Combining automatic segmentation algorithms for large corpora and application to speech synthesis (Contrat France Télécom n°3ZCIF402)

Safaa Jarifi, Dominique Pastor, Olivier Rosec

Collection
des Rapports
de Recherche
de l'ENST Bretagne

RR-2006006-SC

Fusion d'algorithmes de segmentation automatique de la parole de grands corpus et application à la synthèse vocale

Combining automatic segmentation algorithms for large corpora and application to speech synthesis

Contrat France Télécom N°3ZCIF402

2006

Safaa JARIFI,
Dominique PASTOR
Oliver ROSEC

Fusion d'algorithmes de segmentation automatique de la parole de grands corpus et application à la synthèse vocale

Combining automatic segmentation algorithms for large corpora and application to speech synthesis

Contrat France Télécom N° 3ZCIF402

Safaa JARIFI, Dominique PASTOR
ENST Bretagne, CNRS UMR 2872 TAMCIC
Olivier ROSEC
France Télécom, Division R&D, TECH/SSTP/VMI

Abstract

This paper concerns automatic segmentation algorithms of large speech corpora when the phonetic sequences of the speech signals are known. A direct and typical application is Text-To-Speech (TTS) synthesis by unit selection.

We start by proposing a general approach for combining of several segmentations produced by different algorithms. Several fusion methods are derived from this approach.

In the second part, we describe and analyse three automatic segmentation algorithms that will be used to evaluate our fusion approach. The first algorithm is segmentation by Hidden Markov Models (HMM). The second one, called refinement by boundary-model, aims at improving the segmentation performed by HMM via a Gaussian Mixture Model (GMM) of each boundary. The third one is a slightly modified version of Brandt's Generalized Likelihood Ratio (GLR) method; its goal is to detect signal discontinuities within some interval determined by the HMM boundaries.

Performance measurements show that refinement by boundary-model is the most accurate in the sense that its segmentation marks are the closest to the manual ones.

When applied to the three segmentations obtained by the three algorithms mentioned above, any of these fusion methods is more accurate than refinement by boundary-model. With respect to the corpora considered in this paper, the most accurate fusion method, called *optimal fusion by soft supervision*, reduces by 25.5% (resp. 60%) the number of the segmentation errors made by refinement by boundary-model (resp. standard HMM segmentation).

Subjective tests are carried out in the context of corpus-based speech synthesis. They show that the quality of the synthesized speech obtained when the speech corpus is segmented by *optimal fusion by soft supervision* approaches that obtained when the same corpus is manually segmented.

Keywords: Automatic speech segmentation, speech synthesis, HMM, Brandt's GLR algorithm, refinement by boundary-model, mark selection, soft supervision, hard supervision, subjective test.

Résumé

Ce papier concerne la segmentation automatique de grands corpus de parole quand la séquence phonétique des signaux de parole est connue et est supposée correcte. Une application typique de ce genre de corpus est la synthèse vocale par sélection d'unités.

Nous allons dans un premier temps proposer une approche générale pour combiner plusieurs segmentations produites par différents algorithmes. Ensuite, plusieurs méthodes de fusion seront dérivées de cette approche.

Dans une deuxième partie, nous décrivons et analysons trois algorithmes qui vont être utilisés pour évaluer les différentes méthodes de fusion définies. Le premier algorithme est la segmentation par HMM (Hidden Markov Model). Le deuxième, appelé post-traitement par modèle de frontière consiste, à améliorer les marques de la segmentation produite par HMM avec des modèles de frontières. Ces modèles de frontière sont des GMM (Gaussian Mixture Model). Le troisième est une version légèrement modifiée de l'algorithme de Brandt. L'objectif de cet algorithme est de détecter les discontinuités du signal de parole dans un intervalle déterminé par les marques par HMM.

L'évaluation des performances de ces algorithmes montre que le post-traitement par modèle de frontière produit la segmentation la plus précise et donc la plus proche de la segmentation manuelle.

L'évaluation de l'application des méthodes de fusion aux trois algorithmes que nous venons de citer a montré que le taux de segmentation correcte est amélioré par rapport à celui du post-traitement par modèle de frontière. En effet, avec la meilleure méthode de fusion, on arrive à réduire de 25.5% (respectivement 60%) le nombre d'erreurs de segmentation générées par le post-traitement par modèle de frontière (respectivement la segmentation standard par HMM).

Les tests subjectifs réalisés dans le contexte de la synthèse vocale montre également que la qualité de la parole synthétique obtenue avec la segmentation issue de la meilleure méthode de fusion approche celle issue de l'utilisation de la segmentation manuelle.

Mots clés: Segmentation automatique, synthèse vocale, HMM, algorithme de Brandt, post-traitement par modèle de frontière, sélection des marques, supervision douce, supervision dure, test subjectif.

1 Introduction

Corpus-based speech synthesizers are very popular for the synthesized voice quality they achieve. By selecting and concatenating speech segments or units stored in a large database, such synthesizers can select a sequence of units that corresponds to the context of the entry text. By so proceeding, the naturalness of the synthetic voice is significantly improved. Nevertheless, the creation of new voices with this approach is extremely expensive because of the different procedures that must be applied to the speech corpus to obtain the dictionary of units.

Two important procedures are the transcription and the segmentation of the speech signals recorded by a speaker.

This paper focuses on the segmentation phase for the following reason. In fact, automatic segmentation methods are not always accurate enough. Manual checking remains necessary to correct segmentation errors. Thus, with the increasing demand for more synthetic voices, there is still the need to improve the automation and accuracy of the segmentation process for TTS synthesis applications.

Up to now, the HMM approach [10, 4] has been the most widely used for automatic segmentation and it is considered as the most reliable. This approach is linguistically constrained because it needs the true phonetic sequence associated with the recorded utterances in order to estimate the HMM sequence. However, this approach still has some limitations for building voices for TTS systems. The main limitation is that HMMs model steady areas well but are not really suited to detecting locally the transitions between phonemes in a speech signal.

Other automatic segmentation methods are suitable for detecting segmentation marks. Some of these methods are based on the refinement of marks produced by the HMM approach. Such methods can be based on boundary models [9]. Other methods are also suitable for segmenting speech signals by detecting their discontinuities. Brandt's GLR algorithm [3] is one of these. Nevertheless, it produces insertions and omissions because it is linguistically unconstrained.

With respect to the foregoing, the purpose of this paper is to combine global and local automatic segmentation algorithms in order to improve the accuracy of the resulting automatic segmentation. This is the aim of section 2. Several fusion methods are proposed. They are based on a general scheme presented in section 2 for the linear combination of segmentation

marks.

Then, in order to evaluate the performance of some fusion methods, we study and justify the choice of three automatic segmentation algorithms in section 3. The first one is HMM segmentation. It applies a forced alignment between the HMM sequence and the speech signal.

The second segmentation algorithm is refinement by boundary-model. It was originally proposed in [9] to segment a Chinese corpus. It uses a boundary model, which is estimated on a small database, to refine the HMM segmentation marks.

The third algorithm is Brandt's GLR method whose aim is to detect discontinuities in speech signals. Basically, and unlike segmentation by HMM and refinement by boundary-model, this method needs no prior knowledge of the transcription. Since, in the context of corpus-based speech synthesis, the transcription is available, we adapt this method so as to take this transcription into account.

The accuracies of these automatic segmentation methods are then evaluated in section 4 on a French and on an English corpus. These accuracies are computed at a tolerance of 20 ms with respect to a manual segmentation. A tolerance of 20 ms is considered as an acceptable limit in order to produce synthesized speech of good quality. By manual segmentation, we mean the segmentation resulting from the manual checking of standard HMM segmentation. In this section, we show that the three algorithms are complementary in the sense that they are adpted to detect different types of boundaries.

Section 5.1 evaluates the accuracy of some fusion methods and section 5.2 presents the results of subjective tests that evaluate the speech quality when the best fusion method is used to segment the French and the English corpora. The last section concludes this paper and proposes some extensions.

2 A general fusion approach for combining segmentations

Generally, segmentation algorithms behave differently according to the transitions they are asked to detect. The main idea here is to take into account these different behaviours so as to favour more some segmentation marks rather than others, given a certain type of transition to detect.

More specifically, let s be a transition to be detected between two phonemes

and assume that the phonetic class of the phoneme to the right (resp. to the left) of s is c_r (resp. c_ℓ). The principle of the proposed method is to compute a new estimate $\widehat{t}(s)$ of the transition instant on the basis of K time instants $t_1(s), \ldots, t_K(s)$ produced by K segmentation algorithms.

