A STATISTICAL METHOD FOR DATABASE REDUCTION FOR EMBEDDED UNIT SELECTION SPEECH SYNTHESIS

Pirros Tsiakoulis, Aimilios Chalamandaris, Sotiris Karabetsos, Spyros Raptis

Institute for Language and Speech Processing (ILSP)
Artemidos 6 & Epidavrou, Maroussi, GR 151 25, Athens, Greece
{ptsiak,achalam,sotoskar,spy}@ilsp.gr

ABSTRACT

This paper presents a new method for the reduction of an existing speech database in order to be used for domain independent embedded unit selection text-to-speech synthesis. The method relies on statistical data produced by the unit selection process on a large text corpus. It utilizes the selection frequency, as well as the actual score of each unit. Both objective and subjective evaluation of the method is performed in comparison with existing similar techniques.

Index Terms— speech database reduction, unit selection, embedded text-to-speech (TtS)

1. INTRODUCTION

The current state of the art unit selection synthesizers produce highly intelligible, near natural synthetic speech. However, this usually comes at the cost of large resource repositories and increased processing power. This is because unit selection speech synthesis relies on large speech databases. The larger the database is the more natural the synthetic speech is. The speech database usually consists of naturally spoken utterances, carefully annotated to the unit level. Each utterance comes from a text corpus designed to cover as many units as possible in different phonetic and prosodic contexts. The resulting repository of speech units may have little or great redundancy, on which speech variability and overall quality some times depend. However, there are cases where the available resources are limited and the computational power is low.

Various reduction techniques have been proposed with different goals. As outlined in [1] the strategies usually fall in two categories: the *bottom-up* and the *top-down* ones.

According to the top-down approach, the unit repository is investigated for the reduction process and a clustering process is performed, based on prosodic and phonetic properties [2, 3]. By doing so, the search space of the unit selection algorithm is reduced as each target unit is searched within the corresponding cluster. Other approaches, such as in [4], use heuristic phonetic or prosodic criteria to truncate the speech database. The disadvantage of this approach is that objective metrics of similarity may be difficult to identify, and the ones found may not be in accordance with the mechanisms of the unit selection module, as for example, in [5], where it is observed that a small database makes the unit selection algorithm to produce not always the best possible output.

On the other hand, the bottom-up approach is purely a data driven technique since it focuses on the statistical behavior of the unit selection algorithm. The output of the unit selection algorithm is statistically processed in order to reduce the unit repository. The statistical data is collected from the synthesis of a large text corpus, where the selection frequency of each unit is usually calculated and is used in the reduction process. For example, in [1] the removal of the less frequent units is proposed, while in [3] the selection frequency serves as a weight in a vector quantization clustering technique. A possible weakness of using only the selection frequency is that it does not help avoid redundancy. For example two very similar units, that are alternatively selected with high frequency by the algorithm, will both be selected to be included in the reduced database. This may be significant when the reduction rate is high.

Our method falls in the bottom-up category, and overcomes the aforementioned problem, by employing a technique motivated by the clustering idea of the top-down approach. The truncation process is based on the selection frequency as well as the actual score of each unit during the unit selection process. More specifically, the difference of the scores between two instance units of the same abstract unit (diphone) is used as a similarity metric between them.

2. SPEECH DATABASE REDUCTION

As mentioned earlier, there are cases where there is a need for text to speech synthesis in environments with limited resources and low computational power. A clear example of this is the environment of embedded devices such as mobile phones, PDAs etc. Our goal is to scale down an existing unit selection TtS system to fit in such restricted environ-

This work has been partially supported through Community and National funding under the "Information Society" programme of the Greek General Secretariat of Research and Technology.

ments while, at the same time, achieving minimal degradation in speech quality. Techniques that deal with domain specific databases (e.g. [6]) are not suitable to our case, since we are focusing on a general purpose embedded TtS system.

2.1. Overview of the unit selection algorithm

Our unit selection TtS system uses diphones as elementary units and its unit selection algorithm can be shortly described as follows. Let u denote a diphone and u_j be the j-th instance unit of u in the available speech database. Then given an utterance to be synthesized, e.g. a sequence of diphones $u^1, \ldots, u^N$ where N is the length of the utterance in diphones, the algorithms outputs the best path indexes $j_1, \ldots, j_N$ as:

$$\underset{best}{\text{path}} = \underset{j_1,\ldots,j_N}{\text{argmin}} \left(\sum_{i=1}^{N} C^T(u^i_{j_i}) + \sum_{i=2}^{N} C^J(u^i_{j_i}, u^{i-1}_{j_{i-1}}) \right) \quad (1)$$

where $C^T(u^i_j)$ is the weighted sum of N_T target costs for the given unit u^i_j, such as prosodic and phonetic context costs, and $C^J(u^i_j, u^{i-1}_k)$ is the weighted sum of N_J join costs (spectral, prosodic, etc) between the adjacent units $u^i_{j_i}$ and $u^{i-1}_{j_{i-1}}$:

$$C^T(u^i_j) = \sum_{c=1}^{N_T} w_c^{u^i} * C^T_c(u^i_j) \quad (2)$$

$$C^J(u^i_j, u^{i-1}_k) = \sum_{c=1}^{N_J} w_c^{u^i} * C^J_c(u^i_j, u^{i-1}_k) \quad (3)$$

