CentraleSupélec

Forest Fire Detection from Acoustic Signals using Transformer-Based Models and Low-Rank Adaptation (LoRA)

Mayara AYAT, Sarra NAGAZ, Yani HAMMACHE, Mathias MARTINEZ

Supervisor: Frédéric MAGOULÈS

September 15, 2025

Abstract

Forest fires emit characteristic acoustic signatures that can be detected and analyzed for early warning systems. Traditional detection approaches rely on visual monitoring or satellite imagery, but these often suffer from delays or limited coverage. Acoustic methods, by contrast, can provide continuous and real-time monitoring even under poor visibility. Recent advances in deep learning, particularly transformer-based models, have enabled efficient classification of complex audio signals. In this project, we investigate the use of the Audio Spectrogram Transformer (AST) and its fine-tuned variant with Low-Rank Adaptation (LoRA) for forest fire detection from sound recordings. By leveraging pretrained transformer architectures and parameter-efficient fine-tuning, the goal is to achieve high accuracy while keeping computational costs manageable for deployment in real-world monitoring systems.

1 Introduction

Forest fires are a major environmental threat with devastating ecological and social impacts. Early detection is critical, yet traditional approaches (visual monitoring, satellite imagery, ground-based sensors) face limitations in coverage, delay, and reliability. Acoustic monitoring offers an alternative, as fires generate distinct sound patterns that can be continuously captured. In this context, recent advances in transformer architectures, combined with parameter-efficient fine-tuning techniques such as LoRA, provide an opportunity to develop scalable and robust wildfire sound detection systems.

2 Related Works

Early research on acoustic fire detection relied heavily on classical signal processing and dimensionality reduction methods, such as principal component analysis [3], to distinguish fire from non-fire sounds. While these approaches provided initial insights, their results were limited, highlighting the need for more advanced representations of audio data. The transition to deep learning introduced stronger feature extraction pipelines: methods based on log-Mel spectrograms and convolutional neural networks (CNNs) demonstrated that spectrogram-based classification could effectively capture the spectral and temporal properties of fire sounds. Proof-of-concept studies using high-fidelity recordings of controlled burns showed F-scores above 98%, underlining the promise of acoustic approaches in conditions where vision-based systems may fail [17].

Building on these foundations, hybrid models combining CNNs, recurrent layers (e.g., LSTMs), and heuristic optimization have been proposed to improve robustness in noisy forest environments. By leveraging features such as MFCCs, RMSE, and ZCR, these approaches achieved strong accuracy (up to 94.7% with F1 = 94.6%) in challenging conditions [4]. Parallel to this, work on embedded systems has emphasized real-world constraints: low-power IoT devices equipped with lightweight CNNs for audio spectrograms and compact vision models like MobileNetV2 have been shown to achieve F1-scores around 96% while ensuring low latency and energy efficiency for real-time deployment [18]. Together, these studies highlight two key directions: the design of advanced hybrid models to maximize robustness and the adaptation of models for constrained hardware suitable for large-scale deployment.

Beyond model architectures, preprocessing is a critical component in acoustic fire detection. Recent studies have emphasized the importance of transforming raw audio into structured time–frequency representations. PCA-based dimensionality reduction has been explored for distinguishing fire and non-fire audio signals, though results remain inconclusive [3]. More advanced pipelines relying on Fourier analysis and filterbanks have been developed to extract discriminative features from fire sounds [21]. The use of Mel spectrograms, which emphasize perceptually relevant frequencies, has

become standard in sound recognition tasks. Dedicated studies introduced preprocessing methods such as Fourier transforms [2], Mel spectrogram representations [23], and educational resources such as the Hugging Face Audio Data Course [8], all of which provide the technical foundation for modern acoustic classification. These techniques remain central today: they underpin CNN-based proof-of-concept systems for fire detection [17] and continue to serve as the input representation for transformer-based approaches such as AST [11].

Recent advances in transformers have further reshaped the landscape. The transformer architecture, originally introduced for NLP [22], was later adapted to vision tasks through the Vision Transformer (ViT) [6] and then extended to audio classification via the Audio Spectrogram Transformer (AST) [11]. The Vision Transformer (ViT) introduced the idea of treating an image as a sequence of non-overlapping patches, which are linearly projected into embeddings and processed by a standard Transformer encoder. By removing convolutional inductive biases and relying solely on self-attention, ViT demonstrated that Transformers could achieve competitive or superior results to convolutional neural networks (CNNs) when trained on large-scale datasets such as ImageNet. Its success established a general framework for applying attention-based models to spatially structured data, paving the way for domain-specific adaptations like AST in the audio domain.

