A QUANTITATIVE COMPARISON OF BLIND C_{50} ESTIMATORS

P. Peso Parada[1], D. Sharma[1], J. Lainez[1] , D. Barreda[1], P. A. Naylor[2], T. van Waterschoot[3]

[1]Nuance Communications, Inc. Marlow, UK
[2]Dept. of Electrical and Electronic Engineering, Imperial College London, UK
[3]Dept. of Electrical Engineering (ESAT-STADIUS/ETC), KU Leuven, Belgium
{pablo.peso, dushyant.sharma,jose.lainez,daniel.barreda}@nuance.com,
p.naylor@imperial.ac.uk,toon.vanwaterschoot@esat.kuleuven.be

ABSTRACT

The problem of blind estimation of the room acoustic *clarity index* C_{50} from single-channel reverberant speech signals is presented in this paper. We analyze the performance of several machine learning methods for a regression task using 309 features derived from the speech signal and modeled with a Deep Belief Network (DBN), Classification And Regression Tree (CART) and Linear Regression (LR). These techniques are evaluated on a large test database (86 hours) that includes babble noise and reverberation using both artificial and real room impulses responses (RIRs). All methods are trained on a database which contains noise, speech and simulated RIRs different from the test set. The performance results show that the DBN model gives the lowest error for the simulated RIRs whereas the LR model gives the best generalization performance with the highest accuracy for real RIRs.

Index Terms— C_{50}, CART, DBN, Linear regression

1. INTRODUCTION

The propagation of an acoustic signal from the source to the receiver may follow different paths due to reflections off the walls and other objects in the room. These reflections lead to a reverberant signal at the receiver. This acoustic distortion depends on the location of the source and receiver as well as the room acoustic characteristics [1].

Various metrics exist to measure the reverberation in speech signals [2], which can be used for multiple purposes, from estimation of perceived quality [3] to speech de-reverberation [4] [5] or reverberant speech recognition [6] [7]. These metrics require additional information such as the associated room impulse response (RIR). In many real scenarios, such information is unknown *a priori*, motivating the need for the development of methods for blind estimation of measures of reverberation using only the reverberant signal.

The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° ITN-GA-2012-316969 and from the Research Foundation Flanders.

In this work we propose a method to estimate the *clarity index* (C_{50}) non-intrusively (i.e. blindly, without a reference signal) from single-channel reverberant recordings. This measure was chosen due to the high correlation shown with automatic speech recognition (ASR) performance [8] [9]. C_{50} is the ratio of the energy in the first 50 ms of the RIR after the arrival of the direct path component to the remainder of the energy in the RIR.

Kendrick et al. [10] propose two different methods to estimate room acoustic parameters, including C_{80}, from single-channel speech and music recordings. The first algorithm is based on training an artificial neural network with envelope spectrum features extracted from the power spectral density of the sum of Hilbert envelope per octave band. In the second, the acoustic parameters are obtained by estimating the decay phases in a maximum likelihood framework. In this work we use the first approach as a baseline since it outperforms the second method for speech signals [10]. Originally the method was designed to estimate C_{80} but it has been adapted for comparison purposes in this work to estimate C_{50} instead.

The starting point for this work is the Non-Intrusive Room Acoustic (NIRA) estimation method [8] which employs a Classification And Regression Tree (CART). In this current paper we study firstly the substitution of CART with a linear regression model (LR) and in addition a Deep Belief Network (DBN) .

The remainder of the paper is organized as follows. Section 2 presents details of the methods analysed. We describe in Section 3 the database and evaluation metrics employed to obtain the results and provide a discussion in Section 4. Finally, the main conclusions are drawn in Section 5.

2. C_{50} ESTIMATOR

The C_{50} estimator analysed in this paper is a data-driven approach. The method computes a set 309 features from the reverberant signal. This set includes parameters extracted from the spectrum, Hilbert transformation and pitch of the reverberant signal [8]. These features are then used to train a model

that learns a mapping transformation from the input set of features to the known ground truth C_{50}. Finally, the trained model can be used to estimate unknown C_{50} values. Further details of the features employed can be found in [8]. In the following, we study the comparative performance of 3 different machine learning algorithms in this task. We begin below by summarizing the 3 approaches.

