applied sciences

MDPI

Article

TASE: Task-Aware Speech Enhancement for Wake-Up Word Detection in Voice Assistants [†]

Guillermo Cámbara [1,2,*,‡], Fernando López [3,4,‡], David Bonet [1,5,‡], Pablo Gómez [3], Carlos Segura [1], Mireia Farrús [6] and Jordi Luque [1,*]

[1] Telefónica I+D, Research, 08019 Barcelona, Spain; david.bonet.practicas@telefonica.com (D.B.); carlos.seguraperales@telefonica.com (C.S.)
[2] TALN Natural Language Processing Group, Universitat Pompeu Fabra, 08018 Barcelona, Spain
[3] Telefónica I+D, Digital Home, 28050 Madrid, Spain; wiliam.lopezgavilanez@telefonica.com (F.L.); pablo.gomezguerrero@telefonica.com (P.G.)
[4] AUDIAS Audio, Data Intelligence and Speech Group, Universidad Autónoma de Madrid, 28049 Madrid, Spain
[5] TSC Signal Theory and Communications Department, Universitat Politècnica de Catalunya, 08034 Barcelona, Spain
[6] CLiC Language and Computation Center, UBICS UB Institute of Complex Systems, Universitat de Barcelona, 08007 Barcelona, Spain; mfarrus@ub.edu
[*] Correspondence: guillermo.cambara@upf.edu (G.C.); jordi.luque@telefonica.com (J.L.)

[†] This work is an extended version of our previous IberSPEECH2020 conference paper.
[‡] These authors contributed equally to this work.

Abstract: Wake-up word spotting in noisy environments is a critical task for an excellent user experience with voice assistants. Unwanted activation of the device is often due to the presence of noises coming from background conversations, TVs, or other domestic appliances. In this work, we propose the use of a speech enhancement convolutional autoencoder, coupled with on-device keyword spotting, aimed at improving the trigger word detection in noisy environments. The end-to-end system learns by optimizing a linear combination of losses: a reconstruction-based loss, both at the log-mel spectrogram and at the waveform level, as well as a specific task loss that accounts for the cross-entropy error reported along the keyword spotting detection. We experiment with several neural network classifiers and report that deeply coupling the speech enhancement together with a wake-up word detector, e.g., by jointly training them, significantly improves the performance in the noisiest conditions. Additionally, we introduce a new publicly available speech database recorded for the Telefónica's voice assistant, Aura. The OK Aura Wake-up Word Dataset incorporates rich metadata, such as speaker demographics or room conditions, and comprises hard negative examples that were studiously selected to present different levels of phonetic similarity with respect to the trigger words "OK Aura".

Keywords: speech enhancement; wake-up word; keyword spotting; deep learning; convolutional neural network

1. Introduction

Cognitive conversation systems are becoming ubiquitous. They are present at many user's devices or employed for offering company's services and customer care through conversational interfaces. This increase in popularity is mainly due to an efficient interface build upon the most natural way of communication: speech. Commonly, one of the cornerstones of such systems is the speech-to-text (S2T) technology, in charge of properly transcribing the user's speech into text. The resulting transcription is then further processed across the pipeline of a natural language engine, for example, to extract user's intent. The previous design makes it difficult to recover from word errors or inaccurate sentences

check for updates

Citation: Cámbara, G.; López, F.; Bonet, D.; Gómez, P.; Segura, C.; Farrús, M.; Luque, J. TASE: Task-Aware Speech Enhancement for Wake-Up Word Detection in Voice Assistants. *Appl. Sci.* **2022**, *12*, 1974. https://doi.org/10.3390/app12041974

Academic Editor: Douglas O'Shaughnessy

Received: 30 December 2021
Accepted: 7 February 2022
Published: 14 February 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

coming from the S2T interface. Furthermore, S2T modules tend to be highly complex, computationally expensive, and, most of the time, prohibitive for low-resourced or embedded devices. They are required to operate under both highly variable and noisy scenarios and, consequently, they are often specifically fine-tuned to efficiently tackle the diversity of vocabulary size, prosody, or background noises, among others, within a specific language domain. With the aim of avoiding such an excessive usage of resources along the inference stage, it is common to require the pronunciation of a wake-up word (WUW), which triggers the S2T functionality and the rest of conversational mechanisms. The WUW module is only supposed to discern between the trigger word itself and any other kind of acoustic input, thus becoming a two-class hypothesis test, or verification step, that translates into a less computationally and resource demanding system than an always-awake S2T model.

Despite its simplicity with respect to a large vocabulary automatic speech recognizer, the WUW model still needs to be robust enough to handle acoustic distractions, such as TV, music, or overlapping speech. Noisy environments impact the WUW's performance both by waking-up unexpectedly, that translates into false alarm errors, and by not properly catching the trigger word, also known as miss errors. Those errors, especially false alarms, dramatically impact the user experience and reduce the user's expectations on the technology and his/her engagement with it. Therefore, there are common approaches employed for improving robustness in WUW detection. Some of them are based on a second-step verification, typically an automatic speech recognition (ASR) model or a WUW model [2–5]. Other works incorporate a Speech Enhancement (SE) module that employs a dedicated stage, at the audio input, aiming to reduce noise and to obtain an improved version of the acoustic signal. SE tackles the task of improving the perceptual intelligibility and quality of speech by usually removing background noises [6,7]. Although it is typically applied for a better perceptual experience in telecommunications [8] and hearing aids [9], SE has also reported improved results, e.g., as a pre-processing step into the context of ASR systems [10–12].

With respect to the WUW detection and keyword spotting tasks themselves, recent approaches have reported on the benefits of using the most advanced deep learning architectures. For instance, more recent works have introduced systems based on convolutional [13], recurrent, [14–16] and self-attention networks [17]. Dealing with robustness and generalization for previous architectures, we can find a widespread strategy based on synthesizing training data by noise augmentation. Noise data augmentation techniques exploit a deep neural network's appetite for vast amounts of data. They artificially corrupt the original samples, which usually translates into better performance figures, making the models to be more robust with regard to a bigger variety of noises or unseen scenarios. Similar approaches can be seen across different speech tasks, such as in keyword spotting [18], in ASR [19,20], or in WUW detection [21]. Therefore, we adopt similar ideas for our training data employed by all the classifiers described in this work. We artificially mix training samples with additive noise or by creating different kind of artifacts on the original speech, translating into similar findings on performance for our WUW task than those reported in previous works and for other speech tasks.

Classical SE methods, such as Wiener filtering [22], spectral subtraction [9], or subspace algorithms [23], specialize in characterizing noise, so it can be reduced from the speech signal. However, such methods do not provide a robust performance against certain contaminations, such as non-stationary noises or overlapped speech. This gap has been addressed in the last few years with deep learning approaches, with some of them acting at the spectral level [12,24] and others directly at the waveform input signal [25,26]. One widespread architecture is the encoder-decoder–autoencoder. It can be found in Reference [27], which additionally proposes the generative adversarial network paradigm [28] and makes use of skip connections in the style of U-Net [29]. Another popular model is proposed in Reference [30], which operates at the waveform level, using a similar architecture but including a LSTM [31] between the encoder and the decoder for hidden state sequence-to-sequence modeling. Nevertheless, many current approaches are commonly optimized by minimizing a regression loss in time, or by a combination with a spectrogram domain loss [24,30].

Motivated by the performance of such models, we propose to study the application and the effects of SE modules and techniques upon the performance of a WUW detection task, extending our previous work in such matter [1]. We hypothesize that cleaning noisy speech with a dedicated SE front-end should be beneficial for a WUW detector. Aiming to validate the previous assumption, we cover different experimental scenarios in this work:

(a) The isolated classifier: just a WUW classifier is available, which is our baseline with no SE.
(b) The independent SE and WUW models: both systems are trained separately, thus training the SE model exclusively on waveform and spectral regression losses.
(c) The Task-Aware SE (TASE) through frozen WUW model: WUW model is trained beforehand, so it is plugged after the SE model during SE training. This way, the WUW detection logits are available, so the classification loss can be back-propagated at the SE model and summed up to the regression losses. The WUW detector is frozen at SE training.
(d) The end-to-end TASE (TASE-E2E) and WUW model training: both systems are jointly trained at the same time from scratch, optimizing the SE model with joint regression and classification losses.

