LSM–BASED UNIT PRUNING FOR CONCATENATIVE SPEECH SYNTHESIS

Jerome R. Bellegarda

Speech & Language Technologies
Apple Computer, Inc.
Cupertino, California 95014

ABSTRACT

The level of quality that can be achieved in concatenative text-to-speech synthesis is primarily governed by the inventory of units used in unit selection. This has led to the collection of ever larger corpora in the quest for ever more natural synthetic speech. As operational considerations limit the size of the unit inventory, however, *pruning* is critical to removing any instances that prove either spurious or superfluous. This paper proposes a novel pruning strategy based on a data-driven feature extraction framework separately optimized for each unit type in the inventory. A single distinctiveness/redundancy measure can then address, in a consistent manner, the (traditionally separate) problems of outliers and redundant units. Experimental results underscore the viability of this approach for both moderate and aggressive inventory pruning.

Index Terms— Text-to-speech synthesis, unit selection, inventory pruning, outlier removal, unit redundancy management

1. INTRODUCTION

In concatenative text-to-speech (TTS) synthesis, the acoustic signal is generated from pre-recorded speech segments, normally extracted from a large database with varied phonetic and prosodic characteristics. The selection of the best unit sequence is cast as a multivariate optimization task, where the unit inventory is searched to minimize suitable cost criteria across the whole target utterance [1]. In practice, it is often necessary to modify the chosen instances in order to reduce audible discontinuities, and/or more precisely match the target prosody [2]. But because any such manipulation is liable to degrade signal quality, it is highly desirable to select units for which the minimum amount of post-processing is required [3]. Obviously, this is only feasible if the unit inventory comprises enough distinct units to cover all possible acoustico-linguistic events, leading to an exponential growth in the size of the database.

Of course, operational considerations limit this size to some appropriate practical value, and in many situations the optimal unit is simply not available. To mitigate any ensuing degradation in quality, a great deal of attention is typically paid to the level of coverage associated with a given inventory. Still, achieving higher coverage usually means recording a larger corpus, especially when the basic unit type is polyphonic, as in the case of syllables or words. Unit inventories with a footprint close to 1 GB are now routine in server-based applications (cf. [4]). The next generation of unit selection systems could easily bring forth an order of magnitude increase in this footprint, as ever more acoustico-linguistic events are included in the corpus to be recorded.

Unfortunately, such sizes are too large for deployment outside of a server environment. Even after applying standard compression techniques, the resulting TTS sytem could not ship as part of, say, an OS distribution. This has sparked interest in various ways to *prune* a unit inventory, i.e., to decide which units are best kept and which are best discarded, so as to achieve a given overall target size.[1] As in other areas of speech synthesis, there is typically a trade-off between quality and amount of resources expanded (here in terms of inventory size and selection time). As a rule of thumb, pruning 20% of units usually makes no significant difference (and may even improve perception), while up to 50% may be removed without seriously degrading quality [6].

The difficulty is to come up with a consistent, scalable framework for pruning. There are two considerations to take into account. First, no instance of a given unit type should be removed as long as it corresponds to a non-pathological realization, however atypical it may appear at first glance. Second, no unit should be kept when a similar rendition, likely to be perceived as interchangeable, is already present in the corpus. Thus, what is needed is an automatic method to maximize coverage while minimizing redundancy, i.e., to determine *a posteriori* (i) which units are suitably distinctive yet mainstream enough to keep, and (ii) which units are sufficiently spurious or redundant to discard. This paper proposes to treat these two aspects at the same time in a holistic way.

The underlying framework relies on the alternative TTS feature extraction we recently introduced [7] based on the *latent semantic mapping* (LSM) paradigm [8]. With this approach, it is possible to define a global discontinuity metric for characterizing the acoustic (dis-)similarity between two candidate segments [9]. The ensuing global outlook has also been successfully leveraged for the iterative refinement of unit boundaries, thus enabling unsupervised unit boundary optimization [10]. In order to be applied in the context of pruning, the feature extraction of [7] – [10] must be modified so that not just a boundary region, but an entire unit from the inventory, can be mapped into a convenient vector space. The resulting representation can then be leveraged to evaluate distinctiveness and redundancy across units.

