NON-LINEAR SPEECH TRANSITION VISUALIZATION

Klaus Reinhard and Mahesan Niranjan

Cambridge University Engineering Department
Trumpington Street, Cambridge CB2 1PZ, U.K.
email: kr10000@eng.cam.ac.uk and niranjan@eng.cam.ac.uk

ABSTRACT

Modelling context effects and segmental transitions in speech recognition systems is very important. Explicitly modelling segmental transitions in a RNN framework would circumvent these problems. We present an interesting application of *Principal Curves*, an algorithm to extract a non-linear summary of p-dimensional data firstly published in 1989 by Hastie/Stuetzle. The algorithm can be used to visualize non-linear transient characteristics in speech. We will show that between-phone characteristics found within diphones can be used as discriminant information to distinguish ambiguous phones. The technique used is explained and illustrated on the examples */bah/*, */dah/* and */gah/*.

INTRODUCTION

Since speech is a complex time-sequential process, it is well established that particular phones vary acoustically when they occur in different phonetic context. In state-of-the-art systems based on short term spectral analysis this is achieved by enlarging the feature vector to include derivatives (e.g. delta cepstra, delta delta cepstra). It is well known that this increased dimensionality introduces parameter estimation problems when dealing with finite data. Recurrent neural networks (RNN) have been viewed as a natural choice for modelling the temporal structure of speech [7]. RNNs are widely used for context-dependent classification task because they seem to represent time implicitly within the "state" of the network. Nevertheless all RNN architectures in general treat the feedback as an information source which is fed into a black box system which enhances the overall performance of the classification task. The evaluation of their performance is mainly made on large tasks whereas there has been no research in depth to look into the decisions made for individual classes.

Critical questions to the claims of RNNs were made in the early work from Burrows/Niranjan [1], who showed for very simple recurrent neural networks the contribution performed by the feedback connection. Feedback mainly results in generating a switching delay at class boundaries and a smoothing of the output decision by moving the decision boundary. A particular problem was that RNN were operating in saturated regions. Especially for the effectiveness of feedback information using transient information, saturated regions have to be avoided which makes the network insensitive to the order of presentation of the input vector.

In a first step we are primarily aiming at explicitly modelling segmental transitions in the acoustic signal which can be found by employing the principal curve algorithm to important speech units. Incorporating transient models into a RNN framework will hence incorporate contextual feedback on an individual phone-pair base.

SUITABLE SPEECH UNITS

As a typical problem in speech recognition the discrimination of */b/*, */d/* and */g/* will be examined in the context of */ah/*. The classification of the stops */b/*, */d/* and */g/* is very ambiguous in the task of phone classification. Using the diphone based contextual information given by the transition to the succeeding phone as an additional information source would make the decision much more discriminant.

A suitable sub-word unit to extract between-phone contextual information is the diphone. A diphone is defined as half of one phone followed by half of the next phone (see Figure 1). Because the coarticulation influence does not usually extend much further then half way into the next phone characteristics of these speech units should represent useful contextual information which will improve the speech recognition process [5, 6].

DATA REPRESENTATION

The aim is to extract temporal trajectories in a lower dimension whereas the temporal characteristics and the statistics of the

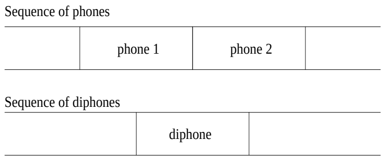


Figure 1: Schematic of a diphone

data are not necessarily correlated. An appropriate dimensionality reduction schemehas to be found which inherently carry a relationship between the temporal transition performed by the data and its statistical distribution. Hence additional constraints to the data has to be added to force algorithms to focus on temporal dependencies.

Time Constraint PCA (TC-PCA)
Employing standard PCA for dimensionality reduction focuses on maintaining maximum variance which is not necessarily correlated to characteristic temporal trajectories. Hence standard PCA dimensionality reduction schemes will not find characteristic trajectories in lower dimensions. Here two ideas were used to overcome the weakness of being unable to visualize the temporal transitions in diphones.

1. The importance of the transient regions between phones leads to an emphasising factor which enhances the resolution of the data in its important region. Latest results found that a quadratic emphasising function delivered the best results.

$$[\mathbf{x}_1 \ldots \mathbf{x}_m \ldots \mathbf{x}_n] * \mathbf{f}(t) = [\mathbf{x}_1^* \ldots \mathbf{x}_m^* \ldots \mathbf{x}_n^*]$$

$$\mathbf{f}(t) = 1 - \left(\frac{t - mid}{mid}\right)^2$$

where mid is the number of the middle frame to ensure a maximum of emphasis onto the middle region of the speech unit and t the actual frame within the speech unit. Every speech unit is constructed of n frames collected from the right and left frames around the phone boundaries, hence every speech unit consists of an odd number of $2n+1$ frames.

2. Adding temporal constraints to our data will force the algorithms to emphasise the time-dependencies. By adding another dimension to the data vector representing the frame order and hence its temporal ordering is forcing the algorithm to focus on temporal transitions. The strength of the added time-constraint can be chosen by a factor.

TC-PCA for dimensionality reduction forces the first principal component almost to be parallel to the time axis depending on the employed strength of the used constraints focusing on the temporal importance of our data representation. An important indicator for a sufficient time constraint can be found by using the ordering of the succeeding frames of each diphone. Assuming that the first principal component line is defined by $f(\lambda) = \bar{\mathbf{x}} + \lambda\mathbf{a}$, where $\mathbf{a}$ is the first linear principal component, then for a correct temporal ordering the line $f(\lambda)$ is represented by n tuples (λ_i, f_i) for each diphone represented by i frames $[\mathbf{x}_1 \ldots \mathbf{x}_n]$, joined up in increasing order of λ to form a straight line. Making sure that the tuples are sorted in increasing order of λ_i, one can maintain the temporal constraints.

PRINCIPAL CURVES

The algorithm for principal curves described by Hastie/Stuetzle [4] was firstly published in 1989 describing a method of extracting a smooth one-dimensional curve that pass through the "middle" of a p-dimensional data set providing a nonlinear summary of the data. The algorithm for constructing principal curves starts with the usual principal component line. This line is then smoothly bent according to the actual data representation, by locally averaging of p-dimensional points and iteratively minimising the orthogonal distances to the new curve.

The principal curve is defined by given $\mathbf{X}$ a random vector in $\mathbf{R}^p$. Let $\mathbf{f}$ denote a smooth unit-speed curve in $\mathbf{R}^p$ parameterised over $\Lambda \subseteq \mathbf{R}^1$, a closed interval, that does not intersect itself ($\lambda_1 \neq \lambda_2 \Longrightarrow \mathbf{f}(\lambda_1) \neq \mathbf{f}(\lambda_2)$) and has finite length inside any finite ball in $\mathbf{R}^p$. The projection index $\lambda_{\mathbf{f}} : \mathbf{R}^p \longrightarrow \mathbf{R}^1$ is defined as:

$$\lambda_{\mathbf{f}}(\mathbf{x}) = \sup_{\lambda}\{\lambda : ||\mathbf{x} - \mathbf{f}(\lambda)|| = \inf_{\mu} ||\mathbf{x} - \mathbf{f}(\mu)||\}$$

The projection index $\lambda_{\mathbf{f}}(\mathbf{x})$ of $\mathbf{x}$ is the value of λ for which $\mathbf{f}(\lambda)$ is closest to $\mathbf{x}$. If there are several values, the largest is picked.

The definition can be interpreted as starting with the first principal line to provide a initial ordering and collecting for any particular parameter value λ all observations that have $\mathbf{f}(\lambda)$ as their closest point on the curve. If $\mathbf{f}(\lambda)$ is the average of those observations, and if this holds for all λ, then $\mathbf{f}$ is called a principal curve. Because there is in general only one observation $\mathbf{x}_i$ related to a certain λ_i the observations projecting into a neighbourhood are locally averaged.

The principal curve algorithm iterates through *Projection-Expectation* steps until the relative change in the distance from the data points to its projections is below a certain threshold. As a initial projection step the ordering given by the first principal component is used.

$$\frac{|D^2(\mathbf{X}, \mathbf{f}^{i-1}) - D^2(\mathbf{X}, \mathbf{f}^i)|}{D^2(\mathbf{X}, \mathbf{f}^{i-1})} < threshold$$

with

$$D^2(\mathbf{X}, \mathbf{f}^i) = \sum_{i=1}^{n} ||\mathbf{x}^i - \mathbf{f}^i(\lambda_i)||^2$$

Expectation

As the expectation step a locally weighted running line smoother is employed to estimate a new $\mathbf{f}(\lambda)$ which will be used to calculate the new projecting points and hence the new distance from the data points to its projections. We used the algorithm for robust locally weighted regression suggested by Cleveland [2] calculating the parameters in a linear regression minimising the following expression, thus that $\hat{\beta}_j(\lambda_i)$ are the values of β_j:

$$\sum_{k=1}^{n} w_k(\lambda_i)(\mathbf{f}(\lambda_k) - \beta_0 - \beta_1 \lambda_k)^2$$

with

$$w_k(\lambda_i) = W(h_i^{-1}(\lambda_k - \lambda_i))$$

and

$$\begin{aligned} W(x) &= (1 - |x|^3)^3 \quad & for \quad & |x| < 1 \\ &= 0 & for \quad & |x| \geq 1 \end{aligned}$$

For each λ_i, weights, $w_k(\lambda_i)$, are defined for all λ_k, $k = 1, \ldots, n$ using the weight function W. This is done by centering W at λ_i and scaling it so that the point at which W first becomes zero is at the rth nearest neighbour of λ_i. For each i let h_i be the distance from λ_i to the rth nearest neighbour of λ_i. That is, h_i is the rth smallest number among $|\lambda_i - \lambda_j|$, for $j = 1, \ldots, n$. This procedure for computing the initial fitted values is referred to as locally weighted linear regression $\hat{\mathbf{f}}(\lambda_i)$.

$$\hat{\mathbf{f}}(\lambda_i) = \sum_{j=0}^{1} \hat{\beta}_j(\lambda_i)\lambda_i^j$$

To maintain the robustness a different set of weights, δ, is now defined for each $(\lambda_i, \mathbf{f}(\lambda_i))$ based on the size of the residual $\mathbf{f}(\lambda_i) - \hat{\mathbf{f}}(\lambda_i)$. Let $e_i = \mathbf{f}(\lambda_i) - \hat{\mathbf{f}}(\lambda_i)$ be the residuals from the current fitted values. Let s be the median of the $|e_i|$, then the robustness weights are defined by

$$\delta_k = B(e_k/6s)$$

with

$$\begin{aligned} B(x) &= (1 - x^2)^2 \quad & for \quad & |x| < 1 \\ &= 0 & for \quad & |x| \geq 1 \end{aligned}$$

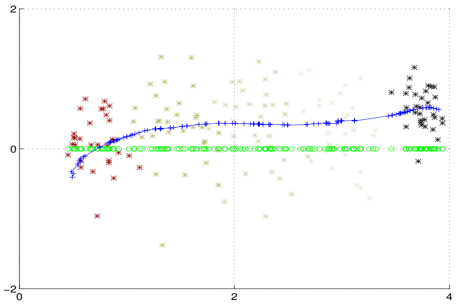
To compute new $\hat{\mathbf{f}}(\lambda_\mathbf{i})$ for each i the linear regression described above is performed with the weights $\delta_k w_k(\lambda_i)$ at $(\lambda_k, \mathbf{f}(\lambda_k))$. To refine the regression the robustness step to perform locally weighted linear regression might be performed several times.

Projection

Aim of the projection step is to reorder the data according to its distance relationship to the new generated curve. Starting from a new curve represented by n points $\mathbf{f}^{(\mathbf{j})}(\cdot)$ one wish to find for each $\mathbf{x_i}$ in the sample the value $\lambda_i = \lambda_{\mathbf{f}^{(\mathbf{j})}}(\mathbf{x_i})$. First we find the closest point on the line segment joining each pair $(\mathbf{f}(\lambda_k), \mathbf{f}(\lambda_{k+1}))$. The closest point to the curve is then either the projection onto a line segment or one of the $\mathbf{f}(\lambda_k)$. Using these values to represent the curve, we replace λ_i by the arc length from $\mathbf{f}_\mathbf{1}^{(\mathbf{j})}$ to $\mathbf{f}_\mathbf{i}^{(\mathbf{j})}$. Potential problems with this projection step can be found where the expected values of the observations project onto $\mathbf{f}(\lambda_{min})$ or $\mathbf{f}(\lambda_{max})$. To circumvent this problem the corresponding data points generate a new $\mathbf{f}(\lambda_{min})$ or $\mathbf{f}(\lambda_{max})$ by projecting the data point onto the first or last line segment extending the original principal curve.

EXTRACTION RESULTS

Initially the experiments to extract and visualize the underlying dynamics in our data representation were made using the published algorithm by Hastie which could be obtained from his ftp-site. The published algorithm is hard to evaluate because for higher dimensional data the algorithm uses the first two vectors to span a plane to perform a scatter-plot which fails to show the initial ordering using the first principal line and the bent principal curve according to the minimum squared distance. Because the algorithm was generating controversial results a MATLAB version of the algorithm was implemented enabling a scatter-plot on a random plane by providing two vectors to span a plane. The following pictures which are generated for the diphone problem rely entirely on our version which is producing the scatter-plot by projecting the data onto the plane spanned by the principal components with the largest eigenvalues. The used data was extracted from the phone labelled TIMIT database [3]. The MATLAB algorithm is then generating a principal curve within that plane showing additionally the first principal line as starting principal curve which results from the dimensionality reduction scheme using TC-PCA. Within the pictures the data belonging to different time frames is coloured differently.

