KERNELIZED RÉNYI DISTANCE FOR SPEAKER RECOGNITION

Balaji Vasan Srinivasan, Ramani Duraiswami, Dmitry N. Zotkin

Perceptual Interfaces and Reality Laboratory, Institute for Advanced Computer Studies (UMIACS),
University of Maryland, College Park, MD 20742, USA
[balajiv,ramani,dz]@umiacs.umd.edu

ABSTRACT

Speaker recognition systems classify a test signal as a speaker or an imposter by evaluating a matching score between input and reference signals. We propose a new information theoretic approach for computation of the matching score using the Rényi entropy. The proposed entropic distance, the Kernelized Rényi distance (KRD), is formulated in a non-parametric way and the resulting measure is efficiently evaluated in a parallelized fashion on a graphical processor. The distance is then adapted as a scoring function and its performance compared with other popular scoring approaches in a speaker identification and speaker verification framework.

Index Terms— Rényi entropy, similarity score, speaker recognition, GPU, fast algorithms

1. INTRODUCTION

Speaker recognition [1] is a broad category of speech-based learning, which includes *speaker identification* and *speaker verification*. The goal of a speaker recognition system is to identify speakers using digital samples of their voice and to verify the speaker's authenticity. Speaker identification deals with determining the speaker corresponding to a particular voice sample by classifying it into one of a predefined set of reference speakers. Speaker verification systems authenticate the claimed identity of a speaker using the voice sample. Speaker recognition can either be *text-dependent* or *text-independent*.

Fig. 1 shows a generic text-independent speaker recognition system that will be used in this paper. The first step in any recognition system is to extract feature vectors from the speech signals (reference and test). Once this is done, there are many approaches to build the speaker model. Gaussian Mixture Models (GMM) [2] build a semi-parametric model in the feature space and are one of the widely used approaches in speaker recognition. Alternatively, it is possible to measure the distance between the feature vectors from the reference and test signals [3]. An advantage of such an approach is very low training time. We use such an approach in this paper. As shown in Fig. 1, a scoring function is used to quantify the measured distance between the reference and test spaces. The main task of the scoring function in a recognition system is to find the similarity (or dissimilarity) between the reference and test signal space and quantify this using a matching score. The matching score can be used to authenticate based on a threshold or to classify a speaker using k-nearest neighbor classifier.

There have been several information-theoretic and statistical measures that have been used to measure scores between speech signals. Second-order statistical measures [3] like sphericity and Gaussian likelihood evaluate similarity scores using only the mean and variance of feature vectors. Soong et al. [4] use a vector quantizer based codebook along with the Euclidean distance to compare speech signals. Information theoretic measures like Kullback-Leibler (KL) divergence and Bhattacharya distance have also been used in the speaker recognition framework [1]. However, the underlying feature distributions are assumed to be Gaussian in all these works. This can be limiting when the underlying distribution is actually deviates from a Gaussian. Semi-parametric Gaussian mixture models [2] address this issue to some extent and are popularly used in speaker recognition. However, a key drawback with semiparametric and non-parametric approaches is the associated computational complexity, which makes them unsuitable for large problems. In this paper, we propose a completely **non-parametric Rényi entropy based similarity measure** and adapt it to create a scoring function between speech signals. We address the computational complexity of our non-parametric scoring function with the use of a graphical processor based parallel algorithm.

The paper is organized as follows. We first derive the non-parametric measure from the classic Rényi entropy [5] and use kernel density estimates to simplify the resulting measure into an analytical form in Section 2. We then introduce the proposed acceleration approach to scale the measure for larger problems in Section 3. We compare the performance of the new measures in a speaker verification and identification problems in Section 4. The paper concludes with a discussion of the results and of possible further work.

