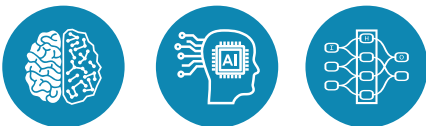




Fundamental of

DEEP LEARNING in Practice



Best Practice Workshop

Machine Learning, AI Capabilities และ AI Applications

ทำความเข้าใจแนวคิด การพัฒนาโมเดล และการประยุกต์ใช้งาน
ด้วยการเขียนโค้ด Python บน Jupyter Notebook
โดยใช้ Library เช่น TensorFlow, Scikit-learn และ NumPy



SOURCE CODE :
SERAZU.COM

ผู้แต่ง อ.ดร.ณัฐชิตี พรหมฤทธิ
อ.ดร.สัจจาภรณ์ ไวยรรยา



Fundamental of DEEP LEARNING in Practice

Writers

ดร.ณัฐโชติ พรหมฤทธิ, ดร.สัจจาภรณ์ ไวจรรยา

Editor

ภิรพล คชาเจริญ

Graphic Designers

ชวนันท์ รัตนะ, สิริลักษณ์ วาระเลิศ

Page Layout

วรวิทย์ วรจินต์

Proofreader

เกษรา พรวัฒนมงคล

Publishing Coordinators

วรพล ณิชกุล, สุพัตรา อาจปรุ, ศรัณย์ คมขำ

Jupyter Notebook เป็นเครื่องหมายการค้าของบริษัท Project Jupyter, CUDA Toolkit เป็นเครื่องหมายการค้าของบริษัท NVIDIA Corporation, Miniconda เป็นเครื่องหมายการค้าของบริษัท Anaconda, Inc. และเครื่องหมายการค้าอื่นๆ ที่อ้างถึงเป็นของบริษัทนั้นๆ สงวนลิขสิทธิ์ตามพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537 โดยบริษัท ไอดีซี พรีเมียร์ จำกัด ห้ามลอกเลียนไม่ว่าส่วนใดส่วนหนึ่งของหนังสือเล่มนี้ ไม่ว่าในรูปแบบใดๆ นอกจากจะได้รับอนุญาตเป็นลายลักษณ์อักษรจากผู้จัดพิมพ์เท่านั้น

บริษัท ไอดีซี พรีเมียร์ จำกัด จัดตั้งขึ้นเพื่อเผยแพร่ความรู้ที่มีคุณภาพสู่ผู้อ่านชาวไทย เรายินดีต้อนรับเขียนของนักวิชาการและนักเขียนทุกท่าน ท่านผู้สนใจกรุณาติดต่อผ่านทางอีเมลที่ infopress@idcpremier.com หรือทางโทรศัพท์หมายเลข 0-2962-1081 (อัตรานาที 10 คู่สาย) โทรสาร 0-2962-1084

สร้างสรรค์โดย



พิมพ์ครั้งที่ 1 ธันวาคม 2564

ข้อมูลทางบรรณานุกรม

ดร.ณัฐโชติ พรหมฤทธิ

Fundamental of DEEP LEARNING
in Practice

นนทบุรี : ไอดีซี, 2564

504 หน้า

1. การเขียนโปรแกรมสำหรับระบบ
ฐานความรู้

I ดร.สัจจาภรณ์ ไวจรรยา (ผู้แต่งร่วม)

II ชื่อเรื่อง

006.336

ISBN 885-916-100-927-6

ราคา 545 บาท

พิมพ์และจัดจำหน่ายโดย



บริษัท ไอดีซี พรีเมียร์ จำกัด

200 หมู่ 4 ชั้น 19 ห้อง 1901

จัสมินอินเตอร์เนชั่นแนลทาวเวอร์

ถ.แจ้งวัฒนะ อ.ปากเกร็ด จ.นนทบุรี 11120

โทรศัพท์ 0-2962-1081 (อัตรานาที 10 คู่สาย)

โทรสาร 0-2962-1084

สมาชิกสัมพันธ์

โทรศัพท์ 0-2962-1081-3 ต่อ 121

โทรสาร 0-2962-1084

ร้านค้าและตัวแทนจำหน่าย

โทรศัพท์ 0-2962-1081-3 ต่อ 112-114

โทรสาร 0-2962-1084

PREFACE

Fundamental of DEEP LEARNING in Practice เป็นหนังสือที่จะปูพื้นฐานที่จำเป็นสำหรับผู้เริ่มต้นศึกษาด้าน AI และ Deep Learning โดยมุ่งให้ผู้อ่านสามารถทำความเข้าใจแนวคิดที่เป็นรากฐานไปจนถึงขั้นการฝึกสอน Model เพื่อนำไปใช้งานทางด้านภาพ การเล่นเกม และระบบการแนะนำ ด้วยการลงมือเขียน Code ภาษา Python บน Jupyter Notebook โดยใช้ Library ต่างๆ อย่างเช่น TensorFlow, Scikit-learn และ NumPy แทนที่จะเน้นการอธิบายด้วยทฤษฎีหนักๆ

ถ้าคุณกำลังเรียนในสาขาวิทยาการคอมพิวเตอร์ เทคโนโลยีสารสนเทศ หรือวิทยาการข้อมูล แต่ละบทของหนังสือเล่มนี้จะเป็นจุด Check Point ที่บอกว่าคุณมีความพร้อมที่จะเดินทางต่อไปยังระดับที่ Advance ขึ้น

ถ้าคุณกำลังทำงานที่ต้องใช้ Skill ทางด้าน AI หรือวิทยาการข้อมูล หนังสือเล่มนี้จะนำคุณกลับมาเรียนรู้พื้นฐานสำคัญของมันโดยใช้เวลาไม่นานนัก

เนื้อหาของหนังสือถูกแบ่งเป็น 2 ส่วน ในส่วนแรก (บทที่ 1 ถึงบทที่ 12) จะพูดถึงแนวคิดพื้นฐานของ AI และ Machine Learning กระบวนการพัฒนา Model แบบ Deep Learning การทำ Feature Engineering พื้นฐาน Neural Network การปรับจูน Hyper Parameter ของ Model การวิเคราะห์ประสิทธิภาพการเรียนรู้ของ Model ด้วย Learning Curve การทำ Regularization โครงสร้างแบบ Convolutional Neural Network ใน Deep Learning Model ที่สามารถเพิ่มประสิทธิภาพให้แก่ Neural Network ได้อย่างไม่เคยเป็นมาก่อน รวมทั้งการเลือกใช้ Loss Function และการประเมิน Classification Model ด้วย Metrics สำคัญๆ

ในส่วนที่สอง (บทที่ 13 ถึงบทที่ 16) ผู้อ่านจะได้เห็นการนำ Model ไปใช้งานด้วยการ Deploy มั่นบน Production Environment ซึ่งเป็นส่วนสำคัญที่จะทำให้เกิดประโยชน์จริงในทาง Business การฝึก AI Agent ให้เรียนรู้การเล่นเกมด้วย Deep Q-Network (DQN) การนำ Weight ของ Pre-trained CNN Models ไปประยุกต์ใช้ และการแนะนำหนังสือจากการทำนายด้วย Autoencoder Model

ในยุคที่ AI และวิทยาการข้อมูลได้เข้ามามีบทบาทสำคัญในการขับเคลื่อนเศรษฐกิจโลก ผู้เขียนหวังเป็นอย่างยิ่งว่า หนังสือเล่มนี้จะเป็นส่วนเล็กๆ ในการ Upskill/Reskill ให้แก่นักศึกษา บุคลากรในภาคอุตสาหกรรมดิจิทัล รวมทั้งผู้ที่สนใจในงานทางด้านนี้

อ. ดร.ณัฐโชติ พรหมฤทธิ์

อาจารย์ประจำภาควิชาคอมพิวเตอร์

รองผู้อำนวยการฝ่ายพัฒนางานองค์กร สำนักดิจิทัลเทคโนโลยี

มหาวิทยาลัยศิลปากร

อ. ดร.สัจจาภรณ์ ไวจรรยา

อาจารย์ประจำภาควิชาคอมพิวเตอร์

มหาวิทยาลัยศิลปากร

EDITOR'S NOTE

กระแสของ AI ปลุกให้มนุษย์ทุกคนบนโลกได้ตระหนักผ่านสื่อต่างๆ หลากหลายช่องทาง อย่างเช่น "AI Transforming the World" โดย Forbes ที่กล่าวว่า "โลกกำลังพัฒนาอย่างรวดเร็วโดยมีปัญญาประดิษฐ์ (AI) เป็นผู้นำในการเปลี่ยนแปลงโลกและวิถีชีวิตของเรา" และ "Artificial Intelligence has changed our world" โดย Wall Street International Magazine กล่าวว่า "AI ได้เปลี่ยนจากนิยายวิทยาศาสตร์ชวนฝันมาเป็นส่วนสำคัญในการใช้ชีวิตประจำวันของเราแล้ว"

ในหนังสือ Human + Machine: Reimagining Work in the Age of AI โดย By Paul R. Daugherty, H. James Wilson ได้อ้างถึงงานของ Accenture Research; Jerry Kaplan: Artificial Intelligence โดยแสดงภาพ The constellation of AI technologies and business applications (p. 61) ประกอบด้วยวงกลม 3 วง ได้แก่ วงในสุด Machine Learning วงกลาง AI Capabilities และวงนอก AI Applications

โดย Machine Learning เป็นวิทยาการคอมพิวเตอร์ด้านอัลกอริทึมที่สามารถเรียนรู้ และพยากรณ์ได้เองโดยไม่ต้องเขียนโปรแกรมที่มีกฎหรือเงื่อนไขที่ซับซ้อนจำนวนมาก แต่ต้องอาศัยข้อมูลจำนวนมากพอในการสอนโมเดลแบบ Deep Learning จนเข้าใจคุณลักษณะ ส่วน AI Capabilities เป็นความสามารถของ AI ในด้านต่างๆ เช่น ระบบทำนาย ระบบผู้เชี่ยวชาญ ระบบคอมพิวเตอร์วิชั่น ระบบประมวลผลเสียงหรือภาษา ระบบแปลงเสียงเป็นข้อความ เป็นต้น และสุดท้าย AI Applications ซึ่งเป็นการประยุกต์ใช้ความสามารถของ AI เพื่อใช้ประโยชน์จริง เช่น Book Recommendation ของ Amazon, Movie Recommendation ของ Netflix, Rekognition ระบบตรวจจับใบหน้าและเนื้อหาที่ไม่เหมาะสมของ Amazon, AIPRO ระบบเทร็ดหุ้นอัตโนมัติ, AI-Chatbot ระบบแชทบอทที่สนทนาได้เหมือนคนของ Deeple, AI-Assisted Robotic Surgery ระบบผู้เชี่ยวชาญด้านศัลยกรรม รวมทั้งการพัฒนาในเชิง AI-Personalization Platform เช่น Personalized Content, Personalized Advisory, Personalized Investment Planning เป็นต้น

Fundamental of DEEP LEARNING in Practice เหมาะจะใช้เพื่อศึกษาระบวนการพัฒนา Model แบบ Deep Learning ผู้อ่านจะได้ทดลองสร้างและเทรน Model ได้ลงนำ Model ไปใช้งานด้วยการ Deploy บน Production Environment และได้ลองสร้างระบบแนะนำหนังสือแบบง่ายๆ ในบทสุดท้าย จึงเป็นหนังสือที่ผู้อ่านจะได้เรียนรู้การพัฒนาครบทั้ง 3 องค์ประกอบ ได้แก่ Machine Learning, AI Capabilities และ AI Applications ซึ่งจะเป็นพื้นฐานสำหรับศึกษาต่อในระดับ Advance ต่อไป

ภิรพล คุชาเจริญ

บรรณาธิการ

CONTENTS

CHAPTER

01

ทำความเข้าใจกับ AI พื้นฐาน Machine Learning และการเตรียม Environment 1

ARTIFICIAL INTELLIGENCE VS MACHINE

LEARNING VS DEEP LEARNING 2

เปรียบเทียบ TRADITIONAL PROGRAMMING

กับ MACHINE LEARNING 3

การแบ่งประเภทของ MACHINE LEARNING.... 4

1. SUPERVISED LEARNING 5

2. UNSUPERVISED LEARNING 5

3. REINFORCEMENT LEARNING 6

รู้จัก AI TOOLBOXES และการเตรียม

ENVIRONMENT..... 7

การเตรียม ENVIRONMENT บน WINDOWS 9

ขั้นตอนการติดตั้ง CUDA TOOLKIT

(กรณีนีมี GPU)..... 9

ขั้นตอนการติดตั้ง cuDNN (กรณีนีมี GPU).... 12

ขั้นตอนการติดตั้ง MINICONDA

(ทั้งกรณีที่มีและไม่มี GPU) 16

ขั้นตอนการติดตั้ง TENSORFLOW และ

LIBRARY อื่นๆ..... 19

ขั้นตอนการทดสอบ NVIDIA GPU 23

การเตรียม Environment บน macOS 24

ขั้นตอนการติดตั้งบน APPLE M1 GPU 24

ขั้นตอนการทดสอบการทำงานของ GPU 31

ขั้นตอนการติดตั้งบน macOS ที่ไม่มี GPU ... 33

สรุปท้ายบท 39

CHAPTER

02

MACHINE LEARNING PIPELINE 40

ศึกษาแนวคิดพื้นฐานของ DEEP LEARNING

และ NEURAL NETWORK 41

การแบ่งประเภทของ NEURAL NETWORK 41

1. PERCEPTRON (P)..... 41

2. CONVOLUTIONAL NEURAL

NETWORK (CNN) 42

3. RECURRENT NEURAL

NETWORK (RNN) 42

การปรับค่า ACTIVATE, LOSS FUNCTION

และ OPTIMIZER ในการพัฒนา

DEEP LEARNING MODEL..... 43

การจำแนกภาพหลายมือจาก MNIST ด้วย CNN

MODEL ที่พัฒนาจาก KERAS TENSORFLOW 46

เตรียมความพร้อมก่อนทำ WORKSHOP 47

ขั้นตอนการสร้าง NOTEBOOK และ

IMPORT PACKAGE..... 47

ขั้นตอนการเพิ่มโค้ดเพื่อทดสอบ

การใช้งาน GPU 47

ขั้นตอนการติดตั้ง Library อื่นๆ 49

กระบวนการพัฒนา MODEL

แบบ DEEP LEARNING 50

STEP 1 DATA PREPARATION 50

STEP 2 DEFINE MODEL 55

STEP 3 COMPILE 56

STEP 4 FIT (TRAIN) 57

STEP 5 SAVE MODEL.....	58
STEP 6 LOAD MODEL.....	59
STEP 7 PREDICTION.....	62
STEP 8 EVALUATION.....	63

สรุปท้ายบท.....	66
-----------------	----

CHAPTER

03

การกำ FEATURE ENGINEERING ด้วย PANDAS 67

เทคนิคการทำ FEATURE ENGINEERING.....	68
วิธีทำความเข้าใจคุณภาพของข้อมูลด้วย PANDAS.....	68
ขั้นตอนการติดตั้ง PANDAS PROFILING LIBRARY.....	69
วิธีอ่านและสำรวจข้อมูลด้วย PANDAS PROFILING REPORT.....	70

การทำ FEATURE ENGINEERING ภาคปฏิบัติ.....	72
1. เทคนิค IMPUTATION.....	72
2. เทคนิค HANDLING OUTLIERS.....	76
2.1 DROP OUTLIER ด้วย STANDARD DEVIATION.....	78
2.2 DROP OUTLIER ด้วย PERCENTILE.....	82
3. เทคนิค BINNING.....	84
4. เทคนิค LOG TRANSFORM.....	85
5. เทคนิค ONE-HOT ENCODING.....	87
สรุปท้ายบท.....	88

CHAPTER

04

การอินพลิเมนต์ BACK PROPAGATION ALGORITHM ด้วย NUMPY 89

การทำ FORWARD & BACKWARD PROPAGATION ใน PERCEPTRON NEURAL NETWORK.....	90
FORWARD PROPAGATION.....	91
ACTIVATION FUNCTION.....	93
LOSS FUNCTION.....	94
BACKWARD PROPAGATION.....	94
ทดลองอินพลิเมนต์ NEURAL NETWORK ด้วย NUMPY.....	96
สรุปท้ายบท.....	102

CHAPTER

05

การใช้ความชันจากการสุ่มแบ่งข้อมูลฝึก เพื่อลดค่า LOSS ในการสอน NEURAL NETWORK MODEL ด้วย TENSORFLOW และ KERAS 103

วิธีการ GRADIENT DESCENT (GD) คืออะไร....	104
GRADIENT DESCENT METHOD.....	105
ทดลองเทรนโมเดลแบบ LINEAR REGRESSION ด้วย TENSORFLOW.....	106
STOCHASTIC GRADIENT DESCENT METHOD.....	118

CONTENTS

ทดลองเทรนโมเดลแบบ LINEAR REGRESSION ด้วย KERAS.....	118
สรุปท้ายบท.....	123

CHAPTER

06

วิธีปรับค่า LEARNING RATE และ MOMENTUM เพื่อเพิ่มประสิทธิภาพให้ MACHINE LEARNING MODEL 124