Wrapping up, one of the main novelties on the present paper is the study of SE applied to WUW detection, which has not been reported in previous works, to the best of our knowledge. Furthermore, we propose a new loss that makes the SE model task-aware, enhancing the speech in order to maximize the performance later on at the WUW detection stage. This is achieved not only by back-propagating the regression loss from the SE module but also by adding the loss of the classification task from the WUW classifier; see Figure 1. Aiming to generalize the results to several noise conditions, we train and test with different signal-to-noise (SNR) ratios, showing that the SE module is specially beneficial as noise increases. Furthermore, we also report the SE benefit obtained in different acoustic scenarios, such as a TV, office, or living room scenario, for instance.

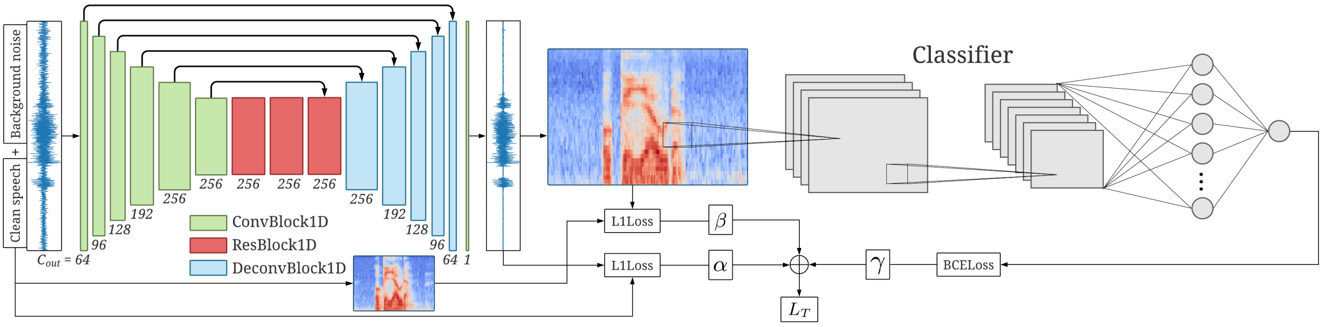


Figure 1. End-to-end TASE model at waveform level concatenated with a classifier. The log-mel spectrogram and waveform reconstruction losses of the SE model can be used together with the task-dependent loss (BCE loss) of the classifier acting as a Quality-Net [32,33] to train the model. The latter term aims at enhancing relevant speech features for the WUW detection task.

2. Task-Aware Speech Enhancement

Speech enhancement of the voice assistant's input is hypothesized to be beneficial for WUW detection. Firstly, removing background noise is supposed to lower the amount of false activations by reducing the variability in the input audio. Secondly, if enhancement is precisely done, speech is captured with higher intelligibility, thus making it easier to detect the trigger word. For the latter, the following sections report an exhaustive comparison of speech enhancement strategies. Furthermore, we introduce the TASE model, that optimizes the prediction loss of the subsequent WUW module, resulting in better figures than those from its task-agnostic counterpart. The resulting task-aware model not only enhances the speech signal but even removes the non-target speech or overlapped speech, that might be confusing for the trigger word detection. Finally, we make use of some of the most common noises found in domestic environments to simulate realistic acoustic conditions and train the different SE models at several SNR levels.

Model

Our model is a fully-convolutional denoising autoencoder with skip connections (Figure 1), in the style of previous effective SE models [27,30,34]. In training, we input a noisy audio waveform $\boldsymbol{x} \in \mathbb{R}^T$, comprised of clean speech signal $\boldsymbol{y} \in \mathbb{R}^T$ and background noise $\boldsymbol{n} \in \mathbb{R}^T$ so that $\boldsymbol{x} = \lambda \boldsymbol{y} + (1-\lambda)\boldsymbol{n}$, where λ is a parameter to control the SNR.

The encoder consists of six convolutional blocks (ConvBlock1D), each being a sequence of a convolutional layer, an instance normalization and a rectified linear unit (ReLU). Kernel size $K = 4$ and stride $S = 2$ are used, except in the first layer, where $K = 7$ and $S = 1$. After the encoder, the compressed signal goes through three intermediate residual blocks (ResBlock1D) that preserve the shape, each formed by two ConvBlock1D with $K = 3$ and $S = 1$. Skip connections are added from the input of each residual block to its output. The signal finally flows through the decoder, which follows the inverse structure of the encoder, where deconvolutional blocks (DeconvBlock1D) replace the convolutional layers of the ConvBlock1D with transposed convolutional layers. The output of the decoder is the enhanced signal with the shape of the input waveform, which is ready to be passed on to the WUW classifier. Both the encoder and decoder blocks are connected with skip connections to ensure that low-level details of the waveform are preserved.

The model is fully convolutional because this reduces the forward delay compared with the same architecture using a Recurrent Neural Network (RNN) to the latent representation of the audio. Table 1 presents a comparison between state-of-the-art architectures and ours, in terms of number of parameters, operations, size, and forward delay. The measurements of the forward time have been done in the same conditions: same CPU and using the same input data, an audio of 1.5 s. We performed 100 forwards and then calculated the average forward time. For the architecture named "gruse", we replaced the residual blocks that process the compressed signal of our architecture by a Gated Recurrent Unit (GRU) with a hidden size of 256. This produces a smaller model that carries out less operations, while considerably increasing the forward delay. Architectures demucs (H = 64 and H = 48) are from the work proposed in Reference [30], and NSNet2 is the baseline network used for the Deep Noise Suppression Challenge [35].

Table 1. Parameters, number of operations (multiplications and additions), size, and forward time of SE models.

Architecture	Parameters	# Operations	Size (MB)	Forward Time (ms)
demucs (H = 64)	33.53M	10,014.87M	278.61	163.21
demucs (H = 48)	18.86M	5644.95M	184.05	98.28
NSNet2	2.80M	-	-	22.00
TASE	2.45M	4156.26M	154.67	65.50
gruse	1.31M	1852.57M	64.11	176.42

Optimization is guided with a regression loss function (L1 loss) at raw waveform level, together with another L1 loss over the log-mel spectrogram, as proposed in Reference [36], to reconstruct the clean signal $\hat{\boldsymbol{y}}$ at the output. Finally, we include the binary cross-entropy classification loss (BCE loss) of the WUW classifier in the TASE use-case. We either train the TASE model jointly with the WUW classifier from scratch, or we just concatenate a frozen pre-trained classifier at its output. In any case, the BCE loss is available to TASE to optimize itself toward WUW detection. Our final loss function is defined as a linear combination of the three losses:

$$L_T = \alpha L_{raw}(\boldsymbol{y}, \hat{\boldsymbol{y}}) + \beta L_{spec}(S(\boldsymbol{y}), S(\hat{\boldsymbol{y}})) + \gamma L_{BCE}, \quad (1)$$

where α, β, and γ are hyperparameters weighting each loss term, and $S(\cdot)$ denotes the log-mel spectrogram of the signal, which is computed using 512 FFT bins, a window of 20 ms with 10 ms of shift, and 40 filters in the mel scale.

3. Materials and Methods

3.1. Databases

The databases used for our experiments contain speech with either the WUW, “OK Aura”, or without it. On the one hand, WUW samples are drawn from two Telefónica’s in-house datasets, one of them being made publicly available for research purposes under End-User License Agreement (EULA). On the other hand, samples without the WUW are taken from the in-house public dataset itself and the Spanish Common Voice (CV) corpus [37]. Background noise contamination is done to test the effectiveness of the speech enhancement models, and the acoustic events for doing so are sampled from further external datasets. More details about each dataset are given in the following subsections.

3.1.1. OK Aura Database

In-house data collection of WUW samples is done in two rounds. During the first round, ~4300 samples (2.8 h) from 360 speakers are collected, constituting the main bulk of positive WUW samples. Furthermore, office background ambient is recorded, as well, in order to obtain samples for posterior acoustic contamination.