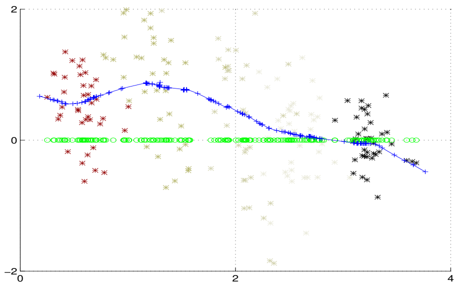


(a) Diphone /*gah*/ after first iteration

Figure 2: Principal Curves generated for the diphone /*gah*/ using 14-dim data generated from the MATLAB algorithm including the first principal line.

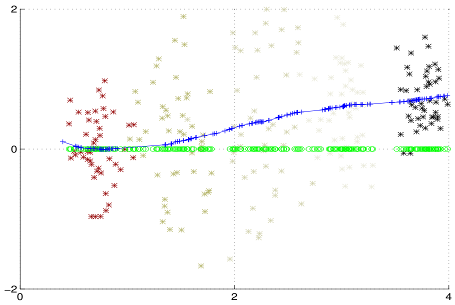
DISCUSSION

The figures 2, 3 and 4 show five temporal consecutive frames for different diphones appearing within different contexts. The middle frame corresponds to the phone boundary. Using time-constrained data one can clearly observe the different data frames where additionally the data points belonging to a certain frame are shaded in different



(a) Diphone /*bah*/ after first iteration

Figure 3: Principal Curves generated for the diphone /*bah*/ using 14-dim data generated from the MATLAB algorithm including the first principal line.



(a) Diphone /*dah*/ after first iteration

Figure 4: Principal Curves generated for the diphone /*dah*/ using 14-dim data generated from the MATLAB algorithm including the first principal line.

colours after the dimensionality reduction process using TC-PCA. Here the dimensionality reduction process is actually used to define an ordering of the data for the principal curve algorithm. This means that no information is thrown away because the principal curve algorithm works with the high dimensional data. Only the results are displayed in a lower dimension to visual evaluate the principal curve. The results of generating principal curves show the different transient characteristics for each phone-pair. The trajectories for the final frames for each diphone seems to be very similar. This follows the underlying structure of speech that phones represent acoustically steady state regions. Observations of different characteristic trajectories within the psecific diphone plane formed by TC-PCA for the different CV syllables can be found. They represent the non linear transitions from one phone to the next. Here further optimisation of the algorithm and the data representation will be employed to get most distinguishable trajectories.

CONCLUSION

Using principal curves to extract transient characteristics in speech units seems to be a promising application to visualize non-linear trajectories within diphones. Employing further optimisations to the algorithm as well as to the data representation we expect results which can be used to distinguish ambiguous phone-pairs. Here in particular we are interested in an extension of principal curves to 3D which might lead to a time invariant trajectory with higher discriminant information. This additional information to discriminate ambiguous phone-pairs is the base for further investigation how a RNN framework can be used to model explicitly discriminant transitions to support the recognition process without necessarily enlarging the input space.

ACKNOWLEDGEMENT

This work is partially supported by the Engineering and Physical Science Research Council studentship No. 95305262.

REFERENCES

[1] T.L. Burrows and M. Niranjan. The use of recurrent neural networks for classification. *IEEE Workshop on Neural Networks for Signal Processing IV*, pages 117–125, 1994.

[2] W.S. Cleveland. Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, 74(368):829–836, December 1979.

[3] 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), 1988.

[4] T. Hastie and W. Stuetzle. Principal curves. *Journal of the American Statistical Association*, 84(406):502–516, 1989.

[5] D. O'Shaughnessy. *Speech Communication:Human and Machine*. Addison Wesley, 1987.

[6] S. Saito and K. Nakata. *Fundamentals of Speech Signal Processing*. Academic Press, 1985.

[7] R.L. Watrous, B. Ladendorf, and G. Kuhn. Complete gradient optimization of a recurrent network applied to /b/,/d/,/g/ discrimination. In *Journal of the Acoustical Society of America*, volume 87(3), pages 1301–1309, 1990.