**國立臺中科技大學資訊工程系碩士班**

**碩士論文**

**藉由深度注意力及轉換器為基礎的神經網路並結合混頻率財務資料與月營收之股票預測**

**Stock Prediction via Deep Attention and Transformer Based Neural Network Combining Mixed Frequency Financial Data and Monthly Revenue**

指導教授：劉冠顯博士 Dr. Kuan-Hsien Liu

研究生：林毓翔 Yu-Shiang Lin

**中華民國 113 年12月**

藉由深度注意力及轉換器為基礎的神經網路並結合混頻率財務資料與月營收之股票預測

學生：林毓翔 指導教授：劉冠顯教授

國立臺中科技大學 資訊工程系 碩士班

# 摘要

股票預測是非常熱門的話題到雜訊程度較大，不容易準確預測，因此市面上有許多不同的學派發展出各式各樣的方法來預測股價走勢進而制定有效的交易策略，期望能在股票市場賺取超額報酬。

本論文中創建的跨頻率資料庫結合每季的財務比率和每月的月營收，結合公司基本面的財務比率套用深度學習模型作分析，並將提出的模型與LSTM、多因子迴歸模型、決策數模型比較，發現該模型的RMSE以及準確率優於其他方法，可作為日後投資策略上的新興參考方法。

關鍵字: 深度學習、股票預測、跨頻率資料庫、超額報酬、MSE

Stock Prediction via Deep Attention and Transformer Based Neural Network Combining Mixed Frequency Financial Data and Monthly Revenue

Student：Yu -SHIANG LIN Advisors：Dr. Kuan-Hsien Liu

# ABSTRACT

Stock prediction is a very hot topic. Since stocks have a lot of noise, it is difficult to predict accurately.Therefore, there are various methods to predict stock price trends and formulate effective trading strategies. to expected that can earn excess returns in the stock market.

The cross-frequency database created in this plan combines quarterly financial ratios and monthly revenue, and applies a deep learning model to the company's fundamental financial ratios for analysis, and compare the proposed model with LSTM and multi-factor regression models and Random Forest Regressor ,it was found that the MSE and accuracy of this model are better than other methods, can be used as an new method for future investment strategies.

Keywords: Deep learning, Stock prediction,

cross-frequency database, excess return

# 目次

[摘要 2](#_Toc91452222)

[ABSTRACT 3](#_Toc91452223)

[目次 4](#_Toc91452224)

[表目次 6](#_Toc91452226)

[圖目次 8](#_Toc91452226)

[第一章 簡介 10](#_Toc91452227)

[第二章 神經網路基礎 14](#_Toc91452228)

[2.1 神經網路 14](#_Toc91452229)

[2.2 激活函數 15](#_Toc91452230)

[2.3 全連接層 17](#_Toc91452231)

[2.4 損失函數 18](#_Toc91452231)

[2.5神經網路的訓練 20](#_Toc91452232)

[第三章 相關研究 24](#_Toc91452234)

[3.1 財務比率對股價的影響 24](#_Toc91452235)

[3.2 深度學習預測股價 25](#_Toc91452236)

[3.3 混合深度學習預測股價 27](#_Toc91452236)

[第四章 研究方法 29](#_Toc91452240)

[4.1 資料蒐集 32](#_Toc91452241)

[4.2 模型介紹 33](#_Toc91452242)

[4.3 財務指標 34](#_Toc91452242)

[4.4 深度學習模型 42](#_Toc91452242)

[第五章 資料庫 46](#_Toc91452243)

[5.1 YAHOO FINANCE 46](#_Toc91452244)

[5.2 公開資訊觀測站 49](#_Toc91452245)

[第六章 實驗與討論 51](#_Toc91452247)

[6.1 實驗設置 51](#_Toc91452248)

[6.2 損失函數 51](#_Toc91452249)

[6.3 實驗結果 52](#_Toc91452249)

[第七章 結論 76](#_Toc91452252)

[參考文獻 77](#_Toc91452253)

# 表目次

[表1 模型參數量 31](#_Toc91545547)

[表2 StockQM資料集架構 34](#_Toc91545547)

[表3 比較dropout比率的影響 52](#_Toc91545547)

[表4 有無使用dropout對模型的影響 56](#_Toc91545547)

[表5 以AAPL資料集比較encoder和decoder個數對模型的影響 58](#_Toc91545547)

[表6 以TSLA資料集比較encoder和decoder個數對模型的影響 59](#_Toc91545547)

[表7 以MSFT資料集比較encoder和decoder個數對模型的影響 59](#_Toc91545547)

[表8 以IBM資料集比較encoder和decoder個數對模型的影響 60](#_Toc91545547)

[表9 以StockQM資料集比較encoder和decoder個數對模型的影響 60](#_Toc91545547)

[表10 激活函數對模型的影響 61](#_Toc91545547)

[表11 模型加入DECODER結果 62](#_Toc91545547)

[表12 比較輸出層是否加入激活函數的結果 63](#_Toc91545547)

[表13 比較不同time\_step的影響 63](#_Toc91545547)

[表14 比較財務資訊對模型的影響 64](#_Toc91545547)

[表15 比較季財報公布月份對模型的影響 65](#_Toc91545547)

[表16 以AAPL資料集比較encoder和decoder個數對Transformer模型的影響 67](#_Toc91545547)

[表17以TSLA資料集比較encoder和decoder個數對Transformer模型的影響 67](#_Toc91545547)

[表18以MSFT資料集比較encoder和decoder個數對Transformer模型的影響 68](#_Toc91545547)

[表19以IBM資料集比較encoder和decoder個數對Transformer模型的影響 68](#_Toc91545547)

[表20以StockQM資料集比較encoder和decoder個數對Transformer模型的影響 69](#_Toc91545547)

[表20以StockQM資料集比較encoder和decoder個數對Transformer模型的影響 69](#_Toc91545547)

[表21以AAPL資料集比較ENCODER和DECODER個數對TRANSFORMER\_TCN模型的影響 69](#_Toc91545547)

[表22以TSLA資料集比較ENCODER和DECODER個數對TRANSFORMER\_TCN模型的影響 70](#_Toc91545547)

[表23以MSFT資料集比較ENCODER和DECODER個數對TRANSFORMER\_TCN模型的影響 70](#_Toc91545547)

[表23以MSFT資料集比較ENCODER和DECODER個數對TRANSFORMER\_TCN模型的影響 70](#_Toc91545547)

[表24以IBM資料集比較ENCODER和DECODER個數對TRANSFORMER\_TCN模型的影響 71](#_Toc91545547)

[表25以StockQM資料集比較ENCODER和DECODER個數對TRANSFORMER\_TCN模型的影響 71](#_Toc91545547)

[表26五個資料集與其他模型比較 72](#_Toc91545547)

[表27近期上市公司股價預測 75](#_Toc91545547)

# 圖目次

[圖 1 神經網路架構 14](#_Toc91545547)

[圖 2 Sigmoid 函數 15](#_Toc91545548)

[圖 3 Relu 函數 16](#_Toc91545549)

[圖 4 Tanh函數 17](#_Toc91545550)

[圖 5 全連接 18](#_Toc91545551)

[圖 6模型架構圖for AAPL、TSLA、MSFT、IBM 30](#_Toc91545551)

[圖 7模型架構圖for StockQM 31](#_Toc91545551)

[圖 8 空洞卷積 43](#_Toc91545551)

[圖 9 attention架構圖 44](#_Toc91545551)

[圖 10 yahoo finance tsla歷史價格 47](#_Toc91545551)

[圖 11 yahoo finance MSFT歷史價格 47](#_Toc91545551)

[圖 12 yahoo finance ibm歷史價格 48](#_Toc91545551)

[圖 13 yahoo finance aapl歷史價格 48](#_Toc91545551)

[圖 14 yahoo finance台泥歷史股價 49](#_Toc91545552)

[圖 15台泥毛利率 50](#_Toc91545552)

[圖 16 台泥月營收資料 52](#_Toc91545552)

[圖 17 AAPL資料集 dropout 25% loss 圖 53](#_Toc91545552)

[圖 18 TSLA資料集 dropout 25% loss 圖 53](#_Toc91545552)

[圖 19 MSFT資料集 dropout 25% loss 圖 53](#_Toc91545552)

[圖 20 IBM資料集 dropout 25% loss 圖 54](#_Toc91545552)

[圖 21 StockQM 資料集dropout 25% loss 圖 54](#_Toc91545552)

[圖 22 AAPL資料集 無dropout loss 圖 54](#_Toc91545552)

[圖 23 TSLA 無dropout loss 圖 55](#_Toc91545552)

[圖 24 MSFT 無dropout loss 圖 55](#_Toc91545552)

[圖 25 IBM 無dropout loss 圖 55](#_Toc91545552)

[圖 26 StockQM 無dropout loss 圖 56](#_Toc91545552)

[圖 27使用提議的模型，在AAPL的表現 73](#_Toc91545552)

[圖 28使用提議的模型，在TSLA的表現 74](#_Toc91545552)

[圖29使用提議的模型，在MSFT的表現 74](#_Toc91545552)

[圖30使用提議的模型，在IBM的表現 75](#_Toc91545552)

# 第一章 簡介

股票分析及預測在金融領域是非常熱門的話題，有許多流派發展出不同方法，其中在市面上最常見的為技術分析、籌碼分析、基本分析三種分析法。

技術分析認為歷史會重演，因此針對股價的變化衍生出的指標來分析，例如均線、K線(Candlestick chart)、Moving average convergence and Divergence (MACD)。籌碼分析則是觀察三大法人或大戶的動向進而採取策略。基本面分析認定公司的體質會反應在股價上面，因此著重於財務指標的分析，例如毛利率、股東權益報酬率、負債比率等等的分析，此學派認為良好的公司財務基本面與股價成正比，像是Ball and Brown (1968)[1] 是第一篇探討盈餘與股價關係的實證研究。

近年來，深度學習用在股價預測上逐漸提高，像是Doshi等人[2]運用LSTM 結合財務比率來預測股價並得到較高的準確率，以及 Lin等人[3]使用GAN模型來捕捉資料的特徵進而預測股價且取得不錯的效果。另外Das等人[4]套用 Transformer模型來預測股價，準確率較其他方法來的高。

本論文除了使用四個美國大型上市公司資料集，分別為Apple Inc.(AAPL) 、Tesla, Inc. (TSLA) 、Microsoft Corporation (MSFT)、International Business Machines Corporation (IBM)來預測股價外。我們想探討當財務資料公布後對股價的影響，因此自製結合了21個基本面分析的財務指標資料集，資料來源為公開資訊觀測站提供的五種公司財務分析能力(分別為償債能力、經營能力、獲利能力、成長能力)共18個季財務指標搭配3個月營收指標共21個指標變數並使用創新的transformer改良模型來預測財務指標公布後的月底股票價格，期望能結合混頻的季資料和月資料透過深度學習模型準確預測重大事件公布後的股價走勢，並與其他參考論文所提供的投資方法進行比較，最後得出本論文的方法優於其他方法。

本論文的安排章節如下，第二章介紹神經網路的基本運作，包含神經網路的基礎架構，神經層之間傳遞的激活函數種類和損失函數種類，以及全連接層的架構。接著介紹神經網路的訓練方法，像是梯度下降法的公式介紹和優化器的選擇種類，另外也介紹了神經訓練模型在資料量太少或是模型太複雜容易產生的過擬合問題以及解決方法，像是丟棄(Dropout)、權重正規化(Weight regularization)和縮減模型的大小。

第三章為相關研究，分為三個議題去探討，分別為財務比率對股價的影響和深度學習預測股價以及混合深度學習預測股價。在財務比率對股價的影響議題上面列舉了許多期刊文章做的主題研究，像是Zhang等人[10]認為股價與營收或淨利的相關性較高。而在深度學習預測股價議題上面也有許多期刊使用不同的深度學習模型並結合財務比率來做股價分析，例如有學者使用LSTM模型和隨機森林迴歸(Random Forest Regressor)預測股價[12]。另外在混合深度學習預測股價方面，

也介紹一些期刊文章列舉不同的混合模型來預測股價並取得不錯的預測效果，像是WANG等人[20]提出的Transformer-TCN模型套用在滬深300股市相較於其他混合模型有較佳的預測效果。

第四章為研究方法，由於上述提出的方法像是LSTM在時間序列的預測上面會隨時間的增長導致訊息的流失最終會影響預測結果，因此本論文使用的創新改良Transformer架構，透過注意力機制[40]加強對關鍵訊息的權重，提升對重要特徵的提取而取得較佳的預測結果，另外本論文自製的資料集StockQM資料架構也會在此章節做介紹，透過StockQM資料集搭配所提議的模型，公司只需提供少量的歷史資料像是近期公布的最新一季財務和最新月營收資料即可預新公司的股價即可預測股價，而非以往的訓練模型需要好幾筆同公司資料做訓練才能預測。接著本章節介紹資料蒐集和StockQM的21個財務指標特徵提取以及詳細介紹所使用的創新改良Transformer模型元件。

第五章為資料庫介紹，本論文採用五種資料庫，包含自製的財務指標資料庫，命名為StockQM，選取18個季財務指標和3個月營收指標共21個財務指標做股價預測。資料期間為2013/01/01至2024/07/31每月的財務比率和包含2013年第一季到2024年第1季共44季(132個月)的財務資料，資料來源為公開交易觀測站。另外四個資料庫分別為美國知名企業公司的每日股價，分別Apple Inc.(AAPL)、Tesla, Inc. (TSLA)、Microsoft Corporation (MSFT)、International Business Machines Corporation (IBM)的股價，此四個資料庫選取時間範圍為1980/1/1~2024/7/31，但由於每家公是上市時間不同因此會造成每間公司資料集的差異，TSLA資料期間為2010/6/29~2024/7/31、MSFT資料期間為1986/03/13~2024/7/31、IBM資料期間為1980/01/01~2024/7/31、APPL資料期間為1980/12/12~2024/7/31的每日股價，資料來源為YAHOO FINANCE網站所提供的股價資訊。

第六章為實驗與討論，介紹本論文實驗設置環境和所使用的損失函數以及模型的評估方式，另外也做了許多消融測試並和不同方法進行比較。

第七章對本論文做出總結，本論文提出的Transformer改良模型結合了注意力機制對重要特徵的加強關注，以及空洞捲積捕捉更大範圍的資料特徵，並將各分支的資訊做相加彙整加特徵的提取，而實驗結果也優於其他方法比較，最後本論文所自製的融合季和月財務指標的創新資料集StockQM搭配本論文提出的模型可有效的預測股價。

本論文的貢獻如下:

1.本論文所使用的創新資料庫為結合季財務資料和月營收資料的混合頻率資料並結合深度學習模型所股價預測分析。

2.使用Transformer創新模型來預測股價，加入了多分支空洞卷積搭配殘差神經網路和注意力機制來強化重要特徵訊息，提升準確率。

3.提出的模型架構證實融入財務比率的因子將提升整體的預測率。

4.使用模型訓練的訓練參數只需要少量資料即可預測股價，像是透過公司

近期公布的最新一季財務和最新月營收資料即可預新公司的股價，而非以往的訓練模型需要好幾筆同公司資料做訓練才能預測。

# 第二章 神經網路介紹

**2.1神經網路(Neural network)**

神經網路為電腦模擬人腦神經傳遞資訊的架構，並建立在錯誤中不斷學習調整最佳方法的模型，為深度學習的重要架構。神經網路可分為輸入層、隱藏層、輸出層，各層神經元靠著權重和偏差及激活函數傳遞訊息，其中激活函數主要處理非線性問題，目前神經元廣泛應用在各領域中，像是影像辨識、訊號解析、自然語言分析、自動駕駛、大數據應用，圖1為神經網路架構)

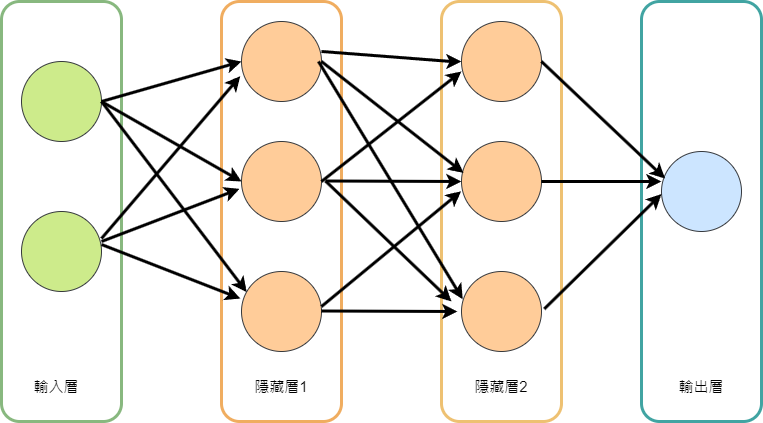


圖1 神經網路架構

**2.2 激活函數 (Activation function)**

各層的神經元加入權重和偏差項後透過激活函數傳遞到下一層，激活函數的存在可讓神經元不只能處理線性問題，更能去探討非線性領域，增加網路的效率性，以下介紹常見的幾種激活函數類型:

Sigmoid函數:為神經元常見的激活函數，函數S曲線的形狀又成為S函數，將輸出值壓縮在0到1之間，常被用在二元分類問題，優點為連續函數，利於求導,可控制反向傳播的幅度，缺點為上下平坦，若變數有些微的變動造成輸出層的差異不大，尤其是在網路層數越多時，反向傳播的梯度趨近於0，導致前幾層的權重很難被更新，造成梯度消失，影響神經網路的效率，Sigmoid函數如公式(1)和Sigmoid函數圖形如圖2:

**(1)**

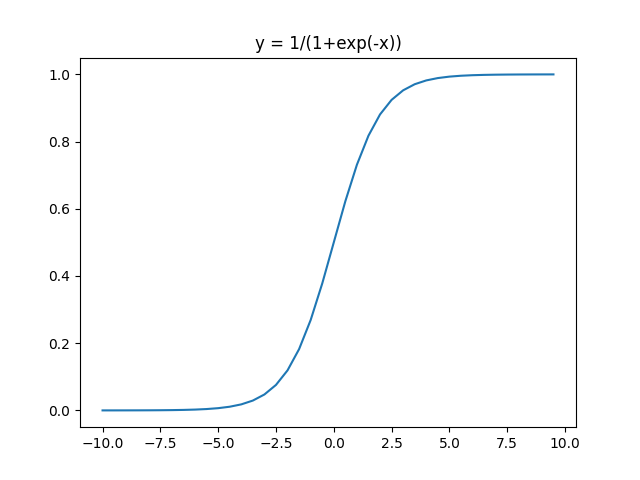


圖2 Sigmoid函數

Relu函數:為最受歡迎的激活函數，優點為在x>0狀況，微分為1

，解決sigmoid函數梯度消失的困擾，收斂速度快，效率佳，缺點為X<0，

梯度是0，網路層權重無法更新，Relu函數如公式(2)和Relu函數圖形如圖3:

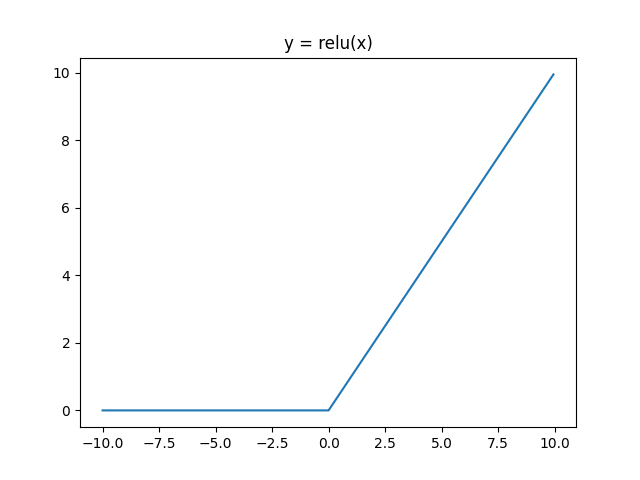


圖3 Relu函數

Tanh 函數:跟Sigmoid函數類似，如圖四所示，差別在於Tanh 函數輸出範圍在-1到1之間為0均值(zero-centered)，收斂速度較快，在下一層的傳遞效率較佳，缺點跟Sigmoid相同，都會有梯度消失的情形，導致運算時間相對耗時，Tanh函數如公式(2)和Relu函數圖形如圖4:

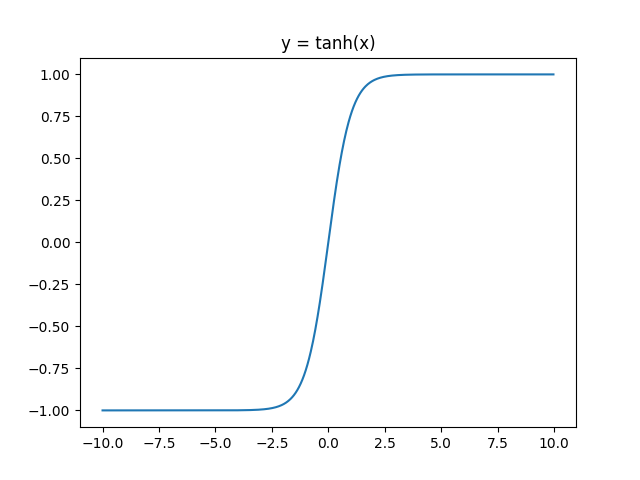
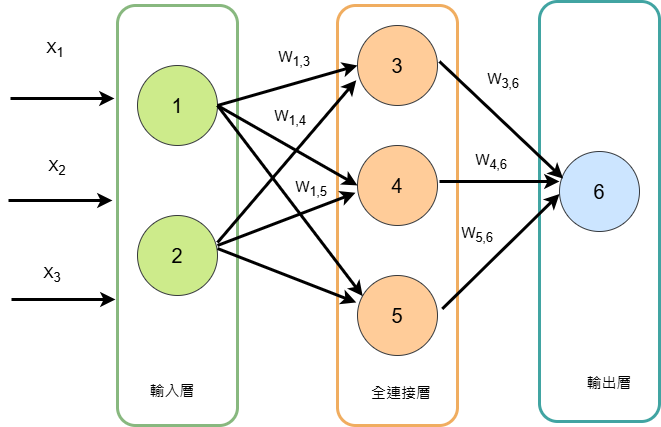


圖4 Tanh函數

**2.3全連接層(Fully connected layer)**

每層神經元與前一層的每個神經元透過權重以及激活函數相接，通常用於

與輸出能連接，如圖5所示:

 **圖5 全連接(Fully connected layer)**

**2.4損失函數（Loss Function）**

大部分的演算法都會設目標函數來測試是否能達成期望的目標，而深度學習的目標函數為損失函數，用來計算實際值與模型預測值之間的差異，並追求損失函數極小化，當模型的損失函數越小代表預測值較接近實際值，模型效率較高，而探討不同問題會使用相對應的損失函數，例如迴歸問題通常使用均方誤差(Mean Squared Error, MSE)，公示(4)如下，或平均絕對誤差(Mean Absolute Error)做損失函數，公示(5)如下，二元分類問題通常使用損失函數為Binary Cross-Entrropy，公示(6)如下，多元分類問題則會通常使用Categorical Cross-Entrropy作損失函數，公示(7)如下:

均方誤差(Mean Squared Error,MSE):

其中為資料輸出值，為深度學習模型預測值，*N*為分批的資料量

Binary Cross Entropy(BCE)，公式(6)如下:

(6)

其中為資料輸出值，為深度學習模型後預測值，*N*為分批的資料量，*f*為激活函數。

Categorical Cross Entropy(CCE)，公式(7)如下:

其中為資料輸出值，為深度學習模型經過softmax後預測值，*N*為分批的資料量,*f*為激活函數,C為類別數量。

**2.5神經網路的訓練(Neural network training)**

以下介紹幾種神經網路訓練方法

**2.5.1梯度下降法（Gradient descent）**

神經網路的目標為設計模型讓損失函數最小化，而透過梯度下降法，權重將在每次更新的到優化並降低損失，公式(8)如下:

其中*L*為損失函數，*Q*為學習率，*W*為權重，模型每一次更新，權重將會隨著學習率朝著減少損失的方向前進，最終得到區域最小值，而梯度分析法可以依資料量的切割而不同，像是批量梯度下降法(Batch Gradient Descent)是將資料依批次大小切割並分批帶入模型，例如批次為32，每次從資料抽取32筆資料並分批訓練來更新權重,相較於傳統梯度下降法一次訓練全部資料更省時且更有效率，另外在學習率的選擇也是非常重要的，若選太大更新幅度會過大權重將會超過最佳值，若選過小更新速率會太慢，整體不具效率。

**2.4.3優化器（Gradient Descent）**

在深度學習的目標是讓損失函數最小化，找出全域最小值，優化器則是在深度學習中用於調整模型參數以最小化損失函數，最常見的方法為梯度下降法，以下介紹常見的幾種優化器方法:

隨機梯度下降法(SGD):與2.5.1的批量梯度下降法(Batch Gradient Descent)類似，差別在於資料為隨機抽取並分批帶入模型訓練。

動量（Momentum）:透過慣性依照過去訓練的方向和速度，同方向的時候學習速度與權重更新會正向變快，但若方向改變時，學習速度和權重更新會變慢找到一條最短路徑通往區域最低點，節省訓練時間,公式(9) (10)如下，其中*L*為損失函數，*Q*為學習率，*W*為權重，*B*為動能參數在0到2之間設值，當*B*為0時，等同於隨機梯度下降法(SGD)。

Adam優化器:為改善Momentum的優化器，將慣性賦予權重，採用指數平滑法，越接近現在的梯度方向給予權重越大，越久的梯度方向的權重較小以此即時更新並修正梯度方向，避免若選擇到錯的方向，依據動量（Momentum）一直往錯的方向走下去,為目前深度學習常用的優化器方法，公式 (11)如下，其中 、

為Momentum參數。

**2.4.4 過擬合問題處理(Overfitting problem handling)**

當訓練資料太少或模型訓練太複雜容易產生過擬合問題，也就是在訓練資料的損失值會隨訓練次數下降，但在測試資料的損失值卻隨訓練次數上升而提高，以下方法為解決過擬合問題。

**2.4.4.1 丟棄(Drop out)**

因為模型訓練參數太多會產生過擬合問題，因此設定Drop out將會在每次訓驗隨機丟棄一定比例的神經元，以避免模型過度依賴特定的神經元特徵去學習。

**2.4.4.2 權重正規化(Weight regularization):**

此方法核心思想一樣為避免模型過度依賴特定神經元特徵，因此在損失函數的設計對權重做一些限制，分為L1 regulation 和 L2 regulation兩種。

L1 regulation,公式(12)如下:

其中為可調整參數來設權重正規化，*w*為權重，*M*為模型的總參數量，為模型的均方誤差。

L2 regulation,公式(13)如下:

其中為可調整參數來設權重正規化，*w*為權重，*M*為模型的總參數量，為模型的均方誤差。

**2.4.4.3 縮減模型大小(Reduce model size):**

將模型的參數所小，例如將在隱藏層的128個神經元數縮減為64個神經元來避免過擬合問題。

**第三章 相關研究**

深度學習已被廣泛的運用在財金領域，其中在股價的預測上面已有許多文獻使用不同方法去探討。

**3.1財務比率對股價的影響 ( Impact of financial ratios on stock prices)**

Ball等人[1]研究1957年至1965年9年間261家美國公司的樣本並發現股價與會計資訊之間有關聯以至於投資人能在公司股票獲得超額獲利。

　　Dechow 等人[8]研究在考慮那些會計財務因子加入會提高股價報酬率，研究發現會計財務比率更能解釋與股價的關聯。

Li等人[7] 研究中國股市A股在上海證券交易所和深圳證交所的表現，發現篩選高本益比（Price to earning ratio,PE）的投資組合表現最好。

Zhang等人[10]探討中國上市股票與五個財務和交易因子(收入、淨利、每股盈餘、每股盈餘、交易量)的關係，結果顯示，在醫藥產業中，股價與這五個指數的相關性最強，在電子資訊產業等一些產業中，股價與營收或淨利的相關性較高。

Yinhong Shi等人[11]使用10個財務變數搭配線性迴歸來預測深圳300指數股價並套用在投資策略上得到不錯的投資績效。

Deng等人[24] 使用財務比率因子搭配XGBoost機器學習方法來預測中國股市崩盤的風險，結果顯示提出的方法優於其他傳統機器學習方法，特別在小型股預測中準確率達78.41%。

Fitriyana等人[25]將16項財務比率透過主成分分析作為降維方法在不遺失資訊的情況下形成影響股價的主要成分，並使用了印尼證券交易所中市值最高的五家上市公司做為研究對象，結果表明，獲利能力比率對股票價格的決定有較高的貢獻。

**3.2深度學習預測股價(Deep learning predicts stock prices)**

[Doshi](https://paperswithcode.com/author/akash-doshi)等人[2]使用長短期記憶網路 (Long Short Term Memory Network, LSTM)與差分整合移動平均自我迴歸模型(Autoregressive Integrated Moving Average model,ARIMA)模型做預測股價的比較，套用美國的汽車公司分別為Ford、Tesla、Toyota, General Motors，發現LSTM在預測高股價波動的部分成效不如預期，但在用LSTM套用在投資策略的報酬會遠高於ARIMA。

Fauzan等人[12]提出了使用LSTM模型和隨機森林迴歸(Random Forest Regressor,RF)來預測Apple Inc. (AAPL)、Amazon.com Inc. (AMZN)、Alphabet Inc. (GOOG)、Microsoft Corporation (MSFT)、Tesla, Inc. (TSLA)五家頂尖的美國公司股票，結果顯示使用LSTM模型的預測股價非常接近實際價格的股票價格且準確率達到97.2938%。平均誤差為 2.3223%。

[Doshi](https://paperswithcode.com/author/akash-doshi)等人[2]使用LSTM模型與ARIMA模型做預測股價的比較，套用美國的汽車公司分別為Ford, Tesla, Toyota, General Motors，發現LSTM在預測高股價波動的部分成效不如預期，但在用LSTM套用在投資策略的報酬會遠高於ARIMA。

Refenes等人[13]研究顯示透過神經網路模型預測股價效果會比一般線性迴歸來的好。

Bareket等人[14]使用多種機器學習包含支援向量機(Support Vector Machine,SVM)、K-近鄰演算法( K Nearest Neighbor,KNN)、集成投票分類(Ensemble Voting Classifier)、隨機森林迴歸(Random Forest Regressor,RF)來預測納斯達克100指數和道瓊工業指數以及德國 Dax 指數，結果發現這些計器學習在三種指數中的預測效果都是不錯。

Kryzanowski等人[15]研究顯示神經網路結合財務比率與宏觀經濟因子可以達到準確率72%分類一年內股價的正負報酬。

Abe1等人[16]使用25變數(包含23個財務指標)套用神經網路模型去預測MSCI日本指數成分股，結果顯示神經網路模型的關聯性與準確率均優於支持向量迴歸（Support Vector Regression, SVR）與隨機森林模型，其中又以8層的神經網路預測效果較佳。

Das等人[17]使用Transforemer模型預測IBM、APPLE 和 TATAMOTORS股價與其他模型相比均取得較小的預測誤差，

Lin等人[18]將WGAN-GP模型預測那斯達克綜合指數(NASDAQ)、紐約證券交易所綜合指數(NYSE) 、標準普爾500指數(S&P 500) 、富時100指數(FTSE100) 、日經平均指數(NIKKI225)......等等一系列指數實驗證實該模型的預測效果較LSTM、GRU、GAN模型好。

Narayana等人[26]使用閘門循環單元(Gated Recurrent Unit, GRU)預測那斯達克綜合指數(NASDAQ)、標準普爾500指數(S&P 500)，結果顯示在準確率和誤差率方面效果均比LSTM模型好。

Mirjebreili等人[27]使用多分支Transforemer模型並套用200個股票資料集進行測試，實驗結果顯示出加入多分支可以更加強對特徵的提取並取得不錯的股票預測結果。

Alhazbi等人[28]使用一維卷積神經網路(1DCNN)來對卡達交易所和日經指數以及油價進行預測，結果顯示有不錯的預測成績。

Li等人[29] 使用長短期記憶（LSTM）和支持向量機（Support vector machine ,SVM）來預測中國上海證券交易所50 指數（SSE 50）中的所有股票價格走勢，結果顯示LSTM 表現優異。

Naresh等人[30] 使用隨機森林 (Random forest,RF)搭配自然語言來預測美國股票市場，結果顯示有不錯的準確率。

Ravikumar等人[30] 使用隨機森林 (Random forest,RF)、支持向量機（Support vector machine ,SVM）、邏輯迴歸 (Logistic regression)來預測標準普爾500指數(S&P 500)，結果顯示出邏輯迴歸的預測效果最佳。

Alamsyah等人[40]使用通貨膨脹、利率和匯率作為宏觀經濟變數搭配神經網路來預測印尼證券交易所綜合指數，結果顯示準確率高達96.38%。

Chang等人[45] 使用基於決策樹演算法改良的LightGBM演算法（Light Gradient Boosting Machine）預測台灣ETF 0050和ETF0056並取得良好的預測效果。

**3.3混合深度學習預測股價**

在近年來，混合的深度學習模型也逐漸熱門， Zhang等人[19]使用CNN\_LSTM模型套用在紐約證券交易所綜合指數(NYSE)、那斯達克綜合指數(NASDAQ) 、印度指數50(NIFTY)，實驗結果顯示CNN-LSTM有不錯的預測效果。

Yousuf等人[42]也將CNN\_LSTM用於預測達卡證券交易所四家熱門公司的歷史價格，該模型表現良好且具有很強的泛化能力，其中 CNN 提取股票數據的特徵，然後利用 LSTM 來尋找模式並預測股票市場也優於其他方法。

Zhang等人[19]提出CNN-BiLSTM-Attention模型預測滬深300指數資料並與LSTM、CNN-LSTM、CNN-LSTM-Attention模型比較實驗結果證實CNN-BiLSTM-Attention模型的預測效果較三個模型佳。

WANG等人[20]提出的Transformer-TCN模型套用在滬深股市相較於其他混合模型有較佳的預測效果。

上述提出的方法像是LSTM在時間序列的預測上面會隨時間的增長導致訊息的流失最終會影響預測結果，因此本論文使用的創新改良Transformer架構，透過注意力機制[40]加強對關鍵訊息的權重，提升對重要特徵的提取而取得較佳的預測結果，且上述提到相關文獻方法均需要實驗股票大量的資料，今天若一個剛上市不久的股票沒那麼多資料就無法套用，而本論文提出的模型搭配自製資料集StockQM只需少量資料(上市公司最新公布的季財務和月營收指標)就可預測股票。

# 第四章 研究方法

本論文實驗使用的模型為基於Transformer模型改良，在encoder部分會先Layer -Normalization 讓每一層的神經元都服從常態分佈，提高模型的泛化能力，接著再進入注意力機制(attention mechanism)加強對重要特徵的權重放大其重要性，然後帶入Dropout 25%神經元避免模型過度依賴某些特徵，再與一開始Layer Normalization的結果近行相加(Add)。接著分兩個分支，擴展不同尺度特徵的萃取，一支分支執行空洞卷積且擴張率為2，另一支分支執行執行空洞卷積且擴張率為4，再將兩支空洞卷積串聯(concatenate)成一個分支再帶入卷積層中，接著再使用Dropout 丟棄 25%神經元，到此一步驟encoder部分就完成了，接著進入GAP層萃取特徵準備進入decoder部分，在decoder部分，本論文將傳統的Transformer的decoder部分替換成將encoder結果也分成兩個分支，第一個分支先保留encoder結果，第二個分支先進入10個神經元的全連接層(Dense(10))再Dropout 25%的神經元，再做Dense(10)和Dropout 25%的神經元，並將結果與第一個分支相加，最後將decoder部分的結果與encoder後的GAP結果互相串聯(concatenate)，最後輸出預測的股價，本論文的架構圖為圖6和圖7，兩者的差異主要為輸入的資料型態不同，本論文使用五個資料庫，其中四個資料庫使用四間美國大型上市公司每日股票資料集來做股票價格預測，分別為Apple Inc.(AAPL)、Tesla, Inc. (TSLA)、Microsoft Corporation (MSFT)、International Business Machines Corporation (IBM)，選取前10天股價來預測第11天股價，資料集的time\_step設為10，輸入的x label特徵只有一個為前10日的股價，因此輸入的資料型態(10,1)並且本論文的decoder部分Dense層設為10個神經元，此四個資料集架構圖為圖6。本論文有自創資料集(命名為StockQM)，選取台灣上市公司共405間搭配季18個財務指標和3個月財務指標共21個財務指標來預測股價，詳細的資料庫資訊會在4.2章節介紹，資料集的time\_step設為1，輸入的x label特徵有21個財務指標來預測隔月股價，因此輸入的資料型態為(1,21) ，其中1是資料的time\_step為1代表本論文取最新公布的季財務指標和當月營收財務比率來做x label特徵，y label為隔月股價，而21代表21個財務指標，且本論文的decoder部分Dense層設為10個神經元，此資料集架構圖為圖7，模型參數量為表1。

