Research Article **Open Access**

Naima Zerari*, Samir Abdelhamid, Hassen Bouzgou, and Christian Raymond

Bidirectional deep architecture for Arabic speech recognition

https://doi.org/10.1515/comp-2019-0004
Received July 20, 2018; accepted March 4, 2019

Abstract: Nowadays, the real life constraints necessitates controlling modern machines using human intervention by means of sensorial organs. The voice is one of the human senses that can control/monitor modern interfaces. In this context, Automatic Speech Recognition is principally used to convert natural voice into computer text as well as to perform an action based on the instructions given by the human. In this paper, we propose a general framework for Arabic speech recognition that uses Long Short-Term Memory (LSTM) and Neural Network (Multi-Layer Perceptron: MLP) classifier to cope with the non-uniform sequence length of the speech utterances issued from both feature extraction techniques, (1) Mel Frequency Cepstral Coefficients MFCC (static and dynamic features), (2) the Filter Banks (FB) coefficients. The neural architecture can recognize the isolated Arabic speech via classification technique. The proposed system involves, first, extracting pertinent features from the natural speech signal using MFCC (static and dynamic features) and FB. Next, the extracted features are padded in order to deal with the non-uniformity of the sequences length. Then, a deep architecture represented by a recurrent LSTM or GRU (Gated Recurrent Unit) architectures are used to encode the sequences of MFCC/FB features as a fixed size vector that will be introduced to a Multi-Layer Perceptron network (MLP) to perform the classification (recognition). The proposed system is assessed using two different databases, the first one concerns the spoken digit recognition where a comparison with other related works in the literature is performed, whereas the second one contains the spoken TV commands. The obtained results show the superiority of the proposed approach.

Keywords: Arabic ASR, digits, command TV, speech recognition, MFCC, delta-delta, FB, deep learning, LSTM, GRU, MLP

1 Introduction

Speech is one of the most direct means of information exchange used by human being. This advantage has given rise to several developments where the aim is the design of systems to recognize spoken words. Automatic Speech Recognition (ASR) is an active area of study allowing the communication between human and machine. It is the process of understanding the human speech by a computer. In this context, Automatic Digit/Command Recognition is considered as one of the most challenging domains in ASR. The growing importance of Digit/Command recognition is mainly due to the increasing demand for applications that deal with human-machine interaction through natural languages such as command systems via pronounced digit [1, 2].

The implementation of these kinds of systems requires a particular process for the speech signal to provide reliable features that can recognize properly the input spoken words. Therefore, wide range of techniques have been proposed in the literature to represent the speech signal [3]. The most commonly used one, is the Mel-Frequency Cepstral Coefficients (MFCC), which is a popular technique that attempt to mimic some parts of the human speech perception and speech production [4].

In the present study, the obtained MFCC coefficients of the spoken utterances will be introduced to a Long Short-Term Memory (LSTM) architecture [5], which treats the general sequence-to-sequence problems. The idea is to use a bidirectional LSTM layer included in the deep architecture to encode the sequence as a fixed size vector, then this vector will be fed to a Multi-Layer Perceptron (MLP)

***Corresponding Author: Naima Zerari:** Laboratory of Automation and Manufacturing, Department of Industrial Engineering, University of Batna 2 Mostefa Ben Boulaid, Batna, 05000, Algeria; E-mail: n.zerari@univ-batna2.dz

Samir Abdelhamid: Laboratory of Automation and Manufacturing, Department of Industrial Engineering, University of Batna 2 Mostefa Ben Boulaid, Batna, 05000, Algeria; E-mail: s.abdelhamid@univ-batna2.dz

Hassen Bouzgou: Department of Industrial Engineering, University of Batna 2 Mostefa Ben Boulaid, Batna, 05000, Algeria; E-mail: h.bouzgou@univ-batna2.dz

Christian Raymond: INSA Rennes, IRISA/INRIA, Rennes, France; E-mail: christian.raymond@irisa.fr

classifier to carry out the recognition task. The whole proposed model is trained to perform two recognition tasks: 1) digit recognition and 2) Command TV recognition. The proposed system with MFCC features is compared with Filter Banks features applied to command TV corpus.

The remainder of the paper is organized in six sections as follows: Section 2 highlights some related works. Section 3 explains the methodologies proposed in this study. Section 4 presents the data used to validate the proposed methodology. Section 5 presents the performance criteria used to evaluate the proposed model, presents the experimental results obtained on the two datasets and compares with other existing approaches in the literature. Finally, section 6 draws the conclusion of this work.

2 Related works

Arabic is the official language of twenty-five countries. It denotes a semitic language and one of the oldest languages in the world. Different studies have been investigated in the literature to propose recognition systems using different approaches [6–8]. However, compared to other languages such as English, the number of research papers in Arabic language is limited. In what follows, some studies concerning ASR systems for Arabic language will be discussed.

In [9], the authors proposed a Speech-And-Speaker (SAS) identification system based on spoken Arabic digit recognition. They treated the speech signals as an image object and used the algorithm of Teplitz matrix minimal eigenvalues as a feature extraction method and both conventional and Artificial Neural Networks methods for classification.

An automatic discrete speech recognition system based on a tree distribution classifier has been described in [10]. The MFCC feature extraction method was used to extract features followed by a Vector Quantization method (VQ). The VQ output was provided as an input to a classifier, which deliver the class-label according to each feature using an optimal spanning tree model in order to approximate the true class probability.

