Noname manuscript No.
(will be inserted by the editor)

Noise invariant feature pooling for the Internet of Audio Things

Christoforos Nalmpantis · Lazaros Vrysis · Danai Vlachava · Lefteris Papageorgiou · Dimitris Vrakas

Received: date / Accepted: date

Abstract This manuscript discusses the robustness to noise of deep learning models for two audio classification tasks. The first task is a speaker recognition application, trying to identify five different speakers. The second one is a speech command identification where the goal is to classify ten voice commands. These two tasks are very important to make the communication between humans and smart devices as smooth and natural as possible. The emergence of smart home devices such as personal assistants and the deployment of audio based applications in noisy environments raise new challenges and reveal the weaknesses of existing speech recognition systems. Despite the

This research has been co-financed by the European Regional Development Fund of the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH–CREATE–INNOVATE (project code: T1EDK-00343 (95699) - Energy Controlling Voice Enabled Intelligent Smart Home Ecosystem).

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

C. Nalmpantis
Aristotle University of Thessaloniki, Thessaloniki, Greece
E-mail: christofn@csd.auth.gr

L. Vrysis
Aristotle University of Thessaloniki, Thessaloniki, Greece
E-mail: lvrysis@auth.gr

D. Vlachava
International Hellenic University, Thessaloniki, Greece
E-mail: danai.vlachava@gmail.com

L. Papageorgiou
Entranet Ltd, Thessaloniki, Greece
E-mail: papageorgiou@entranet.gr

D. Vrakas
Aristotle University of Thessaloniki, Thessaloniki, Greece
E-mail: dvrakas@csd.auth.gr

advances of deep learning in audio tasks, most of the proposed architectures are computationally inefficient and very sensitive to noise. This research addresses these problems by proposing two neural architectures that incorporate a novel pooling operation, named entropy pooling. Entropy pooling is based on the principle of maximum entropy. A detailed ablation study is conducted to evaluate the performance of entropy pooling against the classic max and average pooling layers. The neural networks that are developed are based on two architectures, convolutional networks and residual ones. The study shows that entropy based feature pooling improves the robustness of these architectures in the presence of noise.

Keywords internet of audio things · IoAuT · robust deep learning · noise robustness · entropy pooling · speech commands · speaker recognition

1 Introduction

The Internet of Audio Things (IoAuT) is an emerging sub-field of the Internet of Things (IoT) and has attracted many researchers from different disciplines. It lies in the intersection of Internet of Things, sound recognition, machine learning and human-computer interaction [47].

From the humans point of view, speech is the intrinsic way of communication. Yet, most machine-to-human user interfaces have been restricted to very limited voice interactions or other methods such as touchscreens. IoAuT addresses the interdisciplinary challenges that need to be solved in order to pave the way for new opportunities and applications. Turchet et al. [47] present a taxonomy of these challenges based on the following classes: connectivity, interoperability and standardization, machine analysis of audio content, data collection and representation of audio content, edge computing, synchronization, privacy and security and Audio Things design. In the context of machine analysis of audio content and edge computing, the main challenges we face are large volumes of data, presence of noise, limitations of computing resources and latency during inference.

IoAuT will be generating a huge amount of audio data which will be hard to clean and annotate. Learning from noisy data is a common problem for many machine learning tasks. Even when a model is trained with noisy data, there is no guarantee it will be robust when deployed in the real-world [32, 6, 14]. Furthermore, IoT devices have limited storage, which makes the deployment of a state-of-the-art neural network prohibitive. Modern deep neural networks are usually trained and tested using very powerful GPUs and thus they are not suitable to run on embedded devices with restricted computational and energy resources. Finally, in order to make an IoAuT application to meet the requirements of the end users, it has to perform under low bandwidth connectivity and in real-time.

The goal of this research is to develop efficient deep learning solutions that are robust to nuances in audio classification applications. The tasks for evaluation of the proposed models are speaker recognition and speech commands

identification. The former one regards the recognition of five different speakers using the popular dataset "Speaker Recognition Dataset Prominent Leaders Speeches". The latter one aims to identify ten voice commands from the popular dataset "Google Speech Commands" that is provided by Google. Given these datasets, two novel neural architectures are developed. The performance of these models is on par with equivalent state-of-the-art models, but they are computationally lighter due to the reduced learning parameters. Furthermore, the proposed models utilize entropy based feature pooling, which improves their robustness against noise.

The key contributions of this manuscript are summarized as follows. Firstly, two novel neural architectures are proposed for two different audio classification tasks. To the best of the authors' knowledge it is the first time that entropy based feature pooling is incorporated in deep neural networks for audio classification. Secondly, a novel framework that includes four different noisy environmental setups is suggested. The framework is used to evaluate the noise robustness of audio classification models. Finally, an ablation study is conducted exploring the invariance to nuances of various pooling operations in the context of the proposed evaluation framework.

The paper is organized as follows. Initially, related literature is presented. Next, there is a detailed description of the proposed systems and all aspects of the experimental arrangement. Afterwards the experimental results are demonstrated and analysed. Finally, conclusions and future research directions are presented.

2 Related work

Modern deep learning systems have shown unprecedented performance in many domains outperforming humans. Some non exhaustive areas, where deep learning thrived, are image classification [43, 19], speech recognition [10, 5] and natural language understanding [12, 41]. The performance of neural networks is further boosted by modern hyperparameter optimization methods such as automl [20], quantum genetic algorithms [28] and swarm intelligence [4].

