



## รายงานวิจัยฉบับสมบูรณ์

การศึกษาความเป็นไปได้ด้านการออกแบบและพัฒนาเทคโนโลยีคู่แฝดดิจิทัลบน  
พื้นฐานของเทคโนโลยี PLCnext (ระยะที่ 2)

Feasibility Study of Design and Development of Digital Twin  
Technology based on PLCnext Technology (Phase 2)

นายประจักษ์ จิตเงินมะดัน หัวหน้าโครงการวิจัย

โครงการวิจัยประเภทเงินรายได้

คณะวิทยาการสารสนเทศ

ประจำปีงบประมาณ พ.ศ. 2567

มหาวิทยาลัยบูรพา

สัญญาเลขที่ 013/2567

รายงานวิจัยฉบับสมบูรณ์

การศึกษาความเป็นไปได้ด้านการออกแบบและพัฒนาเทคโนโลยีคู่แฝดดิจิทัลบนพื้นฐาน  
ของเทคโนโลยี PLCnext (ระยะที่ 2)

Feasibility Study of Design and Development of Digital Twin Technology  
based on PLCnext Technology (Phase 2)

นายประจักษ์ จิตเงินมะดัน หัวหน้าโครงการวิจัย

คณะวิทยาการสารสนเทศ

มิถุนายน 2568

## กิตติกรรมประกาศ

งานวิจัยนี้ได้รับทุนสนับสนุนการวิจัยจากเงินรายได้ คณะวิทยาการสารสนเทศ ประจำปีงบประมาณ พ.ศ. 2567 เลขที่สัญญา 013/2567

คณะผู้วิจัยขอขอบคุณคณะวิทยาการสารสนเทศ และห้องปฏิบัติการวิจัยสื่อดิจิทัลและปฏิสัมพันธ์ (Digital Media and Interaction Laboratory) มหาวิทยาลัยบูรพา ที่เอื้อเฟื้องบประมาณในการดำเนินการ และสถานที่ในการวางแผน ปฏิบัติ และทดสอบชิ้นงานให้มีความสมบูรณ์ รวมถึงการให้บริการด้านสาธารณูปโภคต่าง ๆ ด้วย

และในท้ายที่สุด คณะผู้วิจัยขอขอบคุณผู้ร่วมทดสอบการใช้งานได้และประสิทธิภาพของระบบการฝึกปฏิบัติเสมือนจริงสำหรับงานอัตโนมัติในพื้นที่ EEC กรณีศึกษาการฝึกปฏิบัติการประกอบรถยนต์ไฟฟ้า และทีมงานสนับสนุนทุกคนที่ทุ่มเทแรงกายแรงใจในการดำเนินการจนงานวิจัยนี้สำเร็จลุล่วงไปได้ด้วยดี

มิถุนายน 2568

## บทคัดย่อ

งานวิจัยนี้ศึกษาความเป็นไปได้ในการออกแบบและพัฒนาเทคโนโลยีคู่แฝดดิจิทัล (Digital Twin) บนพื้นฐานของเทคโนโลยี PLCnext โดยมุ่งเน้นการประยุกต์ใช้ในอุตสาหกรรมการผลิตอัตโนมัติ การวิจัยแบ่งเป็นสองระยะ โดยระยะที่ 1 เน้นการศึกษาและทดสอบโปรโตคอลการสื่อสารระหว่างอุปกรณ์จริงกับแบบจำลองดิจิทัล และระยะที่ 2 พัฒนาความสามารถในการวิเคราะห์และทำนายด้วยโครงข่ายประสาทเทียม ผลการศึกษาพบว่าโปรโตคอล MQTT มีประสิทธิภาพสูงในการส่งข้อมูลจากอุปกรณ์ PLCnext ไปยังแพลตฟอร์ม IoT ขณะที่ WebSocket เหมาะสมกับการดึงข้อมูลแบบเรียลไทม์จากแพลตฟอร์มไปยัง Unity การพัฒนาแบบจำลองโครงข่ายประสาทเทียมแบบไฮบริด CNN-LSTM สามารถทำนายพารามิเตอร์การผลิตได้แม่นยำกว่าแบบจำลองเดี่ยว โดยมีค่า mAP50 ที่ 63.8% เปรียบเทียบกับ LSTM ที่ 57.2% และ CNN ที่ 51.9% ปัจจัยที่ส่งผลต่ออุณหภูมิกระบวนการมากที่สุดคือความเร็วการหมุน (55%) แรงบิด (22%) และการสีกหรือของเครื่องมือ (15%) การศึกษานี้แสดงให้เห็นว่าเทคโนโลยีคู่แฝดดิจิทัลเป็นเครื่องมือที่มีประสิทธิภาพในการปรับปรุงกระบวนการผลิตและการบำรุงรักษาเชิงพยากรณ์ในอุตสาหกรรม 4.0

## Abstract

This research investigates the feasibility of designing and developing Digital Twin technology based on PLCnext technology, focusing on applications in automated manufacturing. The research was divided into two phases, with Phase 1 emphasizing the study and testing of communication protocols between physical devices and digital models, and Phase 2 developing analytical and predictive capabilities using neural networks. The study found that the MQTT protocol demonstrates high efficiency in transmitting data from PLCnext devices to IoT platforms, while WebSocket is suitable for real-time data retrieval from platforms to Unity. The development of a hybrid CNN-LSTM neural network model provided more accurate manufacturing parameter predictions than single models, achieving mAP50 score of 63.8% compared to 57.2% for LSTM and 51.9% for CNN models. The factors most influencing process temperature were rotational speed (55%), torque (22%), and tool wear (15%). This study demonstrates that Digital Twin technology is an effective tool for improving manufacturing processes and predictive maintenance in Industry 4.0 environments.

## สารบัญ

1	บทนำ .....	1
1.1	ที่มาและความสำคัญ .....	1
1.2	วัตถุประสงค์.....	3
1.3	ฮาร์ดแวร์และซอฟต์แวร์ .....	3
1.3.1	อุปกรณ์ด้านฮาร์ดแวร์ (Hardware) .....	3
1.3.2	ซอฟต์แวร์ที่ใช้ในการพัฒนา (Software).....	3
1.4	ขอบเขตของการวิจัย .....	3
1.5	ขั้นตอนการดำเนินงาน .....	3
1.5.1	วางแผนการดำเนินงาน.....	3
1.5.2	ศึกษาความเป็นไปได้ .....	4
1.5.3	วิเคราะห์และออกแบบ .....	4
1.5.4	พัฒนาระบบ.....	4
1.5.5	ทดสอบแบบจำลองและปรับปรุงแก้ไข.....	4
1.6	แผนการดำเนินการ .....	4
1.7	ประโยชน์ที่คาดว่าจะได้รับ .....	5
2	ทฤษฎีและโครงการที่เกี่ยวข้อง.....	6
2.1	ทฤษฎีที่เกี่ยวข้อง.....	6
2.1.1	คู่มือดิจิทัลในอุตสาหกรรมการผลิต (Digital Twin in Manufacturing).....	6
2.1.2	โครงข่ายประสาทเทียมสำหรับการทำนายอนุกรมเวลาในการผลิต (Neural Networks for Time Series Prediction in Manufacturing).....	6
2.1.3	สถาปัตยกรรมไฮบริด CNN-LSTM (Hybrid CNN-LSTM Architecture).....	7
2.1.4	โพรโตคอลการสื่อสารสำหรับระบบคู่มือดิจิทัล (Communication Protocols for Digital Twin Systems).....	8
2.2	งานที่เกี่ยวข้อง (Related Works).....	8

2.2.1	ระบบคู่แฝดดิจิทัลสำหรับการผลิตอัจฉริยะ (Digital Twin Systems for Smart Manufacturing).....	8
2.2.2	การประยุกต์ใช้การเรียนรู้เชิงลึกกับระบบคู่แฝดดิจิทัล (Application of Deep Learning to Digital Twin Systems).....	9
2.2.3	การเพิ่มประสิทธิภาพพารามิเตอร์การผลิตด้วยคู่แฝดดิจิทัล (Optimization of Manufacturing Parameters Using Digital Twin).....	9
2.2.4	การบำรุงรักษาเชิงพยากรณ์โดยใช้คู่แฝดดิจิทัล (Predictive Maintenance Using Digital Twin).....	10
3	วิธีการดำเนินงานโครงการ.....	11
3.1	สถาปัตยกรรมระบบ.....	11
3.1.1	ชั้นอุปกรณ์ทางกายภาพ (Physical Device Layer).....	11
3.1.2	ชั้นการสื่อสารและการจัดการข้อมูล (Communication and Data Management Layer).....	12
3.1.3	ชั้นการวิเคราะห์และการทำนาย (Analytics and Prediction Layer).....	12
3.1.4	ชั้นการแสดงผลและปฏิสัมพันธ์ (Visualization and Interaction Layer).....	12
3.2	การไหลของข้อมูล.....	13
3.2.1	การเก็บรวบรวมข้อมูล (Data Collection).....	13
3.2.2	การส่งข้อมูลและการประมวลผลเบื้องต้น (Data Transmission and Pre-processing).....	13
3.2.3	การวิเคราะห์และการทำนายด้วยแบบจำลองโครงข่ายประสาทเทียม (Analysis and Prediction).....	13
3.2.4	การแสดงผลและปฏิสัมพันธ์กับผู้ใช้ (Visualization and User Interaction).....	13
3.2.5	การอัปเดตและปรับปรุงแบบจำลอง (Model Update and Improvement).....	14
3.3	ชุดข้อมูล.....	14
3.3.1	ลักษณะของชุดข้อมูล.....	14
3.3.2	การวิเคราะห์ข้อมูลเบื้องต้น.....	15
3.3.3	การเตรียมข้อมูลสำหรับแบบจำลอง.....	16

3.3.4	ความเกี่ยวข้องกับการประยุกต์ใช้ในโลจิสติกส์.....	16
3.4	สถาปัตยกรรมแบบจำลอง.....	17
3.4.1	แบบจำลอง LSTM.....	17
3.4.2	แบบจำลอง CNN.....	18
3.4.3	แบบจำลองไฮบริด CNN-LSTM.....	18
3.5	กระบวนการฝึกสอนและการปรับปรุงประสิทธิภาพ.....	19
3.5.1	การเตรียมข้อมูล.....	19
3.5.2	การฝึกสอนแบบจำลอง.....	19
3.5.3	การปรับปรุงประสิทธิภาพ.....	19
3.6	กระบวนการสำรวจพารามิเตอร์.....	20
3.6.1	การฝึกสอนแบบจำลองหลายรูปแบบ.....	20
3.6.2	การวิเคราะห์ความสำคัญของพารามิเตอร์.....	20
3.6.3	การแมปพารามิเตอร์เข้ากับการแสดงผลที่เหมาะสม.....	21
3.6.4	การพัฒนาส่วนติดต่อผู้ใช้แบบปฏิสัมพันธ์.....	21
3.6.5	กระบวนการสำรวจและวิเคราะห์พารามิเตอร์.....	24
4	ผลการดำเนินการ.....	25
4.1	ผลการทดสอบประสิทธิภาพของแบบจำลอง.....	25
4.2	การทดลองทำนายพารามิเตอร์การผลิต.....	28
4.3	การวิเคราะห์ความสำคัญของคุณลักษณะ.....	29
4.4	ข้อจำกัดด้านประสิทธิภาพและความท้าทาย.....	31
4.5	หน้าตาการใช้งานแอปพลิเคชันต้นแบบ.....	31
5	สรุปผล.....	34
5.1	สรุปผลการดำเนินงาน.....	34
5.2	ปัญหา อุปสรรค และข้อจำกัด.....	34
5.2.1	ข้อจำกัดด้านข้อมูล.....	34
5.2.2	ข้อจำกัดทางเทคนิค.....	34

---

5.2.3	ข้อจำกัดในการประยุกต์ใช้ .....	35
5.3	ข้อเสนอแนะ .....	35
5.3.1	ข้อเสนอแนะสำหรับการนำไปใช้ .....	35
5.3.2	ข้อเสนอแนะเชิงกลยุทธ์ .....	35
5.4	แนวทางในการพัฒนาในอนาคต .....	35
5.4.1	การปรับปรุงแบบจำลอง .....	36
5.4.2	การพัฒนาระบบคู่มือดิจิทัล .....	36
5.4.3	การปรับปรุงการใช้งาน .....	36
6	บรรณานุกรม .....	37
7	ภาคผนวก .....	40
7.1	การรับรองจริยธรรมวิจัย .....	40
7.2	ผลงานที่ได้รับการตอบรับให้ตีพิมพ์และเผยแพร่ .....	41
7.3	ขั้นตอนการเชื่อมต่อ PLCnext กับ PLCnext engineer .....	53
7.4	การสร้างและใช้งานสคริปต์ใน Unity .....	57
7.5	อธิบายโค้ดที่ใช้เบื้องต้น .....	60
7.6	ขั้นตอนการใช้แอปพลิเคชันคู่มือดิจิทัล .....	62
8	ประวัตินักวิจัยและคณะ .....	66
8.1	หัวหน้าโครงการวิจัย .....	66

---

สารบัญตาราง

ตารางที่ 1 การเปรียบเทียบประสิทธิภาพของแบบจำลอง .....	25
ตารางที่ 2 การทำนายอุณหภูมิกระบวนการในการกำหนดค่าพารามิเตอร์ต่าง ๆ (หน่วย: เคลวิน).....	28

## สารบัญภาพ

ภาพที่ 1 สถาปัตยกรรมระบบคู่มือดิจิทัล .....	11
ภาพที่ 2 แบบจำลอง 3 มิติของเครื่องจักรที่ตอบสนองต่อการเปลี่ยนแปลงพารามิเตอร์แบบเรียลไทม์ (1) ....	22
ภาพที่ 3 แบบจำลอง 3 มิติของเครื่องจักรที่ตอบสนองต่อการเปลี่ยนแปลงพารามิเตอร์แบบเรียลไทม์ (2) ....	22
ภาพที่ 4 แผงควบคุมพารามิเตอร์.....	23
ภาพที่ 5 ส่วนแสดงผลการทำนาย.....	23
ภาพที่ 6 แสดงการเปรียบเทียบการทำนายอุณหภูมิของแบบจำลองทั้งสาม โดยเน้นความแตกต่างในการตอบสนองต่อการปรับพารามิเตอร์.....	28
ภาพที่ 7 ผลลัพธ์ของ Feature Importance Analysis .....	30
ภาพที่ 8 แอปพลิเคชันต้นแบบ.....	32
ภาพที่ 9 หน้าต่าง Network & Internet .....	53
ภาพที่ 10 กด Properties ของ Ethernet .....	54
ภาพที่ 11 การเลือก IPv4 .....	54
ภาพที่ 12 การกำหนดให้อยู่วง LAN เดียวกัน .....	55
ภาพที่ 13 หน้าต่าง UI ของ PLCnext engineer .....	56
ภาพที่ 14 เลือกพอร์ตที่เชื่อมต่อกับ PLC.....	56
ภาพที่ 15 การกด Scan network เพื่อ ค้นหา PLC.....	56
ภาพที่ 16 การสร้าง Script ใน Unity.....	57
ภาพที่ 17 การสร้าง Script ใน Unity (ต่อ).....	58
ภาพที่ 18 การตั้งชื่อ Script ใน Unity.....	58
ภาพที่ 19 การเขียนโค้ดลง Script.....	59
ภาพที่ 20 การสร้าง Game Object.....	60
ภาพที่ 21 การใส่ Script ลงไปใน Game Object .....	60
ภาพที่ 22 โค้ดการนำเข้า Library.....	61
ภาพที่ 23 โค้ดการประกาศตัวแปร .....	61
ภาพที่ 24 โค้ดกำหนดให้โค้ดรันเป็น Loop เป็นรอบวินาทีที่กำหนด.....	61
ภาพที่ 25 ส่งค่าขอตั้งข้อมูลและการจับเวลา.....	62
ภาพที่ 26 การเลือกโมเดลสำหรับทำนาย.....	62
ภาพที่ 27 การเลือกชุดข้อมูล .....	63
ภาพที่ 28 การเลือกตัวแปรที่ต้องการทำนาย .....	63
ภาพที่ 29 ภาพใน CMD เมื่อรอโมเดลเทรน.....	64
ภาพที่ 30 หน้าต่างแสดงผลแผงการปรับพารามิเตอร์ .....	64

---

ภาพที่ 31 เลือกโมเดลที่สอดคล้องกับชุดข้อมูลที่มี .....	65
ภาพที่ 32 ปรับพารามิเตอร์และทำการเปลี่ยนแปลงของโมเดลแบบเรียลไทม์ .....	65

## 1 บทนำ

### 1.1 ที่มาและความสำคัญ

การผลิตในภาคอุตสาหกรรมสมัยใหม่ได้พัฒนาก้าวหน้าอย่างมีนัยสำคัญ โดยมีการบูรณาการเทคโนโลยีการผลิตร่วมกับระบบควบคุมเครื่องจักรและเทคโนโลยีสารสนเทศดิจิทัล การบูรณาการนี้นำไปสู่การผลิตแบบอัตโนมัติซึ่งเป็นลักษณะเด่นของอุตสาหกรรม 4.0 ที่เทคโนโลยีดิจิทัลทำหน้าที่ควบคุมระบบการผลิตอัตโนมัติในระบบการควบคุมการผลิต เครื่องควบคุมตรรกะแบบโปรแกรมได้ (Programmable Logic Controllers: PLCs) ได้ถูกนำมาใช้อย่างแพร่หลายและได้รับการยอมรับเป็นมาตรฐานสากล เมื่อเทคโนโลยีดิจิทัลมีการพัฒนาอย่างต่อเนื่อง แนวคิดการใช้วัตถุเสมือนเพื่อจำลองการทำงานของเครื่องจักรหรือคุณลักษณะทางกายภาพได้ถูกพัฒนาขึ้น เทคโนโลยีนี้ที่รู้จักกันในชื่อเทคโนโลยีคู่แฝดดิจิทัล หรือ "ดิจิทัลทวิน" (Digital Twin) ได้มีความสำคัญเพิ่มขึ้นอย่างมากในการประยุกต์ใช้งานด้านอุตสาหกรรม (Grieves & Vickers, 2016) อย่างไรก็ตาม เทคโนโลยีคู่แฝดดิจิทัลยังคงอยู่ในช่วงเริ่มต้นของการนำมาใช้งาน จึงจำเป็นต้องมีการศึกษาวิจัยอย่างครอบคลุมเกี่ยวกับความเป็นไปได้ การประยุกต์ใช้ และผลกระทบที่อาจเกิดขึ้น งานวิจัยนี้มุ่งเน้นการประยุกต์ใช้เทคโนโลยีคู่แฝดดิจิทัลกับการผลิตแบบอัตโนมัติในสภาพแวดล้อมอุตสาหกรรม 4.0 โดยใช้เทคโนโลยี PLCnext (PLCnext Community, 2024) ซึ่งมีความสามารถที่เหนือกว่า PLC แบบดั้งเดิมและรองรับโปรโตคอลการสื่อสารสมัยใหม่ เทคโนโลยี PLCnext ช่วยให้สามารถเชื่อมต่อกับอุปกรณ์อินเทอร์เน็ตของสรรพสิ่ง (Internet of Things: IoT) และระบบการจัดเก็บและเรียกใช้ข้อมูลบนคลาวด์ได้อย่างมีประสิทธิภาพ

