

การประยุกต์ใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันกับ  
รูปภาพกราฟแท่งเทียนเพื่อทำนายการเคลื่อนไหวของราคา  
ข้าวโพด

Using Convolutional Neural Network  
with Candlestick Images to Predict Corn Price  
Movement



สหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
ปริญญาวิทยาศาสตรบัณฑิต (สถิติประยุกต์)  
ภาควิชาสถิติ คณะวิทยาศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2565

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Using Convolutional Neural Network  
with Candlestick Images to Predict Corn Price  
Movement



A COOPERATIVE EDUCATION SUBMITTED IN  
PARTIAL FULFILLMENT OF THE REQUIREMENT FOR  
THE DEGREE OF BACHELOR OF SCIENCE (APPLIED STATISTICS)  
DEPARTMENT OF STATISTICS, SCHOOL OF SCIENCE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารทสวงวนเวลาสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยาดเินหาไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงหรือทำซ้ำโดยไม่ได้รับอนุญาตของเอกสารทุกครั้งที่มีการนำไปใช้

ACADEMIC YEAR 2022

หัวข้อสหกิจศึกษา

การประยุกต์ใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันกับรูปภาพกราฟแท่งเทียนเพื่อทำนายการเคลื่อนไหวของราคาข้าวโพด  
Using Convolutional Neural Network with Candlestick Images to Predict Corn Price Movement

ชื่อนักศึกษา

นายหริวงค์ เจริญชาติรี รหัสนักศึกษา 62050855

ปริญญา

วิทยาศาสตร์บัณฑิต (สถิติประยุกต์)

ภาควิชา

สถิติ

ปีการศึกษา

2565

อาจารย์ที่ปรึกษา

ผศ.ดร.พรพิมล ชัยวุฒิศักดิ์

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.) อนุมัติให้สหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต (สถิติประยุกต์) ประจำปีการศึกษา 2565

คณะกรรมการสอบ	ลายมือชื่อ
รศ.ดร.วลัยลักษณ์ อัครีรวงศ์ ประธานกรรมการ	วลัยลักษณ์ อัครีรวงศ์
ดร.ภาณุ ทองจันทร์ กรรมการ	ภาณุ ทองจันทร์
คุณชยานิน บุญมานะ กรรมการ	ชยานิน บุญมานะ
ผศ.ดร.พรพิมล ชัยวุฒิศักดิ์ กรรมการและอาจารย์ที่ปรึกษา	พรพิมล ชัยวุฒิศักดิ์

ลิขสิทธิของคณะวิทยาศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อสหกิจศึกษา	การประยุกต์ใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันกับรูปภาพกราฟแท่งเทียนเพื่อทำนายการเคลื่อนไหวของราคาข้าวโพด
ชื่อนักศึกษา	นายหริวงค์ เจริญชาติรี รหัสนักศึกษา 62050855
ปริญญา	วิทยาศาสตร์บัณฑิต (สถิติประยุกต์)
ภาควิชา	สถิติ
คณะ	วิทยาศาสตร์
มหาวิทยาลัย	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.)
ปีการศึกษา	2565
อาจารย์ที่ปรึกษา	ผศ.ดร.พรพิมล ชัยวุฒิศักดิ์

### บทคัดย่อ

งานวิจัยครั้งนี้มีวัตถุประสงค์เพื่อศึกษาไฮเปอร์พารามิเตอร์ที่มีผลต่อโมเดลทำนายทิศทางราคาข้าวโพด ได้แก่ ขนาดชุดข้อมูล และอัตราการเรียนรู้ และสร้างโมเดลทำนายทิศทางของราคาข้าวโพดที่มีการซื้อขายล่วงหน้าในตลาดหอกการค้าแห่งนครซิดนีย์ (CBOT) โดยพิจารณาจากรูปภาพกราฟแท่งเทียน ที่กรอบเวลา 5 และ 20 วัน โดยแบ่งข้อมูลออกเป็น 3 ชุด ได้แก่ ชุดฝึกสอน ชุดตรวจสอบ และชุดทดสอบ ตามอัตราร้อยละ 70:10:20 โมเดลที่นำเสนอในงานวิจัยนี้ได้แก่ Customized CNN, VGG-16 และ Efficientnet-B0 ที่ผ่านการปรับแต่ง ผลการศึกษาค่าไฮเปอร์พารามิเตอร์ที่เหมาะสมที่กรอบเวลา 5 วัน พบว่าขนาดชุดข้อมูล และอัตราการเรียนรู้ที่เหมาะสมที่สุดของทั้ง 3 โมเดลคือ ขนาดชุดข้อมูล เท่ากับ 16 ที่อัตราการเรียนรู้ 0.001 และกรอบเวลา 20 วัน ขนาดชุดข้อมูล เท่ากับ 16 โดยอัตราการเรียนรู้ที่เหมาะสมสำหรับโมเดล Customized CNN คือ 0.001 ในขณะที่อัตราการเรียนรู้ที่เหมาะสมสำหรับโมเดล VGG-16 และ Efficientnet-B0 คือ 0.0001 จากนั้นปรับแต่งค่าไฮเปอร์พารามิเตอร์ที่เหมาะสมสำหรับแต่ละโมเดล และทดสอบโมเดลด้วยข้อมูลชุดทดสอบ ผลการศึกษาพบว่า ณ กรอบเวลา 5 วัน โมเดล Customized CNN สามารถทำนายทิศทางของราคาข้าวโพดได้แม่นยำกว่าโมเดลอื่น ๆ โดยให้ค่าความแม่นยำคิดเป็นร้อยละ 55.39 ในขณะที่ ณ กรอบเวลา 20 วัน โมเดลที่ให้ค่าความแม่นยำสูงที่สุดคือ Efficientnet-B0 โดยให้ค่าความแม่นยำคิดเป็นร้อยละ 55.03

**คำสำคัญ :** ทำนายทิศทางราคาข้าวโพด, โครงข่ายประสาทเทียมแบบคอนโวลูชัน, การเรียนรู้เชิงลึก, ไฮเปอร์พารามิเตอร์, กราฟแท่งเทียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<b>Title</b>	Using Convolutional Neural Network with Candlestick Images to Predict Corn Price Movement
<b>Students</b>	Mr. Hariwong Charoenchatree Student ID 62050855
<b>Degree</b>	Bachelor of Science (Applied Statistics)
<b>Department</b>	Statistics
<b>School</b>	Science
<b>University</b>	King Mongkut's Institute of Technology Ladkrabang (KMITL)
<b>Academic Year</b>	2565
<b>Advisor</b>	Asst.Prof.Dr. Pornpimol Chaiwuttisak

### Abstract

The objective of this research is to study the effect of hyperparameters on corn price movement prediction models, namely batch size and learning rate. and create a model to predict the corn price movement in the Chicago Board of Trade (CBOT). based on candlestick images at 5-day and 20-day timeframes. The data is split into three sets, namely, training set, validation set, and test set, with a ratio of 70:10:20. The models presented in this research are Customized CNN, VGG-16, and Efficientnet-B0, which have fine-tuning. The study's findings on hyperparameter values within a 5-day timeframe revealed that the optimal batch size and learning rate for all three models were a batch size of 16 with a learning rate of 0.001 and a timeframe of 20 days with a dataset size of 16. However, the suitable learning rate for the Customized CNN model was 0.001, while for the VGG-16 and EfficientNet-B0 models, it was 0.0001. Subsequently, the hyperparameter values were fine-tuned for each model, and and test the model with the test set. The study findings revealed that at the 5-day timeframe, the Customized CNN model outperformed other models in predicting corn price movement, with an accuracy of 55.39%. While, at a 20-day timeframe, the model with the highest accuracy was EfficientNet-B0, with an accuracy of 55.03%.

**Keywords :** Predict Corn Price Movement, Convolutional Neural Network, Deep Learning, Hyperparameter, Candlestick.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

สหกิจศึกษานี้สำเร็จลุล่วงไปได้ด้วยดี เนื่องจากคณะผู้จัดทำได้รับความอนุเคราะห์และความกรุณาจากคณะอาจารย์และบุคคลผู้มีพระคุณหลายท่าน ดังรายนามต่อไปนี้

ขอขอบพระคุณ ผศ.ดร.พรพิมล ชัยวุฒิศักดิ์ อาจารย์ภาควิชาสถิติ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อาจารย์ที่ปรึกษาสหกิจศึกษาที่ได้ช่วยแนะนำและให้คำปรึกษา รวมถึงเสนอแนะแนวทางการแก้ไขปัญหา ตลอดจนถ่ายทอดประสบการณ์ในการทำงานของท่านเพื่อเป็นประโยชน์ในการคิดวิเคราะห์ การวางแผนในโครงการนี้ โดยท่านได้ให้คำปรึกษา ตั้งแต่การค้นหาข้อมูลตลอดจนการทำงานวิจัยสำเร็จ รวมทั้งตรวจทานแก้ไขสหกิจศึกษาเล่มนี้ให้สมบูรณ์

ขอขอบพระคุณ รศ.ดร.วลัยลักษณ์ อัครธีรวงศ์ ที่กรุณาเป็นกรรมการในการสอบสหกิจศึกษา อีกทั้งยังให้ความรู้ คำแนะนำ และช่วยตรวจสอบแก้ไขให้สหกิจศึกษาเล่มนี้ออกมาสมบูรณ์

ขอขอบพระคุณ ดร.ภาณุ ทองจันทร์ ผู้ช่วยกรรมการผู้จัดการ หน่วยงานวิเคราะห์เชิงปริมาณ (Quantitative Model) ด้านเทคโนโลยีสารสนเทศ กลุ่มธุรกิจการค้าวัตถุดิบอาหารสัตว์ บริษัท กรุงเทพ โปรตี๊วส จำกัด (มหาชน) ที่มอบโอกาส และให้การอนุเคราะห์ในการทำสหกิจครั้งนี้ และขอขอบพระคุณคุณชญาสินี บุญมานะ คุณณรงค์พล วิชัยลักษณ์ คุณศุภพงศ์ คงเจริญ คุณฐิติมา ตโมทรณวงศ์ และพี่ ๆ ในหน่วยงานที่คุณดูแล เอาใจใส่ ให้คำปรึกษา ให้กำลังใจ และคำแนะนำ อีกทั้งยังช่วยแก้ไขข้อผิดพลาดต่าง ๆ ตลอดระยะเวลาการทำสหกิจศึกษาในครั้งนี้จนสามารถสำเร็จลุล่วงไปได้ด้วยดี

สุดท้ายนี้ผู้จัดทำขอขอบคุณบิดา มารดา และบุคคลในครอบครัว รวมทั้งเพื่อน ๆ พี่ ๆ และบุคคลที่ไม่ได้กล่าวถึงมา ณ ที่นี้ที่ให้ความช่วยเหลือ การสนับสนุน และกำลังใจตลอดการทำสหกิจศึกษานี้ให้สำเร็จไปได้ด้วยดี

หริวงค์ เจริญชาติตรี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย .....	ก
บทคัดย่อภาษาอังกฤษ.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ช
สารบัญรูป .....	ซ
<b>บทที่ 1 บทนำ.....</b>	<b>1</b>
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของงานวิจัย .....	4
1.3 ขอบเขตของงานวิจัย .....	4
1.4 ประโยชน์ที่คาดว่าจะได้รับ .....	4
1.5 นิยามศัพท์เฉพาะ .....	5
<b>บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....</b>	<b>6</b>
2.1 แนวคิดและทฤษฎีที่เกี่ยวข้องกับข่าวโศก .....	6
2.1.1 ลักษณะทางพฤกษศาสตร์ของข้าวโพด .....	7
2.1.2 ข้าวโพดในประเทศไทย .....	9
2.1.3 ความสำคัญของข้าวโพดต่อระบบเศรษฐกิจไทย .....	9
2.2 แนวคิดและทฤษฎีที่เกี่ยวข้องกับตลาดซื้อขายสินค้าเกษตรล่วงหน้า .....	11
2.2.1 บทบาทและหน้าที่ของตลาดสินค้าเกษตรล่วงหน้า .....	11
2.2.2 ประโยชน์ของตลาดสินค้าเกษตรล่วงหน้า .....	12
2.3 แนวคิดและทฤษฎีที่เกี่ยวข้องกับกราฟแท่งเทียน .....	12
2.3.1 ที่มาและความหมายของกราฟแท่งเทียน .....	12
2.3.2 ส่วนประกอบต่าง ๆ ของแท่งเทียน.....	13
2.3.3 ลักษณะของแผนภูมิแท่งเทียน.....	14
2.4 โครงข่ายประสาทเทียม (Neural Network).....	16
2.4.1 ฟังก์ชันกระตุ้น (Activation Functions).....	16
2.4.1.1 ฟังก์ชันกระตุ้นซิกมอยด์	
(Sigmoid Activation Function).....	17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
2.4.1.2 ฟังก์ชันกระตุ้นเรกติไฟด์ลินีเยยูนิต (Rectified Linear Units: ReLu) .....	18
2.4.2 ฟังก์ชันการสูญเสีย (Loss Functions).....	18
2.5 การเรียนรู้เชิงลึก (Deep Learning).....	19
2.6 โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Network: CNN) .....	20
2.6.1 ชั้นนำเข้าข้อมูล (Input Layer).....	20
2.6.2 ชั้นคอนโวลูชัน (Convolution Layer) .....	20
2.6.4 ชั้นพูลลิ่ง (Pooling Layer).....	21
2.5.4 ชั้นเชื่อมโยงสมบูรณ์ (Fully Connected Layer).....	22
2.7 การปรับปรุงประสิทธิภาพ (Optimization).....	22
2.8 สถาปัตยกรรมโครงข่ายประสาทแบบคอนโวลูชัน (CNN Architecture) .....	24
2.8.1 Visual Geometry Group (VGG).....	24
2.8.2 EfficientNet .....	25
2.8.2.1 โครงสร้างเริ่มต้น (Initial Structure).....	25
2.8.2.2 การปรับขนาดแบบผสม (Compound Scaling).....	26
2.9 การวัดประสิทธิภาพของโมเดล (Model Performance).....	28
2.10 งานวิจัยที่เกี่ยวข้อง (Related Research) .....	29
<b>บทที่ 3 วิธีการดำเนินงานวิจัย .....</b>	<b>31</b>
3.1 กรอบแนวคิดการวิจัย .....	31
3.2 การรวบรวมข้อมูล .....	33
3.3 การจัดเตรียมข้อมูล (Data Preprocessing).....	33
3.3.1 การจัดการข้อมูลเบื้องต้น .....	33
3.3.2 การสร้างรูปภาพกราฟแท่งเทียบ .....	33
3.3.3 การแปลงราคาข้าวโพด .....	36
3.3.4 การจัดเก็บข้อมูล .....	36
3.3.5 การปรับค่าความเข้มของสี .....	38

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้ในการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
3.4 การสร้างโมเดลทำนายทิศทางราคาข้าวโพด .....	39
3.4.1 โครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNN) .....	40
3.4.2 Visual Geometry Group (VGG-16) .....	41
3.4.3 EfficientNet-B0.....	41
3.4.4 การกำหนดค่าไฮเปอร์พารามิเตอร์ .....	41
3.5 เครื่องมือที่ใช้ในงานวิจัย .....	42
3.5.1 โปรแกรมภาษาไพธอน (Python 3).....	42
3.6 การเปรียบเทียบประสิทธิภาพของโมเดล.....	43
<b>บทที่ 4 ผลการวิจัย .....</b>	<b>44</b>
4.1 การวิเคราะห์ข้อมูลเบื้องต้น.....	44
4.2 ผลการทดสอบเพื่อหาค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุดของโมเดล .....	51
4.2.1 ผลการทดสอบเพื่อหาค่าไฮเปอร์พารามิเตอร์ของโมเดล Customized CNN.....	52
4.2.2 ผลการทดสอบเพื่อหาค่าไฮเปอร์พารามิเตอร์ของโมเดล VGG-16.....	55
4.2.3 ผลการทดสอบเพื่อหาค่าไฮเปอร์พารามิเตอร์ของโมเดล EfficientNet-B0 .....	59
4.2.4 ผลสรุปค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุดในแต่ละกรอบเวลา.....	64
4.3 ผลการทดสอบโมเดลจากค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุด.....	65
4.4 การอภิปรายผล .....	66
<b>บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ.....</b>	<b>67</b>
5.1 สรุปผลการวิจัย .....	67
5.2 ข้อเสนอแนะ .....	68
5.3 แนวทางที่จะศึกษาต่อในอนาคต .....	68
<b>บรรณานุกรม.....</b>	<b>69</b>
<b>ภาคผนวก.....</b>	<b>75</b>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

ตารางที่	หน้า
2.6 เมทริกซ์ความสับสน (Confusion Matrix) ขนาด 2x2 .....	28
3.1 ตัวแปรที่ใช้ในการสร้างกราฟแท่งเทียบ .....	34
3.2 การแปลงราคาข้าวโพด .....	36
3.3 ค่าไฮเปอร์พารามิเตอร์ที่กำหนด.....	42
4.1 สัดส่วนของผลเฉลย ณ ระยะเวลา 5 วัน ในแต่ละชุดข้อมูล .....	48
4.2 สัดส่วนของผลเฉลย ณ ระยะเวลา 20 วัน ในแต่ละชุดข้อมูล.....	49
4.3 ค่าความแม่นยำและค่าของโมเดล Customized CNN บนชุดข้อมูลตรวจสอบ .....	52
4.4 ค่าความแม่นยำและค่าการสูญเสียของโมเดล VGG-16 บนชุดข้อมูลตรวจสอบ .....	55
4.5 ค่าความแม่นยำและค่าการสูญเสียของโมเดล EfficientNet-B0 บนชุดข้อมูลตรวจสอบ.....	59
4.6 ค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุด ณ ระยะเวลา 5 วัน.....	64
4.7 ค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุด ณ ระยะเวลา 5 วัน.....	64
4.8 ค่าการสูญเสียและค่าความแม่นยำของแต่ละโมเดลบนชุดข้อมูลทดสอบ ณ ระยะเวลา 5 วัน.....	65
4.9 ค่าการสูญเสียและค่าความแม่นยำของแต่ละโมเดลบนชุดข้อมูลทดสอบ ณ ระยะเวลา 20 วัน.....	65

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

รูปที่	หน้า
1.1 ปริมาณการนำเข้าข้าวโพดตั้งแต่ปี 2560-2564 .....	1
1.2 ลักษณะของกราฟแท่งเทียน.....	2
2.1 ลักษณะราคาข้าวโพด .....	7
2.2 ลักษณะลำต้นข้าวโพด .....	8
2.3 ลักษณะผลข้าวโพด.....	9
2.4 ปริมาณการนำเข้าและส่งออกของข้าวโพดปี 2560-2564.....	10
2.5 ส่วนประกอบของแท่งเทียน .....	14
2.6 แท่งเทียนขาขึ้น.....	14
2.7 แท่งเทียนขาลง .....	15
2.8 แท่งเทียนแบบโดจี .....	15
2.9 สถาปัตยกรรมโครงข่ายประสาทเทียม .....	16
2.10 ฟังก์ชันกระตุ้นซิกมอยด์.....	17
2.11 ฟังก์ชันกระตุ้นซิกเรลดู.....	18
2.12 ตัวอย่างโครงข่ายประสาทเทียมแบบลึก.....	19
2.13 สถาปัตยกรรมโครงข่ายประสาทเทียมแบบคอนโวลูชัน .....	20
2.14 ตัวอย่างการทำงานของคอนโวลูชัน ขนาด 3 x 3.....	21
2.15 ตัวอย่างการทำงานของชั้น Pooling ด้วยวิธี Max Pooling.....	21
2.16 ตัวอย่างการทำงานของชั้น Pooling ด้วยวิธี Average Pooling.....	22
2.17 ตัวอย่างการทำงานของชั้นเชื่อมโยงสมบูรณ์ .....	22
2.18 สถาปัตยกรรมของ VGG16.....	25
2.19 การปรับขนาดแบบผสม.....	27
3.1 กรอบแนวคิดการวิจัย .....	32
3.2 ตัวอย่างราคาข้าวโพด (รายวัน).....	33
3.3 ตัวอย่างการสร้างกราฟแท่งเทียน (กรอบเวลา 5 วัน).....	34
3.4 ตัวอย่างการสร้างกราฟแท่งเทียน (กรอบเวลา 20 วัน).....	35
3.5 ตัวอย่างการจัดเก็บข้อมูล (กรอบเวลา 5 วัน).....	37
3.6 ตัวอย่างการจัดเก็บข้อมูล (กรอบเวลา 20 วัน).....	38
3.7 รูปภาพที่ไม่ได้มีการปรับค่าความเข้มของสีให้อยู่ในช่วง [0-1].....	39
3.8 รูปภาพที่มีการปรับค่าความเข้มของสีให้อยู่ในช่วง [0-1].....	39
3.9 สถาปัตยกรรมโครงข่ายประสาทเทียมแบบคอนโวลูชันที่ออกแบบโดยผู้วิจัย .....	40

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ในการใช้งานเพื่อการศึกษานานาชาติไปลงเผยแพร่ให้ประโยชน์ด้านการศึกษา  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.10 สถาปัตยกรรม VGG-16 ที่ใช้ในงานวิจัย .....	41
3.11 สถาปัตยกรรม EfficientNet-B0 ที่ใช้ในงานวิจัย .....	41
4.1 การเคลื่อนไหวของราคาข้าวโพด .....	44
4.2 แผนภาพกล่อง (Box-Plot) ของราคาข้าวโพด .....	45
4.3 การวิเคราะห์สถิติเบื้องต้นเกี่ยวกับราคาข้าวโพด.....	46
4.4 กราฟแท่งเทียนที่กรอบเวลา 5 วัน .....	47
4.5 กราฟแท่งเทียนที่กรอบเวลา 20 วัน.....	47
4.6 กราฟแท่งเทียนแสดงการเคลื่อนไหวของราคาข้าวโพด.....	47
4.7 กราฟแท่งแสดงผลเฉลี่ยในกรอบเวลา 5 วัน.....	48
4.8 กราฟแท่งแสดงผลเฉลี่ยในแต่ละชุดข้อมูลของกรอบเวลา 5 วัน.....	49
4.9 กราฟแท่งแสดงผลเฉลี่ยในกรอบเวลา 20 วัน.....	50
4.10 กราฟแท่งแสดงผลเฉลี่ยในแต่ละชุดข้อมูลของกรอบเวลา 20 วัน.....	51
4.11 กราฟเปรียบเทียบค่าการสูญเสีย และค่าความแม่นยำของโมเดล Customized CNN ณ กรอบเวลา 5 วัน.....	53
4.12 กราฟเปรียบเทียบค่าการสูญเสีย และค่าความแม่นยำของโมเดล Customized CNN ณ กรอบเวลา 20 วัน.....	54
4.13 กราฟเปรียบเทียบค่าการสูญเสีย และค่าความแม่นยำของโมเดล VGG-16 ณ กรอบเวลา 5 วัน .....	57
4.14 กราฟเปรียบเทียบค่าการสูญเสีย และค่าความแม่นยำของโมเดล VGG-16 ณ กรอบเวลา 20 วัน .....	58
4.15 กราฟเปรียบเทียบค่าการสูญเสีย และค่าความแม่นยำของโมเดล EfficientNet-B0 ณ กรอบเวลา 5 วัน.....	61
4.16 กราฟเปรียบเทียบค่าการสูญเสีย และค่าความแม่นยำของโมเดล EfficientNet-B0 ณ กรอบเวลา 20 วัน.....	63

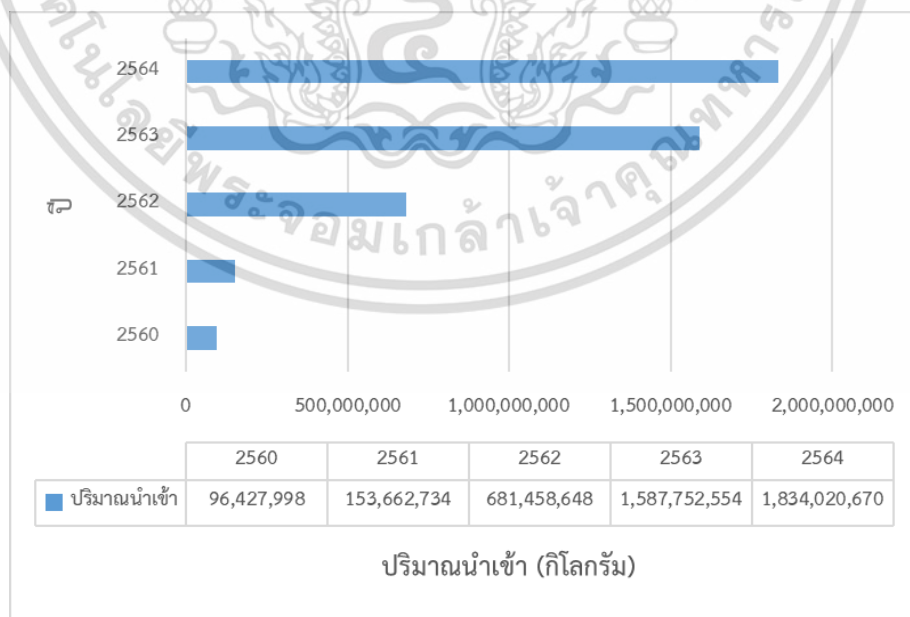
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญ

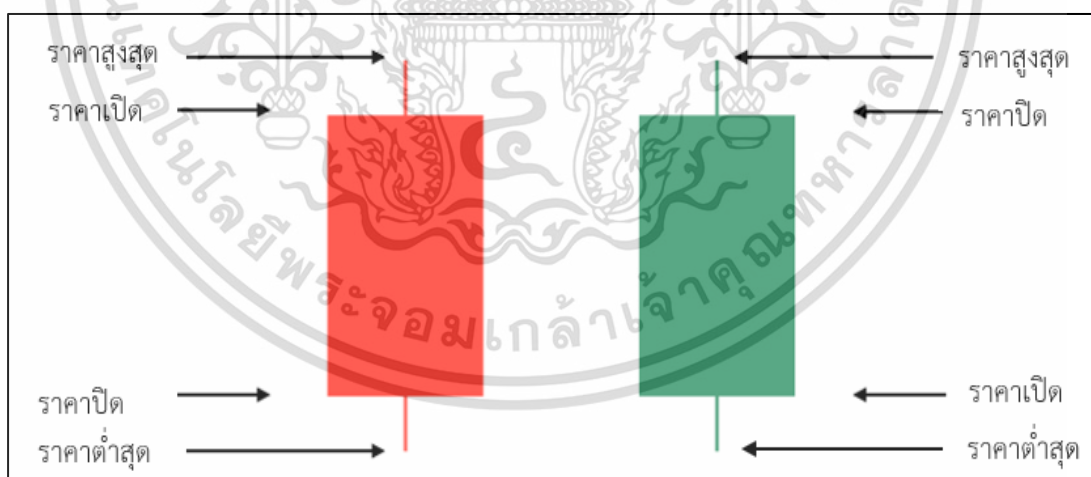
ข้าวโพดเป็นธัญพืชเศรษฐกิจสำคัญที่ใช้เป็นอาหารของมนุษย์และสัตว์ โดยข้าวโพดสามารถปลูกได้ทั้งในพื้นที่ราบเขตร้อน เขตกึ่งร้อน และภูมิอากาศอบอุ่น นอกจากนี้ยังมีคุณสมบัติทางอาหารที่สูง สามารถใช้ประโยชน์ได้ในทุกด้าน ในการปลูกข้าวโพดสามารถทำได้ในเขตภูมิภาคต่าง ๆ ไม่ว่าจะเป็นเขตอบอุ่น เขตกึ่งร้อนชื้น และพื้นที่ราบเขตร้อน (นพพร และคณะ, 2547) โดยแหล่งปลูกมักกระจายอยู่ตามภูมิภาคต่าง ๆ ของโลก ได้แก่ ประเทศสหรัฐอเมริกา บราซิล เม็กซิโก จีน รวมทั้งในทวีป ประโยชน์ของข้าวโพดมีมากมายนอกจากใช้เป็นอาหารโดยตรงของมนุษย์และสัตว์แล้ว ยังสามารถนำมาแปรรูปเป็นผลิตภัณฑ์จากข้าวโพดได้หลายชนิด ทั้งในระดับครัวเรือนและในระดับอุตสาหกรรม ทำให้สินค้าเป็นที่ต้องการของผู้บริโภคทั้งในประเทศและต่างประเทศ ในส่วนของการนำเข้ามีแนวโน้มเพิ่มขึ้นดังแสดงในรูปที่ 1.1 เนื่องจากผลผลิตข้าวโพดเลี้ยงสัตว์ภายในประเทศไม่เพียงพอต่อความต้องการใช้ในอุตสาหกรรมการผลิตอาหารสัตว์ (โชคชัย และเกตุอร, 2561) นอกจากนี้ยังมีการซื้อขายกันในตลาดหอกการค้าแห่งนครชิคาโก (CBOT) โดยสินค้าเกษตรที่ทำการซื้อขาย ได้แก่ ข้าวสาลี ข้าวโพด ข้าวโอ๊ต ถั่วเหลือง ถั่วเหลืองปน น้ำมันถั่วเหลือง และข้าวเปลือก โดยสินค้าเกษตรที่มีการซื้อขายสัญญาล่วงหน้ามากที่สุดคือ ข้าวโพด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น รูปที่ 1.1 ปริมาณการนำเข้าข้าวโพดตั้งแต่ปี 2560-2564 ใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ (ที่มา: สำนักงานเศรษฐกิจการเกษตร, 2564) สารทุกครั้งที่มีการนำไปใช้

ตลาดหอการค้าแห่งนครชิคาโก หรือ Chicago Board of Trade (CBOT) เป็นตลาดซื้อขายสัญญาล่วงหน้า (Futures Contract) อีกทั้งยังเป็นตลาดสินค้าล่วงหน้าแห่งแรกในสหรัฐฯ ซึ่งสามารถใช้ในการประมาณการราคาสินค้าในอนาคต และเป็นวิธีที่ใช้ในการควบคุมความเสี่ยงในการลงทุนในธุรกิจด้านการเกษตร โดยสินค้าชนิดแรกที่มีการซื้อขายโดยการนำมาทำเป็นสัญญาซื้อขายคือ ข้าวโพด โดยจุดเด่นของตลาดซื้อขายล่วงหน้าคือสามารถที่จะวิเคราะห์รูปแบบของกราฟเพื่อที่จะจำแนกข้อมูล (Classification) และช่วยในการตัดสินใจว่าจะทำการซื้อขายอย่างไร ณ ช่วงเวลานั้น ๆ (ดริน, 2547) ซึ่งรูปแบบของกราฟที่จะนำมาวิเคราะห์ในที่นี้คือกราฟแท่งเทียน (Candlestick Chart)

กราฟแท่งเทียน (Candlestick Chart) เป็นกราฟที่แสดงราคาของหุ้นตัวนั้น ซึ่งจะแสดงราคาเปิด (Open Price) เป็นราคาซื้อขายแรกที่เกิดขึ้นตั้งแต่เปิดตลาด ราคาปิด (Close Price) เป็นราคาสุดท้ายที่เกิดขึ้นจากการซื้อขายสิ้นสุดของวัน ราคาสูงสุด (High Price) แสดงการเคลื่อนไหวของราคาหุ้น ณ ระดับราคาสูงสุดในวันทำการ ราคาต่ำสุด (Low Price) แสดงการเคลื่อนไหวของราคาหุ้น ณ ระดับราคาต่ำสุดในวันทำการ ตัวเทียน (Real Body) คือ ช่วงระหว่างราคาเปิดกับราคาปิด โดยถ้าราคาปิดสูงกว่าราคาเปิดตัวเทียนจะแสดงเป็นสีเขียว (Bullish Candlestick) ทางกลับกันถ้าราคาปิดต่ำกว่าราคาเปิดตัวเทียนจะแสดงเป็นสีแดง (Bearish Candlestick) ดังแสดงในรูปที่ 1.2 (สิริพร, 2561)



รูปที่1.2 ลักษณะของกราฟแท่งเทียน

โดยกราฟแท่งเทียนสามารถนำมาวิเคราะห์ทางเทคนิคได้ (Technical Analysis) ซึ่งนั่นหมายความว่านักลงทุนสามารถใช้ประโยชน์จากจุดนี้เพื่อลดความเสี่ยงในการลงทุน และสามารถนำเอกสารนี้เป็นเอกสารที่ลงทุนในหลักทรัพย์หรือการลงทุนเพื่อการศึกษานี้ เมื่อผู้ซื้อได้เห็นว่าประโยชน์ที่ตนได้ราคาไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กราฟแท่งเทียนมาวิเคราะห์ทางเทคนิคเพื่อคาดการณ์แนวโน้มการขึ้นลง (หลักทรัพย์บัวหลวง, 2564) ซึ่งในปัจจุบันยังสามารถใช้งาน ปัญญาประดิษฐ์ มาช่วยในการวิเคราะห์ได้อีกด้วย

ปัญญาประดิษฐ์ (Artificial Intelligence : AI) เป็นเทคโนโลยีการสร้างเครื่องจักรให้มีคุณลักษณะทางด้านสติปัญญาและความฉลาดเหมือนมนุษย์ โดยประกอบไปด้วย การเรียนรู้ของเครื่อง (Machine Learning) ซึ่งเป็นศาสตร์ที่จะช่วยทำให้คอมพิวเตอร์นั้น มีความสามารถในการเรียนรู้เหมือนมนุษย์ นอกจากนี้ ยังมีการพัฒนาขั้นตอนวิธีประเภทใหม่ ๆ ของการเรียนรู้ของเครื่องขึ้นมาอีก ได้แก่ การเรียนรู้เชิงลึก (Deep Learning) โดยเป็นการจำลองเครือข่ายประสาทของมนุษย์ให้มีการประมวลผลแบบขนานต่อกันหลาย ๆ ชั้นในลักษณะต่อเนื่องกันทำให้สามารถประมวลผลข้อมูลจำนวนมากได้อย่างรวดเร็วด้วยประสิทธิภาพที่ไม่ลดลง (จตุรพัชร์, 2564) การเรียนรู้เชิงลึกมีสถาปัตยกรรมที่น่าสนใจมากมายหนึ่งในนั้นคือ โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Network)

โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Network) เรียกอ่อ ๆ ว่า “ConvNet” หรือ “CNN” เป็นการเรียนรู้แบบมีผู้สอน (Supervised Learning) ถือว่าเป็นอีกประเภทหนึ่งของโครงข่ายประสาทเทียม (Neural Network) ที่มีหลายชั้น (Multilayer Perceptron) โดยโครงข่ายประสาทเทียมแบบคอนโวลูชันจะโดดเด่นในงานด้านรูปภาพ มักนำไปใช้ประมวลผลภาพ (Image Processing) หลักการของโครงข่ายประสาทเทียมแบบคอนโวลูชันคือการใช้ชั้นชนิดพิเศษที่เรียกว่า ชั้นคอนโวลูชัน ซึ่งทำหน้าที่สกัดคุณลักษณะพิเศษของภาพออกมา (Feature Extraction) เช่น เส้นขอบของวัตถุต่าง ๆ เพื่อให้โมเดลสามารถเรียนรู้ลักษณะของภาพได้อย่างมีประสิทธิภาพและแม่นยำ อีกทั้งยังสามารถจำแนกประเภท (Classification) ซึ่งที่กล่าวมาถือได้ว่าเป็นจุดเด่นของโครงข่ายนี้ โดยใน โครงข่ายประสาทเทียมแบบคอนโวลูชันจะมีส่วนประกอบหลักได้แก่ ชั้นคอนโวลูชัน (Convolution Layer) ชั้นพูลลิ่ง (Pooling Layer) ชั้นกระตุ้น (Activation Layer) โดยที่นิยมใช้ในโครงข่ายประสาทเทียมแบบคอนโวลูชันคือ Relu Function โดยชั้นที่กล่าวมาจะสกัดคุณลักษณะพิเศษ เช่น เส้น ขอบ มุม โดยใช้แผ่นกรองหรือฟิลเตอร์ (Filter) ซึ่งสามารถทำซ้ำ ๆ กันได้ หลังจากสกัดคุณลักษณะพิเศษ ชั้นเชื่อมโยงสมบูรณ์ (Fully Connected) จะช่วยในการหาความสัมพันธ์ของแต่ละฟิลเตอร์ และนำไปสู่การจำแนกเพื่อได้ผลลัพธ์ของการทำนาย (จตุรพัชร์, 2564)

ในงานวิจัยนี้ได้มีการนำอัลกอริทึมโครงข่ายประสาทเทียมแบบคอนโวลูชันมาสร้างโมเดลทำนายทิศทางราคาข้าวโพด (Customize CNN) จากนั้นได้มีการนำโมเดลที่เป็นสถาปัตยกรรมแบบคอนโวลูชันได้แก่ VGG-16 และ Efficientnet-B0 เพื่อเปรียบเทียบค่าความแม่นยำ และศึกษาไฮเปอร์พารามิเตอร์ที่มีผลต่อโมเดลการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.2 วัตถุประสงค์ของงานวิจัย

- 1) เพื่อสร้างโมเดลทำนายทิศทางของราคาข้าวโพดที่มีการซื้อขายล่วงหน้าในตลาดหอการค้าแห่งนครชิคาโกโดยพิจารณาจากกราฟแท่งเทียน
- 2) เพื่อศึกษาไฮเปอร์พารามิเตอร์ที่มีผลต่อโมเดลทำนายทิศทางราคาข้าวโพดในตลาดหอการค้าแห่งนครชิคาโก
- 3) เพื่อเปรียบเทียบความแม่นยำของโมเดลที่ใช้ในการทำนายทิศทางราคาข้าวโพดที่มีการซื้อขายล่วงหน้าในตลาดหอการค้าแห่งนครชิคาโก

## 1.3 ขอบเขตของงานวิจัย

### 1.3.1. ขอบเขตด้านข้อมูล

ในงานวิจัยนี้ใช้ราคาข้าวโพดจากตลาดหอการค้าแห่งนครชิคาโก หรือ Chicago Board of Trade (CBOT) ซึ่งจะใช้ตัวแปร ราคาเปิด ราคาปิด ราคาสูงสุด และราคาต่ำสุด เพื่อสร้างกราฟแท่งเทียนที่ใช้เป็นข้อมูลสำหรับการทำนายทิศทางราคาข้าวโพด

### 1.3.2. ขอบเขตด้านระยะเวลา

ข้อมูลรายวันถูกเก็บรวบรวมตั้งแต่วันที่ 3 มกราคม พ.ศ.2543 ถึง วันที่ 1 มกราคม พ.ศ.2566 รวมเป็นจำนวน 6,023 วัน

### 1.3.3. ขอบเขตด้านเครื่องมือ

เครื่องมือที่ใช้สำหรับการจัดเตรียมข้อมูลและสร้างโมเดลสำหรับวิเคราะห์ข้อมูล

- 1) โปรแกรม Microsoft Office Excel
- 2) โปรแกรมภาษาไพธอน (Python 3) บน Jupyter Notebook
- 3) โปรแกรมภาษาไพธอน (Python 3) บน Google Colab

