The OCON model: an old but gold solution for distributable supervised classification

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 introduces to a structured application of the *One-Class* approach and the *One-Class-One-Network* model for supervised classification tasks, specifically addressing a *vowel phonemes classification* case study within the Automatic Speech Recognition research field. Through *pseudo*-Neural Architecture Search and Hyper-Parameters Tuning experiments conducted with an *informed* grid-search methodology, we achieve classification accuracy comparable to nowadays complex architectures (90.0 - 93.7%). Despite its simplicity, our model prioritizes generalization of language context and distributed applicability, supported by relevant statistical and performance metrics. The experiments code is openly available at our GitHub.**

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

I. Introduction

The intricate tapestry of human speech is woven through the delicate interplay of various phonemes, each contributing its unique acoustic signature to the rich spectrum of linguistic expression. Among these, vowels stand as fundamental building blocks, playing a pivotal role in shaping the intelligibility and emotive nuances of language. The classification of vowels within the phonetic landscape has been a subject of enduring interest and significance in the realm of linguistics and speech science, due to the broad range of applications, e.g.: language learning and pronunciation assessment [1], dialectology and sociology [2], forensic speaker identification [3], assistive technologies [4], emotion recognition [5], up to brain computer interface [6]. Advancements in the field of vowel recognition have reached a *state-of-the-art* (SoA), leveraging cutting-edge Machine Learning (ML) algorithms and sophisticated signal processing techniques to achieve unprecedented accuracy in deciphering and understanding vocal nuances. Techniques exploited for this purpose span from features extraction in time and transformed domains [7], to advanced ML solutions, from Neural Networks (NNs) to Deep Learning (DL) algorithms and models applied for general purpose Automatic Speech Recognition (ASR), as in [8], [9].

Our proposal aims to assess the reliability of a simplified Neural Architecture Search (NAS) and Hyper Parameters Tuning (HPs-T) combination for designing NNs feature abstraction layers. We propose a modular model, the "One-Class One-Network" (OCON), comprising parallelized binary classifiers focused on simpler phonetic recognition sub-tasks. We evaluate available data constraints and task complexities w.r.t. the current SoA, aiming for a *shallow* and optimizable architecture with a sustainable and straightforward (re)-training cycle. Additionally, we determine the minimum number of formant features needed to achieve current accuracy levels in phonetic recognition.

This work was partially supported by the EU under the Italian NRRP of NextGenerationEU, partnership on "Telecommunications of the Future" (PE00000001 - program "RESTART") and by OPTIMIST project funded from the EXCELLENT SCIENCE - Marie Skłodowska-Curie Actions under grant agreement n° 872866

II. Methodologies

Following a standard DL experiment routine, we begin by gathering a reliable audio *dataset* with heterogeneous phonetic representations and gender diversity among speakers: we initially constrained our linguistic research context to the *General American English* case, as defined by the International Phonetic Association (IPA). Within this contextual dataset production, we encounter several good examples of pre-processed datasets (Sec. II-A) where pre-arranged phraseological (or even specific words like */hVd/*, vowels between an "*h*" and a "*d*") speech segments were already recorded, analyzed and processed so as to elicit specific meaningful features: *formant* frequencies.

These features set represent the input data suitable for our NN model, and are usually retrieved by means of standardized key-steps, which can be summarized in: (*1*) manually or automatically segment speech signals in *semantic frames* according to a pre-arranged semantic grid (words/phonemes and silences); (*2*) analyze isolated fragments by means of Linear Predictive Analysis/Coding (LPA/C) to extract a smoothed time-frequency spectral estimation (we are interested in the frame-by-frame aggregated spectral envelope); (*3*) extract first N spectral *peaks* by means of whatever *peaks estimation* algorithm keeping tracks (*contouring*) of frame-wise continuities.

Next, we'll manipulate and refine these formant frequency tracks to generate a suitable data vector (*features*) for the input stage of our NNs. This pre-processing step plays a vital role in enabling networks to learn abstract representations effectively, thereby maximizing recognition accuracy results.

