesis speech rate can be adjusted online and, if needed, short pauses can be inserted between consecutive words, in order to achieve an even slower speech rate but with lower chance of creating artifacts. The length of these pauses can also be adjusted. For illustration purposes, some audio samples of synthesised words are available on our website.

2.1.2 Phoneme-by-phoneme and syllable-by-syllable synthesis modes

Usually, existing speech synthesisers provide a speaking style corresponding to fluently read text. In contrast, speech therapists or teachers use different speaking styles when interacting with their patients. Additional speaking styles or reading modes increase the effectiveness of the reading tutor, as demonstrated in Heiner et al. (2004). Both phoneme-by-phoneme and syllable-by-syllable speech are needed for modelling and for giving feedback. Two types of syllabified speech are considered, with isolated or lengthened connected syllables. The latter is needed for assisting more advanced readers to build up their fluency and reading speed.

---

Phoneme-by-phoneme mode.

In most current speech synthesizers, orthographic spelling mode, or letter-by-letter spelling, is implemented. A phoneme-by-phoneme mode which allows stressing/emphasis is, however, more useful for training people with reading disabilities. In this mode, each single word is pronounced phoneme by phoneme. If necessary, a phoneme can be pronounced with stress or emphasis to attract the attention of the child to this particular phoneme. Note that stress here does not mean lexical stress.

An overview of the phoneme-by-phoneme mode synthesis is given in figure 1. Phonemes of the same stress level were recorded with similar pitches, speech rates and loudness levels to obtain natural-sounding synthesised speech. There are two different stress levels to choose from, as we recorded each Dutch phoneme from the speaker in two versions: with stress and without stress. No signal processing is involved.

The input text is processed into a list of words. Words are converted into a stream of phonemes using a lexicon and grapheme-to-phoneme conversion. The appropriate phonemes with their associated stress-levels are then selected from a speech inventory and concatenated with a silence in between. Utterances can then be spelled phoneme by phoneme. As expected, the quality of the synthesised speech of this mode is found to be very high. The domain could be extended in order to synthesise feedback like not /h a t/ but /h a d/, which is straightforward to implement.

Syllable-by-syllable mode.

The purpose of this reading mode is to synthesise speech as either isolated or lengthened connected syllables. To synthesise speech in isolated syllables,
silences are inserted between neighbouring syllables by analysing the input text. The output synthesis is then syllabified. As for synthesising lengthened connected syllables, pre-synthesised speech is stretched (Latacz et al., 2006).

2.2 Speech recognition system

The task of the speech recognition system in a reading tutor seems to be easy as the words or sentences that should be read, are known. But the child will not read everything correctly, and it is difficult to predict what the child will say exactly: he/she may even give comments that are not related to the reading task. Therefore, a recogniser set-up was adopted that is very similar to a recognition system which we use for large-vocabulary continuous speech. This section describes the acoustic modelling, the search engine(s) and the language models involved in the system.

The acoustic modelling is based on a 22-hour read speech database in Dutch, which is different from the databases we used for the experiments below in section 3. It contains recordings of continuous sentences read or spoken fluently by children aged between 5 and 11 years. Cross-word context-dependent acoustic models were estimated with 1400 tied Hidden Markov Model (HMM) states and 16000 tied Gaussian distributions in total. A straightforward signal processing scheme based on the Mel-spectrum was adhered to, including cepstral mean subtraction, discriminant analysis and Vocal Tract Length Normalisation (VTLN). As we are interested in tracking where a child is reading, we developed a VTLN system that does not introduce latency in the pre-processing. In the estimation of the warping factor in the VTLN system, latency (or the use of an initial recording for the estimation) could be avoided by using an adaptive warping factor estimation. This way, the only latency in our pre-processing is due to taking first and second order time derivatives, which results in a latency of 3 frames (30 msec).

For the decoding of the speech, a recognition system architecture with two layers was adopted, as depicted in figure 2. This architecture can also be used for other recognition tasks. It was, for instance, applied successfully in large-vocabulary continuous speech recognition. This recognition system is discussed in detail in Demuynck et al. (2006).

