KERNEL-BASED SPEAKER RECOGNITION WITH THE ISIP TOOLKIT

Joseph Picone, Tales Imbiriba, Naveen Parihar,
Aldebaro Klautau and Sridhar Raghavan
Department of Mathematical Modeling, Building 321
Technical University of Denmark, DK-2800 Lyngby, Denmark
Phone: +45 4525 3923,3921
Fax: +45 4587 2599
E-mail: jl,cg@imm.dtu.dk
Web: eivind.imm.dtu.dk

Abstract. Notes:

Picone: We had planned by now to do some SVM/RVM speaker recognition experiments. This would be an excellent topic for collaboration. We could write about the toolkit and core technology, and you could write about your experiments.

Aldebaro: the paper must be at most 10-pages long.

INTRODUCTION

Speaker recognition: verification / identification. GMM as baseline. Universal Background Model. Recent work using BaseIME (the corpus we have).

Identification can be seen as several verifications.

New baseline for BaseIME. Open-source. Reproducible results.

CLASSIFIERS IN SPEAKER VERIFICATION

The speaker recognition problem is closely related to conventional supervised classification. Hence, we start by discussing few related definitions. In the classification scenario one is given a *training set* $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$ containing N *examples*, which are considered to be independently and identically distributed (iid) samples from an unknown but fixed distribution $P(\mathbf{x}, y)$. Each example $(\mathbf{x}, y)$ consists of a vector $\mathbf{x} \in \mathcal{X}$ of dimension L (called instance) and a label $y \in \{1, \ldots, Y\}$. A *classifier* is a mapping $F : \mathcal{X} \rightarrow \{1, \ldots, Y\}$. Of special interest are binary classifiers, for which $Y = 2$, and for mathematical convenience, sometimes the labels are $y \in \{-1, 1\}$ (e.g., SVM) or $y \in \{0, 1\}$ (e.g., RVM).

Some classifiers are able to provide *confidence-valued scores* $f_i(\mathbf{x})$, for each class $i = 1 \ldots Y$. A special case is when the scores correspond to a distribution over the labels, i.e., $f_i(\mathbf{x}) \geq 0$ and $\sum_{i=1}^{Y} f_i(\mathbf{x}) = 1$. Commonly, these classifiers use the *max-wins rule* $F(\mathbf{x}) = \arg_i \max f_i(\mathbf{x})$. When the classifier is binary, only a single score $f(\mathbf{x}) \in \mathbb{R}$ is needed. For example, if $y \in \{-1, 1\}$, the final decision $F(\mathbf{x}) = \text{sign}(f(\mathbf{x}))$ is the sign of the score.

In speech verification systems, the input consists of a matrix $\mathbf{X} \in \mathbb{R}^{T \times Q}$ that corresponds to a *segment* of speech. The number T of rows is the number of *frames* (or blocks) of speech and Q (columns) is the number of parameters representing each frame. If T is fixed (e.g., corresponding to 5 milliseconds), $\mathbf{X}$ could be turned into a vector of dimension $T \times Q$, and one would have a conventional classification problem. However, in unrestricted-text speech verification, any comparison between elements of two such vectors could fail if they represent different sounds. Hence, verification systems often adopt alterative architectures, which are similar, but do not fit exactly the stated definition of a classifier. We organize the most popular architectures as follows.

Architectures for speaker recog. based on classifiers

a) Generative. Classical architecture using Fisher kernel, as proposed in [3] and applied to speech verification, e.g., in Nathan Smith (http://mi.eng.cam.ac.uk/ nds1002/), Vincent Wan (http://www.dcs.shef.ac.uk/ vinny/svmsvm.html), etc.

b) Segmental. An alternative to this approach is to use segmental features as done by ISIP for speech recognition. Need to elaborate a bit on that.

c) Accumulative. Another alternative is to train a *frame-based* system, where the input to confidence-valued classifiers is a frame $\mathbf{x}_t$ with dimension $L = Q$, and the total score of class i is $\frac{1}{T} \sum_{t=1}^{T} f_i(\mathbf{x}_t)$. This is the approach adopted in the conventional GMM-based system. In this case, the number N of training examples can be too large.

