Automatic syllable-based phoneme recognition using ESTER Corpus

OLIVIER LE BLOUCH, PATRICE COLLEN
France Telecom R&D
4 rue du Clos Courtel, 35510 Cesson Sévigné
FRANCE

Abstract: - This paper presents an evaluation of speaker-independent continuous phoneme recognition systems on the French speech database ESTER. The tested systems are syllable-based phoneme recognizers, i.e. they use syllables as basic units together with syllabic bigram language models and HMM topologies adapted to syllables. Once identified, syllables are converted back to phones. In a previous paper, we introduced the transitory models in order to build a system for phonetic transcription guided by syllables that achieved a Phoneme Error Rate of 15.8% when tested on a small part of the French Bref80 corpus [6]. In this study, the transitory model system was tested on a more comprehensive set of test data, ESTER [3]. It was also compared to three other syllable-based systems relying respectively on monophones, on triphones with full cross syllable context expansion and on triphones with simple syllable internal context expansion. The results confirm the benefit of using syllables for phoneme identification, as well as the interest in using the transitory models in terms of complexity and processing speed compared to systems based on context-dependent models.

Key-Words: - speech recognition, audio indexing, phoneme, syllable, Bref80, ESTER.

1 Introduction

Accurate phoneme recognition is essential for many speech applications. On the first hand, it is the starting point of large vocabulary automatic speech recognition systems [4]. Phoneme transcription also plays a major role in spoken document retrieval (SDR) methods [11]. The phoneme-based approach processes the audio data with a lightweight speech recognizer to produce either a phoneme transcription or a certain kind of phoneme lattice. The generated data are directly used for keyword spotting or for keyword search [12].

This paper describes evaluation of four syllable-based approaches to automatically transcribe the 35 French phonemes used by Gauvain et al. [7]. It focuses on two major points : the significant improvements obtained thanks to the use of syllables as guideline for the phonetic transcription compared to pure phoneme recognition systems, and the interest to use the so-called transitory models according to their low complexity and processing speed.

Here, experimental results are reported from the French corpus ESTER, which was originally designed in the context of automatic textual transcription of spoken news [3].

The paper is organized as follows. First, a brief description of the audio databases used in this evaluation is given in section 2. Next, section 3 describes the four systems under evaluation, which are respectively based on monophones, on triphones with full cross syllable context expansion, on triphones with syllable internal context expansion only and on the transitory models. Finally, the results are presented in section 4 and then discussed in section 5.

2 Audio databases

The train, development and test databases used in this evaluation are mostly taken from two audio databases of continuous French speech: the well-known ESTER and BREF80 corpus. The aim of the ESTER evaluation campaign was to evaluate automatic broadcast news rich transcription systems for the French language [3]. One of the main topics of this campaign was orthographic transcription for which the best system obtained around 12% of WER for clean speech. BREF80 was designed to provide continuous speech data to develop dictation machines, for the evaluation of continuous speech recognition systems, and for the study of phonological variations [6]. The spoken texts have been selected from 5 million words in the French newspaper, Le Monde.

All of the sound files are monophonic at a 16 kHz sample rate.

2.1 Train database

The train database covers approximately 20 hours of speech, 42% taken from Bref80, 47% from the training phase of ESTER and 11% from other audio material (Radio, TV, etc.). ESTER data has been created from the phase 1 of the ESTER train corpus, by taking approximately 10 hours of wideband speech out of the 90 hours of manually transcripted audio.

2.2 Development database

The development database, *Dev_Bref80*, is made up of 52 minutes from BREF80 corpus and is not present in the training data.

2.2 Test database

The test database, *Test_Ester*, contains 7h30 of continuous wideband speech segments taken from the 10 hours of radio broadcast news in the ESTER test set; the entire set is kept except the phone bandwidth, non spoken, mixed speech and non transcribed segments.

Creating this corpus required an automatic phonetization of data by building an associated dictionary of 65K distinct words, which corresponds to approximately 300K phonetic forms. This dictionary has been used to perform a forced alignment based on monophones trained from the previous paper [9].