## 1.4 ประโยชน์ที่คาดว่าจะได้รับ

- 1) สามารถนำโมเดลไปใช้ในการวางแผนลงทุนการซื้อขายข้าวโพดในตลาดหอการค้าแห่งนครชิคาโก
- 2) สามารถนำโมเดลไปประยุกต์ใช้กับผลิตภัณฑ์ทางการเกษตรอื่น ๆ ในตลาดหอการค้าแห่งนครชิคาโกได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.5 นิยามศัพท์เฉพาะ

- 1) สัญญาฟิวเจอร์ส (Futures Contract) เป็นสัญญาระหว่างคู่สัญญาสองฝ่ายที่ตกลงราคากัน ณ ปัจจุบัน และจะมีการส่งมอบสินทรัพย์และชำระเงินในอนาคตตามราคาที่ตกลงไว้ไม่ว่าราคาในขณะนั้นจะเป็นเท่าไรก็ตาม
- 2) ตลาดหอยการค้าแห่งนครชิคาโก หรือ Chicago Board of Trade (CBOT) เป็นตลาดสินค้าล่วงหน้าแห่งแรกในสหรัฐอเมริกา
- 3) การประมวลผลภาพ (Image processing) หมายถึงกระบวนการที่ใช้เทคนิคและเครื่องมือต่าง ๆ เพื่อแปลงและประมวลผลภาพดิจิทัล เช่น ภาพนิ่ง (ภาพถ่าย) หรือภาพวิดีโอ (วีดีโอ) (Sonka et al., 2013)
- 4) โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Network : CNN) เป็นโครงข่ายประสาทเทียมที่ใช้ในการประมวลผลภาพมีส่วนประกอบหลักคือการใช้ชั้นการสกัดคุณลักษณะ (Feature Extraction) ซึ่งประกอบด้วยการใช้ชั้นคอนโวลูชัน (Convolutional Layer) เพื่อดึงคุณลักษณะที่สำคัญออกมาจากข้อมูลเข้า และชั้นพูลลิง (Pooling Layer) เพื่อลด หลังจากนั้นใช้ชั้น Fully Connected Layer เพื่อรวมคุณลักษณะที่ได้จากชั้นก่อนหน้าเป็นเวกเตอร์หรือเมทริกซ์เดียว เพื่อทำการจำแนกหรือประมวลผลต่อไป (Sultana et al., 2018)
- 5) Visual Geometry Group 16 (VGG16) เป็นโครงข่ายประสาทเทียมแบบคอนโวลูชันที่ถูกพัฒนาขึ้นโดยทีมวิจัยจาก Visual Geometry Group ที่มหาวิทยาลัยแอ็กซ์ฟอร์ดในปี 2014. VGG16 เป็นหนึ่งในโมเดล CNN ที่ได้รับความนิยมและถูกนำไปใช้ในงานต่าง ๆ ด้วยความลึกของโมเดลที่มี 16 ชั้น (Simonyan et al., 2015)
- 6) Efficientnet B0 เป็นโมเดลโครงข่ายประสาทเทียมแบบคอนโวลูชันที่ถูกพัฒนาขึ้นโดยทีมวิจัยจาก Google Brain ในปี 2019 ซึ่งเน้นในเรื่องของความมั่นคงและประสิทธิภาพในการทำงาน มีคุณสมบัติที่น่าสนใจ อย่าง การปรับขนาดแบบผสม (Compound Scaling) ซึ่งช่วยให้สามารถปรับความลึกของโมเดลและขนาดของโมเดลให้เหมาะสมกับข้อมูลที่ใช้งาน ซึ่งเป็นผลมาจากการประยุกต์ใช้เทคนิคเสริมความสำคัญให้กับชั้นคอนโวลูชัน (Tan and Le, 2019)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

# ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในการศึกษาวิจัยครั้งนี้ ผู้วิจัยได้มีการศึกษาแนวความคิดทฤษฎีที่เกี่ยวข้อง ก่อนการทำการวิจัย เพื่อใช้ประกอบการศึกษางานวิจัยเรื่อง"การประยุกต์ใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันกับรูปภาพกราฟแท่งเทียนเพื่อทำนายการเคลื่อนไหวของราคาข้าวโพด" โดยมีรายละเอียดดังต่อไปนี้

### 2.1. แนวคิดและทฤษฎีที่เกี่ยวข้องกับข้าวโพด

ข้าวโพด (ชื่อวิทยาศาสตร์: *Zea mays* Linn.) เป็นพืชตระกูลเดียวกับหญ้ามีลำต้นสูง โดยเฉลี่ย 2.2 เมตร ขนาดเส้นผ่าศูนย์กลางของลำต้น 0.5-2.0 นิ้ว เมล็ดจากฝักใช้เป็นอาหารคนและสัตว์ อีกทั้งยังเป็นธัญพืชที่มีความสำคัญเป็นอันดับสามของโลก รองมาจากข้าวสาลี และข้าว สามารถปลูกได้ทั่วไปในเขตภูมิอากาศอบอุ่น เขตกึ่งร้อนชื้น และพื้นที่ราบเขตร้อน (นพพร และคณะ, 2547) โดยแหล่งปลูกมักกระจายอยู่ตามภูมิภาคต่าง ๆ ของโลก ได้แก่ ประเทศสหรัฐอเมริกา บราซิล เม็กซิโก จีน รวมทั้งในทวีปแอฟริกาใต้ สำหรับประเทศไทยข้าวโพดถือเป็นพืชเศรษฐกิจที่สำคัญ เนื่องจากมีพื้นที่เพาะปลูกครอบคลุมอยู่ทั่วทุกภาค ทำให้สามารถสร้างรายได้เป็นจำนวนมากให้กับประเทศ ข้าวโพดที่ปลูกในประเทศไทยแบ่งออกเป็น 2 กลุ่มใหญ่ ๆ คือ ข้าวโพดฝักสด และข้าวโพดเลี้ยงสัตว์ โดยข้าวโพดฝักสดปลูกเพื่อใช้สำหรับบริโภคเป็นอาหารและส่งออก เนื่องจากผู้บริโภคนิยมรับประทาน และมีคุณค่าทางโภชนาการสูง ส่วนข้าวโพดเลี้ยงสัตว์เป็นพืชที่มีความสำคัญต่ออุตสาหกรรมอาหารสัตว์ ปลูกเพื่อใช้เป็นวัตถุดิบในการผลิตอาหารสัตว์ ซึ่งจังหวัดที่เป็นแหล่งปลูกข้าวโพดที่สำคัญของประเทศไทย ได้แก่ จังหวัดเพชรบูรณ์ นครราชสีมา เลย ลพบุรี และนครสวรรค์ (โชคชัย และเกตุอร, 2561)

สำหรับประเทศไทย ได้นำข้าวโพดเข้ามาเพื่อใช้ในการเลี้ยงสัตว์ตั้งแต่ช่วงหลังสงครามโลกครั้งที่ 1 โดย หม่อมเจ้าสิทธิพร กฤดากร นั้นได้มีการนำข้าวโพดซึ่งเป็นพันธุ์ที่ใช้เลี้ยงสัตว์มาปลูก และทดลองใช้สำหรับการเลี้ยงสัตว์ จนมาถึงช่วงหลังสงครามโลกครั้งที่ 2 ข้าวโพดเริ่มเป็นที่รู้จัก และใช้แพร่หลายขึ้นเนื่องจาก หลวงสุวรรณวาจกกสิกิจ ได้นำการเลี้ยงไก่แบบการค้ามาเริ่มสาธิต และมีการกระตุ้นให้ประชาชนรู้จักใช้ข้าวโพดมากขึ้นกว่าเดิมเนื่องจากกระแสนั้นข้าวโพดราคาค่อนข้างสูงจึงใช้เป็นเพียงส่วนประกอบของอาหารเท่านั้น โดย ณ ปัจจุบันข้าวโพดเป็นที่รู้จักกันทั่วไปทั้งในหมู่มวลผู้เลี้ยงสัตว์ และพลเรือน ซึ่งได้มีการปลูกข้าวโพดต่อปีนั้นจำนวนมาก (สาธิต, 2563)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประโยชน์ของข้าวโพดนอกจากใช้เป็นอาหารโดยตรงของมนุษย์และสัตว์แล้ว ยังสามารถนำมาแปรรูปเป็นผลิตภัณฑ์จากข้าวโพด ทั้งในระดับครัวเรือนและในระดับอุตสาหกรรม เพื่อช่วยถนอมอาหาร เพิ่มความหลากหลายให้กับผลิตภัณฑ์ ทำให้สินค้าเป็นที่ต้องการของผู้บริโภคทั้งในประเทศและต่างประเทศ เพิ่มมูลค่าให้กับผลผลิตทางการเกษตร ส่งเสริมการใช้ทรัพยากรธรรมชาติให้เกิดประโยชน์สูงสุด รวมทั้งสามารถลดปัญหาขยะจากเศษวัสดุเหลือทิ้งทางการเกษตรได้เป็นอย่างดีอีกด้วย

### 2.1.1 ลักษณะทางพฤกษศาสตร์ของข้าวโพด

ข้าวโพดเป็นพืชล้มลุกจำพวกหญ้า ปลูกง่าย อายุสั้น จัดอยู่ในวงศ์ Gramineae มีชื่อวิทยาศาสตร์ว่า *Zea mays* L. และมีชื่อสามัญว่า Corn หรือ Maize โดยข้าวโพดมีลักษณะทางพฤกษศาสตร์ที่สำคัญ (สำนักหอสมุดและศูนย์สารสนเทศวิทยาศาสตร์และเทคโนโลยี, 2561) ดังนี้

1. ราก ข้าวโพดมีระบบรากแบบรากฝอย (Fibrous Root System) แบ่งออกเป็นหลายชนิด เช่น รากชั้นต้น (Primary Root) รากยึดเหนี่ยว (Brace Root) รากด้านข้าง (Lateral Root) และรากฝอย (Root Hair) แต่ไม่มีรากแก้ว (tap root) ส่วนรากที่เจริญมาจากลำต้น เรียกว่า Adventitious Root ซึ่งเกิดจากข้อส่วนล่างของลำต้น ข้อแรกที่เกิดรากชนิดนี้คือ Coleoptilar Node (รูปที่ 2.1) รากเหล่านี้จะเจริญเติบโตอยู่ตลอดชีวิตของข้าวโพด สามารถเจริญแผ่กระจายรอบลำต้น และหยั่งลึกลงไปใต้ดินได้ 2.1-2.4 เมตร



รูปที่ 2.1 ลักษณะรากข้าวโพด (ที่มา: สาริต, 2563)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ลำต้น มีลักษณะค่อนข้างกลม และตั้งตรง ดังรูปที่ 2.2 ประกอบด้วยข้อ ปล้อง และบริเวณข้อมีเนื้อเยื่อเจริญ (Growth Ring) จุดกำเนิดราก (Root primordia) ตา (Bud) และรอยกาบใบ (Leaf Scar) โดยลำต้นมีความสูงตั้งแต่ 30 เซนติเมตรขึ้นไป ขนาดเส้นผ่าศูนย์กลางของลำต้นประมาณ 2.5-5.0 เซนติเมตร ลำต้นบางพันธุ์มีสีม่วง แต่ส่วนใหญ่ลำต้นสดมักมีสีเขียว



รูปที่ 2.2 ลักษณะลำต้นข้าวโพด (ที่มา: สาริต, 2563)

3. ผลข้าวโพด ผลของข้าวโพดเป็นแบบ Caryopsis โดยข้าวโพดจะสะสมแป้งไว้ในส่วนของเอนโดสเปิร์ม การสะสมแป้งจะสิ้นสุดเมื่อข้าวโพดเจริญเติบโตถึงระยะสุกแก่ทางสรีรวิทยา ผลข้าวโพดจะถูกหุ้มไปด้วยกาบบาง ๆ หลายชั้น ฝักอ่อนเป็นสีเขียว เมื่อแก่แล้วจะเปลี่ยนเป็นนวล เรียกว่าเปลือก และฝักมีลักษณะเป็นรูปทรงกระบอก ในหนึ่งฝักจะมีเมล็ดอยู่รอบฝักเรียงเป็นระเบียบรอบแกนกลางของฝัก ดังแสดงในรูปที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 ลักษณะผลข้าวโพด (ที่มา: สาธิต, 2563)

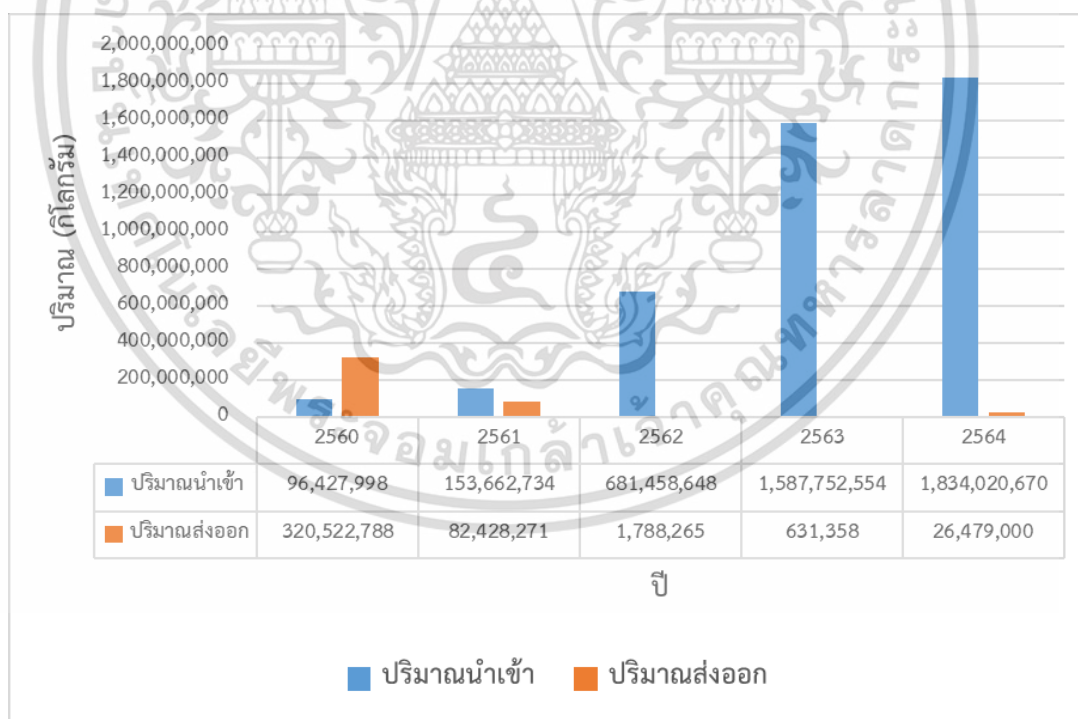
### 2.1.2 ข้าวโพดในประเทศไทย

ปัจจุบันข้าวโพดที่ปลูกในประเทศไทยแบ่งออกเป็น 2 กลุ่มใหญ่ๆ คือ ข้าวโพดฝักสด และข้าวโพดเลี้ยงสัตว์ โดยข้าวโพดฝักสดปลูกเพื่อใช้สำหรับบริโภคและส่งออก ส่วนข้าวโพดเลี้ยงสัตว์ เป็นพืชที่มีความสำคัญต่ออุตสาหกรรมอาหารสัตว์ เนื่องจากใช้เป็นวัตถุดิบในการผลิตอาหารสัตว์ ซึ่งจังหวัดที่เป็นแหล่งปลูกข้าวโพดที่สำคัญของประเทศไทย ได้แก่ จังหวัดนครสวรรค์ อุทัยธานี สระบุรี ลพบุรี ชัยนาทสุพรรณบุรี กาญจนบุรี ราชบุรี และเพชรบุรี เนื่องจากพื้นที่การเกษตรของภาคกลางส่วนใหญ่เป็นพื้นที่นา จึงมีศักยภาพในการปลูกข้าวโพดเพื่อเป็นทางเลือกในการสร้างรายได้ของเกษตรกร โดยเฉพาะในแหล่งที่มีน้ำชลประทาน หรือแหล่งน้ำตามธรรมชาติรวมถึงน้ำใต้ดิน ที่สามารถให้ผลผลิตเฉลี่ยสูงกว่า 1,000 กิโลกรัมต่อไร่ มีต้นทุนการผลิต 3,769–5,150 บาท/ไร่ และมีกำไร 2,639–5,357 บาทต่อไร่ อย่างไรก็ตาม การเพิ่มประสิทธิภาพการผลิตเพื่อเพิ่มผลผลิต ลดต้นทุน รวมถึงการใช้ทรัพยากรในการผลิตอย่างคุ้มค่า และเป็นมิตรกับสิ่งแวดล้อม จำเป็นต้องใช้เทคโนโลยีการผลิตที่เหมาะสมกับสภาพพื้นที่ทั้งในด้านการคัดเลือกพื้นที่ปลูก พันธุ์ การจัดการดิน ปุ๋ย น้ำ การเกษตรกรรม การจัดการศัตรูพืช รวมถึงการเก็บเกี่ยวที่เหมาะสม และข้าวโพดยังมีความสำคัญต่อระบบต่อระบบเศรษฐกิจไทยของประเทศไทยอีกด้วย (กรมวิชาการเกษตร, 2563)

### 2.1.3 ความสำคัญของข้าวโพดต่อระบบเศรษฐกิจไทย

ข้าวโพดเลี้ยงสัตว์เป็นพืชเศรษฐกิจที่สำคัญของไทยที่มีการใช้บริโภคภายในประเทศ และส่งออกมายาวนาน โดยเฉพาะอุตสาหกรรมอาหารสัตว์ ซึ่งช่วยสร้างรายได้ให้แก่เกษตรกร

โดยแต่ละปีไทยมีการส่งออกอาหารสัตว์เพิ่มขึ้น นอกจากนี้ ข้าวโพดเลี้ยงสัตว์ยังเป็นวัตถุดิบสำคัญของอุตสาหกรรมที่เป็นมิตรกับสภาพแวดล้อม เช่น พลาสติกชีวภาพ และเอทานอล ซึ่งแต่ละปีการผลิตเกษตรกรรมส่วนใหญ่จะทำการเพาะปลูกข้าวโพดเลี้ยงสัตว์ 2 ฤดูกาล คือข้าวโพดฤดูฝน จะปลูกในช่วงมีนาคม-ตุลาคม และเก็บเกี่ยวผลผลิตประมาณเดือนมิถุนายน – มกราคมของปีถัดไป และข้าวโพดฤดูแล้ง จะเริ่มปลูกในเดือนพฤศจิกายน – กุมภาพันธ์ ของปีถัดไป และเก็บเกี่ยวในช่วงเดือนมีนาคม – เมษายน โดยคาดว่าความต้องการใช้ข้าวโพดเลี้ยงสัตว์ ในปี 2565 มีปริมาณเพิ่มขึ้นจากปี 2564 เนื่องจากการขยายตัวของภาคอุตสาหกรรมเลี้ยงสัตว์ที่เริ่มฟื้นตัวจากปัญหาการแพร่ระบาดของสถานการณ์การแพร่ระบาดของโควิด-19 และสงครามระหว่างรัสเซียและยูเครน และคาดว่า การส่งออกจะมีแนวโน้มลดลงจากปี 2564 เนื่องจากความต้องการใช้ในภาคอุตสาหกรรมผลิตอาหารสัตว์ภายในประเทศมีเพิ่มมากขึ้น ทำให้การส่งออกข้าวโพดเลี้ยงสัตว์ไปยังตลาดอาเซียนซึ่งเป็นประเทศคู่ค้าที่สำคัญของไทย ได้แก่ ฟิลิปปินส์ ฮังการี อินโดนีเซีย และญี่ปุ่น มีแนวโน้มลดลง ในส่วนการนำเข้า คาดว่าในปี 2565 จะมีแนวโน้มเพิ่มขึ้นจากปี 2564 (กรมเจรจาการค้าระหว่างประเทศ, 2565) โดยรูปที่ 2.4 จะแสดงปริมาณการนำเข้าและส่งออกของข้าวโพด



รูปที่ 2.4 ปริมาณการนำเข้าและส่งออกของข้าวโพดปี 2560-2564

(ที่มา: สำนักงานเศรษฐกิจการเกษตร, 2564)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2. แนวคิดและทฤษฎีที่เกี่ยวข้องกับตลาดซื้อขายสินค้าเกษตรล่วงหน้า

ตลาดสินค้าล่วงหน้า (Futures Exchange) เป็นตลาดรูปแบบใหม่ที่เกิดขึ้นในประเทศไทยเมื่อไม่นานมานี้ แต่สำหรับในต่างประเทศนั้นได้ถูกพัฒนามาเป็นเวลานานแล้ว โดยตลาดซื้อขายสินค้าเกษตรล่วงหน้ามีวิวัฒนาการมาจากตลาดหอกการค้าแห่งนครชิคาโกหรือ Chicago Board of Trade (CBOT) ซึ่งเป็นตลาดสินค้าล่วงหน้าแห่งแรกในสหรัฐ โดยถือกำเนิดจากความร่วมมือกันของกลุ่มพ่อค้าในนครชิคาโก โดยสินค้าชนิดแรกที่มีการซื้อขายโดยการนำมาทำเป็นบัญชีซื้อขายคือ ข้าวโพด ซึ่งเป็นผลผลิตทางเกษตรหลักของเกษตรกรในแถบนั้น และได้ถูกยกให้เป็นแม่แบบของการซื้อขายสินค้าเกษตรในตลาดต่าง ๆ ทั่วโลกในเวลาต่อมา (เริงชัย และคณะ, 2554)

### 2.2.1. บทบาทและหน้าที่ของตลาดสินค้าเกษตรล่วงหน้า

ตลาดสินค้าเกษตรล่วงหน้าจัดตั้งขึ้นเพื่อเสริมการทำหน้าที่ของตลาดสินค้าจริง และเพื่อเพิ่มประสิทธิภาพของระบบตลาดโดยรวม โดยตลาดสินค้าเกษตรล่วงหน้าทำหน้าที่หลัก ๆ อยู่ 3 ประการ ดังนี้

1. ใช้เป็นเครื่องมือสำหรับช่วยบริหารความเสี่ยง (Price Risk Management หรือ Hedging) จุดเริ่มต้นมาจากปัญหาภาคเกษตรโดยเฉพาะในแง่ของปัญหาด้านราคาสินค้าเกษตรที่เกิดจากสถานการณ์ความไม่แน่นอนที่มักจะขึ้นอยู่กัยสภาพดินฟ้าอากาศค่อนข้างมาก ซึ่งส่งผลโดยตรงต่อเกษตรกร รวมถึงผู้ที่เกี่ยวข้องด้วยเช่น ผู้ประกอบการโรงงานสินค้าเกษตร ผู้ส่งออกสินค้าเกษตร ตลอดจนนักลงทุนที่เข้ามาเก็งกำไรในราคาสินค้าเกษตร โดยการเข้ามาทำการซื้อขายสินค้าเกษตรล่วงหน้าในตลาดสินค้าเกษตรล่วงหน้า เช่น เกษตรกร ที่จะขายข้าวโพด ในอีก 3 เดือนข้างหน้า สามารถลดความเสี่ยงจากราคาข้าวโพดที่ตกต่ำโดยการเข้ามาขายล่วงหน้า ในทางตรงกันข้าม ผู้ซื้อสามารถลดความเสี่ยงจากราคาข้าวโพดที่อาจเพิ่มสูงขึ้นในอีก 3 เดือนข้างหน้าโดยการเข้ามาซื้อล่วงหน้าในตลาดสินค้าเกษตรล่วงหน้าได้เช่นเดียวกัน ดังนั้นจะเห็นได้ว่า เกษตรกรหรือผู้ประกอบการจะสามารถใช้ตลาดสินค้าเกษตรล่วงหน้าเพื่อลดความเสี่ยงด้านราคาสินค้าเกษตรได้โดยเข้ามาซื้อขายสินค้าเกษตรล่วงหน้าควบคู่ไปกับกระบวนการค้าในตลาดสินค้าจริงที่ทำกันอยู่แล้วตามปกติ

2. ใช้เป็นทางเลือกใหม่สำหรับนักลงทุน (New Investment Alternative) การลงทุนในตลาดสินค้าเกษตรล่วงหน้าสามารถลดความเสี่ยงจากภาวะเงินเฟ้อ และความผันผวนจากอัตราแลกเปลี่ยนค่าเงินได้ นักเก็งกำไร (Speculator) สามารถเข้ามาทำกำไรจากความผันผวนขึ้นลงของราคาสินค้าเกษตรได้ ซึ่งนักเก็งกำไรอาจจะเป็นบุคคลทั่วไปที่ไม่มีธุรกิจเกี่ยวข้องกับสินค้าเกษตรเหล่านี้ แต่อาศัยการวิเคราะห์ข้อมูลปัจจัยพื้นฐานและปัจจัยทางเทคนิคในการคาดการณ์ทิศทางแนวโน้ม ความเคลื่อนไหวของราคาสินค้าเกษตร จากนั้นจึงเข้ามาทำการซื้อขายทำกำไร โดยอาศัยความผันผวนของราคาสินค้าเกษตรนั้น ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ใช้เป็นเครื่องมือในการค้นพบราคาสินค้าเกษตรในอนาคต (Price Discovery) การซื้อขายสินค้าในตลาดเกษตรล่วงหน้าเป็นการซื้อขายที่มีการเปิดเผยข้อมูลโดยมีสื่อหลายประเภทในการเผยแพร่ราคาซื้อขาย เช่น หนังสือพิมพ์ โทรทัศน์ วิทยุ อินเทอร์เน็ต เหมือนกับราคาหุ้นในตลาดหลักทรัพย์ โดยราคาสินค้าเกษตรดังกล่าวนี้เป็นราคาตามอุปสงค์ และอุปทานของสินค้าเกษตรชนิดนั้น ๆ โดยกลไกตลาดเป็นผู้กำหนดราคาที่เหมาะสมของสินค้าเกษตรชนิดหนึ่ง ๆ ในอนาคต

### 2.2.2. ประโยชน์ของตลาดสินค้าเกษตรล่วงหน้า

1) เกษตรกร ผู้ค้า ผู้ส่งออก ผู้ประกอบการ และผู้บริโภคสามารถทราบถึงแนวโน้มของราคาสินค้าเกษตรในอนาคต ซึ่งจะเป็ประโยชน์ต่อการวางแผนในการเพาะปลูก การเก็บเกี่ยว ผลผลิตรวมถึงการส่งออกผลิตภัณฑ์ที่เกี่ยวข้องกับสินค้าเกษตรต่าง ๆ

2) เกษตรกร ผู้ค้า ผู้ส่งออก ผู้ประกอบการ สามารถใช้ประโยชน์จากการซื้อขายในตลาดสินค้าเกษตรล่วงหน้าเพื่อช่วยลดความเสี่ยงที่เกิดจากการผันผวนของราคาสินค้าเกษตรได้ เช่น เกษตรกรที่เพาะปลูกข้าวโพดที่ต้องการจะขายผลผลิตในอนาคต อาจลดความเสี่ยงจากราคาที่ผันผวน หรือลดภาวะการขาดทุนของราคาข้าวโพดที่ตกต่ำในอนาคตโดยการทำสัญญาขายล่วงหน้าเพื่อตรึงราคาของข้าวโพดไว้ ในขณะที่เดียวกัน ผู้ประกอบที่ต้องการซื้อข้าวโพดเพื่อใช้เป็นวัตถุดิบในการผลิตสินค้า อาจป้องกันความเสี่ยงจากราคาวัตถุดิบที่เพิ่มสูงขึ้นโดยการทำสัญญาซื้อล่วงหน้าเป็นต้น

3) การจัดตั้งตลาดสินค้าเกษตรล่วงหน้าขึ้นนั้น ทำให้เกิดการสร้างความเสมอภาคเท่าเทียมกันในการรับรู้ข่าวสาร ระหว่างเกษตรกรรายย่อย ผู้ผลิต ผู้ประกอบการ ต่าง ๆ ซึ่งสามารถเข้าถึงแหล่งของข้อมูลได้ทันเหตุการณ์ ไม่ว่าจะเป็ราคาซื้อขายในปัจจุบัน ในอดีต และแนวโน้มทิศทางการเคลื่อนไหวของราคาในอนาคต ทำให้ช่วยลดความได้เปรียบเสียเปรียบในการรับรู้ข่าวสารต่าง ๆ ได้

## 2.3. แนวคิดและทฤษฎีที่เกี่ยวข้องกับกราฟแท่งเทียน

### 2.3.1. ที่มาและความหมายของกราฟแท่งเทียน

กราฟแท่งเทียนมีประวัติมาอย่างยาวนานกว่า 200 ปี และมีต้นกำเนิดมาจากประเทศญี่ปุ่น โดยผู้คิดค้นกราฟแท่งเทียน คือ “Homma Munehisa” พ่อค้าข้าวชาวญี่ปุ่น หรือในอีกชื่อหนึ่งว่า “Sokyu Homma” ได้รับการยกย่องอย่างกว้างขวางว่าเป็นหนึ่งในผู้ชั้นนำในยุคแรก ๆ ของการติดตามความเคลื่อนไหวของราคา ก่อนที่กราฟแท่งเทียนจะได้รับความนิยมในชาติตะวันตกนั้น ในช่วงประมาณศตวรรษที่ 17 “Homma Munehisa” ได้ทำการเก็บข้อมูลย้อนหลังของราคาข้าวไว้หลายสิบปีก่อน ด้วยกราฟแท่งเทียนเพื่อศึกษาและวิเคราะห์ข้อมูลเกี่ยวกับจิตวิทยาของคนในการซื้อขาย และ

แนวโน้มของราคาข้าว เรื่องนี้ถูกเผยแพร่ไปสู่ตะวันตกโดยนาย Steve Nison ผู้เขียนหนังสือเรื่อง “Japanese Candlestick Charting Techniques” ตั้งแต่นั้นมากราฟแท่งเทียนก็ได้รับความนิยมไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

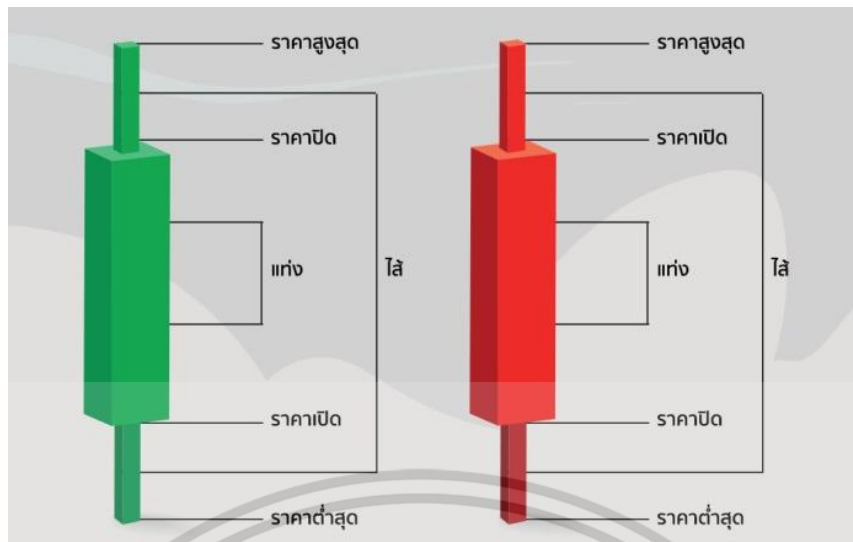
มากขึ้นทุกปี (Lambert, 2009) โดยกราฟแท่งเทียนเป็นที่นิยมอย่างมากสำหรับการวิเคราะห์ทางเทคนิค (Technical Analysis) และมักนิยมใช้กราฟแท่งเทียนเป็นกราฟหลักเพื่อประกอบการพิจารณาแนวโน้มของราคา (Gradojevic and Gencay, 2011)

### 2.3.2. ส่วนประกอบต่างๆของแท่งเทียน

แผนภูมิแท่งเทียนจะแสดงราคาของหุ้น ซึ่งจะมีรูปร่างลักษณะคล้ายกับรูปของแท่งเทียนไขที่คุ้นเคยกันดีดังแสดงในรูปที่ 2.5 โดยในหนึ่งแท่งเทียนจะส่วนประกอบต่าง ๆ ดังนี้ (สิริพร, 2561)

- 1) ราคาเปิด (Open Price) คือราคาซื้อขายแรกที่เกิดขึ้นตั้งแต่เปิดตลาด
- 2) ราคาสูงสุด (High Price) คือราคาที่เคยขึ้นไปสูงที่สุดในช่วงเวลานั้น ๆ
- 3) ราคาต่ำสุด (Low price) คือราคาที่เคยลงไปต่ำที่สุดในช่วงเวลานั้น ๆ
- 4) ราคาปิด (Close Price) คือราคาซื้อขายสุดท้ายที่เกิดขึ้นจากการซื้อขายสิ้นสุดของวัน

โดยมีส่วนประกอบที่สำคัญอีก 2 ส่วน คือ 1) ตัวเทียน (Real Body) เป็นส่วนที่มีลักษณะเป็นแท่งหนาอาจจะเป็นแท่งโปร่ง(บางโปรแกรมอาจแสดงเป็นแท่งเทียนสีเขียวแทนแท่งเทียนแบบโปร่ง หรือแท่งทึบ (บางโปรแกรมแสดงเป็นแท่งเทียนสีแดงแทนแท่งเทียนทึบ) มันคือส่วนต่างของราคาเปิดและราคาปิด แท่งเทียนเป็นสีเขียวแสดงให้เห็นว่าราคาปิดสูงกว่าราคาเปิด(Bullish Candlestick) และในทางกลับกัน แท่งเทียนที่เป็นสีแดงแสดงให้เห็นว่าราคาปิดต่ำกว่าราคาเปิด (Bearish Candlestick) 2) ไส้เทียน (Shadow) คือส่วนที่เป็นเส้นบางซึ่งมี 2 ด้านของแผนภูมิแท่งเทียน โดยด้านบนเรียกว่า ไส้เทียนด้านบน "Upper Shadow" และด้านล่างเรียกว่า ไส้เทียนด้านล่าง "Lower Shadow"

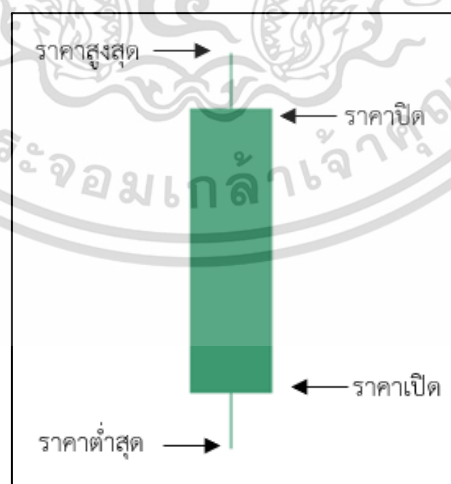


รูปที่ 2.5 ส่วนประกอบของแท่งเทียน (ที่มา : Man, 2020)

### 2.3.3. ลักษณะของแผนภูมิแท่งเทียน

ลักษณะของแผนภูมิแท่งเทียนสามารถแบ่งออกได้เป็น 3 ประเภท ดังนี้ (ไพโรจน์, 2549)

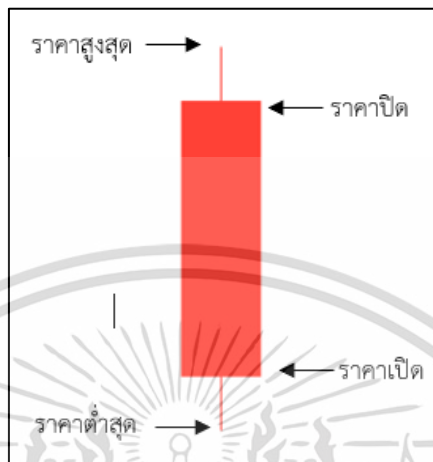
1) แท่งเทียนขาขึ้น (Bullish Candlestick Pattern) เป็นแท่งเทียนที่มีลักษณะโปร่งแสงหรือสีเขียว โดยมีราคาปิดสูงกว่าราคาเปิดของวัน แสดงถึงแนวโน้มที่ดีของราคาหุ้นในวันนั้น เป็นแท่งเทียนสภาวะกระทิง (Bullish Candlestick Pattern) ดังแสดงในรูปที่ 2.6



รูปที่ 2.6 แท่งเทียนขาขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) แท่งเทียนขาลง (Bearish Candlestick Pattern) เป็นแท่งเทียนที่มีลักษณะทึบแสง หรือตัวของแท่งเทียนมีสีดำหรือสีแดง เป็นแท่งเทียนที่แสดงราคาปิดต่ำกว่าราคาเปิดของวันบอกลถึงแนวโน้มที่ไม่ดี เป็นแท่งเทียนสภาวะหมี (Bearish Candlestick Pattern) ดังแสดงในรูปที่ 2.7



รูปที่ 2.7 แท่งเทียนขาลง

3) แท่งเทียนแบบโดจิ (Doji Candlestick Pattern) แท่งเทียนที่ไม่มีตัวของแท่งเทียน เป็นแท่งเทียนที่แสดงว่าราคาเปิด และราคาปิดของวันเท่ากัน โดยราคาดังนั้นจะเป็นราคาสูงสุดหรือราคาต่ำสุด หรือไม่กี่ได้กล่าวคือตัวแท่งเทียนเส้นขีดขวาง มีสภาวะเป็นกลาง โดยการต่อสู้ของแรงซื้อและขายเท่ากัน โดยเมื่อเกิดตอนหุ้นขาขึ้นอาจทำให้แนวโน้มหุ้นลง และเมื่อเกิดตอนหุ้นขาลง อาจทำให้แนวโน้มหุ้นขึ้น

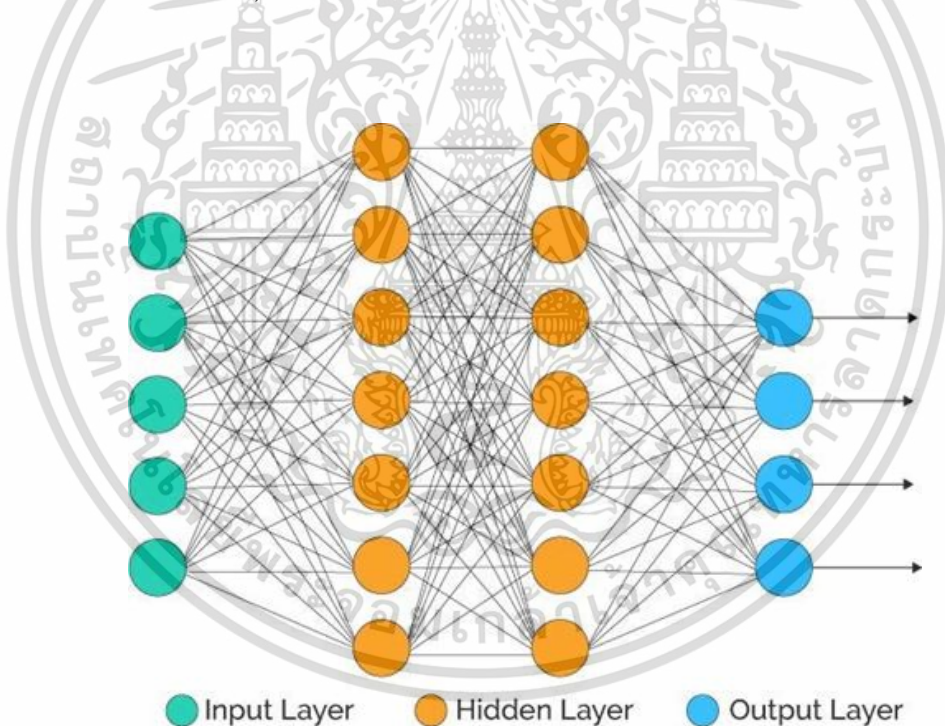


รูปที่ 2.8 แท่งเทียนแบบโดจิ (ที่มา : สิริพร, 2561)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.4. โครงข่ายประสาทเทียม (Artificial Neural Network: ANN)