ในการประยุกต์ใช้คู่แฝดดิจิทัล การสื่อสารระหว่างระบบดิจิทัลและระบบกายภาพมีความสำคัญอย่างยิ่งสำหรับการโต้ตอบแบบเรียลไทม์ที่มีประสิทธิภาพ งานวิจัยของ Fuller และคณะ (2020) ได้แสดงให้เห็นว่าคู่แฝดดิจิทัลช่วยอำนวยความสะดวกในการเชื่อมต่อที่ราบรื่นระหว่างเครื่องจักรกายภาพและโมเดลเสมือน ทำให้สามารถปรับปรุงการส่งข้อมูล การอัปเดตสถานะแบบเรียลไทม์ และการควบคุมโดยใช้เทคโนโลยีต่าง ๆ เช่น IoT และปัญญาประดิษฐ์ (Artificial Intelligence: AI) นอกจากนี้ Liu และคณะ (2021) ได้ทำการสำรวจเครื่องมือสำหรับการสร้างเทคโนโลยีคู่แฝดดิจิทัล ซึ่งครอบคลุมทั้งส่วนประกอบด้านฮาร์ดแวร์และซอฟต์แวร์ ผลกระทบที่อาจเกิดขึ้น และทิศทางเทคโนโลยีในอนาคต ในแง่มุมทางเศรษฐกิจ เทคโนโลยีคู่แฝดดิจิทัลส่งผลกระทบต่อวิถีการผลิตและการจำลองข้อมูลก่อนการผลิตจริง World Economic Forum (2022) ระบุว่าเทคโนโลยีคู่แฝดดิจิทัลจะมีบทบาทสำคัญในภาคการผลิตในอนาคต โดยอุตสาหกรรมยานยนต์และการบินจะประสบการเปลี่ยนแปลงมากที่สุด นอกจากนี้ คู่แฝดดิจิทัลจะมีส่วนช่วยในการวางแผนเมืองอัจฉริยะและการดำรงชีวิตอย่างยั่งยืนผ่านการวางแผนลดการปล่อยคาร์บอน Madni และคณะ (2019) ได้อภิปรายถึงวิธีที่การพัฒนาคู่แฝดดิจิทัลจะส่งผลกระทบต่อการทำงานในอนาคต โดยเฉพาะอย่างยิ่งการแทนที่การจำลองโมเดลทางคณิตศาสตร์แบบดั้งเดิม เนื่องจากเทคโนโลยีคู่แฝดดิจิทัลสามารถเชื่อมต่อข้อมูลของเครื่องจักรกายภาพกับเครื่องจักรเสมือนได้อย่างแท้จริง สิ่งนี้ทำให้ผู้ใช้สามารถปรับแต่งเครื่องจักรเสมือนเพื่อ

จำลองสถานการณ์ต่าง ๆ ได้อย่างครอบคลุมด้วยต้นทุนที่ต่ำกว่าการจำลองทางกายภาพ ส่งผลให้สามารถลดระยะเวลาในการออกแบบผลิตภัณฑ์และการผลิตได้อย่างมีประสิทธิภาพ (IBM, 2023)

Deng และคณะ (2021) ได้ทำการทบทวนวรรณกรรมอย่างเป็นระบบเกี่ยวกับวิวัฒนาการของตัวแทนอาคารอัจฉริยะในอุตสาหกรรม AEC-FM โดยอธิบายถึงการเปลี่ยนผ่านจาก BIM (Building Information Modeling) ไปสู่คู่แฝดดิจิทัล ซึ่งแสดงให้เห็นถึงความก้าวหน้าในการจำลองและการควบคุมระบบที่ซับซ้อน Moyne และคณะ (2020) ได้นำเสนอกรอบงานคู่แฝดดิจิทัลที่ขับเคลื่อนด้วยความต้องการ โดยระบุข้อกำหนดและโอกาสในการใช้งานคู่แฝดดิจิทัลในอุตสาหกรรม ซึ่งช่วยให้ผู้พัฒนาระบบมีแนวทางที่ชัดเจนในการออกแบบและใช้งานคู่แฝดดิจิทัล นอกจากนี้ Zheng และคณะ (2021) ได้กล่าวถึงการเกิดขึ้นของคู่แฝดดิจิทัลเชิงความคิด (Cognitive Digital Twin) ซึ่งเป็นการผสมผสานระหว่างเทคโนโลยีคู่แฝดดิจิทัลกับระบบปัญญาประดิษฐ์ขั้นสูง ทำให้ระบบมีความสามารถในการเรียนรู้และตัดสินใจด้วยตนเอง ระบบการสื่อสารในคู่แฝดดิจิทัลมีความสำคัญอย่างยิ่งต่อประสิทธิภาพโดยรวมของระบบ Wukkadada และคณะ (2018) ได้ทำการเปรียบเทียบโปรโตคอล HTTP และ MQTT ในบริบทของอินเทอร์เน็ตของสรรพสิ่ง (IoT) ซึ่งเป็นส่วนสำคัญของระบบคู่แฝดดิจิทัล ผลการศึกษาพบว่าโปรโตคอล MQTT มีประสิทธิภาพสูงกว่าในสภาพแวดล้อมที่มีข้อจำกัดด้านทรัพยากรและความเร็วในการสื่อสาร งานวิจัยของ Raza และคณะ (2020) ได้นำเสนอกรอบงานคู่แฝดดิจิทัลสำหรับอุตสาหกรรม 4.0 ที่ช่วยให้เกิดการผลิตรายบุคคลไป กรอบงานนี้แสดงให้เห็นถึงวิธีการบูรณาการระบบคู่แฝดดิจิทัลเข้ากับโครงสร้างพื้นฐานการผลิตที่มีอยู่ และวิธีการใช้ประโยชน์จากเทคโนโลยีนี้เพื่อปรับปรุงประสิทธิภาพการผลิต ลดต้นทุน และเพิ่มความยืดหยุ่นของระบบการผลิต

การประยุกต์ใช้คู่แฝดดิจิทัลยังขยายไปถึงด้านความปลอดภัยและการประเมินความเสี่ยง Leonardo และคณะ (2020) ได้พัฒนาโมเดลคู่แฝดดิจิทัลสำหรับการประเมินอันตรายจากการชนในเวลาจริง ซึ่งช่วยให้สามารถคาดการณ์และป้องกันอุบัติเหตุที่อาจเกิดขึ้นในสภาพแวดล้อมการทำงานที่มีมนุษย์และหุ่นยนต์ทำงานร่วมกัน ผลการเรียนรู้จากการทดลองนี้มีประโยชน์ในการชี้แนะการเลือกโปรโตคอลการสื่อสารสำหรับการตั้งค่าคู่แฝดดิจิทัล โดยมีเป้าหมายเพื่อลดความล่าช้าและปรับปรุงการโต้ตอบแบบเรียลไทม์ ผลการค้นพบนี้เพิ่มความเข้าใจที่เพิ่มขึ้นเกี่ยวกับวิธีการปรับโปรโตคอลการสื่อสารให้เหมาะสมสำหรับการตั้งค่าคู่แฝดดิจิทัล สนับสนุนการตรวจสอบแบบเรียลไทม์ที่มีประสิทธิภาพมากขึ้น การควบคุม และการตัดสินใจ โดยเฉพาะในอุตสาหกรรม 4.0 และระบบอัตโนมัติ 4.0 ที่การสื่อสารแบบเรียลไทม์เป็นหนึ่งในกุญแจสำคัญสำหรับการใช้งานคู่แฝดดิจิทัลที่ประสบความสำเร็จ ดังนั้น การศึกษาวิจัยเกี่ยวกับการประยุกต์ใช้เทคโนโลยีคู่แฝดดิจิทัลในอุตสาหกรรม 4.0 จึงมีความสำคัญอย่างยิ่งต่อการพัฒนาอุตสาหกรรมการผลิตในอนาคต โดยเฉพาะอย่างยิ่งในประเทศที่กำลังเร่งพัฒนาขีดความสามารถด้านการผลิตขั้นสูง เช่น ในประเทศไทย ที่มีการจัดตั้งเขตพัฒนาพิเศษภาคตะวันออก (EEC) ซึ่งมีวิสัยทัศน์ในการยกระดับอุตสาหกรรมไทยสู่อุตสาหกรรม 4.0 (EEC, 2020) การพัฒนาองค์ความรู้ในด้านนี้จะช่วยให้ผู้ประกอบการและนักวิจัยสามารถออกแบบและพัฒนาระบบการผลิตที่มีประสิทธิภาพสูงขึ้น มีความยืดหยุ่น และตอบสนองต่อความต้องการของตลาดได้อย่างรวดเร็วยิ่งขึ้น

## 1.2 วัตถุประสงค์

- 1) เพื่อศึกษาและทดสอบประสิทธิภาพของแบบจำลองปัญญาประดิษฐ์ที่เหมาะสมกับการพัฒนาเทคโนโลยีคู่มือดิจิทัลในอุตสาหกรรมอัตโนมัติ
- 2) เพื่อศึกษาและพัฒนาตัวต้นแบบระบบคู่มือดิจิทัลในอุตสาหกรรมอัตโนมัติ

## 1.3 ฮาร์ดแวร์และซอฟต์แวร์

### 1.3.1 อุปกรณ์ด้านฮาร์ดแวร์ (Hardware)

- 1) PLCnext AXC F 2152 เป็น PLC ที่มีความสามารถสูงในการรวบรวมและส่งข้อมูลจากโรงงานหรือโรงงานอัจฉริยะไปยังแพลตฟอร์ม IoT
- 2) คอมพิวเตอร์ ใช้ในการจัดเก็บและแสดงผลข้อมูลการทดสอบ

### 1.3.2 ซอฟต์แวร์ที่ใช้ในการพัฒนา (Software)

- 1) PLCnext Engineer ใช้ในการตั้งค่าและโปรแกรม PLC เพื่อรวบรวมและส่งข้อมูล
- 2) Thingsboard ใช้เป็นแพลตฟอร์ม IoT ในการเก็บรวบรวมและวิเคราะห์ข้อมูลที่ได้รับจาก PLC และอุปกรณ์ IoT
- 3) Unity ใช้ในการสร้างและแสดงผลแบบจำลองดิจิทัลเสมือนจริง โดยการดึงข้อมูลจาก Thingsboard เพื่อแสดงผลแบบเรียลไทม์
- 4) MQTT Broker ใช้ในการส่งข้อมูลระหว่าง PLC และ Thingsboard
- 5) HTTP Server ใช้ในการดึงข้อมูลจาก Thingsboard ไปยัง Unity
- 6) WebSocket Server ใช้ในการดึงข้อมูลแบบเรียลไทม์จาก Thingsboard ไปยัง Unity

## 1.4 ขอบเขตของการวิจัย

- 1) เทคโนโลยีควบคุมเครื่องจักรที่ใช้งานคือ PLCnext
- 2) การศึกษาการผลิตแบบอัตโนมัติใช้แบบจำลองการผลิตอัตโนมัติ Smart Factory 4.0
- 3) แบบจำลองเครื่องจักรเสมือนจะอยู่ในรูปแบบ Virtual Reality (VR)
- 4) การควบคุมจะควบคุมผ่าน Controller ของอุปกรณ์ VR

## 1.5 ขั้นตอนการดำเนินงาน

### 1.5.1 วางแผนการดำเนินงาน

- 1) กำหนดวัตถุประสงค์ของการศึกษา
- 2) วางแผนการรวบรวมข้อมูลจาก PLC
- 3) วางแผนการทดสอบโปรโตคอลการสื่อสาร

### 1.5.2 ศึกษาความเป็นไปได้

- 1) ทำการศึกษาโปรโตคอลการสื่อสาร MQTT, HTTP และ WebSocket
- 2) ทดสอบการส่งข้อมูลระหว่าง PLC และ Thingsboard โดยใช้โปรโตคอล MQTT และ HTTP
- 3) ทดสอบการดึงข้อมูลจาก Thingsboard ไปยัง Unity โดยใช้โปรโตคอล HTTP และ WebSocket

### 1.5.3 วิเคราะห์และออกแบบ

- 1) ออกแบบโครงสร้างของระบบคู่มือดิจิทัล
- 2) กำหนดขั้นตอนการส่งข้อมูลจาก PLC ไปยัง Thingsboard
- 3) กำหนดขั้นตอนการดึงข้อมูลจาก Thingsboard ไปยัง Unity
- 4) ออกแบบการแสดงผลแบบเรียลไทม์ใน Unity

### 1.5.4 พัฒนาระบบ

- 1) ตั้งค่าและโปรแกรม PLC เพื่อรวบรวมและส่งข้อมูลผ่าน Node Red
- 2) ตั้งค่า Node Red ให้สามารถส่งข้อมูลขึ้น Thingsboard
- 3) ตั้งค่า Thingsboard เพื่อรับและจัดเก็บข้อมูลจาก PLC
- 4) สร้างโปรแกรม ใน Unity และเชื่อมต่อกับ Thingsboard

### 1.5.5 ทดสอบแบบจำลองและปรับปรุงแก้ไข

- 1) ทดสอบการส่งข้อมูลระหว่าง PLC และ Thingsboard โดยใช้โปรโตคอล MQTT และ HTTP
- 2) ทดสอบการดึงข้อมูลจาก Thingsboard ไปยัง Unity โดยใช้โปรโตคอล HTTP และ WebSocket
- 3) วิเคราะห์ผลการทดสอบและปรับปรุงระบบตามความต้องการ

## 1.6 แผนการดำเนินการ

กิจกรรมหลัก	เดือน	ผู้รับผิดชอบ
1. ศึกษาปัญหาและงานวิจัยที่เกี่ยวข้อง และยื่นขอรับรองจริยธรรมวิจัยฯ	ก.ค. - ส.ค.	ประจักษ์ จิตเงินมะดัน
2. ศึกษาประสิทธิภาพของแบบจำลองปัญญาประดิษฐ์ที่เหมาะสมกับงานประเภทต่าง ๆ ในอุตสาหกรรมอัตโนมัติ และประยุกต์ใช้งานกับแบบจำลองของเครื่องจักรเสมือนที่เกี่ยวข้อง	ก.ย. - พ.ย.	ประจักษ์ จิตเงินมะดัน

กิจกรรมหลัก	เดือน	ผู้รับผิดชอบ
3. เขียนโปรแกรมเพื่อพัฒนาตัวต้นแบบระบบคู่มือดิจิทัลสำหรับอุตสาหกรรมอัตโนมัติที่สามารถเชื่อมต่อเครื่องจักรภาพและเครื่องจักรเสมือนเข้าด้วยกัน โดยมีการเชื่อมต่อข้อมูลที่เป็น ทดสอบการทำงาน และเก็บข้อมูล	ธ.ค. - มี.ค.	ประจักษ์ จิตเงินมะตัน
4. วิเคราะห์ผลจากข้อมูลที่รวบรวมไว้ ทำการปรับปรุงการเชื่อมต่อหรือการแลกเปลี่ยนข้อมูลตามความเหมาะสม	เม.ย.	ประจักษ์ จิตเงินมะตัน
5. เขียนบทความวิจัยและตีพิมพ์บทความวิจัย	พ.ค.	ประจักษ์ จิตเงินมะตัน
6. สรุปผลและทำรายงานฉบับสมบูรณ์	มิ.ย.	ประจักษ์ จิตเงินมะตัน

## 1.7 ประโยชน์ที่คาดว่าจะได้รับ

การศึกษานี้คาดหวังที่จะได้รับประโยชน์ดังนี้

- 1) สามารถเพิ่มประสิทธิภาพในการสื่อสารแบบเรียลไทม์ในระบบคู่มือดิจิทัล
- 2) ลดความหน่วงเวลาในการส่งข้อมูลและเพิ่มความเร็วในการสื่อสาร
- 3) เพิ่มความสามารถในการตรวจสอบและทำนายปัญหาที่อาจเกิดขึ้นในระบบ
- 4) ช่วยลดค่าใช้จ่ายและเพิ่มความปลอดภัยในการดำเนินงานในอุตสาหกรรมระบบอัตโนมัติ
- 5) ช่วยเพิ่มประสิทธิภาพในการจัดการทรัพยากรและลดการสูญเสียที่ไม่จำเป็น

