

Snap Point

팀 & 프로젝트 소개





안녕하세요~ 저희는 18기 신입 오중균, 김창영으로 이루어진 팀 **JitterBug**라고 합니다.
이번 학기 에코노베이션 머신러닝 프로젝트를 하게 되었습니다.



프로젝트명은 **SnapPoint**인데요.

손가락을 튕기는 행위인 Snap과 PowerPoint의 Point를 합쳐서 만든 이름입니다.

이름에서도 쉽게 알 수 있듯이 Snap으로 PowerPoint를 제어해보자라는 의미입니다.

동기

프레젠테이션 스킬을 높일 수 있는 방법 ##자기PR ##팀프로젝트 ##스피치
나 자신을 특별하게 만들어 줄 아이디어

[프레젠테이션 제어]를 위한 핑거스냅 기반의 인터페이스

- 핑거스냅의 편의성
- 청중의 집중도 향상
- 발표자의 개성 표출

| 인싸템~!!!!

데이터

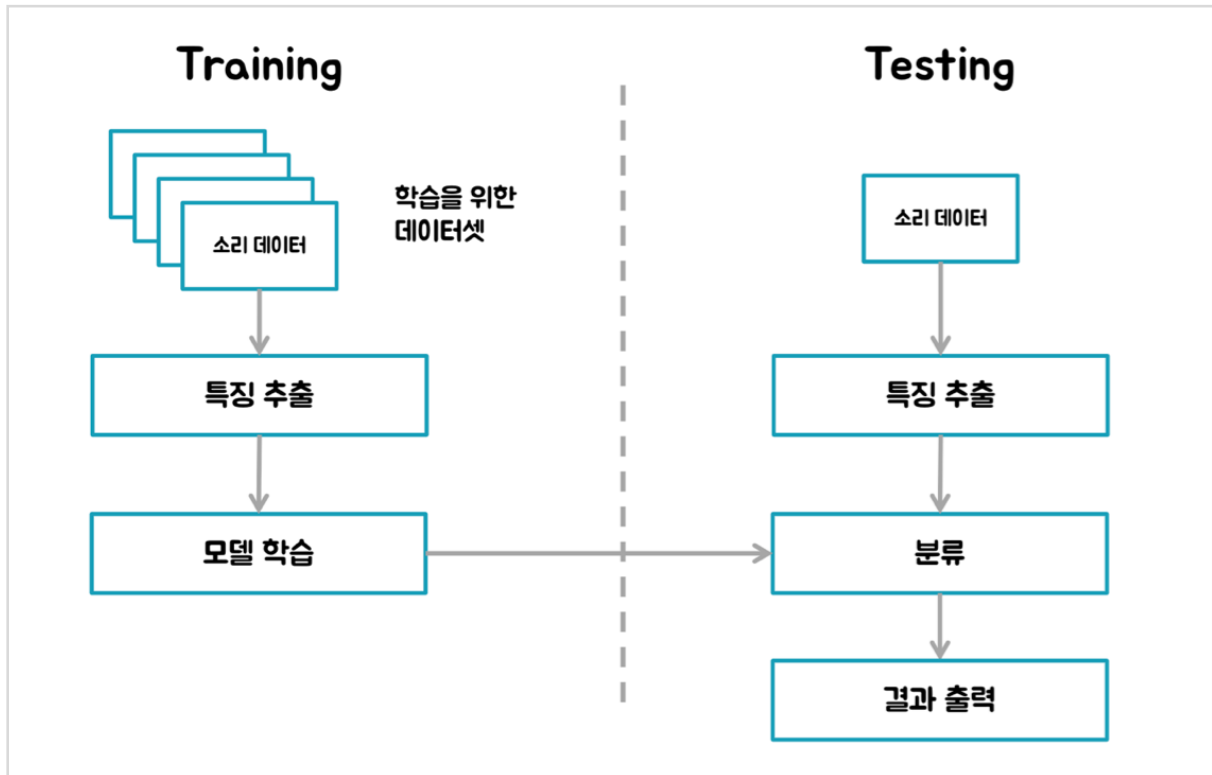
역시나 머신러닝에서 가장 중요한 건 데이터 입니다.

소리를 이용하는 것이기 때문에 당연히 음성데이터가 필요하겠죠?

저희 프로젝트를 귀엽게 봐주신 여러분들이 열심히 데이터를 보내주신 덕분에 총 12명에서 **약 1000개의 데이터**를 모을 수 있었습니다.

이 천개의 데이터를 모델에 넣기에는 약간 수가 모자랄 수 있기 때문에 인위적으로 **데이터를 늘려주는 기법 (Data Augmentation)**을 사용했습니다.

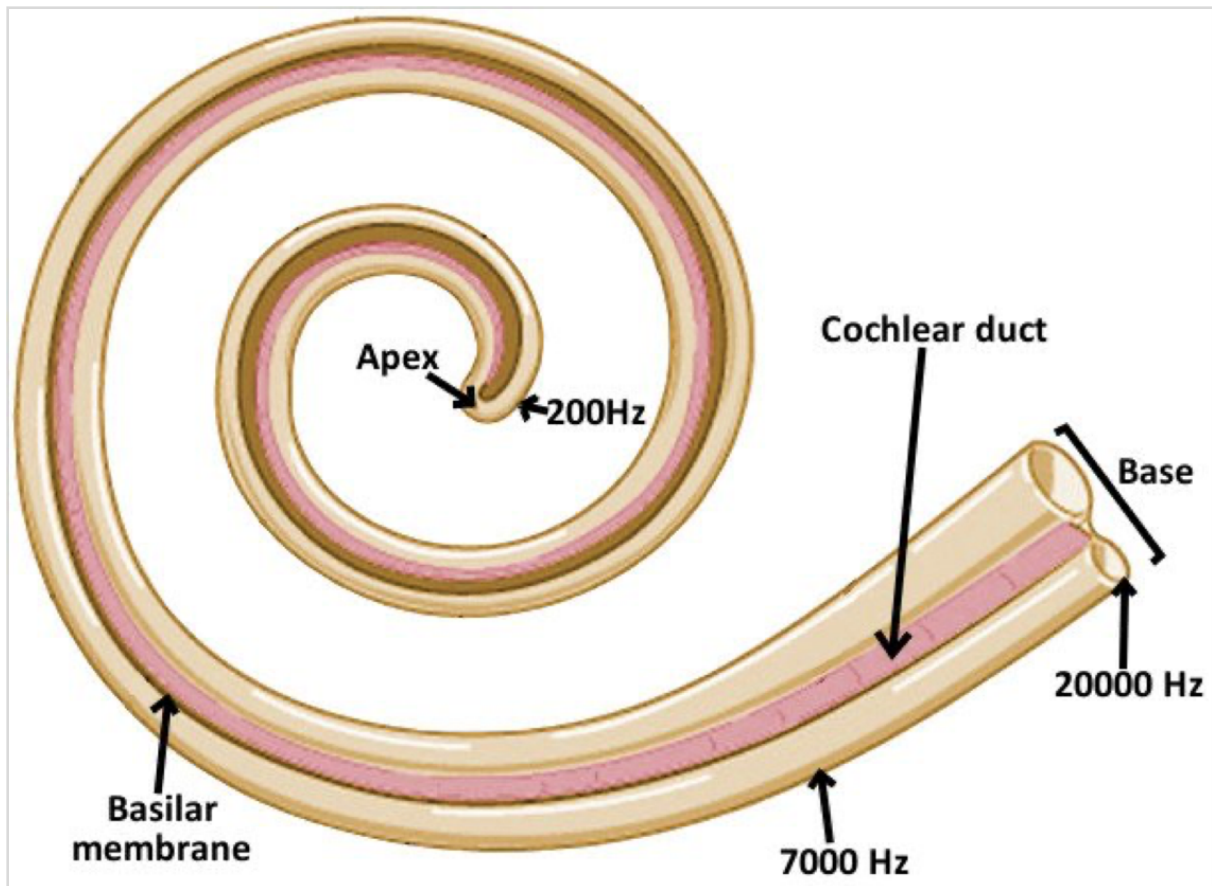
음의 높이를 높이는 방식 과 노이즈를 섞어주는 방식 으로 2000개의 데이터를 더 만들어서 총 데이터수 3000개가 되었습니다.