2.1. Classification And Regression Trees

CART [11] provides a framework to build binary trees where the decision at each node is based on one or more input features, and the predicted output is assigned according to the leaf (terminal node) reached at the end of the process. Three main steps characterize the training process of these trees:

- Tree building: each node, starting from the first node (root node), is recursively split into two child nodes which are assigned to a predicted class. The splitter criteria seek to maximize the purity of the nodes [12].

- Stopping tree building: the tree growing process is stopped based on heuristic rules (e.g. only one observation per child node, maximum number of levels reached).

- Tree pruning: nodes with low impact in the accuracy are removed to generalize the model and reduce its complexity. The pruned tree is cross-validated to choose the structure that best fits the data without overfitting the training data.

We use the CART in a regression mode rather than a classification mode since our target is to estimate a room acoustic parameter within a continuous range.

2.2. Linear Regression

In linear regression [13], the prediction $\widehat{C_{50}}$ is calculate as follows:

$$\widehat{C_{50}} = \sum_{j=1}^{M} \theta_j x_j + \theta_0, \tag{1}$$

where $\mathbf{x} = [x_1, \ldots, x_M]^T$ represents the length-M input feature vector and $\boldsymbol{\theta} = [\theta_0, \ldots, \theta_M]$ is a vector with the $M + 1$ linear regression coefficients.

A cost function, defined as the squared error function, is minimized to find the optimal coefficient vector $\boldsymbol{\theta}$. Additionally, we add an L2 regularization term to the cost function to avoid complex and overfitted models. The minimization problem of the cost function is solved by applying the gradient descent algorithm [14].

2.3. Deep Belief Network

A DBN is a neural network that contains at least one hidden layer, with neurons connected between the layers but not within each layer, that undergoes an unsupervised generative initialization. This generative initialization usually gives good generalization capabilities to the DBN in contrast to traditional neural networks that may perform less well on unseen data. The hidden layers are initialized in a greedy layer-wise fashion [15], one layer at a time, that are finally stacked to compose the initialized DBN. This can be seen as a set of stacked feature detectors that have been learnt from an unlabelled training set and are finally adjusted in a supervised discriminative fine-tuning stage from a labelled training set. This combination of generative and discriminative learning stages allows the DBN to perform successfully in classification tasks and is considered as the state of the art in several machine learning classification problems [16] [17].

Although the use of DBNs for classification tasks is well known, a less extensive literature exists for regression problems. Salakhutdinov et al. [18] propose an elaborate method of using DBNs to learn covariance kernels for a Gaussian process, albeit the simplest and most common approach is to add a new layer on top that performs linear regression [19], as in [20] where the authors develop a DBN for the acoustic-articulatory inversion mapping problem.

The DBN presented in this work follows the same philosophy as [20] by adding a linear regression as the output layer. However, our feature detectors are initialized with sparse autoencoders [21] rather than initialized with a stack of Restricted Boltzmann Machines (RBM) [19].

The initialization of the sparse autoencoders is performed from the training set by using the full-batch optimizer limited memory BFGS (L-BFGS) [22]. The fine-tuning stage is carried out with a 100 samples mini-batch stochastic gradient-descent and adaptive momentum. In this paper, the topology of the DBN used comprises 309 nodes in the input layer, 30 and 14 nodes in the first and second hidden layers respectively and 1 node in the output layer determined experimentally.

