Use of Clustering Information for Coarticulation Compensation in Speech Synthesis by Word Concatenation

Christos Vosnidis and Vassilis Digalakis

Department of Electronics and Computer Engineering
Technical University of Crete
GR-73100 Chania, Greece
vosnidis@telecom.tuc.gr vas@telecom.tuc.gr

Abstract
The Weather Report Synthesizer is a speech synthesis system for weather forecasts in Greek. Instead of trying to improve the synthesis quality of PSOLA based diphone concatenation speech synthesizers, we have chosen to use words as the synthesis units. This approach has the advantage of low complexity and quick implementation, while at the same time it achieves better speech quality due to the fact that the synthesis units inherently possess the necessary prosodic feature diversity. The selection of the optimal sequence of words that form the synthesized speech, however, presents the greatest challenge in the synthesis process. Several features are taken into consideration during the selection, but we have identified Coarticulation at the edges of consecutive words to have the greatest effect on the quality of the synthesized utterance. We present a novel method for evaluating a measure on coarticulation effects among pairs of words, based on feature clustering information obtained from a current Speech Recognition System.

1. Introduction
Concatenative speech synthesis systems have become the de facto standard for speech synthesis. Speech units, varying from diphones and syllables to whole words are concatenated in order to form the desired speech utterance.

The choice of the synthesis unit to be used greatly affects the quality of the synthesized speech. Large units such as whole words result in better speech quality. However, the greater the length of the speech unit, the larger the space requirements for such a synthesizer. We may choose to use words, but the space needed to store the synthesis units rapidly grows with the vocabulary size of the application’s domain. The alternative approach that uses diphones or syllables as the units to be concatenated allows for the coverage of any application domain with constant storage requirements, yet the quality deterioration is obvious.

In certain cases, speech synthesis applications have a restricted vocabulary. The implementation of a system that uses words as the synthesis units and that produces high-quality synthesized speech seems to be the best choice. However the implementation process exhibits certain challenges. We will discuss these challenges and present the course of action that we followed in dealing with them through the description of our Weather Report Synthesizer. Our system is used to synthesize weather forecasts. The domain that it covers is limited, as it features only 389 words. For greater prosodic diversity these words are recorded in multiple instances (98% of the words have at least 3 instances).

2. Corpus Construction
The selection and the recording of the corpus greatly affect the overall quality of the synthesized speech. Our approach to the synthesis problem (use whole words, no need for signal processing) relies heavily to the existence of each word in as many prosodic contexts as possible, thus allowing for the utilization of the instance that better matches the current context.

2.1. Corpus Selection
Having chosen to support the synthesis of weather reports, we obtained a large set of weather reports (1676 sentences - 19138 words), covering a period of a whole year, from the Greek National Meteorological Agency (EMY). These weather reports are multi-part documents, with each part covering a specific weather feature (temperature, wind intensity and direction, weather conditions, etc.) The set was segmented into the parts mentioned above, and each part was subjected to statistical analysis, out of which the sentences that covered each set’s vocabulary were extracted. Using a greedy algorithm that repeatedly extracted the sentence with the maximum number of words not yet included into the corpus, we selected 163 sentences (2494 words), covering a vocabulary of 389 words. Thus, only meaningful, application specific sentences were used for the construction of the Corpus.

2.2. Corpus Recording
A male speaker was asked to read the Corpus sentences in a well articulated and at a natural manner. Thus, we wanted to ensure that the several instances of the words would be available in their natural prosodic contexts.

The sentences were recorded in laboratory environment with a sampling frequency of 16kHz and were stored digitally using PCM encoding with 16bits/sample.
2.3. Corpus Processing and Annotation

After the recording session and before the segmentation of the corpus sentences into words, all the sentences were subjected to a simple energy averaging and DC offset removal operation. Thus, the average energy level of all sentences was set to -17.6dBm. This normalization procedure did not flatten the energy level within the sentences, preserving the prosodic variations of the words.

After the normalization, the sentences and their associated transcriptions were passed to the Nuance® v.7.0.2 Speech Recognition System that identified the word boundaries in the sentences and created their associated phonological transcriptions. The information extracted from this process was used for the automation of the corpus' segmentation and annotation. In addition to that, the corpus was also segmented and annotated manually. Later inspection and evaluation showed that the two Corpora produced synthesized speech of acceptable version if the word were spoken in isolation.

