The OCON model: an old but green solution for distributable supervised classification for acoustic monitoring in smart cities

Stefano Giacomelli [1], Marco Giordano [1], and Claudia Rinaldi [2]

[1]DISIM - University of L'Aquila, L'Aquila, Italy
[2]CNIT - Consorzio Nazionale Interuniversitario per le Telecomunicazioni, L'Aquila, Italy

***Abstract*—This paper explores a structured application of the *One-Class* approach and the *One-Class-One-Network* model for supervised classification tasks, focusing on *vowel phonemes classification* and *speakers recognition* for the Automatic Speech Recognition (ASR) domain. For our case-study, the ASR model runs on a proprietary sensing and lightning system, exploited to monitor acoustic and air pollution on urban streets. We formalize combinations of *pseudo*-Neural Architecture Search and Hyper-Parameters Tuning experiments, using an *informed* grid-search methodology, to achieve classification accuracy comparable to nowadays most complex architectures, delving into the speaker recognition and energy efficiency aspects. Despite its simplicity, our model proposal has a very good chance to generalize the language and speaker genders context for widespread applicability in computational constrained contexts, proved by relevant statistical and performance metrics. Our experiments code is openly accessible on our GitHub.**

***Index Terms*—Artificial Intelligence (AI), Deep Learning (DL), Neural Networks (NNs), Green-AI, Digital Signal Processing (DSP), speech communication, phonetics, phonology, vowel phonemes.**

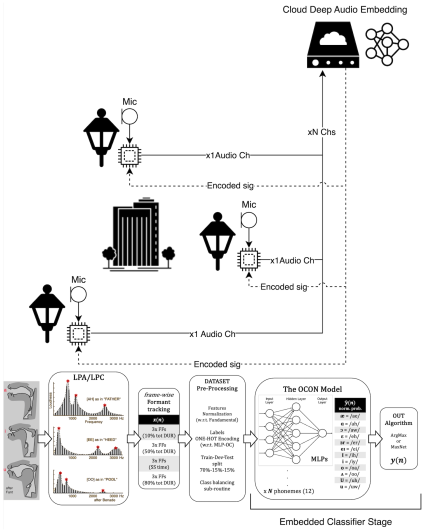

Fig. 1. Our smart city case-study scenario

I. Introduction

Acoustic sensing for the safety, security, and monitoring of urban and non-urban environments is becoming increasingly important. This trend is driven by the widespread adoption of the smart cities vision by Western municipalities [1], [2] and the need to protect ecosystems in wild areas [3]–[5]. Solutions proposed in the literature over the years addressed two critical topics: communication networks and Machine Learning (ML) [6], [7]. Collected data need to be transmitted and processed, with the sequence of these operations depending on many factors (not addressed in this work). Recent technological advancements in both fields are worth exploring and new communication networks like 5G and 6G enable previously impossible applications by introducing new concepts such as network slicing, network functions virtualization, orchestration, multi-access edge computing, Open Radio Access Networks (O-RAN), and software-defined networking [8], [9]. Additionally, ML and Neural Networks (NNs) are extensively spreading across various fields, being highly desirable for acoustic monitoring due to their proved accuracy levels [10]–[13].

In a smart city scenario (Fig.1), key areas of interest are Sound Event Detection (SED) and Audio Tagging (AT), particularly for identifying the cause of exceeding a safe acoustic threshold. This operation can be performed either at the sensor location or remotely, especially when detailed information about the type of sound needs to be transmitted. This scenario fits perfectly within the field of the Internet of Sounds (IoS) [14], which involves the interconnected network of devices capable of capturing, processing, and transmitting audio data with constrained computational resources [15]. When adopting ML solutions for this purpose, care must be taken due to the limited computational capabilities of local sensing hardwares [16], [17]. One possible solution is to send all data to a remote cloud, another is to reduce the computational complexity of the NN to be used locally and improve its features abstraction capability, thus avoiding privacy issues too.

This paper evaluates a streamlined combination of Neural Architecture Search (NAS) and Hyper-Parameters Tuning (HPs-T) for designing abstraction/classification NNs models. We propose a modular "One-Class One-Network" (OCON)

This work was partially supported by the EU under the Italian NRRP of NextGenerationEU, partnership on "Telecommunications of the Future" (PE00000001 - program "RESTART")

model, which consists of parallel binary classifiers (instead of a single multi-class layer) dedicated to simpler and specific SED tasks: phonetic and speakers recognitions [18]. By assessing data constraints and task complexities with respect to the current State-of-the-Art (SoA), we strive to develop a *shallow* and optimizable sub-architecture, characterized by a *sustainable* and straightforward re-training cycle (given the constrained computational and emission requirements). Moreover, we determine the minimum number of formant features required to achieve SoA accuracy levels in phonetic recognition.

Building on recent results [19], we also explore the opportunity of recognizing the gender of the speaker for enhanced contextual understanding, personalized response, and behavioral analysis. Recent progress in gender recognition has reached a SoA level employing advanced ML solutions, in conjunction with sophisticated signal pre-processing techniques, to achieve remarkable accuracy in identifying and interpreting vocal subtleties. These approaches encompass feature extraction in both temporal and (pseudo)-frequency domains [20], as well as NNs and Deep Learning (DL) model ensembles, which are now at the core of almost every Automatic Speech Recognition (ASR) system [21]–[25].

II. Methodologies

We started by collecting a reliable audio *dataset*, including multiple phonetic hues and gender diversity among speakers. We limited our linguistic research to the *General American English* case-study, as defined by the International Phonetic Association (IPA). We decided to focus only on well-established pre-processed datasets (Sec. II-A), designed by means of pre-arranged phraseological segments or specific words, like the */hVd/ containers* (where vowels are placed between an "*h*" and a "*d*"). These segments were recorded, analyzed (formant analysis), and pre-processed to extract meaningful features (*formant* frequencies) suitable for our NNs model, obtained following this steps:

(*1*) segment speech signals into *semantic frames*, either manually or automatically, following a pre-defined semantic grid (words/phonemes, and silences);

(*2*) use Linear Predictive Coding/Analysis (LPC/LPA) to analyze isolated segments, obtaining a smoothed time-frequency aggregated spectral estimate (per audio frame);

(*3*) extract the top N spectral peaks using any *peak estimation* algorithm, ensuring to track frame-by-frame continuities (contouring).

Additional post-processing is added to refine retrieved formant frequency tracks and create a suitable (*features*) vector for the input layer of our NNs model. These pre-processing proves to be crucial in allowing the networks to learn related abstract representations effectively, thereby optimizing recognition accuracy.

TABLE I
HGCW dataset filenames structure

1^{st} **character**	2^{nd}&3^{rd} **ch.s**	4^{th}&5^{th} **ch.s**	**Example**
m (*man*)	spk. n° (50 tot.)	ARPABet ch.s	`m10ae`
b (*boy*)	/ (29 tot.)	/	`b11ei`
w (*woman*)	/ (50 tot.)	/	`w49ih`
g (*girl*)	/ (21 tot.)	/	`g20oo`

TABLE II
HGCW actual classes statistics

Phoneme	**Samples**	**Boys**	**Girls**	**Men**	**Women**	**Label ID**
ae (**ɒ**)	134	25	17	45	47	0
ah (**a**)	135	24	19	45	47	1
aw (**ɔ**)	133	24	18	45	46	2
eh (**ɛ**)	139	27	19	45	48	3
er (**ɤ**)	118	26	18	37	37	4
ei (**e**)	126	25	17	43	41	5
ih (**y**)	139	27	19	45	48	6
iy (**i**)	124	20	18	43	43	7
oa (**o**)	136	25	19	45	47	8
oo (ø)	139	27	19	45	48	9
uh (**u**)	138	26	19	45	48	10
uw (**ɯ**)	136	25	19	44	48	11
TOTAL	1597	301	221	527	548	12

A. Datasets Review

As already discussed in [19], the choice of the dataset has been done basing on various reasoning such as phonetic complexity and gender balance, by analyzing in detail three of those freely available as the *Peterson and Barney (PB)* [26], the *HGCW* database [27] and the Texas Instruments & Massachusetts Institute of Technology (TI-MIT) *Corpus of Read Speech* [28].

