

第 1 章 深度學習(Deep Learning)導論

- [1] Dylan Yeh、陳建鈞,《市值首度超越 Intel ! NVIDIA 贏在哪裡?》, 2020
(<https://www.bnext.com.tw/article/58410/nvidia-valuation-soars-past-intel-on-graphics-chip-boom>)
- [2] OpenAI's 5 Levels Of 'Super AI
(<https://www.forbes.com/sites/jodiecook/2024/07/16/openais-5-levels-of-super-ai-agi-to-outperform-human-capability/>)
- [3] ARC 2024 競賽題目
(<https://www.kaggle.com/competitions/arc-prize-2024>)
- [4] Google Cloud 官網指南
(<https://cloud.google.com/ai-platform/docs/ml-solutions-overview>)
- [5] Anaconda 官網
(<https://www.anaconda.com/download/success>)
- [6] MSVC 2022 runtime
(https://aka.ms/vs/16/release/vc_redist.x64.exe)
- [7] TensorFlow 官網『Install TensorFlow with pip』說明
(<https://www.tensorflow.org/install/pip>)
- [8] NVIDIA 官網說明
(<https://developer.nvidia.com/cuda-toolkit-archive>)
- [9] 陳昭明,《Day 01：輕鬆掌握 Keras》, 2020
(<https://ithelp.ithome.com.tw/articles/10233272>)
- [10] 選擇合適的 Colab 方案
(https://colab.research.google.com/signup?utm_source=notebook_settings&utm_medium=link&utm_campaign=premium_gpu_selector)
- [11] Colaboratory 官網說明
(<https://colab.research.google.com/notebooks/intro.ipynb>)
- [12] Mike Driscoll, Jupyter Notebook: An Introduction
(<https://realpython.com/jupyter-notebook-introduction/>)
- [13] Jupyter Notebook 完整介紹、安裝及使用說明
(<https://medium.com/ai-for-k12/jupyter-notebook-完整介紹-安裝及使用說明-846b5432f044>)

第 2 章 神經網路(Neural Network)原理

- [1] Keith McNulty, 《Decision makers need more math》, 2018
(<https://blog.keithmcnulty.org/decision-makers-need-more-math-78ba07306193>)
- [2] Seaborn 官網
(<https://seaborn.pydata.org/examples/index.html>)
- [3] 台灣彩券官網
(<https://www.taiwanlottery.com.tw/DailyCash/index.asp>)
- [4] Univariate Distribution Relationships
(<http://www.math.wm.edu/~leemis/chart/UDR/UDR.html>)
- [5] 維基百科中關於假設檢定的定義
(<https://zh.wikipedia.org/wiki/%E5%81%87%E8%AA%AA%E6%AA%A2%E5%AE%9A>)
- [6] Scribbr, Central Limit Theorem
(<https://www.scribbr.com/statistics/central-limit-theorem/>)
- [7] An Interactive Guide to Hypothesis Testing in Python
(<https://towardsdatascience.com/an-interactive-guide-to-hypothesis-testing-in-python-979f4d62d85/>)
- [8] 17 Statistical Hypothesis Tests in Python
(<https://machinelearningmastery.com/statistical-hypothesis-tests-in-python-cheat-sheet/>)
- [9] 藍白合民調之統計誤差計算，3%或 6%?
(<https://ithelp.ithome.com.tw/articles/10340531>)
- [10] 陳昭明, 《Day 14： 客服人力規劃(Workforce Planning) -- 線性規劃求解》, 2019
(<https://ithelp.ithome.com.tw/articles/10222877>)
- [11] 維基百科, 神經元
(<https://zh.wikipedia.org/wiki/神經元>)
- [12] 劉子鳴(Ziming Liu)等學者, KAN: Kolmogorov-Arnold Networks
(<https://arxiv.org/abs/2404.19756>)
- [13] Hello KAN, 建構深度學習模型的另一種思維
(<https://ithelp.ithome.com.tw/articles/10342873>)

第 3 章 TensorFlow 架構與主要功能

[1]維基百科中針對常見的深度學習套件之比較圖表

(https://en.wikipedia.org/wiki/Comparison_of_deep_learning_software)

[2] Amol Mavuduru , 《Which deep learning framework is the best?》, 2020

(<https://towardsdatascience.com/which-deep-learning-framework-is-the-best-eb51431c39a>)

[3]開發者傳授 PyTorch 秘笈

[4]同時搞定 TensorFlow、PyTorch

(<https://ithelp.ithome.com.tw/articles/10285361>)

[5] TensorFlow 官方 GitHub

(<https://github.com/tensorflow/docs/blob/master/site/en/r1/guide/extend/architecture.md>)

[6] Sonu Sharma , 《Explained: Deep Learning in Tensorflow》, 2019

(<https://towardsdatascience.com/explained-deep-learning-in-tensorflow-chapter-1-9ab389fe90a1>)

[7] TensorFlow 官網移轉指南

(<https://www.tensorflow.org/guide/migrate>)

[8] TensorFlow 官網升級指南

(<https://www.tensorflow.org/guide/upgrade>)

第 4 章 神經網路實作

- [1] 維基百科 Activation Function 的介紹
(https://en.wikipedia.org/wiki/Activation_function)
- [2] TensorFlow 優化器介紹
(https://www.tensorflow.org/api_docs/python/tf/keras/optimizers)
- [3] Keras 優化器介紹
(<https://keras.io/api/optimizers/>)
- [4] TensorFlow 損失函數介紹
(https://www.tensorflow.org/api_docs/python/tf/keras/optimizers)
- [5] Keras 損失函數介紹
(<https://keras.io/api/losses/>)
- [6] TensorFlow 效能衡量指標介紹
(https://www.tensorflow.org/api_docs/python/tf/keras/metrics)
- [7] TensorFlow 官網
(<https://www.tensorflow.org>)
- [8] Keras 官網
(<https://keras.io>)
- [9] TensorFlow 官網中 FashionMnist 的介紹
(https://www.tensorflow.org/datasets/catalog/fashion_mnist)
- [10] Keras 官網神經層介紹
(<https://keras.io/api/layers/>)
- [11] Keras 官網 Activation Function 說明
(<https://keras.io/api/layers/activations/>)
- [12] Keras 官網 Activation Layers 說明
(https://keras.io/api/layers/activation_layers/)
- [13] 維基百科關於鉸鏈損失函數的介紹
(https://zh.wikipedia.org/wiki/Hinge_loss)
- [14] 《Understanding Nesterov Momentum (NAG)》, 2018
(<https://dominikschmidt.xyz/nesterov-momentum/>)
- [15] Diederik P. Kingma、Jimmy Ba, 《Adam: A Method for Stochastic Optimization》, 2014
(<https://arxiv.org/abs/1412.6980>)

[16] 深度學習於 NLP, 《一文告訴你 Adam、AdamW、Amsgrad 區別和聯繫》, 2019

(<https://zhuanlan.zhihu.com/p/39543160>)

[17] Alec Radford's animations for optimization algorithms

(<https://www.denizyuret.com/2015/03/alec-radfords-animations-for.html>)

[18] Keras 官網效能衡量指標的介紹

(<https://keras.io/api/metrics/>)

[19] Keras 官網超參數測試範圍的設定

(<https://keras-team.github.io/keras-tuner/documentation/hyperparameters/>)

[20] Keras 官網效能調校(Tuners)的介紹(<https://keras-team.github.io/keras-tuner/documentation/tuners/>)

[21] Keras 官網 Oracle 的介紹

(<https://keras-team.github.io/keras-tuner/documentation/oracles/>)

第 5 章 TensorFlow 常用指令與功能

- [1] TensorFlow Save, serialize, and export models
(https://www.tensorflow.org/tutorials/keras/save_and_load)
- [2] Keras Save, serialize, and export models
(https://keras.io/guides/serialization_and_saving/)
- [3] Keras Callbacks API
(<https://keras.io/api/callbacks/>)
- [4] A Walkthrough with UCI Census Data
(<https://pair-code.github.io/what-if-tool/learn/tutorials/walkthrough/>)
- [5] 參考範例
(https://colab.research.google.com/github/pair-code/what-if-tool/blob/master/What_If_Tool_Notebook_Usage.ipynb)
- [6] TensorFlow TensorBoard 指南
(https://www.tensorflow.org/tensorboard/get_started)
- [7] TensorFlow Dataset 說明
(https://www.tensorflow.org/api_docs/python/tf/data/Dataset)
- [8] TensorFlow Dataset 效能說明
(https://www.tensorflow.org/guide/data_performance)

第 6 章 卷積神經網路(Convolutional Neural Network)

- [1] Prateek Karkare, 《Convolutional Neural Networks—Simplified》, 2019
(<https://medium.com/x8-the-ai-community/cnn-9c5e63703c3f>)
- [2] 《Convolutional Neural Networks—Simplified》文中卷積計算的 GIF 動畫
(https://miro.medium.com/max/963/1*wpbLgTW_lopZ6JtDqVByuA.gif)
- [3] TensorFlow 內建的 Cifar 圖像
(<https://www.tensorflow.org/datasets/catalog/cifar10?hl=zh-tw>)
- [4] TensorFlow ImageDataGenerator 說明
(https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/image/ImageDataGenerator)
- [5] TensorFlow Data Augmentation 說明
(https://www.tensorflow.org/tutorials/images/data_augmentation)
- [6] Keras CNN Dog or Cat Classification 範例
(<https://www.kaggle.com/code/uysimty/keras-cnn-dog-or-cat-classification>)
- [7] Albumentations
(<https://github.com/albumentations-team/albumentations>)
- [8] Jason Brownlee, 《How to Visualize Filters and Feature Maps in Convolutional Neural Networks》, 2019
(<https://machinelearningmastery.com/how-to-visualize-filters-and-feature-maps-in-convolutional-neural-networks/>)
- [9] SHAP GitHub
(<https://github.com/slundberg/shap>)
- [10] 維基百科中關於 Shapley value 的介紹
(https://en.wikipedia.org/wiki/Shapley_value)
- [11] LIME 套件的安裝與介紹說明
(<https://github.com/marcotcr/lime>)
- [12] Bolei Zhou, Aditya Khosla, Agata Lapedriza et al, 《Learning Deep Features for Discriminative Localization》, 2015
(<https://arxiv.org/pdf/1512.04150.pdf>)
- [13] Kaggle 中介紹的實作
(<https://www.kaggle.com/aakashnain/what-does-a-cnn-see>)
- [14] Disadvantages of CNN models
(<https://iq.opengenus.org/disadvantages-of-cnn/>)

[15] Capsule Networks

(<https://cedar.buffalo.edu/~srihari/CSE676/9.12%20CapsuleNets.pdf>)

第 7 章 預先訓練的模型(Pre-trained Model)

- [1] Sergey Ioffe 、Christian Szegedy, 《Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift》, 2015
(<http://proceedings.mlr.press/v37/ioffe15.pdf>)
- [2] Kaiming He 、Xiangyu Zhang 、Shaoqing Ren 、Jian Sun, 《Deep Residual Learning for Image Recognition》, 2015
(<https://arxiv.org/abs/1512.03385>)
- [3] Keras 官網關於 Keras Applications 的介紹
(<https://keras.io/api/applications/>)
- [4] Marie Stephen Leo, 《How to Choose the Best Keras Pre-Trained Model for Image Classification》, 2020
(<https://towardsdatascience.com/how-to-choose-the-best-keras-pre-trained-model-for-image-classification-b850ca4428d4>)
- [5] yrevar GitHub
(<https://gist.github.com/yrevar/942d3a0ac09ec9e5eb3a>)
- [6] Ethan Rosenthal, 《Using Keras' Pretrained Neural Networks for Visual Similarity Recommendations》, 2016
(<https://www.ethanrosenthal.com/2016/12/05/recasketch-keras/>)
- [7] Tensorflow 官網範例 『Load images 』
(https://www.tensorflow.org/tutorials/load_data/images)
- [8] Sergey Ioffe 、Christian Szegedy, 《Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift》, 2015
(<https://arxiv.org/pdf/1502.03167.pdf>)
- [9] Aman Sawarn, Why Batch Normalization Matters?, 2020
(<https://medium.com/towards-artificial-intelligence/why-batch-normalization-matters-4a6d753ba309>)
- [10] alexirpan, On The Perils of Batch Norm, 2017
(<https://www.alexirpan.com/2017/04/26/perils-batch-norm.html>)

第 8 章 物件偵測(Object Detection)

- [1] Joseph Redmon 、Anelia Angelova, 《Real-Time Grasp Detection Using Convolutional Neural Networks》, 2015
(https://docs.google.com/presentation/d/1Zc9-iR1eVz-zysinwb7bzLGC2no2ZiaD897_14dGbhw/edit?usp=sharing)
- [2] 2011 年 ImageNet ILSVRC 挑戰賽比賽說明
(<http://image-net.org/challenges/LSVRC/2011/index>)
- [3] 2017 年 ImageNet ILSVRC 挑戰賽比賽說明
(<http://image-net.org/challenges/LSVRC/2017/>)
- [4] Fei-Fei Li 、Justin Johnson 、Serena Yeung, 《Lecture 11: Detection and Segmentation》, 2017
(http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf)
- [5] Adrian Rosebrock, 《Image Pyramids with Python and OpenCV》, 2015
(<https://www.pyimagesearch.com/2015/03/16/image-pyramids-with-python-and-opencv/>)
- [6] IIP Image
(<https://iipimage.sourceforge.io/documentation/images/>)
- [7] Adrian Rosebrock, 《Sliding Windows for Object Detection with Python and OpenCV》, 2015
(<https://www.pyimagesearch.com/2015/03/23/sliding-windows-for-object-detection-with-python-and-opencv/>)
- [8] CV17 HOG 特徵提取算法
(<https://blog.csdn.net/henghuizan2771/article/details/124186797>)
- [9] Adrian Rosebrock, 《Histogram of Oriented Gradients and Object Detection》, 2014
(<https://www.pyimagesearch.com/2014/11/10/histogram-oriented-gradients-object-detection/>)
- [10] Adrian Rosebrock, 《Non-Maximum Suppression for Object Detection in Python》, 2014
(<https://www.pyimagesearch.com/2014/11/17/non-maximum-suppression-object-detection-python/>)
- [11] Tomasz Malisiewicz, 《Ensemble of Exemplar-SVMs for Object Detection and Beyond》
(<http://www.cs.cmu.edu/~tmalisie/projects/iccv11/index.html>)
- [12] Ross Girshick 、Jeff Donahue 、Trevor Darrell 、Jitendra Malik, 《Rich feature

hierarchies for accurate object detection and semantic segmentation》, 2014
(<https://arxiv.org/pdf/1311.2524.pdf>)

[13] RCNN GitHub
(<https://github.com/1297rohit/RCNN>)

[14] Lung-Ying Ling, R-CNN 學習筆記, 2019
(<https://laptrinhx.com/r-cnn-xue-xi-bi-ji-1145354539/>)

[15] Kaiming He、Xiangyu Zhang、Shaoqing Ren、Jian Sun, 《Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition》, 2015
(<https://arxiv.org/abs/1406.4729>)

[16] v1_vivian, 《SPP-Net 論文詳解》, 2017
(<https://www.itread01.com/content/1542334444.html>)

[17] Ross B. Girshick 於 GitHub 上放置的 Faster R-CNN 程式碼
(<https://github.com/rbgirshick/py-faster-rcnn>)

[18] Getting Started with Detectron2
(https://detectron2.readthedocs.io/en/latest/tutorials/getting_started.html)

[19] Detectron2 範例檔
(https://colab.research.google.com/drive/16jcaJoc6bCFAQ96jDe2HwtXj7BMD_-m5)

[20] Segment Anything Model 2 (SAM 2)
(<https://github.com/facebookresearch/sam2>)

[21] You Only Look Once: Unified, Real-Time Object Detection
(<https://docs.google.com/presentation/d/1kAa7NOamBt4calBU9iHgT8a86RRHz9Yz2oh4-GTdX6M/edit#slide=id.p>)

[22] 人工智慧技術 YOLOv4 出自中研院團隊
(<https://news.ltn.com.tw/news/life/breakingnews/3215570>)

[23] Ultralytics 網站首頁
(<https://docs.ultralytics.com/>)

[24] Ultralytics GitHub
(<https://github.com/ultralytics/ultralytics>)

[25] Joseph Redmon、Santosh Divvala、Ross Girshick、Ali Farhadi, 《You Only Look Once: Unified, Real-Time Object Detection》, 2016
(<https://docs.google.com/presentation/d/1kAa7NOamBt4calBU9iHgT8a86RRHz9Yz2oh4-GTdX6M/edit?usp=sharing>)

[26] PyTorch 官網首頁
(<https://pytorch.org/>)

- [27] Ultralytics 說明文件
(<https://docs.ultralytics.com/quickstart/>)
- [28] YOLO 解決方案
(<https://docs.ultralytics.com/solutions/#solutions>)
- [29] COCO 類別
(<https://github.com/ultralytics/ultralytics/blob/main/ultralytics/cfg/datasets/coco.yaml>)
- [30] YOLO 圖像分類文件說明
(<https://docs.ultralytics.com/tasks/classify/>)
- [31] Cat-or-not GitHub
(<https://github.com/rpeden/cat-or-not/releases>)
- [32] YOLO Object detection 文件說明
(<https://docs.ultralytics.com/tasks/detect/>)
- [33] Object Detection Datasets Overview
(<https://docs.ultralytics.com/datasets/detect/>)
- [34] TensorFlow Object Detection
(https://www.tensorflow.org/hub/tutorials/object_detection)
- [35] Tensorflow Object Detection API 官網文件
(<https://tensorflow-object-detection-api-tutorial.readthedocs.io/en/latest/training.html>)
- [36] YOLO v11 Simple Utilities
(<https://docs.ultralytics.com/usage/simple-utilities/>)
- [37] LabelImg
(<https://github.com/tzutalin/labelImg>)
- [38] Label Studio
(<https://labelstud.io/>)
- [39] YOLO v11 Instance Segmentation
(<https://docs.ultralytics.com/tasks/segment/>)
- [40] Instance Segmentation Datasets Overview
(<https://docs.ultralytics.com/datasets/segment/#supported-datasets>)
- [41] YOLO v11 Pose Estimation
(<https://docs.ultralytics.com/tasks/pose/>)
- [42] Pose Estimation Datasets Overview
(<https://docs.ultralytics.com/datasets/pose/#dataset-yaml-format>)
- [43] Oz Ramos, 《Introducing Handsfree.js - Integrate hand, face, and pose gestures

to your frontend》

(<https://dev.to/midiblocks/introducing-handsfree-js-integrate-hand-face-and-pose-gestures-to-your-frontend-4g3p>)

[44] Jen Looper, 《Ombromanie: Creating Hand Shadow stories with Azure Speech and TensorFlow.js Handposes》

(<https://dev.to/azure/ombromanie-creating-hand-shadow-stories-with-azure-speech-and-tensorflow-js-handposes-3cln>)

[45] YOLO v11 Oriented Bounding Boxes Object Detection

(<https://docs.ultralytics.com/tasks/obb/>)

[46] Oriented Bounding Box (OBB) Datasets Overview

(<https://docs.ultralytics.com/datasets/obb/>)

[47] 清大獲得 2018 NVIDIA 機器人冠軍

(<https://www.youtube.com/watch?v=OqdnG4AII8>)

[48] YOLO v11, Multi-Object Tracking with Ultralytics YOLO

(<https://docs.ultralytics.com/modes/track/>)

[49] BoT-SORT GitHub

(<https://github.com/NirAharon/BoT-SORT>)

[50] ByteTrack GitHub

(<https://github.com/ifzhang/ByteTrack>)

[51] 多目標跟蹤的性能評估：指標與計算方法

(<https://cloud.baidu.com/article/3271389>)

[52] Multi-Object Tracking with Ultralytics YOLO

(<https://docs.ultralytics.com/modes/track/#available-trackers>)

[53] Ultralytics GitHub

(<https://github.com/ultralytics/ultralytics/tree/main/ultralytics/cfg/trackers>)

[54] Ultralytics Solutions: Harness YOLO11 to Solve Real-World Problems

(<https://docs.ultralytics.com/solutions/>)

[55] YOLO 各種任務的資料集列表

(<https://docs.ultralytics.com/datasets/#ultralytics-explorer>)

第 9 章 生成式 AI(Generative AI)

[1] An AI-Generated Painting Wins Art Contest, Annoys Artists
(<https://shakiroslann.medium.com/an-ai-generated-painting-wins-art-contest-annoys-artists-f3f7c98e3088>)

[2] Sony 世界攝影大獎優勝作品「竟是 AI 畫的」
(<https://news.ltn.com.tw/news/world/breakingnews/4271589>)

[3] Vijay Badrinarayanan 、 Alex Kendall 、 Roberto Cipolla, 《SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation》, 2015
(<https://arxiv.org/abs/1511.00561>)

[4] PyImageSearch, Introduction to Autoencoders
(<https://pyimagesearch.com/2023/07/10/introduction-to-autoencoders/>)

[5] Convolutional Variational Autoencoder(CAE)
(<https://www.tensorflow.org/tutorials/generative/cvae>)

[6] mingukkang, CVAE
(<https://github.com/mingukkang/CVAE>)

[7] gozsoy, conditional-vae
(<https://github.com/mingukkang/CVAE>)

[8] PyImageSearch, U-Net Image Segmentation in Keras
(<https://pyimagesearch.com/2022/02/21/u-net-image-segmentation-in-keras/>)

[9] TensorFlow 官網提供的範例『Image segmentation』
(<https://www.tensorflow.org/tutorials/images/segmentation>)

[10] Oxford 大學資訊工程學系網址
(<https://www.robots.ox.ac.uk/>)

[11] Nahian Siddique et al., U-Net and its variants for medical image segmentation: theory and applications
(<https://arxiv.org/pdf/2011.01118>)

[12] Walid Ehab a Yongmin Li a, Performance Analysis of UNet and Variants for Medical Image Segmentation
(<https://arxiv.org/pdf/2309.13013>)

[13] fast-style-transfer GitHub
(<https://github.com/lengstrom/fast-style-transfer>)

[14] 翁書婷,《催生全球首位 AI 繪師 Andy，美圖搶攻人工智慧卻面臨一大挑戰》
(<https://www.bnext.com.tw/article/47330/ai-andy-meitu>)

[15] Leon A. Gatys 、 Alexander S. Ecker 、 Matthias Bethge, 《A Neural Algorithm of Artistic Style》

(<https://arxiv.org/abs/1508.06576>)

[16] TensorFlow 官網提供的範例『Neural Style Transfer』

(https://www.tensorflow.org/tutorials/generative/style_transfer)

[17] TensorFlow 官網提供的範例『Fast Style Transfer for Arbitrary Styles』

(https://www.tensorflow.org/hub/tutorials/tf2_arbitrary_image_stylization)

[18] Attention Is All You Need

(<https://arxiv.org/pdf/1706.03762>)

[19] Diffusion Models: A Comprehensive Survey of Methods and Applications

(<https://arxiv.org/pdf/2209.00796>)

第 10 章 生成對抗網路 (GAN)

- [1] 自由時報,《全球首次! AI 創作肖像畫 10 月佳士得拍賣》, 2018
(<https://news.ltn.com.tw/news/world/breakingnews/2529174>)
- [2] 佳士得官網《Is artificial intelligence set to become art's next medium?》
(<https://www.christies.com/features/A-collaboration-between-two-artists-one-human-one-a-machine-9332-1.aspx>)
- [3] 佳士得官網關於 Edmond de Belamy 肖像畫的介紹
(<https://www.christies.com/lot/lot-edmond-de-belamy-from-la-famille-de-6166184>)
- [4] the-gan-zoo GitHub
(<https://github.com/hindupuravinash/the-gan-zoo>)
- [5] Liqian Ma、Xu Jia、Qianru Sun 等人,《Pose Guided Person Image Generation》, 2018
(<https://arxiv.org/pdf/1705.09368.pdf>)
- [6] Yanghua Jin、Jiakai Zhang 等人,《Towards the Automatic Anime Characters Creation with Generative Adversarial Networks》, 2017
(<https://arxiv.org/pdf/1708.05509.pdf>)
- [7] MakeGirlsMoe 網站
(<https://make.girls.moe/#/>)
- [8] MIL WebDNN 網站
(<https://mil-tokyo.github.io/webdnn/>)
- [9] Yanghua Jin、Jiakai Zhang 等人,《Towards the Automatic Anime Characters Creation with Generative Adversarial Networks》, 2017
(<https://arxiv.org/pdf/1708.05509.pdf>)
- [10] StarGAN GitHub
(<https://github.com/yunjey/stargan>)
- [11] Christian Ledig、Lucas Theis、Ferenc Huszár 等人,《Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network》, 2017
(<https://arxiv.org/pdf/1609.04802.pdf>)
- [12] Tero Karras、Samuli Laine、Miika Aittala 等人,《Analyzing and Improving the Image Quality of StyleGAN》, 2020
(<https://arxiv.org/pdf/1912.04958.pdf>)
- [13] StyleGAN2 GitHub
(<https://github.com/NVlabs/stylegan2>)
- [14] Jonathan Hui,《GAN — Some cool applications of GAN》, 2018

(<https://jonathan-hui.medium.com/gan-some-cool-applications-of-gans-4c9ecca35900>)

[15] Deep Convolutional Generative Adversarial Network
(<https://www.tensorflow.org/tutorials/generative/dcgan>)

[16] Keras 官網範例『DCGAN to generate face images』
(https://keras.io/examples/generative/dcgan_overriding_train_step/)

[17] Tero Karras、Timo Aila、Samuli Laine 等人,《Progressive Growing of GANs for Improved Quality, Stability, and Variation》, 2017
(<https://arxiv.org/abs/1710.10196>)

[18] Mehdi Mirza、Simon Osindero,《Conditional Generative Adversarial Nets》, 2014
(<https://arxiv.org/abs/1411.1784>)

[19] Qiwen Fu、Wei-Ting Hsu、Mu-Heng Yang,《Colorization Using ConvNet and GAN》, 2017
(<http://cs231n.stanford.edu/reports/2017/pdfs/302.pdf>)

[20] ColorGAN GitHub
(<https://github.com/bbc/ColorGAN#end-to-end-conditional-gan-based-architectures-for-image-colourisation>)

[21] Phillip Isola、Jun-Yan Zhu、Tinghui Zhou,《Image-to-Image Translation with Conditional Adversarial Networks》, 2016
(<https://arxiv.org/abs/1611.07004>)

[22] CMP Facade Database
(<https://cmp.felk.cvut.cz/~tylecr1/facade/>)

[23] pix2pix datasets
(<https://efrosrans.eecs.berkeley.edu/pix2pix/datasets/>)

[24] Jun-Yan Zhu、Taesung Park、Phillip Isola、Alexei A. Efros,《Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks》, 2017
(<https://arxiv.org/abs/1703.10593>)

[25] Tensorflow 官網有關 CycleGAN 的說明
(https://www.tensorflow.org/datasets/catalog/cycle_gan)

[26] CartoonGAN: Generative Adversarial Networks for Photo Cartoonization
(https://openaccess.thecvf.com/content_cvpr_2018/papers/Chen_CartoonGAN_Generative_Adversarial_CVPR_2018_paper.pdf)

[27] 用 CartoonGAN 及 TensorFlow 2 生成新海誠與宮崎駿動畫
(<https://leemeng.tw/generate-anime-using-cartoongan-and-tensorflow2.html>)

- [28] mnicnc404 CartoonGan-tensorflow GitHub
(<https://github.com/mnicnc404/CartoonGan-tensorflow>)
- [29] SystemErrorWang FacialCartoonization
(<https://github.com/SystemErrorWang/FacialCartoonization>)
- [30] 李宏毅老師的 PPT 『Introduction of Generative Adversarial Network (GAN)』
(https://speech.ee.ntu.edu.tw/~tlkagk/slide/Tutorial_HYLee_GAN.pdf)
- [31] Shakir Mohamed、Danilo Rezende, 《Tutorial on Deep Generative Models》, 2017
(<http://www.shakirm.com/slides/DeepGenModelsTutorial.pdf>)
- [32] You Won't Believe What Obama Says In This Video!
(<https://www.youtube.com/watch?v=cQ54GDm1eL0>)
- [33] Alan Zucconi, 《Understanding the Technology Behind DeepFakes》, 2018
(<https://www.alanzucconi.com/2018/03/14/understanding-the-technology-behind-deepfakes/>)
- [34] Aayush Bansal、Shugao Ma、Deva Ramanan、Yaser Sheikh, 《Recycle-GAN: Unsupervised Video Retargeting》, 2018
(<https://arxiv.org/abs/1808.05174>)
- [35] Jonathan Hui, 《Detect AI-generated Images & Deepfakes》, 2020
(<https://jonathan-hui.medium.com/detect-ai-generated-images-deepfakes-part-1-b518ed5075f4>)
- [36] Louis (What's AI) Bouchard, 《DeepFakes in 5 minutes》, 2020
(<https://pub.towardsai.net/deepfakes-in-5-minutes-155c13d48fa3>)
- [37] DeepFaceLab GitHub
(<https://github.com/iperov/DeepFaceLab>)
- [38] Jonathan Hui, 《Detect AI-generated Images & Deepfakes (Part 1)》, 2020
(<https://jonathan-hui.medium.com/detect-ai-generated-images-deepfakes-part-1-b518ed5075f4>)
- [39] 林妍臻, 《微軟開發能判別 Deepfake 影像及內容變造的技術》, 2020
(<https://www.ithome.com.tw/news/139740>)

第 11 章 擴散模型(Diffusion Model)

- [1] What Is Diffusion in Chemistry?
(<https://www.thoughtco.com/definition-of-diffusion-604430>)
- [2] 實作理解 Diffusion Model:來自 DDPM 的簡化概念
(<https://medium.com/ai-blog-tw/邊實作邊學習-diffusion-model-從-ddpm-的簡化概念理解-4c565a1c09c>)
- [3] Denoising Diffusion Probabilistic Models
(<https://arxiv.org/pdf/2006.11239>)
- [4] Lilian Weng, What are Diffusion Models?
(<https://arxiv.org/pdf/2006.11239>)
- [5] Alec Radford et al., Learning Transferable Visual Models From Natural Language Supervision
(<https://arxiv.org/pdf/2006.11239>)
- [6] Stable Diffusion Wikiperia
(https://en.wikipedia.org/wiki/Stable_Diffusion)
- [7] J. Rafid Siddiqui, What are Stable Diffusion Models and Why are they a Step Forward for Image Generation?
(<https://towardsdatascience.com/what-are-stable-diffusion-models-and-why-are-they-a-step-forward-for-image-generation-aa1182801d46>)
- [8] Robin Rombach et al., High-Resolution Image Synthesis with Latent Diffusion Models
(<https://arxiv.org/pdf/2112.10752>)
- [9] Lilian Weng, What are Diffusion Models?
(<https://arxiv.org/pdf/2006.11239>)
- [10] Jonathan Ho et al., Denoising Diffusion Probabilistic Models (DDPM)
(<https://arxiv.org/pdf/2006.11239>)
- [11] Jiaming Song et al., Denoising Diffusion Implicit Models (DDIM)
(<https://arxiv.org/pdf/2010.02502>)
- [12] Alex Nichol/Prafulla Dhariwal, Improved Denoising Diffusion Probabilistic Models
(<https://arxiv.org/pdf/2102.09672>)
- [13] Jonathan Kernes, Diffusion Models
(<https://towardsdatascience.com/diffusion-models-91b75430ec2>)
- [14] SingleZombie/DL-Demo

(<https://github.com/SingleZombie/DL-Demos/tree/master>)

[15] MidJourney Plan Comparison
(<https://docs.midjourney.com/docs/plans>)

[16] MidJourney Parameter List
(<https://docs.midjourney.com/docs/parameter-list>)

[17] Stable Diffusion 維基百科
(<https://docs.midjourney.com/docs/parameter-list>)

[18] DALL·E: Introducing outpainting
(<https://openai.com/index/dall-e-introducing-outpainting/>)

[19] ControlNet GitHub
(<https://github.com/lllyasviel/ControlNet>)

[20] Stable Diffusion prompt: a definitive guide
(<https://stable-diffusion-art.com/prompt-guide/>)

[21] Hugging Face Sign Up
(<https://huggingface.co/>)

[22] Hugging Face Sign Up
(<https://huggingface.co/>)

[23] Hugging Face Stable Diffusion 3 Pipeline
(https://huggingface.co/docs/diffusers/main/en/api/pipelines/stable_diffusion/stable_diffusion_3)

[24] Awesome LLM Resources List
(<https://github.com/ilsilfverskiold/Awesome-LLM-Resources-List/blob/main/img-gen/README.md#generative-image-tools>)

[25] AUTOMATIC1111 Stable Diffusion Web UI
(<https://github.com/AUTOMATIC1111/stable-diffusion-webui>)

[26] AUTOMATIC1111 Stable Diffusion Web UI 發行網址
(<https://github.com/AUTOMATIC1111/stable-diffusion-webui/releases/tag/v1.0.0-pre>)

[27] AUTOMATIC1111 Stable Diffusion Web UI wiki
(<https://github.com/AUTOMATIC1111/stable-diffusion-webui/wiki>)

[28] sd_xl_refiner_1.0_0.9vae 超連結
(https://huggingface.co/stabilityai/stable-diffusion-xl-refiner-1.0/blob/main/sd_xl_refiner_1.0_0.9vae.safetensors)

[29] w-e-w GitHub

(<https://gist.github.com/w-e-w/0f37c04c18e14e4ee1482df5c4eb9f53>)

[30] Adding Conditional Control to Text-to-Image Diffusion Models
(<https://arxiv.org/abs/2302.05543>)

[31] sd-webui-controlnet GitHub
(<https://github.com/Mikubill/sd-webui-controlnet>)

[32] Stable Diffusion 1.5 / 2.0 中型模型
(https://huggingface.co/comfyanonymous/ControlNet-v1-1_fp16_safetensors/tree/main)

[33] llyasviel/ControlNet GitHub
(<https://github.com/llyasviel/ControlNet>)

[34] llyasviel/ControlNet v1.1 GitHub
(<https://github.com/llyasviel/ControlNet-v1-1-nightly>)

[35] HuggingFace llyasviel/ControlNet-v1-1
(<https://huggingface.co/llyasviel/ControlNet-v1-1/tree/main>)

[36] HuggingFace stable-diffusion-v1-5/stable-diffusion-v1-5
(<https://huggingface.co/stable-diffusion-v1-5/stable-diffusion-v1-5/blob/main/v1-5-pruned.ckpt>)

[37] Ultra fast ControlNet with Diffusers
(<https://huggingface.co/blog/controlnet>)

[38] NitroFusion, Hugging Face
(<https://huggingface.co/ChenDY/NitroFusion>)

[39] Dar-Yen Chen et al., NitroFusion: High-Fidelity Single-Step Diffusion through Dynamic Adversarial Training
(<https://arxiv.org/pdf/2412.02030>)

[40] OpenAI Image generation
(<https://platform.openai.com/docs/guides/images>)

[41] OpenAI's New Sora Model Is Amazing
(<https://medium.com/coding-beauty/new-openai-sora-model-56e399cd259f>)

[42] OpenAI Sora
(<https://openai.com/index/sora/>)

第 12 章 其他影像應用

- [1] Kaipeng Zhang、Zhanpeng Zhang、Zhifeng Li、Yu Qiao, 《Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks》, 2016
(<https://arxiv.org/abs/1604.02878>)
- [2] Face Recognition GitHub 的範例
(https://github.com/ageitgey/face_recognition)
- [3] Shaoqing Ren、Xudong Cao、Yichen Wei 等人, 《Face Alignment at 3000 FPS via Regressing Local Binary Features》, 2014
(http://www.jiansun.org/papers/CVPR14_FaceAlignment.pdf)
- [4] Georgios Tzimiropoulos、Maja Pantic, 《Optimization problems for fast AAM fitting in-the-wild》, 2013
(https://ibug.doc.ic.ac.uk/media/uploads/documents/tzimirop_pantic_iccv2013.pdf)
- [5] V.Kazemi、J. Sullivan, 《One Millisecond Face Alignment with an Ensemble of Regression Trees》, 2014
(http://www.csc.kth.se/~vahidk/face_ert.html)
- [6] Tesseract OCR GitHub
(<https://github.com/UB-Mannheim/tesseract/wiki>)
- [7] Tesseract OCR 使用說明
(<https://github.com/tesseract-ocr/tesseract/blob/master/doc/tesseract.1.asc>)
- [8] Filip Zelic、Anuj Sable, 《A comprehensive guide to OCR with Tesseract, OpenCV and Python》, 2021
(<https://nanonets.com/blog/ocr-with-tesseract/>)
- [9] Tesseract 官網的語言列表
(<https://github.com/tesseract-ocr/tesseract/blob/master/doc/tesseract.1.asc#LANGUAGES>)
- [10] EasyOCR GitHub
(<https://github.com/jwstckhr/python-ocr-sample>)
- [11] Youngmin Baek et al., Character Region Awareness for Text Detection
(<https://arxiv.org/pdf/1904.01941>)
- [12] keras-ocr 套件
(<https://github.com/faustomorales/keras-ocr>)
- [13] imanoop7/Ollama-OCR GitHub
(<https://github.com/imanoop7/Ollama-OCR>)

- [14] Aswinth Raj, 《Car License Plate Recognition using Raspberry Pi and OpenCV》, 2019
(<https://circuitdigest.com/microcontroller-projects/license-plate-recognition-using-raspberry-pi-and-opencv>)
- [15] RembgGitHub
(<https://github.com/danielgatis/rembg>)
- [16] Xuebin Qin et al., U^2 -Net: Going Deeper with Nested U-Structure for Salient Object Detection
(<https://arxiv.org/pdf/2005.09007>)
- [17] Xuebin Qin et al., U^2 -Net GitHub
(<https://github.com/xuebinqin/U-2-Net>)
- [18] CUDA Toolkit Downloads
(<https://developer.nvidia.com/cuda-downloads>)
- [19] CuDNN Downloads
(<https://developer.nvidia.com/cudnn>)
- [20] Rembg 底層支援各種的 U^2 -Net 模型
(<https://github.com/danielgatis/rembg?tab=readme-ov-file#models>)
- [21] U^2 -Net GitHub
(<https://github.com/xuebinqin/U-2-Net>)
- [22] Keras Code Examples
(<https://keras.io/examples/>)

第 13 章 自然語言處理的介紹

- [1] Sebastian Andrei, 《South Korea's Convenience Store Culture》, 2018
(https://medium.com/@sebastian_andrei/south-koreas-convenience-store-culture-187c33a649a6)
- [2] 維基百科關於 tf-idf 的說明
(<https://en.wikipedia.org/wiki/Tf%E2%80%93idf>)
- [3] Tomas Mikolov、Quoc V. Le、Ilya Sutskever, 《Exploiting Similarities among Languages for Machine Translation》, 2013
(<https://arxiv.org/pdf/1309.4168v1.pdf>)
- [4] NSS, 《An Intuitive Understanding of Word Embeddings: From Count Vectors to Word2Vec》, 2017
(<https://www.analyticsvidhya.com/blog/2017/06/word-embeddings-count-word2veec/>)
- [5] Ria Kulshrestha, 《NLP 102: Negative Sampling and GloVe》, 2019
(<https://towardsdatascience.com/nlp-101-negative-sampling-and-glove-936c88f3bc68>)
- [6] Gensim Word2Vec
(<https://radimrehurek.com/gensim/models/word2vec.html>)
- [7] Gensim Word2Vec Tutorial 』 (<https://www.kaggle.com/pierremegret/gensim-word2vec-tutorial>)
- [8] TensorFlow Skip-gram 模型
(<https://www.tensorflow.org/tutorials/text/word2vec>)
- [9] OpenAI Vector embeddings
(<https://platform.openai.com/docs/guides/embeddings>)
- [10] TensorFlow Embedding Projector
(<https://projector.tensorflow.org/>)
- [11] Jeffrey Pennington、Richard Socher、Christopher D. Manning, 《GloVe: Global Vectors for Word Representation》, 2014
(<https://www.aclweb.org/anthology/D14-1162.pdf>)
- [12] 史丹佛大學網站
(<https://nlp.stanford.edu/data/wordvecs/glove.42B.300d.zip>)
- [13] APCLab GitHub
(<https://github.com/APCLab/jieba-tw/tree/master/jieba>)

- [14] 自由時報 蘇金鳳,《中市明第二輪分區限水 百貨業買 20 個水塔桶》, 2021
(<https://news.ltn.com.tw/news/life/breakingnews/3497315>)
- [15] 布丁布丁吃布丁,《彙整中文與英文的詞性標註代號》, 2017
(<http://blog.pulipuli.info/2017/11/fasttag-identify-part-of-speech-in.html>)
- [16] spaCy 安裝網頁
(<https://spacy.io/usage>)
- [17] spaCy Quickstart 網頁
(<https://spacy.io/usage/models>)
- [18] pkuseg GitHub
(<https://github.com/explosion/spacy-pkuseg>)
- [19] spaCy 101 官網『spaCy 101: Everything you need to know』
(<https://spacy.io/usage/spacy-101>)
- [20] spaCy Token
(<https://spacy.io/api/token>)
- [21] spaCy GitHub 的 glossary.py
(<https://github.com/explosion/spaCy/blob/master/spacy/glossary.py>)
- [22] displaCy visualizer 說明文件
(<https://spacy.io/api/top-level#displacy>)
- [23] sense2vec
(<https://github.com/explosion/sense2vec>)
- [24] sense2vec reloaded: contextually-keyed word vectors
(<https://explosion.ai/blog/sense2vec-reloaded>)
- [25] Mastering spaCy
(<https://www.packtpub.com/en-us/product/mastering-spacy-9781800563353>)

第 14 章 自然語言處理的演算法

- [1] Keras Embedding layer
(https://keras.io/api/layers/core_layers/embedding/)
- [2] Christopher Olah, 《Understanding LSTM Networks》, 2015
(<https://colah.github.io/posts/2015-08-Understanding-LSTMs/>)
- [3] Kaggle IMDB 資料集
(<https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews>)
- [4] 史丹佛大學網站
(https://ai.stanford.edu/~amaas/data/sentiment/aclImdb_v1.tar.gz)
- [5] VADER (Valence Aware Dictionary and sEntiment Reasoner)
(<https://github.com/cjhutto/vaderSentiment>)
- [6] Keras 官網 LSTM 的說明
(https://keras.io/api/layers/recurrent_layers/lstm/)
- [7] Jason Brownlee, 《Time Series Prediction with LSTM Recurrent Neural Networks in Python with Keras》, 2016
(<https://machinelearningmastery.com/time-series-prediction-lstm-recurrent-neural-networks-python-keras>)
- [8] Junyoung Chung、Caglar Gulcehre、KyungHyun Cho、Yoshua Bengio,
《Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling》, 2014
(<https://arxiv.org/abs/1412.3555>)
- [9] Michael Phi, 《Illustrated Guide to LSTM's and GRU's: A step by step explanation》, 2018
(<https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21>)
- [10] Andrej Karpathy, 《The Unreasonable Effectiveness of Recurrent Neural Networks》, 2015
(<http://karpathy.github.io/2015/05/21/rnn-effectiveness/>)
- [11] Alexandre Xavier, 《Predicting stock prices with LSTM》, 2019
(<https://medium.com/neuronio/predicting-stock-prices-with-lstm-349f5a0974d4>)
- [12] 亞馬遜企業股票行情(<https://www.kaggle.com/szrlee/stock-time-series-20050101-to-20171231>)
- [13] TA-Lib GitHub

(<https://github.com/TA-Lib/ta-lib-python>)

[14] TA-Lib GitHub

(<https://github.com/TA-Lib/ta-lib-python>)

[15] Prophet 文件說明

(https://facebook.github.io/prophet/docs/quick_start.html)

[16] 陳昭明,《演算法交易(Algorithmic Trading) 實作》, 2021

(<https://ithelp.ithome.com.tw/articles/10255111>)

[17] 維基百科 Augmented Dickey–Fuller test

(https://en.wikipedia.org/wiki/Augmented_Dickey-Fuller_test)

[18] ChatGPT 範例

(<https://platform.openai.com/docs/examples>)

[19] Lilian Weng, Attention? Attention!

(<https://lilianweng.github.io/posts/2018-06-24-attention/>)

[20] Neural machine translation with attention

(https://www.tensorflow.org/text/tutorials/nmt_with_attention)

[21] Named Entity Recognition using Transformers

(https://keras.io/examples/nlp/ner_transformers/)

[22] spaCy Training Pipelines & Models

(<https://spacy.io/usage/training>)

[23] 善用 ChatGPT 幫我們訓練出自訂的 Named Entity Recognition 實體

(<https://medium.com/willhanchen/自然語言處理-spacy-善用-chatgpt 幫我們訓練出自訂的 named-entity-recognition 實體-2450df2127cc>)

[24] Text generation with an RNN

(https://www.tensorflow.org/text/tutorials/text_generation)

第 15 章 大型語言模型(LLM)

[1] Harnessing the Power of LLMs in Practice : A Survey on ChatGPT and Beyond
(<https://arxiv.org/pdf/2304.13712.pdf>)

[2] ChatGPT
(<https://chatgpt.com/>)

[3] Llama
(<https://www.llama.com/>)

[4] Gemini
(<https://gemini.google.com/>)

[5] Gemma
(<https://ai.google.dev/gemma>)

[6] Claude
(<https://claude.ai/>)

[7] TAIDE
(<https://taide.tw/index>)

[8] HuggingFace TAIDE models
(<https://github.com/MiuLab/Taiwan-LLM>)

[9] Project TAME
(<https://taide.tw/index>)

[10] MR Models
(<https://github.com/mtkresearch/MR-Models>)

[11] Don't use DeepSeek-v3!
(<https://medium.com/data-science-in-your-pocket/dont-use-deepseek-v3-895be7b853b0>)

[12] Ashish Vaswani 、Noam Shazeer 、Niki Parmar, 《Attention Is All You Need》 , 2017
(<https://arxiv.org/pdf/1706.03762.pdf>)

[13] Etched 官網
(<https://www.etched.com/announcing-etched>)

[14] 關於機器翻譯，看這一篇就夠了
(<https://www.infoq.cn/article/wboqeghlpvkzhatdyvwf>)

[15] Neural machine translation with a Transformer and Keras
(<https://www.tensorflow.org/text/tutorials/transformer>)

- [16] Positional Encoding Explained: A Deep Dive into Transformer PE
(<https://medium.com/thedeephub/positional-encoding-explained-a-deep-dive-into-transformer-pe-65cfe8cfe10b>)
- [17] Is GPT group of models decoder only model
(<https://community.openai.com/t/is-gpt-group-of-models-decoder-only-model/286586>)
- [18] Hugging Face Transformers Quick Tour
(<https://huggingface.co/docs/transformers/quicktour>)
- [19] ChatGPT Architecture Explained
(<https://medium.com/@sreedevr/chatgpt-architecture-explained-7fc0cbb7426c>)
- [20] Text classification with Transformer
(https://keras.io/examples/nlp/text_classification_with_transformer/)
- [21] Transformers 官網文件
(<https://huggingface.co/docs/transformers/index>)
- [22] Transformers Quick tour
(<https://huggingface.co/transformers/quicktour.html>)
- [23] Pipelines 文件說明
(<https://huggingface.co/transformers/quicktour.html>)
- [24] Using tokenizers from Tokenizers
(https://huggingface.co/transformers/fast_tokenizers.html)
- [25] Transformers 官網『Summary of the tasks』的 Masked Language Modeling
(https://huggingface.co/transformers/task_summary.html#masked-language-modeling)
- [26] Adam Roberts、Staff Software Engineer、Colin Raffel 等人,《Exploring Transfer Learning with T5: the Text-To-Text Transfer Transformer》, 2020
(<https://ai.googleblog.com/2020/02/exploring-transfer-learning-with-t5.html>)
- [27] Transformers Model 頁面(<https://huggingface.co/models>)
- [28] Transformers Fine-tune a pretrained model
(<https://huggingface.co/transformers/training.html#tensorflow>)
- [29] Transformers Trainer
(https://huggingface.co/transformers/main_classes/trainer.html#transformers.TrainingArguments)
- [30] Using DistilBERT for Resource-Efficient Natural Language Processing
(<https://www.kdnuggets.com/distilbert-resource-efficient-natural-language-processing>)

- [31] Llama 模型
(<https://www.llama.com/>)
- [32] Llama 3.2 3B 模型說明頁面
(<https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct>)
- [33] TAIDE 模型說明頁面
(<https://huggingface.co/taide>)
- [34] ChatGPT Prompt examples
(<https://platform.openai.com/docs/examples>)
- [35] OpenAI Function calling 說明
(<https://platform.openai.com/docs/guides/function-calling>)
- [36] ChatGPT Prompt examples
(<https://platform.openai.com/docs/examples>)
- [37] 『AI 很會說話也很會亂講話！從台北捷運公關危機，看智能客服的挑戰』
(<https://www.managertoday.com.tw/columns/view/69785>)
- [38] OpenAI Billing 頁面
(<https://platform.openai.com/settings/organization/billing/overview>)
- [39] OpenAI API Keys 頁面
(<https://platform.openai.com/settings/organization/api-keys>)
- [40] OpenAI Pricing 頁面
(<https://platform.openai.com/settings/organization/api-keys>)
- [41] OpenAI 參考手冊
(<https://platform.openai.com/docs/api-reference/audio/createTranslation>)
- [42] ReAct: Integrating Reasoning and Acting with Retrieval-Augmented Generation
(<https://bluetickconsultants.medium.com/react-integrating-reasoning-and-acting-with-retrieval-augmented-generation-rag-a6c2e869f763>)
- [43] ChatGPT Prompt generation
(<https://platform.openai.com/docs/guides/prompt-generation#meta-schemas>)
- [44] DSPy 套件
(<https://github.com/stanfordnlp/dspy>)
- [45] Training language models to follow instructions with human feedback
(<https://arxiv.org/pdf/2203.02155>)
- [46] Aligning language models to follow instructions

(<https://openai.com/index/instruction-following/>)

[47] ChatGPT Prompt engineering 官方指引

(<https://platform.openai.com/docs/guides/prompt-engineering>)

[48] Chain-of-Thought Prompting Elicits Reasoning in Large Language Models

(<https://arxiv.org/pdf/2201.11903>)

[49] Retrieval-Augmented Generation for Large Language Models: A Survey

(<https://arxiv.org/pdf/2312.10997>)

[50] LangChain 框架

(<https://python.langchain.com/docs/introduction/>)

[51] LangChain Document loaders 文件說明

(https://python.langchain.com/docs/how_to/#document-loaders)

[52] 『How to create and query vector stores』說明

(https://python.langchain.com/docs/how_to/vectorstores/#get-started)

[53] LangChain Templates

(<https://python.langchain.com/v0.1/docs/templates/>)

[54] KAG: A Better Alternative to RAG for Domain-Specific Knowledge Applications

(<https://ai.plainenglish.io/kag-a-better-alternative-to-rag-for-domain-specific-knowledge-applications-046054bedde1>)

[55] 維基百科 Intelligent agent 定義

(https://en.wikipedia.org/wiki/Intelligent_agent)

[56] LangChain Agent

(<https://www.langchain.com/agents>)

[57] CrewAI Multi Agent

(<https://www.crewai.com/>)

[58] Microsoft Semantic Kernel

(<https://learn.microsoft.com/en-us/semantic-kernel/overview/>)

[59] OpenAI Swarm

(<https://github.com/openai/swarm>)

[60] Vertex AI Agent Builder

(<https://cloud.google.com/products/agent-builder>)

[61] 維基百科 Intelligent agent 定義

(https://en.wikipedia.org/wiki/Intelligent_agent)

[62] ReAct: Synergizing Reasoning and Acting in Language Models
(<https://arxiv.org/pdf/2210.03629>)

[63] ReAct Agent Model
(<https://klu.ai/glossary/react-agent-model>)

[64] NVIDIA Puts Grace Blackwell on Every Desk and at Every AI Developer's Fingertips
(<https://nvidianews.nvidia.com/news/nvidia-puts-grace-blackwell-on-every-desk-and-at-every-ai-developers-fingertips>)

第 16 章 語音辨識

- [1] 台大 MIR 實驗室簡介
(http://mirlab.org/about_intro.asp)
- [2] You Won't Believe What Obama Says In This Video!
(<https://www.youtube.com/watch?v=cQ54GDm1eL0>)
- [3] Michael Picheny、Bhuvana Ramabhadran、Stanley F. Chen, 《Lecture 1 Introduction/Signal Processing, Part I》, 2012
(<https://www.ee.columbia.edu/~stanchen/fall12/e6870/slides/lecture1.pdf>)
- [4] Roger Jang (張智星), 《Audio Signal Processing and Recognition (音訊處理與辨識)》, 2005
(<http://mirlab.org/jang/books/audioSignalProcessing/audioIntro.asp?language=chinese>)
- [5] 國立臺灣大學普通物理實驗室官網關於示波器使用教學
(<https://web.phys.ntu.edu.tw/gphyslab/modules/tinyd2/index8803.html?id=7>)
- [6] 別只知道「分貝」！與聲音相關的各種音量單位
(<https://ysolife.com/recording-db-noise-levels/>)
- [7] Pema Grg, 《Audio Signal Processing》, 2020
(<https://blog.ekbana.com/audio-signal-processing-f7e86d415489>)
- [8] 維基百科『相位』
(<https://zh.wikipedia.org/zh-tw/相位>)
- [9] 南臺科技大學商管學院
(https://faculty.stust.edu.tw/~ccchang/vibration/first/second/a/a_4.htm)
- [10] 維基百科 Sampling (signal processing)
([https://en.wikipedia.org/wiki/Sampling_\(signal_processing\)#Sampling_rate](https://en.wikipedia.org/wiki/Sampling_(signal_processing)#Sampling_rate))
- [11] File:CPT-Sound-ADC-DAC.svg Wikimedia Commons
(<https://commons.wikimedia.org/wiki/File:CPT-Sound-ADC-DAC.svg>)
- [12] Muhammad Asif Javed 提供的樣本檔
(<https://github.com/maxifjaved/sample-files>)
- [13] wave 說明文件
(<https://docs.python.org/3/library/wave.html>)
- [14] Download languages and voices for Immersive Reader, Read Mode, and Read Aloud
(<https://support.microsoft.com/en-us/topic/download-languages-and-voices-for->

[immersive-reader-read-mode-and-read-aloud-4c83a8d8-7486-42f7-8e46-2b0fdf753130](https://arxiv.org/abs/1804.03209))

[15] SpeechRecognition 文件說明
(<https://pypi.org/project/SpeechRecognition/>)

[16] Librosa 文件說明
(<https://librosa.org/doc/latest/tutorial.html>)

[17] Vincent Koops, 《Introduction Basic Audio Feature Extraction》, 2017
(http://www.cs.uu.nl/docs/vakken/msmt/lectures/SMT_B_Lecture5_DSP_2017.pdf)

[18] Nagesh Singh Chauhan, 《Audio Data Analysis Using Deep Learning with Python (Part 1)》, 2020
(<https://www.kdnuggets.com/2020/02/audio-data-analysis-deep-learning-python-part-1.html>)

[19] Henry Haefliger, 《Python audio spectrum analyzer》, 2019
(<https://medium.com/quick-code/python-audio-spectrum-analyser-6a3c54ad950>)

[20] ffmpeg 工具程式下載
(<http://ffmpeg.org/download.html>)

[21] ffmpeg 官網文件說明
(<http://ffmpeg.org/documentation.html>)

[22] TensorFlow Audio Datasets
(<https://www.tensorflow.org/datasets/catalog/overview#audio>)

[23] torchaudio Datasets
(<https://pytorch.org/audio/stable/datasets.html>)

[1] Common Voice Datasets
(<https://commonvoice.mozilla.org/en/datasets>)

[24] Common Voice 正體中文
(<https://commonvoice.mozilla.org/zh-TW>)

[25] Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition
(<https://arxiv.org/abs/1804.03209>)

[26] GTZAN Genre Collection
(<http://marsyas.info/downloads/datasets.html>)

[27] Open Speech and Language Resources
(<https://www.openslr.org/1>)

[28] The LJ Speech Dataset
(<https://keithito.com/LJ-Speech-Dataset/>)

- [29] Pyroomacoustics CMU ARCTIC Corpus
(https://pyroomacoustics.readthedocs.io/en/pypi-release/pyroomacoustics.datasets.cmu_arctic.html)
- [30] 維基百科 CMU Pronouncing Dictionary
(https://en.wikipedia.org/wiki/CMU_Pronouncing_Dictionary)
- [31] TensorFlow Simple audio recognition: Recognizing keywords
(https://www.tensorflow.org/tutorials/audio/simple_audio)
- [32] Kunal Vaidya, 《Music Genre Recognition using Convolutional Neural Networks (CNN) — Part 1》, 2020
(<https://towardsdatascience.com/music-genre-recognition-using-convolutional-neural-networks-cnn-part-1-212c6b93da76>)
- [33] 台灣大學李宏毅老師, DLHLP 2020 Speech Recognition
(<https://www.youtube.com/watch?v=AIKu43goh-8>)
- [34] 維基百科 Morpheme 說明
(<https://en.wikipedia.org/wiki/Morpheme>)
- [35] LumenVox UK English Phonemes
(<https://www.lumenvox.com/knowledgebase/index.php?/article/AA-01085/0/UK-English-Phonemes.html>)
- [36] LumenVox UK English Phonemes
(<https://www.lumenvox.com/knowledgebase/index.php?/article/AA-01085/0/UK-English-Phonemes.html>)
- [37] Indian Accent Speech Recognition
(<https://anandai.medium.com/indian-accent-speech-recognition-2d433eb7edac>)
- [38] Automatic Speech Recognition-A Brief History of the Technology Development
(https://web.ece.ucsb.edu/Faculty/Rabiner/ece259/Reprints/354_LALI-ASRHistory-final-10-8.pdf)
- [39] Jonathan Hui, Speech Recognition — GMM, HMM
(<https://jonathan-hui.medium.com/speech-recognition-gmm-hmm-8bb5eff8b196>)
- [40] Oscar Contreras Carrasco, 《Gaussian Mixture Models Explained》, 2019
(<https://towardsdatascience.com/gaussian-mixture-models-explained-6986aaf5a95>)
- [41] 心學-知行合一, 《語音辨識系列 2--基於 WFST 解碼器_u012361418 的博客-程式師宅基地》, 2019
(<http://www.cxyzjd.com/article/u012361418/90289912>)
- [42] 陳柏琳, 《現階段大詞彙連續語音辨識研究之簡介》, 2005
(http://140.122.185.120/Berlin_Research/Manuscripts/2005_ACLCLP-Newsletter)

[現階段大詞彙連續語音辨識研究之簡介 Final.pdf](#))

[43] Kyle Kastne, GMM-HMM 類別 gmmhmm.py
(<https://gist.github.com/kastnerkyle/75483d51641a0c03bf7c>)

[44] Kyle Kastne, GMM-HMM 類別 gmmhmm.py
(<https://gist.github.com/kastnerkyle/75483d51641a0c03bf7c>)

[45] hmm-speech-recognition
(<https://code.google.com/archive/p/hmm-speech-recognition/downloads>)

[46] Keras Code Example 『 Automatic Speech Recognition using CTC 』
(https://keras.io/examples/audio/ctc_asr/)

[47] LJSpeech 資料集
(<https://keithito.com/LJ-Speech-Dataset/>)

[48] 維基百科 Word error rate
(https://en.wikipedia.org/wiki/Word_error_rate)

[49] PyLessons, Text Recognition With TensorFlow and CTC network
(<https://pylessons.com/ctc-text-recognition>)

[50] Whisper GitHub
(<https://github.com/openai/whisper>)

[51] FFmpeg 官網
(<https://ffmpeg.org/download.html>)

[52] 黑暗騎士崛起的對白音檔
(<https://www.moviesoundclips.net/movies1/darkknighttrises/darkness.mp3>)

[53] 200 句適合放入簡介的英文佳句座右銘
(<https://tw.amazingtalker.com/blog/zh-tw/zh-eng/32479/>)

第 17 章 強化學習

- [1] The Story of RLHF: Origins, Motivations, Techniques, and Modern Applications
(https://gymnasium.farama.org/environments/toy_text/frozen_lake/)
- [2] 維基百科關於強化學習的說明
(<https://zh.wikipedia.org/wiki/%E5%BC%BA%E5%8C%96%E5%AD%A6%E4%B9%A0>)
- [3] Gymnasium Classic Control 文件
(https://gymnasium.farama.org/environments/classic_control/)
- [4] ALE Documentation
(<https://ale.farama.org/environments/>)
- [5] Mujoco 官網文件
(<https://mujoco.readthedocs.io/en/stable/overview.html>)
- [6] From Scratch_ AI Balancing Act in 50 Lines of Python
(<https://towardsdatascience.com/from-scratch-ai-balancing-act-in-50-lines-of-python-7ea67ef717>)
- [7] Solving Open AI's CartPole Using Reinforcement Learning Part-1
(<https://medium.com/analytics-vidhya/q-learning-is-the-most-basic-form-of-reinforcement-learning-which-doesnt-take-advantage-of-any-8944e02570c5>)
- [8] Denny Britz Github
(<https://github.com/dennybritz/reinforcement-learning>)
- [9] Reinforcement Learning : An Introduction
(<http://incompleteideas.net/book/the-book-2nd.html>)
- [10] Gymnasium Frozen Lake
(https://gymnasium.farama.org/environments/toy_text/frozen_lake/)
- [11] Nuts & Bolts of Reinforcement Learning: Model Based Planning using Dynamic Programming
(<https://www.analyticsvidhya.com/blog/2018/09/reinforcement-learning-model-based-planning-dynamic-programming/>)
- [12] 維基百科關於蒙地卡羅方法的說明
(<https://zh.wikipedia.org/wiki/蒙地卡羅方法>)
- [13] Gymnasium Cliff Walking 官網文件說明
(https://gymnasium.farama.org/environments/toy_text/cliff_walking/)
- [14] Reinforcement Learning: On Policy and Off Policy
(<https://arshren.medium.com/reinforcement-learning-on-policy-and-off-policy->

[5587dd5417e1](#))

[15] Walking Off The Cliff With Off-Policy Reinforcement Learning
(<https://towardsdatascience.com/walking-off-the-cliff-with-off-policy-reinforcement-learning-7fdbcdfe31ff/>)

[16] Jeremy Zhang, 《Reinforcement Learning — Implement TicTacToe》, 2019
(<https://towardsdatascience.com/reinforcement-learning-implement-tictactoe-189582bea542>)

[17] How to Automatize a Warehouse Robot
(<https://medium.datadriveninvestor.com/get-started-with-q-learning-with-python-how-to-automatize-a-warehouse-robot-7bfac0180301>)

[18] PyTorch Reinforcement Learning (DQN) Tutorial
(https://pytorch.org/tutorials/intermediate/reinforcement_q_learning.html)

[19] Samina Amin, Deep Q-Learning (DQN)
(<https://medium.com/@samina.amin/deep-q-learning-dqn-71c109586bae>)

[20] vision-cartpole-dqn GitHub
(https://github.com/fedebotu/vision-cartpole-dqn/tree/save_model)

[21] Human-level control through deep reinforcement learning
(<https://web.stanford.edu/class/psych209/Readings/MnihEtAlHassibis15NatureControlDeepRL.pdf>)

[22] The Definitive Guide to Policy Gradients in Deep Reinforcement Learning
(<https://arxiv.org/pdf/2401.13662>)

[23] 李宏毅強化學習線上課程『DRL Lecture 1: Policy Gradient』
(<https://www.youtube.com/watch?v=z95ZYgPgXOY>)

[24] OpenAI Vanilla Policy Gradient
(<https://spinningup.openai.com/en/latest/algorithms/vpg.html>)

[25] tsmatsuz, Policy Gradient in Reinforcement Learning GitHub
(<https://github.com/tsmatz/reinforcement-learning-tutorials/tree/master>)

[26] Keras Actor Critic Method
(https://keras.io/examples/rl/actor_critic_cartpole/)

[27] Stable Baselines3
(<https://stable-baselines3.readthedocs.io>)

[28] Stable Baselines3 文件說明
(<https://stable-baselines3.readthedocs.io/en/master/guide/quickstart.html>)

[29] Illustrating Reinforcement Learning from Human Feedback
(<https://huggingface.co/blog/rhlf>)

[30] Yaodong Yang、Jun Wang, 《An Overview of Multi-Agent Reinforcement Learning from Game Theoretical Perspective》, 2020
(<https://arxiv.org/abs/2011.00583>)

[31] 維基百科強化學習介紹
(https://en.wikipedia.org/wiki/Reinforcement_learning)

[32] Reinforcement Learning: An Introduction
(<http://incompleteideas.net/book/the-book-2nd.html>)