Inspection of all words using the following definition:

A phonological word is reduced if it deviates from a canonical form that a native speaker would judge as an acceptable version if the word were spoken in isolation [2].

The first two features are extracted automatically during the segmentation, using the metadata extracted in the previous stage. Reduction labeling was performed by acoustic inspection of all words using the following definition:

A phonological word is reduced if it deviates from a canonical form that a native speaker would judge as an acceptable version if the word were spoken in isolation [2].

3. Unit Selection

Unit selection consists of choosing the best sequence of unit instances to be used for the creation of the synthetic utterance. Following the suggestions of [2] and borrowing the concepts from object oriented programming, we will call the orthographic form of the word and its associated description a word class and a recorded word and its concrete description a word instance.

For each word class given by the utterance description there exist several word instances. All combinations of word instances that will form the correct synthetic utterance are potential solutions to the selection problem. We wish to use the best such combination for the creation of the utterance. By assigning a cost to every word combination, the search for the best word instance combination is reduced to identifying the combination with the minimum cost.

Minimum cost paths in graphs is a well-studied problem in computer science [3]. By representing the word instances as nodes in a directed acyclic graph, and by assigning costs to the edges connecting two successive word instances, a graph representing all possible word instance combinations is formed. Special nodes representing the start and end nodes of the graph are appended, eliminating the inconveniences of multiple start- and end-nodes. Thus, the problem of finding the optimum combination of word instances is reduced to that of identifying the single-source shortest path in the graph. We have chosen to use Dijkstra's Algorithm [4] for finding the solution to this problem.

3.1. Cost Assignment

The success of the selection method however is based on the cost assignment policy that we will utilize. Following, again the suggestions in [2], we used five cost functions to assign costs to the edges of the graph. Although we also distinguish between unit and transition costs, we did not find any use in keeping them stored in separate places. Therefore, for any given pair of successive nodes, the sum of the unit cost of the initiating node and the cost of the transition from this node to the next is stored in the edge connecting the two vertices. Table 1 contains these cost functions.

<table>
<thead>
<tr>
<th>Cost Function</th>
<th>Punitive Cost</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concatenation</td>
<td>1</td>
<td>Words not found consecutively in corpus</td>
</tr>
<tr>
<td>Coarticulation</td>
<td>$c \in [0, 1]$</td>
<td>Evaluated by expression $R_{\text{c}}(u_1, e, u_2, p) + R_{\text{c}}(u_1, n, u_2, s)$</td>
</tr>
<tr>
<td>Word Reduction</td>
<td>1.9</td>
<td>Word has the reduction property</td>
</tr>
<tr>
<td>Word Position</td>
<td>1 or 3</td>
<td>Applied to words found in different position.</td>
</tr>
<tr>
<td>Sentence Modality</td>
<td>1</td>
<td>Word modalities do not match.</td>
</tr>
</tbody>
</table>

All these functions assign no costs when the unit or the transition from the first unit to the next has the desired property. Otherwise, the costs shown in Table 1 are assigned. The first two are transition costs, while the latter three are unit costs. The cost functions are simple in their conception, yet their combination creates a very complex rule system.

3.1.1. Coarticulation Cost

The coarticulation cost for a given pair of word instances is assigned by comparing the last phoneme of the first instance with the first phoneme of the second one. In this comparison we also take into consideration the edge phonemes of the words preceding and following the two instances that form the pair.

![Figure 1](image)

The coarticulation cost for a given pair of word instances is assigned by comparing the last phoneme of the first instance with the first phoneme of the second one. In this comparison we also take into consideration the edge phonemes of the words preceding and following the two instances that form the pair.

For each word instance, four phonemes are also stored in the Word Segment Database, providing information on the phonetic context in which the instance was recorded. In particular, the last phoneme of the previous corpus word ('p'), the first ('s') and the last ('e') phonemes of the current word, and finally the first phoneme of the next corpus word ('n').

