

商業智慧與營運策略

Business Intelligence and Operations Strategy

多模態與技術指標

Week8

授課老師：林冠成 教授

Kuan-Cheng Lin

5501.mis.nchu.edu.tw



課程大綱

- 多模態核心技術

- 多模態方法介紹(融合、協同學習)
- 預訓練與遷移學習 (CLIP、BLIP)
- 多模態可解釋性分析 (決策樹、SHAP、LIME、Grad-CAM)

- K線型態和四大技術指標分類

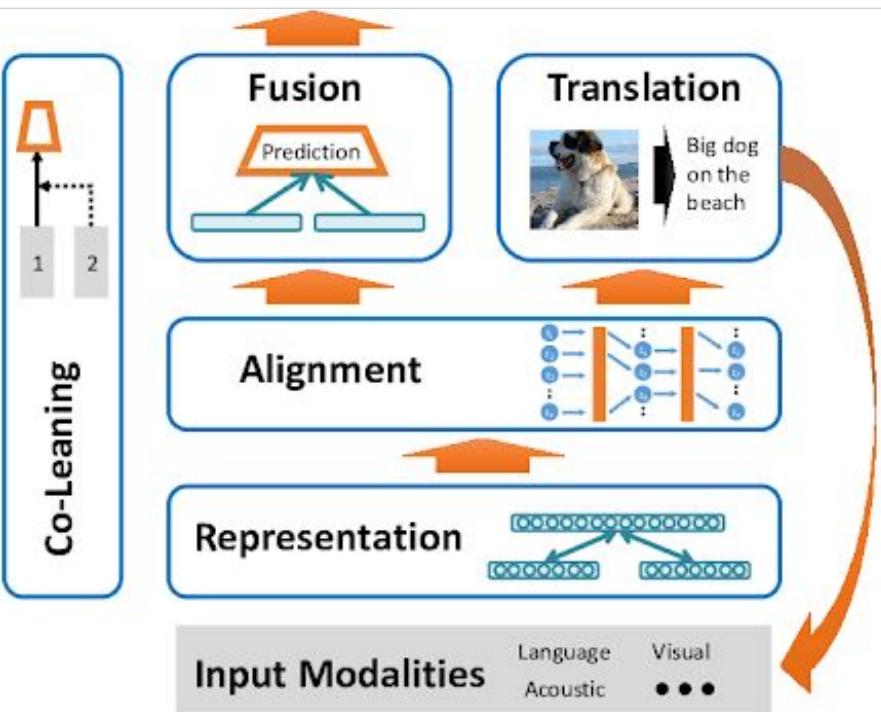
- 技術指標與時間序列模型的定位
- K線圖的基本型態
- 四大技術指標分類(趨勢、動能、波動、成交量)
- 四大技術指標分類與模型適配度關係
- XGBoost、ARIMA、LSTM和Transformer介紹



一、多模態核心技術

二、K線型態和四大技術指標分類

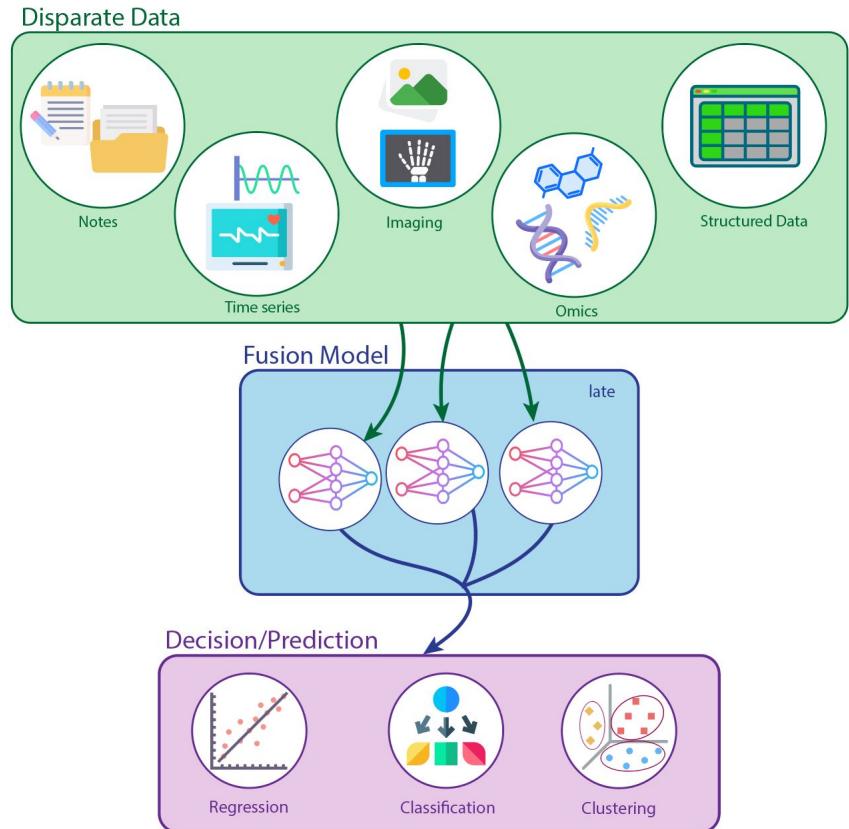
多模態五種方式



- **Representation(表示)**
 - 將不同模態轉換為向量表示
 - CNN 抽圖像特徵、BERT 抽文字特徵；可分 Joint / Coordinated 兩類
- **Alignment(對齊)**
 - 建立不同模態元素的對應關係
 - 文字與圖像區域對齊 (Cross-Attention、對比學習)
- **Fusion(融合)**
 - 整合不同模態資訊
 - Early / Intermediate / Late / Bilinear Fusion
- **Translation(翻譯)**
 - 在不同模態間轉換訊息
 - Image Captioning (圖→文) 、Text-to-Image 生成
- **Co-learning(協同學習)**
 - 模態間互相增強知識
 - 少模態學習時，用有標註的模態幫助另一模態訓練

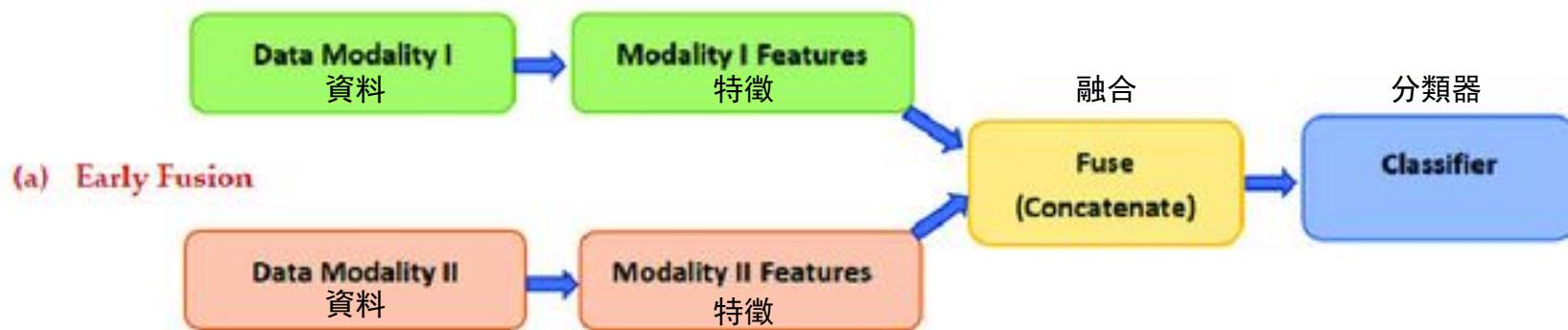
多模態階段融合 (Multimodal Fusion)

- 是指結合多種不同類型或來源的資料，如文字、圖像、音頻、數值等，進行綜合分析。在金融領域，這種方法有助於從多角度獲取資訊，提升模型的全面性和準確性。
- 多模態階段融合的方法：
 - 早期融合 (Early Fusion)
 - 中期融合 (Intermediate Fusion)
 - 晚期融合 (Late Fusion)
 - 混合融合 (Hybrid Fusion)



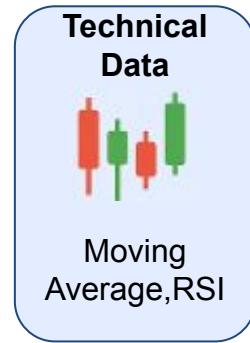
早期融合 (Early Fusion) (資料層融合)

- 定義：**在模型輸入階段，將不同模態的**原始資料**（如文本、圖像、數值等）**直接合併**成一個統一的表示，然後輸入模型。
- 優點：**
 - 融合方法簡單直接，適合資料來源格式一致的場景。
 - 能夠在初期捕捉模態之間的潛在關係。
- 缺點：**
 - 原始資料可能缺乏語義資訊，模型難以捕捉高階關係。
 - 維度高，計算資源需求大。
- 應用案例：**股票市場中，將**技術指標**（如移動平均線、RSI）和**財務資料**（如EPS、市盈率）直接拼接成一個特徵向量，輸入模型預測股價漲跌。



早期融合 (Early Fusion) (資料層融合)

輸入股價相關的
技術特徵



輸入公司的
財務特徵



Predict
Stock
Trend



(a)

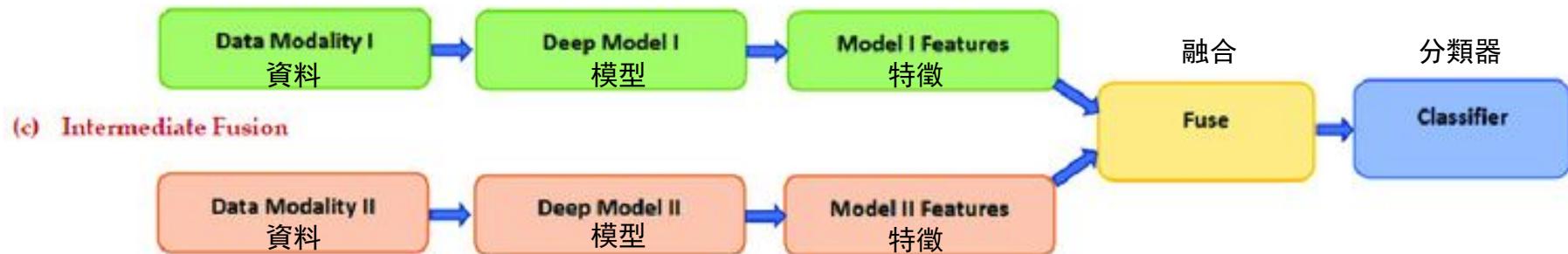
將技術與財務資料在特徵層面
合併 (Concatenate) ,
每一個時間步的特徵向量會包
含兩個模態的資訊

$$x_t = [MA_t, RSI_t, EPS_t, PE_t]$$

使模型能同時學習價格動能與基本面之間的關聯性

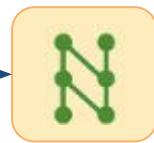
中期融合 (Intermediate Fusion) (特徵層融合)

- 定義：**各模態資料先經過專屬處理提取特徵，形成嵌入向量（Embeddings），然後將這些嵌入向量進行融合，再輸入後續的深度學習層處理。
- 優點：**
 - 能夠捕捉模態之間的深層交互資訊。
 - 利用嵌入向量統一表示不同資料模態，提高模型理解能力。
- 缺點：**
 - 增加了資料處理的步驟，對計算資源需求更高。
- 應用案例：**將股票的技術面資料（如歷史K線圖）通過CNN提取圖像特徵，新聞情緒（如相關財經報導）通過Transformer模型提取文本特徵，然後將兩種模態的特徵向量進行拼接，最終輸出預測股價走勢。



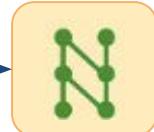
中期融合 (Intermediate Fusion) (特徵層融合)

輸入資料：
歷史K線圖



CNN
提取K線圖中的空間
特徵，捕捉市場結構
與價格形態

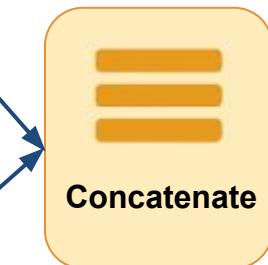
輸入資料：
每日財經新聞、
報導、公司公告



Transformer 提取文
本語意與情緒特徵，
反映市場預期與輿情
變化

(特徵層融合)

預測股價漲跌方向
或預測未來的價格變化趨勢

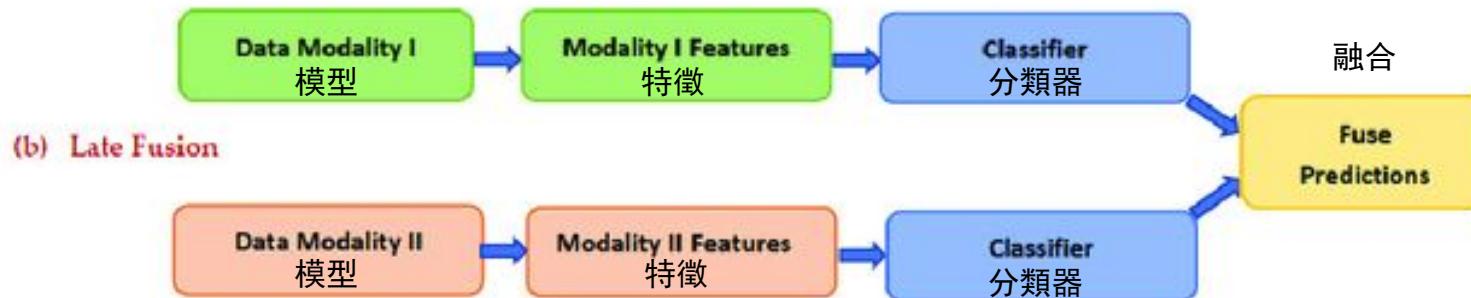


融合圖像與文本的潛在特徵，使模型能同
時學習價格形態與市場語意

拼接 (Concatenation)
加權求和 (Weighted Sum)
交互注意力 (Cross-Attention)

晚期融合 (Late Fusion) (決策層融合)

- **定義**: 各模態資料獨立輸入各自的模型進行預測，然後將**預測結果**（如分數或概率）**在決策層進行融合**。
- **優點**:
 1. 每個模態模型可以獨立設計與訓練，靈活性高。
 2. 避免模態間資料衝突，模型易於解釋。
- **缺點**:
 1. 無法充分挖掘模態間的潛在交互關係。
- **應用案例**: 使用獨立模型分別處理股票的技術面資料（如交易量、價格波動）和社交媒體資料（如投資者情緒），最後在決策層進行加權平均以生成綜合預測結果。



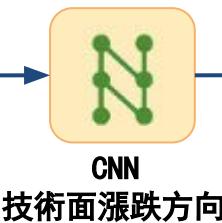
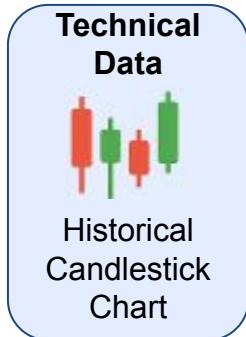
晚期融合 (Late Fusion)

(決策層融合)

