Query-Driven Strategy for On-the-Fly Term Spotting in Spontaneous Speech

Mickael Rouvier, Georges Linarès, Benjamin Lecouteux

Query-driven strategy for on-the-fly term spotting in spontaneous speech

Mickael Rouvier, Georges Linarès, Benjamin Lecouteux

LIA, University of Avignon, France

Abstract

Spoken utterance retrieval was largely studied in the last decades, with the purpose of indexing large audio databases or of detecting keywords in continuous speech streams. While the indexing of closed corpora can be performed *via* a batch process, on-line spotting systems have to synchronously detect the targeted spoken utterances. We propose a two-level architecture for on-the-fly term spotting. The first level performs a fast detection of the speech segments that probably contain the targeted utterance. The second level refines the detection on the selected segments, by using a speech recognizer based on a query-driven decoding algorithm. Experiments are conducted on both broadcast and spontaneous speech corpora. We investigate the impact of the spontaneity level on system performance. Results show that our method remains effective even if the recognition rates are significantly degraded by disfluencies.

1. Introduction

Term detection has been extensively studied in the last decades in the two different contexts of spoken term detection (STD): large speech databases and keyword spotting in continuous speech streams. The first topic recently faced a growing interest, stemming from the critical need of content-based structuring of audio-visual collections. Since the STD task relies on the indexing of the whole speech database, word spotting systems perform a sequential parsing of the speech stream with the purpose of detecting the targeted word sequence. Here, we focus on on-the-fly term spotting, where the detection must be synchronously notified, at the moment where it occurs in the speech stream. This task refers to a usage scenario where early detection is critical, such as supervision and automation of operator-assisted calls (Wilpon et al., 1990; Wohlford et al., 1980).

For all these detection tasks, performances reported in the literature are quite good on clean conditions, especially on broadcast news data that were largely used for speech processing system benchmarking (Fiscus et al., 2007; Garofolo et al., 2000). In more difficult conditions, such as noisy or spontaneous speech, performances are dramatically degraded by recognition errors (Pinto et al., 2008; Yu et al., 2005; Saraclar, 2004).

Efficiency and scalability issues are generally considered as critical in detection tasks, due to the size of speech databases or, in the spotting case, due to the need of as soon as possible (ASAP) detection. Some aspects of this problem are commonly encountered in both spotting and large vocabulary continuous speech recognition (LVCSR) contexts, such as fast likelihood computation (Bocchieri and Mak, 1997; Ortmanns et al., 1997) or fast acoustic matching (Knill and Young, 1996; Cardillo et al., 2002). On the STD task, the search algorithm operates on data that were indexed by an off-line process. On-the-fly term spotting adds new problems due to on-line processing: the entire speech database is not available for indexing, and the full processing chain, from the signal to the final decision, must be performed as fast as possible in order to limit the delay between the speech utterance and the notification of detection.

Even if using only a real-time recognition system could be envisaged, this approach has two major drawbacks: first, strict pruning schemes have to be used to reach real time, impacting dramatically on the word error rate (WER), especially in adverse acoustic conditions; secondly, automatic speech recognition (ASR) usually relies on closed dictionaries, and some specific modeling strategies have to be used for out-of-vocabulary words (OOV) handling. This lexical coverage problem is a key issue in term detection, the system effectiveness being highly sensitive to it: OOV are frequently meaningful words and may probably be queried by the user. A solution is to map the terms in a sub-lexical representation allowing for the search of terms without using any recognition lexicon (Manos, 1996).

Generally, subword-level decoding consists in a fast acoustic matching between the signal and the phonetic or syllabic transcription of the term (Rose, 1993; Lau and Seneff, 1997). Various developments of this idea were evaluated in the past, with the purpose of being able to detect OOV and of improving system robustness (Manos, 1996). In (Pinto et al., 2008), the phonetic search integrates the phoneme confusion matrix in order to limit the impact of recognition errors. Other authors combine complementary acoustic scoring methods, for example Gaussian mixture models (GMM) and multilayer perceptrons (MLP)-based estimators (Pinto et al., 2007; Bourlard et al., 1994). In (Yu et al., 2005), the authors propose, in the context of the STD task, to estimate the scores of the OOV by combining the posterior probabilities of their phonetic substrings. Therefore, many of the fast wordspotters are based on off-line phonetic matching. They generally use two models representing respectively the targeted word (or term) and the "garbage", the latter aiming to "absorb" all non-targeted utterances (Bourlard et al., 1994; Manos and Zue, 1997; Junkawitsch et al., 1996). These models are built from Hidden Markov Models (HMM) representing sub-lexical units, typically phonemes or triphones.

Phonetics-based approaches allow for a high speed spotting and OOV detection, but the system's performance suffers from a lack of linguistic information that help distinguish targeted terms from phonetically close utterances (Szoke et al., 2005), especially on short phonetic sequences (Cardillo et al., 2002). Therefore, many authors proposed hybrid approaches that combine phonetic search and ASR-based detection in off-line detection systems, in both spotting

and STD contexts (Szoke et al., 2008; Logan et al., 2005; Akbacak et al., 2008; Mamou et al., 2007).

