LETTER

Outlier Detection and Removal for HMM-Based Speech Synthesis with an Insufficient Speech Database

Doo Hwa HONG[†a)], **June Sig SUNG**[†b)], **Kyung Hwan OH**[†*c)], *Nonmembers*, *and* **Nam Soo KIM**[†d)], *Member*

SUMMARY Decision tree-based clustering and parameter estimation are essential steps in the training part of an HMM-based speech synthesis system. These two steps are usually performed based on the maximum likelihood (ML) criterion. However, one of the drawbacks of the ML criterion is that it is sensitive to outliers which usually result in quality degradation of the synthesized speech. In this letter, we propose an approach to detect and remove outliers for HMM-based speech synthesis. Experimental results show that the proposed approach can improve the synthetic speech, particularly when the available training speech database is insufficient.

key words: *HMM-based speech synthesis, decision tree-based clustering, outlier detection, insufficient speech database*

1. Introduction

Hidden Markov model (HMM)-based parametric speech synthesis techniques have been developed over the past two decades. With these techniques, the synthetic speech of acceptable quality can be generated with flexible variation of speech characteristics [1], [2]. In HMM-based speech synthesis, spectrum, excitation, and state duration are modeled simultaneously in a unified framework [3]. In order to account for variability of the extracted features, context-dependent models are usually employed to consider prosodic and linguistic contexts. However, it is practically impossible to prepare a speech database which covers all the possible contextual varieties. To alleviate this problem, decision tree-based HMM state clustering techniques are usually adopted [4]. In general, the maximum likelihood (ML) criterion is used to select the best split candidate and a stopping criterion is given by the minimum description length (MDL) principle for the construction of a decision tree [5].

The two essential parts of HMM training, decision tree-based clustering and parameter estimation, are established based on the ML criterion. The ML criterion is, however, sensitive to outliers, which mean the data quite different from the rest. Outliers in a speech database are indicative of incorrect pronunciation or articulation difficult to classify by given labels. The outlying data may increase the complexity of the decision tree inefficiently, hamper the estimation of accurate model parameters, and result in degraded quality of the synthetic speech. Though the speech synthesis system should have a well-balanced, consistent, and clean speech database to generate a high-quality speech, it may practically have inherent deficiencies which cause outliers in acoustic modeling due to lack of the database size, inadequacy of contextual factors through a weak text analysis, inconsistency or varied exuberance of speaking style, segmentation error, and so on.

In this letter, we propose a technique for outlier detection and removal during the training procedure of HMM-based speech synthesis when given an insufficient speech database. In our previous study, we employed decision tree-based clustering incorporating an outlier detection technique [6]. In this work, we add an approach to remove outliers after the clustering step. For decision tree-based clustering, the minimum covariance determinant (MCD)-based method is applied to calculate the log-likelihood which is robust to outlying states. Then, parameters are estimated after removing outlying observations using a mixture-based approach. Experimental results show that the proposed approach can improve the synthetic speech of an HMM-based speech synthesis system with an insufficient database.

2. HMM Training with Decision Tree-Based Clustering

Figure 1 shows a flow chart of the HMM training procedure using decision tree-based clustering [4]. After initialization, model parameters are estimated by embedded training. Following that, the HMM states are tied by decision tree-based clustering, and then model parameters are estimated under the parameter sharing structure. They can be iteratively re-estimated by repeating the same procedure after untying the sharing structure.

For node splitting in decision tree-based clustering, we usually apply the MDL criterion which accounts for both the model specificity and complexity [3]. Let U denote the set of leaf nodes in a decision tree. Then, the description length of U is given by

$$D(U) \equiv -L(U) + dM \log G + C \tag{1}$$

where $L(U)$ is the log-likelihood of the model U, d is the dimensionality of an observation vector, M is the number of leaf nodes, $G = \sum_{m=1}^{M} \Gamma_m$ with Γ_m denoting the summation of the state occupancy probabilities at a leaf node S_m in U, and C is the code length which is here assumed to be

Manuscript received December 28, 2011.
Manuscript revised April 13, 2012.