The second round of data collection gathers 1247 utterances (1.4 h) from 80 speakers. It is designed with two purposes: (1) to address the main cases where Aura’s WUW classifiers typically under-perform and (2) to ask the participants to sign a consent form to make the data publicly available. Therefore, the dataset not only contains positive WUW samples but also other non-WUW samples that are phonetically similar to the WUW. Actually, sentences are scripted in different levels of similarity to the WUW:

1. The WUW itself: *OK Aura.*
2. The WUW within a context sentence: *Perfecto, voy a mirar qué dan hoy. OK Aura.*
3. Contains “Aura”: *Hay un aura de paz y tranquilidad.*
4. Contains “OK”: *OK, a ver qué ponen en la tele.*
5. Contains similar word units to “Aura”: *Hola Laura.*
6. Contains similar word units to “OK”: *Prefiero el hockey al baloncesto.*
7. Contains similar word units to “OK Aura”: *Porque Laura, ¿qué te pareció la película?*

Furthermore, knowing that WUW task performance depends on gender, age, and accent biases, plus other acoustic conditions, such as closeness to the microphone or room size, we also collect such metadata, as seen in Table 2.

Table 2. Metadata in the OK Aura Wake-up Word Dataset.

Metadata	Values
Age	20 s, 30 s, 40 s, 50 s, 60 s. . .
Gender	Female, Male, Non-binary
Distance	Close, Two steps away
Room size	Small (0–10 m^2), Medium (10–20 m^2)
Prosody	Unknown, Neutral, Annoyed, Friendly
Accent	Andalusian, Andean-pacific, Castilian, Non-native. . .

Data acquisition is done from a web-based form service called Jotform (https://form.jotform.com/201694606537056, accessed on 1 February 2022). We actually encourage readers to contribute to the dataset while the form is still open. Meanwhile, the current dataset version has been published as the “OK Aura Wake-up Word Dataset” [38], and it is publicly available (https://zenodo.org/record/5734340, accessed on 1 February 2022) under request to any of the authors via EULA.

3.1.2. External Data

Most of the non-WUW samples are drawn from the validated set of the Spanish CV corpus [37] (~300 h). However, we select a subset of 55 h for training, 7 h for development, and 7 h for testing. This way, we keep a ratio between negative and positive samples of 10:1, which showed good performance in Reference [39]. Regarding background noises, we pick samples from a variety of public datasets, such as Free Music Archive (https://freemusicarchive.org/, accessed on 1 February 2022), or Podcasts in Spanish (https://www.podcastsinspanish.org/, accessed on 1 February 2022), in order to cover different acoustic scenarios (living room, TV, music, etc.), as shown in Table 3.

Table 3. Background noise datasets.

Noise Type	Dataset
Living Room	QUT-NOISE (HOME-LIVINGB) [40]
TV	IberSpeech-RTVE Challenge [41]
Music	Free Music Archive
Conversations	Podcasts in Spanish
Office	In-house OK Aura WUW Dataset

3.1.3. Data Processing

The OK Aura Wake-up Word Dataset is comprised of monaural audio signals. They are stored in Waveform Audio File Format (WAV) by using a Pulse-Code Modulation (PCM) encoding with two bytes per sample at a rate of 16 kHz. All the external audio is also standardized to this format. The speech signal is processed with a Speech Activity Detection (SAD) module, producing timestamps where speech occurs and discarding fragments of inactivity. For this purpose, the tool from pyannote.audio [42] is used, which has been trained with the AMI corpus [43]. This is done to train only with valid speech segments from the collected audios.

The input to the speech enhancement module is the raw audio waveform. To train the model, the L1 regression loss is calculated at the log-mel spectrogram level and at the waveform level. In the case of concatenating the speech enhancement module with the WUW detector, it is the log-mel spectrogram obtained at the autoencoder output that is used as input for the WUW detector. The procedure for extracting the log-mel spectrogram ($S(\cdot)$) is detailed in Section 2.

Training, validation, and test partitions are split, ensuring that neither speaker nor background noise is repeated between partitions, maintaining an 80-10-10 ratio, respectively. The total data, containing internal and external datasets, consists of 50,737 non-WUW samples and 4651 WUW samples.

3.2. Data Augmentation

For the purpose of training the system with representative noise samples of realistic scenarios of the device use case, several Room Impulse Responses (RIR) are created based on the Image Source Method [44], for a room of dimensions (L_x, L_y, L_z), where $2 \leq L_x \leq 4.5, 2 \leq L_y \leq 5.5, 2.5 \leq L_z \leq 4$ m, with microphone and source randomly located at any (x, y) point within a height of $0.5 \leq z \leq 2$ m. Every TV and music original recordings are convolved with different RIRs to simulate the noise signal picked up by the microphone of the device in a given room.

Although we have tested several data augmentation techniques, we have found that background noise addition is the most significant with respect to performance. Thus, we keep it as the main data augmentation technique in this work. Clean speech samples are combined with different noise recordings (TV, music, conversations, and office and living room noise) within a wide range of SNRs ($[5, 30]$ or $[-10, 50]$ dB SNR). This aims at improving the performance of the models against noisy environments. In each epoch, we create different noisy samples by randomly selecting a sample of background noise for each speech event and combining them with a randomly chosen SNR in a specified range. Other data augmentation techniques, such as time stretching, pitch shifting, cropping, clipping distortion, and fading with different probabilities, are discarded since no significant improvements are found in initial tests for very noisy scenarios.

3.3. Wake-Up Word Detection Models

With the purpose of measuring the quality of the task-aware SE models, we report the impact of the SE module on WUW detection performance using several trigger word detection models. Typically, the end device that runs the WUW detector model has constrained capabilities; thus, the forward delay is a relevant parameter to consider while selecting the architecture of the audio classifier. Bigger models tend to perform better but may lead to an undesired delay in the detection, propagating this delay to the whole conversational chain and, consequently, degrading the user experience.

As baseline classifier, a `LeNet` [45] is used, a well-known convolutional neural network (CNN) composed of two convolution layers with ReLU activations and two pooling layers, followed by a final dense block consisting of two fully-connected layers.

Additionally, based on the work of Sainath and Parada [13], which consists of the exploration of lightweight CNNs for keyword spotting, both limiting the number of operations and the number of parameters, and the Tang and Lin's re-implementation in PyTorch [46], we use the `cnn-trad-pool2` architecture. This model consists of two convolutional layers, each one followed by pooling in time and frequency. Tang and Lin also worked with deep residual networks combined with dilated convolutions [47], obtaining comparable results with other CNN-based architectures and giving the possibility to vary the depth and width to achieve small footprint architectures. From this work we use `resnet15`, `resnet15-narrow`, and `resnet8`, which have 15, 15, and 8 ResNet blocks and 45, 19, and 45 feature maps, respectively.

To continue with, we also use two RNN-based models based on the open source tool named Mycroft Precise [48], which is a lightweight WUW detection tool implemented in TensorFlow. Named as `SGRU` and `SGRU2`, these are two bigger variations that we have implemented in PyTorch. The first one has a single GRU with a hidden size of 200, and the second one has two GRUs with a hidden size of 100.

Finally, we adapt an architecture from a kernel [49] in Kaggle's FAT 2019 competition [50], named as `CNN-FAT2019`, which has shown good performance in tasks, such as audio tagging or detection of gender, identity, and speech events from pulse signal [51]. This is the biggest architecture used, and it is conformed by eight convolutional layers with ReLU activations, with pooling layers every two convolutional layers.

In Table 4, we present the number of parameters, operations (multiplications and additions), and the size of every keyword detection architecture used. RNN-based networks are the smallest, and ResNet-based architectures show the variability of operations and parameters, depending on the depth and width.

Table 4. Parameters and number of operations of WUW detection models.