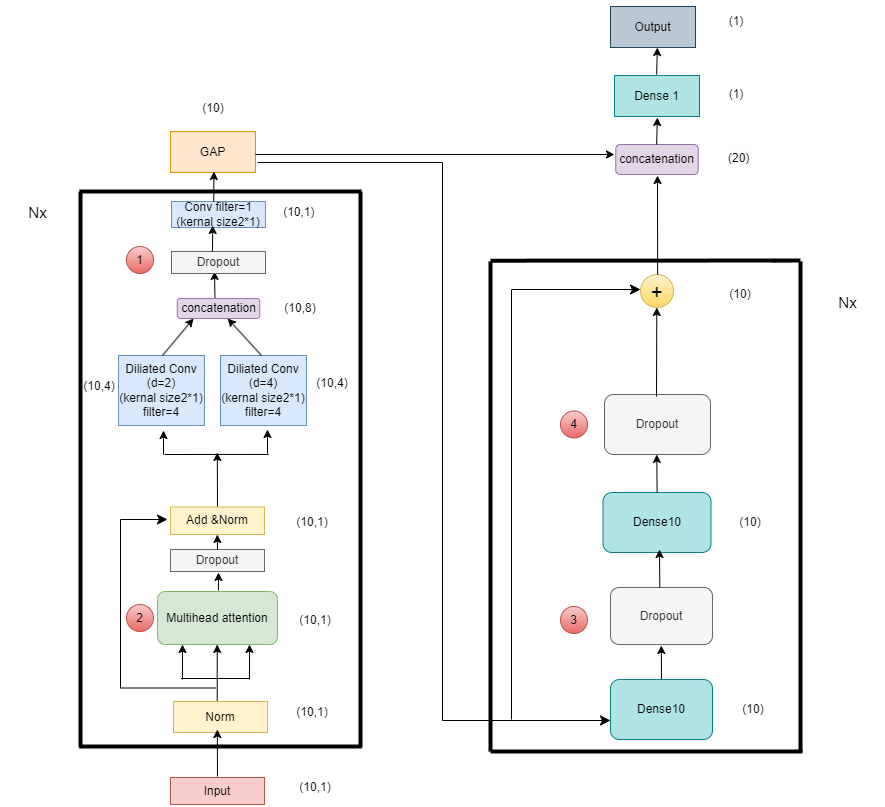


圖6 模型架構圖for AAPL、TSLA、MSFT、IBM

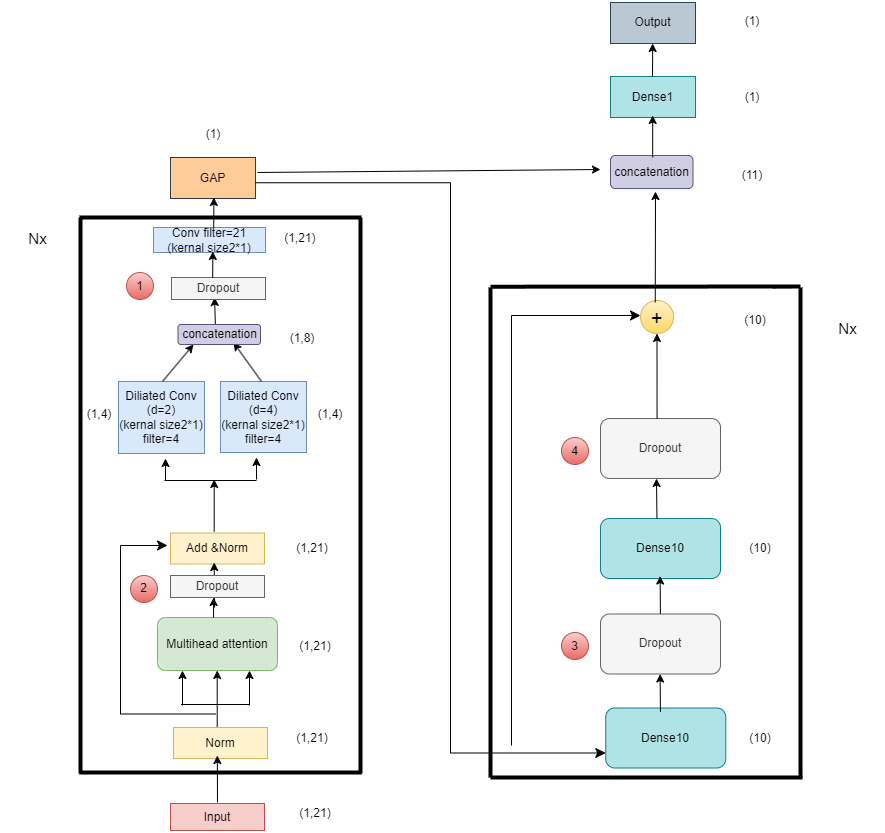


圖7模型架構圖for StockQM

|  |  |  |  |
| --- | --- | --- | --- |
|  | Total params | Flops | Training Time |
| AAPL | 4889 | 20625 | 15min |
| TSLA | 7055 | 17389 | 16min |
| MSFT | 6333 | 3934 | 17min |
| IBM | 8437 | 9172 | 12min |
| StockQM(Ours) | 59932 | 135057 | 15min |

表1 模型參數量

**4.1資料蒐集(Data collection)**

本論文共有五種資料集，包含四間美國大型上市公司資料集和自創資料集StockQM，所有資料集在訓練時將數據分成訓練集、驗證集和測試集，比例分別為60%、20%、20%。

在自創資料集StockQM中，篩選了代表公司財務體質的5種能力面向(財務結構、償債能力、經營能力、獲利能力、現金流量)共18項季財務指標(負債佔資產比率、長期資本佔固定資產比率、流動比率、速動比率、利息保障倍數、應收帳款週轉率、平均收現日數、存貨週轉率、平均銷貨日數、總資產週轉率、毛利率、營業利益率、稅後純益率、資產報酬率、股東權益報酬率、營業現金對流動負債比、營業現金對負債比、營業現金對稅後純益比)和3項月營收指標(上月比較增減(%) 、去年同月增減(%) 、前期比較增減(%))，以上共21個變數。資料期間為2013/01/01至2024/07/31每月的財務比率共132月和包含2013年第一季到2024年第1季共44季的財務資料，符合此條件的上市公司共405間，資料總筆數為53460筆。

本論文另外使用四間美國大型上市公司每日股票資料集來做股票價格預測，分別為Apple Inc.(AAPL)、Tesla, Inc. (TSLA)、Microsoft Corporation (MSFT)、International Business Machines Corporation (IBM) ，其中AAPL資料期間為1980/12/12~2024/7/31資料筆數共10998筆，TSLA資料期間2010/6/29~2024/7/31為資料筆數為10998筆，MSFT資料期間為1986/3/13~2024/7/31，資料筆數共9672筆，IBM資料期間為1980/1/1~2024/7/31，資料筆數共11238筆。

**4.2模型介紹(Model introduction)**

在StockQM資料集方面，本論文所有探討的議題為結合季財務指標和月營收指標搭配深度學習模型預測短期的股價報酬率走勢，因此x label為4.1章節所提的18項季財務指標以及3項月營收指標共21個變數，y label為股票季財報公布後的下個月月底的股價價格作為預測值，之所以選下個月月底的股價價格是月營收會在每月的10號前公布，而在重要財務指標公布後的一段時間內會影響股價價格，因此在季財務指標公布後的隔月月底會同時有上市公司的季財務指標和月財務指標，便可用這些財務指標預測當月月底的股價。由於此資料集每間公司包含132個月財務指標和44個季財務指標，本論文將資料頻率設為月頻率，資料集的結構為表2顯示，舉例來說，台泥第4季財報公布完(第4季財報必須在3/31前公布)，而3月的月營收資料會在4/10之前公布，也就是在4/11號時投資人可取得台積電的第4季財報資料和3月的月營收，進而可以藉著這些上個月的財務指標去預測當月底(4/30)的台泥股價而去選擇是否投資該公司，而在5月底的股價預測方面，我們仍選擇第4季財報指標搭配5/10前出的4月營收來預測5/31股價，直到8月份的8/14公布第4季財報我們的季財務指標才能做更新。其中T代表使用當月份公布的季財務資料搭配當月份公布的月財務資料來預測股價，適用的月份為3、5、8、11月份，T-1為延用上月份公布的季財務資料搭配當月份公布的月財務資料來預測股價，適用的月份為4、6、9、12月份，T-2為延用前兩個月份公布的季財務資料搭配當月份公布的月財務資料來預測股價，適用的月份為5、7、10、11月份，T-3為延用前三個月份公布的季財務資料搭配當月份公布的月財務資料來預測股價，適用的月份只有2月。本資料集在我們的模型timestep設1，也就是使用前1個月的21個財務資料來預測隔月月底股價。

在另外四個大型公司資料集，分別為Apple Inc.(AAPL) 、Tesla, Inc. (TSLA) 、Microsoft Corporation (MSFT)、International Business Machines Corporation (IBM)的每日股價，本資料集在我們的模型timestep設10，即本論文使用前10日股價來預測第11日股價，x label為前10日股價，y label為第11日股價。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 月份 | 季財務資料 | 月財務資料 | 預測股價 | 符號表示 |
| 5 | 5/15前公布第一季財報(1~3月) | 6/10公布5月營收 | 6/30股價 | T |
| 6 | 5/15前公布第一季財報(1~3月) | 7/10公布6月營收 | 7/31股價 | T-1 |
| 7 | 5/15前公布第一季財報(1~3月) | 8/10公布7月營收 | 8/31股價 | T-2 |
| 8 | 8/14前公布第二季財報(4~6月) | 9/10公布8月營收 | 9/30股價 | T |
| 9 | 8/14前公布第二季財報(4~6月) | 10/10公布9月營收 | 10/31股價 | T-1 |
| 10 | 8/14前公布第二季財報(4~6月) | 11/10公布10月營收 | 11/30股價 | T-2 |
| 11 | 11/14前公布第三季財報(7~9月) | 12/10公布11月營收 | 12/31股價 | T |
| 12 | 11/14前公布第三季財報(7~9月) | 1/10公布12月營收 | 1/31股價 | T-1 |
| 1 | 11/14前公布第三季財報(7~9月) | 2/10公布1月營收 | 2/28股價 | T-2 |
| 2 | 11/14前公布第三季財報(7~9月) | 3/10公布2月營收 | 3/31股價 | T-3 |
| 3 | 3/31前公布第四季財報(10~12月) | 4/10公布3月營收 | 4/30股價 | T |
| 4 | 3/31前公布第四季財報(10~12月) | 5/10公布4月營收 | 5/31股價 | T-1 |

表2 StockQM資料集

**4.3財務指標(Financial indicators)**

本論文採用基本面分析來預測股價走勢，基本面分析認為公司的股價最終會回歸到基本面發展,本論文採用18項季財務指標以及3項月營收指標共21個變數作x label，其中18個財務指標又可分類成五個領域，分別是財務結構、償債能力、經營能力、獲利能力、現金流量，以下逐一介紹:

**財務結構(Financial structure)**

1.負債佔資產比率(Debt ratio)

公式為負債和資產的比率，該指標衡量公司的償還負債的能力，代表公司有多少資產可以償還負債，該比率過高代表公司為高槓桿公司，償還債務風險較高，公式(14)如下:

2. 長期資本佔固定資產比率（Long term debt to total assets ratio）

衡量公司的長期資本支應固定資產的比率，該比率越高代表公司長期資本足以支應固定資產，公式(15)如下:

**償債能力(Solvency)**

3.流動比率(Current ratio)

衡量公司短期償債能力的指標，比率越高代表公司短期償債能力越高，其中流動資產為一年內能變現的資產，包括現金、有價證券、存貨等等，流動負債為一年內需償還的負債，例如:短期債務、應付債款，公式(16)如下:

4.速動比率(Quick ratio)

與流動比率同樣為衡量公司短期債債指標，差別在於速動資產為流動資產扣除變現能力較差的存貨與預付帳款，更能圖顯公司的短期償債能力，公式(17)如下:

5.利息保障倍數(Interest coverage ratio)

利息保障倍數代表公司經營的利潤是否足以償還舉債產生的利息，為衡量公司長期償債能力的指標，該比率越高代表公司償債能力越高，其中稅前息前例犉為公司未扣除利息和稅的盈餘，公式(18)如下:

**經營能力(**Operating ability**)**

6.應收帳款週轉率（Accounts receivable turnover ratio）

衡量公司收回應收帳款的指標，若該指標越高代表公司收回應收帳款的能力越強，可降低經營所產生的呆帳風險，其中平均應收帳款為期初應收帳款和期末應收帳款加總的平均，公式(19)如下:

7. 平均收現日數( Average collection days)

該比率計算為365除以應收債款週轉率，代表公司需要幾天才能收回應收帳款，該比率越低代表公司收回應收帳款所需天數越少，在經營能力越能減緩呆帳風險，公式(20)如下:

8.存貨週轉率（Inventory turnover）

該比率衡量存貨的週轉速度，比率越高代表公司賣出庫存的次數越多，公司的營運效率越好，其中平均存貨為期初存貨和期末存貨加總的平均，公式(21)如下:

9. 平均銷貨日數（Days sales of inventory）

衡量存貨周轉天數的指標，該比率為365除以存貨週轉率，代表公司將庫存存貨賣出所需花費多少天數，該比率越低代表公司在產品經營下越有效率，公式(22)如下:

10.總資產週轉率(Asset turnover ratio)

代表公司每單位投入的資產能產出多少營收，為衡量公司經營效率的重要指標，其中總資產為期初資產和期末資產加總的平均，公式(23)如下:

**獲利能力(Profitability** ability**)**

11.毛利率(Gross profit margin)

毛利為營收扣除營業成本，毛利率為毛利除以營業收入，衡量公司的獲利能力，該比率越高說明公司產品的獲利率越好，所創造的商品品牌價值越高，公式(24)如下:

12.營業利益率(Operating profit margin)

公司經營的成本除了商品的營業成本外，還需考量到人事或廣告費用的營業成本，營業利益率率即為衡量公司本業的獲利能力指標，其中營業利益為將毛利再扣除營業成本，公式(25)如下:

13.稅後純益率(Net income margin)

衡量公司每單位的營收能夠創造多少稅後淨利，淨利為稅前淨利減所得稅費用，該比率越高代表公司的獲利能力越強，公式(26)如下:

14.資產報酬率(Return on assets)

衡量公司每單位資產能產生多少稅後淨利，該比率越高代表公司資產運用效率越高，其中平均資產為期初資產總額和期末資產總額加總的平均，公式(27)如下:

15. 股東權益報酬率(Return on equity)

衡量股東投資每單位的金額所能獲得的報酬，為投資人衡量投資公司的重要指標，該比率越高表示股東所獲得較多的報酬，公式(28)如下:

**現金流量(Cash flow)**

16. 營業現金對流動負債比(Operating cash to current liabilities ratio)

營業現金流代表公司在經營過程中實際的現金流入與支出，紀錄在公司的現金流量表，是檢視公司營運的重要指標，而該比率為營業現金流除以流動負債，衡量每一元的流動負債有多少金額的營運現金流量來報障，公式(29)如下:

17. 營業現金對負債比(Operating cash to liabilities ratio)

與上述的營業現金對流動負債比差別在於該比率分母為負債，反映每一元的負債有多少金額的營運現金流量來報障，為衡量公司營運的重要指標，公式(30)如下:

18. 營業現金對稅後純益比(Operating cash to net earnings after tax ratio)

該比率的分母為稅後純益，稅後純益為應記基礎的會計方法記錄，無法反應公司實際的現金流向，因此該比率代表每一元的稅後淨利能收回多少現金，若大於100%代表該公司收回現金的能力較高，若為100%代表公司現金流量與稅後淨利一致，若小於50%代表有太多應收帳款未收回，可能導致公司的周轉問題，為衡量公司現金流量的重要指標，公式(31)如下:

**月營收比率(Monthly revenue ratio)**

19. 上月比較增減(%) (Month on month)

該比率為當月營收較上月營收增減幅度，公式(32)如下:

20. 去年同月增減(%)(Year on year)

該比率為當月營收較去年同月營收增減幅度，公式(33)如下:

21. 前期比較增減(%) (Quarter on quarter)

該比率為當月累計營收較去年累計營收增減幅度，公式(34)如下:

**4.4 深度學習模型(Deep learning model)**

**4.4.1 1DCNN**

1DCNN 為一維的卷積層，廣泛使用在時間序列資料集做資料特徵的萃取。

**4.4.2 空洞卷機(Dilation CONV1D)**

中文名稱為1維空洞卷積又稱1維擴張，空洞卷積是一般卷積層的衍生變化類型，以圖8二維空洞卷積為例，空洞為1，雖然kernal size為7\*7，但實際有數值與圖像卷積只有3\*3的點(圖中紅色的點)，其餘kernal的點為0，而這個卷積用3\*3的值卻可擴大到7\*7的視野，空洞卷積的好處在於不做pooling的條件下加大了視野和減少損失讓每個卷積都輸出更大範圍的資料訊息。

本研究使用三組一維卷積組，前面兩組分為為空洞卷積(空洞為2和4)，大小為(4,)的4個filter,最後一組為卷積為大小為(4,)的1個filter，paddig的方式採用same確保輸出特徵維度和輸入的大小相同，並透過三組卷積堆疊來萃取更多資料特徵訊息。