A fast learning method with a graphical probabilistic model for discrete speech recognition based on spoken Arabic digit recognition is introduced in [11]. The authors proposed a method based on spanning tree structure takes advantage of the temporal nature of speech signal. The obtained results suggests that the proposed method was efficient in terms of time computation than the state-of-the-art algorithms that use the Maximum Weight Spanning Tree (MWST).

An arabic digits classifier system with 450 Arabic spoken digits has been proposed in [12]. The system is based on combining Wavelet Transform with Linear Prediction Coding (LPC) method to extract the features and the Probabilistic Neural Network (PNN) for classification. The proposed classifier provided a high recognition rate, reaching about 93% of accuracy based on a speaker-independent system.

Whereas, in [13], the authors used Sphinx tools to recognize isolated Arabic digits with data issued from six different speakers. The system realized a digits recognition accuracy of 86.66%.

Recently, excellent performances on these systems have been achieved using Deep Neural Networks (DNNs) which are recent and extremely powerful machine learning models [14].

An important study has been presented in [15]. The authors used an end-to-end deep Recurrent Neural Networks (RNN) model with suitable regularization. They concluded that Recurrent Neural Network, more precisely Long Short-Term Memory reach a test error of 17.7% on the TIMIT phoneme recognition benchmark.

However, a Deep Belief Networks (DBN) has been used for a development of a novel context-dependent model for Large-Vocabulary Speech Recognition (LVSR) for phone recognition in [16]. The obtained results outperform significantly the conventional context-dependent Gaussian Mixture Model- Hidden Markov Models GMM-HMMs.

A success of applying DNNs for acoustic modeling in speech recognition has been described in [17]. The authors proposed a new approach to train deep neural networks. They have shown that the proposed model outperforms the Gaussian Mixture models on a variety of speech recognition benchmarks, sometimes by a large margin.

Other researches have used the KALDI toolkit for the development of a recipe and language resources for training and testing Arabic broadcast news speech recognition systems [18]. The authors described in detail the decisions made to build the system using the MADA toolkit. They give results in terms of Word Error Rate (WER), where the broadcast news system obtained 15.81% WER on Broadcast Report and 32.21% WER on Broadcast Conversation, with a combined WER of 26.95%.

An Arabic Multi-Genre Broadcast (MGB-2) Challenge for Spoken Language Technology 2016 has been presented [19]. The authors focused on handling the diversity in dialect in arabic speech. The audio data used are from 19 distinct programs recorded from Aljazeera Arabic TV channel in the period (2005-2015). The authors divided the research

into two tasks: standard speech transcription, and word alignment. On the first task, the baseline WER was 34%, however, on the second one, the baseline system obtained a precision of 0.83 and 0.7 as recall.

Also, an Arabic Multi-Genre Broadcast 3 (MGB-3) Challenge – Arabic Speech Recognition in the Wild has been described in [20]. The MGB-3 emphasises dialectal Arabic using a multi-genre collection of Egyptian YouTube videos. It comprised two tasks: (1) Speech transcription, evaluated on the MGB-3 test set, (2) Arabic dialect identification, introduced in order to distinguish between four major Arabic dialects. The most accurate result on the MGB-3 test set was 37.5% average WER and 29.3% multi-reference WER. The authors reported that the obtained results outperform those obtained by MGB-2.

A description of a system that participated in the broadcast news evaluation for Arabic has been presented in [21]. The authors have shown how to build a phonetic system. They demonstrated that switching to phonetic models is capable to reduce the word error rate by up to 14% compared to the traditional grapheme based approach.

In [22], the authors described the JHU team's Kaldi system submission of the Arabic MGB-3. They used an architecture neural network in the form of Time-Delay Neural Network/Long Short Term Memory (TDNN-LSTM) trained using Lattice-Free Maximum Mutual Information (LF-MMI). The authors reported that their primary submission to the challenge gives a multi reference WER of 32.78% on the MGB-3 test set.

3 Methodology

The speech signal is not an ordinary signal, it represents a complex phenomenon. This complexity is due to its statistical properties, which varies over time [1]. An ASR system takes an audio signal as input and classifies it into a set of words. In order to allow the ASR system to realize its task, it is important to extract and deliver reliable features using feature extraction technique. One of the well known techniques is the Mel Frequency Cepstral Coefficients [23]. Generally, Automatic Speech Recognition researchers investigated spoken alphabets, digits, commands with different languages [7].

In this paper, we propose to recognize a set of isolated Arabic utterances issued from two ASR applications, namely: spoken digit recognition and TV spoken command recognition. The proposed automatic recognition system is composed from several modules namely: signal acquisition, feature extraction, corpus construction, model training and finally a classification, all modules are illustrated in Figure 1.

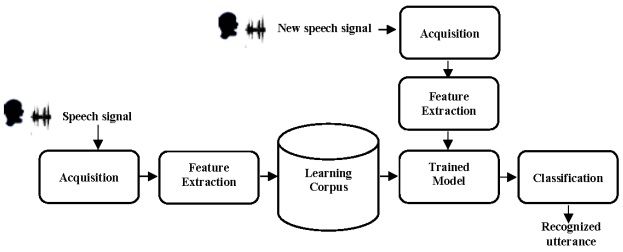


Figure 1: Block diagram of the proposed ASR architecture.

3.1 Acquisition

The first module denotes the acquisition module employed to record the different speech utterances (digits / TV commands) using a microphone. A recorded audio signal conveys, not only the speech, but also an additive noise issued from the recording environment. It should be noted that, for each dataset, all speech signals were recorded in a natural environment under similar setting conditions, which are the same length of recording time, the same sampling frequency and the same recording microphone. Next, the silence recorded with the speech is removed (filtered) leading to new wave forms with different sizes. The different parameters used by the acquisition module are illustrated in Table 1.