3. EXPERIMENTAL SETUP

The database and the RIRs used to evaluate the C_{50} are described in [8]. It comprises a training set and testing set. The former is used to train the models and includes 140 simulated RIRs convolved with 32 TIMIT [23] training recordings and babble noise at SNR ∞ dB and from 0 dB to 30 dB in steps of 5 dB. The testing set is created using three different babble noise conditions (SNR at 3 dB, 17 dB and ∞ dB) and 168 TIMIT testing signals convolved with 140 simulated RIRs and with real RIRs from MARDY database [24]. Within this set Sim3, Sim17 and SimInf represent the subsets of reverberant speech obtained with simulated RIRs including babble noise at signal-to-noise ratio (SNR) of 3 dB, 17 dB and ∞ dB

respectively. Likewise, Real3, Real17 and RealInf denote the reverberant sets generated with real measured RIRs and the same noise conditions. The speech signals are first convolved with the appropriate RIRs and then noise is added at the required level to achieve the specified SNR. All the simulated RIRs employed in this experiment are created with the image method [25].

The following evaluation metrics are used to assess the performance of the methods:

- Root mean square deviation (RMSD)

$$E_n = \widehat{\mathrm{C}_{50,\mathrm{n}}} - \mathrm{C}_{50,\mathrm{n}} \text{ dB},$$

$$\mathrm{RMSD} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(E_n)^2} \text{ dB}. \quad (2)$$

- Pearson coefficient (ρ)

$$A_n = \widehat{\mathrm{C}_{50,\mathrm{n}}} - \overline{\widehat{\mathrm{C}_{50}}},\ B_n = \mathrm{C}_{50,\mathrm{n}} - \overline{\mathrm{C}_{50}},$$

$$\rho = \frac{\sum_{n=1}^{N} A_n B_n}{\sqrt{\sum_{n=1}^{N}(A_n)^2 \sum_{n=1}^{N}(B_n)^2}}. \quad (3)$$

- Mean (μ_E) and standard deviation (σ_E) of the estimation error

$$\mu_E = \frac{1}{N}\sum_{n=1}^{N} E_n \text{ dB}, \quad (4)$$

$$\sigma_E = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(E_n - \mu_E)^2} \text{ dB}. \quad (5)$$

- Real time factor (RTF)

$$\mathrm{RTF} = \frac{T_{process}}{T_{signal}}, \quad (6)$$

where $\overline{\mathrm{C}_{50}}$ and $\overline{\widehat{\mathrm{C}_{50}}}$ are the average of N ground truth values $\mathrm{C}_{50,\mathrm{n}}$ and estimated scores $\widehat{\mathrm{C}_{50,\mathrm{n}}}$ respectively, $T_{process}$ is the time necessary to process the speech signal in Matlab on an Intel i5 at 2.67 GHz and T_{signal} is the duration of this signal.

4. RESULTS AND DISCUSSION

The performance metrics are shown in this section for each of the C_{50} estimators: baseline [10], CART [8], LR and DBN.

Figure 1 displays the RMSD for all C_{50} estimators. The baseline shows significantly higher RMSD values for all cases. On the other hand, DBN exhibits the best performance for simulated RIRs however LR gives the best performance for real RIRs.

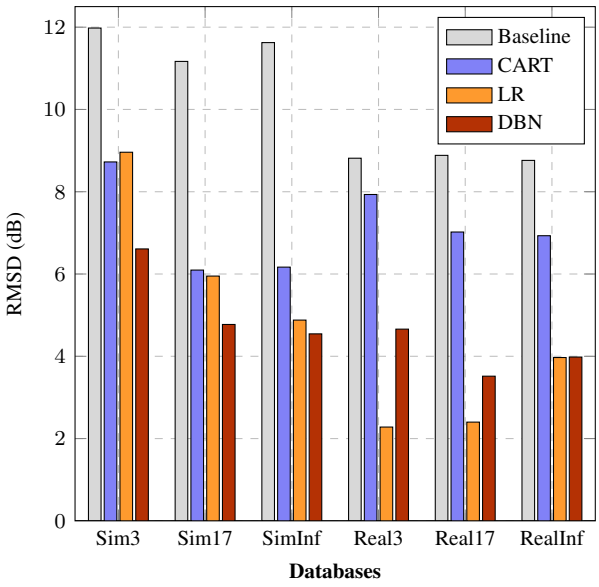

Fig. 1. Root mean square deviation of each method for the different evaluation databases.