Recognizing the limitations of existing datasets for developing *fluid* and robust generalizable solutions, we opted for the HGCW dataset, which offers the highest level of phonetic complexity (although being merely *binary-labeled*, a known limitation for the speaker gender task). By leveraging pre-extracted formant data from the HGCW repository, we aim to expedite the data retrieval process, promote consistency with the literature, and streamline results evaluation.

B. Features Pre-Processing & Classification

The filename structure of the HGCW dataset (Table I) encodes essential phonetic and speaker features, which are crucial for a preliminary statistical analysis.

Pre-processing solutions applied have already been discussed in [19]. We remark that the presence of *null* features (in some samples) caused by authors algorithm failures, required further samples filtering, leading to additional under-representation of certain phoneme and speaker classes (Table II), to maintain balance and thus learning consistency. Fundamental frequency tracks ($F0$) were retrieved by means of a 2-way auto-correlation/zero-crossing pitch tracker, followed by a halving/doubling result evaluation sub-routine [29], while formant frequencies were estimated using LPA and peak retrieval with parabolic interpolation [30]. The resulting frequency trajectories were additionally refined with an interactive audio spectral editor, which was used for manual examination and interpolation of discontinuities.

TABLE III
PHONEMES & SPEAKER RECOGNITION W. PB DATASET
(GM = *geometric mean*)

Task	Data scale	Processing	ML	Accuracy
Phoneme	Hz	*Jacknife*	LDA	81.8%
	Log	None	GLM	87.4%
	/	-GM($F0$), $\cdot 0.333$	LDA	86.3%
	/	-($\bar{F}1$, $\bar{F}2$, $\bar{F}3$)	LDA	89.5%
	Bark	None	GLM	86.2%
	/	*Jacknife*	LDA	85.7%
	/	-GM($F0$)	LDA	85.3%
	/	-($\bar{F}1$, $\bar{F}2$, $\bar{F}3$)	LDA	88.3%
	ERBs	None	GLM	86.8%
	/	-GM($F0$), $\cdot 0.5$	LDA	87%
	/	-($\bar{F}1$, $\bar{F}2$, $\bar{F}3$)	LDA	88.8%
Speaker	Hz	None	LDA	89.6%
	Bark	None	LDA	88%
	/	ΔF_n	LDA	41.7%

We recall that different experimental sub-structures of the original dataset have been obtained, categorizing each sample relying on:

(*1*) *Phonemes grouping*, including $F0$ and the first 3 formant frequencies at the *steady state* (SS);

(*2*) *Speakers grouping* with the same features as above, relying on provided gender labels;

(*3*) *Phonemes grouping*, including $F0$ and a total of 12 formant frequency values, with the first three formants sampled at 10%, 50%, SS, and 80% of the total duration of the vowel nucleus.

To establish a consistent reference baseline, we analyzed classification algorithms evaluated on PB and/or HGCW dataset features (Table III) only. Linear Discriminant Analysis (LDA) [31] and Generalized Linear Regression Models (GLM) [32] resulted the most prominent and effective approaches. These methods were combined with innovative formant feature processing, such as the 3*D-auditory target zones* framework, using logarithmic formant distances [33]. Other studies applied canonical auditory frequency transforms including the Bark scale [34], [35], Mel scale approximations [36], and *lin-to-log* frequency mapping [37].

Research on phonetic NNs recognition has mainly focused on using LPA coefficients directly [38] or spectral/cepstral-derived features [39], often incorporating complex convolutional and/or recurrent modules. The only study on OCON phonetic classification [40] reported improvements exclusively over TI-MIT data, using LPC features. Due to a limited comparative literature, we set a target average accuracy of 90%, aiming to improve results reported in [27], [29] and [19].

Considering significant variations in $F0$ within speakers (due to physiological factors) and related pitch deviations caused by *prosody*, we introduce a *Linear Formant Normalization*, with respect to $F0$:

$$ratio(F_{i,n}) = \frac{F_{i,n}}{F0_n} \quad (1)$$

where F_i is the non-normalized i^{th} formant (in Hz) and $F0_n$ is the fundamental frequency of the n^{th} phoneme.

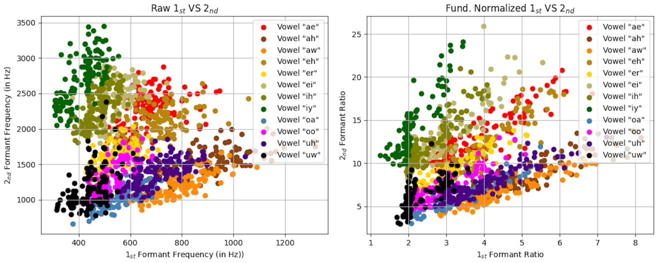

Fig. 2. HGCW dataset normalization (2D formantic projection)

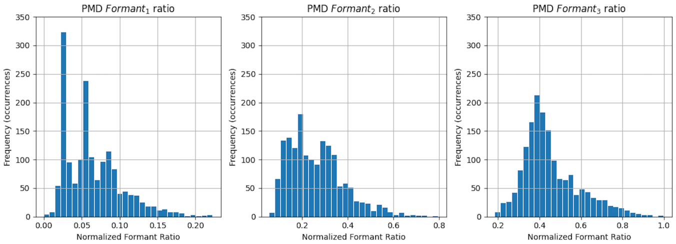

Fig. 3. HGCW Dataset PMDs (across all classes)

Class distributing boundaries are improved, as shown in the 2D formantic projection in Fig.2. To enhance NNs training convergence, we applied *min-max* scaling to normalize the entire feature set and we also examined Probability Mass Distributions (PMDs) of resulting formant ratios to assess the feasibility of *Z-score* (standardization). However, the PMDs consistently exhibited a skewed distribution, resembling either Poissonian or Log-normal spread (Fig. 3).

To preserve data resolution, we encoded all pre-processed features in a binary `NumPy` open-source compressed format (`.npz`), specifically designed to enhance data portability and re-usability.

III. PRACTICAL IMPLEMENTATION

In order to achieve both phonetic and speakers gender classification, we propose the exploitation of a specific OCON proposal, which models multi-output classification tasks using multiple independent exact-copies of the same optimized Multi-Layer Perceptron (MLP) reference architecture.

These configurations are derived through simplified and *informed* NAS experiments (*pseudo*-NAS) combined with HPs-T: in DL research, HPs-tuning involves optimizing architectural and learning parameters (such as layers, nodes, *backpropagation* optimizers, learning rate etc.) to minimize the network cost function, between the predicted result (class) and the provided *ground-truth* (label), in supervised learning contexts.