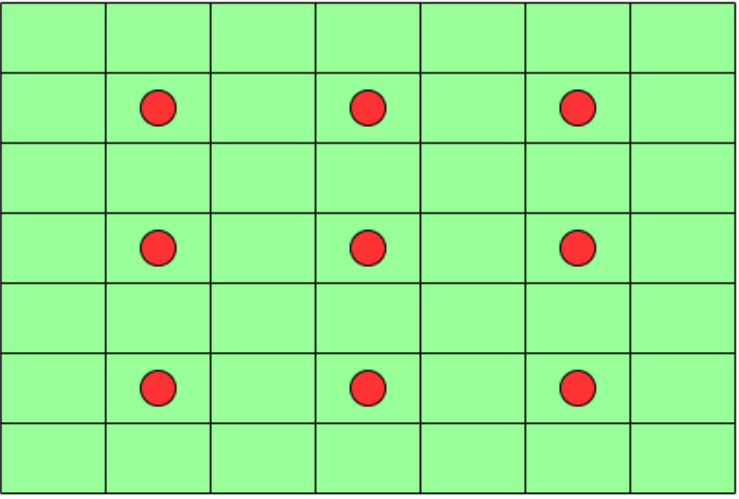


圖8 空洞卷積

**4.4.3 自我注意力機制(Self Attention)**

自我注意力機制(Self attention)為transformer模型裡的重要精隨，架構圖為圖五，此機制可以針對某些重要特徵更加關注且放大其權重，而對其他特徵少關注一點。

概念上，我們有一個描述對象的序列(query序列)，還有一個從中粹取資訊的value序列，每個value序列都被指配一個key序列並計算query和key的匹配程度，接著傳回value進行計算加權總和。

當輸入特徵向量，，，分別乘上與和得到, (query序列) 與，， (key序列)，，， (value序列),針對query中每個元素，計算該元素與每個key序列元素的關聯性強度，透過score公式(公式32)得到attention分數，再將並使用attention分數進行softmax轉換(公式33)，最後再對value序列進行加權總合，得到當前目標特徵注意力向量 (公式34)。

以舉例，為與做相加(additive) ，為與做additive,

為與做additive，然後再將、、經過softmax激活函數得到彼此相關性的attention score (α'1,1 、α'1,2 、α'1,3，其中α'i,j 代表第i個目標特徵對應特徵j的注意力權重，所有的注意力權重加總等於1，再將模型得到的attention score (α'1,1 、α'1,2 、α'1,3)跟，， (value序列)進行加權總合輸出目標特徵向量, 如果attention score數值越大，加權平均後的結果會偏向attention score數值大的v ，整體架構圖為圖9。

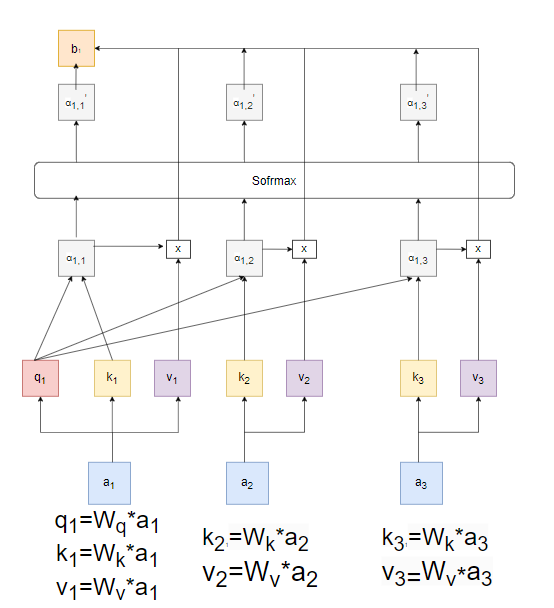
****

圖9 attention架構圖

score( , ) = tanh( + ) (32)

(33)

# 第五章 資料庫

本論文採用的資料庫資料來源為公開網站的數據，股價來源為YAHOO FINANCE網站所公布資料，而公開資訊觀測站為政府公布上市櫃公司財務資料的平台,因此本論文上市公司的季財務資訊和月營收資料皆從公開資訊觀測站擷取！

**5.1 YAHOO FINANCE**

YAHOO FINANCE為YAHOO公司所設立的網站，該網站提供許多財經資訊，包含國內外上市櫃公司的歷史成交價和新聞，本論文有兩個資料集，在四個股票資料集部分，主要擷取該網站的TSLA、MSFT、IBM、APPL的股價，TSLA資料期間為2010/6/29~2024/7/31、MSFT資料期間為1986/3/13~2024/7/31、IBM資料期間為1980/1/1~2024/7/31、APPL資料期間為1980/12/12~2024/7/31的每日股價(圖10、圖11、圖12、圖13) ，另外本論文StockQM資料集擷取國內上市公司資料期間為2013/01/01至2024/07/31每月的財務比率和包含2013年第一季到2024年第1季共44季的財務資料，以及財報發布後隔月月底的股價，圖14為以台泥為例的歷史股價。

