Forest Sounds Classification

Project on Deep Learning Course

MSc in Data Science

Nikolaos Paraskakis / I.D.: 2321

Dimitrios Tselentis / I.D.: 2325

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΛΟΠΟΝΝΗΣΟΥ
UNIVERSITY of the PELOPONNESE

ΔΗΜΟΚΡΙΤΟΣ
DEMOKRITOS

Target Dataset – FSC22

- Year: 2023
- Clips: 2025
- Clip Length: 5 seconds
- Duration: Not explicitly mentioned (≈ 2.81 hours)
- Classes: 27
- Balanced: 75 per class
- Task: Multi-class
- Source: Freesound
- Domain/Task: Forest environmental sounds

FSC22 Dataset Classes

Category	Example Class
Mechanical Sounds	Axe
Mechanical Sounds	Chainsaw
Mechanical Sounds	Handsaw
Mechanical Sounds	Generator
Animal Sounds	Birdsong
Animal Sounds	Insect buzz
Animal Sounds	Mammal call
Animal Sounds	Amphibian croak
Environmental Sounds	Wind
Environmental Sounds	Rain
Environmental Sounds	Thunder
Environmental Sounds	River
Vehicle Sounds	Car
Vehicle Sounds	Truck
Vehicle Sounds	Motorbike
Forest Threat Sounds	Fire
Forest Threat Sounds	Tree felling
Forest Threat Sounds	Gunshot
Human Sounds	Footsteps
Human Sounds	Speech
Human Sounds	Shouting

Universal Preprocessing

For each audio .wav file we apply the following preprocessing:

- Resample (if necessary) audio signal from *original_sample_rate* to *target_sample_rate*
- Mix down (if necessary) audio signal from stereo to mono
- Cut (if necessary) audio signal to have 5s (or 5xSAMPLE_RATE) length
- Right pad (if necessary) audio signal to have 5s (or 5xSAMPLE_RATE) length

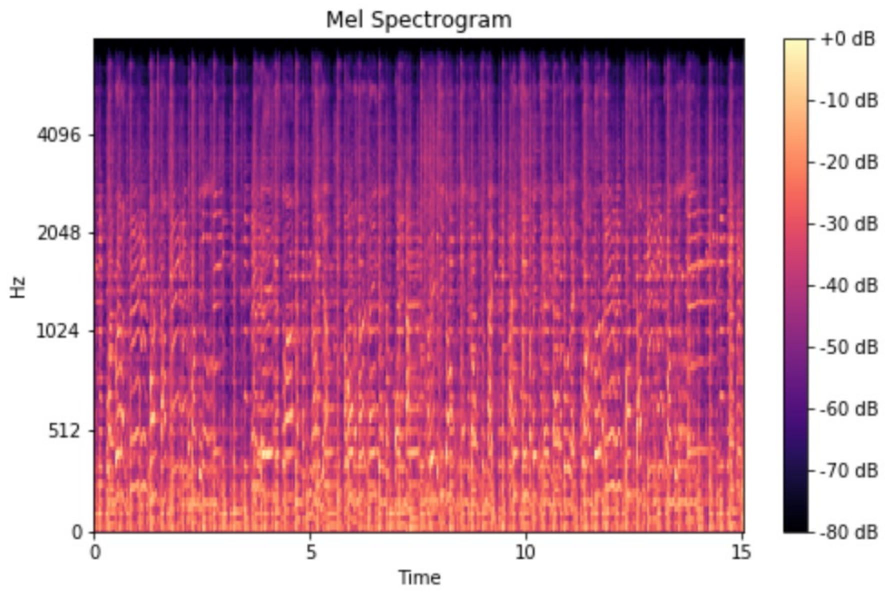
Mel-Spectrograms

Extracting a mel-spectrogram representation for each audio .wav file of the following format:

- Using *torchaudio.transforms.MelSpectrogram* with parameters like *sample_rate=22050, n_fft=2048, hop_length=512, n_mels=128*
- Using *librosa.power_to_db* to convert the power spectrogram to a decibel (dB) scale

Finally, we get mel-spectrograms of shape [128, T]

In this case, *T=216*.