In the first layer, a task-independent phoneme recogniser generates a phoneme lattice. The phoneme recogniser is based on the acoustic modelling detailed above, and on a trigram phoneme sequence model estimated from a Dutch database with correctly read sentences.

In the second layer, the task-dependent information is modelled. As the sentence that should be read is known in a reading tutor, we opted for a finite-
Currently, we use this architecture with 2 layers for the detection of reading errors only. In order to provide low-latency feedback when tracking where a child is reading, a set-up with one layer was preferred to the system with two layers presented above. It is possible to implement feedback for tracking in the system with two layers, but the intermediate representation with phoneme arcs in the lattice will produce additional latency. The set-up with one layer is based on the same search engine and the same acoustic models as the first layer in the set-up with two layers. The difference is in the language model: the FST is used instead of the trigram phoneme sequence model. Every 150 msec, the search engine infers the word in the sentence with the highest probability (for being spoken) at that point in time from the different hypotheses in the search, and sends this information to the reading tutor. This feedback frequency (once every 150 msec) seemed to be fine in practice. It can be decreased, but a high frequency (e.g. once per frame) will slow down the recognition system noticeably. It should be noted that in the reading tutor, there is additional,
not negligible latency due to data transfer (audio, feedback) between modules, screen generation, handling and plotting (the screens with the words to be read), and even the refresh rate of the computer screen.

3 Evaluation of the diagnostic and remedial tools

In this section, we discuss three diagnostic and remedial tools: (1) the automatic assessment of a child’s reading level, (2) the use of synthesis for oral feedback, and (3) the tracking of the child’s reading for automated screen advancement and for visual feedback.

3.1 Automatic assessment of children’s reading level

Early identification of children with reading disabilities in primary school is a major concern as their overall academic development depends heavily on it. Automatic reading level assessment may help in this task as a form of screening, so that speech therapists can have time to provide both a more detailed assessment and adequate intervention for children with (according to the automatic assessment) a low reading level.

3.1.1 Experimental set-up

In Duchateau et al. (2007), we proposed a baseline automatic assessment system. This assessment is based on a reading test with 40 isolated words. The score that expresses the reading performance of a child is defined as the total time needed to read the 40 words divided by the number of correctly read words. Since timing is known from the recording, we only need to know, for every word, whether it was read correctly, to deduce a score for an individual. Therefore, a human score can be determined from a manual annotation, while an automatic score is based on a speech recognition system. Based on these scores, each of the children in a particular grade is classified into one of 5 performance groups. For more background on this scoring and classification method, used in schools in Flanders, the reader is referred to Duchateau et al. (2007).

3 In the Netherlands and Flanders, CITO (Central Institute for Test Development) introduced the use of 5 performance groups of unequal sizes: best performing 25%, above average performing 25%, below average performing 25%, far below average performing 15%, and worst performing 10%.
The experiments are based on the Chorec database (described in Cleuren et al., 2008), which contains reading sessions (of both isolated words and stories) from 400 children, including children with known reading disabilities. We selected the 3 real word reading tasks for the experiments: one task with 40 1-syllable words, one with 40 2-syllable words, and one with 40 3- and 4-syllable words. Previous experiments (see Duchateau et al., 2007) showed that the use of tasks with (non-existent) pseudowords resulted in worse agreements for classification. Each real word reading task was read once by each of the children, however some recordings are lacking, for instance because the task was too difficult for that child. For grades 2, 3 and 4, the Chorec database contains about 75 recordings per real word reading task. For grade 1, this number is 55, 32 and 9 only for the three tasks respectively. Therefore, we will not report classification results for the 3- and 4-syllable word task for grade 1.

