Dependency and Data Collection

In [10]:

```
# !pip freeze
```

In [2]:

```
import sys
print(sys.executable)
```

```
/Users/arpitpatel/Documents/dl/audio_classification/env1/bin/python
```

In [3]:

```
!pip show librosa
```

```
Name: librosa
Version: 0.10.2.post1
Summary: Python module for audio and music processing
Home-page: https://librosa.org
Author: Brian McFee, librosa development team
Author-email: brian.mcfee@nyu.edu
License: ISC
Location: /Users/arpitpatel/Documents/dl/audio_classification/env1/lib/python3.10/site-pa
ckages
Requires: audioread, decorator, joblib, lazy-loader, msgpack, numba, numpy, pooch, scikit
-learn, scipy, soundfile, soxr, typing-extensions
Required-by:
```

In [4]:

```
!pip install librosa
```

```
Requirement already satisfied: librosa in ./env1/lib/python3.10/site-packages (0.10.2.pos
t1)
Requirement already satisfied: audioread>=2.1.9 in ./env1/lib/python3.10/site-packages (f
rom librosa) (3.0.1)
Requirement already satisfied: numpy!=1.22.0,!=1.22.1,!=1.22.2,>=1.20.3 in ./env1/lib/pyt
hon3.10/site-packages (from librosa) (2.0.2)
Requirement already satisfied: scipy>=1.2.0 in ./env1/lib/python3.10/site-packages (from
librosa) (1.15.1)
Requirement already satisfied: scikit-learn>=0.20.0 in ./env1/lib/python3.10/site-package
s (from librosa) (1.6.1)
Requirement already satisfied: joblib>=0.14 in ./env1/lib/python3.10/site-packages (from
librosa) (1.4.2)
Requirement already satisfied: decorator>=4.3.0 in ./env1/lib/python3.10/site-packages (f
rom librosa) (5.1.1)
Requirement already satisfied: numba>=0.51.0 in ./env1/lib/python3.10/site-packages (from
librosa) (0.61.0)
Requirement already satisfied: soundfile>=0.12.1 in ./env1/lib/python3.10/site-packages
(from librosa) (0.13.1)
Requirement already satisfied: pooch>=1.1 in ./env1/lib/python3.10/site-packages (from li
brosa) (1.8.2)
Requirement already satisfied: soxr>=0.3.2 in ./env1/lib/python3.10/site-packages (from l
ibrosa) (0.5.0.post1)
Requirement already satisfied: typing-extensions>=4.1.1 in ./env1/lib/python3.10/site-pac
kages (from librosa) (4.12.2)
Requirement already satisfied: lazy-loader>=0.1 in ./env1/lib/python3.10/site-packages (f
rom librosa) (0.4)
Requirement already satisfied: msgpack>=1.0 in ./env1/lib/python3.10/site-packages (from
librosa) (1.1.0)
Requirement already satisfied: packaging in ./env1/lib/python3.10/site-packages (from laz
y-loader>=0.1->librosa) (24.2)
Requirement already satisfied: llvmlite<0.45,>=0.44.0dev0 in ./env1/lib/python3.10/site-p
ackages (from numba>=0.51.0->librosa) (0.44.0)
Requirement already satisfied: platformdirs>=2.5.0 in ./env1/lib/python3.10/site-packages
(from pooch>=1.1->librosa) (4.3.6)
Requirement already satisfied: requests>=2.19.0 in ./env1/lib/python3.10/site-packages (f
rom pooch>=1.1->librosa) (2.32.3)
Requirement already satisfied: threadpoolctl>=3.1.0 in ./env1/lib/python3.10/site-package
s (from scikit-learn>=0.20.0->librosa) (3.5.0)
Requirement already satisfied: cffi>=1.0 in ./env1/lib/python3.10/site-packages (from sou
ndfile>=0.12.1->librosa) (1.17.1)
Requirement already satisfied: pycparser in ./env1/lib/python3.10/site-packages (from cff
i>=1.0->soundfile>=0.12.1->librosa) (2.22)
Requirement already satisfied: charset-normalizer<4,>=2 in ./env1/lib/python3.10/site-pac
kages (from requests>=2.19.0->pooch>=1.1->librosa) (3.4.1)
Requirement already satisfied: idna<4,>=2.5 in ./env1/lib/python3.10/site-packages (from
requests>=2.19.0->pooch>=1.1->librosa) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in ./env1/lib/python3.10/site-packages
(from requests>=2.19.0->pooch>=1.1->librosa) (2.3.0)
Requirement already satisfied: certifi>=2017.4.17 in ./env1/lib/python3.10/site-packages
(from requests>=2.19.0->pooch>=1.1->librosa) (2025.1.31)
```

In [ ]:

```

```

In [5]:

```
import librosa
import librosa.display
import IPython.display as ipd
import matplotlib.pyplot as plt
import numpy as np
%matplotlib inline
```

In [6]:

```
!pwd
```

```
/Users/arpitpatel/documents/dl/audio_classification
```

In [7]:

```
filename="UrbanSound8K/audio/fold5/100263-2-0-143.wav"
```

In [8]:

```
ipd.Audio(filename)
```

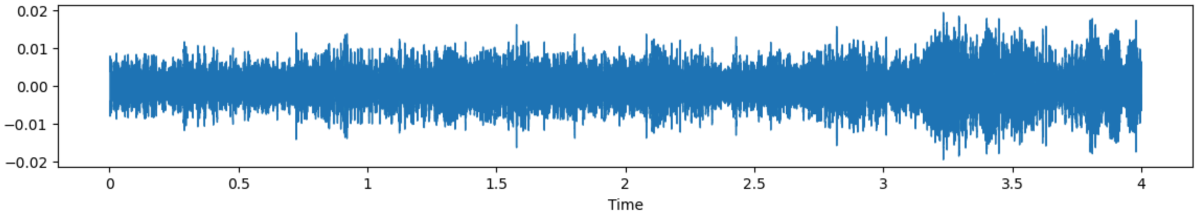
Out[8]:

`librosa.load(filename)` :

- Loads an audio file into a NumPy array (data).
- Returns:
  - **data** → Audio samples as a NumPy array (normalized between -1 and 1).
  - **sample_rate** → Sampling rate (default is 22050 Hz, unless specified).

