INFOPRESS

# 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.ณัฐโชติ พรหมฤทธิ์
อ.ดร.สัจจาภรณ์ ไวจรรยา

□□□□ □□□□□□.indd 1 8/12/2564 BE 13:43

# Fundamental of DEEP LEARNING in Practice


| | |
|---|---|
| Writers | ดร.ณัฐโชติ พรหมฤทธิ์, ดร.สัจจาภรณ์ ไวจรรยา |
| Editor | ภีรพล คชาเจริญ |
| Graphic Designers | ชวนันท์ รัตนะ, สิริลักษณ์ วาระเลิศ |
| Page Layout | วรวิทย์ วรจินต์ |
| Proofreader | เกษรา พรวัฒนมงคล |
| Publishing Coordinators | วรพล ณธิกุล, 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 เรายินดีรับงานเขียนของนักวิชาการและนักเขียนทุกท่าน ท่านผู้สนใจกรุณาติดต่อผ่านทางอีเมลที่ infopress@idcpremier.com หรือทางโทรศัพท์หมายเลข 0-2962-1081 (อัตโนมัติ 10 คู่สาย) โทรสาร 0-2962-1084


สร้างสรรค์โดย

INFOPRESS

พิมพ์ครั้งที่ 1 ธันวาคม 2564

ข้อมูลทางบรรณานุกรม
ดร.ณัฐโชติ พรหมฤทธิ์
Fundamental of DEEP LEARNING in Practice
นนทบุรี : ไอดีซีฯ, 2564
504 หน้า

1. การเขียนโปรแกรมสำหรับระบบฐานความรู้
I ดร.สัจจาภรณ์ ไวจรรยา (ผู้แต่งร่วม)
II ชื่อเรื่อง
006.336
ISBN 978-616-487-274-5
ราคา 545 บาท

พิมพ์และจัดจำหน่ายโดย

IDC PREMIER

บริษัท ไอดีซี พรีเมียร์ จำกัด
200 หมู่ 4 ชั้น 19 ห้อง 1901
จัสมินอินเตอร์เนชั่นแนลทาวเวอร์
ถ.แจ้งวัฒนะ อ.ปากเกร็ด จ.นนทบุรี 11120
โทรศัพท์ 0-2962-1081 (อัตโนมัติ 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 มันบน 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
รองผู้อำนวยการฝ่ายพัฒนาองค์กร สำนักดิจิทัลเทคโนโลยี
มหาวิทยาลัยศิลปากร

อ. ดร.สัจจาภรณ์ ไวจรรยา
อาจารย์ประจำภาควิชาคอมพิวเตอร์
มหาวิทยาลัยศิลปากร



□□□□ □□□□□□.indd 4 8/12/2564 BE 13:43

# 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 และพยากรณ์ได้เองโดยไม่จำเป็นต้องเขียนโปรแกรมที่มีกฎหรือเงื่อนไขที่ซับซ้อนจำนวนมาก แต่ต้องอาศัยข้อมูลจำนวนมากพอในการสอนโมเดลแบบ Deep Learning จนเข้าใจคุณลักษณะ ส่วน AI Capabilities เป็นความสามารถของ AI ในด้านต่างๆ เช่น ระบบทำนาย ระบบผู้เชี่ยวชาญ ระบบคอมพิวเตอร์วิชัน ระบบประมวลผลเสียงหรือภาษา ระบบแปลงเสียงเป็นข้อความ เป็นต้น และสุดท้าย AI Applications ซึ่งเป็นการประยุกต์ใช้ความสามารถของ AI เพื่อใช้ประโยชน์จริง เช่น Book Recommendation ของ Amazon, Movie Recommendation ของ Netflix, Rekognition ระบบตรวจจับใบหน้าและเนื้อหาที่ไม่เหมาะสมของ Amazon, AIPRO ระบบเทรดหุ้นอัตโนมัติ, 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 เหมาะจะใช้เพื่อศึกษากระบวนการพัฒนา Model แบบ Deep Learning ผู้อ่านจะได้ทดลองสร้างและเทรน Model ได้ลองนำ Model ไปใช้งานด้วยการ Deploy บน Production Environment และได้ลองสร้างระบบแนะนำหนังสือแบบง่ายๆ ในบทสุดท้าย จึงเป็นหนังสือที่ผู้อ่านจะได้เรียนรู้การพัฒนาครบทั้ง 3 องค์ประกอบ ได้แก่ Machine Learning, AI Capabilities และ AI Applications ซึ่งจะเป็นพื้นฐานสำหรับศึกษาต่อในระดับ Advance ต่อไป

ภีรพล คชาเจริญ
บรรณาธิการ

□□□□ □□□□□□.indd 5 8/12/2564 BE 13:43

# CONTENTS


CHAPTER

## 01

### ทำความรู้จักกับ 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 (กรณีมี GPU)..... 9

ขั้นตอนการติดตั้ง cuDNN (กรณีมี 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

การจำแนกภาพลายมือจาก MNIST ด้วย CNN MODEL ที่พัฒนาจาก KERAS TENSORFLOW .... 46

เตรียมความพร้อมก่อนทำ WORKSHOP..... 47

ขั้นตอนการสร้าง NOTEBOOK และ IMPORT PACKAGE..... 47

ขั้นตอนการเพิ่มโค้ดเพื่อทดสอบการใช้งาน 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


