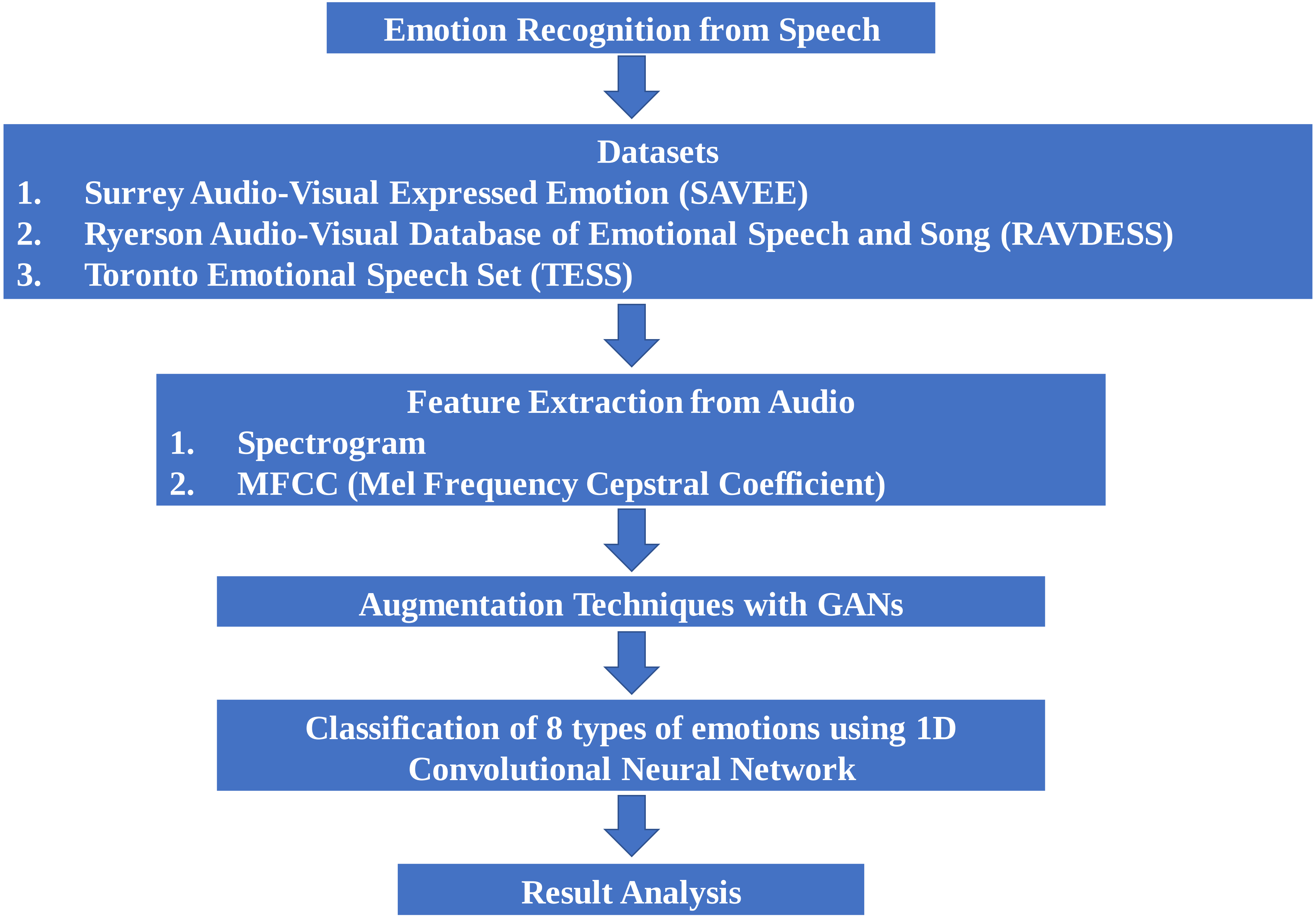
SciPy International Conference 2019

A Novel Convolutional Neural Network Architecture for Audio Emotions Classification

Assistant Prof. Mrudang Pandya
CSPIT-IT, CHARUSAT, Changa.
(NVIDIA AMBASSDOR & Leading India.AI Member)

Prince MAKWANA (STUDENT)
B.Tech 3rd year, DEPSTAR-CSE,
CHARUSAT, Changa.

RESEARCH PAPER FLOW



Emotion Recognition from Speech

Datasets

1. **Surrey Audio-Visual Expressed Emotion (SAVEE)**
2. **Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS)**
3. **Toronto Emotional Speech Set (TESS)**

Feature Extraction from Audio

1. **Spectrogram**
2. **MFCC (Mel Frequency Cepstral Coefficient)**

Augmentation Techniques with GANs

Classification of 8 types of emotions using 1D Convolutional Neural Network

Result Analysis

MOTIVATION FOR SPEECH SIGNALS PROCESSING

- CHATBOTS
- HUMAN COMPUTER INTERACTION
- BANKING
- INTERNET OF THINGS
- LANGUAGE TRANSLATION
- HEALTHCARE
- CALL CENTRES
- EMOTION CLASSIFICATION

Challenges for Speech Emotion Recognition

- Emotions are subjective, people would interpret it differently.
- Hard to define notions of emotions.
- Collecting data is complex task.
- Audio-visual data get from films and news reporting both are biased because news reporting has to be neutral and actors’ imitate the emotions.
- Difficult for machine to differentiate the imitated emotions and actual emotions of humans.
- Cost of labelling/annotating an audio is challenging task.