A. Architecture & Model

MLPs, also referred to as *Feed-forward* NNs or *fully connected* (FC) layers, are essentially stacks of *Perceptrons*

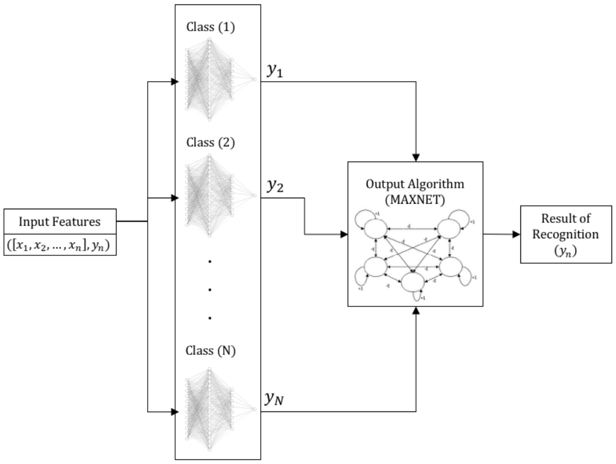
Class (1)
y_1
Class (2)
y_2
Class (N)
y_N
Input Features
$([x_1, x_2, \dots, x_n], y_n)$
Output Algorithm (MAXNET)
Result of Recognition (y_n)

Fig. 4. The OCON model

(neurons) arranged in vertical layers (*shallow* NNs), whose function is:

$$y_n = \varphi\langle x, w_k \rangle = \varphi(x^\top w_k) = \varphi\left(\sum_{k=0}^{K} x_n w_k\right) \tag{2}$$

where x_n are the input features, w_k a set of scaling coefficients (*weights*) and $\varphi(\cdot)$ a non-linear ReLU function (*activation*) [41]:

$$\varphi(x) = \max(0, x) = \begin{cases} x & \text{if } x > 0, \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Introduced in the '90s, the One-Class-One-Network (OCON) model [42] served as a solution for NNs *parallel distributed processing*, aiming to overcome limitations of architectures that required full re-training when altering their dataset classes. Today the OCON resembles a simplified form of architecture *ensembling* 4, where multiple complex networks are combined through other blocks or algorithms, to enhance the overall model accuracy. In the Anomaly Detection and Computer Vision fields [43]–[45], the *One-Class* approach consists of distributing a multi-output classification across a bank of independent sub-networks, each functioning as a *context-specific* binary classifier. In our study, we divided a 12-phoneme and 3-genders classification tasks respectively into a bank of 12 and 3 independent and distributable classifiers, with identical architectural topology, aiming for an optimal *average architecture estimation.*

If a discrete output label is needed, a context-specific output algorithm must be devised. While no literature references were found regarding OCON-specific output algorithms, figures in [42] suggest the involvement of a *MaxNet* sub-network [46].

Algorithm 1 MaxNet Algorithm

```
Require: $f(\cdot)$                                   ▷ Activation function
Require: $n$                                          ▷ Nodes number
Require: $\varepsilon \cong \frac{1}{n}$              ▷ Inhibition magnitude
Require: $\{y_1, \dots, y_n\}$                        ▷ Network outputs
Require: criterion                                    ▷ Winner-takes-all evaluation
  for $k = (1, \dots, n)$ do                          ▷ Weights initialization loop
      if $k = n$ then
          $\theta_k = +1$                             ▷ Self-weight assignment
      else
          $\theta_k = \varepsilon$                    ▷ Inhibition-weight assignment
      end if
  end for
  while criterion do
      for $k = (1, ..., n)$ do
          if $i \neq j$ then
              $y'_k = f(y_k - \theta_k \sum_{i=1}^{n} y_i)$   ▷ Competition
          else
              $y'_k = y_k + \theta_k$                 ▷ Unitary increment
          end if
          $y_k \leftarrow y'_k$                       ▷ New outputs assignment
      end for
  end while
```

The MaxNet can encounter critical flaws when multiple maxima occur in the input state, potentially leading to infinite *competitive looping*. To mitigate this issue, the *argument of the maxima* (*ArgMax*) is employed, which typically returns a single value, representing the first occurrence of a maximum, when multiple exist. However, we find the classification *logits* vector more beneficial for investigating phonetic and speakers class boundaries, features complexities and/or similarities.

During supervised training, each sample label undergoes binarization (*one-hot encoding*) specifically tailored to the One-class architecture, while features are concurrently fed into all classifiers. To automatize this process, we devised a custom one-hot encoding sub-routine (Alg.2), so as to transform labels according to the incoming `True`-One-class. Additionally, to address classes under-representations (observed in Table II), we perform a slight down-sampling of resulting training subsets.

Algorithm 2 HGCW One-Hot encoding

```
Require: $c$                                          ▷ True-class index
Require: $s$                                          ▷ Phoneme groups size
Require: $\mathcal{X}$                                ▷ Features dataset
Require: $\mathcal{Y}$                                ▷ Dataset labels
  $\text{class}_1 = \mathcal{X}(c)$                   ▷ Initialize True-class subset
  $\text{size} = \text{length}(\text{class}_1)$       ▷ Extract True-class size
  $\text{classes}_0 = \text{list[ ]}$                 ▷ Initialize False-class subsets
  $\text{sub-sizes} = round(\frac{size}{11})$         ▷ Compute False-classes size
  for $k$ in $\mathcal{Y}$ do                         ▷ Subsets selection loop
      if $\mathcal{Y}_k \neq c$ then
          $\text{class}_0 = rand(\mathcal{X}_k, \text{sub-sizes})$   ▷ Random downsampling
          $\text{classes}_0\text{.append}(\text{class}_0)$
      else
          pass
      end if
  end for
```

Alg.2 executes the one-hot encoding routine once per architecture training cycle, preceding the *train-eval-test* splitting and the *mini-batch* partitioning of features subsets. It ensures a balanced outliers quantity by allocating the same number of samples for all `False`-classes among the remaining 11 (or 2 for speakers), based on the available size of the `True`-class. If `True`-class sizes are not divisible by 11, a variability of 1 to 3 samples is deemed acceptable. Speaker-based encoding

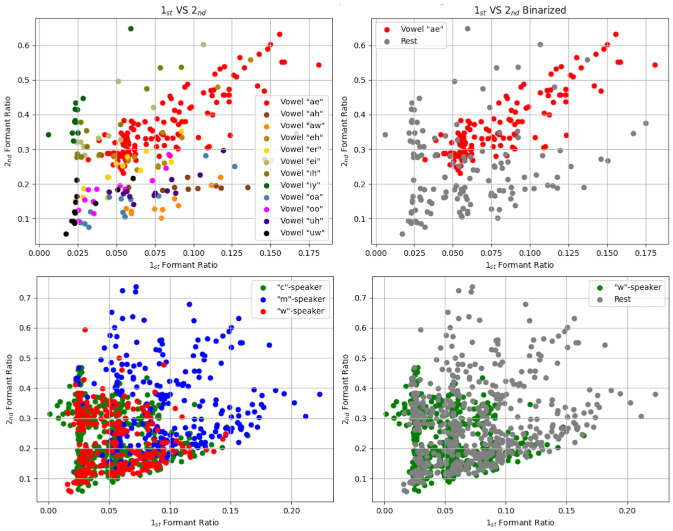

Fig. 5. HGCW dataset *One-Hot encoding* examples

for *male*, *female*, and *children* classes is achievable (Fig. 5) in the same way (with less noticeable variability).

B. Pseudo-NAS & HPs-T search

The term *pseudo*-NAS, as discussed in Sec.III-A and in [19], refers to the *a prìori* constraint applied to the architecture topology (MLP). Our model evaluation will determine the optimal number of layers, and nodes per layer required to effectively address both phoneme and speakers gender recognition.

