APPLIED STATE OF THE ART TECHNIQUES FOR CONCURRENT SPEAKER ESTIMATION

A. Michael Zeolla

Worcester Polytechnic Institute
Computer Science
mjzeolla@wpi.edu

B. Bashima Islam

Worcester Polytechnic Institute
Electrical and Computer Engineering
bislam@wpi.edu

ABSTRACT

Concurrent speaker estimation presents a significant challenge, yet a crucial component for addressing sophisticated tasks in the audio space, including surveillance, source separation, and speaker diarization. However, despite its importance, many implementations solving this issue fail to solve full-scale. This study investigates the application of cutting-edge state-of-the-art (SOTA) deep learning architectures to address the concurrent speaker estimation problem. The research aims to develop a robust, universal, and scalable model capable of tackling speaker estimations in real-world scenarios, such as small conversations, parties, or concerts. This paper delves into the different model architectures applied, the methodology implemented for testing, and their results in the audio domain for real-world datasets.

***Index Terms*—** Speaker Estimation, Deep Learning, Audio Processing, Speech Detection, Classification, Speaker Diarization, Neural Networks, SOTA

1. INTRODUCTION

Audio plays an integral role in human life, and despite audio-related tasks being at the forefront of deep learning, there are no state-of-the-art (SOTA) models that fully replicate all human auditory functions. A particular human skill that is crucial yet challenging to replicate is the ability to listen and accurately estimate the number of speakers in an environment, known as speaker estimation/counting. This skill is essential for comprehensively understanding and contextualizing surroundings; which allows people to make well-informed decisions. For example, speaker counting can have large-reaching impacts in areas like surveillance, and autonomous vehicles, by enabling effective speaker diarisation. Yet, despite the importance of this skill, many deep learning models are unable to estimate the number of speakers in an audio sample accurately, much less who is speaking or when. This is especially evident when considering a real-world scenario, such as a concert, or cocktail party, where multiple speakers and conversions are happening at the same time and audio samples have inherited noise. This paper researches the current SOTA models regarding speaker estimation and makes advances toward building a well-rounded model for replicating human-like estimation.

The work presented in this research study focused on the formulation of generic models with the ability to support speaker estimation. The previous work conducted by the CountNet team [1] considers the application of only traditional deep learning architecture, and the work by Yuan Gong et al [2] applied transformers to audio classification, but solved a more generic task, not focusing on the complexities of speaker estimation. While the present study is related to recent approaches in audio classification, it capitalizes on new SOTA models, which were not considered in alternative studies, and applied architectures to more applicable datasets for the task.

2. METHODOLOGY

The main focus of this research study involved testing and fine-tuning different SOTA models and architectures to produce applicable results. To that end, three main model architectures were analyzed on the problem statement, each involving different techniques to learn and classify audio samples despite their origin.

2.1. Architectures

Model #1: CRNN. The CountNet paper focused heavily on concurrent speaker estimation by applying more traditional deep learning architectures; such as Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN) [1]. The primary research on CountNet showed that Short-time Fourier transforms (STFT) and a CNN + RNN-based architecture (CRNN) produced the best results compared to alternative setups [1].

CRNN is a model that integrates the strengths of CNNs and RNNs by stacking them sequentially. The benefit of layer stacking is to leverage the locality of convolutions in aggregating local features with the RNN's ability to model long-term data [1]. The CNN layers aggregate local time-frequency features, while the LSTM layer tracks long-term temporal structures; creating a context-rich result. To map between the 3D CNN output, with shape D x F x C, and the 2D RNN input, the channel dimension C is stacked with the frequency dimension F,