이 데이터들을 그냥 모델에 넣어주면 안됩니다. 데이터 **전처리(Preprocessing)**를 거쳐야합니다.

모델 성능에 직접적인 영향을 미치는 과정이기 때문에 중요하게 다뤄지는 과정입니다.

음성 데이터의 특성을 추출할 때 주로 사용하는 Librosa의 **MFCC 알고리즘**을 이용하여 전처리를 해주었습니다.



라이브러리

이 프로그램을 위해 파이썬 라이브러리를 여럿 사용했습니다.

음성 데이터의 특징을 추출하기 위한 **Librosa**,

소리를 계속 받아들이기 위한 **PyAudio**,

외부 프로그램을 제어하기 위한 **PyAutoGUI**

그리고 머신러닝할 때 주로 쓰는 **TensorFlow**와 **Keras**를 사용했습니다.

머신러닝 모델 만들기

핑거스냅 소리를 인식하는 머신러닝 모델 (CNN)

```
model = tf.keras.models.Sequential()
model.add(Conv2D(32, kernel_size, padding='same', input_shape=(18,128,1)))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=pool_size))
model.add(Dropout(0.5))

model.add(Conv2D(64, kernel_size, padding='same'))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=pool_size))
```

```

model.add(Dropout(0.5))

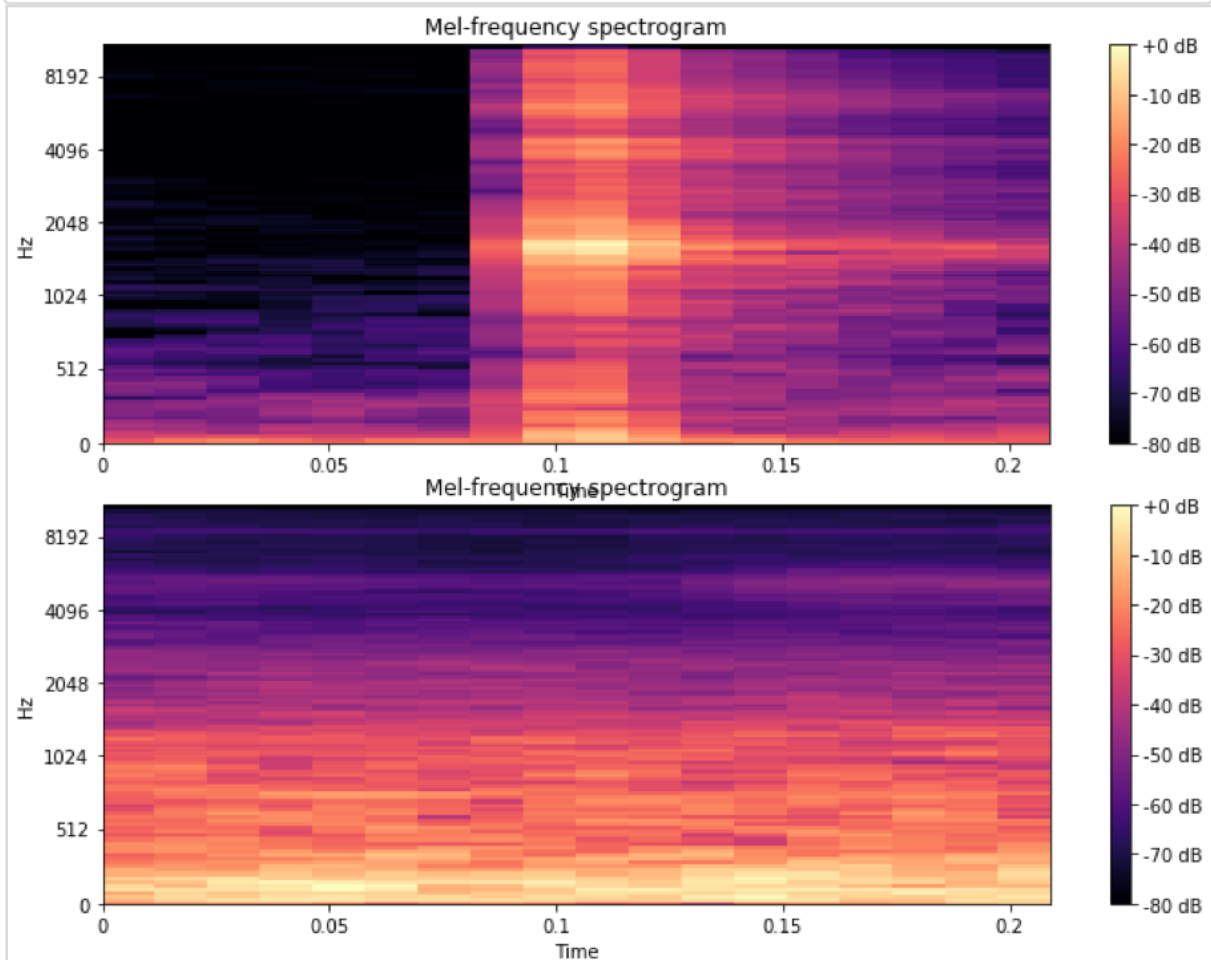
model.add(Conv2D(128, kernel_size, padding='same'))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=pool_size))
model.add(Dropout(0.5))

model.add(Flatten())
model.add(Dense(600))
model.add(Activation('relu'))
model.add(Dropout(0.5))

model.add(Dense(10))
model.add(Activation('softmax'))

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

