10.12871/clicit2014228

Neural Networks Based Methods for Voice Activity Detection in a Multi-room Domestic Environment

Giacomo Ferroni, Roberto Bonfigli, Emanuele Principi, Stefano Squartini, and Francesco Piazza
Department of Information Engineering, Università Politecnica delle Marche
Via Brecce Bianche, 60131, Ancona, Italy
{g.ferroni,r.bonfigli,e.principi,s.squartini,f.piazza}@univpm.it

Abstract

English. Several Voice or Speaker Activity Detection (VAD) systems exist in literature. They are indeed a fundamental part of complex systems that deals with speech processing. In this work the authors exploit neural network based VAD to address the speaker activity detection in a multi-room domestic scenario. The goal is to detect the voice activity in each of the two target rooms in presence of other sounds and speeches occurring in other rooms and outside. A large dataset recorded in a smart-home is provided and interesting results are obtained.

Italiano. *Un rilevatore di attività vocale (Voice Activity Detector, VAD) costituisce una delle parti fondamentali di sistemi più complessi che operano con segnali vocali. Il presente lavoro applica VAD basati su reti neurali per il rilevamento del parlato in uno scenario domestico multimicrofono. Lo scopo è quello di rilevare l'attività vocale presente nelle due stanze di riferimento in presenza di altri suoni e parlatori in altre stanze o all'esterno. Le prestazioni sono state valutate su un ampio dataset ed i risultati ottenuti sono interessanti.*

1 Introduction

Voice Activity Detection (VAD) is a non-trivial task representing one of the fundamental steps of many complex systems like Automatic Speech Recognition (ASR) (Rabiner and Juang, 1993). This work concerns the development and the evaluation of advanced VADs applied in domestic environments[1] (Principi et al., 2013). A large dataset is provided by the DIRHA EU project and it is composed of several scenes recorded using 40 microphones installed in five rooms of a smart-home (Cristoforetti et al., 2014). The approaches presented hereby are based on machine learning techniques, in particular, the first approach exploits the Deep Belief Network (DBN), a neural network obtained by stacking several Restricted Bolzmann Machines (RBMs) whilst the second approach is based on a bidirectional Long Short-Term Memory (LSTM) recurrent neural network. The proposed VADs at their current development stage have been submitted and their performance have been assessed at the Speech Activity detection and Speaker LOcalization in DOMestic environments (SASLODOM) task, part of EVALITA 2014[2].

The reminder of this technical report is structured as follows. A brief overview of the task dataset and an overall description of the proposed systems is given in the next two Sections. Section 4 describes the experimental setup while Section 5 shows the obtained results and Section 6 concludes the article.

2 SASLODOM 2014 dataset

The dataset provided by the DIRHA project refers to an apartment monitored by 40 microphones installed on the walls and the ceiling of its five rooms (cf. Figure 1). The target rooms in which the speech activity has to be detected is the kitchen (top-left) and the livingroom (bottom-left). The dataset is composed of two kind of sets named *Simulated* and *Real*. The first one is composed of 80 scenes 60 seconds long and they consist of a set of utterances and other acoustic events, including a variety of background noises, produced in different rooms and positions. The Real dataset is composed of 22 total scenes having different durations. They are composed of moving speaker utterances and system audio messages played through a ceiling loudspeaker. In these scenes the background

[1] The proposed systems are currently under development.

[2] http://www.evalita.it/2014

noise is low and the speakers are located only in the kitchen and livingroom.

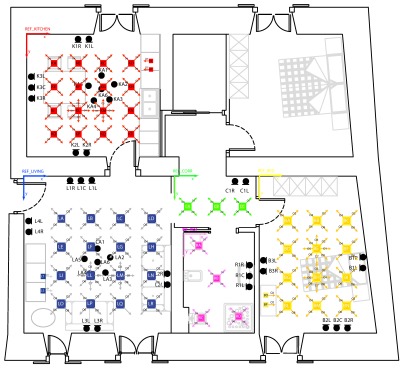

Figure 1: Layout of the experimental set-up for simulated data.

3 Overall description

The overall block scheme of the proposed approaches is depicted in Figure 2. The acquired input audio signals, coming from one or more microphones, is fed to the *feature extraction* block which aims to transform the raw audio data into a well-defined feature space (cf. Section 3.1). The feature matrix is then used as input for the *speech/non-speech* classifier. Finally a post-processing stage leads to the final decision.

3.1 Feature Extraction

Different types of features are extracted from raw audio data after down-sampling it to 16 kHz. The feature sets are normalised following the min-max method:

$$\bar{x}_l = \frac{x_l - x_{\min}}{x_{\max} - x_{\min}}, \quad (1)$$

where

$$x_{\min} = \min_{1 \le l \le L}(x_l), \quad x_{\max} = \max_{1 \le l \le L}(x_l), \quad (2)$$

x_l is an element of the feature vector at the frame index l and L is the total number of frame in the dataset. The complete list is shown in Table 1 whilst, the next sections provide a detailed description.

3.1.1 Mel-Frequency Cepstral Coefficient

The MFCC (Davis and Mermelstein, 1980) is a well-known set of features widely employed in audio applications (e.g., speech, music, etc.). Accordingly with HTK target kind (Young et al., 1997), two set of MFCC-based feature have been extracted: MFCC12_0_D_A and MFCC12_0_D_Z.

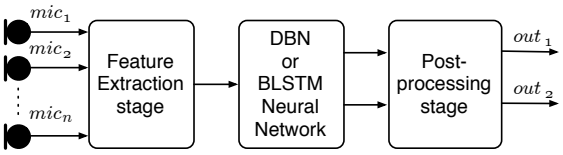

Figure 2: General block scheme of the proposed VADs.

Name	# features
MFCC12_0_D_Z *	26
MFCC12_0_D_A *	39
EVM_wH	1
PITCH *	1
WCLPE	24
RASTAPLP_0_D_A *	54

Table 1: List of features and their dimensionality. The * indicates that the features are extracted using openSMILE toolkit (Eyben et al., 2013).

The former is composed of 13 cepstral coefficients, 0-12, plus their first and second derivatives, Δ and $\Delta\Delta$ whilst the latter differs in the features mean normalisation and in the absence of the second order derivative. Both are extracted using a frame size of 25 ms at a frame rate of 100 fps.

3.1.2 Envelope-Variance measure

This feature relies on the signal intensity envelope smoothing introduced by the reverberation, thus, the dynamic range of a reverberated signal may be reduced (Houtgast and Steeneken, 1985). The extraction process have been slightly modified in order to achieve a temporal evolution. The original version (Wolf and Nadeu, 2014) defines a set of sub-band envelopes as the time sequences of non-linearly compressed filter-bank energies (FBE). Similarly to MFCC computation, the speech signal frame energies is computed and the mean value is subtracted in the log domain from each sub-band:

$$\hat{x}(k,l) = \exp[\log(x(k,l)) - \mu_x(k)], \quad (3)$$

where $x(k,l)$ is the sub-band time sequence, k is the band index, l is the frame index and $\mu_x(k)$ is the k-th band mean value estimated along the entire speech sub-band signal. The variance of a compressed version of Eq. (3) is obtained as follow:

$$V(k) = var[\hat{x}(k,l)^{1/3}]. \quad (4)$$

To obtain a time-varying version of Eq. (4), we compute the variance using a window W shifted

along each sub-band time sequence:

$$EVM(k,l) = var[\hat{x}(k,m)^{1/3}], \qquad (5)$$

where the variance is calculated considering a portion of $\hat{x}(k,m)$ identified by $-\frac{W}{2} + l \leq m \leq \frac{W}{2} + l$. Finally, a hard weighting function is applied to emphasise the voiceband frequencies and to discard the others contents. We use $p = 40$ mel sub-bands and a windows size of 400 ms leading to the EVM_wH set.