Classifier	Parameters	Operations (mult. and add.)	Size (MB)
lenet	4.7M	21M	19.2
cnn-trad-pool2	183k	42M	2.23
resnet15	237.4k	1433M	29.96
resnet15-narrow	42.4k	256M	12.44
resnet8	109k	57M	3.55
sgru	145.6k	144.4k	0.81
sgru2	103.4k	102.2k	0.53
cnn-fat2019	5.2M	1218M	41.9

3.4. Training

Speech utterances are segmented with a fixed window length of 1.5 s, which is typically enough to cover the average duration of the WUW, which is 1.0 s, based on the SAD timestamps. Speech is combined randomly with background noises, following the procedure explained in Section 3.2, with a given SNR range. The SE model is trained to cover a wide SNR range of $[-10, 50]$ dBs, whereas WUW models are trained to cover two scenarios: a classifier trained with the same SNR range as the SE model, and a classifier less aware of noise with a narrower SNR range of $[5, 30]$ dBs. This way, it is possible to study the impact of the SE model regarding whether or not the classifier has been trained with more or less noise.

We address data imbalance by balancing the classes in each batch using a weighted sampler. Besides, batching is done to ensure that negative samples from the OK Aura dataset are always present at each batch. This way, we increase the representation of negative samples which are phonetically similar to the WUW during training.

The loss in Equation (1) allows us to train the models in multiple ways, and we define different SE models and classifiers based on the loss function used:

(a) Isolated classifier: we remove the autoencoder from the architecture (Figure 1) and train any of the classifiers using the noisy audio as input: $\alpha = \beta = 0$ and $\gamma = 1$.
(b) Separate SE and classifier: we remove the classifier from the architecture and optimize the autoencoder based on the reconstruction losses only: $\alpha = \beta = 1$ and $\gamma = 0$.
(c) Task-aware SE (TASE): operations of a frozen pretrained classifier are only backpropagated to the SE model, which is optimized with the reconstruction losses altogether: $\alpha = \beta = \gamma = 1$.
(d) End-to-end TASE (TASE-E2E): autoencoder and classifier are trained jointly using the three losses: $\alpha = \beta = \gamma = 1$.

All the models are trained with early stopping based on the validation loss with 60 epochs of patience, for a maximum number of 200 epochs. Additionally, the learning rate decreases in an order of magnitude if there is not improvement in 20 consecutive epochs. We use the Adam optimizer starting with a learning rate of 10^{-4} in the E2E case and 10^{-3} for the rest, always using a batch size of 50.

3.5. Testing

The following test results are reported, such as for a binary classification task or hypothesis test, that is, by evaluating whether the WUW is contained within a single time window or not. For synthesizing the testing data, both the negative and positive samples are combined with a background noise, by summing it up with a specific SNR level to the original waveform; see Section 3.4 for further details.

Given the output probabilities from a model, the decision threshold is chosen as the one yielding the biggest difference between true and false positive rates, based on Youden's J statistic [52]. Once the threshold is decided, F1-score is computed to analyze and compare models. We compute such scores across all WUW classifiers described in Section 3.3 and for every SNR range.

Additionally and for the sake of comparison, we also report the objective metrics PESQ [53] and STOI [54] on the Valentini et al. [55] benchmark. This dataset is composed of both clean and noisy speech in English and uses a total of 15 different background noises (e.g., babble, metro, and restaurant). From the noisy test set, we randomly select two seconds of each audio clip that have been enhanced with SE models and then we measure PESQ and STOI.

4. Results

Figure 2 shows an example of speech enhanced spectrogram using the TASE architecture. The audio combines background music with the keyword between 1.15 and 1.95 s, and the enhanced audio completely preserves the speech relevant information for the posterior classification.

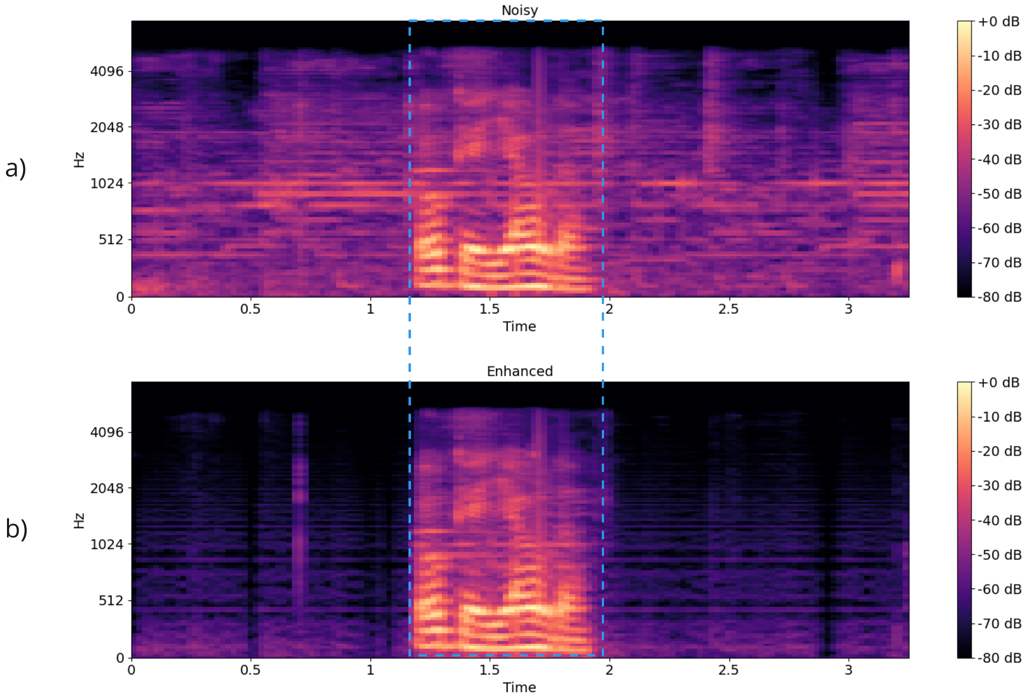
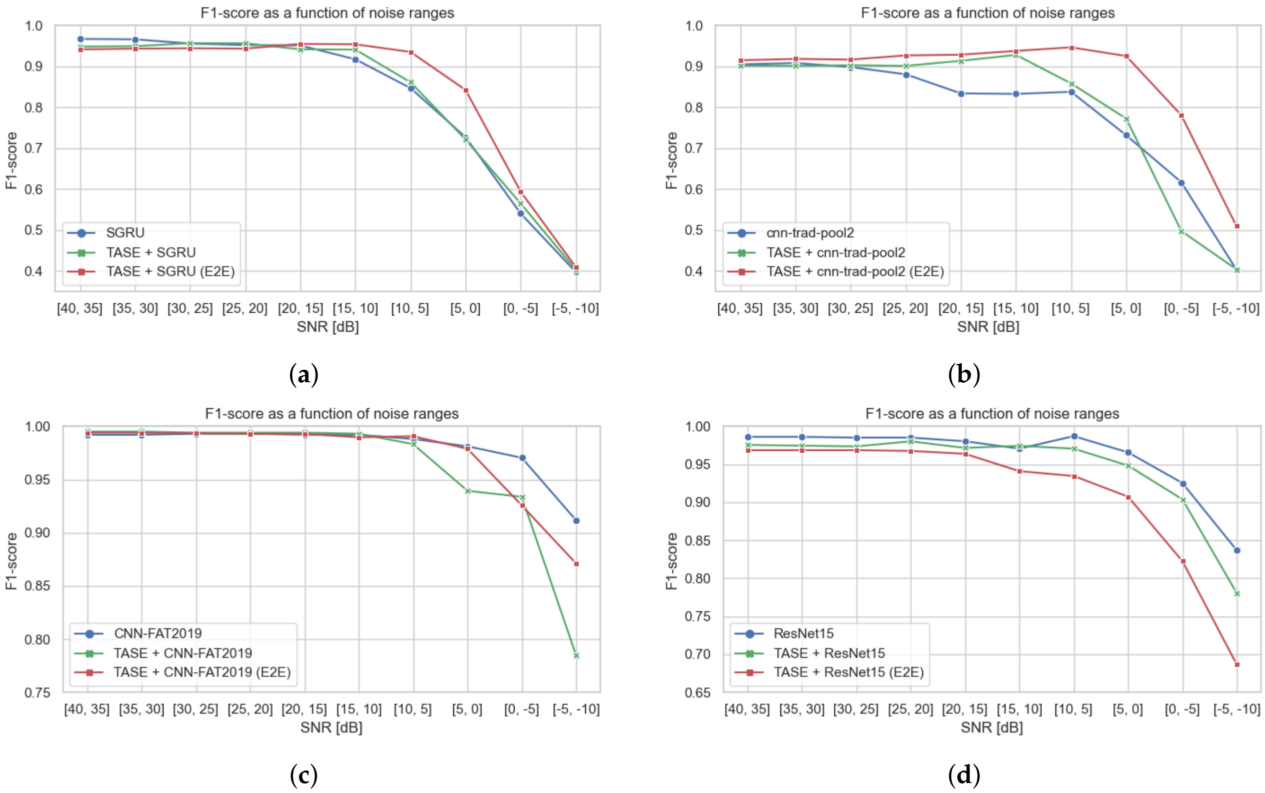


Figure 2. Example of Speech Enhancement spectrograms. Each figure shows (**a**) a noisy log-mel spectrogram and (**b**) an enhanced log-mel spectrogram. The blue rectangle shows where the "OK Aura" keyword is placed.

