Available online at www.sciencedirect.com
ScienceDirect
Speech Communication 51 (2009) 853–863
SPEECH COMMUNICATION
www.elsevier.com/locate/specom

Oral proficiency training in Dutch L2: The contribution of ASR-based corrective feedback

Catia Cucchiarini *, Ambra Neri, Helmer Strik

CLST, Department of Linguistics, Radboud University Nijmegen, Erasmusplein 1, 6525 HT Nijmegen, The Netherlands

Received 30 June 2008; received in revised form 14 January 2009; accepted 15 March 2009

Abstract

In this paper, we introduce a system for providing automatically generated corrective feedback on pronunciation errors in Dutch, *Dutch-CAPT*. We describe the architecture of the system paying particular attention to the rationale behind it, to the performance of the error detection algorithm and its relationship to the effectiveness of the corrective feedback provided. It appears that although the system does not achieve 100% accuracy in error detection, learners enjoy using it and the feedback provided is still effective in improving pronunciation errors after only a few hours of use over a period of one month. We discuss which factors may have led to these positive results and argue that it is worthwhile studying how ASR technology could be applied to the training of other speaking skills.

Keywords: Computer assisted pronunciation training (CAPT); Corrective feedback; Pronunciation error detection; Goodness of pronunciation; Accent reduction

1. Introduction

In a globalized world characterized by increasing internationalization and mobility, the demand for products and material that make it possible to learn new languages efficiently and effectively is growing steadily. To function properly in the second language (L2), non-natives have to acquire various skills and subskills, the importance of which varies depending on factors such as the communicative situation, but also the educational level and the career of the learner. In terms of practice, some of these skills pose more problems than others in the language classroom, because their training requires more time. A good example of a skill that can never be sufficiently practiced in the classroom because of lack of time is represented by speaking, in the broad sense provided by Lennon (1990, p. 389) of "a cover term for oral proficiency, representing the highest point on a scale that measures spoken command of a foreign language". However, speaking is a fundamental skill for all learners, irrespective of their educational level or career: successful interaction in the L2 requires that non-natives speak with a minimal degree of fluency.

The progress made in automatic speech recognition research (ASR) in the last two decades has opened up new opportunities for applying this technology to the training and testing of oral proficiency in second language learning. This is not surprising considering that ASR-based computer assisted language learning (CALL) systems can offer extra learning time and material, specific feedback on individual errors and the possibility for self-paced practice in a private and stress-free environment. However, since it is also well-known that ASR technology still has considerable limitations (Ehsani and Knodt, 1998; Neri et al., 2002) it seems legitimate to question to what extent ASR can successfully be used in CALL systems to improve oral proficiency.

In this paper, we report on our research on the suitability of ASR technology for computer assisted pronunciation training (CAPT). The purpose of this study is to show that ASR technology, if properly tuned to specific pedagogical goals, can be effective in improving pronunciation skills despite occasional errors. We describe the research context

* Corresponding author. Tel.: +31 24 3615785; fax: +31 24 3612907.
E-mail addresses: C.Cucchiarini@let.ru.nl (C. Cucchiarini), ambra.neri@gmail.com (A. Neri), h.strik@let.ru.nl (H. Strik).

0167-6393/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.specom.2009.03.003

(Section 2), the procedure we adopted to select the pronunciation errors to be addressed in the system (Section 3), the architecture of the system we developed (Section 4) and the method used for error detection (Section 5). We then go on to present and discuss the results of a study in which the performance of *Dutch-CAPT* was examined from different viewpoints (Sections 6 and 7). Finally, we briefly describe how we intend to pursue this type of research in new projects (Section 8).

2. Research context

In the 1990s, many schools in the Netherlands, but also penitentiaries and asylum-seekers centers, had been using a multimedia package for teaching Dutch as L2 called *Nieuwe Buren* (New Neighbours). This is a comprehensive course for adult learners with primary- or secondary-school education and with different mother tongues (L1), but already familiar with the Roman alphabet. The course comprises a multimedia program with 40 video-episodes. Each episode, which is further divided into 10 sub-segments with 15 exercises each, presents real-life situations that are particularly likely to be experienced by the learners, e.g. how to obtain a working permit. This reflects the big effort made by the authors of this piece of courseware to foster the integration of the users within the Dutch community. In accordance with communicative language teaching guidelines, this type of material ensures that the students are exposed to near-authentic and meaningful language and that they actively practise even outside the classroom context. The teacher should in principle act as a coach and supervisor whose main task is to encourage communication and interaction because the course is intended to stimulate self-learning.

In *Nieuwe Buren*, it was envisaged that students could record their own speech and that later the teacher would listen to the recordings and provide feedback. In practice this almost never happened, mainly due to lack of time. So the question arose whether it would be possible to enrich *Nieuwe Buren* with an ASR module that would provide automatic feedback on pronunciation. To find an answer to this question, a project was started aimed at investigating the suitability of ASR technology for this purpose. Within the framework of this project a system was developed, called *Dutch-CAPT*, that provides corrective feedback on a selected number of speech sounds that have appeared to be problematic for learners of Dutch from a variety of L1 backgrounds. In the following section we describe how these errors were selected.

3. Selection of pronunciation errors

In accordance with the practice in Dutch-L2 courses, which are characterized by heterogeneous groups with respect to mother tongue, *Dutch-CAPT* had to address pronunciation errors that can be made by any learner, regardless of the learner's L1. In addition, the rationale behind this system was that it should not address all pronunciation errors, but only a selection of relevant ones, in order to optimize the training. To decide which errors should be addressed by *Dutch-CAPT* we adopted the following six criteria:

1. Common across speakers of various L1s.
2. Perceptually salient.
3. Potentially hampering communication.
4. Frequent.
5. Persistent.
6. Suitable for robust automatic detection.

The first criterion is obvious given the Dutch context. The second criterion is in line with current pronunciation training approaches that tend to focus on realizations that are perceived as clearly erroneous by human listeners. Along the same lines, deviant realizations that Dutch listeners are familiar with because they are found in certain well-known non-standard Dutch varieties ought not to be prioritized, whereas attention should be concentrated on perceptually salient deviations that are likely to hamper communication (criterion 3). For instance, errors whereby a phoneme is often realized as another (legal, Dutch) phoneme, leading to the realization of a different word, are likely to hamper communication as they might result in a different message being understood. Furthermore, to improve communication it is important to address frequent errors (criterion 4) and errors that appear to persist over time and possibly over a certain amount of instruction (criterion 5). Finally, in order to limit the amount of erroneous feedback that might be generated by the machine/ASR component, it is advisable to provide feedback on pronunciation errors that can be detected with a considerable degree of reliability (criterion 6). This last criterion can be properly tested only once the error detection algorithm has been developed and fine tuned.

