

國立台北大學資訊工程學系專題報告

使用深度學習預測股票趨勢並決定交易策略

專題編號:PRJ-NTPUCSIE-109-004

執行期間:109 年 9 月 至 110 年 6 月

一、摘要

我們使用深度學習的模型，來預測股價的漲跌幅趨勢，並根據模型結果制定交易策略，最後透過結果評估驗證使用深度學習的方式是否能實際在股票市場中獲取高額報酬。

在深度學習模型的部分我們選擇了三個不同的模型，依序分別是 CNN(Convolutional Neural Network, 卷積神經網路)、LSTM(Long Short-Term Memory, 長短期記憶模型)以及 Transformer Model(基於自注意力機制的 Encoder-Decoder 模型)。

我們使用 2010~2019 年的歷史股價資訊數據做為訓練資料並以 2020 年的股價資訊做為測試資料，進而得出我們的準確率以及損失函數，再藉由股價回測的結果以及相關金融指標判斷來驗證此模型是否可以被接受，以及模型的執行成效和獲利能力。

二、簡介

1. 研製背景

全球化的低利率環境所帶來的影響與衝擊，讓存款利率低到早已被通貨膨脹率所侵蝕。簡而言之，錢存入銀行中只會不斷貶值，能購買的東西只會越來越少。

因此，比起將多餘的資金放進銀

行中作為存款，大眾尋求將多餘的資金投資於金融商品。而股票交易是目前大眾投資最普及的方式，每日股市的漲跌幅也一直都是投資人所關心的焦點。

而許多專業經理人、股票分析師、投資人等等都在找尋如何準確地預測股價，並建構出最佳的交易策略，追求高報酬但是低風險的獲利模式。

因此，我們想嘗試使用深度學習的模型，來預測未來股價的漲跌幅度，並決定我們的交易策略。透過最後的獲利評估，驗證深度學習的模型是否真的能實際在股票市場中獲取高額的報酬。

2. 相關研究文獻探討

深度學習中神經網路的概念被廣泛應用於金融市場及金融商品的預測，並在財務金融的市場佔有一席之地。

吳哲偉(2017)使用卷積神經網路模型 Convolutional Neural Network(CNN)，以台灣 50ETF 作為研究對象，加入股票的 K 線圖(開盤價、最高價、最低價、成交量)和總體經濟(美國道瓊指數、那斯達克指數、S&P500)當作輸入變數，輸出交易訊號，尋找最佳交易策略。許琮苓(2019)使用時間

序列神經網路 Long Short-Term Memory (LSTM)預測台灣四家上市公司，輸入變數為 5 個基本資料(開盤價、最高價、最低價、收盤價、交易量)和 2 個金融指標(相對強弱指標、指數平均數指標、)以及代表新聞關注度的 Google 趨勢，評估此模型的預測準確度。夏鶴芸(2020)使用新深度學習技術 BiLSTM 和 Transformer 預測股價走勢，並以自然語言處理模型 BERT 進行新聞的情緒分類。研究模型如何搭配能使台積電的股價走勢預測成效最好。

3. 目標

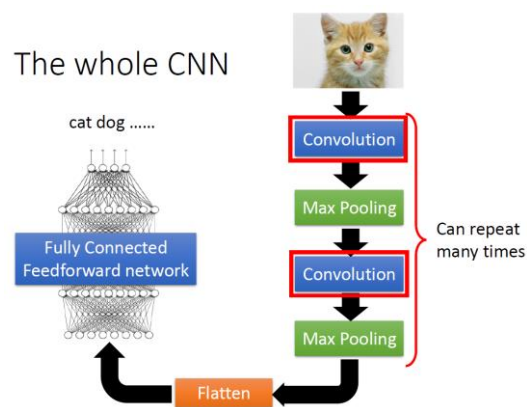
能夠做出一個獲利優於大盤績效的投資模型，並能夠讓使用者選擇任意台灣上市公司之股票，透過此模型的預測結果來判斷是否要投資該股票？若是投資該股票的預期獲利會是多少？可能需要承擔的風險和損失又是多少？

三、專題進行方式

1. CNN 模型

卷積神經網路 (Convolutional Neural Network, CNN)是一種前饋神經網路，常用來處理影像辨識相關內容的任務。CNN 比起一般的神經網路 (fully connected)，透過參數共享以及權值共享 (weight shared) 的方式，可以降低使用的參數數量。

