MAE-AST: Masked Autoencoding Audio Spectrogram Transformer

From the original paper “MAE-AST: Masked Autoencoding Audio Spectrogram Transformer” (Alan Baade, et al.)

Presenter: Edoardo Cappelli

June 2025

UNIVERSITÀ
DEGLI STUDI
FIRENZE

Da un secolo, oltre.

From CNNs to Transformers

Traditional Approach (≈10 years): CNNs for audio recognition.

- Audio → Spectrogram (Image)
- Spectrogram → CNN → Feature Extraction → Classification

CNN Limitation: Excellent at learning **local patterns**, but struggle with **global context**.

Early Hybrid Solutions:

- CNN (for local features) → Transformer (for global relations).

The Big Question:

Do we really need the CNN part?

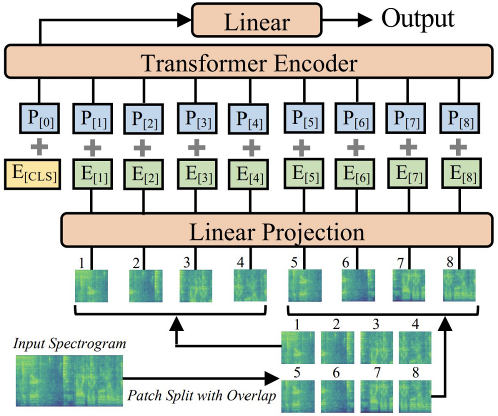
UNIVERSITÀ DEGLI STUDI FIRENZE
Da un secolo, oltre.

AST: Audio Spectrogram Transformer (May 2021)

Core Idea: A pure Transformer can learn directly from spectrogram patches, making CNNs redundant for audio classification.

How it Works:

1. **Input:** Log-mel spectrogram (e.g., a 10s audio clip becomes a 128x1000 "image").
2. **Patching:** The spectrogram is divided into overlapping 16x16 patches.
3. **Embedding:** Each patch is flattened and linearly projected into a 768-dim vector.
4. **[CLS] Token:** A special token is prepended to the sequence to aggregate information for classification.
5. **Positional Embedding:** Learnable embeddings are added to give the model spatial awareness.
6. **Transformer Encoder:** Processes the full sequence of embeddings.
7. **Output:** The final representation of the [CLS] token is fed to a classification head.


Linear
Output
Transformer Encoder
$P_{[0]}$ $P_{[1]}$ $P_{[2]}$ $P_{[3]}$ $P_{[4]}$ $P_{[5]}$ $P_{[6]}$ $P_{[7]}$ $P_{[8]}$
$E_{[CLS]}$ $E_{[1]}$ $E_{[2]}$ $E_{[3]}$ $E_{[4]}$ $E_{[5]}$ $E_{[6]}$ $E_{[7]}$ $E_{[8]}$
Linear Projection
1 2 3 4 5 6 7 8
Input Spectrogram
Patch Split with Overlap
1 2 3 4
5 6 7 8

AST: Leveraging Pre-trained Vision Models

Challenge: Transformers are data-hungry, and large, labeled audio datasets are rare.

Solution: Transfer Learning.

- Use a Vision Transformer (ViT) pre-trained on ImageNet.

Adaptation for Audio:

1. **Input Channels:** Average the ViT's input layer weights (trained on 3-channel RGB images) to create a single-channel filter for spectrograms.
2. **Positional Embeddings:** The pre-trained positional map (e.g., 24x24 for images) is adapted to the spectrogram's rectangular shape (e.g., 12x100) using bilinear interpolation.

Key Insight: Knowledge learned from natural images can be effectively transferred to the visual representation of sound.

SSAST: Self-Supervised AST (Feb 2022)

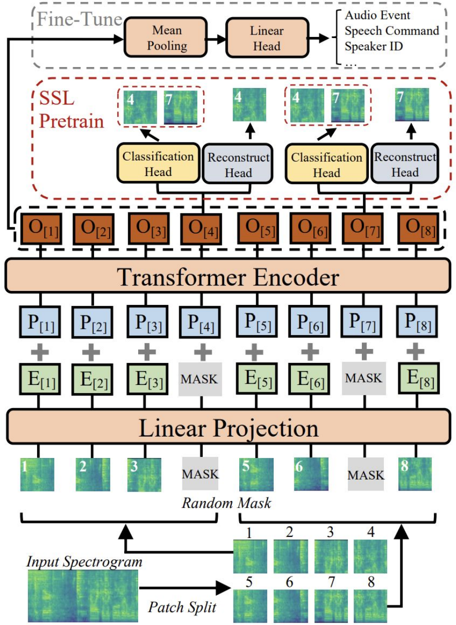
Problem: The original AST relied on supervised pre-training (ImageNet).

How can we pre-train directly on audio without massive labeled datasets?

Idea: Apply **Self-Supervised Learning (SSL)** to the AST architecture using unlabeled audio data.

Pre-training Concept:

1. Divide the spectrogram into non-overlapping patches.
2. **Mask** a high percentage of these patches.
3. Train the model to **predict the original content** of the masked patches based on the visible ones.



SSAST: Pre-training Tasks

The model learns robust audio representations by solving two pretext tasks simultaneously:

- **1. Generative Task (Reconstruction):**
  - **Goal:** Reconstruct the spectrogram content of the masked patches.
  - **Learns:** Fine-grained local patterns and spectral structures.
  - **Loss:** Mean Squared Error (MSE).

- **2. Discriminative Task (Contrastive):**
  - **Goal:** For each masked position, identify the correct original patch from a set of "negative" candidates (the other masked patches from the same audio clip).
  - **Learns:** Global, distinctive, and high-level features.
  - **Loss:** InfoNCE Loss.

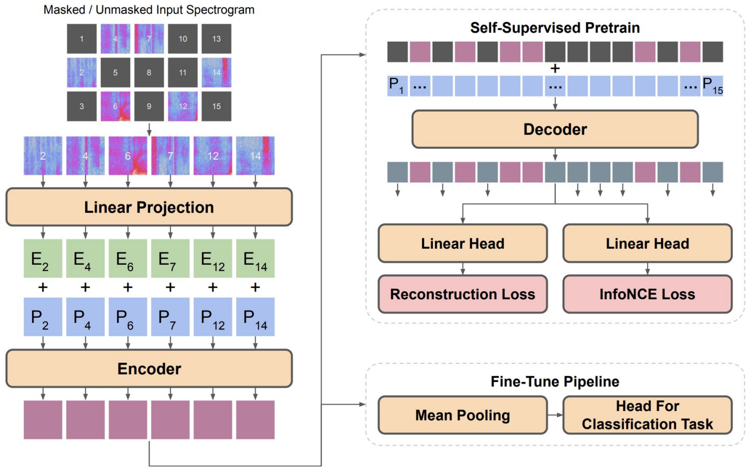
MAE-AST: Masked Autoencoders for Audio (Sept 2022)

Problem with SSAST: The Transformer encoder processes the **entire sequence**, including visible patches and special `[MASK]` tokens. This is computationally inefficient as most of the input (e.g., 75%) is masked.

