

Unit Selection in Concatenative TTS Synthesis Systems Based on Mel Filter Bank Amplitudes and Phonetic Context

T. Lambert¹, A.P. Breen², B. Eggleton², S.J. Cox¹ and B.P. Milner¹

¹University of East Anglia, Norwich

²Nuance Communications, Norwich

t.lambert@uea.ac.uk, abreen@nuance.com, barry.eggleton@nuance.com,
s.j.cox@uea.ac.uk, b.milner@uea.ac.uk

Abstract

In concatenative text-to-speech (TTS) synthesis systems unit selection aims to reduce the number of concatenation points in the synthesized speech and make concatenation joins as smooth as possible.

This research considers synthesis of completely new utterances from non-uniform units, whereby the most appropriate units, according to acoustic and phonetic criteria, are selected from a myriad of similar speech database candidates. A Viterbi-style algorithm dynamically selects the most suitable database units from a large speech database by considering concatenation and target costs. Concatenation costs are derived from mel filter bank amplitudes, whereas target costs are considered in terms of the phonemic and phonetic properties of required units.

Within subjects and between subjects ANOVA [9] evaluation of listeners' scores showed that the TTS system with this method of unit selection was preferred in 52% of test sentences.

1. Introduction

In recent years, unit selection-based concatenative TTS synthesis systems have become the focus of attention of many researchers in the speech synthesis field.

Researchers have long been trying to improve the naturalness of the synthesized speech by increasing the length of basic units used for concatenation, from demi-phones [6], diphones [1], triphones, syllables, words to variable length or non-uniform units [2]. The unit selection process in current speech synthesizers is based on some type of dynamic programming [10] that selects from the speech database units with minimal cost functions. Selection of units from large speech corpora was achieved by considering target and concatenation cost functions [6, 1]. Past research considered unit selection using only linguistic and phonological data [2, 7] or a combination of phonological and acoustic signal information [1, 4]. Mel frequency cepstral coefficients have been most widely used as representatives of the signal's acoustic data. Some researchers have also concentrated on integrating prosodic characteristics of speech segments into the unit selection process [8].

In concatenative TTS systems using large speech corpora the main difficulties for the unit selection process are: selecting a desired unit from a myriad of similar units and joining the selected units in the way that is most consistent with acoustic

and auditory perception of speech. The most important factors for unit selection are: distance measures used for capturing linguistic information of the target input text and acoustic criteria of speech signals stored in the database, search algorithm and the speech database. This research concentrates on the first two factors.

This paper is organized as follows: section 2 describes cost functions used in the proposed unit selection. The system design is given in section 3 and the unit selection method is explained in section 4. The ANOVA experimental evaluation is detailed in section 5. Finally, sections 6 and 7 give conclusions and considerations for future work.

2. Cost Functions

Cost functions often employed in unit selection algorithms are detailed in [1, 3, 4].

In this research, unit selection considers target and concatenation costs using acoustic and phonetic information of data stored in the speech database.

For any target input text $\{t_1, t_2, t_3, \dots, t_n\}$ target costs represent a degree of similarity between target phonemes in the input text, t_i , and available units in the speech database, u_i . Concatenation costs measure the smoothness of the concatenation join between successive speech units selected from the database. Unit selection costs, which seek to minimize the sum of total costs, C_{total} , across a sequence of n target phonemes, are therefore given by the equation (1):

$$C_{total} = \sum_{i=1}^n C^t(t_i, u_1) + \sum_{i=2}^n C^c(u_{i-1}, u_1) \quad (1)$$

where, C^c are concatenation costs and C^t are target costs.

Concatenation costs between successive candidate units, u_i and u_{i-1} are derived from acoustic feature vectors containing twenty mel filter bank amplitude channels. Phonemic identity and phonetic neighbouring context are employed in target costs in order to find a database unit, u_i , which is most similar to synthesizable target unit, t_i . For each phoneme in the target input text concatenation and target costs are used to: find a pool of candidate units from the entire speech database, and select from the pool of candidate units speech segments that join at the lowest total costs as shown in Figure 1.

