Final Project

電機乙 碩一 311511053 徐奕翔

Introduction

情緒辨識可以應用在許多地方，像是電影的試映會，給參加者配戴相關儀器來偵測他們的情緒變化，就能在電影上映前知道這電影上映之後銷量會不會不佳，以利電影公司來做最後的修正。

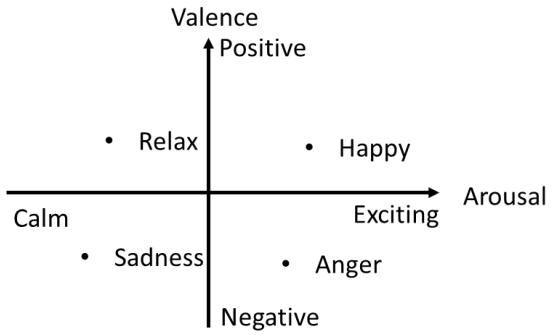
而其中一種情緒辨識的方法是利用心電圖(ECG) ，而目前透過心電圖辨識情緒的 dataset，大多數於靜態的，也就是受測者看完一部影片只 label 一種情緒，而我的 final project 想做的是動態的，也就是受測者在看一部影片中，隨著時間變化情緒也會跟著變化，這也更貼近實際情況。

- Static dataset: only one emotion in watching a video

- Dynamic dataset: may have more then one emotion in watching a video

Dataset

在介紹資料及之前，要先介紹衡量情緒的兩種 metric: valence 以及 arousal，valence 代表的是情緒的正向程度，越高代表情緒越正面，反之代表月負面，arousal 則代表興奮程度，越高表示越興奮，反之則是越冷靜。



- 所有情緒都可以落在這個平面，像是快樂可能落在(High arousal, High Valence)這各象限

接下來要來介紹我使用的 dataset，名稱是 CASE (Continuously Annotated Signals of Emotion) ，找來了 30 位受測者，每位受測者給他看 12 部影片，受測者在觀看影片的同時，透過一個 joystick 記錄自己當下的情緒。

Fig. 1 The typical experiment setup shows a participant watching a video and annotating using JERI. The central figure shows the video-playback window with the embedded annotation interface. The right-most figure shows the annotation interface in detail, where Self-Assessment Manikin that were added to the valence and arousal axes can be seen.

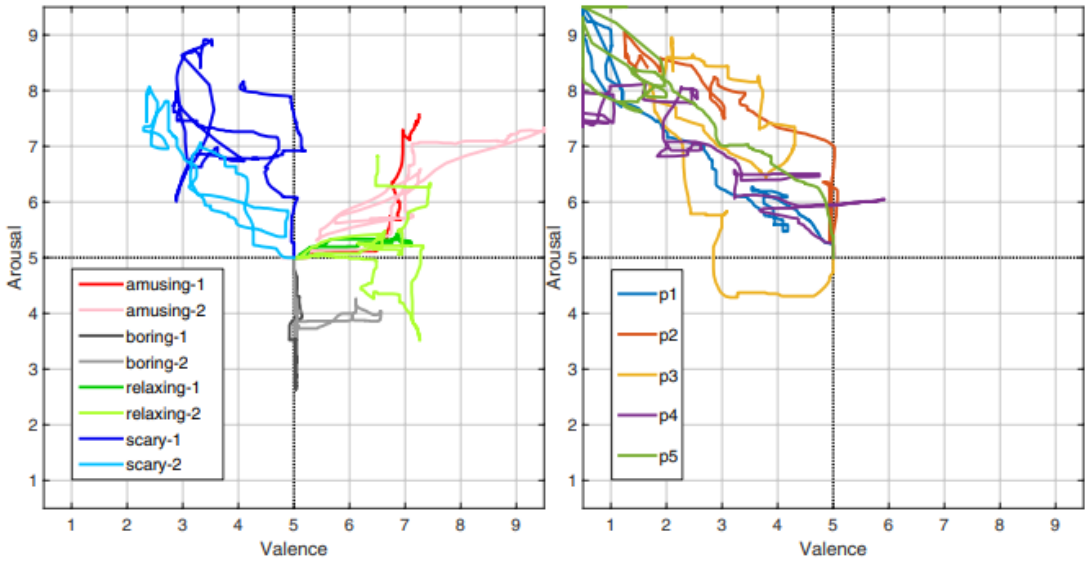


Fig. 2 The plot on the left shows the annotations from one participant for the different videos (see Table 1) in the experiment. The annotations for the 'scary-2' video by the first five participants (labelled as p1–p5) can be seen in the plot on the right.