A. Datasets availability

From a preliminary dataset review, the initial */hVd/* vowel phoneme dataset, by Peterson and Burney (PB) [10] featured

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`

10 vowels spoken by 33 men, 28 women and 15 children. While being crucial for foundational phonetics research e.g. [11], it faced criticism for its measurement and reporting limitations [12].

In contrast, the *HGCW* database [13] includes 45 men, 48 women, 25 boys, and 19 girls' phoneme recordings, meticulously screened for non-native English, speech/language issues, respiratory infections, and inadequate hearing. Released openly in `.dat` format, it already includes formant analysis and fundamental frequency time contourings. Compared to the PB database, the HGCW dataset doubles the sample size, improves samples audio quality and addresses real-word complexity by including overlapping boundaries among available phoneme classes.

The Texas Instruments & Massachusetts Institute of Technology (TI-MIT) *Corpus of Read Speech* dataset [14], originally developed for military research on telephone communications, is the most commonly used dataset for ASR applications. It comprises 6300 utterances from 10 sentences, spoken by 630 speakers, with a notable gender imbalance (70% male, 30% female). One of its most important spin-offs is the VTR database [15], which is specifically tailored for vocal tract resonances research and formant frequencies estimation. The core-test set of the VTR includes 192 utterances, with 24 speakers (each uttering 5 phonemically-compact and 3 phonetically-diverse sentences), while the training set consists of 346 utterances with 173 speakers (each uttering 1 phonemically-compact and 1 phonetically-diverse sentence).

Possible dataset references thus exist for evaluating *formant-based* phonetic classification algorithms, being often underutilized: researchers tend to favor occasional *ad-hoc* sampling, hindering comparative and generalization studies.

Recognizing limitations of existing datasets for robust generalized solutions, we opted for the HGCW dataset (which offers the higher phonetic complexity level) to expedite the data retrieval pipeline relying on pre-extracted formant data. In this way we aim to promote consistency with literature and streamline results evaluation.

B. Features analysis & processing

The HGCW dataset filename structure (Table I) contains encoded phonetic and speaker features crucial for preliminary statistical analysis. Through ad-hoc `Python` scripts, we discovered that only 1597 out of 1668 samples could be effectively used for our task due to null features presence for some samples, caused by authors algorithm failures. To maintain learning consistency, these samples were filtered out, resulting in an additional under-representation of certain phoneme and speaker classes (Table II). Fundamental frequency contours ($F0$) were estimated using a 2-way auto-correlation/zero-crossing pitch tracker with a halving/doubling evaluation subroutine [11], while formants were estimated using LPA spectra estimation and peak retrieval with parabolic interpolation [16]. The resulting frequency trajectories underwent further refinement using an interactive audio spectral editor for manual discontinuity examination and interpolation.

TABLE II
HGCW ACTUAL CLASSES STATISTICS

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

We created various experimental sub-structures of the original dataset, categorizing each sample based on: (*1*) *Phonemes grouping*, including $F0$ and the first 3 formant frequencies obtained at the *steady state* (SS); (*2*) *Speaker grouping*, segregating samples by gender (*men*, *women*, and *children*) with the same features as before; (*3*) *Phonemes grouping*, incorporating $F0$ and a total of 12 formant frequency values, with the first 3 formants sampled at 10%, 50%, SS, and 80% of the total duration of the vowel nucleus.

We analyzed and summarized classification algorithms exclusively evaluated on the PB and/or HGCW datasets features, to establish a consistent reference baseline for our work (Table III): Linear Discriminant Analysis (LDA) [17] and Generalized Linear Regression Models (GLM) [18] emerged as the prominent and most involved statistical-ML approaches, combined with innovative formant features processing as the *3D-auditory target zones* framework, expressed by means of the logarithm of formant distances [19]. Other studies [20] adopted canonical auditory frequency transforms including: the Bark scale [21], [22], a *technical* Mel scale approximation [23], and a *lin-to-log* frequency approximation [24].

