Advances in Intelligent Systems and Technologies
Proceedings ECIT2010 – 6th European Conference on Intelligent Systems and Technologies
Iasi, Romania, October 07-09, 2010

Speech Recognition Experimental Results for Romanian Language

Cristina Sorina Petrea, Andi Buzo, Horia Cucu,

Miruna Paşca, Corneliu Burileanu

Faculty of Electronics, Telecommunications and Information Technology,
University "Politehnica" of Bucharest
cburileanu@messnet.pub.ro

Abstract. *Automatic speech recognition in database-lacking languages like Romanian must imply new system designs or new methods in training state-of-the-art systems. We propose an innovative training strategy for Hidden Markov Models (HMM) based systems. It implies starting with isolated monophones training, going through isolated words training and finally reaching continuous speech training. Several experimental results show how the speech units used, the voice features employed, the HMM parameters utilized and the language restrictions imposed impact the recognition rates and implicitly the performance of the ASR systems.*

Keywords. *Automatic speech recognition, MFCC, HMM, phoneme, monophone, triphone, tied-state.*

1. Introduction

An automatic speech recognition (ASR) system is mainly an informational system that is able to identify the utterances that a person pronounces into a microphone. The aim of an ideal ASR system is to perform real-time recognition with 100% accuracy of all the words that are intelligibly spoken by any person, in a specific language, independent of vocabulary size, noise, speaker characteristics and accent, or channel conditions.

The state-of-the-art systems in ASR are based on Hidden Markov Models (HMM). The main steps in building such an ASR system are as follows:

- Every speech unit (phoneme, syllable, word, etc) is modeled using a HMM.
- The set of HMMs is trained using a training speech database; this step aims to create HMMs that model all the various ways of speech units' pronunciations for different speakers, in different conditions.

- The set of trained HMMs is evaluated with a testing database; this step issues the recognition rates for the ASR system.

The field of speech recognition in the Romanian language has been approached by some studies that mostly focus on isolated words recognition. The first attempts [1, 2] report the usage of the Dynamic Time Warping algorithm, while the latest papers present the results obtained with the state-of-the-art algorithms in speech recognition: Hidden Markov Models (HMM) algorithms [3, 4, 5] and neural networks algorithms [6, 7]. Studies regarding continuous speech are also presented in [5], but, as in the other cases mentioned above, the speech database used is relatively small (500 to 5000 different words) and thus the results cannot be extended to real continuous speech applications.

Our main goal is to create a continuous speech recognition system for the Romanian language. As Romanian did not have a large enough speech database for this purpose, the first step in reaching our goal was to create it. This part is presented in Section 2. Section 3 focuses on the main issues regarding the process of training the HMMs system. These issues are overcome with an innovative hierarchical training strategy for ASR systems. The results (the recognition rates) and several conclusions about fine-tuning the ASR system are summarized in Section 4.

2. Speech Database

A speech database is made up of the following components:

- a set of speech signal samples;
- a set of label files that mark the speech units that are uttered in each speech sample; these files may include information regarding the boundaries between speech unit;
- additional information about every speech sample regarding speech type (isolated words, continuous, spontaneous), speaker identity, etc;

Choosing the most appropriate speech units to be modeled is one of the important factors in obtaining good recognition results. Up to this point we have experimented using monophones and triphones. Although IPA notations are used to distinguish between different phonemes, other phoneme notations were employed for our "in-house" development and testing. The reason is that most of the tools work easier with ASCII encoded text files versus Unicode encoded files. Table 1 summarizes the correspondence between IPA and our SpeeD phoneme notations.

The hierarchical training strategy described in Section 3 and employed in creating the ASR system implies creating several types of speech databases as follows:

- isolated monophones database;
- isolated words database;
- continuous speech database.