MAE-AST Solution: Asymmetric Encoder-Decoder

1. **Encoder:** A deep Transformer (e.g., 6 layers) processes **only the visible patches** (e.g., 25% of the total). This is fast and memory-efficient.
2. **Decoder:** A shallow, lightweight Transformer (e.g., 2 layers) takes the encoded patch representations, re-introduces `[MASK]` tokens at their original positions, and reconstructs the full spectrogram.

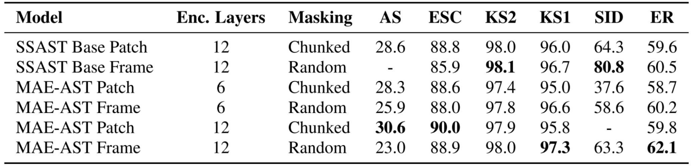
Key Advantage: Focuses computational power on learning from the known information (visible patches).



AST vs. SSAST vs. MAE-AST

	AST	**SSAST**	**MAE-AST**
Pre-training	Supervised on ImageNet	Self-Supervised	Self-Supervised
Efficiency	Baseline	Encoder processes all tokens	Encoder sees only visible tokens
Fine-Tuning Aggregation	`[CLS]` Token	Mean Pooling	Mean Pooling
Patch Overlap	Always Used	Pre-training: No Fine-tuning: Yes	Never Used

Objective of my work



Model	Enc. Layers	Masking	AS	ESC	KS2	KS1	SID	ER
SSAST Base Patch	12	Chunked	28.6	88.8	98.0	96.0	64.3	59.6
SSAST Base Frame	12	Random	-	85.9	**98.1**	96.7	**80.8**	60.5
MAE-AST Patch	6	Chunked	28.3	88.6	97.4	95.0	37.6	58.7
MAE-AST Frame	6	Random	25.9	88.0	97.8	96.6	58.6	60.2
MAE-AST Patch	12	Chunked	**30.6**	**90.0**	97.9	95.8	-	59.8
MAE-AST Frame	12	Random	23.0	88.9	98.0	**97.3**	63.3	**62.1**

The goal is to implement and evaluate MAE-AST, focusing on how different masking strategies impact performance on diverse audio tasks.

- **Model:** MAE with a 6-layer Transformer Encoder and a 2-layer Decoder.
- **Masking strategies:**
  - **Chunked Patches**
  - **Random Frames**
- **Downstream Tasks:**
  - **Audio Event Classification:** ESC-50
  - **Speaker Identification:** VoxCeleb1
- **Core Investigation:** Comparing two masking strategies trying to replicate table 1 of the original paper.

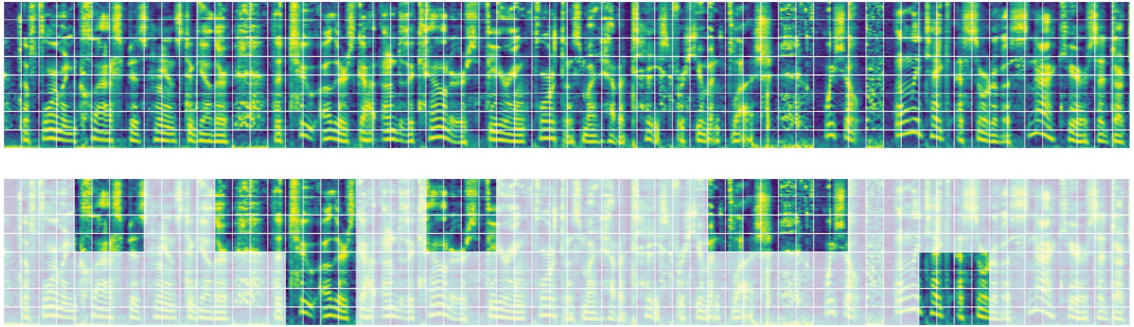
Università degli Studi di Firenze

Da un secolo, oltre.

Masking Strategies

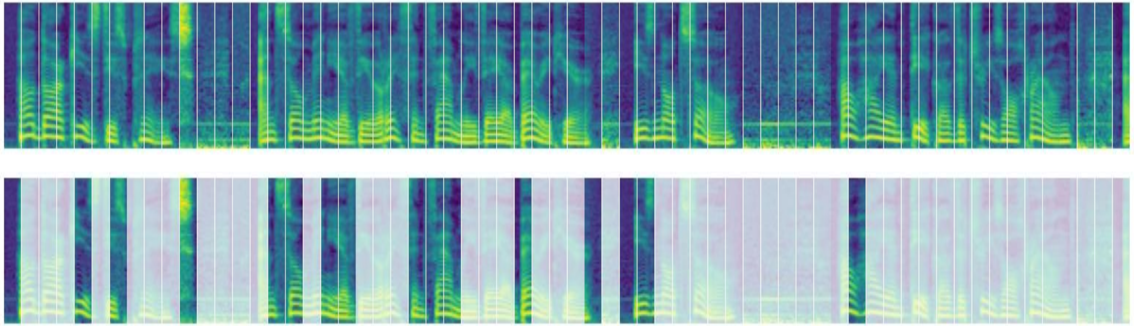
Patch-Chunk based Masking

- The spectrogram is a 2D grid of patches (e.g., 16x16).
- Randomly sample contiguous C×C blocks until the desired fraction of patches is masked.
- Encourages the model to learn generic time-frequency patterns from surrounding context