This technological advancement is already transforming the industry spanning many sectors such as energy [37], food [31], automotive [15, 16], medicine [48] and many others. In this light, deep learning has raised the baseline in many sound recognition tasks such as automatic speech recognition (ASR), speech-to-text (STT), speech emotion recognition, voice commands recognition, environmental sound recognition (ESR) and others. The traditional pipeline of such systems includes a preprocessing step, feature extraction and a learning model [55, 7, 51]. Feature extraction is not always computationally efficient [46] and modern approaches in feature preprocessing suggest fast methods for computing local features from image and video frames [1]. Deep neural networks can provide end-to-end solutions without the need for handcrafted feature engineering and outperforming traditional approaches [50, 49, 55, 18].

So far, the majority of machine learning research has been performance oriented, ignoring other aspects like efficiency. As a result, the best performing models consist of millions or billions of parameters requiring a cloud of powerful GPUs to run. Reaching such computational limits and in accordance to the demands of the industry, researchers realized that building efficient neural networks with limited size is essential. A strong example is modern language models like Bert [12], which is now replaced by smaller versions such as MobileBERT [45]. Similar efforts in IoT applications managed to achieve state-of-the-art performance by reducing the size of the layers and replacing them with more efficient ones such as attention mechanism [17, 13]. Other methodologies suggest quantization aware fine tuning [29] for NLP tasks or dimensionality reduction for time series [36]. Compression methods have also been utilized in speech recognition [30, 33] and there is an increasing interest in audio based applications that can run on embedded or mobile devices [44].

Coucke et al. [11] present a neural network with dilated convolution layers which combined gated activations and residual connections. Their contribution is twofold. The proposed neural network fits in embedded devices and the dataset that they created, named "Hey Snips" is public with utterances recorded by over 2.2K speakers. Kusupati et al. [25] propose a novel recurrent neural network (RNN) architecture named FastGRNN, which includes low-rank, sparse and quantized matrices. The developed neural network is up to 35x smaller than other state-of-the-art RNNs. The goal of this study is to develop neural networks that can easily be deployed on IoT devices. Zeng and Xiao [54] propose a model called DenseNet-BiLSTM for the task of keyword spotting (KWS). DenseNet-BiLSTM is evaluated utilizing the Google Speech Commands dataset [53]. Their main contribution is the combination of a new version of DenseNet named DenseNet-Speech and BiLSTM. DenseNet learns local features and at the same time maintains sequential patterns. BiLSTM learns time depended features. Solovyev et al. [44] use different representations of sound such as Wave frames, Spectrograms, Mel-Spectograms and MFCCs and compare different convolutional neural networks. The outcome is that the best performing networks are the ones inspired by VGG [43] and ResNet [19]. The models are evaluated on the Google Speech Commands dataset, achieving accuracy over 90%.

Apart from high accuracy, small size and efficiency, deploying a sound recognition machine learning model on edge devices and in acoustically noisy environments requires it to be robust. Zhang et al. [55] present an overview of models which are resilient to noise and compare previous approaches with deep learning ones. Deep learning outperforms older methods, however most of these models do not meet the rest of the requirements for deployment on computationally constraint machines and in noisy real environments. Phan et al. [40] focus on both efficiency and robustness proposing a shallow convolutional neural network with only three layers: a convolutional, a max pooling and a softmax one. The convolutional layer consists of many filters that, according to the authors, play the role of a cochlear filter. The proposed system incorporates a preprocessing step, converting an audio signal into spectrogram image

features. The system is evaluated using the Real Word Computing Partnership Sound Scene Database in Real Acoustic Environments [35] and outperforms other larger convolutional neural networks. One important outcome from this study is that the number of filters improve the robustness of the model. Another noteworthy outcome is that the robustness is stronger when training occurs with both clean and noisy data. Adding noise and augmenting data has been a good practice to enhance a model's robustness [9], however it is not always feasible to replicate the real-world data distribution. Therefore, it is equally important for the model itself to encapsulate mechanisms that will make it resilient to unpredictable noise. Pervaiz et al. [39] study the performance of machine learning models when they are trained with noisy data. The authors evaluate a Gaussian Mixture Model (GMM) and some variations of convolutional neural networks using the Google Speech Commands Dataset. The main conclusion is that augmenting noise in training data improves the performance of the models. Wang [52] suggests a hierarchical audio content classification approach that leverages the robustness of the system. The solution consists of three components: voice activity detection based on entropy, speech/music discrimination using support vector machine and post-processing which is rule based. The target classes are noise, speech and music. The method is efficient and robust but more experiments are advised to be conducted on more complex classification tasks e.g. with more labels.

The literature review shows that there is plenty room for improvement in sound recognition tasks. This research focuses on audio classification tasks with regards to robustness and efficiency of deep neural networks without degrading performance or increasing the parameters of the model. The models utilize pooling operations for noise invariance and experimental results show the importance of the maximum entropy (maxent) principle when designing neural networks.

3 Proposed Models

3.1 Preprocessing and Audio Features

The deep learning models that are used in the experiments are trained with audio data in the frequency domain. Since one model consists of 2D convolutional layers and the other one of 1D layers, there are two respective preprocessing steps. For the 2D case, we compute the Spectogram of the input, using the algorithm of Short Time Fourier Transform (STFT). The advantage of STFT is that it maintains temporal information because the Fourier transformation applies to segments of the given data and not to the entire time series. Apart from the audio data, STFT also has two parameters the frame length and the stride. The result of the transformation is a complex matrix. Next, the energy spectogram is computed using the magnitude of the complex elements and calculating their logarithm. Finally, the angle of the complex elements is also concatenated in the feature set. Regarding the 1D case, the Fourier transfor-

mation is applied on the given batch of data. The input to the model is the absolute value of the first half of the frequencies.

