Odyssey 2012
The Speaker and Language Recognition Workshop
25-28 June 2012, Singapore

Generalized Viterbi-based models for time-series segmentation applied to speaker diarization

Itshak Lapidot and Jean-Francois Bonastre

University of Avignon, LIA, 339 Chemin des Meinajaries BP 91228, Avignon, 84911 France
itsikv@netvision.net.il jean-francois.bonastre@univ-avignon.fr

Abstract

Time-series clustering is a process which takes into account the input samples chronological sequence. So, in time-series clustering, the samples are not processed independently as a result for a given sample depends on the clustering result of the whole sequence. One of the popular clustering algorithms to handle such dependency is the well-known Hidden-Markov-Model (HMM) trained by the Viterbi statistics.
In this work we propose a generalization of the broadly used HMM, denoted Hidden-Distortion-Models (HDMs). Our proposal is based on distortion-based models and transition count, for which probabilistic calculations are no longer mandatory. We will introduce our approach by its mathematical bases. It will be shown that Viterbi based HMM can be seen as a special case of HDM. This proximity allows to us to apply similar approaches for state-model training when the new paradigm is used to learn the sequence dependencies.
Speaker diarization application will be presented to show the advantages of the HDM as a clustering algorithm.

1. Introduction

Time-series clustering is a process where the chronological sequence of the input must be taken into account. In time-series clustering, the samples are processed with respect to the dependencies between them. As a result, the clustering for a given sample may depend on the clustering result of the whole sequence. Time series clustering has many applications in different areas as speaker diarization [1]-[4], video segmentation [5], bio-medicine [6], and many others. This task corresponds to an unsupervised process where the samples have to be separated into *k* groups (clusters). Each group has to be homogeneous in some sense, e.g., one speaker per cluster, similar shapes, etc. The clustering process is driven by a criterion and different criteria lead to different clusters. It constitutes one of the main differences relatively to supervised classification processes, like speaker recognition, where training and working phases are clearly separated and the former process is driven by labeled data.
In time-series clustering, taking into account the time dependencies between the samples leads to different strategies depending on the time-context used to process a given sample. The probabilistic Hidden-Markov-Models (HMMs) approach and its variants [1] [7] [8] are one of the most successful approaches in this case.
HMM based clustering has many advantages, but at the same time suffers from several limitations:

1. The model training process is based on Viterbi statistics. Both transition matrix and state models are optimized using Maximum Likelihood criterion. The estimation of the transition parameters of the HMM model is based only on the counts when the state models learning (usually, Gaussian Mixture Models (GMM)) relies on EM algorithm using input samples. So, the whole training process is disjoint and there can be an unbalance between the emission likelihoods and the transition probabilities. It might happen that most of the global likelihood depends on the transition probability, and is almost independent from the input samples. It might be the case if the state changes are rare, so the self loop transition probability is very high (close to one), while other transition probabilities are very small. In this case, a regularization parameter can help to improve the performance. However, in the probabilistic framework (HMM), there is no regularization option to adjust the transition probabilities. To emphasize this point, the aim of the HMM training is to increase the global likelihood, involving both transitions and emissions parts, and not to decrease the clustering error.
2. HMM approach is based on the probabilistic paradigm and the state models (a state model represents one of the classes) have to be statistical models (GMM for example). For some specific situations or tasks like Damerau–Levenshtein distance calculation in strings comparison (DNA protein sequences), it is a limitation as it is difficult to represent such a constraint with probabilistic models.

To overcome these two limitations, we propose in this work an extension of the HMM, which embeds the advantages of HMM-based approach but allows also to use distortion-based approaches. Distortion-based approaches will allow both to learn the time dependencies and to represent the different states/classes by models other than probabilistic ones. We name our approach "Hidden-Distortion-Model" (HDM) as it corresponds to an HMM-like approach but using distortion paradigm. To do so, we limit ourselves to a family of additive distortions, $Distortion(x_1,\ldots,x_N)=\sum_{n=1}^{N} distortion(x_n)$, i.e. the distortion of a N vectors sequence is the sum of N individual distortions applied each on one vector. Unlike distance which is defined as a metric, distortion does not have all the metric properties (non negativity for example in the negation of log-likelihood). On the other hand, like distance, we would like to relate close events with a small distortion. The negation of the log-likelihood is an example of such additive distortion, so Viterbi-based HMM can be seen as a particular case of HDM. Instead of the emission probabilities, emission distortions are calculated; similarly, transition cost matrix and initial cost vector are used as a replacement of transition probability matrix and initial probability vector. An estimation of all the parameters is done in the distortion and transition counts space, without requiring any probability, or likelihood estimation. In this new framework, a regularization of transition costs becomes a natural part of the model. The

regularization parameters have to be determined based on some development data. We compare the HDM approach on the base of the system presented in [1], which is a variant of HMM with self-organizing map (SOM) as a state probability model. First, we use the original system as a baseline, and then replace the HMM by the HDM. It will be shown that better results can be achieved using HDM, compared to the HMM-based baseline system.

As an application of the described framework, we present results obtained in the task of speaker diarization. Speaker diarization has a growing interest in the recent years [1]-[4]. Given a conversation between several unknown participants, speaker diarization comes to answer the question "Who spoke when?" As both the speakers and the speech segment boundaries are unknown, the problem corresponds to a time-series clustering. Sometimes the number of participants is also unknown and has to be estimated. Many different algorithms were proposed to solve this problem and many of them are based on HMMs with Viterbi segmentation [1], [3] and [4]. Such an application is well suited to evaluate the HDM approach we present in this paper. We evaluate it on telephone conversations, where the number of speakers is known and equal to two.

The manuscript is arranged as follows: the classical HMM-based clustering approach is presented in section 2; section 3 describes the new HDM approach, highlighting the theoretical constraints and section 4 provides theoretical solutions to these constraints. Section 5 illustrates how HDM can be applied to fix duration constraints. The comparison between HDM and HMM is discussed in section 6. In section 7, we present several possible constraints on the objective function to be minimized when the experimental results on speaker diarization problem are shown at section 8. Finally, we conclude on the interest of HDM in section 9 together with future extensions of this work.

2. HMM based clustering limitations

In HMM, the log-likelihood of any clustering path is a combination of two sums. One sum relies only on the log-likelihoods of the models given the input data, and the second sum relies only on the logarithm of the transition matrix. During the training phase, at each iteration, the Viterbi algorithm follows the Maximum Likelihood criterion by optimizing separately the emission probabilities and the transition probabilities which are linked to the two terms of the log-likelihood sum. The emission probability models are optimized using only the related samples when the transition matrix optimization is based only on the transition counts. In figure 1 we show an example of two Gaussian distributions with the same variance, $\sigma^2 = 1$ and the means $\mu_1 = -\mu_2 = 1$. In the upper plot (a), both distribution are drown, while in the plot below (b) the log-likelihood ration is given in the solid line. It can be seen that for each data sample the contribution of the emission probabilities to the global log-likelihood of a path is usually less than six (in terms of absolute value of the emission probabilities log-likelihood ratio). If the transition frequency from one model to another is relatively low than the contribution of the transitions to the global path log-likelihood will be comparative to the contribution of the state models. For example, let us assume that the state change rate is each 60 samples on average. In this situation, the self transition probabilities are $a_{11} = a_{22} = \frac{59}{60}$ and the probabilities to change from one state to another are $a_{12} = a_{21} = \frac{1}{60}$. The log ratio is $\ln\left(\frac{a_{11}}{a_{12}}\right) = \ln\left(\frac{a_{22}}{a_{21}}\right) = \ln\left(\frac{59/60}{1/60}\right) \approx 4.1$ (dash lines). It means that if the likelihood of a sample is much higher for a given state model than for the others, a transition (in direction of this state) may be observed. On the other hand, if the transitions are much rarer, like in conversational interview, where the transitions might happen each 8 seconds on average, which is 800 samples (in speech recognition the features are usually extracted each 10msec) then the $\ln\left(\frac{a_{11}}{a_{12}}\right) = \ln\left(\frac{a_{22}}{a_{21}}\right) = \ln\left(\frac{799/800}{1/800}\right) \approx 6.7$ (dot line). In this case, the values of the emission probabilities become irrelevant and the decisions rely only on the transition probabilities. So the algorithm will always tend to stay in the initial state (only out-layers can cause to the system to switch). This situation of staying in the initial state leads to the maximum likelihood of course, but to very poor clustering performances. The opposite situation could also occur if the transitions ratio is largely smaller than the state-emission probabilities (for a state swap every 3 input samples in average the transition log ratio is $\ln\left(\frac{a_{11}}{a_{12}}\right) = \ln\left(\frac{a_{22}}{a_{21}}\right) = \ln\left(\frac{2/3}{1/3}\right) \approx 0.7$, to be compared with a ratio which can be up to 6.0 for the emission probabilities). Although, our goal is to optimize the clustering quality by minimizing the clustering error, the HMM maximization objective function is the log-likelihood function and could be suboptimal in some situations.

In order to solve this problem, another framework has to be developed which can in the same time take into account the transitions but not neglect the emission probabilities and vice versa. In order to estimate the transition costs, it could be also useful to allow the use of other frameworks than the probabilistic one, like distortion-based models. This leads us to propose a more global family which is denoted Hidden-Distortion-Model (HDM). We will show next that HMM is a private case of HDM.

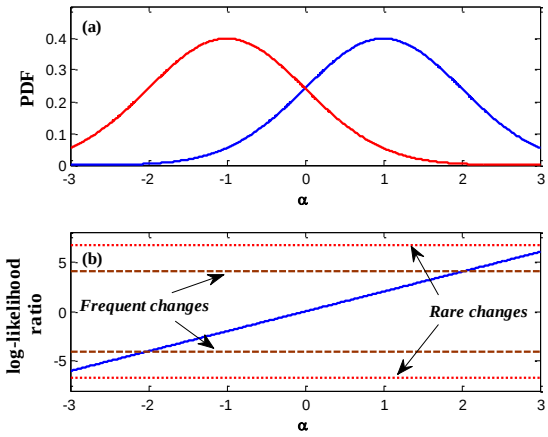

Figure 1: Example of two states HMM. (a) The pdfs of the states. (b) The log-likelihood ratio of the state models (solid line); frequent changes transition-cost ratio (dash lines); rare changes transition-cost ratio (dot line).

3. Problem definition

Assuming we have a system with *K* hidden states. Each state is defined by a distortion model DM_k. Be $C_{qk} = \text{cost}(s_n = q \mid s_{n-1} = k)$ the transition cost of being at discrete time *n* at state *q*, given being at time *n-1* at state *k*. $C = \left[C_{qk}\right]_{|q=1,\ldots,K,\, k=1,\ldots,K}$, is a time constant cost transition matrix. $d_k(x_n)$ is a distortion of the data vector $x_n \in X = \{x_1,\ldots,x_N\}$, when *X* is the sequence of data vectors, given a model DM_k. The distortion have to be additive, meaning, $D(X \mid DM) = \sum_{x_n \in X} d(x_n)$. GMM for example is such a model with $d(x_n) = -\log(l(x_n))$, where $l(x_n)$ is the likelihood of the model given the observation vector x_n.

In addition there is a vector of initial costs, to be at state *k* at time zero, $\Pi = [\pi_1,\ldots,\pi_K]^T$. Our model can be defined as a triple $\mathcal{M} = \{\{DM_k\}, C, \Pi\}$.

The two problems we have for HDM are:

Given the distortion models $\{DM_k\}_{|k=1,\ldots,K}$, the cost transition matrix C and the vector of initial costs Π, find the path which minimizes the cost for a sequence of data samples $X = \{x_1,\ldots,x_N\}$:

$$C_N(X \mid \mathcal{M}) = \min_{\{s_n\}_{|n=1,\ldots N}} \left\{ \pi_{s_1} + d_{s_1}(x_1) + \sum_{n=2}^{N} \left(d_{s_n}(x_n) + C_{s_n s_{n-1}} \right) \right\} \quad (1)$$

This problem can be solved using the well known Viterbi algorithm.

Parameter estimation problem in Viterbi sense: given the data samples *X*, the sequence of states $S = \{s_n\}_{|n=1,\ldots N}$, and the model parameters $\mathcal{M}$, to find a new model $\hat{\mathcal{M}}$ which will minimize the total cost. First let us find the total cost:

$$\begin{aligned} C_N(X,S \mid \mathcal{M}) &= \pi_{s_1} + d_{s_1}(x_1) + \sum_{n=2}^{N} \left(d_{s_n}(x_n) + C_{s_n s_{n-1}} \right) = \\ &= \pi_{s_1} + \sum_{n=1}^{N} d_{s_n}(x_n) + \sum_{n=2}^{N} C_{s_n s_{n-1}} = \\ &= C_N(S \mid \mathcal{M}) + D_N(X \mid S, \mathcal{M}) \end{aligned} \quad (2)$$

When $C_N(S \mid \mathcal{M})$ is the total sum of costs and $D_N(X \mid S, \mathcal{M})$ is the total distortion, given the model and the state sequence. As it can be seen, the distortions part and costs part are disjoint and can be minimized separately. When only one sequence is available, it is not possible to train properly the initial costs as one cost will have a reasonable value and the others, in many cases, will be set to infinity, as it happen in HMM with Viterbi training. In the HMM case, one state will have probability one and all the others will be set to zero. The HMM costs are $C_{qk} = -\log(w_{qk})$, $\pi_k = -\log(\omega_k)$, when w_{qk} is the transition probability from state *k* to state *q*, and ω_k is the initial probability of state *k*.

The initial scores vector can be trained if several sequences from the same environment have to be clustered together. Assuming several records of conversations of the same group of participants are available, it becomes possible to cluster all the conversations together, enabling to train also the initial cost vector.

4. Model parameters estimation

In this section we present the estimation procedure of the HDM parameters. The estimation of the initial cost vector, transition cost matrix and the state model estimation are presented.

4.1. Counts Model

Like in HMM, let us assume first that we do not have hidden variables and instead of observation vector sequences, we have a set of state sequences $\mathbf{S} = \{S_q\}_{q=1}^{Q}$, $S_q = \{s_{q1},\ldots,s_{qN_q}\}$.

We wish to estimate the cost transition matrix C and the initial cost vector Π.

Just like in the 1st order Markov model log-likelihood calculation, the total cost will be the sum of all the cost along the given path. As several sequences are given, the sum will be also over the all sequences:

$$\begin{aligned} C_N(\mathbf{S} \mid \{C,\Pi\}) &= \sum_{q=1}^{Q} \left(\pi_{s_{q1}} + \sum_{n=2}^{N_q} C_{s_{qn} s_{qn-1}} \right) = \\ &= \sum_{k=1}^{K} N_k \pi_k + \sum_{k=1}^{K} \sum_{q=1}^{K} N_{qk} C_{qk} \end{aligned} \quad (3)$$

$$\sum_{k=1}^{K} N_k = Q, \quad \sum_{k=1}^{K} \sum_{p=1}^{K} N_{pk} = \sum_{q=1}^{Q} N_q$$

When N_{qk} is the number of transitions from state *k* to state *q* over all the sequences, and N_k is the number of times to be at state *k* at time zero (beginning of the conversation, for example).

Minimizing the expression $\sum_{k=1}^{K} N_k \pi_k + \sum_{k=1}^{K} \sum_{q=1}^{K} N_{qk} C_{qk}$ of eq. (3) over all the costs is straightforward, by setting all the values to zero. This trivial solution almost does not carry any information. The only constrain for this solution is that all the costs should be non-negative values, which is not always required. An example of such system is a clustering process based on a single codebook. Usually, in such case, each code-word has its Voronoi cell, and the vectors which are in the cell define a cluster. The problem is to find the partition able to minimize the overall distortion. If we do not want the trivial solution, the minimization should be done according to some pre-defined constraints. A first simple constraint is defined by $\forall k \bullet \sum_{q=1}^{K} \frac{1}{C_{qk}} = 1$ and $\sum_{k=1}^{K} \frac{1}{\pi_k} = 1$, this ensures that the sum over the inverse costs will equal to one for each state or initial cost. This constraint observes a somehow "probabilistic" feeling. This implies that more frequent transitions will have lower transition cost than rare transition, and the same for the initial costs. The objective function to be minimized, using the Lagrange multipliers, is:

$$\begin{aligned} J(C) = &\sum_{k=1}^{K} \sum_{q=1}^{K} N_{qk} C_{qk} + \sum_{k=1}^{K} \lambda_k \left[\sum_{q=1}^{K} \frac{1}{C_{qk}} - 1 \right] + \\ &+ \sum_{k=1}^{K} N_k \pi_k + \lambda_\pi \left[\sum_{k=1}^{K} \frac{1}{\pi_k} - 1 \right] \end{aligned} \quad (4)$$

Taking the partial derivation with respect to C_{qk} and compare it to zero gives:

$$\frac{\partial J(C)}{\partial C_{qk}} = N_{qk} - \lambda_k \frac{1}{C_{qk}^2} = 0 \qquad (5)$$

For each *q*, we have

$$\lambda_k = N_{qk} C_{qk}^2 \Rightarrow \forall p,q \bullet N_{pk} C_{pk}^2 = N_{qk} C_{qk}^2 \qquad (6)$$

We can now construct *K-1* linearly independent equations, without a lost in terms of generality, $N_{1k}^{0.5} C_{1k} = N_{qk}^{0.5} C_{qk}$ for $q = 2,\ldots,K$ (we do not want to take the solutions which give negative cost, but theoretically it can be done), and one non-linear equation $\sum_{q=1}^{K} \frac{1}{C_{qk}} = 1$. It is easy to see that the following expression solves all the equations, and all the costs are positive.

$$C_{qk} = \frac{\sum_{p=1}^{K} N_{pk}^{0.5}}{N_{qk}^{0.5}} \qquad (7)$$

The same should be done for the initial costs:

$$\frac{\partial J(C)}{\partial \pi_k} = N_k - \lambda_\pi \frac{1}{\pi_k^2} = 0 \Rightarrow \pi_k = \frac{\sum_{p=1}^{K} N_p^{0.5}}{N_k^{0.5}} \qquad (8)$$

The costs are all non negatives, and even all not less than 1.

4.2. Hidden Distortion Model

In sub-section 4.1 we estimated the $C_N(S \mid \mathcal{M})$ part of eq. (2). In order to estimate the distortion models, we have to minimize the following expression:

$$D_N(X \mid S,\mathcal{M}) = \sum_{n=1}^{N} d_{s_n}(x_n) = \sum_{k=1}^{K} \sum_{\{n \mid s_n = k\}} d_{s_n}(x_n)$$
$$\sum_{k=1}^{K} \#\{n \mid s_n = k\} = \sum_{q=1}^{Q} N_q \qquad (9)$$

From the right side of eq. (9), we see that each distortion model can be minimized independently from all the others, applying the minimization algorithm according to the pre-defined distortion measure.

4.3. The iterative training

Given an HDM of *K* states with distortion models $\{MD_k\}_{|k=1,\ldots,K}$ and the data $\mathbf{X} = \{X_q\}_{q=1}^{Q}$, $X_q = \{x_{q1},\ldots,x_{qN_q}\}$, the algorithm is:

Initialization:

1. For each state, initiate the models $\{MD_k^{(0)}\}_{|k=1,\ldots,K}$. Different ways can be applied depending on the targeted task and the type of model.
2. Initialize the cost matrix $C^{(0)}$ and the initial vector $\Pi^{(0)}$. It can be done randomly according to some assumptions or by finding the path according to the partition of the data, relying only on the scores at step 1.

Iterative part:

3. Segment the data using the model, and get the new partition and the minimum cost path.
4. Train the distortion models with the new partition, according to sub-section 3.1, and get $\{MD_k^{(i+1)}\}_{|k=1,\ldots,K}$.
5. Train the new transition cost matrix $C^{(i+1)}$ and initial cost vector $\Pi^{(i+1)}$ according to eqs. (7) and (8).
6. Set $\mathcal{M}^{(i+1)} = \left\{\{MD_k^{(i+1)}\}_{|k=1,\ldots,K}, C^{(i+1)}, \Pi^{(i+1)}\right\}$ $\rightarrow \mathcal{M}^{(i)} = \left\{\{MD_k^{(i)}\}_{|k=1,\ldots,K}, C^{(i)}, \Pi^{(i)}\right\}$, and iterate steps 3 to 6 until to meet the termination conditions.

If only one sequence is given as input of the algorithm, the training of the initial vector is impossible and the cost should be set accordingly to some prior knowledge (equal costs could be also used if there is no priority of one model over the others).

5. Duration constraint parameters estimation

In speaker diarization, it is reasonable to force the direction from one state to another for several consecutive frames (left-to-right model with one possible transition). Furthermore, usually all the states share the same state model. The time constraints are linked to some physical considerations, such as, speaker cannot speak less than 200ms. This leads to force the system to stay in a "hyper state" for 20 successive input data (frame rate is 100 frames per second). According to eq. (7), the corresponding transition costs will be equal to 1. It differs from the HMM which implies a probability set to one, i.e., zero in terms of transition log-probability. All other transition costs are set according to eq. (7).

At the last state of each hyper-state, only transition to the first state of each "hyper-state" is allowed. It is giving a fixed duration clustering system. The model, the transition matrix and initial transitions vectors estimation are identical than the ones described in section 4. An example of two-state fix duration system is given in figure 2.

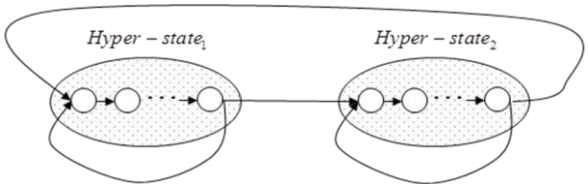

Figure 2: Two-state fix duration HDM system.

In general it is easier to describe the transitions cost matrix as a block matrix, where each block is a transition matrix $\mathbf{C}_{qk}$ between hyper-states *k* and *q*:

$$C = \begin{pmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} & \cdots & \mathbf{C}_{1K} \\ \mathbf{C}_{21} & \mathbf{C}_{22} & \cdots & \mathbf{C}_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_{K1} & \mathbf{C}_{K2} & \cdots & \mathbf{C}_{KK} \end{pmatrix} \qquad (10)$$

If each state has fix duration of length τ, then the diagonal blocks are a intra hyper-state transition costs matrix, $\mathbf{C}_{kk}$, defined in eq. (11). The elements below the main diagonal are all equal to $C_{MinCost}$, as this is the only allowed path. At the last state of the hyper-state, it is allowed to transit to the first state of any hyper-state, including self loop. The upper right

element is the self loop transition cost from the last state of the k^{th} hyper-state to its first state. All other transitions are forbidden and fixed to a maximal transition cost, $C_{MaxCost}$.

The inter hyper-state transition costs matrix is given in (12). As any transition is forbidden except from the last state of the k^{th} hyper-state to first state of the q^{th} hyper-state, all the costs are equal to $C_{MaxCost}$, excluding the upper right one, which equals to C_{kq}.

$$\mathbf{C}_{kk} = \begin{pmatrix} C_{MaxCost} & \cdots & C_{MaxCost} & C_{kk} \\ C_{MinCost} & \cdots & C_{MaxCost} & C_{MaxCost} \\ \vdots & \ddots & \vdots & \vdots \\ C_{MaxCost} & \cdots & C_{MinCost} & C_{MaxCost} \end{pmatrix} \in \mathbb{R}^{\tau\times\tau} \qquad (11)$$

$$\mathbf{C}_{qk} = \begin{pmatrix} C_{MaxCost} & C_{MaxCost} & \cdots & C_{qk} \\ C_{MaxCost} & C_{MaxCost} & \cdots & C_{MaxCost} \\ \vdots & \vdots & \ddots & \vdots \\ C_{MaxCost} & C_{MaxCost} & \cdots & C_{MaxCost} \end{pmatrix} \in \mathbb{R}^{\tau\times\tau} \qquad (12)$$

If we apply it to HMM then $C_{MinCost} = -\ln(1) = 0$, $C_{MaxCost} = -\ln(0) = \infty$, and the cost the $C_{qk} = -\ln(w_{qk}) = -\ln(p(s_n = q \mid s_{n-1} = k))$

6. HDM verses HMM and DTW

In many senses the HMM and the HDM are similar, but HDM is much more flexible than the HMM. The main advantages of the HDM are:

1. HDM do not restrict all transition probabilities from each state to sum to 1. Instead, different constraints can be applied according to some knowledge. In this manuscript, the case of the sum of the inverse costs constraint was presented used in eq. (4).
2. In HMM the "cost" of the probability 1 transitions is the negation of the log-probability and always equal to zero. In the presented case, the "all counts" transitions are equal to one and with other constraint the cost can be any other value.
3. In both approaches, the cost of zero count transitions is infinite. In HMM, it corresponds to the negation of the logarithm of zero and, in the presented work, we have zero in the denominator. In practical systems, if we want to preserve the ability to train this zero count transitions, the cost should be set to some high value, but not infinite value.

To conclude on this comparison, it can be said that HMM in the case of Viterbi training is a private case of HDM, with distortions set to the negation of the log-likelihood of the emission probabilities; the costs are the negation of the logarithms of the transition/initial probabilities. The constraints which are used to calculate the costs in HMM are:

$$\begin{aligned} \sum_{q=1}^{K} w_{qk} &= \sum_{q=1}^{K} e^{-C_{qk}} = 1 \\ \sum_{q=1}^{K} \omega_k &= \sum_{q=1}^{K} e^{-\pi_k} = 1 \end{aligned} \qquad (13)$$

Where w_{qk} is the transition probability to state *q* from state *k*, and ω_k is the initial probability to be in state *k*.

Another comparison can be done with dynamic time warping (DTW) and Gaussian dynamic warping (GDW), presented by Bonastre at el [9]. Both methods are based on finding the best matching path on a grid, by comparing reference templates verses the test template. Both approaches are based on additive distortion constraints as presented in this study. The main advantages of our approach are: 1) it does not required a predefined reference template; 2) the transition costs are trained and do not have to be defined by some rules of thumb, including local and global restrictions of the moves on the grid.

7. Two examples of constraints

In sections 4-7, we have shown how to define the distortions and the costs for the HDM. This can achieve different results according to different distortion models and different transition constraints. It is still does not solve the problem of cost regularization. If the costs are high comparably to the distortions, the problem remains the same as shown in section 2. In this section, we will present several ways to regularize the costs by applying regularization parameters into the constraints.

1. Scaled log-likelihood:

$$\sum_{q=1}^{K} w_{qk} = \sum_{q=1}^{K} e^{-\alpha_1 C_{qk}} = 1 \qquad (14)$$

In this case we used similar constraints than in HMM but, instead of the cost in the exponent, we use a scaled cost. It is easy to show that the costs become:

$$C_{qk} = -\frac{\ln(w_{qk})}{\alpha_1} = \frac{1}{\alpha_1}\ln\left(\frac{\sum_{p=1}^{K} N_{pk}}{N_{qk}}\right) \qquad (15)$$

2. Powered inverse sum:

$$\sum_{q=1}^{K} \frac{1}{C_{qk}^{\alpha_2}} = 1 \qquad (16)$$

It gives: $\frac{\partial J(C)}{\partial C_{qk}} = N_{qk} - \lambda_k C_{qk}^{-\alpha_2 - 1} = 0 \Rightarrow$

$N_{pk} C_{pk}^{\alpha_2+1} = N_{qk} C_{qk}^{\alpha_2+1} \Rightarrow C_{pk} = \left(\frac{N_{qk}}{N_{pk}}\right)^{\frac{1}{\alpha_2+1}} C_{qk}$. Substitute this result into the constraint equation gives:

$$C_{qk} = \left(\frac{\sum_{p=1}^{K} N_{pk}^{\frac{\alpha_2}{\alpha_2+1}}}{N_{qk}^{\frac{\alpha_2}{\alpha_2+1}}}\right)^{\alpha_2^{-1}} \qquad (17)$$

This time, the hyper-parameter α is responsible about the starching or compressing the ration between the costs, due to the presence of α_2 in the power of all the expression. So, it becomes possible to emphasize or deemphasize the frequent transitions compared to the rare transitions.

The α parameter is a scaling hyper-parameter which should be estimated on some development data.

In the 2nd case defined in (17) the confidence on the counts is regularized. It means that small values of α increase both the costs and the ratio between the costs of rare and frequent

transitions. If the value of α increases, all the costs will tend to one, which means that the counts are unreliable.

In figure 3, we return to the example presented in figure 1 where we have the same Gaussian distributions with the same variance, $\sigma^2 = 1$ and the means $\mu_1 = -\mu_2 = 1$ (shown in the upper plot (a)). In the plot below (b) as in figure 1, the log-likelihood ratio is given in the solid line. The rare transition costs case is drown using dot line (approximately 6.7) and the probability for state changing is very low. Appling (15) with a scaling factor $\alpha_1 = 0.5$ allows setting the transition costs to a more reasonable value (dash line). This value should be estimated on a development set. This example is only illustrative as for two states case with symmetrical distributions and same state-change rate, the transition costs depend only on one parameter which can be estimated on the development set, without using any of the above mentioned constraints. This trivial solution becomes unreachable when the number of states increases nor if the distributions differ.

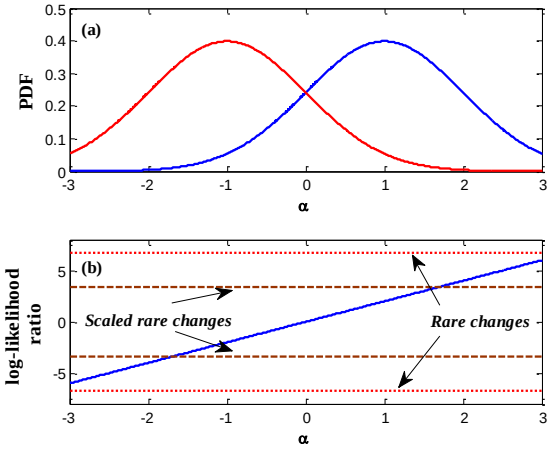

Figure 3: Example of two states HMM. (a) The pdfs of the states. (b) The log-likelihood ratio of the models (solid line); rare changes transition costs (dot line); rare changes transition costs scaled by a factor of $\alpha = 0.5$ (dash lines).

8. Experiments and results

8.1. Speaker Diarization

We apply our HDM approach on a two-speaker telephone speaker diarization task. Non-speech data and overlapped speech can be present in the conversation and the corresponding segments should be detected as well. The system used for this experimental evaluation of HDM reuses mainly the HMM-based system presented in [1]. The system block diagram is presented in figure 4. It is mainly composed by a set of preprocessing steps (feature extraction, non-speech detection, overlapped speech detection…) followed by the diarization system itself.

First, classical Mel-Frequency Cepstral Coefficients (MFCC) are extracted (20ms signal window with 50% of overlap, 12 MFCC coefficients). The speech activity detection is performed by a simple energy threshold. The overlapped speech detection performs in the time domain and is described in [1].

The speaker diarization system has 3 hyper-states (non-speech, speaker A, speaker B). As explained in section 5, a fixed duration constraint of 20 tied states (200ms) is used during the first 5 iterations and, in order to increase the resolution, only 10 tied states are used for the last iteration (giving a total of 6 iterations). Each cluster model is a Self-Organizing Map (SOM) [11], with size of 6×10, used as likelihood estimator [12] (assuming that each code-word is a mean of a Gaussian with an identity covariance matrix). In all HDM experiments, the model distortion measure is the square Euclidian distance. The non-speech model is initialized using the non-speech segments provided by the speech activity detector and the two other models are initialized thanks to a weighted segmental K-means [10] (applied only on the speech segments). As each conversation is diarized separately, no initial costs are used.

In all the presented experiments, "baseline" refers to the HMM-based system (corresponding to [10]), by setting the corresponding HDM parameters to follow HMM transition probability model.

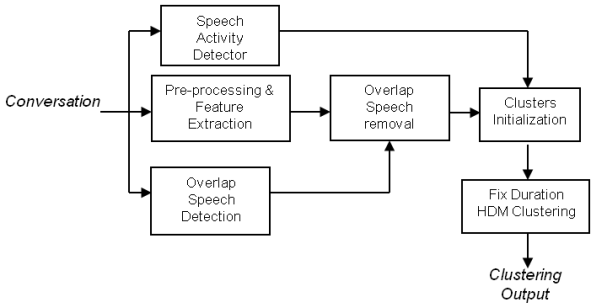

Figure 4: Speaker diarization system.