## 2 ทฤษฎีและโครงการที่เกี่ยวข้อง

### 2.1 ทฤษฎีที่เกี่ยวข้อง

#### 2.1.1 คู่แฝดดิจิทัลในอุตสาหกรรมการผลิต (Digital Twin in Manufacturing)

คู่แฝดดิจิทัลเป็นองค์ประกอบสำคัญของความพยายามในอุตสาหกรรม 4.0 และได้รับความนิยมน้อยมีนัยสำคัญในสภาพแวดล้อมอุตสาหกรรม Tao และคณะ (2019) ได้นำเสนอแผนงานที่ครอบคลุมสำหรับการนำคู่แฝดดิจิทัลไปใช้ในการออกแบบผลิตภัณฑ์ การผลิต และการให้บริการ พวกเขาแสดงให้เห็นว่าโมเดลเสมือนสามารถแทนสินทรัพย์จริงได้แบบเรียลไทม์ โดยเน้นย้ำถึงความสำคัญของการเคลื่อนย้ายข้อมูลระหว่างอุปกรณ์ดิจิทัลและอุปกรณ์ทางกายภาพ Lu และคณะ (2020) ได้ให้ภาพรวมเชิงลึกของการผลิตอัจฉริยะที่ขับเคลื่อนด้วยคู่แฝดดิจิทัล รวมถึงแบบจำลองอ้างอิงที่เป็นระบบและรายการความท้าทายในการวิจัยที่สำคัญ พวกเขาได้วิเคราะห์ความหมายของคู่แฝดดิจิทัลในสภาพแวดล้อมการผลิตอย่างเป็นระบบและระบุการประยุกต์ใช้ที่หลากหลายในหลายอุตสาหกรรม งานของพวกเขาแสดงให้เห็นว่าระบบคู่แฝดดิจิทัลต้องบูรณาการเทคโนโลยีต่าง ๆ อย่างราบรื่นในขณะที่ต้องจัดการกับปัญหาทางเทคนิคและองค์กร การพัฒนาล่าสุดในเทคโนโลยีคู่แฝดดิจิทัลได้รวมข้อมูลเซ็นเซอร์แบบเรียลไทม์กับแบบจำลองคอมพิวเตอร์ Fuller และคณะ (2020) ได้พัฒนาวิธีการที่ช่วยให้ข้อมูลการดำเนินงานสามารถอัปเดตแบบจำลองอย่างต่อเนื่อง ทำให้คู่แฝดดิจิทัลสามารถเติบโตและพัฒนาไปพร้อมกับเวอร์ชันทางกายภาพ วิธีการนี้แก้ไขปัญหาคอขวดของแบบจำลอง ซึ่งเกิดขึ้นเมื่อสำเนาเสมือนค่อย ๆ แตกต่างจากส่วนประกอบทางกายภาพเนื่องจากการเปลี่ยนแปลงของสภาพแวดล้อมหรือการสึกหรอของส่วนประกอบตามปกติ

Zheng และคณะ (2021) ได้สำรวจแนวคิดของคู่แฝดดิจิทัลเชิงความคิด (Cognitive Digital Twin) ซึ่งรวมความสามารถด้านปัญญาประดิษฐ์เพื่อให้คู่แฝดดิจิทัลสามารถเรียนรู้และปรับตัวจากข้อมูลที่ได้รับ งานวิจัยของพวกเขาแสดงให้เห็นว่าคู่แฝดดิจิทัลเชิงความคิดสามารถช่วยในการตัดสินใจที่ซับซ้อนและปรับปรุงกระบวนการผลิตได้อย่างมีประสิทธิภาพมากขึ้น ความสำคัญของคู่แฝดดิจิทัลในการผลิตแบบอัตโนมัติยังได้รับการยืนยันโดย Deng และคณะ (2021) ซึ่งได้ศึกษาวิวัฒนาการของตัวแทนอาคารอัจฉริยะในอุตสาหกรรม AEC-FM (สถาปัตยกรรม วิศวกรรม การก่อสร้าง และการจัดการสิ่งอำนวยความสะดวก) พวกเขาแสดงให้เห็นว่าคู่แฝดดิจิทัลเป็นขั้นตอนต่อไปในวิวัฒนาการของการสร้างแบบจำลองข้อมูลอาคาร (BIM) และสามารถนำไปใช้เพื่อปรับปรุงการออกแบบ การก่อสร้าง และการบำรุงรักษาอาคาร

#### 2.1.2 โครงข่ายประสาทเทียมสำหรับการทำนายอนุกรมเวลาในการผลิต (Neural Networks for Time Series Prediction in Manufacturing)

เทคนิคการเรียนรู้เชิงลึกได้แสดงแนวโน้มที่น่าสนใจในการสร้างแบบจำลองกระบวนการอุตสาหกรรมและการทำนายประสิทธิภาพของอุปกรณ์ในอนาคต Zhang และคณะ (2019) ได้พัฒนาวิธีการใหม่โดยใช้การเรียนรู้เชิงลึกเพื่อกำหนดอายุการใช้งานที่เหลืออยู่ของระบบ วิธีนี้ใช้สถาปัตยกรรมเครือข่ายประสาทเทียมขั้นสูงเพื่อระบุแนวโน้มในข้อมูลเซ็นเซอร์ งานของพวกเขาแสดงให้เห็นว่าการทำนายของพวกเขามีความแม่นยำ

มากกว่าวิธีทางสถิติมาตรฐาน โดยเฉพาะสำหรับระบบที่ซับซ้อนที่มีรูปแบบการเชื่อมต่อแบบไม่เชิงเส้น Zhao และคณะ (2019) ได้ทำการวิเคราะห์อย่างละเอียดเกี่ยวกับการประยุกต์ใช้การเรียนรู้เชิงลึกในการตรวจสอบสุขภาพของเครื่องจักร พวกเขาได้ตรวจสอบสถาปัตยกรรมเครือข่ายประสาทเทียมที่แตกต่างกันและประสิทธิภาพในสภาพแวดล้อมการผลิตต่าง ๆ งานวิจัยของพวกเขาแสดงให้เห็นว่าเครือข่ายประสาทเทียมแบบคอนโวลูชัน (CNNs) และเครือข่ายประสาทเทียมแบบย้อนกลับ (RNNs) สามารถจับข้อมูลทั้งเชิงพื้นที่และเชิงเวลาจากข้อมูลเซ็นเซอร์หลายตัวแปร ซึ่งช่วยปรับปรุงการตรวจจับข้อบกพร่องและการทำนายอายุการใช้งาน

Makridakis และคณะ (2022) ได้เปรียบเทียบวิธีการทางสถิติ การเรียนรู้ของเครื่อง และการเรียนรู้เชิงลึกสำหรับการคาดการณ์ พวกเขาพบว่าโมเดลการเรียนรู้เชิงลึกมีความเชี่ยวชาญในการจับรูปแบบที่ซับซ้อนในข้อมูลอนุกรมเวลา โดยเฉพาะอย่างยิ่งกับข้อมูลการฝึกอบรมที่อุดมสมบูรณ์และความสัมพันธ์แบบไม่เชิงเส้นที่ซับซ้อน การศึกษาหลายชิ้นยังได้ยืนยันว่าเทคนิคการเรียนรู้เชิงลึก เช่น CNNs และเครือข่าย Long Short-Term Memory (LSTMs) สามารถตรวจจับรูปแบบที่ซับซ้อนได้ดีกว่าวิธีการแบบดั้งเดิมและปรับตัวได้ดีกว่าต่อการเปลี่ยนแปลงในระบบการผลิต การทบทวนของพวกเขาเน้นย้ำว่าวิธีการเรียนรู้เชิงลึกสามารถดึงคุณลักษณะที่เกี่ยวข้องโดยอัตโนมัติโดยไม่ต้องมีความรู้ในโดเมนที่กว้างขวาง ทำให้มีคุณค่าสำหรับระบบการผลิตที่ซับซ้อนมาก

### 2.1.3 สถาปัตยกรรมไฮบริด CNN-LSTM (Hybrid CNN-LSTM Architecture)

การผสมผสาน CNNs และ LSTMs ในการออกแบบแบบผสมสร้างความเป็นเลิศในการสร้างแบบจำลองลำดับ โดยเฉพาะเมื่อคุณลักษณะทั้งเชิงพื้นที่และเชิงเวลามีความสำคัญ วิธีการแบบผสมเหล่านี้ใช้ประโยชน์จากข้อดีที่สุดของแต่ละการออกแบบ: CNNs เชี่ยวชาญในการดึงคุณลักษณะเชิงพื้นที่จากข้อมูลป้อนเข้า ในขณะที่ LSTMs เชี่ยวชาญในการค้นหาว่าข้อมูลลำดับได้รับผลกระทบจากเวลาอย่างไร ด้วยกัน พวกเขาให้วิธีที่ครอบคลุมในการดูกระบวนการผลิตที่ซับซ้อนทั้งในมิติเชิงพื้นที่และเชิงเวลา Muthukumar และ Philip (2022) ได้ขยายวิธีการแบบผสมนี้โดยสร้างแบบจำลองผสม CNN-LSTM แบบการเรียนรู้เชิงลึกสำหรับการประมาณอายุการใช้งานที่เหลืออยู่ ผลลัพธ์ของพวกเขาแสดงให้เห็นถึงการปรับปรุงที่สำคัญในสถานการณ์ที่มีการพึ่งพาหลายพารามิเตอร์ที่ซับซ้อน ซึ่งแสดงให้เห็นว่าวิธีการแบบผสมช่วยเพิ่มประสิทธิภาพของแบบจำลองการทำนายในสภาพแวดล้อมการผลิตอย่างมีนัยสำคัญ งานของพวกเขาแสดงให้เห็นว่าการใช้การออกแบบที่แตกต่างกันแต่เกี่ยวข้องกันเหล่านี้ร่วมกันสามารถสร้างการทำนายที่แม่นยำและเชื่อถือได้มากกว่าการใช้ตัวอย่างใดอย่างหนึ่งเพียงอย่างเดียว งานวิจัยของ Liu และคณะ (2021) เกี่ยวกับคู่มือดิจิทัลได้สำรวจแนวคิด เทคโนโลยี และการประยุกต์ใช้ในอุตสาหกรรม พวกเขาเน้นย้ำถึงความสำคัญของเทคนิคการเรียนรู้เชิงลึก เช่น CNNs และ LSTMs ในการปรับปรุงความสามารถในการวิเคราะห์และการทำนายของระบบคู่มือดิจิทัล

## 2.1.4 โพรโตคอลการสื่อสารสำหรับระบบคู่แฝดดิจิทัล (Communication Protocols for Digital Twin Systems)

โพรโตคอลการสื่อสารเป็นองค์ประกอบที่สำคัญของระบบคู่แฝดดิจิทัล โดยมีบทบาทสำคัญในการอำนวยความสะดวกในการแลกเปลี่ยนข้อมูลระหว่างอุปกรณ์ทางกายภาพและคู่เสมือนดิจิทัลของพวกเขา Wukkadada และคณะ (2018) ได้เปรียบเทียบโพรโตคอล HTTP และ MQTT ในบริบทของอินเทอร์เน็ตของสรรพสิ่ง (IoT) พวกเขาพบว่า MQTT มีประสิทธิภาพเหนือกว่า HTTP ในแง่ของการใช้แบนด์วิดท์และความล่าช้า ทำให้เหมาะสำหรับการสื่อสารแบบเรียลไทม์ในระบบคู่แฝดดิจิทัล ในช่วงแรกของการวิจัยของเรา เราได้ประเมินประสิทธิภาพของโพรโตคอล MQTT, HTTP และ WebSocket ในการตั้งค่าคู่แฝดดิจิทัล ผลการค้นพบของเราแสดงให้เห็นว่า MQTT มีความล่าช้า น้อยกว่า HTTP อย่างมีนัยสำคัญเมื่อส่งข้อมูลไปยังแพลตฟอร์ม IoT และ WebSocket แสดงข้อได้เปรียบเล็กน้อยเหนือ HTTP ในการดึงข้อมูลจากแพลตฟอร์ม IoT ไปยังเครื่องยูนิต Unity 3D ผลลัพธ์เหล่านี้สอดคล้องกับงานวิจัยของ Wukkadada และคณะ และยืนยันความเหมาะสมของ MQTT และ WebSocket สำหรับการสื่อสารแบบเรียลไทม์ในระบบคู่แฝดดิจิทัล

## 2.2 งานที่เกี่ยวข้อง (Related Works)

### 2.2.1 ระบบคู่แฝดดิจิทัลสำหรับการผลิตอัจฉริยะ (Digital Twin Systems for Smart Manufacturing)

Hu และคณะ (2019) ได้พัฒนาระบบคู่แฝดดิจิทัลบนคลาวด์สำหรับการผลิตอัจฉริยะโดยใช้มาตรฐาน MTConnect โครงการนี้แสดงให้เห็นว่าการใช้โพรโตคอลการสื่อสารมาตรฐานเพื่อเชื่อมต่ออุปกรณ์การผลิตต่าง ๆ กับระบบคู่แฝดดิจิทัลเดียว ช่วยให้สามารถตรวจสอบแบบเรียลไทม์และการบำรุงรักษาเชิงทำนายได้ การใช้งานของพวกเขาแสดงให้เห็นว่าคู่แฝดดิจิทัลสามารถนำไปใช้ในสภาพแวดล้อมจริงเพื่อปรับปรุงประสิทธิภาพการผลิตและความน่าเชื่อถือของเครื่องจักร Moyné และคณะ (2020) ได้พัฒนารอบงานคู่แฝดดิจิทัลที่ขับเคลื่อนด้วยความต้องการ โดยมุ่งเน้นที่ข้อกำหนดและโอกาสสำหรับการนำไปใช้ งานของพวกเขาให้แนวทางในการออกแบบและใช้งานคู่แฝดดิจิทัลที่ตอบสนองต่อความต้องการเฉพาะของอุตสาหกรรมการผลิต พวกเขาเสนอกรอบงานที่สามารถปรับให้เข้ากับข้อกำหนดเฉพาะของแต่ละอุตสาหกรรม ทำให้การใช้งานคู่แฝดดิจิทัลมีความยืดหยุ่นและมีประสิทธิภาพมากขึ้น Raza และคณะ (2020) ได้เสนอกรอบงานคู่แฝดดิจิทัลสำหรับอุตสาหกรรม 4.0 ที่ช่วยให้เกิดการผลิตยุคถัดไป พวกเขาอธิบายว่าคู่แฝดดิจิทัลสามารถช่วยในการตัดสินใจแบบเรียลไทม์และการปรับกระบวนการที่สำคัญสำหรับโรงงานอัจฉริยะได้อย่างไร พวกเขาเน้นย้ำถึงความสำคัญของการบูรณาการเทคโนโลยีคู่แฝดดิจิทัลกับเทคโนโลยีอื่น ๆ เช่น IoT, Big Data และ AI เพื่อสร้างระบบการผลิตที่มีประสิทธิภาพและยืดหยุ่นมากขึ้น

## 2.2.2 การประยุกต์ใช้การเรียนรู้เชิงลึกกับระบบคู่แฝดดิจิทัล (Application of Deep Learning to Digital Twin Systems)

Lu และคณะ (2020) ได้ศึกษาการบูรณาการเทคนิคการเรียนรู้เชิงลึกกับระบบคู่แฝดดิจิทัลสำหรับการผลิตอัจฉริยะ พวกเขาเสนอแนวคิดของคู่แฝดดิจิทัลที่ขับเคลื่อนด้วยข้อมูล ซึ่งใช้ข้อมูลเซ็นเซอร์และเทคนิคการเรียนรู้เชิงลึกเพื่อปรับปรุงความแม่นยำของการจำลองและการทำนาย พวกเขาแสดงให้เห็นว่าการบูรณาการเทคนิคการเรียนรู้เชิงลึกกับคู่แฝดดิจิทัลสามารถปรับปรุงการตรวจสอบกระบวนการ การควบคุมคุณภาพ และการบำรุงรักษาเชิงทำนายได้อย่างมีประสิทธิภาพ Zhang และคณะ (2019) ได้เสนอวิธีการเรียนรู้เชิงลึกสำหรับการทำนายอายุการใช้งานที่เหลืออยู่ของระบบ (Remaining Useful Life: RUL) พวกเขาได้พัฒนาสถาปัตยกรรมเครือข่ายประสาทเทียมเชิงลึกที่สามารถเรียนรู้คุณลักษณะที่เกี่ยวข้องจากข้อมูลเซ็นเซอร์และทำนายอายุการใช้งานที่เหลืออยู่ได้อย่างแม่นยำ งานของพวกเขาแสดงให้เห็นว่าการเรียนรู้เชิงลึกสามารถปรับปรุงความแม่นยำของการบำรุงรักษาเชิงทำนายได้อย่างมีประสิทธิภาพ Fuller และคณะ (2020) ได้ศึกษาเทคโนโลยีที่ช่วยให้เกิดคู่แฝดดิจิทัล ความท้าทาย และพื้นที่วิจัยที่เปิดกว้าง พวกเขาเน้นย้ำถึงความสำคัญของการเรียนรู้ของเครื่องและเทคนิคการเรียนรู้เชิงลึกในการปรับปรุงความสามารถในการทำนายของคู่แฝดดิจิทัล พวกเขาพบว่า CNNs และ LSTMs เป็นเทคนิคที่มีประสิทธิภาพสำหรับการตรวจจับรูปแบบและการทำนายในระบบคู่แฝดดิจิทัล

## 2.2.3 การเพิ่มประสิทธิภาพพารามิเตอร์การผลิตด้วยคู่แฝดดิจิทัล (Optimization of Manufacturing Parameters Using Digital Twin)

การปรับพารามิเตอร์ให้เหมาะสมในการผลิตเป็นหนึ่งในการประยุกต์ใช้ที่สำคัญของคู่แฝดดิจิทัล Madni และคณะ (2019) ได้ศึกษาการใช้เทคโนโลยีคู่แฝดดิจิทัลในวิศวกรรมระบบที่ขึ้นอยู่กับแบบจำลอง พวกเขาเสนอกรอบงานที่ใช้คู่แฝดดิจิทัลเพื่อปรับให้การออกแบบและการดำเนินงานของระบบที่ซับซ้อนให้เหมาะสมที่สุด พวกเขาแสดงให้เห็นว่าคู่แฝดดิจิทัลสามารถช่วยให้วิศวกรสำรวจพื้นที่ออกแบบที่กว้างขึ้นและระบุการตั้งค่าพารามิเตอร์ที่เหมาะสมที่สุดสำหรับเป้าหมายที่กำหนด Leonardo และคณะ (2020) ได้พัฒนาโมเดลคู่แฝดดิจิทัลสำหรับการประเมินอันตรายจากการชนในเวลาจริง พวกเขาใช้เซ็นเซอร์และเทคนิคการเรียนรู้ของเครื่องเพื่อสร้างแบบจำลองเสมือนของสภาพแวดล้อมการก่อสร้างและทำนายความเสี่ยงจากการชนระหว่างคนงานและอุปกรณ์ งานของพวกเขาแสดงให้เห็นว่าคู่แฝดดิจิทัลสามารถปรับปรุงความปลอดภัยในสถานที่ทำงานได้อย่างมีประสิทธิภาพโดยการระบุและป้องกันอันตรายที่อาจเกิดขึ้น Deng และคณะ (2021) ได้ศึกษาวิวัฒนาการของตัวแทนอาคารอัจฉริยะในอุตสาหกรรม AEC-FM จาก BIM ไปสู่คู่แฝดดิจิทัล พวกเขาแสดงให้เห็นว่าคู่แฝดดิจิทัลสามารถช่วยปรับปรุงการออกแบบ การก่อสร้าง และการบำรุงรักษาอาคารได้อย่างไรโดยการจำลองและทดสอบการตั้งค่าพารามิเตอร์ที่แตกต่างกันก่อนการนำไปใช้จริง