This can be regarded as a problem of fusion. The solution we propose is based on a linear combination of selected segmentation marks. The estimate $\widehat{t}(s)$ is the barycentre given by:

$$\widehat{t}(s) = \sum_{k \in A} \beta_k(c_\ell, c_r) t_k(s), \tag{1}$$

where A is the index set of the selected marks and the coefficients $\beta_k(c_\ell, c_r)$ satisfy the relation

$$\sum_{k \in A} \beta_k(c_\ell, c_r) = 1.$$

The estimate given by equation (1) corresponds to the case of algorithms that make no systematic error (similar errors for similar transitions). If any algorithm, say the k^{th}, made a known systematic error, it would suffice to replace in equation (1) the corresponding estimate $t_k(s)$ by $t_k(s) - m_k$ where m_k is the value of this error.

Figure 1 summarizes the computation of $\widehat{t}(s)$. We now describe the several components of this fusion scheme. To the authors' best knowledge, the fusion scheme we propose is not usual for combining segmentation marks.

By introducing the *mark selection* whose outcome is the index set A used to compute $\widehat{t}(s)$, we take into account the existence of criteria that make it possible to select marks independently of the coefficients $\beta_k(c_\ell, c_r)$. To better understand the purpose of this phase, let us consider 6 different algorithms. Assume that 5 of these algorithms detect the time instant of the transition s within the same interval and that the sixth algorithm gives an estimate of this time instant significantly further away from the other ones. In this case, it is likely that the time instant performed by the sixth algorithm is not correct and, thus, a simple average of the 6 estimations will be less accurate than the average of those located in the same interval. This example shows that it can be relevant to select some estimates among those available in order to compute $\widehat{t}(s)$. For instance, we will use the distance as a criterion to select marks.

The coefficients $\beta_1(c_\ell, c_r), \beta_2(c_\ell, c_r), \ldots, \beta_K(c_\ell, c_r)$ are obtained as follows. With the notations introduced above, we start by *scoring* the K algorithms

on the basis of a training database. The scores $\gamma_k(c_\ell, c_r), k = 1, \ldots, K$, must quantify the respective behaviour of the algorithms for detecting the transition between the classes c_ℓ and c_r. For example, a large value for $\gamma_k(c_\ell, c_r)$ should be assigned to the k^{th} algorithm, if this algorithm performs well on the pair of classes (c_ℓ, c_r). This scoring phase is performed once for all. We thus obtain a set of scores for all the algorithms and for all the pairs of phonetic classes present in the training corpus.

Then, we transform the sequence $\gamma_k(c_\ell, c_r), k = 1, \ldots, K$, of scores into a sequence $\omega_1(c_\ell, c_r), \omega_2(c_\ell, c_r), \ldots, \omega_K(c_\ell, c_r)$ of weights. This step is the *score supervision*. The weights indicate the quality of each algorithm in comparison with the others. In fact, the role of this phase is similar to that of a supervisor who decides to favour some algorithms rather than others on the basis of his experience, his prior knowledge, some heuristic and so forth. Note that the computation of the weights can be achieved regardless of the scores. In particular, the score supervision can favour no algorithm by simply assigning the same weight to every algorithm (see section 2.2.1). Note that if the transitions between two classes c_i and c_j are absent from the training database, $\omega_k(c_i, c_j)$ is not defined and thus, we force $\omega_k(c_i, c_j)$ to 1 for every k.

Only the weights corresponding to the selected marks are normalized to produce the coefficients $\beta_1(c_\ell, c_r), \beta_2(c_\ell, c_r), \ldots, \beta_K(c_\ell, c_r)$:

$$\beta_k(c_\ell, c_r) = \frac{\omega_k(c_\ell, c_r)}{\sum_{j \in A} \omega_j(c_\ell, c_r)}, k = 1, \ldots, K.$$

To perform the combination, we must choose the type of score, the mark selection and the supervision. Many choices are possible. In what follows, we propose and discuss some simple and efficient choices. The analysis of more sophisticated choices is in progress.

In this paper, we propose the accuracy at 20 ms as the score. We choose this score because, on the one hand, we are interested in the precision of the segmentation at a tolerance of 20 ms; on the other hand, this score is a reliable measure of the ability of a given algorithm to detect a type of transition. Other types of score can certainly be proposed.

In the following subsections, we describe two basic choices for the mark selection and three possible types of supervision.

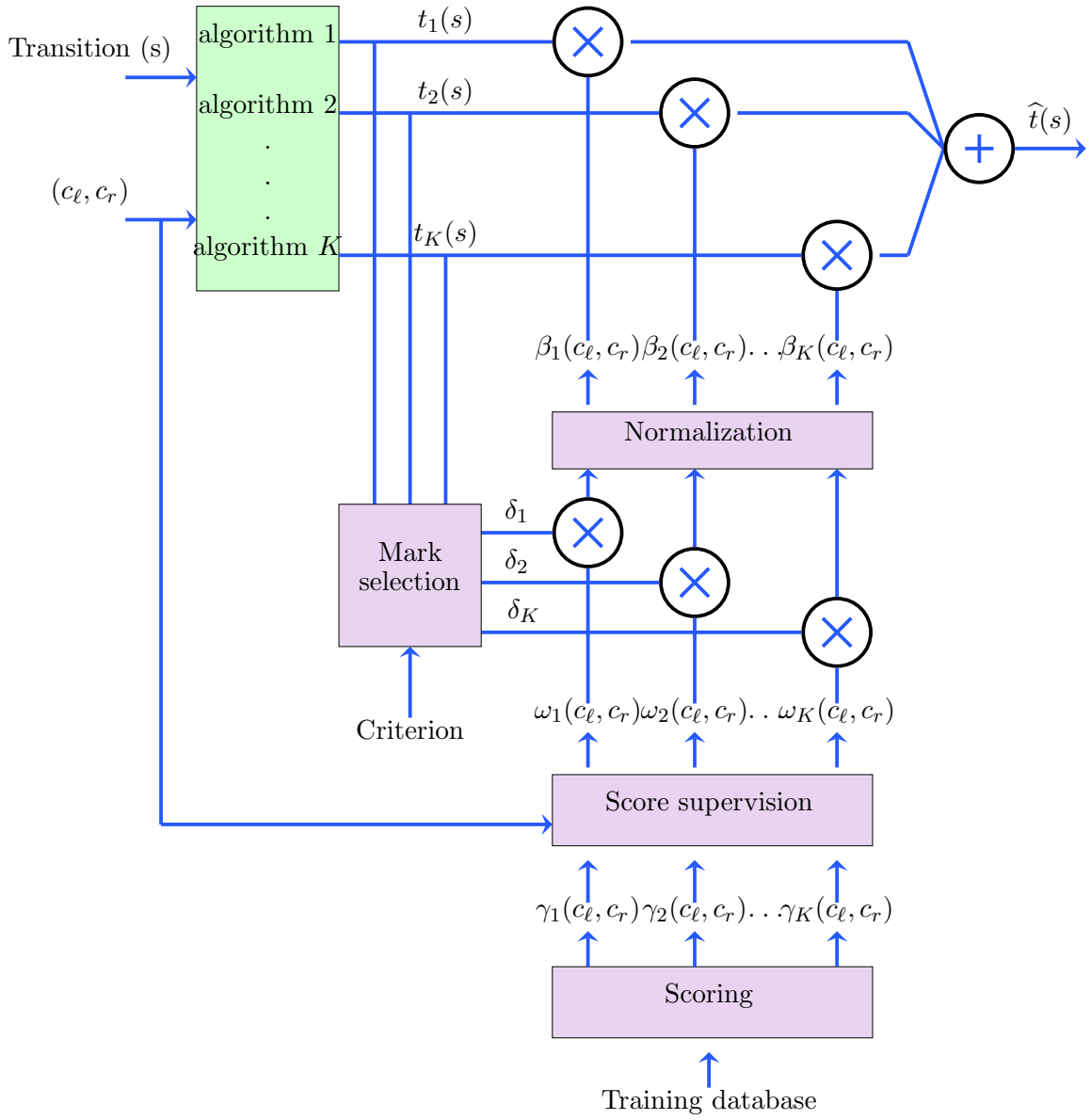

Figure 1: General scheme for computing $\widehat{t}(s)$ by linear fusion of segmentation marks. We have δ_k equal to 1 if $k \in A$, where A is the index set of the selected marks, and 0 otherwise

2.1 Mark selection

The mark selection involves choosing, for each transition, the marks of the algorithms that will be used to estimate the transition time instant. It is thus achieved by a function:

$$f \quad : \mathbb{R}^K \rightarrow \{0,1\}^K$$

$$(t_1, \ldots, t_K) \rightarrow (\delta_1, \ldots, \delta_K)$$

2.1.1 Total selection

This is the basic case where we keep the K marks produced by the K algorithms. Therefore, we have $\delta_k = 1$ for each k and $A = \{1, 2, \ldots, K\}$.