Gaussian Mixture Model

The GMM is the most popular technique adopted in speaker verification systems and corresponds to our baseline. Differently from the other discriminative classifiers we discuss, GMM is a generative classifier (see, e.g., [5]). GMM is a special case of a Bayes classifier that adopts a mixture of G_y multivariate Gaussians as its likelihood model for modeling class y, namely

$$\hat{P}(\mathbf{x}|y) = \sum_{g=1}^{G_y} w_{yg} \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_{yg}, \boldsymbol{\Sigma}_{yg}). \tag{1}$$

with $\boldsymbol{\Sigma}_{yg}$ being a diagonal covariance matrix (that can be different for each Gaussian).

Kernel Learning

Very brief description of SVM and RVM. Review of works about SVM applied to speaker verification.

Here we briefly describe the kernel classifiers used in this work. We note that a Bayes classifier is called by some authors a "kernel" classifier (see, e.g., page 188 in [2]). However, by kernel classifier we mean the ones obtained through kernel learning, as defined, e.g., in [9].

Two of the kernel classifiers (RVM and IVM) that we consider are based on Bayesian learning, while the others (SVM and PSVM) are non-probabilistic.

Non-probabilistic kernel classifiers

SVM and other kernel methods can be related to regularized function estimation in a reproducing kernel Hilbert space (RKHS) [10]. One wants to find the function F that minimizes

$$\frac{1}{N}\sum_{n=1}^{N} L(F(\mathbf{x}_n), y_n) + \lambda ||F||^2_{\mathcal{H}_\mathcal{K}}, \tag{2}$$

where $\mathcal{H}_\mathcal{K}$ is the RKHS generated by the kernel $\mathcal{K}$, $F = h + b$, $h \in \mathcal{H}_\mathcal{K}$, $b \in \mathcal{R}$ and $L(F(\mathbf{x}_n), y_n)$ is a loss function.

Before trying to use a classifier with a non-linear kernel, it is wise to first check if a linear kernel

$$\mathcal{K}(\mathbf{x}, \mathbf{x}') = \mathbf{x} \cdot \mathbf{x}'$$

achieves the desired accuracy on the problem. A linear kernel classifier can be converted to a perceptron, which avoids storing the support vectors and saves computations during the test stage. Among non-linear kernels, the Gaussian radial-basis function kernel (when well-tuned) is competitive in terms of accuracy with any other kernel (see, e.g., [8]). The Gaussian kernel is given by

$$\mathcal{K}(\mathbf{x}, \mathbf{x}') = e^{-\gamma||\mathbf{x}-\mathbf{x}'||^2}.$$

The solution to the optimization problem described in Equation 2, as given by the *representer* theorem [4], is

$$F(\mathbf{x}) = \sum_{n=1}^{N} \omega_n \mathcal{K}(\mathbf{x}, \mathbf{x}_n) + b. \tag{3}$$

This expression indicates that SVM and related classifiers are *example-based* [9]: F is given in terms of the training examples $\mathbf{x}_n$. In other words, assuming a Gaussian kernel, the mean of a Gaussian is restricted to be a training example $\mathbf{x}_n$.

Some examples $\mathbf{x}_n$ may not be used in the final solution (e.g., the learning procedure may have assigned $\omega_n = 0$). We call *support vectors* the examples that are actually used in the final solution, even for classifiers other than

SVM (while some authors prefer to call relevance vectors the support vectors of RVM classifiers, and so on). For saving memory and computations in the test stage, it is convenient to learn a sparse F, with few support vectors. We can obtain sparse solutions only if the loss function L is zero over an interval (see, e.g., problem 4.6 in [9]). SVM achieves a sparse solution (at some degree) by choosing the loss

$$L(F(\mathbf{x}_n), y_n) = (1 - F(\mathbf{x}_n)y_n)_+,$$

where $(z)_+ = \max\{0, z\}$. The PSVM [1] classifier is obtained by choosing

$$L(F(\mathbf{x}_n), y_n) = (F(\mathbf{x}_n) - y_n)^2$$

and, consequently, ends up using all the training set as support vectors. Motivated by the discussion in [8], which takes into account some previous work related to PSVM, hereafter we refer to PSVM as *regularized least-square classifier* (RLSC).