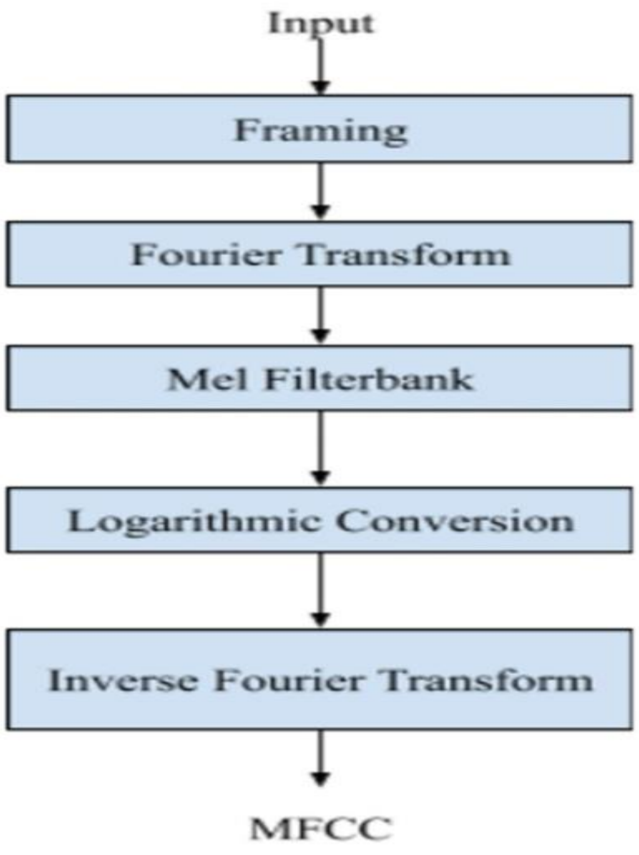
Datasets

- Surrey Audio-Visual Expressed Emotion (SAVEE)
  - 4 male actors
  - 7 emotion classes
  - 480 speech and song audio files
- Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS)
  - 24 professional actors (12 male & 12 female)
  - 8 emotion classes
  - 2422 speech and song audio files
- Toronto Emotional Speech Set (TESS)
  - 200 target words were spoken in the carrier phrase “Say the word _______”
  - 2 actors (24 years old & 80 years old)
  - 2800 audio files

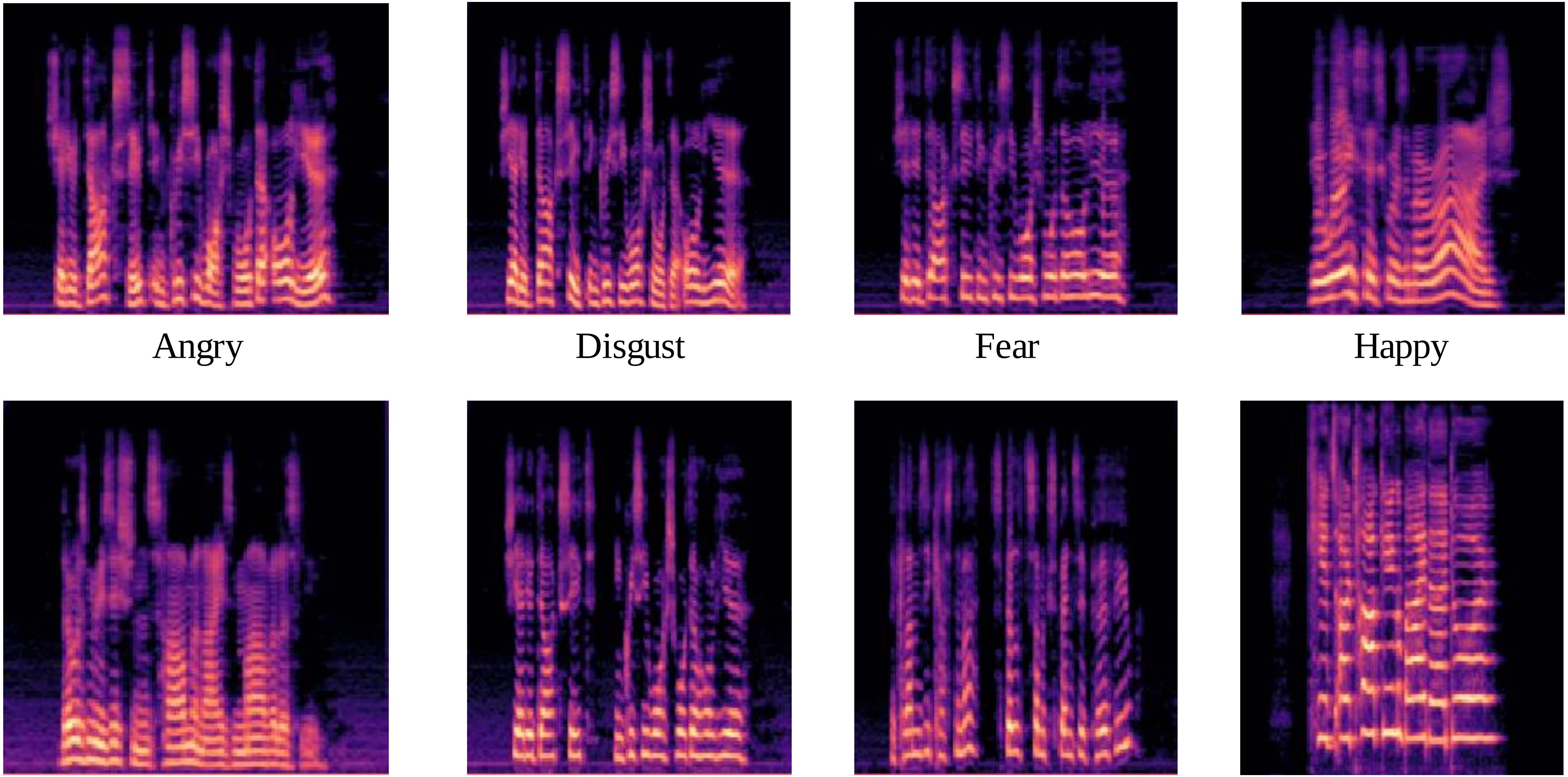
Different Approaches for extracting features from Audio Data

- Spectrograms: It is visual representation of spectrum of frequencies of a signal as it varies with time.

