Audio corruptions example

Load an audio from the IEMOCAP dataset

In [99]:
```
import librosa, os, shutil
from IPython.display import Audio
import soundfile as sf

iemocap_audio_file_path = "../../datasets/iemocap/Session5/sentences/wav/Ses05F_script01_1/Ses05F_script01_1_F0
shutil.copy(iemocap_audio_file_path, "iemocap_audio.wav")
iemocap_audio_file_path = "iemocap_audio.wav"
output_file_path = "tmp.wav"

audio_data, sample_rate = librosa.load(iemocap_audio_file_path, sr=None)

Audio(iemocap_audio_file_path)
```

Out[99]:

Add Gaussian SNR

In [59]:
```
from gaussian import AWGNAugmentation

if os.path.exists(output_file_path):
    os.remove(output_file_path)

config = {
    'snr': 10,
}

gaussian_10 = AWGNAugmentation(config)
corrupted_audio, corruption_type = gaussian_10.run(audio_data, sample_rate)

sf.write(output_file_path, corrupted_audio, sample_rate)
ipd.Audio(output_file_path)
```

Out[59]:

Add clipping distortion

In [66]:
```
from clipping_distortion import AddClippingDistortion

if os.path.exists(output_file_path):
    os.remove(output_file_path)

config = {
    'max_percentile_threshold': 40,
}

clipping_40 = AddClippingDistortion(config)
corrupted_audio, corruption_type = clipping_40.run(audio_data, sample_rate)

sf.write(output_file_path, corrupted_audio, sample_rate)
ipd.Audio(output_file_path)
```

Out[66]:

Compress audio

In [71]:
```
from compression import Compression

if os.path.exists(output_file_path):
    os.remove(output_file_path)

config = {
    'bit_rate': 8,
}

compression_16 = Compression(config)
compression_16.run(iemocap_audio_file_path, sample_rate, output_file_path)

ipd.Audio(output_file_path)
```

Out[71]: 

Add gain_transition

In [72]:
```
from gain_transition import AddGainTransition

if os.path.exists(output_file_path):
    os.remove(output_file_path)

config = {
    'min_max_gain_db': [-30.0, -10.0]
}

gain_transition_30_10 = AddGainTransition(config)
corrupted_audio, corruption_type = gain_transition_30_10.run(audio_data, sample_rate)

sf.write(output_file_path, corrupted_audio, sample_rate)
ipd.Audio(output_file_path)
```

```
removing
```

Out[72]: 

Add reverberation

In [78]:
```
from impulse_response import AddImpulseResponse
import warnings

warnings.filterwarnings("ignore")
if os.path.exists(output_file_path):
    os.remove(output_file_path)

config = {
    'ir_path': "../../datasets/EchoThiefImpulseResponseLibrary/Underground",
    'rt60_range': [0.1, 0.5],
}

reverberation_01_05 = AddImpulseResponse(config)
corrupted_audio, corruption_type = reverberation_01_05.run(audio_data, sample_rate)

sf.write(output_file_path, corrupted_audio, sample_rate)
ipd.Audio(output_file_path)
```

```
Selected 1 impulse responses from ../../datasets/EchoThiefImpulseResponseLibrary/Underground with RT60 in range
[0.1, 0.5]
```

Out[78]: 

Add background noise from ESC50 dataset (0dB)

In [87]:
```
from content import ContentCorruption

if os.path.exists(output_file_path):
    os.remove(output_file_path)

config = {
    'content_dataset_path': '../../datasets/ESC-50-master',
    'snr': 0
}

esc_augment_0_db = ContentCorruption(config)
corrupted_audio, corruption_type = esc_augment_0_db.run(audio_data, sample_rate)

print(f"Corruption file: {corruption_type}")

sf.write(output_file_path, corrupted_audio, sample_rate)
ipd.Audio(output_file_path)
```

```
Corruption file: 4-154793-A-4.wav
```

Out[87]: 

Add background noise from MUSAN (10dB)

In [89]:
```
from content import ContentCorruption

if os.path.exists(output_file_path):
    os.remove(output_file_path)
```

