International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 05 | May - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Objective and Subjective Evaluation of Speech Enhancement Methods in the UDASE task of the 7th CHIME Challenge

Gongalla Mayuri[1] ,ECE, Institute of Aeronautical Engineering,Hyderabad,India
22951a0495@iare.ac.in
Dr. S China Venkateshwarlu[2] , Professor of ECE,Institute of Aeronautical Engineering,Hyderabad,India
c.venkateshwarlu@iare.ac.in
Dr.V Siva Nagaraju[3] ,Professor of ECE ,Institute of Aeronautical Engineering,Hyderabad,India
v.sivanagaraju@iare.ac.in

ABSTRACT

Supervised models for speech enhancement are trained using artificially generated mixtures of clean speech and noise signals. However, the synthetic training conditions may not accurately reflect real-world conditions encountered during testing. This discrepancy can result in poor performance when the test domain significantly differs from the synthetic training domain. To tackle this issue, the UDASE task of the 7th CHiME challenge aimed to leverage real-world noisy speech recordings from the test domain for unsupervised domain adaptation of speech enhancement models. Specifically, this test domain corresponds to the CHiME-5 dataset, characterized by real multi-speaker and conversational speech recordings made in noisy and reverberant domestic environments, for which ground-truth clean speech signals are not available. In this paper, we present the objective and subjective evaluations of the systems that were submitted to the CHiME-7 UDASE task, and we provide an analysis of the results. This analysis reveals a limited correlation between subjective ratings and several supervised nonintrusive performance metrics recently proposed for speech enhancement. Conversely, the results suggest that more traditional intrusive objective metrics can be used for in-domain performance evaluation using the reverberant LibriCHiME-5 dataset developed for the challenge. The subjective evaluation indicates that all systems successfully reduced the background noise, but always at the expense of increased distortion. Out of the four speech enhancement methods evaluated subjectively, only one demonstrated an improvement in overall quality compared to the unprocessed noisy speech, highlighting the difficulty of the task. The tools and audio material created for the CHiME-7 UDASE task are shared with the community

Key Terms: Speech Enhancement, CHIME-7 Challenge ,Deep Neural Networks,
UDASE Task, Audio Restoration, Subjective Evaluation, objective Evaluation

Chapter I

Introduction

The speech enhancement task consists of improving the quality and intelligibility of a degraded speech signal recording. One approach to achieve this is through noise suppression algorithms, which aim to estimate the clean speech signal by removing the additive background noise in the recording. Over the past 50 years, traditional signal-processing-based speech enhancement algorithms (Boll, 1979; Lim & Oppenheim, 1979; Ephraim & Malah, 1984; Martin, 2005; Loizou, 2013) have been progressively outperformed by data-driven approaches using hidden Markov models (HMMs)

In recent years, significant research efforts have focused on the three core components of supervised speech enhancement: the model, the evaluation metric, and the labeled dataset. These efforts have led to remarkable advancements, as demonstrated by works such as Weninger et al. (2015), Fu et al. (2017, 2019, 2021), Pascual et al. (2017), Choi et al. (2018), Zhao et al. (2018), Defossez et al. (2020), Cosentino et al. (2020), Hao et al. (2021), Pandey & Wang (2021), and Richter et al. (2023).

However, the supervised learning paradigm faces several limitations when applied to speech enhancement. First, constructing a synthetic dataset that accurately represents realistic noisy speech mixtures is challenging and demands

substantial engineering effort. Second, supervised models typically perform well only when the acoustic conditions at test time closely match those seen during training. This is difficult to ensure due to the wide variability in real-world acoustic environments—including differences in noise type, signal-to-noise ratio, recording equipment, speaker-to-microphone distance, orientation, reverberation, and more.