In [9]:

```
plt.figure(figsize=(14,2))
librosa_audio_data,librosa_audio_sample_rate=librosa.load(filename)
librosa.display.waveshow(librosa_audio_data,sr=librosa_audio_sample_rate)
plt.show()
```



In [11]:

```
librosa_audio_data
```

Out[11]:

```
array([0.00426048, 0.00792833, 0.00714055, ..., 0.0058561 , 0.00561863,
       0.00630891], dtype=float32)
```

In [12]:

```
librosa_audio_sample_rate
```

Out[12]:

```
22050
```

In [13]:

```
import pandas as pd
df=pd.read_csv("UrbanSound8K/metadata/UrbanSound8K.csv")
df.head()
```

Out[13]:

	slice_file_name	fsID	start	end	salience	fold	classID	class
0	100032-3-0-0.wav	100032	0.0	0.317551	1	5	3	dog_bark
1	100263-2-0-117.wav	100263	58.5	62.500000	1	5	2	children_playing
2	100263-2-0-121.wav	100263	60.5	64.500000	1	5	2	children_playing
3	100263-2-0-126.wav	100263	63.0	67.000000	1	5	2	children_playing
4	100263-2-0-137.wav	100263	68.5	72.500000	1	5	2	children_playing

In [14]:

```
df['class'].value_counts()
```

Out[14]:

```
class
dog_bark            1000
children_playing    1000
air_conditioner     1000
street_music        1000
engine_idling       1000
jackhammer          1000
drilling            1000
siren                929
car_horn             429
gun_shot             374
Name: count, dtype: int64
```

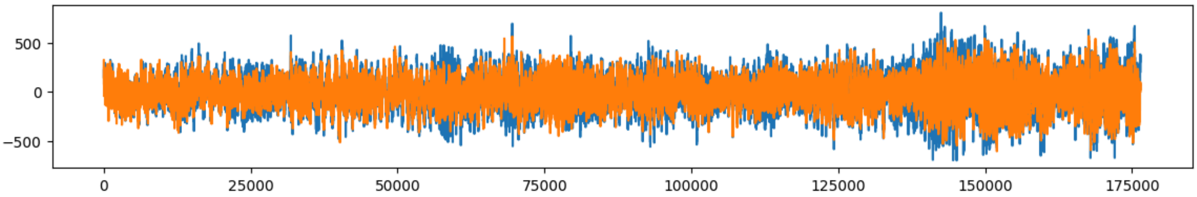
In [15]:
```
# scipy doesn't normalize data, stereo-channel
from scipy.io import wavfile
wave_sample_rate,wave=wavfile.read(filename)
wave_sample_rate
```

Out[15]:
```
44100
```

In [16]:
```
wave
```

Out[16]:
```
array([[219, 143],
       [215, 220],
       [204, 288],
       ...,
       [337,  41],
       [320,  71],
       [311,  92]], dtype=int16)
```

In [17]:
```
plt.figure(figsize=(14,2))
plt.plot(wave)
plt.show()
```



When to Use Which?

- Use `librosa.load()` when working with **speech/audio analysis, machine learning, deep learning, or music feature extraction**.
- Use `scipy.io.wavfile.read()` when you need **raw, unprocessed audio** with original integer values, stereo/multichannel support, or if you're working with **low-level signal processing**.

Feature	`librosa.load()`	`scipy.io.wavfile.read()`
Return Values	`(data, sample_rate)`	`(sample_rate, data)`
Data Type	`float32` (values between -1 and 1)	`int16` or `int32` (raw format)
Default Sample Rate	**22050 Hz** (unless specified)	Uses **original file sample rate**
Resampling	**Yes** (if `sr` is specified)	**No** (keeps original)
Mono vs Stereo	Converts to **mono** by default	Keeps **stereo/multichannel**
Supported Formats	WAV, MP3, FLAC, OGG, etc.	**Only WAV**
Use Cases	Feature extraction, ML, DL, speech/music analysis	Raw audio processing, stereo/multichannel handling

Data preprocessing

- **Mel-Frequency Cepstral Coefficients(MFCC)** : The MFCC summarises the frequency distribution across the window size, so it is possible to analyse both the frequency and time characteristics of the sound. These audio representations will allow us to identify features for classification.

In [18]:
```
mfcc_data=librosa.feature.mfcc(y=librosa_audio_data,sr=librosa_audio_sample_rate,n_mfcc=4
mfcc_data.shape
```