- MFCC (Mel – Frequency Cepstral Coefficient): The MFC coefficients are representation of short term power spectrum, based on linear cosine transform of a log power spectrum on a nonlinear Mel scale.



SPECTROGRAMS OF DIFFERENT CLASSES OF EMOTIONS



Angry

Disgust

Fear

Happy

Neutral

Sad

Surprise

Calm

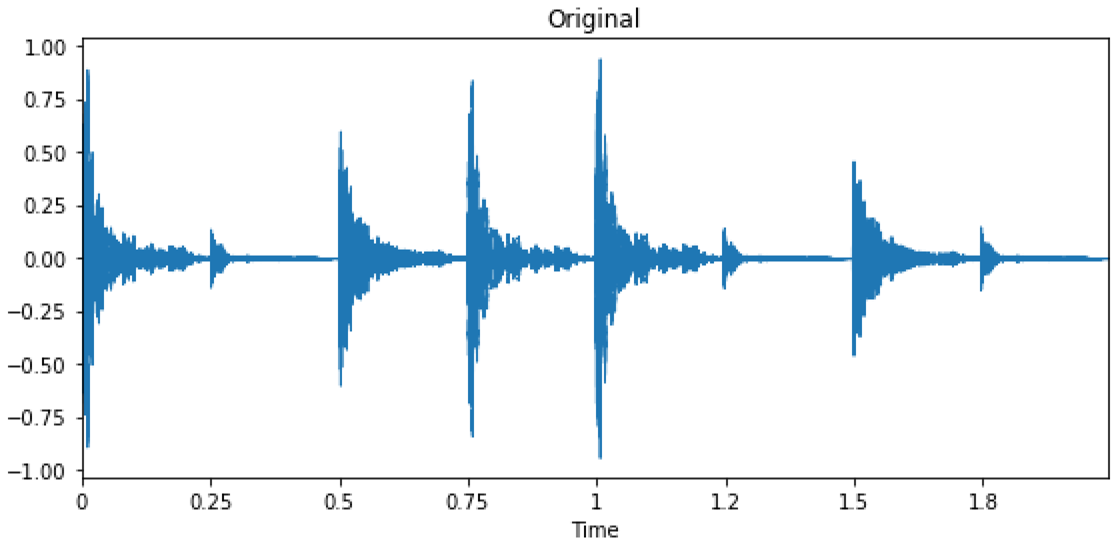
Why MFCC better than Spectrograms?

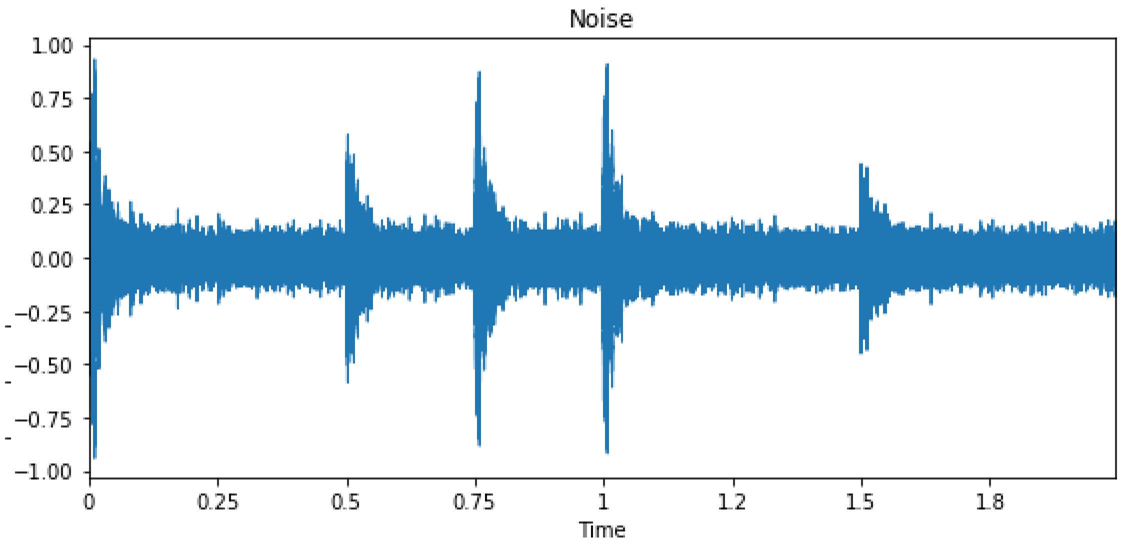
- Mel scale relates perceived frequency by listeners to its actual measured frequency.
- Generally, humans are much better at responding at small changes in pitch at low frequencies than they are at high frequencies.
- Incorporating this feature makes audio features more close to human ear.
- Thus, MFCC features are more biologically inspired.

AUGMENTATION TECHNIQUES

Noise Injection: Adding some random value (noise) into data using numpy.