3 Systems description

The four systems under evaluation process the same input acoustic features to produce output phonemes. They use internal syllable modeling, identification and decomposition of syllables to produce output phonemes. Two language models have been used and the four systems share the same pool of 35 phonemes as in [7] with the following differences: no 'h' and no 'N', but an inspiration 'ii' and a short pause 'sp'.

3.1 Acoustic features

The speech signal is converted into a sequence of MFCC feature vectors with a fixed 32ms frame and a frame rate of 10ms. Each feature vector has 38 dimensions: 12 cepstral coefficients, 12 cepstral plus energy derivatives and 12 cepstral plus energy accelerations. Cepstral mean normalization is applied on each file.

All the models were trained using the Baum Welch algorithm [1] provided by HTK Software [14]. The corresponding labels were initialized using a first phoneme segmentation obtained by a force-alignment procedure. The resulting number of Gaussians is 256 for the monophone system and 32 for transitory and triphone systems.

The phoneme/syllable decoding is carried out by determining the most likely HMM state sequence using the one pass Viterbi beam search algorithm [13].

3.2 Language models

Two bigram language models have been used and trained on textual material in our train corpus using the CMU Toolkit [2].

3.1.1 Phoneme-based language model

The first one is phoneme-based and built upon the 910K phones in our train set, resulting from a forced-alignment procedure. The perplexity evaluated on *Dev_Bref80* reaches 17.24.

3.2.2 Syllable-based language model

The second one is syllable-based. As mentioned by Jones [5], a syllable is a notion quite difficult to define, but we can figure it as presented by Laver [8] and illustrated on Fig. 1. In [9], we explain some choices such as the segmentation in syllables of a whole phrase instead of just segmenting inside words as done by Jones. The purpose of this syllabization is to use syllables as a guideline for phonetic transcription and not as a real new unit.

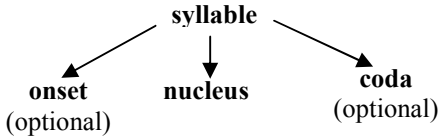

Fig. 1 : *Illustration of syllable components*

In order to build this syllable-based bigram, the phonetic transcriptions are segmented by syllables. The frequencies are computed and the most frequent syllables are selected; in our case, keeping 2000 syllables gives a good compromise between the number of units and the complexity. Finally, to avoid out-of-vocabulary syllables, these syllabic labels are reformatted by splitting all syllables which are not present in the top ones. As a result, our bigram is constructed from approximately 2000 syllables and 35 phones. The train corpus contains 424K syllables and the perplexity on *Dev_Bref80* reaches 126.34.

3.3 Overview of the systems

Four systems have been trained, based on four different kinds of models : monophones, triphones with full cross syllable context expansion, triphones with syllable internal context expansion only and transitory models. The 2000 syllables are then created by concatenation of these units, in order to apply the syllabic bigram.

3.3.1 *ASR_CI* : Context-independent

ASR_CI is the basic phoneme recognition system. It makes use of the 35 French basic phones. The phonemes are modeled by traditional context-independent 3-states HMMs, except a 5-state HMM for the silence model and a 1-state (skip) HMM for the "short pause" model. Following experiments on *Dev_Bref80*, 256-Gaussian mixtures have been selected for each state. According to this unit, a syllable is a concatenation of context-

independent models. A syllable composed of N phones has 3*N states.

3.3.2 *ASR_F* : Full Cross Syllable Context Expansion

This second system is based on state-of-the-art models i.e. context-dependent models with decision tree clustering of states. This pool contains 49283 HMMs but only 19470 physical ones sharing 6053 different mixtures of 32 Gaussians. In more detail, they are 35 "context-free" models, 2592 diphones and 46656 triphones. In the same way as for *ASR_CI*, syllables are built by concatenations of context-dependent models. Also, the context is not limited to the inside of syllables, but is expanded between them. This expansion gives this system an entire coverage of phones and co-articulation effects, but also it makes this system the more complex in terms of nodes and links in the network, as shown in section 4. A syllable composed of N phones has also got 3*N states.