In this paper, we investigate the use of such a hybrid approach in the specific context of on-the-fly term spotting. We propose a two-level architecture in which the first level performs a phonetic filtering of the speech streams, while the second level involves an open-vocabulary LVCSR system. These two cascaded components are optimized in order to sequentially maximize the recall at the first level, and precision at the second.

At the first level, fast-matching is viewed as a filtering task that aims to accept or reject segments, according to the probability of the targeted terms being inside. Starting from this idea, we present a general scheme in which the term pronunciation graph is mapped into a graph of phonetic filters. The resulting graph is then pruned in order to minimize its complexity, while maximizing its detection capacity.

At the second level, speech segments that passed the first filtering step are processed by an ASR-based term spotter with the purpose of refining the term detection. We propose to improve the detection rate by integrating the query (i.e. the searched word sequence). This integration is based on the driven decoding algorithm (DDA) that was previously proposed in Lecouteux et al. (2006)).

The rest of the paper is organized as follows: Section 2 presents the global architecture of our term spotter. Section 3 describes the first level, that aims to identify the speech segments in which the query probably is. We first present a GMM-based approach to acoustic filtering, and we extend the method to neuromimetic filtering. In Section 4, we present the second level, where a query-driven decoding strategy is used for refining the term spotting. In Section 5, we present the experimental framework; results on clean and spontaneous speech are reported and discussed in Section 6. Finally, we conclude the paper and we propose some perspectives.

2. Principle and System Architecture

Starting from a text query composed of a short sequence of words, the term spotting system is supposed to scan a speech stream and to synchronously notify any occurrence of the targeted word sequence.

The global processing chain consists of two stages. In the first stage, the spotter is configured according to the query. This query-dependent adaptation concerns the two main components of the system that are: (i) the phonetic spotter, and (ii) the query-driven ASR system. Obviously, no information about the speech stream is available at this moment. Then, the detection system is ready to perform the synchronous scanning of the speech stream.

The approach that we propose consists in building, at the first stage, a query-dependent detection system that has to be as accurate as possible, while being able to perform on-the-fly detection. We use a two-level architecture, where the first level performs a fast, but poorly accurate detection, the detection

hypotheses being validated by a more costly detection process at the second level.

Written queries are first phonetically transcribed by using a pronunciation lexicon and a rule-based phonetizer, which produces the word-sequence pronunciation graph. Starting from this phonetic representation, an acoustic filter is built, that is composed from a graph of phonetic filters. Phonetic filters may be based on GMM or MLP. In the following, the full filtering graph is named *acoustic filter*, while *phonetic filters* operate at the node level.

At this point, our goal is to maximize the accuracy and the computational efficiency under the constraint of maximal recall rates.

Each speech segment selected by the first level is passed to the second level, as shown in Figure 1. The second level consists of an ASR system based on the driven-decoding algorithm. At this stage, ASR-based processing aims to refine the detection, focusing on precision improvement.

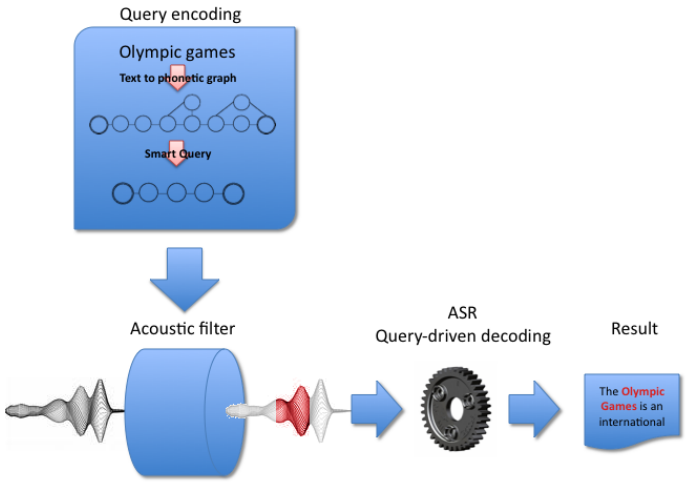

Figure 1: *A two-level architecture for on-the-fly term detection. The query is encoded as an acoustic filter that extracts relevant areas from the speech stream. Speech segments that passed the filter are processed by a query-driven speech recognizer.*

3. Acoustic filtering

3.1. Query encoding

The first step consists in transcribing the written query to phonetic strings. All the pronunciation variants of in-vocabulary terms are extracted from a dictionary that has been manually checked, since the OOV are automatically transcribed by using a rule-based phonetization system. Then, all these phonetic

transcriptions are compiled in a graph of phonemes where each path represents a pronunciation variant.