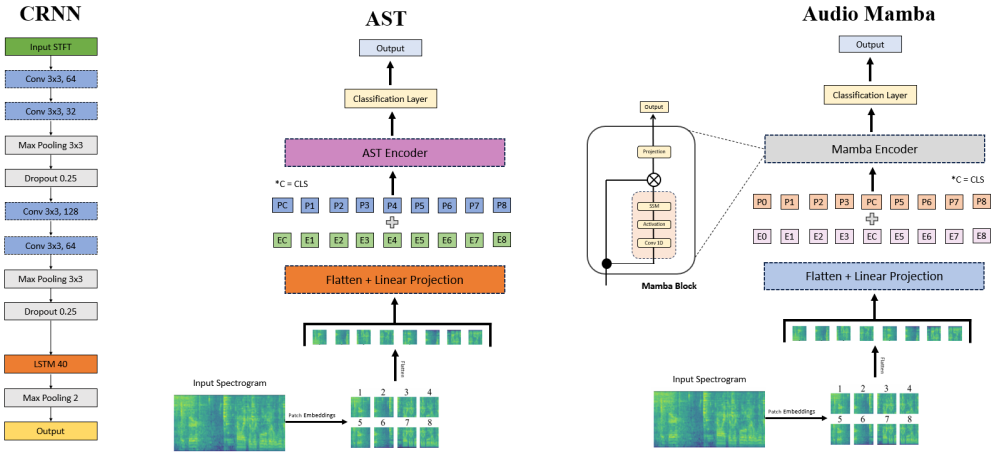


Fig. 1. Implemented Model Architectures [3] [4] [1]

producing a 2D output D X F * C for the LSTM [1]. the CRNN model produced the baseline performance metrics for this study, and was the main model "to beat".

Model #2: Audio Spectrogram Transformer. While traditional models played a major role in concurrent speaker detection, advancements in Deep Learning, such as Transformers, and Attention, altered this landscape. Transformers, originally proposed in the 2017 paper, "Attention is All You Need" [5], opened many more opportunities by allowing models to handle larger data sets with a large context window concurrently. Specifically, the Audio Spectogram Transformer (AST) was the "first convolution-free, purely attention-based" transformer for audio classification [2]. AST was trained on multiple large-form datasets in the speech domain, including, SpeechCommandsV2 and ESC-50 [2]. While these datasets are similar to the speaker estimation task, there is not a 1-to-1 relation since these datasets do not include a large number of concurrent speakers. The dataset for speaker estimation must resemble more diverse samples that mimic real-world situations. Therefore, fine-tuning the AST model on the downstream task of speaker estimation was required. To accomplish this, specific layers in the AST encoder were unfrozen during training, which allowed the model to re-learn the more advanced features saved in the model weights while leveraging the basics learned features in the earlier layers.

Model #3: Audio Mamba. Despite the beauty and versatility of transformer-based architectures, limitations exist when applying transformers to different problem statements. Primarily, transformers are largely hindered by their computational complexity due to their self-attention mechanisms; which is especially true for inputs involving larger data structures, such as audio or images. This is because the attention mechanism has an $O(n^2)$ cost when processing longer sequences [6]. Alternative architectures, labeled "State Space Models" (SSMs) have alleviated the pains of self-attention by employing different techniques to replicate and replace these concerns. Prominent SSM architectures include Mamba, and the derivative Audio-Mamba (AMaM), which apply time-varying parameters that capture input context and produce efficient outputs on different tasks [4] [3]. The model architecture for the AMaM model is visible in Figure 1, which visualizes the embedding process, encoding tokens, and other processing operations for the model. One key difference between the AST and AMaM implementations is the placement of the CLS token, which in AST is placed at the start, while AMaM introduces the CLS token in the middle of the embedding input.

Output Layer. Each model was augmented with the same output regression/classification layer, which applied a linear layer and softmax activation function. The speaker estimation task was treated as a classification problem, and applying the softmax activation function allowed the model to produce a vector of probabilities per class. The output classification layer was appended to the last layer of each model while leveraging the pertained parameters of each model, if applicable.

