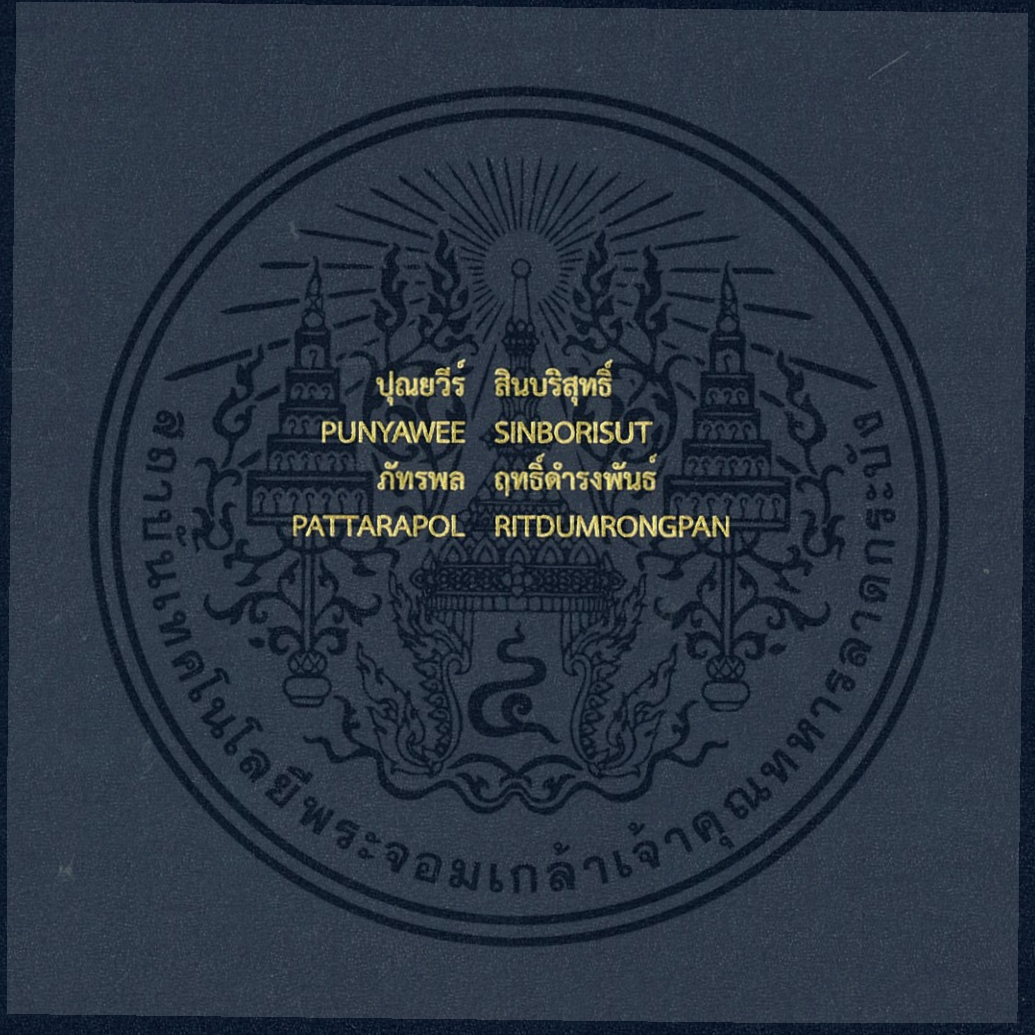
ระบบตรวจสอบและจำแนกพันธุ์กุหลาบ
Verification and Classification Rose Species System



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมสารสนเทศ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560

ระบบตรวจสอบและจำแนกพันธุ์กุหลาบ

Verification and Classification Rose Species System

ปุณยวีร์ สินบริสุทธิ์

PUNYAWEE SINBORISUT

ภัทรพล ฤทธิ์ดำรงพันธ์

PATTARAPOL RITDUMRONGPAN

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิชาวิศวกรรมสารสนเทศ ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Verification and Classification Rose Species System

PUNYAWEE SINBORISUT
PATTARAPOL RITDUMRONGPAN

THESIS IS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INFORMATION ENGINEERING
DEPARTMENT OF COMPUTER ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2017

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญานิพนธ์

หัวข้อปริญญานิพนธ์	ระบบตรวจสอบและจำแนกพันธุ์กุหลาบ
Thesis Title	Verification and Classification Rose Species System
ชื่อนักศึกษา	นายปุณยวีร์ สินบริสุทธิ์ นายภัทรพล ฤทธิ์ดำรงพันธ์
ระดับปริญญา	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมสารสนเทศ
ภาควิชา	วิศวกรรมคอมพิวเตอร์
ปริญญานิพนธ์ปีการศึกษา	2560

(.....)

อ.สรพงษ์ วชิรรัตนพรกุล

อาจารย์ที่ปรึกษาปริญญานิพนธ์

(.....)

ผศ.ไพศาล สิทธิโยภาสกุล

อาจารย์ที่ปรึกษาปริญญานิพนธ์ร่วม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ ระบบตรวจสอบและจำแนกพันธุ์กุหลาบ

Thesis Title Verification and Classification Rose Species System

ชื่อนักศึกษา นายปุณยวีร์ สินบริสุทธิ์ รหัสนักศึกษา 57010795

นายภัทรพล ฤทธิ์ดำรงพันธ์ รหัสนักศึกษา 57010961

ระดับปริญญา วิศวกรรมศาสตรบัณฑิต

สาขาวิชา วิศวกรรมสารสนเทศ

ภาควิชา วิศวกรรมคอมพิวเตอร์

ปีการศึกษา 2560

อาจารย์ที่ปรึกษาปริญญานิพนธ์ อ.สรพงษ์ วชิรรัตนพรกุล

อาจารย์ที่ปรึกษาปริญญานิพนธ์ร่วม ผศ.ไพศาล สิทธิโยภาสกุล

บทคัดย่อ

ปริญญานิพนธ์นี้ได้ถูกจัดทำขึ้นโดยมีจุดประสงค์เพื่อให้ความรู้แก่ผู้ส่งออกกุหลาบรวมถึงผู้ที่สนใจ เนื่องจากผู้ส่งออกไม่มีความรู้ทางด้านสายพันธุ์กุหลาบ ซึ่งก่อให้เกิดการกดราคาของกุหลาบในแต่ละสายพันธุ์ จึงได้ทำการสร้างระบบตรวจสอบและจำแนกพันธุ์กุหลาบขึ้นมา เพื่อช่วยเหลือผู้ส่งออกให้สามารถแยกพันธุ์กุหลาบ และส่งออกในราคาที่เหมาะสมได้ ทำให้มีรายได้เข้าสู่ประเทศเพิ่มขึ้น รวมถึงให้ความรู้แก่ผู้สนใจสายพันธุ์กุหลาบ โดยหลักการทำงานของระบบจะเป็นการรับรูปภาพมาวิเคราะห์และเปรียบเทียบกับโมเดลที่เกิดจากการนำรูปกุหลาบหลายๆ พันธุ์มาวิเคราะห์และจำแนกพันธุ์ที่มีลักษณะใกล้เคียงที่สุด จากนั้นจึงแสดงผลออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Verification and Classification Rose Species System
Student	Mr. Punyawee Sinborisut Student ID. 57010795
	Mr. Pattarapol Ritdumrongpan Student ID. 57010961
Degree	Bachelor of Engineering
Program	Information Engineering
Department	Computer Engineering
Academic Year	2017
Thesis Advisor	Mr. Sorapong Wachirarattanapornkul
Thesis Co-Advisor	Asst.Prof. Paisan Sithiyopasakul

ABSTRACT

This thesis aims to give information for Thai rose-exporter, included who interested in rose species. Surprisingly, most of the rose exporter has lack of knowledge toward the diversity of rose species, and that causes the undersell and unfair trading. Therefore, we developed this system to help them classify the rose species and export with reasonable price which can increase the income to our country along with giving the information to those who interest in rose species. Operation of this system is it will get the image then analyze and compare it with a model that is created from analyzing many rose images and classify it then show the result

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์เล่มนี้สำเร็จได้ด้วยความกรุณาจากอาจารย์ที่ปรึกษา อ.สรพงษ์ วชิรรัตนพรกุล และ ผศ.ไพศาล สิทธิโยภาสกุล ที่ให้ความช่วยเหลือ ให้คำชี้แนะช่วยแก้ปัญหาตลอดจนให้ความรู้และประการณ์ที่ดีแก่ผู้จัดทำ ทางผู้จัดทำขอขอบพระคุณ

ปุณยวีร์ สินบริสุทธิ์
ภัทรพล ฤทธิ์ดำรงพันธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VII

สารบัญรูป VIII

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญ 1

1.2 ภาพรวม หรือโครงสร้างรวมของปริญญานิพนธ์ 1

1.3 วัตถุประสงค์ของปริญญานิพนธ์ 1

1.4 ขอบเขตของปริญญานิพนธ์ 1

1.5 ขั้นตอนการดำเนินงานปริญญานิพนธ์ 2

1.6 ตารางเวลาดำเนินงาน 2

บทที่ 2 ทฤษฎีพื้นฐานที่ใช้ 3

2.1 TensorFlow 3

2.2 Visual studio 3

2.3 ImageNet 3

2.4 Pretrained Models 3

2.4.1 Inception V3 4

2.4.2 MobileNets 4

2.5 TensorBoard 5

2.6 รูปภาพดิจิตอล (Digital Image) 5

2.6.1 ลักษณะของภาพดิจิตอล 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.6.2 ประเภทของภาพดิจิตอล 6

2.6.3 ประเภทของภาพบิตแมป 6

2.7 ไฟร์เบส (Firebase) 7

2.7.1 Firebase Realtime Database 7

2.7.2 Firebase Authentication 7

2.8 Programming Languages 7

2.8.1 C# Programming Language 7

2.8.2 Python Programming Language 7

2.9 Machine Learning 7

2.9.1 วิธีการเรียนรู้ 8

2.9.2 ประเภท 10

2.10 โครงข่ายประสาทเทียม (Artificial neural network) 11

2.10.1 หลักการ 12

2.10.2 การทำงาน 12

2.10.3 ส่วนประกอบของโครงข่ายประสาทเทียม 13

2.10.4 ลักษณะของโครงข่ายประสาทเทียม 14

2.10.5 ประเภทของการเรียนรู้ของโครงข่ายประสาทเทียม 15

2.10.6 การเรียนรู้แบบแพร่ย้อนกลับ 16

2.11 Image Retraining 18

2.11.1 กระบวนการคอขวด (Bottlenecks) 18

2.11.2 การเทรนโมเดล (Training) 19

บทที่ 3 การวิเคราะห์และออกแบบระบบ 20

3.1 รูปภาพที่จะนำมาประมวลผล (Input Image) 21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

3.2 รูปภาพที่นำมาเทรน....21

3.3 ขั้นตอนการเทรน....22

3.4 การทดสอบโมเดล....23

3.5 การแสดงผล....23

บทที่ 4 การทดลองและผลการทดลอง....24

4.1 เริ่มต้นใช้งานโปรแกรม....24

4.2 หน้าแสดงผลลัพธ์ของโปรแกรม....24

4.3 หน้าข้อมูลของกุหลาบ....25

4.4 หน้าสำหรับผู้พัฒนา....26

บทที่ 5 บทสรุปและข้อเสนอแนะ....27

5.1 สรุปผลการดำเนินงาน....27

5.2 ปัญหาที่เกิดขึ้นและแนวทางแก้ไข....27

5.3 แนวทางการพัฒนาต่อ....27

บรรณานุกรม....28

ภาคผนวก....29

ภาคผนวก ก Poster (และรูปผลงานถ้ามี)....30

ภาคผนวก ข การติดตั้ง Visual Studio....34

ภาคผนวก ค การติดตั้ง Python....37

ภาคผนวก ง การติดตั้ง Tensorflow....40

ภาคผนวก จ การติดตั้งโปรแกรมและใช้งาน....44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 1.1 ตารางขั้นตอนการดำเนินงาน.....2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 1.1 รูปขั้นตอนการทำงาน 1
รูปที่ 2.1 โครงสร้างการทำงานของโครงข่ายประสาทเทียม 13
รูปที่ 2.2 โครงสร้างโครงข่ายประสาทเทียมแบบชั้นเดียว 14
รูปที่ 2.3 โครงสร้างโครงข่ายประสาทเทียมแบบหลายชั้น 15
รูปที่ 2.4 ซิกมอยด์ฟังก์ชัน 17
รูปที่ 2.5 ภาพรวมของ Forward propagation และ Backward propagation 18
รูปที่ 3.1 แผนภาพบล็อก(block diagram) แสดงการตรวจสอบและจำแนกสายพันธุ์กุหลาบ 20
รูปที่ 3.2 รูปที่ต้องการมาจำแนกพันธุ์ 21
รูปที่ 3.3 รูปภาพที่ได้จากการถ่าย 21
รูปที่ 3.4 ขั้นตอนการ train model 22
รูปที่ 3.5 ขั้นตอนการจำแนกสายพันธุ์ 22
รูปที่ 3.6 ผลลัพธ์ที่ได้ 23
รูปที่ 4.1 หน้าหลักของโปรแกรม 24
รูปที่ 4.2 หน้าแสดงผลลัพธ์ 25
รูปที่ 4.3 หน้าแสดงสายพันธุ์กุหลาบทั้งหมดในฐานข้อมูล 26
รูปที่ 4.4 หน้าสำหรับ train model 26
รูปที่ ก.1 Poster Verification and Classification Rose Species System 31
รูปที่ ก.2 หน้าหลักของโปรแกรม 32
รูปที่ ก.3 หน้าแสดงผลลัพธ์ 32
รูปที่ ก.4 หน้าแสดงสายพันธุ์กุหลาบทั้งหมดในฐานข้อมูล 33
รูปที่ ก.5 หน้าสำหรับ train model 33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

หน้า

รูปที่ ข.1 เว็บไซต์ดาวน์โหลด Visual Studio 35
รูปที่ ข.2 หน้าต่างแสดงขนาดของตัวติดตั้ง Visual Studio Community 36
รูปที่ ค.1 หน้าเว็บไซต์แสดงขั้นตอนการดาวน์โหลด Python 38
รูปที่ ค.2 หน้าต่างแสดงขนาดของตัวติดตั้ง Python 39
รูปที่ ง.1 ตำแหน่งของ Advanced system settings 41
รูปที่ ง.2 หน้า System Properties 42
รูปที่ ง.3 หน้าต่างเลือก Path 43
รูปที่ ง.4 Path ที่ทำการลง Python 43
รูปที่ จ.1 หน้าติดตั้ง VNCRS 45
รูปที่ จ.2 รูปหน้าต่างตัวติดตั้งโปรแกรม 45
รูปที่ จ.3 รูปหน้าเลือกโฟลเดอร์ที่ต้องการจะติดตั้ง 46
รูปที่ จ.4 รูปหน้ายืนยันการติดตั้ง 46
รูปที่ จ.5 รูปหน้ารอการติดตั้ง 47
รูปที่ จ.6 หน้าแสดงการติดตั้งสำเร็จ 47
รูปที่ จ.7 หน้าแสดงตัวโปรแกรม 48
รูปที่ จ.8 หน้าโปรแกรมวิเคราะห์ภาพ 48
รูปที่ จ.9 หน้าแสดงผลลัพธ์ 49
รูปที่ จ.10 หน้าค้นหาข้อมูล 49
รูปที่ จ.11 หน้าเทรนโมเดล 50
รูปที่ จ.12 หน้าแสดงผลลัพธ์ของการเทรนโมเดล 50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญ

โครงการนี้เกิดขึ้นเนื่องมาจากประเทศไทยเกิดปัญหาด้านการส่งออกกุหลาบ เพราะผู้ส่งออกไม่มีความรู้ทางด้านสายพันธุ์ของกุหลาบ ซึ่งก่อให้เกิดการกดราคาของกุหลาบในแต่ละสายพันธุ์ พวกเราจึงได้ทำการสร้างระบบตรวจสอบและจำแนกพันธุ์กุหลาบขึ้นมาเพื่อช่วยเหลือผู้ส่งออกให้สามารถแยกพันธุ์และส่งออกในราคาที่เหมาะสมได้ ทำให้เกิดการสร้างรายได้เข้าสู่ประเทศเพิ่มขึ้น รวมถึงให้ความรู้แก่ผู้สนใจเกี่ยวกับสายพันธุ์ของกุหลาบ

1.2 ภาพรวม หรือโครงสร้างรวมของปริญญานิพนธ์

ปริญญานิพนธ์นี้ประกอบไปด้วย Software ที่เอาไว้จำแนกพันธุ์ของกุหลาบ และรูปของกุหลาบแต่ละสายพันธุ์สำหรับใช้เปรียบเทียบและวิเคราะห์สายพันธุ์ของกุหลาบโดยดูความแตกต่างของดอกในแต่ละสายพันธุ์ แล้วสามารถนำมาแสดงผลผ่านทางหน้าจอแสดงผลได้

รูปที่ 1.1 รูปขั้นตอนการทำงาน

1.3 วัตถุประสงค์ของปริญญานิพนธ์

เพื่อเพิ่มรายได้ของการส่งออกพันธุ์กุหลาบให้กับประเทศและสามารถใช้เป็นอุปกรณ์อ้างอิงในการเปรียบเทียบสายพันธุ์กุหลาบได้ รวมถึงให้ความรู้แก่ผู้ส่งออก

1.4 ขอบเขตของปริญญานิพนธ์

1. สามารถระบุสายพันธุ์ของกุหลาบได้แม่นยำที่สุด
2. สามารถติดตั้งและใช้งานโปรแกรมบนคอมพิวเตอร์เครื่องอื่นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ขั้นตอนการดำเนินงานปริญญานิพนธ์

1. ศึกษาข้อมูลเกี่ยวกับการจำแนกพันธุ์กุหลาบ
2. ศึกษาเกี่ยวกับเทคโนโลยีในการเขียนโปรแกรม เช่น Visual studio, TensorFlow
3. ทำการเขียนโปรแกรมต้นแบบ
4. ทดสอบโปรแกรมที่เขียนขึ้น
5. แก้ไขข้อผิดพลาดของโปรแกรม
6. นำโปรแกรมไปใช้งานจริง
7. ทำปริญญานิพนธ์

1. แผนผัง หรือตารางเวลาการดำเนินงานโครงงาน

ตารางที่ 1.1 ตารางขั้นตอนการดำเนินงาน

ลำดับ	ขั้นตอนการทำงาน	ระยะเวลา									
		สิงหาคม	กันยายน	ตุลาคม	พฤศจิกายน	ธันวาคม	มกราคม	กุมภาพันธ์	มีนาคม	เมษายน	พฤษภาคม
1.	ศึกษาหาข้อมูลเกี่ยวกับปัญหาการส่งออกพันธุ์กุหลาบ	←	→								
2.	วิเคราะห์ปัญหา หาแนวทางการแก้ไขปัญหา		←	→							
3.	ออกแบบรูปแบบการทำงานที่เหมาะสม		←	→							
4.	ค้นคว้าข้อมูลเกี่ยวกับ C#, TensorFlow และฐานข้อมูลที่จะใช้ร่วมกัน			←	→						
5.	พัฒนาโปรแกรมเพื่อใช้งานกับต้นแบบช่วงที่ 1			←	→						
6.	นำเสนอผลงานในส่วนที่ 1				←	→					
7.	พัฒนาโปรแกรมเพื่อใช้งานช่วงที่2					←	→				
8.	ทดสอบการทำงานผ่านต้นแบบ และแก้ไขปัญหาที่เกิดขึ้นจากการทดสอบการใช้งาน							←	→		
9.	นำเสนอผลงานในส่วนที่ 2									←	→
10.	การจัดทำปริญญานิพนธ์							←	→		→

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีพื้นฐานที่ใช้

2.1 TensorFlow

TensorFlow คือ โอเพนซอร์สสำหรับการคำนวณเชิงตัวเลขโดยการใช้กราฟการไหลของข้อมูล โดยที่โหนด (Node) ของกราฟแทนตัวดำเนินการทางคณิตศาสตร์ และขอบของกราฟแทนข้อมูลอาร์เรย์หลายมิติ (Tensors) เนื่องจากโครงสร้างที่มีความยืดหยุ่นทำให้สามารถประมวลผลโดยใช้ CPUs หรือ GPUs บนเดสก์ทอป (Desktop) เซิร์ฟเวอร์ (Server) หรือ อุปกรณ์เคลื่อนที่ (Mobile device)

TensorFlow ในตอนแรกถูกพัฒนาโดยนักวิจัยและวิศวกรที่ทำงานอยู่ที่ Google Brain Team ซึ่งเป็นทีมที่ทำหน้าที่วิจัยเกี่ยวกับปัญญาประดิษฐ์ให้กับองค์กร โดยมีจุดประสงค์เพื่อปฏิบัติงานและวิจัยเกี่ยวกับ Machine Learning และ Deep Neural Networks และระบบสามารถนำไปประยุกต์ใช้กับเรื่องต่างๆ ได้เป็นอย่างดี

2.2 Visual studio

Visual Studio คือ Integrated Development Environment พัฒนาขึ้นโดยไมโครซอฟท์ (Microsoft) ซึ่งเป็นเครื่องมือช่วยนักพัฒนาซอฟต์แวร์พัฒนาโปรแกรม เช่น เว็บไซต์ โปรแกรม เว็บแอปพลิเคชั่น ระบบที่รองรับการทำงานนั้นมี ไมโครซอฟท์ วินโดวส์ สมาร์ทโฟน และเว็บเบราว์เซอร์ (Web browser) ในปัจจุบัน Visual Studio นั้นสามารถใช้ภาษาโปรแกรมที่เป็นภาษาดอตเน็ต (.NET Language) ในโปรแกรมเช่นเดียวกับ VB.NET C++ C# เป็นต้น

2.3 ImageNet

ImageNet คือ ข้อมูลรูปภาพที่ได้รับการรวบรวมตามขั้นตอนของ WordNet โดยรูปใน 1 โหนด จะประกอบไปด้วยรูปจำนวน 100 ถึง 1000 รูป ซึ่งปัจจุบันมีค่าเฉลี่ยมากกว่า 600 รูป ต่อ 1 โหนด โดยที่รูปในแต่ละโหนด จะมีลักษณะที่มีความใกล้เคียงกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 Pretrained Models

เป็นการนำโมเดล ซึ่งเป็น Convolution Neural Networks (CNN) มาทำการ Train กับ ILSVRC-2012-CLS ซึ่งเป็นข้อมูลไว้สำหรับการจำแนกรูปภาพ โดยด้านล่างจะเป็นรายชื่อของโมเดล ที่เกี่ยวข้องกับ TensorFlow

Inception V1 Inception V2 Inception V3 Inception V4 Inception-ResNet-v2

ResNet V1 50 ResNet V1 101 ResNet V1 152 ResNet V2 50 ResNet V2 101

ResNet V2 152 ResNet V2 200

VGG 16 VGG 19

MobileNet_v1_1.0_224 MobileNet_v1_0.50_160 MobileNet_v1_0.25_128

NASNet-A_Mobile_224# NASNet-A_Large_331#

2.4.1 Inception V3

เป็นโมเดลที่ถูกเทรน โดยการใช้ ImageNet เป็นโมเดลที่ทำหน้าที่แยกประเภทของรูปภาพ เช่น ม้าลาย เครื่องล้างจาน เป็นต้น

2.4.2 MobileNets

เป็นโมเดลที่สามารถนำมาใช้ทำอะไรได้หลายอย่าง เช่น การจำแนกสิ่งของต่างๆ รวมถึงการตรวจจับวัตถุที่อยู่ในรูปภาพ ซึ่งให้ความแม่นยำที่สูงมาก นอกจากนี้ยังสามารถใช้งานบนอุปกรณ์เคลื่อนที่ได้อย่างมีประสิธิภาพโดยใช้ TensorFlow Mobile

ด้านล่างนี้เป็นรายชื่อโมเดลของ MobileNet ที่ได้ถูกเทรนโดยใช้ ข้อมูลรูปภาพ ILSVRC-2012-CLS

MobileNet_v1_1.0_224	MobileNet_v1_1.0_192	MobileNet_v1_1.0_160
MobileNet_v1_1.0_128	MobileNet_v1_0.75_224	MobileNet_v1_0.75_192
MobileNet_v1_0.75_160	MobileNet_v1_0.75_128	MobileNet_v1_0.50_224
MobileNet_v1_0.50_192	MobileNet_v1_0.50_160	MobileNet_v1_0.50_128

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MobileNet_v1_0.25_224 MobileNet_v1_0.25_192 MobileNet_v1_0.25_160

MobileNet_v1_0.25_128 MobileNet_v1_1.0_224_quant569

MobileNet_v1_1.0_192_quant418 MobileNet_v1_1.0_160_quant291

MobileNet_v1_1.0_128_quant186 MobileNet_v1_0.75_224_quant

MobileNet_v1_0.75_192_quant MobileNet_v1_0.75_160_quant

MobileNet_v1_0.75_128_quant MobileNet_v1_0.50_224_quant

MobileNet_v1_0.50_192_quant MobileNet_v1_0.50_160_quant

MobileNet_v1_0.50_128_quant MobileNet_v1_0.25_224_quant

MobileNet_v1_0.25_192_quant MobileNet_v1_0.25_160_quant

MobileNet_v1_0.25_128_quant

2.5 TensorBoard

TensorBoard เป็นเว็บแอปพลิเคชัน (Web applications) สำหรับตรวจสอบและทำความเข้าใจการทำงานและกราฟที่ได้จากผลลัพธ์จากการทำงานของ TensorFlow TensorBoard ทำงานโดยการอ่านไฟล์ที่บรรจุข้อมูลสรุป ซึ่งถูกสร้างขึ้นในขณะที่ TensorFlow ทำงานอยู่

2.6 รูปภาพดิจิตอล (Digital Image)

2.6.1 ลักษณะของภาพดิจิตอล

ภาพดิจิตอล เป็นการแสดงผลภาพในรูปแบบ 2 มิติจะอยู่ในรูปของแผ่นตารางในแต่ละช่องจะเป็นส่วนหนึ่งของภาพ โดยมีจะเรียกช่องเหล่านั้นว่า พิกเซล (Pixel) และในแต่ละพิกเซลจะถูกทำหนดความเข้มของสีและความสว่างซึ่งแสดงในรูปของเลขฐานสอง (Binary numeral system) (0หรือ1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.2 ประเภทของภาพดิจิตอล

ประเภทของภาพดิจิตอล แบ่งออกเป็น2ประเภท คือ ภาพแบบบิตแมป (Bitmap Image) และ ภาพแบบเวกเตอร์ (Vector Image)

1. ภาพแบบบิตแมป

บิตแมปเป็นภาพที่เกิดขึ้นจากการนำเอาจุดสีเล็กๆหลายๆจุดมารวมต่อกันจนทำให้เกิดเป็นภาพซึ่งมาความกว้างยาว X พิกเซล (Pixel) และ y พิกเซล และมีความลึก (color depth) คือ Z พิกเซล เนื่องจากจุดภายในบิตแมป มีความคงที่ หากทำการขยายภาพจุดภายในตารางก็จะมีขนาดใหญ่ขึ้น ทำให้ภาพบิตแมปเมื่อทำการขยายภาพมีความไม่ละเอียด ภาพจึงแตกออก

2. ภาพแบบเวกเตอร์

ภาพแบบเวกเตอร์เป็นไฟล์กราฟฟิคที่เกิดจากการคำนวณทางคณิตศาสตร์ ทั้งค่าระนาบสองมิติ รวมทั้งมุมและระยะทาง ทำให้ก่อเกิดเป็นเส้น หรือรูปภาพ ข้อดีของภาพแบบเวกเตอร์คือสามารถทำการย่อขยายได้อย่างอิสระโดยที่ภาพไม่มีความผิดเพี้ยนไปจากเดิม ข้อเสียคือได้เป็นภาพที่คล้ายกับภาพจริง หรือเป็นได้แค่ภาพเสมือนจริงเท่านั้น

2.6.3 ประเภทของภาพบิตแมป

1. ภาพแบบระดับสีเทา (Gray scale image)

ภาพแบบระดับสีเทามีเพียงสองสีคือสีขาวและสีดำ ภาพแบบระดับสีเทามีความละเอียดเท่ากับ 8 บิต โดยที่มีความเข้มของสีอยู่ที่ 0-255 โดยที่ระดับความเข้มของสีดำคือ 255 และสีขาวคือ 0

2. ภาพสี (Color image)

ภาพสี ใน 1 พิกเซล ของภาพสีจะมีสีอยู่ 3 สี คือ สีแดง (Red) สีเขียว (Green) และสีน้ำเงิน (Blue) โดยเก็บเอาไว้ในรูปแบบของอาร์เรย์

3. ภาพขาวดำ (Binary image)

ภาพขาวดำ เป็นภาพที่ในแต่ละพิกเซลจะแสดง สีขาวหรือไม่ก็สีดำเท่านั้น โดยประกอบไปด้วยค่า1และ0 โดยที่ 1 แสดงเป็นจุดสีขาว และ 0 แสดงเป็นจุดสีดำภายในภาพ

4. ภาพแบบดัชนี (Index image)

เป็นรูปที่มีการเก็บแบบ indexed คือจะเก็บค่าสีเป็น indexed และในแต่ละช่องอาร์เรย์ จะเก็บค่าสีนั้นๆ ไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 ไฟร์เบส(Firebase)

2.7.1 Firebase Realtime Database

เป็นฐานข้อมูลที่เก็บข้อมูลแบบ NoSQL โดยจะเชื่อมโยงข้อมูลกับผู้ใช้แบบเวลาจริง (Realtime) และสามารถทำงานได้แม้โปรแกรมไม่ได้เชื่อมต่ออินเตอร์เน็ต (Internet) โดยจะเก็บข้อมูลในรูปแบบ JSON

2.7.2 Firebase Authentication

เป็นระบบยืนยันตัวตนของ Firebase ใช้สำหรับทำระบบ login ทำให้สามารถใช้งานข้ามแพลทฟอร์ม (Platform) ได้

2.8 Programming Languages

2.8.1 C# Programming Language

C# เป็นภาษาที่ถูกพัฒนาโดยบริษัทไมโครซอฟท์ เพื่อทำงานบนดอตเน็ตเฟรมเวิร์ก (.Net Framework) และมีรากฐานมาจากภาษาซีพลัสพลัส (C++ Language) และภาษาอื่นๆ เช่น ภาษาจาวา (Java Language) โดยมีเป้าหมายให้เป็นภาษาที่ไม่ซับซ้อน ใช้งานได้ทั่วไป

2.8.2 Python Programming Language

ไพธอน(Python) เป็นภาษาโปรแกรมขั้นสูงโดยที่ code ถูกออกแบบมาให้อ่านได้ง่าย และด้วยความที่ไพธอนเป็นภาษาสคริปต์ ทำให้ใช้เวลาในการเขียนและรันไม่มาก จึงเหมาะกับงานด้านการดูและระบบ (System administration) นอกจากนี้ยังได้รับการสนับสนุนให้เป็นส่วนหนึ่งของระบบปฏิบัติการยูนิกซ์ (UNIX) ลินุกซ์ (Linux) นอกจากนี้สามารถเรียกใช้ภาษาอื่นๆ ได้ ทำให้เหมาะที่จะเขียนเพื่อประสานงานโปรแกรมในภาษาที่ต่างกัน

2.9 Machine Learning

Machine Learning คือ การศึกษาและการสร้างอัลกอริทึม (Algorithm) ที่สามารถเรียนรู้ข้อมูลและสามารถทำนาย (Predict) ข้อมูลได้ ซึ่งเป็นการพัฒนามาจากการรู้จำแบบ (Pattern Recognition) โดยอัลกอริทึมจะทำงานโดยอาศัยโมเดล ที่สร้างมาจากชุดข้อมูลตัวอย่างขาเข้า (Input) เพื่อทำนายในภายหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Machine Learning มีความเกี่ยวข้องกับสถิติศาสตร์ เนื่องจากมีการวิเคราะห์ข้อมูลเพื่อการทำนาย เหมือนกัน Machine Learning สามารถนำไปประยุกต์ใช้ได้หลายแบบ เช่น การกรองอีเมลล์ขยะ เครื่องมือค้นหา เป็นต้น

2.9.1 วิธีการเรียนรู้

2.9.1.1 การเรียนรู้ต้นไม้ตัดสินใจ (Decision tree learning)

เป็น Model ทางคณิตศาสตร์ ที่ใช้ทำนายประเภทของวัตถุโดยพิจารณาจากลักษณะของวัตถุ โดยอัลกอริทึมที่ใช้มีชื่อว่า ID3 โดยจะเป็นการสร้าง Decision tree จากบนลงล่างโดยจะทำการถามว่าตัวแปรต้นใด เป็นรากของ Decision tree นี้ โดยจะถามซ้ำไปเรื่อยๆ โดยจะเลือกตัวแปรที่ดีที่สุดจาก Information gain ซึ่งก่อนจะได้ Information gain เราต้องหา เอนโทรปี (Entropy) ให้ได้ก่อน โดยนิยามของ เอนโทรปี (Entropy) คือ การหาค่าความไม่แน่นอนของข้อมูลใน set นั้นๆ

$$H(S) = \sum_{x \in X} -p(x)\, log_2\, p(x) \tag{2.1}$$

โดยที่

S คือ set ของข้อมูลปัจจุบันที่ต้องการจะคำนวณ

X คือ set ของ classes ใน S

$p(x)$ คือ อัตราส่วนของสมาชิกใน class x ต่อ สมาชิกใน set S

Information gain คือ การหาว่ามีค่าความไม่แน่นอนลดลงไปเท่าไรหลังจากการ split set S โดยใช้ลักษณะของ A หรือตัวแปรต้น

$$IG(A,S) = H(S) - \sum_{t \in T} p(t)H(t) \tag{2.2}$$

โดยที่

$H(S)$ คือ Entropy ของ set S

T คือ subset จากการแยก set S โดยใช้ลักษณะ A

$p(t)$ คือ อัตราส่วนของสมาชิกใน t ต่อ สมาชิกใน set S

$H(t)$ คือ Entropy ของ subset t

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9.1.2 กฎความสัมพันธ์ (Association rule learning)

เป็นกระบวนการหนึ่งในการทำ Data Mining โดยจะหาความสัมพันธ์ของข้อมูลสองชุดดหรือ มากกว่าสองชุดขึ้นไปภายในกลุ่มข้อมูลที่มีขนาดใหญ่

2.9.1.3 โครงข่ายประสาทเทียม (Artificial neural networks)

เป็นอัลกอริทึมที่ได้รับแรงบันดาลใจมาจากโครงสร้างและการทำงานของเซลล์ประสาทในสมอง

2.9.1.4 การโปรแกรมตรรกะเชิงอุปนัย (Inductive logic programming)

เป็นการนำ Machine learning มาใช้ร่วมกับ Logic programming โดยมีแนวคิดหลักคือ พยายามสร้างทฤษฎีที่ครอบคลุมตัวอย่างบวกแต่ไม่ครอบคลุมตัวอย่างลบ

2.9.1.5 ซัพพอร์ตเวกเตอร์แมชชีน (Support vector machines)

ใช้เพื่อการแบ่งประเภทของข้อมูล โดยข้อมูลจะถูกจัดอยู่ในประเภทใดประเภทหนึ่งจากสองประเภท โดยจะสร้างแบบจำลองว่าจะอยู่ประเภทใด

2.9.1.6 การแบ่งกลุ่มข้อมูล (Clustering)

เป็นการวิเคราะห์ข้อมูล โดยจะแบ่งชุดข้อมูลออกเป็นกลุ่มและนำข้อมูลที่มีลักษณะเหมือนกันหรือคล้ายกันจัดไว้ในกลุ่มเดียวกันโดยจะใช้ความเหมือน (Similarity) หรือ ความใกล้ชิด (Proximity) โดยคำนวณจากการวัดระยะแบบต่างๆ เช่น การวัดระยะแบบยูคลิด (Euclidean distance) การวัดระยะแบบแมนฮัตตัน (Manhattan distance) เป็นต้น

2.9.1.7 เครือข่ายแบบเบย์ (Bayesian networks)

เป็นโมเดลความน่าจะเป็นเชิงกราฟที่แทนกลุ่มของตัวแปรสุ่ม

2.9.1.8 การเรียนรู้แบบเสริมกำลัง (Reinforcement learning)

เป็นการพิจารณาว่า A ควรจะกระทำสิ่งใด ภายใต้สภาพแวดล้อม เพื่อที่จะได้รางวัล

2.9.1.9 การเรียนรู้ด้วยการแทน (Representation learning)

เป็นการค้นหาการแทนข้อมูลขาเข้าที่ดีขึ้นเมื่อมีชุดข้อมูลฝึก

2.9.1.10 การเรียนรู้ด้วยความคล้าย (Similarity and metric learning)

เป็นการมองหาความคล้ายของตัวอย่างในแต่ละคู่ที่มีความคล้ายมากและมีความคล้ายน้อย โดยสามารถบอกได้ว่ามีความคล้ายมากน้อยเพียงใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9.1.11 ขั้นตอนวิธีเชิงพันธุกรรม (Genetic algorithms)

เป็นการค้นหาโดยใช้เทคนิคการกลายพันธุ์ของยีนและการเปลี่ยนของโครโมโซมในการหาประชากรที่น่าจะอยู่รอด

2.9.2 ประเภท

2.9.2.1 การเรียนรู้แบบมีผู้สอน (Supervised learning)

เป็นการเรียนรู้ของ Machine learning โดยมี Data มาสอน เช่น การแยกประเภทของผลไม้โดยเรานำข้อมูลของผลไม้แต่ละชนิดมา เช่น สี ลักษณะ จากนั้นแปลงเป็นตัวเลขโดยเราเรียกว่า Features และบอกชื่อของผลไม้ชนิดนั้นโดยให้เป็นข้อมูลตัวเลข เรียกว่า Labels ซึ่งถ้าเกิดว่าโปรแกรมเจอข้อมูลดังกล่าวตรงตามข้อมูลหรือใกล้เคียงที่มีอยู่ก็จะระบุออกมาได้เลยว่าเป็นผลไม้ชนิดใด

Support vector machine (SVM) เป็นโมเดลที่ใช้ในการคำนวณทางคณิตศาสตร์ซึ่งจัดอยู่ในกลุ่มเดียวกับ Neural Network แต่มีข้อดีตรงที่จะไม่ค่อยเกิดปัญหา Overfitting มากเหมือนกับ Neural Network โดยหลักการของ SVM จะทำการหาค่าโดยใช้ linear model จากนั้นดูค่าที่ไม่เอนเอียงไปทางกลุ่มใดกลุ่มหนึ่งมากจนเกินแล้วเลือกใช้ linear model นั้น โดย SVM นั้นเหมาะสำหรับจำแนกข้อมูลที่มี 2 คลาสหรือมากกว่า รวมถึงข้อมูลประเภท Multi label หรือการที่ข้อมูล 1 ตัวมีคำตอบได้มากกว่า 1 คลาส และ เหมาะกับข้อมูลประเภท Hierarchical Multi-label หรือก็คือข้อมูลที่มีหลาย label และมีการเรียงตัวเป็นชั้นๆ

Overfitting คือ การที่โมเดลจดจำรูปแบบของข้อมูลการเทรนมากเกินไป จนไม่สามารถทำนายข้อมูลที่ไม่เคยพบได้ ตัวอย่างเช่น นักศึกษาคนนึงเตรียมตัวสอบโดยการนำข้อสอบปีก่อนหน้ามาจำแต่พอถึงตอนสอบกลับไม่มีคำถามในปีก่อนๆ จึงทำให้ได้คำแนนต่ำ

Naïve Bayes เป็นเทคนิคการทำ Classification โดยจะใช้การคำนวณความน่าจะเป็นโดย Bayes Theorem เป็นไปตามสมการต่อไปนี้

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)} \tag{2.3}$$

โดยที่

$P(B|A)$ หรือ Posterior probability คือ ความน่าจะเป็นที่ข้อมูลที่มีแอตทริบิวต์ (Attribute) เป็น A จะมีคลาส B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$P(A|B)$ หรือ Likelihood คือ ความน่าจะเป็นที่ข้อมูล training data มีคลาส B และมี attribute A

$P(B)$ หรือ Prior probability คือความน่าจะเป็นของคลาส B

Random Forest เป็นโมเดลสำหรับ classification โดยใช้การสุ่มข้อมูลที่ได้จากการเทรน และ ลักษณะเด่น(Feature) ออกเป็นหลายๆชุด และสร้างโมเดลโดยใช้ Decision tree หลายๆต้น

K-Nearest Neighbor เป็นอัลกอริทึมที่ใช้ในการจัดกลุ่มข้อมูลที่อยู่ใกล้กัน โดยที่ K หมายถึงจำนวนข้อมูลที่นำมาพิจารณา เช่นถ้า K=2 จะพิจารณาข้อมูล 2 จุดที่ใกล้เคียงจุดที่ต้องการจะพิจารณา โดยเราควรจะกำหนดให้ K เป็นเลขคี่เสมอ โดย K-Nearest Neighbor มีการทำงานดังนี้

1. กำหนดขนาดของ K โดยเราจะกำหนดให้เป็นเลขคี่
2. คำนวณหา Distance ของข้อมูลที่ต้องการจะพิจารณากับข้อมูลตัวอย่าง
3. จัดเรียงลำดับของ Distance และพิจารณาข้อมูลที่ใกล้จุดที่ต้องการพิจารณา
4. สังเกตว่า Class ไหนใกล้จุดที่พิจารณามากที่สุด
5. กำหนด Class ให้กับจุดที่ใกล้มากที่สุด

2.9.2.2 การเรียนรู้แบบไม่มีผู้สอน (Unsupervised learning)

Unsupervised learning จะตรงข้ามกับ Supervised learning โดยที่จะไม่มี Data มาสอน หรือก็คือจะทำหารแยกชนิดของผลไม้โดยอาจจะแยกตามสี ส่วนสูง ก็ได้ โดย Model ที่ใช้ทำการแยกก็คือ K-Means

K-Means เป็นวิธีสำหรับการทำ clustering โดยจะแบ่งข้อมูลออกเป็น K กลุ่ม โดยแทนแต่ละกลุ่มด้วยค่าเฉลี่ยของแต่ละกลุ่มซึ่งใช้เป็นจุดศูนย์กลางในการวัดระยะห่างของข้อมูลในกลุ่มเดียวกัน

2.9.2.3 การเรียนรู้แบบเสริมกำลัง (Reinforcement learning)

เป็นการเรียนรู้โดยจะเปลี่ยนแปลงตามสภาพแวดล้อม เช่น การฝึกสัตว์เลี้ยง

2.10 โครงข่ายประสาทเทียม (Artificial neural network)

คือ โมเดลทางคณิตศาสตร์ สำหรับประมวลผลสารสนเทศด้วยการคำนวณแบบคอนเนคชันนิสต์ (Connectionist) เพื่อจำลองการทำงานของเครือข่ายประสาทสมองในมนุษย์ เพื่อที่จะสร้างเครื่องมือซึ่งมีความสามารถในการจดจำรูปแบบ (Pattern recognition) และการสร้างความรู้ใหม่ (Knowledge Extraction) เช่นเดียวกับความสามารถที่มีในสมองมนุษย์ แนวคิดเริ่มต้นของเทคนิคนี้ได้มาจากการศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงข่ายไฟฟ้าชีวภาพ (Bioelectric Network) ในสมอง ซึ่งประกอบด้วย เซลล์ประสาท หรือ "นิวรอน" (Neurons) และ "จุดประสานประสาท" (Synapses) แต่ละเซลล์ประสาทประกอบด้วยปลายในการรับกระแสประสาท เรียกว่า "เดนไดรท์" (Dendrite) ซึ่งเป็นข้อมูลตัวอย่างขาเข้า และปลายในการส่งกระแสประสาทเรียกว่า "แอคซอน" (Axon) ซึ่งเป็นเหมือน output ของเซลล์ เซลล์เหล่านี้ทำงานด้วยปฏิกิริยาไฟฟ้าเคมี เมื่อมีการกระตุ้นด้วยสิ่งเร้าภายนอกหรือกระตุ้นด้วยเซลล์ด้วยกัน กระแสประสาทจะวิ่งผ่านเดนไดรท์เข้าสู่นิวเคลียสซึ่งจะเป็นตัวตัดสินว่าต้องกระตุ้นเซลล์อื่นๆ ต่อหรือไม่ ถ้ากระแสประสาทแรงพอนิวเคลียสก็จะกระตุ้นเซลล์อื่นๆ ต่อไปผ่านทางแอคซอนของมัน

2.10.1 หลักการ

หลักการเอามาประยุกต์ใช้ สำหรับในคอมพิวเตอร์ Neurons ประกอบด้วย ข้อมูลตัวอย่างขาเข้า และ output เหมือนกัน โดยจำลองให้ข้อมูลตัวอย่างขาเข้าแต่ละอันมี weight เป็นตัวกำหนดน้ำหนักของข้อมูลตัวอย่างขาเข้า โดย neuron แต่ละหน่วยจะมีค่า threshold เป็นตัวกำหนดว่าน้ำหนักรวมของข้อมูลตัวอย่างขาเข้าต้องมากขนาดไหนจึงจะสามารถส่ง output ไปยัง neurons ตัวอื่นได้ เมื่อนำ neuron แต่ละหน่วยมาต่อกันให้ทำงานร่วมกันการทำงานนี้ในทางตรรกะแล้วก็จะเหมือนกับปฏิกิริยาเคมีที่เกิดในสมอง เพียงแต่ในคอมพิวเตอร์ทุกอย่างเป็นตัวเลขเท่านั้นเอง

2.10.2 การทำงาน

การทำงานของ Neural Networks คือเมื่อมีข้อมูลตัวอย่างขาเข้า เข้ามายัง network ก็เอาข้อมูลตัวอย่างขาเข้า มาคูณกับ weight ของแต่ละขา ผลที่ได้จากข้อมูลตัวอย่างขาเข้าทุกๆ ขาของ neuron จะเอามารวมกันแล้วก็เอามาเทียบกับ threshold ที่กำหนดไว้ ถ้าผลรวมมีค่ามากกว่า threshold แล้ว neuron ก็จะส่ง output ออกไป output นี้ก็จะถูกส่งไปยัง ข้อมูลตัวอย่างขาเข้า ของ neuron อื่นๆ ที่เชื่อมกันใน network ถ้าค่าน้อยกว่า threshold ก็จะไม่เกิด output สิ่งสำคัญคือเราต้องทราบค่า weight และ threshold สำหรับสิ่งที่เราต้องการเพื่อให้คอมพิวเตอร์รู้จำ ซึ่งเป็นค่าที่ไม่แน่นอน แต่สามารถกำหนดให้คอมพิวเตอร์ปรับค่าเหล่านั้นได้โดยการสอนให้มันรู้จัก pattern ของสิ่งที่เราต้องการให้มันรู้จำ เรียกว่า "back propagation" ซึ่งเป็นกระบวนการย้อนกลับของการรู้จำ ในการฝึก feed-forward Neural Networks จะมีการใช้อัลกอริทึมแบบ back-propagation เพื่อใช้ในการปรับปรุงน้ำหนักคะแนนของเครือข่าย (Network Weight) หลังจากใส่รูปแบบข้อมูลสำหรับฝึกให้แก่เครือข่ายในแต่ละครั้งแล้ว ค่าที่ได้รับ output จากเครือข่ายจะถูกนำไปเปรียบเทียบกับผลที่คาดหวัง แล้วทำการคำนวณหาค่าความผิดพลาด ซึ่งค่าความผิดพลาดนี้จะถูกส่งกลับเข้าสู่เครือข่ายเพื่อใช้แก้ไขค่าน้ำหนักคะแนนต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10.3 ส่วนประกอบของโครงข่ายประสาทเทียม

1. ข้อมูลตัวอย่างขาเข้า (Input) เป็นข้อมูลที่เป็นตัวเลข หากเป็นข้อมูลเชิงคุณภาพ ต้องแปลงให้อยู่ในรูปเชิงปริมาณที่โครงข่ายประสาทเทียมยอมรับได้

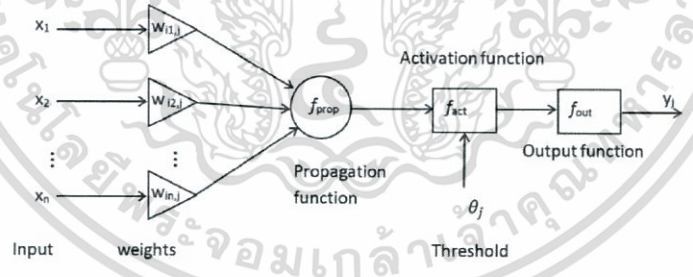
2. ข้อมูลส่งออก (Output) คือ ผลลัพธ์ที่เกิดขึ้นจริง (Actual output) จากระบวนการเรียนรู้ของโครงข่ายประสาทเทียม

3. ค่าน้ำหนัก (Weights) คือ สิ่งที่ได้จากการเรียนรู้ของโครงข่ายประสาทเทียม หรือเรียกอีกอย่างหนึ่งว่า ค่าความรู้ (Knowledge) ค่านี้จะถูกเก็บเป็นทักษะเพื่อใช้ในการจดจำข้อมูลอื่นๆ ที่อยู่ในรูปแบบเดียวกัน

4. ฟังก์ชันผลรวม (Summation function : S) เป็นผลรวมของข้อมูลป้อนเข้า (a_i) และค่าน้ำหนัก (w_i)

$$S = \sum_{i=1}^{n} a_i w_i \tag{2.4}$$

5. ฟังก์ชันการแปลง (Transfer Function) เป็นการคำนวณการจำลองการทำงานของโครงข่ายประสาทเทียม เช่น ซิกมอยด์ฟังก์ชัน (Sigmoid Function) ฟังก์ชันไฮเปอร์โบลิกแทนเจนต์ (Hyperbolic tangent function) เป็นต้น



รูปที่ 2.1 โครงสร้างการทำงานของโครงข่ายประสาทเทียม

(อ้างอิงโดย https://www.researchgate.net/figure/A-sample-neural-network-in-which-a-rectangle-is-an-input-node-and-a-circle-is-a-neuron_fig4_279757450)

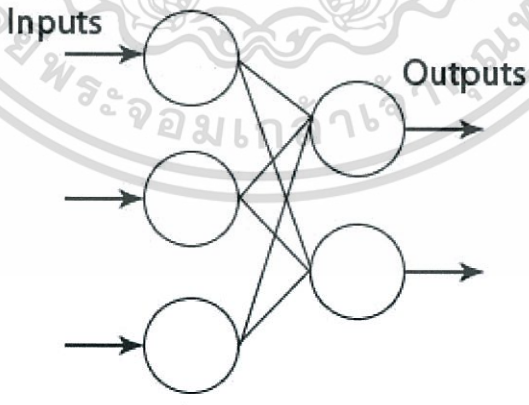
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10.4 ลักษณะของโครงข่ายประสาทเทียม

ลักษณะของโครงข่ายประสาทเทียมโครงข่ายประสาทเทียมประกอบด้วยเซลล์ประสาทเทียมหรือโหนดจำนวนมากเชื่อมต่อกันซึ่งการเชื่อมต่อแบ่งออกเป็นกลุ่มย่อย เรียกว่า ชั้น(layer) ชั้นแรก เป็นชั้นนำข้อมูลเข้า เรียกว่า ชั้นรับข้อมูลป้อนเข้า (input layer) ส่วนชั้นสุดท้ายเรียกว่า ชั้นส่งข้อมูลออก (output layer) และชั้นที่อยู่ระหว่างชั้นรับข้อมูลป้อนเข้าและชั้นส่งข้อมูลออกเรียกว่าชั้นแอบแฝง (hidden layer) ซึ่งโดยทั่วไปชั้นแอบแฝงอาจมีมากกว่า 1 ชั้นก็ได้ ด้วยเหตุนี้จึงสามารถแบ่งประเภทของโครงข่ายประสาทเทียมตามจำนวนชั้นของโครงข่ายแบบกว้าง ๆ ได้ 2 แบบ ได้แก่ โครงข่ายแบบชั้นเดียว (single layer) และโครงข่ายแบบหลายชั้น Multi layer)

2.10.4.1 โครงข่ายแบบชั้นเดียว (Single layer)

เป็นโครงข่ายประสาทเทียมอย่างง่ายที่มีเพียงชั้นรับข้อมูลป้อนเข้าและชั้นส่งข้อมูลออกเท่านั้น โหนดในชั้นรับข้อมูลป้อนเข้าทำหน้าที่รับข้อมูลเข้า (input value) แล้วส่งข้อมูลผ่านเส้นเชื่อมโยงต่าง ๆ ไปให้โหนดในชั้นส่งข้อมูลออก ความเข้มของสัญญาณ หรือ ปริมาณข้อมูลที่นำเข้าสู่โหนดในชั้นส่งข้อมูลออกจะขึ้นอยู่กับค่าน้ำหนักที่อยู่บนเส้นเชื่อมโยงโหนดในชั้นส่งข้อมูลออกจะนำข้อมูลที่ได้รับมาคำนวณโดยใช้ฟังก์ชันทางคณิตศาสตร์ที่เรียกว่า ฟังก์ชันการแปลง ที่เหมาะสมกับปัญหา แล้วส่งผลลัพธ์ที่ได้ออกมาเป็นข้อมูลส่งออก เช่น โครงข่ายแบบชั้นเดียวแบบเพอเซปตรอนอย่างง่าย (Simple perceptron) และโครงข่ายโฮปฟิลด์ (Hopfield networks) ลักษณะโครงข่ายแบบชั้นเดียวแสดงดังรูป



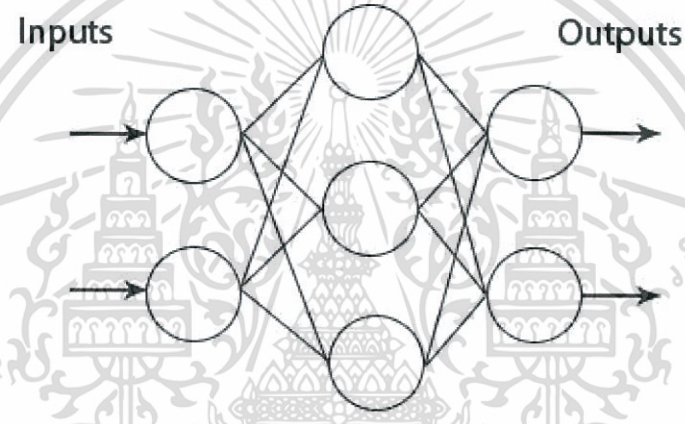
รูปที่ 2.2 โครงสร้างโครงข่ายประสาทเทียมแบบชั้นเดียว

(อ้างอิงโดย http://neuroph.sourceforge.net/tutorials/Perceptron.html)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10.4.2 โครงข่ายแบบหลายชั้น (Multi layer)

โครงข่ายแบบหลายชั้น เป็นโครงข่ายที่มีชั้นแอบแฝง (Hidden layer) ตั้งแต่ 1 ชั้นขึ้นไป โครงข่ายแบบหลายชั้นจะใช้ในกรณีที่ปัญหามีความซับซ้อน ซึ่งโครงข่ายแบบชั้นเดียวไม่สามารถแก้ปัญหาได้ จึงเพิ่มจำนวนโหนดที่มีการคำนวณ หรือชั้นแอบแฝงให้กับโครงข่าย ตัวอย่างของโครงข่ายแบบหลายชั้น เช่น การแพร่ย้อนกลับ (Back propagation) เซลฟ์ออร์แกนไนซิ่งแมปซ์ (Self-organizing maps) และเคาน์เตอร์พรอพะเกชัน (Counter propagation) เป็นต้น ลักษณะโครงสร้างโครงข่ายแบบหลายชั้นแสดงดังรูป



รูปที่ 2.3 โครงสร้างโครงข่ายประสาทเทียมแบบหลายชั้น
(อ้างอิงโดย http://neuroph.sourceforge.net/tutorials/MultiLayerPerceptron.html)

2.10.5 ประเภทของการเรียนรู้ของโครงข่ายประสาทเทียม

2.10.5.1 การเรียนรู้แบบมีผู้สอน (Supervised learning)

ข้อมูลจะประกอบด้วยตัวอย่างข้อมูลที่ต้องการสอน และผลลัพธ์ที่ต้องการให้โครงข่ายสร้าง เมื่อมีการนำข้อมูลในลักษณะเดียวกันมาเป็นข้อมูลป้อนเข้า โครงข่ายจะกำหนดค่าผลลัพธ์ที่เป็นเป้าหมายให้กับข้อมูลป้อนเข้าแต่ละตัวโครงข่ายจะนำค่าผิดพลาดระหว่าง ค่าเป้าหมายกับ ค่าผลลัพธ์ที่ได้มาใช้ในการปรับค่าน้ำหนัก เพื่อให้ค่าผลลัพธ์ที่ใกล้เคียงกับเป้าหมายมากที่สุด ถ้าหากเปรียบเทียบกับมนุษย์จะเหมือนกับการสอนนักเรียนโดยมีครูผู้สอนคอยให้คำแนะนำ ตัวอย่างแบบจำลองนี้ได้แก่ การแพร่ย้อนกลับ และเพอเซบตรอน (Perceptron) เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10.5.2 การเรียนรู้แบบไม่มีผู้สอน (Unsupervised learning)

การเรียนรู้แบบนี้จะสอนโครงข่ายโดยการนำข้อมูลป้อนเข้าอย่างต่อเนื่องเพียงอย่างเดียว ไม่มีการส่งค่าผลลัพธ์เป้าหมายให้กับข้อมูลป้อนเข้าแต่ละตัว การปรับน้ำหนักจะใช้ข้อมูลที่นำมาสอนเป็นตัวปรับค่า โดยค่าน้ำหนักจะปรับตามกลุ่มที่ข้อมูลป้อนเข้าที่มีรูปแบบคล้ายคลึงกันถ้าหากเปรียบเทียบกับมนุษย์จะเหมือนกับการที่เราสามารถแยกแยะพันธุ์พืช พันธุ์สัตว์ ตามลักษณะรูปร่างของมันได้ด้วยตนเอง ตัวอย่างแบบจำลองนี้ได้แก่ เคาน์เตอร์พรอพะเกชัน (Counter propagation : CPN) แบบจำลองอะแดปทีฟรีโซแนนซ์เทียรี่ (Adaptive Resonance Theory neural networks : ART) เป็นต้น

2.10.6 การเรียนรู้แบบแพร่ย้อนกลับ

Werbos ได้เสนอแนวคิดของการเรียนรู้แบบแพร่ย้อนกลับ จากนั้น Parker ได้นำเสนออีกครั้ง และเป็นที่รู้จักมากยิ่งขึ้นโดย Rumelhart และคณะ ในหนังสือ Parallel Distributed Processing ซึ่งได้กล่าวถึงศักยภาพของโครงข่ายประสาทเทียม และ การเรียนรู้ แบบแพร่ย้อนกลับการเรียนรู้แบบแพร่ย้อนกลับ สามารถแก้ปัญหาที่ต้องการรูปแบบ โดยการป้อนรูปแบบเข้าไป โครงข่ายประสาทเทียมจะให้รูปแบบผลลัพธ์ที่เกี่ยวข้องกัน การเรียนรู้แบบแพร่ย้อนกลับเป็นวิธีการหนึ่งของโครงข่ายประสาทเทียมที่ง่ายต่อการเข้าใจ เนื่องจากกระบวนการเรียนรู้ และปรับปรุงแก้ไขนั้นเป็นไปด้วยตนเอง ถ้าโครงข่ายประสาทเทียมให้คำตอบที่ผิด ค่าน้ำหนัก การเรียนรู้แบบแพร่ย้อนกลับจะถูกปรับจนกว่าค่าความผิดพลาดจะน้อยลง หรืออยู่ในเกณฑ์ที่ยอมรับได้ นั่นคือ ค่าที่ได้ในครั้งถัดไปจะมีความถูกต้องมากยิ่งขึ้น ด้วยโครงสร้างประสาทเทียมที่มีลักษณะเป็นชั้น แต่ละชั้นเชื่อมโยงกันอย่างทั่วถึง เมื่อโครงข่ายประสาทเทียมได้รับข้อมูลป้อนเข้า จะคำนวณค่าน้ำหนักของหน่วยรับข้อมูลป้อนเข้าไปยังชั้นแอบแฝง และจากชั้นแอบแฝงไปยังชั้นส่งข้อมูลออก เมื่อเกิดผลต่างระหว่างค่าผลลัพธ์จริงกับค่าผลลัพธ์เป้าหมาย โครงข่ายประสาทเทียมจะปรับค่าความผิดพลาดจากชั้นส่งข้อมูลออก และแพร่ย้อนกลับไปยังชั้นแอบแฝงจากนั้นจึงแพร่ย้อนกลับไปยังชั้นรับข้อมูลป้อนเข้าตามลำดับขั้นตอนการเรียนรู้แบบแพร่ย้อนกลับ ประกอบด้วย 2 ขั้นตอน ดังนี้

2.10.6.1 การแพร่เดินหน้า (Forward propagation)

การแพร่เดินหน้า (forward propagation) ขั้นตอนนี้เริ่มเมื่อเซลล์ประสาทได้รับข้อมูลป้อนเข้า ซึ่งเป็นการคำนวณ ผลรวมของผลลัพธ์ที่เข้ามายังหน่วยที่ j ดังสมการ

$$S = \sum_{i=1}^{n} a_i w_{ji} \qquad (2.5)$$

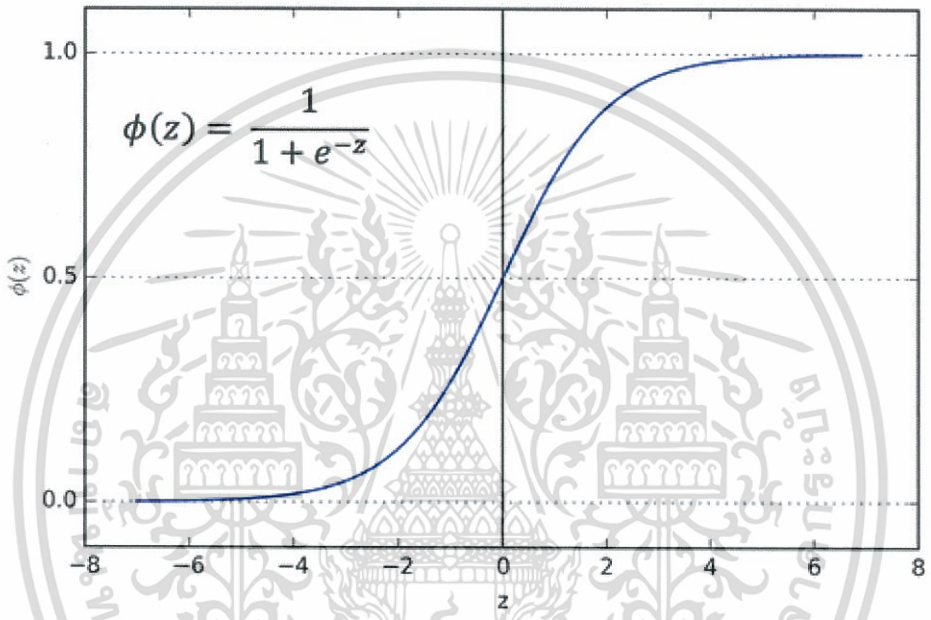
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่

a_i คือ ข้อมูลจากหน่วยที่ i

w_{ji} คือ ค่าน้ำหนักจากหน่วยที่ I ไปยังหน่วยที่ j

จากนั้นทำการแปลงค่าข้อมูลโดยการคำนวณค่าผลลัพธ์ $f(s_j)$โดยใช้ซิกมอยด์ฟังก์ชัน



รูปที่ 2.4 ซิกมอยด์ฟังก์ชัน

(อ้างอิงโดย https://www.kdnuggets.com/2016/08/role-activation-function-neural-network.html)

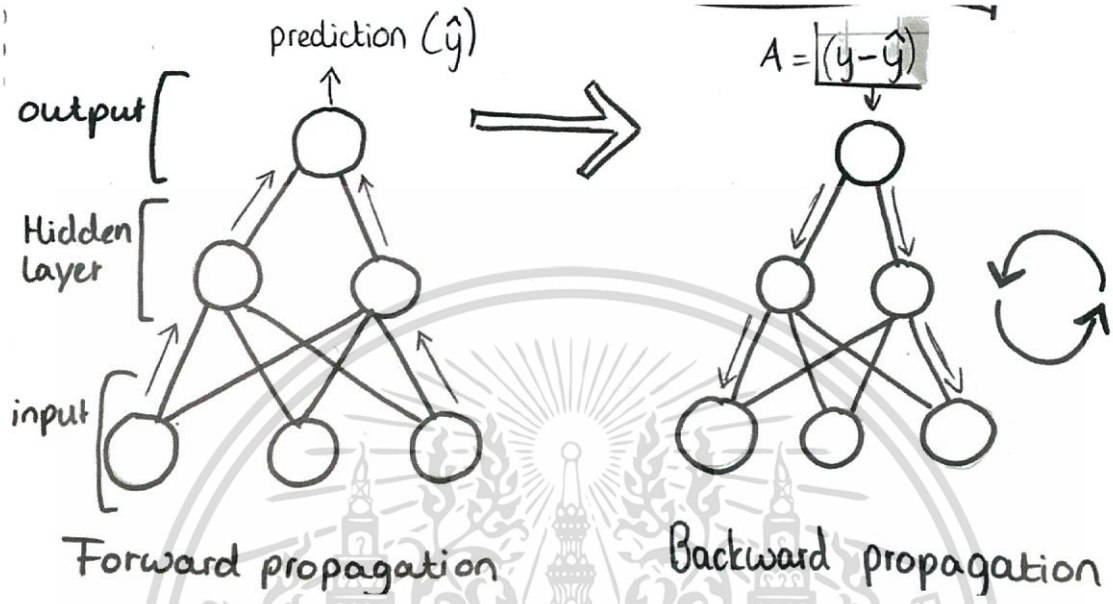
สมการของซิกมอยด์คือเมื่อ x เป็นค่าผลรวมของหน่วยที่ j ดังนั้นสมการนี้เมื่อได้รับค่า $f(S_j)$แล้ว ค่า $f(S_j)$ จะกลายเป็นผลลัพธ์ของหน่วยที่ j $f(x) = 1$ สำหรับชั้นรับข้อมูลป้อนเข้า จะไม่มีการประมวลผลเกิดขึ้น และไม่มีการแปลงค่าข้อมูลเนื่องจากแต่ละหน่วยประมวลผลจะใช้ค่าของข้อมูลป้อนเข้าเป็นค่าของตนเอง โครงข่ายประสาทเทียมแบบการแพร่เดินหน้าแสดงดังรูปที่ 2.5

2.10.6.2 การแพร่ย้อนกลับ (Back propagation)

ขั้นตอนนี้ ค่าความผิดพลาด (δ) จะถูกคำนวณที่หน่วยประมวลผลทั้งหมด และคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าน้ำหนักที่เปลี่ยนของทุกการเชื่อมโยง การคำนวณนี้เริ่มที่ชั้นส่งข้อมูลออก และแพร่ย้อนกลับไปยังชั้นรับข้อมูลป้อนเข้า



รูปที่ 2.5 ภาพรวมของ Forward propagation และ Backward propagation (อ้างอิงโดย https://medium.freecodecamp.org/building-a-3-layer-neural-network-from-scratch-99239c4af5d3)

2.11 Image Retraining

2.11.1 กระบวนการคอขวด (Bottlenecks)

กระบวนการคอขวด คือการวิเคราะห์รูปภาพทั้งหมดที่อยู่ในโฟลเดอร์ หลังจากนั้นจึงทำการคำนวณแล้วเก็บค่าของกระบวนการคอขวด (Bottleneck values) ของแต่ละภาพ ค่าของกระบวนการคอขวดของแต่ละภาพจะมีความแตกต่างกันมากพอที่จะจัดจำแนกระหว่างประเภทของภาพได้ เพราะฉะนั้นจะใช้ปริมาณข้อมูลน้อยที่สุดในการจำแนกระหว่างประเภทของภาพ

เพราะว่าในแต่ละภาพจะถูกใช้ในการคำนวณหลายครั้ง ในระหว่างขั้นตอนการเทรนโมเดล เพราะฉะนั้นจึงใช้เวลาเป็นจำนวนมาก เราจึงต้องใช้ กระบวนการคอขวดมาใช้เพื่อเพิ่มความเร็วในการเทรนโมเดลจะได้ไม่ต้องคำนวณซ้ำหลายๆ ครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11.2 การเทรนโมเดล (Training)

หลังจากทำการกระบวนการคอขวดเสร็จสิ้น ระบบจะเริ่มทำการประมวลผลตั้งแต่ขั้นตอนนี้จะแสดงผลออกมา 3อย่างคือค่าความถูกต้องในการเทรนโมเดล (Training accuracy), ค่าความถูกต้องในการตรวจสอบ (Validation accuracy) และ ครอสเอนโทรปี (Cross entropy)

1. Training accuracy คือเปอร์เซ็นต์ของภาพที่ถูกใช้ในชุดการเทรนโมเดลปัจจุบันเทียบกับประเภทของภาพ

2. Validation accuracy คือความถูกต้องในการสุ่มเลือกภาพจากกลุ่มที่แตกต่าง

3. Cross entropy คือความแม่นยำในการเทรนโมเดลโดยขึ้นอยู่กับภาพในการเทรนโมเดลว่าสามารถเรียนรู้ได้เท่าไร ซึ่งในขั้นตอนนี้จะมีโอกาสเกิดค่ารบกวนหรือเกิดความไม่ถูกต้องขึ้นได้

การประเมินประสิทธิภาพของเครือข่ายคือการประเมินประสิทธิภาพของเซตของดาต้าที่ไม่อยู่ในขั้นตอนการเทรนโมเดล โดยหาได้จากค่า Validation accuracy ถ้า Training accuracy มีค่าเยอะแต่ค่า validation accuracy มีค่าที่ต่ำ จะหมายถึงเครือข่ายจะอ้างอิงและจดจำที่ไม่ถูกต้อง Cross entropy คือฟังก์ชั่นการสูญเสีย (Loss function) ที่บอกถึงว่ากระบวนการเรียนรู้ที่กำลังดำเนิน สามารถดำเนินได้ดีเพียงใด เป้าหมายของการเทรนโมเดลคือการทำให้ค่ามีค่าการสูญเสียต่ำที่สุดจะสามารถทำให้ระบบสามารถไม่สนใจสัญญาณรบกวนที่มีขนาดเล็กได้

โดยปกติแล้วขั้นตอนนี้จะใช้ 4,000 ขั้นตอนในการเทรนโมเดล ในแต่ละขั้นตอนจะใช้ 10ภาพที่ได้มาจากสุ่มในเซตการเทรน (Training set) แล้วทำการหาค่ากระบวนคอขวดที่เก็บเอาไว้ แล้วนำค่านั้นมาป้อนเข้าสู่ final layer เพื่อหาการคาดการณ์ ค่าการคาดการณ์ จะถูกเปรียบเทียบกับ label ที่แท้จริง แล้วทำการปรับปรุงค่า final layer's weights ผ่านกระบวนการย้อนกลับ (Back Propagation)

เมื่อกระบวนการนี้ดำเนินต่อไปค่าความแม่นยำจะเพิ่มขึ้นเรื่อยๆ หลังจากขั้นตอนทั้งหมดได้จบลงจะมีค่า final test accuracy จะเป็นการเทียบภาพที่ถูกแยกออกมาจากการขั้นตอนการเทรนและการทำvalidation การหา final test accuracy คือการประเมินประสิทธิภาพทั้งหมดของโมเดลที่ได้จากการทำกระบวนการทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

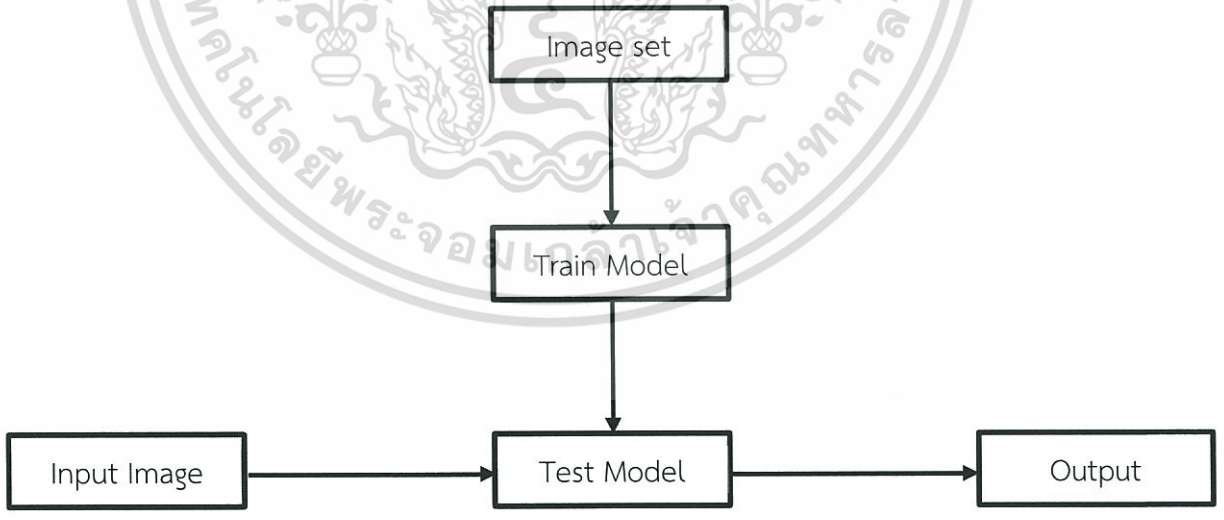
การวิเคราะห์และออกแบบระบบ

โครงสร้างการทำงานของ ระบบตรวจสอบและจำแนกพันธุ์กุหลาบ

ประกอบไปด้วย 5 ส่วนดังนี้

1. รูปภาพที่จะนำมาประมวลผล (Input image)
2. รูปภาพที่นำมา Train (Image set)
3. ขั้นตอนการ Train model เพื่อจำแนกสายพันธุ์
4. ขั้นตอนการ Test Model เพื่อหาสายพันธุ์ที่มีความใกล้เคียงกับสายพันธุ์ที่อยู่ในฐานข้อมูลมากที่สุด
5. แสดงผลลัพธ์การทำงานออกมา

กระบวนการตรวจสอบและจำแนกสายพันธุ์กุหลาบแสดงดังรูปที่ 3.1



รูปที่ 3.1 แผนภาพบล็อก(block diagram) แสดงการตรวจสอบและจำแนกสายพันธุ์กุหลาบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 รูปภาพที่จะนำมาประมวลผล (Input Image)

เป็นรูปภาพของสายพันธุ์ที่มีอยู่ในระบบแต่ไม่ได้นำไป train กับ model

รูปที่ 3.2 รูปที่ต้องการมาจำแนกพันธุ์

3.2 รูปภาพที่นำมา train

ใช้รูปภาพที่ได้จากการถ่ายรูปรวมถึงหาจากในอินเตอร์เน็ต(Internet)

รูปที่ 3.3 รูปภาพที่ได้จากการถ่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ขั้นตอนการ Train

เป็นขั้นตอนการ train model เพื่อนำไฟล์ graph.pb และ label.txt ไปใช้ในขั้นตอนต่อไป

```
INFO:tensorflow:2018-03-14 01:47:33.900452: Step 3930: Train accuracy = 100.0%
INFO:tensorflow:2018-03-14 01:47:33.900452: Step 3930: Cross entropy = 0.002961
INFO:tensorflow:2018-03-14 01:47:33.981456: Step 3930: Validation accuracy = 91.0% (N=100)
INFO:tensorflow:2018-03-14 01:47:34.832505: Step 3940: Train accuracy = 100.0%
INFO:tensorflow:2018-03-14 01:47:34.832505: Step 3940: Cross entropy = 0.002666
INFO:tensorflow:2018-03-14 01:47:34.914510: Step 3940: Validation accuracy = 87.0% (N=100)
INFO:tensorflow:2018-03-14 01:47:35.760558: Step 3950: Train accuracy = 100.0%
INFO:tensorflow:2018-03-14 01:47:35.760558: Step 3950: Cross entropy = 0.003089
INFO:tensorflow:2018-03-14 01:47:35.842563: Step 3950: Validation accuracy = 95.0% (N=100)
INFO:tensorflow:2018-03-14 01:47:36.702612: Step 3960: Train accuracy = 100.0%
INFO:tensorflow:2018-03-14 01:47:36.702612: Step 3960: Cross entropy = 0.003138
INFO:tensorflow:2018-03-14 01:47:36.783617: Step 3960: Validation accuracy = 94.0% (N=100)
INFO:tensorflow:2018-03-14 01:47:37.637666: Step 3970: Train accuracy = 100.0%
INFO:tensorflow:2018-03-14 01:47:37.637666: Step 3970: Cross entropy = 0.002728
INFO:tensorflow:2018-03-14 01:47:37.720670: Step 3970: Validation accuracy = 93.0% (N=100)
INFO:tensorflow:2018-03-14 01:47:38.601721: Step 3980: Train accuracy = 100.0%
INFO:tensorflow:2018-03-14 01:47:38.602721: Step 3980: Cross entropy = 0.003255
INFO:tensorflow:2018-03-14 01:47:38.683725: Step 3980: Validation accuracy = 88.0% (N=100)
INFO:tensorflow:2018-03-14 01:47:39.565776: Step 3990: Train accuracy = 100.0%
INFO:tensorflow:2018-03-14 01:47:39.565776: Step 3990: Cross entropy = 0.002958
INFO:tensorflow:2018-03-14 01:47:39.651781: Step 3990: Validation accuracy = 88.0% (N=100)
INFO:tensorflow:2018-03-14 01:47:40.430825: Step 3999: Train accuracy = 100.0%
INFO:tensorflow:2018-03-14 01:47:40.430825: Step 3999: Cross entropy = 0.002703
INFO:tensorflow:2018-03-14 01:47:40.512830: Step 3999: Validation accuracy = 93.0% (N=100)
INFO:tensorflow:Final test accuracy = 89.5% (N=172)
INFO:tensorflow:Froze 2 variables.
2018-03-14 01:41:09.596471: I C:\tf_jenkins\workspace\rel-win\M\windows\PY\35\tensorflow\core\platform\cpu
```

รูปที่ 3.4 ขั้นตอนการ train model

3.4 การทดสอบ Model

ขั้นตอนการตรวจสอบและจำแนกสายพันธุ์

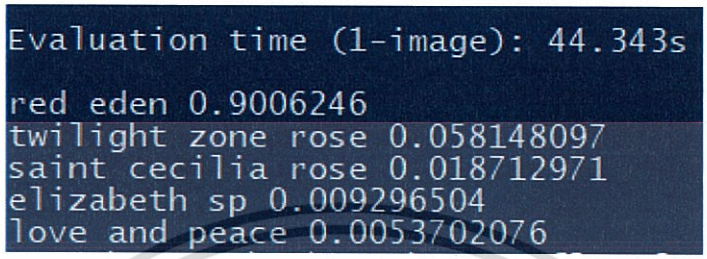
```
37] Your CPU supports instructions that this TensorFlow binary was not compiled to use: AVX AVX2
2018-03-14 01:57:10.336755: I C:\tf_jenkins\home\workspace\rel-win\M\windows-gpu\PY\36\tensorflow\core\common_runtime\gpu\gpu_device.c
c:1030] Found device 0 with properties:
name: GeForce GTX 950M major: 5 minor: 0 memoryClockRate(GHz): 0.928
pciBusID: 0000:01:00.0
totalMemory: 4.00GiB freeMemory: 3.35GiB
2018-03-14 01:57:10.356632: I C:\tf_jenkins\home\workspace\rel-win\M\windows-gpu\PY\36\tensorflow\core\common_runtime\gpu\gpu_device.c
c:1120] Creating TensorFlow device (/device:GPU:0) -> (device: 0, name: GeForce GTX 950M, pci bus id: 0000:01:00.0, compute capability
: 5.0)
2018-03-14 01:57:34.372851: I C:\tf_jenkins\home\workspace\rel-win\M\windows-gpu\PY\36\tensorflow\core\common_runtime\gpu\gpu_device.c
c:1120] Creating TensorFlow device (/device:GPU:0) -> (device: 0, name: GeForce GTX 950M, pci bus id: 0000:01:00.0, compute capability
: 5.0)
2018-03-14 01:57:36.114323: W C:\tf_jenkins\home\workspace\rel-win\M\windows-gpu\PY\36\tensorflow\core\framework\op_def_util.cc:334] O
p BatchNormWithGlobalNormalization is deprecated. It will cease to work in GraphDef version 9. Use tf.nn.batch_normalization().
```

รูปที่ 3.5 ขั้นตอนการจำแนกสายพันธุ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การแสดงผล

แสดงผลลัพธ์ที่ได้จากการประมวลผลหารูปที่ใกล้เคียงมากที่สุดแล้วนำมาแสดงผล และสามารถบอกข้อมูลของสายพันธุ์ที่ใกล้เคียงด้วย



รูปที่ 3.6 ผลลัพธ์ที่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

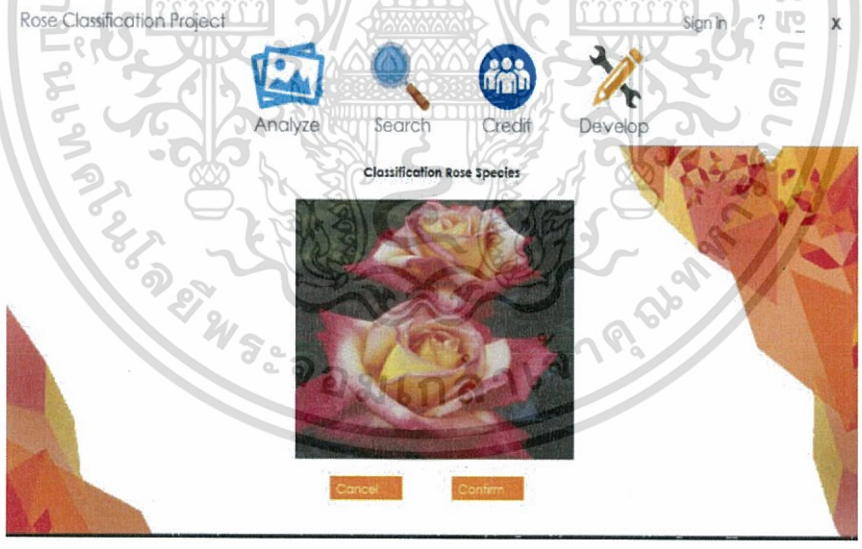
บทที่ 4

การทดลองและผลการทดลอง

ในบทนี้จะกล่าวถึงการทดลองและผลการทดลองของโปรแกรมระบบตรวจสอบและจำแนกพันธุ์กุหลาบโดยโปรแกรมจะประกอบไปด้วย 4 ส่วนหลักคือ หน้าโปรแกรมหลัก หน้าแสดงผลลัพธ์ หน้าข้อมูลของกุหลาบแต่ละสายพันธุ์ และหน้าสำหรับtrain model

4.1 เริ่มต้นใช้งานโปรแกรม

เมื่อผู้ใช้ทำการกดเข้าใช้งานโปรแกรม ตัวโปรแกรมจะปรากฏหน้าหลักของตัวโปรแกรมขึ้นมา โดยหน้านี้จะเป็นหน้าที่ให้ผู้ใช้ทำการเลือกรูปภาพโดยคลิกที่คำว่า Drop file here to attach them ซึ่งเมื่อคลิกจะปรากฏ Browse dialog ขึ้นมาเพื่อให้ผู้ใช้เลือกไฟล์ภาพที่ต้องการ หรือ ผู้ใช้จะสามารลากรูปมาใส่ในตัวโปรแกรมเลยก็ได้

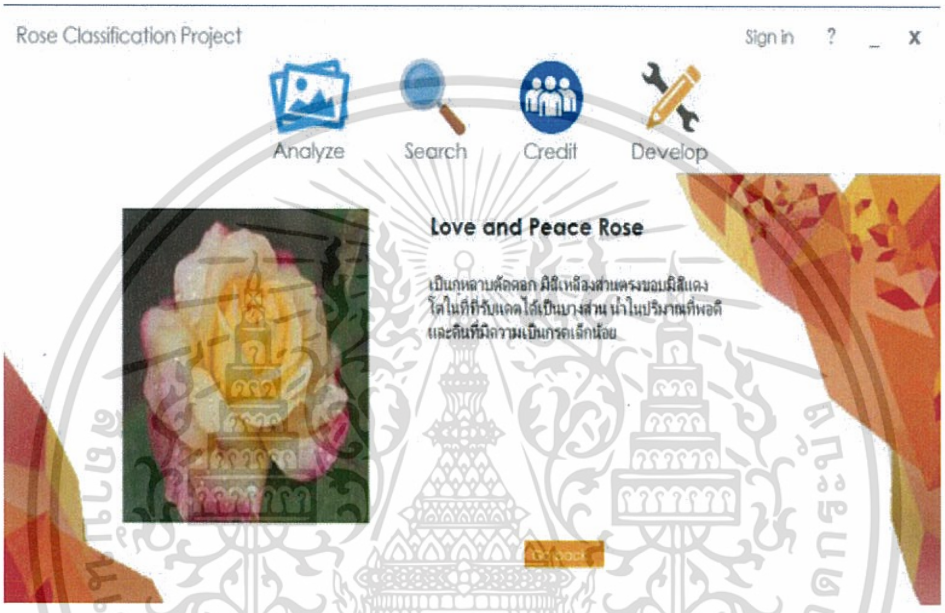


รูปที่ 4.1 หน้าหลักของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 หน้าแสดงผลลัพธ์ของโปรแกรม

หน้านี้จะเป็นหน้าหลังจากที่เราเลือกรูปภาพและทำการกดปุ่ม Confirm โดยโปรแกรมจะทำการวิเคราะห์ว่ารูปภาพที่ผู้ใช้ทำการป้อนเข้ามามีความคล้ายคลึงกับข้อมูลสายพันธุ์กุหลาบใดมากที่สุดจากนั้นจึงทำการแสดงผลออกมาทางหน้านี้

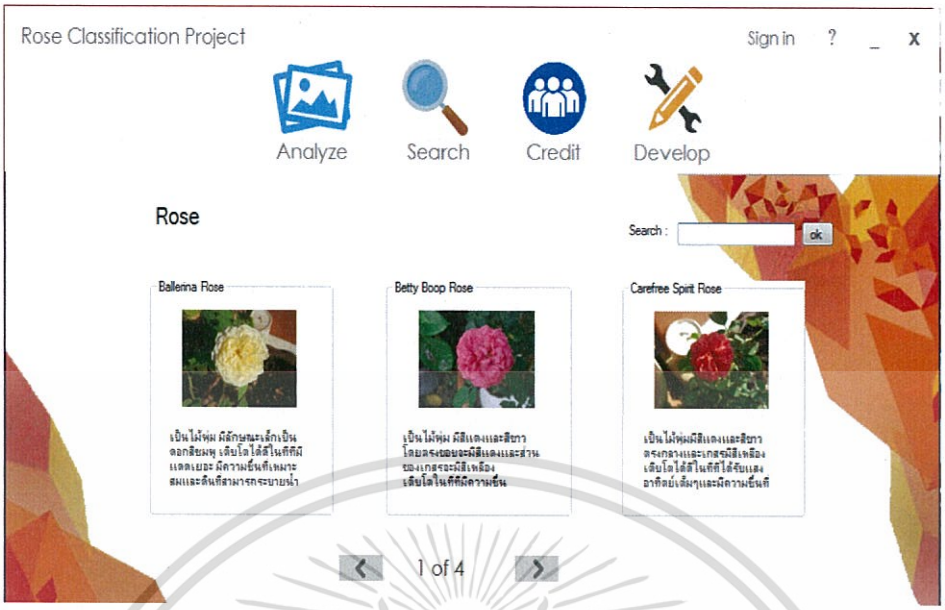


รูปที่ 4.2 หน้าแสดงผลลัพธ์

4.3 หน้าข้อมูลของกุหลาบ

หน้านี้จะเป็นหน้าที่แสดงข้อมูลของกุหลาบในฐานข้อมูลที่เรามีอยู่ทั้งหมดโดยเรามีระบบ Search engine เพื่อช่วยในการหาสายพันธุ์ที่ต้องการได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 หน้าแสดงสายพันธุ์กุหลาบทั้งหมดในฐานข้อมูล

4.4 หน้าสำหรับผู้พัฒนา

หน้านี้เป็นหน้าที่ถูกออกแบบมาเพื่ออำนวยความสะดวกสำหรับผู้พัฒนาโดยจะเป็นหน้าสำหรับการ Train model ซึ่ง เมื่อกดปุ่ม Browse จะแสดงหน้า Browse dialog ขึ้นมาเพื่อเลือก Folder ที่บรรจุรูปภาพที่เราต้องการจะเทรนซึ่งเราสามารดูสถานะได้ทางด้านขวามือ

รูปที่ 4.4 หน้าสำหรับ train model

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

โครงงานนี้นำเสนอในเรื่องของการพัฒนาระบบตรวจสอบและจำแนกพันธุ์กุหลาบ โดยระบบสามารถจำแนกสายพันธุ์กุหลาบได้ 10 สายพันธุ์ซึ่งมีความแม่นยำอยู่ที่ประมาณ 70 เปอร์เซ็นต์ นอกจากนี้ยังมีระบบค้นหา ช่วยในการค้นหาข้อมูลของสายพันธุ์ที่ต้องการ และยังมีส่วนของการเทรนโมเดล

5.2 ปัญหาที่เกิดขึ้นและแนวทางแก้ไข

- ข้อมูลมีจำนวนไม่มากพอทำให้ตัวระบบสามารถวิเคราะห์ได้แม่นยำมากขึ้น
- วิธีการ Image processing ในตอนแรกมีปัญหาจึงต้องเปลี่ยนมาใช้ machine learning มาช่วยแทน
- Machine learning เป็นเรื่องที่ค่อนยากจึงต้องใช้เวลานานในการเรียนรู้
- มี Bug ระหว่างที่ Test model อยู่ซึ่งไม่ทราบสาเหตุและ Bug หายไปเอง

5.3 แนวทางการพัฒนาต่อ

สามารถนำไปพัฒนาให้อยู่บนอุปกรณ์เคลื่อนที่ แล้วสามารถวิเคราะห์แบบเวลาจริง ได้ รวมถึงสามารถเพิ่มความแม่นยำให้กับตัวโปรแกรมได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] Pete Warden. **TensorFlow for Poets.** [Online].
Available : https://petewarden.com/2016/02/28/tensorflow-for-poets 2016.

[2] **How to Retrain an Image Classifier for New Categories.** [Online].
Available : https://www.tensorflow.org/tutorials/image_retraining 2018.

[3] **โครงข่ายประสาทเทียม Artificial Neural Network.** [Online].
Available : http://journal.hcu.ac.th/pdffile/5.%20%E0%B9%82%E0%B8%84%E0%B8%3%E0%B8%87%E0%B8%82%E0%B9%88%E0%B8%B2%2073-87.pdf 2009.

[4] **Visual Studio วิชวลสตูดิโอ คืออะไร.** [Online].
Available : http://www.mindphp.com/%E0%B8%84%E0%B8%B9%E0%B9%88%E0%B.8%A1%E0%B8%B7%E0%B8%AD/73-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3/3639-visual-studio-%E0%B8%A7%E0%B8%B4%E0%B8%8A%E0%B8%A7%E0%B8%A5%E0%B8%AA%E0%B8%95%E0%B8%.B9%E0%B8%94%E0%B8%B4%E0%B9%82%E0%B8%AD-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3.html 2016.

[5] **TensorFlow.** [Online]. Available : https://www.tensorflow.org

[6] **Firebase.** [Online]. Available : https://firebase.google.com

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

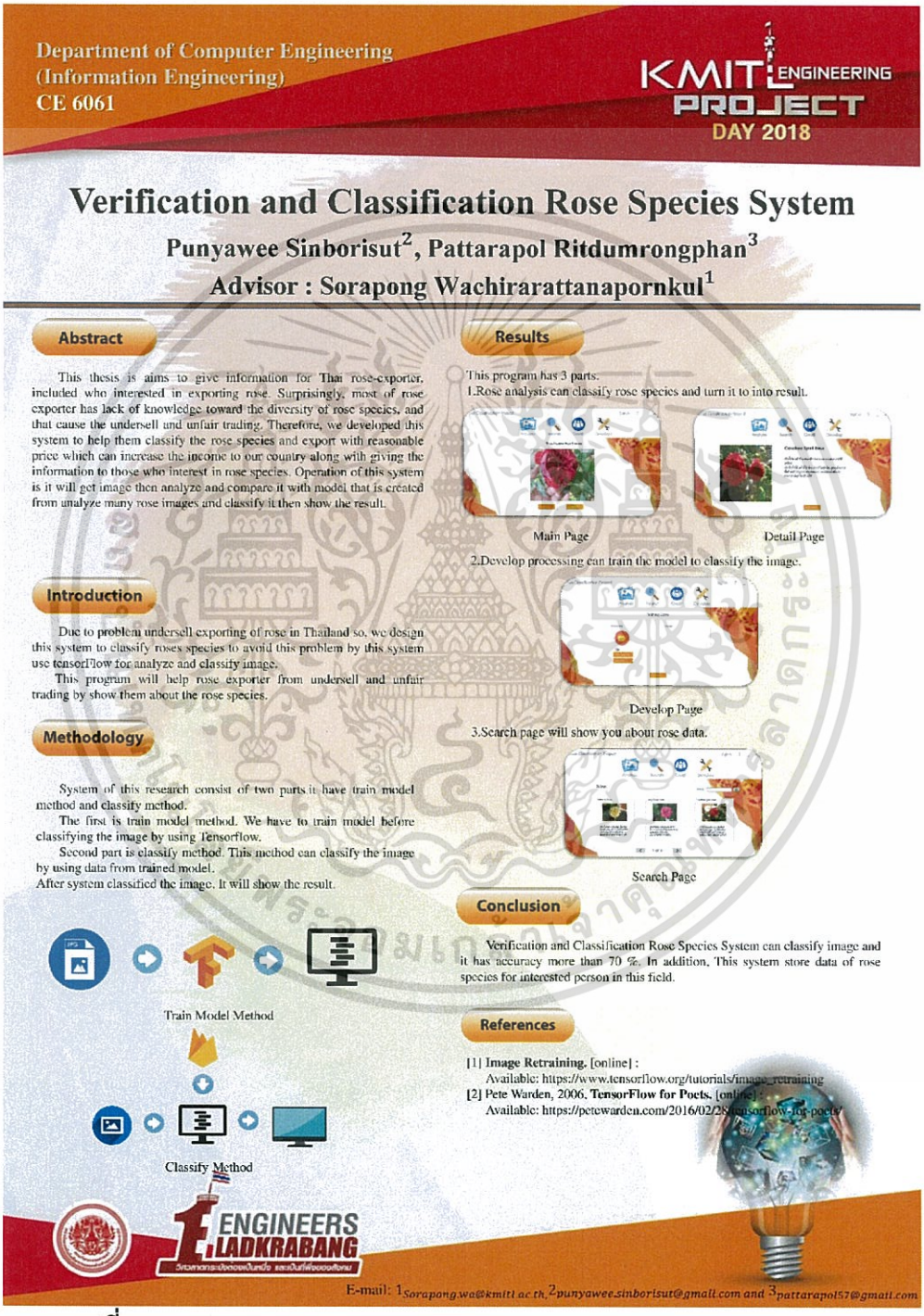
ภาคผนวก ก

Poster (และรูปผลงานถ้ามี)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

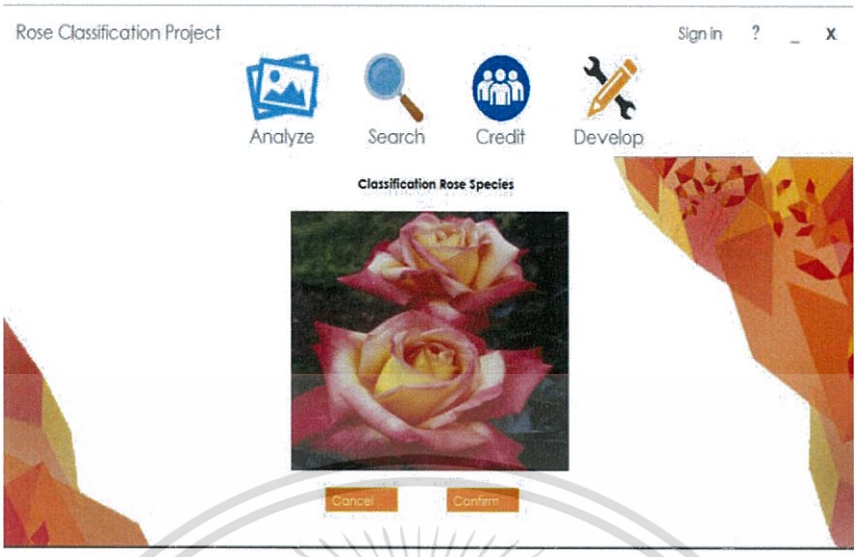
ภาคผนวก ก

Poster (และรูปผลงานถ้ามี)



รูปที่ ก.1 Poster Verification and Classification Rose Species System

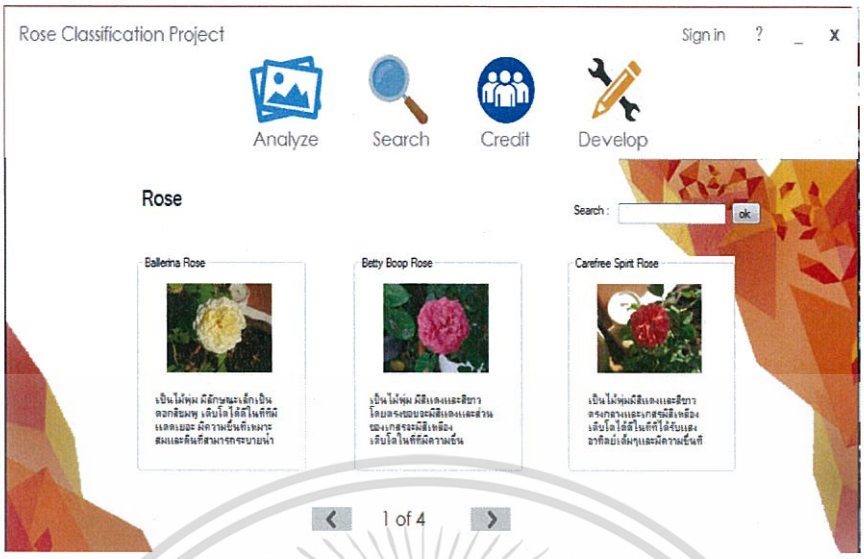
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.2 หน้าหลักของโปรแกรม

รูปที่ ก.3 หน้าแสดงผลลัพธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.4 หน้าแสดงสายพันธุ์กุหลาบทั้งหมดในฐานข้อมูล

รูปที่ ก.5 หน้าสำหรับ train model

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข
ตัวอย่าง การติดตั้งโปรแกรม Visual Studio Community

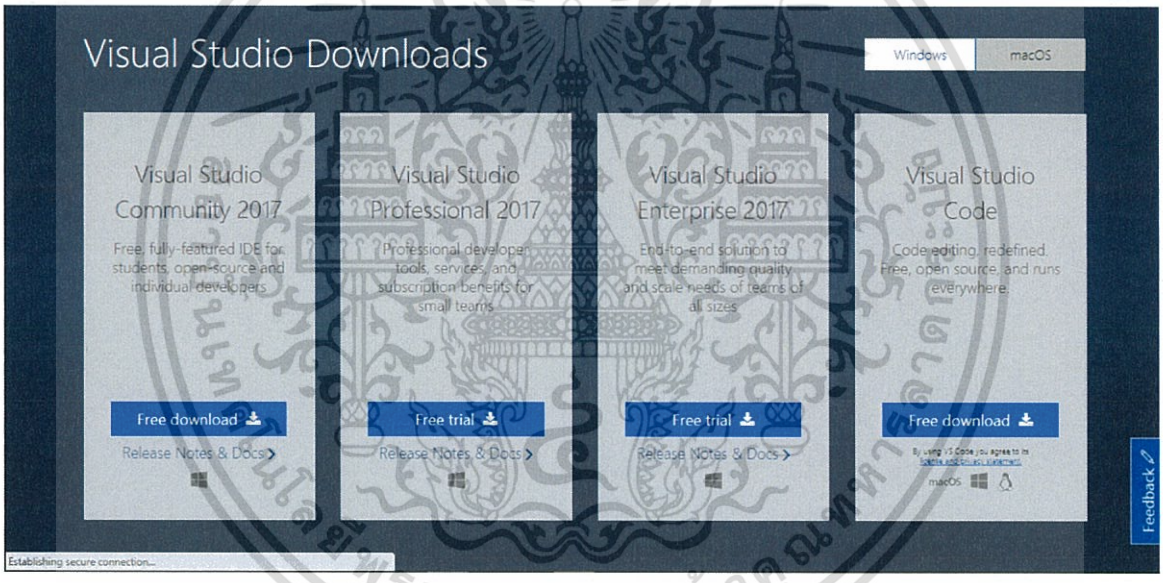
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข
ตัวอย่าง การติดตั้งโปรแกรม Visual Studio Community

การติดตั้งโปรแกรม Visual Studio Community

โปรแกรมบนระบบปฏิบัติการ Windows สามารถอธิบายได้ดังนี้

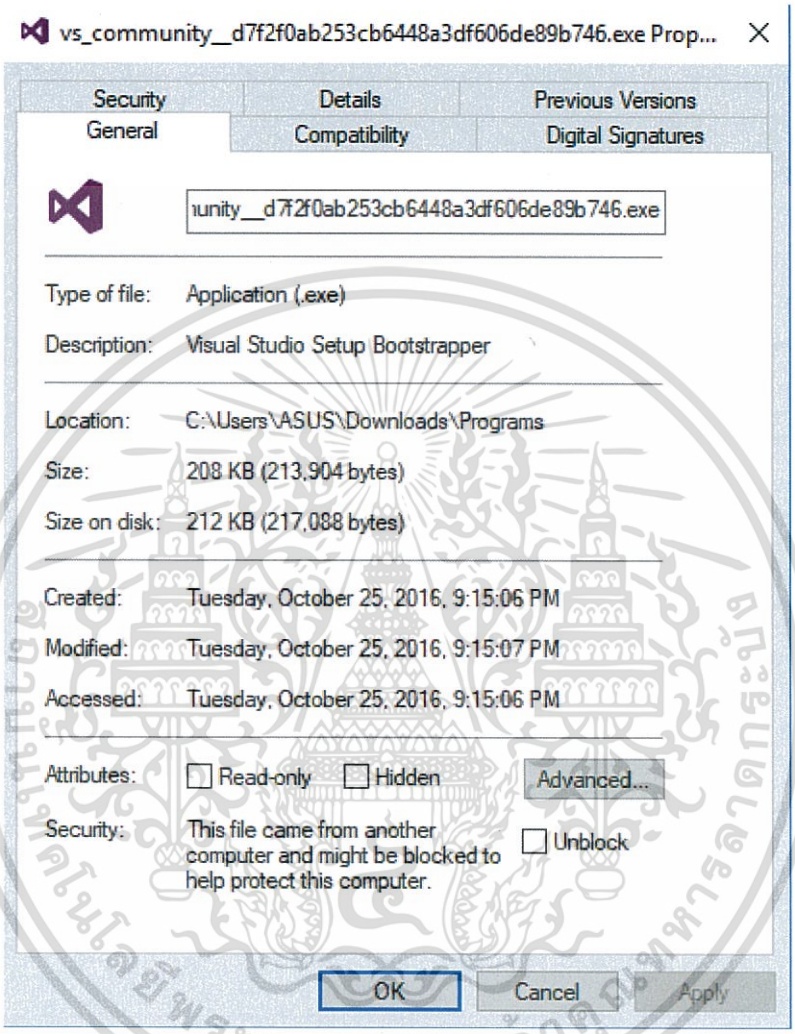
1. ทำการดาวน์โหลด Visual Studio Community ได้จาก https://www.visualstudio.com/downloads/?utm_source=web&utm_medium=documentation&utm_campaign=vs2017upgrade&utm_term=vs2017/ จากนั้นจึงกดปุ่ม Free download ตรง Visual Studio Community 2017



รูปที่ ข.1 เว็บไซต์ดาวน์โหลด Visual Studio

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เมื่อทำการดาวน์โหลด vs_community__d7f2f0ab253cb6448a3df606de89b746.exe เรียบร้อย ไฟล์จะมีขนาดและฟอร์แมตไฟล์ดังรูปที่ ข.2



รูปที่ ข.2 หน้าต่างแสดงขนาดของตัวติดตั้ง Visual Studio Community

3. double-click ไฟล์ vs_community__d7f2f0ab253cb6448a3df606de89b746.exe เพื่อเริ่มต้นติดตั้งโปรแกรม ซึ่งขั้นตอนนี้จำเป็นต้องใช้ Internet ในการติดตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค
ตัวอย่าง การติดตั้ง Python

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค
ตัวอย่าง การติดตั้ง Python

การติดตั้ง Python

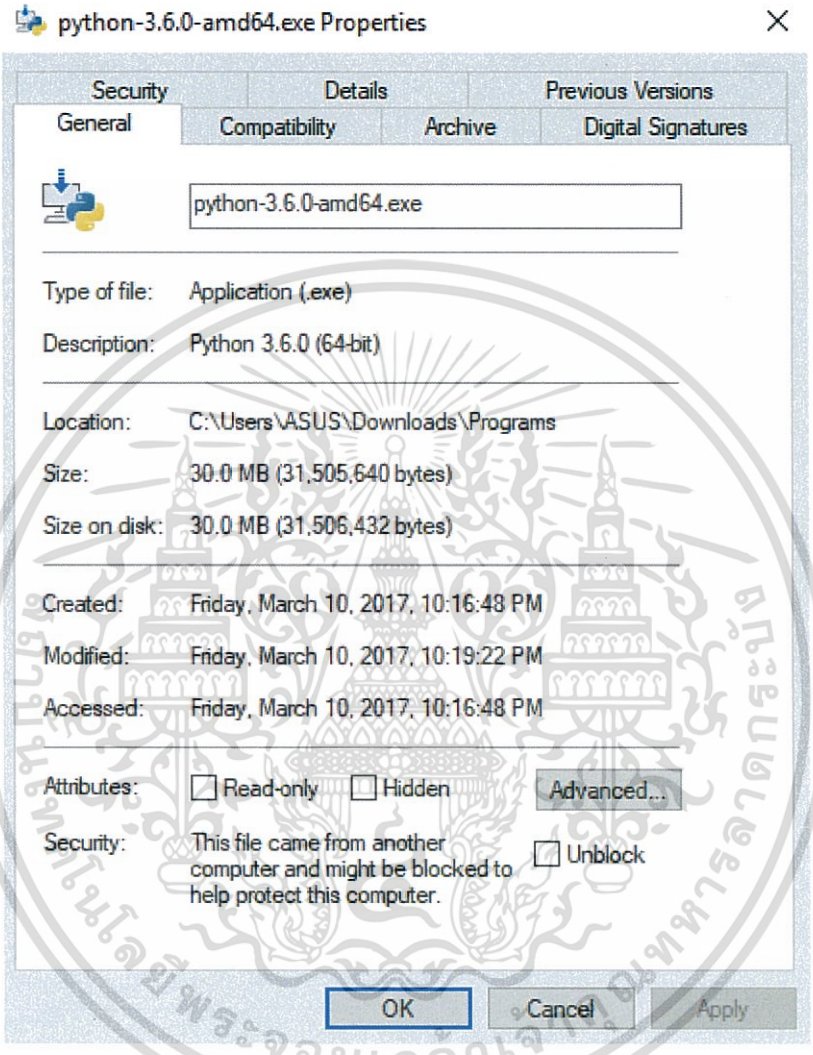
บนระบบปฏิบัติการ Windows สามารถอธิบายได้ดังนี้

1. ทำการดาวน์โหลด Python ได้จาก https://www.python.org/downloads/windows/ แล้วจึงทำการเลือก Version ที่ต้องการ ดาวน์โหลด จากนั้นจึงกด Download Windows x86-64 executable installer

รูปที่ ค.1 หน้าเว็บไซต์แสดงขั้นตอนการดาวน์โหลด Python

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เมื่อทำการดาวน์โหลด python-3.6.0-amd64.exe เรียบร้อย ไฟล์จะมีขนาดและฟอร์แมตไฟล์ดังรูปที่ ค.2



รูปที่ ค.2 หน้าต่างแสดงขนาดของตัวติดตั้ง Python

3. double-click ไฟล์ python-3.6.0-amd64.exe เพื่อเริ่มต้นติดตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง
ตัวอย่าง การติดตั้ง TensorFlow

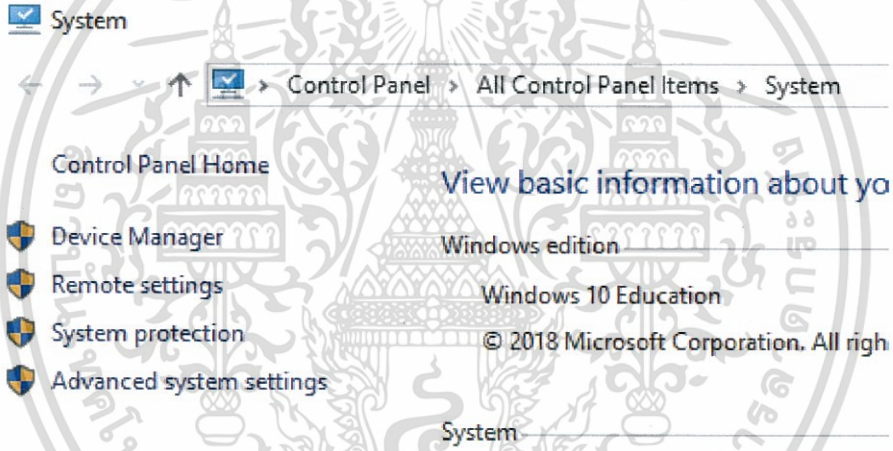
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง
ตัวอย่าง การติดตั้ง TensorFlow

การติดตั้ง TensorFlow

บนระบบปฏิบัติการ Windows สามารถอธิบายได้ดังนี้

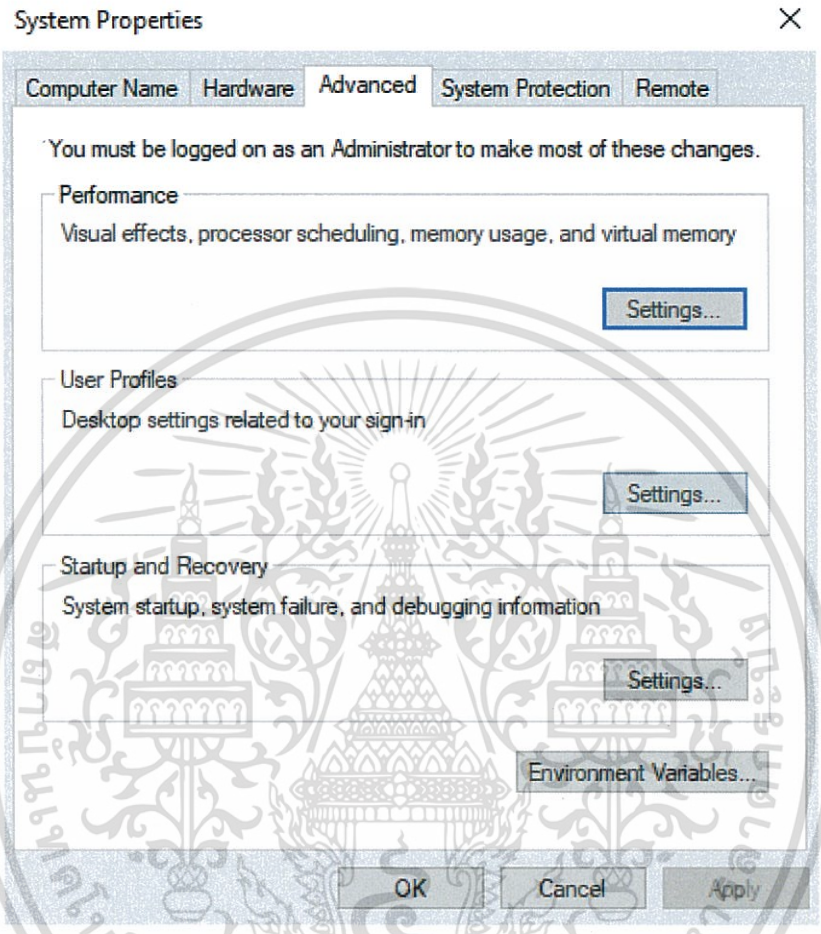
1. ทำการดาวน์โหลด TensorFlow ได้จากการใช้ pip3 ซึ่งคือตัว Package manager ที่มีอยู่ใน Python 3 โดยปัจจุบัน TensorFlow รองรับการทำงานใน Python 3.5.X และ 3.6.x เท่านั้น โดยการจะรัน pip3 ต้องทำการ set path ของตัว Python เพื่อไม่ให้เกิดความยุ่งยากโดยขั้นแรก คลิกขวาที่ My Computer และเลือก Properties และจากนั้นกด Advanced system settings ที่อยู่ทางซ้ายมือ



รูปที่ ง.1 ตำแหน่งของ Advanced system settings

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

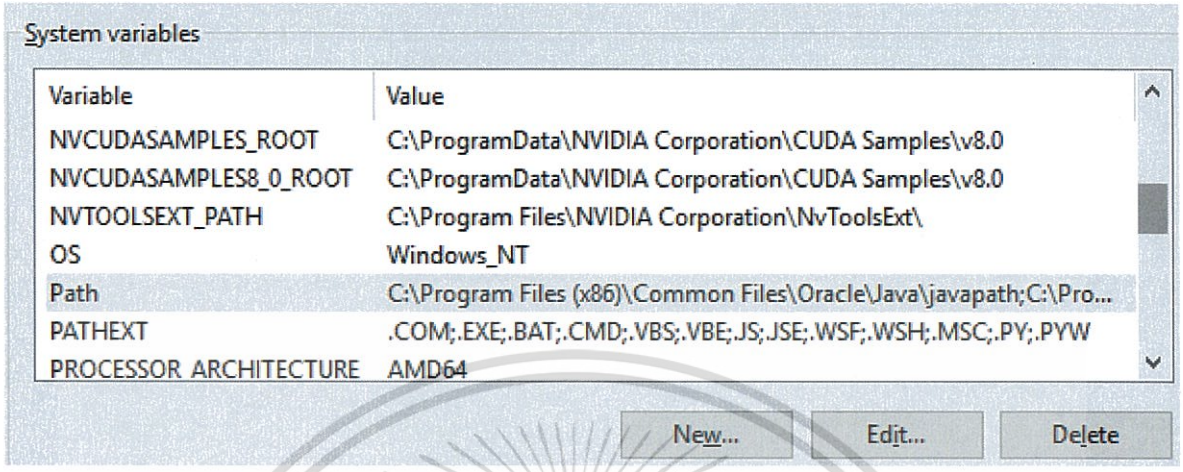
2. เมื่อทำการกดที่ Advanced system settings จะแสดงหน้าต่างดังรูปที่ ง.2 จากนั้นจึงกดที่ Environment Variables



รูปที่ ง.2 หน้า System Properties

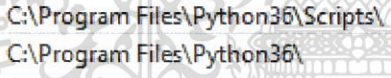
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. จากนั้นไปที่ System Variables และเลือกคำว่า Path กด Edit ดังรูปที่ ง.3



รูปที่ ง.3 หน้าต่างเลือก Path

4. จากนั้นทำการเลือก Path ที่ลง Python ไว้ก่อนหน้านี้ ดังรูปที่ ง.4 จากนั้นกด OK



รูปที่ ง.4 Path ที่ทำการลง Python

5. หลังจาก set path เรียบร้อยให้ทำการเปิด cmd โดยกด windows + R จากนั้นพิมพ์ cmd และพิมพ์คำสั่งตามนี้ pip3 install --upgrade tensorflow จากนั้นกด Enter เพื่อทำการติดตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก จ
ตัวอย่าง การติดตั้งโปรแกรม VNCRS และวิธีการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

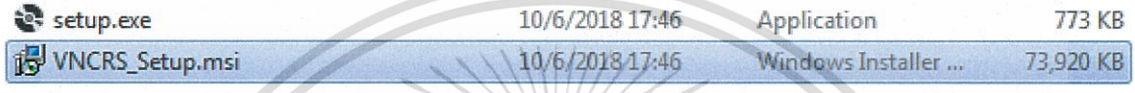
ภาคผนวก จ

ตัวอย่าง การติดตั้งโปรแกรม VNCRS และวิธีการใช้งาน

การติดตั้ง VNCRS

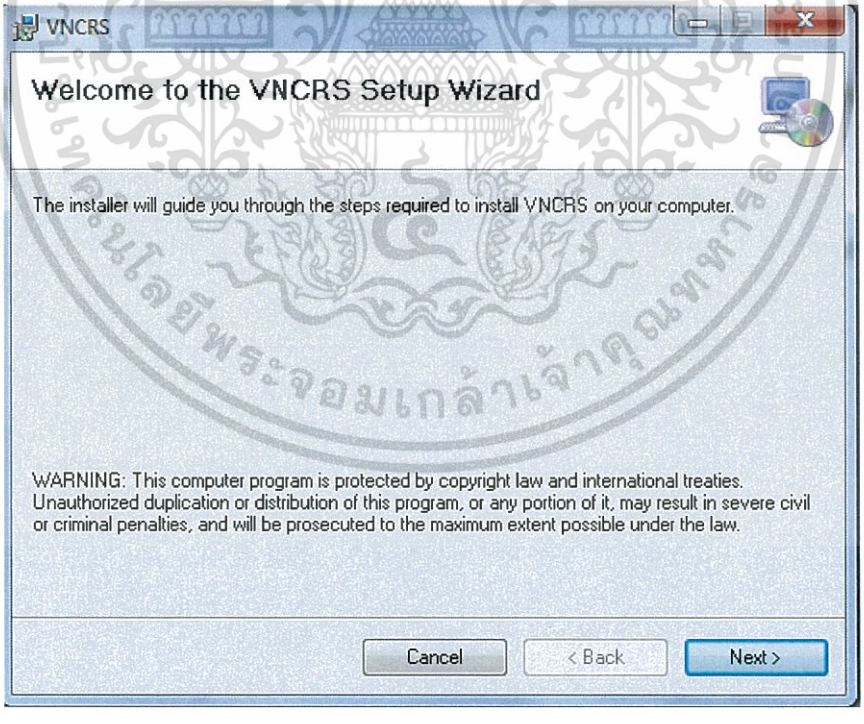
บนระบบปฏิบัติการ Windows สามารถอธิบายได้ดังนี้

1. ทำการดับเบิ้ลคลิกที่ VNCRS_Setup.msi ดังรูปที่ จ.1



รูปที่ จ.1 หน้าติดตั้ง VNCRS

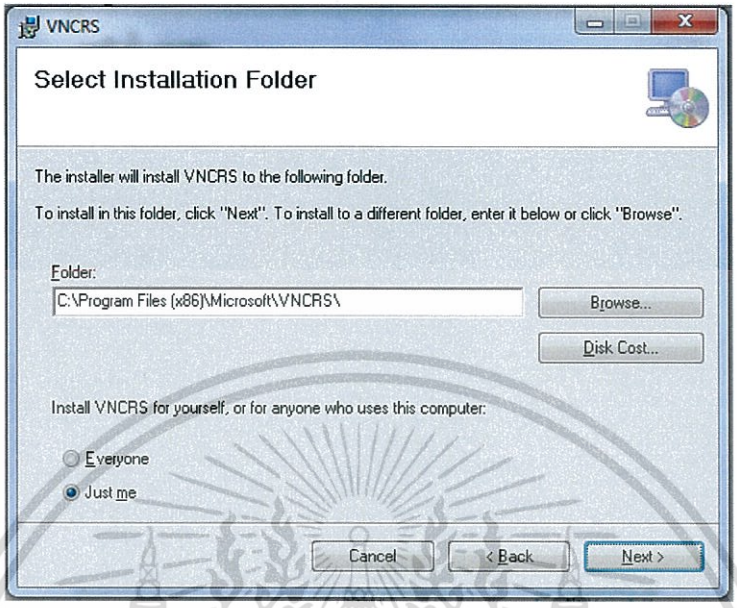
2. กด Next



รูปที่ จ.2 รูปหน้าต่างตัวติดตั้งโปรแกรม

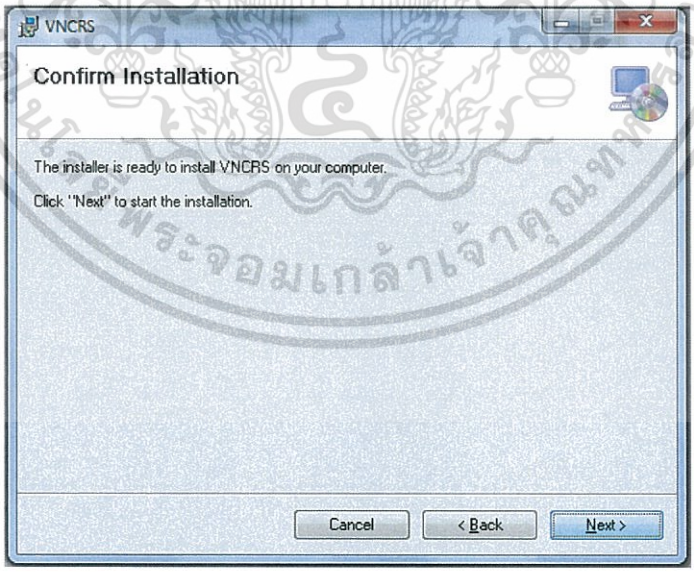
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ทำการเลือก Folder ที่ต้องการจะติดตั้ง แล้วทำการกด Next ดังรูปที่ จ.3



รูปที่ จ.3 รูปหน้าเลือกโฟลเดอร์ที่ต้องการจะติดตั้ง

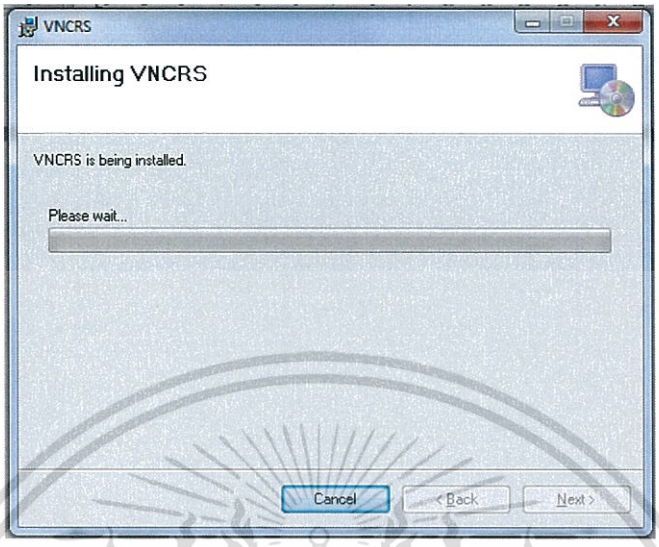
4. กด Next



รูปที่ จ.4 รูปหน้ายืนยันการติดตั้ง

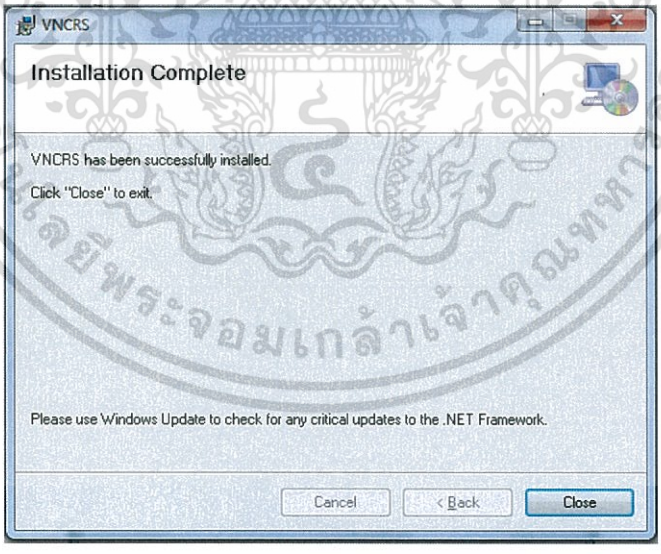
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. โปรแกรมกำลังทำการติดตั้ง



รูปที่ จ.5 รูปหน้ารอการติดตั้ง

6. โปรแกรมทำการติดตั้งเสร็จสิ้น กด Close เพื่อทำการออกจากหน้าติดตั้ง



รูปที่ จ.6 หน้าแสดงการติดตั้งสำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. ตัวโปรแกรมที่ได้หลังจากติดตั้งเสร็จสิ้นจะมีชื่อว่า VNCRS.exe ดังรูปที่ จ.7

System.Xml.ReaderWriter.dll	28/5/2014 4:35	DLL File	21 KB
System.Xml.XDocument.dll	28/5/2014 4:35	DLL File	21 KB
System.Xml.XmlSerializer.dll	28/5/2014 4:35	DLL File	21 KB
Table_1.cs	10/6/2018 17:26	Visual C# Source f...	1 KB
test.cs	12/11/2017 22:22	Visual C# Source f...	1 KB
test1.cs	10/6/2018 17:26	Visual C# Source f...	1 KB
VNCRS.exe	10/6/2018 17:32	Application	17,265 KB
VNCRS.exe.config	2/4/2018 0:05	XML Configuratio...	4 KB
ZedGraph.dll	28/4/2017 11:11	DLL File	300 KB

รูปที่ จ.7 หน้าแสดงตัวโปรแกรม

8. ให้กดคลิกที่ช่องตรงกลางเพื่อทำการเลือกรูปภาพแล้วกดปุ่ม Confirm ดังรูปที่ จ.8

รูปที่ จ.8 หน้าโปรแกรมวิเคราะห์ภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. โปรแกรมจะแสดงผลลัพธ์ออกมา ดังรูปที่ จ.9

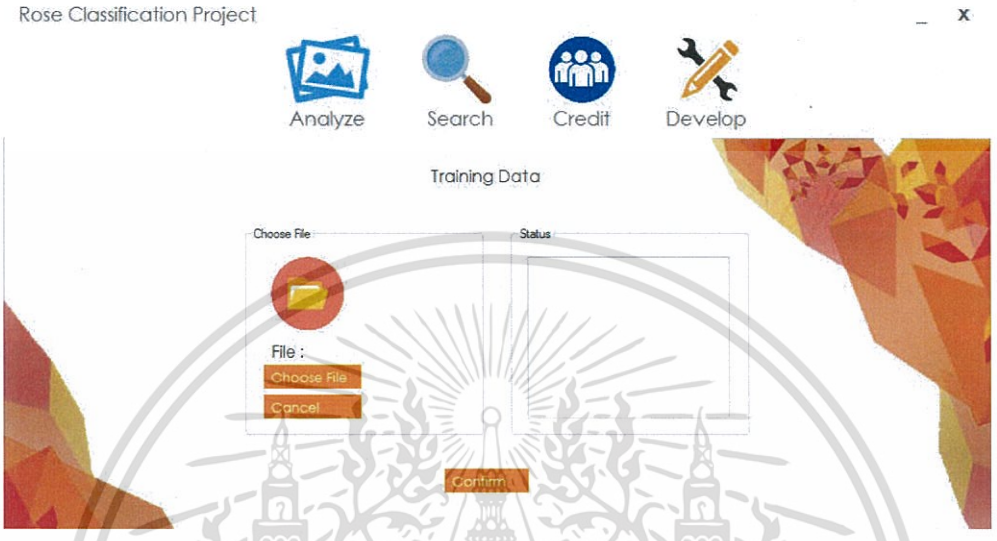
รูปที่ จ.9 หน้าแสดงผลลัพธ์

10. เมื่อทำการกดปุ่ม Search ด้านบนจะเปลี่ยนหน้าเป็นหน้าค้นหาข้อมูล

รูปที่ จ.10 หน้าค้นหาข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. เมื่อกดปุ่ม Develop จะเป็นหน้าที่เอาไว้สำหรับการเทรนข้อมูลให้ทำการกด Choose File แล้วเลือกโฟลเดอร์ที่ประกอบไปด้วยรูปภาพที่ต้องการเทรนแล้วทำการกดปุ่ม Confirm เพื่อเป็นการยืนยัน ดังรูปที่ จ.11



รูปที่ จ.11 หน้าเทรนโมเดล

12. รอจนขึ้นคำว่า Finished เป็นอันเสร็จสิ้นกระบวนการเทรนข้อมูล

รูปที่ จ.12 หน้าแสดงผลลัพธ์เทรนโมเดล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้