To automatically decide if a word was read correctly, we used the recognition system with 2 layers described in section 2.2. Since, by definition, a word is read correctly when the child reads it correctly at his/her final attempt, we can easily tag a word as read correctly or not by inspecting the end of the best path through the sentence FST (see figure 2), which is the result of the recogniser. If the best path ends in the word FST, the word is tagged as read correctly, if it ends in the garbage FST, the word is tagged as read incorrectly.

3.1.2 Experimental results

Improvements over the baseline.

In the baseline, a word recognised as being read correctly contributes one in the score calculation. But the recogniser is not always correct, so we improved on that by contributing only the probability that the word is really read correctly given that it was recognised as correct. This task-dependent probability is estimated from the Chorec data. Similarly, for words recognised as read wrongly, we contribute the probability for a false alarm given that the word is recognised as wrong (rather than contributing zero as in the baseline). In the result tables, we call this system general probabilities.

Furthermore, we found that these probabilities depend on the width of the phoneme lattice that is generated by the first pass of the recogniser (more precisely, we use the average phoneme lattice width over the final attempt by the child for that word). This dependency is shown in figure 3 for the 1-syllable task read by the grade 3 children. This is not surprising as this information source is also used in typical confidence measures in large-vocabulary recognisers, either explicitly as in Duchateau et al. (2002) or implicitly e.g. by using posterior probabilities in lattices. In practice, we divided the range of the phoneme lattice width into 7 bins and estimated task-dependent (not word-dependent) probabilities for each bin. We call this system lattice-width-
Fig. 3. Distribution of words annotated in Chorec as wrong (solid line) and correct (dashed line) as a function of the lattice width. The figure to the left is for words tagged as wrong by the recogniser, the one to the right is for words tagged as correct.

**dependent probabilities.** Note that in order to calculate the score of a particular child, all of the above probabilities are estimated on the data from the other children only, to avoid a bias.

The results of the reading level assessment experiments are given in table 1. For each grade, for the three tasks and for the three classifiers, the agreement (linearly weighted Cohen’s Kappa 4) with the reference human classification is given, and also the percentage of correctly classified children. We can see that the agreements are typically over 0.8 (known as *almost perfect*). Moreover, wrongly classified children can be put into a neighbouring performance group in all cases. Given the results in the table, we can conclude that there is a significant improvement from the baseline to the systems with probabilities. The systems with probabilities perform equally on average. The value of the added information in the system with lattice-width-dependent probabilities seems to be too small to improve the classification.

**Human-human vs. human-machine agreements.**

One third of the Chorec database is annotated a second and third time, by different annotators (the aim being the assessment of inter-annotator agreements). Annotators may disagree if a word has been read correctly because some decisions are disputable and because annotations are never error-free. Therefore, based on the additional annotations, other human classifications can be made. Comparing these with the (in fact arbitrarily chosen) reference human classification results in human-human agreements on the classification task.

In table 2, the average agreement (and the percentage correct) between the 2 new human classifiers and the reference are compared with the agreements between the automatic classifiers and the reference. As classification of the