Consequently, the performance of supervised speech enhancement systems can degrade significantly when there is a mismatch between training and testing conditions (Pandey & Wang, 2020; Bie et al., 2022; Richter et al., 2023; Gonzalez et al., 2023). Moreover, when the test domain differs from the synthetic training domain, adapting the model typically requires recreating the training dataset and retraining the system—an approach that is both time-consuming and computationally expensive.

A more scalable and effective alternative would be to enable models to automatically adapt to real, unlabeled noisy speech recordings, thereby eliminating the dependency on ground-truth clean speech signals.

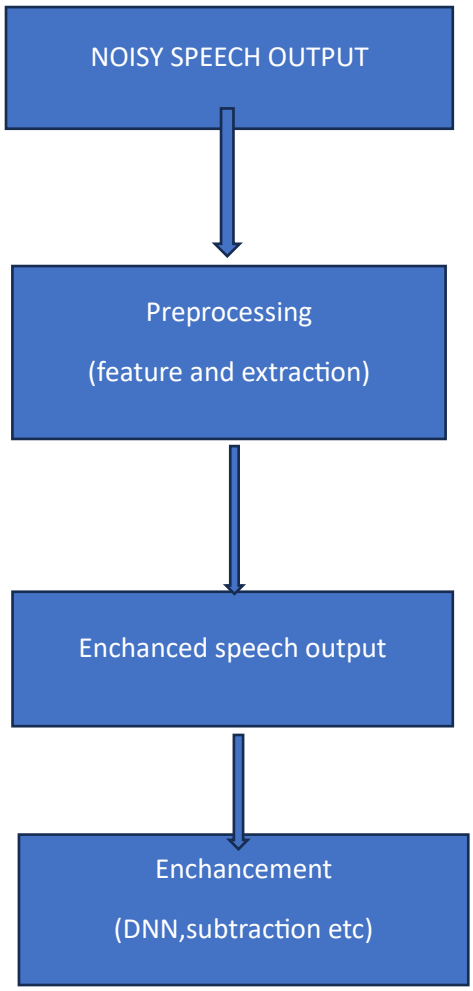
TABLE 1.1: LITERARURE SURVEY

Author/Year	Methodology	Summary	Remarks
Haoyang Li, Yuchen Hu, Chen Chen, Eng Siong Chng ,2024	The study investigates the use of **Kolmogorov-Arnold Networks (KAN)** for speech enhancement by replacing traditional activation functions with learnable KAN activations. Two variants, **GR-KAN** (using rational functions in Demucs and MP-SENet) and **RBF-KAN** (using radial basis functions in MP-SENet's decoder), are tested on the **VoiceBank-DEMAND** dataset. Evaluated using **PESQ** and other metrics, the models with KAN activations show improved speech quality with minimal impact on model size and computational cost, highlighting KAN's potential for advancing speech enhancement.	This work investigates KAN, a competitive alternative to MLP on these domain, by adapting 2novel KAN variants based on rational function and radial basis function to existing SE solutions.	The study effectively demonstrates the potentialof **Kolmogorov-Arnold Networks (KAN)** in speech enhancement by introducing learnable activation functions that improve speech quality. The proposed **GR-KAN and RBF-KAN** variants show promising results with minimal computational overhead. However, further research is needed to evaluate their scalability across different datasets and real-world applications
Kunpeng Xu, Lifei Chen, Shengrui Wang,2024	It proposed two variants: **T-KAN**, designed to detect concept drift and explain nonlinear relationships in univariate time series through symbolic regression, and **MT-KAN**, aimed at improving predictive performance by uncovering complex interdependencies in multivariate time series. Experiments demonstrate that	The study introduces **Kolmogorov-Arnold Networks (KAN)**, replacing traditional weights with **spline-parametrized functions** to improve interpretability and predictive accuracy in time series analysis. **T-KAN** detects concept drift and explains	The introduction of **T-KAN and MT-KAN** provides a novel approach to handling nonlinear dependencies and concept drift. However, further research is needed to assess scalability across diverse real-world datasets and computational efficiency in large-scale applications.

