Neurocomputing 494 (2022) 116–131

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Survey paper

A comprehensive empirical review of modern voice activity detection approaches for movies and TV shows

Mayank Sharma [a,*], Sandeep Joshi [b,1], Tamojit Chatterjee [a,1], Raffay Hamid [a]

[a] Amazon Prime Video
[b] Kognitos, Inc.

ARTICLE INFO

Article history:
Received 9 June 2021
Revised 14 April 2022
Accepted 17 April 2022
Available online 21 April 2022
Communicated by Zidong Wang

Keywords:
Voice Activity Detection
Digital Entertainment Content
Convolutional Neural Network
Transformer Network
Recurrent Network
WebRTC VAD
Speech Processing
Input Noise
Label Noise

ABSTRACT

A robust and language agnostic Voice Activity Detection (VAD) is crucial for Digital Entertainment Content (DEC). Primary examples of DEC include movies and TV series. Some ways in which VAD systems are used for DEC creation include augmenting subtitle creation, subtitle drift detection and correction, and audio diarisation. Majority of the previous work on VAD focuses on scenarios that: (a) have minimal background noise, and (b) where the audio content is delivered in English language. However, movies and TV shows can: (a) have substantial amounts of non-voice background signal (*e.g.* musical score and environmental sounds), and (b) are released worldwide in a variety of languages. This makes most of the previous standard VAD approaches not readily applicable for DEC related applications. Furthermore, there does not exist a comprehensive analysis of Deep Neural Network's (DNN) performance for the task of VAD applied to DEC. In this work, we present a thorough survey on DNN based VADs on DEC data in terms of their accuracy, Area Under Curve (AUC), noise sensitivity, and language agnostic behaviour. For our analysis we use 1100 proprietary DEC videos spanning 450 h of content in 9 languages and 5 + genres, making our study the largest of its kind ever published. The key findings of our analysis are: (a) even high quality timed-text or subtitle [2] files contain significant levels of label-noise (up to 15%). Despite high label noise, deep networks are robust and are able to retain high AUCs ($\sim$ 0.94). (b) Using larger labelled dataset can substantially increase neural VAD model's True Positive Rate (TPR) with up to 1.3% and 18% relative improvement over current state-of-the-art methods in Hebbar et al. (2019) and Chaudhuri et al. (2018) respectively. This effect is more pronounced in noisy environments such as music and environmental sounds. This insight is particularly instructive while prioritizing domain specific labelled data acquisition versus exploring model structure and complexity. (c) Currently available sequence based neural models show similar levels of competence in terms of their language agnostic behaviour for VAD at high Signal-to-Noise Ratios (SNRs) and for clean speech, (d) Deep models exhibit varied performance across different SNRs with CLDNN (Zazo et al., 2016) being the most robust, and (e) models with comparatively larger number of parameters ($\sim$2 M) are less robust to input noise as opposed to models having smaller number of parameters ($\sim$ 0.5 M).

1. Introduction

An automated system that detects human speech or voice activity within an audio segment has multiple uses in digital entertainment domain. Such a system can be deployed to: a) aid subtitle creation by automating the identification of time boundaries associated with dialogues [4], b) improve subtitle quality by detecting sync issues between audio and subtitles [3], and c) improve accessibility of content by identifying segments of audio that do not have subtitle coverage, or by identifying the non-voice segments that may need an audio description. Such a Voice Activity Detection (VAD) system can be further enhanced to aid caption [4] and subtitle creation by detecting background noises [5–7], music and singing voice detection [8,9], speaker differentiation [10–12], emotion recognition [13–18], clean speech identification [19–22], and identifying

* Corresponding author.
E-mail addresses: mysharm@amazon.com (M. Sharma), raffay@amazon.com (R. Hamid).

[1] Work done while at Amazon.
[2] subtitles and timed-text are used interchangeably in the manuscript
[3] known as subtitle drifts.
[4] plot pertinent description of sounds.

https://doi.org/10.1016/j.neucom.2022.04.084
0925-2312/

high density audio segments. Traditionally VAD systems are also foundational for speaker recognition and ASR systems.

A generic VAD for DEC is a hard problem for multiple reasons. First, DEC voice segments typically coexist with various noises like embedded foreground or background music, title track, and multiple contextual background noises like traffic noises, gun shots, crowd buzz, door closing, motors, air conditioning, etc. Second, being an artistic medium, presence of atypical speech patterns like whispering, shouting, singing and electronic voices are more prevalent in DEC than other conversational speech problems. Third, and most importantly, majority of previous work in VAD is focused on English language. Any useful VAD system for DEC needs to scale for multiple languages, locales and genres. This makes most of the previous standard VAD approaches not readily applicable for DEC related applications. Furthermore, there does not exist a comprehensive analysis of DNN performance for the task of VAD when applied to DEC at scale.

In this paper, we provide a comprehensive empirical analysis of various deep neural models [23,24] for the task of VAD when applied to DEC. We use timed-text specification guidelines to create a training dataset from 1100 proprietary DEC videos spanning 9 languages and 5 + genres. This dataset consists of ∼450 h of content, making our study the largest scale analysis of this problem ever published. We divide the videos into 800 ms non-overlapping clips and label them into speech and non-speech audio based on their corresponding timed-text information[5].

On this dataset, we train several deep neural VAD models such as, the Gated Recurrent Unit (GRU) [25,26], the Temporal Convolution Network (TCN) [27], the Convolutional and Self Attention (STNET) [28] transformer encoder based network [29] and the VGG net based Time Distributed CNN (CNNTD) [1] respectively using spectrogram based features. We also train raw audio waveform based CLDNN [3]. Similarly, we train shallow neural models using Mel Frequency Cepstrum Coefficients (MFCCs) [30], Spectral Contrast [31] and Fluctograms [32] as features.

We test the performance of the models on several public and proprietary datasets. First, we analyze the AUCs, precisions, recalls and F-scores on a proprietary human annotated DEC test set. Second, we compare the TPRs on the publicly available AVA [2] dataset. Third, we examine the language agnostic characteristics on the publicly available MUSAN-librivox [33] dataset and a proprietary DEC language test set respectively and finally, we evaluate the input noise sensitivities of various models. The key findings of our work are summarized below:

1. GRU based models results in 1.3% relative improvement in TPR over current state-of-the-art CNNTD model [1] on two of three subsets of the publicly available AVA [2] dataset.
2. GRU, CNN and Self-Attention based models utilizing the sequence information in spectrograms and audio waveforms outperform the fully connected feed-froward neural models using MFCCs, Fluctogram and Spectral Contrast features. They show 8.8% relative improvement in AUC on our proprietary human annotated DEC test set.
3. Sequence based neural models result in similar levels of competence in terms of their language agnostic behaviour for clean speech and speech with high SNRs.
4. SNR acts as differentiator between sequence based deep learning models. CNNTD outperforms other models in terms of recall in noisy environments at high SNRs (10 and 5). However, CLDNN which is an amalgamation of CNN and Long-short term memory (LSTM) architecture, is the most robust model for medium SNRs (0 and −5) as it learns noise independent features from raw audio waveform, as outlined in [3]. Practically, such low SNRs are not observed during dialogues in DEC as they hamper the listening experience.
5. CNN based architectures with large number of parameters such as STNET and TCN are least robust to perturbations in the input.

This paper is divided into the following sections. Section 2 presents a brief overview of the previous efforts for VAD, both in statistical and neural domains. Section 3 presents various datasets and feature representations used and outlines various neural models for VAD. Section 4 presents the training pipeline, comprehensive empirical evaluation of various neural models on human validated datasets (in-house as well as public), multi-lingual (for languages inside and outside training sets) and speech mixed with noisy datasets. Finally, Section 5 presents the conclusion and future directions.

2. Related Works

The development of VAD systems has been an active area of research for several decades. An early example of a frame-based VAD system involved two Gaussian Mixture Models (GMMs), one trained on speech frames and the other on non-speech frames, to predict the per-frame likelihood of speech, followed by a Hidden Markov Model (HMM) that penalizes transitions between speech and non-speech states to give temporal continuity to the prediction [34]. WebRTC VAD [35] is an example of GMM based VAD model with input features as log energies of six frequency bands between 80 Hz and 4000 Hz. It uses fixed point operations and is optimized for real-time use in web transmission. Other early VAD approaches includes energy threshold methods [36,37], methods based on zero crossing rates [38], auto-regressive models [39], formant based methods [40], maximum margin based unsupervised VAD [41], combination of multiple acoustic models via likelihood ratio weighting [42], and higher order statistics of speech such as kurtosis [43]. A better approach is to learn a classifier using time and frequency domain audio features such as mean, variance, higher order moments, coefficient of variation, percentiles of the probability distribution of the audio signal and Mel-frequency cepstral coefficients (MFCCs) [44] as input [45,46] to a GMM, maxent, Support Vector Machines (SVM) [47] Random Forest (RF), neural networks [48–52], deep belief networks [53] and Conditional Random Field (CRF) [54,55]. In a similar approach, Benatan et al. [56] used cross covariance scores of MFCCs and trained a RF as VAD for DEC and presented a large jump in Equal Error Rate (EER) on 4 movies.

These approaches suffer from two major problems. First, models involving HMM cannot learn long range dependencies because of Markov property and a small discrete state space. Recent studies confirm that the use of features spanning a longer duration improves the performance as they can encode contextual information more accurately [57]. Second, the performance of these methods degrade when background noise with spectral characteristics similar to speech is present [58].

Recently, there has been tremendous progress in deep learning for sequences, especially for VAD in DEC. Mateju et al. [59] used a deep neural network trained on noise augmented dataset along with smoothing of the output for speech activity detection in movies. Jang et al. [60] used a 2 layered DNN with MFCC as the input feature for VAD in movies. Zhang et al. [61] used boosted deep neural network bDNN that generated multiple predictions from different contexts of a single frame by only one DNN and then aggregated the predictions for a better prediction of the frame. Hwang [62] used ensemble of DNNs. Kang et al. [63] used Multi

[5] See § 3.1 for the practical imperatives behind choosing this temporal resolution and for information on the training and test sets.

Task Learning (MTL) with DNN to estimate clean features from noisy features as well as VAD probabilities.

Recurrent Neural Network (RNN) variants such as Long Short Term Memory (LSTMs) [64], Gated Recurrent Units (GRUs) [25] and Convolution Neural Network (CNN) variant such as Temporal Convolution Networks (TCNs) have been used for sequence learning task such as voice activity detection [26,65]. Eyben et al. [26] used LSTMs with RASTA-PLP as the input features for VAD in movies. Similarly, several deep recurrent architectures have been used to create VAD systems [66–69] with various feature sets as input. For instance, Sainath et al., [70] and Zazo et al., [65] used raw audio waveform along with a hybrid CNN-LSTM model (CLDNN) for VAD. Both use raw audio waveform as input and show that the network can achieve performance similar to the network trained using MFCCs. Moreover, they show noise robustness of using CNN based backbone in CLDNN. Ferroni et al., [71] used a multiple features such as Wavelet Coefficient, RASTA-PLP and MFCCs for VAD in multi-room domestic scenarios. Tong et al., [58] studied the robustness of deep learning methods (LSTMs and CNNs) under varying noise conditions on Aurora 4 database. They also proposed a noise-aware training (NAT) to improve robustness to input noise. They show that through NAT, LSTM based VAD results in noise robust AUCs. To actualize noise awareness, the network is fed with noisy speech features (log-mel spectrogram) augmented with extra estimated information about current environmental conditions. Thomas et al., [72] used CNN for creating VAD system with inputs as 2D spectrogram and log-mel features. On a similar problem, Eyben et al., [26] use RASTA-PLP [73] features as input to LSTM [64] and show substantial improvements over statistical methods for VAD when tested on DEC. Their model is trained on TIMIT [74] and Buckeye [75] corpora mixed with synthetic noise. Recently, Hebbar et al. [1] proposed a noise robust CNN based Time Distributed (CNNTD) VAD model with VGGish CNN backbone [76]. This model was developed to analyze movie contents. They developed a Subtitle Aligned Movie (SAM) corpus consisting of limited 23 h of movie audio and trained CNNTD model on it. They showed considerable improvements over SVM baseline of Lehnar et al. [31] using in-domain data. Similarly, Lee et al. [28] proposed a noise robust model consisting of spectral and temporal attention mechanisms known as STNET. They used TIMIT corpus [77] augmented with 8 types of noise samples from NOISEX-92 dataset [78] to train the model. Finally, they demonstrated significant improvements over DNN and LSTM based VAD models under noisy conditions.