Figure 3a–d show the behavior of TASE when plugged to some of the different WUW classifiers described in Section 3.3. We find that TASE is notably beneficial to models, such as SGRU or cnn-trad-pool2, which present lowest robustness to noise, as compared to ResNet15 or CNN-FAT2019, where TASE yields equal or worse performance at some noise ranges. We hypothesize that ResNet15 and CNN-FAT2019 do not benefit of the speech enhancement as much, since they are bigger and more complex architectures that already handle the nuances of noise with more precision. However, an exhaustive fine-tuning of hyperparameters has not been done for every architecture, as we have prioritized covering more models, instead of deeply fine-tuning a few ones, due to computational restrictions. Therefore, we do not discard that our default hyperparameter choice might be biased toward certain architectures, yielding worse performances for cases, such as the TASE-E2E in ResNet15. Detailed metrics for SGRU and cnn-trad-pool2 can be found in Tables A1 and A2.

Furthermore, Figure 4 shows the improvement of the WUW detection in noisy scenarios by concatenating our TASE model with the remaining classifiers described in Section 3.3 that are neither large nor robust to noise (SGRU2, ResNet8, ResNet15-narrow), plus LeNet, which architecture has not been fine-tuned for audio tasks. Classifiers are trained with low noise ($[5, 30]$ dB SNR), to simulate a voice assistant system that has not been exposed to overwhelming amounts of noise at training. Applying SE in quiet scenarios maintains fairly good results and, especially, improves the models in lower SNR ranges.

Nonetheless, in the case when the classifiers are trained with a wider SNR range ($[-10, 50]$ dB SNR) by data augmentation, the performance gap between using TASE or not using is significantly reduced. F1-scores between both choices are on par for most of the SNR ranges. The noisiest range of $[-5, -10]$ dB SNR shows a small advantage for the model with respect to TASE, but it is not as large as the averaged improvement reported in Figure 4; see Figure 5.



(**a**)

(**b**)

(**c**)

(**d**)

Figure 3. WUW detection performance comparison for different models in terms of F1-score, with and without TASE. All models are trained in the range of $[-10, 50]$ dB SNR. TASE is not beneficial in noisy scenarios for large architectures (bottom row), while it does contribute positively to smaller models, especially when trained jointly end-to-end (upper row). (**a**) SGRU. (**b**) cnn-trad-pool2. (**c**) CNN-FAT2019. (**d**) ResNet15.

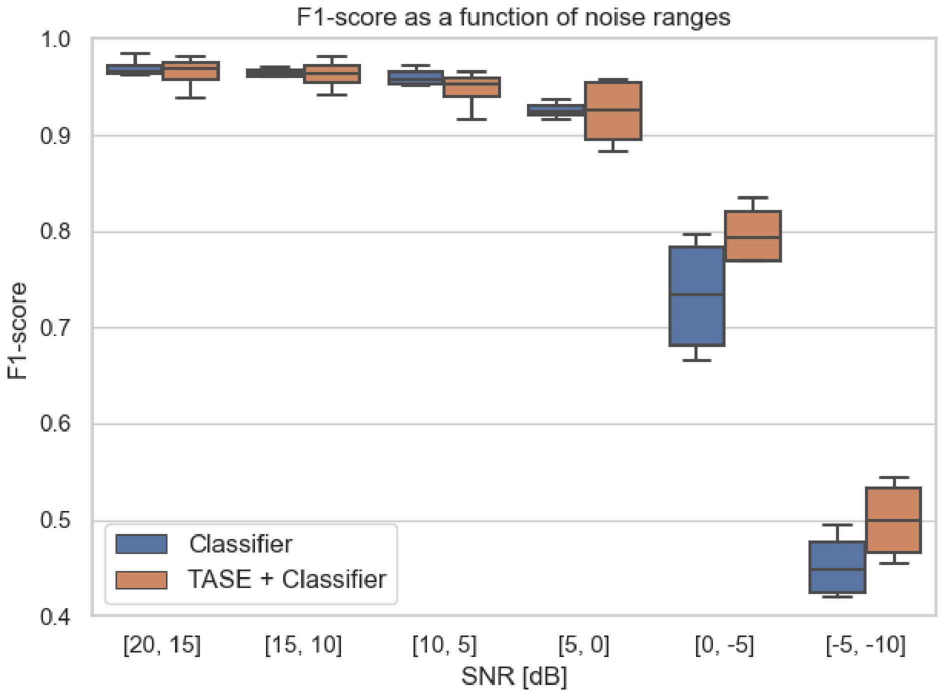


Figure 4. F1-score box plot for different SNR ranges. Classifiers trained with a limited range of noise ($[5, 30]$ dB SNR).

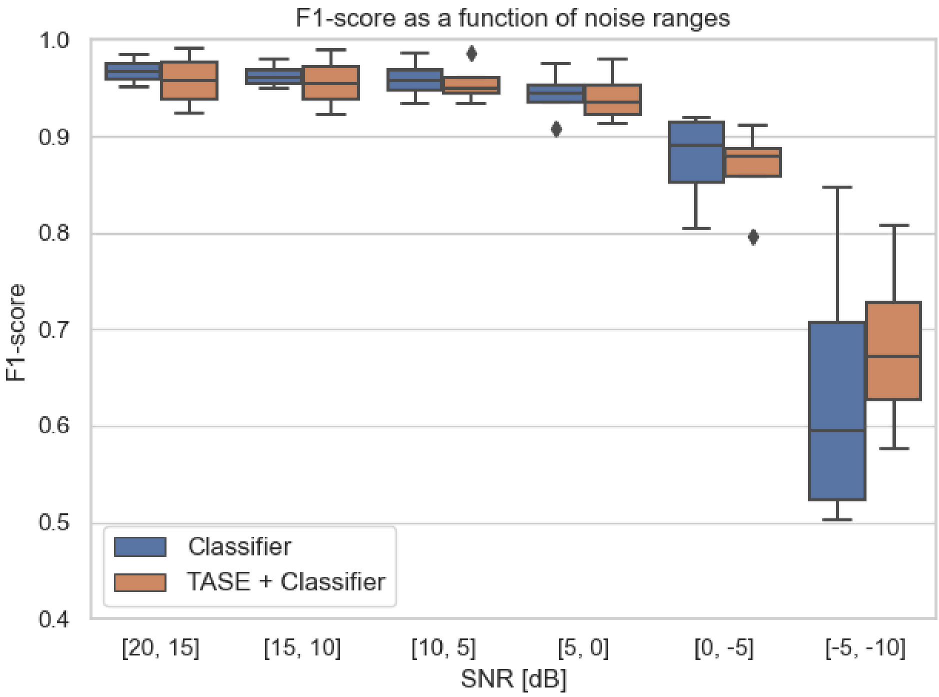


Figure 5. F1-score box plot for different SNR ranges. Classifiers trained with a very wide range of noise ($[-10, 50]$ dB SNR).