เข้าใจผลลัพธ์ที่เกิดจากการปรับค่า LEARNING RATE และ MOMENTUM.....	125
ทดลองปรับ LEARNING RATE แบบพื้นฐาน ด้วยตนเอง	126
เทคนิค MOMENTUM.....	130
ทดลอง TRAIN MODEL ด้วยเทคนิค MOMENTUM.....	130
เทคนิค LEARNING RATE DECAY	132
ทดลอง TRAIN MODEL ด้วยเทคนิค LEARNING RATE	132
เทคนิค DROP LEARNING RATE ON PLATEAU.....	135
ทดลอง TRAIN MODEL ด้วยเทคนิค DROP LEARNING RATE ON PLATEAU	136
เทคนิค ADAPTIVE LEARNING RATE ALGORITHM (ปรับค่า LEARNING RATE แบบอัตโนมัติ).....	141

ทดลอง TRAIN MODEL ด้วยเทคนิค ADAPTIVE LEARNING RATES ALGORITHM.....	141
สรุปท้ายบท.....	143

CHAPTER

07

การวิเคราะห์ประสิทธิภาพ MACHINE LEARNING MODEL ด้วย LEARNING CURVE 144

การวัดประสิทธิภาพโมเดลด้วย LEARNING CURVE.....	145
UNDERFIT LEARNING CURVE.....	145
OVERFIT LEARNING CURVE.....	152
GOOD FIT LEARNING CURVE.....	156
UNREPRESENTATIVE TRAIN DATASET	160
UNREPRESENTATIVE VALIDATION DATASET	164
สถานการณ์ที่ 1 (Validation Dataset น้อย และไม่สามารถเป็นตัวแทนของ Validation Dataset ได้).....	165
สถานการณ์ที่ 2 (Validation Dataset น้อย และง่ายเกินไป)	169
สรุปท้ายบท.....	173

CHAPTER

08

การทำ REGULARIZATION แบบสมัยใหม่ ด้วย AUGMENTATION, BATCH NORMALIZATION และ DROPOUT 174

การเตรียมและสอน MODEL ด้วย

FASHION-MNIST DATASET	176
การเตรียม DATASET สำหรับ TRAIN และ VALIDATION	177
การ TRAIN MODEL ด้วย FASHION-MNIST DATASET	180

ทดลองแก้ปัญหา OVERFITTING ของ
NEURAL NETWORK ด้วยเทคนิค

IMAGE AUGMENTATION	185
การเตรียมภาพสำหรับทำ IMAGE AUGMENTATION	186
ทดลองทำ IMAGE AUGMENTATION ด้วยการ VERTICAL SHIFT	186
ทดลองทำ IMAGE AUGMENTATION ด้วยการ HORIZONTAL SHIFT	187
ทดลองทำ IMAGE AUGMENTATION ด้วยการ SHEAR	188
ทดลองทำ IMAGE AUGMENTATION ด้วยการ ZOOM	189
ทดลองทำ IMAGE AUGMENTATION ด้วยการ VERTICAL FLIP	189
ทดลองทำ IMAGE AUGMENTATION ด้วยการ HORIZONTAL FLIP	190

ทดลองทำ IMAGE AUGMENTATION

ด้วยการ ROTATE 191

ทดลองทำ IMAGE AUGMENTATION

ด้วยการ FILL MODE 191

การ FILL MODE แบบ CONSTANT VALUES..... 192

การ FILL MODE แบบ NEAREST NEIGHBOR... 192

การ FILL MODE แบบ REFLECT VALUES..... 193

การ FILL MODE แบบ WRAP VALUES..... 194

การเทรน MODEL ด้วย FASHION-MNIST

IMAGE AUGMENTATION 194

ทดลองแก้ปัญหา OVERFITTING ของ
NEURAL NETWORK ด้วยเทคนิค

BATCH NORMALIZATION..... 199

ทดลองทำ BATCH NORMALIZATION200

ทดลองแก้ปัญหา OVERFITTING ของ NEURAL

NETWORK ด้วยเทคนิค DROPOUT205

ทดลองทำ DROPOUT206

สรุปท้ายบท 210

CHAPTER

09

การ VISUALIZING KERNELS และ FEATURE MAPS ใน DEEP LEARNING MODEL (CNN) 211

เข้าใจแนวคิดของ CONVOLUTION.....212

การจัดการกับ INPUT IMAGE แบบ

1 CHANNEL ด้วยการทำให้ PADDING.....213

CONTENTS

การจัดการกับ INPUT IMAGE แบบ	
1 CHANNEL ด้วยการทำ STRIDING	214
การจัดการกับ INPUT IMAGE แบบ	
1 CHANNEL ด้วยการทำ POOLING	214
การจัดการกับ INPUT IMAGE แบบ	
MULTI-CHANNEL.....	215
ทดลองทำ VISUALIZING CNN	
กับภาพตัวอย่าง.....	217
ขั้นตอนการ IMPORT LIBRARY	217
ขั้นตอนการเตรียมภาพตัวอย่าง	218
ขั้นตอนการสร้างโมเดลแบบ 2D CNN	
LAYER	218
การ VISUALIZING DEEP LEARNING MODEL	
ด้วย FASHION-MNIST	229
ขั้นตอนการ VISUALIZING KERNEL.....	236
ขั้นตอนการ VISUALIZING FEATURE MAP.....	238
สรุปท้ายบท	245

CHAPTER

10

การเลือกใช้ LOSS FUNCTION ในการ TRAIN DEEP LEARNING MODEL ตอนที่ 1 246

การเลือกใช้ LOSS FUNCTION ที่เหมาะสม	
สำหรับการ TRAIN MODEL.....	247
การใช้ LOSS FUNCTION เทรน	
REGRESSION MODEL	248
MEAN SQUARED ERROR LOSS (MSE).....	248

ทดลอง TRAIN MODEL REGRESSION	
ด้วย MSE	249
MEAN SQUARED LOGARITHMIC ERROR	
LOSS (MSLE)	254
ทดลอง TRAIN MODEL แบบ REGRESSION	
ด้วย MSLE	254

การใช้ LOSS FUNCTION เทรน	
AUTOENCODER	264
MEAN ABSOLUTE ERROR LOSS (MAE)....	265
ทดลอง TRAIN MODEL แบบ	
AUTOENCODER ด้วย MAE.....	265
สรุปท้ายบท	272

CHAPTER

11

การเลือกใช้ LOSS FUNCTION ในการ TRAIN DEEP LEARNING MODEL ตอนที่ 2 274

แนวคิดของ INFORMATION, ENTROPY และ	
CROSS-ENTROPY	275
INFORMATION THEORY	275
ENTROPY.....	278
CROSS-ENTROPY	280
การใช้ LOSS FUNCTION สำหรับเทรน	
BINARY CLASSIFICATION MODEL.....	281
BINARY CROSS-ENTROPY LOSS.....	282

ทดลอง TRAIN MODEL	
แบบ BINARY CLASSIFICATION	
ด้วย BINARY CROSS-ENTROPY LOSS	288

การใช้ LOSS FUNCTION สำหรับเทรน	
MULTI-CLASS CLASSIFICATION MODEL....	293

CATEGORICAL CROSS-ENTROPY LOSS	294
ทดลอง TRAIN MODEL	
แบบ MULTI-CLASS CLASSIFICATION	
ด้วย CATEGORICAL CROSS-ENTROPY	
LOSS	297

SPARSE CATEGORICAL CROSS-ENTROPY	
LOSS	302

ทดลอง TRAIN MODEL	
แบบ MULTI-CLASS CLASSIFICATION	
ด้วย SPARSE CATEGORICAL	
CROSS-ENTROPY LOSS	302

สรุปท้ายบท	307
------------------	-----

CHAPTER

12

การประยุกต์ใช้ EVALUATION METRICS	
เพื่อหา MODEL ที่ใช้งานได้จริง	308

การประเมิน MODEL	
แบบ BINARY CLASSIFICATION	309
การประยุกต์ใช้ EVALUATION METRICS :	
CONFUSION MATRIX, RECALL และ	
PRECISION	310

ทดลอง TRAIN MODEL แบบ	
BINARY CLASSIFICATION	313

การประเมิน MODEL แบบ MULTI-CLASS CLAS-	
SIFICATION	325

ทดลอง TRAIN MODEL แบบ MULTI-CLASS	
CLASSIFICATION	325

สรุปท้ายบท	336
------------------	-----

CHAPTER

13

การ DEPLOY MACHINE LEARNING	
MODEL บน PRODUCTION ด้วย	
FASTAPI, UVICORN และ DOCKER	337

การเตรียม ENVIRONMENT เพื่อจำลอง	
PRODUCTION.....	338

สถาปัตยกรรมและรูปแบบการรับส่งข้อมูล	
จาก WEB APPLICATION	338

การเตรียม PROJECT STRUCTURE	
เพื่อจัดเก็บไฟล์ที่ต้อง DEPLOY	339

WORKSHOP การสร้าง ENVIRONMENT	
สำหรับ TRAIN และ SAVE MODEL.....	341

ทดลองสร้าง ENVIRONMENT ใหม่	341
ขั้นตอนการสร้าง CONDA ENVIRONMENT	
ใหม่	341

ขั้นตอนการ TRAIN และ SAVE MODEL...	343
------------------------------------	-----

ทดสอบการใช้เครื่องมือต่างๆ เพื่อเตรียม	
DEPLOY.....	352

CONTENTS

ขั้นตอนการใช้งาน FASTAPI ร่วมกับ UVICORN	352
ขั้นตอนการใช้งาน API DOCUMENTATION	355
ขั้นตอนการพิสูจน์ตัวตนด้วย BASIC AUTHEN	358
ทดลอง DEPLOYMENT และทดสอบ MODEL	361
ขั้นตอนการ DEPLOYMENT โดยใช้ DOCKER IMAGE	361
ขั้นตอนทดสอบ API REQUEST ผ่าน JUPYTER NOTEBOOK	363
ขั้นตอนทดสอบประสิทธิภาพด้วย LOCUST (LOAD TESTING)	363
สรุปท้ายบท	368

CHAPTER

14

การสอน AI เรียนรู้การเล่นเกมด้วย DEEP REINFORCEMENT LEARNING บน GOOGLE COLAB PRO 369

DeepMind ฝึกสอน AI อย่างไรจนเล่นเกม ชนะมนุษย์	370
แนวคิดหลักของ Q-TABLE LEARNING	372
ทำความเข้าใจ FROZENLAKE ENVIRONMENT ของเกม	372
ขั้นตอนการรัน FROZENLAKE ENVIRONMENT บน GOOGLE COLAB PRO	373

ขั้นตอนการสร้าง Q-TABLE ไว้ให้ AI Agent ใช้ในการตัดสินใจเลือก Action	376
ทดลองฝึกสอนการเล่นเกม PONG ด้วย DEEP Q-NETWORK	381
ขั้นตอนการสร้าง DEEP LEARNING MODEL แบบ CNN	382
ขั้นตอน SCREEN PIXEL PREPROCESSING แปลงภาพให้เหมาะกับการสอน	384
ขั้นตอนการ SAVE MODEL ไปเก็บบน GOOGLE DRIVE	386
ขั้นตอนการ RENDER OPENAI GYM บน GOOGLE COLAB	389
ขั้นตอน EXPLORATION STRATEGY เพื่อสุ่ม ACTION ที่มากพอเพื่อใช้ฝึกสอน	392
ขั้นตอนการ COLLECT DATASET ไว้ใช้ฝึกสอน AI AGENT	394
ขั้นตอนการฝึกสอน AI AGENT โดยใช้ DOUBLE DQN	395
ขั้นตอน PLAY THE GAME การแข่งขันระหว่าง PONG กับ BOT	402
ขั้นตอนการบันทึก STATE เพื่อเก็บทุกเหตุการณ์ ในการแข่งขัน	408
สรุปท้ายบท	413

CHAPTER

15

การ TRANSFER LEARNING ด้วย KERAS สำหรับ COMPUTER VISION APPLICATIONS 414

TRANSFER LEARNING คืออะไร	415
โครงสร้างทางสถาปัตยกรรมของ	
CNN Model	416
การนำ PRE-TRAINED MODEL	
ไปใช้งาน	418
การประยุกต์ใช้งานจริงด้าน	
COMPUTER VISION	418
การประยุกต์ใช้ด้วยวิธี CLASSIFIER.....	419
ขั้นตอนการทำ CLASSIFIER	419
การประยุกต์ใช้ด้วยวิธี IMAGE SEARCH....	427
ขั้นตอนการทำ IMAGE SEARCH.....	427
การประยุกต์ใช้ด้วยวิธี FINE-TUNING	
(CLASSIFIER).....	436
ขั้นตอนการทำ FINE-TUNING	
ด้วย STRATEGY S3	436
ขั้นตอนการทำ FINE-TUNING	
ด้วย STRATEGY S4	447
ขั้นตอนการทำ FINE-TUNING	
ด้วย STRATEGY S5	451
ขั้นตอนการทำ FINE-TUNING	
ด้วย STRATEGY S6	456
สรุปท้ายบท	462

CHAPTER

16

การทำระบบแนะนำหนังสือ (BOOK RECOMMENDATION WORKSHOP) 463

การทำ DATA PREPARATION สำหรับ	
TRAIN MODEL.....	464
การคัดกรองข้อมูล (FILTER DATA).....	464
ขั้นตอนการเตรียมข้อมูล	465
การแปลงข้อมูลให้เป็นชุดข้อมูล	
(TRANSFORM TO DATASET).....	477
ขั้นตอนการแปลงข้อมูล	477
การ TRAIN MODEL แบบ AUTOENCODER....	481
ขั้นตอนการนิยาม MODEL.....	482
ขั้นตอนการ COMPILE และ	
TRAIN MODEL.....	483
การแปลงผลการ PREDICT เป็น TABLE	
ลงใน SQLite DATABASE	486
การ QUERY เพื่อหา TOP 10 BOOK RATING....	489
สรุปท้ายบท	490

CHAPTER

01

ทำความรู้จักกับ AI
พื้นฐาน Machine
Learning และการเตรียม
Environment

ปัญญาประดิษฐ์ หรือ Artificial Intelligence (AI) เป็น Keyword ที่นับวันยังถูกนำมาพูดถึงมากขึ้น ในการพัฒนาระบบงาน และการพัฒนาผลิตภัณฑ์ต่างๆ จนเสมือนว่างานใดก็ตามที่ไม่มี AI เป็น ส่วนหนึ่งในนั้น มันจะกลายเป็นงานที่ไม่สามารถทัดเทียมคู่แข่งได้ นั่นทำให้ AI กลายเป็นศาสตร์ที่มี บทบาทสำคัญในการวิวัฒนาการของดิจิทัลเทคโนโลยี



ARTIFICIAL INTELLIGENCE VS MACHINE LEARNING VS DEEP LEARNING

Artificial Intelligence คือ ศาสตร์ในการสร้างให้คอมพิวเตอร์มีความฉลาด และการตัดสินใจที่เหมือนมนุษย์ โดยเทคนิคหนึ่งที่จะทำให้คอมพิวเตอร์มีความฉลาด และสามารถตัดสินใจได้ นั่นก็คือ การทำให้คอมพิวเตอร์สามารถเรียนรู้ได้เองจากคุณลักษณะของข้อมูล (Data) ที่นำมาสอน (Train) มัน โดยปราศจากการป้อนกฎหรือเขียน Program โดย Programmer ซึ่งเราเรียกวิธีการในลักษณะนี้ว่า

Machine Learning

Machine Learning จึงเป็นเทคนิคที่ถูกใช้สำหรับการแก้ไขปัญหาที่มีความซับซ้อนหรือเกี่ยวข้องกับความรู้สึก ซึ่งเป็นการยากที่จะเขียนเงื่อนไขให้ครอบคลุม

ขอยกตัวอย่างการที่มนุษย์สามารถแยกแยะข้อความได้ว่า เป็นเชิงบวก เชิงลบ หรือเป็นกลาง มนุษย์จะต้องเข้าใจบริบท ซึ่งต้องผ่านประสบการณ์มาพอสมควร โดยไม่มีใครสามารถบอกได้ว่าเรามีกฎในสมองที่ซับซ้อนในการแยกแยะความรู้สึกของบทสนทนานั้นอย่างไร ซึ่งหากคอมพิวเตอร์หรือ AI จะตัดสินใจข้อความเหล่านั้นได้อย่างมนุษย์ คอมพิวเตอร์ต้องใช้เวลาเรียนรู้จากข้อมูลเป็นจำนวนมาก พอ เหมือนเป็นการสะสมประสบการณ์ในการแยกแยะ

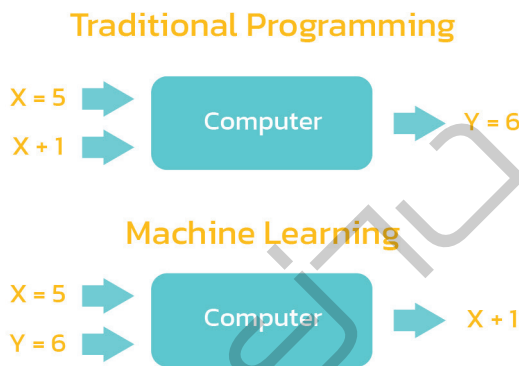
หลักๆ แล้วในการทำงานทางด้าน Machine Learning เราจะสร้างสิ่งที่เรียกว่า "Model" ที่ถูกสร้างขึ้นมาจากการ Train ด้วยชุดข้อมูล (Datasets) เพื่อให้ Machine เรียนรู้รูปแบบ และความสัมพันธ์ของข้อมูล แทนที่จะต้องเขียนกฎที่มีความซับซ้อนขึ้นมาเองเป็นจำนวนมาก