3.2 Entropy Pooling

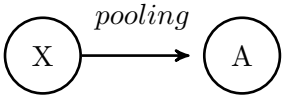
Modern deep learning architectures often include pooling operations, contributing to invariance to data variation and robustness to perplexity. Pooling is popular in models of image recognition, but is also used in speech recognition, natural language processing, signal processing etc. The objective of pooling is to subsample a joint feature representation into a smaller, more compact one that maintains as much information as possible. The most common pooling techniques are max [21] and average [26, 27], whereas it is not unusual for a neural net to include both of them. Despite the success of pooling layers the choice of which one to use is decided through many experimental trials and there is no established theoretical framework.

In the literature there are two main theoretical analyses, trying to fill the gap between theory and practice and understand the dynamics of pooling in a neural network. Boureau et al. [8] study the statistical properties of max and average visual feature pooling in a two-class classification problem. The authors also assume that the features are independent and identically distributed (i.i.d.) Bernoulli random variables. They conclude that the sparsity of the features and the sample cardinality are two important properties that affect the performance of the model. Due to the complexity of the problem, the underlying reasons that analytically justify their performance are obscured.

Nalmpantis et al. [38] study the behaviour of pooling operations from the information theory perspective. The study is based on the maximum entropy principle, also known as maxent or infomax. The principle has been used in previous research showing that it reduces redundancy in nonlinear feed-forward neural nets [34]. Theoretical analysis as well as experimental work show that max pooling is not always compliant with the maxent principle. Average pooling shows more consistent results because its output tends to be closer to the uniform distribution, which is attributed to the computation of means. Both of these operations cannot guarantee high entropy and depend on the distribution of the input. In order to understand better the dynamics of pooling, Nalmpantis et al. [38] present a novel pooling operation, called entropy pooling. Entropy pooling is guaranteed to select a feature set with high entropy, regardless of the input distribution of the data. Below, a rigorous description of the operation is described.

Assume a deep neural network with a pooling operation after hidden layer i. Let the output of the hidden layer i be a random variable X, which will be the input of the pool. Let A be the random variable of its output. Then pooling operation can be seen as an information channel. The Markov chain that describes it is depicted in Fig. 3.2. The mutual information I(X,A) and the capacity C of the channel, are expressed by the following equations accordingly:

$$I(X;A) = H(A) - H(A|X) \tag{1}$$



$$C = \max_{p(x)} I(X;A) \tag{2}$$

with H(A) the entropy and H(A|X) the conditional entropy. It is obvious that I(X;A) is max when H(A) is maximized and H(A|X) is minimized. The conditional entropy H(A|X) can be equal to zero when pooling is a deterministic function and (1) becomes:

$$I(X;A) = H(A) \tag{3}$$

Entropy pooling finds a maximum of the channel capacity. This maximum is not global because the optimal solution is NP hard [42]. It computes the probabilities p of N given features. Next, the probabilities are filtered by selecting the least frequent features, giving an output with high entropy. The process is illustrated in Fig. 1. The mathematical formula for a region of size r, is:

$$f_{entr}(X_r) = X_r[g(P_r)], \tag{4}$$

$$g(P_r) = \underset{1 \leq i \leq r}{\arg\min}\, p_i \tag{5}$$

, where X_r is the input feature map, g returns the indices of the smallest probability and P_r the constructed map of probabilities. Consequently:

$$I(X;A) = H(f_{entr}(X_r)) \tag{6}$$

3.3 Configuration of Neural Networks

This research is inspired by two popular neural network architectures that demonstrated state-of-the-art results in computer vision named AlexNet [24] and ResNet [19]. Variations of AlexNet, ResNet and others like InceptionV3, Xception and VGG have been employed in speech commands recognition by prior research [44]. Despite the high performance of these models, there is still room for improvement in terms of both efficiency and robustness.

Efficiency is achieved by reducing the parameters without large performance sacrifices. A key component for better efficiency is the introduction of batch normalization layers in between of the main ones. Robustness is leveraged via the pooling layers. The type of these layers has been found through a systematic experimentation and evaluation of combinations of max, average and entropy functions.

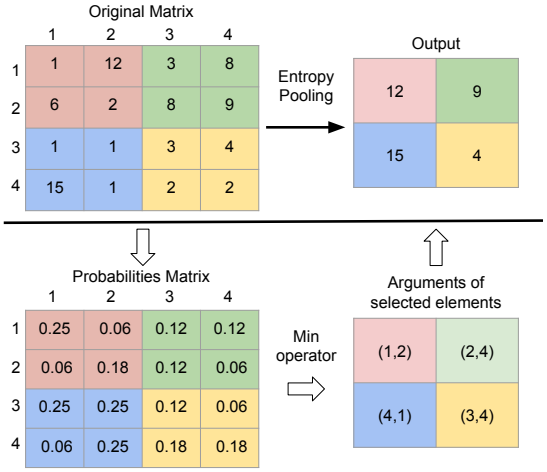


Fig. 1 The process of entropy pooling. Min operator is selecting the most rare features.

The first model is a convolutional neural network with six 2D convolutional layers and activation function ReLU. After each of the first four convolutional layers there is a batch normalization layer and a pooling one. The architecture is shown in 1. Pooling layers are not specified yet. As it will be explained later on, different configurations are found to work better in the four different environmental setups of the experiments. This architecture is evaluated on the speech commands dataset.

The second model is a residual neural network. The residual block consists of a convolutional layer and a parameterized number of convolutional layers in parallel to the first one. The output of the residual convolutions is concatenated, pass through a ReLU activation function and a pooling layer. The entire architecture includes five residual blocks, followed by a pooling operation and three dense layers. The details of the architecture are shown in 2. The two pooling layers are combinations of entropy, max and average ones and the best configuration is found to be different depending on the robustness scenario. The residual architecture is evaluated on the speaker recognition dataset.

