**AI辭典**

[第 1 章｜基礎概念 8](#_Toc207716048)

[Artificial Intelligence (AI)人工智慧 8](#_Toc207716049)

[Machine Learning (ML)機器學習 8](#_Toc207716050)

[Deep Learning (DL)深度學習 8](#_Toc207716051)

[Neural Network (NN)神經網路 9](#_Toc207716052)

[Natural Language Processing (NLP)自然語言處理 9](#_Toc207716053)

[Computer Vision (CV)電腦視覺 9](#_Toc207716054)

[Reinforcement Learning (RL)強化學習 10](#_Toc207716055)

[Supervised Learning監督式學習 10](#_Toc207716056)

[Unsupervised Learning非監督式學習 10](#_Toc207716057)

[Semi-Supervised Learning半監督式學習 11](#_Toc207716058)

[Generative AI 生成式AI 11](#_Toc207716059)

[Artificial General Intelligence (AGI)通用人工智慧 11](#_Toc207716060)

[第 2 章 常見模型與架構 12](#_Toc207716061)

[Transformer轉換器模型 12](#_Toc207716062)

[Encoder-Decoder編碼器解碼器架構 12](#_Toc207716063)

[Self-Attention自注意力機制 12](#_Toc207716064)

[RNN (Recurrent Neural Network)循環神經網路 13](#_Toc207716065)

[LSTM (Long Short-Term Memory)長短期記憶網路 13](#_Toc207716066)

[GRU (Gated Recurrent Unit)閘控循環單元 13](#_Toc207716067)

[CNN (Convolutional Neural Network)卷積神經網路 13](#_Toc207716068)

[ResNet殘差網路 14](#_Toc207716069)

[GAN (Generative Adversarial Network) 生成對抗網路 14](#_Toc207716070)

[StyleGAN風格生成網路 14](#_Toc207716071)

[Diffusion Models擴散模型 14](#_Toc207716072)

[Autoencoder自編碼器 15](#_Toc207716073)

[VAE (Variational Autoencoder)變分自編碼器 15](#_Toc207716074)

[BERT雙向編碼表示轉換器 15](#_Toc207716075)

[GPT (Generative Pre-trained Transformer) 預訓練生成轉換器 15](#_Toc207716076)

[T5 (Text-to-Text Transfer Transformer) 16](#_Toc207716077)

[RoBERTa強化版 BERT 16](#_Toc207716078)

[DistilBERT精簡版 BERT 16](#_Toc207716079)

[LLaMAMeta 大型語言模型 17](#_Toc207716080)

[Whisper語音識別模型 17](#_Toc207716081)

[第 3 章 訓練與推理技術（20 條） 18](#_Toc207716082)

[Backpropagation反向傳播 18](#_Toc207716083)

[Gradient Descent梯度下降 18](#_Toc207716084)

[Stochastic Gradient Descent (SGD)隨機梯度下降 18](#_Toc207716085)

[Adam OptimizerAdam 優化器 19](#_Toc207716086)

[Loss Function損失函數 19](#_Toc207716087)

[Cross-Entropy Loss交叉熵損失 19](#_Toc207716088)

[Mean Squared Error (MSE)均方誤差 20](#_Toc207716089)

[Overfitting過擬合 20](#_Toc207716090)

[Underfitting欠擬合 20](#_Toc207716091)

[Regularization正則化 20](#_Toc207716092)

[Dropout隨機失活 21](#_Toc207716093)

[Batch Normalization批次正規化 21](#_Toc207716094)

[Early Stopping提前停止 21](#_Toc207716095)

[Epoch訓練週期 21](#_Toc207716096)

[Batch Size批次大小 22](#_Toc207716097)

[Fine-Tuning微調 22](#_Toc207716098)

[Transfer Learning遷移學習 22](#_Toc207716099)

[Pre-training預訓練 22](#_Toc207716100)

[Zero-shot Learning零樣本學習 23](#_Toc207716101)

[Few-shot Learning小樣本學習 23](#_Toc207716102)

[第 4 章 大型語言模型 (LLM) 與工具 24](#_Toc207716103)

[Large Language Model (LLM)大型語言模型 24](#_Toc207716104)

[Token標記 24](#_Toc207716105)

[Embedding向量嵌入 24](#_Toc207716106)

[Context Window上下文視窗 25](#_Toc207716107)

[Prompt Engineering提示工程 25](#_Toc207716108)

[Chain of Thought (CoT)思維鏈 25](#_Toc207716109)

[RAG (Retrieval-Augmented Generation)– 檢索增強生成 26](#_Toc207716110)

[Instruction Tuning指令微調 26](#_Toc207716111)

[RLHF (Reinforcement Learning from Human Feedback)人類回饋強化學習 26](#_Toc207716112)

[Tool Use /Function Calling工具使用 / 函數調用 26](#_Toc207716113)

[Multi-Modal AI多模態 AI 27](#_Toc207716114)

[Knowledge Graph知識圖譜 27](#_Toc207716115)

[Vector Database向量資料庫 27](#_Toc207716116)

[Pinecone向量資料庫平台 28](#_Toc207716117)

[FAISSFacebook AI Similarity Search 28](#_Toc207716118)

[第 5 章 AI 應用領域 29](#_Toc207716119)

[Chatbot聊天機器人 29](#_Toc207716120)

[Virtual Assistant虛擬助理 29](#_Toc207716121)

[Recommendation System推薦系統 29](#_Toc207716122)

[Predictive Analytics預測分析 30](#_Toc207716123)

[Fraud Detection詐欺檢測 30](#_Toc207716124)

[Sentiment Analysis情感分析 30](#_Toc207716125)

[Speech Recognition語音識別 30](#_Toc207716126)

[Text-to-Speech (TTS)語音合成 31](#_Toc207716127)

[OCR(Optical Character Recognition)–光學文字識別 31](#_Toc207716128)

[Autonomous Vehicle自駕車 31](#_Toc207716129)

[Robotics機器人 31](#_Toc207716130)

[Digital Twin數位孿生 32](#_Toc207716131)

[Smart Manufacturing智慧製造 32](#_Toc207716132)

[Predictive Maintenance預測性維護 32](#_Toc207716133)

[Edge AI邊緣人工智慧 32](#_Toc207716134)

[AI in Healthcare醫療中的 AI 33](#_Toc207716135)

[AI in Finance金融中的 AI 33](#_Toc207716136)

[AI in Education教育中的 AI 33](#_Toc207716137)

[AI in Retail零售中的 AI 33](#_Toc207716138)

[Deepfake深偽技術 34](#_Toc207716139)

[第 6 章 資料處理與數據技術 35](#_Toc207716140)

[Dataset資料集 35](#_Toc207716141)

[Data Labeling資料標註 35](#_Toc207716142)

[Data Augmentation資料增強 35](#_Toc207716143)

[Data Cleaning資料清理 36](#_Toc207716144)

[Feature Engineering特徵工程 36](#_Toc207716145)

[Feature Selection特徵選擇 36](#_Toc207716146)

[Dimensionality Reduction降維 36](#_Toc207716147)

[PCA (Principal Component Analysis)主成分分析 37](#_Toc207716148)

[t-SNE (t-distributed Stochastic Neighbor Embedding) 37](#_Toc207716149)

[UMAP (Uniform Manifold Approximation and Projection) 37](#_Toc207716150)

[Data Pipeline數據管線 37](#_Toc207716151)

[ETL (Extract, Transform, Load) 38](#_Toc207716152)

[Big Data大數據 38](#_Toc207716153)

[Data Warehouse數據倉儲 38](#_Toc207716154)

[Data Lake數據湖 38](#_Toc207716155)

[第 7 章 數學與理論基礎 40](#_Toc207716156)

[Linear Algebra線性代數 40](#_Toc207716157)

[Vector向量 40](#_Toc207716158)

[Matrix矩陣 40](#_Toc207716159)

[Eigenvalue & Eigenvector特徵值與特徵向量 41](#_Toc207716160)

[Probability機率 41](#_Toc207716161)

[Bayes’ Theorem貝葉斯定理 41](#_Toc207716162)

[Conditional Probability條件機率 41](#_Toc207716163)

[Random Variable隨機變數 42](#_Toc207716164)

[Expectation & Variance期望與變異數 42](#_Toc207716165)

[Entropy熵 42](#_Toc207716166)

[Information Gain資訊增益 43](#_Toc207716167)

[Gradient梯度 43](#_Toc207716168)

[Derivative微分 43](#_Toc207716169)

[Convex Function凸函數 43](#_Toc207716170)

[Optimization最佳化 44](#_Toc207716171)

[第 8 章 AI 工具與平台 45](#_Toc207716172)

[TensorFlowGoogle 開源深度學習框架 45](#_Toc207716173)

[PyTorchFacebook 開源深度學習框架 45](#_Toc207716174)

[Keras高階神經網路 API 45](#_Toc207716175)

[JAX高效數值計算與自動微分庫 46](#_Toc207716176)

[Scikit-learn機器學習工具庫 46](#_Toc207716177)

[Hugging Face TransformersNLP 模型庫 46](#_Toc207716178)

[LangChainLLM 應用框架 47](#_Toc207716179)

[OpenAI APIOpenAI 應用程式介面 47](#_Toc207716180)

[Google Cloud AIGoogle 雲端 AI 服務 47](#_Toc207716181)

[AWS AI Services亞馬遜雲端 AI 服務 47](#_Toc207716182)

[Microsoft Azure AI微軟雲端 AI 48](#_Toc207716183)

[IBM WatsonIBM AI 平台 48](#_Toc207716184)

[NVIDIA CUDA平行運算平台 48](#_Toc207716185)

[NVIDIA TensorRT推理加速引擎 48](#_Toc207716186)

[ONNX開放神經網路交換格式 49](#_Toc207716187)

[MLflow機器學習實驗管理平台 49](#_Toc207716188)

[Weights & Biases (W&B)機器學習實驗追蹤 49](#_Toc207716189)

[Ray分散式運算框架 49](#_Toc207716190)

[Apache Spark MLlib大數據機器學習庫 50](#_Toc207716191)

[RapidMiner機器學習平台 50](#_Toc207716192)

# 第 1 章｜基礎概念

## Artificial Intelligence (AI)人工智慧

人工智慧（AI）是讓電腦或機器模擬人類智慧行為的科學與技術，涵蓋感知、推理、學習與決策等能力。AI 可分為針對單一任務的窄域 AI（如語音助理、垃圾信過濾），以及理想上能跨領域解題的廣域/通用 AI。在日常生活中，AI 已廣泛應用於醫療影像判讀、金融風控、語音助理、物流配送與自駕車導航等。其快速發展也帶來隱私、偏見、公平性與就業結構轉變等議題，成為科技與社會共同關注的焦點。

[**📺 推薦 YouTube：What is Artificial Intelligence?**](https://www.youtube.com/@crashcourse)

**🌐** [**建議網站：IBM AI Explained（概覽與教學資源）**](https://www.ibm.com/artificial-intelligence)

## Machine Learning (ML)機器學習

機器學習是 AI 的核心分支，強調讓模型從資料中「學習規律」而非以人工規則硬寫。典型任務包含分類、迴歸與分群等，常見方法有監督式、非監督式與強化學習。應用情境包括信用卡詐欺偵測、影像辨識、商品推薦、需求預測與客服自動化等。在台灣大學部學習時，建議從線性模型、決策樹、SVM 入門，再銜接深度學習與大型語言模型。

**📺** [**推薦 YouTube：李宏毅｜機器學習課程（NTU）**](https://www.youtube.com/@HungyiLeeNTU)

**🌐** [**建議網站：Google ML Crash Course（互動式教材）**](https://developers.google.com/machine-learning/crash-course)

## Deep Learning (DL)深度學習

深度學習是以多層神經網路為核心的機器學習方法，能自動從大量資料中學習高階特徵表示。它在電腦視覺、語音識別、自然語言處理與多模態生成等任務表現突出。代表性成果包括影像辨識突破、AlphaGo/AlphaZero 系列、以及各式大型語言與擴散模型。學習建議：熟悉張量運算、反向傳播、常見網路架構（CNN/RNN/Transformer）與訓練技巧（正則化、初始化、優化器）。

**📺** [**推薦 YouTube：Deep Learning 概念影片（IBM Technology）**](https://www.youtube.com/@IBMTechnology)

**🌐** [**建議網站：DeepLearning.AI（學習專欄與課程）**](https://www.deeplearning.ai/)

## Neural Network (NN)神經網路

神經網路模仿人腦神經元連結的數學模型，由多層節點與非線性激活函數組成，能近似複雜函數。前饋式網路常見於表格與影像任務；循環式與注意力機制網路擅長處理序列資料。訓練透過反向傳播與梯度下降更新權重，並以驗證集監控過擬合問題。理解過程可參考可視化教學資源，從單層感知器逐步推進至多層與殘差、注意力架構。

📺 [推薦 YouTube：Neural Networks（3Blue1Brown 可視化系列）](https://www.youtube.com/@3blue1brown)

🌐 [建議網站：MIT OpenCourseWare：Neural Networks 相關課程](https://ocw.mit.edu/)

## Natural Language Processing (NLP)自然語言處理

NLP 讓機器能理解、生成與操作人類語言，涵蓋分詞、詞性標註、語法/語義分析、對話管理與文本生成等。深度學習與 Transformer 的引入，使機器翻譯、問答、摘要與對話系統大幅進步。實務應用含客服自動化、社群輿情分析、法務與醫療文本處理、跨語言搜尋等。入門可結合語料蒐集、標註與評估指標（如 BLEU、ROUGE），再進一步學習預訓練語言模型。

📺 [推薦 YouTube：NLP 入門（IBM Technology / 初學友善）](https://www.youtube.com/@IBMTechnology)

🌐 [建議網站：Stanford CS224N（NLP 旗艦課程）](https://web.stanford.edu/class/cs224n/)

## Computer Vision (CV)電腦視覺

電腦視覺致力於讓機器理解影像與影片，常見任務有分類、檢測、分割、追蹤與姿態估計。深度學習的 CNN/Transformer 使 CV 在醫療影像、智慧製造、安防與自駕車感知上表現卓越。資料品質（標註一致性、增強策略）與部署需求（延遲、記憶體）會影響方案選型。學習上可從影像卷積與資料增強開始，進一步理解檢測（如 YOLO）與分割（如 U-Net）等框架。

📺 [推薦 YouTube：What is Computer Vision?（IBM Technology）](https://www.youtube.com/@IBMTechnology)

🌐 [建議網站：Stanford CS231n（Convolutional Neural Networks for Visual Recognition）](http://cs231n.stanford.edu/)

## Reinforcement Learning (RL)強化學習

強化學習讓智能體（agent）在環境中互動，以獎勵為依據學習最佳策略，核心元素包含狀態、動作、回饋與策略。典型演算法有 Q-learning、Policy Gradient、Actor-Critic 及其深度變體（DQN、PPO 等）。應用涵蓋遊戲對戰、機器人控制、資源排程與推薦系統策略優化。入門可透過 OpenAI Gym 之類的模擬環境實作，理解探索/利用、折扣回饋與穩定訓練技巧。

📺 [推薦 YouTube：DeepMind / UCL RL Lectures（入門到進階）](https://www.youtube.com/@DeepMind)

🌐 [建議網站：Spinning Up in Deep RL（OpenAI 教程）](https://spinningup.openai.com/)

## Supervised Learning監督式學習

監督式學習使用帶標註的資料訓練模型，學會從輸入預測目標（標籤），常見於分類與迴歸。關鍵在於資料標註品質、訓練/驗證切分與適當的評估指標（如 Accuracy、F1、RMSE）。典型方法含線性/邏輯迴歸、決策樹、隨機森林、梯度提升、SVM 與神經網路。實務上需避免過擬合，並透過交叉驗證、正則化與早停等技巧提升泛化能力。

📺 [推薦 YouTube：Supervised vs Unsupervised Learning（StatQuest）](https://www.youtube.com/@statquest)

🌐 [建議網站：scikit-learn：Supervised learning 指南](https://scikit-learn.org/)

## Unsupervised Learning非監督式學習

非監督式學習在無標註資料中尋找結構與模式，常見任務包含分群（K-Means、DBSCAN）與降維（PCA、t-SNE、UMAP）。它可用於顧客分群、異常偵測、資料可視化與特徵學習等場景。挑戰在於評估困難（缺乏標籤）、結果解釋性與對超參數敏感。實務上常作為前處理或探索性分析的第一步，協助後續監督式建模。

📺 [推薦 YouTube：Unsupervised Learning（StatQuest）](https://www.youtube.com/@statquest)

🌐 [建議網站：scikit-learn：Clustering / Dimensionality reduction](https://scikit-learn.org/)

## Semi-Supervised Learning半監督式學習

半監督式學習結合少量標註資料與大量未標註資料進行訓練，降低昂貴標註成本。典型策略包含一致性正則化（consistency regularization）、偽標註（pseudo-labeling）與自訓練。在醫療影像、語音與工業瑕疵檢測等標註昂貴領域特別實用。關鍵在於控制噪聲標籤與資料分佈移轉，確保模型不被錯誤訊號牽引。

📺 [推薦 YouTube：Semi-Supervised Learning（入門概念）](https://www.youtube.com/@IBMTechnology)

🌐 [建議網站：Papers with Code：Semi-Supervised Learning 資源匯總](https://paperswithcode.com/task/semi-supervised-learning)

## Generative AI 生成式AI

生成式 AI 能從資料中學習分佈並合成新內容，涵蓋文字（LLM）、影像（GAN、擴散模型）、音樂與多模態生成。代表性應用包含內容創作輔助、行銷素材產生、原型設計、資料擴增與教育輔助。挑戰包括版權、深偽內容、偏見與安全性，需配合水印、檢測器與治理政策。學習重點：理解機率生成、對抗訓練與擴散過程，以及提示工程與檢索增強生成（RAG）。

📺 [推薦 YouTube：What is Generative AI?（IBM Technology）](https://www.youtube.com/@IBMTechnology)

🌐 [建議網站：DeepLearning.AI：Generative AI 專題](https://www.deeplearning.ai/short-courses/)

## Artificial General Intelligence (AGI)通用人工智慧

AGI 指具備接近人類的通用理解與推理能力，能跨領域學習、遷移與自我改進。目前尚未實現 AGI，但大型模型、多模態學習、長上下文推理與工具使用等方向被視為可能路徑。討論焦點包含對社會、經濟與安全的長期影響，以及如何實施對齊（alignment）與治理（governance）。作為學生，理解 AGI 概念有助於從技術、倫理與政策層面全面思考 AI 的未來。

📺 [推薦 YouTube：What is AGI?（ColdFusion / 解說科技史與趨勢）](https://www.youtube.com/@ColdFusion)

🌐 [建議網站：センター資料 / AI Governance（綜覽型資源，含政策與安全）](<https://ai.ethics>、<https://oecd.ai>)

# 第 2 章 常見模型與架構

## Transformer轉換器模型

Transformer 是深度學習的一種重要架構，由 Google 在 2017 年提出。其核心是自注意力機制（Self-Attention），能在處理序列資料時捕捉長距依賴關係，而不需像 RNN 一樣逐步計算，訓練速度大幅提升。Transformer 是現今大型語言模型（如 GPT、BERT）的基礎。它也廣泛應用於機器翻譯、文本摘要、語音處理與電腦視覺，幾乎成為 AI 研究與應用的主流架構。

📺 [推薦 YouTube：Attention is All You Need 解說](https://www.youtube.com/watch?v=ugWDIIOHtPA)

🌐 [建議網站：The Illustrated Transformer](http://jalammar.github.io/illustrated-transformer/)

## Encoder-Decoder編碼器解碼器架構

Encoder-Decoder 是一種神經網路架構，將輸入序列壓縮成隱藏表示（Encoder），再由解碼器（Decoder）輸出目標序列。它被廣泛應用於機器翻譯、語音識別、文本摘要等任務。Transformer、Seq2Seq 模型都屬於 Encoder-Decoder 結構。這種架構的關鍵挑戰是如何在壓縮與解壓過程中保留足夠的語意資訊，因此注意力機制成為後來改進的重要元素。

📺 [推薦 YouTube：Seq2Seq and Attention（Stanford）](https://www.youtube.com/watch?v=XXtpJxZBa2c)

🌐 [建議網站：Stanford CS224N：Seq2Seq Notes](https://web.stanford.edu/class/cs224n/)

## Self-Attention自注意力機制

Self-Attention 是 Transformer 的核心技術，它允許模型在處理序列時，動態調整對不同位置資訊的關注程度。這解決了 RNN 訓練時難以捕捉長距離依賴的問題。在自然語言處理中，Self-Attention 可讓模型在翻譯或生成句子時同時考量前後文語意。

📺 [推薦 YouTube：Self-Attention Mechanism 解釋](https://www.youtube.com/watch?v=OyFJWRnt_AY)

🌐 [建議網站：The Illustrated Self-Attention](http://jalammar.github.io/illustrated-transformer/)

## RNN (Recurrent Neural Network)循環神經網路

RNN 是處理序列資料的神經網路，透過隱藏狀態將前一步資訊傳遞到下一步。它能建模時間序列與語言數據中的上下文關係。缺點是難以捕捉長距依賴，且容易出現梯度消失或爆炸問題。常見應用包括語音識別、語言模型與股票價格預測。

📺 [推薦 YouTube：RNN Explained（Welch Labs）](https://www.youtube.com/watch?v=WCUNPb-5EYI)

🌐 [建議網站：Colah’s Blog: Understanding LSTMs](https://colah.github.io/posts/2015-08-Understanding-LSTMs/)

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

LSTM 是 RNN 的改進版本，設計了『記憶單元』與『閘門機制』，能更好地處理長距依賴。它能選擇性地保留或忘記資訊，避免梯度消失問題。常應用於語音辨識、機器翻譯與時間序列預測。

📺 [推薦 YouTube：LSTM Explained（StatQuest）](https://www.youtube.com/watch?v=8HyCNIVRbSU)

🌐 [建議網站：Colah’s Blog: Understanding LSTMs](https://colah.github.io/posts/2015-08-Understanding-LSTMs/)

## GRU (Gated Recurrent Unit)閘控循環單元

GRU 是 LSTM 的簡化版本，僅保留更新閘與重置閘。它參數更少、訓練速度更快，但能保有類似的性能。GRU 常應用於需要快速訓練與即時應用的場景。

📺 [推薦 YouTube：GRU vs LSTM（DeepLearning.TV）](https://www.youtube.com/watch?v=8HyCNIVRbSU)

🌐 [建議網站：Understanding GRUs](https://towardsdatascience.com/understanding-gru-networks-2ef37df6c9be)

## CNN (Convolutional Neural Network)卷積神經網路

CNN 是一種專門用於處理網格結構數據（如影像）的神經網路。它透過卷積層自動提取局部特徵，並使用池化層壓縮訊息，最後透過全連接層進行分類。CNN 在影像分類、物件檢測、醫療影像分析上有極大突破。

📺 [推薦 YouTube：Convolutional Neural Networks（3Blue1Brown）](https://www.youtube.com/watch?v=iaSUYvmCekI)

🌐 [建議網站：Stanford CS231n](http://cs231n.stanford.edu/)

## ResNet殘差網路

ResNet 是一種深層 CNN 架構，透過引入『殘差連接（skip connection）』解決梯度消失問題。它使得神經網路可以非常深（超過 100 層）而不會退化。ResNet 被廣泛應用於影像分類、檢測與醫學影像分析等領域。

📺 [推薦 YouTube：ResNet Explained](https://www.youtube.com/watch?v=GWt6Fu05voI)

🌐 [建議網站：Deep Residual Learning (原始論文)](https://arxiv.org/abs/1512.03385)

## GAN (Generative Adversarial Network) 生成對抗網路

GAN 由生成器（Generator）與判別器（Discriminator）組成，透過對抗訓練產生逼真的資料。它可用於圖像生成、影像修復、超解析度與藝術創作。但也帶來深偽影像（deepfake）等爭議。

📺 [推薦 YouTube：GANs in 5 Minutes](https://www.youtube.com/watch?v=8L11aMN5KY8)

🌐 [建議網站：GANs Paper (Goodfellow, 2014)](https://arxiv.org/abs/1406.2661)

## StyleGAN風格生成網路

StyleGAN 是 GAN 的進階版本，由 NVIDIA 提出，特別適合生成高品質人臉圖像。它能控制生成圖像的風格（如髮型、膚色、姿態），應用於遊戲、設計與娛樂。

📺 [推薦 YouTube：StyleGAN Explained](https://www.youtube.com/watch?v=kSLJriaOumA)

🌐 [建議網站：NVIDIA StyleGAN](https://github.com/NVlabs/stylegan)

## Diffusion Models擴散模型

擴散模型是一類生成模型，透過逐步將隨機噪聲轉換為結構化數據來生成影像或聲音。它已成為生成圖像的主流方法，例如 Stable Diffusion 與 DALL·E 2。相較於 GAN，擴散模型更穩定，能生成更多樣化與高品質的影像。

📺 [推薦 YouTube：Diffusion Models Explained](https://www.youtube.com/watch?v=HoKDTa5jHvg)

🌐 [建議網站：Stable Diffusion (官方)](https://stability.ai/)

## Autoencoder自編碼器

自編碼器是一種無監督學習模型，透過壓縮（編碼器）與重建（解碼器）來學習數據特徵。它常用於降維、異常檢測與生成模型的基礎。

📺 [推薦 YouTube：Autoencoders Explained](https://www.youtube.com/watch?v=4XT23X0Fjfk)

🌐 [建議網站：DeepLearning.ai Autoencoder Tutorial](https://www.deeplearning.ai/)

## VAE (Variational Autoencoder)變分自編碼器

VAE 是自編碼器的一種改良版，能進行生成任務。它在隱變量空間中加入機率分布，使模型能生成更多樣化的新數據。VAE 廣泛應用於影像生成、資料增強與醫學影像合成。

📺 [推薦 YouTube：VAE Explained](https://www.youtube.com/watch?v=9zKuYvjFFS8)

🌐 [建議網站：Auto-Encoding Variational Bayes (原始論文)](https://arxiv.org/abs/1312.6114)

## BERT雙向編碼表示轉換器

BERT 是 Google 在 2018 年提出的語言模型，能同時考慮上下文（雙向）來理解語意。它在問答、分類、命名實體識別等任務上大幅提升效果。BERT 成為後續多種 NLP 模型（如 RoBERTa、DistilBERT）的基礎。

📺 [推薦 YouTube：BERT Explained](https://www.youtube.com/watch?v=xI0HHN5XKDo)

🌐 [建議網站：BERT 原始論文](https://arxiv.org/abs/1810.04805)

## GPT (Generative Pre-trained Transformer) 預訓練生成轉換器

GPT 是 OpenAI 開發的生成式語言模型，基於 Transformer 架構，透過大規模文本預訓練後進行下游任務微調。它能生成流暢、連貫的文字，並能進行問答、翻譯與總結。最新版本（GPT-4、GPT-5）已支持多模態輸入，並展現出更強的推理與對話能力。

📺 [推薦 YouTube：How GPT Works](https://www.youtube.com/watch?v=esRzPqzQGqY)

🌐 [建議網站：OpenAI GPT Paper](https://openai.com/research)

## T5 (Text-to-Text Transfer Transformer)

T5 是 Google 提出的模型，將所有 NLP 任務轉換為『文本到文本』的形式。例如翻譯、摘要、問答都能被統一表達為輸入文本到輸出文本的轉換。這使得模型結構更簡單，適合多任務學習。

📺 [推薦 YouTube：T5 Explained](https://www.youtube.com/watch?v=uf_9nrsl0l8)

🌐 [建議網站：Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer](https://arxiv.org/abs/1910.10683)

## RoBERTa強化版 BERT

RoBERTa 是 Facebook 提出的 BERT 改進版，透過更多數據、更長時間訓練與移除下一句預測任務來提升性能。它在多個 NLP 任務上超越了原始 BERT。

📺 [推薦 YouTube：RoBERTa Explained](https://www.youtube.com/watch?v=z3F3c3hJdZs)

🌐 [建議網站：RoBERTa 論文](https://arxiv.org/abs/1907.11692)

## DistilBERT精簡版 BERT

DistilBERT 是 Hugging Face 提出的模型，透過知識蒸餾將 BERT 縮小一半，保留大部分性能。它速度更快、資源需求更低，適合移動端與即時應用。

📺 [推薦 YouTube：DistilBERT Explained](https://www.youtube.com/watch?v=2lmvdoUG-Lg)

🌐 [建議網站：DistilBERT 論文](https://arxiv.org/abs/1910.01108)

## LLaMAMeta 大型語言模型

LLaMA 是 Meta 在 2023 年提出的開源大型語言模型系列。它以相對較小的規模展現高性能，為研究社群提供便利。LLaMA 的出現推動了開源社群的快速發展，催生許多衍生模型。

📺 [推薦 YouTube：LLaMA Explained](https://www.youtube.com/watch?v=FQHK8hN3m8Y)

🌐 [建議網站：Meta AI Research](https://ai.meta.com/)

## Whisper語音識別模型

Whisper 是 OpenAI 開發的多語言語音識別模型。它能將語音轉換為文字，並支援翻譯功能。Whisper 對多種口音與背景噪音有很好的魯棒性，被廣泛應用於字幕生成、語音助手與無障礙科技。

📺 [推薦 YouTube：Whisper ASR Demo](https://www.youtube.com/watch?v=j3Q0DUJG8X4)

🌐 [建議網站：OpenAI Whisper](https://openai.com/research/whisper)

# 第 3 章 訓練與推理技術（20 條）

## Backpropagation反向傳播

反向傳播（Backpropagation）是訓練神經網路的核心演算法。它透過鏈式法則計算損失函數對每個權重的偏導數，並將誤差由輸出層往前傳遞，更新網路中的參數。這種方法大幅提升了深度神經網路的可訓練性，使得模型能有效學習複雜模式。在實務應用中，反向傳播配合梯度下降與優化器（如 Adam）能快速收斂，成為深度學習不可或缺的基礎。

📺 [推薦 YouTube：Backpropagation Explained（3Blue1Brown）](https://www.youtube.com/watch?v=tIeHLnjs5U8)

🌐 [建議網站：CS231n: Backpropagation](http://cs231n.stanford.edu/)

## Gradient Descent梯度下降

梯度下降（Gradient Descent）是一種優化演算法，用來最小化損失函數。透過計算損失對模型參數的梯度，並沿著梯度反方向更新參數，模型逐步逼近最佳解。梯度下降分為批量（Batch）、隨機（SGD）與小批量（Mini-Batch）等形式。這一方法在訓練深度學習模型中被廣泛使用，是所有優化技術的基礎。

📺 [推薦 YouTube：Gradient Descent Explained（StatQuest）](https://www.youtube.com/watch?v=sDv4f4s2SB8)

🌐 [建議網站：CS229: Machine Learning (Stanford)](http://cs229.stanford.edu/)

## Stochastic Gradient Descent (SGD)隨機梯度下降

SGD 是梯度下降的一種變體，每次僅用一筆或小批量數據來近似整體梯度，更新模型參數。這使得訓練效率更高，且在處理大規模數據集時特別有效。然而 SGD 更新方向有隨機性，可能導致收斂速度不穩定，因此常搭配動量（Momentum）或學習率調整。SGD 是深度學習中最常用的優化方法之一。

📺 [推薦 YouTube：Stochastic Gradient Descent Explained](https://www.youtube.com/watch?v=vMh0zPT0tLI)

🌐 [建議網站：Optimization for Machine Learning (Book)](https://optimizationbook.com/)

## Adam OptimizerAdam 優化器

Adam（Adaptive Moment Estimation）是最常用的優化器之一，結合了動量（Momentum）與自適應學習率方法（如 RMSprop）。它會同時考慮一階動量（平均梯度）與二階動量（梯度平方平均），使得每個參數都有獨立的學習率。Adam 在深度學習中具有收斂快、穩定性高的特點，被廣泛應用於 NLP、CV 與推薦系統等領域。

📺 [推薦 YouTube：Adam Optimizer Explained（StatQuest）](https://www.youtube.com/watch?v=JXQT_vxqwIs)

🌐 [建議網站：Adam: A Method for Stochastic Optimization (原始論文)](https://arxiv.org/abs/1412.6980)

## Loss Function損失函數

損失函數用來衡量模型預測與實際標籤之間的差距，是模型訓練的核心指標。常見損失函數包括迴歸問題的均方誤差（MSE）、分類問題的交叉熵損失（Cross-Entropy）。設計合適的損失函數能引導模型學習到更有意義的表示與決策邊界。

📺 [推薦 YouTube：Loss Functions Explained](https://www.youtube.com/watch?v=0LhiS6yu2qQ)

🌐 [建議網站：CS231n: Loss Functions](http://cs231n.stanford.edu/)

## Cross-Entropy Loss交叉熵損失

交叉熵損失常用於分類問題，特別是多分類任務。它衡量預測的機率分佈與真實分佈之間的差異。交叉熵在神經網路分類中是最常見的選擇，與 softmax 輸出層搭配效果良好。

📺 [推薦 YouTube：Cross Entropy Loss Explained](https://www.youtube.com/watch?v=ErfnhcEV1O8)

🌐 [建議網站：Cross Entropy Loss Tutorial](https://ml-cheatsheet.readthedocs.io/en/latest/loss_functions.html)

## Mean Squared Error (MSE)均方誤差

MSE 是迴歸問題中最常用的損失函數，計算預測值與真實值之差的平方平均。它對離群值敏感，因為誤差平方會放大偏差。MSE 的數學簡單，便於梯度計算，因此被廣泛應用。

📺 [推薦 YouTube：MSE Explained](https://www.youtube.com/watch?v=QDW0MWTxR6I)

🌐 [建議網站：MSE Definition (Wikipedia)](https://en.wikipedia.org/wiki/Mean_squared_error)

## Overfitting過擬合

過擬合指模型在訓練集上表現良好，但在測試集上表現差。這通常發生在模型過於複雜、參數過多或訓練次數過長時。解決方法包括正則化、Dropout、資料增強與交叉驗證。

📺 [推薦 YouTube：Overfitting Explained](https://www.youtube.com/watch?v=0moyMq45c0c)

🌐 [建議網站：Avoiding Overfitting (scikit-learn)](https://scikit-learn.org/stable/overfitting.html)

## Underfitting欠擬合

欠擬合指模型過於簡單，無法捕捉數據中的模式。表現為訓練誤差與測試誤差都很高。解決方法包括使用更複雜的模型、增加特徵或延長訓練時間。

📺 [推薦 YouTube：Underfitting Explained](https://www.youtube.com/watch?v=0moyMq45c0c)

🌐 [建議網站：Bias-Variance Tradeoff](https://en.wikipedia.org/wiki/Bias%E2%80%93variance_tradeoff)

## Regularization正則化

正則化是一種防止過擬合的技術，透過在損失函數中加入懲罰項限制模型複雜度。常見方法包括 L1（Lasso）、L2（Ridge）與 Elastic Net。正則化能提升模型的泛化能力。

📺 [推薦 YouTube：Regularization Explained](https://www.youtube.com/watch?v=Q81RR3yKn30)

🌐 [建議網站：Regularization in ML](https://en.wikipedia.org/wiki/Regularization_(mathematics))

## Dropout隨機失活

Dropout 是一種正則化方法，訓練時隨機丟棄部分神經元，以減少過擬合。這使模型不會過度依賴某些特徵，提升泛化性。在深度神經網路中特別有效。

📺 [推薦 YouTube：Dropout Explained](https://www.youtube.com/watch?v=ARq74QuavAo)

🌐 [建議網站：Dropout: A Simple Way to Prevent Neural Networks from Overfitting](https://jmlr.org/papers/volume15/srivastava14a/srivastava14a.pdf)

## Batch Normalization批次正規化

BatchNorm 在訓練時對每一層輸入做標準化，讓資料分佈更穩定。它能加速收斂、減少梯度消失並提高泛化能力。BatchNorm 幾乎成為深度學習中的標準技術。

📺 [推薦 YouTube：Batch Normalization Explained](https://www.youtube.com/watch?v=dXB-KQYkzNU)

🌐 [建議網站：Batch Normalization (Wikipedia)](https://en.wikipedia.org/wiki/Batch_normalization)

## Early Stopping提前停止

Early Stopping 是防止過擬合的一種方法，當驗證集誤差不再下降時就停止訓練。這能避免模型在訓練集表現好但測試集表現差。

📺 [推薦 YouTube：Early Stopping Explained](https://www.youtube.com/watch?v=ZVFeW798-2I)

🌐 [建議網站：Early Stopping (Keras)](https://keras.io/api/callbacks/early_stopping/)

## Epoch訓練週期

Epoch 表示模型完整看過一次訓練數據的過程。深度學習通常需要多個 epoch 才能收斂。過少會欠擬合，過多可能過擬合。

📺 [推薦 YouTube：Epochs, Batches and Iterations Explained](https://www.youtube.com/watch?v=7YV0EtkWyno)

🌐 [建議網站：Epoch (ML Glossary)](https://developers.google.com/machine-learning/glossary)

## Batch Size批次大小

Batch Size 是一次用來更新模型參數的數據量。小批次能提升泛化性，但訓練較慢；大批次訓練快，但可能過擬合。

📺 [推薦 YouTube：Batch Size Explained](https://www.youtube.com/watch?v=mDy1nZ6aKGo)

🌐 [建議網站：Mini-batch Gradient Descent](https://en.wikipedia.org/wiki/Stochastic_gradient_descent)

## Fine-Tuning微調

Fine-Tuning 指在預訓練模型基礎上，使用特定任務數據進一步訓練。這能大幅減少數據需求與計算成本，同時獲得優秀的性能。Fine-Tuning 已成為 NLP 與 CV 任務的常見做法。

📺 [推薦 YouTube：Fine-Tuning Explained](https://www.youtube.com/watch?v=d2XB5-tuCWU)

🌐 [建議網站：Fine-Tuning Pretrained Models (Hugging Face)](https://huggingface.co/docs/transformers/training)

## Transfer Learning遷移學習

遷移學習指將一個任務中學到的知識應用到另一個相關任務。這在資料不足的場景特別有效，例如醫療影像診斷。

📺 [推薦 YouTube：Transfer Learning Explained](https://www.youtube.com/watch?v=AlXlxDnJ2SE)

🌐 [建議網站：Transfer Learning (Wikipedia)](https://en.wikipedia.org/wiki/Transfer_learning)

## Pre-training預訓練

預訓練指先用大規模資料訓練模型，再針對下游任務進行微調。這是目前 NLP 與 CV 主流的方法，能顯著提升模型性能。

📺 [推薦 YouTube：Pretraining and Fine-Tuning](https://www.youtube.com/watch?v=WK6m0E8FiS8)

🌐 [建議網站：Pretraining in NLP (Stanford)](https://web.stanford.edu/class/cs224n/)

## Zero-shot Learning零樣本學習

零樣本學習指模型在未見過的任務或類別上，能利用已有知識進行推論。這常依賴語言模型的語意理解能力與提示工程（prompting）。應用包括多語言翻譯與知識問答。

📺 [推薦 YouTube：Zero-shot Learning Explained](https://www.youtube.com/watch?v=MbDKu1C7n7Q)

🌐 [建議網站：Zero-shot Learning (Papers with Code)](https://paperswithcode.com/task/zero-shot-learning)

## Few-shot Learning小樣本學習

小樣本學習指模型僅需少量範例就能學會新任務。這在數據標註昂貴的領域特別重要，例如醫學與法律。大型語言模型（如 GPT-4）展現了強大的小樣本能力。

📺 [推薦 YouTube：Few-shot Learning Explained](https://www.youtube.com/watch?v=VZry9WH0FJk)

🌐 [建議網站：Few-shot Learning (Wikipedia)](https://en.wikipedia.org/wiki/Few-shot_learning)

# 第 4 章 大型語言模型 (LLM) 與工具

## Large Language Model (LLM)大型語言模型

大型語言模型（LLM）是一類基於 Transformer 架構的深度學習模型，透過在大規模文本數據上進行預訓練，學會語言的結構與知識，能夠執行多種自然語言處理任務，例如翻譯、摘要、對話與推理。代表模型包括 GPT、BERT、LLaMA 等。LLM 的能力來自於龐大的參數量（通常數十億以上）以及多樣化的訓練數據。近年來，LLM 已被廣泛應用於客服自動化、教育、程式碼生成、醫療文本分析等領域，並推動生成式 AI 熱潮。

📺 [推薦 YouTube：What are Large Language Models?（IBM Technology）](https://www.youtube.com/watch?v=5sLYAQS9sWQ)

🌐 [建議網站：Stanford CS324: Large Language Models](https://web.stanford.edu/class/cs324/)

## Token標記

Token 是 LLM 中處理文字的最小單位。它可以是一個字母、一個詞或詞的一部分。模型會將輸入的文字切分成 token，再進行編碼處理。例如『人工智慧』可能會被分成「人工」與「智慧」兩個 token。token 的數量影響模型的計算量與輸入長度限制（context window）。

📺 [推薦 YouTube：What is a Token in LLMs?](https://www.youtube.com/watch?v=zIz6U9iRk0o)

🌐 [建議網站：OpenAI Tokenizer](https://platform.openai.com/tokenizer)

## Embedding向量嵌入

Embedding 是將文字轉換為數值向量的技術，使電腦能理解語意關係。透過嵌入，語意相近的詞在向量空間中的距離也會接近。Embedding 廣泛應用於搜尋、推薦、語意檢索與文本分類。向量資料庫通常依賴 embedding 來進行相似度比對。

📺 [推薦 YouTube：Word Embeddings Explained](https://www.youtube.com/watch?v=ERibwqs9p38)

🌐 [建議網站：OpenAI Embeddings API](https://platform.openai.com/docs/guides/embeddings)

## Context Window上下文視窗

Context Window 指大型語言模型在單次推理時所能處理的最大輸入長度（以 token 為單位）。它限制了模型在對話或生成文本時，能夠同時考慮多少上下文資訊。例如 GPT-3 的 context window 是 2048 個 token，而 GPT-4 可以達到數萬 token。Context Window 的大小直接影響模型的應用場景，例如長文本分析、程式碼生成或文檔摘要。

📺 [推薦 YouTube：Context Windows in LLMs Explained](https://www.youtube.com/watch?v=Qg8FHNv8wFI)

🌐 [建議網站：OpenAI API Documentation](https://platform.openai.com/docs/)

## Prompt Engineering提示工程

Prompt Engineering 是設計與優化提示詞（prompt）來引導模型產生期望輸出的技巧。透過不同的提示方式，模型的回答品質可能有顯著差異。應用包括問答系統、程式碼生成、創意寫作等。近年來，提示工程成為一門新興技能，甚至出現專門的職位。

📺 [推薦 YouTube：Prompt Engineering for LLMs](https://www.youtube.com/watch?v=dOxUroR57xs)

🌐 [建議網站：Learn Prompting (Open Source Guide)](https://learnprompting.org/)

## Chain of Thought (CoT)思維鏈

Chain of Thought 是讓模型透過逐步推理來產生答案的方法。它模仿人類解題時『寫下中間步驟』的方式，幫助模型更準確地回答複雜問題。在數學推理、邏輯問題與多步驟任務中特別有效。

📺 [推薦 YouTube：Chain of Thought Prompting](https://www.youtube.com/watch?v=dOxUroR57xs)

🌐 [建議網站：Google Research: Chain of Thought Paper](https://arxiv.org/abs/2201.11903)

## RAG (Retrieval-Augmented Generation)– 檢索增強生成

RAG 是結合檢索與生成的方法，先從外部知識庫檢索相關資訊，再由語言模型生成答案。它能減少模型幻覺（hallucination），並提升專業知識的準確性。RAG 常用於企業知識問答、文件查詢與法律/醫療領域的應用。

📺 [推薦 YouTube：What is RAG? (Retrieval-Augmented Generation)](https://www.youtube.com/watch?v=s3k-3N9qu4A)

🌐 [建議網站：Meta AI: Retrieval-Augmented Generation](https://ai.meta.com/research/publications/retrieval-augmented-generation-for-knowledge-intensive-nlp-tasks/)

## Instruction Tuning指令微調

Instruction Tuning 是讓模型透過訓練來更好地遵循人類指令。它使用成對的『指令-回答』數據，讓模型學會依據不同任務需求產生輸出。Instruction Tuning 是 ChatGPT 能正確回應使用者指令的重要原因之一。

📺 [推薦 YouTube：Instruction Tuning Explained](https://www.youtube.com/watch?v=hBXQp4lJ0Uo)

🌐 [建議網站：FLAN: Instruction Tuning Paper](https://arxiv.org/abs/2210.11416)

## RLHF (Reinforcement Learning from Human Feedback)人類回饋強化學習

RLHF 是利用人類標註者的偏好來優化模型行為的方法。流程包括先訓練一個獎勵模型，再用強化學習（如 PPO）調整語言模型的輸出。這種方法讓 ChatGPT 能產生更符合人類價值觀與需求的回答。

📺 [推薦 YouTube：RLHF Explained](https://www.youtube.com/watch?v=RLHF_Paper)

🌐 [建議網站：OpenAI Blog: Fine-Tuning with RLHF](https://openai.com/research/learning-from-human-feedback)

## Tool Use /Function Calling工具使用 / 函數調用

Tool Use 指語言模型能透過外部 API 或工具完成任務，例如查詢資料庫、計算數學、發送郵件。OpenAI 的 Function Calling 讓開發者定義函數，並由模型決定何時呼叫。這使得語言模型不僅是文字生成器，更能成為實用的數位助理。

📺 [推薦 YouTube：OpenAI Function Calling Demo](https://www.youtube.com/watch?v=FnG3Ot_MHok)

🌐 [建議網站：OpenAI Function Calling Documentation](https://platform.openai.com/docs/guides/gpt/function-calling)

## Multi-Modal AI多模態 AI

多模態 AI 能同時處理文字、圖片、音訊與影片等不同型態的數據。例如 GPT-4、Gemini 都能進行文字 + 圖片輸入。多模態 AI 擴展了應用場景，如醫療影像診斷、影音分析與輔助創作。

📺 [推薦 YouTube：What is Multimodal AI?](https://www.youtube.com/watch?v=2RwgkD0N7xI)

🌐 [建議網站：Google DeepMind Multimodal AI](https://deepmind.google/)

## Knowledge Graph知識圖譜

知識圖譜是一種用節點與關係表示知識的結構化資料庫。它能讓機器更好地理解概念之間的關聯。Google 搜尋、推薦系統與醫療知識管理常使用知識圖譜。

📺 [推薦 YouTube：Knowledge Graphs Explained](https://www.youtube.com/watch?v=1O_BenficgE)

🌐 [建議網站：Google Knowledge Graph](https://developers.google.com/knowledge-graph)

## Vector Database向量資料庫

向量資料庫專門用於儲存與檢索高維度向量（embedding）。它能透過相似度搜尋快速找到語意相近的內容，是 RAG 系統的核心組件。常見應用包括搜尋引擎、推薦系統與問答系統。

📺 [推薦 YouTube：Vector Databases Explained](https://www.youtube.com/watch?v=QxFjl9_2WJ8)

🌐 [建議網站：Vector Databases (Weaviate, Pinecone)](https://www.pinecone.io/learn/vector-database/)

## Pinecone向量資料庫平台

Pinecone 是一個商業化的雲端向量資料庫，提供高效能的相似度檢索服務。它特別適合用於構建 RAG 系統、語意搜尋與推薦引擎。Pinecone 提供 API 方便與 LLM 整合，讓開發者能快速搭建智慧應用。

📺 [推薦 YouTube：Pinecone Tutorial](https://www.youtube.com/watch?v=wACD8WJQz0Y)

🌐 [建議網站：Pinecone 官方網站](https://www.pinecone.io/)

## FAISSFacebook AI Similarity Search

FAISS 是 Facebook AI 開源的向量檢索庫，能快速計算高維向量的相似度。它常用於建立語意檢索系統與向量資料庫的基礎。FAISS 對於大規模數據集特別有效，是研究與開發的常用工具。

📺 [推薦 YouTube：FAISS Tutorial](https://www.youtube.com/watch?v=U3H3WxQdYpQ)

🌐 [建議網站：FAISS GitHub](https://github.com/facebookresearch/faiss)

# 第 5 章 AI 應用領域

## Chatbot聊天機器人

聊天機器人（Chatbot）是利用自然語言處理（NLP）與機器學習技術實現人機對話的系統。它能回答常見問題、處理客服任務，甚至進行日常對話。隨著 LLM 的發展，現代 Chatbot 已不再局限於固定腳本，而能生成更自然、靈活的回應。應用場景包括網站客服、銀行諮詢、教育輔導與心理健康支持。

📺 [推薦 YouTube：How Chatbots Work (IBM Technology)](https://www.youtube.com/watch?v=4jmsHaJ7xEA)

🌐 [建議網站：Dialogflow (Google Chatbot Platform)](https://cloud.google.com/dialogflow)

## Virtual Assistant虛擬助理

虛擬助理是 Chatbot 的進階版本，能理解語音或文字指令並執行任務。典型例子包括 Apple Siri、Amazon Alexa 與 Google Assistant。虛擬助理能幫助用戶查詢資訊、設定提醒、控制智慧家居，甚至進行購物。隨著 AI 的進步，它們正在成為生活與工作的數位助手。

📺 [推薦 YouTube：Virtual Assistants Explained](https://www.youtube.com/watch?v=6b0QFqDjcJg)

🌐 [建議網站：Amazon Alexa Official](https://developer.amazon.com/alexa)

## Recommendation System推薦系統

推薦系統利用使用者行為與偏好來提供個性化建議。它是電商與影音平台（如 Netflix、YouTube、Shopee）的核心技術之一。推薦系統方法包括協同過濾、內容為本與混合模型。這些系統能提升使用者體驗與商業收益，但也引發過濾泡泡與資訊偏見的問題。

📺 [推薦 YouTube：Recommendation Systems Explained](https://www.youtube.com/watch?v=ZspR5PZemcs)

🌐 [建議網站：Recommender Systems Handbook](https://link.springer.com/book/10.1007/978-0-387-85820-3)

## Predictive Analytics預測分析

預測分析是利用統計與機器學習模型，從歷史數據中找出模式並預測未來事件。它常用於需求預測、設備維護、醫療診斷與財務分析。透過結合 AI，預測分析能更快、更準確地捕捉趨勢。

📺 [推薦 YouTube：Predictive Analytics Explained](https://www.youtube.com/watch?v=s0K4m3OubZk)

🌐 [建議網站：SAS Predictive Analytics](https://www.sas.com/en_us/insights/analytics/predictive-analytics.html)

## Fraud Detection詐欺檢測

詐欺檢測系統透過 AI 模型分析交易行為，判斷是否存在異常或欺詐。在金融業特別重要，用於信用卡交易、保險理賠與網路詐騙防範。機器學習能自動學習正常與異常模式，提高檢測的準確性與即時性。

📺 [推薦 YouTube：Fraud Detection with AI](https://www.youtube.com/watch?v=E-dhYJzA7P8)

🌐 [建議網站：IBM Fraud Detection Solutions](https://www.ibm.com/analytics/fraud-detection)

## Sentiment Analysis情感分析

情感分析是 NLP 的一個應用，旨在從文字或語音中判斷情緒（如正向、負向、中立）。它廣泛應用於社群媒體監測、品牌口碑分析、客服反饋等領域。隨著深度學習與 BERT 類模型的出現，情感分析的準確率顯著提升。

📺 [推薦 YouTube：Sentiment Analysis Explained](https://www.youtube.com/watch?v=5WZl3MMT0Eg)

🌐 [建議網站：Stanford Sentiment Treebank](https://nlp.stanford.edu/sentiment/)

## Speech Recognition語音識別

語音識別技術能將口語轉換為文字，是人機互動的重要基礎。應用於語音助理、字幕生成、客服系統與醫療記錄。深度學習與端到端模型（如 Whisper）大幅提升了語音識別的準確性與魯棒性。

📺 [推薦 YouTube：How Speech Recognition Works](https://www.youtube.com/watch?v=1vxJlxgIdXs)

🌐 [建議網站：CMU Sphinx Project](https://cmusphinx.github.io/)

## Text-to-Speech (TTS)語音合成

TTS 技術將文字轉換為自然語音，應用於導航系統、閱讀輔助、虛擬助理等場景。近年來，神經網路模型（如 Tacotron、WaveNet）使語音更自然、更接近人類聲音。

📺 [推薦 YouTube：Text-to-Speech Explained](https://www.youtube.com/watch?v=H4n3dZ4lG-8)

🌐 [建議網站：Google Cloud TTS](https://cloud.google.com/text-to-speech)

## OCR(Optical Character Recognition)–光學文字識別

OCR 技術能將掃描影像或照片中的文字轉換為可編輯的數位文字。應用於檔案數位化、車牌辨識、銀行票據處理等。深度學習方法提升了 OCR 在複雜字體與噪聲環境下的表現。

📺 [推薦 YouTube：OCR Explained](https://www.youtube.com/watch?v=1FON8yx1sLs)

🌐 [建議網站：Tesseract OCR](https://github.com/tesseract-ocr/tesseract)

## Autonomous Vehicle自駕車

自駕車結合電腦視覺、感測器融合、深度學習與控制系統，實現自動駕駛功能。應用於智慧交通、物流運輸與無人計程車。代表性企業包括 Tesla、Waymo、百度 Apollo。自駕車的發展涉及安全、法律與倫理挑戰。

📺 [推薦 YouTube：How Self-Driving Cars Work](https://www.youtube.com/watch?v=tj4nMxBE3G8)

🌐 [建議網站：Waymo Official](https://waymo.com/)

## Robotics機器人

機器人技術結合 AI、感測器與控制系統，能執行複雜任務。應用於製造、自動化倉儲、醫療手術、救援與家用清潔。AI 讓機器人能更智慧化，例如透過電腦視覺與語音互動適應環境。

📺 [推薦 YouTube：AI in Robotics](https://www.youtube.com/watch?v=VyKXG4UVX5U)

🌐 [建議網站：IEEE Robotics and Automation Society](https://www.ieee-ras.org/)

## Digital Twin數位孿生

數位孿生是物理系統的數位化映射，能即時模擬、監控與優化實體資產。應用於製造、建築、能源與智慧城市。AI 結合數位孿生能進行預測性維護與流程優化。

📺 [推薦 YouTube：What is a Digital Twin?](https://www.youtube.com/watch?v=J0XzLZ5qEBo)

🌐 [建議網站：Siemens Digital Twin](https://new.siemens.com/global/en/company/topic-areas/digital-twin.html)

## Smart Manufacturing智慧製造

智慧製造結合 IoT、AI 與自動化，提升生產效率與品質。應用於工業 4.0，包括智慧工廠、即時監控與流程優化。

📺 [推薦 YouTube：Smart Manufacturing Explained](https://www.youtube.com/watch?v=FQxg1n8zJHQ)

🌐 [建議網站：World Economic ForumSmart Manufacturing](https://www.weforum.org/)

## Predictive Maintenance預測性維護

預測性維護利用 AI 與感測器數據，預測設備可能故障的時間。能降低維修成本並避免停機。常應用於製造、能源與運輸。

📺 [推薦 YouTube：Predictive Maintenance with AI](https://www.youtube.com/watch?v=pMxEvMGMazY)

🌐 [建議網站：IBM Predictive Maintenance](https://www.ibm.com/topics/predictive-maintenance)

## Edge AI邊緣人工智慧

Edge AI 指在本地設備（如手機、IoT 裝置）上運行 AI 模型，而非依賴雲端伺服器。優勢是低延遲、更安全（資料不需上傳）、可離線運作。應用於智慧家居、可穿戴裝置、智慧相機與工業控制。

📺 [推薦 YouTube：What is Edge AI?](https://www.youtube.com/watch?v=0rDM9yx6-pA)

🌐 [建議網站：Edge AI (NVIDIA)](https://www.nvidia.com/en-us/edge-computing/)

## AI in Healthcare醫療中的 AI

AI 在醫療中應用廣泛，包括醫學影像診斷、藥物研發、病歷分析與個人化治療。AI 輔助醫生提高診斷準確性並提升醫療效率。

📺 [推薦 YouTube：AI in Healthcare](https://www.youtube.com/watch?v=lnjE1KQJPz8)

🌐 [建議網站：WHO: Ethics and Governance of AI in Health](https://www.who.int/publications/i/item/9789240029200)

## AI in Finance金融中的 AI

AI 在金融領域應用包括詐欺偵測、信用評分、演算法交易與個人化投資建議。它能提升效率與風險控制，但也需要注意合規與透明度。

📺 [推薦 YouTube：AI in Finance Explained](https://www.youtube.com/watch?v=I3FW2lyE6to)

🌐 [建議網站：OECD AI in Finance](https://www.oecd.org/finance/)

## AI in Education教育中的 AI

AI 在教育中的應用包括自適應學習平台、智慧輔導、學習分析與自動批改。它能根據學生需求提供個性化學習路徑。

📺 [推薦 YouTube：AI in Education](https://www.youtube.com/watch?v=2p0gG82NfRs)

🌐 [建議網站：UNESCO: AI and Education](https://unesdoc.unesco.org/ark:/48223/pf0000376709)

## AI in Retail零售中的 AI

AI 在零售業的應用包括商品推薦、需求預測、供應鏈優化與智慧客服。零售商利用 AI 提升顧客體驗並降低成本。

📺 [推薦 YouTube：AI in Retail Explained](https://www.youtube.com/watch?v=G7D6z_J7zEY)

🌐 [建議網站：McKinsey: AI in Retail](https://www.mckinsey.com/industries/retail)

## Deepfake深偽技術

Deepfake 技術利用生成模型（如 GAN）製造以假亂真的影像與影片。雖可用於娛樂與創意產業，但也帶來嚴重的資訊安全與倫理問題。政府與企業正致力於開發檢測工具以識別 Deepfake。

📺 [推薦 YouTube：Deepfake Explained](https://www.youtube.com/watch?v=Ro8b69VeL9U)

🌐 [建議網站：Deepfake Detection Challenge](https://ai.facebook.com/datasets/dfdc/)

# 第 6 章 資料處理與數據技術

## Dataset資料集

資料集是機器學習與深度學習模型訓練的基礎，包含輸入數據與對應標籤（在監督式學習中）。常見的資料集包括 MNIST（手寫數字）、ImageNet（影像分類）、COCO（物件檢測）與 GLUE（自然語言處理）。一個良好的資料集需具備多樣性、平衡性與高品質標註，以確保模型具備泛化能力。在 AI 開發流程中，資料集佔據了成功與否的關鍵地位，往往比演算法選擇更影響最終效果。

📺 [推薦 YouTube：What is a Dataset?](https://www.youtube.com/watch?v=5c8vKfLP4C8)

🌐 [建議網站：Kaggle Datasets](https://www.kaggle.com/datasets)

## Data Labeling資料標註

資料標註是將資料加上正確標籤的過程，例如在影像上標註物件邊界框，或在文本中標記情感傾向。標註是監督式學習必不可少的步驟，直接影響模型的學習效果與準確率。隨著 AI 的應用擴展，標註需求龐大，許多公司依賴眾包平台或自動標註技術來提升效率。資料標註的品質管理至關重要，錯誤標註可能導致模型學習錯誤模式。

📺 [推薦 YouTube：What is Data Labeling?](https://www.youtube.com/watch?v=U1x0r4H4t34)

🌐 [建議網站：Scale AI Data Labeling](https://scale.com/)

## Data Augmentation資料增強

資料增強是透過對原始數據進行轉換（如影像旋轉、翻轉、裁剪，或在文本中進行同義詞替換）來產生新的數據樣本。它能提升模型的泛化能力，減少過擬合問題。在影像識別中，增強技術尤為常見，例如隨機裁切或光線調整。隨著生成式 AI 的發展，合成數據（synthetic data）也逐漸成為增強的一部分。

📺 [推薦 YouTube：Data Augmentation Explained](https://www.youtube.com/watch?v=J2Bl1JL3UO0)

🌐 [建議網站：Albumentations Library](https://albumentations.ai/)

## Data Cleaning資料清理

資料清理是數據處理中至關重要的一步，旨在去除或修正錯誤、不完整或重複的資料。常見步驟包括處理缺失值、刪除重複資料、標準化格式與識別異常值。高品質的資料清理能確保模型在訓練時不會受到噪聲影響，提升準確度。

📺 [推薦 YouTube：Data Cleaning Explained](https://www.youtube.com/watch?v=q60dG7F5fSU)

🌐 [建議網站：Data Cleaning (Towards Data Science](https://towardsdatascience.com/the-ultimate-guide-to-data-cleaning-3969843991d4)

## Feature Engineering特徵工程

特徵工程是將原始數據轉換為更適合模型學習的特徵。它可能包括數值轉換、類別編碼、特徵縮放、特徵組合與衍生新特徵。特徵工程的品質往往決定模型的最終性能，是數據科學中最重要的環節之一。

📺 [推薦 YouTube：Feature Engineering Explained](https://www.youtube.com/watch?v=N3tRFayqVtk)

🌐 [建議網站：Feature Engineering Guide (Kaggle)](https://www.kaggle.com/learn/feature-engineering)

## Feature Selection特徵選擇

特徵選擇是從眾多特徵中選出最具代表性的子集，去除冗餘或無關的特徵。常見方法包括過濾法（Filter）、包裝法（Wrapper）、嵌入法（Embedded）。良好的特徵選擇能降低模型複雜度、減少過擬合並提升效能。

📺 [推薦 YouTube：Feature Selection Explained](https://www.youtube.com/watch?v=JUuQZRPoOJ4)

🌐 [建議網站：Feature Selection Techniques](https://scikit-learn.org/stable/modules/feature_selection.html)

## Dimensionality Reduction降維

降維技術能將高維數據壓縮到低維空間，同時保留主要結構與資訊。這不僅能降低計算成本，也能提升可視化效果。典型應用包括資料探索、壓縮與加速模型訓練。

📺 [推薦 YouTube：Dimensionality Reduction Explained](https://www.youtube.com/watch?v=9iol3Lk6KyI)

🌐 [建議網站：Dimensionality Reduction (Wikipedia)](https://en.wikipedia.org/wiki/Dimensionality_reduction)

## PCA (Principal Component Analysis)主成分分析

PCA 是最常見的降維方法，透過線性變換找到數據中方差最大的方向作為主成分。它能有效去除冗餘特徵，並提升資料可視化的表達能力。PCA 被廣泛應用於圖像壓縮、金融數據分析與探索性數據分析。

📺 [推薦 YouTube：PCA Explained (StatQuest)](https://www.youtube.com/watch?v=FgakZw6K1QQ)

🌐 [建議網站：PCA in scikit-learn](https://scikit-learn.org/stable/modules/decomposition.html#pca)

## t-SNE (t-distributed Stochastic Neighbor Embedding)

t-SNE 是一種非線性降維方法，特別適合用於高維數據的可視化。它能將相似的數據點映射到相近的位置，常用於 NLP 與影像特徵的視覺化探索。缺點是計算成本高，且不適合非常大規模的數據。

📺 [推薦 YouTube：t-SNE Explained](https://www.youtube.com/watch?v=NEaUSP4YerM)

🌐 [建議網站：t-SNE (Wikipedia)](https://en.wikipedia.org/wiki/T-distributed_stochastic_neighbor_embedding)

## UMAP (Uniform Manifold Approximation and Projection)

UMAP 是一種新興的降維方法，比 t-SNE 更快，且能保留更多全域結構。它已成為數據科學中高維可視化的熱門工具，特別適合大數據集。

📺 [推薦 YouTube：UMAP Explained](https://www.youtube.com/watch?v=nq6iPZVUxZU)

🌐 [建議網站：UMAP Documentation](https://umap-learn.readthedocs.io/en/latest/)

## Data Pipeline數據管線

數據管線是數據處理流程的自動化框架，涵蓋數據收集、清理、轉換、訓練與部署。良好的數據管線能提升效率並保證數據一致性。在 AI 工程中，數據管線與 MLOps 密切相關。

📺 [推薦 YouTube：What is a Data Pipeline?](https://www.youtube.com/watch?v=x8gOpEJz16A)

🌐 [建議網站：Google Cloud Data Pipeline](https://cloud.google.com/architecture/data-pipeline-design)

## ETL (Extract, Transform, Load)

ETL 是傳統數據處理的三個步驟：擷取（Extract）、轉換（Transform）、載入（Load）。它是數據倉儲與數據整合的重要流程。在現代 AI 應用中，ETL 常被擴展到 ELT 或與即時流數據處理結合。

📺 [推薦 YouTube：ETL Explained](https://www.youtube.com/watch?v=Y1tg0H_1G0A)

🌐 [建議網站：ETL (Wikipedia)](https://en.wikipedia.org/wiki/Extract,_transform,_load)

## Big Data大數據

大數據指的是數據量龐大、類型多樣且生成速度極快的數據集合。它的特徵通常用 3V 描述：Volume（大量）、Variety（多樣）、Velocity（高速）。AI 技術能從大數據中提取價值，應用於商業分析、醫療、金融與政府治理。

📺 [推薦 YouTube：What is Big Data?](https://www.youtube.com/watch?v=8rjU7FM8TwI)

🌐 [建議網站：Big Data (IBM)](https://www.ibm.com/topics/big-data-analytics)

## Data Warehouse數據倉儲

數據倉儲是集中存放與管理結構化數據的系統，支援企業的商業智慧（BI）分析。它通常使用 ETL 流程將來自不同來源的數據整合後再儲存。數據倉儲適合查詢與報表，但不適合即時數據處理。

📺 [推薦 YouTube：Data Warehouse Explained](https://www.youtube.com/watch?v=tD3bxp-i4nE)

🌐 [建議網站：Snowflake Data Warehouse](https://www.snowflake.com/en/data-cloud/workloads/data-warehouse/)

## Data Lake數據湖

數據湖是能儲存結構化與非結構化數據的大型存儲庫。與數據倉儲不同，數據湖允許原始數據以原生格式存放，適合大數據與 AI 訓練需求。雲端服務（如 AWS S3、Azure Data Lake）常被用來建構數據湖。

📺 [推薦 YouTube：What is a Data Lake?](https://www.youtube.com/watch?v=RrKRN9zRBWs)

🌐 [建議網站：AWS Data Lake](https://aws.amazon.com/big-data/datalakes-and-analytics/what-is-a-data-lake/)

# 第 7 章 數學與理論基礎

## Linear Algebra線性代數

線性代數是 AI 與機器學習的數學基礎，研究向量、矩陣及其運算。在深度學習中，資料通常以矩陣形式表示，例如影像是一個像素矩陣，文字向量化後也以矩陣處理。矩陣乘法是神經網路運算的核心，用於權重與輸入的線性組合。掌握線性代數能幫助理解神經網路結構、降維方法（PCA）、以及優化過程中的數值穩定性。

📺 [推薦 YouTube：Essence of Linear Algebra（3Blue1Brown）](https://www.youtube.com/playlist?list=PLZHQObOWTQDPD3MizzM2xVFitgF8hE_ab)

🌐 [建議網站：Linear Algebra (Khan Academy)](https://www.khanacademy.org/math/linear-algebra)

## Vector向量

向量是線性代數的核心概念，表示具有大小與方向的數學對象。在 AI 中，向量用來表示特徵，例如一張圖片的像素值或一段文字的詞嵌入。向量運算（點積、內積、外積）是計算相似度與轉換特徵空間的基礎。理解向量能幫助我們更好地掌握模型如何表示與處理資料。

📺 [推薦 YouTube：Vectors Explained (3Blue1Brown)](https://www.youtube.com/watch?v=fNk_zzaMoSs)

🌐 [建議網站：Vector Basics (Khan Academy)](https://www.khanacademy.org/math/linear-algebra/vectors-and-spaces)

## Matrix矩陣

矩陣是數字的二維陣列，是線性代數的重要元素。在機器學習中，矩陣用來表示資料集（行代表樣本，列代表特徵）、權重參數與轉換操作。矩陣分解（如 SVD、特徵分解）在降維、推薦系統與數據壓縮中非常重要。矩陣的運算效率也決定了深度學習的訓練速度，因此 GPU 被廣泛用於加速矩陣運算。

📺 [推薦 YouTube：Matrices Explained (3Blue1Brown)](https://www.youtube.com/watch?v=juCQZwzFzLU)

🌐 [建議網站：Matrix Operations (Khan Academy)](https://www.khanacademy.org/math/linear-algebra/matrix-transformations)

## Eigenvalue & Eigenvector特徵值與特徵向量

特徵值與特徵向量是線性代數中的核心概念。在線性變換下，特徵向量的方向保持不變，而特徵值則描述了其縮放比例。在機器學習中，PCA（主成分分析）利用特徵分解來找到數據的主要變異方向。這對於降維、數據壓縮與模式識別至關重要。

📺 [推薦 YouTube：Eigenvalues and Eigenvectors (3Blue1Brown)](https://www.youtube.com/watch?v=PFDu9oVAE-g)

🌐 [建議網站：Eigenvalues and Eigenvectors (Khan Academy)](https://www.khanacademy.org/math/linear-algebra/alternate-bases/eigen-everything)

## Probability機率

機率是數學中研究隨機事件發生可能性的分支。在 AI 中，機率用來建模不確定性，例如在貝葉斯網路中表示變數間的依賴關係。理解機率分布（如常態分布、伯努利分布）是掌握機器學習演算法的基礎。

📺 [推薦 YouTube：Probability Basics (Khan Academy)](https://www.youtube.com/watch?v=uzkc-qNVoOk)

🌐 [建議網站：Probability (Khan Academy)](https://www.khanacademy.org/math/statistics-probability/probability-library)

## Bayes’ Theorem貝葉斯定理

貝葉斯定理提供了一種在獲取新證據後更新事件發生機率的方法。公式為 P(A|B) = P(B|A)P(A)/P(B)。它是貝葉斯推論的基礎，應用於分類（如朴素貝葉斯）、醫療診斷與垃圾郵件過濾。

📺 [推薦 YouTube：Bayes’ Theorem Explained](https://www.youtube.com/watch?v=HZGCoVF3YvM)

🌐 [建議網站：Bayes’ Theorem (Wikipedia)](https://en.wikipedia.org/wiki/Bayes%27_theorem)

## Conditional Probability條件機率

條件機率是指在已知某事件發生的前提下，另一事件發生的機率。公式為 P(A|B) = P(A∩B)/P(B)。它在機器學習中廣泛應用，例如在隱馬可夫模型與貝葉斯分類器中。

📺 [推薦 YouTube：Conditional Probability Explained](https://www.youtube.com/watch?v=9rIy0xY99a0)

🌐 [建議網站：Conditional Probability (Khan Academy)](https://www.khanacademy.org/math/statistics-probability/probability-library)

## Random Variable隨機變數

隨機變數是將隨機事件結果數值化的函數。它分為離散型與連續型。在 AI 中，隨機變數常用於描述不確定性，例如隨機梯度下降中的樣本選取。

📺 [推薦 YouTube：Random Variables Explained](https://www.youtube.com/watch?v=9g1dR7afFJM)

🌐 [建議網站：Random Variables (Khan Academy)](https://www.khanacademy.org/math/statistics-probability/random-variables-stats-library)

## Expectation & Variance期望與變異數

期望值是隨機變數的加權平均，表示其長期平均結果。變異數衡量數據與期望的偏離程度。在機器學習中，期望與變異數用於描述模型的偏差-變異權衡（Bias-Variance Tradeoff）。

📺 [推薦 YouTube：Expectation and Variance Explained](https://www.youtube.com/watch?v=MRqtXL2WX2M)

🌐 [建議網站：Expectation and Variance (Khan Academy)](https://www.khanacademy.org/math/statistics-probability/random-variables-stats-library)

## Entropy熵

熵是衡量資訊不確定性的指標。在機器學習中，熵常用於決策樹的分裂判斷。例如 ID3 演算法使用資訊增益（基於熵的變化）來選擇最佳特徵。

📺 [推薦 YouTube：Entropy Explained (StatQuest)](https://www.youtube.com/watch?v=YtebGVx-Fxw)

🌐 [建議網站：Entropy (Wikipedia)](https://en.wikipedia.org/wiki/Entropy_(information_theory))

## Information Gain資訊增益

資訊增益衡量因特徵劃分數據後，資料不確定性（熵）的減少量。在決策樹中，常以資訊增益作為選擇分裂特徵的標準。

📺 [推薦 YouTube：Information Gain Explained](https://www.youtube.com/watch?v=IRkdJxjAjNc)

🌐 [建議網站：Decision Trees and Information Gain](https://scikit-learn.org/stable/modules/tree.html)

## Gradient梯度

梯度是函數對變數的偏導數向量，表示函數在某點的最陡上升方向。在機器學習中，梯度用於優化演算法，幫助模型找到最小化損失的參數。

📺 [推薦 YouTube：Gradients Explained (3Blue1Brown)](https://www.youtube.com/watch?v=IHZwWFHWa-w)

🌐 [建議網站：Gradient (Wikipedia)](https://en.wikipedia.org/wiki/Gradient)

## Derivative微分

微分描述函數在某一點的瞬時變化率。它是梯度下降與反向傳播的數學基礎。理解微分有助於掌握模型如何學習與更新參數。

📺 [推薦 YouTube：Derivatives Explained (Khan Academy)](https://www.youtube.com/watch?v=JYBqT2hU6rY)

🌐 [建議網站：Differential Calculus (Khan Academy)](https://www.khanacademy.org/math/differential-calculus)

## Convex Function凸函數

凸函數是數學分析中的重要概念，其特點是任意兩點連線都不低於函數圖像。在最佳化中，凸函數保證局部最小值即為全局最小值。許多機器學習演算法（如 SVM、線性迴歸）依賴凸函數來確保可解性。

📺 [推薦 YouTube：Convex Functions Explained](https://www.youtube.com/watch?v=9uu3RLAj-Jo)

🌐 [建議網站：Convex Optimization (Stanford)](https://web.stanford.edu/~boyd/cvxbook/)

## Optimization最佳化

最佳化是尋找目標函數最小值或最大值的過程，是機器學習模型訓練的核心。方法包括梯度下降、牛頓法與隨機優化。最佳化影響模型的收斂速度與最終性能，是深度學習成功的關鍵因素。

📺 [推薦 YouTube：Optimization Explained](https://www.youtube.com/watch?v=XdM6ER7zTLk)

🌐 [建議網站：Convex Optimization (Boyd & Vandenberghe)](https://web.stanford.edu/~boyd/cvxbook/)

# 第 8 章 AI 工具與平台

## TensorFlowGoogle 開源深度學習框架

TensorFlow 是 Google Brain 團隊開發的開源深度學習框架，自 2015 年發布以來成為 AI 領域的主流工具之一。它支援多種語言（Python、C++、Java 等），並能在 CPU、GPU、TPU 上運行。TensorFlow 特別適合大規模分散式訓練，廣泛應用於電腦視覺、自然語言處理與強化學習。其生態系統包含 TensorFlow Lite（行動裝置）、TensorFlow.js（網頁端）與 TensorFlow Extended（MLOps）。

📺 [推薦 YouTube：TensorFlow in 5 Minutes](https://www.youtube.com/watch?v=2FmcHiLCwTU)

🌐 [建議網站：TensorFlow 官方網站](https://www.tensorflow.org/)

## PyTorchFacebook 開源深度學習框架

PyTorch 是 Facebook AI Research (FAIR) 開發的開源深度學習框架，強調動態計算圖與易用性。它在研究界特別受歡迎，被廣泛用於 NLP 與 CV 領域。PyTorch 支援 GPU 加速，並有強大的社群支持。近年來，PyTorch 也逐漸成為產業應用的標準，許多新模型（如 GPT、BERT）都是基於 PyTorch 訓練的。

📺 [推薦 YouTube：PyTorch in 5 Minutes](https://www.youtube.com/watch?v=GIsg-ZUy0MY)

🌐 [建議網站：PyTorch 官方網站](https://pytorch.org/)

## Keras高階神經網路 API

Keras 是由 François Chollet 開發的高階神經網路 API，最初基於 TensorFlow 和 Theano。它提供簡單直觀的介面，讓使用者能快速建立深度學習模型。Keras 特別適合入門者與需要快速原型設計的研究人員。目前 Keras 已成為 TensorFlow 官方高階 API，廣泛應用於教育與實務專案。

📺 [推薦 YouTube：Keras Crash Course](https://www.youtube.com/watch?v=qFJeN9V1ZsI)

🌐 [建議網站：Keras 官方網站](https://keras.io/)

## JAX高效數值計算與自動微分庫

JAX 是 Google 開發的數值計算庫，結合了 NumPy 介面與自動微分功能。它支援 GPU 與 TPU 加速，特別適合大規模機器學習研究。JAX 在研究界被廣泛應用於深度學習、強化學習與科學計算。其優勢在於函數式編程風格與高效能，並能與 Flax、Haiku 等框架搭配使用。

📺 [推薦 YouTube：Introduction to JAX](https://www.youtube.com/watch?v=wG_nF1awSSY)

🌐 [建議網站：JAX 官方網站](https://jax.readthedocs.io/)

## Scikit-learn機器學習工具庫

Scikit-learn 是 Python 的開源機器學習套件，提供分類、迴歸、聚類、降維等演算法。它特別適合傳統機器學習方法，例如決策樹、SVM、隨機森林與 PCA。Scikit-learn 的 API 簡單直觀，是入門機器學習的首選工具之一。

📺 [推薦 YouTube：Scikit-learn Tutorial](https://www.youtube.com/watch?v=0Lt9w-BxKFQ)

🌐 [建議網站：Scikit-learn 官方網站](https://scikit-learn.org/)

## Hugging Face TransformersNLP 模型庫

Hugging Face Transformers 是目前最受歡迎的 NLP 模型庫，收錄 BERT、GPT、T5、RoBERTa 等上千種預訓練模型。它提供簡單 API，可快速載入並微調模型，支援文本分類、生成、翻譯與問答。Hugging Face 也推出 Hub 與 Datasets 平台，推動了 AI 開源社群的繁榮。

📺 [推薦 YouTube：Hugging Face Transformers Tutorial](https://www.youtube.com/watch?v=tiZFewofSLM)

🌐 [建議網站：Hugging Face 官方網站](https://huggingface.co/transformers)

## LangChainLLM 應用框架

LangChain 是專門為大型語言模型（LLM）設計的開源框架。它讓開發者能將 LLM 與外部工具、知識庫與 API 整合，構建複雜應用。應用場景包括 RAG 系統、聊天機器人、自動化代理（AI Agent）。

📺 [推薦 YouTube：LangChain Crash Course](https://www.youtube.com/watch?v=aywZrzNaKjs)

🌐 [建議網站：LangChain 官方網站](https://www.langchain.com/)

## OpenAI APIOpenAI 應用程式介面

OpenAI API 提供對 GPT、Whisper、DALL·E 等模型的雲端存取服務。開發者能透過 API 實現對話、生成文字、圖像生成與語音轉換。OpenAI API 已廣泛應用於客服、教育、程式碼助理與創作工具。

📺 [推薦 YouTube：OpenAI API Tutorial](https://www.youtube.com/watch?v=kCc8FmEb1nY)

🌐 [建議網站：OpenAI 官方網站](https://platform.openai.com/)

## Google Cloud AIGoogle 雲端 AI 服務

Google Cloud 提供一系列 AI 與機器學習服務，包括 AutoML、Vertex AI、Vision API、Translation API。它讓企業能快速部署 AI 應用，而無需自行訓練模型。Google Cloud AI 特別適合需要可擴展性與穩定性的企業級應用。

📺 [推薦 YouTube：Google Cloud AI Overview](https://www.youtube.com/watch?v=k8asfS24Mbk)

🌐 [建議網站：Google Cloud AI](https://cloud.google.com/ai)

## AWS AI Services亞馬遜雲端 AI 服務

AWS 提供多種 AI 服務，包括 Rekognition（影像識別）、Comprehend（文本分析）、Polly（TTS）、Lex（聊天機器人）。AWS 是全球最大的雲端服務供應商之一，AI 服務被廣泛應用於金融、零售與醫療領域。

📺 [推薦 YouTube：AWS AI Services Overview](https://www.youtube.com/watch?v=Tw8UjbiJbt0)

🌐 [建議網站：AWS AI Services](https://aws.amazon.com/machine-learning/ai-services/)

## Microsoft Azure AI微軟雲端 AI

Azure AI 提供 Cognitive Services（語音、文字、影像 API）、Azure Machine Learning 平台，支援模型訓練與部署。Azure AI 與微軟 Office 產品緊密整合，適合企業應用。

📺 [推薦 YouTube：Azure AI Overview](https://www.youtube.com/watch?v=ux0rT5rMZd0)

🌐 [建議網站：Azure AI 官方網站](https://azure.microsoft.com/en-us/products/ai-services/)

## IBM WatsonIBM AI 平台

IBM Watson 是 IBM 推出的 AI 平台，專注於自然語言處理與企業應用。它提供聊天機器人、語音分析與醫療解決方案。Watson 曾因在《Jeopardy!》節目中戰勝人類選手而聞名。

📺 [推薦 YouTube：IBM Watson Explained](https://www.youtube.com/watch?v=3nK0JjKxqmg)

🌐 [建議網站：IBM Watson 官方網站](https://www.ibm.com/watson)

## NVIDIA CUDA平行運算平台

CUDA 是 NVIDIA 開發的平行運算平台與 API，允許開發者利用 GPU 加速計算。深度學習的訓練大多依賴 CUDA 來提升效率。CUDA 與 cuDNN 庫一起成為 AI 基礎設施的重要組件。

📺 [推薦 YouTube：NVIDIA CUDA Explained](https://www.youtube.com/watch?v=86FAWCzIe_4)

🌐 [建議網站：NVIDIA CUDA](https://developer.nvidia.com/cuda-zone)

## NVIDIA TensorRT推理加速引擎

TensorRT 是 NVIDIA 提供的深度學習推理優化工具，能顯著加快模型在 GPU 上的推理速度。它特別適合需要低延遲的應用，如自駕車、即時翻譯與醫療影像分析。

📺 [推薦 YouTube：NVIDIA TensorRT Overview](https://www.youtube.com/watch?v=xwdS9wzE6C8)

🌐 [建議網站：NVIDIA TensorRT](https://developer.nvidia.com/tensorrt)

## ONNX開放神經網路交換格式

ONNX 是由 Microsoft 與 Facebook 發起的開放格式，用於在不同框架間交換模型。它允許在 PyTorch 訓練的模型轉換到 TensorFlow 或其他框架中使用。ONNX 增強了 AI 模型的可移植性與部署靈活性。

📺 [推薦 YouTube：ONNX Explained](https://www.youtube.com/watch?v=55Bxpt5c1h8)

🌐 [建議網站：ONNX 官方網站](https://onnx.ai/)

## MLflow機器學習實驗管理平台

MLflow 是一個開源平台，用於追蹤機器學習實驗、模型管理與部署。它支援多種框架（TensorFlow、PyTorch、Scikit-learn），幫助團隊進行 MLOps。

📺 [推薦 YouTube：MLflow Tutorial](https://www.youtube.com/watch?v=5VymS63T_tM)

🌐 [建議網站：MLflow 官方網站](https://mlflow.org/)

## Weights & Biases (W&B)機器學習實驗追蹤

Weights & Biases 是一個實驗追蹤工具，支援超參數記錄、訓練過程視覺化與模型版本管理。它與主流深度學習框架無縫整合，被廣泛應用於研究與產業。

📺 [推薦 YouTube：Weights & Biases Tutorial](https://www.youtube.com/watch?v=9saM3JYLMnY)

🌐 [建議網站：Weights & Biases 官方網站](https://wandb.ai/site)

## Ray分散式運算框架

Ray 是一個高效能的分散式運算框架，支援大規模機器學習與強化學習。它提供 Ray Tune（超參數調整）、Ray Serve（模型部署）等模組。

📺 [推薦 YouTube：Ray Tutorial](https://www.youtube.com/watch?v=F6T9XoZhvxU)

🌐 [建議網站：Ray 官方網站](https://www.ray.io/)

## Apache Spark MLlib大數據機器學習庫

Apache Spark MLlib 是建構在 Spark 平台上的機器學習庫。它能處理分散式大數據，支援分類、迴歸、聚類與推薦系統。MLlib 特別適合需要同時處理 AI 與大數據的應用。

📺 [推薦 YouTube：Spark MLlib Tutorial](https://www.youtube.com/watch?v=tL5jGJnoZ2c)

🌐 [建議網站：Apache Spark MLlib](https://spark.apache.org/mllib/)

## RapidMiner機器學習平台

RapidMiner 是一個無程式碼/低程式碼的數據科學平台。它支援資料處理、模型建立與部署，特別適合商業使用者。RapidMiner 提供可視化介面，降低了 AI 應用的門檻。

📺 [推薦 YouTube：RapidMiner Tutorial](https://www.youtube.com/watch?v=5gq2l8Z3G4A)

🌐 [建議網站：RapidMiner 官方網站](https://rapidminer.com/)