In such a context, the coarticulation cost for a combination of two word instances provides a measure on how smooth, from a phonetic point of view, the transition will be. We would have preferred that the two instances that are
used to form the pair were recorded in matching phonetic contexts. That is, for the first word, we had rather used an instance whose ‘n’ phoneme would be the same with the ‘s’ phoneme of the next word, while for the second one, we had rather used an instance whose ‘p’ phoneme is the same with the ‘e’ phoneme of the previous one.

However, hardly is such a case to be expected. Thus, a similarity function is used to evaluate the coarticulation cost:

\[
Cost = 0.5 R_{eq}(u_1, e, u_2, p) + 0.5 R_{eq}(u_1, n, u_2, s) \tag{1}
\]

where, \(u_1\) denotes the first and \(u_2\) the second word of the pair.

The first part refers to the phonetic similarity of the first word with the second one, and the second to the similarity of the second word with the first one. Each similarity function \(R_{eq}(\cdot)\) evaluates to \([0, 1]\), hence the 0.5 multiplier. We will focus on the implementation of the similarity function.

### 3.1.2. Similarity Function

In HMM-based speech recognition systems triphones are represented as three-state HMM processes. Transitions from a state to the next model the temporal change of the speech signal for that triphone. Each state has an output distribution that is associated with the acoustical vector of that particular segment of the triphone. Thus, each state represents part of the spectral features of the triphone.

![Figure 2 HMM representation for the triphone n[E]k](image)

Each HMM state produces as output a spectral feature vector. Due to the relatively large number of triphones for Greek (approx. 14,500), a clustering scheme is used to reduce the number of distinct triphone states.

![Figure 3 Clustering of Feature Vectors](image)

Triphone states are clustered together according to their similarity. Thus, a single vector, representing the whole collection, replaces each collection of similar states. Since the similarity of feature vectors is used as a criterion for the clustering procedure, clustering information may be utilized to obtain a notion on the phonetic similarity of any given set of triphones.

When we examine the phonetic similarity of the first word with the second one, we should determine the similarity of the classes of triphones that have \(u_1.e\) as their center phoneme, and \(u_1.n\) and \(u_2.s\) respectively, as their right phoneme. Thus, we should compare the feature vector classes \(*[u_1.e]u_1.n\) and \(*[u_1.e]u_2.s\), where * denotes all phonemes. In addition to that, the rightmost feature vectors, those that are produced by the last stage (denoted by -2) of the associated HMM should be used. The first half of the coarticulation cost will then be:

\[
R_{eq}(u_1, n, u_2, s) = \left\{ \begin{array}{ll}
1 & \text{if } C_{eq}(u_1, n) = 0 \\
\frac{c_{eq}(u_1, n, u_2, s)}{C_{eq}(u_1, n)} & \text{otherwise}
\end{array} \right. \tag{2}
\]

where

\[
c_{eq}(u_1, n, u_2, s) = \sum_{i \in \text{cluster}} \#([u_1.e]u_1.n-2), \#([u_1.e]u_2.s-2) \tag{3}
\]

\[
C_{eq}(u_1, n) = \sum_{i} c_{eq}(u_1, n, i) \tag{4}
\]

where \(\#([u_1.e]u_1.n-2)\) and \(\#([u_1.e]u_2.s-2)\) is the number of occurrences of each triphone state in the indicated cluster.

The reasoning behind these formulas is the following: We want to use a word instance that has been recorded in another phonetic context than the one defined by the next word instance in the pair, only if this context is similar to that of the next word. Equation 3 provides a measure of the co-occurrence of the two triphone classes in the same cluster. It calculates the sum of the products of the number of occurrences of triphones belonging to these two classes, in the same cluster. Equation 4 provides a measure of the co-occurrence of any triphone class with \(u_1.e\) as the central phoneme, along with triphones belonging in the \(*[u_1.e]u_1.n\) class. Dividing these two quantities, as in Equation 2, we get a measure on the phonetic similarity of the two triphone classes.