3.3.3 *ASR_I* : Syllable Internal Context Expansion

With *ASR_I*, the goal is to keep context-dependent models as basic units while avoiding inter-syllables expansion in order to keep the complexity low. In the case of *ASR_F*, not forcing this expansion significantly decreases the performance because of non-adapted training. Consequently, the main difference between *ASR_I* and *ASR_F* concerns the training phase where labels are modified as illustrated in Fig. 2. The new labels are built relatively to syllables here, in order to make diphones and related tied triphones more representative of "inter-syllables" zones. The resulting coverage will obviously be more blurred than in *ASR_F*.

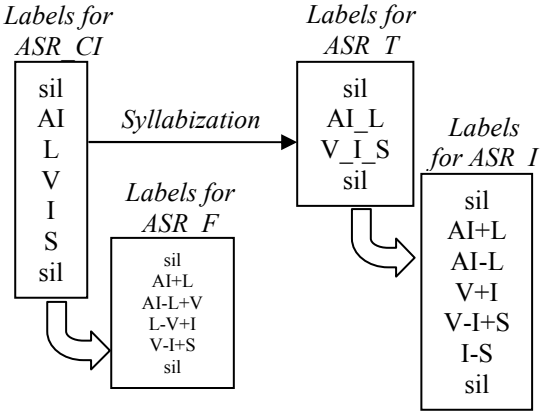

Fig. 2 : *Creation of labels for word "Elvis"*

The final pool contains 34852 HMMs but only 28687 physical ones sharing 2442 different mixtures of 32 Gaussians. A syllable composed of N phones has also got 3*N states.

3.3.4 *ASR_T* : Transitory models

ASR_T is a system based on the work presented in [9], where syllables are built from one-state models called transitory models, where X_S designs a start state, X_C a center state, X_E an end state and X_2Y a transition between phonemes X and Y. This method is illustrated by Fig. 3.

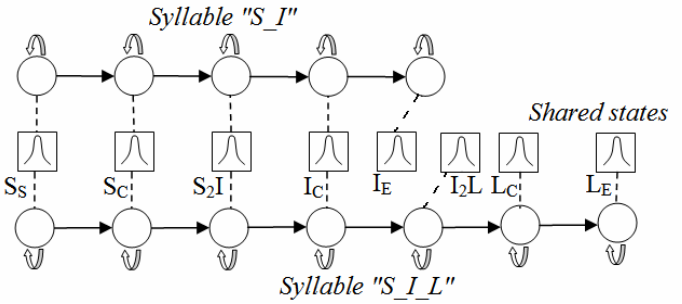

Fig. 3 : *Syllables from transitory models*

For 32 phones (silence models are treated as context-independent models), there are 1120 one-state models with a mixture composed of 32 Gaussians. Some of these models may not appear in training corpus; this is why an "untrained model" is added, containing a single global Gaussian learned on all training data.

Training *ASR_T* requires a new format for the label files (Fig. 2), i.e. transform the phoneme transcriptions into syllable transcriptions, as described in the language model section. This pool contains the 2000 syllables and the 35 phones, sharing 1103 different mixtures. A syllable composed of N phones has got 2*N+1 states.

4 Results

Tables 1 and 2 display the performances of the different systems respectively on the *Dev_Bref80* and the *Test_Ester* corpora. Recognition performances are noted relatively to both the percentage of correct recognized phones and to the Phoneme Error Rate (PER).

$$PER = 100 - Accuracy = \frac{100 * (S + I + D)}{N} \quad (1)$$

Where N is the number of phones of the reference transcription, I is the number of insertions, S is the number of substitutions and D is the number of omitted phones. Silences and inspirations have been removed prior to scoring. In the case of syllabic bigram, resulting syllables are re-split into phones before computing this error rate.

Complexity and processing speed are given in table 4 and 5. Note that we fixed a same beam threshold strategy for all experiments in the decoding process.