	both models outperform traditional methods in forecasting tasks, achieving higher accuracy and better interpretability.	nonlinear patterns in **univariate time series**, while **MT-KAN** enhances forecasting by capturing complex dependencies in **multivariate time series**, outperforming traditional methods.	4o
Alexander Dylan Bodner, Jack NatanSpolski , 2024	The study introduces **Convolutional Kolmogorov-Arnold Networks (Convolutional KANs)**, which integrate **learnable non-linear activation functions** into convolutional layers. Instead of traditional linear transformations, each pixel undergoes a unique **spline-parametrized activation**, enhancing flexibility. Tested on **MNIST and Fashion-MNIST**, Convolutional KANs achieve similar accuracy to standard CNNs while using **half the parameters**, making them a more **efficient alternative** for image classification.	**Efficient & Accurate:** Convolutional KANs match CNN accuracy while using **half the parameters**. **Flexible & Scalable:** Learnable non-linear activations enhance **model flexibility**, needing further study for **larger datasets**.	The current implementation of KANs with B-Splines is considerably slow due to its impossibility of being GPU parallelizable, making it very difficult to apply KANs in real world problems. Many authors are working on solutions to this by replacing B-Splines by other function approximators, such as Radial Basis Function ??, which are GPU parallelizable and open up a wide range of possibilities with KANs.
Wei-Lun Chen, Yu-Wen Chen, 2023.	The study proposes integrating pre-trained Speech Enhancement (SE) and Automatic Speech Recognition (ASR) models using a lightweight bridge module, which refines SE outputs to better match ASR input requirements. The observation addition technique further improves recognition by combining enhanced speech with the original noisy input. This method enhances ASR robustness in noisy environments without requiring fine-tuning, making it adaptable for real-world applications.	Enhanced ASR Accuracy: The bridge module improves speech recognition in noisy environments by refining SE outputs. Scalable & Adaptable: No fine-tuning needed, making it practical for real-world applications.	The approach may not generalize well across **highly diverse datasets and real-world noise conditions**, requiring further validation. The integration of a **bridge module** introduces additional processing, which could impact **real-time speech recognition performance**.

George Close , Thomas Hain , 2022	The study extends **MetricGAN+** by introducing a **de-generator** network to improve speech enhancement on **unseen noise data**. The de-generator generates diverse perceptual metric scores, training the **predictor network** for better generalization. This approach helps **reduce overfitting** and enhances robustness to unseen conditions. **Evaluated on the VoiceBank-DEMAND dataset**, it shows a **3.8% PESQ improvement** (from 3.05 to 3.22). Results confirm **better noise reduction and speech quality** in unpredictable environments.	**Enhanced Robustness:** The de-generator improves **generalization to unseen noise**, reducing overfitting. **Improved Speech Quality:** Achieves **3.8% PESQ improvement**, ensuring better **noise reduction and clarity**.	**Generalization Scope:** While it improves robustness, performance on **extreme or highly variable noise types** may still be limited. **Increased Complexity:** The addition of a **de-generator** increases **computational cost** and training time.

1.2 EXISTING BLOCK DIAGRAM



BLOCK DIAGRAM DISCRIPTION

Noisy Speech Input

- The process begins with a speech signal contaminated by background noise.
- This noisy input typically comes from real-world environments.

Preprocessing (e.g., STFT)

- The noisy signal is converted into a time-frequency representation, such as a spectrogram using Short-Time Fourier Transform (STFT).
- This step prepares the data for effective processing by the model.

Trained Enhancement Model

- A pre-trained deep learning model takes the processed input and performs noise suppression.
- The model has been trained on paired examples of clean and noisy speech signals in a supervised learning setup.

Enhanced Speech Output

- The model outputs a denoised speech signal.
- The result aims to improve clarity and intelligibility for human listeners or automatic speech recognition systems.