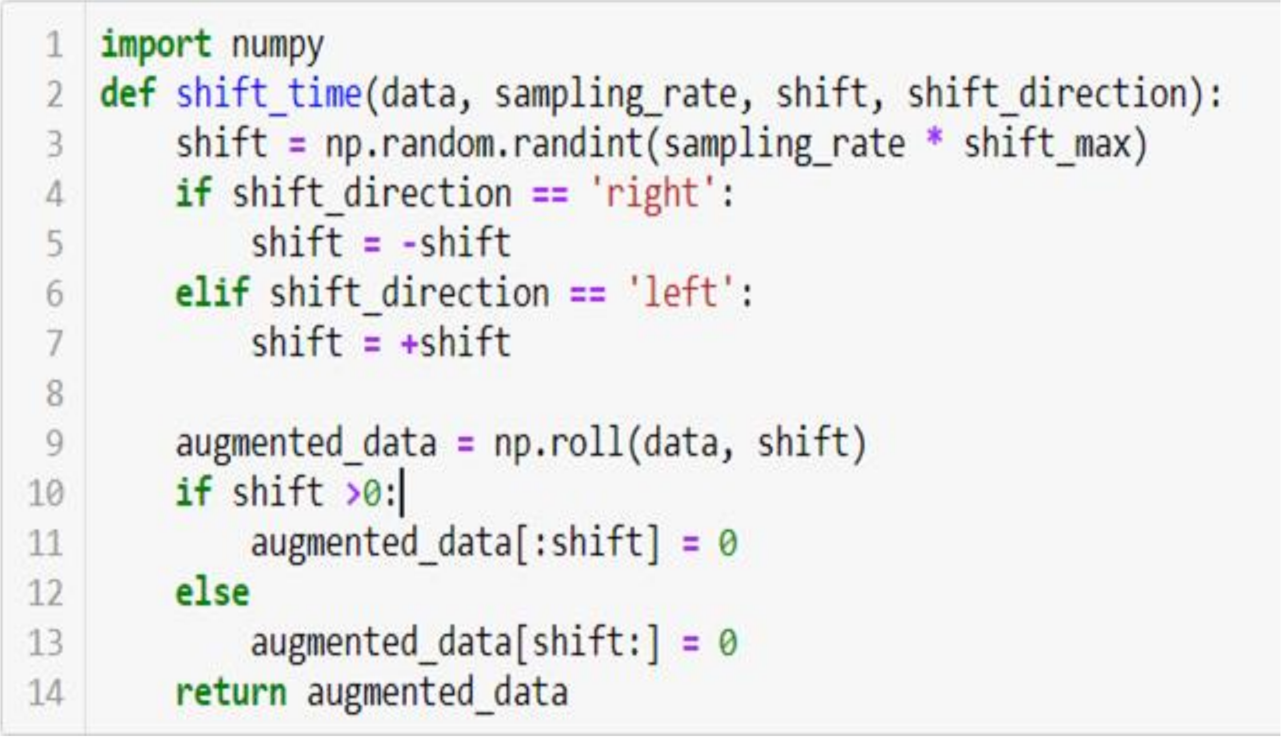
augmented_data = data + noise_factor*noise





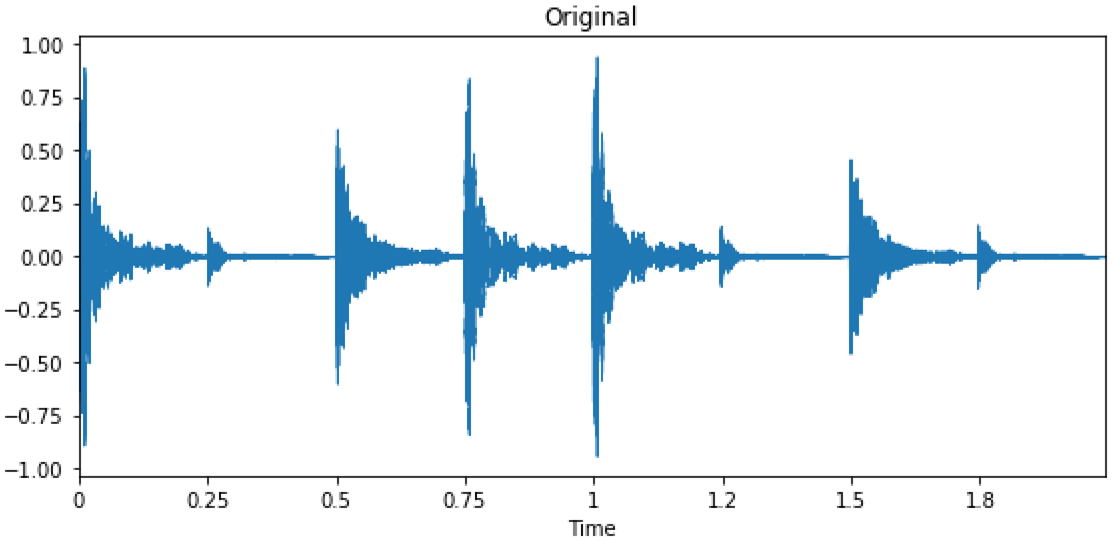
AUGMENTATION TECHNIQUES

- **Shifting Time:** Shifting audio to left/right just fast-forward or back-forward the audio by a random second. In shifting audio to right with x seconds, last x seconds will mark as 0.