8.2. Database

Two databases are used for the experiments: LDC America CallHome [14] and NIST 2005 [15]. 108 conversations CallHome conversations are used for LDC of about 30 minutes duration each, but with only about 10 minutes with human transcription. Only this transcribed part is used here. 2048 conversations are selected for NIST, with duration of about 5 minutes for each conversation. The data are sampled at 8kHz in a 2 channel μ-law format and the two channels are summed in order to have one channel conversations.

8.3. Diarization Error Rate (DER)

The performance is evaluated thanks to the frame-based Diarization Error Rate, as defined by NIST in [16]. The DER calculation is performed excluding a 0.5 seconds time-window around the changing points (i.e., the errors inside 0.25 second on each side of the changing points are not taken into account).

8.4. Experiments with LDC America CallHome

Table 1 presents the DER for the baseline (HMM). Results using the same system but without transition costs (the transitions probabilities are all equal) is also presented. It can

be seen that the HMM transitions give about 30% relative DER improvement.

Table 1: Results of the baseline system and without transition cost system.

	Baseline	**No Costs system**
DER [%]	17.18	24.37

1st Experiment: The transition log-likelihoods are scaled according to (15). Notice that setting the meta-parameter α to 1 corresponds to the baseline.

Table 2: Results with scaled log-likelihood.

	α=0.02	**α=0.2**	**α=0.75**	**Baseline**	**α=10**
DER [%]	45.20	**13.46**	18.55	17.18	23.08

The results are presented in table 2. The best results are obtained for $\alpha = 0.2$, outperforming significantly the baseline results. It shows that the baseline costs are too close one to each other. When α becomes very small, the transition costs become very large and the diarization relies mostly on them, giving unpredictable results. On the other hand, large α shadow the transitions and give results close to baseline without transition costs.

2nd Experiment: in this experiment we apply the powered inverse sum constraint, according to (17).

Table 3: Results with powered inverse sum.

	Baseline	**α=0.05**	**α=1.0**	**α=100**
DER [%]	17.18	98.13	**12.71**	23.93

Table 3 presents the related results. The HDM performs clearly better than the baseline with results a bit better than the ones of the previous experiments. Another time, the worst case gives results where the transition costs are very high.

In figure 5 we show how the DER depends on the hyper-parameter. Large value of the hyper-parameter leads to equal cost and the DER close to the no-cost DER. For very small values the cost are very large and the state distortions have no effect. This leads so all the data falls mainly to one cluster and the DER is extremely high. The optimal value is found empirically and in this experiment $\alpha = 1.0$ reaches the lowest DER (12.71%).

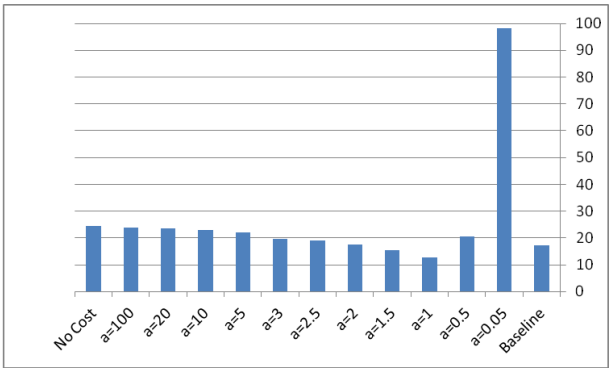

Figure 5: DER as a function of the hyper-parameter.

Table 4 presents some examples of transition costs for a given file. The cost variation is very large and has an important impact on diarization performance. Optimal costs allow an important gain in terms of DER compared to the baseline. In this example, the no-cost system performs worse than the baseline but the difference is not huge.

Table 4: Example of costs for different constraints for the file en_4065 (LDC).

Baseline – DER=12.08%		
0.03	4.45	4.23
5.73	0.01	5.83
5.83	6.14	0.01
Scaled likelihood, α=0.2 – DER=7.20%		
0.08	22.35	27.84
37.97	0.01	32.91
36.98	36.48	0.01
No Cost – DER=14.99%		

8.5. Experiments with NIST 2005

Following the best result obtained on the LDC database, we apply our HDM to the NIST 2005 database, using the same meta parameters.

Table 5: Results with LDC parameters on NIST 2005.

	Baseline	**scaled likelihood α=0.2**	**powered inverse sum α=1.0**	**No Cost**
DER [%]	14.56	18.98	16.28	17.96

Table 5 summarizes the results. The HDM performs clearly worse than the baseline and sometimes worse that the No Cost (without cost matrix) case. One explanation could be bad values of the meta-parameters, estimated on LDC and applied on NIST, knowing that these two databases are very different. To assess this explanation, we divided the database into a development set (500 conversations) dedicated to meta-parameter estimation and an evaluation set (1548 conversations) to compute the performance. The best results on the development set are given in table 6:

Table 6: Results with NIST 2005 development set.

	Baseline	**scaled likelihood α=0.8**	**powered inverse sum α=1.5**	**No Cost**
DER [%]	14.64	14.48	14.51	18.46

The first observation is that, as expected, good estimation of the scaling parameters can usually give results at least as good as the baseline system. However, even if HDM systems performed slightly better than the baseline, the improvement due to HDM is not clearly shown like for LDC.

Table 7: Results with NIST 2005 evaluation set.

	Baseline	**scaled likelihood α=0.8**	**powered inverse sum α=1.5**	**No Cost**
DER [%]	14.53	14.34	15.12	17.79

Table 7 presents the results obtained on the NIST evaluation set, using the meta-parameters estimated on the development set. The results are similar to the results presented in Table 6.