3.1.3 Pitch

The pitch feature is extracted accordingly to the Sub-Harmonic-Summation (SHS) method (Hermes, 1988). It computes N_f shifts of the input spectrum along the log-frequency axis, each of them is scaled due to a compression factor and summed up leading to a sub-harmonic summation spectrum. Standard peak picking and a quadratic curve fitting interpolation are applied to identify the F_0 value. They are extracted using a frame size of 50 ms sampled every 10 ms.

3.1.4 RASTA-PLP

This feature set is the standard RASTA-PLP set (Hermansky, 1990) composed of 18 cepstral coefficients including the 0-th one plus their first and second derivatives. They are extracted using a frame size of 25 ms sampled every 10 ms.

3.1.5 WC-LPE Feature

The Wavelet Coefficient (WC) and Linear Prediction Errore (LPE) feature set is based on a sub-band multi-resolution representation due to the exploitation of the Discrete Wavelet Transformation of the input. A set of Linear Prediction Error Filters (LPEFs) is then applied to each sub-band in order to extract the Forward Prediction Errors (FPE). The latter, the WCs and their first average derivatives constitute the feature set presented in (Marchi et al., 2014). To guarantee a frame alignment with respect to other feature sets, the reference frequency has been set to 100 Hz.

3.2 Deep Belief Network

The DBN is well-defined in (Deng, 2012) as a probabilistic generative models composed of multiple layers of stochastic, hidden variables. The top two layers have undirected, symmetric connections between them. The lower layers receive top-down, directed connections from the layer above. A DBN is built by a stack of Restricted

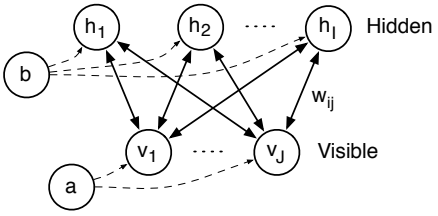

Figure 3: Restricted Boltzmann Machine.

Boltzmann Machines (RBMs) and the interest in this generative model began to increase since the introduction of an efficient layer-by-layer unsupervised training algorithm, also called pre-training (Hinton et al., 2006). DBNs are typically used to initialise the weights of a Multi-Layer Perceptron (MLP) neural network, especially when the MLP is composed of many layers (i.e., deep neural network, DNN). Following this initialisation, a standard back-propagation fine-tunes the network leading to much better results than that achieved by randomly initialise the MLP. When DBN in exploited for initialisation of a DNN, the obtained network is called DBN-DNN.

RBMs are composed of one layer of Bernoulli stochastic hidden units $\mathbf{h}$ and one layer of Bernoulli or Gaussian stochastic visible units $\mathbf{v}$, where $\mathbf{h}$ and $\mathbf{v}$ are the vector of hidden and visible unit values. With respect to Boltzmann Machines, RBMs have not hidden-to-hidden and visible-to-visible connections. Figure 3 shows a RBM with I visible units and J hidden units, w_{ij} indicates the weights between i-th visible unit v_i and j-th hidden unit h_j, and b_i and a_j are respectively the bias terms for visible and hidden layers. Following (Hinton, 2010), a RBM can be easily trained by means of Contrastive Divergence (CD-1) algorithm which allows to compute the approximation of the gradient of the log likelihood $\log p(\mathbf{v};\theta)$, where θ is the model parameters, by exploiting a full step of the Gibbs sampling method. A full step consists in sampling $\mathbf{h}_0$ from $\mathbf{v}_0$, then sampling $\mathbf{v}_1$ from $\mathbf{h}_0$ and, finally sampling $\mathbf{h}_1$ from $\mathbf{v}_1$. Hence, the weights update rule for the RBM is:

$$\Delta w_{ij} = \epsilon[\langle v_1 h_1 \rangle - \langle v_0 h_0 \rangle], \qquad (6)$$

where ϵ is the learning rate and the vector of visible units $\mathbf{v}_0$ are initialised using the input data.