Learning a RLSC classifier requires solving

$$(\mathbf{K} + \lambda\mathbf{I})\omega = \mathbf{y},$$

where $\mathbf{K}$ is the kernel matrix, $\mathbf{y} = \{y_n\}$ is the vector with labels and λ plays a role similar to the C constant in SVM. Training the classifier requires "only" a matrix inversion, but the matrix has size $N \times N$.

Assuming the SVM is implementing the optimal discriminant function between two classes that have a given Bayes error, the number of examples that are misclassified will scale linearly with the size of the training set, and so will the number of support vectors. Therefore, techniques to obtain sparser solutions are a current topic of research. Two of these techniques, namely RVM [11] and IVM [6], are discussed in the next subsection.

Probabilistic kernel classifiers

Both RVM and IVM are sparse Bayesian learning methods that can be related to the regularized risk functional of Eq. (2) (see, e.g., [9]). They adopt a model for the posterior given by $g(y_n F(\mathbf{x}_n))$, where F is given by Eq (3) and g is a sigmoid link function that converts scores into probabilities. Because the link function is used along the training stage, these methods potentially provide probabilities that are more "calibrated" [12] than the ones obtained, for example, by fitting a sigmoid after training a SVM [7].

We note that having a basic understanding of logistic regression (see, e.g., [2]) helps the study of probabilistic kernel methods such as IVM and RVM. For example, while linear regression obtains the "best fitting" equation by using the least-squares criterion, logistic regression uses the "logit" transformation to restrict the output of the classifier to the range [0, 1], and then uses a maximum likelihood method to train the classifier. This is the same approach used in some probabilistic kernel methods. They also use a

Table 1: Identificacao

	20seg	10seg	5seg
wired phone	98.73	98.74	98.55
cell phone	93.67	92.13	89.71

link function (e.g., the logit transformation for RVM and the probit transformation for IVM) to convert scores (real numbers) into an indication of probability, and a maximum likelihood method to train the classifier.

The Bayesian framework described in [11] allows, e.g., for solving multiclass problems and estimating the variances of each Gaussian. However, the computational cost is very high. Even when solving a binary problem with the standard RVM, the method is slow because the first iterations require inverting an $N \times N$ matrix.

IVM is an alternative to RVM with a faster training procedure. IVM adopts a different strategy and, instead of starting with all N and then eliminating examples as RVM, it uses a heuristic to select potential support vectors in a greedy way.

We note that many other probabilistic kernel methods have been recently proposed. For example, in [13], the import vector machine was derived by choosing the following loss function for the margin

$$L(F(\mathbf{x}_n), y_n) = \log(1 + e^{-F(\mathbf{x}_n) y_n}).$$

Only more research will indicate which probabilistic kernel methods prevail. For the moment, RVM and IVM seem to be a reasonable sample of these methods.

ISIP SYSTEM

(if you want to collapse section I and II into one longer introduction and split this section into two: a) ISIP and b) speaker recognition using ISIP, it's ok with me)

EXPERIMENTAL RESULTS

GMM is the baseline. SVM using conventional kernels (Gaussian, polynomial, linear) SVM using Fisher kernels, based on generative (GMM) models

Improving the baseline for the BaseIME corpus

Here we describe how we improved upon IME. Check what they published in the SBT paper because we don't want to mention the results in the classified thesis.