We observed most of the work in previous literature have presented analysis with a limited size corpora and feature sets for training and evaluation of the models. Therefore, in this work, we compare several architectures with various input feature representations such as linear and mel magnitude spectrograms, Instantaneous Frequencies (IF), Spectral Variances, Fluctograms and MFCCs on several large-scale in-domain DEC corpora. Details of these models and feature sets are mentioned in the Section 3.6.

3. Comparative Analysis Setup

In this section, we provide details of the proprietary and publicly available datasets used in this work for generating speech and non-speech labels. Further, we describe our feature representation and outline the various models used.

3.1. Datasets

Domain specific DEC datasets tend to result in better VAD models for DEC compared to the models trained on non-DEC focused datasets [1]. However, most of the publicly available VAD datasets are not DEC focused. The few that exists, are either of limited size or do not contain substantial background noise and embedded music characteristics. We therefore use our proprietary DEC videos curated from *Amazon Originals* [6] to create large-scale labelled VAD dataset focused on DEC containing a vast variety of embedded music and background noises. The details of these proprietary datasets are summarized below:

DEC-1100: This dataset comprises 1100 proprietary videos of movies and TV shows along with their high quality timed-text files amounting to 450 h of content in 9 languages and 5 + genres (Action, Comedy, Documentary, Drama, Animation etc.). The dataset is curated from Amazon Originals TV shows and movies for whom the subtitles are curated through multiple iterations of careful time stamps matching with the audio and hence we assume it to be of high quality. This dataset is used to train various models. Table 1 presents the language distribution of the dataset. All the datasets described henceforth are used in model testing.

DEC-test: This proprietary dataset comprises 40 h of content, which is manually labelled as speech or no-speech after processing. This dataset is derived from 33 movies which are not a part of DEC-1100 dataset.

DEC-language: This proprietary dataset comprises 28 h of the human labelled content curated from 23 movies and TV shows spanning 6 languages. The movies are not a part of DEC-1100 dataset. The audio language set comprises of English (en), German (de), Japanese (ja), Hindi (hi), Spanish (es) and Tamil (ta). This dataset is used to test language agnostic behavior.

Besides these three proprietary datasets, we also use the following publicly available datasets:

AVA: It [2] is a human labelled dataset of speech activity in movies that comprises 36 h of content curated from 192 YouTube movies. The dataset consists of 0.3–10 s clips with one of the 4 labels per clip namely, No-Speech, Clean Speech, Speech + Music and Speech + Noise. The clip distribution is defined in the Table 2.

MUSAN: The corpus [33] consists of 20 h of clean speech from the Librivox dataset, 40 h of speech from US government hearings, 42 h of music from various sources and 6 h of various noises from Freesound and Sound Bible database. We use Librivox part of MUSAN for our experiment due to its multilingual nature. The dataset contains languages which are part of training set as well as not in train set. Hence, this dataset is used to evaluate language agnostic characteristics of the models. Moreover, this dataset is also used to test the input noise sensitivity of models, by addition of several noise clips sampled from Audioset [80] at various SNRs.

3.2. Dataset Creation Process Of DEC-1100

Fig. 1 describes the industry standard timed-text creation guidelines [7]. We used these specifications to create a labelled dataset from DEC-1100 videos as follows:

1. We extracted the audio clip from the given video file and sampled it at 32 kHz.
2. We removed the time duration associated with captions (description of non-speech elements such as sound effects, relevant musical cues and other relevant audio information) from the audio clip using the timed-text/subtitle file.
3. We divided the remaining audio clip into speech and non-speech segments according to the time duration present in the subtitle. Each speech segment can range from 800 ms to 7 s in length as per subtitle guidelines, however there is no such constraint for non-speech segment.

[6] https://en.wikipedia.org/wiki/List_of_Amazon_original_programming

[7] https://partnerhelp.netflixstudios.com/hc/en-us/articles/215758617-Timed-Text-Style-Guide-General-Requirements.

Table 1
DEC-1100 video distribution by language, where the language code is identified using ISO-639 [79] (639–1) nomenclature.

Language Code	en	de	hi	ja	ko	fr	te	ta	es
%	68	1	13	13	2	1	1	1	1

Table 2
Original clip distribution of AVA dataset.

Clean speech	Speech with Music	Speech with Noise	Non Speech	Total
6,431	5,311	10,506	17,624	39,872

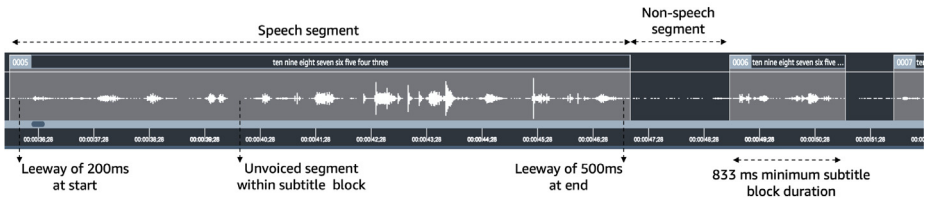

Fig. 1. Industry standard timed-text creation guidelines followed by human annotators. A timed-text file contains both speech and non-speech segments. The minimum duration of a speech segment is 5/6 s or ∼833 ms. While creating a timed-text file, the annotators may use a leeway of 200 ms and 500 ms at start and end of speech segment respectively to match the speed of speech and shot synchronization.

4. We divided each of the speech and non-speech segments into non-overlapping 800 ms (ms) clips.
5. Each 800 ms clip in speech segment was labelled as 1 and clip in non-speech segment was labelled as 0.

We chose the value of 800 ms for two reasons. First, a human speech block in a timed-text file persists on the screen for a minimum duration between $5/6^{th}$ of a second to one second [81] asrecommended by several industry standard guidelines[8]. These guidelines are based on the studies conducted on the reading speed of viewers. Second, disambiguation of a clip below 500 ms into speech and non-speech is difficult for human evaluators based on our manual inspection of clips.

We used the above procedure to divide the clips in other datasets into non-overlapping 800 ms clips. The datasets used in the paper with the number of 800 ms clips are described in the Table 3. We padded the clips by trailing zeros where the original clip duration was less than 800 ms. In our experiments, DEC-1100 is used to train the models whereas, DEC-test, DEC-language, AVA, MUSAN-librivox, FSDKaggle2019 and FMA_small are used for final validation of the models.

While AVA, MUSAN-librivox are human annotated, DEC-test is not. We needed a human annotated dataset for DEC for a fair evaluation. The following section outlines the labelling procedure used to tag the DEC-test dataset.

Table 3
Datasets used and the number of 800 ms clips.

Primary Dataset Name	Secondary Dataset Name	Non-Speech	Speech
DEC	DEC-1100	1,200,000	1,200,000
DEC	DEC-test	32,585	21,282
DEC-language	German	4,861	4,861
	English	5,632	5,632
	Spanish	11,044	11,044
	Hindi	9,268	9,268
	Japanese	5,234	5,234
	Tamil	858	858
MUSAN-librivox	Arabic	871	871
	Chinese	2,640	2,640
	Danish	817	817
	Dutch	5,991	5,991
	English	57,318	57,318
	French	3,811	3,811
	German	12,671	12,671
	Hebrew	2,618	2,618
	Hungarian	825	825
	Italian	6,443	6,443
	Japanese	1,375	1,375
	Latin	345	345
	Polish	1,522	1,522
	Portuguese	3,474	3,474
	Russian	3,901	3,901
	Spanish	4,944	4,944
	Tagalog	1,971	210,000

3.3. Human Labelling

The DEC-test dataset initially consisted of 53,867 clips (32,585 non-speech and 21,282 speech as annotated by timed-text files) as mentioned in the Table 3. We used a human labelling experiment to quantify the noise associated with timed-text annotations and generate a gold standard dataset for testing various models.

The 800 ms clips from DEC-test were manually annotated into one of the three categories, (a) human speech present, (b) human speech absent and (c) don't know. The instructions to human annotators were:

1. Listen to each 800 ms clip.
2. Tag the non speech human sounds such as laughing, crying, grunts, humming etc. as 'human speech absent'.
3. Tag the decipherable human speech such as human singing as 'human speech present'.
4. Discard human voice in the clip when present for a small duration such as ∼100 ms and tag the clip as 'human speech absent'.
5. In case of disambiguation, tag the clip as 'don't know'.

This tagging was performed in two phases. Fig. 2 presents the two phases used to label the DEC-test dataset:

[8] https://bbc.github.io/subtitle-guidelines.

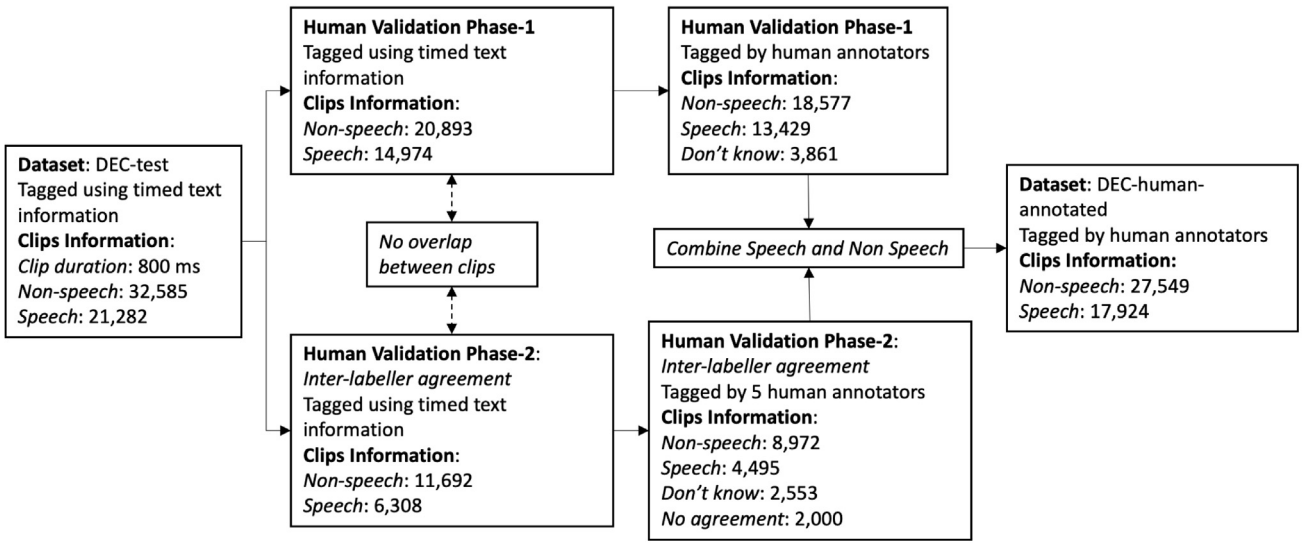

Fig. 2. Human labelling experiment pipeline. Phase 1 tagging was performed by a single annotator per clip. Phase 2 tagging involved estimation of inter-labeller agreement. Here, each clip was tagged by 5 labellers. Finally, we combine the output from two phases to obtain DEC–human-annotated dataset.

Phase-1: In the first phase, we tagged 35,867 of 53,867 clips (20,893 non-speech and 14,974 speech as annotated by timed-text guidelines) where, each clip was annotated by one labeller. Following the labelling, we obtained 18,577 non-speech, 13,429 speech and 3,861 don't know clips. Following a similar procedure, the DEC-language dataset was also tagged by human annotators and its number of clips in the Table 3 are post discarding 'don't know' clips.