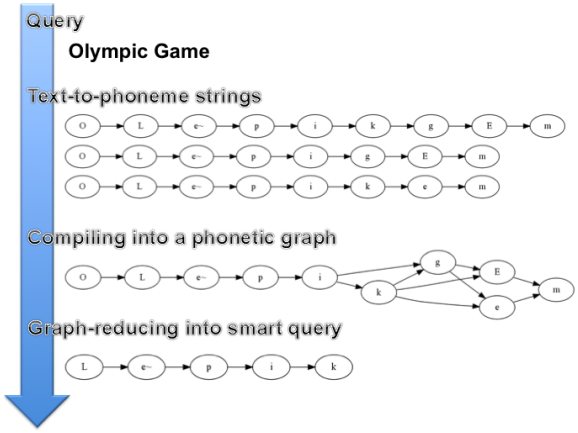

Figure 2: *From the requested term to the smart query. The written query is transcribed into a pronunciation graph. The best subgraph, which maximizes its accuracy while minimizing its CPU cost, is extracted to build the smart query.*

Classical approaches use such representations for spotting words by aligning the graph and the signal in a sliding window, the global path probability being used for taking the final detection decision. This approach is sub-optimal in terms of CPU-resource consumption: evaluation of the full path probability is generally useless as the intermediate scores could be sufficient to prune the low-probability paths. We implement such an *as soon as possible* cutoff by plugging, to each graph node, a phonetic filter that will be able to stop or to continue the path evaluation process. This filtering process is described in detail in the next two sections, where GMM-based and neuromimetic filters are presented.

Considering this strategy of ASAP cutoff, it is clear that the most discriminative parts of the graph should be evaluated first, with the purpose of reducing the CPU-time while preserving the spotting accuracy. Therefore, the graph may be reduced according to both the complexity and the discriminative capacity of its sub-graphs. We propose a graph reduction algorithm that is fully described in Section 3.4.

3.2. GMM-based phonetic filters

GMM-based filters use the acoustic models involved in the ASR system. Each filter f_i is associated to an emitting state S_i extracted from the HMM set of the ASR system. The phonetic graph is developed according to the HMM topologies, each phoneme-dependent node being splitted into a sequence of n state-dependent nodes.

The resulting state-dependent filters should be able to stop the graph exploration when observation X_t is out of the model domain. This is achieved by

specifying, for each filter, a lower limit c_i for the normalized likelihood $ll(X_t|S_i)$:

$$ll(X_t|S_i) = \frac{P(X_t|S_i)}{P(X_t|UBM)} \quad (1)$$

where X_t is a speech frame of 39 coefficients, composed of 12 PLP coefficients, energy and their first and second order derivatives. The *Universal Background Model* (UBM) is a generic model that represents the speech signal, independently of the phonetic units. Here, UBM is a GMM of 64 Gaussian components, estimated by using the *Expectation-Maximization* procedure on the training corpus.

The filter-dependent cutoff thresholds c_i are estimated on the training set, by computing the upper bound of c_i values, under the constraint $ll(X_t|S_i) > c_i, \forall X_t \in \Omega_i$, where Ω_i is the subset of the training corpus emitted by state S_i.

When the final node of the graph is reached (i.e. all phonetic filters were passed), a last selection rule is applied at the segment level. This rule relies on the full path probabilities of the targeted terms, normalized by the segment duration. We search first, in the training corpus, the lowest probability of the targeted terms. We use this lower bound C as a rejection threshold. Therefore, all accepted speech sequences $X = \{X_t\}$ satisfy the constraint:

$$P(X|S) > C \quad (2)$$

where $S = \{S_i\}$ is the state sequence corresponding to the phonetic string, and C is the query-dependent threshold.

3.3. Neuromimetic phonetic filters

Discriminative methods for word spotting have been recently investigated in (Keshet et al., 2009; Ezzat and Poggio, 2008; Benayed et al., 2004). This approach is motivated by the fact that spotting should be stated as a classification task (in rejected/accepted hypotheses), rather than a probability estimation task. The goal of acoustic filtering is to reject non-relevant segments. Considering that discriminative approaches should be more efficient for segment filtering, we propose the use of multi-layer neural networks as phonetic filters.

MLP-based filtering integrates the general scheme that was used with GMM-based filters, GMM-based phonetic filters being simply substituted for MLP classifiers as probability estimators.

We use one MLP classifier for each of the emitting states S_i that compose an initial context-independent HMM set. We follow the modeling method proposed in (Ellis and Morgan, 1999; Zhu et al., 2005). MLP filters operate at the frame level. The input vectors are composed of 351 coefficients, resulting from the concatenation of 9 frames of 39 coefficients each. The latter are classical 12 PLP coefficients, energy, and the first and second order derivatives of these 13 components. The hidden layer is composed of 1024 cells. MLPs are trained on a large corpus by using the classical back-propagation learning rule. This training step relies on a state-level segmentation that is performed by using the ASR system and its HMM-based acoustic models.

Each classifier has one output layer that is supposed to provide an estimation of the probability $P(X_t|S_i)$ of the frame X_t, given the state S_i. MLP-based phonetic filters are then integrated in the filtering graphs in a similar way to GMM-based filters: a cutoff threshold c_i is associated to each of these neural nets, allowing for the rejection of the detection hypothesis when the output score is low. The c_i values are computed on the training corpus, by estimating the lowest output value obtained by the positive training examples emitted by the S_i states. A segment-level threshold C is used for rejecting the detection hypothesis when the full path probability $P(X|ph)$ is lower than C. This full path probability is estimated by a Viterbi alignment based on neural probability estimators, and normalized according to the size of the considered path.