2. KERNELIZED RÉNYI DISTANCE

Entropy of a random variable measures the amount of information contained in the distribution. The widely used Shannon entropy of a random variable X is given by $H(X) = -\int p(x) \log p(x) dx$. The $p(x)$ here is the density of the random variable X. In a speech framework, the X can be thought as a feature vector extracted from a single frame of a speech signal. The Shannon entropy is a specific case of a more generalized family of Rényi entropies. The Rényi entropy [5] of order α is given by $H_\alpha(x) = \frac{1}{1-\alpha} \log \int p(x)^\alpha dx$.

As $\alpha \to 1$, the Rényi entropy reduces to the Shannon entropy. This case is special since for $\alpha = 1$ the entropy of a joint probability distribution can be separated into the entropies of the individual random variables of the joint distribution [5]. This, coupled with the tractability of the measures for the common parametric distributions, has made the Shannon entropy the preferred choice for many problems. However, the Shannon entropy has certain deficiencies. For data known via sampling, the nonparametric Shannon measure is relatively expensive to calculate and is often computed via histograms or order statistics, which leads to biased estimates [6, 7]. Ref. [6] shows that Shannon entropy estimation approaches do not converge to the actual sample entropy even after bias compensation and further shows the need for $1 < \alpha < 4$ to achieve convergence of the sample entropy to the actual entropy. Motivated by this, we derive our measure from the Rényi entropy with $\alpha = 2$. Throughout this paper, *Rényi entropy* will denote the case of $\alpha = 2$.

The Rényi entropy is given by

$$H_2(x) = -\log \int p(x)^2 dx. \tag{1}$$

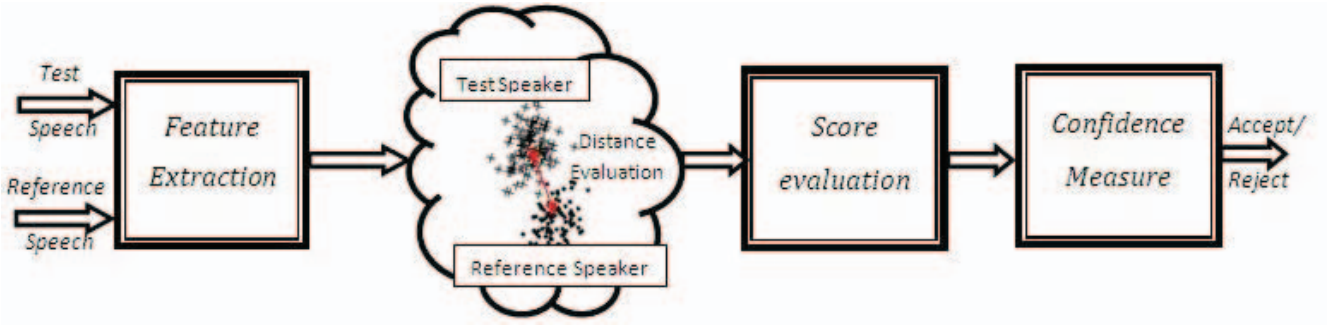

Fig. 1. *A modular representation of a generic speaker recognition system.*

If $p(x)$ is known, the entropy can be computed by solving the integral above. In many practical scenarios, the density is not known and has to be estimated from samples of the distribution. Density estimation can be parametric and non-parametric. In the parametric case, a particular form for the density is assumed and the parameters associated with the form are estimated from the samples (e.g., via expectation-maximization). A non-parametric approach to density estimation uses a kernel window and estimates the density as a sum of kernel functions of the available samples from the distribution. Using kernel density estimation for $p(x)$ as in [8], we get

$$p(x) = \frac{1}{N}\sum_{i=1}^{N} K(x, x_i), \tag{2}$$

where $K(x, x_i)$ is a kernel function. Quite often the Gaussian kernel $K(x_1, x_2) = \frac{1}{h\sqrt{2\pi}} \exp\left(-\frac{|x_1 - x_2|^2}{h^2}\right)$ is used, where h is the bandwidth that must be selected according to the data. The non-parametric approach is preferred when the underlying distribution is unknown; assuming that sufficient number of samples is available, it is generic and will not provide biased estimates. Using Eq. (2) for $p(x)$ in Eq. (1),

$$\begin{aligned} H_2(x) &= -\log \int \left(\frac{1}{N}\sum_{i=1}^{N} K(x, x_i)\right)^2 dx \qquad (3) \\ &= -\log \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} \int K(x, x_i)K(x, x_j)dx. \end{aligned}$$