AST demonstrated that purely attention-based models can match or surpass CNN-based methods in learning long-range dependencies in spectrograms, achieving state-of-the-art results on ESC-50 [19] and other benchmarks. However, fine-tuning such large models is computationally expensive. To address this, parameter-efficient fine-tuning (PEFT) methods have been developed, including adapters [13], prompt tuning [16], and most recently Low-Rank Adaptation (LoRA) [14]. LoRA reparameterizes weight updates as the product of two low-rank matrices, dramatically reducing the number of trainable parameters while keeping the base model frozen. This idea builds on the classical theory of low-rank matrix approximation [7], later applied to neural network compression [5], and has become a cornerstone in efficient adaptation of large transformers. In this work, we extend these ideas by applying LoRA to AST for the specific task of wildfire sound detection, aiming to combine the representational power of transformers with the efficiency required for practical deployment.

3 Methodology

Our methodology consists of three complementary approaches:

- **Transfer learning with AST:** Using the pretrained Audio Spectrogram Transformer (AST) as a frozen feature extractor with a Multi-Layer Perceptron (MLP) classifier. [20]

- **LoRA fine-tuning of AST:** Fine-tuning AST with Low-Rank Adaptation (LoRA) to reduce the number of trainable parameters while maintaining model expressivity.
- **Transfer learning with ViT:** Leveraging a pretrained Vision Transformer (ViT) model, originally trained on large-scale image datasets, to classify spectrograms converted into RGB images. The ViT backbone was frozen, and only the final classifier head was trained.

The motivation for testing both AST and ViT lies in their complementary design philosophies. AST is specifically designed for audio spectrograms and directly optimizes the Transformer architecture for time–frequency representations. In contrast, ViT is a generic vision model that has shown remarkable transferability across domains. By evaluating both, we can compare a domain-specialized Transformer (AST) against a domain-agnostic Transformer (ViT) to determine which architecture is more effective for environmental sound classification and wildfire detection.

Datasets

For this project, we leveraged two datasets to develop and evaluate our wildfire detection model based on sound.

The first dataset is **ESC-50**, a labeled collection of 2,000 environmental audio recordings, each 5 seconds long, commonly used for benchmarking environmental sound classification methods. For our purposes, recordings from this dataset were assigned **Class 0 – No Fire** [19].

The second dataset is the **Forest Wildfire Sound Dataset** from Kaggle, which contains approximately 280 audio recordings, each around 50 seconds in duration. These recordings were labeled **Class 1 – Fire** [1].

By combining these two datasets, we created a dataset representing both fire and non-fire sounds, suitable for training and evaluating our detection model.

Audio Spectrogram Transformer (AST) Architecture

The AST model processes audio signals using a Transformer-based architecture. The pipeline can be described as follows:

1. **Input Conversion:** The raw audio waveform is converted into a 128-dimensional log Mel filterbank (fbank) spectrogram, where energy is expressed on a decibel (dB) scale.
2. **Patch Splitting:** The 2D spectrogram is divided into 16×16 patches, with an overlap of 6 along both the time and frequency dimensions.

3. **Patch Embedding:** Each patch is flattened and projected into a 768-dimensional embedding vector.

4. **Positional Embedding:** A learnable 768-dimensional positional embedding is added to each patch embedding to encode order information, as the Transformer architecture does not inherently preserve sequential or spatial structure.

5. **Classification Token ([CLS]):** A special [CLS] token is prepended to the sequence of embeddings, following standard Transformer practice.

6. **Transformer Encoder:** The sequence of embeddings is passed through a Transformer encoder consisting of 12 layers and 12 attention heads, with an embedding size of 768. Only the encoder is used.

7. **Classification Output:** The final embedding corresponding to the [CLS] token is used as the spectrogram representation. This embedding is then fed into a linear layer with sigmoid activation to predict classification labels.

The AST model was added to Hugging Face Transformers on 2022-11-21 [10]. Its feature extractor can handle the entire preprocessing pipeline, including converting audio waveforms into log Mel filterbank spectrograms and preparing inputs for the Transformer model.

General Pipeline

1. **Input preprocessing:**
   (a) Convert raw audio waveforms into 128-dimensional log-Mel spectrograms.
   (b) Segment spectrograms into 16×16 overlapping patches (stride = 6).
   (c) Flatten patches and project them into 768-dimensional embeddings.
   (d) Add positional embeddings and prepend a [CLS] token.

2. **Dataset split:**
   (a) Use 80% of the dataset for training and 20% for validation.
   (b) Datasets: ESC-50 (environmental sounds) and wildfire dataset.

3. **Training setup:**
   (a) Optimizer: Adam. [15]
   (b) Loss function: Binary Cross-Entropy. [12]
   (c) Epochs: 5.
   (d) Metrics: Accuracy, Precision, Recall, F1-score, Confusion Matrix.

Approach 1: AST Fine-Tuning

1. Load pretrained AST weights (from Hugging Face).
2. Extract [CLS] token embeddings as audio representations.
3. Feed [CLS] embeddings into a Multi-Layer Perceptron (MLP) with two ReLU layers for classification.
4. Train only the MLP classifier head while keeping AST frozen.

Approach 2: AST with LoRA Fine-Tuning

1. Load pretrained AST weights (base model frozen).
2. Apply LoRA modules to linear layers (rank $r \ll \min(d, k)$) where d and k are the weight matrices dimensions.
3. Train only LoRA parameters and the classifier head.

ViT Architecture

The Vision Transformer (ViT) adapts the Transformer architecture, originally introduced for natural language processing, to image classification. In this work, Mel spectrograms were converted into RGB images and processed as visual inputs. [9] The architecture is as follows:

1. **Patch Splitting:** The input spectrogram image is divided into non-overlapping 16×16 patches.
2. **Patch Embedding:** Each patch is flattened and linearly projected into a 768-dimensional embedding vector.
3. **Positional Embedding:** Learnable positional embeddings are added to encode spatial order information.
4. **Classification Token ([CLS]):** A special [CLS] token is prepended to the patch sequence.
5. **Transformer Encoder:** The sequence is processed through 12 Transformer encoder layers with 12 self-attention heads and hidden size 768.
6. **Classification Output:** The final embedding corresponding to the [CLS] token is passed through a linear classifier. We replaced the pretrained classifier with a new linear layer producing two outputs (fire crackling vs. non-fire crackling).

General Pipeline (ViT)

1. **Input preprocessing:**
   (a) Audio waveforms were resampled to 22,050 Hz.
   (b) Converted into 128-dimensional log-Mel spectrograms, expressed in decibels (dB) and scaled to [0, 1].
   (c) Spectrograms were converted to RGB images for compatibility with ViT.
   (d) Images were divided into non-overlapping 16×16 patches, flattened, projected into embeddings, with positional embeddings added and a [CLS] token prepended.
2. **Dataset split:**
   (a) 80% of the dataset was used for training and 20% for validation.
   (b) Datasets: ESC-50 (environmental sounds) and wildfire dataset.
3. **Training setup:**
   (a) Optimizer: Adam.
   (b) Loss function: Cross-entropy. [12]
   (c) Epochs: 5.
   (d) Metrics: Accuracy (primary) and validation loss.
   (e) Backbone frozen; only the classifier head (1538 parameters) was trained.

4 Results and Discussion

The two approaches (AST baseline and AST+LoRA) were evaluated on the combined ESC-50 and wildfire datasets. Table 1 summarizes the main metrics.

Table 1: Performance comparison between AST baseline and AST+LoRA.

Model	Train Loss	Train Acc.	Val Loss	Val Acc.	Val F1
AST baseline	0.000	100%	1.81×10^{-6}	100%	100%
AST + LoRA	0.069	96.3%	0.023	99.8%	99.8%
ViT	0.395	85.6%	0.373	85.5%	85.9%

The corresponding confusion matrices are:

$$\text{AST baseline: } \begin{bmatrix} 387 & 0 \\ 0 & 71 \end{bmatrix} \qquad \text{AST+LoRA: } \begin{bmatrix} 400 & 0 \\ 1 & 57 \end{bmatrix}$$

Both AST approaches achieved near-perfect performance. The baseline AST reached 100% across all metrics, while the LoRA-enhanced AST achieved 99.8% validation accuracy and F1, with only a single misclassification in the validation set.

Despite these promising outcomes, several limitations remain. First, the datasets used (ESC-50 and a curated wildfire dataset) do not fully capture the complexity of real forest environments, including background noise, wind, rain, animal sounds, or overlapping events. Second, the confusion matrices indicate nearly perfect separation between classes, which raises questions about robustness when moving to larger, noisier, and more diverse datasets. Third, training over only five epochs may not fully explore model convergence dynamics, although early stopping indicators suggest stable performance.

5 Conclusion

In this work, we explored transformer-based methods for forest fire detection from acoustic signals. Three approaches were compared: transfer learning with a pretrained Vision Transformer (ViT), the pretrained Audio Spectrogram Transformer (AST) as a frozen feature extractor, and parameter-efficient fine-tuning of AST with Low-Rank Adaptation (LoRA). Both methods using AST demonstrated strong performance on the combined ESC-50 and wildfire datasets. The baseline AST achieved perfect classification accuracy, while the AST+LoRA model reached nearly the same performance while updating only 0.34% of the parameters.

These results confirm that transformer architectures are highly effective for audio-based fire detection, and that LoRA provides a practical means to adapt large pretrained models with minimal computational cost. This efficiency is especially valuable for deployment in real-time monitoring systems and resource-constrained environments such as IoT devices in forested areas.

Future work will focus on expanding the dataset with real-world wildfire recordings, evaluating robustness in noisy outdoor conditions, and integrating the system into low-power embedded platforms to enable scalable and reliable early-warning networks.

References

[1] Forest wildfire sound dataset. `https://www.kaggle.com/datasets/teertha/us-forest-wildfire-sound-dataset`, 2020.

[2] Saman Arzaghi. Audio pre-processing for deep learning. Technical report, University of Tehran, School of Mathematics, Statistics and Computer Science, December 2020.

[3] Robert-Nicolae Boştinaru, Nicu Bizon, Sebastian-Alexandru Drăguşin, Gabriel-Vasile Iana, and Denisa Toma. Dimensionality reduction with principal component analysis for fire and non-fire audio classification: A new approach. In *Proceedings of the 17th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, pages 1–7. IEEE, 2025.

[4] Rytis Damaševičius, Achmad Qurthobi, and Rytis Maskeliunas. A hybrid machine learning model for forest wildfire detection using sounds. pages 99–106, 2024.

[5] Misha Denil, Babak Shakibi, Laurent Dinh, Marc'Aurelio Ranzato, and Nando de Freitas. Predicting parameters in deep learning. *arXiv preprint arXiv:1306.0543*, 2013.

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[7] Carl Eckart and Gale Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.

[8] Hugging Face. Audio data course. `https://huggingface.co/learn/audio-course`, 2020. Accessed 2025.

[9] Hugging Face. Vision transformer (vit) - google/vit-base-patch16-224, 2020. `https://huggingface.co/google/vit-base-patch16-224`.

[10] Hugging Face. Audio spectrogram transformer (ast), 2022. `https://huggingface.co/models`.

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

[12] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[13] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 2019.

[14] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Wang, and Yongdong Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[15] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[16] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3045–3059. Association for Computational Linguistics, 2021.

[17] John Martinsson, Magnus Runefors, Henrik Frantzich, et al. A novel method for smart fire detection using acoustic measurements and machine learning: Proof of concept. *Fire Technology*, 58:3385–3403, 2022.

[18] Giovanni Peruzzi, Andrea Pozzebon, and Marco Vanetti. Fight fire with fire: Detecting forest fires with embedded machine learning models dealing with audio and images on low power iot devices. *Sensors*, 23(2):783, 2023.

[19] Karol Piżczak. Esc-50: Dataset for environmental sound classification. `https://github.com/karolpiczak/ESC-50`, 2015.

[20] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

[21] Author(s) Unknown. Advanced audio signal processing methods for automatic classification of "fire" and "fireless" sounds. *Journal/Conference name*, 2025. As cited in FMR Biblio study.

[22] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[23] Boyang Zhang, Jared Leitner, and Sam Thornton. Audio recognition using mel spectrograms and convolutional neural networks. Technical Report Report 38, ECE228, University of California, San Diego, 2019.