Finally, the filtering strategy is strictly similar to the one used in the GMM case. MLPs are used as probability estimators, and integrated as phonetic filters with respect to the global filtering scheme initially designed for the GMM-based filtering.

3.4. Smart phonetic queries

The basic idea of this mechanism is that some parts of the phonetic query may have a discriminative capacity significantly better than others, for different reasons; first, the less frequent a phoneme-sequence is, the more specific to the targeted term this sequence is. Secondly, according to the phonetic filter performances, the use of partial queries may provide a better complexity/accuracy trade-off. For example, the search for "olympic games" could be reduced to the phonetic pattern "ympic g...", with a significant computational gain and without any significant negative impact on accuracy. It is important to note that the recall rates are not influenced by the query reduction, an utterance spotted by the full phonetic string being necessarily spotted by any of its substrings.

A similar issue has been addressed in a different context by (Yu et al., 2005; Allauzen et al., 2004). The authors proposed to handle OOV queries in an audio search task. They approximated term frequency by backing-off to the frequency of the phonetic substrings of the targeted terms. Our idea is to find the optimal substring in terms of both accuracy and complexity, with the purpose of maximizing the former, while minimizing the latter.

At this point, the question is how to find the best subgraph. The first step is to define an objective function $F_{ob}(f)$ that quantifies the complexity/accuracy trade-off for a given filter f associated to a multi-word query W.

For simplicity, we first linearize the graph by merging competing models into a common phonetic filter. The resulting filter $f = \{f_i\}_{i=0,n}$ is composed of the cascade of the n phonetic filters f_i, corresponding to a phonetic sequence ph and to the associated state sequence S_i. The relevance of f is estimated *via* the objective function $F_{ob}(f)$ that combines a computational cost term $cpx(f)$ with an accuracy index $acc(f)$.

We use a complexity index $cpx()$ that relies on an estimate of the number of frames that may be submitted to each phonetic filter f_k. The probability of reaching f_i depends on the probability of passing all the previous filters $f_{k,i>k\geq 0}$

in the cascade of filters. In order to estimate the probability of passing a filter f_i, we associate, to each of them, a random variable $D_i(X_t)$ that indicates whether a frame passed the filter, or not. Therefore, D_i is set to 1 when the inequality $ll(X_t|S_i) > c_i$ holds, and D_i is set to 0 otherwise. The prior probability of passing f_i is denoted by $P(D_i = 1)$. Prior probabilities are estimated by counting the number of frames that pass the filter on the training corpus, normalized by the total number of frames.

The prior probability of reaching the phonetic filter i is the product of the prior probabilities $P(D_k = 1), k < i$ of passing the previous filters f_k.

Finally, the computational cost of the cascade f of filters is estimated by summing over all prior probabilities of reaching the filters from the cascade:

$$cpx(f) = g * (1 + \sum_{k=0}^{n} \prod_{i=0}^{k} P(D_i = 1)) \tag{3}$$

were g is a constant computational cost factor that will be set to 1 in our experiments.

The accuracy of the filter $f = \{f_i, f_{i-}...f_0\}$ can be defined as the prior probability that f performs a correct detection. This value depends on two elements. First, the smart phonetic query may match an incorrect word utterance even if the two phonetic strings are identical. For example, the search for "Olympic games" by using the very short sub-query "pic" will probably return many wrong, but acoustically close, words such "picture". Secondly, the phonetic filters may fail, by accepting false utterances.

The first element may be evaluated by estimating, in the training corpus, the probability of the targeted term W when the phonetic sequence ph is encountered. This value is computed as follows:

$$P(W|ph) = \frac{|W|}{|ph|} \tag{4}$$

where $|W|$ is the number of utterances of the term W in the training corpus, and $|ph|$ is the number of utterances of the phonetic sequence ph in the same corpus.

In a similar way, the phonetic filter accuracy $P(S_i|D_i = 1)$ represents the prior probability that the filter f_i perform a correct detection. This value is estimated on the training corpus, by counting the number of frames that passed the filter, while actually being emitted by the state S_i.

Finally, the global accuracy of the filter f is estimated according to the accuracy of each of its phonetic filters f_i and to the accuracy of the phonetic sequence $ph = \{S_i\}$:

$$acc(f) = P(W|ph) * \prod_{i=0}^{n} P(S_i|D_i = 1) \tag{5}$$

The objective function is defined as:

$$F_{ob}(f) = acc(f) - \gamma * cpx(f) \tag{6}$$

where γ is a fudge factor empirically determined.

This function is used for sub-queries ranking, the selected *smart phonetic query* being the one that maximizes F_{ob}:

$$f^{sq} = \arg\max_{k} F_{ob}(f^k) \tag{7}$$

For each query W, the sub-query selection is achieved by an exhaustive evaluation of all parts of the cascade of filters f. Then, the initial full filter f is substituted for the sub-query f^{sq}, and this reduced filter is used for the acoustic filtering achieved at the first level in our system.

This technique of best phonetic substring search is used for both GMM-based and MLP-based system. Nevertheless, the F_{ob} function relies on the accuracy of the phonetic filters f_i that are dependent on the frame-level probability estimators. Therefore, the smart phonetic query selection process is performed independently for the GMM and MLP based filtering methods.