4 See e.g. [http://faculty.vassar.edu/lowry/kappaexp.html](http://faculty.vassar.edu/lowry/kappaexp.html)
<table>
<thead>
<tr>
<th>Grade 1</th>
<th>1-syllable</th>
<th>2-syllable</th>
<th>3+4-syll.</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline</td>
<td>0.79/73%</td>
<td>0.72/63%</td>
<td>- / -</td>
</tr>
<tr>
<td>general probs.</td>
<td>0.81/76%</td>
<td>0.77/69%</td>
<td>- / -</td>
</tr>
<tr>
<td>lattice-width-dep.</td>
<td>0.84/80%</td>
<td>0.70/63%</td>
<td>- / -</td>
</tr>
<tr>
<td>Grade 2</td>
<td>1-syllable</td>
<td>2-syllable</td>
<td>3+4-syll.</td>
</tr>
<tr>
<td>baseline</td>
<td>0.82/74%</td>
<td>0.90/85%</td>
<td>0.85/77%</td>
</tr>
<tr>
<td>general probs.</td>
<td>0.83/76%</td>
<td>0.91/88%</td>
<td>0.89/84%</td>
</tr>
<tr>
<td>lattice-width-dep.</td>
<td>0.86/80%</td>
<td>0.94/91%</td>
<td>0.85/79%</td>
</tr>
<tr>
<td>Grade 3</td>
<td>1-syllable</td>
<td>2-syllable</td>
<td>3+4-syll.</td>
</tr>
<tr>
<td>baseline</td>
<td>0.85/79%</td>
<td>0.74/70%</td>
<td>0.81/75%</td>
</tr>
<tr>
<td>general probs.</td>
<td>0.93/92%</td>
<td>0.89/86%</td>
<td>0.89/86%</td>
</tr>
<tr>
<td>lattice-width-dep.</td>
<td>0.93/90%</td>
<td>0.85/82%</td>
<td>0.89/86%</td>
</tr>
<tr>
<td>Grade 4</td>
<td>1-syllable</td>
<td>2-syllable</td>
<td>3+4-syll.</td>
</tr>
<tr>
<td>baseline</td>
<td>0.85/80%</td>
<td>0.87/82%</td>
<td>0.80/72%</td>
</tr>
<tr>
<td>general probs.</td>
<td>0.88/84%</td>
<td>0.95/93%</td>
<td>0.89/84%</td>
</tr>
<tr>
<td>lattice-width-dep.</td>
<td>0.86/81%</td>
<td>0.93/90%</td>
<td>0.92/88%</td>
</tr>
</tbody>
</table>

Table 1
Evaluation of the proposed automatic classifiers for the different grades and reading tasks. Agreement (Kappa value) and percentage of correctly classified children given.

<table>
<thead>
<tr>
<th></th>
<th>1-syllable</th>
<th>2-syllable</th>
<th>3+4-syll.</th>
</tr>
</thead>
<tbody>
<tr>
<td>human (average)</td>
<td>0.94/92%</td>
<td>0.92/88%</td>
<td>0.93/90%</td>
</tr>
<tr>
<td>baseline</td>
<td>0.91/88%</td>
<td>0.90/88%</td>
<td>0.88/84%</td>
</tr>
<tr>
<td>general pros.</td>
<td>0.94/91%</td>
<td>0.90/85%</td>
<td>0.90/86%</td>
</tr>
<tr>
<td>lattice-width-dep.</td>
<td>0.95/92%</td>
<td>0.92/88%</td>
<td>0.86/80%</td>
</tr>
</tbody>
</table>

Table 2

Children per grade based on only one third of the recordings would lead to results with a high variance, we decided to classify all children from grades 1 to 4 together. From the table we can conclude that the human-machine agreements obtained equal the human-human agreements. Only in the case of the 3- and 4-syllable word task, the human-machine agreement is slightly, but insignificantly, worse.
3.2 Synthesis for oral feedback

One of the reading tutor tasks is inspired by the strategy of a human reading tutor (a teacher or a clinician) to provide feedback and assistance while a child is reading aloud and struggles or makes a mistake. Feedback has been shown to be of great importance for the development of reading for both children with and without reading difficulties (e.g. McCoy and Pany, 1986; Pany and McCoy, 1988; Perkins, 1988; Spaai et al., 1991; Wise and Olson, 1998). Since a beginning or struggling reader who makes a reading mistake has great difficulties correcting himself or herself, feedback should be provided by someone else. Therefore, the role of an intervening human tutor has always been indispensable for children with reading difficulties. As MacArthur et al. (2001) discussed, a computerised reading tutor could also be well suited to provide feedback and decoding assistance to a struggling reader by means of digitised (recorded) or synthesised speech, and the ability to simultaneously highlight words or word parts.