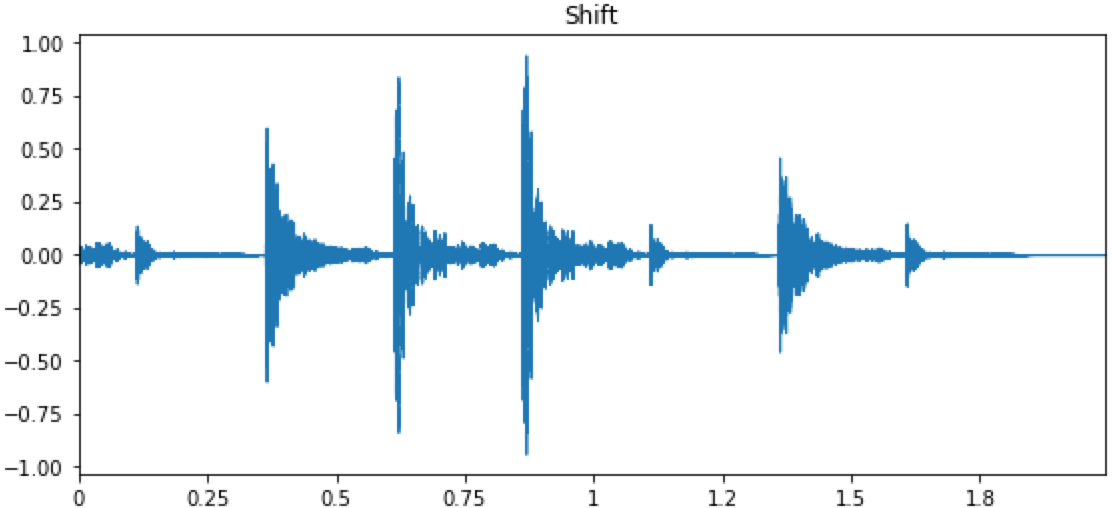


```
import numpy
def shift_time(data, sampling_rate, shift, shift_direction):
    shift = np.random.randint(sampling_rate * shift_max)
    if shift_direction == 'right':
        shift = -shift
    elif shift_direction == 'left':
        shift = +shift

    augmented_data = np.roll(data, shift)
    if shift >0:
        augmented_data[:shift] = 0
    else
        augmented_data[shift:] = 0
    return augmented_data
```



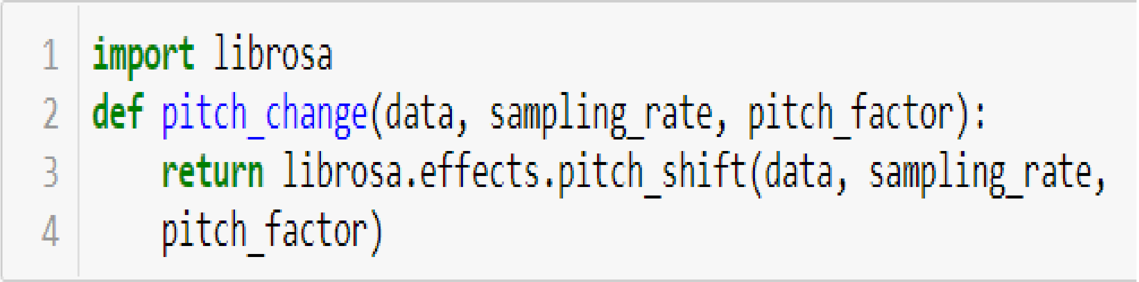
Original



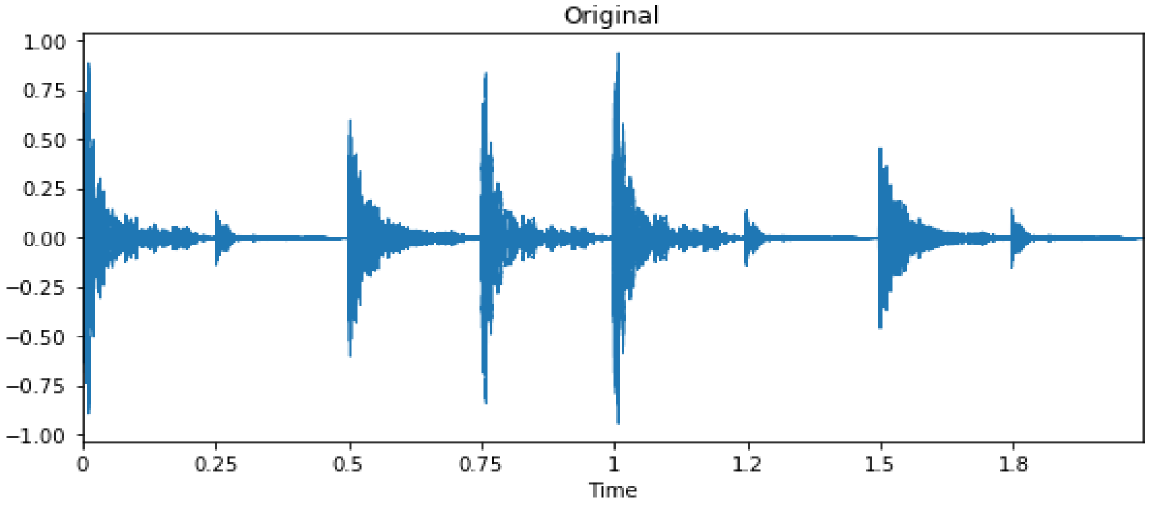
Shift

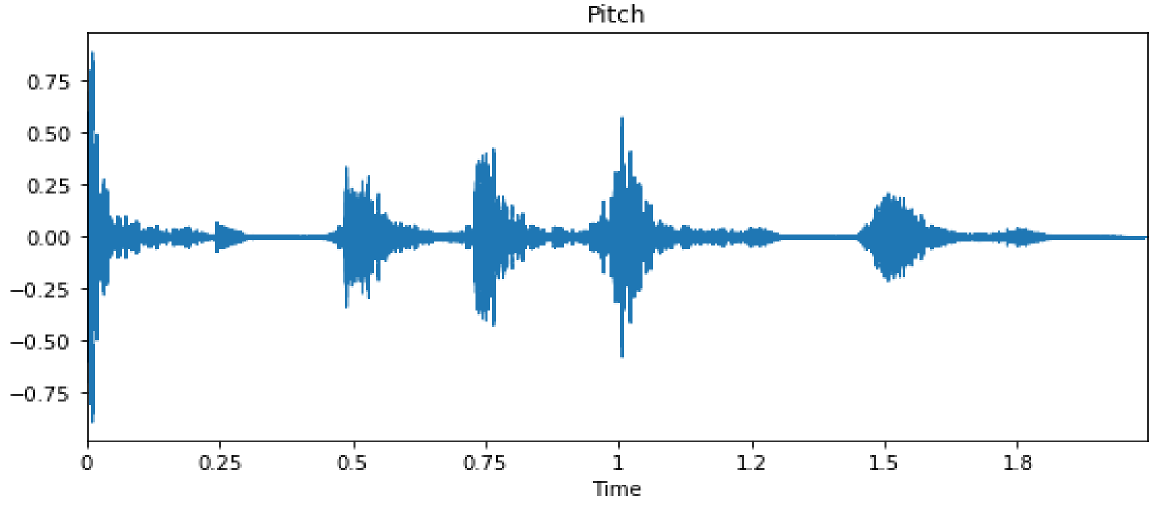
AUGMENTATION TECHNIQUES

Changing Pitch: Changing the pitch randomly to augment audio data.