4. Query-driven decoding

The goal of this step is to refine the detection achieved at the first level. Speech segments that passed the filtering process are submitted to the ASR system for a full decoding pass. In order to be sure that the speech segment contains the full targeted speech utterance even if only a part of the phonetic string is spotted (due to smart queries), we enlarge the segment before and after the spotted area. In our experiments, we used an offset of 0.5 second from the segment borders.

Spotting by using ASR systems is known to be focused on accuracy, since the prior probability of having the targeted terms in a transcription is low. On the other hand, transcription errors may introduce mistakes and lead to misses of correct utterances, especially on large queries: the longer the searched term is, the higher the probability of encountering an erroneous word is. In order to limit this risk, the prior probability of the query is slightly boosted by the driven decoding algorithm (DDA) (Lecouteux et al., 2006).

This algorithm aims to align a priori transcripts by using a speech recognition engine. The algorithm proceeds in two steps. First, the provided transcripts h_p and the current hypothesis h_c are synchronized by using an alignment algorithm by minimization of the editing distance between the two word strings h_p and h_c.

Once the hypothesis is aligned with the transcript, the algorithm estimates the matching transcript-to-hypothesis score (denoted α). This score is based on the number of words in the short-term history, which are correctly aligned with the transcript: only three values are used, corresponding respectively to a full alignment of the current trigram, a full alignment of the current bi-gram and an alignment of one word only. Values of α are empirically determined, by testing various configurations on a development corpus. Then, trigram probabilities are modified by using the following re-scoring rule:

$$\tilde{P}(w_i|w_{i-1}, w_{i-2}) = P^{1-\alpha}(w_i|w_{i-1}, w_{i-2}) \quad (8)$$

where $\tilde{P}(w_i|w_{i-1}, w_{i-2})$ is the updated trigram probability of a word w_i knowing the history w_{i-1}, w_{i-2}, and $P(w_i|w_{i-1}, w_{i-2})$ is the initial probability of the trigram.

Here, we used DDA as a post-processor operating on a segment previously identified as a good candidate by the acoustic filter. The targeted terms are used as a priori transcripts, leading to a slight boosting of the linguistic scores of the hypotheses that match the query.

At this step, OOV probabilities are interpolated by backing off to unknown word probabilities. Unknown word probabilities are estimated classically: we tag as *unk* all the words in the training set that are out of the recognition vocabulary. Then, *unk* is viewed as a word and its linguistic probabilities are classically estimated.

Then, a trigram probability of an OOV word w_{oov} can be decomposed according to the conditional probability of the unknown word and the probability of w_{oov}, given *unk*:

$$P(w_{oov}|w_{-1}, w_{-2}) = P(w_{oov}|unk) * P(unk|w_{-1}, w_{-2}) \quad (9)$$

Here, we use a priorly fixed value for $P(w_{oov}|unk)$. In the following experiments, this probability is set to 10^{-4}.

5. Experimental framework

5.1. The LIA broadcast news system

The experiments reported in this paper are carried out by using the LIA broadcast news (BN) system, which was involved in the ESTER evaluation campaign (Linarès et al., 2007). This system relies on an A* decoder with HMM-based context-dependent acoustic models and trigram language models. HMMs are classical three-state left-right models; state tying is achieved by using decision trees. Acoustic vectors are composed of 12 PLP coefficients, the energy, and first and second order derivatives of these 13 parameters. Two configurations are involved in the experiments, according to their decoding speed expressed as a real time factor, i.e. the time required by the system to decode one hour of speech signal. We used the real-time (noted 1xRT) and the three times real-time (noted 3xRT) systems in the experiments. The 1xRT system uses acoustic models that have only 24 Gaussian components per state and a strict pruning scheme, whereas the 3xRT system relies on 64 Gaussians per state models.

5.2. The EPAC and ESTER corpora

ESTER is a large corpus developed in the framework of the ESTER-2005 evaluation campaign. It is composed of 80 hours of French broadcast news. We use these materials as a training set, for both GMM and MLP estimates. Tests

are conducted on the EPAC corpus, which is provided by the EPAC project (Dufour et al., 2009). This project aims to investigate methods for spontaneous speech recognition and understanding. With this purpose, about 11 hours of spontaneous speech were extracted from the non-transcribed ESTER database and manually labeled according to their degree of spontaneity: degree 1 stands for read speech, and degree 10 stands for highly disfluent speech. Here, we consider two classes: *medium*, corresponding respectively to degrees 1 to 4, and *high*, corresponding to degrees 5 and above.

In the sequel, the EPAC corpus is used only for testing. Acoustic filtering and smart querying are calibrated on the ESTER training materials. 270 test queries composed the test set, including 130 in-vocabulary (IV) queries, 70 OOV and 70 hybrid queries, the latter including both known words and OOV. The query size is 1 to 4 words long, hybrid queries being composed of at least 2 words. The baseline performance of the ASR system in the 1xRT configuration is 40.3% WER, corresponding to WERs of 33.2% and 47.2% on medium and highly spontaneous subsets, respectively. In the 3xRT configuration, these rates decrease to 31.1% and 43.5%.

