『IMU 센서 데이터를 이용한 동작분류 모델 만들기』

CNN 분류모델 만들기

충북대학교 산업인공지능학과

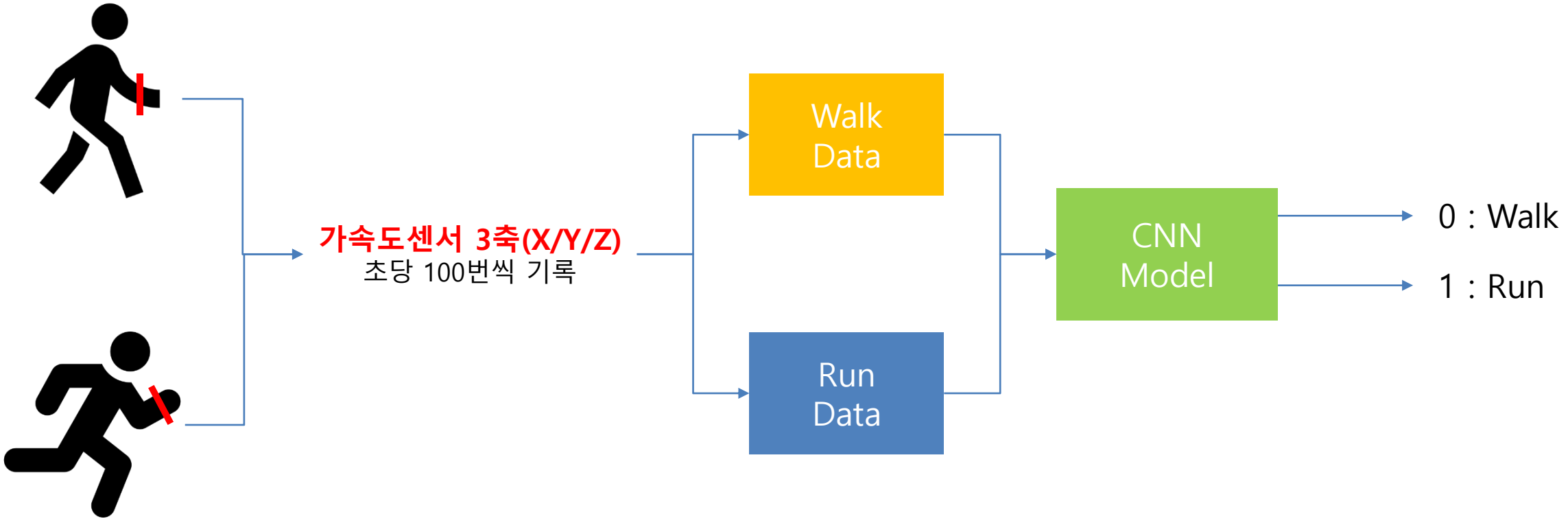
2023254015 장욱진

충북대학교
CHUNGBUK NATIONAL UNIVERSITY

1. 주제선정 이유

IMU 센서 – 자이로(3축), 가속도(3축), 지자기(3축)

2. 프로젝트 목표


가속도센서 3축(X/Y/Z)
초당 100번씩 기록
Walk Data
Run Data
CNN Model
0 : Walk
1 : Run

3. 데이터셋

PAMAP2 Physical Activity Monitoring Data Set

Download: Data Folder, Data Set Description

Abstract: The PAMAP2 Physical Activity Monitoring dataset contains data of 18 different physical activities, performed by 9 subjects wear

Data Set Characteristics:	Multivariate, Time-Series	**Number of Instances:**	3850505	**Area:**	Computer
Attribute Characteristics:	Real	**Number of Attributes:**	52	**Date Donated**	2012-08-06
Associated Tasks:	Classification	**Missing Values?**	Yes	**Number of Web Hits:**	95176

archive.ics.uci.edu/ml/datasets/PAMAP2+Physical+Activity+Monitoring

Attribute

1. Timestamp
2. Activity ID
3. 심박수(bpm)

4~20. IMU 손

21~37. IMU 가슴

38~54. IMU 발목

Activity ID

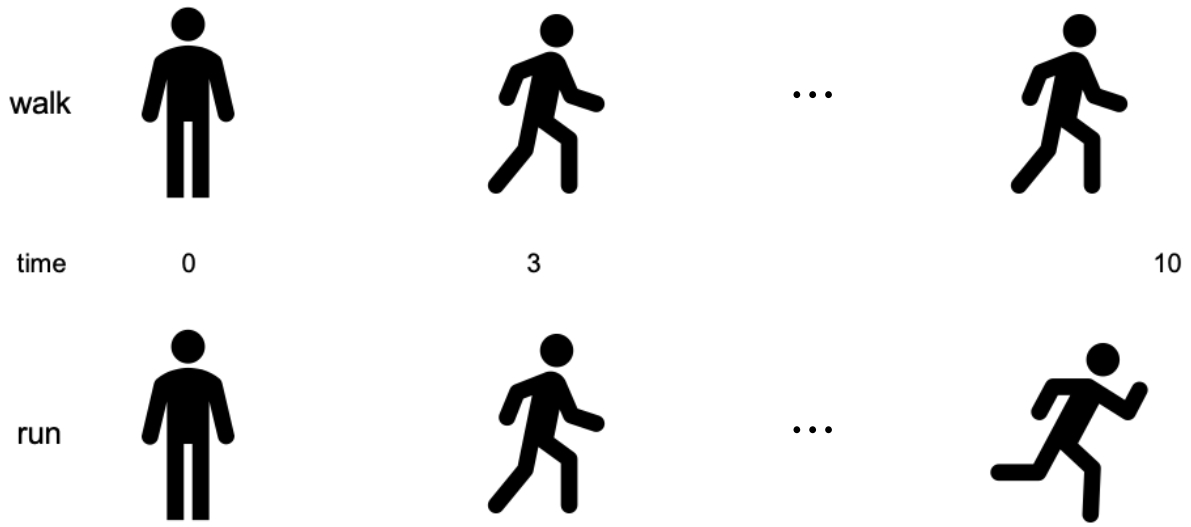
- 1 lying
- 2 sitting
- 3 standing
- 4 walking
- 5 running
- 6 cycling
- 7 Nordic walking
- 9 watching TV
- 10 computer work
- 11 car driving
- 12 ascending stairs
- 13 descending stairs
- 16 vacuum cleaning
- 17 ironing
- 18 folding laundry
- 19 house cleaning
- 20 playing soccer
- 24 rope jumping
- 0 other (transient activities)

1	2	3	4	5	6	7	8	9	10
Time stamp	Activity ID	Heart Rate	temperature	3D acceleration	3D acceleration	3D acceleration	3D acceleration	3D acceleration	3D acceleration
11	12	13	14	15	16	17	18	19	20
3D gyroscope	3D gyroscope	3D gyroscope	3D magnetometer	3D magnetometer	3D magnetometer	orientation	orientation	orientation	orientation
21	22	23	24	25	26	27	28	29	30
temperature	3D acceleration	3D acceleration	3D acceleration	3D acceleration	3D acceleration	3D acceleration	3D gyroscope	3D gyroscope	3D gyroscope
31	32	33	34	35	36	37	38	39	40
3D magnetometer	3D magnetometer	3D magnetometer	orientation	orientation	orientation	orientation	temperature	3D acceleration	3D acceleration
41	42	43	44	45	46	47	48	49	50
3D acceleration	3D acceleration	3D acceleration	3D acceleration	3D gyroscope	3D gyroscope	3D gyroscope	3D magnetometer	3D magnetometer	3D magnetometer
51	52	53	54						
orientation	orientation	orientation	orientation						

파일당 행 구조는 다음과 같고,
추출하고자 하는 정보(열)는 5개

Hand IMU
Chest IMU
Ankle IMU

4-1. 데이터 전처리

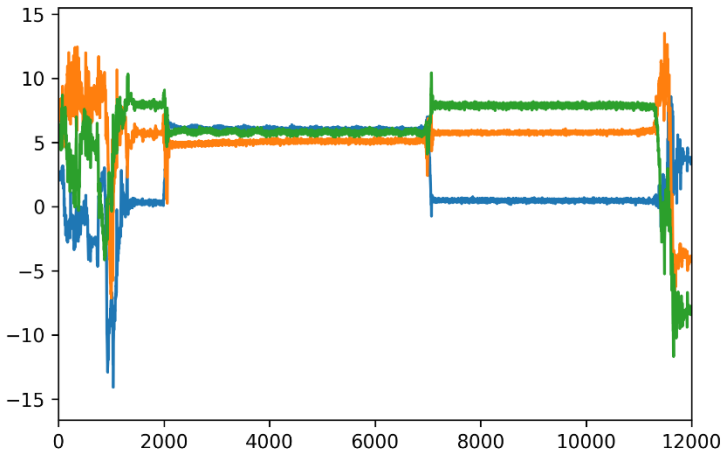


문제점

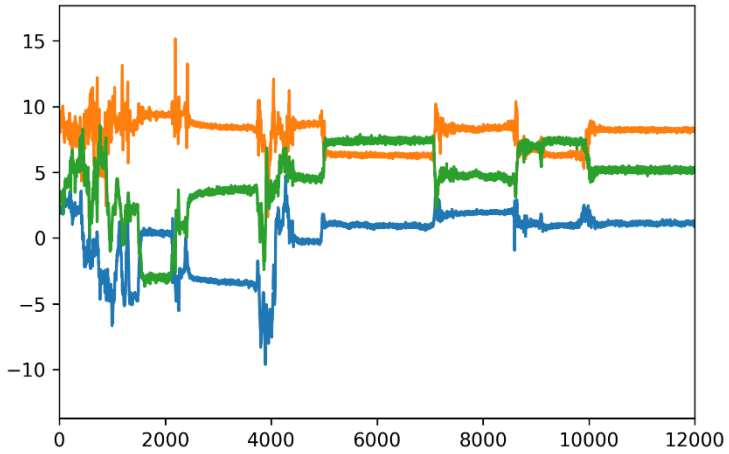
1. 사람마다 걷기/뛰기 속도가 다름.
2. 24가지 동작을 같은 단위로 잘랐을때
   동작마다 다른 특성 때문에 특정 단위가
   모든동작에 적절한 동작이 되기 어려움

Ex) 걷는동작 달리는동작은 10초 단위 구분가능,
    하지만 눕는동작 앉는동작엔 부적합할수 있음.

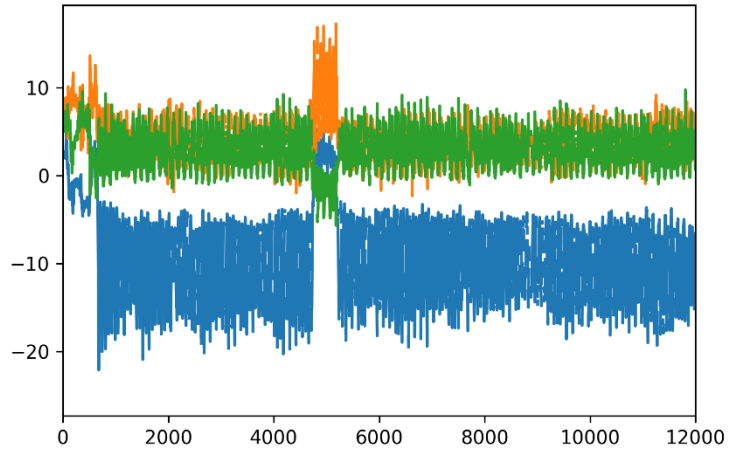
4-2. 데이터 전처리



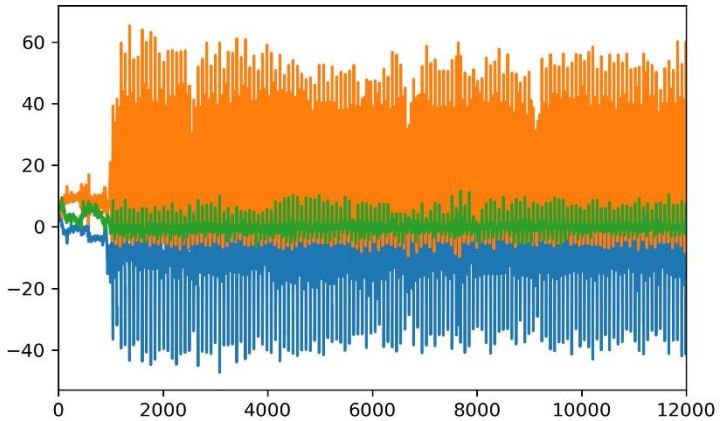
1. Lying



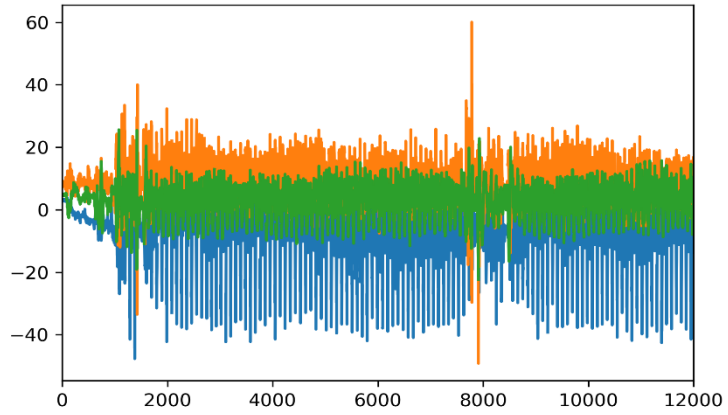
2. Sitting



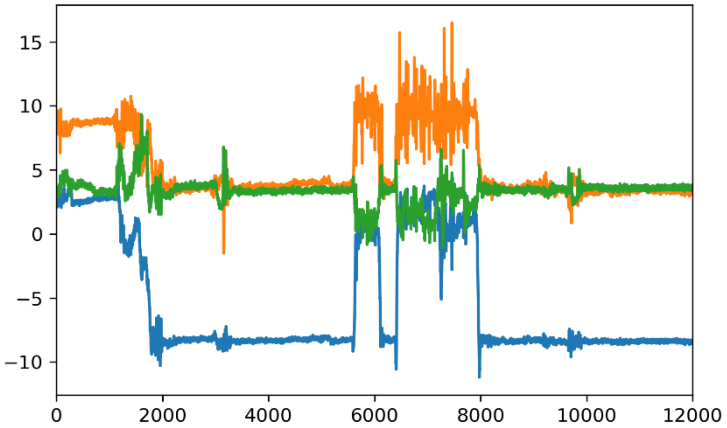
3. Walking



4. Running



5. Rope Jumping



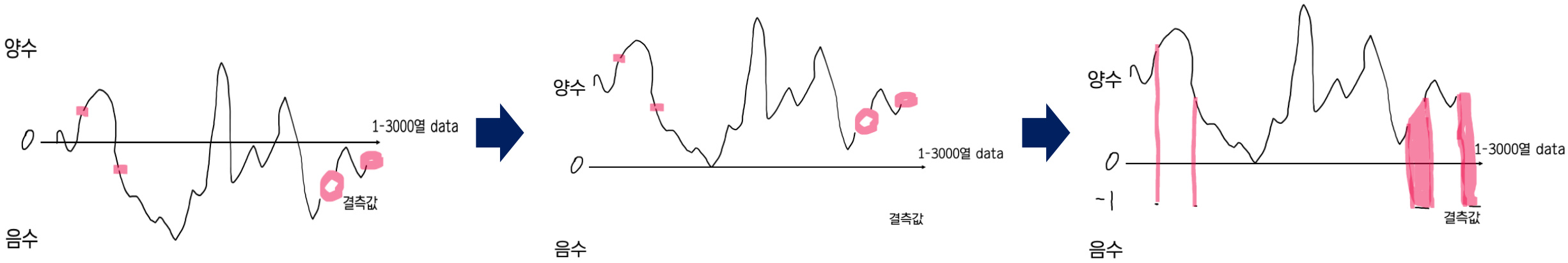
6. Standing

4-3. 데이터 전처리

-4.13618	-4.13778	-4.16675	-4.2769	-4.46291	-4.83892	-4.97831	-5.50789	-6.06029	-7.02785	-8.03934	4	walking	subject1
-7.23051			-9.39931	-10.2886	-11.34	-12.1285		-11.9114	-11.5503	-11.4152	4	walking	subject1
-12.0875	-11.1998	-10.3716	-9.7407	-9.11673	-8.36254	-7.98706	-7.72976	-7.42745	-7.13155	-7.05784	4	walking	subject1
-3.64746	-4.19854	-5.00238	-6.17395	-7.53607	-8.85665	-9.58999	-10.2445	-10.8025	-11.6326	-12.5701	4	walking	subject1

33	-14.8085	-15.0843	-15.1652	-15.1692	-14.9101	-14.4686	-14.2536	-13.872	-13.4233	-13.2009	-12.6474	-11.8238	-10.9192	4 walking subject101
34														4 walking subject101
35	-8.96762	-8.85721	-9.09768	-9.5418	-9.97817	-10.5429	-10.7072	-10.3917	-11.2348	-13.6575	-17.8654	-26.5863	-37.6843	5 running subject101

결측값 처리



최솟값을 모든 데이터에 더해주어
최솟값을 0으로 만들기

결측값 -1로 채우기

4-4. 데이터 전처리

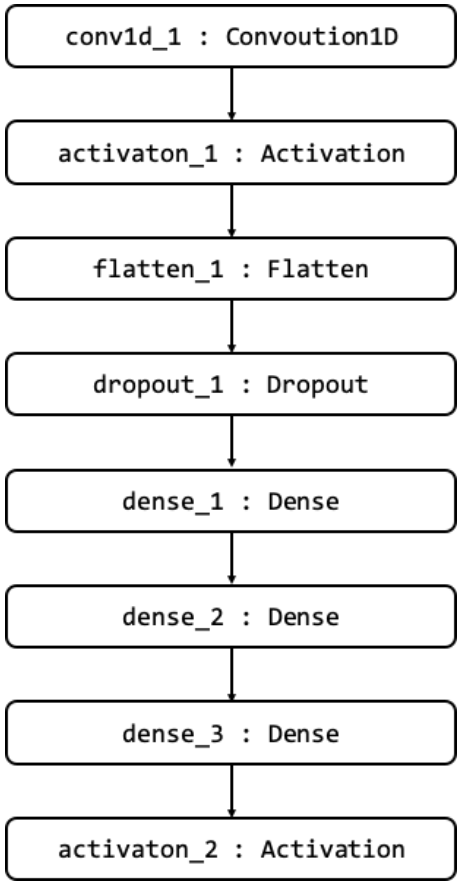
학습데이터 | 검증 데이터 | 테스트 데이터 → 6.5 : 1.5 : 2

```
# Keras model with one Convolution1D layer
# unfortunately more number of covnolutional layers, filters and filters lenght
# don't give better accuracy
model = Sequential()
model.add(Convolution1D(nb_filter=512, filter_length=1, input_shape=(nb_features, 3)))
model.add(Activation('relu'))
model.add(Flatten())
model.add(Dropout(0.4))
model.add(Dense(2048, activation='relu'))
model.add(Dense(1024, activation='relu'))
model.add(Dense(nb_class))
model.add(Activation('softmax'))


y_train = np_utils.to_categorical(y_train, nb_class)
y_valid = np_utils.to_categorical(y_valid, nb_class)

sgd = SGD(lr=0.01, nesterov=True, decay=1e-6, momentum=0.9)
model.compile(loss='categorical_crossentropy',optimizer=sgd,metrics=['accuracy'])

nb_epoch = 15
model.fit(X_train_r, y_train, nb_epoch=nb_epoch, validation_data=(X_valid_r, y_valid), batch_size=16)
```



5. 변인

- **Convolution1D를 사용할 것인가? 2D를 사용할 것인가?**
- **Convolution층은 몇개로 구성할 것인가?**
- **각각 층의 필터 사이즈 / 필터 수 / stride 는 어떻게 할 것인가?**
- Activation층의 활성화 함수로는 어떤 것을 사용할 것인가?
- **Dense층은 몇개로 구성할 것인가? 뉴런의 수는?**
- Dropout은 몇으로 설정할 것인가?
- **batch size는 몇으로 지정할 것인가?**
- epoch는 몇으로 지정할 것인가?
- 더 나아가면 model.compile부분에서 loss함수로는 무엇을 사용할 것인지? 모맨텀은?

Conv1d vs Conv2d -> Conv층수 -> 필터수 -> 필터 사이즈 -> 배치사이즈 ...

감사합니다

충북대학교
CHUNGBUK NATIONAL UNIVERSITY