2.2. Data Features

When working with audio data there are many different ways to represent the audio, each with a different level of complexity and purpose. However, the primary feature that has shown promise concerning audio classification is STFT [1]. STFT is valuable because it extracts time-frequency features that help to distinguish between different types of audio events and patterns. Therefore, each of the audio models trained applied the STFT feature, either by direct STFT input or by computing the STFT via a custom encoder. Additionally, Mel-frequency was a secondary feature metric applied in audio processing to represent the frequency content of a signal.

3. EXPERIMENTS

3.1. Datasets

While many deep learning datasets exist for audio classification, few have complex speaker overlap, or speaking patterns resembling real life. Therefore, new datasets were generated to properly replicate a true "cocktail party", which resembles a more complex speaking environment. For example, the generated audio samples have multiple speakers, speaker overlap, and variation in speech patterns. The datasets utilized during experimentation were: (1) LibriCount10 and (2) AudioCocktailParty. Both datasets were generated as a subset of the LibriSpeech dataset, a corpus of approximately 1000 hours of 16kHz English speech [7] [8].

(1) LibriCount10. The LibriCount10 dataset is an auto-generated dataset for speaker estimation; which contains normalized audio samples within a range of 0-10 speakers, and their configuration data [1] [8]. The audio samples are mixed with 0dB SNR from random utterances of different speakers from the LibriSpeech-TrainClean dataset and are in the format of 5-second 16kHz audio clips. This dataset contains 5720 unique audio samples, totaling 8 hours of content.

(2) AudioCocktailParty. The AudioCocktailParty dataset is a new dataset generated for this research. Similarly to the LibriCount10 dataset, AudioCocktailParty was generated as a subset of the LibirSpeech audio samples, however, AudioCocktailParty is significantly larger, boasting over 20,000 audio samples, each 10 seconds in length; for over 55 hours of audio content. AudioCocktailParty was also generated from more diverse audio, taking samples from LibriSpeechTrainClean and LibriOther, which contain a more lengthy set of audio samples. While LibriCoun10 dataset is a feasible dataset for training simple architectures, when training an AST or AMaM model, the smaller dataset size is prone to overfitting. Additionally, the AudioCocktailParty dataset exhibits more naturalistic human speech, by not normalizing the audio samples, and including alternative non-human sounds. Overall, this dataset is more representative of real-world scenarios. Audio samples in AudioCocktailParty also vary from a range of [0-10] concurrent speakers in an audio frame.

3.2. Loss & Evaluation Metrics

Model Loss. Count estimation as a (a) regression problem requires the model to output the number of speakers as a numeric value, and for (b) classification, the model generates a probability distribution over X classes. However, speaker estimation produces better results when solving for classification [1]. Therefore, the majority of the models trained treated the problem statement as a classification problem and applied Cross-Entropy Loss.

	LibriCount10		CocktailParty	
Model Type	$\bar{MAE}$	Acc.	$\bar{MAE}$	Acc.
CRNN-MEL	0.54	56%	0.59	54%
CRNN-STFT	0.53	58%	0.54	56%
Norm-MEL	0.51	59%	0.54	57%
Norm-STFT	0.44	62%	0.52	58%
AST-4	0.30	72%	0.47	61%
AST-6	0.26	75%	0.44	64%
AMaM Fo-Bi-2	0.42	64%	0.54	56%
AMaM Fo-Bi-4	0.41	64%	0.53	58%

Table 1. Model Testing Results Breakdown

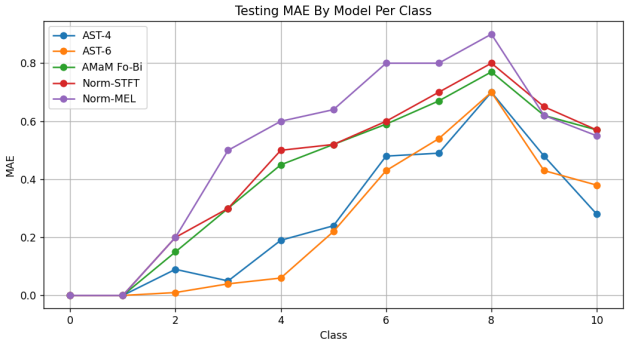
Evaluation Metrics. The main evaluation metrics utilized were Mean Average Error (MAE) and Accuracy (Acc). MAE was calculated at an average level and per class; while accuracy provided an overall evaluation.