The paper is organized as follows. The next section goes over the different aspects of pruning, and motivates the use of the LSM feature extraction framework. In Section 3, we describe in greater detail the resulting LSM space, along with the criterion used to assess distinctiveness and redundancy in that space. Section 4 presents the procedure followed for inventory pruning. Finally, in Sections 5 and 6 we report on experimental evaluations which illustrate some of the benefits of this pruning strategy.

2. PRUNING OVERVIEW

For the sake of clarity, let us define a *unit type* as an acoustico-linguistic event of interest (be it an individual demi-phone, phoneme, diphone, syllable, word, or sequence thereof, possibly in a specific acoustic and/or prosodic context), and an *unit* as an individual observation, or instance, of that unit type in the training database. In

[1] The term "pruning" can be traced to unit selection systems based on decision trees [5], but its usage is now more generic.

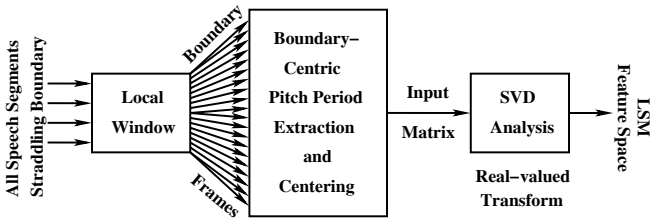

Fig. 1. *Original LSM Feature Extraction Framework.*

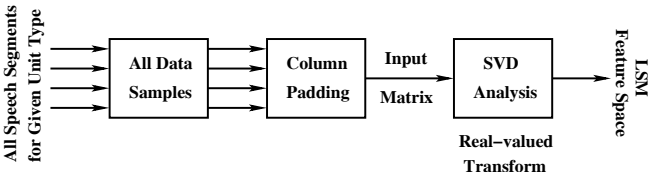

Fig. 2. *Pruning-Specific LSM Feature Extraction Framework.*

general, multiple units are available for each unit type considered. The collection of all units for all unit types forms the underlying unit selection inventory. As unsupervised pruning is normally done at the level of units rather than unit types, we will adopt this outlook throughout.

There are two distinct aspects to inventory pruning. The first seeks to remove spurious units, known as "outliers," which may have been caused by mislabeling, poor articulation, or other artifacts in the original recording. The second seeks to remove those units which are so common that there is no significant distinction between candidates for a given unit type. Both rely on some abstract representation of the units which lends itself well to clustering. The general idea is to cluster together units that are "similar," and compare units from each cluster to the relevant cluster center. Pruning is then achieved by removing those instances that are "furthest away" from the cluster center.

For example, in [5] each unit is represented as a sequence of frames, or vectors of MFCC coefficients, and decision tree clustering proceeds based on questions concerning prosodic and phonetic context; units are then assessed based on their frame-based distance to each cluster center. In [11], this evaluation relies on the underlying HMM framework: only instances with the highest HMM scores are kept to represent a cluster of similar ones. In [12], the parameterization involves LPC-based cepstral coefficients, and the distance measure is suitably weighted so as to prune away the least frequently used instances, as determined by synthesizing a large corpus of about 20,000 sentences. Alternatively, in [13], each unit is characterized by a small number of prosodic features, while the evaluation criterion is carefully tuned to favor both prosodically neutral and frequently used instances.

Common to the above systems is a marked sensitivity of the pruning outcome to the particular distance measure adopted for calculating the impurity of a cluster (as well as, if applicable, to the particular corpus chosen for establishing the relative importance of units). The selected metrics tend to be local in nature, which typically results in suboptimal (greedy) clustering. Also, in some cases, looking at the distribution of the distances within clusters to quantify what is meant by "close enough" can be a fairly opaque process. In the approach of [5], for example, this is best done before the final splits in the tree, and only for the most common unit types [6]. This illustrates a certain lack of scalability, and the need for at least some human supervision. It is therefore legitimate to ask whether an alternative unit representation might not allow for a more robust pruning solution.

