

ChatGPT詠唱術



透過ChatGPT增進程式開發效率與金融市場預測模型實作



使用正確的指令像個老師引導chatgpt協助寫程式

金融市場預測的資料科學流程



1. 蒐集相關資料



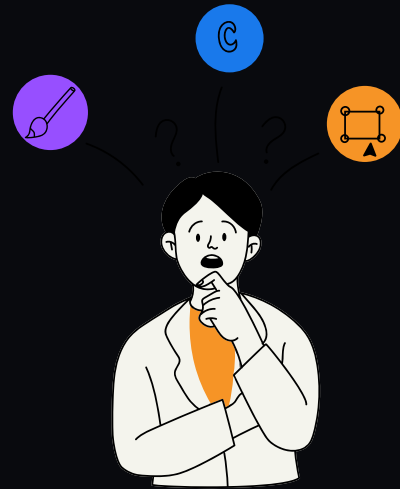
2. 檢查資料



3. 數據預處理



4. 特徵工程



5. 模型選擇



6. 模型訓練



7. 模型評估



8. 模型回測優化

課程大綱

1. 介紹chatgpt以及常見應用
2. 程式環境架設 colab 開發環境介紹
3. 程式基本教學 yfinance 股價資料取得教學
4. 程式基本教學 pandas 操縱股價數據
5. 程式基本教學 設計程式流程圖(資料驅動模型開發流程為範例)
6. 程式基本教學 finta快速製作技術指標
7. chatgpt詠唱術 AI協作股價資料取得程式
8. chatgpt詠唱術 AI協作邏輯複雜的數據操作
9. chatgpt詠唱術 AI協作神經網路模型架構
10. chatgpt詠唱術 AI協作模型評估函數
11. chatgpt詠唱術 AI協作回測框架
12. chatgpt詠唱術 AI協作自動推播交易訊號的telegram bot

chatgpt 常見應用

- 1 程式碼生成/補全/改良
- 2 學術問答
- 3 投資金融問答
- 4 旅遊問答
- 5 文學創作

chatgpt模型是如何誕生的

A prompt is sampled from our prompt dataset.

🔄
Explain reinforcement learning to a 6 year old.

A labeler demonstrates the desired output behavior.

👤
We give treats and punishments to teach...

This data is used to fine-tune GPT-3.5 with supervised learning.

SFT
🧠
📄📄📄

A prompt and several model outputs are sampled.

🔄
Explain reinforcement learning to a 6 year old.

A In reinforcement learning, the agent is...

C In machine learning...

B Explain rewards...

D We give treats and punishments to teach...

A labeler ranks the outputs from best to worst.

👤
D > C > A > B

This data is used to train our reward model.

RM
🧠
D > C > A > B

A new prompt is sampled from the dataset.

🦖
Write a story about otters.

The PPO model is initialized from the supervised policy.

PPO
🧠

The policy generates an output.

Once upon a time...

The reward model calculates a reward for the output.

RM
🧠

The reward is used to update the policy using PPO.

r_k

程式碼生成

希望

希望它幫忙做一個螢幕錄製軟體

提示詞:give me code to make a screen recorder using html css javascript
getusermedia outputfile path in current dir and filename is test.mp4



Unfortunately, it's not possible to create a screen recorder solely using HTML, CSS, and JavaScript. Screen recording requires access to the user's operating system and hardware, which cannot be achieved using these web technologies alone.



However, you can use a third-party library like RecordRTC, which uses WebRTC technology to capture screen and audio activity and save it to a file. Here's an example code snippet that uses RecordRTC to record the screen and save the file as "test.mp4" in the current directory:

html

Copy code

```
<!DOCTYPE html>
<html>
  <head>
    <title>Screen Recorder</title>
  </head>
  <body>
    <video id="video" width="640" height="480" autoplay muted></video>
    <button id="start">Start Recording</button>
    <button id="stop">Stop Recording</button>
```

程式碼生成

由於第一個版本有些問題這裡請他疑難排解

提示詞:I not see test.mp4



I apologize for the confusion. The code I provided in my previous answer is designed to upload the recorded video to a server using an AJAX request, and it does not save the file to the local file system.



If you want to save the recorded video to the local file system, you can use the `download` attribute of the `a` tag in the HTML code. Here is the updated HTML code for the screen recorder:

html

Copy code

```
<!DOCTYPE html>
<html>
  <head>
    <title>Screen Recorder</title>
  </head>
  <body>
```

```
< index.html X
index.html > html > body >
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Screen Recorder
5 </head>
6 <body>
7 <video id="video" width=
8 <button id="start">St
9 <button id="stop">Sto
10 <a id="downloadLink" >
11
12 <script src="https://
13 <script>
14 var recorder, stream
15
16 var video = document
17 var start = document
18 var stop = document
19 var downloadLink =
20
21 start.addEventListener
22 navigator.mediaDe Download
23 .getDisplayMedia
24 .then(function(s
25 stream = s;
26 video.srcObje
27 recorder = Re
```

