Keyword spotting for multimedia document indexing

Philippe Gelin[a] and Chris. J. Wellekens[b]

[a]Speech Technology laboratory, Suite #202, 3888 State Street,
Santa Barbara, CA 93105, USA.

[b]Institut Eurécom, Department of Multimedia Communications,
2229 route des Crêtes, BP 193, F-06904 Sophia Antipolis, France

ABSTRACT

We tackle the problem of multimedia indexing using keyword spotting on the spoken part of the data. Word spotting systems for indexing have to meet very hard specifications: short response times to queries, speaker independent mode, open vocabulary in order to be able to track any keyword. To meet these constraints keyword models should be build according to their phonetic spelling and the process should be divided in two parts: preprocessing of the speech signal (off-line) and query over a lattice of hypotheses (on-line). Different classification criteria have been studied for hypothesis generation: frame labelling, maximum likelihood and maximum a posteriori (MAP). The hypothesis probability is computed either through standard gaussian model or through an hybrid Hidden Markov Model-Neural Network (HMM-NN). The training of the phonemic models is based either on Viterbi alignment or on Recursive Estimation and Maximization of A posteriori Probabilities (REMAP). In the latter discriminant properties between phonemes are enforced. Tests have been conducted on TIMIT database as well as on TV news soundtracks. Interesting results have been obtain in time saving for the documentalist. The ultimate goal is to couple the soundtrack indexing with tools for video indexing in order to enhanced the robustness of the system.

Keywords: keyword spotting, indexing, speech, information retrieval, REMAP, Hidden Markov Chain, frame labelling, speech indexing.

1. INTRODUCTION

The amount of accessible multimedia information has been growing drastically. The increasing number of personal computers enables the layman to contribute to the development of internet, this new worldwide source of information. At the same time, there is an increasing number of cables and satellites operators providing new video services. Access to this growing amount of information is not easy and indexing tools are needed not only by database managers and professionals involved in archiving but also by private users.

Although text indexing has been widely used for several decades, content based indexing of other media (still images, video, music, speech) is still in its infancy. The search for semantically described events may rely on the video content itself (face recognition, scene understanding) but also on the soundtrack. Few works[10,11,12] on this topic have been reported so far. Among the useful indices that can be extracted from the soundtrack, localization of keywords plays a prominent role.

To be of general use, the word spotter should be speaker independent and able to detect any word of an open vocabulary. Due to the latter constraint, phonemics description of words is unavoidable. Most of the existing keyword spotters[7,13] have been proposed for specific applications such as phone dialling or vocal messages sorting. While most of them achieve speaker independency, none of them deals with the open vocabulary problem. The description of speech data in terms of phonemes has also been used[2].

This paper describe three indexing tools we developed which satisfy these constraints.

Given a series of acoustic segments, the first tool computes the probabilities to associate their phoneme utterance with given phonemes. From this information, the tool identifies signal locations where the probability of presence is high and uses these "phonetic hypotheses" to build a lattice that will be saved and used for query processing. When searching for a word, the lattice is scanned to find the corresponding phonetic transcription. In this manner, indexing task is separated from the query and is achieved off-line in a preliminary sophisticated and accurate

processing while the query can be quickened.
The second indexing tool uses the same strategy of task separation but uses Markov models of the phonemes and the language. It is observed that this method accelerates the query and in addition increases the scores.
The third tool relies on a recently described theory of discriminative training[9] where Hidden Markov Models and neural networks are blended to efficiently train the a posteriori probabilities of phonemes given an utterance. The aim is to increase the reliability of the phonetic lattice and increase the scores of the word spotter.

In section 2, language models for the three different approaches are described. For each of them, the underlying phonetic models are given and if necessary, we will expand the model training algorithm.

Section 3 deals in more details with the lattice of hypotheses generation. We will show how this lattice generation is conducted according to the approach used. For the labelling method, integration over time of local probabilities is used in order to smooth these curves. Moreover, a multi-level of hypotheses detection is used to find various phonemes transitions. In the HMM method, we will describe the algorithm used to detect hypotheses. Finally, we will fit this last algorithm for the REMAP approach.

Section 4 deals with the search algorithm over the lattice and described the blocking effect that may occur during the parsing. Next, given the used approach, we will detail differences solution to avoid this effect.

In section 5, comparative results between our indexing tools are given.

Conclusions are drawn in section 6 and perspectives for future work are proposed.

2. LANGUAGE MODEL

As in all recognition systems, speech is preprocessed over short time frames (32 msec here), shifted by 10 msec. Each frame is described by a vector, x_t, in a so-called feature space $\Re^n$. This vector is composed with the 17 first cepstrum coefficients and a voiceness estimator based on cepstrum coefficients.
As it is well known[5], a voiced frame can be easily detected on a cepstrum analysis since the pitch-periodical nature of the spectrum corresponds to a peak in the cepstral domain. Knowing that fundamental frequency ranges from 40Hz to 250Hz, we can isolate the cepstrum part of this range to detect the highest energy frequency. The ratio between this energy and the average energy over this frequency domain gives us a measurement of voiceness.