Results for *ASR_I* and *ASR_T* are only presented for the syllabic decoding because of their specific properties.

	Phonetic Decoding		Syllabic Decoding	
	Correct	PER	Correct	PER
ASR_CI	77.23	23.96	85.33	15.85
ASR_F	85.24	20.45	89.34	15.21
ASR_I			87.87	**15.12**
ASR_T			84.91	15.20

Table 1 : *Results on Dev_Bref80*

	Phonetic Decoding		Syllabic Decoding	
	Correct	PER	Correct	PER
ASR_CI	71.15	30.66	78.34	23.73
ASR_F	79.52	28.16	83.43	23.46
ASR_I			81.60	**22.34**
ASR_T			80.83	23.19

Table 2 : *Results on Test_Ester*

4.1 Complexity and processing speed

Key characteristics of automatic phoneme recognition not only include PER but also concern system complexity. Table 3 recalls the complexity of each system in terms of number of Gaussians per mixture, number of physical models, number of different states and number of states per syllable.

	# of mixtures	Models	States	States per syllable (N phones)
ASR_CI	256	35	105	3*N
ASR_F	32	19470	6053	3*N
ASR_I	32	28687	2442	3*N
ASR_T	32	2000	1103	2*N+1

Table 3 : *Topology complexity*

Table 4 summarizes the complexity of networks generated by HTK in terms of the number of nodes and number of links. Compared to other networks built for syllabic bigram, *ASR_F* is the more complex, with approximately 12 times more nodes and 6 times more links. This is due to extended context between syllables. Note that a node here represents a unit, that's why *ASR_T* has twice as many nodes than *ASR_I*. Indeed, according to their topology, they have a different number of states per unit. In addition, *ASR_I* has fewer nodes than *ASR_CI* because of tied models.

	Phonetic Decoding		Syllabic Decoding	
	Nodes	Links	Nodes	Links
ASR_CI	111	1389	10208	97313
ASR_F	30481	70006	199198	605369
ASR_I			8196	94435
ASR_T			16369	99419

Table 4 : *Network complexity*

Concerning the processing speed, it depends mainly on the decoding strategy. The values given below are informative and are just here to compare the different systems. Table 5 summarizes the average processing speed for the decoding of speech using the HVite decoder of HTK on a Xeon CPU, 3.4GHz with 2Go of RAM. A same beam threshold of value 180 is used; this value was found to lead to a good tradeoff between complexity and surviving in the token passing algorithm.

	Phonetic LM	Syllabic LM
ASR_CI	0.88	1.60
ASR_F	5.81	10.38
ASR_I		2.23
ASR_T		1.84

Table 5 : *Processing speed (x real-time)*

5 Discussion

Though training context-dependent models requires a huge database, and the 20 hours of the train database are probably not sufficient to draw definitive conclusions, this paragraph presents an analysis of the results obtained using this database.

First of all, Table 1 and Table 2 show that the use of syllabic bigram significantly improves performances compared to phonetic bigram with a gain of at least 5 points on PER on the two basic systems, *ASR_CI* and *ASR_F*. Performances achieved by *ASR_I* and *ASR_T* also highlight the benefit of adapting models to syllables. The results presented in section 4 indicate that the best performances on *Test_Ester* are obtained by *ASR_I* with 22.34% of PER. However, all of the tested systems have similar performances and accuracies, so the main difference will be made on processing speed and complexity.

ASR_CI results show that mixtures with a high number of Gaussians allow context-independent models to provide performances close to systems using context-dependent ones. Of course, if more Gaussians are to be computed, this will slow down the decoding process, but these computations can be optimized if needed, as explained in [10], for example.

The context-expansion between syllables and the number of tied states make *ASR_F* much more complex and slower than the other systems with very similar results in the end. On the other hand, not using context-expansion between syllables with these models increase the phoneme error rate (approximately 39% whereas *ASR_F* achieves 23.46% of PER). This is due to non-adapted training of context-dependent models for syllables.

ASR_I deals with this issue by using an adapted training, resulting in a lower network complexity.