Table 1: Recording parameters.

Parameters	Values
Sampling rate	16000 Hz, 16 bits
Number of bits	16 bits
Time	2 s

3.2 Feature extraction with MFCCs

To design an ASR system, it is very important to select the best parametric representation of acoustic data. The common purposes in selecting the best representation are to compress the speech data and eliminate information not pertinent for recognition of speech. Several techniques are defined in literature and used by researchers. For a long time, Mel-Frequency Cepstral Coefficients was the most popular and used features extraction technique [24].

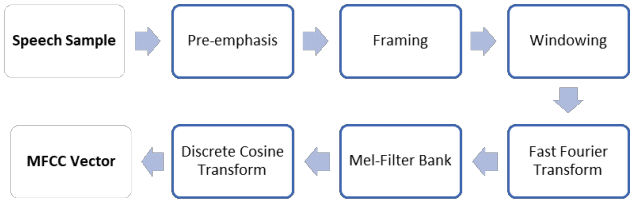


Figure 2: MFCC block diagram.

This technique is used to approximate the nonlinear frequency resolution of the human ear using the following formula [25]:

$$Mel(f) = 2595 * \log_{10}(1 + f/700) \tag{1}$$

MFCC technique gives a valuable representation of the speech signal by extracting the significant information from it. The different steps involved in the conventional MFCC parameterization techniques are: 1) Pre-emphasis, 2) Framing, 3) Windowing, 4) Fast Fourier Transform (FFT), 5) Mel scale Filter Bank, and 6) Discrete Cosine Transform (DCT) as shown in Figure 2. Thus, after signal digitization, a pre-emphasis step is performed to increase the amplitude of high frequency bands and decrease that of lower frequency bands. Next, Discrete Fourier Transform is applied to extract the spectral information of each windowed frame speech signal. Then, the latest result is passed through a bank of triangular Mel filters, which provide a natural logarithm of the filter bank energies. Finally, a DCT is used to decorrelate the log energies output of a filter banks [26–28].

3.3 Dynamic features of MFCC

The MFCC features have been used effectively in a range of speech processing systems[1],where they provide a significant features of the speech spectra. However, the speech is a natural dynamic signal varying in time. Hence, it is necessary to use a representation that contains some aspects of the dynamic nature of the speech signal [29]. Here, the MFCC derivatives can be an appropriate mean to get these features. The first derivative is called delta coefficient and the second order derivative is called delta-delta coefficient [30]. Therefore, the delta and delta-delta coefficients are added to the original MFCCs. This addition can significantly improve speech recognition performance by removing the distortion effects using the differencing operation [31, 32].

The delta features are computed from the static features using the following formula [32]:

$$d_t = \frac{\sum_{n=1}^{N} n(C_{t+n} - C_{t-n})}{2\sum_{n=1}^{N} n^2} \tag{2}$$

where d_t is a delta feature, i.e., dynamic coefficient at time t computed in terms of the corresponding static coefficients C_{t-n} to C_{t+n}. Whereas, the N value is, in general, equal to 2.

Similarly, the delta-delta coefficients are calculated using the same formula (2) by replacing C_t with d_t (features with delta features).

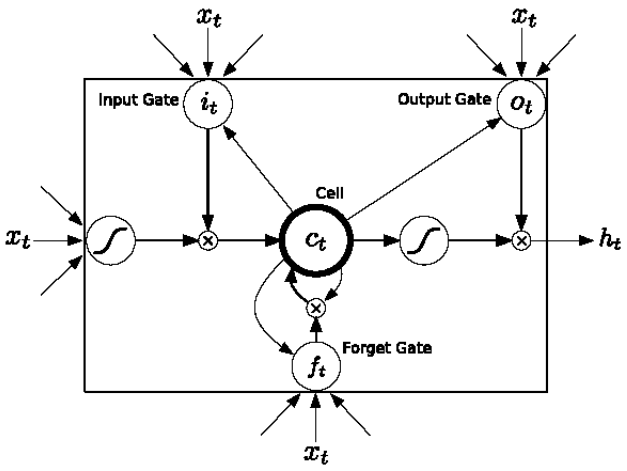
3.4 Recurrent neural networks and gated recurrent neural networks

In this subsection, we describe the elementary theory of Recurrent Neural Networks (RNN). These later are considered as a powerful models for sequential and time series data [5]. They are trained in a discriminative way, and their internal state provides a powerful, general framework to model time series. Furthermore, they tend to be robust to temporal and spatial noise [33]. RNNs comprise a loop, making them recurrent and permitting information to persist. Simple recurrent neural networks contain just one loop while other, more complex RNN, are composed with one or more gates allowing them to retain and forget information [34].

The success of this type of neural network is due mainly to the specific variant, which are the Long Short Term Memory (LSTM) proposed by *Sepp Hochreiter* and *Jrgen Schmidhuber* in 1997 [5] and the Gated Recurrent Unit (GRU) proposed by *Junyoung Chung et al* in 2014 [35].

The main idea behind these networks is to use several gates to control the information flow from previous steps to the current steps [36]. By employing the gates, any recurrent unit can learn a mapping from one point to another.