2.1 Frame Labelling

Traditionally, phonemes are modelled upon Hidden Markov Models (HMM).

But in this first approach, a single local probability distribution is associated with each phoneme which is just equivalent to work with a one-state model. This probability measures the degree of membership of an acoustic vector to the corresponding phonemic class. But since no Viterbi alignment is used in our word spotter, HMM concept is in fact irrelevant. Three different definitions of the probabilities will be tested. Gaussian distributions and multi-Gaussian distributions will model $P(x_t|\varphi)$ (where φ denotes the current phoneme and x_t the feature vector) while the a posteriori probability $P(\varphi|x_t)$ is generated by a multi-layer perceptron (MLP). Bayes relation links these two probabilities. While the first two parametric distribution families have been used for classification with HMM for a long time, neural network generation of probabilities has drawn recently a lot of interest[6]. It is out of the scope of this paper to describe in the details the training techniques for the determination of the distribution parameters[4]. The MLP is trained with the error backpropagation algorithm.

- *Gaussian distribution.*

The parameters for each phoneme are a mean vector and a covariance matrix which is assumed to be diagonal for simplification purpose. The phoneme segmentation used to estimate theses parameters results from the optimal alignment of the labelled database by a standard Viterbi algorithm[5].

- *Multi-Gaussian distribution.*

It seems to be a severe constraint to restrict the distribution shape to a unimodal Gaussian hyper-surface. A multimodal Gaussian distribution could fit better with the actual underlaying distribution of speech data and would lead to more accuracy. The determination of multi-gaussian parameters has been achieved along two chained techniques. In the beginning, a fast iterative processing is used in which each vector is associated with the distribution giving the highest probability. The results are then fine tuned by using the exact processing in which each vector is associated with all distributions according to its probabilities. The order of the distribution has been chosen according to the size of the class. If the number of patterns associated with one gaussian distribution falls under a predetermined threshold (typically 30), the corresponding mode is skipped.

- *Neural Network.*

It is now well accepted that the outputs of an MLP trained with a classification criterion (one active output only with all the others fired off) approximate the a posteriori probabilities. Using Bayes rule and the a priori probabilities of the classes, a posteriori probabilities can be converted into local probabilities within an irrelevant scaling factor[6]. We used an MLP with a single hidden layer containing 20 units in a first test and 200 units in a second one. No distribution shape constrains the resulting distribution and the a posteriori probabilities are trained according to a discriminant criterion. The learning algorithm is a standard error backpropagation with a cross-validation test for iteration control to avoid over training.

2.2 Hidden Markov Model

In this second approach, the language model is based on a HMM structure composed by sub-models of phonemes, $\varphi_i \in \Phi$, connected in order to generate any possible phonetic sequence. The inter-phoneme connections are the time-shift invariant phoneme transition probabilities $P(\varphi_{i,t+1}|\varphi_{j,t})$. These probabilities are based on the number of phonetic transition, $N(\varphi_{j,t}, \varphi_{i,t+1})$, found in the training database:

$$P(\varphi_{i,t+1}|\varphi_{j,t}) = \frac{N(\varphi_{j,t}, \varphi_{i,t+1})}{\sum_{v} N(\varphi_{j,t}, \varphi_{v,t+1})}$$

Each phoneme, $\varphi_p \in \Phi$, is modelled by a standard 3 states HMM, where the states are denoted: $q_{3p}, q_{3p+1}, q_{3p+2}$. Each state may generate the local a priori probability $P(x_t|q)$, for an acoustic vector x_t, to be produced by a given state q. Theses models are estimated through a standard training iterative process[4].

Two different state probabilities are tested: Gaussian density based model giving $P_G(x_t|q)$ and a multi layer perceptron (MLP) based model giving an a posteriori probability, $P_{MLP}(q|x_t)$. In the latter case, Bayes rule is used to deduce an a priori probability.

- *Gaussian Distribution*

Here, the parameters for each phoneme consist in a mean vector and a covariance matrix which is assumed to be diagonal for simplification purpose[4].

- *Multi Layer Perceptron*

As in the first approach, each output of the neural network is associated with a specific phoneme.

Using Bayes rule and the a priori probabilities of the classes, a posteriori probabilities can be converted into local probabilities within an irrelevant scaling factor in the Forward Backward or Viterbi algorithms[6].