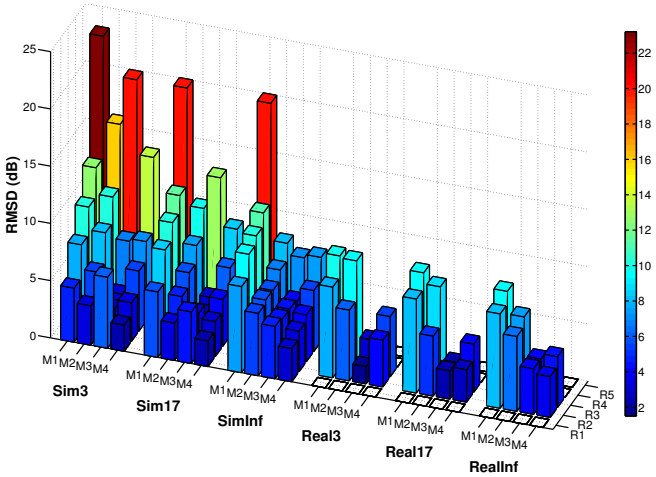

Fig. 2. Root mean square deviation of each method (M1: Baseline; M2: CART; M3: LR; M4: DBN) for the different evaluation databases split in different bands according to the ground truth C_{50} (R1: (-4, 4] dB; R2: (4, 9] dB; R3: (9, 15] dB; R4: (15, 20] dB; R5: (20, 40] dB).

Figure 2 provides more insights into the RMSD performance of the four methods by splitting the deviation into C_{50} bands. Note that the range of C_{50} values in real RIRs ($\approx$ [10, 20] dB) is lower than in simulated RIRs ($\approx$ [-4, 40] dB). The figure suggests that, in general, the methods exhibit higher estimation errors for high values of C_{50}, i.e. low reverberant environments, which is more significant for noisy environments. This might be caused by the sparse C_{50} distribution in this range of the RIRs included in the training set. Additionally, Fig. 2 shows a general trend of RMSD: all methods decrease the RMSD when the C_{50} increases except for the LR method

in Sim3 and Sim17. This method, for the middle range of C_{50}, provides considerably lower RMSD compared to the extreme values (R1 and R5). Consequently, since the real RIRs have C_{50} values in this middle range, LR provides the best estimation in Real3, Real17 and RealInf. The main reason for this behaviour is that LR builds a simple model which cannot model all C_{50} values, only a narrow range of C_{50}, and the cost function is minimized by adapting the model to the middle range of C_{50}.

The correlation coefficient was used to examine the linear relationship between the estimated C_{50} and the ground truth. Figure 3 presents these coefficients for each method. The maximum correlation is achieved with DBN in simulated RIRs without noise, $\rho = 0.93$, suggesting a very strong relationship between the estimates and the ground truth. In contrast with Fig. 1, all methods provide, in terms of correlation, higher performance with simulated RIRs rather than with real RIRs where they exhibit a moderate-strong correlation (excluding the baseline).

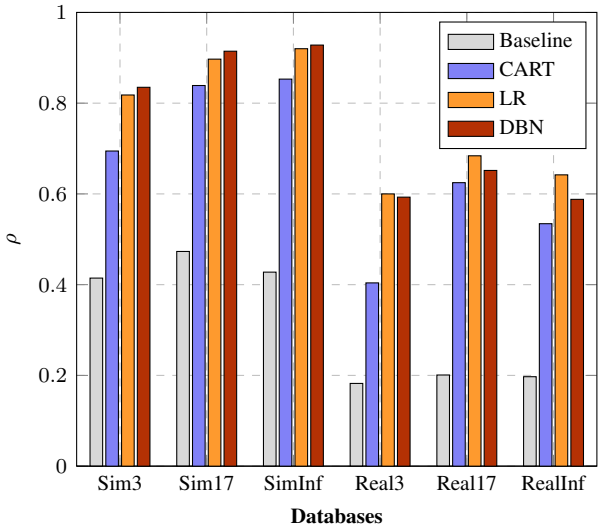

Fig. 3. Correlation of each method for the different evaluation databases.