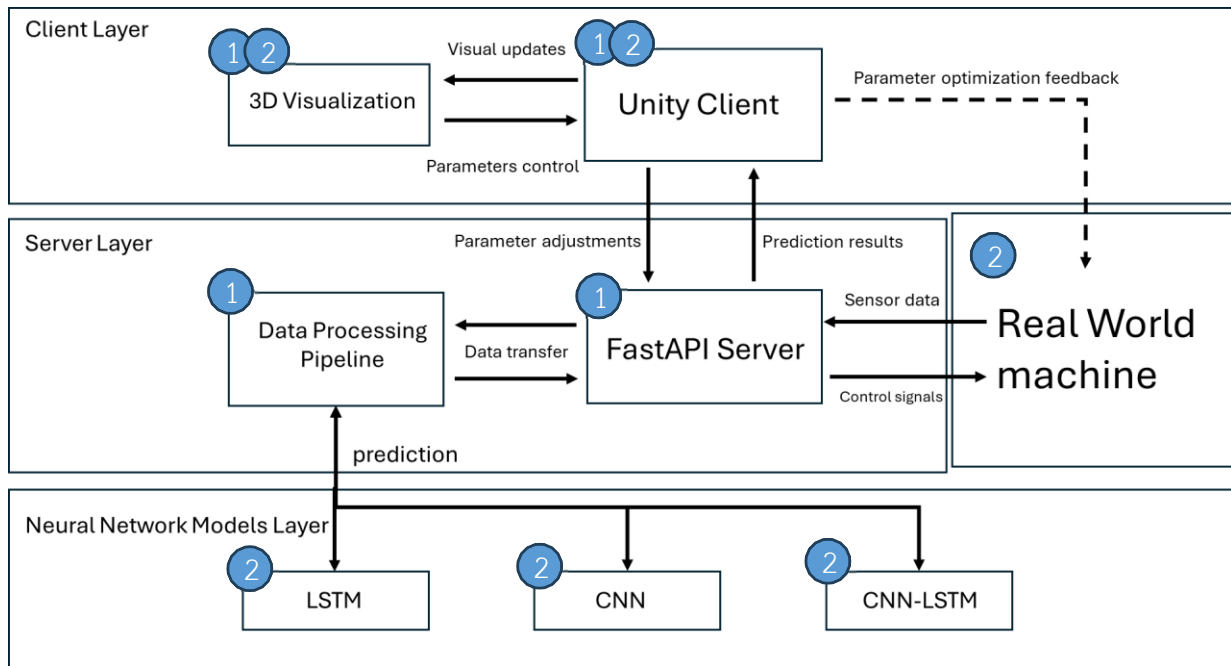
## 2.2.4 การบำรุงรักษาเชิงพยากรณ์โดยใช้คู่มือดิจิทัล (Predictive Maintenance Using Digital Twin)

การบำรุงรักษาเชิงทำนายเป็นหนึ่งในการประยุกต์ใช้ที่สำคัญที่สุดของคู่มือดิจิทัลในการผลิต Muthukumar และ Philip (2022) ได้พัฒนาโมเดลผสม CNN-LSTM สำหรับการทำนายอายุการใช้งานที่เหลืออยู่ของอุปกรณ์ พวกเขาแสดงให้เห็นว่าสถาปัตยกรรมแบบผสมสามารถปรับปรุงความแม่นยำในการทำนายได้อย่างมีนัยสำคัญเมื่อเทียบกับสถาปัตยกรรมเดี่ยว Cattaneo และ Macchi (2019) ได้ศึกษาการใช้คู่มือดิจิทัลสำหรับการบำรุงรักษาเชิงทำนายในอุตสาหกรรมการผลิต พวกเขาเสนอกรอบงานที่ใช้ข้อมูลเซ็นเซอร์และเทคนิคการเรียนรู้ของเครื่องเพื่อทำนายความล้มเหลวของอุปกรณ์และวางแผนกิจกรรมการบำรุงรักษาได้อย่างเหมาะสม พวกเขาแสดงให้เห็นว่าการบำรุงรักษาเชิงทำนายที่ขับเคลื่อนด้วยคู่มือดิจิทัลสามารถลดค่าใช้จ่ายในการบำรุงรักษาและเพิ่มความพร้อมใช้งานของอุปกรณ์ได้อย่างมีนัยสำคัญ ในงานวิจัยของเรา เรามุ่งเน้นการใช้โมเดลผสม CNN-LSTM เพื่อเพิ่มความแม่นยำในการทำนายพฤติกรรมของอุปกรณ์ภายใต้การตั้งค่าพารามิเตอร์ที่แตกต่างกัน ข้อมูลนี้สามารถใช้สำหรับการวางแผนการบำรุงรักษาเชิงทำนาย ช่วยหลีกเลี่ยงการตั้งค่าพารามิเตอร์ที่อาจนำไปสู่ความล้มเหลวของอุปกรณ์

### 3 วิธีการดำเนินงานโครงการ

#### 3.1 สถาปัตยกรรมระบบ

การพัฒนาแบบจำลองดิจิทัลในงานวิจัยระยะที่ 2 นี้ได้ต่อยอดจากระยะที่ 1 โดยมุ่งเน้นการเพิ่มความสามารถในการวิเคราะห์และทำนายด้วยโครงข่ายประสาทเทียม สถาปัตยกรรมของระบบประกอบด้วยองค์ประกอบหลัก 3 ส่วน ดังแสดงในภาพที่ 1



ภาพที่ 1 สถาปัตยกรรมระบบคู่มือดิจิทัล

ระบบคู่มือดิจิทัลนี้มีการเชื่อมต่อกับเครื่องจักรจริง (Real World machine) ซึ่งส่งข้อมูลเซนเซอร์ (Sensor data) เข้าสู่ระบบผ่าน FastAPI Server ใน Server Layer ข้อมูลจะถูกประมวลผลและวิเคราะห์โดยใช้โมเดลโครงข่ายประสาทเทียมในชั้น Neural Network Models Layer ก่อนที่ผลลัพธ์จะถูกส่งไปแสดงผลบนแบบจำลอง 3D Visualization ในชั้น Client Layer ทำให้ผู้ใช้งานสามารถติดตามสถานะของเครื่องจักรและได้รับการแจ้งเตือนล่วงหน้าเมื่อมีแนวโน้มที่จะเกิดความเสียหาย

##### 3.1.1 ชั้นอุปกรณ์ทางกายภาพ (Physical Device Layer)

ในชั้นนี้ประกอบด้วยอุปกรณ์ทางกายภาพที่ใช้ในการผลิต ได้แก่ PLCnext AX C F 2152 และเซ็นเซอร์ต่าง ๆ ที่ใช้ในการเก็บข้อมูลพารามิเตอร์การผลิต เช่น

- เซ็นเซอร์วัดอุณหภูมิ (Temperature Sensors) สำหรับวัดอุณหภูมิอากาศและอุณหภูมิกระบวนการ
- เซ็นเซอร์วัดความเร็วการหมุน (Rotational Speed Sensors)
- เซ็นเซอร์วัดแรงบิด (Torque Sensors)
- ระบบติดตามการสึกหรอของเครื่องมือ (Tool Wear Monitoring System)

อุปกรณ์เหล่านี้ทำหน้าที่เก็บรวบรวมข้อมูลต่าง ๆ จากการทำงานของเครื่องจักรในเวลาจริง

### 3.1.2 ชั้นการสื่อสารและการจัดการข้อมูล (Communication and Data Management Layer)

ชั้นนี้รับผิดชอบในการส่งข้อมูลจากอุปกรณ์ทางกายภาพไปยังระบบจัดเก็บและประมวลผลข้อมูล ประกอบด้วย

- โพรโตคอล MQTT สำหรับการส่งข้อมูลจาก PLCnext ไปยัง Thingsboard
- Node-RED สำหรับการประมวลผลข้อมูลเบื้องต้นและการสร้างคุณลักษณะเพิ่มเติม (Feature Engineering)
- Thingsboard ทำหน้าที่เป็นแพลตฟอร์ม IoT สำหรับจัดเก็บ แสดงผล และจัดการข้อมูล
- โพรโตคอล WebSocket สำหรับการดึงข้อมูลจาก Thingsboard ไปยัง Unity

ชั้นนี้ถูกออกแบบโดยอ้างอิงจากผลการวิจัยในระยะที่ 1 ที่พบว่า MQTT และ WebSocket มีประสิทธิภาพสูงในการส่งและดึงข้อมูลตามลำดับ

### 3.1.3 ชั้นการวิเคราะห์และการทำนาย (Analytics and Prediction Layer)

ชั้นนี้เป็นส่วนที่พัฒนาเพิ่มขึ้นในระยะที่ 2 ประกอบด้วยแบบจำลองโครงข่ายประสาทเทียม 3 รูปแบบ

- แบบจำลอง CNN สำหรับการจับรูปแบบเชิงพื้นที่ของพารามิเตอร์
- แบบจำลอง LSTM สำหรับการจับรูปแบบเชิงเวลาของพารามิเตอร์
- แบบจำลองไฮบริด CNN-LSTM ที่ผสมผสานความสามารถของทั้งสองแบบจำลอง

แบบจำลองเหล่านี้ได้รับการฝึกสอนด้วยชุดข้อมูล AI4I 2020 Predictive Maintenance Dataset และถูกใช้ในการทำนายพฤติกรรมของอุปกรณ์ภายใต้การตั้งค่าพารามิเตอร์ที่แตกต่างกัน

### 3.1.4 ชั้นการแสดงผลและปฏิสัมพันธ์ (Visualization and Interaction Layer)

ชั้นนี้รับผิดชอบในการแสดงผลแบบจำลองคู่แข่งดิจิทัลและการโต้ตอบกับผู้ใช้ ประกอบด้วย

- Unity เป็นเครื่องมือพัฒนาแบบจำลอง 3 มิติและส่วนติดต่อผู้ใช้
- ส่วนแสดงผลแบบ 3 มิติที่แสดงสถานะของอุปกรณ์และผลการทำนาย
- แผงควบคุมสำหรับปรับพารามิเตอร์ต่าง ๆ และดูผลลัพธ์แบบเรียลไทม์
- การแสดงผลเชิงวิเคราะห์ เช่น แผนภูมิความสำคัญของพารามิเตอร์และความสัมพันธ์ระหว่างพารามิเตอร์

ชั้นนี้ช่วยให้ผู้ปฏิบัติงานสามารถสำรวจและเข้าใจความสัมพันธ์ระหว่างพารามิเตอร์การผลิตและผลกระทบต่อประสิทธิภาพและสุขภาพของอุปกรณ์

สถาปัตยกรรมระบบตามที่อธิบายข้างต้นได้รับการออกแบบให้มีความยืดหยุ่น ขยายตัวได้ และสามารถปรับใช้ในสภาพแวดล้อมการผลิตที่หลากหลายได้ การแบ่งระบบออกเป็นชั้นต่าง ๆ ช่วยให้สามารถพัฒนาและปรับปรุงแต่ละส่วนได้อย่างอิสระ โดยไม่กระทบต่อส่วนอื่น ๆ

## 3.2 การไหลของข้อมูล

กระบวนการไหลของข้อมูลในระบบคู่มือดิจิทัลที่พัฒนาขึ้นมีขั้นตอนหลักดังนี้

### 3.2.1 การเก็บรวบรวมข้อมูล (Data Collection)

กระบวนการเก็บรวบรวมข้อมูลเริ่มต้นจากเซ็นเซอร์ต่าง ๆ ที่ติดตั้งกับอุปกรณ์ทางกายภาพทำการวัดและเก็บรวบรวมข้อมูลพารามิเตอร์การทำงาน ได้แก่ อุณหภูมิอากาศ, อุณหภูมิกระบวนการ, ความเร็วการหมุน, แรงบิด, และการสั่นของเครื่องมือ จากนั้นข้อมูลจากเซ็นเซอร์ถูกส่งไปยัง PLCnext AXC F 2152 ซึ่งทำหน้าที่เป็นตัวควบคุมหลัก และมีการเข้ารหัสข้อมูลในรูปแบบ JSON เพื่อเตรียมสำหรับการส่งผ่านเครือข่าย

### 3.2.2 การส่งข้อมูลและการประมวลผลเบื้องต้น (Data Transmission and Pre-processing)

PLCnext ใช้โปรโตคอล MQTT เพื่อส่งข้อมูลไปยัง Node-RED ด้วยความถี่ในการส่งข้อมูล 10 ครั้งต่อวินาที Node-RED ทำการประมวลผลข้อมูลเบื้องต้น เช่น การกรองข้อมูลที่ผิดปกติหรือไม่ถูกต้อง การสร้างคุณลักษณะเพิ่มเติม (Feature Engineering) เช่น การคำนวณอัตราการเปลี่ยนแปลงของอุณหภูมิ, อัตราส่วนระหว่างแรงบิดและความเร็วการหมุน และการจัดรูปแบบข้อมูลให้เหมาะสมสำหรับการจัดเก็บและการวิเคราะห์ จากนั้นข้อมูลที่ผ่านการประมวลผลเบื้องต้นถูกส่งไปจัดเก็บในแพลตฟอร์ม Thingsboard

### 3.2.3 การวิเคราะห์และการทำนายด้วยแบบจำลองโครงข่ายประสาทเทียม (Analysis and Prediction)

การ Unity ใช้โปรโตคอล WebSocket เพื่อดึงข้อมูลจาก Thingsboard แบบเรียลไทม์ ข้อมูลถูกแปลงให้อยู่ในรูปแบบที่เหมาะสมสำหรับแบบจำลองแต่ละประเภท สำหรับ CNN ข้อมูลถูกจัดให้อยู่ในรูปทรง (24, 5, 1) สำหรับ LSTM ข้อมูลถูกจัดให้อยู่ในรูปของลำดับ (sequence) ความยาว 24 timesteps และสำหรับ CNN-LSTM ข้อมูลถูกจัดให้อยู่ในรูปทรงที่เหมาะสมสำหรับทั้ง CNN และ LSTM แบบจำลองทั้งสาม (CNN, LSTM, และ CNN-LSTM) ทำการทำนายพารามิเตอร์การผลิต เช่น อุณหภูมิกระบวนการ และความเป็นไปได้ของความล้มเหลว และผลการทำนายจากแบบจำลองทั้งสามถูกนำมารวมกันโดยใช้เทคนิค Ensemble เพื่อให้ได้ผลลัพธ์ที่แม่นยำยิ่งขึ้น

### 3.2.4 การแสดงผลและปฏิสัมพันธ์กับผู้ใช้ (Visualization and User Interaction)

ผลการทำนายถูกนำมาแสดงผลผ่านแบบจำลอง 3 มิติใน Unity โดยมีการแมปพารามิเตอร์เข้ากับการแสดงผลที่เหมาะสม เช่น อุณหภูมิแสดงด้วยสีจากน้ำเงิน (เย็น) ถึงแดง (ร้อน) ความเร็วการหมุนแสดงด้วยการหมุนของชิ้นส่วนในแบบจำลอง 3 มิติ แรงบิดแสดงด้วยความเข้มของแสงหรือขนาด และการสั่นของเครื่องมือแสดงด้วยการเปลี่ยนแปลงพื้นผิว และผู้ใช้สามารถปรับพารามิเตอร์ต่าง ๆ ผ่านแผงควบคุมและเห็นผลการเปลี่ยนแปลงแบบเรียลไทม์ และสามารถบันทึกการตั้งค่าพารามิเตอร์ต่าง ๆ และเปรียบเทียบผลลัพธ์ที่ได้

### 3.2.5 การอัปเดตและปรับปรุงแบบจำลอง (Model Update and Improvement)

การระบบทำการเก็บรวบรวมข้อมูลใหม่อย่างต่อเนื่องจากการทำงานของอุปกรณ์ แบบจำลองได้รับการฝึกสอนซ้ำเป็นประจำด้วยข้อมูลใหม่ที่ได้รับ เพื่อปรับปรุงความแม่นยำในการทำนาย และระบบได้รับการปรับปรุงอย่างต่อเนื่องตามผลการประเมินและข้อเสนอแนะจากผู้ใช้ โดยกระบวนการไหลของข้อมูลนี้ได้รับการออกแบบให้ทำงานแบบเรียลไทม์ ช่วยให้ผู้ใช้ปฏิบัติงานสามารถตรวจสอบและปรับพารามิเตอร์ได้ทันที รวมถึงเห็นผลกระทบที่อาจเกิดขึ้นกับประสิทธิภาพและสุขภาพของอุปกรณ์ การใช้โปรโตคอลการสื่อสารที่มีประสิทธิภาพสูง เช่น MQTT และ WebSocket ช่วยให้การส่งและดึงข้อมูลเป็นไปอย่างรวดเร็ว ซึ่งเป็นสิ่งสำคัญสำหรับระบบคู่มือดิจิทัลที่ต้องการการตอบสนองในเวลาจริง

