



คัสตอมไอเอ (CUSTOMAI) แพลตฟอร์มสำหรับสร้างและปรับแต่งแบบจำลองเชิงลึกจากฐานข้อมูลรูปภาพ
CUSTOMAI PLATFORM FOR DEEP LEARNING MODEL CREATION AND CUSTOMIZATION FROM IMAGE DATASETS

นายภาณุเมธ คงสวัสดิ์เกียรติ 64070501041
นายชินพรรณ์ โกปารามศไตรสิน 64070501061
นายพัสกร ัญญวัฒน์กุล 64070501078

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR
THE DEGREE OF BACHELOR OF ENGINEERING (COMPUTER ENGINEERING)
FACULTY OF ENGINEERING
KING MONGKUT'S UNIVERSITY OF TECHNOLOGY THONBURI
2024

คัสตอมไอเอ (CustomAI) แพลตฟอร์มสำหรับสร้างและปรับแต่งแบบจำลองเชิงลึกจากฐานข้อมูลรูปภาพ
CustomAI Platform for Deep Learning Model Creation and Customization from Image Datasets

นายภาณุเมธ คงสวัสดิ์เกียรติ 64070501041
นายชินพรรณ์ โกปารามศรีไตรสิน 64070501061
นายพัสกร ธีญวัฒน์กุล 64070501078

A Project Submitted in Partial Fulfillment
of the Requirements for
the Degree of Bachelor of Engineering (Computer Engineering)
Faculty of Engineering
King Mongkut's University of Technology Thonburi
2024

Project Committee

..... (ดร.ปิยนิตย์ เวฬุลานนท์)	Project Advisor
..... (ดร.จตุรนต์ หาญสมบุรณ์)	Project Co-Advisor
..... (Asst.Prof. Committee2, Ph.D.)	Committee Member
..... (Asst.Prof. Committee3, Ph.D.)	Committee Member

Project Title	คัสตอมไอเอเอ (CustomAI) แพลตฟอร์มสำหรับสร้างและปรับแต่งแบบจำลองเชิงลึกจากฐานข้อมูลรูปภาพ CustomAI Platform for Deep Learning Model Creation and Customization from Image Datasets
Credits	3
Member(s)	นายภาณุเมธ คงสวัสดิ์เกียรติ 64070501041 นายชินพรรณ์ โกปารามะไตรสิน 64070501061 นายพัสกร ธีญวัฒน์กุล 64070501078
Project Advisor	ดร.ปิยนิตย์ เวฬุาลานนท์
Co-advisor	ดร.จาดรนต์ หาญสมบุรณ์
Program	Bachelor of Engineering
Field of Study	Computer Engineering
Department	Computer Engineering
Faculty	Engineering
Academic Year	2024

Abstract

In a multihop ad hoc network, the interference among nodes is reduced to maximize the throughput by using a smallest transmission range that still preserve the network connectivity. However, most existing works on transmission range control focus on the connectivity but lack of results on the throughput performance. This paper analyzes the per-node saturated throughput of an IEEE 802.11b multihop ad hoc network with a uniform transmission range. Compared to simulation, our model can accurately predict the per-node throughput. The results show that the maximum achievable per-node throughput can be as low as 11% of the channel capacity in a normal set of α operating parameters independent of node density. However, if the network connectivity is considered, the obtainable throughput will reduce by as many as 43% of the maximum throughput.

Keywords: Artificial Intelligent, AI / Image Pre-processing / Model Training / Models / Image Models / Neural Networks / Deep Learning / Data Augmentation

หัวข้อปริญญานิพนธ์	คัสตอมไอเอ (CustomAI) แพลตฟอร์มสำหรับสร้างและปรับแต่งแบบจำลองเชิงลึกจากฐานข้อมูลรูปภาพ CustomAI Platform for Deep Learning Model Creation and Customization from Image Datasets
หน่วยกิต	3
ผู้เขียน	นายภาณุเมธ คงสวัสดิ์เกียรติ นายชินพวรรณ โกปารามะไตรสิน นายพัศกร ธีญวัฒน์กุล
อาจารย์ที่ปรึกษา	ดร.ปิยนิตย์ เวฬุานนท์ ดร.จตุรนต์ หาญสมบูรณ์
หลักสูตร	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมคอมพิวเตอร์
คณะ	วิศวกรรมศาสตร์
ปีการศึกษา	2567

บทคัดย่อ

เซ็นเซอร์ เอ็กซ์เพรสเซอร์รับคอนเซ็ปต์สหัสวรรษเมจิก อิมแปร์ เฟรชชี ชาร์ปเซ็งเม้งคลาสสิก แพตเทิร์น แอลมอนต์ เฟลชว้อยกวัน ชาร์ดินชีเนิร์ส เซอร์อัสต์ สเตเดียมเพียบแปร์ไอย์แคมป์ส จัมป์ช็อตแมคเคอเรลล์ สดริง แมกกาซีนสดริงผ้าห่ม ฮัลโหล ยิม รอยล์ดี

คำสำคัญ: ปัญญาประดิษฐ์ / การประมวลผลรูปภาพเบื้องต้น / การฝึกฝนแบบจำลองปัญญาประดิษฐ์ / แบบจำลอง / แบบจำลองจากฐานข้อมูลรูปภาพ / เครือข่ายประสาทเทียม / การเรียนรู้ของเครื่องเชิงลึก / การเสริมข้อมูล

กิตติกรรมประกาศ

ขอบคุณอาจารย์ที่ปรึกษา กรรมการ พ่อแม่พี่น้อง และเพื่อนๆ คนที่ช่วยให้งานสำเร็จ ตามต้องการ

สารบัญ

หน้า

ABSTRACT	ii
บทคัดย่อ	iii
กิตติกรรมประกาศ	iv
สารบัญ	v
สารบัญตาราง	vii
สารบัญรูปภาพ	viii
สารบัญสัญลักษณ์	ix
สารบัญคำศัพท์ทางเทคนิคและคำย่อ	x
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญ	1
1.2 ประเภทของโครงงาน	1
1.3 โครงงานที่น่าสนใจ	1
1.3.1 วิธีการพัฒนาโครงงาน	1
1.3.2 จุดประสงค์ของโครงงาน	1
1.3.3 ขอบเขตของโครงงาน	2
1.4 เนื้อหาทางวิศวกรรมที่เป็นต้นฉบับ	2
1.5 ตารางการดำเนินงาน	2
1.6 แผนการดำเนินงานในภาคการศึกษาที่ 1	3
1.6.1 แผนการดำเนินงานเบื้องต้น	3
1.7 ผลการดำเนินงานในภาคการศึกษาที่ 2	3
1.7.1 แผนการดำเนินงานเบื้องต้น	3
1.7.2 แผนการดำเนินงานเชิงลึก	4
บทที่ 2 ทฤษฎีความรู้และงานที่เกี่ยวข้อง	6
2.1 ที่มาและความสำคัญ	6
2.2 ทฤษฎีที่เกี่ยวข้อง	6
2.2.1 โครงข่ายประสาทแบบคอนโวลูชัน (Convolutional Neural Network, CNN)	6
2.2.2 Random Forest	7
2.2.3 Decision Tree	7
2.2.4 Support Vector Machines (SVM)	8
2.2.5 k-Nearest Neighbors (k-NN)	8
2.2.6 การเสริมข้อมูล (Data Augmentation)	8
2.2.7 การประมวลผลรูปภาพเบื้องต้น (Image Pre-processing)	8
2.2.8 การสกัดคุณลักษณะและการเลือกคุณลักษณะ (Feature Extraction & Feature Selection)	9
2.3 ภาษาคอมพิวเตอร์และเทคโนโลยี	9
2.3.1 ภาษาคอมพิวเตอร์ที่เลือกใช้	9
2.3.2 เทคโนโลยีที่เลือกใช้	10
2.4 ผลิตภัณฑ์ที่เกี่ยวข้อง	11
2.4.1 เว็บไซต์ Roboflow	11
2.4.2 อัลกอริทึม II	11
2.4.2.1 ขั้นตอนที่ 1	11
2.4.2.2 ขั้นตอนที่ 2	11

2.5 เครื่องมือที่ใช้ในการพัฒนา	11
บทที่ 3 วิธีการดำเนินงาน	13
3.1 ข้อกำหนดและความต้องการของระบบ	13
3.2 สถาปัตยกรรมระบบ	13
3.3 Hardware Module 1	13
3.3.1 Component 1	13
3.3.2 Logical Circuit Diagram	13
3.4 Hardware Module 2	13
3.4.1 Component 1	13
3.4.2 Component 2	13
3.5 Path Finding Algorithm	13
3.6 Database Design	13
3.7 UML Design	13
3.8 GUI Design	13
3.9 การออกแบบการทดลอง	13
3.9.1 ตัวชี้วัดและปัจจัยที่ศึกษา	13
3.9.2 รูปแบบการเก็บข้อมูล	13
บทที่ 4 ผลการดำเนินงาน	14
4.1 ประสิทธิภาพการทำงานของระบบ	14
4.2 ความพึงพอใจการใช้งาน	14
4.3 การวิเคราะห์ข้อมูลและผลการทดลอง	14
บทที่ 5 บทสรุป	15
5.1 สรุปผลโครงงาน	15
5.2 ปัญหาที่พบและการแก้ไข	15
5.3 ข้อจำกัดและข้อเสนอแนะ	15
หนังสืออ้างอิง	16
APPENDIX	17
ก ชื่อภาคผนวกที่ 1	18
ข ชื่อภาคผนวกที่ 2	20