4 Materials, Methods and Experimental Results

4.1 Datasets and Setup of Environments

Two datasets are chosen for the evaluation of the developed models. The first one is the "Speaker Recognition Dataset Prominent Leaders Speeches" which can be downloaded from kaggle using the following link `https://www.kaggle.com/kongaevans/speaker-recognition-dataset`. It includes speeches of the popular leaders: Benjamin Netanyahu, Jens Stoltenberg, Julia Gillard, Mar-

Table 1 Architecture of 2D convolutional neural network.

2DConvNN	Output shape
Input	(batch size, 122, 257, 2)
BatchNormalization	(batch size, 122, 257, 2)
Conv2D 32 filters	(batch size, 122, 257, 32)
BatchNormalization	(batch size, 122, 257, 32)
Pooling Layer	(batch size, 61, 128, 32)
Conv2D 32 filters	(batch size, 59, 126, 32)
BatchNormalization	(batch size, 59, 126, 32)
Pooling Layer	(batch size, 29, 63, 32)
Conv2D 128 filters	(batch size, 27, 61, 128)
BatchNormalization	(batch size, 27, 61, 128)
Pooling Layer	(batch size, 13, 30, 128)
Conv2D 256 filters	(batch size, 11, 28, 256)
BatchNormalization	(batch size, 11, 28, 256)
Pooling Layer	(batch size, 5, 14, 256)
Conv2D 128 filters	(batch size, 5, 14, 128)
Conv2D 64 filters	(batch size, 5, 14, 64)
Flatten	(batch size, 4480)
Dense	(batch size, 12)
BatchNormalization	(batch size, 12)

Table 2 Architecture of the residual neural network.

Residual Block m= # of convolutions	Residual Network f= # of filters
Conv1D	Residual Block (m=16, f=2)
m x Conv1D	Residual Block (m=32, f=2)
Conv1D	Residual Block (m=64, f=3)
ReLU	Residual Block (m=128, f=3)
Pooling Layer	Residual Block (m=128, f=3)
	Pooling Layer
	Flatten
	3 x Dense

garet Thatcher and Nelson Mandela. The length of each audio is one second with sampling rate 16kHz and PCM encoded. The dataset also contains a folder with audio files representing background noise like laughing, clapping etc. The goal is to recognize the speaker taking into consideration background noise. The second dataset is the Speech Commands dataset [53], which has been a standard one for the task of speech commands classification targeting devices with limited computational resources. It includes 60K audio files with length around 1 second. The audio files are PCM encoded with sampling rate 16kHz. There are 32 different labels, from which only 10 are the target ones. The rest of the labels are considered as silence or unknown. The target labels are left, right, up, down, yes, no, go, stop, on, off. Figure 2 depicts a pie chart with the proportion of each target command in the training set.

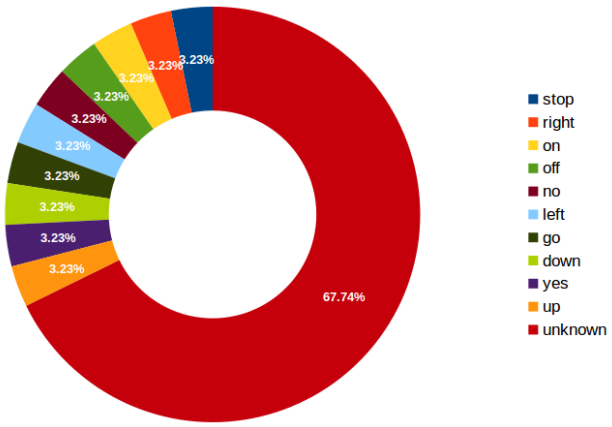


Fig. 2 Proportions of target commands in Speech Commands training dataset.

Table 3 Environment setup based on the presence of noise. The four scenarios are also represented by the abbreviation of three letters. E.g. FFT stands for train and evaluation without noise and test with noise.

Scenario	Training	Evaluation	Testing
Naive approach (FFF)	False	False	False
Known noise (TTT)	True	True	True
Noise augmentation (TTF)	True	True	False
Out of distribution (FFT)	False	False	True

The current research focuses on training and evaluating audio classification models in terms of performance, efficiency and robustness. In order to make a thorough comparative analysis a new evaluation framework is proposed with respect to the environmental setup. Four scenarios are suggested with different training, evaluation and testing environments considering whether noise should be included or not. The first one is the naive approach where there is no noise. Train, evaluation and test data are clean and are assumed to come from the same distribution. The second scenario regards noise that can be predicted, which means that we can add noise during training and evaluation. For example it would be expected to hear people laughing in an office. The test environment is supposed to have similar noise. The third case is when data are augmented with noise and then testing includes clean data. This is a rare scenario in the real world, but we include it for completeness. The last and maybe closest to real conditions scenario is when we train our data and the model has to make robust predictions even when there is unpredictable noise. In this case noise is anything that is out of distribution or not included in the target classes. Table 3 summarizes the four different scenarios with the following names respectively: "naive approach (FFF)", "known noise (TTT)", "noise augmentation (TTF)" and "out of distribution (FFT)". Because of space limitations most of the tables in this document refer to the four scenarios

Table 4 Experimental results showing the accuracy of the residual neural network with different pooling operations. The experiments cover the four different scenarios: "naive approach (FFF)", "known noise (TTT)", "noise augmentation (TTF)" and "out of distribution (FFT)".