In the SPACE project, synthesised speech is used to provide three different kinds of oral corrective feedback of which the effectiveness has been proven in the literature: whole-word-feedback, syllable-by-syllable feedback, and phoneme-by-phoneme feedback (e.g. McCoy and Pany, 1986; Perkins, 1988; Spaai et al., 1991; Wise and Olson, 1998). With respect to the first type of feedback, the computerised reading tutor supplies the child with the whole word whenever an error is made against that word (e.g. *mother*). The second and third types of feedback concern the provision of segmentation cues by segmenting the erroneously read word into its constituent syllables (e.g. *mo-ther*) or phonemes (e.g. *m-o-th-e-r*). Immediately following the feedback presented by the reading tutor, the child is asked to read or reread the word. If the word is again not read without any reading error, feedback is given again until the child succeeds in decoding the word correctly (with a maximum of three feedback prompts for any particular word). During feedback, the word of interest is highlighted in the text. After the feedback has been given, the next word to read is underlined in order to indicate to the child where to continue his/her reading.

Additionally, the SPACE project has made an effort to build a synthesiser that is able to emphasise particular phonemes in phoneme-by-phoneme feedback. Inspiration for this feature comes from reading therapy experience that shows a tendency to emphasise the wrongly read phoneme in a word. For example, if a child has to read the word *wood* but reads the word *mood* instead, the human tutor tends to overemphasise the phoneme /w/ in the segmental (phoneme-by-phoneme) feedback presented to the child.

To evaluate the reading tutor’s speech synthesis abilities, an intervention study
was carried out in a group of ten Flemish elementary school children with reading difficulties. During a period of four weeks, each child received twenty minutes of computerised reading intervention on each school day. In each reading session, the child read instructional-level story material that was presented paragraph by paragraph on a touch screen connected to a laptop computer. Whenever the child experienced difficulties in decoding a particular word, help could be asked for by touching that word on the screen. If the child did not ask for help but did not succeed in reading the word correctly, feedback was automatically given. Although no automatic speech recognition was used yet in the presented study, the children were told that the computer could really listen to them reading aloud. In reality, a human listener, hidden behind the large touch screen, controlled the feedback supply in case of reading errors.

From this study, based on the different opinions of the three examiners involved, and based on the lack of reaction with respect to this matter from the children that exercised with the reading tutor, we conclude that the speech synthesis quality was good for all three of the feedback types: feedback sounded natural, and was clearly audible and well understandable. Children got quickly used to the reading tutor’s voice and were very understanding with respect to the reading tutor’s occasional mistakes (e.g. giving feedback although the word was decoded correctly) that were introduced by the human listener to emulate a computer’s behaviour.

However, with respect to the phoneme-by-phoneme feedback mode, some considerations for future improvements should be taken into account, for instance concerning the progressive and regressive assimilation processes in Dutch. These processes imply that a phoneme takes over the characteristics of the preceding (progressive assimilation) or following (regressive assimilation) phoneme in a word or a sentence. To obtain correctly pronounced whole-word and syllable-by-syllable feedback, it is important that these processes are accounted for by the speech synthesiser. But when synthesising phoneme-by-phoneme feedback, these processes should be neglected and phonemes should be produced as if they were standing alone. For example: in the Dutch word *zakdoek* [handkerchief], the phoneme /k/ before the /d/ becomes voiced (/zAgduk/). Nevertheless, when synthesising phoneme-by-phoneme feedback for this word, that phoneme needs to remain voiceless (/z/-/A/-/k/-/d/-/u/-/k/).

The situation is different with respect to the more specific phenomenon of *final devoicing*, where a voiced phoneme in a word becomes voiceless at the end of the word when followed by a pause. When synthesising phoneme-by-phoneme feedback for such a word, the phoneme at the end of the word needs to remain voiceless. For example, in the Dutch word *hond* [dog], the voiced phoneme /d/ becomes voiceless because it is placed at the end of the word: *hond* is pronounced as /hOnt/, not as /hOnd/. The phoneme-by-phoneme feedback should be /h/-/O/-/n/-/t/, keeping the final devoiced consonant, not altering
it in the way described in the previous paragraph.