```



프로그램 작동 원리

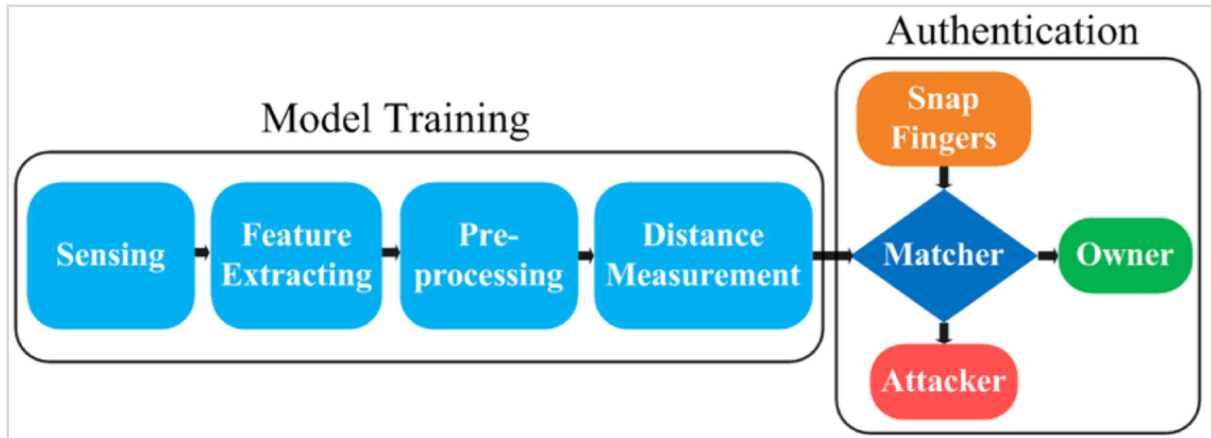
프로그램이 돌아가는 원리는 아주 간단합니다.