TABLE 1. IPA – SpeeD Notation Correspondence

IPA Symbol	SpeeD Symbol		IPA Symbol	SpeeD Symbol		IPA Symbol	SpeeD Symbol
a	a		o̯	o1		f	f
ə	@		w	w		v	v
e	e		c	k2		h	h
i	i		b	b		ʒ	j
ɨ	i1		p	p		ʃ	s1
i̯	i2		k	k		l	l
o	o		ʧ	k1		m	m
u	u		g	g		n	n
y	y		ʤ	g1		s	s
ø	o2		ɟ	g2		z	z
e̯	e1		d	d		r	r
j	i3		t	t		ʦ	t1

The isolated monophones database contains 36 phonemes and was acquired by labeling (at monophone level) audio books and other spoken materials [8]. Monophone level labeling means that the speech samples are associated with label files that contain information regarding the monophones uttered in the audio file, their occurrence order and the time borders between them. All these information are really useful and important for initializing the ASR system. Due to timing reasons (monophone level labeling is time consuming) labeling was limited to only a few occurrences for the most usual phonemes in Romanian language. Table 2 summarizes the number of occurrences for each phoneme.

TABLE 2. Phonemes in isolated monophones database

Symbol	**Occur.**		**Symbol**	**Occur.**		**Symbol**	**Occur.**		**Symbol**	**Occur.**
@	30		g	25		k	26		s1	31
a	561		h	26		l	38		S	38
b	31		i1	0		m	35		t1	40
d	32		i2	26		n	48		T	23
e1	26		i3	22		o1	22		u	21
e	52		i	27		o2	0		v	32
f	26		j	22		o	13		w	31
g1	38		k2	0		p	7		y	0
g2	0		k1	33		r	38		z	26

The isolated words database was acquired by recording a set of 10000 words chosen to cover all the syllables in the Romanian language. Five speakers (two males and three females) recorded the whole set of words resulting in a database of 50000 words (one word per audio file). The label files for these speech samples were programmatically generated. The additional information regarding speaker identity and speech type are obvious as our team is the author of the recordings.

Several reasons triggered us to create this database by directly recording words, instead of labeling audio books or other spoken materials:

- The errors that can occur during the recording process are less numerous.
- It is easier and less time consuming to record single words than to find and mark the boundaries between words in an already recorded speech clip.
- The control we have over the spoken words, speaker identities, type of noise, etc is clearly higher. This enables us to create speaker dependent versus speaker independent ASR tests (see Section 4), database enlargement tests, speaker dependent recognition rates comparisons, etc.

There are also a few disadvantages identified for this method of database creation:

- Speaking isolated words is clearly different that speaking continuous phrases in a given context. Continuous speech implies specific pronunciations of the same word depending on the context, thus this database solely is not enough for creating a continuous speech recognition system.
- Moreover, there is no spontaneity in speaking isolated words, thus this database solely is not enough for creating a spontaneous speech recognition system.

Figure 1 presents a histogram of occurrences for the phonemes composing the 10000 different words in this database.

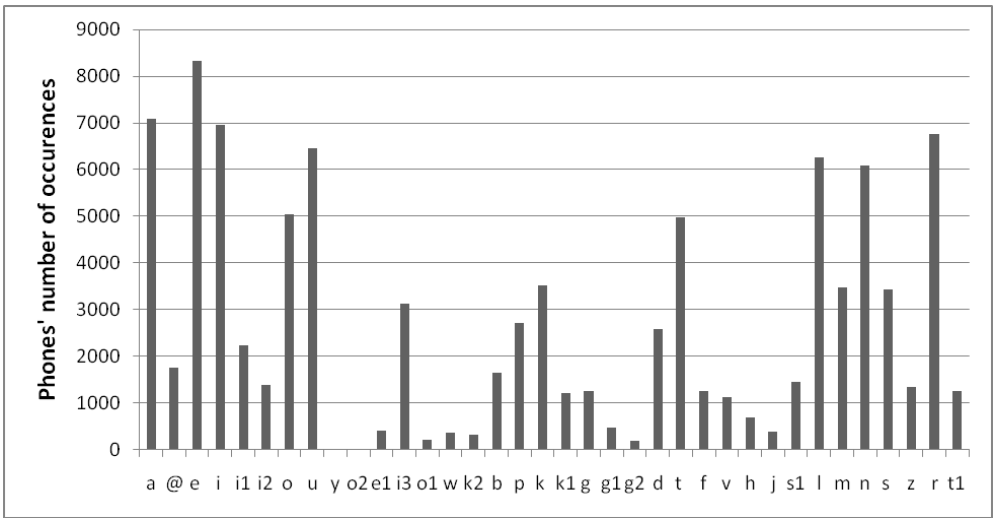

Fig. 1. Phones in isolated words database

The acquisition of the continuous speech database is currently in progress. We have selected a list of phrases from some of the Romanian online newspapers trying to obtain a balance between the treated subjects. These phrases are to be recorded by the same five speakers that created the isolated words database.

