Applied Intelligence (2023) 53:20561–20576
https://doi.org/10.1007/s10489-023-04571-y

Full single-type deep learning models with multihead attention for speech enhancement

Noel Zacarias-Morales[1] · **José Adán Hernández-Nolasco**[1] · **Pablo Pancardo**[1]

Accepted: 11 March 2023 / Published online: 15 April 2023

Abstract

Artificial neural network (ANN) models with attention mechanisms for eliminating noise in audio signals, called speech enhancement models, have proven effective. However, their architectures become complex, deep, and demanding in terms of computational resources when trying to achieve higher levels of efficiency. Given this situation, we selected and evaluated simple and less resource-demanding models and utilized the same training parameters and performance metrics to conduct a fair comparison among the four selected models. Our purpose was to demonstrate that simple neural network models with multihead attention are efficient when implemented on computational devices with conventional resources since they provide results that are competitive with those of hybrid, complex and resource-demanding models. We experimentally evaluated the efficiency of multilayer perceptron (MLP), one-dimensional and two-dimensional convolutional neural network (CNN), and gated recurrent unit (GRU) deep learning models with and without multiheaded attention. We also analyzed the generalization capability of each model. The results showed that although these architectures were composed of only one type of ANN, multihead attention increased the efficiency of the speech enhancement process, yielding results that were competitive with those of complex models. Therefore, this study is helpful as a reference for building simple and efficient single-type ANN models with attention.

Keywords Artificial neural network · Attention · Deep learning models · Speech enhancement

1 Introduction

Attention involves selectively focusing on specific elements while ignoring less relevant or important elements according to a goal. The notion of selective attention in humans is related to senses such as sight, hearing and smell. Auditory attention is a selection process or set of processes that focuses sensory and cognitive resources on the most relevant events in the sound landscape [5]. Therefore, attention has allowed humans to focus their interest on only a fraction of the available information, making optimal use of resources to better achieve their goals [8].

Attention is a mechanism that is being increasingly used in a wide range of deep learning architectures. The mechanism itself is available in various formats [3]. The use of attention mechanisms has become more relevant due to the publication of the proposed transformer architecture based mainly on an attention mechanism, operating even with recurrence and convolutions [26]. The authors highlighted its high degree of parallelization, thus increasing its computational speed.

An increasing number of architectures are using attention mechanisms and are being applied to natural language problems [3] and different types of data, especially sequential data. Examples include the improvement of biomedical image segmentation [24], speech recognition [2, 33], and speech enhancement [11, 32].

Among all the different attention mechanisms in the literature [1, 15], we selected multihead attention because it is a mechanism that can be efficiently parallelized and

José Adán Hernández-Nolasco and Pablo Pancardo are contributed equally to this work.

✉ José Adán Hernández-Nolasco
adan.hernandez@ujat.mx

Noel Zacarias-Morales
201h18002@alumno.ujat.mx

Pablo Pancardo
pablo.pancardo@ujat.mx

[1] Academic Division of Sciences and Information Technology, Juarez Autonomous University of Tabasco, Cunduacan, 86690, Tabasco, Mexico

implemented in many models. Furthermore, its implementations with different types of neural networks are less complex and allow us to keep the general architecture of the attention module; also, the number of trainable parameters added is not excessive.

Work in the literature have been conducted on neural networks incorporating attention into speech enhancement in a single channel with a single type of noise at various signal-to-noise ratios (SNRs). We did not find works in the literature that investigated different models consisting entirely of single types of neural network incorporating multihead attention mechanisms for speech enhancement. Furthermore, model performance needs to be assessed under fair conditions during training and evaluation (same input data, training settings, and performance metrics) to determine the impact of multihead attention on performance. Similarly, the generalization capabilities of models must be evaluated by feeding them with new data. The generalization capability of a model is its ability to obtain efficient results when we introduce new data from a new dataset. For example, data from a dataset other than the one used for training may include new features that are not contained in the training dataset [28].

To address this lack of knowledge, we used performance metrics to show the degrees of improvement exhibited by models when incorporating an attention module into different types of specific neural networks, as well as their generalization capabilities. Then, we compared our with against those provided by other authors using complex architectures

Artificial neural networks (ANNs) can be categorized into three common types of networks with their own mathematically defined internal operations, and they serve as the basis for various architectures. Multilayer perceptrons (MLPs) are the most basic neural networks, consisting of three or more fully connected layers with an activation function at each node. Convolutional networks incorporate one or more layers with a convolution process, which consists of multiplying a kernel (weight matrix) by the input data matrix, which can possess one or more dimensions. Recurrent networks are networks in which the connections between nodes can form a cycle, allowing the outputs of some nodes to affect the nodes in the same or subsequent layers.

The neural network models we selected were a MLP, one-dimensional and two-dimensional convolutional neural networks (CNNs), and a gated recurrent unit (GRU) because these are all conventional and commonly used models. Furthermore, all models were full neural networks with simple (not complex) architectures containing less than two million hyperparameters.

The experimental evaluation carried out in this work is essential for determining the performance of each type of neural network and the improvement that each type presents when incorporating an attention mechanism, as well as for analyzing the generalization capacity of each model. The results of this study are helpful as a reference for the construction of solutions in which speech enhancement models are not complex but simultaneously efficient, representing an opportunity when deploying our proposal in computational devices with limited resources.

Our research questions were as follows: (i) How much do neural network models (MLP, 1D-CNN, 2D-CNN, and GRU) improve by incorporating a multihead attention mechanism to solve the speech enhancement problem? (ii) What are the generalization capabilities of the MLP, 1D-CNN, 2D-CNN, and GRU models with a multihead attention mechanism?

We organized the rest of this paper as follows. First, Sections 2 and 3 explain characteristics, theory and similar proposals in the literature. Then, Section 4 presents the details of the four constructed models and the implemented attention module. Section 5 explains the datasets used and the experimental setups. Next, Section 6 presents the results and a discussion of the experiments performed. Finally, Section 7 provides the conclusion of this article.

2 Background

For speech enhancement, a fundamental challenge involves selectively listening to a single audio signal since the sounds that we commonly perceive result from mixing acoustic signals. Extracting features from a single sound source is especially difficult in single-channel recordings. Speech enhancement is the process of removing or attenuating the added noise in a speech signal. It is generally concerned with improving the intelligibility and quality of speech that suffers from degradation due to the inclusion of noise. Speech enhancement acts as a preprocessing technique in applications such as automatic speech recognition. Single-channel speech enhancement aims to solve the problem regarding the use of recordings made with a single microphone. However, single-channel speech enhancement is considered a challenging problem since there are no directional clues for determining the origins of the various audio signals that compose the presented noises.

In the real world, speech signals are easily corrupted by noise. Noises generally belong to stationary noises (which do not change over time) and nonstationary noises (which change over time). Some noises that belong to the nonstationary category are street noises, the noise of a train, babbling noises (other people's voices), and the sounds of musical instruments. Some noises that belong to the stationary category come from air conditioners, fans, compressors, or impeller pumps.

Audio signals are usually artificial when training neural network models since adding speech with noise signals is necessary. To emulate natural noise environments in speech signals, it is necessary to collect noise signals from databases. These databases contain environmental sounds from different sources and different areas, where the aggregated noise is the sum of sounds with variable shapes and magnitudes. The use of environmental sounds impacts the complexity of the speech enhancement problem since it is difficult to establish a noise signal pattern. Researchers typically use SNR values ranging from + 10 dB to − 10 dB, where an audio signal at + 10 dB is equivalent to a school classroom (the speech signal is greater than the noise signal) and -10 dB is equivalent to a train station (the noise signal is greater than the speech signal) [13]. In more negative SNR ranges, the values of the resulting metrics will not be high because the speech signals are more corrupt than expected.

We can define the relationship between speech and noise signals in the time domain as (1).

$$y(t) = x(t) + n(t), \tag{1}$$

where *x(t)* is a clean speech signal, and *n(t)* is the added noise, resulting in *y(t)* being a noisy speech signal. Let *t* be the time index; we can represent the signal as *y = [y(1), ..., y(T)]*, where *t* is the length of the audio fragment.