Audiofeatures

Extracting a representation for each audio .wav file of the following format:

- MFCCs (Mel-Frequency Cepstral Coefficients) : [13, T]
- Chroma STFT (Short-Time Fourier Transform) : [12, T]
- Tonnetz (Tonnetz Representation) : [6, T]
- Spectral Contrast : [7, T]
- Spectral Centroids : [1, T]
- Spectral Bandwidth : [1, T]
- Spectral Flatness : [1, T]
- Spectral Rolloff : [1, T]
- Zero Crossing Rate : [1, T]

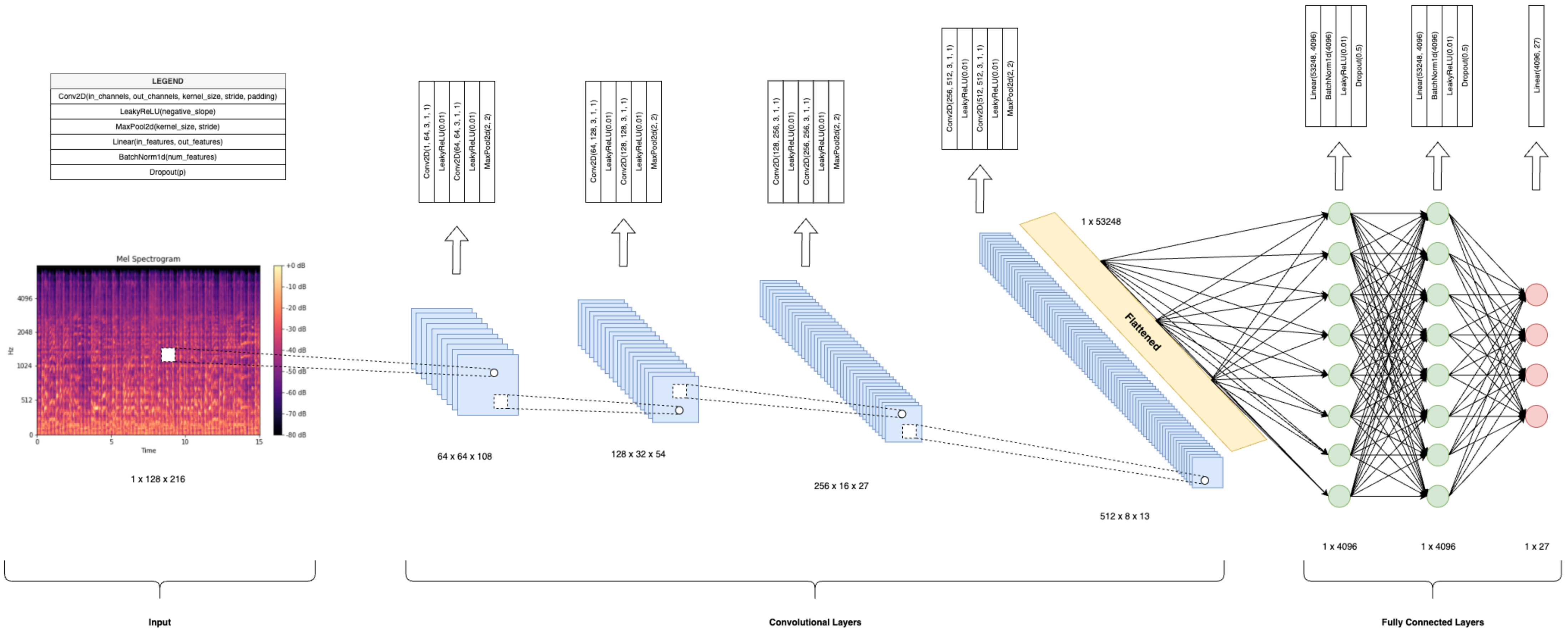
Dimension T depends on the length of the audio signal and the parameters *hop_length=512* and *n_fft=2048*.

Specifically, T represents the number of frames (or time steps) into which the audio signal is divided during feature extraction.

Finally, after concatenation we have a representation of shape [43, T].

Here, also, *T=216*.

Baseline Model



Training the Baseline Model on FSC22

- It is a multi-class classification problem
- Loss Function: *CrossEntropyLoss( )*
- During training we monitor average batch accuracy on the validation set
- Early Stopping Patience: 25 epochs
- Learning Rate Sceduler: ReduceLROnPlateau by monitoring average batch accuracy on the validation set
- Learning Rate Sceduler Patience: 10 epochs
- Starting with Learning Rate: 1e-5