2.1.2 Partial selection

Partial selection involves choosing a subset of the K marks we have. This selection is achieved in two steps. The first step is to determine clusters of marks located within the same zone. Here, we use a distance to find these clusters. The second step is to choose one or more clusters on the basis of a criterion. For example, we can choose the cluster that contains the largest number of marks.

The separation of the marks into clusters is a complicated problem in the general case. Relatively sophisticated algorithms, such as k-NN (k-Nearest Neighbours) and genetic algorithms, can be used.

When $K = 3$ and since segmentation marks are real numbers, which is our case, the marks can be easily determined as follows. We compute the distance d_{ij} between the boundaries $t_i(s)$ and $t_j(s)$ obtained with the i^{th} and j^{th} algorithms respectively, where $(i, j) \in \{1, 2, 3\} \times \{1, 2, 3\}$ and $i \neq j$. The selected marks are those that minimize the distance. With this definition, we choose only one cluster, which is the set of those marks that minimize the distance. This cluster is directly A and A contains two or three indices. In fact, it suffices that two distances are equal to each other to have A equal to $\{1, 2, 3\}$. In what follows, this selection will be called *partial selection by distance criterion*.

2.2 Score supervision

The score supervision is basically a function that assigns the weights $\omega_1(c_\ell, c_r)$, $\ldots, \omega_k(c_\ell, c_r)$ to the scores $\gamma_1(c_\ell, c_r), \ldots, \gamma_k(c_\ell, c_r)$. In this paper we consider the particular case where the computation of the weights is achieved by using one single function f, called a *weighting function*, such that $\omega_k(c_\ell, c_r) = f(\gamma_k(c_\ell, c_r))$ for $k = 1, \ldots, K$.

Equation (1) becomes:

$$\widehat{t}(s) = \frac{\sum_{k \in A} f(\gamma_k(c_\ell, c_r)) t_k(s)}{\sum_{k \in A} f(\gamma_k(c_\ell, c_r))}. \tag{2}$$

Remark 2.1 *The supervision must be adapted to the type of score. If the larger the score $\gamma_k(c_\ell, c_r)$, the more accurate the k^{th} algorithm, the weighting function f must be non-decreasing. Otherwise, if the larger the score $\gamma_k(c_\ell, c_r)$, the less accurate the k^{th} algorithm, the weighting function f must be non-increasing.*

2.2.1 Uniform supervision

This is the simplest supervision that we can suggest: $f(\gamma_k(c_\ell, c_r))$ is equal to 1, for every type of score, every algorithm and every type of transition. In other words, the supervisor favours no algorithm. The outcome of the linear fusion is thus the average value of the selected marks:

$$\widehat{t}(s) = \frac{1}{K} \sum_{k \in A} t_k(s). \tag{3}$$

2.2.2 Hard supervision

The weights assigned by the supervision are 0 or 1, hence the name *hard supervision*. These binary weights are computed as follows. Let $\gamma_{\max}$ be the maximum value of the scores $\gamma_k(c_\ell, c_r), k = 1, 2, \ldots, K$. The elements of the set $I = \{k : \gamma_k(c_\ell, c_r) = \gamma_{\max}\}$ are the most appropriate algorithms for detecting transition s. In this case, the weighting function f is defined by:

$$f(\gamma_k(c_\ell, c_r)) = \begin{cases} 1 & \text{if} \quad k \in I \\ 0 & \text{otherwise} \end{cases}$$

The estimate $\widehat{t}(s)$ is then given by:

$$\widehat{t}(s) = \frac{1}{Card(I \cap A)} \sum_{k \in I \cap A} t_k(s), \tag{4}$$

where $Card(I \cap A)$ is the cardinality of $I \cap A$.

2.2.3 Soft supervision

In contrast to hard supervision, soft supervision assigns a non binary value. In this paper, we propose two different weighting functions valued in $\mathbb{R}$. These functions are increasing ones. This follows from remark 2.1 since the score we consider is the accuracy at 20 ms and thus the larger the score, the larger the weight must be.

The two weighting functions studied in this paper are:

$$f(x) = x$$

and

$$f(x) = \frac{1}{1-x},$$

where x is the accuracy at 20 ms.

Many other functions can be proposed. With the first function, we consider that the accuracy is a sufficiently good confidence measure. Since x is the accuracy at 20 ms, $1 - x$ is the error rate at 20 ms; therefore, the value of the second function at x is the inverse of this error rate. Similarly to the accuracy, the inverse error rate at 20 ms is also a good confidence measure. With this second function, we discriminate more between the different algorithms. For instance, given a pair of classes (c_ℓ, c_r), suppose that the accuracies at 20 ms of 2 algorithms are 80% and 90% respectively. The corresponding inverse error rates are then 0.05 and 0.1. The weight of the second algorithm is thus twice as large as that of the first one when the fusion by soft supervision is performed on the basis of the inverse error rates, whereas the weights in terms of accuracies are of the same order.

Remark 2.2 *Note the analogy between the "hard" and "soft" supervisions proposed above and hard and soft fusion procedures such as those described in [13] or [16].*

3 The three automatic segmentation algorithms

We describe the three segmentation algorithms that will be combined via the general fusion approach proposed above. The three algorithms are: segmentation by HMM, refinement by boundary-model and Brandt's GLR method. This choice is justified by the fact that the algorithms behave differently following the classes of the transitions to be detected. In this sense, we can say that these algorithms are complementary.

3.1 Segmentation by HMM

This approach is considered as the standard method for speech segmentation and basically consists of two main steps. The first step is training that aims at estimating the acoustic models. The second step uses these models to segment the speech signal by means of the Viterbi algorithm. The latter applies a forced alignment between the models associated with the known phonetic sequence and the speech signal.

The training phase is crucial because the accuracy of the segmentation by HMM depends closely on the quality of the estimated models and thus on the initialization of these models. To initialize the models, several methods exist.

For example, we can use iterative training [8] on the whole corpus. The boundaries resulting from the previous iteration are used to initialize and re-estimate the models via the Baum-Welch algorithm After a few iterations of the training process, mismatches between the manual labels and the phone labels produced by the HMM approach are significantly reduced. The approach that uses this type of training is standard HMM segmentation. This approach is our reference.

Another method that can be considered is illustrated in figure 2. It uses a small speech database segmented and labelled manually to estimate the models [5]. Then, we segment the whole corpus with these models. The initialization of these models is the same as in the first method. If the small corpus contains several realizations of each phone of the database, the initialization of the models on this small corpus is good and this latter processing performs better than the former [7]. For this reason, we prefer to apply the general fusion approach to the HMM segmentation that uses this training. In what follows, we call *HMMSeg* the segmentation performed by using this training procedure.

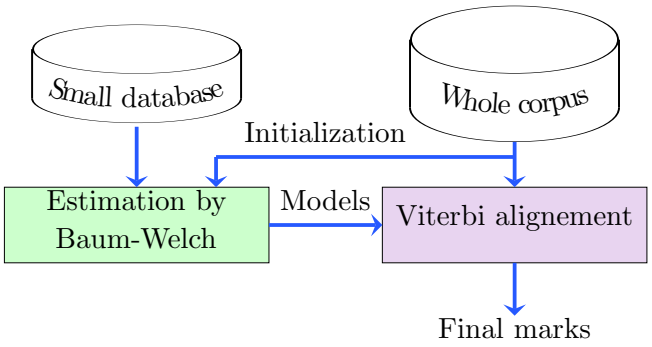

Figure 2: Segmentation by HMM based on a small corpus manually segmented

3.2 Refinement by boundary model [9]

The main idea of this method is to train a set of boundary models by using a small database manually segmented and labelled. Then, these models serve to refine an initial segmentation ([9]). More specifically, this method is carried out in two steps as shown in figure 3.

For each boundary of the training database, we create a super vector by concatenating the acoustic vectors of the $(2N + 1)$ frames that are around the manual boundary (see figure 4). Since each boundary B depends on the phoneme X to its left and on the phoneme Y to its right, the boundary B is henceforth called the pseudo-triphone $X - B + Y$ as proposed in [9] (see figure 5). Because the number of labelled data is limited in practice, the pseudo-triphones are clustered into a reduced number of classes via a Classification And Regression Tree (CART). A Gaussian Mixture Model (GMM) is then estimated for each class. The questions put during the construction of the CART concern the phonetic classes and phonemic identity.