Out[18]:
```
(40, 173)
```

In [19]:
```
mfcc_data
```

```
Out[19]:  array([[-4.7153787e+02, -4.4912976e+02, -4.4744662e+02, ...,
                  -4.4174380e+02, -4.3948465e+02, -4.4826764e+02],
                 [ 1.0039217e+02,  9.8589615e+01,  9.5115524e+01, ...,
                   1.1791698e+02,  1.1520094e+02,  1.0672363e+02],
                 [-2.1784363e+01, -2.0289974e+01, -2.0491684e+01, ...,
                  -2.7890923e+01, -1.9016792e+01, -5.2057099e+00],
                 ...,
                 [-7.6563931e-01,  3.7504408e-01,  1.2162297e+00, ...,
                   1.0650102e+00, -8.0676794e-02,  1.8818033e+00],
                 [ 2.7869036e+00,  1.5277616e+00,  2.1518292e+00, ...,
                  -1.1723332e+01, -1.1164776e+01, -6.5244632e+00],
                 [ 3.9944468e+00, -3.0450654e+00, -6.5705299e+00, ...,
                  -6.1732082e+00, -4.8374453e+00, -1.0105597e-02]], dtype=float32)
```

```
In [45]: def feature_extractor(file):
             data,sample_rate=librosa.load(file,res_type='kaiser_fast')
             mfcc_features=librosa.feature.mfcc(y=data,sr=sample_rate,n_mfcc=30)
             mfcc_features_scaled=np.mean(mfcc_features.T,axis=0)
             return mfcc_features_scaled
```

```
In [48]: !pip install tqdm
```

```
Collecting tqdm
  Using cached tqdm-4.67.1-py3-none-any.whl.metadata (57 kB)
Using cached tqdm-4.67.1-py3-none-any.whl (78 kB)
Installing collected packages: tqdm
Successfully installed tqdm-4.67.1
```

```
In [49]: from tqdm import tqdm   # used for progress bars
         import os

         audio_dataset_path='UrbanSound8K/audio/'
         extracted_features=[]
         for index_num,row in tqdm(df.iterrows()):
             filename=os.path.join(os.path.abspath(audio_dataset_path),'fold'+str(row['fold'])+'/'
             file_class_labels=row['class']
             data=list(feature_extractor(filename))
             extracted_features.append([data,file_class_labels])
```

```
/Users/arpitpatel/Documents/dl/audio_classification/env1/lib/python3.10/site-packages/lib
rosa/core/spectrum.py:266: UserWarning: n_fft=2048 is too large for input signal of lengt
h=1323
  warnings.warn(
/Users/arpitpatel/Documents/dl/audio_classification/env1/lib/python3.10/site-packages/lib
rosa/core/spectrum.py:266: UserWarning: n_fft=2048 is too large for input signal of lengt
h=1103
  warnings.warn(
/Users/arpitpatel/Documents/dl/audio_classification/env1/lib/python3.10/site-packages/lib
rosa/core/spectrum.py:266: UserWarning: n_fft=2048 is too large for input signal of lengt
h=1523
  warnings.warn(

732it [06:38, 21.91it/s]
```

```
In [50]: len(extracted_features)
```

```
Out[50]:  8732
```

```
In [51]: extracted_features
```

```
  np.float32(14.228637),
  np.float32(6.1115594),
  np.float32(14.896896),
  np.float32(1.4612195),
  np.float32(2.2964754),
  np.float32(-1.0694754),
  np.float32(0.9927515),
  np.float32(-3.2890992),
  np.float32(5.6211214),
  np.float32(-0.96039313),
  np.float32(4.8448606),
  np.float32(2.3618069),
  np.float32(7.5745354)],
 'dog_bark'],
[[np.float32(-264.8894),
  np.float32(65.682304),
  np.float32(-21.607237),
  np.float32(-24.419447),
  np.float32(-58.740257),
  np.float32(55.399963),
  np.float32(18.664648),
  np.float32(43.128883),
  np.float32(-7.5318074),
  np.float32(8.568053),
  np.float32(-12.943659),
  np.float32(24.42157),
  np.float32(5.400291),
  np.float32(12.968349),
  np.float32(-13.487241),
  np.float32(10.05713),
  np.float32(-8.033789),
  np.float32(16.260433),
  np.float32(5.783643),
  np.float32(14.884938),
  np.float32(0.35865042),
  np.float32(1.8251077),
  np.float32(-3.3328238),
  np.float32(0.6744989),
  np.float32(-3.7196715),
  np.float32(6.0922318),
  np.float32(-1.1218809),
  np.float32(7.684655),
  np.float32(3.1977363),
  np.float32(10.047348)],
 'dog_bark'],
...]
```

In [55]:

```
dataframe = pd.DataFrame(extracted_features, columns=['feature', 'class'])
dataframe.head()
```

Out[55]:

	feature	class
0	[-217.35526, 70.22339, -130.38527, -53.282898,...	dog_bark
1	[-424.09818, 109.34076, -52.919525, 60.86475, ...	children_playing
2	[-458.79114, 121.3842, -46.520653, 52.00812, -...	children_playing
3	[-413.89984, 101.66371, -35.42945, 53.036358, ...	children_playing
4	[-446.60352, 113.68541, -52.402218, 60.302044,...	children_playing

In [56]:

```
# dataframe.to_csv('extracted_features.csv', index=False)
```

In [57]:

```
# dataframe=pd.read_csv('extracted_features.csv')
# dataframe.head()
```