NNs research in phonetic recognition has predominantly centered on using LPA coefficients [25] or spectral/*cepstral* derived features [26], often employing far more complex convolutional and/or recurrent stages. The only phonetic-OCON research retrieved [27] reported improvements exclusively over TI-MIT data, combined with LPC features. Due to this scarcity of comparative literature, we set a target *average accuracy* of 90% aiming to improve results reported in [11], [13].

Considering the significant variation in $F0$s within speakers due to physiological factors and related pitch variations due

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%

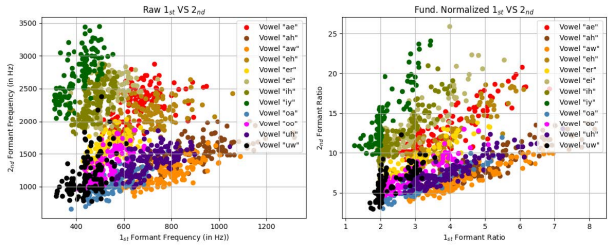

Fig. 1. HGCW dataset normalization (2D formantic projection)

to *prosody*, we introduce linear formants normalization w.r.t. $F0$s. No prior usage of this pre-processing method were found, which appears to enhance class segregation by directly expressing distances in the linear frequency space (Fig. 1). To foster NNs training convergence, we applied *min-max* scaling to normalize the entire feature set. We also assessed *formant ratios* Probability Mass Distributions to determine Z-score (standardization) feasibility but they exhibited a consistent tendency towards skewed *Poisson* or *Log-normal* distributing.

For enhanced data portability and reusability, we encoded all preprocessed features in a binary `NumPy` open-source compressed format (`.npz`) to preserve data resolution.

III. IMPLEMENTATION

In this section, we introduce the NN architectures for our experimentation, beginning with the Multi-Layer Perceptron (MLP). Since we are working with pre-processed features, our heuristic search experiments will focus on the architecture topology and characteristics of the MLP. We will then present our OCON proposal, which models multi-output classification tasks using multiple independent copies of the same optimized MLP architecture setup. These configurations are derived through simplified and *informed* NAS (*pseudo*-NAS) combined with HPs-T: in NNs 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 known as *fully connected* (FC)-layers or *Feed-forward* NNs, are simply *Perceptrons* (*neurons*) stacked in vertical layers (*shallow* NNs), whose function is:

$$y_n = \varphi\langle x, w_k\rangle = x^\top w_k = \varphi\left(\sum_{k=0}^{K} x_n w_k\right) \tag{1}$$

where x_n are the input features, w_k a set of scaling coefficients (*weights*) and $\varphi(\cdot)$ a non-linear function (often *activation*): in our case a standard ReLU [28].

The One-Class-One-Network (OCON) model [29], introduced in the '90s, served as a *parallel distributed processing* solution to overcome limitations of architectures which required full re-training when altering dataset classes. Today, OCON resembles a simplified form of architecture *ensembling* e.g. when multiple complex networks are combined through other networks or algorithms, to enhance overall models accuracy. In this case, a multi-output classification is distributed across independent sub-networks, each functioning as binary classifier: an approach widely used in anomaly detection and computer vision [30], [31]. In our case, we split a 12-phoneme classification task into a bank of 12 independent classifiers with identical architectural topology, seeking an *optimal average* architecture estimate.

If a single output label is needed, a context-specific output algorithm must be devised. However, we find the classification *logits* vector more beneficial for understanding phonetic class boundaries and feature complexities.

While no literature references were found regarding OCON-specific output algorithms, the *argument of the maxima* (*ArgMax*) approach can be employed, as it typically returns a single value representing the first occurrence of the maximum, when multiple occurrences exist.