We based our experiments on magnitude spectrogram mapping. In the mapping-based method, the training objective of the model is to map a nonlinear function *F* from the noisy speech signal *y(t)* to an enhanced clean speech signal *x(t)*, as shown in (2):

$$y(t) \rightarrow^{F} x(t). \tag{2}$$

Because problems associated with fast variation occur when using raw speech signals (in the time domain), researchers usually apply the mapping-based method to the magnitude spectrogram of the speech signals (frequency domain). First, the short-time Fourier transform (STFT) creates a magnitude spectrogram by using the time windowing responses of a filter bank. Subsequently, the inverse operation of the STFT reconstructs the spectrogram back to the signal in the time domain by using the phase information of the original speech signal with noise [30].

Based on the mapping method, neural networks learn to reconstruct the output data from the input data during training. The output data come from the clean speech signal *x(t)*, while the input data come from the speech signal mixed with noise *y(t)*. The neural network model learns a function *F* by minimizing the mean squared error (MSE) loss between the input spectrogram and its reconstructed input, as in (3):

$$L_{MSE} = \|Y - F(X)\|^2. \tag{3}$$

The fusion of models based on deep learning and attention mechanisms has helped the resulting models to emphasize the most informative features and suppress the less useful features. The attention mechanisms used in deep learning originated as improvements of the encoder-decoder architecture used in natural language processing (NLP). Later, this mechanism and its variants were applied to other areas, such as computer vision and speech processing. Before attention mechanisms were developed, the encoder-decoder architecture was based on stacked units of recurrent-type ANNs and long short-term memory (LSTM) [31].

As parts of neural network architectures, attention mechanisms dynamically highlight the relevant features contained in the input data. The central idea behind an attention mechanism is not to discard the intermediate states of the encoder but to use them to build the context vectors required by the decoder to generate the output data; this is done by calculating the distribution of weights in the input sequence and assigning higher values to the most relevant elements while assigning lower weights to the less relevant elements [3].

One of the most recently used attention mechanisms is multihead attention, which is a module that runs several attention mechanisms in parallel [26]. The independent attention outputs are concatenated and linearly transformed into the expected dimension. Intuitively, the use of multiple attention heads allows the mechanism to treat various parts of the sequence differently, and we express this concept as (4):

$$MultiHead(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = Concat[head_1, ..., head_h]W^O, \tag{4}$$

where:

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V), \tag{5}$$

and W denotes all trainable parameter matrices. Multihead attention is a module that uses scaled dot product attention, an attention mechanism in which dot products are scaled in the form $\sqrt{d_k}$. Formally, we have a query $\mathbf{Q}$, a key $\mathbf{K}$, and a value $\mathbf{V}$, and we compute the corresponding attention as (6):

$$Attention(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = softmax(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}})\mathbf{V}. \tag{6}$$

Figure 1 shows the graphical representations of scaled dot product attention and multihead attention.

3 Related works

To our knowledge, there are several works that have used single type (simple) architectures for the speech enhancement process [16–18], however, these works have

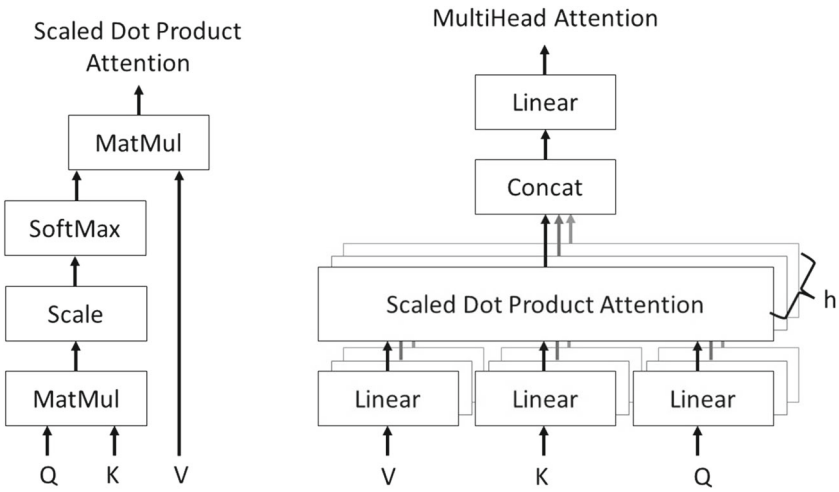

Fig. 1 Graphical representations of scaled dot product attention (left) and multihead attention (right)

not used an attention mechanism to improve efficiency. In [18], authors compare seven deep learning models belonging to the three network types analyzed in our paper. The authors evaluate overall output speech quality performance using several metrics. The results are useful for understanding how deep neural networks perform the task of speech enhancement, and authors also highlight the strengths and weaknesses of each architecture. In [17] authors made a comparison between mapping and masking approaches applied to four DNN models. The authors conclude that according to the problem at hand, it is the selection of the approach that should be applied to obtain the best results.

To our knowledge, several works have used single-type (simple) architectures for the speech enhancement process [16–18]; however, these works have not used an attention mechanism to achieve improved efficiency. Nossier et al. [18] compared seven deep learning models belonging to the three network types analyzed in our paper. The authors evaluated the overall output speech quality performance using several metrics. These results are useful for understanding how deep neural networks (DNNs) perform the task of speech enhancement, and the authors also highlighted the strengths and weaknesses of each architecture. Nossier et al. [17] conducted a comparison between mapping and masking approaches when applied to four DNN models. The authors concluded that according to the problem at hand, approach selection should be applied to obtain the best results.

Nossier et al. [16] presented a comparison of five DNNs by first implementing them in the time domain and later implementing the models in the frequency domain; their goal was to show how different networks' performances are affected by the operating domain and to determine the best-performing architecture in each domain.

Other works have included multihead attention in their neural network models; however, these are deep and complex network models [9, 10, 19, 22]. Pandey and Wang [19] proposed a dense convolutional network (DCN) with self-attention for speech enhancement in the time domain using an encoder- and decoder-based architecture with skip connections. In this architecture, each layer in the encoder and decoder comprises a dense block and an attention module.

Koizumi et al. [10] investigated a self-adaptation method for speech enhancement. They adopted multitask learning to achieve speech enhancement. In addition, the authors used multihead self-attention for capturing long-term dependencies in speech and noise. Kim et al. [9] proposed a transformer with Gaussian-weighted self-attention (T-GSA), whose attention weights are attenuated according to the distances between the target and the context symbols. The experimental results showed that the proposed T-GSA achieved significantly improved speech enhancement performance compared to the transformer and recurrent neural networks (RNNs).

In the approach presented by [22], the researchers investigated a multihead attention network for linear prediction coefficient (LPC) estimation. They aimed to produce clean speech and noise LPC parameters with negligible bias. With this, they attempted to produce higher-quality and more intelligible enhanced speech than that provided the current augmented Kalman filter-based speech enhancement algorithm. To this end, they investigated multihead attention networks within the DeepLPC framework.

4 Methodology: deep learning models

In this work, we implemented four neural network models belonging to three categories: an MLP, CNNs , and a GRU neural network. We established the four models' architectures in their basic forms to conduct an equitable comparison among the models and show the effects of each

specific network's parameters and the effect of attention on the overall performance. We kept the training setup and other speech enhancement-related elements the same for all models to implement an equitable evaluation and comparison. Figure 2 shows the implemented MLP and CNN models. Figure 3 shows the implemented GRU model and the attention module. Table 1 describes their hyperparameter configurations, and Table 2 shows the total number of trainable parameters for each model.