For the Gaussian kernel, $\int K(x, x_i)K(x, x_j)dx = \hat{K}(x_i, x_j)$, where $\hat{K}$ is also a Gaussian kernel with bandwidth equal to the sum of the bandwidths of the two Gaussian kernels [9]. Using this relation in Eq. (3), we get $H_2(x) = -\log\left(\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\hat{K}(x_i, x_j)\right)$.

Consider two distinct distributions with densities p and q defined by two sets of data points $D_p = \{x_{p1}, \ldots, x_{pN}\}$ and $D_q = \{x_{q1}, \ldots, x_{qM}\}$, respectively. The distance between $p(x)$ and $q(x)$ is

$$H_2(p\|q) = -\log\left(\frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}\hat{K}(x_{pi}, x_{qj})\right). \tag{4}$$

This is called Rényi Information Potential [8]. We will refer to this measure as the **Kernelized Rényi Distance (KRD)** and will use it for calculating matching scores. A variant of this measure (Eq. 4) have been used in visual tracking [7] and object recognition [10]. In this paper, we adapt Eq. (4) to measure similarities between speech samples.

KRD between speech samples: The KRD in Eq (4) represents the distance between two distributions $p(x)$ and $q(x)$. In order to use this as a scoring function, it is necessary to formulate the speech signals (reference and test) as samples from distributions. The feature selection in the recognition system extracts features from multiple overlapping frames of the speech signal. Suppose there are N and M overlapping frames in the reference and the test signal respectively, and d features are extracted. Then we have $N \times d$ vector and $M \times d$ vector representing the reference and the test signals, respectively. We formulate this feature set to be samples drawn from the corresponding feature distribution of the speaker. Using Eq. (4), we can thus evaluate the matching score.

3. ACCELERATING KRD EVALUATION VIA GPUS

Evaluating the KRD between two distributions, each represented by N data points, would require $O(N^2)$ operations. It should be noted that the core computation in Eq. (4) is summation of the Gaussian kernel. Yang et al. [7] used improved fast Gauss transform [11] (IFGT) to solve this computational issue. Srinivasan et al. [10] used FIGTREE [12], which combines IFGT with a tree-based approach for improved performance, to accelerate the summation. Both these approaches (IFGT and FIGTREE) do not scale well to high-dimensional problems typically encountered with speech processing. We propose to use graphical processing units (GPU) and NVIDIA CUDA toolkit [13] to accelerate the summation.

Graphics processors were designed to cater to the demands of real-time high-definition graphics. A GPU is a highly parallel, multi-threaded, multi-core processor with tremendous computational power and high memory bandwidth. GPUs are particularly well-suited for data parallel computation and are designed as a single-program-multiple-data (SPMD) architecture with very high arithmetic intensity (ratio of arithmetic operation to memory operations). In November 2006, NVIDIA introduced *Compute Unified Device Architecture (CUDA)* [13], a parallel programming model that leverages the parallel computing engine in NVIDIA GPUs to solve general purpose computational problems. With CUDA, GPUs can be seen as a bunch of parallel co-processors that can assist the main processor in the computations. We assign each GPU processor (i.e., a GPU execution thread) to the task of evaluating the sum corresponding to a particular i in Eq. (4).