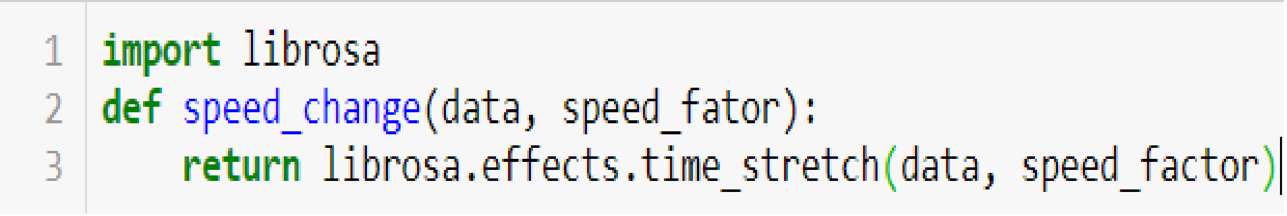
```
import librosa
def pitch_change(data, sampling_rate, pitch_factor):
    return librosa.effects.pitch_shift(data, sampling_rate,
    pitch_factor)
```



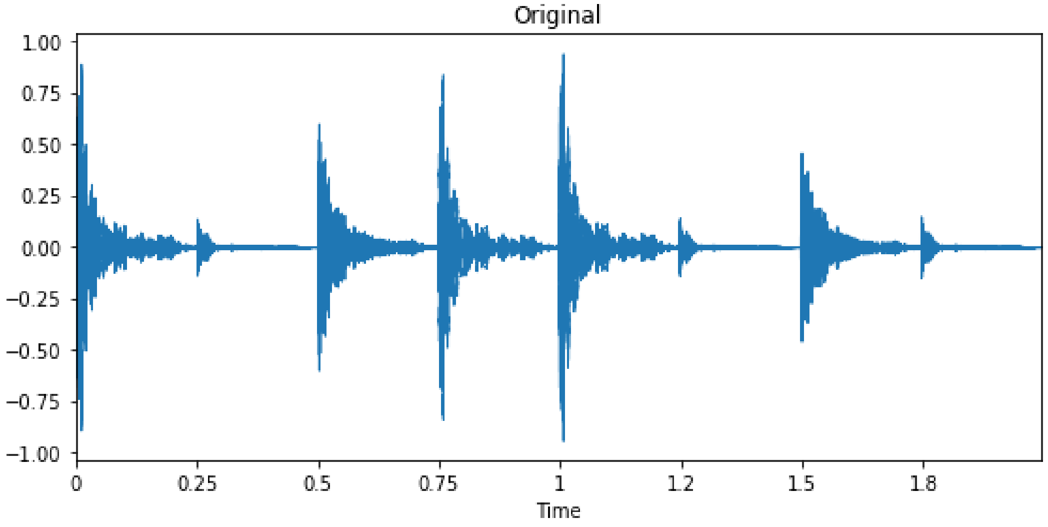


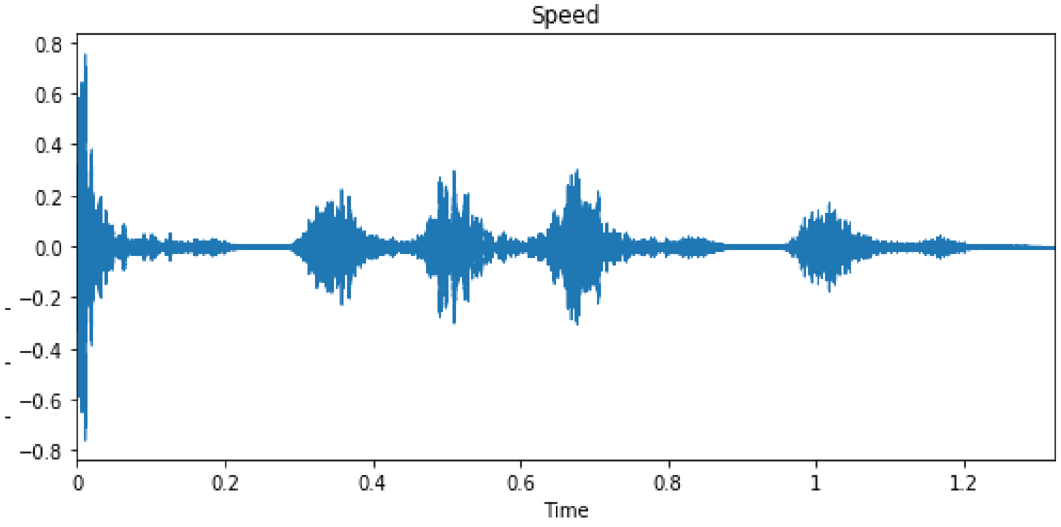
AUGMENTATION TECHNIQUES

Changing Speed: Changing speed stretches the time series by a fixed rate.