2.3 REMAP model

- *Overview*

The a posteriori probability approach can be viewed as follows:
given a spoken sentence to be learned, let denote M, the HMM model to be associated with, q_i the states of this model, each of them representing a specific phoneme φ_i and $X = X_1^N = \{x_1, x_2, \ldots, x_N\}$ the sequence of acoustic vectors extracted from this sentence.
To learn the model, we tend to maximize the a posteriori probability, $P(M|X, \Theta, L)$, where Θ and L, respectively represent the parameter set of the acoustic model and language model.

The a posteriori probability, $P(M|X, \Theta, L)$ can be written as the sum associated with the valid paths in the model:

$$P(M|X,\Theta,L) = \sum_{\gamma_j \in \Gamma} P(\gamma_j|X,\Theta,L)\,P(M|\gamma_j, X,\Theta,L)$$

where Γ ,is the set of all valid paths in M.

This representation better matches with the Baum Welch approach than the Viterbi one[9].
The first factor of the right hand side denotes the acoustic model and the second factor denotes the language model

- *Acoustic model.*

If we denote $q_{j,n}$ the state visited at time n by the path γ_j, we can write the acoustic model as follows:

$$P(\gamma_j|X,L,\Theta) = \prod_{n=1}^{N} P\left(q_{j,n}|X_{n-c}^{n+d}, q_{j,n-1}, \Theta\right).$$

where we make the successive hypotheses:

- The acoustic model is independent of the language parameters, L.
- We use a first order Markov model.
- This probability is only dependant of a temporal window of length $c+d+1$ of acoustic coefficients, X_{n-c}^{n+d}.

Note that these local probabilities can easily be evaluated with a MLP[9].

- *Language model.*

The language model can be describe by:

$$P(M|\gamma_j, X,\Theta,L) = \left(\prod_{n=1}^{N} \frac{P(q_{j,n}|q_{j,n-1}, M, L)}{P(q_{j,n}|q_{j,n-1}, L)}\right) P(M|L$$

According to the successive assumptions that, knowing the path γ_j, i.e. the phonetic sequence:

- The model can be found without an explicit dependence on X.
- The language model is independent of the acoustic parameters, Θ.
- A first order Markov model is used.

- *Training algorithm.*

In a similar way, the REMAP approach is based on a successive MLP training scheme. But unlike the classical HMM-NN learning method, no Viterbi algorithm and no segmentation are used.
In this iterative scheme, the MLP trained in the previous iteration is used to estimate transition probabilities from which new targets will be derived. These one are then used for the next MLP training. Convergence of this iterative process has been proved[9].

- *Phoneme transition probabilities.*

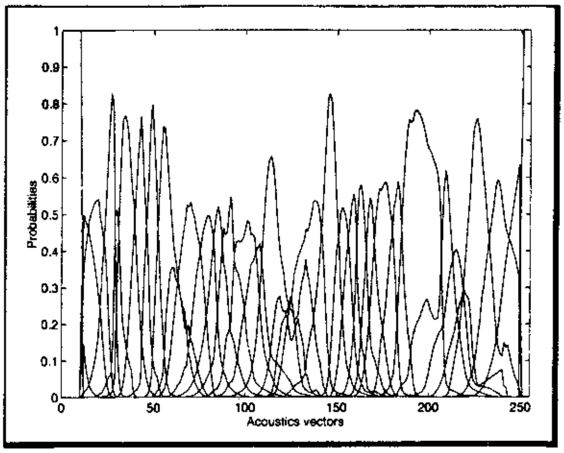

FIGURE 1.

In fig. *(1)*, the local phonemes probabilities estimated by (1) are plotted along the time axe. It can be easily noticed that the transitions between most probable phoneme are smooth and let the system take less abrupt decision. This will lead to a more flexible recognition system than an standard HMM-NN approach.

3. LATTICE GENERATION.

3.1 Frame Labelling

- *Integration over time.*

Plotting the a posteriori probability of a given phoneme $P(\varphi | x_t)$ as function of time shows segments of higher probability (cf. fig. *(2)* (a)). However, the curve fluctuates between successive frames and should be smoothed. Integration over time by summing the values in a rectangular window will achieve low pass filtering. Cut off frequency will be controlled by the average duration of the current phoneme: indeed, the width of the rectangular window should be adjusted to the resolution required for a satisfactory analysis of the current phoneme. Doing so, we can reduce the importance of erratic peaks inside long duration phonemes and still detect peaks for the short duration phonemes. Smoothed curves will ease the determination of hypotheses on phoneme boundaries (cf. fig. *(2)* (b)).