GPU Speedup: We compared the accelerations in distance evaluation using our GPU-based approach and using direct evaluation for a 39-dimensional data. All our experiments were performed on an Intel quad-core processor and the GPU used was the 240-core NVidia GTX280. The GPU algorithm was written in CUDA [13] with Matlab linkages, and the direct summation was written in C++ with Matlab linkages as well. We evaluate the score (Eq. 4) between randomly generated data points using direct summation and

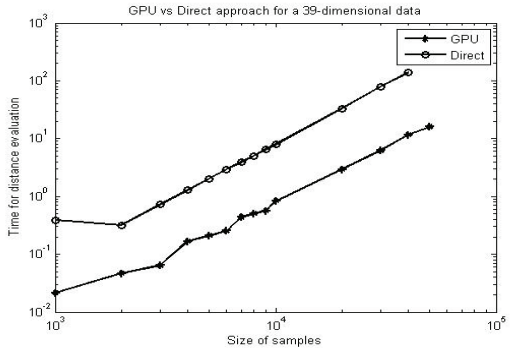

Fig. 2. *Comparison between our GPU-based approach and a direct approach to evaluating Eq. (4).*

our GPU-based approach. The timing results are shown in Fig. 2. It is evident that for large amounts of high-dimensional data GPU-based approach is significantly faster than the direct approach.

4. EXPERIMENTS

Before illustrating the performance of our scores with speaker recognition problems, we first validate the proposed measure with other information theoretic scores to show the unbiased nature of the proposed measure.

4.1. Score Validation

Hegde et al. [6] theoretically show the bias in sample-based entropies for $\alpha < 1$ in the generalized Rényi entropy. In this experiment, we illustrate this fact using synthetic data and also show that the proposed measure is not biased when evaluated from samples. We first generate samples from 2 Gaussian distributions $N(\mu, \mathbf{I})$ and $N(-\mu, \mathbf{I})$, where $\mu = \{1, \ldots, 1\}$ and $\mathbf{I}$ is the identity matrix, for dimensions 1 to 10. We then evaluate the proposed KRD between samples for all the dimensions along with the KL divergence based on the samples. For comparison we also evaluate the analytically computed KL-divergence between Gaussian distribution. As the dimensionality increases, the distance between distributions increases; this is expected to be reflected in the corresponding measure. The normalized distance scores are shown in Fig. 3. It can be seen that our KRD score computed directly from the samples compares favorably with analytically computed distance, whereas the KL divergence computed from samples is biased when number of dimensions is high. A similar trait is also observed with the popular Bhattacharya distance as well [7].

4.2. Dataset and Features

For all our experiments we used the speech signals from the TIMIT database, which consists of data from 630 different speakers. TIMIT is a noise-free database hence providing an unbiased platform to evaluate the performance of our measure.

We extracted 13 MFCC coefficients from 25ms speech frames with 10ms overlap. For all our experiments, we normalized the feature vectors using the Znorm [14] (except for the approaches that used only the first and second order statistics of the feature vectors). Although there are more complex sets of features used for speaker recognition (e.g., the above features can be augmented with velocity and energy and then pruned via split vector quantization), here we used just the MFCC because the objective was the comparison of distance measures. The method is of course generic enough to be extended with other features, and benefits obtained from GPU implementation are very significant for a number of dimensions more than 13 we used here.

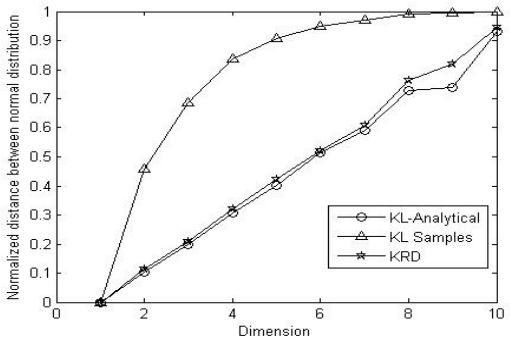

Fig. 3. *Entropic distances between distributions for varying dimensionality. Distances are evaluated analytically (based on the underlying distribution) and from samples (based on the density estimates).*

4.3. Experiment 1: Likelihood Ratio for Speaker Verification

Speaker verification system accepts a sample X as a speaker S if the likelihood ratio $\frac{P(X|S)}{P(X|S')} > T$, where T denotes a threshold. The likelihood $P(X|S)$ denotes the probability that the features from the sample X were generated by speaker S. Similarly, $P(X|S')$ denote the probability the features are from an imposter. The threshold T can be adjusted so that the false acceptance rate (FAR) (an imposter being identified as a speaker) and the false rejection rate (FRR) (a speaker being rejected as an imposter) are equal. We used this Equal Error Rate (EER) to evaluate the performance of our measure.