The second step aims at refining each boundary of an initial segmentation. Given a labelled sentence and its initial segmentation, we seek in a certain vicinity of each boundary the time instant that maximizes the likelihood of the super vector corresponding to this instant. This likelihood is computed as follows. For each possible time instant around the initial boundary, we form a super vector centred on the current frame as in the training step; since this super vector is assumed to represent a given pseudo-triphone, we use the

CART [11] to determine the class corresponding to this pseudo-triphone; the likelihood is finally calculated according to the GMM associated with the class thus obtained and the super vector.

This algorithm is linguistically constrained because it needs prior knowledge of the phonetic sequence in order to create the boundary models. However, it can be applied to any segmentation that contains no omission and no insertion. For example, in [9], refinement by boundary-model was applied to HMM segmentation based on forced alignment.

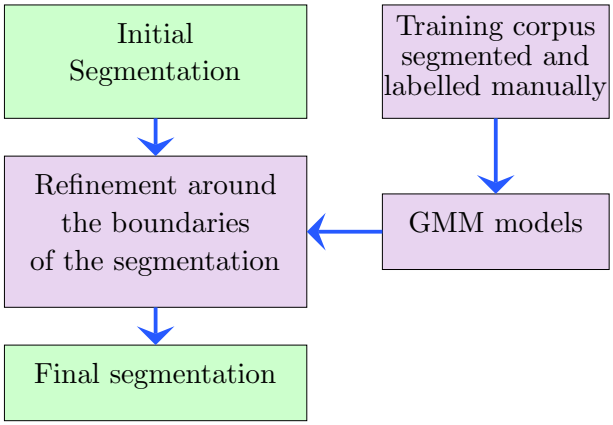

Figure 3: The different steps of refinement by boundary-model

3.3 Brandt's GLR algorithm

3.3.1 The basic algorithm

The aim of this method is to detect discontinuities in speech signals. Speech signals are assumed to be sequences of homogeneous segments. Each segment w is a finite sequence $w = (y_n)$ of samples that are assumed to obey an autoregressive (AR) model:

$$y_n = \sum_{i=1}^{p} a_i y_{n-i} + e_n$$

In this equation, p is the model order, which is assumed to be constant for all the segments, and e_n is a zero mean white Gaussian noise with variance

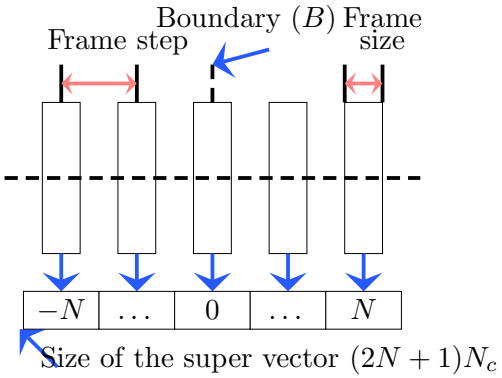

Figure 4: Construction of a super vector. We consider N non overlapping frames to the right and N non overlapping frames to the left of a boundary. In addition, we take into account the frame centred on the boundary. The $(2N+1)$ acoustic vectors of these $(2N+1)$ frames form the super vector.

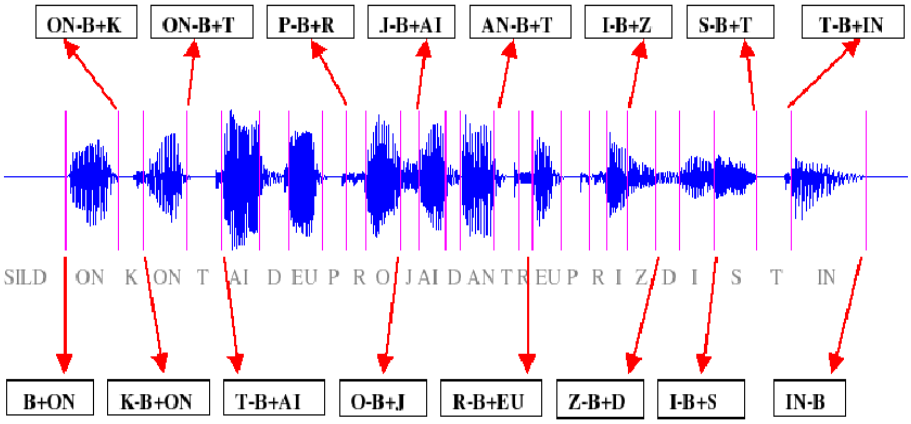

Figure 5: The pseudo triphones of the French sentence "On comptait deux projets d'entreprise distincts"

equal to σ^2. Such a segment is thus characterized by the parameter vector $\Theta = (a_1, \ldots, a_p, \sigma)$. Let w_0 be some segment of N samples and Θ_0 be the corresponding parameter vector. Brandt attempts in [3] to decide whether

w_0 should be split into two subsegments w_1 and w_2 or not. A possible split results from the detection of a jump between the parameter vectors Θ_1 and Θ_2 of w_1 and w_2 respectively. Brandt's GLR method decides that such a jump has occurred by comparing:

$$max_r(D_N(r))$$

to a predefined threshold λ. In this equation, for the Gaussian case, $D_N(r)$ has a simple expression :

$$D_N(r) = Nlog\hat{\sigma}_0 - rlog\hat{\sigma}_1 - (N-r)log\hat{\sigma}_2$$

Note that D_N is the generalized likelihood ratio (GLR). In the equation above, r is the size of the time interval covered by w_1, whereas $\hat{\sigma}_1$ and $\hat{\sigma}_2$ are the noise standard deviation estimates of the models characterized by the parameter vectors Θ_1 and Θ_2 respectively. Thus, the change instant is the value $\hat{r} = arg(max_r(D_N(r)) \geq \lambda)$.

A direct implementation of this method is computationally expensive. A sub-optimal version is recommended in [2]. In particular, the length of w_2 is fixed to a predefined value L. For further details, the reader can refer to [2].

3.3.2 Brandt's GLR algorithm with known phonetic transcription

As mentioned above, the purpose of Brandt's GLR method is to detect discontinuities of speech signals without any further knowledge of the phonetic sequence. This algorithm is linguistically unconstrained and makes insertions and omissions.

We propose here an adaptation of Brandt's GLR method when the pronounced phonetic sequence is available as often assumed for the segmentation of speech synthesis corpora. In such a case, an initial segmentation can be obtained, for example, by an HMM-based method. For each initial segmentation mark, we define a time interval over which a modified version of Brandt's GLR method is applied so as to provide one single segmentation mark.

More specifically, let $(U_0, U_1, \ldots, U_L)$ be the boundaries of the initial segmentation. For i in $\{1, \ldots, L-1\}$, we seek a speech discontinuity between $V_i = \frac{(U_{i-1}+U_i)}{2}$ and $V_{i+1} = \frac{(U_i+U_{i+1})}{2}$ by determining the time instant that maximizes the GLR. By removing the thresholding, we make no omission and no insertion. This is the method used below and despite the modification proposed, we still call it Brandt's GLR method.

4 Evaluation of the three segmentation algorithms

The general fusion approach proposed in section 2 is basically aimed at achieving a more accurate segmentation than that produced by the different segmentation algorithms that are combined. Hence, we evaluate the performance of each of the three algorithms proposed above and we will verify that they are complementary.

4.1 Description of the corpora

The performance of each algorithm is evaluated on a French and on an English corpus. The French corpus, hereafter called *FRcorpus*, contains 7300 sentences uttered by a woman and sampled at 16 kHz. The English corpus, called *ENcorpus*, is also pronounced by a female speaker and sampled at 16 kHz. It contains 8900 sentences. The training phase of refinement by boundary-model and that required to compute *HMMSeg* (see section 3.1) are carried out successively on databases containing 100, 300 and 700 sentences.

Each database is chosen randomly within the speech corpora. This random choice is made up to a minimum number of realizations per phone. We choose this minimum equal to 3.

In order to provide a rigorous analysis, we use a cross-validation procedure that includes three different training databases of the same size.

4.2 Parameters

The segmentation *HMMSeg* is performed by using the HTK tool [17] for the acoustic analysis, the model training and the segmentation. For each phone, we consider a left-to-right three-state model; the observation probabilities are modelled by the mixture of two Gaussian distributions. The acoustic vectors contain 39 coefficients each. These coefficients are the 12 Mel Frequency Cepstral Coefficients (MFCCs), the normalized energy, and the first and second derivatives of these 13 coefficients. Twenty iterations of the Baum-Welch algorithm are applied to train the HMM.