Phase-2: In the second phase, we computed the inter-labeller consistency. We tagged the remaining 18,000 clips (11,692 non-speech and 6,308 speech as annotated by timed-text guidelines), where each clip was tagged by 5 labellers. The inter-labeller experiment is described below.

3.4. Inter-Labeller Agreement

To verify the labelling procedure, we calculated the inter-labeller agreement. We asked 5 labellers to tag 18,000 clips from DEC-test dataset into the three categories mentioned. We categorized the clip as 'No-agreement', if none of the three labels achieved a majority and considered the rest of them to be in agreement. Table 4 presents the distribution of clips for the inter-labeller agreement experiment. We observe that 10.5% (2 K/18 K) of the clips had no agreement.

We added the clips from the two phases to generate the DEC–human-annotated dataset. We discarded the 2 K clips where the labellers were in disagreement. The resulting distribution of the 800 ms clips for DEC–human-annotated dataset is defined in the Table 5. For all our experiments, we do not use the clips with labels as 'don't know'.

3.5. Label Noise Estimation In DEC-1100

We used a random subset of 37,448 (19,529 non-speech, 14,078 speech and 3,841 don't know) clips from DEC–human-annotated dataset (post discarding the No-agreement clips) to approximate the label noise present in DEC-1100 dataset labelled using timed-text information. The results of a comparison between the labels obtained through human labelling experiment described above and timed-text files are summarized in the Table 6. We observe that the amount of noise in speech segments is 8%, while in non-speech segments it is 22%. This label noise was mainly because of the three reasons: (a) Drift between the audio and text in timed-text files because of labelling errors or file format conversion issues, (b) Human subjectivity within timed-text specifications to incorporate artistic intent or scene changes and (c) Unvoiced segments within the speech segment of a subtitle block as mentioned in the Fig. 1 which may contain silence, music or sounds other than speech.

Table 4
Inter-labeller agreement.

Non-speech	Speech	Don't know	No-agreement
8,972	4,495	2,553	2,000

Table 5
Clip distribution for DEC–human-annotated dataset after discarding 'No-agreement' clips.

Non-Speech	Speech	Don't know
27,549	17,924	6,394

Table 6
The confusion matrix between the labels obtained from human annotators and timed-text file across 37,488 samples.

Annotations		Labels from Subtitles	
		Human Speech	No Human Speech
Labels from Human Annotators	Human Speech	12,975 (92%)	1,103 (8%)
	No Human Speech	4,204 (21%)	15,325 (79%)
	Don't Know	1,592	2,249

3.6. Overview of Neural Architectures and Feature Representations

Spectrogram representation of an audio signal is a sequence through which we can capture long term dependencies in the sig-

nal. These dependencies enables us to differentiate between speech and other sounds. Spectrogram representation of the audio have been shown to outperform other feature representations such as MFCCs, Spectral Contrast etc. [82–84]. Hence, we compare neural architectures that use a time sequence as their input particularly, GRU [26], TCN [27], STNET [28] and CNNTD [1] models.

For the deep sequence based architectures considered, we compute log linear, log mel magnitude spectrogram (log mel-STFT) and Instantaneous Frequencies (IF) [85] as features. We use a 25 ms window (window length corresponding to 800 samples) and 10 ms (480 samples) hop length, which results in a $F \times T$ dimensional signal for each 800 ms clip sampled at 32 kHz (signal length of 25600 samples). Here, $F = 401$ is the number of frequency bins for the linear log magnitude STFT and IF. The value of F is calculated as $1 + \lfloor \frac{\text{window length}}{2} \rfloor$. The value of F for log mel-STFT varies with the models considered. $T = 54$ is the number of time bins calculated as $\lceil \frac{\text{signallength}}{\text{hop length}} \rceil$. For our experimentation, we consider, first 128 frequency components out of 401 corresponding to 0–10 kHz frequency range. We also compare shallow neural models using one dimensional features such as MFCCs, Fluctogram, Spectral Contrast and phase information (phase difference, Instantaneous Frequencies) etc. Finally, we also compare CLDNN [3] model which uses raw waveform as input. All our models use cross-entropy as the loss function. We trained all our models using python-based *PyTorch* API [9]. We now describe various neural architectures and input feature representations used in them.

1. TCN: Chang et al. [27] proposed a 36 layered causal dilated gated residual CNN for VAD followed by two fully connected (FC) layers with 128 neurons in each. The input to the network is $F = 80$ dimensional log-mel STFT with $T = 54$ time bins. The network consists of 2.8 M parameters.

2. GRU: LSTMs and GRUs have been used for VAD. Following Eyben et al. [26] we use a 2 layered GRU model followed by 2 FC layers. We did not use RASTA-PLP feature representation and replaced the LSTM in [26] with GRU. We compare following three feature sets for the GRU:

2a. GRU-MS: $F = 96$ dimensional log-mel STFT with $T = 54$ time bins. The model consists of 550 K parameters.

2b. GRU-S: $F = 128$dimensional log STFT with $T = 54$ time bins. The model consists of 600 K parameters.

2c. GRU-SF: We use two feature maps, a) $F = 128$ dimensional log STFT with $T = 54$ time bins and b) $F = 128$ dimensional reassigned frequency or Instantaneous frequencies (IF) [85] with $T = 54$ time bins as input to the model. We use additional information in terms of phase following the work of Longbiao et al. and Iain et al., [86,87] where, they proposed the use of IF and other phase related features to improve VAD. The IFs were calculated using the Eq. 1.

$$\hat{\omega} = \omega - \Im\left(\frac{S_{dh}}{S_h}\right), \tag{1}$$

where, $\Im$ denotes the imaginary part of the matrix, S_h is the complex STFT calculated using *hann* window and S_{dh} is the complex STFT calculated using the derivative of *hann* window. The two features are passed through two GRU models (each containing 2 GRU layers). The output of the GRUs are concatenated and then passed through 2 FC layers. This model has 1.2 M parameters.

3. CNNTD: Proposed in [1], the CNNTD model uses the convolution blocks of the VGG net [76] type architecture with 3 convolutional blocks containing 2 convolutional layers each, followed by a time distributed (TD) layer and 2 FC layers. We compare three feature sets for this model:

3a. CNNTD-MS: $F = 96$ dimensional log-mel STFT with $T = 54$ time bins. The model has 500 K parameters.

3b. CNNTD-S: $F = 128$ dimensional log STFT with $T = 54$ time bins. The model has 550 K parameters.

3c. CNNTD-SF: Similar to GRU-SF we used log STFT and IF as the input to the model. Instead of passing them through two models as in GRU-SF, we concatenate the two feature maps and pass them through one CNNTD model. This model has 550 K trainable parameters.

4. STNET: Lee et al. [28] proposed Spectro-Temporal network (STNET) for a noise robust VAD. The model is an amalgamation of 4 convolutional, 2 FCs (Pipe Net) and a transformer encoder layer (self-attention + FC) [29]. The input to model consists of $F = 96$ dimensional log-mel STFT with $T = 54$ time bins and contains 2.4 M parameters.

5. CLDNN: Zazo et al. [3] proposed Convolutional, Long Short Term Memory, Deep Neural network (CLDNN) architecture that uses raw audio waveform as input feature. The model consists of 3 convolutional layers with max pooling followed by 2 layer LSTM and 2 FC layers with $F = 128$ dim in each. The model has 850 K parameters. They show that using raw waveform allows the network to learn powerful features comparable to log-mel features, especially for noisy environments.

6. FC-1: Following Jang et al. and Lehner et al. [60,88] we use a 2 hidden layered FC network with each layer containing 256 neurons. We use a 40 dimensional MFCC averaged over $T = 54$ time bins as input. The network consists of 77 K parameters.

7. FC-2: Mateju et al. [59] proposed a 5 layered FC network with 128 neurons in each layer for speech activity detection in online broadcast transcription. The model uses 40 dimensional MFCCs averaged over $T = 54$ time bins as the input features and consists of 72 K parameters.

8. FC-3: Wang et al. [86] proposed a 1 hidden layer FC network that utilizes both magnitude and phase information for VAD. They used MFCC, IF derivative, Baseband Phase Difference (BPD), and Modified Group Delay Cepstral Coefficients (MGDCC) as the features and show that their network outperform networks using only magnitude information. The input to this network consists of 128 dimensional IF derivative, BPD and MGDCC with their means and standard deviations over $T = 54$ time bins and 40 dimensional MFCCs averaged over $T = 54$ time bins resulting in 808 dimensional input vector. We use a network with 2 hidden FC layers with 256 neurons in each layer. The network has 270 K parameters.

9. FC-4: Following the works of Lehner et al. [31,88,32] and Lee et al. [89] we use 40 dimensional MFCCs averaged over $T = 54$ time bins, 17 dimensional each of bandwise (17 mel scaled frequency bands from 164 Hz to 10548 Hz) Fluctorgram variances, Spectral contraction variances and Spectral flatness means resulting in 91 dimensional input vector for a given 800 ms clip. We use a network with 2 hidden FC layers with 256 neurons in each layer. The network has 100 K parameters.

10. WebRTC: A good baseline for VAD performance is the WebRTC VAD [35]. It uses GMM models of speech and non-speech sounds with input features as log energies of six frequency bands between 80 Hz-4000 Hz. It utilizes fixed point operations and is optimized for real-time use for web transmission. We use py-webtrcvad [10] implementation of VAD with an aggressiveness level of 3 for VAD. The method generates a boolean output for a 30 ms frame. We consider a 800 ms clip to contain speech if VAD generates at least 40% frames containing speech. The threshold of 40% was a result of hyperparameter tuning on DEC-1100 dataset.

In the following section, we describe the experimental settings, results and inferences.

[9] https://pytorch.org/

[10] https://github.com/wiseman/py-webrtcvad

4. Results

In this section, we discuss the hyperparameter settings and the experimentations. All the experiments were conducted on a p3.2xlarge EC2 instance [11]. We used Adam optimizer [90] in all our experiments with a batch size of 512 for all the models, except for STNET and TCN, where we used a batch size of 32. We used a learning rate of 10^{-3}, weight decay parameter of 10^{-7} and gradient clipping of 10000 for all the models. We trained all the models for 20 epochs and used a learning rate scheduler where we decreased the learning rate by a factor of 10 at 10^{th} and 15^{th} epochs. We stopped training if the relative decrease in the validation loss 5 epochs apart is less than 10^{-3}. We trained all the models on DEC-1100 dataset where, we used 1,150,000 speech and 1,150,000 non-speech clips for training the models and 100 K clips (50 K speech and 50 K non-speech) for validation. We tested the performance of models trained on DEC-1100 on the benchmark datasets DEC–human-annotated and AVA datasets.

4.1. Model Performance On DEC–human-annotated And AVA Datasets

DEC–human-annotated: Table 7 presents the accuracy (Acc), Area Under Curve (AUC), precision (p), recall (r) and F-score (f) for models trained on DEC-1100 dataset. We did not compute AUC for WebRTC VAD as the method did not generate probabilistic output. We observe that CNNTD-SF has the highest AUC of 0.955 closely followed by GRU-S model. In terms of accuracy, the CNNTD-S has the highest accuracy of 0.876 followed by GRU-SF model 0.871. We observe that models using mean statistics over time of MFCCs, Fluctograms, Spectral Contrast features (FC-1 to FC-4) have considerably smaller accuracies and AUCs as compared to deep models with Spectrogram variants as input. We attribute this performance gain of deep networks to the use of sequence information available in spectrograms by deep models. The WebRTC VAD's hyperparameters were optimized on DEC-1100 dataset. We observe that WebRTC VAD results in a low accuracy of 61% and provides empirical evidence of the superiority of deep architectures as opposed to simpler models such as GMMs for audio data with various degrees of input noises. For the rest of the paper, we report AUC as the metric of choice, following the works of [91,26] and do not compare WebRTC VAD further.