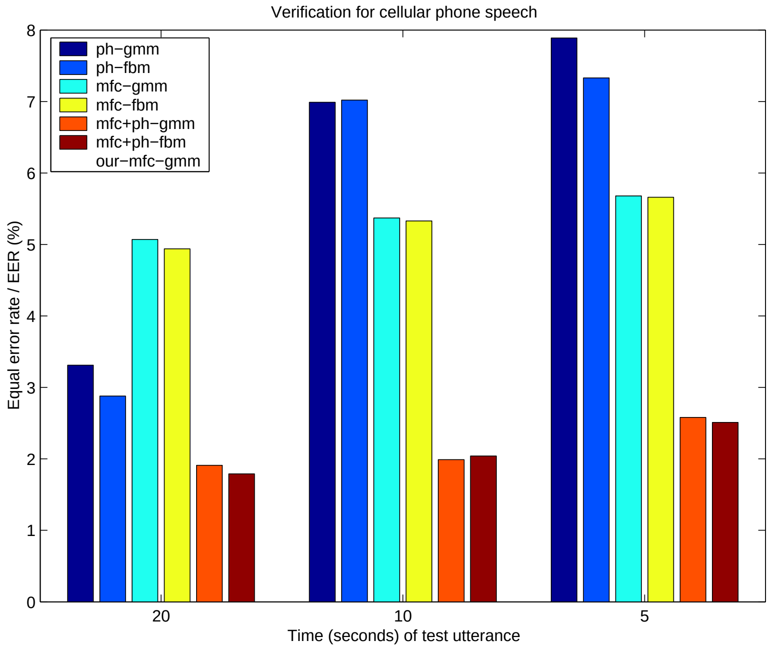

Figure 1: Verification for cellular phone speech

CONCLUSIONS

REFERENCES

[1] G. Fung and O. Mangasarian, "Proximal Support Vector Classifiers," in **KDD**, 2001, pp. 77–86.

[2] T. Hastie, R. Tibshirani and J. Friedman, **The elements of statistical learning**, Springer Verlag, 2001.

[3] T. Jaakkola and D. Haussler, "Exploiting generative models in discriminative classifiers," 1998.

[4] G. Kimeldorf and G. Wahba, "Some results on Tchebychean spline functions," **J. Math. Anal. Applic.**, vol. 33, pp. 82–95, 1971.

[5] A. Klautau, N. Jevtić and A. Orlitsky, "Discriminative Gaussian mixture models: A comparison with kernel classifiers," in **ICML, to appear**, 2003.

[6] N. Lawrence, M. Seeger and R. Herbrich, "Fast Sparse Gaussian Process Methods: The Informative Vector Machine," in **Neural Information Processing Systems 15**, 2002.

[7] J. Platt, "Probabilities for SV Machines," in A. Smola, P. Bartlett, B. Scholkopf and D. Schuurmans (eds.), **Advances in Large Margin Classifiers**, MIT Press, pp. 61–74, 1999.

[8] R. Rifkin, **Everything Old Is New Again: A Fresh Look at Historical Approaches in Machine Learning**, Ph.D. thesis, MIT, 2002.

[9] B. Scholkopf and A. Smola, **Learning with kernels**, MIT Press, 2002.

[10] A. Tikhonov and V. Arsenin, **Solutions of Ill-Posed Problems**, Winston, 1977.

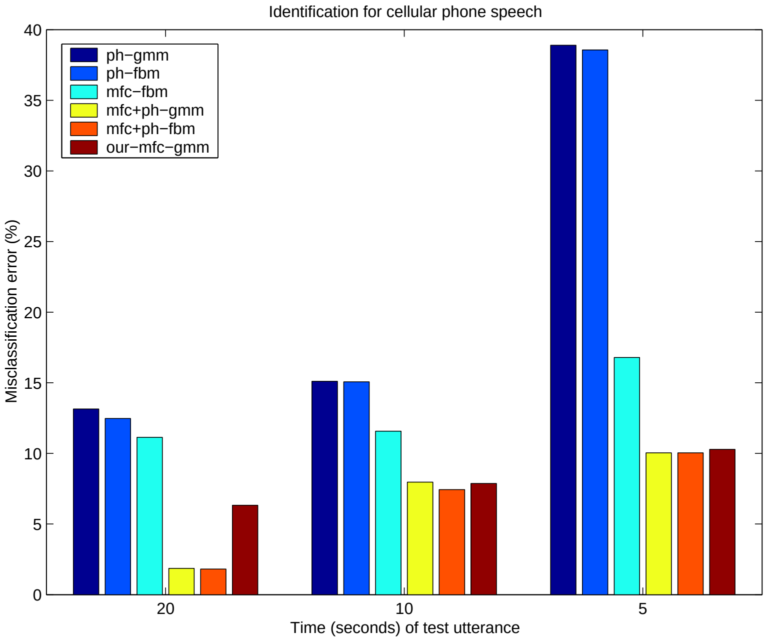

Figure 2: Identification for cellular phone speech

[11] M. Tipping, "Sparse Bayesian Learning and the Relevance Vector Machine," **Journal of Machine Learning Research**, vol. 1, pp. 211–244, 2001.