A spontaneous speech database can only be acquired by labeling previously recorded spontaneous spoken materials. Spontaneity can be found in dialogues between people that do not know they are being recorded, thus such a database cannot be recorded in laboratory environments.

3. Training Strategy

There are several issues to be considered while establishing the training strategy for HMMs. Firstly, the most important decision is choosing the appropriate speech unit to be modeled. Speech units may be monophones, triphones, syllables or even words. Studies have shown that the best results are obtained by using triphones [9] and our results have confirmed it, too.

Secondly, building a large database labeled at monophone level is an exhaustive and time-consuming process. Moreover, it is objectively difficult to find the boundary between two consecutive monophones within a word. Given these reasons, the embedded training technique has been used. The label files do not contain the boundaries between monophones but only the pronunciation order. Then, in the training process, all models within a label file are chained to form one giant HMM. After estimating the parameters, models are separated to form the original HMMs.

Thirdly, the embedded training is a comfortable technique but it is also very vulnerable to alignment. The estimation process stops once the probability that the estimated models generate the given speech utterance reaches the maximum. If, for some reason, this maximum is a local one and not a global one, training will not be optimal. In order to have the best alignment, we have proposed a hierarchical training starting from a small database of isolated monophones. Models obtained at the first step are then used as initial models for isolated words training and finally, isolated words trained models are used for continuous speech recognition.

Fourthly, since the isolated words database is not large enough, for some triphones, the number of occurrences is not sufficient and the models may not be well trained. Beside this, studies have shown [10], [11], [12] that triphones with the same central phone have the same model parameters for the central states. By considering these two aspects, we have used the tied-states technique which consists in using the same data for training certain state parameters of triphones that we consider to be similar. Obviously, after this process central states of similar triphones will have the same parameters.

Fifthly, another important aspect is defining the parameters of the model. Principally we have to decide the appropriate number of states, voice features (LPC, MFCC, etc., and their eventual derivates of first or second order) and the distribution probability for the output function. For the output function we have used the Gaussian mixtures which is a sum of weighted normal distributions. Their main advantage is the reduced number of parameters needed (only mean and variance) but the number of Gaussians must be determined experimentally as shown in Section 4.

In order to deal with the above issues, the following steps have been designed for training:

First step: isolated monophones training using the small monophones database. The alternative to isolated monophones training is flat starting, consisting in calculating global parameters for all speech utterances and using the resulting models as initial HMMs.

Second step: isolated words training using monophones as speech units. Several parameter variations have been made at this step. First of all speaker independent (by using the whole database) and speaker dependent (by using the database of a single speaker) training have been made. Then several numbers of states and mixtures have been tried and results are shown in Section 4. Most of the experiments have been made at this stage because of time constraints. Making these experiments in the further states would take more time because step three implies that step one and two are already completed.

Third step: isolated words training using triphones as speech units. By using the best models obtained at step two (the ones that have the best recognition rate), we have built triphone models by cloning the model of the central phone. The tied-state technique is also used for the central states of similar triphones.

Fourth step: continuous speech training is performed by using the models obtained at step three. At this point both isolated words database and continuous speech database are used. We needed to differentiate two kinds of silences. Silence at the beginning of the phrase or between phrases is modeled differently from the silence between words. The second type of silence is shorter and its model is obtained from the first one by cloning just one central state, resulting in a one-emitting-state HMM.

Fifth step: The last step that is not implemented yet is to add grammar. However, some tests have been performed by using isolated words database and a simple grammar. In this context, grammar must be understood as a set of restrictions that do not allow any sequence of words to be recognized. This is very useful in applications were the dialogue between human and machine is limited to a restricted vocabulary.

4. Experimental Results

The following paragraphs summarize the ASR rates for the monophones database and the isolated words database. Due to the small amount of data in the monophones database we have used the same files (all of them) for both the training and the testing processes. In the case of isolated words database we have selected 9000 files for the training process and we have used the other 1000 for the testing process.

The first results (obtained after training step one – Section 3) were for the isolated monophones database.

As described in Section 3 this step is only an initialization step needed for isolated words training thus we have computed the results only for the basic HMM system: monophones models with six states (among which four states are emissive), two Gaussian mixtures per state, twelve Mel Frequency Cepstral Coefficients (MFCC) with appended energy coefficient and first order derivates.

MFCC_E stands for Mel Frequency Cepstral Coefficients (twelve coefficients) with an appended energy coefficient (the thirteenth coefficient); MFCC_E_D stands for Mel Frequency Cepstral Coefficients with appended energy coefficient and the first order

derivates for these thirteen coefficients; and MFCC_E_D_A stands for Mel Frequency Cepstral Coefficients with appended energy coefficient and the first and second order derivates for the thirteen coefficients.

The recognition rates are presented in Table 3. The HMM recognition system initialized as mentioned above was afterwards trained with all the training files in the isolated words database, resulting a speaker independent ASR system. For comparison, we have also trained five speaker dependent ASR systems (one ASR for each of the speakers in the database). The recognition rates for these basic ASR systems are presented in Table 4.

TABLE 3. Isolated Monophones Recognition Rates

Symbol	Rec. Rate [%]	Symbol	Rec. Rate [%]	Symbol	Rec. Rate [%]	Symbol	Rec. Rate [%]
@	80	g	88	k	100	s1	84
a	78	h	92	l	74	s	89
b	87	i1	-	m	80	t1	90
d	78	i2	77	n	71	t	87
e1	92	i3	95	o1	100	u	81
e	77	i	89	o2	-	v	78
f	85	j	91	o	100	w	87
g1	92	k2	-	p	100	y	-
g2	-	k1	97	r	92	z	77

TABLE 4. Speaker dependent/independent systems

ASR system id	Speaker	HMM	Language restrictions	Rec. Rate [%]
011	all speakers	• models for monophones • 6 states • 2 Gaussian mixtures • MFCC_E_D • initialization: step 1	• no grammar restrictions • full word dictionary	48.93
019	1			77.67
020	2			78.07
021	3			69.42
022	4			74.45
065	5			71.43

Table 4 shows that, as expected, the recognition rate for the speaker independent ASR is smaller than the recognition rate for the speaker dependent ASR. In our opinion this issue can mainly be overcome by enlarging the database. As the embedded training technique does not issue the best results after the first training/testing iteration of Forward/Backward algorithm, several training/testing iterations have to be employed until the recognition rate for the ASR system saturates. Figure 2 presents the recognition rate variation over several training/testing iterations for ASR system 011.

Fig. 2. Recognition Rate Variation

As Table 5 shows and as described in Section 3, creating and training models for triphones implies creating and training models for monophones. Taking this into consideration and also due to time reasons we have decided it is better to focus on finding the HMM models for monophones that best suite our training database (and thus obtain better recognition rates), and only afterwards create and train HMM models for triphones. Obviously, we will only create and train models for triphones starting with the monophones models that had the best recognition rates. The tests for finding the best monophones models involved modifying the number of states in the HMMs, modifying the number of Gaussian mixtures per state and trying different types of coefficients.

TABLE 5. Varying the type of models: for monophones / for triphones

ASR system id	**Speaker**	**HMM**		**Language restrictions**	**Rec. Rate [%]**
011	all speakers	• 6 states • 2 Gaussian mixtures • MFCC_E_D	• models for monophones • initialization: step 1	• no grammar restrictions • full word dictionary	48.93
019	1				77.67
042	all speakers		• models for triphones • initialization: HMMs for monophones		73.21
038	1				90.05

Table 6 presents the recognition rates for ASR systems that differ by the number of coefficients. A lot of tests were performed in order to determine the optimal number of states for the models and the number of Gaussian mixtures per state. Tables 7 and 8 summarize some of the tests results.

TABLE 6. Varying the type of coefficients

ASR system id	Speaker	HMM		Language restrictions	Rec. Rate [%]
011	all speakers	• models for monophones • 6 states • 2 Gaussian mixtures	MFCC_E_D	• no grammar restrictions • full word dictionary	48.93
018			MFCC_E_D_A		35.00
048			MFCC_E		19.66

TABLE 7. Varying the number of states per HMM

ASR system id	Speaker	HMM		Language restrictions	Rec. Rate [%]
011	all speakers	• models for monophones • 2 Gaussian mixtures • MFCC_E_D	6 states	• no grammar restrictions • full word dictionary	48.93
013			7 states		47.52
014			8 states		56.40
051			9 states		63.77
055			10 states		67.79
057			11 states		69.67

TABLE 8. Varying the number of Gaussian mixtures per HMM state

ASR system id	Speaker	HMM		Language restrictions	Rec. Rate [%]
011	all speakers	• models for monophones • 6 states • MFCC_E_D	2 Gaussian mixtures	• no grammar restrictions • full word dictionary	48.93
016			3 Gaussian mixtures		56.04
017			4 Gaussian mixtures		35.23
029	2		1 Gaussian mixture		78.07
020			2 Gaussian mixtures		78.07
059			3 Gaussian mixtures		76.36

Several conclusions arise from the previous tables. It is obvious that increasing the number of states in a monophone model has issued only better results so far. At this time several tests are still being done to decide if the optimum number of states has been reached or not. The good results for the systems with high number of states are somehow surprising because the literature [13] shows that the smallest word error rate for small vocabularies is obtained for Bakis model with only six states.

Secondly, it is obvious that the optimum number of Gaussian mixtures for the speaker independent ASR system and for the given database is three. Also, it is clear that for a speaker dependent ASR system this parameter is less important and could be set to one in order to optimize the computation time. Here, the results are not surprising at all:

- For a speaker dependent system one Gaussian mixture is enough to model a single voice. Even though the monophones are not uttered exactly the same, they are very similar and so are the cepstral coefficients for every state in the model.
- For a speaker dependent system (and in our case only five speakers in the training database) three Gaussian models are needed to best model the five different voices. The monophones are uttered in a particular manner by every speaker and thus the cepstral coefficients are also quite different. We expect that for a database with more than five speakers the optimum number of Gaussian mixtures to be bigger.
- As the HMMs for the monophones better model the training database, the recognition rate difference between the speaker independent and speaker dependent ASR systems gets smaller and should become in the end insignificant.

The best results for the monophone models that have been obtained so far (at least the number of states for the models is still to be fine-tuned) are presented in Table 9.

TABLE 9. Best Recognition Result

ASR system id	**Speaker**	**HMM**	**Language restrictions**	**Rec. Rate [%]**
054	all speakers	• models for monophones • 10 states • 3 Gaussian mixtures • MFCC_E_D	• no grammar restrictions • full word dictionary	81.69

Although the timing is not necessarily right (the best model for monophones has not been found yet), some language restrictions tests have also been performed in order to see if the results obtained are significantly improved or not. Table 10 presents encouraging results.

TABLE 10. Varying the language restrictions

ASR system id	**Speaker**	**HMM**		**Language restrictions**	**Rec. Rate [%]**
019	1	• 6 states • 2 Gaussian mixtures • MFCC_E_D	• models for monophones • initialization: step 1	• no grammar restrictions • full word dictionary	77.67
038			• models for triphones • initialization: HMMs for monophones		90.05
039				• simple grammar • full word dictionary	94.68
040				• simple restrictions • short dictionary	98.49

"Simple grammar" means that the testing process "was told" that only a single word has been uttered in every audio file, while "no grammar restrictions" means that the testing process "was told" that any combination of any words in the dictionary could have been uttered in the audio files. The difference between full and short dictionary resides in the number of words that compose the dictionary. In the first case the ASR system could choose from any of the 10000 words that make up the isolated words, while in the second case the ASR system is restricted to choose only from a subset of 1000 words.

5. Conclusions and future work

The results presented in the previous section show that the proposed training strategy works and issues better and better results as the training/testing steps are being made. Summarizing the results several conclusions can be drawn:

- ASR systems with triphone models are better than those with monophone models.
- The optimum number of Gaussian mixtures per HMM state is three (in a speaker independent ASR system and for this particular database).
- MFCC_E_D coefficients used as voice features for the HMMs issued the best results.
- Language restrictions are very important even in isolated words ASR systems.

Future work is intended to firstly fine-tune the ASR system. We still have not reached a conclusion regarding the optimum number of states for the system and there are also some other important methods that have not been tried yet: creating different models for different phonemes based on their linguistic characteristics, using other voice features (coefficients) for recognition, creating models for syllables instead of phonemes.

Secondly, we intend to continue our efforts in creating a continuous speech database by direct recording and use it as specified in Section 3 to create a continuous speech ASR system. This would yield two types of results: recognition rates for isolated words and recognition rates for continuous speech.

Finally, we expect the results for continuous speech to be worse than for isolated words without proper language restrictions. Therefore we intend to use a solid grammar in order to overcome this issue.

Acknowledgments. The research reported in this paper was funded by the Romanian Government, under the National Research Authority CNCSIS grant "IDEI" no. 114/2007, code ID_930.

References

[1]. Mihaela Ioniță, C. Burileanu, M. Ioniță, „DTW Algorithm with Associated Matrix for a Passworded Access System", Emerging techniques for communication terminals (ENSEEIHT), pp. 265 – 270, 1997.

[2]. B. Sabac, Z. Valsan, I. Gavat, „Isolated Word Recognition Using the DTW Algorithm", in Proceedings of the International Conference COMMUNICATIONS '98, pp. 245 – 251, 1998.

[3]. C. Burileanu, V. Teodorescu, G. Stolojanu, C. Radu, „Sistem cu logică programată pentru recunoaşterea cuvintelor izolate", independent de vorbitor, Inteligența

artificială şi robotica, Editura Academiei Române, Bucureşti, vol. 1, pp. 266 – 274, 1983.

[4]. Diana Militaru, Inge Gavăt, Octavian Dumitru, Tiberiu Zaharia, Svetlana Segărceanu, „Protologos, System for Romanian Language Automatic Speech Recognition and Understanding", Proceedings of Sped 2009, pp. 21 – 32, 2009

[5]. D. Munteanu, "Contribuţii la realizarea sistemelor de recunoaştere a vorbirii continue pentru limba română", Teză de doctorat, 2006.

[6]. D. Burileanu, M. Sima, C. Burileanu, V. Croitoru, „A Neuronal Network-Based Speaker-Independent System for Word Recognition in Romanian Language", Text, Speech, Dialogue - Proc. of the First Workshop on Text, Speech, Dialogue, pp. 177 – 182, 1998.

[7]. J. Domokos, „Contribuţii la recunoaşterea vorbirii continue şi la procesarea limbajului natural", Teză de doctorat, 2009.

[8]. H.-N. Teodorescu, L. Pistol, M. Feraru, M. Zbancioc, D. Trandabăţ, "Sounds of the Romanian Language Corpus", © 2010, http://www.etc.tuiasi.ro/sibm/ romanian_spoken_language/index.htm.

[9]. Eugeniu Oancea, Doru Munteanu, Corneliu Burileanu, Continuous Speech Recognition System Improvements, Proc. of the 3rd Conference on Speech Technology and Human – Computer Dialogue "SpeD 2005", Editura Academiei Române, Bucureşti (2005) 81-91.

[10]. Yi Liu, Pascale N Fung, Decision tree-based triphones are robust and practical for Mandarian speech recognition, Proceedings of the 6th European Conference on Speech Communication and Technology, Eurospeech (1999), 895-898.

[11]. Klaus Beulen, Elmar Bransch, Hermann Ney, State tying for context dependent phoneme models, Eurospeech (1997) 1179-1182.

[12]. Darryl Stewart, Ming Ji, Hanna Ming, Smith Philip, F. Jack, A state-tying approach to building syllable HMMs, ICSLP (2002) 2649-2652.

[13]. Lawrence R. Rabiner, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proceedings of the IEEE, vol. 77, no. 2 (1989) 257-286.