4.1 MLP model

We implemented the model as in Fig. 2(A) for the MLP category. The architecture has eight fully connected dense

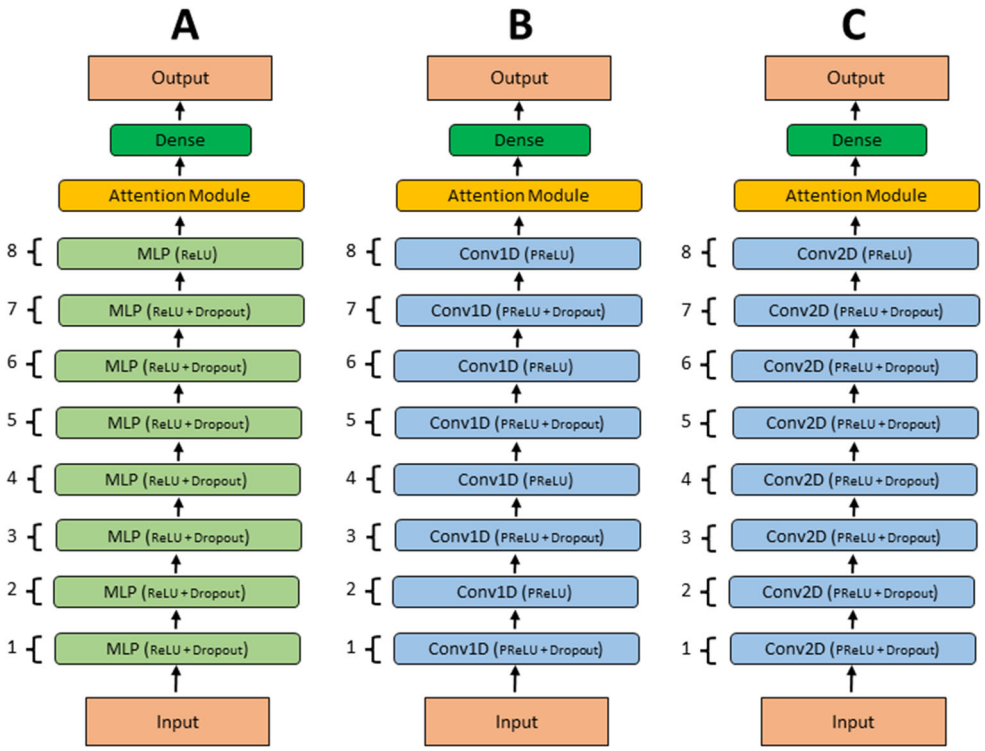

Fig. 2 The speech enhancement models: an MLP (A), a one-dimensional CNN (B), and a two-dimensional CNN (C)

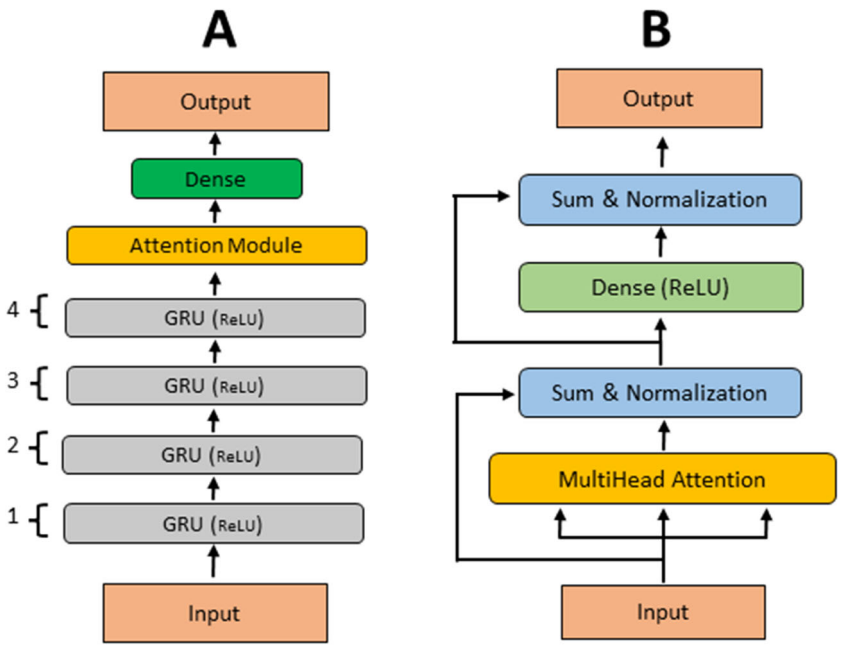

Fig. 3 The GRU speech enhancement model (A) and the attention module (B)

Table 1 Configurations of the four implemented models

Type	Kernel	Filters	Units	Activation	Dropout	Strides	Padding
MLP							
Dense	–	–	512	ReLU	0.05	–	–
Dense	–	–	512	ReLU	0.05	–	–
Dense	–	–	512	ReLU	0.05	–	–
Dense	–	–	512	ReLU	0.05	–	–
Dense	–	–	512	ReLU	0.05	–	–
Dense	–	–	512	ReLU	0.05	–	–
Dense	–	–	512	ReLU	0.05	–	–
Dense	–	–	512	ReLU	0.05	–	–
Dense	–	–	256	Linear	–	–	–
CNN–1D							
1D–Conv	16	64	–	PReLU	0.1	1	same
1D–Conv	16	64	–	PReLU	–	1	same
1D–Conv	16	64	–	PReLU	0.1	1	same
1D–Conv	16	64	–	PReLU	–	1	same
1D–Conv	16	64	–	PReLU	0.1	1	same
1D–Conv	16	64	–	PReLU	–	1	same
1D–Conv	16	64	–	PReLU	0.1	1	same
1D–Conv	16	1	–	PReLU	–	1	same
Dense	–	–	256	Linear	–	–	–
CNN–2D							
2D–Conv	4x4	64		PReLU	0.1	1	same
2D–Conv	4x4	64		PReLU	0.1	1	same
2D–Conv	4x4	64		PReLU	0.1	1	same
2D–Conv	4x4	64		PReLU	0.1	1	same
2D–Conv	4x4	64		PReLU	0.1	1	same
2D–Conv	4x4	64		PReLU	0.1	1	same
2D–Conv	4x4	64		PReLU	0.1	1	same
2D–Conv	4x4	1		PReLU	–	1	same
Dense	–	–	256	Linear	–	–	–
GRU							
GRU	–	–	256	ReLU	–	–	–
GRU	–	–	256	ReLU	–	–	–
GRU	–	–	256	ReLU	–	–	–
GRU	–	–	256	ReLU	–	–	–
Dense	–	–	256	Linear	–	–	–

layers containing 512 units with rectified linear unit (ReLU) activations. A dropout layer with a 5% rate follows each of the first seven hidden layers to avoid overfitting and stabilize the training process. The last layer is a dense layer without an activation function (also known as a dense layer with linear activation) with 256 units.

4.2 CNN models

We implemented two models for the convolution category.

The first architecture, shown in Fig. 2(B), has eight one-dimensional convolutional layers with parametric ReLU (PReLU) activations, which are followed by a dense layer containing 256 units without an activation function. We set the number of filters in the first seven convolution layers to 64, with one filter in the last convolution layer. Additionally, we used kernels of size 16 and dropout layers with 10% rates in layers one, three, five, and seven.

The second architecture, presented in Fig. 2(C), has two-dimensional convolutional layers with PReLU activations

Table 2 Total number of trainable parameters for each of the four models with and without the attention module

Model	Parameters
MLP	1,904,640
MLP + Att	1,975,820
CNN–1D	576,449
CNN–1D + Att	647,629
CNN–2D	1,381,057
CNN–2D + Att	1,480,993
GRU	1,644,800
GRU + Att	1,715,980

and a final dense layer containing 256 units without an activation function. We used the same filters as in the first convolutional architecture but utilized kernels of size 4x4 in all convolutional layers. Additionally, we employed dropout with a 10% rate in the first seven convolutional layers. Finally, we used stride lengths and padding values of 1 in the two models.

4.3 GRU model

We implemented the GRU model as shown in Fig. 3(A). The architecture has four fully connected layers containing 256 units with ReLU activations, and the last layer is a dense layer without an activation function containing 256 units. We did not use dropout layers in this model.

4.4 Attention module

We assembled and incorporated an attention module to help the models identify the most important features to improve their performance. We based the attention module on the use of multihead attention. First, the attention module consists of a multihead attention layer with four heads and dropout with a 10% rate; then, the input and output of the multihead attention layer are summed and passed to a normalization layer. Next, we add a dense layer with ReLU activation, and then we sum the input and output of the dense layer and pass the result to a second normalization layer. The attention module is shown in Fig. 3(B).

