

รายงาน
เรื่อง AI สำหรับจำแนกสมุนไพรรักษาโรค
AI for Agriculture

รายชื่อสมาชิกกลุ่ม Hayday

ภรลดา	บุญธรรม	6509490048
ชนินทร์	ศรีสารคาม	6509611627
ธนพนธ์	ศิริวัฒน์	6509611742
นิติกร	คณะแพ่ง	6509611825
แพรวา	สุรางค์วัฒนกุล	6509611924
ศรวณ	เจริญวัฒน์	6509612039

เสนอ

อาจารย์ธนาธร ทะนานทอง

สารบัญ

เรื่อง	หน้า
บทที่ 1 บทนำ	
1.1 ที่มาและความสำคัญ	1
1.2 วัตถุประสงค์	1
1.3 ประโยชน์ของโครงการ	1
บทที่ 2 งานวิจัยที่เกี่ยวข้อง	
2.1 ทฤษฎีที่ใช้	3
2.2 งานวิจัยที่เกี่ยวข้อง	4
2.3 ตัวอย่าง Application ที่เกี่ยวข้องเพื่อเปรียบเทียบ	7
บทที่ 3 วิธีการจัดทำโครงการ	
3.1 Problem Framing	9
3.2 การเก็บข้อมูลและการเตรียมข้อมูล	9
3.3 การพัฒนาโมเดลการเรียนรู้เชิงลึกเพื่อวิเคราะห์และจำแนกใบสมุนไพรมะนาว	12
3.4 การออกแบบ UX/UI ของ Application	20
บทที่ 4 ผลการศึกษา	
4.1 ระบบและเฟรมเวิร์กที่ใช้ในการทดลอง	23
4.2 ประสิทธิภาพของ model เป้าหมาย	23
4.3 การประเมินโมเดล	23
4.4 ผลการทดลอง	23
4.5 อภิปรายผลการทดลอง	28
บทที่ 5 สรุปผล และข้อเสนอแนะ	
5.1 สรุปผล	29
5.2 ข้อเสนอแนะในการนำผลงานวิจัยไปใช้ประโยชน์	29
ภาคผนวก	31
บรรณานุกรม	34

บทที่ 1

บทนำ

ที่มาและความสำคัญของโครงการ

ประเทศไทยมีพืชสมุนไพรพื้นบ้านมากมายที่ถูกนำมาใช้ในการรักษาโรคและบำรุงสุขภาพตามภูมิปัญญาดั้งเดิม อย่างไรก็ตาม ปัญหาที่มักพบในการใช้พืชสมุนไพรคือเกิดความสับสนในการแยกพืชสมุนไพร จึงเป็นเหตุให้นำไปใช้อย่างผิดประเภท มักเกิดมาจากคล้ายคลึงกันของใบพืชสมุนไพร ซึ่งจำเป็นต้องอาศัยความเชี่ยวชาญและประสบการณ์ในการจำแนก ในการลดความผิดพลาดการใช้สมุนไพรไทย การระบุชนิดของสมุนไพรอาจเป็นเรื่องยากสำหรับคนทั่วไปที่ขาดความรู้ด้านพฤกษศาสตร์ ซึ่งปัญหานี้อาจส่งผลให้เกิดความเข้าใจผิดและการใช้สมุนไพรที่ผิดประเภท จนอาจก่อให้เกิดอันตรายต่อสุขภาพได้

โครงการนี้จึงถูกคิดค้นขึ้นเพื่อพัฒนาระบบปัญญาประดิษฐ์ (AI) ที่สามารถจำแนกและระบุชนิดของสมุนไพรไทยที่มีสรรพคุณในการรักษาโรคได้อย่างแม่นยำ ด้วยการนำเทคนิคการประมวลผลภาพของใบพืชมาประยุกต์ใช้โดยมีลักษณะรูปร่าง ขนาด สี สัน เส้นใบ และเนื้อสัมผัสที่แตกต่างกันในแต่ละชนิด ดังนั้นการนำเทคนิคการประมวลผลภาพมาใช้ในการจำแนกลักษณะเฉพาะของใบพืชแต่ละชนิดจึงเป็นวิธีที่มีประสิทธิภาพ โดยระบบ AI จะถูกฝึกฝนด้วยข้อมูลภาพใบพืชจำนวนมากเพื่อให้สามารถเรียนรู้และจดจำลักษณะเฉพาะของแต่ละชนิดได้อย่างถูกต้องเพื่ออนุรักษ์และเผยแพร่องค์ความรู้ด้านสมุนไพรไทย ช่วยให้ประชาชนสามารถใช้สมุนไพรในการดูแลสุขภาพอย่างถูกวิธีและปลอดภัย นอกจากนี้ยังเป็นการนำนวัตกรรมด้านปัญญาประดิษฐ์และการประมวลผลภาพมาประยุกต์ใช้กับภูมิปัญญาท้องถิ่น สร้างมูลค่าเพิ่มและความน่าเชื่อถือของผลิตภัณฑ์สมุนไพรไทย พร้อมทั้งส่งเสริมการพัฒนาเทคโนโลยีด้านสุขภาพและการแพทย์ให้ก้าวหน้ามากยิ่งขึ้น

วัตถุประสงค์

1. เพื่อพัฒนาโมเดลการจำแนกประเภทของสมุนไพรไทย
2. เพื่อลดเวลาที่ใช้ในการคัดแยกสมุนไพรไทยโดยใช้เทคโนโลยี image processing
3. เพื่อส่งเสริมการใช้ประโยชน์จากสมุนไพรไทย

ประโยชน์ของโครงการ

1. ทำให้มีการใช้สมุนไพรไทยอย่างปลอดภัย ป้องกันการนำสมุนไพรไปใช้ผิดประเภท
2. ทำให้กระบวนการคัดแยกประเภทของสมุนไพรไทยรวดเร็วและมีประสิทธิภาพมากยิ่งขึ้น
3. ให้ข้อมูลที่แม่นยำและเชื่อถือได้เกี่ยวกับการใช้ประโยชน์จากสมุนไพรไทยในการรักษาโรค
4. ลดปัญหาที่อาจเกิดขึ้นจากความไม่เชี่ยวชาญของผู้แยก โดยการใช้เทคโนโลยีที่ช่วยในการแยกแยะสมุนไพรไทยอย่างถูกต้อง และลดความผิดพลาดที่อาจเกิดขึ้นจากการประเมินโดยมนุษย์
5. เสริมความเชื่อมั่นให้กับสมุนไพรไทยในวงการแพทย์แผนไทยและวงการที่เกี่ยวข้อง โดยการพัฒนาโมเดล AI ที่สามารถช่วยจำแนกแยกสมุนไพรไทยได้อย่างแม่นยำและเชื่อถือได้

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

สมุนไพรไทยมีคุณประโยชน์ต่อสุขภาพและวิถีชีวิตคนไทยมาช้านานเนื่องจากมีประโยชน์หลากหลาย ด้าน อย่างเช่น ด้านการรักษาโรคและบรรเทาอาการเครียดหรืออาการนอนไม่หลับ เพราะในยุคปัจจุบันที่มีความเครียดและความกดดันสูง การใช้สมุนไพรเพื่อช่วยผ่อนคลายเป็นวิธีการที่ได้รับความนิยมมากขึ้น โดยเฉพาะสมุนไพรไทยที่หาได้ง่ายในท้องถิ่น อย่างเช่น ฟักทะลายโจร โหระพา และกะเพรา ซึ่งพืชเหล่านี้เป็นพืชสมุนไพรที่พบว่ามีสารสำคัญ ได้แก่ apigenin, luteolin และ rosmarinic acid ซึ่งมีฤทธิ์ต้านอนุมูลอิสระ ลดการอักเสบ และมีฤทธิ์คลายสารต้านเครียด ช่วยให้อารมณ์ผ่อนคลายและลดอาการเครียดลงได้ (รัชฎาพร อุ่นศิริไธย์ และ อนามัย เทศกะทีก, 2561, น.3) นอกจากนี้ มะกรูดก็เป็นอีกหนึ่งสมุนไพรที่มีประโยชน์ โดยพบว่ามะกรูดมีผลช่วยลดระดับคอร์ติซอล ซึ่งเป็นฮอร์โมนเครียด ทำให้อารมณ์ผ่อนคลายมากขึ้น จึงช่วยให้นอนหลับง่ายและมีคุณภาพการนอนที่ดีขึ้น (Rahmat and Mohammad, 2022) ยิ่งไปกว่านั้น น้ำมันหอมระเหยจากสารสกัดสะระแหน่และตะไคร้ ก็มีฤทธิ์ผ่อนคลายความตึงเครียดและช่วยให้นอนหลับได้ดีขึ้นเช่นกัน เนื่องจากกลิ่นหอมมีผลต่อระบบประสาททำให้รู้สึกผ่อนคลายและสงบนิ่ง ดังนั้น สมุนไพรเหล่านี้จึงเป็นตัวช่วยที่ดีในการบรรเทาอาการเครียดได้อย่างเป็นธรรมชาติ โดยใช้ประโยชน์จากสารสำคัญในสมุนไพรซึ่งมีคุณสมบัติเฉพาะตัวในการช่วยให้อารมณ์ผ่อนคลาย นอนหลับสนิท จนส่งผลดีต่อสุขภาพโดยรวม (Julien Cases et al., 2010) นอกจากนี้พืชสมุนไพรทั้ง 6 ยังมีสรรพคุณและประโยชน์ต่อสุขภาพในด้านอื่น ๆ อีกเช่น

- กะเพราสามารถบรรเทาความเครียดทางกายภาพเคมีเมตาบอลิซึม และจิตใจ มีสารต้านอนุมูลอิสระหลายชนิด เช่น วิตามินซี เบต้าแคโรทีน สารฟลาโวนอยด์ และสารฟีนอลิกหลายชนิด
- โหระพามีสรรพคุณช่วยลดโอกาสการเกิดโรคหลอดเลือดสมอง โดยสารสกัดจากโหระพามีคุณสมบัติต้านแบคทีเรีย ต้านเชื้อราและไวรัสได้บางชนิดสามารถต้านอนุมูลอิสระ ลดระดับไขมันเลว และเพิ่มระดับไขมันดีในเลือด ฯ
- ฟักทะลายโจรสามารถใช้ใบในการรักษาอาการไข้ ไอ และท้องเสีย
- มะกรูดมีสารต้านอนุมูลอิสระ มีความสามารถในการลดการเกิดอาหารอักเสบในหลอดเลือด ช่วยขับลมในลำไส้ แก้อาการท้องอืด ช่วยป้องกันรังแคทำให้เส้นผมดกดำเป็นเงางามและช่วยลดความเสี่ยงต่อโรคหัวใจ
- สะระแหน่ มีคุณสมบัติทำให้รู้สึกสงบ สามารถรับประทานได้ในชา ช่วยต้านไวรัส ต้านการอักเสบ และต้านอนุมูลอิสระ ช่วยรักษาโรคที่เกี่ยวข้องกับอาการอักเสบและปวด สามารถบรรเทาผลที่เกี่ยวข้องกับความเครียด ดูแลการย่อยอาหารและลำไส้ให้ดีขึ้น รวมถึงยังมีสารต้านอนุมูลอิสระเป็นจำนวนมาก

- ตะไคร้ มีคุณสมบัติทางเคมีที่ช่วยลดการอักเสบและฆ่าเชื้อโรคได้น้ำมันหอมระเหยจากตะไคร้ เช่น citral, geranial และ neral มีฤทธิ์ต้านทานจุลินทรีย์ที่มีประสิทธิภาพ สามารถลดการเจริญเติบโตของเซลล์มะเร็งได้

2.1 ทฤษฎีที่ใช้

ทฤษฎีที่ 1 Center Crop Object Detection

การตัดส่วนกลางของภาพออกมา (Center Cropping) จะตัดเอาเฉพาะส่วนกลางของภาพออกมา โดยสมมติว่าวัตถุที่สนใจจะปรากฏอยู่ในบริเวณกลางของภาพเป็นหลัก การตัดส่วนกลางจะทำให้ภาพมีขนาดเล็กลง ซึ่งจะช่วยลดภาระการประมวลผลได้มาก จุดเด่นของวิธีนี้คือ ความง่ายและประสิทธิภาพในการประมวลผล เนื่องจากเราได้ลดขนาดของปัญหาโดยการ Crop ภาพจากนั้นสามารถนำข้อมูลพื้นที่นี้ไปใช้ในการแยกแยะและจำแนกประเภทของใบพืช ซึ่งเป็นข้อมูลที่สามารถนำมาวิเคราะห์และใช้ในการสร้างโมเดลการเรียนรู้ของเครื่องเพื่อจำแนกพืชสมุนไพรได้

ทฤษฎีที่ 2 Image processing

เป็นกระบวนการที่ใช้การประมวลผลเพื่อดำเนินการกับภาพในการวิเคราะห์ ปรับปรุง หรือการแก้ไขภาพ เช่น การตรวจจับวัตถุ การตรวจสอบคุณภาพ การเพิ่มความชัดเจน เป็นต้น การนำทฤษฎี Image processing มาปรับใช้ในโครงการการจำแนกใบพืชสมุนไพรในกระบวนการประมวลผลภาพใบพืช เพื่อให้สามารถระบุและติดตามตำแหน่งของใบพืชในภาพ โดยสามารถปรับขนาดภาพ ปรับแก้ หรือดำเนินการอื่น ๆ เพื่อเพิ่มความชัดเจนให้กับภาพ และช่วยในกระบวนการตรวจจับและจำแนกประเภทของใบพืช

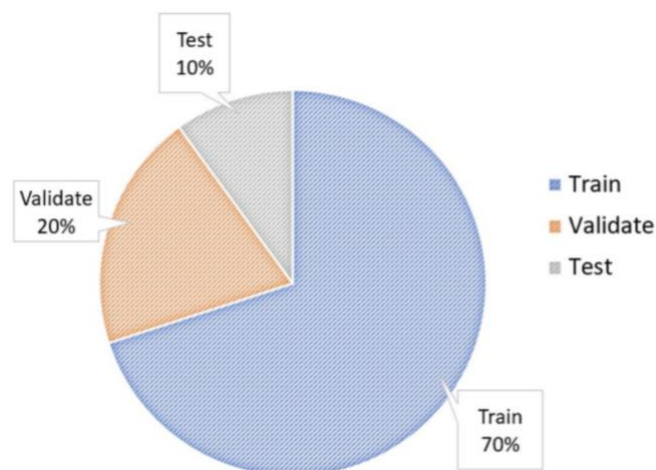
ทฤษฎีที่ 3 Predict image classification

การทำนายหรือจัดหมวดหมู่รูปภาพโดยอัตโนมัติในรูปแบบที่สอดคล้องกับการใช้สถาปัตยกรรม 4 ตัว คือ MobileNetV3Small, ResNet-50, EfficientNetV2L และ NasNetMobile ในการ "การจำแนกภาพ" หรือ "การจำแนกวัตถุ" โดยเราใช้การเรียนรู้ของเครื่องเพื่อสร้างโมเดลที่สามารถระบุหรือจัดหมวดหมู่วัตถุในรูปภาพเป็นคลาสที่กำหนดไว้ล่วงหน้า โดยโมเดลที่ถูกฝึกสอนด้วยชุดข้อมูลรูปภาพที่มีป้ายชื่อแล้วจะสามารถทำนายหรือจัดหมวดหมู่รูปภาพที่ไม่เคยเห็นมาก่อนได้โดยอัตโนมัติ การนำทฤษฎี Predictive Image Classification มาปรับใช้ในโครงการการจำแนกใบพืชสมุนไพร นำมาใช้เพื่อสร้างโมเดลที่สามารถจำแนกประเภทของใบพืชสมุนไพรจากรูปภาพ โดยระบบจะถูกฝึกสอนด้วยชุดข้อมูลรูปภาพของใบพืชที่มีป้ายชื่อแต่ละชนิด จากนั้นระบบสามารถทำนายหรือจำแนกประเภทของใบพืชในภาพที่ไม่เคยเห็นมาก่อนได้อย่างแม่นยำ

2.2 งานวิจัยที่เกี่ยวข้อง

งานวิจัยที่ 1 Medicinal Plant Identification in Real-Time Using Deep Learning Model

การวิจัยเกี่ยวกับการจำแนกใบของพืชสมุนไพร 6 ชนิด ซึ่งประกอบด้วย ใบพลู, ใบแกง, ใบโหระพา, ใบสะระแหน่, ใบสะเดา, และ ใบหยินน้ำ ในรูปแบบ real-time ผ่านโทรศัพท์มือถือโดยใช้เทคโนโลยีการประมวลผลภาพ (Image Processing) โดยนำรูปภาพที่ถ่ายของใบพืชสมุนไพรมาเก็บรวบรวมเพื่อสร้าง Dataset ซึ่งประกอบด้วยข้อมูลรูปภาพสมุนไพรทั้งหมด 6 ชนิด แต่ละชนิดมีข้อมูลรูปภาพจำนวน 500 รูป จากนั้นนำรูปภาพจำนวน 350 รูปต่อสมุนไพรแต่ละชนิดไปใช้สำหรับการฝึกโมเดล และใช้รูป 100 รูปต่อชนิดสำหรับการตรวจสอบความถูกต้องของโมเดล และใช้รูป 50 รูปสำหรับการทดสอบโมเดล โดยแบ่งสัดส่วนการใช้ภาพเป็น 70% สำหรับการฝึกโมเดล, 20% สำหรับการตรวจสอบความถูกต้องของโมเดล และ 10% สำหรับการทดลองโมเดล หลังจากนั้นนำ Dataset ที่ได้ไปเข้าสู่กระบวนการปรับเปลี่ยนขนาดและเพิ่มข้อมูลรูปภาพ (Data Augmentation) โดยการปรับขนาดของรูปทั้งหมดใน Dataset ให้มีขนาด 224x224 ซึ่งเป็นขนาดที่เหมาะสมกับการใช้โมเดล MobileNet หลังจากนั้นส่งไปประมวลผลที่โมเดล MobileNet ที่ถูก deploy ไว้บน cloud (GPC) จะแสดงข้อมูลชื่อและสรรพคุณกลับมาและแสดงบนแอปพลิเคชันบนมือถือ



ภาพแสดงสัดส่วน Dataset จากงานวิจัยที่เกี่ยวข้อง

งานวิจัยที่ 2 Plant phenomics: image acquisition technologies and data analysis algorithms

เป็นการศึกษาลักษณะทางพันธุกรรมของพืชที่แสดงออกมาทางกายภาพ ซึ่งมีประโยชน์ในการปรับปรุงพันธุ์พืชให้มีคุณสมบัติตามต้องการ เทคโนโลยีการรับภาพและการประมวลผลภาพได้เข้ามามีบทบาทสำคัญในการศึกษาฟีโนมิกส์ของพืชโดยใช้ระบบการรับภาพที่หลากหลายและภาพที่ได้จากระบบเหล่านี้จะต้องผ่านกระบวนการประมวลผลก่อนนำไปวิเคราะห์ โดยการประมวลผลภาพประกอบด้วยการปรับปรุงคุณภาพภาพ, การแบ่งส่วนภาพเพื่อแยกวัตถุที่สนใจออกจากพื้นหลัง, การสกัดคุณลักษณะจากภาพ และการประยุกต์ใช้เทคนิคการเรียนรู้ของเครื่องเพื่อจำแนกและจัดกลุ่มข้อมูล แต่ก็ยังมีข้อจำกัดและความท้าทายบาง

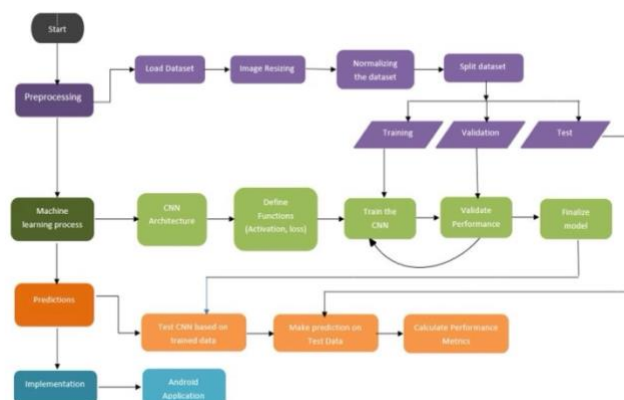
ประการ อาทิเช่น การประมวลผลข้อมูลจำนวนมากเพื่อให้ข้อมูลที่ถูกต้องและตรงตามวัตถุประสงค์ การพัฒนาระบบสำหรับการศึกษานอกพื้นที่ที่มีสิ่งแวดล้อมหลากหลายและไม่สามารถควบคุมได้ การพัฒนามาตรฐานในการเก็บข้อมูลภาพและการประมวลผลเพื่อให้ข้อมูลมีความถูกต้องและสามารถนำไปใช้ได้



ภาพแสดงการทำ Center Crop Object Detection

งานวิจัยที่ 3 Automatic Recognition of Medicinal Plants Based on Multispectral and Texture Features using Hidden Deep Learning Model

งานวิจัยนี้มุ่งเน้นการระบุพืชสมุนไพรโดยใช้การเรียนรู้เชิงลึกโดยใช้โมเดล Convolutional Neural Networks (CNN) เพื่อระบุพืชสมุนไพรผ่านใบพืชอย่างแม่นยำ โดยใช้ภาพของใบพืชที่ได้มาจากธรรมชาติเป็นชุดข้อมูลการทดลอง ผู้วิจัยเก็บรวบรวมภาพใบพืชจากพืชสมุนไพร 5 ชนิดต่าง ๆ หลังจากการรวบรวมภาพและต้องทำการดำเนินการประมวลผลเบื้องต้นก่อนที่จะทำการแบ่งประเภทภาพ เป็นชุดข้อมูลสำหรับการฝึกและทดสอบ งานวิจัยนี้เป็นการพัฒนาระบบตรวจจับพืชสมุนไพรโดยอัตโนมัติที่สามารถมีผลกระทบต่อการใช้งานในภาคการแพทย์ได้อย่างเหมาะสม และสามารถเรียนรู้เพิ่มเติมเกี่ยวกับพืชสมุนไพรในอนาคต โดยงานวิจัยนี้ใช้ Deep Learning โมเดลต่าง ๆ ในการฝึกและทดสอบ เพื่อการจำแนกประเภทของใบพืชสมุนไพรได้อย่างแม่นยำ โดยรวมแล้วมีโมเดลดังนี้ VGG16, VGG19, Inception-V3, Xception, MobileNetDenseNet121, ResNet50 โมเดลทั้งหมดถูกฝึกด้วยการใช้ Transfer Learning บนชุดข้อมูลใบพืชสมุนไพรที่มีเพื่อการจำแนกประเภทของภาพใบพืชได้อย่างแม่นยำ โดยมีการทดสอบความแม่นยำของแต่ละโมเดลบนชุดข้อมูลทดสอบเพื่อประเมินประสิทธิภาพของโมเดลที่ใช้ในงานวิจัยนี้



ภาพแสดง Process Flowchart

งานวิจัยที่4 Plant Disease Diagnostic System Using NASNet-Mobile Deep Learning

งานวิจัยนี้ได้บรรยายถึงการใช้โครงสร้างการเรียนรู้เชิงลึกปัจจุบันชื่อ NASNet ที่ใช้พารามิเตอร์ที่มีประสิทธิภาพบนชุดข้อมูลที่เฉพาะเจาะจงมาก ซึ่งสามารถวินิจฉัยโรคพืชได้อย่างมีประสิทธิภาพด้วยแอปพลิเคชันบนโทรศัพท์มือถือที่สร้างขึ้นมา แอปพลิเคชันใช้โมเดล NASNet-Mobile โดยโมเดลจะถูกนำไปใช้งานผ่าน flask บนเครื่อง EC2 ของ Amazon และทำการสื่อสารผ่านบริการไมโครเซอร์วิสเพื่อวินิจฉัยโรคโดยใช้ภาพใบพืชที่ถูกถ่ายทำจากแอปพลิเคชันบนโทรศัพท์มือถือ โมเดลที่พัฒนาขึ้นได้รับผลลัพธ์ที่น่าพอใจที่มีค่าเท่ากับ 99.24%, 99.28%, 99.31%, และ 99.28% อย่างได้เสียเปรียบ อย่างไรก็ตาม ช่วงเวลารอคอยการตอบสนองบน API ของไมโครเซอร์วิสที่ให้บริการวินิจฉัยดี แต่จะช้าลงเมื่อขนาดของภาพใหญ่ขึ้น และแอปพลิเคชันบนโทรศัพท์มือถือที่พัฒนาขึ้นสำหรับการวินิจฉัยโรคพืชจากระยะไกล

Batch Size	Training Loss	Validation Loss	Accuracy
8	0.251792	0.182867	0.953043
16	0.171711	0.120710	0.964184
32	0.151575	0.099513	0.965749
64	0.154251	0.110910	0.962619

ตารางแสดงการเปรียบเทียบ Batch Size กับ Accuracy ในสถาปัตยกรรม NASNet-Mobile

2.3 ตัวอย่าง Application ที่เกี่ยวข้องเพื่อเปรียบเทียบ

การเปรียบเทียบ Feature เพื่อการวิเคราะห์ความสำคัญของ Features แอปพลิเคชัน โดยอาศัยข้อมูลจากรีวิวหรือความคิดเห็นของผู้ใช้งาน หาก Features ที่ได้รับคะแนนสูงหรือถูกกล่าวถึงบ่อยมักจะเป็นสิ่งที่ผู้ใช้ให้ความสำคัญและนำข้อมูลที่ได้ไปเปรียบเทียบกับแอปพลิเคชันที่มีอยู่แล้วในท้องตลาดเพื่อหา Feature ที่สำคัญ จากข้อมูลวิเคราะห์ไปใช้ในการสร้างแอปพลิเคชันต่อผู้ใช้

โดย Feature ของแอปพลิเคชันทั้ง 6 แอปพลิเคชันที่นำมาเปรียบเทียบทั้งหมดจะมี การจำแนกพืชด้วยรูปภาพ, สรรพคุณและผลข้างเคียง, Community แลกเปลี่ยนความรู้, ค้นหาด้วยชื่อ, คุณลักษณะเด่นของพืช และข้อมูลการดูแลพืช มาเปรียบเทียบกับยอดคะแนนการรีวิว เพื่อหา Feature ที่สำคัญ ในการสร้างแอปพลิเคชัน AI สำหรับจำแนกสมุนไพรที่ใช้รักษาโรค

แอปพลิเคชัน	จำแนกพืชด้วยรูปภาพ	สรรพคุณและผลข้างเคียง	Community แลกเปลี่ยนความรู้	ค้นหาด้วยชื่อ	คุณลักษณะเด่นของพืช	ข้อมูลการดูแลพืช	ยอดการรีวิว
Greg – Plan Care	✓	X	✓	X	✓	✓	4.9/5 จากจำนวน 7 รีวิว
Plant Identifier - PlantMe	✓	X	✓	✓	X	✓	4.8/5 จากจำนวน 225 รีวิว
Plantum – AI Plant Identifier	✓	X	✓	✓	✓	✓	4.7/5 จากจำนวน 1,500 รีวิว
AI Plant Identifier App - PLNT	✓	X	✓	✓	✓	✓	4.6/5 จากจำนวน 44 รีวิว
Herb ID	✓	✓	X	✓	X	X	3.3/5 จากจำนวน 8 รีวิว
Medicinal plants: herbs, bark	X	✓	✓	✓	X	X	ไม่มีรีวิว

ตารางแสดงการเปรียบเทียบระหว่าง Feature กับ ยอดการรีวิว

หลังจากการเปรียบเทียบ Feature กับแอปพลิเคชันที่มีความคล้ายคลึงกับโครงการ จะพบว่าแอปพลิเคชันที่มี Feature บอกข้อมูลการดูแลพืช และแอปพลิเคชันที่ใช้งานง่ายมี Feature ที่ช่วยเสริมความสะดวกสบายให้แก่ผู้ใช้งาน (User Friendly) จะถูกพูดถึงและมีความสำคัญต่อการรีวิว ทางผู้จัดทำโครงการจึงเลือก Feature ที่บอกข้อมูลวิธีการดูแลพืช เป็นหนึ่งใน Feature ของแอปพลิเคชัน AI สำหรับจำแนกสมุนไพรที่ใช้รักษาโรค และเลือก Feature การค้นหาด้วยชื่อและการบอกคุณลักษณะเด่นของพืชเพื่ออำนวยความสะดวกให้ผู้ใช้งานสามารถค้นหาพืชจากชื่อของสมุนไพรนั้นพร้อมบอกคุณลักษณะเด่น (Short description) ในการนำไปใช้ได้ตามที่ต้องการเพื่อส่งเสริมการใช้งานง่ายของแอปพลิเคชัน

สุดท้ายจะเป็น Feature ตามวัตถุประสงค์ของโครงการ คือ การจำแนกพืชและการบอกสรรพคุณผลข้างเคียงของสมุนไพร ดังนั้น Feature ทั้งหมดของแอปพลิเคชัน AI สำหรับจำแนกสมุนไพรที่ใช้รักษาโรคจะแบ่งเป็นหมวดหมู่หลัก ดังนี้

1. Feature ตามวัตถุประสงค์ของโครงการ
 - การจำแนกพืช
 - การบอกสรรพคุณและผลข้างเคียงของสมุนไพร
2. Feature ที่ส่งผลให้ง่ายต่อการใช้งาน (User Friendly)
 - การค้นหาพืชด้วยชื่อ
 - คุณลักษณะเด่น (Short description)
3. Feature ที่สำคัญต่อความต้องการของผู้ใช้
 - ข้อมูลการดูแลพืช

จำแนกพืชด้วยรูปภาพ	สรรพคุณและผลข้างเคียง	Community แลกเปลี่ยนความรู้	ค้นหาด้วยชื่อ	คุณลักษณะเด่นของพืช	ข้อมูลการดูแลพืช
✓	✓	x	✓	✓	✓

ตารางแสดง Feature ของแอปพลิเคชัน AI สำหรับจำแนกสมุนไพรที่ใช้รักษาโรค

บทที่ 3

วิธีการจัดทำโครงงาน

3.1 Problem Framing

เหตุผลที่ใช้ใบพืชสมุนไพรทั้ง 6 ชนิด คือ ฟ้าทะลายโจร โหระพา มะกรูด สะระแหน่ ตะไคร้ และกะเพรา เพราะในทางแพทย์แผนไทย สมุนไพรสามัญประจำบ้านเหล่านี้มีคุณสมบัติที่เป็นประโยชน์ในการรักษาโรค ทั้งบรรเทาอาการเครียด อาการนอนไม่หลับ และลดการอักเสบ ฯลฯ แต่อย่างไรก็ตามความท้าทายในเรื่องของการขาดมาตรฐานและความปลอดภัยของสมุนไพรบางชนิด โดยเฉพาะสมุนไพรสามัญประจำบ้านที่หาใช้รักษาได้ง่ายทำให้สามารถเกิดโอกาสการใช้สมุนไพรผิดประเภทมากยิ่งขึ้น ซึ่งในการสร้างความเชื่อมั่นและความปลอดภัยในการใช้สมุนไพรทั้ง 6 ชนิดนี้ ทางคณะผู้จัดทำจึงพิจารณานำสมุนไพรทั้ง 6 ชนิดมาสร้างโมเดลในการจำแนกโดยการพัฒนาปัญญาประดิษฐ์ซึ่งจะสามารถช่วยในการระบุชื่อสมุนไพรและข้อมูลที่เกี่ยวข้องของสมุนไพรนั้น เพื่อป้องกันไม่ให้เกิดการนำไปใช้งานผิดรูปแบบ และสนับสนุนการให้เกิดการใช้งานสมุนไพรให้เหมาะสม สามารถให้ข้อมูลเกี่ยวกับคุณประโยชน์และวิธีการใช้ที่ถูกต้อง เพื่อให้ผู้ใช้สามารถเลือกสมุนไพรในการรักษาที่ปลอดภัยต่อสุขภาพ

3.2 การเก็บและการเตรียมข้อมูล

1. เลือกแหล่งข้อมูลที่เหมาะสม

ในขั้นตอนแรกของโครงการนี้ การเลือกแหล่งข้อมูลที่เหมาะสมเป็นสิ่งสำคัญ เพื่อให้โมเดลที่สร้างขึ้นมีความแม่นยำและเชื่อถือได้ ฉะนั้น ทางคณะผู้จัดทำจึงค้นหาแหล่งข้อมูลที่มีภาพของสมุนไพรไทยที่มีมาตรฐานและความชัดเจน โดยการเลือกเว็บไซต์หรือฐานข้อมูลที่เชื่อถือได้ เช่น Kaggle ที่มีชุดข้อมูลที่มีคุณภาพและครอบคลุมการจำแนกประเภทของสมุนไพรไทย

2. ค้นหาชุดข้อมูลที่หลากหลาย

ค้นหาชุดข้อมูลใบพืชที่หลากหลายเพื่อให้โมเดลที่สร้างขึ้นสามารถจำแนกความแตกต่างของสมุนไพรไทยได้อย่างถูกต้อง เพื่อให้โมเดลสามารถจำแนกประเภทของสมุนไพรได้อย่างแม่นยำ

3. ตรวจสอบความถูกต้องของข้อมูล

เพื่อให้ได้ผลลัพธ์ที่มีความแม่นยำ การตรวจสอบความถูกต้องของข้อมูลเป็นสำคัญมาก โดยต้องประเมินความชัดของรูปภาพและตรวจสอบว่ารูปภาพที่ได้ค้นหามีความสอดคล้องกันในกลุ่มสมุนไพรประเภทเดียวกันและมีข้อมูลที่เชื่อถือได้ ซึ่งได้เก็บข้อมูลใบพืชสมุนไพรไทยจากแหล่งต่าง ๆ และใช้กล้องถ่ายรูปเพื่อรวบรวมข้อมูล เพื่อให้ได้ข้อมูลที่มีคุณสมบัติที่ต้องการ

ในการดำเนินวิจัยเพื่อพัฒนาโมเดลและวิเคราะห์ความเหมาะสมของโมเดลการวิเคราะห์สมุนไพรร
ผู้วิจัยได้เก็บรวบรวมข้อมูล ประเภทรูปภาพจาก 2 แหล่งข้อมูล ได้แก่

1. เว็บไซต์ที่มีความน่าเชื่อถือในด้านของข้อมูลรูปภาพ เช่น เว็บไซต์ Kaggle (Thammatat Tantipitam, 2021)
2. รูปภาพที่ถ่ายจากกลุ่มของผู้วิจัยเอง เพื่อต้องการรูปภาพที่มีสภาพแวดล้อมคล้ายคลึงกับรูปภาพที่ผู้ใช้จะทำการอัปโหลด

โดยพิจารณาการเก็บข้อมูลที่มีคุณสมบัติ ดังนี้

- รูปภาพใบพืชสมุนไพรที่มีความชัดเจน
- ขนาด Dataset หรือ จำนวนภาพมากพอที่ให้ AI เรียนรู้รูปแบบของใบสมุนไพร
- รูปภาพใบพืชมีความหลากหลาย ในสภาพแวดล้อมและแสงที่แตกต่างกัน
- ข้อมูลเกี่ยวกับพืชสมุนไพร เช่น ชื่อสมุนไพร ลักษณะทางพฤกษศาสตร์ และข้อมูลพื้นผิวภาพ เช่น สี ลวดลาย และพื้นผิว
- คุณภาพของข้อมูล เช่น ความถูกต้องชัดเจนของภาพ

มีการแบ่งข้อมูลออกเป็น 3 ส่วน ดังนี้

1. ข้อมูลสำหรับการ train มีการแบ่งเป็น 70% ของจำนวนรูปภาพในสมุนไพรชนิดนั้น ๆ
2. ข้อมูลสำหรับการ valid มีการแบ่งเป็น 20% ของจำนวนรูปภาพในสมุนไพรชนิดนั้น ๆ
3. ข้อมูลสำหรับการ test มีการแบ่งเป็น 10% ของจำนวนรูปภาพในสมุนไพรชนิดนั้น ๆ

รายละเอียดข้อมูล

ชื่อชุดข้อมูล	ประเภทของข้อมูล	จำนวนข้อมูล	ขนาดของภาพ
ชุดข้อมูล training set	ภาพใบฟ้าทะลายโจร	70 ภาพ	224 x 224 pixel
	ภาพใบโหระพา	70 ภาพ	224 x 224 pixel
	ภาพใบมะกรูด	70 ภาพ	224 x 224 pixel
	ภาพใบสะระแหน่	70 ภาพ	224 x 224 pixel
	ภาพใบตะไคร้	70 ภาพ	224 x 224 pixel
	ภาพใบกะเพรา	70 ภาพ	224 x 224 pixel
ชุดข้อมูล validation set	ภาพใบฟ้าทะลายโจร	20 ภาพ	224 x 224 pixel
	ภาพใบโหระพา	20 ภาพ	224 x 224 pixel
	ภาพใบมะกรูด	20 ภาพ	224 x 224 pixel
	ภาพใบสะระแหน่	20 ภาพ	224 x 224 pixel
	ภาพใบตะไคร้	20 ภาพ	224 x 224 pixel
	ภาพใบกะเพรา	20 ภาพ	224 x 224 pixel
ชุดข้อมูล test set	ภาพใบฟ้าทะลายโจร	10 ภาพ	224 x 224 pixel
	ภาพใบโหระพา	10 ภาพ	224 x 224 pixel
	ภาพใบมะกรูด	10 ภาพ	224 x 224 pixel
	ภาพใบสะระแหน่	10 ภาพ	224 x 224 pixel
	ภาพใบตะไคร้	10 ภาพ	224 x 224 pixel
	ภาพใบกะเพรา	10 ภาพ	224 x 224 pixel

โดยมีการเลือกให้ทำ vertical flip เพื่อให้ข้อมูลมีความหลากหลาย และ ปรับ batch size ให้เท่ากับ 32 เพื่อเพิ่มความแม่นยำ และความเร็วของการคำนวณ

3.3 การพัฒนาโมเดลการเรียนรู้เชิงลึกเพื่อวิเคราะห์และจำแนกใบสมุนไพรมะนาว

1. ข้อมูลที่นำมาใช้ในการวิเคราะห์ด้วยการเรียนรู้ของเครื่องและการเรียนรู้เชิงลึก

การเรียนรู้ของเครื่องและการเรียนรู้เชิงลึกนั้นเป็นการวิเคราะห์ที่ควรใช้ข้อมูลที่มีความหลากหลาย และมีจำนวนที่เหมาะสม ซึ่งข้อมูลชุดฝึกฝนที่มีคุณภาพจะส่งผลทำให้ประสิทธิภาพความถูกต้องในการทำนายสูงและมีความคงเส้นคงวาของความถูกต้อง ฉะนั้นนอกจากข้อมูลที่เราได้เก็บรวบรวมมาเองแล้ว จึงได้เลือกนำฐานข้อมูลจากเว็บไซต์ออนไลน์ที่มีความน่าเชื่อถืออย่าง kaggle นำมาใช้ในการสร้างชุดข้อมูลสำหรับการพัฒนาโมเดลนี้ด้วย ซึ่งเว็บไซต์ดังกล่าวเป็นเว็บไซต์ให้นักวิทยาศาสตร์ข้อมูลมาแลกเปลี่ยนโมเดลข้อมูลกัน ข้อมูลที่มีอยู่จึงมีความหลากหลายและมีคุณภาพ ทางผู้จัดทำจึงได้เลือกข้อมูลเหล่านี้มารวมกับฐานข้อมูลที่ได้ จากการกระบวนกรเก็บรวบรวมข้อมูลรูปภาพของตัวเองด้วย

2. การประเมินโมเดลที่ดีที่สุดสำหรับการวิเคราะห์

เพื่อที่จะประเมิน Deep learning model ในลำดับแรกผู้วิจัยได้เลือก Deep learning model ทั้งหมด 4 รูปแบบ มาพัฒนาสำหรับงานจำแนกภาพใบสมุนไพรมะนาว โดย Deep learning model ที่เรียกมานั้นมีสถาปัตยกรรม ดังต่อไปนี้

1. MobileNetV3Small
2. ResNet-50
3. EfficientNetV2L
4. NasNetMobile

ค่าพารามิเตอร์สำหรับแต่ละโมเดล

MobileNetV3Small , ResNet-50, EfficientNetV2L และ NasNetMobile

ค่าพารามิเตอร์	ค่า
batch size	32
epoch	30 รอบ
early stopping	10 รอบ

3) ขั้นตอนการพัฒนาโมเดลตามขั้นตอน

ทางผู้จัดทำได้พัฒนาโมเดลชุดนี้บน Google Colab และใช้ภาษา Python โดย Library ที่ใช้คือ tensorflow และ keras

```
[ ] !pip install tensorflow # Installs TensorFlow library
!pip install keras # Installs keras library

from google.colab import drive
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers, models
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.layers.experimental import preprocessing
from tensorflow.keras.callbacks import EarlyStopping
from tensorflow.keras.preprocessing import image
from tensorflow.keras.applications.resnet50 import ResNet50, preprocess_input
from tensorflow.keras.layers import Input, Lambda, Dense, Flatten
from tensorflow.keras.models import Model
from tensorflow.keras.applications import efficientnet_v2, NASNetMobile

import tensorflow_hub as hub
from sklearn.metrics import classification_report
import cv2
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import warnings
import os
from datetime import datetime

%matplotlib inline

[ ] drive.mount('/content/gdrive',force_remount=True)

Mounted at /content/gdrive
```

นำเข้า Library และ dataset ที่จะใช้งานในการ train โมเดล

```
[ ] train_path = "/content/gdrive/MyDrive/datasetplant/6ClassDatasetPlant/train" # @param(type:"string")
test_path = "/content/gdrive/MyDrive/datasetplant/6ClassDatasetPlant/test" # @param(type:"string")
valid_path = "/content/gdrive/MyDrive/datasetplant/6ClassDatasetPlant/valid" # @param(type:"string")
```

train_path:

test_path:

valid_path:

2.1. Set up Custom parameter Datagen

```
# @title 2.1. Set up Custom parameter Datagen { display-mode: "form" }
ROTATION_RANGE = 0 # @param (type:"number")
HORIZONTAL_FLIP = False # @param (type:"boolean")
VERTICAL_FLIP = False # @param (type:"boolean")
ZOOM_RANGE = 0 # @param (type:"number")
WIDTH_SHIFT_RANGE = 0 # @param (type:"number")
HEIGHT_SHIFT_RANGE = 0 # @param (type:"number")

BATCH_SIZE = 32 # @param (type:"integer")
IMAGE_SIZE = 224 # @param (type:"integer")
```

ROTATION_RANGE:

HORIZONTAL_FLIP: ☐

VERTICAL_FLIP: ☐

ZOOM_RANGE:

WIDTH_SHIFT_RANGE:

HEIGHT_SHIFT_RANGE:

BATCH_SIZE:

IMAGE_SIZE:

กำหนด path ให้แต่ละโฟลเดอร์ของ dataset ที่แยกมาเป็น 3 โฟลเดอร์ คือ train,valid และ test เพื่อนำไปใช้งานต่อ จากนั้นกำหนดพารามิเตอร์สำหรับการทำ data preparation ก่อนการนำไป train โดยการ train โมเดลทั้ง 4 ตัวนี้มีการกำหนดพารามิเตอร์ ดังนี้ กำหนดให้มีการกลับรูปในแนวตั้ง, batch size เท่ากับ 32 และ ขนาดของรูปภาพเท่ากับ 224 x 224 pixels

```

# @title 2.2 Run ImageDataGenerator

train_datagen = ImageDataGenerator( width_shift_range = WIDTH_SHIFT_RANGE ,
                                   height_shift_range = HEIGHT_SHIFT_RANGE ,
                                   rotation_range = ROTATION_RANGE ,
                                   horizontal_flip = HORIZONTAL_FLIP ,
                                   vertical_flip = VERTICAL_FLIP ,
                                   zoom_range = ZOOM_RANGE )
train_generator = train_datagen.flow_from_directory(
    directory=train_path,
    target_size=(IMAGE_SIZE, IMAGE_SIZE),
    batch_size=BATCH_SIZE,
    class_mode='categorical')

test_datagen = ImageDataGenerator( width_shift_range = WIDTH_SHIFT_RANGE ,
                                   height_shift_range = HEIGHT_SHIFT_RANGE ,
                                   rotation_range = ROTATION_RANGE ,
                                   horizontal_flip = HORIZONTAL_FLIP ,
                                   vertical_flip = VERTICAL_FLIP ,
                                   zoom_range = ZOOM_RANGE )
test_generator = test_datagen.flow_from_directory(
    directory=test_path,
    target_size=(IMAGE_SIZE, IMAGE_SIZE),
    batch_size=BATCH_SIZE,
    shuffle = False,
    class_mode='categorical')

valid_datagen = ImageDataGenerator( width_shift_range = WIDTH_SHIFT_RANGE ,
                                   height_shift_range = HEIGHT_SHIFT_RANGE ,
                                   rotation_range = ROTATION_RANGE ,
                                   horizontal_flip = HORIZONTAL_FLIP ,
                                   vertical_flip = VERTICAL_FLIP ,
                                   zoom_range = ZOOM_RANGE )
valid_generator = valid_datagen.flow_from_directory(
    directory=valid_path,
    target_size=(IMAGE_SIZE, IMAGE_SIZE),
    batch_size=BATCH_SIZE,
    shuffle = False,
    class_mode='categorical')

```

ImageDataGenerator ตามที่กำหนดไว้ซึ่งค่าพารามิเตอร์ต่าง ๆ มี ดังนี้

1. Random Rotation (สุ่มหมุนภาพ) เป็นการกำหนดช่วงการหมุนภาพในหน่วยองศา เมื่อกำหนดค่าเริ่มต้นและสิ้นสุดของช่วงการหมุน โมเดลจะสุ่มหมุนภาพในช่วงนั้นเพื่อให้โมเดลเรียนรู้ได้ดีกว่าในสถานะที่ภาพไม่ได้ถูกหมุน
2. Random Shifts (สุ่มเคลื่อนย้ายภาพ) เคลื่อนย้ายภาพไปในทิศทางต่าง ๆ อย่างสุ่มเพื่อเพิ่มความหลากหลายในชุดข้อมูลที่ใช้ในการฝึกสอนโมเดลเชิงลึก เป็นหนึ่งในเทคนิคในการเตรียมข้อมูลภาพที่มีประสิทธิภาพเพื่อป้องกันการเรียนรู้จากลักษณะเฉพาะของภาพในชุดข้อมูล
3. Random Flips (สุ่มพลิกภาพ) คือการทำการพลิกภาพในแนวนอนหรือในแนวตั้งอย่างสุ่มขณะเตรียมข้อมูลภาพสำหรับการฝึกสอนโมเดลเชิงลึก โดยมักจะใช้กับชุดข้อมูลภาพเพื่อเพิ่มความหลากหลายในการเรียนรู้ของโมเดลซึ่งช่วยลดโอกาสของการเกิดการเรียนรู้จากลักษณะของภาพที่มีเพียงแค่มุมมองหรือทิศทางการถ่ายทอด
4. Random Zoom (สุ่มซูมภาพ) คือการซูมภาพเข้าหรือซูมออกอย่างสุ่มเพื่อเตรียมข้อมูลสำหรับการฝึกสอนโมเดลเชิงลึก โดยการปรับการซูมนี้สามารถช่วยเพิ่มความหลากหลายในชุดข้อมูลและช่วยให้โมเดลมีความสามารถในการรับมือกับการเปลี่ยนแปลงขนาดของวัตถุในภาพได้


```
[ ]
class_names = train_generator.class_indices
class_names
```

```
{'Green chiretta': 0,
'Holy basil': 1,
'Kaffir lime': 2,
'Lemon balm': 3,
'Lemon grass': 4,
'Thai basil': 5}
```

ดิงค่าชื่อของแต่ละคลาสที่มีใน train_generator ในการจัดการข้อมูลสำหรับการฝึกอบรมโมเดล

```
▶ CLASS_SIZE = len(class_names)
```

```
[ ] import random
SEED = 42
os.environ["PYTHONHASHSEED"] = str(SEED)
os.environ["TF_CUDNN_DETERMINISTIC"] = '1' # TF 2.1
random.seed(SEED)
np.random.seed(SEED)
tf.random.set_seed(SEED)

#set seed
```

คำนวณจำนวนคลาสในชุดข้อมูลและกำหนดค่า CLASS_SIZE และทำการสุ่มให้มีลักษณะที่เหมือนกันทุกครั้ง เพื่อให้ผลลัพธ์ที่ได้มีความน่าเชื่อถือที่มีความเหมาะสม โดยการตั้งค่า seed จะช่วยให้สามารถควบคุมความสัมพันธ์ระหว่างการสุ่มและการฝึกโมเดลได้ดีขึ้น และทำให้ผลลัพธ์มีความน่าเชื่อถือมากยิ่งขึ้น

```
▶ # @title เลือก Model ที่นำมาใช้ Train
model_name = "MobileNetV3Small" # @param ["MobileNetV3Small", "ResNet-50", "EfficientNetV2L", "NASNetMobile"]
model_name: MobileNetV3Small

[ ] if model_name == "MobileNetV3Small":
    base_model = keras.applications.MobileNetV3Small(input_shape=(IMAGE_SIZE, IMAGE_SIZE, 3), include_top=False, weights='imagenet') # Load the MobileNetV3Small model
    base_model.trainable = True # Enable to train Pre-Trained Model
if model_name == "ResNet-50":
    base_model = ResNet50(
        input_shape=(IMAGE_SIZE, IMAGE_SIZE, 3), # Making the image into 3 Channel, so concating 3.
        weights='imagenet', # Default weights.
        include_top=False
    )
    base_model.trainable = True # Enable to train Pre-Trained Model
if model_name == "EfficientNetV2L":
    base_model = efficientnet_v2.EfficientNetV2B0(include_top=False,
        input_shape=(IMAGE_SIZE, IMAGE_SIZE, 3),
        weights="imagenet")

if model_name == "NASNetMobile":
    base_model = NASNetMobile(
        include_top=False,
        input_shape=(IMAGE_SIZE, IMAGE_SIZE, 3),
        weights="imagenet"
    )

inputs = tf.keras.layers.Input(shape=(224, 224, 3), name="input_layer") # Define input layer
```

เลือกโมเดลที่ต้องการนำมาใช้ train โดยทางผู้จัดทำได้เลือกมา 4 โมเดลที่จะนำมาใช้นั้นคือ MobileNetV3Small, ResNet-50, EfficientNetV2L, และ NasNetMobile

```
x = inputs # Apply Augmentation to the input images. this use for decrease overfit
x = base_model(x, training=False) # Pass Augmentation Data to base_model
x = layers.GlobalAveragePooling2D(name="GloPool")(x) # Create Global Average Pooling and input x to get output tensor that value = Dense's filter. this use for decrease feature map
outputs = tf.keras.layers.Dense(CLASS_SIZE, activation="softmax", name="output_layer")(x) # output layer that (output unit = CLASS_SIZE as image class, activation function is softmax )
model_0 = tf.keras.Model(inputs, outputs) # Create Model name model_0 that have input layer, base model, output layer
model_0.compile(loss='categorical_crossentropy',
optimizer=tf.keras.optimizers.Adam(0.0001),
metrics=['accuracy']) #Compile model_0 with categorical_crossentropy, adam's learning rate = 0.0001, metric by accuracy

[ ] model_0.summary()
```

สร้างโมเดล neural network โดยประกอบด้วยชั้นต่าง ๆ จากนั้นคอมไพล์โมเดล ใช้ optimizer เป็น Adam และกำหนด learning rate เป็น 0.0001 และกำหนด metrics ให้เป็น accuracy เพื่อให้ง่ายต่อการติดตามประสิทธิภาพของโมเดลในขณะที่ฝึกและทดสอบ

```
custom_early_stopping = EarlyStopping( # EarlyStopping is keras's callback function that stop training model before overfitting
monitor='val_loss', # monitor at validation loss
patience=10,
min_delta=0.000000001, # if validation loss is not decrease at least 0.001 in 10 time
mode='min' # need validation loss to decrease
)
```

ใช้ early stopping (custom_early_stopping) เพื่อหยุดการฝึกอบรมเมื่อเกิดการ overfitting หรือก็คือการที่โมเดล ทำงานได้ดี มีความแม่นยำสูงเมื่อเป็นการเรียนรู้ข้อมูลจาก Training Data set แต่เมื่อนำไปใช้กับข้อมูลที่ไม่เคยเห็นมาก่อนมีความแม่นยำต่ำและผิดพลาดมาก

```
[ ] start = datetime.now()

history = model_0.fit(train_generator, # Use from train generator
epochs=30, # training round
workers=0,
steps_per_epoch=len(train_generator), # Use from batch size that can update each epoch for training
validation_data=valid_generator, # Use from test generator
validation_steps=len(valid_generator),
callbacks=[custom_early_stopping]) # Stop training when overfitting

print('Execution Time: ',datetime.now()-start)
```

Train Model โดยใช้ชุดข้อมูลจาก train_generator และมีรอบการ train (epochs) อยู่ที่ 30 รอบ

```
#@title 3.2. Evaluate the model on the test set
test_loss, test_acc = model_0.evaluate_generator(test_generator, steps=len(test_generator))
print('Test loss:', test_loss)
print('Test accuracy:', test_acc)
```

```
[ ] # Evaluate the model on the test set
start = datetime.now()
test_loss, test_acc = model_0.evaluate_generator(test_generator, steps=len(test_generator))
print('Test loss:', test_loss)
print('Test accuracy:', test_acc)
print('Test Time:',datetime.now()-start)
```

ทำการประเมินประสิทธิภาพของโมเดล Neural Network ด้วยชุดข้อมูล Test set โดยใช้ฟังก์ชัน evaluate_generator จากนั้นแสดงผลค่าความสูญเสีย (loss) และค่าความแม่นยำออกมา (accuracy)

```

import sklearn as scikit_learn
from sklearn.metrics import ConfusionMatrixDisplay
from sklearn.metrics import confusion_matrix
import numpy as np
from sklearn.metrics import precision_recall_fscore_support

test_true=test_generator.classes[test_generator.index_array]
test_pred_raw = model_0.predict(test_generator)
test_pred = np.argmax(test_pred_raw, axis=1)

cm = confusion_matrix(test_true, test_pred)
disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=class_names)
fig, ax = plt.subplots(figsize=(10,10))
disp.plot(ax=ax,cmap=plt.cm.Blues)
plt.show()

res = []
for i in range(CLASS_SIZE):
    pres,recall,_ = precision_recall_fscore_support(np.array(test_true)==i,np.array(test_pred)==i,pos_label=True,average=None)
    res.append([i,recall[0],recall[1]])

pd.DataFrame(res,columns = ['class','specificity','sensitivity'])

```

วัดประสิทธิภาพของโมเดล โดยใช้วิธีการวัดผลคือการคำนวณและแสดงผล Confusion Matrix นั่นคือเทียบค่าคลาสจริง (actual class) กับค่าคลาสที่โมเดลทำนาย (predicted class) และสร้างเมทริกซ์ที่แสดงสถิติเกี่ยวกับการจำแนกประเภทออกมา และคำนวณค่าความแม่นยำ (precision), ความละเอียด (recall), และ F1 score เพื่อประเมินความสามารถของโมเดลในการแยกแยะคลาสของข้อมูล

```

from sklearn.metrics import classification_report

report = classification_report(test_true, test_pred, target_names=class_names)
print(report)

```

สร้างผลรายงานการจำแนกประเภท (classification report) สำหรับผลลัพธ์การทำนายของโมเดลในแต่ละคลาส

```

acc = history.history['accuracy']
val_acc = history.history['val_accuracy']
loss = history.history['loss']
val_loss = history.history['val_loss']

epochs = range(len(acc))

plt.plot(epochs, acc, 'r', label='Training accuracy')
plt.plot(epochs, val_acc, 'b', label='Validation accuracy')
plt.title('Training and validation accuracy')
plt.legend(loc=0)
plt.figure(figsize=(6,6))

plt.plot(epochs, loss, 'r', label='Training loss')
plt.plot(epochs, val_loss, 'b', label='Validation loss')
plt.title('Training and validation loss')
plt.legend(loc=0)
plt.figure(figsize=(6,6))

plt.show()

```

วิเคราะห์และแสดงผลการ Train ของโมเดลผ่านการสร้างกราฟ โดยจะแสดงความแม่นยำและการสูญเสีย (loss) ทั้งในระหว่างการ Train และการตรวจสอบความถูกต้อง (validation) จากนั้นแสดงออกมาเป็นกราฟ

```

# Save the trained model to a .h5 file
model_0.save('/content/gdrive/MyDrive/datasetplant/model/trainedmodel.h5')

```

หลังจากการ Train Model เสร็จ save model ที่ได้เป็นไฟล์ h.5 ไปยัง Google drive เพื่อนำไปใช้งานต่อ

```

from keras.models import load_model
model = load_model('/content/gdrive/MyDrive/datasetplant/model/trainedmodel.h5')
converter = tf.lite.TFLiteConverter.from_keras_model(model)
converter.optimizations = [tf.lite.Optimize.OPTIMIZE_FOR_LATENCY]
tflite_model = converter.convert()
open("/content/gdrive/MyDrive/datasetplant/model/tflitemodel.tflite", "wb").write(tflite_model)

```

โหลดโมเดล Keras จากไฟล์ .h5 และแปลงโมเดลนั้นเป็นรูปแบบ .tflite ซึ่งเป็นรูปแบบโมเดลขนาดเล็กของ TensorFlow

```

[ ] # Upload image file
from google.colab import files
uploaded = files.upload()

# Iterate through uploaded files
for filename, file_content in uploaded.items():
    print("User uploaded file \"{name}\" with length {length} bytes".format(
        name=filename, length=len(file_content)))

# Convert file content to OpenCV image
nparr = np.frombuffer(file_content, np.uint8)
img = cv2.imdecode(nparr, cv2.IMREAD_COLOR)

```

ผู้ใช้งานสามารถอัปโหลดไฟล์ภาพจากคอมพิวเตอร์ ขึ้นสู่ Colab และแปลงไฟล์ภาพนั้นเป็นภาพในรูปแบบ OpenCV เพื่อให้สามารถนำภาพไปใช้กับโมเดลต่อไปได้

```

# Get dimensions of the image
height, width, _ = img.shape

# Define the dimensions of the middle portion you want to crop
crop_width = 400 # Width of the cropped region
crop_height = 400 # Height of the cropped region

# Calculate the starting coordinates for the middle portion
x = (width - crop_width) // 2
y = (height - crop_height) // 2

# Crop the image
cropped_image = img[y:y+crop_height, x:x+crop_width]

# Display the original and cropped images
plt.imshow(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
plt.title('Original Image')
plt.axis('off')
plt.show()

plt.imshow(cv2.cvtColor(cropped_image, cv2.COLOR_BGR2RGB))
plt.title('Cropped Image')
plt.axis('off')
plt.show()

# Save the cropped image
cv2.imwrite('/content/gdrive/My Drive/Imagecrop/cropped.jpg', cropped_image)

```

ทำการครอบตัดภาพ เพื่อให้ได้พื้นที่กลางภาพในขนาดที่กำหนด จากนั้นแสดงภาพต้นฉบับและภาพที่ถูกครอบตัดแล้ว และบันทึกภาพที่ครอบตัดลงใน Google drive เพื่อใช้งานต่อ

```

# Load the cropped image
cropped_img_path = '/content/gdrive/My Drive/Imagecrop/cropped.jpg' # Update this path with the path where you saved the cropped image
img = image.load_img(cropped_img_path, target_size=(224, 224))
img_array = image.img_to_array(img)
img_array = np.expand_dims(img_array, axis=0)
img_array = tf.keras.applications.mobilenet_v3.preprocess_input(img_array)

# Make prediction
prediction = model.predict(img_array)
class_names = train_generator.class_indices
class_names = {v: k for k, v in class_names.items()} # reverse the class_indices dictionary
predicted_class = np.argmax(prediction, axis=1)
predicted_class_name = class_names[predicted_class[0]]
predicted_prob = np.max(prediction, axis=1)

# Print the result
print("Image name:", file_name)
print("Predicted class: {} {:.2f}%".format(predicted_class_name, predicted_prob[0]*100))
prediction_list = prediction.tolist() # Convert the NumPy array to a Python list
prediction_formatted = ["{:.2f}%".format(p) for p in prediction_list[0]]
print("Prediction: ", prediction_formatted)
plt.imshow(img)
plt.title("Predicted class: {} {:.2f}%".format(predicted_class_name, predicted_prob[0]*100))
plt.show()
print("\n")

```

โหลดภาพที่ครอบตัดแล้วทำการทำนายด้วยโมเดล จากนั้นทำการทำนายคลาสของภาพและแสดงผลการทำนายออกมา

3.4 การออกแบบ UX/UI ของ Application

หลังจากการวิเคราะห์เปรียบเทียบหา Feature ที่สำคัญในการสร้างแอปพลิเคชัน AI สำหรับจำแนกสมุนไพรที่ใช้รักษาโรค คือ การจำแนกพืชด้วยรูปภาพ, สรรพคุณและผลข้างเคียง, การค้นหาด้วยชื่อ, คุณลักษณะเด่นของพืช และข้อมูลการดูแลพืช มาออกแบบ UX/UI ของ Application จะได้ Phototype ดังนี้

1. หน้าเริ่มต้นแอปพลิเคชัน



2. หลังจากลงทะเบียนใช้เสร็จก็จะเข้าสู่การค้นหาหรือจำแนกประเภทใบพืชได้ทั้งการอัปโหลดรูปภาพเพื่อจำแนกจากใบและการค้นหาด้วยชื่อพืชเพื่อดูรายละเอียดของพืช

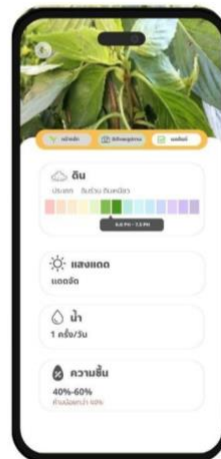


3. หลังจากอัปโหลดรูปภาพหรือค้นหาพืชโดยใช้ชื่อเสร็จแล้วจะแสดงข้อมูลพืชดังกล่าวพร้อมคุณลักษณะเด่น สรรพคุณ ผลข้างเคียง และข้อมูลการดูแลพืช



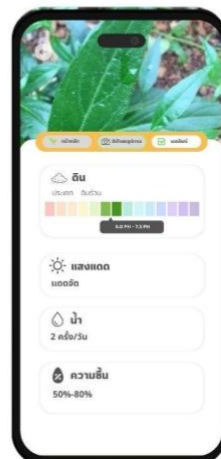
ตัวอย่างการแสดงผลพืช

โหระพา



ตัวอย่างการแสดงผลพืช

ฟ้าทะลายโจร



ตัวอย่างการแสดงผลพืช

กะเพรา



บทที่ 4

ผลการศึกษา

4.1 ระบบและเฟรมเวิร์กที่ใช้ในการทดลอง

ในการทดลองนี้จะใช้ Google Colab ในการทำการทดลองและใช้เฟรมเวิร์กคือ Keras และ tensorflow สำหรับการพัฒนา deep learning model

4.2 ประสิทธิภาพของ model เป้าหมาย

ทำการพัฒนา deep learning model โดยใช้การตั้งค่าตามที่กล่าวไป เมื่อการ พัฒนาเสร็จสิ้นนำ model ที่เลือกในแต่ละสถาปัตยกรรมมาประเมินประสิทธิภาพกับชุดข้อมูล test set โดยได้ผลดังนี้

สถาปัตยกรรม	ค่า Accuracy	ค่า Sensitivity	ค่า Specificity
MobileNetV3Small	0.57	0.57	0.9258
ResNet-50	0.59	0.68	0.9356
EfficientNetV2L	0.76	0.81	0.9633
NasNetMobile	0.20	0.629	0.9266

4.3 การประเมินโมเดล

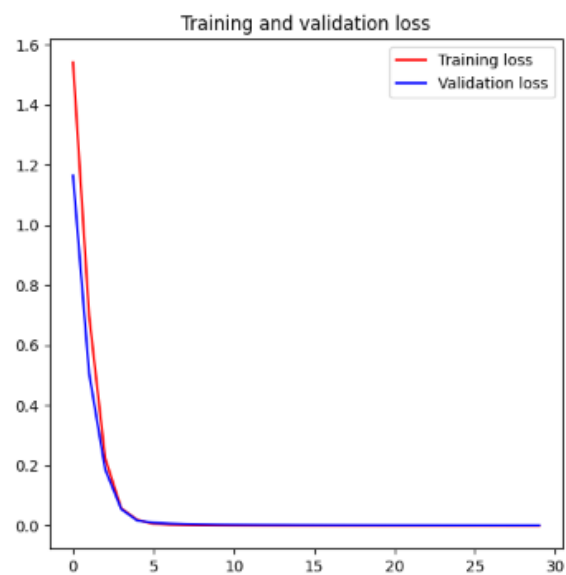
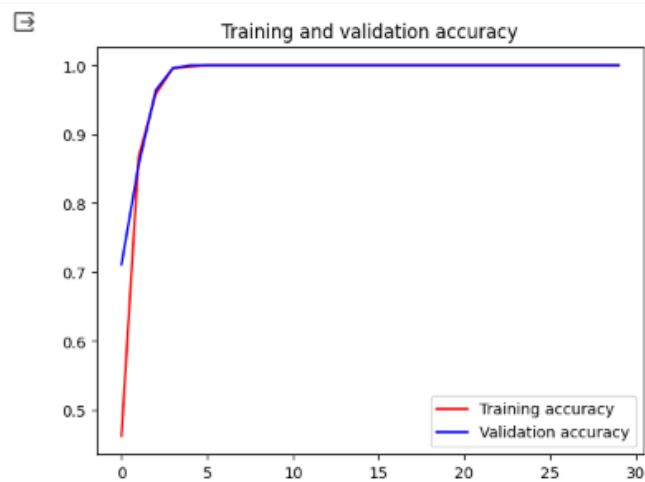
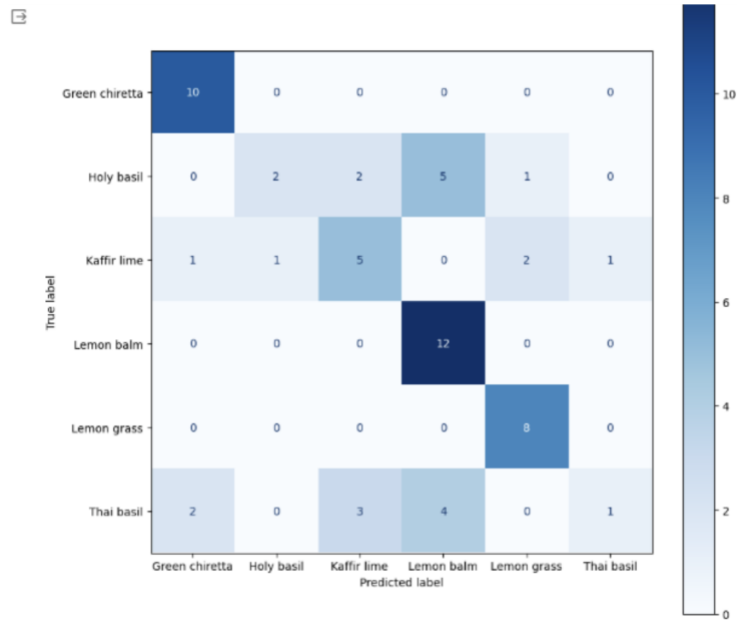
ทำการ train โมเดลทั้งหมด 30 รอบในแต่ละโมเดล และทุก ๆ รอบจะได้ค่าแสดงความแม่นยำ ค่าความสูญเสีย และค่าระยะเวลาที่ใช้ในการประเมินผล จากนั้นค่อยทำการสรุปผลโมเดลอีกรอบโดยวัดผลคือการทำ Confusion Matrix และคำนวณค่าความแม่นยำ (precision), ความละเอียด (recall), และ F1 score เพื่อประเมินความสามารถของโมเดลในการแยกแยะคลาสต่าง ๆ ของข้อมูล

4.4 ผลการทดลอง

โดยผลการทดลองนั้นจะแบ่งเป็นทีละโมเดลโดยมี 4 โมเดลและแต่ละโมเดลจะทำการ train 30 รอบ โดยประกอบไปด้วยผลการทดลองโมเดล 1.MobileNetV3Small 2.ResNet-50 3.EfficientNetV2L 4.NasNetMobile ได้ผลการทดลองที่ใช้ Confusion matrix วัดผลดังด้านล่างนี้

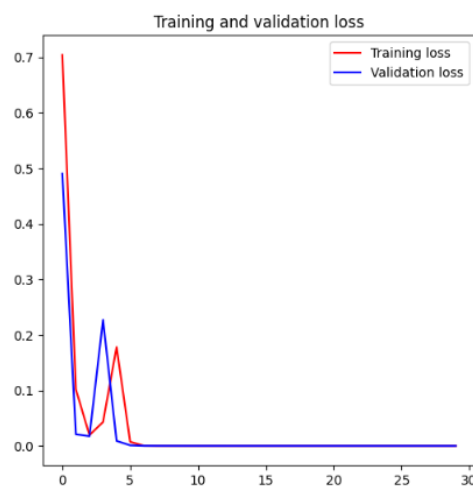
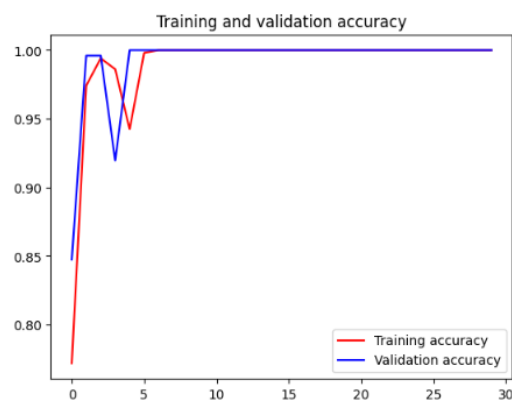
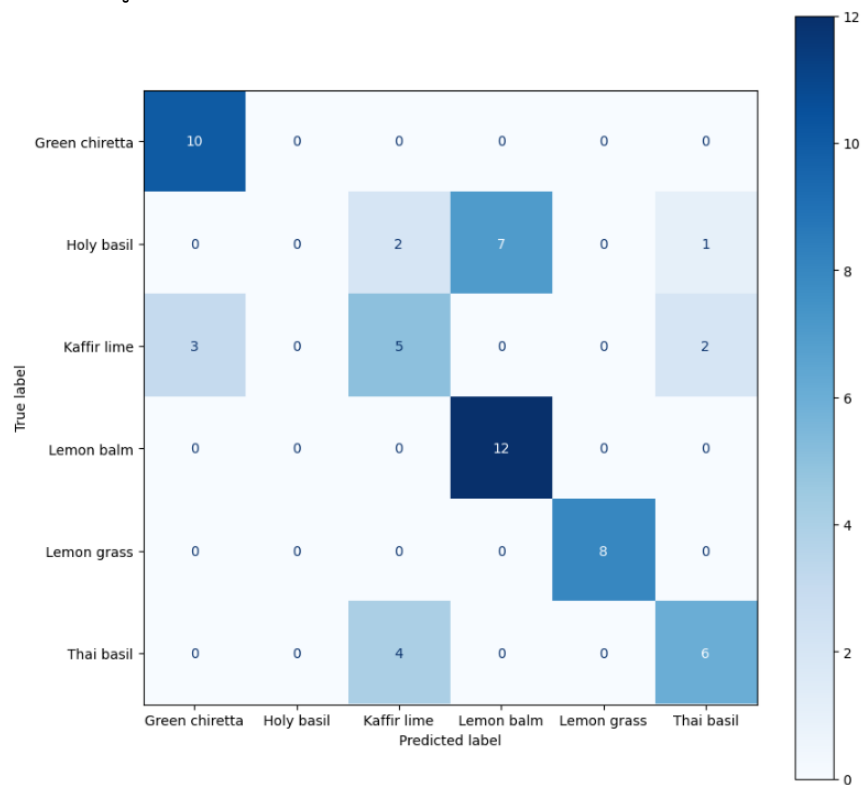
MobileNetV3Small

จำนวนตัวอย่างที่จะถูกประมวลผลโดยโมเดลในแต่ละรอบ : 32



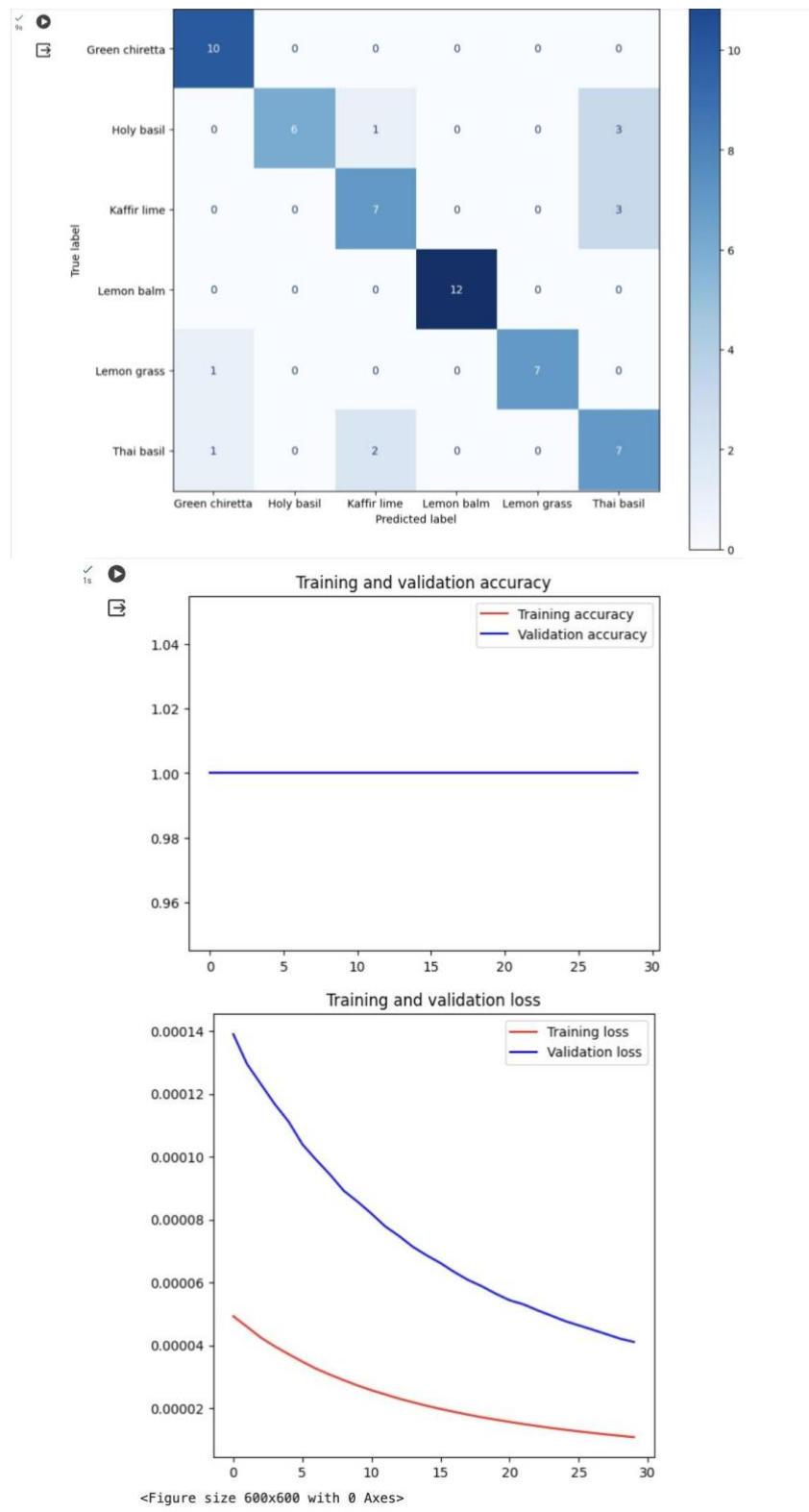
ResNet-50

จำนวนตัวอย่างที่จะถูกประมวลผลโดยโมเดลในแต่ละรอบ : 32



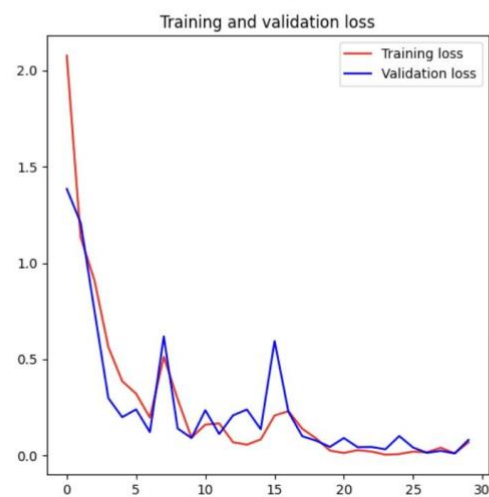
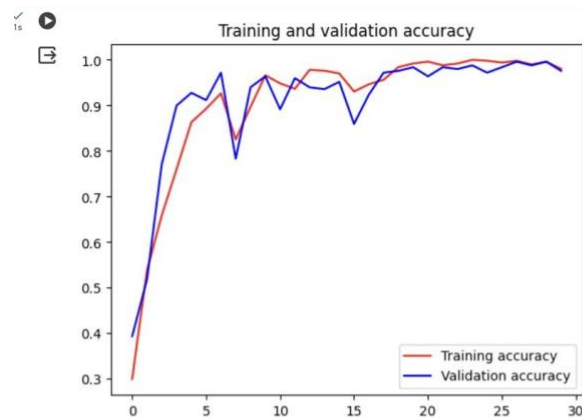
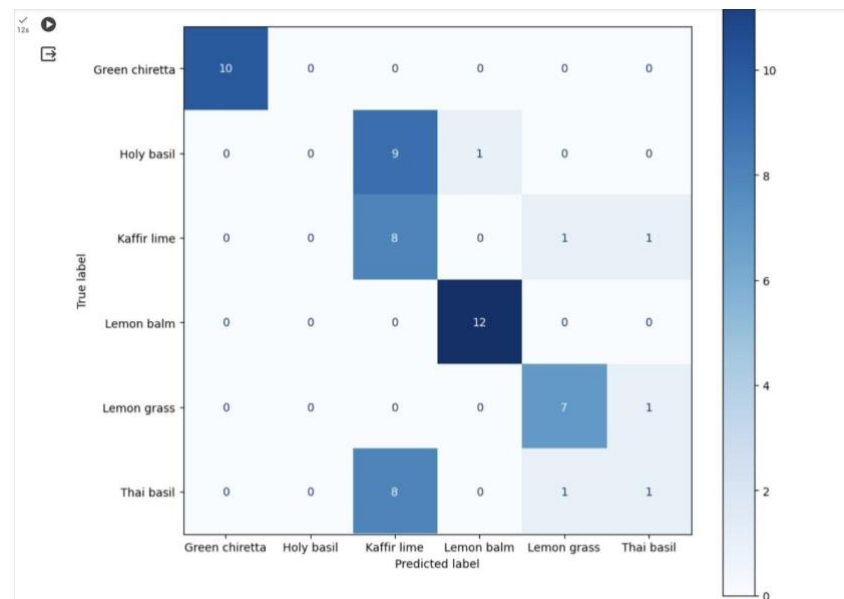
EfficientNetV2L

จำนวนตัวอย่างที่จะถูกประมวลผลโดยโมเดลในแต่ละรอบ : 32



NasNetMobile

จำนวนตัวอย่างที่จะถูกประมวลผลโดยโมเดลในแต่ละรอบ : 16



<Figure size 600x600 with 0 Axes>

จากภาพจะเห็นได้ว่า Confusion matrix นั้นจะสามารถบอกได้ถึงความแม่นยำของตัวโมเดลโดยโมเดลที่มีความแม่นยำมากที่สุดนั้นคือ โมเดล EfficientNetV2L

หลังจากนั้นในการดำเนินการต่อจะทำการครอบตัดภาพที่ผู้ใช้ต้องการเพื่อให้ได้พื้นที่กลางภาพในขนาดที่กำหนดเพื่อช่วยเพิ่มความแม่นยำในการทำนายผลของตัวโมเดลให้สามารถดูลักษณะทางกายภาพของใบไม้ได้ชัดเจนยิ่งขึ้น

4.5 อภิปรายผลการทดลอง

จากการทดลองแล้วแสดงผลโดยใช้ Confusion matrix นั้นจะเห็นได้ว่า Confusion matrix จะช่วยให้ทราบถึง ค่าความแม่นยำ (precision), ความละเอียด (recall) และ F1 score ที่ช่วยบ่งบอกถึงประสิทธิภาพของการทำนายของโมเดลตัวนั้นได้โดยโมเดล MobilenetV3small จะมีค่าความแม่นยำที่มีประสิทธิภาพในเรื่องของความแม่นยำรองลงมา สังเกตได้จาก Confusion matrix และค่าความแม่นยำ ของตัวโมเดลที่มีค่า 0.57 โดยโมเดลที่มีประสิทธิภาพความแม่นยำลดลงมากก็คือโมเดล NasNetMobile ที่ให้ค่าความแม่นยำอยู่ที่ 0.20 เช่นกัน ต่อมาโมเดลที่มีความแม่นยำมากขึ้นคือโมเดล ResNet-50 ที่มีค่าความแม่นยำ 0.59 มากกว่าโมเดล MobilenetV3small และโมเดล NasNetMobile ส่วนโมเดลที่มีค่าความแม่นยำมากที่สุดคือ EfficientNetV2L ที่ให้ค่าความแม่นยำสูงถึง 0.76 เป็นโมเดลที่มีประสิทธิภาพในการทำนายรวมถึงสามารถให้ผลลัพธ์ตรงความต้องการมากกว่าโมเดลตัวอื่น ๆ

ถึงจุดนี้ผู้วิจัยได้แสดงให้เห็นว่า deep learning model ที่มีประสิทธิภาพสูงในการใช้ทำนายลักษณะใบของพืช คือโมเดล EfficientNetV2L ซึ่งมี Confusion matrix ที่ช่วยเน้นย้ำถึงประสิทธิภาพและผลลัพธ์การทำนาย

บทที่ 5

สรุปผล และข้อเสนอแนะ

5.1 สรุปผล

จากวัตถุประสงค์หลักของโครงการคือการพัฒนาระบบปัญญาประดิษฐ์เพื่อจำแนกประเภทของสมุนไพรไทย 6 ชนิด ได้แก่ ฟ้าทะลายโจร โหระพา มะกรูด สะระแหน่ ตะไคร้ และกะเพรา ด้วยภาพลักษณะของใบพืชโดยใช้เทคนิค Deep Learning และขอบเขตของงานวิจัยครอบคลุมการเก็บข้อมูล การจัดเตรียมข้อมูล การพัฒนาโมเดล และการประเมินประสิทธิภาพของระบบ โดยผลการพัฒนาระบบ AI ในการจำแนกสมุนไพร ที่ได้พัฒนาโมเดลด้วยสถาปัตยกรรม Deep learning model ทั้งหมด 4 รูปแบบ ได้แก่ MobileNetV3Small, ResNet-50, EfficientNetV2L และ NasNetMobile ผลการทดสอบพบว่าโมเดล EfficientNetV2L มีประสิทธิภาพสูงสุดในการจำแนกสมุนไพร โดยให้ค่าความแม่นยำสูงถึง 0.76 และจากการทดสอบประสิทธิภาพของระบบได้ใช้ Confusion Matrix ในการประเมินผลการจำแนกช่วยเน้นย้ำถึงประสิทธิภาพและผลลัพธ์การทำนายของ EfficientNetV2L ที่สูงกว่าโมเดลอื่น ๆ อย่างเห็นได้ชัด นอกจากนี้ยังได้ทำการทดสอบด้วยการให้ผู้ใช้งานอัปโหลดภาพใบสมุนไพร แล้วใช้เทคนิค Center Crop เพื่อครอบตัดส่วนที่สำคัญของใบก่อนนำเข้าสู่โมเดล สามารถเพิ่มความแม่นยำได้มากขึ้น

การใช้ระบบ AI ในการจำแนกสมุนไพรมีความแม่นยำและรวดเร็วมากกว่าวิธีการดั้งเดิมที่ต้องใช้ผู้เชี่ยวชาญในการระบุชนิดสมุนไพร ซึ่งใช้ระยะเวลานานและอาจมีความคลาดเคลื่อนได้หากผู้ระบุขาดความชำนาญ ระบบจึงพัฒนาขึ้นมีประสิทธิภาพสูงกว่าเนื่องจากได้ใช้สถาปัตยกรรมร่วมกับเทคนิคการเตรียมข้อมูลที่เหมาะสม ทำให้มีการใช้สมุนไพรไทยสามัญประจำบ้านอย่างปลอดภัยและมีประสิทธิภาพมากยิ่งขึ้น

5.2 ข้อเสนอแนะในการนำผลงานวิจัยไปใช้ประโยชน์

ระบบที่พัฒนาขึ้นสามารถนำไปใช้งานได้จริงในการจำแนกประเภทสมุนไพรไทย เพื่อส่งเสริมการใช้สมุนไพรอย่างถูกต้องและมีประสิทธิภาพมากขึ้น โดยสามารถนำไปประยุกต์ใช้กับภาคส่วนต่าง ๆ เช่น สถานพยาบาลแผนไทย ร้านขายยาสมุนไพร เกษตรกรผู้ปลูกสมุนไพร ตลอดจนผู้บริโภคทั่วไปโดยในอนาคตสามารถพัฒนาระบบให้สามารถจำแนกสมุนไพรเพิ่มเติมได้มากกว่า 6 ชนิด หรือสามารถขยายขอบเขตไปสู่การจำแนกพืชสมุนไพรในรูปแบบอื่น ๆ ร่วมด้วยกับการจำแนกจากใบ เช่น ดอก ราก หรือเปลือกของพืช เพื่อเพิ่มความแม่นยำในการจำแนกประเภทสมุนไพรรวมถึงพัฒนาฟังก์ชันเพิ่มเติม เช่น การระบุปริมาณพืชสมุนไพรในภาพ การแนะนำวิธีใช้ที่เหมาะสมสำหรับสมุนไพรแต่ละชนิด เพื่อเพิ่มประโยชน์ในการนำไปใช้งานต่อได้ง่ายขึ้น นอกจากนี้ ยังสามารถสร้างหรือพัฒนาแนวทางการต่อยอด ดังนี้

- Camera Tracking เพื่อประมวลผลจำแนกประเภทพืชอัตโนมัติเรียลไทม์ผ่านกล้อง
- AI ช่วยผสมสมุนไพร สำหรับการผลิตยาสมุนไพร
- AI คัดกรองและควบคุมคุณภาพสมุนไพรในอุตสาหกรรม
- แอปพลิเคชันคำนวณสูตรยาสมุนไพรเฉพาะบุคคล

การพัฒนาระบบจำแนกประเภทสมุนไพรไทยโดยใช้ปัญญาประดิษฐ์นี้ จะก่อให้เกิดประโยชน์ต่อประชาชนให้สามารถเข้าถึงข้อมูลเกี่ยวกับพืชสมุนไพรไทยได้ง่ายและสะดวกมากขึ้นเพื่อส่งเสริมให้เกิดการใช้สมุนไพรไทยอย่างถูกต้องและปลอดภัยมากขึ้นให้เกษตรกรสามารถปลูกและจำหน่ายสมุนไพรไทยได้อย่างมีประสิทธิภาพและมั่นใจในคุณภาพทำให้กระบวนการคัดแยกประเภทของสมุนไพรไทยรวดเร็วและมีประสิทธิภาพมากยิ่งขึ้นเป็นการอนุรักษ์และส่งเสริมการใช้ประโยชน์จากสมุนไพรไทยอย่างยั่งยืน ด้วยการนำเทคโนโลยีปัญญาประดิษฐ์เข้ามามีบทบาท เพื่อให้สมุนไพรสามัญประจำบ้านเป็นที่รู้จักและได้รับการใช้อย่างปลอดภัยต่อไป

ภาคผนวก

อุปสรรคและปัญหาที่พบ

1. RAM ไม่เพียงพอต่อโมเดลที่จะใช้งาน

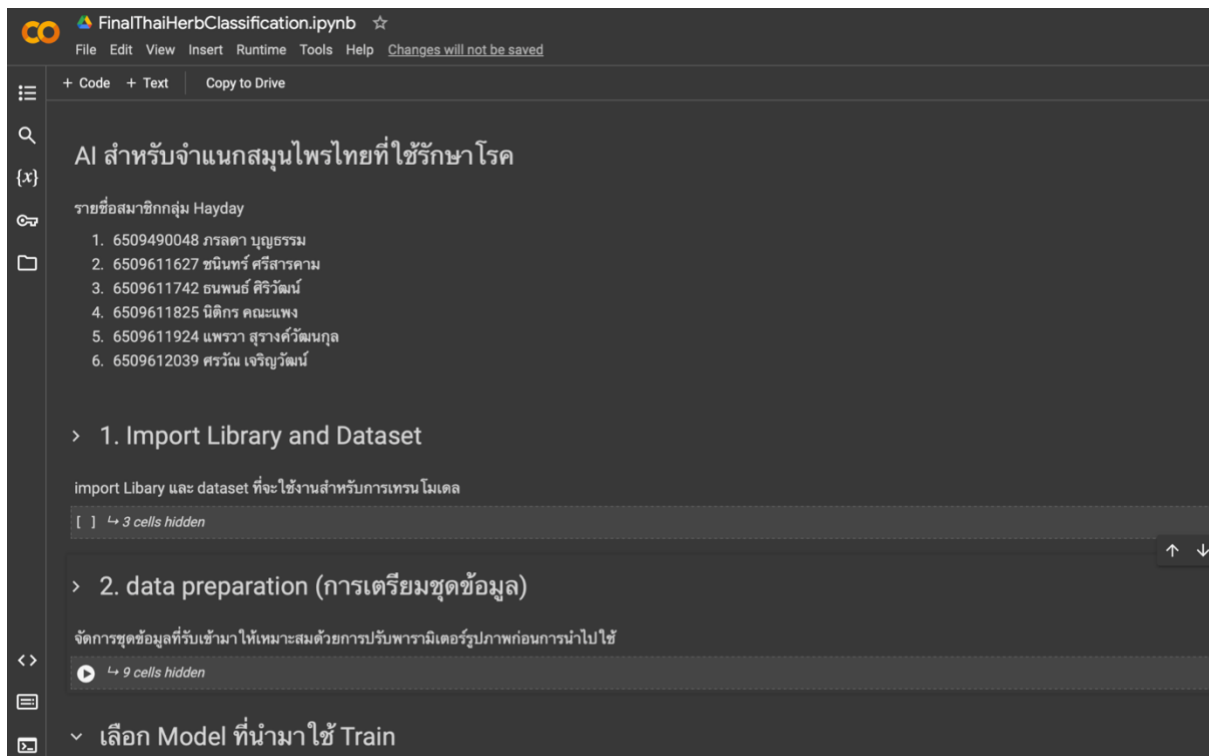
เริ่มแรกทางผู้จัดทำวิจัยได้เลือกโมเดล NASNetLarge สำหรับการ train แต่เกิดปัญหาเนื่องจาก RAM บน cloud ใน Google Colab ไม่เพียงพอต่อการรัน NASNetLarge Model ทางผู้จัดทำวิจัยจึงแก้ปัญหาโดยการลดขนาดของ batch size ลงและเปลี่ยนไปใช้โมเดล NASNetMobile การแก้ไขข้อจำกัดของ RAM ไม่เพียงพอสำหรับการปรับใช้และใช้โมเดล NASNet-Large ในแอปพลิเคชันโดยการไ้รุ่น NASNetMobile ซึ่งมีพารามิเตอร์น้อยกว่า จึงสามารถแก้ไขปัญหาระ RAM ไม่เพียงพอที่อาจเกิดขึ้นกับรุ่น NASNetLarge ดังนั้น NASNetMobile ที่มีน้ำหนักเบาและสามารถโหลดและรันบนอุปกรณ์ที่มี RAM ที่จำกัด ทำให้เหมาะสำหรับแอปพลิเคชันบนมือถือและสภาพแวดล้อมที่จำกัดทรัพยากรอื่น ๆ

2. โมเดลที่ผ่านการ train มาแล้ววิเคราะห์ได้ไม่แม่นยำ

ทางผู้จัดทำวิจัยจึงได้คิดหาวิธีเพื่อที่จะทำให้โมเดลวิเคราะห์ได้แม่นยำมากขึ้นจึงได้เลือกใช้วิธีคือ ให้มีการตรวจจับเฉพาะภาพใบแล้วให้โมเดลนำภาพนั้นไปใช้วิเคราะห์ จึงได้มีการใช้ model YOLOv8 ที่ให้บริการโดย roboflow ในการตรวจจับใบของสมุนไพรแล้วนำรูปที่ตรวจจับมาวิเคราะห์ต่อ แต่เกิดปัญหาคือ ไม่สามารถนำข้อมูลที่ผ่านการตรวจจับโดย model YOLOv8 มาใช้ต่อได้ ทางผู้จัดทำวิจัยจึงได้เปลี่ยนจากการใช้ YOLOv8 ในการตรวจจับเปลี่ยนเป็นการ crop ภาพใบไม่เฉพาะกลางรูป โดยใช้ Library ของ OpenCV2 ในการ crop รูปที่ input เข้ามาเพื่อให้โมเดลวิเคราะห์ได้แม่นยำมากขึ้น ซึ่งประสบผลสำเร็จ

โปรแกรมทั้งหมดที่ใช้ในการทำโครงงาน

1. โปรแกรม Google Colab สำหรับการพัฒนาโมเดล



Link Google Colab:

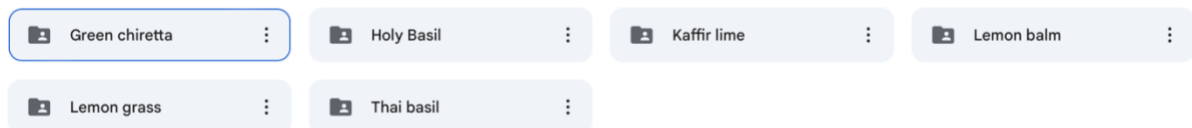
<https://colab.research.google.com/drive/1s8MXo2iQ1XSTjU4gdhMlBj5AbZD0orRV?usp=sharing>

2. โปรแกรม Google Drive สำหรับการ Train Model

Folders



Folders



Link Download Dataset :

https://drive.google.com/file/d/1L_tD2MFnaSd4huWSltSmjKnsBHeyM0nw/view?usp=sharing

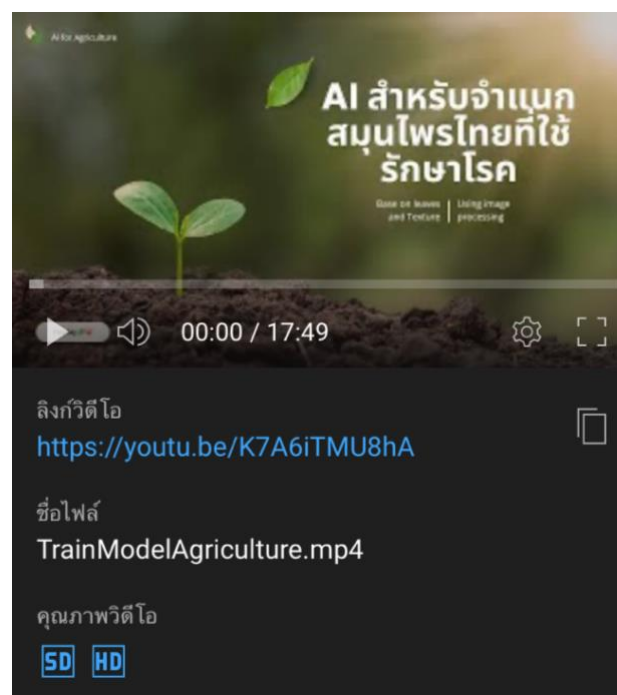
3. โปรแกรม Canva สำหรับการนำเสนอและออกแบบ UX/UI



Link Canva:

https://www.canva.com/design/DAF9m2V7W_8/wHmYJStjuNlMLxUEYW7-mg/edit?utm_content=DAF9m2V7W_8&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton

4. Video สำหรับการติดตั้ง Model



Link Video: <https://youtu.be/K7A6iTMU8hA>

บรรณานุกรม

- Adedamola O. Adedaja, P. Owolawi, T. Mapayi. (2021). Intelligent Mobile Plant Disease Diagnostic System Using NASNet-Mobile Deep Learning. *IAENG International Journal of Computer Science*, 49(1), retrieved from <https://www.semanticscholar.org/paper/Intelligent-Mobile-Plant-Disease-Diagnostic-System-Adedaja-Owolawi/6ffd60e21be26573faa42b2c6e432af4d92928b5>
- Adrian L. Lopresti, Stephen J. Smith, Alexandra P. Metse, Peter D.Drummond.(2022).A randomized, double-blind, placebo-controlled trial investigating the effects of an *Ocimum tenuiflorum* (Holy Basil) extract (Holixer™) on stress, mood, and sleep in adults experiencing stress.retrieved from <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9524226/>
- AIBY. (2022). Plantum - AI Plant Identifier. App Store. retrieved from <https://apps.apple.com/us/app/plantum-ai-plant-identifier/id1476047194>
- Appvillis.(2023).AI Plant Identifier App - PLNT. App Store. retrieved from <https://apps.apple.com/us/app/ai-plant-identifier-app-plnt/id1560965917>
- Fernando Perez-Sanz, Pedro J Navarro, Marcos Egea-Cortines. (2017). Plant phenomics: an overview of image acquisition technologies and image data analysis algorithms. *GigaScience*, 6(11), <https://doi.org/10.1093/gigascience/gix092>
- GREGARIOUS, INC.(2022).Greg - Plant Care. App Store. retrieved from <https://apps.apple.com/th/app/greg-plant-care/id1512912236>
- J R Leite et al.(1986).Pharmacology of lemongrass (*Cymbopogon citratus* Stapf). III. Assessment of eventual toxic, hypnotic and anxiolytic effects on humans. retrieved from <https://pubmed.ncbi.nlm.nih.gov/2429120/>

Julien Cases et al.(2010).Pilot trial of *Melissa officinalis* L. leaf extract in the treatment of volunteers suffering from mild-to-moderate anxiety disorders and sleep disturbances. retrieved from <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3230760/>

Kavitha, S., Kumar, T.S., Naresh, E. *et al.* (2024). Medicinal Plant Identification in Real-Time Using Deep Learning Model. *SN COMPUT. SCI.* **5**, 73. <https://doi.org/10.1007/s42979-023-02398-5>

Megan Taslamam.(2014).The efficacy and safety of herbal medicine for insomnia in adults: an overview of recent research.retrieved from <https://sleepdrops.co.nz/wp-content/uploads/2018/10/Herbal-medicines-for-insomnia-review-Copy.pdf>

Murad Kabir Md. Rakib et al.(2023).Automatic Recognition of Medicinal Plants Based on Multispectral Texture Features using Hidden Deep Learning Model. Retrieved from https://www.researchgate.net/publication/370059103_Automatic_Recognition_of_Medicinal_Plants_Based_on_Multispectral_and_Texture_Features_using_Hidden_Deep_Learning_Model

Rahmat Budiart ,Mohammad Miftakhus Sholikin.(2022).Kaffir Lime Essential Oil Variation in the Last Fifty Years: A Meta-Analysis of Plant Origins, Plant Parts and Extraction Methods.retrieved from <https://www.mdpi.com/2311-7524/8/12/1132>

Shasha Luo, He Li , Jingjing Liu.(2020).Andrographolide ameliorates oxidative stress, inflammation and histological outcome in complete Freund's adjuvant-induced arthritis. retrieved from <https://www.sciencedirect.com/science/article/abs/pii/S0009279719316497?via%3Dihub>

Straiton Limited.(2022). Plant Identifier - PlantMe. App Store. retrieved from <https://apps.apple.com/us/app/plant-identifier-plantme/id1532875668>

Thammatat Tantipitam. (2021). Thai Herb. kaggle. retrieved from <https://www.kaggle.com/datasets/thammatattantipitham/thaiherb>

Yacynth Ndonga.(2021).Medicinal plants: herbs, bark. App Store. retrieved from
<https://apps.apple.com/th/app/medicinal-plants-herbs-bark/id1587896252?l=th&platform=ipad>

กรมการแพทย์แผนไทยและการแพทย์ทางเลือก.(2019).Herb ID. App Store. retrieved from
<https://apps.apple.com/th/app/herb-id/id1468047481?l=th>

รัชฎาพร อุ่นศิริไธย,รศ.ดร.อนามัย เทศกะทีก.(2018).คุณสมบัติสารออกฤทธิ์ทางชีวภาพของสารสกัดโหระพาไทย. แหล่งที่มา
<http://sutir.sut.ac.th:8080/jspui/bitstream/123456789/8206/2/Fulltext.pdf>