Ideally, this representation should encapsulate all acoustic and prosodic aspects of the units, while still providing a low dimensionality description so as to facilitate clustering. Also, any distance measure defined between units should reflect a consistent and scalable outlook, preferrably connected to perceived quality differences. As it turns out, such properties are exhibited by the LSM-based TTS feature extraction recently introduced in [7], [9], [10]. This leads us to consider this approach for pruning as well.

3. LSM FRAMEWORK

In its original incarnation, the LSM-based framework for TTS feature extraction followed the approach of Fig. 1, in which a modal analysis of the signal is carried out through a pitch synchronous real-valued transform for a given segment of speech. In that application, each speech segment was limited to a fairly narrow region, centered around the boundary between two units from the unit inventory. And since the focus was on representing possible concatenations between units, it made sense to map individual pitch periods, rather than the whole segment, in order to obtain a fine enough resolution.

In the present work, we no longer need to resort to pitch period extraction, since we are now seeking a representation for entire units, i.e., generally multi-phonemic entities. This requires the notion of segment to be extended so it can cover a whole unit. The resulting framework is illustrated in Fig. 2.

Assume that for the unit type of interest, M instances are present in the unit inventory. The first step is to gather these M instances, retaining all time-domain samples associated with each unit. If N denotes the maximum number of samples observed over this collection, we then zero-pad all units to N, as necessary. The outcome is a $(M \times N)$ matrix W with elements w_{ij}, where each row w_i corresponds to a particular unit, and each column t_j corresponds to a slice of time samples. This matrix W, illustrated in the left-hand side of Fig. 3, encapsulates all the evidence regarding the unit type that can be observed from the training data. Typically, M and N are on the order of a few thousands to a few tens of thousands.

At this point we perform the eigenanalysis of W via singular value decomposition (SVD) as [9]:

$$W = U\,S\,V^{\,T}\,, \tag{1}$$

where U is the $(M \times R)$ left singular matrix with row vectors u_i $(1 \leq i \leq M)$, S is the $(R \times R)$ diagonal matrix of singular values $s_1 \geq s_2 \geq \ldots \geq s_R > 0$, V is the $(N \times R)$ right singular matrix with row vectors v_j $(1 \leq j \leq N)$, $R < \min(M, N)$ is the order of the decomposition, and ^{T} denotes matrix transposition. As is well known, both left and right singular matrices U and V are column-orthonormal, i.e., $U^{\,T}U = V^{\,T}V = I_R$ (the identity matrix of order R). Thus, the column vectors of U and V each define an orthornormal basis for the space of dimension R spanned by the (R-dimensional) u_i's and v_j's. By analogy with the latent semantic analysis framework,[2] the resulting feature space is called the *LSM space* $\mathcal{L}$ [8].

The interpretation of (1) in Fig. 3 focuses on the orthornormal basis obtained from V. Projecting the row vectors of W onto that basis defines a representation for the units in terms of their coordinates

[2]This is where the expression "semantic" in LSM comes from, although in the present context "global" would be a more accurate terminology.

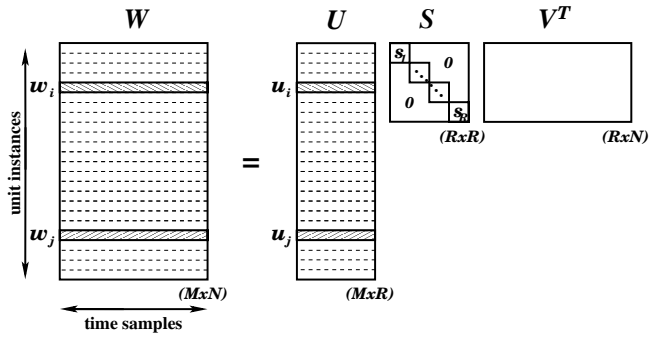

Fig. 3. *Decomposition of the Input Matrix.*

in this projection, namely the rows of US. Thus, (1) defines a mapping between the set of units and (after appropriate scaling by the singular values) the set of R-dimensional feature vectors $\bar{u}_i = u_i S$. These vectors can then be viewed as feature vectors analogous to, e.g., the usual cepstral vectors.