卷積神經網路架構是由卷積層、池化層和全連接層所構成。



圖一、CNN 架構圖

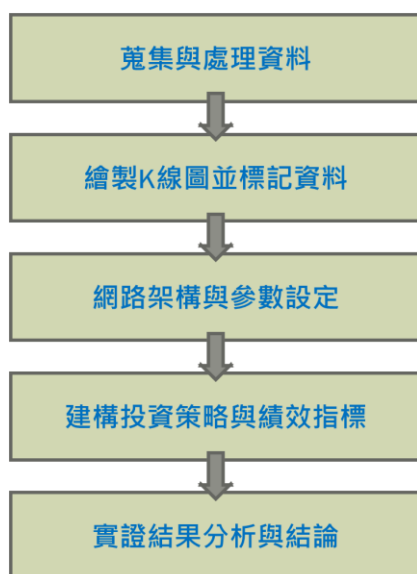
資料來源:(李弘毅，ML Lecture 10: Convolutional Neural Network)

卷積層 (Convolution layer)，透過卷積核 (kernel、filter) 的滑動對圖像進行特徵提取，有多少的 filter 就會得到多少的 Feature Map。

池化層 (Pooling layer)，可以縮小輸入圖片的大小。最大池化 (Max Pooling) 是最常使用的池化方式。將圖片卷積過後得到的 Feature Map 切割成多個矩形區塊，在每一個區塊中選取最大值保留下來，其餘值捨棄，得到一個較原圖小的新圖，此圖的維度即為 filter 的數量。

全連接層 (Fully connected layer)，池化過後經由攤平 (Flatten)，將數值丟入全連接層，最後得到輸出結果。

1.1 CNN 流程圖



圖二、CNN 流程圖

1.2 CNN 輸入變數挑選

我們選擇了台灣 50 的成分股將 50 家公司股票的開盤價、收盤價、最高價、最低價、調整後收盤價及成交量的資料，作成以 20 日為交易區間繪製的 K 線圖。

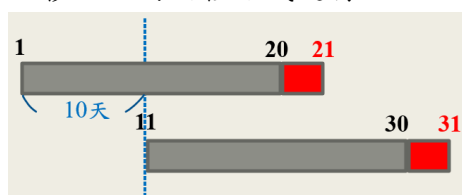
資料時間從 2013 年~2020 年，分成 training data(2013 年~2017 年)、valid data(2018 年~2019 年)和 testing data(2020 年)。