LSTM network, which was designed to model temporal sequences and their long-range dependencies more accurately than conventional RNNs, contains three gates: an input gate, an output gate and a forget gate. At each iteration, the three gates try to remember when and how much the information in the memory cell should be updated [36]. A single LSTM memory cell is depicted in Figure 3 [37, 38].



where

- $f_t = \sigma_g(W_f x_t + U_f h_t + b_f)$
- $i_t = \sigma_g(W_i x_t + U_i h_{t-1+b_i})$
- $o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o)$
- $c_t = f_t \circ c_{t-1} + i_t o \sigma_c(W_c x_t + U_c h_{t-1} + b_c)$
- $h_t = o_t \circ \sigma_h(c_t)$
- $\circ$ is Hadamard product

and

- x_t = input vector, h_t = output vector
- c_t cell state vector
- W,U,b: parameter matrices and vector
- f_t, i_t and o_t
  - f_t: Forget gate vector: weight of remembering old information
  - i_t: Input gate vector: weight of acquiring new information
  - o_t: Output gate vector: Output candidate

Figure 3: A long short-term memory cell.

3.5 Classification with multi-layer perceptron

The last step in the ASR process is the classification, where the goal is to classify the input speech utterances, based either on a priory knowledge or statistical information extracted from the speech signals [39].

In the present study, the classifier input is the set of equal-size vectors with low dimensional features delivered by LSTM network. In what follows, a brief presentation of the classifier used in this study is given. A MultiLayer Perceptron (MLP) is a subclass of Artificial Neural Network, widely applied in classification [40].

The MLP architecture is variable, but generally organized in several layers of neurons. It consists of three sequential layers: input, hidden and output layer. In this study, the MLP is used as multi-classifier, where its inputs represent the MFCC (or FB) fixed size vector given by LSTM.

The MLP neural networks classifier acts usually in a supervised manner. To build an MLP classifier, a set of training data including the inputs and their associated outputs are requested. Hence, the classification is done by assigning a maximum value to the output neurons to represent the desired class [6, 41]. Figure 4 represents the MLP architecture used in this study.

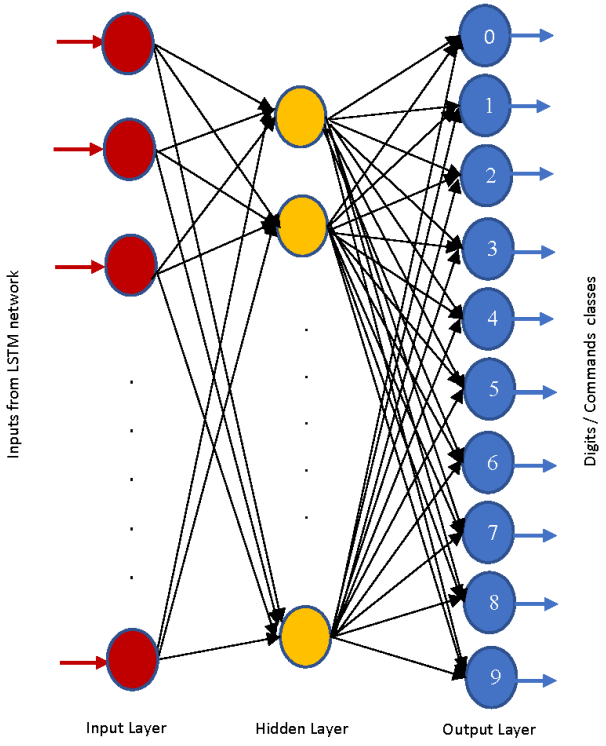


Figure 4: MLP architecture used in the present study.

4 Experimental data

To assess the different proposed learning strategies, we have used two data resources: (1) the digit data set [11] where the authors have introduced a way to speed up the learning of graphical model for speech recognition based on spoken arabic digit and (2) the TV command data set that was created using different individual (male/female) records with different age groups for the two sexes.

4.1 Spoken digit dataset

The first dataset is the spoken Arabic digit which contains time series of MFCCs corresponding to spoken Arabic digits (as show in Figure 5) collected by the laboratory of automatic and signals, University of Badji-Mokhtar - Annaba, Algeria. A number of 88 (44 males and 44 females) arabic native speakers were asked to utter all digits ten times. Accordingly, the database consists of 8800 tokens (10 digits x 10 repetitions x 88 speakers) [42].

Digit	Arabic Writing
0	صفر
1	واحد
2	اثنان
3	ثلاثة
4	أربعة
5	خمسة
6	ستة
7	سبعة
8	ثمانية
9	تسعة

Figure 5: Arabic digit and their writing.

The MFCCs of the spoken digit dataset were computed with the following parameters, illustrated in Table 2.

Table 2: MFCC parameters used for the Spoken digit dataset.

Parameters	Values
Sampling rate	11025 Hz, 16 bits
Filter pre-emphasized	$1\text{-}0.97*Z^{-1}$
Applied window	Hamming

4.2 Spoken command TV dataset

In the second dataset, speaker-independent mode is considered, where one hundred of Arabic native speakers were participated (50 males comprising 37 adults and 13 kids whereas 50 females including 31 adults and 19 kids) to the construction of the corpus as shown in Table 3. The native speakers record ten Arabic commands illustrate in Figure 6 for ten times. Consequently, the dataset contains 10000 tokens (10 arabic commands x 10 repetitions x 100 speakers)

Table 3: Category and gender distribution of the speakers for the Spoken command TV dataset.

Gender	Male		Female		
Category	Adult	Kid	Adult	Kid	Total
Speakers	37	13	31	19	100
Utterances	3700	1300	3100	1900	10000

N	Arabic command	Signification
1	تشغيل	ON
2	اغلاق	OFF
3	أمام	NEXT
4	خلف	BACK
5	رفع	UP
6	خفض	DOWN
7	صامت	MUTE
8	قائمة	LIST
9	خروج	EXIT
10	ايقاف	QUIT

Figure 6: Arabic TV commands.