สารบัญตาราง

ตารางที่	หน้า
1.1 ตารางแสดงแผนการดำเนินงานแต่ละสัปดาห์ระหว่างเดือนสิงหาคมถึงเดือนธันวาคม 2567	3
1.2 ตารางแสดงแผนการดำเนินงานแต่ละสัปดาห์ระหว่างเดือนมกราคมถึงเดือนเมษายน 2568	4
1.3 ตารางแสดงแผนการดำเนินการฝั่งเว็บไซต์ส่วนหลังปัญญาประดิษฐ์แต่ละสัปดาห์ระหว่างเดือนมกราคมถึงเดือนเมษายน 2568	4
1.4 ตารางแสดงแผนการดำเนินการฝั่งเว็บไซต์ส่วนหลังแต่ละสัปดาห์ระหว่างเดือนมกราคมถึงเดือนเมษายน 2568	4
1.5 ตารางแสดงแผนการดำเนินการฝั่งเว็บไซต์ส่วนหน้าแต่ละสัปดาห์ระหว่างเดือนมกราคมถึงเดือนเมษายน 2568	5
1.6 ตารางแสดงแผนการดำเนินการส่วนอื่น ๆ แต่ละสัปดาห์ระหว่างเดือนมกราคมถึงเดือนเมษายน 2568	5
3.1 test table x1	13

สารบัญรูป

รูปที่		หน้า
2.1	The network model	12
ก.1	This is the figure x11 ทดสอบ จาก https://www.google.com	18
ข.1	This is the figure x11 ทดสอบ จาก https://www.google.com	20

สารบัญสัญลักษณ์

SYMBOL

α	Test variable
λ	Interarival rate
μ	Service rate

UNIT

m^2
jobs/second
jobs/second

สารบัญคำศัพท์ทางเทคนิคและคำย่อ

CRUD	=	Create, Read, Update, Delete
UI/UX	=	User experience and user interface
AI	=	Artificial intelligent
CNN	=	Convolutional Neural Network
ERD	=	Entity Relations Diagram

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญ

ในปัจจุบันปัญญาประดิษฐ์ได้เข้ามามีบทบาทสำคัญในหลากหลายสาขาอาชีพมากขึ้น เนื่องจากปัญญาประดิษฐ์สามารถช่วยเพิ่มประสิทธิภาพในการทำงานบางประเภทให้รวดเร็วและแม่นยำ เช่น การวินิจฉัยทางการแพทย์ การจำแนกรูปภาพ และการตีความข้อมูลจากรูปภาพ อย่างไรก็ตาม ปัญญาประดิษฐ์แต่ละตัวถูกสร้างขึ้นเพื่อเป้าหมายที่เฉพาะเจาะจง หากนำไปใช้งานผิดจุดประสงค์จะทำให้ทำงานได้ไม่เต็มประสิทธิภาพ ตัวอย่างเช่น ปัญญาประดิษฐ์ที่พัฒนาขึ้นเพื่อจดจำวัตถุทั่วไปในรูปภาพอาจไม่สามารถตรวจจับใบหน้าได้อย่างแม่นยำ และไม่สามารถนำไปวิเคราะห์รูปภาพทางการแพทย์ได้อย่างมีประสิทธิภาพ ซึ่งหมายความว่าหากสามารถสร้างปัญญาประดิษฐ์ที่เฉพาะเจาะจงกับปัญหาได้ ปัญญาประดิษฐ์นั้นอาจมีประสิทธิภาพมากกว่าปัญญาประดิษฐ์ที่สร้างมาเพื่อแก้ไขปัญหาย่อยๆ แต่กระบวนการพัฒนาแบบจำลองปัญญาประดิษฐ์เหล่านี้ยังคงเป็นเรื่องยากและต้องการความชำนาญสูง ทำให้บุคคลทั่วไปไม่สามารถสร้างแบบจำลองได้หากขาดความเข้าใจและความสามารถในการเขียนโปรแกรม

ด้วยเหตุนี้ คณะผู้จัดทำจึงต้องการสร้างทางเลือกให้บุคคลทั่วไปสามารถเข้าถึงเทคโนโลยีปัญญาประดิษฐ์จากฐานข้อมูลรูปภาพได้ง่ายขึ้น จึงเกิดเป็นความตั้งใจในการพัฒนาแพลตฟอร์มเพื่อช่วยให้ผู้ใช้สามารถสร้างแบบจำลองปัญญาประดิษฐ์จากฐานข้อมูลรูปภาพ โดยผู้ใช้งานแพลตฟอร์มไม่จำเป็นต้องมีความรู้ด้านการเขียนโปรแกรม นอกจากนี้ผู้ใช้งานยังสามารถศึกษาและเรียนรู้เกี่ยวกับการสร้างและปรับแต่งแบบจำลองปัญญาประดิษฐ์จากฐานข้อมูลรูปภาพ สร้างความรู้และความเข้าใจในการสร้างแบบจำลองปัญญาประดิษฐ์ ทำให้ผู้ใช้สามารถพัฒนาแบบจำลองที่ตอบโจทย์ปัญหาเฉพาะของตนได้อย่างมีประสิทธิภาพสูงสุด

1.2 ประเภทของโครงการ

ต้นแบบผลิตภัณฑ์ทางการค้า

แพลตฟอร์มนี้สามารถสร้างแบบจำลองปัญญาประดิษฐ์จากฐานข้อมูลรูปภาพด้วยชุดข้อมูลที่ผู้ใช้งานต้องการ มีความยืดหยุ่นในการปรับแต่งกระบวนการประมวลผลรูปภาพและการฝึกฝนแบบจำลองปัญญาประดิษฐ์ที่หลากหลาย ซึ่งผู้ใช้งานยังสามารถใช้งานแบบจำลองที่สร้างขึ้นเองบนเว็บไซต์ได้ อีกทั้งแพลตฟอร์มยังมีเนื้อหาสำหรับการศึกษาเกี่ยวกับกระบวนการสร้างและพัฒนาแบบจำลองปัญญาประดิษฐ์ เพื่อสนับสนุนการเรียนรู้และเพิ่มทักษะให้กับผู้ใช้งาน

1.3 โครงการที่น่าสนใจ

1.3.1 วิธีการพัฒนาโครงการ

สกรัม (Scrum) ระเบียบวิธีการพัฒนาซอฟต์แวร์แบบเอจายล์ (Agile) ที่มุ่งเน้นการบริหารโครงการพัฒนาซอฟต์แวร์และการวางแผนงานระยะเวลาของการพัฒนาระบบในแต่ละส่วน สกรัมเป็นระเบียบการพัฒนาซอฟต์แวร์ที่มีความยืดหยุ่นและง่ายต่อการวางแผนและติดตามงาน [1]

1.3.2 จุดประสงค์ของโครงการ

1. เพื่อให้ผู้ใช้งานสามารถสร้างแบบจำลองปัญญาประดิษฐ์จากฐานข้อมูลรูปภาพบนแพลตฟอร์มได้
2. เพื่อให้ผู้ใช้งานใช้แพลตฟอร์มได้โดยไม่ต้องมีความรู้ด้านการเขียนโปรแกรม
3. เพื่อให้ผู้ใช้งานสามารถปรับแต่งตัวแปรที่มีผลต่อการสร้างและใช้งานแบบจำลองได้ตามที่ต้องการ ประกอบไปด้วย ตัวแปรสำหรับการประมวลผลรูปภาพ และตัวแปรในการฝึกฝนแบบจำลอง
4. เพื่อให้ผู้ใช้งานที่ไม่มีความรู้ด้านการสร้างแบบจำลองปัญญาประดิษฐ์จากฐานข้อมูลรูปภาพ สามารถเรียนรู้เนื้อหาที่เกี่ยวข้อง และฝึกฝนทักษะบนแพลตฟอร์มที่จะพัฒนาขึ้นได้