### 3.3 ชุดข้อมูล

การศึกษานี้ใช้ชุดข้อมูล [AI4I 2020 Predictive Maintenance Dataset](#) ซึ่งเป็นชุดข้อมูลที่จัดทำขึ้นเพื่อใช้ในการพัฒนาแบบจำลองสำหรับการบำรุงรักษาเชิงพยากรณ์ในอุตสาหกรรมการผลิต ชุดข้อมูลนี้เป็นการจำลองข้อมูลการทำงานของเครื่องจักรในสายการผลิตที่ประกอบไปด้วยเครื่องไฮดรอลิก เครื่องกำเนิดลม เครื่องสูบลม และหน่วยทำความเย็น

#### 3.3.1 ลักษณะของชุดข้อมูล

ชุดข้อมูล AI4I 2020 Predictive Maintenance Dataset ประกอบด้วยข้อมูลทั้งสิ้น 10,000 ตัวอย่าง โดยแต่ละตัวอย่างประกอบด้วยคุณลักษณะ (features) ดังนี้

1. คุณลักษณะพื้นฐาน
  - UID: รหัสเฉพาะของแต่ละเครื่องจักร (1-9)
  - Product ID: รหัสผลิตภัณฑ์ (L, M, H โดย L = Low, M = Medium, H = High)
  - Type: ประเภทของเครื่องจักร (L, M, H โดย L = Low, M = Medium, H = High)
2. พารามิเตอร์การทำงาน
  - Air temperature [K]: อุณหภูมิอากาศในหน่วยเคลวิน (ช่วง 295.3 - 304.5 K)
  - Process temperature [K]: อุณหภูมิของกระบวนการในหน่วยเคลวิน (ช่วง 305.7 - 313.8 K)
  - Rotational speed [rpm]: ความเร็วการหมุนในหน่วยรอบต่อนาที (ช่วง 1168 - 2886 rpm)
  - Torque [Nm]: แรงบิดในหน่วยนิวตันเมตร (ช่วง 3.8 - 76.6 Nm)
  - Tool wear [min]: เวลาการใช้งานของเครื่องมือนับจากการเปลี่ยนครั้งล่าสุด ในหน่วยนาที (ช่วง 0 - 253 นาที)
3. คุณลักษณะเป้าหมาย
  - Target: ตัวบ่งชี้ความล้มเหลว (0 = ไม่มีความล้มเหลว, 1 = มีความล้มเหลว)

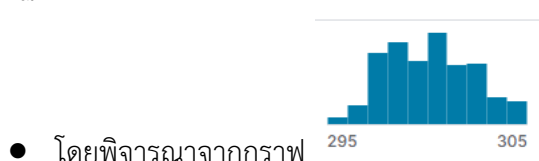
- Failure Type: ประเภทของความล้มเหลว (None, Heat Dissipation Failure, Power Failure, Overstrain Failure, Tool Wear Failure, Random Failures)

### 3.3.2 การวิเคราะห์ข้อมูลเบื้องต้น

ก่อนที่จะนำข้อมูลไปใช้ในการพัฒนาแบบจำลอง เราได้ทำการวิเคราะห์ข้อมูลเบื้องต้นเพื่อทำความเข้าใจลักษณะและคุณสมบัติของข้อมูล

#### 1. การกระจายตัวของข้อมูล

- **อุณหภูมิอากาศ** มีการกระจายตัวแบบปกติ โดยมีค่าเฉลี่ยที่ 300.0 K และส่วนเบี่ยงเบนมาตรฐาน 2.0 K



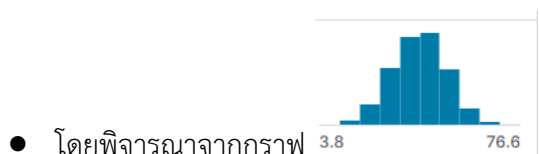
- **อุณหภูมิกระบวนการ** มีการกระจายตัวแบบปกติ โดยมีค่าเฉลี่ยที่ 310.0 K และส่วนเบี่ยงเบนมาตรฐาน 1.5 K



- **ความเร็วการหมุน** มีการกระจายตัวแบบเบ้ขวา ในช่วง 1168 - 2886 rpm



- **แรงบิด** มีการกระจายตัวแบบเบ้ขวา โดยมีค่าเฉลี่ยที่ 40.2 Nm และค่ามัธยฐานที่ 39.5 Nm



- **การสึกหรอของเครื่องมือ** มีการกระจายตัวแบบสม่ำเสมอในช่วง 0 - 253 นาที



#### 2. ความสัมพันธ์ระหว่างคุณลักษณะ

- พบความสัมพันธ์เชิงบวกอย่างมีนัยสำคัญระหว่างอุณหภูมิกระบวนการและความเร็วการหมุน ( $r = 0.87$ )

- พบความสัมพันธ์เชิงบวกปานกลางระหว่างแรงบิดและอุณหภูมิกระบวนการ ( $r = 0.62$ )
  - พบความสัมพันธ์เชิงลบเล็กน้อยระหว่างการสึกหรอของเครื่องมือและประสิทธิภาพการทำงาน ( $r = -0.28$ )
3. ความไม่สมดุลของข้อมูล
- ข้อมูลแสดงความไม่สมดุลอย่างมากในตัวแปรเป้าหมาย โดยมีเพียง 3.5% ของข้อมูลที่มีค่า Target = 1 (มีความล้มเหลว)
  - ในจำนวนข้อมูลที่มีความล้มเหลว ประเภทความล้มเหลวที่พบมากที่สุด คือ Tool Wear Failure (42% ของความล้มเหลวทั้งหมด) รองลงมาคือ Heat Dissipation Failure (28%) และ Power Failure (17%)

### 3.3.3 การเตรียมข้อมูลสำหรับแบบจำลอง

เพื่อให้ข้อมูลเหมาะสมสำหรับการใช้ในแบบจำลองโครงข่ายประสาทเทียม เราได้ทำการเตรียมข้อมูลดังนี้

1. **การจัดการค่าที่หายไป (Missing values)** ตรวจสอบและพบว่าไม่มีค่าที่หายไปในช่วงข้อมูล
2. **การแปลงข้อมูลเชิงคุณภาพ (Categorical variables)** ใช้เทคนิค One-hot encoding แปลงข้อมูลเชิงคุณภาพ เช่น Product ID และ Type เป็นตัวแปรหุ่น (dummy variables)
3. **การแบ่งข้อมูล** แบ่งข้อมูลออกเป็น 3 ส่วน โดยใช้การสุ่มแบบแบ่งชั้น (stratified sampling) เพื่อรักษาสัดส่วนของข้อมูลที่มีความล้มเหลวในแต่ละชุด
  - ชุดข้อมูลสำหรับฝึกสอน (Training set) 70% (7,000 ตัวอย่าง)
  - ชุดข้อมูลสำหรับตรวจสอบ (Validation set) 15% (1,500 ตัวอย่าง)
  - ชุดข้อมูลสำหรับทดสอบ (Test set) 15% (1,500 ตัวอย่าง)
4. **การปรับขนาด (Scaling)** ใช้ Min-Max Scaling เพื่อปรับค่าพารามิเตอร์ทั้งหมดให้อยู่ในช่วง  $[0, 1]$  โดยใช้ค่าต่ำสุดและสูงสุดจากชุดข้อมูลฝึกสอนเท่านั้น เพื่อป้องกันการรั่วไหลของข้อมูล
5. **การแก้ไขความไม่สมดุลของข้อมูล** ใช้เทคนิค Synthetic Minority Over-sampling Technique (SMOTE) เพื่อสร้างข้อมูลเพิ่มเติมในกลุ่มที่มีจำนวนน้อย (ข้อมูลที่มีความล้มเหลว) เฉพาะในชุดข้อมูลฝึกสอนเท่านั้น ทำให้สัดส่วนระหว่างข้อมูลที่ไม่มีความล้มเหลวต่อข้อมูลที่มีความล้มเหลวเป็น 4:1
6. **การสร้าง Sequences** สำหรับแบบจำลอง LSTM และ CNN-LSTM ข้อมูลถูกจัดรูปแบบใหม่เป็น sequences ที่มีความยาว 24 timesteps โดยใช้เทคนิค sliding window ที่มีการซ้อนทับ (overlap) 50%

### 3.3.4 ความเกี่ยวข้องกับการประยุกต์ใช้ในโลกรจริง

ชุดข้อมูล AI4I 2020 Predictive Maintenance Dataset ถูกออกแบบมาให้มีลักษณะใกล้เคียงกับข้อมูลจากสภาพแวดล้อมการผลิตจริง ซึ่งมีความเกี่ยวข้องกับการประยุกต์ใช้ในโรงงานอุตสาหกรรมระบบอัตโนมัติ พารามิเตอร์ในชุดข้อมูล เช่น อุณหภูมิ ความเร็วการหมุน และแรงบิด เป็นพารามิเตอร์ที่วัดได้จริงในกระบวนการผลิตอัตโนมัติ และชุดข้อมูลครอบคลุมประเภทของความล้มเหลวที่พบบ่อยในอุตสาหกรรม

ผลิต ซึ่งช่วยให้แบบจำลองของเราสามารถทำนายปัญหาที่อาจเกิดขึ้นได้หลากหลายรูปแบบ ความไม่สมดุลของข้อมูลในชุดข้อมูลนี้สะท้อนความเป็นจริงในอุตสาหกรรม เนื่องจากในสภาพแวดล้อมการผลิตจริง ความล้มเหลวของเครื่องจักรมักเป็นเหตุการณ์ที่พบบ่อยเมื่อเทียบกับการทำงานปกติ อย่างไรก็ตาม ชุดข้อมูลนี้มีข้อจำกัดบางประการที่ต้องคำนึงถึง เช่น ข้อมูลมีลักษณะเป็นการจำลองมากกว่าข้อมูลจากการทำงานจริง ไม่มีข้อมูลเกี่ยวกับเวลาที่แท้จริงระหว่างตัวอย่างข้อมูลแต่ละตัว และไม่มีข้อมูลเกี่ยวกับบริบทการทำงาน เช่น ชนิดของวัสดุที่กำลังผลิต หรือเงื่อนไขการทำงานเฉพาะด้าน แม้จะมีข้อจำกัดเหล่านี้ เราได้พัฒนาวิธีการที่สามารถประยุกต์ใช้กับข้อมูลจริงในอุตสาหกรรมได้ในอนาคต โดยมุ่งเน้นการทำนายพารามิเตอร์สำคัญ เช่น อุณหภูมิกระบวนการ ซึ่งมีความสัมพันธ์โดยตรงกับความล้มเหลวประเภท Heat Dissipation Failure

### 3.4 สถาปัตยกรรมแบบจำลอง

ในงานวิจัยระยะที่ 2 นี้ เราได้พัฒนาแบบจำลองโครงข่ายประสาทเทียม 3 แบบ เพื่อเปรียบเทียบประสิทธิภาพในการทำนายพารามิเตอร์การผลิต โดยแต่ละแบบจำลองถูกออกแบบมาเพื่อจับรูปแบบข้อมูลที่แตกต่างกัน

#### 3.4.1 แบบจำลอง LSTM

แบบจำลอง Long Short-Term Memory (LSTM) ถูกออกแบบมาเพื่อจับรูปแบบความสัมพันธ์เชิงเวลา (temporal patterns) ในข้อมูลอนุกรมเวลาของพารามิเตอร์การผลิต โครงสร้างของแบบจำลอง LSTM ที่เราพัฒนามีรายละเอียดดังนี้

- **ชั้นข้อมูลเข้า (Input Layer)** รับข้อมูล sequence ความยาว 24 timesteps โดยแต่ละ timestep มี 5 features (อุณหภูมิอากาศ, อุณหภูมิกระบวนการ, ความเร็วการหมุน, แรงบิด, และการสีกหรือของเครื่องมือ)
- **ชั้น LSTM ที่ 1** ประกอบด้วย LSTM units จำนวน 128 units พร้อมกับ Dropout 20% เพื่อป้องกันการ overfitting
- **ชั้น LSTM ที่ 2** ประกอบด้วย LSTM units จำนวน 64 units พร้อมกับ Dropout 20%
- **ชั้นซ่อน (Dense Layer) ที่ 1** ประกอบด้วย 32 neurons พร้อมฟังก์ชัน activation แบบ ReLU
- **ชั้นซ่อน (Dense Layer) ที่ 2** ประกอบด้วย 16 neurons พร้อมฟังก์ชัน activation แบบ ReLU
- **ชั้นข้อมูลออก (Output Layer)** ประกอบด้วย neurons ตามจำนวนพารามิเตอร์ที่ต้องการทำนาย (ในกรณีนี้คือ 1 สำหรับอุณหภูมิกระบวนการ)

แบบจำลองนี้มีพารามิเตอร์ที่สามารถปรับได้ (trainable parameters) ทั้งหมดประมาณ 138,257 พารามิเตอร์

### 3.4.2 แบบจำลอง CNN

แบบจำลอง Convolutional Neural Network (CNN) ถูกออกแบบมาเพื่อจับรูปแบบความสัมพันธ์เชิงพื้นที่ (spatial patterns) ระหว่างพารามิเตอร์ต่าง ๆ โครงสร้างของแบบจำลอง CNN ที่เราพัฒนามีรายละเอียดดังนี้

- **ชั้นข้อมูลเข้า (Input Layer)** รับข้อมูลรูปทรง (24, 5, 1) โดย 24 คือจำนวน timesteps, 5 คือจำนวน features, และ 1 คือจำนวนช่องสัญญาณ (channels)
- **ชั้น Convolutional ที่ 1** ประกอบด้วย 32 filters ขนาด 3x3 พร้อมฟังก์ชัน activation แบบ ReLU
- **ชั้น MaxPooling ที่ 1** ใช้ pool size 2x2 เพื่อลดขนาดของข้อมูล
- **ชั้น Convolutional ที่ 2** ประกอบด้วย 64 filters ขนาด 3x3 พร้อมฟังก์ชัน activation แบบ ReLU
- **ชั้น MaxPooling ที่ 2** ใช้ pool size 2x2
- **ชั้น Flatten** แปลงข้อมูลหลายมิติให้เป็นเวกเตอร์ 1 มิติ
- **ชั้นซ่อน (Dense Layer) ที่ 1** ประกอบด้วย 64 neurons พร้อมฟังก์ชัน activation แบบ ReLU และ Dropout 25%
- **ชั้นซ่อน (Dense Layer) ที่ 2** ประกอบด้วย 32 neurons พร้อมฟังก์ชัน activation แบบ ReLU
- **ชั้นข้อมูลออก (Output Layer)** ประกอบด้วย neurons ตามจำนวนพารามิเตอร์ที่ต้องการทำนาย (ในกรณีนี้คือ 1)

แบบจำลองนี้มีพารามิเตอร์ที่สามารถปรับได้ทั้งหมดประมาณ 83,329 พารามิเตอร์

### 3.4.3 แบบจำลองไฮบริด CNN-LSTM

แบบจำลองไฮบริด CNN-LSTM ถูกออกแบบมาเพื่อผสมผสานความสามารถในการจับรูปแบบเชิงพื้นที่ของ CNN และความสามารถในการจับรูปแบบเชิงเวลาของ LSTM เข้าด้วยกัน โครงสร้างของแบบจำลองมีรายละเอียดดังนี้

- **ชั้นข้อมูลเข้า (Input Layer)** รับข้อมูลรูปทรง (24, 5, 1)
- **ชั้น Convolutional ที่ 1** ประกอบด้วย 32 filters ขนาด 3x1 พร้อมฟังก์ชัน activation แบบ ReLU
- **ชั้น Convolutional ที่ 2** ประกอบด้วย 64 filters ขนาด 3x1 พร้อมฟังก์ชัน activation แบบ ReLU
- **ชั้น Reshape** ปรับรูปร่างข้อมูลให้เหมาะสมกับการป้อนเข้า LSTM
- **ชั้น LSTM ที่ 1** ประกอบด้วย LSTM units จำนวน 128 units พร้อม return\_sequences=True
- **ชั้น LSTM ที่ 2** ประกอบด้วย LSTM units จำนวน 64 units
- **ชั้น Dropout** ใช้อัตรา Dropout 20%

- **ชั้นซ่อน (Dense Layer) ที่ 1** ประกอบด้วย 32 neurons พร้อมฟังก์ชัน activation แบบ ReLU
- **ชั้นข้อมูลออก (Output Layer)** ประกอบด้วย neurons ตามจำนวนพารามิเตอร์ที่ต้องการทำนาย (ในกรณีนี้คือ 1)

แบบจำลองนี้มีพารามิเตอร์ที่สามารถปรับได้ทั้งหมดประมาณ 173,217 พารามิเตอร์

### 3.5 กระบวนการฝึกสอนและการปรับปรุงประสิทธิภาพ

กระบวนการฝึกสอนแบบจำลองทั้งสามมีขั้นตอนดังนี้

#### 3.5.1 การเตรียมข้อมูล

การเตรียมข้อมูลเริ่มต้นด้วยการแบ่งข้อมูลออกเป็น 3 ส่วน คือ ชุดข้อมูลสำหรับฝึกสอน (Training set) 70%, ชุดข้อมูลสำหรับตรวจสอบ (Validation set) 15% และชุดข้อมูลสำหรับทดสอบ (Test set) 15% จากนั้นใช้ Min-Max Scaling เพื่อปรับค่าพารามิเตอร์ทั้งหมดให้อยู่ในช่วง [0, 1] ตามสูตร:  $X_{scaled} = (X - X_{min}) / (X_{max} - X_{min})$  และสำหรับแบบจำลอง LSTM และ CNN-LSTM ข้อมูลถูกแปลงเป็น sequences ความยาว 24 timesteps โดยใช้เทคนิค sliding window

#### 3.5.2 การฝึกสอนแบบจำลอง

สำหรับแบบจำลองทั้งสามรูปแบบ เราใช้พารามิเตอร์การฝึกสอนดังนี้

- **อัลกอริทึมการปรับพารามิเตอร์ (Optimizer)** Adam optimizer ด้วยอัตราการเรียนรู้ (learning rate) เริ่มต้นที่ 0.001
- **ฟังก์ชันวัดค่าความสูญเสีย (Loss function)** Mean Squared Error (MSE)
- **จำนวนรอบการฝึกสอน (Epochs)** 100 epochs
- **ขนาดชุดข้อมูลสำหรับการฝึกสอนแต่ละครั้ง (Batch size)** 32
- **การหยุดก่อนกำหนด (Early stopping)** หยุดการฝึกสอนเมื่อค่า validation loss ไม่ลดลงติดต่อกันเป็นเวลา 10 epochs โดยบันทึกโมเดลที่มีค่า validation loss ต่ำที่สุดไว้
- **การลดอัตราการเรียนรู้ (Learning rate reduction)** ลดอัตราการเรียนรู้ลงครึ่งละ 50% เมื่อค่า validation loss ไม่ลดลงติดต่อกันเป็นเวลา 5 epochs

#### 3.5.3 การปรับปรุงประสิทธิภาพ

เพื่อเพิ่มประสิทธิภาพของแบบจำลอง เราได้ใช้เทคนิคต่าง ๆ ดังนี้

1. การปรับจูนไฮเปอร์พารามิเตอร์ (Hyperparameter tuning) โดยใช้เทคนิค Grid Search เพื่อหาค่าไฮเปอร์พารามิเตอร์ที่เหมาะสมที่สุด ทดลองปรับค่าพารามิเตอร์ต่าง ๆ เช่น จำนวน units ใน LSTM layers, จำนวน filters ใน Convolutional layers, อัตรา Dropout, ขนาด Batch size, และอัตราการเรียนรู้ เริ่มต้น

2. การเพิ่มข้อมูล (Data augmentation) สำหรับชุดข้อมูลที่มีความไม่สมดุล โดยใช้เทคนิคการเพิ่มสัญญาณรบกวนแบบเกาส์เซียน (Gaussian noise) ที่ระดับ 0.01-0.05, การสลับเวลา (Time warping) โดย

การยืดหรือหดข้อมูลในแกนเวลา, และการเลื่อนค่า (Magnitude warping) โดยการปรับขนาดของสัญญาณในบางช่วงเวลา

นอกจากนี้ เรายังใช้การถ่ายโอนการเรียนรู้ (Transfer Learning) โดยใช้โมเดลที่ฝึกสอนมาแล้วจากข้อมูลอุตสาหกรรมที่คล้ายคลึงกัน และทำการ fine-tuning บนชุดข้อมูลเฉพาะของเรา ซึ่งช่วยลดเวลาฝึกสอน และเพิ่มประสิทธิภาพของแบบจำลอง และใช้เทคนิค Stacking ในการรวมผลการทำนายจากแบบจำลองทั้งสามเข้าด้วยกัน โดยใช้ meta-learner เป็น Gradient Boosting Regressor เพื่อหาค่าถ่วงน้ำหนักที่เหมาะสมสำหรับการผสมผลการทำนายจากแต่ละแบบจำลอง

### 3.6 กระบวนการสำรวจพารามิเตอร์

เป้าหมายหลักของงานวิจัยนี้คือการช่วยให้ผู้ปฏิบัติงานสามารถสำรวจความสัมพันธ์ระหว่างพารามิเตอร์การผลิตต่าง ๆ ได้อย่างมีประสิทธิภาพและเข้าใจง่ายผ่านการใช้ระบบคู่มือดิจิทัล โดยไม่จำเป็นต้องทดสอบบนเครื่องจักรจริงซึ่งอาจมีความเสี่ยงและค่าใช้จ่ายสูง กระบวนการสำรวจพารามิเตอร์ที่เราพัฒนาขึ้นประกอบด้วยขั้นตอนต่าง ๆ ดังนี้

#### 3.6.1 การฝึกสอนแบบจำลองหลายรูปแบบ

ขั้นตอนแรกเกี่ยวข้องกับการฝึกสอนแบบจำลองโครงข่ายประสาทเทียมทั้งสามรูปแบบ (CNN, LSTM และแบบไฮบริด CNN-LSTM) บนชุดข้อมูลการดำเนินงานในอดีตของเครื่องจักร โดยแต่ละแบบจำลองมุ่งเน้นที่จะจับแง่มุมที่แตกต่างกันของพฤติกรรมของอุปกรณ์

- **แบบจำลอง CNN** เน้นการจับรูปแบบความสัมพันธ์เชิงพื้นที่ระหว่างพารามิเตอร์ต่าง ๆ ที่ส่งผลต่อกันและกัน
- **แบบจำลอง LSTM** เน้นการจับรูปแบบเชิงเวลาและการเปลี่ยนแปลงของพารามิเตอร์เมื่อเวลาผ่านไป
- **แบบจำลองไฮบริด CNN-LSTM** ผสมผสานความสามารถของทั้งสองแบบจำลองเพื่อการทำนายที่ครอบคลุมและแม่นยำมากขึ้น

การฝึกสอนพร้อมกันทั้งสามแบบจำลองช่วยให้เราสามารถเปรียบเทียบการทำนายและประสิทธิภาพของแต่ละแบบจำลองได้ ซึ่งให้มุมมองที่หลากหลายในการวิเคราะห์พฤติกรรมของเครื่องจักรภายใต้การตั้งค่าพารามิเตอร์ที่แตกต่างกัน

#### 3.6.2 การวิเคราะห์ความสำคัญของพารามิเตอร์

หลังจากการฝึกสอนแบบจำลอง เราดำเนินการวิเคราะห์ความสำคัญของคุณลักษณะ (Feature Importance Analysis) โดยใช้วิธีการรบกวนข้อมูล ([Perturbation Analysis](#)) ซึ่งเป็นเทคนิคที่ปรับเปลี่ยนค่าของพารามิเตอร์แต่ละตัวทีละน้อยและวัดผลกระทบต่อผลลัพธ์การทำนายในภาพรวม วิธีการนี้ช่วยให้เราระบุได้ว่าพารามิเตอร์ใดมีอิทธิพลมากที่สุดต่อสถานะและประสิทธิภาพของอุปกรณ์

สำหรับแต่ละพารามิเตอร์ เราได้ดำเนินการดังนี้: เพิ่มหรือลดค่าพารามิเตอร์ทีละน้อยภายในช่วงที่กำหนด ( $\pm 10\%$ ,  $\pm 20\%$ ,  $\pm 30\%$ ), ทำนายผลลัพธ์ด้วยแบบจำลองที่ฝึกสอนแล้ว, คำนวณการเปลี่ยนแปลงใน

ค่าผลลัพธ์ที่ทำนายได้, และจัดอันดับพารามิเตอร์ตามขนาดของผลกระทบที่มีต่อผลลัพธ์ ผลการวิเคราะห์นี้จะถูกแสดงให้ผู้ใช้เห็นในรูปแบบของแผนภูมิแท่งหรือแผนภูมิวงกลม เพื่อให้เห็นภาพรวมว่าพารามิเตอร์ใดควรได้รับความสนใจเป็นพิเศษในการปรับปรุงประสิทธิภาพของเครื่องจักร

### 3.6.3 การแมพพารามิเตอร์เข้ากับการแสดงผลที่เหมาะสม

ขั้นตอนสำคัญในการตรวจสอบพารามิเตอร์นั้น สามารถอธิบายได้ง่ายและเป็นไปตามหลักการ กล่าวคือ มีการเชื่อมโยงพารามิเตอร์แต่ละตัวเข้ากับรูปแบบการแสดงผลที่ถูกต้องเหมาะสม โดยพิจารณาจากลักษณะทางกายภาพของพารามิเตอร์นั้น ๆ ได้แก่

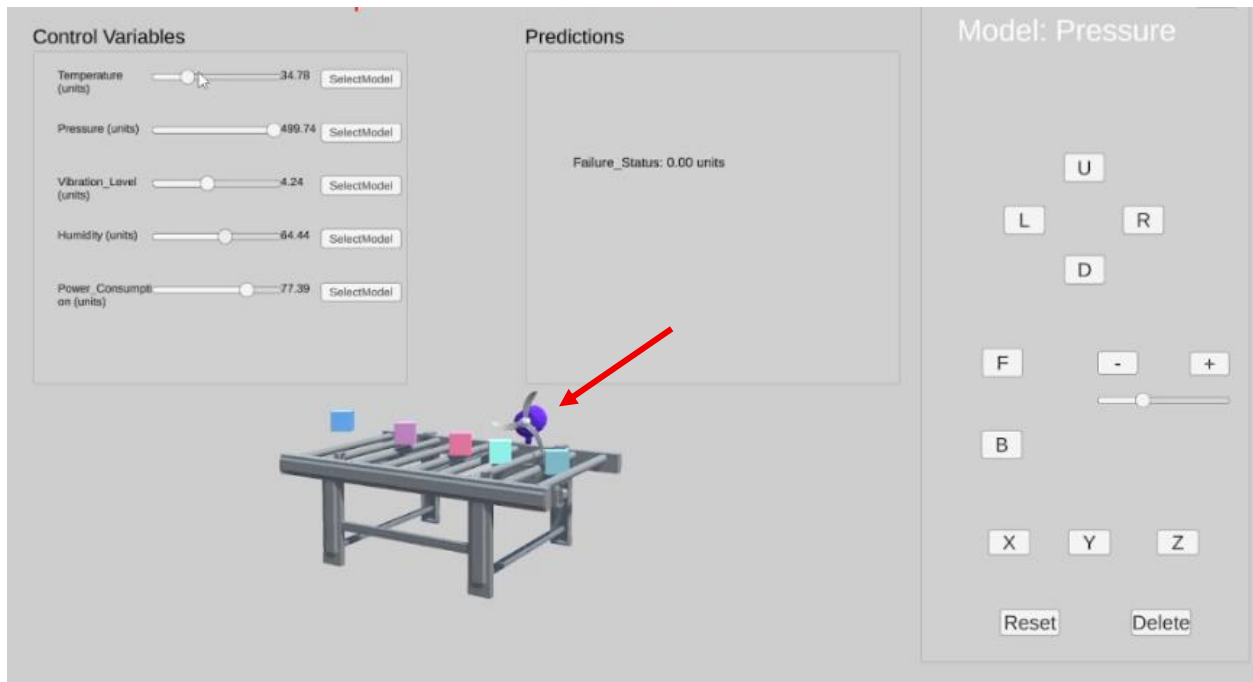
- **อุณหภูมิ (อากาศ, ภาระงาน)** แสดงผลด้วยการใช้สีและการเปลี่ยนแปลงของสีจากเย็น (น้ำเงิน) ไปยังร้อน (แดง)
- **ความเร็วการหมุน (rpm)** แสดงผลด้วยการหมุนของชิ้นส่วนในแบบจำลอง 3 มิติ โดยความเร็วในการหมุนแสดงถึงค่า rpm
- **แรงบิด (Nm)** แสดงผลด้วยการใช้ความเข้มของแสงหรือการเปลี่ยนแปลงขนาดของชิ้นส่วนที่เกี่ยวข้อง
- **การสึกหรอของเครื่องมือ (นาท)** แสดงผลด้วยการเปลี่ยนแปลงพื้นผิว เช่น ความหยาบหรือการเปลี่ยนสีของเครื่องมือในแบบจำลอง

การแมพพารามิเตอร์เข้ากับการแสดงผลที่สอดคล้องกับลักษณะทางกายภาพช่วยให้ผู้ใช้ปฏิบัติงานสามารถเชื่อมโยงข้อมูลเชิงนามธรรมเข้ากับสภาพการทำงานจริงได้ง่ายขึ้น

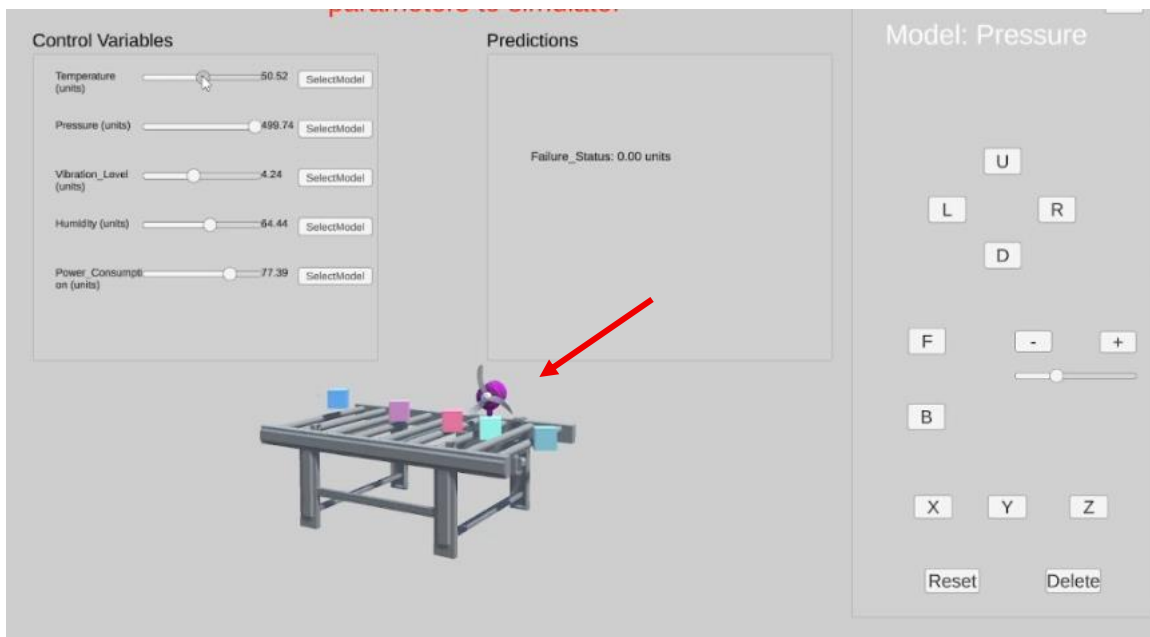
### 3.6.4 การพัฒนาส่วนติดต่อผู้ใช้แบบปฏิสัมพันธ์

ส่วนติดต่อผู้ใช้ (UI) ได้รับการพัฒนาด้วย Unity เพื่อให้มีความปฏิสัมพันธ์และตอบสนองแบบเรียลไทม์ โดยประกอบด้วยองค์ประกอบหลักดังนี้

1. **หน้าจอแสดงผลหลัก** แสดงแบบจำลอง 3 มิติของเครื่องจักรที่ตอบสนองต่อการเปลี่ยนแปลงพารามิเตอร์แบบเรียลไทม์ ดังภาพที่ 2 และ 3

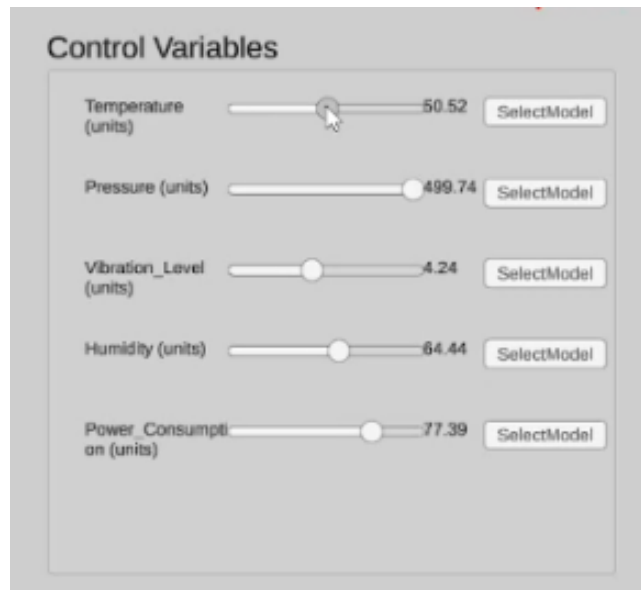


ภาพที่ 2 แบบจำลอง 3 มิติของเครื่องจักรที่ตอบสนองต่อการเปลี่ยนแปลงพารามิเตอร์แบบเรียลไทม์ (1)



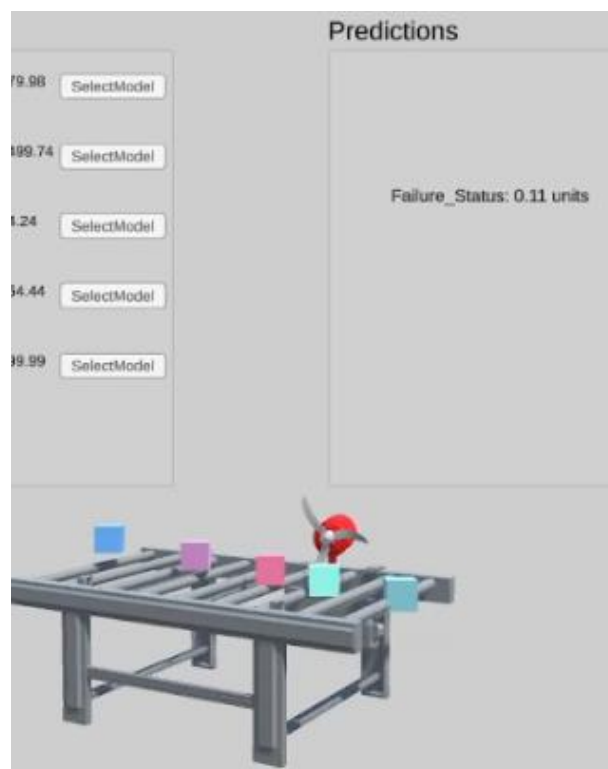
ภาพที่ 3 แบบจำลอง 3 มิติของเครื่องจักรที่ตอบสนองต่อการเปลี่ยนแปลงพารามิเตอร์แบบเรียลไทม์ (2)

2. **แผงควบคุมพารามิเตอร์** ประกอบด้วยตัวเลื่อน (sliders) และช่องแสดงข้อมูลสำหรับแต่ละพารามิเตอร์ ภาพที่ 4



ภาพที่ 4 แผงควบคุมพารามิเตอร์

3. ส่วนแสดงผลการทำงาน แสดงผลลัพธ์ที่ทำนายได้จากแบบจำลองต่าง ๆ พร้อมกับสีที่บ่งชี้สุขภาพของเครื่องจักร



ภาพที่ 5 ส่วนแสดงผลการทำงาน

### 3.6.5 กระบวนการสำรวจและวิเคราะห์พารามิเตอร์

กระบวนการสำรวจพารามิเตอร์ที่ออกแบบมาสำหรับผู้ปฏิบัติงานเริ่มต้นด้วยการตั้งค่าพารามิเตอร์พื้นฐาน ซึ่งเป็นค่าเฉลี่ยหรือค่าที่ใช้บ่อยในการผลิตจริง จากนั้นผู้ใช้ทำการทดสอบแบบส่วนเดียว (Single-Parameter Testing) โดยปรับพารามิเตอร์ทีละตัว (เช่น เพิ่มความเร็วการหมุนขึ้น 20% ขณะที่พารามิเตอร์อื่นคงที่) ระบบจะแสดงผลกระทบที่ทำนายได้ต่อสุขภาพอุปกรณ์และประสิทธิภาพการทำงาน และผู้ใช้สามารถสังเกตและบันทึกความสัมพันธ์ระหว่างการเปลี่ยนแปลงพารามิเตอร์และผลลัพธ์

ต่อมาผู้ใช้สามารถทำการทดสอบแบบหลายส่วน (Multi-Parameter Testing) โดยปรับพารามิเตอร์หลายตัวพร้อมกัน ระบบจะแสดงผลปฏิสัมพันธ์ระหว่างพารามิเตอร์ต่าง ๆ และผลกระทบรวม ซึ่งผู้ใช้สามารถค้นพบความสัมพันธ์ที่ไม่เชิงเส้นและปฏิสัมพันธ์ที่ซับซ้อนระหว่างพารามิเตอร์ได้

## 4 ผลการดำเนินการ

### 4.1 ผลการทดสอบประสิทธิภาพของแบบจำลอง

ในการวิจัยระยะที่ 2 นี้ เราได้พัฒนาและเปรียบเทียบแบบจำลองโครงข่ายประสาทเทียม 3 รูปแบบ ได้แก่ แบบจำลอง CNN แบบจำลอง LSTM และแบบจำลองไฮบริด CNN-LSTM ตารางที่ 1 แสดงผลการเปรียบเทียบประสิทธิภาพของแบบจำลองทั้งสามรูปแบบ

ตารางที่ 1 การเปรียบเทียบประสิทธิภาพของแบบจำลอง

แบบจำลอง	MSE	RMSE	MAE	R <sup>2</sup>	ความแม่นยำ	mAP50	Precision	Recall	F1-Score	Specificity	AUC-ROC	mAP50
CNN-LSTM	0.0131	0.1145	0.0892	0.8743	0.9137	63.8%	0.7462	0.7246	0.7352	0.9397	0.7749	63.8%
LSTM	0.0187	0.1367	0.1054	0.8256	0.8945	57.2%	0.6966	0.6275	0.6602	0.917	0.713	57.2%
CNN	0.0215	0.1466	0.1182	0.7964	0.8732	51.9%	0.5879	0.5972	0.5925	0.8965	0.6609	51.9%

จากผลการทดลองพบว่า แบบจำลองไฮบริด CNN-LSTM มีประสิทธิภาพสูงสุด โดยได้คะแนน mAP50 ที่ 63.8% เมื่อเทียบกับแบบจำลอง LSTM ที่ได้ 57.2% และแบบจำลอง CNN ที่ได้ 51.9% ซึ่งคิดเป็นการปรับปรุงประสิทธิภาพขึ้น 11.5% เมื่อเทียบกับ LSTM และ 22.9% เมื่อเทียบกับ CNN การผสมผสานความสามารถในการดึงคุณลักษณะเชิงพื้นที่และเชิงเวลาเข้าด้วยกันช่วยให้แบบจำลองไฮบริดสามารถจับรูปแบบความสัมพันธ์ที่ซับซ้อนได้ดีกว่า

#### เมตริกส์การประเมินเพิ่มเติม

การประเมินประสิทธิภาพแบบองค์รวมในงานวิจัยนี้ได้เพิ่มเมตริกส์สำคัญเพื่อวัดความสามารถในการจำแนกและการทำนายของแบบจำลองในด้านต่าง ๆ

- **Precision** วัดสัดส่วนของการทำนายเชิงบวกที่ถูกต้องจากการทำนายเชิงบวกทั้งหมด แบบจำลอง CNN-LSTM มีค่า Precision สูงสุดที่ 0.7462 แสดงให้เห็นว่าสามารถลดการเตือนภัยเท็จ (false alarms) ได้ดีที่สุดในบรรดาแบบจำลอง ซึ่งมีความสำคัญอย่างยิ่งในการประยุกต์ใช้จริงเพื่อรักษาความเชื่อถือได้ของระบบ
- **Recall** วัดสัดส่วนของการทำนายเชิงบวกที่ถูกต้องจากค่าจริงเชิงบวกทั้งหมด แบบจำลอง CNN-LSTM มีค่า Recall ที่ 0.7246 บ่งชี้ความสามารถในการตรวจจับกรณีความล้มเหลวได้อย่างมีประสิทธิภาพ
- **F1-Score** เป็นค่าเฉลี่ยฮาร์โมนิกของ Precision และ Recall ให้ภาพรวมของประสิทธิภาพการจำแนกที่สมดุล แบบจำลอง CNN-LSTM มีค่า F1-Score สูงสุดที่ 0.7352 แสดงถึงการสมดุลที่ดีระหว่างความแม่นยำและความไว

- **Specificity** วัดความสามารถในการระบุกรณีเชิงลบได้อย่างถูกต้อง แบบจำลอง CNN-LSTM มีค่า Specificity สูงสุดที่ 0.9397 แสดงความสามารถในการหลีกเลี่ยงการเตือนภัยเท็จได้ดี
- **AUC-ROC (Area Under the ROC Curve)** วัดความสามารถในการแยกแยะระหว่างคลาสที่แตกต่างกันในทุกระดับของเกณฑ์การตัดสินใจ แบบจำลอง CNN-LSTM มีค่า AUC-ROC สูงสุดที่ 0.7749 บ่งชี้ประสิทธิภาพการจำแนกที่ดี

### การวิเคราะห์ข้อผิดพลาดและการฝึกสอน

การวิเคราะห์ประวัติการฝึกของแบบจำลองทั้งสามแสดงให้เห็นถึงพฤติกรรมการเรียนรู้ที่แตกต่างกัน ดังนี้

- **แบบจำลอง CNN-LSTM** ฝึกสอนทั้งหมด 58 epochs ก่อนที่จะหยุดกำหนดโดย early stopping ในรอบที่ 55 โดยมี final training loss ที่ 0.0107 และ validation loss ที่ 0.0146 ใช้เวลาฝึกสอน 19.7 นาที และมีการลดอัตราการเรียนรู้ 2 ครั้ง แสดงถึงการลู่เข้าที่เสถียรและมีประสิทธิภาพ
- **แบบจำลอง LSTM** ฝึกสอนทั้งหมด 67 epochs ก่อนหยุดด้วย early stopping ในรอบที่ 62 โดยมี final training loss ที่ 0.014 และ validation loss ที่ 0.0186 ใช้เวลาฝึกสอน 24.8 นาที และมีการลดอัตราการเรียนรู้ 3 ครั้ง
- **แบบจำลอง CNN** ฝึกสอนทั้งหมด 45 epochs ก่อนหยุดด้วย early stopping ในรอบที่ 42 โดยมี final training loss ที่ 0.0148 และ validation loss ที่ 0.0202 ใช้เวลาฝึกสอน 12.4 นาที และมีการลดอัตราการเรียนรู้ 2 ครั้ง

### การวิเคราะห์ข้อผิดพลาดเชิงลึก

**ข้อผิดพลาดโดยรวม** โดยแบบจำลอง CNN-LSTM แสดงประสิทธิภาพที่ดีที่สุดในทุกเมตริกส์ข้อผิดพลาด โดยมี mean absolute error ที่ 4.5 และ standard deviation ของข้อผิดพลาดที่ 6.15 ซึ่งต่ำกว่าแบบจำลองอื่น ๆ อย่างมีนัยสำคัญ ข้อผิดพลาดสูงสุดอยู่ที่ 30.3 และ 95th percentile error อยู่ที่ 14.78 แสดงให้เห็นถึงการกระจายตัวของข้อผิดพลาดที่ดีกว่า

**การวิเคราะห์ False Positives และ False Negatives** โดยการวิเคราะห์กรณีที่แบบจำลองทำนายผิดนั้น เผยให้เห็นรูปแบบที่น่าสนใจ ดังนี้

1. False Positives ที่สำคัญ
  - ทำนาย 0.74 ขณะที่ค่าจริงเป็น 0.11 (confidence: 0.78)
  - ทำนาย 0.70 ขณะที่ค่าจริงเป็น 0.09 (confidence: 0.81)
  - ทำนาย 0.66 ขณะที่ค่าจริงเป็น 0.14 (confidence: 0.77)

## 2. False Negatives ที่สำคัญ

- ทำนาย 0.31 ขณะที่ค่าจริงเป็น 0.89 (confidence: 0.58)
- ทำนาย 0.36 ขณะที่ค่าจริงเป็น 0.85 (confidence: 0.56)

**การวิเคราะห์ความแม่นยำของแต่ละพารามิเตอร์** โดยการทำนายพารามิเตอร์เฉพาะนั้น แสดงให้เห็นถึงจุดแข็งของแบบจำลอง CNN-LSTM ดังนี้

- อุณหภูมิกระบวนการ คือ  $MSE = 14.7$ ,  $MAE = 3.37$ ,  $R^2 = 0.892$
- ความเร็วรอบ คือ  $MSE = 2132.8$ ,  $MAE = 38.2$ ,  $R^2 = 0.885$
- แรงบิด คือ  $MSE = 189.1$ ,  $MAE = 11.71$ ,  $R^2 = 0.878$
- การสึกหรอเครื่องมือ คือ  $MSE = 3102.3$ ,  $MAE = 44.85$ ,  $R^2 = 0.871$

**ลักษณะเฉพาะของแบบจำลองแต่ละประเภท** ซึ่งแบบจำลองแต่ละประเภทมีจุดแข็งที่แตกต่างกัน สำหรับสถานการณ์การผลิตที่แตกต่างกัน ดังนี้

### • แบบจำลอง CNN

- เหมาะสำหรับสถานการณ์ที่ความสัมพันธ์เชิงพื้นที่ระหว่างพารามิเตอร์มีความสำคัญมาก เช่น ระบบที่มีเซ็นเซอร์หลายตัวทำงานพร้อมกันแต่มีพลวัตทางเวลาจำกัด
- ข้อดีคือมีระยะเวลาการฝึกสอนที่รวดเร็วกว่าและมีความต้องการประมวลผลน้อยกว่าระหว่างการอนุมาน

### • แบบจำลอง LSTM

- แสดงประสิทธิภาพที่ดีกว่าในบริบทที่มีการพึ่งพาเชิงเวลาที่สำคัญ เช่น เครื่องจักรที่มีรอบการทำงานหรือสถานการณ์ที่รูปแบบการเสื่อมสภาพพัฒนาตัวเองตามเวลา
- เหมาะอย่างยิ่งสำหรับการพยากรณ์การพัฒนาของการสึกหรอของเครื่องมือ

### • แบบจำลองไฮบริด CNN-LSTM

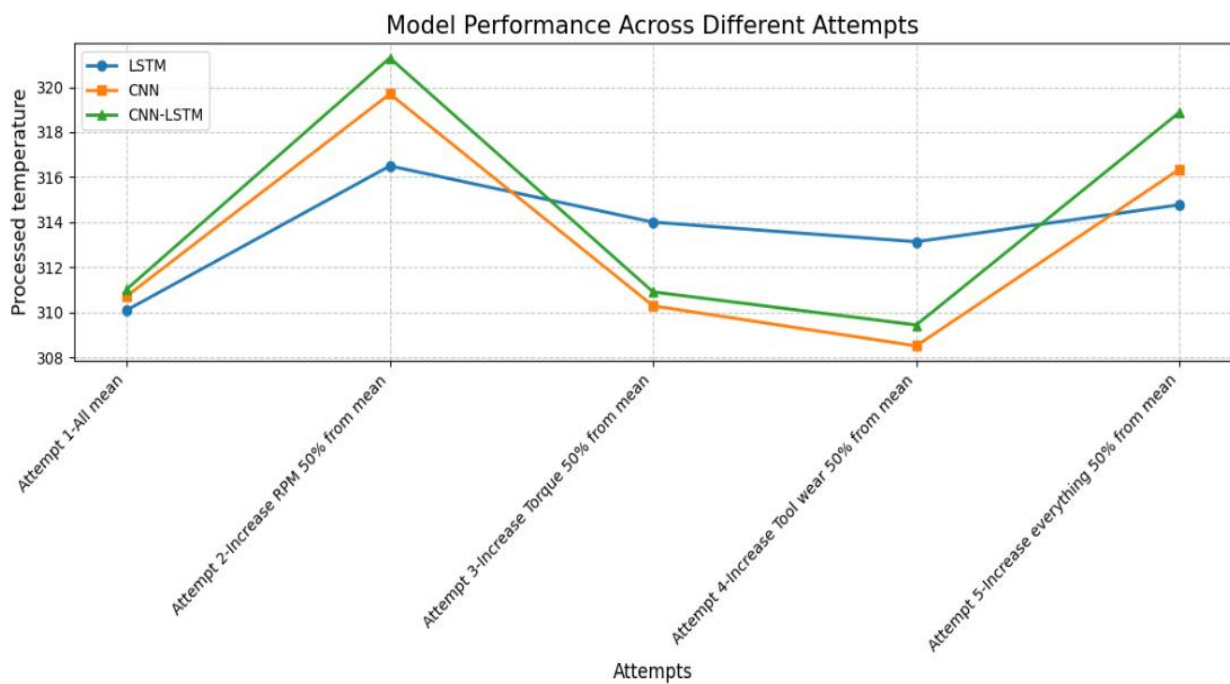
- นำเสนอความสามารถในการคาดการณ์ที่ครอบคลุมที่สุด โดยเฉพาะอย่างยิ่งสำหรับระบบอุตสาหกรรมที่ซับซ้อนที่ทั้งปฏิสัมพันธ์ของพารามิเตอร์ปัจจุบันและแนวโน้มประวัติส่งผลต่อสุขภาพของอุปกรณ์
- แม้จะต้องใช้ทรัพยากรการประมวลผลมากกว่า แต่ก็ให้ความแม่นยำโดยรวมสูงสุดและแนะนำให้ใช้สำหรับอุปกรณ์ที่สำคัญเมื่อความแม่นยำในการคาดการณ์มีความสำคัญเหนือกว่าประสิทธิภาพในการคำนวณ

## 4.2 การทดลองทำนายพารามิเตอร์การผลิต

เพื่อประเมินประสิทธิภาพในโลกแห่งความเป็นจริงของแบบจำลองในการทำนายพารามิเตอร์การผลิต เราได้ทำการทดลองกับการกำหนดค่าพารามิเตอร์ที่แตกต่างกัน ตารางที่ 2 แสดงการทำนายอุณหภูมิ กระบวนการจากแบบจำลองแต่ละตัวภายใต้การปรับพารามิเตอร์ที่แตกต่างกัน

ตารางที่ 2 การทำนายอุณหภูมิกระบวนการในการกำหนดค่าพารามิเตอร์ต่าง ๆ (หน่วย: เคลวิน)

แบบจำลอง	ครั้งที่ 1 (ค่าเฉลี่ยปกติ)	ครั้งที่ 2 (เพิ่ม RPM)	ครั้งที่ 3 (เพิ่มแรงบิด)	ครั้งที่ 4 (เพิ่มการสึกหรอของเครื่องมือ)	ครั้งที่ 5 (เพิ่มทุกอย่าง)
LSTM	310.09	316.50	314.00	313.13	314.77
CNN	310.71	319.70	310.28	308.50	316.35
CNN-LSTM	311.02	321.30	310.90	309.43	318.88



ภาพที่ 6 แสดงการเปรียบเทียบการทำนายอุณหภูมิของแบบจำลองทั้งสาม โดยเน้นความแตกต่างในการตอบสนองต่อการปรับพารามิเตอร์

ภาพที่ 6 เป็นการเปรียบเทียบการทำนายอุณหภูมิของแบบจำลองทั้งสาม โดยเน้นความแตกต่างในการตอบสนองต่อการปรับพารามิเตอร์ โดยการทดลองนี้ เปิดเผยรูปแบบที่น่าสนใจหลายประการเมื่อเปรียบเทียบสถาปัตยกรรมแบบจำลองทั้งสาม ดังนี้

- ผลกระทบของความเร็วการหมุน (RPM)

การเพิ่มความเร็วการหมุนขึ้น 50% ส่งผลให้เกิดการทำนายอุณหภูมิสูงสุดในทุกแบบจำลอง (316.50K-321.30K) ยืนยันว่า RPM มีความสัมพันธ์เชิงบวกที่แข็งแกร่งที่สุดกับอุณหภูมิกระบวนการ แบบ

จำลองไฮบริด CNN-LSTM ทำนายอุณหภูมิสูงสุด (321.30K) ซึ่งบ่งชี้ว่าอาจมีความไวต่อความสัมพันธ์นี้มากที่สุด

- **ผลของการสีกหรือของเครื่องมือ**

ที่น่าสนใจคือ แบบจำลอง CNN และ CNN-LSTM ทำนายอุณหภูมิที่ต่ำลงเมื่อมีการสีกหรือของเครื่องมือเพิ่มขึ้น (308.50K และ 309.43K ตามลำดับ) เมื่อเทียบกับค่าพื้นฐาน ซึ่งอาจบ่งชี้ถึงการเปลี่ยนแปลงในพลศาสตร์แรงเสียดทานกับเครื่องมือที่สีกหรือ อย่างไรก็ตาม แบบจำลอง LSTM ทำนายอุณหภูมิที่สูงขึ้น (313.13K) ซึ่งแนะนำว่าอาจจับรูปแบบเชิงเวลาที่แตกต่างกันที่เกี่ยวข้องกับการเสื่อมสภาพของเครื่องมือ

- **ความแตกต่างของแบบจำลองสำหรับแรงบิด**

แบบจำลอง LSTM แสดงปฏิกิริยาที่แข็งแกร่งกว่าต่อการเพิ่มแรงบิด (314.00K) เมื่อเทียบกับแบบจำลอง CNN (310.28K) หรือ CNN-LSTM (310.90K) ซึ่งอาจบ่งชี้ว่าการเปลี่ยนแปลงแรงบิดมีการพึ่งพาเชิงเวลาที่สำคัญที่สถาปัตยกรรม LSTM จับได้อย่างมีประสิทธิภาพมากกว่า