In Section 3.4, we have defined the parameters of the loss function (1) to train a classifier (see case (a) in Section 1 list), and three different approaches to train the SE model: standalone (b) or coupled with the classifier (c, d). In Figure 6, we analyze all the cases using a LeNet as WUW detector. We see how TASE-E2E performs better than all the other cases in almost every SNR range. From 40 dB to 10 dB of SNR, the results are very similar for the 4 models. In contrast, in the noisiest ranges the classifiers without SE model are the worst performers, followed by the separate SE case where only the waveform and spectral reconstruction losses are used. We find that the TASE case, which includes the classification loss in the training stage, improves the results for the WUW detection task. However, the best results are obtained with the TASE-E2E case, where the SE models and the classifiers are trained jointly using all three losses.

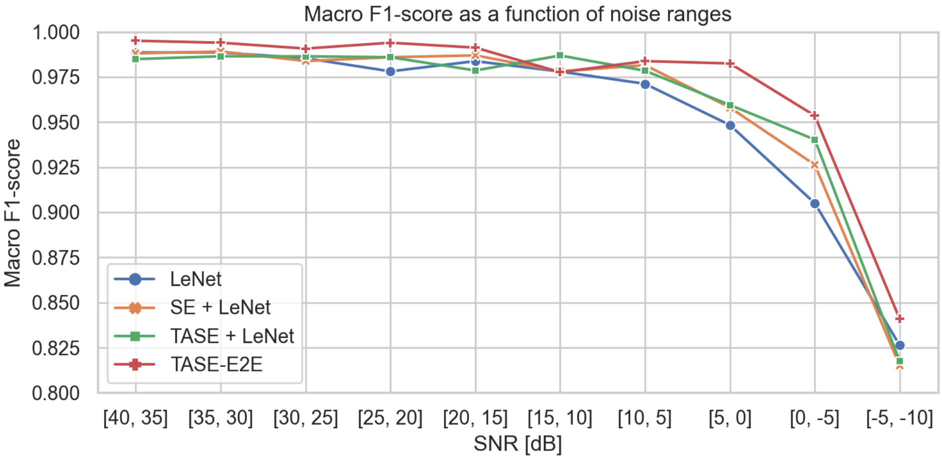


Figure 6. Comparison of different training methods for the SE models and LeNet classifier, in terms of the macro F1-Score for different SNR ranges. All models trained in the range of $[-10, 50]$ dB SNR.

We compare the WUW detection results of TASE-E2E with other state-of-the-art SE models (SEGAN [27] and Denoiser [30]), followed by a classifier (data augmented LeNet) in different noise scenarios. In Table 5, it can be observed that, when training the models together with the task loss, the results in our setup are better than with other, more powerful, but more general, SE models. We hypothesize that this is due to the natural adaptation of the SE to the classifier during the end-to-end training, as well as having been trained with a focus on common home noises. TASE-E2E improves the detection over the no SE model case, especially in scenarios with background conversations, loud office noise, or loud TV; see Table 6.

Table 7 shows that our SE system does not improve speech quality with respect to the case where we do not use any model to enhance speech. This was expected because our models have not been trained to remove generic background noises that the Valentini dataset contains. Instead, we train an SE system to learn to remove background conversations and TV noise that could trigger the device, which can lead to speech deterioration for the Valentini data. Nevertheless, we observe that PESQ improves in the case of TASE coupled with a LeNet classifier compared with SE, and the best results are obtained in the end-to-end case, where the PESQ and STOI results obtained without an SE module are preserved. This demonstrates that the consideration of the classification task in the loss improves the capacity of the SE model to clean speech.

Table 5. Macro F1-score enhancing the noisy audios with state-of-the-art SE models and using a LeNet as a classifier.

SNR [dB]		No SE	SEGAN	Denoiser	JointSE
$[20, 10]$	Clean	0.980	0.964	0.980	**0.990**
$[10, 0]$	Noisy	0.969	0.940	0.955	**0.972**
$[0, -10]$	Very noisy	0.869	0.798	0.851	**0.902**

Table 6. Macro F1-score percentage difference between JointSE and LeNet without SE, for different background noises.

SNR [dB]		Music	TV	Office	Living Room	Conversations
$[20, 10]$	Clean	1.0	-0.9	1.4	0.4	**2.3**
$[10, 0]$	Noisy	0.0	-1.2	0.8	0.4	**1.9**
$[0, -10]$	Very noisy	0.5	3.9	**11.2**	3.1	3.8

Table 7. Objective evaluation of speech quality.

Architecture	PESQ	STOI
None	2.02	0.93
SE	1.89	0.93
TASE	1.97	0.92
TASE-E2E	2.02	0.93

5. Conclusions

To the best of our knowledge, we have reported the first exploration of neural-based speech enhancement applied to wake-up word detection, and we validated its benefits with respect to classification performance. Furthermore, we proposed a way of making the SE module task-aware, by back-propagating the classification loss of the WUW model along the training. We call this task-aware speech enhancement (TASE), and it yields even further improvements than training SE and WUW modules separately. We show that TASE can be done by freezing the WUW module during SE training, or jointly training both from scratch, which we call end-to-end task-aware speech enhancement (TASE-E2E). The latter reports the best classification performance results of all the setups studied herein. Across all the experiments, we find that gains from SE are especially significant at noisier SNR ranges, between 10 and -10 dBs. We have also evaluated the effectiveness of such SE techniques when compared to a standalone WUW classifier that has been trained on a wide SNR range between 50 and −10 dBs. In that case, the results between applying TASE or not using it are on par, with TASE having a slight advantage in the most severely noisy setups, that is, between −5 and −10 dBs SNR. Thus, we have reported the potential of TASE at improving the performance of standard neural net classifiers that are not specifically trained to be resilient to noise, and we encourage further experiments in the comparison between speech enhancement and noise data augmentation techniques. Finally, as we have corroborated our hypotheses with a manually segmented corpus, we motivate further work for the online streaming case, with the aim to explore the particularities and challenges that may arise in such a setup.

Author Contributions: Conceptualization, J.L.; methodology, D.B., J.L., C.S., F.L. and G.C.; software, F.L., D.B. and G.C.; validation, F.L., D.B. and G.C.; formal analysis, D.B., J.L., F.L. and G.C.; investigation, D.B., J.L., F.L. and G.C.; resources, J.L., M.F. and P.G.; data curation, G.C., J.L., M.F., D.B. and F.L.; writing—original draft preparation, G.C., F.L., D.B. and J.L.; writing—review and editing, G.C., F.L., D.B., J.L., M.F., C.S. and P.G.; visualization, F.L., D.B., G.C. and J.L.; supervision, J.L.; project administration, J.L. and P.G.; funding acquisition, J.L., P.G. and M.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by both the INGENIOUS and ACCORDION projects within the European Union's Horizon 2020 Research and Innovation Program under grant numbers 833435 and 871793, respectively. The sixth author has been funded by the Agencia Estatal de Investigación (AEI), Ministerio de Ciencia, Innovación y Universidades and the Fondo Social Europeo (FSE) grant number RYC-2015-17239 (AEI/FSE, UE).

Institutional Review Board Statement: Ethical review and approval were waived for this study since the data collection has strictly followed the GDPR rules, having all participants signed a Data Protection Information notice and Consent Form.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: OK Aura data are available upon request through https://zenodo.org/record/5734340, accessed on 21 December 2021.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ASR	Automatic Speech Recognition
BCE	Binary Cross-Entropy
CNN	Convolutional Neural Network
CV	Common Voice
FAT	Freesound Audio Tagging
GRU	Gated Recurrent Unit
LSTM	Long Short-Term Memory
PESQ	Perceptual Evaluation of Speech Quality
ReLU	Rectified Linear Unit
ResNet	Residual Network
RIR	Room Impulse Responses
RNN	Recurrent Neural Network
S2T	Speech-to-Text
SAD	Speech Activity Detection
SE	Speech Enhancement
SEGAN	Speech Enhancement Generative Adversarial Network
SNR	Signal-to-Noise Ratio
STOI	Short-Time Objective Intelligibility
TASE	Task-Aware Speech Enhancement
TASE-E2E	End-to-end Task-Aware Speech Enhancement
WUW	Wake-up Word

Appendix A

Table A1. SGRU model experiments. F1-score, precision, recall, and AUC from the ROC curve.