We introduce the attention module before the last linear dense layer to obtain the new performance of each of the four models.

5 Experimental setup

We trained all models using an NVIDIA Tesla P100 graphic card with 16GB memory. In addition, we performed all the experiments using a conventional computational device with 8GB memory and a 2 GHz AMD Ryzen 4-Core Processor.

5.1 Dataset

We trained and evaluated the four proposed models by utilizing audio mixtures with four datasets; we used TIMIT [4] as a speech dataset, and we used NoiseX-92 [25], DEMAND [23], and PNL-100 [6] as noise datasets.

TIMIT is a dataset containing recordings of utterances from 630 speakers representing eight dialectal divisions of American English, each speaking ten sentences with different phonetics for male and female speakers. The authors of TIMIT subdivided the material into balanced portions for training and testing (they described the subdivision criteria in [4]); we used the training and testing portions as the authors divided them. NoiseX-92 is a dataset composed of recordings of several types of acoustic noises, such as electric cutting and welding equipment noises, white noise, military noises, and vehicle noises. DEMAND contains recordings of various acoustic noises in indoor environments (domestic, office, public, and transportation noises) and outdoor environments (street and nature noises). Finally, PNL-100 contains environmental sounds such as machine noises, animal sounds, footsteps, and door movements.

We selected the training portion of the TIMIT dataset and the noise from the NoiseX-92 and DEMAND datasets for the training and validation sets. Then, we created five hours of audio mixes in one-minute clips with SNRs uniformly sampled between -10 dB and 10 dB (whereby the different noises corrupted the speech signal). We randomly chose both speech and noise clips. Then, we sampled the audio at 8 kHz to feed the model with the most relevant frequency bands. Next, we calculated the power spectrum of the signal magnitude using the STFT with a size of 510-point fast Fourier transform (FFT) which yields 256 frequency bins, a frame length window of 32 ms (256 samples), and an overlap of 50% (128 samples). Finally, we normalized the training and validation data to zero means and unit variances to facilitate the training process.

To evaluate the models, we made two test sets. First, we created new audio mixtures with uniformly sampled SNRs of -10 dB, -5 dB, 0 dB, 5 dB, and 10 dB using the testing portion of the TIMIT dataset and the noise from the NoiseX-92 and DEMAND datasets. Furthermore, we repeated the same process for the second test set but used the testing portion of the TIMIT dataset and the noise from PNL-100; we used this second group of audio mixtures to evaluate the generalization capabilities of the trained models.

We retained the signal's phase only during the model prediction process and then added it to the estimated clean signal, similar to the approach shown in [12] and [18].

5.2 Training setup

Our training strategy consisted of mapping the magnitude spectrogram as the training target. We implemented the four models using the Keras library with TensorFlow. The loss function we used during the training process was the MSE because our goal was to improve all evaluation metrics, not a specific one. We employed Adam as an optimizer with a learning rate of 0.0001, a b1 value of 0.9, a b2 value of 0.999, and an epsilon value of 1e-08. We employed a batch size of 64 and used 10% of the training data for validation to monitor and control the network's performance and prevent overfitting. In addition, we selected accuracy as the metric to monitor during training.

We set the training duration to 60 epochs and implemented two strategies during training. The first was an early stopping strategy whereby the training process stopped if the monitored metric stopped improving after six consecutive epochs. Second, a learning rate reduction strategy, with a factor of 0.8 and a minimum learning rate of 0.000001, was applied to every epoch in which the monitored metric stopped improving.

We kept the same training setup for all architectures to conduct an equitable evaluation and comparison.

6 Results and discussion

This section presents the results and discussion. We performed two experiments. The first experiment showed the models' performance on new sound mixes created with the testing portion of the TIMIT dataset and noise from NoiseX-92 and DEMAND (the same noises as those used during training). In the second experiment, we tested the generalization capabilities of the models on sound mixes created with the testing portion of the TIMIT dataset and noise from PNL-100.

To evaluate the performance of our models, we used three evaluation metrics: the perceptual evaluation of speech quality (PESQ) [20]; the short-term objective intelligibility (STOI) [7]; and the scale-invariant signal-to-distortion ratio (SI-SDR) [21], which are the standard metrics that are most commonly used to evaluate the performance of proposals for solving the speech enhancement problem.

PESQ values range from -0.5 to 4.5; the higher the value is, the better the speech quality. STOI values typically range from 0 to 1, where a higher value indicates better intelligibility (usually converted to a percentage). The SI-SDR is used to calculate the amount of distortion introduced by the speech separation process. The SDR is one of the standard speech separation evaluation metrics that measures the amount of distortion introduced by the separated signal; the SDR is the ratio between the energy of the clean signal and the distortion energy. Therefore, higher SDR values indicate better speech separation performance.

6.1 MLP results

With the inclusion of the attention module in the MLP neural network, we obtained worse results in terms of the STOI and SI-SDR metrics at low SNR levels (-10 dB to 0 dB), while PESQ showed that the improvements were minimal. However, at higher SNR levels (5 dB to 10 dB), the inclusion of the attention module yielded improvement in all three metrics used, which indicates that MLP neural networks can be improved by including an attention module as long as the volume of noise in the audio signals remains low. Table 3 shows the complete results obtained with SNR levels of -10 dB, -5 dB, 0 dB, 5 dB, and 10 dB.

6.2 CNN results

When we incorporated the attention module into the one-dimensional convolutional architecture, the STOI and SI-SDR metrics showed improvements as the SNR level ranged from -5 dB to 10 dB. In contrast, the PESQ metric exhibited improvements only at SNR levels of -10 dB, 0 dB, and 5 dB. However, at -5 dB and 10 dB, the PESQ

Table 3 The STOI, PESQ, and SI-SDR results obtained by the MLP model on the test set in a context with seen noise

	STOI (%)		PESQ		SI-SDR	
SNR	single	attention	single	attention	single	attention
-10	68.25%	68.16%	2.42	2.43	6.08	6.07
-5	79.79%	79.77%	2.70	2.71	10.68	10.61
0	87.94%	88.08%	3.03	3.04	14.05	13.94
5	92.21%	92.61%	3.37	3.39	19.67	19.95
10	94.88%	95.45%	3.63	3.65	18.27	18.48

measure showed that the speech quality remained the same. In the case with the lowest SNR level (− 10 dB), we found no improvement in the STOI and SI-SDR metrics. Table 4 shows the complete results.

Next, when using two-dimensional convolutional networks with the attention module, we observed improvements in the STOI metric with SNR levels from − 5 dB to 10 dB; PESQ showed improvements except at an SNR of 0 dB, and the SI-SDR only exhibited improvements from 0 dB to 10 dB. Table 5 shows the complete results.

6.3 GRU results

Finally, after adding the attention module to the GRU model, Table 6 shows that improvements were achieved in terms of most of the three metrics, except for STOI at − 10 dB or PESQ at 5 dB and 10 dB, where the results did not change.

6.4 Generalization capability results

One of the problems with neural network-based speech enhancement is having a network that performs well on the training dataset but cannot generalize and maintain the same performance on new data. This problem is known as a variance or overfitting problem. Therefore, testing the generalization capabilities of the models is crucial for conducting an equitable comparison among them.

The model generalization capabilities were evaluated by comparing the results of models with/without the attention module in terms of the metrics obtained on the Noise-x and DEMAND datasets (seen noise dataset) and with those obtained on the noise from the NL-100 dataset (unseen noise dataset).