During supervised training, each sample label needs to be binarized (*one-hot encoded*) according to the class architecture, while features are simultaneously fed to all classifiers. To accomplish this, we developed a custom one-hot encoding technique (Alg. 1) to transform labels based on the incoming `True`-One-class. Additionally, due to previous observations (Table II), we needed to slightly down-sample resulting subsets, injecting further under-representation, to achieve nearly perfect training class balance.

Algorithm 1 HGCW One-Hot encoding

Require: c ▷ True-class index
Require: s ▷ Phoneme groups size
Require: $\mathcal{X}$ ▷ Features dataset
Require: $\mathcal{Y}$ ▷ Dataset labels
class$_1$ $= \mathcal{X}(c)$ ▷ Initialize True-class subset
size $=$ length(class$_1$) ▷ Extract True-class size
classes$_0$ $=$ list[ ] ▷ Initialize False-class subsets
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**
    class$_0$ $= rand(\mathcal{X}_k$, sub-sizes) ▷ Random downsampling
    classes$_0$.append(class$_0$)
  else
    pass
  end if
end for

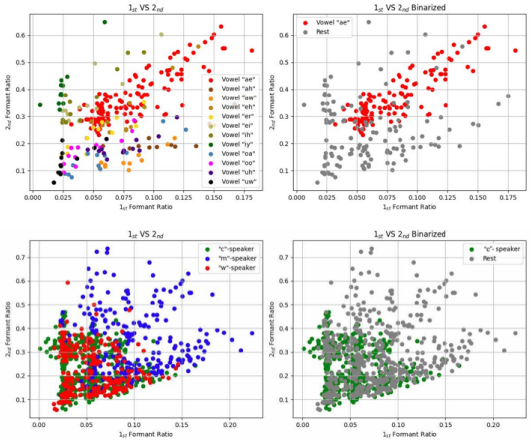

Fig. 2. HGCW dataset *One-Hot encoding* examples

Alg. 1 creates subsets of the HGCW dataset, with sizes determined by the `True`-class original size, ensuring balanced outliers by allocating the same number of samples for all `False`-classes, evenly distributed among the remaining 11. If `True`-class sizes are not divisible by 11, a variability of 1 to 3 samples was considered acceptable. The one-hot encoding routine is executed once per architecture training cycle, before *train-eval-test* splitting and *mini-batch* partitioning. Speaker-based encoding for *male*, *female*, and *children* classes is achievable as well (Fig. 2).

B. Pseudo NAS & HPs-T heuristic search

The term *pseudo*-NAS refers to the *a prìori* constraint applied to the architecture topology (MLP), as reported in Sec. III-A. This evaluation assesses the optimal number of layers and nodes (per layer) required to effectively solve both phoneme and gender classification tasks. On the other hand, *Grid-based* HPs search is a statistical method where all possible combinations of NNs HPs are independently sampled and evaluated through direct learning procedures. While theoretically effective, it can become time-consuming due to the exponentially increasing computational requirements for narrowing resolutions: in a standard procedure, all possible combinations must be tested before selecting the *optimal* one. In our scenario, we achieved a good trade-off establishing independent resolutions for each HP beforehand, employing an *informed* iterative approximation approach, outlined as follows: (*1*) define a specific HPs subset (not necessarily all at once, eventually fix others); (*2*) sample each HP with an arbitrary resolution; (*3*) test each HPs combination and resulting *temporary best estimates* can be considered: *inheritable* in following heuristic stages (*optimal estimates*) or used to narrow parameters resolution sampling around *local good estimates* (searching for better sets); (*4*) go to step (*2*) and re-iterate as much as needed.

We acknowledge that this simplified approach *roughly* approximates theoretical grid-search and that may lead to misleading local minima in the model costs. However our aim is to find

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* [28], $b = 0$) 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**

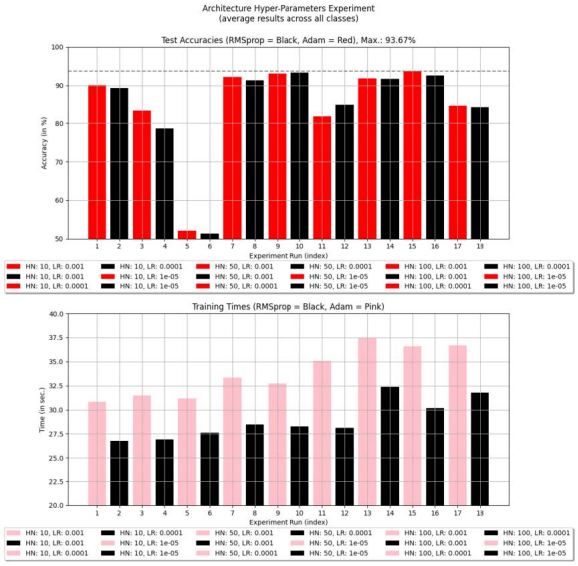

Fig. 3. $1st$ heuristic stage results

an *average* One-Class topology in a computationally feasible manner. Our heuristic learning experiments implies dataset partitioning into train (70%), dev (15%), and test (15%) sets with seeded initial states. We measure accuracy and mini-batch training times, averaging results over a 3-fold validation procedure for each One-Class

In the 1^{st} heuristic stage (Table IV), which combined pseudo-NAS and HP-T experiments, two architectural combinations (10^{th} and 15^{th}) yielded similar average accuracies (93.67%, Fig. 3). The RMSProp optimizer [32] appears to mitigate the increasing trend in learning times better than Adam [33] does, but we opted for the top-performing configuration (HL: 1, HN=100, LR=10^{-4}, Backprop: Adam).

Next heuristic stages were structured to evaluate gradual introduction of *regularization* techniques, evaluating potential benefits. In *DropOut* [34] tests (Table V, Fig. 4), we found that the best accuracy run (the 1^{st}) is also among the fastest. We achieved 0.19% increase in prediction accuracy at the expense of +3.4sec. in training time: resulting DropOut probabilities are 80% for input nodes and 50% for hidden nodes. In *Batch-norm* [35] tests, after re-evaluating LRs, it was confirmed

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**

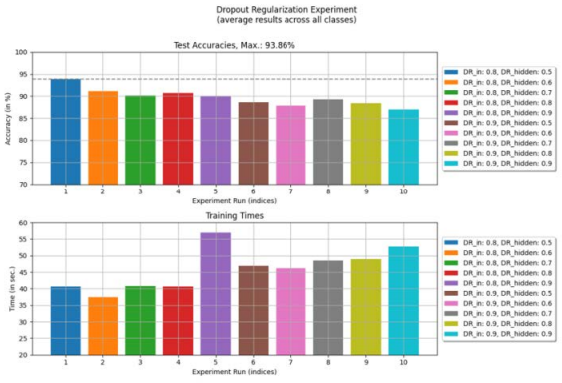

Fig. 4. 2^{nd} heuristic stage results (Dropout)

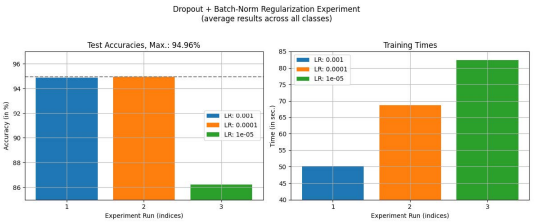

Fig. 5. 3^{rd} heuristic stage results (Batch-Norm)

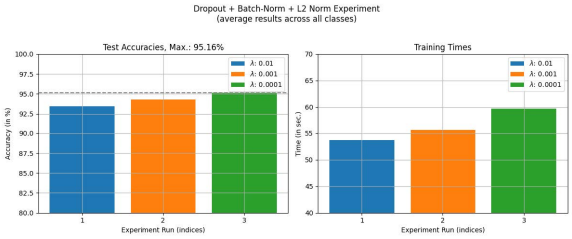

Fig. 6. 4^{th} heuristic stage results (L2-Norm)

that 10^{-4} yielded the best results: a significant $+1.1\%$ in test accuracy, although average training times were nearly doubled. In *L2-Norm* [36] (*Ridge penalty*) tests (Table V), the best λ (*weight decay*) value was found to be 10^{-4} (Fig. 6), resulting in a $+0.19\%$ in average accuracy. Interestingly, this improvement was accompanied by a decrease in training times (now below 60sec.) The resulting overall averaged One-Class proposal can be reviewed in Table VI.

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}

IV. MODEL TRAINING & RESULTS DISCUSSION

A parallelized set of independently trainable One-Class architectures was scripted and CPU runtimes (Google Colab) were utilized to efficiently measure isolated training cycle performance and resources consumption. The OCON architecture relies on the backprop loop of each MLP for consistent learning, while its *inference* involves extracting sample features array, computing 12 parallel one-hot encodings and conducting an ArgMax search to find the maximum value (predicted label) within the 12-logit probabilities vector.

It follows a review of the *phoneme recognition* experiments, where we tested the efficiency of each dataset substructure (Sec. II-B). An *Early-stopping* training strategy [37] was adopted, setting a 2-variables-match escape condition: a *minimum loss* threshold (averaging among last 50 training samples loss) and a *minimum test accuracy* threshold (w.r.t. last batch results). These variables were further empirically assessed to ensure practical convergence of training cycles, with each cycle not exceeding 25-30min. While the learning phases may not be optimally exploited, they were deemed satisfactory for our study's purposes.

A. Phonemes recognition

Initially we re-evaluated the OCON model using the SS dataset variant (Table VII):

few loss functions and training accuracy curves reached visible *plateaus*, with periodic artifacts (*spikes*) indicating batch re-shuffling instances. Surprisingly, the *er* and *iy* phoneme classes resulted well-represented, showing little or no changes in trend after encoding or re-shuffling: this suggests that under-represented classes remained the best represented (Table VII). Classification accuracies were evaluated over the whole dataset (binary classification threshold set at 0.5), with certain MLPs achieving a sufficient probabilities segregation: however, errors were high between *aurally closest classes* (e.g.: *ae* and *eh*, *er* and *ei*).

Hidden dataset biases, such as *children vs women utterances* (exceptional similarities in formantic disposition) were re-examined, leading to slight improvements in class boundaries upon filtering out (due to lack of insights about *vocal maturity* of children samples) albeit increasing training duration: attempts to re-introduce $F0$s data to improve gender recognition

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
AVG Acc.: 87.79%	**AVG Time:** 235.40sec.	**OCON Acc.:** 70%

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
AVG Acc.: 93.72%	**AVG Time:** 167.68sec.	**OCON Acc.:** 90%

were unsuccessful (AVG acc.: 88.80%, OCON acc.: 74%). The most performative features-set comprised temporal-tracks of formant ratios (Table VIII), leading to significant improvements in accuracy, reduced training times and mitigated Early-stopping side-effects, approaching the reference [38] accuracy goal of 90% (Table IX).

V. CONCLUSIONS AND FUTURE WORKS

We know that a single *Perceptron* can actually predict a speech signal sample [39] approximating LPA results: our model proposal could be considered then an *ad-hoc head*-integration for a complex *Perceptron-based* formant neural framework. We are aware of researches on formants estimation [40], [41] leveraging convolutional and recurrent layers (*backbone stages*): despite accuracies achieved, we believe our approach, employing pseudo-NAS/HP-T techniques completely hand-scripted and fully conducted on Colab notebooks, could be broaderly re-applied to effectively assess newer SoA NNs building blocks efficiency, in terms of parameters reduction.

TABLE IX
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

Our model exhibits high distributability, with each classifier independently re-trainable and sufficiently lightweight to be suitable for constrained computational contexts and integration into complex architectures. Optimization techniques, like *parameters prunings* and *quantization* could further enhance its memory consumption at inference time. Additionally, its modular structure facilitates adaptation into different language contexts. We challenge the notion that larger datasets or models inherently yield better accuracies, asserting that our approach offers good generalizability, despite observed limitations in training sample size. We encountered difficulty finding more extensive pre-processed datasets but we aim to validate our findings by expanding our dataset sources, potentially including datasets like TI-MIT, UCLAPhoneticsSet and/or AudioSet. Our proposal for linear features processing demonstrates that altering speech signal spectra in auditory-based non linear ways isn't always the optimal method for descriptive speech modeling. However, we intend to re-examine solutions from existing literature.

Future research could focus on enhancing label selection, considering the potential for *increased reliability* by applying *training assurance* scaling coefficients to output One-Class probabilities: this could involve analyzing epochs spent by the classifier to maintain loss below the specified Early-stopping threshold and considering derivatives of the loss curve to further refine the output probabilities, especially in cases of too rapid training error minimization.

ACKNOWLEDGMENTS

We express our gratitude to the Electronic Music (meAQ) Dpt. at A. Casella Conservatory (L'Aquila) for their assistance in developing this project. ChatGPT (OpenAI) was used here to support typos correction and text formatting.

REFERENCES

[1] A. Asif, H. Mukhtar, F. Alqadheeb, H. F. Ahmad, and A. Alhumam, "An approach for pronunciation classification of classical arabic phonemes using deep learning," *Applied Sciences*, vol. 12, no. 1, 2022. [Online]. Available: https://www.mdpi.com/2076-3417/12/1/238

[2] X. Liu, J. Tao, Y. Han, C. Wang, X. Zheng, and Z. Wen, "Which phonemes will distinguish the different regions within the same dialect?" in *2021 24th Conference of the Oriental COCOSDA International Committee for the Co-ordination and Standardisation of Speech Databases and Assessment Techniques (O-COCOSDA)*, 2021, pp. 152–157.

[3] T. B. Amin, P. Marziliano, and J. S. German, "Glottal and vocal tract characteristics of voice impersonators," *IEEE Transactions on Multimedia*, vol. 16, no. 3, pp. 668–678, 2014.

[4] S. Chandrakala, S. Malini, and S. V. Veni, "Histogram of states based assistive system for speech impairment due to neurological disorders," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, pp. 2425–2434, 2021.

[5] O. Caballero, "Recognition of emotions in mexican spanish speech: An approach based on acoustic modelling of emotion-specific vowels," *TheScientificWorldJournal*, vol. 2013, p. 162093, 01 2013.

[6] L. C. Sarmiento, S. Villamizar, O. López, A. C. Collazos, J. Sarmiento, and J. B. Rodríguez, "Recognition of eeg signals from imagined vowels using deep learning methods," *Sensors*, vol. 21, no. 19, 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/19/6503

[7] 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

[8] 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.

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

[10] 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.

[11] 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.

[12] M. G. Di Benedetto, "Frequency and time variations of the first formant: Properties relevant to the perception of vowel height," *The Journal of the Acoustical Society of America*, vol. 86, no. 1, pp. 67–77, 1989.

[13] 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.

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

[15] L. Deng, X. Cui, R. Pruvenok, J. Huang, S. Momen, Y. Chen, and A. Alwan, "A database of vocal tract resonance trajectories for research in speech processing," in *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, vol. 1, 2006, pp. I–I.

[16] 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.

[17] 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.

[18] 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.

[19] 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.

[20] A. K. Syrdal, "Aspects of a model of the auditory representation of american english vowels," *Speech Communication*, vol. 4, no. 1, pp. 121–135, 1985.

[21] 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.

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

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

[24] W. Koening, "A new frequency scale for acoustic measurements," BTL, New Jersey, Tech. Rep., 1949.

[25] 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.

[26] 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.

[27] 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.

[28] 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.

[29] 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.

[30] 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

[31] 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

[32] 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

[33] 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

[34] 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.

[35] 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.

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

[37] 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.

[38] 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

[39] 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.

[40] 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

[41] 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