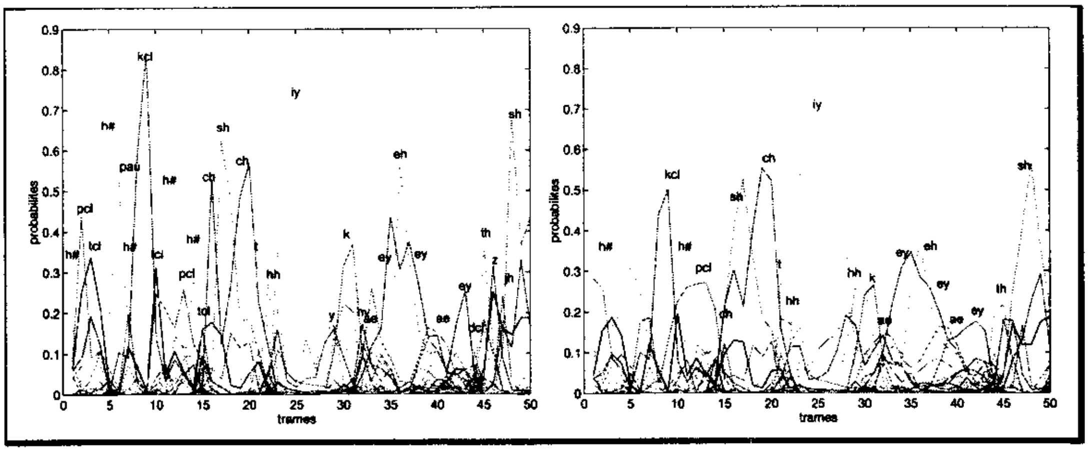

fig. 2:

- *Multi-level hypotheses on phoneme occurrences.*

To generate hypotheses, we define different thresholds of probability (typically 0.1, 0.01, 0.001). For a given threshold and for each phoneme φ, we detect segments where the curve $P(\varphi|x_t)$ runs above the threshold. Such a segment is denoted $X_b^e = [x_b, ..., x_e]$. The probability over this segment is $P(\varphi|X_b^e)$. Using the Bayes rule and the usual assumption that each acoustic vectors x_i independent, we can write[8]:

$$P(\varphi|X_b^e) = \frac{\prod_{i=b}^{e} P(\varphi|x_i)}{P(\varphi)^{e-b}},$$

making it independent of the segment duration. This will be abbreviated by P when no confusion is possible. Thus, an hypothesis consists of this probability and the beginning and ending frame indexes of the current phoneme. It is denoted as $h(\varphi, P, b, e)$.

Different thresholds generate different boundaries and probabilities. Lower thresholds give rise to new hypotheses containing less likely phonemes which do not appear with higher thresholds. When using all possible φ we generate the lattice of hypotheses, denoted $L = \{h_1, ..., h_M\}$, where we assume all the hypotheses sorted according to their beginning frame number, b. The number of hypotheses, M, can be modified by decreasing or increasing the thresholds

3.2 Hidden Markov Models

The a priori probability that the vector sequence X_1^T can be associated with a specific path, › $= \{q_\wp(1), q_\wp(2), ..., q_\wp(T)$ through the different states of the HMM is given by:

$$\wp = \prod_{t=1}^{T} P(x_t|q_\wp(t))P(q_\wp(t+1)|q_\wp(t).$$

Each sequence of states generated by a path through the HMM can also be viewed as a sequence of sub-paths through phoneme models, $\wp = \{\wp_{\varphi_1}, \wp_{\varphi_2}, ..., \wp_{\varphi_V}\}$, where V phonemes have been generated and $\partial_{\varphi_v} = \{q\left(\varphi_v, b_v\right), ..., q(\varphi_v, e_v)\}$, where $q(\varphi, t) = q_{3\varphi}$ or $q_{3\varphi+1}$ or $q_{3\varphi+2}$.
Then we can write:

$$P_\wp = \prod_{v=1}^{V} P\left(X_{b_v}^{e_v}\middle|\varphi_v\right)P(\varphi_{v+1}|\varphi_v),$$

where

$$\left(X_{b_v}^{e_v}\middle|\varphi_v\right) = \prod_{t=b_v}^{e_v} P(x_t|q(\varphi_v, t))P(q(\varphi_v, t+1)|q(\varphi_v, t).$$

At each time t, during the forward part of the Viterbi process, the probability associated with the best path finishing in each state is known.
It has been shown[1] that not all possible backtrack informations collected during the forward process should be saved.

- *Forward*

In a Viterbi approach, we only need to keep at each time t, the best finishing phoneme $\varphi(t)$ and its duration $\delta t(t)$. In a lattice making approach, as in the N-best approach, we need to keep more information. At each time t, we need to keep the N best finishing phonemes, $›_1(t), \varphi_2(t), ..., \varphi_N(t$, their respective duration, $t_1(t), \delta t_2(t), ..., \delta t_N(t$, and the probabilities associated with them $'_1(t), P_2(t), ..., P_N(t$, where $P_i(t)$ stands for $'(X_{t-\delta t_i(t)}^t|\varphi_i(t)$.

These associations can already be viewed as hypotheses.