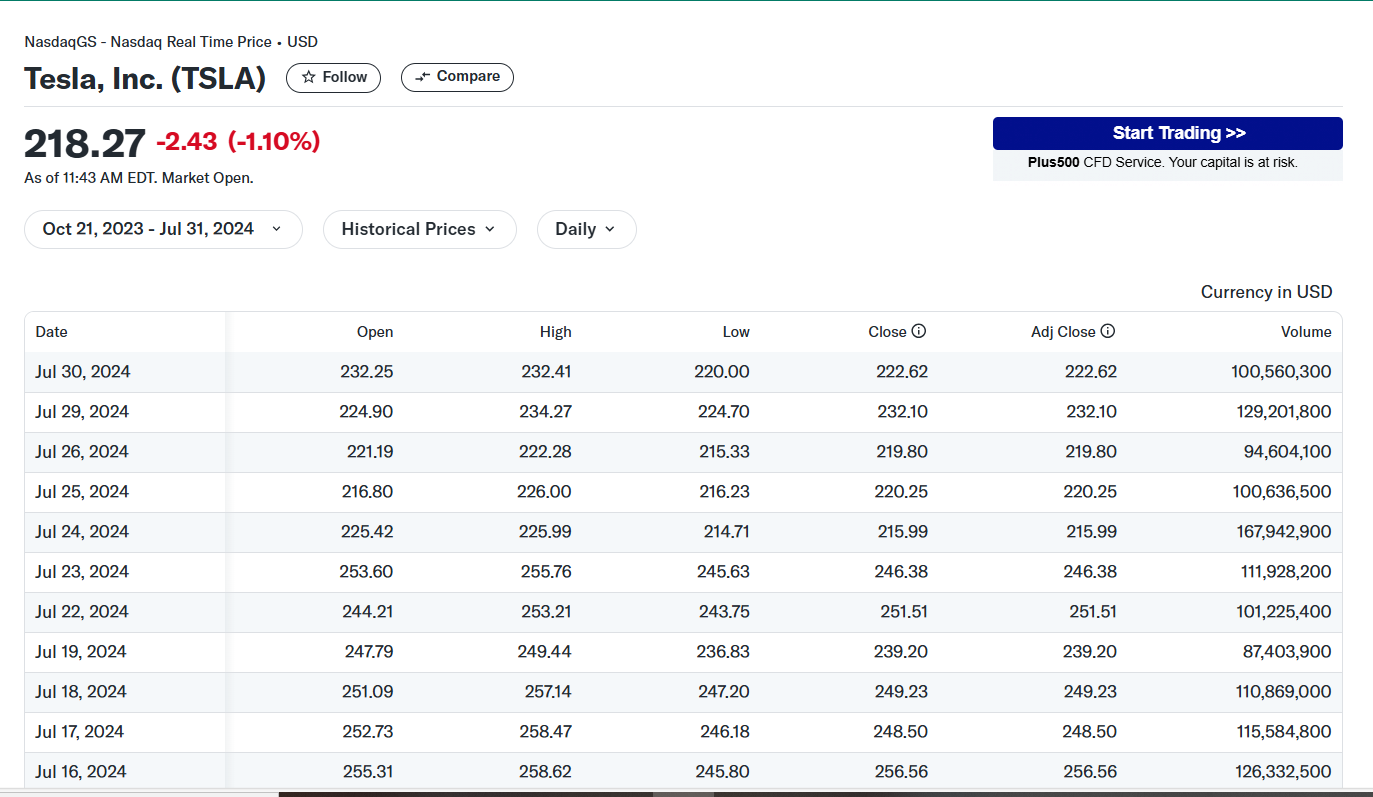


圖10 Yahoo Finance TSLA 歷史價格

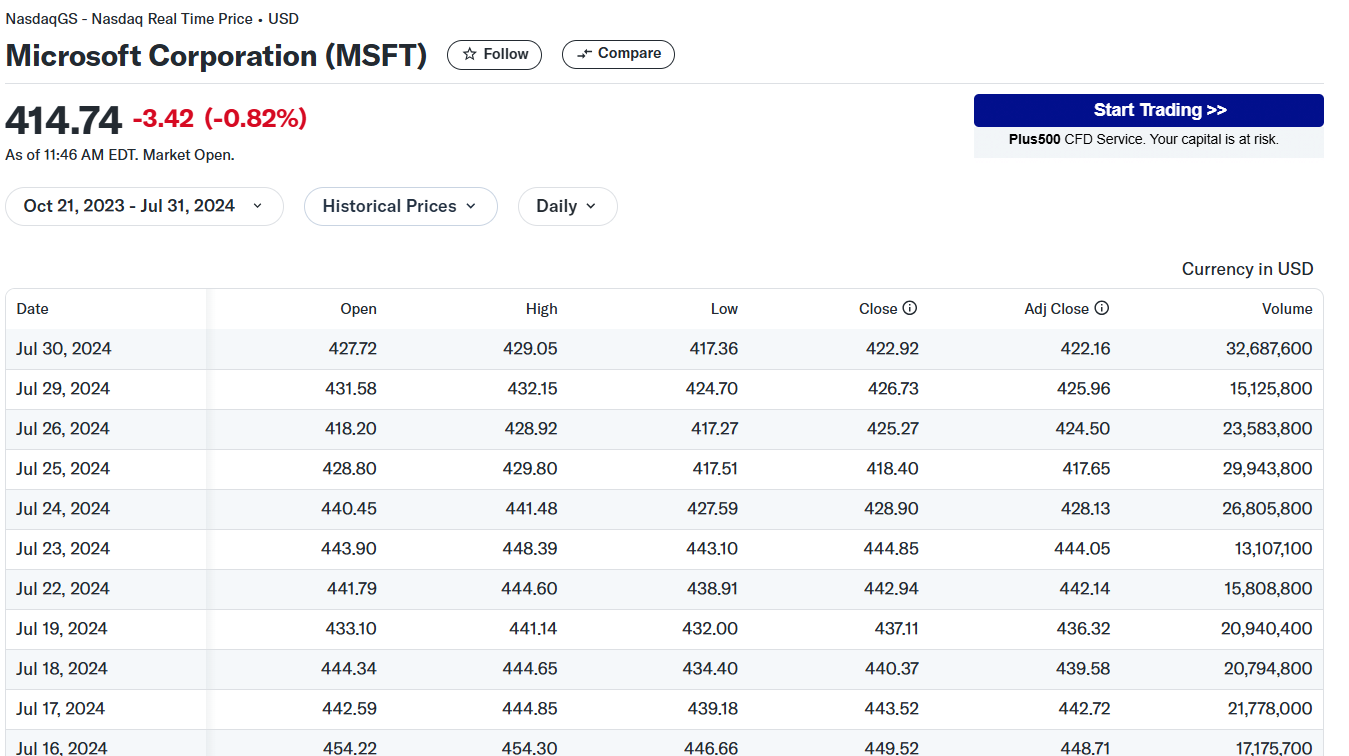


圖11Yahoo Finance MSFT 歷史價格

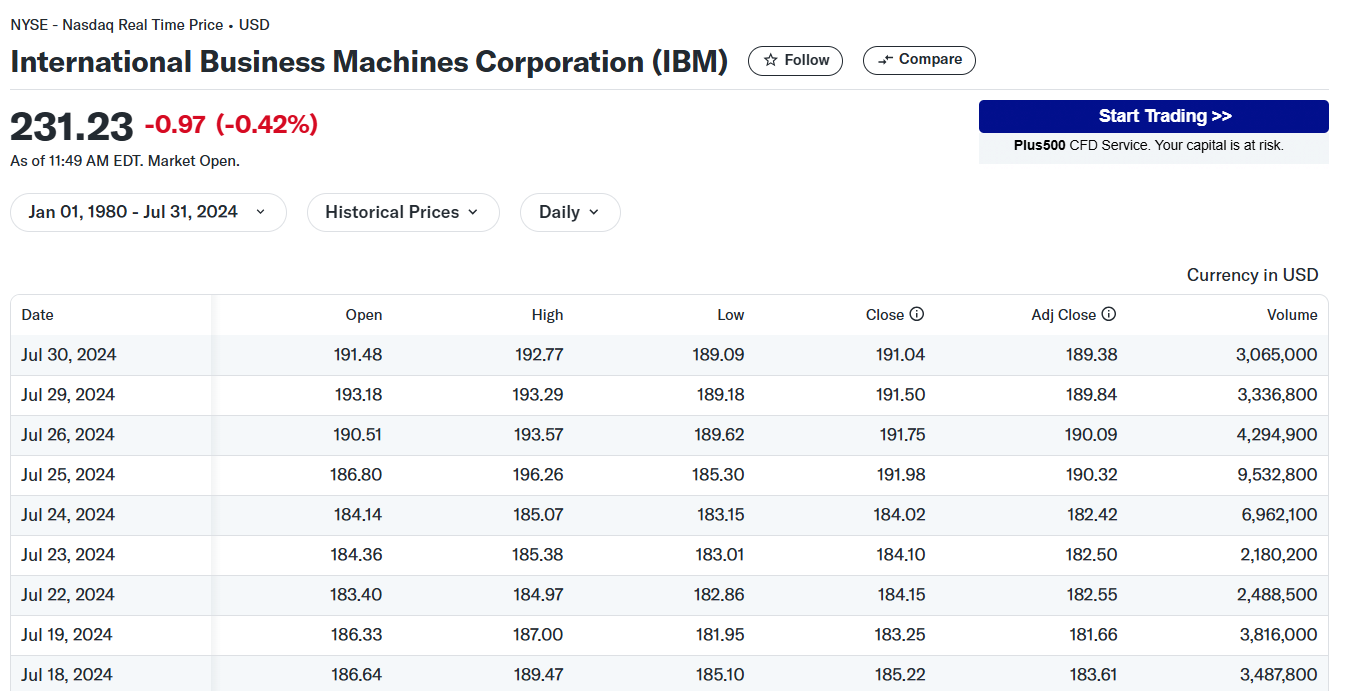
****

圖12Yahoo Finance IBM 歷史價格

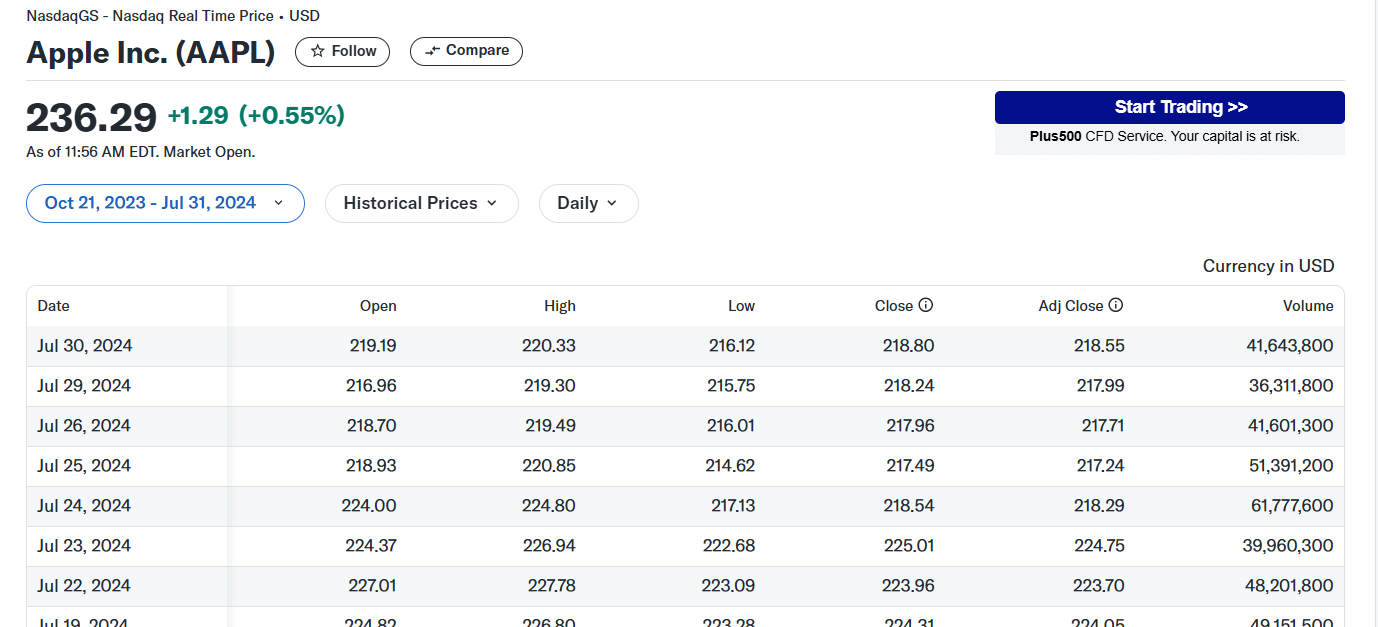


圖13Yahoo Finance AAPL 歷史價格

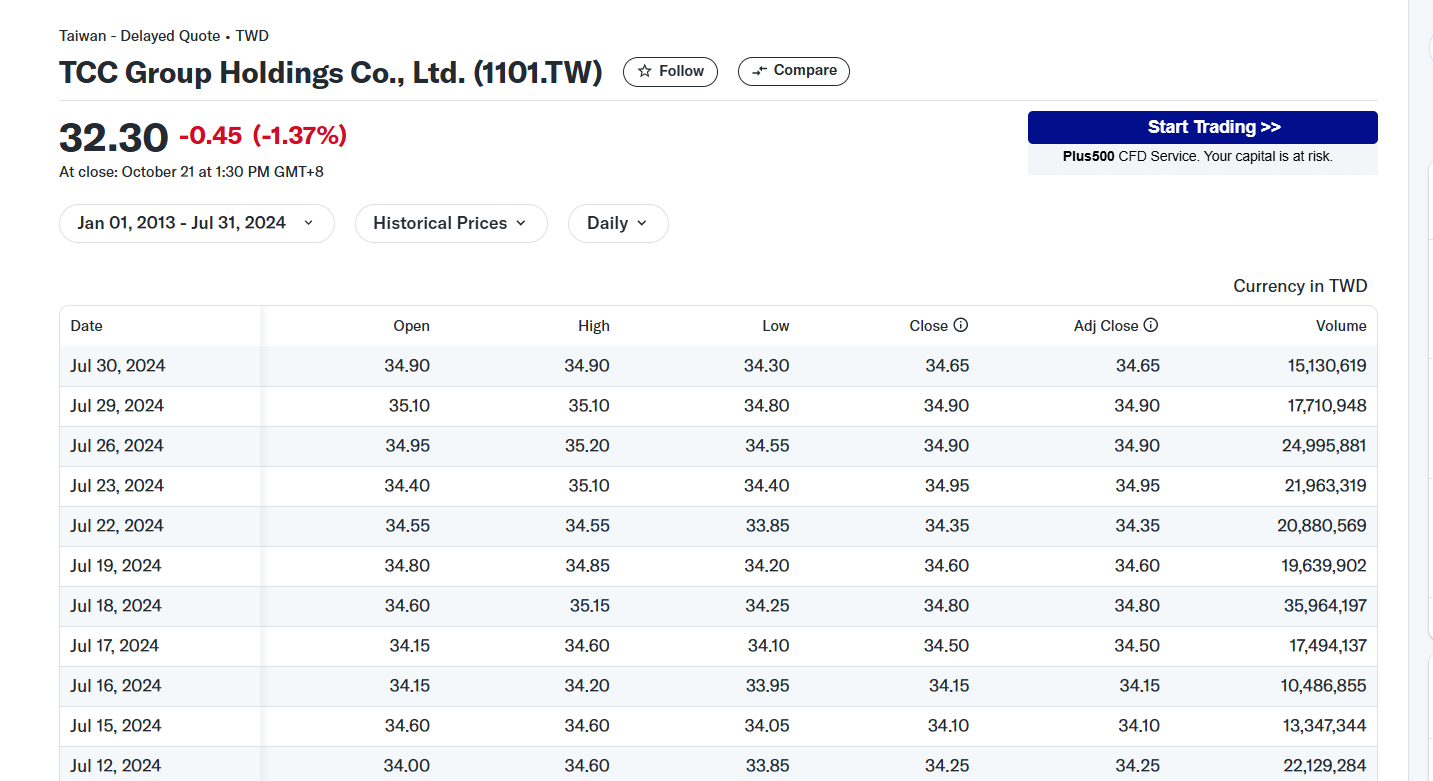
****

圖14Yahoo Finance 台泥 歷史價格

**5.2 公開資訊觀測站(Public information observatory)**

公開資訊觀測站為台灣證券交易所所創立的公開平台，提供投資人國內上市櫃公司的各種公開資訊，包括法說會、重大訊息和公司財務資訊的揭露等等相關訊息，本論文主要選取該網站提供的台灣上市公司季財務資料和月營收資料(如圖15圖16台泥為例)搭配自YAHOO FINANCE網站提供的股價歷史資料整合成創新的結合月營收和季財務資料的資料庫。

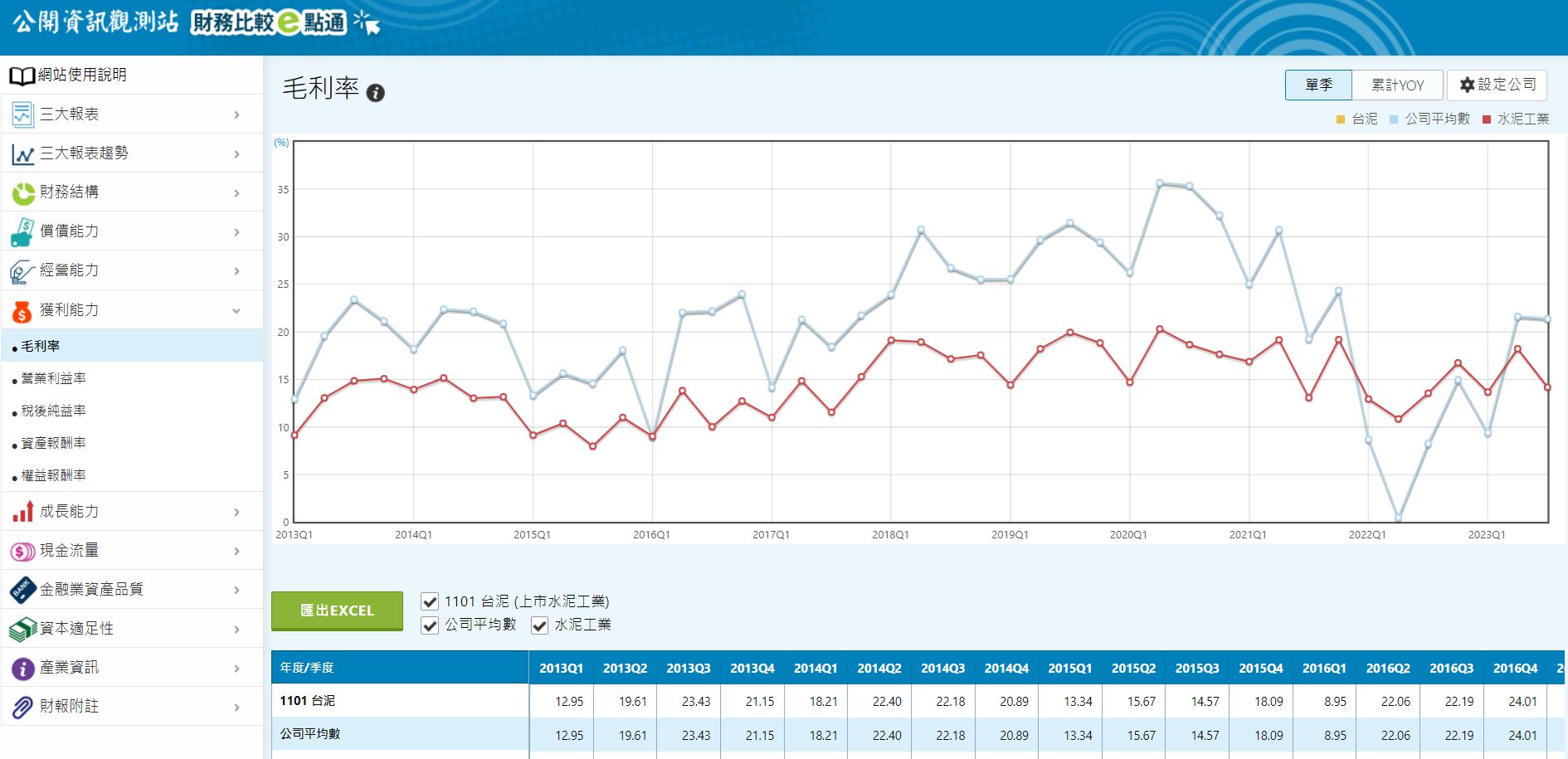


圖15台泥毛利率



圖16台泥月營收資料

# 第六章 實驗與討論

**6.1 實驗設置(Experimental setup)**

本論文實驗會在 Intel(R) CoreTM i7-6700HQ的CPU和Intel(R) HD Graphics Family 的GPU以及 8192MB 的 RAM 內存上來進行訓練。並使用 Keras 深度學習網絡的開發框架下使用Python3.8 語言實現。

資料庫的選擇，使用五大公司股票股價，包括TSLA資料期間為2010/6/29~2024/7/31、MSFT資料期間為1986/03/13~2024/7/31、IBM資料期間為1980/1/1~2024/7/31、APPL資料期間為1980/12/12~2024/7/31的每日股價以及自己創建的國內上市公司資料集，資料期間為2013/1/1~2024/7/31的每月和每季財務資料。

**6.2 損失函數(Loss function)**

本論文為採用21個變數預測短期股價報酬，屬於迴歸問題，一般迴歸採用MSE，公式為(35)，本論文預期制定融合MSE的創新損失函數來評估模型的效能。

其中為資料輸出值，為深度學習模型預測值，*N*為分批的資料量

**6.3實驗結果(Experimental results)**

**實驗一: 比較dropout比率的影響(Compare the impact of dropout ratios)**

本論文在模型encoder和decoder個數各一次條件下以dropout比率分別為10%、15%、20%、25%、30% 以及不加入dropout進行實驗，實驗結果如表3，紅色表示最好的成績，在各個資料集中，dropout比率為25%的MAE值均為最低，因此在接下來的實驗，本論文模型的dropout比率設為25%，另外在loss部分，各資料集選取dropout 25%的loss圖為圖17~圖21，不加dropout的loss圖為

圖22~圖26，可以顯示有加入dropout的驗證集validation的loss會隨epoch執行次數增加而顯著下降，而模型沒有dropout的loss圖的驗證集validation的loss不大會隨epoch執行而有顯著下降，代表加入dropout會降低模型的loss和減少MAE。

表3 比較dropout比率的影響

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | |
| 資料集 | dropout10% | dropout15% | dropout 20% | dropout 25% | dropout 30% | 不加dropout |
| AAPL | 15.52 | 22.09 | 28.93 | 14.74 | 49.82 | 30.01 |
| TSLA | 186.79 | 242.69 | 98.32 | 35.38 | 105.37 | 419.98 |
| MSFT | 43.25 | 69.96 | 11.67 | 7.46 | 12.82 | 123.57 |
| IBM | 4.08 | 2.76 | 3.89 | 2.51 | 2.83 | 34.52 |
| StockQM | 32.33 | 43.54 | 29.37 | 31.74 | 33.19 | 38.07 |

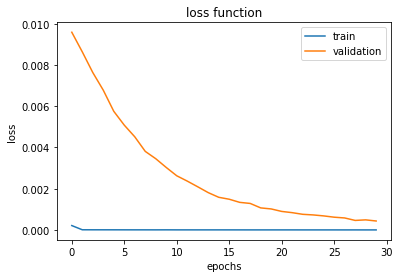
****

圖17 AAPL資料集 dropout 25% loss 圖

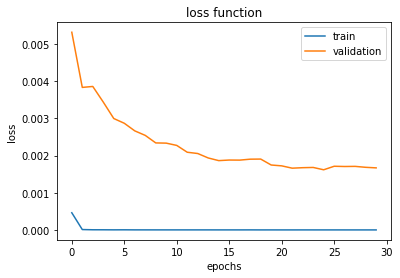
****

圖18 TSLA資料集 dropout 25% loss 圖

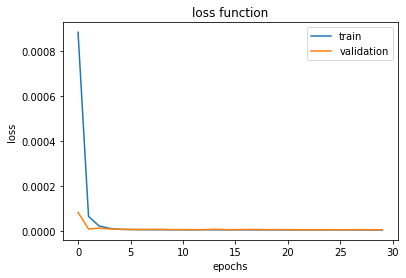
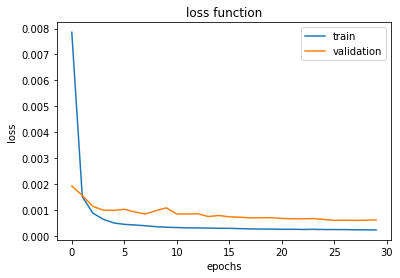


圖19 MSFT資料集 dropout 25% loss 圖

 圖20 IBM 資料集dropout 25% loss 圖

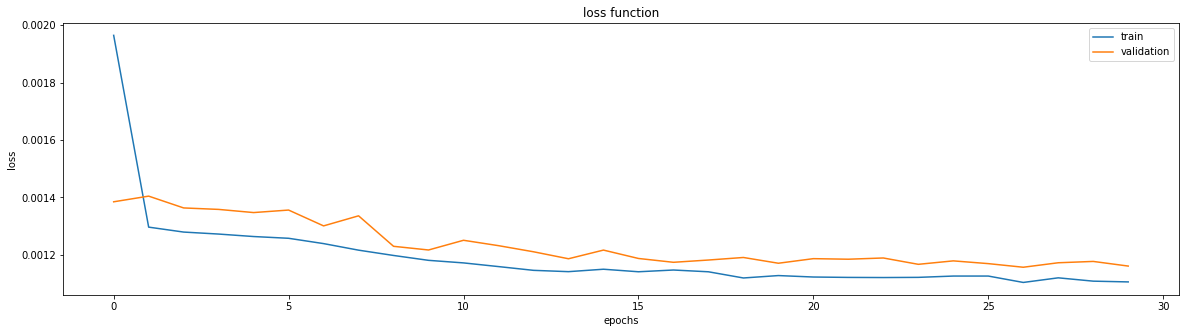


圖21 StockQM資料集 dropout 25% loss 圖

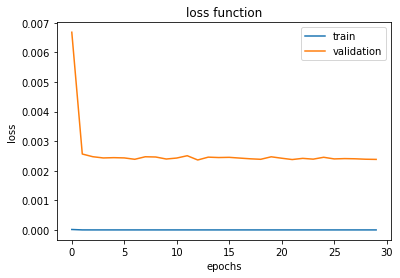
****

圖22 AAPL資料集 無dropout loss 圖

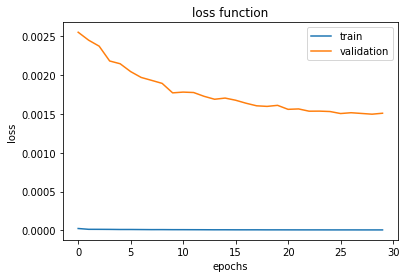


圖23 TSLA資料集 無dropout loss 圖

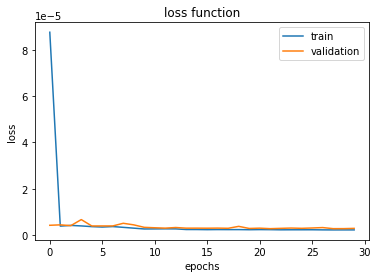
****

圖24 MSFT資料集 無dropout loss 圖

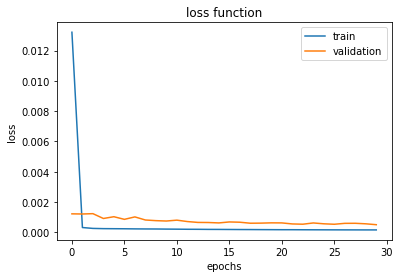


圖25 IBM資料集 無dropout loss 圖

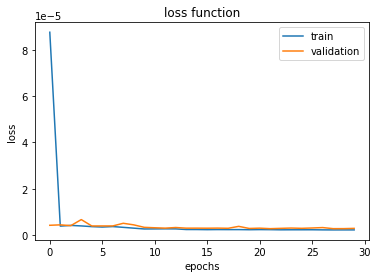
****

圖26 StockQM資料集 無dropout loss 圖

**實驗二: 比較有無使用dropout對模型的影響(Compare the impact on the model with and without using dropout)**

本論文的模型架構為圖6和圖7共有四個dropout,我們針對模型在encoder和decoder個數各一次不同dropout位置進行消融測試，dropout位置1代表架構圖中左上的dropout(紅1表示)，位置2為架構圖中左下的dropout(紅2表示)，位置3為架構圖中右上的dropout(紅3表示)，位置4為架構圖中右下的dropout(紅4表示)，實驗結果如表4,其中紅色代表最好的MAE成績，實驗顯示模型在dropout四個有最小的MAE達到較佳的預測結果。

表4 有無使用dropout對模型的影響

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 資  料  集 | MAE | | | | | | | | | | | | | |
| dropout1個(1) | dropout1個(2) | dropout1個(3) | dropout1個(4) | dropout2個(12) | dropout2個(13) | dropout2個(14) | dropout2個(34) | dropout3個(123) | dropout3個(134) | dropout3個(124) | dropout3個(234) | dropout4個(1234) | 沒有dropout |
| AAPL | 96.63 | 96.41 | 125.71 | 45.9 | 177.55 | 192.3 | 96.41 | 24.7 | 68.5 | 97.59 | 111.07 | 96.41 | 53.2 | 96.41 |
| TSLA | 309.25 | 237.7 | 125.02 | 45.3 | 215.6 | 237.7 | 2.54 | 32.3 | 81.62 | 95.05 | 237.7 | 237.7 | 65.8 | 237.7 |
| MSFT | 32.68 | 209.11 | 209.11 | 34.17 | 25.63 | 24.04 | 35.92 | 16.85 | 17.35 | 20.96 | 69.07 | 73.04 | 7.07 | 209.11 |
| IBM | 128.45 | 128.45 | 128.45 | 2.6 | 62.57 | 67.53 | 4.63 | 2.87 | 16.35 | 2.68 | 3.17 | 2.75 | 3.17 | 2.18 |
| Stock  QM | 29.87 | 28.82 | 31.67 | 36.41 | 30.03 | 30.71 | 33.59 | 37.37 | 32.56 | 31.78 | 30.32 | 30.72 | 29.73 | 30.53 |

**實驗三: 比較不同encoder和decoder個數的影響(Compare the impact of different numbers of encoders and decoders)**

比較模型使用不同encoder對decoder個數的影響，本論文對每個資料集設計ecoder個數(0到9次)與decoder個數(0到9次)共100次的訓練並得到訓練結果為表5 (AAPL資料集)、表6(TSLA資料集)、表7 (MSFT資料集) 、表8 (IBM資料集) 、表9(StockQM資料集)，在表5中，可以顯示出在APPL資料集中的

encoder個數為0的狀況下，decoder為5次得到的MAE值36.46會是最好，而在decoder個數為0的狀況下，decoder為3次得到的MAE值2.86會是最好，其中模型在encoder為4次和decoder為5次的範圍內MAE值分布較穩定，又以encoder為4次和decoder為5次表現最佳，MAE值為2.62 (紅色表示)。在表6中，可以發現在TSLA資料集， encoder個數為0的狀況下，decoder為5次得到的MAE值為17.31會是最好，而在decoder個數為0的狀況下，encoder為3次得到的MAE值16.04會是最好，其中模型在encoder為8次和decoder為2次的範圍內MAE值分布較穩定，又以encoder為7次和decoder為2次表現最佳，MAE值為16.34(紅色表示)。