Random Frame based Masking

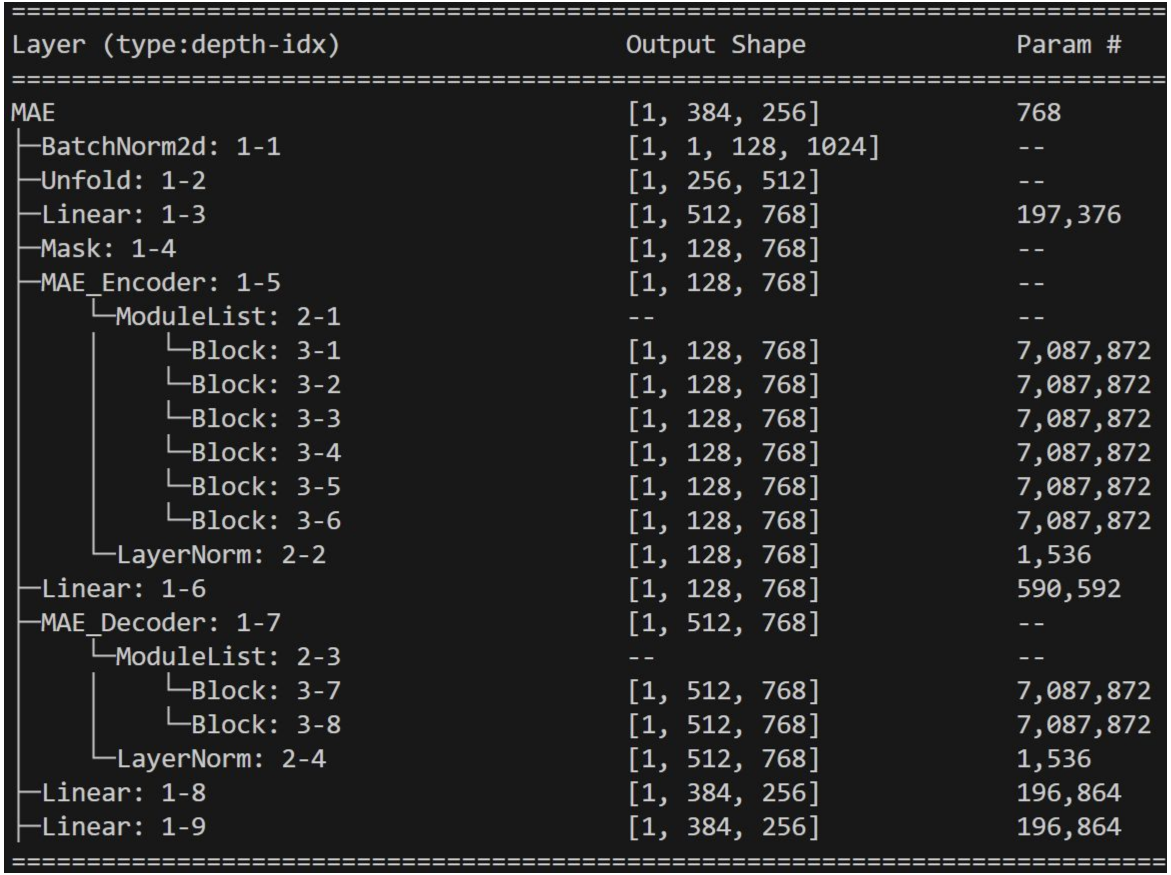
- The spectrogram is divided into vertical slices (e.g., 128x2). A random subset of these full-frequency frames is discarded.
- A harder task that forces the model to learn temporal dynamics and relationships over time, as it cannot "cheat" by using adjacent frequency information.



MY-MAE-AST

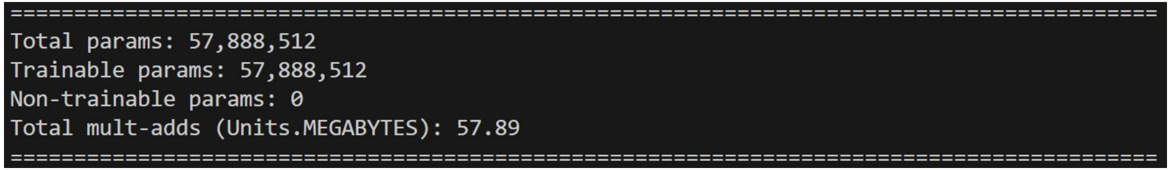
Encoder
enc_embed_dim=768,
enc_hidden_layers = 6,
enc_attention_heads = 12
enc_mlp_ratio = 4

Decoder
dec_embed_dim = 768,
dec_hidden_layers = 2,
dec_attention_heads = 12,
dec_mlp_ratio = 4



```
==============================================================================
Layer (type:depth-idx)                   Output Shape              Param #
==============================================================================
MAE                                      [1, 384, 256]             768
├─BatchNorm2d: 1-1                       [1, 1, 128, 1024]         --
├─Unfold: 1-2                            [1, 256, 512]             --
├─Linear: 1-3                            [1, 512, 768]             197,376
├─Mask: 1-4                              [1, 128, 768]             --
├─MAE_Encoder: 1-5                       [1, 128, 768]             --
│    └─ModuleList: 2-1                   --                        --
│    │    └─Block: 3-1                   [1, 128, 768]             7,087,872
│    │    └─Block: 3-2                   [1, 128, 768]             7,087,872
│    │    └─Block: 3-3                   [1, 128, 768]             7,087,872
│    │    └─Block: 3-4                   [1, 128, 768]             7,087,872
│    │    └─Block: 3-5                   [1, 128, 768]             7,087,872
│    │    └─Block: 3-6                   [1, 128, 768]             7,087,872
│    └─LayerNorm: 2-2                    [1, 128, 768]             1,536
├─Linear: 1-6                            [1, 128, 768]             590,592
├─MAE_Decoder: 1-7                       [1, 512, 768]             --
│    └─ModuleList: 2-3                   --                        --
│    │    └─Block: 3-7                   [1, 512, 768]             7,087,872
│    │    └─Block: 3-8                   [1, 512, 768]             7,087,872
│    └─LayerNorm: 2-4                    [1, 512, 768]             1,536
├─Linear: 1-8                            [1, 384, 256]             196,864
├─Linear: 1-9                            [1, 384, 256]             196,864
==============================================================================
```

Params



```
==========================================================================================
Total params: 57,888,512
Trainable params: 57,888,512
Non-trainable params: 0
Total mult-adds (Units.MEGABYTES): 57.89
==========================================================================================
```

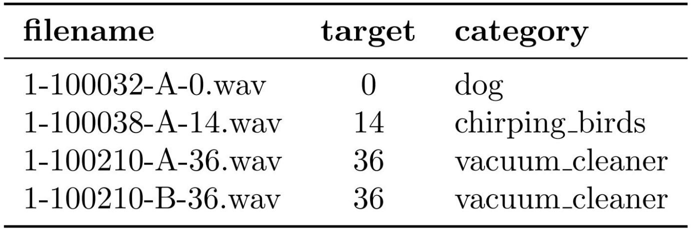
Datasets: Pre-training & Fine-tuning

Pre-training (My Setup):

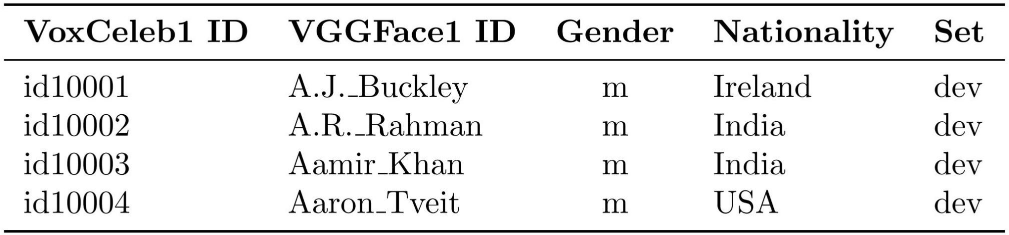
- A small-scale dataset to establish a proof-of-concept.
- **AudioSet:** 10739 clips (~30 hours)
- **LibriSpeech:** 14269 clips (~39 hours)
- *Note: This is significantly smaller than the datasets used in the original papers (~6,500 hours), which is a key limitation.*

Fine-tuning (Downstream Tasks):

- **ESC-50:** 2,000 5-second clips of environmental sounds across 50 classes *(~2.7 hours)*. Metric: **Accuracy**
- **VoxCeleb1:** ~200 hours of speech from 1,251 speakers. Metric: **Accuracy**.

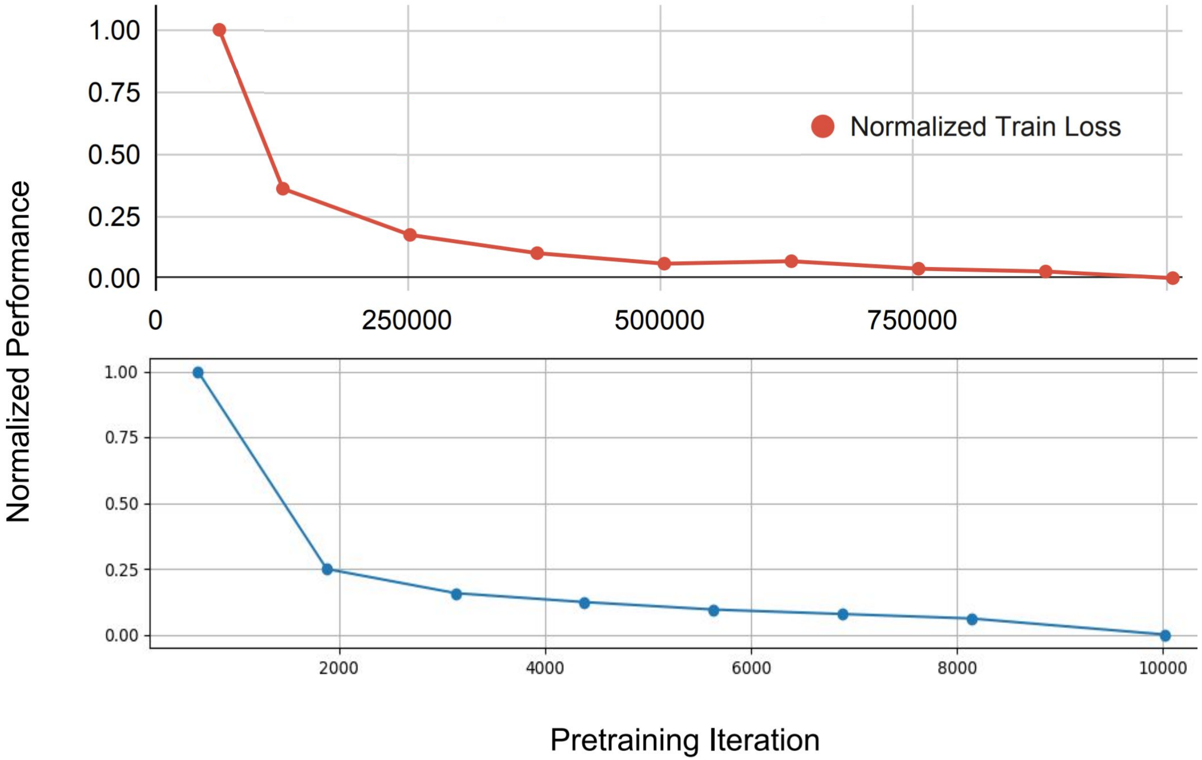


filename	target	category
1-100032-A-0.wav	0	dog
1-100038-A-14.wav	14	chirping_birds
1-100210-A-36.wav	36	vacuum_cleaner
1-100210-B-36.wav	36	vacuum_cleaner

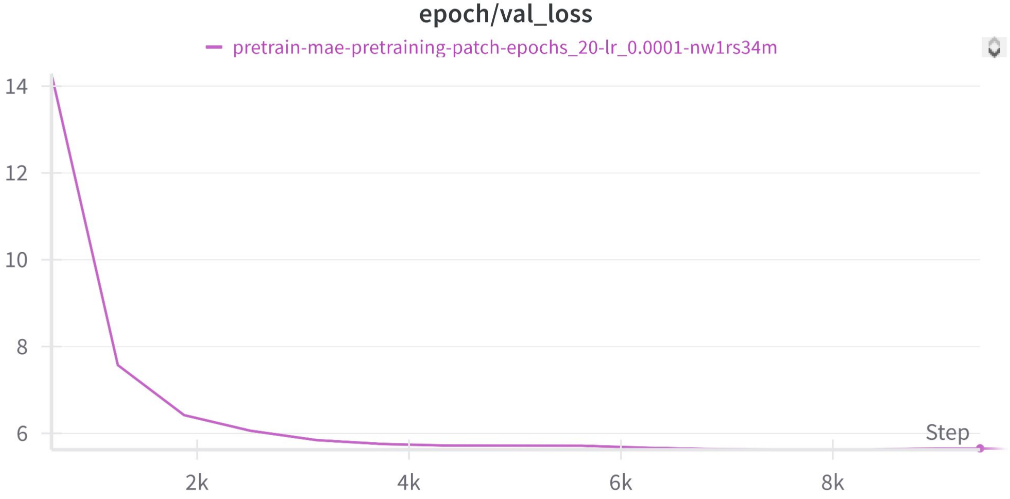


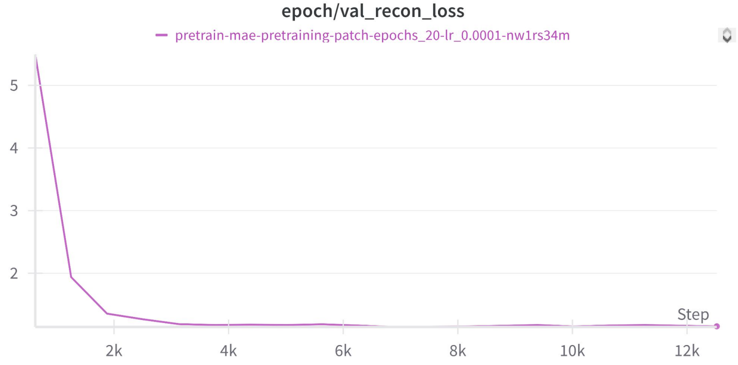
VoxCeleb1 ID	VGGFace1 ID	Gender	Nationality	Set
id10001	A.J._Buckley	m	Ireland	dev
id10002	A.R._Rahman	m	India	dev
id10003	Aamir_Khan	m	India	dev
id10004	Aaron_Tveit	m	USA	dev

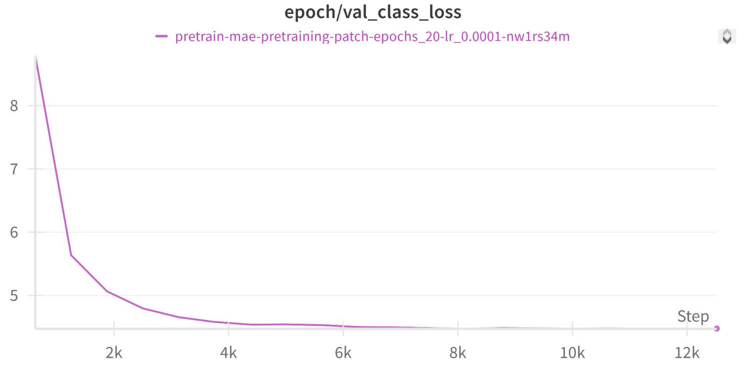
Pre-training Results (MAE-AST vs MY-MAE-AST)



Pre-training on LibriSpeech & AudioSet


epoch/val_loss


epoch/val_recon_loss


epoch/val_class_loss

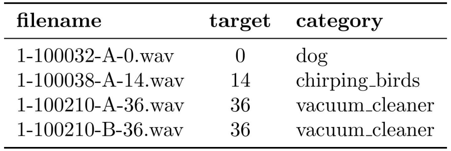
Experiments - ESC & VoxCeleb Fine Tuning

Analyze how different MAE-AST masking strategies impact performance in *Audio Event Classification* on the ESC-50 and VoxCelebe1 dataset.

- **Model**: MAE-AST with a 6-layer Transformer Encoder and a 2-layer Decoder.
- **Pre-training (My Setup)**:
  - A small-scale dataset (AudioSet + LibriSpeech, ~69 hours) to validate the proof-of-concept.
  - *Note*: This is significantly smaller than the datasets used in the original papers (~6,500 hours).
- **Masking Strategies Compared**:
  - Patch-based Masking
  - Frame-based Masking

ESC Fine-Tuning

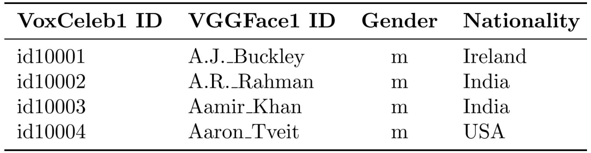
- **Dataset**: ESC-50 (2,000 5-second clips, 50 environmental event classes, ~2.7 hours).
- **Metric**: Accuracy.



filename	target	category
1-100032-A-0.wav	0	dog
1-100038-A-14.wav	14	chirping_birds
1-100210-A-36.wav	36	vacuum_cleaner
1-100210-B-36.wav	36	vacuum_cleaner

VoxCeleb1 Fine-Tuning

- **Dataset**: VoxCeleb1 (2,000 5-second clips, 1251 speakers, ~200 hours).
- **Metric**: Accuracy



VoxCeleb1 ID	VGGFace1 ID	Gender	Nationality
id10001	A.J._Buckley	m	Ireland
id10002	A.R._Rahman	m	India
id10003	Aamir_Khan	m	India
id10004	Aaron_Tveit	m	USA

Results - ESC & VoxCeleb Fine Tuning

Model	Enc. Layers	Masking	ESC
MAE-AST Patch	6	Chunked	88.6
MY-MAE Patch	6	Chunked	50.5
MAE-AST Frame	6	Random	88.0
MY-MAE Frame	6	Random	56.25

Model	Enc. Layers	Masking	SID
MAE-AST Patch	6	Chunked	37.6
MY-MAE Patch	6	Chunked	21.4
MAE-AST Frame	6	Random	58.6
MY-MAE Frame	6	Random	43.0

MY-MAE implementation shows **significantly lower performance** compared to the reference MAE-AST model.

Pre-training Impact: this performance gap is primarily attributable to the reduced size of the pre-training dataset (~69 hours) compared to standard practices (~6,500 hours). Large-scale pre-training is crucial for learning robust and generalizable audio representations.

The Frame-based strategy appears marginally more effective.

Attempts to address this by reducing the number of classes or the model's depth were unsuccessful.

Conclusions

- Implemented an **efficient MAE-AST model** for audio processing, building on the concepts of AST and SSAST.

Limitations:

- The pre-training dataset was small, which likely limited the model's full potential.

References & Code

- MAE-AST: Masked Autoencoding Audio Spectrogram Transformer
- AST: Audio Spectrogram Transformer
- SSAST: Self-Supervised Audio Spectrogram Transformer

Code at https://github.com/EdoardoCappelli/MAE-AST

MAE-AST: Masked Autoencoding Audio Spectrogram Transformer

Thank you for listening!

Self-Supervised Audio Spectrogram Transformer (SSAST)

Vengono usate **due loss**:

1. **Discriminativa (InfoNCE loss)**:
   - Il modello deve capire quale x□ è quello giusto dato c_i.
   - Formula:

$$\mathcal{L}_d = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{c_i^\top x_i}}{\sum_{j=1}^{N} e^{c_i^\top x_j}}$$

2. **Generativa (MSE loss)**:
   - La patch ricostruita r_i deve essere simile alla patch originale x_i.

$$\mathcal{L}_g = \frac{1}{N}\sum_{i=1}^{N}\|r_i - x_i\|^2$$

3. **Loss Totale**: $\mathcal{L} = \mathcal{L}_d + \lambda\mathcal{L}_g \quad \text{con } \lambda = 10$

Obiettivo generativo insegna a catturare dettagli locali e spettrali dello spettrogramma.

Obiettivo discriminativo forza la rappresentazione a essere robusta e distintiva per ogni patch, migliorando la qualità finale delle embedding per i task di classificazione downstream.

Losses

Finally, we update the weights of the AST model M to minimize L with the optimizer (line 19-20). Note that for the discriminative task, the negative samples are sampled from the same spectrogram, i.e., the model aims to pick the correct patch for each masked position from all patches being masked. On one hand, this increases the difficulty of the pretext task to avoid the model learning trivial things such as recording environment for prediction; on the other hand, this also avoids building a memory bank of patches from different spectrograms and makes the algorithm less computationally intensive and less affected by the mini-batch size.

Masked Autoencoders Are Scalable Vision Learners

SSAST è inefficiente: Nel pretraining, viene mascherato il 75 % delle patch. Tuttavia, la maggior parte dei layer Transformer continua a calcolare l’attenzione anche sui token mascherati, sprecando computazione e memoria.

Soluzione: **scartare** completamente i token mascherati **durante la fase di encoding** passando all’encoder solo i token visibili.

Efficienza

- **3× più veloce** in pretraining rispetto a SSAST con architetture e dataset comparabili.
- **2× meno memoria** occupata grazie alla riduzione del numero di token nell’encoder.

Prestazioni

- Con lo stesso numero di layer di encoder, **MAE-AST ottiene risultati migliori** di SSAST su vari task (speech e audio classification).
- MAE-AST funziona bene anche usando *soltanto* l’obiettivo generativo di ricostruzione, mentre SSAST soffre un calo di performance se si rimuove l’obiettivo discriminativo durante il pretraining.

Masked Autoencoding Audio Spectrogram Transformer

Waveform → Spettrogramma

- Prendiamo un segnale audio a 16 kHz e lo convertiamo in log Mel filterbank di dimensione 128 (frame di 25 ms, shift di 10 ms).
- Normalizziamo ogni spettrogramma a media 0 e deviazione standard ½, come in AST/SSAST.

Tokenizzazione

- **Patch-based**: blocchi di 16 filter × 16 frame → ciascun "patch token" copre 16 bande × 16 passi temporali.
- **Frame-based**: porzioni di 128 filter × 2 frame → ogni token è più "alto" (tutte le bande) ma più stretto (solo 2 frame).
- Applichiamo quindi un masking (es. togliamo casualmente il 75 % dei token), lasciando un sottoinsieme "visibile" per l'encoder.

Masked Autoencoding Audio Spectrogram Transformer

Positional Embeddings

- Usiamo **sinusoidal embeddings 1D** (come in [12]) lungo l’asse temporale:
  - Per i patch flattenati, ordiniamo prima per canale (filtro) e poi per tempo, ottenendo una sequenza unidimensionale di token.
  - Questo ci permette di gestire sequenze audio di durata variabile senza dover ridimensionare gli embeddings.

Encoder

- **Solo sui token non mascherati**: l’encoder ViT-style riceve in input i token “visibili” (circa il 25 % del totale).
- Ogni token viene proiettato linearmente in uno spazio latente **768-dimensionale**, a cui sommiamo il corrispondente positional embedding.
- Passiamo infine questa sequenza “compressa” attraverso 6 layer Transformer (12 teste, larghezza 768).

Masked Autoencoding Audio Spectrogram Transformer

Decoder

- **Usato solo in pretraining** per ricostruire lo spettrogramma completo.
- Prende in input sia:
  1. Le uscite dell'encoder (rappresentazioni dei token visibili).
  2. Token speciali di maschera (stessa embedding appresa, più positional embeddings).
- Composto da 2 layer Transformer ("shallow"), sempre con 12 teste e width 768.
- L'output corrispondente ai token mascherati viene proiettato tramite due layer lineari, uno per ciascuna componente di loss (es. L2 sul valore del filtro e/o L1 sul log-magnitude).

Fine-tuning

- **Discardiamo completamente il decoder**: su task di classificazione (audio event, speech commands, ecc.) usiamo solo l'encoder.
- Applichiamo una **mean-pooling** sull'ultima hidden state dell'encoder per ottenere un vettore fisso da passare al classificatore.

MAE-AST vs SSAST

Differenze con SSAST

Overlap dei patch

- SSAST lo toglieva in pretraining e lo rimetteva in fine-tuning; MAE-AST **non usa overlap né in pretraining né in fine-tuning**, per coerenza con MAE vision.

Positional embeddings

- SSAST li intercettava o interpolava per durate variabili; MAE-AST **non li modifica** durante il fine-tuning.

MAE-AST Masking

Obiettivi del masking

1. **Difficoltà del compito**

   ○ Se mascheri troppo poco o in modo troppo “sparso” (fully random con bassa percentuale), il modello può semplicemente interpolare le parti mancanti usando i pixel/spettrogrammi adiacenti.

   ○ Se mascheri troppo molto, il compito diventa eccessivamente difficile e il modello fatica a ricostruire informazioni davvero utili.

2. **Efficienza computazionale**

   ○ Nei Transformer, computazione e memoria crescono quadraticamente con il numero di token. Un masking aggressivo (alto p) riduce la lunghezza dell’input al decoder, generando significativi speedup.

MAE-AST Pre-training Dataset

- **AudioSet2M**
  - ~2 milioni di clip audio da YouTube, ciascuno di 10 secondi. (circa 5.500 ore)
  - Contiene etichette per eventi audio generici (sirene, clacson, pioggia…) e per il parlato.
  - È diviso in tre partizioni:
    - **Unbalanced**: distribuzione “naturale” delle classi (molte clip di alcune classi, poche di altre).
    - **Balanced**: sottoinsieme con numero simile di clip per etichetta.
    - **Eval** (evaluation): altro sottoinsieme bilanciato, usato solo in test.
- **LibriSpeech**
  - ~960 ore di audiolibri in inglese.
  - Incluso perché AudioSet non garantisce presenza di parlato continuo.

Cosa viene usato

- Tutte le clip (labels scartate) delle partizioni **Unbalanced** e **Balanced** di AudioSet2M.
- Tutte le clip di training di LibriSpeech.

MAE-AST Finetuning Dataset

Audio Event Classification

- **ESC-50 (ESC)**: 2.000 clip di suoni ambientali (pioggia, animali, strumenti, ecc.).
- ESC-50 We use the ESC-50 dataset (Piczak 2015) for single audio event classification task. ESC-50 is an audio classification dataset consists of 2,000 5-second environmental audio recordings organized into 50 classes. For this task, we follow the standard 5-fold cross-validation to evaluate our model and report the accuracy. The difference between ESC-50 and AudioSet-20K is that each ESC-50 audio clip only contains a single event and the total data volume size is 10 times smaller than AudioSet-20K.

Speech Classification

- **VoxCeleb1 (SID)**: identificazione dello speaker. VoxCeleb 1 We use VoxCeleb 1 dataset (Nagrani et al. 2020) for the speaker identification task. The VoxCeleb 1 dataset contains 352 hours of speech from 1,251 speakers. The goal of this task is to classify each utterance for its speaker identity where speakers are in the same predefined set for both training and testing. For VoxCeleb 1, we use the SUPERB evaluation framework (Yang et al. 2021) and report the accuracy on the test set.

My Dataset

Pretraining

1000 clip da AudioSet (≈2.78 ore), per mantenere le stesse proporzioni del paper dovresti usare:

~29 minuti di LibriSpeech (≈ 0.48 ore).

LibriSpeech

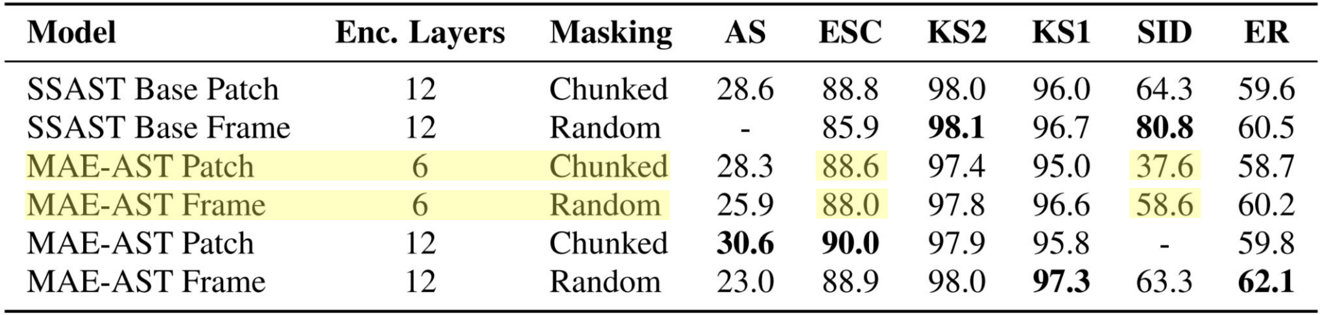
```
find data/LibriSpeech -type f -name '*.flac' | wc -l
28539

find /home/ing2025edocap/snap/snapd-desktop-integration/253/Scrivania/DeepLearning/data/LibriSpeech -type f -name '*.flac' -print0 | du
--files0-from=- -ch | tail -n1
6,3G    totale
```

79 ore di audio

Finetuning

Results of the paper that I want to replicate



Model	Enc. Layers	Masking	AS	ESC	KS2	KS1	SID	ER
SSAST Base Patch	12	Chunked	28.6	88.8	98.0	96.0	64.3	59.6
SSAST Base Frame	12	Random	-	85.9	**98.1**	96.7	**80.8**	60.5
MAE-AST Patch	6	Chunked	28.3	88.6	97.4	95.0	37.6	58.7
MAE-AST Frame	6	Random	25.9	88.0	97.8	96.6	58.6	60.2
MAE-AST Patch	12	Chunked	**30.6**	**90.0**	97.9	95.8	-	59.8
MAE-AST Frame	12	Random	23.0	88.9	98.0	**97.3**	63.3	**62.1**

Utilizzati tutti i 2000 campioni da ESC-50.
Dataset suddiviso: 1800 campioni per il training, 200 per la validation.

Key Results

Risultato chiave (come in MAE-vision): **anche con soli 2 layer di decoder** si ottengono quasi gli stessi risultati di uno con 12 layer.

Ciò significa che **il decoder può restare “leggero”** (poche decine di layer), mentre l’effettivo “lavoro” di rappresentazione viene fatto dall’encoder.

Dataset

Dataset 1 – AudioSet-2M

- È un dataset enorme di **2 milioni** di clip audio da 10 secondi ciascuna, estratte da video YouTube.
- È **multi-label**: ogni clip può appartenere a più di **527 classi audio**, come:
  - suoni umani
  - animali
  - strumenti
  - musica
  - suoni ambientali ecc.
- **Nota**: Anche se circa la metà delle clip contengono parlato, **questo parlato può essere solo una piccola parte della clip**, quindi **non è sufficiente da solo per addestrare bene il modello sullo speech**.

Pretraining Datasets

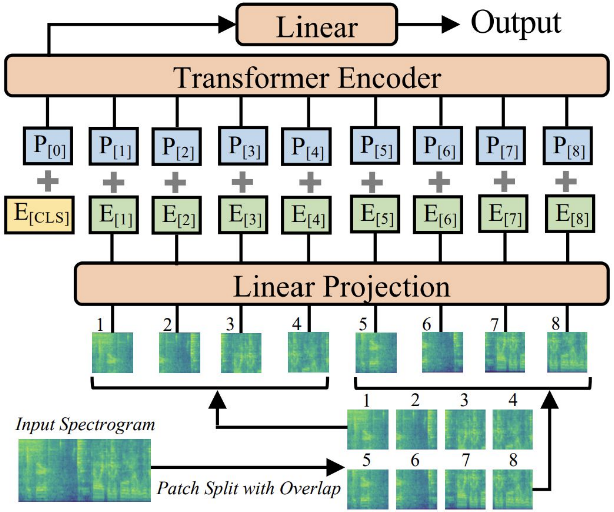
Dataset 2 – Librispeech

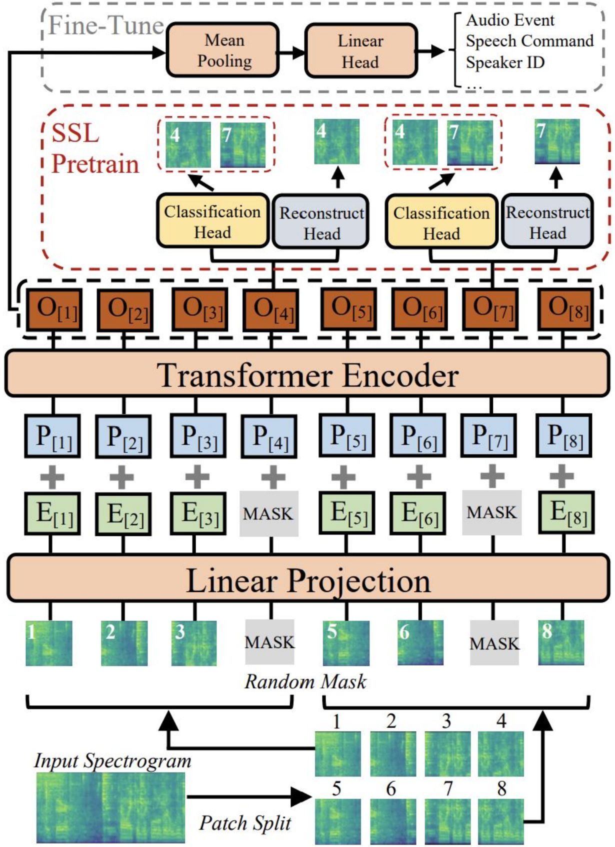
- È un dataset standard per **speech recognition**.
- Contiene **960 ore** di audiolibri in inglese letti da oltre **1.000 speaker**.
- Fornisce una copertura **più completa del parlato** rispetto ad AudioSet.

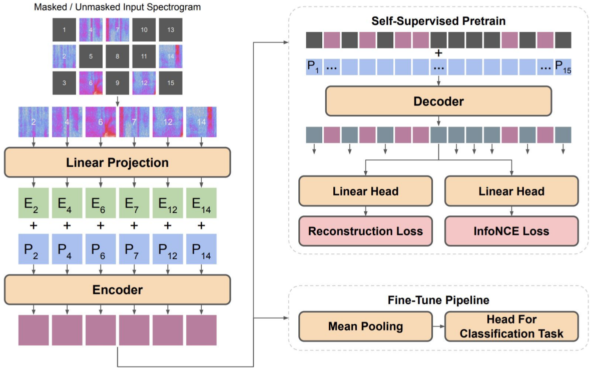
Come vengono usati

- I dati audio sono **tagliati o riempiti a 10 secondi**.
- Totale:
  - 1.953.000 clip da AudioSet
  - 281.000 clip da Librispeech
  - Totale complessivo = **2.234.000 clip**
- Le clip dei due dataset vengono **mescolate e mischiate casualmente** durante il pretraining.

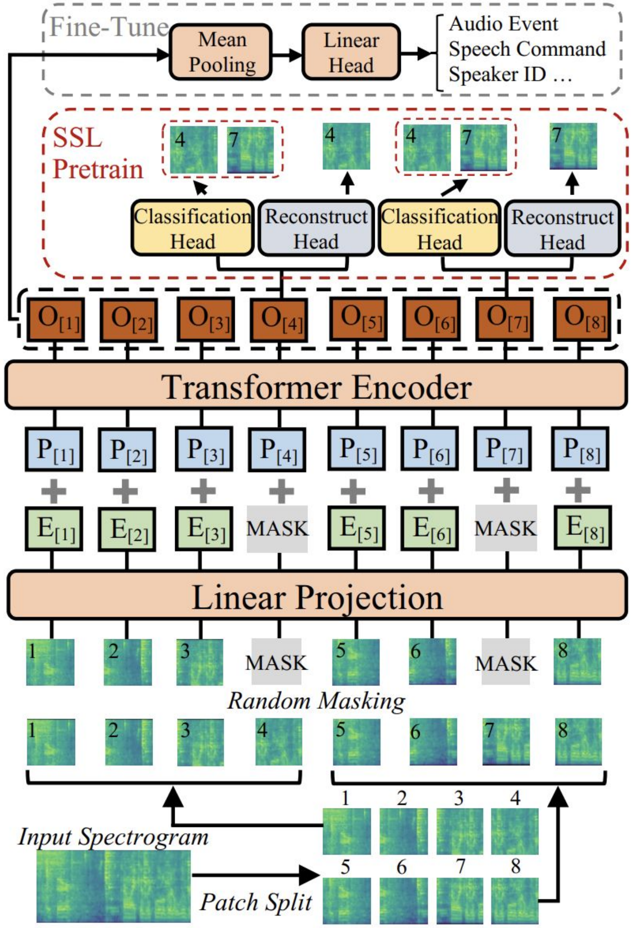
AST vs SSAST vs MAE-AST







Self-Supervised Audio Spectrogram Transformer (SSAST)



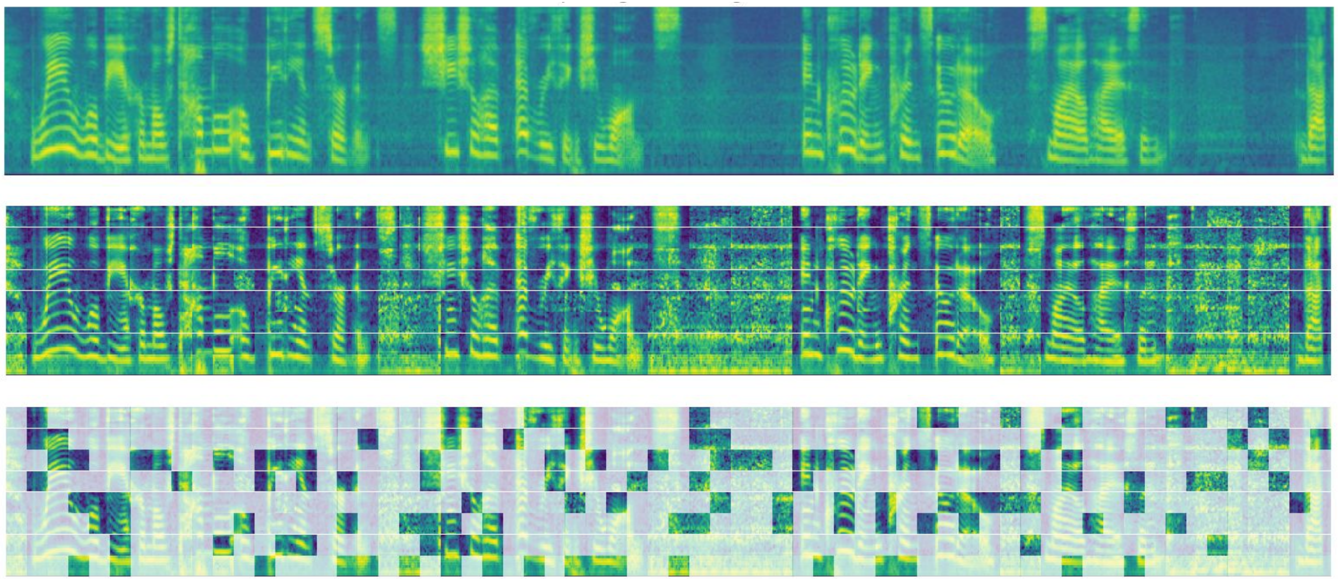
Why “Y”?

Problems, idea

Why “Z”?

Problems, idea

Random Masking



Patch size (16,16)
Spectrogram size (128,1024)
Num patches 512
Mask percentage 75%

Experiments

Results

My results vs paper ones

Results

My results vs paper ones

Results

My results vs paper ones

Results

My results vs paper ones

Results

My results vs paper ones