Figure 4 investigates the properties of the estimation error histogram by computing two statistics from it: mean and standard deviation. Overall, the statistic μ_E displays a decreasing trend with noise. Concerning the σ_E, all methods show higher deviations with simulated RIRs due to the range of C_{50} values considered in this set except for CART in Real17, where it exhibits a slightly higher deviation compared to Sim17. Moreover, these deviations increase with the noise level for simulated RIRs whereas the deviations in real RIRs are barely affected by the noise.

Even though the source to estimate the ground truth C50 is speech, consequently not all RIR frequency bands might be excited, our method is able to provide reasonable results.

The real time factor provides an indication of the relative

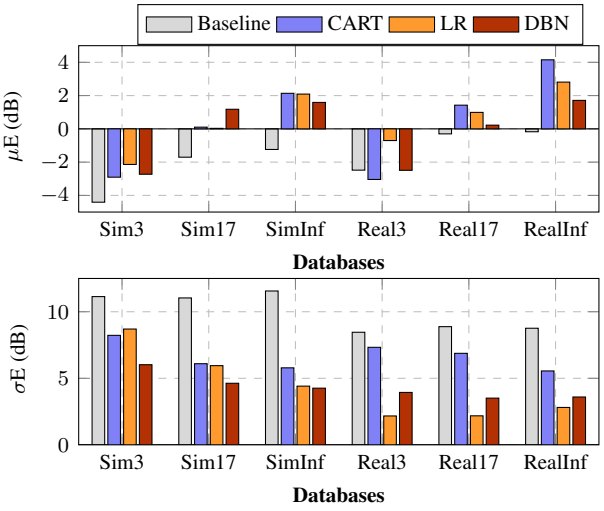

Fig. 4. Mean (upper plot) and standard deviation (lower plot) of each method for the different evaluation databases.

complexity of the analysed method. The dominant source of complexity of the presented C_{50} estimators resides in the feature extraction. The usage of the model to estimate C_{50} from the feature vector adds only a small overhead which does not depend on the signal time length. The baseline method exhibits a RTF of 0.024 while the RTF for CART, LR and DBN is 0.205 in our Matlab implementations. The former method computes 40 features and the latter methods extract 309 features which causes the RTF difference. These results suggest that the methods have the potential for real time implementation. It can be noted that the CART model performs a feature sub-selection which leads to only 118 of the 309 features being used to perform the estimation in this experiment. Due to this characteristic, CART RTF would be reduced in a further implementation that computes only the features in the sub-selection set.

5. CONCLUSIONS

We have presented a comparison of four methods to estimate C_{50}: a baseline method based on envelope spectrum features and the NIRA [8] method based on 309 features computed from the signal which are employed to train the different models (CART, LR and DBN). The experiments were performed on simulated and real RIRs including babble noise at different SNR levels. We have shown that the new proposed models (LR and DBN) to estimate C_{50} significantly improve the NIRA [8] method. On average, DBN reduces the RMSD by 1.7 dB for simulated RIRs and LR reduces the RMSD by 4.4 dB for real RIRs. This deviation was shown to be dependent on the targeted C_{50} in all methods. In terms of the correlation coefficient, increase for both cases is approximately 0.1. Additionally, we have shown that all C_{50} estimators studied can potentially be implemented in real-time applications.

6. REFERENCES

[1] P. A. Naylor and N. D. Gaubitch, "Speech dereverberation," in *Proc. Intl. Workshop Acoust. Echo Noise Control (IWAENC)*, 2005.

[2] P. A. Naylor and N. D. Gaubitch, Eds., *Speech Dereverberation*, Springer, 2010.

[3] J. M. F. del Vallado, A. A. de Lima, T. de M. Prego, and S. L. Netto, "Feature analysis for the reverberation perception in speech signals," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 8169–8173.

