PRONUNCIATION MODELING AND LEXICAL ADAPTATION USING SMALL TRAINING SETS

Louis ten Bosch (1), Nick Cremelie (2)

(1) University of Nijmegen, The Netherlands (2) ScanSoft BVBA., Merelbeke, Belgium
l.tenbosch@let.kun.nl, nick.cremelie@scansoft.com

ABSTRACT

A method for data-driven lexical adaptation on the basis of a limited number of acoustic training tokens is discussed. The method is closely related to pronunciation modeling techniques. A set of pronunciation variants is generated by forced alignment, followed by a step to select promising pronunciation candidates by using a ranking function. The method has been validated on a database consisting of short utterances (proper names) spoken by native and non-native speakers. In the case of 5 training tokens per word, an improvement of 10-30 percent relative could be obtained compared to the baseline. A number of possible improvements of this method are discussed as well.

1. INTRODUCTION

As is well-known, a large number of factors influence the acoustic realization of speech sounds. One important factor is speaking style, ranging from carefully spoken isolated words to highly reduced spontaneous speech. Another relevant factor is the dialectal background of the speaker or, in case of a non-native speaker, the mother language (L1) background. As a result of these factors, speech is characterized by showing a substantial amount of variation in pronunciation of words. The adequate treatment of these pronunciation variation forms one of the substantial challenges for automatic speech recognition (ASR) systems. In recent years, when the techniques of acoustic modeling and language modeling became sufficiently mature, the modeling of pronunciation variation has been paid more attention to, as testified by the number of recent workshops and papers on this topic (Strik & Cucchiarini, 1999; Amdall et al., 2000; Saraclar et al, 2000, Wester, 2000).

Pronunciation modeling (PM) is a broad research field and the involved techniques may serve different goals. Firstly, PM can be applied to improve the quality of the (phoneme) segmentation that is obtained by aligning the speech signal with ASR models (Wester et al., 2001; Kessens, 2002). Second, PM is potentially useful for generating phonetic transcriptions of words to improve ASR recognition performance (Saraclar et al., 2000). PM has shown to yield an improvement of the recognition rate of an ASR system in specific cases (e.g. for non-native test speakers: Cremelie & ten Bosch, 2001). However, for general recognition of spontaneous or conversational speech the reported gain is only moderate (Ma et al., 1998; see also Jurafsky et al., 2001). Third, techniques related to PM prove their usefulness is the area of phonetic research, to extract information from the speech signal about the application frequency of phonetic/phonological rules (e.g. Kessens, 2002).

It appears quite difficult to define one *general* method that is applicable to all these PM-related problems. Pronunciation modeling deals with the complex interactions between the utterance, acoustic models and transcriptions, and is closely related to minimization of between-word confusions, the richness of the set of pronunciation variants, and, in some applications, to the language model.

In this paper, pronunciation modeling is understood as the data-driven improvement of the phonetic transcriptions of words in a lexicon for automatic speech recognition. The method is based on N-best hypotheses from a forced alignment using a phoneme lattice with weighted arcs, without retraining the acoustic models. This 'lexical adaptation' is data-driven and based on a limited number of acoustic realizations of a word ('training tokens'). The acoustic training tokens and the canonical transcriptions of the considered words are the input of the method. The method derives candidate alternative word pronunciations from the training tokens, with the goal to achieve a better ASR performance.

The focus of the method that will be described here is on tasks where the goal is to recognize isolated words or short utterances (e.g. full names or locations) spoken by native as well as non-native speakers. Apart from the usual factors determining pronunciation variation for native speakers, the speech from non-native speakers can deviate from the native pronunciation by two additional factors. Firstly, limited language proficiency may lead to decreased ability to guess the correct pronunciation from the orthography, resulting in aberrant grapheme-phoneme mappings. The second additional factor is related to the acoustic difference between phone sets. The non-native speaker will try to pronounce the 'xeno-phones' by using similarly sounding phones that occur in his/her native language, thereby implicitly using a sort of similarity measure between sounds.

In summary, it is clear that the targeted task is indeed in need of some sort of pronunciation modeling to cope with the non-native pronunciations. As such, improving the recognition of non-native speakers is the goal of our lexical adaptation method.

The organization of this paper is as follows. In the next section, we will describe an overview of the adaptation method. In the third section, a more detailed description of the algorithm for generating and selecting pronunciation variants is discussed. Section 4 presents N-best results, whereas section 5 deals with the ASR experiments and results. The final section concludes with a discussion.

2. SET-UP OF THE ADAPTATION METHOD

Our lexical adaptation procedure is based on a data driven pronunciation modeling principle. It consists of two steps. In the variant *generation* step, the acoustic tokens are used to produce numerous candidate pronunciation variants. In the *selection* step, the number of variants is pruned to the ones that are most promising in an independent test. Several issues are to be addressed:

1. How to deal with (phoneme) insertions, substitutions and deletions? What is the status of the canonical transcriptions?
2. How to select the promising candidate variants? To what extent are acoustic scores, ranking positions or confidence measures useful in a selection criterion?

In the current approach, pronunciation modeling will focus on improvements at the word level. This means that the generation and pruning of pronunciation variants operate on the word level. For each word do be adapted, the method takes its canonical transcription plus a number of acoustic training tokens (i.e. utterances of the word) as a starting point.

The database that is used for our experiments consists of names. The names are combinations of a first name (which is optional) and a family name, both from English, French or Flemish/Dutch origin. The speakers have either a Dutch or French language background. The utterances in the database are logged recordings (telephone quality) from an automated attendant application in operation. A subset of the data is used for training. The number of acoustic training tokens that were used for the lexical adaptation is a parameter in the procedure.

The method to be presented here aims at finding phonetic transcriptions of a name (a combination of an optional first name followed by a family name), based on a limited number of acoustic tokens of that name, such that the adapted phonetic transcriptions show an improved match on an independent test set by non-native speakers of the same language background as the one of the training speakers. The canonical transcriptions of the names in the lexicon ('bootstrap lexicon') are assumed to more or less cover the native pronunciations of the words. The pronunciation modeling is done for the first names and the family names separately, as separate recognition units.

3. DESCRIPTION OF THE ALGORITHM

In this section, the steps followed in the adaptation method are explained in more detail.

1. Graph expansion. For each word, the canonical transcriptions as defined in the bootstrap lexicon define a word-dependent decoding graph. Each node in this graph is associated with a phone model. In theory, this phone model can be context-dependent or context-independent, but in the current approach the phone graph as well as the canonical transcriptions are based on context-independent models. This graph can be extended to allow substitutions, insertions, and deletions at phone level.

In the current approach, substitutions are modeled in the decoding graph by expanding each phoneme into a parallel set of similarly sounding phonemes. These acoustically 'neighboring' phonemes are based on a predefined table of broad phonetic classes (BPC). A small part of this table is presented below:

```
A a
A~ A a
E E: e
p t b k
t d p k
d t g b
…
```

This table fragment indicates that the short vowel phone [A] has only one neighboring phone [a], the nasalised short vowel [A~] has two neighbors [A] and [a], and so on.

The neighborhood relation between phones is of course reflexive, but neither necessarily symmetric nor necessarily transitive. The neighborhood relation is phonetically inspired, and based on phonetic similarity between phones by using phonetic features. Usually, all phones that are 'acoustically similar' to a certain phone are listed among the neighbors of that phone. For example, the neighbors of the voiceless velar stop [t] are its voiceless counterparts [p], [k], as well as its voiced velar counterpart [d]. (An alternative data-driven way to construct the decoding graph would be to use the similarity of phones based on the distances between trained acoustic context-independent (HMM-)models, applying techniques comparable to many tying schemes in ASR approaches).

Phone deletions and insertions can be modeled in a similar fashion, by adding arcs and nodes in the phoneme graph. However, to avoid a large number of arcs, skips of one phone at the time were allowed, and insertions were not taken into account by default.

In the resulting phoneme graph, each arc has been allotted a weight. Arcs modeling a substitution are associated with a substitution penalty, and deletion penalties have been added to each phone skip. The values of these penalties are parameters of the PM method, but reasonable lowerbounds and upperbounds can be derived from the collection of acoustic scores of phone models obtained by forced alignment.

In the current approach, context-dependent transcriptions were not taken into account, mainly because of the tremendous size of the resulting pronunciation graph for a word of medium phone length (in the order of 100000 arcs). For comparison, a ci-graph modeling substitutions and deletions for a word of 6 phones may contain about 200 arcs.

2. Generation of variants. Pronunciation variants of a word are generated by a forced alignment of a (small) number of acoustic training tokens of the word with the expanded word graph, and extracting the N-best list of resulting phone sequences as well as acoustic scores on word and phone level. This step needs a rich combination of dynamic programming features in the forced alignment procedure: N-best, trace-back (phone level), the ability to produce acoustic scores at word level, the ability to mark word boundaries and word identities in the phone back-

trace output string, and the possibility of penalizing specific arcs of the decoding phone graph in a flexible way.

The actual number M of acoustic training tokens that are aligned with the corresponding pronunciation graphs is a parameter in the method, as is the depth N of the N-best lists. In the current approach, N is equal to 400, and values of M between 3 and 10 have been considered. The result of this step is a two dimensional matrix ({acoustic token} x {pronunciation variant}) called a "tableau", in which the cells are filled with phonetic variants and the associated acoustic scores. Each column in this tableau corresponds with the N-best list of variants obtained for that token. In a natural way, a ranking index (0 for the best candidate, 1 for the next, etc.) can be associated to each instance in a column.

An example of a tableau is presented below.

Token1	Token 2	...
'bot' (542.1)	'bid' (613.0)	
'bod' (548.3)	'bid' (620.4)	
'bod' (559.2)	'bid' (630.8)	
'bot' (561.7)	'bod' (639.2)	
...	...	

3. Ranking. For each token, a particular variant (such as 'bid' in the second column) may evidently occur more than once in the corresponding column – in such a case, the corresponding alignment and score will in general be different. The goal of the third step in the algorithm is to make one single ranked list of all pronunciation variants that occur in the tableau, on the basis of their acoustic scores and ranking indices.

A number of remarks can be made. Firstly, for an individual transcription variant, one could assume that the average acoustic score (averaged across all instances of this transcription variant in the tableau) is a reasonable measure for the acoustic match between the transcription variant and the articulatory-acoustic variation represented by the tokens. However, this is not entirely true, since these acoustic scores also depend on the segmentation associated with the alignment, and therefore indirectly depend on the LM. Secondly, compared to the acoustic scores themselves, the ranking indices may provide a less accurate, but certainly less elusive measure for the quality of a particular variant compared to other candidate variants. Thirdly, what essentially matters in a ranking criterion is the relation between the scores and eventually the performance of the top ranked variants in an independent ASR test.

An accurate but expensive way of testing a ranking criterion would be an exhaustive procedure: for M training tokens, create N-best lists, for various values of deletion and substitution penalties. From these lists, collect all (promising and less promising) pronunciation variants (based on ranking, scores etc.), and test them in an independent test. Next, find a correlation between the ranking index, average acoustic scores, number of instances etc. and the resulting test accuracy numbers on a word-by-word basis. Evidently, such a procedure is infeasible, and we therefore have cut down the search effort by doing a number of preliminary experiments to figure out the effects of radical choices at an early stage. As a result, we have come to a ranking function of a particular type that generates a shortlist of "best guess variants" from the tableau. We observe that there are a number of choices to be made here: as observed earlier, an N-best list per token is likely to contain multiple instances of the same variant – with possibly different acoustic scores –, so a proper definition of the 'average' score of a variant across tokens is required. Prior experiments have shown that the ranking index in the N-best list is much more relevant than the acoustic scores themselves, and that for the eventual ranking of each pronunciation variant, the ranking of the *winning* instance (i.e. the highest ranked instance for that token) of that particular variant is of special interest (see also ten Bosch & Cremelie, 2001). As a consequence, the ranking function takes the following 'ranking parameters' as inputs: (a) the number of training tokens (Nocc) for which the variants occurs at all in the N-best lists (b) the average ranking (Rbest) in the N-best lists of the best instance of the considered variant, counting the top position as 0, and (c) the average ranking (Rbest_rel) of the best instance *among all best instances*, counting the top position as 0.