To obtain a detailed inventory of segmental errors in non-native Dutch speech, we analyzed three different databases of Dutch non-native speech produced by a total of 116 learners with different mother tongues and proficiency levels (Neri et al., 2006). We made annotations of perceptually salient errors, checking the reliability of the annotations, subsequently studied the possible source and impact of the errors to remove unlikely or irrelevant ones (Neri et al., 2006). Finally, the errors that met all or most of the predetermined criteria were selected and tested to check the degree of reliability with which they could be correctly identified. This resulted in the following final selection of 11 Dutch phonemes, which were addressed in *Dutch-CAPT*:/ʏ/, /x/, /ɑ/, /y/, /œy/, /aː/, /ɛi/, /h/, /u/, /øː/, /ɪ/.

4. Functional description of *Dutch-CAPT*

We designed and developed a system to provide feedback on errors related to these eleven target phonemes, *Dutch-CAPT*. The system is comprised of two main parts (see Fig. 1): (1) the client, with the user interface (UI),

audio handling (including endpoint detection), and the didactic content of the system and (2) the server, with the ASR module that analyses the students' utterances. More details are provided below. The client and server where running on different computers; several clients (which resided on the computers of the subjects) were sometimes running simultaneously, and all were connected over the internet to one central server. The two parts communicate through two sockets, one to exchange control signals (coded in XML), the other to exchange speech data.

4.1. The UI

The UI contains the didactic content of *Dutch-CAPT* and is available in an English and a Dutch version. When the program is started, the user is prompted to enter a personal four-digit ID. This ID, which is needed to keep logs of the students' activities, also indicates the student's gender, which must be specified because the ASR module makes use of different parameter settings for the acoustic analyses of male and female speakers.

The navigation through the exercises is constrained and sequential and requires users to complete an exercise before proceeding to the following one. This constrained navigation could only be overruled by the experimenter and was chosen for testing purposes (see Section 6), to ensure that all subjects would receive the same amount of training.

The didactic content consists of 106 exercises grouped into four units, with which users can practise pronunciation. Each unit also contains a video from *Nieuwe Buren* presenting typical communicative situations (buying groceries, going to the cinema, etc.) and providing words and expressions to be practised orally in that unit. The exercises include:

- Twenty-two exercises in the form of dialogues. These exercises simulate parts of the dialogues presented in the videos. The user has to choose one role and to pronounce the character's lines in a flow, as if actually speaking with the other character (see Fig. 2).
- Forty-six exercises consisting of questions that are either written out or are only available in audio-format.
- Thirty-eight exercises consisting of individual words that have to be pronounced on the basis of the model utterances recorded by male and female native speakers (see Fig. 3). These exercises include several minimal pairs containing the target phonemes.

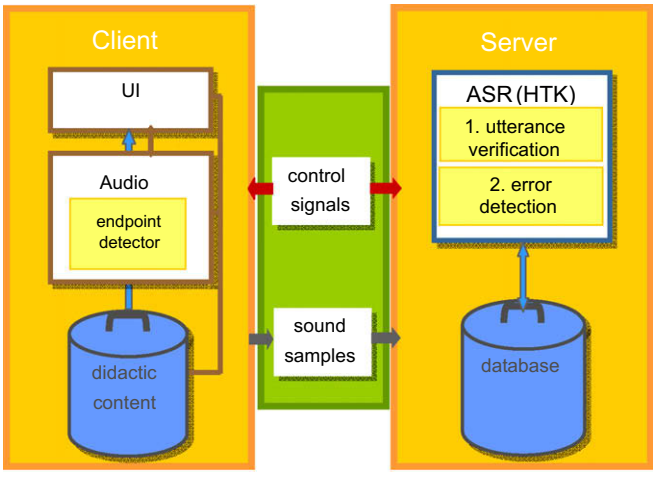

Fig. 1. Schematic representation of the *Dutch-CAPT* system.

4.2. The ASR module

The ASR module contains the technology that performs the analyses of the users' utterances and is able to handle multiple simultaneous client processes, for which it creates separate IDs. A log file is maintained in the ASR module, which contains a list of important tasks happening in both parts of the program. These tasks are labelled with unique process IDs and time stamps, e.g. the start of the recognition of an utterance or the results of the analysis of its quality. This log file is saved in the ASR module, while a copy is simultaneously updated on a website, so that, during testing, an experimenter can monitor several users in real time.

4.3. How Dutch-CAPT works

The exercises are presented on the top half of the screen and the feedback on the lower half (see Figs. 2 and 3). For each utterance, example pronunciations are available that can be listened to before speaking and recording an answer. When a user starts recording the answer, an endpoint detector is started so that the recording can be automatically stopped once the user has finished speaking (based on tests, a minimum delay of 2.5 s was applied to ensure that the recording would not be erroneously stopped in the case of hesitations or pauses in a student's utterance).

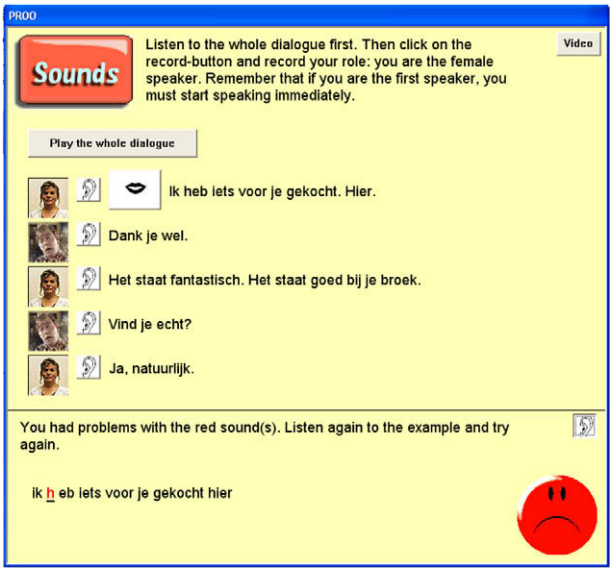

Fig. 2. Screenshot taken after the female user has received negative feedback.