6. Results

6.1. Phonetic filtering evaluation

The acoustic filtering is evaluated in various configurations. The baseline system consists of a classical phonetic matching, which uses a Viterbi alignment between the phonetic graph and the signal window. The acoustic models are context-independent HMMs trained on ESTER data. We first study the impact of the ASAP pruning technique (A-GMM). Then, smart querying is added to the previous filtering system (A+SR-GMM). Finally, we evaluate MLP-based filtering, with ASAP pruning and smart querying (A+SR+MLP). In Table 1 we show the results on the EPAC spontaneous speech database in terms of recall rates, real-time factor (RT-factor) and filtering rates, the latter being the cumulated duration of the selected speech segments, normalized by the whole duration of the speech stream.

Table 1: *Acoustic filtering performed by a simple Viterbi alignment on context-free HMMs (Baseline), with GMM-filtering and ASAP pruning (A-GMM), with ASAP and smart querying coupled with GMM-based filters (A-SR-GMM) and MLP-based filters (A-SR-MLP). Performances are reported in terms of recall, filtering rates and CPU time consumption. Tests are conducted on the EPAC spontaneous speech corpus.*

	Baseline	A-GMM	A-SR-GMM	A-SR-MLP
Recall	0.99	0.97	0.97	0.97
Filt. rate	0.65	0.33	0.37	0.23
RT-fact.	0.1	0.05	0.03	0.05

Results show that the ASAP pruning technique allows for a drastic reduction of the number of accepted speech segments. Smart querying does not impact significantly on the filtering rates, but provides a strong CPU-time saving, the filtering time being reduced by a factor of two. Comparisons between GMM and MLP filters demonstrate the efficiency of a discriminative approach in such a filtering task. As expected, MLP performs a much more selective filtering of the speech segments (from 37% to 23%), at similar recall rates.

6.2. Evaluation of the query-driven decoding strategy

Here, the performance of the full system is evaluated. We report baseline results obtained with the LIA real-time ASR system (ASR-1xRT). In order to have a glimpse on the performance of the system without strong constraints on the decoding time, results for the 3xRT system are also reported. For these two systems, no query-dependent mechanisms are used, the search of terms being directly performed on the outputs of the ASR system.

Then, we estimate the detection rates by using DDA only, without acoustic filtering (DDA-1xRT). Considering the filtering of speech streams, only 37% of the whole speech duration has to be processed by the recognition system (and 23% for MLP).

Methods based on both acoustic filtering and driven decoding are evaluated by using a more accurate ASR system. We take advantage of the filtering by using a 3xRT configuration, the full process satisfying the real-time constraint.

Performances obtained with the full filtering method based on GMM (GMM+ DDA-3xRT) and on MLP (MLP+DDA-3xRT) are reported in Table 2 in terms of the F-measure, which is computed as the harmonic mean of the recall and the precision:

$$F = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}} \tag{10}$$

Table 2: *The F-measure on the EPAC spontaneous speech test set, for the ASR-only approach in real-time (ASR-1xRT) and 3xRT configurations (3xRT), for the ASR with driven decoding algorithm and without acoustic filtering (DDA-1xRT), on the full system including acoustic filtering and driven decoding, with GMM-based filters (DDA-AF-GMM) and MLP-based filters (DDA-AF-MLP).*

System	IV	OOV	Hybrid	Total
ASR-3xRT	0.66	x	x	x
ASR-1xRT	0.56	x	x	x
DDA-1xRT	0.65	0.79	0.75	0.72
DDA-AF-GMM	0.78	0.86	0.76	0.77
DDA-AF-MLP	0.76	0.89	0.80	0.80

The results show that DDA provides significant improvements in all cases: by using the real-time DDA algorithm, the F-measure is similar to the one obtained

with the best ASR-only 3xRT configuration, which is clearly out of the speed requirement for on-the-fly processing. The full two-level system benefits from both acoustic filtering and query-driven decoding; the absolute F-measure gain is of 20%, compared to the ASR-1xRT on IV queries. Compared to the *DDA-1xRT* system that handles OOV queries, the combination of acoustic filtering and query-driven decoding provides an absolute F-measure gain of about 8%. The boosting of linguistic probabilities seems to be really efficient for ASR-based spotting: on multi-word queries, the prior probability of making a mistake in the whole sequence is high. By integrating the query itself in the recognition process, we provide additional information that tends to limit the errors on the targeted utterance.

The last point is that all DDA-based systems perform better on OOV and hybrid queries. OOV are relatively long, and the size of the phonetic sequences clearly helps in the identification process. Moreover, the use of penalized unknown word probabilities tends to increase the real probability of the targeted terms, which would be very low on OOV, the recognition lexicon being built by collecting the most frequent words from the training corpus.

6.3. Detection performance according to the spontaneity level

The following experiments investigate the impact of the spontaneity level on the detection rates. We use the classification in medium and high spontaneity level, by relying on our two-level STD system.