Performance of the Baseline Model

EPOCHS : 80

MELSPECTROGRAMS

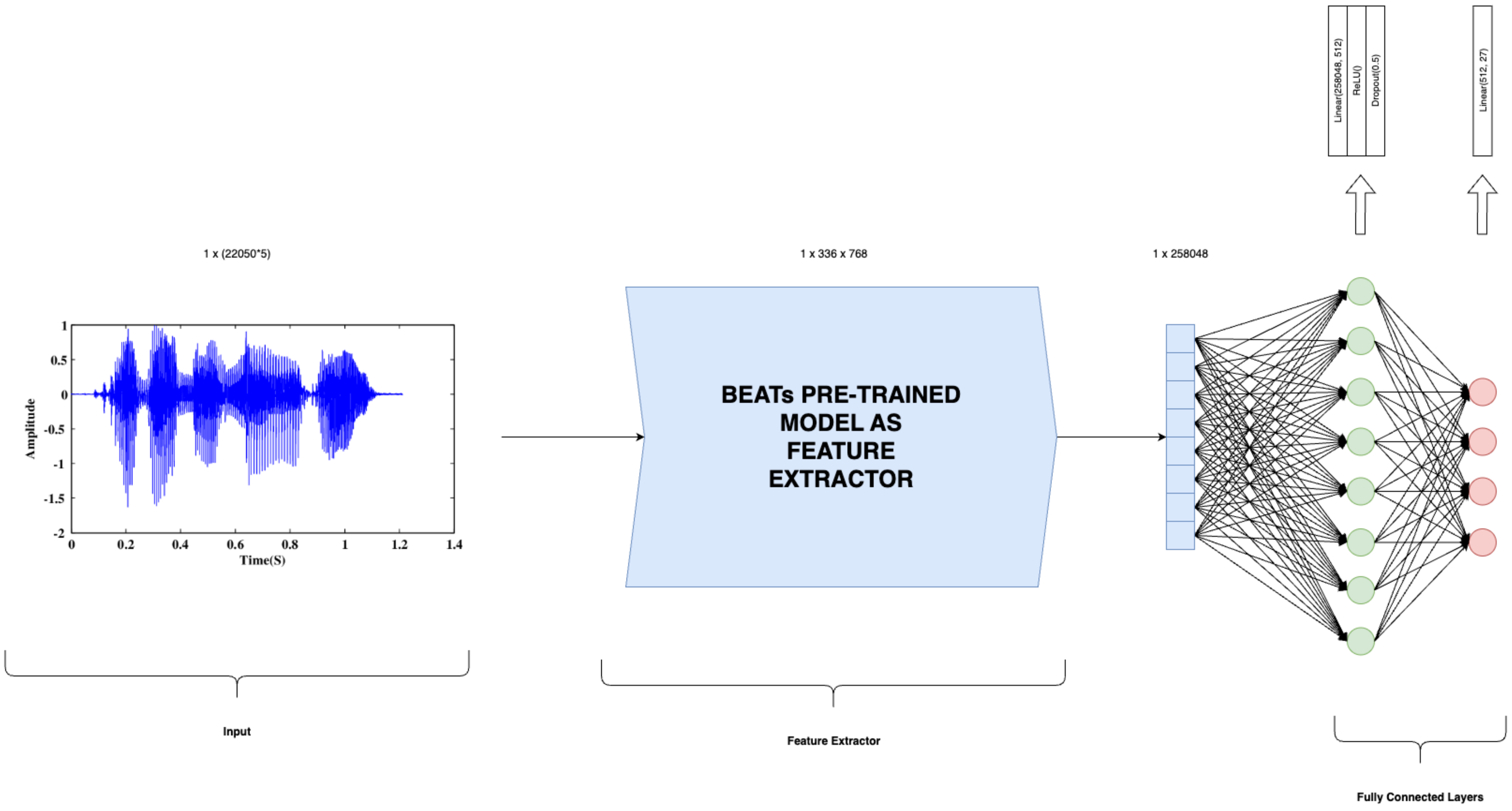
Metric	Training Set	Validation Set	Test Set
Accuracy	100%	66%	70%
Precision	100%	67%	71%
Recall	100%	66%	70%
F1 Score	100%	65%	69%
Loss Function	0.011	1.216	1.133

EPOCHS : 148

AUDIOFEATURES

Metric	Training Set	Validation Set	Test Set
Accuracy	99%	57%	56%
Precision	99%	57%	58%
Recall	99%	57%	55%
F1 Score	99%	56%	55%
Loss Function	0.105	1.525	1.434

State-of-the-art Model



Performance of the State-of-the-art Model

EPOCHS : 60

Metric	Training Set	Validation Set	Test Set
Accuracy	100%	87%	89%
Precision	100%	87%	90%
Recall	100%	87%	89%
F1 Score	100%	86%	89%
Loss Function	0.004	0.452	0.368

BEATsFEATURES

Augmentation of FSC22

We will do augmentation of the training data to boost the performance of the baseline model.

We choose a percentage of the training data to augment, while retaining proportions between classes.

Augmentation Types:

- Type A:

  Using *audiomentations.Shift* we shift the audio .wav files by a random value between -0.5 and 0.5 seconds.

- Type B:

  Using *audiomentations.Gain we randomly adjust the audio gain between -12 and 12 dB.*

  Using *audiomentations.TimeStretch* we stretch/compress the audio duration by a factor between 0.9 and 1.2 (no change of pitch).

  Using *audiomentations.Shift* we shift the audio .wav files by a random value between -0.5 and 0.5 seconds.

- Type AB:

  We do both: Type A, and Type B.

Percentage Types:

- 50%
- 100%

Training of the Baseline Model on Augmented Data

- Loss Function: *CrossEntropyLoss( )*
- During training we monitor average batch accuracy on the validation set
- Early Stopping Patience: 25 epochs
- Learning Rate Sceduler: ReduceLROnPlateau by monitoring average batch accuracy on the validation set
- Learning Rate Sceduler Patience: 10 epochs
- Starting with Learning Rate: 1e-5

Performance of the Baseline Model on Augmented Data

EPOCHS : 79

Metric	Training Set	Validation Set	Test Set
Accuracy	100%	67%	75%
Precision	100%	69%	77%
Recall	100%	67%	75%
F1 Score	100%	67%	75%
Loss Function	0.006	1.242	0.975

MELSPECTROGRAMS
TYPE_B_50

Transfer Dataset – FSD50K

- Year: 2020
- Clips: 51,197
- Clip Length: 0.3-30 seconds
- Duration: 108 hours
- Classes: 200
- Task: Multi-label
- Source: Freesound
- Unbalanced: 97 - 14K

FSD50K Dataset Classes

Category	Example Class
Human Sounds	Speech
Human Sounds	Laugh
Human Sounds	Cry
Human Sounds	Cough
Animal Sounds	Dog bark
Animal Sounds	Cat meow
Animal Sounds	Birdsong
Animal Sounds	Cow moo
Natural Sounds	Thunderstorm
Natural Sounds	Rain
Natural Sounds	Ocean waves
Natural Sounds	Wind
Musical Instruments	Piano
Musical Instruments	Guitar
Musical Instruments	Violin
Musical Instruments	Drum
Transportation	Car
Transportation	Train
Transportation	Airplane
Transportation	Bicycle
Tools	Hammer
Tools	Drill
Tools	Saw

Training the Baseline Model on FSD50K

From the architecture shown before regarding the Baseline Model, we made the following modifications:

- Last dense layer has size of 200 (number of classes)
- Now we have a multi-label classification problem
- Loss Function: *BCEWithLogitsLoss( )*
- During training we monitor global F1 score with macro averaging on the validation set
- Early Stopping Patience: 25 epochs
- Learning Rate Sceduler: ReduceLROnPlateau by monitoring average batch accuracy on the validation set
- Learning Rate Sceduler Patience: 10 epochs
- Starting with Learning Rate: 1e-5

Performance of the Baseline Model on FSD50K

EPOCHS : 50

Metric	Training Set	Validation Set	Test Set
Accuracy	100%	99%	98%
Precision	100%	53%	63%
Recall	100%	20%	17%
F1 Score	100%	27%	23%
Loss Function	0.0007	0.049	0.069

MELSPECTROGRAMS

Transfer learning to FSC22

We have trained the Baseline Model on FSD50K.

We will do transfer learning of that model on the target dataset FSC22:

- Freeze all layers except the last fully connected layers (dense layers)
- Loss Function: *CrossEntropyLoss( )*
- During training we monitor average batch accuracy on the validation set
- Early Stopping Patience: 25 epochs
- Learning Rate Sceduler: ReduceLROnPlateau by monitoring average batch accuracy on the validation set
- Learning Rate Sceduler Patience: 10 epochs
- Starting with Learning Rate: 1e-5

Performance of the Baseline Model on FSC22 after Transfer Learning

EPOCHS : 97

Metric	Training Set	Validation Set	Test Set
Accuracy	100%	72%	77%
Precision	100%	73%	78%
Recall	100%	72%	77%
F1 Score	100%	72%	77%
Loss Function	0.0015	0.995	0.81

MELSPECTROGRAMS

Transfer Dataset – UrbanSound8K

- Year: 2014
- Clips: 8732
- Clip Length: < 4 seconds
- Duration: 9 hours
- Classes: 10
- Task: Multi-Class
- Source: Freesound

UrbanSound8K Dataset Classes

Category	Example Class
Home/Indoor Sounds	Air Conditioner
Home/Indoor Sounds	Washing Machine
Home/Indoor Sounds	Microwave
Street Sounds	Car Horn
Street Sounds	Engine Idling
Street Sounds	Jackhammer
Street Sounds	Siren
Street Sounds	Street Music
People Sounds	Children Playing
People Sounds	Talking
People Sounds	Laughing
Construction Sounds	Drilling
Construction Sounds	Hammering
Emergency Sounds	Gun Shot
Emergency Sounds	Ambulance
Park Sounds	Bird Chirping
Park Sounds	Footsteps on Grass
Park Sounds	Dog Bark

Training the Baseline Model on UrbanSound8K

From the architecture shown before regarding the Baseline Model, we made the following modifications:

- Last dense layer has size of 10 (number of classes)
- Loss Function: *CrossEntropyLoss( )*
- During training we monitor average batch accuracy on the validation set
- Early Stopping Patience: 25 epochs
- Learning Rate Sceduler: ReduceLROnPlateau by monitoring average batch accuracy on the validation set
- Learning Rate Sceduler Patience: 10 epochs
- Starting with Learning Rate: 1e-5

Performance of the Baseline Model on UrbanSound8K

EPOCHS : 54

Metric	Training Set	Validation Set	Test Set
Accuracy	100%	95%	94%
Precision	100%	95%	94%
Recall	100%	95%	95%
F1 Score	100%	95%	95%
Loss Function	0.0002	0.1982	0.2071

MELSPECTROGRAMS

Transfer learning to FSC22

We have trained the Baseline Model on UrbanSound8K.

We will do transfer learning of that model on the target dataset FSC22:

- Freeze all layers except the last fully connected layers (dense layers)
- Loss Function: *CrossEntropyLoss( )*
- During training we monitor average batch accuracy on the validation set
- Early Stopping Patience: 25 epochs
- Learning Rate Sceduler: ReduceLROnPlateau by monitoring average batch accuracy on the validation set
- Learning Rate Sceduler Patience: 10 epochs
- Starting with Learning Rate: 1e-5

We experimented with the following freezing scenarios (freeze all except):

- classifier[4:]
- classifier
- layer4, classifier [best one]

Performance of the Baseline Model on FSC22 after Transfer Learning

EPOCHS : 59

Metric	Training Set	Validation Set	Test Set
Accuracy	100%	70%	77%
Precision	100%	71%	78%
Recall	100%	70%	77%
F1 Score	100%	70%	77%
Loss Function	0.0041	0.1078	0.9689

MELSPECTROGRAMS

Transfer Learning of pretrained VGG16 from ImageNet to FSC22

We loaded VGG16 with pretrained weights (trained on ImageNet).

We did the following modifications on VGG16:

- Changed the first convolutional layers to get input images of one channel
- Resized our mel-spectrograms to 224 x 224
- Changed the last dense layer to have size 27 (number of classes)

We experimented with the following freezing scenarios (freeze all except):

- classifier[-1]
- classifier[3:]
- features[24:], avgpool, classifier
- features[17:], avgpool, classifier
- features[10:], avgpool, classifier [best one]

Performance of VGG16 on FSC22 after Transfer Learning

EPOCHS : 37

Metric	Training Set	Validation Set	Test Set
Accuracy	100%	71%	74%
Precision	100%	72%	75%
Recall	100%	71%	74%
F1 Score	100%	71%	73%
Loss Function	0.031	1.319	1.143

MELSPECTROGRAMS

Performance of all models on test set of FSC22

Model	F1 Score	Epochs
Baseline Audio Features	55%	148
Baseline Spectrograms	69%	80
Transfer Learning VGG16 (Trained on ImageNet)	73%	37
Baseline Spectrograms Augmented	75%	79
Transfer Learning (Trained on FSD50K)	77%	97
Transfer Learning (Trained on UrbanSound8K)	77%	59
State of the Art (BEATs)	89%	60

Future Work

- Experiment more with CNN architectures and different configurations on training.
- Experiment more with configurations on feature extraction (mel-spectrograms and audio features).
- Transfer learning from a CNN model pretrained on mel-spectrograms.
- Get a pretrained model, transfer learning on FSD50K or UrbanSound8K, and then transfer learning on FSC22.
- Get a pretrained model, transfer learning on FSD50K or UrbanSound8K, and then transfer learning on augmented FSC22.
- Experiment more with audio features in all these cases.