In the stacking procedure, the RBMs are trained using the CD-1 algorithm layer by layer leading to a DBN as shown in Figure 4. Firstly RBM_1 is pre-

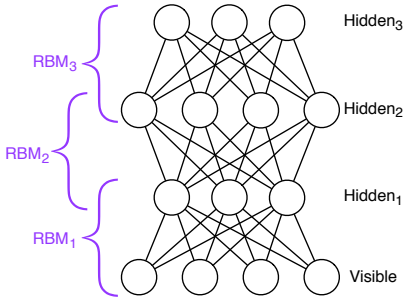

Figure 4: Deep Belief Network obtained by stacking three RBMs.

trained, then the hidden unit activation probabilities of RBM_1 became the visible units of RBM_2 and the pre-training algorithm is applied to RBM_2. Finally the hidden unit activation probabilities of RBM_2 became the visible units of RBM_3 which is pre-trained. This process proceeds iteratively for each layer in the network. It is important to note that this training procedure is unsupervised, thus, it does not require the targets or labels knowledge. For classification tasks, the pre-training is followed by a supervised training algorithm (e.g., back-propagation) which, on the contrary, exploits the targets to fine-tune the network weights.

3.3 Bidirectional LSTM-RNN

A BLSTM-RNN is a recurrent neural network in which the usual non-linear neurons (i.e., sigmoid function) are replaced by the long short-term memory blocks.

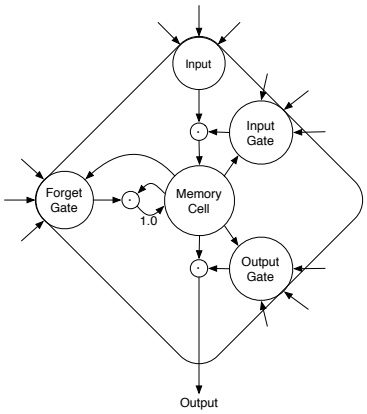

Figure 5: Long Short-Term Memory block.

The LSTM block is composed of one or more self connected linear memory cells and three multiplicative gates, as shown in Figure 5. The memory cell maintains the internal state for a long time through a constant weighted connection (i.e., 1.0). The content of the memory cell is controlled by the multiplicative input, output and forget gates which act respectively as the memory write, read and reset operations. More details can be found in (Hochreiter and Schmidhuber, 1997; Graves, 2012).

The recurrent nature of the network allows a kind of *memory* in the network internal state which is exploited to compute the output of the network. To deal with the future context, an elegant solution is to duplicate the hidden layers and connect them to the same input and output. The input values and corresponding output targets are thus given in a forward and backward direction. This network architecture is called Bidirectional LSTM-RNN (BLSTM-RNN).

4 Experimental Setup

The given dataset has been divided as provided by the SASLODOM 2014 organisers:

- **Development Set**: 40 scenes from the Simulated set and 12 scenes from the Real set.
- **Test Set**: 40 scenes form the Simulated set and 10 scenes from the Real set.

The Test Set has been provided to the participants at the end of the development phase in order to evaluate the performance, hence the feature selection, the network parameters identification and the post-processing variables tuning have been computed by means of a 10-fold cross validation over the Development Set.

4.1 DBN-VAD

The proposed DBN-VAD (cf. Figure 2) has two different configurations. In particular, the feature set and the network topology are different due to the diverse nature of the Simulated and Real sets. The feature set employed with the simulated dataset is composed of 106 coefficients/frame for each microphone: MFCC12_0_D_Z, EVM_wH, PITCH, WC-LPE and RASTAPLP_0_D_A. The network has 212 input units, two hidden layers of, respectively, 20 and 10 units and an output layer of two units, one for each target rooms. We refer to this configuration as DBN-VAD_S. On the other hand, both the feature set and the network size for the real dataset are smaller: 27 coefficients/frame MFCC12_0_D_Z and PITCH, and 57 inputs units, two hidden layers of 10 and 5 units and two output units. We refer to this configuration as DBN-VAD_R.