SNR [dB]	SGRU				TASE + SGRU				TASE + SGRU (E2E)			
	F1	Pr.	Re.	AUC	F1	Pr.	Re.	AUC	F1	Pr.	Re.	AUC
$[40, 45]$	0.969	0.963	0.976	**0.999**	0.947	0.919	0.976	**0.999**	0.941	0.900	0.986	0.998
$[35, 40]$	0.967	0.959	0.976	**0.999**	0.949	0.921	0.978	**0.999**	0.942	0.901	0.986	0.998
$[30, 35]$	0.966	0.957	0.976	**0.999**	0.950	0.923	0.978	**0.999**	0.944	0.905	0.986	0.998
$[25, 30]$	0.956	0.935	0.978	**0.999**	0.957	0.938	0.976	**0.999**	0.944	0.906	0.986	0.998
$[20, 25]$	0.952	0.928	0.978	**0.999**	0.957	0.942	0.972	0.998	0.944	0.905	0.986	0.998
$[15, 20]$	0.951	0.926	0.978	**0.998**	0.942	0.914	0.972	0.997	0.955	0.930	0.982	**0.998**
$[10, 15]$	0.918	0.872	0.968	**0.997**	0.941	0.923	0.960	0.996	0.954	0.931	0.978	**0.997**
$[5, 10]$	0.847	0.762	0.954	0.991	0.862	0.787	0.952	0.992	0.936	0.914	0.958	**0.996**
$[0, 5]$	0.727	0.608	0.904	0.970	0.722	0.597	0.912	0.974	0.843	0.780	0.916	**0.979**
$[-5, 0]$	0.541	0.401	0.830	**0.922**	0.565	0.449	0.762	0.907	0.594	0.467	0.816	0.919
$[-10, -5]$	0.398	0.287	0.648	**0.812**	0.404	0.307	0.588	0.796	0.410	0.320	0.570	0.775

Table A2. cnn-trad-pool2 (CNN-TP2) experiments. F1-score, precision, recall, and AUC from the ROC curve.

SNR [dB]	CNN-TP2				TASE + CNN-TP2				TASE + CNN-TP2 (E2E)			
	F1	Pr.	Re.	AUC	F1	Pr.	Re.	AUC	F1	Pr.	Re.	AUC
$[40, 45]$	0.906	0.841	0.982	**0.999**	0.916	0.858	0.982	**0.999**	0.916	0.851	0.992	**0.999**
$[35, 40]$	0.905	0.839	0.982	**0.999**	0.902	0.831	0.986	**0.999**	0.915	0.849	0.992	**0.999**
$[30, 35]$	0.908	0.845	0.982	**0.999**	0.901	0.830	0.986	**0.999**	0.919	0.855	0.992	**0.999**
$[25, 30]$	0.899	0.827	0.984	**0.999**	0.903	0.832	0.988	**0.999**	0.917	0.852	0.992	**0.999**
$[20, 25]$	0.881	0.797	0.984	**0.999**	0.901	0.829	0.988	**0.999**	0.927	0.871	0.990	**0.999**
$[15, 20]$	0.834	0.720	0.992	0.998	0.914	0.851	0.986	0.998	0.929	0.875	0.990	**0.999**
$[10, 15]$	0.833	0.723	0.982	0.998	0.928	0.890	0.970	0.998	0.938	0.896	0.984	**0.999**
$[5, 10]$	0.838	0.736	0.974	0.996	0.859	0.781	0.954	0.995	0.947	0.917	0.978	**0.998**
$[0, 5]$	0.732	0.598	0.942	0.984	0.773	0.667	0.918	0.981	0.925	0.897	0.956	**0.995**
$[-5, 0]$	0.617	0.513	0.774	0.924	0.498	0.353	0.846	0.917	0.781	0.697	0.888	**0.972**
$[-10, -5]$	0.403	0.313	0.566	0.776	0.404	0.317	0.558	0.770	0.510	0.402	0.698	**0.853**

References

1. Bonet, D.; Cámbara, G.; López, F.; Gómez, P.; Segura, C.; Luque, J.; Farrús, M. Speech Enhancement for Wake-Up-Word detection in Voice Assistants. In Proceedings of the IberSPEECH 2021, Valladolid, Spain, 24–25 March 2021; pp. 41–45. [CrossRef]
2. Ge, F.; Yan, Y. Deep neural network based wake-up-word speech recognition with two-stage detection. In Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; pp. 2761–2765.
3. Kumar, R.; Rodehorst, M.; Wang, J.; Gu, J.; Kulis, B. Building a Robust Word-Level Wakeword Verification Network. In Proceedings of the INTERSPEECH, Shanghai, China, 25–29 October 2020; pp. 1972–1976.
4. Michaely, A.H.; Zhang, X.; Simko, G.; Parada, C.; Aleksic, P. Keyword spotting for Google assistant using contextual speech recognition. In Proceedings of the 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Okinawa, Japan, 16–20 December 2017; pp. 272–278.
5. Hey Siri: An On-device DNN-Powered Voice Trigger for Apple's Personal Assistant. 2017. Available online: https://machinelearning.apple.com/research/hey-siri (accessed on 29 December 2021).
6. Loizou, P.C. *Speech Enhancement: Theory and Practice*; CRC Press: Boca Raton, FL, USA, 2013.
7. Xu, Y.; Du, J.; Dai, L.R.; Lee, C.H. An experimental study on speech enhancement based on deep neural networks. *IEEE Signal Process. Lett.* **2013**, *21*, 65–68. [CrossRef]
8. Reddy, C.K.; Beyrami, E.; Pool, J.; Cutler, R.; Srinivasan, S.; Gehrke, J. A scalable noisy speech dataset and online subjective test framework. *arXiv* **2019**, arXiv:1909.08050.
9. Yang, L.P.; Fu, Q.J. Spectral subtraction-based speech enhancement for cochlear implant patients in background noise. *J. Acoust. Soc. Am.* **2005**, *117*, 1001–1004. [CrossRef] [PubMed]