우선 프로그램이 시작되면 이미 만들어놓은 머신러닝 모델을 가져와서 로드를 합니다.

로드를 하면 PyAudio를 이용하여 프로그램이 주위의 소리를 들을 수 있게 스트림이 실행시킵니다.

스트림이 돌아가는 동안 받은 음성데이터를 빠른 속도로 가공(전처리)하여 모델에 집어넣을 수 있게 합니다.

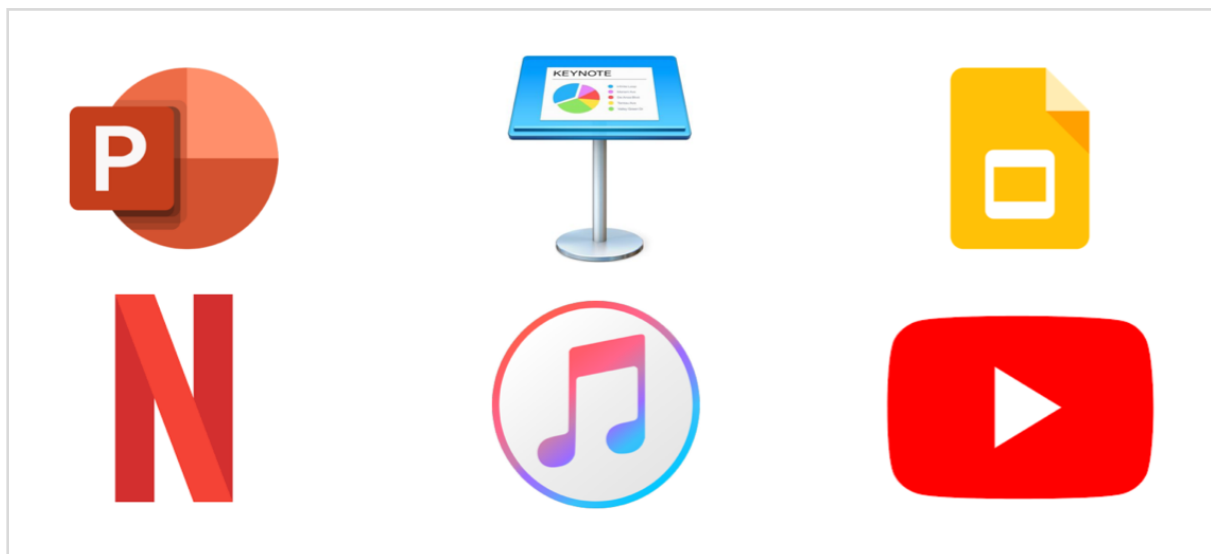
그래서 스냅이 발생하는 순간, 전처리된 데이터가 모델에 들어가 스냅이 맞다라고 판단하면 PyAutoGUI를 통해 우리가 원하는 동작을 할 수 있게 합니다.



사용법

1. 실행시킨다.
2. 활용하고자 하는 프로그램(e.g. Youtube)에 포커스를 준다
3. Snap (핑거스냅을 못하는 사람은 박수소리로 제어해도 무관)

활용



- YouTube, Netflix, Music Player - Toggle Play & Pause

- YouTube, Music Player - Next Video, Next Track
- PowerPoint, Keynote - Next Slide

상상의 나라

- Snap Beeper : 열쇠, 리모콘, 에어팟등과 같이 잃어버리는 쉬운 물건에 작은 기기를 달아서 손가락을 튕기면 소리가 울리게 하여 위치를 알림.
- Snap Snoozer : 좀만 더 자고 싶을때 손가락을 튕기면 몇분동안 Snooze 되게 함.