The weights in the above sums are diphone dependent, as superscripts imply, in order to account for differences between diphones (e.g. no need for F0 cost in non-voiced joins). All C^T_c and C^J_c are configured to lie in the same range of values.

2.2. The proposed reduction method

The main idea of the proposed method is to keep the units that are most frequently used by the unit selection algorithm, and at the same time, avoid to keep similar units. The unit selection algorithm is ran upon a sufficiently large text corpus, and statistical data is collected for the truncation process.

With reference to the algorithm described in section 2.1 we define a score function for each unit in a synthesized utterance as the combined local target and join cost:

$$S(u^i_j) = C^T(u^i_j) + \min_k(C^J(u^i_j, u^{i-1}_k)) \quad (4)$$

The second term of the score function is a look-behind cost function and it expresses the best join cost of the unit u^i_j from all the unit instances of the previous diphone u^{i-1} in the utterance under consideration. This is used because a forward Viterbi search is used to find the best path. Alternatively a look-ahead cost function or both could be used, with the main principle of the method remaining the same.

We assume that if two units of the same diphone score the same (or similar) in a given utterance, they are seen as similar ones, as far as the algorithmic point of view is concerned, regardless of their objective similarity. This derives from (1) as C^T_c and C^J_c are summed to find the best path. Thus, we can use the difference of scores, averaged over the whole corpus, as a similarity metric between units of the same diphone.

For all the utterances processed by the algorithm, we basically collect the following statistical data:

- the selection frequency f^i_j for each u^i_j, namely the total number of times the unit was selected
- the mean score difference $D^i_{j,k} = \overline{|S(u^i_j) - S(u^i_k)|}$ for all pairs of units of the same diphone (with scores referring each time to same utterance)

The reduction method relies on both f^i_j and $D^i_{j,k}$ in order to select the appropriate units of a specific diphone u^i. Let K be the number of instances of u^i in the available database and $M < K$ the desired number of instances in the small database, then a greedy algorithm is used to select M units:

1. Initialize $F = [f^i_1, \ldots, f^i_K]$
2. Select $m = \text{argmax}_n\, F[n]$
3. Update $F[n] = F[n] * D^i_{n,m}$, for $n = 1, \ldots K$
4. If $\#selected < M$ goto to step 2

We define F as the fitness vector of the instance units, initialized with the selection frequencies. Next, we iteratively select the unit with the best fitness. The most important step of the algorithm is step 3, in which we update the fitness vector to avoid selecting similar units. This is not done explicitly, but motivated by the idea of *fitness sharing* used in genetic algorithms, the fitness vector is updated after each selection. The fitness of each unit is multiplied with its mean score difference with the last selected instance. By doing so, the fitness of the similar units to the selected ones deteriorates, while different become more fit. As a side-effect, $F[m]$ becomes zero, thus already selected units cannot be reselected.

The target value M for the number of units in the reduced database is determined by the desired reduction rate. The coverage meeting criterion (e.g. as proposed in [1]) cannot be used for two reasons. Firstly, because frequent units may be discarded by the method, and secondly, a specific coverage does not guarantee the reduction rate in all diphones. We propose an explicit function to determine the number of units: $M = \min(M_{max}, \max(M_{min}, \log_b(K)))$, where the parameters M_{max} and M_{min} ($M_{max} > M_{min}$) explicitly define the maximum and minimum number of instance units per diphone, while parameter b determines a logarithmic reduction rate distribution among diphones.

For comparison purposes we also experimented with other methods that rely on statistical behavior of the unit selection algorithm. We did not implement any top-down approach since, as mentioned earlier, it is difficult to find metrics of similarity between units, and the ones found in the literature are usually tailored to the specifics of the examined unit selection system. The most obvious method for comparison is the *"select most frequent units"* method [1]. In order to have meaningful results we use the same number of units per diphone M across methods. For a reference point we compare the methods with a *random selection method* and the complete database. Hereafter we refer to our method as P_F, to the most frequent selection as P_S and as P_R the random one.

3. EXPERIMENTAL SETUP

The experiments were carried out on a database of a Greek female speaker, which consists of a total of 1291 annotated utterances from a phonetically balanced corpus of modern Greek. The resulting complete database has 1098 diphones and contains about $115K$ instances. After benchmarking with various target embedded devices, we concluded that reasonably high reduction rates, up to 95%, are necessary. At such high reduction rates, a degradation of output speech quality is inevitable, especially as far as variability is concerned.