中期融合：融合**特徵向量**（數字向量）

晚期融合：融合**預測結果**（具體的機率值）

輸入資料：
歷史K線圖



預測結果

預測股價漲跌方向

輸入資料：
每日財經新聞、
報導、公司公告



預測結果



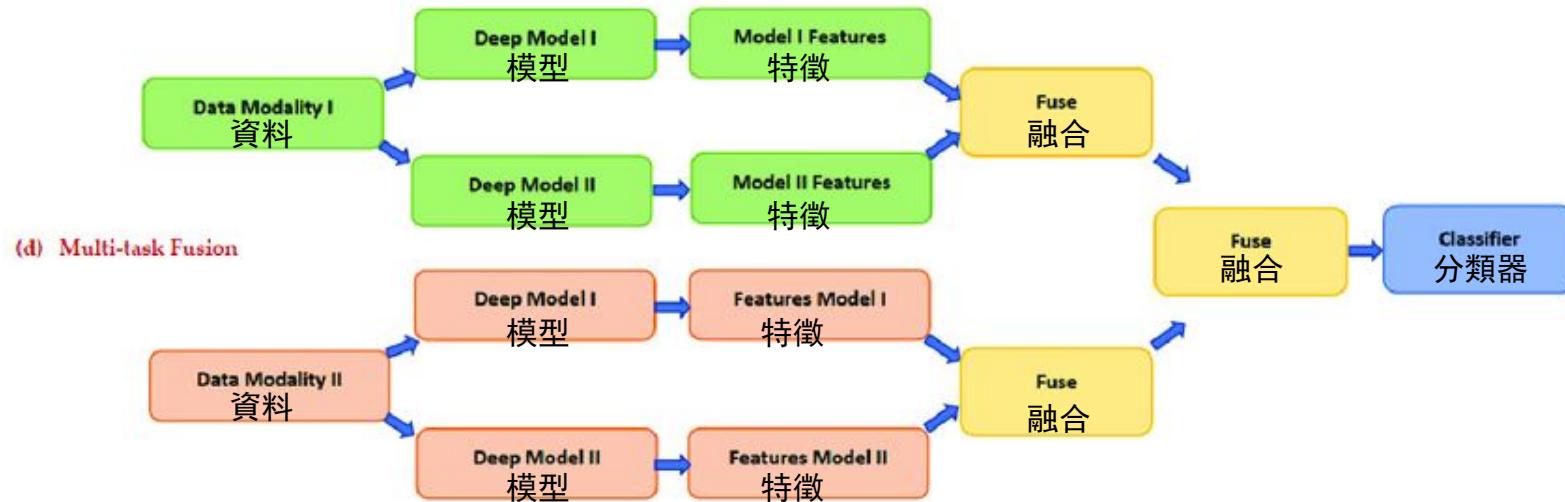
將兩個模型的**預測結果**（而非特徵）進行整合

加權平均 (Weighted Averaging)
投票法 (Voting)
Stacking Ensemble (堆疊融合)

各模型獨立學習不同模態特徵，產生各自預測結果

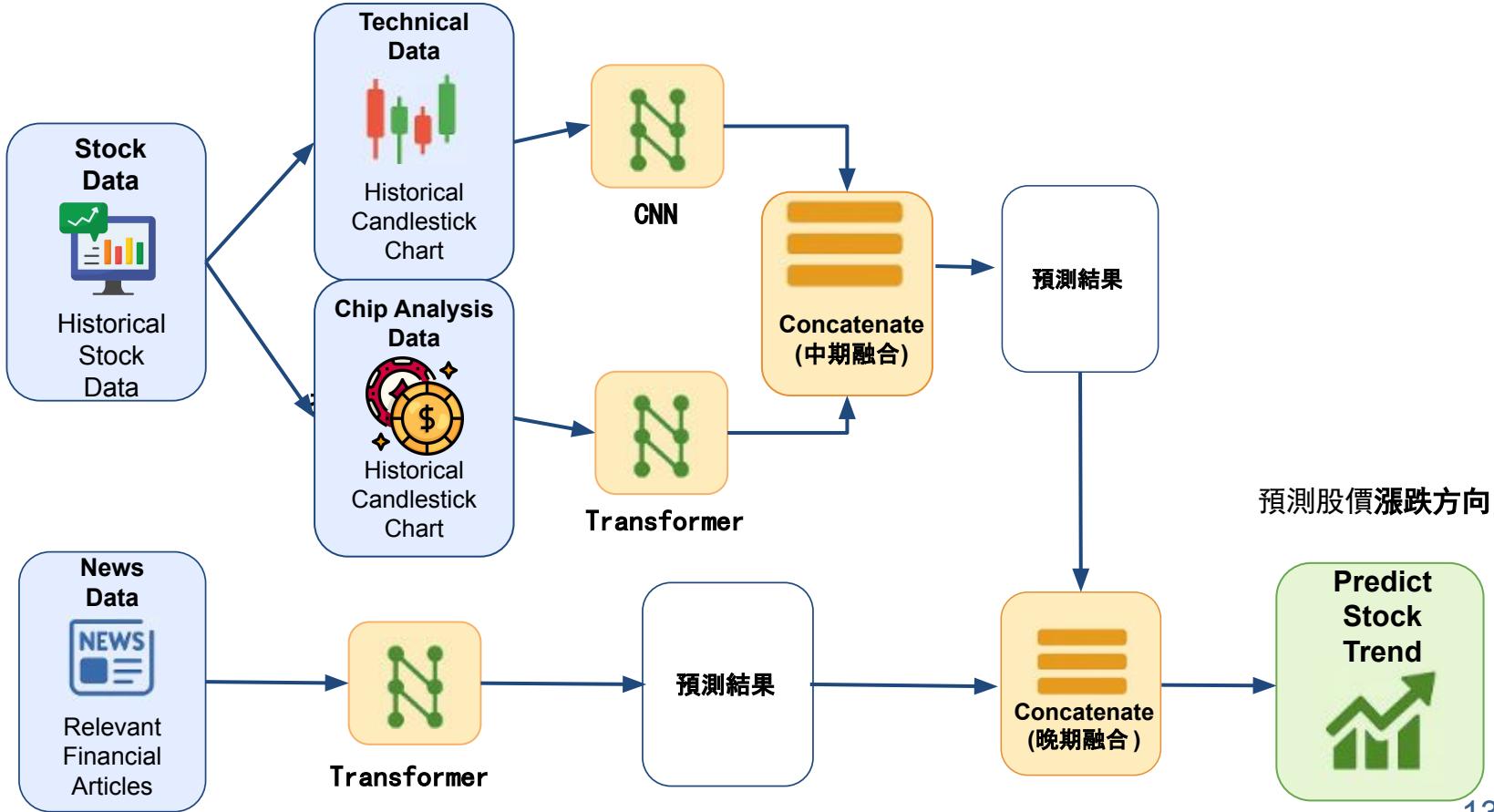
混合融合 (Hybrid Fusion)

- 定義：**結合早期融合、中期融合和晚期融合的優點，根據需求靈活設計融合流程。
- 優點：**兼具多種融合方法的特性，適合複雜的多模態資料分析場景。
- 缺點：**設計和實現的複雜性高。
- 應用案例：**模型同時處理技術面資料（如價格、成交量、波動率）與籌碼面資料（如外資持股比例、融資融券變化），各模態透過獨立深度模型提取特徵後，於特徵層與任務層共享資訊，再在決策層進行加權融合，生成同時考量技術信號與資金動向的綜合股價預測結果。



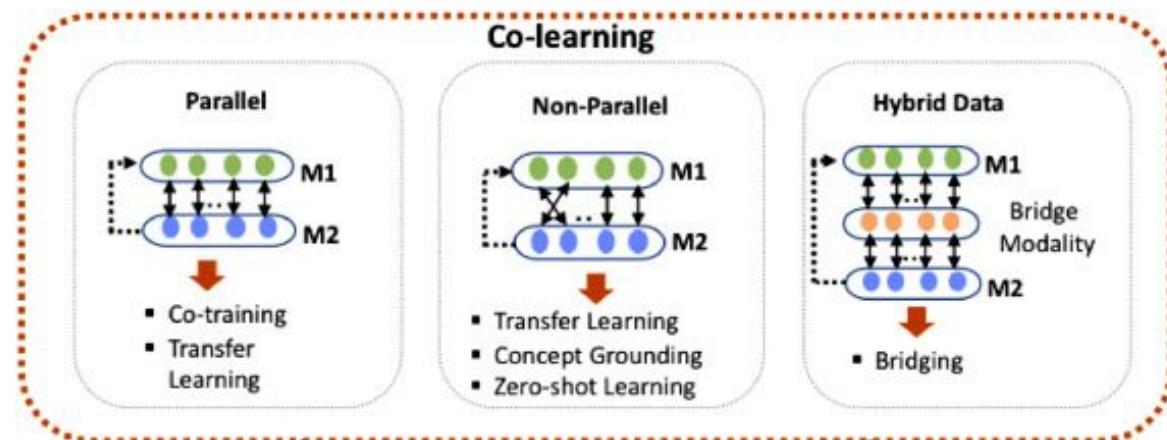
混合融合 (Hybrid Fusion)

輸入資料：
技術面
籌碼面



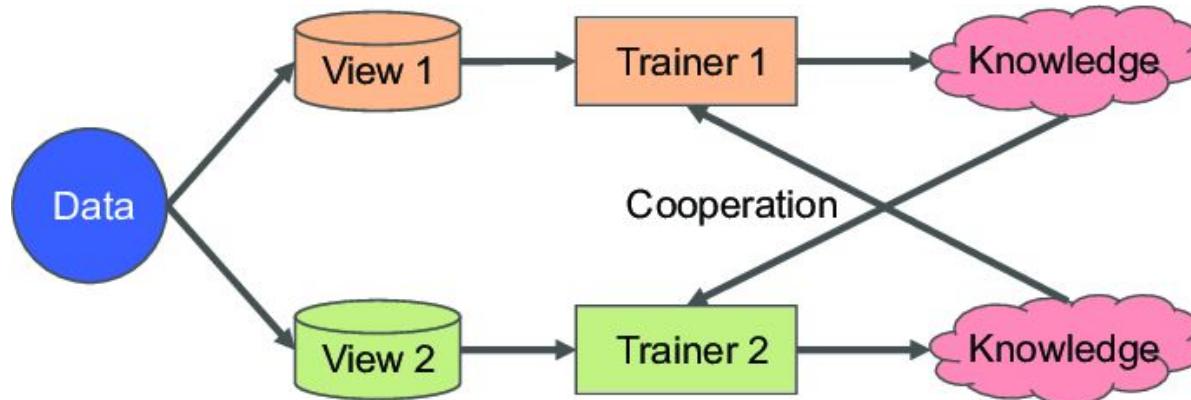
多模態-Co learning (協同學習)

- 結合利用一種模態的知識來幫助或增強另一種模態的學習，達成跨模態的互補與共同提升。
- **Parallel Data (平行資料)**
 - 兩種模態之間有明確的一對一對應（例如圖像-文字配對資料）
 - Co-training：雙方模型交替提供標籤，互相強化。
 - Transfer Learning：將一種模態學到的知識遷移到另一模態，提升弱模態的效果。
 - 適用於標註齊全、對應關係明確的資料集



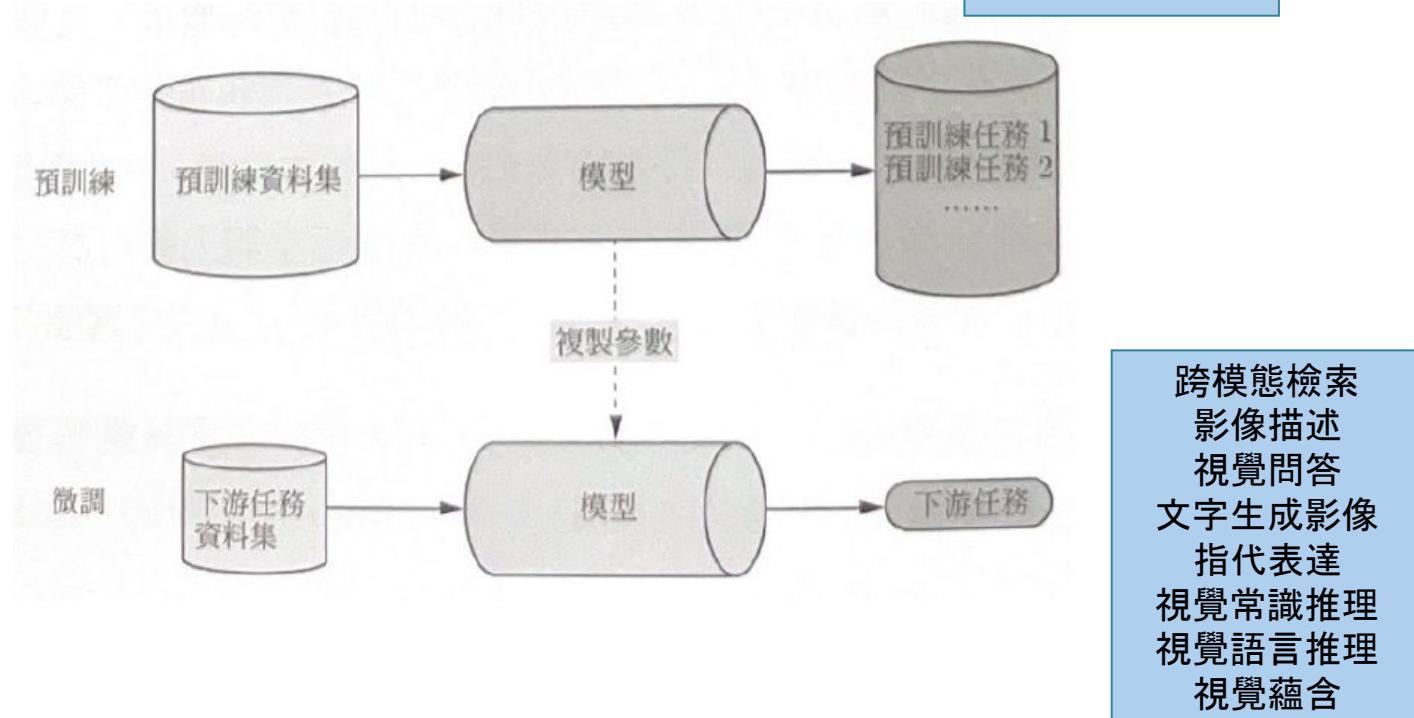
Co-training (協同訓練)

- 透過分別訓練多個模型來互相補充資訊，從而提高對未標記數據的分類性能。
- Trainer 1 與 Trainer 2 分別從 View 1、View 2 的資料中學習，形成各自的知識表示。
- 接著，兩個訓練器會進行 Cooperation (合作)：
 - Trainer 1 將自己從 View 1 學到的知識，用於協助 Trainer 2 的學習（例如產生 pseudo-label 或提供輔助特徵）。
 - Trainer 2 同樣將從 View 2 學到的知識，回饋給 Trainer 1，補充另一個模態的訊息。
- 透過這樣的互相教學與知識共享，雙方在迭代過程中不斷提升彼此的準確率與泛化能力。