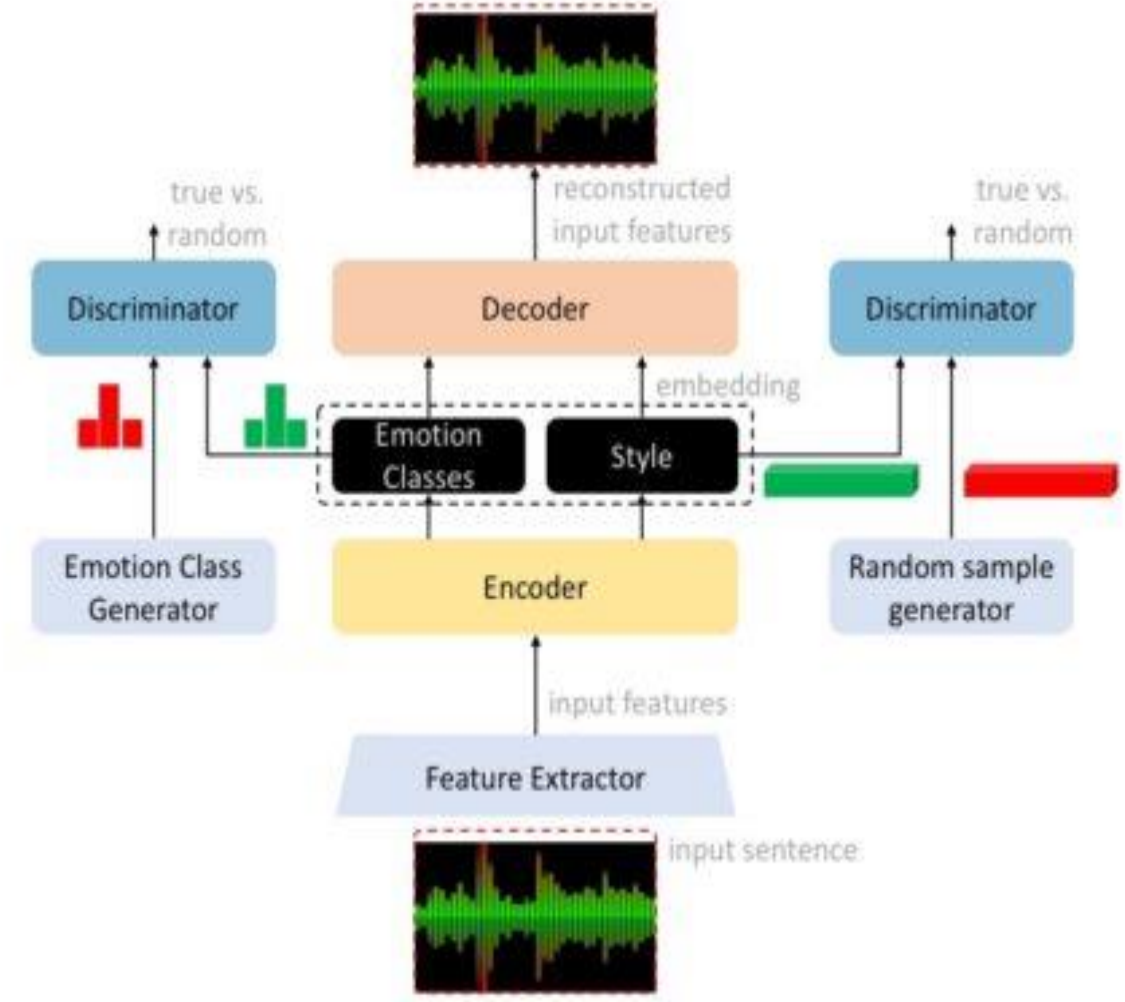
```python
import librosa
def speed_change(data, speed_fator):
    return librosa.effects.time_stretch(data, speed_factor)
```