Screen Recorder

檔案

D:/Users/Administrator/Desktop/chatgpt_screen_recorder/index.html

☆

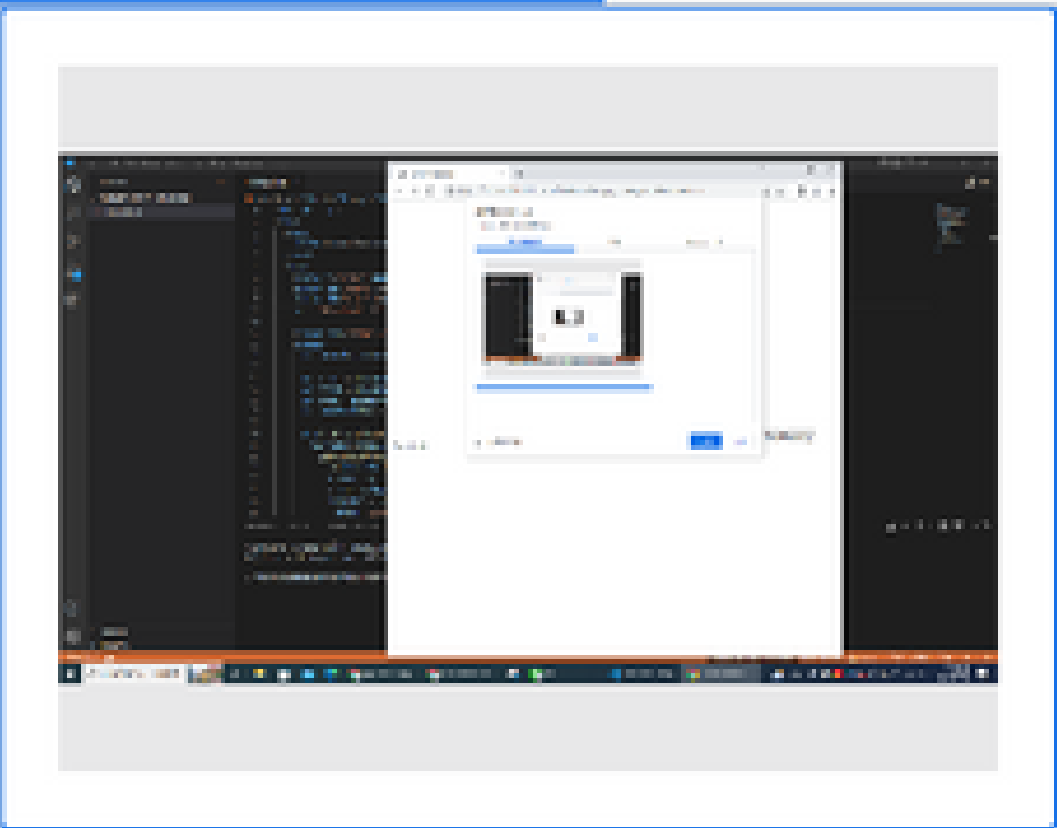
選擇要分享的内容

'file://' 要求分享螢幕内容。

整個螢幕畫面

視窗

Chrome 分頁



☐ 分享系統資訊

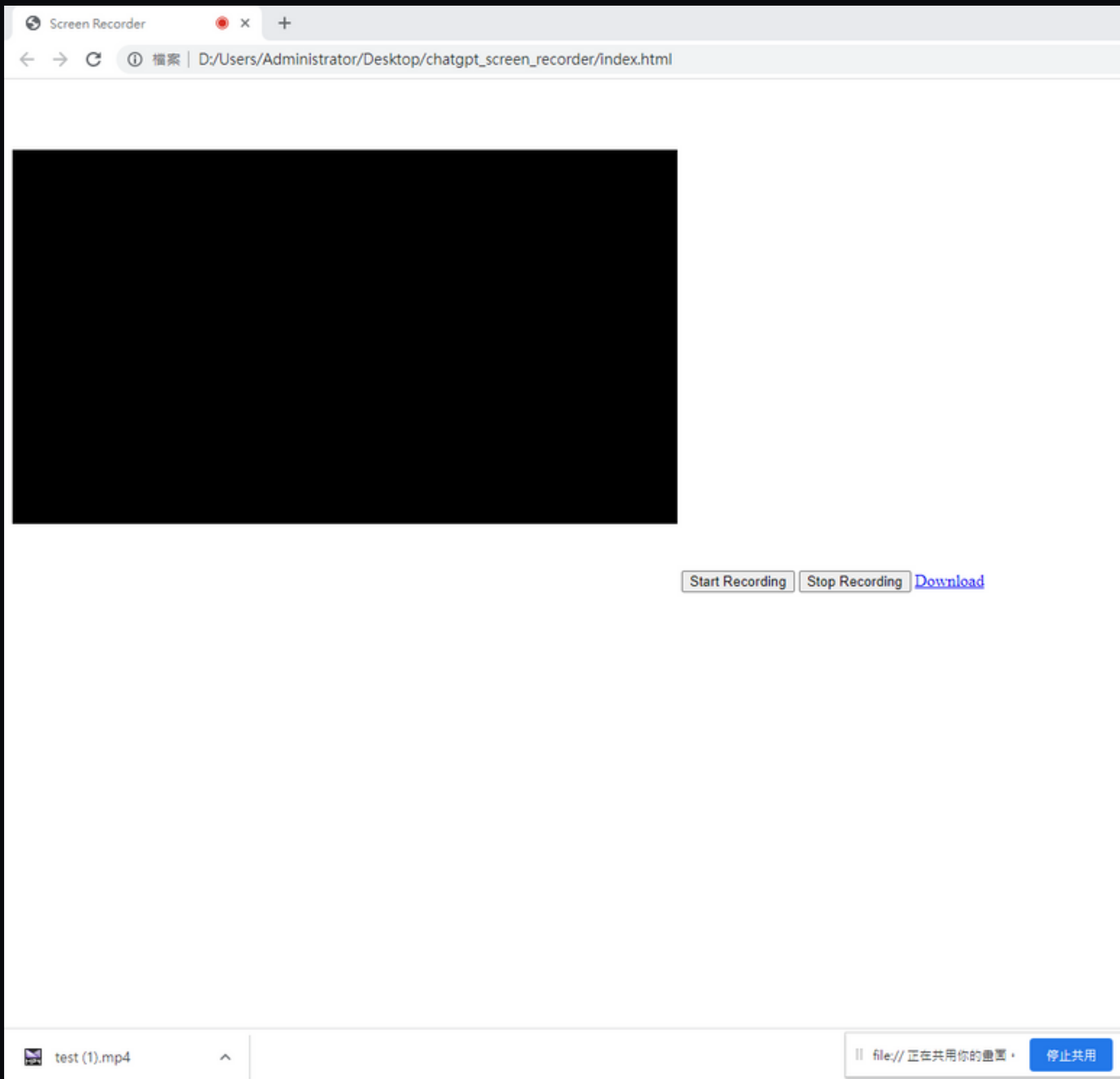
分享

取消

Stop Recording

程式碼開發成功

完整程式碼連結



程式碼改良

提示詞:

<這裡寫你的程式碼>

改良以上代碼讓他更有效率更簡潔



以下是可能的優化和簡化方法：



1. 整合技術指標函數

您可以將RSI、Williams %R、SMA、EMA等指標的計算合併到一個函數中。這樣可以減少代碼量並提高代碼可讀性。

2. 優化數據處理流程

您可以將數據處理流程放在單獨的函數中，以減少代碼的複雜性。這個函數可以包括數據清理、特徵工程和數據轉換等步驟。

3. 優化Xs和indexs的生成

您可以使用pandas.DataFrame.shift()函數來創建Xs和indexs。這樣可以減少循環次數和代碼量。

4. 優化模型預測流程

您可以將模型預測的部分優化為向量運算。這樣可以大大減少循環次數，提高代碼效率。

5. 優化繪圖流程

您可以使用matplotlib的向量化繪圖功能來優化繪圖流程。這樣可以減少繪圖時間和代碼複雜性。

下面是一個可能的優化和簡化代碼的示例。請注意，這只是一個示例，您需要根據您的實際情況進行修改和優化。

程式碼改良

改良前

```
def process(df,min_max_scaler,model):
    df['RSI'] = TA.RSI(df)#1
    df['Williams %R'] = TA.WILLIAMS(df)#2
    df['SMA'] = TA.SMA(df)#3
    df['EMA'] = TA.EMA(df)#4
    df['WMA'] = TA.WMA(df)#5
    df['HMA'] = TA.HMA(df)#6
    df['TEMA'] = TA.TEMA(df)#7
    df['CCI'] = TA.CCI(df)#8
    df['CMO'] = TA.CMO(df)#9
    df['MACD'] = TA.MACD(df)['MACD'] - TA.MACD(df)['SIGNAL']#10
    df['PPO'] = TA.PPO(df)['PPO'] - TA.PPO(df)['SIGNAL']#11
    df['ROC'] = TA.ROC(df)#12
    df['CFI'] = TA.CFI(df)#13
    df['DMI'] = TA.DMI(df)['DI+'] - TA.DMI(df)['DI-']#14
    df['SAR'] = TA.SAR(df)#15
```

chatgpt改良後

python

```
# 整合技術指標計算函數
def calculate_ta(df):
    ta_functions = [TA.RSI, TA.WILLIAMS, TA.SMA, TA.EMA, TA.WMA, TA.HMA, TA.TEMA,
                    TA.CCI, TA.CMO, TA.MACD, TA.PPO, TA.ROC, TA.CFI, TA.DMI, TA.SAR]
    ta_names = ['RSI', 'Williams %R', 'SMA', 'EMA', 'WMA', 'HMA', 'TEMA', 'CCI',
                'CMO', 'MACD', 'PPO', 'ROC', 'CFI', 'DMI+', 'DMI-', 'SAR']
    for i, ta_func in enumerate(ta_functions):
        df[ta_names[i]] = ta_func(df)

    return df
```

程式碼補全

提示詞範本:

我想請你幫忙做程式碼補全

#字元後面的描述代表我想做的事情

<you codeing here>代表請你輸入代碼

我會貼給你半完成的程式碼

請你幫我補全程式碼

我的第一個程式碼如下:

```
def func_name(arg1,arg2,arg3):
```

```
'''
```

<這裡描述函數內部邏輯>

```
'''
```

```
# 描述你要做什麼
```

```
<you codeing here>
```

```
# 描述你要做什麼
```

```
<you codeing here>
```

```
# 描述你要做什麼
```

```
<you codeing here>
```

一個例子:

<https://sharegpt.com/c/Lbb07>

[KU](#)

學術問答

希望

提示詞:解釋神經網路工作原理



神经网络是一种计算机程序或模型，它受到人脑中神经元的结构和功能的启发。神经网络通过大量的数据来学习输入和输出之间的关系，以便在没有显式编程的情况下进行任务。以下是神经网络的一般工作原理：

1. 神经元：神经网络的基本单元是神经元，它接收输入并生成输出。神经元的输出由一个激活函数来计算。常见的激活函数包括sigmoid、ReLU等。
2. 层：神经元通常按层组织在一起，层之间的神经元相互连接。神经网络通常包括输入层、隐藏层和输出层。
3. 前向传播：当一个输入被提供给神经网络时，输入信号被送到输入层，然后通过隐藏层传递，最后产生输出。这个过程称为前向传播。
4. 反向传播：在神经网络学习过程中，通过与训练数据比较来评估网络的性能。反向传播是用于计算损失函数梯度的过程，然后通过梯度下降法来更新权重，以改进网络的性能。
5. 权重：权重是连接神经元的强度。权重的优化是训练神经网络的关键。权重的调整可以使网络更好地预测输出。
6. 学习率：学习率是权重更新的步长。过大的学习率可能会导致权重震荡，而过小的学习率可能会导致学习缓慢。通常需要进行调整以实现最佳结果。

总之，神经网络是一种可以通过数据学习输入和输出之间关系的计算机程序。它通过层、神经元和权重相互连接来工作。前向传播用于产生输出，反向传播用于计算损失函数梯度并更新权重以提高性能。

投資金融問答

提示詞:你可以預測下週美股大盤走勢嗎?



很抱歉，我作為一個AI語言模型，不能預測未來的股市走勢。股市受到許多因素的影響，包括全球經濟、政治和自然災害等等，因此其走勢難以預測。投資者應該謹慎評估自己的風險承受能力和投資目標，進行充分的研究，並在進行任何投資前諮詢專業人士的建議。



希望

還沒結束其實這告訴我們應該換個方式問



如果從資料科學的流程,來對金融市場價格走勢建模,大概要分哪些步驟?

這樣就可以把原始問題轉化成一個工程技術問題,通常chatgpt不會拒絕回答工程技術問題

<https://sharegpt.com/c/eZgwz1e>

投資金融問答



如果從資料科學的流程,來對金融市場價格走勢建模,大概要分哪些步驟?



對金融市場價格走勢建模的資料科學流程可以概括為以下步驟：



1. 定義問題：確定要預測的金融市場價格走勢以及預測的目的，例如預測股價是否上漲或下跌。
2. 數據收集：收集與問題相關的數據，包括金融市場價格、財務報表和經濟指標等。
3. 數據預處理：對數據進行清洗、轉換和縮放等操作，以便於後續建模使用。
4. 特徵工程：從原始數據中提取有意義的特徵，以便於後續建模使用。這可能包括技術指標、財務指標和基本面指標等。
5. 模型選擇：選擇適合問題的模型，例如支持向量機、決策樹或深度學習模型等。
6. 模型訓練：使用歷史數據訓練模型，以便模型能夠學習與問題相關的模式。
7. 模型評估：使用測試數據評估模型的性能，例如準確率、精確率和召回率等。
8. 模型優化：優化模型的超參數和結構，以提高模型的性能。
9. 部署模型：將模型部署到實際環境中，以便於使用者可以使用它進行預測。

需要注意的是，金融市場的波動性和不確定性很高，因此在建模時需要謹慎考慮這些因素。此外，金融市場數據可能存在非常複雜的關係，因此需要使用適當的建模方法和技術。



旅遊問答

提示詞

從新竹出發到花蓮
三天兩夜遊,推薦3
種旅遊行程



Day 1:

1. 早上8點從新竹出發，前往台東縣成功鎮(約3.5小時車程)
2. 下午前往蕃薯寮文化村(約2小時車程)，體驗客家文化及客家美食
3. 晚上前往台東市，入住當地民宿或飯店

Day 2:

1. 早上前往太魯閣國家公園(約1.5小時車程)，欣賞太魯閣峽谷的壯麗景色
2. 下午前往七星潭(約1.5小時車程)，欣賞美麗的海灘風景
3. 晚上前往花蓮市區，入住當地民宿或飯店

Day 3:

1. 早上前往花蓮市的光復糖廠(約30分鐘車程)，了解台灣糖業歷史
2. 下午前往七星潭附近的太平洋公路，欣賞海岸風光
3. 晚上返回新竹(約4.5小時車程)，結束旅程

以上是一個基本的旅遊行程，可依照個人喜好和時間調整，例如可以加入梅花湖、清水斷崖等景點，或是增加活動如泛舟、鐵道自行車等，讓旅程更豐富有趣。

文學創作

提示詞

用《銀翼殺手
2049》劇情寫一首
詩：



黑暗之城
污染之中
未來電子之聲
與人類的淚
同聲共鳴

「銀翼殺手」
重生的謎團
在殘酷的現實中
追尋生命的意義
每個存在都被質疑

「2049」
時光漸遠
回憶如煙
未來與過去
共同的故事

寂靜的夜晚
鋼鐵般的心
走過命運的迷宮
在虛幻與真實間
尋找靈魂的出路

程式環境架設,colab介紹

colab連結:<https://colab.research.google.com/>

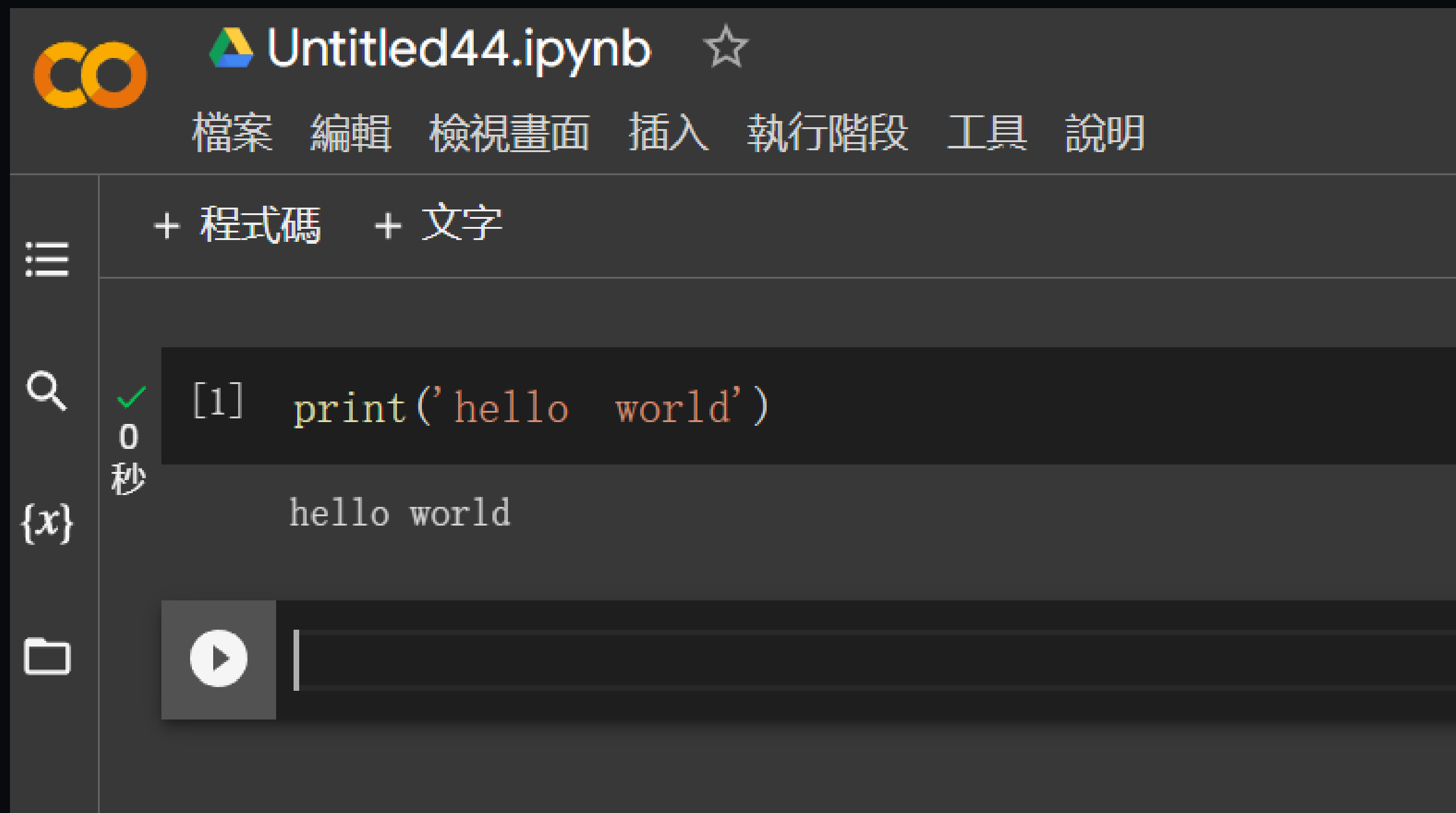
新建空白筆記本:
檔案>新增筆記本



程式環境架設,colab介紹

撰寫程式碼:

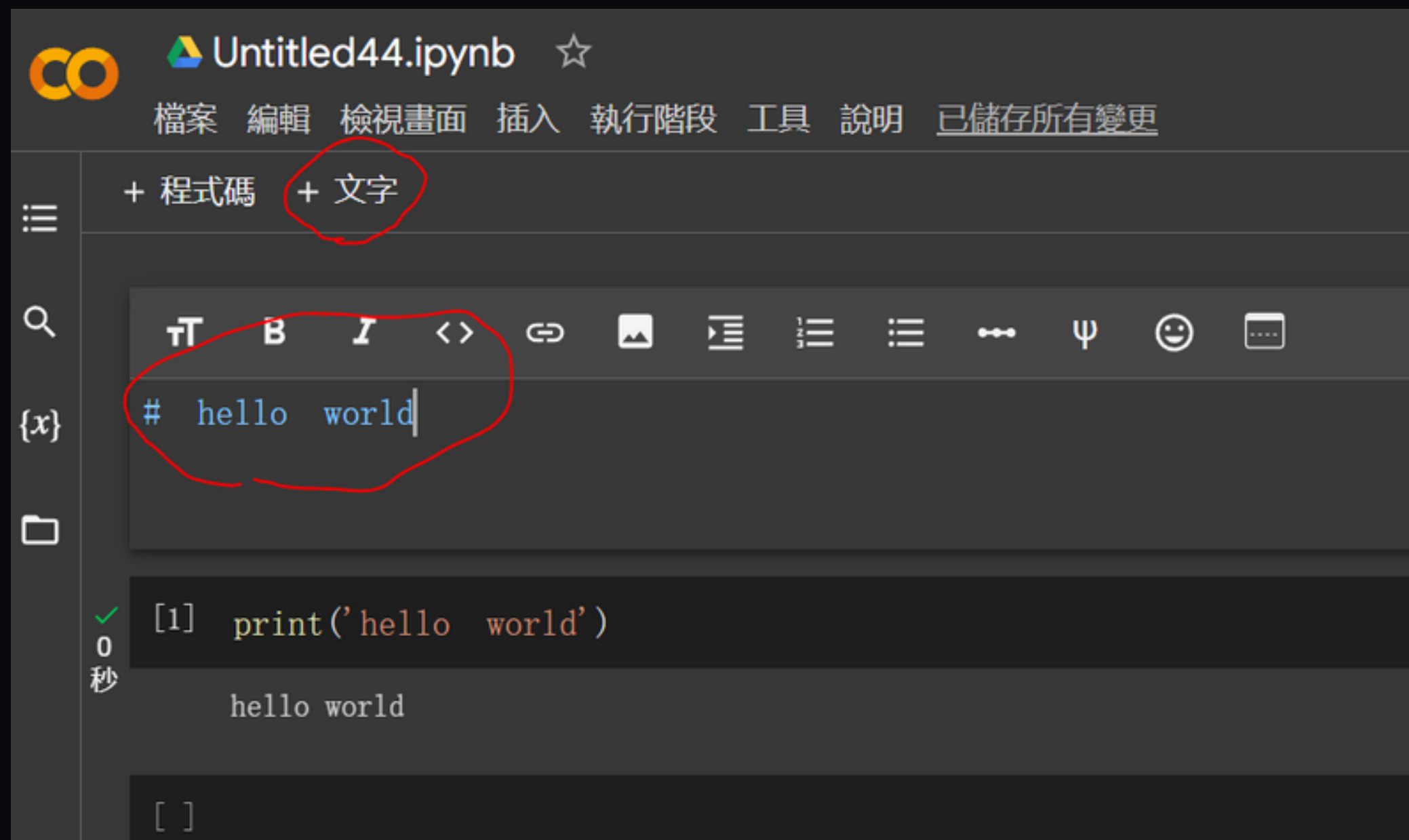
第一行程式碼：`print('h e l l o w o r l d')`



程式環境架設,colab介紹

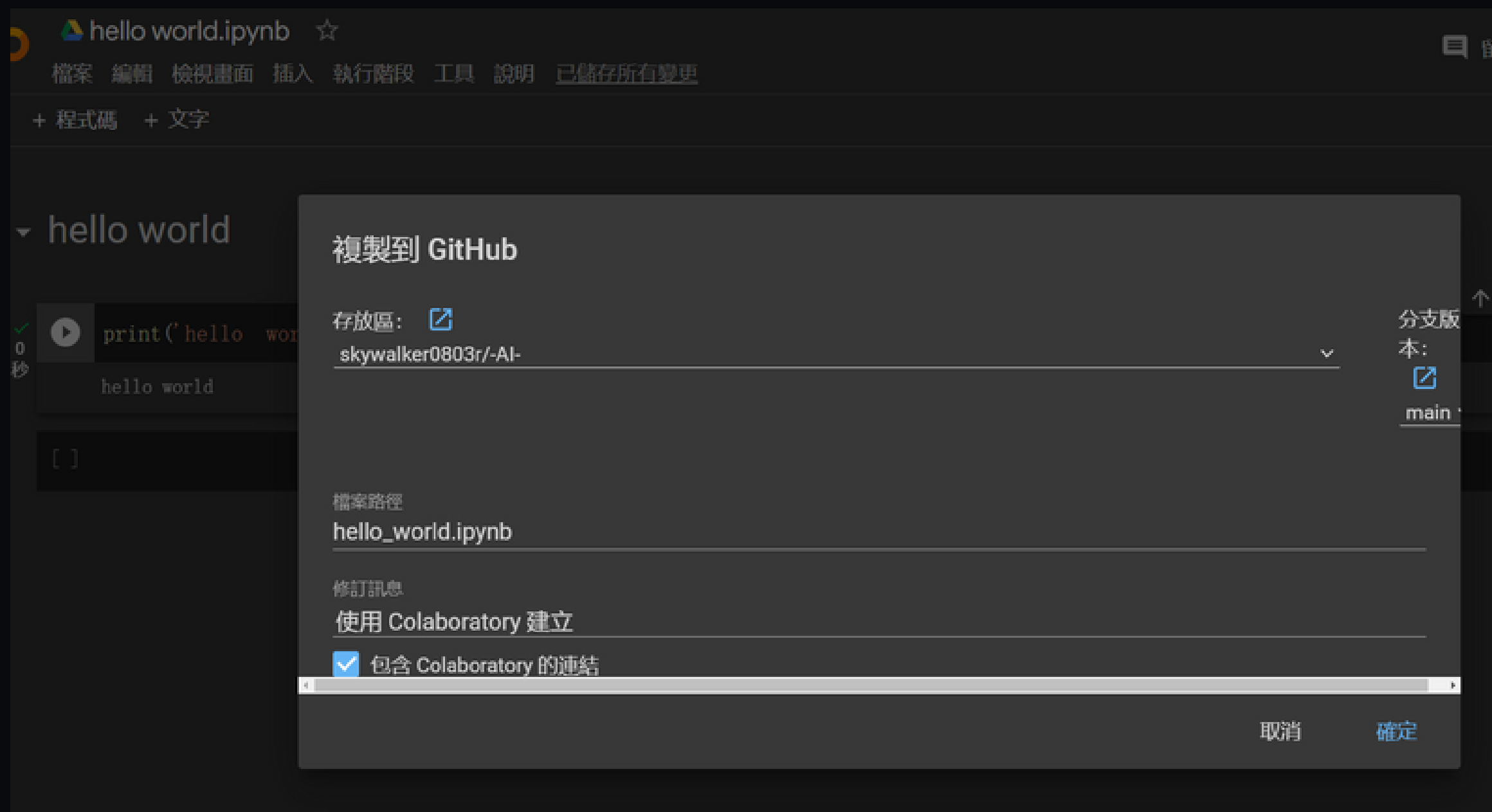
撰寫文字描述

點選+文字 # <輸入文字>



程式環境架設,colab介紹

保存檔案至github:
檔案>在github儲存副本



程式環境架設,colab介紹

執行所有單元格:
執行階段>全部執行



來實作一篇利用CNN做金融交易的論文

Algorithmic Financial Trading with Deep Convolutional Neural Networks: Time Series to Image Conversion Approach