Usefulness of Each Block Diagram Component

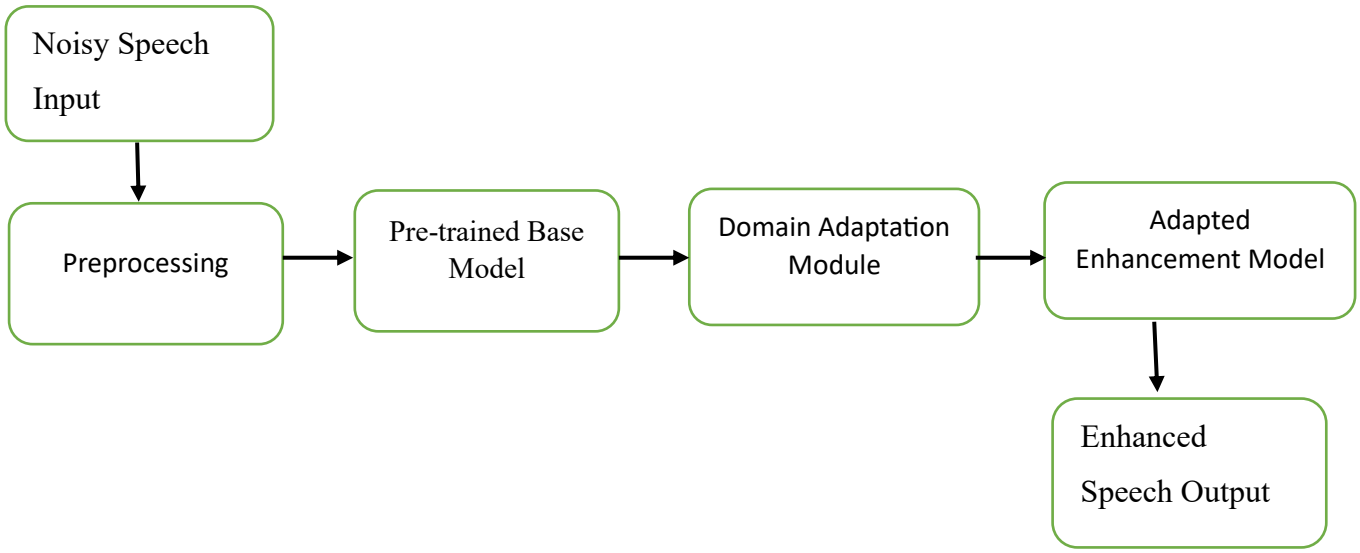
- **Noisy Speech Input**
  - Represents real-world conditions where speech is mixed with various types of background noise.
  - Serves as the starting point for enhancement and performance evaluation.
  - Helps in designing systems that are robust to practical, non-ideal environments.
- **Preprocessing (e.g., STFT)**
  - Converts raw time-domain signals into time-frequency representations for better analysis and processing.
  - Facilitates easier separation of speech and noise components.
  - Enables models to focus on spectral patterns relevant to speech.
- **Trained Enhancement Model**
  - Learns to map noisy inputs to clean outputs using supervised learning.
  - Leverages powerful deep learning architectures (e.g., CNNs, RNNs, Transformers) for accurate enhancement.
  - Improves speech intelligibility and quality, benefiting both human listeners and downstream applications like speech recognition.
- **Enhanced Speech Output**
  - Provides a cleaner, more intelligible version of the input speech.
  - Reduces listener fatigue and improves comprehension in noisy environments.
  - Enhances the performance of voice-controlled systems, hearing aids, telecommunication devices, and speech-based AI assistant

Problem Identification

The supervised speech enhancement process faces several limitations that hinder its effectiveness in real-world scenarios. It relies heavily on large amounts of labeled data, specifically paired clean and noisy speech, which are difficult and costly to collect. Models trained on synthetic datasets often fail to generalize well to real-world conditions, leading to performance drops when faced with unseen noise types, varying recording environments, or different speaker and device characteristics. Additionally, adapting these models to new domains requires significant computational resources and retraining efforts, making the process time-consuming and inefficient. Their inability to dynamically adjust to changing acoustic conditions further limits their practicality, highlighting the need for more flexible, adaptive, and label-efficient approaches.