Conversely, *grid-based* HPs search is a statistical method where all possible combinations of NNs HPs are independently sampled and evaluated through straightforward learning cycles. While theoretically effective, it can be a time-consuming solution due to the exponential increase in computational requirements (for narrowing resolutions): typically, all possible combinations must be tested before selecting the optimal one. We achieved a good trade-off by establishing independent resolutions for each HP beforehand, employing an *informed* iterative approximation, summarized as follows:

(*1*) define a specific subset of HPs (not necessarily all at once, potentially fixing others);

(*2*) sample each HP with an arbitrary resolution;

(*3*) test each combination of HPs and evaluate resulting *temporary best estimates*. These can either serve as *inheritable optimal estimates* for subsequent heuristic stages or guide parameter resolution sampling towards *local good estimates*, in search of better sets;

(*4*) repeat steps (*2*) to (*3*) as much as needed, to refine and improve the model configuration.

Acknowledging that this simplified approach roughly approximates theoretical grid-search, leading to potential misleading local minima in model costs, our goal remains to identify an average One-class topology in a computationally feasible manner.

TABLE IV
NAS & LEARNING HEURISTIC STAGE
IN = *input nodes*, LR = *learning rate*, HN/L = *hidden nodes/layers*

Input Features	Fixed HPs	Testing HPs
SS formant ratios	IN (3) HL (1) activations (ReLU) states init. (*standard* [41], $b = 0$) epochs (1000) batch size (32) k-folds (3)	HN $(10, 50, 100)$ Backprop (*Adam*, *RMSProp*) LR $(10^{-3}, 10^{-4}, 10^{-5})$
TOT sets: 18	**TOT architectures: 12**	**TOT cycles: 648**

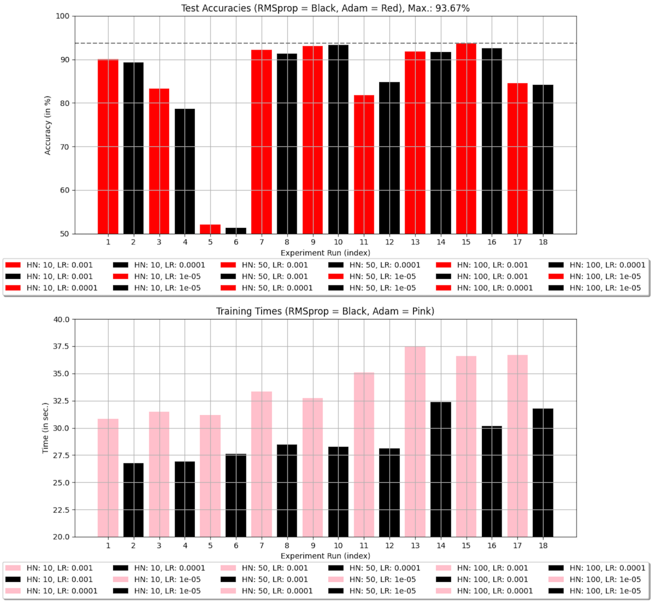

Fig. 6. $1st$ heuristic stage results

Heuristic learning experiments involved partitioning the dataset into train (70%), dev (15%), and test (15%) sets, with seeded initial states (for random initialization processes involved). Accuracy and mini-batch training times are measured, and results are averaged over a 3-folded validation procedure for each One-class.

In the first *architectural* heuristic stage (Table IV), two combinations (10th and 15th) yielded similar average accuracies (93.67%, Fig.6). The RMSProp optimizer [47] demonstrated better mitigation of the increasing trend in learning times, compared to Adam [48]. However, we opted for the top-performing setup: HL: 1, HN=100, LR=10^{-4}, Backprop: Adam.

Following heuristic stages were designed to assess the incremental introduction of *regularization* techniques, and to evaluate potential advantages.

In DropOut tests [49] (Table V, Fig.7), we observed that the fastest run (1st) also reached the highest accuracy. We achieved a +0.19% accuracy, at the expense of +3.4sec. in the average training time, with DropOut probabilities set to 80% for input nodes and 50% for hidden nodes.

TABLE V
$2^{nd}, 3^{rd}$ & 4^{th} HEURISTIC STAGE (HP-T REGULARIZATION)
IN = *input nodes*, LR = *learning rate*, HN/L = *hidden nodes/layers*

DropOut HPs	**Batch-norm HPs**	**L2-Norm HPs**
IN DropOut rate (0.8, 0.9) HN DropOut rate ([0.5, 1.], res.: 0.1)	LR (10^{-3}, 10^{-4}, 10^{-5}) Batch-Norm	L2-Norm $\lambda(10^{-2}, 10^{-3}, 10^{-4})$
LR (10^{-4}) k-folds (6) epochs (3000)	k-folds (10) batch size (32) epochs (1000)	LR (10^{-4}) k-folds (10) batch size (32) epochs (1000)
TOT cycles: 864	**TOT cycles: 360**	**TOT cycles: 360**

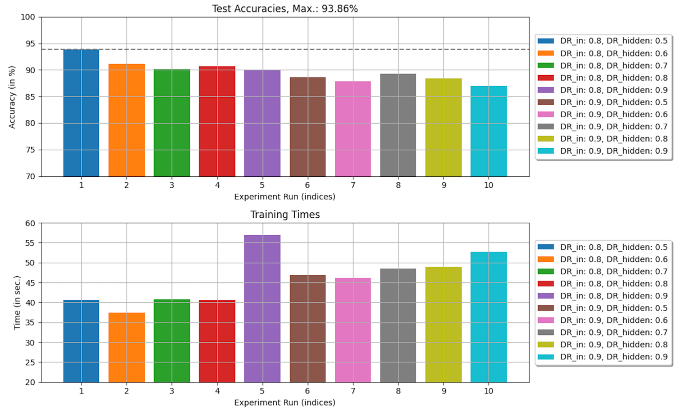

Fig. 7. 2^{nd} heuristic stage results (Dropout)

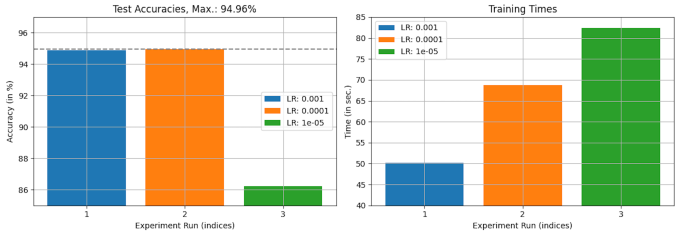

Fig. 8. 3^{rd} heuristic stage results (Batch-Norm)

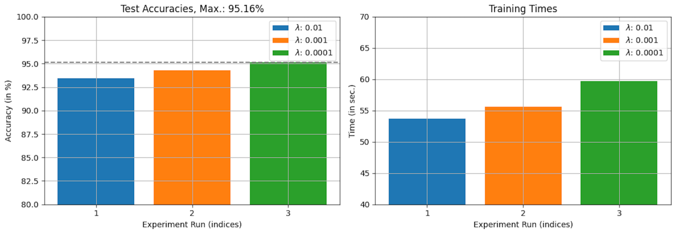

Fig. 9. 4^{th} heuristic stage results (L2-Norm)

TABLE VI
ONE-CLASS ARCHITECTURE (MLP)
IL = *input layer*, LR = *learning rate*, HN/L = *hidden nodes/layers*,
ON = *output nodes*

Architecture	**Features**	**Learning**
IL: 3 nodes HL: 1, 100 nodes ON: 1 (*logit*) ReLU (common)	ω init.: *Kaiming-He* norm. b init.: 0 One-hot encoding (Dataset *re-shuffling*)	*Adam* optim. LR: 10^{-4} Mini-bacth (32 samples)
IN-DR: 0.8 HL-DR: 0.5	Batch-Norm	L2-λ: 10^{-4}

In Batch-norm tests [50], after re-evaluating LRs, it was confirmed that LR$= 10^{-4}$ yielded the best results: a significant $+1.1\%$ in test accuracy, despite nearly doubling average training times.