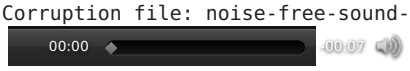
```
config = {
    'content_dataset_path': '../../datasets/musan',
    'snr': 10
}

musan_augment_10_db = ContentCorruption(config)
corrupted_audio, corruption_type = musan_augment_10_db.run(audio_data, sample_rate)

print(f"Corruption file: {corruption_type}")

sf.write(output_file_path, corrupted_audio, sample_rate)
ipd.Audio(output_file_path)
```

```
Corruption file: noise-free-sound-0629.wav
```

Out[89]: 

Add background noise from urbansound8k (20dB)

In [90]:
```
from content import ContentCorruption

if os.path.exists(output_file_path):
    os.remove(output_file_path)

config = {
    'content_dataset_path': '../../datasets/urbansound8k',
    'snr': 20
}

urban_augment_20_db = ContentCorruption(config)
corrupted_audio, corruption_type = urban_augment_20_db.run(audio_data, sample_rate)

print(f"Corruption file: {corruption_type}")

sf.write(output_file_path, corrupted_audio, sample_rate)
ipd.Audio(output_file_path)
```

```
Corruption file: 129356-2-0-118.wav
```

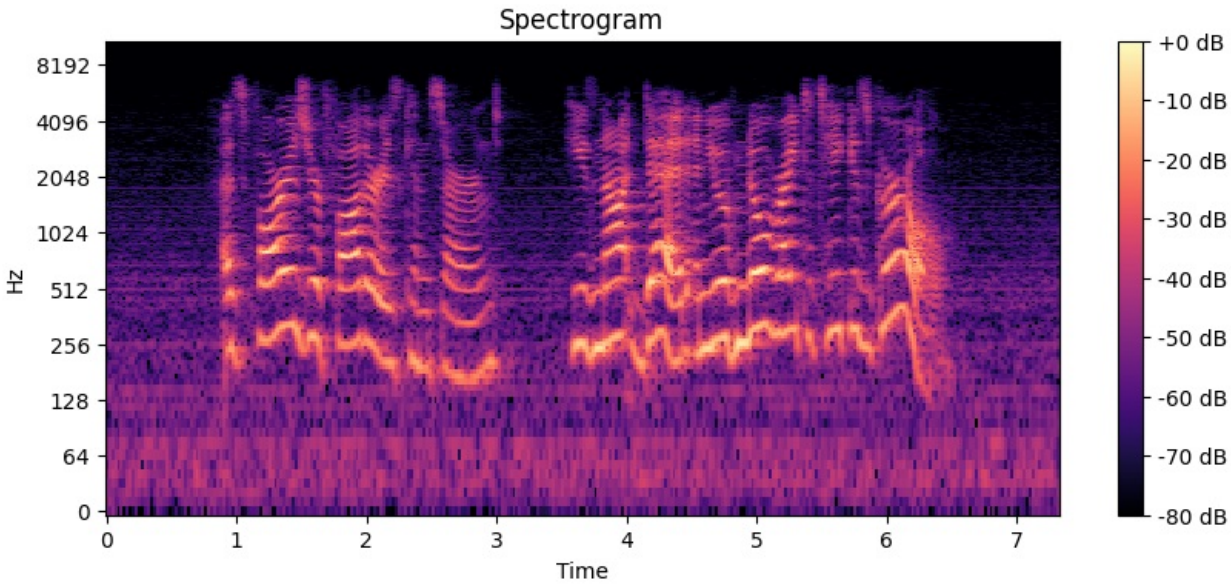
Out[90]: 

In [98]:
```
import numpy as np
import matplotlib.pyplot as plt
import librosa.display


def get_spectrogram(audio_path):
    audio_data, sample_rate = librosa.load(audio_path, sr=None)
    spectrogram = np.abs(librosa.stft(audio_data))

    # Display spectrogram
    plt.figure(figsize=(10, 4))
    librosa.display.specshow(librosa.amplitude_to_db(spectrogram, ref=np.max), sr=sample_rate, x_axis='time', y
    plt.colorbar(format='%+2.0f dB')
    plt.title('Spectrogram')
    plt.show()
```

In [97]:
```
get_spectrogram(iemocap_audio_file_path)
```



In [ ]: