MANIPAL
ACADEMY of HIGHER EDUCATION
(Institution of Eminence Deemed to be University)

Speech Enhancement Using AST

Mini-Project II Synopsis

submitted to

Manipal School of Information Sciences, MAHE, Manipal

Reg. Number	Name	Branch
241058020	Nikhil M	Big Data Analytics
241058024	Nikhil S G	Big Data Analytics
241058030	Vinayashree M Shet	Big Data Analytics

Under The Guidance Of

Mr. Raghudathesh G P

14/08/2024

MANIPAL SCHOOL OF INFORMATION SCIENCES
MANIPAL
(A constituent unit of MAHE, Manipal)

Table of Contents

1. Introduction

This project highlights the innovative **Audio Spectrogram Transformer (AST)**, a ground-breaking model that revolutionizes audio classification by leveraging a purely attention-based architecture. AST applies a Vision Transformer to audio spectrograms, treating them as images, which allows it to capture long-range dependencies and global context more effectively than traditional convolutional neural networks (CNNs). Developed by Yuan Gong, Yu-An Chung, and James Glass, AST achieves state-of-the-art results on various benchmarks, including Audio Set, ESC-50, and Speech Commands, with remarkable accuracy such as 0.485 mAP on Audio Set and 95.6% accuracy on ESC-50. By eliminating the need for convolutions, AST simplifies the architecture while maintaining superior performance, making it a versatile tool for diverse audio classification tasks. Its ability to handle variable-length inputs and adapt to different tasks without architectural changes further enhances its utility across various applications, from speech recognition to environmental sound detection.

2. Objective

2.1 Data Preparation

Objective: Prepare a comprehensive dataset for speech enhancement tasks using AST.

Tasks:

Data Collection: Gather a large dataset of noisy speech samples with corresponding clean versions.

Data Pre-processing: Convert audio files into spectrograms and normalize them for consistent input.

Data Augmentation: Apply techniques like noise injection and time stretching to increase dataset diversity.

2.2 Modelling the Data

Objective: Develop and fine-tune an AST model for speech enhancement.

Tasks:

Model Selection: Choose a pre-trained AST model or train one from scratch based on dataset size and complexity.

Model Adaptation: Modify the AST architecture to accommodate speech enhancement tasks, potentially integrating speaker embedding or conformer layers.

Fine-Tuning: Adjust the model parameters using the prepared dataset to optimize performance for speech enhancement.

2.3 Benchmarking the Results

Objective: Evaluate the performance of the AST-based speech enhancement model against existing benchmarks.

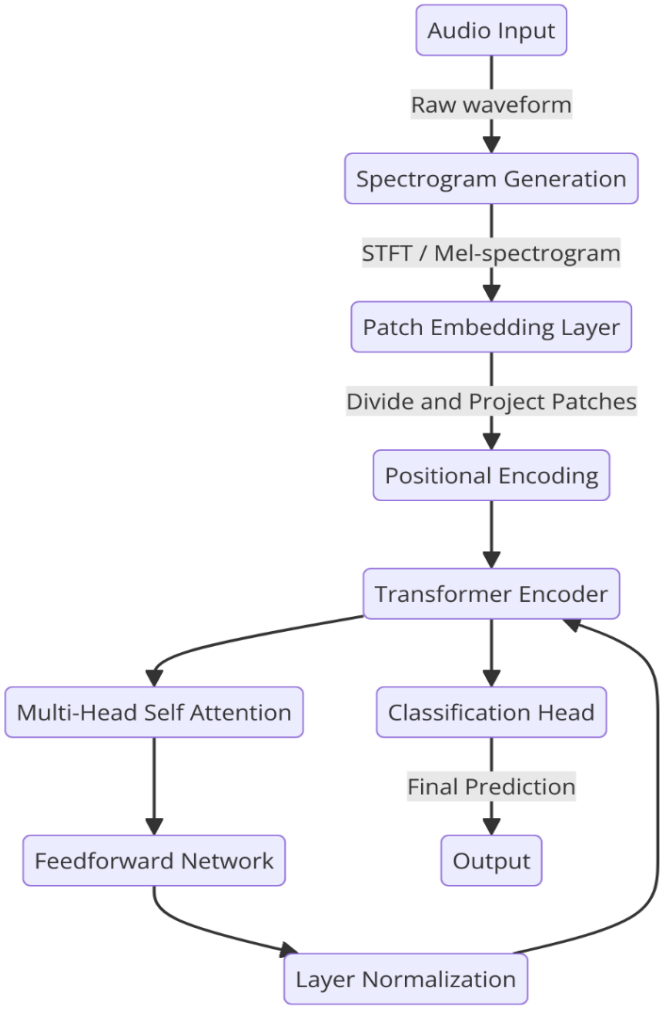
Tasks:

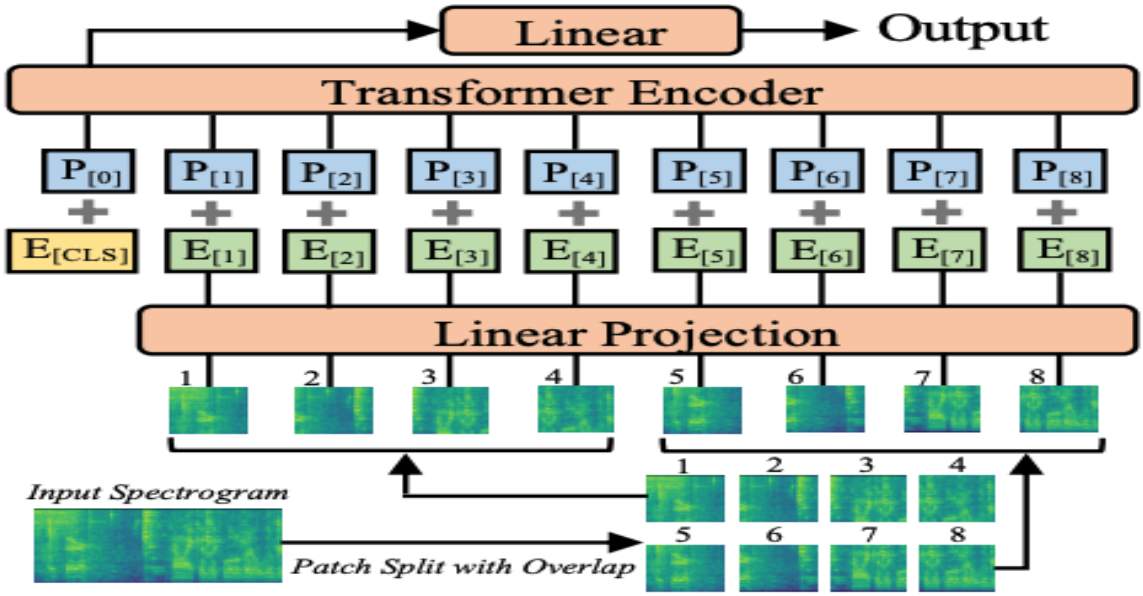
Metrics Selection: Use metrics such as Signal-to-Distortion Ratio (SDR), Short-Time Objective Intelligibility (STOI), and Word Error Rate (WER) for evaluation.

Comparison with Baselines: Compare the AST model's performance with state-of-the-art speech enhancement models, such as those using CNNs or GANs.

Cross-Validation: Perform cross-validation to ensure the model's robustness across different noisy conditions and datasets.

3. Block Diagram/Flowchart





4. Applications

4.1. Speech Recognition and Classification

Application: AST can be used to enhance speech recognition systems by accurately classifying audio signals into different categories, such as speech, music, or environmental sounds. This is particularly useful in noisy environments where traditional methods might struggle to distinguish between different audio types.

Examples:

Virtual Assistants: Improving the ability of virtual assistants like Siri, Alexa, or Google Assistant to recognize voice commands in noisy environments.

Speech-to-Text Systems: Enhancing the accuracy of speech-to-text systems used in transcription services or voice-controlled applications.

Environmental Sound Detection: Identifying specific sounds in environmental monitoring systems, such as detecting bird species or alerting to potential hazards.

4.2. Music and Audio Analysis

Application: AST can be applied to music analysis tasks, such as genre classification, mood detection, or identifying specific instruments within a track. This can help in music recommendation systems or music production software.

Examples:

Music Streaming Services: Enhancing music recommendation algorithms by accurately classifying music genres or moods.

Music Production Software: Helping producers identify and isolate specific instruments or sounds within a track for remixing or editing purposes.

Music Therapy: Analysing audio features to create personalized playlists for therapeutic purposes.

4.3. Cross-Domain Applications

Application: AST's versatility allows it to be applied across various domains beyond audio classification, including speech recognition, environmental sound detection, and even potential applications in healthcare or security.

Examples:

Healthcare: Analysing audio signals from medical devices to detect anomalies or predict health conditions.

Security Systems: Identifying suspicious sounds in surveillance systems to alert authorities.

Environmental Monitoring: Detecting changes in environmental sounds to monitor wildlife populations or detect natural disasters.

5. Software & Hardware Requirements

Software:

- Python: Main programming language.
- TensorFlow/Keras: For implementing and training the AST model.
- LibriSpeech, ESC-50, SiSec datasets: For obtaining clean speech and noise data.
- Google Colab: For using free GPU resources.

Hardware:

- GPU (Graphics Processing Unit): For faster training of deep learning models.
- High-Performance CPU: Supports general data processing and model training tasks.
- Storage: Enough space for storing and processing audio datasets, preferably over 15GB.

References

[1] Y. Gong, Y.-A. Chung, and J. R. Glass, "AST: Audio Spectrogram Transformer," arXiv preprint arXiv:2104.08730, 2021.

[2] O. Shavkatov and L. Niyazmetov, "Audio Spectrogram Transformer (AST): Advantages over Traditional Spectrogram Techniques," Journal of Transformations and Artificial Intelligence, vol. 2, no. 1, 2024.

[3] M. Abadi, P. Barham, J. Chen, et al., "TensorFlow: A system for large-scale machine learning," in Proc. USENIX Symposium on Operating Systems Design & Implementation, 2016.

[4] A. Naman, "FAST: Fast Audio Spectrogram Transformer," arXiv preprint arXiv:2501.01104, 2025.

[5] K. Wang, Y. Zhang, and J. Li, "TSTNN: Two-stage Transformer based Neural Network for Speech Enhancement in the Time Domain," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 30, pp. 1311-1323, 2022.