CHAPTER II : PROPOSED METHODOLOGY WITH BLOCK DIAGRAM

PROPOSED BLOCK DIAGRAM



Description of the Proposed Block Diagram

The proposed block diagram represents a more flexible and robust speech enhancement system designed to overcome the limitations of traditional supervised approaches1. Fully Connected Layer

1.Noisy Speech Input (Real-World):

- Captures speech in natural, uncontrolled environments with background noise.

. May include various noise types (traffic, crowd, machinery, etc.).

. Lacks paired clean references, making supervised learning difficult.

- Represents the real deployment scenario that the model must handle.

2. Preprocessing (e.g., STFT, Feature Extraction):

- Converts raw waveform into time-frequency representations like spectrograms.
- Enhances important features while reducing irrelevant noise.
- Standardizes input format for consistency across domains.
- Enables the model to better identify speech patterns.

3. Pre-trained Base Model (Supervised or Self-supervised)

- Provides a strong starting point based on general training data.
- May use supervised or self-supervised learning techniques.
- Trained on large-scale datasets to learn basic speech/noise separation.
- Helps reduce the training time and data requirements for new tasks.

4.Domain Adaptation Module / Self-supervised Fine-tuning

- Adapts the base model to new acoustic environments without clean labels.
- Learns domain-specific noise characteristics and speech patterns.
- Uses unlabeled noisy speech for fine-tuning in the target domain.
- Reduces mismatch between training and deployment conditions.

5. Adapted Enhancement Model

- Incorporates domain-specific knowledge from the adaptation stage.
- Offers improved performance in diverse and unseen real-world scenarios.
- Maintains speech intelligibility while suppressing noise effectively.
- Becomes robust to variations in devices, environments, and speakers.

6. Enhanced Speech Output:

- Produces a denoised, clearer, and more intelligible speech signal.
- Enhances listening experience for humans and speech-based systems.
- Useful in applications like hearing aids, telephony, and voice assistants.
- Supports downstream tasks like ASR (automatic speech recognition).

2.1 Technical Specifications on Methodology and Flowchart

Technical Specifications

1. Noisy Speech Input (Real-World)

- Input Format: 16-bit PCM WAV, 16kHz or 48kHz sampling rate
- Signal Type: Single-channel (mono) or multi-channel audio
- Typical Duration: 1–10 seconds per utterance
- Characteristics: Includes diverse background noises (stationary, non-stationary)

2. Preprocessing (e.g., STFT, Feature Extraction)

- Transform Method: Short-Time Fourier Transform (STFT)

- Window Size: 25 ms (e.g., 400 samples at 16kHz)
- Hop Length: 10 ms (e.g., 160 samples)
- FFT Points: 512 or 1024
- Output Features:
  - Magnitude Spectrogram
  - Log-Mel Spectrogram (optional)
  - Phase information (optional, used in phase-aware models)

3. Pre-trained Base Model (Supervised or Self-supervised)

- Model Types:
  - Convolutional Neural Network (CNN)
  - Recurrent Neural Network (RNN) / LSTM / GRU
  - Transformer-based Models (e.g., SEFORMER, DPTNet)
- Input Shape: [Batch, Time, Frequency]
- Output: Estimated clean spectrogram or mask
- Training Dataset: Public datasets like VoiceBank+DEMAND, DNS Challenge, LibriSpeech-noisy
- Loss Functions:
  - MSE / L1 loss on spectrograms
  - SI-SNR / SDR loss (time-domain)
  - Perceptual losses (e.g., PESQ, STOI-based)

4. Domain Adaptation Module / Self-supervised Fine-tuning

- Adaptation Strategy:
  - Unsupervised Domain Adaptation (e.g., adversarial, contrastive learning)
  - Pseudo-labeling or consistency regularization
  - Test-time adaptation or online fine-tuning
- Data Requirements: Only unlabeled noisy speech from target domain
- Optimization:
  - Learning rate: 1e-4 to 1e-6
  - Optimizer: Adam or AdamW
- Epochs: 1–5 (lightweight fine-tuning)
- Regularization: Dropout, weight decay, learning rate scheduling

5. Adapted Enhancement Model

- Architecture: Same as base model, but fine-tuned on target domain
- Adaptability: Improved robustness to acoustic mismatch
- Inference Time: Real-time or low-latency (~10–100 ms depending on model)
- Deployment Format: ONNX, TensorRT, or TorchScript for edge devices

6. Enhanced Speech Output

- Output Format: Cleaned waveform (WAV)
- Sampling Rate: Same as input (16kHz or 48kHz)
- Quality Metrics:
  - PESQ (Perceptual Evaluation of Speech Quality)
  - STOI (Short-Time Objective Intelligibility)
  - SI-SNR, SDR
- Use Cases: Hearing aids, ASR, video conferencing, mobile apps

2.2 Software Used

1.Programming Languages

- **Python** – Guido van Rossum developed and released Python, a high-level, interpreted programming language, in 1991. Python is known for its readable syntax and dynamic typing, and it supports a wide range of programming paradigms, including object-oriented, procedural, and functional programming. Because it's open source and has a large standard library, it's perfect for creating applications quickly.
Python is a common language used in data analysis, machine learning, scientific computing, automation, and web development. Its popularity among developers and instructors is due to its cross-platform compatibility and user-friendly layout. The language's name is a nod to the British comedy troupe Monty Python, and it reflects its creator's priority on making programming simple and enjoyable.

- **MATLAB** – Used for speech signal processing, feature extraction, and visualization. Developed by MathWorks for data analysis, numerical computation, and visualization, MATLAB (Matrix Laboratory) is a high-level programming language and environment. It is particularly well-liked in academia, scientific research, and engineering. MATLAB has built-in functions and toolboxes for machine learning, picture processing, control systems, signal processing, linear algebra, and other areas. It has a simple syntax that is designed for matrix and vector operations. Matlab has an interactive interface for creating algorithms, running simulations, and seeing results. Despite being proprietary software, its vast libraries and powerful features have made it a commonplace instrument in many technical fields and engineering processes.

2. Deep Learning Frameworks

- **TensorFlow / Keras** – Python is the language in which Keras, an open source, high-level neural network API, is written. It was first created by François Chollet and offers a simple way to create and train deep learning models. Both convolutional and recurrent networks are supported by Keras, which is made to facilitate quick deep

neural network experiments. It operates on top of low-level frameworks such TensorFlow (default), Theano, and Microsoft CNTK. Due to its straightforward and succinct code, Keras is a great tool for novices and for prototyping complicated processes. Due to its modular architecture and comprehensive documentation, Keras has gained popularity as a deep learning tool in both industry and academia.

- **PyTorch** – Facebook's AI Research lab created PyTorch, an open source machine learning framework. It facilitates the creation, modification, and debugging of deep learning models by providing a flexible and dynamic computational graph. PyTorch is often employed in reinforcement learning, computer vision, and natural language processing. It integrates smoothly with Python for simple model development and supports GPU acceleration for high-performance training. PyTorch offers the torch library for tensors, torch. nn for neural networks, and torch. optim for optimization. PyTorch has gained popularity among researchers and professionals alike because of its simplicity, dynamic graphing, and robust community backing.

3. Signal Processing & Feature Extraction

- Librosa – Python library for speech feature extraction (MFCC, Spectrograms, STFT). SciPy – For implementing signal processing algorithms.
- MATLAB Signal Processing Toolbox – For spectral analysis and denoising techniques.