□□□□ □□□□□□.indd 1 8/12/2564 BE 13:43

STEP 5 SAVE MODEL....58
STEP 6 LOAD MODEL....59
STEP 7 PREDICTION....62
STEP 8 EVALUATION....63
สรุปท้ายบท....66

CHAPTER 03

การทำ 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

การอิมพลิเมนต์ 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
ทดลองอิมพลิเมนต์ NEURAL NETWORK ด้วย NUMPY....96
สรุปท้ายบท....102

CHAPTER 05

การใช้ความชันจากการสุ่มแบ่งข้อมูลฝึกเพื่อลดค่า 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 แบบพื้นฐานด้วยตนเอง ..... 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


□□□□ □□□□□□.indd 3 8/12/2564 BE 13:43

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 NORMALIZATION ..... 200

ทดลองแก้ปัญหา OVERFITTING ของ NEURAL NETWORK ด้วยเทคนิค DROPOUT ..... 205
- ทดลองทำ DROPOUT ..... 206

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 ด้วยการทำ 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 ที่เหมาะสมสำหรับการ 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
เพื่อจัดเก็บไฟล์ที่จะต้อง 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 อย่างไรจนเล่นเกมชนะมนุษย์ .....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 เพื่อเก็บทุกเหตุการณ์ในการแข่งขัน .....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 ที่นับวันยิ่งถูกนำมาพูดถึงมากขึ้น ในการพัฒนาระบบงาน และการพัฒนาผลิตภัณฑ์ต่างๆ จนเสมือนว่างานใดก็ตามที่ไม่มี AI เป็นส่วนหนึ่งในนั้น มันจะกลายเป็นงานที่ไม่สามารถทัดเทียมคู่แข่งได้ นั่นทำให้ AI กลายเป็นศาสตร์ที่มีบทบาทสำคัญในการวิวัฒน์ของดิจิทัลเทคโนโลยี

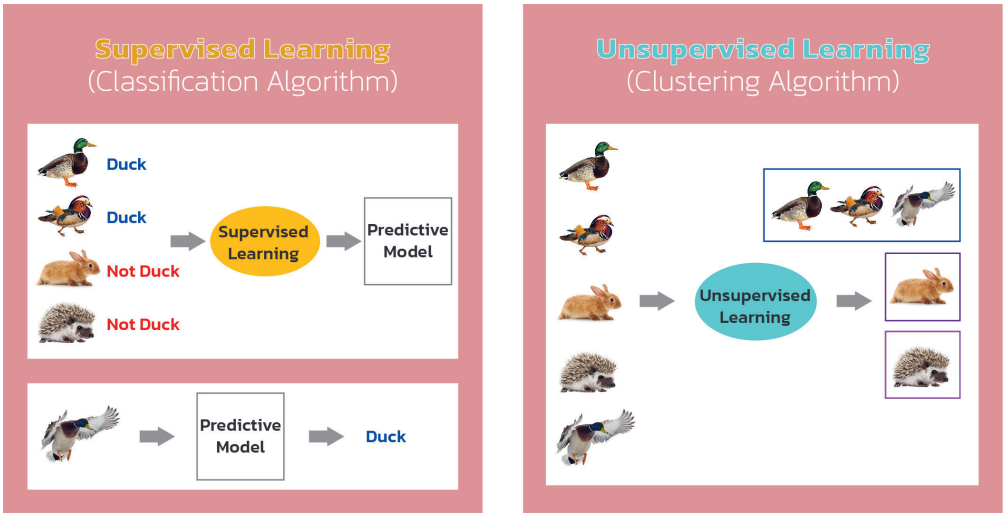
ที่รับเข้ามาเป็น **Duck** หรือ **Not Duck** ซึ่งเป็นงานที่ค่อนข้างยากเมื่อต้องเขียน Program แบบ Traditional Programming

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

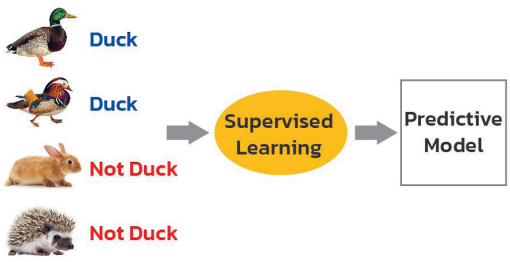
1. Supervised Learning
2. Unsupervised Learning
3. Reinforcement Learning




รูปที่ 1.3 Supervised Learning และ Unsupervised Learning
ภาพดัดแปลงจาก : https://blog.westerndigital.com/machine-learning-pipeline-object-storage/

## 1. SUPERVISED LEARNING

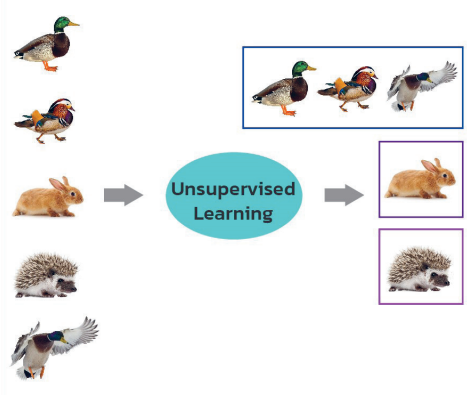
ในการใช้เทคนิคทาง Machine Learning เพื่อแยกประเภทภาพเป็ด เราเพียงนำภาพเป็ดและภาพสัตว์อื่นๆ พร้อมผลเฉลย (Label Data) มาฝึกสอนเพื่อ Predictive Model ซึ่งเราเรียกวิธีการเรียนรู้แบบนี้ว่า Supervised Learning