3.3. Results

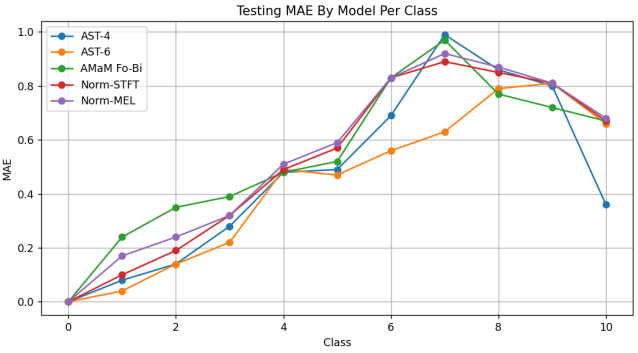
CRNN. When training the CRNN architecture, rather than using a pre-trained checkpoint, a new model was trained. The input features, STFT and MEL were utilized to evaluate the effectiveness of each. Additionally, the effect of normalizing the input features was tested by training alternative models; and a Multi-Step Learning Rate reduction was applied every 15 epochs by a factor of $\frac{1}{2}$, starting at 3e-4, to optimize convergence. Table 1 visualizes the results of each CRNN model trained, and based on the results, the normalized STFT model produced the best testing results with a lower $\bar{MAE}$ and higher accuracy for each dataset. The model produced an $\bar{MAE}$ and accuracy of 0.44, and 72% for LibriCount10, and for AudioCocktailParty, 0.52 $\bar{MAE}$, and 58% accuracy.

AST. A pre-trained checkpoint was applied for the AST architecture to reduce the time spent retraining the transformer encoder architecture. Since the AST model was originally trained on a similar audio dataset, AudioSet, ESC-50, and SpeechCommandsV2, the prior weights could be leveraged with transfer learning. The main optimization point for the AST architecture was evaluating the best set of unfrozen weights while balancing model overfitting. Different AST architectures were trained per dataset, each with various unfrozen layers. Additionally, the team experimented with the learning rate, varying the value slightly between the ranges of 3e-1 and 3e-5. The best result for the AST occurred with ($\bar{MAE}$: 0.26, Accuracy: 75%) for LibirCount10 and ($\bar{MAE}$: 0.44, Accuracy: 64%) for AudioCocktailParty.

While unfreezing may lead to better results, it can indirectly lead to rampant overfitting within the model, as it memorizes training inputs without considering testing values. Furthermore, a larger learning rate caused the AST model to fail to converge, even when training for 150+ epochs. In contrast, using smaller learning rates—



(a) LibriCount10 Results



(b) AudioCocktailParty Results

Fig. 2. Testing MAE per Class Breakdown

approximately 10 times smaller than the typical learning rate values for CRNN models— yielded significantly better results for the models trained.

Audio-Mamba. In a manner resembling the AST implementations, a pre-trained checkpoint from an Audio-Mamba model, trained on analogous audio datasets, was applied. Several layers were unfrozen, and transfer learning was applied to each of the datasets when training. The Audio-Mamba model architecture configuration applied was a Forward Bidirectional (Fo-Bi) model. The original Auio-Mamba paper displayed that bidirectional variants of the model produced better performance over forward-only alternatives, especially for larger datasets, which is the case for AudioCocktailParty [3].