We evaluated the generalization capabilities of the four implemented models by testing the models' performance with new noises from the PNL-100 dataset. Tables 7, 8, 9,

Table 4 The STOI, PESQ, and SI-SDR results obtained by the one-dimensional CNN model on the test set in a context with seen noise

	STOI (%)		PESQ		SI-SDR	
SNR	single	attention	single	attention	single	attention
− 10	69.36%	69.30%	2.44	2.45	6.28	6.27
− 5	81.55%	81.67%	2.79	2.79	11.48	11.49
0	89.79%	90.06%	3.15	3.16	15.97	16.13
5	94.24%	95.43%	3.48	3.55	20.44	24.28
10	97.17%	97.44%	3.79	3.79	22.77	23.38

Table 5 The STOI, PESQ, and SI-SDR results obtained by the two-dimensional CNN model on the test set in a context with seen noise

	STOI (%)		PESQ		SI-SDR	
SNR	single	attention	single	attention	single	attention
− 10	68.80%	68.75%	2.45	2.46	6.38	6.23
− 5	80.88%	81.02%	2.79	2.80	11.36	11.34
0	89.29%	89.54%	3.16	3.15	15.52	15.69
5	93.51%	94.98%	3.47	3.56	18.17	23.24
10	96.28%	96.91%	3.77	3.78	21.23	22.01

Table 6 The STOI, PESQ, and SI-SDR results obtained by the GRU model on the test set in a context with seen noise

	STOI (%)		PESQ		SI-SDR	
SNR	single	attention	single	attention	single	attention
− 10	69.45%	69.44%	2.42	2.43	6.16	6.19
− 5	81.37%	81.42%	2.76	2.77	11.38	11.44
0	89.81%	89.90%	3.10	3.12	15.92	16.03
5	95.30%	95.39%	3.52	3.52	24.27	24.47
10	97.17%	97.37%	3.75	3.75	22.88	23.03

Table 7 The STOI, PESQ, and SI-SDR results obtained by the MLP model on the test set in a context with unseen noise

	STOI (%)		PESQ		SI-SDR	
SNR	single	attention	single	attention	single	attention
− 10	73.05%	73.88%	2.49	2.50	2.75	2.78
− 5	86.55%	87.05%	3.10	3.11	10.28	10.22
0	88.19%	88.96%	3.14	3.17	12.32	12.38
5	92.32%	93.03%	3.43	3.46	16.14	16.34
10	94.51%	95.17%	3.67	3.70	18.64	18.99

Table 8 The STOI, PESQ, and SI-SDR results obtained by the one-dimensional CNN model on the test set in a context with unseen noise

	STOI (%)		PESQ		SI-SDR	
SNR	single	attention	single	attention	single	attention
− 10	76.32%	76.28%	2.52	2.55	3.17	3.65
− 5	88.72%	89.60%	3.18	3.17	10.64	10.73
0	90.84%	91.63%	3.20	3.23	12.86	13.25
5	94.66%	95.55%	3.49	3.53	17.31	17.91
10	96.65%	97.60%	3.75	3.78	21.00	22.21

Table 9 The STOI, PESQ, and SI-SDR results obtained by the two-dimensional CNN model on the test set in a context with unseen noise

	STOI (%)		PESQ		SI-SDR	
SNR	single	attention	single	attention	single	attention
− 10	74.70%	75.28%	2.46	2.48	2.79	2.93
− 5	87.65%	89.21%	3.18	3.17	10.23	10.55
0	89.94%	90.64%	3.16	3.18	12.46	12.77
5	93.79%	94.89%	3.47	3.49	16.64	17.40
10	95.72%	97.02%	3.72	3.76	19.71	21.37

and 10 show the complete generalization results obtained by the four models with and without the attention module for SNR levels of − 10 dB, − 5 dB, 0 dB, 5 dB, and 10 dB.

From a general perspective, the inclusion of an attention module yielded improvements in the generalization capabilities of the MLP and convolution-based models, with some exceptions. However, the inclusion of the attention module in the GRU model produced results contrary to those expected in terms of most metrics.

6.5 Overall results

Figures 4, 5, and 6 provide the average SNR results for PESQ, STOI, and the SI-SDR, respectively. The models with the text "seen dataset" show the mixes of the testing portion of the TIMIT dataset with noise from NoiseX-92 and DEMAND. The models with the text "unseen dataset" show the mixes of the testing portion of the TIMIT dataset with noise from PNL-100.

For the averaged results of the PESQ metric, the models tested with seen noise in Fig. 4(A) showed that the convolutional models obtained similar results in audio mixtures with noises used during training. On the other hand, the MLP model achieved the lowest performance. However, all four models exhibited improvements when including the attention module. Nevertheless, in the experiments that showed the generalization capacities capabilities of the models with unseen noise (Fig. 4(B)), the one-dimensional convolutional model yielded the best performance with and without the attention module. It is also relevant to mention that the GRU-based model obtained the second-best generalization capabilicity result without the attention mechanism but obtained worse results when incorporating the attention module.

Table 10 The STOI, PESQ, and SI-SDR results obtained by the GRU model on the test set in a context with unseen noise

	STOI (%)		PESQ		SI-SDR	
SNR	single	attention	single	attention	single	attention
− 10	75.87%	75.62%	2.49	2.49	2.97	2.87
− 5	89.39%	89.35%	3.15	3.14	10.83	10.73
0	91.02%	90.84%	3.17	3.16	12.84	12.79
5	95.17%	95.13%	3.48	3.47	17.64	17.63
10	97.38%	97.43%	3.74	3.74	22.08	22.14

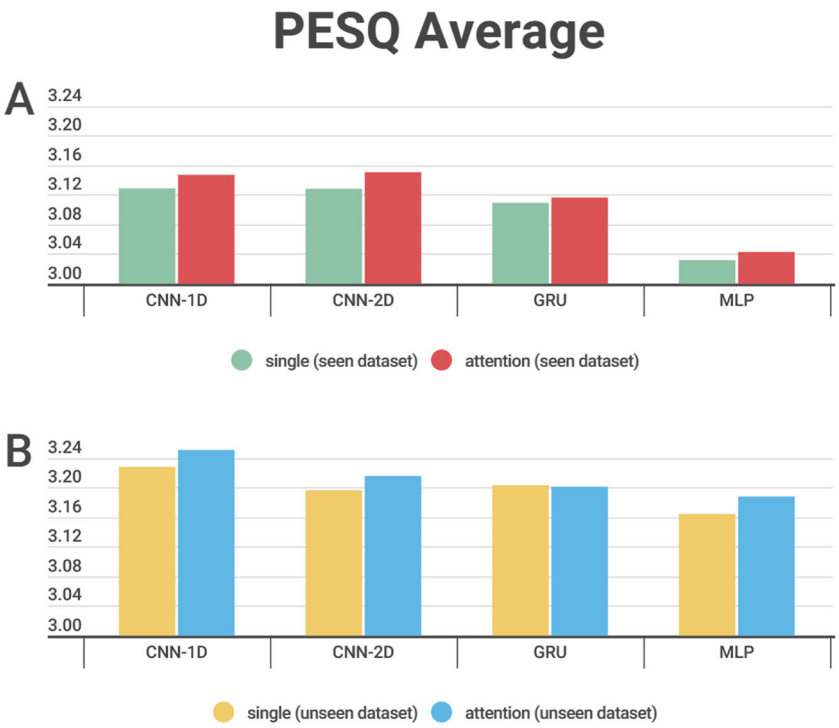

Fig. 4 Average PESQ results of the four models (with and without attention). Results obtained on the testing portion of the TIMIT dataset and noise from NoiseX-92 and DEMAND (A); results obtained on the testing portion of the TIMIT dataset and noise from PNL-100 (B)

Regarding the average STOI metric results obtained with seen noise, Fig. 5(A) shows that the one-dimensional CNN with the attention module reached the best result, followed by the GRU-based model with the attention module. Similar to the PESQ metric results, the MLP model achieved the lowest performance. In the experiments showing the generalization capabilities of the models (Fig. 5(B)), the same one-dimensional convolutional model with the attention module also obtained the highest values. Finally, similar to the PESQ results, the GRU-based model underperformed when including the attention module.

In terms of the average results of the SI-SDR metric, Fig. 6(A) shows the same pattern as that of STOI with seen noise, where the one-dimensional neural network with the attention module achieved the best result, followed by the GRU-based model with the attention module. On the other hand, Fig. 6(B) shows that the average SI-SDR metric results were lower with the unseen noise. Consequently, we can interpret the result as the amount of distortion introduced by the separation process of these data being higher.

