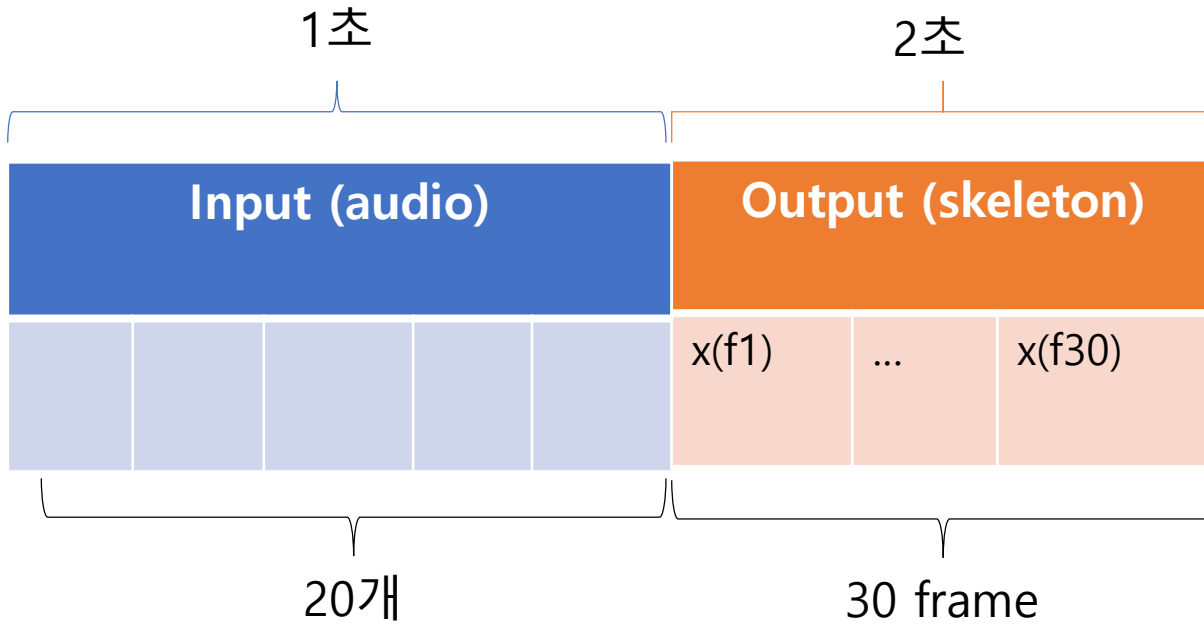
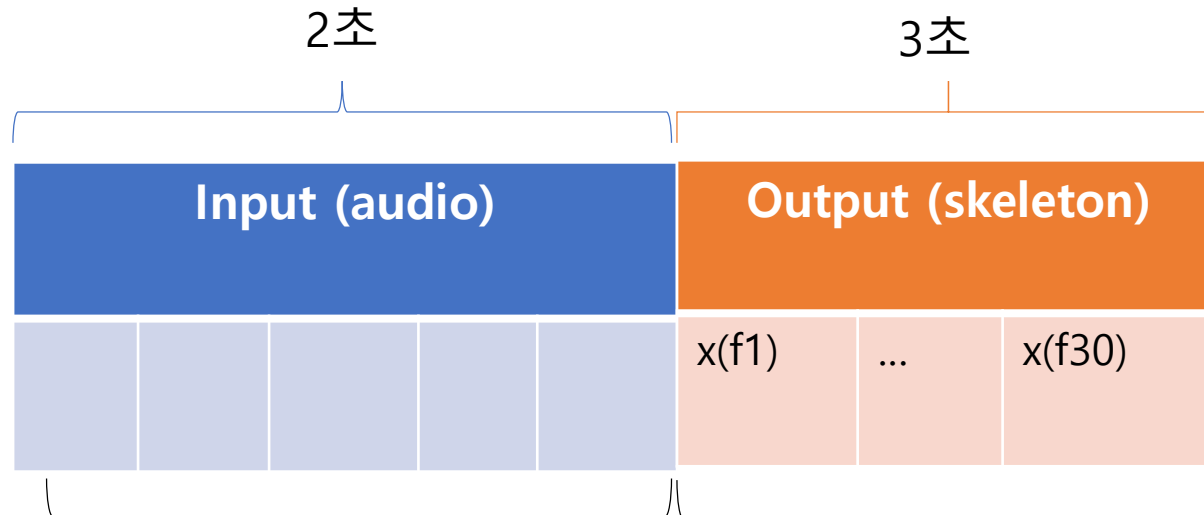