For L2-Norm tests (Table V) (also Ridge penalty [51]), we found an optimal λ (weight decay) of 10^{-4} (Fig.9), resulting in a $+0.19\%$ for the average accuracy and a decrease in the average training time, now below 60 seconds. Table VI illustrates our overall averaged One-Class proposal.

IV. MODEL TRAINING & RESULTS DISCUSSION

A parallelized set of independently trainable One-Class architectures was implemented and trained using CPU runtimes (on Google Colab) to efficiently measure isolated training cycle performances and resource consumptions. We stress that the OCON architecture learning relies on the backpropagation loop of each MLP. During inference, it involves extracting sample features, computing 12 parallel one-hot encodings, and performing an ArgMax search to determine the maximum value (predicted label) within the 12-logit probabilities vector. Following this, we conducted phoneme recognition experiments to evaluate the efficiency of each dataset substructure (Sec.II-B). An Early-Stopping training strategy [52] was adopted, incorporating a two-variable escape condition: a minimum loss threshold (averaging among the last 50 training samples' loss) and a minimum test accuracy threshold based on the last batch results. These variables were further empirically assessed to ensure practical convergence of training cycles, with each cycle not exceeding a maximum amount of 25-30 minutes. While the learning phases may not be fully optimized, they were deemed satisfactory for the purposes of our study.

A. Phonemes recognition

In [19] we evaluated the OCON model using the steady-state (SS) dataset variant (Table VII): training revealed that several loss functions and training accuracy curves visibly plateaued, punctuated by periodic *spikes* indicating instances of consistent-learning batch re-shuffling (Fig.10). Interestingly, the *er* and *iy* phoneme classes exhibited substantial representation, showing almost no changes (in curve trends) post-encoding or re-shuffling (Table VII).

Overall accuracies were computed using a binary threshold of 0.5 across the entire dataset classes. While certain MLPs effectively segregated probabilities, notable errors persisted between phonemically (*aural*) similar classes, such as *ae* and *eh*, and *er* and *ei*.

Hidden dataset biases, such as similarities in formantic disposition between *children* and *women* utterances, were re-examined and filtering out these biases led to slight improvements in class boundaries separation, despite increasing

TABLE VII
1^{st} EXPERIMENT: *SS*-PHONETIC CLASSIFICATION

Features	**Training**	**Early-Stopping**
SS formant ratios	epochs: 1000 (for each *batch-set*) Re-shuffling balancing tol.: 0.01	Loss thresh.: 0.2 Accuracy thresh.: 90%
Phonemes	**Test Accuracy** (%)	**Training times** (sec.)
ae	86.27	247.62
ah	90.85	85.67
aw	86.09	117.71
eh	89.05	345.20
er	91.90	25.71
ei	84.97	539.79
ih	87.38	207.92
iy	92.21	33.78
oa	82.31	120.20
oo	85.96	396.59
uh	85.65	485.34
uw	90.91	219.23
OCON Acc.: 70%	**AVG Acc.:** 87.79%	**AVG Time:** 235.40sec.

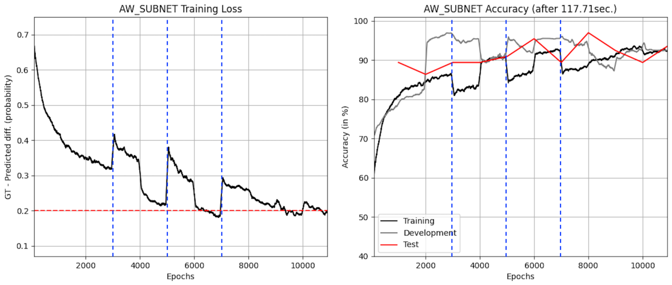

Fig. 10. Early-stopping spike examples

TABLE VIII
3^{rd} EXPERIMENT: *Time-Tracks*-PHONETIC CLASSIFICATION

Features	**Training**	**Early-Stopping**
10%, 50% SS, 80% formant ratios	epochs: 1000 (for each *batch-set*) Re-shuffling balancing tol.: 0.01	Loss thresh.: 0.15 Accuracy thresh.: 95%
Phonemes	**Test Accuracy** (%)	**Training times** (sec.)
ae	94.55	72.17
ah	91.80	156.37
aw	89.86	71.47
eh	93.74	540.39
er	93.43	28.05
ei	96.37	104.98
ih	94.55	97.20
iy	96.49	38.59
oa	93.49	49.43
oo	95.62	96.17
uh	90.98	649
uw	93.74	108.30
OCON Acc.: 90%	**AVG Acc.:** 93.72%	**AVG Time:** 167.68sec.

TABLE IX
Time-Tracks-SPEAKER CLASSIFICATION

Features	**Training**	**Early-Stopping**
10%, 50% SS, 80% formant ratios + *min-maxed* F0s	epochs: 1000 (for each *batch-set*) Re-shuffling balancing tol.: 0.01	Loss thresh.: $0.36, 0.08, 0.45$ Accuracy thresh.: $80\%, 97\%, 80\%\%$
Speakers	**Test Accuracy** (%)	**Training times** (sec.)
children	82.03	310.26
men	97.75	154.15
women	75.70	503.28
OCON Acc.: 80%	**AVG Acc.:** 85.15%	**AVG Time:** 322.56sec.

training loops duration. Attempts to enhance speakers gender boundaries by re-introducing F0s data, proved to be unsuccessful (AVG acc.: 88.80%, OCON acc.: 74%).

The most effective feature set consisted of 4 temporal tracks of 3 formant ratios (Table VIII), which significantly boosted accuracy, reduced training times, and mitigated side effects of Early-Stopping, approaching the accuracy goal referenced in [53] of 90% (Table XI).

B. Speaker recognition

We aim to determine the minimum amount of formant features required for identifying speakers' gender with our best model. The overall architecture was simplified to 3x One-Classes (Table IX): *men*, *women* and *children*, with normalized F0s reintroduced in the input set, to improve classes separability. Class-dependent Early-stopping criteria were defined due to the significant amount of adjustments required for proper training convergence: 0.36, 0.08, 0.45 loss thresholds, 80%, 97%, 80% accuracy thresholds (respectively for *children, male* and *women*).

Women and *children* classes faced challenges in loss minimization while the *men* MLP converged rapidly to low error rates (almost 100% of accuracy). These results suggest better class representation for *men* and confirmed known difficulties in aural partitioning between *children* and certain adult *female* voices (*aural* similarities). Evaluation conducted upon the entire dataset's inference revealed lower False Positives (FPs) for the *male* class and higher FP rates for *children* and *women* inferences.

The OCON model achieved a speaker genders recognition accuracy between 80% to 85%, suggesting potential increasing reliability according to higher time-tracks number (more than 3x formant ratios, per speaker). To finalize the statistical overview [54], [55] we provide related *confusion matrices* (Tables X), *Receiver Operating Characteristics* (ROC) curves analysis, *Area-Under-the-Curve* (AUC) computations and *Detection Error Tradeoffs* (DET) evaluations [55]–[57]: see related notebooks in our GitHub repository.