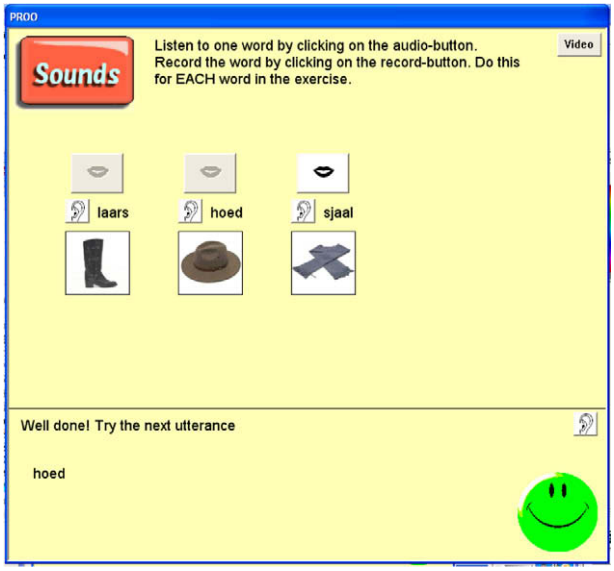

Fig. 3. Screenshot taken after the user has received positive feedback.

The ASR module must first of all establish whether the audio file received matches one of the possible answers for that given exercise (see Fig. 1, utterance verification) for two reasons.

One has to do with the credibility of the system: if a user can utter any word or sentence including, for instance, 'I am tired of using this system' without the system noticing that this is not one of the correct options, s/he will soon stop trusting the system as a useful tool. Though this observation may sound obvious, this problem is a serious one in CAPT systems providing automatic feedback without ASR technology. These systems provide a visual representation of the student's speech, often in combination with that of a model utterance by a native speaker. The representation can vary from an oscillogram, to a pitch curve, or a more intuitive type of display such as steps, or blocks, representing syllables of various duration (the length of the step) and with different vowel's energy (the step's height) (for an overview of these systems, see Neri et al., 2002). But as these systems analyse the incoming speech signal without first 'recognizing' the utterance, there is no guarantee that the student's utterance does indeed correspond to the intended one. This implies that the system can be easily fooled, which obviously would negatively affect its credibility as a training tool.

The other reason is a practical one: if it cannot be established which utterance the user has pronounced, it is also impossible to provide information on the words and sounds in that utterance. In such cases, the system will reject the utterance and prompt the user to try again.

If the ASR module finds a suitable match to the user's audio file, it immediately starts analyzing pronunciation quality. If no error is found, (a) a message congratulating the student, (b) the orthographic transcription of the utterance recognized, and (c) a green, happy smiley will appear on the lower half of the UI together with a play button enabling the user to listen again to his/her own pronunciation (see Fig. 3).

If an error is found, the UI will display (a) a message informing the student that there was a problem with one or more sounds and prompting him/her to try again after listening to the target pronunciation, (b) the transcription of the utterance with the grapheme(s) corresponding to the erroneous phoneme(s) coloured red and underlined, and (c) a red, disappointed smiley (see Fig. 2). No more than three errors are signalled each time in order not to discourage the student.

If the user does not succeed in pronouncing a target utterance correctly over three successive attempts, a message indicating that there are still some problems is shown and the student is allowed to move on to the following exercise.

As can be seen, the way that feedback is presented to the user is quite simple. The choice for this presentation form was motivated by indications from various studies on negative user experience effects: other feedback forms such as digital waveform plotting (which is readily available in most signal processing software), spectrograms, as well as different types of scores of pronunciation quality are often found incomprehensible and/or uninformative by users (see Reeser, 2001; Carey, 2002; Neri et al., 2002; Zheng, 2002; Giuliani et al., 2003). Obviously, such effects may nullify the pedagogical value of providing feedback.

5. Error detection in *Dutch-CAPT*

5.1. The ASR module

The ASR module was implemented using the hidden Markov model Toolkit (HTK, see Young et al., 2000). It makes use of 37 context-independent, monophone hidden Markov models (HMM) (see Van Bael et al., 2003). These HMMs are gender-dependent and were trained on read material from the library of the blind and the broadcast news of the Spoken Dutch Corpus (Oostdijk, 2002). The phone set includes a general speech model to account for unintelligible speech as well as a silence and a short pause model. Except for the short pause model, each HMM has three states and 32 Gaussian mixtures per state. The single state of the short pause model is tied to the central state of the silence model. Optimal word insertion penalty (WIP) and language model factor (LMF) values were determined using the development test set. The acoustic pre-processing implemented in the current version of *Dutch-CAPT* includes channel normalization by means of cepstral mean subtraction at utterance level, to account for the effect of using different microphones.

5.2. Utterance verification

As mentioned above, before any errors can be detected, it must be established which utterance has been spoken by the student. This is done in the following way. An HTK compatible Bachus–Naur form (BNF) grammar is generated

for each exercise. This grammar contains an orthographic transcription of each possible target utterance for the exercise, and an extra path representing any possible non-target utterance. The canonical phonetic transcriptions of the target utterances are stored in a previously created lexicon, and the non-target utterance is represented by a free phone loop. The path obtaining the highest likelihood is chosen, and in this way it is verified whether one of the possible target utterances is chosen or a non-target utterance. The WIP and LMF values used to implement utterance verification were determined during pre-tests, in order to optimize the performance of utterance verification.

5.3. The error detection algorithm

The error detection algorithm was implemented by means of confidence measures according to the GOP algorithm proposed by Witt and Young (2000) and Witt (1999). The GOP algorithm calculates the likelihood ratio that a phone realization corresponds to the phoneme that should have been spoken (the so-called Goodness of Pronunciation value). The student's speech is subjected to both a forced and a free phonetic recognition phase. During forced recognition, the orthographic transcription of the speech signal that was identified during utterance verification is used to force the recognition of the speech and in the free recognition phase the phoneme sequence most likely to be spoken is calculated. A GOP value of a specific phone realization is then calculated by taking the absolute difference of the log probability of the forced and the log probability of the free recognition phase. Phones with GOP values above a pre-defined threshold are probably mispronounced and are for that reason rejected by the algorithm. Likewise, phones with values lower than the pre-defined threshold will probably be well-pronounced and are accepted.

5.4. Performance measures

To achieve optimal performance, an error detection algorithm should detect pronunciation errors and, at the same time, it should not flag as erroneous those realizations that were actually correct. For this reason both the amount of correctly rejected (CR) and correctly accepted (CA) realizations are important in the performance calculation.

The performance of an error detection algorithm can be calculated in different ways. One way is to measure the scoring accuracy (SA), which is calculated by Eq. (1) shown below:

$$\text{Scoring Accuracy} = \left(\frac{\text{CA} + \text{CR}}{\text{CA} + \text{CR} + \text{FA} + \text{FR}}\right) \times 100 \tag{1}$$

where FA stands for false accepts, mispronunciations that were falsely judged as correct, and FR indicates false rejects, correct realizations that were falsely flagged as errors.

Other widely used measures for calculating the performance of a classification algorithm are precision, recall and the *F*-measure. These metrics can be calculated both for the correct accepts and the correct rejects (see (2)–(6)).

$$\text{Precision of CA} = \left(\frac{\text{CA}}{\text{CA} + \text{FA}}\right) \times 100 \tag{2}$$

$$\text{Precision of CR} = \left(\frac{\text{CR}}{\text{CR} + \text{FR}}\right) \times 100 \tag{3}$$

$$\text{Recall of CA} = \left(\frac{\text{CA}}{\text{CA} + \text{FR}}\right) \times 100 \tag{4}$$

$$\text{Recall of CR} = \left(\frac{\text{CR}}{\text{CR} + \text{FA}}\right) \times 100 \tag{5}$$

$$F\text{-measure} = 2 \times \left(\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}\right) \tag{6}$$

5.5. Establishing thresholds

An important aspect in the algorithm optimization procedure is choosing the thresholds that maximize the scoring accuracy. In our specific case, an additional criterion was added: we aimed at finding GOP thresholds that maximize SA while keeping FR below 10%. The rationale behind this decision was that erroneously rejecting correct pronunciations would be more detrimental for learners than erroneously accepting mispronunciations, considering that many of our target users were expected to be beginner learners. Therefore, the optimization criterion used here consisted in maximizing scoring accuracy, established by Eq. (1) above, for FR lower than 10%.

Since not enough non-native speech material was available to train the error detection algorithm, pronunciation errors were simulated by changing the phonemic representations in the lexicon of a corpus of native speech. Such a simulation procedure had been applied by Witt (1999) as well. For this purpose, two sub-corpora of the Spoken Dutch Corpus (Corpus Gesproken Nederlands; CGN) were used. The CGN is a corpus of about 9 million words that constitutes a plausible sample of standard Dutch as spoken in the Netherlands and Flanders, with speech recordings that were annotated at different levels (Oostdijk, 2002). We chose two sub-corpora of Dutch spoken by native speakers from the Netherlands: one was used as development corpus (CGN-dev), and an independent one as test corpus (CGN-test).

Optimal GOP thresholds were established in the following way. The artificial errors were introduced in the pronunciation dictionary for the 11 target phonemes, phone by phone. For each phone, for half of the entries containing that phone, the correct pronunciation (i.e. phone) was replaced by an incorrect pronunciation (i.e. another phone, see examples in Table 1). The scheme according to which correct phones were replaced by erroneous ones was based on information that we had collected on how Dutch phones are frequently mispronounced by L2 learners (Neri

Table 1
Common pronunciation errors used to create artificial non-native data.

Target phoneme	Example	Incorrect realization	Example
/øː/	deur (door)	/y/	duur (duration)
/œy/	huid (skin)	/ʌu/	houd (keep)
/ɑ/	man (man)	/aː/	maan (moon)
/aː/	maan	/ɑ/	man
/ɛ/	leg (put)	/eː/	leeg (empty)
/ɛi/	wijs (wise)	/eː/	wees (orphan)
/ɪ/	lip (lip)	/i/	liep (walked)
/y/	bus (bus)	/u/	boek (book)
/y/	buur (neighbour)	/u/	boer (farmer)
/x/	geel (yellow)	/k/	keel (throat)

The incorrect realizations are all legal Dutch phonemes.

Table 2
The number of phoneme realizations, their distribution into CA, CR, FA, and FR, and the performance results on CGN-test.

	CGN-test
Tot # realizations	191,147
CA	40.25%
CR	41.42%
FA	8.54%
FR	9.79%
SA	81.67%
Precision of CA	82.49%
Recall of CA	80.43%
F-measure of CA	81.45%
Precision of CR	80.88%
Recall of CR	82.90%
F-measure of CR	81.88%

et al., 2006). Optimal thresholds were then established for each gender-phoneme combination by carrying out an exhaustive search. Preliminary experiments had shown that a step size of about 0.25 was sufficient, since generally there is a range of threshold values for which the values of SA do not differ significantly. The GOP thresholds were established by using the development corpus CGN-dev and were evaluated on the independent test corpus CGN-test. The results are presented in Table 2.

6. Evaluation of *Dutch-CAPT*

The performance of *Dutch-CAPT* was evaluated from different angles. To establish the potential, pedagogical usefulness of *Dutch-CAPT* in realistic conditions, we studied a group of immigrants who were learning Dutch in the Netherlands. We collected data on (1) the learners' appreciation of the system (Section 6.1.3), (2) their global pronunciation quality as assessed by experts (Section 6.1.4) and (3) their performance on the specific aspects of segmental quality addressed in *Dutch-CAPT* (Section 6.1.5). To establish the technical performance of *Dutch-CAPT*, we measured the scoring accuracy of the error detection algorithm (Section 6.2).

6.1. Pedagogical usefulness of Dutch-CAPT

6.1.1. Subjects

The participants were 30 adult immigrants varying with respect to age, occupation, and length of residence in the Netherlands, who were following beginner courses of Dutch at UTN, the language center of the Radboud University Nijmegen. They were assigned to three different groups according to instructions from the Dutch-L2 coordinator at UTN, who required that students from one class would follow the same training:

- Experimental group (EXP). Fifteen participants, 10 female and 5 male.
- Control group 1 (NiBu). Ten (4 female and 6 male) participants.
- Control group 2 (noXT). Five (3 female and 2 male) participants.

6.1.2. Training procedure

All three groups followed the regular classes, which included self-study sessions in the language lab. NiBu and EXP had one CAPT session per week for 4 weeks, with each session lasting from 30 min to 1 h, depending on the participant's training pace. NiBu worked with a reduced version of *Nieuwe Buren* that matched *Dutch-CAPT*. The students in this group could record their own utterances and compare them to example utterances, but they did not receive any feedback and thus had to rely on their own auditory discrimination skills. Log files of each student's activities allowed the experimenter to check that all students completed all exercises as requested. EXP used *Dutch-CAPT*, which was similar to part of *Nieuwe Buren*, the only difference being that it provided automatic feedback on segmental quality. The noXT group received no extra training in addition to the regular lessons.