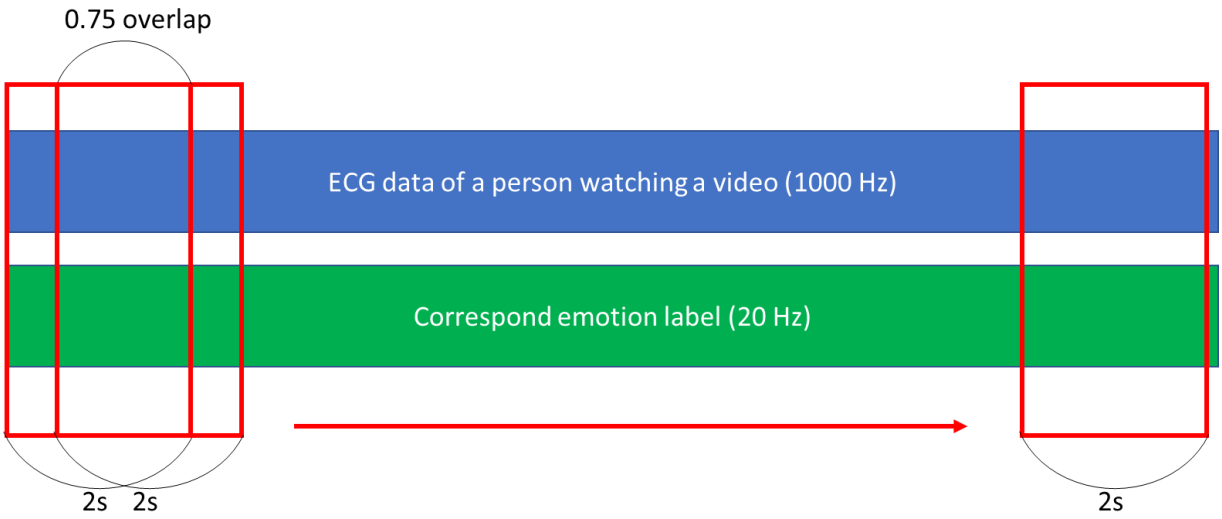
Source	Video-Label	Video-ID	Intended Attributes		Dur. [s]
			Valence	Arousal	
Hangover	amusing-1	1	med/high	med/high	185
When Harry Met Sally	amusing-2	2	med/high	med/high	173
European Travel Skills	boring-1	3	low	low	119
Matcha: The way of Tea	boring-2	4	low	low	160
Relaxing Music with Beach	relaxing-1	5	med/high	low	145
Natural World: Zambezi	relaxing-2	6	med/high	low	147
Shutter	scary-1	7	low	high	197
Mama	scary-2	8	low	high	144
Great Barrier Reef	startVid	10	—	—	101
Blue screen with end credits	endVid	12	—	—	120
Blue screen	bluVid	11	—	—	120

Table 1. The source, label, ID used, intended valence-arousal attributes and the duration of the videos used for the dataset.

Preprocess

接著是資料前處理的部分，將原始資料每兩秒切成一個 clip，相鄰的 clip 會

有 75%的重疊，原始資料中 ECG 的部分是 1000Hz，所以 ECG 每個 clip 長度為 2000，情緒 label 的部分是 20Hz，也就是情緒 label 每個 clip 會有兩個(valence 及 arousal)各 40 個值 array。