Table 3: *Detection rates for the real-time ASR system with query-driven decoding* **DDA-1xRT***, according to the level of spontaneity. Tests are conducted on the 11-hour EPAC test corpus, by using 270 queries composed of 1 to 4 words (70 OOV queries, 70 hybrid and 130 IV queries) .*

Spontaneity Level	Recall	Precision	F-measure
Medium	0.63	0.97	0.76
High	0.62	0.65	0.63

Table 4: *Detection rates of the two-level system with GMM-based acoustic filtering and query-driven decoding* **DDA-AF-GMM***, according to the level of spontaneity. Tests are conducted on the 11-hour EPAC test corpus, by using 270 queries composed of 1 to 4 words (70 OOV queries, 70 hybrid and 130 IV queries).*

Spontaneity Level	Recall	Precision	F-measure
Medium	0.65	0.97	0.78
High	0.74	0.81	0.77

The results for the ASR system with query-driven decoding (*DDA-1xRT*) are reported in Table 3. As expected, the performances are significantly affected

Table 5: *Detection rates of the two-level system with Neuromimetic acoustic filtering and query-driven decoding* **DDA-AF-MLP**, *according to the level of spontaneity. Tests are conducted on the 11-hour EPAC test corpus, by using 270 queries composed of 1 to 4 words (70 OOV queries, 70 hybrid and 130 IV queries).*

Spontaneity Level	Recall	Precision	F-measure
Medium	0.73	0.97	0.83
High	0.73	0.83	0.78

by disfluent speech, the F-measure decreasing from 0.76 to 0.63; the recall rates remain stable, but the precision rate decreases by about 0.32 in absolute value. The acoustic filtering clearly provides a gain in all the conditions, but the more interesting point is that it seems to be highly robust to spontaneous speech: the results reported in Table 4 show that GMM-based filtering leads to similar results on medium and high spontaneity levels in terms of F-measure, the degradation of the precision rate being compensated by the improvement of the recall rate. The MLP-based system (see Table 6.3) outperforms the GMM-based system on medium spontaneity degrees (from 0.78 to 0.83), but the F-measure is affected by speech spontaneity. For highly spontaneous speech, GMM and MLP -based approaches perform similarly.

7. CONCLUSIONS AND PERSPECTIVES

We presented a two-level architecture for on-the-fly term spotting, where the full process is query-driven. The first level relies on an optimized representation of the query as a cascade of phonetic filters. The second level performs a query-driven decoding on speech segments that passed the first-level filter. We evaluated the performance of this technique on spontaneous speech. Results demonstrated that ASAP pruning combined with sub-query search improves significantly the phonetic matching efficiency, in all test conditions. Moreover, query-driven decoding provides a significant improvement compared to unconstrained decoding. The performances according to the level of spontaneity show that the proposed methods are be more robust to disfluencies than ASR-only ones, with respect to the real-time constraint.

Globally, experiments demonstrate the interest of integrating query-dependent information in the detection process, especially with ASR-based spotting. At the acoustic level, this allows for a fast matching that benefits from the particularities of the query phonetic sequence. At the linguistic level, boosting the n-gram probabilities of the word sequence improves significantly the performance of ASR-based spotting systems.

Since the proposed architecture is designed for on-the-fly term spotting, some of the techniques herein could be used in the spoken term detection task as well. We now plan to develop our proposal in this way.

References

Akbacak, M., Compernolle, D., Teodorescu, R., Stolcke, A., 2008. Open-vocabulary spoken term detection using graphone-based hybrid recognition systems. In: IEEE International Conference on Acoustics, Speech and Signal Processing. Las Vegas, USA, pp. 5240–5243.

Allauzen, C., Mohri, M., Saraclar, M., 2004. General indexation of weighted automata application to spoken utterance retrieval. In: Workshop on Interdisciplinary Approaches to Speech Indexing and Retrieval (HLT/NAACL 2004). pp. 33–40.

Benayed, Y., Fohr, D., Haton, J., Chollet, G., 2004. Confidence measure for keyword spotting using support vector machines. In: IEEE International Conference on Acoustics, Speech and Signal Processing. Montreal, Canada, pp. 588 – 591.

Bocchieri, E., Mak, B., 1997. Subspace distribution clustering for continuous observation density hidden markov models. In: Proceedings of Eurospeech '97. Rhodes, Greece, pp. 107–110.

Bourlard, H., D'hoore, B., Boite, J., Apr. 1994. Optimizing recognition and rejection performance in wordspotting systems. In: IEEE International Conference on Acoustics, Speech and Signal Processing. Adelaide, Australia, pp. 373–376.

Cardillo, P. S., Clements, M., Miller, M. S., 2002. Phonetic searching vs. LVCSR: How to find what you really want in audio archives. International Journal of Speech Technology 5, 9–22.

Dufour, R., Jousse, V., Estève, Y., Béchet, F., Linarès, G., 2009. Spontaneous speech characterization and detection in large audio databases. In: Proceedings of Speech and Computer, SPECOM. Saint Petersburg, Russia.

Ellis, D., Morgan, N., 1999. Size matters: an empirical study of neural network training for large vocabulary continuous speech recognition. In: IEEE International Conference on Acoustics, Speech and Signal Processing. Phoenix, Arizona, USA, pp. 1013–1016.