C. Energy efficiency

We focused on experimental sustainability grabbing insights from the Green-AI field [58], [59]. Using `CodeCarbon`, a custom Python-API for `Intel-RAPL` and `Nvidia-smi` libraries, we tracked CPU, disk, and RAM usage during model training. Energy metrics and estimated CO2 emissions were recorded in a `.csv` file and analyzed through a web-based applet developed by the GESSI research group (Universitat Politècnica de Catalunya), as part of the GAISSA research project. This analysis provided efficiency and accuracy labels (Fig.11) via HuggingFace database comparison, indicating

TABLE X
OCON NORMALIZED ACCURACY METRICS

One-Class	Accuracy	Precision	Recall	F1-Score
ae	0.9737	0.9568	0.9925	0.9744
ah	0.9625	0.9310	1.0000	0.9643
aw	0.9509	0.9110	1.0000	0.9534
eh	0.9631	0.9388	0.9928	0.9650
er	0.9605	0.9291	1.0000	0.9633
ei	0.9717	0.9542	0.9921	0.9728
ih	0.9742	0.9521	1.0000	0.9754
iy	0.9837	0.9688	1.0000	0.9841
oa	0.9664	0.9379	1.0000	0.9680
oo	0.9852	0.9720	1.0000	0.9858
uh	0.9593	0.9262	1.0000	0.9617
uw	0.9664	0.9441	0.9926	0.9677
One-Class	**TPs**	**FPs**	**FNs**	**TNs**
ae	133	6	1	126
ah	135	10	0	122
aw	133	13	0	119
eh	138	9	1	123
er	118	9	0	101
ei	125	6	1	115
ih	139	7	0	125
iy	124	4	0	117
oa	136	9	0	123
oo	139	4	0	128
uh	138	11	0	121
uw	135	8	1	124

TABLE XI
OCON NORMALIZED ROC-AUC/DET METRICS

One-Class	ER	FDR	FOR	NPV	AUC
ae	0.02	0.03	0.01	0.99	0.9986
ah	0.03	0.06	0.00	1.00	0.9866
aw	0.03	0.06	0.00	1.00	0.9980
eh	0.02	0.03	0.01	0.99	0.9934
er	0.02	0.03	0.00	1.00	0.9935
ei	0.03	0.05	0.01	0.99	0.9979
ih	0.03	0.05	0.00	1.00	0.9996
iy	0.01	0.02	0.00	1.00	0.9994
oa	0.04	0.07	0.00	1.00	0.9898
oo	0.01	0.01	0.00	1.00	1.0000
uh	0.03	0.05	0.00	1.00	0.9950
uw	0.03	0.06	0.01	0.99	0.9965

TABLE XII
OCON MODEL ENERGY PROFILE

Feature	Value
TOT Parameters	140412 (11701 each)
Estimated Size	0.6MB (0.05 each)
TOT mul-adds	$1.2 \cdot 10^5$
Dataset size	287KB (compressed)
Energy Measurement Date	2023-12-04
Energy Measurement Time	10:51:38
Profiling Software	`CodeCarbon`
Emissions	0.0081KgCO2
Emission rate	$3.7453 \cdot 10^{-6}$KgCO2h
Training Architecture	CPU (x2)
Model	Intel(R) Xeon(R) 2.20GHz
Cache size	56320KB
Power	42.5W
Energy	0.0255kWh
TOT RAM	12.67GB
RAM Power	4.7543W
RAM Energy	0.0028kWh
TOT Energy consumed	0.0284kWh
Cloud Service	Google Colab
Server Location	South Carolina (USA)
OS	`Linux5.15.120x86_64 glibc2.35`
Python	`3.10.12`
Ext. packages	`NumPy, MatPlotLib, PyTorch, SciPy, SKlearn`

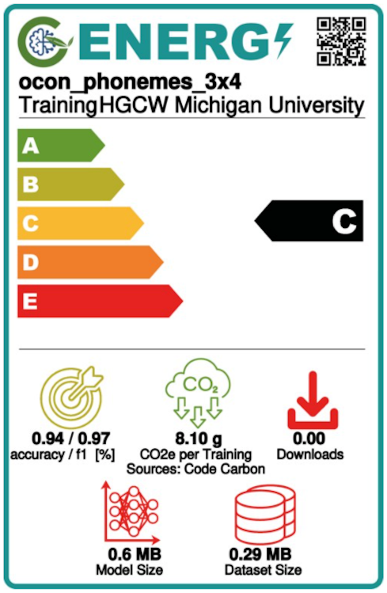

Fig. 11. OCON Energy label

strong sustainability of our pseudo-NAS approach, without compromising resulting accuracies.

The entire speakers task training cycle required approximately 36 minutes, with an average consumption of 42.5W for CPU and 4.27W for RAM. Carbon dioxide emissions (CO_2eq) were estimated as the product between grams of CO_2 emitted per KW-hour of electricity (0.025KW/h for CPU, 0.003KW/h for RAM) and the energy consumed by the computational infrastructure: resulting in 0.008Kg, with an emission rate of 3.75×10^{-6}Kg/s.

V. CONCLUSIONS AND FUTURE WORKS

We are aware that a single Perceptron can easily predict speech signal samples approximating LPA results [60]. Our model proposal can therefore be seen as an ad-hoc integration *head* for a complex Perceptron-based *formant neural framework*. Despite the active research on formant estimation leveraging convolutional and recurrent layers (backbone stages) [61], [62], we believe that our approach, employing pseudo-NAS/HP-T techniques entirely scripted and executed on Colab free-tier notebooks, can be broadly re-applied to effectively evaluate the efficiency of newer SoA CNNs building blocks (lihtweight-CNNs) in terms of parameters reduction and computational complexity.

Our foundational research model demonstrates high distributability, with each classifier independently re-trainable and sufficiently lightweight for constrained computational contexts (or hardwares), making it suitable for integration into pre-existent complex architectures and on-board sensory constrained hardware. Optimization techniques such as parameters *pruning* and *quantization* could further improve its memory consumption at inference time. Additionally, its modular structure allows for easy adaptation to different language and speakers grouping contexts (being more LGBTQIA+ friendly, despite the use of simple binary-labeled dataset). A pre-compiled `TorchScript` version of our classification stages, successfully run on testbed E2E sensing and processing devices (kindly provided by the developer company).

We try to challenge the notion that larger (and heterogeneous) datasets or complex (Transformer-based) models inherently yield better accuracies: asserting instead that our approach offers good generalizability and adaptability, despite known limitations in training sample size. While we encountered difficulty in finding extensive pre-processed datasets, we will re-validate our findings by expanding our dataset sources, potentially validating TI-MIT, UCLAPhoneticsSet and AudioSet.

Our proposal for linear features processing confirms that altering speech signal spectra in non-linear auditory-based ways it's not always optimal for *descriptive* speech modeling. However, we intend to reconsider solutions proposed in the existing literature.

Regarding sustainability, we're pleased to find that the CO2e emissions for fully retraining our model are just over half the emissions from the entire lifecycle of a *single cigarette*.

Future research could explore enhancing label (class) selection by applying training assurance scaling coefficients to output One-Class probabilities, aiming to increase model *reliability*. This approach involves analyzing epochs during which the classifier maintains the loss below specified Early-stopping thresholds. Further refinement of output probabilities could utilize derivatives of the loss curve, particularly useful in cases of too rapid training error minimization.

Acknowledgments

We extend our sincere thanks to the Electronic Music Department at A. Casella Conservatory (L'Aquila) for their invaluable assistance in developing this project. ChatGPT (OpenAI) was used here to support typos correction and text formatting.

References

[1] F. Alías and R. M. Alsina-Pagès, "Review of wireless acoustic sensor networks for environmental noise monitoring in smart cities," *Journal of Sensors*, vol. 2019, no. 1, p. 7634860, 2019. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1155/2019/7634860

[2] A. Pastor-Aparicio, J. Segura-Garcia, J. Lopez-Ballester, S. Felici-Castell, M. García-Pineda, and J. J. Pérez-Solano, "Psychoacoustic annoyance implementation with wireless acoustic sensor networks for monitoring in smart cities," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 128–136, 2020.

[3] J. Sueur and A. Farina, "Ecoacoustics: the ecological investigation and interpretation of environmental sound," *Biosemiotics*, vol. 8, no. 3, pp. 493–502, 2015.

[4] M. Markolf, M. Zinowsky, J. K. Keller, J. Borys, A. Cillov, and O. Schülke, "Toward passive acoustic monitoring of lemurs: Using an affordable open-source system to monitor phaner vocal activity and density," *International Journal of Primatology*, vol. 43, pp. 409 – 433, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:247535664

[5] L. Johnson, M. Butler, and S. Sesnie, "Optimizing detection of the bobwhite reproduction call using passive acoustic monitoring," *The Journal of Wildlife Management*, vol. 88, 10 2023.

[6] J. Segura-Garcia, J. M. A. Calero, A. Pastor-Aparicio, R. Marco-Alaez, S. Felici-Castell, and Q. Wang, "5g iot system for real-time psycho-acoustic soundscape monitoring in smart cities with dynamic computational offloading to the edge," *IEEE Internet of Things Journal*, vol. 8, no. 15, pp. 12 467–12 475, 2021.

[7] T. S. Gouvea, H. Kath, I. Troshani, B. Lüers, P. P. Serafini, I. B. Campos, A. S. Afonso, S. Leandro, L. H. Swanepoel, N. Theron, A. M. Swemmer, and D. Sonntag, "Interactive machine learning solutions for acoustic monitoring of animal wildlife in biosphere reserves," *International Joint Conference on Artificial Intelligence*, 2023.

[8] C. Rinaldi, F. Franchi, A. Marotta, F. Graziosi, and C. Centofanti, "On the exploitation of 5G multi-access edge computing for spatial audio in cultural heritage applications," *IEEE Access*, vol. 9, pp. 155 197–155 206, 2021.

[9] A. Marotra, C. Rinaldi, C. Centofanti, K. Kondepu, D. Cassioli, and F. Graziosi, "O-ran neutral hosting as a viable solution for first responders seamless connectivity," in *2023 International Conference on Information and Communication Technologies for Disaster Management (ICT-DM)*, 2023, pp. 1–4.

[10] J. Renaud, R. Karam, M. Salomon, and R. Couturier, "Deep learning and gradient boosting for urban environmental noise monitoring in smart cities," *Expert Systems with Applications*, vol. 218, p. 119568, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417423000696

[11] K. Marciniuk and B. Kostek, "Machine learning applied to acoustic-based road traffic monitoring," *Procedia Computer Science*, vol. 207, pp. 1087–1095, 2022, knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 26th International Conference KES2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050922010468

[12] G. Ciaburro and G. Iannace, "Improving smart cities safety using sound events detection based on deep neural network algorithms," *Informatics*, vol. 7, no. 3, 2020. [Online]. Available: https://www.mdpi.com/2227-9709/7/3/23

[13] D. Nieto-Mora, S. Rodríguez-Buritica, P. Rodríguez-Marín, J. Martínez-Vargaz, and C. Isaza-Narváez, "Systematic review of machine learning methods applied to ecoacoustics and soundscape monitoring," *Heliyon*, vol. 9, no. 10, p. e20275, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405844023074832

[14] L. Turchet, M. Lagrange, R. C., G. Fazekas, N. Peters, J. Østergaard, F. Font, T. Bäckström, and C. Fischione, "The internet of sounds: Convergent trends, insights and future directions," *IEEE Internet of Things Journal*, vol. 10, no. 13, pp. 11 264–11 292, 2023.

[15] S. Mondal and A. Das Barman, "Deep learning technique based real-time audio event detection experiment in a distributed system architecture," *Computers and Electrical Engineering*, vol. 102, p. 108252, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045790622004852

[16] D. Stefani and L. Turchet, "Real-time embedded deep learning on elk audio os," in *2023 4th International Symposium on the Internet of Sounds*, 2023, pp. 1–10.

[17] ——, "A comparison of deep learning inference engines for embedded real-time audio classification," in *25th International Conference on Digital Audio Effects (DAFx2022)*, 2022, pp. 1–10. [Online]. Available: http://www.lucaturchet.it/PUBLIC_DOWNLOADS/publications/conferences/A_Comparison_of_Deep_Learning_Inference_Engines_for_Embedded_Real-Time_Audio_Classification.pdf

[18] J. Jia1, P. Zhao1, and D. Wang, "A real-time voice activity detection based on lightweight neural network," *ArXiV*, 2023. [Online]. Available: https://arxiv.org/pdf/2405.16797

[19] S. Giacomelli, M. Giordano, and C. Rinaldi, "The OCON model: An old but gold solution for distributable supervised classification (in pubblication)," in *2024 IEEE Symposium on Computers and Communications (ISCC): workshop on Next-Generation Multimedia Services at the Edge: Leveraging 5G and Beyond (NGMSE2024)*, Paris, France, June 2024, p. 7.

[20] M. R. Hasan, M. M. Hasan, and M. Z. Hossain, "How many mel-frequency cepstral coefficients to be utilized in speech recognition? a study with the bengali language," *The Journal of Engineering*, vol. 2021, no. 12, pp. 817–827, 2021. [Online]. Available: https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/tje2.12082

[21] I. Al-Janabi, A. Sameer and A. Lateef, A. Azawii, “Applications of deep learning approaches in speech recognition: A survey,” in *Proceedings of International Conference on Computing and Communication Networks*. Springer Nature Singapore, 2022, pp. 189–196.

[22] S. Amandeep and S. Williamjeet, “A comprehensive survey on automatic speech recognition using neural networks,” *Multimedial Tools and Applications*, 2023.

[23] S. Fahmeeda, M. Ayan, M. Shamsuddin, and A. Amreen, “Voice based gender recognition using deep learning,” *International Journal of Innovative Research & Growth*, vol. 3, pp. 649–654, 12 2022.

[24] A. A. Alnuaim, M. Zakariah, C. Shashidhar, W. A. Hatamleh, H. Tarazi, P. K. Shukla, and R. Ratna, “Speaker gender recognition based on deep neural networks and resnet50,” *Wireless Communications and Mobile Computing*, vol. 2022, no. 1, p. 4444388, 2022. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1155/2022/4444388

[25] S. Mavaddati, “Voice-based age, gender, and language recognition based on resnet deep model and transfer learning in spectro-temporal domain,” *Neurocomputing*, vol. 580, p. 127429, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231224002005

[26] G. E. Peterson and H. L. Barney, “Control methods used in a study of the vowels,” *The Journal of the Acoustical Society of America*, vol. 24, no. 2, pp. 175–184, 2005.