- *Backward*

In order to be time efficient during the lattice search process, we cannot keep trace of all the exact N-best paths, as each N-best path has its own phoneme segmentation, and therefore would need to generate too much nodes. In order to limit the node generation, a two step process is used as shown in fig. *(3)*.

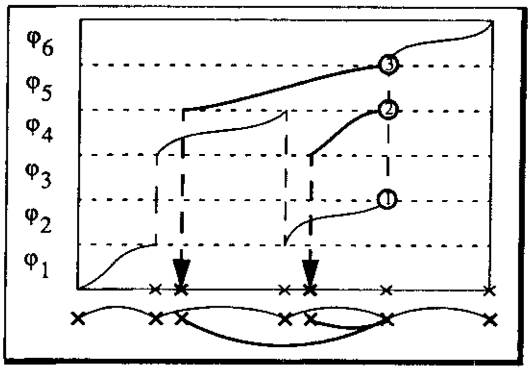

fig. 3.

First, the node generated by phonemes transitions of the best path (Viterbi) is kept. Second, for each selected node, we inspect the N finishing phonemes associated with, and select their beginning nodes. Third, we keep in the lattice hypotheses containing all the phonemes beginning and ending on selected nodes.
Doing so, we define *Groups of Hypotheses*, containing the same beginning and ending nodes.

3.3 REMAP approach

A similar method than in the HMM approach is use to extract this lattice out of the acoustical vector sequence.

For each segment between two consecutive nodes of the lattices the N best phonemes, φ_k are considered. The probability of φ_k over this segment is computed by:

$$(Q_k(b,e)|X) = \left(\sum_{l \neq k} P\left(q_k^b \middle| X_{b-c}^{b+d}, q_l^{b-1}\right)\right)\left(\prod_{t=b}^{e} P\left(q_k^{t+1} \middle| X_{t-c}^{t+d}, q_k^t\right)\right)\left(\sum_{l \neq k} P\left(q_k^e \middle| X_{e-c}^{e+d}, q_l^{e-1}\right.\right.$$

where $Q_k(b,e)$ stands for $\{q_l^{b-1}, q_k^b, \ldots, q_k^e, q_m^{e+1}\}$ with $l, m \neq k$.

4. SEARCH OVER THE LATTICE.

4.1 Confusion Matrix

Since hypotheses are obtained through a thresholding process or an N-Best like strategy, not all phonemic hypotheses appear in the lattice as opposed to a continuous HMM word recognizer where all hypotheses are considered every frame, even those with extremely low probabilities. Thus a blocking effect may result in the search strategy in case of mispronunciation, non-standard pronunciation or even simply due to the high variability of speech.

To alleviate this blocking effect, we used a so called confusion matrix, containing the estimated probability of confusion, $P(\varphi_p|\varphi_d)$, between pronounced phoneme, φ_p, and detected phoneme, φ_d. This probability known, we can associate with each detected phoneme, φ_d, a probability of having pronounced a specific phoneme, φ_p The estimation of this matrix relies on the specific approach used.

- *Frame Labelling*

In this approach, we estimated this matrix by:

$$'(\varphi_p|\varphi_d) = \sum_{x \in X} P(x|\varphi_d)P(\varphi_p|x),$$

where $P(x|\varphi_d)$ is given by the appropriate phoneme modelling (gaussian, multi-gaussian or NN), $P(\varphi_p|x)$ is 0 or 1 according to the given database segmentation and X the entire training set.

- *Hidden Markov Models*

Let us note X_p, the set of all acoustic vectors labelled by the correct phoneme φ_p. If we note φ_d the detected phoneme, we can then compute the confusion probability:

$$(\varphi_p|\varphi_d) = \frac{[P(q_{3d}|\varphi_p) + P(q_{3d+1}|\varphi_p) + P(q_{3d+2}|\varphi_p)]}{P(q_{3d}) + P(q_{3d+1}) + P(q_{3d+2})} P(\varphi_l$$

where q_{3d}, q_{3d+1} and q_{3d+2} are the states used during the training with the generic language model, and φ_p when using the Viterbi segmentation on the known phoneme sequence.

- *REMAP*

Given a standard HMM of the language, composed with the states q_d, associated with the detected phonemes φ_d, we can compute[14] the confusion probability:

$$P(\varphi_p|\varphi_d) = \sum_{t=1}^{N} \sum_{k=1}^{K} \frac{P\left(\varphi_p^t \middle| \varphi_k^{t-1}, X\right) P\left(\varphi_k^{t-1} \middle| X_{t-c}^{t+d}\right)}{P\left(\varphi_d^t\right) N_x} P\left(\varphi_p^t \middle| \varphi_d^t, \varphi_k^{t-1}, X_{t-c}^{t+d}\right) \quad ,$$

where:

- $P\left(\varphi_p^t \middle| \varphi_k^{t-1}, X\right)$ is the target for the Neural Network training,
- $P\left(\varphi_k^{t-1} \middle| X_{t-c}^{t+d}\right)$ is the a priori probability,
- $P\left(\varphi_d^t \middle| \varphi_k^{t-1}, X_{t-c}^{t+d}\right)$is given by the Neural Net according to the input vector$\left(\varphi_k^{t-1}, X_{t-c}^{t+d}\right)$
- $P\left(\varphi_d^t\right)$ is supposed constant and estimated through the neural network.