[12] B. Zadrozny and C. Elkan, "Transforming Classifier Scores into Accurate Multiclass Probability Estimates," in **KDD**, 2002.

[13] J. Zhu and T. Hastie, "Kernel Logistic Regression and the Import Vector Machine," in **NIPS**, 2001.

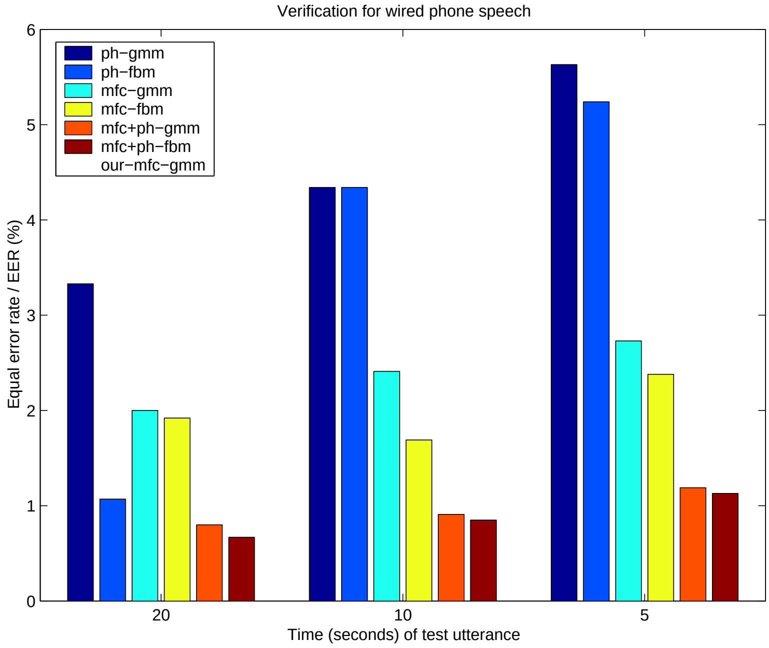

Figure 3: Verification for wired phone speech

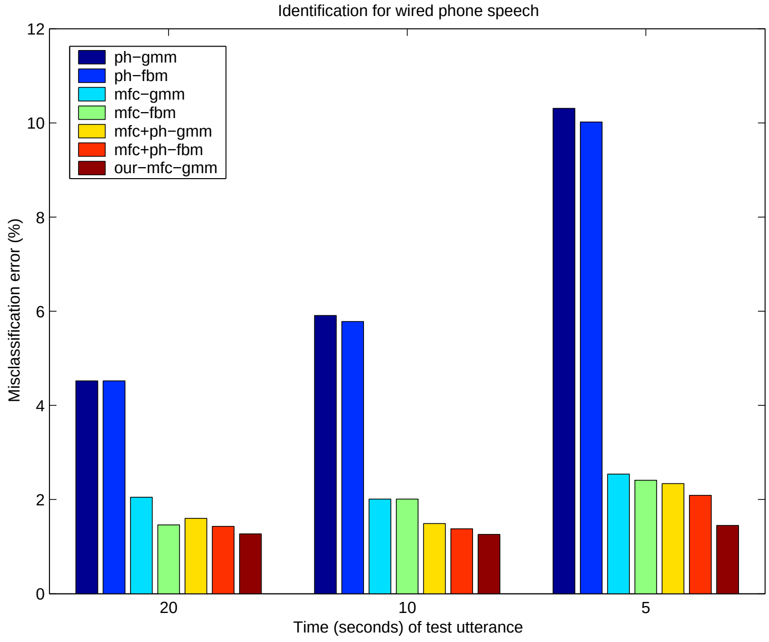

Figure 4: Identification for wired phone speech