การจำแนกภาพขวดแบบเซ็ตเปิดด้วยโครงข่ายประสาทเทียมแบบคอนโวลูชัน

OPEN-SET BOTTLE CLASSIFICATION USING CONVOLUTION NEURAL NETWORK

ศุภณัฐ จินตวัฒน์สกุล

บัณฑิตวิทยาลัย มหาวิทยาลัยศรีนครินทรวิโรฒ

2562

การจำแนกภาพขวดแบบเซ็ตเปิดด้วยโครงข่ายประสาทเทียมแบบคอนโวลูชัน

ศุภณัฐ จินตวัฒน์สกุล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ
ปีการศึกษา 2562

OPEN-SET BOTTLE CLASSIFICATION USING CONVOLUTION NEURAL NETWORK

SUPANAT JINTAWATSAKOON

A Thesis Submitted in partial Fulfillment of Requirements

for MASTER OF SCIENCE (Information Technology)

Faculty of Science Srinakharinwirot University

2019

ปริญญานิพนธ์

เรื่อง

การจำแนกภาพขวดแบบเซ็ตเปิดด้วยโครงข่ายประสาทเทียมแบบคอนโวลูชัน

ของ

ศุภณัฐ จินตวัฒน์สกุล

ได้รับอนุมัติจากบัณฑิตวิทยาลัยให้นับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

ของมหาวิทยาลัยศรีนครินทรวิโรฒ

---------- คณบดีบัณฑิตวิทยาลัย

(รองศาสตราจารย์ นายแพทย์ฉัตรชัย เอกปัญญาสกุล)

คณะกรรมการสอบปากเปล่าปริญญานิพนธ์

..... ที่ปรึกษาหลัก ..... ประธาน

(อาจารย์ ดร.วีรยุทธ เจริญเรืองกิจ) (ดร.สุทธิพงศ์ ธัชยพงษ์)

..... กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร.นุวีย์ วิวัฒนวัฒนา)

ชื่อเรื่อง	การจำแนกภาพขวดแบบเซ็ตเปิดด้วยโครงข่ายประสาทเทียมแบบคอนโวลูชัน
ผู้วิจัย	ศุภณัฐ จินตวัฒน์สกุล
ปริญญา	วิทยาศาสตรมหาบัณฑิต
ปีการศึกษา	2562
อาจารย์ที่ปรึกษา	อาจารย์ ดร. วีรยุทธ เจริญเรืองกิจ

การจำแนกภาพ (Image Classification) ด้วยโครงข่ายประสาทเทียมแบบคอนโวลูชัน ได้รับความนิยมอย่างมากในช่วงไม่กี่ปีที่ผ่านมา เนื่องจากมีความแม่นยำสูงสำหรับปัญหาการจำแนกแบบเซ็ตปิด(close-set) วิทยานิพนธ์นี้เป็นศึกษาประสิทธิภาพในแง่ความแม่นยำเมื่อใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันกับปัญหาการจำแนกแบบเซ็ตเปิด (open-set) ซึ่งเป็นสภาพแวดล้อมที่คลาสเป้าหมาย(taget class) ไม่ปรากฏในขั้นตอนการฝึกฝน โดยสนใจแก้ปัญหาการจำแนกภาพขวดเครื่องดื่มออกเป็น 3 คลาสหลักและจำแนกเป็นคลาสที่ไม่รู้จัก ผ่านหลายการทดลองได้แก่ N-Binary, N+unknown, N+combination, 2 Layer Classification(OC-SVM and Multi-class model) and 2 Layer Classification(Isolation Forest and Multi-class model) ผลการทดลองแสดงให้เห็นว่า N+unknown มีความแม่นยำสูง 92% เหนือกว่าวิธีการอื่นๆ วิทยานิพนธ์กล่าวถึงผลลัพธ์ในแง่ของความถูกต้องและเวลาการฝึกฝนแบบจำลองสำหรับการวิจัยในอนาคตและการประยุกต์ใช้การจำแนกภาพ

คำสำคัญ : โครงข่ายประสาทเทียม, โครงข่ายประสาทเทียมแบบคอนโวลูชัน, การจำแนกภาพ, เซ็ตเปิด

Title	OPEN-SET BOTTLE CLASSIFICATION USING CONVOLUTION NEURAL NETWORK
Author	SUPANAT JINTAWATSAKOON
Degree	MASTER OF SCIENCE
Academic Year	2019
Thesis Advisor	Dr. Werayuth Charoenruengkit

The image classification technique based on the convolution neural network (CNN) has been gaining popularity in recent years due to its high accuracy for close-set classification problems. This thesis explores the accuracy performance of CNN on an open-set problem with a target class that is not specified in the training model. By focusing on the three classes of beverage bottle images, the models must be able to classify the input beverage image as one of the three classes and able to identify unknown or unseen types. The proposed methods to investigate several approaches based on the N-Binary, N+unknown, N+combination, two-layer Classification (OC-SVM and multi-class model) and two-layer classification (Isolation Forest and Multi-Class Model). The results showed that N+unknown approach achieved 92% overall accuracy and outperformed other models. This thesis discusses the results in terms of accuracy and training time for future research and image classification applications.

Keyword : Convolution neural network, open-set, Image classification, Neural network

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สำเร็จลุล่วงได้ด้วยดี ด้วยความอนุเคราะห์จาก อ. ดร. วีรยุทธ เจริญเรืองกิจ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ผู้ให้คำแนะนำและข้อเสนอแนะต่างๆ

ขอกราบขอบพระคุณคณะกรรมการสอบปริญญานิพนธ์สำหรับข้อเสนอแนะอันเป็นประโยชน์ต่อการปรับปรุงปริญญานิพนธ์

ขอกราบขอบพระคุณบัณฑิตวิทยาลัย มหาวิทยาลัยศรีนครินทรวิโรฒ สำหรับทุนสนับสนุนการวิจัยและนำเสนอผลงาน

ศุภณัฐ จินตวัฒน์สกุล

สารบัญ

หน้า

สารบัญตาราง

หน้า

สารบัญรูปภาพ

หน้า

บทที่ 1
บทนำ

1.1 ภูมิหลัง

Image Classification หรือการจำแนกข้อมูลภาพออกเป็นคลาสต่าง ๆ เป็นงานที่ได้รับการพัฒนามาอย่างต่อเนื่องและมีความสำคัญอย่างมาก เห็นได้จากการนำไปประยุกต์ใช้ในงานหลายๆแขนง ในปัจจุบันการจำแนกภาพ(Image Classification) โดยการใช้โครงข่ายแบบคอนโวลูชัน Convolution neural network (CNN) นั้นเป็นวิธีที่ได้รับความนิยมอย่างมาก ด้วยปัจจัยที่การเรียนรู้เชิงลึก(Deep learning) สามารถทำได้ง่ายขึ้นเนื่องด้วยจำนวนข้อมูลภาพที่มีเพิ่มขึ้นอย่างมหาศาลและการมีการประมวลผลแบบกลุ่มเมฆ (Cloud computing service) ที่ทำให้สามารถเข้าถึง Computation unit ได้ง่ายขึ้นในราคาที่ถูกลง และปัจจัยที่ CNN สามารถลดขั้นตอนที่มีความยุ่งยากในการจำแนกภาพโดยใช้การเรียนรู้ของเครื่อง (Machine learning based image classification) แบบดั้งเดิมที่ต้องสกัดฟีเจอร์(feature extraction) เพื่อหาเส้น (edge) สี (color) รูปทรง (shape) ฯลฯ ด้วยตัวเอง (hand-crafted features) เพื่อให้ได้ฟีเจอร์(feature) ที่เหมาะสมกับปัญหาและชุดข้อมูลนั้น ๆ ก่อนแล้วนำไปสร้างโมเดล ซึ่งต่างจาก CNN ที่จะใช้การเรียนรู้ตั้งแต่ต้นจนจบกระบวนการ (end-to-end learning) ตั้งแต่การหาฟีเจอร์ (feature) จนถึงการสร้างแบบจำลอง และปัจจัยที่สำคัญที่สุดคือเรื่องประสิทธิภาพที่ให้ความแม่นยำสูงกว่ามากเมื่อเทียบการจำแนกภาพโดยใช้การเรียนรู้ของเครื่องแบบดั้งเดิม

อย่างไรก็ตามการจำแนกภาพโดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน จะมีประสิทธิภาพดีและมีความน่าเชื่อถือภายใต้สภาพแวดล้อมแบบ Close-set เท่านั้น เนื่องจากแบบจำลองจะถูกฝึกขึ้นมาจากคลาสที่เราทราบทั้งหมดและในขั้นตอนการทำนายนั้นภาพที่ใช้ในการทำนาย จะต้องเป็นภาพที่อยู่ในคลาสที่ปรากฏอยู่ในขั้นตอนการเทรนโมเดลเท่านั้น อย่างไรก็ตามในการนำแบบจำลองไปใช้จริง มักจะอยู่ภายใต้สภาพแวดล้อมแบบ Open-Set ซึ่งเป็นสภาพแวดล้อมที่มีความเป็นไปได้เสมอที่ภาพที่ใช้ในการทำนายจะเป็นภาพที่ไม่ได้อยู่ในคลาสใด ๆ เลยในขั้นตอนการฝึกฝนแบบจำลอง

สำหรับปัญหาในงานวิจัยนี้มาจากระบบ Bottle Beverage Image Classification ซึ่งเป็นการจำแนกภาพขวดเครื่องดื่มออกเป็นชนิดต่าง ๆ โดยภาพเหล่านี้เป็นภาพที่ถูกถ่ายด้วยกล้องจากโทรศัพท์จากตู้แช่ของตัวแทนจำหน่ายทั่วประเทศ ในการฝึกแบบจำลองนั้นเป็นไปได้ยากมากที่สามารถสร้างแบบจำลองให้รู้จักภาพของขวดเครื่องดื่มทั้งหมดในท้องตลาด และมีโอกาสเสมอที่

จะพบเครื่องดื่มที่ไม่เคยปรากฏมาก่อน กล่าวคือหากใช้แบบจำลองกับภาพเหล่านี้ซึ่งเป็นภาพที่ไม่ปรากฏอยู่ในคลาสใด ๆ ในขั้นตอนการฝึกแบบจำลอง ผลการทำนายที่ได้ก็จะยังเป็นคลาสหนึ่งที่อยู่ในขั้นตอนฝึกแบบจำลองเสมอ ซึ่งเป็นคำตอบที่ไม่ถูกต้อง

โดยวิทยานิพนธ์นี้ได้นำเสนอวิธีการแก้ปัญหาการจำแนกภาพแบบ Open-set ของภาพขวดเครื่องดื่ม ซึ่งภาพที่ใช้ทำนายจะเป็นภาพที่อยู่ในคลาสที่ปรากฏในขั้นตอนการฝึกฝนแบบจำลอง หรืออาจไม่ได้อยู่ในคลาสใดเลยในขั้นตอนการฝึกฝนแบบจำลองเลยก็ได้

1.2 **ความมุ่งหมายของงานวิจัย**

ในการวิจัยครั้งนี้ผู้วิจัยได้ตั้งความมุ่งหมายไว้ดังนี้

1. สร้างแบบจำลองบนพื้นฐานโครงข่ายประสาทเทียมแบบคอนโวลูชัน สำหรับการจำแนกภาพขวดเครื่องดื่มแบบเซ็ตเปิดที่มีความแม่นยำสูง
2. สามารถจำแนกภาพขวดเครื่องดื่มออกเป็นประเภทที่กำหนด (Target class) และสามารถระบุว่าเป็นประเภทที่ไม่รู้จัก (Unknown) เมื่อภาพที่ใช้ทำนายไม่ได้อยู่ในกลุ่มของคลาสใดในขั้นตอนการฝึกฝนแบบจำลอง
3. เพื่อหาวิธีที่เหมาะสมในการจำแนกภาพที่ไม่รู้จัก (Unknown class) ซึ่งเป็นภาพที่ไม่อยู่ในคลาสใดในขั้นตอนการฝึกฝนแบบจำลอง
4. เพื่อทราบถึงหลักการทำงานของโครงข่ายประสาทเทียมแบบคอนโวลูชัน และนำไปปรับใช้ได้อย่างเหมาะสม

1.3 **ความสำคัญของการวิจัย**

งานวิจัยนี้เป็นการศึกษาและนำเสนอวิธีแก้ปัญหาของการจำแนกภาพขวดเครื่องดื่มภายใต้สภาพแวดล้อมแบบ Open-set โดยภาพเครื่องดื่มเหล่านี้จะถูกจำแนกออกเป็นสินค้าประเภทต่าง ๆ ผลการทำนายจะถูกแปรผลและนำไปใช้ในการวิเคราะห์ส่วนแบ่งทางการตลาดในแต่ละพื้นที่ แบบจำลองที่ใช้ในการทำนายจะถูกฝึกฝนขึ้นมาโดยอาศัยภาพสินค้าที่ถูกรวบรวมและทำการแยกเป็นคลาสต่าง ๆ ไว้ แต่เมื่อโมเดลถูกนำไปใช้งานจริงพบว่ามีโอกาสเสมอที่ข้อมูลภาพที่ส่งมาทำนายจะเป็นภาพที่ไม่ได้อยู่ในกลุ่มของคลาสใด ๆเลยที่ปรากฏในขั้นตอนการฝึกฝนแบบจำลองซึ่งจะส่งผลให้เกิดความผิดพลาดในการทำนาย ส่งผลให้การแปรผลผิดพลาดไปด้วยซึ่งจะเป็นปัญหา ในภาคธุรกิจที่ต้องนำผลที่ได้ไปใช้ต่อ ทั้งในการวางแผนการตลาด แผนการกระจายสินค้า ฯลฯ

1.4 ขอบเขตของการวิจัย

ผู้วิจัยได้กำหนดขอบเขตของงานวิจัยไว้ดังต่อไปนี้

1. พัฒนาแบบจำลองสำหรับการจำแนกภาพโดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน สำหรับการจำแนกภาพขวดเครื่องดื่มแบบเซ็ตเปิด โดยแบบจำลองที่ได้จะมาจากการทดลอง N-Binary Classifier, N+unknown Class classifier, N+combination Class Classsifier, 2 Layer Classification(OC-SVM and Multi-class model) and 2 Layer Classification(Isolation Forest and Multi-class model)
2. เปรียเทียบประสิทธิภาพแบบจำลองในแง่ของความแม่นยำและเวลาที่ใช้ในการฝึกฝนแบบจำลอง โดยใช้ชุดข้อมูลภาพเครื่องดื่ม ซึ่งประกอบด้วย
   - ชุดข้อมูลภาพถ่ายสินค้าเครื่องดื่ม 44 คลาส (44 ประเภทสินค้า) จำนวน 11,309 ภาพ ซึ่งเป็นภาพ
   - ชุดข้อมูลภาพสินค้าเครื่องดื่ม 1,592 ภาพ ซึ่งเป็นข้อมูลที่เป็นสินค้าที่มี SKU ไม่ซ้ำกับ SKU ในชุดข้อมูลแรกเลย

1.5 กรอบแนวคิดงานวิจัย

งานวิจัยนี้เป็นการศึกษาและนำเสนอวิธีแก้ปัญหาของการจำแนกภาพขวดเครื่องดื่ม โดยใช้เทคนิคโครงข่ายประสาทเทียมแบบคอนโวลูชัน เพื่อสร้างแบบจำลองในการจำแนกภาพขวดเครื่องดื่มภายใต้สภาพแวดล้อมแบบ Open-set ที่ภาพในการทำนายอาจจะเป็นภาพที่อยู่นอกกลุ่มของภาพที่ใช้ในการฝึกฝนแบบจำลอง

บทที่ 2
เอกสารและงานวิจัยที่เกี่ยวข้อง

ในการวิจัยครั้งนี้ผู้วิจัยได้ศึกษาเอกสารและงานวิจัยที่เกี่ยวข้อง และได้จำแนกออกตามหัวข้อดังต่อไปนี้

1. ทฤษฎีที่เกี่ยวข้องกับ Open-set classification
2. ทฤษฎีที่เกี่ยวข้องกับ Convolution Neural Network
3. ทฤษฎีที่เกี่ยวข้องกับ Residual Networks (ResNet)
4. ทฤษฎีที่เกี่ยวข้องกับ Generative Adversarial Networks (GANs)
5. ทฤษฎีที่เกี่ยวข้องกับ Variational Autoencoders (VAEs)
6. ทฤษฎีที่เกี่ยวข้องกับ One-Class Support Vector Machine
7. ทฤษฎีที่เกี่ยวข้องกับ Isolation Forest
8. ทฤษฎีที่เกี่ยวข้องกับ Principal Component Analysis

2.1 Open-set classification

โดยปกติแล้วในการสร้างแบบจำลองเพื่อการจำแนก (Classification Model) แบบจำลองจะถูกฝึกฝนมาจากข้อมูลที่ถูกแยกไว้เป็นคลาสต่าง ๆ (class label) จากนั้นเมื่อโมเดลถูกนำไปใช้งานจริง ข้อมูลที่ใช้ในการทำนายจะต้องเป็นข้อมูลที่เป็นคลาสใดคลาสหนึ่งที่อยู่ในขั้นตอนการปฝึกฝนแบบจำลองเท่านั้น ซึ่งถือเป็นสภาพแวดล้อมแบบ close-set ในทางตรงกันข้ามสภาพแวดล้อมแบบ open-set เมื่อนำแบบจำลองไปใช้งานจริงแล้ว ภาพที่ใช้ทำนายสามารถเป็นภาพที่อยู่นอกเหนือกลุ่มของภาพที่ใช้ในขั้นตอนฝึกฝนแบบจำลองได้นั้น (Scheirer, Rocha, Sapkota, & Boult, 2013) (Bendale & Boult, 2015)ภาพประกอบที่ 1 แสดงให้เห็น Decision Boundaries ของคลาส สุนัข, แมว และ ม้า ในการจำแนกภาพแบบ close-set หากนำแบบจำลองนี้ไปใช้จริงแล้วสมมติภาพที่ใช้ทำนายเป็นภาพของเสือ ผลการทำนายที่ได้ก็จะยังเป็น สุนัข, แมว และ ม้า เท่านั้น

ภาพประกอบ 1 decission boundaries ของ close-set classification

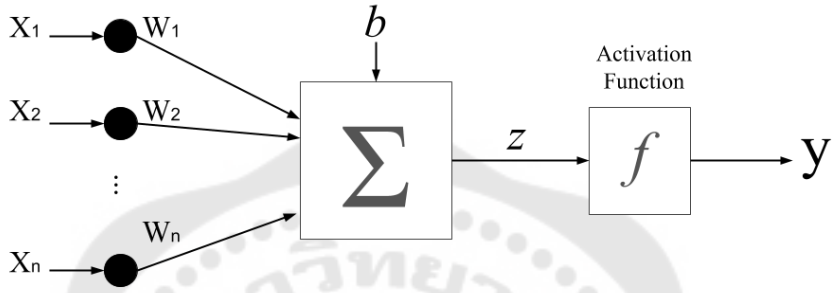
จากภาพประกอบที่ 2 ซึ่งแสดงให้เห็นถึง Open-set classification จากภาพจะเห็นว่านอกเหนือจากคลาสหลัก(known class) ซึ่งประกอบด้วย สุนัข, แมว และ ม้า แล้ว ยังมีคลาสอื่น ๆ (unknown class) ที่ยังไม่ทราบอยู่ด้วย

ภาพประกอบ 2 open-set classification

กล่าวโดยสรุปแล้วในปัญหา Open-set classification นั้น ในขั้นตอนการทดสอบแบบจำลอง (testing) จะมีทั้งคลาสที่รู้จัก(known class) ซึ่งเป็นคลาสที่ใช้ในขั้นตอนการฝึกฝนและคลาสที่เราไม่รู้จัก(unknown class) ซึ่งเป็น class ที่ไม่ปรากฏอยู่ในขั้นตอนการฝึกฝนมาก่อนเลย ซึ่งจะแตกต่างจากปัญหา Close-set classification ทั้งแบบ binary และ multi-class classification ตรงที่ในขั้นตอนการฝึกฝนและทดสอบจะใช้แค่ known class เพียงเท่านั้น

2.2 ทฤษฎีที่เกี่ยวข้องกับโครงข่ายประสาทเทียม

โครงข่ายประสาทเทียมเป็นแบบจำลองที่มีการทำงานเลียนแบบการทำงานของสมอง อาศัยการเรียนรู้โดยใช้หน่วยประมวลผลจำนวนมากรวมกันเป็นโครงข่ายซึ่งมีลักษณะเหมือนเซลล์สมองมนุษย์ โดยมีส่วนข้อมูลรับเข้าหรืออินพุต (Input) และส่งค่าถ่วงน้ำหนัก (Weight) ผ่านการประมวลผลแล้วจะได้ข้อมูลออกหรือเอาต์พุต (Output)



ภาพประกอบ 3 เพอร์เซ็ปตรอน

ในโครงข่ายประสาทเทียมนั้นจะประกอบขึ้นจากเพอร์เซ็ปตรอน(Perceptron) ซึ่งเป็นหน่วยพื้นฐานที่เล็กที่สุด ภาพประกอบที่ 3 แสดงโครงสร้างของเพอร์เซ็ปตรอน ซึ่งสามารถเขียนในรูปสมการได้ดังสมการที่ 1

$$y = f\left(b + \sum_{i=1}^{n} w_i x_i\right) \quad (1)$$

ข้อมูลรับเข้า (Input)

$$X = [x_1, x_2, \dots, x_n]$$

x_i คือ ข้อมูลรับเข้า (Input) แต่ละตัว

พารามิเตอร์ (Parameter)

$$W = [w_1, w_2, \dots, w_n], b$$

w_i คือ ค่าถ่วงน้ำหนัก (Weight)

ดังนั้น $x_i \cdot w_i$ คือ Input ที่ถูกถ่วงน้ำหนัก

b คือ ค่าความเอนเอียง (Bias)

เอาท์พุต (Output)

$$z = w_1 \cdot x_1 + w_2 \cdot x_2 + \cdots + w_n \cdot x_n + b$$
$$y = f(z)$$

z คือ ผลรวมของข้อมูลเข้าที่ถ่วงน้ำหนักแล้ว $\sum_{i=1}^{n} w_i x_i + b$

$f(.)$ คือ ฟังก์ชั่นกระตุ้น (Activation function)

y คือ ข้อมูลส่งออก (output) จากเพอร์เซ็ปตรอน

2.2.1 **ฟังก์ชั่นกระตุ้น** (Activation Function)