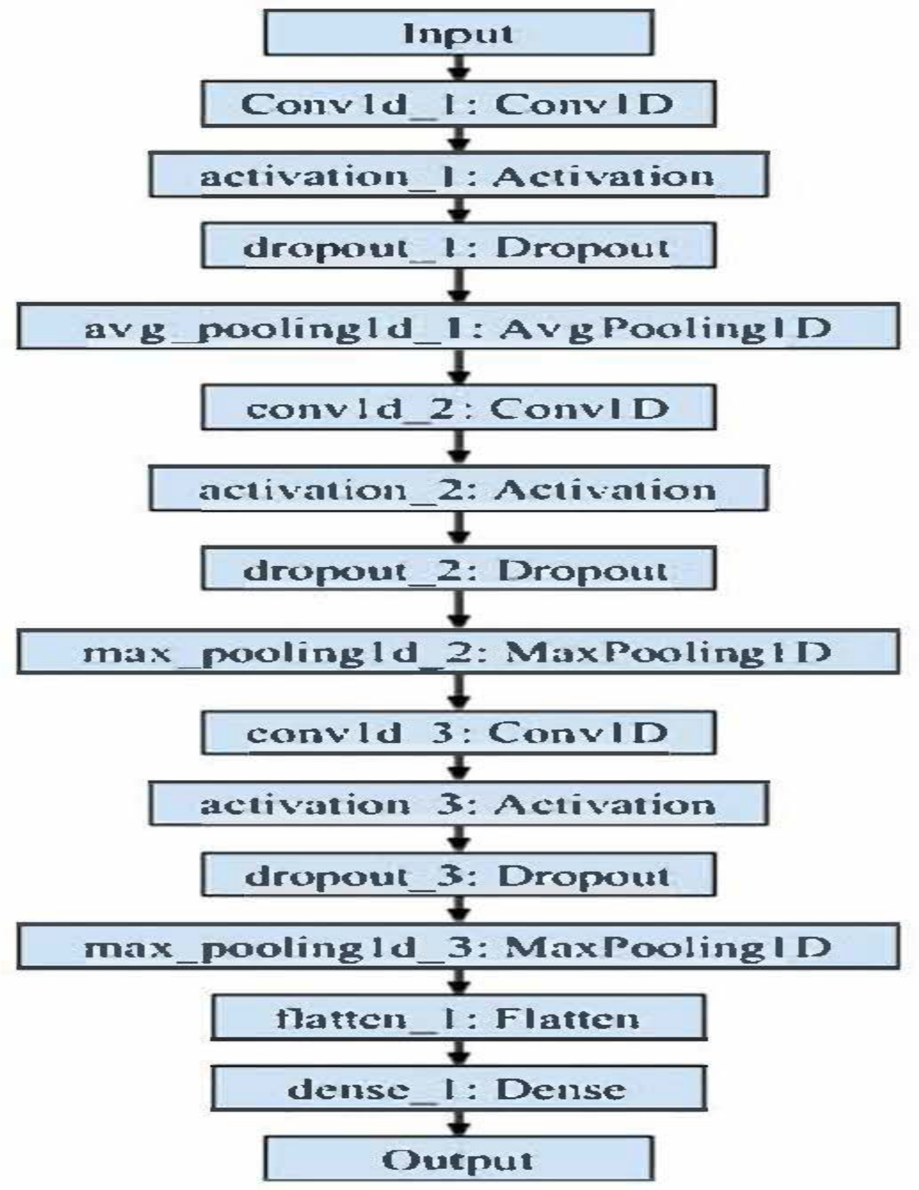
AUGMENTATION TECHNIQUES

Generative Adversial Networks

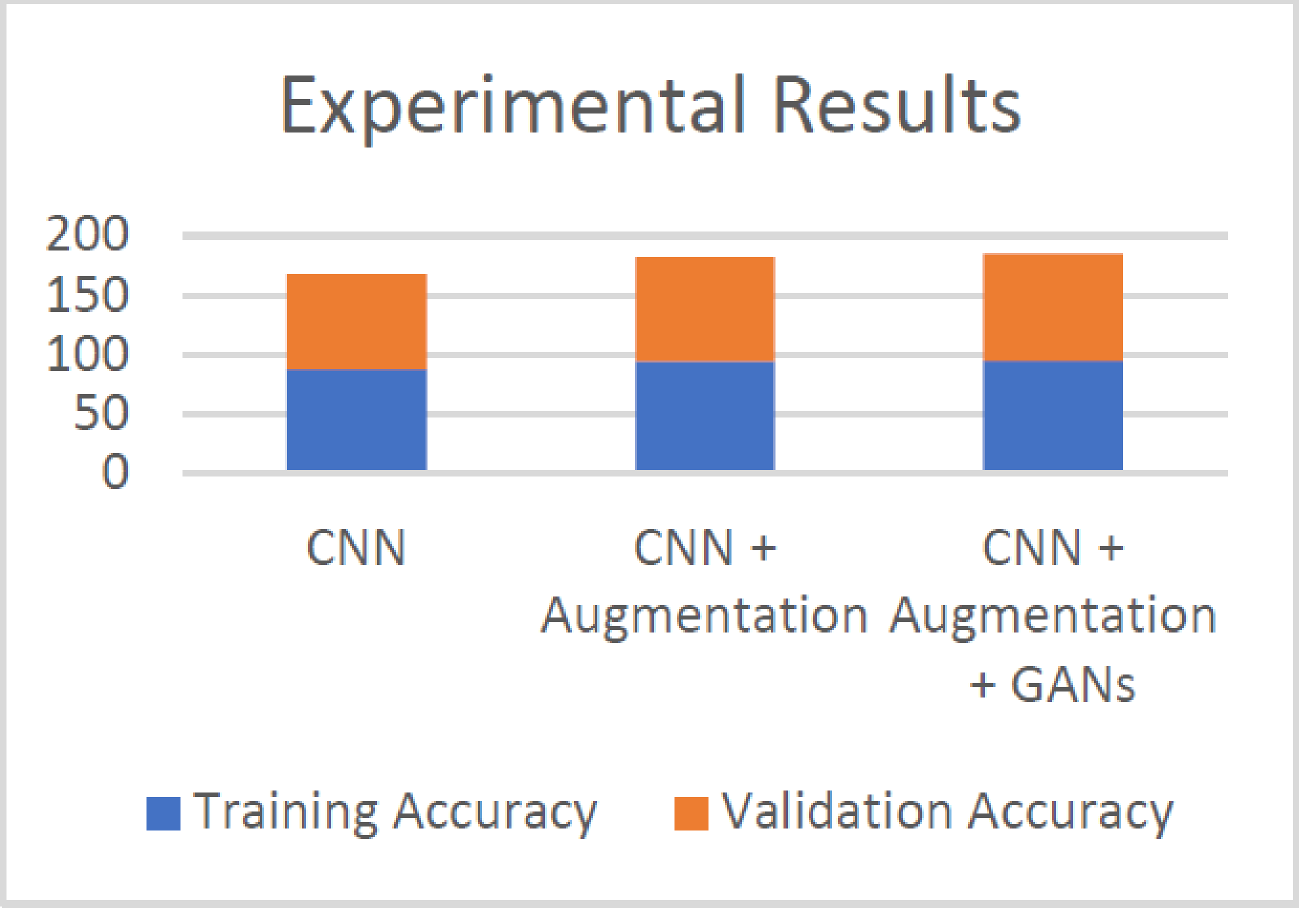


® Improving Emotion Classification through Variational Inference of Latent Variables, IEEE document 8682823

Proposed Novel Convolutional Neural Network Architecture



RESULTS


Experimental Results
200
150
100
50
0
CNN
CNN + Augmentation
CNN + Augmentation + GANs
Training Accuracy
Validation Accuracy

Thank you & Questions