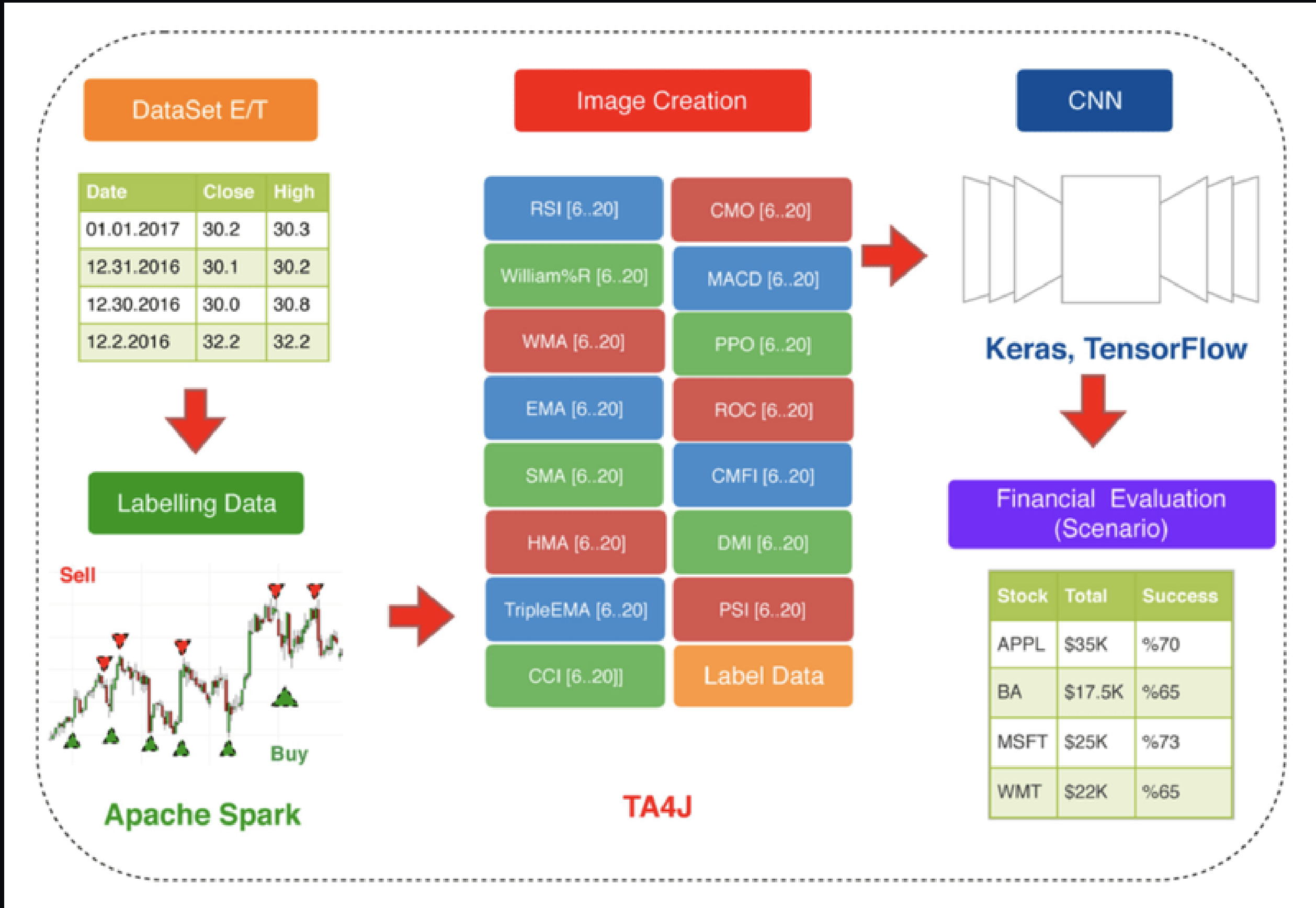
Omer Berat Sezer^{a,*}, Ahmet Murat Ozbayoglu^a

^a*TOBB University of Economics and Technology, Ankara, 06560, Turkey*

Abstract

Computational intelligence techniques for financial trading systems have always been quite popular. In the last decade, deep learning models start getting more attention, especially within the image processing community. In this study, we propose a novel algorithmic trading model CNN-TA using a 2-D Convolutional Neural Network based on image processing properties. In order to convert financial time series into 2-D images, 15 different technical indicators each with different parameter selections are utilized. Each indicator instance generates data for a 15 day period. As a result, 15x15 sized 2-D images are constructed. Each image is then labelled as Buy, Sell or Hold depending on the hills and valleys of the original time series. The results indicate that when compared with the Buy & Hold Strategy and other common trading systems over a long out-of-sample period, the trained model provides better results for stocks and ETFs.

來實作一篇利用CNN做金融交易的論文



來實作一篇利用CNN做金融交易的論文

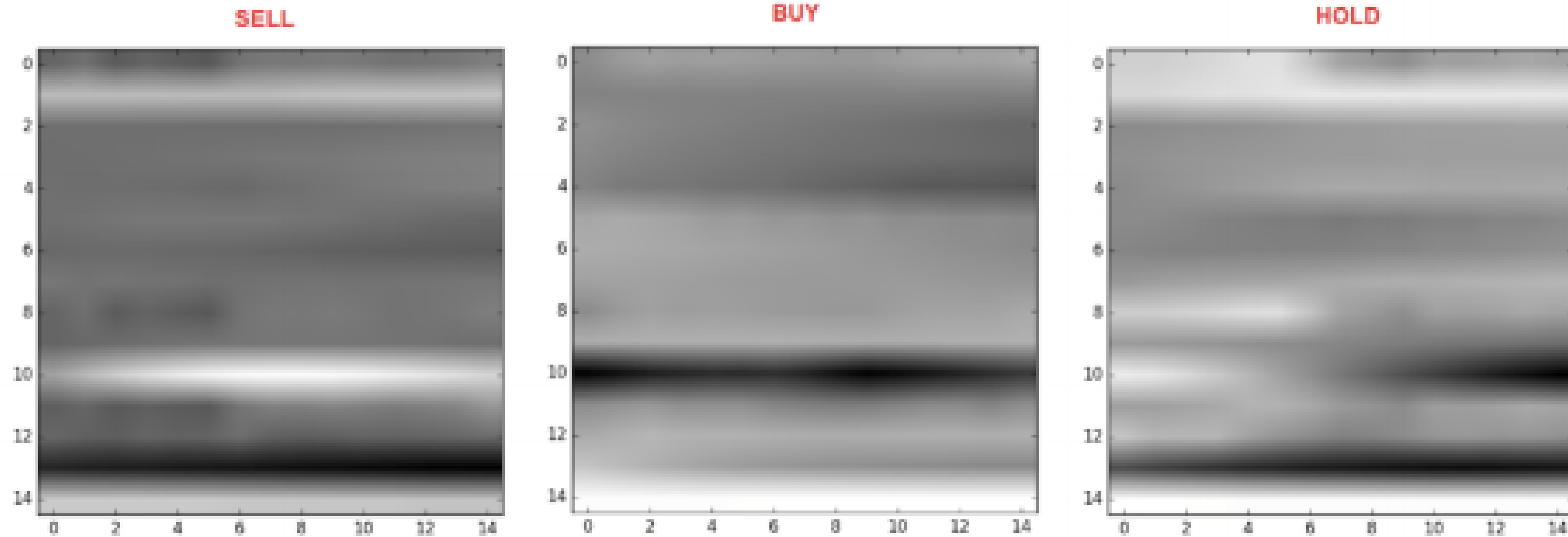


Figure 4: 15x15 Pixel Labelled Sample Images After Image Creation Phase

來實作一篇利用CNN做金融交易的論文

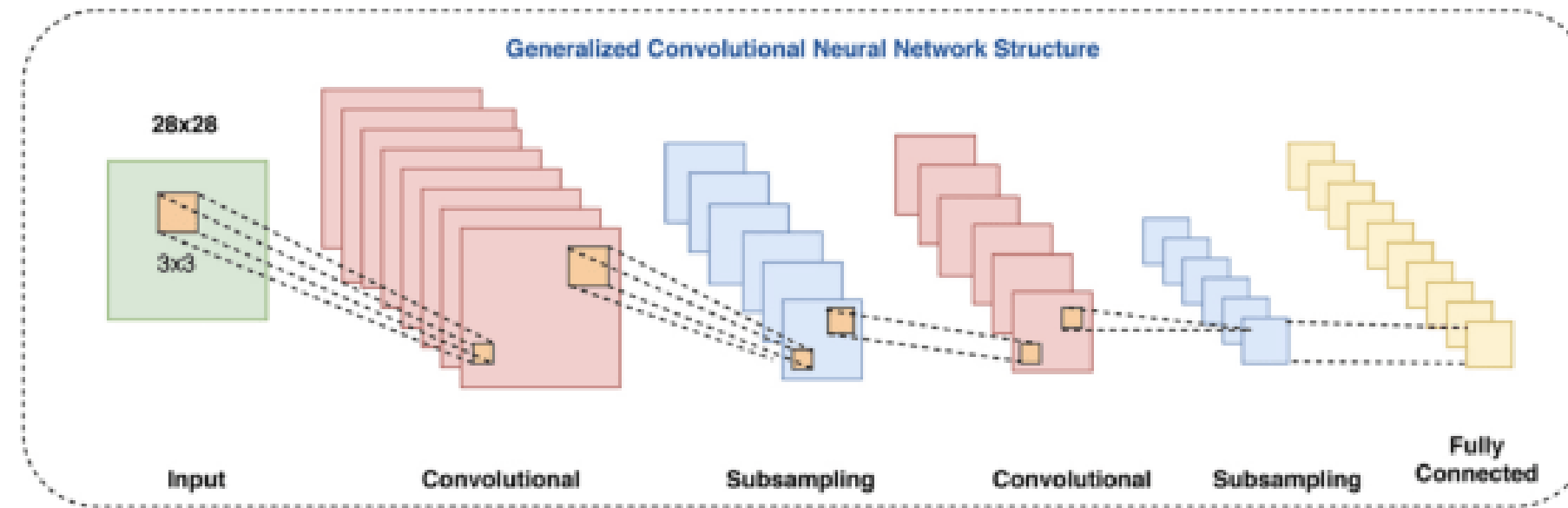


Figure 1: Generalized Convolutional Neural Network

- input layer (15×15),
- two convolutional layers ($15 \times 15 \times 32$, $15 \times 15 \times 64$),
- max pooling ($7 \times 7 \times 64$),
- two dropout (0.25, 0.50),
- fully connected layers (128),
- output layer (3)

來實作一篇利用CNN做金融交易的論文

XLE - Profits Over the Years (2007 - 2012)



XLF - Profits Over the Years (2007 - 2017)