In contrast to such traditional Fourier-derived features, however, the relative positions of the LSM vectors in the space $\mathcal{L}$ are determined by the overall characteristics observed in the relevant units, as opposed to an analysis restricted to a particular unit (be it frequency domain processing or otherwise). Hence, two vectors $\bar{u}_i$ and $\bar{u}_j$ "close" (in some suitable metric) to one another in $\mathcal{L}$ can be expected to reflect a high degree of similarity in the relevant units, and thus potentially a small amount of perceived difference between them. This forms the basis for eliminating redundancy across perceptually interchangeable units.

4. PRUNING

To meaningfully compare $\bar{u}_i$ and $\bar{u}_j$ in the LSM space $\mathcal{L}$, we follow the standard reasoning underlying latent semantic mapping. Recall from [7]–[10] that the expression for the closeness between two (row) vectors is given by:

$$c(\bar{u}_i, \bar{u}_j) = \cos(u_i S, u_j S) = \frac{u_i \, S^2 \, u_j^T}{\|u_i S\| \, \|u_j S\|} \,, \qquad (2)$$

for any $1 \leq i, j \leq M$. In other words, two vectors $\bar{u}_i$ and $\bar{u}_j$ with a high value of the measure (2) are considered closely related, and thus potentially interchangeable. Conversely, two vectors with a low value of (2) are considered perceptually distinct. We refer to (2) as the *distinctiveness/redundancy measure (DRM)* induced over the LSM space $\mathcal{L}$.

The measure (2) allows us to proceed with the clustering of the vectors in the LSM space, using any of a variety of standard algorithms. Since for some unit w the number of such vectors may be large, it may be advisable to perform this clustering in stages, using, for example, K-means and bottom-up clustering sequentially. In that case, K-means clustering is used to obtain a coarse partition of the units into a small set of superclusters. Each supercluster is then itself partitioned using bottom-up clustering. The outcome is a final set of clusters C_k, $1 \leq k \leq K$, where the ratio M/K defines the reduction factor achieved.

The overall pruning procedure follows the flowchart of Fig. 4. The basic idea is to focus on each unit type in turn, and proceed as detailed earlier to gather the relevant M units and derive the resulting LSM space $\mathcal{L}$ associated with this unit type. This results

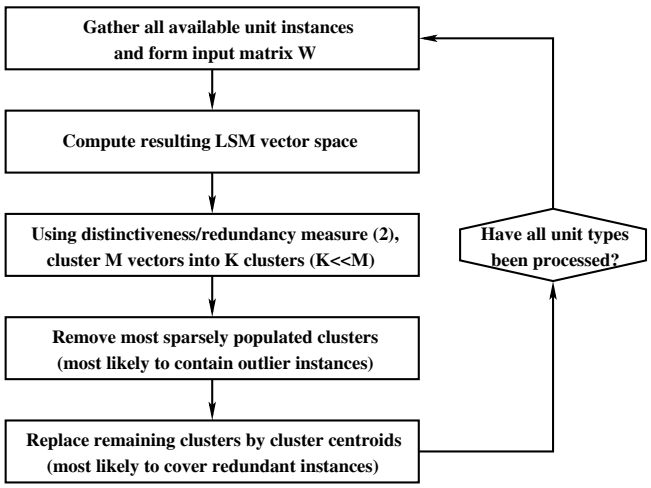

Fig. 4. *Data–Driven Pruning Procedure.*

in M feature vectors in the LSM space. We then use the distinctiveness/redundancy measure (2) to cluster these vectors into K clusters, where $K \ll M$. Once these K clusters have been obtained in $\mathcal{L}$, we proceed to eliminate all clusters with less than n vectors, which are most likely to be associated with outlier units. The remaining clusters, by definition, comprise vectors which are very close to one another in the LSM space, and which are therefore good candidates for interchangeability. It is thus safe to replace them by their centroid, or, in practice, the actual unit which maps closest to that centroid in the space $\mathcal{L}$. All other instances of that unit type in the same cluster can therefore be pruned away. The procedure iterates on the set of unit types until all of them have been processed. The collection of retained vectors then forms the basis for the pruned inventory of units.