Statistical Significance Test: We used DeLong's test [92,93] to obtain a p-value that identifies whether one model has a significantly different AUC than another model. Fig. 3 presents the statistically significant AUCs. We used a p-value $< 10^{-3}$ to define statistical significance. Following the AUC and p-values values from Table 7 and Fig. 3 we observe the following:

1. CNN and GRU based networks retain high AUCs and accuracy despite $\sim$ 15% label noise in the dataset. This shows that deep networks are able to generalize well even in noisy conditions.
2. CNNTD-SF is statistically similar to CNNTD-MS and is better than CNNTD-S. For GRU based models we observe that GRU with linear spectrogram results in the highest AUC.
3. Models utilizing temporal information and spectrogram based features such as CNN and GRU based methods results in better AUCs than FC based methods.

AVA: Following [1,2] we compare the True Positive Rate (TPR) of the models at False Positive Rate (FPR) of 0.315. The AVA dataset provide annotations at 0.3 to 10 s granularity of each clip, however, the trained models provide probability of speech at 800 ms granularity. To obtain the probability of speech for the clips in AVA we used the following procedure:

1. The clips of size smaller than 800 ms were appended with trailing zeros to make their size equal to 800 ms. We computed the probability of speech for each 800 ms clip.
2. The clips of size greater than 800 ms (for e.g., 3 s) were divided into smaller 800 ms clips with 50% overlap between consecutive 800 ms clips. We computed probability of speech for each 800 ms clip and averaged over all the 800 ms clips present in the larger clip.

We computed the TPRs of the model at an FPR of 0.315 as shown in the Table 8 and present the TPRs on 'Clean Speech', 'Speech with Music', 'Speech with Noise' sections of the AVA dataset and the full AVA dataset ('All'). We also compared the current state-of-the-art ResNet-960 model (30 M parameters) trained on Audioset [2] and CNNTD model trained on the SAM corpus [1] (740 K parameters). We observe that for the models trained on DEC-1100 dataset, GRU-SF model outperforms other models and

[11] https://aws.amazon.com/ec2/pricing/on-demand/

Table 7
Measures compared with models trained on DEC-1100 and tested on DEC–human-annotated.

Model Name	Acc	AUC	p	r	f
CLDNN	0.852	0.915	0.877	0.852	0.854
CNNTD-SF	0.873	**0.955**	0.881	0.873	0.874
CNNTD-S	**0.876**	0.932	**0.887**	**0.876**	**0.878**
CNNTD-MS	0.866	0.947	0.876	0.866	0.867
FC-1	0.774	0.933	0.797	0.774	0.776
FC-2	0.774	0.869	0.795	0.774	0.777
FC-3	0.822	0.871	0.830	0.822	0.823
FC-4	0.816	0.878	0.831	0.816	0.818
GRU-SF	0.871	0.902	0.887	0.871	0.872
GRU-S	0.870	0.951	0.877	0.870	0.871
GRU-MS	0.860	0.936	0.875	0.860	0.861
STNET	0.863	0.940	0.863	0.863	0.861
TCN	0.875	0.900	**0.887**	0.875	0.876
WebRTC	0.615	-	0.757	0.615	0.597

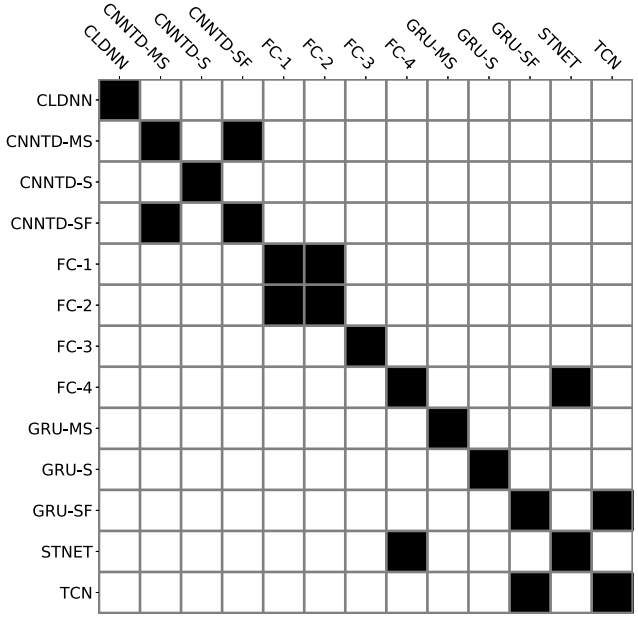

Fig. 3. Pairwise statistical significance test for difference in AUCs as defined by Delong's test for models tested on DEC–human-annotated. White boxes denotes statistical significant difference at p-value threshold of 10^{-3}.

Table 8
TPRs compared with models trained on DEC-1100 and tested on original clips of AVA. Methods marked with * were not trained on DEC-1100 dataset and the results were obtained from their respective papers.

	TPR at FPR of 0.315			
Model Name	**Clean Speech**	**Speech with Music**	**Speech with Noise**	**All**
CLDNN	0.964	0.922	0.946	0.946
CNNTD-SF	0.980	0.916	0.947	0.949
CNNTD-S	0.979	0.913	0.949	0.949
CNNTD-MS	0.976	0.921	0.953	0.952
FC-1	0.880	0.701	0.821	0.809
FC-2	0.864	0.663	0.806	0.788
FC-3	0.952	0.843	0.928	0.915
FC-4	0.925	0.826	0.899	0.889
GRU-SF	0.981	**0.929**	**0.957**	**0.957**
GRU-S	0.966	0.896	0.930	0.932
GRU-MS	0.967	0.888	0.919	0.926
STNET	0.958	0.907	0.940	0.937
TCN	0.973	0.906	0.941	0.942
***CNNTD** [1]	0.983	0.917	0.939	0.945
***ResNet-960** [2]	**0.992**	0.787	0.944	0.917

improves the state-of-the-art [1] by 1.3% (relative) on both 'Speech with Music' and 'Speech with Noise'. Further, GRU-SF shows up to 18% relative improvement in noisy conditions over ResNet-960. Models trained on DEC-1100 outperforms the models trained on other corpora [1,2] justifies the need for domain-specific datasets for VAD to be used for DEC. For the subsequent experiments, we do not compare the FC variants as they have shown to obtain lower AUCs and accuracy as compared to their deep convolutional and recurrent variants on both DEC–human-annotated and AVA dataset.

4.2. Ability To Generalize Across Languages

The process of timed-text creation is primarily manual, where human operators take up to 20 h of effort for 1 h of content. Through our baselining exercise we found that up to 8 h (40%) of manual labour can be spent on identifying the time stamps where a human dialogue needs subtitling. In this situation, VAD presents a compelling solution where it automatically identifies the time duration associated with human speech, leading to an approximately 40% efficiency improvement in timed-text creation process. Moreover, since each movie or TV episode can have subtitles in multiple languages which is imperative to obtain world-wide reach, training VAD models that are language agnostic and can perform accurately across multiple languages is of paramount interest.

We evaluated the AUC of neural VAD models on DEC-language and MUSAN-librivox datasets (Table 3) consisting of 6 and 17 languages respectively.

Table 10 presents the AUC of various models tested on DEC-language dataset. We observe that GRU variants and CLDNN outperform STNET consistently across languages with AUCs $\geqslant 0.92$, while other neural models empirically demonstrate language agnostic behaviour with AUCs $\geqslant 0.9$ on multiple languages.

Table 9 presents the AUCs across MUSAN-librivox dataset which consists of clean speech. We observe that CNNTD and GRU variants along with CLDNN outperform STNET consistently across languages with AUCs greater than 0.99. We observe higher AUCs on this dataset since it consists of clean recordings with minimal noise. Further, neural models retain very high AUCs on 12 of 17 languages which were not a part of training dataset, highlighting the true language agnostic behaviour of the deep neural models.

4.3. Input Noise Analysis

DEC-1100 dataset comprises of 5 primary genres namely, *action*, *animation*, *comedy*, *documentary* and *drama*. Table 11 presents the distribution of genres present in DEC-1100 dataset. However, 9% of the content is unclassified (others) and contains genres such as *adventure*, *thriller*, *horror* and *musical*. These genres have their typical background scores and ambient noises occurring at various magnitudes. Although we tried to retain the widest representative coverage of genres in our training dataset, a complete coverage of all genres was infeasible. Hence, we require a VAD that maximally generalises to these types of genre-dependent noises occurring at various SNRs. Therefore, we compared the recall of models for predicting speech at various signal-to-noise (SNR) ratios.

To simulate the noise conditions present in the movies and TV shows, we selected at random 2 clean wav files for each language present in the MUSAN-librivox dataset (outlined in the Table 3). The 2 files consists of one male and one female speaker. To each clean wav file, we added 7 different noise signals at SNRs of 10, 5, 0, −5, −10 and −15 respectively to create noisy speech files. The SNRs are divided into three sets; high SNRs (10 and 5), medim SNRs (0 and −5) and low SNRs (-10 and −15). These noisy speech files were further broken into model acceptable 800 ms chunks, with each chunk given a label of 1 denoting 'contains speech'. The noise wav files were selected from Audioset [80]. The noise types consists of babble noise, environmental noise, pink noise,

Table 9
AUC of various models trained on **DEC-1100** and tested on **MUSAN-librivox**.

Languages	**CLDNN**	**CNNTD-MS**	**CNNTD-S**	**CNNTD-SF**	**GRU-MS**	**GRU-S**	**GRU-SF**	**STNET**	**TCN**
Arabic	0.997	0.999	0.999	0.999	**1.000**	0.999	0.999	0.996	**1.000**
Chinese	**0.997**	0.993	0.996	**0.997**	0.988	0.992	0.995	0.973	0.995
Danish	0.998	0.999	**1.000**	0.998	0.998	**1.000**	0.998	0.999	**1.000**
Dutch	0.997	0.998	**0.999**	0.995	0.997	0.996	0.996	0.993	0.998
English	**0.996**	**0.996**	**0.996**	0.995	0.992	0.992	0.995	0.990	0.995
French	0.990	0.992	**0.994**	0.992	0.983	0.988	0.990	0.982	0.992
German	0.997	0.997	**0.998**	0.996	0.995	0.995	0.995	0.993	0.997
Hebrew	0.991	0.992	**0.995**	0.993	0.992	0.990	0.991	0.982	0.990
Hungarian	0.999	**1.000**	**1.000**	0.999	**1.000**	**1.000**	**1.000**	0.998	**1.000**
Italian	0.987	0.985	0.987	0.986	0.982	0.984	0.985	0.982	**0.989**
Japanese	**0.993**	0.988	0.989	0.987	0.985	0.988	0.972	0.987	0.967
Latin	0.997	0.998	**0.999**	**0.999**	**0.999**	**0.999**	0.998	0.987	**0.999**
Polish	0.998	0.997	**0.999**	0.997	0.994	0.995	0.993	0.999	**0.999**
Portuguese	0.984	0.984	0.988	0.983	0.985	0.982	0.980	0.966	**0.989**
Russian	**0.996**	0.995	**0.996**	0.995	0.990	0.992	0.993	0.988	0.994
Spanish	**0.995**	0.991	0.991	0.989	0.989	0.987	0.982	0.986	0.986
Tagalog	0.991	0.994	**0.997**	**0.997**	0.987	0.991	0.993	0.990	0.993

Table 10
AUC of various models on DEC-language.

Model Name	German	English	Spanish	Hindi	Japansese	Tamil
CLDNN	**0.963**	**0.953**	0.957	**0.923**	**0.945**	0.921
CNNTD-MS	0.939	0.925	0.927	0.891	0.934	0.885
CNNTD-S	0.947	0.940	0.943	0.903	0.943	0.905
CNNTD-SF	0.934	0.918	0.922	0.890	0.934	0.875
GRU-MS	0.949	0.943	0.961	0.902	0.930	0.924
GRU-S	0.941	0.930	0.940	0.900	0.927	0.896
GRU-SF	0.954	0.949	**0.962**	0.913	0.939	**0.928**
STNET	0.869	0.886	0.889	0.846	0.912	0.844
TCN	0.951	0.948	0.958	0.909	0.947	0.921

Table 11
Distribution of genres across DEC-1100 dataset.

Genre	Animation	Comedy	Documentary	Drama	Action	Others
%	41	20	14	14	2	9

subway noise, theme music, traffic noise and white noise. The selected noise types are some of the most common noises present in DEC. Babble noise is usually present in DEC as part of scenes where people are chatting/murmuring in the background such as, inside a hall, in meetings and in gatherings. Environmental noise is usually present in outdoor scenes and is associated with birds and animal sounds. Pink noise is similar to the sound of rustling leaves, wind and steady rain. Subway noise is associated with scenes inside a subway or metro station, which includes announcements in background and train sounds. Theme music contains the background scores/music from several instruments such as piano, guitar etc., and is the most common type of sound present in DEC along with speech. Traffic noise is associated with vehicular noise such as engine revving and horns. Finally, white noise is usually associated with indoor scenes and includes sounds of fan whirring, air conditioner, radio static and old movies which have a prominent continuous hissing, closely imitating radio static sound. We performed inference for all the neural models on the noisy speech files and computed the average recall of the models across languages. For this experiment, the precision will always be 1 and thus recall is the metric of choice, as there are no explicit non-speech signals in the noisy speech files.

We divided the results into two parts, a) Average recall score for the languages which are present in the training set (DEC-1100) including, English, German, French and Spanish, and b) Average recall score for the languages which are not part of training set (DEC-1100) including, Arabic, Chinese, Danish, Dutch, Hebrew, Hungarian, Italian, Latin, Polish, Portuguese, Russian and Tagalog. Using the two parts, we also demonstrate the language agnostic characteristics of the deep learning models. We shall now describe the individual noise experiments.

Result with babble noise: Tables 12a and 12b presents the average recall scores of various models trained on DEC-1100 across the two sets; a) containing languages present in the training set, and b) not containing languages in the training set. We observe that CNNTD-SF and CNNTD-MS results in the highest recall scores

Table 12
Mean Recall across languages for babble noise added to clean MUSAN-librivox clips.

(a) For languages in train set.

Recall	SNR					
Models	**−15**	**−10**	**−5**	**0**	**5**	**10**
CLDNN	**0.282**	0.275	0.354	0.537	0.711	0.793
CNNTD-SF	0.151	**0.304**	**0.489**	**0.647**	**0.791**	**0.852**
CNNTD-S	0.063	0.140	0.277	0.502	0.712	0.814
CNNTD-MS	0.182	0.239	0.380	0.632	0.772	0.826
GRU-SF	0.040	0.091	0.171	0.216	0.594	0.802
GRU-S	0.044	0.094	0.198	0.338	0.510	0.660
GRU-MS	0.040	0.091	0.216	0.394	0.567	0.689
STNET	0.149	0.258	0.388	0.529	0.682	0.773
TCN	0.034	0.077	0.192	0.454	0.665	0.746

(b) For languages not in train set.

Recall	SNR					
Models	**−15**	**−10**	**−5**	**0**	**5**	**10**
CLDNN	**0.277**	**0.251**	0.370	0.587	0.748	0.811
CNNTD-SF	0.177	0.329	**0.518**	**0.696**	0.803	0.841
CNNTD-S	0.077	0.157	0.322	0.577	0.758	0.821
CNNTD-MS	0.201	0.250	0.413	0.686	**0.825**	**0.848**
GRU-SF	0.056	0.108	0.160	0.302	0.700	0.819
GRU-S	0.050	0.105	0.226	0.424	0.606	0.733
GRU-MS	0.056	0.108	0.247	0.471	0.657	0.757
STNET	0.148	0.239	0.395	0.583	0.720	0.786
TCN	0.046	0.086	0.232	0.545	0.756	0.814

at high SNRs (10 and 5). At medium SNRs (0, −5) of CNNTD-SF results in highest recall. At low SNRs (-10 and −15) CLDNN outperforms others. The CNNTD variants outperform others by a large margin at high SNRs for babble noise as babble noise consists of human speech sounds and network is able to differentiate between clean and noisy speech files. The networks show a similar performance for both the cases of languages present and absent in DEC-1100 train set.

Result with environmental noise: Tables 13a and 13b presents the average recall scores of various models trained on DEC-1100 across the two sets. We observe that CNNTD-SF and CNNTD-MS results in highest recall scores at high SNRs (10 and 5). At medium SNRs of 0 and −5 CNNTD-MS results in highest recall. At low SNRs, CLDNN has the highest recall. The networks show a similar performance for both languages present and absent in DEC-1100 train set. However, CLDNN is the most robust architecture with smallest decline in recall with increasing noise, which is due to noise robust features learnt in the lower CNN layers as outlined in [3].

Result with pink noise: Tables 14a and 14b presents the average recall scores of various models trained on DEC-1100 across the two sets. We observe that at high SNR of 10, CNNTD-SF results in highest recall score, while for medium and low SNRs, CLDNN is the best performing network. The networks show similar performance for both languages present and absent in DEC-1100 train

Table 13
Mean Recall across languages for environmental noise added to clean MUSAN-librivox clips.

(a) For languages in train set.

Recall	SNR					
Models	**−15**	**−10**	**−5**	**0**	**5**	**10**
CLDNN	**0.541**	**0.649**	**0.694**	0.726	0.784	0.808
CNNTD-SF	0.166	0.333	0.536	0.722	0.815	**0.855**
CNNTD-S	0.093	0.194	0.399	0.622	0.778	0.829
CNNTD-MS	0.302	0.453	0.622	**0.742**	**0.822**	0.844
GRU-SF	0.061	0.139	0.171	0.216	0.594	0.802
GRU-S	0.078	0.155	0.290	0.453	0.588	0.681
GRU-MS	0.061	0.139	0.309	0.503	0.640	0.717
STNET	0.237	0.393	0.529	0.648	0.718	0.785
TCN	0.108	0.233	0.418	0.614	0.729	0.772

(b) For languages not in train set.

Recall	SNR					
Models	**−15**	**−10**	**−5**	**0**	**5**	**10**
CLDNN	**0.563**	**0.647**	**0.701**	0.755	0.799	0.818
CNNTD-SF	0.196	0.345	0.550	0.746	0.823	**0.840**
CNNTD-S	0.120	0.228	0.431	0.668	0.786	0.822
CNNTD-MS	0.331	0.480	0.637	**0.762**	**0.827**	0.839
GRU-SF	0.088	0.169	0.160	0.302	0.700	0.819
GRU-S	0.102	0.175	0.327	0.520	0.671	0.744
GRU-MS	0.088	0.169	0.358	0.564	0.711	0.774
STNET	0.220	0.382	0.526	0.643	0.731	0.779
TCN	0.143	0.288	0.472	0.676	0.787	0.814

Table 14
Mean Recall across languages for pink noise added to clean MUSAN-librivox clips.

(a) For languages in train set.

Recall	SNR					
Models	**−15**	**−10**	**−5**	**0**	**5**	**10**
CLDNN	**0.516**	**0.625**	**0.712**	**0.754**	**0.796**	0.816
CNNTD-SF	0.000	0.006	0.072	0.332	0.665	**0.828**
CNNTD-S	0.000	0.000	0.010	0.166	0.557	0.777
CNNTD-MS	0.002	0.001	0.011	0.221	0.615	0.784
GRU-SF	0.000	0.000	0.171	0.216	0.594	0.802
GRU-S	0.000	0.000	0.004	0.064	0.310	0.586
GRU-MS	0.000	0.000	0.005	0.105	0.411	0.637
STNET	0.101	0.064	0.006	0.080	0.438	0.687
TCN	0.000	0.000	0.004	0.162	0.545	0.721

(b) For languages not in train set.

Recall	SNR					
Models	**−15**	**−10**	**−5**	**0**	**5**	**10**
CLDNN	**0.508**	**0.609**	**0.691**	**0.768**	**0.812**	0.826
CNNTD-SF	0.000	0.007	0.088	0.444	0.739	**0.835**
CNNTD-S	0.000	0.001	0.014	0.234	0.641	0.789
CNNTD-MS	0.004	0.001	0.014	0.299	0.716	0.813
GRU-SF	0.000	0.000	0.160	0.302	0.700	0.819
GRU-S	0.000	0.000	0.006	0.101	0.439	0.688
GRU-MS	0.000	0.000	0.004	0.133	0.500	0.712
STNET	0.002	0.002	0.004	0.109	0.485	0.698
TCN	0.000	0.000	0.005	0.218	0.666	0.794

set. Networks other than CLDNN results in lower performance with decreasing SNRs, as the dialogues in DEC-1100 train dataset, does not contain speech mixed with pink noise with such low SNRs.

Result with subway noise: Tables 15a and 15b presents the average recall scores of various models trained on DEC-1100 across the two sets. We observe that at high SNR of 10, CNNTD-SF results in highest recall score, while for medium and low SNRs, CLDNN is the best performing network. The networks show similar performance for both languages present and absent in DEC-1100 train set. Networks other than CLDNN results in lower performance with decreasing SNRs, as the dialogues in DEC-1100 train dataset, does not contain speech mixed with subway noise with such low SNRs.

Result with theme noise: Tables 16a and 16b presents the average recall scores of various models trained on DEC-1100 across the two sets. We observe that at high SNRs, CNNTD-SF results in highest recall score, while for very low SNRs, no model can identify speech. The networks show similar performance for both languages present and absent in DEC-1100 train set. The low performance of the networks at low and medium SNR is attributed to the fact that musical instruments have fundamental frequencies

Table 15
Mean Recall across languages for subway noise added to clean MUSAN-librivox clips.

(a) For languages in train set.

Recall	SNR					
Models	**−15**	**−10**	**−5**	**0**	**5**	**10**
CLDNN	**0.558**	**0.639**	**0.707**	**0.740**	**0.786**	0.814
CNNTD-SF	0.028	0.071	0.165	0.416	0.678	**0.823**
CNNTD-S	0.012	0.027	0.071	0.250	0.580	0.770
CNNTD-MS	0.044	0.057	0.114	0.382	0.670	0.789
GRU-SF	0.016	0.025	0.171	0.216	0.594	0.802
GRU-S	0.024	0.027	0.048	0.139	0.356	0.568
GRU-MS	0.016	0.025	0.060	0.203	0.471	0.642
STNET	0.132	0.071	0.074	0.216	0.511	0.712
TCN	0.016	0.031	0.061	0.263	0.598	0.725

(b) For languages not in train set.

Recall	SNR					
Models	**−15**	**−10**	**−5**	**0**	**5**	**10**
CLDNN	**0.538**	**0.619**	**0.703**	**0.759**	**0.794**	0.821
CNNTD-SF	0.038	0.078	0.208	0.523	0.756	**0.822**
CNNTD-S	0.015	0.030	0.079	0.333	0.670	0.791
CNNTD-MS	0.041	0.061	0.135	0.454	0.748	0.816
GRU-SF	0.024	0.035	0.160	0.302	0.700	0.819
GRU-S	0.024	0.035	0.063	0.189	0.473	0.678
GRU-MS	0.018	0.026	0.063	0.239	0.553	0.724
STNET	0.058	0.058	0.075	0.250	0.574	0.736
TCN	0.021	0.034	0.072	0.346	0.699	0.802

Table 16
Mean Recall across languages for theme music added to clean MUSAN-librivox clips.

(a) For languages in train set.

Recall	SNR					
Models	**−15**	**−10**	**−5**	**0**	**5**	**10**
CLDNN	**0.069**	**0.009**	0.000	0.043	0.404	0.707
CNNTD-SF	0.000	0.002	0.053	**0.407**	**0.753**	**0.862**
CNNTD-S	0.000	0.000	0.027	0.291	0.680	0.821
CNNTD-MS	0.004	0.003	0.047	0.383	0.720	0.831
GRU-SF	0.000	0.000	**0.171**	0.216	0.594	0.802
GRU-S	0.002	0.010	0.080	0.371	0.650	0.759
GRU-MS	0.000	0.000	0.025	0.225	0.530	0.677
STNET	0.006	0.002	0.079	0.478	0.719	0.794
TCN	0.000	0.000	0.015	0.257	0.618	0.747

(b) For languages not in train set.

Recall	SNR					
Models	**−15**	**−10**	**−5**	**0**	**5**	**10**
CLDNN	**0.054**	0.004	0.000	0.062	0.469	0.759
CNNTD-SF	0.000	0.004	0.067	**0.522**	**0.804**	**0.851**
CNNTD-S	0.000	0.000	0.034	0.391	0.728	0.817
CNNTD-MS	0.001	0.003	0.049	0.462	0.789	0.841
GRU-SF	0.000	0.000	**0.160**	0.302	0.700	0.819
GRU-S	0.002	**0.007**	0.101	0.487	0.736	0.798
GRU-MS	0.000	0.000	0.028	0.298	0.623	0.750
STNET	0.000	0.002	0.106	0.554	0.749	0.798
TCN	0.000	0.000	0.019	0.346	0.728	0.815

in the range of 100–400 Hz, which overlaps largely with speech fundamental frequency range of 85–300 Hz.

Result with traffic noise: Tables 17a and 17b presents the average recall scores of various models trained on DEC-1100 across the two sets. We observe that at high SNRs, CNNTD-SF results in highest recall score, while for medium and low SNRs, CLDNN is the best performing network. The networks show similar performance for both languages present and absent in DEC-1100 train set. Networks other than CLDNN results in lower performance with decreasing SNRs, as the dialogues in DEC-1100 train dataset, does not contain speech mixed with traffic noise with such low SNRs.

Result with white noise: Tables 18a and 18b presents the average recall scores of various models trained on DEC-1100 across the two sets. We observe that at high SNR of 10 and 5, CNNTD-SF results in highest recall score, while for medium SNRs, STNET is the best performing network and at low SNRs, no model is able to distinguish speech. The networks show similar performance for both languages present and absent in DEC-1100 train set. White noise is not usually present with high SNRs in DEC, as it deteriorates the audio perceptual quality and hampers listening experience. Therefore, the networks do not perform well at medium and low SNRs.

Table 17
Mean Recall across languages for traffic noise added to clean MUSAN-librivox clips.

(a) For languages in train set.

Recall	SNR					
Models	**−15**	**−10**	**−5**	**0**	**5**	**10**
CLDNN	**0.331**	**0.344**	**0.461**	**0.634**	**0.750**	0.797
CNNTD-SF	0.035	0.074	0.154	0.369	0.696	**0.832**
CNNTD-S	0.010	0.030	0.101	0.303	0.635	0.787
CNNTD-MS	0.018	0.045	0.120	0.409	0.704	0.815
GRU-SF	0.000	0.007	0.171	0.216	0.594	0.802
GRU-S	0.005	0.021	0.059	0.192	0.454	0.640
GRU-MS	0.000	0.007	0.045	0.222	0.507	0.676
STNET	0.032	0.027	0.060	0.232	0.563	0.734
TCN	0.005	0.016	0.067	0.306	0.616	0.728

(b) For languages not in train set.

Recall	SNR					
Models	**−15**	**−10**	**−5**	**0**	**5**	**10**
CLDNN	**0.313**	**0.316**	**0.471**	**0.665**	0.774	0.818
CNNTD-SF	0.033	0.080	0.167	0.467	0.763	**0.834**
CNNTD-S	0.016	0.036	0.102	0.373	0.706	0.807
CNNTD-MS	0.021	0.047	0.136	0.491	**0.775**	0.830
GRU-SF	0.001	0.008	0.160	0.302	0.700	0.819
GRU-S	0.010	0.022	0.071	0.264	0.586	0.726
GRU-MS	0.001	0.008	0.050	0.292	0.599	0.743
STNET	0.002	0.018	0.064	0.283	0.612	0.753
TCN	0.008	0.018	0.080	0.401	0.719	0.805

Table 18
Mean Recall across languages for white noise added to clean MUSAN-librivox clips.

(a) For languages in train set.

Recall	SNR					
Models	**−15**	**−10**	**−5**	**0**	**5**	**10**
CLDNN	**0.047**	0.004	0.010	0.213	0.596	0.767
CNNTD-SF	0.002	0.041	0.333	0.722	**0.861**	**0.887**
CNNTD-S	0.000	0.010	0.237	0.663	0.825	0.867
CNNTD-MS	0.001	0.007	0.144	0.598	0.796	0.857
GRU-SF	0.000	0.007	0.171	0.216	0.594	0.802
GRU-S	0.000	**0.018**	0.176	0.584	0.777	0.814
GRU-MS	0.000	0.007	0.123	0.505	0.721	0.759
STNET	0.018	0.087	**0.501**	**0.797**	0.829	0.843
TCN	0.000	0.004	0.074	0.418	0.684	0.774

(b) For languages not in train set.

Recall	SNR					
Models	**−15**	**−10**	**−5**	**0**	**5**	**10**
CLDNN	**0.037**	0.004	0.016	0.248	0.654	0.798
CNNTD-SF	0.003	0.047	0.400	0.768	**0.859**	**0.865**
CNNTD-S	0.000	0.010	0.297	0.721	0.836	0.852
CNNTD-MS	0.000	0.009	0.175	0.685	0.845	0.858
GRU-SF	0.000	0.006	0.160	0.302	0.700	0.819
GRU-S	0.001	0.020	0.215	0.663	0.818	0.832
GRU-MS	0.000	0.006	0.161	0.582	0.772	0.810
STNET	0.014	**0.099**	**0.545**	**0.808**	0.835	0.833
TCN	0.000	0.003	0.091	0.521	0.780	0.827

Overall, CNNTD-SF is the best performing model in presence of noise at high SNRs, which are typically present in DEC. The instantaneous frequency feature concatenated with magnitude spectrogram allows the model to differentiate non-speech sounds with speech sounds at high SNRs. Moreover, CLDNN is the most robust model in presence of noise due to its noise robust features. We re-establish through experimental validation that CLDNN is able to learn noise robust features through its CNN backbone using raw audio features as described in [3]. At low SNRs of −10 and −15, no model is able to distinguish speech from noise as evident from the recall rate described in the tables (12–18). Hence, −10 SNR acts as a limit of sequence based deep architectures for speech detection. Such high noise cases are not commonly present in the movies and TV shows, as the presence of low SNRs during dialogues in movies hampers the listening experience. Therefore, no model is able to identify speech in such low SNR regimes. The GRU based networks are not robust to input noise despite having same input feature representation as CNNTD variants. This is because, CNN based architectures are better feature learners compared to GRU based networks. They are able to detect short utterances within a segment, and thus are robust to silence/noise within these segments as outlined in [1]. In terms of language sensitivity at high SNRs (10 and 5), all the sequence based models retain their respective recall rates for the languages present and absent in training set (DEC-1100), thereby, can transfer their learning to identify speech in languages absent from train set.

5. Conclusions

This paper presents a comprehensive empirical analysis of various neural models for VAD applied to DEC. A robust and language agnostic VAD is crucial for subsequent downstream tasks such as creation and validation of DEC subtitles and metadata. Previous standard VAD approaches are not applicable to DEC due to its noisy structure spanning multiple languages. Hence, we used a proprietary corpus of 1100 DEC videos spanning 450 h of content, 5 + genres and 9 languages, making it the largest DEC dataset used till date to develop VAD models. In this work, we made the following observations:

(a) Deep architectures are robust to label noises and results in AUCs up to $\sim$ 0.95 when tested on clean dataset. (b) Using a larger labelled training corpus substantially increases the neural VAD model's TPR by 1.3% (relative) from the previous state-of-the-art results on AVA dataset. (c) Various sequence based deep neural VADs results in similar levels of competence in terms of language agnostic behaviour. These models were trained on a set of 9 languages with highly skewed distribution, containing English as the major language. However, our experiments highlighted in the Tables 9 and 12–18 demonstrate the models are able to generalize even for languages which are not a part of training set. This is attributed to the language agnostic features learned by sequence based models. (d) We tested the models on speech only clips mixed with seven different noise types across 6 SNRs. We observed that across several feature representations for the sequence based deep networks, CNNTD-SF and CLDNN outperforms other methods. For high SNRs, CNNTD-SF is the better model. Such high SNRs are usually associated with dialogues in movies and TV shows. However, CLDNN is the most noise robust model as it is able to identify speech at medium SNRs. No model is able to identify speech at low SNRs of −10 and −15. Practically, such low and medium SNR cases do not occur in DEC, as it greatly hampers listening experience of the viewer. Therefore, CNNTD-SF is our choice of deep architecture to be used as VAD for DEC. e) We found the CNN based architectures with large number of parameters ($>$ 2 M parameters) are least robust.

The GRU and CNN based architectures do not perform well under medium to low SNR setting. This can be further improved using data augmentation techniques such as Mixup [94] and adding several explicit noise types during training time. Currently, the training set is highly biased towards English and other languages are limited in samples, therefore, language sensitivity can also be improved by increasing language coverage in the training set. Moreover, we also plan to experiment by concatenating spectrograms with other feature representations such as MFCCs and raw audio waveforms. Wav2vec2.0 is a recent architecture that has emerged as leading network of choice for audio feature extraction [95]. We also plan to experiment by using audio embeddings extracted from wav2vec2.0 as input to these CNN and GRU based architectures. In the future, we shall also extend this work into, a) Multiclass and multilabel setting like *caption classification* (sound categorization which includes traffic noises, human sounds such as grunts, sighs, cough, sneeze etc. and several animal, machine, weapons and environmental sounds) which finds its application in creating Subtitles for Deaf and Hard of Hearing (SDH), b) We are also working in the direction of using this language agnostic and robust VAD to create a model for automatic subtitle drift localization and correction and c) Identifying the time stamps associated with missing subtitles.

CRediT authorship contribution statement

Mayank Sharma: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing - original draft, Visualization. **Sandeep Joshi:** Conceptualization, Methodology, Software, Investigation, Data curation, Writing - review & editing. **Tamojit Chatterjee:** Investigation, Data curation, Writing - review & editing. **Raffay Hamid:** Conceptualization, Methodology, Writing - review & editing, Supervision.

Declaration of Competing Interest

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.

Acknowledgements

The authors would like to thank Mr. Keshav Kumar for his key insights into the problem space and his help with the manuscript. The authors would also like to thank Amazon Studios for providing the high quality dataset.

References

[1] R. Hebbar, K. Somandepalli, S. Narayanan, Robust speech activity detection in movie audio: Data resources and experimental evaluation, in: ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2019, pp. 4105–4109.

[2] S. Chaudhuri, J. Roth, D.P. Ellis, A. Gallagher, L. Kaver, R. Marvin, C. Pantofaru, N. Reale, L.G. Reid, K. Wilson, et al., Ava-speech: A densely labeled dataset of speech activity in movies, arXiv preprint arXiv:1808.00606.

[3] R. Zazo, T.N. Sainath, G. Simko, C. Parada, Feature learning with raw-waveform cldnns for voice activity detection, in: Interspeech 2016, 17th Annual Conference of the International Speech Communication Association, San Francisco, CA, USA, September 8–12, 2016, 2016, pp. 3668–3672. doi:10.21437/Interspeech.2016-268. url:https://doi.org/10.21437/Interspeech.2016-268.

[4] M. Kotti, E. Benetos, C. Kotropoulos, I. Pitas, A neural network approach to audio-assisted movie dialogue detection, Neurocomputing 71 (1–3) (2007) 157–166, https://doi.org/10.1016/j.neucom.2007.08.006.

[5] Y. Wang, G. Zhao, K. Xiong, G. Shi, Y. Zhang, Multi-scale and single-scale fully convolutional networks for sound event detection, Neurocomputing 421 (2021) 51–65, https://doi.org/10.1016/j.neucom.2020.09.038.
[6] I. Ozer, Z. Ozer, O. Findik, Noise robust sound event classification with convolutional neural network, Neurocomputing 272 (2018) 505–512, https://doi.org/10.1016/j.neucom.2017.07.021.
[7] R.V. Sharan, T.J. Moir, An overview of applications and advancements in automatic sound recognition, Neurocomputing 200 (2016) 22–34, https://doi.org/10.1016/j.neucom.2016.03.020.
[8] F. Li, M. Akagi, Blind monaural singing voice separation using rank-1 constraint robust principal component analysis and vocal activity detection, Neurocomputing 350 (2019) 44–52, https://doi.org/10.1016/j.neucom.2019.04.030.
[9] Y. Yu, S. Luo, S. Liu, H. Qiao, Y. Liu, L. Feng, Deep attention based music genre classification, Neurocomputing 372 (2020) 84–91, https://doi.org/10.1016/j.neucom.2019.09.054.
[10] L. Chen, Y. Liu, W. Xiao, Y. Wang, H. Xie, Speakergan: Speaker identification with conditional generative adversarial network, Neurocomputing 418 (2020) 211–220, https://doi.org/10.1016/j.neucom.2020.08.040.
[11] T. Bian, F. Chen, L. Xu, Self-attention based speaker recognition using cluster-range loss, Neurocomputing 368 (2019) 59–68, https://doi.org/10.1016/j.neucom.2019.08.046.
[12] Y. Wu, C. Guo, H. Gao, J. Xu, G. Bai, Dilated residual networks with multi-level attention for speaker verification, Neurocomputing 412 (2020) 177–186, https://doi.org/10.1016/j.neucom.2020.06.079.
[13] K. Wang, G. Su, L. Liu, S. Wang, Wavelet packet analysis for speaker-independent emotion recognition, Neurocomputing 398 (2020) 257–264, https://doi.org/10.1016/j.neucom.2020.02.085.
[14] T. Young, V. Pandelea, S. Poria, E. Cambria, Dialogue systems with audio context, Neurocomputing 388 (2020) 102–109, https://doi.org/10.1016/j.neucom.2019.12.126.
[15] J. Yan, W. Zheng, Z. Cui, C. Tang, T. Zhang, Y. Zong, Multi-cue fusion for emotion recognition in the wild, Neurocomputing 309 (2018) 27–35, https://doi.org/10.1016/j.neucom.2018.03.068.
[16] Y. Dong, X. Yang, A hierarchical depression detection model based on vocal and emotional cues, Neurocomputing 441 (2021) 279–290, https://doi.org/10.1016/j.neucom.2021.02.019.
[17] M. Hao, W. Cao, Z. Liu, M. Wu, P. Xiao, Visual-audio emotion recognition based on multi-task and ensemble learning with multiple features, Neurocomputing 391 (2020) 42–51, https://doi.org/10.1016/j.neucom.2020.01.048.
[18] S. Poria, E. Cambria, N. Howard, G. Huang, A. Hussain, Fusing audio, visual and textual clues for sentiment analysis from multimodal content, Neurocomputing 174 (2016) 50–59, https://doi.org/10.1016/j.neucom.2015.01.095.
[19] W. Zhou, Z. Zhu, A new online bayesian NMF based quasi-clean speech reconstruction for non-intrusive voice quality evaluation, Neurocomputing 349 (2019) 261–270, https://doi.org/10.1016/j.neucom.2019.03.051.
[20] E. Trentin, M. Gori, A survey of hybrid ANN/HMM models for automatic speech recognition, Neurocomputing 37 (1–4) (2001) 91–126, https://doi.org/10.1016/S0925-2312(00)00308-8.
[21] S.M. Siniscalchi, D. Yu, L. Deng, C. Lee, Exploiting deep neural networks for detection-based speech recognition, Neurocomputing 106 (2013) 148–157, https://doi.org/10.1016/j.neucom.2012.11.008.
[22] Z. Li, Y. Ming, L. Yang, J. Xue, Mutual-learning sequence-level knowledge distillation for automatic speech recognition, Neurocomputing 428 (2021) 259–267, https://doi.org/10.1016/j.neucom.2020.11.025.
[23] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, F.E. Alsaadi, A survey of deep neural network architectures and their applications, Neurocomputing 234 (2017) 11–26, https://doi.org/10.1016/j.neucom.2016.12.038.
[24] M. Alam, M.D. Samad, L. Vidyaratne, A. Glandon, K.M. Iftekharuddin, Survey on deep neural networks in speech and vision systems, Neurocomputing 417 (2020) 302–321, https://doi.org/10.1016/j.neucom.2020.07.053.
[25] K. Cho, B. Van Merriënboer, D. Bahdanau, Y. Bengio, On the properties of neural machine translation: Encoder-decoder approaches, arXiv preprint arXiv:1409.1259.
[26] F. Eyben, F. Weninger, S. Squartini, B. Schuller, Real-life voice activity detection with lstm recurrent neural networks and an application to hollywood movies, in: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE, 2013, pp. 483–487.
[27] S. Chang, B. Li, G. Simko, T.N. Sainath, A. Tripathi, A. van den Oord, O. Vinyals, Temporal modeling using dilated convolution and gating for voice-activity-detection, in: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada, April 15–20, 2018, 2018, pp. 5549–5553. doi:10.1109/ICASSP.2018.8461921. url:https://doi.org/10.1109/ICASSP.2018.8461921.
[28] Y. Lee, J. Min, D.K. Han, H. Ko, Spectro-temporal attention-based voice activity detection, IEEE Signal Process. Lett. 27 (2020) 131–135, https://doi.org/10.1109/LSP.2019.2959917.
[29] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in Neural Information Processing Systems (2017) 5998–6008.
[30] L. Muda, M. Begam, I. Elamvazuthi, Voice recognition algorithms using mel frequency cepstral coefficient (MFCC) and dynamic time warping (DTW) techniques, CoRR abs/1003.4083. arXiv:1003.4083. url:http://arxiv.org/abs/1003.4083.
[31] B. Lehner, G. Widmer, R. Sonnleitner, Improving voice activity detection in movies, in: INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6–10, 2015, ISCA, 2015, pp. 2942–2946. url:http://www.isca-speech.org/archive/interspeech_2015/i15_2942.html.
[32] B. Lehner, G. Widmer, R. Sonnleitner, On the reduction of false positives in singing voice detection, in: IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2014, Florence, Italy, May 4–9, 2014, 2014, pp. 7480–7484. doi:10.1109/ICASSP.2014.6855054. url:https://doi.org/10.1109/ICASSP.2014.6855054.
[33] D. Snyder, G. Chen, D. Povey, MUSAN: A music, speech, and noise corpus, CoRR abs/1510.08484. arXiv:1510.08484. url:http://arxiv.org/abs/1510.08484.
[34] J. Sohn, N.S. Kim, W. Sung, A statistical model-based voice activity detection, IEEE Signal Processing Letters 6 (1) (1999) 1–3.
[35] WebRTC VAD, url:https://webrtc.org/.
[36] A. Davis, S. Nordholm, R. Togneri, Statistical voice activity detection using low-variance spectrum estimation and an adaptive threshold, IEEE Trans. Audio, Speech, Language Processing 14 (2) (2006) 412–424.
[37] S.G. Tanyer, H. Ozer, Voice activity detection in nonstationary noise, IEEE Trans. Speech Audio Processing 8 (4) (2000) 478–482.
[38] K.-H. Woo, T.-Y. Yang, K.-J. Park, C. Lee, Robust voice activity detection algorithm for estimating noise spectrum, Electronics Letters 36 (2) (2000) 180–181.
[39] S. Mousazadeh, I. Cohen, Ar-garch in presence of noise: Parameter estimation and its application to voice activity detection, IEEE Trans. Audio, Speech, Language Process. 19 (4) (2011) 916–926.
[40] I. Yoo, H. Lim, D. Yook, Formant-based robust voice activity detection, IEEE ACM Trans, Audio Speech Lang. Process. 23 (12) (2015) 2238–2245, https://doi.org/10.1109/TASLP.2015.2476762.
[41] J. Wu, X. Zhang, Maximum margin clustering based statistical VAD with multiple observation compound feature, IEEE Signal Process. Lett. 18 (5) (2011) 283–286, https://doi.org/10.1109/LSP.2011.2119482.
[42] Y. Suh, H. Kim, Multiple acoustic model-based discriminative likelihood ratio weighting for voice activity detection, IEEE Signal Process. Lett. 19 (8) (2012) 507–510, https://doi.org/10.1109/LSP.2012.2204978.
[43] E. Nemer, R.A. Goubran, S. Mahmoud, Robust voice activity detection using higher-order statistics in the LPC residual domain, IEEE Trans. Speech Audio Process. 9 (3) (2001) 217–231, https://doi.org/10.1109/89.905996.
[44] L. Muda, M. Begam, I. Elamvazuthi, Voice recognition algorithms using mel frequency cepstral coefficient (mfcc) and dynamic time warping (dtw) techniques, arXiv preprint arXiv:1003.4083.
[45] A. Misra, Speech/nonspeech segmentation in web videos, in: in: Thirteenth Annual Conference of the International Speech Communication Association, 2012.
[46] S. Mousazadeh, I. Cohen, Voice activity detection in presence of transient noise using spectral clustering, IEEE Trans. Speech Audio Process. 21 (6) (2013) 1261–1271, https://doi.org/10.1109/TASL.2013.2248717.
[47] T. Kinnunen, E. Chernenko, M. Tuononen, P. Fränti, H. Li, Voice activity detection using mfcc features and support vector machine, in: Int. Conf. on Speech and Computer (SPECOM07), Moscow, Russia, Vol. 2, 2007, pp. 556–561.
[48] C.E. Galván-Tejada, J.I. Galván-Tejada, J.M. Celaya-Padilla, J.R. Delgado-Contreras, R. Magallanes-Quintanar, M.L. Martinez-Fierro, I. Garza-Veloz, Y. López-Hernández, H. Gamboa-Rosales, An analysis of audio features to develop a human activity recognition model using genetic algorithms, random forests, and neural networks, Mobile Information Systems (2016).
[49] N. Ryant, M. Liberman, J. Yuan, Speech activity detection on youtube using deep neural networks., in: INTERSPEECH, Lyon, France, 2013, pp. 728–731.
[50] I. Tashev, S. Mirsamadi, Dnn-based causal voice activity detector, in: Information Theory and Applications Workshop, 2016.
[51] T. Drugman, Y. Stylianou, Y. Kida, M. Akamine, Voice activity detection: Merging source and filter-based information, IEEE Signal Process. Lett. 23 (2) (2016) 252–256, https://doi.org/10.1109/LSP.2015.2495219.
[52] K. Phapatanaburi, L. Wang, Z. Oo, W. Li, S. Nakagawa, M. Iwahashi, Noise robust voice activity detection using joint phase and magnitude based feature enhancement, J. Ambient Intell. Humaniz. Comput. 8 (6) (2017) 845–859, https://doi.org/10.1007/s12652-017-0482-8.
[53] X. Zhang, J. Wu, Deep belief networks based voice activity detection, IEEE Trans. Speech Audio Process. 21 (4) (2013) 697–710, https://doi.org/10.1109/TASL.2012.2229986.
[54] C. Gao, G. Saikumar, S. Khanwalkar, A. Herscovici, A. Kumar, A. Srivastava, P. Natarajan, Online speech activity detection in broadcast news, in: in: Twelfth Annual Conference of the International Speech Communication Association, 2011.
[55] P. Teng, Y. Jia, Voice activity detection via noise reducing using non-negative sparse coding, IEEE Signal Process. Lett. 20 (5) (2013) 475–478, https://doi.org/10.1109/LSP.2013.2252615.
[56] M. Benatan, K. Ng, Cross-covariance-based features for speech classification in film audio, J. Vis. Lang. Comput. 31 (2015) 215–221, https://doi.org/10.1016/j.jvlc.2015.10.011.
[57] T. Ng, B. Zhang, L. Nguyen, S. Matsoukas, X. Zhou, N. Mesgarani, K. Veselỳ, P. Matějka, Developing a speech activity detection system for the darpa rats program, in: Thirteenth Annual Conference of the International Speech Communication Association, 2012.
[58] S. Tong, H. Gu, K. Yu, A comparative study of robustness of deep learning approaches for vad, in: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2016, pp. 5695–5699.
[59] L. Mateju, P. Cerva, J. Zdánský, J. Málek, Speech activity detection in online broadcast transcription using deep neural networks and weighted finite state transducers, in: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2017, New Orleans, LA, USA, March 5–9, 2017, 2017,