多模態預訓練與遷移學習

➤ 整體框架



多模態預訓練與遷移學習

➤ 下游任務

- **視覺常識推理:**

- 紿定一張以隻很多物件區域的圖片
- 任務要求完成:根據問題選擇答案、解釋選擇該答案的原因

- **視覺語言推理**

- 紿定兩張圖片和一句文字描述
- 任務要求判斷文字描述是否正確地描述了兩張圖片的內容

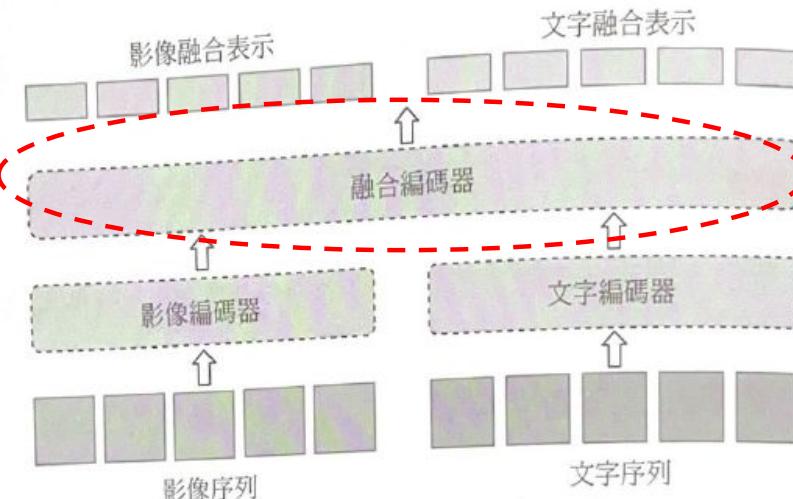
- **視覺蘊含:**

- 紿定一張圖片和一句文字描述
- 任務要求判斷由圖片(前提)推斷句子(假設)是否合適(蘊含/中立/矛盾)

多模態預訓練與遷移學習

➤ 模型結構-基於融合編碼器(encoder)

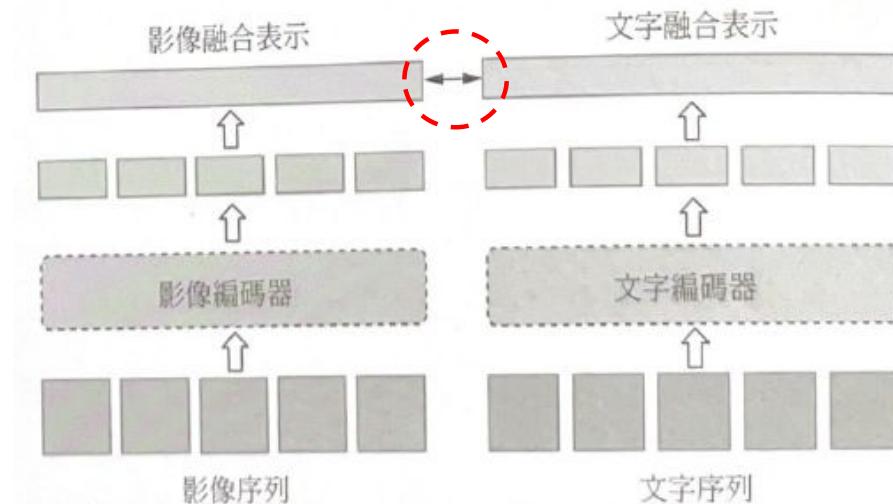
- 使用兩個單模態模型分別編碼影像資訊和文字資訊，之後再利用多模態融合模型對圖文單模態編碼進行建模，獲得圖文融合表示。
- 其中單模態模型往往採用標準的 transformer 模型，而最常用的多模態融合模型則採用交叉 transformer 模型。



多模態預訓練與遷移學習

> 模型結構-基於雙編碼器

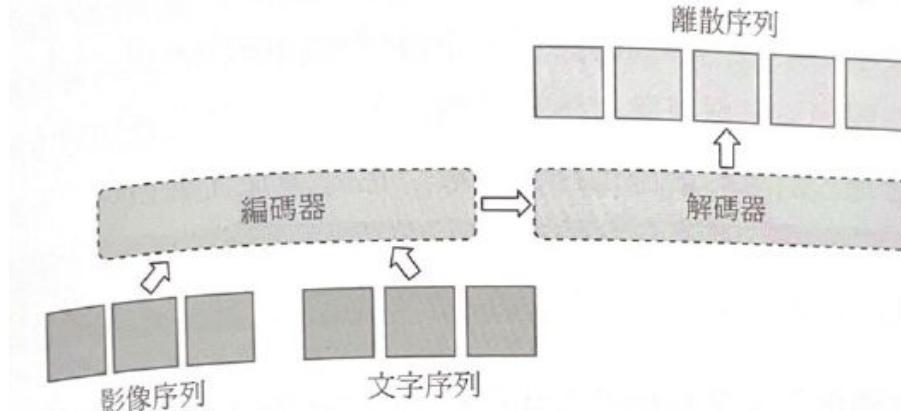
- 使用兩個單獨的編碼器分別學習影像和文字的對應表示
- 通常在對應表示空間中增加圖文相似性連結約束以建立圖文連結



多模態預訓練與遷移學習

➤ 模型結構-基於編解碼框架的模型

- 將影像序列和文字序列拼接成一個輸入序列，並將多個任務的輸出轉化成共用詞表的離散序列，最終使用編解碼模型學習輸入和輸出的連結。
- 此類模型不再關注通用多模態表示的學習，而是將不同任務的輸入/輸出轉化成統一的形式，以達到使用一個單一的模型同時建模多種任務的目標。



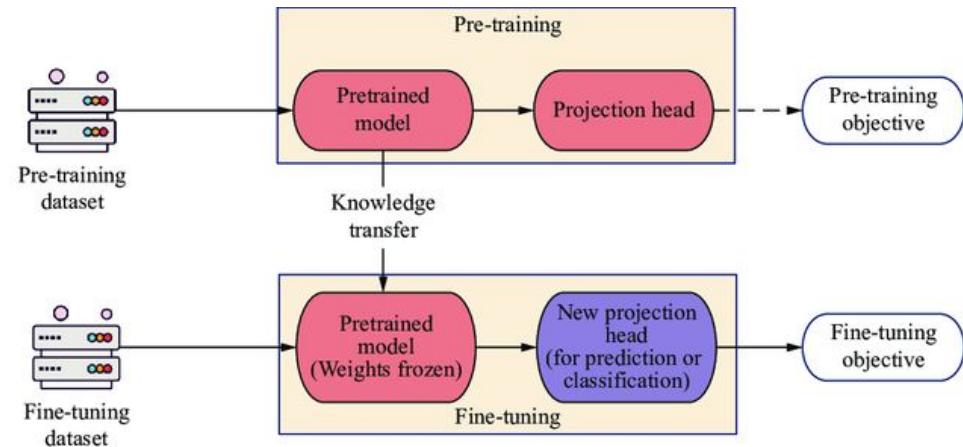
多模態預訓練與遷移學習

➤ 多模態預訓練與遷移學習的價值

多模態預訓練模型透過大量的圖文對資料進行訓練，學習到圖像與文字之間的對應關係。這些模型在訓練後，能夠將不同模態的資料映射到同一個語意空間，實現跨模態的理解與生成。遷移學習則允許這些預訓練模型在特定任務上進行微調，快速適應新的應用場景。

在商業智慧應用中，這種技術能夠

- 提升資料處理效率，減少人工標註成本。
- 加強模型的泛化能力，適應多變的商業環境。
- 實現跨模態的資料分析與決策支持。



Pre-training in Medical Data A Survey

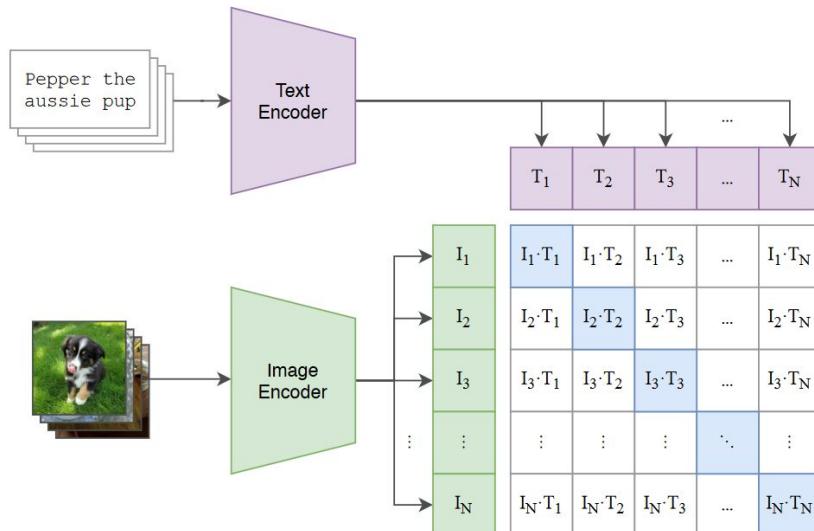
多模態預訓練與遷移學習—CLIP



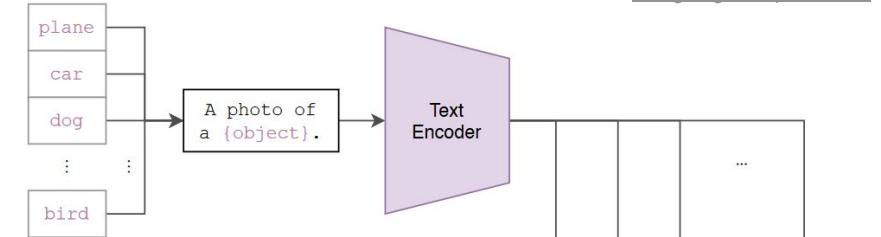
➤ CLIP: 對比式語言-圖像預訓練模型

CLIP (Contrastive Language - Image Pre-training) 由 OpenAI 開發，透過對比學習的方法，將圖像與其對應的文字描述映射到同一個嵌入空間。在訓練過程中，CLIP 同時處理大量的圖像與文字對，學習到圖像與文字之間的語意對應關係。

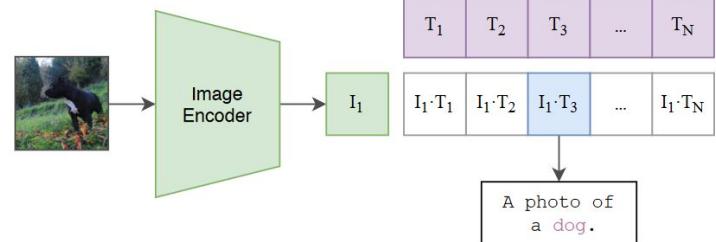
(1) Contrastive pre-training



(2) Create dataset classifier from label text



(3) Use for zero-shot prediction



CLIP Learning Transferable
Visual Models From Natural
Language Supervision

多模態預訓練與遷移學習—CLIP



➤ CLIP：對比式語言-圖像預訓練模型

CLIP (Contrastive Language - Image Pre-training) 由 OpenAI 開發，透過**對比學習**的方法，**將圖像與其對應的文字描述映射到同一個嵌入空間**。在訓練過程中，CLIP 同時處理大量的圖像與文字對，學習到圖像與文字之間的語意對應關係。

實際應用案例：電商平台的商品推薦系統

假設某電商平台希望提升商品推薦的準確性，考慮結合用戶的瀏覽行為（文字描述）與商品圖片（影像資料）。

應用方式：

- 使用 CLIP 模型，將商品的文字描述與圖片映射到同一個表示空間。
- 在推薦模型中，利用用戶的瀏覽歷史作為查詢，商品的多模態表示作為鍵和值，計算注意力權重，生成個性化的推薦列表。

預期成效：

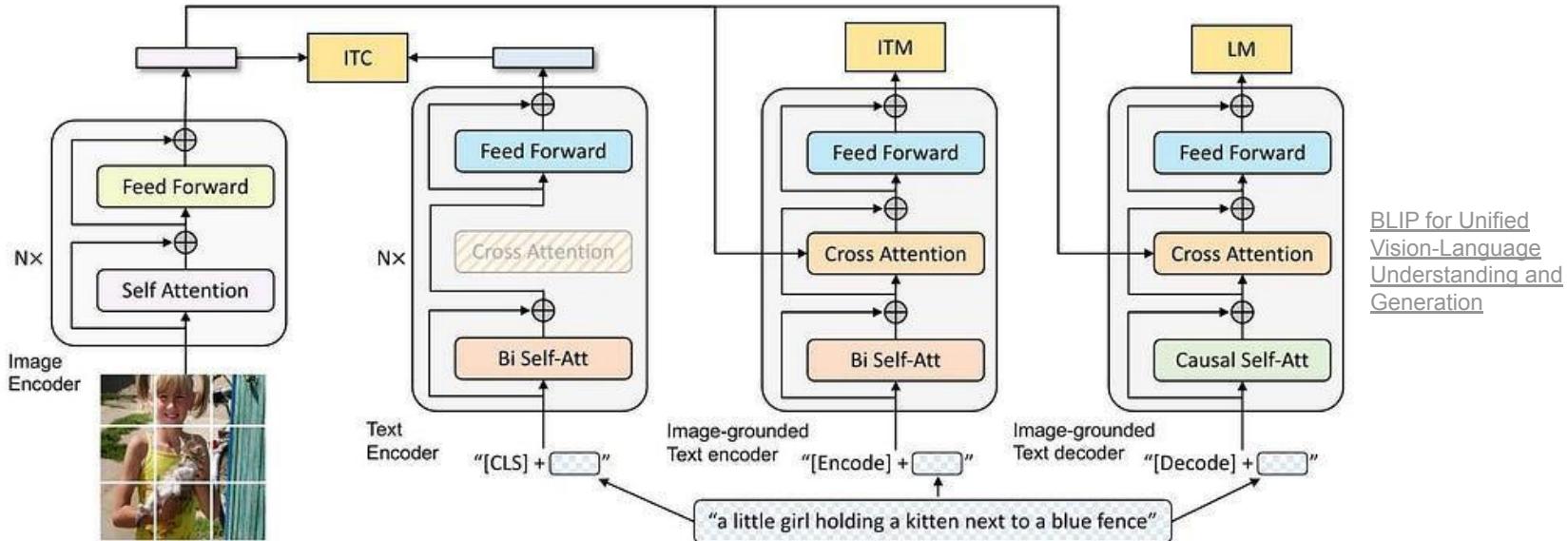
- 推薦點擊率提升 15%
- 用戶停留時間增加 10%

多模態預訓練與遷移學習—BLIP



➤ BLIP：語言-圖像預訓練的統一框架

BLIP (Bootstrapping Language-Image Pre-training) 由 Salesforce AI Research 開發，旨在統一視覺-語言的理解與生成任務。BLIP 引入了多模態的編碼器-解碼器架構，能夠同時處理圖像與文字資料，並進行跨模態的生成任務。



多模態預訓練與遷移學習—BLIP

➤ BLIP：語言-圖像預訓練的統一框架

BLIP (Bootstrapping Language-Image Pre-training) 由 Salesforce AI Research 開發，旨在統一視覺-語言的理解與生成任務。BLIP 引入了多模態的編碼器-解碼器架構，能夠同時處理圖像與文字資料，並進行跨模態的生成任務。



實際應用案例：金融風險評估系統

金融機構需要對大量的財務報告與新聞文本進行情緒分析，以評估潛在的風險。

應用方式：

- 使用 BLIP 模型，將財務報告中的圖表與文字描述進行整合，生成統一的語意表示。
- 對新聞文本進行情緒分析，識別可能影響市場的負面訊息。
- 結合上述資訊，建立風險評估模型，提供決策支持。

預期成效：

- 風險預測準確率提升 20%
- 風險識別時間縮短 30%

多模態預訓練與遷移學習

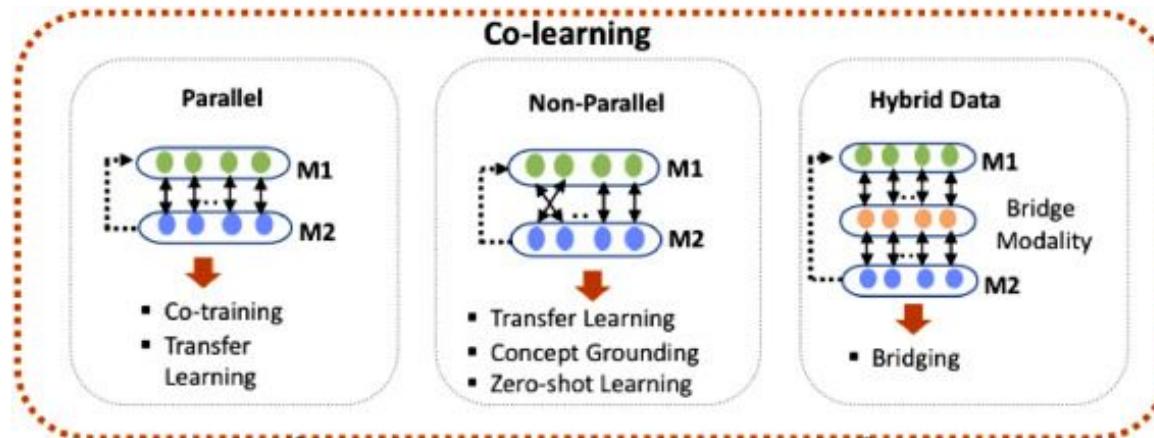
➤ CLIP VS. BLIP

模型	開發者	架構	主要特點	適用場景
CLIP	OpenAI 	雙編碼器	對比學習、零樣本學習	圖像分類、文本-圖像檢索
BLIP	Salesforce 	編碼器-解碼器	統一理解與生成、多任務學習	圖像描述、視覺問答、風險評估

多模態-Co learning (協同學習)

- Non-Parallel Data (非平行資料)

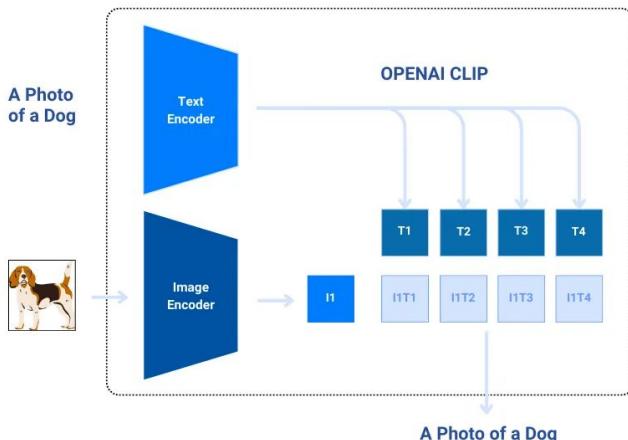
- 不同模態之間沒有直接對應關係（例如圖像資料集與文字資料集分開）
- Concept Grounding：將兩種模態映射到共同的語意空間，以建立間接對應。
- Zero-shot Learning：即使未見過配對資料，也能跨模態進行任務（如 CLIP 能以文字檢索圖片）。
- 適用於資料來源異質、缺乏對應的情境。
- 模型如CLIP、BLIP為代表



Concept Grounding (概念對應) - CLIP

- 將一種模態中的抽象語意對應到另一種模態的具體訊息
- 不需要平行資料，也能在共享概念空間中完成跨模態對齊
- 常用於 Non-parallel Co-learning，例如：
 - 文字概念「dog」對應到影像中的狗的特徵
 - 利用語意空間連結文字與視覺，使模型跨模態共享知識

輸入：左邊有一張「狗的照片」，以及一段對應的文字描述「A photo of a dog」



兩個編碼器 (Encoders) :

CLIP 使用 **Image Encoder** (圖像編碼器) 將圖片轉換成一個向量表示 (I_1)。

同時，使用 **Text Encoder** (文字編碼器) 把文字轉成文字向量 ($T_1 \sim T_4$)。

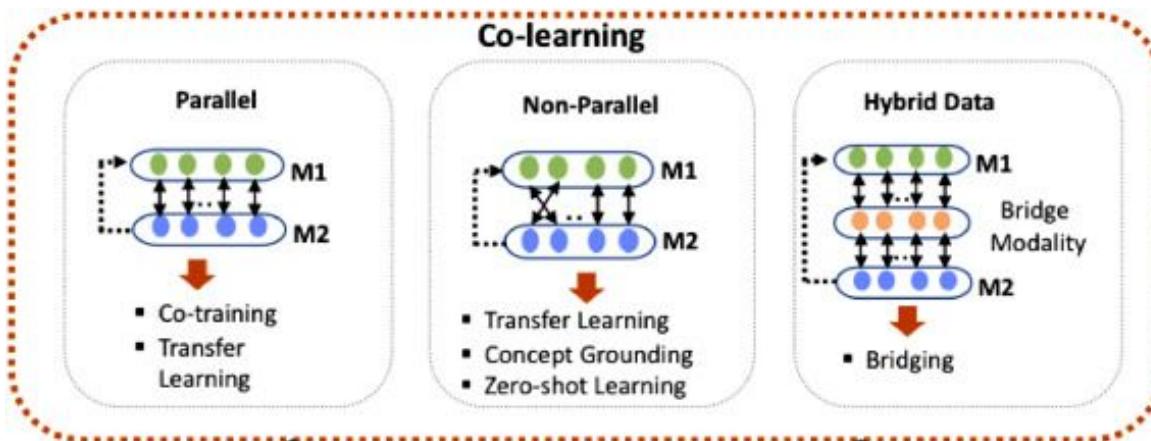
投影到共同語意空間 :

這兩種向量會被映射到 **同一個語意空間** 中。

目標是：相對應的圖文（例如「狗的照片」與「a photo of a dog」）要在這個空間中距離很近；不相符的圖文則距離遠。

多模態-Co learning (協同學習)

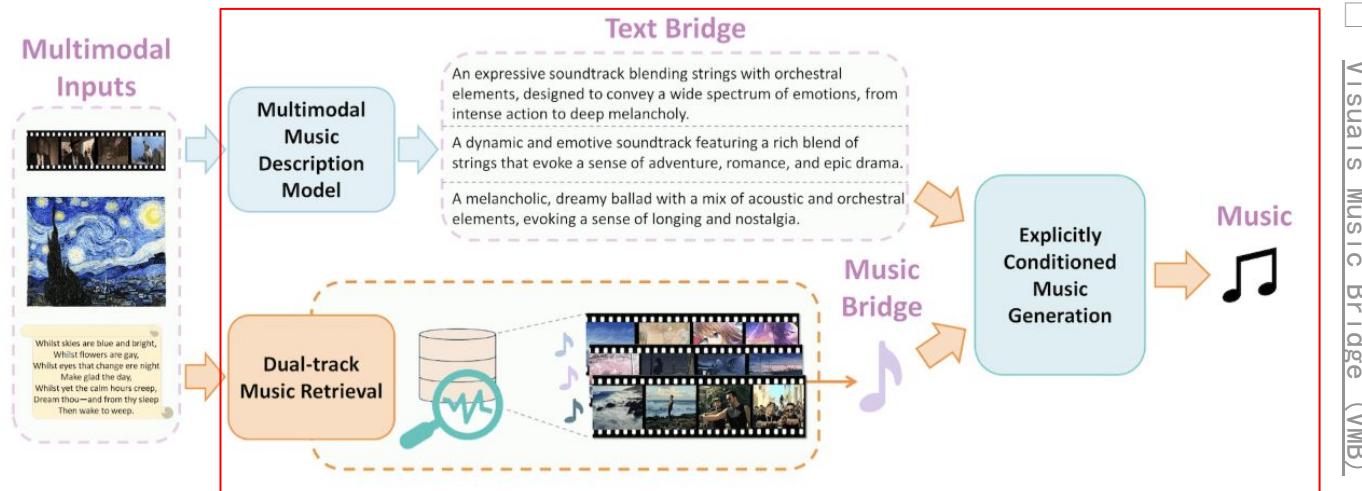
- Hybrid Data (混合資料)
 - 結合部分平行資料與大量非平行資料，是實務中最常見的情況
 - Bridging (橋接模態)：引入第三種「橋接模態」（如共同語意標籤、語音、meta feature），作為中介，幫助不同模態建立聯繫。
 - 適用於標註資料有限，但有大量單一模態資料的應用場景（例如醫療影像 + 病歷文字）。



Bridging (橋接模態) -Visuals Music Bridge (VMB)

圖像與音樂之間 沒有直接配對資料集，引入「第三方模態」作為橋樑
透過文字描述 (Text Bridge) 或音樂檢索 (Music Bridge) 間接建立跨模態對應

1. 輸入圖像 → 透過 **Multimodal Music Description Model** 產生描述性文字（如「一首交響弦樂與史詩冒險氛圍的曲子」）
2. 這些文字描述再被用來指導音樂生成模型 → 形成音樂
3. 相當於先把視覺內容轉成語意描述，再讓音樂模型依據文字生成對應音樂



圖像（電影畫面藝術圖像）
文字（詩句）

1. 文字輸入（如詩）→ 使用 Dual-track Music Retrieval (雙軌音樂檢索)
2. 從資料庫中找到對應音樂，進而引導音樂生成模型
3. 當於利用現有音樂片段作為「中間媒介」，實現跨模態對應

可解釋AI (XAI)

• 什麼是模型可視化？

模型可視化是指以圖形或可視化方式來呈現和理解機器學習模型的內部結構、運作方式和學習到的特徵。

這些視覺化工具和技術幫助我們更好地理解模型在處理資料時的行為，並有助於調試、優化和改進機器學習模型。



可解釋AI (XAI)

• 不同類型的模型可視化

- **資料探索** – 資料探索是使用探索性資料分析(EDA) 完成的。應用 T 分佈隨機鄰域嵌入 (t-SNE) 或主成分分析 (PCA) 技術來理解該特徵。
- **構建模型** – 用於衡量分類和回歸模型的各種指標。分類中使用準確率、精確率和召回率、混淆指標、對數損失和 F1 分數，回歸中使用均方誤差 (MSE)、均方對數誤差、均方根誤差 (RMSE)。構建模型後的所有這些指標都用於理解和衡量性能。
- **決策樹模型** – 靜態特徵摘要，例如從模型中檢索的特徵重要性。它僅存在於基於決策樹的算法中，例如隨機森林和XGBoost。
- **評估模型** – 評估模型的錯誤預測。

可解釋AI (XAI)

• 訓練期間的模型可視化

- **標量（損失和準確度）** – 標量可用於顯示訓練過程中誤差的趨勢。除了定期將損失和準確性記錄到標準輸出之外，我們還記錄並繪製它們以分析其長期趨勢。
- **直方圖** – 可視化模型圖中張量的分佈如何隨時間變化。顯示不同時間點張量的許多直方圖可視化。
- **權重和偏差** – 通過在直方圖上可視化來監控訓練期間的權重和偏差。
- **激活** – 為了使梯度下降發揮最佳性能，節點通常在激活函數分發之前輸出。
- **梯度** – 每層的梯度都可以可視化，以識別深度學習問題，例如梯度遞減或梯度爆炸問題。
- **圖表** – 圖表可視化模型的內部結構或體系結構。
- **圖像** – 訓練每一步的圖像意味著生成的中間圖像可以可視化並可視化張量。

可解釋AI (XAI)

- 為什麼模型可視化很重要？

- 有必要了解所有這些算法如何做出決策。
- 認識模型的基本特徵可以讓我們深入了解其內部運作方式，並為消除偏差和提高其性能提供方向。
- 有助於調試模型
- 提供預測解釋的主要原因是可解釋的機器學習模型對於獲得最終用戶的信任是必要的。

可解釋AI (XAI)

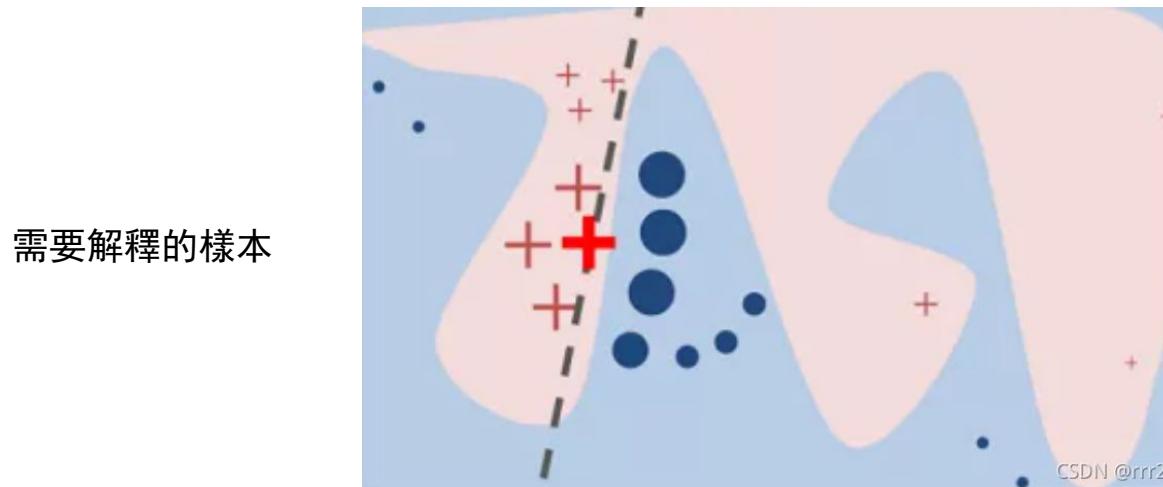
• 為什麼模型可視化很重要？

- 我們需要在大多數情況下進行解釋：
 - **可信度**—如果使用分類或預測結果，則需要像股票交易者一樣了解一些領域知識，以提供購買或出售特定股票的決策。
 - **透明度**—機器學習不能成為黑匣子，而應該為客戶、消費者和管理層提供模型結構和清晰度。就像開源一樣，模型理解也應該是開源的。
 - **責任**—模型應該有責任向消費者提供正確的答案。作為模型所有者，我們應該驗證模型特徵以保證其對決策的幫助。

LIME

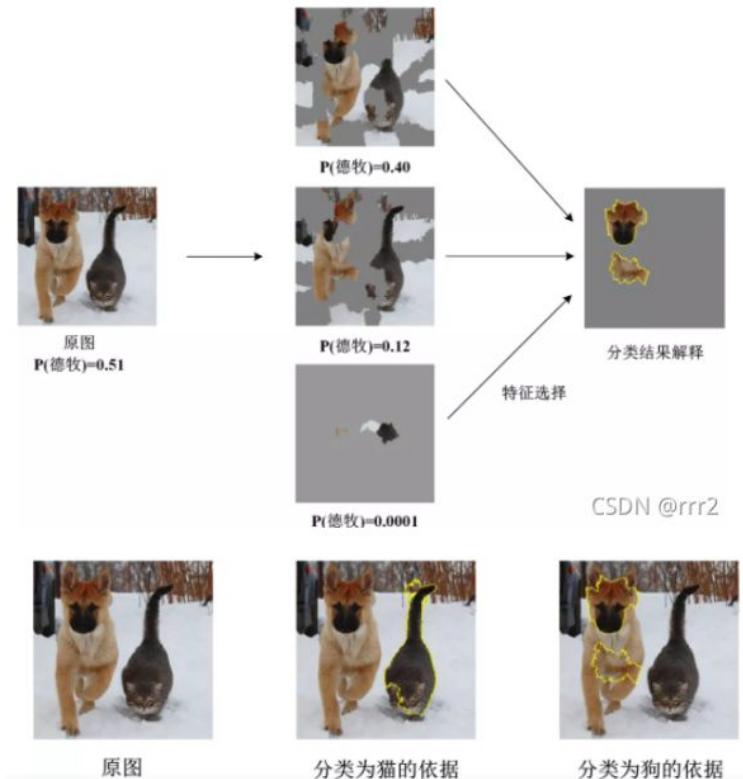
(Local Interpretable Model Agnostic Explanation)

定義：對一個複雜的分類模型（黑盒），在局部擬合出一個簡單的可解釋模型。



紅色和藍色區域表示一個複雜的分類模型（黑盒），從加粗的紅色十字樣本周圍採樣，將採樣出的樣本用分類模型分類並得到結果，同時根據採樣樣本與加粗紅十字的距離賦予權重。虛線表示通過這些採樣樣本學到的局部可解釋模型

LIME (Local Interpretable Model Agnostic Explanation)



從特徵的角度考慮，不再以單個像素為特徵，而是以超像素為特徵，整個圖片的特徵空間就小了很多

LIME找出對分類結果影響最大的幾個超像素，也就是說模型僅通過這幾個像素塊就已經能夠做出預測。

右圖為貓狗辨識舉例
將圖片利用超像素切割取得特徵，將各為狗的
特徵組合分析出結果

多模態可解釋分析 實際案例-LIME

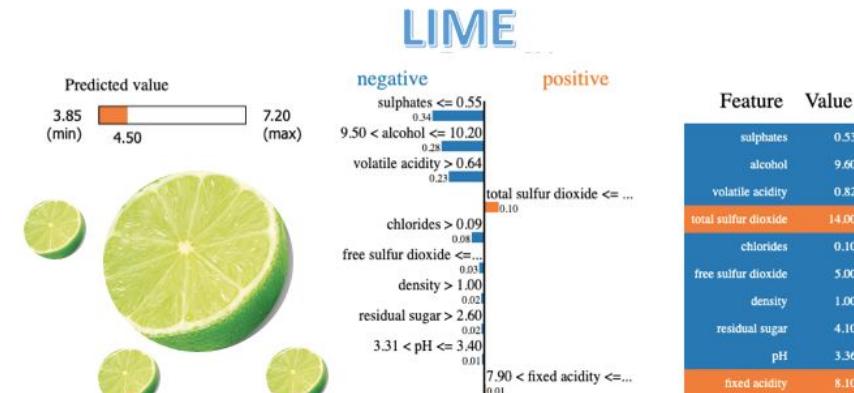
➤ 結構化資料：LIME 在信貸風險評估的應用

在信用風險管理中，銀行常使用機器學習模型（如 XGBoost）來預測借款人的違約風險。然而，這些模型的黑箱特性使得解釋預測結果變得困難。

- **LIME (Local Interpretable Model-agnostic Explanations)**

透過在特定資料點附近生成擾動樣本，**建立簡單模型解釋原始模型的預測**，適用於各種模型。

- LIME 通過在申請人資料附近生成擾動樣本，建立簡單模型解釋原始模型的預測，幫助銀行理解特定預測的原因。



[Explain Your Model with LIME. Compare SHAP and LIME | by Chris Kuo/Dr. Dataman](#)

多模態可解釋分析 實際案例-SHAP

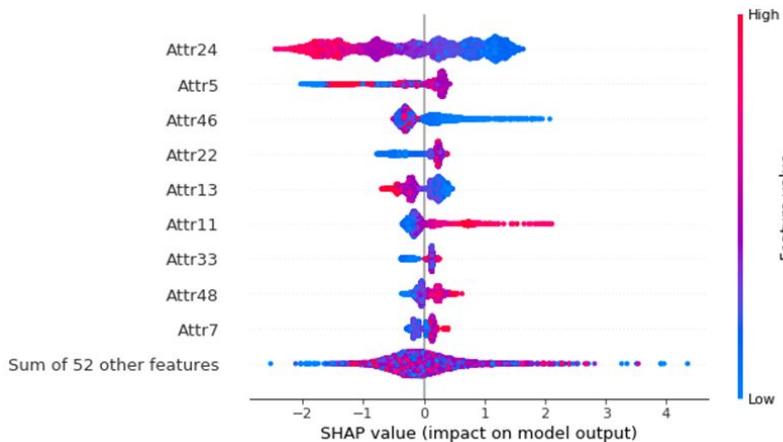
➤ 結構化資料：金融風控中的 SHAP 解釋

在信用風險管理中，銀行常使用機器學習模型（如 XGBoost）來預測借款人的違約風險。然而，這些模型的黑箱特性使得解釋預測結果變得困難。

- **可解釋性方法：SHAP (SHapley Additive exPlanations)**

SHAP 基於合作博弈論中的 Shapley 值，計算每個特徵對模型預測的貢獻度。

- SHAP 提供了整體特徵重要性排序，幫助銀行了解哪些變數（如收入、負債比、信用分數）對模型預測影響最大。
- 對於特定申請人，SHAP 可以指出該申請人哪些特徵值（如高負債比）導致模型預測其為高風險。



[\[1705.07874\] A Unified Approach to Interpreting Model Predictions](#)

多模態可解釋分析 實際案例—Grad CAM

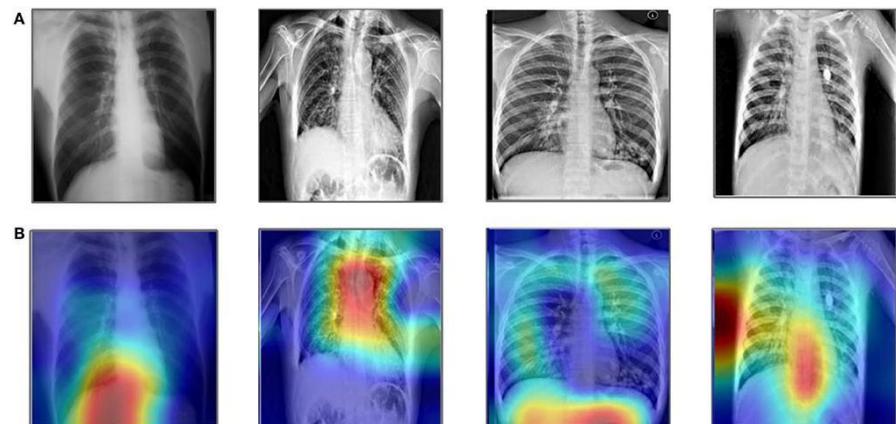
➤ 實際案例：肺炎 X 光影像診斷

醫院使用卷積神經網絡（CNN）模型分析胸部 X 光影像，以診斷患者是否患有肺炎。為了提高診斷的透明度和可信度，醫院引入了 Grad-CAM 方法來解釋模型的決策依據。

- **可解釋性方法：Grad-CAM (Gradient-weighted Class Activation Mapping)**

Grad-CAM 是一種視覺化技術，通過計算模型對特定類別的梯度，生成對應的熱力圖，顯示模型在影像中關注的區域。

- Grad-CAM 生成的熱力圖可以顯示模型在影像中關注的區域，幫助醫生確認模型是否關注了正確的肺部區域。
- 如果熱力圖顯示模型關注了非肺部區域，可能表明模型學習到了資料中的偏差，需進一步調整模型。



COVID-19 classification using chest X-ray images

多模態可解釋分析 實際案例—Grad CAM

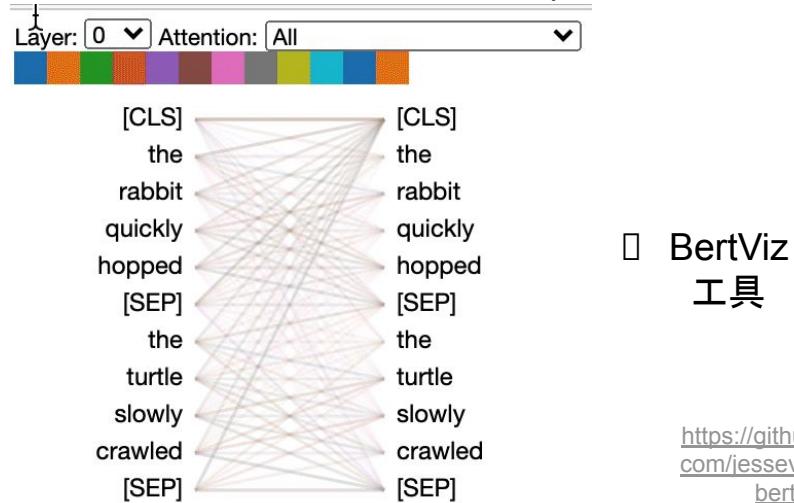
➤ 實際案例：產品評論的情感分析

電商平台使用 BERT 模型對產品評論進行情感分析，以了解顧客對產品的滿意度。為了提高模型的解釋性，平台引入了 Attention Heatmap 方法來可視化模型對評論中各詞語的關注程度。

- **可解釋性方法：Attention Heatmap**

在自然語言處理中，注意力機制（Attention Mechanism）允許模型在處理每個詞語時，根據其與其他詞語的關係分配不同的權重。注意力熱力圖（Attention Heatmap）可視化這些權重，幫助理解模型的決策過程。

- 可視化模型對評論中各詞語的注意力權重，幫助理解模型是如何做出情感分類的。
- 例如，模型可能對「優秀」、「糟糕」等情感詞給予較高的注意力權重。



多模態可解釋分析統整

模態	方法	說明
圖像	Grad-CAM, Integrated Gradients	顯示關鍵區域
文字	LIME, SHAP, Attention Visualization	顯示關鍵詞
時序	Occlusion, Saliency Map, Temporal SHAP	顯示關鍵時間段



一、多模態核心技術

二、K線型態和四大技術指標分類

技術指標與時間序列模型的定位

技術指標 (Technical Indicators) :

- 人工設計的特徵：基於領域知識提取的價格與成交量特徵
- 代表方法：移動平均 (MA)、相對強弱指標 (RSI)、MACD、布林通道
- 優勢：可解釋性強、計算快速、有實務經驗支持
- 限制：固定公式、無法自適應、難以捕捉複雜非線性關係

時間序列模型 (Time Series Models) :

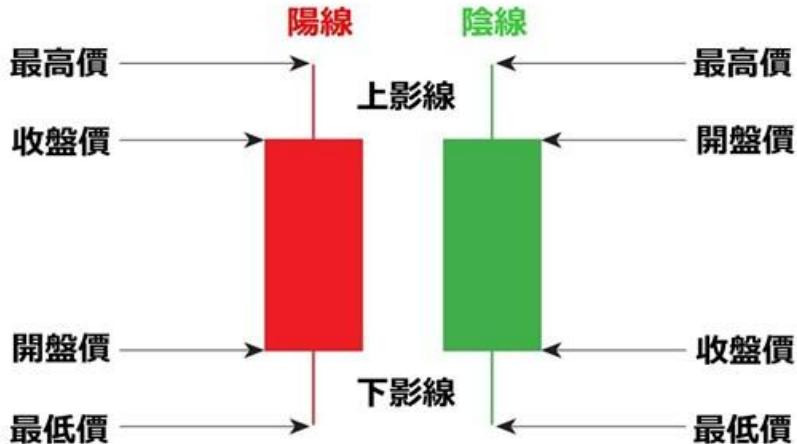
- 數據驅動的學習：從歷史數據中自動學習模式
- 代表方法：ARIMA、LSTM、Transformer
- 優勢：自動特徵學習、捕捉複雜動態、適應性強
- 限制：黑盒子、需要大量數據、可解釋性弱

為何需要整合？

- **互補性**：技術指標提供領域知識，模型提供學習能力
- **增強預測**：指標作為輸入特徵，提升模型性能
- **提升可解釋性**：指標幫助理解模型在關注什麼

K線圖的基本結構與應用

K線的組成：



- **開盤價 (Open)**：一個交易週期內的首個交易價格。
- **收盤價 (Close)**：一個交易週期內的最後交易價格。
- **最高價 (High)**：該週期內的最高交易價格。
- **最低價 (Low)**：該週期內的最低交易價格。

- K線由 **實體** 和 **上下影線** 組成，實體代表開盤價和收盤價之間的區域，影線表示最高價和最低價。

➤ 陽線與陰線的意義

- **陽線**：收盤價高於開盤價，表示價格上漲，通常以紅色表示。
- **陰線**：收盤價低於開盤價，表示價格下跌，通常以綠色或黑色表示。

➤ K線的時間週期選擇

K線圖可以基於不同的時間週期，例如1分鐘、5分鐘、日線、週線和月線。**選擇適當的時間週期有助於分析不同的市場趨勢**。

常見K線型態與應用

• 單根K線型態

• 錘子線 (Hammer)

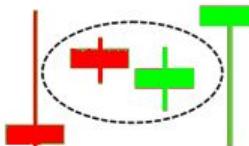
出現在下降趨勢的末端，具有短實體、長下影線，表示買方開始進入市場，可能出現反轉上漲。



↑ 潛在方向

• 十字線 (Doji)

開盤價和收盤價幾乎相同，市場多空力量均衡，可能預示趨勢反轉或持續。



↔ 潛在方向

• 吊人線 (Hanging Man)

出現在上升趨勢的末端，形態類似錘子線，但可能預示價格將下跌。



↓ 潛在方向

• 流星線 (Shooting Star)

出現在上升趨勢中，具有長上影線，表示上方壓力較大，價格可能下跌。



↓ 潛在方向

常見K線型態與應用

• 複數K線型態

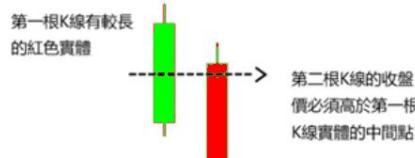
• 早晨之星 (Morning Star)

三根K線 組成的反轉型態。早晨之星出現在下降趨勢末端，預示價格上漲。



• 黃昏之星 (Evening Star)

黃昏之星出現在上升趨勢末端，預示價格下跌。

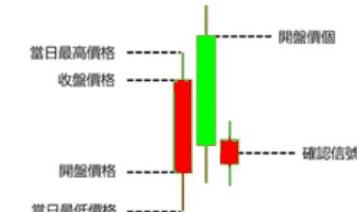


• 反轉形態 (Piercing Line/ Dark Cloud Cover)

在下降或上升趨勢中，價格出現強烈反向運動，可能預示趨勢反轉。

• 吞噬形態 (Bullish/Bearish Engulfing)

由**兩根K線**組成，第二根K線的實體完全包覆前一根K線的實體。**看漲吞噬** 預示價格上漲，**看跌吞噬** 預示價格下跌



一、趨勢型 (Trend Indicators)

核心概念：

- 用於偵測市場長期方向性（上升、下降或橫盤）。
- 強調「價格的慣性」——漲勢傾向繼續上漲，跌勢傾向繼續下跌。
- 適合「順勢交易」（Trend Following），不適合盤整區。

指標	公式 / 原理	關鍵訊號
SMA (Simple Moving Average)	$SMA_n = \frac{1}{n} \sum_{i=0}^{n-1} P_{t-i}$	價格上穿 SMA → 多頭趨勢
EMA (Exponential Moving Average)	$EMA_t = \alpha P_t + (1 - \alpha) EMA_{t-1}, \alpha = \frac{2}{n+1}$	EMA(12) > EMA(26) → 多頭
MACD	MACD = EMA(12) - EMA(26), Signal = EMA(9)	柱狀圖由負轉正 → 看多
ADX (Average Directional Index)	測量 +DI 與 -DI 的差異強度	ADX > 25: 有趨勢
Ichimoku 雲圖	五線計算：Tenkan、Kijun、SpanA/B、Chikou	價在雲上且 Tenkan > Kijun → 強多

一、趨勢型 (Trend Indicators)

特徵工程延伸：

- 均線距離比率 $(Close - MA20) / ATR14$ ($Close - MA_{\{20\}}$) / $ATR_{\{14\}}$ ($Close - MA20$) / $ATR14$
- 趨勢強度 (ADX 值、MACD 直方圖)
- 趨勢 Regime 標籤 ($ADX > 25 \rightarrow$ Trend; 否則 Range)

常見陷阱：

- 盤整時假突破多 (Whipsaw)
- 均線交叉有延遲性
- 趨勢策略常需結合波動濾波 (如 ATR 上升時才入場)

二、震盪型 (Momentum / Oscillator Indicators)

核心概念：

- 用於觀察價格動能與超買超賣狀態。
- 適合「逆勢交易」(Contrarian) 或盤整區間操作。
- 在趨勢明顯時反而容易產生誤導。

指標	公式 / 原理	關鍵訊號
RSI (Relative Strength Index)	$RSI = 100 - \frac{100}{1+RS}$, $RS = \frac{\text{平均漲幅}}{\text{平均跌幅}}$	RSI > 70 → 超買；RSI < 30 → 超賣
Stochastic Oscillator (%K, %D)	$\%K = (\text{Close} - \text{Lowest}) / (\text{Highest} - \text{Lowest}) \times 100$	%K > 80 超買, < 20 超賣；%K 與 %D 交叉判斷轉折
CCI (Commodity Channel Index)	$CCI = \frac{\text{Price}-\text{MA}}{0.015 \times \text{MeanDev}}$	正值高 → 偏強；負值低 → 偏弱
ROC (Rate of Change)	$ROC = (P_t - P_{t-n}) / P_{t-n} \times 100\%$	衡量短期動能加速度

二、震盪型 (Momentum / Oscillator Indicators)

特徵工程延伸：

- RSI、Stoch、CCI 的 z-score 化
- RSI 區域 One-Hot (超買、中性、超賣)
- 背離 (價格創新高但 RSI 下滑) → 反轉信號

常見陷阱：

- 趨勢市中 RSI 長期維持高檔/低檔
- Oscillator 適用盤整市、慎用於強勢單邊行情

三、波動型 (Volatility Indicators)

核心概念：

- 測量市場的「不確定性」與「風險幅度」。
- 並非告訴你漲跌方向，而是**風險水位與壓力/支撐區**。
- 常與趨勢指標搭配使用（例如：突破布林帶+均線上揚）。

指標	公式 / 原理	判讀方式
Bollinger Bands (布林通道)	上下軌 = 移動平均 (MA) $\pm k \times$ 標準差 (σ) 常見參數：期間 $n = 20$, 倍率 $k = 2$	- 價格 突破上軌 → 市場可能進入強勢或過熱階段 - 價格 觸及下軌 → 市場可能超賣或反彈 - 帶寬 變大 → 波動上升； 帶寬縮小 → 盤整收斂 (Volatility Squeeze)
ATR (Average True Range, 平均真實波幅)	$TR = \max \{H - L, H - C_{(t-1)} , L - C_{(t-1)} \}$	ATR 上升 → 市場波動擴大、不確定性提高。 ATR 下降 → 市場穩定、波動收斂。
HV (Historical Volatility, 歷史波動率)	對報酬率序列取標準差後年化： $HV = \sigma_{returns} \times \sqrt{252}$	- 衡量價格的 長期波動水準 (年化) - HV 高 → 市場劇烈波動、風險上升 - HV 低 → 市場平靜、波動收斂
Chaikin Volatility (柴金波動率)	$CV = \frac{EMA(H-L,10) - EMA(H-L,10)_n}{EMA(H-L,10)_n} \times 100\%$	- 比較「高低價差」的變化速率 - CV 上升 → 波動擴張，價格劇烈震盪 - CV 下降 → 波動收縮，市場轉趨平穩

三、波動型 (Volatility Indicators)

特徵工程延伸：

- 布林帶寬度比率 ($\text{BandWidth} = (\text{Upper}-\text{Lower})/\text{MA}$)
- 價格在通道內的位置 (Band%)
- ATR/Close 標準化波動
- 波動 regime: ATR 高於其長均線 → 高波動區

常見陷阱：

- 波動擴張 ≠ 一定反轉；須搭配方向性指標
- 單靠布林上軌賣、下軌買容易錯誤（要搭成交量或 RSI 濾波）

四、量能型 (Volume-based Indicators)

核心概念：

- 量能 (Volume) 代表「資金的參與度與信心」。
- 成交量增加 → 趨勢更具可信度。
- 價量背離是重要反轉訊號。

指標	公式 / 原理	判讀方式
OBV (On-Balance Volume)	上漲日加量，下跌日減量	價漲 OBV 升 → 健康上漲；價漲 OBV 跌 → 背離
MFI (Money Flow Index)	RSI × 成交量權重	>80 超買, <20 超賣
VWAP (Volume Weighted Avg Price)	$\frac{\sum Price \times Volume}{\sum Volume}$	價高於 VWAP → 強勢區
Volume Oscillator	(短期量均 - 長期量均)/長期量均	正值：量能擴張

四、量能型 (Volume-based Indicators)

特徵工程延伸：

- 成交量 z-score、Volume Rate of Change
- 價與 OBV 的差異率（量價背離指標）
- 價距 VWAP / ATR
- 量能 regime: Volume > MA(Volume, 20) → 高活躍期

常見陷阱：

- 成交量需配合時間或市場（開盤/收盤）調整
- 不同資產成交量不可直接比較（須相對標準化）

整合應用建議

組合	應用邏輯	對應策略
趨勢型 + 波動型	趨勢方向 + 波動濾波	順勢入場、ATR 停損
震盪型 + 趨勢型	趨勢過熱判斷	RSI 背離過熱時出場
趨勢型 + 量能型	趨勢確認	價漲量增 → 有效突破
量能 + 波動型	觀察市場活躍度	高波動+高量 → 關鍵事件期

技術指標 × 模型適配矩陣

模型	最適合指標	適合但有限	不適合指標	核心特點
ARIMA	趨勢型 (MA, EMA, MACD) 波動率型 (ATR, Bollinger)	成交量型 (OBV)	震盪型 (RSI, KD, CCI)	線性模型、適合趨勢與波動
LSTM	震盪型 (RSI, KD) 成交量型 (OBV, VWAP) 波動率型 (ATR)	趨勢型 (MA, MACD)	過多重複均線	非線性、捕捉長短期依賴
Transformer	多時間尺度組合 (5日/20日/60日 MA) 多模態 (指標+新聞+ESG)	波動率型 (ATR, BB)	單一線性指標	Attention 機制、跨週期整合
XGBoost	震盪型 (RSI, KD, CCI) 成交量型 (OBV, VWAP) 波動率型 (ATR, BB)	趨勢型 (MA, MACD)	原始價格序列	擅長捕捉非線性規則與交互作用

技術指標 × 模型適配矩陣 — 選擇原則

- ARIMA：少量、低共線性、解釋性高的趨勢與波動指標
- LSTM：多維度、非線性組合、時間依賴的指標
- Transformer：長序列、多時間尺度、多模態指標
- XGBoost：豐富特徵、非線性規則、交叉特徵組合

MA/MACD 與模型適配度

模型	適配度	使用建議	注意事項
ARIMA	★★★ (中等)	保留 1 個代表性趨勢指標（如 MACD）	多均線同時輸入 → 共線性問題
LSTM	★★★★★ (中高)	保留 MA/MACD 作為輔助特徵	可自行學習平滑/差分，MA/MACD 提供額外信號
Transformer	★★★ (中等)	可加入 1-2 個 MA/MACD 輔助	多均線冗餘 → Attention 分散
XGBoost	★★★★★ (高)	適合保留交叉特徵（如 $MA5 > MA20$, MACD 柱狀圖）	擅長捕捉均線交叉/死亡交叉規則

實務建議

ARIMA:

- 選擇單一代表性指標（如 MACD）避免共線性
- 適合捕捉趨勢，但過多均線會重複資訊

LSTM:

- 可保留多個 MA/MACD，模型能學習非線性組合
- 自動提取平滑與差分特徵，指標作為補充

Transformer:

- 適合不同週期的 MA（如 5日/20日/60日）
- Attention 機制能動態分配權重，但避免過多冗餘

XGBoost:

- 適合保留原始收盤價 + 多條均線 + 交叉特徵
- 擅長學習黃金交叉/死亡交叉等非線性規則
- 可處理大量特徵，無共線性問題

XGBoost (eXtreme Gradient Boosting)

介紹：

- Extreme Gradient Boosting：一種「梯度提升決策樹（Gradient Boosted Trees）」的高效實作。
- 概念：逐棵小樹修正前一次模型的錯誤（**集成學習 + 殘差修正**）。
- 每棵樹學習「前一次的錯誤」，讓整體模型逐步逼近真實值。

主要特點：

- 能處理非線性、特徵交互（如 RSI × ATR、MACD × Volume）。
- 支援遺漏值、自動特徵選擇。
- 內建正則化（L1 / L2）避免過擬合。
- 訓練速度快、可平行化（優於傳統 GBDT）。

為什麼適合金融資料？

- 不需序列平穩化，可直接用「特徵表格式」資料。
- 可混合多種輸入（價格技術指標、成交量、新聞情緒、Fear & Greed、VIX 等）。
- 可進行漲跌分類（二元）或**報酬預測**（回歸）。

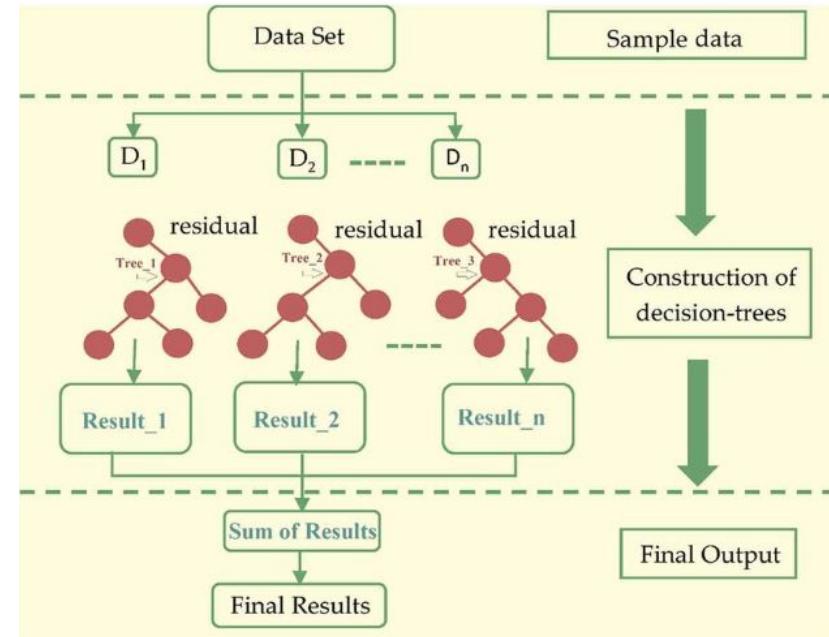
XGBoost 核心概念 — 逐棵修正殘差的「梯度提升法」

核心概念：

- 屬於「**集成學習 (Ensemble Learning)**」的一種
- 採用 **Boosting (提升法)**：逐棵樹學習前一棵的錯誤（Residual）。
- 每棵樹不是獨立訓練，而是「**連續修正誤差**」的過程。
- 最終將所有樹的預測結果加總，得到更準確的模型。

模型特點：

- 結合多棵弱樹（Weak Learners）形成強模型。
- 自動控制過擬合（內建正則化 L1/L2）。
- 適合非線性關係強、變數多的金融資料。
- 可用於分類（漲跌）與回歸（報酬率）問題。



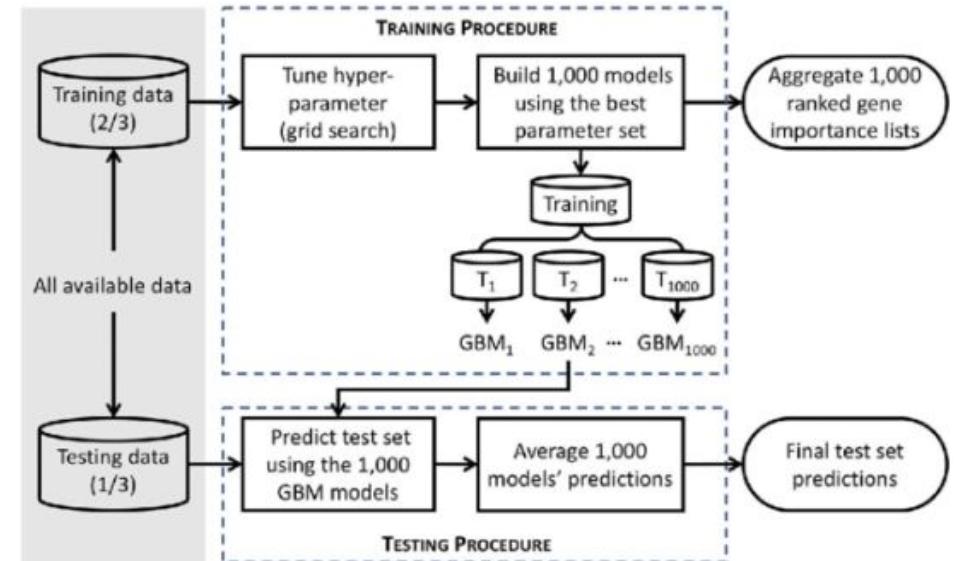
XGBoost 模型訓練與預測流程

訓練階段 (Training Procedure) :

- 將資料分為訓練集與測試集（如 2/3 與 1/3）。
- 進行 **超參數調整 (Grid Search)**，找出最佳組合。
- 建立多組 XGBoost 模型並訓練（不同隨機抽樣）。
- 聚合每個模型的特徵重要度 (Feature Importance)。

測試階段 (Testing Procedure) :

- 使用訓練好的模型預測測試資料。
- 平均多組模型的預測結果，得到最終輸出。
- 可用於日報酬預測、價格方向分類或信號強度分析。



XGBoost 在金融時間序列的應用

應用流程：

1. **原始資料收集**：OHLC、成交量、VIX、Fear & Greed、推文情緒。
2. **特徵工程**：建立技術指標（MA、RSI、MACD、BBands、ATR）。
3. **特徵矩陣建構**：轉為表格形式（每列一時間點，每欄一指標）。
4. **標籤生成**：下一期報酬率或漲跌方向。
5. **模型訓練**：使用 Walk-Forward 檢驗、Early Stopping 防過擬合。
6. **策略轉換**：根據預測機率決定買入 / 賣出 / 觀望信號。

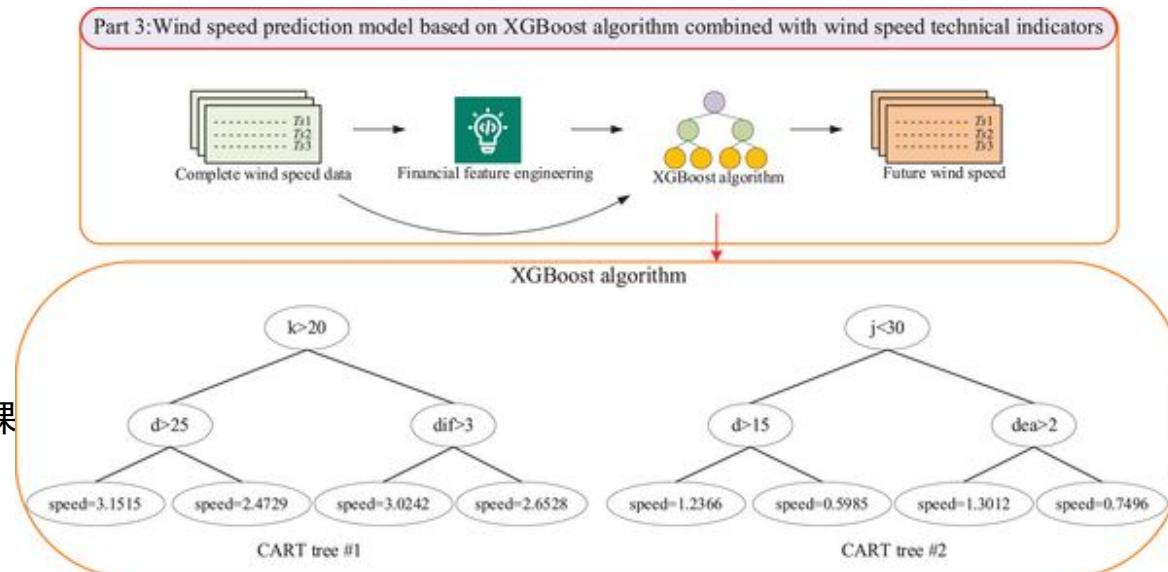
金融應用範例：

- 預測股價的「隔日漲跌方向」。
- 將情緒與技術特徵融合，作為 XGBoost 的輸入。
- 可延伸至多資產迴歸、風險預測或投資組合優化。

XGBoost 模型流程與內部結構示意

模型流程：

- 以完整的資料集（例：價格、技術指標、情緒分數等）作為輸入。
- 經過 **特徵工程 (Feature Engineering)**，轉換成模型可理解的特徵矩陣。
- 進入 **XGBoost 演算法核心**：多棵決策樹依序訓練，每棵樹學習前一棵的錯誤（Residual）。
- 模型最終輸出「下一期預測值」或「漲跌方向機率」。



ARIMA (AutoRegressive Integrated Moving Average, 自迴歸整合移動平均模型)

核心想法：

- 整合框架：結合自迴歸 (AutoRegressive, AR)、差分 (Integrated, I)、移動平均 (Moving Average, MA) 三大元素
- 處理非平穩：透過差分使數據平穩，再進行建模
- 線性組合：用過去的觀測值與預測誤差的線性組合預測未來
- 參數化模型：ARIMA(p, d, q) 三個參數控制模型複雜度

ARIMA模型的公式：

$$\left(1 - \sum_{i=1}^p \phi_i L^i \right) (1 - L)^d X_t = \left(1 + \sum_{i=1}^q \theta_i L^i \right) \varepsilon_t$$

其中 L 是滯後算子 (Lag operator) · $d \in \mathbb{Z}, d > 0$

模型組成 — ARIMA(p, d, q) 三大部分

AR (AutoRegressive, 自迴歸, p) :

- 當前值依賴過去 p 期的觀測值
- 捕捉數據的「記憶效應」

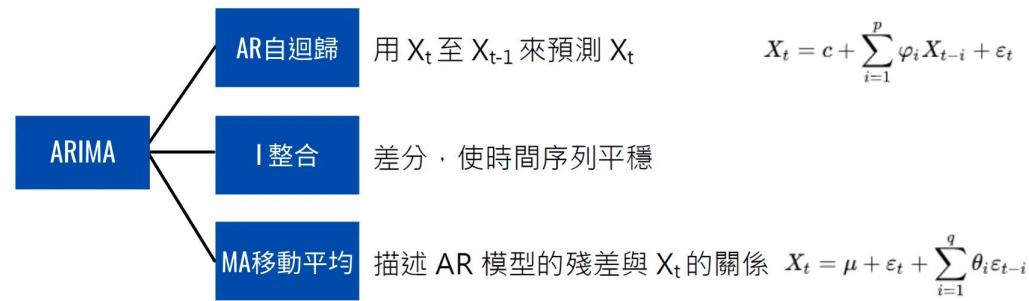
I (Integrated, 差分, d) :

- 對數據進行 d 次差分使其平穩
- 消除趨勢與季節性

MA (Moving Average, 移動平均, q) :

- 當前值依賴過去 q 期的預測誤差
- 捕捉短期波動與衝擊影響

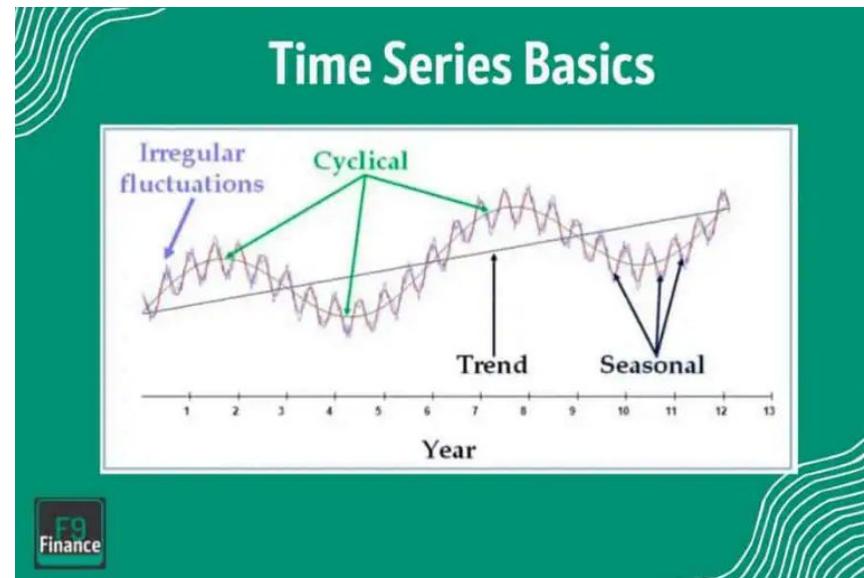
完整模型：先差分 d 次使數據平穩 → 再套用 ARMA(p, q) 模型



時間序列的基本構成 (Time Series Basics)

時間序列通常可分為四種主要成分：

- **趨勢 (Trend)**
 - 長期上升或下降的變化方向，反映整體成長或衰退。
- **季節性 (Seasonal)**
 - 固定週期內重複出現的變動（如季度銷售高峰、節慶效應）。
- **週期性 (Cyclical)**
 - 非固定週期的波動，常與經濟循環或外部因素有關。
- **不規則波動 (Irregular / Noise)**
 - 隨機且不可預測的短期擾動，通常由突發事件造成



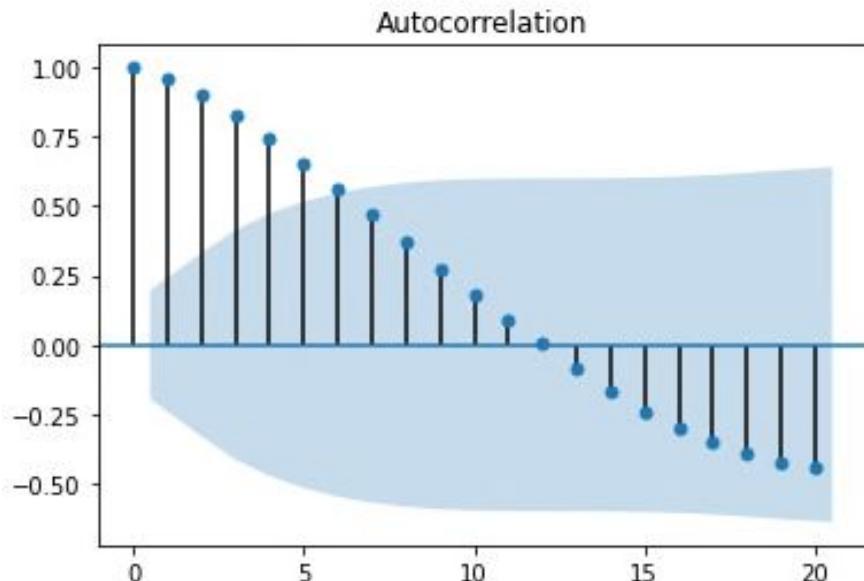
自相關函數 (ACF) 圖

用途：

- 檢查時間序列是否具有自相關特性
- 判斷資料是否平穩，是否需要差分處理
- 幫助決定 ARIMA 模型中 AR（自迴歸）項的階數

觀察重點：

- 若自相關係數緩慢衰減 → 資料「非平穩」
- 若迅速掉落到 0 附近 → 資料「近似平穩」



優缺點

優點：

- 統計理論完備、可解釋性強
- 計算效率高、工具成熟
- 適用於簡單、平穩的時間序列

缺點：

- 線性假設：無法捕捉非線性關係
- 平穩性要求：需要數據平穩
- 單變量限制：難以整合多維資訊
- 短期記憶：無法捕捉長期依賴

應用領域

- 金融市場：
 - 股票價格
 - 汇率
 - 利率交易量預測
- 經濟指標：
 - GDP 成長率
 - 失業率
 - 通膨率預測
- 其他領域：
 - 銷售預測
 - 需求預測
 - 用電量預測

定位

作為 Baseline 的價值：

- 提供可靠的基準線，評估深度學習模型的改進程度
- 對簡單、平穩的時間序列仍有競爭力
- 可解釋性高，便於理解與溝通

LSTM (Long Short-Term Memory, 長短期記憶網路)

核心想法

- 從 RNN 到 LSTM 的演進：
 - RNN (Recurrent Neural Network, 循環神經網路)：第一代序列深度學習模型，但存在梯度消失問題
 - LSTM (Long Short-Term Memory, 長短期記憶網路)：透過門控機制解決 RNN 無法捕捉長期依賴的問題
- 核心突破：
 - 門控機制：選擇性地記憶或遺忘資訊
 - 細胞狀態：長期記憶的儲存通道
 - 非線性建模：打破 ARIMA 的線性假設
 - 自動特徵學習：從原始數據中自動提取時間模式

模型組成 — LSTM 的三大門控機制

遺忘門 (Forget Gate) :

- 決定要從細胞狀態中丟棄哪些資訊
- 控制「忘記多少過去的資訊」

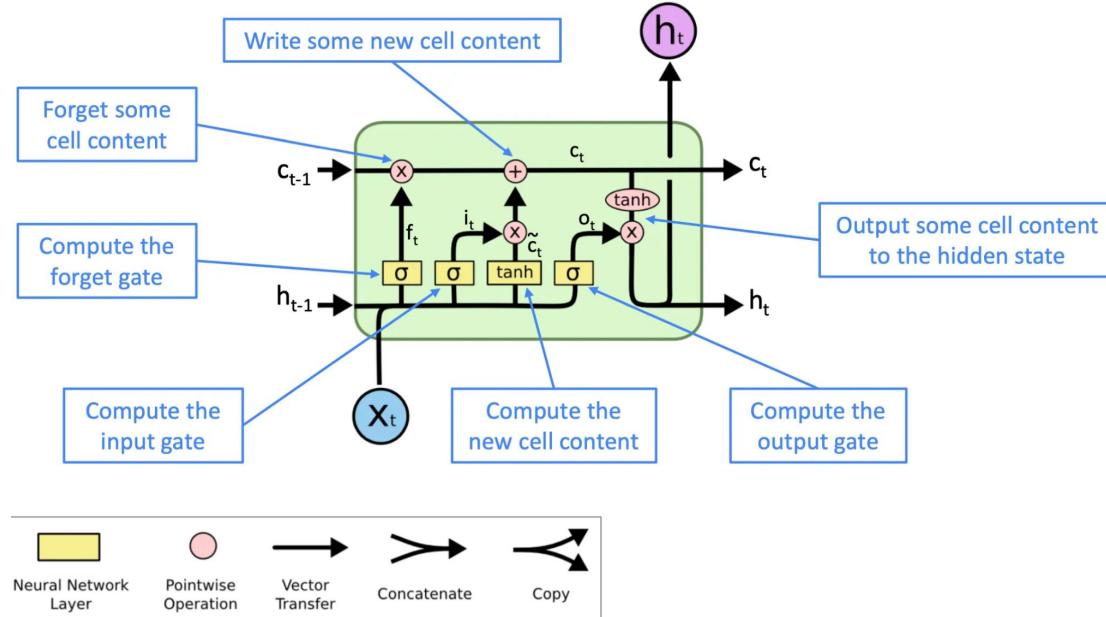
輸入門 (Input Gate) :

- 決定要將哪些新資訊儲存到細胞狀態
- 控制「記住多少新資訊」

輸出門 (Output Gate) :

- 決定要輸出細胞狀態的哪些部分
- 控制「輸出什麼資訊給下一層」

You can think of the LSTM equations visually like this:



模型組成 — LSTM 的兩種內部狀態

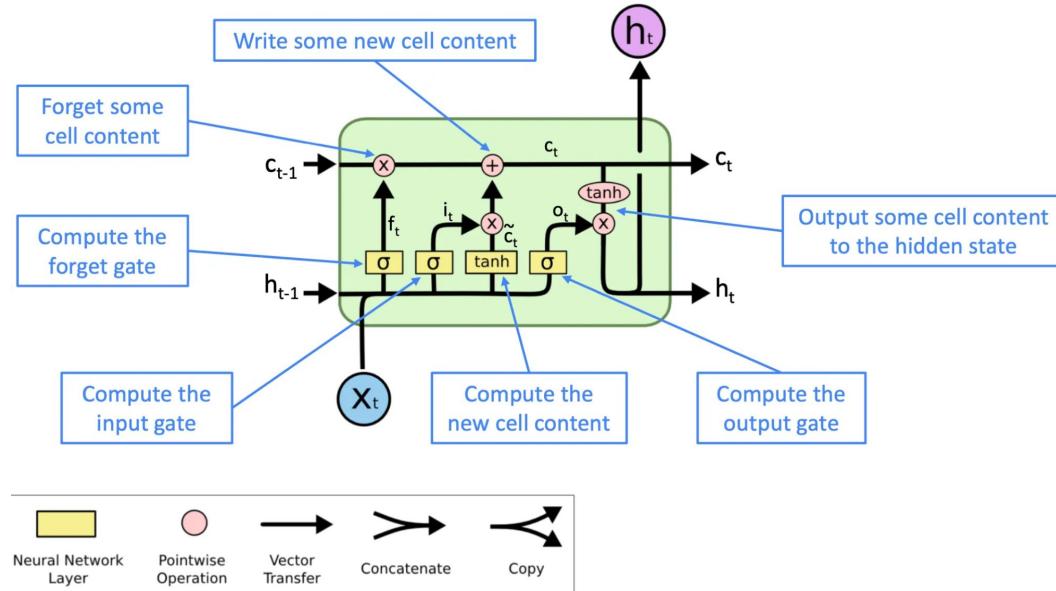
細胞狀態 (Cell State) :

- 長期記憶的「高速公路」
- 資訊可以在時間軸上流動而不衰減
- 保存重要的長期依賴資訊

隱藏狀態 (Hidden State) :

- 短期記憶
- 傳遞給下一時間步和輸出層
- 結合當前輸入與長期記憶

You can think of the LSTM equations visually like this:



模型組成 — 處理時間序列的流程

1. 輸入序列:

- $[x_1, x_2, \dots, x_t]$ (如: 過去 60 天的股價)

2. 逐步處理:

- 每個時間步, LSTM 接收當前輸入 x_t 和前一時刻的隱藏狀態 $h_{\{t-1\}}$
- 三個門控決定如何更新細胞狀態和隱藏狀態
- 細胞狀態在時間軸上傳遞長期資訊

3. 輸出預測:

- 最後一個時間步的隱藏狀態 h_t 用於預測未來值

關鍵特性:

- 選擇性記憶: 門控機制自動學習哪些資訊重要
- 長期依賴: 細胞狀態可以保留數十甚至上百個時間步的資訊
- 梯度穩定: 門控結構緩解梯度消失問題

建模流程

1. 資料準備與前處理

- **序列切分**：將時間序列依照滑動視窗方式切分成（輸入段、預測目標）
- **標準化**：將特徵縮放至 $[0, 1]$ 或 $[-1, 1]$ 範圍，確保各變數尺度一致
- **資料分割**：依時間順序區分訓練集、驗證集、測試集，避免資料洩漏
- **特徵整理**：可加入技術指標、情緒特徵、或外部變數（如VIX、成交量）

2. 模型架構設計

- **模型選擇**：LSTM、GRU、Transformer 等時序模型
- **層數設計**：調整隱藏層深度與神經元數量以控制表現與複雜度
- **正則化策略**：加入 Dropout、LayerNorm 或 Early Stopping 防止過擬合
- **輸出層設計**：Dense 層產生連續型預測值（如價格或報酬率）

建模流程

3. 模型訓練與優化

- **損失函數**: MSE (均方誤差) 或 MAE (平均絕對誤差)
- **優化器**: Adam、RMSprop 等自適應學習率方法
- **超參數設定**: 學習率 (lr)、批次大小 (batch size)、訓練回合數 (epochs)
- **監控策略**: 追蹤驗證集誤差、觀察過擬合情形

4. 模型評估與測試

- **性能指標**: RMSE、MAE、MAPE 等衡量預測精度
- **可視化分析**: 比較預測曲線與實際價格，觀察模型是否能跟隨趨勢
- **誤差診斷**: 分析殘差分佈、檢查是否存在系統性偏差
- **跨期驗證**: 以不同時間區間進行測試，檢驗模型穩定性

案例說明：預測股票收盤價

- 情境：
 - 使用過去 60 天的收盤價預測明天的價格
 - 數據: [Day 1 價格, Day 2 價格, . . . , Day 60 價格]
- 建模步驟：
 - 數據準備：
 - 輸入：過去 60 天的價格序列
 - 輸出：第 61 天的價格
 - 標準化：將價格縮放到 [0, 1] 範圍
 - 模型架構：
 - LSTM 層 1: 128 個單元（捕捉複雜模式）
 - LSTM 層 2: 64 個單元（更高層次的特徵）
 - Dense 層：輸出預測值
 - 訓練：使用歷史數據訓練模型
 - 預測：輸入最近 60 天 → 預測明天價格

LSTM 的優勢體現：

- 自動學習「前幾天的急跌可能影響未來反彈」
- 捕捉「20 天前的重大事件仍對今天有影響」
- 不需要手動設計特徵（如 ARIMA 的 p, d, q）

優缺點

優點：

- 捕捉長期依賴：可以記住數十甚至上百個時間步的資訊
- 非線性建模：處理複雜的非線性市場動態
- 自動特徵學習：不需要手動特徵工程
- 多變量整合：可以同時輸入價格、交易量、技術指標等

缺點：

- 需要大量數據：通常需要數千到數萬筆訓練樣本
- 訓練成本高：訓練時間長、需要更多計算資源
- 可解釋性較弱：模型是「黑盒子」，難以解釋預測原因
- 超參數調整複雜：層數、單元數、學習率等需要調整
- 序列處理限制：仍需按時間順序逐步處理，無法完全平行化

應用領域

- 金融市場：
 - 股票價格預測（日內、日度、週度）
 - 交易訊號生成（買入/賣出時機）
 - 波動率預測（VIX 指數）
 - 投資組合優化
- 多變量預測：
 - 結合價格 + 交易量 + 技術指標
 - 整合新聞情緒分析
 - 跨市場關聯預測（股市 + 匯率 + 商品）
- 其他時間序列：
 - 需求預測、能源消耗預測
 - 異常檢測（欺詐交易、設備故障）

定位

作為 Baseline 的價值：

- 深度學習時間序列預測的經典方法，評估新模型的標準基準
- 非線性建模能力，突破傳統統計方法的線性限制
- 自動特徵學習，減少人工特徵工程需求
- 長期依賴捕捉，適合複雜市場動態建模

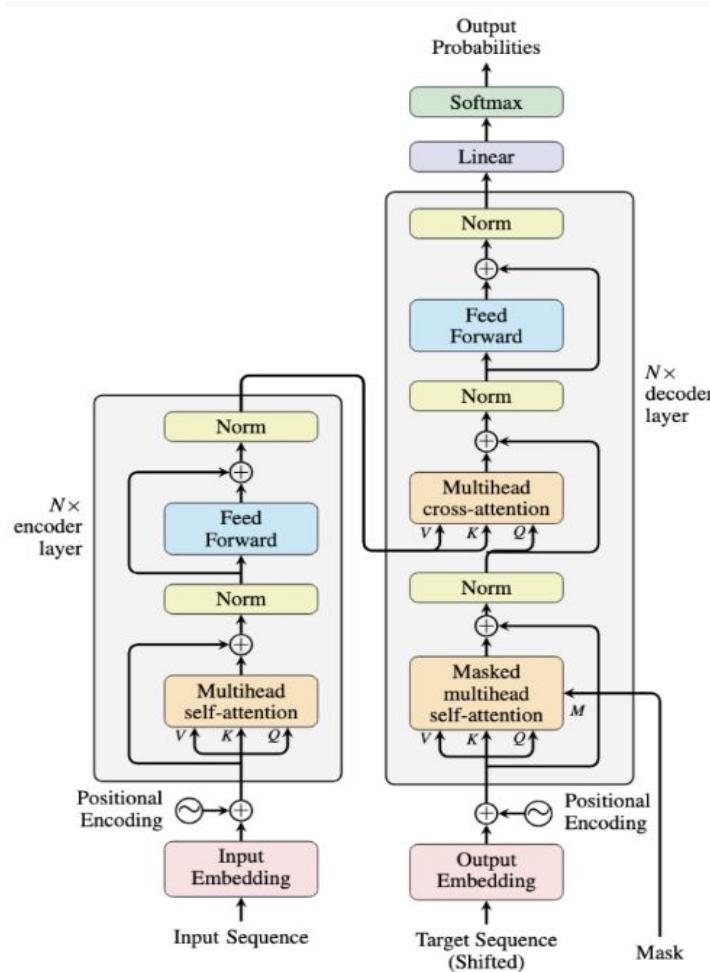
Transformer

核心想法

- 從 LSTM 到 Transformer 的演進：
 - LSTM 的限制：序列處理無法平行化、長序列效果下降
 - Transformer 的突破：完全拋棄循環結構，使用自注意力機制（Self-Attention）
- 核心創新：
 - 自注意力機制：同時關注序列中所有位置，捕捉全局依賴
 - 平行化計算：所有時間步可以同時處理，大幅提升訓練效率
 - 位置編碼：用數學方式編碼時間順序資訊
 - 多頭注意力：從多個角度理解序列模式

Transformer 整體架構圖

- Encoder: 理解整段輸入序列的時間關聯
- Decoder: 根據過去的預測與 Encoder 輸出進行生成
- Multi-Head Attention: 同時從不同角度觀察序列特徵
- Positional Encoding: 補足序列位置信息
- Mask 機制: 防止模型偷看未來資訊



模型組成 — 注意力機制

自注意力機制 (Self-Attention) :

- 計算序列中每個位置與其他所有位置的關聯性
- 自動學習「哪些時間點對預測最重要」
- 可以直接捕捉長距離依賴

多頭注意力 (Multi-Head Attention) :

- 使用多個注意力頭從不同角度理解數據
- 每個頭關注不同的特徵模式
- 綜合多個視角得到更豐富的表示

位置編碼 (Positional Encoding) :

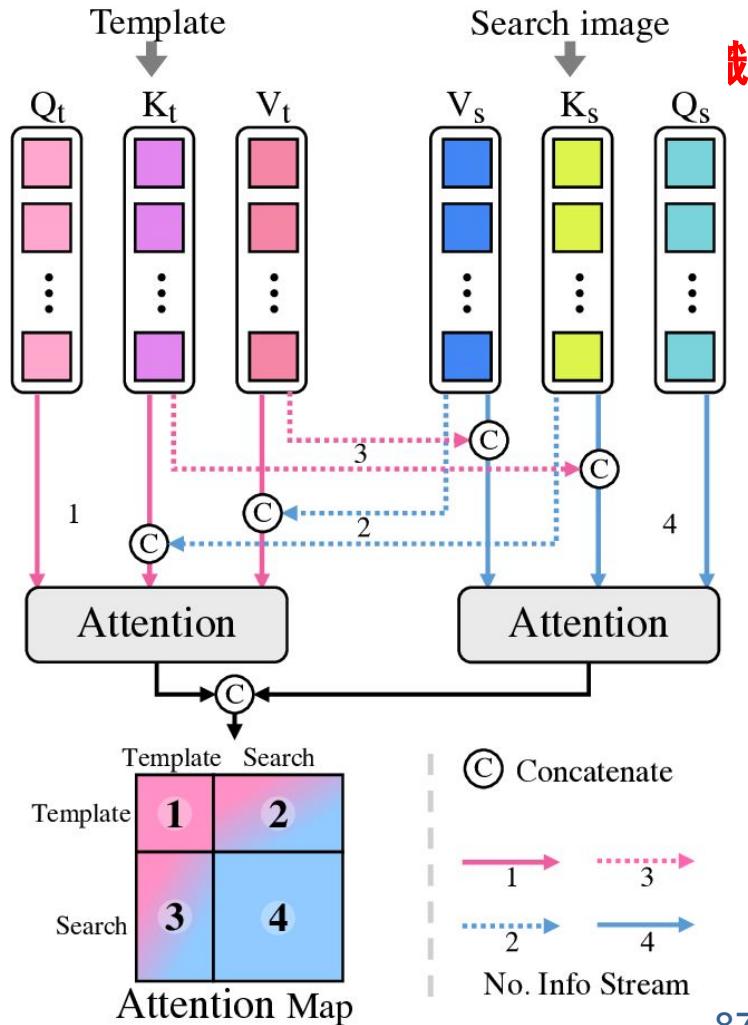
- 加入時間順序資訊（因為自注意力本身不考慮順序）
- 使模型理解「第 1 天」和「第 60 天」的位置差異
- 保留時間序列的時間特性

注意力機制(Q - K - V 資訊流)

- Query (Q)：代表「我想知道什麼」
- Key (K)：代表「資訊的標籤」
- Value (V)：代表「實際內容」
- 注意力機制：

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^T}{\sqrt{d_k}} \right) V$$

- Attention Map：顏色越深代表關注度越高
- 每個時間點都會問：『在整段序列裡，誰跟我最相關？』模型透過 Q - K - V 計算出權重，再加權所有資訊



模型組成 — 架構組件

前饋神經網路 (Feed-Forward Network) :

- 對每個位置獨立進行非線性轉換
- 增強模型的表達能力
- 通常包含兩層全連接層

殘差連接 (Residual Connection) :

- 將輸入直接加到輸出上: $\text{output} = F(x) + x$
- 緩解梯度消失問題
- 使深層網路更容易訓練

層標準化 (Layer Normalization) :

- 標準化每層的輸出，穩定訓練過程
- 加快收斂速度
- 提升模型泛化能力

模型組成 — 處理時間序列的流程

1. 輸入序列 + 位置編碼:

- 將時間序列 $[x_1, x_2, \dots, x_t]$ 轉換為向量表示
- 加上位置編碼，保留時間順序資訊

2. 自注意力計算:

- 計算每個時間點與所有其他時間點的關聯性（注意力權重）
- 根據關聯性加權整合資訊

3. 多層堆疊處理:

- 通過多層 Transformer 編碼器
- 每層提取更高層次的時間模式

4. 輸出預測:

- 使用最終的表示向量進行預測

關鍵特性:

- 全局視野：可以同時看到所有時間點，不受距離限制
- 平行處理：所有計算可以同時進行，訓練速度快
- 可解釋性：注意力權重可視化，了解模型關注哪些時間點

建模流程

1. 數據準備與預處理

- 序列構建：構建（輸入窗口，預測目標），與 LSTM 類似
- 標準化與特徵工程：數據縮放 + 加入時間特徵（星期、月份等）

2. 模型架構設計

- 層數與維度：編碼器層數（4-12 層）、隱藏維度（128-512）
- 注意力設計：多頭注意力頭數（4-16 個）、位置編碼方式

3. 訓練與優化

- 損失函數與優化器：MSE/MAE/Huber Loss + Adam（搭配學習率調度）
- 正則化：Dropout、Label Smoothing

4. 預測與評估

- 性能評估：計算 RMSE、MAE、MAPE
- 注意力分析：視覺化注意力權重，理解模型關注哪些時間點
- 基準比較：與 LSTM 比較改進程度

案例說明：預測股票收盤價

- 情境：
 - 使用過去 60 天的收盤價預測未來 5 天
 - 數據: [Day 1 價格, Day 2 價格, ..., Day 60 價格]
- Transformer 建模步驟：
 - 數據準備：
 - 輸入: 過去 60 天的價格序列
 - 輸出: 未來 5 天的價格
 - 加入位置編碼標記時間順序
 - 模型架構：
 - 6 層 Transformer 編碼器
 - 8 個注意力頭
 - 256 維隱藏層
 - 訓練: 使用歷史數據訓練模型
 - 預測: 輸入最近 60 天 → 預測未來 5 天

Transformer 的優勢體現：

- 自動發現「20 天前的重大事件對今天的影響」
- 同時處理所有 60 天的數據，捕捉複雜的時間模式
- 注意力權重顯示模型關注哪些關鍵時間點

優缺點

優點：

- 全局依賴捕捉：可以直接建模任意距離的時間依賴
- 平行化訓練：訓練速度遠快於 LSTM（大規模數據下優勢明顯）
- 可解釋性較好：注意力權重提供預測的可視化解釋
- 可擴展性強：容易擴展到更大的模型和更長的序列

缺點：

- 數據需求更大：通常需要比 LSTM 更多的訓練數據
- 計算資源需求高：參數量大、記憶體佔用高
- 超參數複雜：層數、頭數、維度等需要仔細調整
- 對短序列優勢不明顯：在小規模數據上可能不如 LSTM

應用領域

- 金融市場：
 - 股票價格預測（特別是多步預測）
 - 高頻交易策略
 - 市場異常檢測
- 多變量時間序列：
 - 整合價格、交易量、新聞情緒、宏觀經濟指標
 - 跨市場關聯建模（股市、匯率、大宗商品）
- 長序列預測：
 - 季度/年度經濟指標預測
 - 長期趨勢分析

定位

作為 Baseline 的價值：

- 注意力機制模型的代表，評估專門變體（Informer、Autoformer）改進的基準
- 全局依賴建模能力，直接捕捉任意距離的時間關係
- 平行化訓練優勢，大規模數據下的高效方案
- 標準架構成熟，便於理解核心機制與實作

模型比較

特性	ARIMA	LSTM	Transformer
模型類型	傳統統計	深度學習(循環)	深度學習(注意力)
建模能力	線性	非線性	非線性
長期依賴	弱	中	強
數據需求	少(數百筆)	中(數千筆)	多(數萬筆)
訓練速度	快	中	快(平行化)
可解釋性	強	弱	中

選擇建議

- ARIMA：數據少、需快速建模、需可解釋性
- LSTM：中等數據量、複雜非線性模式
- Transformer：大數據量、長序列、需高準確度

Week8 程式作業 程式碼說明

比較四類技術指標（趨勢型、震盪型、波動型、量能型）在實驗中的差異

[參考程式碼連結](#)