The segmentation obtained by applying refinement by boundary-model to *HMMSeg* is called *RefinedHMMSeg*. For every boundary of the HMM segmentation, the refined boundary is searched for within an interval of 60

ms centred on this boundary with a search step fixed to 5 ms. The super vector is computed with $N = 2$, a frame step equal to 30 ms and a frame size equal to 20 ms. The length of the acoustic vector of each frame is still 39. Thus, each super vector contains $39(2 \times 2 + 1) = 165$ coefficients. The parameter values given above were originally determined for a Chinese corpus in [9]. In [6], it is shown that these values remain suitable for a French corpus.

For Brandt's GLR method, the input segmentation is *HMMSeg*. Therefore, we search for a discontinuity around each HMM boundary. The segmentation obtained is called *BrandtSeg*. The AR model order is set to 12 and the minimal length of w_1 and w_2 is equal to 10 ms.

4.3 Results and discussion

All the accuracies presented below were computed on the whole corpus except the sentences used for the training processes.

Tables 1 and 2 depict the accuracies at a tolerance equal to 20 ms with respect to the manual segmentation.

We choose a tolerance of 20 ms because, as mentioned in the introduction, this tolerance is an acceptable limit for TTS applications.

Table 1: Accuracies of the standard HMM segmentation

	Accuracies
FRcorpus	88.53%
ENcorpus	87.77%

Table 2: Accuracies of *HMMSeg*, *RefinedHMMSeg* and *BrandtSeg*

	AlgSize	*HMMSeg*	*RefinedHMMSeg*	*BrandtSeg*
FRcorpus	100	91.71%	91.08%	83.22%
ENcorpus		91.98%	89.58%	86.78%
FRcorpus	300	92.51%	93.26%	83.39%
ENcorpus		92.95%	92.46%	87.10%
FRcorpus	700	92.47%	94.00%	83.38%
ENcorpus		93.00%	93.50%	87.09%

As explained in section 4.1, *HMMSeg* and *RefinedHMMSeg* require a preliminary training phase. The size of the database used for this training phase

Table 3: The performance limit of each algorithm

	HMMSeg	*RefinedHMMSeg*	*BrandtSeg*
FRcorpus	92.68%	95.00%	83.22%
ENcorpus	93.17%	94.30%	87.19%

is hereafter called *AlgSize*. Three different values for *AlgSize*, namely 100, 300 and 700, are tested. Table 1 presents the accuracies of the standard HMM segmentation which is our reference and table 2 presents those of the segmentations *HMMSeg*, *RefinedHMMSeg* and *BrandtSeg*.

According to these tables, we can make the following remarks:

- *HMMSeg* is always more accurate than the standard HMM segmentation. This shows that an initialization of the models via a small manually segmented database yields better results than a standard HMM initialization based on the whole corpus;

- *RefinedHMMSeg* is more accurate than *HMMSeg* provided that the boundary models are well trained i.e. if the number of boundaries available in the training database is large enough. 300 sentences for the French corpus and 700 for the English corpus are sufficient for *RefinedHMMSeg* to outperform the others; 300 sentences of *FRcorpus* correspond approximatly to 10000 boundaries and 700 sentences of *ENcorpus* contain around 30000 boundaries;

- Brandt's GLR method is inaccurate at 20 ms in comparison with the other algorithms.

Table 3 shows the performance limit of each algorithm. The performance limit of a given algorithm is the accuracy obtained by training this algorithm on the whole database. With a training database of 700 sentences, we can observe that the accuracies of the three algorithms approaches those of their perfomance limits.

Because the accuracy at 20 ms is regarded as an important objective criterion in TTS applications, it seems reasonable to conclude from table 2 that refinement by boundary-model is the most accurate algorithm. Nevertheless, we should not forget that the algorithms are not suitable for the same phonetic classes. During these tests, it turned out that Brandt's GLR method

detects some boundaries well, like silence/speech and voiced/voiceless transitions. Similarly, refinement by boundary-model and the HMM approach well detect the time instants of some transitions that are not well estimated by Brandt's GLR method. We say in this case that refinement by boundary-model and Brandt's GLR method are complementary.

To convince the reader that the three algorithms behave differently, for the french corpus, we determined (see table 4) the best algorithm for each pair of phonetic classes. To construct this table, by using the same test corpus, we computed the error rate at 20 ms for each algorithm and each class of transition. We observe from this table that each algorithm is useful. In fact, for a given algorithm, there is a number of transition classes for which this algorithm gives the most accurate marks. For example, Brandt's GLR method is the best algorithm for detecting the boundaries between voiced plosives and unvoiced plosives. Refinement by boundary-model is the best algorithm to find marks between nasal vowels and unvoiced plosives. Finally, the HMM approach is the most adapted to detecting the marks between voiced plosives and nasal vowels.

5 Experimental results for the general fusion approach

In section 5.1, we presented a score, two types of selection marks and three types of supervisions. In what follows, by fusion method, we mean a linear combination defined by a score, a mark selection, a score supervision and a weighting function. For hard and uniform supervisions, the weighting functions are determined. However, for soft supervision, we defiend two possible weighting functions. Therefore, we have 8 possible combinations between the score, the two selection marks, the three supervisions and the two weighting functions of the soft supervision. These 8 combinations are our fusion methods.

In this section, we start by identifying the best fusion method among the 8 chosen. This is achieved in section 5.1 by comparing the segmentation accuracies computed when the fusion methods are applied to the triplet (*HMMSeg*, *RefinedHMMSeg*, *BrandtSeg*).

We conclude from this comparison that fusion with total selection, and soft supervision of the inverse error rates, that is, when $f(x) = \frac{1}{1-x}$, give the

Table 4: The best algorithm for each pair of phonetic classes and for the French corpus. The terms "H", "G" et "B" refer to HMM segmentation, refinement by boundary-model and Brandt's GLR method respectively. The French phonetic classes are: oral vowels (OV), nasal vowels (NV), unvoiced plosives (UVP), voiced plosives (VP), unvoiced fricatives (UVF), voiced fricatives (VF), diphthongs (DIPH), nasal consonants (NC), liquid consonants (LC), semivowels (SV), pauses (SP) and silences (SIL). $---$ means that no transition between the pair of classes is available in the corpus.

	OV	*NV*	*DIPH*	*VP*	*UVP*	*VF*	*UVF*	*NC*	*LC*	*SV*	*SP*	*SIL*
OV	B	G	B	B	G	H	H	B	G	B	B	B
NV	G	G	B	H	G	B	B	B	H	B	H	H
DIPH	H	H/G	B	B	B	H	H	B	G	G	B	H
VP	H	G	H	B	B	H	G	H	G	H	B	B
UVP	H	H	H	B	G	H	H/G	G	G	H	B	G
VF	G	G	H	B	B	H	B	H	B	H	B	B
UVF	H	H/B	G	H/B	G	H	B	H/G	B	H	G	B
NC	G	G	H	H	G	H/G/B	G/B	B	H	G	G	B
LC	G	H	H	B	B	G/B	B	G	B	H/G	G	G
SV	G	G	B	B	B	B	B	B	B	G/B	G	B
SP	B	G/B	H/G/B	G	G	G	H	G	G	B	H/G/B	H/G/B
SIL	B	B	H/G/B	G	G	B	G	G/B	G	G/B	H/G/B	H/G/B

best results. For the sake of brevity, this fusion method will be called *the optimal fusion by soft supervision*. Then, in section 5.2, we compare the quality of the synthetic speech obtained by using the segmentation produced by this fusion method to that achieved when the HMM and manual segmentations are used.

5.1 Accuracies

Let *SizeComb* denote the number of sentences of the training database used to score *HMMSeg*, *RefinedHMMSeg* and *BrandtSeg*. Three different values for *SizeComb* are considered: 100, 300 and 700. The sentences of the training databases used for the scoring are chosen randomly within the whole corpus and are different from those used for the training required to compute *HMMSeg* and *RefinedHMMSeg*.

The accuracies given in this section are computed at a tolerance of 20

ms and evaluated on all the sentences of the database except those needed to train the models for the computation of *HMMSeg*, *RefinedHMMSeg* and the different fusion methods we use to combine them. As in section 3, the results presented here are obtained by averaging the accuracies using a cross-validation procedure.