โครงข่ายประสาทเทียม (ANN) คือแบบจำลองมนุษย์ เป็นโปรแกรมคอมพิวเตอร์ที่ออกแบบมาเพื่อจำลองวิธีที่สมองมนุษย์ประมวลผลข้อมูล โครงข่ายประสาทเทียมเรียนรู้โดยวิธีการฝึกฝนข้อมูลที่เหมาะสมเช่นเดียวกับมนุษย์ ไม่ใช่จากการเขียนโปรแกรม โครงข่ายประสาทเทียมรวบรวมความรู้ โดยการตรวจจ็รูปแบบและความสัมพันธ์ในข้อมูล อย่างเช่น เมื่อเรามองปากกา เราสามารถรับรู้ได้ว่ามันเป็นปากกาเพราะเซลล์ประสาททางชีวภาพในพื้นที่หนึ่งของสมองของเราเคยพบรูปแบบการป้อนข้อมูลที่คล้ายคลึงกันในครั้งก่อน ๆ และได้เรียนรู้ที่จะเชื่อมโยงรูปแบบเฉพาะนั้นกับคำอธิบายวัตถุ เนื่องจากสมองของเรามีเซลล์ประสาทหลายพันล้านเซลล์ที่เชื่อมต่อถึงกันอย่างสมบูรณ์ เราจึงสามารถเรียนรู้และรับรู้รูปแบบการป้อนข้อมูลที่หลากหลายอย่างไม่มีที่สิ้นสุด โดยรูปที่ 2.9 จะแสดงส่วนประกอบหลักของโครงข่ายประสาทเทียมโครงข่ายประสาทเทียม ซึ่งประกอบไปด้วย 3 ส่วนที่สำคัญ ได้แก่ ชั้นนำเข้า (Input Layer) ชั้นซ่อน (Hidden Layer) และชั้นส่งออก (Output Layer) (Kustrin and Beresford, 1999)



รูปที่ 2.9 สถาปัตยกรรมโครงข่ายประสาทเทียม (ที่มา : Mack, 2017)

- 1) ชั้นนำเข้า (Input Layer) เป็นชั้นแรกของโครงข่ายประสาทเทียม โดยที่จะมีนิวรอน (Neuron) ทำหน้าที่เหมือนเซลล์ประสาทของมนุษย์ในการรับข้อมูล และส่งข้อมูล โดยในชั้นนี้จำนวนนิวรอนจะมีจำนวนเท่ากับจำนวนของข้อมูลอินพุต ซึ่งจะทำหน้าที่รับ

เอกสารนี้เป็นเอกสารที่ข้อมูลเข้าสู่ตัวแบบโครงข่ายประสาทเทียมนั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2) ชั้นซ่อน (Hidden Layer) จะเป็นชั้นระหว่างกลางของโครงข่ายประสาทเทียมซึ่งจะมีผลเป็นอย่างมากต่อประสิทธิภาพในการเรียนรู้ของโมเดลโดยชั้นซ่อนนั้นจะมีกี่ชั้นก็ได้โดยระหว่างขั้นตอนการฝึกสอน ข้อมูลจะถูกแปลงโดยค่าน้ำหนักการเชื่อมต่อ (connection weight) ดังนั้นจำนวนการเชื่อมต่อจึงมีผลอย่างมากต่อประสิทธิภาพของโครงข่ายประสาทเทียม เนื่องจากถ้าเซลล์ประสาทที่ซ่อนอยู่น้อยเกินไปจะลดความสามารถในการเรียนรู้ และเซลล์ประสาทที่มากเกินไปจะลดความสามารถในการทำนายผล
- 3) ชั้นส่งออก (Output Layers) ข้อมูลชั้นส่งออกเกิดจากอินพุต คูณด้วยน้ำหนักการเชื่อมต่อก่อนนำมารวมกัน แล้วจึงส่งผ่านฟังก์ชันกระตุ้นเพื่อสร้างเอาต์พุตสำหรับเซลล์ประสาทรุ่น ซึ่งจะเป็นชั้นที่นำข้อมูลที่ได้จากการคำนวณไปใช้ และเป็นชั้นสุดท้ายในโครงข่ายประสาทเทียม ซึ่งจำนวนนิเวรอนของข้อมูลจะเท่ากับจำนวนของผลลัพธ์ในการทำนาย

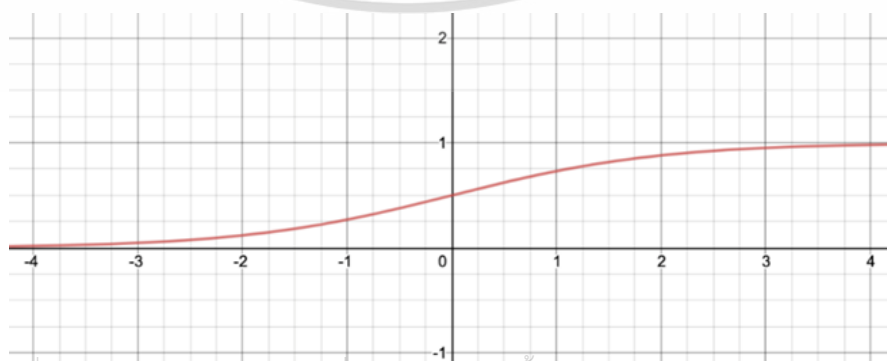
#### 2.4.1. ฟังก์ชันกระตุ้น (Activation Functions)

ฟังก์ชันกระตุ้น เป็นฟังก์ชันในการคำนวณเพื่อทำนายค่าของข้อมูลส่งออก ซึ่งปัญหาในโลกความเป็นจริงมีลักษณะเป็นแบบสมการเส้นตรงน้อยมาก ดังนั้นจึงมีการใช้ฟังก์ชันกระตุ้นเพื่อช่วยกำหนดเอาต์พุตของโครงข่ายประสาทเทียมเช่น ใช่ หรือ ไม่ใช่ โดยจะจับคู่ค่าผลลัพธ์ระหว่าง 0 ถึง 1 หรือ -1 ถึง 1 เป็นต้น (SHARMA, 2017) ซึ่งจะแสดงฟังก์ชันกระตุ้นที่ใช้ในงานวิจัยครั้งนี้

##### 2.4.1.1. ฟังก์ชันกระตุ้นซิกมอยด์ (Sigmoid Activation Function)

ฟังก์ชันกระตุ้นซิกมอยด์ เป็นฟังก์ชันที่มีลักษณะเป็นเส้นโค้งรูปตัว S ดังแสดงในรูปที่ 2.10 และ Output ของ Sigmoid Function มีค่าระหว่าง 0 – 1 ซึ่งมีจุดตัดสินใจอยู่ที่ 0.5 ดังแสดงในสมการที่ 2.10

$$S(x) = \frac{1}{1 + e^{-x}} \quad (2.1)$$

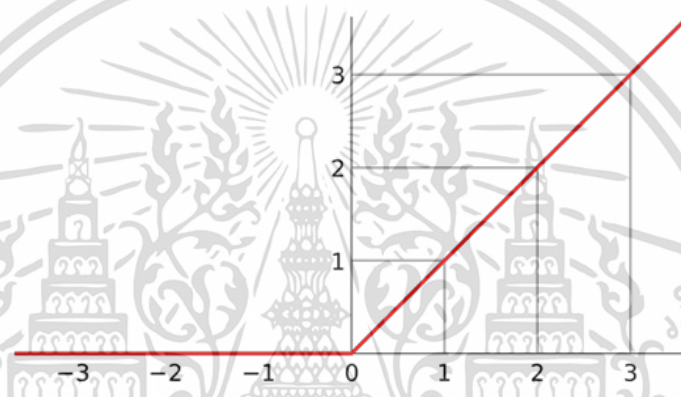


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้ง รูปที่ 2.10 ฟังก์ชันกระตุ้นซิกมอยด์ (ที่มา : ชิตพงษ์, 2563) ครั้งที่มีการนำไปใช้

### 2.4.1.2. ฟังก์ชันกระตุ้นเรกติไฟด์ไลน์เนี่ยยูนิต (Rectified Linear Units: ReLu)

ฟังก์ชันเรกติไฟด์ไลน์เนี่ยยูนิต เป็นฟังก์ชันเส้นตรงที่ถูกปรับแก้ Rectified เรียกว่า Ramp function ซึ่งจะแสดงในรูปที่ 2.11 โดยถ้า Input มีค่าตั้งแต่ 0 ขึ้นไป Output ก็จะมีค่าเท่ากับ ส่วน Input ที่น้อยกว่า 0 จะกำหนด Output เป็น 0 เสมอ ดังแสดงในสมการที่ 2.2

$$S(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \quad (2.2)$$



รูปที่ 2.11 ฟังก์ชันกระตุ้นซิกเรลลู (ที่มา : ชิตพงษ์, 2563)

### 2.4.2. ฟังก์ชันการสูญเสีย (Loss Functions)

ฟังก์ชันการสูญเสียเป็นส่วนหนึ่งที่สำคัญของโครงข่ายประสาทเทียมโดยจะเป็นการเปรียบเทียบผลลัพธ์ค่าที่ได้จากการทำนายและค่าที่แท้จริง ซึ่งสามารถใช้ในการวัดว่าโมเดลมีความผิดพลาดมากน้อยเพียงใด โดยในงานวิจัยนี้เป็นการทำนายทิศทางราคาขึ้นลงของข้าวโพดซึ่งเป็นการจำแนกแบบ 2 กลุ่ม ผู้วิจัยจึงเลือกใช้ฟังก์ชันการสูญเสีย Binary Cross Entropy ดังสมการที่ 2.3 (Guo et al., 2022)

$$BCELoss = -\frac{1}{N} \sum_{i=1}^N [y_i \log(p(y_i)) + (1 - y_i) \log(1 - p(y_i))] \quad (2.3)$$

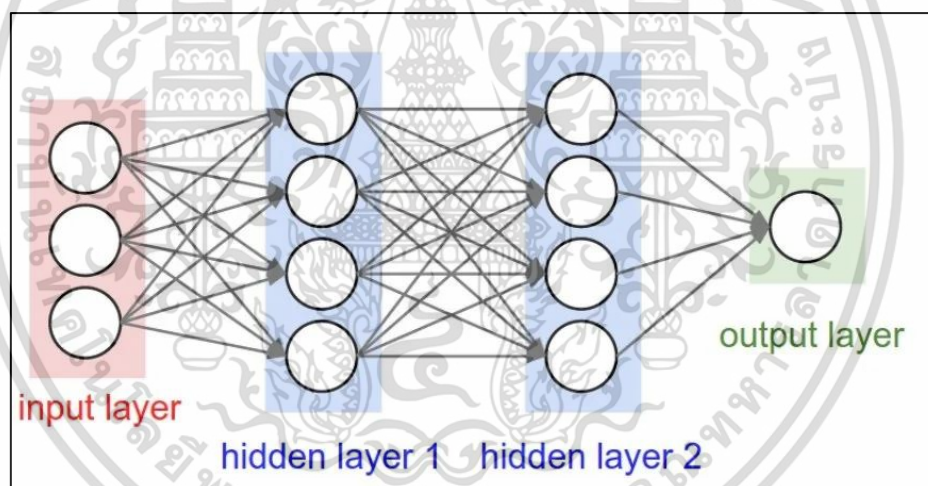
เมื่อ  $N$  คือ จำนวนข้อมูลทั้งหมดที่ใช้ในการฝึกสอน

$y_i$  คือ ค่าที่แท้จริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5. การเรียนรู้เชิงลึก (Deep Learning)

การเรียนรู้เชิงลึก (Deep Learning) เป็นส่วนหนึ่งของการเรียนรู้ของเครื่อง (Machine Learning) คือทำให้คอมพิวเตอร์เกิดการเรียนรู้และนำความรู้นั้นมาใช้งาน ซึ่งถ้ามีจำนวนชั้นที่น้อยหรือมีจำนวนชั้นซ่อน (Hidden Layer) เพียงชั้นเดียวจะเรียกว่าโครงข่ายประสาทเทียมแบบตื้น (Shallow Neural Networks) แต่โครงข่ายประสาทเทียมแบบลึก (Deep Neural Networks) จะเป็นโครงข่ายประสาทเทียมที่มีความลึกหลายชั้นโดยมีชั้นซ่อนมากกว่า 1 ชั้น ดังแสดงในรูปที่ 2.12 โดยการเลือกความลึกในชั้นซ่อนที่เหมาะสมนั้นจะทำให้โมเดลมีประสิทธิภาพที่ดียิ่งขึ้นในการสร้างโมเดล โดยจำนวนนิวรอนในแต่ละชั้นซ่อนคืออีกหนึ่งไฮเปอร์พารามิเตอร์โดยสามารถปรับเปลี่ยนเพื่อหาจำนวนนิวรอนที่ให้ประสิทธิภาพที่ดีที่สุดได้ (Kim and Gofman, 2018) การเรียนรู้เชิงลึกเป็นการประยุกต์ใช้ความรู้ทางด้านสถิติในการสร้างโมเดลสำหรับทำนายผลลัพธ์ของการวิเคราะห์ข้อมูล ซึ่งช่วยให้คอมพิวเตอร์สามารถเรียนรู้จากข้อมูล และสามารถตัดสินใจได้เช่นเดียวกับมนุษย์ (Goodfellow et al., 2016)



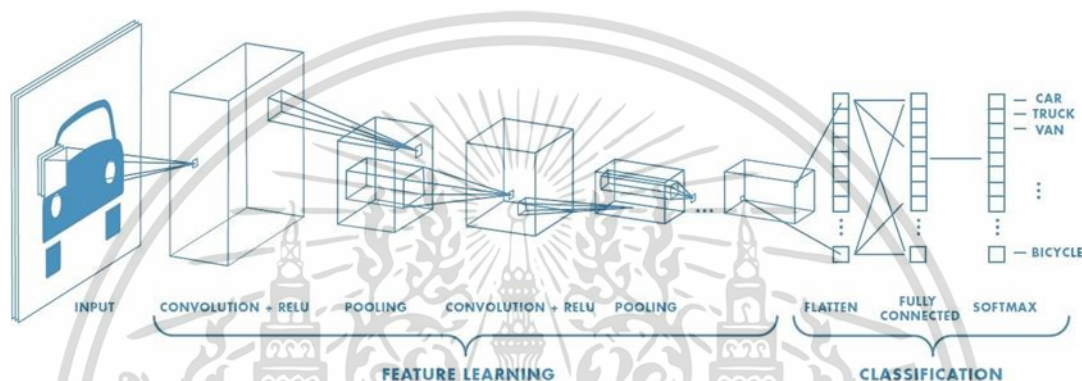
รูปที่ 2.12 ตัวอย่างโครงข่ายประสาทเทียมแบบลึก (ที่มา : Allibhai, 2018)

เทคนิคการเรียนรู้เชิงลึกได้รับความนิยมเนื่องจากคอมพิวเตอร์มีประสิทธิภาพการทำงานที่ดีขึ้น โดยการเรียนรู้เชิงลึกนั้นมีความสามารถในการจดจำ และจำแนกประเภทของรูปภาพเหนือกว่ามนุษย์ โดยจะมีอัลกอริทึมโครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNN) ซึ่งให้ประสิทธิภาพการทำงานที่ดีกับรูปภาพ และช่วยในการสกัดคุณลักษณะต่าง ๆ ของรูปภาพ (Mathew et al. 2021)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.6. โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Network: CNN)

Convolutional Neural Network (CNN) หรือโครงข่ายประสาทเทียมแบบคอนโวลูชัน จัดเป็นการเรียนรู้เชิงลึก (Deep Learning) ที่สามารถจดจำ และจำแนกประเภทคุณลักษณะเฉพาะจากรูปภาพโดยใช้กันอย่างแพร่หลายในการวิเคราะห์ภาพ (Gurucharan, 2022) ซึ่งโครงข่ายประสาทเทียมแบบคอนโวลูชันประกอบด้วยชั้นต่าง ๆ ดังนี้



รูปที่ 2.13 สถาปัตยกรรมโครงข่ายประสาทเทียมแบบคอนโวลูชัน (ที่มา : Saha, 2018)

### 2.6.1. ชั้นนำเข้าข้อมูล (Input Layer)

ชั้นรับข้อมูลเป็นชั้นแรกของระบบการทำงานโดยจะทำหน้าที่ในการรับข้อมูลรูปภาพ รับขนาดความสูง (Height) ความกว้าง (Width) และความลึก (Depth) ของรูปภาพ โดยความสูงและความกว้างจะมีหน่วยเป็นพิกเซล และความลึกจะเป็นเลขตามสี เช่น ภาพสีเทา (Gray Scale) จะมีค่าเท่ากับ 1 และภาพสี (RGB) จะมีค่าเท่ากับ 3

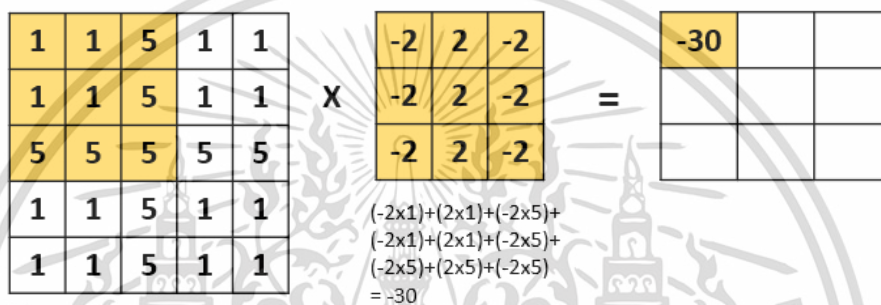
### 2.6.2. ชั้นคอนโวลูชัน (Convolution Layer)

ชั้นคอนโวลูชันเป็นชั้นแรกที่อยู่ถัดจากชั้นรับข้อมูล โดยจะทำหน้าที่ในการแยกคุณสมบัติ (Feature Extract) เช่น สี ขอบ รูปทรง จากข้อมูลที่ได้รับมาจากชั้นรับข้อมูล โดยจะทำการเปรียบเทียบรูปภาพที่ได้รับจากชั้นรับข้อมูลกับตัวกรอง (Filter) โดยที่ขนาดของตัวกรองจะสามารถปรับได้ตามความเหมาะสมเช่น  $3 \times 3$  หรือ  $5 \times 5$  จะทำงานโดยการนำตัวเลขในขนาดของตัวกรอง มาคูณกับตัวเลขในขนาดของรูปภาพที่ตำแหน่งตรงกับตัวกรอง และทำการเลื่อนตามขนาดของก้าวย่าง (Stride) ที่กำหนดจากซ้ายไปขวา บนลงล่าง โดยแสดงตัวอย่างในรูปที่ 2.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก้าวย่าง (Stride) เป็นตัวกำหนดว่าจะเลื่อนตัวกรองด้วยขนาดเท่าไร สามารถกำหนดค่าของ Stride ให้มากขึ้นได้ ถ้าต้องการลดขนาดให้คุณลักษณะ (Feature Map) มีพื้นที่เล็กลง ก็สามารถกำหนดค่าของ Stride ให้เพิ่มขึ้นได้

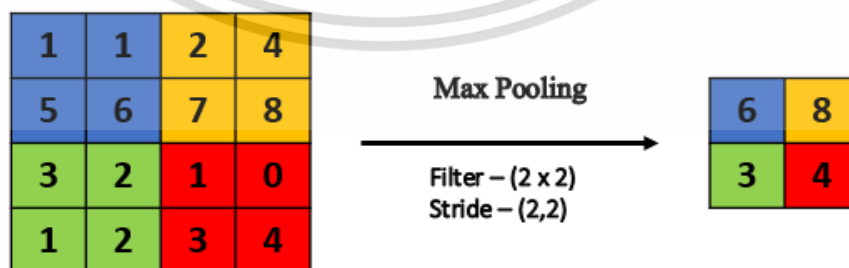
การเติม (Padding) คือการกำหนดค่าให้มีการเติม 0 เข้าไปที่บริเวณขอบของอินพุต โดยปกติแล้วถ้าไม่มีการทำ Padding ผลที่ได้จากการคอนโวลูชันจะทำให้ข้อมูลมีขนาดเล็กกว่าขนาดอินพุต โดยถ้าต้องการข้อมูลบริเวณขอบของรูปภาพจะมีทางเลือกในการทำ Padding เพื่อให้สามารถสกัดข้อมูลทั้งหมดของรูปภาพได้



รูปที่ 2.14 ตัวอย่างการทำงานของคอนโวลูชัน ขนาด 3 x 3

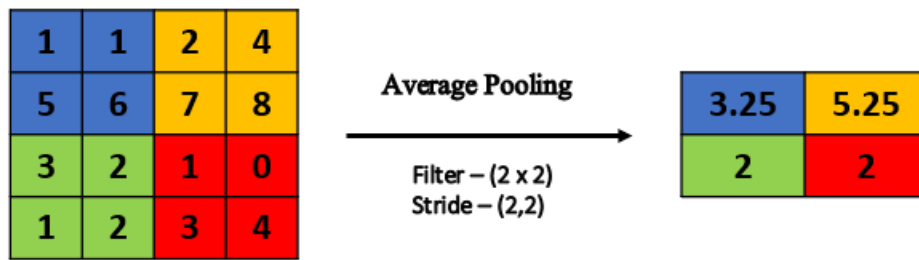
### 2.6.3. ชั้นพูลลิ่ง (Pooling Layer)

ชั้นพูลลิ่งเป็นชั้นที่จะต่อกับชั้นคอนโวลูชัน จะทำหน้าที่ดึงค่าของบริเวณที่สนใจซึ่งแบ่งเป็นการดึงค่าสูงสุด (Max Pooling) หรือ การดึงค่าเฉลี่ย (Average Pooling) จากชั้นคอนโวลูชันเพื่อลดขนาดของข้อมูลลง แต่ยังคงไว้ซึ่งลักษณะเด่นของข้อมูล ดังแสดงในรูปที่ 2.15 และรูปที่ 2.16 ตามลำดับ



รูปที่ 2.15 ตัวอย่างการทำงานของชั้น Pooling ด้วยวิธี Max Pooling

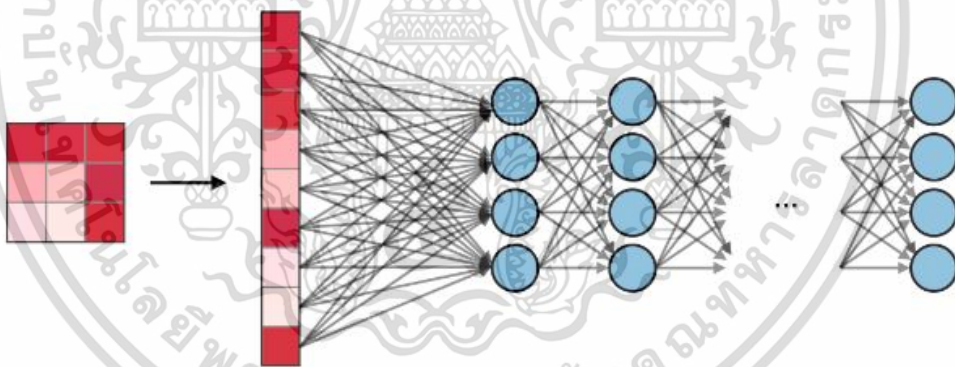
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.16 ตัวอย่างการทำงานของชั้น Pooling ด้วยวิธี Average Pooling

#### 2.6.4. ชั้นเชื่อมโยงสมบูรณ์ (Fully Connected Layer)

ชั้นเชื่อมโยงสมบูรณ์เป็นชั้นที่นำเอาต์พุตของชั้นก่อนหน้ามาทำการ Flatten หรือทำให้ข้อมูลอยู่ในลักษณะของเวกเตอร์หรือเมทริกซ์ขนาด 1 มิติ ตามรูปประกอบที่ 2.17 โดยจะทำการเชื่อมโยงคุณลักษณะพิเศษ (Feature Map) ของชั้นก่อนหน้าซึ่งเป็นการทำซ้ำตั้งแต่ชั้นคอนโวลูชันไปและชั้นพูลลิ่งแล้วนำไปสู่ชั้นผลลัพธ์ ดังนั้นชั้นเชื่อมโยงสมบูรณ์จึงเป็นชั้นสุดท้ายเพราะไม่สามารถนำไปเข้าชั้นคอนโวลูชัน หรือ ชั้นพูลลิ่งได้อีก



รูปที่ 2.17 ตัวอย่างการทำงานของชั้นเชื่อมโยงสมบูรณ์

(ที่มา : Afshine and Shervine, 2018)

#### 2.7. การปรับปรุงประสิทธิภาพ (Optimization)

การปรับปรุงประสิทธิภาพ (Optimization) เป็นกระบวนการค้นหาค่าพารามิเตอร์ที่ดีที่สุดของโมเดลในการเรียนรู้เชิงลึกโดยใช้ อัลกอริทึม เพื่อปรับปรุงค่าพารามิเตอร์ของโมเดลเพื่อลดค่าเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชันการสูญเสีย จนกว่าจะได้ค่าพารามิเตอร์ที่ให้ผลการทำนายที่ดีที่สุด โดยเครื่องมือที่ช่วยในการปรับปรุงประสิทธิภาพโมเดลที่ใช้ในงานวิจัยนี้คือ Adam

Adaptive Moment Estimation (Adam) เป็นอัลกอริทึมที่ใช้ Momentum และ Adaptive Learning Rate เพื่อช่วยให้ การปรับปรุงประสิทธิภาพทำได้เร็วขึ้น นิยมใช้กันอย่างแพร่หลายในการพัฒนา โมเดลการเรียนรู้เชิงลึก และได้รับการเสนอแนะเป็นอัลกอริทึมที่ดีที่สุด ในหลายงานวิจัย (Kingma and Ba, 2015) โดยในสมการที่ 2.4 จะแสดงการปรับค่าน้ำหนักพารามิเตอร์ของ Adam

$$W_{t+1} = W_t - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \quad (2.4)$$

โดย  $\hat{m}_t$  และ  $\hat{v}_t$  เป็นการคำนวณค่าหลังปรับแก้ bias โดยจะแสดงในสมการที่ 2.5 และ 2.6

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \quad (2.5)$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \quad (2.6)$$

ซึ่ง  $m_t$  และ  $v_t$  เป็นการปรับค่า bias ซึ่งจะแสดงในสมการที่ 2.7 และ 2.8

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) \left( \frac{\partial(Loss)}{\partial(w_t)} \right) \quad (2.7)$$

$$v_t = \beta_2 m_{t-1} + (1 - \beta_2) \left( \frac{\partial(Loss)}{\partial(w_t)} \right)^2 \quad (2.8)$$

โดยที่

$W_{t+1}$  คือ ค่าน้ำหนักใหม่หลังจากทำการปรับปรุง ประสิทธิภาพ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ที่  $W_t$  อีก คือ ค่าน้ำหนักปัจจุบันในการปรับปรุงประสิทธิภาพ เอกสารทุกครั้งที่มีการนำไปใช้

$\hat{m}_t$  คือ ค่า  $m_t$  หลังปรับแก้ bias

$\hat{v}_t$  คือ ค่า  $v_t$  หลังปรับแก้ bias

$m_t$  คือ ค่า  $m_t$  ปัจจุบัน

$v_t$  คือ ค่า  $v_t$  ปัจจุบัน

$\beta_1, \beta_2$  คือ ค่าคงที่ในการคำนวณ exponential moving average

$\alpha$  คือ ค่าอัตราการเรียนรู้ (Learning Rate)

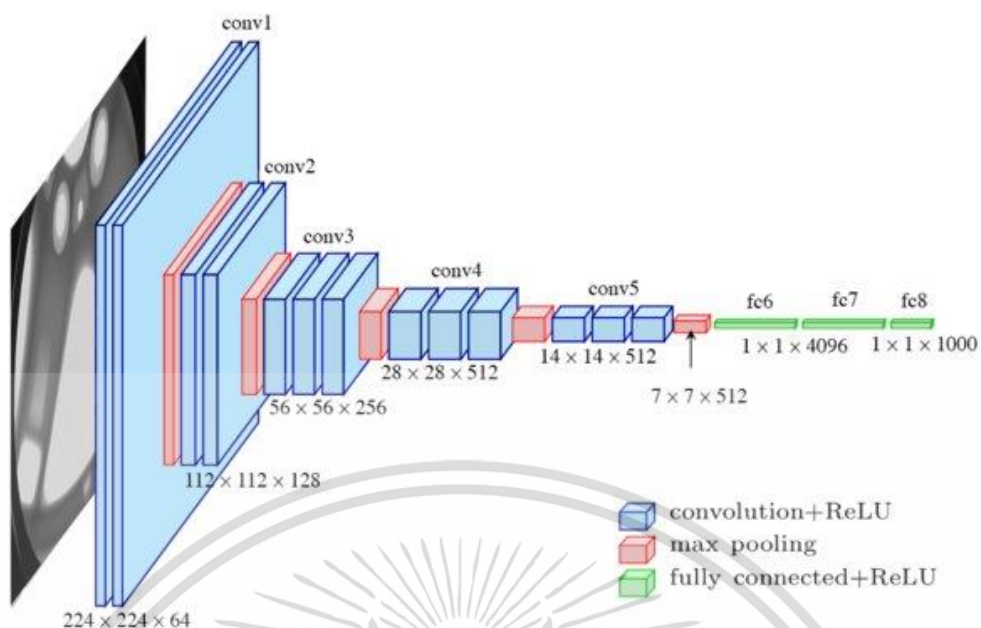
## 2.8. สถาปัตยกรรมโครงข่ายประสาทแบบคอนโวลูชัน (CNN Architecture)

ในปี 2010 ได้มีการจัดการแข่งขัน ImageNet LSVRC-2010 โดยการแข่งขันนี้มีการใช้ชุดข้อมูลภาพจำนวนมากที่สุด ประกอบไปด้วยภาพทั้งหมด 1.2 ล้านภาพ จากอุตสาหกรรมการถ่ายภาพทั่วโลก โดยมีภาพความละเอียดสูงกว่า 15,000,000 ภาพ ซึ่งมีจำนวน 22,000 คลาส ส่วน ILSVRC (The Imagenet Large Scale Visual Recognition Challenge) เป็นการแข่งขันที่จัดโดยบริษัท google โดยใช้ข้อมูล subset ของ ImageNet โดยจะมีประเภทของภาพ 1,000 classes ซึ่งในแต่ละคลาสจะมีภาพอยู่ประมาณ 1,000 ภาพ ทำให้มีชุดข้อมูลของ ILSVRC รวมประมาณ 1,000,000 ภาพ เพื่อใช้ในการฝึกโมเดล ซึ่งได้มีการนำโมเดล CNN มาแข่งขันโดยมีเป้าหมายเพื่อประเมินและเปรียบเทียบประสิทธิภาพของโมเดล ซึ่งถือว่าประสบความสำเร็จในการทดสอบโดยได้รับอัตราความผิดพลาดน้อยติดอันดับ 1 และ 5 ซึ่งดีกว่ารุ่นก่อนหน้าอย่างมาก (Krizhevsky et al., 2012) โดยสถาปัตยกรรมโครงข่ายประสาทแบบคอนโวลูชันที่นำมาใช้ในงานวิจัยนี้คือ VGG-16 และ EfficientNet-B0

### 2.8.1. Visual Geometry Group (VGG)

VGG (Visual Geometry Group) เป็นโมเดลที่ถูกพัฒนาขึ้นโดยนักวิจัยจาก Oxford University ในปี 2014 โดยมีเป้าหมายเพื่อสร้างโมเดลที่มีความลึกมากและทำงานได้ดีกับงานวิเคราะห์ภาพ โดยชื่อของ VGG16 หมายถึงมี 16 ชั้น ซึ่งเป็นเครือข่ายที่ใหญ่และมีพารามิเตอร์ประมาณ 138 ล้าน ดังรูปที่ 2.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.18 สถาปัตยกรรมของ VGG-16 (Ferguson et al., 2017)

VGG16 คือโมเดล VGG ที่มีความลึก 16 ชั้น มี 13 ชั้นเป็น ชั้นคอนโวลูชัน และ 3 ชั้นเป็นชั้นเชื่อมโยงสมบูรณ์ โมเดลนี้ใช้ ชั้นคอนโวลูชันที่มีขนาดตัวกรองเป็น 3x3 และใช้ padding ขนาด 1 pixel เพื่อให้ขนาดของ output หลัง convolution เท่ากับ input หลัง pooling ใช้ Max pooling ขนาด 2x2 และใช้ ReLU activation function หลัง convolution layer ทุกชั้นเพื่อเพิ่มประสิทธิภาพในการทำงานของโมเดล (Simonyan et al., 2015)

### 2.8.2. EfficientNet

สถาปัตยกรรมโครงข่ายประสาทเทียมแบบคอนโวลูชันที่มีชื่อว่า EfficientNet ได้ถูกนำเสนอเป็นครั้งแรกในช่วงต้นปี 2019 โดยที่ความแม่นยำเป็นอันดับที่ 1 คิดเป็นร้อยละ 84.4 ในงานจำแนกประเภท ImageNet เท่านั้น อีกทั้งยังมีพารามิเตอร์น้อยกว่าสถาปัตยกรรมแบบอื่น ๆ ของโครงข่ายประสาทเทียมแบบคอนโวลูชันหลายเท่า (Tan and Le, 2019) โดยจะแสดงโครงสร้างดังต่อไปนี้

#### 2.8.2.1. โครงสร้างเริ่มต้น (Initial Structure)

โมเดล EfficientNet มีหลายขนาด โดยตั้งชื่อจาก B0 ถึง B7 โดยขนาดจะเพิ่มขึ้นตามความซับซ้อนของโมเดล โมเดลเบสไลน์ของ EfficientNet คือโมเดล B0 และจากนั้นจะทำการเปลี่ยนขนาดของโมเดลเพื่อให้ได้ขนาด B1-B7 โดยการใช้การปรับขนาดเพื่อให้โมเดลมีความซับซ้อนและความละเอียดเพิ่มขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

EfficientNet-B0 เป็นโมเดลพื้นฐานของ EfficientNet ที่มีขนาดเล็กและเรียบง่าย จากนั้นจึงนำโมเดลนี้มาปรับขนาดให้ใหญ่ขึ้นเพื่อได้โมเดล B1-B7 โดยการปรับขนาดนี้จะช่วยให้โมเดลมีความซับซ้อนและประสิทธิภาพการทำงานที่ดีขึ้น โดย B1-B7 จะมีความซับซ้อนมากขึ้นเรื่อย ๆ ตามลำดับจนถึง B7 ซึ่งเป็นโมเดลที่ซับซ้อนและใช้ความจำมากที่สุดใน EfficientNet ทั้งหมด โมเดลที่มีขนาดใหญ่ขึ้นนั้นมักจะมีประสิทธิภาพการทำงานที่ดีกว่าและมีความแม่นยำสูงกว่าโมเดลที่มีขนาดเล็กลง แต่ก็ต้องใช้เวลาในการทำงานและทรัพยากรคอมพิวเตอร์ที่มากขึ้นด้วย

สถาปัตยกรรมของแบบจำลองพื้นฐาน B0 เริ่มต้นด้วยชั้นคอนโวลูชันขนาด  $3 \times 3$  จำนวน 1 ชั้น หลังจากนั้นจะมีชั้น Mobile Inverted Bottleneck Layers (MBConv) จำนวน 16 ชั้น โดยชั้น MBConv ช่วยลดความซับซ้อนและระยะเวลาในการทำงานของโมเดล ซึ่งจะมีการสร้างคอนโวลูชันแบบสองระยะ คือ Inverted Residuals และ Linear Bottlenecks

### 2.8.2.2. การปรับขนาดแบบผสม (Compound Scaling)

EfficientNet ใช้สิ่งที่เรียกว่าการปรับขนาดแบบผสมโดยแสดงในรูปที่ 2.19 ซึ่งจะทำการปรับค่าของความลึก (Depth), ความกว้าง (Width), และความละเอียด (Resolution) ของโมเดลพร้อมกัน การปรับขนาดดังกล่าวเกิดขึ้นโดยมีข้อจำกัด ดังแสดงในสมการที่ 2.9 2.10 และ 2.11 ตามลำดับ

$$d = \alpha^\phi \quad (2.9)$$

$$w = \beta^\phi \quad (2.10)$$

$$r = \gamma^\phi \quad (2.11)$$

โดยสมการที่ 2.12 จะแสดงข้อจำกัดของการกำหนดค่า  $\alpha$ ,  $\beta$ ,  $\gamma$

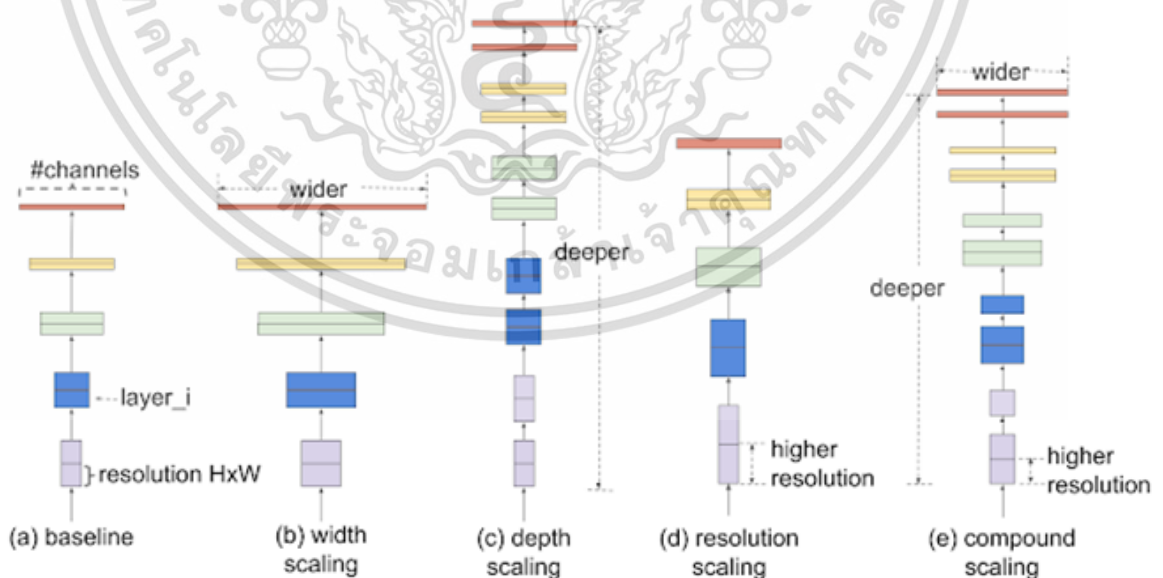
$$s.t. \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2 \quad (2.12)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งาน  $\alpha \geq 1, \beta \geq 1, \gamma \geq 1$ . อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่