4.2 Algorithm.

The frame labelling algorithm[8] differs from the HMM and the REMAP one only by the mechanism used to select the next phoneme hypotheses.

- *Frame Labelling*

In this case, hypotheses are deduced from the local probabilities. Therefore the hypotheses boundaries cannot be grouped as in HMM or REMAP approach. So, the end of an hypothesis does not necessarily match exactly with the beginning of the next hypothesis. Between successive hypotheses temporal jumps are then required.

Lets select two hypotheses, $h_1(\varphi_1, P_1, b_1, e_1)$ and $h_2(\varphi_2, P_2, b_2, e_2)$. We allow the transition between the two hypotheses if:

$$b_2 - e_1| < \alpha \frac{|\mu(\varphi_1) + \mu(\varphi_2)|}{2},$$

where α is a given threshold and $\mu(\varphi)$ is the mean duration of the phoneme φ.

- *Hidden Markov Model and REMAP*

Now, as boundaries hypotheses are grouped according to the Viterbi segmentation and its second iteration (see 3.2), jumps are no longer needed. This reduces searching time as hypotheses transitions boundaries are fixed.

Let $\phi = \{\varphi_1, \ldots, \varphi_N\}$ be the phonetic transcription of the searched keyword.
For each *group of hypotheses* having the same boundaries, $h(\varphi_h, P_h, b, e) \in H_b^e$, where $P_h = P(\varphi_h|X_b^e)$, we compute, using the confusion matrix:

$$P\left(\varphi_j|X_b^e\right) = \sum_{i=1}^{\#H_b^e} P_{h_i} P_{conf}\left(\varphi_j|\varphi_{h_i}\right), \forall j = 1, \dots, N$$

generating a new lattice, L, of M new hypotheses, specific to the keyword.
Next, we search the best sequence of hypotheses denoted $H = \{h_{l_1}, \dots, h_{l_N}\}$, which maximizes the probability:

$$P(H) = \prod_{i \in [l_1, \dots, l_N]} P(h_i), \text{ where } l_n \in [1, M],$$

and such that if $h_{l_i} \in H_{b_i}^{e_i}$ $\forall i = 1, \dots, N$, then $e_i = b_{i+1}$ $\forall i = 1, \dots, N-1$.
The search of the optimal sequence of hypotheses is based on a recursive process.

The initialization process consists in searching over all the lattice L, each hypothesis $h_{l_1}(\varphi_1, P, s, t)$, $l_1 \in [1, \dots, (M-N+1)]$ of occurrence of the first phoneme φ_1.

1. For each occurrence of the first phoneme, initialize $H^1 = \{h_{l_1}\}$ and $P(H^1) = P(h_i)$,

2. In each step $k = 2, \dots, N$, for the last hypothesis of H^{k-1}, denoted $h_{l_{k-1}}(\varphi_{k-1}, P, s, t)$, we next search for hypotheses $h_{l_k}(\varphi_k, p', s', t')$ of occurrence of φ_k, such that $t = s'$.

3. For each hypothesis h_{l_k} found, we build $H^k = \{H^{k-1}, h_{l_k}\}$, and calculate $P(H^k) = P(H^{k-1})P(h_i)$.
   If no hypothesis is found, let $k = k-1$ and go to 2.

4. if $k < N$, set $k = k+1$,and go to 2.

5. if $k = N$ and if $P(H^N)$ is the maximum sequence probability encountered, we keep this sequence H^N.

At the end of this process which runs over the whole lattice, the sequence ϕ showing the maximum probability, $P(\phi|L) = \max_L \left[P(H^N)\right]$ is found.

5. RESULTS.

The tests we have done are based on the DARPA TIMIT corpus (90). As shown in table *(1)*, we randomly chose 20 SX sentences of the test part and in each of them a keyword, ϕ_k was selected.

sx113 muscular	sx95 alligators	sx14 thursday
sx10 grades	sx100 proceeding	sx101 decorate
sx110 problems	sx20 overalls	sx199 exposure
sx103 ambulance	sx290 informative	sx99 society
sx137 tradition	sx109 ankle	sx102 kidnappers
sx53 vocabulary	sx373 superb	sx280 mirage
sx133 pizzerias	sx8 silly	

Table 1:

Each SX sentence occurred 7 times in the test database as there are spoken by 7 different speakers. We keep these 140 sentences (7* 20) as the test corpus in order to be speaker independent.

- For each sentence, we generate the corresponding lattice, L_i, $i = 1, \dots, 140$.
- For each keyword, ϕ_k, $k = 1, \dots, 7$:

1. We compute its probability of occurrence in every sentences: $P(\phi_k|L_i)$.
2. We sort the 140 sentences according to their probability.

- *Parameter evaluation*

Before comparing the methods, the number of parameters are given in the table *(2)* according to the respective method.

Method	Local Probability	Parameters
Frame Labeling	Gaussian	1952
	Multi Gaussian	31232
	Neural Network	15661
HMM	Gaussian	9577
	Neural Network	19382
REMAP	Neural Network	44982

Table 2:

- *"Position" measurement*

Collecting the mean position (over the different keywords) of the 7 occurrences of each keyword in the sorted list leads to the table *(3)* :

Method	Local Probability	1 occ.	2 occ.	3 occ.	4 occ.	5 occ.	6 occ.	7occ.
Frame Labeling	Gaussian	2.1	4.45	8.75	14.40	24.3	40.65	60.15
	Multi Gaussian	2.1	4.35	8.45	11.90	19.2	33.55	61.50
	Neural Network	1.4	3.50	6.30	12.65	17.5	39.85	64.85
HMM	Gaussian	1.2	4.55	7.00	17.93	23.35	38.05	57.15
	Neural Network	1.1	2.55	7.05	12.30	23.30	29.00	47.20
REMAP	Neural Network	1.1	2.35	4.5	7.7	12.95	21.25	31.20

Table 3:

- *"Precision" measurement*

The quality of an indexing system can be measured by the "precision" and "mean precision". This figure out the proportion of correctly selected sentences out of a given part of the sorted corpus[2,14].

This measurement is given in table *(4)*:

Method	Local Probability	1 occ.	2 occ.	3 occ.	4 occ.	5 occ.	6 occ.	7occ.	Mean
Frame Labeling	Gaussian	0.77	0.71	0.60	0.52	0.38	0.25	0.17	0.48
	Multi Gaussian	0.80	0.73	0.62	0.57	0.45	0.26	0.17	0.52
	Neural Network	0.90	0.71	0.64	0.56	0.53	0.29	0.17	0.54
HMM	Gaussian	0.94	0.83	0.76	0.64	0.57	0.46	0.34	0.65
	Neural Network	0.95	0.87	0.72	0.63	0.50	0.49	0.33	0.64
REMAP	Neural Network	0.95	0.89	0.78	0.65	0.55	0.46	0.38	0.67

Table 4:

- *"Time saving rate" measurement*

This measurement[14] is based on the real used of such a system by a documentalist. Let assume a documentalist using a REMAP approach system to search for 4 occurrences of a specific keyword. He will need to listen to approximately 8 sentences to extract its 4 occurrences. Without this tool, He would have to listen to about 71 sentences. Therefore, this tools manage to save about 83% of its time. Doing such a estimation for each approach leads to the table *(5)*:

Method	Local Probability	1 occ.	2 occ.	3 occ.	4 occ.	5 occ.	6 occ.	7occ.	Mean
Frame Labeling	Gaussian	88.4	87.53	83.49	79.58	72.48	61.52	51.14	74.88
	Multi Gaussian	88.4	87.81	84.05	83.13	78.26	68.25	50.04	77.14
	Neural Network	92.2	90.14	88.11	82.06	80.11	62.23	47.28	77.45
HMM	Gaussian	93.37	87.25	86.79	74.62	73.56	63.99	53.57	76.16
	Neural Network	93.92	92.85	86.69	82.56	73.62	72.55	61.82	80.57
REMAP	Neural Network	93.9	93.4	91.5	89.1	85.3	79.9	74.6	86.8

Table 5:

- *"Receiver Operation Curves Characteristics" ("R.O.C") measurements*

According to the known association·[14] between "Precision" and "R.O.C", we can draw the operation curve characteristic in the fig. *(4)* :

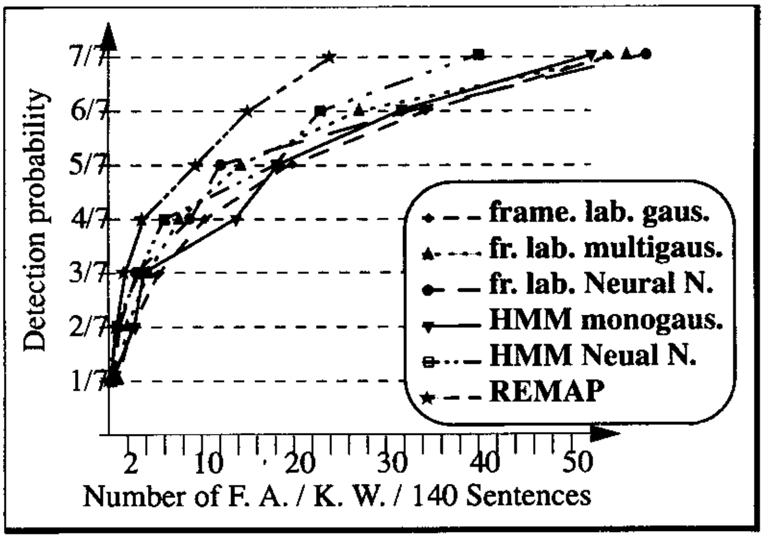

FIGURE 4.