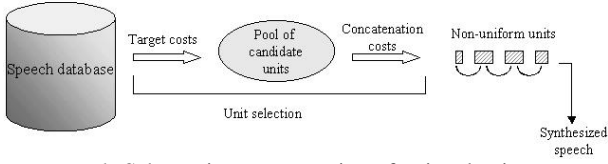


Figure 1: Schematic representation of unit selection costs

3. The System Design

The underlining speech database consists of de-contextualized recorded speech by a professional female North American speaker. The extensive database coverage ensures that at least one database candidate unit for every target phoneme can be found in the database.

3.1 Phoneme Context Tree

Speech units are organized in a phoneme context tree similar to the design used in [2].

The context tree has three levels – level 0 contains phone units; levels 1 and 2 hold units with wider phonetic context in relation to phones at level 0. Level 1, holds triphones (central phone with left and right neighbouring phones). Level 2 widens the context of triphones from level 1 by storing one extra phone on either side of the triphone i.e. a central phone, two phones preceding and two phones following the central phone. Figure 2 is a graphical representation of a small section of the phoneme context tree.

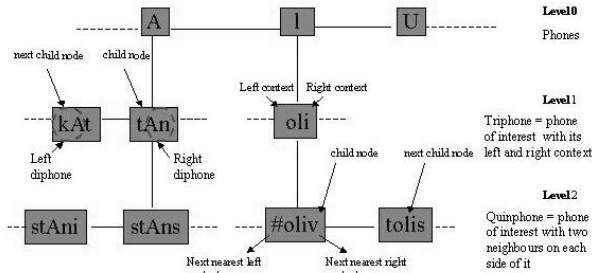


Figure 2: Organization of Speech Units in the Context Tree

Each tree node in the phoneme context tree holds acoustic, prosodic and phonetic feature vectors associated with individual phone segments.

4. Unit Selection

The objective of the unit selection method is to extract and select speech units from the speech database that best match the phonetic and prosodic environment of the target input text $\{t_1, t_2, t_3, t_4, \dots, t_n\}$.

Unit selection is carried out in two stages. The first stage deals with the target costs as discussed in section 2. The entire phoneme context tree is searched for units whose phonetic neighbouring context with respect to the central phone corresponds to the phonetic context of the phoneme in the target input text. Examining tree units at the quinphone level ensures that allophonic variations of the required central phone are taken into consideration during unit selection. The first stage of the unit selection process produces a pool of database candidate units, consisting of left diphones, right diphones, triphones and phones. The number of possible

database candidates returned from the context tree search depends on the database coverage. Phones are considered only if no left diphones, right diphones or triphones are present in the tree. The motivation behind this is that diphones and triphones are better at capturing the coarticulation effects of the preceding and following phones. Consequently, spectral discontinuities are less noticeable in the speech synthesis based on units longer than a phone, as can be seen in Figure 3.

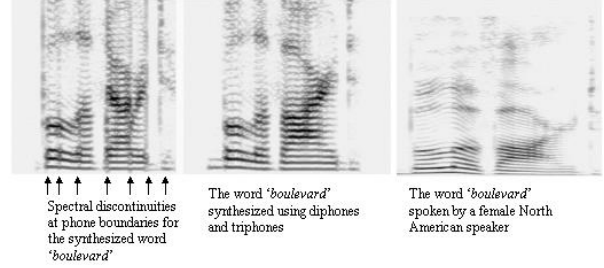


Figure 3: Narrowband spectrograms for synthesized speech; phone units (image on the left), non-uniform units (image in the middle) and normal speech (image on the right) for the word 'boulevard'

The second stage in the unit selection process selects from the pool of candidate units only those units that can be joined together at lowest concatenation costs. The distance metric used for finding the difference between pairs of candidate units and the Viterbi algorithm used for the selection of units are described in the next two sections.