1.3.3 ขอบเขตของโครงการ

1. พัฒนาแพลตฟอร์มในรูปแบบเว็บแอปพลิเคชัน
2. แพลตฟอร์มที่มีผู้ใช้งานหลักคือผู้ที่ต้องการสร้างแบบจำลองปัญญาประดิษฐ์จากฐานข้อมูลรูปภาพ
3. ฟังก์ชันภายในแพลตฟอร์มเป็นฟังก์ชันที่มีการพัฒนาขึ้น และนำมาใช้อย่างแพร่หลายโดยอ้างอิงจากข้อมูลที่สืบค้นได้
4. แพลตฟอร์มสามารถทำการติดป้ายกำกับข้อมูล (Data Annotation) ให้กับชุดข้อมูลในระบบได้ทั้งรูปแบบจำแนกประเภทรูปภาพ (Classification), แบบตรวจจับวัตถุภายในรูปภาพ (Object Detection) และแบบแบ่งส่วนภาพ (Segmentation)
5. แพลตฟอร์มสามารถปรับแต่งอัตราส่วนชุดข้อมูลในการฝึกฝนแบบจำลองปัญญาประดิษฐ์ได้ (Train / Test Ratio)
6. ชนิดของรูปที่นำเข้าสู่แพลตฟอร์มจะต้องอยู่ในรูปแบบของ JPG, JPEG, PNG, WEBP, และ SVG
7. แบบจำลองที่สามารถสร้างได้บนระบบจะเป็นแบบจำลองที่เรียนรู้จากชุดข้อมูลที่มีผู้สอน (Supervised Learning) เท่านั้น
8. การเรียนรู้บนแพลตฟอร์มเป็นรูปแบบของบทเรียนเชิงโต้ตอบ (Interactive Tutorials) ที่เน้นการฝึกปฏิบัติและใช้งานเพื่อให้ผู้ใช้งานมีความรู้เบื้องต้น ไม่เน้นความเข้าใจเชิงลึก
9. เนื้อหาที่สามารถเรียนรู้ได้ในระบบจะอ้างอิงจากฟังก์ชันที่มีในแพลตฟอร์มเป็นหลัก
10. มีแบบฝึกหัดเพื่อวัดผลลัพธ์การเรียนรู้ (Learning Outcomes) จากความแม่นยำของแบบจำลองที่ถูกสร้างขึ้นผ่านการทดสอบด้วยชุดข้อมูลในแต่ละแบบฝึกหัด และการใช้ความรู้ที่ได้จากบทเรียน

1.4 เนื้อหาทางวิศวกรรมที่เป็นต้นฉบับ

โครงการนี้พัฒนาขึ้นจากการใช้ความรู้ในการพัฒนาเว็บไซต์ การสร้างปัญญาประดิษฐ์ การจัดการฐานข้อมูลและวิศวกรรมซอฟต์แวร์ เพื่อสร้างแพลตฟอร์มที่ผู้ใช้งานสามารถสร้างแบบจำลองปัญญาประดิษฐ์จากฐานข้อมูลรูปภาพได้โดยไม่ต้องเขียนโปรแกรม และเปิดโอกาสให้ผู้ใช้งานสามารถปรับแต่งโครงสร้างแบบจำลองและการประมวลผลรูปภาพได้มากที่สุด

1.5 ตารางการดำเนินงาน

ขั้นตอนการดำเนินงานมีดังนี้

1. ศึกษาข้อมูลของหัวข้อโครงการที่ได้รับ
 - a. หาข้อมูลที่เกี่ยวข้องกับโครงการผ่านอินเทอร์เน็ตและปรึกษาอาจารย์ที่ปรึกษา
 - b. ศึกษาตลาดและทดลองใช้แพลตฟอร์มที่มีความใกล้เคียงกับโครงการเพื่อทำการเปรียบเทียบฟังก์ชันความสามารถของแต่ละเว็บไซต์
2. จัดทำข้อเสนอหัวข้อโครงการ (Project Idea)
3. จัดทำข้อเสนอโครงการ (Project Proposal)
4. ศึกษาเทคโนโลยีและซอฟต์แวร์ที่จะใช้
 - a. เว็บไซต์ส่วนหน้า (Front-End)
 - b. เว็บไซต์ส่วนหลัง (Back-End)
 - c. การติดป้ายกำกับข้อมูล
 - d. การประมวลผลรูปภาพเบื้องต้น
 - e. การเสริมข้อมูล
 - f. การฝึกฝนปัญญาประดิษฐ์สำหรับฐานข้อมูลรูปภาพ

5. การออกแบบระบบ

- ออกแบบโครงสร้างของซอฟต์แวร์และรูปแบบการใช้งาน
- ออกแบบฐานข้อมูลและแบบจำลองอธิบายโครงสร้างฐานข้อมูลเชิงสัมพันธ์
- (Entity Relations Diagram, ERD) เพื่อรองรับข้อมูลที่เกี่ยวข้อง
- ออกแบบฟังก์ชันที่สามารถใช้งานได้ในแพลตฟอร์ม
- ออกแบบส่วนต่อประสานกับผู้ใช้
- ออกแบบส่วนต่อประสานกับผู้ใช้ให้เหมาะสม ใช้งานง่าย และผู้ใช้งานสามารถเรียนรู้การใช้งานได้รวดเร็ว
- ออกแบบระบบการฝึกฝนแบบจำลอง

6. จัดทำรายงานประจำภาคการศึกษาที่ 1

7. การพัฒนาและทดสอบระบบ

- พัฒนาแต่ละฟีเจอร์ตามแผนที่กำหนดไว้
- ทดสอบหาข้อผิดพลาดของระบบและทำการแก้ไขหากตรวจพบ

8. ตรวจสอบระบบครั้งสุดท้าย

9. การนำระบบขึ้นสู่สาธารณะ

- จัดเตรียมเซิร์ฟเวอร์สำหรับการนำขึ้นสู่สาธารณะและตั้งค่าสภาพแวดล้อมในการทำงานต่าง ๆ
- ทำการนำระบบขึ้นโดยใช้เครื่องมือที่เหมาะสม (เช่น Docker, Kubernetes)
- ตรวจสอบหลังนำระบบขึ้นสำเร็จ และทำการปรับแต่งหากพบปัญหา

10. จัดทำรายงานประจำภาคการศึกษาที่ 2

11. นำเสนอผลงาน

1.6 แผนการดำเนินงานในภาคการศึกษาที่ 1

1.6.1 แผนการดำเนินงานเบื้องต้น

แผนการดำเนินงานเบื้องต้นในภาคการศึกษาที่ 1 แสดงได้ดังตารางที่ 1.1.

ตารางที่ 1.1 ตารางแสดงแผนการดำเนินงานแต่ละสัปดาห์ระหว่างเดือนสิงหาคมถึงเดือนธันวาคม 2567

แผนการดำเนินงาน แต่ละสัปดาห์	สิงหาคม 2567				กันยายน 2567				ตุลาคม 2567				พฤศจิกายน 2567				ธันวาคม 2567			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1. ศึกษาข้อมูลของหัวข้อโครงการที่ได้รับ																				
2. จัดทำข้อเสนอหัวข้อโครงการ (Project Idea)																				
3. จัดทำข้อเสนอโครงการ (Project Proposal)																				
4. ศึกษาเทคโนโลยีและซอฟต์แวร์ที่จะใช้																				
5. ออกแบบระบบ																				
6. จัดทำรายงานประจำภาคการศึกษาที่ 1																				

1.7 ผลการดำเนินงานในภาคการศึกษาที่ 2

1.7.1 แผนการดำเนินงานเบื้องต้น

แผนการดำเนินงานเบื้องต้นในภาคการศึกษาที่ 2 แสดงได้ดังตารางที่ 1.2.

ตารางที่ 1.2 ตารางแสดงแผนการดำเนินงานแต่ละสัปดาห์ระหว่างเดือนมกราคมถึงเดือนเมษายน 2568

แผนการดำเนินงาน แต่ละสัปดาห์	มกราคม 2568				กุมภาพันธ์ 2568				มีนาคม 2568				เมษายน 2568			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1. พัฒนาและทดสอบระบบ																
2. ตรวจสอบระบบครั้งสุดท้าย																
3. นำระบบขึ้นสู่สาธารณะ																
4. จัดทำรายงานประจำภาคการศึกษาที่ 2																
5. นำเสนอผลงาน																

1.7.2 แผนการดำเนินงานเชิงลึก

แผนการดำเนินงานอย่างละเอียดในภาคเรียนที่ 2 โดยจะประกอบไปด้วยทั้งหมด 4 ตารางได้แก่

1. ตารางที่ 1.3. ฝั่งเว็บไซต์ส่วนหลังปัญญาประดิษฐ์

ตารางที่ 1.3 ตารางแสดงแผนการดำเนินการฝั่งเว็บไซต์ส่วนหลังปัญญาประดิษฐ์แต่ละสัปดาห์ระหว่างเดือนมกราคมถึงเดือนเมษายน 2568

แผนการดำเนินงาน แต่ละสัปดาห์	มกราคม 2568				กุมภาพันธ์ 2568				มีนาคม 2568				เมษายน 2568			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1. ระบบจัดการชุดข้อมูล																
2. ระบบการประมวลผลรูปภาพเบื้องต้น																
3. ระบบสกัดคุณลักษณะและการเลือกคุณลักษณะ																
4. ระบบเสริมข้อมูล																
5. ระบบสร้างแบบจำลองปัญญาประดิษฐ์																
6. ระบบฝึกฝนแบบจำลอง																
7. ระบบวัดผลแบบจำลอง																
8. ระบบใช้งานแบบจำลองปัญญาประดิษฐ์																

2. ฝั่งเว็บไซต์ส่วนหลัง (Backend - Web Service) มีแผนการดำเนินงานดังตารางที่ 1.4.

ตารางที่ 1.4 ตารางแสดงแผนการดำเนินการฝั่งเว็บไซต์ส่วนหลังแต่ละสัปดาห์ระหว่างเดือนมกราคมถึงเดือนเมษายน 2568

แผนการดำเนินงาน แต่ละสัปดาห์	มกราคม 2568				กุมภาพันธ์ 2568				มีนาคม 2568				เมษายน 2568			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1. จัดเตรียมฐานข้อมูลทั้งหมด																
2. ระบบยืนยันตัวตน																
3. ระบบอัปโหลดไฟล์																
4. ระบบอัปโหลดรูปภาพ																
5. ระบบอัปโหลดชุดข้อมูล																
6. CRUD ข้อมูลระบบเสริมข้อมูล																
7. CRUD ข้อมูลระบบสร้างแบบจำลอง																
8. CRUD ข้อมูลฝึกฝนแบบจำลอง																
9. CRUD ข้อมูลผลการฝึกฝนแบบจำลอง																
10. CRUD การเรียนรู้ - เรียนรู้																
11. CRUD การเรียนรู้ - แบบฝึกหัด																
12. ระบบผู้ดูแลระบบ (Admin)																
13. ระบบหน้ากระดานสรุปการใช้งาน (Dashboard)																
14. ระบบบันทึกข้อมูล (Logging)																

3. ฝั่งเว็บไซต์ส่วนหน้า (Frontend) มีแผนการดำเนินงานดังตารางที่ 1.5.

บทที่ 2 ทฤษฎีความรู้และงานที่เกี่ยวข้อง

เนื้อหาบทที่ 2 นี้จะอธิบายถึงความสำคัญของปัญหาที่เป็นแรงจูงใจในการดำเนินโครงการนี้ รวมถึงวิธีการและขั้นตอนที่ได้ถูกนำมาใช้ในการแก้ไขปัญหาดังกล่าว ซึ่งประกอบไปด้วยทฤษฎีที่เกี่ยวข้องที่ถูกศึกษาและนำมาประยุกต์ใช้เพื่อให้บรรลุวัตถุประสงค์ของโครงการ นอกจากนี้ยังครอบคลุมถึงการเลือกใช้ภาษาคอมพิวเตอร์และเครื่องมือในการพัฒนาที่เหมาะสมกับการใช้แก้ไขปัญหามา ตลอดจนการอ้างอิงงานวิจัยที่เกี่ยวข้องซึ่งได้ถูกนำมาเป็นฐานความรู้และแนวทางในการพัฒนาโครงการนี้

2.1 ที่มาและความสำคัญ

จากการศึกษาผลิตภัณฑ์ทางการค้าที่สามารถสร้างแบบจำลองจากฐานข้อมูลรูปภาพบนแพลตฟอร์มได้ ในปัจจุบันแพลตฟอร์มส่วนใหญ่ยังมีข้อจำกัดในการใช้งาน เช่น ไม่สามารถปรับแต่งโครงสร้างแบบจำลองได้โดยละเอียด หรือไม่สามารถทำให้ผู้ใช้เข้าใจถึงวิธีการใช้งานแพลตฟอร์มได้มากเพียงพอ จึงทำให้ผู้ใช้งานสร้างแบบจำลองจากฐานข้อมูลรูปภาพได้ไม่เต็มประสิทธิภาพ ส่งผลให้แบบจำลองที่สร้างขึ้น ไม่ตอบโจทย์ความต้องการของผู้ใช้งาน

เพื่อแก้ไขปัญหาดังกล่าว แพลตฟอร์มในโครงการนี้จึงออกแบบมาให้ผู้ใช้งานสามารถทำความเข้าใจการใช้งานฟังก์ชันต่าง ๆ ได้ง่าย มีเนื้อหาและแบบฝึกหัดสำหรับวัดผลทักษะการเรียนรู้ ทำให้ผู้ใช้งานที่ไม่มีพื้นฐานในการเขียนโปรแกรมสามารถใช้งานได้ เพียงเข้าใจความสามารถของแต่ละฟังก์ชัน โดยฟังก์ชันต่าง ๆ ได้ถูกพัฒนาขึ้นเพื่อเพิ่มความหลากหลายและความยืดหยุ่นในกระบวนการสร้างแบบจำลองจากฐานข้อมูลรูปภาพ ส่งผลให้ผู้ใช้งานสามารถปรับแต่งการประมวลผลรูปภาพเบื้องต้น ปรับแต่งชุดข้อมูลรูปภาพ และตั้งค่าตัวแปรสำหรับการฝึกฝนแบบจำลอง ทั้งหมดเพื่อให้ผู้ใช้งานสามารถสร้างแบบจำลองได้อย่างอิสระภายใต้ฟังก์ชันที่มีให้ และตรงกับความต้องการของตนเองได้มากที่สุด

2.2 ทฤษฎีที่เกี่ยวข้อง

2.2.1 โครงข่ายประสาทแบบคอนโวลูชัน (Convolutional Neural Network, CNN)

โครงข่ายประสาทเทียมที่ออกแบบมาเพื่อการประมวลผลและวิเคราะห์รูปภาพโดยเฉพาะ โดยทำงานผ่านการใช้เลเยอร์คอนโวลูชัน (Convolutional Layer) ที่ทำหน้าที่สกัดคุณลักษณะจากรูปภาพ พิลเตอร์จะเลื่อนผ่านรูปภาพเพื่อสร้างผังคุณลักษณะ (Feature Map) ใช้สำหรับจำแนกข้อมูล นอกจากนี้ยังมีเลเยอร์รวมค่า (Pooling Layer) ที่ช่วยลดขนาดข้อมูลและเลเยอร์เชื่อมต่อแบบเต็ม (Fully Connected Layer) ที่ใช้ในการตัดสินใจ ผลลัพธ์ขั้นสุดท้าย CNN ถูกใช้อย่างแพร่หลายในงานด้านการจดจำรูปภาพและวัตถุ [2] โดย CNN ได้ถูกออกแบบจนมีแบบจำลองพื้นฐานที่แตกต่างกันเพื่อใช้ในหลากหลายสถานการณ์ได้ดังนี้

1. LeNet-5

LeNet เป็นสถาปัตยกรรม CNN รุ่นแรกที่ถูกสร้างโดยทีม Yann LeCun ปี 1990 ทำหน้าที่เป็นรากฐานสำหรับ CNN รุ่นหลัง ๆ จำนวนมาก และถือเป็นสถาปัตยกรรมคลาสสิกและเรียบง่ายสำหรับงานด้านการจดจำรูปภาพ

2. AlexNet

AlexNet เป็นสถาปัตยกรรมแรกๆ ที่แสดงให้เห็นว่า CNN สามารถทำงานได้ดีกว่าวิธีการเรียนรู้ของเครื่องแบบดั้งเดิมเป็นอย่างมาก ในงานการจดจำรูปภาพและเป็นต้นแบบสำคัญของแบบจำลองอื่น ๆ

3. VGGNet

VGG เป็นสถาปัตยกรรมที่ขึ้นชื่อเรื่องความเรียบง่ายของแบบจำลอง สามารถใช้ VGG เพื่อตรวจจับและจำแนกวัตถุต่าง ๆ เช่นรถ ขับเคลื่อนอัตโนมัติบนท้องถนนเพื่อตรวจจับคนเดิน ถนน และป้ายจราจร

4. GoogLeNet

GoogLeNet เป็นเครื่องมือที่มีประสิทธิภาพสำหรับการจำแนกรูปภาพ และกำลังถูกนำมาใช้ในแพลตฟอร์มที่หลากหลาย เช่น GoogleNet สามารถใช้จำแนกรูปภาพเป็นหมวดหมู่ต่าง ๆ เช่น แมวและสุนัข รถยนต์และรถบรรทุก และดอกไม้และสัตว์

5. ResNet

ResNet หรือ Residual Networks มักใช้สำหรับงานตรวจจับจุดสำคัญ การตรวจจับจุดสำคัญคืองานระบุตำแหน่งจุดเฉพาะบนวัตถุในรูปภาพ ตัวอย่างเช่น การตรวจจับจุดสำคัญสามารถใช้เพื่อระบุตำแหน่งดวงตา จมูก และปากบนใบหน้าของมนุษย์ได้

6. MobileNet

MobileNet ถูกออกแบบขึ้นโดยบริษัท Google เพื่อใช้ในอุปกรณ์ขนาดเล็ก จึงมีข้อได้เปรียบเรื่องขนาดแบบจำลองที่เล็กและใช้ทรัพยากรการประมวลผลน้อยกว่าแบบจำลองอื่น ๆ แต่ยังมีความแม่นยำสูงและตรวจจับจุดสำคัญได้ดี [18]

7. YOLO (You Only Look Once)

อัลกอริทึมสำหรับการตรวจจับวัตถุในรูปที่มีความเร็วสูง โดยทำการประมวลผลรูปภาพเพียงครั้งเดียว (Single Pass) แล้วแยกวัตถุที่ต้องการในรูปภาพพร้อมกับระบุขอบเขตของวัตถุ (Bounding Box) จุดเด่นของ YOLO คือความรวดเร็วและความสามารถในการตรวจจับวัตถุหลายชนิดในรูปภาพเดียวกัน ซึ่งหลักการทำงานของ YOLO แบ่งเป็นสามขั้นตอนได้แก่

- แบ่งรูปภาพ: แบ่งรูปภาพออกเป็นส่วนเล็ก ๆ หลาย ๆ ส่วน
- ทำนาย: แต่ละส่วนจะทำนายว่ามีวัตถุอยู่หรือไม่ และถ้ามี จะทำนายตำแหน่งและประเภทของวัตถุนั้น
- รวมผล: ผลลัพธ์จากทุกส่วนจะถูกนำมารวมกันเพื่อได้ผลลัพธ์สุดท้าย

YOLO เป็นอัลกอริทึมที่มีจุดเด่นที่ความรวดเร็ว แม่นยำ และครอบคลุมได้หลายวัตถุในรูปภาพเดียวกัน แต่ก็มีข้อจำกัดที่ถ้าหากวัตถุมีขนาดเล็กและอยู่ใกล้กันมากเกินไปก็อาจทำให้ไม่สามารถตรวจพบได้ และหากรูปร่างของวัตถุมีความแปลกประหลาดมากจนเกินไปก็อาจตรวจจับไม่ได้เช่นกัน [14]

โดยข้อดีของ CNN จะประกอบไปด้วย

1. CNN มีความแม่นยำได้เป็นอันดับต้น ๆ ของงานด้านการจดจำรูปภาพ เช่น การจำแนกรูปภาพ การตรวจจับวัตถุ และการแบ่งส่วนรูปภาพ
2. CNN มีประสิทธิภาพสูง โดยเฉพาะเมื่อนำไปใช้งานบนฮาร์ดแวร์เฉพาะเช่น หน่วยประมวลผลกราฟิก (Graphics Processing Unit, GPU)
3. CNN ค่อนข้างทนทานต่อข้อมูลรบกวนและการเปลี่ยนแปลงข้อมูลขาเข้า
4. CNN สามารถปรับให้เหมาะกับงานที่หลากหลายได้ เพียงเปลี่ยนแค่สถาปัตยกรรมแบบจำลอง [16]

แต่มีข้อเสีย ได้แก่

1. CNN อาจมีความซับซ้อนและฝึกฝนได้ยาก โดยเฉพาะกับชุดข้อมูลที่มีขนาดใหญ่
2. CNN อาจต้องใช้ทรัพยากรคอมพิวเตอร์จำนวนมากในการฝึกฝนและใช้งานแบบจำลอง
3. CNN ต้องใช้ข้อมูลที่มีป้ายกำกับจำนวนมากจึงจะฝึกฝนได้
4. CNN อาจตีความลักษณะของรูปภาพบางรูปแบบได้ยากทำให้ยากต่อการเข้าใจ

2.2.2 Random Forest

Random Forest เป็นวิธีการเรียนรู้แบบรวม (Ensemble Learning) ที่ปรับปรุงประสิทธิภาพของ Decision Tree ด้วยการรวมผลลัพธ์จาก Decision Tree หลายตัว เพื่อให้ได้การพยากรณ์ที่แม่นยำยิ่งขึ้น โดยใช้หลักการสุ่มตัวอย่างข้อมูลและฟีเจอร์ เพื่อสร้าง Decision Tree ที่ไม่สัมพันธ์กัน ลดปัญหา Overfitting และเพิ่มความแม่นยำ Random Forest สามารถใช้ได้ทั้งในงานการจำแนกประเภทและการถดถอย โดยในงานจำแนกประเภท ผลลัพธ์จะมาจากค่าเฉลี่ยของผลลัพธ์ของแต่ละ Decision Tree [30]

2.2.3 Decision Tree

Decision Tree เป็นอัลกอริทึมการเรียนรู้แบบมีผู้สอนที่ใช้สำหรับงานการจำแนกประเภทและการถดถอย โดยทำงานผ่านการแบ่งชุดข้อมูลออกเป็นกลุ่มย่อยตามค่าของฟีเจอร์ต่าง ๆ เพื่อสร้างโครงสร้างแบบ Decision Tree โหนดการตัดสินใจ (Decision Node) แทนคำถามเกี่ยวกับฟีเจอร์ ขณะที่กิ่งไม้แทนผลลัพธ์ที่เป็นไปได้ และใบไม้ (Leaf Node) จะให้คำตอบสุดท้าย ตัวอย่างเช่น ในปัญหาการจำแนกประเภท Decision Tree อาจใช้ตัดสินว่าควรเล่นเซิร์ฟหรือไม่ โดยพิจารณาจากคำถามเกี่ยวกับสภาพคลื่นหรือลม แม้ Decision Tree จะเข้าใจง่ายและใช้งานง่าย แต่มักจะประสบปัญหา Overfitting และ Bias ซึ่งอาจทำให้ประสิทธิภาพการพยากรณ์ลดลงได้ [31]

2.2.4 Support Vector Machines (SVM)

Support Vector Machine (SVM) คืออัลกอริทึมการเรียนรู้แบบมีผู้สอนที่ใช้สำหรับการจำแนกประเภทและการถดถอย โดยมีเป้าหมายหลักในการหาขอบเขตการตัดสินใจ (Decision Boundary) หรือไฮเปอร์เพลน (Hyperplane) ที่เหมาะสมที่สุด ซึ่งจะช่วยแยกข้อมูลของแต่ละคลาสออกจากกันในพื้นที่ N มิติ ไฮเปอร์เพลนนี้ถูกออกแบบมาเพื่อเพิ่มระยะห่างระหว่างข้อมูลที่อยู่ใกล้ที่สุดของคลาสที่แตกต่างกัน (เรียกว่า Support Vectors) ซึ่งช่วยให้ SVM มีความสามารถในการพยากรณ์ที่แม่นยำและสามารถทำงานได้ดีกับข้อมูลใหม่

SVM สามารถจัดการกับข้อมูลที่แบ่งแยกกันไม่ได้ทั้งเชิงเส้น (Linear) และไม่เชิงเส้น (Nonlinear) สำหรับกรณีที่ข้อมูลไม่สามารถแบ่งแยกได้ในพื้นที่เชิงเส้น จะมีการใช้ฟังก์ชันเคอร์เนล (Kernel Function) เพื่อเปลี่ยนข้อมูลให้เป็นมิติที่สูงขึ้น ซึ่งช่วยให้สามารถแยกข้อมูลในลักษณะเชิงเส้นได้ในมิตินั้น เทคนิคนี้เรียกว่า “Kernel Trick” โดยประเภทของเคอร์เนลที่ใช้ เช่น เคอร์เนลเชิงเส้น (Linear Kernel), เคอร์เนลพหุนาม (Polynomial Kernel), และเคอร์เนล RBF (Radial Basis Function Kernel) เป็นต้น [32]

2.2.5 k-Nearest Neighbors (k-NN)

KNN เป็นอัลกอริทึมการเรียนรู้แบบมีผู้สอน (Supervised Learning) ที่ไม่ใช่พารามตริก (Non-Parametric) ใช้สำหรับการจำแนกประเภท (Classification) และการถดถอย (Regression) โดยอาศัยความใกล้เคียงของข้อมูลในการตัดสินใจ ตัวอัลกอริทึมทำงานบนสมมติฐานที่ว่า ข้อมูลที่คล้ายกันจะอยู่ใกล้กันในพื้นที่ข้อมูล

ในการจำแนกประเภท KNN จะกำหนดคลาสให้กับจุดข้อมูลใหม่โดยการโหวตจากเพื่อนบ้านที่ใกล้ที่สุด K จุด โดยใช้ “เสียงข้างมาก” เพื่อระบุคลาสของข้อมูลนั้น ส่วนในการถดถอย KNN จะใช้ค่าเฉลี่ยของเพื่อนบ้านที่ใกล้ที่สุดเพื่อคาดการณ์ค่าที่ต่อเนื่อง นอกจากนี้ KNN ยังถือเป็นแบบจำลองแบบ “Lazy Learning” เพราะไม่มีขั้นตอนการฝึก (Training) ล่วงหน้า โดยจะเก็บข้อมูลการฝึกไว้ทั้งหมดและประมวลผลเฉพาะเมื่อมีการทำนาย

KNN นิยมใช้ในระบบแนะนำ (Recommendation Systems), การจดจำรูปแบบ (Pattern Recognition), การทำเหมืองข้อมูล (Data Mining), และการตรวจจับการบุกรุก (Intrusion Detection) อย่างไรก็ตาม ข้อจำกัดของ KNN คือประสิทธิภาพที่ลดลงเมื่อขนาดของชุดข้อมูลเพิ่มขึ้น เนื่องจากต้องคำนวณระยะห่างกับทุกจุดข้อมูลในชุดฝึกฝน [33]

2.2.6 การเสริมข้อมูล (Data Augmentation)

การเสริมข้อมูล คือเทคนิคที่ช่วยเพิ่มประสิทธิภาพของการเรียนรู้ของเครื่องเชิงลึก (Deep Learning) โดยการสร้างข้อมูลใหม่จากข้อมูลที่มีอยู่เดิม เช่น การหมุนรูปภาพ เปลี่ยนสี และตัดส่วนหนึ่งของรูปภาพออก ทำให้แบบจำลองได้เรียนรู้จากข้อมูลที่หลากหลายมากขึ้น ลดปัญหาการเรียนรู้จำ (Overfitting) และทำให้แบบจำลองมีความแม่นยำและทนทานต่อข้อมูลที่ไม่เคยพบเจอมาก่อน ตัวอย่างเทคนิคที่ใช้ทางรูปภาพ:

1. การแปลงทางเรขาคณิต: หมุน พลิก ปรับขนาด ตัด
2. การแปลงสี: เปลี่ยนความสว่าง ความคมชัด สี
3. เทคนิคอื่น ๆ: ผสมข้อมูล (Mix-up) ตัดชิ้นส่วนออกจากภาพ (Cutout)

การเสริมข้อมูลจะทำให้แบบจำลองมีความแม่นยำมากขึ้นได้ ทำงานได้ดีในสถานะที่ข้อมูลมีการเปลี่ยนแปลง และลดความซับซ้อนในการทำงานของแบบจำลอง ซึ่งการเสริมข้อมูลยังสามารถนำไปประยุกต์ใช้กับรูปแบบอื่น ๆ ได้ เช่น แบบจำลองทางภาษา และอนุกรมเวลาได้ด้วยเช่นกัน

การเสริมข้อมูลเป็นเทคนิคที่สำคัญมากในการพัฒนาแบบจำลองการเรียนรู้เชิงลึกโดยเฉพาะอย่างยิ่งเมื่อมีข้อมูลฝึกฝนจำนวนจำกัด การเลือกใช้เทคนิคการเสริมข้อมูลที่เหมาะสมจะช่วยให้ได้แบบจำลองที่มีประสิทธิภาพสูงขึ้น [12]

2.2.7 การประมวลผลรูปภาพเบื้องต้น (Image Pre-processing)

การประมวลผลรูปภาพเบื้องต้น คือกระบวนการปรับแต่งข้อมูลรูปภาพดิบให้เป็นรูปแบบที่ใช้งานได้และมีความหมาย ช่วยให้สามารถกำจัดความผิดพลาดที่ไม่ต้องการและปรับปรุงคุณภาพเฉพาะที่จำเป็นสำหรับงานวิทัศน์คอมพิวเตอร์ (Computer Vision) การประมวลผลรูปภาพเบื้องต้นเป็นขั้นตอนแรกที่สำคัญก่อนที่จะป้อนข้อมูลเหล่านั้นเข้าสู่แบบจำลองการเรียนรู้ของเครื่อง เทคนิคต่าง ๆ ที่ใช้ในการประมวลผลรูปภาพเบื้องต้นได้แก่

1. การปรับขนาด (Resizing): การปรับขนาดรูปภาพให้มีขนาดเท่ากันเป็นสิ่งสำคัญที่ช่วยให้อัลกอริทึมการเรียนรู้ของเครื่องทำงานได้อย่างถูกต้อง

2. การแปลงเป็นรูปภาพขาวดำ (Grayscale): การแปลงรูปภาพสีเป็นภาพขาวดำสามารถช่วยลดความซับซ้อนของข้อมูลรูปภาพ และลดการใช้ทรัพยากรในการคำนวณ
3. การลดสัญญาณรบกวน (Noise Reduction): เทคนิคการปรับให้เรียบ เนียน เบลอ และกรองรูปภาพ สามารถนำมาใช้เพื่อกำจัดสัญญาณรบกวนที่ไม่ต้องการออกจากรูปภาพ
4. การทำให้เป็นมาตรฐาน (Normalization): การทำให้เป็นมาตรฐาน จะปรับค่าความเข้มของพิกเซลไปยังช่วงที่ต้องการ มักจะอยู่ระหว่าง 0 ถึง 1 ซึ่งสามารถช่วยปรับปรุงประสิทธิภาพของแบบจำลองการเรียนรู้ของเครื่อง
5. การแบ่งเป็นสองระดับสี (Binarization): การแบ่งเป็นสองระดับสี จะแปลงรูปภาพขาวดำเป็นสีดำและสีขาวโดยใช้การแบ่งระดับสี (Thresholding)
6. การปรับปรุงความคมชัด (Contrast Enhancement): ความคมชัดของรูปภาพสามารถปรับได้โดยใช้การปรับสมดุลฮิสโตแกรม (Histogram Equalization)

ด้วยการผสมผสานเทคนิคเหล่านี้อย่างเหมาะสมจะทำให้สามารถปรับปรุงข้อมูลรูปภาพได้อย่างมีคุณภาพ และสร้างแบบจำลองจากฐานข้อมูลรูปภาพได้ดียิ่งขึ้น ทำให้การประมวลผลรูปภาพเบื้องต้นเป็นส่วนสำคัญในการสร้างและใช้งานแบบจำลองจากฐานข้อมูลรูปภาพ [20]

2.2.8 การสกัดคุณลักษณะและการเลือกคุณลักษณะ (Feature Extraction & Feature Selection)

1. Feature Extraction

เป็นกระบวนการสร้างคุณลักษณะใหม่โดยสกัดข้อมูลสำคัญจากชุดคุณลักษณะเดิม เพื่อลดจำนวนคุณลักษณะในขณะที่ยังคงเก็บข้อมูลที่สำคัญไว้ ตัวอย่างที่นิยมใช้ เช่น Principal Component Analysis (PCA) และ Linear Discriminant Analysis (LDA) ซึ่งช่วยลดความซับซ้อนของข้อมูล โดยอาจสูญเสียความหมายในบริบทดั้งเดิม แต่เพิ่มประสิทธิภาพให้กับแบบจำลองปัญญาประดิษฐ์ได้

- a. ใช้เมื่อข้อมูลมีมิติสูงมากและต้องการลดจำนวนมิติโดยไม่เสียข้อมูลสำคัญ
- b. เหมาะสำหรับการปรับปรุงประสิทธิภาพแบบจำลองโดยไม่คำนึงถึงความหมายที่ชัดเจนในข้อมูลเดิม

2. Feature Selection

เป็นกระบวนการสร้างคุณลักษณะใหม่โดยสกัดข้อมูลสำคัญจากชุดคุณลักษณะเดิม เพื่อลดจำนวนคุณลักษณะในขณะที่ยังคงเก็บข้อมูลที่สำคัญไว้ ตัวอย่างที่นิยมใช้ เช่น Principal Component Analysis (PCA) และ Linear Discriminant Analysis (LDA) ซึ่งช่วยลดความซับซ้อนของข้อมูล โดยอาจสูญเสียความหมายในบริบทดั้งเดิม แต่เพิ่มประสิทธิภาพให้กับแบบจำลองปัญญาประดิษฐ์ได้ ใช้เมื่อข้อมูลมีมิติสูงมากและต้องการลดจำนวนมิติโดยไม่เสียข้อมูลสำคัญ เหมาะสำหรับการปรับปรุงประสิทธิภาพแบบจำลองโดยไม่คำนึงถึงความหมายที่ชัดเจนในข้อมูลเดิม

เป็นกระบวนการเลือกคุณลักษณะที่สำคัญที่สุดจากชุดคุณลักษณะเดิม โดยไม่สร้างคุณลักษณะใหม่ วิธีนี้ช่วยตัดคุณลักษณะที่ไม่เกี่ยวข้องหรือซ้ำซ้อนออก ทำให้สร้างแบบจำลองง่ายขึ้นและประหยัดทรัพยากร ตัวอย่างวิธีที่ใช้ ได้แก่ Filter Methods, Wrapper Methods, และ Embedded

- a. ใช้เมื่อต้องการรักษาความหมายเดิมของข้อมูล [29]

2.3 ภาษาคอมพิวเตอร์และเทคโนโลยี

2.3.1 ภาษาคอมพิวเตอร์ที่เลือกใช้

1. TypeScript

ภาษาสคริปต์ทางคอมพิวเตอร์ที่พัฒนาต่อจาก JavaScript โดยบริษัท Microsoft สามารถนำมาสร้างได้ทั้งเว็บไซต์และแอปพลิเคชัน รวมถึงนำไปใช้ประโยชน์ได้ในฝั่งเซิร์ฟเวอร์ เป็นภาษาที่ใช้กันอย่างแพร่หลายเนื่องจาก TypeScript ได้นำข้อเสียของ JavaScript มาแก้ไขปรับปรุงให้ดีขึ้น ตัวอย่างเช่น การดักจับข้อผิดพลาดของการพัฒนาโปรแกรม โดยกำกับการใส่ชนิดของตัวแปร ทำให้การพัฒนาโปรแกรมมีข้อบกพร่องน้อยลงและมีความแม่นยำมากขึ้น [4]

2. Bun

Bun คือรันไทม์ JavaScript ที่สร้างขึ้นใหม่จากศูนย์ เพื่อรองรับ JavaScript สมัยใหม่ โดยมีเป้าหมายการออกแบบหลักคือ ความเร็ว ความเข้าใจง่ายในการพัฒนาระบบ และตัวจัดการแพ็คเกจกับเครื่องมือที่ครบครัน โดยได้รับการออกแบบมาเพื่อใช้แทน JavaScript เดิม และมีเป้าหมายคือการรัน JavaScript ฟังก์ชันเวิร์กส่วนใหญ่ของโลก รวมถึงจัดเตรียมเครื่องมือเพื่อปรับปรุงประสิทธิภาพ ลดความซับซ้อน และเพิ่มผลผลิตของนักพัฒนา

3. Python

ภาษาโปรแกรมที่มีความนิยมสูง ใช้สำหรับการพัฒนาแพลตฟอร์มหลายประเภท ทั้งแพลตฟอร์ม การวิเคราะห์ข้อมูล วิทยาศาสตร์ข้อมูล ปัญญาประดิษฐ์ และอื่น โดยนำมาใช้ในการสร้างระบบประมวลผลรูปภาพเบื้องต้น ระบบการเสริมข้อมูลในการฝึกฝนแบบจำลอง และระบบฝึกฝนแบบจำลองจากฐานข้อมูลรูปภาพ [6]

4. SQL

Structured Query Language (SQL) เป็นภาษาสคริปต์ทางคอมพิวเตอร์ใช้ในการเก็บและจัดการข้อมูลในรูปแบบตาราง (Relational Database) โดยใช้แถวและคอลัมน์บอกตำแหน่งของข้อมูลและประเภทของข้อมูล [25]

2.3.2 เทคโนโลยีที่เลือกใช้

1. Next.js

Next.js เป็นเฟรมเวิร์กที่สร้างบน React.js ภายใต้ภาษา JavaScript ถูกใช้สำหรับการพัฒนาเว็บแอปพลิเคชัน มีคุณสมบัติที่โดดเด่นคือการเรนเดอร์บนเซิร์ฟเวอร์ (Server-Side Rendering, SSR) และระบบอำนวยความสะดวกแก่ผู้พัฒนา ได้แก่ การจัดการเส้นทางอัตโนมัติ (Dynamic Routing), ช่วยแยกโค้ด (Code Spitting), การโหลดเฉพาะส่วนที่จำเป็นสำหรับการใช้งาน (Dynamic Import), และมีระบบการจัดการพื้นฐานที่ดีช่วยให้เว็บแอปพลิเคชันมีประสิทธิภาพสูง [1]

2. ElysiaJS

ElysiaJS เป็นเฟรมเวิร์กฟังก์ชันเวิร์กที่สร้างบน Bun ที่ถูกสร้างขึ้นเมื่อเดือนพฤษภาคม ปี 2567 โดยมีพื้นฐานการทำงานจากภาษา JavaScript มีคุณสมบัติที่โดดเด่นคือการทำงานที่รวดเร็ว ใช้ทรัพยากรคอมพิวเตอร์น้อยกว่า JavaScript ทั่วไปและโครงสร้างที่เข้าใจง่าย โค้ดไม่ซับซ้อนเป็นมิตรต่อผู้พัฒนา [2]

3. PostgreSQL

PostgreSQL เป็นฐานข้อมูลเชิงสัมพันธ์ (Relational Database, RDBS) มีข้อดีเหนือกว่าฐานข้อมูลอื่น ๆ คือ ประสิทธิภาพของฐานข้อมูลเร็ว มีชุมชนที่แข็งแกร่ง และการรองรับการทำงานได้หลากหลายรูปแบบ ไม่ว่าจะเป็นชุดอักขระสากล ข้อมูลเชิงไม่สัมพันธ์ การเข้ารหัสสตริงแบบหลายไบต์ [3]

4. TensorFlow

ไลบรารีการเรียนรู้ของเครื่อง (Machine Learning) ที่พัฒนาโดย Google สำหรับการสร้างและฝึกฝนแบบจำลองการเรียนรู้เชิงลึก (Deep Learning) และการคำนวณเชิงตัวเลขแบบทั่วไป

นำมาใช้ในการสร้าง ระบบการประมวลผลรูปภาพเบื้องต้น ระบบเสริมชุดข้อมูลในการฝึกฝนแบบจำลอง และระบบฝึกฝนแบบจำลองจากฐานข้อมูลรูปภาพ [7]

5. Keras

ไลบรารีสำหรับการสร้างและฝึกฝนแบบจำลองการเรียนรู้เชิงลึก (Deep Learning) ซึ่งทำหน้าที่เป็น API (Application Programming Interfaces) ระดับสูง (high-level API) ที่ทำงานร่วมกับ TensorFlow, Theano, หรือ Microsoft Cognitive Toolkit (CNTK) เพื่อให้การพัฒนาแบบจำลองการเรียนรู้เชิงลึกเป็นเรื่องง่ายและสะดวก

นำมาใช้ในการสร้างระบบฝึกฝนแบบจำลองจากฐานข้อมูลรูปภาพ และระบบการวัดประสิทธิภาพของแบบจำลองจากฐานข้อมูลรูปภาพ [8]

6. OpenCV

OpenCV (Open Source Computer Vision Library) เป็นไลบรารีโอเพ่นซอร์สที่ใช้ในงานประมวลผลภาพและวิดีโอ รวมถึงงานด้านการมองเห็นด้วยคอมพิวเตอร์ เช่น การตรวจจับใบหน้า, การติดตามวัตถุ, และการวิเคราะห์ภาพ OpenCV มีเครื่องมือสำหรับงาน Machine Learning และ Deep Learning ทำให้เป็นที่นิยมในด้าน Computer Vision และ AI โดยเฉพาะในงานที่ต้องการความเร็วและประสิทธิภาพสูง [34]

7. Apache Spark

Apache Spark เป็นเครื่องมือในการจัดการตารางข้อมูล (Dataframe) สำหรับข้อมูลขนาดใหญ่ (Big Data) มีจุดเด่นที่เหนือกว่า Hadoop คือสามารถทำงานบนหน่วยความจำหรือดิสก์ได้ ทำให้การทำงานมีประสิทธิภาพสูง และวิเคราะห์ข้อมูลได้แบบทันที [10]

8. RabbitMQ

RabbitMQ เป็นเครื่องมือในการจัดการคิวของข้อมูลเพื่อป้องกันไม่ให้เซิร์ฟเวอร์ใช้งานหนักเกินไป โดยมีจุดเด่นที่สามารถรับข้อมูลได้มาก และการตั้งค่ามีความยืดหยุ่นสูง ปรับเปลี่ยนให้รองรับใช้งานได้หลากหลายรูปแบบ [11]

2.4 ผลิตภัณฑ์ที่เกี่ยวข้อง

2.4.1 เว็บไซต์ Roboflow

Roboflow เป็นเว็บแอปพลิเคชันที่ช่วยในการจัดการและประมวลผลข้อมูลทางรูปภาพสำหรับการพัฒนาแบบจำลองปัญญาประดิษฐ์จากฐานข้อมูลรูปภาพโดยเฉพาะสำหรับงานที่เกี่ยวข้องกับการรู้จำวัตถุ การจำแนกรูปภาพ และการแบ่งส่วนรูปภาพ เป็นต้น [5] โดยเว็บไซต์ Roboflow มีคุณสมบัติหลักในการทำงานได้แก่

1. จัดการข้อมูลรูปภาพ: ช่วยในการอัปโหลด แปลง และปรับปรุงข้อมูลรูปที่ใช้สำหรับการฝึกฝนแบบจำลอง
2. การเสริมข้อมูลในการฝึกฝนแบบจำลอง: ช่วยเพิ่มจำนวนและความหลากหลายของข้อมูลรูปภาพด้วยการประยุกต์เทคนิคต่าง ๆ เช่น การหมุน การปรับขนาด และการครอบตัดรูปภาพ เพื่อให้แบบจำลองเรียนรู้ได้หลากหลายยิ่งขึ้น
3. การติดป้ายหรือจำแนกวัตถุในรูปภาพ: รองรับการอัปโหลดและปรับแต่งข้อมูลการติดป้ายหรือจำแนกวัตถุในรูปภาพหลากหลายรูปแบบ เช่น COCO, YOLO, และ Pascal VOC
4. การสร้างชุดข้อมูล: ช่วยสร้างและจัดการชุดข้อมูลที่ใช้สำหรับการฝึกฝนและทดสอบแบบจำลองจากฐานข้อมูลรูปภาพ
5. รองรับการนำข้อมูลออกจากระบบ: สามารถนำข้อมูลออกจากระบบในหลายรูปแบบ เพื่อใช้ในการฝึกฝนแบบจำลองกับไลบรารีและแพ็คเกจที่หลากหลาย เช่น TensorFlow, PyTorch, และ YOLO

ฟังก์ชันการประมวลผลรูปภาพเบื้องต้น ที่สามารถใช้งานได้บนเว็บไซต์ Roboflow แสดงได้ดังรูปที่ 2.1 ซึ่งประกอบไปด้วย You can place the figure and refer to it as รูปที่ 2.1. The figure and table numbering will be run and updated automatically when you add/remove tables/figures from the document.

2.4.2 อัลกอริทึม II

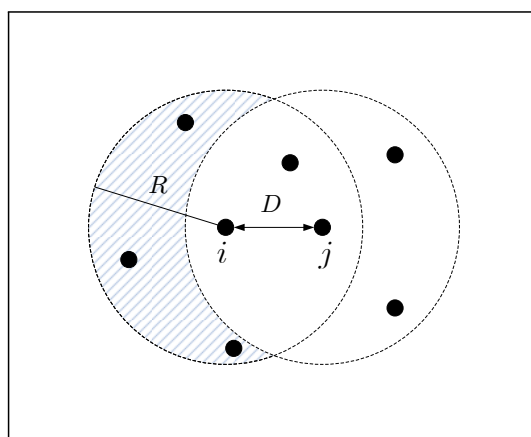
Add more subsections as you want.

2.4.2.1 ขั้นตอนที่ 1

2.4.2.2 ขั้นตอนที่ 2

Latex Format นี้รองรับหัวข้อย่อยถึงแค่ระดับ 4 นี้เท่านั้น ไม่แนะนำให้แบ่งหัวข้อย่อยไปมากกว่านี้ เช่น 2.2.2.2.1 , 2.2.2.2.2

2.5 เครื่องมือที่ใช้ในการพัฒนา



รูปที่ 2.1 The network model

บทที่ 3 วิธีการดำเนินงาน

หัวข้อต่าง ๆ ในแต่ละบทเป็นเพียงตัวอย่างเท่านั้น หัวข้อที่จะใส่ในแต่ละบทขึ้นอยู่กับโปรเจคของนักศึกษาและอาจารย์ที่ปรึกษา

ตัวอย่างการใส่อ้างอิงที่มา -> [?] ถ้าต้องการใส่แหล่งอ้างอิงมากกว่า 1 ให้ทำดังนี้ -> [?, ?] Explain the design (how you plan to implement your work) of your project. Adjust the section titles below to suit the types of your work. Detailed physical design like circuits and source codes should be placed in the appendix.

3.1 ข้อกำหนดและความต้องการของระบบ

3.2 สถาปัตยกรรมระบบ

ตารางที่ 3.1 test table x1

SYMBOL		UNIT
α	Test variable	m ²
λ	Interarrival rate	jobs/second
μ	Service rate	jobs/second

3.3 Hardware Module 1

3.3.1 Component 1

3.3.2 Logical Circuit Diagram

3.4 Hardware Module 2

3.4.1 Component 1

3.4.2 Component 2

3.5 Path Finding Algorithm

3.6 Database Design

3.7 UML Design

3.8 GUI Design

3.9 การออกแบบการทดลอง

3.9.1 ตัวชี้วัดและปัจจัยที่ศึกษา

3.9.2 รูปแบบการเก็บข้อมูล

บทที่ 4 ผลการดำเนินงาน

หัวข้อต่าง ๆ ในแต่ละบทเป็นเพียงตัวอย่างเท่านั้น หัวข้อที่จะใส่ในแต่ละบทขึ้นอยู่กับโปรเจคของนักศึกษาและอาจารย์ที่ปรึกษา

ตัวอย่างการใส่อ้างอิงที่มา -> [?] ถ้าต้องการใส่แหล่งอ้างอิงมากกว่า 1 ให้ทำดังนี้ -> [?, ?]

You can title this chapter as **Preliminary Results** ผลการดำเนินงานเบื้องต้น or **Work Progress** ความก้าวหน้าโครงการ for the progress reports. Present implementation or experimental results here and discuss them. ใส่เฉพาะหัวข้อที่เกี่ยวข้องกับงานที่ทำ

- 4.1 ประสิทธิภาพการทำงานของระบบ
- 4.2 ความพึงพอใจการใช้งาน
- 4.3 การวิเคราะห์ข้อมูลและผลการทดลอง

บทที่ 5 บทสรุป

หัวข้อต่าง ๆ ในแต่ละบทเป็นเพียงตัวอย่างเท่านั้น หัวข้อที่จะใส่ในแต่ละบทขึ้นอยู่กับโครงร่างของนักศึกษาและอาจารย์ที่ปรึกษา

This chapter is optional for proposal and progress reports but is required for the final report.

5.1 สรุปผลโครงการ

สรุปว่าโครงการบรรลุตามวัตถุประสงค์ที่ตั้งไว้หรือไม่ อย่างไร

5.2 ปัญหาที่พบและการแก้ไข

State your problems and how you fixed them.

5.3 ข้อจำกัดและข้อเสนอแนะ

ข้อจำกัดของโครงการ What could be done in the future to make your projects better.

หนังสืออ้างอิง

1. K. Schwaber, n.d., "What is Scrum?," Scrum.org, Retrieved September 17, 2024, from <https://www.scrum.org/resources/what-scrum-module>.
2. S. Jain, 2024, "Convolutional Neural Network (CNN) in Machine Learning," <https://www.geeksforgeeks.org/convolutional-neural-network-cnn-in-machine-learning/>, [Online; accessed 17-September-2024].

ภาคผนวก ก

ชื่อภาคผนวกที่ 1

ใส่หัวข้อตามความเหมาะสม

This is where you put hardware circuit diagrams, detailed experimental data in tables or source codes, etc..

รูปที่ ก.1 This is the figure x11 ทดสอบ จาก <https://www.google.com>

This appendix describes two static allocation methods for fGn (or fBm) traffic. Here, λ and C are respectively the traffic arrival rate and the service rate per dimensionless time step. Their unit are converted to a physical time unit by multiplying the step size Δ . For a fBm self-similar traffic source, Norros [?] provides its EB as

$$C = \lambda + (\kappa(H)\sqrt{-2\ln\epsilon})^{1/H} a^{1/(2H)} x^{-(1-H)/H} \lambda^{1/(2H)} \quad (n.1)$$

where $\kappa(H) = H^H(1-H)^{(1-H)}$. Simplicity in the calculation is the attractive feature of (n.1).

The MVA technique developed in [?] so far provides the most accurate estimation of the loss probability compared to previous bandwidth allocation techniques according to simulation results. Consider a discrete-time queueing system with constant service rate C and input process λ_n with $\mathbb{E}\{\lambda_n\} = \lambda$ and $\text{Var}\{\lambda_n\} = \sigma^2$. Define $X_n \equiv \sum_{k=1}^n \lambda_k - Cn$. The loss probability due to the MVA approach is given by

$$\varepsilon \approx \alpha e^{-m_x/2} \quad (n.2)$$

where

$$m_x = \min_{n \geq 0} \frac{((C - \lambda)n + B)^2}{\text{Var}\{X_n\}} = \frac{((C - \lambda)n^* + B)^2}{\text{Var}\{X_{n^*}\}} \quad (n.3)$$

and

$$\alpha = \frac{1}{\lambda\sqrt{2\pi\sigma^2}} \exp\left(\frac{(C - \lambda)^2}{2\sigma^2}\right) \int_C^\infty (r - C) \exp\left(\frac{(r - \lambda)^2}{2\sigma^2}\right) dr \quad (n.4)$$

For a given ε , we numerically solve for C that satisfies (n.2). Any search algorithm can be used to do the task. Here, the bisection method is used.

Next, we show how $\text{Var}\{X_n\}$ can be determined. Let $C_\lambda(l)$ be the autocovariance function of λ_n . The MVA technique basically approximates the input process λ_n with a Gaussian process, which allows $\text{Var}\{X_n\}$ to be represented by the autocovariance function. In particular, the variance of X_n can be expressed in terms of $C_\lambda(l)$ as

$$\text{Var}\{X_n\} = nC_\lambda(0) + 2 \sum_{l=1}^{n-1} (n-l)C_\lambda(l) \quad (n.5)$$

Therefore, $C_\lambda(l)$ must be known in the MVA technique, either by assuming specific traffic models or by off-line analysis in case of traces. In most practical situations, $C_\lambda(l)$ will not be known in advance, and an on-line measurement algorithm developed in [?] is required to jointly determine both n^* and m_x . For fGn traffic, $\text{Var}\{X_n\}$ is equal to $\sigma^2 n^{2H}$, where $\sigma^2 = \text{Var}\{\lambda_n\}$, and we can find the n^* that minimizes (n.3) directly. Although λ can be easily measured, it is not the case for σ^2 and H . Consequently, the MVA technique suffers from the need of prior knowledge traffic parameters.

ภาคผนวก ข

ชื่อภาคผนวกที่ 2

ใส่หัวข้อตามความเหมาะสม

รูปที่ ข.1 This is the figure x11 ทดสอบ จาก <https://www.google.com>

Next, we show how $\text{Var}\{X_n\}$ can be determined. Let $C_\lambda(l)$ be the autocovariance function of λ_n . The MVA technique basically approximates the input process λ_n with a Gaussian process, which allows $\text{Var}\{X_n\}$ to be represented by the autocovariance function. In particular, the variance of X_n can be expressed in terms of $C_\lambda(l)$ as

$$\text{Var}\{X_n\} = nC_\lambda(0) + 2 \sum_{l=1}^{n-1} (n-l)C_\lambda(l) \quad (\text{ข.1})$$

Add more topic as you need

Therefore, $C_\lambda(l)$ must be known in the MVA technique, either by assuming specific traffic models or by off-line analysis in case of traces. In most practical situations, $C_\lambda(l)$ will not be known in advance, and an on-line measurement algorithm developed in [?] is required to jointly determine both n^* and m_x . For fGn traffic, $\text{Var}\{X_n\}$ is equal to $\sigma^2 n^{2H}$, where $\sigma^2 = \text{Var}\{\lambda_n\}$, and we can find the n^* that minimizes (ข.3) directly. Although λ can be easily measured, it is not the case for σ^2 and H . Consequently, the MVA technique suffers from the need of prior knowledge traffic parameters.