3.1. Data Collection

A large text corpus of no specific domain was collected for testing purposes. The total of about $12.5K$ sentences contain about $1.5M$ diphone instances. A 95% portion of the corpus was used to collect statistical data by the unit selection synthesis algorithm, and the rest was used for the objective evaluation process as such is described in section 3.2.

Various sets of small databases were built using the three methods described in section 2.2. Each set was built with the same reduction rate across the three methods. Figure 1 shows unit overlap ratios between database pairs of the same reduction rate. We observe that P_F and P_S have high overlap ratio when the reduction rate is low which stabilizes around 50% for very high reduction rates. The overlap of both P_F and P_S with P_R is lower as one would expect, and rapidly decreases with reduction rate. It must be noted that actual values in overlap ratios are rather high. This is because the reduction rate per diphone for the less frequent diphones is less than the total reduction rate. For example, in extreme cases where $K \leq M_{min}$ all reduction methods fully overlap.

3.2. Evaluation

For the evaluation of the proposed method and of the resulting databases, we used objective metrics derived from statistical parameters describing the behaviour of the unit selection algorithm. In other words, we decided to investigate the performance of the unit selection module for every database set.

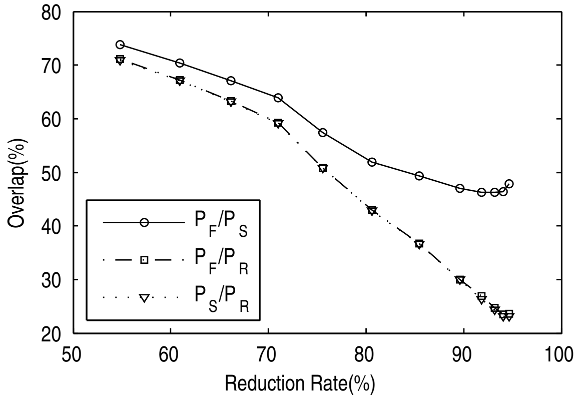

Fig. 1. Overlap ratios of databases built with different methods. P_F/P_R though not visible slightly differs from P_S/P_R.

For this purpose the commonly used statistics are; the mean values of target, join and total costs over the best path units.

In addition to this we introduce another set of objective metrics, also derived from the statistics of the unit selection algorithm. We consider the maximum target, join and total cost. By taking into account the maximum cost per utterance, we try to identify glitches in the synthetic speech, since places of high cost are potential prosodic, spectral or other types of discontinuities. Such cases, are usually avoided with the use of a large database, but this may be inevitable at high reduction rates. All the above statistical metrics are calculated per utterance and averaged over the whole test corpus.

3.3. Results and discussion

The results of the objective evaluation of the proposed method P_F versus P_S are illustrated in figure 2, using the statistical metrics described in section 3.2. It is noted that metrics for P_R are not shown since it performs worse than both P_F and P_S, and offers no other significant conclusion. As a reference point, the corresponding measures for the complete database system are $\{total, join, target\}_{mean} = \{0.15, 0.07, 0.07\}$ and $\{total, join, target\}_{max} = \{0.50, 0.27, 0.34\}$.

One can note in figure 2 that although P_S performs slightly better in terms of mean costs, P_F has a far lower average maximum cost per utterance, which becomes more pronounced as the reduction rate increases. This behavior indicates two main presumptions. The P_S method produces databases that result in synthetic utterances with good scores if averaged, but also having units with poor scores. On the other hand, P_F produced databases resulting in utterances with far better target cost at the cost of a slightly higher join cost.

In order to support our findings of the objective evaluation, we conducted some small scale listening tests. 35 short sentences (2 to 16 words long) were arbitrarily selected from the test corpus. The sentences were synthesized with databases produced by P_F and P_S with a reduction rate of $\sim 93\%$. A group of 15 listeners, speech experts and listeners with no ex-

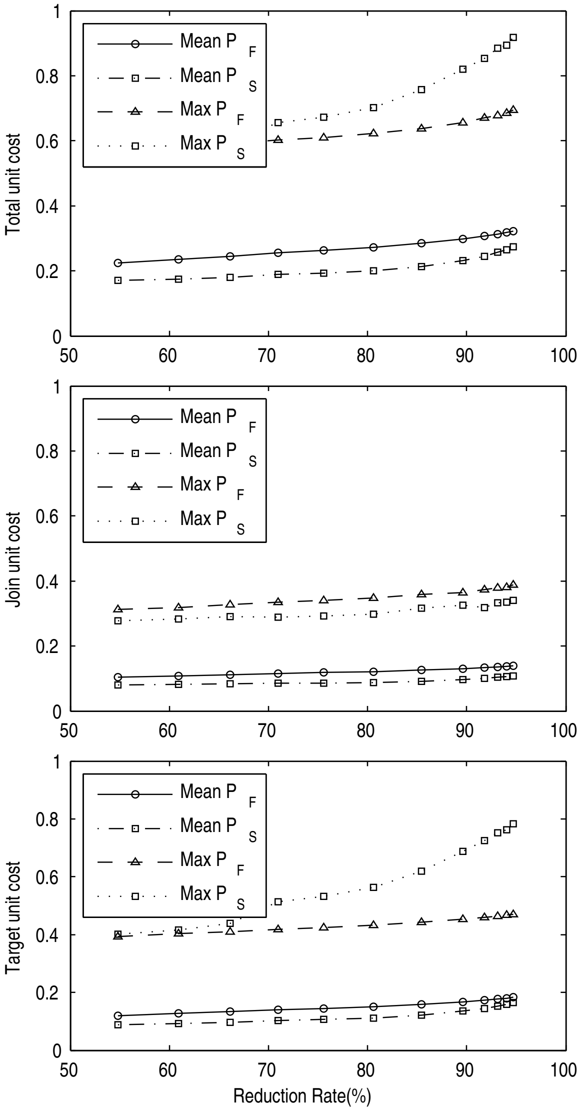

Fig. 2. Comparative objective evaluation between P_F and P_S. Top: the averaged mean (solid) and max (dashed) total cost per utterance for P_F and for P_S with dash-dotted and dotted lines respectively. Middle, bottom: the metrics for the join and target cost respectively are depicted with same notation.

perience in synthetic speech, were asked to evaluate each pair of sentences, presented in a shuffled order each time. The results are summarized in table 1 where the mean opinion score (MOS) is shown together with the objective metrics (total, join, target costs).

The results show that P_F produces better synthetic speech than P_S. Also, there is an agreement of the MOS values and the averaged maximum total cost per utterance. This seems to verify our initial hypothesis that P_S could result in redundant units in terms of target features, by selecting more similar units and leaving at the same time no room for other units to cover other less frequent but equally important cases met in a general purpose TtS. The better correlation of the MOS results with the Max Costs than the Mean Costs also indicates a possible room for improvement on the cost function [7], but we rather focus on the database reduction keeping the same tuned selection algorithm.

Table 1. Subjective vs Objective Evaluation

	MOS	Mean Costs	Max Costs
P_F	4.01	$0.32, 0.14, 0.18$	$0.61, 0.34, 0.40$
P_S	3.92	$0.27, 0.11, 0.16$	$0.84, 0.30, 0.71$

4. CONCLUSIONS

In this paper we presented a new method for the reduction of the speech database used by unit selection TtS system. The method relies on statistical data derived from the unit selection process in order to select the units that are frequently used by the system and to avoid at the same time similar units that would produce similar results, as far as the output speech quality is concerned. By doing so we achieve small-sized speech databases, which ensure though high diversity and little redundancy, making them suitable for a general purpose embedded unit selection TtS system. The method was evaluated on an objective and on a subjective basis, showing that it performs better than existing similar techniques and confirming our hypotheses regarding appropriateness of the method for producing general purpose embedded TtS systems.

5. REFERENCES

[1] Peter Rutten, et al, “A statistically motivated database pruning technique for unit selection synthesis,” in *Proc. ICSLP*, Denver, Colorado, USA, 2002, pp. 125–128.

[2] Alan W. Black and Paul A. Taylor, “Automatically clustering similar units for unit selection in speech synthesis,” in *Proc. Eurospeech*, Rhodes, Greece, 1997, pp. 601–604.

[3] Sanghun Kim, et al, “Pruning of redundant synthesis instances based on weighted vector quantization,” in *Proc. EUROSPEECH*, Aalborg, Denmark, 2001, pp. 2231–2234.

[4] Rohit Kumar and S. Prahallad Kishore, “Automatic pruning of unit selection speech databases for synthesis without loss of naturalness,” in *Proc. INTERSPEECH*, Jeju Island, Korea, 2004, pp. 1377–1380.

[5] Matthew P. Aylett, et al, “The cerevoice blizzard entry 2006: A prototype small database unit selection engine,” in *Proc. Blizzard Challenge Workshop*, 2006.

[6] Aleksandra Krul, et al, “Approaches for adaptive database reduction for text-to-speech synthesis,” in *Proc. INTERSPEECH*, Antwerp, Belgium, 2007, pp. 2881–2884.

[7] Tomoki Toda, et al, “An evaluation of cost functions sensitively capturing local degradation of naturalness for segment selection in concatenative speech synthesis,” *Speech Communication*, vol. 48, no. 1, pp. 45–56, January 2006.