4.1 Calculating Local Distances from Mel Filter Bank Amplitudes

To calculate local distance costs between the mel filter bank channels of two adjoining candidate units the Euclidean equation (2) is used,

$$D_{mfb} = \sum_{k=1}^{20} (mfbcontext_{k,l} - mfbcontext_{k,r})^2 \quad (2)$$

where k is the number of mel filter bank channels, l and r are left and right candidate units in a pair of units, $mfbcontext$ represents the context of the candidate unit (e.g. a triphone has left, central and right $mfbcontext$). The mel filter bank amplitude values for $mfbcontext$ depend on whether the candidate unit is a triphone, left diphone, right diphone or a phone. For example, to consider distance costs for concatenating a triphone and a phone, we find the Euclidean distance between the triphone's right mel filter banks and the phone's central mel filter banks. The right database candidate unit, r , corresponds to target t_i , but the left database candidate, l , can correspond to targets t_{i-1} or t_{i-2} from the target input text, depending on the available units in the pool of candidate units.

Local distance costs are accumulated along the path of successive units. The Viterbi-style algorithm, as described in the next section, searches through units with minimum local distance costs and selects the ones that can be concatenated at minimum global cost.

4.2 Viterbi-style Search

Two candidate units can be merged at low concatenation costs if the last phone of the first candidate unit is the same as the first phone of the second candidate unit. Synthesized speech

where speech segments were joined by overlapping the adjoining phones displays fewer spectral and audible discontinuities than the speech synthesized when abutting speech segments (in Figure 3, the left image shows abutted phones, whereas in the non-uniform synthesis (image in the middle) the adjoining units were overlapped). Table 1 shows a very small group of possible database candidates for target sequence /#lʊk/ where $t_1 = \#/\$, $t_2 = /l/\$, $t_3 = /ʊ/\$, $t_4 = /k/$ (N.B. /#/ represents silence). Different cell shadings indicate which database candidate units are expected to merge at low concatenation costs. For example, candidate units in diagonally shaded cells for t_3 are likely to incur low concatenation costs when merged with database candidates from the diagonally shaded cells for t_2 and t_1 . Local distance costs are calculated between database candidates that are likely to produce low concatenation costs based on their phonetic environment.

Target Phonemes	Possible Database Candidates			
	Phone	Left Diphone	Right Diphone	Triphone
t_1	#		#l	
t_2	l	#l	lʊ	#lʊ
t_3	ʊ	lʊ	ʊk	lʊk

Table 1: Database candidates that can be merged at low concatenation costs

All candidate units in the pool of database candidates are placed into a Viterbi matrix. Accumulated local distances are calculated between database units (u_i, u_{i-1}) and (u_i, u_{i-2}) depending on their phonetic context, and minimum accumulated local distances are recorded in the Viterbi matrix. The task of the Viterbi algorithm is to backtrack through the Viterbi matrix and return an optimal global path showing which units can be joined at the lowest global concatenation costs.

The steps in the Viterbi search implemented here are outlined below:

- For each target phoneme in the target input text the entire database is searched for possible candidate units;
- All candidate units are labeled with a left diphone, a right diphone, a triphone or, in absence of diphones and triphones, a phone label accordingly;
- For successive target phonemes and depending on the candidate unit's label the Euclidean distance is calculated between the adjoining phone segments for all available database candidates utilizing the amplitude values stored in twenty mel filter bank channels; minimum local Euclidean distances are stored in the Viterbi matrix.
- Local Euclidean distances are accumulated across the sequence of the target input text keeping record of units along the path.
- The optimal global path is generated by backtracking through the Viterbi matrix and selecting candidate units with minimum concatenation costs.