[†]The authors are with the School of Electrical Engineering and the Institute of New Media and Communications, Seoul National University, Seoul 151–742, Korea.
[*]Presently, with Cowon Systems, Inc., Seoul 135–917, Korea.
a) E-mail: dhhong@hi.snu.ac.kr
b) E-mail: jssung@hi.snu.ac.kr
c) E-mail: khoh@hi.snu.ac.kr
d) E-mail: nkim@snu.ac.kr

DOI: 10.1587/transinf.E95.D.2351

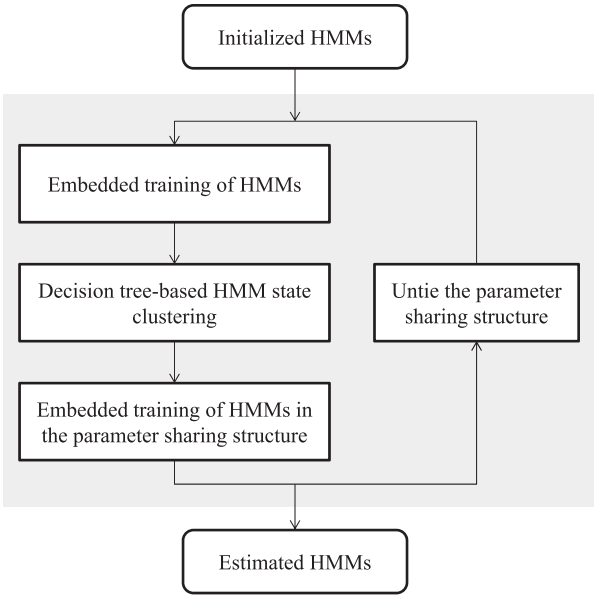

Fig. 1 HMM training with decision tree-based clustering.

constant. It is noted that the description length consists of a likelihood term and a penalty for model complexity. Now suppose that a leaf node S_m is split into S_{mqy} and S_{mqn} according to a binary (yes or no) question q. U' is the set of leaf nodes obtained after splitting the node S_m into two child nodes. Let $\Delta_m(q)$ represent the difference of the description length before and after node splitting with the question q, i.e., $\Delta_m(q) = D(U') - D(U)$. Then, the question $\hat{q}$ which minimizes $\Delta_m(q)$ is selected as the best splitting question for the node S_m. Node splitting is stopped when $\Delta_m(q) > 0$ for all the possible questions. This method provides an efficient way to cluster HMM states [5].

In (1), $L(U)$ indicates the sum of the log-likelihoods for generating observations from the nodes of U. Let $\{s_1^m, s_2^m, ..., s_{L_m}^m\}$ represent the set of HMM states merged into the node S_m where L_m indicates the total number of states. Then, the log-likelihood computed at the node S_m is given by

$$L(S_m) = -\frac{1}{2}\sum_{l=1}^{L_m}\sum_{t=1}^{T_l}\gamma_l^m(t)[(\boldsymbol{o}_t - \boldsymbol{\mu}_m)'\Sigma_m^{-1}(\boldsymbol{o}_t - \boldsymbol{\mu}_m) + d\log 2\pi + \log(\det(\Sigma_m))] \quad (2)$$

where the prime denotes the transpose of a vector or a matrix, $\boldsymbol{o}_t$ is the observation vector at time t, T_l denotes the number of data frames for state s_l^m, $\gamma_l^m(t)$ denotes the a posteriori probability of the state s_l^m at the t-th frame and $\boldsymbol{\mu}_m$, and Σ_m are the mean vector and covariance matrix of the Gaussian distribution at the node S_m, respectively.

3. Training with Outlier Detection and Removal

3.1 Outlier Detection in Decision Tree-Based Clustering

If a node in the decision tree contains outlying HMM states which have extraneous model parameters, the likelihood term in (1) would be distorted. In order to overcome the problems arising from the outlying data in the conventional decision tree-based clustering method, we apply a step which detects the outlying HMM states and calculates the likelihood while ignoring their contribution in the construction of the decision tree. Anomalous states are detected by means of an outlier detection algorithm for multivariate data, which is done for each cluster before node splitting. Observation vectors corresponding to the detected outlying states in the cluster are ignored when calculating the likelihood term $L(S_m)$ in (2) so that the contribution of outliers can be removed or decreased. As a result, we can get a more robust decision tree for speech synthesis.

The outlying HMM states have extraneous model parameters compared to the rest of HMM states at the same node. Treating the mean vector of the output probability distribution as a representative data point of an HMM state, a simple way to detect outliers for multivariate data is to calculate the distance from the centroid of the cluster to each data point. A data point with a distance larger than a predetermined threshold would be a possible outlier. In this work, the distance of a data point $\boldsymbol{x}_i$ from the centroid of the cluster of which mean and covariance are respectively $\boldsymbol{\mu}$ and Σ is defined by:

$$\delta(\boldsymbol{x}_i) = (\boldsymbol{x}_i - \boldsymbol{\mu})'\Sigma^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}). \quad (3)$$

This quadratic form is often called the Mahalanobis distance which is a useful metric to determine the dissimilarity between two sample data [7]. Since, however, some of the data points in the cluster are outliers, it is not easy to obtain robust estimates for the mean and covariance. For that reason, we apply the MCD method which estimates the cluster mean and covariance such that they are resistant to outliers [8], [9].