Ezzat, T., Poggio, T., 2008. Discriminative word-spotting using ordered spectro-temporal patch features. In: SAPA Workshop, Interspeech 2008. Brisbane, Australia, pp. 35–40.

Fiscus, J. G., Ajot, J., Garofolo, J. S., Doddington, G., 2007. Results of the 2006 spoken term detection evaluation. In: Searching Spontaneous Conversational Search, ACM SIGIR Workshop. Amsterdam, NL, pp. 51–55.

Garofolo, J. S., Auzanne, C. G. P., Voorhees, E. M., 2000. The TREC spoken document retrieval track: A success story. In: in Text Retrieval Conference (TREC) 8. pp. 16–19.

Junkawitsch, J., Neubauer, L., Höge, H., Ruske, G., 1996. A new keyword spotting algorithm with pre-calculated optimal thresholds. In: Proceedings of ICSLP '96. Vol. 4. Philadelphia, PA, pp. 2067–2070.

Keshet, J., Grangier, D., Bengio, S., 2009. Discriminative keyword spotting. Speech Communication 51 (4), 317 – 329.

Knill, K., Young, S., May 1996. Fast implementation methods for Viterbi-based word-spotting. In: IEEE International Conference on Acoustics, Speech and Signal Processing. Atlanta, GA, pp. 522–525.

Lau, R., Seneff, S., 1997. Providing sublexical constraints for word spotting within the angie framework. In: In Proceedings of Eurospeech '97. pp. 263–266.

Lecouteux, B., Linarès, G., Bonastre, J., Nocéra, P., 2006. Imperfect transcript driven speech recognition. In: Proceedings of InterSpeech'06. Pitsburg, USA.

Linarès, G., Massonié, D., Nocéra, P., Lévy, C., 2007. A scalable system for embedded large vocabulary continuous speech recognition. In: IEEE Workshop on DSP in Mobile and vehicular systems. Istanbul, Turkey.

Logan, B., Thong, J. M. B., Moreno, P., 2005. Approaches to reduce the effects of OOV queries on indexed spoken audio. IEEE Transactions on Multimedia 7, 899–906.

Mamou, J., Ramabhadran, B., Siohan, O., 2007. Vocabulary independent spoken term detection. In: SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, New York, NY, USA, pp. 615–622.

Manos, A. S., 1996. A study on out-of-vocabulary word modelling for a segment-based keyword spotting system. Master's thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA.

Manos, A. S., Zue, V. W., 1997. A segment-based wordspotter using phonetic filler models. In: IEEE International Conference on Acoustics, Speech and Signal Processing. Munich, Germany, pp. 899–902.

Ortmanns, S., Firzlaff, T., Ney, H., Sep. 1997. Fast likelihood computation methods for continuous mixture densities in large vocabulary speech recognition. In: Proceedings of Eurospeech '97. Rhodes, Greece, pp. 139–142.

Pinto, J., Lovitt, A., Hermansky, H., 2007. Exploiting phoneme similarities in hybrid HMM-ANN keyword spotting. In: Proceedings of InterSpeech '07. pp. 1817–1820.

Pinto, J., Szoke, I., Prasanna, S., Hermansky, H., 2008. Fast Approximate Spoken Term Detection from Sequence of Phonemes. In: Workshop on Searching Spontaneous Conversational Speech at SIGIR. IDIAP-RR 08-45.

Rose, R., 1993. Definition of subword acoustic units for wordspotting. In: Eurospeech '93. Berlin, Germany, pp. 1049–1052.

Saraclar, M., 2004. Lattice-based search for spoken utterance retrieval. In: In Proceedings of HLT-NAACL 2004. pp. 129–136.

Szoke, I., Fapso, M., Burget, L., Cernocky, J., 2008. Hybrid word-subword decoding for spoken term detection. In: The 31st Annual International ACM SIGIR Conference 20-24 July 2008, Singapore. pp. 42–48.

Szoke, I., Schwarz, P., Burget, L., Fapso, M., Karafiat, M., Cernocky, J., Matejka, P., 2005. Comparison of keyword spotting approaches for informal continuous speech. In: Proceedings of Interspeech '05. pp. 633–636.

Wilpon, J. G., Rabiner, L. R., Lee, C., Goldman, E. R., 1990. Automatic recognition of keywords in unconstrained speech using hidden Markov models. IEEE Transactions on Acoustics, Speech and Signal Processing ASSP-38 (11), 1870–1878.

Wohlford, R., Smith, A., Sambur, M., 1980. The enhancement of wordspotting techniques. In: IEEE International Conference on Acoustics, Speech and Signal Processing. Vol. 5. pp. 209–212.

Yu, P., Chen, K., Ma, C., Seide, F., 2005. Vocabulary-independent indexing of spontaneous speech. IEEE Transactions on Speech and Audio Processing 13 (5), 635–643.

Zhu, Q., Stolcke, A., Chen, B. Y., Morgan, N., 2005. Using MLP features in SRI's conversational speech recognition system. In: Proceedings of Interspeech '05. Lisbon, Portugal.