4. Data Handling & Preprocessing

- **NumPy** – For managing and preprocessing speech datasets.

In Python, numerical calculations rely heavily on NumPy (Numerical Python), an opensource library. In addition to a vast array of mathematical operations that may be performed on them effectively, it also offers support for big, multidimensional arrays and matrices. NumPy is built on C and powers many scientific and data analysis libraries, such as pandas, SciPy, and scikitlearn, with its high-performance computing capabilities. It is capable of producing random numbers, Fourier transformations, linear algebra, and broadcasting. Because of its speed, simplicity, and seamless integration with other Python tools and frameworks, NumPy is widely used in data science, machine learning, and scientific computing.

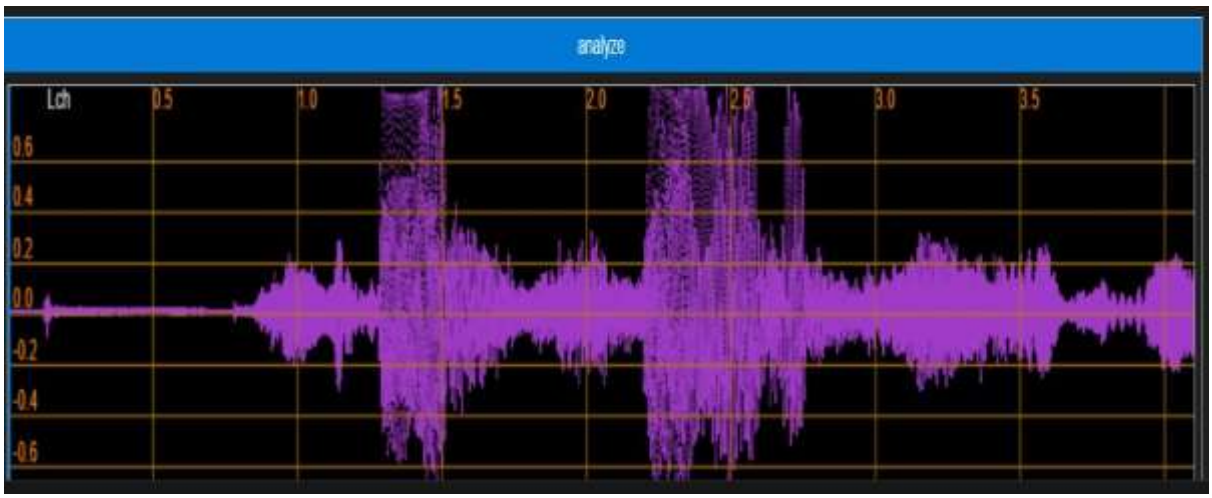
CHAPTER III : RESULTS

The optimal balance between speech enhancement and perceptual quality enhancement is provided by MetricGAN+. Its main benefit over traditional methods such as SEGAN and MetricGAN is that it concentrates on directly optimizing measures linked to how human perceive speech.

Libraries used in this program are:

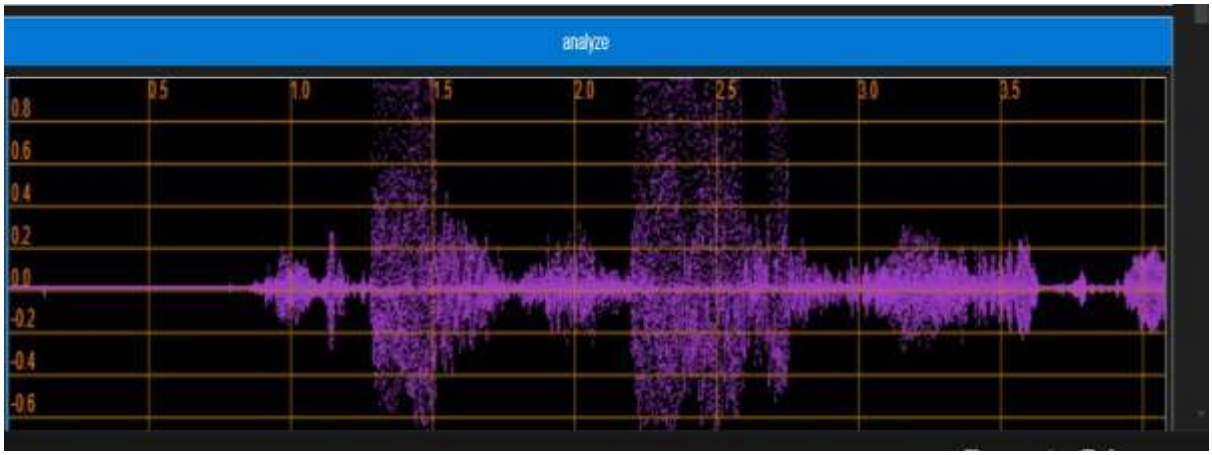
```python
import torch
import torch.nn as nn
import torch.optim as optim
from torch.utils.data import DataLoader, TensorDataset
from script.generator import Generator
from script.discriminator import Discriminator
from script.preprocess import process_data

# Device
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
print(f"Using device: {device}")

# Load data
clean_data_list, noisy_data_list = process_data()
print(f"Loaded {len(clean_data_list)} clean and {len(noisy_data_list)} noisy samples.")

# Preprocessing: Convert lists to tensors
clean_data = torch.stack([torch.tensor(x, dtype=torch.float32) for x in clean_data_list])
noisy_data = torch.stack([torch.tensor(x, dtype=torch.float32) for x in noisy_data_list])

# Reshape: [batch, channels, time]
clean_data = clean_data.unsqueeze(1)
noisy_data = noisy_data.unsqueeze(1)
```

3.1Input



3.2Output

The model is used to analyze fresh, unseen noisy speech signals after training. The result is a better speech signal that includes:
Better clarity and comprehension (e. g. , clearer speech sounds, less background noise).
• A better perception quality, which makes the voice sound more organic.
• Maintaining key speech features, making sure the fundamental components of the speech signal (such as phonetic details) are kept.



CHAPTER IV : CONCLUSIONS AND FUTURE SCOPE

4.1 Conclusions

By incorporating KolmogorovArnold Networks (KANs) into a MetricGAN+ framework, I concentrated on improving speech quality in loud settings. The goal of this method was to improve the modeling of complicated and nonlinear connections in noisy speech signals by utilizing KANs' strong function approximation capabilities. The integration of metric-driven training with KANs consistently demonstrated significant gains in the perceptual quality of the improved speech when compared to conventional approaches throughout the development.

Despite the project's promising outcomes, some constraints were discovered. The KANbased model brought about a modest rise in computational complexity, which might be problematic for resource-constrained or real-time applications. Furthermore, the majority of the system's performance validation was conducted on a controlled dataset; additional testing in more realistic and diverse acoustic environments would increase the approach's universality.

In general, this endeavor represents a positive advance in the development of speech augmentation systems that are more intelligent and aware of their environment. It offers thrilling potential for using KANs in a variety of audiovisual machine learning applications, not just in speech processing. The results of this study promote more investigation into how to improve and modify the suggested system for use in the real world, so that speech communication is improved in noisy situations.

4.2 Future Scope

The incorporation of KolmogorovArnold Networks (KANs) into a Metric-driven speech enhancement system was investigated in this study. This work might continue in a number of fascinating directions in the future. One crucial aspect is to improve the system for real-time usage by lowering complexity without compromising quality. The model may be made more effective in a wider range of real-world scenarios and noise kinds by training it using a wider set of evaluation metrics. To make the system more widely applicable, we may also investigate its application to different languages and acoustic settings. Lastly, this strategy has the potential to be paired with other deep learning techniques, like transformers or generative models, to improve robustness and performance in challenging situations.