We compared our scoring function with the Gaussian-likelihood measure [3] (GaussLL), Euclidean distance between vector quantized codebook [4] (VQ), KL-based measure [1] (KL_a), KL-scores evaluated from the samples (KL_s), and GMM-UBM based score [2]. The Matlab *kmeans* function was used to build the codebook of size 50. The GMM was built using statistical toolbox in Matlab, and number of mixtures was chosen to be 32 with diagonal covariance for each speaker. The universal background model [2] (UBM) for the imposter was built by collecting feature samples from a large number of speakers in the database. For the GMM, the UBM had 256 mixtures, whereas for other measures the entire set of UBM samples was used.

We evaluate each of the above scores for a test signal with respect to the reference speaker and imposter speaker models and compute the ratio between the two, which is then used for threshold comparisons. The equal error rate obtained in this way is reported in Table 1 for each of the scores. It can be seen that the proposed scoring function outperforms the other approaches in all the cases.

In Table 1, we have also reported the average time taken to evaluate the score between two sets of feature vectors (speaker/imposter). The measures *GaussLL* and KL_s take the least time. However, these measures use only the first and second order statistics for score evaluation and hence inexpensive to compute. While our score is more expensive, it still takes less time than all advanced approaches (VQ, GMM, and KL_a). It should be noted that the improvement of our measure relative to the GMM-UBM [2] is roughly equivalent to that attained in NIST SRE 2006 [15].

4.4. Experiment 2: k-NN for Speaker Identification

In speaker identification problem, the speaker is known *a priori* to be a member of a set of N speakers and a new test sample must be classified into one of N classes. In this experiment, we used our

No. of speakers	VQ	KL_s	KL_a	GaussLL	GMM	KRD
Time	0.7s	4.5s	0.03s	0.04s	0.4s	0.16s
50	2.57	2.00	2.00	2.00	2.00	**2.00**
100	2.80	2.57	2.43	2.71	3.00	**2.40**
150	3.75	3.33	3.20	3.00	5.67	**2.80**
200	4.00	3.67	3.67	3.50	6.50	**3.50**
250	4.00	4.00	3.75	4.00	7.00	**3.50**
300	4.50	5.00	4.50	4.33	7.83	**4.00**
350	5.33	6.67	6.67	6.00	8.00	**4.67**

Table 1. *EER for various methods in speaker verification experiment. Time reported is the average time of one score evaluation. Time to build the imposter models for GMM and VQ is not included.*

No. of speakers	VQ	GaussLL	SVM	KRD
10	96.00	94.00	94.00	**96.00**
25	90.40	91.20	82.40	**92.00**
50	70.67	73.87	66.80	**78.40**
75	64.40	71.60	61.07	**74.40**
100	54.80	63.20	55.80	**64.80**

Table 2. *Classification accuracy for various methods in speaker identification experiment.*

KRD measure with a 3-nearest neighbor classifier for speaker identification. We repeated the experiment with the GaussLL and VQ measures also using the 3-nearest neighbor classifier. We also built an SVM (with radial basis function kernel) based speaker identification system [16] for comparison.

For each case, we use 5 samples for each speaker to do the training and the remaining samples for testing. We evaluated the performance of each of the approaches for 10, 25, 50, 75, and 100 classes. The classification results are shown in Table 2. It can be seen that the proposed approach performs better than the other approaches for all the cases.

5. CONCLUSIONS

We have proposed a new Rényi entropy based measure for evaluating matching scores between speech signals. We illustrated the unbiased nature of our non-parametric distance with a synthetic example. We also showed how the KL-divergence is biased when estimated from the samples.