Pooling config.	FFF	TTT	FFT	TTF
AVG - AVG	0.988	0.967	0.683	0.961
ENTR - AVG	0.984	0.949	0.644	0.961
MAX - AVG	0.993	0.955	0.640	0.971
AVG - ENTR	**0.996**	0.966	**0.685**	0.970
ENTR - ENTR	0.983	0.956	0.659	0.975
MAX - ENTR	0.987	0.943	0.655	**0.977**
AVG - MAX	0.992	**0.968**	0.681	0.976
ENTR - MAX	0.989	0.962	0.656	0.955
MAX - MAX	0.992	0.966	0.654	0.976

with their abbreviations. The abbreviations refer to whether there is noise in the training, evaluation or testing environment with F for false and T for true. Thus, FFT means that noise is included only in the test data.

4.2 Experimental Results and Discussion

The neural network based on the residual architecture is evaluated on the speaker recognition task using the dataset "Speaker Recognition Dataset Prominent Leaders Speeches". The model is configured through a systematic experimental study evaluating all the possible combinations of max, average and entropy pooling layers. The process is repeated for all the four scenarios that were described previously. In order to make the comparison of different variations of the model fair, each one is trained, evaluated and tested several times. Next, the mean and standard deviation of accuracy results are calculated and compared to find the most robust model.

Table 4 shows the results of all the different combinations of the three pooling operations for the residual neural network. In this table the name of the pooling configuration has two parts. The first part refers to the pooling of the residual block and the second one to the last pooling of the entire network. For example if the residual block has the max pooling layer and the last pooling of the network is the entropy one the pooling configuration is referred as MAX-ENTR. The most robust versions of the architecture are AVG-ENTR, AVG-MAX, AVG-ENTR and MAX-ENTR for the scenarios "naive approach", "known noise", "out of distribution" and "noise augmentation" respectively.

In order to shed light on the impact of the different pooling types on the robustness of the model, the neural networks with the configurations ENTR-ENTR, MAX-MAX and AVG-AVG are separately evaluated in terms of the cross entropy error. Figure 3 presents the experimental results in the four different environmental setups. The scenario "out of distribution" is scaled down 10 times in order to fit with the bars of the other three scenarios. The conclu-

sion is that max pooling helps the neural network to achieve high performance under known conditions, especially in the "naive approach". Average pooling shows similar results, however it is easily biased by noise if it is present in the training data. In the case where noise is present only in the test data, average pooling is robust. Finally, entropy pooling has the lowest error in the scenario "noise augmentation" and performs on par with average pooling in the scenario "out of distribution". Entropy pooling is robust when the distribution of testing data is different from the distribution of training data. An example of the validation error during training of the three versions of the model is shown at Fig. 4. According to the diagram the three versions of the model all converge with the one that uses entropy pooling achieving the lowest error. The configuration with average pooling follows and the worse performance is from max pooling.

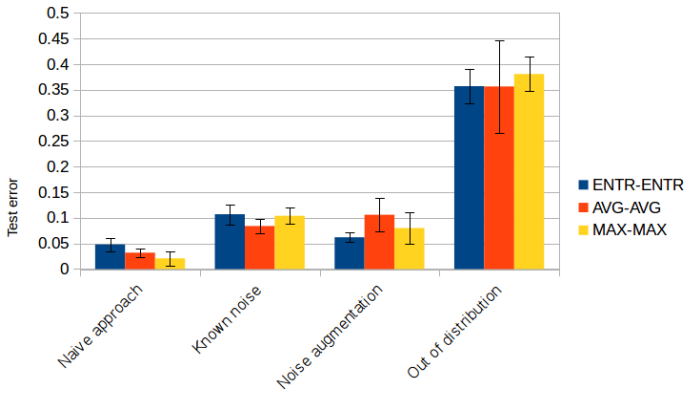


Fig. 3 Evaluation of the residual neural network using one type of pooling each time. The out of distribution testing results are scaled down 10 times to fit the diagram with the rest of the results.

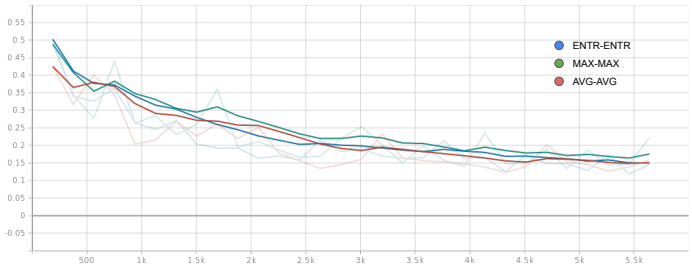


Fig. 4 The diagram depicts the cross entropy error of the residual neural network during training. The three versions of the network correspond to the pooling types max, average and entropy. All three versions converge for the speaker recognition task. Average and entropy pooling show similar performance while max one has higher error.

Table 5 Experimental results of residual neural network with different pooling operations when testing dataset is out of distribution and in the presence of different scaling factors of the magnitude of the noise.

Pooling config.	x0.25	x0.5	x0.75	x1	x2
AVG - AVG	0.808	0.683	0.558	0.491	0.368
ENTR - AVG	0.810	0.644	0.558	0.491	0.369
MAX - AVG	**0.819**	0.640	0.527	0.488	0.377
AVG - ENTR	0.794	**0.685**	0.579	0.506	**0.404**
ENTR - ENTR	0.793	0.659	0.539	0.484	0.389
MAX - ENTR	0.790	0.655	0.557	0.508	0.372
AVG - MAX	0.802	0.681	0.578	0.510	0.375
ENTR - MAX	0.764	0.656	0.549	**0.512**	**0.404**
MAX - MAX	0.810	0.654	**0.589**	0.492	0.391

Table 6 Results showing the accuracy of the 2D convolutional network with different pooling operations. M stands for max pooling and E for entropy pooling. The first column covers the naive approach and the rest include noise in testing with a scaling factor.

Pool conf.	FFF	x0.25	x0.5	x1	x2
MMMM	**0.932**	0.856	0.782	0.706	0.620
MMEM	0.931	0.875	**0.811**	**0.720**	0.637
MMME	0.926	0.861	0.805	0.707	0.611
EMMM	0.923	**0.876**	0.805	0.677	0.512
MEMM	0.923	0.870	0.810	0.713	0.605
EMEM	0.923	0.836	0.775	0.686	0.556
MEEM	0.917	0.860	0.795	0.717	0.641
EMME	0.916	0.854	0.787	0.712	**0.644**
EEMM	0.916	0.826	0.748	0.645	0.540
EEME	0.915	0.830	0.755	0.624	0.581
EEEM	0.914	0.846	0.785	0.702	0.619
MEME	0.909	0.833	0.746	0.602	0.437
MMEE	0.907	0.843	0.773	0.696	0.634
EEEE	0.901	0.792	0.724	0.668	0.638
EMEE	0.898	0.841	0.776	0.708	0.628
MEEE	0.888	0.816	0.734	0.668	0.616

Among the four scenarios, "out of distribution" is the most common in the real world because a real environment is unpredictable and any kind of sound can happen spanning music, traffic, appliances, animals and others. More experiments are conducted taking into account different scaling factors of the amplitude of noises. Table 5 presents the results from five different cases of noise with scaling factors x0.25, x0.5, x0.75, x1 and x2. The versions of the network AVG-ENTR and ENTR-MAX seem the most robust ones and should be preferred especially when we cannot predict or control the intensity of noise in the real environment.

The 2D convolutional architecture is evaluated in the Google Speech Commands Dataset. This model includes four pooling layers. The experiments include all the variations of the model by combining max and entropy pooling

Table 7 Results showing the cross-entropy error of the 2D convolutional network with different pooling operations. M stands for max pooling and E for entropy pooling. The first column covers the naive approach and the rest include noise in testing with a scaling factor.

Pool conf.	FFF	x0.25	x0.5	x1	x2
MMMM	**0.289**	0.553	0.797	1.102	1.448
MMEM	0.321	0.587	0.791	1.088	1.380
MMME	0.328	0.583	0.797	1.151	1.513
EMMM	0.325	0.610	0.882	1.260	1.689
MEMM	0.341	0.580	0.786	1.097	1.441
EMEM	0.325	0.733	0.924	1.220	1.595
MEEM	0.362	0.563	0.764	**1.040**	**1.344**
EMME	0.346	**0.539**	**0.746**	1.058	1.406
EEMM	0.362	0.728	0.967	1.295	1.605
EEME	0.374	0.625	0.858	1.646	1.831
EEEM	0.365	0.617	0.804	1.089	1.408
MEME	0.411	0.708	0.967	1.367	1.800
MMEE	0.408	0.620	0.829	1.110	1.405
EEEE	0.405	0.757	0.973	1.224	1.434
EMEE	0.419	0.620	0.808	1.069	1.371
MEEE	0.456	0.663	0.924	1.258	1.582

layers. For brevity we refer to the architectures with four letters depending on the type of the pooling layers. For example MEME would be the architecture where the first to the last pooling types are max, entropy, max and entropy. The metrics that are used are the accuracy and the cross entropy error. The experimental environments are the naive approach and the out of distribution scenario. Table 6 presents the results with the accuracy of each model. The first column represents the naive approach where there is no noise. The two best models are MMMM and MMEM, with the first one performing slightly better. The other four columns show the results when there is noise in the testing environment with various scaling factors. For the case where the amplitude of noise is scaled by 0.25 the most robust model is EMMM. MEMM and MMEM show very similar performance. For the cases with scaling factors 0.5 and 1, the best model is MMEM followed by MEMM in both cases. In the last case with scaling factor 2, the best performance is shown by EMME, followed by MEEM, EEEE and MMEM. Overall MMEM looks the most promising one when the model has to be deployed in an environment with unexpected noises. Now, considering the metric of cross entropy error, the results are slightly different. As shown in table 7 the models MEEM and EMME show the lowest error. Comparing MMEM which shows the best overall accuracy and MEEM or EMME which show the lowest overall error, it is concluded that MMEM shows relatively low error as well. MEEM and EMME also show descent accuracy, but MEEM seems to outperform EMME in most cases. Thus, the two most robust models with respect to both metrics are MMEM and MEEM.

5 Conclusion

This work evaluates two deep neural networks evaluated in two different audio classification tasks. The architectures of the networks are built with efficiency in mind, having as less parameters as possible without compromising a lot of performance. This indicates that existing deep neural networks are far from an optimal architecture. Therefore, there is a need to discover formal mathematical methods for designing neural networks. One research direction is to design novel neural layers based on fundamental principles such as the entropy pooling.

Furthermore, the impact of pooling layers on the robustness of the models is examined. A systematic evaluation process is followed taking into consideration different environmental conditions of noise. The main outcome is that entropy pooling seems promising in making a deep neural network robust to noise. The combination of pooling layers is also shown to demonstrate strong performance. Further experiments are advised to be conducted by exploring the impact of pooling on the performance of other neural architectures as well as on other datasets. Entropy pooling could also be improved. One possible improvement of entropy pooling is to find a more optimal solution. However, this is considered an NP-hard problem and better solutions trade off a lot of computational complexity [23].