9. Conclusions and perspectives

In this work we defined a Hidden-Distortion-Model. This model allows exploring a large family of distortions and transition constraints. Our proposal includes also the classical HMM approach which becomes a specific case of HDM. We proposed different examples of transition cost models which do not require probabilistic assumptions.

The HDM, by the possibility to add some constraints on the transition costs, allows to scale the transition costs versus the state-models distortions such that more frequent transitions will have lower cost than the rare transitions (which is logical). An important difference between the standard HMM and our approach concerns the tied states, usually used to embed durations constraints. In HMM, tied states have transition probabilities of one (or in log domain, zero cost), while in the presented system, the costs can differ from zero and depend on the chosen constraints.

Several experiments with different costs were presented on telephone conversation (with two speakers) diarization. Our HDM approach was able to provide a significant improvement in performance on LDC (12.71% DER to be compared with 17.18% DER). It appeared that the hyper-parameters (scale or regularization parameter) tuning is important and depends on the data to be clustered: on NIST the HDM performed slightly better than the baseline only when the hyper-parameters are correctly tuned on a NIST development set (from 18.98% DER without tuning to 14.34% DER after tuning, to be compared with 14.56% DER for the baseline). It is also interesting to remark that, as expected, the state models are playing a more important role than the transition costs in the performance. For example, using equal cost for the transitions for the baseline system leads to an absolute DER loss of 7.19% for LDC and 3.4% for NIST.

It is also interesting to remark that the optimal costs are very different depending on the database: on LDC, the optimal log-likelihood scaling parameter is 0.2, which means multiplying the baseline system costs by a factor of five, when for NIST the optimal value is 0.8, which corresponds to add only 25% to the original costs. It means that the baseline HMM NIST 2005 costs are almost optimal and it is hard to have a significant improvement, while for LDC America CallHome the original costs are far to be optimal, and HDM gives much larger improvement.

In this paper, we focused on two transition cost systems when many other options could be examined. We showed that the choice of cost constraints should be driven by the targeted task, as the nature of the speech recordings seems to play a major role. In addition, the meta-parameters should be also optimized in order to match well with the data.

This study, we showed that the classical HMM-based clustering is a private case of a much wider family. Our approach allows a better modeling of the information gathered from the input data temporal sequence without to lose the well-known advantages of HMM/Viterbi systems.

The experimental part of this paper was done on a two speaker diarization task (but the task included the non-speech and overlapped speech detection). We wish to investigate in future works the role of our HDM approach in the case of recordings with unknown and large number of speakers. We hope that the flexibility of HDM, compared to HMM, will allow a better modeling of transition-related information.

10. References

[1] Ben-Harush, O., Lapidot, I., and Guterman, H., "Entropy based overlapped speech detection as a pre-processing stage for speaker diarization," *Interspeech*, September 6-10, 2009.

[2] Kenny, P., Reynolds, D. and Castaldo, F., "Diarization of Telephone Conversations using Factor Analysis," *IEEE Journal of Special Topics in Signal Processing*, 4(6):1059–1070, December 2010.

[3] Ajmera, J., Bourlard, H., Lapidot, I., and McCowan, I., "Unknown-multiple speaker clustering using HMM," *Proc. International Conference on Spoken Language Processing*, 573-576, September 16-20, 2002.

[4] Fredouille, C., Bozonnet, S., and Evans, N. W. D., "The LIA-EURECOM RT'09 Speaker Diarization System," *RT'09, NIST Rich Transcription Workshop*, May 28-29, 2009.

[5] Lim, T., Han, B., and Han, J. H., "Modeling and segmentation of floating foreground and background in videos," *Pattern Recognition*, 45(4):1696–1706, April 2012.

[6] Ye, J., Lazar, N. A., and Li, Y., "Sparse geostatistical analysis in clustering fMRI time series, " *Journal of Neuroscience Methods*, 199(2):336– 345, August 2011.

[7] Chamroukhi, F., Same, A., Aknin, P., and Govaert, G., "Model-based clustering with hidden Markov model regression for time series with regime changes," *Proc. of Int. Joint Conf. on Neural Networks*, 2814-2821, 2011.

[8] Oates, T., Firoiu, L., and Cohen, P. R., "Using dynamic time warping to bootstrap HMM-based clustering of time series," **Sequence Learning: Paradigms, Algorithms and Applications**, 1828:35-52, R. Sum and C. L. Giles, Ed. Springer, 2001.

[9] Bonastre, J.-F., Morin P., and Junqua, J.-C., "Gaussian dynamic warping (GDW) method applied to text-dependent speaker detection and verification," *Eurospeech*, September, 2003.

[10] Ben-Harush, O., Lapidot, I., and Guterman, H., "Weighted segmental K-means initialization for SOM-based speaker clustering," Interspeech, 2008.

[11] Kohonen, T. K., "The self-organizing map," *Proc. IEEE*, 78(9):1464-1480, September, 1990.

[12] I. Lapidot, "SOM as Likelihood Estimator for Speaker Clustering," Proc. Eurospeech'03, pp. 3001-3004, September 1-4, 2003, Geneva, Switzerland.

[13] Ben-Harush, O., "Speaker diarization," Ph.D. dissertation, Dept. Elect. And Comp. Eng., Ben-Gurion Univ., Beer-Sheva, Israel, 2010.

[14] Liguistic data consortium. LDC97S42, Catalog, 1997. Available: http://www.ldc.upenn.edu/Catalog.

[15] "National institute of standards and technology," The NIST 2005 Speaker Recognition Evaluation, 2005, available: http://www.itl.nist.gov/iad/894.01/tests/spk/2005.

[16] "Nist diarization criterion," available: http://www.itl.nist.gov/iad/mig/tools/.