10. Zorilă, C.; Boeddeker, C.; Doddipatla, R.; Haeb-Umbach, R. An investigation into the effectiveness of enhancement in ASR training and test for chime-5 dinner party transcription. In Proceedings of the 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Singapore, 14–18 December 2019; pp. 47–53.
11. Maas, A.L.; Le, Q.V.; O'Neil, T.M.; Vinyals, O.; Nguyen, P.; Ng, A.Y. Recurrent Neural Networks for Noise Reduction in Robust ASR. In Proceedings of the Thirteenth Annual Conference of the International Speech Communication Association, Portland, OR, USA, 9–13 September 2012.
12. Weninger, F.; Erdogan, H.; Watanabe, S.; Vincent, E.; Le Roux, J.; Hershey, J.R.; Schuller, B. Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR. In *International Conference on Latent Variable Analysis and Signal Separation*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 91–99.
13. Sainath, T.N.; Parada, C. Convolutional neural networks for small-footprint keyword spotting. In Proceedings of the Sixteenth Annual Conference of the International Speech Communication Association, Dresden, Germany, 6–10 September 2015.
14. Kumar, R.; Yeruva, V.; Ganapathy, S. On Convolutional LSTM Modeling for Joint Wake-Word Detection and Text Dependent Speaker Verification. In Proceedings of the Interspeech, Hyderabad, India, 2–6 September 2018; pp. 1121–1125.
15. Arik, S.O.; Kliegl, M.; Child, R.; Hestness, J.; Gibiansky, A.; Fougner, C.; Prenger, R.; Coates, A. Convolutional recurrent neural networks for small-footprint keyword spotting. *arXiv* **2017**, arXiv:1703.05390.
16. Yamamoto, T.; Nishimura, R.; Misaki, M.; Kitaoka, N. Small-Footprint Magic Word Detection Method Using Convolutional LSTM Neural Network. In Proceedings of the INTERSPEECH, Graz, Austria, 15–19 September 2019; pp. 2035–2039.
17. Shan, C.; Zhang, J.; Wang, Y.; Xie, L. Attention-based end-to-end models for small-footprint keyword spotting. *arXiv* **2018**, arXiv:1803.10916.
18. Raju, A.; Panchapagesan, S.; Liu, X.; Mandal, A.; Strom, N. Data augmentation for robust keyword spotting under playback interference. *arXiv* **2018**, arXiv:1808.00563.
19. Hsiao, R.; Ma, J.; Hartmann, W.; Karafiát, M.; Grézl, F.; Burget, L.; Szöke, I.; Černockỳ, J.H.; Watanabe, S.; Chen, Z.; et al. Robust speech recognition in unknown reverberant and noisy conditions. In Proceedings of the 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), Scottsdale, AZ, USA, 13–17 December 2015; pp. 533–538.
20. Hannun, A.; Case, C.; Casper, J.; Catanzaro, B.; Diamos, G.; Elsen, E.; Prenger, R.; Satheesh, S.; Sengupta, S.; Coates, A.; et al. Deep speech: Scaling up end-to-end speech recognition. *arXiv* **2014**, arXiv:1412.5567.
21. Yoon, K.M.; Kim, W. Small-Footprint Wake Up Word Recognition in Noisy Environments Employing Competing-Words-Based Feature. *Electronics* **2020**, *9*, 2202. [CrossRef]
22. Meyer, J.; Simmer, K.U. Multi-channel speech enhancement in a car environment using Wiener filtering and spectral subtraction. In Proceedings of the 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing, Munich, Germany, 21–24 April 1997; Volume 2, pp. 1167–1170.
23. Ephraim, Y.; Van Trees, H.L. A signal subspace approach for speech enhancement. *IEEE Trans. Speech Audio Process.* **1995**, *3*, 251–266. [CrossRef]
24. Park, S.R.; Lee, J. A fully convolutional neural network for speech enhancement. *arXiv* **2016**, arXiv:1609.07132.
25. Rethage, D.; Pons, J.; Serra, X. A wavenet for speech denoising. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 5069–5073.
26. Phan, H.; McLoughlin, I.V.; Pham, L.; Chén, O.Y.; Koch, P.; De Vos, M.; Mertins, A. Improving GANs for speech enhancement. *arXiv* **2020**, arXiv:2001.05532.
27. Pascual, S.; Bonafonte, A.; Serra, J. SEGAN: Speech enhancement generative adversarial network. *arXiv* **2017**, arXiv:1703.09452.
28. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. *Adv. Neural Inf. Process. Syst.* **2014**, *27*, 2672–2680.
29. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 234–241.
30. Défossez, A.; Synnaeve, G.; Adi, Y. Real time speech enhancement in the waveform domain. *arXiv* **2020**, arXiv:2006.12847.
31. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
32. Fu, S.W.; Tsao, Y.; Hwang, H.T.; Wang, H.M. Quality-Net: An end-to-end non-intrusive speech quality assessment model based on BLSTM. *arXiv* **2018**, arXiv:1808.05344.
33. Fu, S.W.; Liao, C.F.; Tsao, Y. Learning with learned loss function: Speech enhancement with quality-net to improve perceptual evaluation of speech quality. *IEEE Signal Process. Lett.* **2019**, *27*, 26–30. [CrossRef]
34. Llombart, J.; Ribas, D.; Miguel, A.; Vicente, L.; Ortega, A.; Lleida, E. Progressive loss functions for speech enhancement with deep neural networks. *EURASIP J. Audio Speech Music. Process.* **2021**, *2021*, 1–16. [CrossRef]
35. Braun, S.; Tashev, I. Data augmentation and loss normalization for deep noise suppression. In *International Conference on Speech and Computer*; Springer: Cham, Switzerland, 2020.
36. Yamamoto, R.; Song, E.; Kim, J.M. Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 6199–6203.
37. Ardila, R.; Branson, M.; Davis, K.; Henretty, M.; Kohler, M.; Meyer, J.; Morais, R.; Saunders, L.; Tyers, F.M.; Weber, G. Common Voice: A massively-multilingual speech corpus. *arXiv* **2019**, arXiv:1912.06670.

38. Cámbara, G.; Luque, J.; Bonet, D.; López, F.; Farrús, M.; Gómez, P.; Segura, C. OK Aura Wake-up Word Dataset. *Zenodo* **2021**. [CrossRef]
39. Hou, J.; Shi, Y.; Ostendorf, M.; Hwang, M.Y.; Xie, L. Mining effective negative training samples for keyword spotting. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 7444–7448.
40. Dean, D.B.; Sridharan, S.; Vogt, R.J.; Mason, M.W. The QUT-NOISE-TIMIT corpus for the evaluation of voice activity detection algorithms. In Proceedings of the 11th Annual Conference of the International Speech Communication Association, Chiba, Japan, 26–30 September 2010.
41. Lleida, E.; Ortega, A.; Miguel, A.; Bazán-Gil, V.; Pérez, C.; Gómez, M.; de Prada, A. Albayzin 2018 evaluation: The IberSpeech-RTVE challenge on speech technologies for Spanish broadcast media. *Appl. Sci.* **2019**, *9*, 5412. [CrossRef]
42. Bredin, H.; Yin, R.; Coria, J.M.; Gelly, G.; Korshunov, P.; Lavechin, M.; Fustes, D.; Titeux, H.; Bouaziz, W.; Gill, M.P. Pyannote.audio: Neural building blocks for speaker diarization. In Proceedings of the ICASSP 2020, IEEE International Conference on Acoustics, Speech, and Signal Processing, Barcelona, Spain, 4–8 May 2020.
43. Carletta, J. Unleashing the killer corpus: Experiences in creating the multi-everything AMI Meeting Corpus. *Lang. Resour. Eval.* **2007**, *41*, 181–190. [CrossRef]
44. Allen, J.B.; Berkley, D.A. Image method for efficiently simulating small-room acoustics. *J. Acoust. Soc. Am.* **1979**, *65*, 943–950. [CrossRef]
45. LeCun, Y. LeNet-5, Convolutional Neural Networks. 2015; Volume 20, p. 14. Available online: http://yann.lecun.com/exdb/lenet (accessed on 1 February 2022).
46. Tang, R.; Lin, J. Honk: A PyTorch Reimplementation of Convolutional Neural Networks for Keyword Spotting. *arXiv* **2017**, arXiv:1710.06554.
47. Tang, R.; Lin, J. Deep Residual Learning for Small-Footprint Keyword Spotting. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018.
48. Scholefield, M.D. Mycroft Precise. 2019. Available online: https://github.com/MycroftAI/mycroft-precise (accessed on 1 February 2022).
49. mhiro2. Freesound Audio Tagging 2019: Simple 2D-CNN Classifier with PyTorch. 2019. Available online: https://www.kaggle.com/mhiro2/simple-2d-cnn-classifier-with-pytorch/ (accessed on 1 February 2022).
50. Fonseca, E.; Plakal, M.; Font, F.; Ellis, D.P.; Serra, X. Audio tagging with noisy labels and minimal supervision. *arXiv* **2019**, arXiv:1906.02975.
51. Cámbara, G.; Luque, J.; Farrús, M. Detection of speech events and speaker characteristics through photo-plethysmographic signal neural processing. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 7564–7568.
52. Youden, W.J. Index for rating diagnostic tests. *Cancer* **1950**, *3*, 32–35. [CrossRef]
53. Rix, A.W.; Beerends, J.G.; Hollier, M.P.; Hekstra, A.P. Perceptual evaluation of speech quality (PESQ)—A new method for speech quality assessment of telephone networks and codecs. In Proceedings of the 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing, Proceedings (Cat. No. 01CH37221), Salt Lake City, UT, USA, 7–11 May 2001; Volume 2, pp. 749–752.
54. Taal, C.H.; Hendriks, R.C.; Heusdens, R.; Jensen, J. A short-time objective intelligibility measure for time-frequency weighted noisy speech. In Proceedings of the 2010 IEEE international conference on acoustics, speech and signal processing, Dallas, TX, USA, 14–19 March 2010; pp. 4214–4217.
55. Valentini Botinhao, C.; Wang, X.; Takaki, S.; Yamagishi, J. Speech Enhancement for a Noise-Robust Text-to-Speech Synthesis System using Deep Recurrent Neural Networks. In Proceedings of the 7th Annual Conference of the International Speech Communication Association, San Francisco, CA, USA, 8–12 September 2016; pp. 352–356.