5. PRELIMINARY RESULTS

To validate the basic approach, we first conducted a preliminary experiment focused on the word $w =$ see. Specifically, we randomly extracted from a unit inventory $M = 8$ units[3] associated with the unit type "see". For each of these units, we gathered all associated time-domain samples, and observed a maximum number of samples across all units of $N = 10721$. This led to a (8×10721) input matrix. We then computed the SVD of this matrix and obtained the associated feature vectors as described in Section 3. Note that, because of the low value of M, we used $R = 8$ for the dimension of the LSM space.

We then clustered these feature vectors using bottom-up clustering. In this simple case, the most natural outcome was 3 distinct clusters, for a reduction factor of 2.67. Each cluster was analyzed in detail for acoustico-linguistic similarities and differences. We found that the first cluster predominantly contained instances of "see" spoken with an accented vowel and a flat or falling pitch. The second cluster predominantly contained instances of "see" spoken with an unaccented vowel and a rising pitch. Finally, the third cluster predominantly contained instances of "see" spoken with a distinctly tense version of the vowel and a falling pitch.

[3]The reason we purposely limited M to this unusually low value was to keep the later analysis of every individual unit tractable. Also note that for this unit type no obvious outlier was present in the database.

Table I: Mean Opinion Scores. Maximum Score Achievable is 5. $RF =$ Reduction Factor.

Utterance Number	Baseline $RF = 1$	Moderate Pruning $RF = 1.25$	Aggress. Pruning $RF = 2$
1	3.3	3.0	3.0
2	2.4	2.9	2.9
3	4.1	4.0	4.0
4	3.0	2.6	2.5
5	2.9	2.8	2.6
Average MOS	3.1	3.1	3.0
95% Confid.	± 1.1	± 1.0	± 1.0

In all cases, it anecdotally felt that replacing one unit by another from the same cluster would largely maintain the "sound and feel" of the utterance, while replacing it by another from a different cluster would be seriously disruptive to the listener. Thus, the LSM-based pruning strategy seemed to be working, and motivated the larger-scale experiments reported below.

6. FORMAL EVALUATION

In this section we briefly summarize the results of an investigation conducted using a phonetically and prosodically varied voice database currently deployed in MacinTalk, Apple's TTS offering on MacOS X. This database is fairly similar to the Victoria corpus described in detail in [14]. In particular, recording conditions closely follow those mentioned in [14], though individual utterances generally differ. The underlying corpus, called Alex, is about an order of magnitude larger than the Victoria corpus.

As stimuli, we generated a set of 5 sentences synthesized from each of 3 different unit inventories: (i) the baseline inventory, where no pruning was performed, which corresponds to a reduction factor of $RF = 1$, (ii) the inventory obtained by setting the target reduction factor to $RF = 1.25$, which corresponds to a moderate pruning of 20% of all units, and (iii) the inventory obtained by setting the target reduction factor to $RF = 2$, which corresponds to a more aggressive pruning of 50% of all units. Note that the former type of pruning is more closely aligned with outlier removal, while the latter is largely dominated by redundancy pruning. Thus, this test was thought to properly exercise the two distinct aspects of pruning discussed in Section 2.

Eight participants were asked to score each of the 5 utterances from the 3 different databases on the typical MOS scale, where 5 is the best. Tabulating the results yields the score distributions presented in Table I. This table shows that, on the average, the sentences synthesized from the pruned inventories were not rated noticeably worse than those synthesized from the baseline inventory. This bodes well for the viability of the proposed approach when it comes to reducing the size of the unit inventory in concatenative text-to-speech synthesis.

7. CONCLUSION

We have proposed a consistent solution to the data-driven pruning of outliers and redundant units in unit selection TTS. This approach leverages the LSM decomposition of information gathered across a given speech segment, which was previously exploited in [7], [9], [10] in the case of unit boundaries. Here we extend this approach to include entire units. Compared to standard Fourier analysis, the LSM framework allows all relevant units to be mapped onto the same, separately optimized feature space of relatively low dimension. It then becomes possible to define a single measure on this space to assess the degree of distinctiveness and/or redundancy between individual units.