6.1.3. Analysis of learners' appreciations of the system

Anonymous questionnaires were used in which participants had to indicate whether or not they agreed with a number of statements on a 1–5 Likert scale and to answer two open-ended questions. The questions concerned the accessibility of the exercises, the usability of the interface in general, the students' feelings about the usefulness of the specific CAPT for improving pronunciation, and their opinion about specific features of the system used.

Overall, the responses indicated a positive reaction to the two CAPT programs, with mean scores per statement ranging between 2.4 and of 4.6 for EXP, and between 2.3 and 4.7 for NiBu. This result is in line with other studies on student appreciation of CAPT, including ASR-based CAPT (Mak et al., 2003). More specifically, the answers indicate that the students enjoyed working with the CAPT system provided and that participants generally believed in the usefulness of the training. With respect to *Dutch-CAPT*, 8 of the 14 participants who provided comments

on the system said that it was helpful, mostly in improving their pronunciation and in making them aware of specific pronunciation problems.

6.1.4. Analysis of global segmental quality

The subjects were tested before and after the training (pre-test and post-test). To ensure that the rating process would not be influenced by possible lexical or morphosyntactical errors, read speech containing every phoneme from the Dutch phonemic inventory at least once was used (phonetically rich sentences).

Two groups of three expert raters evaluated the speech independently on a 10-point scale, where 1 indicated very poor segmental quality and 10 very good segmental quality. They were instructed to focus on segmental quality only, and to ignore aspects such as word stress, sentence accent, and speech rate, since these aspects were not the focus of the training. No further instructions were given as to how to assess segmental quality. However, the raters were provided with examples of native spoken utterances and non-native spoken utterances of 'poor' segmental quality of the experiment stimuli, to help them anchor their ratings (Cucchiarini et al., 2000). Pre- and post-test recordings were presented in random order.

Before assessing the effect of the training on overall segmental quality for each group, we checked the reliability of the ratings – i.e. to what extent the ratings of the judges varied in the same way. The measure we used is the coefficient of reliability Cronbach's alpha, which indicates to what extent a set of variables measure the same construct. For the two groups of raters, inter-rater reliability was 0.96 and 0.95 for all scores and 0.83 and 0.87 when the scores assigned to the native speech fragments were removed. Intra-rater reliability was higher than 0.94. These coefficients are high, especially if we consider that no clear, pre-specified criteria for assessment were provided.

After having established that the non-natives had not received scores in the range of the natives at pre-test, we looked at the average improvement made by the three groups after training, finding that overall segmental accuracy improved for all groups at post-test (see Fig. 4).

Subsequently, an analysis of variance (ANOVA) with repeated measures was conducted on the subjects' scores, with Test time (Levels: pre-test, post-test) as within-subjects factor and Training group (Levels: Exp, NiBu, NoXT) as between-subjects factor. The results indicated a significant effect for Test time, with $F(1,27) = 18.806$, $p < .05$ with the post-test scores reflecting significantly greater segmental accuracy ($M = 5.19$, $SD = 1.53$) than the pre-test scores ($M = 4.42$, $SD = 1.54$). The interaction between Test time and Training group was not significant, indicating that there were no significant differences in the mean improvements of the training groups.

To summarize, all three groups improved global segmental quality after the training, with the group receiving ASR-based corrective feedback showing the largest improvements, followed by the group receiving extra

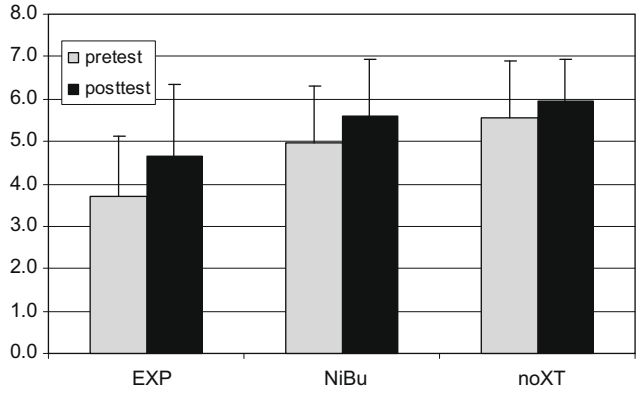

Fig. 4. Mean scores (based on a 10-point scale) for each training group at each testing condition. Bars indicate one SD from the mean.

CAPT, but no corrective feedback. However, the difference in improvements in the three groups is not statistically significant. Several explanations can be advanced for these results, such as the limited amount of training (only a few hours of use over a period of one month), the small sample size and the relatively large variation in overall segmental quality within each training group and between training groups. This variation is partly a result of the impossibility to match participants prior to the training, to comply with the instructions from the UTN coordinator. This resulted in EXP's and noXT's mean scores being significantly different at pre-test which in turn could suggest that EXP's larger improvement might be due to their initial lower level and to noXT having reached a ceiling. To better understand these results, we carried out a fine-grained analysis of the segmental errors made by the participants before and after the training.

6.1.5. In-depth analysis of segmental quality

An additional, detailed analysis was carried out of the specific errors made by the participants, in order to obtain more fine-grained information on the effectiveness of the computer-generated feedback. For this investigation, auditory analyses were carried out on a subset of the participants' recordings, and annotations were made of specific segmental errors.

An expert annotator listened to the recordings and indicated whenever a phoneme was pronounced incorrectly. We used these annotations to examine possible post-test improvements on the 11 target phonemes and on the remaining, untargeted phonemes for the participants who received automatic feedback (EXP) and those who did not (NiBu).

To quantify possible decreases in errors, we calculated the percentage of errors made by each student at pre-test and post-test for each of the two types of phonemes (targeted and untargeted) relative to the amount of total phonemes of the same type in the stimuli. The results (see Fig. 5) show, first of all, that there were more errors on

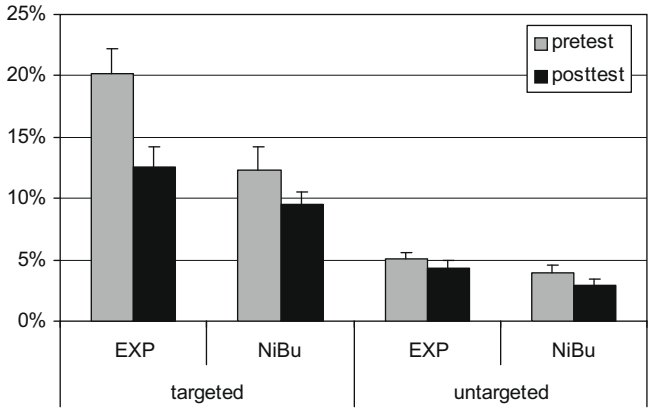

Fig. 5. Mean error percentages (and *SEM*s) for errors on the targeted and untargeted phonemes.

the targeted phonemes than on the untargeted ones. As these data did not meet all assumptions necessary for an ANOVA, non-parametric tests were performed. We examined the difference between the improvements of each group on the two types of errors by using the Wilcoxon Rank-sum test. The test on targeted errors indicated a significant difference between EXP and NiBu ($z = -2.827$, $p = .002$, one-tailed), with EXP making a significantly larger improvement than NiBu on segmental quality of the targeted phonemes.

For untargeted errors a different trend appeared (see Fig. 5): while both groups produced fewer errors at posttest, the decreases in untargeted errors were much smaller and more similar across the two groups (0.7% for EXP and 1.1% for NiBu) than those for the targeted errors (7.6% for EXP and 2.4% for NiBu). The Wilcoxon Rank-sum test on untargeted errors indicated no significant difference between the mean improvements of the two groups on untargeted phonemes. As a matter of fact, in the case of untargeted errors it was NiBu that achieved the largest mean improvement, which seems to suggest that the improvement achieved by EXP might indeed be the result of the specific training provided, rather than being related to the fact that this group had more room for improvement.

In summary, these results show that (a) the participants produced relatively more errors for the targeted phonemes, which is an indication that these phonemes are, indeed, particularly problematic and segmental training should focus on these sounds and (b) the group receiving feedback on these errors made a significantly larger improvement on the targeted phonemes than the group receiving no automatic feedback, whereas no statistically significant difference was found for the phonemes for which no feedback was provided, suggesting that the automatic feedback provided in *Dutch-CAPT* was effective in improving the quality of the targeted phonemes and that training with automatic feedback at segmental level can offer added value with respect to training without such feedback.

6.2. Scoring accuracy in Dutch-CAPT

The scoring accuracy of the algorithm was measured for a total of 2174 phones deriving from 437 utterances produced by the 15 Dutch immigrants who used *Dutch-CAPT*. These utterances were selected as follows: for each participant, a maximum of 30 utterances was selected, with each utterance containing at least one error as detected by the GOP algorithm. A Dutch expert annotator carried out auditory analyses based on the system's output. She was asked to indicate if she disagreed with the algorithm by indicating whether the errors identified were actually correct realizations (FR) and whether phonemes that had been seriously mispronounced had not been identified by the algorithm (FA). The tagged orthographic transcription was used as a starting point in order to make the task more efficient. The results indicate a mean scoring accuracy of 86% (SD = 4%), a precision of CA of 86.55%, a recall of CA of 92.55%, an *F*-measure of CA of 89.45%, a precision of CR of 84.71%, a recall of CR of 74.13%, and an *F*-measure of CR of 79.06%. The results per subject are in Table 3. As can be seen from this table, the percentage of FR stays below 10%.

7. Discussion

The results on the evaluation of *Dutch-CAPT* presented in Section 6 show that (a) it is possible to develop a system that provides automatic feedback on a number of problematic Dutch sounds, (b) the students appreciate such a system, (c) the feedback provided was effective in improving the pronunciation of the problematic speech sounds addressed in the training, (d) the scoring accuracy was relatively good and (e) a system whose scoring accuracy is not 100% can still help to significantly improve the pronunciation of notoriously problematic sounds.

There are a number of considerations underlying *Dutch-CAPT* that are worth examining when trying to understand why *Dutch-CAPT* was successful:

(1) The errors to be addressed in the training were selected by analyzing speech data collected from speakers that belong to the eventual target group. Although this might sound obvious, in practice this is not always possible because such data are not always available.
(2) In a similar vein, the tests were carried out with target users. Again, this might also sound obvious, but is not, because in fact you are testing a system whose usefulness has not yet been proven and students are reluctant to participate.
(3) For establishing the GOP thresholds we used simulations of pronunciation errors that were very realistic. As a matter of fact, in simulating the pronunciation errors we first checked how Dutch phones are usually mispronounced and used this information in changing the phonemic representations. In Witt (1999),

Table 3
Classification accuracy of the GOP algorithm per subject.

Subject	Number of							
	Total phones	Target phones	CA	CR	FA	FR	SA	FR
1	424	130	68	40	13	9	0.83	0.07
2	386	108	57	33	9	9	0.83	0.08
3	480	148	83	43	18	4	0.85	0.03
4	500	146	89	41	11	5	0.89	0.03
5	462	122	63	39	15	5	0.84	0.04
6	680	170	106	40	16	8	0.86	0.05
7	413	125	79	31	8	7	0.88	0.06
8	448	120	74	34	8	4	0.90	0.03
9	398	133	60	39	26	8	0.74	0.06
10	512	156	94	30	21	11	0.79	0.07
11	412	130	64	50	13	3	0.88	0.02
12	481	141	86	41	9	5	0.90	0.04
13	485	137	84	35	5	13	0.87	0.09
14	591	164	117	32	5	10	0.91	0.06
15	847	244	169	48	24	3	0.89	0.01
Total	7519	2174	1293	576	201	104	0.86	0.05

on the other hand, artificial errors were created by replacing in the lexicon all realizations of a given phoneme, say /a/ by another one, say /i/. However, the chance that language learners will make that type of error is smaller than that they will confuse or mispronounce phonemes that are acoustically more similar such as /i/ and /ɪ/, or /χ/ and /k/. Likewise, the GOP algorithm will have a harder time in distinguishing /i/ from /ɪ/, or /χ/ from /k/ than in distinguishing /a/ from /i/. This might explain why thresholds based on artificial data on native speech worked properly for real errors in non-native speech.