Deep learning is a research domain that grows very fast and there are several neural layers that could be evaluated using noisy audio data. For future research it is recommended to explore the robustness of audio classifiers using other types of layers such as dropout and compare it with modern approaches like variational dropout [22] or information dropout [2]. Other variational approaches, such as deep variational information bottleneck [3], are shown to be resilient to adversarial attacks and should enhance the performance of neural networks under distribution shifts.

References

1. Abdulhussain SH, Rahman Ramli A, Mahmmod BM, Iqbal Saripan M, Al-Haddad S, Baker T, Flayyih WN, Jassim WA (2019) A fast feature extraction algorithm for image and video processing. In: 2019 International Joint Conference on Neural Networks (IJCNN), pp 1–8, DOI 10.1109/IJCNN.2019.8851750
2. Achille A, Soatto S (2018) Information dropout: Learning optimal representations through noisy computation. IEEE Transactions on Pattern Analysis and Machine Intelligence
3. Alemi A, Fischer I, Dillon J, Murphy K (2017) Deep variational information bottleneck. In: ICLR, URL https://arxiv.org/abs/1612.00410
4. Bacanin N, Bezdan T, Venkatachalam K, Al-Turjman F (2021) Optimized convolutional neural network by firefly algorithm for magnetic resonance

image classification of glioma brain tumor grade. Journal of Real-Time Image Processing pp 1–14
5. Bahdanau D, Chorowski J, Serdyuk D, Brakel P, Bengio Y (2016) End-to-end attention-based large vocabulary speech recognition. In: 2016 IEEE international conference on acoustics, speech and signal processing (ICASSP), IEEE, pp 4945–4949
6. Bharitkar S (2019) Generative feature models and robustness analysis for multimedia content classification. In: 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), pp 105–110, DOI 10.1109/ICMLA.2019.00025
7. Bountourakis V, Vrysis L, Konstantoudakis K, Vryzas N (2019) An enhanced temporal feature integration method for environmental sound recognition. In: Acoustics, Multidisciplinary Digital Publishing Institute, vol 1, pp 410–422
8. Boureau YL, Ponce J, LeCun Y (2010) A theoretical analysis of feature pooling in visual recognition. In: Proceedings of the 27th international conference on machine learning (ICML-10), pp 111–118
9. Chen S, Dobriban E, Lee J (2020) A group-theoretic framework for data augmentation. Advances in Neural Information Processing Systems 33
10. Chorowski JK, Bahdanau D, Serdyuk D, Cho K, Bengio Y (2015) Attention-based models for speech recognition. Advances in neural information processing systems 28:577–585
11. Coucke A, Chlieh M, Gisselbrecht T, Leroy D, Poumeyrol M, Lavril T (2019) Efficient keyword spotting using dilated convolutions and gating. In: ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp 6351–6355
12. Devlin J, Chang MW, Lee K, Toutanova K (2019) BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, pp 4171–4186, DOI 10.18653/v1/N19-1423, URL `https://www.aclweb.org/anthology/N19-1423`
13. Dong B, Lumezanu C, Chen Y, Song D, Mizoguchi T, Chen H, Khan L (2020) At the speed of sound: Efficient audio scene classification. In: Proceedings of the 2020 International Conference on Multimedia Retrieval, pp 301–305
14. Esmaeilpour M, Cardinal P, Lameiras Koerich A (2020) A robust approach for securing audio classification against adversarial attacks. IEEE Transactions on Information Forensics and Security 15:2147–2159, DOI 10.1109/TIFS.2019.2956591
15. Falcini F, Lami G (2017) Deep learning in automotive: Challenges and opportunities. In: International Conference on Software Process Improvement and Capability Determination, Springer, pp 279–288
16. Fayyad J, Jaradat MA, Gruyer D, Najjaran H (2020) Deep learning sensor fusion for autonomous vehicle perception and localization: A review.

Sensors 20(15):4220
17. Gkalinikis NV, Nalmpantis C, Vrakas D (2020) Attention in recurrent neural networks for energy disaggregation. In: International Conference on Discovery Science, Springer, pp 551–565
18. Han W, Zhang Z, Zhang Y, Yu J, Chiu CC, Qin J, Gulati A, Pang R, Wu Y (2020) Contextnet: Improving convolutional neural networks for automatic speech recognition with global context. arXiv preprint arXiv:200503191
19. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
20. He X, Zhao K, Chu X (2021) Automl: A survey of the state-of-the-art. Knowledge-Based Systems 212:106622
21. Jarrett K, Kavukcuoglu K, LeCun Y, et al. (2009) What is the best multi-stage architecture for object recognition? In: Computer Vision, 2009 IEEE 12th International Conference on, IEEE, pp 2146–2153
22. Kingma DP, Salimans T, Welling M (2015) Variational dropout and the local reparameterization trick. In: Advances in neural information processing systems, pp 2575–2583
23. Ko CW, Lee J, Queyranne M (1995) An exact algorithm for maximum entropy sampling. Operations Research 43(4):684–691
24. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: NIPS
25. Kusupati A, Singh M, Bhatia K, Kumar A, Jain P, Varma M (2018) Fastgrnn: A fast, accurate, stable and tiny kilobyte sized gated recurrent neural network. In: Advances in Neural Information Processing Systems, pp 9017–9028
26. LeCun Y, Boser BE, Denker JS, Henderson D, Howard RE, Hubbard WE, Jackel LD (1990) Handwritten digit recognition with a back-propagation network. In: Advances in neural information processing systems, pp 396–404
27. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proceedings of the IEEE 86(11):2278–2324
28. Lentzas A, Nalmpantis C, Vrakas D (2019) Hyperparameter tuning using quantum genetic algorithms. In: 2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI), IEEE, pp 1412–1416
29. Li B, Huang K, Chen S, Xiong D, Jiang H, Claesen L (2020) Dfqf: Data free quantization-aware fine-tuning. In: Asian Conference on Machine Learning, PMLR, pp 289–304
30. Li S, Raj D, Lu X, Shen P, Kawahara T, Kawai H (2019) Improving transformer-based speech recognition systems with compressed structure and speech attributes augmentation. In: INTERSPEECH, pp 4400–4404
31. Makridis G, Mavrepis P, Kyriazis D, Polychronou I, Kaloudis S (2020) Enhanced food safety through deep learning for food recalls prediction. In: International Conference on Discovery Science, Springer, pp 566–580