- $d$  คือ ความลึกโดยจะส่งผลต่อจำนวนชั้นภายในโมเดล
- $W$  คือ ความกว้างโดยจะส่งผลต่อความลึกของโมเดล
- $r$  คือ ความละเอียดหรือขนาดภาพที่ใช้ในการฝึกสอนโมเดล
- $\alpha$  คือ ค่าคงที่ที่สามารถหาค่าได้ของความลึก
- $\beta$  คือ ค่าคงที่ที่สามารถหาค่าได้ของความกว้าง
- $\gamma$  คือ ค่าคงที่ที่สามารถหาค่าได้ของความละเอียด
- $\phi$  คือ ค่าสัมประสิทธิ์ที่ใช้ในการปรับขนาดแบบผสม

หลังจากที่ได้กำหนดค่าตัวแปร  $\phi$  เป็น 1 แล้ว ผู้วิจัยพบว่าค่าที่ดีที่สุดสำหรับตัวแปร  $\alpha$ ,  $\beta$  และ  $\gamma$  ในการปรับขนาดแบบผสมนี้คือ 1.20, 1.10 และ 1.15 ตามลำดับ ซึ่งสร้างโมเดล EfficientNet-B0 ขึ้นมาได้ โดยถ้าหากค่า  $\alpha$ ,  $\beta$  และ  $\gamma$  ถูกตั้งค่าไว้เป็นค่าคงที่โดยเพิ่มค่าตัวแปร  $\phi$  โดยใช้สมการที่ได้อธิบายไว้ในสมการ 2.9, 2.10 และ 2.11 ก็จะได้โมเดล EfficientNet-B1 ถึง EfficientNet-B7 ตามลำดับ (Tan and Le, 2019)



รูปที่ 2.19 การปรับขนาดแบบผสม (Tan and Le, 2019)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.9. การวัดประสิทธิภาพของโมเดล (Model Performance)

เมทริกซ์ความสับสน (Confusion Matrix) คือ ตารางในการวัดความสามารถของการเรียนรู้ของเครื่อง (Machine learning) ในการแก้ปัญหาการจำแนกประเภท (Classification) โดยมีอักษรย่อ ดังแสดงในตารางที่ 2.6

- True Positive (TP) คือ ทำนายออกมาว่า "บวก" และ มีค่าเป็น "บวก"
- True Negative (TN) คือ ทำนายออกมาว่า "ลบ" และ มีค่าเป็น "ลบ"
- False Positive (FP) คือ ทำนายออกมาว่า "บวก" และ มีค่าเป็น "ลบ"
- False Negative (FN) คือ ทำนายออกมาว่า "ลบ" และ มีค่าเป็น "บวก"

ตารางที่ 2.6 เมทริกซ์ความสับสน (Confusion Matrix) ขนาด 2x2

		Predicted	
		Positive	Negative
Actual Values	Positive	TP	FN
	Negative	FP	TN

โดยการคำนวณค่าวัดผลต่าง ๆ มีดังนี้

- ค่าความแม่นยำ (Accuracy) คือ การหาความถูกต้องของแบบจำลองซึ่งพิจารณาผลรวมทุกค่า  $y$  ที่เกี่ยวข้อง

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

- ค่าความเที่ยง (Precision) คือ อัตราส่วนของการทำนายที่คาดการณ์อย่างถูกต้องต่อการทำนายทั้งหมดที่คาดการณ์ไว้

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

- ค่าความระลึก (Recall) คือ อัตราส่วนของการทำนายไว้อย่างถูกต้องกับค่าจริงทั้งหมดในแต่ละประเภท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ  $Recall = \frac{TP}{TP + FN}$  ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ค่าความถ่วงดุล (F1 - Score) คือ ค่าเฉลี่ยถ่วงน้ำหนักของค่าความแม่นยำ (Precision) และค่าความระลึก (Recall)

$$F1 \text{ Score} = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)}$$

## 2.10. งานวิจัยที่เกี่ยวข้อง (Related Research)

Padmavathi et al. (2022) ได้ทำการศึกษาเนื้องอกในสมองที่เป็นเนื้องอกที่ร้ายแรงหรือไม่ร้ายแรงโดยเกิดจากการเจริญเติบโตของเซลล์ในสมองที่ไม่ปกติ การเกิดเนื้องอกหลักเกิดจากการเจริญเติบโตโดยตรงจากเนื้อเยื่อสมอง ในขณะที่เนื้องอกรองเกิดจากการแพร่กระจายมาจากส่วนอื่นของร่างกายไปยังสมอง ขึ้นอยู่กับชนิดขนาดและตำแหน่งของเนื้องอก จะมีตัวเลือกในการรักษาในหลายวิธี รังสีแพทย์ใช้ภาพ MRI เพื่อการจำแนกเนื้องอก เนื่องจากความซับซ้อนของเนื้องอกในสมองและคุณลักษณะของมันการตรวจสอบด้วยวิธีการดูแล้วอาจเกิดความผิดพลาดได้ เพื่อช่วยแพทย์ทั่วโลก ผู้วิจัยได้เสนอวิธีการใช้เทคนิคการเรียนรู้เชิงลึก (Deep Learning) โดยใช้โครงข่ายประสาทเชิงคอนโวลูชัน (Convolutional Neural Networks) และการโอนย้ายเรียนรู้ (Transfer Learning) ในการสร้างโมเดลที่สามารถจำแนกเนื้องอกในสมองได้อย่างแม่นยำ ในงานวิจัยนี้ได้จำแนกภาพ MRI เป็น 4 ประเภท คือ Glioma, No Tumor, Meningioma, และ Pituitary โดยทำการฝึกโมเดลด้วย EfficientNet B0 และได้ทำการเปรียบเทียบค่าความแม่นยำกับโมเดล VGG-16, VGG-19 และ ResNet-50 จากผลการทดลองพบว่าเทคนิคการโอนย้ายการเรียนรู้มีประสิทธิภาพเมื่อชุดข้อมูลมีจำกัด และโมเดล EfficientNet B0 สามารถทำนายเนื้องอกในสมองได้ด้วยความแม่นยำถึง 99%

Liu et al. (2020) ได้นำเสนอวิธีการใช้การเรียนรู้เชิงลึก (Deep Learning) ในการจำแนกโรคไขข้ออักเสบจากภาพไขข้ออักเสบ โดยใช้เทคนิคการถ่ายทอดการเรียนรู้ (Transfer Learning) จากโมเดล EfficientNet ซึ่งช่วยเพิ่มความแม่นยำและความเร็วในการรู้จำของโมเดลสำหรับชุดข้อมูลโรคไขข้ออักเสบขนาดเล็ก กระบวนการทำงานของงานวิจัยประกอบไปด้วยขั้นตอนการทำความสะอาดข้อมูลและการเพิ่มข้อมูลภาพ (Data Augmentation) เพื่อให้ได้ข้อมูลภาพที่หลากหลายมากขึ้น จากนั้นทำการถ่ายทอดโมเดลที่ได้รับการฝึกสอนก่อนหน้านี้จาก EfficientNet ที่ฝึกสอนด้วยชุดข้อมูล ImageNet เข้ากับวิธีการในงานวิจัยนี้โดยทำการแทนที่ชั้นสุดท้ายของตัวจำแนก EfficientNet ด้วยตัวจำแนกแบบ Softmax ที่มีคลาส 8 คลาส และฝึกสอนโมเดลทั้งหมดเพื่อทำนายโรคไขข้ออักเสบ ในงานวิจัยนี้ได้ทำการทดสอบโดยใช้โมเดล VGG-16, Inception-v3, และ Resnet-50 เพื่อใช้ในการเปรียบเทียบ

เอกสารนี้เป็นเอกสารที่คัดลอกและแสดงให้เห็นว่าค่าความแม่นยำในการรู้จำของโมเดล EfficientNet B0 ไม่ต่ำกว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สูงกว่าโมเดลอื่น ๆ ซึ่งให้ค่าสูงสุดถึง 98.52% ซึ่งสามารถนำไปประยุกต์ใช้ในการผลิตทางการเกษตรได้

ธนัช และวารารภรณ์. (2022) ได้ทำการศึกษาการจำแนกภาพรถที่เสียหายและไม่เสียหายโดยใช้เทคนิคการเรียนรู้เชิงลึกเพื่อพัฒนาโมเดลการจำแนกความเสียหายของรถด้วยโครงข่ายประสาทเชิงคอนโวลูชัน (Convolutional Neural Network, CNN) โดยโมเดลที่ใช้ในการพัฒนาจะเป็น VGG16, ResNet50, และ InceptionV3 โดยใช้เทคนิคการโอนย้ายเรียนรู้ (Transfer Learning) การจำแนกแบ่งออกเป็นสองหมวดหมู่ คือรถที่เสียหาย และรถที่ไม่เสียหาย ในการพัฒนาและปรับปรุงโมเดลใช้ชุดข้อมูลจาก Kaggle โดยโมเดลได้ถูกปรับพารามิเตอร์ได้แก่ batch size, learning rate เพื่อให้มีประสิทธิภาพที่ดีที่สุด ผลการทดลองแสดงให้เห็นว่าความแม่นยำของโมเดล VGG16, InceptionV3, และ ResNet50 มีค่าเท่ากับ 0.83, 0.81 และ 0.68 ตามลำดับ

สุริยะ. (2020) ได้นำเสนอวิธีการเรียนรู้เชิงลึกมาจำแนกภาพการตกผลึกน้ำตาลจากกระบวนการน้ำตาลเคี้ยวตกผลึก ซึ่งในการทดแทนแรงงานที่มีประสิทธิภาพการจำแนกผลึกน้ำตาลระหว่างกระบวนการผลิตอย่างถูกต้องเป็นพื้นฐานสำหรับการควบคุมกระบวนการเคี้ยวตกผลึกน้ำตาล โดยได้มีนำแนวคิดของการเรียนรู้เชิงลึกมาทำการสร้างโมเดล Simple DCNNs, Inception-v3, VGG16 และ ResNet50-v2 และนำมาฝึกฝนผ่านชุดข้อมูลภาพผลึกน้ำตาลจากกระบวนการผลิตนี้แล้วทำการวัดและเปรียบเทียบประสิทธิภาพ โดยโมเดล VGG-16 ที่ผ่านการปรับแต่ง มีความแม่นยำในการจำแนกผลึกน้ำตาลระหว่าง 80% ถึง 94% ของภาพผลึกน้ำตาลโดยแยกตามสภาวะการเกิดผลึกทั้ง 4 ประเภท ซึ่งผลการทดลองแสดงให้เห็นว่าโมเดล VGG-16 มีความแม่นยำมากกว่าโมเดลอื่น และสามารถใช้ตรวจสอบการตกผลึกของกระบวนการผลิตน้ำตาล

Kusuma et al. (2019) ได้สำรวจความสามารถในการพยากรณ์ตลาดหุ้นโดยใช้ Deep Convolutional Network และกราฟแท่งเทียน (Candlestick Charts) โดยจากข้อมูลอนุกรมเวลาที่ผ่านมา ในการวิเคราะห์ความสัมพันธ์ระหว่างช่วงเวลาต่าง ๆ กับการเคลื่อนไหวของตลาดหุ้นตลาดหุ้นในประเทศไต้หวันและอินโดนีเซีย ผู้วิจัยได้แบ่งข้อมูลที่ใช้ในการสร้างแผนภูมิแท่งเทียนเป็นกรอบเวลา 3 ช่วง ได้แก่ 5 10 และ 20 วันซื้อขาย นอกจากช่วงเวลาแล้วยังแบ่งเป็นที่ใช้ตัวบ่งชี้ปริมาณและไม่ใช้ตัวบ่งชี้ปริมาณ โดยกราฟแท่งเทียนเหล่านี้จะถูกนำมาเป็นอินพุตในการฝึกฝนโมเดล และงานวิจัยนี้ใช้โมเดลการเรียนรู้เชิงลึกและโมเดลการเรียนรู้ของเครื่องเพื่อนำมาเปรียบเทียบ ซึ่งโมเดลที่ใช้ในงานวิจัยนี้คือ CNN, VGG, ResNet-50, Random Forest และ KNN ซึ่งผลลัพธ์ของงานวิจัยนี้โมเดล CNN ที่กรอบเวลา 20 ที่ไม่มีตัวบ่งชี้ปริมาณมีประสิทธิภาพของค่าความแม่นยำที่ดีที่สุดซึ่งให้ค่า

เอกสารนี้ ความแม่นยำ 92.2% และ 92.1% สำหรับชุดข้อมูลตลาดหุ้นในประเทศไต้หวันและอินโดนีเซียตามลำดับ  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

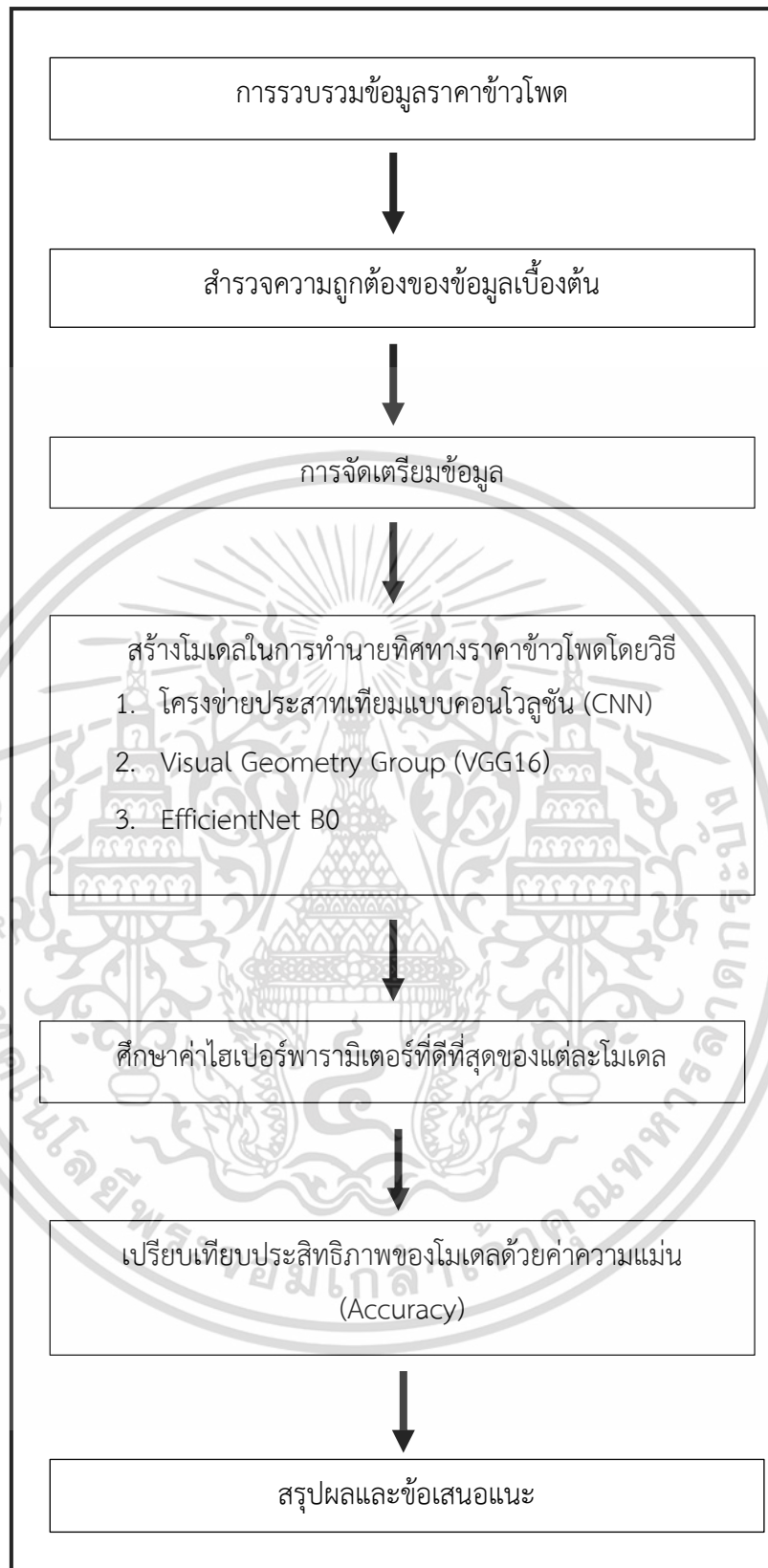
### วิธีการดำเนินงานวิจัย

งานวิจัยนี้ผู้วิจัยสนใจเกี่ยวกับการนำเทคนิคการเรียนรู้เชิงลึกเพื่อทำนายทิศทางของราคาข้าวโพดที่มีการซื้อขายล่วงหน้าในตลาดหอกการค้าแห่งนครชิคาโก (CBOT) โดยใช้รูปภาพกราฟแท่งเทียน ผู้วิจัยได้นำแนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้องมาดำเนินการดังต่อไปนี้

#### 3.1. กรอบแนวคิดการวิจัย

แนวคิดหลักของงานวิจัยนี้คือการนำเทคนิคการเรียนรู้เชิงลึกเพื่อทำนายทิศทางของราคาข้าวโพดที่มีการซื้อขายล่วงหน้าในตลาดหอกการค้าแห่งนครชิคาโก (CBOT) โดยใช้รูปภาพกราฟแท่งเทียน ซึ่งโมเดลที่ใช้ในการทำนายทิศทางราคาข้าวโพดได้แก่ โครงข่ายประสาทเทียมแบบคอนโวลูชันที่ออกแบบโดยผู้วิจัย (Customized CNN), Visual Geometry Group (VGG-16) และ EfficientNet-B0 เพื่อให้เกิดความเข้าใจในข้อมูล และการจัดเตรียมข้อมูลในการวิเคราะห์ สามารถแบ่งออกเป็นขั้นตอนต่าง ๆ ดังรูปที่ 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 กรอบแนวคิดการวิจัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2. การเก็บรวบรวมข้อมูล

ข้อมูลที่ใช้ในงานวิจัยครั้งนี้เป็นข้อมูลรายวัน โดยเก็บรวบรวมจากเว็บไซต์ bloomberg ตั้งแต่วันที่ 1 มกราคม พ.ศ.2543 ถึง วันที่ 1 กุมภาพันธ์ พ.ศ.2566 จำนวน 6,023 วัน ดังแสดงในรูปที่ 3.2

Dates	OPEN	HIGH	LOW	LAST
2000-01-03	204.5	205.75	200.5	200.75
2000-01-04	200.75	203.5	200.75	203
2000-01-05	202.5	203.25	201	203
2000-01-06	203	205.75	203	203.75
2000-01-07	203.25	207.75	202.5	207
2000-01-10	206.25	209.25	204.75	208.5
2000-01-11	208	209.25	206.25	207.25
2000-01-12	207	216.75	206.75	215.5
2000-01-13	214.5	221.5	214.5	220.75
2000-01-14	220	221.75	218.25	219
2000-01-17	220	221.75	218.25	219
2000-01-18	217	220.5	215.5	220

รูปที่ 3.2 ตัวอย่างราคาข้าวโพด (รายวัน)

### 3.3. การจัดเตรียมข้อมูล (Data Preprocessing)

#### 3.3.1. การจัดการข้อมูลเบื้องต้น

ในงานวิจัยนี้เป็นการสร้างรูปภาพกราฟแท่งเทียน และให้ผลเฉลยโดยใช้ผลต่างราคาปิดของวันถัดไปกับวันปัจจุบัน ดังนั้นทำให้มีรูปที่ไม่มีผลเฉลยซึ่งไม่สามารถนำเข้าไปโมเดลได้ จึงต้องทำการลบรูปนั้นเพื่อไม่ให้เกิดปัญหาขณะโปรแกรมประมวลผล

#### 3.3.2. การสร้างรูปภาพกราฟแท่งเทียน

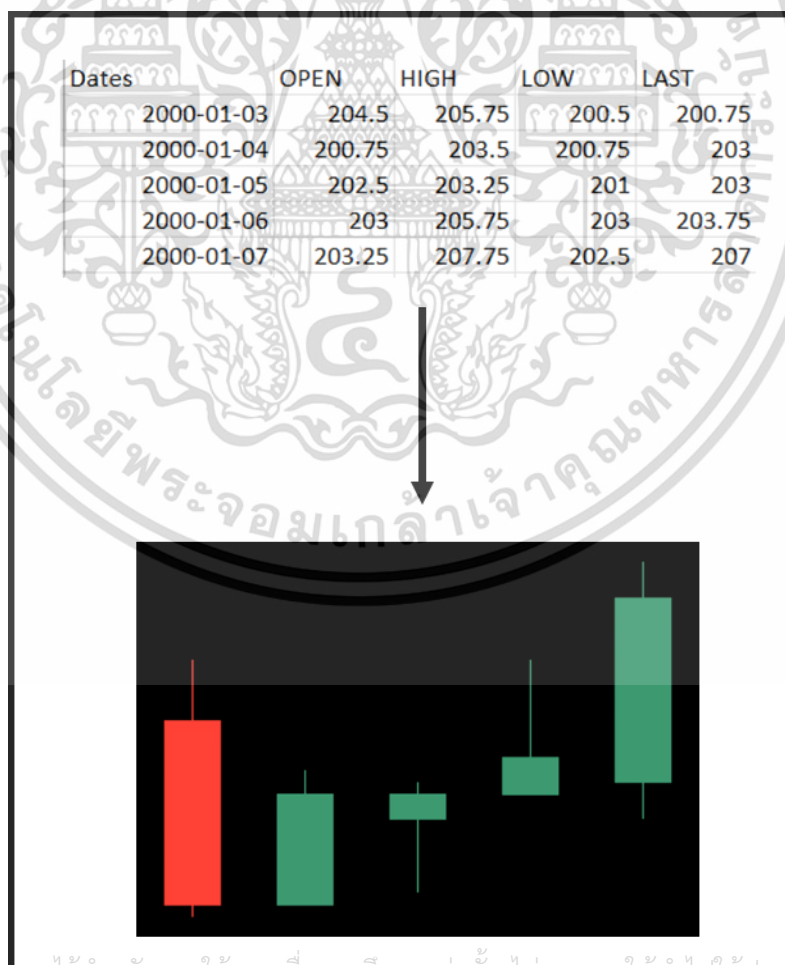
ในการสร้างรูปภาพกราฟแท่งเทียนนั้นจะใช้ข้อมูลราคาข้าวโพดซึ่งประกอบไปด้วยราคาเปิด (Open) ราคาสูงสุด (High) ราคาต่ำสุด (Low) และราคาปิด (Close) ดังแสดงในตารางที่ 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

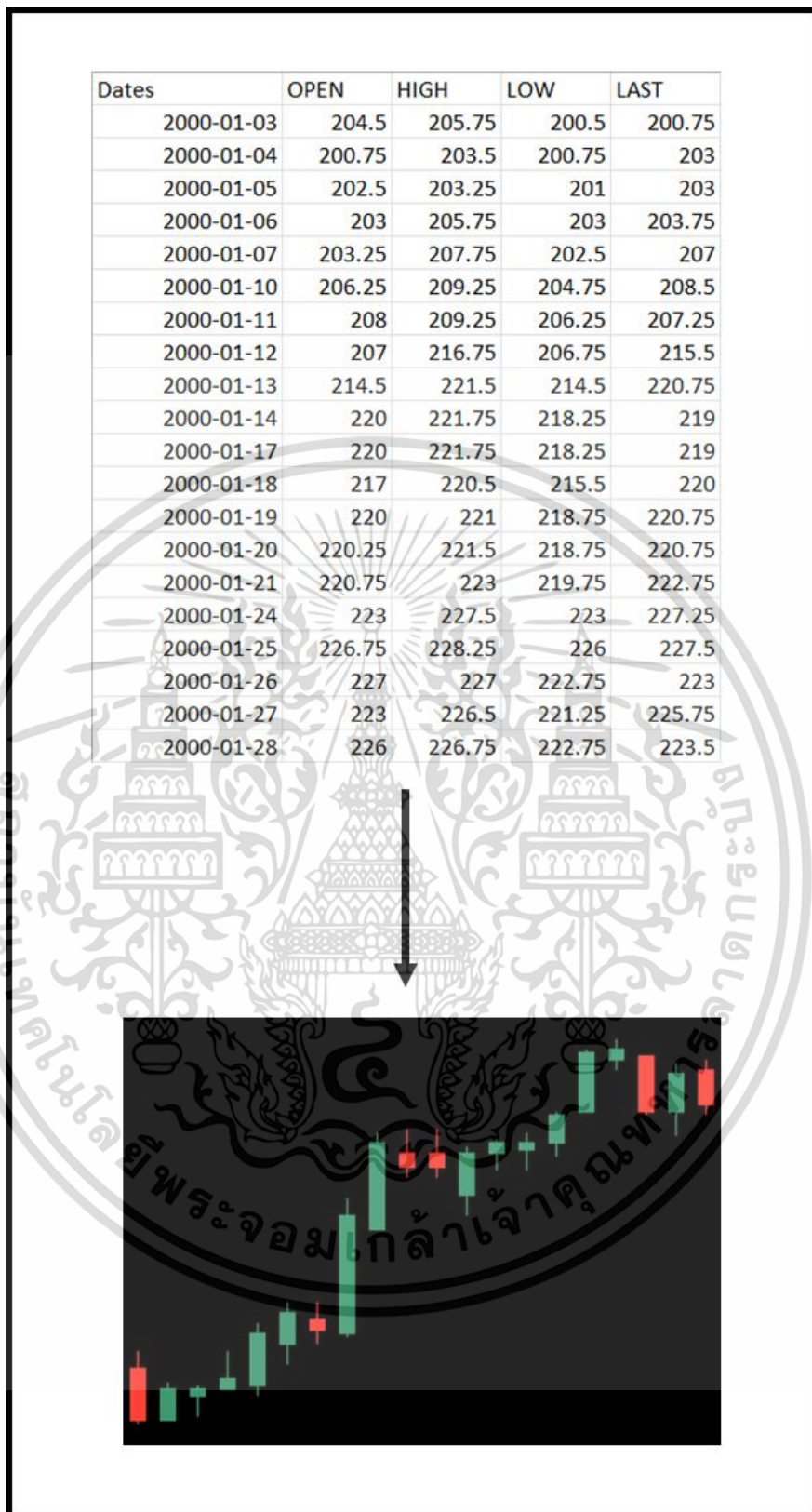
ตารางที่ 3.1 ตัวแปรที่ใช้ในการสร้างกราฟแท่งเทียน

ลำดับ	ชื่อตัวแปร	คำอธิบาย
1	Open	ราคาเปิด
2	High	ราคาสูงสุด
3	Low	ราคาต่ำสุด
4	Last (Close)	ราคาปิด

จากนั้นผู้วิจัยได้สร้างรูปภาพซึ่งมีรอบเวลา 5 และ 20 วัน โดยจะพิจารณาเป็นการซื้อขายรายสัปดาห์ และการซื้อขายรายเดือนตามลำดับ ซึ่งใช้ไลบรารี Plotly ในภาษา Python ในการสร้างรูปภาพขึ้นมาซึ่งจะแสดงแท่งเทียนการซื้อขายของวันนั้น ๆ โดยแท่งสีเขียวหมายถึงราคาปิดมากกว่าราคาเปิด และแท่งสีแดงหมายถึงราคาเปิดมากกว่าราคาปิด ดังแสดงในรูปที่ 3.3 และรูปที่ 3.4 ตามลำดับ



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ไปยังระบบงานด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งรูปที่ 3.3 ตัวอย่างการสร้างกราฟแท่งเทียน (กรอบเวลา 5 วัน) รังที่มีการนำไปใช้



รูปที่ 3.4 ตัวอย่างการสร้างกราฟแท่งเทียน (กรอบเวลา 20 วัน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.3. การแปลงราคาข้าวโพด

เนื่องจากงานวิจัยนี้เป็นการทำนายทิศทางของราคาข้าวโพด ดังนั้นจำเป็นที่จะต้องแปลงค่าของราคาข้าวโพดให้อยู่ในรูปของทิศทางราคา ก่อน โดยสามารถคำนวณได้ดังสมการที่ 3.1 (Lee, 2009)

$$R_t = Last(t + 1) - Last(t) \quad (3.1)$$

เมื่อ

$R_t$  คือ ค่าผลต่างของราคาปิดวันถัดไปกับวันปัจจุบัน

$Last(t + 1)$  คือ ราคาปิด ณ วันถัดไป

$Last(t)$  คือ ราคาปิด ณ วันปัจจุบัน

จากนั้นสามารถสรุปได้ดังตารางที่ 3.2 และกำหนดให้เป็นผลเฉลยของรูปภาพแต่ละรูป

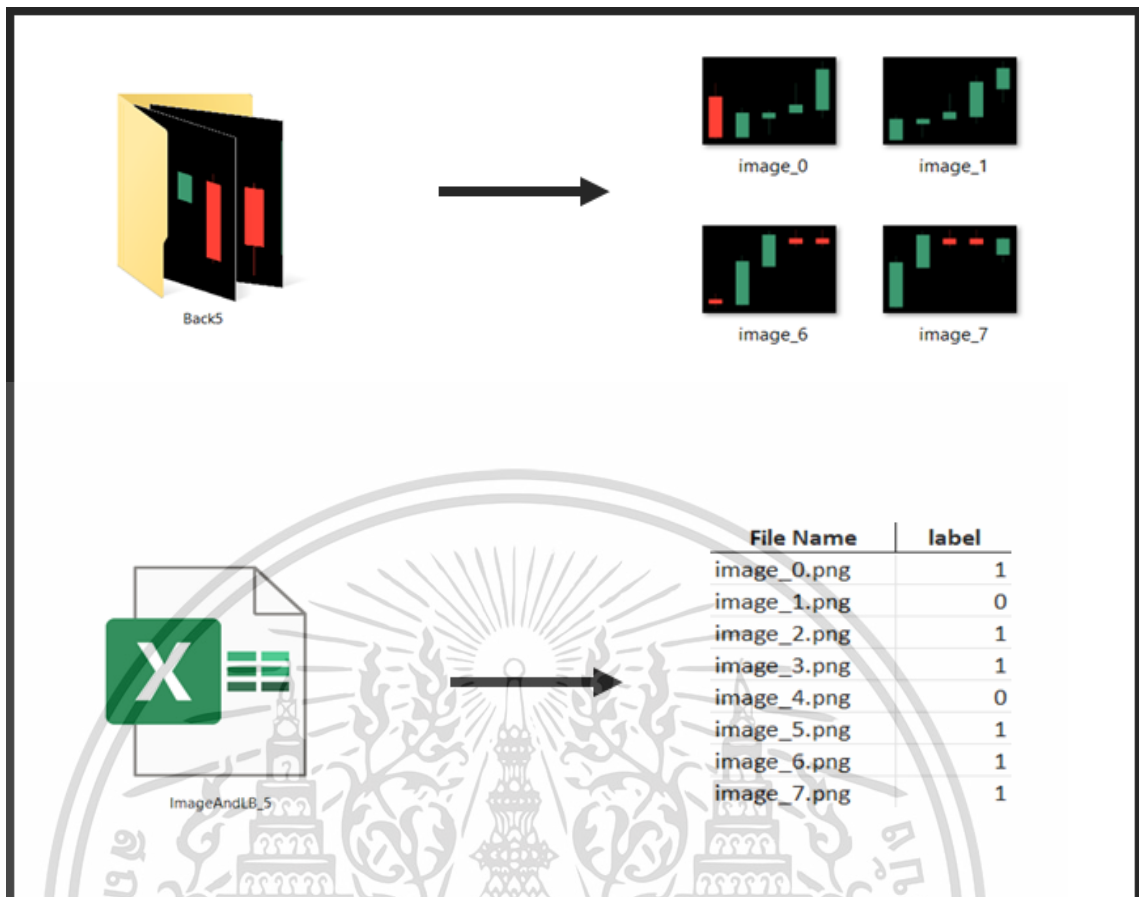
ตารางที่ 3.2 การแปลงราคาข้าวโพด

ก่อนแปลง	หลังแปลง
$R_t \geq 0$	1 (Bullish)
$R_t < 0$	0 (Bearish)

### 3.3.4. การจัดเก็บข้อมูล

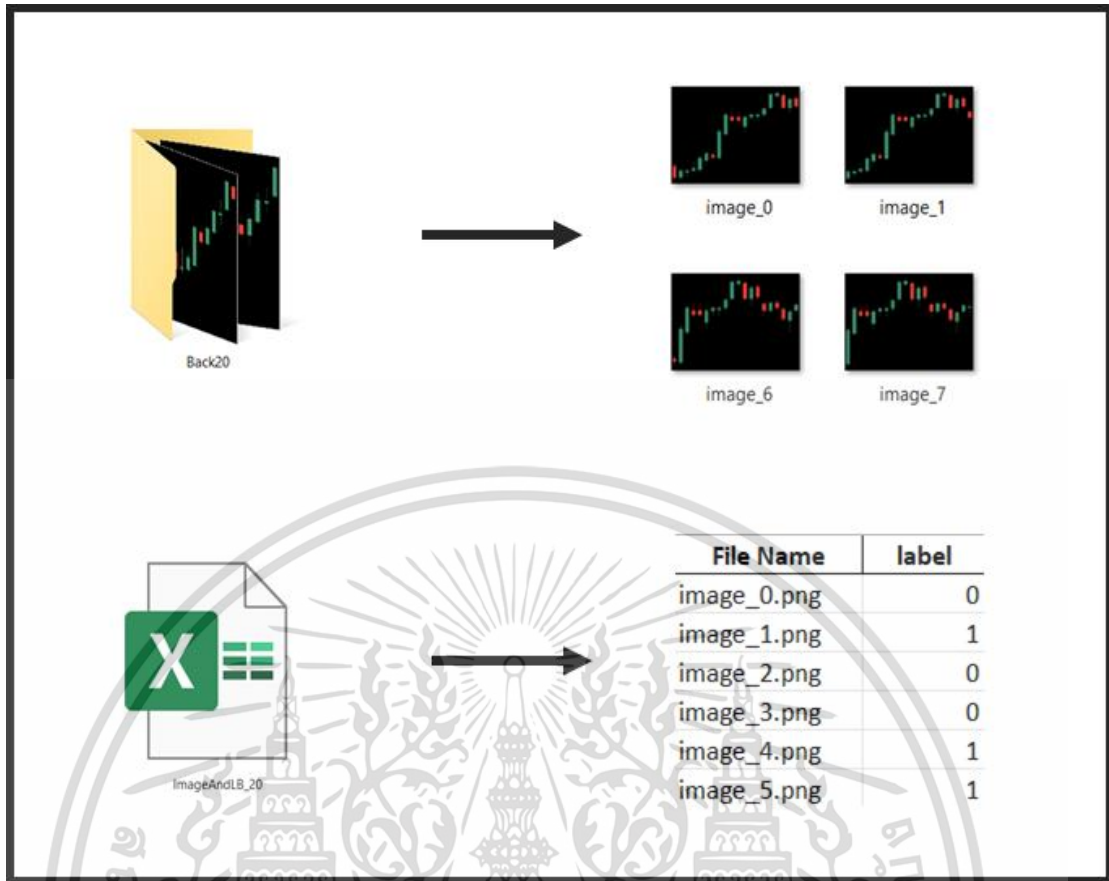
เมื่อทำการสร้างรูปภาพกราฟแท่งเทียน และแปลงค่าราคาข้าวโพดให้อยู่ในรูปแบบของทิศทางแล้ว ผู้วิจัยได้ทำการจัดเก็บรูปภาพกราฟแท่งเทียนในโฟลเดอร์ และทำการจัดเก็บชื่อของภาพและทิศทางราคาข้าวโพดในไฟล์เอกซ์เซล โดยการจัดเก็บจะแยกตามกรอบเวลาซึ่งก็คือ 5 และ 20 วัน ดังแสดงในรูปที่ 3.5 และรูปที่ 3.6 ตามลำดับ จากนั้นนำข้อมูลทั้งหมดจัดเก็บไว้ใน Google Drive ซึ่งกรณีนี้ผู้วิจัยสามารถนำเข้าข้อมูลไปใช้กับ Keras โดยใช้ไลบรารี Flow from Dataframe เพื่อโหลดข้อมูลและส่งต่อไปยังโมเดลได้อย่างสะดวกและรวดเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 ตัวอย่างการจัดเก็บข้อมูล (กรอบเวลา 5 วัน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



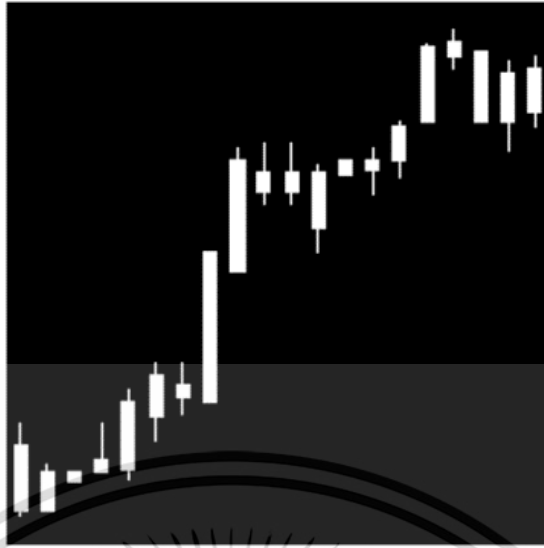
รูปที่ 3.6 ตัวอย่างการจัดเก็บข้อมูล (กรอบเวลา 20 วัน)

### 3.3.5. การปรับค่าความเข้มของสี

โดยปกติรูปภาพจะมีค่าความเข้มของสีในแต่ละพิกเซลอยู่ในช่วงระหว่าง 0 ถึง 255 โดยการทำให้ปรับช่วงขนาดของข้อมูลรูปภาพจะแปลงค่าความเข้มของสีในภาพให้อยู่ในช่วง 0-1 การปรับความเข้มสีในแต่ละพิกเซลของข้อมูลช่วยให้โมเดลเรียนรู้ และส่งผลลัพธ์ที่ดีขึ้น (Kwok et al, 2011)

ในงานวิจัยนี้ได้มีการปรับขนาดข้อมูลให้อยู่ในช่วง 0-1 ซึ่งดังรูปที่ 3.7 และ รูปที่ 3.8 จะแสดงรูปภาพที่ไม่ได้ทำการปรับขนาดข้อมูล และรูปภาพที่ทำการปรับขนาดข้อมูล ตามลำดับ ซึ่งกรณีภาพที่ไม่ได้ทำการปรับขนาดข้อมูลรูปภาพจะมีลักษณะเป็นสีขาวดำ ส่วนภาพที่ทำการปรับขนาดของข้อมูลจะมีภาพที่มีสี RGB ตามรูปภาพต้นฉบับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 รูปภาพที่ไม่ได้มีการปรับค่าความเข้มของสีให้อยู่ในช่วง [0-1]



รูปที่ 3.8 รูปภาพที่มีการปรับค่าความเข้มของสีให้อยู่ในช่วง [0-1]

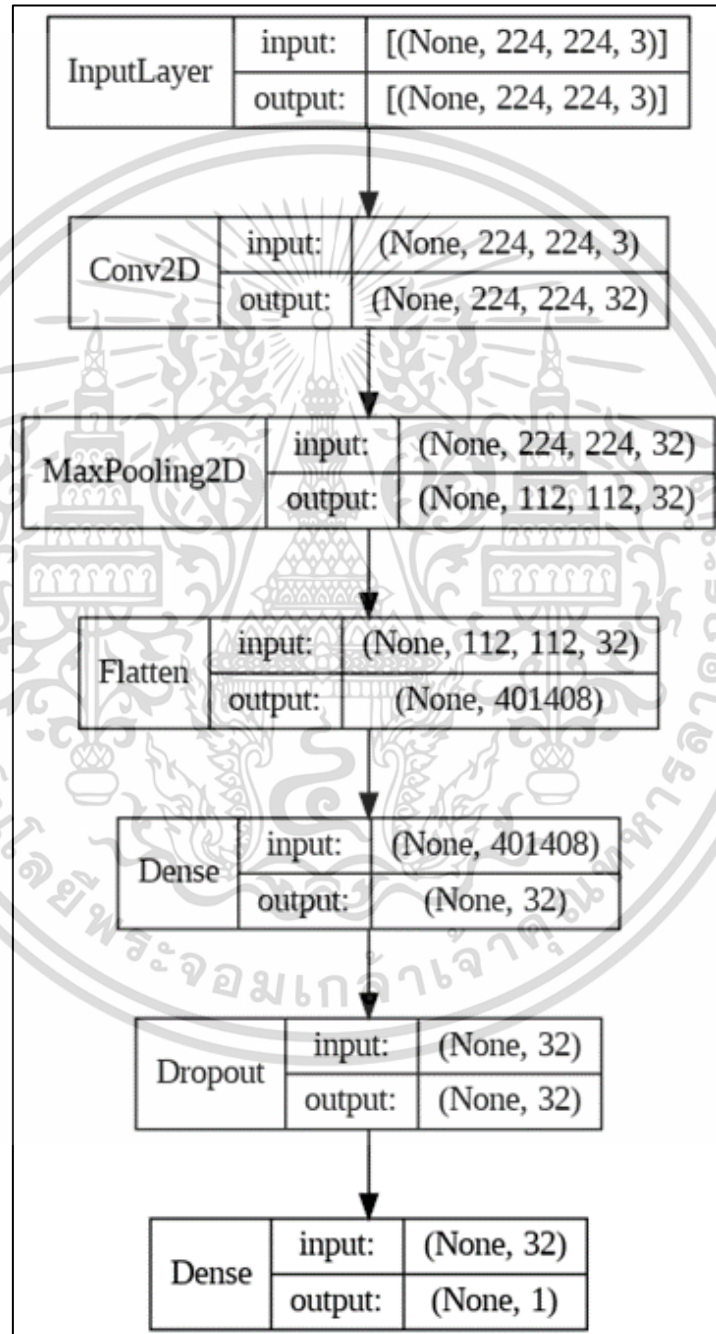
### 3.4. การสร้างโมเดลในการทำนายทิศทางราคาข้าวโพด

ในงานวิจัยนี้สร้างตัวโมเดลการเรียนรู้เชิงลึกในการทำนายทิศทางราคาข้าวโพดกับรูปภาพกราฟแท่งเทียนโดยใช้กรอบเวลา 5 และ 20 วัน จำนวน 6018 และ 6003 รูปตามลำดับ ซึ่งแต่ละโมเดลจะแบ่งเป็นชุดฝึกสอน (Training Set) ชุดตรวจสอบ (Validation Set) และชุดทดสอบ (Test Set) โดยอัตราส่วน 70:10:20 เปอร์เซ็นต์ (Mustafa et al., 2019) โดยโมเดลที่นำมาทำนายทิศทางราคาข้าวโพดมีดังนี้

เอกสารนี้เป็นเอกสารที่... สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.1. โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Network)

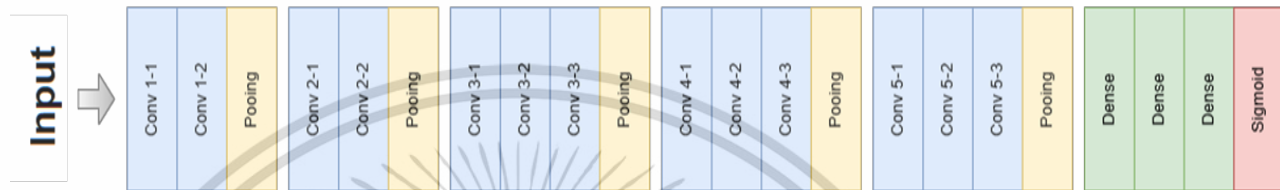
ผู้วิจัยได้มีการศึกษาอัลกอริทึมโครงข่ายประสาทเทียมแบบคอนโวลูชัน เพื่อนำมาออกแบบโครงสร้างของโมเดลในการทำนายทิศทางราคาข้าวโพด โดยรูปที่ 3.9 แสดงรูปภาพของโครงข่ายประสาทเทียมแบบคอนโวลูชันที่ได้มีการออกแบบในงานวิจัยนี้



รูปที่ 3.9 สถาปัตยกรรมโครงข่ายประสาทเทียมแบบคอนโวลูชันที่ออกแบบโดยผู้วิจัย เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.2. Visual Geometry Group (VGG-16)

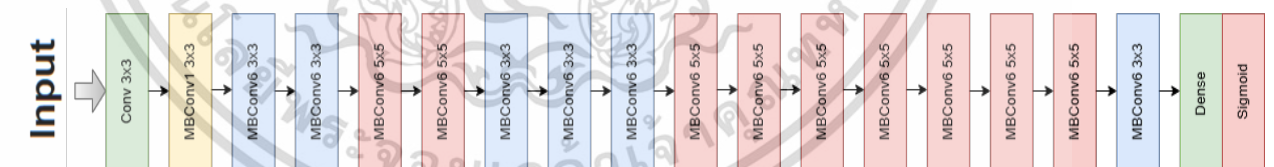
ผู้วิจัยได้มีการเรียกใช้งานโมเดล VGG-16 โดยใช้ไลบรารี Keras ในการนำเข้าโมเดล ซึ่งได้มีการปรับขนาดรูปภาพเป็น 224x224 พิกเซล เพื่อให้สามารถนำเข้าโมเดลได้ โดยโมเดล VGG-16 จะมีชั้นคอนโวลูชัน 13 ชั้น ชั้นเชิงโยงสมบурณ์ 3 ชั้น และผู้วิจัยได้มีการปรับแต่งชั้นส่งออกและฟังก์ชันกระตุ้นให้เหมาะสมกับในงานวิจัยนี้ โดยจะแสดงในรูปแบบที่ 3.10



รูปที่ 3.10 สถาปัตยกรรม VGG-16 ที่ใช้ในงานวิจัย

### 3.4.3. EfficientNet-B0

ผู้วิจัยได้มีการเรียกใช้งานโมเดล EfficientNet B0 โดยใช้ไลบรารี Keras ในการนำเข้าโมเดล ซึ่งได้มีการปรับขนาดรูปภาพเป็น 224x224 พิกเซล เพื่อให้สามารถนำเข้าโมเดลได้ โดยโมเดล VGG-16 จะมีชั้นคอนโวลูชัน 1 ชั้น MBConv 16 ชั้น และชั้นเชิงโยงสมบурณ์ 1 ชั้น และผู้วิจัยได้มีการปรับแต่งชั้นส่งออกและฟังก์ชันกระตุ้นให้เหมาะสมกับในงานวิจัยนี้ โดยจะแสดงในรูปแบบที่ 3.11



รูปที่ 3.11 สถาปัตยกรรม EfficientNet-B0 ที่ใช้ในงานวิจัย

### 3.4.4 การกำหนดค่าไฮเปอร์พารามิเตอร์

งานวิจัยนี้มีวัตถุประสงค์ศึกษาค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุดของโมเดล โดยจะมีการกำหนดขนาดชุดข้อมูล (Batch Size) อัตราการเรียนรู้ (Learning Rate) ซึ่งได้มีการอ้างอิงจากงานวิจัยของ Kandel and Castelli (2020) และจำนวนรอบการฝึกฝน (Epochs) ที่ใช้ในงานวิจัยนี้

ได้มีการอ้างอิงมาจากงานวิจัยของ Ahmed and Karim (2020) โดยจะแสดงในตารางที่ 3.3

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3 ค่าไฮเปอร์พารามิเตอร์ที่กำหนด

ขนาดชุดข้อมูล (Batch Size)	16
	256
อัตราการเรียนรู้ (Learning Rate)	0.001
	0.0001
จำนวนรอบการฝึกฝน (Epochs)	20

### 3.5. เครื่องมือที่ใช้ในงานวิจัย

#### 3.5.1. โปรแกรมภาษาไพธอน (Python)

Python เป็นหนึ่งในภาษาโปรแกรมระดับสูงที่ใช้กันอย่างแพร่หลาย ถูกออกแบบเพื่อให้มีโครงสร้างและ ไวยากรณ์ของภาษาที่ไม่ซับซ้อน เข้าใจง่าย ซึ่งมีไลบรารี (Library) เปรียบเสมือนโปรแกรมสำเร็จรูปที่เก็บฟังก์ชันการทำงานที่เฉพาะทาง ประโยชน์ของไลบรารีสำเร็จรูปที่มากับโปรแกรมก็คือผู้ใช้งานไม่จำเป็นต้องสร้างไลบรารีขึ้นมาใหม่เองทั้งหมด แต่สามารถนำไลบรารีที่ได้มีการพัฒนาไว้อยู่แล้วมาใช้งานได้เลย โดยงานวิจัยนี้มีไลบรารีที่ใช้ในการวิเคราะห์ ดังตารางที่ 3.4

ตารางที่ 3.4 ไลบรารีที่ใช้ในการวิเคราะห์

Library	คำอธิบายเพิ่มเติม
Pandas	ใช้สำหรับการจัดการข้อมูลและวิเคราะห์ข้อมูล
Numpy	ใช้คำนวณทางคณิตศาสตร์ สามารถคำนวณใน Array หลายมิติ หรือ Matrix ได้
Scikit-Learn	ใช้สำหรับการสร้างโมเดลแบบการเรียนรู้ของเครื่อง (Machine Learning) เช่น โครงข่ายประสาทเทียม และ ซัพพอร์ตเวกเตอร์การถดถอย เป็นต้น
Mathplotlib	ใช้สำหรับแสดงผลข้อมูล (Data Visualization)
Plotly	ใช้สำหรับสร้างรูปภาพกราฟแท่งเทียน
Keras	ใช้สำหรับสร้างและฝึกโมเดล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.6. การเปรียบเทียบประสิทธิภาพของโมเดล

ในการเปรียบเทียบประสิทธิภาพในการทำนายทิศทางราคาข้าวโพดของโมเดล โครงข่ายประสาทเทียมแบบคอนโวลูชันที่ออกแบบโดยผู้วิจัย (Customized CNN), Visual Geometry Group (VGG-16) และ EfficientNet B0 จะใช้ค่าความแม่นยำ (Accuracy) ในการเปรียบเทียบประสิทธิภาพของโมเดลเนื่องจากงานวิจัยนี้ไม่มีปัญหาของความไม่สมดุลของคลาส (Imbalanced Class) โดยมีสูตรการคำนวณดังสมการที่ 3.2

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3.2)$$



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

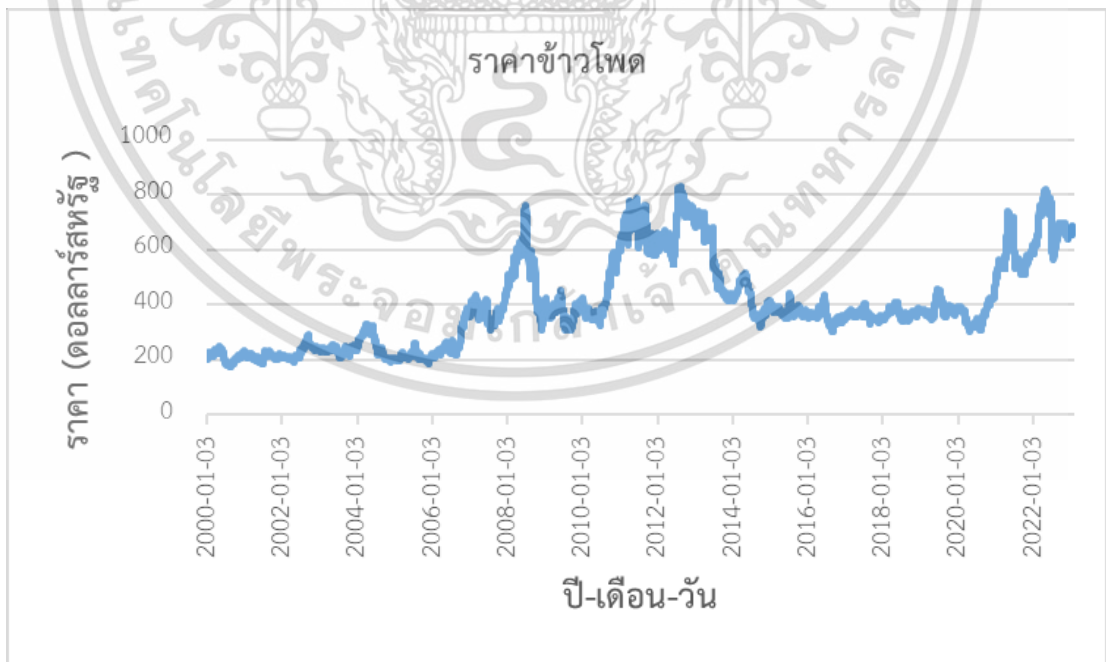
## บทที่ 4

### ผลการวิจัย

ในการศึกษาครั้งนี้มีวัตถุประสงค์เพื่อทำนายทิศทางราคาข้าวโพดมีการซื้อขายล่วงหน้าในตลาดหออการค้าแห่งนครชิคาโก (CBOT) โดยใช้รูปภาพกราฟแท่งเทียน และได้นำความรู้โครงข่ายประสาทเทียมแบบคอนโวลูชันเพื่อนำมาสร้างโมเดล โดยโมเดลที่ใช้ในงานวิจัยนี้คือ โมเดลโครงข่ายประสาทเทียมแบบคอนโวลูชันที่มีการออกแบบโดยผู้วิจัย (Customized CNN) และโมเดลที่เป็นสถาปัตยกรรมของโครงข่ายประสาทเทียมแบบคอนโวลูชันได้แก่ VGG-16 และ EfficientNet-B0 เพื่อนำมาเปรียบเทียบค่าความแม่นยำ และศึกษาไฮเปอร์พารามิเตอร์ที่มีผลกับโมเดลในการทำนายทิศทางราคาข้าวโพด

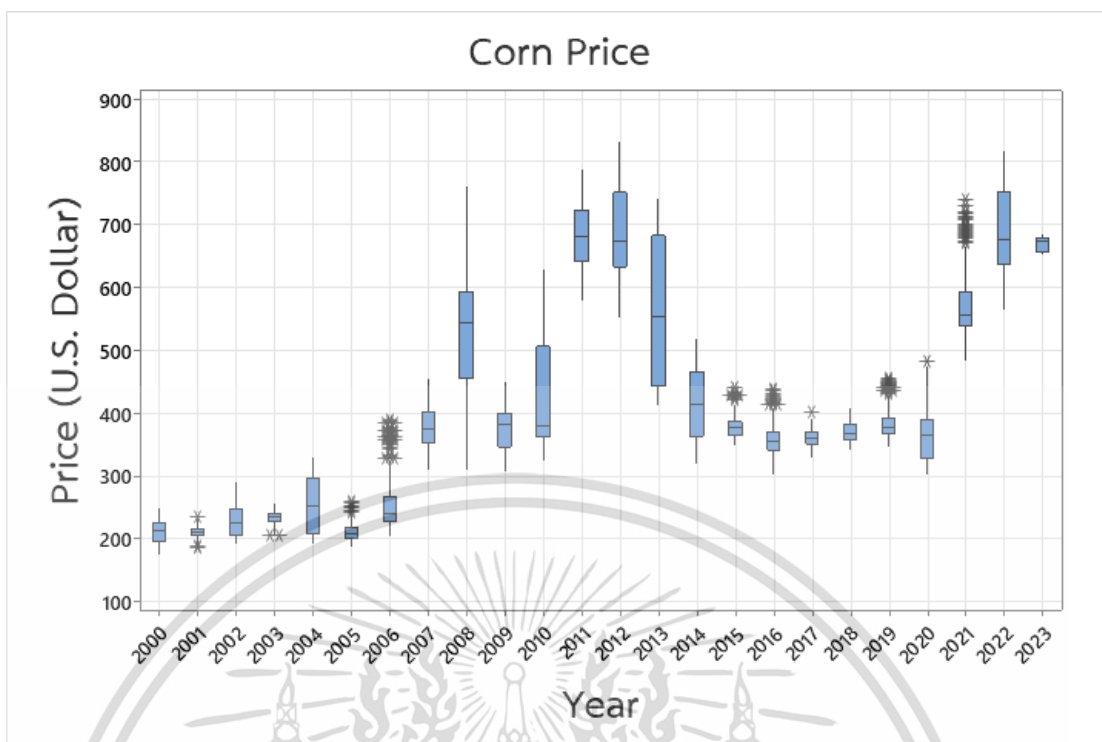
#### 4.1. การวิเคราะห์ข้อมูลเบื้องต้น

งานวิจัยครั้งนี้ได้มีการนำข้อมูลราคาข้าวโพดที่เก็บรวบรวมจากตลาดหออการค้าแห่งนครชิคาโก (CBOT) ตั้งแต่วันที่ 3/1/2000 ถึงวันที่ 1/2/2023 โดยจะแสดงการเคลื่อนไหวของราคาข้าวโพดที่มีแนวโน้มที่เพิ่มขึ้น ดังแสดงในรูปที่ 4.1



รูปที่ 4.1 การเคลื่อนไหวของราคาข้าวโพด

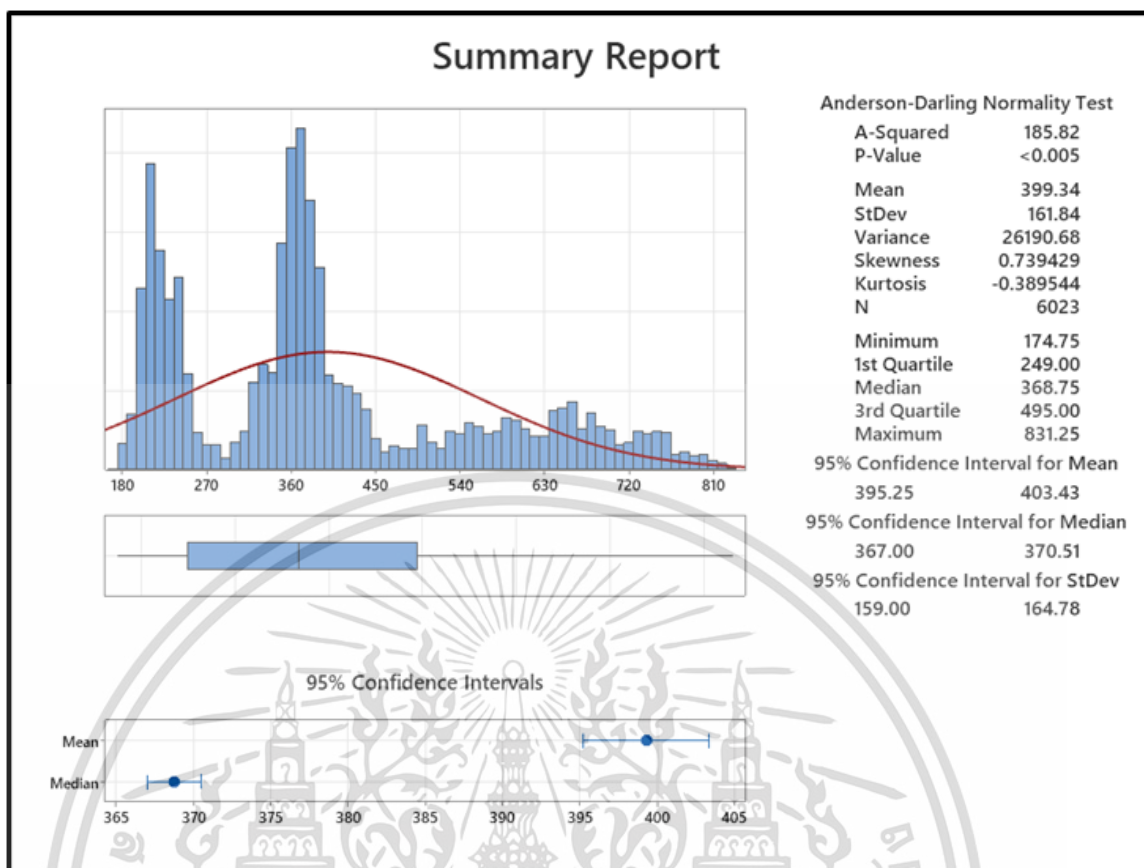
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 แผนภาพกล่อง (Box-Plot) ของราคาข้าวโพด

จากการวิเคราะห์รูปที่ 4.1 และรูปที่ 4.2 จะเห็นว่าราคาข้าวโพดตั้งแต่ปี 2000-2006 จะมีราคาไม่เกิน 400 ดอลลาร์สหรัฐ และปรับตัวสูงขึ้นในปี 2007-2008 โดยในช่วงปี 2010 แผนภาพแสดงให้เห็นว่าข้อมูลมีลักษณะที่เบ้ขวา ซึ่งหมายถึงราคาข้าวโพดในปีนั้นจะราคาโดยส่วนมากจะต่ำกว่า 381 ดอลลาร์สหรัฐ โดยมีการปรับตัวของราคาเพิ่มขึ้นเรื่อย ๆ จนถึงปี 2012 ซึ่งมีราคาในตลาดสูงถึง 831.25 ดอลลาร์สหรัฐ จากนั้นราคาค่อย ๆ ลดลงโดยสาเหตุเนื่องมาจากปัจจัยข่าวและปัจจัยทางด้านสภาพอากาศทำให้ราคาข้าวโพดลดลงมาจนถึงปี 2015 และมีการปรับฐานราคาอยู่ในช่วง 378.13 – 484 ดอลลาร์สหรัฐ และเนื่องด้วยสถานการณ์การแพร่ระบาดของเชื้อโคโรนาไวรัส (COVID-19) ค่อย ๆ ฟื้นตัวจึงทำให้ในช่วงปลายปี 2020 ราคาข้าวโพดมีการปรับตัวสูงขึ้นจนมีราคาสูงถึง 818.25 ดอลลาร์สหรัฐ ในปี 2022 และเหมือนจะมีการปรับฐานราคามาอยู่ในช่วง 671.22 – 685.25 ในต้นปี 2023

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

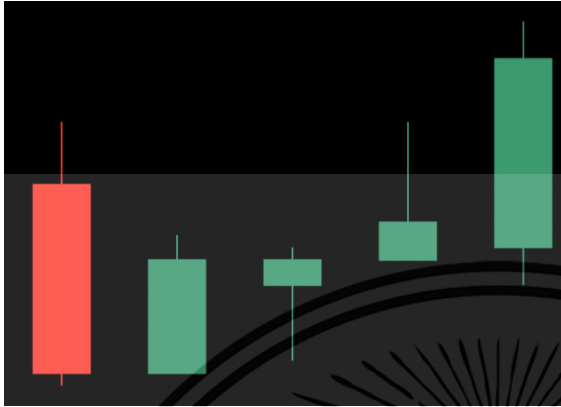


รูปที่ 4.3 การวิเคราะห์สถิติเบื้องต้นเกี่ยวกับราคาข้าวโพด

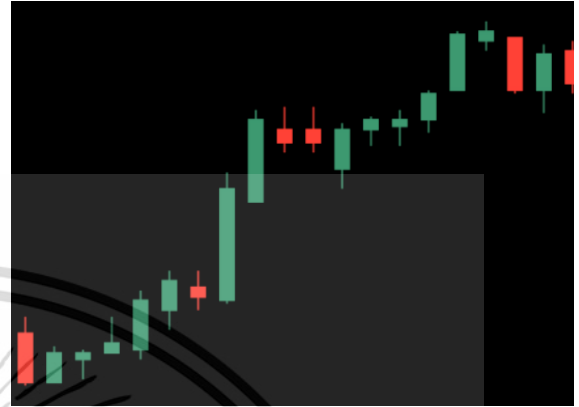
จากรูปที่ 4.3 แสดงให้เห็นว่าราคาของข้าวโพดในตลาดหอการค้าแห่งนครชิคาโก (CBOT) ตั้งแต่ปี 2000 – 2023 มีจำนวน 6,023 วัน โดยข้อมูลไม่ได้มีการแจกแจงแบบปกติซึ่งมีการแจกแจงแบบ 2 ยอด (Bimodal Distribution) โดยบ่งบอกว่าฐานของราคามีการปรับตัวจากราคาที่ต่ำไปสู่ราคาที่สูงขึ้น ซึ่งให้ค่าความเบ้ (Skewness) เป็นบวกบ่งบอกถึงกราฟที่มีลักษณะเบ้ขวา และมีค่าความโด่ง (Kurtosis) ที่น้อยกว่า 0 ซึ่งแสดงถึงกราฟมีความโด่งที่ต่ำ ข้อมูลดังกล่าวแสดงให้เห็นว่าราคาข้าวโพดโดยส่วนมากมีราคาต่ำและข้อมูลมีการกระจายมาก โดยราคาข้าวโพดมีราคาต่ำสุดที่ 174.75 ดอลลาร์สหรัฐ และราคาสูงสุดถึง 831.25 ดอลลาร์สหรัฐ ซึ่งมีค่าเฉลี่ยอยู่ที่ 399.34 ดอลลาร์สหรัฐ โดยที่ความมั่นใจระดับ 95% ว่าค่าเฉลี่ยของราคาจะอยู่ในช่วงค่าระหว่าง 395.25 ถึง 403.43 ดอลลาร์สหรัฐ

โดยงานวิจัยนี้ใช้รูปภาพเป็นข้อมูลนำเข้า 2 รูปแบบได้แก่ รูปที่เป็นกรอบเวลา 5 และ 20 วัน โดยจะแสดงในรูปที่ 4.4 และรูปที่ 4.5 ซึ่งขนาดของรูปภาพต้นฉบับมีความละเอียดอยู่ที่ 700x500 พิกเซล (Pixel) โดยรูปที่ถูกบันทึกจะมีชนิดของไฟล์ภาพเป็น PNG และถูกปรับขนาดโดยฟังก์ชันเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

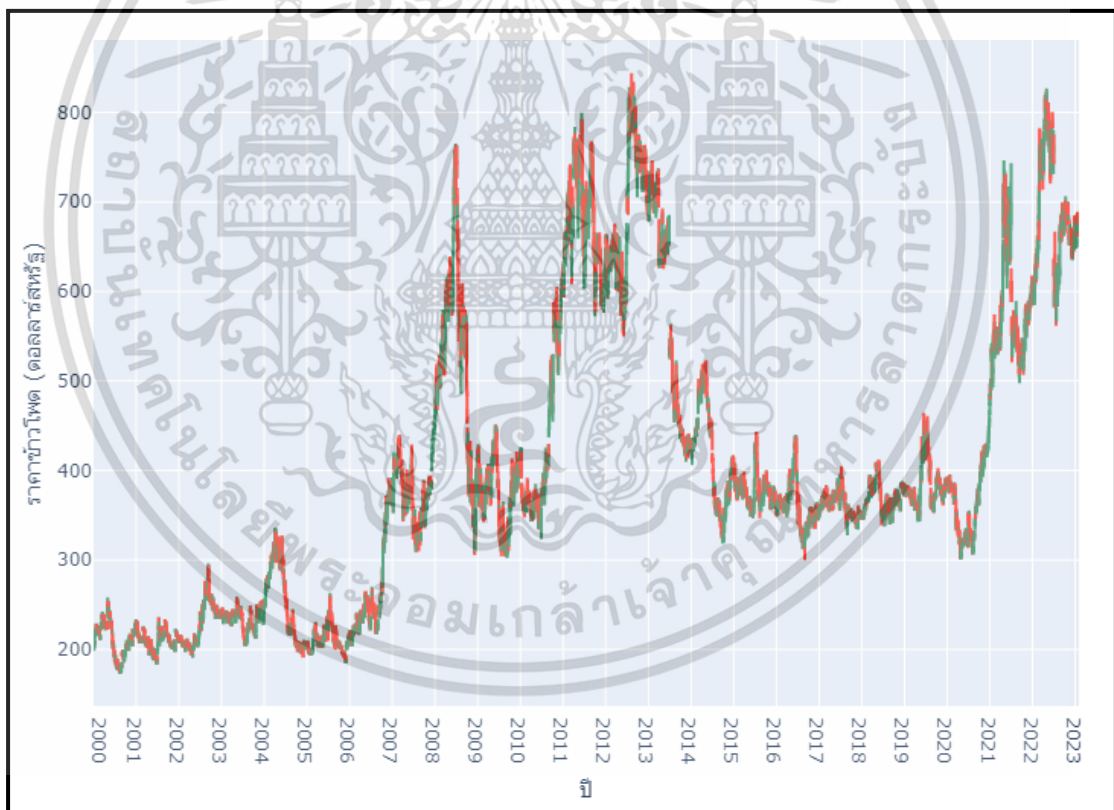
Flow\_From\_Dataframe ในภาษา Python เป็น 224x224 พิกเซล (Pixel) เพื่อให้สามารถนำเข้าโมเดลได้



รูปที่ 4.4 กราฟแท่งเทียนที่กรอบเวลา 5 วัน



รูปที่ 4.5 กราฟแท่งเทียนที่กรอบเวลา 20 วัน



รูปที่ 4.6 กราฟแท่งเทียนแสดงการเคลื่อนไหวของราคาข้าวโพด

โดยรูปที่ 4.6 เป็นกราฟแท่งเทียนแสดงการเคลื่อนไหวของราคาข้าวโพดในตลาดหอการค้าแห่งนครซิกาโก (CBOT) ตั้งแต่วันที่ 3/1/2000 ถึงวันที่ 1/2/2023 ดังรูปจะเห็นว่ากราฟแท่งมีเอกลักษณ์เป็นสีแดงกับสีเขียว ซึ่งแท่งสีแดงหมายถึงราคาเปิดมากกว่าราคาปิด และแท่งสีเขียวหมายถึงไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

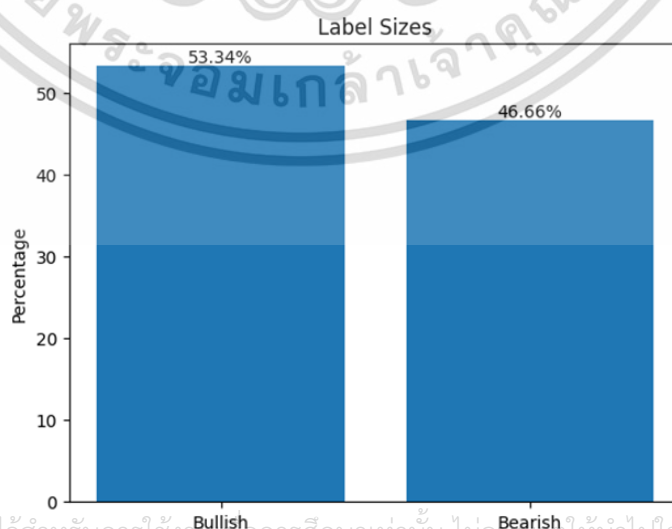
ราคาปิดมากกว่าราคาเปิด โดยจะเห็นว่ากราฟมีลักษณะบางช่วงที่ขาดไปเนื่องจากตลาดจะไม่ได้เปิดให้ทำการซื้อขายสัญญาในวันเสาร์-อาทิตย์ ซึ่งลักษณะกราฟเช่นนี้เกิดขึ้นเมื่อตลาดมีการปิดทำการ และราคาตอนเปิดตลาดมีราคาที่ต่ำกว่าเดิมมาก

ผู้วิจัยได้ทำการแบ่งข้อมูลออกเป็น 3 ชุด ได้แก่ ชุดฝึกฝน (Training Set) ชุดตรวจสอบ (Validation Set) และชุดทดสอบ (Test Set) โดยอัตราส่วน 70:10:20 เปอร์เซนต์ โดยจะแสดงสัดส่วนของผลเฉลี่ยในกรอบเวลา 5 วัน และสัดส่วนของผลเฉลี่ยในแต่ละชุดข้อมูลทั้ง 3 ชุด ในตารางที่ 4.1 จากนั้นจะแสดงสัดส่วนของผลเฉลี่ยในกรอบเวลา 20 วัน และสัดส่วนของผลเฉลี่ยในแต่ละชุดข้อมูลทั้ง 3 ชุด ในตารางที่ 4.2

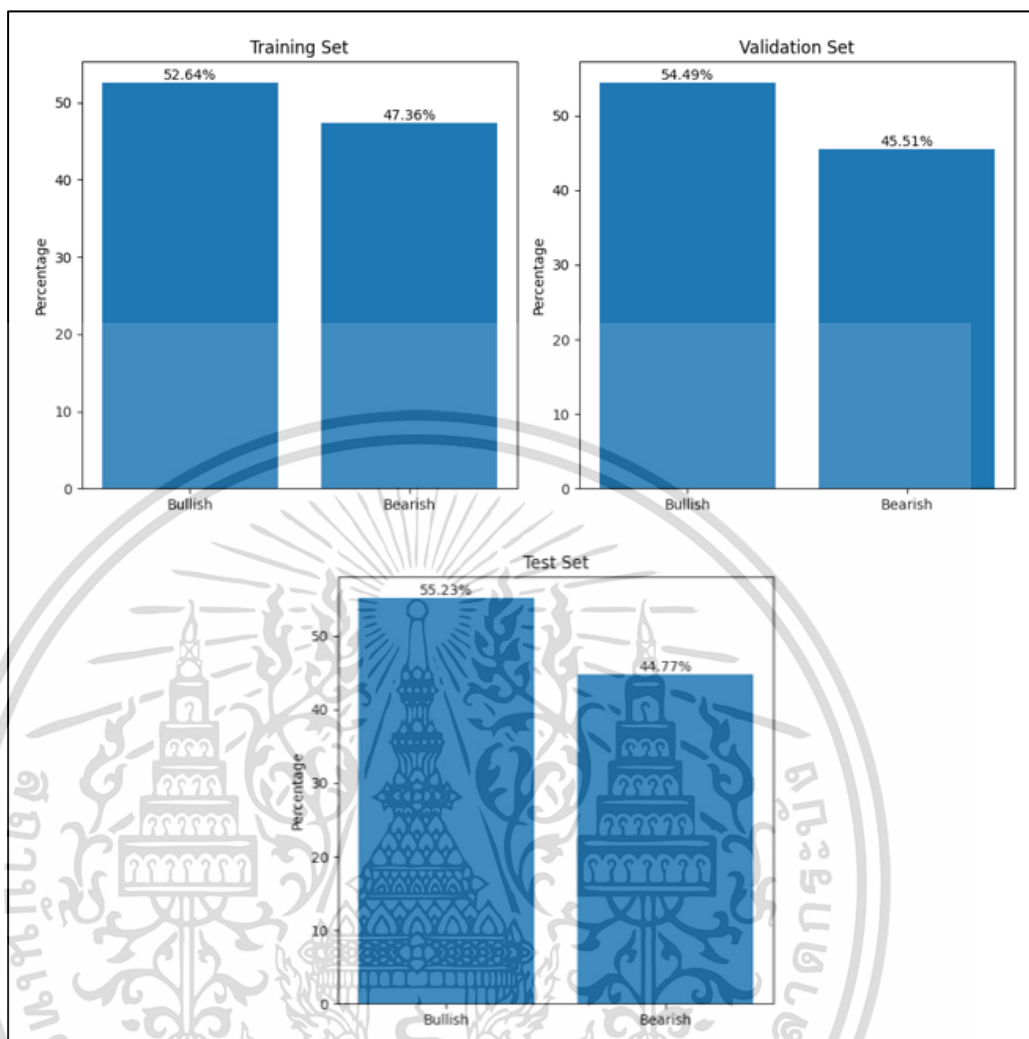
ตารางที่ 4.1 สัดส่วนของผลเฉลี่ย ณ กรอบเวลา 5 วัน ในแต่ละชุดข้อมูล

กรอบเวลา 5 วัน	ชุดฝึกฝน (Training Set)	ชุดตรวจสอบ (Validation Set)	ชุดทดสอบ (Test Set)	รวม
ตลาดกระทิงหรือตลาดขาขึ้น (Bullish)	2,217	328	665	3,210
ตลาดหมีหรือตลาดขาลง (Bearish)	1,995	274	539	2,808
รวม	4,212	602	1,204	6,018

โดยจะแสดง อัตราส่วน ณ กรอบเวลา 5 วัน ของตลาดขาขึ้น และตลาดขาลง ดังรูปที่ 4.7 และแสดงอัตราส่วนในแต่ละชุดข้อมูล ดังรูปที่ 4.8



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามรูปที่ 4.7 กราฟแท่งแสดงผลเฉลี่ยในกรอบเวลา 5 วัน ทุกครั้งที่มีการนำไปใช้



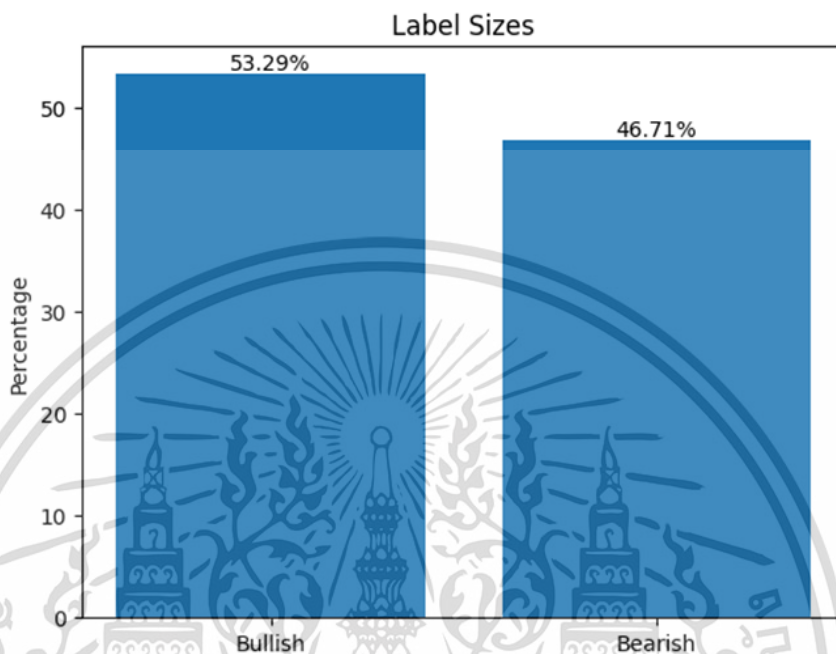
รูปที่ 4.8 กราฟแท่งแสดงผลเฉลี่ยในแต่ละชุดข้อมูลของกรอบเวลา 5 วัน

ตารางที่ 4.2 สัดส่วนของผลเฉลี่ย ณ กรอบเวลา 20 วัน ในแต่ละชุดข้อมูล

กรอบเวลา 20 วัน	ชุดฝึกฝน (Training Set)	ชุดตรวจสอบ (Validation Set)	ชุดทดสอบ (Test Set)	รวม
ตลาดกระทิงหรือตลาดขาขึ้น (Bullish)	2,210	326	663	3,199
ตลาดหมีหรือตลาดขาลง (Bearish)	1,991	275	538	2,804
รวม	4,201	601	1,201	6,003

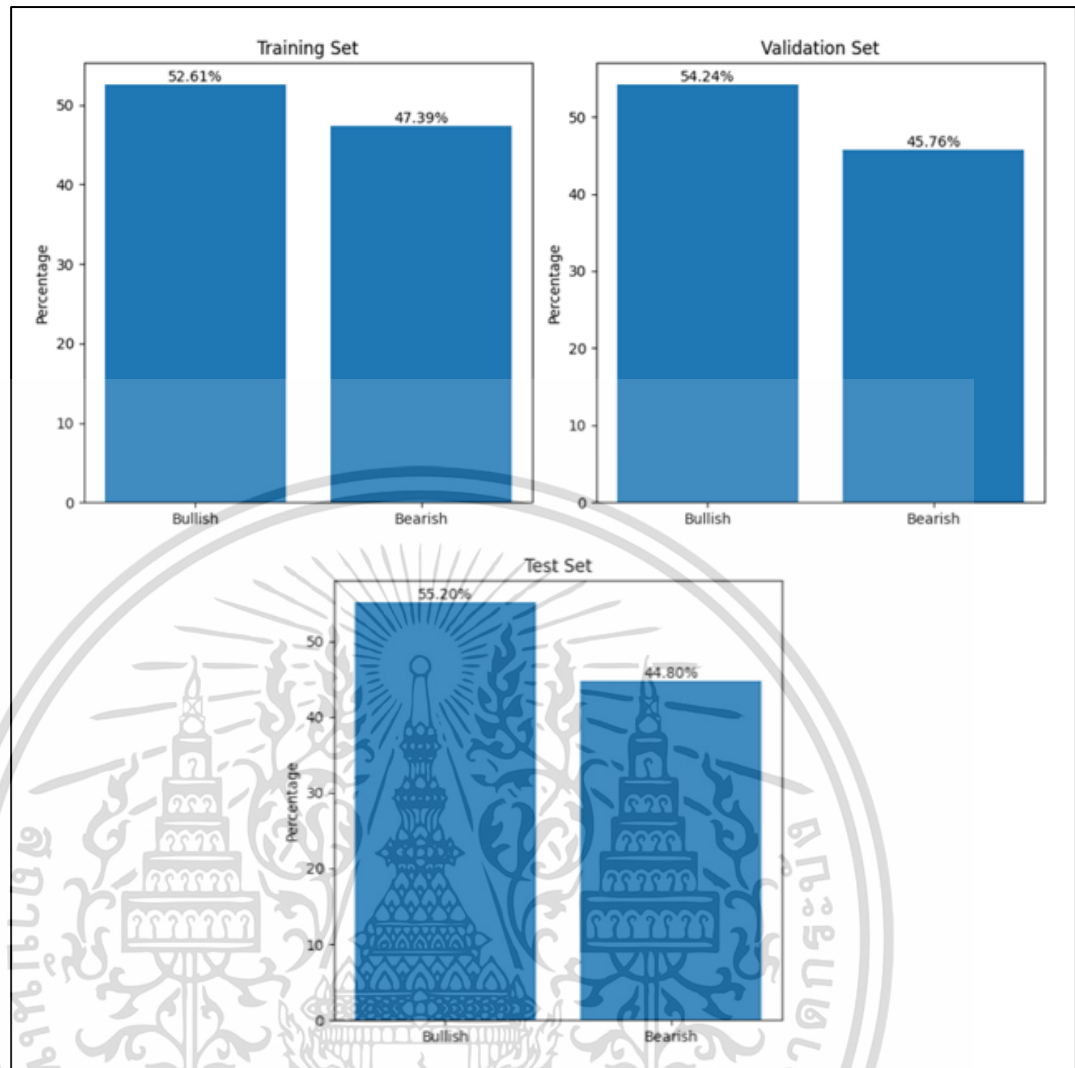
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยจะแสดง อัตราส่วน ณ กรอบเวลา 20 วัน ของตลาดขาขึ้น และตลาดขาลง ดังรูปที่ 4.9 และแสดงอัตราส่วนในแต่ละชุดข้อมูล ดังรูปที่ 4.10



รูปที่ 4.9 กราฟแท่งแสดงผลเฉลี่ยในกรอบเวลา 20 วัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 กราฟแท่งแสดงผลเฉลี่ยในแต่ละชุดข้อมูลของกรอบเวลา 20 วัน

โดยจากอัตราส่วนในรูปที่ 4.7 ถึง รูปที่ 4.10 แสดงให้เห็นว่าข้อมูลของผลเฉลี่ยที่ใช้ในงานวิจัยนี้ไม่มีปัญหาความไม่สมดุลของคลาส (Imbalanced Class)

#### 4.2. ผลการทดสอบเพื่อหาค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุดของโมเดล

งานวิจัยนี้ได้มีวัตถุประสงค์เพื่อหาค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุดสำหรับโมเดลต่าง ๆ โดยผู้วิจัยได้ใช้ชุดข้อมูลตรวจสอบ (Validation Set) ในการคัดเลือกไฮเปอร์พารามิเตอร์ก่อนที่จะนำไปใช้กับชุดข้อมูลทดสอบ (Test Set) โดยจะพิจารณาจากค่าการสูญเสียที่ต่ำที่สุด และค่าความแม่นยำที่สูงที่สุดในการหาค่าไฮเปอร์พารามิเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.1 ผลการทดสอบเพื่อหาค่าไฮเปอร์พารามิเตอร์ของโมเดล Customized CNN

ตารางที่ 4.3 ค่าความแม่นยำและค่าของโมเดล Customized CNN บนชุดข้อมูลตรวจสอบ

โมเดล	กรอบเวลา	ขนาดชุดข้อมูล (Batch Size)	อัตราการเรียนรู้ (Learning Rate)	ค่าการสูญเสีย (Validation Loss)	ค่าความแม่นยำ (Validation Accuracy)
Customized CNN	5	16	0.001	68.93%	54.82%
			0.0001	73.36%	54.32%
		256	0.001	69.69%	53.72%
			0.0001	73.27%	53.16%
	20	16	0.001	72.10%	53.74%
			0.0001	72.86%	53.41%
		256	0.001	73.80%	53.24%
			0.0001	87.12%	51.41%

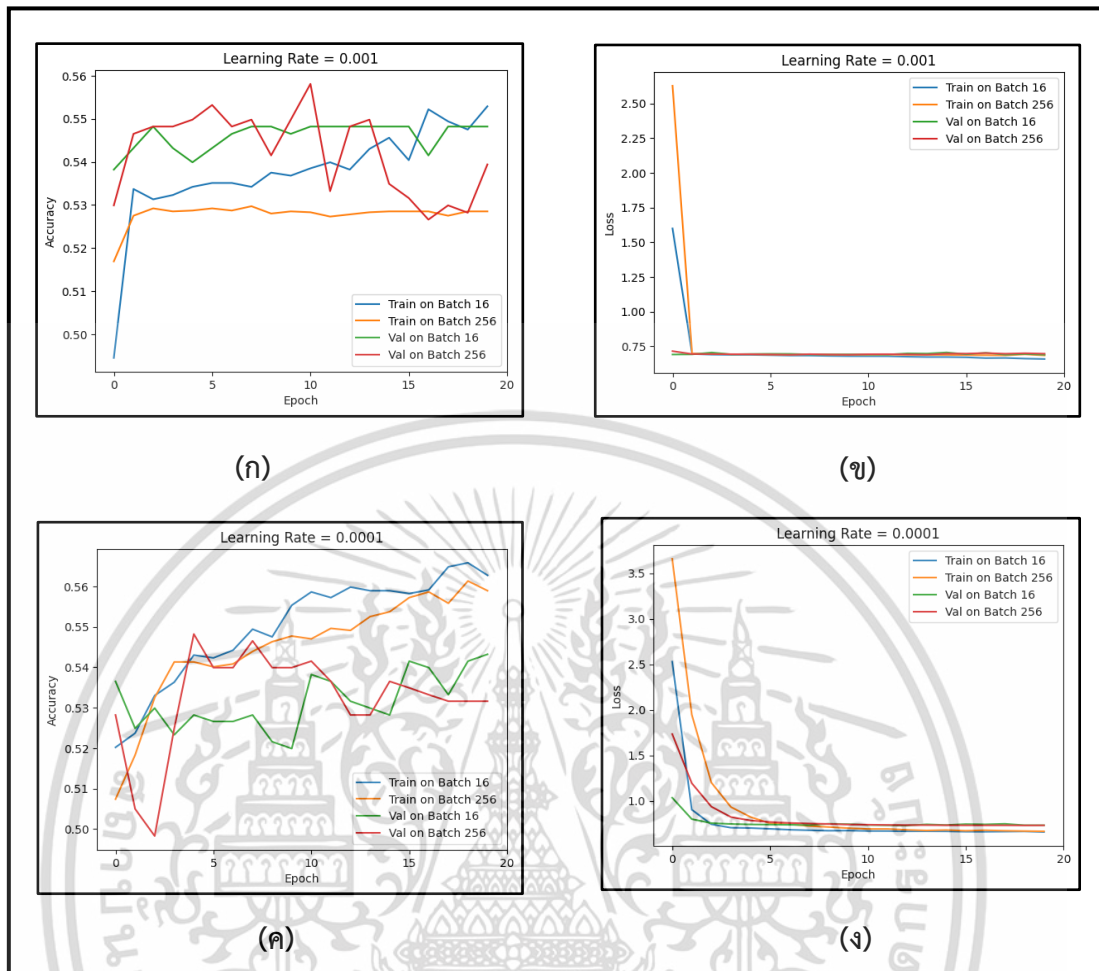
จากตารางที่ 4.3 ณ กรอบเวลา 5 วัน จะเห็นว่า Batch Size ขนาด 16 และ 256 ให้ค่าการสูญเสียที่ต่ำ และค่าความแม่นยำที่สูงในอัตราการเรียนรู้ (Learning Rate) ที่ 0.001 โดยจากค่าที่ดีที่สุดของแต่ละ Batch Size จะเห็นว่า Batch Size ขนาด 16 ให้ค่าการสูญเสีย และให้ค่าความแม่นยำที่ดีที่สุด

ณ กรอบเวลา 20 วัน จะเห็นว่า Batch Size ขนาด 16 และ 256 ให้ค่าการสูญเสียที่ต่ำ และค่าความแม่นยำที่สูงในอัตราการเรียนรู้ (Learning Rate) ที่ 0.001 โดยจากค่าที่ดีที่สุดของแต่ละ Batch Size จะเห็นว่า Batch Size ขนาด 16 ให้ค่าการสูญเสีย และให้ค่าความแม่นยำที่ดีที่สุด

ดังนั้นค่าไฮเปอร์พารามิเตอร์ที่จะนำไปใช้สำหรับชุดข้อมูลทดสอบ (Test Set) สำหรับโมเดล Customized CNN

- ณ กรอบเวลา 5 วัน คือ Batch Size ขนาด 16 และ อัตราการเรียนรู้ (Learning Rate) ที่ 0.001
- ณ กรอบเวลา 20 วัน คือ Batch Size ขนาด 16 และ อัตราการเรียนรู้ (Learning Rate) ที่ 0.001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

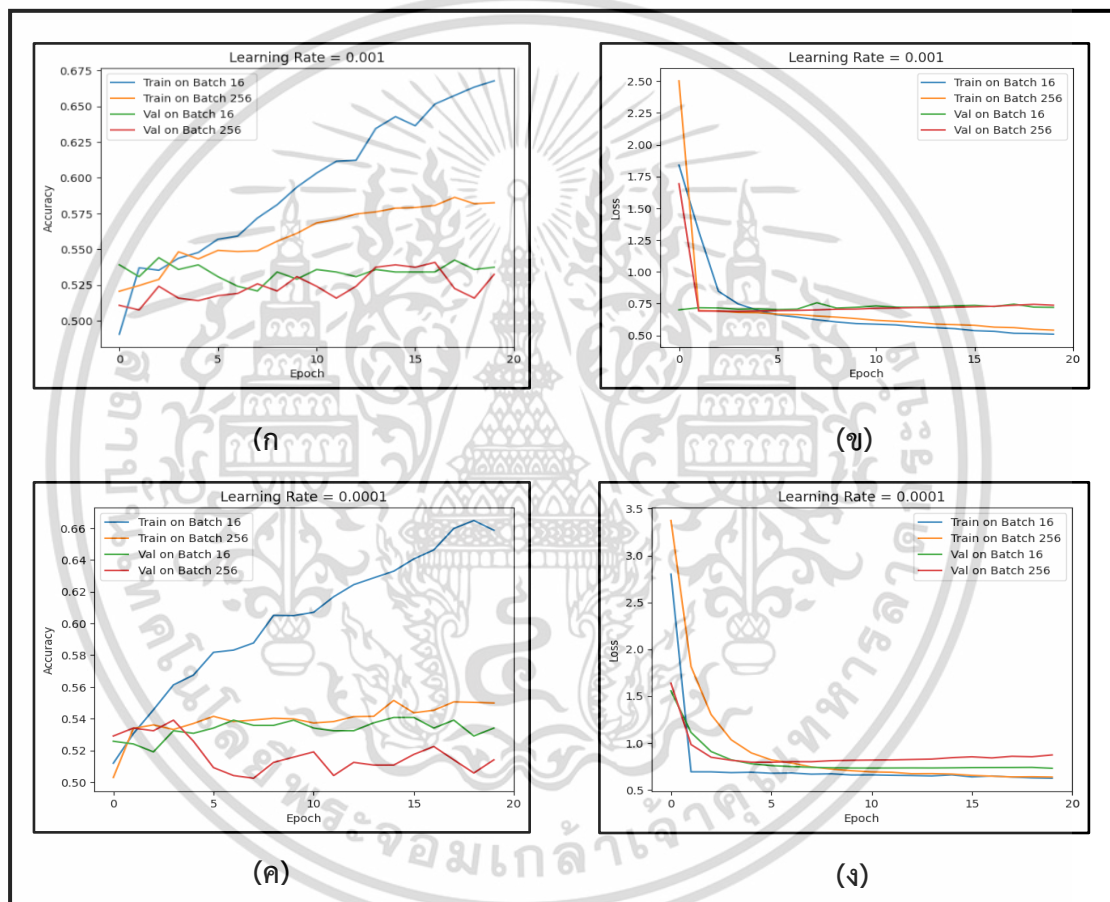


รูปที่ 4.11 กราฟเปรียบเทียบค่าการสูญเสีย และค่าความแม่นยำของโมเดล Customized CNN ณ ระยะเวลา 5 วัน

จากรูปที่ 4.11 (ก) และ รูปที่ 4.11 (ข) จะแสดงการเปรียบเทียบค่า Batch Size ขนาด 16 และ 256 ของชุดข้อมูลฝึกฝน (Training Set) และชุดข้อมูลตรวจสอบ (Validation Set) ที่อัตราการเรียนรู้ (Learning Rate) 0.001 โดยรูปที่ 4.11 (ก) จะแสดงการเปรียบเทียบค่าความแม่นยำซึ่งจะเห็นว่าค่าความแม่นยำในชุดข้อมูลฝึกฝน (Training Set) ของ Batch Size ขนาด 16 ให้ค่าที่สูงกว่า Batch Size ขนาด 256 และค่าความแม่นยำในชุดข้อมูลตรวจสอบ (Validation Set) ของ Batch Size ขนาด 256 จะให้ค่าความแม่นยำที่สูงในช่วงแรกและลดลงมามีค่าต่ำกว่า Batch Size ขนาด 16 โดยที่ Batch Size ขนาด 16 จะค่อนข้างให้ค่าความแม่นยำที่คงที่ และรูปที่ 4.11 (ข) จะแสดงการเปรียบเทียบค่าการสูญเสีย ซึ่งค่าการสูญเสียของแต่ละ Batch Size จะลู่เข้าไปในทิศทางเดียวกัน

จากรูปที่ 4.11 (ค) และ รูปที่ 4.11 (ง) จะแสดงการเปรียบเทียบค่า Batch Size ขนาด 16 และ 256 ของชุดข้อมูลฝึกฝน (Training Set) และชุดข้อมูลตรวจสอบ (Validation Set) ที่อัตราการ

เรียนรู้ (Learning Rate) 0.0001 โดยรูปที่ 4.11 (ค) จะแสดงการเปรียบเทียบค่าความแม่นยำซึ่งจะเห็นว่าค่าความแม่นยำในชุดข้อมูลฝึกฝน (Training Set) ของ Batch Size ขนาด 16 ให้ค่าที่สูงกว่า Batch Size ขนาด 256 และค่าความแม่นยำในชุดข้อมูลตรวจสอบ (Validation Set) ของ Batch Size ขนาด 256 จะให้ค่าความแม่นยำที่สูงในช่วงแรกและลดลงมาต่ำกว่า Batch Size ขนาด 16 โดยที่ Batch Size ขนาด 16 มีค่าความแม่นยำที่ต่ำกว่าในช่วงแรกและเพิ่มขึ้นในช่วงรอบการเรียนรู้ที่ 15 และรูปที่ 4.11 (ง) จะแสดงการเปรียบเทียบค่าการสูญเสีย ซึ่งค่าการสูญเสียของแต่ละ Batch Size จะลู่เข้าไปในทิศทางเดียวกัน



รูปที่ 4.12 กราฟเปรียบเทียบค่าการสูญเสีย และค่าความแม่นยำของโมเดล Customized CNN ผนวกกรอบเวลา 20 วัน

จากรูปที่ 4.12 (ก) และ รูปที่ 4.12 (ข) จะแสดงการเปรียบเทียบค่า Batch Size ขนาด 16 และ 256 ของชุดข้อมูลฝึกฝน (Training Set) และชุดข้อมูลตรวจสอบ (Validation Set) ที่อัตราการเรียนรู้ (Learning Rate) 0.001 โดยรูปที่ 4.12 (ก) จะแสดงการเปรียบเทียบค่าความแม่นยำซึ่งจะเห็นว่าค่าความแม่นยำในชุดข้อมูลฝึกฝน (Training Set) ของ Batch Size ขนาด 16 ให้ค่าที่สูงกว่า Batch Size ขนาด 256 และค่าความแม่นยำในชุดข้อมูลตรวจสอบ (Validation Set) ของ Batch Size ขนาด 256 จะให้ค่าความแม่นยำที่สูงในช่วงแรกและลดลงมาต่ำกว่า Batch Size ขนาด 16 โดยที่ Batch Size ขนาด 16 มีค่าความแม่นยำที่ต่ำกว่าในช่วงแรกและเพิ่มขึ้นในช่วงรอบการเรียนรู้ที่ 15 และรูปที่ 4.12 (ง) จะแสดงการเปรียบเทียบค่าการสูญเสีย ซึ่งค่าการสูญเสียของแต่ละ Batch Size จะลู่เข้าไปในทิศทางเดียวกัน

Size ขนาด 256 และค่าความแม่นยำในชุดข้อมูลตรวจสอบ (Validation Set) ของ Batch Size ขนาด 256 จะให้ค่าความแม่นยำที่ต่ำกว่า Batch Size ขนาด 16 ตั้งแต่ในช่วงรอบการเรียนรู้ที่ 5 และรูปที่ 4.12 (ข) จะแสดงการเปรียบเทียบค่าการสูญเสีย ซึ่งค่าการสูญเสียของแต่ละ Batch Size จะลู่เข้าไปในทิศทางเดียวกัน

จากรูปที่ 4.12 (ค) และ รูปที่ 4.12 (ง) จะแสดงการเปรียบเทียบค่า Batch Size ขนาด 16 และ 256 ของชุดข้อมูลฝึกฝน (Training Set) และชุดข้อมูลตรวจสอบ (Validation Set) ที่อัตราการเรียนรู้ (Learning Rate) 0.0001 โดยรูปที่ 4.12 (ค) จะแสดงการเปรียบเทียบค่าความแม่นยำซึ่งจะเห็นว่าค่าความแม่นยำในชุดข้อมูลฝึกฝน (Training Set) ของ Batch Size ขนาด 16 ให้ค่าที่สูงกว่า Batch Size ขนาด 256 และค่าความแม่นยำในชุดข้อมูลตรวจสอบ (Validation Set) ของ Batch Size ขนาด 256 จะให้ค่าความแม่นยำที่สูงในช่วงแรกและลดลงมาต่ำกว่า Batch Size ขนาด 16 โดยที่ Batch Size ขนาด 16 มีค่าความแม่นยำที่ต่ำกว่าในช่วงแรกและเพิ่มขึ้นในช่วงรอบการเรียนรู้ที่ 15 และรูปที่ 4.12 (ง) จะแสดงการเปรียบเทียบค่าการสูญเสีย ซึ่งค่าการสูญเสียของแต่ละ Batch Size ของชุดข้อมูลฝึกฝน (Training Set) จะลู่เข้าไปในทิศทางเดียวกัน และในชุดข้อมูลตรวจสอบ (Validation Set) ขนาด Batch Size 16 จะให้ค่าการสูญเสียที่ต่ำกว่า

#### 4.2.2 ผลการทดสอบเพื่อหาค่าไฮเปอร์พารามิเตอร์ของโมเดล VGG-16

ตารางที่ 4.4 ค่าความแม่นยำและค่าการสูญเสียของโมเดล VGG-16 บนชุดข้อมูลตรวจสอบ

โมเดล	กรอบเวลา	ขนาดชุดข้อมูล (Batch Size)	อัตราการเรียนรู้ (Learning Rate)	ค่าการสูญเสีย (Validation Loss)	ค่าความแม่นยำ (Validation Accuracy)
VGG-16	5	16	0.001	69.41%*	53.65%*
			0.0001	69.86%	53.32%
		256	0.001	69.60%	52.99%
			0.0001	70.68%	53.49%
	20	16	0.001	69.66%	53.74%
			0.0001	69.13%*	54.58%*
		256	0.001	70.23%	52.58%
			0.0001	70.18%	52.91%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

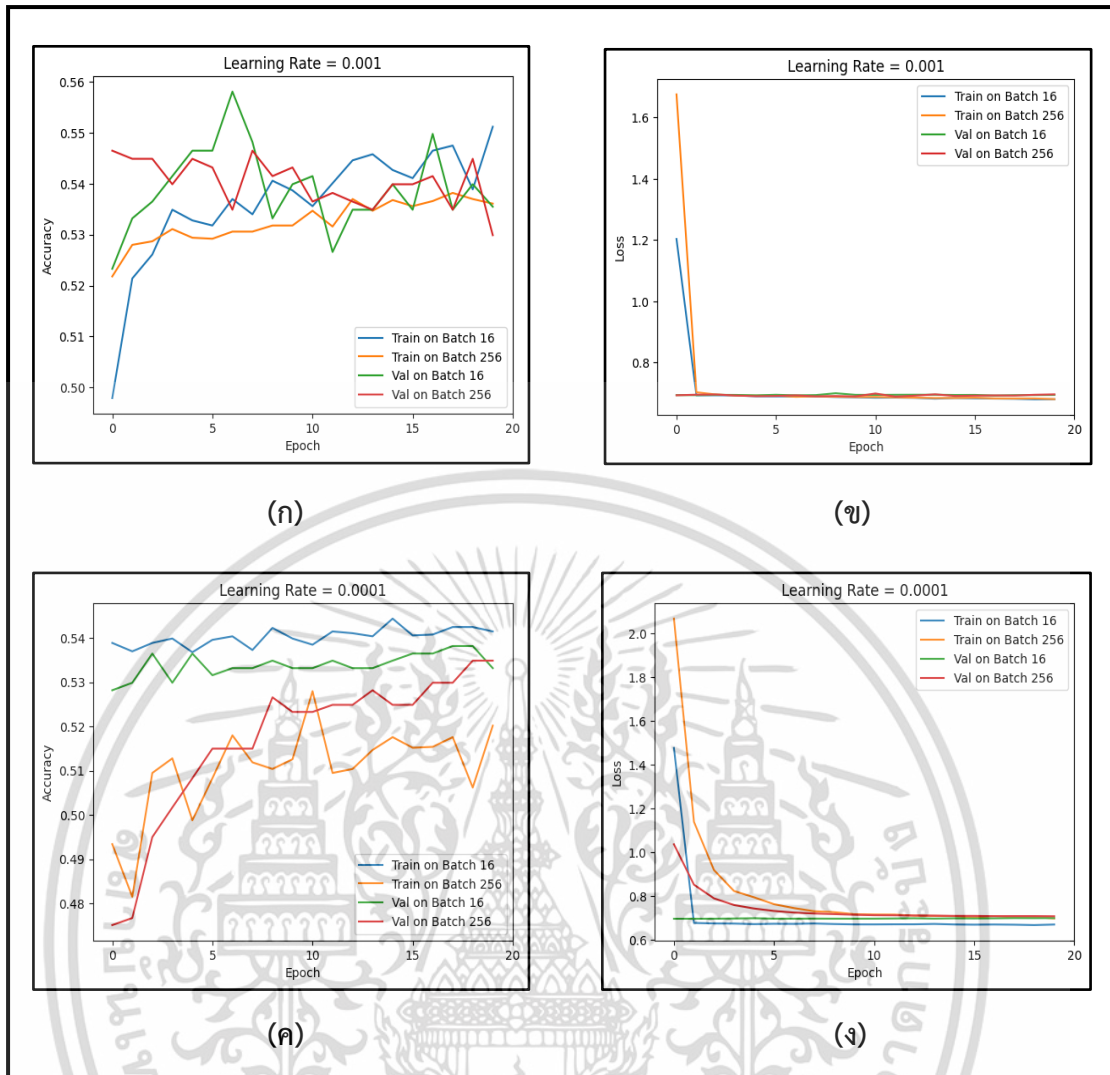
จากตารางที่ 4.4 ณ ระยะเวลา 5 วัน จะเห็นว่า Batch Size ขนาด 16 ให้ค่าการสูญเสียที่ต่ำ และค่าความแม่นยำในอัตราการเรียนรู้ (Learning Rate) ที่ 0.001 ซึ่ง Batch Size ขนาด 256 ให้ค่าการสูญเสียที่ต่ำที่อัตราการเรียนรู้ 0.001 แต่ให้ค่าความแม่นยำในอัตราการเรียนรู้ 0.0001 โดยจากค่าที่ดีที่สุดของแต่ละ Batch Size จะเห็นว่า Batch Size ขนาด 16 ให้ค่าการสูญเสีย และให้ค่าความแม่นยำที่ดีที่สุดในชุดข้อมูลตรวจสอบ

ณ ระยะเวลา 20 วัน จะเห็นว่า Batch Size ขนาด 16 และ 256 ให้ค่าการสูญเสียที่ต่ำ และค่าความแม่นยำในอัตราการเรียนรู้ (Learning Rate) ที่ 0.0001 โดยจากค่าที่ดีที่สุดของแต่ละ Batch Size จะเห็นว่า Batch Size ขนาด 16 ให้ค่าการสูญเสีย และให้ค่าความแม่นยำที่ดีที่สุดในชุดข้อมูลตรวจสอบ

ดังนั้นค่าไฮเปอร์พารามิเตอร์ที่จะนำไปใช้สำหรับชุดข้อมูลทดสอบ (Test Set) สำหรับโมเดล VGG-16

- ณ ระยะเวลา 5 วัน คือ Batch Size ขนาด 16 และ อัตราการเรียนรู้ (Learning Rate) ที่ 0.001
- ณ ระยะเวลา 20 วัน คือ Batch Size ขนาด 16 และ อัตราการเรียนรู้ (Learning Rate) ที่ 0.0001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

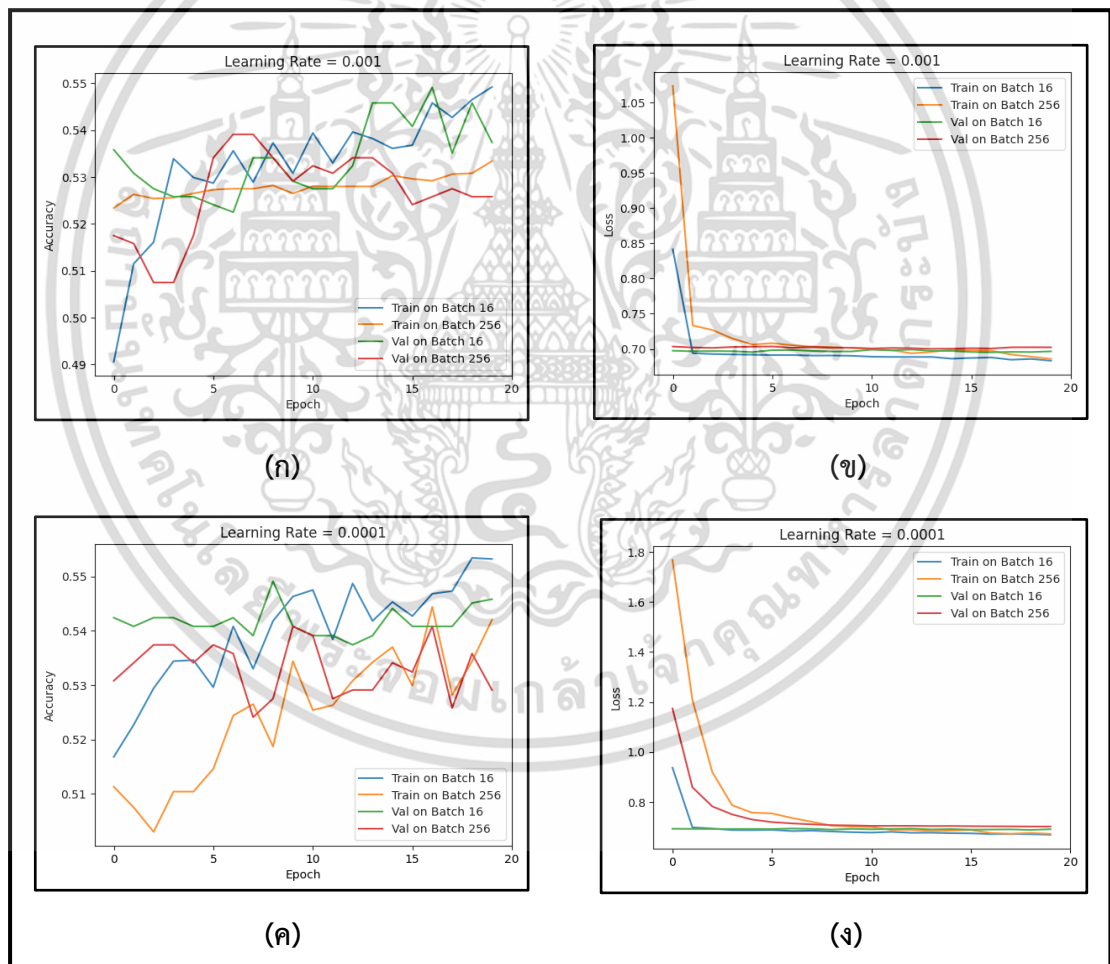


รูปที่ 4.13 กราฟเปรียบเทียบค่าการสูญเสีย และค่าความแม่นยำของโมเดล VGG-16  
ณ กรอบเวลา 5 วัน

จากรูปที่ 4.13 (ก) และ รูปที่ 4.13 (ข) จะแสดงการเปรียบเทียบค่า Batch Size ขนาด 16 และ 256 ของชุดข้อมูลฝึกฝน (Training Set) และชุดข้อมูลตรวจสอบ (Validation Set) ที่อัตราการเรียนรู้ (Learning Rate) 0.001 โดยรูปที่ 4.13 ก. จะแสดงการเปรียบเทียบค่าความแม่นยำซึ่งจะเห็นว่าค่าความแม่นยำในชุดข้อมูลฝึกฝน (Training Set) ของ Batch Size ขนาด 16 ให้ค่าที่ต่ำกว่า Batch Size ขนาด 256 จนกระทั่งในรอบการเรียนรู้ที่ 4 Batch Size ขนาด 16 จึงให้ค่าความแม่นยำที่สูงกว่า และค่าความแม่นยำในชุดข้อมูลตรวจสอบ (Validation Set) ของ Batch Size ขนาด 256 จะให้ค่าความแม่นยำที่ต่ำกว่า Batch Size ขนาด 16 และรูปที่ 4.13 (ข) จะแสดงการเปรียบเทียบค่าการสูญเสีย ซึ่งค่าการสูญเสียของแต่ละ Batch Size จะลู่เข้าไปในทิศทางเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.13 (ค) และ รูปที่ 4.13 (ง) จะแสดงการเปรียบเทียบค่า Batch Size ขนาด 16 และ 256 ของชุดข้อมูลฝึกฝน (Training Set) และชุดข้อมูลตรวจสอบ (Validation Set) ที่อัตราการเรียนรู้ (Learning Rate) 0.0001 โดยรูปที่ 4.13 (ค) จะแสดงการเปรียบเทียบค่าความแม่นยำซึ่งจะเห็นว่าค่าความแม่นยำในชุดข้อมูลฝึกฝน (Training Set) ของ Batch Size โดยที่ Batch Size ขนาด 16 ให้ค่าที่สูงกว่า Batch Size ขนาด 256 และค่าความแม่นยำในชุดข้อมูลตรวจสอบ (Validation Set) ของ Batch Size ขนาด 16 จะให้ค่าความแม่นยำที่สูงและคงที่กว่า Batch Size ขนาด 256 โดยที่ Batch Size ขนาด 256 มีค่าความแม่นยำที่ต่ำกว่าในช่วงแรกและเพิ่มขึ้นจนมีค่าที่สูงกว่า Batch Size ขนาด 16 ในช่วงรอบการเรียนรู้ที่ 20 และรูปที่ 4.13 (ง) จะแสดงการเปรียบเทียบค่าการสูญเสีย ซึ่งค่าการสูญเสียของแต่ละ Batch Size จะลู่เข้าไปในทิศทางเดียวกัน



รูปที่ 4.14 กราฟเปรียบเทียบค่าการสูญเสีย และค่าความแม่นยำของโมเดล VGG-16

ณ กรอบเวลา 20 วัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.14 (ก) และ รูปที่ 4.14 (ข) จะแสดงการเปรียบเทียบค่า Batch Size ขนาด 16 และ 256 ของชุดข้อมูลฝึกฝน (Training Set) และชุดข้อมูลตรวจสอบ (Validation Set) ที่อัตราการเรียนรู้ (Learning Rate) 0.001 โดยรูปที่ 4.14 (ก) จะแสดงการเปรียบเทียบค่าความแม่นยำซึ่งจะเห็นว่าค่าความแม่นยำในชุดข้อมูลฝึกฝน (Training Set) ของ Batch Size โดยที่ Batch Size ขนาด 16 ให้ค่าที่สูงกว่า Batch Size ขนาด 256 ตั้งแต่รอบการเรียนรู้ที่ 3 และค่าความแม่นยำในชุดข้อมูลตรวจสอบ (Validation Set) ของ Batch Size ขนาด 16 จะให้ค่าความแม่นยำที่สูงกว่า Batch Size ขนาด 256 และรูปที่ 4.14 (ข) จะแสดงการเปรียบเทียบค่าการสูญเสีย ซึ่งค่าการสูญเสียของแต่ละ Batch Size จะลู่เข้าไปในทิศทางเดียวกัน

จากรูปที่ 4.14 (ค) และ รูปที่ 4.14 (ง) จะแสดงการเปรียบเทียบค่า Batch Size ขนาด 16 และ 256 ของชุดข้อมูลฝึกฝน (Training Set) และชุดข้อมูลตรวจสอบ (Validation Set) ที่อัตราการเรียนรู้ (Learning Rate) 0.0001 โดยรูปที่ 4.14 (ค) จะแสดงการเปรียบเทียบค่าความแม่นยำซึ่งจะเห็นว่าค่าความแม่นยำในชุดข้อมูลฝึกฝน (Training Set) และค่าความแม่นยำในชุดข้อมูลตรวจสอบ (Validation Set) ของ Batch Size โดย Batch Size ขนาด 16 จะให้ค่าความแม่นยำที่สูงกว่า Batch Size ขนาด 256 ตั้งแต่รอบการเรียนรู้ที่ 1 และรูปที่ 4.14 (ง) จะแสดงการเปรียบเทียบค่าการสูญเสีย ซึ่งค่าการสูญเสียของแต่ละ Batch Size จะลู่เข้าไปในทิศทางเดียวกัน

### 4.2.3 ผลการทดสอบเพื่อหาค่าไฮเปอร์พารามิเตอร์ของโมเดล EfficientNet-B0

ตารางที่ 4.5 ค่าความแม่นยำและค่าการสูญเสียของโมเดล EfficientNet-B0 บนชุดข้อมูลตรวจสอบ

โมเดล	กรอบเวลา	ขนาดชุดข้อมูลตอนฝึกฝน (Batch Size)	อัตราการเรียนรู้ (Learning Rate)	ค่าการสูญเสีย (Validation Loss)	ค่าความแม่นยำ (Validation Accuracy)
EfficientNet-B0	5	16	0.001	73.06%*	54.49%*
			0.0001	81.40%	50.17%
		256	0.001	83.24%	45.68%
			0.0001	85.15%	43.69%
	20	16	0.001	77.84%	50.58%
			0.0001	74.35%*	53.24%*
		256	0.001	86.29%	49.08%
			0.0001	82.56%	51.41%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ประโยชน์ใด ๆ แก่บุคคลอื่นโดยไม่ได้รับอนุญาตจากผู้จัดทำเอกสารนี้

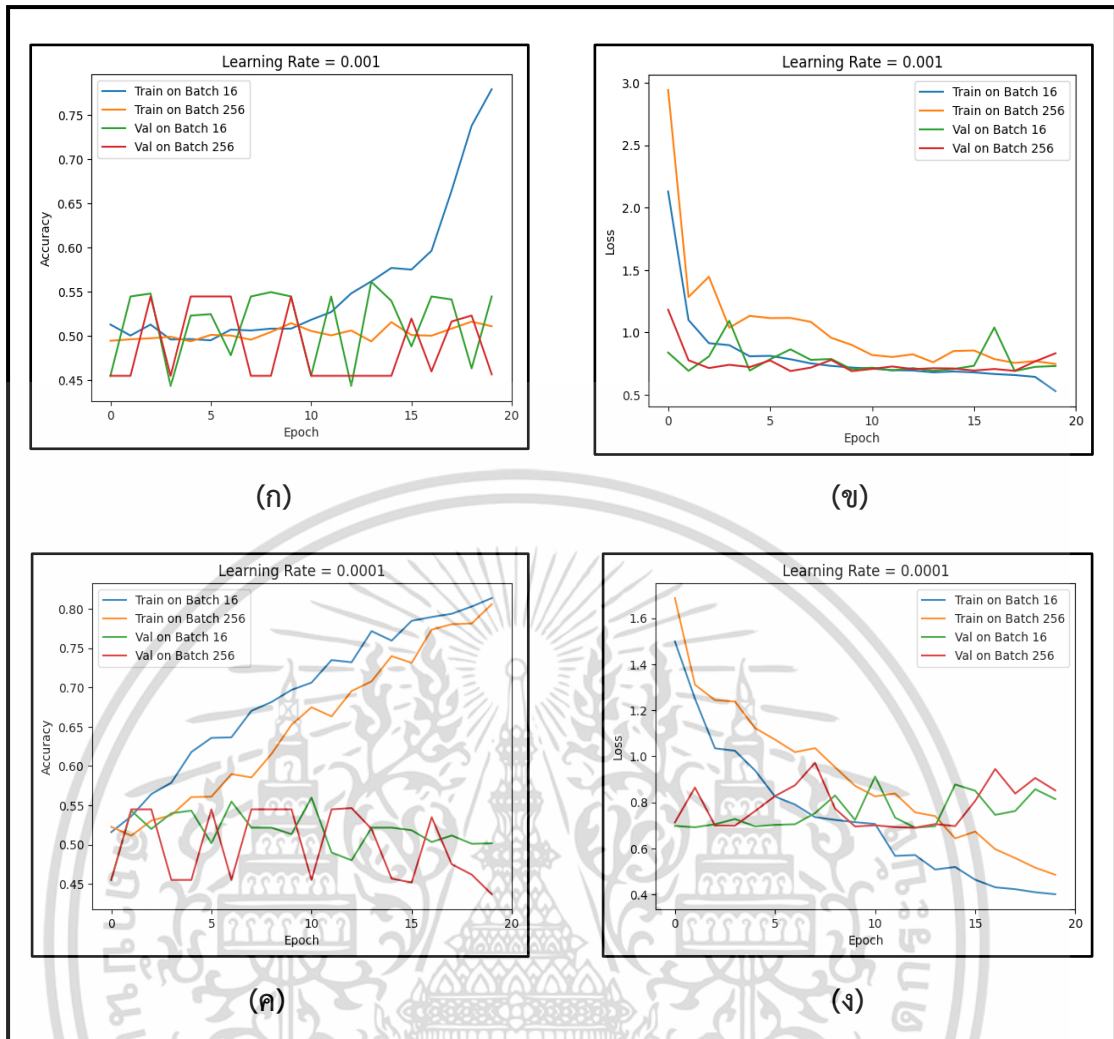
จากตารางที่ 4.5 ณ ระยะเวลา 5 วัน จะเห็นว่า Batch Size ขนาด 16 และ 256 ให้ค่าการสูญเสียที่ต่ำ และค่าความแม่นยำที่สูงในอัตราการเรียนรู้ (Learning Rate) ที่ 0.001 โดยจากค่าที่ดีที่สุดของแต่ละ Batch Size จะเห็นว่า Batch Size ขนาด 16 ให้ค่าการสูญเสีย และให้ค่าความแม่นยำที่ดีที่สุดในช่วงข้อมูลตรวจสอบ

ณ ระยะเวลา 20 วัน จะเห็นว่า Batch Size ขนาด 16 และ 256 ให้ค่าการสูญเสียที่ต่ำ และค่าความแม่นยำที่สูงในอัตราการเรียนรู้ (Learning Rate) ที่ 0.0001 โดยจากค่าที่ดีที่สุดของแต่ละ Batch Size จะเห็นว่า Batch Size ขนาด 16 ให้ค่าการสูญเสีย และให้ค่าความแม่นยำที่ดีที่สุดในช่วงข้อมูลตรวจสอบ

ดังนั้นค่าไฮเปอร์พารามิเตอร์ที่จะนำไปใช้สำหรับชุดข้อมูลทดสอบ (Test Set) สำหรับโมเดล EfficientNet-B0

- ณ ระยะเวลา 5 วัน คือ Batch Size ขนาด 16 และอัตราการเรียนรู้ (Learning Rate) ที่ 0.001
- ณ ระยะเวลา 20 วัน คือ Batch Size ขนาด 16 และอัตราการเรียนรู้ (Learning Rate) ที่ 0.0001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.15 กราฟเปรียบเทียบค่าการสูญเสีย และค่าความแม่นยำของโมเดล EfficientNet-B0 ณ ระยะเวลา 5 วัน

จากรูปที่ 4.15 (ก) และ รูปที่ 4.15 (ข) จะแสดงการเปรียบเทียบค่า Batch Size ขนาด 16 และ 256 ของชุดข้อมูลฝึกฝน (Training Set) และชุดข้อมูลตรวจสอบ (Validation Set) ที่อัตราการเรียนรู้ (Learning Rate) 0.001 โดยรูปที่ 4.15 (ก) จะแสดงการเปรียบเทียบค่าความแม่นยำซึ่งจะเห็นว่าค่าความแม่นยำในชุดข้อมูลฝึกฝน (Training Set) ของ Batch Size ขนาด 16 ให้ค่าที่สูงกว่า Batch Size ขนาด 256 และค่าความแม่นยำในชุดข้อมูลตรวจสอบ (Validation Set) ของ Batch Size ขนาด 16 และ 256 จะให้ค่าความแม่นยำที่ค่อนข้างเหยียงซึ่ง Batch Size ขนาด 16 ค่าความแม่นยำที่สูงกว่า และรูปที่ 4.15 (ข) จะแสดงการเปรียบเทียบค่าการสูญเสีย ซึ่งค่าการสูญเสียของชุดข้อมูลฝึกฝน (Training Set) ที่ Batch Size ขนาด 16 จะให้ค่าที่ต่ำกว่า Batch Size ขนาด 256 และค่าการสูญเสียของชุดข้อมูลตรวจสอบ (Validation Set) ที่ Batch Size ขนาด 16 และ Batch Size ขนาด

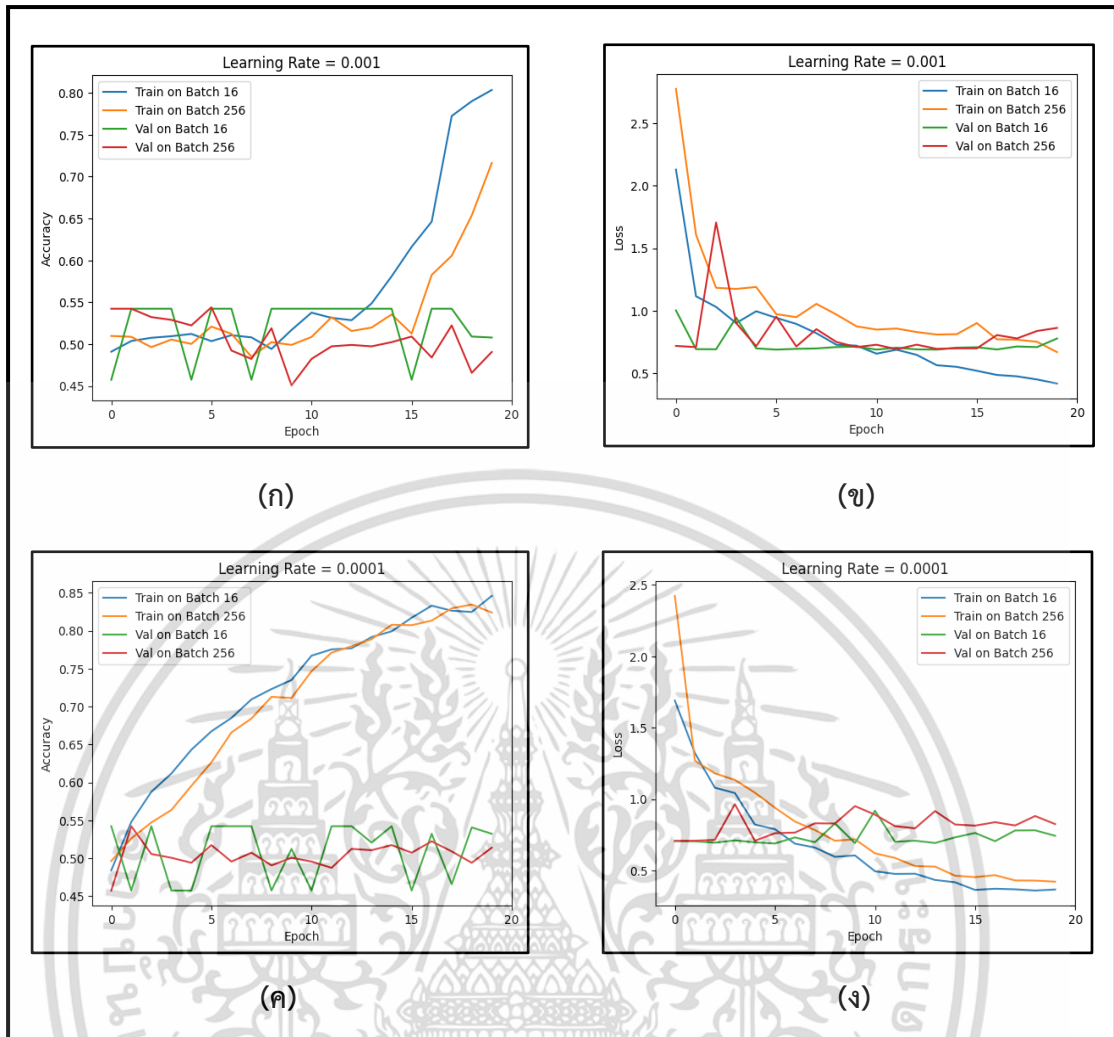
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

256 จะให้ค่าการสูญเสียที่ใกล้เคียงกัน แต่ในรอบการเรียนรู้ที่ 20 Batch Size ขนาด 16 จะให้ค่าการสูญเสียที่ต่ำกว่า

จากรูปที่ 4.15 (ค) และ รูปที่ 4.15 (ง) จะแสดงการเปรียบเทียบค่า Batch Size ขนาด 16 และ 256 ของชุดข้อมูลฝึกฝน (Training Set) และชุดข้อมูลตรวจสอบ (Validation Set) ที่อัตราการเรียนรู้ (Learning Rate) 0.0001 โดยรูปที่ 4.15 (ค) จะแสดงการเปรียบเทียบค่าความแม่นยำซึ่งจะเห็นว่าค่าความแม่นยำในชุดข้อมูลฝึกฝน (Training Set) ของ Batch Size โดยที่ Batch Size ขนาด 16 ให้ค่าที่สูงกว่า Batch Size ขนาด 256 และค่าความแม่นยำในชุดข้อมูลตรวจสอบ (Validation Set) ของ Batch Size ขนาด 16 จะให้ค่าความแม่นยำที่สูงกว่า Batch Size ขนาด 256 และรูปที่ 4.15 (ง) จะแสดงการเปรียบเทียบค่าการสูญเสีย ซึ่งค่าการสูญเสียของชุดข้อมูลฝึกฝน (Training Set) ที่ Batch Size ขนาด 16 จะให้ค่าที่ต่ำกว่า Batch Size ขนาด 256 และค่าการสูญเสียของชุดข้อมูลตรวจสอบ (Validation Set) ที่ Batch Size ขนาด 16 จะให้ค่าที่ต่ำกว่า Batch Size ขนาด 256 แต่ไม่ได้ลดลงตามชุดข้อมูลฝึกสอน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 กราฟเปรียบเทียบค่าการสูญเสีย และค่าความแม่นยำของโมเดล EfficientNet-B0 ณ กรอบเวลา 20 วัน

จากรูปที่ 4.16 (ก) และ รูปที่ 4.16 (ข) จะแสดงการเปรียบเทียบค่า Batch Size ขนาด 16 และ 256 ของชุดข้อมูลฝึกฝน (Training Set) และชุดข้อมูลตรวจสอบ (Validation Set) ที่อัตราการเรียนรู้ (Learning Rate) 0.001 โดยรูปที่ 4.16 (ก) จะแสดงการเปรียบเทียบค่าความแม่นยำซึ่งจะเห็นว่าค่าความแม่นยำในชุดข้อมูลฝึกฝน (Training Set) ของ Batch Size ขนาด 16 ให้ค่าที่สูงกว่า Batch Size ขนาด 256 และค่าความแม่นยำในชุดข้อมูลตรวจสอบ (Validation Set) ของ Batch Size ขนาด 16 จะให้ค่าความแม่นยำที่ค่อนข้างเหวี่ยงแต่ให้ค่าความแม่นยำที่สูงกว่า Batch Size ขนาด 256 และรูปที่ 4.16 (ข) จะแสดงการเปรียบเทียบค่าการสูญเสีย ซึ่งค่าการสูญเสียของชุดข้อมูลฝึกฝน (Training Set) ที่ Batch Size ขนาด 16 จะให้ค่าที่ต่ำกว่า Batch Size ขนาด 256 และค่าการสูญเสียของชุดข้อมูลตรวจสอบ (Validation Set) ที่ Batch Size ขนาด 16 และ Batch Size ขนาด 256 จะให้ค่าการ

เอกสารนี้เป็นเอกสารของงานวิจัยที่จัดทำขึ้นเพื่อใช้ในการศึกษาวิจัยเท่านั้น ไม่สามารถนำข้อมูลไปใช้  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สูญเสียที่ใกล้เคียงกัน โดยตั้งแต่รอบการเรียนรู้ที่ 15 Batch Size ขนาด 16 จะให้ค่าการสูญเสียที่ต่ำกว่า

จากรูปที่ 4.16 (ค) และ รูปที่ 4.16 (ง) จะแสดงการเปรียบเทียบค่า Batch Size ขนาด 16 และ 256 ของชุดข้อมูลฝึกฝน (Training Set) และชุดข้อมูลตรวจสอบ (Validation Set) ที่อัตราการเรียนรู้ (Learning Rate) 0.0001 โดยรูปที่ 4.16 (ค) จะแสดงการเปรียบเทียบค่าความแม่นยำซึ่งจะเห็นว่าค่าความแม่นยำในชุดข้อมูลฝึกฝน (Training Set) ของ Batch Size โดยที่ Batch Size ขนาด 16 และ Batch Size ขนาด 256 จะให้ค่าความแม่นยำที่ใกล้เคียงกัน แต่ในรอบการเรียนรู้ที่ 20 Batch Size ขนาด 16 จะให้ค่าความแม่นยำที่สูงกว่า และค่าความแม่นยำในชุดข้อมูลตรวจสอบ (Validation Set) ของ Batch Size ขนาด 16 จะให้ค่าความแม่นยำที่สูงกว่า Batch Size ขนาด 256 และรูปที่ 4.16 (ง) จะแสดงการเปรียบเทียบค่าการสูญเสีย ซึ่งค่าการสูญเสียของชุดข้อมูลฝึกฝน (Training Set) ที่ Batch Size ขนาด 16 จะให้ค่าที่ต่ำกว่า Batch Size ขนาด 256 และค่าการสูญเสียของชุดข้อมูลตรวจสอบ (Validation Set) ที่ Batch Size ขนาด 16 จะให้ค่าที่ต่ำกว่า Batch Size ขนาด 256 แต่ไม่ได้ลดลงตามชุดข้อมูลฝึกสอน

#### 4.2.4 ผลสรุปค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุดในแต่ละรอบเวลา

โดยในตารางที่ 4.6 และตารางที่ 4.7 จะแสดงค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุดของแต่ละโมเดล ณ รอบเวลา 5 และ วัน ตามลำดับ

ตารางที่ 4.6 ค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุด ณ รอบเวลา 5 วัน

รอบเวลา	โมเดล	ขนาดชุดข้อมูลตอนฝึกฝน (Batch Size)	อัตราการเรียนรู้ (Learning Rate)
5	Customized CNN	16	0.001
	VGG-16	16	0.001
	EfficientNet-B0	16	0.001

ตารางที่ 4.7 ค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุด ณ รอบเวลา 20 วัน

รอบเวลา	โมเดล	ขนาดชุดข้อมูล (Batch Size)	อัตราการเรียนรู้ (Learning Rate)
20	Customized CNN	16	0.001
	VGG-16	16	0.0001
	EfficientNet-B0	16	0.0001

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้เพื่อการพาณิชย์  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3. ผลการทดสอบโมเดลจากค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุด

งานวิจัยนี้ได้มีวัตถุประสงค์เพื่อเปรียบเทียบความแม่นยำจากค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุดสำหรับโมเดลต่าง ๆ โดยผู้วิจัยได้ใช้ค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุดของแต่ละโมเดลที่ได้จากชุดข้อมูลตรวจสอบ (Validation) โดยนำมาใช้กับชุดข้อมูลทดสอบ (Test Set) เพื่อหาโมเดลที่ดีที่สุดในแต่ละกรอบเวลา

**ตารางที่ 4.8** ค่าการสูญเสียและค่าความแม่นยำของแต่ละโมเดลบนชุดข้อมูลทดสอบ ณ กรอบเวลา 5 วัน

กรอบเวลา	โมเดล	ขนาดชุดข้อมูลตอนฝึกฝน (Batch Size)	อัตราการเรียนรู้ (Learning Rate)	ค่าการสูญเสีย (Test Loss)	ค่าความแม่นยำ (Test Accuracy)
5	Customized CNN	16	0.001	69.31%	55.39%
	VGG-16	16	0.001	69.49%	53.37%
	EfficientNet-B0	16	0.001	77.73%	49.58%

จากตารางที่ 4.8 จะเห็นว่า ณ กรอบเวลา 5 วัน โมเดลที่ให้ค่าความแม่นยำสูงสุด และค่าการสูญเสียที่ต่ำสุด คือ Customized CNN โดยกำหนดขนาดชุดข้อมูล (Batch Size) 16 และอัตราการเรียนรู้ (Learning Rate) เท่ากับ 0.001 ซึ่งให้ค่าความแม่นยำที่ 55.39% และรองลงมาคือ VGG-16 และ EfficientNet-B0 ตามลำดับ

**ตารางที่ 4.9** ค่าการสูญเสียและค่าความแม่นยำของแต่ละโมเดลบนชุดข้อมูลทดสอบ ณ กรอบเวลา 20 วัน

กรอบเวลา	โมเดล	ขนาดชุดข้อมูลตอนฝึกฝน (Batch Size)	อัตราการเรียนรู้ (Learning Rate)	ค่าการสูญเสีย (Test Loss)	ค่าความแม่นยำ (Test Accuracy)
20	EfficientNet-B0	16	0.0001	72.16%	55.03%
	Customized CNN	16	0.001	73.14%	53.37%
	VGG-16	16	0.0001	69.95%	52.78%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ ณ ระยะเวลาที่ 20 วัน โมเดลที่ให้ค่าความแม่นยำสูงสุด จากตารางที่ 4.9 คือ EfficientNet-B0 และให้ค่าการสูญเสียที่ 72.16% โดยกำหนดขนาดชุดข้อมูล (Batch Size) 16 และอัตราการเรียนรู้ (Learning Rate) เท่ากับ 0.0001 ซึ่งให้ค่าความแม่นยำที่ 55.03% และรองลงมาคือ Customized CNN และ VGG-16 ตามลำดับ

#### 4.4. การอภิปรายผล

ในงานวิจัยนี้ได้มีวัตถุประสงค์เพื่อที่จะศึกษาค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุดซึ่งมีการเปรียบเทียบ Batch Size และอัตราการเรียนรู้ (Learning Rate) ซึ่งในงานวิจัยของ ธนัช (2564) ได้มีการเปรียบเทียบค่าความแม่นยำของ Batch Size ขนาด 16 128 และ 256 ซึ่ง Batch Size ที่ให้ค่าความแม่นยำสูงสุดคือ Batch Size ขนาด 16 รองลงมาคือ Batch Size ขนาด 128 และ 256 ตามลำดับ ซึ่งสอดคล้องกับงานวิจัยนี้ และจากการศึกษางานวิจัยที่เกี่ยวข้องกับการเปรียบเทียบอัตราการเรียนรู้ (Learning Rate) พบว่างานวิจัยของงานวิจัยของ Agustika et al. (2021) ได้มีการเปรียบเทียบอัตราการเรียนรู้ 0.1 0.01 0.001 0.0001 0.00001 ซึ่งอัตราการเรียนรู้ที่ให้ค่าความแม่นยำมากที่สุดคือ 0.001 ซึ่งจะสอดคล้องกับระยะเวลา 5 วัน ของงานวิจัยนี้ และจากการศึกษางานวิจัยของ Kusuma et al. (2019) ที่ได้มีการใช้รูปภาพแห่งเทียน ณ ระยะเวลา 5 10 และ 20 วัน ซึ่งงานวิจัยดังกล่าวได้ประยุกต์ใช้อัลกอริทึมของโครงข่ายประสาทเทียมแบบคอนโวลูชัน ซึ่งให้ค่าความแม่นยำสูงสุดที่ 92.1% ซึ่งไม่สอดคล้องกับงานวิจัยนี้ได้ค่าความแม่นยำสูงสุดเพียง 55.39% โดยผู้วิจัยได้ตั้งข้อสังเกตว่าอาจเนื่องมาจากตลาดหุ้นที่นำมาวิเคราะห์นั้นแตกต่างกัน โดยในงานวิจัยนี้ได้ทำการศึกษาตลาดหุ้นที่มีการซื้อขายสัญญาล่วงหน้า แต่งานวิจัยดังกล่าวได้ศึกษาตลาดหุ้นที่ไม่ได้เป็นการซื้อขายสัญญาล่วงหน้า ซึ่งเหตุผลดังกล่าวอาจเป็นหนึ่งเหตุผลที่ทำให้ผลลัพธ์ของค่าความแม่นยำไม่สอดคล้องกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

# สรุปผลการวิจัยและข้อเสนอแนะ

### 5.1 สรุปผลการวิจัย

งานวิจัยเรื่อง “การประยุกต์ใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันกับรูปภาพกราฟแท่งเทียนเพื่อทำนายการเคลื่อนไหวของราคาข้าวโพด” มีวัตถุประสงค์เพื่อสร้างโมเดลทำนายทิศทางของราคาข้าวโพดที่มีการซื้อขายล่วงหน้าในตลาดหอกการค้าแห่งนครซิดาโกโดยพิจารณาจากกราฟแท่งเทียน และศึกษาไฮเปอร์พารามิเตอร์ที่มีผลต่อโมเดลทำนายทิศทางราคาข้าวโพดเพื่อนำค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุดของแต่ละโมเดลมาเปรียบเทียบความแม่นยำ โดยประโยชน์ที่คาดหวังของการทำงานวิจัยนี้คือสามารถนำโมเดลไปใช้ในการวางแผนลงทุนการซื้อขายข้าวโพดในตลาดหอกการค้าแห่งนครซิดาโก และสามารถนำโมเดลไปประยุกต์ใช้กับผลิตภัณฑ์ทางการเกษตรอื่น ๆ ในตลาดหอกการค้าแห่งนครซิดาโกได้ โดยผู้วิจัยได้มีการนำข้อมูลราคาข้าวโพดโดยใช้ ราคาเปิด (Open) ราคาสูงสุด (High) ราคาต่ำสุด (Low) และราคาปิด (Close) เพื่อนำมาสร้างรูปภาพกราฟแท่งเทียนสำหรับใช้เป็นข้อมูลนำเข้า (Input) ของโมเดล ซึ่งผู้วิจัยได้สร้างรูปภาพกราฟแท่งเทียนที่กรอบเวลา 5 และ 20 วัน โดยจะพิจารณาเป็นการซื้อขายรายสัปดาห์ และการซื้อขายรายเดือนตามลำดับ โดยในงานวิจัยนี้จึงได้มีการนำโครงข่ายประสาทเทียมแบบคอนโวลูชัน เพื่อสร้างโมเดลทำนายทิศทางราคาข้าวโพดที่มีการซื้อขายล่วงหน้าในตลาดหอกการค้าแห่งนครซิดาโกโดยพิจารณาจากกราฟแท่งเทียน ซึ่งได้มีการนำอัลกอริทึมโครงข่ายประสาทเทียมแบบคอนโวลูชันมาสร้างโมเดลทำนายทิศทางราคาข้าวโพด (Customize CNN) จากนั้นได้มีการนำโมเดลที่เป็นสถาปัตยกรรมแบบคอนโวลูชันได้แก่ VGG-16 และ Efficientnet-B0 เพื่อเปรียบเทียบค่าความแม่นยำ และเพื่อศึกษาไฮเปอร์พารามิเตอร์ที่มีผลต่อโมเดลในการทำนายทิศทางราคาข้าวโพดซึ่งในงานวิจัยนี้ได้แก่ ขนาดชุดข้อมูล (Batch Size) และอัตราการเรียนรู้ (Learning Rate) โดยผู้วิจัยได้ทำการแบ่งข้อมูลออกเป็น 3 ชุด ได้แก่ ชุดฝึกฝน (Training Set) ชุดตรวจสอบ (Validation Set) และชุดทดสอบ (Test Set) โดยอัตราส่วน 70:10:20 เปอร์เซนต์ และมีการเลือกใช้เครื่องมือเปรียบเทียบเป็นค่าความแม่นยำ (Accuracy) เนื่องจากข้อมูลผลเฉลยของตลาดขาขึ้น (Bullish) และตลาดขาลง (Bearish) นั้นไม่มีปัญหาความไม่สมดุลของคลาส (Imbalanced Class)

ผลการวิจัยเพื่อศึกษาค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุดของแต่ละโมเดลนั้นจะใช้ชุดฝึกฝน (Training Set) และชุดตรวจสอบ (Validation Set) เพื่อพิจารณาค่าการสูญเสียที่ต่ำที่สุด และค่าความแม่นยำที่สูงที่สุดของโมเดลในชุดตรวจสอบ (Validation Set) ซึ่งจากผลการวิจัยนี้กรอบเวลา 5 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วัน ขนาด Batch Size และอัตราการเรียนรู้ (Learning Rate) ที่ให้ค่าที่ดีที่สุดของโมเดล Customized CNN, VGG-16 และ Efficientnet-B0 คือ Batch Size ขนาด 16 ที่อัตราการเรียนรู้ 0.001 และกรอบเวลา 20 วัน ขนาด Batch Size และอัตราการเรียนรู้ (Learning Rate) ที่ให้ค่าที่ดีที่สุดของโมเดล Customized CNN คือ Batch Size ขนาด 16 ที่อัตราการเรียนรู้ 0.001 โดยไฮเปอร์พารามิเตอร์ที่ให้ค่าที่ดีที่สุดของโมเดล VGG-16 และ Efficientnet-B0 คือ Batch Size ขนาด 16 ที่อัตราการเรียนรู้ 0.0001

โดยหลังจากได้ค่าไฮเปอร์พารามิเตอร์ที่ดีที่สุดของโมเดล Customized CNN, VGG-16 และ Efficientnet-B0 ที่กรอบเวลา 5 และ 20 วัน จากนั้นทำการทดสอบค่าความแม่นยำด้วยข้อมูลชุดทดสอบ (Test Set) ผลการศึกษาพบว่าโมเดล Customized CNN สามารถทำนายทิศทางของราคาได้แม่นยำด้วยค่าความแม่นยำให้ค่าความแม่นยำที่สูงที่สุด ณ กรอบเวลา 5 วัน โดยให้ค่าความแม่นยำ 55.39% และ โมเดล Efficientnet-B0 ให้ค่าความแม่นยำที่สูงที่สุด ณ กรอบเวลา 20 วัน โดยให้ค่าความแม่นยำ 55.03%

## 5.2 ข้อเสนอแนะ

- 1) ในการศึกษาวิจัยในอนาคตผู้วิจัยเสนอให้ใช้เทคนิคการวิเคราะห์รูปแบบของราคา หรือเรียกอีกอย่างหนึ่งว่า Pattern Analysis
- 2) การทำโมเดลในอนาคตผู้วิจัยเสนอให้มีการใช้เทคนิค Combine Model โดยใช้โมเดลที่ให้ความสำคัญในเรื่องลำดับเวลาอย่าง LSTM มารวมกับโมเดล CNN โดยจะเรียกว่า CNN-LSTM
- 3) ในการศึกษาวิจัยในอนาคตควรนำโมเดลไปทำการซื้อขายแล้ววัดผลตอบแทนในด้านการลงทุนว่าสอดคล้องกับค่าวัดประสิทธิภาพที่ได้หรือไม่

## 5.3 แนวทางที่จะศึกษาต่อในอนาคต

งานวิจัยนี้ได้มีการใช้ข้อมูลรูปภาพแท่งเทียนเพื่อทำนายทิศทางราคาข้าวโพด ดังนั้น การศึกษาวิจัยในอนาคตผู้วิจัยเสนอให้มีการเพิ่มปัจจัยด้านอื่น ๆ อย่างเช่น ปัจจัยข่าว ปัจจัยทางสภาพอากาศมาวิเคราะห์ร่วมกับรูปภาพแท่งเทียนโดยใช้ Stacking Model

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

- กรมเจรจาการค้าระหว่างประเทศ. 2565. สินค้าข้าวโพดเลี้ยงสัตว์. [Online]. เข้าถึงได้จาก <https://www.dtn.go.th/th/content/page/index/id/1048>
- กรมวิชาการเกษตร. 2563. เทคโนโลยีการผลิตข้าวโพดเลี้ยงสัตว์หลังนาในเขตพื้นที่ภาคกลาง. [Online]. เข้าถึงได้จาก <https://www.doa.go.th/share/showthread.php?tid=2431>.
- จตุรพัชร พัฒนทรงศิริโล. 2562. AI ไม่ยาก (เล่ม 1+2) เข้าใจได้ด้วยเลข ม.ปลาย. [Online]. เข้าถึงได้จาก <https://www.mebmarket.com/ebook-108246-AI-ไม่ยาก-เล่ม-1-2-เข้าใจได้ด้วยเลขมัธยมปลาย>.
- ชิตพงษ์ กิตตินราดร. 2563. Neural Network Vanishing Gradients Problem. [Online]. เข้าถึงได้จาก <https://guopai.github.io/ml-blog16.html>.
- โชคชัย เอกทัศนาวรรณ และเกตุอร ทองเครือ. 2561. การปลูกข้าวโพด. [Online]. เข้าถึงได้จาก [http://eto.ku.ac.th/neweto/e-book/plant/herb\\_gar/corn2.pdf](http://eto.ku.ac.th/neweto/e-book/plant/herb_gar/corn2.pdf).
- ดริน พันธุมโกมล. 2547. ความสำคัญและกลไกการทำงานของตลาดสินค้าเกษตรล่วงหน้า. พิมพ์ครั้งที่ 3. กรุงเทพฯ. สำนักงานคณะกรรมการกำกับการซื้อขายสินค้าเกษตรล่วงหน้าตลาดสินค้าเกษตรล่วงหน้าแห่งประเทศไทย.
- ธนัช เบญจอนุอาชา และวราภรณ์ วิทยานนท์. 2022. การจำแนกความเสียหายรถยนต์โดยการเรียนรู้เชิงลึก. วิทยาศาสตร์มหาบัณฑิตสาขาวิทยาการข้อมูล. มหาวิทยาลัยศรีนครินทรวิโรฒ
- นพพร คล้ายพงษ์พันธุ์, เรวัต เลิศฤทัยโยธิน, รังสฤษดิ์ กาวิตะ และสนธิชัย จันทร์เปรม. 2547. พีชเศรษฐกิจ. พิมพ์ครั้งที่ 2. กรุงเทพฯ. สำนักพิมพ์มหาวิทยาลัยเกษตรศาสตร์.
- บรรหาร ทินประบุตร, พุฒิสีทธิ์ ธราพัฒนะพงศ์ และวัลภา ประเสริฐศิลา. 2554. ข้าวโพดกินอร่อย ต้านโรค. กรุงเทพฯ. แบงก์คอกบู้คส์.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไฟโรจน์ วิเศษชาติ. 2549. วิเคราะห์เจาะลึกหุ่นด้วย Candlesticks. กรุงเทพฯ. ซีเอ็ดยูเคชั่น.  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เริงชัย ดันสุชาติ, อารีย์ เชื้อเมืองพาน และมนตรี สิงหะวาระ. 2554. พฤติกรรมราคา ความเสี่ยง และ การพยากรณ์ราคาสินค้าเกษตรล่วงหน้าของประเทศไทย. วิทยาศาสตร์บัณฑิตสาขา เศรษฐศาสตร์. มหาวิทยาลัยนเรศวร

สาธิต จันทร์เอี่ยม. 2563. ศึกษาศักยภาพการเพาะปลูกข้าวโพดเลี้ยงสัตว์: กรณีศึกษา เกษตรกร อำเภอแม่พริก จังหวัดลำปาง. ปริญญาวิทยาศาสตรบัณฑิตสาขาภูมิศาสตร์. มหาวิทยาลัย นเรศวร

สำนักงานเศรษฐกิจการเกษตร. 2564. ข้าวโพดเลี้ยงสัตว์. [Online]. เข้าถึงได้จาก <https://www.oae.go.th/view/1/ภาวะเศรษฐกิจการเกษตร/TH-TH>.

สำนักหอสมุดและศูนย์สารสนเทศวิทยาศาสตร์และเทคโนโลยี. 2561. ประมวลสารสนเทศพร้อมใช้ เรื่อง “ข้าวโพด (Corn)”. [Online]. เข้าถึงได้จาก <http://siweb1.dss.go.th/repack/fulltext/IR47.pdf>.

สิริพร ธรรมเกษตร. 2561. กลยุทธ์การซื้อขายโดนการใช้คุณลักษณะของกราฟแท่งเทียน. วิทยาศาสตร์ มหาบัณฑิต สาขาสถิติประยุกต์. สถาบันพัฒนาบริหารศาสตร์

สุริยะ ชยะธรรมกุล. 2020. การจำแนกผลึกน้ำตาลด้วยการเรียนรู้เชิงลึก. วิทยาศาสตร์มหาบัณฑิต สาขาวิชาวิศวกรรมซอฟต์แวร์. จุฬาลงกรณ์มหาวิทยาลัย.

หลักทรัพย์บัวหลวง. 2564. วิเคราะห์กราฟหุ้นเบื้องต้นแบบนักลงทุนสายเทคนิค. [Online]. เข้าถึงได้ จาก <https://knowledge.bualuang.co.th/knowledge-base/howtouseindicators/>.

Afshine, A and Shervine, A. 2018. VIP Cheatsheet: Convolutional Neural Networks. [Online]. Available <https://github.com/afshinea/stanford-cs-230-deep-learning/blob/master/en/cheatsheet-convolutional-neural-networks.pdf>.

Agustika, D, Ariyanti, N., Wardana, I., Iliescu, D. and Leeson, M. 2021. Classification of Chili Plant Origin by Using Multilayer Perceptron Neural Network. 2021 8th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI). Semarang.

Ahmed, W. and Karim, A. 2020. The Impact of Filter Size and Number of Filters on Classification Accuracy in CNN. 2020 International Conference on Computer Science and Software Engineering (CSASE). Duhok.

Allibha, E. 2018. Building A Deep Learning Model using Keras. [Online]. Available <https://towardsdatascience.com/building-a-deep-learning-model-using-keras-1548ca149d37>.

Ferguson, M, Ak, R., Lee, Y. and Law, K. 2017. Automatic Localization of Casting Defects with Convolutional Neural Networks. In 2017 IEEE International Conference on Big Data (BIGDATA). Boston.

Goodfellow, I, Bengio, Y. and Courville, A. 2016. Deep Learning. United States of America. The MIT Press.

Gradojevic, N, and Gençay, R. 2011. Financial Applications of Nonextensive Entropy [Applications Corner]. IEEE Signal Processing Magazine, 28(5), 116–141.

Guo, C, Chen, X. and Yu, C. 2022. Multi-Stage Attentive Network for Motion Deblurring via Binary Cross-Entropy Loss. MDPI Journals Awarded Impact Factor. 24: 1414.

Gurucharan, M. 2022. Basic CNN Architecture: Explaining 5 Layers of Convolutional Neural Network. [Online]. Available <https://www.upgrad.com/blog/basic-cnn-architecture/>.

Kandel, I. and Castelli, M. 2020. The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset. The ICT Express journal. 3: 312-315.

Kim, D. and Gofman, M. 2018. Comparison of Shallow and Deep Neural Networks for Network Intrusion Detection. In 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC). Las Vegas.

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อเผยแพร่บนเว็บไซต์ของศูนย์วิจัยนโยบายด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Kingma, D and Ba, J. 2015 Adam: A Method for Stochastic Optimization. [Online]. Available <https://arxiv.org/pdf/1412.6980.pdf>.
- Krizhevsky, A, Sutskever, I. and Hinton, G. 2012. ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems. 25: 1090–1098.
- Kustrin, S. and Beresford, R. 1999. Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research. Journal of Pharmaceutical and Biomedical Analysis. 22: 717–727.
- Kusuma, R, Ho, T., Kao, W., Ou, Y. and Hua, K. 2019. Using Deep Learning Neural Networks and Candlestick Chart Representation to Predict Stock Market. Department of Computer Science and Engineering. Yuan Ze University
- Kwok, N, Wang, D., Jia, X., Chen, S., Fang, G. and Ha, Q. 2011. Gray world based color correction and intensity preservation for image enhancement. 2011 4th International Congress on Image and Signal Processing. Shanghai.
- Lambert, C. 2009. Candlestick Charts: An introduction to using candlestick charts. United Kingdom. MPG Books Group.
- Liu, J, Wang, M, Bao, L. and Li, X. 2020. EfficientNet based recognition of maize diseases by leaf image classification. Journal of Physics Conference Series. 1693: 012148.
- Mack, C. 2017. Neural networks. [Online]. Available <https://towardsdatascience.com/machine-learning-fundamentals-ii-neural-networks-f1e7b2cb3eef>.
- Man, R. 2020. รู้จักแท่งเทียน เขียว -แดงในหุ้น. [Online]. เข้าถึงได้จาก <https://stockradars.news/2020/05/27/longruay-candlestick/>.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Mathew, A, Amudha, P. and Sivakumari, S. 2021. Deep Learning Techniques: An Overview. *Advanced Machine Learning Technologies and Applications*. 584-593.
- Mustafa, E, Elshafey, M. and Fouad, M. 2019. Accuracy Enhancement of a Blind ImageSteganalysis Approach Using Dynamic Learning Rate-Based CNN on GPUs. *The 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications*. Metz.
- Padmavathi, K, Thalla, O., Reddy, S., Yadlapalli, P., Roshan, T. and Charan, T. 2022. Transfer Learning Using EfficientNet for Brain Tumor Classification from MRI Images. *International Conference on Computer Communication and Informatics (ICCCI)*. Coimbatore, INDIA.
- Saha, S. 2018. A Comprehensive Guide to Convolutional Neural Networks — the ELI5. [Online]. Available <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>.
- SHARMA, S. 2017. Activation Functions in Neural Networks. [Online]. Available <https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6>.
- Simonyan, K, and Zisserman, A. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. Department of Engineering Science, University of Oxford
- Sonka, M, Hlavac, V. and Boyle, R. 2013. *Image Processing, Analysis, and Machine Vision*. Cengage Learning. United States of America. Cengage Learning.
- Sultana, F, Sufian, A. and Dutta, P. 2018. Advancements in Image Classification Using Convolutional Neural Network. [Online]. Available <https://arxiv.org/pdf/1905.03288.pdf>.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Tan, M., Le, Q., 2019. Efficientnet: Rethinking Model Scaling for Convolutional Neural Networks. In: International Conference on Machine Learning. PMLR, pp. 6105–6114. [Online]. Available <https://arxiv.org/pdf/1905.11946.pdf>



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ก

### ชุดคำสั่งที่ใช้ในงานวิจัย

การทำนายการเคลื่อนไหวของราคาข้าวโพดในตลาดหอการค้าแห่งนครชิคาโกโดยใช้รูปภาพกราฟแท่งเทียนในภาษาไพธอน (Python) มีคำสั่งดังต่อไปนี้

#### 1. คำสั่งสำหรับสร้างกราฟแท่งเทียน

##### 1.1 คำสั่งจัดเตรียมข้อมูลก่อนสร้างกราฟแท่งเทียน

```
# Import library
import pandas as pd
import plotly.graph_objects as go

df = pd.read_excel('./Price.xlsx',sheet_name='Corn')
df['Dates'] = pd.to_datetime(df['Dates']).dt.date
data = df[:].copy().reset_index(drop=True)

Name_DATA = pd.DataFrame(columns=['File Name'])
```

##### 1.2 คำสั่งสร้างรูปภาพกราฟแท่งเทียนกรอบเวลา 5 วัน

```
window = 5
stride = 1
for i in range(0, len(data) - window + 1, stride):
    start = i
    end = i+window if i+window < len(data) else len(data)
    tmp = data[start:end]
    print(start,end,len(tmp))
    start_date = tmp.at[start,'Dates'].strftime("%Y%m%d")
    end_date = tmp.at[end-1,'Dates'].strftime("%Y%m%d")
    print("{}_{}".format(start_date,end_date)) #แสดงข้อมูล (start_date,end_date)
    filename = "{}_{}.png".format("image",i)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อวารสารวิจัยเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

print(filename)
#graph
fig = go.Figure(data=[go.Candlestick(x=tmp.Dates,
    open=tmp.OPEN, high=tmp.HIGH,
    low=tmp.LOW, close=tmp.LAST)])

fig.update_traces(increasing=dict(fillcolor='#3D9970',line=dict(color='#3D9970')),
decreasing=dict(fillcolor='#FF4136', line=dict(color='#FF4136')))

fig.update_layout(xaxis_rangeslider_visible=False,margin=dict(l=0,r=0,b=0,t=0),
plot_bgcolor = "black")

fig.update_xaxes(showgrid=True, gridwidth=1,
gridcolor='black',showticklabels=False,type='category')
fig.update_yaxes(showgrid=True, gridwidth=1,
gridcolor='black',showticklabels=False)

fig.show()
fig.write_image("picture graph/Back5/{}".format(filename), format='png')

Name_DATA = Name_DATA.append({'File Name': filename},ignore_index=True)
Name_DATA.to_excel('picture graph/file_5.xlsx', index=False)

```

### 1.3 แปลงราคาข้าวโพดก่อนเข้าโมเดลรอบเวลา 5 วัน

```

tmp5 = data[['Dates','LAST']].copy()
tmp5

tmp5['LAST1'] = data['LAST'].shift(-4)
tmp5['LAST2'] = data['LAST'].shift(-5)
tmp5['pctDiff_Next'] = (tmp5['LAST2'] - tmp5['LAST1'])

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น ยกเว้นผู้ที่มีเหตุเปลี่ยนแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
def dir(pct,criteria=0):
```

```

if pct >= criteria:
    return 1
elif pct < criteria:
    return -0
else:
    return None

```

```
tmp5['label'] = tmp5['pctDiff_Next'].apply(lambda x: dir(x))
```

```

df = pd.concat([Name_DATA, tmp1['label']], axis=1)
df.to_excel('picture graph/ImageAndLB_5.xlsx', index=False)

```

#### 1.4 คำสั่งสร้างรูปภาพกราฟแท่งเทียนกรอบเวลา 20 วัน

```

window = 20
stride = 1
for i in range(0, len(data) - window + 1, stride):
    start = i
    end = i+window if i+window < len(data) else len(data)
    tmp = data[start:end]
    print(start,end,len(tmp))
    start_date = tmp.at[start,'Dates'].strftime("%Y%m%d")
    end_date = tmp.at[end-1,'Dates'].strftime("%Y%m%d")
    print("{} {}".format(start_date,end_date)) #แสดงข้อมูล (start_date,end_date)
    filename = "{}_{}.png".format("image",i)
    print(filename)

#graph
fig = go.Figure(data=[go.Candlestick(x=tmp.Dates,
    open=tmp.OPEN, high=tmp.HIGH,
    low=tmp.LOW, close=tmp.LAST)])

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของโรงเรียนเพื่อการศึกษาเท่านั้น เมื่อผู้ใดเห็นประโยชน์ของเอกสารนี้  
ไม่ว่ากรณีใดๆ ทั้งสิ้น ยกเว้น มิได้เห็นแต่เพียงเนื้อหา และต้องอ้างอิงถึงชื่อของเอกสารทุกครั้งที่มีการนำไปใช้

```
fig.update_layout(xaxis_rangeslider_visible=False,margin=dict(l=0,r=0,b=0,t=0),
plot_bgcolor = "black")
```

```
fig.update_xaxes(showgrid=True, gridwidth=1,
gridcolor='black',showticklabels=False,type='category')
```

```
fig.update_yaxes(showgrid=True, gridwidth=1,
gridcolor='black',showticklabels=False)
```

```
fig.show()
```

```
fig.write_image("picture graph/Back5/{}".format(filename), format='png')
```

```
Name_DATA = Name_DATA.append({'File Name': filename},ignore_index=True)
```

```
Name_DATA.to_excel('picture graph/file_20.xlsx', index=False)
```

### 1.5 แปลงราคาข้าวโพดก่อนเข้าโมเดลรอบเวลา 20 วัน

```
tmp20 = data[['Dates','LAST']].copy()
```

```
tmp20['LAST1'] = data['LAST'].shift(-19)
```

```
tmp20['LAST2'] = data['LAST'].shift(-20)
```

```
tmp20['pctDiff_Next'] = (tmp20['LAST2'] - tmp20['LAST1'])
```

```
def dir(pct,criteria=0):
```

```
    if pct >= criteria:
```

```
        return 1
```

```
    elif pct < criteria:
```

```
        return -0
```

```
    else:
```

```
        return None
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของกรมส่งเสริมการค้าระหว่างประเทศ กระทรวงพาณิชย์ และผู้จัดทำเพื่อให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
df = pd.concat([Name_DATA, tmp20['label']], axis=1)
df.to_excel('picture graph/ImageAndLB_20.xlsx', index=False)
```

## 2. คำสั่งสำหรับจัดเตรียมข้อมูลก่อนเข้าโมเดล

### 2.1 คำสั่งสำหรับการเชื่อมต่อข้อมูล และเรียกใช้งานไลบรารีเบื้องต้น

```
#เชื่อมต่อข้อมูลกับGoogle Drive
```

```
from google.colab import drive
```

```
drive.mount('/content/drive/')
```

```
!pwd
```

```
%cd /content/drive/MyDrive
```

```
!pwd
```

```
# Import library
```

```
import os
```

```
import pandas as pd
```

```
import numpy as np
```

```
import tensorflow as tf
```

```
from tensorflow import keras
```

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

```
from sklearn.model_selection import train_test_split
```

```
import matplotlib.pyplot as plt
```

```
from tensorflow.keras.layers import Dense, GlobalAveragePooling2D
```

```
from tensorflow.keras.models import Model
```

```
from tensorflow.keras.optimizers import Adam
```

### 2.2 การนำเข้าข้อมูลในไฟล์ Excel ของกรอบเวลา 5 และ 20 วัน

```
#นำเข้าข้อมูลในไฟล์Excelของกรอบเวลา5วัน
```

```
df5 = pd.read_excel("Image/JPTNB_TEST/picture
```

```
graph/ImageAndLB_5.xlsx",sheet_name='Sheet1')
```

```
# Convert the 'label' column to string type
```

```
df5['label'] = df5['label'].astype(str)
```

```
#นำเข้าข้อมูลของกรอบเวลา20วัน
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์หรือการแจ้งขึ้นเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
df = pd.read_excel("Image/JPTNB_TEST/picture
graph/ImageAndLB_20.xlsx",sheet_name='Sheet1')
# Convert the 'label' column to string type
df['label'] = df['label'].astype(str)
```

### 2.3 การแบ่ง Train:Validation:Test อัตราส่วน 70:10:20 ของกรอบเวลา 5 และ 20 วัน

```
# แบ่งข้อมูลเป็น train val test ของกรอบเวลา 5 วัน
train_df5, test_df5 = train_test_split(df5, test_size=0.2,
shuffle=False,random_state=1)
train_df5, val_df5 = train_test_split(train_df5, test_size=0.1/0.8,
shuffle=False,random_state=1)
```

```
# แบ่งข้อมูลเป็น train val test ของกรอบเวลา 20 วัน
train_df, test_df = train_test_split(df, test_size=0.2,
shuffle=False,random_state=1)
train_df, val_df = train_test_split(train_df, test_size=0.1/0.8,
shuffle=False,random_state=1)
```

### 2.4 การโหลดรูปภาพ และกำหนดขนาดของ Batch Size ของกรอบเวลา 5 วัน

```
#กำหนดตัวแปรเรียกใช้รูปภาพกรอบเวลา5วัน
image_back5 = 'Image/JPTNB_TEST/picture graph/Back5'
seed_value = 16
```

```
# กำหนดตัวแปร image data generators สำหรับ Batch Size 16 ที่กรอบเวลา5วัน
train_image_gen5_BZ16 = ImageDataGenerator(rescale = 1./255)
val_image_gen5_BZ16 = ImageDataGenerator(rescale = 1./255)
test_image_gen5_BZ16 = ImageDataGenerator(rescale = 1./255)
```

```
# กำหนดตัวแปร image data generators สำหรับ Batch Size 256 ที่กรอบเวลา20วัน
train_image_gen5_BZ256 = ImageDataGenerator(rescale = 1./255)
val_image_gen5_BZ256 = ImageDataGenerator(rescale = 1./255)
test_image_gen5_BZ256 = ImageDataGenerator(rescale = 1./255)
```

```
#กำหนดค่า Batch Size = 16
```

```
train_generator5_BZ16 = train_image_gen5_BZ16.flow_from_dataframe(train_df5,
directory=image_back5, x_col='File Name', y_col='label',class_mode='binary',seed
= seed_value, target_size=(224,224),
batch_size=16,color_mode='rgb',subset="training")
```

```
val_generator5_BZ16 = val_image_gen5_BZ16.flow_from_dataframe(val_df5,
directory=image_back5, x_col='File Name', y_col='label',class_mode='binary',
seed= seed_value,target_size=(224,224),batch_size=16,color_mode='rgb')
```

```
test_generator5_BZ16 = test_image_gen5_BZ16.flow_from_dataframe(test_df5,
directory=image_back5, x_col='File Name',y_col='label',class_mode='binary',seed=
seed_value, target_size=(224,224),batch_size=16, color_mode='rgb')
```

```
#กำหนดค่า Batch Size = 256
```

```
train_generator5_BZ256 = train_image_gen5_BZ256
.flow_from_dataframe(train_df5, directory=image_back5, x_col='File Name',
y_col='label',class_mode='binary',seed = seed_value, target_size=(224,224),
batch_size=256, color_mode='rgb',subset="training")
```

```
val_generator5_BZ256 = val_image_gen5_BZ256 .flow_from_dataframe(val_df5,
directory=image_back5, x_col='File Name', y_col='label',class_mode='binary', seed
= seed_value,target_size=(224,224),batch_size=256,color mode='rgb')
```

```
test_generator5_BZ256 = test_image_gen5_BZ256 flow_from_dataframe(test_df5,
directory=image_back5, x_col='File Name', y_col='label',class_mode='binary',seed
= seed_value, target_size=(224,224),batch_size=256, color_mode='rgb')
```

## 2.5 การไหลรูปภาพ และกำหนดขนาดของ Batch Size ของกรอบเวลา 20 วัน

```
#กำหนดตัวแปรเรียกใช้รูปภาพกรอบเวลา20วัน
```

```
image_back20 = 'Image/JPTNB_TEST/picture graph/Back20'
```

```
seed_value = 16
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น ยกเว้นที่ มิมีเหตุพิเศษใดๆ และต้องขออนุญาตจากเจ้าของเอกสารไว้ก่อนนำไปใช้

```

train_image_gen20_BZ16 = ImageDataGenerator(rescale = 1./255)
val_image_gen20_BZ16 = ImageDataGenerator(rescale = 1./255)
test_image_gen20_BZ16 = ImageDataGenerator(rescale = 1./255)

# กำหนดตัวแปร image data generators สำหรับ Batch Size 256 ที่กรอบเวลา20วัน
train_image_gen20_BZ256 = ImageDataGenerator(rescale = 1./255)
val_image_gen20_BZ256 = ImageDataGenerator(rescale = 1./255)
test_image_gen20_BZ256 = ImageDataGenerator(rescale = 1./255)

#กำหนดค่า Batch Size = 16
train_generator20_BZ16 = train_image_gen20_BZ16
.flow_from_dataframe(train_df, directory=image_back20, x_col='File Name',
y_col='label',class_mode='binary',seed = seed_value, target_size=(224,224),
batch_size=16,color_mode='rgb',subset="training")

val_generator20_BZ16 = val_image_gen20_BZ16 .flow_from_dataframe(val_df,
directory=image_back20, x_col='File Name', y_col='label',class_mode='binary',
seed = seed_value,target_size=(224,224),batch_size=16,color_mode='rgb')

test_generator20_BZ16 = test_image_gen20_BZ16 .flow_from_dataframe(test_df,
directory=image_back20, x_col='File Name',
y_col='label',class_mode='binary',seed = seed_value, target_size=(224,224),
batch_size=16, color_mode='rgb')

#กำหนดค่า Batch Size = 256
train_generator20_BZ256 = train_image_gen20_BZ256
.flow_from_dataframe(train_df, directory=image_back20, x_col='File Name',
y_col='label',class_mode='binary',seed = seed_value, target_size=(224,224),
batch_size=256,color_mode='rgb',subset="training")

val_generator20_BZ256 = val_image_gen20_BZ256 .flow_from_dataframe(val_df,
directory=image_back20, x_col='File Name', y_col='label',class_mode='binary',
seed = seed_value,target_size=(224,224),batch_size=256,color_mode='rgb')

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของโรงเรียนเพื่อการศึกษาเท่านั้น เมื่อผู้ใดเห็นหรือใช้ประโยชน์จากการค้า  
ไม่ว่ากรณีใดๆ พงษ์สัน ยกให้หา ไม่มีเห็นแต่เพียงผู้เดียว และต้องอ้างอิงถึงชื่อของเอกสารนี้ทุกครั้งที่นำไปใช้

```
test_generator20_BZ256 = test_image_gen20_BZ256
.flow_from_dataframe(test_df, directory=image_back20, x_col='File Name',
y_col='label',class_mode='binary',seed = seed_value,
target_size=(224,224),batch_size=256, color_mode='rgb')
```

### 3. การสร้างโมเดล ณ ระยะเวลา 5 วัน

#### 3.1 การสร้างโมเดล Customized CNN ณ ระยะเวลา 5 วัน

#สร้างโมเดล CNN ระยะเวลา5วัน

```
modelCNN5 = keras.Sequential([
    keras.layers.Conv2D(32, (3,3), activation='relu',
input_shape=(224,224,3),padding = 'same'),
    keras.layers.MaxPooling2D((2,2)),
    keras.layers.Flatten(),
    keras.layers.Dense(32, activation='relu'),
    keras.layers.Dropout(0.5),
    keras.layers.Dense(1, activation='sigmoid')
])
```

##### 3.1.1 โมเดล Customized CNN ที่ Batch Size = 16 และ Learning rate = 0.001

```
adam_opt_001 = tf.keras.optimizers.Adam(learning_rate=0.001)
modelCNN5.compile(optimizer=adam_opt_001, loss='binary_crossentropy',
metrics=['accuracy'])
```

# Train the model

```
modelCNN5_16_lr001_history = modelCNN5.fit(train_generator5_BZ16,
epochs=20, validation_data=val_generator5_BZ16)
```

# Evaluate model on the testing data

```
test_loss, test_acc = modelCNN5.evaluate(test_generator5_BZ16, verbose=1)
```

```
print(f"Test loss: {test_loss}")
```

```
print(f"Test accuracy: {test_acc}")
```

เอกสารนี้เป็นเอกสารที่ลงนามและประทับตราของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.2 โมเดล Customized CNN ที่ Batch Size = 16 และ Learning rate = 0.0001

```
adam_opt_0001 = tf.keras.optimizers.Adam(learning_rate=0.0001)
modelCNN5.compile(optimizer=adam_opt_0001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
```

```
modelCNN5_16_lr0001_history = modelCNN5.fit(train_generator5_BZ16,
epochs=20, validation_data=val_generator5_BZ16)
```

```
# Evaluate model on the testing data
```

```
test_loss, test_acc = modelCNN5.evaluate(test_generator5_BZ16, verbose=1)
print(f"Test loss: {test_loss}")
print(f"Test accuracy: {test_acc}")
```

### 3.1.3 โมเดล Customized CNN ที่ Batch Size = 256 และ Learning rate = 0.001

```
adam_opt_001 = tf.keras.optimizers.Adam(learning_rate=0.001)
modelCNN5.compile(optimizer=adam_opt_001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
```

```
modelCNN5_256_lr001_history = modelCNN5.fit(train_generator5_BZ256,
epochs=20, validation_data=val_generator5_BZ256)
```

```
# Evaluate model on the testing data
```

```
test_loss, test_acc = modelCNN5.evaluate(test_generator5_BZ256, verbose=1)
print(f"Test loss: {test_loss}")
print(f"Test accuracy: {test_acc}")
```

### 3.1.4 โมเดล Customized CNN ที่ Batch Size = 256 และ Learning rate = 0.0001

```
adam_opt_0001 = tf.keras.optimizers.Adam(learning_rate=0.0001)
modelCNN5.compile(optimizer=adam_opt_0001, loss='binary_crossentropy',
metrics=['accuracy'])
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับใช้ทำงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
# Train the model
modelCNN5_256_lr0001_history = modelCNN5.fit(train_generator5_BZ256,
epochs=20, validation_data=val_generator5_BZ256)

# Evaluate model on the testing data
test_loss, test_acc = modelCNN5.evaluate(test_generator5_BZ256, verbose=1)
print(f"Test loss: {test_loss}")
print(f"Test accuracy: {test_acc}")
```

### 3.2 การสร้างโมเดล VGG-16 ณ ระยะเวลา 5 วัน

#สร้างโมเดล VGG-16 ระยะเวลา5วัน

```
from tensorflow.keras.applications.vgg16 import VGG16

vgg_model = VGG16(include_top=False,
input_shape=(224,224,3),weights="imagenet")

for layer in vgg_model.layers:
    layer.trainable = False

top_model = keras.Sequential([
    keras.layers.Flatten(input_shape=vgg_model.output_shape[1:]),
    keras.layers.Dense(32, activation='relu'),
    keras.layers.Dropout(0.5),
    keras.layers.Dense(1, activation='sigmoid')])

modelVGG5 = keras.models.Model(inputs=vgg_model.input,
outputs=top_model(vgg_model.output))
```

#### 3.2.1 โมเดล VGG-16 ที่ Batch Size = 16 และ Learning rate = 0.001

```
adam_opt_001 = tf.keras.optimizers.Adam(learning_rate=0.001)
modelVGG5.compile(optimizer=adam_opt_001, loss='binary_crossentropy',
metrics=['accuracy'])
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
# Train the model
modelVGG5_16_lr001_history = modelVGG5.fit(train_generator5_BZ16,
epochs=20, validation_data=val_generator5_BZ16)

# Evaluate model on the testing data
test_loss, test_acc = modelVGG5.evaluate(test_generator5_BZ16, verbose=1)
print(f"Test loss: {test_loss}")
print(f"Test accuracy: {test_acc}")
```

### 3.2.2 โมเดล VGG-16 ที่ Batch Size = 16 และ Learning rate = 0.0001

```
adam_opt_0001 = tf.keras.optimizers.Adam(learning_rate=0.0001)
modelVGG5.compile(optimizer=adam_opt_0001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
modelVGG5_16_lr0001_history = modelVGG5.fit(train_generator5_BZ16,
epochs=20, validation_data=val_generator5_BZ16)

# Evaluate model on the testing data
test_loss, test_acc = modelVGG5.evaluate(test_generator5_BZ16, verbose=1)
print(f"Test loss: {test_loss}")
print(f"Test accuracy: {test_acc}")
```

### 3.2.3 โมเดล VGG-16 ที่ Batch Size = 256 และ Learning rate = 0.001

```
adam_opt_001 = tf.keras.optimizers.Adam(learning_rate=0.001)
modelVGG5.compile(optimizer=adam_opt_001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
modelVGG5_256_lr001_history = modelVGG5.fit(train_generator5_BZ256,
epochs=20, validation_data=val_generator5_BZ256)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น ยกเว้นที่ ผมได้เห็นแบบลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
# Evaluate model on the testing data
```

```
test_loss, test_acc = modelVGG5.evaluate(test_generator5_BZ256, verbose=1)
print(f"Test loss: {test_loss}")
print(f"Test accuracy: {test_acc}")
```

### 3.2.4 โมเดล VGG-16 ที่ Batch Size = 256 และ Learning rate = 0.0001

```
adam_opt_0001 = tf.keras.optimizers.Adam(learning_rate=0.0001)
modelVGG5.compile(optimizer=adam_opt_0001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
```

```
modelVGG5_256_lr0001_history = modelVGG5.fit(train_generator5_BZ256,
epochs=20, validation_data=val_generator5_BZ256)
```

```
# Evaluate model on the testing data
```

```
test_loss, test_acc = modelVGG5.evaluate(test_generator5_BZ256, verbose=1)
print(f"Test loss: {test_loss}")
print(f"Test accuracy: {test_acc}")
```

### 3.3 การสร้างโมเดล EfficientNet-B0 ณ ระยะเวลา 5 วัน

```
#สร้างโมเดล EfficientNet-B0 ระยะเวลา5วัน
```

```
base_model = EfficientNetB0(include_top=False, weights="imagenet",
input_shape=(224,224,3))
```

```
for layer in base_model.layers:
```

```
    layer.trainable = False
```

```
x = base_model.output
```

```
x = GlobalAveragePooling2D()(x)
```

```
output = Dense(1, activation='sigmoid')(x)
```

```
modelEF5 = Model(inputs=base_model.input, outputs=output)
```

### 3.3.1 โมเดล EfficientNet-B0 ที่ Batch Size = 16 และ Learning rate = 0.001

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์เพื่อการศึกษาเท่านั้น เมื่อผู้จัดทำเอกสารนี้ไปเผยแพร่โดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆ ทั้งสิ้น ผู้จัดทำเอกสารนี้ขอสงวนสิทธิ์และขอสงวนใจในเอกสารนี้ทุกครั้งที่มีการนำไปใช้

```
adam_opt_001 = tf.keras.optimizers.Adam(learning_rate=0.001)
```

```
modelEF5.compile(optimizer=adam_opt_001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
```

```
modelEF5_16_lr001_history = modelEF5.fit(train_generator5_BZ16,
epochs=20, validation_data=val_generator5_BZ16)
```

```
# Evaluate model on the testing data
```

```
test_loss, test_acc = modelEF5.evaluate(test_generator5_BZ16, verbose=1)
print(f"Test loss: {test_loss}")
print(f"Test accuracy: {test_acc}")
```

### 3.3.2 โมเดล EfficientNet-B0 ที่ Batch Size = 16 และ Learning rate = 0.0001

```
adam_opt_0001 = tf.keras.optimizers.Adam(learning_rate=0.0001)
modelEF5.compile(optimizer=adam_opt_0001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
```

```
modelEF5_16_lr0001_history = modelEF5.fit(train_generator5_BZ16,
epochs=20, validation_data=val_generator5_BZ16)
```

```
# Evaluate model on the testing data
```

```
test_loss, test_acc = modelEF5.evaluate(test_generator5_BZ16, verbose=1)
print(f"Test loss: {test_loss}")
print(f"Test accuracy: {test_acc}")
```

### 3.3.3 โมเดล EfficientNet-B0 ที่ Batch Size = 256 และ Learning rate = 0.001

```
adam_opt_001 = tf.keras.optimizers.Adam(learning_rate=0.001)
modelEF5.compile(optimizer=adam_opt_001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
```

```
modelEF5_256_lr001_history = modelEF5.fit(train_generator5_BZ256,
epochs=20, validation_data=val_generator5_BZ256)
```

```
# Evaluate model on the testing data
test_loss, test_acc = modelEF5.evaluate(test_generator5_BZ256, verbose=1)
print(f"Test loss: {test_loss}")
print(f"Test accuracy: {test_acc}")
```

### 3.3.4 โมเดล EfficientNet-B0 ที่ Batch Size = 256 และ Learning rate = 0.0001

```
adam_opt_0001 = tf.keras.optimizers.Adam(learning_rate=0.0001)
modelEF5.compile(optimizer=adam_opt_0001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
modelEF5_256_lr0001_history = modelEF5.fit(train_generator5_BZ256,
epochs=20, validation_data=val_generator5_BZ256)

# Evaluate model on the testing data
test_loss, test_acc = modelEF5.evaluate(test_generator5_BZ256, verbose=1)
print(f"Test loss: {test_loss}")
print(f"Test accuracy: {test_acc}")
```

## 4. การสร้างโมเดล ณ ระยะเวลา 20 วัน

### 4.1 การสร้างโมเดล Customized CNN

#สร้างโมเดล CNN ระยะเวลา20วัน

```
modelCNN20 = keras.Sequential([
    keras.layers.Conv2D(32, (3,3), activation='relu',
input_shape=(224,224,3),padding = 'same'),
    keras.layers.MaxPooling2D((2,2)),
    keras.layers.Flatten(),
    keras.layers.Dense(32, activation='relu'),
    keras.layers.Dropout(0.5),
    keras.layers.Dense(1, activation='sigmoid')
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์หรือการสงวนเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.1 โมเดล Custimized CNN ที่ Batch Size = 16 และ Learning rate = 0.001

```
adam_opt_001 = tf.keras.optimizers.Adam(learning_rate=0.001)
modelCNN20.compile(optimizer=adam_opt_001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
```

```
modelCNN20_16_lr001_history = modelCNN20.fit(train_generator20_BZ16,
epochs=20, validation_data=val_generator20_BZ16)
```

```
# Evaluate model on the testing data
```

```
test_loss, test_acc = modelCNN20.evaluate(test_generator20_BZ16,
verbose=1)
print(f"Test loss: {test_loss}")
print(f"Test accuracy: {test_acc}")
```

#### 4.1.2 โมเดล Custimized CNN ที่ Batch Size = 16 และ Learning rate = 0.0001

```
adam_opt_0001 = tf.keras.optimizers.Adam(learning_rate=0.0001)
modelCNN20.compile(optimizer=adam_opt_0001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
```

```
modelCNN20_16_lr0001_history = modelCNN20.fit(train_generator20_BZ16,
epochs=20, validation_data=val_generator20_BZ16)
```

```
# Evaluate model on the testing data
```

```
test_loss, test_acc = modelCNN20.evaluate(test_generator20_BZ16,
verbose=1)
print(f"Test loss: {test_loss}")
print(f"Test accuracy: {test_acc}")
```

#### 4.1.3 โมเดล Custimized CNN ที่ Batch Size = 256 และ Learning rate = 0.001

```
adam_opt_001 = tf.keras.optimizers.Adam(learning_rate=0.001)
```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
modelCNN20.compile(optimizer=adam_opt_001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
```

```
modelCNN20_256_lr001_history = modelCNN20.fit(train_generator20_BZ256,
epochs=20, validation_data=val_generator20_BZ256)
```

```
# Evaluate model on the testing data
```

```
test_loss, test_acc = modelCNN20.evaluate(test_generator20_BZ256,
verbose=1)
```

```
print(f"Test loss: {test_loss}")
```

```
print(f"Test accuracy: {test_acc}")
```

#### 4.1.4 โมเดล Customized CNN ที่ Batch Size = 256 และ Learning rate = 0.0001

```
adam_opt_0001 = tf.keras.optimizers.Adam(learning_rate=0.0001)
```

```
modelCNN20.compile(optimizer=adam_opt_0001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
```

```
modelCNN20_256_lr0001_history = modelCNN20.fit(train_generator20_BZ256,
epochs=20, validation_data=val_generator20_BZ256)
```

```
# Evaluate model on the testing data
```

```
test_loss, test_acc = modelCNN20.evaluate(test_generator20_BZ256,
verbose=1)
```

```
print(f"Test loss: {test_loss}")
```

```
print(f"Test accuracy: {test_acc}")
```

## 4.2 การสร้างโมเดล VGG-16

```
#สร้างโมเดล VGG-16 ครอบเวลา20วัน
```

```
from tensorflow.keras.applications.vgg16 import VGG16
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
vgg_model = VGG16(include_top=False,
input_shape=(224,224,3),weights="imagenet")
```

```
for layer in vgg_model.layers:
```

```
    layer.trainable = False
```

```
top_model = keras.Sequential([
    keras.layers.Flatten(input_shape=vgg_model.output_shape[1:]),
    keras.layers.Dense(32, activation='relu'),
    keras.layers.Dropout(0.5),
    keras.layers.Dense(1, activation='sigmoid')])
```

```
modelVGG20 = keras.models.Model(inputs=vgg_model.input,
outputs=top_model(vgg_model.output))
```

#### 4.2.1 โมเดล VGG-16 ที่ Batch Size = 16 และ Learning rate = 0.001

```
adam_opt_001 = tf.keras.optimizers.Adam(learning_rate=0.001)
modelVGG20.compile(optimizer=adam_opt_001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
```

```
modelVGG20_16_lr001_history = modelVGG20.fit(train_generator20_BZ16,
epochs=20, validation_data=val_generator20_BZ16)
```

```
# Evaluate model on the testing data
```

```
test_loss, test_acc = modelVGG20.evaluate(test_generator20_BZ16,
verbose=1)
```

```
print(f"Test loss: {test_loss}")
```

```
print(f"Test accuracy: {test_acc}")
```

#### 4.2.2 โมเดล VGG-16 ที่ Batch Size = 16 และ Learning rate = 0.0001

```
adam_opt_0001 = tf.keras.optimizers.Adam(learning_rate=0.0001)
```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสถาบันวิจัยและพัฒนาเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
modelVGG20.compile(optimizer=adam_opt_0001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
```

```
modelVGG20_16_lr0001_history = modelVGG20.fit(train_generator20_BZ16,
epochs=20, validation_data=val_generator20_BZ16)
```

```
# Evaluate model on the testing data
```

```
test_loss, test_acc = modelVGG20.evaluate(test_generator20_BZ16,
verbose=1)
```

```
print(f"Test loss: {test_loss}")
```

```
print(f"Test accuracy: {test_acc}")
```

#### 4.2.3 โมเดล VGG-16 ที่ Batch Size = 256 และ Learning rate = 0.001

```
adam_opt_001 = tf.keras.optimizers.Adam(learning_rate=0.001)
```

```
modelVGG20.compile(optimizer=adam_opt_001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model with the callbacks
```

```
modelVGG20_256_lr001_history = modelVGG20.fit(train_generator20_BZ256,
epochs=20, validation_data=val_generator20_BZ256)
```

```
# Evaluate the best model on the testing data
```

```
test_loss, test_acc = modelVGG20.evaluate(test_generator20_BZ256,
verbose=1)
```

```
print(f"Test loss: {test_loss}")
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
print(f"Test accuracy: {test_acc}")
```

#### 4.2.4 โมเดล VGG-16 ที่ Batch Size = 256 และ Learning rate = 0.0001

```
adam_opt_0001 = tf.keras.optimizers.Adam(learning_rate=0.0001)
```

```
modelVGG20.compile(optimizer=adam_opt_0001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
```

```
modelVGG20_256_lr0001_history = modelVGG20.fit(train_generator20_BZ256,
epochs=20, validation_data=val_generator20_BZ256)
```

```
# Evaluate model on the testing data
```

```
test_loss, test_acc = modelVGG20.evaluate(test_generator20_BZ256,
verbose=1)
```

```
print(f"Test loss: {test_loss}")
```

```
print(f"Test accuracy: {test_acc}")
```

#### 4.3 การสร้างโมเดล EfficientNet-B0

```
#สร้างโมเดล EfficientNet-B0 ครอบคลุมเวลา20วัน
```

```
base_model = EfficientNetB0(include_top=False, weights="imagenet",
input_shape=(224,224,3))
```

```
for layer in base_model.layers:
```

```
    layer.trainable = False
```

```
x = base_model.output
```

```
x = GlobalAveragePooling2D()(x)
```

```
output = Dense(1, activation='sigmoid')(x)
```

```
modelEF5 = Model(inputs=base_model.input, outputs=output)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.3.1 โมเดล EfficientNet-B0 ที่ Batch Size = 16 และ Learning rate = 0.001

```
adam_opt_001 = tf.keras.optimizers.Adam(learning_rate=0.001)

modelEF20.compile(optimizer=adam_opt_001, loss='binary_crossentropy',
metrics=['accuracy'])

# Train the model

modelEF20_16_lr001_history = modelEF20.fit(train_generator20_BZ16,
epochs=20, validation_data=val_generator20_BZ16)

# Evaluate model on the testing data

test_loss, test_acc = modelEF20.evaluate(test_generator20_BZ16, verbose=1)

print(f"Test loss: {test_loss}")
print(f"Test accuracy: {test_acc}")
```

#### 4.3.2 โมเดล EfficientNet-B0 ที่ Batch Size = 16 และ Learning rate = 0.0001

```
adam_opt_0001 = tf.keras.optimizers.Adam(learning_rate=0.0001)

modelEF20.compile(optimizer=adam_opt_0001, loss='binary_crossentropy',
metrics=['accuracy'])

# Train the model

modelEF20_16_lr0001_history = modelEF20.fit(train_generator20_BZ16,
epochs=20, validation_data=val_generator20_BZ16)

# Evaluate model on the testing data

test_loss, test_acc = modelEF20.evaluate(test_generator20_BZ16, verbose=1)

print(f"Test loss: {test_loss}")
print(f"Test accuracy: {test_acc}")
```

#### 4.3.3 โมเดล EfficientNet-B0 ที่ Batch Size = 256 และ Learning rate = 0.001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น เมื่อผู้ยัดหน้าไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น ผู้พิมพ์ฯ มิได้รับผิดชอบต่อผลเสียหายใดๆ และต้องรับผิดชอบต่อเอกสารทุกครั้งที่มีการนำไปใช้

```
modelEF20.compile(optimizer=adam_opt_001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
```

```
modelEF20_256_lr001_history = modelEF20.fit(train_generator20_BZ256,
epochs=20, validation_data=val_generator20_BZ256)
```

```
# Evaluate model on the testing data
```

```
test_loss, test_acc = modelEF20.evaluate(test_generator20_BZ256,
verbose=1)
```

```
print(f"Test loss: {test_loss}")
```

```
print(f"Test accuracy: {test_acc}")
```

#### 4.3.4 โมเดล EfficientNet-B0 ที่ Batch Size = 256 และ Learning rate = 0.0001

```
adam_opt_0001 = tf.keras.optimizers.Adam(learning_rate=0.0001)
```

```
modelEF20.compile(optimizer=adam_opt_0001, loss='binary_crossentropy',
metrics=['accuracy'])
```

```
# Train the model
```

```
modelEF20_256_lr0001_history = modelEF20.fit(train_generator20_BZ256,
epochs=20, validation_data=val_generator20_BZ256)
```

```
# Evaluate model on the testing data
```

```
test_loss, test_acc = modelEF20.evaluate(test_generator20_BZ256,
verbose=1)
```

```
print(f"Test loss: {test_loss}")
```

```
print(f"Test accuracy: {test_acc}")
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ประวัติผู้วิจัย



ชื่อ-สกุล: นายหริวงศ์ เจริญชาตรี  
ชื่อเล่น: ราม  
วันเดือนปีเกิด: วันที่ 2 เดือนพฤศจิกายน พ.ศ. 2543  
ที่อยู่ปัจจุบัน: 91 ซอยแฮปปี้เพลซ13 ถนนเจ้าคุณทหาร แขวงคลองสามประเวศ เขต  
ลาดกระบัง กรุงเทพฯ 10520  
เบอร์ติดต่อ: 063-9944628  
Email: hariwong.ch@gmail.com

### ประวัติการศึกษา

- ระดับมัธยมศึกษา  
โรงเรียนเทพศิรินทร์ร่วมเกล้า
- ระดับปริญญาตรี  
วิทยาศาสตรบัณฑิต สาขาสถิติประยุกต์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



งานทะเบียนคณะวิทยาศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
คำรับรองเล่มสหกิจศึกษา

วันที่ 7 เดือน กรกฎาคม พ.ศ 2566

ข้าพเจ้า นายหริวงศ์ เจริญชาติ รหัสประจำตัว 62050855  
นักศึกษาหลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชา สถิติประยุกต์ ภาควิชา สถิติ  
ขอรับรองว่าสหกิจศึกษา เรื่อง

ชื่อภาษาไทย การประยุกต์ใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันกับรูปภาพกราฟแท่งเทียน  
เพื่อทำนายการเคลื่อนไหวของราคาข้าวโพด

ชื่อภาษาอังกฤษ USING CONVOLUTIONAL NEURAL NETWORK WITH CANDLESTICK IMAGES TO  
PREDICT CORN PRICE MOVEMENT

ปีการศึกษา 2565

เป็นผลงานวิจัยที่ได้คัดลอกหรือละเมิดลิขสิทธิ์ของผู้อื่นและได้ผ่านการตรวจสอบความซ้ำซ้อนเรียบร้อยแล้ว  
และได้แนบเอกสารการตรวจสอบการลอกเลียนงานวรรณกรรมที่ตรวจสอบจากเล่ม สหกิจศึกษาฉบับสมบูรณ์  
แล้ว

โปรแกรมอักขราวิสุทธิ์ 1.33%

ลงชื่อ หริวงศ์ เจริญชาติ  
(นายหริวงศ์ เจริญชาติ)  
นักศึกษา

ข้าพเจ้า ผศ.ดร.พรพิมล ชัยวุฒิศักดิ์ อาจารย์ที่ปรึกษาสหกิจศึกษา ได้ตรวจสอบสหกิจศึกษาของนักศึกษา  
ข้างต้นแล้ว ขอรับรองว่าเป็นผลงานวิจัยของนักศึกษาจริงและมีเนื้อหาสมบูรณ์ จึงลงชื่อไว้เป็นหลักฐาน

ลงชื่อ พ.ศ.ดร.พรพิมล ชัยวุฒิศักดิ์  
อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้