

參考資料 (References)

第 1 章

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-ongraphics-chip-boom>)
2. Orhan G. Yalçın,《Top 5 Deep Learning Frameworks to Watch in 2021 and Why TensorFlow》, 2021
(<https://towardsdatascience.com/top-5-deep-learning-frameworks-to-watchin-2021-and-why-tensorflow-98d8d6667351>)
3. 陳昭明,《深度學習最佳入門邁向AI 專題實戰》, 2021
(https://www.tenlong.com.tw/products/9789860776263?list_name=b-r7-zh_tw)
4. Google Cloud 官網指南
(<https://cloud.google.com/ai-platform/docs/ml-solutions-overview>)
5. NVIDIA 官網說明
(<https://developer.nvidia.com/cuda-toolkit-archive>)
6. Colaboratory 官網說明
(<https://colab.research.google.com/notebooks/intro.ipynb>)
7. Mike Driscoll,《Jupyter Notebook: An Introduction》
(<https://realpython.com/jupyter-notebook-introduction/>)

第 2 章

1. Keith McNulty,《Decision makers need more math》, 2018
(<https://towardsdatascience.com/decision-makers-need-more-math-ed4d4fe3dc09>)
2. PyTorch 線性代數函數庫說明文件
(<https://pytorch.org/docs/stable/linalg.html>)

第 3 章

1. Quora, 《What is the difference between a Tensor and a Variable in Pytorch?》, 2019
(<https://www.quora.com/What-is-the-difference-between-a-Tensor-and-a-Variable-in-Pytorch>)
2. PyTorch 官網關於神經層的說明
(<https://pytorch.org/docs/stable/nn.html>)
3. 維基百科關於完全神經層(Linear Layers)的介紹
(https://en.wikipedia.org/wiki/Activation_function)

第 4 章

1. TensorFlow 官網的手寫阿拉伯數字辨識
(<https://www.tensorflow.org/overview>)
2. PyTorch 官網 Normalize 用法的介紹
(<https://pytorch.org/vision/stable/transforms.html#torchvision.transforms.Normalize>)
3. PyTorch 官網 CrossEntropyLoss 的說明
(<https://pytorch.org/docs/stable/generated/torch.nn.CrossEntropyLoss.html#crossentropyloss>)
4. fledlingbird, 《交叉熵損失, softmax 函数和 torch.nn.CrossEntropyLoss() 中文》, 2019 (<https://www.cnblogs.com/fledlingbird/p/10718096.html>)
5. PyTorch 官網 Saving and Loading Models
(https://pytorch.org/tutorials/beginner/saving_loading_models.html#saving-loading-model-across-devices)
6. Antonio Gulli、Amita Kapoor、Sujit Pal, 《Deep Learning with TensorFlow 2 and Keras》, 2019
(<https://www.amazon.com/Deep-Learning-TensorFlow-Keras-Regression/dp/1838823417>)
7. 維基百科 Activation Function 的介紹
(https://en.wikipedia.org/wiki/Activation_function)
8. PyTorch 優化器的介紹

(<https://pytorch.org/docs/stable/optim.html#algorithms>)

9. PyTorch 損失函數的介紹
(<https://pytorch.org/docs/stable/nn.html#loss-functions>)
10. PyTorch 官網 torch.nn.functional 的說明
(<https://pytorch.org/docs/stable/nn.functional.html>)
11. PyTorch 官網 Negative Log Likelihood Loss 的說明
(https://pytorch.org/docs/stable/generated/torch.nn.functional.nll_loss.html)
12. PyTorch 神經層的介紹 (<https://pytorch.org/docs/stable/nn.html>)
13. Geoffrey E. Hinton、Nitish Srivastava、Alex Krizhevsky, 《Improving neural networks by preventing co-adaptation of feature detectors》, 2012 (<https://arxiv.org/abs/1207.0580>)
14. PyTorch Activation Function 的介紹
(<https://pytorch.org/docs/stable/nn.html#non-linear-activations-weighted-sum-nonlinearity>)
15. Naoki, 《Gradient Descent Optimizers》, 2021
(<https://naokishibuya.medium.com/gradient-descent-optimizers-80d29f22deb5>)
16. Raimi Karim, 《10 Stochastic Gradient Descent Optimisation Algorithms + Cheat Sheet》, 2018
(<https://towardsdatascience.com/10-gradient-descent-optimisation-algorithms-86989510b5e9>)
17. Sanket Doshi, 《Various Optimization Algorithms For Training Neural Network》, 2019
(<https://towardsdatascience.com/optimizers-for-training-neural-network-59450d71caf6>)
18. Dominik Schmidt, 《Understanding Nesterov Momentum (NAG)》, 2018
(<https://dominikschmidt.xyz/nesterov-momentum/>)
19. Diederik P. Kingma、Jimmy Ba, 《Adam: A Method for Stochastic Optimization》, 2014 (<https://arxiv.org/abs/1412.6980>)
20. 深度學習於 NLP, 《一文告訴你 Adam、AdamW、Amsgrad 區別和聯繫》, 2018
(<https://zhuanlan.zhihu.com/p/39543160>)

21. Deniz Yuret,《Alec Radford's animations for optimization algorithms》, 2015
(<http://www.denizyuret.com/2015/03/alec-radfords-animations-for.html>)
22. Keras 官網中效能衡量指標的介紹 (<https://keras.io/api/metrics/>)
23. scikit-learn 文件
(https://scikit-learn.org/stable/modules/model_evaluation.html)
24. TorchMetrics 文件
(<https://torchmetrics.readthedocs.io/en/stable/pages/quickstart.html>)
25. PyTorch 官網關於 Ray Tune 的介紹
(https://pytorch.org/tutorials/beginner/hyperparameter_tuning_tutorial.html)
26. Search Space API
(https://docs.ray.io/en/latest/tune/api_docs/search_space.html#random-distributions-api)
27. Ray Tune 使用手冊
(<https://docs.ray.io/en/latest/tune/user-guide.html#auto-filled-metrics>)
28. Ray Tune 官網範例 (<https://docs.ray.io/en/latest/tune/index.html>)

第 5 章

1. Pytorch 官網 torchvision.datasets
(<https://pytorch.org/vision/stable/datasets.html>)
2. Pytorch 官網 torchaudio.datasets
(<https://pytorch.org/audio/stable/datasets.html>)
3. Pytorch 官網 torchtext.datasets
(<https://pytorch.org/text/stable/datasets.html>)
4. Pytorch 官網 torchvision.transforms
(<https://pytorch.org/vision/stable/transforms.html>)

5. Pytorch 官網 Illustration of transforms
(https://pytorch.org/vision/stable/auto_examples/plot_transforms.html#illustration-of-transforms)
6. 余霆嵩, 《PyTorch 學習筆記 (三): transforms 的二十二個方法》, 2018
(<https://zhuanlan.zhihu.com/p/53367135>)
7. TensorFlow 官網的 TensorBoard 指南
(https://www.tensorflow.org/tensorboard/get_started)
8. TorchServe GitHub (<https://github.com/pytorch/serve>)
9. TorchServe 官網 (<https://pytorch.org/serve/>)

第 6 章

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_l0pZ6JtDqVByuA.gif)
3. Pytorch 官網 Transforming and augmenting images
(<https://pytorch.org/vision/master/transforms.html>)
4. PyTorch 官網 Illustration of transforms
(https://pytorch.org/vision/stable/auto_examples/plot_transforms.html#sphx-glr-auto-examples-plot-transforms-py)
5. Antti Isosalo, PyTorch Implementation of CIFAR-10 Image Classification Pipeline Using VGG Like Network
(<https://github.com/aisosalo/CIFAR-10>)
6. Dhruvil Karani, 《How Data Augmentation Improves your CNN performance?》, 2020
(<https://medium.com/swlh/how-data-augmentation-improves-your-cnn-performance-an-experiment-in-pytorch-and-torchvision-e5fb36d038fb>)
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. Siladittya Manna, 《Extracting Features from an Intermediate Layer of a Pretrained ResNet Model in PyTorch》, 2021
(<https://medium.com/the-owl/extracting-features-from-an-intermediate-layer-of-a-pretrained-model-in-pytorch-easy-way-62631c7fa8f6>)
 10. SHAP 套件的安裝與介紹說明 (<https://github.com/slundberg/shap>)
 11. 維基百科中關於 Shapley value 的介紹
(https://en.wikipedia.org/wiki/Shapley_value)
 12. PyTorch Deep Explainer MNIST example
(https://shap.readthedocs.io/en/latest/example_notebooks/image_examples/image_classification/PyTorch%20Deep%20Explainer%20MNIST%20example.html)
 13. LIME 套件的安裝與介紹說明 (<https://github.com/marcotcr/lime>)
 14. Bolei Zhou, Aditya Khosla, Agata Lapedriza et al, 《Learning Deep Features for Discriminative Localization》, 2015
(<https://arxiv.org/pdf/1512.04150.pdf>)
 15. Kaggle 中介紹的實作
(<https://www.kaggle.com/aakashnain/what-does-a-cnn-see>)

第 7 章

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. Vishwesh Shrimali, 《PyTorch for Beginners: Image Classification using Pre-trained models》, 2019
(<https://learnopencv.com/pytorch-for-beginners-image-classification-using-pre-trained-models/>)
4. Pytorch 官網 MODELS AND PRE-TRAINED WEIGHTS
(<https://pytorch.org/vision/stable/models.html>)

5. Keras 官網 Keras Applications (<https://keras.io/api/applications/>)
6. 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>)
7. Yagnesh Revar GitHub
(<https://gist.github.com/yrevar/942d3a0ac09ec9e5eb3a>)
8. torchvision 原始程式碼 (<https://github.com/pytorch/vision>)
9. Sketchfab 網站 (<https://sketchfab.com/>)
10. Ethan Rosenthal, 《Using Keras' Pretrained Neural Networks for Visual Similarity Recommendations》, 2016
(<https://www.ethanrosenthal.com/2016/12/05/recasketch-keras/>)
11. Pytorch 官網 Transfer learning for computer vision tutorial
(https://pytorch.org/tutorials/beginner/transfer_learning_tutorial.html)
12. Pytorch 官網 Quantized transfer learning for computer vision tutorial
(https://pytorch.org/tutorials/intermediate/quantized_transfer_learning_tutorial.html)
13. Sergey Ioffe、Christian Szegedy, 《Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift》, 2015
(<https://arxiv.org/pdf/1502.03167.pdf>)
14. Aman Sawarn, 《Why Batch Normalization Matters?》, 2020
(<https://medium.com/towards-artificial-intelligence/why-batch-normalization-matters-4a6d753ba309>)
15. alexirpan, 《On The Perils of Batch Norm》, 2017
(<https://www.alexirpan.com/2017/04/26/perils-batch-norm.html>)

第 8 章

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. IIPImage (<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. 素娜 93, 《方向梯度直方圖 (HOG)》, 2017
(<https://www.jianshu.com/p/6f69c751e9e7>)
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. Jatin Prakash, 《Non Maximum Suppression: Theory and Implementation in PyTorch》, 2021
(<https://learnopencv.com/non-maximum-suppression-theory-and-implementation-in-pytorch/>)

13. 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>)
14. J. R. R. Uijlings、K. E. A. van de Sande、T. Gevers、A. W. M. Smeulders, 《Selective Search for Object Recognition》, 2012
(<https://github.com/object-detection-algorithm/selectivesearch>)
15. Github 『R-CNN: Regions with Convolutional Neural Network Features』
(<https://github.com/rbgirshick/rcnn>)
16. 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>)
17. vl_vivian, 《SPP-Net 論文詳解》, 2017
(<https://www.itread01.com/content/1542334444.html>)
18. Github 『A simple Spatial Pyramid Pooling layer which could be added in CNN』 (<https://github.com/yueruchen/sppnet-pytorch>)
19. Github 『Spatial pyramid pooling layers for keras』
(<https://github.com/yhenon/keras-spp>)
20. 白裳, 《捋一捋 pytorch 官方 FasterRCNN 代碼》, 2021
(<https://zhuanlan.zhihu.com/p/145842317>)
21. Johannes Schmidt, 《Train your own object detector with Faster-RCNN & PyTorch》, 2021
(<https://johschmidt42.medium.com/train-your-own-object-detector-with-faster-rcnn-pytorch-8d3c759cfc70>)
22. Ross B. Girshick 於 GitHub 上放置的 Faster R-CNN 程式碼
(<https://github.com/rbgirshick/py-faster-rcnn>)
23. Shangeth Rajaa、Satya Mallick, 《Faster R-CNN Object Detection with PyTorch》, 2019
(<https://learnopencv.com/faster-r-cnn-object-detection-with-pytorch/>)
24. Tsung-Yi Lin、Piotr Dollár、Ross Girshick, 《Feature Pyramid Networks for Object Detection》, 2016 (<https://arxiv.org/abs/1612.03144>)

第 9 章

1. Vijay Badrinarayanan、Alex Kendall、Roberto Cipolla, 《SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation》, 2015 (<https://arxiv.org/abs/1511.00561>)
2. Eugenia Anello, 《Denoising Autoencoder in Pytorch on MNIST dataset》, 2021 (<https://ai.plainenglish.io/denoising-autoencoder-in-pytorch-on-mnist-dataset-a76b8824e57e>)
3. 月下花弄影, 《ConvTranspose2d 原理, 深度網路如何進行上採樣》, 2019 (https://blog.csdn.net/qg_27261889/article/details/86304061)
4. Alexander Van de Kleut, 《Variational AutoEncoders (VAE) with PyTorch》, 2020 (<https://avandekleut.github.io/vae/>)
5. Satyam Kumar, 《7 Applications of Auto-Encoders every Data Scientist should know》, 2021 (<https://towardsdatascience.com/6-applications-of-auto-encoders-every-data-scientist-should-know-dc703cbc892b>)
6. Liang-Chieh Chen、George Papandreou 等人, 《DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs》, 2017 (<https://arxiv.org/pdf/1606.00915.pdf>)
7. Guosheng Lin、Anton Milan、Chunhua Shen, 《RefineNet: Multi-Path Refinement Networks for High-Resolution Semantic Segmentation》, 2016 (<https://arxiv.org/pdf/1611.06612.pdf>)
8. Hengshuang Zhao、Jianping Shi、Xiaojuan Qi, 《Pyramid Scene Parsing Network》, 2017 (<https://arxiv.org/pdf/1612.01105.pdf>)
9. Olaf Ronneberger、Philipp Fischer、Thomas Brox, 《U-Net: Convolutional Networks for Biomedical Image Segmentation》, 2015 (<https://arxiv.org/pdf/1505.04597.pdf>)
10. Naoto Usuyama, 『usuyama_pytorch-unet』 範例 (<https://github.com/usuyama/pytorch-unet>)
11. 皮特潘, 《語義分割之 dice loss 深度分析》, 2020 (<https://zhuanlan.zhihu.com/p/269592183>)
12. Mateusz Buda, 《PyTorch U-NET FOR BRAIN MRI》

(https://pytorch.org/hub/mateuszbuda_brain-segmentation-pytorch_unet)

13. Mateusz Buda , 《Kaggle Brain MRI segmentation》, 2019
(<https://www.kaggle.com/mateuszbuda/lgg-mri-segmentation>)
14. Kaiming He、Georgia Gkioxari、Piotr Dollár, Ross Girshick 等人, 《Mask R-CNN》, 2017 (<https://arxiv.org/pdf/1703.06870.pdf>)
15. Satya Mallick, 《Mask R-CNN Instance Segmentation with PyTorch》, 2019
(<https://learnopencv.com/mask-r-cnn-instance-segmentation-with-pytorch/>)
16. Penn-Fudan Database for Pedestrian Detection and Segmentation
(https://www.cis.upenn.edu/~jshi/ped_html/)
17. MODNet Github (<https://github.com/ZHKKKe/MODNet>)
18. PyTorch 官網『Torchvision object detection finetuning tutorial』
(https://pytorch.org/tutorials/intermediate/torchvision_tutorial.html)
19. Detectron2 Github (<https://github.com/facebookresearch/detectron2>)
20. Detectron2 官網文件
(<https://detectron2.readthedocs.io/en/latest/tutorials/install.html>)
21. fast-style-transfer GitHub
(<https://github.com/lengstrom/fast-style-transfer>)
22. 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》, 2017 (<https://www.bnext.com.tw/article/47330/ai-andy-meitu>)
23. Leon A. Gatys、Alexander S. Ecker、Matthias Bethge,《A Neural Algorithm of Artistic Style》, 2015 (<https://arxiv.org/abs/1508.06576>)
24. PyTorch 官網提供的範例『Neural transfer using pytorch』
(https://pytorch.org/tutorials/advanced/neural_style_tutorial.html#neural-transfer-using-pytorch)
25. 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>)

26. MTCNN_face_detection_alignment GitHub
(https://github.com/kpzhang93/MTCNN_face_detection_alignment)
27. mtcnn Github (<https://github.com/ipazc/mtcnn>)
28. facenet-pytorch Github
(<https://github.com/timesler/facenet-pytorch>)
29. 陳昭明,《dlib 安裝心得 -- Windows 環境》, 2020
(<https://ithelp.ithome.com.tw/articles/10231535>)
30. face-recognition GitHub 的範例
(https://github.com/ageitgey/face_recognition)
31. 羅之盈,《訊連養出 14 億美元獨角獸,玩美移動憑什麼赴美 IPO?》, 2022
(<https://www.gvm.com.tw/article/87786>)
32. 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)
33. Georgios Tzimiropoulos、Maja Pantic,《Optimization problems for fast AAM fitting in-the-wild》, 2013
(https://ibug.doc.ic.ac.uk/media/uploads/documents/tzimiropantic_iccv2013.pdf)
34. V.Kazemi、J. Sullivan,《One Millisecond Face Alignment with an Ensemble of Regression Trees》, 2014
(http://www.csc.kth.se/~vahidk/face_ert.html)
35. Tesseract OCR 官網
(<https://github.com/tesseract-ocr/tesseract/blob/master/doc/tesseract.1.asc>)
36. Filip Zelic、Anuj Sable,《A comprehensive guide to OCR with Tesseract, OpenCV and Python》, 2021
(<https://nanonets.com/blog/ocr-with-tesseract/>)
37. Tesseract 官網的語言列表
(<https://github.com/tesseract-ocr/tesseract/blob/master/doc/tesseract.1.asc#LANGUAGES>)
38. 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>)

39. Disadvantages of CNN models

(<https://iq.opengenus.org/disadvantages-of-cnn/>)

第 10 章

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 (<https://make.girls.moe/#/>)
8. MIL WebDNN (<https://mil-tokyo.github.io/webdnn/>)
9. Yunje Choi、Minje Choi、Munyoung Kim 等人,《StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation》, 2017 (<https://arxiv.org/abs/1711.09020>)
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. DCGAN-MNIST-pytorch Github
(<https://github.com/Ksuryateja/DCGAN-MNIST-pytorch>)
16. PyTorch 官網範例『DCGAN Tutorial』
(https://pytorch.org/tutorials/beginner/dcgan_faces_tutorial.html)
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. Fair Hdgan, 《PyTorch Progressive growing of GANs》
(https://pytorch.org/hub/facebookresearch_pytorch-gan-zoo_pgan/)
19. Mehdi Mirza、Simon Osindero, 《Conditional Generative Adversarial Nets》, 2014 (<https://arxiv.org/abs/1411.1784>)
20. Artur Machado Lacerda, 《Kaggle PyTorch Conditional GAN》, 2018
(<https://www.kaggle.com/arturlacerda/pytorch-conditional-gan>)
21. Qiwen Fu、Wei-Ting Hsu、Mu-Heng Yang, 《Colorization Using ConvNet and GAN》, 2017 (<http://cs231n.stanford.edu/reports/2017/pdfs/302.pdf>)
22. Marc Górriz Blanch、Marta Mrak、Alan F. Smeaton 等人,《End-to-End Conditional GAN-based Architectures for Image Colourisation》, 2019
(<https://github.com/bbc/ColorGAN#end-to-end-conditional-gan-based-architectures-for-image-colourisation>)
23. Phillip Isola、Jun-Yan Zhu、Tinghui Zhou, 《Image-to-Image Translation with Conditional Adversarial Networks》, 2016
(<https://arxiv.org/abs/1611.07004>)
24. Kooose, 《Kaggle Pix2Pix PyTorch》, 2022

- (<https://www.kaggle.com/kooose/pix2pix-pytorch>)
25. CMP Facade Database (<https://cmp.felk.cvut.cz/~tylecrl/facade/>)
 26. pytorch-CycleGAN-and-pix2pix Github
(<https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix>)
 27. 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>)
 28. CycleGAN-PyTorch Github
(<https://github.com/Lornatang/CycleGAN-PyTorch>)
 29. 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)
 30. tensorflow-generative-model-collections Github
(<https://github.com/hwalsuklee/tensorflow-generative-model-collections>)
 31. Shakir Mohamed、Danilo Rezende, 《Tutorial on Deep Generative Models》, 2017 (<http://www.shakirm.com/slides/DeepGenModelsTutorial.pdf>)
 32. BuzzFeedVideo, 《You Won't Believe What Obama Says In This Video!》, 2018 (<https://www.youtube.com/watch?v=cQ54GDmleL0>)
 33. 蔣曜宇, 《網紅小玉製作「換臉片」遭法辦！為何 Deepfake 讓 AI 變頭號網路犯罪公敵？》, 2021
(<https://www.bnext.com.tw/article/57260/deepfake-ai-deep-learning>)
 34. 鄭鴻達, 《遏阻 Deepfake 換臉 A 片 法務部提修法最重關七年》, 2021
(<https://udn.com/news/story/7321/5897568>)
 35. Alan Zucconi, 《Understanding the Technology Behind DeepFakes》, 2018
(<https://www.alanzucconi.com/2018/03/14/understanding-the-technology-behind-deepfakes/>)
 36. Aayush Bansal、Shugao Ma、Deva Ramanan、Yaser Sheikh, 《Recycle-GAN: Unsupervised Video Retargeting》, 2018
(<https://arxiv.org/abs/1808.05174>)

37. Jonathan Hui, 《Detect AI-generated Images & Deepfakes》, 2020
(<https://jonathan-hui.medium.com/detect-ai-generated-images-deepfakes-part-1-b518ed5075f4>)
38. Louis Bouchard, 《DeepFakes in 5 minutes》, 2020
(<https://pub.towardsai.net/deepfakes-in-5-minutes-l55cl3d48fa3>)
39. DeepFaceLab GitHub (<https://github.com/iperov/DeepFaceLab>)
40. Jonathan Hui, 《Detect AI-generated Images & Deepfakes (Part 1)》, 2020
(<https://jonathan-hui.medium.com/detect-ai-generated-images-deepfakes-part-1-b518ed5075f4>)
41. 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

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. Dr. Srijith Rajamohan, 《Word2Vec in Pytorch – Continuous Bag of Words and Skipgrams》, 2018 (<https://srijithr.gitlab.io/post/word2vec/>)
7. Gensim 官網關於 Word2Vec 的說明
(<https://radimrehurek.com/gensim/models/word2vec.html>)

8. Pierre Megret, 《Gensim Word2Vec Tutorial》, 2019
(<https://www.kaggle.com/pierremegret/gensim-word2vec-tutorial>)
9. Embedding Projector (<https://projector.tensorflow.org/>)
10. Jeffrey Pennington、Richard Socher、Christopher D. Manning, 《GloVe: Global Vectors for Word Representation》, 2014
(<https://www.aclweb.org/anthology/D14-1162.pdf>)
11. glove.42B.300d.zip
(<https://nlp.stanford.edu/data/wordvecs/glove.42B.300d.zip>)
12. glove.840B.300d.zip
(<https://nlp.stanford.edu/data/wordvecs/glove.840B.300d.zip>)
13. glove.6B.300d.zip
(<https://nlp.stanford.edu/data/wordvecs/glove.6B.zip>)
14. glove.twitter.27B.zip
(<https://nlp.stanford.edu/data/wordvecs/glove.twitter.27B.zip>)
15. 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>)
16. 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>)
17. spaCy (<https://spacy.io/usage>)
18. spaCy Quickstart (<https://spacy.io/usage/models>)
19. pkuseg GitHub (<https://github.com/explosion/spacy-pkuseg>)
20. spaCy 『spaCy 101: Everything you need to know』
(<https://spacy.io/usage/spacy-101>)
21. spaCy Token (<https://spacy.io/api/token>)
22. glossary.py GitHub
(<https://github.com/explosion/spaCy/blob/master/spacy/glossary.py>)
)

23. displaCy visualizer 的說明文件
(<https://spacy.io/api/top-level#displacy>)

第 12 章

1. PyTorch 官網嵌入層說明
(<https://pytorch.org/docs/stable/generated/torch.nn.Embedding.html>)
2. PyTorch 官網 RNN 層說明
(<https://pytorch.org/docs/stable/generated/torch.nn.RNN.html>)
3. PyTorch 所支援的詞向量
(https://pytorch.org/text/stable/_modules/torchtext/vocab/vectors.html#GloVe)
4. Martin Pellarolo, 《How to use Pre-trained Word Embeddings in PyTorch》, 2018
(<https://medium.com/@martinpella/how-to-use-pre-trained-word-embeddings-in-pytorch-71ca59249f76>)
5. PyTorch 官網 TorchText Datasets
(<https://pytorch.org/text/stable/datasets.html>)
6. PyTorch 官網 TorchData Tutorial
(<https://pytorch.org/data/beta/tutorial.html#using-datapipes>)
7. PyTorch 官網 Text classification with the torchtext library
(https://pytorch.org/tutorials/beginner/text_sentiment_ngrams_tutorial.html)
8. Antonio Gulli, 《AG's corpus of news articles》
(<https://paperswithcode.com/dataset/ags-corpus>)
9. Christopher Olah, 《Understanding LSTM Networks》, 2015
(<https://colah.github.io/posts/2015-08-Understanding-LSTMs/>)
10. PyTorch LSTM 神經層
(<https://pytorch.org/docs/stable/generated/torch.nn.LSTM.html>)
11. 我唱歌比較走心, 《pytorch 實現 IMDB 資料集情感分類》, 2021
(<https://blog.csdn.net/Delusional/article/details/113357449>)
12. 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/>)

13. 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>)
14. 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>)
15. Alexandre Xavier,《Predicting stock prices with LSTM》, 2019
(<https://medium.com/neuronio/predicting-stock-prices-with-lstm-349f5a0974d4>)
16. szrlee,《DJIA 30 Stock Time Series》, 2018
(<https://www.kaggle.com/szrlee/stock-time-series-20050101-to-20171231>)
17. 陳昭明,《演算法交易(Algorithmic Trading)實作》, 2021
(<https://ithelp.ithome.com.tw/articles/10255111>)
18. 張俊林博客,《深度學習中的注意力機制(2017版)》, 2017
(<https://blog.csdn.net/malefactor/article/details/78767781>)
19. Meng Lee,《淺談神經機器翻譯 & 用 Transformer 與 TensorFlow 2 英翻中》, 2019
(<https://leemeng.tw/neural-machine-translation-with-transformer-and-tensorflow2.html>)
20. Lilian Weng,《Attention? Attention!》, 2018
(<https://lilianweng.github.io/posts/2018-06-24-attention/>)
21. PyTorch 官網範例『Translation with a sequence to sequence network and attention』
(https://pytorch.org/tutorials/intermediate/seq2seq_translation_tutorial.html)
22. Andrej Karpathy,《The Unreasonable Effectiveness of Recurrent Neural Networks》, 2015
(<http://karpathy.github.io/2015/05/21/rnn-effectiveness/>)

23. Ashish Vaswani、Noam Shazeer、Niki Parmar,《Attention Is All You Need》, 2017 (<https://arxiv.org/pdf/1706.03762.pdf>)
24. Raimi Karim,《Illustrated: Self-Attention》, 2019 (<https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a>)
25. Jay Alammar,《The Illustrated Transformer》, 2018 (<http://jalammar.github.io/illustrated-transformer/>)
26. GeeksforGeeks,《Self-attention in NLP》, 2020 (<https://www.geeksforgeeks.org/self-attention-in-nlp/>)
27. Jason Brownlee,《A Gentle Introduction to Calculating the BLEU Score for Text in Python》, 2019 (<https://machinelearningmastery.com/calculate-bleu-score-for-text-python/>)
28. Jacob Devlin、Ming-Wei Chang、Kenton Lee、Kristina Toutanova,《BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding》, 2018 (<https://arxiv.org/abs/1810.04805>)
29. Hugging Face Transformers (<https://huggingface.co/docs/transformers/quicktour>)
30. Rani Horev,《BERT Explained: State of the art language model for NLP》, 2018 (<https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270>)
31. BERT GitHub (<https://github.com/google-research/bert>)
32. Anna Rogers、Olga Kovaleva、Anna Rumshisky,《A Primer in BERTology: What we know about how BERT works》, 2020 (<https://arxiv.org/abs/2002.12327>)
33. Transformers GitHub (<https://github.com/huggingface/transformers>)
34. Transformers『Quick tour』 (<https://huggingface.co/transformers/quicktour.html>)
35. Transformers 官網『Summary of the tasks』的 Extractive Question Answering (https://huggingface.co/transformers/task_summary.html#extractive)

[-question-answering\)](#)

36. Transformers 官網『Using tokenizers from Tokenizers』
(https://huggingface.co/transformers/fast_tokenizers.html)
37. Transformers 官網『Summary of the tasks』的 Masked Language Modeling
(https://huggingface.co/transformers/task_summary.html#masked-language-modeling)
38. Transformers 官網『Summary of the tasks』的 Text Generation
(https://huggingface.co/transformers/task_summary.html#text-generation)
39. Transformers 官網『Summary of the tasks』的 Named Entity Recognition
(https://huggingface.co/transformers/task_summary.html#named-entity-recognition)
40. Transformers 官網『Summary of the tasks』的 Summarization
(https://huggingface.co/transformers/task_summary.html#summarization)
41. 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>)
42. Transformers 官網『Summary of the tasks』的 Translation
(https://huggingface.co/transformers/task_summary.html#translation)
43. Transformers 官網『Training and fine-tuning』
(<https://huggingface.co/transformers/training.html#tensorflow>)
44. https://colab.research.google.com/github/huggingface/notebooks/blob/master/examples/text_classification.ipynb
45. Transformers 官網的『DatasetDict 說明文件』
(https://huggingface.co/docs/datasets/package_reference/main_classes.html#datasetdict)
46. Transformers 官網的『TrainingArguments 說明文件』
(https://huggingface.co/transformers/main_classes/trainer.html#transformers.TrainingArguments)

47. CKIP Transformers Github
(<https://github.com/ckiplab/ckip-transformers>)
48. CKIP Transformers 官網說明
(<https://ckip-transformers.readthedocs.io/en/latest/main/readme.html>)
49. Transformers Pretrained models
(https://huggingface.co/transformers/pretrained_models.html)
50. 王若樸,《AI 趨勢周報第 142 期：推理能力新突破！OpenAI 新作 GPT-f 能自動證明數學定理》，2020 (<https://www.ihome.com.tw/news/140030>)
51. 王若樸,《AI 趨勢周報第 167 期：臉書新模型融合自監督和 Transformer，不需標註資料還能揪出複製圖》，2021
(<https://www.ihome.com.tw/news/144208>)

第 13 章

1. Adnan Rehan,《10 Best Chatbot Development Frameworks to Build Powerful Bots》，2020 (<https://geekflare.com/chatbot-development-frameworks/>)
2. Data Flair 《Learn to build your first chatbot using NLTK & Keras》
(<https://data-flair.training/blogs/python-chatbot-project>)
3. ChatterBot GitHub
(<https://github.com/gunthercox/ChatterBot/tree/3eccccddd2a14eccc0aef12df7fa68513a464a00>)
4. ChatBotAI GitHub (<https://github.com/ahmadfaizalbh/Chatbot>)
5. Rasa 官網 (<https://rasa.com/>)
6. mathparse GitHub (<https://github.com/gunthercox/mathparse>)
7. Wikipedia GitHub (<https://github.com/goldsmith/Wikipedia>)
8. Dialogflow 的官網說明 (<https://cloud.google.com/dialogflow/docs>)
9. Google Dialogflow Editions
(<https://cloud.google.com/dialogflow/docs/editions>)
10. Google Dialogflow System entities reference
(<https://cloud.google.com/dialogflow/es/docs/reference/system-ent>)

ities)

11. Google Dialogflow Predefined follow-up intents
(<https://cloud.google.com/dialogflow/es/docs/reference/follow-up-intent-expressions>)
12. Dialogflow Integrations 說明
(<https://dialogflow.cloud.google.com/#/agent/get-agent-name-ojw9/integrations>)
13. DialogFlow Prebuilt Agents 說明
(<https://dialogflow.cloud.google.com/#/agent/get-agent-name-ojw9/prebuiltAgents/>)
14. LINE Developers 文件
(<https://developers.line.biz/zh-hant/docs/messaging-api/getting-started/#using-oa-manager>)

第 14 章

1. 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>)
2. 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>)
3. 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>)
4. Gfycat 說明振幅、頻率及相位的動畫
(<https://gfycat.com/ickyfilthybobolink>)
5. Pema Grg, 《Audio Signal Processing》, 2020
(<https://blog.ekbana.com/audio-signal-processing-f7e86d415489>)
6. 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))

7. CPT-Sound-ADC-DAC.svg Wikimedia Commons
(<https://commons.wikimedia.org/wiki/File:CPT-Sound-ADC-DAC.svg>)
8. sample-files Github (<https://github.com/maxifjaved/sample-files>)
9. wave 說明文件 (<https://docs.python.org/3/library/wave.html>)
10. SpeechRecognition Project description
(<https://pypi.org/project/SpeechRecognition/2.1.2/>)
11. Librosa 說明文件 (<https://librosa.org/doc/latest/tutorial.html>)
12. Vincent Koops, 《Introduction Basic Audio Feature Extraction》, 2017
(http://www.cs.uu.nl/docs/vakken/msmt/lectures/SMT_B_Lecture5_DSP_2017.pdf)
13. 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>)
14. Henry Haefliger, 《Python audio spectrum analyzer》, 2019
(<https://medium.com/quick-code/python-audio-spectrum-analyser-6a3c54ad950>)
15. FFmpeg 官網 (<http://ffmpeg.org/download.html>)
16. FFmpeg 官網文件 (<http://ffmpeg.org/documentation.html>)
17. PyTorch Audio 教學文件
(https://pytorch.org/tutorials/beginner/audio_io_tutorial.html)
18. Pytorch Torchaudio.Backend
(<https://pytorch.org/audio/stable/backend.html>)
19. sox 工具軟體 (<https://sourceforge.net/projects/sox/files/sox>)
20. pysox Github (<https://github.com/rabitt/pysox>)
21. Pytorch Torchaudio.Transforms
(<https://pytorch.org/audio/stable/transforms.html>)
22. Hohyub Jeon、Yongchul Jung、Seongjoo Lee、Yunho Jung, 《Area-Efficient Short-Time Fourier Transform Processor for Time - Frequency Analysis

- of Non-Stationary Signals》, 2020
(<https://www.mdpi.com/2076-3417/10/20/7208/htm>)
23. PyTorch Torchaudio.Datasets
(<https://pytorch.org/audio/stable/datasets.html>)
 24. Pyroomacoustics CMU ARCTIC Corpus
(https://pyroomacoustics.readthedocs.io/en/pypi-release/pyroomacoustics.datasets.cmu_arctic.html)
 25. 維基百科 CMU Pronouncing Dictionary
(https://en.wikipedia.org/wiki/CMU_Pronouncing_Dictionary)
 26. Common Voice Datasets
(<https://commonvoice.mozilla.org/en/datasets>)
 27. Common Voice 繁體中文 (<https://commonvoice.mozilla.org/zh-TW>)
 28. GTZAN Genre Collection
(<http://marsyas.info/downloads/datasets.html>)
 29. Open Speech and Language Resources (<https://www.openslr.org/1>)
 30. Pete Warden, 《Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition》, 2018 (<https://arxiv.org/abs/1804.03209>)
 31. 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>)
 32. prasad213 music-genre-classification
(<https://jovian.ai/prasad213/music-genre-classification>)
 33. 陳昭明, 《Day 25: 自動語音辨識(Automatic Speech Recognition) -- 觀念與實踐》, 2018 (<https://ithelp.ithome.com.tw/articles/10195763>)
 34. kaggle 官網『TensorFlow Speech Recognition Challenge』
(<https://www.kaggle.com/c/tensorflow-speech-recognition-challenge>)
 35. 維基百科關於音素的說明 (<https://zh.wikipedia.org/wiki/音位>)
 36. Anand P V, 《Indian Accent Speech Recognition》, 2020

(<https://anandai.medium.com/indian-accent-speech-recognition-2d433eb7edac>)

37. Amazon Polly 支援語言的音素
(https://docs.aws.amazon.com/zh_tw/polly/latest/dg/ref-phoneme-tables-shell.html)
38. Oscar Contreras Carrasco, 《Gaussian Mixture Models Explained》, 2019
(<https://towardsdatascience.com/gaussian-mixture-models-explained-6986aaf5a95>)
39. 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>)
40. 『愛丁堡大學語音辨識課程』第 11 章
(<http://www.inf.ed.ac.uk/teaching/courses/asr/lectures-2019.html>)
41. 陳柏琳, 《現階段大詞彙連續語音辨識研究之簡介》, 2005
(http://berlin.csie.ntnu.edu.tw/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)
42. Kaldi 官網操作說明文件 (<http://kaldi-asr.org/doc/index.html>)
43. 台灣大學李琳山教授 Introduction to Digital Speech Processing 2019 Spring (<http://speech.ee.ntu.edu.tw/DSP2019Spring/>)
44. 哥倫比亞大學 Speech Recognition EECS E6870 — Fall 2012
(<https://www.ee.columbia.edu/~stanchen/fall12/e6870/outline.html>)
45. 愛丁堡大學 Automatic Speech Recognition (ASR) 2018-19: Lectures
(<https://www.inf.ed.ac.uk/teaching/courses/asr/lectures-2019.html>)
46. OpenSLR Resources (<https://www.openslr.org/resources.php>)
47. TIMIT Acoustic-Phonetic Continuous Speech Corpus
(<https://catalog.ldc.upenn.edu/LDC93S1>)
48. WikiProject Spoken Wikipedia
(https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Spoken_Wikipedia)

第 15 章

1. 維基百科關於強化學習的說明(<https://zh.wikipedia.org/wiki/强化学习>)
2. Gym 官網 (<https://gym.openai.com/>)
3. Gym GitHub (<https://github.com/openai/gym#installing-everything>)
4. Sayan Mandal, 《Install OpenAI Gym with Box2D and Mujoco in Windows 10》, 2019
(<https://medium.com/@sayanmnd121/install-openai-gym-with-box2d-and-mujoco-in-windows-10-e25ee9b5c1d5>)
5. Yuval Tassa、Saran Tunyasuvunakool、Nimrod Gileadi, 《Opening up a physics simulator for robotics》, 2021
(<https://deepmind.com/blog/announcements/mujoco>)
6. Mujoco 官網操作說明文件
(<https://mujoco.readthedocs.io/en/latest/overview.html#examples>)
7. Mike Shi, 《From Scratch: AI Balancing Act in 50 Lines of Python》, 2018
(<https://towardsdatascience.com/from-scratch-ai-balancing-act-in-50-lines-of-python-7ea67ef717>)
8. Maciej Balawejder, 《Solving Open AI' s CartPole Using Reinforcement Learning Part-1》, 2021
(<https://medium.com/analytics-vidhya/q-learning-is-the-most-basic-form-of-reinforcement-learning-which-doesnt-take-advantage-of-any-8944e02570c5>)
9. Denny Britz Github
(<https://github.com/dennybritz/reinforcement-learning>)
10. Richard S. Sutton、Andrew G. Barto, 《Reinforcement Learning: An Introduction》, 2018
(<http://incompleteideas.net/book/the-book-2nd.html>)
11. 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>)
12. 維基百科關於二十一點的說明 (<https://zh.wikipedia.org/wiki/二十一點>)

13. Jeremy Zhang, 《Reinforcement Learning — Implement TicTacToe》, 2019
(<https://towardsdatascience.com/reinforcement-learning-implement-tictactoe-189582bea542>)
14. Volodymyr Mnih、Koray Kavukcuoglu、David Silver 等人, 《Human-level control through deep reinforcement learning》, 2015
(<https://web.stanford.edu/class/psych209/Readings/MnihEtAlHassibi15NatureControlDeepRL.pdf>)
15. Keras 官網 Apoorv Nandan, 《Actor Critic Method》, 2020
(https://keras.io/examples/rl/actor_critic_cartpole/)
16. Pytorch 官網 Adam Paszke, 《Reinforcement Learning (DQN) Tutorial》
(https://pytorch.org/tutorials/intermediate/reinforcement_q_learning.html)
17. Stable Baselines3 官網 Stable-Baselines3 Docs - Reliable Reinforcement Learning Implementations
(<https://stable-baselines3.readthedocs.io/en/master/>)
18. Rob, 《Get Started With Q-Learning With Python: How To Automatize A Warehouse Robot》, 2020
(<https://medium.datadriveninvestor.com/get-started-with-q-learning-with-python-how-to-automatize-a-warehouse-robot-7bfae0180301>)
19. Yaodong Yang、Jun Wang, 《An Overview of Multi-Agent Reinforcement Learning from Game Theoretical Perspective》, 2020
(<https://arxiv.org/abs/2011.00583>)
20. 維基百科關於強化學習的介紹
(https://en.wikipedia.org/wiki/Reinforcement_learning)

第 16 章

1. Univariate Distribution Relationships
(<http://www.math.wm.edu/~leemis/chart/UDR/UDR.html>)
2. Software Catepentry, Advanced NumPy
(<https://paris-swc.github.io/advanced-numpy-lesson/03-broadcasting.html>)
3. NetworkX Algorithms
(<https://networkx.org/documentation/stable/reference/algorithms/i>

[ndex.html](#))

4. Bumstead 教授早上起床後的穿戴順序
(<https://networkx.org/nx-guides/content/algorithms/dag/index.html>)
5. NetworkX Graph Layout
(<https://networkx.org/documentation/stable/reference/drawing.html#module-networkx.drawing.layout>)
6. community detection 社區發現, 隨勛所欲, 2020
(<https://smiliu.xyz/posts/22257>)
7. Community Detection 演算法, pegohty, 2019
(<https://blog.csdn.net/itplus/article/details/9286905>)
8. NetworkX Louvain Communities
(https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.community.louvain.louvain_communities.html#louvain-communities)
9. PyTorch Geometric(PyG)官網文件
(<https://pytorch-geometric.readthedocs.io/en/latest/notes/installation.html>)
10. Christopher Morris 等學者, TUDatasets
(<https://chrsmrrs.github.io/datasets/>)
11. TUDatasets 網站
(<https://chrsmrrs.github.io/datasets/docs/datasets/>)
12. Torch Geometric transforms
(<https://pytorch-geometric.readthedocs.io/en/latest/modules/transforms.html>)
13. Thomas kipf, Graph Convolutional Network
(<http://tkipf.github.io/graph-convolutional-networks/>)
14. Planetoid 類別說明
(https://pytorch-geometric.readthedocs.io/en/latest/modules/datasets.html#torch_geometric.datasets.Planetoid)
15. Creating Message Passing Networks
(https://pytorch-geometric.readthedocs.io/en/latest/notes/create_

[gnn.html](#))

16. Keyulu Xu 等學者, How Powerful are Graph Neural Networks?, 2018
(<https://arxiv.org/abs/1810.02244>)
17. Christopher Morris 等學者, Weisfeiler and Leman Go Neural:
Higher-order Graph Neural Networks, 2018
(<https://arxiv.org/abs/1810.00826>)