In [58]:

```
X=np.array(dataframe['feature'].tolist())
y=np.array(dataframe['class'].tolist())
```

```
In [59]: X.shape

Out[59]: (8732, 30)

In [60]: y.shape

Out[60]: (8732,)

In [61]: y=pd.get_dummies(y)

In [62]: y.columns

Out[62]: Index(['air_conditioner', 'car_horn', 'children_playing', 'dog_bark',
                'drilling', 'engine_idling', 'gun_shot', 'jackhammer', 'siren',
                'street_music'],
               dtype='object')

In [63]: y=np.array(y)

In [64]: y.shape

Out[64]: (8732, 10)

In [65]: y

Out[65]: array([[False, False, False, ..., False, False, False],
                [False, False,  True, ..., False, False, False],
                [False, False,  True, ..., False, False, False],
                ...,
                [False,  True, False, ..., False, False, False],
                [False,  True, False, ..., False, False, False],
                [False,  True, False, ..., False, False, False]])

In [66]: y=y.astype(int)

In [67]: y

Out[67]: array([[0, 0, 0, ..., 0, 0, 0],
                [0, 0, 1, ..., 0, 0, 0],
                [0, 0, 1, ..., 0, 0, 0],
                ...,
                [0, 1, 0, ..., 0, 0, 0],
                [0, 1, 0, ..., 0, 0, 0],
                [0, 1, 0, ..., 0, 0, 0]])

In [68]: from sklearn.model_selection import train_test_split

         X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2,random_state=0)
         X_train.shape,y_train.shape

Out[68]: ((6985, 30), (6985, 10))

In [69]: print(X_train.dtype, y_train.dtype)  # Check data types

        float32 int64

In [70]: X_train
```

```
Out[70]: array([[-1.3110471e+02,  1.1250591e+02, -2.2574696e+01, ...,
                 -1.0572993e+00, -2.0611889e+00,  3.5242116e+00],
                [-1.3670342e+01,  9.1085083e+01, -7.7927332e+00, ...,
                 -1.9672565e+00, -9.3592281e+00, -6.0962262e+00],
                [-4.9871544e+01,  2.6535299e-01, -2.0500937e+01, ...,
                  1.3284229e+00, -7.9714413e+00,  7.0145518e-01],
                ...,
                [-4.2701236e+02,  9.2623047e+01,  3.1293974e+00, ...,
                 -1.7838219e+00, -1.5918899e+00, -1.7047207e+00],
                [-1.4575461e+02,  1.3626578e+02, -3.3515522e+01, ...,
                  4.1547985e+00, -2.2235024e+00,  3.5612748e+00],
                [-4.2103134e+02,  2.1065454e+02,  3.4906609e+00, ...,
                 -4.6783547e+00, -4.6562533e+00, -6.1745815e+00]], dtype=float32)
```

```
In [71]: y_train
```

```
Out[71]: array([[0, 0, 0, ..., 1, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                ...,
                [0, 0, 0, ..., 0, 0, 0],
                [1, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 1, 0]])
```

Model Creation

```
In [72]: import tensorflow as tf
         print(tf.__version__)
```

```
2.18.0
```

```
In [73]: tf.__path__
```

```
Out[73]: ['/Users/arpitpatel/Documents/dl/audio_classification/env1/lib/python3.10/site-packages/
         keras/api/_v2',
          '/Users/arpitpatel/Documents/dl/audio_classification/env1/lib/python3.10/site-packages/
         keras/_tf_keras',
          '/Users/arpitpatel/Documents/dl/audio_classification/env1/lib/python3.10/site-packages/
         tensorflow',
          '/Users/arpitpatel/Documents/dl/audio_classification/env1/lib/python3.10/site-packages/
         tensorflow/_api/v2']
```

```
In [74]: from tensorflow.keras.models import Sequential
         from tensorflow.keras.layers import Dense, Activation, Dropout, Flatten
         from tensorflow.keras.optimizers import Adam
         from sklearn import metrics
```

```
In [75]: no_of_classes=y.shape[1]
         no_of_classes
```

```
Out[75]: 10
```

```
In [76]: model=Sequential()
         model.add(Dense(100,input_shape=(30,)))
         model.add(Activation('relu'))
         model.add(Dropout(0.5))
         model.add(Dense(200))
         model.add(Activation('relu'))
         model.add(Dropout(0.5))
         model.add(Dense(100))
         model.add(Activation('relu'))
         model.add(Dropout(0.5))
         model.add(Dense(no_of_classes))
         model.add(Activation('softmax'))
         model.summary()
```

```
/Users/arpitpatel/Documents/dl/audio_classification/env1/lib/python3.10/site-packages/ker
as/src/layers/core/dense.py:87: UserWarning: Do not pass an `input_shape`/`input_dim` arg
ument to a layer. When using Sequential models, prefer using an `Input(shape)` object as
the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
Model: "sequential_2"
```

```
┏━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┳━━━━━━━━━━━━━━━━━━━━━━━━┳━━━━━━━━━━━━━━━┓
┃ Layer (type)                      ┃ Output Shape           ┃       Param # ┃
┡━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━╇━━━━━━━━━━━━━━━━━━━━━━━━╇━━━━━━━━━━━━━━━┩
│ dense_5 (Dense)                   │ (None, 100)            │         3,100 │
├───────────────────────────────────┼────────────────────────┼───────────────┤
│ activation_5 (Activation)         │ (None, 100)            │             0 │
├───────────────────────────────────┼────────────────────────┼───────────────┤
│ dropout_3 (Dropout)               │ (None, 100)            │             0 │
├───────────────────────────────────┼────────────────────────┼───────────────┤
│ dense_6 (Dense)                   │ (None, 200)            │        20,200 │
├───────────────────────────────────┼────────────────────────┼───────────────┤
│ activation_6 (Activation)         │ (None, 200)            │             0 │
├───────────────────────────────────┼────────────────────────┼───────────────┤
│ dropout_4 (Dropout)               │ (None, 200)            │             0 │
├───────────────────────────────────┼────────────────────────┼───────────────┤
│ dense_7 (Dense)                   │ (None, 100)            │        20,100 │
├───────────────────────────────────┼────────────────────────┼───────────────┤
│ activation_7 (Activation)         │ (None, 100)            │             0 │
├───────────────────────────────────┼────────────────────────┼───────────────┤
│ dropout_5 (Dropout)               │ (None, 100)            │             0 │
├───────────────────────────────────┼────────────────────────┼───────────────┤
│ dense_8 (Dense)                   │ (None, 10)             │         1,010 │
├───────────────────────────────────┼────────────────────────┼───────────────┤
│ activation_8 (Activation)         │ (None, 10)             │             0 │
└───────────────────────────────────┴────────────────────────┴───────────────┘
 Total params: 44,410 (173.48 KB)
 Trainable params: 44,410 (173.48 KB)
 Non-trainable params: 0 (0.00 B)
```

In [77]:

```python
model.compile(loss='categorical_crossentropy',
              metrics=['accuracy'],
              optimizer='adam')
```

