



การแยกแยะความเจ็บปวดของแมวจากภาพถ่ายด้วยการเรียนรู้เชิงลึก

FELINE FEELINGS UNLEASHED: HARNESSING DEEP LEARNING THROUGH PHOTOS

FOR CAT PAIN DETECTION



ชุตินันท์ นามบุญลือ

บัณฑิตวิทยาลัย มหาวิทยาลัยศรีนครินทรวิโรฒ

2566

การแยกแยะความเจ็บปวดของแมวจากภาพถ่ายด้วยการเรียนรู้เชิงลึก



สารนิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
วิทยาศาสตร์มหาบัณฑิต สาขาวิชาวิทยาการข้อมูล  
คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ  
ปีการศึกษา 2566

ลิขสิทธิ์ของมหาวิทยาลัยศรีนครินทรวิโรฒ

FELINE FEELINGS UNLEASHED: HARNESSING DEEP LEARNING THROUGH PHOTOS  
FOR CAT PAIN DETECTION



CHUTIMON NAMBOONLUE

A Master's Project Submitted in Partial Fulfillment of the Requirements  
for the Degree of MASTER OF SCIENCE  
(Data Science)

Faculty of Science, Srinakharinwirot University

2023

Copyright of Srinakharinwirot University

สารนิพนธ์

เรื่อง

การแยกแยะความเจ็บปวดของแมวจากภาพถ่ายด้วยการเรียนรู้เชิงลึก

ของ

ชุตินนท์ นามบุญลือ

ได้รับอนุมัติจากบัณฑิตวิทยาลัยให้นับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการข้อมูล

ของมหาวิทยาลัยศรีนครินทรวิโรฒ

(รองศาสตราจารย์ นายแพทย์ฉัตรชัย เอกปัญญาสกุล)

คณบดีบัณฑิตวิทยาลัย

คณะกรรมการสอบปากเปล่าสารนิพนธ์

ที่ปรึกษาหลัก

(อาจารย์ ดร.วีระ สอึ้ง)

ประธาน

(รองศาสตราจารย์ ดร.สุพัฒนา เขื่อนทิวเกียรติ)

กรรมการ

(อาจารย์ ดร.ศุภร คนธภักดิ์)

|                  |  |
|------------------|--|
| ชื่อเรื่อง       | การแยกแยะความเจ็บปวดของแมวจากภาพถ่ายด้วยการเรียนรู้เชิงลึก |
| ผู้วิจัย         | ชุตินนท์ นามบุญลือ   |
| ปริญญา           | วิทยาศาสตร์มหาบัณฑิต                                       |
| ปีการศึกษา       | 2566   |
| อาจารย์ที่ปรึกษา | อาจารย์ ดร. วีระ สอิ่ง                                     |

ในปัจจุบันการตรวจจับความเจ็บปวดของแมวนั้นยังต้องอาศัยความเชี่ยวชาญและประสบการณ์อยู่มาก งานวิจัยชิ้นนี้จึงทำการฝึกสอนแบบจำลองโครงข่ายประสาทเทียมแบบคอนโวลูชันเพื่อสร้างแบบจำลองในการแยกแยะความเจ็บปวดของแมวจากรูปถ่ายใบหน้าออกเป็นกลุ่มที่มีความเจ็บปวดกับกลุ่มที่ไม่มีความเจ็บปวด ชุดข้อมูลที่ใช้ประกอบด้วยภาพถ่ายใบหน้าแมวจากประเทศไทยกลุ่มละ 57 ตัวที่มีการจำแนกความเจ็บปวดโดยสัตวแพทย์มาก่อนแล้วนำมาฝึกสอนแบบจำลองโดยใช้การถ่ายทอดองค์ความรู้และเปรียบเทียบประสิทธิภาพระหว่างสถาปัตยกรรม EfficientNetB7 และ ResNet50V2 โดยมีการปรับค่าไฮเปอร์พารามิเตอร์อันได้แก่อัลกอริทึมที่ใช้ในการหาค่าที่ดีที่สุด อัตราการเรียนรู้ และจำนวนตัวอย่างที่แบบจำลองใช้ในการปรับค่าน้ำหนักของข้อมูลในการฝึกสอนแต่ละรอบที่แตกต่างกัน แบบจำลองที่มีความแม่นยำในการทำนายมากที่สุดนั้นใช้สถาปัตยกรรม EfficientNetB7 ใช้อัลกอริทึม SGD ในการหาค่าที่ดีที่สุด อัตราการเรียนรู้ 0.001 และใช้จำนวนตัวอย่างที่แบบจำลองใช้ในการปรับค่าน้ำหนักของข้อมูลในการฝึกสอนแต่ละรอบเท่ากับ 100 โดยแบบจำลองนี้มีความแม่นยำ 79% ความเที่ยงตรง 74% และความไว 90% ผลของการทดลองนี้ชี้ให้เห็นถึงศักยภาพในการพัฒนาเครื่องมือที่ใช้ช่วยตรวจจับความเจ็บปวดโดยปัญญาประดิษฐ์เพื่อพัฒนาคุณภาพชีวิตของสัตว์ได้ต่อไปในอนาคต

คำสำคัญ : โครงข่ายประสาทเทียมแบบคอนโวลูชัน, ความเจ็บปวดของแมว, การเรียนรู้เชิงลึก

|                |  |
|----------------|--|
| Title          | FELINE FEELINGS UNLEASHED: HARNESSING DEEP<br>LEARNING THROUGH PHOTOS FOR CAT PAIN DETECTION |
| Author         | CHUTIMON NAMBOONLUE  |
| Degree         | MASTER OF SCIENCE  |
| Academic Year  | 2023   |
| Thesis Advisor | Vera Sa-ing , Ph.D.  |

Effective pain detection in cats, who cannot verbally communicate, is challenging and crucial. This study analyzed the performances of pre-trained convolutional neural networks, EfficientNetB7 and ResNet50V2, for classifying feline pain. The dataset consists of 57 images per category, labeled as 'pain' or 'no pain' by Thai veterinarians. The images were preprocessed and run through various configurations, differing in batch sizes and learning rates, with ImageNet weights as the initial training parameters. The models were evaluated based on accuracy, precision, and recall. The most effective model, which utilized EfficientNetB7 as the backbone architecture, a SGD optimizer with a learning rate of 0.001, and a batch size of 100, achieved 79% accuracy, 74% precision, and 90% recall. These findings demonstrated the potential of deep learning for non-verbal pain detection in veterinary settings, especially with the high recall rate essential for identifying animals in distress. This research opens avenues for integrating such AI models into veterinary practice, and enhancing animal welfare.

Keyword : convolutional neural network, cat pain detection, Deep learning

## กิตติกรรมประกาศ

งานวิจัยชิ้นนี้ของข้าพเจ้าจะไม่สามารถเกิดขึ้นได้หากไม่มีสพ.ญ. พีรดา วีรังคบุตร สัตวแพทย์ผู้เชี่ยวชาญผู้อยู่เบื้องหลังการประเมินความเจ็บปวดของแมว และเป็นผู้วิเคราะห์ข้อมูลภาพถ่ายเพื่อนำมาฝึกสอนแบบจำลอง และรายงานฉบับนี้จะไม่สามารถสำเร็จลุล่วงไปได้เช่นกันหากไม่มีอ.ดร.วีระ สอิ้ง ผู้คอยผลักดัน ให้ความช่วยเหลือ ให้คำแนะนำที่ดีและให้กำลังใจข้าพเจ้าเสมอมา ขอขอบคุณบุคคลทั้งสองเป็นอย่างสูงมา ณ ที่นี้

ขอขอบคุณคณาจารย์ในภาควิชาวิทยาการข้อมูล คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ ทั้งในแง่ของการเป็นผู้มอบความรู้ และการเป็นผู้เปิดโลกทัศน์ให้ข้าพเจ้าได้เข้าถึงศาสตร์แห่งการวิเคราะห์ข้อมูลในหลากหลายแง่มุมอย่างไม่เคยทราบว่ามีอยู่มาก่อน

ขอขอบคุณน.สพ.พนัส วงศ์นนทิ สำหรับการช่วยประสานงานกับโรงพยาบาลสัตวแพทย์ 4 ผู้เอื้อเพื่อให้ข้าพเจ้าเก็บข้อมูลภาพถ่ายหน้าแมวด้วยตนเอง

ท้ายที่สุดนี้ ขอขอบคุณลาเต้ ลาเกอร์ ลามี และซีต้า แมวทั้งสี่ผู้เป็นแรงบันดาลใจให้ข้าพเจ้าทำงานวิจัยชิ้นนี้ขึ้นมา

ชุตินันท์ นามบุญลือ

## สารบัญ

|   | หน้า |
|---|------|
| บทคัดย่อภาษาไทย .....   | ง    |
| บทคัดย่อภาษาอังกฤษ .....  | จ    |
| กิตติกรรมประกาศ.....  | ฉ    |
| สารบัญ .....  | ช    |
| สารบัญตาราง.....  | ญ    |
| สารบัญรูปภาพ .....  | ฎ    |
| บทที่ 1 บทนำ.....   | 1    |
| 1.1 ที่มาและความสำคัญของงานวิจัย.....   | 1    |
| 1.2 วัตถุประสงค์ของงานวิจัย .....   | 3    |
| 1.3 เป้าหมายและขอบเขตของงานวิจัย .....  | 3    |
| 1.4 ขั้นตอนในการทำงานวิจัย .....  | 3    |
| 1.5 ประโยชน์ของการทำงานวิจัย.....   | 3    |
| บทที่ 2 ทบทวนวรรณกรรม และงานวิจัยที่เกี่ยวข้อง .....  | 4    |
| 2.1 ทฤษฎีการวิเคราะห์ข้อมูลด้วยหลักการทางคอมพิวเตอร์.....   | 4    |
| 2.1.1 การเรียนรู้ของเครื่อง (Machine Learning) .....  | 4    |
| 2.1.2 โครงข่ายประสาทเทียม (Artificial Neural Network) .....   | 5    |
| 2.1.3 การเรียนรู้เชิงลึก (Deep Learning).....   | 5    |
| 2.1.4 โครงข่ายประสาทคอนโวลูชัน (Convolutional Neural Network) .....   | 7    |
| 2.1.5 การถ่ายทอดองค์ความรู้ (Transfer Learning) และตัวอย่างแบบจำลองที่ได้รับการฝึกสอนมาแล้ว (pre-trained model) ..... | 12   |
| 2.1.6 การเตรียมข้อมูล (Data Pre-processing) .....   | 14   |



|   |    |
|---|----|
| 2.1.7 การสร้างแบบจำลอง (Model Development) .....  | 14 |
| 2.2 งานวิจัยที่เกี่ยวข้องกับอาการแสดงเมื่อแมวมีความเจ็บปวด .....  | 24 |
| 2.2.1 Validation of the English version of the UNESP-Botucatu multidimensional composite pain scale for assessing postoperative pain in cats [4]..... | 24 |
| 2.2.2 Preliminary appraisal of the reliability and validity of the Colorado State University Feline Acute Pain Scale [28].....                        | 24 |
| 2.2.3 Facial expressions of pain in cats: the development and validation of a Feline Grimace Scale [5] .....  | 25 |
| 2.2.4 Agreement and reliability of the Feline Grimace Scale among cat owners, veterinarians, veterinary students and nurses [29] .....                | 26 |
| 2.2.5 Geometric morphometrics for the study of facial expressions in non-human animals, using the domestic cat as an exemplar [6].....                | 26 |
| 2.2.6 Automated recognition of pain in cats [7].....  | 27 |
| 2.2.7 Explainable automated pain recognition in cats [8] .....  | 27 |
| บทที่ 3 กระบวนการ และวิธีการดำเนินการวิจัย .....  | 29 |
| 3.1 การรวบรวมข้อมูล (Data Acquisition).....   | 29 |
| 3.2 การเตรียมข้อมูล (Data Preprocessing) .....  | 31 |
| 3.3 การสร้างและปรับปรุงแบบจำลอง (Modeling) .....  | 32 |
| 3.4 การประเมินแบบจำลอง (Evaluation) .....   | 33 |
| บทที่ 4 การทดลอง และผลลัพธ์ของการวิจัย .....  | 35 |
| 4.1 การทดลองเบื้องต้น.....  | 35 |
| 4.1.1 การแบ่งกลุ่มข้อมูล .....  | 35 |
| 4.1.2 การเพิ่มจำนวนข้อมูลด้วยการดัดแปลงภาพ (data augmentation) .....  | 36 |
| 4.1.3 การปรับค่าต่างๆ ในการฝึกสอนแบบจำลอง .....   | 36 |

|  |    |
|--|----|
| 4.2 การทดลองหลังจากการปรับชุดข้อมูล .....                            | 38 |
| 4.2.1 การปรับชุดข้อมูลและแบ่งกลุ่มข้อมูล.....                        | 38 |
| 4.2.2 การเพิ่มจำนวนข้อมูลด้วยการตัดแปลงภาพ .....                     | 38 |
| 4.2.3 การปรับค่าต่างๆ ในการฝึกสอนแบบจำลอง .....                      | 38 |
| 4.3 การอธิบายความสำคัญของพีเจอร์รี่ในแบบจำลอง.....                   | 42 |
| บทที่ 5 การสรุปผลการวิจัย การอภิปรายผล และข้อเสนอแนะของการวิจัย..... | 44 |
| 5.1 การแบ่งกลุ่ม pain และ no pain .....                              | 44 |
| 5.2 การอธิบายความสำคัญของพีเจอร์รี่ในแบบจำลอง.....                   | 44 |
| 5.3 ข้อจำกัดของงานวิจัย .....  | 46 |
| 5.4 ข้อเสนอแนะสำหรับงานวิจัย.....                                    | 48 |
| บรรณานุกรม .....   | 50 |
| ประวัติผู้เขียน.....   | 55 |

## สารบัญตาราง

|   | หน้า |
|---|------|
| ตาราง 1 ผลลัพธ์จากการทดลองเบื้องต้น.....                                      | 37   |
| ตาราง 2 ผลลัพธ์จากการทำนายด้วยแบบจำลองที่มีโครงสร้างเป็น EfficientNetB7 ..... | 39   |
| ตาราง 3 ผลลัพธ์จากการทำนายด้วยแบบจำลองที่มีโครงสร้างเป็น ResNet50v2 .....     | 40   |



## สารบัญรูปภาพ

|   | หน้า |
|---|------|
| ภาพประกอบ 1 ภาพแสดงมัดกล้ามเนื้อแสดงสีหน้าของแมว.....   | 1    |
| ภาพประกอบ 2 ภาพแสดงโครงสร้างโครงข่ายประสาทเทียม.....  | 5    |
| ภาพประกอบ 3 การสกัดคุณลักษณะเฉพาะในชั้นต่างๆ ของโครงข่ายประสาทเทียม.....  | 6    |
| ภาพประกอบ 4 ดัดแปลงจากที่มา.....  | 7    |
| ภาพประกอบ 5 เมทริกซ์สองมิติของภาพขาวดำ.....   | 8    |
| ภาพประกอบ 6 ภาพขาวดำขนาด 30 * 40 พิกเซลที่แทนค่าด้วยเมทริกซ์สองมิติด้านบน.....                                  | 8    |
| ภาพประกอบ 7 ภาพ RGB ที่ประกอบด้วยเมทริกซ์ค่าสีแดง เขียว และน้ำเงินซ้อนทับกัน.....                               | 9    |
| ภาพประกอบ 8 กระบวนการคอนโวลูชัน.....  | 9    |
| ภาพประกอบ 9 การตรวจจับขอบเขต (edge detection) ของวัตถุด้วยฟิลเตอร์ดิงภาพ.....                                   | 10   |
| ภาพประกอบ 10 แสดงการทำ max pooling และ average pooling.....   | 11   |
| ภาพประกอบ 11 แผนผังพีเจอร์จากชั้นต่างๆ ของแบบจำลอง.....   | 11   |
| ภาพประกอบ 12 การเปรียบเทียบการฝึกสอนอัลกอริทึมใหม่ทั้งหมด (training from scratch) กับการถ่ายทอดองค์ความรู้..... | 13   |
| ภาพประกอบ 13 ภาพแสดงสถาปัตยกรรมของแบบจำลอง VGG-16.....  | 13   |
| ภาพประกอบ 14 การทำงานของโหนด.....   | 15   |
| ภาพประกอบ 15 แสดงการทำนายด้วยฟังก์ชัน sigmoid.....  | 17   |
| ภาพประกอบ 16 Gradient Descent.....  | 18   |
| ภาพประกอบ 17 แสดงความสัมพันธ์ระหว่างโหนดในชั้นต่างๆ ของโครงข่าย.....  | 22   |
| ภาพประกอบ 18 กราฟแสดงฟังก์ชันกระตุ้น.....   | 24   |
| ภาพประกอบ 19 การประเมินความเจ็บปวดด้วย Feline Grimace Scale.....  | 25   |
| ภาพประกอบ 20 principal components จากจุดอ้างอิงบนหน้าแมว.....   | 27   |

|   |    |
|---|----|
| ภาพประกอบ 21 การรวบรวมข้อมูล .....  | 29 |
| ภาพประกอบ 22 ภาพแสดงพีเจอรืบนไบหน้าแมวที่ต้องการนำมาฝึกสอน.....                       | 30 |
| ภาพประกอบ 23 ตัวอย่างภาพถ่ายที่สามารถนำมาฝึกสอนแบบจำลอง .....                         | 30 |
| ภาพประกอบ 24 ตัวอย่างเกณฑ์การคัดออก .....   | 31 |
| ภาพประกอบ 25 การหมุน จัดกึ่งกลาง และตัดบางส่วนของภาพ .....                            | 32 |
| ภาพประกอบ 26 การเพิ่มข้อมูลฝึกสอน.....  | 32 |
| ภาพประกอบ 27 การทดลองฝึกสอนแบบจำลองด้วยการถ่ายทอดองค์ความรู้.....                     | 33 |
| ภาพประกอบ 28 การปรับไฮเปอร์พารามิเตอร์ในการทดลองเบื้องต้น .....                       | 36 |
| ภาพประกอบ 29 การปรับไฮเปอร์พารามิเตอร์ในการทดลองหลังจากการปรับชุดข้อมูล.....          | 38 |
| ภาพประกอบ 30 Confusion matrix แสดงผลลัพธ์จากแบบจำลองที่ดีที่สุด.....                  | 41 |
| ภาพประกอบ 31 ค่าความแม่นยำในการฝึกสอนแบบจำลอง.....                                    | 42 |
| ภาพประกอบ 32 ค่าความสูญเสียในการฝึกสอนแบบจำลอง.....                                   | 42 |
| ภาพประกอบ 33 ตัวอย่างของภาพในชุดข้อมูลทดสอบที่มีการเน้นจุดสนใจโดย Grad-CAM .....      | 43 |
| ภาพประกอบ 34 สีแสดงระดับความสำคัญของแต่ละบริเวณของภาพในการทำนาย.....                  | 43 |
| ภาพประกอบ 35 ตัวอย่างการอธิบายภาพที่แบบจำลองทำนายถูกต้อง .....                        | 45 |
| ภาพประกอบ 36 ตัวอย่างการอธิบายภาพที่แบบจำลองทำนายผิด .....                            | 46 |
| ภาพประกอบ 37 ภาพความสัมพันธ์ระหว่างรอบในการเรียนรู้และค่าความสูญเสียของชุดข้อมูล..... | 47 |
| ภาพประกอบ 38 ตัวอย่างภาพแมว .....   | 48 |

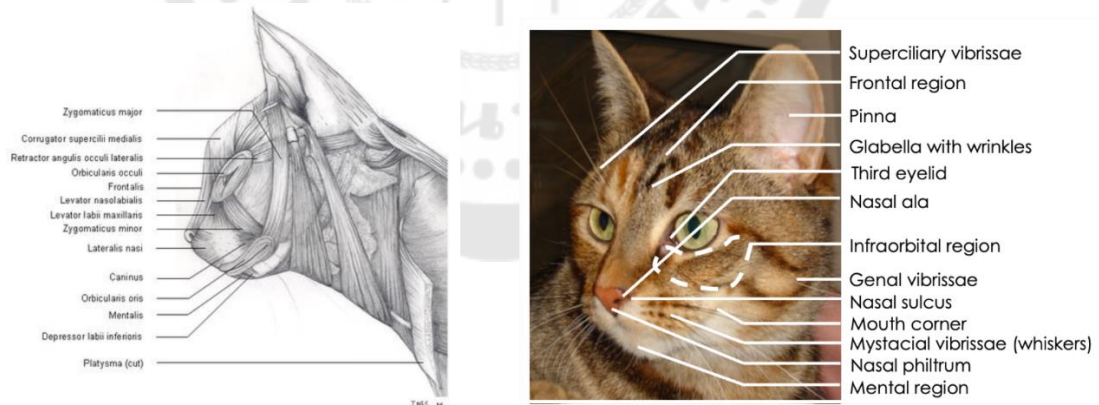
# บทที่ 1

## บทนำ

### 1.1 ที่มาและความสำคัญของงานวิจัย

ในภาวะปกติ มนุษย์สามารถถ่ายทอดอารมณ์ความรู้สึกออกมาทางสีหน้าได้ หนึ่งในความรู้สึกนั้นก็คือความเจ็บปวด ซึ่งการแสดงสีหน้าดังกล่าวเกิดจากการขยับตัวของกล้ามเนื้อบางกลุ่มบนใบหน้า กลุ่มกล้ามเนื้อเหล่านี้จะถูกระบุเรียกว่า Action Unit

ในปี 1978 Ekman และคณะ [1] ได้ทำการศึกษาเกี่ยวกับความสัมพันธ์ระหว่างการทำงานของ Action Unit กลุ่มต่างๆ กับอารมณ์ความรู้สึก และเรียกระบบที่แสดงความสัมพันธ์นี้ว่า Facial Action Coding System (FACS) ซึ่งได้ถูกนำไปใช้อ้างอิงในการศึกษาวิจัยเรื่องการวิเคราะห์อารมณ์ความรู้สึกของมนุษย์ต่ออีกมากมาย [2] จึงได้มีผู้นำแนวคิดนี้ไปประยุกต์ใช้สร้างระบบที่แสดงความสัมพันธ์ระหว่าง Action Unit ต่างๆ ดังตัวอย่างในภาพประกอบ 1 กับการแสดงสีหน้าของสัตว์เพื่อนำมาใช้ในการศึกษาพฤติกรรมที่เกี่ยวข้องกับสีหน้าต่างๆ สัตว์แต่ละชนิดจะมีระบบของตัวเองอิงตามกายวิภาคของสัตว์ชนิดนั้นๆ เช่น ระบบในแมวเรียกว่า CatFACS [3]



ภาพประกอบ 1 ภาพแสดงมัดกล้ามเนื้อแสดงสีหน้าของแมว

ที่มา : C. C. Caeiro, A. M. Burrows, และ B. M. Waller, "Development and application of CatFACS: Are human cat adopters influenced by cat facial expressions?", Applied Animal Behaviour Science, ปี 189, น . 66–78, เม . ย . 2017, doi: 10.1016/j.applanim.2017.01.005.

ระบบ UNESP-Botucatu multidimensional composite pain scale (MCPS) [4] เป็นระบบมาตรฐานระบบแรกที่ใช้ประเมินความเจ็บปวดในแมว โดยใช้การพิจารณาจากหลายมาตรวัด อาทิ การตรวจร่างกาย การสังเกตการณ์ สัญญาณชีพ ซึ่งการพิจารณาเหล่านี้ต้องอาศัยความชำนาญของผู้ประเมินเพื่อให้ได้ค่าความเจ็บปวดที่แม่นยำ จึงมีผู้นำระบบ FACS ในสัตว์ (animalFACS) ถูกนำไปศึกษาต่อในการหาความสัมพันธ์ระหว่างสีหน้าที่แสดงออกมาและความเจ็บปวด เพื่อสร้างมาตรวัดความเจ็บปวด (pain scale) ที่ประเมินจากสีหน้าของสัตว์ เรียกว่า Grimace Scale [5] เป็นการสังเกตอวัยวะหลักๆ บนหน้าแทนที่จะเป็นจุดเกาะกล้ามเนื้อต่างๆ ทำให้ใช้งานได้ง่าย ไม่จำเป็นต้องเป็นผู้เชี่ยวชาญก็สามารถประเมินได้ แต่กระนั้นเนื่องจากการประเมินโดยการสังเกตการณ์นั้นอาจก่อให้เกิดอคติในการประเมินได้ ปี 2019 Lauren R. Finka และ [6] คณะจึงได้ศึกษาการแสดงสีหน้าของแมวเมื่อมีความเจ็บปวดโดยการพล็อตจุดอ้างอิง 48 จุดบนภาพถ่ายหน้าแมวเพศเมีย 29 ตัวที่เข้ารับการผ่าตัดทำหมัน อิงตาม CatFACS และเทียบการเปลี่ยนแปลงของจุดเหล่านี้เมื่อแมวมีความเจ็บปวดระดับที่แตกต่างกันด้วยวิธี principal components analysis นับเป็นงานแรกที่ใช้วิธีการทางสถิติเพื่อวัดการแสดงสีหน้าของแมวเมื่อมีการเจ็บปวดและรายงานออกมาเป็นข้อมูลเชิงปริมาณ หลังจากนั้นในปี 2022 Marcelo Feighelstein และคณะ [7] ได้นำชุดข้อมูลที่ใช้ในการศึกษา [6] มาสร้างแบบจำลองเพื่อทำนายความเจ็บปวดจากภาพถ่ายใบหน้าแมว โดยเปรียบเทียบประสิทธิภาพระหว่างแบบจำลอง multilayer perceptron และแบบจำลองโครงข่ายประสาทคอนโวลูชัน พบว่าความแม่นยำในการทำนายใกล้เคียงกัน และในปี 2023 Marcelo Feighelstein และคณะ [8] ได้ทดลองเพิ่มความหลากหลายของชุดข้อมูล โดยใช้เป็นแมวเพศผู้และเพศเมียที่มีความเจ็บป่วยต่างกัน 84 ตัว นำมาฝึกสอนแบบจำลอง random forest, multilayer perceptron และโครงข่ายประสาทคอนโวลูชัน พบว่าแบบจำลอง random forest ได้ผลการทดลองดีที่สุด

อย่างไรก็ดีกลุ่มตัวอย่างในงานวิจัยดังกล่าวเป็นแมวในทวีปยุโรปซึ่งอาจไม่แสดงถึงกลุ่มประชากรแมวในประเทศไทยเนื่องจากลักษณะโครงหน้าที่แตกต่างกัน และผลการทดลองในงานวิจัยมีรายงานผลจากการใช้การถ่ายทอดองค์ความรู้จากแบบจำลองที่ถูกฝึกสอนมาแล้วเพียง 1 แบบคือ ResNet50 [9] ผู้วิจัยจึงต้องการที่จะฝึกสอนแบบจำลองด้วยข้อมูลภาพแมวที่เก็บในประเทศไทยและทดลองใช้การถ่ายทอดองค์ความรู้จากแบบจำลองที่ถูกฝึกสอนมาแล้วชนิดต่างๆ กันและเปรียบเทียบประสิทธิภาพเพื่อหาแบบจำลองที่สามารถทำนายความเจ็บปวดจากภาพถ่ายใบหน้าแมวได้แม่นยำที่สุด

## 1.2 วัตถุประสงค์ของงานวิจัย

1. เพื่อออกแบบแบบจำลองโครงข่ายประสาทคอนโวลูชันในการทำนายความเจ็บปวดจากภาพถ่ายใบหน้าแมว
2. เพื่อเปรียบเทียบประสิทธิภาพของแบบจำลองที่สร้างโดยการถ่ายทอดองค์ความรู้จากแบบจำลองที่ถูกฝึกสอนมาแล้วชนิดต่างๆ กัน

## 1.3 เป้าหมายและขอบเขตของงานวิจัย

งานวิจัยนี้รวบรวมข้อมูลจากโรงพยาบาลสัตว์ คลินิกสัตวแพทย์ หน่วยทำหมันแมวจรเคลื่อนที่สังกัดกทม. ในช่วงเดือนสิงหาคม พ.ศ. 2566 ถึงเดือนมกราคม พ.ศ. 2567 ซึ่งจะถูกระบุว่ามีความเจ็บปวด (pain) หรือไม่ (no pain) โดยสัตวแพทย์ และนำมาฝึกสอนแบบจำลอง

## 1.4 ขั้นตอนในการทำงานวิจัย

1. ศึกษาการสร้างมาตรวัดความเจ็บปวดจากใบหน้าของแมว
2. ศึกษาหลักการในการสร้างแบบจำลองเพื่อจำแนกประเภทด้วยการเรียนรู้เชิงลึก
3. เก็บรวบรวมข้อมูลภาพถ่ายใบหน้าแมวจากแหล่งต่างๆ
4. นำข้อมูลภาพมาเตรียมให้อยู่ในรูปแบบที่พร้อมใช้งาน
5. นำข้อมูลภาพไปฝึกสอนแบบจำลองและปรับปรุงไฮเปอร์พารามิเตอร์
6. บันทึกและเปรียบเทียบผลการทดลอง
7. วิเคราะห์ผลการทดลอง

## 1.5 ประโยชน์ของการทำงานวิจัย

1. สามารถนำแบบจำลองไปทำนายความเจ็บปวดจากภาพถ่ายใบหน้าแมวได้
2. นำเสนอแบบจำลองที่สร้างขึ้นที่สามารถช่วยเจ้าของแมวในการตรวจจับสัญญาณที่อาจบ่งถึงความผิดปกติในร่างกายของแมวได้ในอนาคต



## บทที่ 2

### ทบทวนวรรณกรรม และงานวิจัยที่เกี่ยวข้อง

#### 2.1 ทฤษฎีการวิเคราะห์ข้อมูลด้วยหลักการทางคอมพิวเตอร์

##### 2.1.1 การเรียนรู้ของเครื่อง (Machine Learning)

การเรียนรู้ของเครื่องเป็นแขนงหนึ่งของปัญญาประดิษฐ์ (Artificial Intelligence) และวิทยาการคอมพิวเตอร์ ซึ่งมุ่งเป้าไปยังการใช้ข้อมูลที่มีและอัลกอริทึมต่างๆ เพื่อตอบคำถามบางอย่าง ซึ่งอัลกอริทึมเหล่านี้สามารถเพิ่มความถูกต้องแม่นยำของคำตอบอย่างค่อยเป็นค่อยไปเลียนแบบการเรียนรู้ของมนุษย์

การเรียนรู้ของเครื่องสามารถแบ่งได้เป็น 3 ประเภทหลัก [10] ได้แก่

1. การเรียนรู้แบบมีผู้สอน (Supervised machine learning) เป็นการเรียนรู้โดยการใช้ชุดข้อมูลที่มีผลลัพธ์ระบุไว้ (labeled data) เพื่อฝึกอัลกอริทึมให้สามารถจำแนกข้อมูล (classification) หรือทำนายผลลัพธ์ (regression) ได้อย่างแม่นยำ เมื่อป้อนข้อมูลนำเข้า (input data) ให้กับอัลกอริทึม จะเกิดการคำนวณน้ำหนักของข้อมูลต่างๆ เพื่อนำไปหาผลลัพธ์ให้ได้ใกล้เคียงกับผลลัพธ์ที่ระบุไว้มากที่สุด และนำอัลกอริทึมที่ได้ไปใช้จำแนกข้อมูลหรือทำนายผลลัพธ์จากชุดข้อมูลนำเข้าที่ไม่มีผลลัพธ์ระบุไว้ในอนาคต

2. การเรียนรู้แบบไม่มีผู้สอน (Unsupervised machine learning) เป็นการฝึกอัลกอริทึมเพื่อวิเคราะห์และจัดกลุ่มชุดข้อมูลที่ไม่มีผลลัพธ์ระบุไว้ (unlabeled data) โดยอัลกอริทึมจะค้นหาแบบแผน (pattern) ในการจัดกลุ่มของข้อมูลโดยไม่พึ่งการฝึกสอนจากมนุษย์ ความสามารถในการค้นพบความคล้ายคลึงและความแตกต่างของข้อมูลทำให้เป็นวิธีการที่เหมาะสมสำหรับการวิเคราะห์ข้อมูลเชิงการตรวจสอบ (explanatory data analysis) การสร้างกลยุทธ์การขาย (cross-selling strategies) การวิเคราะห์โครงสร้างข้อมูลลูกค้า (customer segmentation) การจดจำภาพและแบบแผน (image and pattern recognition) เป็นต้น

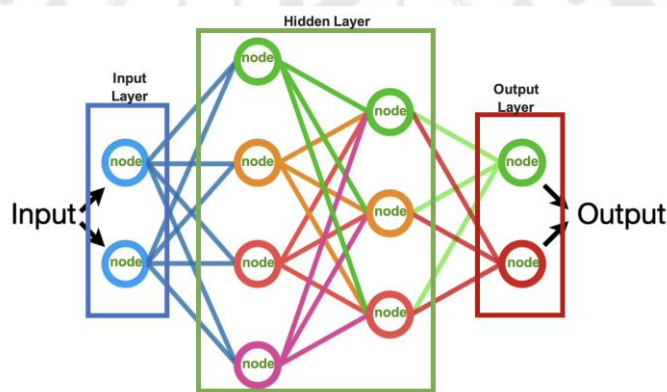
3. การเรียนรู้แบบกึ่งมีผู้สอน (Semi-supervised machine learning) เป็นการเรียนรู้โดยใช้วิธีทั้งสองแบบที่กล่าวมา นั่นก็คือการใช้ชุดข้อมูลขนาดเล็กที่มีผลลัพธ์ระบุไว้เพื่อเป็นแนวทางให้อัลกอริทึมสามารถจำแนกข้อมูลและสกัดฟีเจอร์ (feature extraction) เพื่อฝึกสอนในชุดข้อมูลขนาดใหญ่ที่ไม่มีผลลัพธ์ระบุไว้ได้ มักจะใช้การเรียนรู้ชนิดนี้ในชุดข้อมูลที่มีจำนวนผลลัพธ์ระบุไว้ไม่มากพอที่จะเกิดการเรียนรู้แบบมีผู้สอน

### 2.1.2 โครงข่ายประสาทเทียม (Artificial Neural Network)

โครงข่ายประสาทเทียมเป็นแขนงหนึ่งของการเรียนรู้ด้วยเครื่อง ประกอบด้วยชุดของอัลกอริทึมที่พยายามจะหาความสัมพันธ์ที่ซ่อนอยู่ในชุดข้อมูลโดยการเรียนรู้ที่เลียนแบบกระบวนการการคิดของมนุษย์ ซึ่งโครงสร้างของโครงข่ายประสาทเทียมจะมีลักษณะคล้ายโครงสร้างทางชีววิทยาของสมองมนุษย์ [11]

โครงข่ายประสาทเทียมมีโครงสร้างที่ใช้ในการประมวลผลข้อมูล โครงสร้างนี้ประกอบด้วยหน่วยประมวลผลหลายๆ หน่วยที่เรียกว่าเซลล์ประสาท (neurons) หรือที่นิยมเรียกว่าโหนด (node) และหน่วยในการเชื่อมต่อที่เรียกว่าเส้นเชื่อมต่อ (connections) ซึ่งเป็นส่วนประกอบที่สำคัญในการส่งสัญญาณและประมวลผลข้อมูล

โครงข่ายประสาทเทียมจะประกอบด้วยชั้นต่างๆ โดยมีชั้นต้นทาง (input layer) ชั้นซ่อน (hidden layers) และชั้นปลายทาง (output layer) ซึ่งแต่ละชั้นก็จะมีหน่วยประมวลผลและเส้นเชื่อมต่อกันให้เกิดกระบวนการประมวลผลข้อมูล ดังภาพประกอบ 2



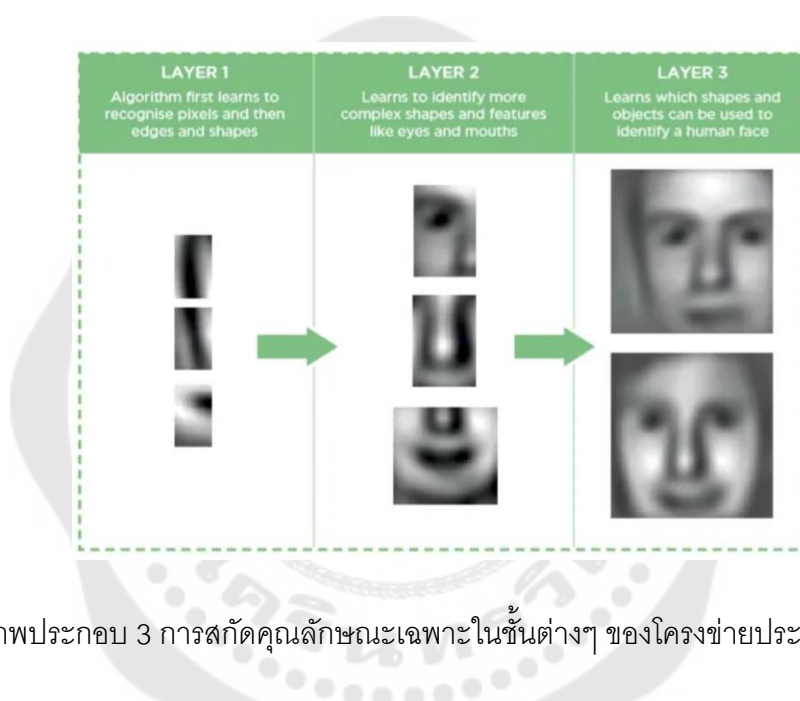
ภาพประกอบ 2 ภาพแสดงโครงสร้างโครงข่ายประสาทเทียม

ที่มา : J. Starmer, "Neural Networks Part 1: Inside the black box", StatQuest. [ออนไลน์]. Available at: <https://statquest.org/neural-networks-part-1-inside-the-black-box/>

### 2.1.3 การเรียนรู้เชิงลึก (Deep Learning)

การเรียนรู้เชิงลึกเป็นแขนงย่อยหนึ่งของโครงข่ายประสาทเทียม เป็นการเรียนรู้ของระบบคอมพิวเตอร์ที่ใช้โครงข่ายประสาทเทียมที่มีความซับซ้อนและมีจำนวนชั้นมากกว่ากระบวนการเรียนรู้ของเครื่องแบบดั้งเดิม (Traditional Machine Learning) จึงสามารถนำมาใช้แก้ปัญหาที่

ซับซ้อนและเชิงลึกขึ้นได้ เช่นการตรวจจับวัตถุ การแปลภาษา การทำระบบถามตอบอัตโนมัติ การแยกชนิดประเภทของวัตถุจากภาพ เป็นต้น คุณสมบัติหนึ่งที่ทำให้การเรียนรู้เชิงลึกเป็นที่นิยมนำมาใช้ในการประมวลผลภาพด้วยคอมพิวเตอร์ (image processing) คือความสามารถในการแยกแยะและสกัดคุณลักษณะเฉพาะของข้อมูล (feature extraction) โดยปกติแล้วหากเป็นกระบวนการเรียนรู้ของเครื่องแบบดั้งเดิมนั้น มนุษย์จะต้องเป็นผู้กำหนดคุณลักษณะของข้อมูลด้วยตนเองก่อนที่จะทำการฝึกสอนอัลกอริทึม แต่การเรียนรู้เชิงลึกสามารถเรียนรู้คุณลักษณะของข้อมูลได้ด้วยตนเอง ยกตัวอย่างเช่นปัญหาการตรวจจับใบหน้า [13]



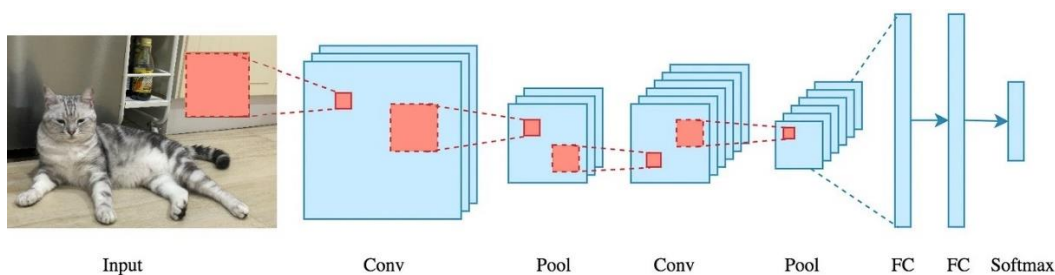
ภาพประกอบ 3 การสกัดคุณลักษณะเฉพาะในชั้นต่างๆ ของโครงข่ายประสาทเทียม

ที่มา: S. Mahapatra, "Why Deep Learning over Traditional Machine Learning?", Towards Data Science. [ออนไลน์]. Available at: <https://towardsdatascience.com/why-deep-learning-is-needed-over-traditional-machine-learning-1b6a99177063>

จากภาพตัวอย่างจะเห็นได้ว่าการเรียนรู้เชิงลึกเริ่มจากการตรวจจับความมืดและสว่างก่อนที่จะแบ่งประเภทเส้นและรูปร่างเพื่อให้สามารถรู้จำใบหน้าได้ แต่ละโหนดในโครงข่ายจะแสดงคุณลักษณะย่อยที่สุดท้ายจะถูกนำมาประมวลผลร่วมกันในภาพ 1 ภาพ โดยจะมีการให้น้ำหนักแตกต่างกันซึ่งแสดงถึงความสัมพันธ์ระหว่างโหนดหรือคุณลักษณะย่อยนั้นๆ กับผลลัพธ์ เมื่อมีการพัฒนาเพื่อเพิ่มประสิทธิภาพของแบบจำลอง จะมีการปรับเปลี่ยนของน้ำหนักเหล่านี้ไปจนกว่าจะได้แบบจำลองที่มีประสิทธิภาพที่ผู้สร้างพึงพอใจ

## 2.1.4 โครงข่ายประสาทคอนโวลูชัน (Convolutional Neural Network)

โครงข่ายประสาทคอนโวลูชันเป็นเทคนิคการเรียนรู้เชิงลึกที่มีความคล้ายคลึงกับสมองของมนุษย์ในการประมวลผลภาพ เปรียบได้กับการมองแต่ละตำแหน่งบนภาพเป็นพื้นที่ย่อยๆ และนำกลุ่มของพื้นที่ย่อยมาผสมกัน เพื่อประมวลผลว่าสิ่งที่เห็นคืออะไร [14] โดยมีโครงสร้างแบ่งเป็นชั้นดังนี้



ภาพประกอบ 4 ดัดแปลงจากที่มา

ที่มา : A. Dertat, "Applied Deep Learning - Part 4: Convolutional Neural Networks", Towards Data Science. [ออนไลน์]. Available at: <https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2>

### 1. ชั้นข้อมูลนำเข้า (Input Layer)

ข้อมูลนำเข้าซึ่งเป็นภาพในคอมพิวเตอร์จะถูกแทนด้วยเมทริกซ์ (matrix) หรืออาร์เรย์ (array) ของตัวเลขที่เรียกว่า "พิกเซล" (Pixel) ซึ่งแต่ละพิกเซลจะแทนค่าความเข้มของสีในแต่ละจุดบนภาพ โดยมีช่วงอยู่ระหว่าง 0 (เข้มที่สุด) ถึง 255 (สว่างที่สุด) ในระบบภาพขาวดำ (Grayscale) จะถูกแทนค่าด้วยเมทริกซ์สองมิติที่มีขนาดเท่ากับความกว้าง \* ความยาวของภาพเป็นหน่วยพิกเซล และมีค่าเป็นความเข้มของสีเทาของตำแหน่งนั้นๆ บนภาพตัวอย่างเช่น

```

1218 215 214 214 214 214 196 181 180 179 178 178 176 176 178 178 177 176 176 175 174 175 176 177 176 177 178 177 175 174 173 172 170 168 167 165 164 158 153 179
1214 211 210 209 210 210 192 178 177 177 183 185 179 182 184 181 179 176 174 173 172 173 174 174 175 176 176 175 174 173 172 170 169 168 166 165 163 158 153 180
1212 207 203 202 203 194 162 174 174 178 185 185 182 189 185 179 178 177 172 171 171 172 172 173 174 174 174 172 171 170 169 167 167 165 164 163 157 154 181
1207 202 196 195 188 181 180 170 171 175 176 177 181 180 201 187 178 176 177 169 169 169 170 170 171 172 173 172 171 170 169 167 166 166 165 163 162 157 154 181
1203 196 193 187 174 188 175 167 168 170 170 169 170 166 156 202 205 190 173 169 168 169 169 169 170 171 172 172 171 170 170 167 166 166 165 163 170 159 155 182
1198 194 191 174 180 186 170 164 166 167 168 168 171 172 150 143 180 218 212 193 193 183 173 169 169 170 170 171 170 171 172 168 169 171 166 176 204 172 158 183
1195 193 186 180 182 180 166 161 162 163 160 160 166 173 166 130 135 145 153 154 151 170 190 182 185 176 172 161 169 183 169 168 180 197 190 196 215 221 210 199
1146 164 182 181 172 164 157 156 159 157 152 148 137 108 86 122 130 139 130 123 115 135 153 165 157 158 147 173 204 176 167 166 169 165 167 184 198 207 227 244
1126 123 138 143 152 161 161 169 176 165 129 88 81 42 61 123 121 110 109 107 96 119 131 164 165 156 168 188 203 163 160 149 147 133 130 154 188 194 204 233
1117 107 106 103 114 134 145 166 176 198 190 126 82 106 118 119 71 35 70 101 101 104 78 100 154 178 182 193 117 93 91 120 130 124 124 135 171 199 202 232
1 88 79 77 74 81 93 102 132 145 151 175 202 173 119 110 110 88 62 75 107 125 102 38 61 123 173 194 173 67 79 91 111 125 122 127 137 165 197 201 223
1 69 68 70 82 91 88 100 108 95 89 122 175 138 113 102 108 94 103 102 81 117 130 105 105 138 173 159 199 113 108 122 118 123 120 126 150 165 165 177 202
1 82 85 95 93 81 82 93 84 61 64 70 87 118 76 88 98 102 107 108 73 87 136 122 122 150 162 138 181 155 141 144 145 150 148 151 160 176 167 171 178
1 71 82 94 92 78 83 83 64 57 54 50 72 91 80 72 97 107 102 98 78 101 128 136 157 140 148 138 185 166 166 166 165 164 161 162 166 205 203 171 161
1 76 85 91 84 76 77 67 55 56 43 46 69 79 83 70 82 102 101 82 77 88 118 144 150 138 142 156 147 85 82 83 86 88 90 123 169 215 208 180 171
1 88 95 91 78 68 66 58 54 49 40 55 73 88 81 72 57 84 100 101 105 119 138 144 147 144 166 163 95 77 76 74 73 74 75 122 172 224 211 179 169
1 88 93 75 57 58 57 56 56 51 54 62 61 95 78 69 65 55 66 87 101 117 134 138 139 144 161 113 97 75 71 76 76 60 52 121 175 226 208 182 173
1 98 80 60 50 61 68 60 58 74 69 62 53 65 85 68 62 68 58 51 59 78 77 73 84 128 190 73 96 79 55 61 72 67 48 121 176 225 205 179 169
1 89 73 54 56 82 84 52 74 84 74 69 56 47 76 80 59 64 81 81 60 63 72 99 86 74 166 61 61 64 67 48 54 43 44 125 177 222 201 173 165
1 69 57 50 66 92 80 64 86 91 101 94 77 60 52 79 90 55 69 72 79 77 67 57 57 71 113 48 53 66 81 74 51 43 51 130 179 220 195 165 160
1 57 45 49 69 85 71 67 84 95 99 86 67 66 60 48 79 117 77 56 65 60 53 52 62 77 107 43 37 39 73 70 72 51 55 133 181 218 189 157 155
1 46 45 51 66 71 60 58 74 77 76 75 85 88 71 50 46 67 124 106 64 54 53 59 69 87 103 39 32 37 41 67 83 49 47 137 182 215 184 150 150
1 40 41 43 56 57 52 50 59 61 60 74 75 64 59 59 55 57 62 85 122 109 100 94 108 139 94 42 34 43 49 49 45 48 46 141 184 211 177 145 147
1 55 56 51 48 45 40 35 44 50 51 50 56 61 57 64 61 55 47 63 80 113 170 158 167 196 174 154 128 96 63 41 48 48 47 144 185 207 170 139 144
1 85 80 82 80 78 78 74 71 63 54 41 36 36 37 40 51 65 57 60 75 96 165 160 150 176 206 182 176 178 179 167 140 85 54 148 188 202 163 134 141
1 99 98 104 107 105 106 108 109 111 103 93 79 62 47 37 37 43 42 56 68 100 163 203 181 121 135 180 196 173 176 180 187 190 162 167 189 197 155 131 137
111 112 113 113 115 117 120 118 121 121 118 104 85 70 72 89 93 65 32 65 116 157 188 181 91 110 153 196 162 174 175 178 183 190 196 190 148 126 134
114 117 119 119 123 124 128 131 133 135 137 136 131 117 104 110 122 133 130 70 24 64 114 131 137 92 54 104 175 154 162 159 165 169 157 165 177 141 123 130
1121 121 124 127 128 131 134 138 138 141 143 146 146 141 133 133 139 145 151 126 67 25 54 103 150 154 97 110 145 146 149 132 142 159 145 138 154 135 119 127
123 129 128 132 134 136 137 141 144 146 144 148 151 147 142 141 143 147 149 147 127 98 61 88 143 122 108 116 117 131 142 119 114 136 132 131 130 127 115 123

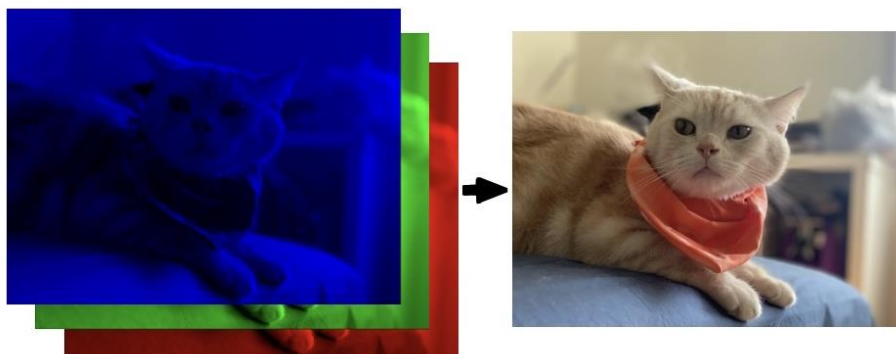
```

### ภาพประกอบ 5 เมทริกซ์สองมิติของภาพขาวดำ



### ภาพประกอบ 6 ภาพขาวดำขนาด 30 \* 40 พิกเซลที่แทนค่าด้วยเมทริกซ์สองมิติด้านบน

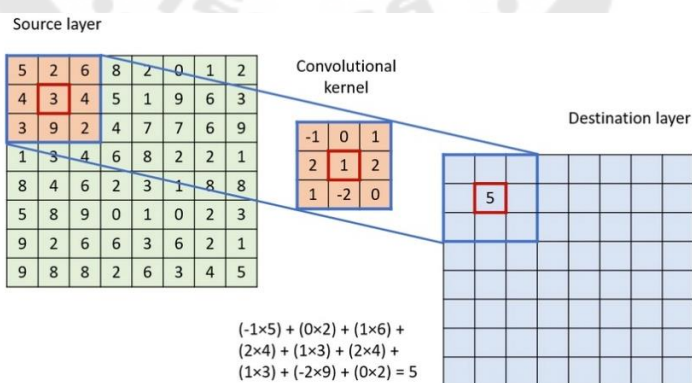
ในระบบ RGB (Red, Green, Blue) แต่ละพิกเซลจะประกอบด้วย 3 ค่า คือ ค่าสีแดง (R) ค่าสีเขียว (G) และค่าสีน้ำเงิน (B) ซึ่งจะนำเมทริกซ์ของค่าสีทั้งสามมาซ้อนกันเพื่อให้ได้สีที่เรามองเห็นในภาพ ภาพในระบบ RGB จึงถูกแทนค่าด้วยเมทริกซ์สามมิติ มิติที่เพิ่มขึ้นมาจากระบบภาพขาวดำเป็นมิติบ่งช่องสี (color channel) นั่นเอง



ภาพประกอบ 7 ภาพ RGB ที่ประกอบด้วยเมตริกซ์ค่าสีแดง เขียว และน้ำเงินซ้อนทับกัน

## 2. ชั้นคอนโวลูชัน (Convolutional Layer)

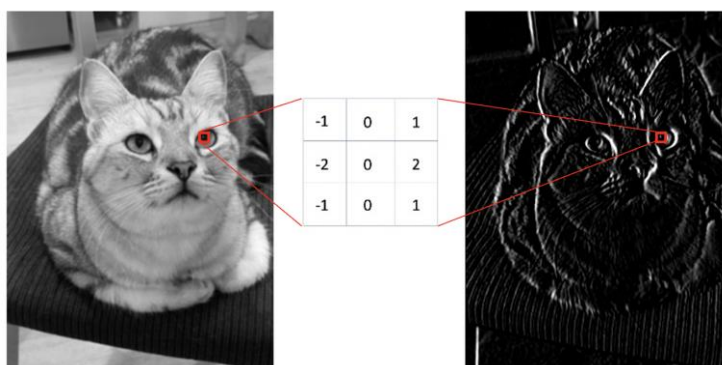
ชั้นคอนโวลูชันเป็นหัวใจสำคัญในการสร้างโครงข่ายประสาทชนิดนี้ “คอนโวลูชัน” คือกระบวนการทางคณิตศาสตร์ที่ใช้รวม (merge) ชุดสารสนเทศ (information) 2 ชุดเข้าด้วยกัน โดยจะใช้เมตริกซ์ที่เรียกว่าฟิลเตอร์ (filter) หรือเคอร์เนล (kernel) ที่มีลักษณะคล้ายหน้าต่าง (นิยมใช้เป็นรูปสี่เหลี่ยมจัตุรัส) ทาบไปบนเมตริกซ์ข้อมูลนำเข้าเพื่อหาผลรวมของการคูณค่าพิกเซลกับค่าบนฟิลเตอร์ที่มีตำแหน่งตรงกันแล้วขยับจากซ้ายไปขวาเรื่อยๆ จนครบทั้งแถว แล้วจึงเริ่มใหม่ที่ด้านซ้ายสุดของแถวที่อยู่ล่างลงมา ผลลัพธ์ที่ได้จะเป็นเมตริกซ์ที่มีขนาดเล็กกว่าหรือเท่ากับข้อมูลนำเข้า ขึ้นอยู่กับการทำ padding หรือการต่อเติมขอบให้กับข้อมูลนำเข้า [16]



ภาพประกอบ 8 กระบวนการคอนโวลูชัน

ที่มา : D. Podareanu, V. B. Codreanu, S. Aigner, C. V. Leeuwen, และ V. Weinberg, “Best Practice Guide - Deep Learning”, 2019, doi: 10.13140/RG.2.2.31564.05769.

กระบวนการคอนโวลูชันจะช่วยสกัดคุณลักษณะจำเพาะหรือฟีเจอร์ (feature) ที่ใช้ในการรู้จำวัตถุ (object recognition) ออกมา โดยผลลัพธ์จากการทำคอนโวลูชันด้วยฟิลเตอร์ 1 ชั้น จะได้เป็นแผนผังฟีเจอร์ (feature map) เมื่อเปลี่ยนค่าบนฟิลเตอร์ก็จะสามารถสกัดแผนผังฟีเจอร์ที่แตกต่างกันจากข้อมูลนำเข้า โครงข่ายประสาทคอนโวลูชันจึงประกอบไปด้วยฟิลเตอร์หลายชนิดเพื่อสกัดฟีเจอร์ต่างๆ ออกมาจากข้อมูลนำเข้าให้ได้ข้อมูลปริมาณมากเพื่อนำไปฝึกสอนอัลกอริทึมต่อไป

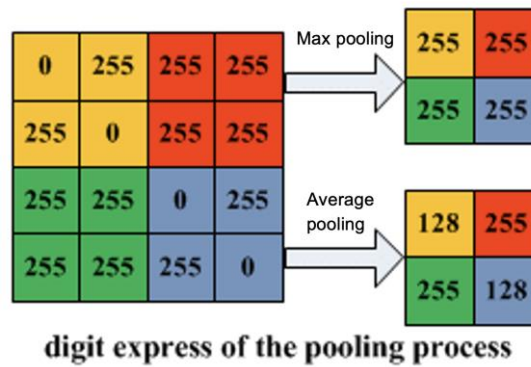


ภาพประกอบ 9 การตรวจจับขอบเขต (edge detection) ของวัตถุด้วยฟิลเตอร์ดึงภาพ

### 3. ชั้นพูลลิง (Pooling Layer)

หลังจากที่ได้แผนผังฟีเจอร์จำนวนมากจากการทำคอนโวลูชันด้วยฟิลเตอร์หลายชนิด ชั้นพูลลิงเป็นการย่อขนาดแผนผังฟีเจอร์ลง มีจุดประสงค์เพื่อลดจำนวนข้อมูลเพื่อลดภาระในการประมวลผลของคอมพิวเตอร์ ลดการใช้หน่วยความจำ และลดจำนวนของพารามิเตอร์ลงเพื่อลดภาวะ overfitting ซึ่งคือภาวะไม่เพียงประสงค์ที่เกิดขึ้นเมื่อรูปแบบการเรียนรู้ของอัลกอริทึมให้การคาดการณ์ที่ถูกต้องสำหรับข้อมูลที่ฝึก แต่ไม่ใช่สำหรับข้อมูลใหม่

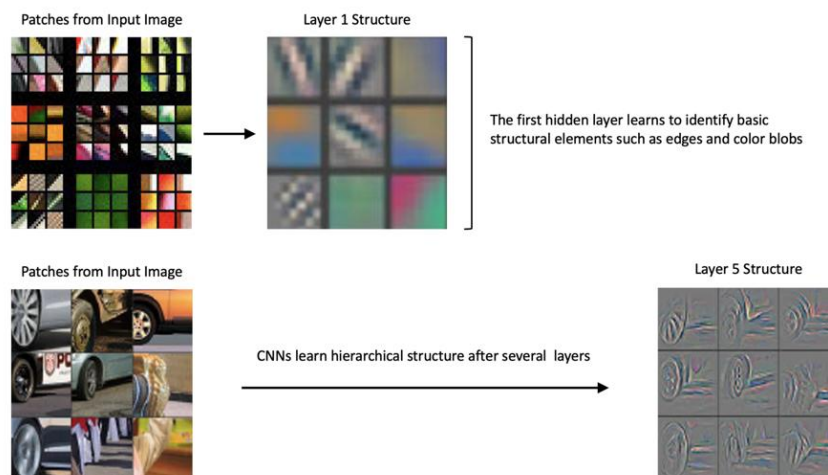
การทำพูลลิงสามารถทำได้ 3 แบบ คือ max pooling, min pooling และ average pooling โดยหลักการร่วมกันคือการกำหนดขนาดของหน้าต่างขึ้นมา (นิยมใช้เป็นขนาด  $2 \times 2$ ) เพื่อหาไปยังเมทริกซ์ของข้อมูลนำเข้าจากซ้ายไปขวาค่อยๆ กับในชั้นคอนโวลูชัน แต่ในการทำพูลลิงจะเป็นการเลือกตัวแทนจากค่าของข้อมูลนำเข้าที่มีตำแหน่งตรงกับหน้าต่าง หากเป็น max pooling จะเป็นการเลือกค่าที่สูงที่สุด, min pooling จะเป็นการเลือกค่าที่ต่ำที่สุด และ average pooling จะเป็นการหาค่าเฉลี่ยของค่าทั้งหมดที่ปรากฏในหน้าต่างนั้น [17]



ภาพประกอบ 10 แสดงการทำ max pooling และ average pooling

ที่มา : R. Qayyum, “Introduction To Pooling Layers In CNN”, Towards AI. [ออนไลน์]. Available at: <https://towardsai.net/p//introduction-to-pooling-layers-in-cnn>

ในหนึ่งโครงข่ายประสาทคอนโวลูชันมักจะมีชั้นคอนโวลูชันและชั้นพูลลิ่งสลับกันไปมา ผลลัพธ์ที่ได้คือภาพที่เล็กลงเรื่อยๆ แต่ก็มีคุณลักษณะที่ประกอบไปด้วยแผนผังพีเจอร์มากขึ้นเช่นกัน



ภาพประกอบ 11 แผนผังพีเจอร์จากชั้นต่างๆ ของแบบจำลอง

ที่มา : “Understanding Convolutional Neural Networks (CNNs): A Complete Guide”, LearnOpenCV. [ออนไลน์]. Available at: <https://learnopencv.com/understanding-convolutional-neural-networks-cnn/>



#### 4. ชั้นเชื่อมโยงสมบูรณ์ (Fully Connected Layers)

หลังจากที่ได้แผนผังพีเจอรจากชั้นก่อนหน้าแล้ว ชั้นเชื่อมโยงสมบูรณ์จะนำแผนผังทั้งหมดมาจัดให้อยู่ในรูปเมทริกซ์ 1 มิติ หรือที่เรียกว่าพีเจอรเวกเตอร์ (feature vector) ก่อนใช้เป็นข้อมูลนำเข้าเพื่อให้โหนดต่างๆ นำไปคำนวณหาน้ำหนักความสัมพันธ์ (weight) ระหว่างพีเจอรและผลลัพธ์ต่อไป โดยชั้นสุดท้ายของชั้นเชื่อมโยงสมบูรณ์มักจะเป็นอัลกอริทึม SoftMax หรือ Logistic Regression ซึ่งให้ผลลัพธ์เป็นค่าความน่าจะเป็นของกลุ่มข้อมูล

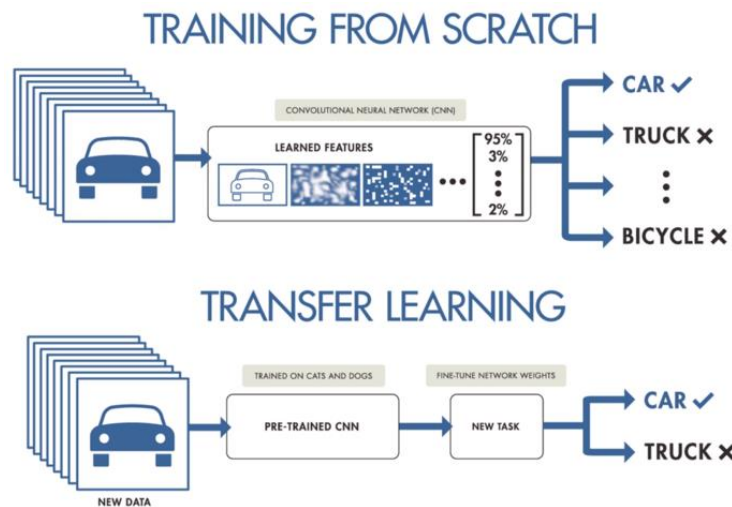
##### 2.1.5 การถ่ายทอดองค์ความรู้ (Transfer Learning) และตัวอย่างแบบจำลองที่ได้รับการฝึกสอนมาแล้ว (pre-trained model)

จากที่ได้กล่าวไปในหัวข้อโครงข่ายประสาทคอนโวลูชัน จะเห็นได้ว่าแบบจำลองการเรียนรู้เชิงลึก 1 โมเดล ต้องใช้พีเจอรในการฝึกสอนจำนวนมาก ซึ่งเกิดจากกระบวนการสกัดพีเจอรในชั้นคอนโวลูชันและชั้นพูลลิงสลับไปมา การคำนวณหาน้ำหนักบนฟิลเตอร์หลายๆ ชั้นเพื่อนำมาสกัดให้ได้พีเจอรที่ดีพอที่จะใช้ฝึกสอนอัลกอริทึมในชั้นเชื่อมโยงสมบูรณ์ได้อย่างมีประสิทธิภาพนั้นอาจใช้ข้อมูลตั้งต้นปริมาณมหาศาลและใช้เวลามากในการฝึกสอน

การถ่ายทอดองค์ความรู้ในภาพประกอบ 12 เป็นการนำแบบจำลองที่ได้รับการฝึกสอนมาแล้ว (pre-trained model) มักจะใช้แบบจำลองที่ถูกฝึกสอนด้วยข้อมูลปริมาณมหาศาลที่มีความเกี่ยวข้องกับแบบจำลองที่จะสร้าง) มาปรับแต่งในบางชั้น เช่น นำแบบจำลองที่ใช้จำแนกวัตถุต่างๆ มาปรับแก้ในชั้นเชื่อมโยงสมบูรณ์ เพื่อสร้างแบบจำลองที่ใช้แยกพันธุ์แมวจากภาพถ่าย ก็จะสามารถลดเวลาในการฝึกสอนแบบจำลองใหม่ในการสร้างฟิลเตอร์เพื่อสกัดพีเจอรในชั้นคอนโวลูชันและพูลลิงลงได้

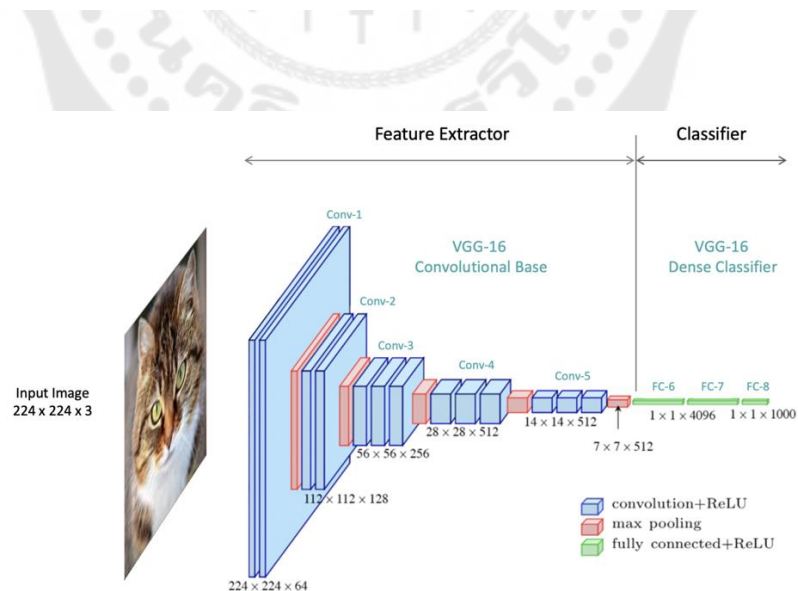
แบบจำลองที่ได้รับการฝึกสอนมาแล้วที่สามารถนำไปถ่ายทอดองค์ความรู้เพื่อสร้างแบบจำลองจำแนกภาพถ่ายนั้นในปัจจุบันมีตัวเลือกหลายหลาย ซึ่งในแต่ละแบบจำลองจะประกอบด้วยโครงสร้างสถาปัตยกรรมที่แตกต่างกัน

ภาพประกอบ 13 เป็นภาพที่แสดงให้เห็นสถาปัตยกรรมของแบบจำลอง VGG-16 มิติของแต่ละชั้นย่อยของชั้นคอนโวลูชัน (สีฟ้า) จะแทนความสูงภาพ \* ความกว้างภาพ \* จำนวนฟิลเตอร์ เมื่อผ่านไปในพื้นที่ลึกขึ้นจะมีการทำพูลลิง (สีแดง) ทำให้ความสูงและความกว้างของภาพเล็กลง หลังจากนั้นจะมีการเพิ่มจำนวนฟิลเตอร์ขึ้น ทำให้ได้แผนผังพีเจอรที่มีความลึกมากขึ้นก่อนถูกนำไปแปลงให้เป็นเมทริกซ์ 1 มิติ (พีเจอรเวกเตอร์) เพื่อนำไปป้อนเป็นข้อมูลนำเข้าสำหรับชั้นเชื่อมโยงสมบูรณ์ (สีเขียว) ต่อไป



ภาพประกอบ 12 การเปรียบเทียบการฝึกสอนอัลกอริทึมใหม่ทั้งหมด (training from scratch) กับ การถ่ายทอดองค์ความรู้

ที่มา: H. D. Regua, "Introducing Transfer Learning as Your Next Engine to Drive Future Innovations", Medium. [ออนไลน์]. Available at: <https://medium.datadriveninvestor.com/introducing-transfer-learning-as-your-next-engine-to-drive-future-innovations-5e81a15bb567>



ภาพประกอบ 13 ภาพแสดงสถาปัตยกรรมของแบบจำลอง VGG-16

ที่มา : “Understanding Convolutional Neural Networks (CNNs): A Complete Guide”, LearnOpenCV. [ออนไลน์]. Available at: <https://learnopencv.com/understanding-convolutional-neural-networks-cnn/>

### 2.1.6 การเตรียมข้อมูล (Data Pre-processing)

ก่อนที่จะนำข้อมูลที่เก็บรวบรวมได้นั้นไปฝึกสอนแบบจำลอง จะต้องมีการเตรียมข้อมูลให้อยู่ในรูปแบบพร้อมใช้งาน ซึ่งในบริบทของการสร้างแบบจำลองเพื่อจำแนกชนิดนั้นคือการปรับขนาดรูปภาพให้สอดคล้องกับมิติของข้อมูลนำเข้าที่ถูกกำหนดในแบบจำลอง เช่น หากจะใช้ VGG-16 ก็ควรจะปรับขนาดภาพที่จะนำไปฝึกสอนให้มีขนาด  $224 * 224$  และเป็นภาพ RGB เป็นต้น

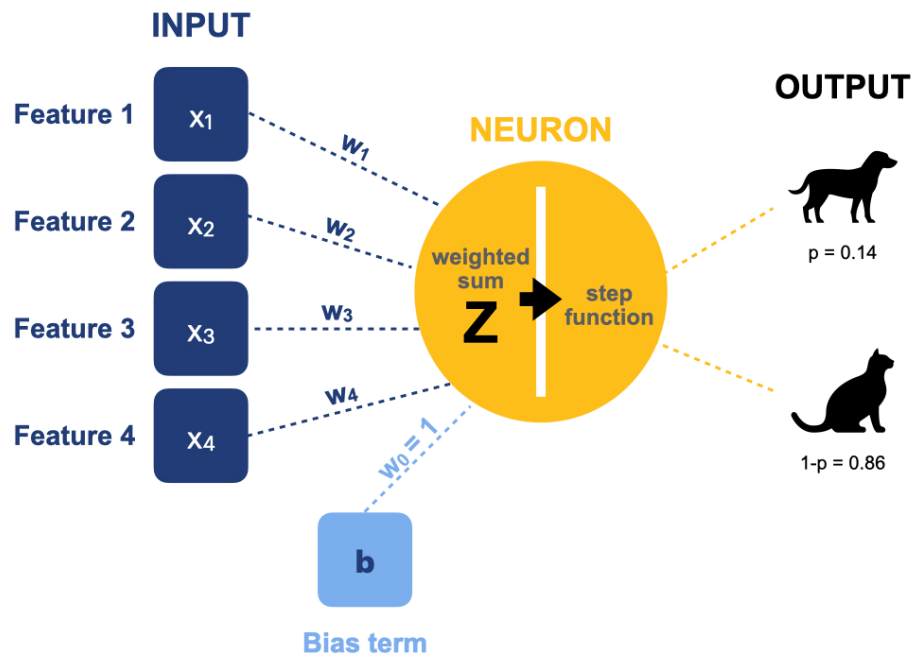
นอกเหนือจากการปรับรูปแบบของข้อมูลแล้ว ยังมีเทคนิคการเพิ่มข้อมูล (data augmentation) ซึ่งมีประโยชน์ในการฝึกสอนแบบจำลองเนื่องจากการเพิ่มจำนวนชุดข้อมูลฝึกสอน ส่งผลให้ประสิทธิภาพในการทำนายของแบบจำลองดีขึ้น

ในปี 2021 Ramaprasad และคณะ [21] ได้นำเสนอประสิทธิภาพที่ดีขึ้นในการแยกภาพสุนัขและแมวด้วยแบบจำลองโครงข่ายประสาทเทียมชนิดคอนโวลูชันหลังจากใช้เทคนิคการเพิ่มข้อมูลโดยการหมุน การย่อขยาย และการพลิกรูป และสร้างแบบจำลองด้วยหลักการการถ่ายทอดองค์ความรู้ ผู้จัดทำได้เทียบประสิทธิภาพแบบจำลอง VGG-16 และ ResNet50 ก่อนและหลังการเพิ่มข้อมูล พบว่าความแม่นยำในการทำนายชุดข้อมูลทดสอบของ VGG-16 เพิ่มจาก 80% เป็น 88% และของ ResNet50 เพิ่มจาก 82% เป็น 90% อีกทั้งความแตกต่างระหว่างความแม่นยำในการทำนายชุดข้อมูลฝึกสอน (training accuracy) และชุดข้อมูลทดสอบ (test accuracy) ลดลงหลังจากทำการเพิ่มข้อมูล ผู้ทดลองจึงสรุปว่าจากการทดลองนี้ การเพิ่มข้อมูลสามารถช่วยลดปัญหาโอเวอร์ฟิตติ้ง (overfitting) ได้

### 2.1.7 การสร้างแบบจำลอง (Model Development)

#### 1. การทำงานของโหนด

หากพิจารณาการทำงาน (ในที่นี้คือการจำแนกวัตถุ 2 ประเภท) ของโหนด 1 ตัว สามารถเขียนความสัมพันธ์ระหว่างข้อมูลนำเข้า (input) โหนด (neuron) และผลลัพธ์ (output) ได้คร่าวๆ ดังรูป



ภาพประกอบ 14 การทำงานของโหนด

จะเห็นได้ว่าข้อมูลนำเข้านั้นเชื่อมอยู่กับโหนดด้วยน้ำหนักของความสัมพันธ์ (weight) ที่แตกต่างกัน ซึ่งชี้ว่าเมื่อค่าของพีเจอรันนั้นเปลี่ยนไปจะทำให้ผลลัพธ์เปลี่ยนไปด้วยมากน้อยเพียงใด และมีค่าอคติ (bias term) ที่เชื่อมอยู่กับโหนดด้วยน้ำหนักเท่ากับ 1 โหนดจะรับข้อมูลทั้งหมดมาอยู่ในรูปตัวแปร  $z$  ซึ่งเป็นผลรวมของค่าอคติและผลคูณน้ำหนักและค่าของแต่ละพีเจอร์ เขียนในรูปสมการได้ดังนี้

$$z = \sum_{i=1}^n w_i x_i + b \quad (1)$$

เมื่อได้ค่า  $z$  มาแล้ว จะถูกนำไปป้อนให้ step function ซึ่งสามารถเลือกได้ตามประเภทและจำนวนของผลลัพธ์ที่ต้องการ ในที่นี้ขอยกตัวอย่างการจำแนกวัตถุ 2 ประเภท (binary classification) ด้วยวิธี logistic regression ซึ่งเขียนในรูปฟังก์ชัน sigmoid ดังนี้

$$f(z) = \frac{1}{1 + e^{-z}} \quad (2)$$

จะได้ผลลัพธ์ออกมาเป็นค่าความน่าจะเป็น จากภาพสามารถอธิบายได้ว่าในแบบจำลองนี้ เมื่อกำหนดให้ค่า Feature 1 =  $x_1$ , Feature 2 =  $x_2$ , Feature 3 =  $x_3$  และ Feature 4 =  $x_4$  นั้น จะได้ค่าความน่าจะเป็น ( $p$ ) ที่ตัวอย่างนี้จะเป็นสุนัขเท่ากับ 0.14 และเนื่องจากเป็นการจำแนกวัตถุเพียง 2 ประเภท (แยกสุนัขและแมว) ค่าความน่าจะเป็นที่ตัวอย่างนี้จะเป็นแมวจึงเท่ากับ  $1-p = 0.86$  ซึ่งมากกว่าความน่าจะเป็นสุนัข จึงสามารถทำนายได้ว่าตัวอย่างนี้เป็นแมว

## 2. ฟังก์ชันสูญเสีย (Loss Function)

เราสามารถบอกได้ว่าแบบจำลองหนึ่งทำนายผลเพี้ยนไปไกลจากผลลัพธ์จริง (ground truth) เท่าใดด้วยฟังก์ชันสูญเสีย แบบจำลองต่างประเภทกัน อาจมีฟังก์ชันสูญเสียที่ต่างกัน ยิ่งค่าฟังก์ชันสูญเสียน้อยเท่าไร แปลว่าการทำนายของแบบจำลองคล้ายผลลัพธ์จริงมากเท่านั้น หากแบบจำลองทำนายได้เหมือนผลลัพธ์จริง 100% ค่าฟังก์ชันสูญเสียจะเท่ากับ 0

จากตัวอย่างด้านบนเป็นการจำแนกวัตถุ 2 ประเภท ซึ่งจะใช้ฟังก์ชันสูญเสียล็อกลอส (log loss) หรือที่เรียกอีกชื่อว่าเอนโทรปีครอสเอนโทรปี (binary cross-entropy) ซึ่งมีสมการดังนี้

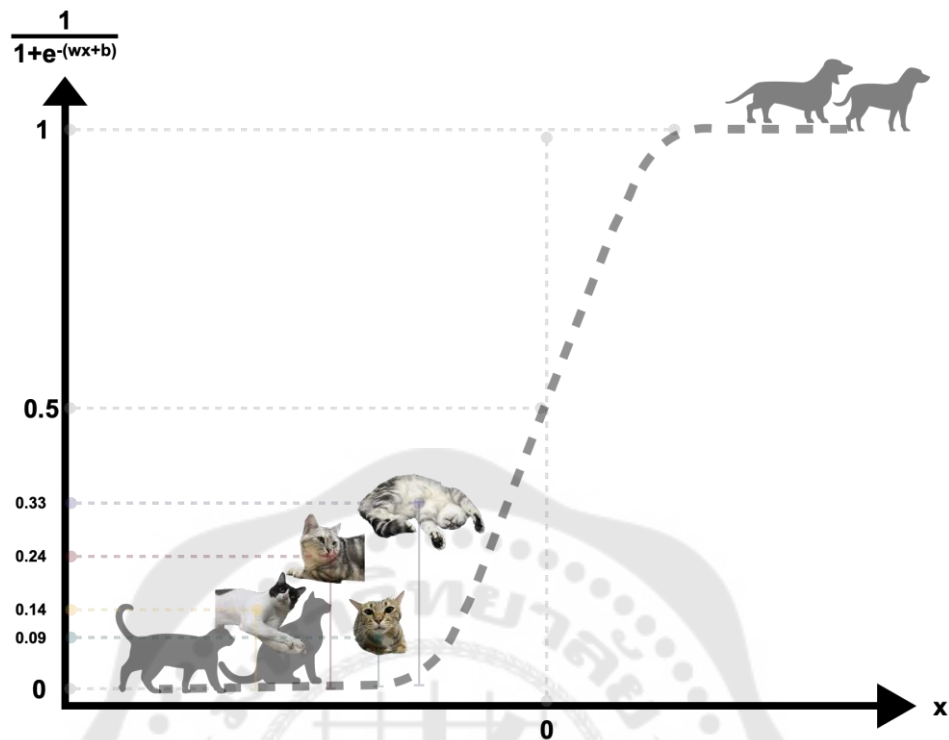
$$loss = -\frac{1}{n} * \left( \sum_{i=1}^n y_i (\log p(\hat{y}_i)) + (1 - y_i) (\log 1 - p(\hat{y}_i)) \right) \quad (3)$$

เมื่อ  $n$  คือจำนวนของตัวอย่างทั้งหมด

$y_i$  คือความน่าจะเป็นของผลลัพธ์จริง ซึ่งในตัวอย่างนี้เป็นได้ 2 ค่า คือ 1 (เป็นกลุ่มที่ 1) กับ 0 (เป็นกลุ่มที่ 2)

$p(\hat{y}_i)$  คือค่าความน่าจะเป็นที่ตัวอย่างนั้นจะเป็นกลุ่มที่ 1 ที่ได้จากการทำนาย ดังนั้นความน่าจะเป็นที่ตัวอย่างนั้นจะเป็นกลุ่มที่ 2 ที่ได้จากการทำนายจึงเท่ากับ  $1 - p(\hat{y}_i)$

ตัวอย่างเมื่อเราใช้แบบจำลองทำนายโดยป้อนข้อมูลนำเข้าของตัวอย่างเข้าไป และได้ผลลัพธ์ออกมามีภาพประกอบ 14



ภาพประกอบ 15 แสดงการทำนายด้วยฟังก์ชัน sigmoid

จากกราฟเป็นความสัมพันธ์ระหว่างค่า  $z$  (แนวแกน  $x$ ) และค่าที่ได้จากฟังก์ชัน sigmoid (แนวแกน  $y$ ) เมื่อให้ผลลัพธ์จริงของสุนัขมีค่าเป็น 1 ผลลัพธ์จริงของแมวจะมีค่าเป็น 0 หากนำไปแทนค่าในฟังก์ชันล๊อคกลอส จะสามารถแจกแจงได้ดังนี้

$$\begin{aligned}
 loss = & -\frac{1}{4} * ( \\
 & (0)(\log p(\hat{y}_i)) \\
 & + (1)(\log(1 - 0.09)) \\
 & + (1) \log(1 - 0.14) \\
 & + (1)(\log(1 - 0.24)) \\
 & + (1)(\log(1 - 0.33)) \\
 & )
 \end{aligned} \tag{4}$$

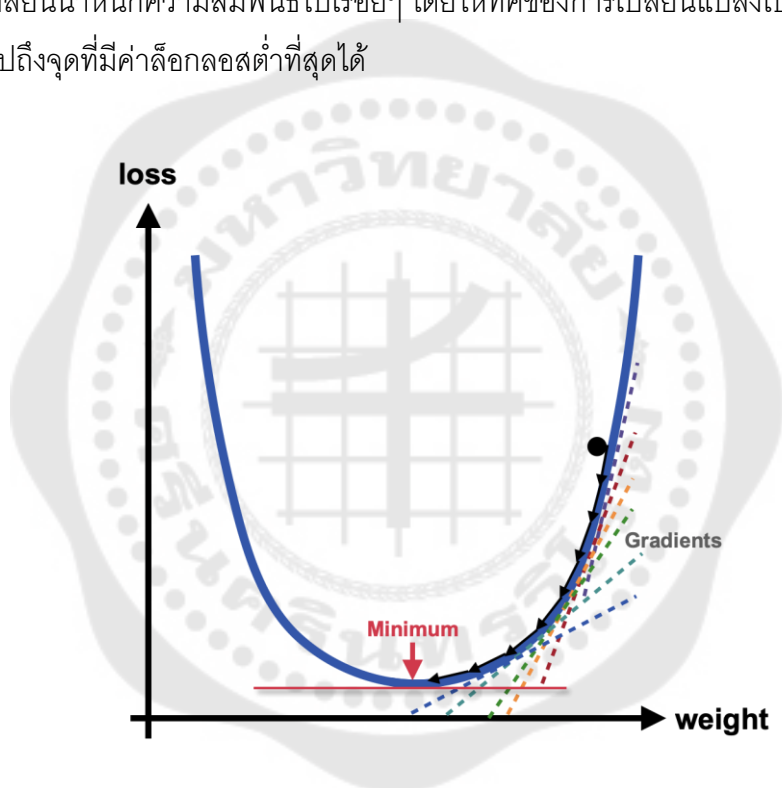
เนื่องจากเราต้องการให้แบบจำลองสามารถทำนายผลได้ใกล้เคียงกับผลลัพธ์จริงมากที่สุด เราจึงต้องการค่าที่ได้จากฟังก์ชันล๊อคกลอสน้อยที่สุดเท่าที่จะเป็นไปได้ การฝึกสอนแบบจำลองจึงต้องมีการปรับน้ำหนักของความสัมพันธ์และค่าอคติไปเรื่อยๆ จนกว่าจะได้ค่าล๊อคกลอสที่ต่ำที่สุด

### 3. การหาค่าที่เหมาะสมและค่าอคติที่ดีที่สุด (Optimization)

การปรับน้ำหนักของความสัมพันธ์ ( $w$ ) และค่าอคติ ( $b$ ) เพื่อหาค่าที่ดีที่สุดนั้น สามารถเลือกใช้สมการได้หลายแบบ ซึ่งมีการดัดแปลงไปจากสมการ Gradient Descent ในที่นี้จะขอพูดถึง 4 สมการ ได้แก่ Gradient Descent, Momentum, RMSProp และ Adam

#### 3.1 Gradient Descent

เมื่อเราวาดเส้นแสดงความสัมพันธ์ระหว่างค่าล๊อคกอสและน้ำหนัก ความสัมพันธ์ดังภาพประกอบ 16 จะเห็นได้ว่าหากเราแทนน้ำหนักตั้งต้นด้วยรูปวงกลม เมื่อทดลองเปลี่ยนน้ำหนักความสัมพันธ์ไปเรื่อยๆ โดยให้ทิศของการเปลี่ยนแปลงเป็นไปตามลูกศร จะสามารถไปถึงจุดที่มีค่าล๊อคกอสต่ำที่สุดได้



ภาพประกอบ 16 Gradient Descent

จะเห็นได้ว่าเส้นความชันของกราฟจะเป็นอนุพันธ์ของฟังก์ชันสูญเสียเทียบกับน้ำหนัก ( $\frac{d(loss)}{d(w)}$ ) เมื่อกำหนดให้ฟังก์ชันสูญเสียเป็นฟังก์ชันล๊อคกอส จะเขียนอนุพันธ์ของกราฟได้ดังนี้

$$\frac{d(loss)}{d(w)} = x(p(\hat{y}_i) - y_i) \quad (5)$$

ส่วนอนุพันธ์ของฟังก์ชันสูญเสียเทียบกับค่าอคตินั้นวิธีคิดคล้ายกับสมการด้านบน สามารถเขียนอนุพันธ์ของกราฟได้ดังนี้

$$\frac{d(loss)}{d(b)} = p(\hat{y}_i) - y_i \quad (6)$$

น้ำหนักและค่าอคติที่ปรับใหม่ จะหาได้จากสมการดังนี้

$$w_t = w_{t-1} - \alpha * \frac{d(loss)}{d(w)} \quad (7)$$

$$b_t = b_{t-1} - \alpha * \frac{d(loss)}{d(b)} \quad (8)$$

โดยกำหนดให้

$w_t$  และ  $b_t$  คือน้ำหนักและค่าอคติที่ปรับใหม่

$w_{t-1}$  และ  $b_{t-1}$  คือน้ำหนักและค่าอคติเดิม

$\alpha$  คืออัตราการเรียนรู้ (learning rate) ค่าที่นิยมใช้คือ 0.001 ถึง 0.01 [22]

$x$  คือข้อมูลนำเข้า ซึ่งก็คือค่าของพีเจอร์ต่างๆ

เมื่อได้ค่าน้ำหนักและความสัมพันธ์ใหม่ ก็จะสามารถนำมาคำนวณหา  $p(\hat{y}_i)$  ณ ตำแหน่งใหม่ได้ ซึ่งการปรับค่า  $w$  และ  $b$  นี้จะเกิดขึ้นซ้ำๆ จนกว่าค่าล๊อคอสจะมีการเปลี่ยนแปลงไปน้อยมากหรือไม่เปลี่ยนแปลง (จุด minimum) ข้อเสียของวิธีนี้คือการปรับค่าน้ำหนักและอคติใช้เวลานาน เนื่องจากการใช้สมการ gradient descent มักจะใช้อัตราการเรียนรู้ที่ต่ำเพื่อป้องกันปัญหาการแยกออก (divergence) ซึ่งหากดูจากรูปกราฟด้านบนคือการเคลื่อนที่ลงมาใกล้กับจุด minimum จึงมีผู้คิดค้นสมการเพื่อลดเวลาที่ใช้ในการไปสู่จุดต่ำสุด

### 3.2 Momentum

โมเมนตัม (Momentum) หรือ “Gradient Descent with Momentum” เป็นการหาน้ำหนักและค่าอคติที่ประยุกต์จากสมการ gradient descent โดยใช้หลักการ “Exponentially Weighted Moving Average” เพื่อลดสัญญาณรบกวน (noise) เมื่อมีการปรับน้ำหนักและค่าอคติ [23] ซึ่งจะทำให้สามารถใช้อัตราการเรียนรู้ที่สูงขึ้นได้ และทำให้เคลื่อนที่ไปสู่จุด minimum ได้เร็วขึ้น หลักการ Exponential Moving Average นั้นเป็นการหาค่าเฉลี่ยของจุด



ข้อมูลในช่วงเวลาหนึ่ง ซึ่งจะให้น้ำหนักความสำคัญของข้อมูล ณ เวลาใกล้เคียงกับปัจจุบัน มากกว่าข้อมูลในอดีตที่ผ่านมาแล้ว สามารถเขียนในรูปสมการได้ดังนี้

$$V_t = (\beta)V_{t-1} + (1 - \beta)\theta_t \quad (9)$$

โดยกำหนดให้

$V_t$  คือค่าเฉลี่ย ณ เวลาปัจจุบัน

$\beta$  คือน้ำหนักความสำคัญของข้อมูลในอดีต อยู่ในช่วง 0 ถึง 1

$V_{t-1}$  คือค่าเฉลี่ย ณ จุดข้อมูลก่อนหน้า

$\theta_t$  คือจุดข้อมูลปัจจุบัน

ในสมการ gradient descent ปกติ ทิศของการปรับน้ำหนัก (การเคลื่อนที่ลงสู่จุด minimum) จะหาได้จาก  $\frac{d(loss)}{d(w)}$  แต่ใน gradient descent with momentum จะใช้ทิศของการปรับน้ำหนักเป็นค่าเฉลี่ยของ  $\frac{d(loss)}{d(w)}$  แทน ซึ่งสามารถเขียนได้ดังนี้

$$V_{\frac{d(loss)}{d(w)}}_t = (\beta)V_{\frac{d(loss)}{d(w)}}_{t-1} + (1 - \beta)\frac{d(loss)}{d(w)}_t \quad (10)$$

โดยกำหนดให้

$V_{\frac{d(loss)}{d(w)}}_t$  คือค่าเฉลี่ย ของอนุพันธ์ของฟังก์ชันสูญเสียเทียบกับน้ำหนัก ณ

เวลาปัจจุบัน

$V_{\frac{d(loss)}{d(w)}}_{t-1}$  คือค่าเฉลี่ย ของอนุพันธ์ของฟังก์ชันสูญเสียเทียบกับน้ำหนัก ณ

จุดข้อมูลก่อนหน้า

เมื่อนำไปแทนในสมการ gradient descent with momentum จะได้

$$w_t = w_{t-1} - \alpha * V_{\frac{d(loss)}{d(w)}}_t \quad (11)$$

ดังนี้

การปรับค่าอคตินั้นใช้หลักการเดียวกับการปรับค่าน้ำหนัก เขียนดังสมการได้

$$b_t = b_{t-1} - \alpha * \frac{V_{d(loss)}}{d(b)_t} \quad (12)$$

### 3.3 Root Mean Squared Propagation (RMSProp)

จากการปรับน้ำหนักและค่าอคติในวิธีที่กล่าวมาในข้อก่อนหน้า จะเห็นว่า อัตราการเรียนรู้เป็นค่าที่ใช้ร่วมกันในการปรับน้ำหนักของทุกพีเจอร์และค่าอคติ RMSProp จึงเป็นหนึ่งในสมการกลุ่ม “Adaptive Optimizers” ซึ่งคิดค้นขึ้นมาเพื่อปรับอัตราการเรียนรู้ให้เหมาะกับค่าน้ำหนักแต่ละตัว เขียนเป็นสมการได้ดังนี้

$$\begin{aligned} S_{\frac{d(loss)}{d(w)}_t} &= (\beta_2) S_{\frac{d(loss)}{d(w)}_{t-1}} + (1 - \beta_2) \left( \frac{d(loss)}{d(w)}_t \right)^2 \\ S_{\frac{d(loss)}{d(b)}_t} &= (\beta_2) S_{\frac{d(loss)}{d(b)}_{t-1}} + (1 - \beta_2) \left( \frac{d(loss)}{d(b)}_t \right)^2 \\ w_t &= w_{t-1} - \frac{\alpha}{\sqrt{S_{\frac{d(loss)}{d(w)}_t} + \epsilon}} * \frac{d(loss)}{d(w)} \\ b_t &= b_{t-1} - \frac{\alpha}{\sqrt{S_{\frac{d(loss)}{d(b)}_t} + \epsilon}} * \frac{d(loss)}{d(b)} \end{aligned} \quad (13)$$

โดยกำหนดให้

$S_{\frac{d(loss)}{d(w)}_t}$  คือค่าเฉลี่ย ของอนุพันธ์ของฟังก์ชันสูญเสียเทียบกับน้ำหนัก ณ

เวลาปัจจุบัน ยกกำลังสอง

$S_{\frac{d(loss)}{d(b)}_t}$  คือค่าเฉลี่ย ของอนุพันธ์ของฟังก์ชันสูญเสียเทียบกับค่าอคติ ณ

เวลาปัจจุบัน ยกกำลังสอง

$\beta_2$  คือน้ำหนักความสำคัญของค่า S ในอดีต

$\epsilon$  คือค่าคงที่เพื่อป้องกันไม่ให้อส่วนของ  $\alpha$  มีค่าเป็น 0 กำหนดเป็นค่า  $10^{-8}$

จะเห็นได้ว่า เมื่ออนุพันธ์ของฟังก์ชันสูญเสียมีค่ามาก จะทำหน้าที่ลดอัตราการเรียนรู้ (ตัวหรมีค่ามากขึ้น) ในทางกลับกันหากอนุพันธ์ของฟังก์ชันสูญเสียมีค่าน้อย จะไปเพิ่มอัตราการเรียนรู้ให้มากขึ้น [24]

### 3.4 Adam

Adam Optimizer เป็นการนำข้อดีของ Momentum และ RMSProp มาผนวกเข้าด้วยกัน [25] ดังในสมการที่ 14

$$w_t = w_{t-1} - \alpha * V_{\frac{d(\text{loss})}{d(w)}}_t$$

$$w_t = w_{t-1} - \frac{\alpha}{\sqrt{S_{\frac{d(\text{loss})}{d(w)}}_t + \epsilon}} * \frac{d(\text{loss})}{d(w)}$$

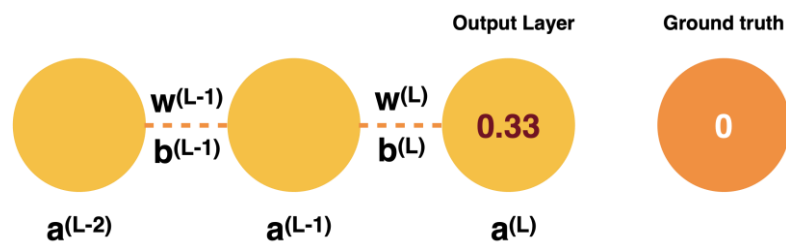
$$w_t = w_{t-1} - \frac{\alpha}{\sqrt{S_{\frac{d(\text{loss})}{d(w)}}_t + \epsilon}} * V_{\frac{d(\text{loss})}{d(w)}}_t$$

(14)

ในงานการเรียนรู้เชิงลึกนิยมใช้ Adam เป็น Optimizer พื้นฐาน เนื่องจากใช้งานได้ดี อย่างไรก็ตามในปัจจุบันมี optimizing algorithm ที่พัฒนาต่อยอดจาก Adam ไปจำนวนหนึ่ง ตัวอย่างเช่น DiffGrad และ Ranger [26]

### 4. Backpropagation

จากหัวข้อที่กล่าวมาด้านบนนั้น เปรียบเสมือนการขยายให้เห็นการทำงานของโหนด 1 ตัว ในความเป็นจริงแล้วโครงข่ายประสาทเทียมนั้นประกอบด้วยโหนดหลายตัวที่เชื่อมต่อกันหลายชั้น ซึ่งสามารถวาดภาพสมมติได้ดังนี้



ภาพประกอบ 17 แสดงความสัมพันธ์ระหว่างโหนดในชั้นต่างๆ ของโครงข่าย

จะเห็นได้ว่าการเชื่อมต่อกันของโหนดในชั้น  $a^{(L-2)}$ ,  $a^{(L-1)}$  และ  $a^{(L)}$  นั้นก็คือผลลัพธ์จากชั้นก่อนหน้า จะกลายเป็นข้อมูลนำเข้า (input) สำหรับชั้นถัดไป เขียนเป็นสมการได้ดังนี้

$$\begin{aligned} a^{(L)} &= g(z^{(L)}) \\ z^{(L)} &= w^{(L)} * a^{(L-1)} + b^{(L)} \\ &= w^{(L)} * g(z^{(L-1)}) + b^{(L)} \\ &= w^{(L)} * g(w^{(L-1)} * a^{(L-2)} + b^{(L-1)}) + b^{(L)} \end{aligned} \quad (15)$$

โดยกำหนดให้  $g(z^{(L)})$  คือฟังก์ชันกระตุ้น (activation function) ของค่า  $z^{(L)}$

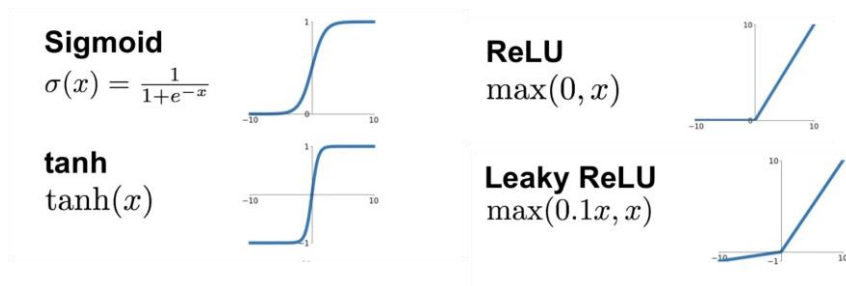
จากสมการด้านบน เราสามารถหาค่าอนุพันธ์และค่าอคติในแต่ละชั้นได้ ด้วยการใช้อนุพัทธ์ (chain rule) ในการหาอนุพันธ์ของฟังก์ชันสูญเสียเทียบกับน้ำหนักในชั้นนั้นๆ เช่นหากเราอยากหาค่า  $\frac{d(loss)}{d(w^{L-1})}$  จะเขียนสมการจากอนุพัทธ์ได้ว่า

$$\frac{d(loss)}{d(w^{L-1})} = \frac{d(loss)}{d(a^{(L)})} * \frac{d(a^{(L)})}{d(z^{(L)})} * \frac{d(z^{(L)})}{d(a^{(L-1)})} * \frac{d(a^{(L-1)})}{d(z^{(L-1)})} * \frac{d(z^{(L-1)})}{d(w^{(L-1)})} \quad (16)$$

การหาค่าน้ำหนักของชั้นก่อนหน้าจากการคูณกันของอนุพันธ์เช่นนี้เรียกว่าวิธี backpropagation โดยการหาค่าน้ำหนักและค่าอคติในชั้นต่างๆ ก็จะใช้หลักการหาค่าน้ำหนักและค่าอคติที่ดีที่สุด (Optimization) ดังที่ได้กล่าวไว้ก่อนหน้านี้

### 5. ฟังก์ชันกระตุ้น (Activation function)

ฟังก์ชันกระตุ้น คือฟังก์ชันที่จะถูกกระตุ้นเมื่อค่าข้อมูลนำเข้าที่รับเข้ามานั้นสูงกว่าค่าขั้นต่ำ (threshold) ที่กำหนดไว้ ฟังก์ชันกระตุ้นจะทำให้แบบจำลองสามารถทำนายความสัมพันธ์ที่ไม่ใช่เส้นตรง (non-linearity) ระหว่างฟีเจอร์ต่างๆ และผลลัพธ์ได้ ตัวอย่างฟังก์ชันกระตุ้นเช่น sigmoid, tanh, ReLU และ Leaky ReLU



ภาพประกอบ 18 กราฟแสดงฟังก์ชันกระตุ้น

ที่มา : A. Nikishae, "How to debug neural networks. Manual.", Medium. [ออนไลน์]. Available at: <https://medium.com/machine-learning-world/dc2a200f10f2>

## 2.2 งานวิจัยที่เกี่ยวข้องกับอาการแสดงเมื่อแมวมีความเจ็บปวด

เนื่องจากแมวไม่สามารถสื่อสารกับมนุษย์โดยตรงเพื่อบอกความเจ็บปวดของตนได้ สัตวแพทย์จึงสร้างมาตรวัดความเจ็บปวด (pain scale) ของแมวขึ้นมาจากการสังเกตลักษณะท่าทางที่แมวแสดงออกมามากกว่าปกติเมื่อมีความเจ็บปวด โดยมีงานวิจัยที่เกี่ยวข้องดังนี้

### 2.2.1 Validation of the English version of the UNESP-Botucatu multidimensional composite pain scale for assessing postoperative pain in cats [4]

เป็นงานวิจัยที่วัดความแม่นยำในการประเมินความเจ็บปวดด้วย UNESP-Botucatu multidimensional composite pain scale (MCPS) ฉบับแปลภาษาอังกฤษซึ่งมีต้นฉบับเป็นภาษาโปรตุเกส ซึ่ง MCPS นี้เป็นมาตรวัดที่ใช้ประเมินความเจ็บปวดของแมวเพศเมียหลังการผ่าตัดทำหมัน เพื่อหาเกณฑ์บ่งชี้ว่าจะพิจารณาให้ยาระงับปวดเพิ่มอีกเมื่อใด โดยพิจารณาจากท่าทางการนอน ความสุขสบาย การขยับตัว การตอบสนองต่อสิ่งเร้า พฤติกรรมอื่นๆ การตอบสนองเมื่อมีการคลำที่แผลผ่าตัดและบริเวณช่องท้อง ความดัน ความอยากอาหาร และเสียง โดย MCPS มีคะแนนรวมอยู่ในช่วง 0-30 ซึ่งพบว่าเกณฑ์ที่ถือว่ามีอาการเจ็บปวดคือคะแนน > 7 จากการทดลองพบว่าค่า intra-class correlation coefficients (ICC) ของคะแนนระหว่างผู้ประเมินทั้งหมดเท่ากับ 0.98 ซึ่งแปลผลได้ว่ามีความสอดคล้องกันดีมากระหว่างผู้ประเมิน อย่างไรก็ตามการประเมิน MCPS นั้นใช้เวลานานเนื่องจากมีเกณฑ์ที่ใช้พิจารณาหลายข้อ

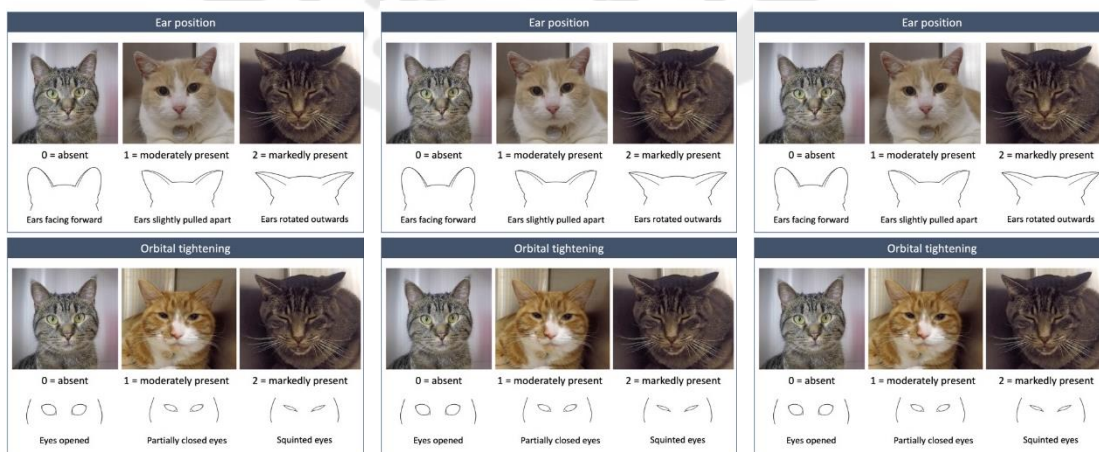
### 2.2.2 Preliminary appraisal of the reliability and validity of the Colorado State University Feline Acute Pain Scale [28]

ได้ศึกษาประสิทธิภาพของมาตรวัดความเจ็บปวด Colorado State University Feline Acute Pain Scale หรือ CSU-FAPS โดยกลุ่มตัวอย่างเป็นแมวเพศเมียที่เข้ารับการผ่าตัดทำหมัน

มาตรวัดนี้พิจารณาจากลักษณะพฤติกรรม การตอบสนองต่อการสัมผัส และความเกร็งของกล้ามเนื้อ จากการทดลองพบว่าค่า intra-class correlation coefficients (ICC) ของคะแนนรวมระหว่างกลุ่มวิสัญญีแพทย์เท่ากับ 0.61 และระหว่างกลุ่มแพทย์ประจำบ้านเท่ากับ 0.67 ซึ่งแปลผลได้ว่ามีความสอดคล้องกันดีระหว่างผู้ประเมิน เป็นมาตรวัดที่ประเมินหลายมิติและใช้เวลาน้อย เหมาะกับบุคลากรที่มีประสบการณ์

### 2.2.3 Facial expressions of pain in cats: the development and validation of a Feline Grimace Scale [5]

เป็นงานวิจัยที่จัดทำขึ้นเพื่อสร้างมาตรวัดประเมินความเจ็บปวดระยะเฉียบพลันในแมวที่ เกิดขึ้นจากสาเหตุอื่นนอกเหนือจากการผ่าตัด เช่น ตับอ่อนอักเสบ นิ่วในไต ฯลฯ โดยประเมินจาก หน้าแมวเป็นหลัก ภาพหน้าแมวนี้ได้จากการบันทึกคลิปวิดีโอแมวที่เข้ารับการตรวจที่ Centre hospitalier universitaire vétérinaire (CHUV) โดยบันทึกทั้งก่อนและได้รับยาแก้ปวด พบว่ามีการ เคลื่อนไหวของกล้ามเนื้อบางกลุ่มที่มีการเปลี่ยนแปลงไปอย่างมีนัยยะสำคัญเมื่ออาการเจ็บปวด ลดลง สรุปได้เป็น action unit 5 ตำแหน่งได้แก่ ตำแหน่งใบหู (ear position) การหรี่ตา (orbital tightening) ความริษของปากบน (muzzle tension) การวางตัวของหนวด (whiskers position) และตำแหน่งศีรษะ (head position) โดยแต่ละตำแหน่งจะมีคะแนน 0 (ตำแหน่งปกติ) 1 (ตำแหน่ง ผิดปกติเล็กน้อย หรือไม่สามารแยกได้ชัดเจน) 2 (ผิดปกติดังชัดเจน) ซึ่งมาตรวัดนี้สรุปไว้ว่า คะแนนที่บ่งชี้ว่าแมวอาจมีความเจ็บปวดคือ 4 คะแนนขึ้นไป



ภาพประกอบ 19 การประเมินความเจ็บปวดด้วย Feline Grimace Scale

ที่มา : M. C. Evangelista และคณะ , “Facial expressions of pain in cats: the development and validation of a Feline Grimace Scale”, *Sci Rep*, ปี 9, ฉบับที่ 1, น. 19128, ธ.ค. 2019, doi: 10.1038/s41598-019-55693-8.

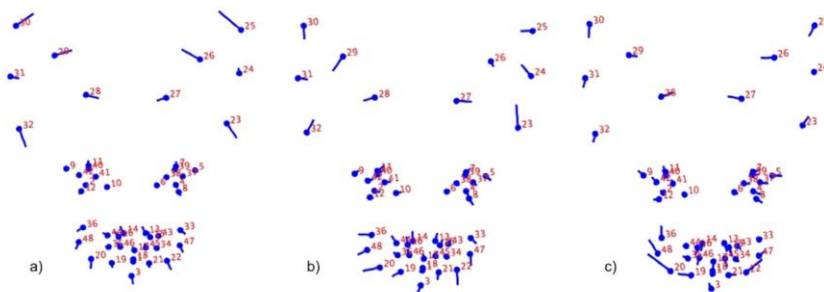
แม้ว่าการประเมิน Feline Grimace Scale นั้นทำได้ไม่ยาก แต่เนื่องจากเป็นการประเมินโดยสายตามนุษย์ จึงอาจจะมีอคติได้

#### 2.2.4 Agreement and reliability of the Feline Grimace Scale among cat owners, veterinarians, veterinary students and nurses [29]

มีผู้ทำงานวิจัยเพื่อศึกษาความน่าเชื่อถือและความสอดคล้องของการประเมินความเจ็บปวดของแมวด้วย Feline Grimace Scale โดยแบ่งผู้ประเมินเป็น 4 กลุ่ม ได้แก่ เจ้าของแมว สัตวแพทย์ นักศึกษาสัตวแพทย์ และพยาบาล โดยใช้ intra-class correlation coefficients (ICC) เป็นมาตรวัด พบว่าความสอดคล้องของคะแนน FGS โดยรวมนั้นอยู่ในเกณฑ์ดี (ICC > 0.8) แต่จะมี action unit บางตำแหน่งที่พบความไม่ไปด้วยกันในการประเมิน ได้แก่ ความรีของปากบน (muzzle tension) และการวางตัวของหนวด (whisker position) ซึ่งได้คะแนน ICC ในกลุ่มเจ้าของแมวเพียง 0.39 และ 0.47 ตามลำดับ เนื่องจากอาจจะมีประสบการณ์ในการประเมินน้อยกว่า

#### 2.2.5 Geometric morphometrics for the study of facial expressions in non-human animals, using the domestic cat as an exemplar [6]

เป็นงานวิจัยที่หาความเปลี่ยนแปลงของใบหน้าจากจุดอ้างอิงบนหน้าแมวที่เปลี่ยนแปลงไปเมื่อมีอาการเจ็บปวด กลุ่มตัวอย่างประกอบด้วยแมวพันธุ์ทาง (Domestic short hair) เพศเมีย สุขภาพแข็งแรงจำนวน 29 ตัวที่เข้ารับการผ่าตัดมดลูกและรังไข่เพื่อทำหมัน ผู้วิจัยได้บันทึกวีดีโอแมวก่อนผ่าตัด, 1 ชั่วโมงหลังจากการผ่าตัด (ก่อนที่จะให้ยาระงับปวดหลังผ่าตัด), 4 ชั่วโมงหลังผ่าตัด (ให้ยาระงับปวดหลังผ่าตัดแล้ว), และ 24 ชั่วโมงหลังการผ่าตัด และให้สัตวแพทย์ประเมินคะแนนความเจ็บปวดของแมวในทุกช่วงเวลาด้วย MCPS พบว่าในทุกช่วงเวลานั้นแมวมีระดับความเจ็บปวดต่างกันอย่างมีนัยยะสำคัญทางสถิติ หลังจากนั้นผู้วิจัยจึงคัดเลือกภาพหนึ่งจากวีดีโอในทุกช่วงเวลาได้ 932 ภาพ นำภาพมาสร้างจุดอ้างอิงบนใบหน้า (facial landmarks) 48 จุดอิงจาก catFACS แล้วดูว่าจุดเหล่านั้นมีการเปลี่ยนแปลงไปมากน้อยเพียงใดเมื่อแมวมีความเจ็บปวด โดยใช้ principal component analysis (PCA) พบว่าใน PC3 มี คะแนน principal component ต่างกันในทุกช่วงเวลาอย่างมีนัยยะสำคัญทางสถิติ และเมื่อนำมาพล็อตให้เห็นภาพ (visualize) จะเห็นได้ว่าเมื่อแมวมีความเจ็บปวดรูปร่างของใบหูและปากจะมีการเปลี่ยนไปมากที่สุด



**Figure 3.** Geometric representation of the face based on average landmarks from 932 images extracted from 29 cats, created using lollipop graphs. Relative facial shape changes associated with higher PC values for PC3 (a), PC6 (b) and PC7 (c) are represented by the lines protruding from each landmark, highlighting the direction and magnitude of movement. Higher PC scores are therefore reflected by a greater distance along the line from the circular nodes, with lower PC scores reflecting less distance from the nodes. For ease of reference between the PC shape change descriptions (excluding i-vii below), and corresponding images, images have been 'flipped' horizontally so that the left side of the cat's face is located on the left side of the image. Images produced using MorphoJ, Version 1.06d<sup>47</sup>.

ภาพประกอบ 20 principal components จากจุดอ้างอิงบนหน้าแมว

ที่มา: L. R. Finka และคณะ, "Geometric morphometrics for the study of facial expressions in non-human animals, using the domestic cat as an exemplar", *Scientific Reports*, ปี 9, ฉบับที่ 1, น. 9883, ก.ค. 2019, doi: 10.1038/s41598-019-46330-5.

### 2.2.6 Automated recognition of pain in cats [7]

ได้ทำการศึกษาเปรียบเทียบประสิทธิภาพของการประเมินความเจ็บปวดของแมวระหว่างการให้โครงข่ายประสาทเทียมแบบคอนโวลูชัน (convolutional neural networks) และการใช้โครงข่ายประสาทเทียมแบบ multilayer perceptron โดยใช้การวิเคราะห์เวกเตอร์ที่สร้างจากจุดอ้างอิงบนใบหน้าของแมว โดยชุดข้อมูลที่ใช้มาจากงานวิจัยในข้อ 2.2.5 [6] ข้อมูลคำตอบ (label) แบ่งเป็น pain และ no pain พบว่าความแม่นยำของทั้ง 2 วิธีนั้นมากกว่า 72% ซึ่งบ่งชี้ว่าการใช้ระบบอัตโนมัติในการประเมินความเจ็บปวดจากภาพแมวนั้นมีประสิทธิภาพที่ดี อย่างไรก็ตาม กลุ่มตัวอย่างที่ทำมาทดลองนั้นยังขาดความหลากหลายเนื่องจากเป็นแมวพันธุ์เดียวกัน เพศเดียวกัน และความเจ็บปวดที่เกิดจากสาเหตุเดียวกัน

### 2.2.7 Explainable automated pain recognition in cats [8]

ได้ทำการศึกษาเปรียบเทียบประสิทธิภาพของการประเมินความเจ็บปวดของแมวระหว่างแบบจำลอง random forest, multilayer perceptron และ โครงข่ายประสาทเทียมแบบคอนโวลูชัน โดยเพิ่มความหลากหลายของกลุ่มตัวอย่างให้มากขึ้นทั้งเพศและสายพันธุ์ เป็นแมวจำนวน 84 ตัว



เพศผู้และเพศเมียที่มีความเจ็บป่วยที่ต่างกัน แบ่งเป็นกลุ่ม pain 42 ภาพ และ no pain 42 ภาพ เพื่อนำไปฝึกสอนแบบจำลอง พบว่าความแม่นยำของแบบจำลอง random forest, multilayer perceptron และโครงข่ายประสาทเทียมแบบคอนโวลูชันมีค่าเท่ากับ 77%, 69% และ 63% ตามลำดับ นอกจากนี้ผู้วิจัยยังได้ทดลองบดบังฟิเจอร์ (occlusion) สำคัญต่างๆ ได้แก่ หู ตา ปาก แล้วนำไปฝึกสอนแบบจำลอง เพื่อเปรียบเทียบว่าเมื่อมีการบดบังฟิเจอร์นั้นๆ แล้ว ความแม่นยำในการทำนายจะลดลงหรือเพิ่มขึ้นเพียงใด เทียบกับการฝึกสอนกับรูปถ่ายที่สมบูรณ์ พบว่าแบบจำลองทั้งสองให้ผลสอดคล้องกันคือการบดบังปากทำให้ประสิทธิภาพในการทำนายจะลดลงมากที่สุด เมื่อบดบังหูประสิทธิภาพในการทำนายจะเพิ่มขึ้นมากที่สุด ซึ่งบ่งชี้ว่าฟิเจอร์ที่มีความสำคัญที่สุดคือบริเวณปาก และน้อยที่สุดคือบริเวณหู อย่างไรก็ตามการศึกษานี้ใช้จำนวนตัวอย่างในการฝึกสอนน้อย ประสิทธิภาพของแบบจำลองโครงข่ายประสาทเทียมจึงมีแนวโน้มที่จะได้ค่าความแม่นยำต่ำ

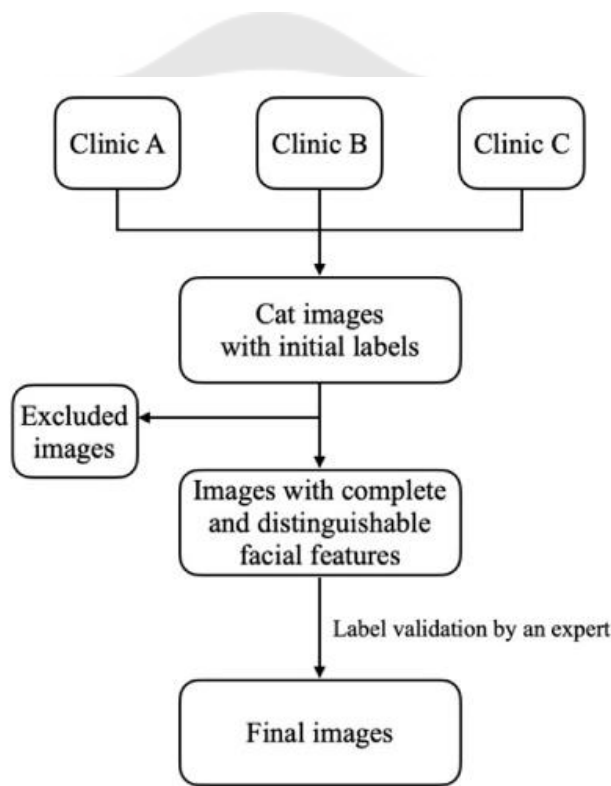


### บทที่ 3

## กระบวนการ และวิธีการดำเนินการวิจัย

### 3.1 การรวบรวมข้อมูล (Data Acquisition)

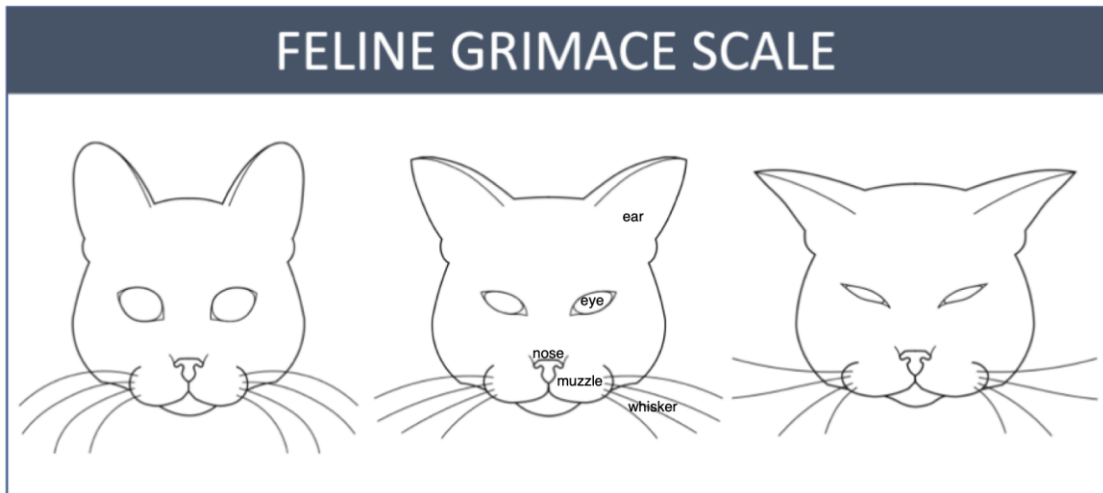
เนื่องจากข้อมูลภาพถ่ายของใบหน้าแมวที่มีผลลัพธ์ระบุไว้ว่าเป็นหน้าแมวที่เจ็บ (pain) หรือไม่เจ็บ (no pain) นั้น ไม่มีบันทึกไว้ในชุดข้อมูลสาธารณะ จึงต้องอาศัยการรวบรวมข้อมูลเอง ดังแผนภาพ



ภาพประกอบ 21 การรวบรวมข้อมูล

แหล่งข้อมูลภาพถ่ายหน้าแมวมาจากคลินิกสัตวแพทย์ โรงพยาบาลสัตว์ และหน่วยทำหมันแมวจรเคลื่อนที่ ซึ่งจะถูกระบุข้อมูลเฉลี่ย (label) ไว้โดยสัตวแพทย์ประจำแหล่งข้อมูลโดยว่าเป็นกลุ่มหน้าที่มีหรือไม่มีอาการเจ็บปวด

ลักษณะภาพถ่ายแมวที่นำมาใช้เป็นภาพถ่ายหน้าตรงที่มีฟีเจอรบใบหน้าครบถ้วน ได้แก่ ใบหูทั้ง 2 ข้าง ดวงตาทั้ง 2 ข้าง จมูก ปากบน (muzzle) หนวด (whisker) ดังตัวอย่างในภาพประกอบ 22



ภาพประกอบ 22 ภาพแสดงพีเจอรบนใบหน้าแมวที่ต้องการนำมาฝึกสอน

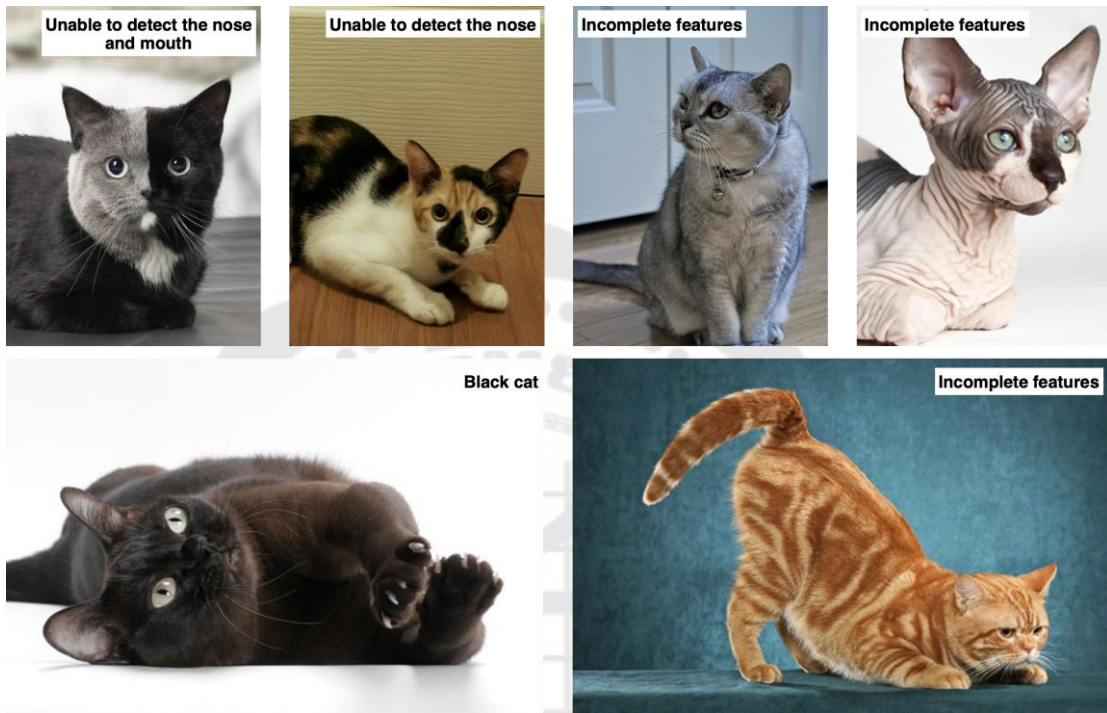


ภาพประกอบ 23 ตัวอย่างภาพถ่ายที่สามารถนำมาฝึกสอนแบบจำลอง

เกณฑ์การคัดออก (exclusion criteria) ได้แก่

- 1) ภาพถ่ายที่เห็นพีเจอรไม่ครบ

2) แมวที่มีสีดำบริเวณพีเจอร์ที่สำคัญจนการมองเห็นขอบเขตของอวัยวะเป็นไปได้ยาก เช่น แมวดำ แมวสามสี แมวไคเมร่า (chimera cat) ซึ่งอาจทำให้ไม่สามารถสกัดขอบเขตของพีเจอร์ที่สนใจได้

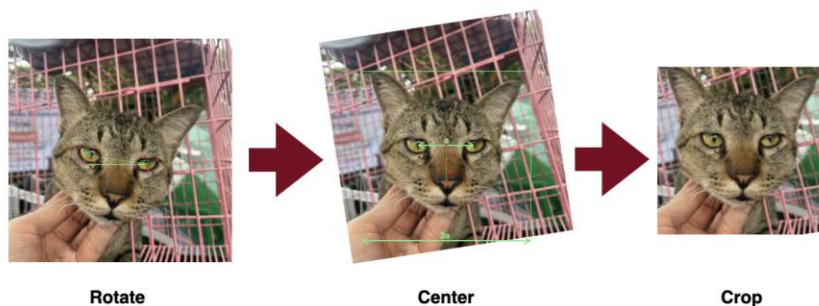


ภาพประกอบ 24 ตัวอย่างเกณฑ์การคัดออก

หลังจากที่ตัดภาพที่อยู่ในเกณฑ์การคัดออกออกไปแล้ว ภาพที่ได้จะถูกนำมาตรวจสอบข้อมูลเฉลี่ยโดยสัตวแพทย์ผู้เชี่ยวชาญอีกครั้ง ก่อนนำไปสร้างแบบจำลอง

### 3.2 การเตรียมข้อมูล (Data Preprocessing)

เมื่อได้ภาพที่มีข้อมูลผลลัพธ์ระบุไว้แล้ว จะนำมาปรับเปลี่ยนการจัดวางตัวของใบหน้า (face alignment) โดยการหมุน กำหนดจุดกึ่งกลาง ตัดบางส่วนของภาพออกไป (crop) ดังรูป



ภาพประกอบ 25 การหมุน จัดกึ่งกลาง และตัดบางส่วนของภาพ

หลังจากนั้นจะทำการปรับขนาดให้เหมาะสมกับแบบจำลองที่จะเลือกใช้ นำภาพที่ถูกปรับขนาดแล้วมาแบ่งเป็นกลุ่มฝึกสอน (training set) 80%, กลุ่มใช้ปรับจูนพารามิเตอร์ (validation set) 10%, และกลุ่มทดสอบ (test set) 10% แล้วใช้เทคนิคการเพิ่มข้อมูลกับกลุ่มฝึกสอน ตัวอย่างเช่นการสร้างภาพที่มีความเข้มแสงแตกต่างกัน ความมืดสว่าง การบิดมุมต่างๆ การกลับซ้ายขวา เป็นต้น



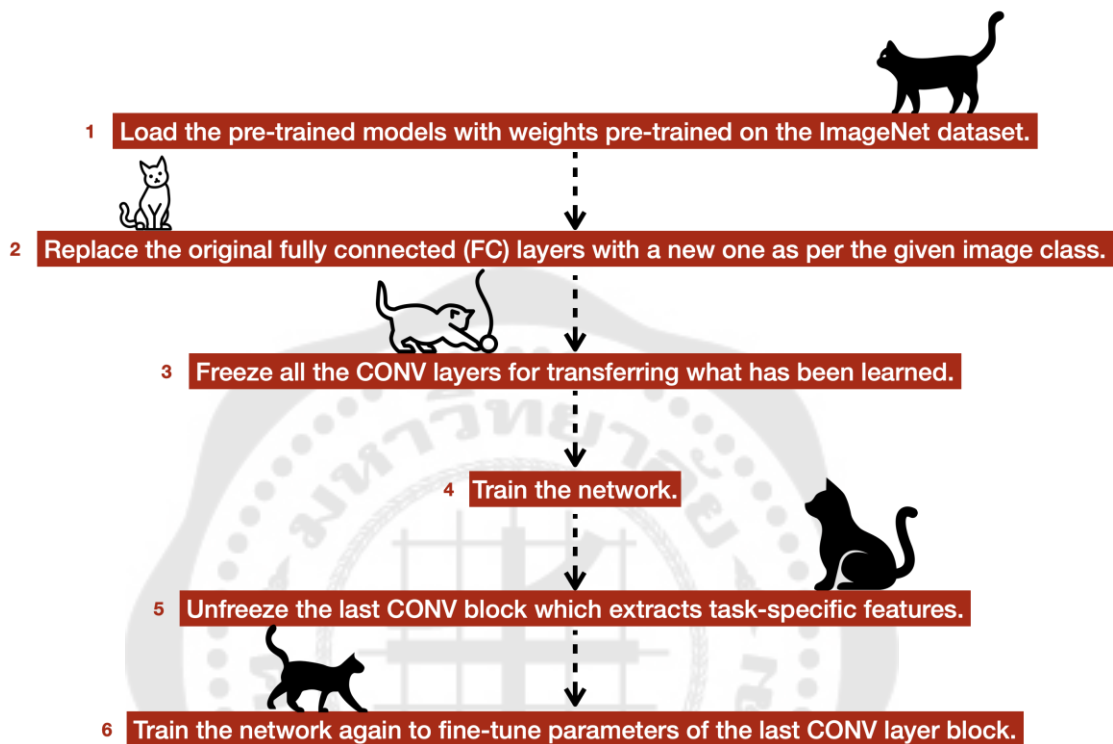
ภาพประกอบ 26 การเพิ่มข้อมูลฝึกสอน

### 3.3 การสร้างและปรับปรุงแบบจำลอง (Modeling)

เมื่อได้ภาพที่พร้อมนำไปฝึกสอนแบบจำลองแล้ว กลุ่มฝึกสอนจะถูกนำไปฝึกสอนแบบจำลอง โดยสร้างแบบจำลองด้วยวิธีการถ่ายทอดองค์ความรู้ คือการนำแบบจำลองที่ได้รับการฝึกสอนมาแล้ว 2 แบบ คือแบบจำลองสถาปัตยกรรม ResNet50 และ EfficientNetB7 มาดัดแปลงในชั้นเชื่อมโยงสมบรูณ์เพื่อให้ได้ผลลัพธ์ออกมาเป็นกลุ่ม 'pain' (แมวที่มีความเจ็บปวด) และ 'no pain' (แมวที่ไม่มีความเจ็บปวด)

ในขั้นตอนนี้จะใช้แบบจำลองและค่าน้ำหนักเริ่มต้นที่ได้รับการฝึกสอนมาแล้วจาก ImageNet แล้วทำการดัดแปลงชั้นเชื่อมโยงสมบรูณ์ คงค่าน้ำหนักและอคติในชั้นคอนโวลูชันไว้ (freeze) แล้วฝึกสอนแบบจำลองด้วยข้อมูลภาพถ่ายแมวที่มีข้อมูลคำตอบเป็น pain และ no pain

หลังจากนั้นจึงหยุดแช่แข็ง (unfreeze) ชั้นคอนโวลูชันชั้นสุดท้าย และฝึกสอนแบบจำลองอีกครั้ง เพื่อปรับค่าน้ำหนักและอคติในชั้นคอนโวลูชันชั้นสุดท้ายให้จำเพาะเจาะจงกับงานวิจัยต่อไป



ภาพประกอบ 27 การทดลองฝึกสอนแบบจำลองด้วยการถ่ายถอดองค์ความรู้

โดยแต่ละแบบจำลองที่ได้รับการฝึกสอนนั้นจะค่าพารามิเตอร์ที่แตกต่างกันไปได้แก่ ค่าอัตราการเรียนรู้ อัลกอริทึมที่ใช้หาค่าที่ดีที่สุด และการปรับขนาดจำนวนตัวอย่างที่แบบจำลองใช้ในการปรับค่าน้ำหนักของข้อมูลในการฝึกสอนแต่ละรอบ

### 3.4 การประเมินแบบจำลอง (Evaluation)

ในการประเมินประสิทธิภาพแบบจำลอง จะแบ่งออกเป็น การหาความแม่นยำ (accuracy) ความเที่ยงตรง (precision) ความไว (recall) และค่าเฉลี่ยแบบฮาร์โมนิกของความเที่ยงตรงและความไว (F1 Score) ดังสูตรคำนวณด้านล่าง

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\begin{aligned}
 \text{Recall} &= \frac{TP}{TP + FN} \\
 \text{F1 Score} &= \frac{1}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}} \\
 &= \frac{2TP}{2TP + FP + FN}
 \end{aligned}$$

โดยกำหนดให้

TP คือ ผลบวกจริง (True positive)

TN คือ ผลลบจริง (True negative)

FP คือ ผลบวกลวง (False positive)

FN คือ ผลลบลวง (False negative)



## บทที่ 4

### การทดลอง และผลลัพธ์ของการวิจัย

งานวิจัยนี้ได้ใช้ทรัพยากรจาก Google Colab ในการฝึกสอนแบบจำลอง โดยใช้ GPU ชนิด A100 และชุดเครื่องมือ (library) ที่ใช้เป็นหลักในการทำงานคือ TensorFlow ผู้ทดลองได้ทำการสร้างแบบจำลองเพื่อจำแนกใบหน้าแมวที่มีความเจ็บปวดออกจากใบหน้าแมวที่ไม่มีความเจ็บปวด โดยใช้วิธีการถ่ายทอดองค์ความรู้จากสถาปัตยกรรม (architecture) EfficientNetB7 และ ResNet50v2 ได้เลือกใช้ค่าน้ำหนักเบื้องต้น (initial weight) จากการฝึกสอนแบบจำลองด้วยชุดข้อมูลจาก ImageNet และบันทึกผลการทดลองเพื่อเปรียบเทียบประสิทธิภาพของการฝึกสอนแบบจำลองทั้งสองสถาปัตยกรรม

เนื่องจากชุดข้อมูลภาพที่ได้รับนั้นมาจากหลากหลายแหล่งข้อมูลโดยภาพถ่ายส่วนใหญ่ใช้กล้องจากโทรศัพท์มือถือ การควบคุมสภาวะแวดล้อมขณะถ่ายภาพจึงเป็นไปได้ยาก ภาพในชุดข้อมูลจึงมีแสงเงาและมุมต่างๆ ที่แตกต่างกัน ก่อนที่จะนำภาพไปเข้าสู่กระบวนการแบ่งชุดข้อมูลจึงได้มีการคัดเลือกภาพที่เห็นอวัยวะบนหน้าแมวที่สนใจ (ได้แก่ ตาและหูทั้ง 2 ข้าง จมูก ปาก หนวด) ไม่ครบถ้วนออกไป

ชุดข้อมูลภาพที่ผ่านการคัดเลือกและการประเมินจากสัตวแพทย์ผู้เชี่ยวชาญแล้วนั้นมาประกอบไปด้วยภาพแมวกลุ่มที่ไม่มีความเจ็บปวด (no pain) ซึ่งเป็นกลุ่มที่มีคะแนนความเจ็บปวดเท่ากับ 0 จำนวน 100 ตัว และภาพแมวกลุ่มที่มีความเจ็บปวด (pain) เป็นกลุ่มที่มีคะแนนความเจ็บปวดอยู่ในช่วง 1-4 จำนวน 100 ตัว โดยคะแนนความเจ็บปวดนั้นจะเรียงตามระดับความเจ็บปวดจากน้อยไปมาก ในส่วนของการทดลองจะแบ่งเป็น 2 ช่วง คือการทดลองเบื้องต้นเนื่องจากผู้ทดลองได้ทำการทดลองเบื้องต้นจากชุดข้อมูลภาพกลุ่มละ 100 ตัวนี้ และการทดลองหลังจากปรับชุดข้อมูล เนื่องจากเมื่อได้ผลการทดลองเบื้องต้นแล้วได้ลองปรับชุดข้อมูลแบ่งกลุ่มที่ไม่มีความเจ็บปวดเป็นกลุ่มที่มีคะแนนความเจ็บปวดเท่ากับ 0 กับกลุ่มที่มีความเจ็บปวดเป็นกลุ่มที่มีคะแนนความเจ็บปวดอยู่ในช่วง 2-4 จำนวนกลุ่มละ 57 ตัว

#### 4.1 การทดลองเบื้องต้น

##### 4.1.1 การแบ่งกลุ่มข้อมูล

ชุดข้อมูลนี้ประกอบไปด้วยภาพแมวทั้งหมด 200 ภาพ ได้ถูกแบ่งเป็นกลุ่มที่ใช้ฝึกสอน (training data) กลุ่มที่ใช้ตรวจสอบ (validating data) และกลุ่มที่ใช้ทดสอบ (testing data) แบบ



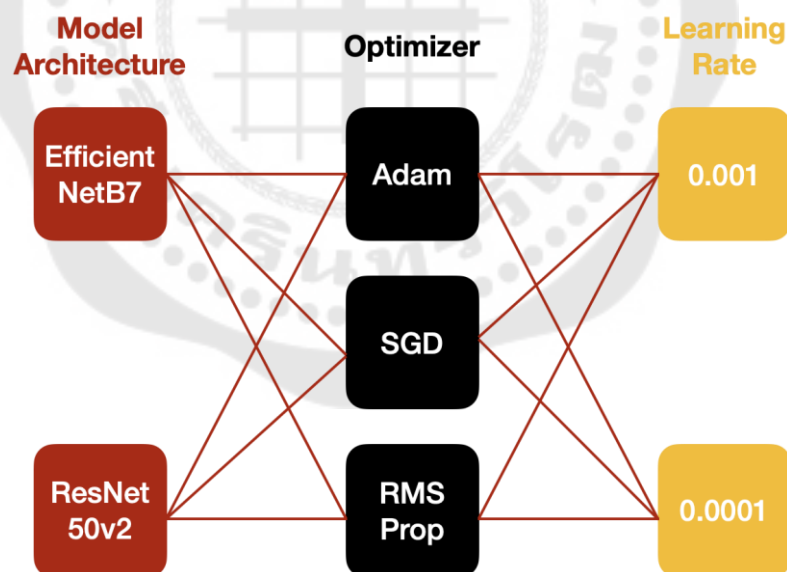
สุม ในอัตราส่วน 8:1:1 ผลลัพธ์ที่ได้จากขั้นตอนนี้คือภาพแมวที่ใช้ฝึกสอนจำนวนทั้งหมด 160 ภาพ กลุ่มตรวจสอบ 20 ภาพ และกลุ่มทดสอบ 20 ภาพ

#### 4.1.2 การเพิ่มจำนวนข้อมูลด้วยการดัดแปลงภาพ (data augmentation)

เนื่องจากชุดข้อมูลมีขนาดเล็กและมีความหลากหลายแสงเงาและมุมของภาพ ผู้ทดลองจึงได้ทำการเพิ่มจำนวนข้อมูลหลังจากที่แบ่งชุดข้อมูลแล้วเพื่อป้องกันการรั่วไหลของข้อมูล (data leakage) ด้วยการดัดแปลงภาพ โดยใช้ชุดเครื่องมือ albumentations กระทำการหมุนภาพ (rotation) ในช่วงมุม -5 ถึง 5 องศา กลับภาพจากซ้าย-ขวา (horizontal flip) และปรับคอนทราสต์ (contrast) ของภาพ โดยกระบวนการนี้จะเกิดขึ้นแบบสุ่ม และได้ผลลัพธ์ออกมาเป็นกลุ่มที่ใช้ฝึกสอนทั้งหมด 3200 ภาพ กลุ่มที่ใช้ตรวจสอบทั้งหมด 400 ภาพ และกลุ่มที่ใช้ทดสอบทั้งหมด 400 ภาพ

#### 4.1.3 การปรับค่าต่างๆ ในการฝึกสอนแบบจำลอง

ในการทดลองเบื้องต้นได้มีการทดลองฝึกสอนแบบจำลองด้วยไฮเปอร์พารามิเตอร์ (hyperparameter) ที่แตกต่างกัน ดังในภาพประกอบด้านล่าง



ภาพประกอบ 28 การปรับไฮเปอร์พารามิเตอร์ในการทดลองเบื้องต้น

ในการฝึกสอนแบบจำลองนั้น ผู้ทดลองได้ใช้โครงสร้างสถาปัตยกรรมหลัก (backbone architecture) แต่ละโครงสร้างเป็นส่วนที่ใช้สกัดฟีเจอร์ นำมาต่อกับชั้นเชื่อมโยงสมบูรณ์ (fully connected layers) ซึ่งประกอบไปด้วยชั้นที่มีหนดจำนวน 64 โหนด, 32 โหนด, และชั้นสุดท้ายซึ่ง

เป็นส่วนที่ใช้ทำนายผลลัพธ์เป็นโหนดที่ใช้อัลกอริทึม logistic regression โดยจำนวนตัวอย่างที่แบบจำลองใช้ในการปรับค่าน้ำหนัก (batch size) ของข้อมูลในการฝึกสอนแต่ละรอบมีค่าเท่ากับ 100 ทำให้ได้แบบจำลองที่มีสถาปัตยกรรม อัลกอริทึมที่ใช้ในการหาค่าที่ดีที่สุด (optimizer) และอัตราการเรียนรู้ที่แตกต่างกันทั้งสิ้น 12 แบบ ได้ผลลัพธ์ของการทำนายดังตาราง 1

ตาราง 1 ผลลัพธ์จากการทดลองเบื้องต้น

| สถาปัตยกรรมและไฮเปอร์พารามิเตอร์ |                |               | ผลการทดลอง  |            |             |             |
|----------------------------------|----------------|---------------|-------------|------------|-------------|-------------|
| Architecture                     | Optimizer      | Learning Rate | Accuracy    | Precision  | Recall      | F1-score    |
| EfficientNetB7                   | Adam           | 0.001         | 0.53        | 1.0        | 0.53        | 0.69        |
| EfficientNetB7                   | Adam           | 0.0001        | 0.55        | 1.0        | 0.55        | 0.71        |
| EfficientNetB7                   | SGD            | 0.001         | 0.45        | 1.0        | 0.45        | 0.62        |
| EfficientNetB7                   | SGD            | 0.0001        | 0.58        | 1.0        | 0.58        | 0.74        |
| EfficientNetB7                   | RMSProp        | 0.001         | 0.63        | 1.0        | 0.63        | 0.77        |
| <b>EfficientNetB7</b>            | <b>RMSProp</b> | <b>0.0001</b> | <b>0.73</b> | <b>1.0</b> | <b>0.73</b> | <b>0.84</b> |
| ResNet50v2                       | Adam           | 0.001         | 0.50        | 1.0        | 0.50        | 0.66        |
| ResNet50v2                       | Adam           | 0.0001        | 0.43        | 1.0        | 0.43        | 0.60        |
| ResNet50v2                       | SGD            | 0.001         | 0.0         | 0.0        | 0.0         | 0.0         |
| ResNet50v2                       | SGD            | 0.0001        | 0.56        | 1.0        | 0.55        | 0.71        |
| ResNet50v2                       | RMSProp        | 0.001         | 0.42        | 1.0        | 0.42        | 0.59        |
| ResNet50v2                       | RMSProp        | 0.0001        | 0.53        | 1.0        | 0.53        | 0.69        |

จากตารางด้านบนจะเห็นได้ว่าแบบจำลองที่ใช้กลุ่มสถาปัตยกรรม EfficientNetB7 มีแนวโน้มที่จะให้ผลการทำนายที่แม่นยำมากกว่ากลุ่ม ResNet50v2 โดยค่าไฮเปอร์พารามิเตอร์ที่ทำให้แบบจำลองมีความแม่นยำในการทำนายสูงที่สุดได้แก่ optimizer=RMSProp, learning rate 0.0001

## 4.2 การทดลองหลังจากการปรับชุดข้อมูล

### 4.2.1 การปรับชุดข้อมูลและแบ่งกลุ่มข้อมูล

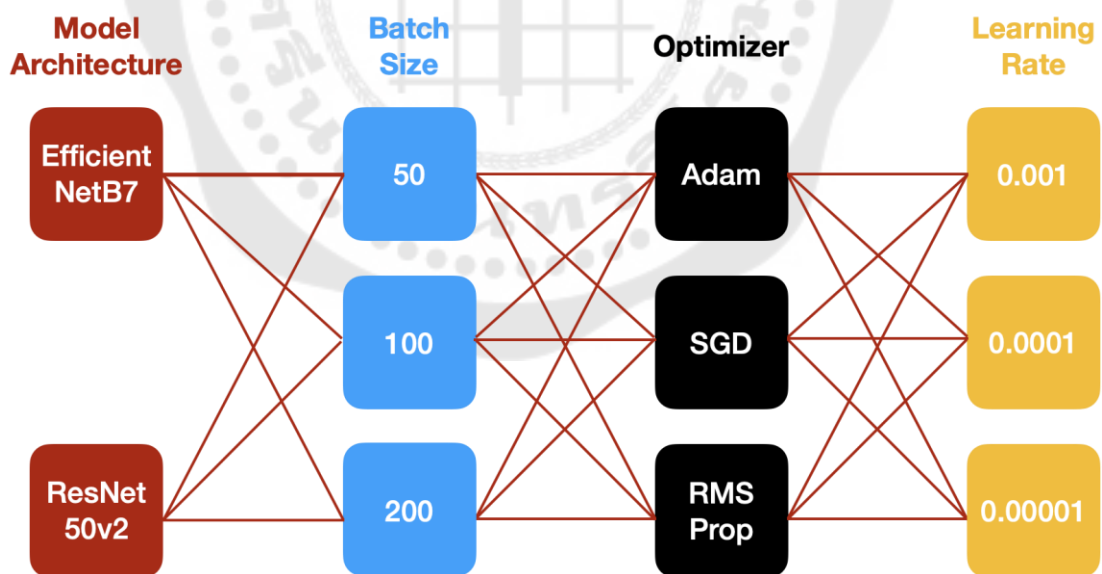
ชุดข้อมูลที่ใช้ในการทดลองครั้งนี้ได้นำภาพแมวที่ได้คะแนนความเจ็บปวดเท่ากับ 1 ออกจากกลุ่มแมวที่มีความเจ็บปวด ทำให้เหลือภาพแมวที่มีความเจ็บปวดทั้งสิ้น 57 ภาพ และได้ทำการสุ่มภาพจากกลุ่มแมวที่ไม่มีความเจ็บปวดมา 57 ภาพ รวมเป็น 114 ภาพ ภาพทั้งหมดได้ถูกแบ่งเป็นกลุ่มที่ใช้ฝึกสอน 94 ภาพ กลุ่มตรวจสอบ 10 ภาพ และกลุ่มที่ใช้ทดสอบทั้งหมด 10 ภาพ

### 4.2.2 การเพิ่มจำนวนข้อมูลด้วยการตัดแปลงภาพ

หลังจากที่ได้แบ่งกลุ่มข้อมูลแล้ว ภาพในแต่ละกลุ่มถูกนำมาเพิ่มจำนวนด้วยการตัดแปลงภาพเช่นเดียวกับใน 4.1.2 ได้ผลลัพธ์ออกมาเป็นกลุ่มที่ใช้ฝึกสอนทั้งหมด 940 ภาพ กลุ่มที่ใช้ตรวจสอบทั้งหมด 100 ภาพ และกลุ่มที่ใช้ทดสอบทั้งหมด 100 ภาพ

### 4.2.3 การปรับค่าต่างๆ ในการฝึกสอนแบบจำลอง

ในการทดลองเบื้องต้นได้มีการทดลองฝึกสอนแบบจำลองด้วยไฮเปอร์พารามิเตอร์ที่แตกต่างกัน โดยในการทดลองครั้งนี้ได้มีการเพิ่มการปรับขนาดจำนวนตัวอย่างที่แบบจำลองใช้ในการปรับค่านำหนักของข้อมูลในการฝึกสอนแต่ละรอบเข้ามาด้วย ดังในภาพประกอบด้านล่าง



ภาพประกอบ 29 การปรับไฮเปอร์พารามิเตอร์ในการทดลองหลังจากการปรับชุดข้อมูล

ทำให้ได้แบบจำลองที่มีค่าไฮเปอร์พารามิเตอร์แตกต่างกันทั้งสิ้น 54 แบบ และได้ผลลัพธ์ของการทำนายดังตาราง 2 และ 3

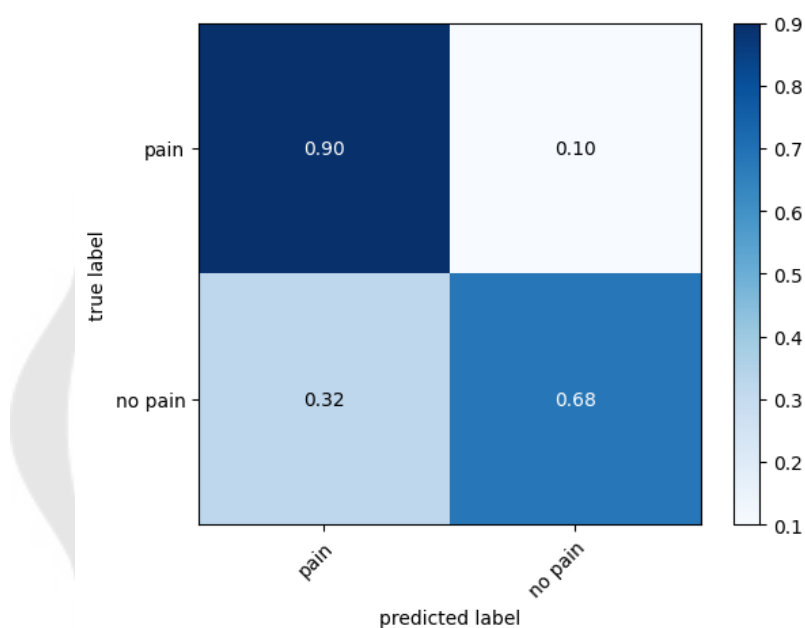
ตาราง 2 ผลลัพธ์จากการทำนายด้วยแบบจำลองที่มีโครงสร้างเป็น EfficientNetB7

| สถาปัตยกรรมและไฮเปอร์พารามิเตอร์ |            |               | ผลการทดลอง   |             |             |            |
|----------------------------------|------------|---------------|--------------|-------------|-------------|------------|
| Architecture                     | Optimizer  | Learning Rate | Accuracy     | Precision   | Recall      | F1-score   |
| EfficientNetB7                   | 50         | Adam          | 0.001        | 0.54        | 0.53        | 0.64       |
| EfficientNetB7                   | 100        | Adam          | 0.001        | 0.73        | 0.65        | 1.0        |
| EfficientNetB7                   | 200        | Adam          | 0.001        | 0.66        | 0.61        | 0.92       |
| EfficientNetB7                   | 50         | Adam          | 0.0001       | 0.66        | 0.68        | 0.6        |
| EfficientNetB7                   | 100        | Adam          | 0.0001       | 0.61        | 0.58        | 0.78       |
| EfficientNetB7                   | 200        | Adam          | 0.0001       | 0.62        | 0.61        | 0.66       |
| EfficientNetB7                   | 50         | Adam          | 0.00001      | 0.63        | 0.6         | 0.78       |
| EfficientNetB7                   | 100        | Adam          | 0.00001      | 0.59        | 0.59        | 0.6        |
| EfficientNetB7                   | 200        | Adam          | 0.00001      | 0.71        | 0.65        | 0.9        |
| EfficientNetB7                   | 50         | SGD           | 0.001        | 0.55        | 0.54        | 0.76       |
| <b>EfficientNetB7</b>            | <b>100</b> | <b>SGD</b>    | <b>0.001</b> | <b>0.79</b> | <b>0.74</b> | <b>0.9</b> |
| EfficientNetB7                   | 200        | SGD           | 0.001        | 0.62        | 0.6         | 0.72       |
| EfficientNetB7                   | 50         | SGD           | 0.0001       | 0.5         | 0.51        | 0.66       |
| EfficientNetB7                   | 100        | SGD           | 0.0001       | 0.58        | 0.59        | 0.54       |
| EfficientNetB7                   | 200        | SGD           | 0.0001       | 0.37        | 0.4         | 0.52       |
| EfficientNetB7                   | 50         | SGD           | 0.00001      | 0.37        | 0.39        | 0.44       |
| EfficientNetB7                   | 100        | SGD           | 0.00001      | 0.37        | 0.41        | 0.6        |
| EfficientNetB7                   | 200        | SGD           | 0.00001      | 0.22        | 0.11        | 0.08       |
| EfficientNetB7                   | 50         | RMSProp       | 0.001        | 0.63        | 0.61        | 0.74       |
| EfficientNetB7                   | 100        | RMSProp       | 0.001        | 0.66        | 0.63        | 0.78       |
| EfficientNetB7                   | 200        | RMSProp       | 0.001        | 0.57        | 0.58        | 0.52       |
| EfficientNetB7                   | 50         | RMSProp       | 0.0001       | 0.7         | 0.67        | 0.8        |
| EfficientNetB7                   | 100        | RMSProp       | 0.0001       | 0.71        | 0.63        | 1.0        |
| EfficientNetB7                   | 200        | RMSProp       | 0.0001       | 0.74        | 0.68        | 0.9        |
| EfficientNetB7                   | 50         | RMSProp       | 0.00001      | 0.64        | 0.59        | 0.9        |
| EfficientNetB7                   | 100        | RMSProp       | 0.00001      | 0.5         | 0.5         | 0.54       |
| EfficientNetB7                   | 200        | RMSProp       | 0.00001      | 0.61        | 0.6         | 0.66       |

ตาราง 3 ผลลัพธ์จากการทำนายด้วยแบบจำลองที่มีโครงสร้างเป็น ResNet50v2

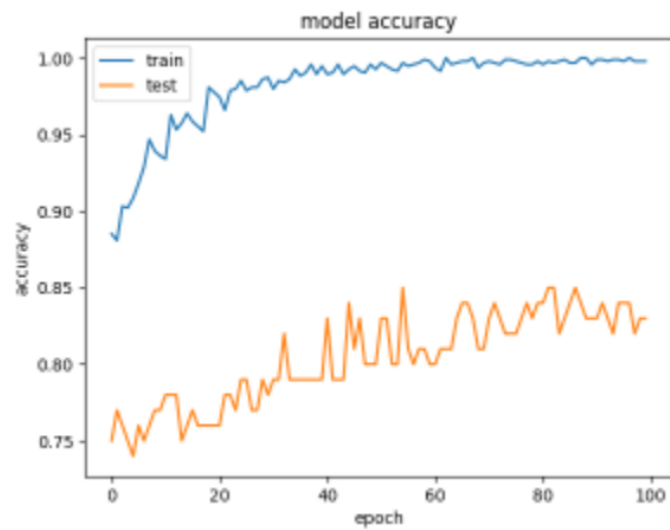
| สถาปัตยกรรมและไฮเปอร์พารามิเตอร์ |           |               | ผลการทดลอง |           |        |          |
|----------------------------------|-----------|---------------|------------|-----------|--------|----------|
| Architecture                     | Optimizer | Learning Rate | Accuracy   | Precision | Recall | F1-score |
| ResNet50v2                       | 50        | Adam          | 0.001      | 0.59      | 0.57   | 0.76     |
| ResNet50v2                       | 100       | Adam          | 0.001      | 0.56      | 0.57   | 0.48     |
| ResNet50v2                       | 200       | Adam          | 0.001      | 0.50      | 0.0    | 0.0      |
| ResNet50v2                       | 50        | Adam          | 0.0001     | 0.51      | 0.51   | 0.58     |
| ResNet50v2                       | 100       | Adam          | 0.0001     | 0.62      | 0.73   | 0.38     |
| ResNet50v2                       | 200       | Adam          | 0.0001     | 0.70      | 0.66   | 0.82     |
| ResNet50v2                       | 50        | Adam          | 0.00001    | 0.67      | 0.68   | 0.64     |
| ResNet50v2                       | 100       | Adam          | 0.00001    | 0.54      | 0.54   | 0.58     |
| ResNet50v2                       | 200       | Adam          | 0.00001    | 0.57      | 0.58   | 0.52     |
| ResNet50v2                       | 50        | SGD           | 0.001      | 0.50      | 0.0    | 0.0      |
| ResNet50v2                       | 100       | SGD           | 0.001      | 0.50      | 0.5    | 1.0      |
| ResNet50v2                       | 200       | SGD           | 0.001      | 0.49      | 0.0    | 0.0      |
| ResNet50v2                       | 50        | SGD           | 0.0001     | 0.60      | 0.57   | 0.8      |
| ResNet50v2                       | 100       | SGD           | 0.0001     | 0.48      | 0.49   | 0.78     |
| ResNet50v2                       | 200       | SGD           | 0.0001     | 0.57      | 0.62   | 0.36     |
| ResNet50v2                       | 50        | SGD           | 0.00001    | 0.47      | 0.47   | 0.44     |
| ResNet50v2                       | 100       | SGD           | 0.00001    | 0.58      | 0.79   | 0.22     |
| ResNet50v2                       | 200       | SGD           | 0.00001    | 0.35      | 0.38   | 0.5      |
| ResNet50v2                       | 50        | RMSProp       | 0.001      | 0.62      | 0.61   | 0.68     |
| ResNet50v2                       | 100       | RMSProp       | 0.001      | 0.66      | 0.69   | 0.58     |
| ResNet50v2                       | 200       | RMSProp       | 0.001      | 0.50      | 0.0    | 0.0      |
| ResNet50v2                       | 50        | RMSProp       | 0.0001     | 0.66      | 0.63   | 0.8      |
| ResNet50v2                       | 100       | RMSProp       | 0.0001     | 0.69      | 0.69   | 0.7      |
| ResNet50v2                       | 200       | RMSProp       | 0.0001     | 0.67      | 0.65   | 0.72     |
| ResNet50v2                       | 50        | RMSProp       | 0.00001    | 0.58      | 0.62   | 0.42     |
| ResNet50v2                       | 100       | RMSProp       | 0.00001    | 0.63      | 0.64   | 0.6      |
| ResNet50v2                       | 200       | RMSProp       | 0.00001    | 0.60      | 0.69   | 0.36     |

จากตารางผลการทดลอง แบบจำลองที่ให้ค่าความแม่นยำในการทำนายสูงที่สุดมีโครงสร้างมาจากสถาปัตยกรรม EfficientNetB7 โดยใช้จำนวนตัวอย่างที่แบบจำลองใช้ในการปรับค่าน้ำหนักเท่ากับ 100 อัลกอริทึมที่ใช้ในการหาค่าที่ดีที่สุดเป็น SGD และอัตราการเรียนรู้เป็น 0.001 ซึ่งแบบจำลองนี้ให้ค่าความแม่นยำในการทำนาย 79% ค่าความเที่ยงตรง 74% ค่าความไว 90% และค่าเฉลี่ยแบบฮาร์โมนิกของความเที่ยงตรงและความไว 81% ดังแสดงใน confusion matrix ในภาพประกอบ 29

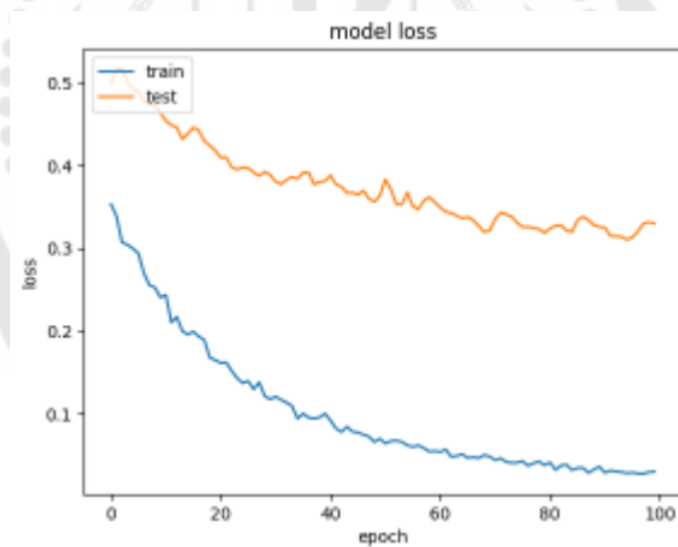


ภาพประกอบ 30 Confusion matrix แสดงผลลัพธ์จากแบบจำลองที่ดีที่สุด

ความสัมพันธ์ระหว่างค่าความแม่นยำและรอบในการฝึกสอนแบบจำลองของกลุ่มข้อมูลฝึกสอนและกลุ่มข้อมูลตรวจสอบ สามารถแสดงในรูปแบบกราฟได้ดังในภาพประกอบ 30 และความสัมพันธ์ระหว่างค่าความสูญเสียและรอบในการฝึกสอนแบบจำลองของกลุ่มข้อมูลฝึกสอนและกลุ่มข้อมูลตรวจสอบ สามารถแสดงในรูปแบบกราฟได้ดังในภาพประกอบ 31



ภาพประกอบ 31 ค่าความแม่นยำในการฝึกสอนแบบจำลอง



ภาพประกอบ 32 ค่าความสูญเสียในการฝึกสอนแบบจำลอง

#### 4.3 การอธิบายความสำคัญของฟีเจอร์ในแบบจำลอง

ผู้ทดลองได้ใช้อัลกอริทึม Grad-CAM หรือ Gradient-weighted Class Activation Mapping ซึ่งเป็นการอธิบายให้เห็นว่าพิกเซลหรือบริเวณใดของภาพมีความสำคัญต่อการทำนายผลลัพธ์ของแบบจำลอง แสดงให้เห็นตัวอย่างได้ดังภาพประกอบ 32



ภาพประกอบ 33 ตัวอย่างของภาพในชุดข้อมูลทดสอบที่มีการเน้นจุดสนใจโดย Grad-CAM

โดย Grad-CAM จะใช้เฉดสีม่วงเพื่อแสดงให้เห็นถึงกลุ่มพิกเซลที่มีความสำคัญน้อยที่สุดในการทำนาย เรียงเฉดสีไปจนถึงสีแดงที่มีความสำคัญมากที่สุด ดังในภาพประกอบ 33



Least  
important

Most  
important

ภาพประกอบ 34 สีแสดงระดับความสำคัญของแต่ละบริเวณของภาพในการทำนาย



## บทที่ 5

### การสรุปผลการวิจัย การอภิปรายผล และข้อเสนอแนะของการวิจัย

เนื่องจากผู้ทดลองเป็นเจ้าของแมวที่ไม่ได้มีประสบการณ์มากนัก แนวคิดของงานวิจัยชิ้นนี้เกิดขึ้นจากการได้รู้จักมาตรวจวัดความเจ็บปวดจากการแสดงสีหน้าของแมวขณะที่แมวของตนเองมีอาการป่วย ซึ่งเครื่องมือนี้อาจช่วยให้เจ้าของแมวสามารถสังเกตอาการของสัตว์เบื้องต้นด้วยตนเองได้ง่ายขึ้น อย่างไรก็ตามการใช้งานมาตรวัดนี้ต้องมีประสิทธิภาพต้องอาศัยประสบการณ์ของเจ้าของแมวและการฝึกอบรมร่วมด้วย จึงเป็นที่มาของการสร้างแบบจำลองโครงข่ายประสาทเทียมเพื่อเป็นเครื่องมือช่วยจำแนกความเจ็บปวดจากภาพถ่ายใบหน้าของแมว

#### 5.1 การแบ่งกลุ่ม pain และ no pain

สาเหตุของการแบ่งการทดลองออกเป็นการทดลองเบื้องต้น และการทดลองหลังจากการปรับชุดข้อมูล เนื่องจากการทดลองเบื้องต้นนั้นใช้ภาพถ่ายใบหน้าแมวที่มีคะแนนความเจ็บปวดเท่ากับ 1 รวมอยู่ในกลุ่ม pain ด้วย ซึ่งแมวกุุ่มนี้มักจะเป็นแมวที่มีอาการน้อย หรือมีความกำกวมในการประเมินอยู่ อาจส่งผลให้แบบจำลองมีความสับสนในการทำนายผลลัพธ์ จึงได้ทำการทดลองซ้ำโดยตัดภาพถ่ายแมวกุุ่มนี้ออกจากการฝึกสอนแบบจำลอง ซึ่งจากผลการทดลองความแม่นยำของแบบจำลองที่ฝึกสอนหลังภาพกำกวมเหล่านี้ออกมีแนวโน้มสูงขึ้น

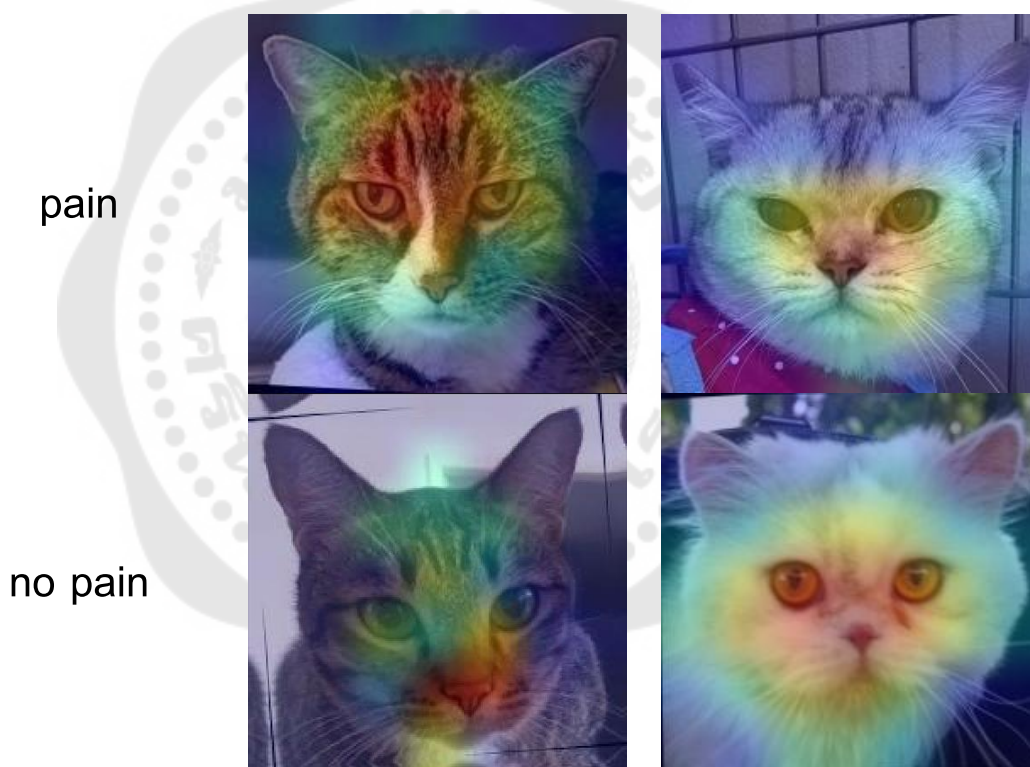
#### 5.2 การอธิบายความสำคัญของพีเจอรืในแบบจำลอง

โดยทั่วไปในแบบจำลองเชิงเส้น ความสำคัญของพีเจอรืสามารถสังเกตได้จากสัมประสิทธิ์ของพีเจอรืในสมการ หากสัมประสิทธิ์ของพีเจอรืมีค่าสัมบูรณ์ (absolute) มากสามารถตีความได้ว่าพีเจอรืนั้นๆ มีความสำคัญต่อการทำนายผลลัพธ์ของแบบจำลองมาก

ในแบบจำลองโครงข่ายประสาทเทียมชนิดคอนโวลูชันนั้นเราสามารถอธิบายความสำคัญของพีเจอรืด้วยอัลกอริทึม Grad-CAM ซึ่งทำงานโดยใช้หลักการ backpropagation จากชั้นเชื่อมโยงสมบูรณ์ชั้นสุดท้ายกลับไปยังโครงสร้างหลักที่ใช้สกัดพีเจอรืโดยมักจะทำการ backpropagate ไปยังชั้นท้ายของโครงสร้างที่ใช้สกัดพีเจอรืเนื่องจากผลลัพธ์ที่ได้จากชั้นเหล่านี้เป็นพีเจอรืที่มีความเป็นรูปธรรมมากกว่า (high level features) พีเจอรืที่ได้จากชั้นต้น (low level features)

ผลจากการทำ backpropagation จะเป็นตัวบ่งบอกอิทธิพลของแผนผังพีเจอร์ในภาพที่ส่งผลต่อการทำนายผลลัพธ์ แสดงออกมาในรูปแบบแผนภูมิความร้อน (heatmap) ดังตัวอย่างในภาพประกอบ 34

จะเห็นได้ว่าบริเวณที่แบบจำลองให้ความสำคัญกับบริเวณกลางใบหน้า (central) เช่น ตา จมูก ปากบน มากกว่าบริเวณรอบนอก (peripheral) เช่น ใบหู ซึ่งมีความสอดคล้องกับการทดลองของ M. Feigelstein และคณะ [8] ข้อสันนิษฐานหนึ่งของผู้ทดลองคือ เนื่องจากขนาดของอวัยวะกลางใบหน้าของแมวในแต่ละสายพันธุ์มักมีความใกล้เคียงกัน เทียบกับใบหูที่มีความเล็กใหญ่แตกต่างกันไป จึงทำให้การทำนายจากใบหูนั้นมีความยากกว่า



ภาพประกอบ 35 ตัวอย่างการอธิบายภาพที่แบบจำลองทำนายถูกต้อง

นอกจากนี้ยังมีภาพที่มีปัจจัยกวนจากสิ่งแวดล้อมที่ทำให้แบบจำลองทายพลาดไป ดังตัวอย่างในภาพประกอบ 35

no pain



ภาพที่แบบจำลองทำนายผิด



แผนภูมิความร้อน

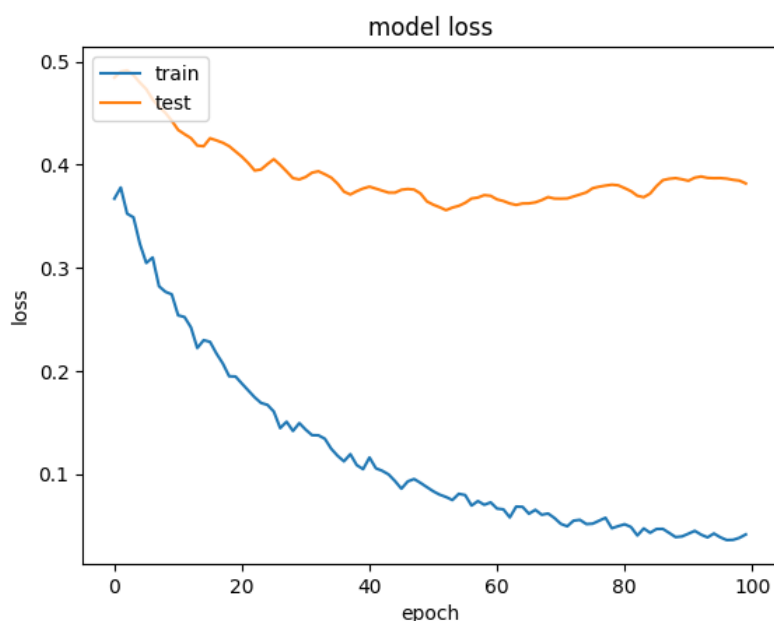
ภาพประกอบ 36 ตัวอย่างการอธิบายภาพที่แบบจำลองทำนายผิด

จากภาพจะเห็นได้ว่าแบบจำลองนั้นมีการจับความสนใจไปยังบริเวณภาพพื้นหลังซึ่งอาจมีลักษณะที่คล้ายคลึงกับแผนภูมิพีเจอร์ที่เคยเรียนรู้ จึงเกิดความผิดพลาดในการทำนายได้

### 5.3 ข้อจำกัดของงานวิจัย

ข้อจำกัดของงานวิจัยชิ้นนี้คือชุดข้อมูลที่ใช้ในการฝึกสอนแบบจำลอง เนื่องจากภายใต้ระยะเวลาและทรัพยากรบุคคลที่จำกัด การเก็บข้อมูลจึงอาศัยความอนุเคราะห์ของสัตวแพทย์ในการส่งภาพถ่ายหน้าแมวจากโทรศัพท์มือถือซึ่งไม่มีการควบคุมยี่ห้อหรือรุ่นของโทรศัพท์ ทำให้ภาพถ่ายที่ได้อาจมีการแตกต่างกันในแง่ของความคมชัด ความสดของสี การถ่ายภาพไม่ได้มีการควบคุมแสงและองศาของการถ่ายอย่างเคร่งครัดจึงทำให้มีความเข้มแสงที่ต่างกัน และการบิดของมุมต่างๆ ของหน้าแมว ทำให้การจดจำแบบแผนโดยแบบจำลองเป็นไปได้ยากขึ้น การเพิ่มจำนวนข้อมูลโดยการดัดแปลงภาพมีจุดประสงค์เพื่อเพิ่มคุณลักษณะของภาพให้สอดคล้องกับความคลาดเคลื่อนที่เกิดขึ้นจากการไม่ได้ควบคุมสภาวะแวดล้อมของการถ่ายภาพ อีกทั้งมีข้อจำกัดทางด้านจำนวนของข้อมูลภาพที่รวบรวมได้นั้นมีปริมาณน้อยเนื่องจากแมวเป็นสัตว์ที่มีนิสัยชอบบดบังอาการอ่อนแอของตนเองและมีความพยายามที่จะไม่แสดงอาการอ่อนแอออกมา ทำให้โอกาสที่จะได้ภาพถ่ายหน้าแมวที่มีลักษณะเจ็บปวดอย่างชัดเจนนั้นเป็นไปได้ยาก โดยทั่วไปในการฝึกสอนแบบจำลองใดๆ นั้นหากปริมาณค่าพารามิเตอร์หรือค่าน้ำหนักถ่วงมีมากเมื่อเทียบกับจำนวนข้อมูล มักจะก่อให้เกิดภาวะ overfitting หรือการรู้จำเพียงลักษณะของข้อมูลฝึกสอนเมื่อนำไปทำนายข้อมูลใหม่ที่ไม่เคยเจอมาก่อนกลับให้ผลลัพธ์ความแม่นยำที่ไม่ดี ซึ่งในการทดลองเพื่อฝึกสอนแบบจำลองโครงข่ายประสาทเทียมนั้นสถาปัตยกรรม EfficientNetB7 มีปริมาณค่าพารามิเตอร์ที่ปรับได้ทั้งหมดเท่ากับ 1,804,417 และสถาปัตยกรรม ResNet50V2 มี

ปริมาณค่าพารามิเตอร์ที่ปรับได้ทั้งหมดเท่ากับ 133,249 จึงเป็นสาเหตุของการเกิดภาวะ overfitting อย่างเล็งไม่ได้ในงานวิจัยชิ้นนี้ ตัวอย่างด้านล่างแสดงความสัมพันธ์ระหว่างปริมาณรอบในการเรียนรู้และค่าความสูญเสียของชุดข้อมูลฝึกสอนและชุดข้อมูลทดสอบซึ่งเริ่มมีการแยกออกจากกัน นั่นคือแม้ค่าความสูญเสียของชุดข้อมูลฝึกสอนยังคงมีแนวโน้มลดลง แต่ความสูญเสียของชุดข้อมูลทดสอบเกิดภาวะอิมิตัว และมีแนวโน้มที่จะเพิ่มขึ้น บ่งถึงการเกิดภาวะ overfitting



ภาพประกอบ 37 ภาพความสัมพันธ์ระหว่างรอบในการเรียนรู้และค่าความสูญเสียของชุดข้อมูล

เมื่อปริมาณของข้อมูลน้อย การกระจายตัว (distribution) ของลักษณะโครงหน้าของแมวในชุดข้อมูลกลุ่มแมวที่มีความเจ็บปวดและไม่เจ็บปวดก็เป็นปัจจัยหนึ่งที่ทำให้การฝึกสอนแบบจำลองทำได้ยากขึ้น เนื่องจากไม่ได้มีการควบคุมสายพันธุ์ของแมวในชุดข้อมูล จึงอาจเกิดความแตกต่างของการกระจายตัวของลักษณะโครงหน้าระหว่างชุดข้อมูลฝึกสอนและชุดข้อมูลทดสอบได้ เนื่องจากการกระบวนกรแบ่งกลุ่มเกิดขึ้นแบบสุ่ม

ในแง่ของการเก็บข้อมูลงานวิจัย เนื่องจากข้อมูลส่วนใหญ่เป็นข้อมูลภาพจากแมวที่เข้ารับการรักษาโรคหรือรับการฉีดวัคซีนที่คลินิก สภาวะอารมณ์ของแมวนั้นอาจไม่เหมือนกับแมวในสภาวะปกติเนื่องจากแมวอาจมีความตื่นเต้นหรือความกลัวได้ นอกจากนี้การแสดงสีหน้าของแมวในบางภาวะ เช่น ภาวะเบื่อหรือง่วงนอน ก็อาจมีความคล้ายคลึงกับอาการเจ็บปวดได้เช่นกัน ทัศนียภาพเหล่านี้ทำให้แบบจำลองมีผลบวกหลงได้ ดังตัวอย่างภาพแมวปกติในภาพประกอบ 38 จะสังเกตเห็นว่า

มีการหรีตาเล็กน้อย ซึ่งมีความคล้ายคลึงกับแมวที่มีอาการเจ็บปวด โดยในภาพประกอบ 38 นี้แบบจำลองทำนายว่าแมวในภาพมีความเจ็บปวด และให้ค่าความน่าจะเป็นอยู่ที่ 0.847



ภาพประกอบ 38 ตัวอย่างภาพแมว

ที่มา: smudge\_lord, "Smudge - Table Cat | If you're going to post my face on the internet, at least tag me 🙄 #sassy #sassycat #catmeme #smudge #whitecat #memes #memelord #salad... | Instagram". สืบค้น: 4 พฤษภาคม 2024. [ออนไลน์]. Available at: <https://www.instagram.com/p/Bx-UY1Qp1I9/?igsh=eG5yN205emhwcm1i>

#### 5.4 ข้อเสนอแนะสำหรับงานวิจัย

หากข้อมูลภาพถ่ายที่นำมาใช้ฝึกสอนแบบจำลองมีการควบคุมสภาวะแวดล้อมเช่น พื้นหลัง แสงเงา ความเข้มแสงของภาพ ใช้อุปกรณ์ถ่ายภาพชนิดเดียวกัน มุมภาพถ่ายเดียวกัน อาจทำให้การฝึกสอนแบบจำลองเป็นไปได้ง่ายขึ้น หากภาพที่นำมาใช้ฝึกสอนมีพื้นหลังที่ต่างกันควรมีการจัดการข้อมูลก่อนฝึกสอนเพื่อลดสัญญาณรบกวนจากสภาพแวดล้อม เช่น อาจใช้เทคนิค segmentation ร่วมด้วยเพื่อตัดเฉพาะรูปใบหน้าแมวมาฝึกสอน

การเพิ่มขนาดของภาพที่ใช้ฝึกสอนอาจช่วยเพิ่มประสิทธิภาพในการเรียนรู้ของแบบจำลองได้ ในงานวิจัย [31] กล่าวถึงการเพิ่มประสิทธิภาพของการเรียนรู้โดยการเพิ่มความลึก ความกว้าง และความละเอียดของภาพในอัตราส่วนที่พอเหมาะ ความลึกหมายถึงจำนวนชั้นในแบบจำลอง เมื่อใช้ภาพที่มีขนาดใหญ่ขึ้นควรเพิ่มความลึกเพื่อตรวจจับภาพรวมของภาพให้ได้ทั้งหมด ความกว้างหมายถึงปริมาณฟิลเตอร์ที่ใช้เพื่อเพิ่มความสามารถในการจดจำแบบแผนที่มีความละเอียดสูง (fine-grained pattern) สถาปัตยกรรม EfficientNetB7 นั้นได้มีใช้ความลึกและความกว้างที่ค่อนข้างมากอยู่แล้ว หากในอนาคตต้องการฝึกสอนแบบจำลองชนิดนี้ อาจทำได้โดยการใส่ภาพที่มีขนาดและความละเอียดที่สูงขึ้น

ความหลากหลายทางกายวิภาคของใบหน้าแมวในชุดข้อมูลฝึกสอนเป็นปัจจัยที่ทำให้แบบจำลองนั้นมีความสามารถในการทำนายประชากรทั่วไปได้มากขึ้น ไม่จำเป็นเจาะจงต่อแมวลักษณะใดเป็นพิเศษ ชุดข้อมูลที่นำมาฝึกสอนจึงควรมีทั้งปริมาณและความหลากหลายของภาพถ่ายที่มากขึ้น

หากมีปริมาณข้อมูลฝึกสอนปริมาณมาก การทำนายความเจ็บปวดโดยใช้แบบจำลองโครงข่ายประสาทเทียมอาจทำได้ละเอียดขึ้น เช่นแบบจำลองในการทำนายคะแนนความเจ็บปวดของแมว



บรรณานุกรม



- [1] P. Ekman และ W. V. Friesen, "Facial action coding system", *Environmental Psychology & Nonverbal Behavior*, 1978.
- [2] E. A. Clark และคณะ, "The Facial Action Coding System for Characterization of Human Affective Response to Consumer Product-Based Stimuli: A Systematic Review", *Front. Psychol.*, ปี 11, น. 920, พ.ค. 2020, doi: 10.3389/fpsyg.2020.00920.
- [3] C. C. Caeiro, A. M. Burrows, และ B. M. Waller, "Development and application of CatFACS: Are human cat adopters influenced by cat facial expressions?", *Applied Animal Behaviour Science*, ปี 189, น. 66–78, เม.ย. 2017, doi: 10.1016/j.applanim.2017.01.005.
- [4] J. T. Brondani และคณะ, "Validation of the English version of the UNESP-Botucatu multidimensional composite pain scale for assessing postoperative pain in cats", *BMC Vet Res*, ปี 9, ฉบับที่ 1, น. 143, ธ.ค. 2013, doi: 10.1186/1746-6148-9-143.
- [5] M. C. Evangelista และคณะ, "Facial expressions of pain in cats: the development and validation of a Feline Grimace Scale", *Sci Rep*, ปี 9, ฉบับที่ 1, น. 19128, ธ.ค. 2019, doi: 10.1038/s41598-019-55693-8.
- [6] L. R. Finka และคณะ, "Geometric morphometrics for the study of facial expressions in non-human animals, using the domestic cat as an exemplar", *Scientific Reports*, ปี 9, ฉบับที่ 1, น. 9883, ก.ค. 2019, doi: 10.1038/s41598-019-46330-5.
- [7] M. Feighelstein, I. Shimshoni, L. R. Finka, S. P. L. Luna, D. S. Mills, และ A. Zamansky, "Automated recognition of pain in cats", *Sci Rep*, ปี 12, ฉบับที่ 1, น. 9575, มิ.ย. 2022, doi: 10.1038/s41598-022-13348-1.
- [8] M. Feighelstein และคณะ, "Explainable automated pain recognition in cats", *Sci Rep*, ปี 13, ฉบับที่ 1, น. 8973, มิ.ย. 2023, doi: 10.1038/s41598-023-35846-6.
- [9] K. He, X. Zhang, S. Ren, และ J. Sun, "Deep Residual Learning for Image Recognition", ใน *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA: IEEE, มิ.ย. 2016, น. 770–778. doi: 10.1109/CVPR.2016.90.
- [10] "What is machine learning?", IBM. [ออนไลน์]. Available at: <https://www.ibm.com/topics/machine-learning>



- [11] J. Chen, "What Is a Neural Network?", Investopedia. [ออนไลน์]. Available at: <https://www.investopedia.com/terms/n/neuralnetwork.asp>
- [12] J. Starmer, "Neural Networks Part 1: Inside the black box", StatQuest. [ออนไลน์]. Available at: <https://statquest.org/neural-networks-part-1-inside-the-black-box/>
- [13] S. Mahapatra, "Why Deep Learning over Traditional Machine Learning?", Towards Data Science. [ออนไลน์]. Available at: <https://towardsdatascience.com/why-deep-learning-is-needed-over-traditional-machine-learning-1b6a99177063>
- [14] N. Phongchit, "Convolutional Neural Network (CNN) คืออะไร", Medium. [ออนไลน์]. Available at: <https://medium.com/@natthawatphongchit/e3f5d73eebaa>
- [15] A. Dertat, "Applied Deep Learning - Part 4: Convolutional Neural Networks", Towards Data Science. [ออนไลน์]. Available at: <https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2>
- [16] D. Podareanu, V. B. Codreanu, S. Aigner, C. V. Leeuwen, และ V. Weinberg, "Best Practice Guide - Deep Learning", 2019, doi: 10.13140/RG.2.2.31564.05769.
- [17] A. Géron, "14. Deep Computer Vision Using Convolutional Neural Networks", ใน *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow*, 2nd พิมพ์ครั้งที่, Sebastopol, CA: O'Reilly Media, Inc., น. 456–458.
- [18] R. Qayyum, "Introduction To Pooling Layers In CNN", Towards AI. [ออนไลน์]. Available at: <https://towardsai.net/p//introduction-to-pooling-layers-in-cnn>
- [19] "Understanding Convolutional Neural Networks (CNNs): A Complete Guide", LearnOpenCV. [ออนไลน์]. Available at: <https://learnopencv.com/understanding-convolutional-neural-networks-cnn/>
- [20] H. D. Regua, "Introducing Transfer Learning as Your Next Engine to Drive Future Innovations", Medium. [ออนไลน์]. Available at: <https://medium.datadriveninvestor.com/introducing-transfer-learning-as-your-next-engine-to-drive-future-innovations-5e81a15bb567>

- [21] R. Poojary, R. Raina, และ A. Kumar Mondal, “Effect of data-augmentation on fine-tuned CNN model performance”, *IJ-AI*, ปี 10, ฉบับที่ 1, น. 84, มี.ค. 2021, doi: 10.11591/ijai.v10.i1.pp84-92.
- [22] R. Pramoditha, “How to Choose the Optimal Learning Rate for Neural Networks”, Towards Data Science. [ออนไลน์]. Available at: <https://towardsdatascience.com/how-to-choose-the-optimal-learning-rate-for-neural-networks-362111c5c783>
- [23] Coding Lane, “Momentum Optimizer in Deep Learning | Explained in Detail”, YouTube. [ออนไลน์]. Available at: <https://www.youtube.com/watch?v=Vce8w1sy0e8>
- [24] DeepLearningAI, “RMSProp (C2W2L07)”, YouTube. [ออนไลน์]. Available at: [https://www.youtube.com/watch?v=\\_e-LFe\\_igno](https://www.youtube.com/watch?v=_e-LFe_igno)
- [25] DeepLearningAI, “Adam Optimization Algorithm (C2W2L08)”, YouTube. [ออนไลน์]. Available at: [https://www.youtube.com/watch?v=JXQT\\_vxqwlS](https://www.youtube.com/watch?v=JXQT_vxqwlS)
- [26] P. Nakwijit, “ทำความเข้าใจ Optimizer”, Medium. [ออนไลน์]. Available at: <https://medium.com/@chameleontk/a44455615c32>
- [27] A. Nikishaev, “How to debug neural networks. Manual.”, Medium. [ออนไลน์]. Available at: <https://medium.com/machine-learning-world/dc2a200f10f2>
- [28] H. Shipley, A. Guedes, L. Graham, E. Goudie-DeAngelis, และ E. Wendt-Hornickle, “Preliminary appraisal of the reliability and validity of the Colorado State University Feline Acute Pain Scale”, *Journal of Feline Medicine and Surgery*, ปี 21, ฉบับที่ 4, น. 335–339, เม.ย. 2019, doi: 10.1177/1098612X18777506.
- [29] M. C. Evangelista และ P. V. Steagall, “Agreement and reliability of the Feline Grimace Scale among cat owners, veterinarians, veterinary students and nurses”, *Sci Rep*, ปี 11, ฉบับที่ 1, น. 5262, มี.ค. 2021, doi: 10.1038/s41598-021-84696-7.
- [30] smudge\_lord, “Smudge - Table Cat | If you’re going to post my face on the internet, at least tag me 🙄 #sassy #sassycat #catmeme #smudge #whitecat #memes #memelord #salad... | Instagram”. สืบค้น: 4 พฤษภาคม 2024. [ออนไลน์]. Available at: <https://www.instagram.com/p/Bx-UY1Qp1I9/?igsh=eG5yN205emhwcm1i>

- [31] M. Tan และ Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks". arXiv, 11 กันยายน 2020. สืบค้น: 29 มกราคม 2024. [ออนไลน์]. Available at: <http://arxiv.org/abs/1905.11946>



ประวัติผู้เขียน