pp. 5460–5464. doi:10.1109/ICASSP.2017.7953200. url:https://doi.org/10.1109/ICASSP.2017.7953200.
[60] I. Jang, C. Ahn, J. Seo, Y. Jang, Enhanced feature extraction for speech detection in media audio, in: Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20–24, 2017, 2017, pp. 479–483. url:http://www.isca-speech.org/archive/Interspeech_2017/abstracts/0792.html.
[61] X. Zhang, D. Wang, Boosting contextual information for deep neural network based voice activity detection, IEEE ACM Trans, Audio Speech Lang. Process. 24 (2) (2016) 252–264, https://doi.org/10.1109/TASLP.2015.2505415.
[62] I. Hwang, H. Park, J. Chang, Ensemble of deep neural networks using acoustic environment classification for statistical model-based voice activity detection, Comput. Speech Lang. 38 (2016) 1–12, https://doi.org/10.1016/j.csl.2015.11.003.
[63] T.G. Kang, N.S. Kim, Dnn-based voice activity detection with multi-task learning, IEICE Trans, Inf. Syst. 99-D (2) (2016) 550–553, https://doi.org/10.1587/transinf.2015EDL8168.
[64] K. Greff, R.K. Srivastava, J. Koutník, B.R. Steunebrink, J. Schmidhuber, Lstm: A search space odyssey, IEEE Trans. Neural Networks Learn. Syst. 28 (10) (2017) 2222–2232.
[65] R. Zazo Candil, T.N. Sainath, G. Simko, C. Parada, Feature learning with raw-waveform cldnns for voice activity detection.
[66] T. Hughes, K. Mierle, Recurrent neural networks for voice activity detection, in: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, 2013, pp. 7378–7382.
[67] J. Kim, J. Kim, S. Lee, J. Park, M. Hahn, Vowel based voice activity detection with LSTM recurrent neural network, in: Proceedings of the 8th International Conference on Signal Processing Systems, ICSPS 2016, Auckland, New Zealand, November 21–24, 2016, 2016, pp. 134–137. doi:10.1145/3015166.3015207. url:https://doi.org/10.1145/3015166.3015207.
[68] P. Sertsi, S. Boonkla, V. Chunwijitra, N. Kurpukdee, C. Wutiwiwatchai, Robust voice activity detection based on LSTM recurrent neural networks and modulation spectrum, in: 2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA ASC 2017, Kuala Lumpur, Malaysia, December 12–15, 2017, 2017, pp. 342–346. doi:10.1109/APSIPA.2017.8282048. url:https://doi.org/10.1109/APSIPA.2017.8282048.
[69] B. Lehner, G. Widmer, S. Böck, A low-latency, real-time-capable singing voice detection method with LSTM recurrent neural networks, in: 23rd European Signal Processing Conference, EUSIPCO 2015, Nice, France, August 31 - September 4, 2015, 2015, pp. 21–25. doi:10.1109/EUSIPCO.2015.7362337. url:https://doi.org/10.1109/EUSIPCO.2015.7362337.
[70] T.N. Sainath, R.J. Weiss, A. Senior, K.W. Wilson, O. Vinyals, Learning the speech front-end with raw waveform cldnns, in: Sixteenth Annual Conference of the International Speech Communication Association, 2015.
[71] G. Ferroni, R. Bonfigli, E. Principi, S. Squartini, F. Piazza, A deep neural network approach for voice activity detection in multi-room domestic scenarios, in: 2015 International Joint Conference on Neural Networks (IJCNN), IEEE, 2015, pp. 1–8.
[72] S. Thomas, S. Ganapathy, G. Saon, H. Soltau, Analyzing convolutional neural networks for speech activity detection in mismatched acoustic conditions, in: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2014, pp. 2519–2523.
[73] H. Hermansky, Perceptual linear predictive (plp) analysis of speech, J. Acoust. Soc. Am. 87 (4) (1990) 1738–1752.
[74] V. Zue, S. Seneff, J. Glass, Speech database development at mit: Timit and beyond, Speech Commun. 9 (4) (1990) 351–356.
[75] E. Fosler-Lussier, L. Dilley, N. Tyson, M. Pitt, The buckeye corpus of speech: updates and enhancements, in: Eighth Annual Conference of the International Speech Communication Association, 2007.
[76] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556.
[77] J.S. Garofolo, L.F. Lamel, W.M. Fisher, J.G. Fiscus, D.S. Pallett, Darpa timit acoustic-phonetic continous speech corpus cd-rom. nist speech disc 1-1.1, NASA STI/Recon technical report n 93 (1993) 27403. doi:https://doi.org/10.35111/17gk-bn40.
[78] A. Varga, H.J.M. Steeneken, 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) (1993) 247–251. doi:10.1016/0167-6393(93)90095-3. url:https://doi.org/10.1016/0167-6393(93)90095-3.
[79] List of ISO 639–1 codes, url:https://en.wikipedia.org/wiki/List_of_ISO_639-1_codes.
[80] J.F. Gemmeke, D.P.W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R.C. Moore, M. Plakal, M. Ritter, Audio set: An ontology and human-labeled dataset for audio events, in: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2017, New Orleans, LA, USA, March 5–9, 2017, IEEE, 2017, pp. 776–780. doi:10.1109/ICASSP.2017.7952261. url:https://doi.org/10.1109/ICASSP.2017.7952261.
[81] P. Gupta, M. Sharma, K. Pitale, K. Kumar, Problems with automating translation of movie/tv show subtitles, CoRR abs/1909.05362. arXiv:1909.05362. url:http://arxiv.org/abs/1909.05362.
[82] M. Huzaifah, Comparison of time-frequency representations for environmental sound classification using convolutional neural networks, CoRR abs/1706.07156. arXiv:1706.07156. url:http://arxiv.org/abs/1706.07156.
[83] H. Lee, P.T. Pham, Y. Largman, A.Y. Ng, Unsupervised feature learning for audio classification using convolutional deep belief networks, in: Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009. Proceedings of a meeting held 7-10 December 2009, Vancouver, British Columbia, Canada, 2009, pp. 1096–1104. url:http://papers.nips.cc/paper/3674-unsupervised-feature-learning-for-audio-classification-using-convolutional-deep-belief-networks.
[84] T. Kim, J. Lee, J. Nam, Comparison and analysis of samplecnn architectures for audio classification, IEEE J. Sel. Top. Signal Process. 13 (2) (2019) 285–297, https://doi.org/10.1109/JSTSP.2019.2909479.
[85] F. Auger, P. Flandrin, Y. Lin, S. McLaughlin, S. Meignen, T. Oberlin, H. Wu, Time-frequency reassignment and synchrosqueezing: An overview, IEEE Signal Process. Mag. 30 (6) (2013) 32–41, https://doi.org/10.1109/MSP.2013.2265316.
[86] L. Wang, K. Phapatanaburi, Z. Oo, S. Nakagawa, M. Iwahashi, J. Dang, Phase aware deep neural network for noise robust voice activity detection, in: 2017 IEEE International Conference on Multimedia and Expo, ICME 2017, Hong Kong, China, July 10–14, 2017, 2017, pp. 1087–1092. doi:10.1109/ICME.2017.8019414. url:https://doi.org/10.1109/ICME.2017.8019414.
[87] I. McCowan, D. Dean, M. McLaren, R. Vogt, S. Sridharan, The delta-phase spectrum with application to voice activity detection and speaker recognition, IEEE Trans. Speech Audio Process. 19 (7) (2011) 2026–2038, https://doi.org/10.1109/TASL.2011.2109379.
[88] B. Lehner, J. Schlüter, G. Widmer, Online, loudness-invariant vocal detection in mixed music signals, IEEE ACM Trans, Audio Speech Lang. Process. 26 (8) (2018) 1369–1380, https://doi.org/10.1109/TASLP.2018.2825108.
[89] K. Lee, K. Choi, J. Nam, Revisiting singing voice detection: A quantitative review and the future outlook, in: Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018, Paris, France, September 23–27, 2018, 2018, pp. 506–513. url:http://ismir2018.ircam.fr/doc/pdfs/38_Paper.pdf.
[90] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980.
[91] Z. Fan, Z. Bai, X. Zhang, S. Rahardja, J. Chen, AUC optimization for deep learning based voice activity detection, in: IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2019, Brighton, United Kingdom, May 12–17, 2019, 2019, pp. 6760–6764. doi:10.1109/ICASSP.2019.8682803. url:https://doi.org/10.1109/ICASSP.2019.8682803.
[92] X. Sun, W. Xu, Fast implementation of delong's algorithm for comparing the areas under correlated receiver operating characteristic curves, IEEE Signal Process. Lett. 21 (11) (2014) 1389–1393, https://doi.org/10.1109/LSP.2014.2337313.
[93] E.R. DeLong, D.M. DeLong, D.L. Clarke-Pearson, Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach, Biometrics (1988) 837–845.
[94] H. Zhang, M. Cissé, Y.N. Dauphin, D. Lopez-Paz, mixup: Beyond empirical risk minimization, in: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings, OpenReview.net, 2018. url:https://openreview.net/forum?id=r1Ddp1-Rb.
[95] A. Baevski, Y. Zhou, A. Mohamed, M. Auli, wav2vec 2.0: A framework for self-supervised learning of speech representations, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6–12, 2020, virtual, 2020. url:https://proceedings.neurips.cc/paper/2020/hash/92d1e1eb1cd6f9fba3227870bb6d7f07-Abstract.html.

Mayank Sharma received the B.Tech degree in electrical engineering from the Indian Institute of Technology (IIT), Kharagpur, West Bengal, India, and the PhD degree from the Department of Electrical Engineering, Indian Institute of Technology (IIT), Delhi, India. He is working as an applied scientist with Amazon. His research interests include large scale machine learning, audio signal processing, NLP and deep learning. He has published several papers in international journals and conferences and hold 2 US patents under his belt.

Sandeep S. Joshi received his B.E. in computer engineering from Univ. of Pune, India. He has about 18 years of experience across distributed systems, machine learning, operating systems and databases. He is currently working part of the core team at Kognitos, Inc. which is building an innovative NLP-based solution for business process automation.

Tamojit Chatterjee received his B.Tech degree from Kalyani Government Engineering college in 2014 and masters form IIT-Bombay in 2017. He worked as an applied scientist at Prime Video International Expansion Group at Amazon till 2021. His research interest includes audio signal processing, deep Learning and natural language processing.

Raffay Hamid (M'15) received the Ph.D. degree from the Georgia Institute of Technology, Atlanta, GA, USA, in 2008. He has lead several web-scale systems with visual intelligence that have been used by Disney, ESPN, and eBay. He has also served as a Technical Advisor to large-scale machine learning companies, helping them build scalable computational platforms with vision capabilities. He is currently leading a team of scientists and engineers at Amazon Prime Video, Seattle, WA, USA, working on developing large scale video understanding systems. He is a Researcher in computer vision and machine learning, with close to 30 research papers and multiple patents under his belt. His work has been featured on BBC, Time, MIT Technology Review, and Techcrunch, among several other forums.