In [78]:

```python
# model training
from tensorflow.keras.callbacks import ModelCheckpoint
from datetime import datetime

num_epochs = 100
num_batch_size = 32

checkpointer = ModelCheckpoint(filepath='saved_models/audio_classification.keras',
                               verbose=1, save_best_only=True)
start = datetime.now()

model.fit(X_train,
          y_train,
          batch_size=num_batch_size,
          epochs=num_epochs,
          validation_data=(X_test, y_test),
          callbacks=[checkpointer],
          verbose=1)


duration = datetime.now() - start
print("Training completed in time: ", duration)
```

```
Epoch 1/100

Epoch 1: val_loss improved from inf to 2.27894, saving model to saved_models/audio_classi
fication.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.1168 - loss: 22.1450 - val_accurac
y: 0.1162 - val_loss: 2.2789
Epoch 2/100

Epoch 2: val_loss improved from 2.27894 to 2.26441, saving model to saved_models/audio_cl
assification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.1454 - loss: 2.5664 - val_accurac
y: 0.1265 - val_loss: 2.2644
Epoch 3/100

Epoch 3: val_loss improved from 2.26441 to 2.24483, saving model to saved_models/audio_cl
assification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.1490 - loss: 2.3376 - val_accurac
y: 0.1328 - val_loss: 2.2448
Epoch 4/100

Epoch 4: val_loss improved from 2.24483 to 2.19504, saving model to saved_models/audio_cl
assification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.1448 - loss: 2.2705 - val_accurac
y: 0.1643 - val_loss: 2.1950
Epoch 5/100

Epoch 5: val_loss improved from 2.19504 to 2.16022, saving model to saved_models/audio_cl
assification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.1584 - loss: 2.2326 - val_accurac
y: 0.1671 - val_loss: 2.1602
Epoch 6/100

Epoch 6: val_loss improved from 2.16022 to 2.11677, saving model to saved_models/audio_cl
assification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.1609 - loss: 2.2092 - val_accurac
y: 0.1895 - val_loss: 2.1168
Epoch 7/100

Epoch 7: val_loss improved from 2.11677 to 2.11282, saving model to saved_models/audio_cl
assification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.1641 - loss: 2.1868 - val_accurac
y: 0.1946 - val_loss: 2.1128
Epoch 8/100

Epoch 8: val_loss improved from 2.11282 to 2.06041, saving model to saved_models/audio_cl
assification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.1898 - loss: 2.1512 - val_accurac
y: 0.2112 - val_loss: 2.0604
Epoch 9/100

Epoch 9: val_loss improved from 2.06041 to 2.03099, saving model to saved_models/audio_cl
assification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.1993 - loss: 2.1290 - val_accurac
y: 0.2238 - val_loss: 2.0310
Epoch 10/100

Epoch 10: val_loss improved from 2.03099 to 2.02967, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.2035 - loss: 2.1033 - val_accurac
y: 0.2124 - val_loss: 2.0297
Epoch 11/100

Epoch 11: val_loss improved from 2.02967 to 2.00268, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.2137 - loss: 2.0938 - val_accurac
y: 0.2444 - val_loss: 2.0027
Epoch 12/100

Epoch 12: val_loss improved from 2.00268 to 1.96240, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.2245 - loss: 2.0662 - val_accurac
```

```
y: 0.2610 - val_loss: 1.9624
Epoch 13/100

Epoch 13: val_loss did not improve from 1.96240
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.2341 - loss: 2.0488 - val_accurac
y: 0.2610 - val_loss: 1.9690
Epoch 14/100

Epoch 14: val_loss improved from 1.96240 to 1.92774, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.2399 - loss: 2.0377 - val_accurac
y: 0.2816 - val_loss: 1.9277
Epoch 15/100

Epoch 15: val_loss improved from 1.92774 to 1.91477, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.2439 - loss: 2.0145 - val_accurac
y: 0.2713 - val_loss: 1.9148
Epoch 16/100

Epoch 16: val_loss improved from 1.91477 to 1.90720, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.2742 - loss: 1.9757 - val_accurac
y: 0.2999 - val_loss: 1.9072
Epoch 17/100

Epoch 17: val_loss improved from 1.90720 to 1.88826, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.2542 - loss: 1.9940 - val_accurac
y: 0.3085 - val_loss: 1.8883
Epoch 18/100

Epoch 18: val_loss improved from 1.88826 to 1.83863, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.2730 - loss: 1.9681 - val_accurac
y: 0.3251 - val_loss: 1.8386
Epoch 19/100

Epoch 19: val_loss improved from 1.83863 to 1.81907, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.2930 - loss: 1.9201 - val_accurac
y: 0.3194 - val_loss: 1.8191
Epoch 20/100

Epoch 20: val_loss improved from 1.81907 to 1.78003, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.2918 - loss: 1.9097 - val_accurac
y: 0.3543 - val_loss: 1.7800
Epoch 21/100

Epoch 21: val_loss improved from 1.78003 to 1.76133, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.3190 - loss: 1.8645 - val_accurac
y: 0.3623 - val_loss: 1.7613
Epoch 22/100

Epoch 22: val_loss improved from 1.76133 to 1.75052, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.3289 - loss: 1.8318 - val_accurac
y: 0.3726 - val_loss: 1.7505
Epoch 23/100

Epoch 23: val_loss improved from 1.75052 to 1.67383, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.3408 - loss: 1.7919 - val_accurac
y: 0.4047 - val_loss: 1.6738
Epoch 24/100

Epoch 24: val_loss improved from 1.67383 to 1.62895, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.3533 - loss: 1.7628 - val_accurac
```

```
y: 0.4253 - val_loss: 1.6289
Epoch 25/100

Epoch 25: val_loss improved from 1.62895 to 1.62070, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.3607 - loss: 1.7634 - val_accurac
y: 0.4236 - val_loss: 1.6207
Epoch 26/100

Epoch 26: val_loss improved from 1.62070 to 1.58497, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.3793 - loss: 1.7110 - val_accurac
y: 0.4293 - val_loss: 1.5850
Epoch 27/100

Epoch 27: val_loss improved from 1.58497 to 1.56545, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.3824 - loss: 1.7150 - val_accurac
y: 0.4482 - val_loss: 1.5655
Epoch 28/100

Epoch 28: val_loss improved from 1.56545 to 1.53407, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.4004 - loss: 1.6499 - val_accurac
y: 0.4448 - val_loss: 1.5341
Epoch 29/100

Epoch 29: val_loss improved from 1.53407 to 1.52533, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.3964 - loss: 1.6726 - val_accurac
y: 0.4539 - val_loss: 1.5253
Epoch 30/100