Train data 구축



- 검지의 6번 관절에 대한 x 좌표를 의미
(일단은 x만 해서 테스트해보고 나중에 y, z 추가하는 걸로 해요..)
- f는 frame을 의미



Train data 구축 알고리즘 만들때 참고 사항

<구현 참고사항>

- Mfcc feature 개수 변경 가능하게 20개, 40개, 60개
- 총길이 1초 조절 가능하도록 (ex. 1초, 2초, 3초 등)
- Train / test 데이터 개수 설정 가능하도록

확인해야할것

- 아래 소스코드의 x_train 데이터 타입이 numpy 인지 pandas 인지 확인하고 우리 데이터 type도 동일하게 맞춰야함
- https://github.com/minji-OH/Analysis_in_Python/blob/master/Comparing%20GRU%20and%20LSTM

```
# scale the data
from sklearn.preprocessing import MinMaxScaler
sc = MinMaxScaler(feature_range=(0,1))
ts_train_scaled = sc.fit_transform(ts_train)

# create training data of s samples and t time steps
X_train = []
y_train = []
for i in range(time_steps, ts_train_len-1):
    X_train.append(ts_train_scaled[i-time_steps:i, 0])
    y_train.append(ts_train_scaled[i:i+for_periods, 0])
X_train, y_train = np.array(X_train), np.array(y_train)

# Reshaping X_train for efficient modelling
X_train = np.reshape(X_train, (X_train.shape[0], X_train.shape[1], 1))

inputs = pd.concat((all_data["Adj Close"][:, '2018'], all_data["Adj Close"][:, '2019']), axis=0).values
inputs = inputs[len(inputs)-len(ts_test)-time_steps:]
inputs = inputs.reshape(-1,1)
inputs = sc.transform(inputs)

# Preparing X_test
X_test = []
for i in range(time_steps, ts_test_len + time_steps - for_periods):
    X_test.append(inputs[i-time_steps:i,0])

X_test = np.array(X_test)
X_test = np.reshape(X_test, (X_test.shape[0], X_test.shape[1], 1))

return X_train, y_train, X_test, sc
```

확인해야할 것

- 결과적으로 아래와 같은 train, test 데이터가 나와야함
- test data는 200개정도 나오면 될듯

	X_train					Y_train	
	x_{t-4}	x_{t-3}	x_{t-2}	x_{t-1}	x_t	y_{t+1}	y_{t+2}
	0	1	2	3	4	0	1
0	257.31	258.48	259.15	268.46	266.38	266.35	265.34
1	258.48	259.15	268.46	266.38	266.35	265.34	267.94
2	259.15	268.46	266.38	266.35	265.34	267.94	272.73
3	268.46	266.38	266.35	265.34	267.94	272.73	271.90
4	266.38	266.35	265.34	267.94	272.73	271.90	268.93
...
1499	1591.91	1520.91	1551.48	1495.08	1460.83	1377.45	1343.96
1500	1520.91	1551.48	1495.08	1460.83	1377.45	1343.96	1470.90
1501	1551.48	1495.08	1460.83	1377.45	1343.96	1470.90	1461.64
1502	1495.08	1460.83	1377.45	1343.96	1470.90	1461.64	1478.02
1503	1460.83	1377.45	1343.96	1470.90	1461.64	1478.02	1501.97

1504 rows x 7 columns

	X_test				
	x_{t-4}	x_{t-3}	x_{t-2}	x_{t-1}	x_t
	0	1	2	3	4
0	1343.96	1470.90	1461.64	1478.02	1501.97
1	1470.90	1461.64	1478.02	1501.97	1539.13
2	1461.64	1478.02	1501.97	1539.13	1500.28
3	1478.02	1501.97	1539.13	1500.28	1575.39
4	1501.97	1539.13	1500.28	1575.39	1629.51
...
244	1760.33	1760.94	1769.21	1790.66	1784.03
245	1760.94	1769.21	1790.66	1784.03	1792.28
246	1769.21	1790.66	1784.03	1792.28	1786.50
247	1790.66	1784.03	1792.28	1786.50	1793.00
248	1784.03	1792.28	1786.50	1793.00	1789.21

249 rows x 5 columns