1.3 CNN 資料前處理

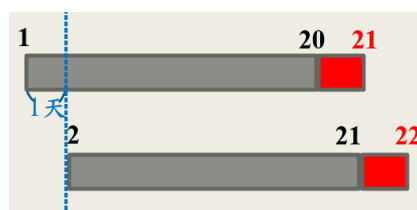
修改 K 線圖的圖形，讓圖形縮小至符合我們模型設定的大小 (287*287)。

我們設定以 20 日為一個單位，預測第 21 日收盤價的趨勢。

分成每次位移 10 天以及每次位移 1 天的兩種方式進行。



圖三、每次位移 10 天



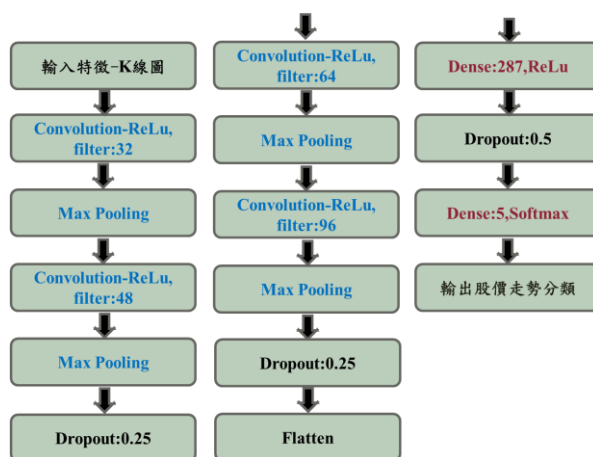
圖四、每次位移 1 天

以 5%作為分界，分別標籤：大漲(5%~)、小漲(0%~5%)、不變(0%)、小跌(0%~(-5%))、大跌(-5%~)。

把圖片跟標籤分開。原先是將 K 線圖儲存位置和標籤一起記錄在 txt 文字檔中，所以在丟入模型前，需要將兩者分開成兩個 list 來儲存。

1.4 CNN 網路架構與參數設定

網路架構的部分，我們使用了 4 層卷積層，4 層池化層(最大池化)。



圖三、CNN 網路圖

而其他重要的參數設定如下：

- Kernel Size: 3*3
- Stride = 1
- Pool Size: 2*2
- Batch size: 20

- Epochs: 20
- 損失函數:
categorical_crossentropy
- Optimizer: Adam

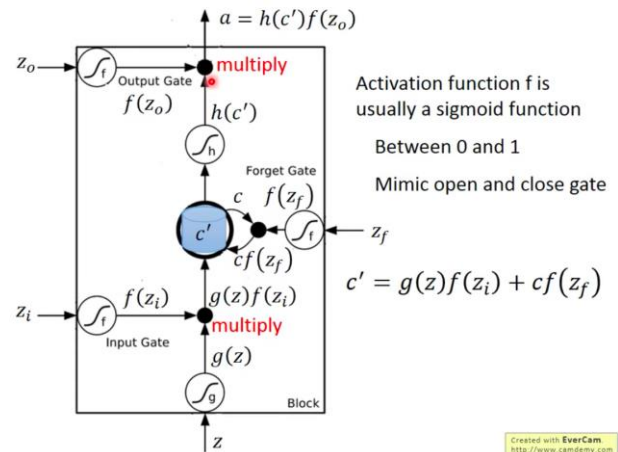
1.5 CNN 預測結果

發生資料過擬合(overfitting)的情況，valid data 的損失函數(loss function)數值趨近 2，loss 太高；並且 valid data 以及 testing data 的準確率(Accuracy)為 0.4，準確率太低。經過調整輸入變數以及模型參數等多方嘗試仍無法改善，故我們不繼續進行後續的建構投資策略以及績效評估，改換為 LSTM 的模型，繼續進行股票預測的嘗試。

2. LSTM 模型

長短期記憶模型 (Long Short-Term Memory, LSTM)，是遞歸神經網路(RNN)的一種，主要是用來解決時間序列相關的模型。

LSTM 架構是由四個部分組成，分別是輸入門(Input gate)、輸出門(Output gate)、遺忘門(Forget gate)以及儲存單元 (Memory cell)。LSTM 需要四個輸入值(1 個欲存入 memory cell 的值以及 3 個控制不同 gate 的值)，才能產生一個輸出值。



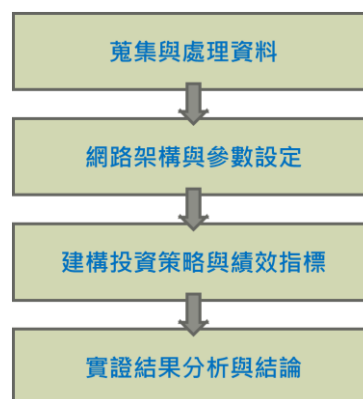
圖四、LSTM 模型圖

資料來源：(李弘毅，ML Lecture 21-1: Recurrent Neural Network (Part I))

- z : 要被存進 cell 的 input value
- z_i : 控制 input gate 的 signal
- z_o : 控制 output gate 的 signal
- z_f : 控制 forget gate 的 signal
- a : output

f 激勵函數通常用 sigmoid function 因為 sigmoid function 值介在 0~1 之間，代表 gate 被打開的程度。Output:1 表示 gate 打開；反之，output:0 表示 gate 關閉。

3.1 LSTM 流程圖



圖五、LSTM 流程圖

3.2 LSTM 輸入變數挑選

我們選擇了台灣 50 的成分股將 50 家公司股票的開盤價、收盤價、最高價、最低價、調整後收盤價及成交量；金融指標：指數平均數指標(EMA)；籌碼面指標：trading money、spread、trading turnover。

資料時間從 2010 年~2020 年，分成 training data 和 valid data (2010 年~2019 年)、testing data(2020 年~2021 年 5 月 4 日)。

3.3 LSTM 資料前處理

因為時間序列的資料不能以 shuffle 的方式處理，否則會喪失時間序列的意義。故資料是以滾動式視窗(sliding windows)的方式處理。

再將這些數據全部都進行正規化，使範圍縮小至-1~1。

我們設定以 20 日為一個單位，預測第 21 日收盤價的趨勢。分成每次位移 10 天以及每次位移 1 天的兩種方式進行。

以 5%作為分界，分別標籤：大漲(5%~)、小漲(0%~5%)、不變(0%)、小跌(0%~-5%)、大跌(-5%~)。

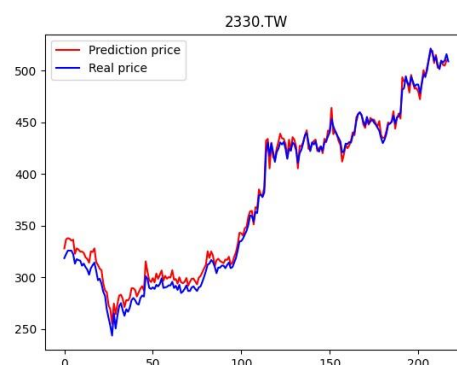
3.4 LSTM 網路架構與參數設定

使用單層的 (many to one)LSTM 模型：

- 損失函數:MSE
- 優化函數:Adam

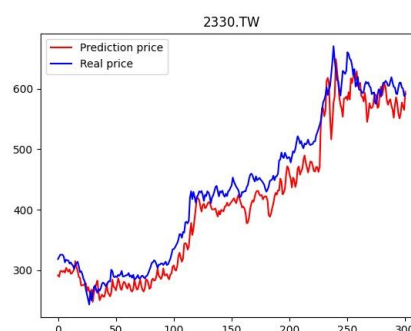
3.5 LSTM 預測結果

(1) 一次位移十天準確率：56%



圖六、LSTM 模型 2020 年預測價格與實際價格折線圖(一次位移十天)

(2) 一次位移一天準確率：47%



圖七、LSTM 模型 2020 年預測價格與實際價格折線圖(一次位移一天)

根據圖六和圖七可以發現 LSTM 模型一次位移十天的準確率較一次位移一天的成效更好，因此在後續的交易策略建構中，我們會選擇使用 LSTM 一次位移十天的模型來進行。

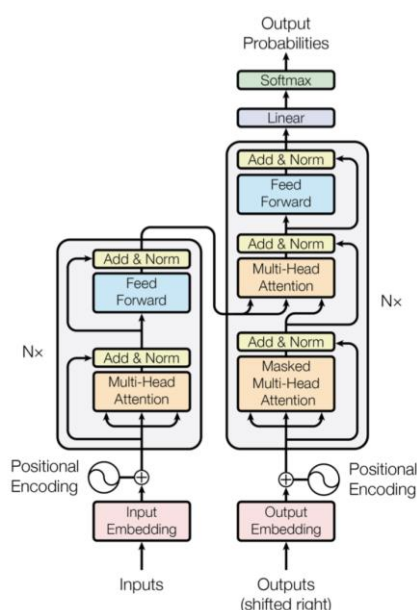
3. Transformer 模型

Transformer Model(基於自注意力機制的 Encoder-Decoder 模

型)，此模型廣泛應用於自然語言處理(NLP)的領域中。

Transformer Model 直觀來看是由 Encoder 和 Decoder 所構成。

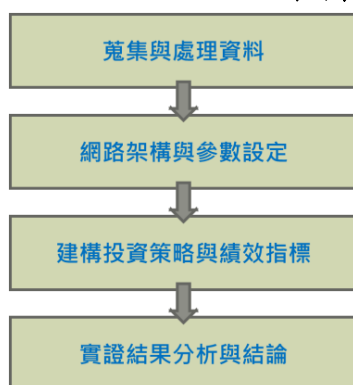
其中在 Encoder 和 Decoder 中有自注意力機制(Self-Attention mechanism)，此機制可以改善 RNN 無法進行平行運算的缺點；亦可以同時處理時間序列和平行運算。



圖八、Transformer 模型圖

資料來源：(Vaswani, Shazzer et al. (2017).Attention Is All You Need.)

3.1 Transformer 流程圖



圖九、Transformer 流程圖

3.2 Transformer 輸入變數挑選

我們選擇了台灣 50 的成分股將 50 家公司股票的開盤價、收盤價、最高價、最低價、調整後收盤價及成交量；金融指標:指數平均數指標(EMA)；籌碼面指標:trading money、spread、trading turnover。

資料時間從 2010 年~2020 年，分成 training data 和 valid data (2010 年~2019 年)、testing data(2020 年~2021 年 5 月 4 日)。

3.3 Transformer 資料前處理

以滾動式視窗(sliding windows)的方式處理資料。

將這些數據全部都進行正規化，使範圍縮小至-1~1。

我們設定以 20 日為一個單位，預測第 21 日收盤價的趨勢。

分成每次位移 10 天以及每次位移 1 天的兩種方式進行。

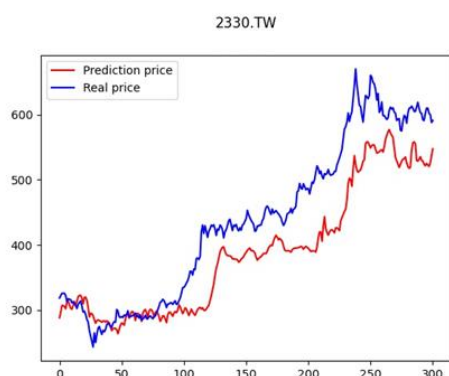
標籤分成三種:漲、不變、跌。

3.4 Transformer 網路架構與參數設定

- Sequence length: 20
- Epochs: 10000
- Batch size: 256
- 損失函數:MSE
- 優化函數:Adam

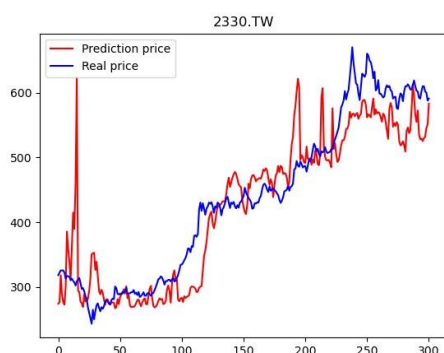
3.5 Transformer 預測結果

(1) 一次位移十天準確率：52%



圖十、Transformer 模型 2020 年預測價格與實際價格折線圖(一次位移十天)

(2) 一次位移一天準確率：47%



圖十一、Transformer 模型 2020 年預測價格與實際價格折線圖(一次位移一天)

根據圖十和圖十一可以發現 Transformer 模型一次位移十天的準確率較一次位移一天的成效更好，因此在後續的交易策略建構中，我們會選擇使用 Transformer 一次位移十天的模型來進行。

四、主要成果與評估

在交易策略的部份，我們採取的是個股分開買賣的交易方式。我們的

原始本金設定為 1000 萬；手續費買賣時皆須收取，為交易金額之 0.1425%；股票交易稅為交易金額之 0.3%，僅賣出時收取。

分別使用 LSTM 與 Transformer 的一次位移十天模型來進行真實台股市場交易策略的模擬。

績效評估的部分，我們以簡單的最後持有現金以及報酬率來呈現我們交易策略的成果。

1. LSTM 模型-一次位移十天

以台積電為例(2330.TW): 交易期間為 2020 年 2 月 10 日~2021 年 5 月 4 日。若台積電的隔一日的預測報酬超過今日 3%(上漲)，我們就買進；若隔一日的預測報酬低過今日 -3%(下跌)，我們就賣出。

買進的金額設定為持有現金的一定比例；賣出的數量(張=1000 股)為手中持有股票張數的一半。

表一、買進金額設定與報酬率的關係-LSTM Model

買進金額設定	最後持有現金(元)	報酬率(%)
10%	16159077.8	61.59
20%	19468986.3	94.69
30%	21039382.8	110.39
40%	21834332.8	118.34
50%	22362856.5	123.63
60%	22629680.1	126.30
70%	22845972.4	128.46
80%	22872357.1	128.72
90%	23063952.2	130.64
100%	23069807.7	130.70

根據表一我們發現此交易策略獲

得的報酬率和買進金額比例的設定呈現正比關係。買進金額比例越高，報酬率也越高。

2. Transformer 模型-一次位移十天

以台積電為例(2330.TW): 交易期間為 2020 年 2 月 10 日~2021 年 5 月 4 日若台積電的隔一日的預測報酬超過今日 3%(上漲)，我們就買進；若隔一日的預測報酬低過今日 -3%(下跌)，我們就賣出。

買進的金額設定為持有現金的一定比例；賣出的數量(張=1000 股)為手中持有股票張數的一半。

表二、買進金額設定與報酬率的關係-Transformer Model

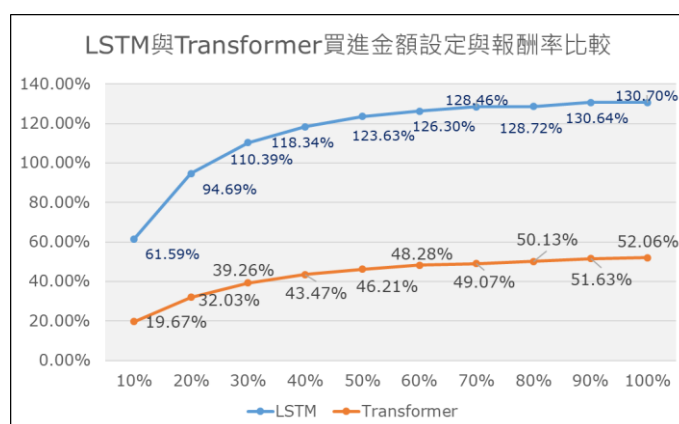
買進金額設定	最後持有現金(元)	報酬率(%)
10%	11966533.7	19.67
20%	13203401.2	32.03
30%	13925664.9	39.26
40%	14346892.8	43.47
50%	14621008.0	46.21
60%	14827503.0	48.28
70%	14906881.6	49.07
80%	15012943.9	50.13
90%	15162649.6	51.63
100%	15206340.9	52.06

根據表二我們發現此交易策略獲得的報酬率和買進金額比例的設定呈現正比關係。買進金額比例越高，報酬率也越高。

3. 交易策略成果小結

以台積電為例的交易策略結果，我們發現使用 LSTM 的模型比使用

Transformer 的模型，所獲得的報酬率高出不少。且會隨著買進金額比例越高，報酬率越高。



圖十二、LSTM 和 Transformer 模型買進金額設定與報酬率關係之折線圖(一次位移一天)

除了台積電股票以外，我們也對其他的股票進行了交易策略的模擬，發現 LSTM 模型的報酬率幾乎皆優於 Transformer 模型的報酬率。

五、結語與展望

1. 結語

股市的預測與獲利的最佳化，是眾多投資人所關切的議題。本專題使用深度學習的三個不同模型，制定交易策略後所獲得的報酬率，來評估模型的績效。而績效成果可以提供投資人判斷是否要買進或賣出該股票。

2. 未來展望

我們在進行 LSTM 和 Transformer 的模型準確率評估時，從預測趨勢圖中發現若是把預測的結果進行延遲一天，就會發現準確率大幅上升！

我們也發現因為在新冠肺炎疫情的影響之下，導致 2020 年的預測準確率相較一般正常年度的準確率低了不

少。因此之後若是要改善模型可以建議再加入影響總體經濟面的重要指標。

模型的使用部分，我們建議除了純數值的資料以外，還可以再增加對股票相關新聞的情緒分析，使得 Transformer 可以發揮其自然語言處理的強大功能。

交易策略的部分，投資人亦可以選擇透過自行修改交易策略的設定或是變數，優化交易股票的報酬率。

在交易策略評估的部分，可以建議之後若是要改善績效的評估方式，可以加入風險波動程度(σ)，並使用夏普指標(Sharpe ratio，每承擔一單位風險可獲得的報酬率)，這樣可以更客觀地找尋風險低但是報酬率高的交易策略。

六、 銘謝

首先，最先感謝我們的指導教授，在我們專題組員有兩位都是外系雙主修成員的情況下，願意收我們作為專題生。並且在這一年來從我們題目的發想到模型建立的過程中給予我們許多寶貴的建議與方向以及相關的學習資源，幫助我們建立深度學習的基礎概念，才有了我們今天的專題成果。

再來，非常感謝實驗室的學長姐們，從我們一開始接觸到相關的程式環境建置以及起初迷惘找不到方向時，給予我們許多的協助，過程中遇到程式問題時，學長姐也都很願意撥出自己的時間為我們詳細解答。

此外，也要特別感謝資管所的戴敏育教授，在百忙之中也特為我們撥出自己的時間，讓我們詢問專題的

相關問題以及提供寶貴的建議與方向。

最後，要感謝專題的每一位組員，有了大家的互相合作以及對專題的付出，才有我們今天的專題成果。這一年來的努力，大家都辛苦了！

七、 附錄

以下表格是 LSTM 一次位移 10 天模型，對其他間台灣上市公司股票(台積電以外之股票)進行交易策略模擬，得到買入金額設定和報酬率之關係。

表三、中鋼買進金額設定與報酬率的關係-LSTM Model

本金:一千萬	買進金額設定	Final cash	Returns
2002(中鋼)	0.1	13289772.7	32.90%
	0.2	13572628.45	35.73%
	0.3	13571811.88	35.72%
	0.4	13526166.63	35.26%
	0.5	13472376.79	34.72%
	0.6	13423260.91	34.23%
	0.7	13384613.83	33.85%
	0.8	13348592.45	33.49%
	0.9	13309318.2	33.09%
	1	13282270.75	32.82%

表四、中華電信買進金額設定與報酬率的關係-LSTM Model

本金:一千萬	買進金額設定	Final cash	Returns
2412(中華電信)	0.1	10497058.54	4.97%
	0.2	10717868.94	7.18%
	0.3	10802769.68	8.03%
	0.4	10832130.9	8.32%
	0.5	10843975.96	8.44%
	0.6	10844977.39	8.45%
	0.7	10838968.84	8.39%
	0.8	10844292.94	8.44%
	0.9	10837783.68	8.38%
	1	10828270.14	8.28%

表五、玉山金買進金額設定與報酬率的關係-LSTM Model

本金:一千萬	買進金額設定	Final cash	Returns
2884(玉山金)	0.1	11306483.99	13.06%
	0.2	11843090.17	18.43%
	0.3	12029753.53	20.30%
	0.4	12059378.96	20.59%
	0.5	12017650.21	20.18%
	0.6	11935999.51	19.36%
	0.7	11844716.29	18.45%
	0.8	11758439.89	17.58%
	0.9	11687441.08	16.87%
	1	11634203.09	16.34%

表八、中華電信買進金額設定與報酬率的關係-Transformer Model

本金:一千萬	買進金額設定	Final cash	Returns
2412(中華電信)	0.1	10273169.96	2.70%
	0.2	10443249.13	4.40%
	0.3	10537641.92	5.30%
	0.4	10583088.75	5.80%
	0.5	10601137.02	6.00%
	0.6	10607485.95	6.00%
	0.7	10605483.1	6.00%
	0.8	10608827.75	6.00%
	0.9	10603319.91	6.00%
	1	10594807.8	5.90%

表六、台塑買進金額設定與報酬率的關係-LSTM Model

本金:一千萬	買進金額設定	Final cash	Returns
1301(台塑)	0.1	15096591.96	50.97%
	0.2	15951981.15	59.52%
	0.3	15955965.02	59.56%
	0.4	15787503.91	57.88%
	0.5	15568568.29	55.69%
	0.6	15356068.61	53.56%
	0.7	15135536.66	51.36%
	0.8	14960251.67	49.60%
	0.9	14815086.98	48.15%
	1	14691831.22	46.92%

表九、玉山金買進金額設定與報酬率的關係-Transformer Model

本金:一千萬	買進金額設定	Final cash	Returns
2884(玉山金)	0.1	10161406.59	1.60%
	0.2	10322862.98	3.20%
	0.3	10484269.57	4.80%
	0.4	10645825.51	6.40%
	0.5	10807232.11	8%
	0.6	10969036.94	9.60%
	0.7	11130443.54	11.30%
	0.8	11291899.92	12.90%
	0.9	11453455.85	14.50%
	1	11614762.9	16.10%

以下表格是 **Transformer** 一次位移 10 天模型，對其他間台灣上市公司股票(台積電以外之股票)進行交易策略模擬，得到買入金額設定和報酬率之關係。

表七、中鋼買進金額設定與報酬率的關係-Transformer Model

本金:一千萬	買進金額設定	Final cash	Returns
2002(中鋼)	0.1	15973123.22	59%
	0.2	16128192.69	61%
	0.3	16241299.58	62%
	0.4	16362635.32	63%
	0.5	16466946.38	64%
	0.6	16563170.36	65%
	0.7	16655207.56	66%
	0.8	16734376.27	67%
	0.9	16823657.79	68%
	1	16916965.35	69%

表十、台塑買進金額設定與報酬率的關係-Transformer Model

本金:一千萬	買進金額設定	Final cash	Returns
1301(台塑)	0.1	13374135.4	33%
	0.2	13948408.13	39%
	0.3	13963293.27	39%
	0.4	13817662.77	38%
	0.5	13632437.77	36%
	0.6	13463818.79	34%
	0.7	13331878.35	33%
	0.8	13245120.38	32%
	0.9	13150833.52	31%
	1	13112934.56	31%

八、參考文獻

[1] 李弘毅, ML Lecture 10: Convolutional Neural Network, 2017, 取自

https://www.youtube.com/watch?v=FrKWiRv254g&ab_channel=Hung-yiLee

[2] 李弘毅, ML Lecture 21-1: Recurrent Neural Network (Part I), 2017,

取
https://www.youtube.com/watch?v=xCGidAeyS4M&ab_channel=Hung-yiLee
[3] 李弘毅, Transformer, 2020, 取自
https://www.youtube.com/watch?v=ugWDIIOHtPA&t=13s&ab_channel=Hung-yiLeeHung-yiLee
[4] 夏鶴芸, 2020, 〈應用深度學習與自然語言處理新技術預測股票走勢-以台積電為例〉, 台北大學資訊管理研究所碩士論文
[5] 許琮苓, 2019, 〈基於長短期記憶網路模型之股市趨勢預測〉, 成功大學工程科學系碩士論文
[6] 吳哲緯, 2017, 〈使用深度學習卷積神經網路預測股票買賣策略之分類研究〉, 中山大學資訊管理學系碩士論文
[7] 張力元, 2018, 〈深度學習應用於股價走勢之研究:以大陸市場為例〉, 國立政治大學風險管理與保險學系研究所碩士論文
[8] 賴嘉蔚, 2018, 〈卷積神經網路預測時間序列能力分析〉, 國立政治大學金融學系研究所碩士論文
[9] TengYuan Chang, 淺談遞歸神經網路 (RNN) 與長短期記憶模型 (LSTM), 2019, 取自
<https://tengyuanchang.medium.com/%E6%B7%BA%E8%AB%87%E9%81%9E%E6%AD%B8%E7%A5%9E%E7%B6%93%E7%B6%B2%E8%B7%AF-rnn-%E8%88%87%E9%95%B7%E7%9F%AD%E6%9C%9F%E8%A8%98%E6%86%B6%E6%A8%A1%E5%9E%8B-lstm-300cbe5efcc3>
[10] 郭祿丁, 2018, 〈Using Deep Learning Neural Networks and

Candlestick Chart Representation to Predict Stock Market〉, 元智大學資訊工程學系碩士論文

[11] 黃生豪, 2020, 〈基於 LSTM 深度學習預測 ETF50 市場趨勢並結合交易策略之實證分析〉, 高雄科技大學金融資訊系碩士論文

[12] 張嘉文, 2019, 〈以 LSTM 模型預測台股期 1 分鐘漲跌結果〉, 國立台北大學統計學系碩士論文

[13] PJ Wang, 利用 Keras 建構 LSTM 模型, 以 Stock Prediction 為例 1, 2018, 取

自
<https://daniel820710.medium.com/%E5%88%A9%E7%94%A8keras%E5%BB%BA%E6%A7%8B1stm%E6%A8%A1%E5%9E%8B-%E4%BB%A5stock-prediction-%E7%82%BA%E4%BE%8B-1-67456e0a0b>

[14] Dou Wei, Prediction of Stock Price Based on LSTM Neural Network. In: 2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM). IEEE, 2019.

[15] Tingwei Gao, et al. Applying Long Short Term Memory Neural Networks for Predicting Stock Closing Price. In: 2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS), 2017.