Consider a set $X = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\}$ of d dimensional data points and let the number of robust observations in the set X be h. The h can be set to any integer satisfying $[n+d+1]/2 \leq h \leq n$, but a good compromise between breakdown value and statistical efficiency is given by

$$h = [(1-p)\times(n+d+1)] \quad (4)$$

where p means a lower bound of the outlier ratio. Given X and h, a fast algorithm to find the local minimum of the covariance determinant is described as follows [9]:

1. Initialize the robust mean $\hat{\boldsymbol{\mu}} := \boldsymbol{\mu}_0$ and covariance $\hat{\Sigma} := \Sigma_0$ where $\boldsymbol{\mu}_0$ and Σ_0 are the classical mean and covariance, respectively.
2. If $\det(\hat{\Sigma}) \neq 0$, compute the distances $\delta(i)$ for $i = 1...n$ by
$$\delta(i) = (\boldsymbol{x}_i - \hat{\boldsymbol{\mu}})'\hat{\Sigma}^{-1}(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}). \quad (5)$$
3. Sort these distances in increasing order, which yields a mapping π for which $\delta(\pi(1)) \leq \delta(\pi(2)) \leq \cdots \leq \delta(\pi(n))$.
4. Put $H := \{\pi(1), \pi(2), \cdots, \pi(h)\}$.
5. Compute $\hat{\boldsymbol{\mu}}$ and $\hat{\Sigma}$ with H according to

$$\hat{\boldsymbol{\mu}} = \frac{1}{h}\sum_{i\in H}\boldsymbol{x}_i, \quad \hat{\Sigma} = \frac{1}{h}\sum_{i\in H}(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}})(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}})^{'}. \tag{6}$$

6. Iterate 2 to 5 until the robust estimates of the mean $\hat{\boldsymbol{\mu}}$ and covariance $\hat{\Sigma}$ of the cluster no longer change.

3.2 Parameter Estimation with Outlier Removal

Although the decision tree can be generated while ignoring the outlying states found by the method described previously, each cluster corresponding to the leaf node still has outlying observation vectors. To address this problem, we apply an outlier removal technique to parameter estimation.

Since the Gaussian mixture model (GMM) is generally used to represent the observation probability at each leaf node of the decision tree, it is appropriate to apply a mixture model-based outlier detection technique to HMM parameter estimation [10]. We propose a step to remove outliers which works after the conventional parameter estimation. Consider a GMM parameter set $\Lambda = \{\lambda_1, \lambda_2, ..., \lambda_M\}$ with each λ_m denoting the K-mixture model parameter set corresponding to the leaf node S_m and being given by

$$\lambda_m = \{w_m^{(1)}, ..., w_m^{(K)}, \boldsymbol{\mu}_m^{(1)}, ..., \boldsymbol{\mu}_m^{(K)}, \Sigma_m^{(1)}, ..., \Sigma_m^{(K)}\} \tag{7}$$

where $w_m^{(k)}$, $\boldsymbol{\mu}_m^{(k)}$, and $\Sigma_m^{(k)}$ are the weight, mean, and covariance of the mixture component k, respectively. In the proposed training algorithm, additional mixture components are trained to estimate the distribution of the outlying observations, and then they are removed so as to get the robust model parameters. The detail of the proposed procedure is described as follows [10]:

1. Estimate Λ by the conventional EM algorithm.
2. Initialize the $2K$-mixture model $\tilde{\Lambda} = \{\tilde{\lambda}_1, \tilde{\lambda}_2, ..., \tilde{\lambda}_M\}$ with each $\tilde{\lambda}_m = \{\tilde{w}_m^{(1)}, ..., \tilde{w}_m^{(2K)}, \tilde{\boldsymbol{\mu}}_m^{(1)}, ..., \tilde{\boldsymbol{\mu}}_m^{(2K)}, \tilde{\Sigma}_m^{(1)}, ..., \tilde{\Sigma}_m^{(2K)}\}$ by splitting each mixture component of λ_m as follows:
$$\begin{aligned}
&\tilde{w}_m^{(k)} = \tilde{w}_m^{(k+K)} = \frac{1}{2}w_m^{(k)},\\
&\tilde{\boldsymbol{\mu}}_m^{(k)} = \tilde{\boldsymbol{\mu}}_m^{(k+K)} = \boldsymbol{\mu}_m^{(k)},\\
&\tilde{\Sigma}_m^{(k)} = \frac{1}{\alpha}\Sigma_m^{(k)}, \quad \tilde{\Sigma}_m^{(k+K)} = \alpha\Sigma_m^{(k)}
\end{aligned} \tag{8}$$
where α is a covariance scaling factor.
3. Iteratively update $\tilde{\Lambda}$ by applying the EM algorithm.
4. Finally, find the robust K-mixture model $\hat{\Lambda}$ by merging the pairs of mixture components of $\tilde{\Lambda}$ according to
$$\begin{aligned}
&j = \underset{i\in\{k,k+K\}}{\operatorname{argmin}}\{\det(\tilde{\Sigma}_m^{(i)})\},\\
&\hat{w}_m^{(k)} = \tilde{w}_m^{(k)} + \tilde{w}_m^{(k+K)},\\
&\hat{\boldsymbol{\mu}}_m^{(k)} = \tilde{\boldsymbol{\mu}}_m^{(j)}, \quad \hat{\Sigma}_m^{(k)} = \tilde{\Sigma}_m^{(j)}.
\end{aligned} \tag{9}$$

Once the above procedure is completed, the contribution of the outliers to estimate the parameters of each cluster is removed or decreased.

4. Experiments

In the experiments, we applied an English speech database spoken by both male and female speakers. The utterances were spoken in four different emotional styles: neutral, angry, joyful, and sad. Each style of the speech database consists of 600 phonetically balanced sentences, including more than 10,000 words. We used 41 phones including silence as basic units of speech synthesis. We also applied quinphone models for which we used the contextual factors listed in [6] and the speech data was labeled automatically by forced alignment using the monophone HMMs. All systems were implemented as a modified version of the HMM-based Speech Synthesis System (HTS) version 2.1.1 [12].

Speech signals were sampled at 16 kHz and windowed by a 25 ms Hamming window with a 5 ms shift. The acoustic features were obtained by STRAIGHT analysis [11]. Feature vectors for HMM training included spectral parameters, excitation parameters, and their first- and second-order derivatives. As for the spectral parameters, we applied 25 mel-cepstral coefficients including the zeroth gain coefficient. The excitation parameters consisted of logarithm of the fundamental frequency and the mean band aperiodicity measure over five frequency bands. A 5-state left-to-right structure with no skips was adopted to represent each context-dependent HMM model. Furthermore, we used hidden semi-Markov model (HSMM) with explicit state duration distribution [13].

We simulated two conditions of data insufficiency to verify the effect of the proposed algorithm on insufficient training data. In the first case, we applied the algorithm to construct an emotion-dependent speech synthesis system where the training database was rather small while it had wider variability in speech characteristics compared to the neutral or reading-style speech. In the second case, we applied a weak text analysis to neutral style speech, for which we artificially removed the part-of-speech information of the original text analysis results. Both of them are considered as representative cases of exuberant variation of speaking style and inadequacy of contextual factors, respectively.

In each experimental condition, we compared the performances between two different models. One was trained by the conventional training technique, and the other was obtained by the proposed technique where the outlier ratio of each cluster was kept to be 10%, i.e. h in (4) was set to about 90% of the number of states in the cluster and the covariance scaling factor α in (8) was set to 2. The number of mixture components of both models was one. We conducted the comparison category rating (CCR) tests [14] to evaluate the quality of the synthetic speech generated from both techniques. In these tests, listeners used the following scale to evaluate the quality of the target speech sample compared to the reference speech: much better (3), better (2), slightly better (1), about the same (0), slightly worse (-1), worse (-2), and much worse (-3). The order of presenting a pair of samples was chosen at random for each trial.

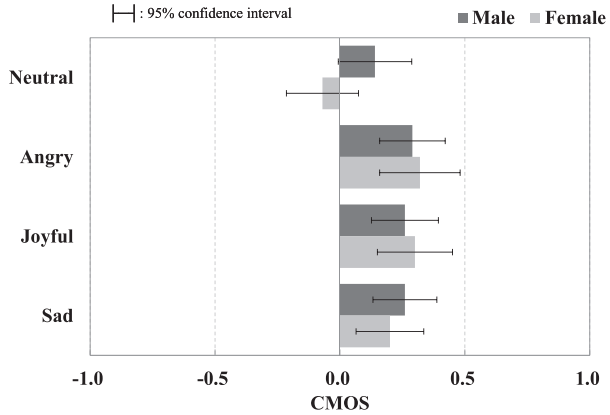

Fig. 2 Scores of CCR test using an emotional speech database.

