AI - MUSİC TRANSCRİPTION

Francky Ronsard SAAH
TECHNOLOGY FACULTY
INFORMATION SYSTEM INGENEERİNG
KOCAELI, TURKEY
francky877832@gmail.com

KHADIM DIEYE
TECHNOLOGY FACULTY
INFORMATION SYSTEM INGENEERİNG
KOCAELI, TURKEY
Khadimd978@gmail.com

Abstract—*This project focuses on training transformer-based models for the automatic transcription of guitar notes. The dataset comprises recordings of seven distinct guitar notes organized into folders. Models including HuBERT, Wav2Vec, and AST were implemented. Each model's performance was evaluated using classification metrics such as Accuracy, Recall, Precision, F1-Score, and AUC.*

Keywords—*Music transcription, model*

I. INTRODUCTION

The transcription of audio signals into meaningful information has numerous applications, including music analysis, assistive technologies, and digital learning. The purpose of this project is to utilize state-of-the-art transformer models to recognize individual guitar notes from audio recordings. We aim to evaluate model performance comprehensively and provide a comparison based on key metrics.

II. DATASET

The dataset consists of seven folders, each representing a unique guitar note. Each folder contains audio recordings labeled with the corresponding note. The data preprocessing pipeline included:

- Noise removal to enhance signal clarity.
- Normalization of audio signals.
- Splitting audio into train-test datasets using an 80:20 ratio.

III. MODELS

The following transformer-based models were implemented and trained for the audio transcription task:

HuBERT (Hidden-Unit BERT): Known for robust feature extraction from audio, particularly in unsupervised learning scenarios, making it a strong candidate for audio processing tasks.

Wav2Vec: A model designed for self-supervised learning, focusing on learning high-quality representations of speech, which can be transferred to various speech-related tasks.

AST (Audio Spectrogram Transformer): Specializes in spectrogram-based audio analysis, converting raw audio signals into spectrogram representations for better understanding of frequency and time-based patterns.

YAMNet (Yet Another Mobile Network): A deep neural network for audio event recognition, trained on a wide range of audio events and capable of learning both fine and coarse features from raw audio.

ViT (Vision Transformer): Although originally designed for image recognition, ViT can be applied to audio by treating spectrograms as images. It has shown promising results in various audio classification tasks by leveraging transformer-based mechanisms.

IV. LIMITATIONS

V. EVALUATION METRICS

The classification performance was assessed using:

Accuracy, Recall, Precision, F1-Score, Sensitivity, Specificity, and AUC.

Training and inference times were recorded for comparison.
For each model, training and test loss vs. epoch graphs were plotted. Confusion matrices and ROC curves were also generated.

VI. RESULTS

The convergence behavior of each model was observed as follows:

HuBERT, Wav2Vec, and AST: These models demonstrated consistent convergence trends, although their performance was affected by the reduced dataset.

YAMNet: Due to its larger model size and the diversity of its training data, YAMNet took longer to converge, showing gradual improvements in the metrics over time.

ViT: This model also demonstrated steady progress in terms of training loss and validation accuracy. However, the additional time required for each training epoch made it challenging to complete training on the full dataset within the given resources.

-

VII. LIMITATION OF TRAINNING ENVIRONMENT

Limitations of Training Environment

The performance of our models, was limited by significant constraints in the training environment. These include:

Dataset Size

The dataset comprised approximately 35,000 audio files representing individual guitar notes. This large volume of data was difficult to handle on a machine with limited computational resources, creating bottlenecks in memory allocation and processing speed.

Training Hardware

My local machine lacked sufficient GPU and memory capabilities to process the dataset efficiently. This meant that training on the full dataset would take several days, making it impractical given the available resources.

Google Colab Free Tier

To utilize GPU acceleration, I opted for Google Colab's free version, which introduced several constraints:

Session Timeouts: Sessions were automatically terminated after a fixed period, regardless of whether training was complete.

Loss of Progress: Upon termination, session data and progress were lost, requiring training to be restarted from scratch.

To address these limitations, I trained the models on a reduced subset of the dataset. While this approach allowed me to observe the models' behavior, it limited their ability to generalize to the full dataset.

Impact of Pretrained Models on Workflow

Pretrained models such as HuBERT, Wav2Vec, and AST introduce a unique workflow that directly impacts the training process and efficiency. Unlike traditional machine learning workflows, where audio data is often converted into structured formats like CSV files with extracted features (e.g., Mel-frequency cepstral coefficients or spectrograms), these models operate on raw audio signals.

Raw Audio Input

Pretrained transformer models are designed to work directly with raw audio signals, avoiding the need for manual feature extraction.

While this simplifies data preparation, it drastically increases the computational complexity of training because the models process high-dimensional inputs during runtime.

Time and Resource Constraints

The additional processing burden of handling raw audio data increases training time significantly.

Feature extraction performed offline (e.g., converting audio to CSV before training) could reduce the computational load during training, but this approach is not compatible with the architecture of these models.

Trade-offs in Workflow
The reliance on raw audio data shifts the computational demand from data preprocessing to model training. This approach imposes a longer training duration but offers the advantage of using robust, domain-specific pretrained representations of audio data.

Observations on Model Convergence

Despite the constraints, the models demonstrated promising convergence trends when trained on a reduced dataset:

Training Loss: Consistently decreased with each epoch, reflecting effective learning even with limited data.

Validation Metrics: Showed steady improvement, though plateaued earlier due to the restricted dataset size.

These trends indicate that the models are capable of learning meaningful representations of the audio signals and would likely perform significantly better given more resources and a larger dataset.

Figure X illustrates the training and validation loss over epochs, highlighting the gradual convergence of the model.

Potential with Enhanced Resources

The limitations observed in this project underscore the potential for improved performance under better conditions:

Enhanced Computational Power
Access to high-performance machines with sufficient GPU and memory resources would enable:

Training on the full dataset, capturing its diversity and improving generalization.

Faster convergence, reducing overall training time.

Google Colab Premium
Upgrading to Google Colab's premium plan would address session timeout issues by providing:

Extended session durations of over 24 hours.

Guaranteed continuity of training without the loss of session data.

Alternative Workflows
Future efforts could explore hybrid workflows that combine the strengths of pretrained models with offline feature extraction. This approach may reduce the computational burden during training while maintaining high-quality representations of the audio data.

Broader Implications of Workflow Adaptations

The reliance on pretrained models that process raw audio data emphasizes the need to adapt workflows accordingly:

Data Handling: Training workflows must integrate efficient data pipelines to load and preprocess large audio datasets dynamically during training.

Resource Allocation: Resource-intensive processes, such as on-the-fly feature extraction from raw audio, require powerful hardware configurations to achieve practical training times.

Scalability: Optimizing the handling of raw audio data could open pathways to scaling such models for larger and more complex audio datasets.

By addressing these challenges, future implementations can unlock the full potential of transformer models for audio transcription tasks.

VIII. Future Work

Future improvements could include:

- **Advanced Training**: Each model can be developed by training it with a larger set of datas within an appropriate computer.

IX. Conclusion

This project successfully demonstrated the use of transformer models for guitar note transcription. HuBERT emerged as the best-performing model, offering both speed and accuracy. Future work could explore ensemble methods and additional data augmentation techniques to enhance model generalization..

References

[1] D. P. W. Ellis, "Automatic music transcription: A survey of the state of the art," *Foundations and Trends® in Signal Processing*, vol. 5, no. 3–4, pp. 197–291, 2011.

[2] https://discuss.huggingface.co/t/trainer-class-compute-metrics-and-evalprediction/1698/3