Each configuration was trained under matching conditions, trained for 150 epochs, with a decaying learning rate scheduler described in section 3.3. The results of the experiments demonstrated the following average performance metrics: ($\bar{MAE}$: 0.42, Acc: 64%) for LibriCount10 and ($\bar{MAE}$: 0.54, Acc: 56%) for AudioCocktailParty. Unfortunately, despite the optimization techniques applied, the Audio-Mamba results are closer in performance to the CRNN results than AST for this particular task. Additional optimizations for the Audio-Mamba models may lead to better model performance for both speaker estimation datasets, however, such optimizations were outside the scope of this research due to time constraints. One potential area for improvement is testing the alternative model architectures for AMaM, which vary the Forward (Fo) and Bidirectional (Bi) model to produce better results for different data types: (1) Fo-Bi, (2) Bi-Bi, and (3) Fo-Fo [3].

Cross Comparison. Overall, the AST architecture produces the best results compared to the other tested frameworks; outputting the best average MAE and accuracy score for each model. While the amount of variation in the unfrozen layers did play a large role in model performance, the best overall model was the AST-6, with 6 unfrozen encoder layers; which outputted the best score for both LibriCount10 and AudioCocktailParty. However, this performance versus the other models comes with a cost, as the AST model has significantly more parameters, resulting in a longer training time.

The results visible in Figure 2 show a direct comparison of each model's testing MAE breakdown per class on each dataset. From this diagram, it can be inferred that the models suffer the most when predicting samples with a higher number of speakers, [6-8]. Yet, when classifying samples with 9 or 10 speakers, the models have an improvement in performance over the [6-8] audio predictions. This is anticipated to be a consequence of the model knowing the total number of max speakers, in this case [0-10] speakers due to it being a classification task. For classification, the model inherently knows the maximum number of speakers, as opposed to regression, where the model does not inherently know the speaker range. The model can apply this encoded bias when predicting samples with a higher number of speakers to always output the known maximum number of speakers, rather than predicting people in the range [6-8].

The CRNN and AMaM models also produced prominent results for both datasets evident by their ability to classify audio samples with speakers in the range of [0-5], but they suffer when considering inputs with more concurrent speakers. Specifically, samples between [6-7] speakers had significantly higher MAE values when compared to the AST model for the same sample range, as seen in Figure 2. Overall. each model is capable of outputting performance levels significantly better than randomly estimating the number of speakers (10%), with accuracy values between 50%-70% for each dataset.

4. CONCLUSIONS

In this research study different architectures and methodologies were implemented to aptly model human auditory functions; focusing on concurrent speaker estimation. Utilizing advanced audio processing techniques and state-of-the-art deep learning models, speaker estimation was proven possible and feasible using alternative architectures. Specifically, this research has shown that non-traditional models, such as transformers, and state space models, can play a major role in complex audio classification tasks, even when considering more complex audio classification objectives, like speaker diarization.

5. REFERENCES

[1] F.-R. Stöter, S. Chakrabarty, B. Edler, and E. A. P. Habets, "Countnet: Estimating the number of concurrent speakers using supervised learning," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 2, pp. 268–282, February 2019.

[2] Yuan Gong, Yu-An Chung, and James Glass, "Ast: Audio spectrogram transformer," *arXiv preprint arXiv:2104.01778*, 2021, Preprint.

[3] Mehmet Hamza Erol et al., "Audio mamba: Bidirectional state space model for audio representation learning," *arXiv preprint arXiv:2406.03344*, 2024, Preprint.

[4] Albert Gu and Tri Dao, "Mamba: Linear-time sequence modeling with selective state spaces," *arXiv preprint arXiv:2312.00752*, 2023, Preprint.

[5] Ashish Vaswani, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.

[6] Daniel Y. Fu et al., "Hungry hungry hippos: Towards language modeling with state space models," *arXiv preprint arXiv:2212.14052*, 2022, Preprint.

[7] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, South Brisbane, QLD, Australia, 2015, pp. 5206–5210.

[8] Fabian-Robert Stöter, Soumitro Chakrabarty, Emanuël Habets, and Bernd Edler, "Libricount, a dataset for speaker count estimation (v1.0.0)," ICASP 2018, Calgary, Canada, 2018, Zenodo. https://doi.org/10.5281/zenodo.1216072.