(4) The decision to provide feedback only on a limited number of well-selected problematic phonemes.
(5) The decision to limit the feedback to a maximum of three errors per utterance so as to avoid overwhelming the learners with too much information on many different mistakes.
(6) The simple and straightforward feedback provided. Research on the effectiveness of feedback in L2 teaching has revealed that providing examples of the mistakes produced is probably unnecessary, while it suffices to point out the problematic areas. The type of corrective feedback provided in *Dutch-CAPT* comes very close to what is considered to be optimal corrective feedback "clear enough to be perceived as such" and allowing for self-repair and modified output (El Tatawy, 2002).
(7) A final point that probably deserves attention is the decision made in *Dutch-CAPT* with respect to the balance between FAs and FRs. As is well-known, there is a trade-off between FAs and FRs. In *Dutch-CAPT* we decided to keep FRs below 10% and tolerate FAs on the grounds that, for beginner learners, erroneously rejecting correct sounds would be more detrimental than erroneously accepting incorrect ones. As a matter of fact, beginners are likely to produce many errors and, accordingly, to receive a considerable amount of negative feedback. To avoid the risk of frustrating the students by rejecting correct utterances too (Egan, 1999; Herron et al., 1999) it seemed wiser to tune the error detection algorithm so that the chance of false rejects would be as low as possible. In other words, in *Dutch-CAPT* only patently wrong sounds were (usually correctly) rejected, while a number of incorrect sounds were accepted as correct. This might have had a positive effect by enabling the learners to concentrate only on the most serious errors and to gain self-confidence (Egan, 1999).

At this point, the question arises as to how *Dutch-CAPT* could be improved to get a more substantial impact on global pronunciation quality thus leading to accent reduction. First of all, by providing a more intensive training, as the amount of training in our experiment was very limited. For instance, one could think of providing dedicated exercises whenever one of the targeted error is detected. A more intensive training might also have the effect of extending the learning effect to other, similar phonetic contrasts, for instance that between /oː/ and /ɔ/ and that between /eː/ and /ɛ/ as a result of training the /aː/-/ɑ/ contrast. These are issues that we intend to address in future research.

Furthermore, focusing the training on a limited number of phonemes that were problematic for a heterogeneous group of learners entailed that, for each specific learner, few problematic phonemes could be addressed. It is therefore possible that a different approach in which individual learners receive feedback on a larger number of problematic phonemes is likely to be more effective at the global level. Such an approach could either consist in having a system that simply addresses more phonemes, or a system that is specifically designed for addressing pronunciation errors pertaining to specific language combinations such

as, for instance, Turkish-Dutch or French-Dutch. Such an L1-specific CAPT system might be more effective. The point then is how feasible it is to develop a variety of such systems for all L1s that might be relevant in a given linguistic context.

The finding that the effect of the corrective feedback did not emerge from the global ratings of pronunciation quality, but was evident in the fine-grained analyses of the students' utterances is a result that deserves attention in future evaluations of such CAPT systems. Global ratings of pronunciation quality are an appropriate dependent variable in such evaluations, because at the end of the day CAPT should improve overall pronunciation quality. At the same time, it is clear that when evaluating systems that address specific pronunciation problems, a type of analysis with higher resolution may be required to better understand the ultimate effect of the training. In our case, this more detailed analysis has shown that the ASR-based feedback was effective in improving the errors addressed in the training, but the results of the overall pronunciation ratings have made clear that this is not enough to get a significant difference in improvement with respect to the control groups.

8. Future research: from *Dutch-CAPT* to DISCO

The results presented above are promising and show that it is possible to use speech technology in CALL applications to improve speaking proficiency. In the Netherlands, speaking proficiency plays an important role within the framework of civic integration examinations. Foreigners who wish to acquire Dutch citizenship have to show that they are able to get by in Dutch society and that they speak the Dutch language at the common European framework (CEF) A2 level, which means that they can make themselves understood in Dutch.

However, pronunciation is only one of the skills required for speaking a second language. There are also other aspects of spoken language that are important and that have to be mastered in order to be comprehensible and proficient in a second language. For instance, morphology and syntax also play an important role in language comprehension and language learning. It is known that learners tend to make different morphologic and syntactic mistakes when they speak than when they write. It is generally acknowledged in the L2 literature that awareness of certain grammatical rules in L2 learners (i.e. those concerning subject-verb concord of number, tenses for strong and weak verbs, and plural formation) does not automatically entail that they also manage to marshal this knowledge on line while speaking. In other words, in order to learn to speak properly in a second language, L2 learners need to practice speaking and need to receive corrective feedback on their performance on line, not only on pronunciation, but also on morphology and syntax.

A CALL system that is able to detect errors in speaking performance, point them out to the learners and give them the opportunity to try again until they manage to produce the correct form would be very useful because in L2 classes there is not enough time for this type of practice and feedback. We therefore decided to explore the possibility of extending the approach adopted in *Dutch-CAPT* to other aspects of speaking proficiency like morphology and syntax. It turned out that there are a number of errors in morphology and syntax that are common in spoken Dutch L2 and that could be addressed in an ASR-based CALL system.

Problems with morphology are persistent in L2 learning (DeKeyser, 2005) and phonetic-phonological properties play a prominent role in this learning process. As stated in (DeKeyser, 2005, p. 2): "The meaning of morphemes and the distribution of their allomorphs cannot be acquired without the phonological capacity to extricate them from the flood of sounds in every sentence". To develop this capacity learners first have to notice the contrast between their own erroneous realization (output) and the target form (input), as explained in Schmidt's Noticing Hypothesis (Schmidt, 1990). Difficulties in learning Dutch verbal morphology are related to perception and production of L2 phonemes such as schwa and /t/. As to perception, it is crucial to perceive the differences in (1) in order to understand the Dutch agreement paradigm, and in (2) in order to understand the tense system (present vs. past tense).

(1) /maak/, /maakt/, /make(n)/.
(2) /maakt/, /maakte/.
On the production side, difficulties in pronouncing certain sound combinations may lead a Moroccan learner to say (3) when trying to pronounce /loopt/.
(3) /lopet/, /loopte/.