Epoch 30: val_loss improved from 1.52533 to 1.52177, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.4067 - loss: 1.6454 - val_accurac
y: 0.4539 - val_loss: 1.5218
Epoch 31/100

Epoch 31: val_loss improved from 1.52177 to 1.49991, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.3901 - loss: 1.6727 - val_accurac
y: 0.4574 - val_loss: 1.4999
Epoch 32/100

Epoch 32: val_loss improved from 1.49991 to 1.47650, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.4242 - loss: 1.6041 - val_accurac
y: 0.4740 - val_loss: 1.4765
Epoch 33/100

Epoch 33: val_loss improved from 1.47650 to 1.44389, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.4238 - loss: 1.5877 - val_accurac
y: 0.5077 - val_loss: 1.4439
Epoch 34/100

Epoch 34: val_loss improved from 1.44389 to 1.41058, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.4380 - loss: 1.5843 - val_accurac
y: 0.5312 - val_loss: 1.4106
Epoch 35/100

Epoch 35: val_loss improved from 1.41058 to 1.32957, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.4665 - loss: 1.5272 - val_accurac
y: 0.5638 - val_loss: 1.3296
Epoch 36/100

Epoch 36: val_loss did not improve from 1.32957
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.4655 - loss: 1.5114 - val_accurac
```

```
y: 0.5547 - val_loss: 1.3351
Epoch 37/100

Epoch 37: val_loss improved from 1.32957 to 1.28407, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.4722 - loss: 1.5226 - val_accurac
y: 0.5690 - val_loss: 1.2841
Epoch 38/100

Epoch 38: val_loss improved from 1.28407 to 1.27228, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.4757 - loss: 1.4879 - val_accurac
y: 0.5804 - val_loss: 1.2723
Epoch 39/100

Epoch 39: val_loss improved from 1.27228 to 1.25518, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5207 - loss: 1.4128 - val_accurac
y: 0.5793 - val_loss: 1.2552
Epoch 40/100

Epoch 40: val_loss improved from 1.25518 to 1.22691, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.4843 - loss: 1.4714 - val_accurac
y: 0.5942 - val_loss: 1.2269
Epoch 41/100

Epoch 41: val_loss improved from 1.22691 to 1.21897, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5091 - loss: 1.4455 - val_accurac
y: 0.6073 - val_loss: 1.2190
Epoch 42/100

Epoch 42: val_loss did not improve from 1.21897
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5050 - loss: 1.4123 - val_accurac
y: 0.6033 - val_loss: 1.2237
Epoch 43/100

Epoch 43: val_loss improved from 1.21897 to 1.15891, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5149 - loss: 1.4039 - val_accurac
y: 0.6251 - val_loss: 1.1589
Epoch 44/100

Epoch 44: val_loss did not improve from 1.15891
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5225 - loss: 1.3756 - val_accurac
y: 0.6073 - val_loss: 1.1762
Epoch 45/100

Epoch 45: val_loss improved from 1.15891 to 1.14098, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5269 - loss: 1.3756 - val_accurac
y: 0.6251 - val_loss: 1.1410
Epoch 46/100

Epoch 46: val_loss did not improve from 1.14098
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5388 - loss: 1.3494 - val_accurac
y: 0.6348 - val_loss: 1.1454
Epoch 47/100

Epoch 47: val_loss did not improve from 1.14098
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5334 - loss: 1.3693 - val_accurac
y: 0.6113 - val_loss: 1.1513
Epoch 48/100

Epoch 48: val_loss improved from 1.14098 to 1.12701, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5484 - loss: 1.3486 - val_accurac
y: 0.6325 - val_loss: 1.1270
Epoch 49/100
Epoch 49: val_loss improved from 1.12701 to 1.08793, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5507 - loss: 1.3194 - val_accurac
y: 0.6554 - val_loss: 1.0879
Epoch 50/100

Epoch 50: val_loss did not improve from 1.08793
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5465 - loss: 1.3156 - val_accurac
y: 0.6428 - val_loss: 1.1145
Epoch 51/100

Epoch 51: val_loss improved from 1.08793 to 1.08770, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5494 - loss: 1.3071 - val_accurac
y: 0.6537 - val_loss: 1.0877
Epoch 52/100

Epoch 52: val_loss did not improve from 1.08770
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5675 - loss: 1.2884 - val_accurac
y: 0.6422 - val_loss: 1.0998
Epoch 53/100

Epoch 53: val_loss improved from 1.08770 to 1.07320, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5636 - loss: 1.2666 - val_accurac
y: 0.6497 - val_loss: 1.0732
Epoch 54/100

Epoch 54: val_loss did not improve from 1.07320
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5582 - loss: 1.2994 - val_accurac
y: 0.6365 - val_loss: 1.1161
Epoch 55/100

Epoch 55: val_loss improved from 1.07320 to 1.05821, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5546 - loss: 1.2949 - val_accurac
y: 0.6651 - val_loss: 1.0582
Epoch 56/100

Epoch 56: val_loss improved from 1.05821 to 1.03267, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5789 - loss: 1.2370 - val_accurac
y: 0.6795 - val_loss: 1.0327
Epoch 57/100

Epoch 57: val_loss did not improve from 1.03267
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5768 - loss: 1.2626 - val_accurac
y: 0.6760 - val_loss: 1.0572
Epoch 58/100

Epoch 58: val_loss did not improve from 1.03267
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5724 - loss: 1.2415 - val_accurac
y: 0.6726 - val_loss: 1.0336
Epoch 59/100

Epoch 59: val_loss improved from 1.03267 to 1.02431, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5939 - loss: 1.2138 - val_accurac
y: 0.6697 - val_loss: 1.0243
Epoch 60/100

Epoch 60: val_loss improved from 1.02431 to 1.00273, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5917 - loss: 1.2063 - val_accurac
y: 0.6875 - val_loss: 1.0027
Epoch 61/100

Epoch 61: val_loss improved from 1.00273 to 0.97002, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5903 - loss: 1.1890 - val_accurac
y: 0.6898 - val_loss: 0.9700
```

```
Epoch 62/100

Epoch 62: val_loss did not improve from 0.97002
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5922 - loss: 1.2165 - val_accurac
y: 0.6829 - val_loss: 0.9813
Epoch 63/100

Epoch 63: val_loss improved from 0.97002 to 0.96846, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6032 - loss: 1.1928 - val_accurac
y: 0.6903 - val_loss: 0.9685
Epoch 64/100

Epoch 64: val_loss did not improve from 0.96846
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6050 - loss: 1.1723 - val_accurac
y: 0.6772 - val_loss: 0.9923
Epoch 65/100

Epoch 65: val_loss did not improve from 0.96846
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5973 - loss: 1.1958 - val_accurac
y: 0.6703 - val_loss: 1.0066