For the constructed dataset (Arabic commands), the list of parameters used to compute the MFCCs are enumerated in Table 4. Once MFCC algorithm is applied, a numerical values (features) of speech data were obtained and saved in the dataset (Learning corpus).

Table 4: Parameters list of MFCC used in the TV command dataset.

Parameters	Values
Sampling rate	16000 Hz, 16 bits
Filter pre-emphasized	$1\text{-}0.97*Z^{-1}$
Applied window	Hamming
Window Size	256
FFT Size	512
Linear filters	13
logFilters	27
Cepstral coefficients	13

5 Experimental results

The different experiments in this study have been performed using Python programming language and keras/tensorflow libraries [43].

5.1 Performance criteria

To evaluate the performance of the ASR systems, several performance measures can be used [44]. Namely, recall, precision, f- measure (F1), % of error and % of success. These ones are used in this study as the standard classification criteria. They are defined below:

$$precision = \frac{\text{number of correct predictions}}{\text{number of predictions}}$$

$$recall = \frac{\text{number of correct predictions}}{\text{number of samples}}$$

$$F1 = 2 * \frac{(precision * recall)}{(precision + recall)}$$

In the first dataset, 8.800 tokens were used, wheras in the second corpus 10.000 instances were utilized. In both cases, the two datasets were splited in two parts. The first one is used for training while the second one is used to test the different proposed models. The training data contains 70% of the whole dataset and the test data comprises 30% of the whole dataset. This repartition is used to build the learning models using a simple Hold-Out model selection [45].

All models are trained using 50 epochs and the best model on the training set is kept for final evaluation. This is not an optimal model selection technique, since it may privilege the model that have over-fitted the training data, but we expect that the dropout regularization approach counteract this effect. In order to have a better estimation of the model's performances, all experiments are conducted 10 times by averaging the obtained results.

The proposed neural network receives as input the sequences of MFCC features and gives as output the class of the uttered digit/command. First, the LSTM will encode the sequence of MFCC coefficients as a fixed size vector. Then the MLP network receipts this fixed size vector to classify the MFCC coefficients. This can be performed through different steps, as follow:

- Encoding data as a matrix:
  For instance in the second dataset (Command TV), the data are encoded as a matrix of (7000,198,13) where 7000 denotes the size of samples in the training, 198 represents the size of the longest sequence of MFCC coefficients (corresponding to the longest recorded sequence) and 13 designates the MFCC's coefficients number used in this study. In case where the size of the sequence does not reach 198, the sequence is padded by a zeroed vector of size 13 until attaining the maximum size of 198.
- Encoding sequence as a fixed size:
  The bidirectional LSTM layer receives the data resulting from the previous step and tries to encode the sequence as a fixed size vector. The choice of bidirectional approach is justified by the results obtained in earlier studies [34, 46].
- Classification with MLP:
  The vector resulting from LSTM encoding is fed to the MLP structure with one hidden layer. The various parameters have been fixed intuitively as follow:
  1. The recurrent layer's output is set to 100. 100 output neurons is fixed for forward or backward recurrent layers and 2*50 for bidirectional models whose outputs are concatenated to obtain the final output vector.
  2. The hidden layer size has been fixed to 50 with Rectified Linear Unit "ReLU" as a non-linear activation function.
  3. The output layer size is defined by the number of classes (10 classes: Digits/ Commands) using a standard "softmax" activation function with cross entropy loss.
     With the purpose to regularize the network, the Dropout technique is implemented. This technique allows to temporarily remove (hidden and visible) units from the network, along with all its incoming and outgoing connections, as shown in Figure 7.
     The Dropout technique is used to prevent the overfitting where the choice of which units to drop is random. Thus, two Dropout layers are inserted respectively after the LSTM output and the MLP hidden layer with a dropout probability of 0.2 and 0.5 respectively [47].

Figure 8 illustrates an example of neural network architecture with Bidirectional LSTM topology before and after dropout operation.

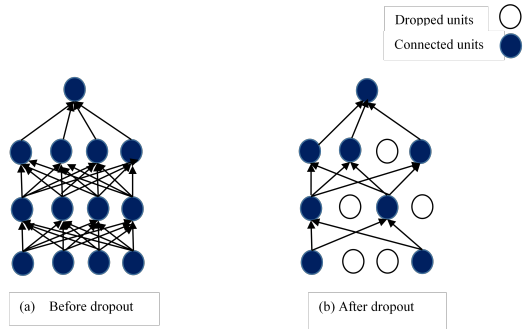


Figure 7: Neural network architecture: (a) before dropout and (b) after dropout.

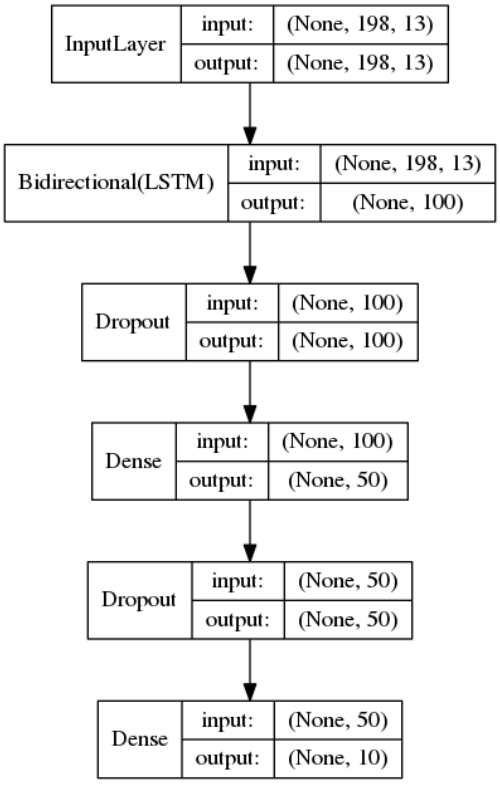


Figure 8: Block scheme of the proposed model.