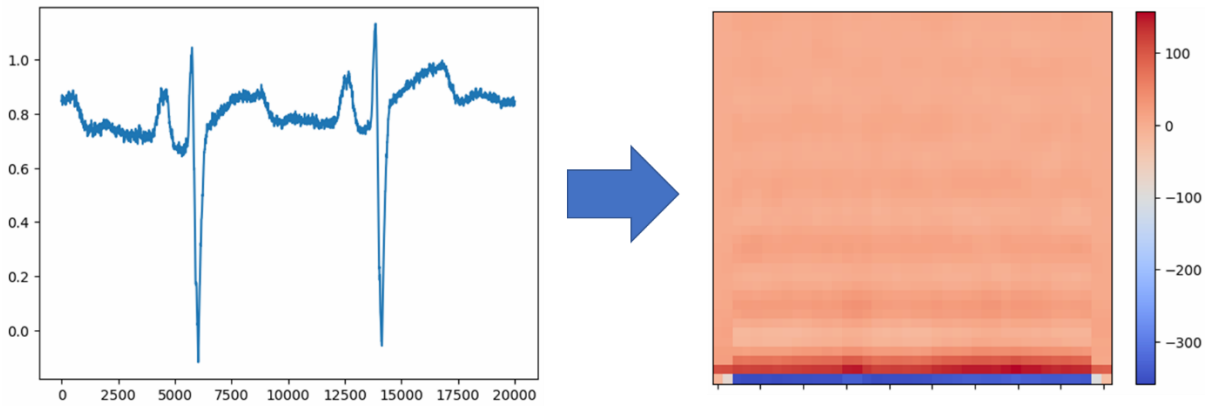
➢ 資料切割過程

接著將每個情緒 label clip 的 40 個值取平均，再根據這個值大小 label 到個類別中，0~3: Low class, 3~6: Medium class, 6~9: High class。

[0, 3)	[3, 6)	[6, 9)
low	Medium	High

➢ 3 class classification

最後將每個 ECG clip 先從長度 2000 resample 到 20000，再轉成 MFCC，最後的大小為(40,40)。



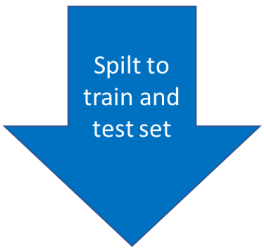
➢ 左圖為原始語音訊號，右圖為轉成 MFCC 的結果

因為 valence 及 arousal 結果差不多，後面就只會呈現 valence 的結果，下圖中上面的表格是 valence 各個 class 的數量，明顯的看出 Medium 這個 class 比另外兩個 class 的數量來的多不少，如果按照等比例將資料分成 training 及 testing

set 的話，最後在做 testing 時，可能會因為 Medium 數量太多，大部分都預測成 Medium，導致 testing accuracy 變得很高，並不能呈現實際的狀況，所以我先將數量最少的類別，也就是 low class，取一伴的數量當作 testing set，另外兩個類別也隨機取相同數量的資料當作 testing set，剩下的作為 training set。

low	Medium	High
7473	114421	25106

➢ Distribution of valence



low	Medium	High
3737	110685	21370

➢ Training set

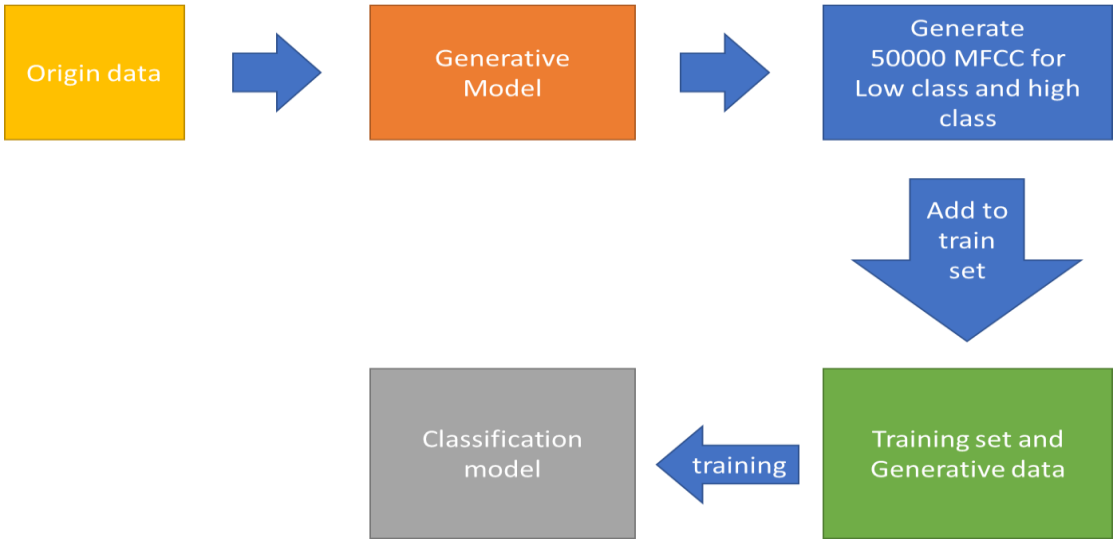
low	Medium	High
3736	3736	3736

➢ Testing set

➢ 上面的表格為整個資料各個類別的數量，下面則分別是 training 及 testing set 各類別的數量

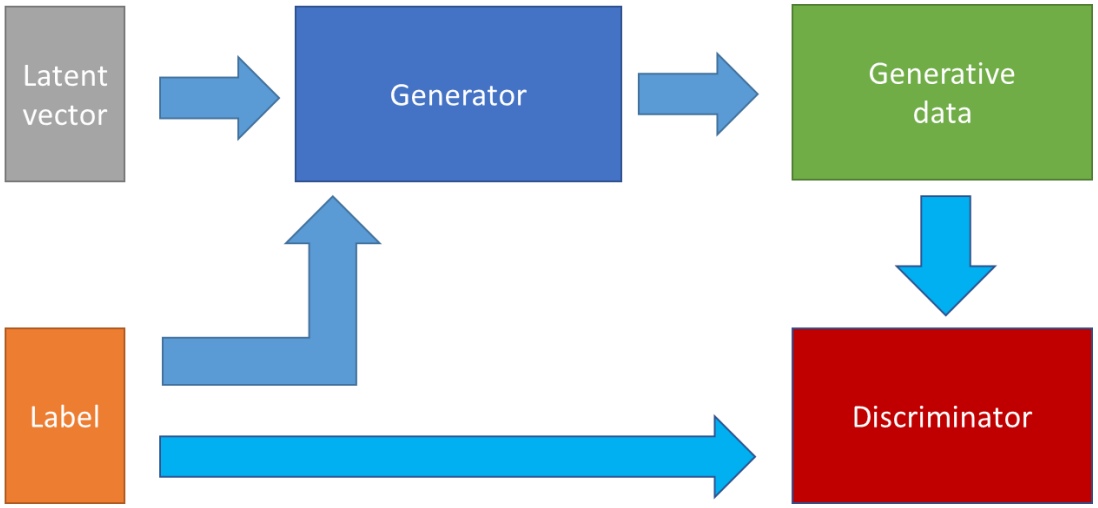
Experiment

接著要來介紹實驗的部分，前面有提到資料有嚴重的 unbalance 的問題，所以我希望能透過 Generative Model，產生數量較少類別(low class 及 high class)的資料，每個類別各產生 50000 筆資料，這邊是直接產生 MFCC，而非原始語音訊號，最後加入到 training data 中，希望能夠提升 performance。



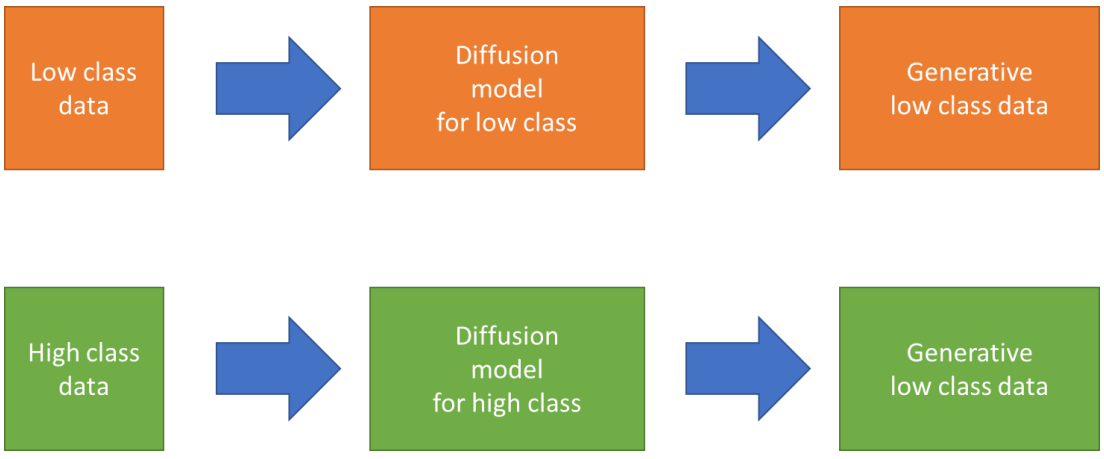
➢ 實驗流程圖

Generative Model 我做了兩種 Model 的嘗試，第一種是 Condition GAN，同時 input 一個隨機的 latent vector (這邊設定大小為 100)以及 label 到 Generator 中，Generator 就會產生指定類別的資料，接著將產生的資料及 label input 到 discriminator 中，就能判斷產生的資料與真實資料是否有差異，epoch 數設為 20。