32. Martin-Morato I, Cobos M, Ferri FJ (2018) On the robustness of deep features for audio event classification in adverse environments. In: 2018 14th IEEE International Conference on Signal Processing (ICSP), pp 562–566, DOI 10.1109/ICSP.2018.8652438
33. McGraw I, Prabhavalkar R, Alvarez R, Arenas MG, Rao K, Rybach D, Alsharif O, Sak H, Gruenstein A, Beaufays F, et al. (2016) Personalized speech recognition on mobile devices. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, pp 5955–5959
34. Nadal JP, Brunel N, Parga N (1998) Nonlinear feedforward networks with stochastic outputs: infomax implies redundancy reduction. Network: Computation in neural systems 9(2):207–217
35. Nakamura S, Hiyane K, Asano F, Yamada T, Endo T (1999) Data collection in real acoustical environments for sound scene understanding and hands-free speech recognition. In: Sixth European Conference on Speech Communication and Technology
36. Nalmpantis C, Vrakas D (2019) Signal2vec: Time series embedding representation. In: International Conference on Engineering Applications of Neural Networks, Springer, pp 80–90
37. Nalmpantis C, Vrakas D (2020) On time series representations for multi-label nilm. NEURAL COMPUTING & APPLICATIONS
38. Nalmpantis C, Lentzas A, Vrakas D (2019) A theoretical analysis of pooling operation using information theory. In: 2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI), IEEE, pp 1729–1733
39. Pervaiz A, Hussain F, Israr H, Tahir MA, Raja FR, Baloch NK, Ishmanov F, Zikria YB (2020) Incorporating noise robustness in speech command recognition by noise augmentation of training data. Sensors 20(8):2326
40. Phan H, Hertel L, Maass M, Mertins A (2016) Robust audio event recognition with 1-max pooling convolutional neural networks. In: Proceedings of Interspeech, ISCA, pp 3653–3657
41. Qu Y, Liu P, Song W, Liu L, Cheng M (2020) A text generation and prediction system: Pre-training on new corpora using bert and gpt-2. In: 2020 IEEE 10th International Conference on Electronics Information and Emergency Communication (ICEIEC), pp 323–326, DOI 10.1109/ICEIEC49280.2020.9152352
42. Shewry MC, Wynn HP (1987) Maximum entropy sampling. Journal of applied statistics 14(2):165–170
43. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:14091556
44. Solovyev RA, Vakhrushev M, Radionov A, Romanova II, Amerikanov AA, Aliev V, Shvets AA (2020) Deep learning approaches for understanding simple speech commands. In: 2020 IEEE 40th International Conference on Electronics and Nanotechnology (ELNANO), IEEE, pp 688–693
45. Sun Z, Yu H, Song X, Liu R, Yang Y, Zhou D (2020) MobileBERT: a compact task-agnostic BERT for resource-limited devices. In: Proceedings of

the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, pp 2158–2170, DOI 10.18653/v1/2020.acl-main.195
46. Tippaya S, Sitjongsataporn S, Tan T, Khan MM, Chamnongthai K (2017) Multi-modal visual features-based video shot boundary detection. IEEE Access 5:12563–12575
47. Turchet L, Fazekas G, Lagrange M, Ghadikolaei HS, Fischione C (2020) The internet of audio things: state-of-the-art, vision, and challenges. IEEE Internet of Things Journal
48. Viswanathan J, Saranya N, Inbamani A (2021) Deep learning applications in medical imaging: Introduction to deep learning-based intelligent systems for medical applications. In: Deep Learning Applications in Medical Imaging, IGI Global, pp 156–177
49. Vrysis L, Thoidis I, Dimoulas C, Papanikolaou G (2020) Experimenting with 1d cnn architectures for generic audio classification. In: Audio Engineering Society Convention 148, Audio Engineering Society
50. Vrysis L, Tsipas N, Thoidis I, Dimoulas C (2020) 1d/2d deep cnns vs. temporal feature integration for general audio classification. Journal of the Audio Engineering Society 68(1/2):66–77
51. Vrysis L, Tsipas N, Thoidis I, Dimoulas C (2020) Enhanced temporal feature integration in audio semantics. Journal of the Audio Engineering Society 68(1/2):66–77
52. Wang KC (2020) Robust audio content classification using hybrid-based smd and entropy-based vad. Entropy 22(2):183
53. Warden P (2018) Speech commands: A dataset for limited-vocabulary speech recognition. arXiv preprint arXiv:180403209
54. Zeng M, Xiao N (2019) Effective combination of densenet and bilstm for keyword spotting. IEEE Access 7:10767–10775
55. Zhang Z, Geiger J, Pohjalainen J, Mousa AED, Jin W, Schuller B (2018) Deep learning for environmentally robust speech recognition: An overview of recent developments. ACM Trans Intell Syst Technol 9(5), DOI 10.1145/3178115, URL https://doi.org/10.1145/3178115