Consequently, *ASR_I* is 5 times quicker than *ASR_F* and achieves two times real-time decoding for a same beam threshold. In addition, *ASR_I* PER is one point better than *ASR_F* : this result highlights the fact that, in our context, an accurate coverage between syllables not necessarily improves the system.

ASR_T can be considered as a "light" *ASR_I*, which deals with half as many different states thanks to a maximal use of state-sharing. As a consequence, *ASR_T* is the best tradeoff between speed and performances, by achieving 23.19% of PER.

6 Conclusion and further work

We have presented the primary results of phonetic transcription on ESTER corpus. Four different systems have been tested and their results confirm the benefit of using syllables as phonetic transcription guidelines. Two of them give particularly interesting results.

The first one is based on context-dependent models trained with an adaptation to syllables and performed with syllabic bigram and syllable internal context expansion. It gives a phoneme error rate of 22.34%. For this particular system, we also have introduced a new way of training context-dependent models. It is based on a syllabic creation of context-dependent labels in order to outperform systems using full cross syllable context expansion. The second one, based on transitory models and syllabic bigram, gives a phoneme error rate of 23.19%. It offers the best tradeoff between performances and speed.

Future work will focus on extended language models, i.e trigram and more. Furthermore, it would be interesting to work on Gaussians optimizations in order to improve the processing speed. In the end, we will study the influence of these systems on wordspotting.

7 Acknowledgements

The authors wish to thank Guillaume Gravier of IRISA for sharing his phonetic dictionary.

References:

[1] Baum, L.E., Egon J.A., "An inequality with applications to statistical estimation for probabilistic functions of a Markov process and to a model ecology", Bull. Amer. Math. Soc., 73:360-363, 1967.

[2] Clarkson, P.R., Rosenfeld, R., "Statistical language modeling using the CMU-Cambridge toolkit", Proc. ESCA EUROSPEECH 1997.

[3] Galliano, S., Geoffrois, E., and al., "The ESTER Phase II Evaluation Campaign for the Rich Transcription of French Broadcast News", INTERSPEECH 2005.

[4] Gauvain, J.L., Lamel, L., "Continuous speech recognition : Advances and applications", Proc. Of the IEEE, nol. 88, no. 8, August 2000

[5] Jones, R. J., Downey, S., Mason, J.S., "Continuous speech recognition using syllables", Proc. Eurospeech, Volume 3, pp. 1171-1174, 1997.

[6] Lamel, L.F., Gauvain, J.L., M. Eskenazi, M., "BREF, a Large Vocabulary Spoken Corpus for French", EUROSPEECH, 1991.

[7] Lamel, L.F.,Gauvain, J.L., "Experiments on Speaker-Independent Phone Recognition Using BREF", ICASSP 1992.

[8] Laver, J., "Principles of phonetics", Cambridge University Press, pp. 517-518, 1994.

[9] Le Blouch, O., Collen, P., "Reconnaissance automatique de phonemes guide par les syllables", Journées d'Etude de la parole, Dinard (France), 2006.

[10] Pellom, B.L., Sarikaya R., Hansen J.H.L., "Fast Likelihood Computation Techniques in Nearest-Neighbor Based Search for Continuous Speech Recognition", IEEE Signal Processing Letters, Vol. 8, No. 8, August 2001.

[11] Renals, S., Abberley, D., Kirby, D., Robinson, T., "Indexing and retrieval of broadcast news", Speech Communication, vol.32, pp. 5-20, 2000

[12] Seide, F., Yu, P., "A Hybrid Word / Phoneme-based Approach for Improved Vocabulary-Independent Search in Spontaneous Speech", Proc. ICLSP'04, Korean, 2004.

[13] Young, S.J., and al., "Token Passing : a simple conceptual model for connected speech recognition systems", Technical report CUED/F-INFENG/TR38, Cambridge University Engineering Dept, 1989.

[14] Young, S.J., and al., "The HTK Book (for HTK version 3.3)", Cambridge University Engineering Department, April 2005.