➢ Condition GAN 架構

第二種 Generative Model 使用的是在作業三使用過的 diffusion model，分別針對 low class 及 high class 建立獨立的 diffusion model，各自 training，最後產生對應類別的資料，epoch 及 diffusion step 分別為 10 以及 800。



➢ Diffusion model 流程圖

最後是做分類的 model，在這邊並沒有使用複雜的 model，而是使用簡單的 CNN，總共四層 2D convolution layer，接上 Flatten layer 將三維資料轉成一維，最後接上 dimension 為 3 的全連接層，epoch 數及 batch size 分別為 30 及 32。

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 40, 40, 32)	224
conv2d_1 (Conv2D)	(None, 40, 40, 64)	12352
conv2d_2 (Conv2D)	(None, 40, 40, 32)	12320
conv2d_3 (Conv2D)	(None, 40, 40, 64)	12352
flatten (Flatten)	(None, 102400)	0
dense (Dense)	(None, 3)	307203

Total params: 344,451
Trainable params: 344,451
Non-trainable params: 0

- Classification model 架構

Result

首先是沒加 generative data 到 training set 的結果，total accuracy 為 0.42，根據下圖這個 confusion matrix，可以看到不管是哪個類別大部分都預測成 medium 這個類別。

		Predict		
		Low	Medium	High
Label	Low	499	2726	521
	Medium	40	3362	334
	High	71	2804	861

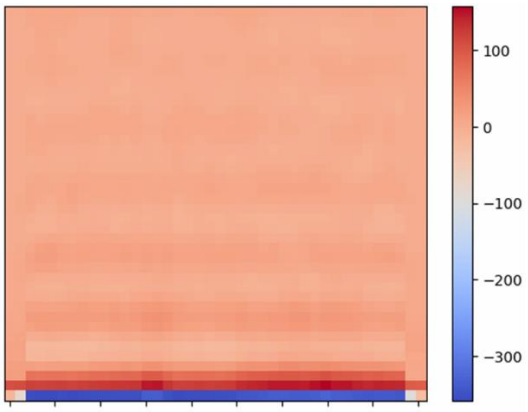
- Low accuracy: 0.13
- Medium accuracy: 0.89
- High accuracy: 0.22

- Confusion matrix

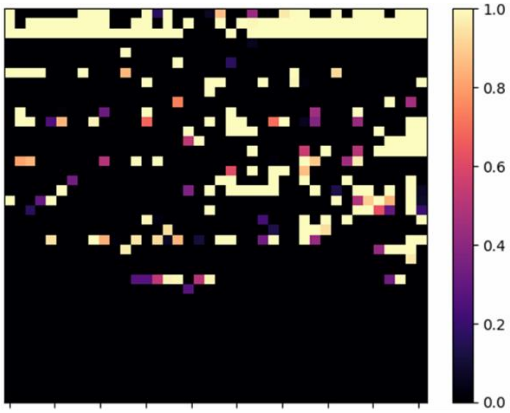
Accuracy: 0.42

- 沒加上 generative data 的分類結果

接著是 Condition GAN 的部分，首先來看 generative data，下面左邊的圖為原始資料，右邊是 model 產生出的資料，很明顯兩張圖差異很大，再將產生的各類別的資料各自跟原始各對應的類別計算 FID，low class 為 343.834，high class 為 320.356。



➢ Original data



➢ Generative data

FID : 343.834(low class), 320.356(high class)

➢ Condition GAN 產生的資料與原始資料的比較，以及各類別的 FID

接著是將 Condition GAN 產生的資料加到 training data 中丟入 model 分類的結果，可以看出來一樣是大部分都預測到 Medium class，total accuracy 為 0.41，並沒有什麼改善。