[4] R. Gomez and T. Kawahara, "Dereverberation based on wavelet packet filtering for robust automatic speech recognition," in *Proc. INTERSPEECH*, 2012.

[5] A.W. Mohammed, M. Matassoni, H. Maganti, and M. Omologo, "Acoustic model adaptation using piecewise energy decay curve for reverberant environments," in *Proc. of the 20th European Signal Processing Conference (EUSIPCO)*, 2012, pp. 365–369.

[6] P. Peso Parada, D. Sharma, P. A. Naylor, and T. van Waterschoot, "Single-channel reverberant speech recognition using C50 estimation," in *2014 REVERB Challenge Workshop (REVERB'14)*, Florence, Italy, May 2014.

[7] L. Couvreur, C. Ris, and C. Couvreur, "Model-based blind estimation of reverberation time: application to robust ASR in reverberant environments.," in *Proc. INTERSPEECH*, 2001, pp. 2635–2638.

[8] P. Peso Parada, D. Sharma, and P. A. Naylor, "Non-intrusive estimation of the level of reverberation in speech," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014.

[9] A. Tsilfidis, I. Mporas, J. Mourjopoulos, and N. Fakotakis, "Automatic speech recognition performance in different room acoustic environments with and without dereverberation preprocessing," *Computer Speech & Language*, vol. 27, no. 1, pp. 380–395, 2013.

[10] P. Kendrick, T. J. Cox, F. F. Li, Y. Zhang, and J. A. Chambers, "Monaural room acoustic parameters from music and speech," *The Journal of the Acoustical Society of America*, vol. 124, no. 1, pp. 278–287, 2008.

[11] L. Breiman, J. Friedman, C. J. Stone, and R.A. Olshen, *Classification and Regression Trees*, CRC Press, 1984.

[12] W. Loh, "Classification and regression trees," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 1, no. 1, pp. 14–23, 2011.

[13] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer-Verlag New York, Inc., 2006.

[14] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, John Wiley and Sons, second edition, 2001.

[15] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Proc. Advances in neural information processing systems (NIPS)*. 2007, vol. 19, pp. 153–160, MIT Press.

[16] A. Mohamed, G. E. Dahl, and G. Hinton, "Acoustic modeling using deep belief networks," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 14–22, 2012.

[17] N. Lopes and B. Ribeiro, "Towards adaptive learning with improved convergence of deep belief networks on graphics processing units," *Pattern Recognition*, vol. 47, no. 1, pp. 114–127, 2014.

[18] R. Salakhutdinov and G. Hinton, "Using deep belief nets to learn covariance kernels for Gaussian processes," in *Advances in Neural Information Processing Systems (NIPS)*. 2008, vol. 20, pp. 1249–1256, MIT Press.

[19] G. Hinton, "A practical guide to training restricted boltzmann machines," Tech. Rep. UTML TR 2010-003, Department of Computer Science, University of Toronto, 2010.

[20] B. Uria, S. Renals, and K. Richmond, "A deep neural network for acoustic-articulatory speech inversion," in *Proc. Advances in neural information processing systems (NIPS)*. 2011, MIT Press.

[21] A. Ng, "Sparse autoencoder," *Stanford University, CS294A Lecture notes*, 2011.

[22] D. C. Liu and J. Nocedal, "On the limited memory bfgs method for large scale optimization," *Mathematical programming*, vol. 45, no. 1-3, pp. 503–528, 1989.

[23] J. S. Garofolo, "Getting started with the DARPA TIMIT CD-ROM: An acoustic phonetic continuous speech database," Technical report, National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, Dec. 1988.

[24] J. Wen, N. D. Gaubitch, E. A. P. Habets, T. Myatt, and P. A. Naylor, "Evaluation of speech dereverberation algorithms using the MARDY database," in *Proc. Intl. Workshop Acoust. Echo Noise Control (IWAENC)*, 2006.

[25] J. B. Allen and D. A Berkley, "Image method for efficiently simulating small-room acoustics," *The Journal of the Acoustical Society of America*, vol. 65, pp. 943–950, 1979.