Figure 4 is an illustration of possible selection paths for the target input text 'look at' (/lʊkæt/) where $t_1 = \#/\$, $t_2 = /l/\$, $t_3 = /ʊ/\$, $t_4 = /k/$, $t_5 = /æ/$ and $t_6 = /t/$; row headings triphone, left diphone, right diphone and phone indicate possible database candidates. The optimal Viterbi path is shown as a solid line in Figure 4.

	t_1	t_2	t_3	t_4	t_5	t_6
Triphone		#lʊ	lʊk	ʊkæt	kæt	æt#
Left Diphone		#l	lʊ	ʊk	kæt	ætʊ
Right Diphone	#l	lʊ	ʊk	kæt	æt	t#
Phone			ʊ			

Figure 4: Possible selection paths through database candidates.

5. Evaluation

For the purpose of the experimental evaluation twenty sentences were synthesized using our own unit selection method (referred to hereafter as synthesizer A) and a reference synthesizer (referred to hereafter as synthesizer B). Thirteen listeners evaluated each sentence according to ease of listening [5], intelligibility and naturalness of the synthesized speech on the scale of 1 to 5, 1 being bad and 5 being excellent. All listeners were native English speakers with no known hearing problems. The listeners listened to pairs of sentences, which were randomized per synthesizer. Both synthesizers used the same underlining speech corpus and voice.

Frequency distributions of scores for synthesizers A and B are presented in Table 2. Figure 5 gives a comparison of mean scores for three evaluation test types per synthesizer.

Evaluation Scores	Synthesizer A		Synthesizer B	
	Freq. Distr.	% Distr.	Freq. Distr.	% Distr.
2	9	1.2	14	1.8
3	142	18.2	152	19.5
4	346	44.4	360	46.2
5	283	36.3	254	32.6
TOTAL	780	100	780	100

Table 2: Frequency distribution of scores per synthesizer

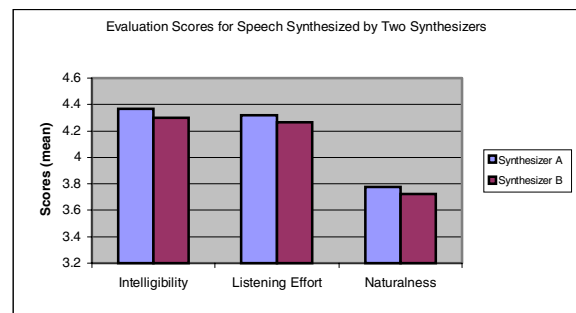


Figure 5: Comparison of Intelligibility, Listening Effort and Naturalness scores for two synthesizers.

A one-way within subjects ANOVA with three levels (intelligibility, listening effort and naturalness) was performed on evaluation scores for each synthesizer. It was found that differences between the means for the three test types in synthesizer A were significant at 95% and 99% confidence levels. Our value of $F(2,777) = 54.52$ exceeds the critical value of 3.01 alpha [.05] and 6.97 alpha [.01], $p < 0.0001$. The Tukey test [9] was used to make pairwise comparisons between the means for intelligibility, listening effort and naturalness. The mean differences for intelligibility v naturalness = 0.59, and listening effort v naturalness = 0.54 exceed the Tukey HSD critical values at 95% and 99%

confidence levels. The Tukey test revealed no significant difference between intelligibility and listening effort for the synthesizer A.

The differences between the mean scores for the three test types for synthesizer B were also found to be significant. $F(2,777)=54.04$, $p<0.0001$. The mean differences for intelligibility v naturalness = 0.58, for listening effort v naturalness = 0.55 exceed the Tukey HSD critical values at 95% and 99% confidence levels. The Tukey test revealed no significant difference between intelligibility and listening effort.

The collected data suggests that listeners are capable of differentiating between intelligibility and naturalness as well as listening effort and naturalness but that they cannot differentiate well between intelligibility and listening effort.

The between subjects ANOVA was performed on the scores collected to ascertain whether there is a significant difference between the two synthesizers. The calculated statistic $F(1,518) = 1.26$ and the estimated p-value < 0.26 at 95% confidence level suggests there is not a statistically significant difference between the intelligibility sample means for the two synthesizers. Based on a 95% confidence level, the estimated p-value < 0.03 for naturalness, $p<0.41$ for listening effort, and $p<0.26$ for intelligibility do not exceed the critical value $F(1,518)=3.86$ suggesting that there is no statistically significant difference between intelligibility, listening effort and naturalness of the two synthesizers.

Listeners' preference scores revealed that the synthesizer A had 136 votes (52%), synthesizer B 102 votes (39%) and on 22 counts (9%) both synthesizer were judged the same.

6. Conclusions

This paper presented a novel unit selection method based on a Viterbi-style algorithm using variable length units and acoustic and phonetic information stored in the large speech database. Phonetic and phonological environment of the target phonemes in the target input text were crucial to creating a pool of possible database candidates from the entire speech database. Smoother concatenations of successive speech units were achieved by calculating Euclidean distances between mel filter bank amplitude channels of the adjoining candidate segments. Considering database candidate units with a wider phonetic context ensured that allophonic variations of the central phone in the candidate unit were catered for.

The within and between subjects ANOVA revealed that the synthesis with our unit selection was not significantly different from the high quality reference system, but the listeners' preference scores showed significant preference for the speech synthesized by our unit selection method.

The length of candidate units selected from the speech database is heavily influenced by the speech database and its design. The final synthesized speech may contain sequences

of diphones and triphones that are a subset of a longer recorded speech sequence but the length of such subsets in the synthesized speech is determined by the recorded database coverage.

7. Further Work

Future work will be looking into database design concentrating on syllables and phonetic and stress environment within which they appear. The intention is to complement the current unit selection method with suprasegmental information contained in longer units, and make a selection of units from a much closer examination of acoustic and linguistic (particularly lexical and semantic) data.

8. Acknowledgements

The author wishes to thank Nuance Communications and EPSRC for sponsoring this research. The participation of the university's staff and students in the evaluation experiment is also thankfully acknowledged.

9. References

- [1] Hunt, A. J. and Black, A. W. "Unit Selection in a Concatenative Speech Synthesis System using a Large Speech Database." ICASSP, pp 373-376, 1996
- [2] Breen, A. P. and Jackson, P. "Non-Uniform Unit Selection and the Similarity Metric Within BT's Laureate TTS System." in Proc. of the Third ESCA/COCOSDA Workshop on Speech Synthesis, pp. 201-206, November 1998.
- [3] Beutnagel, M., Mohri, M., and Riley, M. "Rapid Unit Selection from a Large Speech Corpus for Concatenative Speech Synthesis." Proc. Eurospeech, 2:607-610, 1999
- [4] Black, A. W. and Campbell, N. "Optimising Selection of Units from Speech Databases for Concatenative Synthesis." Eurospeech '95, Madrid, Spain
- [5] Johnston, R. D. "Beyond Intelligibility - the Performance of Text-to-Speech Synthesizers." Speech Technology for Telecommunications, Chapman & Hall, 1998
- [6] Conkie, A., Beutnagel, M., Syrdal, A. K., and Brown, P. E. "Preselection of Candidate Units in a Unit Selection-Based Text-to-Speech Synthesis System." ICSLP 2000
- [7] Taylor, P. and Black, A. W. "Speech Synthesis by Phonological Structure Matching." Proceedings of the 6th European Conf. on Speech Communication and Technology, Budapest, Hungary, 1999, vol. II, p623-626.
- [8] Bulyko, I. and Ostendorf, M. "Joint Prosody Prediction and Unit Selection for Concatenative Speech Synthesis." In Proc. of ICASSP, 2001
- [9] Roberts, M. J. and Russo, R. "A Student's Guide to the Analysis of Variance", London, Routledge, 1999.
- [10] Rabiner L and Juang B-H. "Fundamentals of Speech Recognition." Prentice Hall, 1993;