[27] J. Hillenbrand, L. A. Getty, M. J. Clark, and K. Wheeler, “Acoustic characteristics of american english vowels,” *The Journal of the Acoustical Society of America*, vol. 97, no. 5, pp. 3099–3111, 1995.

[28] J. Garofolo, L. Lamel, J. Fisher, J. Fiscus, D. Pallett, and N. Dahlgren, “Darpa timit acoustic phonetic continuous speech corpus,” 1993.

[29] J. Hillenbrand and R. T. Gayvert, “Vowel classification based on fundamental frequency and formant frequencies,” *Journal of Speech, Language, and Hearing Research*, vol. 36, no. 4, pp. 694–700, 1993.

[30] X. Serra and J. Smith III, “Spectral modeling synthesis: A sound analysis/synthesis system based on a deterministic plus stochastic decomposition,” *Computer Music Journal*, vol. 14, no. 4, pp. 12–24, 1990.

[31] A. K. Syrdal and H. S. Gopal, “A perceptual model of vowel recognition based on the auditory representation of American English vowels,” *The Journal of the Acoustical Society of America*, vol. 79, no. 4, pp. 1086–1100, 1986.

[32] T. M. Nearey, “Applications of generalized linear modeling to vowel data,” in *Proc. 2nd International Conference on Spoken Language Processing (ICSLP 1992)*, 1992, pp. 583–586.

[33] J. D. Miller, “Auditory-perceptual interpretation of the vowel,” *The Journal of the Acoustical Society of America*, vol. 85, no. 5, pp. 2114–2134, 1989.

[34] E. Zwicker and E. Terhardt, “Analytical expressions for critical-band rate and critical bandwidth as a function of frequency,” *The Journal of the Acoustical Society of America*, vol. 68, no. 5, pp. 1523–1525, 1980.

[35] H. Traunmüller, “Perceptual dimension of openness in vowels,” *The Journal of the Acoustical Society of America*, vol. 69, no. 5, pp. 1465–1475, 1981.

[36] G. Fant, *Acoustic Theory of Speech Production, With Calculations based on X-Ray Studies of Russian Articulations*. Berlin, Boston: De Gruyter Mouton, 1971.

[37] W. Koening, “A new frequency scale for acoustic measurements,” BTL, New Jersey, Tech. Rep., 1949.

[38] R. Shinde and V. Pawar, “Vowel classification based on lpc and ann,” *International Journal of Computer Applications*, vol. 50, no. 6, pp. 27–31, 2012.

[39] T. Kohonen, K. Mäkisara, and T. Saramäki, “Phonotopic maps—Insightful representation of phonological features for speech recognition,” in *Proceedings of the 6th International Conference on Pattern Recognition*. IEEE Computer Society Press, 1984, pp. 182–185.

[40] S. Haskey and S. Datta, “A comparative study of ocon and mlp architectures for phoneme recognition,” in *Proc. 5th International Conference on Spoken Language Processing (ICSLP 1998)*, 1998.

[41] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1026–1034.

[42] I. Chang Jou, Y.-J. Tsay, S.-C. Tsay, Q.-Z. Wu, and S.-S. Yu, “Parallel distributed processing with multiple one-output back-propagation neural networks,” in *1991 IEEE International Symposium on Circuits and Systems (ISCAS)*, vol. 3, 1991, pp. 1408–1411.

[43] M. M. Moya and D. R. Hush, “Network constraints and multi-objective optimization for one-class classification,” *Neural Networks*, vol. 9, no. 3, pp. 463–474, 1996. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0893608095001204

[44] R. Chalapathy, A. K. Menon, and S. Chawla, “Anomaly detection using one-class neural networks,” *CoRR*, vol. abs/1802.06360, 2018. [Online]. Available: http://arxiv.org/abs/1802.06360

[45] P. Perera and V. M. Patel, “Learning deep features for one-class classification,” *Trans. Img. Proc.*, vol. 28, no. 11, p. 5450–5463, nov 2019. [Online]. Available: https://doi.org/10.1109/TIP.2019.2917862

[46] L. Mezher, “Design and implementation maxnet neural network with matlab,” *International Journal of Wireless Communications and Network Technologies*, vol. 8, pp. 56–62, 06 2019.

[47] T. Tieleman and G. Hinton, “Rmsprop - divide the gradient by a running average of its recent magnitude,” *Journal of Machine Learning Research*, vol. 4, pp. 26–31, 2012. [Online]. Available: https://www.cs.toronto.edu/∼tijmen/csc321/slides/lecture_slides_lec6.pdf

[48] D. Kingma and L. Ba, “Adam: A method for stochastic optimization,” in *International Conference on Learning Representations ICLR*, 2015. [Online]. Available: https://arxiv.org/abs/1412.6980

[49] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[50] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in *Proceedings of the 32nd International Conference on International Conference on Machine Learning*, ser. ICML’15, vol. 37, 2015, pp. 448–456.

[51] A. E. Hoerl and R. W. Kennard, “Ridge regression: Biased estimation for nonorthogonal problems,” *Technometrics*, vol. 42, no. 1, pp. 80–86, 2000.

[52] Y. Bai, E. Yang, B. Han, Y. Yang, J. Li, Y. Mao, G. Niu, and T. Liu, “Understanding and improving early stopping for learning with noisy labels,” 2021.

[53] C. Sridhar and A. Kanhe, “Performance comparison of various neural networks for speech recognition,” in *Journal of Physics Conference Series, 4th National Conference on Communication Systems (NCOCS 2022)*, vol. 2466, 2023, p. 012008. [Online]. Available: https://dx.doi.org/10.1088/1742-6596/2466/1/012008

[54] A. Tharwat, “Classification assessment methods,” *Applied Computing and Informatics*, vol. 17, no. 1, pp. 168–192, 2020.

[55] D. Powers, “Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation,” *International Journal of Machine Learning Technology*, vol. 2, no. 1, pp. 37–63, 2011.

[56] S. Morasca and L. Lavazza, “On the assessment of software defect prediction models via roc curves,” *Empirical Software Engineering*, vol. 25, no. 5, pp. 3977–4019, 2020.

[57] J. Navratil and D. Klusacek, “On linear dets,” in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP ’07*, vol. 4, 2007, pp. 229–232.

[58] P. Henderson, J. Hu, J. Romoff, E. Brunskill, D. Jurafsky, and J. Pineau, “Towards the systematic reporting of the energy and carbon footprints of machine learning,” *Journal of Machine Laearning Research*, vol. 21, pp. 1–43, 2020. [Online]. Available: https://arxiv.org/pdf/2002.05651.pdf

[59] R. Schwartz, J. Dodge, N. A. Smith, and O. Etzioni, “Green ai,” *Communications of the ACM*, vol. 63, no. 12, pp. 54–63, 2020.

[60] I. Ibrahim and M. El-Adawy, “Determination of the lpc coefficients using neural networks,” in *Proceedings of ICNN’95 - International Conference on Neural Networks*, vol. 5, 1995, pp. 2669–2673.

[61] P. Alku, S. R. Kadiri, and D. Gowda, “Refining a deep learning-based formant tracker using linear prediction methods,” *Computer Speech & Language*, vol. 81, p. 101515, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0885230823000347

[62] Y. Dissen, J. Goldberger, and J. Keshet, “Formant estimation and tracking: A deep learning approach,” *The Journal of the Acoustical Society of America*, vol. 145, no. 2, pp. 642–653, 2019. [Online]. Available: https://doi.org/10.1121/1.5088048