Deep Learning เป็นเทคนิคหนึ่งทาง Machine Learning ที่ประสบความสำเร็จอย่างสูงในปัจจุบัน โดย Deep Learning ถูกออกแบบโดยได้รับแรงบันดาลใจจากลักษณะของ Neuron Network (โครงข่ายประสาท) ในสมองมนุษย์ ซึ่งในหนังสือเล่มนี้เราจะได้เรียนรู้ และนำเทคนิคต่างๆ ทาง Deep Learning ไปแก้ปัญหาในงานด้าน AI

แต่ก่อนจะกล่าวถึง Deep Learning ที่ละเอียดขึ้น เราจะทำความเข้าใจพื้นฐานของ Machine Learning และการติดตั้งเครื่องมือที่จำเป็นในการพัฒนา Model กันก่อน

เปรียบเทียบ TRADITIONAL PROGRAMMING กับ MACHINE LEARNING

เพื่อให้ผู้อ่านเห็นภาพมากขึ้น จะขอเปรียบเทียบ Machine Learning กับ Traditional Programming ดังภาพต่อไปนี้



รูปที่ 1.1 Traditional Programming VS Machine Learning

ในการแก้ปัญหาแบบ Traditional Programming นั้น โปรแกรมเมอร์จะต้องทำความเข้าใจปัญหาและเงื่อนไขที่ Program จะตอบสนอง หรือให้ผลลัพธ์เมื่อมีการรับข้อมูลแบบต่างๆ เข้ามา แล้วจึงเขียน Program เช่น เมื่อต้องการแก้ปัญหการเพิ่มค่าตัวเลขแบบจำนวนเต็มขึ้นอีก 1 โปรแกรมเมอร์จะต้องออกแบบ Function ที่ทำให้ได้ผลลัพธ์ตามที่ต้องการ นั่นคือ ฟังก์ชัน $f(x) = x + 1$ แล้วจึงแปลงเป็น Program นำไปรันในคอมพิวเตอร์ ซึ่งเมื่อมีการรับเลขจำนวนเต็ม 5 มันจะบวกค่าอีก 1 ซึ่งได้ผลลัพธ์เท่ากับ 6

ในขณะที่การแก้ปัญหาแบบ Machine Learning เราจะนำ Data มาสอนโดยใช้เทคนิคอย่างเช่น Deep Learning ซึ่งในการสอน เราอาจต้องใส่ Input Data และผลเฉลยที่ต้องการ เช่น Input Data = 5 ผลเฉลยคือ 6

ด้วยการเรียนรู้จากจำนวนข้อมูลที่มากพอ มันจะสร้าง Model ที่คล้ายกับฟังก์ชัน $x + 1$ ซึ่งสามารถทำนายคำตอบที่ใกล้เคียงกับผลเฉลยที่ได้สอนมัน (Predictive Model)

เทคนิคทาง Machine Learning จะมีประโยชน์อย่างมากเมื่อปัญหาที่ต้องการแก้ เป็นปัญหาที่มนุษย์ต้องคิดกฎหรือเงื่อนไขจำนวนมากเพื่อให้ครอบคลุมเพียงพอ ตัวอย่างเช่น การจะบอกว่าภาพ

ที่รับเข้ามาเป็น **Duck** หรือ **Not Duck** ซึ่งเป็นงานที่ค่อนข้างยากเมื่อต้องเขียน Program แบบ Traditional Programming



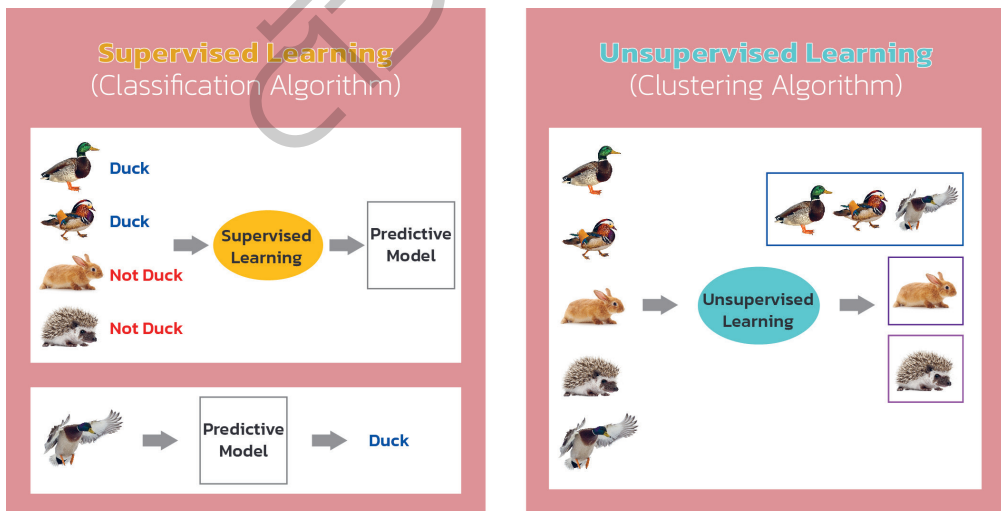
รูปที่ 1.2 การทำนายคำตอบโดยใช้เทคนิคทาง Machine Learning

ภาพดัดแปลงจาก : <https://blog.westerndigital.com/machine-learning-pipeline-object-storage>

การแบ่งประเภทของ MACHINE LEARNING

เราสามารถแบ่งประเภทของ Machine Learning ออกเป็น 3 ประเภทคือ

1. Supervised Learning
2. Unsupervised Learning
3. Reinforcement Learning

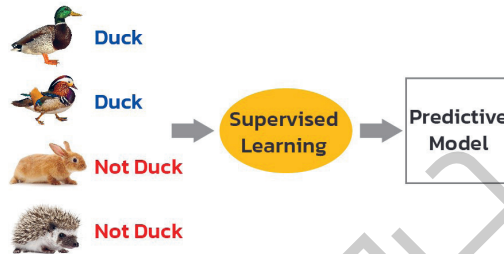


รูปที่ 1.3 Supervised Learning และ Unsupervised Learning

ภาพดัดแปลงจาก : <https://blog.westerndigital.com/machine-learning-pipeline-object-storage/>

1. SUPERVISED LEARNING

ในการใช้เทคนิคทาง Machine Learning เพื่อแยกประเภทภาพเปิด เราเพียงนำภาพเปิดและภาพสัตว์อื่นๆ พร้อมผลเฉลย (Label Data) มาฝึกสอนเพื่อ Predictive Model ซึ่งเราเรียกวิธีการเรียนรู้แบบนี้ว่า Supervised Learning

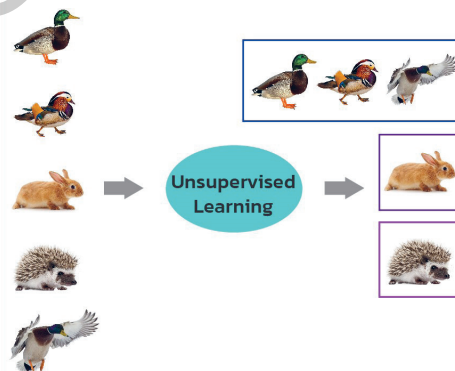


รูปที่ 1.4 Supervised Learning

ภาพดัดแปลงจาก : <https://blog.westerndigital.com/machine-learning-pipeline-object-storage/>

2. UNSUPERVISED LEARNING

ในทางกลับกัน การเรียนรู้แบบ Unsupervised Learning เราจะใช้เพียง Input Data โดยไม่จำเป็นต้องใส่ผลเฉลย ซึ่งคำตอบที่ได้จากการเรียนรู้แบบนี้คือ รูปแบบของข้อมูลที่อาจบอกได้ว่าภาพสัตว์ทั้งหมดที่มีจะแบ่งออกเป็น 3 กลุ่ม ได้แก่ ภาพเปิด ภาพกระต่าย และภาพตัวเม่น เป็นต้น



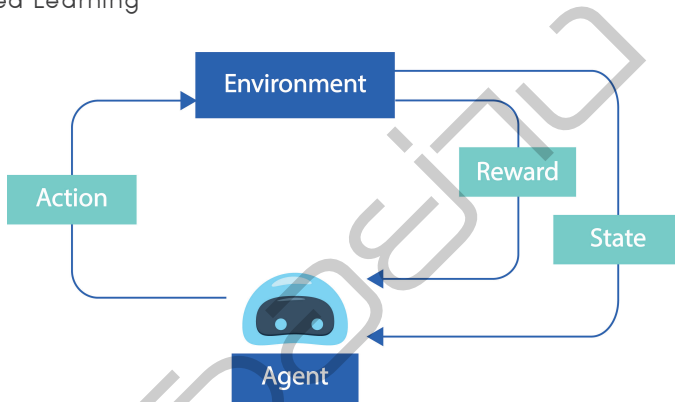
รูปที่ 1.5 Unsupervised Learning

ภาพดัดแปลงจาก : <https://blog.westerndigital.com/machine-learning-pipeline-object-storage/>

3. REINFORCEMENT LEARNING

ในขณะที่ประเภทของ Machine Learning ประเภทสุดท้ายคือ Reinforcement Learning จะมีการเรียนรู้วิธีการแก้ปัญหาจากสถานะ (State) รางวัล (Reward) และการกระทำ (Action) ของ Agent AI ต่อระบบ เช่น AI ที่เอาชนะแชมป์โกะ (Go) ชาวเกาหลีใต้ในปี ค.ศ. 2016 (AlphaGo ของ Google DeepMind)

ซึ่งก่อนที่ AI จะมีความสามารถในการตัดสินใจในแต่ละสถานการณ์ (State) ที่ดีที่สุด ในช่วงแรกมันจะต้องลองผิดลองถูก (สุ่มเลือก Action) เพื่อเก็บข้อมูลเป็นจำนวนมากพอ เช่นเดียวกับวิธีการแบบ Supervised Learning



รูปที่ 1.6 Reinforcement Learning



รูปที่ 1.7 Machine Learning ที่พัฒนาโดย Google DeepMind ใช้เทคนิค Deep Reinforcement Learning
เครดิตวิดีโอ : YouTube, Match 1 - Google DeepMind Challenge Match: Lee Sedol vs AlphaGo

รู้จัก AI TOOLBOXES และการเตรียม ENVIRONMENT

ก่อนที่จะเตรียม Environment สำหรับการฝึกโมเดล เราจะมาทำความรู้จักกับกระบวนการหรือขั้นตอน เพื่อแก้ปัญหาในงานทางด้าน AI/Machine Learning กันก่อน ซึ่งในงานด้าน AI/Machine Learning จะมีขั้นตอนที่ต้องทำ ดังนี้

STEP 1 Business Understanding

STEP 2 Data Acquisition

STEP 3 Data Preparation

STEP 4 Exploratory Data Analysis

STEP 5 Data Modeling

STEP 6 Model Evaluation

STEP 7 Model Deployment

โดยในการลงมือทำตามตัวอย่างในเล่ม เราจะใช้ Python, Jupyter Notebook และ Package/Library ต่างๆ เช่น TensorFlow, Scikit-learn ฯลฯ เป็นเครื่องมือหลักสำหรับการทำงานในขั้นตอนตั้งแต่การทำ Data Acquisition ไปจนถึง Data Deployment (STEP 2-7)

การลง Library ใหม่ เราสามารถใช้ Conda ซึ่งเป็นเครื่องมือหลักที่มาพร้อมกับ Anaconda สำหรับการติดตั้ง Library และสร้าง Environment เฉพาะในการรันโปรแกรมโดยไม่ให้ Library ใหม่ๆ ไปปะปนกับส่วนอื่นภายในเครื่องคอมพิวเตอร์ แต่ถ้าใครยังไม่ได้ติดตั้ง Anaconda ผู้เขียนแนะนำให้ติดตั้ง Miniconda แทน เนื่องจากเราต้องการ Feature เพียงบางอย่างในการทำงานเท่านั้น เพื่อจะรันโปรแกรมบน Environment เฉพาะ ไม่ให้ Package ใหม่ๆ ไปปะปนกับส่วนอื่นภายในเครื่องคอมพิวเตอร์ เราสามารถใช้ Conda จาก Miniconda ในการสร้าง Environment ใหม่

TensorFlow เป็นหนึ่งใน Deep Learning Framework ยอดนิยมที่ใช้ในการพัฒนางานทางด้าน AI และ Data Science ที่สามารถเพิ่มความเร็วในการประมวลผลได้ด้วย NVIDIA GPU, CUDA Toolkit และ cuDNN ซึ่งเป็นส่วนเสริมของ CUDA ซึ่งในหนังสือเล่มนี้ เราจะติดตั้ง CUDA 11.0, cuDNN 8.0 และ Python 3.8 ที่ Compatible กับ TensorFlow 2.4 ตามตาราง

GPU

Version	Python version	Compiler	Build tools	cuDNN	CUDA
tensorflow-2.4.0	3.6-3.8	GCC 7.3.1	Bazel 3.1.0	8.0	11.0
tensorflow-2.3.0	3.5-3.8	GCC 7.3.1	Bazel 3.1.0	7.6	10.1
tensorflow-2.2.0	3.5-3.8	GCC 7.3.1	Bazel 2.0.0	7.6	10.1
tensorflow-2.1.0	2.7, 3.5-3.7	GCC 7.3.1	Bazel 0.27.1	7.6	10.1
tensorflow-2.0.0	2.7, 3.3-3.7	GCC 7.3.1	Bazel 0.26.1	7.4	10.0
tensorflow_gpu-1.15.0	2.7, 3.3-3.7	GCC 7.3.1	Bazel 0.26.1	7.4	10.0
tensorflow_gpu-1.14.0	2.7, 3.3-3.7	GCC 4.8	Bazel 0.24.1	7.4	10.0
tensorflow_gpu-1.13.1	2.7, 3.3-3.7	GCC 4.8	Bazel 0.19.2	7.4	10.0
tensorflow_gpu-1.12.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.15.0	7	9
tensorflow_gpu-1.11.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.15.0	7	9
tensorflow_gpu-1.10.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.15.0	7	9
tensorflow_gpu-1.9.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.11.0	7	9
tensorflow_gpu-1.8.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.10.0	7	9
tensorflow_gpu-1.7.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.9.0	7	9
tensorflow_gpu-1.6.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.9.0	7	9
tensorflow_gpu-1.5.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.8.0	7	9
tensorflow_gpu-1.4.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.5.4	6	8
tensorflow_gpu-1.3.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.4.5	6	8
tensorflow_gpu-1.2.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.4.5	5.1	8
tensorflow_gpu-1.1.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.4.2	5.1	8
tensorflow_gpu-1.0.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.4.2	5.1	8

รูปที่ 1.8 Tested Build Configurations

เครดิตตารางข้อมูล : <https://www.tensorflow.org/install/source#gpu>

เพื่อให้ท่านผู้อ่านทั้งที่มี GPU และไม่มี GPU สามารถฝึกปฏิบัติได้ รูปแบบของ Environment ที่เราจะเตรียมคือ

■ การเตรียม Environment บน Windows

- การติดตั้งบน NVIDIA GPU
- การติดตั้งบน Windows ที่ไม่มี GPU

■ การเตรียม Environment บน macOS

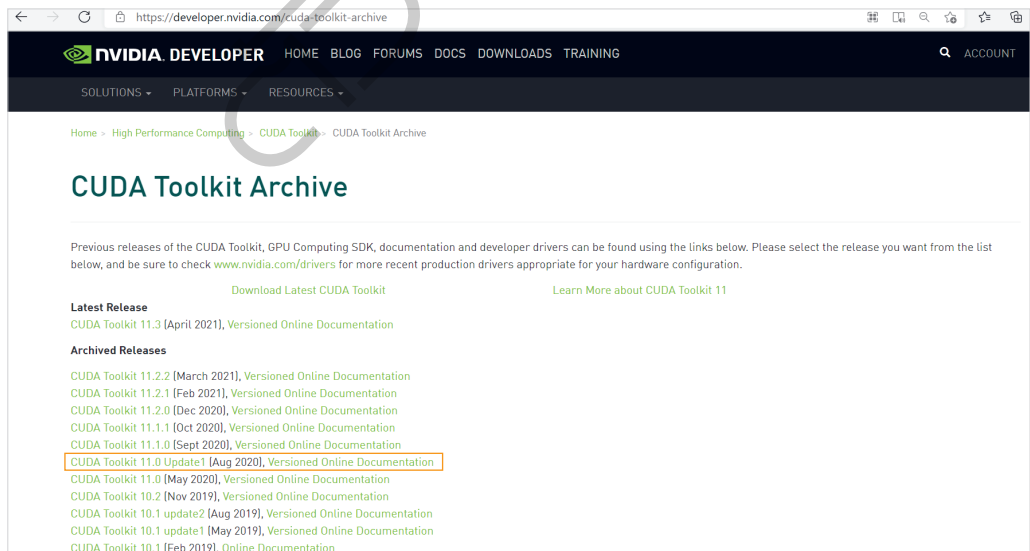
- การติดตั้งบน Apple M1 GPU
- การติดตั้งบน macOS ที่ไม่มี GPU

การเตรียม ENVIRONMENT บน WINDOWS

ขั้นตอนการติดตั้ง CUDA TOOLKIT (กรณีมี GPU)

สำหรับการติดตั้งบน Windows ที่ไม่มี GPU ให้ข้ามไปที่หัวข้อ ขั้นตอนการติดตั้ง Miniconda ได้เลย

1. ดาวน์โหลด **CUDA Toolkit 11.0 Update1** จากเว็บไซต์ <https://developer.nvidia.com/cuda-toolkit-archive>



2. เมื่อเราเลือก CUDA Version แล้วจะปรากฏหน้าจอเพื่อเลือก Platform

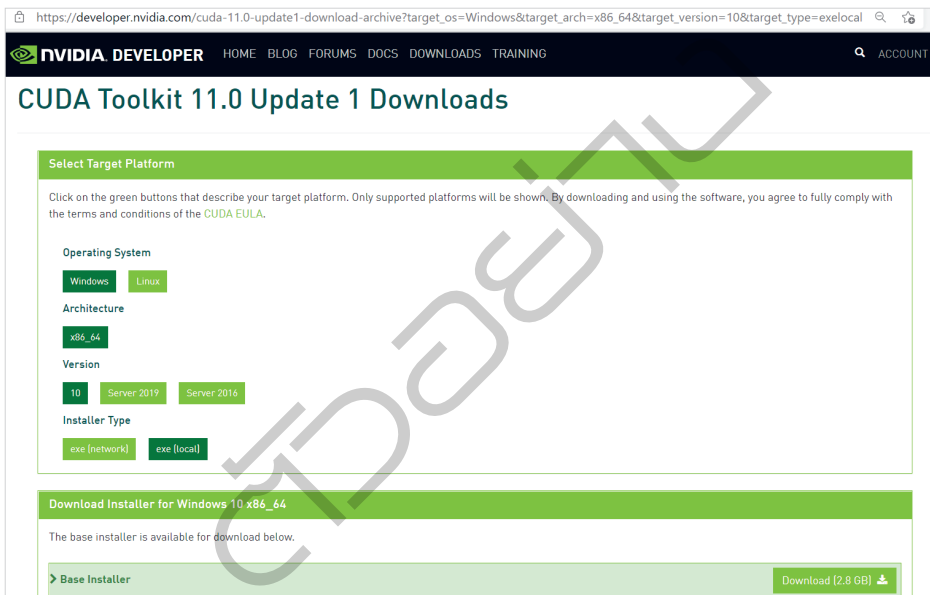
2.1 คลิกเลือก Operating System : **Windows**

2.2 คลิกเลือก Architecture : **x86_64**

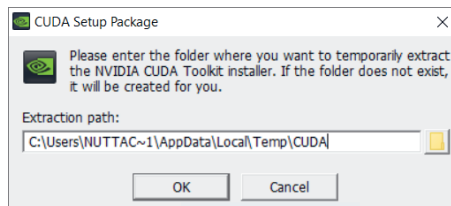
2.3 คลิกเลือก Version : **10**

2.4 คลิกเลือก Install Type : **exe (local)**

2.5 คลิกปุ่ม **Download (2.8 GB)**



3. ดับเบิลคลิกที่ไฟล์เมื่อดาวน์โหลดสำเร็จแล้ว คลิกปุ่ม OK

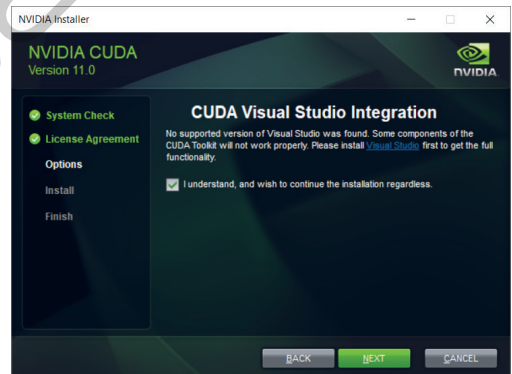
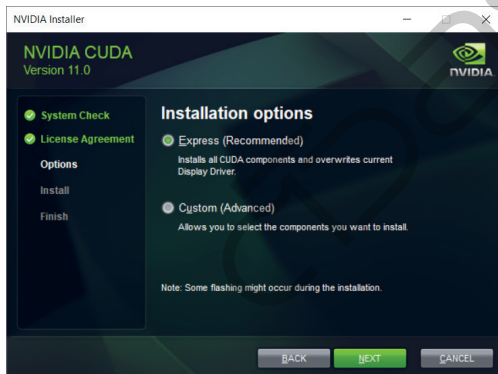


4. ที่หน้า NVIDIA software license agreement คลิกปุ่ม **AGREE AND CONTINUE**



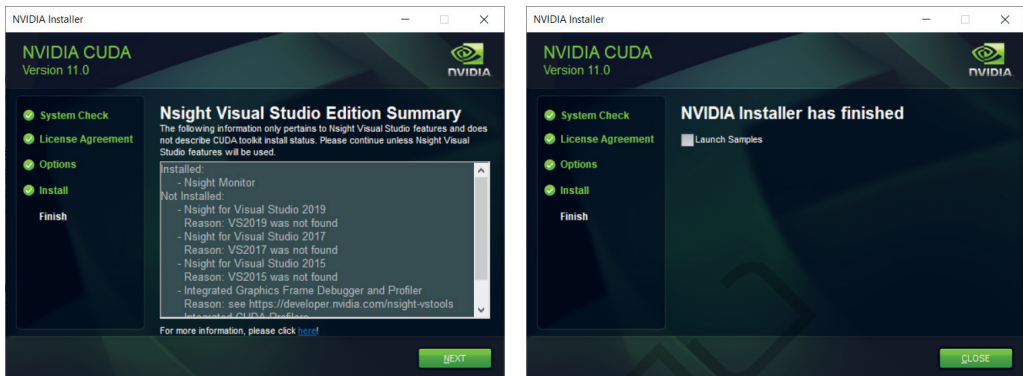
5. ที่หน้า Installation options คลิกเลือก **Express** แล้วคลิกปุ่ม **NEXT**

6. ที่หน้า CUDA Visual Studio Integration คลิกเลือก **I Understand...** แล้วคลิกปุ่ม **NEXT**



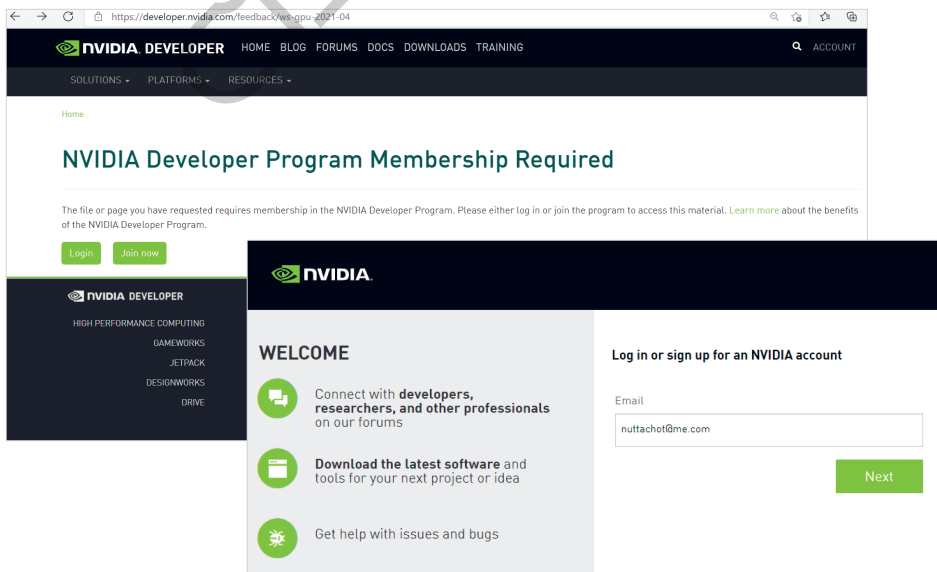
7. ที่หน้า **Nsight Visual Studio Edition Summary** คลิกปุ่ม **NEXT**

8. ที่หน้า **NVIDIA Installer has finished** คลิกปุ่ม **CLOSE**



ขั้นตอนการติดตั้ง cuDNN (กรณีมี GPU)

1. ดาวน์โหลด **cuDNN** จากเว็บไซต์ <https://developer.nvidia.com/feedback/wr-gpu-2021-04> โดยคลิกปุ่ม **Login** (ถ้ายังไม่เคยลงทะเบียน)
2. ป้อน **Email Address** ของตัวเอง แล้วคลิกปุ่ม **Next**



3. คลิกที่ **Create account**

4. ป้อนข้อมูลให้ครบ ยืนยันว่าเป็นมนุษย์ด้วย **hCaptcha** แล้วคลิกปุ่ม **CREATE ACCOUNT**

5. ไปที่ Mailbox ของตัวเอง แล้วคลิกที่ Link เพื่อ **Verified Email**

6. แล้วคลิกปุ่ม **SUBMIT**

Almost done!

Please confirm the information below to complete registration.

Recommendation Settings

☐ Yes, recommend content that I might enjoy based on how I engage with NVIDIA's websites, software, and events.

Be the first to learn about new SDKs, developer tools and training

☐ Send me the latest developer news, announcements, and more from NVIDIA. I can unsubscribe at any time.

We promise to protect your privacy. We never sell your data. You can change your settings anytime at privacy.nvidia.com.

SUBMIT

7. ป้อนข้อมูลส่วนตัวโดยละเอียด แล้วคลิกปุ่ม **Submit**



NVIDIA | Sign Up

☒ Link Account ☒ Verify Email **3** Enter Details

First Name*

Nuttachot

Last Name*

Promrit

Areas(s) of Interest*

☒ AI / Deep Learning
☐ AR / VR
☐ Autonomous Machines
☐ Autonomous Vehicles
☒ Data Science
☐ Design & Visualization
☐ Game Development
☐ HPC
☐ IoT / Smart Cities
☐ Networking

Gender ⓘ

Male ▼

Location*

Thailand ▼

Organization*

cpsu

Organization URL

https://www.cp.su.ac.th

Industry Segment*

Higher Education / Research ▼

☒ Join the **NVIDIA Developer Program** to access downloads (like cuDNN), how-to videos, and more.

By clicking "Submit", you agree to these [terms and conditions](#).

Submit

8. ทำแบบสำรวจ แล้วคลิกปุ่ม Submit

cuDNN Workstation GPU Survey

Your feedback helps us make products you need.
Which GPUs are you using? (select all that apply)

- ☐ A100 in the cloud
- ☐ A100 HGX on premise
- ☐ A100 PCIe on premise
- ☐ V100 in the cloud
- ☐ V100 HGX on premise
- ☐ V100 PCIe on premise
- ☒ Other...

GeForce RTX 2060

Would you be interested in using an NVIDIA A100 in a workstation or desktop if it was available?

☒ Yes
☐ No

Can you provide more insight about your last answer?

-

What are the key attribute(s) you would like to see improved in your next GPU? (select all that apply)

- ☒ Memory capacity
- ☐ Memory bandwidth
- ☒ FP64 FLOPs
- ☐ FP32 FLOPs
- ☒ Tensor cores
- ☒ CUDA cores
- ☐ Multi-GPU instances
- ☐ Encoders
- ☐ Decoders
- ☐ Other...

Please provide an email if NVIDIA can contact you about your feedback

nuttachot@me.com

Submit

9. เลือก cuDNN v8.0.5 สำหรับ CUDA 11.0 เพื่อติดตั้งบน Windows

cuDNN Archive | NVIDIA Developer

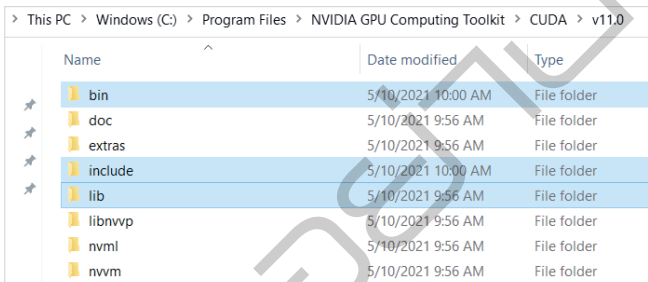
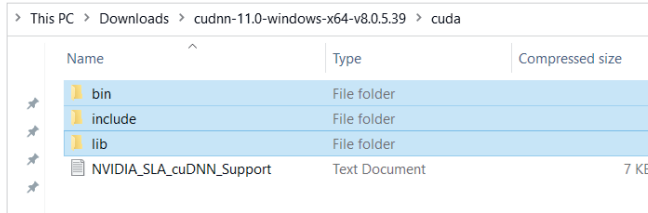
Download cuDNN v8.0.5 (November 9th, 2020), for CUDA 11.1

Download cuDNN v8.0.5 (November 9th, 2020), for CUDA 11.0

Library for Windows and Linux, Ubuntu(x86_64 & PPC architecture)

- cuDNN Library for Linux (x86_64)
- cuDNN Library for Linux (PPC)
- cuDNN Library for Windows (x86)
- cuDNN Runtime Library for Ubuntu20.04 x86_64 (Deb)
- cuDNN Developer Library for Ubuntu20.04 x86_64 (Deb)
- cuDNN Code Samples and User Guide for Ubuntu20.04 x86_64 (Deb)
- cuDNN Runtime Library for Ubuntu18.04 x86_64 (Deb)
- cuDNN Developer Library for Ubuntu18.04 x86_64 (Deb)
- cuDNN Code Samples and User Guide for Ubuntu18.04 x86_64 (Deb)
- cuDNN Runtime Library for Ubuntu16.04 x86_64 (Deb)
- cuDNN Developer Library for Ubuntu16.04 x86_64 (Deb)

10. Unzip ไฟล์ แล้วคัดลอกโฟลเดอร์ **bin**, **include** และ **lib** ไปวางที่ C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v11.0



ขั้นตอนการติดตั้ง MINICONDA (ทั้งกรณีที่มีและไม่มี GPU)

1. ดาวน์โหลดและติดตั้ง **Miniconda** บน Windows สำหรับ Python Version ล่าสุดจากเว็บไซต์ <https://docs.conda.io/en/latest/miniconda.html>

Miniconda

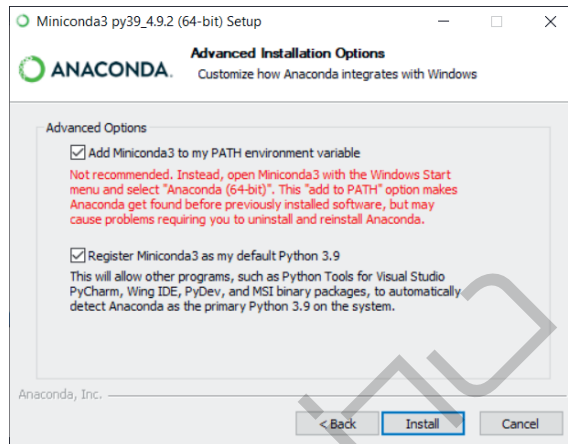
Miniconda is a free minimal installer for conda. It is a small, bootstrap version of Anaconda that includes only conda, Python, the packages they depend on, and a small number of other useful packages, including pip, zlib and a few others. Use the `conda install` command to install 720+ additional conda packages from the Anaconda repository.

See if Miniconda is right for you.

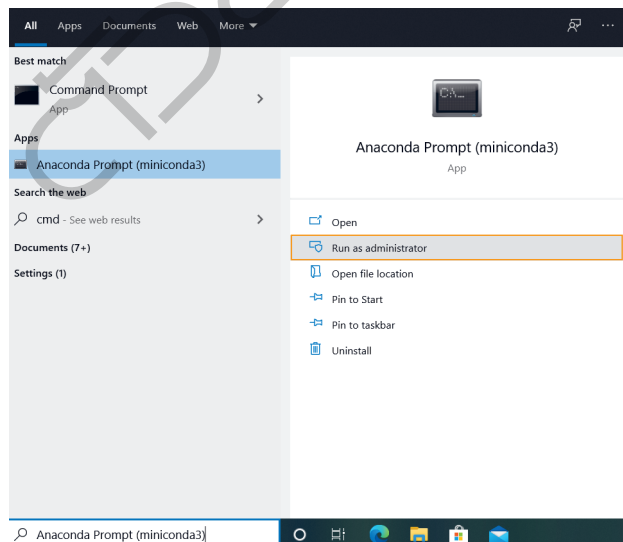
Windows installers

Python version	Name	Size	SHA256 hash
Python 3.9	Miniconda3 Windows 64-bit	57.7 MiB	c3a83b0dc4c4fa92454d0fa036c1803a045d875d7602b33a773b04b3f4c208
	Miniconda3 Windows 32-bit	54.9 MiB	5b85f0d6-448f-b0a2-2185-4202-7d380b4d1a8739c1a50838c3b0208f231a046
Python 3.8	Miniconda3 Windows 64-bit	57.0 MiB	4fa220a047ba0505048b3b043145373a09f5331c8120899ac1080f927c43
	Miniconda3 Windows 32-bit	54.2 MiB	9c2af70ba97246c85c28c733ca30f41f0bda03f9b0a2a51f4a081c974b0c661
Python 2.7	Miniconda2 Windows 64-bit	54.1 MiB	6973825404832544a074f92b4dc405a080eeed47070513ba8f0bda4d420c
	Miniconda2 Windows 32-bit	47.7 MiB	c084026f0b00f046579c5a409a0721f4a13f4a0b23ba6a64159443703617b

- เมื่อถึงหน้า **Advanced Installation Options** คลิกเลือก **Add Miniconda3 to my PATH environment variable** แล้วคลิกปุ่ม **Install**



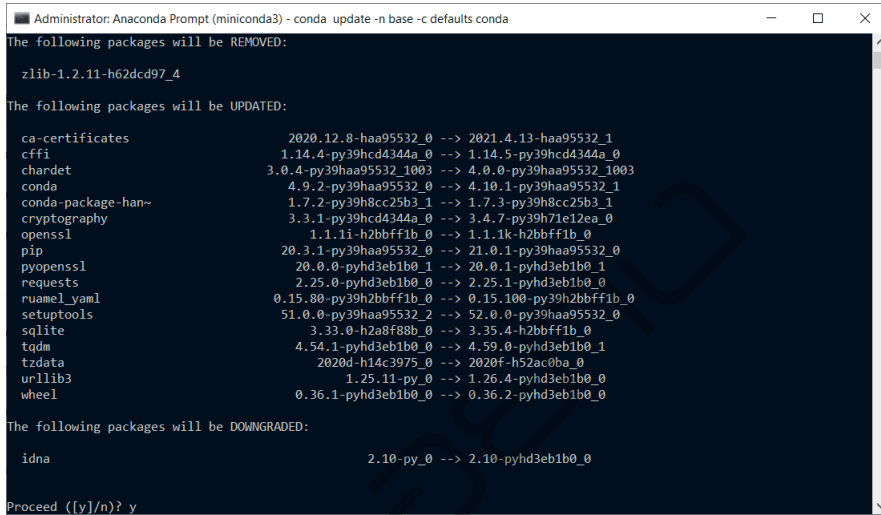
- ไปที่ **Apps : Anaconda Prompt (miniconda3)** คลิกเลือก **Run as administrator** รันด้วยสิทธิ์แอดมิน



4. แล้วอัปเดต Conda เป็นรุ่นล่าสุดด้วยคำสั่ง **conda update** และพารามิเตอร์ ดังนี้

```
conda update -n base -c defaults conda
```

5. เมื่อถึงหน้าจอที่แสดง packages ที่จะ update พิมพ์ y แล้วกดปุ่ม **Enter**



```
Administrator: Anaconda Prompt (miniconda3) - conda update -n base -c defaults conda

The following packages will be REMOVED:

zlib-1.2.11-h62dcd97_4

The following packages will be UPDATED:

ca-certificates          2020.12.8-haa95532_0 --> 2021.4.13-haa95532_1
cffi                    1.14.4-py39hcd4344a_0 --> 1.14.5-py39hcd4344a_0
charset                 3.0.4-py39haa95532_1003 --> 4.0.0-py39haa95532_1003
conda                   4.9.2-py39haa95532_0 --> 4.10.1-py39haa95532_1
conda-package-handl~  1.7.2-py39h8cc25b3_1 --> 1.7.3-py39h8cc25b3_1
cryptography            3.3.1-py39hcd4344a_0 --> 3.4.7-py39h71e12ea_0
openssl                 1.1.1i-h2bbff1b_0 --> 1.1.1k-h2bbff1b_0
pip                     20.3.1-py39haa95532_0 --> 21.0.1-py39haa95532_0
pyopenssl               20.0.0-pyhd3eb1b0_1 --> 20.0.1-pyhd3eb1b0_1
requests                2.25.0-pyhd3eb1b0_0 --> 2.25.1-pyhd3eb1b0_0
ruamel_yaml             0.15.80-py39h2bbff1b_0 --> 0.15.100-py39h2bbff1b_0
setuptools              51.0.0-py39haa95532_2 --> 52.0.0-py39haa95532_0
sqlite                  3.33.0-h2a8f88b_0 --> 3.35.4-h2bbff1b_0
tqdm                    4.54.1-pyhd3eb1b0_0 --> 4.59.0-pyhd3eb1b0_1
tzdata                  2020d-h14c3975_0 --> 2020f-h52ac0ba_0
urllib3                 1.25.11-py_0 --> 1.26.4-pyhd3eb1b0_0
wheel                   0.36.1-pyhd3eb1b0_0 --> 0.36.2-pyhd3eb1b0_0

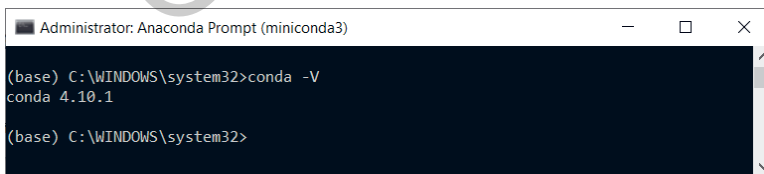
The following packages will be DOWNGRADED:

idna                    2.10-py_0 --> 2.10-pyhd3eb1b0_0

Proceed ([y]/n)? y
```

6. เรียกดู Version ของ Conda ด้วยคำสั่ง **conda** และพารามิเตอร์ ดังนี้

```
conda -V
```



```
Administrator: Anaconda Prompt (miniconda3)

(base) C:\WINDOWS\system32>conda -V
conda 4.10.1

(base) C:\WINDOWS\system32>
```

7. สร้าง Environment ใหม่ โดยตั้งชื่อเป็น **"deep_workshop1"** สำหรับรัน Python Version 3.8 และติดตั้ง Jupyter Notebook ด้วยคำสั่ง **conda create** และพารามิเตอร์ ดังนี้

```
conda create -n deep_workshop1 python=3.8 jupyter
```

```
Administrator: Anaconda Prompt (miniconda3) - conda create -n deep_workshop1 python=3.8 jupyter

terminado          pkgs/main/win-64::terminado-0.9.4-py38haa95532_0
testpath           pkgs/main/noarch::testpath-0.4.4-pyhd3eb1b0_0
tornado            pkgs/main/win-64::tornado-6.1-py38h2bfff1b_0
traitlets           pkgs/main/noarch::traitlets-5.0.5-pyhd3eb1b0_0
vc                 pkgs/main/win-64::vc-14.2-h21ff451_1
vs2015_runtime      pkgs/main/win-64::vs2015_runtime-14.27.29016-h5e58377_2
wcwidth            pkgs/main/noarch::wcwidth-0.2.5-py_0
webencodings        pkgs/main/win-64::webencodings-0.5.1-py38_1
wheel              pkgs/main/noarch::wheel-0.36.2-pyhd3eb1b0_0
widgetsnbextension pkgs/main/win-64::widgetsnbextension-3.5.1-py38_0
wincertstore        pkgs/main/win-64::wincertstore-0.2-py38_0
winpty             pkgs/main/win-64::winpty-0.4.3-4
zeromq             pkgs/main/win-64::zeromq-4.3.3-ha925a31_3
zipp               pkgs/main/noarch::zipp-3.4.1-pyhd3eb1b0_0
zlib               pkgs/main/win-64::zlib-1.2.11-h62dc97_4

Proceed ([y]/n)? y
```

8. เข้าใช้ Environment ใหม่ โดยพิมพ์คำสั่ง **conda activate** ตามด้วยชื่อ Environment : **deep_workshop1**

```
conda activate deep_workshop1
```

```
Select Administrator: Anaconda Prompt (miniconda3)

(base) C:\WINDOWS\system32>conda activate deep_workshop1

(deep_workshop1) C:\WINDOWS\system32>
```

9. เรียกดู Version ของ Python ด้วยคำสั่ง **python**

```
python
```

10. กดคีย์ลัด **Ctrl** + **Z** แล้วกดปุ่ม **Enter** เพื่อออกมายัง Command Prompt

ขั้นตอนการติดตั้ง TENSORFLOW และ LIBRARY อื่นๆ

1. ติดตั้ง **TensorFlow 2.4** ด้วยคำสั่ง **pip install**

```
pip install tensorflow==2.4
```

2. เรียกดู Version ของ TensorFlow ด้วยคำสั่ง **python** แล้วกดปุ่ม **Enter** แล้วพิมพ์คำสั่งต่อไปนี้

```
import tensorflow as tf
tf.__version__
```

```
Administrator: Anaconda Prompt (miniconda3) - python
(deep_workshop1) C:\WINDOWS\system32>python
Python 3.8.8 (default, Apr 13 2021, 15:08:03) [MSC v.1916 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import tensorflow as tf
2021-05-10 23:19:36.975399: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library cudart64_110.dll
>>> tf.__version__
'2.4.0'
>>>
```

3. กดคีย์ลัด **Ctrl** + **Z** แล้วกดปุ่ม **Enter** เพื่อออกมายัง Command Prompt

4. ติดตั้ง Scikit-learn ด้วยคำสั่ง **conda install**

```
conda install scikit-learn
```

5. สร้าง Project ชื่อ **"deepbasic_pj"** ด้วยคำสั่ง **mkdir deepbasic_pj** แล้วไปยัง Folder ดังกล่าว

```
mkdir deepbasic_pj
cd deepbasic_pj
```

```
Administrator: Anaconda Prompt (miniconda3)
(deep_workshop1) C:\WINDOWS\system32>cd c:\
(deep_workshop1) c:\>mkdir deepbasic_pj
(deep_workshop1) c:\>cd deepbasic_pj
(deep_workshop1) c:\deepbasic_pj>
```

6. เปิด **Jupyter Notebook** ด้วยคำสั่ง

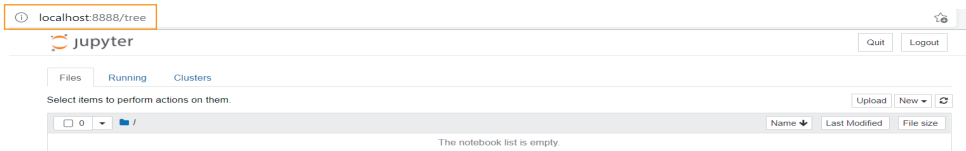
```
jupyter notebook
```

7. ที่ Anaconda Prompt จะแสดง URL ที่เปิด Jupyter Notebook และ Logs ต่างๆ

```
Administrator: Anaconda Prompt (miniconda3) - jupyter notebook
[1 23:27:03.642 NotebookApp] Serving notebooks from local directory: c:\deepbasic_pj
[1 23:27:03.643 NotebookApp] Jupyter Notebook 6.3.0 is running at:
[1 23:27:03.643 NotebookApp] http://localhost:8888/?token=e758adb044a2197549387e9f8a45419be873dffc827b2f2e
[1 23:27:03.643 NotebookApp] or http://127.0.0.1:8888/?token=e758adb044a2197549387e9f8a45419be873dffc827b2f2e
[1 23:27:03.643 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 23:27:03.682 NotebookApp]

To access the notebook, open this file in a browser:
file:///C:/Users/nuttachot/AppData/Roaming/jupyter/runtime/nbsrvr-6280-open.html
Or copy and paste one of these URLs:
http://localhost:8888/?token=e758adb044a2197549387e9f8a45419be873dffc827b2f2e
or http://127.0.0.1:8888/?token=e758adb044a2197549387e9f8a45419be873dffc827b2f2e
```

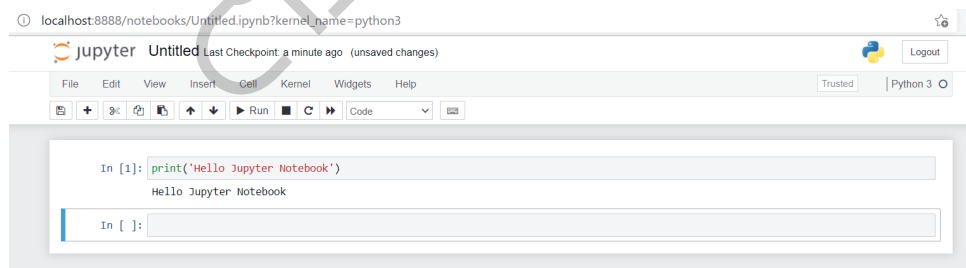
8. Jupyter Notebook จะถูก Run ขึ้นโดย Default URL จะเป็น **localhost:8888** และหน้าหลักของ Project ที่เราเปิดขึ้นมาจะอยู่ที่ **/tree**



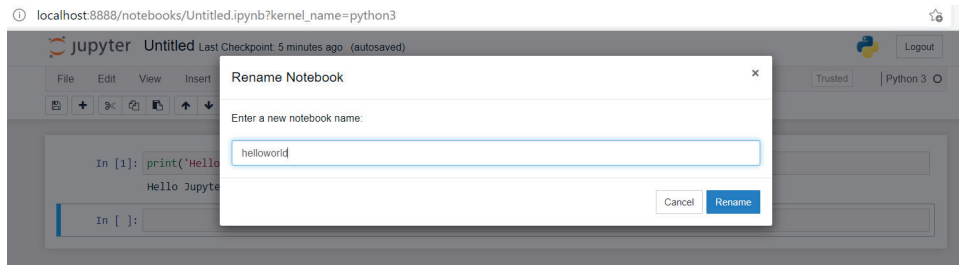
9. สร้าง Notebook โดยคลิกที่ **New > Python 3**



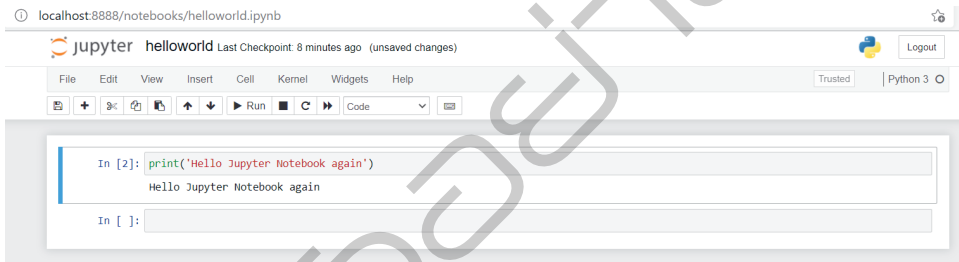
10. พิมพ์ **print('Hello Jupyter Notebook')** ใน Cell แรก แล้วกดคีย์ลัด **Shift + Enter** เพื่อรันโปรแกรม และสร้าง Cell ใหม่ไปพร้อมกัน



11. เปลี่ยนชื่อ Notebook โดยคลิกที่ **Untitled** แล้วพิมพ์ชื่อไฟล์ที่ต้องการ แล้วคลิกปุ่ม **Rename**



12. กลับไปแก้ไข Code ใน Cell แรกเป็น **print('Hello Jupyter Notebook again')** แล้วกดคีย์ลัด **Ctrl + Enter** เพื่อรันโปรแกรมโดยไม่มีการสร้าง Cell ใหม่



ขั้นตอนการทดสอบ NVIDIA GPU

1. แสดงรุ่นของ **NVIDIA GPU** ด้วยคำสั่ง **nvidia-smi** ตามด้วยพารามิเตอร์ ดังนี้

```
!nvidia-smi -L
```

```
In [3]: !nvidia-smi -L
```

```
GPU 0: NVIDIA GeForce RTX 2060 (UUID: GPU-4305db38-4ba6-02b7-6948-cae88b6b3ce7)
```

2. แสดง **Version** ของ **TensorFlow** ด้วยคำสั่ง ดังนี้

```
import tensorflow as tf
import tensorflow.python.platform.build_info as build

tf.__version__
```

```
In [4]: import tensorflow as tf
import tensorflow.python.platform.build_info as build
tf.__version__
```

```
Out[4]: '2.4.0'
```

3. แสดงจำนวน **GPU** และชื่อ **Device** ด้วยคำสั่ง ดังนี้

```
print("Num GPUs Available: ", len(tf.config.experimental.list_physical_devices('GPU')))
```

```
print(f'Default GPU Device:{tf.test.gpu_device_name()}')
```

```
In [5]: print("Num GPUs Available: ", len(tf.config.experimental.list_physical_devices('GPU')))
```

```
print(f'Default GPU Device:{tf.test.gpu_device_name()}')
```

```
Num GPUs Available: 1
Default GPU Device:/device:GPU:0
```

4. แสดง **Version** ของ **CUDA** และ **cuDNN** ด้วยคำสั่ง ดังนี้

```
print(build.build_info['cuda_version'])
print(build.build_info['cudnn_version'])
```

```
In [6]: print(build.build_info['cuda_version'])
print(build.build_info['cudnn_version'])
```

```
64_110
64_8
```

การเตรียม Environment บน macOS

ขั้นตอนการติดตั้งบน APPLE M1 GPU

TensorFlow เป็นหนึ่งใน Deep Learning Framework ยอดนิยมนที่สามารถเพิ่มความเร็วในการประมวลผลได้ด้วย NVIDIA GPU, CUDA Toolkit และ cuDNN แต่น่าเสียดายที่ Apple ได้หยุดให้การสนับสนุน GPU ของ NVIDIA มาระยะเวลาหนึ่งแล้ว

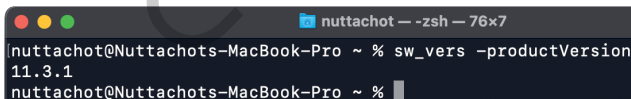
จนกระทั่งในยุคสมัยของ Apple Silicon เราจึงสามารถใช้งาน TensorFlow 2.4 เวอร์ชัน Fork ที่ Apple ได้ปรับแต่งให้สามารถทำงานได้ดีขึ้นบนเครื่องคอมพิวเตอร์ที่ใช้ชิป M1 (รวมทั้งรุ่นที่ใช้ CPU Intel) ทำให้เราสามารถดึงประสิทธิภาพ CPU และ GPU 8 Core ของ M1 ออกมาได้มากขึ้น

Requirements : เพื่อจะติดตั้ง TensorFlow เวอร์ชัน Fork เราจะต้องเตรียมซอฟต์แวร์ต่างๆ ให้พร้อม ดังนี้

- macOS 11.0+
- Xcode Command Line Tools
- Python 3.8

1. ตรวจสอบ **Version** ของ **macOS** ด้วยคำสั่ง **sw_vers** ตามด้วยพารามิเตอร์ ดังนี้

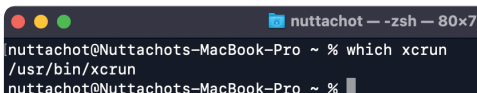
```
sw_vers -productVersion
```



```
nuttachot ~ % sw_vers -productVersion
11.3.1
nuttachot ~ %
```

2. ตรวจสอบ **Xcode Command Line Tools** ด้วยคำสั่ง **which xcrun** ตามด้วยพารามิเตอร์ ดังนี้

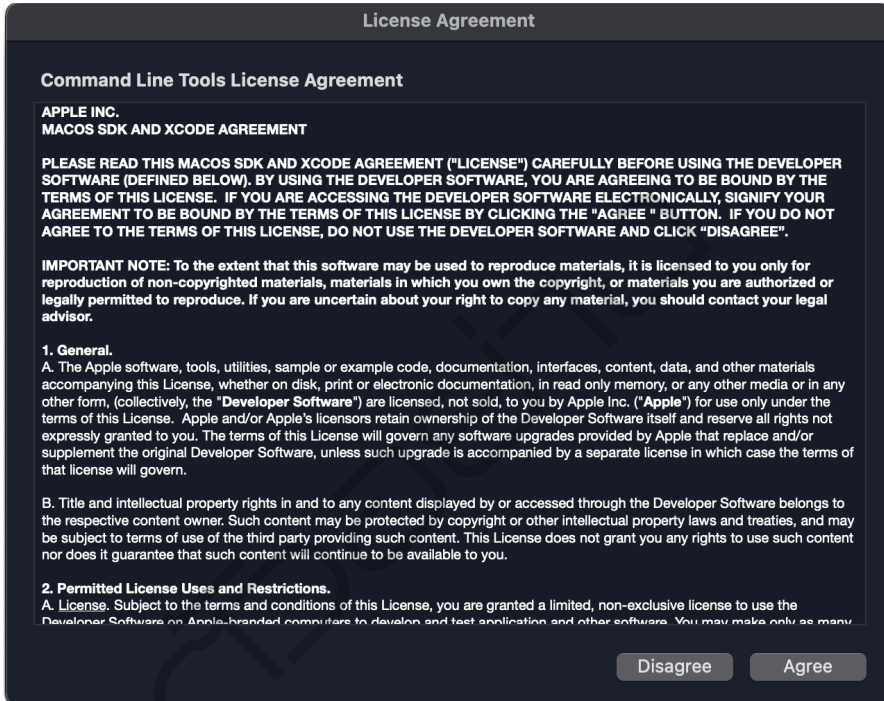
```
which xcrun
```



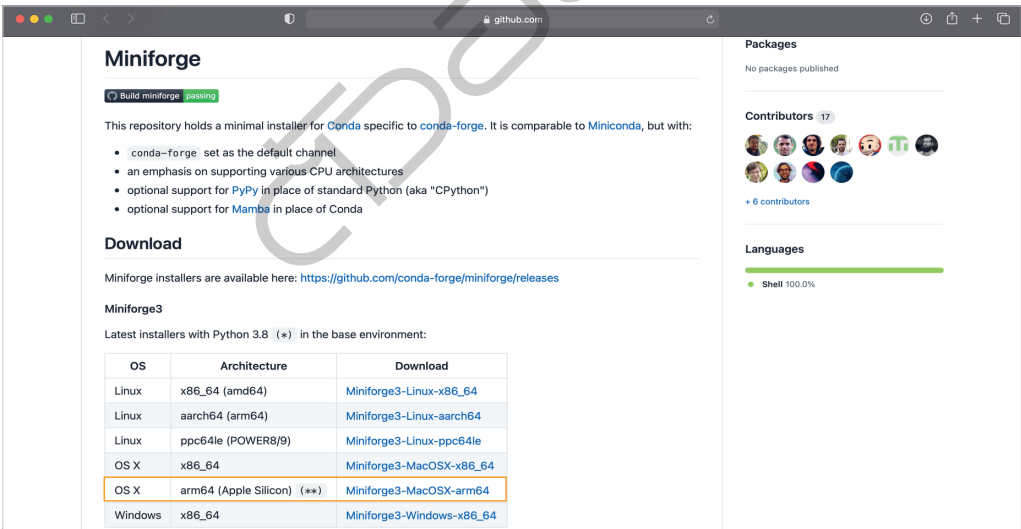
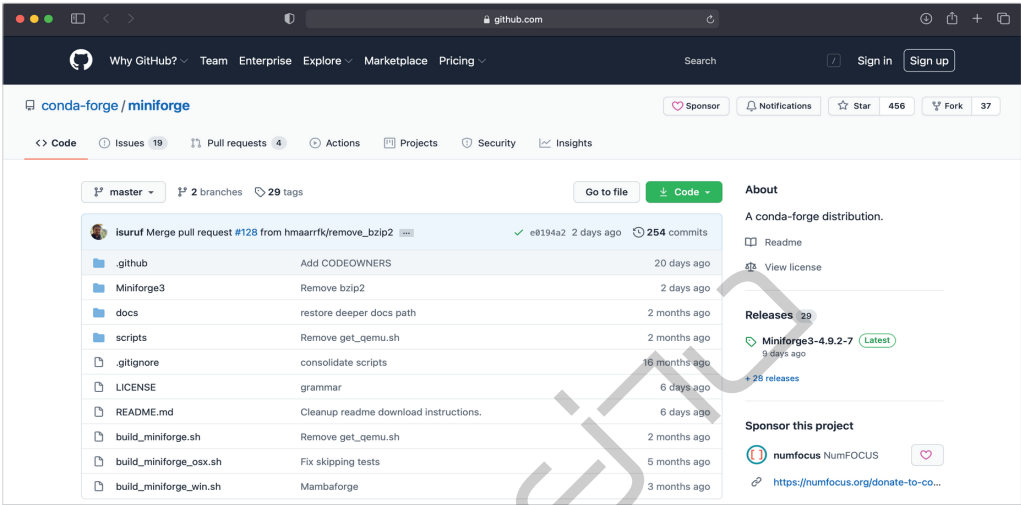
```
nuttachot ~ % which xcrun
/usr/bin/xcrun
nuttachot ~ %
```

3. ถ้าใน Terminal ของเราไม่แสดง Path: `/usr/bin/xcrun` ตามภาพด้านบน จะต้องมีการติดตั้ง **Xcode Command Line Tools** ก่อน ด้วยคำสั่ง ดังนี้

```
xcode-select --install
```

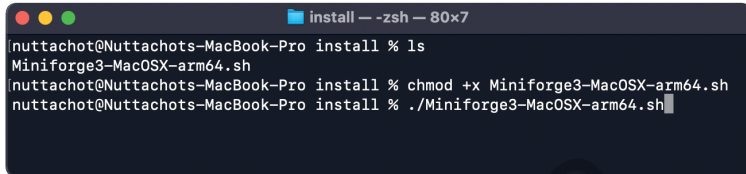


4. สำหรับ Python 3.8 เราจะติดตั้งมันด้วย Conda จาก Miniforge (https://github.com/conda-forge/miniforge#miniforge3)



5. เลือกดาวน์โหลด **Miniforge** ที่ติดตั้งแบบ **arm64 (Apple Silicon)** แล้วรันโปรแกรมด้วยคำสั่ง ดังนี้

```
chmod +x Miniforge3-MacOSX-arm64.sh
./Miniforge3-MacOSX-arm64.sh
```

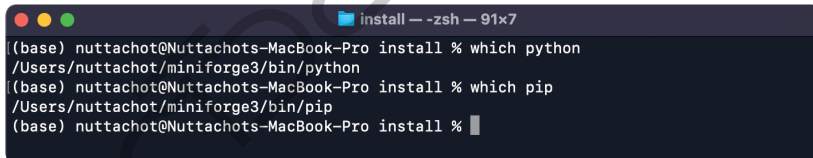


```
install — zsh — 80x7
nuttachot@Nuttachots-MacBook-Pro install % ls
Miniforge3-MacOSX-arm64.sh
nuttachot@Nuttachots-MacBook-Pro install % chmod +x Miniforge3-MacOSX-arm64.sh
nuttachot@Nuttachots-MacBook-Pro install % ./Miniforge3-MacOSX-arm64.sh
```

6. ปิดแล้วเปิด Terminal ขึ้นมาใหม่ เมื่อติดตั้งเสร็จแล้ว เราควรได้ผลลัพธ์คล้ายภาพด้านล่าง ขณะที่ใช้คำสั่ง **which python** และ **which pip** เพื่อตรวจสอบ Path ของ Python และ pip

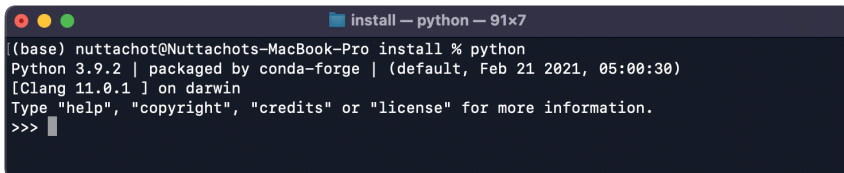
```
which python
```

```
which pip
```



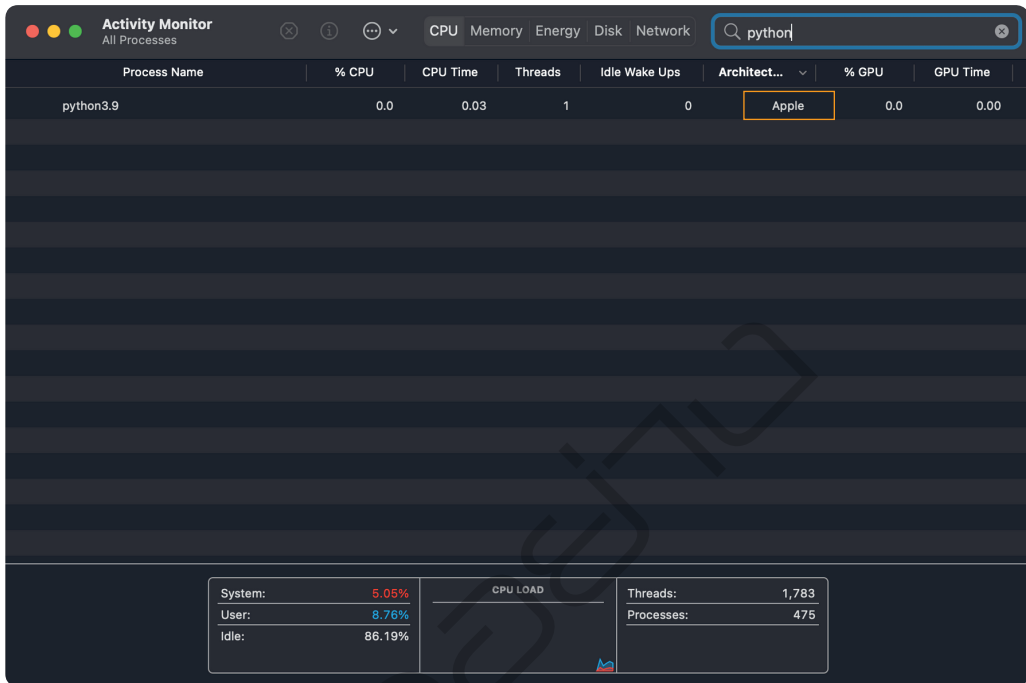
```
install — zsh — 91x7
(base) nuttachot@Nuttachots-MacBook-Pro install % which python
/Users/nuttachot/miniforge3/bin/python
(base) nuttachot@Nuttachots-MacBook-Pro install % which pip
/Users/nuttachot/miniforge3/bin/pip
(base) nuttachot@Nuttachots-MacBook-Pro install %
```

7. พิมพ์ **python** ใน **Terminal** และดู **Activity Monitor** เพื่อยืนยันว่ามันถูกรันอยู่บน Apple Architecture หรือไม่



```
install — python — 91x7
(base) nuttachot@Nuttachots-MacBook-Pro install % python
Python 3.9.2 | packaged by conda-forge | (default, Feb 21 2021, 05:00:30)
[Clang 11.0.1 ] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

8. Activity Monitor จะถูกเปิดขึ้น แสดงว่าเรากำลังรันอยู่บน Apple Architecture



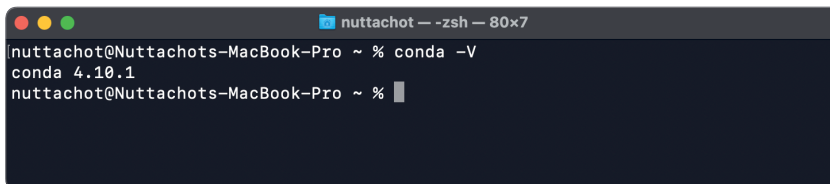
9. จะเห็นว่า Python ที่ติดมากับ Miniforge จะเป็น Python 3.9 ดังนั้น เราจะติดตั้ง Python 3.8 บน Environment ใหม่ด้วยคำสั่ง **conda env create**

10. ก่อนอื่น **Update Conda** ให้เป็น Version ล่าสุดด้วยคำสั่ง

```
conda update -n base -c defaults conda
```

11. เรียกดู Version ของ **Conda** ด้วยคำสั่ง

```
conda -V
```



12. เพื่อจะสร้าง Environment ใหม่ เราจะเตรียมรายการของ Library ที่ต้องติดตั้ง โดยบันทึกลงไฟล์ **environment.yml** ดังนี้

environment.yml

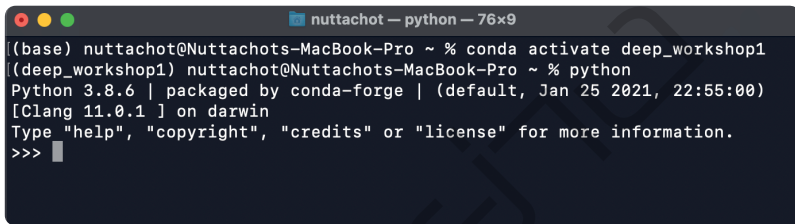
```
name: deep_workshop1
channels:
  - conda-forge
  - nodefaults
dependencies:
  - grpcio
  - h5py
  - ipython
  - numpy
  - pip
  - python=3.8.6
  - scipy
  - termcolor
  - typeguard
  - wheel
  - absl-py
  - astunparse
  - python-flatbuffers
  - gast
  - google-pasta
  - keras-preprocessing
  - opt_einsum
  - protobuf
  - tensorboard
  - tensorflow-estimator
  - termcolor
  - typing_extensions
  - wrapt
  - pandas
  - matplotlib
  - scikit-learn
  - jupyterlab
```

13. สร้าง Conda Environment ชื่อ `deep_workshop1` ด้วยคำสั่ง ดังนี้

```
conda env create --file=environment.yml
```

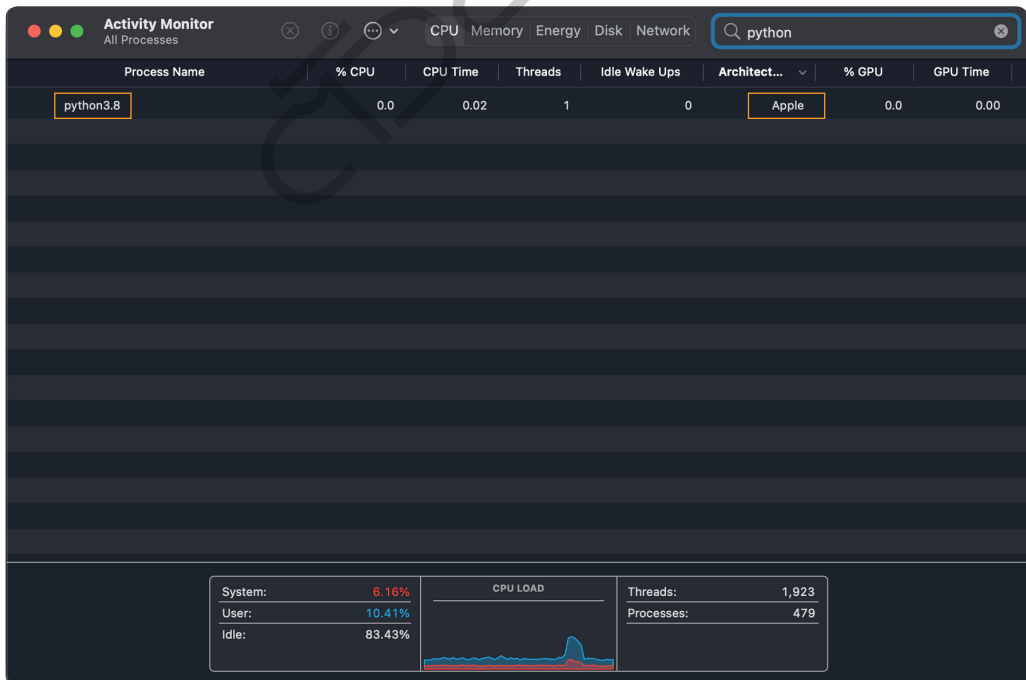
14. ไปที่ `deep_workshop1` Environment ด้วยคำสั่ง `conda activate deep_workshop1` แล้วพิมพ์ `python` ใน Terminal และดู Activity Monitor เพื่อยืนยันว่า Python 3.8 อยู่นอยู่บน Apple Architecture หรือไม่

```
conda activate deep_workshop1
```



```
nuttachot — python — 76x9
(base) nuttachot@Nuttachots-MacBook-Pro ~ % conda activate deep_workshop1
(deep_workshop1) nuttachot@Nuttachots-MacBook-Pro ~ % python
Python 3.8.6 | packaged by conda-forge | (default, Jan 25 2021, 22:55:00)
[Clang 11.0.1 ] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> 
```

15. Activity Monitor ที่แสดงให้เห็นว่า Python 3.8 อยู่นอยู่บน Apple Architecture

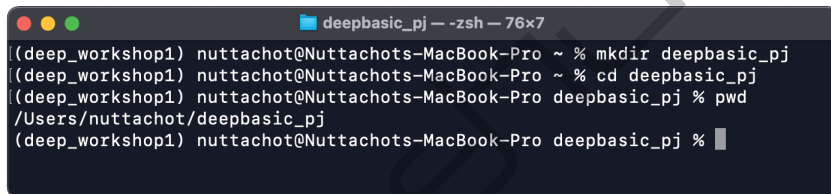


16. ติดตั้ง TensorFlow 2.4 ด้วยคำสั่ง pip Install

```
pip install --upgrade --force --no-dependencies https://github.com/apple/tensorflow_macos/releases/download/v0.1.alpha2/tensorflow_addons_macos-0.1a2-cp38-cp38-macosx_11_0_arm64.whl
```

ขั้นตอนการทดสอบการทำงานของ GPU

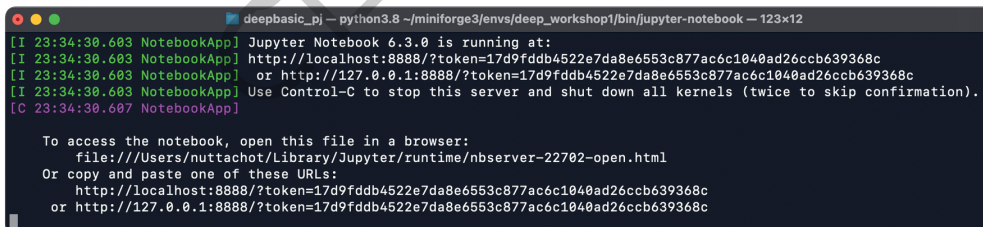
1. สร้าง Project ชื่อ **deepbasic_pj** ด้วยคำสั่ง **mkdir deepbasic_pj** แล้วไปยังโฟลเดอร์ดังกล่าว



```
deepbasic_pj — zsh — 76x7
(deep_workshop1) nuttachot@Nuttachots-MacBook-Pro ~ % mkdir deepbasic_pj
(deep_workshop1) nuttachot@Nuttachots-MacBook-Pro ~ % cd deepbasic_pj
(deep_workshop1) nuttachot@Nuttachots-MacBook-Pro deepbasic_pj % pwd
/Users/nuttachot/deepbasic_pj
(deep_workshop1) nuttachot@Nuttachots-MacBook-Pro deepbasic_pj %
```

2. เปิด Jupyter Notebook

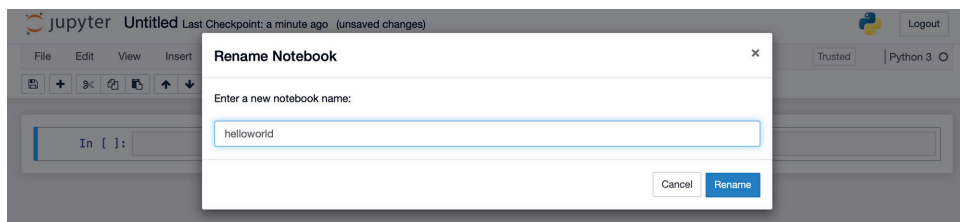
```
jupyter notebook
```



```
deepbasic_pj — python3.8 ~/miniforge3/envs/deep_workshop1/bin/jupyter-notebook — 123x12
[I 23:34:30.603 NotebookApp] Jupyter Notebook 6.3.0 is running at:
[I 23:34:30.603 NotebookApp] http://localhost:8888/?token=17d9fddb4522e7da8e6553c877ac6c1040ad26ccb639368c
[I 23:34:30.603 NotebookApp] or http://127.0.0.1:8888/?token=17d9fddb4522e7da8e6553c877ac6c1040ad26ccb639368c
[I 23:34:30.603 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 23:34:30.607 NotebookApp]

To access the notebook, open this file in a browser:
file:///Users/nuttachot/Library/Jupyter/runtime/nbserver-22702-open.html
Or copy and paste one of these URLs:
http://localhost:8888/?token=17d9fddb4522e7da8e6553c877ac6c1040ad26ccb639368c
or http://127.0.0.1:8888/?token=17d9fddb4522e7da8e6553c877ac6c1040ad26ccb639368c
```

3. สร้าง Notebook โดยคลิกที่ **New > Python 3** แล้วคลิกที่ **Untitled** พิมพ์ชื่อไฟล์ที่ต้องการ แล้วคลิกปุ่ม **Rename**



4. Import Library ที่ต้องใช้ด้วยคำสั่ง ดังนี้

```
import tensorflow as tf
from tensorflow.python.compiler.mlcompute import mlcompute
from tensorflow.python.framework.ops import disable_eager_execution
```

5. ตรวจสอบ TensorFlow Version

```
tf.__version__
```

ผลลัพธ์คือ '2.4.0-rc0'

6. ตรวจสอบการทำงานของ TensorFlow กับ Apple ML Compute

```
mlcompute.is_apple_mlc_enabled()
```

ผลลัพธ์คือ True

```
mlcompute.is_tf_compiled_with_apple_mlc()
```

ผลลัพธ์คือ True

7. Disable TensorFlow ในโหมด Eager Execution ที่ทำให้รันทีละคำสั่งได้เหมือนการรันโปรแกรมปกติ

```
disable_eager_execution()
tf.executing_eagerly()
```

ผลลัพธ์คือ False

8. คอนฟิกให้ Train Model บน MI GPU

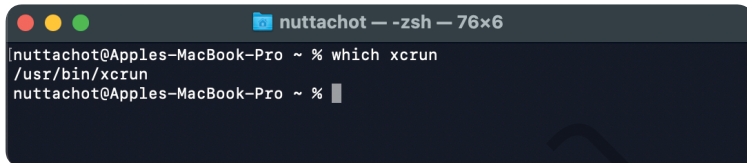
```
mlcompute.set_mlc_device(device_name='gpu')
```

ขั้นตอนการติดตั้งบน macOS ที่ไม่มี GPU

สำหรับ macOS เราจะต้องติดตั้ง Xcode Command Line Tool ต่างๆ ก่อน ดังนี้

1. ให้ตรวจสอบ **Xcode Command Line Tool** ด้วยคำสั่ง **which xcrun**

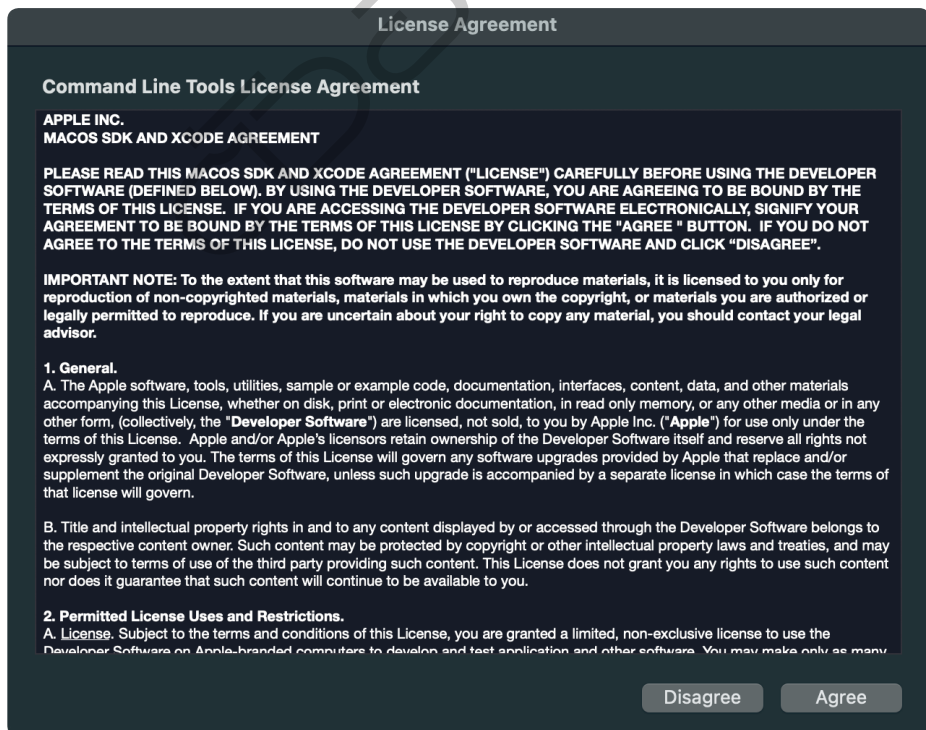
```
which xcrun
```



```
nuttachot@Apples-MacBook-Pro ~ % which xcrun
/usr/bin/xcrun
nuttachot@Apples-MacBook-Pro ~ %
```

2. ถ้าใน Terminal ของเราไม่แสดง Path : **/usr/bin/xcrun** ตามภาพข้อ 1 จะต้องมีการติดตั้ง Xcode Command Line Tool ก่อน ด้วยคำสั่ง **xcode-select --install** แล้วคลิกปุ่ม **Agree**

```
xcode-select --install
```



เบื้องต้นเราจะสร้าง Environment ชื่อ **deep_workshop1** สำหรับรัน Python Version 3.8 โดยมี Library ต่างๆ ได้แก่ TensorFlow, Scikit-learn รวมทั้ง Jupyter Notebook เช่นเดียวกันกับการติดตั้งบน Windows โดยมีขั้นตอน ดังนี้

3. ดาวน์โหลดและติดตั้ง **Miniconda** สำหรับ Python Version ล่าสุดบน macOS จากเว็บไซต์ <https://docs.conda.io/en/latest/miniconda.html>

The screenshot shows the Miniconda website with a sidebar on the left containing links like 'Conda', 'Conda-build', 'Miniconda', 'Windows installers', 'MacOSX installers', 'Linux installers', 'Installing', 'Other resources', 'Help and support', 'Contributing', and 'Conda license'. The main content area is titled 'See if Miniconda is right for you.' and features two sections: 'Windows installers' and 'MacOSX installers'. Each section contains a table with columns for 'Python version', 'Name', 'Size', and 'SHA256 hash'.

Windows			
Python version	Name	Size	SHA256 hash
Python 3.9	Miniconda3 Windows 64-bit	57.7 MB	c3a43d8b4c4f492454dbf636cc8b59a45d8754f682b31ae7109ebc37ec3b8
	Miniconda3 Windows 32-bit	54.9 MB	5845f09dc4485dbba21854262b7d184ba61a8739c1a56818cc6258f233a0d46
Python 3.8	Miniconda3 Windows 64-bit	57.0 MB	4fa220ba497ba0506680c8843545372a9915131c81208993a1083f627c81
	Miniconda3 Windows 32-bit	54.2 MB	9c2ef76ba97246c85c286733ca38f4116884163f99a2a033fca681c470c681
Python 2.7	Miniconda2 Windows 64-bit	54.1 MB	69738254848329446f84b10235a9c4584988e6d4787b053baa4f706a4b7326c
	Miniconda2 Windows 32-bit	47.7 MB	c884926f8b06954057bc4996c7201ffa1374a8218e64e6415843752617b

MacOSX			
Python version	Name	Size	SHA256 hash
Python 3.9	Miniconda3 MacOSX 64-bit bash	42.2 MB	1b3b777cbbf1e6235cc6d58346a2a84228f8ecdb614678838c39baac5acbed1
	Miniconda3 MacOSX 64-bit pkg	49.7 MB	298f1f088817921a98c2181608f93b44afce67aac8a9242890137413cffe
Python 3.8	Miniconda3 MacOSX 64-bit bash	54.5 MB	a8ca8a7ba3935087288132364950649eac8ffacce2ea898f04c35796139f6478
	Miniconda3 MacOSX 64-bit pkg	62.0 MB	1a651803c1fa0b53095c9c368a8a5b7583bc784764558744927f5a05e317fa
Python 2.7	Miniconda2 MacOSX 64-bit bash	40.3 MB	8a7961628a2239c148766456380ba6a638f8c86982826b1878c30840988a43
	Miniconda2 MacOSX 64-bit pkg	48.4 MB	9ca4313e181024939c7a5a4f4805722594f8b9a9472883063ca3a7f66fa1da

4. เพื่อจะทำงานกับ **Zsh Shell** หรือ **Bash Shell** บน macOS ให้รันคำสั่งต่อไปนี้

```
source /opt/miniconda3/bin/activate
```

The terminal screenshot shows a macOS window with the title 'bin -- -zsh -- 80x5'. The prompt is 'nuttachot@Apples-MacBook-Pro bin %'. The user enters the command 'source /opt/miniconda3/bin/activate'. The prompt changes to '(base) nuttachot@Apples-MacBook-Pro bin %'.

5. แล้วรันคำสั่ง **conda init**

```
#สำหรับ Zsh Shell
conda init zsh
```

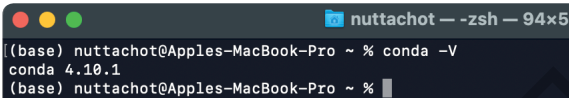
```
#สำหรับ Bash Shell
conda init
```

6. อัปเดต Conda เป็น Version ล่าสุด

```
conda update -n base -c defaults conda
```

7. เรียกดู Version ของ Conda

```
conda -V
```

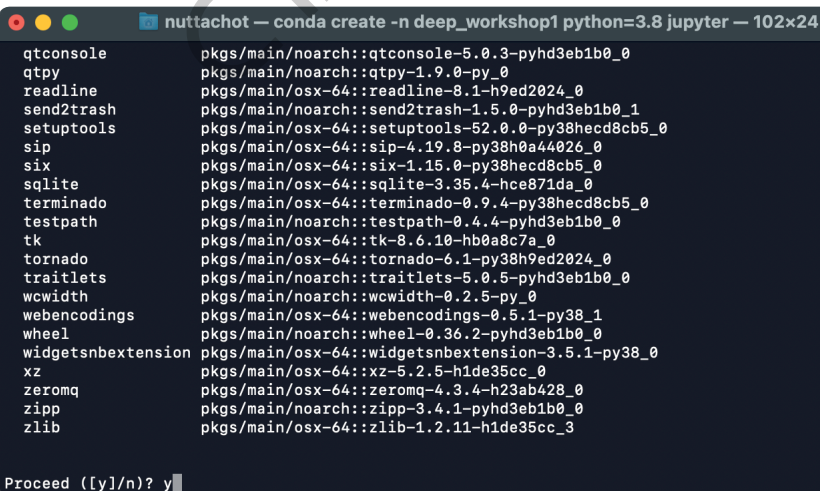


```
nuttachot — zsh — 94x5
(base) nuttachot@Apples-MacBook-Pro ~ % conda -V
conda 4.10.1
(base) nuttachot@Apples-MacBook-Pro ~ %
```

8. สร้าง Environment ใหม่ ตั้งชื่อเป็น deep_workshop1 สำหรับรัน Python Version 3.8 และติดตั้ง Jupyter Notebook ด้วยคำสั่ง conda create -n

```
conda create -n deep_workshop1 python=3.8 jupyter
```

9. พิมพ์ y แล้วกดปุ่ม Enter เพื่อสร้าง Environment ใหม่

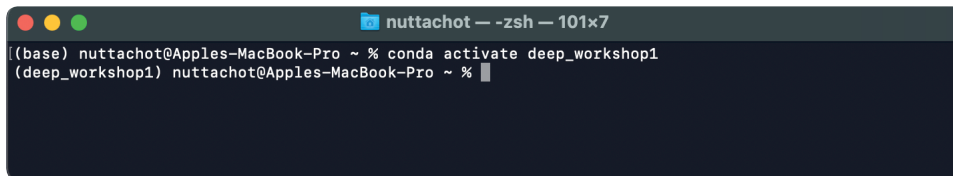


```
nuttachot — conda create -n deep_workshop1 python=3.8 jupyter — 102x24
qtconsole          pkgs/main/noarch::qtconsole-5.0.3-pyhd3eb1b0_0
qtpy                pkgs/main/noarch::qtpy-1.9.0-py_0
readline            pkgs/main/osx-64::readline-8.1-h9ed2024_0
send2trash          pkgs/main/noarch::send2trash-1.5.0-pyhd3eb1b0_1
setuptools          pkgs/main/osx-64::setuptools-52.0.0-py38h8c8cb5_0
sip                 pkgs/main/osx-64::sip-4.19.8-py38h0a44026_0
six                 pkgs/main/osx-64::six-1.15.0-py38h8c8cb5_0
sqlite              pkgs/main/osx-64::sqlite-3.35.4-hce871da_0
terminado           pkgs/main/osx-64::terminado-0.9.4-py38h8c8cb5_0
testpath            pkgs/main/noarch::testpath-0.4.4-pyhd3eb1b0_0
tk                  pkgs/main/osx-64::tk-8.6.10-hb0a8c7a_0
tornado             pkgs/main/osx-64::tornado-6.1-py38h9ed2024_0
traitlets           pkgs/main/noarch::traitlets-5.0.5-pyhd3eb1b0_0
wcwidth             pkgs/main/noarch::wcwidth-0.2.5-py_0
webencodings        pkgs/main/osx-64::webencodings-0.5.1-py38_1
wheel               pkgs/main/noarch::wheel-0.36.2-pyhd3eb1b0_0
widgetsnbextension pkgs/main/osx-64::widgetsnbextension-3.5.1-py38_0
xz                  pkgs/main/osx-64::xz-5.2.5-h1de35cc_0
zeromq              pkgs/main/osx-64::zeromq-4.3.4-h23ab428_0
zipp                pkgs/main/noarch::zipp-3.4.1-pyhd3eb1b0_0
zlib                pkgs/main/osx-64::zlib-1.2.11-h1de35cc_3

Proceed ([y]/n)? y
```

10. เข้าใช้ **Environment** ใหม่ด้วยคำสั่ง **conda activate** ตามด้วยชื่อ Environment

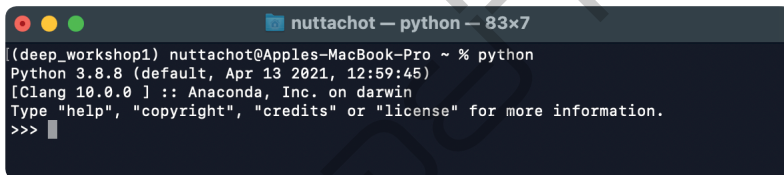
```
conda activate deep_workshop1
```



```
nuttachot — zsh — 101x7
(base) nuttachot@Apples-MacBook-Pro ~ % conda activate deep_workshop1
(deep_workshop1) nuttachot@Apples-MacBook-Pro ~ %
```

11. ตรวจสอบ **Python Version** ด้วยคำสั่ง **python** แล้วกดปุ่ม **Enter**

```
python
```



```
nuttachot — python — 83x7
(deep_workshop1) nuttachot@Apples-MacBook-Pro ~ % python
Python 3.8.8 (default, Apr 13 2021, 12:59:45)
[Clang 10.0.0 ] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

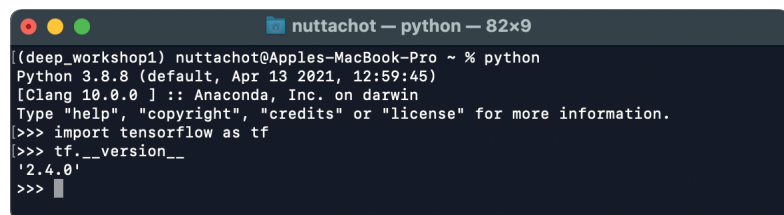
12. กลับมายัง **Command Line** อีกครั้งโดยกดคีย์ลัด **Ctrl** + **D**

13. ติดตั้ง **TensorFlow 2.4** ด้วยคำสั่ง **pip install**

```
pip install tensorflow==2.4
```

14. เรียกดู **TensorFlow Version** โดยพิมพ์ **python** แล้วกดปุ่ม **Enter** แล้วพิมพ์คำสั่งต่อไปนี้

```
import tensorflow as tf
tf.__version__
```



```
nuttachot — python — 82x9
(deep_workshop1) nuttachot@Apples-MacBook-Pro ~ % python
Python 3.8.8 (default, Apr 13 2021, 12:59:45)
[Clang 10.0.0 ] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> import tensorflow as tf
>>> tf.__version__
'2.4.0'
>>>
```

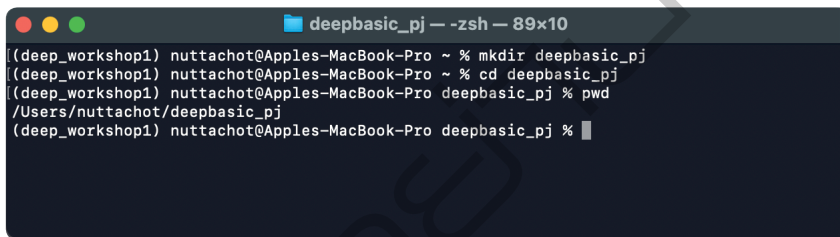
15. กดคีย์ลัด `Ctrl` + `D` เพื่อออกมายัง **Command Line**

16. ติดตั้ง **Scikit-learn** ด้วยคำสั่ง **conda install** ตามด้วยพารามิเตอร์ ดังนี้

```
conda install scikit-learn
```

17. สร้าง **Project** ชื่อ **deepbasic_pj** ด้วยคำสั่ง **mkdir deepbasic_pj** แล้วไปยังโฟลเดอร์ดังกล่าว

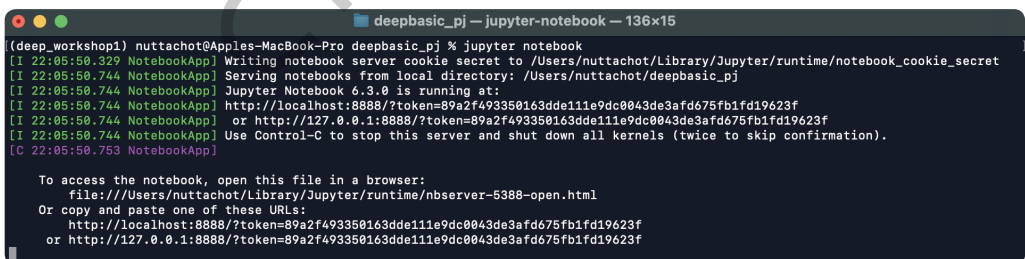
```
mkdir deepbasic_pj
cd deepbasic_pj
```



```
deepbasic_pj — zsh — 89x10
(deep_workshop1) nuttachot@Apples-MacBook-Pro ~ % mkdir deepbasic_pj
(deep_workshop1) nuttachot@Apples-MacBook-Pro ~ % cd deepbasic_pj
(deep_workshop1) nuttachot@Apples-MacBook-Pro deepbasic_pj % pwd
/Users/nuttachot/deepbasic_pj
(deep_workshop1) nuttachot@Apples-MacBook-Pro deepbasic_pj %
```

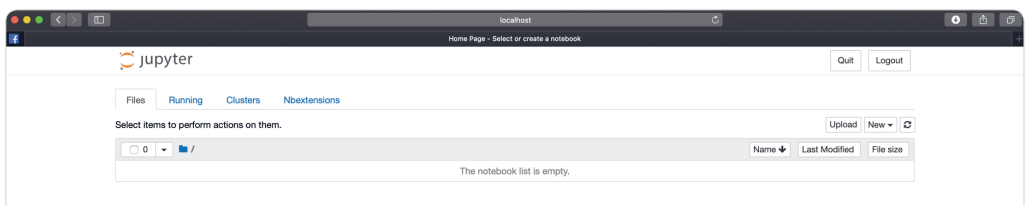
18. เปิด **Jupyter Notebook**

```
jupyter notebook
```

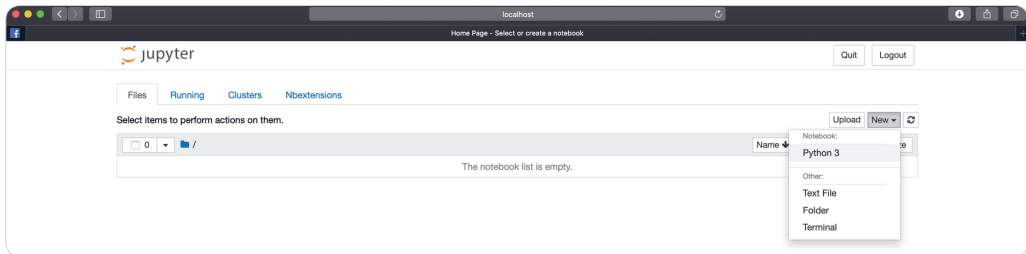


```
deepbasic_pj — jupyter-notebook — 136x15
(deep_workshop1) nuttachot@Apples-MacBook-Pro deepbasic_pj % jupyter notebook
[I 22:05:50.329 NotebookApp] Writing notebook server cookie secret to /Users/nuttachot/Library/Jupyter/runtime/notebook_cookie_secret
[I 22:05:50.744 NotebookApp] Serving notebooks from local directory: /Users/nuttachot/deepbasic_pj
[I 22:05:50.744 NotebookApp] Jupyter Notebook 6.3.0 is running at:
[I 22:05:50.744 NotebookApp] http://localhost:8888/?token=89a2f493350163dde11e9dc0043de3afd675fb1fd19623f
[I 22:05:50.744 NotebookApp] or http://127.0.0.1:8888/?token=89a2f493350163dde11e9dc0043de3afd675fb1fd19623f
[I 22:05:50.744 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 22:05:50.753 NotebookApp]

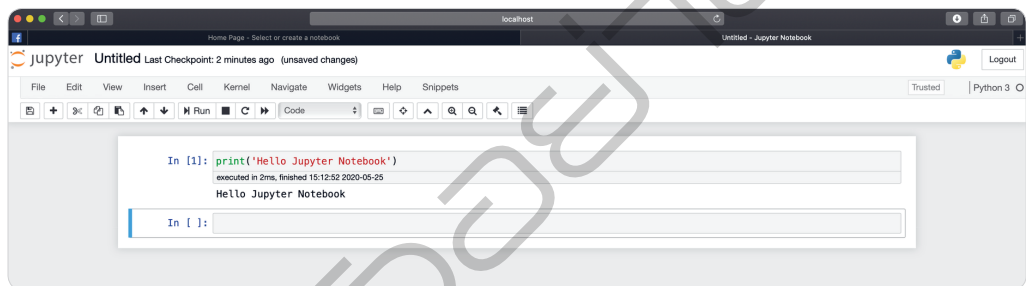
To access the notebook, open this file in a browser:
file:///Users/nuttachot/Library/Jupyter/runtime/nbserver-5388-open.html
Or copy and paste one of these URLs:
http://localhost:8888/?token=89a2f493350163dde11e9dc0043de3afd675fb1fd19623f
or http://127.0.0.1:8888/?token=89a2f493350163dde11e9dc0043de3afd675fb1fd19623f
```



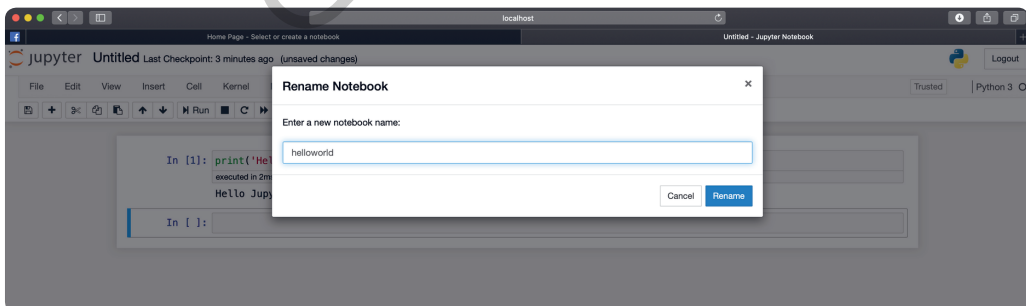
19. สร้าง Notebook โดยคลิกที่ New > Python 3



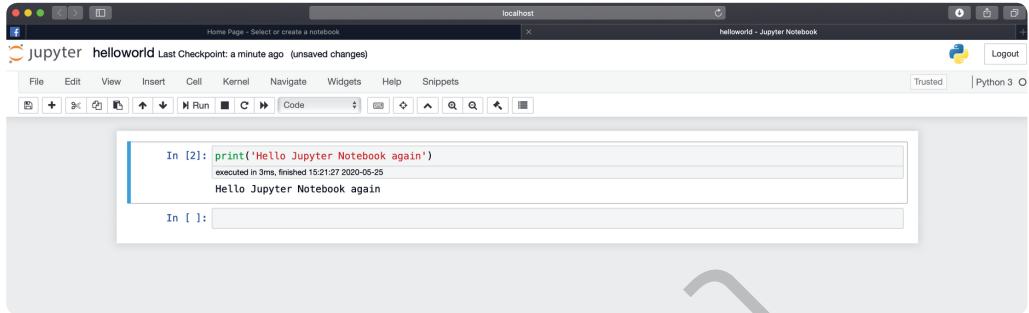
20. พิมพ์ `print('Hello Jupyter Notebook')` ใน Cell แรก แล้วกดคีย์ลัด **Shift** + **Enter** เพื่อรันโปรแกรมและสร้าง Cell ใหม่ไปพร้อมกัน



21. คลิกที่ **Untitled** พิมพ์ชื่อไฟล์ที่ต้องการ แล้วคลิกปุ่ม **Rename**



22. กลับไปแก้ไข Code ใน Cell แรกเป็น `print('Hello Jupyter Notebook again')` แล้วกดคีย์ลัด `Ctrl` + `Enter` เพื่อรันโปรแกรมโดยไม่มีการสร้าง Cell ใหม่



สรุปท้ายบท

ในบทนี้ ผู้อ่านได้เข้าใจเนื้อหา Introduction to AI ดังนี้

- AI และการนำวิธีการเรียนรู้ทาง Machine Learning มาแก้ปัญหา
- ความแตกต่างระหว่าง Machine Learning กับ Traditional Programming
- Machine Learning ประเภทต่างๆ

รวมทั้งได้ติดตั้งเครื่องมือที่จำเป็นสำหรับการทำ Workshop ในงานทางด้าน AI โดยใช้ Python และ Jupyter Notebook ในบทต่อไป เราจะทดลองสร้าง Model แบบ Deep Learning กันครับ



ปัญญาประดิษฐ์ หรือ Artificial Intelligence (AI) เป็น Keyword ที่นับวันยิ่งถูกนำมาพูดถึงมากขึ้นในการพัฒนา ระบบงาน และการพัฒนาผลิตภัณฑ์ต่างๆ จนเสมือนว่า งานใดก็ตามที่ไม่มี AI เป็นส่วนหนึ่งในนั้น มันจะกลายเป็น งานที่ไม่สามารถทัดเทียมคู่แข่งได้ นั่นทำให้ AI กลายเป็น ศาสตร์ที่มีบทบาทสำคัญในการวิวัฒนาการของดิจิทัลเทคโนโลยี

หนังสือเล่มนี้จะปูพื้นฐานที่จำเป็นสำหรับผู้เริ่มต้นศึกษาด้าน AI และ Deep Learning โดยมุ่งให้ผู้อ่านสามารถทำความเข้าใจแนวคิด ที่เป็นรากฐานไปจนถึงขั้นการฝึกสอน Model เพื่อนำไปใช้งานทางด้านภาพ การเล่นเกม และระบบการแนะนำ ด้วยการลงมือเขียน Code ภาษา Python บน Jupyter Notebook โดยใช้ Library เช่น TensorFlow, Scikit-learn และ NumPy แทนที่จะเน้นการอธิบาย ด้วยทฤษฎีหนักๆ โดยหนังสือเล่มนี้จะครอบคลุม AI Technologies และ Business Applications ใน 3 ส่วน ดังนี้



1 Machine Learning

เข้าใจแนวคิดและกระบวนการพัฒนา Model แบบ Deep Learning



2 AI Capabilities

ฝึกพัฒนา Model ให้มีความสามารถ ทางด้านภาพ การเล่นเกม และระบบการแนะนำ



3 AI Applications

เวิร์คช็อปการแนะนำหนังสือ จากการทำนายด้วย Model

Best Practices for Reskill and Upskilling

Part 1 : Chapter 1-12

1. แนวคิดพื้นฐานของ AI และ Machine Learning
2. กระบวนการพัฒนา Model แบบ Deep Learning
3. การทำ Feature Engineering
4. พื้นฐาน Neural Network
5. การใช้ Gradient แบ่งข้อมูลฝึก เพื่อลดค่า Loss
6. การปรับจูน Hyper Parameter ของ Model
7. การวิเคราะห์ประสิทธิภาพการเรียนรู้ของ Model ด้วย Learning Curve
8. การทำ Regularization
9. โครงสร้างแบบ Convolutional Neural Network ใน Deep Learning Model
10. การเลือกใช้ Loss Function ตอน 1
11. การเลือกใช้ Loss Function ตอน 2
12. การประเมิน Classification Model ด้วย Metrics สำคัญๆ

Part 2 : Chapter 13-16

13. การนำ Model ไป Deploy บน Production Environment
14. การฝึก AI Agent ให้เรียนรู้การ เล่นเกมด้วย Deep Q-Network
15. การนำ Weight ของ Pre-trained CNN Models ไปประยุกต์ใช้
16. การแนะนำหนังสือที่ทำนายด้วย Autoencoder Model

Programs and Fields of Study

✓ AI/Data Science ✓ Software Engineer ✓ Computer Science ✓ IT/ICT ✓ AI/ML Skills สำหรับคนที่สนใจข้ามสายงาน



อ.ดร.นิรุจน์ ปรกนกฤทธิ์
อ.ดร.สังจากรณ์ วิจารณ์
Instructor at Silpakorn University
บรรณาธิการ กิรพล ชาญเจริญ

PERAZU
สำนักส่งเสริม
สโลว์สโตร์
www.serazu.com



จัดจำหน่ายโดย **IDC**
ISBN 885-916-100-927-6
8 859161 009276
สำนัก 545 บาท