For *FRcorpus*, the fusion was achieved by using 12 classes: unvoiced plosives, voiced plosives, unvoiced fricatives, voiced fricatives, oral vowels, nasal vowels, diphthongs, nasal consonants, liquid consonants, semivowels, pauses and silences. For *ENcorpus*, 11 classes were considered: vowels, voiced/unvoiced plosives, voiced/unvoiced fricatives, affricates, nasal consonants, liquid consonants, semivowels, pauses and silences.

The accuracies at a tolerance of 20 ms achieved by the fusion methods on *FRcorpus* and *ENcorpus* are given in tables 5 and 6. For every pair (*SizeComb,AlgSize*), every fusion method yields a segmentation more accurate than *HMMSeg*, *RefinedHMMSeg* and *BrandtSeg*. For instance, according to table 5, *optimal fusion by soft supervision* achieves an accuracy of 94.98% for *FRcorpus* when $(SizeComb, AlgSize) = (300, 300)$. If we compare this accuracy to those given in table 2 for the same corpus and $AlgSize = 300$, we observe a reduction of 25.50% for the error rate as compared to *RefinedHMMSeg*.

Similarly to table 3, table 7 displays the results obtained by using the whole corpus for the training phases needed to compute *HMMSeg*, *RefinedHMMSeg* and estimate the scores for the fusion methods. These results are the maximum accuracies that the fusion methods can attain and can thus be regarded as the limit performance measurements of these fusion methods. The accuracies given in tables 5 and 6 are reasonably close to these performance limits when $AlgSize \geq 300$.

For instance, the maximum accuracies attained by *optimal fusion by soft supervision* are 95.72% and 95.53% for *FRcorpus* and *ENcorpus* respectively. For $(SizeComb, AlgSize) = (700, 700)$, the accuracies of the same method are 95.22% and 95.23% for *FRcorpus* and *ENcorpus* respectively.

The training database used to tune the fusion methods is different from that used to train the models for HMM segmentation and refinement by boundary-model. Of course, in practice, it is more appropriate to choose the same database so as to reduce the number of sentences to segment manually. This is possible without any significant performance loss. Table 8 shows the accuracies at 20 ms when the database is the same for the scoring and for training the models to produce *HMMSeg* and *RefinedHMMSeg*. To compute

these accuracies, we use the 4 fusion methods that are derived from the use of the total mark selection and the three types of supervision. For a training database of size 700, the accuracies attained by *optimal fusion by soft supervision* are 95.26% and 95.17% for *FRcorpus* and *ENcorpus* respectively. However, with hard supervision, we obtain accuracies equal to 94.39% and 94.36%. This means that in comparison with the uniform supervision, the number of segmentation errors is reduced by 15.5% and 14.3% respectively when we use soft supervision. Moreover, the computational loads of the two processings are the same. We can conclude that the estimation of the scores via the training phase is useful.

The results presented in this section show that *optimal fusion by soft supervision* significantly improves the accuracy at 20 ms in comparison with standard HMM segmentation. It is now interesting to see if *optimal fusion by soft supervision* is capable of removing most of the coarse errors. By coarse error, we mean a segmentation error larger than 50 ms. In this respect, table 9 presents, for different tolerances, the accuracies of standard HMM segmentation and the segmentation achieved by *optimal fusion by soft supervision* when the same database of size 300 is used for the scoring and the computation of *HMMSeg* and *RefinedHMMSeg*. From this table, we can observe that the number of coarse errors made by the standard HMM segmentation is reduced by a fifth via *optimal fusion by soft supervision*.

Table 5: Accuracies at 20 ms for *FRcorpus* when linear fusion is achieved with different score supervisions and mark selections

SizeComb	*AlgSize*	*Total selection*				*Selection by distance criterion*			
		uniform	*hard*	*soft*		*uniform*	*hard*	*soft*	
				$f(x)=x$	$f(x)=\frac{1}{1-x}$			$f(x)=x$	$f(x)=\frac{1}{1-x}$
100	*100*	93.67%	93.04%	94.20%	94.13%	93.13%	93.02%	93.16%	93.08%
	300	94.38%	93.81%	94.82%	94.75%	94.06%	93.99%	94.07%	94.02%
	700	94.58%	94.14%	94.97%	94.84%	94.32%	94.28%	94.33%	94.29%
300	*100*	93.68%	92.89%	94.23%	94.34%	94.14%	93.02%	93.15%	93.16%
	300	94.39%	93.77%	94.88%	94.98%	94.07%	94.01%	94.10%	94.14%
	700	94.58%	94.18%	95.07%	95.17%	94.32%	94.28%	94.35%	94.36%
700	*100*	93.66%	93.10%	94.22%	94.45%	93.12%	93.01%	93.14%	93.18%
	300	94.40%	93.88%	94.91%	95.10%	94.07%	94.00%	94.09%	94.15%
	700	94.58%	94.32%	95.08%	95.22%	94.33%	94.28%	94.34%	94.40%

Table 6: Accuracies at 20 ms for *ENcorpus* when linear fusion is achieved with different score supervisions and mark selections

SizeComb	*AlgSize*	*Total selection*				*Selection by distance criterion*			
		uniform	*hard*	*soft*		*uniform*	*hard*	*soft*	
				$f(x) = x$	$f(x) = \frac{1}{1-x}$			$f(x) = x$	$f(x) = \frac{1}{1-x}$
100	*100*	93.68%	93.02%	93.96%	93.98%	93.26%	93.21%	93.29%	93.15%
	300	94.36%	93.74%	94.69%	94.64%	94.11%	94.10%	94.13%	94.03%
	700	94.58%	94.10%	94.91%	94.97%	94.41%	94.41%	94.42%	94.36%
300	*100*	93.66%	93.08%	93.98%	94.17%	93.24%	93.18%	93.27%	93.24%
	300	94.37%	93.80%	94.70%	94.89%	94.12%	94.11%	94.13%	94.13%
	700	94.58%	94.25%	94.92%	95.14%	94.40%	94.40%	94.42%	94.43%
700	*100*	93.66%	93.21%	93.97%	94.25%	93.25%	93.19%	93.27%	93.33%
	300	94.37%	93.97%	94.69%	94.98%	94.11%	94.11%	94.14%	94.17%
	700	94.60%	94.23%	94.93%	95.23%	94.41%	94.41%	94.43%	94.46%

Table 7: The limit performance of the fusion methods with different score supervisions and mark selections

	Total selection				*Selection by distance criterion*			
	uniform	*hard*	*soft*		*uniform*	*hard*	*soft*	
			$f(x) = x$	$f(x) = \frac{1}{1-x}$			$f(x) = x$	$f(x) = \frac{1}{1-x}$
FRcorpus	94.86%	95.11%	95.39%	95.72%	94.75%	94.75%	94.77%	94.88%
ENcorpus	94.85%	94.70%	95.19%	95.53%	94.77%	94.77%	94.78%	94.82%

Table 8: Accuracies of the segmentation obtained by fusion by soft supervision when the same database is used for the scoring and the computation of *HMMSeg*, *RefinedHMMSeg*

		uniform	*hard*	*soft*	
				$f(x) = x$	$f(x) = \frac{1}{1-x}$
100	*FRcorpus*	93.68%	92.50%	94.08%	93.77%
	ENcorpus	93.67%	92.35%	93.92%	93.77%
300	*FRcorpus*	94.39%	93.83%	94.87%	94.92%
	ENcorpus	94.36%	93.10%	94.67%	94.77%
700	*FRcorpus*	94.59%	94.31%	95.09%	95.26%
	ENcorpus	94.58%	93.81%	94.93%	95.17%

Table 9: Accuracies, for different tolerances, of the standard HMM segmentation and the segmentation obtained by *optimal fusion by soft supervision* when the same database with size 300 is used for the scoring and the training of the models needed to create *HMMSeg*, *RefinedHMMSeg*

		10 ms	20 ms	50 ms	80 ms
300	*FRcorpus*	79.90%	94.92%	99.47%	99.90%
$(f(x) = \frac{1}{1-x})$	*ENcorpus*	81.71%	94.77%	99.43%	99.87%
HMM	*FRcorpus*	67.12%	88.53%	97.21%	98.92%
segmentation	*ENcorpus*	66.16%	87.77%	97.44%	99.43%

5.2 Subjective tests

In the previous section, *optimal fusion by soft supervision* turned out to be the most accurate method among those studied. In the present section, we want to assess this method in terms of speech quality. This can be achieved by means of objective or subjective tests. We focus our attention on subjective tests because they are often regarded as more reliable than objective ones. This is because subjective tests are based on direct ratings by human listeners and thus predict user satisfaction.