2.1 Sensibility over keyword apparition frequency

We also had analyse the sensibility)f the system according to the keyword occurrence frequency. In order to figure out this effect, we used the first method (frame labelling) and compute the "time saving" with a equal number of keyword occurrence in database having a growing number of non keyword sentences.

These results are given in table *(6)* :

Sentences	1 occ.	2 occ.	3 occ.	4 occ.	5 occ.	6 occ.	7occ.	Mean
140	92.22	90.14	88.11	82.06	80.11	62.23	47.28	77.45
800	91.47	92.20	89.87	87.17	81.54	77.28	55.87	82.21
1095	91.25	92.07	89.72	86.68	80.27	75.45	52.55	81.14

TABLEAU 6.

This figures shows that the results, in term of "time saving" are independent of keywords occurrence frequency.

2.2 Comments

Theses results show the efficienceness of the markovian models against the frame labelling methods. This can be understood knowing the lexical constraints imposed by the markovian model.

In same order, comparison on "position" and "time saving" between gaussian models and hybrid model shows an advantage for the hybrid methods against the gaussian one.

Good REMAP results can be explained with the language model built in the network, leading to more flexible modelisation, but its major enhancement probability came from the context dependent input and the higher number of parameters used in the model.

3. CONCLUSION.

The ambitious task of keyword spotting on speaker independent data without restrictions on the vocabulary has been tackled. A frame labelling approach has been compared with standard HMM and REMAP approaches. This comparison shows that better results can be expected with this new approach. Due to the lattice structure, an improvement in the search time, compared to more classical approach has been shown. As emphased by the "time saving" measurement, this paper shown the feasability of such an indexing tool.

REFERENCES

1. H. Bourlard, Y. Kamp, H. , Ney, C. J. Wellekens, *Speaker-Dependent Connected Speech Recognition via Dynamic Programming and Statistical Methods*, Speech and Speaker Recognition, Karger, 1985.
2. D. A. James, S.J. Young, "A Fast Lattice-Based Approach to Vocabulary Independent Wordspotting", Proc. Int. Conf. Acoust. Speech and Signal Processing, pp. I 377-I 380, 1994.
3. H. Bourlard, N. Morgan, "Connectionist speech recognition: a hybrid approach", Kluwer Academic Publishers,1994.
4. L. Rabiner, B.-H. Juang, *Fundamentals of Speech Recognition*, PTR Prentice-Hall Inc, 1993.
5. J. R. Deller, J. G. Proakis, J. H. L. Hansen, *Discrete-Time Processing of Speech Signals*, Macmillan Publishing C°, 1993.
6. H. Bourlard, C. J. Wellekens, "Links between Markov Models and Multilayer Perceptrons", Ed. D. Touretzky, pp.502-510, Morgan-Kaufmann Publishers,Denver, CO, 1989.
7. R.C. Rose, E.I. Chang, R.P. Lippmann, "Techniques for Information Retrieval from Voice Messages", Proc. Int. Conf. Acoust. Speech end Signal Processing, 1991.
8. Ph. Gelin, C. J. Wellekens, "Keyword spotting enhancement for video soundtrack indexing", ICSLP, 1996, Philadelphia, PA, 1996.
9. H. Bourlard, K. Yochai, N. Morgan, "REMAP: Recursive Estimation and Maximization ôf A Posteriori Probabilities in connectionist speech recognition", Proc. EUROSPEECH'95, Madrid, September 1995.
10. D.A. James, S. J. Young, "A Fast Lattice-Based Approach to Vocabulary Independent Wordspotting", Proc. ICASSP, 1994.
11. A. P. de Vries, "Television information filtering through speech recognition", European Workshop IDMS'96.
12. G. J. F. Jones, J. T. Foote, K. Sparck Jones, S. J.Young, "Retrieving spoken documents by combining multiple index sources.", SIGIR, Zurich, 1996.
13. S. Nakamura, T. Akabane, S. Hamaguchi, "Robust word spotting in adverse car environment", Proc. Eurospeech, Berlin 1993.
14. Ph. Gelin, "Détection de mot clé dans un flux de parole, application à l'indexation de document multimédia", PhD. Thesis (1658), Ecole polytechnique Fédérale de Lausanne, Mai 1997.