This measure in turn allows for standard clustering in the LSM space, and thus for the specification of empirical thresholds to identify pathological outliers from otherwise distinctive units, as well as redundant instances from otherwise representative units. The outcome is a flexible pruning framework with clear trade-offs between the inventory size desired and the amount of pruning necessary to achieve it. Empirical evaluations indicate that this framework allows, in a fully unsupervised manner, the size of the unit inventory to be reduced by up to a factor of 2 without noticeable degradation in perceived TTS quality.

8. REFERENCES

[1] A. Hunt and A. Black, "Unit Selection in a Concatenative Speech Synthesis System Using Large Speech Database," in *Proc. ICASSP*, Atlanta, GA, pp. 373–376, 1996.

[2] W.N. Campbell and A. Black, "Prosody and the Selection of Source Units for Concatenative Synthesis," in *Progress Speech Synth.*, J. van Santen, R. Sproat, J. Hirschberg, and J. Olive, Eds., New York: Springer, pp. 279–292, 1997.

[3] M. Balestri, A. Pachiotti, S. Quazza, P.L. Salza, and S. Sandri, "Choose the Best to Modify the Least: A New Generation Concatenative Synthesis System," in *Proc. 6th Eurospeech*, Budapest, Hungary, pp. 2291–2294, September 1999.

[4] M. Beutnagel, A. Conkie, J. Schroeter, Y. Stylianou, and A. Syrdal, "The AT&T Next–Gen TTS System," in *Proc. 137th Meeting Acoust. Soc. Am.*, pp. 18–24, 1999.

[5] A.W. Black and P. Taylor, "Automatically Clustering Similar Units for Unit Selection in Speech Synthesis," in *Proc. 5th Eurospeech*, Rhodes, Greece, Vol. 2, pp. 601–604, September 1997.

[6] A.W. Black and K. Lenzo, "Optimal Data Selection for Unit Selection Synthesis," in *Proc. 4th ISCA Speech Synth. Workshop*, Perthshire, Scotland, paper 129, August 2001.

[7] J.R. Bellegarda, "A Novel Discontinuity Metric for Unit Selection Text–to–Speech Synthesis," in *Proc. 5th ISCA Speech Synth. Workshop*, Pittsburgh, PA, pp. 133–138, June 2004.

[8] J.R. Bellegarda, "Latent Semantic Mapping," *Signal Proc. Magazine, Special Issue Speech Technol. Syst. Human–Machine Communication*, L. Deng, K. Wang, and W. Chou, Eds., Vol. 22, No. 5, pp. 70–80, September 2005.

[9] J.R. Bellegarda, "A Global, Boundary–Centric Framework for Unit Selection Text–to–Speech Synthesis," *IEEE Trans. Audio Speech Language Proc.*, Vol. ASL–14, No. 3, pp. 990–997, May 2006.

[10] J.R. Bellegarda, "Globally Optimal Training of Unit Boundaries in Unit Selection Text–to–Speech Synthesis," *IEEE Trans. Audio Speech Language Proc.*, to appear, Vol. ASL–15, No. 2, February 2007.

[11] H. Hon, A. Acero, X. Huang, J. Liu, and M. Plumpe, "Automatic Generation of Synthesis Units for Trainable Text-to-Speech Systems," in *Proc. ICASSP*, Seattle, WA, Vol. 1, pp. 293-296, 1998.

[12] S. Kim, Y. Lee, and K. Hirose, "Pruning of Redundant Synthesis Instances Based on Weighted Vector Quantization," in *Proc. 7th Eurospeech*, Aalborg, Denmark, Vol. 3, pp. 2231-2234, September 2001.

[13] Y. Zhao, M. Chu, H. Peng, and E. Chang, "Custom–Tailoring TTS Voice Font—Keeping Naturalness when Reducing Database Size," in *Proc. 8th Eurospeech*, Geneva, Switzerland, pp. 2957-2960, September 2003.

[14] J.R. Bellegarda, K.E.A. Silverman, K.A. Lenzo, and V. Anderson, "Statistical Prosodic Modeling: From Corpus Design to Parameter Estimation," *IEEE Trans. Speech Audio Proc., Special Issue Speech Synthesis*, N. Campbell, M. Macon, and J. Schroeter, Eds., Vol. SAP–9, No. 1, pp. 52–66, January 2001.