Epoch 66/100

Epoch 66: val_loss improved from 0.96846 to 0.96692, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5901 - loss: 1.2131 - val_accurac
y: 0.6995 - val_loss: 0.9669
Epoch 67/100

Epoch 67: val_loss improved from 0.96692 to 0.96267, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6044 - loss: 1.1614 - val_accurac
y: 0.6926 - val_loss: 0.9627
Epoch 68/100

Epoch 68: val_loss did not improve from 0.96267
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6125 - loss: 1.1585 - val_accurac
y: 0.6783 - val_loss: 0.9798
Epoch 69/100

Epoch 69: val_loss improved from 0.96267 to 0.93169, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6143 - loss: 1.1356 - val_accurac
y: 0.6926 - val_loss: 0.9317
Epoch 70/100

Epoch 70: val_loss did not improve from 0.93169
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6064 - loss: 1.1864 - val_accurac
y: 0.6955 - val_loss: 0.9426
Epoch 71/100

Epoch 71: val_loss did not improve from 0.93169
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6135 - loss: 1.1374 - val_accurac
y: 0.6972 - val_loss: 0.9563
Epoch 72/100

Epoch 72: val_loss did not improve from 0.93169
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.5979 - loss: 1.1895 - val_accurac
y: 0.6949 - val_loss: 0.9393
Epoch 73/100

Epoch 73: val_loss did not improve from 0.93169
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6233 - loss: 1.1024 - val_accurac
y: 0.7001 - val_loss: 0.9594
Epoch 74/100

Epoch 74: val_loss improved from 0.93169 to 0.92755, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6313 - loss: 1.0916 - val_accurac
y: 0.6961 - val_loss: 0.9276
Epoch 75/100
Epoch 75: val_loss did not improve from 0.92755
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6184 - loss: 1.1373 - val_accurac
y: 0.6920 - val_loss: 0.9287
Epoch 76/100

Epoch 76: val_loss improved from 0.92755 to 0.92472, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6199 - loss: 1.1038 - val_accurac
y: 0.7201 - val_loss: 0.9247
Epoch 77/100

Epoch 77: val_loss improved from 0.92472 to 0.92197, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6382 - loss: 1.0923 - val_accurac
y: 0.7041 - val_loss: 0.9220
Epoch 78/100

Epoch 78: val_loss improved from 0.92197 to 0.90033, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6120 - loss: 1.1876 - val_accurac
y: 0.7127 - val_loss: 0.9003
Epoch 79/100

Epoch 79: val_loss did not improve from 0.90033
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6290 - loss: 1.1232 - val_accurac
y: 0.7149 - val_loss: 0.9076
Epoch 80/100

Epoch 80: val_loss did not improve from 0.90033
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6188 - loss: 1.1342 - val_accurac
y: 0.7029 - val_loss: 0.9106
Epoch 81/100

Epoch 81: val_loss did not improve from 0.90033
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6418 - loss: 1.0827 - val_accurac
y: 0.6995 - val_loss: 0.9174
Epoch 82/100

Epoch 82: val_loss did not improve from 0.90033
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6345 - loss: 1.0887 - val_accurac
y: 0.7178 - val_loss: 0.9105
Epoch 83/100

Epoch 83: val_loss improved from 0.90033 to 0.88252, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6320 - loss: 1.0872 - val_accurac
y: 0.7138 - val_loss: 0.8825
Epoch 84/100

Epoch 84: val_loss did not improve from 0.88252
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6296 - loss: 1.1071 - val_accurac
y: 0.7149 - val_loss: 0.9009
Epoch 85/100

Epoch 85: val_loss did not improve from 0.88252
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6369 - loss: 1.0663 - val_accurac
y: 0.7161 - val_loss: 0.8929
Epoch 86/100

Epoch 86: val_loss improved from 0.88252 to 0.87390, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6221 - loss: 1.0994 - val_accurac
y: 0.7178 - val_loss: 0.8739
Epoch 87/100

Epoch 87: val_loss improved from 0.87390 to 0.85530, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.6398 - loss: 1.0760 - val_accurac
y: 0.7224 - val_loss: 0.8553
Epoch 88/100
Epoch 88: val_loss did not improve from 0.85530
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6386 - loss: 1.0824 - val_accurac
y: 0.7212 - val_loss: 0.8740
Epoch 89/100

Epoch 89: val_loss did not improve from 0.85530
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6385 - loss: 1.1122 - val_accurac
y: 0.7258 - val_loss: 0.8776
Epoch 90/100

Epoch 90: val_loss did not improve from 0.85530
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6460 - loss: 1.0709 - val_accurac
y: 0.7287 - val_loss: 0.8578
Epoch 91/100

Epoch 91: val_loss did not improve from 0.85530
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6420 - loss: 1.0643 - val_accurac
y: 0.7161 - val_loss: 0.8704
Epoch 92/100

Epoch 92: val_loss did not improve from 0.85530
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6440 - loss: 1.0747 - val_accurac
y: 0.7201 - val_loss: 0.8683
Epoch 93/100

Epoch 93: val_loss improved from 0.85530 to 0.83615, saving model to saved_models/audio_c
lassification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6483 - loss: 1.0612 - val_accurac
y: 0.7338 - val_loss: 0.8361
Epoch 94/100

Epoch 94: val_loss did not improve from 0.83615
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6274 - loss: 1.0960 - val_accurac
y: 0.7230 - val_loss: 0.8422
Epoch 95/100

Epoch 95: val_loss did not improve from 0.83615
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6349 - loss: 1.0802 - val_accurac
y: 0.7355 - val_loss: 0.8598
Epoch 96/100

Epoch 96: val_loss did not improve from 0.83615
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6295 - loss: 1.0864 - val_accurac
y: 0.7201 - val_loss: 0.8869
Epoch 97/100

Epoch 97: val_loss did not improve from 0.83615
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6450 - loss: 1.0604 - val_accurac
y: 0.7224 - val_loss: 0.8580
Epoch 98/100

Epoch 98: val_loss did not improve from 0.83615
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6522 - loss: 1.0449 - val_accurac
y: 0.7189 - val_loss: 0.8459
Epoch 99/100

Epoch 99: val_loss did not improve from 0.83615
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6471 - loss: 1.0394 - val_accurac
y: 0.7258 - val_loss: 0.8612
Epoch 100/100

Epoch 100: val_loss improved from 0.83615 to 0.79725, saving model to saved_models/audio_
classification.keras
219/219 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.6571 - loss: 1.0184 - val_accurac
y: 0.7624 - val_loss: 0.7972
Training completed in time:  0:00:40.056719
```