Another valuable conclusion inferred from the intervention study, is that emphasis on the wrongly decoded phoneme does not seem to be feasible in every word under any circumstance. Although a more detailed observational study is recommended to detect the optimal conditions for specific phoneme emphasis, some general conclusions can already be made. The first conclusion is that specific phoneme emphasis does not feel natural if the child makes multiple errors within one word. Only for words in which the child makes an error on only one phoneme, a human tutor would consider to emphasise that phoneme when giving feedback to the child. Another observation is that emphasising a particular phoneme seems to be most effective for simple, monosyllabic words (e.g. when the child reads \textit{wood} for \textit{mood}, or \textit{mad} for \textit{map}). For example, if the child makes an error on the word \textit{handkerchief}, it is not very helpful for the child to emphasise the one phoneme that has not been decoded correctly. However, for such long words, specific phoneme feedback seems to sound natural again when the error falls on the first phoneme of the word.

3.3 Tracking where the child is reading

In an automated reading tutor, it is also important to track where the child is reading so that the progress through the reading task is known. As explained in section 2.2, the recognition system therefore provides the reading tutor with the current position (the most probable position according to the hypotheses in the search engine) in the FST that models the sentence being read. This tracker can be used for diagnostic and for remedial purposes. On the one hand, the tracker will be used for advancing automatically from one screen to the next in reading tasks that are presented on consecutive screens. On the other hand, the tracker can be used for generating feedback to the child, for instance, by highlighting the word he/she should read. This section presents our practical tests on children with known reading disabilities using the reading tutor supported by automatic tracking, and the improvements to our system based on the tests.

3.3.1 Automated screen advancement

To evaluate the automated screen advancement, 8 reading disabled children were asked to read a word reading exercise (7 exercises in total) or a story (2 in total). The selected reading tasks were \textit{difficult} compared to the child’s reading level. The word reading tasks consist of 40 single words presented one by one on the screen, the stories are presented on 4 or 5 consecutive screens. To prevent the children from getting confused when the tracker was making
a mistake, they were told that even a computer can make mistakes, and that they did not need to pay attention to the computer's possible mishearings. In total, the examiner had to intervene twice by reading the word herself, in 2 word reading exercises read by 2 different children. In 3 out of the 7 word reading exercises, the child needed to repeat one word (out of the 40 words) to get the tracker to advance to the next word. In 2 other word reading exercises, the child needed to repeat 2 respectively 3 words. For 9 words (from 6 different word reading exercises), the system advanced too fast so that the child did not get enough time to try to read that particular word. Automatic screen advancement in the two stories went perfectly. Thus, from this first qualitative evaluation, it became clear that some robustness had to be added to the system. As a result, four timing parameters were introduced to define the behaviour of the automatic screen advancement during a reading exercise:

- Minimal time of screen appearance: avoids that the system advances before anyone can read the screen.
- Maximal time of screen appearance: forces the system to advance, so that the system will not get stuck just because the tracker does.
- Amount of time the tracker feedback indicates the child is ready before the tutor really advances to the next screen, typically this is about 1 second. Basically, this means that the tutor advances when the child is silent for this amount of time: in this case, the child is supposed not to have another attempt.
- Amount of time the system is waiting between 2 screens, typically half a second. This avoids that the child, out of surprise that the tutor advances to the next screen, still says something that belongs to the previous screen but that will erroneously be interpreted by the tracker as part of the next.

A second qualitative evaluation of this automatic screen advancement option was carried out by getting another 22 reading disabled children read one (and in 2 cases; 2 respectively 3) of the relatively difficult word reading exercises (16 in total) or stories (9 in total). Again, they were asked not to pay attention to the computer’s possible mishearings. This time, for the word reading exercises, the examiner did not have to intervene once, and no repetition of words was needed in order to get the tracker proceed to the next word. The tracker moved too fast for only one word in one word reading exercise. Automatic screen advancement for all 9 stories went flawlessly. In summary, no real difficulties were encountered. Thus, by using proper values for the timing parameters mentioned above, the overall system for advancing to the next screen seems to work satisfactorily for diagnostic purposes when used in daily school practice.
3.3.2 Highlighting the word to be read