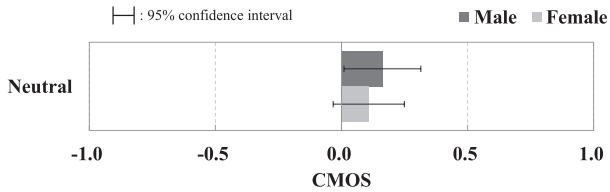

Fig. 3 Scores of CCR test employing a weak text analysis.

For each preference test, thirty sentences, which were not included in the training data, were used and eight native English speakers participated to the subjective quality evaluation. The results obtained in both cases of emotional speech and a weak text analysis are given in Figs. 2 and 3, respectively. From the results, it is shown that the synthetic speech generated from the proposed technique was preferred to the conventional technique in most cases although there is a less significant quality improvement for the neutral speech synthesis, which applied a quite sufficient database. A majority of the listeners remarked that the speech sounds generated from the proposed technique were perceived more natural and clearer and the prosody of them was smoother compared with those obtained from the conventional method.

The size of each decision tree constructed by the proposed technique was 39.6% less on average than that obtained from the conventional technique. This demonstrates that the speech quality of the proposed system could be better than or equal to the conventional system while the footprint of the proposed system was smaller than the conventional system. The training time required for the clustering process of the proposed method, without any strict optimization, was about five times longer than the conventional algorithm. It can be somewhat overcome if a strict optimization or parallel processing [15] is applied.

5. Conclusions

In this letter, we have proposed the outlier detection and removal technique for HMM-based speech synthesis. In this method, outlying states of each cluster are ignored in calculating the likelihood for decision tree-based clustering and outlying observations are removed in parameter estimation. In the experiments, we could confirm that the proposed approach can improve the speech quality of the HMM-based speech synthesis system when given an insufficient speech database.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No. 20110020407) and by the MKE (The Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agency) (NIPA-2012-H0301-12-2005).

References

[1] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," Proc. ICASSP, pp.1315–1318, June 2000.
[2] T. Yoshimura, T. Masuko, K. Tokuda, T. Kobayashi, and T. Kitamura, "Speaker interpolation in HMM-based speech synthesis system," Proc. Eurospeech, pp.2523–2526, Sept. 1997.
[3] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," Proc. Eurospeech, pp.2374–2350, Sept. 1999.
[4] S. Young, J. Odell, and P. Woodland, "Tree-based state tying for high accuracy acoustic modeling," Proc. ARPA Human Language Technology Workshop, pp.307–312, March 1994.
[5] K. Shinoda and T. Watanabe, "Acoustic modeling based on the mdl principle for speech recognition," Proc. EuroSpeech, pp.99–102, Sept. 1997.
[6] K.H. Oh, J.S. Sung, D.H. Hong, and N.S. Kim, "Decision tree-based clustering with outlier detection for HMM-based speech synthesis," Proc. Interspeech, pp.101–104, Aug. 2011.
[7] P.C. Mahalanobis, "On the generalised distance in statistics," Proc. National Institute of Sciences of India 2, pp.49–55, Nov. 1936.
[8] J. Hardin, Multivariate Outlier Detection and Robust Clustering with Minimum Covariance Determinant Estimation and S-Estimation, Doctoral thesis, Uinversity of Califonia, pp.1–55, 2000.
[9] P. Rousseeuw and K. Driessen, "A fast algorithm for the minimum covariance determinant estimator," Technometrics 41, pp.212–223, Aug. 1999.
[10] M. Aitkin and G.T. Wilson, "Mixture models, outliers, and the EM algorithm," Technometrics 22, pp.325–331, Aug. 1980.
[11] H. Kawahara, "Speech representation and transformation using adaptive interpolation of weighted spectrum: Vocoder revisited," Proc. ICASSP, pp.1303–1306, April 1997.
[12] K. Tokuda, H. Zen, J. Yamagishi, T. Masuko, S. Sako, A.W. Black, T. Toda, T. Nose, K. Oura, K. Hashimoto, and S. Shiota, "The HMM-based speech synthesis system (HTS)," http://hts.sp.nitech.ac.jp/, 2010.
[13] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "A hidden semi-Markov model-based speech synthesis system," IEICE Trans. Inf. & Syst., vol.E97-D, no.5, pp.825–834, May 2007.
[14] ITU-T Recommendation P.800 "Methods for subjective determination of transmission quality," Geneva, pp.23–25, May 1996.
[15] N. Pilkington and H. Zen, "An implementation of decision tree-based context clustering on graphics processing units," Proc. INTERSPEECH, pp.833–836, Sept. 2010.