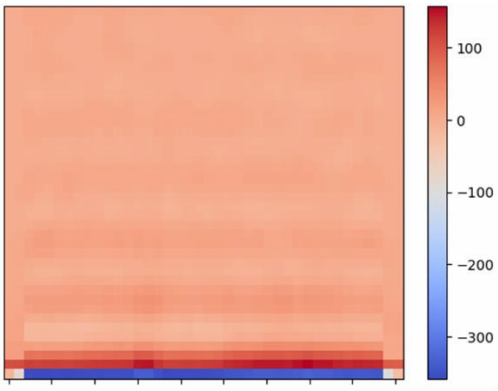
		Predict			
		Low	Medium	High	
Label	Low	404	2838	494	➢ Low accuracy: 0.10
	Medium	17	3542	177	➢ Medium accuracy: 0.94
	High	40	3045	651	➢ High accuracy: 0.17

➢ Confusion matrix

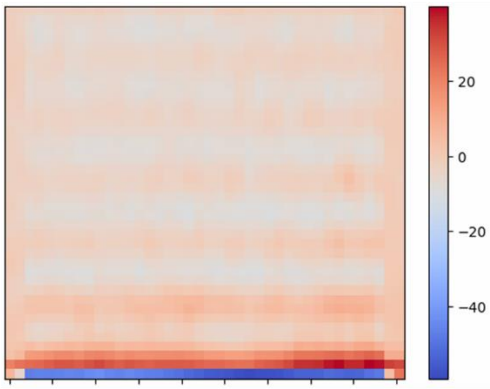
Accuracy: 0.41

➢ 加上 Condition GAN generative data 的分類結果

最後是 diffusion model 的部分，一樣先來看產生出來的資料，下面左圖為原始資料，右圖為產生出來的資料，明顯的比 Condition GAN 好的許多，最後依樣將產生出來的各類別資料各自跟原始各對應的類別計算 FID，low class 為值 56.564，high class 為 104.039，也是比 Condition GAN 來的好。



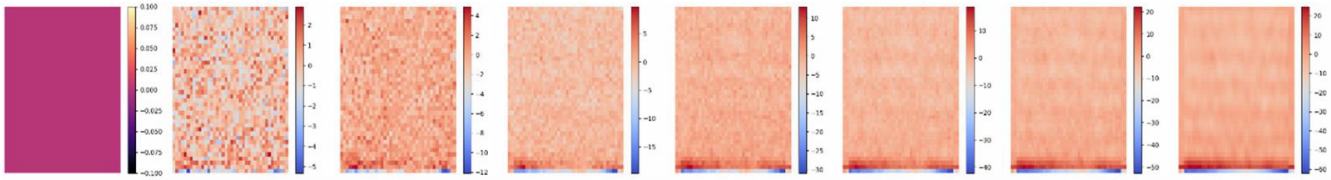
➢ Original data



➢ Generative data

FID : 56.564(low class), 104.039(high class)

➢ Diffusion model 產生的資料與原始資料的比較，以及各類別的 FID



➢ Diffusion model 產生的資料的過程

再來是將 diffusion model 產生的資料加到 training set 丟到 model 分類的結果，看的出來儘管 diffusion model 產生出來的資料看似不錯，但最後並每有對 performance 有顯著的提升，total accuracy 只有 0.42，還是非常容易預測成 Medium 這個類別。

		Predict		
		Low	Medium	High
Label	Low	545	2770	421
	Medium	43	3478	215
	High	70	2937	729

➢ Low accuracy: 0.14

➢ Medium accuracy: 0.92

➢ High accuracy: 0.19

➢ Confusion matrix

Accuracy: 0.42

➢ 加上 diffusion model generative data 的分類結果

	Low accuracy	Medium accuracy	High accuracy	Total accuracy
Without generative model	0.13	0.89	0.22	0.42
With Condition GAN	0.10	0.94	0.17	0.41
With Diffusion model	0.14	0.92	0.19	0.42

- 所有結果的比較

Conclusion

最後對結果不理想做一些分析，第一，可能不該直接透過 generative model 產生 MFCC，而是產生原始語音訊號，第二，要設計一個更 robust 的 generative model，產生的資料對分類結果才會有顯著的影響，第三，做分類的 model 太弱，應該使用像是 ResNet 等較為複雜的 model，而不是簡單的 CNN，第四，generative model 產生出來的資料跟各類別的相關性可能不大，第五，原始資料本身就有問題，像是各 ECG 跟對應的類別相關性不大，儘管 generative model 產生的資料跟原始資料很像，但因為跟各類別相關性不大，所以對分類結果每有太大的影響。