Both the configurations exploits two microphones installed on the kitchen wall (i.e., K2L) and on the livingroom wall (i.e., L1C). The choice of these two microphones relies on their position (cf. Figure 1) and also as a result of intensive tests conducted on several microphone pairs.

The DBN-VAD$_{S|R}$ pre-training consists in 1000 iterations using a mini-batch size of 100 frames and a step-ratio of 0.1. The learning rate is obtained dividing the step-ratio by the size of the training set leading to a value close to 4×10^{-7}. The fine-tuning training has the same parameters.

4.2 BLSTM-VAD

The second proposed VAD is BLSTM-based (cf. Figure 2) and exploits the two microphones used with the DBN-VAD (i.e., K2L and L1C). This VAD employs a different feature set composed of MFCC12_0_D_A, PITCH and WC-LPE leading to a total feature space of 64 coefficients per frame per microphone. The final network topology is composed of four hidden layers (i.e., two for each direction due to bi-directionality) with 40 and 20 LSTM units for each direction. The input layer has 128 units while the output layer has only one unit. Indeed, for this VAD approach, better performance has been achieved using one network for each room.

For BLSTM-VAD training, the CURRENNT toolkit (Weninger et al., 2014) is used. In particular, supervised learning with early stopping is used. Standard gradient descend with back propagation of the output errors is used to iteratively update the network weights. The latter are initialized by a random Gaussian distribution with mean 0 and standard deviation 0.1.

4.3 Post-processing

A post-processing of the network output is needed in order to handle slow transition from speech to non-speech. This technique is commonly named *hangover* and a number of different implementation have been developed. The simplest implementation, used in this work, exploits a counter. In particular, a threshold value is fixed and if at least two consecutive network outputs are above the threshold, the counter is reset to a predefined value (equal to 8). On the contrary, when the network output is below the threshold, the counter is decreased by 1 and the actual frame is classified as non-speech only if the counter value is zero.

5 Results

The result published by SASLODOM 2014 organisers are shown in this section.

5.1 Performance metrics

The metrics used to assess the VAD performance are:

- Deletion Error Rate (DER): number of missing detection over all speech frames.
- False Alarm Rate (FAR): number of false detection over all non-speech frames.
- Overall Speaker Activity Detection error (SAD): global metric defined as:

$$\text{SAD} = \frac{N_{fa} + \beta N_{del}}{N_{nsp} + \beta N_{sp}}, \tag{7}$$

  where N_{del}, N_{fa} are the total number of deletions and false alarms respectively, N_{sp} and N_{nsp} are the total number of speech and non-speech frames. The term $\beta = \frac{N_{nsp}}{N_{sp}}$ acts as regulator term for the unbalance of the class non-speech with respect to the speech one.

Table 2 shows the performance achieved by the proposed VADs with respect to the Test Set. The proposed VADs at their current development stage are characterised by moderate performance with respect to the Real dataset. This fact is due to the *raw* approach that authors decided to undertake as first step. In particular, the data-driven nature of our VADs does not exploit higher level information to finalise the decision. For instance it could be possible to exploit the envelope-variance measure (cf. Eq. (4)) to perform a channel selection and hence further post-processing the network decisions. This solution would reasonably improve the performance on Real dataset. Indeed, the absence of noise in its scenes leads to a high accuracy of the channel selection measure. Performance against the Simulated data are significantly better due to the grater dimension with respect to the Real data.