In syntax problems have been observed with word order, finite verb position, and pronominal subject omission. In order to extend ASR-based feedback to morphology and syntax it is necessary to design exercises that are appropriate for practicing these aspects of spoken proficiency on the one hand, but that are controlled enough to be handled by ASR. For pronunciation it is possible to use imitation and reading exercises and these can be handled by ASR because the vocabulary is known in advance. For morphology and syntax such exercises cannot be used because learners then have no freedom to show whether they are able to produce correct forms. So, the exercises that are required have to be such that they allow some freedom to the learners in formulating answers, but that are predictable enough to be handled by ASR. For instance, suitable exercises can be designed by stimulating students to produce utterances containing the required morphological and syntactic forms by showing them words on the screen, without declensions, or pictograms, possibly in combination with stills or short video fragments representing scenes (e.g. a girl reading a book). In addition, as in *Dutch-CAPT*, use can be made of dialogues and scenarios illustrating so-called "crucial practice situations", which

correspond to realistic situations in which learners might find themselves in Dutch society and in which they have to interact with other citizens. These CPSs form the basis of the various civic integration examinations. The students can be asked to play a certain dialogue by using simple prompts concerning the vocabulary to be used and they have to formulate the correct sentences themselves.

The idea of extending the *Dutch-CAPT* approach to morphology and syntax was elaborated in a research proposal named DISCO (Development and Integration of Speech technology into COurseware for language learning), which was eventually financed within the framework of the HLT stimulation programme STEVIN, which is funded by the Dutch and Flemish governments. The aim of the DISCO project is to develop a prototype of an ASR-based CALL application for Dutch as a second language (DL2). The application optimizes learning through interaction in realistic communication situations and provides intelligent feedback on important aspects of DL2 speaking, viz. pronunciation, morphology, and syntax.

9. Conclusions

In this paper, we have presented a system for providing automatic corrective feedback on pronunciation errors in Dutch, focusing particularly on error detection, scoring accuracy, and feedback effectiveness. We have shown that, while this system did not achieve 100% accuracy in error detection, it was still effective in improving pronunciation errors after only a few hours of use over a period of one month, and learners enjoyed using it. It is our opinion that these positive results were achieved by fine-tuning the ASR-technology to specific pedagogical goals and by adopting a user-centered approach to implement the feedback presentation.

These results justify deploying ASR technology for pronunciation training, and make it worthwhile to study how this technology could be applied to the training of other speaking skills such as morphology and syntax.

Acknowledgements

The present research was supported by the Dutch Organization for Scientific Research (NWO). We are indebted to Ming-Yi Tsai, F. de Wet, M. Hulsbosch, L. ten Bosch, C. van Bael, J. Kerkhoff, L. Boves, and A. Russel for their help building *Dutch-CAPT*, to L. Aul and S. Kanters for the analyses of scoring accuracy, and to Malmberg Publishers for allowing us to use the material from *Nieuwe Buren*. We would like to thank two anonymous reviewers for their useful comments.

References

Carey, M., 2002. An L1-Specific CALL Pedagogy for the Instruction of Pronunciation with Korean Learners of English. Unpublished Doctoral Dissertation, Macquarie University, Australia.

Cucchiarini, C., Strik, H., Boves, L., 2000. Quantitative assessment of second language learners' fluency by means of automatic speech recognition technology. J. Acoust. Soc. Amer. 107, 989–999.

DeKeyser, R., 2005. What makes learning second-language grammar difficult? A review of issues. Language Learning 55, 1–25.

Egan, K., 1999. Speaking: a critical skill and challenge. CALICO J. 16, 277–293.

Ehsani, F., Knodt, E., 1998. Speech technology in computer-aided learning: strengths and limitations of a new CALL paradigm. Language Learning Technol. 2, 45–60. <http://llt.msu.edu/vol2-num1/article3/index.html> (retrieved 05.27.08).

El-Tatawy, M., 2002. Corrective Feedback in Second Language Acquisition. Working Papers in TESOL & Applied Linguistics. Teacher's College, Columbia University. <http://journals.tc-library.org/index.php/tesol/article/view/160/158> (retrieved 05.30.08).

Giuliani, D., Mich, O., Nardon, M., 2003. A study on the use of a voice interactive system for teaching English to Italian children. In: Proc. IEEE Internat. Conf. on Advanced Learning Technologies, Athens, Greece, pp. 376–377.

Herron, D., Menzel, W., Atwell, E., Bisiani, R., Daneluzzi, F., Morton, R., et al., 1999. Automatic localization and diagnosis of pronunciation errors for second-language learners of English. In: Proc. Eurospeech, Budapest, Hungary, pp. 855–858.

Lennon, P., 1990. Investigating fluency in EFL: a quantitative approach. Language Learning 3, 387–417.

Mak, B. Siu, M., Ng, M., Tam, Y.-C., Chan, Y.-C., Chan, K.-W., et al., 2003. PLASER: pronunciation learning via automatic speech recognition. In: Proc. HLT-NAACL 2003 Workshop on Building Educational Applications using Natural Language Processing, Edmonton, Canada, pp. 23–29.

Neri, A., Cucchiarini, C., Strik, H., Boves, L., 2002. The pedagogy-technology interface in computer assisted pronunciation training. Comput. Assist. Language Learning 15, 441–467.

Neri, A., Cucchiarini, C., Strik, H., 2006. Selecting segmental errors in L2 Dutch for optimal pronunciation training. Internat. Rev. Appl. Linguist. 44, 357–404.

Oostdijk, N., 2002. The design of the spoken Dutch corpus. In: Peters, P., Collins, P., Smith, A. (Eds.), New Frontiers of Corpus Research. Rodopi, Amsterdam, pp. 105–112.

Reeser, T.W., 2001. Calico Software Review. Review of Tell Me More French. CALICO Rev. <http://calico.org/p-63-Tell%20Me%20More%20-%20French%20(32001).html> (retrieved 05.30.08).

Schmidt, R.W., 1990. The role of consciousness in second language learning. Appl. Linguist. 11, 129–158.

Van Bael, C.P.J., Binnenpoorte, D., Strik, H., van den Heuvel, H., 2003. Validation of phonetic transcriptions based on recognition performance. In: Proc. Eurospeech, Geneva, Switzerland, pp. 1545–1548.

Witt, S.M., 1999. Use of Speech Recognition in Computer-Assisted Language Learning. Doctoral Dissertation, Department of Engineering, University of Cambridge, Cambridge, UK.

Witt, S.M., Young, S.J., 2000. Phone-level pronunciation scoring and assessment for interactive language learning. Speech Comm. 30, 95–108.

Young, S., Kershaw, D., Odell, J., Ollason, D., Valtchev, V., Woodland, P., 2000. The HTK Book Version 3.0. Cambridge University, Cambridge.

Zheng, T., 2002 Calico Software Review. Tell Me More Chinese. CALICO Rev. <http://calico.org/p-112-Tell%20Me%20More%20Chinese%20(112002).html> (retrieved 05.30.08).