The following example will clarify the procedure. We take the previous example as a starting point. In this example we have just two acoustic training tokens available. The length of the N-best list for each token equals 4. So M = 2 and N = 4, and we obtain the following tableau:

Rank	Token 1	Token 2
0	'bot'	'bid'
1	'bod'	'bid'
2	'bod'	'bid'
3	'bot'	'bod'

The resulting ranking results read:

	Nocc	Rbest	Rbest_rel
bot	1	0	0
bod	2	2	1
bid	1	0	0

Observe the difference for the variant 'bod' in the case of Rbest and Rbest_rel. In the former case, the resulting ranking is based on the absolute ranking indices 1 and 3, respectively. In the case of Rbest_rel, the resulting ranking is based on 1 and 1, since in the second column 'bod' has ranking index 1 after the best instance of 'bid' which has ranking index 0.

In the current approach, we used a ranking function based on the number of tokens for which the variant is figuring somewhere in the N-best list (Nocc) and the relative ranking of the best instance of the variant, i.e. ranking parameter Rbest_rel. The larger parameter Nocc is, the more promising the corresponding variant will be. The reverse is true for larger values of parameter Rbest_rel. Hence, we propose as a ranking function

$$\text{Rank} = \text{WF}*\text{Nocc} - \Sigma\, \text{Rbest\_rel}/\text{Nocc}$$

the sum being taken over all tokens, and WF denoting a weighting factor. This function weights the contribution of Nocc and Rbest_rel in such a way that Rank can be considered to be the expected rank of a transcription for an arbitrary token, under the assumption that the shape of the ranking distribution of a particular transcription is invariant except for its mean. The parameter WF is related to a penalty that is associated with a

default ranking cost in the case when a particular transcription does not occur in an N-best-list. By applying the ranking function, each pronunciation variant occurring anywhere in the N-best lists receives a final ranking score. This construction allows implicit correction of ranking scores for variants that do not occur in an N-best list.

4. Selection. In the final step, the 'most promising' pronunciation candidates are selected based on their ranking scores obtained in step 3. After ranking, only the topN candidates with the highest rank are considered and put into a new, updated test lexicon. In the current approach, this number topN is fixed across words for practical reasons only, since a word-dependent number would introduce at least another additional threshold., Nevertheless, we believe that taking a variable number of variants per word through a more elaborated selection procedure could further improve the method, especially if the selection procedure would include techniques to trace and avoid cross-word confusion, possibly even taking the structure of the application's syntax into account.

Steps (2) and (3) are the most critical ones. An important parameter in step (2) is the choice of penalization of deletions and substitutions in the arcs of the decoding graph, before creating the N-best lists. During our experiments, it appeared to be favorable to start with an initial value of the deletion penalty equal to infinity (i.e. allowing no deletions), in order to start with relatively clean N-best lists. Furthermore, the more acoustic tokens are available for training, the lower the penalties for deviant paths in de decoding graph can be: for small number of training tokens, higher penalties will be necessary to avoid spurious transcription intruders to appear in the top of the resulting list of candidates.

4. N-BEST RESULTS

In this section we will discuss a number of observations based on the results that were obtained after the N-best forced alignment step. After step 1 (generation of the decoding phone graph) and step 2 (forced alignment and N-best extraction) the first observation is that N-best results on the phone level are quite scattered even for moderate values of the substitution penalty. The larger this penalty, the closer the N-best list resembles the list of canonical transcriptions that were used as starting point in step 1, but it is not necessarily true that the variants emerging for slightly lower values of the substitution penalty are the most interesting non-canonical ones. The more parallel branches in the pronunciation graph, the more the N-best list gets 'polluted' by many alternatives, and the probability that one *single* pronunciation candidate stands out in all N-best lists will be very low compared to the case where the pronunciation graph just allows a few variants. In other words: the N-best list gets many alternative variants of which a large number are not the variants with a good ranking for other tokens. In this sense, longer words get more polluted than shorter words, and, in theory, the substitution penalty should be accordingly tuned *upwards* with word length to avoid too many scattered results for the longer words. As a basic rule of thumb, on the basis of alignment experiments the optimal value for the substitution penalty was found to be between 20 to 40 percent of the *average frame-state alignment score* (measured in speech) for short words of a transcription length of 4-5 context-independent phones; for words with a length of about 8 phones and more, this penalty is to be chosen a factor of 2-3 larger. (Evidently these numbers also depend on the quality of the acoustic models.)

An example of the pollution is presented in table I. The table is based on one word 'Kandinsky' with canonical transcription [kA~dE~ski]. The phonetic transcriptions are SAMPA (short vowels are indicated by capitals, the tilde represents nasalization). The table is a compilation of five N-best lists, derived from five acoustic realizations of 'Kandinsky' (M equals 5, N equals 400). It gives only a fraction of all variants observed: for the word Kandinsky, over 1000 pronunciation variants emerge. The canonical solution is marked by a <-sign. For each variant, the three numbers at the end of each line are used to rank the corresponding pronunciation variant in step 3. The first two numbers are related to the ranking position in the N-best lists (these number being the sum of the ranking indices related to Rbest and Rbest_rel, respectively); the third number represents Nocc which is the number of N-best lists in which the particular pronunciation variant occurs. All the variants shown in the table occur at least once in each of the five N-best lists. One observes that the canonical transcription is relatively highly ranked (the sum of the ranks being equal to 8) across the 5 N-best lists..

The observed pollution is a normal phenomenon, since the decoding graph allows a large number of degrees of freedom compared to a small number of canonical transcriptions. In theory, the penalties on deviant arcs could be trained on a corpus to circumvent too many aberrant solutions. Such a training, however, would take a large amount of training data.

Table I. Example of pollution in the N-best lists obtained by a forced alignment on phone level for 5 acoustic tokens of the word 'Kandinsky'. Only the variants are shown that occur at least once in *each* N-best list. For a further description see the text.

```
g  A~  d  E~  f  k  i 409 409 5
g  A~  d  E~  s  k  i 241 241 5
g  A~  d  E~  s  t  i 862 862 5
k  A~  d  E~  f  k  i 311 311 5
k  A~  d  E~  s  k  i 157 157 5
k  A~  d  E~  s  t  i 725 725 5
k  A~  g  E~  f  k  i 635 635 5
k  A~  g  E~  s  k  i 374 374 5
k  A~  b  E~  f  k  i 199 199 5
k  A~  b  E~  s  k  i 85 85 5
k  A~  b  E~  s  t  i 585 585 5
k  A~  d  E~  f  g  i 701 701 5
k  A~  d  E~  f  k  i 16 16 5
k  A~  d  E~  f  p  i 413 413 5
k  A~  d  E~  f  t  i 126 126 5
k  A~  d  E~  s  k  i 8 8 5 < canonical
k  A~  d  E~  s  p  i 301 301 5
k  A~  d  E~  s  t  i 81 81 5
k  A~  d  E~  z  k  i 530 530 5
k  A~  g  E~  f  k  i 124 124 5
k  A~  g  E~  f  t  i 462 462 5
k  A~  g  E~  s  k  i 45 45 5
k  A~  g  E~  s  t  i 346 346 5
k  A~  t  E~  f  k  i 362 362 5
k  A~  t  E~  s  k  i 148 148 5
k  A  b  E~  s  k  i 904 904 5
```

```
k  A  d  E~  f  k  i 813 813 5
k  A  d  E~  s  k  i 371 371 5
k  A  g  E~  s  k  i 714 714 5
p  A~  d  E~  f  k  i 830 830 5
p  A~  d  E~  s  k  i 526 526 5
p  A~  b  E~  f  k  i 1098 1098 5
p  A~  b  E~  s  k  i 606 606 5
p  A~  d  E~  f  k  i 201 201 5
p  A~  d  E~  f  t  i 582 582 5
p  A~  d  E~  s  k  i 102 102 5
p  A~  d  E~  s  t  i 516 516 5
p  A~  g  E~  f  k  i 761 761 5
p  A~  g  E~  s  k  i 426 426 5
t  A~  d  E~  f  k  i 272 272 5
t  A~  d  E~  f  t  i 640 640 5
t  A~  d  E~  s  k  i 175 175 5
```

5. ASR EXPERIMENTS AND RESULTS

As mentioned before, the Automated Attendant data consist of utterances with combinations of an (optional) first name followed by a family name, spoken by native and non-native speakers. The number of name combinations (types) is 140. From this material, two sets have been selected, one serving as training set and the other as independent test set. The training set contains 400 utterances: 10 tokens of 40 different name combinations, spoken by randomly selected speakers. The development and evaluation test databases also contain 10 tokens (but different from the training tokens) from the same 40 names, spoken by a random selection of speakers. The *test lexicon* was a plain combination of two sub-lexicons: one containing the 40 words that underwent the improvement scheme together with their automatically derived pronunciation transcriptions, the other sub-lexicon containing *all* other names with their original (canonical) transcriptions (210 entries total). The second sub-lexicon (unaltered across all experiments) was added to render the recognition task realistic with regard to perplexity.
Examples of the first and family names that were subject to adaptation are presented in table II. Most names have 4 phones or more.

Table II. Examples of first names and family names used in test

```
First names:
Andre, Brigitte, Christophe, Eddy,
Francoise, Jeroen

Family names:
Goossens, Lemaire, Maes, Menard, Moons,
Sabbe, Vogel
```

The acoustic data have been recorded at 8 kHz sample frequency. A cepstral mean subtraction has been applied on utterance level. The acoustic models that are used for alignment are context-independent.
For each adaptation token, N-best list were constructed according to the method described above. Different values were investigated for the most important model parameters:

1. the penalties used to weight the substitution and deletion arcs in the phone decoding graph
2. M: the number of acoustic training tokens used
3. WF: the weighting factor between the two arguments in the ranking function
4. topN: the number of variants included in the test lexicon

In the sequel, we will discuss results that are obtained with M set equal to 5. The optimal values for the remaining parameters have been found by systematic exploration of combinations on the training set. The optimal arc penalty for substitutions as well as for deletions has a range between 20 and 40 percent of the average alignment cost per frame – the optimal value is related to the rate of 'pollution' observed in the resulting N-best lists. The optimal number of variants (topN) to be included in the test lexicon was found to range between 3 and 5 with a good practical value equal to 4. The optimal value of the weighting factor WF is between 25 and 50 (WF should be decreased when M increases).

Table III. ASR performance (word accuracy) on the test set as a function of the weighting factor WF between Nocc and Rbest_rel in the ranking function (shown along the rows) and the number of pronunciation variants topN in the lexicon (along the columns). All numbers are percentages (%).

WF	1	2	3	4	5	6
0	67.3	70.9	74.5	77.0	78.2	77.6
10	70.3	73.3	77.0	77.6	78.2	78.2
25	70.3	73.3	76.4	77.6	78.2	78.2
50	71.5	74.5	77.0	78.8	78.8	78.8

Table III presents results on the test set for a number of combinations of two parameters: the parameter WF used in the ranking function to balance between the two ranking parameters Nocc and Rbest_rel defined earlier, and the number of variants (topN) included in the test lexicon. The baseline is presented in the left upper cell (67.3 percent), which is a result based on the canonical lexicon (containing one variant per word and modeling native canonical pronunciations). In almost all cases this winner in the candidate list equals the canonical transcription.

6. DISCUSSION

The results in table II show that, in the case of non-native speakers pronouncing foreign name combinations, the baseline performance can be improved substantially by a proper selection criterion of variants and by adding more variants in the test lexicon. Even in the case where these variants are based on the ranking of N-best hypotheses from just 5 acoustic training tokens, the improvement can be substantial (on average 25 percent, between 10-30 percent on word level).
The absolute accuracy seems quite low. However, these results are based on context independent acoustic models and are speaker independent. No speaker adaptation has been performed. Moreover, the acoustic training and test tokens have not necessarily been uttered by the same speaker. The perplexity of the test is approximately 210. No (i.e. a flat) language model was applied.
By exhaustive search of variant combinations, we could obtain a performance of about 83 percent, but unfortunately no search algorithm has been found that is able to find the winning

solution on the basis of a few acoustic training tokens. It is implausible whether such an algorithm exist.

The current method can be improved. The number of variants topN that is included in the test lexicon is now fixed across words. The associated selection criterion can be refined by taking into account the between-word confusion probability, and syntactical constraints.

Acknowledgement

The research that led to this paper was performed when both authors were at Lernout & Hauspie Speech Products NV, Ieper, Belgium

REFERENCES

Amdall, I., Korkmazskiy, F. & Surendran, A. (2000). Joint pronunciation modeling of non-native speakers using data-driven methods. *Proc. ICSLP 2000, Beijing*. Vol. 3, pp 622-625.

Bosch, L. ten, & Cremelie, N. (2001). Pronunciation modeling and lexical adaptation in mid-size vocabulary ASR. *Proc. Eurospeech 2001*, Aalborg, Denmark.

Cremelie, N., and Martens, J.P.(1998. In search of pronunciation rues. In: Strik, H., Kessens, J.M., Wester, M. (eds.) Proceedings of the ESCA Workshop on Modeling Pronunciation Variation for Automatic Speech Recognition, Rolduc, Kerkrade, May 1998. University of Nijmegen, pp. 23-27.

Cremelie, N. & ten Bosch, L. (2001). Improving the Recognition of Foreign Names and Non-Native Speech by Combining Multiple Grapheme-to-Phoneme Converters. *Proc. ISCA ITRW Workshop Adaptation Methods For Speech Recognition*, Sophia-Antipolis, France. Pp. 151-154.

Jurafsky, D. et al. (2001). What Kind of Pronunciation Variation is Hard for Triphones to Model? *Proc. ICASSP-01*, vol. I., pp. 577-580.

Kessens, J. M. (2002). Making a difference. On automatic transcription and modeling of Dutch pronunciation variation for automatic speech recognition. PHD thesis, University of Nijmegen, 2002. (ISBN 90-9015829-4).

Ma, K., Zavaliagkos, G. & Iyer, R. (1998). Pronunciation modeling for large vocabulary conversation speech recognition. *Proc. ICSLP 1998,* Sydney. Vol. 6, pp. 2455-2458.

Saraclar, M. & Khudanpur, S. (2000). Pronunciation ambiguity vs. pronunciation variability in speech recognition. *Proc. ICASSP 2000,* Istanbul, pp. 1679-1682.

Strik, H. & Cucchiarini, C. (1999). Modeling pronunciation variation for ASR: A survey of the literature. *Speech Communication* 29, 225-246.

Wester, M. Fosler-Lussier, E. (2000). A comparison of data-derived and knowledge-based modeling of pronunciation variation. Proceedings ICSLP, Beijing, Ocober 2000. Vol. 4, pp. 270-273.

Wester, M., Kessens, J., Cucchiarini, C. & Strik, H. (2001). Obtaining phonetic transcriptions: A comparison between expert listeners and a continuous speech recognizer. *Language and Speech*, 2001, 44(3), pp 377-403.