6 Conclusion

The proposed VADs exploit DBN-DNN and BLSTM-RNN neural networks in order to detect the speaker activity in a multi-room scenario. Indeed, the task goal is the detection of when and where a human is talking with respect to target rooms. Hence, the system is required to be robust

VAD	Simulated data			Real data		
	DER (%)	FAR (%)	SAD (%)	DER (%)	FAR (%)	SAD (%)
DBN-VAD$_{S\|R}$	10.3	8.7	9.5	14.7	9.7	12.2
BLSTM-VAD	12.3	11.9	12.1	5.6	33.7	19.7

Table 2: Result assessed against the Test Set.

and reliable in a noise environment and a multiple speaker scenario. Furthermore, the VAD is also required to identify in which room, kitchen or livingroom, the speaker is actually talking discarding other speaker(s) in other room(s). The performance of the proposed approaches have been assessed on the SASLODOM-EVALITA 2014 task. Further intensive test sessions focused to preprocess the multiple microphone signals available and to the evaluation of deeper networks represent future efforts. Moreover, due to the so-called *curse of dimensionality*, better performance are expected by the exploitation of the whole DIRHA dataset.

Acknowledgment

The project has been developed by the audio team of Multimedia Assistive Technology Laboratory (MATeLab) at the Università Politecnica delle Marche, which operates in the ambient assisted living context exploiting audio-visual domain features. This research is part of the HDOMO 2.0 project founded by the National Research Centre on Aging (INRCA) in partnership with the Government of the Marche region under the action "Smart Home for Active and Healthy Aging".

References

L. Cristoforetti, M. Ravanelli, M. Omologo, A. Sosi, A. Abad, M. Hagmüller, and P. Maragos. 2014. The dirha simulated corpus. In *Proc. of LREC*, volume 5.

S. Davis and P. Mermelstein. 1980. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *Acoustics, Speech and Signal Proc., IEEE Transactions on*, 28(4):357–366.

L. Deng. 2012. Three classes of deep learning architectures and their applications: A tutorial survey. *APSIPA Transactions on Signal and Information Processing*.

F. Eyben, F. Weninger, F. Gross, and B. Schuller. 2013. Recent developments in opensmile, the munich open-source multimedia feature extractor. In *Proc. of the 21st ACM international conference on Multimedia*, pages 835–838. ACM.

A. Graves. 2012. *Supervised sequence labelling with recurrent neural networks*, volume 385. Springer.

H. Hermansky. 1990. Perceptual linear predictive (plp) analysis of speech. *the Journal of the Acoustical Society of America*, 87(4):1738–1752.

D. J. Hermes. 1988. Measurement of pitch by subharmonic summation. *The journal of the acoustical society of America*, 83(1):257–264.

G. Hinton, S. Osindero, and Y. Teh. 2006. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554.

G. Hinton. 2010. A practical guide to training restricted boltzmann machines. *Momentum*, 9(1):926.

S. Hochreiter and J. Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

T. Houtgast and H. J. M. Steeneken. 1985. A review of the mtf concept in room acoustics and its use for estimating speech intelligibility in auditoria. *The Journal of the Acoustical Society of America*, 77(3):1069–1077.

E. Marchi, G. Ferroni, F. Eyben, L. Gabrielli, S. Squartini, and B. Schuller. 2014. Multi-resolution linear prediction based features for audio onset detection with bidirectional lstm neural networks. *Proc. of 39th IEEE ICASSP*.

E. Principi, S. Squartini, F. Piazza, D. Fuselli, and M. Bonifazi. 2013. A distributed system for recognizing home automation commands and distress calls in the italian language. In *Interspeech*, pages 2049–2053.

L. R. Rabiner and B. Juang. 1993. *Fundamentals of speech recognition*, volume 14. PTR Prentice Hall Englewood Cliffs.

F. Weninger, J. Bergmann, and B. Schuller. 2014. Introducing CURRENNT – the Munich Open-Source CUDA RecurREnt Neural Network Toolkit. *Journal of Machine Learning Research*, 15.

M. Wolf and C. Nadeu. 2014. Channel selection measures for multi-microphone speech recognition. *Speech Communication*, 57:170–180.

S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, et al. 1997. *The HTK book*, volume 2. Entropic Cambridge Research Laboratory Cambridge.