5.2 Results obtained on digit corpus

To approve the choice of the bidirectional recurrent architecture as encoder, different encoders using both combination of GRU and LSTM layers with forward/backward and bidirectional encoding strategies are evaluated in terms of performance. We fix the size of the vector that encode the sequence to 100.

In the present study, the proposed architecture takes as input the sequences of MFCC features and as output the class of the spoken digit. First, the network will encode the sequence of MFCC coefficients as a fixed size vector. This fixed sized vector will feed an MLP network to classify the MFCC coefficients. To do so, we, first, encode the data as a matrix of (6600,93,13) where 6600 is the size of samples in our training data, 93 is the size of the longest sequence of MFCC coefficients (corresponding to the long time of recording) and 13 is the number of MFCC coefficients used in this study. When the size of the sequence is smaller than 93, the sequence is padded by a zeroed vector of size 13 until reaching the maximum size of 93. The obtained results are compared with those in [10] and [11], as shown in Table 5, in terms of the following performance criteria: precision, recall, F1 and % error.

5.3 Results obtained on TV commands corpus

To evaluate the different strategies using FB, MFCC and MFCC + double delta, practically the same approaches are maintained and are listed below:

- Bidirectional LSTM of size 50;
- Bidirectional GRU of size 50;
- Bidirectional GRU of size 67;
- Bidirectional GRU of size 80;
- Forward LSTM of size 100;
- Forward GRU of size 100.

All encoder variants show good results. The performed experiments confirm that bidirectional architectures are more efficient than single direction ones.

The obtained results by the FBs using the same parameters as those used in MFCC's experiments are less effective. Since, in the case of FBs, there are 40 features instead of 13 in the case of MFCC, the network may need more parameters to learn from them. We did the experiment increasing the size of the GRU embedding to 100 and the next hidden layer from 50 to 75 and report results in Table 7. Indeed, with more parameters the networks is much more effective, anyway it still less efficient using FB than MFCC as input while using twice more parameters. Also, the experiments done with double delta features (39 features), the network need more parameters to learn from them.

Thus, It turns out that filter bank coefficients computed in the early steps of MFCCs are highly correlated [48], which could negatively affect the accuracy of machine learning algorithms (see Table 8).

Hence, Discrete Cosine Transform is used to decorrelate the filter bank coefficients and produce a compressed representation of the filter banks. Typically, for Automatic Speech Recognition, the resulting cepstral coefficients (in our case 13) are maintained and the rest are discarded, in this case the results are reported in Table 9.

Table 10 illustrates the different results obtained using the dynamic features that provide more information about the signal evolution (delta delta features) added to the static ones (MFCC).

As it is shown, the delta-delta results (compare Tables 8, 9 and 10) outperform those obtained using the standard MFCCs and FBs, this proves the utility to use the delta-delta operators to the MFCCs which can affects positively the classification accuracy.

It is worth noting that the experiments done with double delta features (39 features) need more parameters to learn the network, which increase the computational complexity accordingly (see Table 10).

Table 5: Results and comparison of the proposed approach with some previous published approaches on the digits dataset.

(a) Results of the bidirectional approach.

dig.	precision	recall	F1	%error
0	95.98	98.73	97.33	4.14
1	98.92	99.86	99.39	1.09
2	99.63	98.91	99.27	1.09
3	98.67	97.45	98.06	2.55
4	99.64	99.32	99.48	0.68
5	99.32	99.91	99.61	0.68
6	99.81	96.36	98.06	3.64
7	98.55	98.82	98.68	1.45
8	98.32	98.41	98.36	1.68
9	98.96	99.91	99.43	1.05
All	98.77	98.77	98.77	1.23

(b) Comparison with the approaches by [10] and [11] in terms of % success.

dig.	[10]	[11]	our BiLSTM
0	91.00	85.55	**95.86**
1	**99.00**	98.36	98.91
2	91.50	92.91	**98.91**
3	88.00	94.09	**97.45**
4	81.50	89.91	**99.32**
5	94.50	94.00	**99.32**
6	84.50	93.82	**96.36**
7	89.50	90.18	**98.55**
8	92.50	99.00	**98.32**
9	91.00	93.36	**98.95**
All	90.35	93.12	**98.77**

Table 6: Averaged results over 10 experiments on digits corpus with different encoders.

Type of encoder	#params	F1	%error
*BiLSTM 2*50*	31.560	98.77	1.23
*BiGRU 2*50*	25.060	98.63	1.37
Forward GRU 100	40.060	97.26	2.74
Backward GRU 100	40.060	98.85	1.15
Forward LSTM 100	51.560	97.41	2.59
Backward LSTM 100	51.560	98.33	1.67

Table 7: Results obtained with more parameters on the TV commands corpus with bidirectional GRU using FBs.

type of encoder	#params	F1	%error
*BiGRU 2*100*	100,435	95.7	4.3

Table 8: Averaged results over 10 experiments on TV commands corpus with different encoders using FB.