For synthesis systems, several subjective tests are available. The procedure of such tests is simple. Human subjects are asked to listen to speech signals and rate them according to the categories chosen for the subjective test.

The Mean Opinion Score (MOS) [12] is the most widely used subjective method. It uses an Absolute Category Rating (ACR) procedure because subjects are asked to rate the quality of several speech utterances without listening to the original signal. The scores given by the subjects must belong to $\{1, 2, 3, 4, 5\}$. These values refer to the categories shown in table 10. The MOS score is simply the mean of the scores collected from the listeners.

We apply the MOS test to assess the quality of French and English synthesized speech signals. The synthesis is performed by unit selection and performed by the *baratinoo* system developed by *France Telecom*. It needs a large database of segmented and labelled diphones. A diphone starts from the middle of one phoneme steady zone to the middle of the next phoneme steady zone. Because the middle of the steady zone of a phoneme can reasonably be approximated by the middle of this phoneme, the segmented diphones derive easily from the phonetic segmentation of the initial speech database.

It would be very interesting to carry out MOS tests so as to compare, in terms of synthesized speech quality, the segmentation obtained by *optimal fusion by soft supervision* versus the reference HMM segmentation, the manually produced reference segmentation and the *HMMSeg*, *RefinedHMMSeg* and *BrandtSeg* segmentations that are used for the fusion. In a forthcoming work, we plan to carry out this complete experiment. Here, we focus our attention on comparing the segmentation provided by *optimal fusion by soft supervision* with the two references (standard HMM segmentation and the manual segmentation).

Given the diphone segmentations derived from these 3 phonetic segmentations, three diphone acoustic dictionaries are built. Then, with the France telecom TTS system, these three dictionaries are used to synthesize a set of test sentences. This set of sentences needs to be carefully designed. Indeed, given the relatively low error rate provided by most segmentation algorithms, no segmentation problem should occur when synthesizing a given sentence. Therefore, we propose to carry out subjective tests on synthetic utterances containing diphones for which segmentation problems occured, i.e. whose distance to the corresponding manual mark exceeds 20 ms.

To achieve this, we first collected 2000 sentences from books of the "Gutenberg" project. The Gutenberg project was started in 1971 and consists of a large electronic library of nearly 17000 books that are free downloadable. Next, the 2000 sentences were synthesized via the diphone corpus derived from the HMM segmentation. We then computed the number of segmentation errors for each synthesized sentence and selected the 20 sentences with the largest numbers of errors. The synthesis of these 20 sentences gave us a first set. We considered the synthesized speech signals that contain the largest numbers of erroneous diphone marks due to HMM segmentation because *optimal fusion by soft supervision* can be expected to provide less erroneous diphone marks.

To select the text corpus, segmentation errors made on pauses and silences of diphones are not counted for two reasons. On the one hand, HMM segmentation performs poorly on silences and pauses. On the other hand, our purpose is to consider the largest variety of HMM segmentation errors. Thus, if we took into account the errors on silences and pauses, we might select sentences where most errors are due to silences.

The listeners were asked to mark these 60 synthesized sentences. Note that all the listeners are native speakers and naive and a phase of training with 5 sentences is done before the test. This training phase allows the

listeners to have a good idea about the quality of the synthetic voice in order to use the whole range of marks appropriately.

The results for the French and the English voices and each segmentation are given in table 11. Each value in the fourth column of this table represents the average mark of the synthesized voice quality calculated on the whole set of sentences and listeners.

Table 10: MOS and speech quality

Score	*Category*
5	Excellent
4	Good
3	Acceptable
2	Poor
1	Bad

Table 11: Results of the MOS test for the French and the English voices

		Number of subjects	*Score*	*Standard deviation*
French	HMM segmentation	16	2.86	0.41
	Soft fusion		3.15	0.37
	Manual segmentation		3.35	0.4
English	HMM segmentation	11	3.04	0.37
	Soft fusion		3.13	0.41
	Manual segmentation		3.06	0.44

The results given in table 11 for the French and the English voices show that the synthesized voice quality achieved by using a database segmented using *optimal fusion by soft supervision* is better than the quality obtained by using the standard HMM segmentation.

It also turns out that the synthetic voice quality achieved by using a database segmented with *optimal fusion by soft supervision* is closer to the quality obtained by using the manual segmentation for the French voice and outperforms this quality for the English voice.

A possible explanation is the following one. Actually, the so-called manual segmentation results from the manual correction of the segmentation errors made by the standard HMM algorithm. As such, it may still contain some

errors. This could also explain that the MOS for the English and French voices is quite poor (around 3) when the synthesis is performed with the manually segmented corpus.

6 Conclusion and extensions

In this paper, we have proposed a general approach capable of merging several segmentations produced by different automatic algorithms in order to obtain a more accurate segmentation than the combined segmentations. The idea of this approach was a result of the following fact. Segmentation algorithms do not perform equally on the same type of boundary.

To evaluate the performance of this approach, we have proposed to combining the segmentations performed by three methods: HMM segmentation, refinement by boundary-model and Brandt's GLR method. In this respect, we have proposed and tested several fusion methods.

From a more general point of view, combining several segmentations seems to be a good solution for segmenting large corpora, especially for TTS synthesis applications. The accuracy improves: for instance, *optimal fusion by soft supervision* reduces by 60% the number of errors made by the standard HMM segmentation.

Because the time duration required to perform *HMMSeg*, *RefinedHMMSeg* and *BrandtSeg* together is about twice that needed to achieve the standard HMM segmentation and since the fusion process is, moreover, negligible, the overall computational load of the proposed method is reasonable.

The fusion approach is flexible. In fact, the approach proposed in section 2 and summarized in figure 1 is a general framework and many other types of score, many other weighting functions and different criteria for the mark selection can be proposed. For example, we are currently studying the use of polynomials as weighting functions. In this paper, the algorithms are scored by their accuracies at 20 ms because a deviation of at most 20 ms is considered to be an acceptable upper limit for guaranteeing good quality of synthesized voice; however, the standard deviation of the segmentation error at 20 ms could also be a relevant type of score.

This approach can involve other segmentations in addition to or instead of those studied above. In order to obtain good performance measurements, we recommend the following: the segmentations that are to be combined should contain no insertion and no omission; the segmentation methods should per-

form differently depending on the type of transition classes so as to guarantee some complementarity. The fusion approach can, for example, apply to algorithms such as those presented in [1, 8, 14, 15]

In order to attain better accuracy, attention should be given to the types of boundary that still cause many segmentation errors so as to develop some processing dedicated to them. We highlight such types of boundary as follows.

Given a pair of phonetic classes and thus a type of boundary, we compute the number of segmentation errors at a tolerance of 20 ms; we also compute the ratio between this number of errors and the total number of boundaries of this type. The results are presented in tables 12 and 13. The former concerns *FRcorpus* and the latter corresponds to *ENcorpus*. These results were obtained on the basis of the segmentation achieved by *optimal fusion by soft supervision* when $(SizeComb, AlgSize) = (300, 300)$. From these figures, we note that most errors are made in detecting a transition between a phonetic class and the classes SIL (silence) and SP (pause).

To show the reader that decreasing the number of errors for these pairs of classes is important, we compute the accuracy at 20 ms by using manual segmentation to correct all the errors between any class and SIL. We attain an accuracy of 95.70% at 20 ms for *FRcorpus* and of 95.47% for *ENcorpus*. By removing all the errors between any class and SP, we arrive at 96.20% for *FRcorpus* and 95.73% for *ENcorpus*.

In terms of accuracy, these values clearly suggest using very accurate speech/silence detection. In terms of speech synthetic quality, it would be relevant to evaluate the gain provided by such speech/silence detection.

Another focus of interest would be to evaluate *optimal fusion by soft supervision* when the phonetic transcription contains errors.

It would also be interesting to assess *optimal fusion by soft supervision* for the segmentation of large corpora dedicated to other applications such as speech recognition. This would make it possible to verify the robustness of the fusion approach to uncontrolled or variable recording conditions and to noisy signals.

References

[1] J. Adell and A. Bonafonte. Towards phone segmentation for concatenative speech synthesis. *in Proceedings of the 5th ISCA Workshop on*

Speech Synthesis, November 1983.

[2] R. André-Obrecht. A new statistical approach for the automatic segmentation of continous speech signals. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 36:29–40, January 1988.

[3] A.V. Brandt. Detecting and estimating parameters jumps using ladder algorithms and likelihood ratio test. *Proc. ICASSP,Boston, MA*, November 1983.