ฟังก์ชั่นกระตุ้น (Activation Function) เป็นฟังก์ชันที่เป็นตัวกำหนดว่าเพอ์เซปตรอนจะส่งข้อมูลออกในลักษณะใด ซึ่งฟังก์ชันกระตุ้นมีหลายรูปแบบ เช่น ฟังก์ชันซิกมอยด์ (Sigmoid function), ฟังก์ชันไฮเพอร์โบลิกแทนเจนต์ (Hyperbolic Tangent Function; Tanh), ฟังก์ชันเรคติไฟด์เชิงเส้น (Rectified Linear Unit Function; ReLU)

$$\sigma(z) = \frac{1}{1 + \exp(-z)} \tag{2}$$

$$tanh\,(z) = \frac{\exp(z) - \exp(-z)}{\exp(z) + \exp(-z)} \tag{3}$$

$$\mathrm{ReLU(z)} = \max(0, \mathrm{z}) \tag{4}$$

สมการที่ 2 คือฟังก์ชันซิกมอยด์จะให้ผลลัพธ์อยู่ในช่วง 0 ถึง 1

สมการที่ 3 ฟังก์ชันไฮเพอร์โบลิกแทนเจนต์จะให้ผลลัพธ์อยู่ในช่วง -1 ถึง 1

และ สมการที่ 4 ฟังก์ชันเรคติไฟด์เชิงเส้นจะให้ผลลัพธ์เป็น 0 หรือ z

2.2.2 การฝึกฝนแบบจำลองโครงข่ายประสาทเทียม

เป้ามายของการงฝึกฝนแบบจำลองก็เพื่อให้ได้แบบจำลองที่มีความแม่นยำที่สุด ในการวัดว่าแบบจำลองสามารถทำนายผลได้ดีแค่ไหนต้องมีการนิยามฟังก์ชันต้นทุน (loss function) ขึ้นมาเมื่อจะหาความแตกต่างระหว่างค่าจริง y เปรียบเทียบกับค่าที่ทำนาย $\hat{y}$

ในขั้นตอนการฝึกโครงข่ายประสาทเทียม ค่าถ่วงน้ำหนัก $(weight)$ และค่าความเอนเอียง $(bias)$ จะถูกปรับเปลี่ยนเพื่อให้ได้ฟังก์ชันต้นทุนน้อยลงที่สุด ดังที่แสดงในสมการที่ 5

$$\mathbf{W}^{*}=\arg\min_{\mathbf{w}}\frac{1}{N}\sum_{i=1}^{N}L\left(y^{(i)},\hat{\mathbf{Y}}^{(i)}\right) \tag{5}$$

จากสมการ

y คือ ค่าจริง

$\hat{y}$ คือ คือค่าที่ทำนาย

$L(y,\hat{y})$ คือ ฟังก์ชันต้นทุน (loss function)

2.2.3 ฟังก์ชันสูญเสีย (Loss Function)

ในการฝึกฝนแบบจำลองค่าจาก loss function เป็นตัวแปรที่ใช้ในการปรับค่าถ่วงน้ำหนักเพื่อหาค่าถ่วงน้ำหนักที่ดีที่สุดที่ทำให้ค่าของฟังก์ชันต้นทุนน้อยที่สุด ฟังก์ชันต้นทุนมีอยู่หลายฟังก์ชันโดยต้องเลือกใช้ให้เหมาะสมปัญหา เช่น Cross-entropy, MSE, Kullback Leibler ฯลฯ

ครอส-เอนโทรปี ล็อซ (Cross-entropy loss)

ครอส-เอนโทรปี ล็อซ (Cross-entropy loss) เป็นฟังก์ชันใช้บ่งชี้ประสิทธิภาพของแบบจำลองว่าสามารถจำแนกได้ผิดพลาดแค่ไหน ในสมการที่ 6 แสดงการหา Cross-entropy loss สำหรับการจำแนกเป็นหลายประเภท(Multi-class classification) เพื่อบอกประสิทธิภาพในการจำแนกซึ่งผลลัพธ์คือค่าความน่าจะเป็นระหว่าง 0 ถึง 1

$$L\left(y^{(i)},\hat{\mathbf{Y}}^{(i)}\right)=-\sum_{k=1}^{K}1\left(y^{(i)}=k\right)\log\left(\hat{y}_{k}^{(\mathrm{i})}\right) \tag{6}$$

2.2.4 **การหาค่าที่ดีที่สุด** (Optimization)

การหาค่าที่ดีที่สุด (Optimization) มีเป้าหมายเพื่อหาพารามิเตอร์หรือค่าถ่วงน้ำหนักที่ดีที่สุดเพื่อปรับปรุงแบบจำลองให้มีประสิทธิภาพดีขึ้น โดยวิธีการที่นิยมใช้กันก็คือการเคลื่อนลงตามความชัน (Gradient Descent Algorithm)

จากสมการที่ 7 เกรเดียนต์ $\nabla g(w)$ คือ เวคเตอร์ของอนุพันธ์ย่อย(Partial derivatives) ทั้งหมด ซึ่งจะบอกถึงความเปลี่ยนแปลงของ $g(w)$ เมื่อ w เปลี่ยน

$$\nabla g(w) = \left(\frac{\partial g}{\partial w_1}(w), \ldots, \frac{\partial g}{\partial w_K}(w) \right) \tag{7}$$

$$\mathbf{W} \leftarrow \mathbf{W} - \alpha * \frac{\partial L}{\partial \mathbf{W}} \tag{8}$$

ค่าถ่วงน้ำหนักใหม่จะหาได้จากสมการที่ 8 โดยที่

α คือ อัตราการเรียนรู้ (learning rate)

$\frac{\partial L}{\partial W}$ คือ เกรเดียนต์ของฟังก์ชันต้นทุนเทียบกับค่าถ่วงน้ำหนัก

Full-Batch Gradient Descent

ฟูลแบตช์เกรเดียนต์เดสเซนท์ (Full-Batch Gradient Descent) จะใช้ข้อมูลทุกจุดในการคำนวณเกรเดียนต์ ดังที่แสดงในสมการที่ 9 แสดงให้เห็นการคำนวณเกรเดียนต์จากคือข้อมูลตัวที่ 1 ถึง N

$$\mathbf{W} \leftarrow \mathbf{W} - \alpha^* \sum_{i=1}^{N} \frac{\partial L\left(y_i, \hat{\mathbf{Y}}^{(i)}\right)}{\partial \mathbf{W}} \tag{9}$$

Stochastic Gradient Descent

สโตแคสติกเกรเดียนต์เดสเซนท์ (Stochastic Gradient Descent; SGD) จะใช้ข้อมูลเพียงจุดเดียวในการคำนวณเกรเดียนต์แทนที่การใช้ทุกจุดแบบดังที่แสดงในสมการที่ 10

$$\mathbf{W} \leftarrow \mathbf{W} - \alpha * \frac{\partial L\left(y_i, \hat{\mathbf{Y}}^{(i)}\right)}{\partial \mathbf{W}} \tag{10}$$

Mini-Batch Gradient Descent

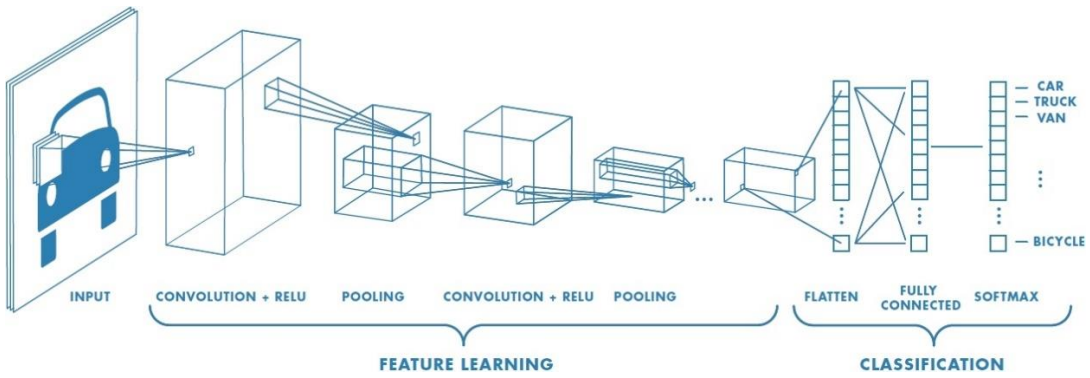
มินิแบตช์เกรเดียนต์เดสเซนท์ (Mini-Batch Gradient Descent) จะใช้ข้อมูลบางส่วนในการคำนวณเกรเดียนต์ จากสมการที่ 11 แสดงให้เห็นการคำวณเกรเดียนต์จากคือข้อมูลตัวที่ k ถึง $k+m$

$$\mathbf{W} \leftarrow \mathbf{W} - \alpha * \sum_{i=k}^{k+M} \frac{\partial L\left(y_i, \hat{\mathbf{Y}}^{(i)}\right)}{\partial \mathbf{W}} \tag{11}$$

2.3 ทฤษฎีที่เกี่ยวข้องกับโครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolution Neural Network)

โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolution Neural Network; CNN) เป็นโครงข่ายประสาทเทียมเชิงลึก (Deep Neural Network) ที่ประกอบด้วยเซ็ตของฟิลเตอร์ (Filter) ซึ่งฟิลเตอร์เหล่านี้ในจะใช้เพื่อทำคอนโวลูชันกับอินพุต และในระหว่างขั้นตอนการฝึกฝน(train neural network) ฟิลเตอร์เหล่านี้ก็จะถูกปรับค่าเพื่อให้ได้ฟิลเตอร์ที่เหมาะสมที่สุด โดยมีจุดประสงค์คือการสกัดฟีเจอร์จากภาพ (Feature Extraction)

โครงข่ายประสาทเทียมแบบคอนโวลูชันได้รับความนิยมอย่างมากหลังจาก Krizhevsky ได้นำเสนอ AlexNet (Krizhevsky, Sutskever, & Hinton, 2012) ซึ่งสามารถเอาชนะในการแข่งขัน ImageNet Large Scale Visual Recognition Challenge (ILSVRC) โดยสามารถทำคะแนนได้สูงกว่าเทคนิคแบบดั้งเดิม และ CNN Based Architecture ยังถูกพัฒนาอย่างต่อเนื่องเห็นได้จาก top 5 error rate ของชุดข้อมูล ImageNet ลดจาก ~25% จนเหลือ ~2.25% ภายใน 5 ปีหลังจาก AlexNet ถูกนำเสนอ ซึ่ง error rate ที่ ~2.5 % นั้นถือว่าน้อยกว่า human error อีกด้วย ปัจจุบัน CNN ถูกประยุกต์ไปใช้ในคอมพิวเตอร์วิทัศน์ (Computer vision) ไม่ว่าจะเป็นการทำ Image Classification, Object Detection, Image and Video Recognition ฯลฯ



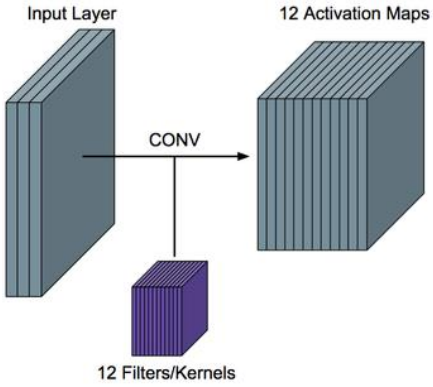
ภาพประกอบ 4 แสดงโครงสร้าง Convolution Neural Network

ที่มา: Convolutional Neural Network. (n.d.). Retrieved December 1, 2019, from https://www.mathworks.com/solutions/deep-learning/convolutional-neural-network.html

จากภาพประกอบที่ 1 ได้แสดงโครงสร้างของโครงข่ายประสาทเทียมแบบคอนโวลูชัน โดยเริ่มต้นที่ ชั้นข้อมูลเข้า (Input Layer) จะรับข้อมูลเข้าเป็นรูปภาพเพื่อเข้าสู่การดำเนินการที่ชั้นต่าง ๆ โดยสามารถจัดประเภทของชั้นต่างๆ ได้เป็น ชั้นคอนโวลูชัน (Convolution Layer), ชั้นพูลลิ่ง (Pooling Layer), ชั้นเชื่อมโยงโดยสมบูรณ์ (Fully-Connected Layer)

2.3.1 **ชั้นคอนโวลูชัน** (Convolution Layer)

ชั้นคอนโวลูชัน(Convolution Layer) มีไว้เพื่อทำการคอนโวลูชันระหว่างอินพุต(Input) กับฟิลเตอร์/เคอร์เนล(Filter/Kernel) ผลลัพธ์ที่ได้คือฟีเจอร์แมพ/แอคทิเวชันแมพ (Feature Map/ Activation Map) ดังที่แสดงในภาพประกอบที่ 5 โดยที่ฟิลเตอร์จะมีการสุ่มและปรับเปลี่ยนในขั้นตอนการเรียนรู้



ภาพประกอบ 5 Convolution Layer

ที่มา: Khan, S., Rahmani, H., Shah, S. A. A., Bennamoun, M., Medioni, G., & Dickinson, S. (2018). A Guide to Convolutional Neural Networks for Computer Vision. Retrieved from https://ieeexplore.ieee.org/document/8295029

ขนาดอินพุต $(W_{\text{input}} \times H_{\text{input}} \times D_{\text{input}})$ จะถูกกำหนดโดยความกว้าง, ความสูงและความลึก โดยปกติแล้วขนาดความกว้างและความสูงจะมีขนาดเท่ากัน ที่ชั้นอินพุต (Input Layer) ความลึกของอินพุตจะมาจากจำนวน Color depth เช่น หากเป็นภาพสี RGB ความลึก (D_{input}) จะเท่ากับ 3 และในชั้นที่ลึกขึ้น (Hidden Layer) ความลึก (D_{input}) จะเท่ากับจำนวนฟิลเตอร์ที่ใช้ในชั้นก่อนหน้า

ที่ชั้นคอนโวลูชัน(Convolution Layer)จะมีพารามิเตอร์สำหรับฟิลเตอร์ดังต่อไปนี้

จำนวนฟิลเตอร์ (K) จำนวนฟิลเตอร์ที่ใช้ จะส่งผลต่อความลึกของเอาต์พุต (Feature Map/Activation Map) ดังที่แสดงในภาพประกอบที่ 5 ความลึกของเอาต์พุต (12 Activation map)จะเท่ากับจำนวนฟิลเตอร์ที่ใช้ (12 Filter)

ขนาดฟิลเตอร์ $(F \times F)$ เป็นการกำหนดขนาดความกว้างและความสูงของฟิลเตอร์ เช่น ฟิลเตอร์ขนาด 3x3, 5x5, 7x7, 11x11

Stride (S) ฟิลเตอร์จะมีการขยับเพื่อดำเนินการคอนโวลูชันกับอินพุต ดังนั้นจะต้องมีการกำหนดว่าขนาดของ Stride เพื่อกำหนดการขยับของฟิลเตอร์

Zero-Padding (P) คือการกำหนดว่าให้มีการเติม 0 เข้าไปที่บริเวณขอบของอินพุตหรือไม่ ปกติแล้วถ้าไม่มีการทำ padding ผลที่ได้จากการคอนโวลูชันจะทำให้ได้ฟีเจอร์แมพที่มีขนาดเล็กลงกว่าขนาดอินพุต จึงมีทางเลือกในการทำ padding ในกรณีที่ต้องการรักษาขนาดของ

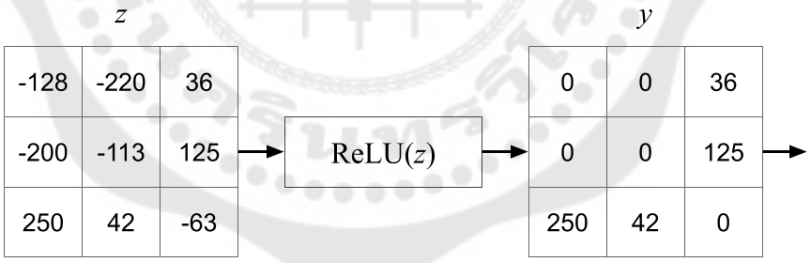
ฟีเจอร์แมพไม่ให้หดลงและจำนวน padding จะหาได้จากสมการที่ 12 เมื่อต้องการรักษาขนาดของเอาต์พุตให้เท่ากับขนาดของอินพุต

$$p = \frac{f-1}{2} \tag{12}$$

เอาต์พุต(Output) ที่ได้จากการทำคอนโวลูชันจะเรียกว่า ฟีเจอร์แมพหรือแอคทิเวชันแมพ (Feature Map; Activation Map) ซึ่งจะมีขนาดเป็น $(W_{\text{output}} \times H_{\text{output}} \times D_{\text{output}})$ โดยที่

$$\begin{aligned} \mathrm{W_{output}} &= \Big(\big(\mathrm{W_{input}} - \mathrm{F} + 2\mathrm{P}\big)/\mathrm{S}\Big) + 1 \\ H_{\text{output}} &= \Big(\big(H_{\text{input}} - F + 2P\big)/S\Big) + 1 \\ D_{\text{output}} &= K \end{aligned} \tag{13}$$

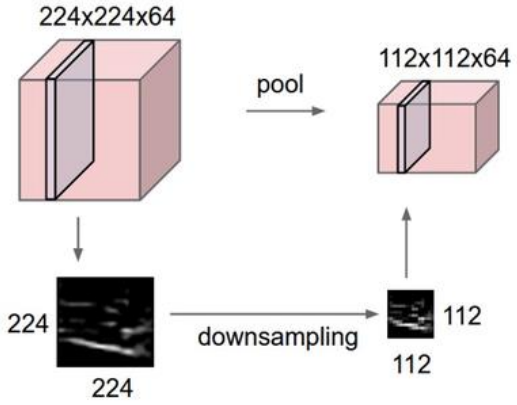
ผลที่ได้จากการทำคอนโวลูชันจะต้องผ่านฟังก์ชันกระตุ้น (Activation Function) เพื่อทำการปรับค่าที่ได้จากการทำคอนโวลูชันและส่งต่อไปเป็นอินพุตในชั้นถัดไป โดยฟังก์ชันกระตุ้นที่นิยมใช้ใน CNN มีหลายฟังก์ชัน เช่น ฟังก์ชันเรคติไฟด์เชิงเส้น (ReLU) ภาพประกอบที่ 6 แสดงค่าที่ได้หลังผ่านฟังก์ชันฟังก์ชันเรคติไฟด์เชิงเส้น



ภาพประกอบ 6 ผลลัพธ์ที่ได้จากฟังก์ชันเรคติไฟด์เชิงเส้น (ReLU)

2.3.2 **ชั้นพูลลิ่ง** (Pooling Layer)

ชั้นพูลลิ่ง (Pooling Layer) มักวางต่อจากชั้นคอนโวลูชัน จะถูกใช้เพื่อลดขนาดของฟีเจอร์แมพ (Feature Map, Activation Map) กล่าวคือฟีเจอร์แมพที่ได้จากชั้นคอนโวลูชัน จะถูกส่งไปยังชั้นพูลลิ่งเพื่อทำการลดขนาดความกว้างและความสูง ทำให้ข้อมูลที่จะส่งไปยังชั้นถัดไปทำให้เป็นการมีขนาดเล็กลง ซึ่งจะเป็นการลดพารามิเตอร์ไปด้วย



ภาพประกอบ 7 Pooling Operation

ที่มา: CS231n Convolutional Neural Networks for Visual Recognition. (n.d.). Retrieved, from http://cs231n.github.io/convolutional-networks/#architectures

ในชั้นพูลลิ่งจะรับข้อมูลเข้าซึ่งจะมีขนาดเป็น $(W_{\text{input}} \times H_{\text{input}} \times D_{\text{input}})$ และจะมีพารามิเตอร์ที่ต้องกำหนดดังต่อไปนี้

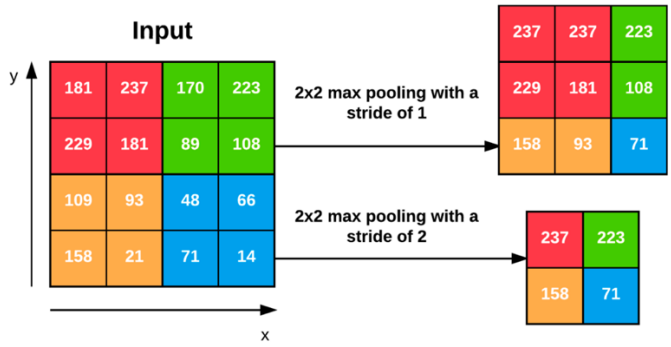
Pool size ขนาดของพูล $(F \times F)$

Stride (S) เพื่อกำหนดการขยับของ Pool

เอาต์พุตจากชั้นพูลลิ่งจะมีขนาดเป็น $(W_{\text{output}} \times H_{\text{output}} \times D_{\text{output}})$ โดยที่

$$W_{\text{output}} = \left((W_{\text{input}} - F)/S\right) + 1$$
$$H_{\text{output}} = \left((H_{\text{input}} - F)/S\right) + 1 \quad (14)$$
$$D_{\text{output}} = D_{\text{input}}$$

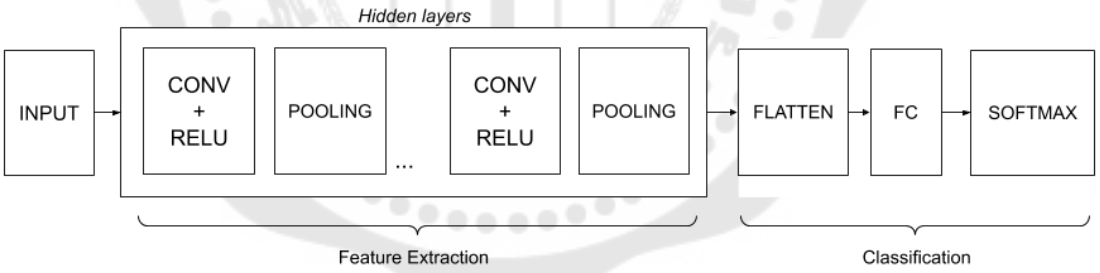
การดำเนินการบนชั้นพูลลิ่งสามารถทำได้ 2 วิธีคือพูลค่ามากที่สุด(Max Pooling) และพูลค่าเฉลี่ย (Average Pooling)



ภาพประกอบ 8 การทำพูลลิ่ง

ที่มา: CS231n Convolutional Neural Networks for Visual Recognition. (n.d.). Retrieved, from http://cs231n.github.io/convolutional-networks/#architectures

จุดประสงค์ของชั้นคอนโวลูชันและชั้นพูลลิ่งนั้นก็เพื่อทำการสกัดฟีเจอร์จากภาพ โดยในชั้นช่อนแรกๆ จะได้ฟีเจอร์ระดับล่างเช่น เส้น ในชั้นที่ลึกขึ้นลงไปก็จะเป็นการสกัดฟีเจอร์ที่ซับซ้อนขึ้น ซึ่งจำนวนชั้นคอนโวลูชันและชั้นพูลลิ่งที่เพิ่มขึ้นก็จะส่งผลให้ความลึกของโครงข่ายที่เพิ่มขึ้น



ภาพประกอบ 9 โครงสร้างอย่างง่ายของ CNN

2.3.3 **ชั้นเชื่อมโยงสมบูรณ์** Fully Connected Layer

ชั้นเชื่อมโยงสมบูรณ์ (Fully Connected Layer) จะอยู่ในส่วนท้ายของเครือข่ายซึ่งอาจจะมี 1-2 ชั้นก่อนส่งข้อมูลเข้า Softmax Function โดยให้ทุก Neuron ในชั้นนี้จะต่อกับฟีเจอร์แมพก่อนหน้าแบบ Fully-Connected โดยจะต้องทำการ (Flatten) ให้เป็นเวคเตอร์หนึ่งมิติก่อน

2.3.4 **ฟังก์ชันซอฟต์แม็กซ์** (Softmax function)

ฟังก์ชันซอฟต์แม็กซ์ (Softmax function) จะถูกวางไว้ที่ชั้นสุดท้ายเพื่อหาความน่าจะเป็นของแต่ละคลาส คลาสที่มีความน่าจะเป็นสูงสุดคือผลการทำนาย จากสมการที่ 15 คะแนนความน่าจะเป็นของแต่ละคลาส. โดย z_i คือ logit score ในเวคเตอร์ z ความน่าจะเป็นของแต่ละคลาสหาได้จากเอกซ์โพเนนเชียลของ logit หารด้วยผลรวมของเอกซ์โพเนนเชียลของ logit ทุกตัว ผลลัพธ์ที่ได้คือคะแนนความน่าจะเป็นของคลาส ซึ่งผลรวมของความน่าจะเป็นของทุกคลาสจะมีค่าเท่ากับ 1

$$softmax(z_i) = \frac{\exp(z_i)}{\sum_{k=1}^{K} \exp(z_k)} \quad (15)$$

2.4 Residual Networks (ResNet)

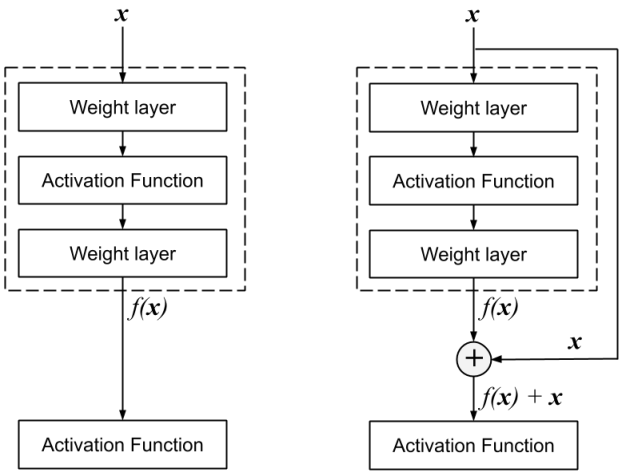
สถาปัตยกรรมเป็นตัวกำหนดโครงสร้างในแต่ละชั้นจะมีการต่อกันอย่างไรหรือใช้ฟังก์ชันกระตุ้นแบบไหน ขนาดฟิลเตอร์เป็นเท่าไหร่ โดยมีสถาปัตยกรรมหลายๆตัวที่มชื่อเสียงและทำประสิทธิภาพได้ดีที่สุด ณ ช่วงเวลานั้น (state of the art) เช่น AlexNet, GoogLeNet, VGG เป็นต้น สถาปัตยกรรมใหม่ๆที่ถูกนำเสนอมีประสิทธิภาพดีขึ้นจะมาพร้อมกับความลึกที่มากขึ้น(เพิ่มชั้นซ่อนมากขึ้น) อย่างไรก็ตามมีงานวิจัยที่แสดงให้เห็นว่าเมื่อความลึกเพิ่มขึ้นถึงจุดหนึ่งประสิทธิภาพกลับไม่ได้ดีขึ้น หนึ่งในสาเหตุสำคัญก็คือการ Optimization เช่น ปัญหา gradient vanishing ส่งผลให้ค่าเกรเดียนต์สูญหาย เป็นต้น

Residual Networks (ResNet) (He, Zhang, Ren, & Sun, 2015) เป็นสถาปัตยกรรมหนึ่งที่มีการออกแบบที่สำคัญที่ทำให้ได้ประสิทธิภาพดีถึงแม้ว่าจะมีความลึกมากก็ตาม โดยได้นำเสนอเทคนิคที่เรียกว่า skip connection ภาพประกอบที่ 11 แสดงโครงสร้างของ ResNet ซึ่งเป็นโครงสร้างที่ประกอบขึ้นจาก Residual block ต่อกัน ResNet สามารถให้ประสิทธิภาพดีถึงแม้จะมีความลึกมากขึ้นก็ตาม โดยโครงสร้างที่แสดงในภาพคือ ResNet18 ซึ่งมีความลึก 18 ชั้น และสามารถสร้างให้ลึกขึ้นได้ เช่น ResNet50, ResNet152 ที่มีความลึกถึง 50 และ 152 ชั้นตามลำดับ

2.4.1 Residual Blocks

Residual Blocks ใช้วิธีการทำ Skip Connection เป็นทางลัดเพื่อใช้ข้ามชั้นคอนโวลูชันในบางชั้น เนื่องจากการคอนโวลูชันในบางชั้นอาจไม่มีความจำเป็นค่าถ่วงน้ำหนักสามารถให้เป็น 0 ได้ โดยที่ข้อมูลเข้าก็จะส่งผ่านไปชั้นถัดไปได้โดยไม่มีการสูญหาย

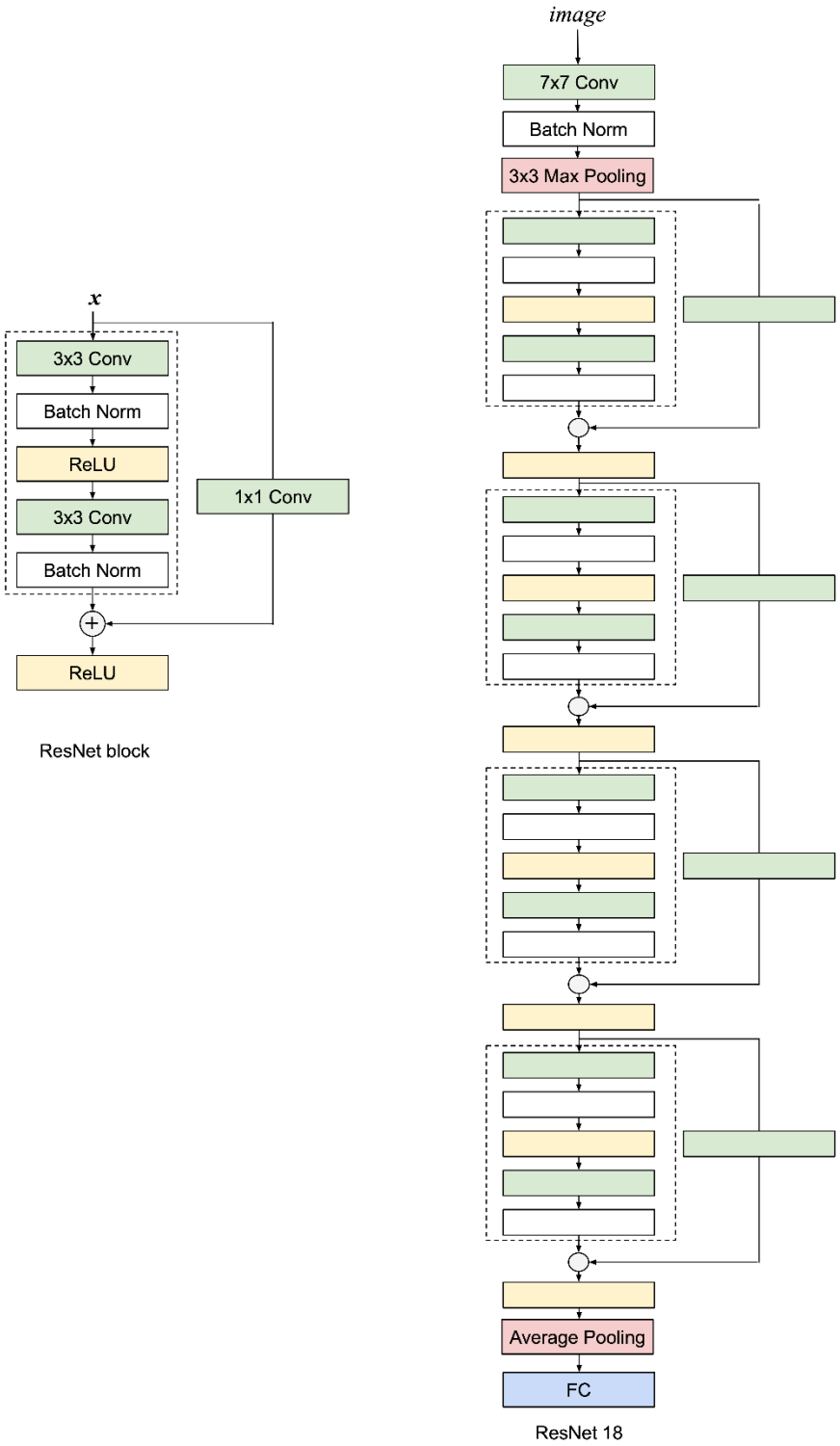
ภาพประกอบที่ 10 แสดงโครงสร้างของ Residual Block (รูปขวา) เทียบกับแบบปกติที่ไม่มี Skip Connection (รูปซ้าย) เมื่อ X คือข้อมูลเข้า ข้อมูลเข้าจะผ่านการถ่วงน้ำหนัก $f(x)$ แล้วส่งต่อไปยังฟังก์ชันกระตุ้น แต่ในกรณีของ Residual Block ข้อมูลเข้าถ่วงน้ำหนักแล้วจะนำมารวมกับข้อมูลเข้าอีกที $f(x) + x$ ก่อนส่งไปฟังก์ชันกระตุ้น



ภาพประกอบ 10 Residual block

2.4.2 ResNet Model

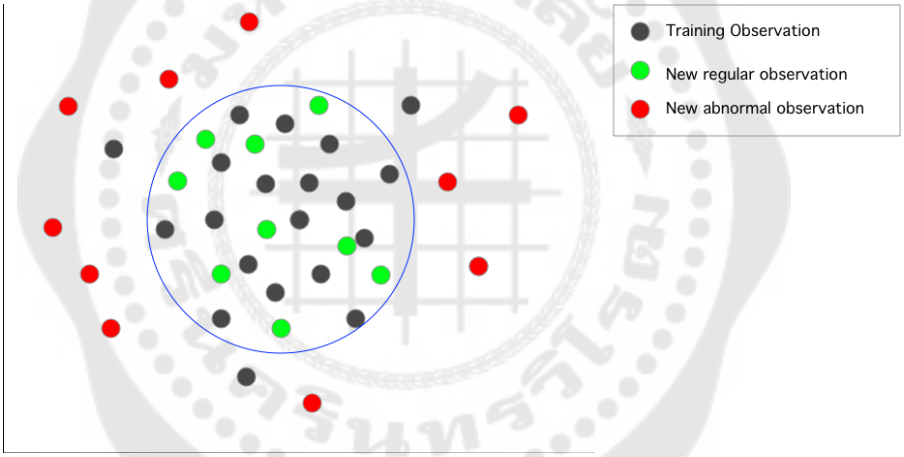
ResNet จะมีโครงสร้างที่ประกอบขึ้นจาก residual block ต่อกัน ดังที่แสดงในภาพประกอบที 11 ที่ชั้นแรกของโครงข่ายจะใช้ฟิลเตอร์ขนาด 7x7 และให้เอาท์พุตเป็น 64 และกำหนด stride เป็น 2 ตามด้วยการทำ Batch Normalization และการทำพูลลิ่งค่าสูงสุด (Max pooling) ในชั้นต่อๆไปจะประกอบขึ้นจาก Residual Blocks ซึ่งมีโครงสร้างข้างในประกอบด้วยประกอบขึ้นด้วย ชั้นคอนโวลูชันที่ใช้ฟิลเตอร์ขนาด 3x3, Batch Normalization และ ฟังก์ชันกระตุ้นเรคติไฟด์เชิงเส้น (ReLU) ดังที่แสดงในภาพประกอบที 11 (ซ้าย) และในส่วนท้ายของโครงข่ายจะทำพูลลิ่งแบบค่าเฉลี่ย (Average pooling) ก่อนจะเข้าสู่ขั้นตอนการจำแนก (classification) ต่อไป



ภาพประกอบ 11 โครงสร้างของ ResNet

2.4 One-Class Support Vector Machine

One-Class Support Vector Machine (OC-SVM) (Schölkopf, Williamson, Smola, Shawe-Taylor, & Platt, 2000) ถูกพัฒนามาจาก Support Vector Machine เพื่อใช้ในงาน Anomaly Detection โดยมีแนวคิดให้ทำการฝึกแบบจำลองด้วยชุดข้อมูลที่มีเพียงข้อมูลปกติ (normal data) และจะนำไปใช้ตรวจจับข้อมูลผิดปกติ (Anomaly) เมื่อมีข้อมูลใหม่เข้ามา (new observation) ภาพประกอบที่ 14 แสดงให้เห็น Decision boundary ของ OC-SVM เมื่อจุดสีเทาคือข้อมูลที่ใช้ในการฝึกแบบจำลอง (training observation) วงกลมคือ decision boundary ที่จะเป็นตัวแบ่งระหว่างข้อมูลปกติกับข้อมูลผิดปกติ หากข้อมูลใหม่ที่โมเดลยังไม่เคยเห็นมาก่อนเข้ามาแล้วแล้วอยู่ภายใน decision boundary ก็จะถูกจำแนกเป็นข้อมูลปกติ(จุดสีเขียว) ในทางตรงกันข้ามหากข้อมูลใหม่ตกอยู่นอก decision boundary แสดงว่าเป็นข้อมูลที่ผิดปกติ(จุดสีแดง)



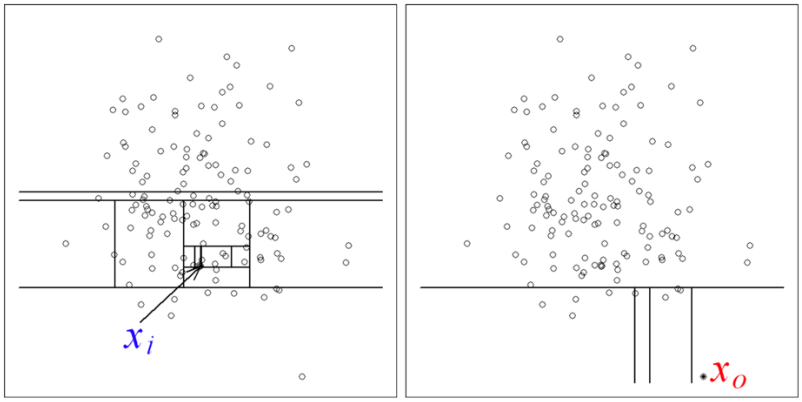
ภาพประกอบ 12 Decision boundary

2.5 Isolation Forest

Isolation Forest (F. T. Liu, K. M. Ting, & Z.-H. Zhou, 2008) เป็นเทคนิคที่ใช้ในการตรวจจับสิ่งผิดปกติ (Anomaly Detection) มีโครงสร้างที่เป็น Tree ที่ทำหน้าที่แบ่งข้อมูล (Partitioning) จนข้อมูลแต่ละตัว (Instance) แยกออกจากข้อมูลที่เหลือ (Isolated)

การทำงานของของ Isolation Forest จะเริ่มจากการแบ่งข้อมูล(Partitioning) ซึ่งการ partitioning เริ่มจากการเลือกฟีเจอร์ (Feature) มาแล้วทำการสุ่มค่าระหว่างค่าน้อยที่สุดและมากที่สุดของฟีเจอร์ที่เลือกมาและค่าที่สุ่มมาได้นี้จะใช้ในการแบ่งข้อมูล โดยกระบวนการ Partitioning จะถูกทำซ้ำจนกระทั่งแต่ละ Instance แยกออกจาก Instance ที่เหลือ การ Partitioning ทำให้ได้โครงสร้างแบบ Tree โดย Instance ที่ผิดปกติจะมี path จาก root node ที่สั้นกว่า Instance ปกติ

จากภาพประกอบที่ 15 เมื่อ Xi เป็นข้อมูลปกติ (normal point) ซึ่งจะต้อง Partitioning ถึง 12 ครั้งถึงจะ Isolated ในขณะที่ $X0$ ซึ่งเป็น anomaly จะ Partitioning เพียง 4 ครั้ง

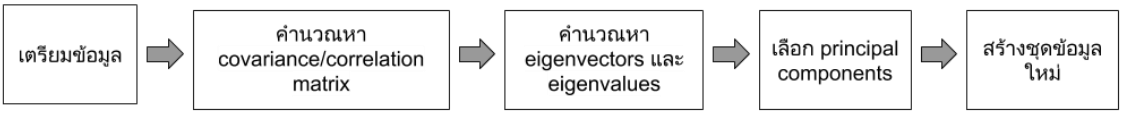


ภาพประกอบ 13 Isolation Forest Partitioning

2.6 การวิเคราะห์องค์ประกอบหลัก (Principal Component Analysis)

การวิเคราะห์องค์ประกอบหลัก (Principal Component Analysis; PCA) เป็นวิธีที่ใช้ในการลดจำนวนมิติของข้อมูล (Dimensionality Reduction) ให้มีขนาดที่เหมาะสมที่ยังสามารถอธิบายข้อมูลนั้นได้ สมมติให้ข้อมูลมีตัวแปรจำนวนมากเช่นมีตัวแปรจำนวน n ตัว จากนั้น PCA จะทำการสร้างเซ็ตของตัวแปรใหม่ซึ่งเป็นฟังก์ชันของตัวแปรเดิม n ตัว โดยจะดึงรายละเอียดหรือความแปรปรวนจากตัวแปรเดิมมาไว้ในตัวแปรใหม่ให้ได้มากที่สุด

PCA ได้ถูกนำมาใช้ในการคัดเลือกฟีเจอร์ (Feature Selection) เพื่อให้ได้ฟีเจอร์ใหม่ไปใช้ในขั้นตอนการเทรนโมเดลด้วย learning algorithm ใด ๆ แทนที่การใช้ฟีเจอร์ทั้งหมด โดย PCA จะแปลงข้อมูลที่มีฟีเจอร์จำนวนมากให้มีจำนวนฟีเจอร์น้อยลง ส่งผลให้ใช้เวลาในการประมวลผลและวิเคราะห์น้อยลง



ภาพประกอบ 14 ขั้นตอนการลดมิติข้อมูลด้วย PCA

สำหรับขั้นตอนการทำ PCA จะประกอบด้วย 5 ขั้นตอนหลัก ดังที่แสดงในภาพประกอบที่ 16 ในขั้นตอนแรกคือการเตรียมข้อมูลโดยการทำนอมัลไลซ์ หากความแปรปรวนของตัวแปรในข้อมูลแตกต่างกันอย่างมีนัยสำคัญ สามารถทำได้โดยให้แต่ละตัวแปรหารด้วยค่าเบี่ยงเบนมาตรฐาน จากนั้นในขั้นตอนที่ 2 จะเป็นการคำนวณหา covariance/correlation matrix เพื่อนำไปคำนวณหา eigenvectors และ eigenvalues ในขั้นตอนที่ 3 โดยที่ eigenvalues จะเป็นค่าที่บอกถึงความความสำคัญของแต่ละ eigenvectors (ค่า eigenvectors จะถูกเรียงจากมากไปน้อยตาม eigenvalues) ในขั้นตอนที่ 4 จะเป็นการเลือก eigenvectors ที่มีค่า eigenvalues สูงตามลำดับ และ eigenvectors ที่เลือกทั้งหมดจะนำไปสร้างเป็นชุดข้อมูลใหม่

2.7 งานวิจัยที่เกี่ยวข้อง

งานวิจัยที่เกี่ยวข้องที่ผู้วิจัยได้ศึกษาสามารถแบ่งออกได้ 3 กลุ่มได้แก่ งานวิจัยลุ่ม CNN-Based Image Classification จะเป็นการศึกษาสถาปัตยกรรมและปัญหาของโครงข่ายประสาทเทียมแบบคอนโวลูชัน, งานวิจัยที่เกี่ยวข้องกลุ่ม Open-set classification ทั้งแบบ traditional machine learning based และ deep learning based, งานวิจัยที่เกี่ยวข้องกลุ่ม Anomaly detection โดยหากพิจารณางานวิจัยที่เกี่ยวข้องทั้งหมดในกลุ่ม Open-set classification และ Anomaly detection จากข้อมูลที่ใช้ในการฝึกฝนแบบจำลอง จะสามารถแบ่งงานวิจัยได้ออกเป็น 2 กลุ่มย่อย

(1) ใช้ข้อมูล known class เท่านั้นในการฝึกฝนแบบจำลอง

(2) ใช้ข้อมูล know class และ unknown class ในการฝึกฝนแบบจำลอง โดยที่ unknown class คือตัวอย่างข้อมูลที่ใช้แทน unknown class ซึ่งอาจจะเป็นข้อมูลภาพจริง หรือเป็นภาพที่ได้จากการสังเคราะห์

ผู้วิจัยได้แสดงรายละเอียดงานวิจัยที่เกี่ยวข้องทั้งหมดดังต่อไปนี้

2.7.1 งานวิจัยกลุ่ม CNN-Based Image Classification

Anh Nguyen Jason Yosinski และ Jeff Clune (Bendale & Boult, 2015) นำเสนองานวิจัยที่ได้ทดลองหาวิธีเพื่อหลอก Deep Neural Network-Based Classifier ด้วยภาพที่ทำการสังเคราะห์ขึ้นมาใหม่ ซึ่งเป็นภาพที่ไม่สามารถแยกแยะประเภทได้ด้วยตามนุษย์ ผลการทดลอง

พบว่าสามารถหลอก Classifier ด้วยคะแนน Confidence ที่สูง ซึ่งจริงๆแล้วภาพเหล่านี้ไม่ได้เป็นภาพที่อยู่ในคลาสใดเลย

Alex Krizhevsky, Ilya Sutskever, และ Geoffrey Hinton (Krizhevsky et al., 2012) นำเสนอบทความวิจัยเรื่อง ImageNet Classification with Deep Convolutional Neural Networks โดยงานวิจัยนี้นับเป็นผลงานชิ้นสำคัญที่ทำให้ CNN เป็นที่สนใจของนักวิจัย โดยนำเสนอ AlexNet ซึ่งเป็น CNN Based Architecture ตัวแรกที่สามารถเอาชนะในการแข่งขัน ImageNet Large Scale Visual Recognition Challenge (ILSVRC) โดยสามารถทำคะแนนได้สูงกว่าเทคนิคแบบดั้งเดิม AlexNet ซึ่งเป็น CNN ตัวแรกๆที่ออกแบบมาเพื่อประมวลผลด้วย GPU มีจุดเด่นที่ใช้ filter ขนาดใหญ่ (11x11) , ประกอบด้วย 8 hidden layer และเป็น CNN โมเดลแรกที่ใช้ Rectified Linear Units (ReLUs) เป็น Activation Function งานวิจัยนี้ยังได้ใช้เทคนิคการทำ Data Augmentation ให้เกิดความหลากหลายของภาพ ซึ่งช่วยให้ลดปัญหา overfit ได้

Karen Simonyan และ Andrew Zisserman (Simonyan & Zisserman, 2014) ได้นำเสนอโมเดล VGG ซึ่งใช้เทคนิคการออกแบบที่แตกต่างไปจาก AlexNet โดยทำการลดขนาดของ filter เป็น 3x3 (AlexNet ใช้ filterขนาด 11x11) แต่ใช้การเพิ่มความลึกใน hidden layer เข้าไปเป็น 16-19 layer ผลที่ได้คือประสิทธิภาพดีขึ้นความแม่นยำที่สูงขึ้น

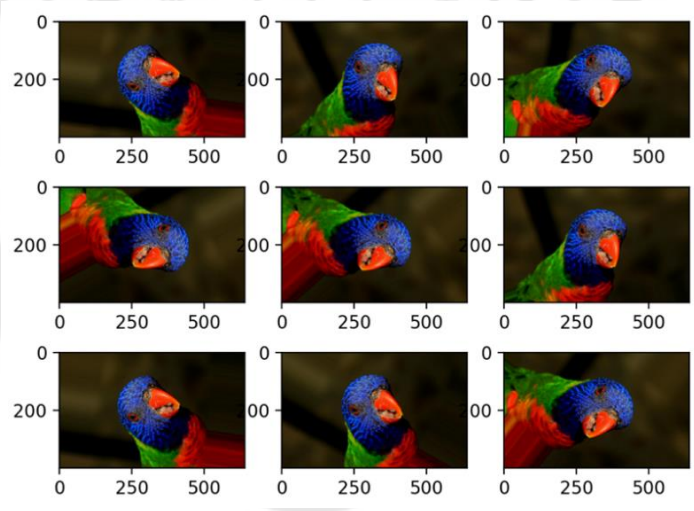
ในงานวิจัยหลัง AlexNet จะเห็นได้ว่าความลึกของ hidden layer นั้นมีการเพิ่มขึ้นมาโดยตลอดหลายปี ความลึกที่เพิ่มขึ้นนี้ส่งผลให้ความแม่นยำสูงขึ้น แต่เมื่อจุดนึงความแม่นยำจะลดลงอันเนื่องมาจากปัญหาค่าเกรเดียนต์สูญหาย (gradient vanishing)

Kaiming He, Xiangyu Zhang, Shaoqing Ren และ Jian Sun (He et al., 2015) ได้นำเสนอ CNN Architecture ที่สามารถมีความลึกมากและมี Accuracy สูงมากได้ โดยมี hidden layer สูงสุดถึง 152 layer และแก้ปัญหา gradient vanishing ด้วยเทคนิค skip connection, batch normalization ทำให้ได้ความแม่นยำสูง ถึงแม้จะเป็นโมเดลที่มีความลึกมากก็ตาม

Jonathan Huang และคณะ (Huang et al., 2016) ได้นำเสนองานวิจัยนี้เป็นการศึกษาและเปรียบเทียบประสิทธิภาพทั้งในแง่ความเร็วและความแม่นยำของ Object Detector โดยเลือก Detector ที่ใช้ Meta Architecture เป็น SSD (Single Shot Multibox Detector) , Faster R-CNN) และ R-FCN (Region-based Fully Convolutional ซึ่ง Meta Architecture เหล่านี้พัฒนาบนพื้นฐานของ Convolution Neural Network สำหรับการตรวจจับวัตถุ โดยใช้ CNN Architecture อย่าง Inception , VGG , Resnet ,Mobile Net ในการทำ Feature Extraction โดยผลการศึกษาการเปรียบเทียบ Accuracy กับเวลาการประมวลผล (GPU Time) พบว่า Faster

RCNN ที่ใช้ Inception Resnet V2 เป็น Feature Extractor ให้ Accuracy สูงสุด ตามด้วย Faster RCNN ที่ใช้ Resnet เป็น Feature Extractor ซึ่งใช้เวลาในการประมวลผลนานที่สุดตามลำดับ ส่วน SSD ที่ใช้ Feature Extractor เป็น MobileNet

Data Augmentation (Krizhevsky et al., 2012) ถูกใช้เพื่อเพิ่มจำนวนตัวอย่างรูปโดยอาศัยรูปเดิมมาปรับซึ่งสามารถทำได้โดยหลายวิธีเช่นทำ Transforms โดยการทำ Flip Horizontal, Flip Vertical, การปรับขนาดหรือเลือกมาบางส่วน (Random Crops/Scale) การขยับภาพ, (Translation) , Rotation หมุนภาพ , ยืดภาพออก (Stretching), การเฉือนภาพ (Shearing) รวมถึงการปรับค่า RGB หรือ การปรับ Contrast (Wong, Gatt, Stamatescu, & McDonnell, 2016) ตัวอย่างของการทำ Data Augmentation แสดงในภาพประกอบที่ 4 และนอกจาก Data Augmentation จะถูกใช้มาเพื่อเพิ่มจำนวนตัวอย่างรูปแล้ว ยังมีประโยชน์ในการลด Overfitting อีกด้วย



ภาพประกอบ 15 Data Augmentation

ที่มา: How to Configure Image Data Augmentation When Training Deep Learning Neural Networks. (n.d.). Retrieved from https://machinelearningmastery.com

2.7.2 งานวิจัยกลุ่ม Open-set classification

Walter J. Scheirer และคณะ(Scheirer et al., 2013) ได้ทำการศึกษาปัญหา Open-set และได้นิยามทฤษฎีที่น่าสนใจหลายหัวข้อเช่น การหาความซับซ้อนของปัญหา (openness score), Open-set risk minimization และยังนำเสนออัลกอริทึมหลายตัว เช่น 1-vs-set machine algorithm ซึ่งพัฒนามาจาก SVM กล่าวคือ 1-vs-set machine จะใช้ 2 hyperplane โดยที่ hyperplane แรกจะได้มาจาก SVM และ hyperplane อีกอันจะสร้างให้ขนานกับ hyperplane แรก ข้อมูลที่นำมาทดสอบใดที่อยู่ระหว่าง 2 hyperplane จะจำแนกให้เป็น target class (known class) ส่วนข้อมูลที่อยู่นอก 2 hyperplane จะถูกจำแนกเป็น unknown class

Abhijit Bendale และคณะ (Bendale & Boult, 2015) ได้นำเสนอวิธีการแก้ปัญหา Open-set บน Deep Neural Network โดยงานวิจัยนี้สะท้อนให้เห็นว่าที่ชั้น SoftMax Layer เป็นปัญหาและต้องปรับเปลี่ยนเพื่องาน Open-set classification จึงนำเสนอ OpenMax และ Multi-class Meta-Recognition เพื่อใช้ในการคำนวณหา class score แทนที่ SoftMax function ทำให้สามารถระบุ unkown class ได้ดีขึ้นเมื่อเปรียบเทียบกับ 1-vs-set machine
OpenMax ยังถูกนำไปพัฒนาต่อในงานของ Zongyuan Ge และคณะ (Ge, Demyanov, Chen, & Garnavi, 2017) ที่ได้นำเสนอ G-OpenMax โดยนำเอา OpenMax ไปใช้ร่วมกับ Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) เพื่อทำการสังเคราะห์ภาพ unknown class ขึ้นมา (synthetic image) ดังนั้นภาพ known class และ ภาพ unknown class จะถูกใช้ในการเทรนโมเดล ซึ่งจะแตกต่างจาก OpenMax เดิมที่จะใช้แค่ภาพ known class ร่วมกับ EVT ในการประมาณค่าความน่าจะเป็น

Lawrence Neal, Matthew Olson และคณะ (Neal, Olson, Fern, Wong, & Li, 2018) ได้นำเสนอการแก้ปัญหาโดยใช้เทคนิค data augmentation ร่วมกับ GANs ในการสร้างภาพสังเคราะห์ภาพที่มีลักษณะคล้ายกับภาพต้นฉบับ (counterfactual image) โดยใช้ภาพในชุดข้อมูลที่เป็นคลาสที่รู้จัก (known class) เป็นต้นฉบับในการสร้างภาพแบบคล้าย ภาพที่สังเคราะห์ขึ้นมาจะไม่ถูกจัดให้อยู่ไปรวมในคลาสที่รู้จัก (known class) คลาสใด แต่จะเป็นคลาสพิเศษที่เพิ่มขึ้นมาอีกคลาสเพื่อใช้ในการเทรน ผลการทดลองแสดงให้เห็นประสิทธิภาพที่ดีกว่า Open-Max และ G-OpenMax บนชุดข้อมูล CIFAR10, CIFAR50 และ TinyImagenet

Lei Shu, Hu Xu และคณะ (Shu, Xu, & Liu, 2018) ได้ใช้เทคนิคการทำ classification แบบ open-set โดยนำเสนอ joint model framework ที่ประกอบด้วยการทำ classification,

auto-encoding และclustering ผลที่ได้คือนอกจากจะสามารถจำแนกคลาสที่ไม่รู้จัก unknown class แล้วยังสามารถจัดกลุ่ม (clustering) คลาสที่ไม่รู้จักได้อีกด้วย

Kumar Sricharan และ Ashok Srivastava (Sricharan & Srivastava, 2018) ได้นำเสนอการสร้าง Classifiers เพื่อให้สามารถทำนาย sample ที่อยู่นอกกลุ่ม (out-of-distribution) โดยได้ใช้ GAN เพื่อสร้าง out of distribution sample ที่จะใช้สำหรับการเทรนโมเดล GANs โมเดลที่เลือกใช้ได้แก่ Standard GAN, ConfGAN, DCGAN, BoudaryGAN มาสร้าง out of distribution sample เพื่อใช้เทรนบน VGGNet และชุดข้อมูลที่ใช้คือ MNIST, CIFAR10, SVHN, ImageNet

2.7.3 งานวิจัยกลุ่ม Anomaly Detection

Bernhard Schölkopf และคณะ (Schölkopf et al., 2000) ได้นำเสนอ One-Class Support Vector Machine (OC-SVM) ซึ่งเป็นเทคนิคที่พัฒนาจาก Support Vector Machine โดยจะใช้เพียงแค่ข้อมูลปกติ (normal data) ในขั้นตอนการ training และใช้โมเดลนี้เพื่อหาข้อมูลที่ผิดปกติในขั้นตอนการ testing รายละเอียดได้กล่าวไว้ในหัวข้อที่ 2.4

Tax และคณะ ได้นำเสนอ Support Vector Data Description (SVDD) (Tax & Duin, 2004) โดยได้รับอิทธิพลมาจาก SVM ในขั้นตอนการ training ของ SVDD จะเป็นการค้นหา hyper-sphere ที่เล็กที่สุด (minimum radius) ที่สามารถบรรจุข้อมูล inlier ได้ เมื่อมีข้อมูลใหม่ (new observation) หากตกอยู่ใน hyper-sphere ก็จะถือเป็นข้อมูลปกติ แต่หากตกอยู่ข้างนอก hyper-sphere ก็จะถือว่าเป็นข้อมูลผิดปกติ

Fei Tony Liu, Kai Ming Ting และ Zhi-Hua Zhou (F. T. Liu, K. M. Ting, & Z. Zhou, 2008) ได้นำเสนอ Isolation Forest สำหรับการทำ Anomaly Detection โดยใช้เทคนิคการ partitioning ข้อมูลเพื่อสร้างเป็น tree และใช้ความลึกของ tree เพื่อหาว่าข้อมูลใดเป็น Anomaly รายละเอียดได้กล่าวไว้ในหัวข้อที่ 2.5

นอกจากงานวิจัยที่เป็น traditional machine learning based แล้ว ยังมีงานวิจัยที่ใช้เทคนิค deep neural netework อีกหลายงานวิจัยเช่น

Poojan Oza และ Vishal M. Patel (Oza & Patel, 2019) ได้นำเสนอการทำ OC-CNN (One-Class Classification บน Convolution Neural Network) โดยนำ Pre-trained AlexNet และ VGG มาปรับโครงสร้างในบาง Layer และใช้การ generate sample สำหรับ negative class ขึ้นมา กล่าวคือที่ Layer ท้าย เมื่อ flatten activation map เป็น feature vector แล้วจะยังไม่ส่งไปชั้น Softmax Layer ทันที แต่จะนำไปรวมกับ pseudo-negative class data ที่ generate ขึ้นมา

จาก Gaussian Function ก่อนแล้วจึงส่งไป Softmax Layer ในการทดลองได้ใช้ชุดข้อมูลภาพ Abnormality-1001 , UMDAA-02 Face และ FouderType-200 และเปรียบเทียบประสิทธิภาพกับวิธีการอื่นได้แก่ OC-SVM (One-Class Support Vector Machine) , BSVM (Binary SVM) , MPM (MiniMax Probability Machine), SVDD (Support Vector Data Description), OC-NN (One-Class Neural Network) และ OC-SVM^{+} ผลการวิจัยพบว่า OC-CNN และ OC-SVM^{+} ให้ประสิทธิภาพสูงสุดสุดใกล้เคียงกันกับทั้ง 3 ชุดข้อมูล ทั้งบน AlexNet และ VGGNet

Mohammad Sabokrou และคณะ (Sabokrou, Khalooei, Fathy, & Adeli, 2018) ได้นำ Generative Adversarial Networks (GANs) มาประยุกต์เพื่อทำ Novelty Detection โดยใช้ Deep Neural Network 2 ตัว โดย Network แรกเรียกว่า Refine (R) และ ตัวท่าสองคือ Detector (D) มีหลักการทำงานคือ R Network จะทำการ reconstruct ข้อมูลใดๆ ในกรณีที่ข้อมูลนั้นเป็น positive sample แล้ว R จะทำการ reconstruct เพื่อให้ข้อมูลใหม่ที่มีค่าความน่าจะเป็น positive class สูงขึ้น แต่ถ้ากรณีที่ข้อมูลเป็น negative sample (outlier) R จะไม่สามารถทำ Reconstruct ได้ดีนักทำให้ค่าความน่าจะเป็น positive class ลดลดนั่นเอง จากนั้นข้อมูลที่ผ่านการ reconstruct แล้ว จะส่งต่อไปยัง Network D เพื่อทำการแยก positive sample กับ negative sample ในการทดลองได้ใช้ชุดข้อมูล MNIST และ Caltech-256 โดยมีการสุ่มภาพจาก Categories อื่นเข้ามาใส่ เพื่อให้เป็นภาพ Outlier ผลการวิจัยพบว่าสามารถ detect outlier ที่ไม่ได้อยู่ในคลาสที่ใช้เทรนได้ดีและเมื่อเทียบกับวิธีอื่น ๆ แล้วมีประสิทธิภาพที่ดีกว่า

Raghavendra Chalapathy, Aditya Krishna Menon และ Sanjay Chawla (Chalapathy, Menon, & Chawla, 2018) ได้นำเสนอ OC-NN (One-Class Neural Network) โมเดลสำหรับการทำ Anomaly Detection โดยการสร้าง Feed Forward Network และใช้ OC-SVM (One-Class Support Vector Machine) มาใช้เสมือนเป็น loss function ในกระบวนการเทรนโมเดล และได้นำเสนอ alternating minimization algorithms เป็น optimizer วิธีการที่งานวิจัยนี้ได้นำเสนอจะแตกต่างจากวิธีแบบไฮบริด ที่จะใช้ autoencoder, GANs เพื่อให้ได้ feature แล้วค่อยส่งต่อไปให้ Detector ตรวจสอบว่าเป็น anomaly หรือไม่ ซึ่งแตกต่างจาก OC-NN ตรงที่ feature ที่ได้จาก hidden layer จะถูกสร้างขึ้นเพื่อการทำ anomaly detection ในการทดลองได้ใช้ชุดข้อมูล Synthetic,MNIST,CIFAR-10,GTSRB และได้เปรียบเทียบกับวิธีการอื่น ที่ใช้ Deep Learning เช่น Robust Convolutional Autoencoder, Deep Convolutional Autoencoder (DCAE) หรือวิธีแบบเชิงเส้นเช่น OC-SVM/SVDD ผลที่ได้คือ OC-NN มีประสิทธิภาพใกล้เคียงและดีกว่ากลุ่ม OC-SVM/SVDD ในบางกรณี

บทที่ 3
วิธีดำเนินงานวิจัย

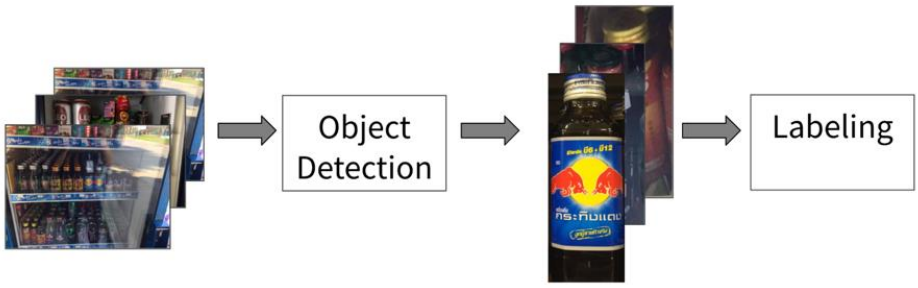
ในการทำวิจัยครั้งนี้ผู้วิจัยผู้วิจัยได้ดำเนินการตามขั้นตอนดังนี้

1. การเตรียมข้อมูลสำหรับการวิจัย
2. การสร้างแบบจำลองการจำแนกภาพ
3. การวัดประสิทธิภาพและวิเคราะห์ผล

3.1 การเตรียมข้อมูลสำหรับการวิจัย

ในงานวิจัยนี้ผู้วิจัยได้สร้างชุดข้อมูลเครื่องดื่ม ซึ่งได้มาจากสกัดมาจากภาพถ่ายตู้แช่สินค้าในประเทศไทยโดยภาพถ่ายเหล่านี้ถูกถ่ายด้วยกล้องสมาทโฟน ตัวอย่างภาพถ่ายแสดงในภาพประกอบที่ 16

ภาพประกอบ 16 ภาพถ่ายตู้แช่ต้นฉบับ



ภาพประกอบ 17 แสดงขั้นตอนสร้างชุดข้อมูล

เนื่องจากชุดข้อมูลเครื่องดื่มนี้เป็นชุดข้อมูลที่สร้างขึ้นมาใหม่จึงต้องมีกระบวนการสร้างและเตรียมชุดข้อมูลให้ได้มาตรฐานตามขั้นตอนดังต่อไปนี้

3.1.1 **ตรวจจับวัตถุที่สนใจ** (Object Detection)

Object Detection มีหน้าที่เพื่อตรวจหาขวดใดๆในภาพถ่ายตู้แช่ ดังที่แสดงในภาพประกอบที่ 18 พัฒนาขึ้นมาโดยใช้ Faster R-CNN detector (Girshick, 2015) บน TensorFlow Object Detection API (Huang et al., 2016) และ Detector Model ทำงานโดยใช้ Pre-trained model ที่ถูกฝึกฝนบนชุดข้อมูล COCO-dataset (Lin et al., 2014) ภาพขวดที่ตรวจจับได้จะถูกนำไปทำการจัดประเภทของภาพ (Labeling)

ภาพประกอบ 18 Object Detection

3.1.2 **จัดประเภทของภาพ** (Labeling)

ในขั้นตอนนี้จะนำภาพที่ได้จากขั้นตอนแรกมาจัดประเภท ตารางที่ 1 แสดงจำนวนภาพขวดเครื่องดื่มแยกตามยี่ห้อซึ่งมีทั้งหมด 44 ยี่ห้อ แต่ในงานวิจัยนี้สนใจเฉพาะเครื่องดื่ม 3 ยี่ห้อ ได้แก่ cbd, m150 และ redbull โดยนำภาพ 3 ยี่ห้อนี้จัดให้เป็น 3 คลาสหลัก (Known class) ได้แก่ p1,p2 และ p3 ตามลำดับ และเครื่องดื่มยี่ห้ออื่นๆที่เหลือ 41 ยี่ห้อจะจัดให้เป็นภาพในคลาสอื่นๆ (Other Class) แสดงในภาพประกอบ และ ภาพประกอบที่ 19 และ 20 ตามลำดับ

ตาราง 1 แสดงจำนวนภาพที่สกัดได้แยกตามยี่ห้อเครื่องดื่ม

id	**ยี่ห้อ**	**จำนวนภาพ**
0	aje big grape	150
1	aje big green	148
2	aje big orange	108
3	aje big strawberry	152
4	c-vitt lemon	151
5	c-vitt orange	164
6	cbd	1027
7	chalarm black galingale	213
8	coke	285
9	est	181
10	est play cream soda	180
11	fanta strawberry	182
12	gsd tang gui jub	327
13	ishiton genmaicha	223
14	jubjai	187
15	kratingdaeng-extra abc	217
16	kratingdaeng-extra zinc	295
17	kratingdaeng-g2	362
18	kratingdaeng-g3	290
19	lipo	200

ตาราง 1 (ต่อ)

id	**ยี่ห้อ**	**จำนวนภาพ**
20	lipo plus	174
21	lipton	175
22	m-150 storm	252
23	m150	1024
24	mirinda mix-it blueberry	176
25	mirinda mix-it grape	107
26	mirinda orange	204
27	mirinda strawberry	191
28	oishi green tea genmai	188
29	oishi green tea original	215
30	oishi green tea water grape	177
31	oishi green tea water lemon	135
32	pharyanak	137
33	pharyanak herbal	162
34	ranger	133
35	ready	231
36	redbull	1032
37	som in sum	243
38	sponsor	237
39	theoplex-l	357
40	vitamix v1000	111
41	white shark	227
42	yen yen herb	186
43	yen yen honey	193
	รวมทั้งสิ้น	**11309**

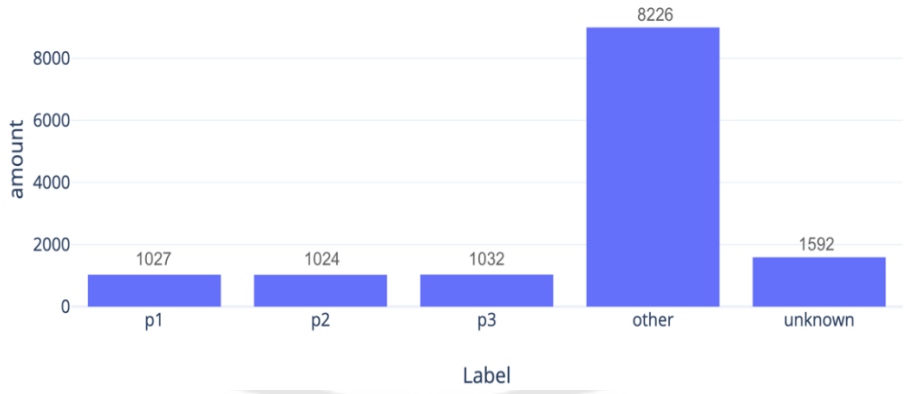
ภาพประกอบ 19 เครื่องดื่มคลาส p1, p2 และ p3 ซึ่งเป็นคลาสหลัก (Known class) ที่ต้องทำการจำแนก

ภาพประกอบ 20 ภาพเครื่องดื่มในประเภทอื่นๆ (Other Class) โดยมาจากการนำภาพเครื่องดื่ม 41 ยี่ห้อ ที่อยู่นอกเหลือจาก (Knonw class) มารวมกัน

หลังจากกระบวนการจัดประเภทของภาพแล้ว สามารถสรุปได้เป็น ประเภท p1 จำนวน 1027 รูป ประเภท p2 จำนวน 1024 รูป ประเภท p3 จำนวน 1032 รูป และประเภท other จำนวน 8226 รูป

เนื่องจากจุดประสงค์ของงานวิจัยนี้คือสามารถทำนายภาพที่อยู่นอกเหนือกลุ่มภาพที่ใช้เทรนโมเดลได้ (Unknown class) ผู้วิจัยจึงได้เตรียมชุดข้อมูลอีกชุด โดยการดึงภาพมาจากอินเทอร์เน็ต จำนวน 1,592 รูป ซึ่งเป็นภาพของเครื่องดื่มซึ่งไม่มียี่ห้อซ้ำกับ 44 ยี่ห้อที่ใช้ในการเทรนโมเดล ดังที่แสดงในภาพประกอบที่ 21

ภาพประกอบ 21 ตัวอย่างภาพขวดเครื่องดื่มในกลุ่ม Unknown Class



ภาพประกอบ 22 กราฟแสดงจำนวนภาพขวดแยกเป็นคลาสต่างๆ

3.2 การสร้างแบบจำลองการจำแนกภาพ

จุดประสงค์ของการสร้างแบบจำลองก็เพื่อจำแนกภาพขวดเครื่องดื่มออกเป็นคลาสต่างๆ และหากขวดเครื่องดื่มไม่อยู่ในคลาสใดในขั้นตอนการเรียนรู้ก็ต้องจำแนกออกเป็นคลาสที่ไม่รู้จักได้

$$Dataset = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\} \quad (16)$$

$$y_i \in \{label_1, \dots, label_n\}$$

จากสมการที่ 16 ชุดข้อมูล (Dataset) ประกอบขึ้นจากเซ็ตของภาพโดยที่

x_i คือ ภาพที่ 1 ถึงภาพที่ n

y_i คือ คลาส (known class label) ของภาพที่ 1 ถึงภาพที่ n

ภาพ x_i จะถูกจำแนกออกเป็นคลาส(known class label) และสามารถจำแนกเป็นคลาสที่ไม่รู้จัก (unknown class) ในกรณีที่ไม่อยู่ในคลาสใด(known class label) ในขั้นตอนการฝึกฝน จากเป้าหมายข้างต้น ผู้วิจัยได้ออกแบบการทดลองออกเป็น 3 โครงสร้าง เพื่อทำการทดลองและหาวิธีการที่มีประสิทธิภาพดีที่สุดดังต่อไปนี้

1. N-Binary Classifier
2. N+1 Class Classifier
3. 2 Layer Classification

โดยที่ระบบทั้งหมดจะทำงานอยู่บน Google Compute Engine โดยกำหนดความต้องการไว้เป็น Machine type n1-highmem-2 (2 vCPUs, 13 GB memory) ,CPU platform Intel Broadwell, GPUs 1 x NVIDIA Tesla K80

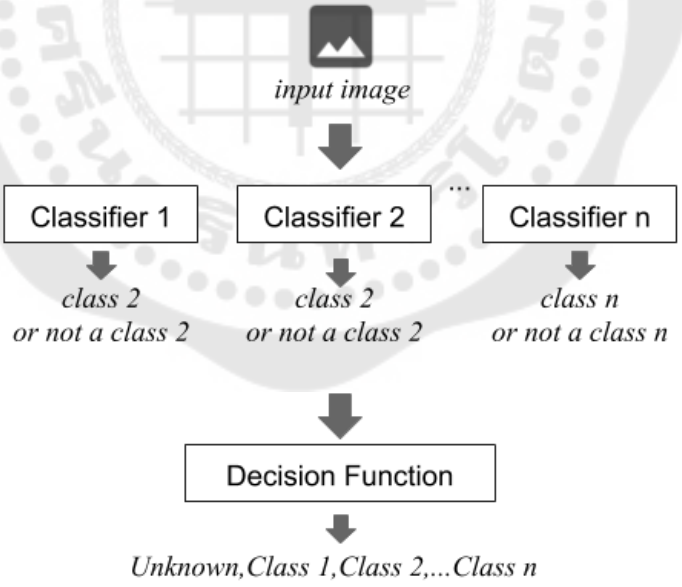
3.2.1 N-Binary Classifier

N-Binary Classifier เป็นการฝึกแบบจำลองเท่ากับจำนวนประเภท (N Classifier) โดยแต่ละโมเดลจะใช้ข้อมูลภาพ 2 ประเภท (Binary Classification) คือภาพประเภทที่สนใจ 1 คลาสเท่านั้น และภาพประเภทอื่นๆ ซึ่งทำให้แต่ละแบบจำลองจะรับผิดชอบในการทำนายเพียงแค่ 1 คลาส ซึ่งคำตอบที่ได้จะเป็นใช่หรือไม่ใช่คลาสนั้นเท่านั้น ด้วยวิธีการฝึกฝนแบบจำลองแบบนี้ทำให้ได้แบบจำลองทั้งหมด N ชุดเท่ากับจำนวนคลาสที่มี (known Classes)

ในขั้นตอนสุดท้ายคือการสรุปการทำนายจากทุกแบบจำลอง สามารถสรุปได้เป็น 3 กรณี ดังต่อไปนี้

1. กรณีที่ทุกแบบจำลอง ทำนายว่าภาพเป็นที่ไม่อยู่ในคลาสใดเลยที่ใช้ในขั้นตอนการฝึกฝน (Known classes) จะสามารถสรุปได้ว่าเป็นประเภทที่ไม่รู้จัก (Unknown Class)
2. กรณีที่มี 1 แบบจำลอง ทำนายว่าภาพเป็นที่อยู่ในกลุ่มคลาสที่ใช้ในขั้นตอนการฝึกฝน (Known class) จะสามารถสรุปได้ว่าเป็นภาพนั้นเป็นประเภทที่ Classifier นั้นทำนาย
3. กรณีที่มีมากกว่า 1 แบบจำลอง ทำนายว่าภาพเป็นที่อยู่ในกลุ่มคลาสที่ใช้ในขั้นตอนการฝึกฝน(Known class) ให้เลือกคำตอบจากแบบจำลองที่มีคะแนนความเชื่อมั่น (SoftMax probability score) สูงที่สุด

ภาพประกอบที่ 23 แสดงโครงสร้างของ N-Binary Classifier และขั้นตอนการทำนาย (Prediction procedure) โดยภาพใดๆที่ใช้ในการทำนาย จะถูกจำแนกออกเป็นคลาสใดๆตั้งแต่ Class1 – Class N หรือ Unknown Class



ภาพประกอบ 23 N-Binary classifier prediction procedure

ผู้วิจัยได้ออกแบบ N-Binary Classifier โดยมีที่มาจากเทคนิค One-vs.-rest ที่ใช้ในการทำ multi-class class โดยข้อมูลที่ใช้ในการฝึกฝนแต่ละแบบจำลองจะใช้ข้อมูลภาพ known class คือ $\{p1, p2, p3\}$ และใช้ภาพ other class เพื่อเป็นตัวแทนของ unknown class ซึ่งการฝึกฝน

แบบจำลองที่ภาพ known class ร่วมกับ unknown class เป็นเทคนิคที่ปรากฏอยู่ในหลายๆ งานวิจัยที่เกี่ยวข้องดังที่ผู้วิจัยได้สรุปไว้ในบทที่ 2

3.2.2 N+1 Class Classifier

N+1 Class Classifier เป็นการฝึกแบบจำลองโดยใช้ข้อมูลภาพที่รู้จักทั้งหมด N คลาส (known Classes) และข้อมูลภาพคลาสเพิ่มเติม (Additional class) อีก 1 คลาส ซึ่งจะเป็นภาพที่อยู่นอกเหนือจาก N ประเภท (N Classes) โดยในการทดลองคลาสเพิ่มเติมยังสามารถแบ่งออกได้เป็น 2 แบบคือ N+Unknown และ N+Combination

(1) N+Unknown ใช้ข้อมูลภาพที่รู้จักทั้งหมด N คลาส (known Classes) และข้อมูลภาพประเภทอื่นๆ (other class) ที่ใช้เป็นตัวแทนของ unknown class ดังนั้นในการฝึกฝนแบบจำลองจะใช้ภาพ $\{p1, p2, p3, other\}$

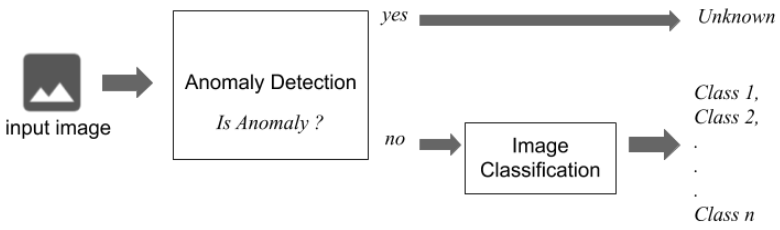
(2) N+Combination ใช้ข้อมูลภาพที่รู้จักทั้งหมด N คลาส (known Classes) และข้อมูลภาพคลาสพิเศษซึ่งมาจากการนำภาพทั้งหมด N คลาส (known Classes) มารวมกัน ดังนั้นในการฝึกฝนแบบจำลองจะใช้ภาพ $\{p1, p2, p3, (p1 + p2 + p3)\}$

ผู้วิจัยได้ออกแบบ N+1 Class Classifier โดยได้อ้างอิงจากงานวิจัยที่เกี่ยวข้องที่ใช้ภาพ unknown class ในการฝึกฝนแบบจำลอง ในกรณีของ N+Unknown คือการใช้ภาพที่อยู่นอกเหนือ known class มาเป็นตัวแทนของ unknown class และในกรณีของ Combination ผู้วิจัยได้ออกแบบการทดลองโดยสมมติให้ไม่มีตัวอย่างภาพที่จะใช้เป็นตัวแทนของ unknown class อยู่เลย จะสามารถนำภาพ known class $\{p1, p2, p3\}$ มารวมกันเป็นคลาสใหม่เพื่อใช้เป็นตัวแทนของ unknown class ได้หรือไม่

3.2.3 2 Layer Classification (Two Layer Classification)

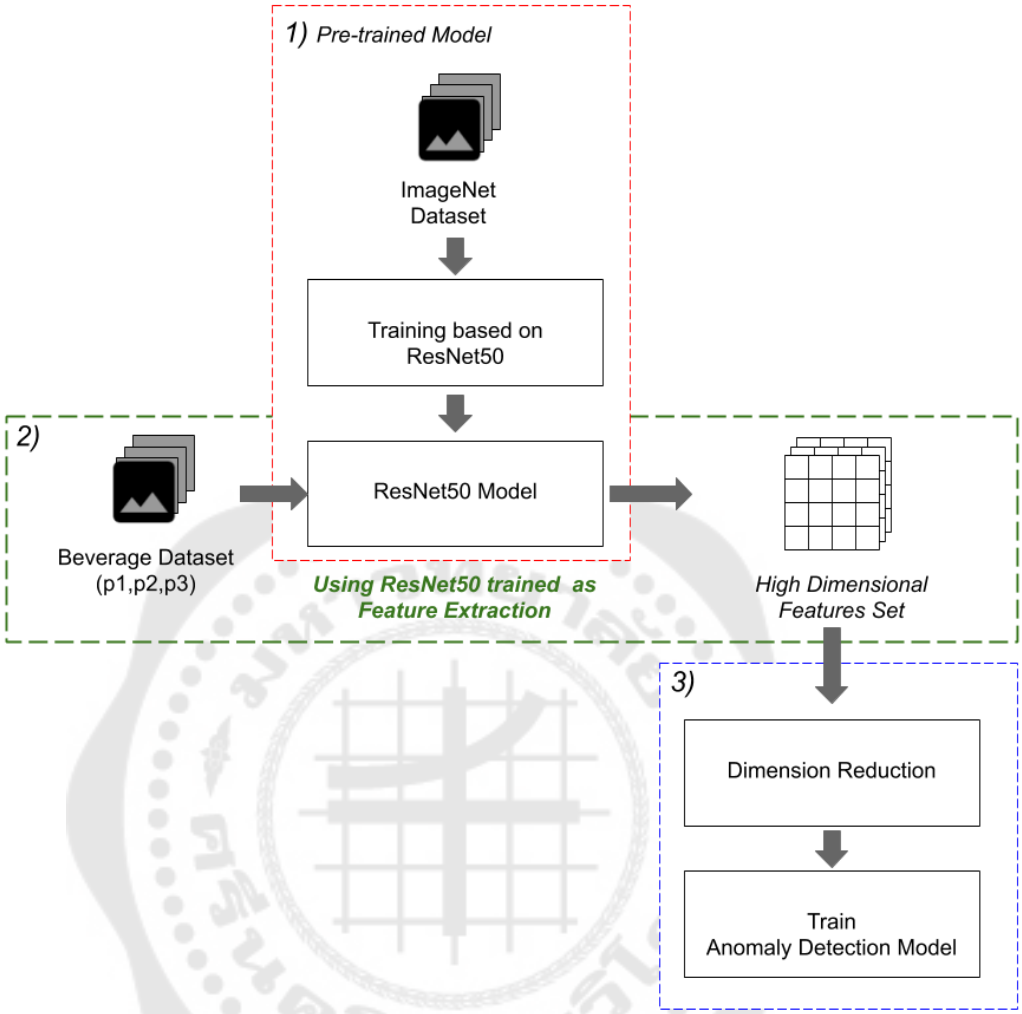
ผู้วิจัยได้ออกแบบ 2 Layer Classification ภายใต้สมมติฐานว่าข้อมูล unknow class ก็คือข้อมูลผิดปกติ(anomaly data) ที่สามารถใช้การคัดกรองโดยใช้อัลกอริทึมอย่าง OC-SVM หรือ Isolation Forest ได้ ดังนั้น Two Layer Classification จะประกอบขึ้นด้วยแบบจำลองที่แบ่งออกเป็น 2 ชั้น ในชั้นแรกจะเป็นการตรวจสอบว่าภาพที่ใช้ทำนายเป็นภาพที่เป็นภาพที่อยู่ในคลาสใด ๆ ในขั้นตอนการฝึกฝนหรือไม่ โดยใช้วิธีการตรวจหาสิ่งผิดปกติ (Anomaly detection) หากภาพใดเป็นภาพผิดปกติก็จะถูกจำแนกเป็นคลาสที่ไม่รู้จัก (unknown class) แต่หากภาพใดเป็น

ภาพปกติก็จะถูกส่งไปยังชั้นที่สอง เพื่อทำการจำแนกภาพออกเป็นคลาส $p1, p2, p3$ ดังที่แสดงในภาพประกอบที่ 24



ภาพประกอบ 24 Procedure

อย่างไรก็ตามในขั้นตอนการฝึกฝน Two Layer Classification จะมีความซับซ้อนกว่าวิธี N-Binary Classifier และN+Combination สามารถสรุปเป็นขั้นตอนได้ตามภาพประกอบที่ 25 อธิบายได้ออกเป็น 3 ส่วนคือ Pre-trained model, การสกัดฟีเจอร์(Feature Extraction), การฝึกฝนแบบจำลองสำหรับหาข้อมูลผิดปกติ (Train Anomaly Detection)



ภาพประกอบ 25 ขั้นตอนการฝึกฝนของ Multi-class classifier + OC-SVM และ Multi-class classifier + Isolation Forest

1. **Pre-trained model**

Pre-trained model คือแบบจำลองที่ถูกฝึกฝนมาเรียบร้อยแล้ว โดยผู้วิจัยได้เลือกใช้ pre-trained ResNet50 model จาก Keras Framework ที่ถูกฝึกฝนบนชุดข้อมูล ImageNet

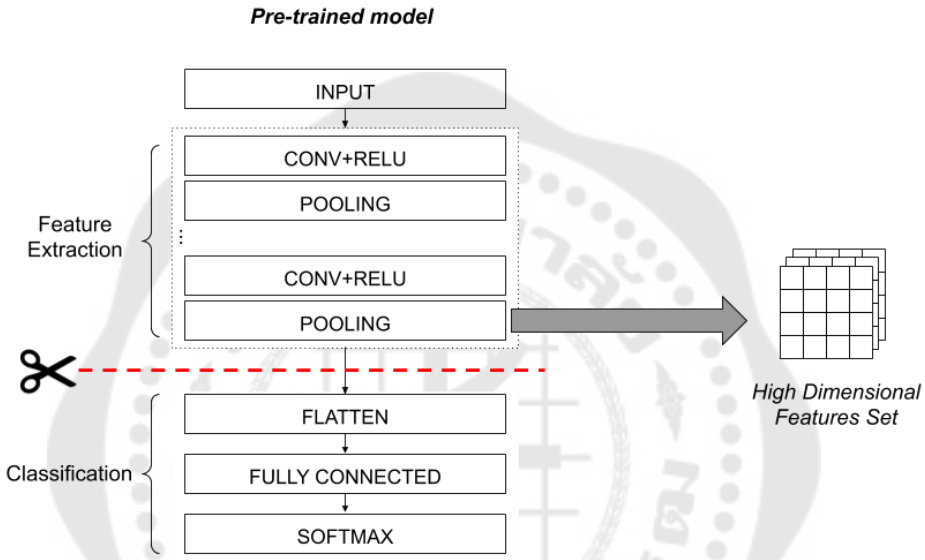
2. **Feature Extraction**

Resnet50 pre-trained model จะถูกใช้ในการสกัดฟีเจอร์ (Feature Extraction) โดยใช้ภาพจากชุดข้อมูลเครื่องดื่ม ส่งเข้า pre-trained model ที่มีการปรับโครงสร้างตามภาพประกอบที่ 26 แสดงให้เห็นว่ามีการปรับโครงสร้าง

โดยตัดชั้นที่ใช้ในการทำ classification ออกไป เนื่องจากฟีเจอร์ที่สกัดได้นี้จะใช้เพื่อการสร้างโมเดล (Anomaly Detection Model)

3. Anomaly Detection Model

ฟีเจอร์ที่สกัดได้จะเป็นข้อมูลที่มีหลายมิติ ดังนั้นผู้วิจัยจึงออกแบบให้มีการลดมิติของข้อมูล (Dimension Reduction) ก่อนที่ถูกนำไปใช้ในการขั้นตอนการฝึกฝน โดยใช้ OC-SVM และ Isolation Forest



ภาพประกอบ 26 การใช้ Pre-trained model เพื่อสกัดฟีเจอร์ โดยการตัดโครงสร้างสำรับการทำ classification ออกไป

การใช้ pre-trained model มาสกัดฟีเจอร์เป็นเทคนิคที่ปรากฏในหลายๆงานวิจัย เช่น งานวิจัยที่ผู้วิจัยสรุปมาในหัวข้อ 2.7.3

ตาราง 2 เปรียบเทียบรูปแบบการฝึกฝนแบบจำลอง

รูปแบบการฝึกฝนและทดสอบ	จำนวนแบบจำลอง	หมายเหตุ
N-Binary Classifier	N	
N+1 Class Classifier	1	
2-Layer Classification	2	1 anomaly, 1 Image Classification

3.2.3 การวัดประสิทธิภาพและวิเคราะห์ผล

ในการวัดประสิทธิภาพของแบบจำลองที่สร้างขึ้นด้วยเทคนิคต่าง ๆ เพื่อดูประสิทธิภาพในการจำแนก รวมถึงเวลาที่ใช้การฝึกฝนแบบจำลอง ผู้วิจัยได้กำหนดเกณฑ์ในการประเมินประสิทธิภาพโดยใช้ Accuracy score, Precision score, Recall score, F-1 Score สามารถคำนวณค่าได้จากสมการที่ 17-20 ตามลำดับ

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP} \tag{17}$$

$$Precision = \frac{TP}{TP + FP} \tag{18}$$

$$Recall = \frac{TP}{TP + FN} \tag{19}$$

$$F1 = \frac{2 \times Precission \times Recall}{Precission + Recall} \tag{20}$$

จากสมการ

TP คือ True Positive

TN คือ True Negative

FP คือ False Positive

FN คือ False Negative

บทที่ 4
การดำเนินการวิจัยและผลการวิจัย

ผู้วิจัยได้ดำเนินการวิจัยโดยทำการทดลองตลอดจนการวัดประสิทธิภาพเพื่อให้บรรลุจุดประสงค์ของการวิจัยที่ได้กำหนดไว้ โดยกำหนดการทดลองไว้ดังต่อไปนี้

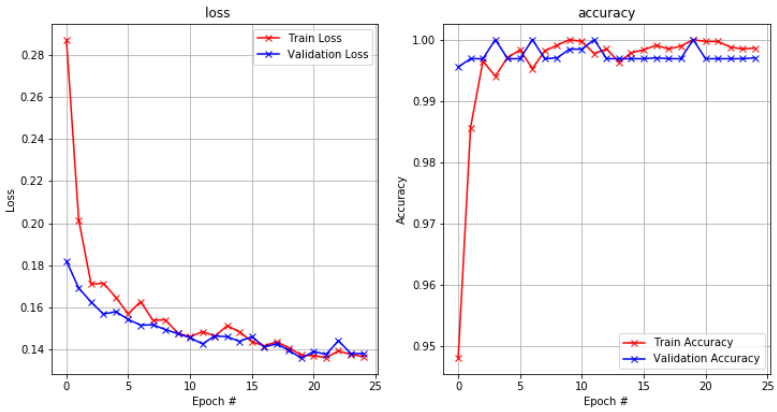
1. Based Model (Multi-Class Model)
2. การทดลอง Exploring SoftMax Probability Score
3. การทดลอง N-Binary Classifier
4. การทดลอง N+Unknown Class Classifier
5. การทดลอง N+Combination Class Classifier
6. การทดลอง 2 Layer Classification (OC-SVM)
7. การทดลอง 2 Layer Classification (Isolation Forest)
8. Dimension Reduction

4.1 Based Model (Multi-Class Model)

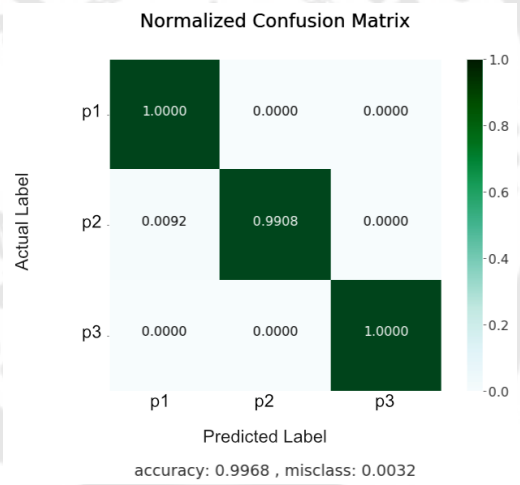
ในการทดลองแรกนี้ผู้วิจัยต้องการสร้างโมเดลโดยมีจุดประสงค์เพื่อดูประสิทธิภาพของการทำนายภาพกลุ่มหลัก (known class) ก่อน จึงสร้างแบบจำลองในการจำแนกภาพขวดโดยใช้ภาพกลุ่ม known class ซึ่งประกอบด้วยคลาส $p1, p2, p3$ เพียงเท่านั้นทั้งในขั้นตอนการฝึกฝนและทดสอบแบบจำลองดังนั้นผลการทำนายของแบบจำลองนี้ที่ได้จะเป็น $\hat{\mathbf{y}} \in \{\mathrm{p1}, \mathrm{p2}, \mathrm{p3}\}$. โดยเลือกสถาปัตยกรรมแบบ Resnet50 และมีการกำหนด hyperparameter เป็น batch size = 32, epoch = 25, optimizer = SGD, learning rate = 0.1, Momentum = 0.9, Weight decay = 0.004, Loss function = cross entropy loss

ภาพประกอบที่ 27 แสดง loss และ accuracy ของการ train และ validation ซึ่งจะเห็นว่าเมื่อ epoch เพิ่มขึ้นค่า loss จะลดลงและ accuracy ก็สูงขึ้นทั้งการ train และ validation ภาพประกอบที่ 28 แสดง confusion matrix ซึ่งจะเห็นว่ามีประสิทธิภาพดีในทุกคลาส

Based Model จะนำไปใช้ต่อในการทดลอง Exploring SoftMax Probability Score, 2-Layer Classification (OC-SVM) 2-Layer Classification (Isolation Forest)

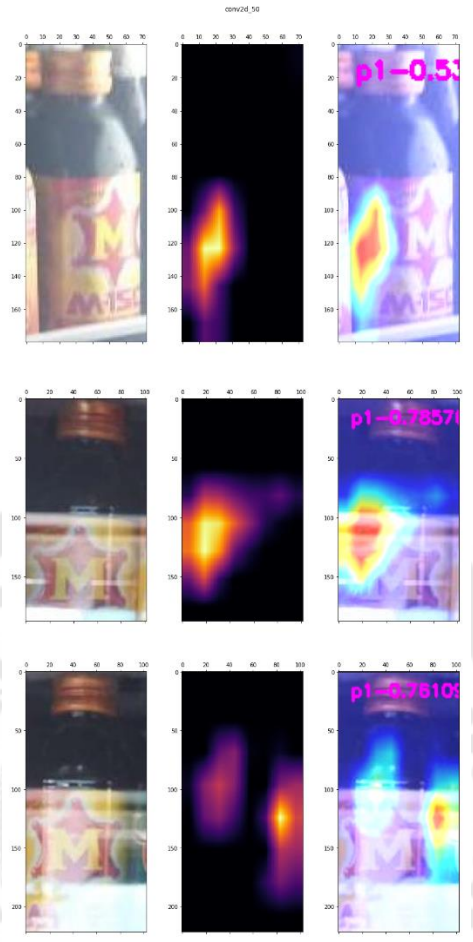


ภาพประกอบ 27 train loss, validation loss / epoch และ train accuracy, validation accuracy/epoch



ภาพประกอบ 28 Confusion Matrix ของ Based Multi-class model

ภาพประกอบ 29 Heatmap แสดงจุดสำคัญของภาพที่แบบจำลองสกัดได้ ของภาพที่ทำนายถูกต้อง



ภาพประกอบ 30 Heatmap แสดงจุดสำคัญของภาพที่แบบจำลองสกัดได้ ของภาพที่ทำนายผิด

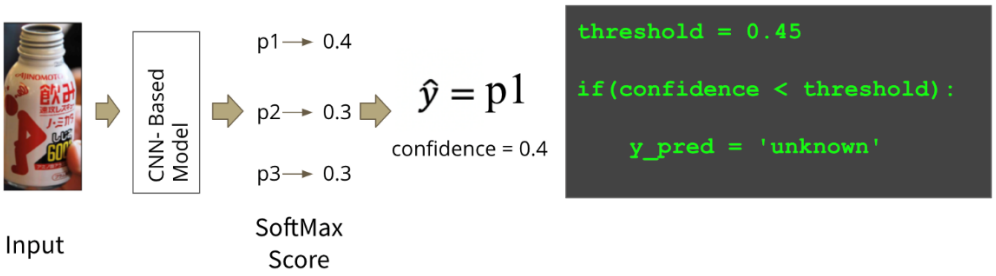
ภาพประกอบที่ 29 -30 แสดงให้เห็นถึงบริเวณที่สำคัญที่มีผลต่อการจำแนกโดยการนำ Activation map ที่ได้จากชั้นคอนโวลูชันสุดท้ายมาสร้างเป็น heatmap โดยบริเวณที่สว่างของภาพใดๆคือบริเวณที่มีความสำคัญที่แบบจำลองสกัด จากภาพประกอบที่ 29 ซึ่งเป็นภาพที่แบบจำลองทำนายได้ถูกต้องในขั้นตอนการทดสอบจะเห็นว่าบริเวณที่มีความสว่างสูงคือบริเวณฉลากของเครื่องดื่ม ส่วนภาพประกอบที่ 30 เป็นภาพที่แบบจำลองทำนายผิดในขั้นตอนการทดสอบ เมื่อพิจารณา heatmap แล้วจะเห็นว่าบริเวณที่สว่างจะไม่ใช่บริเวณฉลาก

4.2 Exploring SoftMax Probability Score

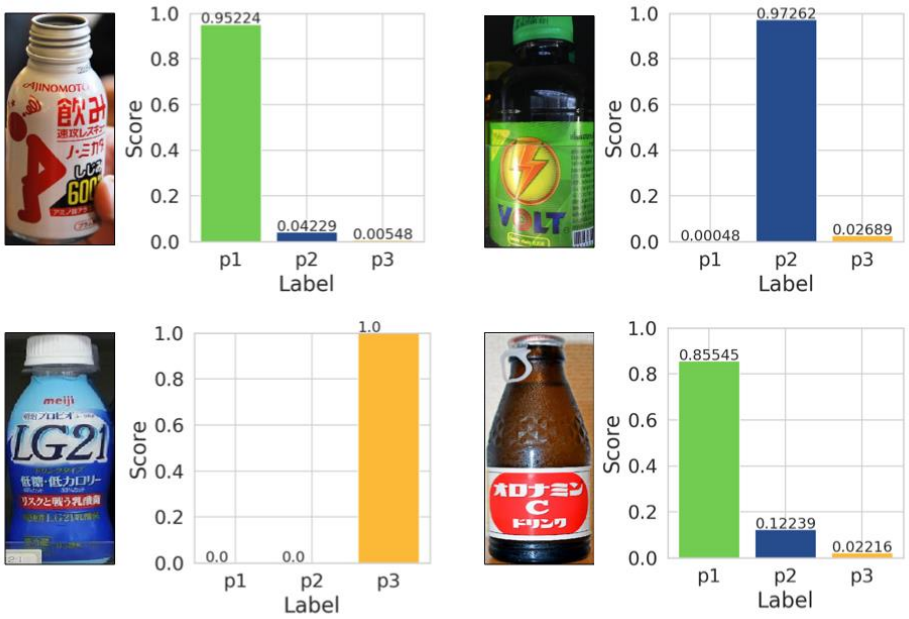
ในการทดลองนี้เป็นการนำ Based Model(Multi-Class Model) ซึ่งให้ความแม่นยำที่สูงกับข้อมูลคลาสหลัก มาทำการทดสอบเพื่อทำการสำรวจการกระจายคะแนนความเชื่อมั่น เนื่องจาก Softmax Function จะถูกใช้ในการหาความเชื่อมั่นหรือคะแนนความน่าจะเป็นว่าภาพขวดใดๆ จะทำนายเป็นคลาสใดได้แก่ $p1, p2, p3$ และคลาสใดที่มีค่าความน่าจะเป็นสูงที่สุดก็จะเป็นคำตอบ ซึ่งคะแนนความน่าจะเป็นของทุกคลาสจะมีผลรวมเป็น 1 เสมอ

ดังนั้นการทดลองนี้จะถูกออกแบบโดยมีจุดประสงค์เพื่อสำรวจคะแนนความน่าจะเป็น ที่ได้จาก Softmax Function ของภาพกลุ่ม unknown class ซึ่งภาพเหล่านี้จะเป็นภาพที่ไม่ได้เป็นภาพคลาสใด ๆ ทั้ง $p1, p2, p3$

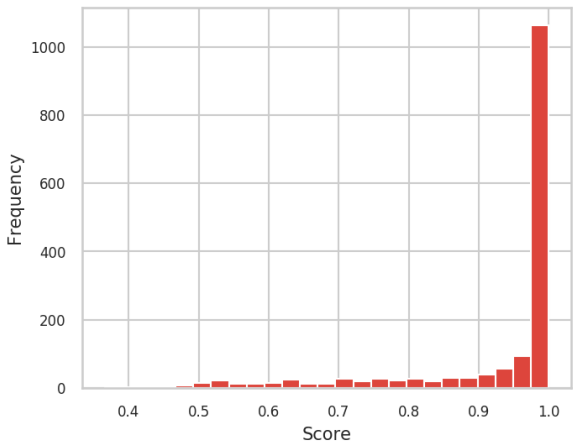
สมมติฐานของการทดลองนี้ก็คือ เมื่อทำนายภาพกลุ่ม unknown class คะแนนความน่าจะเป็นของแต่ละคลาสจะแตกต่างกันไม่มาก มีการกระจายเท่าๆกัน ถ้าหากผลที่ได้เป็นไปตามสมมติฐานข้างต้น ก็จะสามารถหาช่วงคะแนนความน่าจะเป็นที่เหมาะสมเพื่อทำ threshold ในการแยก unknown class ได้ ดังที่แสดงในภาพประกอบที่ 31 คือตัวอย่างการกระจายคะแนนตามแบบอุดมคติ ที่สามารถกำหนดค่า threshold ได้ง่าย อย่างไรก็ตามคะแนนความน่าจะเป็นที่ได้จริงไม่เป็นไปตามสมมติฐานดังที่แสดงในภาพประกอบที่ 32,33 และตารางที่ 3



ภาพประกอบ 31 การกำหนดค่า threshold เพื่อระบุ unknown class ในอุดมคติ



ภาพประกอบ 32 ตัวอย่างภาพขวดและการกระจายของคะแนนความน่าจะเป็นของภาพ unknown class



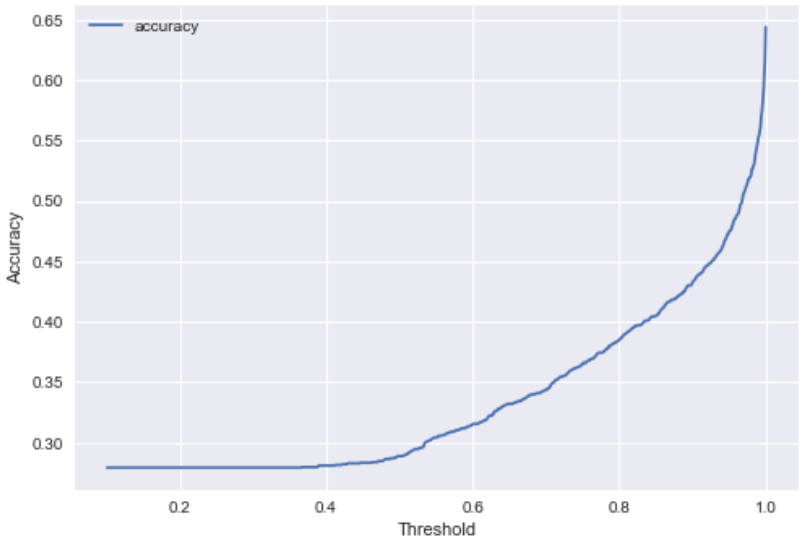
ภาพประกอบ 33 การกระจายของคะแนนของภาพกลุ่ม unknown

ตาราง 3 แสดงสถิติของคะแนนความเชื่อมั่น ของภาพ unknown class จำนวน 1,592 ภาพ

count	1,592
mean	0.929013
std	0.132177
min	0.365441
25%	0.935618
50%	0.998164
75%	0.999943
max	1

ภาพประกอบที่ 32 แสดงตัวอย่างของภาพขวดของภาพขวดในกลุ่ม unknown class โดยจะเห็นว่าคะแนนไม่ได้มีการกระจายไปยังคลาส $p1, p2, p3$ เท่าๆกัน แต่คะแนนความน่าจะเป็นจะสูงมากที่คลาสใดคลาสหนึ่งเท่านั้น การกระจายของคะแนนไม่เป็นไปตามสมมติฐานที่วางไว้

ภาพประกอบที่ 33 และตารางที่ 4 แสดงให้เห็นการกระจายคะแนนความน่าจะเป็น ของภาพขวดในกลุ่ม unknown class ทั้งหมด 1,592 ภาพซึ่งภาพเหล่านี้เป็นภาพที่ไม่ได้อยู่ในคลาสใด จะเห็นว่าคะแนนความเชื่อมั่นส่วนใหญ่จะเกิน 99% ซึ่งเป็นคะแนนความเชื่อมั่นที่สูงมาก จากผลการทดลองข้างต้น สามารถสรุปได้ว่า ภาพ unknown class นั้นถูกทำนายเป็นภาพ known class ซึ่งเป็นการทำนายที่ผิดด้วยระดับคะแนนความเชื่อมั่นที่สูงมาก



ภาพประกอบ 34 แสดงความแม่นยำในการจำแนก unknow และ known class ที่ระดับ threshold ตั้งแต่ 0.1 – 1.0

เมื่อกำหนดค่า threshold ในช่วง 0.1-1.0 เพื่อทำการแยก ภาพ unknown class และ known class (p1,p2,p3) หากคะแนนความเชื่อมั่นของภาพใดมีค่าต่ำกว่าค่า threshold ที่ตั้งไว้ก็จะถูกจำแนกเป็น unknown class ผลการทดลองแสดงให้เห็นในภาพประกอบที่ 33 จะเห็นว่าการกำหนดค่า thresholdที่ 1.0 ซึ่งเป็นค่าความเชื่อมั่นสูงสุด จะสามารถแยก unknown class และ known class ที่ความแม่นยำ (Accuracy) 64% เท่านั้น

ดังนั้นสามารถสรุปได้ว่าการกำหนด threshold ยังมีความแม่นยำไม่เพียงพอต่อการจำแนก unknow และ known class

4.3 N-Binary Classifier

ในการทดลองนี้จะทำการสร้าแบบจำลอง 3 แบบจำลอง จากชุดข้อมูลเครื่องดื่มโดยมีรายละเอียดดังตารางที่ 5 โดยทุกแบบจำลองจะใช้สถาปัตยกรรมแบบ Resnet50 โดยมีการกำหนด hype-parameter เป็น batch size = 32, epoch = 25, optimizer = SGD, learning rate = 0.1, Momentum = 0.9, Weight decay = 0.004, Loss function = binary-cross entropy loss

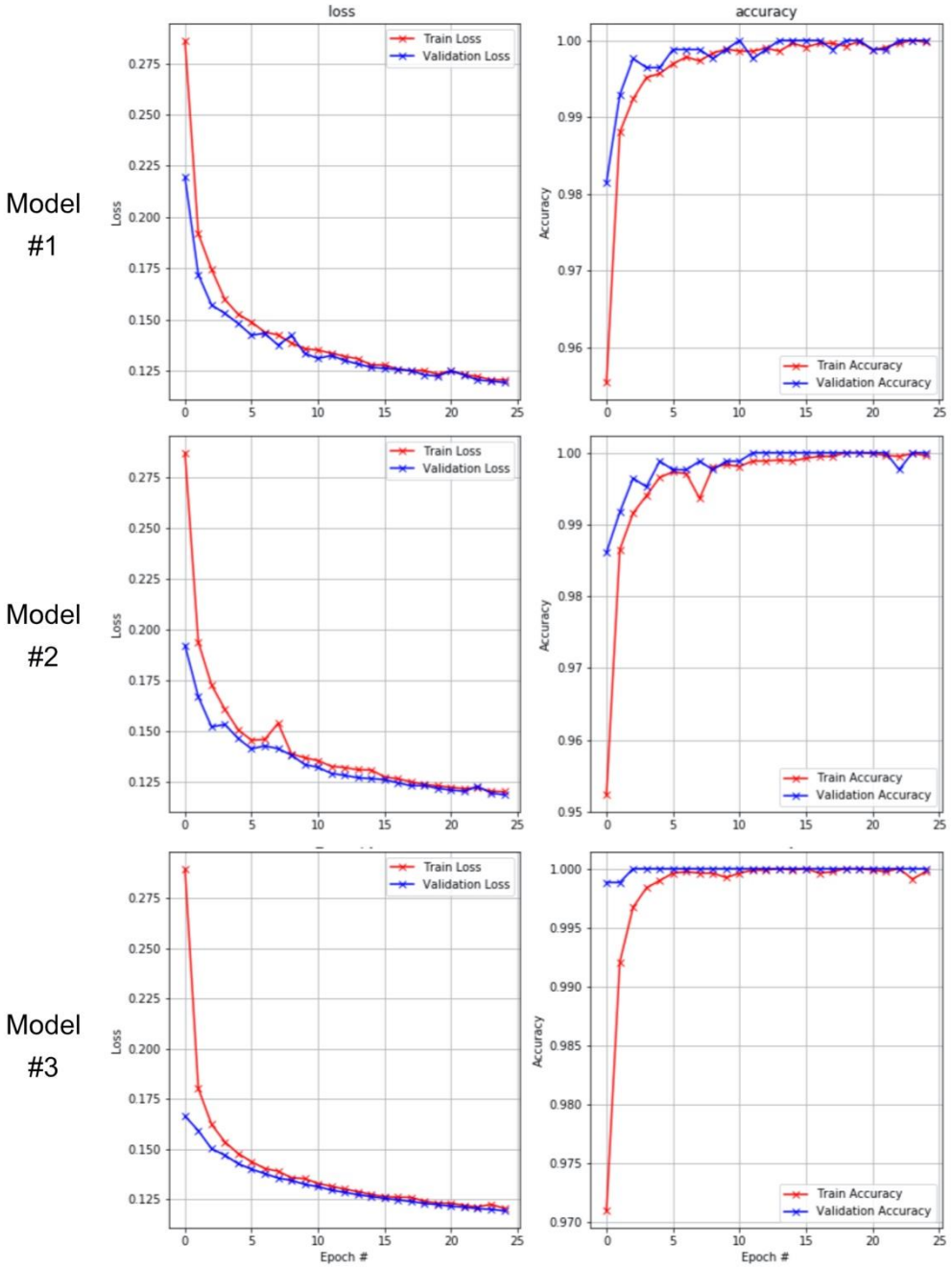
ตาราง 4 N-Binary Classifier Training Procedure

#	คลาสที่ใช้ฝึกฝน	วัตถุประสงค์
Model 1	p1, other	จำแนกว่าเป็น p1 หรือ ไม่ใช่ p1 $\hat{y} \in \{p1, \sim p1\}$
Model 2	p2, other	จำแนกว่าเป็น p2 หรือ ไม่ใช่ p2 $\hat{y} \in \{p2, \sim p2\}$
Model 3	p3, other	จำแนกว่าเป็น p3 หรือ ไม่ใช่ p3 $\hat{y} \in \{p3, \sim p3\}$

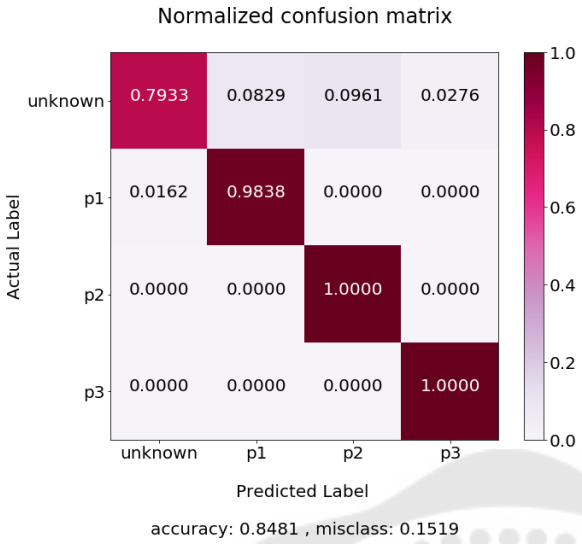
สุดท้าย N-Binary Classifier จะใช้ทั้ง Model1, Model2, Model3 ร่วมกันในการทำนายภาพขวดเครื่องดื่มโดยใช้ Decision Function ตามที่แสดงในสมการที่ 21 โดยผลการทายจะเป็น $\hat{y} \in \{p1, p2, p3, \text{unknown}\}$

$$\hat{y} = \begin{cases} unknown, & \text{ทุกแบบจำลองจำแนกเป็น } negative \\ p1, & \text{แบบจำลอง 1 จำแนกเป็น } postive \text{ และได้คะแนนสูงสุด} \\ p2, & \text{แบบจำลอง 2 จำแนกเป็น } postive \text{ และได้คะแนนสูงสุด} \\ p3, & \text{แบบจำลอง 3 จำแนกเป็น } postive \text{ และได้คะแนนสูงสุด} \end{cases} \quad (21)$$

ภาพประกอบที่ 35 แสดง loss และ accuracy ในการ train และ validation ของ Model1, Model2, Model3 โดยทั้ง 3 Model มีผลที่แสดงให้เห็นแนวโน้มที่สอดคล้องกันคือ เมื่อจำนวน epoch เพิ่มขึ้น ค่า loss จะลดและ accuracy จะเพิ่มขึ้น



ภาพประกอบ 35 train loss, validation loss / epoch และ train accuracy, validation accuracy/epoch ของ N-Binary Classifier แต่ละแบบจำลอง



	precision	recall	F1
unknown	0.998	0.793	0.934
p1	0.580	0.984	0.729
p2	0.581	1	0.735
p3	0.816	1	0.899

ภาพประกอบ 36 Classification Report ของ N-Binary Classifier

ผลการทดลอง N-Binary Classifier แสดงให้ประสิทธิภาพดังที่แสดงในภาพประกอบที่ 36 โดยมีค่า accuracy โดยรวมคือ 84% และหากพิจารณารายละเอียดในการจำแนกคลาสต่างๆโดยอาศัยค่า precision, recall และ f1 ร่วมด้วย จะสามารถสรุปได้ดังต่อไปนี้

(1) ประสิทธิภาพการจำแนก unknown มาจากการพิจารณาค่า recall ซึ่งจะเห็นว่าสามารถตรวจหาภาพ unknown class ได้ 79% และหากพิจารณาค่า precision แล้วสามารถสรุปได้ว่าภาพที่ถูกทำนายเป็น unknown ทั้งหมดมีความถูกต้องที่ 99% เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 93%

(2) ประสิทธิภาพในการจำแนก p1 พิจารณาจากการที่สามารถตรวจหา p1 ด้ 98% โดยพิจารณาจากค่า recall และ ค่า precision ที่ 58% แสดงให้เห็นความถูกต้องของภาพทั้งหมดที่ถูกจำแนกเป็น p1 เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 72%

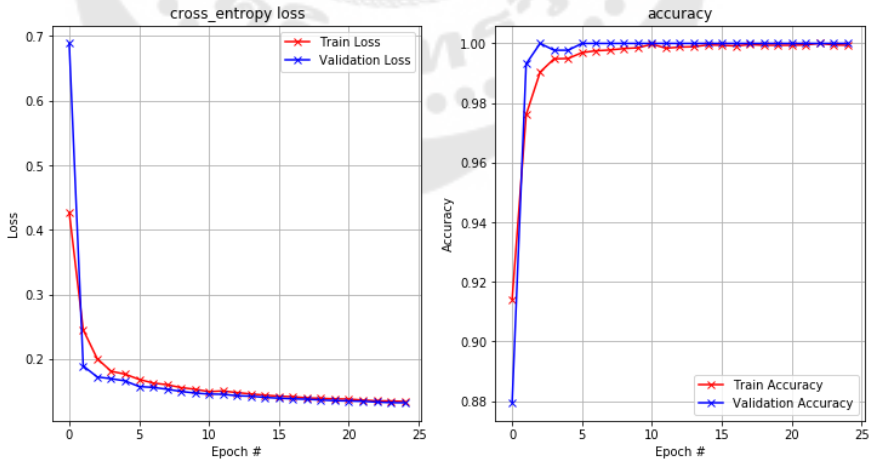
(3) ประสิทธิภาพในการจำแนก p2 พิจารณจากความสามารถตวรจหา p2 ได้ 100% โดยดูจากค่า recall และความถูกต้องของภาพทั้งหมดที่ถูกจำแนกเป็น p2 มาจากค่า precision ที่ 58% เท่านั้น เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 73%

(4) ประสิทธิภาพในการจำแนก p3 พิจารณาจากค่า recall ที่ได้ 100% แสดงให้เห็นถึงประสิทธิภาพในการตรวจหา p3 ได้ทั้งหมด แต่เมื่อพิจารณาค่า precession ที่ 100% จะเห็นว่าภาพที่ถูกทำนายเป็น p3 จะมีความถูกต้องที่ 81% เท่านั้น เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 89 %

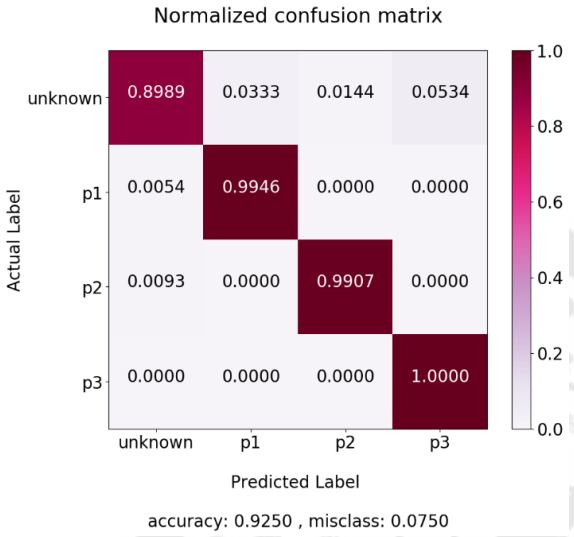
4.4 N+Unknown Class Classifier

ในการทดลองนี้ N+Unknown Class Classifier จะถูกสร้างขึ้นมาโดยใช้ภาพขวดเครื่องดื่ม 3 คลาสหลักคือ $p1, p2, p3$ ร่วมกับภาพ unknown class แบบจำลองจะถูกสร้างโดยใช้สถาปัตยกรรมแบบ ResNet50 มีการกำหนด hyperparameter เป็น batch size = 32, epoch = 25, optimizer = SGD, learning rate = 0.1, Momentum = 0.9, Weight decay = 0.004, Loss function = cross entropy loss โดยโมเดลที่ได้จะให้ผลการทำนายเป็น $\hat{y} \in \{p1, p2, p3, \text{unknown}\}$

ภาพประกอบที่ 37 แสดง loss และ accuracy ในการ train และ validation โดยผลที่ได้จค่า loss จะลดลงและค่า accuracy สูงขึ้นตามจำนวน epoch ที่เพิ่มขึ้น และ accuracy ที่ 25 epoch มีค่าสูงกว่า 99%



ภาพประกอบ 37 train loss, validation loss / epoch และ train accuracy, validation accuracy/epoch ของ N+Unknown Classifier



	precision	recall	F1
unknown	0.998	0.899	0.953
p1	0.776	0.995	0.872
p2	0.902	0.991	0.944
p3	0.696	1	0.821

ภาพประกอบ 38 Classification Report ของ N+Unknown Class Classifier

ผลการทดลองที่ได้แสดงในภาพประกอบที่ 38 ค่า accuracy โดยรวมคือ 92% และหากพิจารณารายละเอียดในการจำแนกคลาสต่างๆโดยอาศัยค่า precision, recall และ f1 ร่วมด้วย จะสามารถสรุปได้ดังต่อไปนี้

(1) ประสิทธิภาพการจำแนก unknown มาจากการพิจารณาค่า recall ซึ่งจะเห็นว่าสามารถตรวจหาภาพ unknown class ได้ 89% และหากพิจารณาค่า precision แล้วสามารถสรุปได้ว่าภาพที่ถูกทำนายเป็น unknown ทั้งหมดมีความถูกต้องที่ 99% เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 95%

(2) ประสิทธิภาพในการจำแนก p1 พิจารณาจากการที่สามารถตรวจหา p1 ได้ 99% โดยพิจารณาจากค่า recall และ ค่า precision ที่ 77% แสดงให้เห็นความถูกต้องของภาพทั้งหมดที่ถูกจำแนกเป็น p1 เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 87%

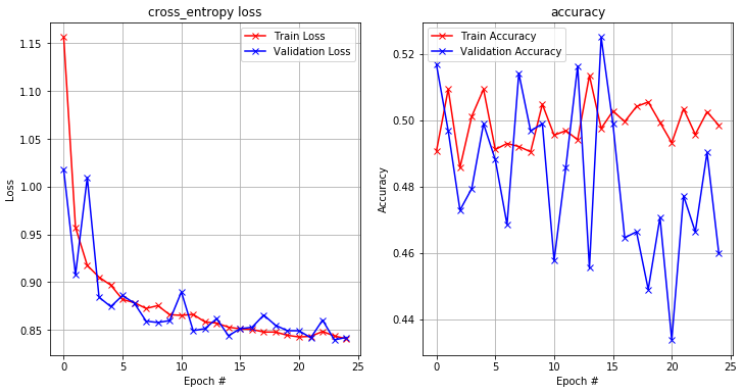
(3) ประสิทธิภาพในการจำแนก p2 พิจารณจากความสามารถตวรจหา p2 ได้ 99% โดยพิจารณาจากค่า recall และความถูกต้องของภาพทั้งหมดที่ถูกจำแนกเป็น p2 มาจากค่า precision ที่ 99% เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 94%

(4) ประสิทธิภาพในการจำแนก p3 พิจารณาจากค่า recall ที่ 100% แสดงให้เห็นถึงประสิทธิภาพในการตรวจหา p3 ได้ทั้งหมด แต่เมื่อพิจารณาค่า precession ที่ 69% จะเห็นว่าภาพที่ถูกทำนายเป็น p3 จะมีความถูกต้องที่ 69% เท่านั้น เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 82 %

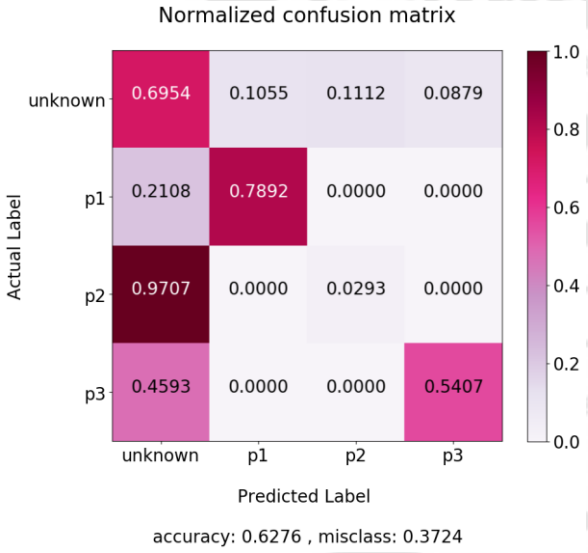
4.5 N+Combination Class Classifier

ในการทดลองนี้ N+Combination Class Classifier จะถูกสร้างขึ้นมาโดยใช้ภาพขวดเครื่องดื่ม 3 คลาสหลักคือ $p1, p2, p3$ ร่วมกับภาพกลุ่ม Combination ซึ่งมาจากการนำภาพ $p1, p2, p3$ มารวมกันเป็นคลาสใหม่เรียกว่า combination class แบบจำลองจะถูกสร้างโดยใช้สถาปัตยกรรมแบบ ResNet50 มีการกำหนด hyperparameter เป็น batch size = 32, epoch = 25, optimizer = SGD, learning rate = 0.1, Momentum = 0.9, Weight decay = 0.004, Loss function = cross entropy loss โดยโมเดลที่ได้จะให้ผลการทำนายเป็น $\hat{y} \in \{p1, p2, p3, \text{unknown}\}$

loss และ accuracy ของการ train และ validation แสดงในภาพประกอบที่ 39 ซึ่งจะเห็นว่าค่าที่ได้จะลดลงหรือเพิ่มขึ้นไม่แน่นอน หากพิจารณาค่า train accuracy จะต่ำกว่า validation accuracy เมื่อจำนวน epoch เพิ่มขึ้นซึ่งแสดงให้เห็นแนวโน้มของการที่แบบจำลองมีการเรียนรู้มากเกินไป (overfit) โดยผู้วิจัยได้สันนิษฐานการเกิด overfit ว่ามีสาเหตุมาจาก Combination class ล้วนเป็นภาพที่ซ้ำกับภาพที่อยู่ใน $\{p1, p2, p3\}$ ทั้งสิ้น



ภาพประกอบ 39 train loss, validation loss / epoch และ train accuracy, validation accuracy/epoch ของ N+Combination class classifier



	precision	recall	F1
unknown	0.766	0.695	0.774
p1	0.489	0.789	0.604
p2	0.033	0.029	0.030
p3	0.447	0.541	0.489

ภาพประกอบ 40 Classification Report ของ N+Combination class classifier

ภาพประกอบที่ 40 แสดงประสิทธิภาพของ N+Combination Class Classifier โดยมีค่า accuracy โดยรวมคือ 62% และหากพิจารณารายละเอียดในการจำแนกคลาสต่างๆโดยอาศัยค่า precision, recall และ f1 ร่วมด้วย จะสามารถสรุปได้ดังต่อไปนี้

(1) ประสิทธิภาพการจำแนก unknown พิจารณาได้จากค่า recall ที่แสดงให้เห็นว่าสามารถตรวจหาภาพประเภท unknown class ได้ 69% และหากพิจารณาค่า precision แล้วสามารถสรุปได้ว่าภาพที่ถูกทำนายเป็น unknown ทั้งหมดมี

ความถูกต้องที่ 76% เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 77%

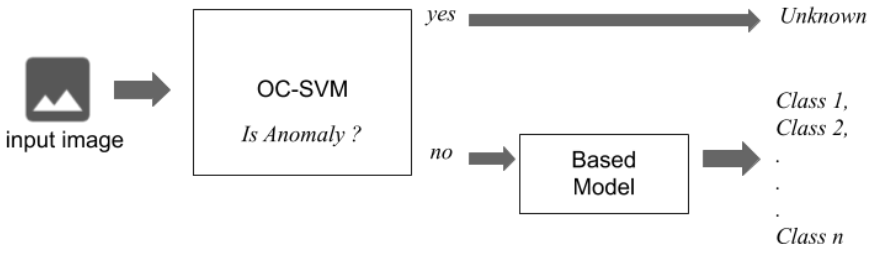
(2) ประสิทธิภาพในการจำแนก p1 พิจารณาจากการที่สามารถตรวจหา p1 ได้ 78% โดยดูจากค่า recall ส่วนค่า precision จะแสดงให้เห็นความถูกต้องของภาพทั้งหมดที่ถูกจำแนกเป็น p1 ซึ่งทำได้เพียง 48% เท่านั้น เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 60%

(3) ประสิทธิภาพในการจำแนก p2 พิจารณาจากความสามารถตวรจหา p2 ได้ 2.9% เท่านั้นโดยพิจารณาจากค่า recall ส่วนความถูกต้องของภาพทั้งหมดที่ถูกจำแนกเป็น p2 มาจากค่า precision ที่ 3.3% เท่านั้น และเมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 3%

(4) ประสิทธิภาพในการจำแนก p3 พิจารณาจากค่า recall แสดงให้เห็นถึงประสิทธิภาพในการตรวจหา p3 ที่ 54% แต่เมื่อพิจารณาค่า precession ที่ 44% จะเห็นว่าภาพที่ถูกทำนายเป็น p3 จะมีความถูกต้องที่ 44% เท่านั้น เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 48 %

4.6 2-Layer Classification (OC-SVM and Multi-Class Model)

การทดลอง 2-Layer Classification (OC-SVM and Multi-Class Model) จะใช้ OC-SVM เพื่อแยกภาพ unknown class และ known class หากภาพใดที่ OC-SVM ทำนายเป็น known จะต้องนำไปจำแนกต่อโดยใช้ Multi-Class Model ซึ่งได้มาจาก Based Model (จากการทดลอง 4.1) เพื่อจำแนกภาพออกเป็นคลาสต่างๆดังที่แสดงในภาพประกอบที่ 41



ภาพประกอบ 41 2-Layer Classification (OC-SVM and Multi-Class Model) prediction procedure

ในการสร้างแบบจำลองโดยใช้ OC-SVM ชุดข้อมูลที่ใช้จะมาจากการใช้ Keras Pre-trained Resnet50 เพื่อทำการสกัดฟีเจอร์โดยใช้ข้อมูลภาพคลาสหลักคือ $p1, p2, p3$ ได้ จากนั้นนำฟีเจอร์ที่สกัดได้ไปใช้ฝึกฝนแบบจำลอง

ฟีเจอร์ที่สกัดได้มีทั้งหมด 1000 ฟีเจอร์จากภาพ known class ทั้งหมด 3,083 ภาพ ภาพประกอบที่ 42 แสดงตัวอย่างข้อมูลของฟีเจอร์ที่สกัดได้

	0	1	2	3	4	5	...	999
0	9.630594e-07	1.729456e-06	1.366023e-06	8.322789e-07	1.506345e-06	1.190922e-06	...	1.987513e-04
1	1.000958e-07	1.733838e-05	1.135899e-07	2.765273e-08	1.776408e-07	5.361180e-06	...	2.786681e-05
2	1.254831e-07	4.765277e-06	1.885328e-07	3.264021e-08	1.550844e-07	6.787843e-07	...	2.428988e-05
3	1.143332e-05	1.586538e-04	2.638514e-06	2.144318e-07	1.460799e-06	1.590859e-05	...	2.411850e-03
4	1.405296e-07	3.318228e-07	6.143244e-07	1.961726e-07	8.697195e-07	4.984757e-07	...	7.652139e-04
5	3.398011e-06	1.510083e-05	2.521010e-06	6.836207e-07	2.780532e-06	1.504976e-05	...	6.129723e-05
...	...	...	...	...	...	...	...	...

ภาพประกอบ 42 ตัวอย่างฟีเจอร์ที่สกัดได้จาก Pre-trained ResNet50

ฟีเจอร์ทั้งหมดจะนำไปเทรน OC-SVM โดยกำหนด hyperparameter เป็น nu = 0.5, kernel = rbf, gamma = 0.1 ภาพประกอบที่ 43 แสดงประสิทธิภาพของ 2-Layer Classification (OC-SVM + Based Model) โดยมีค่า accuracy โดยรวมคือ 71% และหากพิจารณารายละเอียดในการจำแนกคลาสต่างๆโดยอาศัยค่า precision, recall และ f1 ร่วมด้วย จะสามารถสรุปได้ดังต่อไปนี้

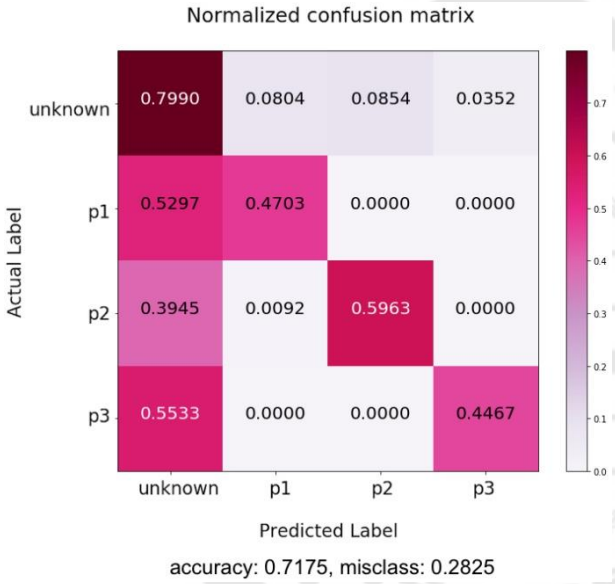
(1) ประสิทธิภาพการจำแนก unknown พิจารณาได้จากค่า recall ที่แสดงให้เห็นว่าสามารถตรวจหาภาพประเภท unknown class ได้ 79% และหากพิจารณาค่า precision แล้วสามารถสรุปได้ว่าภาพที่ถูกทำนายเป็น unknown ทั้งหมดมีความถูกต้องที่ 81% เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 84%

(2) ประสิทธิภาพในการจำแนก p1 พิจารณาจากการที่สามารถตรวจหา p1 ได้ 47% โดยดูจากค่า recall ส่วนค่า precision จะแสดงให้เห็นความถูกต้องของภาพทั้งหมดที่ถูกจำแนกเป็น p1 ซึ่งทำได้เพียง 42% เท่านั้น เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 44%

(3) ประสิทธิภาพในการจำแนก p2 พิจารณจากความสามารถตวรจหา p2 ได้ 60% โดยพิจารณาจากค่า recall ส่วนความถูกต้องของภาพทั้งหมดที่ถูกจำแนกเป็น

p2 มาจากค่า precision ที่ 49% และเมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 54%

(4) ประสิทธิภาพในการจำแนก p3 พิจารณาจากค่า recall แสดงให้เห็นถึงประสิทธิภาพในการตรวจหา p3 ที่ 44% แต่เมื่อพิจารณาค่า precession ที่ 44% จะเห็นว่าภาพที่ถูกทำนายเป็น p3 จะมีความถูกต้องที่ 61% เท่านั้น เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 51%



	precision	recall	F1
unknown	0.810	0.798	0.840
p1	0.423	0.475	0.448
p2	0.493	0.606	0.543
p3	0.611	0.447	0.516

ภาพประกอบ 43 Classification Report ของ OC-SVM and Multi-class

4.7 2-Layer Classification (Isolation Forest and Multi-Class Model)

การทดลอง 2-Layer Classification (Isolation Forest and Multi-Class Model) จะใช้ Isolation Forest เพื่อแยกภาพ unknown class และ known class หากภาพใดที่ Isolation Forest ทำนายเป็น known จะต้องนำไปจำแนกต่อโดยใช้ Multi-Class Model ซึ่งได้มาจาก Based Mode จากการทดลองที่ 4.1 เพื่อจำแนกภาพออกเป็นคลาสต่างๆ ในขั้นตอนการฝึกฝนแบบจำลองก็ใช้ฟีเจอร์เดียวกันกับการทดลองก่อหน้านี้ โดยมี hyperparameter เป็น behavior = new, max_samples =10, contamation = auto, randomstate = 42

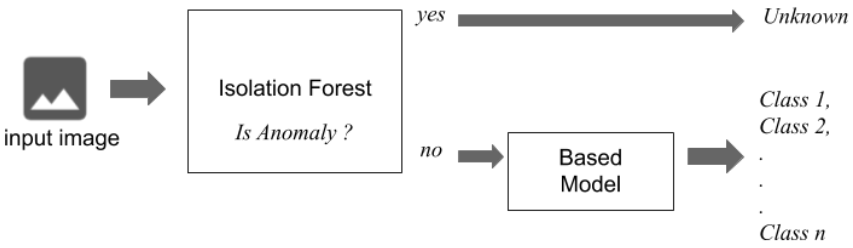
ผลการทดลองที่ได้จะเห็นว่าประสิทธิภาพค่อนข้างต่ำดังที่ได้แสดงในภาพประกอบที่ 45 โดยค่า accuracy โดยรวมคือ 29% และหากพิจารณารายละเอียดในการจำแนกคลาสต่างๆโดยอาศัยค่า precision, recall และ f1 ร่วมด้วย จะสามารถสรุปได้ดังต่อไปนี้

(1) ประสิทธิภาพการจำแนก unknown มาจากการพิจารณาค่า recall ซึ่งจะเห็นว่าสามารถตรวจหาภาพ unknown class ได้ 12% และหากพิจารณาค่า precision แล้วสามารถสรุปได้ว่าภาพที่ถูกทำนายเป็น unknown ทั้งหมดมีความถูกต้องที่ 55% เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 29%

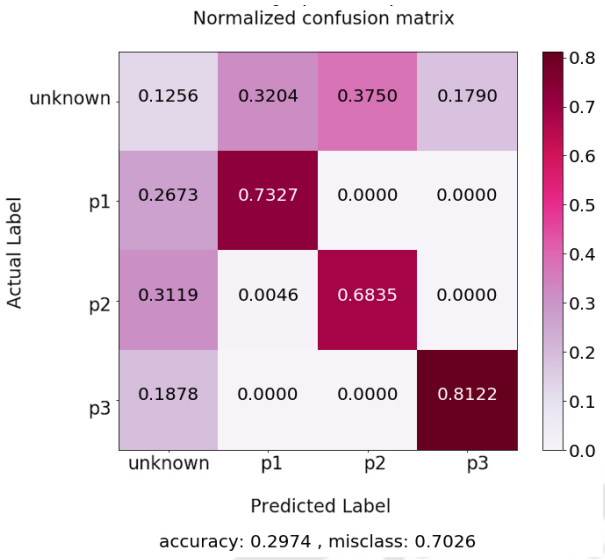
(2) ประสิทธิภาพในการจำแนก p1 พิจารณาจากการที่สามารถตรวจหา p1 ได้ 73% โดยพิจารณาจากค่า recall และ ค่า precision ที่ 22% แสดงให้เห็นความถูกต้องของภาพทั้งหมดที่ถูกจำแนกเป็น p1 เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 34%

(3) ประสิทธิภาพในการจำแนก p2 พิจารณจากความสามารถตวรจหา p2 ได้ 68% โดยพิจารณาจากค่า recall และความถูกต้องของภาพทั้งหมดที่ถูกจำแนกเป็น p2 มาจากค่า precision ที่ 20% เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 30%

(4) ประสิทธิภาพในการจำแนก p3 พิจารณาจากค่า recall ที่ 81% แสดงให้เห็นถึงประสิทธิภาพในการตรวจหา p3 แต่เมื่อพิจารณาค่า precession ที่ 36% จะเห็นว่าภาพที่ถูกทำนายเป็น p3 จะมีความถูกต้องที่ 36% เท่านั้น เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 49 %



ภาพประกอบ 44 2-Layer Classification (Isolation Forest + Based Model)



	precision	recall	F1
unknown	0.557	0.126	0.295
p1	0.225	0.733	0.344
p2	0.200	0.683	0.309
p3	0.360	0.812	0.498

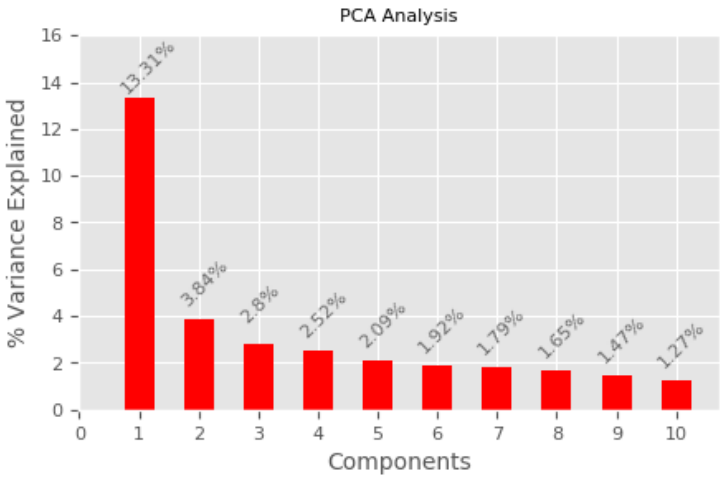
ภาพประกอบ 45 Classification Report ของ Isolation forest and Multi-class

4.8 Dimension Reduction

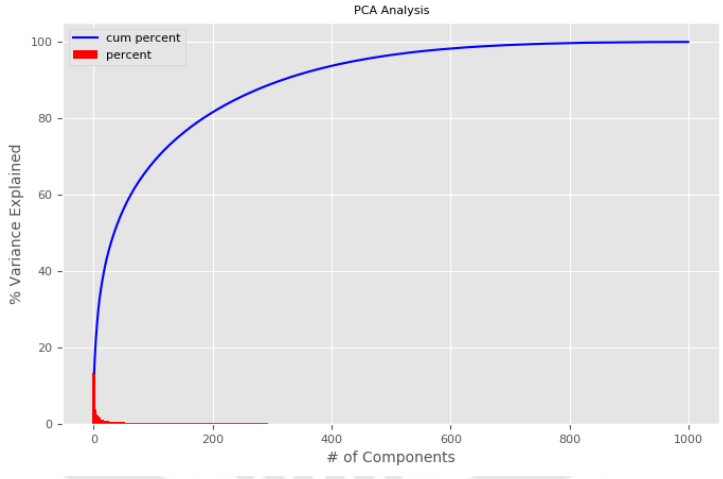
การทดลอง OC-SVM และ Isolation Forest ยังให้ประสิทธิภาพที่ไม่ดีนัก ดังนั้นในขั้นต่อไปจะเป็นการปรับปรุงประสิทธิภาพ โดยหากพิจารณาที่จำนวนฟีเจอร์ที่สกัดได้ซึ่งมีทั้งหมด 1000 ฟีเจอร์จากภาพ known class ทั้งหมด 3,083 ภาพ จะเห็นว่าจำนวนมิติของของมูลสูงมาก ผู้วิจัยจึงได้ใช้การวิเคราะห์องค์ประกอบหลัก (PCA) เพื่อทำการลดมิติของข้อมูล

ภาพประกอบที่ 46 แสดงให้เห็น PC1 จะมีครอบคลุม variance สูงสุดที่ 13.31. % และ PC2-PC10 ก็จะมี variance ลดลงตามลำดับ และภาพประกอบที่ 47 แสดงให้เห็นว่าเพียงแค่ 200 component ก็ครอบครุม variance เกิน 80%

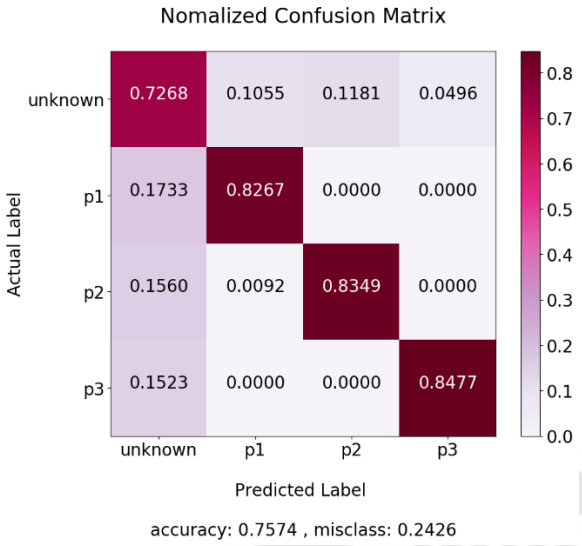
จากผลการวิเคราะห์องค์ประกอบหลัก ผู้วิจัยได้เลือกการลดมิติข้อมูลโดยใช้แค่ 400 component ซึ่งสามารถอธิบาย variance ได้ 93% หลังจากลดมิติข้อมูลแล้วก็ได้นำข้อมูลที่ได้เพื่อไปสร้างโมเดลโดยใช้ OC-SVM, และ Isolation Forest และทำ hyperparameter tuning



ภาพประกอบ 46 แสดง variance/components

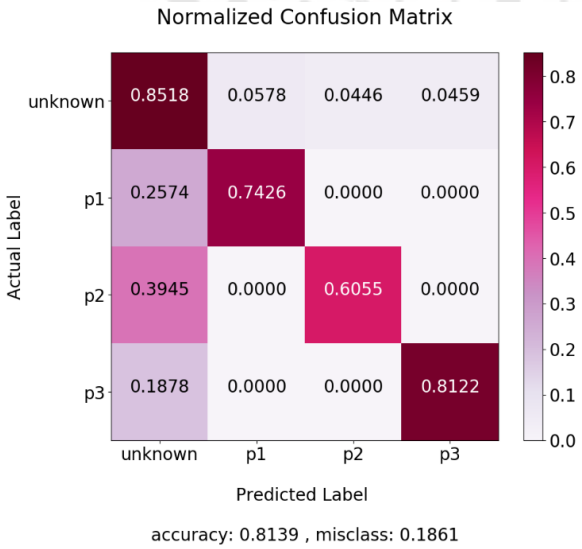


ภาพประกอบ 47 กราฟแสดง variance/จำนวน component



	precision	recall	F1
unknown	0.921	0.727	0.870
p1	0.493	0.825	0.617
p2	0.492	0.835	0.619
p3	0.679	0.848	0.754

ภาพประกอบ 48 Classification Report หลังทำ PCA ของ OC-SVM and Multi-class



	precision	recall	F1
unknown	0.885	0.852	0.8884
p1	0.619	0.748	0.6771
p2	0.652	0.601	0.6253
p3	0.688	0.807	0.7430

ภาพประกอบ 49 Classification Report หลังทำ PCA ของ Isolation Forest and Multi-class

หลังจากทำ PCA ผลการทดลองที่ได้จาก Multi-class and Isolation Forest จะเห็นว่าประสิทธิภาพดีขึ้นในทุกตัวชี้วัด จากภาพประกอบที่ 49 แสดงให้เห็น accuracy โดยรวมที่ 85% และหากพิจารณารายละเอียดในการจำแนกคลาสต่างๆโดยอาศัยค่า precision, recall และ f1 ร่วมด้วย จะสามารถสรุปได้ดังต่อไปนี้

(1) ประสิทธิภาพการจำแนก unknown พิจารณาได้จากค่า recall ที่แสดงให้เห็นว่าสามารถตรวจหาภาพประเภท unknown class ได้ 85% และหากพิจารณาค่า precision แล้วสามารถสรุปได้ว่าภาพที่ถูกทำนายเป็น unknown ทั้งหมดมีความถูกต้องที่ 88% เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 88%

(2) ประสิทธิภาพในการจำแนก p1 พิจารณาจากการที่สามารถตรวจหา p1 ได้ 74% โดยดูจากค่า recall ส่วนค่า precision จะแสดงให้เห็นความถูกต้องของภาพทั้งหมดที่ถูกจำแนกเป็น p1 ซึ่งทำได้ 61% เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 67%

(3) ประสิทธิภาพในการจำแนก p2 พิจารณจากความสามารถตวรจหา p2 ได้ 60% โดยพิจารณาจากค่า recall ส่วนความถูกต้องของภาพทั้งหมดที่ถูกจำแนกเป็น p2 มาจากค่า precision ที่ 65% และเมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 62%

(4) ประสิทธิภาพในการจำแนก p3 พิจารณาจากค่า recall แสดงให้เห็นถึงประสิทธิภาพในการตรวจหา p3 ที่ 80% แต่เมื่อพิจารณาค่า precession ที่ 44% จะเห็นว่าภาพที่ถูกทำนายเป็น p3 จะมีความถูกต้องที่ 68% เท่านั้น เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 74 %

ภาพประกอบที่ 48 แสดงประสิทธิภาพของ Multi-class and OC-SVM ที่ให้ประสิทธิภาพที่ดีขึ้นลงในภาพกลุ่ม known class โดยมีค่า accuracy โดยรวมคือ 71% และหากพิจารณารายละเอียดในการจำแนกคลาสต่างๆโดยอาศัยค่า precision, recall และ f1 ร่วมด้วย จะสามารถสรุปได้ดังต่อไปนี้

(1) ประสิทธิภาพการจำแนก unknown มาจากการพิจารณาค่า recall ซึ่งจะเห็นว่าสามารถตรวจหาภาพ unknown class ได้ 72% และหากพิจารณาค่า precision แล้วสามารถสรุปได้ว่าภาพที่ถูกทำนายเป็น unknown ทั้งหมดมีความถูกต้องที่ 92% เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 87%

(2) ประสิทธิภาพในการจำแนก p1 พิจารณาจากการที่สามารถตรวจหา p1 ด้ 82% โดยพิจารณาจากค่า recall และ ค่า precision ที่ 49% แสดงให้เห็นความถูกต้องของภาพทั้งหมดที่ถูกจำแนกเป็น p1 เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 61%

(3) ประสิทธิภาพในการจำแนก p2 พิจารณจากความสามารถตวรจหา p2 ได้ 83% โดยดูจากค่า recall และความถูกต้องของภาพทั้งหมดที่ถูกจำแนกเป็น p2 พิจารณาจากค่า precision ที่ 49% เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 61%

(4) ประสิทธิภาพในการจำแนก p3 พิจารณาจากค่า recall ที่ได้ 84% แสดงให้เห็นถึงประสิทธิภาพในการตรวจหา p3 ได้ทั้งหมด แต่เมื่อพิจารณาค่า precession ที่ 100% จะเห็นว่าภาพที่ถูกทำนายเป็น p3 จะมีความถูกต้องที่ 67% เท่านั้น เมื่อนำค่า precision และ recall มาหาค่าเฉลี่ยจะได้ค่า f1 ที่ 75

บทที่ 5
สรุปผลการวิจัย อภิปรายผล และข้อเสนอแนะ

ผู้วิจัยได้วัดประสิทธิภาพของแบบจำลองแต่ละอัลกอริทึมเพื่อนำมาเปรียบเทียบและสรุปผล โดยสามารถแบ่งหัวข้อในการสรุปผลได้ดังต่อไปนี้

1. สรุปผลการวิจัย
2. อภิปรายผลการวิจัย
3. ข้อเสนอแนะและแนวทางการวิจัยในอนาคต

5.1 สรุปผลการวิจัย

งานวิจัยนี้ได้มุ่งเน้นที่การแก้ปัญหาจำแนกภาพขวดเครื่องดื่มภายใต้สภาพแวดล้อมแบบ open-set ที่ในขั้นตอนการทดสอบแบบจำลองจะมีภาพที่อยู่นอกประเภทที่ปรากฏในขั้นตอนการฝึกฝนแบบจำลอง ภาพขวดเครื่องดื่มคลาส p1,p2,p3 จะเป็นคลาสหลักที่ใช้ในการฝึกฝนแบบจำลอง โดยมีเป้าหมายเพื่อจำแนกภาพใด ๆออกเป็นคลาส p1,p2,p3 แต่หากภาพนั้นไม่อยู่ในคลาสใดใน p1,p2,p3 เลย ก็ต้องจำแนกออกมาเป็นคลาสที่ไม่รู้จัก (unknown class)

โดยวิธีการที่ผู้วิจัยได้ทำการทดลองและนำเสนอนั้นจะเป็นการนำโครงข่ายประสาทเทียมแบบคอนโวลูชัน เข้ามาใช้สร้างแบบจำลองร่วมกับอัลกอริทึมอื่น ๆเพื่อให้แบบจำลองนี้สามารถจำแนกภาพแบบ open-set ได้ โดยสามารถสรุปประสิทธิภาพทั้ง accuracy และเวลาที่ใช้ในการสร้างแบบจำลอง รวมถึงประสิทธิภาพในการจำแนก unknown class ได้ดังต่อไปนี้

ตาราง 5 สรุปประสิทธิภาพ

Approaches	accuracy	misclass	Training time (hours)
N-Binary Classifiers	0.841	0.151	9.45
N+Unknown Class Classifiers	0.925	0.075	3.23
N+Combination	0.627	0.372	1.80
2-Layer (OC-SVM+Multi class)	0.757	0.242	1.31
2-Layer (Isolation Forest+Multi class model)	0.813	0.186	1.32

ตาราง 6 ประสิทธิภาพในการจำแนก unknown class

Approaches	Unknown class		
	precision	recall	F1
N-Binary Classifiers	0.998	0.793	0.934
N+Unknown Class Classifiers	0.998	0.995	0.953
N+Combination	0.766	0.695	0.774
2-Layer (OC-SVM+Multi class)	0.921	0.727	0.870
2-Layer (Isolation Forest+Multi class model)	0.885	0.852	0.888

5.2 อภิปรายผลการวิจัย

จากผลการทดลองข้างต้นจะเห็นได้ว่า N+Unknown Class Classifiers ให้ Accuracy สูงที่สุดที่ 92% และหากพิจารณาเฉพาะการจำแนกภาพที่ไม่รู้จะเห็นว่ายังสามารถจำแนก unknown class ได้ดีที่สุดอีกด้วย เมื่อพิจารณาทั้งค่า precision, recall และ f1

N-Binary Classifiers เป็นวิธีที่ให้ประสิทธิภาพที่ดีรองจาก N+Unknown อย่างไรก็ตาม เวลาที่ใช้ในการฝึกฝนแบบจำลองก็ใช้เวลาสูงกว่าวิธีการอื่นมาก ๆ เนื่องจาก N-Binary Classifiers จะแบบจำลองเท่ากับจำนวนคลาสที่ต้องการจำแนก จึงใช้เวลาสูงกว่าวิธีการอื่น N-Binary Classifiers จึงไม่เหมาะในกรณีที่มีคลาสที่ต้องการจำแนกเป็นจำนวนมาก

N+Combination เป็นวิธีการที่ให้ความแม่นยำด้อยกว่าวิธีการอื่น ๆทั้งหมด จากการทดลองนี้ผู้วิจัยสามารถสรุปได้ว่าการสร้าง Additional class ที่มาจากการนำภาพของ p1, p2, p3 มารวมกัน ไม่สามารถเป็น ตัวแทนของ unknown class ได้ดี

ทั้ง 3 วิธีข้างต้นเป็นเทคนิคการฝึกฝนแบบจำลองโดยใช้ภาพจากคลาสที่ต้องการจำแนกร่วมกับคลาสพิเศษที่เป็นตัวแทนของ unknown class สำหรับงานวิจัยนี้ก็คือภาพขวดอื่น ๆ ที่อยู่นอกเหนือ p1, p2, p3 ซึ่งเทคนิคลักษะนี้สามารถใช้ได้ในกรณีที่มีตัวอย่างของภาพที่อยู่นอกเหนือคลาสที่ต้องการจำแนก

2-Layer Classification เป็นวิธีที่แตกต่างกันออกไป โดยนำ pre-trained model มาใช้เพื่อสกัดฟีเจอร์ของภาพที่ต้องการจำแนกทั้ง p1, p2, p3 ทำให้ประหยัดเวลาการฝึกฝนแบบจำลองเองตั้งแต่ต้น ฟีเจอร์ที่ได้จะถูกใช้ในการสร้างแบบจำลองโดยใช้อัลกอริทึม OC-SVM และ Isolation

forest ซึ่งให้ผลความแม่นยำค่อนข้างดีเมื่อฟีเจอร์ที่ใช้ถูกลดมิติของข้อมูลลง เมื่อพิจารณาร่วมกับเวลาที่ใช้ในการฝึกฝนแบบจำลองที่น้อยมากแล้ว

5.3 ข้อเสนอแนะและแนวทางการวิจัยในอนาคต

ในการวิจัยครั้งนี้ผู้วิจัยได้มีข้อเสนอแนะเพื่อเป็นแนวทางในการพัฒนาต่อไปไว้ดังต่อไปนี้

1. Decision function ที่ใช้ใน N-Binary Classifiers ส่งผลอย่างมากต่อการทำนาย หากมีการออกแบบ decision function ใหม่ก็จะส่งผลให้ผลการทำนายต่างออกไป

2. สำหรับงานวิจัยนี้ได้สนใจเฉพาะวิธีการสำหรับโครงข่ายประสาทเทียมเท่านั้น จึงไม่มีการ pre-process ภาพ เช่น การปรับสี แสง ความคมชัด ตัดสัญญาณรบกวนเพื่อให้ได้ข้อมูลที่ดีที่สุดในการสร้างแบบจำลอง หากมีการทำ pre-process ที่ดีก็จะส่งผลให้ประสิทธิภาพของแบบจำลองดีขึ้นได้

3. จากการทดลองทั้งหมด จะเห็นได้ว่า heatmap แสดงจุดสำคัญที่ทุกโมเดลสกัดได้ที่ทำให้ได้คำตอบที่ถูกต้องจะเป็นภาพบริเวณฉลาก ดังนั้นหากต้องการจำแนกภาพขวดก็สามารถตัดมาเฉพาะบริเวณฉลากได้เลย

4. ในงานวิจัยชิ้นนี้มีการทดลองการจำแนกแบบ 2-Layer classification ที่แสดงให้เห็นว่าการการใช้ pre-trained model มาใช้ในการจำแนก known และ unknown class เป็นอีกวิธีที่มีประสิทธิภาพดี แต่อย่างไรก็ตาม pre-trained model ยังไม่ได้มีการฝึกฝนเพิ่มเติม ซึ่งหากมีการทำ transfer learning ร่วมด้วย ก็มีแนวโน้มที่จะได้ประสิทธิภาพสูงขึ้น

5. ผู้วิจัยได้เผยแพร่ซอร์สโค้ดผ่านช่องทาง https://github.com/supanat/OPEN-SET-BOTTLE-CLASSIFICATION-USING-CONVOLUTION-NEURAL-NETWORK.git

บรรณานุกรม

Bendale, A., & Boult, T. (2015). Towards Open Set Deep Networks. *arXiv:1511.06233 [cs]*.

Chalapathy, R., Menon, A. K., & Chawla, S. (2018). Anomaly Detection using One-Class Neural Networks. *arXiv:1802.06360 [cs, stat]*.

Ge, Z., Demyanov, S., Chen, Z., & Garnavi, R. (2017). Generative OpenMax for Multi-Class Open Set Classification. *arXiv:1707.07418 [cs]*.

Girshick, R. (2015). Fast R-CNN. *arXiv:1504.08083 [cs]*.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., . . . Bengio, Y. (2014). Generative Adversarial Networks. *arXiv:1406.2661 [cs, stat]*.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. *arXiv:1512.03385 [cs]*.

Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., . . . Murphy, K. (2016). Speed/accuracy trade-offs for modern convolutional object detectors. *arXiv:1611.10012 [cs]*.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In F. Pereira, C. J. C. Burges, L. Bottou, & K. Q. Weinberger (Eds.), *Advances in Neural Information Processing Systems 25* (pp. 1097–1105): Curran Associates, Inc.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., . . . Zitnick, C. L. (2014, 2014). *Microsoft COCO: Common Objects in Context*.

Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008, 12/2008). *Isolation Forest.* Paper presented at the 2008 Eighth IEEE International Conference on Data Mining (ICDM).

Liu, F. T., Ting, K. M., & Zhou, Z. (2008, December 2008). *Isolation Forest.* Paper presented at the 2008 Eighth IEEE International Conference on Data Mining.

Neal, L., Olson, M., Fern, X., Wong, W.-K., & Li, F. (2018). Open Set Learning with Counterfactual Images. In V. Ferrari, M. Hebert, C. Sminchisescu, & Y. Weiss (Eds.), *Computer Vision – ECCV 2018* (Vol. 11210, pp. 620-635). Cham: Springer International Publishing.

Nguyen, A., Yosinski, J., & Clune, J. (2014). Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images. *arXiv:1412.1897 [cs]*.

Oza, P., & Patel, V. M. (2019). One-Class Convolutional Neural Network. *IEEE Signal Processing Letters, 26*(2), 277-281. doi:10.1109/LSP.2018.2889273

Sabokrou, M., Khalooei, M., Fathy, M., & Adeli, E. (2018, 2018). *Adversarially Learned One-Class Classifier for Novelty Detection.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Scheirer, W. J., Rocha, A. d. R., Sapkota, A., & Boult, T. E. (2013). Toward Open Set Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35*(7), 1757-1772. doi:10.1109/TPAMI.2012.256

Schölkopf, B., Williamson, R. C., Smola, A. J., Shawe-Taylor, J., & Platt, J. C. (2000). Support Vector Method for Novelty Detection. In S. A. Solla, T. K. Leen, & K. Müller (Eds.), *Advances in Neural Information Processing Systems 12* (pp. 582–588): MIT Press.

Shu, L., Xu, H., & Liu, B. (2018). Unseen Class Discovery in Open-world Classification. *arXiv:1801.05609 [cs]*.

Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv:1409.1556 [cs]*.

Sricharan, K., & Srivastava, A. (2018). Building robust classifiers through generation of confident out of distribution examples. *arXiv:1812.00239 [cs, stat]*.

Tax, D. M. J., & Duin, R. P. W. (2004). Support Vector Data Description. *Machine Learning, 54*(1), 45-66. doi:10.1023/B:MACH.0000008084.60811.49

Wong, S. C., Gatt, A., Stamatescu, V., & McDonnell, M. D. (2016). *Understanding data augmentation for classification: when to warp?* Paper presented at the 2016 international conference on digital image computing: techniques and applications (DICTA).

ประวัติผู้เขียน

ชื่อ-สกุล	นายศุภณัฐ จินตวัฒน์สกุล
วัน เดือน ปี เกิด	8 พฤษภาคม 2529
สถานที่เกิด	นครศรีธรรมราช