From the results of the experiments, we conclude that the convolutional models with the attention mechanism could extract the necessary features and learn much better patterns that allowed them to reconstruct the magnitude spectrograms of the enhanced speech signals on both the seen and unseen datasets. Furthermore, the degree of overfitting to the training data was meager. The MLP models were also able to obtain the features and patterns needed to reconstruct the spectrogram but to a lesser degree than the convolutional models. Finally, we observed that the GRU models did not benefit from including the attention module in the generalization experiments. After performing an analysis of the GRU architecture with the

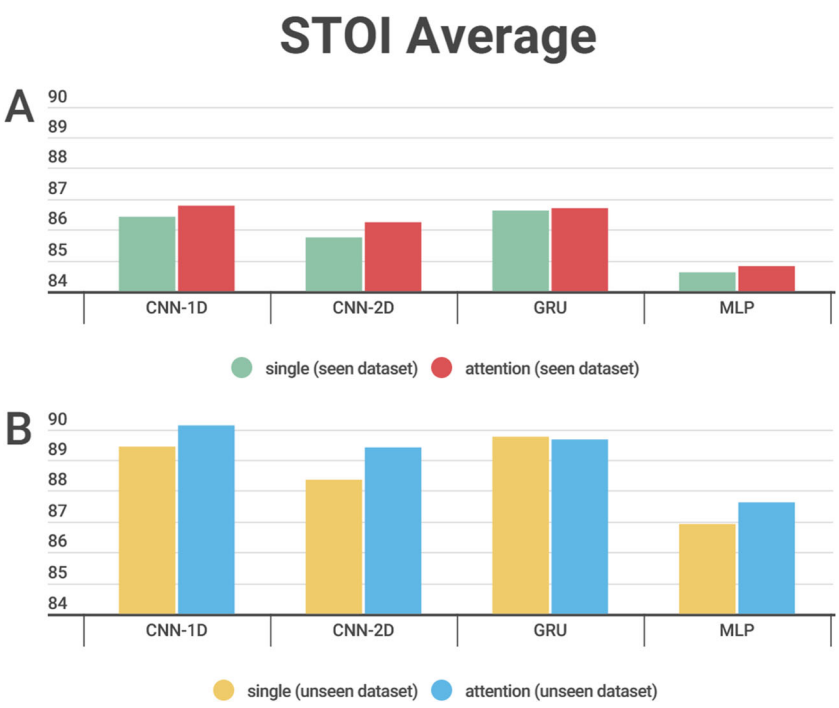

Fig. 5 Average STOI results of the four models (with and without attention). Results obtained on the testing portion of the TIMIT dataset and noise from NoiseX-92 and DEMAND (A); results obtained on the testing portion of the TIMIT dataset and noise from PNL-100 (B)

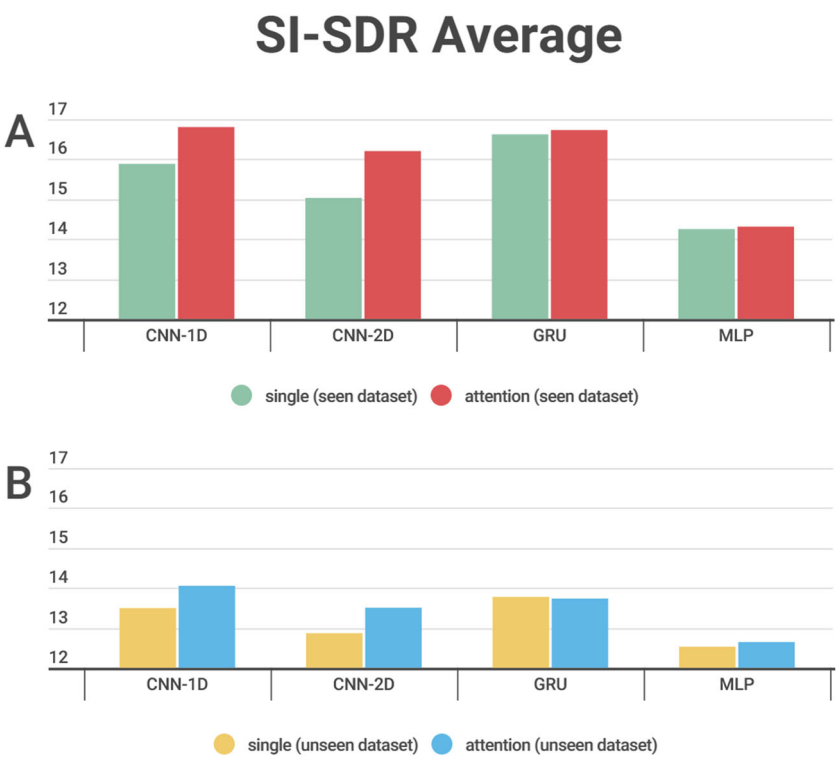

Fig. 6 Average SI-SDR results of the four models (with and without attention). Results obtained on the testing portion of the TIMIT dataset and noise from NoiseX-92 and DEMAND (A); results obtained on the testing portion of the TIMIT dataset and noise from PNL-100 (B)

attention module, we conclude that this behavior is the result of overfitting the noise used during training. A further analysis for improving the performance of the GRU models by incorporating an attention module based on multihead attention is beyond the scope of this research; however, we will consider it in future research.

6.6 Training observations

Figure 7 shows the training and validation data loss curves of the four models (eight when considering the models with the attention module) to compare the stability of their model training processes. The data loss curves show the complete

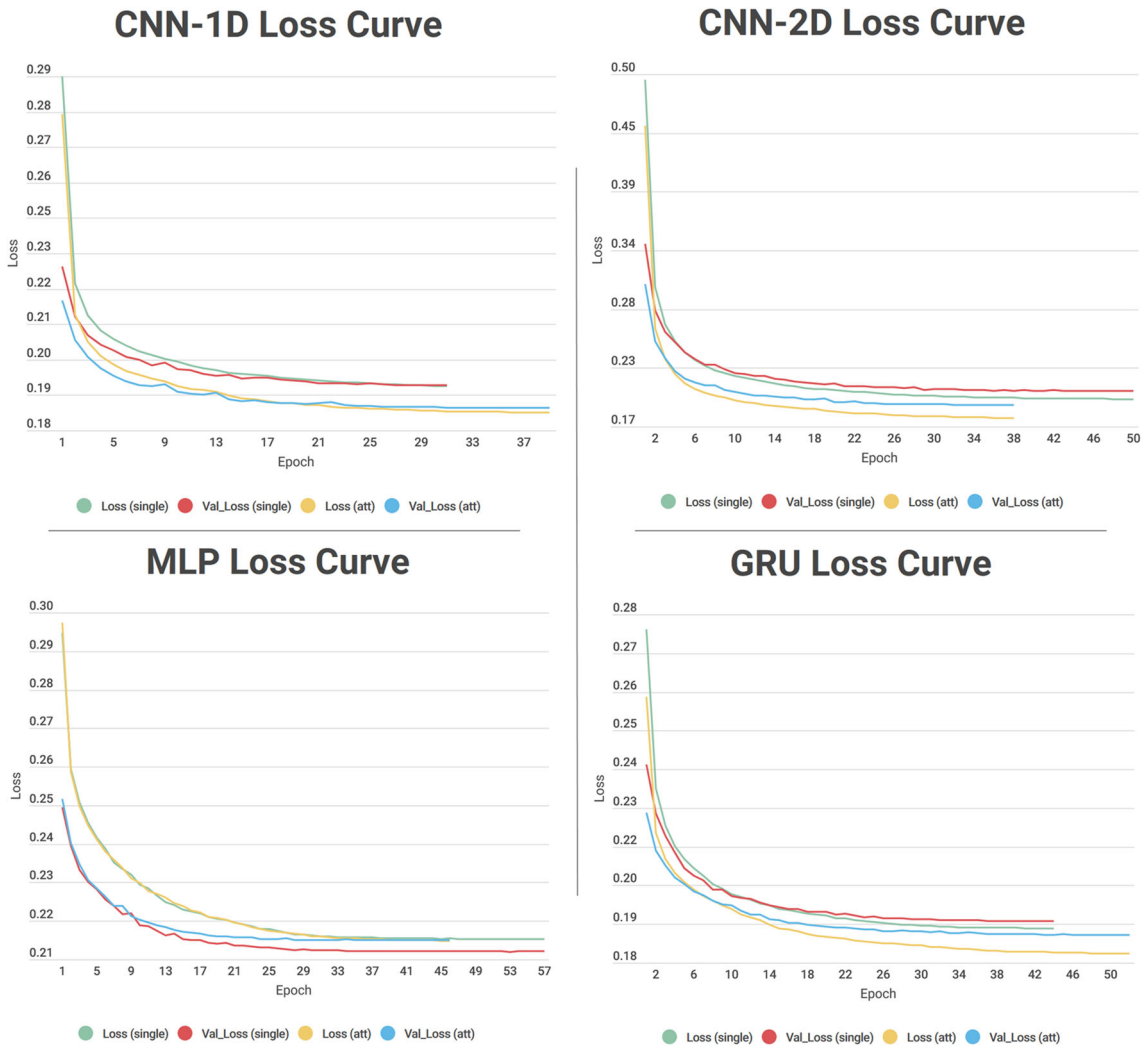

Fig. 7 The training loss curves of the four models (with and without attention) for the training and validation data