The feedback provided by the tracker can also be used to visually highlight a word (or words) on the screen. For 2 stories (read by 2 different children), the reading tutor highlighted the word the tracker feedback indicates as the word the child is reading. However, this form of feedback appeared to be useless for reading intervention because the child knows what he/she tries to read at that moment, so the feedback is not needed (in case the tracker is correct) or confusing (in case the tracker is wrong). Therefore, the word which the child should be reading, is highlighted. This is implemented by highlighting the word indicated by the tracker’s feedback, except in two cases:

- On the one hand, if the tracker’s feedback points to a word in an earlier part of the sentence (because the child repeats some words, or because the tracker makes an error), the highlighted word does not follow, it can only stick to the same word or advance. Besides maintaining the meaning of the highlighted word (namely that this is the word that should be read), this also avoids the situation when the possibly jumpy behaviour of the raw tracker feedback is shown directly on the screen.

- On the other hand, robustness is added to the system by allowing the teacher to set a maximum reading speed (in characters per second). If the tracker erroneously skips several words, the highlighted word will follow only slowly, so that the tracker is able to correct its error.

When providing the altered highlighting feedback to 8 other reading disabled children (9 stories in total), another problem became immediately clear: when a child is silent between two words (while decoding the next word), the tracker indicated the previous word. Because in this case, the next word should be highlighted, the tracker’s behaviour needed to be changed. This idea was supported by a third evaluation, in which another 10 reading disabled children were asked to read relatively difficult stories supported by highlighting feedback. However, overall, the current feedback system tends to proceed to the next word quite quickly, even if the previous word has not been pronounced correctly or completely. This behaviour is satisfactory for slow, both interior/silent and exterior/aloud decoding readers who need to learn to read faster. For fast and erroneous readers, however, highlighting seems to be of little use as a remedial tool. It may be more useful in this case to indicate words read incorrectly after reading; these readers need to be made aware of their errors and need to be forced to read at a slower pace.

\[\text{At the word recognition level, the slow/accurate-fast/inaccurate dichotomy has been associated with the indirect versus direct word approach (Coltheart, 1978).}\]
4 Conclusions and future work

In this paper, we presented several diagnostic and remedial reading tutor tools that are based on speech technology: tools for synthesis in phoneme or syllable mode, for tracking where the child is reading, and for assessment of the child’s reading level. In general, these tools seem to work satisfactorily when evaluated on children, including children with reading disabilities. As for the assessment, which was based on a reading task with only 40 words, we found that a child can be assessed in 5 performance groups equally well by a computer than by a human. The outcome of the extensive evaluation of the remedial tools was dual. On the one hand, as described in detail in this paper, the evaluation resulted in several improvements of the tutor and the speech technology involved, as to make the overall system more robust. On the other hand, the evaluation proved that the presented tools work in practice, in a real school environment and for the target users, children.

Future work on the reading tutor includes the development of an automatic quality measure for the synthesised speech, which may, for instance, help the teacher to prepare the exercises in the tutor. Also, the current automatic detection of reading errors needs to be improved as it is insufficient for adequate word-by-word feedback to the child.

Furthermore, one of the planned features of the SPACE reading tutor is synchronised reading, the ability to read along with the child. The reading tutor plays the role of a speech therapist, which sets the reading pace. While reading, the reading speed is maintained, unless the feedback from the recogniser indicates that the child has difficulties to follow the tempo. Then the reading tutor slows down, or falls back to the syllable mode for synthesis.

5 Acknowledgements

The research in this paper was supported by IWT project SPACE (sbo/040102): SPeech Algorithms for Clinical and Educational applications, home page: http://www.esat.kuleuven.be/psi/spraak/projects/SPACE. We would like to thank the participating schools, teachers and pupils for their collaboration in this project.

References


McCoy, K., Pany, D., 1986. Summary and analysis of oral reading corrective
feedback research. The Reading Teacher 39 (6), 548–554.