在表7中，可以發現在MSFT資料集，encoder個數為0的狀況下,decoder為2次得到的MAE值7.55會是最好，而在decoder個數為0的狀況下，encoder為9次得到的MAE值6.09會是最好，其中模型在encoder為9次和decoder為5次的範圍內MAE值分布較穩定，又以encoder為6次和decoder為3次表現最佳，MAE值為5.24(紅色表示)。在表8中，可以發現在IBM資料集，encoder個數為0的狀況下，decoder為6次得到的MAE值3.51會是最好，而在decoder次個數為0的狀況下，encoder為4次得到的MAE值2.44會是最好，其中模型在encoder為9次和decoder為7次的範圍內MAE值分布較穩定，又以encoder為4次和decoder為4次表現最佳，MAE值為2.33。在表8中，可以發現在本論文StockQM資料集，encoder個數為0的狀況下，decoder為8次得到的MAE值29.45會是最好，而在decoder個數為0的狀況下，encoder為9次得到的MAE值29.53會是最好，其中模型在encoder為8次和decoder為9次的範圍內MAE值分布較穩定，又以encoder為5次和decoder為9次表現最佳，MAE值為28.11(紅色表示)。

表5 以AAPL資料集比較encoder和decoder個數對模型的影響

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | | | | | |
| encoder/decoder個數 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 52.85 | 53.34 | 132.95 | 125.52 | 42.26 | 36.46 | 137.66 | 91.97 | 191.19 | 200.74 |
| 1 | 6.13 | 28.93 | 38.01 | 48.37 | 154.71 | 11.28 | 43.62 | 21.51 | 162.68 | 402.89 |
| 2 | 12.11 | 11.48 | 84.26 | 17.22 | 141.35 | 141.69 | 185.26 | 42.73 | 159.68 | 233.28 |
| 3 | 2.86 | 3.23 | 40.79 | 14.22 | 197.22 | 409.27 | 68.95 | 58.77 | 165.68 | 307.32 |
| 4 | 81.05 | 5.86 | 89.19 | 61.88 | 46.31 | 2.62 | 449.56 | 819.89 | 152.68 | 337.35 |
| 5 | 116.28 | 152.09 | 85.82 | 25.22 | 44.94 | 407.98 | 553.42 | 511.91 | 141.68 | 131.31 |
| 6 | 172.72 | 15.75 | 8.63 | 177.44 | 164.33 | 257.29 | 158.58 | 165.28 | 125.68 | 114.66 |
| 7 | 2.96 | 140.04 | 192.08 | 158.62 | 173.17 | 36.98 | 57.37 | 542.55 | 136.68 | 480.09 |
| 8 | 73.75 | 150.27 | 17.23 | 94.67 | 167.02 | 145.42 | 298.47 | 415.04 | 139.68 | 176.82 |
| 9 | 67.83 | 371.15 | 6.92 | 59.15 | 44.02 | 438.98 | 266.83 | 172.58 | 118.68 | 26.52 |

表6 以TSLA資料集比較encoder和decoder個數對模型的影響

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | | | | | |
| encoder/decoder個數 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 25.45 | 87.14 | 18.44 | 59.96 | 139.54 | 17.31 | 76.79 | 208.61 | 287.04 | 140.07 |
| 1 | 30.91 | 35.38 | 55.25 | 28.88 | 297.71 | 370.18 | 105.69 | 490.99 | 493.24 | 321.88 |
| 2 | 18.87 | 19.38 | 27.39 | 356.11 | 40.08 | 489.16 | 279.73 | 257.32 | 873.27 | 195.77 |
| 3 | 16.04 | 31.88 | 124.48 | 112.62 | 985.69 | 147.95 | 255.44 | 647.96 | 1444.73 | 378.12 |
| 4 | 25.25 | 35.88 | 50.78 | 80.76 | 95.31 | 117.36 | 54.62 | 791.38 | 327.04 | 441.93 |
| 5 | 17.53 | 150.04 | 71.85 | 125.16 | 102.45 | 108.67 | 107.41 | 506.88 | 434.09 | 794.54 |
| 6 | 19.62 | 73.33 | 122.04 | 257.59 | 160.67 | 697.99 | 648.75 | 393.97 | 460.84 | 173.53 |
| 7 | 16.67 | 50.53 | 16.34 | 439.22 | 197.61 | 73.02 | 50.52 | 879.5 | 412.34 | 254.78 |
| 8 | 36.67 | 93.66 | 47.51 | 53.61 | 268.85 | 269.94 | 960.04 | 175.98 | 358.79 | 759.61 |
| 9 | 17.07 | 64.15 | 473.65 | 88.82 | 322.66 | 523.46 | 378.84 | 105.71 | 169.72 | 341.11 |

表7 以MSFT資料集比較encoder和decoder個數對模型的影響

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | | | | | |
| encoder/decoder個數 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 6.93 | 34.67 | 7.55 | 50.83 | 44.86 | 133.38 | 35.26 | 38.63 | 151.24 | 259.8 |
| 1 | 8.25 | 7.46 | 7.16 | 68.09 | 190.65 | 17.92 | 15.58 | 184.15 | 38.47 | 129.39 |
| 2 | 7.23 | 27.89 | 16.76 | 77.08 | 99.06 | 32.17 | 216.36 | 762.89 | 138.52 | 216.66 |
| 3 | 7.22 | 18.24 | 57.99 | 122.08 | 6.33 | 22.96 | 84.92 | 50.75 | 226.68 | 41.14 |
| 4 | 7.58 | 60.48 | 13.23 | 21.95 | 65.17 | 9.88 | 49.83 | 173.06 | 169.35 | 109.16 |
| 5 | 7.63 | 60.43 | 72.67 | 24.48 | 138.83 | 296.38 | 142.62 | 477.17 | 38.82 | 21.86 |
| 6 | 9.96 | 38.16 | 19.31 | 5.24 | 18.12 | 246.99 | 109.58 | 114.14 | 275.71 | 646.28 |
| 7 | 6.77 | 11.23 | 17.16 | 33.8 | 12.36 | 33.29 | 64.4 | 274.28 | 33.82 | 373.52 |
| 8 | 6.15 | 11.74 | 16.27 | 57.01 | 60.23 | 126.6 | 280.5 | 60.77 | 1131.42 | 66.17 |
| 9 | 6.09 | 6.43 | 37.75 | 69.21 | 159.25 | 217.44 | 104.62 | 244.57 | 148.85 | 112.68 |

表8 以IBM資料集比較encoder和decoder個數對模型的影響

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | | | | | |
| encoder/decoder個數 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 3.53 | 4.51 | 4.47 | 3.55 | 7.36 | 13.74 | 3.51 | 18.09 | 4.74 | 19.35 |
| 1 | 2.54 | 2.51 | 2.69 | 2.98 | 7.28 | 3.05 | 8.57 | 9.67 | 5.92 | 40.23 |
| 2 | 2.55 | 4.06 | 3.37 | 3.86 | 4.88 | 12.09 | 3.48 | 2.7 | 23.65 | 12.41 |
| 3 | 3.76 | 3.62 | 6.16 | 8.95 | 2.97 | 2.81 | 6.16 | 3.79 | 3.98 | 4.96 |
| 4 | 2.44 | 3.21 | 3.48 | 3.93 | 7.69 | 4.49 | 5.92 | 4.06 | 2.94 | 6.92 |
| 5 | 3.99 | 2.49 | 3.68 | 3.19 | 2.41 | 5.94 | 12.69 | 3.21 | 4.14 | 15.74 |
| 6 | 5.71 | 4.75 | 2.73 | 3.91 | 3.73 | 8.55 | 8.18 | 3.99 | 56.52 | 5.95 |
| 7 | 5.05 | 6.82 | 5.48 | 3.05 | 3.74 | 9.77 | 17.97 | 7.32 | 4.16 | 6.95 |
| 8 | 3.62 | 5.58 | 3.27 | 4.33 | 2.33 | 11.22 | 7.15 | 4.18 | 10.31 | 43.45 |
| 9 | 3.98 | 2.67 | 3.25 | 2.71 | 2.42 | 6.25 | 4.93 | 5.56 | 30.34 | 62.87 |

表9 以StockQM資料集比較encoder和decoder個數對模型的影響

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | | | | | |
| encoder/decoder個數 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 35.98 | 31.11 | 33.49 | 35.66 | 32.93 | 35.36 | 30.76 | 30.06 | 29.45 | 30.43 |
| 1 | 31.76 | 31.74 | 36.18 | 34.79 | 38.07 | 33.28 | 30.71 | 35.52 | 36.45 | 29.37 |
| 2 | 36.72 | 38.73 | 41.11 | 28.68 | 38.31 | 36.32 | 31.31 | 32.27 | 33.05 | 31.51 |
| 3 | 36.97 | 33.67 | 31.82 | 31.93 | 31.61 | 27.86 | 35.04 | 31.55 | 32.01 | 36.48 |
| 4 | 34.83 | 29.26 | 30.61 | 29.62 | 32.81 | 33.62 | 31.43 | 34.01 | 39.93 | 39.86 |
| 5 | 36.84 | 32.01 | 33.59 | 33.66 | 31.9 | 31.17 | 30.26 | 42.67 | 43.06 | 28.11 |
| 6 | 42.36 | 35.18 | 32.36 | 32.26 | 29.98 | 31.26 | 37.21 | 30.57 | 41.36 | 33.81 |
| 7 | 33.56 | 30.24 | 36.99 | 32.97 | 32.24 | 34.89 | 33.92 | 29.28 | 31.13 | 34.28 |
| 8 | 29.78 | 32.4 | 34.96 | 31.88 | 31.22 | 33.52 | 34.09 | 89.27 | 31.95 | 28.32 |
| 9 | 29.53 | 32.14 | 30.99 | 33.68 | 30.72 | 32.76 | 36.98 | 31.47 | 32.58 | 32.34 |

**實驗三: 比較不同激活函數的影響(Compare the impact of different activation functions)**

本論文以不同激活函數分別套用在五個資料集(AAPL、TSLA、MSFT、IBM、我自己資料集)得出得實驗結果於表10，紅色表示最佳成績，我們以實驗二結果得出每個資料集的最佳的encoder和decoder個數進行此次實驗個數的比較。實驗結果顯示使用relu激活函數(紅色部分)相較於tanh和sigmoid有較低的MAE，顯示模型使用relu會有較高的準確率，其中tanh值在AAPL資料集與relu有較大的差距下，其他資料集的MAE略大於relu,而在sigmoid激活函數部分，各個資料集的MAE均與relu和tanh激活函數有較大差距，故在模型的激活函數部分，我們選擇使用relu。

表10 激活函數對模型的影響 number of encoder

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| MAE | | | | | |
| 資料集 | relu | tanh | sigmoid | encoder | decoder個數 |
| AAPL | 2.62 | 68.25 | 93.96 | 4 | 5 |
| TSLA | 16.34 | 13.35 | 257.77 | 7 | 2 |
| MSFT | 5.24 | 9.18 | 175.93 | 6 | 3 |
| IBM | 2.33 | 5.74 | 45.92 | 8 | 4 |
| StockQM | 28.11 | 28.95 | 30.05 | 5 | 9 |

**實驗四: 比較加入decoder的影響(Compare the impact of adding decoder)**

此實驗比較模型有無decoder部分對MAE的影響，我們分別以五個資料集(AAPL、TSLA、MSFT、IBM、StockQM)比較有無decoder的模型結果，並選擇以實驗二結果得出每個資料集的最佳的encoder和decoder個數進行此次實驗個數的比較。實驗結果為表11，，紅色表示最佳成績，而我們比較的encoder和decoder個數基準為在實驗二得出各個資料集最小MAE的個數，資料顯示不加入decoder的MAE來的比有加入decode(紅色部份)大許多，顯示本論文模型decoder部分使用resnet網路，讓特徵在深層網路之間傳遞不會流失，有助於提升模型的準確度。

表11 模型加入decoder結果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MAE | | | | |
| 資料集 | 不加decoder | 加入decoder | encoder次數 | decoder次數 |
| AAPL | 96.38 | 2.62 | 4 | 5 |
| TSLA | 222.29 | 16.34 | 7 | 2 |
| MSFT | 86.53 | 5.24 | 6 | 3 |
| IBM | 15.32 | 2.33 | 8 | 4 |
| StockQM | 30.37 | 28.11 | 5 | 9 |

**實驗五:比較輸出層是否加入激活函數對模型的影響(Compare the impact of adding an activation function to the output layer on the model)**

在上述實驗中，輸出層是不加入激活函數，此實驗比較模型輸出層加入relu激活函數對模型預測的影響，本篇論文將五個資料集(AAPL、TSLA、MSFT、IBM、StockQM)套用於輸出層加入relu激活函數並與輸出層不加入relu激活函數進行MAE比較。模型的encoder和decoder次數設定為實驗二各個資料集最佳次數，實驗結果如表19，最佳成績以紅色表示，可以發現在五個資料集，輸出層不加入relu激活函數的MAE都會比加入relu激活函數低，這是因為模型預測的股價為線性輸出，故輸出層不使用激活函數效果會更佳，實驗結果如表12。

表12 比較輸出層是否加入激活函數的結果

|  |  |  |
| --- | --- | --- |
| MAE | | |
| 資料集 | 輸出層加入relu | 輸出層不加入relu |
| AAPL | 25.31 | 2.62 |
| TSLA | 29.53 | 16.34 |
| MSFT | 4.43 | 5.24 |
| IBM | 3.72 | 2.33 |
| StockQM | 39.26 | 28.11 |

**實驗六: 比較不同**time\_step**對預測結果的影響(Compare the impact of different time\_steps on prediction results)**

本論文在四個股票資料集(AAPL、TSLA、MSFT、IBM)使用的time\_step為10天，本實驗測試在四個股票資料集中不同time\_step對模型的預測表現，我們將time\_step分別設為5天、10天、15天、20天、25天，套用本論文提出的模型架構，而encoder及decoder次數使用10天最佳次數(紅色表示)，實驗結果為表13，由實驗結果得出四個資料集中time\_step為10天的MAE值均為最低，顯示出前10天股價已有非常充足訊息可用來做股票預測。

表13 比較不同time\_step的影響

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | | |
| 資料集 | time\_step  5天 | 10天 | 15天 | 20天 | 25天 | encoder次數 | decoder次數 |
| AAPL | 48.18 | 2.62 | 72.94 | 6.76 | 106.67 | 4 | 5 |
| TSLA | 69.53 | 16.34 | 120.35 | 55.43 | 37.14 | 7 | 2 |
| MSFT | 95.98 | 5.24 | 6.42 | 15.26 | 34.82 | 6 | 3 |
| IBM | 5.15 | 2.33 | 2.88 | 5.46 | 2.76 | 8 | 4 |

**實驗七: 財務資訊對模型的影響(The impact of financial information on the model)**

由實驗六顯示出time\_step設為10天在四個資料集對模型的預測結果最佳，本實驗將自創資料集StockQM的405間公司的time\_step設為10天，也就是只使用前10天股價預測第11天股價而不加入財務資訊來預測股價,資料期間為2013/07/01~2024/07/01並與StockQM資料集中使用財務資料預測股價相比，實驗結果為表14，紅色是表示最佳成績，由實驗結果顯示，有加入財務資料的StockQM資料集套用模型價購得出的MAE值(紅色表示)比單純使用前10天股價預測11天股價的MAE來的小，顯示出財務資訊對模型有正向的影響。

表14 比較財務資訊對模型的影響

|  |  |  |
| --- | --- | --- |
| 資料集 | 10天MAE | 套用財務資料MAE |
| StockQM | 28.65 | 26.51 |

**實驗八:季財報公布月份對模型的影響(The impact of quarterly financial report release month on the model)**

自創資料集StockQM中，由4.2章節中的表1顯示出季財報公布當月份為T,季財報公布隔月份為T+1,以此類推，T月份資料適用月份為3、5、8、11月，T-1為4、6、9、12月，T-2為5、7、10、1月，T-3只有2月，本實驗想探討不同的時間間隔對模型預測的影響，因此把StockQM資料集中405間公司分成T、T-1、T-2、T-3進行比較，比較結果為表15，紅色表示最好成績，舉例來說當台泥第四季財報在3/31公布，我們可在4月使用3/31公布的第四季財報比率與4/10前公布的3月營收比率來預測4/30股價，此時4/30股價預測設定為T。而在5/31股價預測上面只能用最新的季財報資訊為第四季財報比率與5/10前公布的4月營收比率來預測5/31股價，以此類推。由實驗結果顯示模型預測MAE會隨著時間間隔擴大而增加，顯示出離季報公布越近的月份搭配公布的月營收資料會有較佳的預測結果(MAE較小)，也就是在4/30(T)股價預測效果會比5/30(T-1)預測效果來的好，因為第四季報公布時間離4月最近，價格反應較明顯。

表15 比較季財報公布月份對模型的影響

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 特徵 | T | T-1 | T-2 | T-3 |
| MAE | 27.04 | 28.38 | 28.41 | 28.81 |

**實驗八: 在Transformer模型Transformer\_TCN模型下比較不同encoder和decoder個數的影響(Compare the impact of different numbers of encoders and decoders under the Transformer model Transformer\_TCN model)**

本論文的比較模型有包含TRANSFORMER和TRANSFORMER\_TCN，此兩個模型均有encoder和decoder,此實驗對兩個模型在encoder、decoder個數我們針對每個資料集如實驗二中各做10次因此每個資料集會在TRANSFORMER、TRANSFORMER\_TCN模型下各有100個實驗結果(如表16~表25)。

在TRANSFORMER模型下的AAPL資料集，實驗結果為表16，模型在encoder為3次和decoder為4次的範圍內MAE值分布較穩定，又以encoder為2次和decoder為1次MAE值為5.66表現最佳。在TSLA資料集，實驗結果為表17，模型在encoder為7次和decoder為6次的範圍內MAE值分布較穩定，又以encoder為7次和decoder為6次MAE值為18.36表現最佳(紅色表示)。在TSLA資料集，實驗結果為表18，模型在encoder為7次和decoder為6次的範圍內MAE值分布較穩定，又以encoder為7次和decoder為6次MAE值為18.36表現最佳(紅色表示)。在MSFT資料集，實驗結果為表19，模型在encoder為8次和decoder為9次的範圍內MAE值分布較穩定，又以encoder為6次和decoder為2次MAE值為5.33表現最佳(紅色表示)。在IBM資料集實驗結果為表20，模型在encoder為8次和decoder為9次的範圍內MAE值分布較穩定，又以encoder為4次和decoder為3次MAE值為3.02表現最佳(紅色表示)。在StockQM資料集實驗結果為表21，模型在encoder為2次和decoder為8次的範圍內MAE值分布較穩定，又以encoder為4次和decoder為8次MAE值為29.73表現最佳(紅色表示)。

在TRANSFORMER\_TCN模型下的AAPL資料集，實驗結果為表22，模型在encoder為6次和decoder為2次的範圍內MAE值分布較穩定，又以encoder為6次和decoder為1次MAE值為25.11表現最佳(紅色表示)。在TSLA資料集，實驗結果為表23，模型在encoder為4次和decoder為1次的範圍內MAE值分布較穩定，又以encoder為4次和decoder為0次MAE值為30.51表現最佳(紅色表示)。在MSFT資料集實驗結果為表24，模型在encoder為8次和decoder為0次的範圍內MAE值分布較穩定，又以encoder為7次和decoder為0次MAE值為7.25表現最佳。在IBM資料集實驗結果為表18，模型在encoder為8次和decoder為3次的範圍內MAE值分布較穩定，又以encoder為4次和decoder為1次MAE值為11.56表現最佳(紅色表示)。在StockQM資料集實驗結果為表25，模型在encoder為6次和decoder為8次的範圍內MAE值分布較穩定，又以encoder為2次和decoder為2次MAE值為28.19表現最佳(紅色表示)。

表16以AAPL資料集比較encoder和decoder個數對Transformer模型的影響

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | | | | | |
| encoder/decoder個數 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 8.85 | 9.79 | 5.73 | 12.08 | 83.1 | 65.4 | 169.7 | 142.03 | 67.75 | 35.86 |
| 1 | 9.16 | 17.61 | 5.66 | 15.04 | 68.56 | 38.63 | 20.24 | 15.17 | 66.99 | 224.4 |
| 2 | 9.33 | 53.21 | 8.57 | 29.6 | 38.34 | 126.49 | 54.37 | 145 | 198.91 | 24.43 |
| 3 | 168.61 | 39.95 | 60.84 | 22.68 | 14.98 | 47.84 | 71.19 | 16.36 | 131.28 | 106.64 |
| 4 | 168.71 | 40.83 | 31.39 | 8.69 | 168.17 | 192.65 | 26.49 | 39.51 | 133.23 | 119.57 |
| 5 | 12.76 | 151.77 | 131.47 | 88.2 | 114.34 | 47.93 | 21.18 | 127.97 | 9.36 | 87.52 |
| 6 | 99.87 | 55.38 | 206.49 | 87.9 | 164.61 | 53.96 | 73.84 | 52.98 | 156.54 | 11.87 |
| 7 | 7.26 | 11.56 | 100.29 | 61.86 | 341.46 | 45.04 | 169.73 | 32.76 | 164.25 | 62.34 |
| 8 | 122.21 | 184.46 | 14.85 | 139.59 | 28.79 | 159.73 | 209.13 | 44.11 | 146.43 | 180.78 |
| 9 | 8.98 | 22.99 | 135.11 | 262.47 | 31.32 | 45.77 | 74.82 | 123.15 | 40.72 | 65.81 |

表17以TSLA資料集比較encoder和decoder個數對Transformer模型的影響

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | | | | | |
| encoder/decoder個數 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 378.51 | 30.52 | 27.04 | 19.83 | 53.63 | 18.26 | 249.46 | 127.52 | 66.79 | 32.97 |
| 1 | 260.81 | 51.59 | 20.15 | 15.39 | 40.4 | 145.24 | 114.21 | 26.45 | 23.29 | 414.53 |
| 2 | 27.12 | 24.09 | 19.04 | 25.52 | 21.2 | 94.13 | 122.56 | 254.13 | 53.82 | 63.04 |
| 3 | 22.63 | 22.92 | 28.38 | 18.03 | 17.56 | 97.05 | 29.05 | 130.85 | 95.78 | 45.18 |
| 4 | 30.26 | 56.49 | 19.91 | 41.38 | 121.4 | 189.74 | 216.42 | 173.83 | 439.89 | 170.13 |
| 5 | 21.16 | 26.34 | 200.19 | 102.6 | 23.72 | 22.27 | 97.83 | 43.61 | 172.25 | 57.99 |
| 6 | 28.53 | 20.13 | 438.54 | 25.87 | 196.6 | 33.92 | 108.34 | 77.43 | 68.54 | 136.61 |
| 7 | 236.88 | 52.13 | 98.35 | 33.71 | 176 | 184.26 | 18.36 | 53.07 | 55.31 | 18.28 |
| 8 | 40.05 | 111.56 | 58.87 | 28.88 | 41.22 | 71.73 | 60.98 | 227.21 | 27.99 | 136.83 |
| 9 | 69.12 | 157.58 | 190.52 | 360.68 | 58.79 | 214.91 | 119.32 | 602.85 | 79.03 | 51.38 |

表18以MSFT資料集比較encoder和decoder個數對Transformer模型的影響

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | | | | | |
| encoder/decoder個數 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 7.02 | 4.53 | 8.13 | 8.46 | 7.23 | 6.56 | 5.4 | 6.93 | 6.97 | 5.56 |
| 1 | 9.15 | 6.27 | 7.37 | 7.31 | 7.51 | 5.13 | 5.45 | 6.86 | 9.38 | 8.47 |
| 2 | 5.87 | 8.2 | 10.17 | 6.67 | 6.65 | 7.78 | 5.29 | 7.79 | 8.15 | 6.89 |
| 3 | 5.46 | 7.82 | 5.66 | 5.43 | 5.49 | 7.85 | 6.08 | 11.99 | 5.79 | 8.39 |
| 4 | 5.75 | 5.87 | 8.88 | 5.63 | 5.53 | 7.71 | 5.09 | 9.74 | 6.42 | 7.95 |
| 5 | 10.11 | 7.01 | 5.94 | 6.05 | 12.14 | 5.07 | 8.24 | 5.73 | 6.78 | 6.7 |
| 6 | 5.62 | 6.71 | 5.33 | 10.18 | 4.82 | 14.67 | 5.66 | 5.82 | 18.01 | 7.95 |
| 7 | 5.71 | 6.09 | 6.48 | 7.14 | 11.07 | 5.61 | 12.35 | 6.08 | 5.62 | 5.04 |
| 8 | 8.16 | 6.63 | 6.43 | 8.62 | 5.34 | 5.45 | 10.64 | 6.48 | 5.73 | 5.16 |
| 9 | 10.22 | 8.55 | 6.18 | 8.34 | 7.52 | 6.82 | 52.9 | 5.29 | 5.23 | 5.06 |

表19以IBM比較encoder和decoder個數對Transformer模型的影響

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | | | | | |
| encoder/decoder個數 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 3.53 | 3.98 | 2.64 | 4.97 | 5.93 | 4.85 | 5.68 | 4.84 | 6.7 | 4.39 |
| 1 | 3.98 | 3.68 | 5.54 | 3.87 | 3.91 | 4.19 | 3.41 | 3.13 | 3.49 | 7.02 |
| 2 | 4.19 | 4.76 | 3.58 | 4.25 | 4.81 | 3.84 | 5.48 | 3.94 | 3.67 | 4.58 |
| 3 | 3.65 | 3.11 | 3.77 | 4.42 | 3.68 | 5.26 | 3.98 | 3.79 | 3.44 | 3.02 |
| 4 | 4.07 | 3.73 | 5.01 | 3.02 | 5.99 | 3.69 | 3.81 | 3.35 | 4.28 | 6.43 |
| 5 | 3.41 | 3.59 | 3.78 | 3.98 | 5.44 | 4.18 | 5.91 | 3.26 | 4.86 | 3.21 |
| 6 | 3.25 | 5.34 | 4.89 | 3.42 | 4.95 | 4.81 | 3.47 | 3.25 | 5.01 | 4.93 |
| 7 | 3.75 | 4.34 | 6.27 | 4.19 | 3.72 | 3.41 | 3.24 | 3.23 | 4.78 | 3.13 |
| 8 | 4.46 | 5.21 | 3.25 | 3.77 | 4.14 | 3.15 | 6.69 | 5.06 | 3.88 | 2.96 |
| 9 | 6.51 | 4.19 | 5.71 | 3.21 | 3.92 | 3.42 | 3.68 | 4.91 | 5.59 | 6.92 |

表20以StockQM集比較encoder和decoder個數對Transformer模型的影響

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | | | | | |
| encoder/decoder個數 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 31.48 | 33.39 | 32.85 | 31.4 | 36.98 | 41.59 | 37.66 | 39.38 | 41.23 | 34.67 |
| 1 | 31.35 | 32.38 | 39.71 | 40.29 | 32.11 | 28.78 | 42.95 | 31.71 | 30.13 | 33.81 |
| 2 | 34.59 | 46.81 | 38.88 | 40.2 | 39.03 | 30.78 | 39.65 | 35.25 | 39.13 | 38.57 |
| 3 | 31.84 | 35.62 | 38.59 | 36.09 | 30.95 | 44.88 | 47.83 | 31.43 | 37.73 | 32.92 |
| 4 | 34.37 | 44.21 | 30.85 | 34.98 | 36.24 | 38.03 | 57.57 | 31.15 | 41.36 | 42.83 |
| 5 | 39.74 | 34.82 | 56.07 | 40.39 | 36.62 | 31.36 | 44.07 | 36.53 | 33.67 | 30.99 |
| 6 | 29.73 | 31.64 | 29.92 | 39.56 | 35.52 | 34.62 | 36.46 | 35.94 | 31.24 | 32.07 |
| 7 | 31.73 | 40.97 | 30.97 | 29.28 | 37.05 | 31.53 | 40.25 | 31.54 | 36.63 | 35.43 |
| 8 | 41.16 | 32.32 | 38.84 | 39.29 | 36.86 | 30.15 | 34.73 | 29.93 | 32.95 | 34.61 |
| 9 | 31.84 | 37.05 | 32.82 | 36.93 | 28.46 | 36.75 | 37.33 | 35.68 | 37.85 | 32.52 |

表21以AAPL資料集比較encoder和decoder個數對Transformer\_TCN模型影響

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | | | | | |
| encoder/decoder個數 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 33.86 | 92.79 | 76.35 | 56.91 | 268.87 | 141.69 | 40.49 | 155.15 | 90.12 | 83.59 |
| 1 | 32.98 | 66.79 | 75.33 | 310.88 | 101.24 | 130.13 | 75.38 | 80.49 | 95.35 | 83.18 |
| 2 | 37.93 | 42.11 | 1030.1 | 87.4 | 97.47 | 87.47 | 145.23 | 101.69 | 86.05 | 94.57 |
| 3 | 77.16 | 2162.7 | 91.56 | 87.93 | 102.4 | 104.08 | 107.22 | 96.64 | 91.01 | 107.66 |
| 4 | 13.38 | 101.33 | 112.56 | 94.42 | 97.03 | 90.01 | 97.54 | 96.62 | 104.1 | 788.54 |
| 5 | 37.59 | 401.94 | 112.23 | 100.9 | 89.15 | 85.41 | 96.73 | 135.61 | 106.6 | 103.95 |
| 6 | 185.02 | 25.11 | 154.45 | 86.75 | 74.28 | 107.45 | 97.57 | 106.24 | 88.17 | 66.23 |
| 7 | 161.65 | 250.84 | 103.43 | 104.14 | 108.29 | 73.79 | 61.74 | 102.9 | 110.84 | 132.54 |
| 8 | 118.25 | 827.19 | 139.69 | 102.79 | 85.12 | 87.69 | 69.3 | 86.07 | 85.27 | 71.71 |
| 9 | 117.24 | 95.84 | 97.63 | 81.63 | 105.63 | 84.38 | 95.14 | 98.97 | 84.08 | 82.24 |

表22以TSLA資料集比較encoder和decoder個數對Transformer\_TCN模型影響

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | | | | | |
| encoder/decoder個數 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 39.92 | 270.39 | 459.03 | 83.09 | 39.31 | 80.95 | 126.21 | 205.59 | 253.86 | 315.72 |
| 1 | 36.55 | 244.37 | 271.03 | 158.37 | 248.47 | 240.65 | 215.73 | 229.62 | 235.35 | 195.56 |
| 2 | 38.42 | 90.71 | 393.46 | 757.52 | 412.91 | 167.16 | 218.97 | 198.68 | 266.88 | 246.89 |
| 3 | 48.74 | 808.21 | 76.88 | 364.73 | 222.35 | 197.07 | 237.85 | 189.54 | 240.62 | 165.17 |
| 4 | 30.51 | 436.14 | 242.22 | 252.45 | 169.17 | 223.17 | 188.22 | 141.39 | 235.94 | 194.48 |
| 5 | 142.37 | 75.57 | 335.89 | 70.52 | 311.11 | 300.52 | 231.43 | 260.18 | 263.32 | 189.41 |
| 6 | 113.15 | 70.82 | 223.72 | 225.89 | 242.27 | 53.02 | 181.07 | 178.13 | 262.15 | 270.24 |
| 7 | 65.07 | 100.06 | 232.68 | 208.77 | 224.47 | 222.8 | 45.04 | 220.34 | 230.97 | 240.56 |
| 8 | 35.47 | 93.55 | 2952.35 | 279.45 | 130.73 | 172.84 | 229.86 | 188.34 | 327.39 | 173.87 |
| 9 | 38.41 | 69.11 | 230.6 | 246.28 | 249.13 | 221.93 | 169.29 | 252.52 | 209.55 | 216.53 |

表23以MSFT資料集比較encoder和decoder個數對Transformer\_TCN模型影響

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | | | | | |
| encoder/decoder個數 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 17.78 | 229.26 | 290.46 | 793.94 | 166.96 | 194.91 | 185.78 | 161.69 | 200.17 | 160.87 |
| 1 | 9.15 | 125.86 | 223.39 | 175.59 | 174.04 | 412.86 | 255.81 | 204.33 | 230.63 | 185.89 |
| 2 | 8.63 | 196.57 | 448.09 | 254.27 | 244.98 | 180.82 | 129.19 | 176.22 | 159.65 | 311.62 |
| 3 | 5.37 | 295.76 | 535.1 | 181.36 | 223.32 | 212.97 | 181.42 | 140.4 | 142.11 | 217.09 |
| 4 | 7.31 | 145.83 | 196.21 | 108.54 | 194.29 | 212.63 | 142.04 | 183.43 | 194.27 | 140.57 |
| 5 | 6.72 | 168.44 | 185.49 | 149.67 | 160.42 | 206.27 | 190.53 | 191.03 | 285.22 | 206.59 |
| 6 | 7.85 | 162.35 | 193.04 | 180.38 | 158.05 | 148.38 | 196.13 | 222.56 | 158.41 | 168.34 |
| 7 | 7.25 | 231.94 | 223.91 | 205.28 | 245.87 | 120.81 | 211.13 | 120.13 | 149.29 | 194.19 |
| 8 | 15.91 | 147.54 | 318.29 | 221.59 | 189.82 | 192.01 | 153.94 | 149.91 | 189.34 | 129.54 |
| 9 | 16.31 | 233.85 | 173.21 | 190.74 | 432.02 | 179.23 | 234.59 | 174.96 | 182.83 | 260.02 |

表24以IBM資料集比較encoder和decoder個數對Transformer\_TCN模型影響

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | | | | | |
| encoder/decoder個數 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 24.28 | 17.04 | 54.95 | 59.54 | 48.63 | 87.46 | 104.39 | 73.78 | 90.64 | 132.81 |
| 1 | 23.02 | 22.31 | 80.15 | 16.12 | 61.19 | 95.52 | 128.46 | 84.23 | 94.25 | 92.82 |
| 2 | 13.23 | 29.07 | 29.11 | 102.76 | 131.24 | 112.72 | 89.43 | 86.08 | 93.78 | 122.95 |
| 3 | 12.69 | 15.56 | 15.59 | 69.97 | 95.25 | 70.68 | 105.84 | 109.36 | 89.96 | 93.38 |
| 4 | 14.29 | 11.56 | 35.28 | 78.12 | 95.41 | 99.57 | 92.63 | 90.78 | 94.87 | 89.79 |
| 5 | 23.35 | 14.54 | 14.28 | 24.07 | 21.28 | 89.32 | 93.47 | 94.85 | 108.77 | 103.83 |
| 6 | 23.83 | 16.14 | 29.19 | 51.89 | 118.82 | 108.15 | 55.39 | 41.79 | 94.5 | 109.62 |
| 7 | 23.82 | 16.99 | 42.04 | 80.87 | 100.28 | 83.49 | 102.22 | 70.83 | 47.52 | 91.62 |
| 8 | 19.76 | 10.73 | 73.06 | 44.7 | 125.9 | 89.89 | 89.41 | 140.02 | 113.76 | 93.57 |
| 9 | 15.87 | 11.77 | 21.37 | 82.31 | 51.25 | 96.57 | 91.92 | 113.99 | 92.35 | 126.06 |

表25以StockQM資料集比較encoder和decoder個數對Transformer\_TCN模型影響

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | | | | | |
| encoder/decoder個數 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 29.46 | 28.59 | 35.29 | 30.84 | 28.71 | 28.09 | 28.41 | 29.75 | 29.37 | 29.24 |
| 1 | 29.59 | 28.47 | 30.51 | 28.74 | 29.59 | 28.24 | 28.92 | 26.56 | 29.06 | 29.86 |
| 2 | 28.87 | 28.81 | 28.19 | 30.57 | 29.24 | 30.34 | 29.04 | 29.49 | 30.38 | 28.57 |
| 3 | 29.73 | 28.98 | 30.67 | 30.67 | 28.37 | 28.56 | 29.64 | 29.75 | 29.27 | 29.93 |
| 4 | 28.54 | 29.38 | 28.59 | 28.37 | 29.09 | 28.85 | 28.42 | 29.33 | 29.37 | 28.93 |
| 5 | 28.99 | 28.56 | 28.55 | 30.95 | 30.87 | 28.25 | 30.77 | 29.26 | 28.58 | 28.62 |
| 6 | 29.13 | 29.52 | 30.76 | 28.85 | 29.03 | 29.53 | 30.58 | 28.07 | 29.15 | 29.83 |
| 7 | 28.27 | 28.41 | 29.04 | 28.29 | 31.57 | 28.32 | 27.81 | 29.67 | 28.29 | 28.92 |
| 8 | 29.74 | 28.92 | 28.73 | 29.94 | 29.75 | 29.85 | 29.51 | 29.35 | 28.39 | 29.12 |
| 9 | 29.19 | 29.63 | 29.58 | 29.71 | 28.48 | 29.07 | 29.18 | 28.84 | 28.82 | 29.52 |

**實驗九: 與其他模型比較(Compare with other models)**

本論文以我們的模型與其他五種混合模型進行比較，包含WGAN\_GP[18]、CNN\_LSTM[21]、CNN\_BILSTM\_ATTENTION[19]、TRANSFORMER[17] 、TRANSFORMER\_TCN[20]，實驗結果如表26，紅色表示最好成績,其中TRANSFORMER、TRANSFORMER\_TCN模型本論文選定以實驗八的最佳化encoder、decoder個數來比較。表26顯示各個資料集在WGAN\_GP模型得出的MAE最大，代表該模型在五個資料集預測效果較差，而TRANSFORMER、TRANSFORMER\_TCN模型的MAE來的比WGAN\_GP、CNN\_LSTM、CNN\_BILSTM\_ATTENTION小許多，代表加入注意力機制加強對重要數據的關注史的預測效果提升，而本論文提議的模型MAE最小，表現優於其他模型。

表26 五個資料集與其他模型比較

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| MAE | | | | | | |
| 資料集 | WGAN\_GP  [18] | CNN\_LSTM  [21] | CNN\_BILSTM  \_ATTENTION  [19] | TRANSFORMER  [17] | TRANSFORMER  \_TCN [20] | OURS |
| AAPL | 96.16 | 58.77 | 27.57 | 5.66 | 25.11 | 2.62 |
| TSLA | 236.89 | 45.74 | 37.69 | 18.36 | 30.51 | 16.34 |
| MSFT | 207.87 | 35.62 | 35.56 | 5.33 | 7.25 | 5.24 |
| IBM | 128.47 | 20.46 | 17.22 | 3.02 | 11.56 | 2.33 |
| StockQM | 32.81 | 31.68 | 33.97 | 29.73 | 28.19 | 28.11 |

**實驗十:使用提議的模型，在四種資料集的預測表現(Predictive performance on four data sets using the proposed model)**

本論文使用提議的模型套用在四個資料集(AAPL、TSLA、MSFT、IBM)。模型預測表現如(圖27~圖30)藍色曲線是真實值，紅色為預測值，可以發現在各個資料集中，我們的預測值相當貼近真實值，代表模型的預測準確度高。

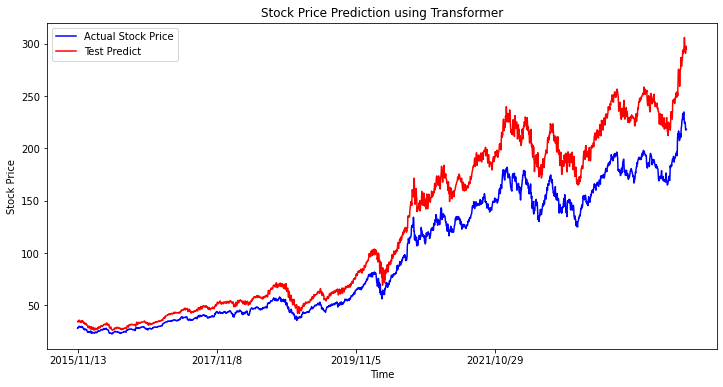


圖27用提議的模型，在AAPL的表現

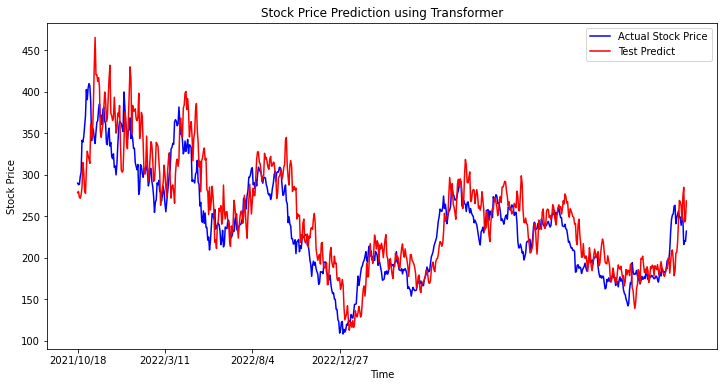
****

圖28用提議的模型，在TSLA的表現

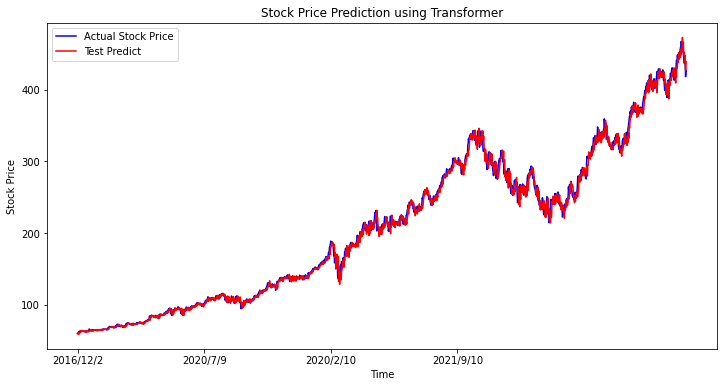
****

圖29使用提議的模型，在MSFT的表現

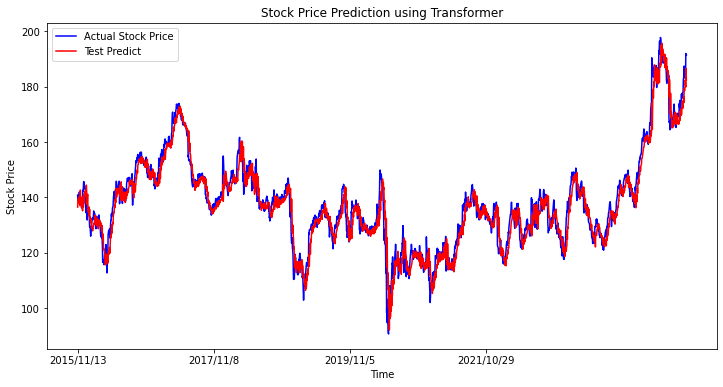


圖30使用提議的模型，在IBM的表現

**實驗十一:使用提議的模型，在近期上市公司預測的表現**

本論文所提議的模型的優點為只需少許的歷史資料即可預測公司股價，

因此特別挑選五家2023年才上市的公司股票，分別為台灣虎航、龍德造船、華懋、長榮航太、聯寶，使用StockQM資料集2024/7/31之前的資料並輸入2024年第二季季財務資料和8月月營收資料來預測9/30的股價，最後得出五家公司預測股價如表27，由結果顯示出預測效果跟實際股價差距不大。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 台灣虎航 | 龍德造船 | 華懋 | 長榮航太 | 天虹 |
| 9/30預測股價 | 52.6 | 119.6 | 157.57 | 92.2 | 275.72 |
| 9/30股價 | 59.3 | 123 | 168.5 | 97.7 | 283.5 |

表27近期上市公司股價預測

# 第七章 結論

本論文提出基於transformer模型改良的深度學習模型，結合了注意力機制對重要特徵的加強關注，以及空洞捲積捕捉更大範圍的資料特徵，並將各分支的資訊做相加彙整，再進入殘差神經網路(Resnet)，讓特徵在深層網路之間傳遞不會流失，而由實驗結果顯示出我們提出的模型優於其他模型方法，代表我們模型在股票預測方面是相當有效的。

本論文為了探討重大財務事件發布後的股價走勢，因此創建了結合混合頻率的季財務資料和月財務資料庫StockQM，並將提出的模型套用此資料庫和其他四間美國大型上市公司股票資料集，而由實驗結果顯示以混合頻率的財務資料擷取特徵搭配本論文提出的模型可有效的預測股價，也證實財務資訊的發布顯著影響股價未來一段時間的走勢，另外今後若有一個新上市公司需要預測未來股價，即可套用本論文使用StockQM資料庫訓練的參數搭配新上市公司最新公布的月和季財務資訊即可預測股價，而非在以往股票預測部分需要該股票過往一段時間的資料來預測股價。

由於混合頻率資料的研究相對較少，我們日後也會繼續朝此方向鑽研，期望能結合更多的金融知識和深度學習網路架構來提升股價預測的準確率。

# 第八章 參考文獻

[1] Ball, Ray, and Philip Brown. "Ball and Brown (1968) after fifty years." Pacific-Basin Finance Journal 53 (2019): 410-431.

[2] Doshi, Akash, et al. "Deep stock predictions." arXiv preprint arXiv:2006.04992 (2020).

[3] Lin, H., et al. "Stock price prediction using generative adversarial networks." Journal of Computer Science 17.3 (2021): 188-196.

[4] Das, Sourav, and Bharti Nagpal. "Stock Price Prediction Using Time Embedding and Attention Mechanism in Transformers." 2023 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI). IEEE, 2023.

[5] 類神經網路架構<https://medium.com/marketingdatascience/快速反應機制-類神經網路-a3bbdee4a6f6> Retrieved on December,2023.

[6] 全連接層 <https://www.dohkoai.com/usr/show?id=33> Retrieved on December,2023

[7] Li, ZhaoHui, GuangYun Deng, and HuiChung Che. "Patent-based predictive price-to-earnings ratio on increasing investment performance of china stock market." 2020 13th international symposium on computational intelligence and design (ISCID). IEEE, 2020.

[8] Dechow, P. M. "Accounting Earnings and Cash Flows as Measures of Firm Performance: the Role of Accounting Accruals." The Accounting Review (1994).

[9] Chen, Long, David A. Lesmond, and Jason Wei. "Corporate yield spreads and bond liquidity." The journal of finance 62.1 (2007): 119-149.

[10]Zhang, Encai, et al. "Correlation analysis between stock prices and four financial indexes for some listed companies of mainland China." 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI). IEEE, 2017.

[11]Shi, Yinhong. "Analysis of multi-factor stock market choice portfolio model based on regression." 2021 5th Annual International Conference on Data Science and Business Analytics (ICDSBA). IEEE, 2021.

[12]Fauzan, Achmad, et al. "Predicting Stock Market Movements Using Long Short-Term Memory (LSTM)." 2023 4th International Conference on Artificial Intelligence and Data Sciences (AiDAS). IEEE, 2023.

[13] Refenes, Apostolos Nicholas, Achileas Zapranis, and Gavin Francis. "Stock performance modeling using neural networks: a comparative study with regression models." Neural networks 7.2 (1994): 375-388.

[14] Bareket, A., and B. Pârv. "Predicting Medium-term Stock Index Direction Using Constituent Stocks and Machine Learning." IEEE Access (2024).

[15] Kryzanowski, Lawrence, Michael Galler, and David W. Wright. "Using artificial neural networks to pick stocks." Financial Analysts Journal 49.4 (1993): 21-27.

[16] Abe, Masaya, and Hideki Nakayama. "Deep learning for forecasting stock returns in the cross-section." Advances in Knowledge Discovery and Data Mining: 22nd Pacific-Asia Conference, PAKDD 2018, Melbourne, VIC, Australia, June 3-6, 2018, Proceedings, Part I 22. Springer International Publishing, 2018.

[17] Das, Sourav, and Bharti Nagpal. "Stock Price Prediction Using Time Embedding and Attention Mechanism in Transformers." 2023 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI). IEEE, 2023.

[18] Lin, H., et al. "Stock price prediction using generative adversarial networks." Journal of Computer Science 17.3 (2021): 188-196.

[19] Zhang, Jilin, Lishi Ye, and Yongzeng Lai. "Stock price prediction using CNN-BiLSTM-Attention model." Mathematics 11.9 (2023): 1985.

[20] Wang, Shuzhen. "A stock price prediction method based on BiLSTM and improved transformer." IEEE Access (2023).

[21] Aadhitya, A., et al. Predicting Stock Market time-series data using CNN-LSTM Neural Network model. No. 2305.14378. 2023.

[22] <https://medium.com/@adea820616/activation-functions-sigmoid-relu-tahn-20e3ae726ae>

[23 ]Zhang, Encai, et al. "Correlation analysis between stock prices and four financial indexes for some listed companies of mainland China." 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI). IEEE, 2017.

[24] Deng, Shangkun, et al. "Stock Price Crash Warning in the Chinese Security Market Using a Machine Learning-Based Method and Financial Indicators." Systems 10.4 (2022): 108.

[25] Fitriyana, Rahma Firsty, et al. "Principal component analysis to determine main factors stock price of consumer goods industry." 2020 International Conference on Data Science and Its Applications (ICoDSA). IEEE, 2020.

[26] Narayana, S., et al. "Predicting the stock market index using GRU for the year 2020." 2024 International Conference on Emerging Systems and Intelligent Computing (ESIC). IEEE, 2024.

[27] Mirjebreili, Seyed Morteza, et al. "Multi-Task Transformer for Stock Market Trend Prediction." 2022 12th International Conference on Computer and Knowledge Engineering (ICCKE). IEEE, 2022.

[28] Alhazbi, Saleh, Ahmed Ben Said, and Alanoud Al-Maadid. "Using deep learning to predict stock movements direction in emerging markets: the case of Qatar stock exchange." 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT). IEEE, 2020.

[29]Li, Zhixi, and Vincent Tam. "A comparative study of a recurrent neural network and support vector machine for predicting price movements of stocks of different volatilites." 2017 IEEE Symposium Series on Computational Intelligence (SSCI). IEEE, 2017.

[30] Naresh, E., et al. "Predicting the stock price using natural language processing and random forest regressor." 2022 IEEE International Conference on Data Science and Information System (ICDSIS). IEEE, 2022.

[31]https://terrifyzhao.github.io/2018/02/16/%E5%8A%A8%E9%87%8F%E6%A2%AF%E5%BA%A6%E4%B8%8B%E9%99%8D%E6%B3%95Momentum.html

[32]<https://kilong31442.medium.com/%E6%B7%B1%E5%BA%A6%E5%AD%B8%E7%BF%92%E5%90%84%E7%A8%AE%E6%96%B0%E5%84%AA%E5%8C%96%E5%99%A8%E4%BB%8B%E7%B4%B9-lookahead-ranger-lars-830ca2250dd5>

[33] <https://finance.yahoo.com/quote/IBM/>

[34] https://mops.twse.com.tw/mops/web/index

[35] Dael, Fares Abdulhafidh, Ömer Çagri Yavuz, and Ugur Yavuz. "Stock Market Prediction Using Generative Adversarial Networks (GANs): Hybrid Intelligent Model." Comput. Syst. Sci. Eng. 47.1 (2023): 19-35.

[36] <https://mkh800.medium.com/%E7%AD%86%E8%A8%98-attention-%E5%8F%8A-transformer-%E6%9E%B6%E6%A7%8B%E7%90%86%E8%A7%A3-c9c5479fdc8a>

[37]<https://medium.com/%E6%88%91%E5%B0%B1%E5%95%8F%E4%B8%80%E5%8F%A5-%E6%80%8E%E9%BA%BC%E5%AF%AB/dilation-convolution-d322febe0621>

[38] Ravikumar, Srinath, and Prasad Saraf. "Prediction of stock prices using machine learning (regression, classification) Algorithms." 2020 International Conference for Emerging Technology (INCET). IEEE, 2020.

[39] Long-Term Stock returns using Long Short Term Memory Neural Network." 2023 Second International Conference on Advances in Computational Intelligence and Communication (ICACIC). IEEE, 2023.

[40] Alamsyah, Andry, and Asri Nurfathi Zahir. "Artificial Neural Network for predicting Indonesia stock exchange composite using macroeconomic variables." 2018 6th International Conference on Information and Communication Technology (ICoICT). IEEE, 2018.

[41]<https://medium.com/@hupinwei/%E6%B7%B1%E5%BA%A6%E5%AD%B8%E7%BF%92%E5%8D%B7%E7%A9%8D%E7%A5%9E%E7%B6%93%E7%B6%B2%E8%B7%AF-convolution-neural-network-cnn-15b413c4a1c8>

[42] Yousuf, Md Sakif Ibn, SM Rashidul Hasan Nijhum, and Muhammad Ibrahim Khan. "HybStock: A Hybrid Deep Learning-Based Model to Predict Stock Market on Dhaka Stock Exchange." 2024 6th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT). IEEE, 2024

[43] <https://brohrer.mcknote.com/zh-Hant/how_machine_learning_works/how_rnns_lstm_work.html>

[44]<https://tengyuanchang.medium.com/%E6%AF%94%E8%BC%83%E9%95%B7%E7%9F%AD%E6%9C%9F%E8%A8%98%E6%86%B6%E6%A8%A1%E5%9E%8B-lstm-%E8%88%87%E6%94%B9%E8%89%AF%E5%BE%8C%E7%9A%84%E9%81%9E%E6%AD%B8%E7%A5%9E%E7%B6%93%E7%B6%B2%E8%B7%AF%E6%A8%A1%E5%9E%8B-gru-813dec22ec6d>

[45] Chang, Chin-Chih, Kei-Wen Luo, and Sean Hsiao. "Stock Price Prediction and Recommendation Approach Based on Machine Learning." 2022 IEEE 4th Eurasia Conference on IOT, Communication and Engineering (ECICE). IEEE, 2022.