We used GPU programming to accelerate distance evaluation for our metric. There are many computations that are very similar to those in Eq. (4). For example, in GMM it is often essential to evaluate the PDF of a number of points in the Gaussian mixture distribution. It is possible to utilize the power of graphical processors in these cases for efficient evaluations similar to the approach in this paper.

Finally, the measure was adapted into a scoring function and used in a speaker verification and identification system. The results compare favorably with scores based on popular approaches, and further improvements are ongoing. It would be interesting to analyze the performance of our scores over a state-of-the-art *joint factor analysis* based feature transformations [17].

6. REFERENCES

[1] J. P. Campbell, "Speaker recognition: a tutorial," *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1437–1462, Sep 1997.

[2] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," in *Digital Signal Processing*, 2000.

[3] F. Bimbot, I. Magrin-Chagnolleau, and L. Mathan, "Second-order statistical measures for text-independent speaker identification," *Speech Communication*, vol. 17, pp. 51–54, 1995.

[4] F. Soong, A. Rosenberg, L. Rabiner, and B. Juang, "A vector quantization approach to speaker recognition," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Apr 1985, vol. 10, pp. 387–390.

[5] Rényi A., "On measures of information and entropy," *Proceedings of the 4th Berkeley Symposium on Mathematics, Statistics and Probability*, pp. 547–561, 1960.

[6] A. Hegde, T. Lan, and D. Erdogmus, "Order statistics based estimator for Renyi entropy," *IEEE Workshop on Machine Learning for Signal Processing*, pp. 335–339, Sept. 2005.

[7] C. Yang, R. Duraiswami, and L. Davis, "Efficient mean-shift tracking via a new similarity measure," in *IEEE International Conference on Computer Vision and Pattern Recognition*, June 2005, vol. 1, pp. 176–183.

[8] E. Gokcay and J. C. Principe, "Information theoretic clustering," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 158–171, Feb 2002.

[9] D. Xu, J. C. Principe, J. Fisher, and H. Wu, "A novel measure for independent component analysis (ICA)," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 1998, vol. 2, pp. 1161–1164.

[10] B. V. Srinivasan and R. Duraiswami, "Efficient subset selection via the kernelized Rényi distance," in *12th IEEE International Conference on Computer Vision*, September 2009.

[11] VC. Raykar and R Duraiswami, "The improved fast Gauss transform with applications to machine learning," in *Large Scale Kernel Machines*, pp. 175–201. MIT Press, 2007.

[12] V. Morariu, B. V. Srinivasan, V. C. Raykar, R. Duraiswami, and L. Davis, "Automatic online tuning for fast Gaussian summation," in *Advances in Neural Information Processing Systems*. 2008, MIT Press.

[13] NVIDIA, *NVIDIA CUDA Programming Guide 2.0*, 2008.

[14] F. Bimbot, J. F. Bonastre, C. Fredouille, G. Gravier, M. I. Chagnolleau, S. Meignier, T. Merlin, O. J. Garcia, D. Petrovska-Delacrétaz, and D. A. Reynolds, "A tutorial on text-independent speaker verification," *EURASIP Journal on Applied Signal Processing*, vol. 4, pp. 430–451, 2004.

[15] D. E. Sturim, W. M. Campbell, D. A. Reynolds, R. B. Dunn, and T. F. Quatieri, "Robust speaker recognition with cross-channel data: Mit-ll results on the 2006 nist sre auxiliary microphone task," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, April 2007, vol. 4, pp. 49–52.

[16] V. Wan and W. M. Campbell, "Support vector machines for speaker verification and identification," in *IEEE Signal Processing Society Workshop on Neural Networks for Signal Processing*, 2000, vol. 2, pp. 775–784.

[17] N. Dehak, P. Kenny, R. Dehak, O. Glembek, P. Dumouchel, L. Burget, V. Hubeika, and F. Castaldo, "Support vector machines and joint factor analysis for speaker verification," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2009, pp. 4237–4240.