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 ภาพเป็ด ภาพกระต่าย และภาพตัวเม่น เป็นต้น

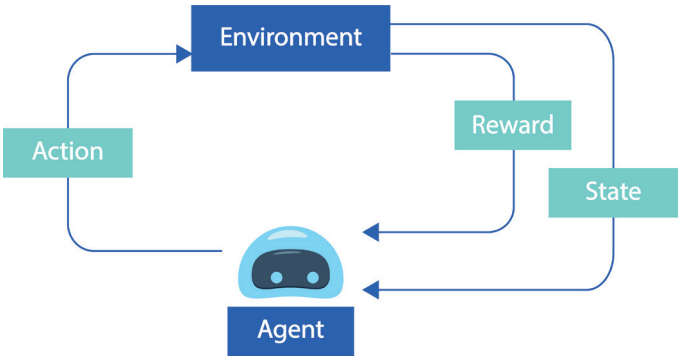



รูปที่ 1.5 Unsupervised Learning
ภาพดัดแปลงจาก : https://blog.westerndigital.com/machine-learning-pipeline-object-storage/

## 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) เพื่อเก็บข้อมูลเป็นจำนวนที่มากพอ 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)

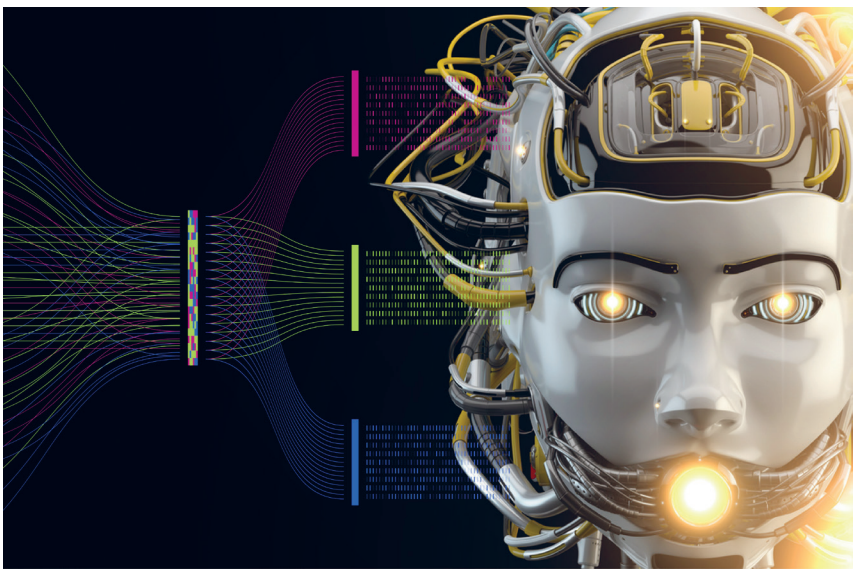
การลง Library ใหม่ เราสามารถใช้ Conda ซึ่งเป็นเครื่องมือหลักที่มาพร้อมกับ Anaconda สำหรับการติดตั้ง Library และสร้าง Environment เฉพาะในการรันโปรแกรมโดยไม่ให้ Library ใหม่ๆ ไปปะปนกับส่วนอื่นภายในเครื่องคอมพิวเตอร์ แต่ถ้าใครยังไม่ได้ติดตั้ง Anaconda ผู้เขียนแนะนำให้ติดตั้ง Miniconda แทน เนื่องจากเราต้องการ 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

CHAPTER

# 03 การทำ FEATURE ENGINEERING ด้วย PANDAS

"Data is the new oil!" หากเป็นเรื่องเกี่ยวกับ AI และ Machine Learning แล้ว Data น่าจะเหมือนกับน้ำมันดิบ ซึ่งก่อนที่จะนำมาใช้ในการวิเคราะห์ หรือใช้ในการ Train Model ก็ต้องมีการสกัดเอา Features ที่เกี่ยวข้องออกมาเสียก่อน เรียกว่า การทำ Feature Engineering และก่อนจะสกัด Features ของข้อมูล ก็ต้องมีการสำรวจข้อมูลเพื่อพิจารณาคุณภาพเสียก่อน แล้วจึงเลือกเทคนิคในการทำ Feature Engineering ที่เหมาะสม ในบทนี้ ผู้อ่านจะได้ฝึกปฏิบัติการทำ Feature Engineering กับข้อมูล Wine Magazine Dataset ด้วยเทคนิคต่างๆ



# เทคนิคการทำ FEATURE ENGINEERING

**Data is the new oil!** หลายคนคงเคยได้ยินประโยคนี้ใช่ไหมครับ แต่อันที่จริงหากเป็นเรื่องเกี่ยวกับ AI และ Machine Learning แล้ว Data น่าจะเป็นเหมือนกับน้ำมันดิบ (Petroleum) มากกว่า เพราะโดยมากแล้วก่อนที่จะนำ Data มาใช้ในการวิเคราะห์ รวมทั้งใช้ในการ Train Model ให้มี Accuracy ที่สูงได้ เราจะต้องมีการกลั่น หรือสกัดเอา Features ที่เกี่ยวข้องออกมาเสียก่อน โดยเราเรียกกระบวนการในการสกัด Features จากข้อมูลดิบ (Raw Data) รวมทั้งการทำความสะอาดข้อมูล (Data Cleaning) ว่า การทำ **Feature Engineering**

ประโยชน์หลักๆ ของการทำ Feature Engineering คือ

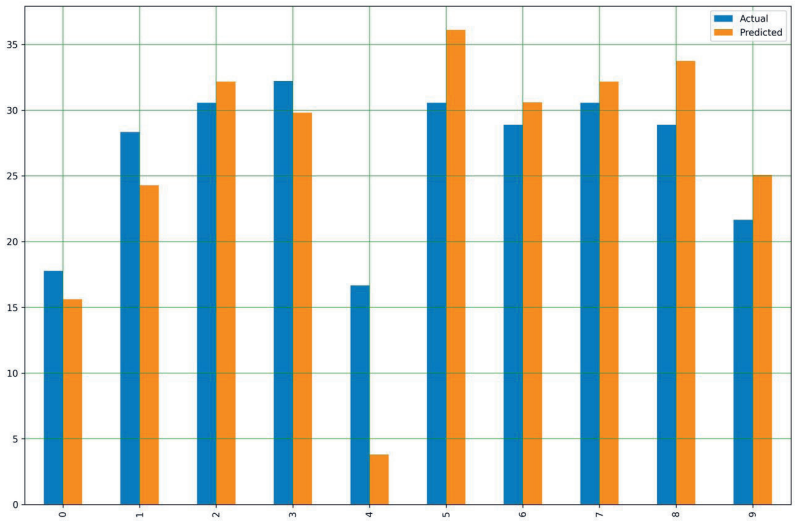
- เพื่อเตรียม Dataset ให้พร้อมสำหรับการทำ Data Analytics หรือเป็น Input ของ Machine Learning Algorithm
- เพื่อเพิ่มประสิทธิภาพให้ Machine Learning Model

ในบทนี้ ผู้อ่านจะได้ฝึกปฏิบัติการทำ Feature Engineering กับข้อมูล Wine Magazine Dataset โดยใช้เทคนิคต่างๆ 5 เทคนิค ได้แก่

1. Imputation
2. Handling Outlier
3. Binning
4. Log Transform
5. One-Hot Encoding

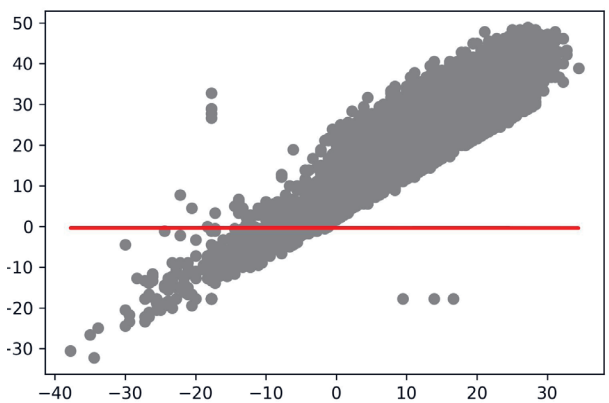
## วิธีทำความเข้าใจคุณภาพของข้อมูลด้วย PANDAS

ก่อนจะสกัด Features ของข้อมูล เราจะต้องมีการสำรวจข้อมูลเพื่อพิจารณาคุณภาพของมันในแง่ต่างๆ แล้วจึงเลือกเทคนิคในการทำ Feature Engineering ที่เหมาะสม



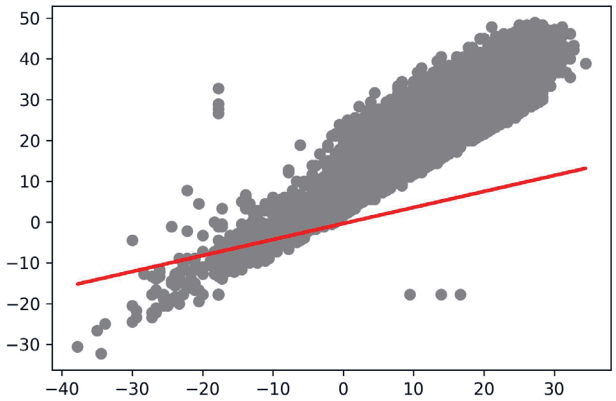

**23.** แสดง **Model** ที่ถูกสร้างขึ้นเมื่อมีการ Train ใน Epoch ที่ 1

```
M = [i[0] for i in wb]
L = [i[2] for i in wb]
C = [i[1] for i in wb]
y_pred = [predict(i, M[0], C[0]) for i in x_test]

plt.scatter(x_test, y_test,  color='gray')
plt.plot(x_test, y_pred, color='red', linewidth=2)
plt.savefig('min_max_temp1.jpeg', dpi=300)
```



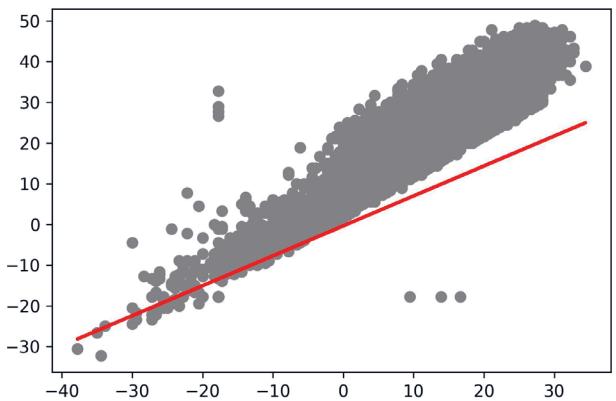
**24.** แสดง **Model** ที่ถูกสร้างขึ้นเมื่อมีการ Train ใน Epoch ที่ 5

```
y_pred = [predict(i, M[4], C[4]) for i in x_test]

plt.scatter(x_test, y_test,  color='gray')
plt.plot(x_test, y_pred, color='red', linewidth=2)
plt.savefig('min_max_temp5.jpeg', dpi=300)
```




**25.** แสดง **Model** ที่ถูกสร้างขึ้นเมื่อมีการ Train ใน Epoch ที่ 10

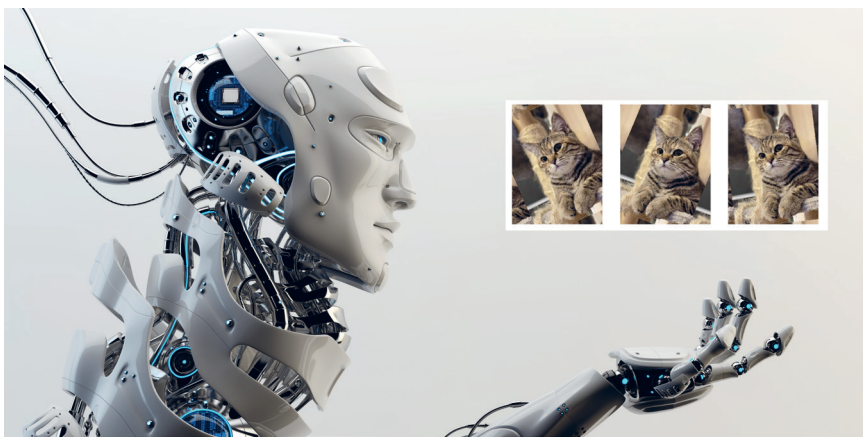
```
y_pred = [predict(i, M[9], C[9]) for i in x_test]

plt.scatter(x_test, y_test,  color='gray')
plt.plot(x_test, y_pred, color='red', linewidth=2)
plt.savefig('min_max_temp10.jpeg', dpi=300)
```



CHAPTER

# 08 การทำ REGULARIZATION แบบสมัยใหม่ด้วย AUGMENTATION, BATCH NORMALIZATION และ DROPOUT

บทนี้เราจะกล่าวถึงการทำ Regularization ซึ่งหน้าที่ของ Regularization คือ การปรับแต่งให้ Model มีประสิทธิภาพในการทำนายที่ดี สามารถลดการเกิด Error จากข้อมูลที่มันไม่เคยเห็นมาก่อนด้วยการเรียนรู้จาก Training Dataset และเราจะทดลองแก้ปัญหา Overfitting ของ Neural Network แบบ Classification Model ที่มีการ Train ด้วย Fashion-MNIST Dataset โดยยกตัวอย่างการทำ Regularization แบบสมัยใหม่ ด้วยเทคนิคที่เป็นที่นิยมในปัจจุบัน 3 เทคนิคคือ Augmentation, Batch Normalization และ Dropout



# การเตรียมและสอน MODEL ด้วย FASHION-MNIST DATASET

รูปที่ 8.1 แสดง Fashion-MNIST Dataset ของภาพเสื้อผ้า กระเป๋า และรองเท้า

Fashion-MNIST เป็น Dataset ของภาพเสื้อผ้า กระเป๋า และรองเท้า ขนาด 28 x 28 Pixels แบบ Grayscale แบ่งเป็นข้อมูล Train 60,000 ภาพ และข้อมูล Test อีก 10,000 ภาพ รวมทั้งหมด 10 ประเภท โดยมีการกำหนด Label ตั้งแต่ 0-9 ดังนี้

| | |
|---|---|
| 0 : T-shirt/top | 1 : Trouser |
| 2 : Pullover | 3 : Dress |
| 4 : Coat | 5 : Sandal |
| 6 : Shirt | 7 : Sneaker |
| 8 : Bag | 9 : Ankle boot |

# การเตรียม DATASET สำหรับ TRAIN และ VALIDATION

สิ่งที่ต้องทำก็คือ เราจะต้องมีการ Load Dataset แล้วขยายมิติของ Dataset ทำ Scaling ข้อมูลระหว่าง 0-1 เข้ารหัสผลเฉลยแบบ One-Hot Encoding และ Split Dataset สำหรับการ Train และ Validation ดังต่อไปนี้

1. **Import Library** และกำหนดค่า Parameter ที่จำเป็น

```
import tensorflow as tf
Adam = tf.keras.optimizers.Adam
to_categorical = tf.keras.utils.to_categorical
ImageDataGenerator = tf.keras.preprocessing.image.ImageDataGenerator
fashion_mnist = tf.keras.datasets.fashion_mnist
load_img = tf.keras.preprocessing.image.load_img
img_to_array = tf.keras.preprocessing.image.img_to_array

from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt

import plotly.graph_objs as go
from plotly import subplots
import plotly

import warnings
warnings.filterwarnings('ignore')
```

2. กำหนดขนาดภาพ จำนวน Class สัดส่วนข้อมูลทดสอบและขนาด Batch Size

```
IMG_ROWS = 28
IMG_COLS = 28
NUM_CLASSES = 10
VAL_SIZE = 0.2
RANDOM_STATE = 99
BATCH_SIZE = 256
```

# การเตรียมภาพสำหรับทำ IMAGE AUGMENTATION

แต่ก่อนอื่นเราจะอ่านไฟล์ภาพน้องเหมียวตังค์ฟูลแห่งเพจเจ้านายลายสลิด (ดาวน์โหลดได้ที่ serazu.com) มาทดลองทำ Image Augmentation ตามขั้นตอนดังนี้

1. อ่านไฟล์ภาพน้องตังค์ฟูล

```
cat = load_img('cat.jpg')
cat
```

2. ขยายมิติของภาพจาก 3 มิติเป็น 4 มิติ เพื่อเตรียมนำเข้า Function ทำ Image Augmentation

```
cat = img_to_array(cat)
print(cat.shape)
cat = cat.reshape(1,cat.shape[0],cat.shape[1],cat.shape[2])
print(cat.shape)
```

```
(1440, 1080, 3)
(1, 1440, 1080, 3)
```

# ทดลองทำ IMAGE AUGMENTATION ด้วยการ VERTICAL SHIFT

ทดลองทำ Vertical Shift ด้วยการเลื่อนภาพขึ้นลงแบบสุ่มไม่เกิน 20%

```
datagen = ImageDataGenerator(height_shift_range=0.2)
aug_iter = datagen.flow(cat, batch_size=1)
fig, ax = plt.subplots(nrows=1, ncols=3, figsize=(15,15))
for i in range(3):
    image = next(aug_iter)[0].astype('uint8')
    ax[i].imshow(image)
    ax[i].axis('off')

fig.savefig('cat1.jpeg', dpi=300)
```

## ทดลองทำ IMAGE AUGMENTATION ด้วยการ HORIZONTAL SHIFT

ทดลองทำ Horizontal Shift ด้วยการเลื่อนภาพซ้ายขวาแบบสุ่มไม่เกิน 20%

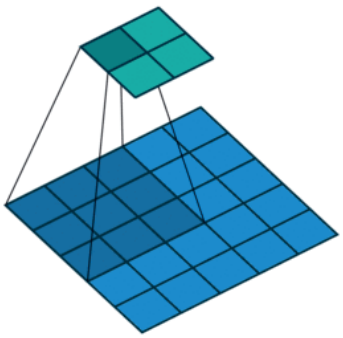
```
datagen = ImageDataGenerator(width_shift_range=0.2)
aug_iter = datagen.flow(cat, batch_size=1)
fig, ax = plt.subplots(nrows=1, ncols=3, figsize=(15,15))
for i in range(3):
    image = next(aug_iter)[0].astype('uint8')
    ax[i].imshow(image)
    ax[i].axis('off')
fig.savefig('cat2.jpeg', dpi=300)
```

# การจัดการกับ INPUT IMAGE แบบ 1 CHANNEL ด้วยการทำ STRIDING

การ Striding เป็นกระบวนการในการทำ Convolution โดยการเลื่อนแผ่น Kernel ไปบน Input Image ซึ่งโดย Default ของ Convolution แล้ว Stride จะมีค่าเท่ากับ 1 คือ จะมีการเลื่อน Kernel ไปบน Input Image ครั้งละ 1 Pixel

เราสามารถลดขนาดของภาพที่แต่ละ Pixel มีความสัมพันธ์กันในเชิงพื้นที่ได้โดยการเพิ่มค่า Stride ซึ่งเมื่อมีการกำหนดค่า Stride มากขึ้น จะทำให้การเหลื่อมกันของ Kernel ตอนที่มีการทาบกับ Input Image และขนาดของ Feature Map ลดลง

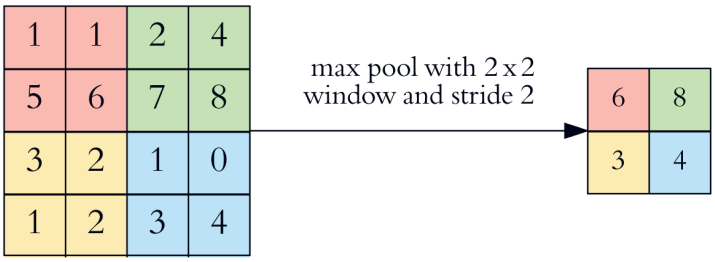
เมื่อกำหนด Stride เท่ากับ 2 แล้ว Kernel ขนาด 3 x 3 จะถูก Slide ข้าม Pixel ของ Input Image ขนาด 5 x 5 ทีละ 2 Pixels ทำให้ได้ Feature Map ขนาด 2 x 2 ดังรูปที่ 9.3



รูปที่ 9.3 การทำ Stride โดยการเลื่อนแผ่น Kernel ไปบน Input Image
เครดิตภาพ : https://towardsdatascience.com
โดยผู้อ่านสามารถเปิดชมภาพเคลื่อนไหวได้ที่ไฟล์ A stride 2 convolution.gif

# การจัดการกับ INPUT IMAGE แบบ 1 CHANNEL ด้วยการทำ POOLING

นอกจากการลดขนาดของภาพด้วยการเพิ่มค่า Stride โดยการ Slide Kernel ข้าม Pixel ของ Input Image ตามระยะทางที่กำหนดแล้ว ยังมีอีกวิธีหนึ่งในการลดขนาดของภาพ นั่นก็คือ การทำ Max Pooling หรือ Average Pooling โดย Pooling จะเป็นกระบวนการทำงานภายนอก CNN Layer

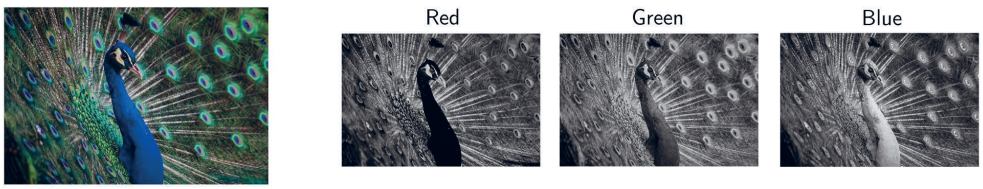



รูปที่ 9.4 Max Pooling เป็นการเลือกตัวแทนของภาพด้วยการหาค่ามากที่สุดหรือค่าเฉลี่ยจาก Pixel
เครดิตภาพ : https://towardsdatascience.com

Max Pooling หรือ Average Pooling จะเป็นการเลือกตัวแทนของภาพด้วยการหาค่ามากที่สุดหรือค่าเฉลี่ยจาก Pixel ใน Window ตามขนาดที่กำหนด เช่น ขนาด 2 x 2 ซึ่งจะทำให้มีการลดขนาดของภาพลงได้ครึ่งหนึ่ง ดังตัวอย่างด้านบน

## การจัดการกับ INPUT IMAGE แบบ MULTI-CHANNEL

อย่างไรก็ตาม ในตัวอย่างที่ผ่านมาจะเป็นการจัดการกับ Input Image แบบ 1 Channel เช่น ภาพแบบ Grayscale แต่บ่อยครั้งที่ Input Image ของเราจะเป็นภาพสีแบบ 3 Channels (โดยทั่วไปจำนวน Channel ของ Input Image จะเพิ่มขึ้นเมื่อมันถูกส่งเข้าสู่ชั้น CNN Layer ที่ลึกขึ้น)




รูปที่ 9.5 การจัดการกับภาพ RGB แบบ 3 Channels
เครดิตภาพ : https://unsplash.com/photos/_d3sppFprWI

เพื่อจะจัดการกับ Input Image แบบ 3 Channels อย่างเช่น ภาพสีในระบบ RGB เราจะต้องใช้ Kernel จำนวน 3 ตัวในการ Slide ไปบน Input Image แต่ละ Channel ซึ่งเราเรียก Kernel ทั้ง 3 ตัวว่า Filter (ในที่นี้ 1 Filter ประกอบด้วย Kernel 3 Kernels)

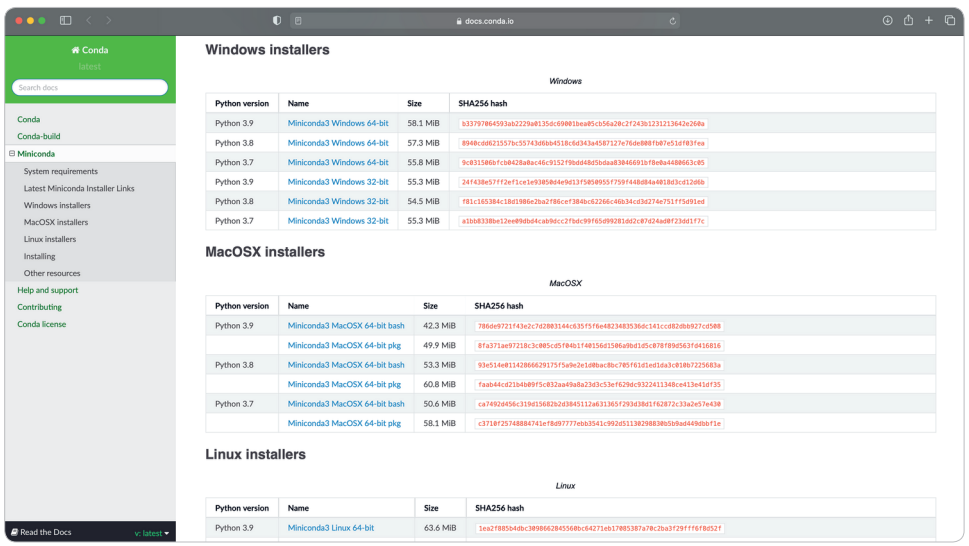
# WORKSHOP การสร้าง ENVIRONMENT สำหรับ TRAIN และ SAVE MODEL

## ทดลองสร้าง ENVIRONMENT ใหม่

เราจะสร้าง Environment ชื่อ basic_model สำหรับรัน Python 3.8 โดยที่มี Package/Library ต่างๆ ได้แก่ FastAPI, Uvicorn, Python-dotenv, Pydantic, TensorFlow, Locust, Plotly, Scikit-learn, Seaborn รวมทั้ง Jupyter Notebook ซึ่งมีขั้นตอน ดังต่อไปนี้

### ขั้นตอนการสร้าง CONDA ENVIRONMENT ใหม่

1. สำหรับผู้อ่านที่ยังไม่ได้ติดตั้ง **Miniconda** ท่านสามารถดาวน์โหลดและติดตั้ง Miniconda ได้จากเว็บไซต์ https://docs.conda.io/en/latest/miniconda.html




2. สร้าง Environment ใหม่ ตั้งชื่อเป็น **basic_model** สำหรับรัน Python 3.8 และติดตั้ง Library ที่จำเป็น รวมทั้ง Jupyter Notebook โดยใช้คำสั่ง **conda create -n**

```
conda create -n basic_model python=3.8 fastapi uvicorn python-dotenv pydantic
locust plotly scikit-learn seaborn jupyter -c conda-forge
```

CHAPTER

# 14

# การสอน AI เรียนรู้การเล่นเกมด้วย DEEP REINFORCEMENT LEARNING บน GOOGLE COLAB PRO

ในบทนี้ ผู้อ่านจะได้ฝึก AI Agent ให้เล่นเกม Pong หรือ Table Tennis ซึ่งเป็นหนึ่งใน Arcade Game ของ Gym Framework จาก OpenAI ที่มี Elon Musk เป็นหนึ่งในผู้ก่อตั้ง โดยเราจะให้ AI Agent เรียนรู้การเล่นเกมด้วย Deep Q-Network (DQN) ที่เป็น Deep Learning Model เช่นเดียวกับที่ DeepMind ได้ใช้ฝึกสอน AI ซึ่งก่อนฝึก AI Agent ผู้อ่านจะได้ทำความเข้าใจ Q-Table Learning ซึ่งเป็น Reinforcement Learning Algorithm สำหรับการประมาณค่า Action-values หรือ Q-values และการคำนวณค่า Q-values โดยใช้ DEEP Q-NETWORK

CHAPTER

# 16 การทำระบบแนะนำหนังสือ (BOOK RECOMMENDATION WORKSHOP)

ในบทนี้ ผู้เขียนจะยกตัวอย่างการสร้างระบบแนะนำหนังสือ (Book Recommendation) ด้วยเทคนิค Collaborative Filtering จากการทำนายด้วย Autoencoder Model โดยอาศัยข้อมูลการให้คะแนนหนังสือจาก Book-Crossing Dataset ซึ่งมีการแยกเก็บข้อมูล User, Book และ Book Ratings เป็นไฟล์ *.csv ทั้งหมด 3 ไฟล์ และใช้ Autoencoder Model เพื่อเรียนรู้ และทำนายคะแนนความชื่นชอบในหนังสือแต่ละเล่ม เพื่อจะแนะนำหนังสือที่มีคะแนนสูงสุด 10 อันดับ ให้แก่ User แต่ละคน

| UserID | ISBN | BookRating | BookTitle |
|---|---|---|---|
| 243 | 0060915544 | 10 | The Bean Trees |
| 243 | 0375400117 | 10 | Memoirs of a Geisha |
| 243 | 0316601950 | 9 | The Pilot's Wife : A Novel |
| 243 | 0316776963 | 9 | Me Talk Pretty One Day |
| 243 | 0425163407 | 9 | Unnatural Exposure |
| 243 | 0446364800 | 9 | The General's Daughter |
| 243 | 0553580388 | 8 | The Patient |
| 243 | 0060977493 | 7 | The God of Small Things |
| 243 | 0316899984 | 7 | River, Cross My Heart |
| 243 | 0385720106 | 7 | A Map of the World |

# สรุปท้ายบท

ในบทนี้ เราได้สร้างตัวอย่างการแนะนำหนังสือด้วยเทคนิค Collaborative Filtering จากการทำนายด้วย Autoencoder Model ซึ่ง Collaborative Filtering เป็นเทคนิคหนึ่งที่ใช้ในการทำ Recommendation โดยอาศัยข้อมูลความพึงพอใจของ User ที่มีต่อ Item ต่างๆ (หนังสือ ภาพยนตร์ หรือเพลง ฯลฯ)

ในการ Train Autoencoder Model เราได้นำข้อมูลการให้คะแนนหนังสือจาก Book-Crossing Dataset ซึ่งมีการแยกเก็บข้อมูล User, Book และ Book Ratings เป็น CSV ไฟล์มา Transform ไปเป็น Dataset หรือ Matrix ขนาด M x N โดย M คือ จำนวน User และ N คือ จำนวนหนังสือ ด้วย SQL Query Command และ Pandas

ซึ่งข้อมูลข้อมูลความพึงพอใจของ User ต่อหนังสือจะถูกนำมาเข้าเป็น Input ของ Autoencoder Model ดังนั้น ขนาด Input ของ Model (จำนวน Input Node) ก็คือจำนวนหนังสือ หรือจำนวน Column (N) ของ Matrix นั่นเอง

แนวคิดสำคัญของการแนะนำหนังสือด้วย Autoencoder Model คือ การนำ Dataset หรือ Matrix ที่ได้มานำเข้า Model เป็นทั้ง Input และผลเฉลยของ Model เพื่อฝึกสอน Model ให้สร้างผลการทำนายให้เหมือนกับ Input Data ซึ่งขณะที่ฝึกสอน Model จะรู้จัก Pattern ของความพึงพอใจของ User ทำให้มันสามารถทำนายคะแนนหนังสือของ User แม้ User จะไม่เคย Review มันมาก่อน