The equations used for the computation of the second part of Equation 1 follow. The reasoning is similar to that of Equations 2-4.

\[
R_{eq}(u_1.e, u_2.p) = \left\{ \begin{array}{ll}
1 & \text{if } C_{eq}(u_1.e) = 0 \\
\frac{c_{eq}(u_1.e, u_2.p)}{C_{eq}(u_1.e)} & \text{otherwise}
\end{array} \right. \tag{5}
\]

where

\[
c_{eq}(u_1.e, u_2.p) = \sum_{i \in \text{cluster}} \#([u_2.s]u_1.e-0), \#([u_2.s]u_2.p-0) \tag{6}
\]

\[
C_{eq}(u_1.e) = \sum_{i \in \text{cluster}} c_{eq}(u_1.e, i) \tag{7}
\]

### 4. Signal Manipulation

We have already stated that during the recording procedure, the average energy of all the Corpus sentences is normalized
to a common level. The energy normalization procedure is applied to the sentences rather than the words in order to preserve the intra-sentence energy fluctuations that are part of each word’s prosodic characteristics.

However, due to the automated segmentation process that we have incorporated in our system, it is most usual that the energy levels of any two non-consecutive word instances that are to be concatenated will be dissimilar, a mismatch that is perceived as plosive sounds between these words in the synthesized utterance. We are dealing with this problem by applying during the concatenation procedure a simple energy smoothing operation at the common edges of each word pair formed by word instances that were not consecutively spoken in the corpus. Thus, the left or the right half of a 640-point Hamming window, equal to 20msec of speech signal, is multiplied with the samples near the right or the left edge respectively of such a word instance, just prior to its concatenation.

5. Quality Evaluation

The acoustic aspects of our synthesizer were evaluated by a group of 18 subjects. Evaluation assessment was performed using an application specific test that has been suggested by ITU-T’s standardization sector for the quality evaluation of telephone speech, which has been appropriately modified to fit our needs.

It is a judgment test, meaning that listeners are asked to judge the performance of the system along a number of scales. It comprises rating on eight scales, namely one 2-point scale (0-1) acceptance, and seven 5-point scales (0-4) overall impression, listening effort, comprehension problems, articulation, pronunciation, speaking rate, and voice pleasantness.

The first four scales may be captured under the heading overall quality, while the latter four are directed at more specific aspects of the output and require analytic listening. The speech samples are excerpts from real weather forecasts.

During the first stage of the evaluation procedure, when the impact of the selection criteria on overall quality was to be assessed, the subjects were asked to listen to nine weather reports, each one synthesized with a combination of the selection criteria. Among them were reports that were synthesized by taking into consideration only the reduction property (None), all the selection criteria (All), combinations of them for instance a combination of concatenation and coarticulation costs (Concat+Cooc), and a prerecorded report (PreRec). No form of signal manipulation was applied to the word units used for the synthesis.

During the second stage of the procedure, we wanted to determine the affect of signal manipulation on the quality. This time the subjects were asked to listen to five reports, of which one made use of all the selection criteria (All+DSP), another used only the coarticulation cost (Cooc), and another only reduction property (None). All of these used the signal manipulation discussed previously. They were compared to the previous best output of the synthesizer (All) and a prerecorded weather report.

![Figure 4 Evaluation of Criteria](image)

The first four scales may be captured under the heading overall quality, while the latter four are directed at more specific aspects of the output and require analytic listening. The speech samples are excerpts from real weather forecasts.

During the first stage of the evaluation procedure, when the impact of the selection criteria on overall quality was to be assessed, the subjects were asked to listen to nine weather reports, each one synthesized with a combination of the selection criteria. Among them were reports that were synthesized by taking into consideration only the reduction property (None), all the selection criteria (All), combinations of them for instance a combination of concatenation and coarticulation costs (Concat+Cooc), and a prerecorded report (PreRec). No form of signal manipulation was applied to the word units used for the synthesis.

During the second stage of the procedure, we wanted to determine the affect of signal manipulation on the quality. This time the subjects were asked to listen to five reports, of which one made use of all the selection criteria (All+DSP), another used only the coarticulation cost (Cooc), and another only reduction property (None). All of these used the signal manipulation discussed previously. They were compared to the previous best output of the synthesizer (All) and a prerecorded weather report.

![Figure 5 Evaluation of Signal Manipulation](image)

We see that 1) the coarticulation cost is the single most significant feature in terms of quality; 2) the combination of all features produces better results than any combination of up to three features; and 3) with the signal processing enhancements, the average acceptance score is 0.7, which compares favorably to the average score of 0.8 for the prerecorded sentences.

6. References