Type of encoder	#params	F1	%error
*BiLSTM 2*50*	41.960	81.3	18.70
*BiGRU 2*50*	32.860	84.8	15.20
*BiGRU 2*67*	50.676	87.1	12.90
*BiGRU 2*80*	66.640	88.6	11.40
ForwardLSTM 100	61.960	79.13	20.87
ForwardGRU 100	47.860	85.26	14.73

In summary, in the case of the digit dataset, all encoder variants exhibit good results and outperform those by [10] and [11] by at least 5% of accuracy. Whereas, for the constructed dataset (TV commands corpus), all encoders show good and comparable results (see Table 6) where the global performance is over 95% for all models. We may note that backward models reveal sometimes difficulties

Table 9: Averaged results over 10 experiments on TV commands corpus with different encoders using MFCC.

Type of encoder	#params	F1	%error
*BiLSTM 2*50*	31.560	96.23	3.77
*BiGRU 2*50*	25.060	96.14	3.86
*BiGRU 2*67*	40.224	96.93	3.07
*BiGRU 2*80*	54.160	97.06	2.96
ForwardLSTM 100	51.560	97.03	2.97
ForwardGRU 100	40.060	97.11	2.89

Table 10: Averaged results over 10 experiments on TV commands corpus with different encoders using MFCC+ double Delta features.

Type of encoder	#params	F1	%error
*BiLSTM 2*100*	133.110	97.36	2.64
*BiGRU 2*100*	105.110	97.66	2.34
ForwardLSTM 200	217.330	97.23	2.77
ForwardGRU 200	169.330	97.27	2.73

to converge and when they converge exhibits lower performances (still over 95%). The fact that backward models are less efficient on this task may explain the relative equivalent performances between forwards and bidirectional model for an equivalent number of parameters.

6 Conclusion

In this study, an approach based on recurrent neural networks to process sequences of variable lengths of (1) MFCCs, (2) FBs and (3) delta-delta features of the different spoken digits/commands was presented. The extracted

features using the different techniques are, first, encoded as a fixed size vector by a recurrent LSTM/GRU neural network, next, a standard Multi-Layer Perceptron is used to classify the spoken digits/commands with the obtained vector as input. The efficiency of the proposed methodologies is confirmed through the results and discussions presented in this paper.

In all the experiments carried out on the two datasets, the proposed system presents an improved performance and obtains promising results. The obtained results show that Delta-delta features (introduced to a classification system) are efficient enough to characterize the speech signal for the two studied tasks (Accuracy over 96%) compared to those obtained using FBs and MFCCs as feature extraction techniques.

The Challenge for future works is to assess this kind of systems with other datasets constructed through a recorder speech signals in a noisy (more realistic) environments.

Acknowledgements: The authors express their gratitude to the dedicated personnel who made the Arabic digits dataset used in this study freely available and all the participants in the recording of the Command TV dataset. They are also grateful for the support of NVIDIA Corporation with the donation of the GTX Titan X GPU used in this research work.

References

[1] Rabiner L. R., Juang B. H., Fundamentals of speech recognition, PTR Prentice Hall Englewood Cliffs, 1993

[2] Jelinek F., Statistical methods for speech recognition, MIT press, 1997

[3] Desai N., Dhameliya K., Desai V., Feature extraction and classifcation techniques for speech recognition: A review, International Journal of Emerging Technology and Advanced Engineering, 2013, 3(12), 367–371

[4] Ittichaichareon C., Suksri S., Yingthawornsuk T., Speech recognition using mfcc, International Conference on Computer Graphics, Simulation and Modeling, 2012, 28–29

[5] Hochreiter S., Schmidhuber J., Long short-term memory, Neural computation, 1997, 9(8), 1735–1780

[6] Lippmann R. P., Review of neural networks for speech recognition, Neural computation, 1989, 1(1), 1–38

[7] Juang B. H., Rabiner L. R., Automatic Speech Recognition – A Brief History of the Technology Development, Georgia Institute of Technology, Atlanta, Rutgers University and the University of California, Santa Barbara, 2005

[8] Anusuya M. A., Katti S. K., Speech recognition by machine, a review, arXiv preprint arXiv:1001.2267, 2010

[9] Saeed K., Nammous M. K., A speech-and-speaker identifcation system: feature extraction, description, and classification of speech signal image, IEEE transactions on industrial electronics, 2007, 54(2), 887–897

[10] Hammami N., Sellam M., Tree distribution classifier for automatic spoken arabic digit recognition, IEEE International Conference for Internet Technology and Secured Transactions, 2009, 1–4

[11] Hammami N., Bedda M., Improved tree model for arabic speech recognition, International Conference on Computer Science and Information Technology, 2010, (5), 521–526

[12] Daqrouq K., Alfaouri M., Alkhateeb A., Khalaf E., Morfeq A., Wavelet lpc with neural network for spoken arabic digits recognition system, British Journal of Applied Science & Technology, 2014, 4(8), 1238–1255

[13] Satori H., Harti M., Chenfour N., Introduction to arabic speech recognition using cmu sphinx system, arXiv preprint arXiv:0704.2083, 2007

[14] LeCun Y., Bengio Y., Hinton G., Deep learning, nature, 2015, 521, 436–444

[15] Graves A., Mohamed A. R., Hinton, G., Speech recognition with deep recurrent neural networks, IEEE International conference on acoustics, speech and signal processing, 2013, 6645–6649

[16] Dahl G. E., Yu D., Deng, L., Acero A., Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition, IEEE Transactions on audio, speech, and language processing, 2012, 20(1), 30–42