In [81]:

```
filename="UrbanSound8K/audio/fold2/100652-3-0-2.wav"
prediction_feature=feature_extractor(filename)
prediction_feature=prediction_feature.reshape(1,-1)
```

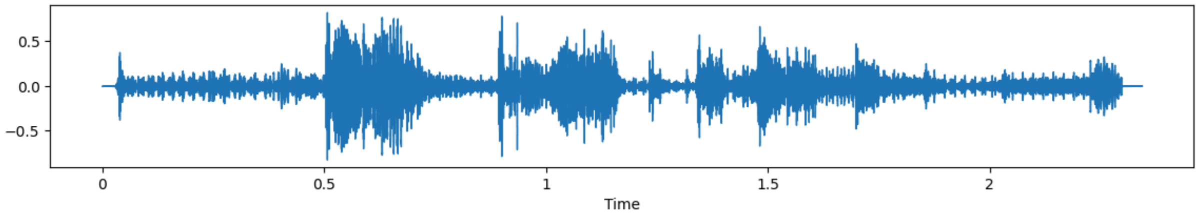
```
prediction = model.predict(prediction_feature)

predicted_class = np.argmax(prediction, axis=1)  # Get class index
print(predicted_class)
```

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 52ms/step
[3]
```

In [84]:

```
filename="chin_tapak_dum_dum.mp3"
plt.figure(figsize=(14,2))
librosa_audio_data,librosa_audio_sample_rate=librosa.load(filename)
librosa.display.waveshow(librosa_audio_data,sr=librosa_audio_sample_rate)
plt.show()
```



In [85]:

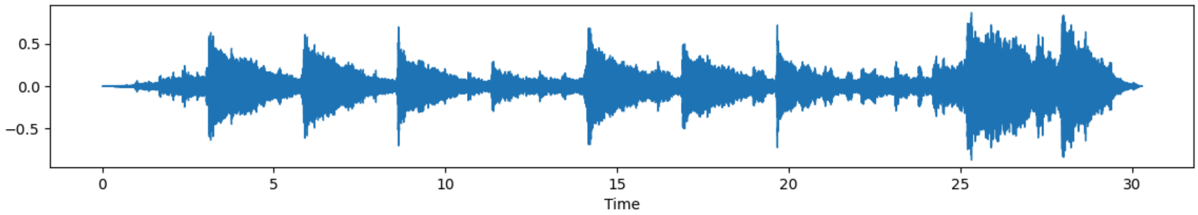
```
ipd.Audio(filename)
```

Out[85]:

▶ 0:00 / 0:02

In [86]:

```
filename="DARK theme Song Ringtone _ Dark Series Whatsapp status tamil _ Netflix series (
plt.figure(figsize=(14,2))
librosa_audio_data,librosa_audio_sample_rate=librosa.load(filename)
librosa.display.waveshow(librosa_audio_data,sr=librosa_audio_sample_rate)
plt.show()
```



In [87]:

```
ipd.Audio(filename)
```

Out[87]:

▶ 0:00 / 0:30

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: