

Machine learning final project

我覺得未來20年AI可能可以做到全自主的交通網路，AI將整合車輛、基礎設施和數據系統，形成一個無縫、零事故的移動生態。具體來說是，人們可以將自己的所有行程及目的地跟AI說，讓AI可以排程、規劃所有人的路線，加上未來自駕車或是無人大眾交通工具的普及，分配所有人任何時刻的移動方式，不僅可以有效避免塞車所導致的時間浪費，也大大降低了，發生事故的可能性。而這次我想把問題聚焦每個路口的車流預測。我認為對AI來說預測未來路線的車流是必須的，才能判斷是否需要提前分流或主動讓道等等(若有緊急情況像救護車等)，所以想先嘗試做路口的車流分析。

首先資料部分為達到城市級的交通預測，需要有多模態交通資料，例如：即時車流 / 速率 (traffic speed/flow)、路網拓樸 (graph structure)、事件資料 (事故、施工、壅塞區)、天氣、多日作息模式、影像 / CCTV 等等。並且要有大規模資料量，畢竟若城市級 GNN 通常要 數千～數萬 nodes(就是senser)，而且為了捕捉季節性，時間序列要至少連續一年以上。

接著工具的部分，由於交通路口車流量屬於時間空間都需要考慮的資料，空間的部分我們採用Graph Neural Network，研究路口間的相連情況，畢竟一個路口和鄰近路口的車流肯定是息息相關。而時間部分我們使用LSTM來解決，這邊也是我們toy model會嘗試實作的內容，而未來若要進一步發展則可以加入強化學習來訓練機器控制燈號變化。而因為是利用已知每分鐘流量利用預測下一步流量，GGN、LSTM的部分都是監督式學習，而未來若要加入交通號誌控制則需要加入 Reinforcement learning。

Toy model

(1) traffic-model-1 (MLP)

Data : (code 亂數生成)

input: vector: (x_1, x_2, x_3, x_4)

x_1 : 車流

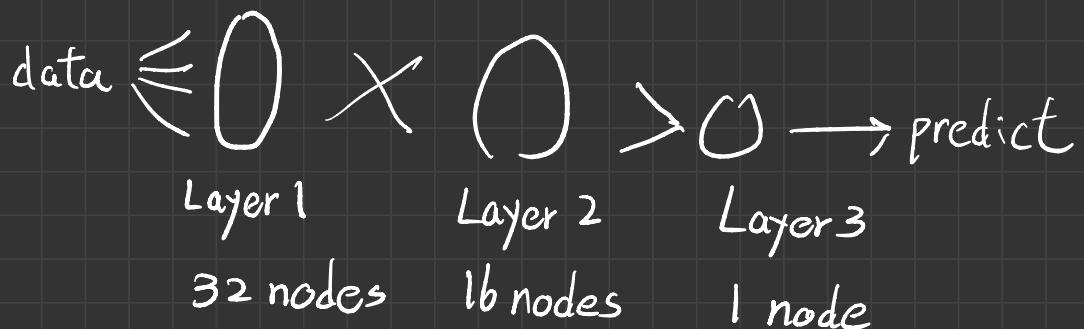
x_2 : 天氣 (0: sunny, 1: rainy)

x_3 : 時段 (0: 早, 1: 中, 2: 晚, 3: 尖峰)

x_4 : 事故與否 (0: 無, 1: 有)

output: 下一分鐘車流量

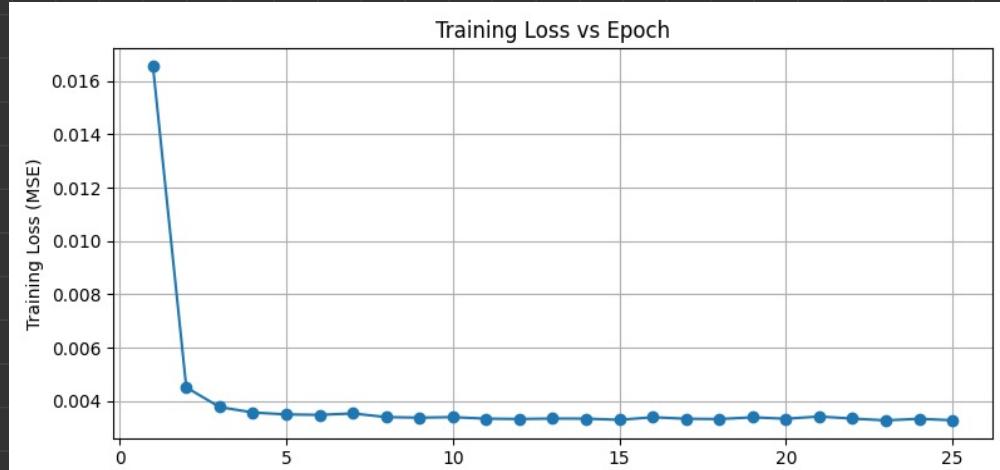
Model = 25 epochs,



Loss : MSE

Result :

```
Epoch 24/25
63/63 0s 1ms/step - loss: 0.0033
Epoch 25/25
63/63 0s 1ms/step - loss: 0.0033
Saved loss plot to: training_loss_final213.png
1/1 0s 58ms/step
預測下一分鐘車流量 (normalized): 8.796595
```



(2) traffic_model_2 (GNN)

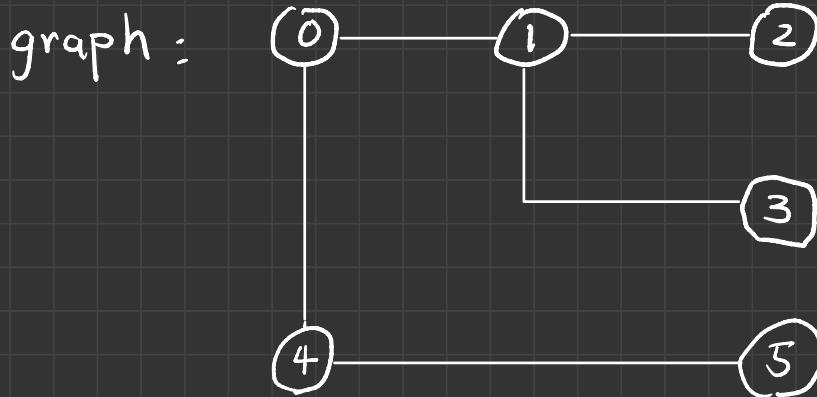
Data : (code 隨機生成)

input: 各路口 vector : (π_1, π_2, π_3)

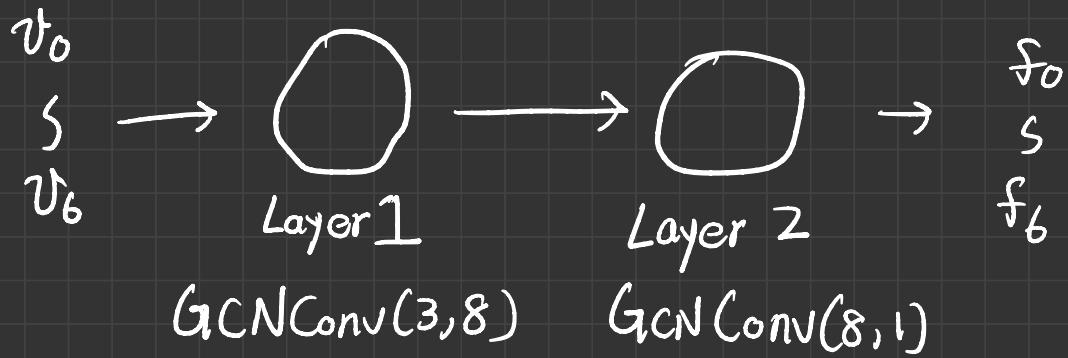
π_1 : 車流 , π_2 : 天氣 , π_3 : 事故

output = ($f_0 \sim f_5$)

↑ 各路口下一分鐘車流



Model = 300 epochs



Loss = 每個路口 MSE 平均值

Result =

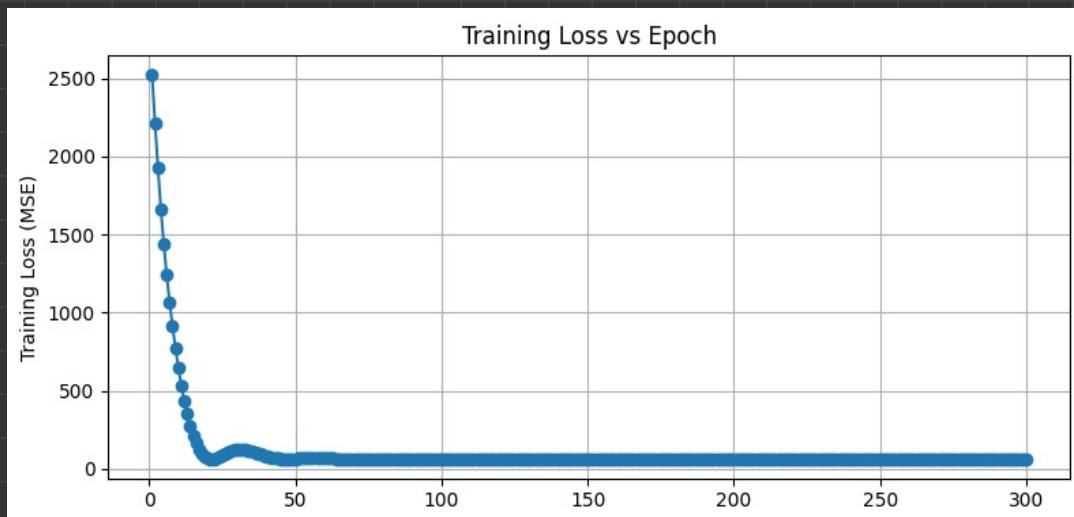
Epoch 290 - loss: 62.4193

Epoch 300 - loss: 62.3370

Final training loss: 62.3286

真實下一分鐘: tensor([44.0736, 50.1472, 17.6061, 49.2244, 36.2757, 25.9561])

GNN 預測值: tensor([41.9154, 47.3220, 31.0581, 37.0047, 38.8848, 30.8801])



(3) traffic_model_3 (GNN + LSTM)

Data = (code 隨機生成)

input : 各路口 過去 10 時間段 資料

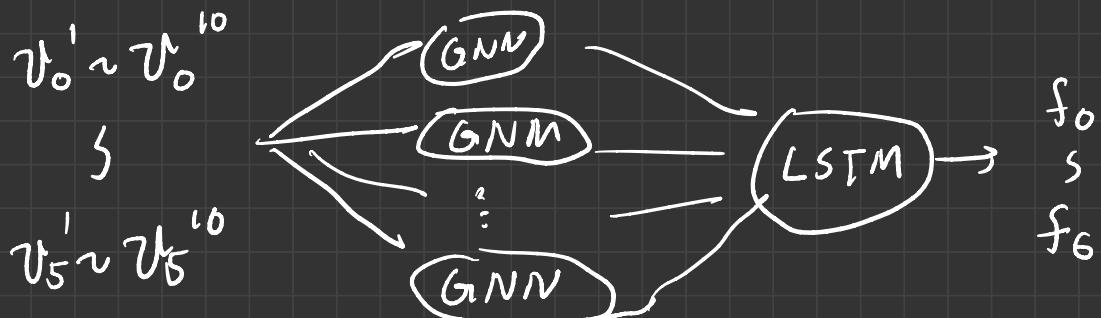
$$V = (x_1, x_2, x_3),$$

x_1 = 車流 , x_2 = 天氣 . x_3 = 事故

output = ($f_0 \sim f_5$) , 各路口下一分鐘 車流

graph = same as model_2

Model : 300 epoches



Loss : 各路口 MSE 平均

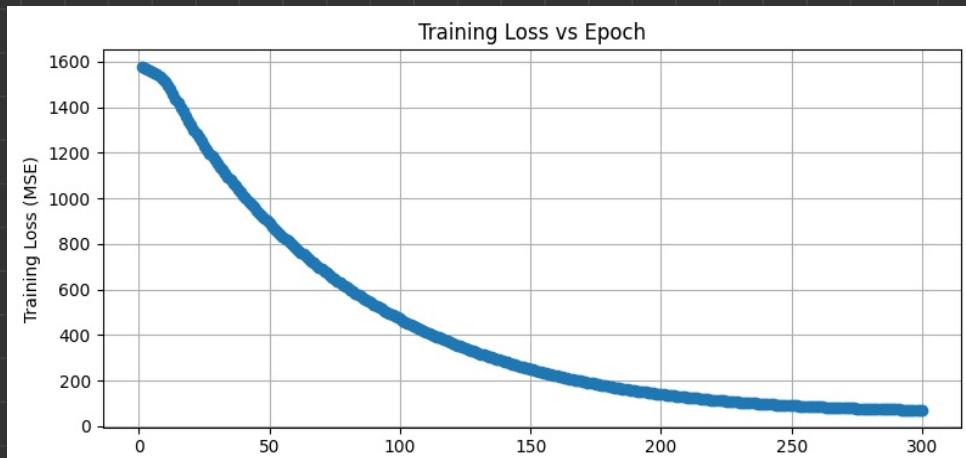
Result:

Epoch 290 - loss: 72.702850

Epoch 300 - loss: 70.150230

真實下一步流量: tensor([48.1186, 49.2769, 28.5947, 41.1518, 31.6412, 36.5022])

模型預測流量: tensor([36.0752, 36.0752, 36.0752, 36.0752, 36.0752, 36.0752])



Conclusion :

這次toy model的實作蠻好玩的，一步步從僅有純數據資料的預測，漸漸有了空間的概念，再來加上時間維度，慢慢刻畫出想做到的模型。雖然加上時空間之後，loss都不是很漂亮，總有幾個路口的預測會差很多，不知道試模型沒有設計好還是這個問題本身不好做，看來離精準預測來有一段距離呢！除了模型之外，我也發現了為什麼全自動交通沒人有要做，光是數以億計的路口都要安裝sensor收集資料以外，還需要保存一年以上的資料來預測季節的變化，這個資料量跟成本想想就令人卻步。