[17] Hinton G., Deng L., Yu D., Dahl G., Mohamed A. R., Jaitly N., et al., Deep neural networks for acoustic modeling in speech recognition, IEEE Signal processing magazine, 2012, 29(6), 82–97

[18] Ali A., Zhang Y., Cardinal P., Dahak N., Vogel S., Glass J., A complete kaldi recipe for building arabic speech recognition systems, IEEE spoken language technology workshop, 2014, 525–529

[19] Ali A., Bell P., Glass J., Messaoui Y., Mubarak H., Renals S., et al., The MGB-2 challenge: Arabic multi-dialect broadcast media recognition, IEEE Spoken Language Technology Workshop, 2016, 279–284

[20] Ali A., Vogel S., Renals S., Speech recognition challenge in the wild: Arabic MGB-3, IEEE Automatic Speech Recognition and Understanding Workshop, 2017, 316–322

[21] Afify M., Nguyen L., Xiang B., Abdou S., Makhoul J., Recent progress in Arabic broadcast news transcription at BBN, Ninth European Conference on Speech Communication and Technology, 2005

[22] Manohar V., Povey D., Khudanpur S., JHU Kaldi system for Arabic MGB-3 ASR challenge using diarization, audio-transcript alignment and transfer learning, Automatic Speech Recognition and Understanding Workshop, 2017, 346–352

[23] Young S. J., Young S., The HTK hidden Markov model toolkit: Design and philosophy, University of Cambridge, Department of Engineering, 1993

[24] Davis S., Mermelstein P., Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences, Transactions on acoustics, speech, and signal processing, 1980, 28(4), 357–366

[25] Wang J. C., Wang J. F., Weng Y. S., Chip design of MFCC extraction for speech recognition, Integration the VLSI journal, 2002, 32(1-2), 111–131

[26] Lalitha S., Geyasruti D., Narayanan R., Shravani M., Emotion detection using MFCC and cepstrum features, Procedia Computer Science, 2015, 70, 29–35

[27] Ai O. C., Hariharan M., Yaacob S., Chee L. S., Classification of speech dysfluencies with MFCC and LPCC features, Expert Systems with Applications, 2012, 39(2), 2157–2165
[28] Al-Anzi F. S., AbuZeina D., The Capacity of Mel Frequency Cepstral Coefficients for Speech Recognition, International Journal of Computer and Information Engineering, 2017, 11(10), 1162–1166
[29] Rabiner L. R., Schafer R. W., Theory and applications of digital speech processing, Upper Saddle River, NJ: Pearson, 2011, 64
[30] Furui S., Speaker-independent isolated word recognition based on emphasized spectral dynamics, International Conference on Acoustics, speech and Signal Processing, 1986, 1991–1994
[31] Kumar K., Kim C., Stern R. M., Delta-spectral cepstral coefficients for robust speech recognition, IEEE international conference on acoustics, speech and signal processing, 2011, 4784–4787
[32] San-Segundo R., Montero J. M., Barra-Chicote R., Fernández F., Pardo, J. M., Feature extraction from smartphone inertial signals for human activity segmentation, Signal Processing, 2016, 120, 359–372
[33] Graves A., Fernández S., Gomez F., Schmidhuber J., Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks, The 23rd International Conference on Machine Learning ACM, 2006, 369–376
[34] Vukotic V., Raymond C., Gravier G., A step beyond local observations with a dialog aware bidirectional GRU network for Spoken Language Understanding, 17th Annual Conference of the International Speech Communication Association, 2016, 3241–3244
[35] Chung J., Gulcehre C., Cho K., Bengio Y., Gated feedback recurrent neural networks, International Conference on Machine Learning, 2015, 2067–2075
[36] Yuan Gao., Dorota Glowacka., Deep gate recurrent neural network, Asian Conference on Machine Learning, 2016, 350–365
[37] Graves A., Jaitly N., Mohamed A. R., Hybrid speech recognition with deep bidirectional LSTM, IEEE workshop on automatic speech recognition and understanding, 2013, 273–278
[38] Huang Z., Xu W., Yu K., Bidirectional LSTM-CRF models for sequence tagging, arXiv preprint arXiv:1508.01991, 2015
[39] Duda R. O., Hart P. E., Stork D. G., Pattern classification, John Wiley & Sons, 2012
[40] Haykin S. S., Neural networks and learning machines, Pearson Education, Upper Saddle River, NJ, 2009
[41] Bishop C. M., Neural networks for pattern recognition. Oxford University Press, 1995
[42] Lichman M., UCI Machine Learning Repository, University of California, http://archive.ics.uci.edu/ml, 2013
[43] Chollet F., Keras: The python deep learning library, Astrophysics Source Code Library, 2018
[44] Jiang H., Confidence measures for speech recognition: A survey, Speech communication, 2005, 45(4), 455–470
[45] Bouzgou H., Automatic Analysis of High dimensional Signals: Advanced Wind Speed Forecasting Techniques, Lambert Academic Publishing, 2012
[46] Zerari N., Abdelhamid S., Bouzgou H., Raymond C., Bidirectional recurrent end-to-end neural network classifier for spoken Arab digit recognition, International Conference on Natural Language and Speech Processing, 2018, 1–6
[47] Srivastava N., Hinton G., Krizhevsky A., Sutskever I., Salakhutdinov R., Dropout: a simple way to prevent neural networks from over-fitting, Journal of Machine Learning Research, 2014, 15(1), 1929–1958
[48] Sahidullah M., Saha, G., Design, analysis and experimental evaluation of block based transformation in MFCC computation for speaker recognition, Speech Communication, 2012, 54(4), 543–565