training phase; however, we used an early stopping strategy, so the models converged six epochs before the last epoch shown on each curve. The one-dimensional convolutional model converged in the least number of epochs with the slightest degree of overfitting. On the other hand, the MLP model training process took the most epochs, 57 in total, converging at epoch 51.

6.7 Discussion

Different experiments showed that including an attention module based on multihead attention in different neural networks improves models' performance in speech enhancement problems. Moreover, the achieved improvements were more significant in one-dimensional convolutional networks than in MLP or GRU models, even when we evaluated the model with new data. In the GRU neural networks, we observed model improvements when incorporating the attention module based on multihead attention into the experiments with the same noise we used during training. However, we obtained worse results when testing the model generalization capacity with the attention module on new data. The MLP models exhibited improvement by including the attention module; even so, their performance was lower than that of the convolutional networks, which can have between a third and a quarter of the number of trainable parameters utilized by the MLP.

Some considerations used in our experiments could have influenced the results.

First, although we corrupted the speech signals using only one noise signal at a time, the noise of the DEMAND dataset (used during training) contained recordings of acoustic noises in indoor and outdoor environments where different noise sources (of different natures) were simultaneously present. The use of the DEMAND dataset introduces complex noises from real natural environments,

which are very challenging to process for neural network models.

Second, we decided to implement only the STFT at 256 FFT and normalization as preprocessing steps for the audio signals, without applying any additional frequency filtering or other processing strategies; thus, we did not add additional computational loads and test the true capabilities of the different types of neural networks.

Third, we used SNRs from -10 dB to 10 dB, as this is a range in which research studies are commonly conducted, and tried to cover an important variety of noise scenarios. However, scenarios with very high noise levels, such as aircraft cabin noise conditions during takeoff, were not considered. This represents a limitation of this study.

In the literature, some works have explored the applications of attention (based on multihead attention) to the problem of speech enhancement [14, 19, 22, 29]; these works showed that the implementation of attention improves the results of neural network models. However, it is necessary to mention that these works proposed complex architectures such as hybrid neural networks with many trainable parameters, residual connections, or ramifications and used different datasets with different processing techniques.

Table 11 shows the average results in terms of two objective metrics that are commonly used in speech enhancement. The table compares the values obtained by our four neural network models against those in the works of [14, 19, 22], and [29].

It is necessary to clarify that the values obtained by [19] are the average SNRs from -5 dB to 5 dB; the values obtained by [14, 22] and [29] are the average SNRs from -5 dB to 15 dB, and the values obtained by [9] are the average SNRs from -10 dB to 15 dB. Some of the works used different speech and noise corpora. Our results in Table 11 are the average SNRs from -5 dB to 10 dB obtained by the models on the test set in a context with seen noise.

The results of our experiments reflect that the inclusion of an attention module allows neural network model architectures to be less complex without sacrificing efficiency. Furthermore, our results intend to help other researchers reduce the time and resources commonly invested during the iterative trial-and-error process of training new MLP, CNN, or GRU neural network models that incorporate attention-based mechanisms.

Table 11 Average PESQ and STOI comparisons among different models

Model	PESQ average	STOI average
MLP + Att	3.20	88.98%
CNN–1D + Att	3.32	91.15%
CNN–2D + Att	3.32	90.61%
GRU + Att	3.29	91.02%
DCN–SM [19]	2.83	90.40%
DeepLPC–MHANet [22]	2.59	88.41%
MHANet [14]	2.88	93.60%
SE–T [29]	2.62	93.00%

We considered it essential that the evaluated models were neural network models with simple architectures. By doing so, we conclude that it is possible to create models with simple architectures (and relatively few trainable parameters) that obtain competitive results and consume few computational resources in their implementations, which is appropriate when we have computational devices with conventional or limited resources.

Finally, since this research did not use predefined architectures, but we built all of them from scratch under the considerations of simplicity, it was not necessary to perform an ablation study, which is typical of transfer learning and style transfer.

7 Conclusion

In this work, we proposed three types of simple neural networks (MLP, CNN and GRU models) with attention mechanisms (based on multihead attention) to solve the speech enhancement problem. Thus, we evaluated four models belonging to the above three types of neural networks with respect to speech quality using the objective PESQ, STOI, and SI-SDR, evaluation metrics as well as their generalization capabilities when integrating attention modules.

Regarding the evaluation and comparison of the different models, the one-dimensional convolutional model produced the best results when integrating the attention module for speech enhancement (according to the utilized metrics). Our finding is consistent with that of [30], where the authors reported that a CNN could efficiently learn information from speech signals.

In contrast, the GRU-based model yielded worse values when integrating the attention module and when evaluated with audio signals that were not part of the signals used for training (generalization capability). Furthermore, challenging noisy environments, such as audio signals with SNRs of -10 dB and -5 dB, negatively affected the performance of the evaluated models. However, the overall performance remained acceptable.

Since it is possible to divide the training targets in speech enhancement into two types, mapping and masking targets [27], in future works, we will design a comparison using mapping and masking strategies. This task can be defined as a regression problem if the target is to map a time-frequency representation of clean speech directly or as a classification problem if the target is to produce a matrix (known as a

mask) that classifies each portion of the signal (as speech or noise) to filter the noisy speech with this mask to generate an enhanced clean speech signal. We will also experiment using more simultaneously present noise at different SNRs to generate an environment that is more similar to a natural environment in which different noise sources with different natures are simultaneously present.

Acknowledgements We would like to thank Dr. Matias Garcia-Constantino for his useful comments that improved the quality of this paper. The authors would like to thank the Laboratorio Nacional de Supercómputo del Sureste de México (LNS), a member of CONACYT, for the computational resources, support and technical assistance provided through project No. 202103086N.

Data Availability The datasets generated and analyzed during the current study are available from the corresponding author upon reasonable request.

Declarations

Competing interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Brauwers G, Frasincar F (2021) A general survey on attention mechanisms in deep learning. IEEE Trans Knowl Data Eng:1–1. https://doi.org/10.1109/TKDE.2021.3126456
2. Fan C, Yi J, Tao J et al (2021) Gated recurrent fusion with joint training framework for robust end-to-end speech recognition. IEEE/ACM Trans Audio Speech Language Process 29:198–209. https://doi.org/10.1109/TASLP.2020.3039600
3. Galassi A, Lippi M, Torroni P (2020) Attention in natural language processing. IEEE Trans Neural Netw Learn Syst 32(10):4291–4308. https://doi.org/10.1109/TNNLS.2020.3019893
4. Garofolo J, Lamel L, Fisher W et al (1992) Timit acoustic-phonetic continuous speech corpus. Linguis Data Consortium. https://doi.org/10.35111/17gk-bn40
5. Hatzopoulos S, Ciorba AH, Skarzynski P (eds) (2020) The human auditory system - basic features and updates on audiological diagnosis and therapy. IntechOpen, Rijeka. https://doi.org/10.5772/intechopen.77713
6. Hu G, Wang D (2010) A tandem algorithm for pitch estimation and voiced speech segregation. IEEE Trans Audio Speech Lang Process 18(8):2067–2079. https://doi.org/10.1109/TASL.2010.2041110
7. Jensen J, Taal CH, Jensen J et al (2016) An algorithm for predicting the intelligibility of speech masked by modulated noise maskers. IEEE/ACM Transactions on Audio. Speech Lang Process 24(11):2009–2022. https://doi.org/10.1109/TASLP.2016.2585878
8. Kamath U, Graham K, Emara W (2022) Transformers for Machine Learning: A Deep Dive. Chapman and Hall/CRC, New York. https://doi.org/10.1201/9781003170082
9. Kim J, El-Khamy M, Lee J (2020) T-GSA: transformer with Gaussian-weighted self-attention for speech enhancement. In: IEEE international conference on acoustics, speech and signal processing, pp 6649–6653. https://doi.org/10.1109/ICASSP40776.2020.9053591 . iSSN: 2379-190X
10. Koizumi Y, Yatabe K, Delcroix M et al (2020) Speech enhancement using self-adaptation and multi-head self-attention. In: IEEE international conference on acoustics, speech and signal processing, pp 181–185. https://doi.org/10.1109/ICASSP40776.2020.9053214 . iSSN: 2379-190X
11. Lan T, Ye W, Lyu Y et al (2020) Embedding encoder-decoder with attention mechanism for monaural speech enhancement. Ieee Access 685(96):677–96. https://doi.org/10.1109/ACCESS.2020.2995346
12. Li L, Lu Z, Watzel T et al (2021) Light-weight self-attention augmented generative adversarial networks for speech enhancement. Electronics 10(13):1586. https://doi.org/10.3390/electronics10131586
13. McLoughlin I (2009) Applied Speech and Audio Processing: with Matlab Examples. Cambridge University Press, Cambridge. https://doi.org/10.1017/CBO9780511609640
14. Nicolson A, Paliwal KK (2020) Masked multi-head self-attention for causal speech enhancement. Speech Comm 125:80–96. https://doi.org/10.1016/j.specom.2020.10.004
15. Niu Z, Zhong G, Yu H (2021) A review on the attention mechanism of deep learning. Neurocomputing 452:48–62. https://doi.org/10.1016/j.neucom.2021.03.091
16. Nossier SA, Wall J, Moniri M et al (2020a) A comparative study of time and frequency domain approaches to deep learning based speech enhancement. In: International Joint Conference on Neural Networks, pp 1–8. https://doi.org/10.1109/IJCNN48605.2020.9206928 . iSSN: 2161-4407
17. Nossier SA, Wall J, Moniri M et al (2020b) Mapping and masking targets comparison using different deep learning based speech enhancement architectures. In: International joint conference on neural networks, pp 1–8. https://doi.org/10.1109/IJCNN48605.2020.9206623 . iSSN: 2161-4407
18. Nossier SA, Wall J, Moniri M et al (2021) An experimental analysis of deep learning architectures for supervised speech enhancement. Electronics 10(1):17. https://doi.org/10.3390/electronics10010017
19. Pandey A, Wang D (2021) Dense CNN with self-attention for time-domain speech enhancement. IEEE/ACM Trans Audio Speech Lang Process 29:1270–1279. https://doi.org/10.1109/TASLP.2021.3064421
20. Rix A, Beerends J, Hollier M et al (2001) Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs. In: IEEE international conference on acoustics, speech, and signal processing, vol 2, pp 749–752. https://doi.org/10.1109/ICASSP.2001.941023. iSSN: 1520-6149
21. Roux JL, Wisdom S, Erdogan H et al (2019) SDR – Half-baked or well done? In: IEEE international conference on acoustics, speech and signal processing, pp 626–630. https://doi.org/10.1109/ICASSP.2019.8683855. iSSN: 2379-190X
22. Roy SK, Nicolson A, Paliwal KK (2021) DeepLPC-MHANet: multi-head self-attention for augmented kalman filter-based speech enhancement. IEEE Access 9:70,516–70,530. https://doi.org/10.1109/ACCESS.2021.3077281
23. Thiemann J, Ito N, Vincent E (2013) The diverse environments multi-channel acoustic noise database (demand): a database of multichannel environmental noise recordings. Proc Meetings Acoust 19(1):035–081. https://doi.org/10.1121/1.4799597
24. Tomar NK, Jha D, Riegler MA et al (2022) FANEt: a feedback attention network for improved biomedical image segmentation. IEEE Trans Neural Netw Learn Syst:1–14. https://doi.org/10.1109/TNNLS.2022.3159394
25. Varga A, Steeneken HJM (1993) Assessment for automatic speech recognition ii: Noisex-92: a database and an experiment to study the effect of additive noise on speech recognition systems. Speech

Commun 12(3):247–251. https://doi.org/10.1016/0167-6393(93)90095-3
26. Vaswani A, Shazeer N, Parmar N et al (2017) Attention is all you need. Adv Neural Inf Process Syst 30
27. Wang D, Chen J (2018) Supervised speech separation based on deep learning: an overview. IEEE/ACM Trans Audio, Speech and Lang Proc 26(10):1702–1726. https://doi.org/10.1109/TASLP.2018.2842159
28. Ye JC (2022) Geometry of Deep Learning: A Signal Processing Perspective, Mathematics in Industry, vol 37. Springer, Singapore. https://doi.org/10.1007/978-981-16-6046-7
29. Yu W, Zhou J, Wang H et al (2022) SETRansformer: speech enhancement transformer. Cognit Comput 14(3):1152–1158. https://doi.org/10.1007/s12559-020-09817-2
30. Yuliani AR, Amri MF, Suryawati E et al (2021) Speech enhancement using deep learning methods: a review. J Elektronika dan Telekomunikasi 21(1):19–26. https://doi.org/10.14203/jet.v21.19-26
31. Zacarias-Morales N, Pancardo P, Hernández-Nolasco JA et al (2021) Attention-inspired artificial neural networks for speech processing: a systematic review. Symmetry 13(2):214. https://doi.org/10.3390/sym13020214
32. Zhang L, Wang M, Zhang Q et al (2020) Environmental attention-guided branchy neural network for speech enhancement. Appl Sci Basel 10(3):1167. https://doi.org/10.3390/app10031167
33. Zhu T, Cheng C (2020) Joint CTC-attention end-to-end speech recognition with a triangle recurrent neural network encoder. J Shanghai Jiaotong University (Sci) 25(1):70–75. https://doi.org/10.1007/s12204-019-2147-6

Noel Zacarias-Morales is a graduate of the PhD in Computer Science at the Academic Division of Information Sciences and Technologies at the Universidad Juárez Autónoma de Tabasco. He obtained his Master's degree in Information Technology Management at the Universidad Juárez Autónoma de Tabasco in 2019. His research interests are the development of models based on deep learning (artificial neural networks) applied to signal analysis and processing for prediction and classification.

José Adán Hernández-Nolasco received the bachelor's degree in electronic and communications engineering from the Autonomous University of Nuevo Leon, in 1996, the M.Sc. degree in electronic engineering (telecommunications) from the Monterrey Institute of Technology and Higher Education, in 2003, and the Ph.D. degree in optics from the National Institute for Astrophysics, Optics and Electronics, in 2012. He has been a Research Professor with the Universidad Juárez Autónoma de Tabasco, for 25 years. He has authored or coauthored over 25 publications in the areas of ambient intelligence and AI applications, and over 30 participations in conferences. His research interests include artificial intelligence, fuzzy logic, IoT and Optics.

Pablo Pancardo received the M.Sc. degree in information technology from the Monterrey Institute of Technology and Advanced Studies (ITESM), in 1998, and the Ph.D. degree in computer science from the Juarez Autonomous University of Tabasco (UJAT), in 2016. He is a Senior Lecturer leading the Intelligent Data Sensing and Processing Group at UJAT. He has more than 40 publications in relevant international conferences and journals on Ubiquitous Computing and AI applications related to welfare and healthcare. He has led/taken part in several research projects involving the adoption of ubiquitous mobile computing and sensor networks. His research interests include artificial intelligence, fuzzy logic, the IoT, and HCI.