[4] D. Falavigna F. Brugnara and M. Omologo. Automatic segmentation and labelling of speech based on hidden markov models. *Speech Communications*, 12:357–370, 1993.

[5] D. Tihelka J. Matousek and J. Psutka. Automatic segmentation for czech concatenative speech synthesis using statistical approach with boundary-specific correction. *EUROSPEECH, Geneva*, 2003.

[6] S. Jarifi, D. Pastor, and O. Rosec. Brandt's GLR method & refined HMM segmentation for TTS synthesis application. *13th European Signal Processing Conference (EUSIPCO)*, 2005.

[7] S. Jarifi, D. Pastor, and O. Rosec. Cooperation between global and local methods for the automatic segmentation of speech synthesis corpora. *ICSLP*, 2006.

[8] Y.J. Kim and A. Conkie. Automatic segmentation combining an HMM-based approach and spectral boundary correction. *ICSLP*, September 2002.

[9] L.Wang, Y. Zhao, M. Chu, J. Zhou, and Z. Cao. Refining Segmental Boundaries for TTS Database Using Fine Contextual-Dependent Boundary Models. *ICASSP*, I:641–644, 2004.

[10] S. Nefti. *Segmentation automatique de la parole en phones. Correction d'étiquetage par l'introduction de mesures de confiance.* PhD thesis, Université de Rennes I, 2004.

[11] J. J. Odell. *The use of context in large vocabulary speech recognition.* PhD thesis, The university of Cambridge, 1995.

[12] ITU-T Recommendation P.800.1. Mean opinion score (MOS) terminology. 2003.

[13] S. Pigeon. *Authentification multimodale d'identité*. PhD thesis, Université Catholique de Louvain, 1999.

[14] D. Torre Toledano, M.A. Rodríguez Crespo, and J.G. Escalada Sardina. Trying to mimic human segmentation of speech using HMM and fuzzy logic post-correction rules. *Third ESCA/COSCOSDA International Workshop on Speech Synthesis, Australia*, pages 26–29, November 1998.

[15] D.T. Toledano, L.A. Hernàndez Gòmez, and L. Villarubia Grande. Automatic phonetic segmentation. *IEEE Transactions on Speech and Audio processing*, 11:617–625, November 2003.

[16] P. K. Varshney. *Distributed detection and data fusion*. Springer-Verlag, 1997.

[17] S. Young, G. Evermann, T. Hain, D. Kershaw, G. Moore, and J. Odell. The HTK Book for HTK V 3.2.1. Cambridge, UK, 2002.

Table 12: For every given pair of French phonetic classes, the number of segmentation errors at tolerance 20 ms and error rate conditionally to this pair of classes. The error rate is defined as the ratio between the number of segmentation errors at tolerance 20 ms and the total number of boundaries available for the pair of classes under consideration. The French phonetic classes are: oral vowels (OV), nasal vowels (NV), unvoiced plosives (UVP), voiced plosives (VP), unvoiced fricatives (UVF), voiced fricatives (VF), diphthongs (DIPH), nasal consonants (NC), liquid consonants (LC), semivowels (SV), pauses (SP) and silences (SIL). A class in the first column represents the phonetic class of the phoneme located to the left of a boundary and a class in the first line represents the class of the phoneme which is at the right of this boundary. For instance, if we consider the pair (OV, UVP) of phonetic classes, there were 676 errors in detecting the transitions between these two classes when the phonemes to the left are oral vowels and the phonemes to the right are voiceless stops. The error rate for this pair of classes equals 4.10%: this means that 4.10% of the boundaries of this type are erroneously detected. In this table, and in the subsequent one, we emphasize the pairs of classes with large numbers of errors and large error rates. $---$ means that no transition between the pair of classes is available in the corpus.

	OV	*NV*	*DIPH*	*VP*	*UVP*	*VF*	*UVF*	*NC*	*LC*	*SV*	*SP*	*SIL*
OV	639/23.9	102/13.8	12/32.4	58/0.79	676/4.10	70/0.83	34/0.33	35/0.41	1277/6.32	137/16.5	416/29.1	653/29.8
NV	96/15.6	76/42.7	6/25.0	172/6.14	183/3.96	3/0.31	2/0.07	29/2.50	44/5.19	5/29.4	111/14.8	278/26.1
DIPH	17/35.4	2/10.0	1/100	0/0.00	65/26.0	4/3.92	0/0.00	1/2.17	23/25.0	17/44.7	34/64.15	32/46.3
VP	135/1.41	22/1.97	14/10.3	31/28.9	38/39.1	10/5.32	13/12.2	6/5.26	121/6.36	51/10.9	69/82.1	118/75.1
UVP	408/2.60	5/0.18	11/5.58	54/15.34	66/7.50	20/19.2	47/4.97	18/5.64	155/2.88	7/1.20	94/18.8	154/25.3
VF	80/1.11	5/0.31	2/3.70	3/1.38	30/20.0	4/7.14	12/12.90	1/0.39	9/2.33	52/5.86	61/24.90	68/21.6
UVF	2/0.02	0/0.00	0/0.00	14/4.59	85/4.19	7/12.96	43/15.8	12/6.38	34/5.14	17/0.84	51/14.4	51/11.3
NC	201/2.48	62/3.37	14/34.1	43/10.1	40/10.9	5/4.27	4/1.42	38/20.1	21/11.4	141/13.8	50/18.1	82/27.3
LC	482/2.57	52/2.35	68/26.3	38/2.45	63/3.22	15/2.05	22/1.41	43/2.94	112/11.5	80/8.86	237/24.8	401/26.5
SV	785/18.1	284/14.1	2/100	6/8.33	7/14.8	0/0.00	2/5.00	4/10.0	8/21.6	3/100	14/24.6	25/20.5
SP	1/0.07	1/0.29	0/0.00	20/4.59	378/51.4	5/1.77	17/2.64	0/0.00	2/0.40	0/0.00	$---$	$---$
SIL	3/0.17	3/0.60	0/0.00	15/3.93	283/43.5	4/0.93	26/2.69	3/0.73	7/0.42	1/12.5	$---$	$---$

Table 13: For every given pair of English phonetic classes, the number of segmentation errors at tolerance 20 ms and error rate conditionally to this pair of classes. The English phonetic classes are: vowels (V), voiced plosives (VP), unvoiced plosives (UVP), voiced fricatives (VF), unvoiced fricatives (UVF), affricates (AF), nasal consonants (NC), liquid consonants (LC), semivowels (SV), pauses (SP) and silences (SIL). This table reads like table 12.

	V	*VP*	*UVP*	*VF*	*UVF*	*AF*	*NC*	*LC*	*SV*	*SP*	*SIL*
V	1325/20.7	116/0.81	366/1.54	99/0.56	215/1.15	58/2.16	319/1.06	2137/16.1	240/7.75	288/25.6	544/26.6
VP	101/0.74	124/18.5	86/12.9	111/8.23	81/4.75	31/23.8	29/3.30	40/1.20	73/6.71	71/19.1	142/17.1
UVP	109/0.48	186/23.4	433/20.2	244/23.9	255/5.89	70/41.1	38/4.74	69/1.34	264/13.6	111/20.6	331/23.8
VF	167/1.04	92/7.41	85/5.64	90/5.32	243/10.9	17/11.9	58/3.26	9/0.86	60/6.83	83/18.5	144/14.6
UVF	682/3.25	56/11.9	408/7.13	42/9.61	117/12.2	17/12.1	45/2.26	21/0.96	197/23.4	100/25.5	174/19.2
AF	23/0.80	12/7.69	14/5.88	5/6.10	46/22.6	7/23.3	11/13.4	1/1.09	11/11.7	14/25.0	25/24.04
NC	397/2.76	85/1.79	66/1.27	34/1.07	32/0.84	3/0.60	97/12.4	112/10.2	137/10.3	160/24.2	472/29.1
LC	1728/8.15	35/2.99	30/3.73	18/2.49	14/1.33	8/8.79	38/13.1	53/24.7	52/14.0	89/33.8	168/32.5
SV	1232/11.3	– – –	– – –	– – –	– – –	– – –	0/0.00	4/44.4	– – –	– – –	– – –
SP	46/3.22	148/39.9	46/16.6	66/26.1	63/8.39	0/0.00	15/6.76	21/14.3	24/6.67	– – –	– – –
SIL	18/0.71	183/21.8	86/20.8	138/8.48	128/9.36	3/3.45	23/5.03	13/8.72	32/3.59	– – –	– – –

Ecole nationale supérieure des Télécommunications de Bretagne – Brest
Dépôt légal : 4ème trimestre 2006
ISSN : 1255-2275