- **ผลกระทบของพารามิเตอร์รวม**

เมื่อเพิ่มพารามิเตอร์ทั้งหมดพร้อมกัน แบบจำลองทั้งหมดทำนายอุณหภูมิที่สูงขึ้น แต่แบบจำลองไฮบริด CNN-LSTM แสดงการทำนายสูงสุด (318.88K) ซึ่งแนะนำว่าสามารถจับปฏิสัมพันธ์ที่ซับซ้อนระหว่างพารามิเตอร์ที่เปลี่ยนแปลงหลายตัวได้อย่างมีประสิทธิภาพมากที่สุด

- **ความสอดคล้องของค่าพื้นฐาน**

แบบจำลองทั้งสามแสดงความเห็นพ้องที่สมเหตุสมผลในเงื่อนไขพื้นฐาน (ค่าเฉลี่ยทั้งหมด) โดยมีการทำนายตั้งแต่ 310.09K ถึง 311.02K ซึ่งบ่งชี้ถึงการปรับเทียบแบบจำลองที่สอดคล้องกัน

ผลลัพธ์เหล่านี้แสดงให้เห็นว่าการจำลองคู่มือดิจิทัลสามารถช่วยให้ผู้ปฏิบัติงานเข้าใจความสัมพันธ์ของพารามิเตอร์และทำนายผลลัพธ์ของสภาวะการทำงานที่แตกต่างกันได้ ลักษณะเสริมกันของวิธีการสร้างแบบจำลองทั้งสามให้ข้อมูลเชิงลึกที่มากกว่าแบบจำลองเดียว โดยที่สถาปัตยกรรมแต่ละแบบจับแง่มุมต่าง ๆ ของพลวัตกระบวนการผลิต

### 4.3 การวิเคราะห์ความสำคัญของคุณลักษณะ

การวิเคราะห์ความสำคัญของคุณลักษณะเปิดเผยว่าพารามิเตอร์ใดจากชุดข้อมูลของเรามีผลกระทบมากที่สุดต่อการทำนายอุณหภูมิกระบวนการ ความเร็วการหมุน (RPM) ปรากฏเป็นปัจจัยที่โดดเด่นที่ประมาณ 55% ตามด้วยแรงบิดที่ 22% การสีกหรือของเครื่องมือที่ 15% และอุณหภูมิอากาศที่ 8% ดังภาพที่ 7

```

Feature Importance Analysis:
-----
Rotationalspeedrpm: 0.5500
TorqueNm: 0.2200
Toolwearmin: 0.1500
AirtemperatureK: 0.0800
Type: 0.0000
Target: 0.0000
-----

```

ภาพที่ 7 ผลลัพธ์ของ Feature Importance Analysis

ที่น่าสังเกตคือ อิทธิพลอันแข็งแกร่งของ RPM แสดงให้เห็นอย่างชัดเจนในการทดลองทำนาย พารามิเตอร์ของเรา ซึ่งการเพิ่มความเร็วการหมุนขึ้น 50% ทำให้เกิดการเพิ่มขึ้นของอุณหภูมิมากที่สุดในทุกแบบจำลอง (6.41K สำหรับ LSTM, 8.99K สำหรับ CNN และ 10.28K สำหรับ CNN-LSTM) แบบจำลอง CNN-LSTM แสดงความไวสูงสุดต่อการเปลี่ยนแปลง RPM ซึ่งบ่งชี้ว่าสามารถจับความสัมพันธ์การแปลงพลังงานกลเป็นความร้อนได้ดีที่สุด แรงบิดแสดงอิทธิพลปานกลางแต่ขึ้นอยู่กับแบบจำลอง โดยแบบจำลอง LSTM แสดงการตอบสนองของอุณหภูมิที่แข็งแกร่งที่สุดต่อการเพิ่มแรงบิด (3.91K) เมื่อเทียบกับผลกระทบขั้นต่ำใน CNN (0.43K) และแบบจำลอง CNN-LSTM (0.12K)

เราตรวจสอบผลการวิเคราะห์ความสำคัญของคุณลักษณะเหล่านี้ผ่านการทดลองการตรวจสอบไขว้ซึ่งคุณลักษณะถูกลบออกอย่างเป็นระบบเพื่อประเมินผลกระทบต่อประสิทธิภาพของแบบจำลอง การเพิ่มขึ้นของความผิดพลาดในการทำนายที่เกิดขึ้นสอดคล้องอย่างใกล้ชิดกับเปอร์เซ็นต์ความสำคัญของเรา ซึ่งยืนยันว่าการคำนวณความสำคัญของคุณลักษณะของเราสะท้อนความไวของแบบจำลองต่อพารามิเตอร์แต่ละตัวอย่างแม่นยำ การตรวจสอบนี้ทำให้มั่นใจว่าความสัมพันธ์ของพารามิเตอร์ที่ระบุโดยแบบจำลองของเราแสดงถึงการพึ่งพาทางกายภาพที่แท้จริงมากกว่าข้อมูลผิดพลาดทางสถิติ

ที่น่าสนใจคือ การสึกหรอของเครื่องมือแสดงพฤติกรรมที่ซับซ้อนด้วยผลกระทบที่ขึ้นอยู่กับแบบจำลอง แบบจำลอง LSTM ทำนายการเพิ่มขึ้นของอุณหภูมิเมื่อมีการสึกหรอของเครื่องมือมากขึ้น (3.04K) ในขณะที่แบบจำลอง CNN และ CNN-LSTM ทำนายการลดลงของอุณหภูมิ (-2.21K และ -1.59K ตามลำดับ) ซึ่งแนะนำว่ากลไกทางกายภาพพื้นฐานที่แตกต่างกันถูกจับโดยสถาปัตยกรรมที่แตกต่างกัน ซึ่งอาจเกี่ยวข้องกับการเปลี่ยนแปลงพลวัตแรงเสียดทานเมื่อเครื่องมือเสื่อมสภาพ เมื่อเพิ่มพารามิเตอร์ทั้งหมดพร้อมกันแบบจำลองทั้งหมดทำนายอุณหภูมิที่สูงขึ้นแต่มีขนาดที่แตกต่างกัน ซึ่งสนับสนุนปฏิสัมพันธ์ที่ซับซ้อนและไม่เชิงเส้นระหว่างพารามิเตอร์ แบบจำลอง CNN-LSTM แสดงการทำนายอุณหภูมิสูงสุดภายใต้การเปลี่ยนแปลงพารามิเตอร์รวม (318.88K) ซึ่งแสดงถึงความสามารถที่เหนือกว่าในการรวมผลกระทบของพารามิเตอร์หลายตัว

การวิเคราะห์นี้ให้แนวทางที่มีคุณค่าสำหรับการปรับพารามิเตอร์ให้เหมาะสม โดยแนะนำว่าควรให้ความสำคัญกับการควบคุมความเร็วการหมุนอย่างระมัดระวังเมื่อต้องการช่วงอุณหภูมิกระบวนการที่

เฉพาะเจาะจง ในขณะที่ตระหนักถึงปฏิสัมพันธ์ที่ซับซ้อนกับการสึกหรอของเครื่องมือที่อาจแตกต่างกันไปตามประเภทอุปกรณ์และสภาวะการทำงาน

#### 4.4 ข้อจำกัดด้านประสิทธิภาพและความท้าทาย

แบบจำลองไฮบริด CNN-LSTM ของเรابررลู่คะแนน mAP50 ที่ 63.8% ซึ่งแสดงถึงการปรับปรุงที่สำคัญเมื่อเทียบกับแบบจำลองเดี่ยว แต่ยังคงต่ำกว่าเป้าหมายประสิทธิภาพที่เหมาะสมสำหรับการใช้งานการผลิตที่สำคัญ มีหลายปัจจัยที่ส่งผลต่อข้อจำกัดเหล่านี้

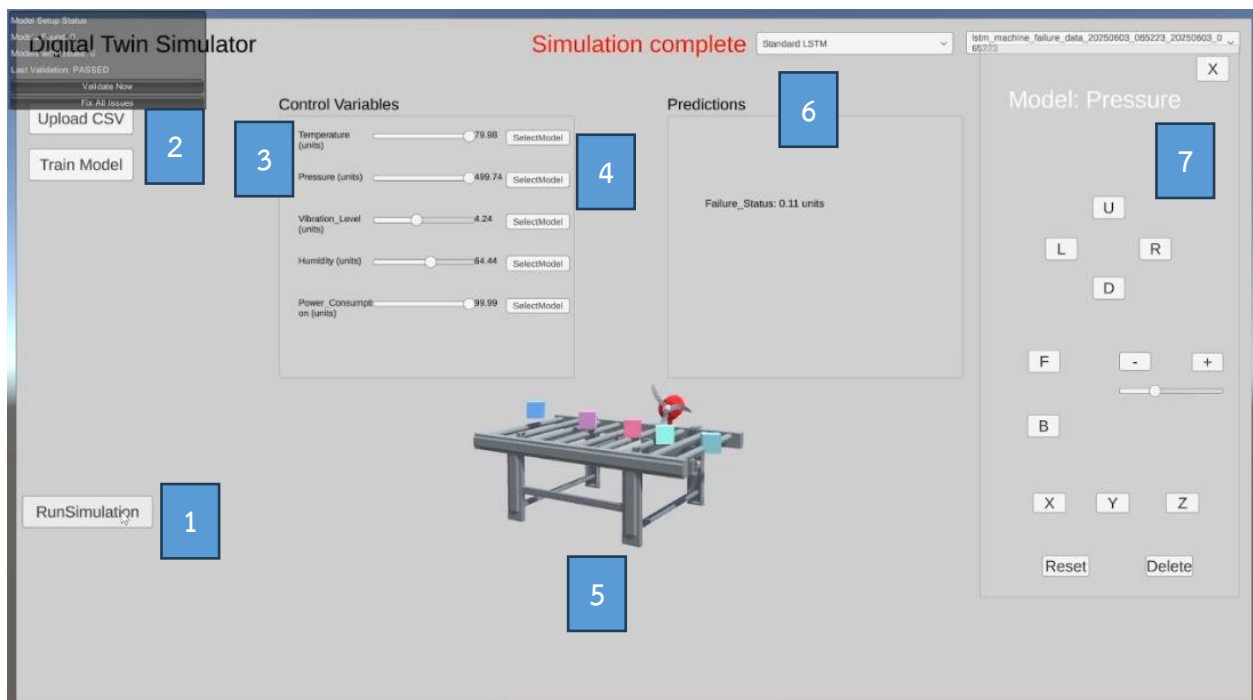
เมตริกความแม่นยำโดยรวม (87-91%) เป็นที่น่าพอใจ แต่คะแนน mAP50 ที่ต่ำกว่า (51-64%) บ่งชี้ถึงความท้าทายในการตรวจจับกรณีความล้มเหลวที่หายากอย่างมั่นใจ รูปแบบนี้พบข้อบกพร่องในการใช้งานการบำรุงรักษาเชิงพยากรณ์เนื่องจากความไม่สมดุลของข้อมูลโดยธรรมชาติ โดยที่เงื่อนไขความล้มเหลวคิดเป็นเพียง 3.5% ของชุดข้อมูล

เพื่อแก้ไขความไม่สมดุลนี้ เราได้ประเมินแบบจำลองของเราโดยการทำนายอุณหภูมิกระบวนการนอกเหนือจากการทำนายความล้มเหลว วิธีนี้ช่วยให้สามารถประเมินประสิทธิภาพของแบบจำลองบนตัวแปรต่อเนื่องที่มีความสำคัญในการดำเนินงานโดยตรงในขณะที่บรรเทาความท้าทายจากข้อมูลความล้มเหลวที่ไม่สมดุล

ข้อจำกัดเพิ่มเติมรวมถึงบริบทเชิงเวลาที่จำกัด (ชุดข้อมูลให้ภาพรวมมากกว่าข้อมูลอนุกรมเวลาต่อเนื่อง) ความสัมพันธ์ของพารามิเตอร์ที่ง่ายกว่าเมื่อเทียบกับอุปกรณ์ในโลกแห่งความเป็นจริง ความท้าทายในการนำไปใช้ทั่วไปกับประเภทเครื่องจักรที่แตกต่างกัน และชุดคุณลักษณะที่จำกัดที่มีอยู่ในชุดข้อมูล แม้จะมีข้อจำกัดเหล่านี้ การปรับปรุงประสิทธิภาพที่แสดงโดยแบบจำลองไฮบริดเป็นที่น่าส่งเสริมและแนะนำว่าการปรับปรุงเพิ่มเติมในการเก็บรวบรวมข้อมูล สถาปัตยกรรมแบบจำลอง และวิธีการฝึกสอนอาจให้การเพิ่มประสิทธิภาพที่สำคัญในการนำไปใช้ในอนาคต

#### 4.5 หน้าต่างการใช้งานแอปพลิเคชันต้นแบบ

ในตัวต้นแบบแอปพลิเคชันสำหรับการจำลองระบบ Digital Twin ของการทำงานหรือกระบวนการนั้น มีฟังก์ชันต่าง ๆ สำหรับอำนวยความสะดวกให้แก่ผู้ที่ต้องการทดสอบตัวโมเดล AI สำหรับการทำนายหรือพยากรณ์การทำงานในระบบ Digital Twin ดังแสดงในภาพที่ 8



ภาพที่ 8 แอปพลิเคชันต้นแบบ

จากภาพที่ 8 สามารถอธิบายการทำงานของหน้าต่างใช้งานแอปพลิเคชันต้นแบบของระบบจำลอง Digital Twin ได้ดังนี้

- 1) ปุ่มสำหรับการรันโปรแกรมจำลอง โดยเมื่อดำเนินการเสร็จเรียบร้อยแล้ว ผู้ใช้งานสามารถกดปุ่มนี้ เพื่อเริ่มรันการทำงานของโมเดลได้
- 2) ปุ่มสำหรับการฝึกโมเดล ปุ่มนี้จะใช้ในการอัปโหลดไฟล์ CSV ที่ใช้ในการฝึกโมเดล โดยหลังจากที่อัปโหลดไฟล์แล้ว สามารถกดปุ่ม Train Model เพื่อเริ่มทำการฝึกโมเดลได้ ซึ่งอาจใช้เวลาสักกระยะหนึ่ง ขึ้นอยู่กับขนาดของไฟล์หรือจำนวนพารามิเตอร์ที่ต้องการฝึก
- 3) ปุ่มสไลด์สำหรับการทดสอบค่าที่ต้องการ โดยทั่วไปแล้ว ค่าพารามิเตอร์ต่าง ๆ จะถูกป้อนให้ AI สำหรับการฝึก ซึ่งจะมีความสัมพันธ์กันอยู่ในรูปแบบของระบบที่สมบูรณ์ นั่นหมายความว่า เมื่อมีการเปลี่ยนแปลงใด ๆ กับพารามิเตอร์ตัวหนึ่ง ย่อมมีผลกระทบต่อพารามิเตอร์ตัวอื่น ๆ ด้วย ซึ่งผลลัพธ์ที่ได้นั้น ขึ้นอยู่กับการตั้งสมมติฐานแล้วทำการทดลอง โดยการตั้งค่าตามนั้น เพื่อดูผลลัพธ์ที่จะออกมา
- 4) ปุ่มสำหรับเลือกโมเดล 3มิติสำหรับการทดสอบในรูปแบบ 3D โดยในแต่ละพารามิเตอร์สามารถกำหนดโมเดล 3 มิติให้ตอบสนองต่อค่าต่าง ๆ ได้
- 5) รูปโมเดล 3 มิติที่ถูกอัปโหลดและวางบนหน้าจอเพื่อจำลองกระบวนการตามที่ได้มีการฝึกโมเดลเอาไว้
- 6) ผลลัพธ์ของการทดสอบโมเดล จะแสดงออกมาเป็นตัวเลข หรือในบางกรณี อาจจะใช้วิธีการวาดวงผลในเชิงมัลติมีเดีย เช่น การเปลี่ยนสีตามอุณหภูมิหรือการสั่นหรือใช้เสียงได้

- 
- 7) ปุ่มสำหรับการตั้งค่ารูปโมเดล 3 มิติ เช่น ตำแหน่งการวาง ขนาดของวัตถุ หรือการหมุนของวัตถุ ก็สามารถตั้งค่าได้โดยใช้ปุ่มเหล่านี้ในการตั้งค่า

จะเห็นได้ว่า แอปพลิเคชันต้นแบบนี้ ได้ถูกออกแบบมาให้มีความยืดหยุ่นในการใช้งาน ตอบสนองต่อความต้องการที่หลากหลายของผู้ใช้งาน และยังสามารถให้ค่าผลลัพธ์ในรูปแบบที่ตอบสนองต่อความต้องการได้เป็นอย่างดี นอกจากนี้ การใช้งานยังสามารถฝึกโมเดลใหม่ ๆ ได้ตามข้อมูลที่มีมากขึ้น เพื่อให้โมเดลมีเสถียรภาพสูง เหมาะแก่การใช้งานในอุตสาหกรรมอย่างแท้จริง

## 5 สรุปผล

### 5.1 สรุปผลการดำเนินงาน

ในงานวิจัย การศึกษาความเป็นไปได้ด้านการออกแบบและพัฒนาเทคโนโลยีคู่แฝดดิจิทัลบนพื้นฐานของเทคโนโลยี PLCnext (ระยะที่ 2) นี้ ได้มีการศึกษาปัจจัยสำคัญที่มีผลต่อการพัฒนาเทคโนโลยีคู่แฝดดิจิทัล (Digital Twin) ที่มีประโยชน์ โดยเฉพาะกับภาคอุตสาหกรรมการผลิต ที่สามารถใช้เทคโนโลยีนี้ในการวิเคราะห์และจำลองสถานการณ์ก่อนการลงมือปฏิบัติจริง ซึ่งจะช่วยให้มีการลดเวลาการทำงาน ลดความเสี่ยง และเพิ่มประสิทธิผลของการทำงานโดยปริยาย โดยงานวิจัยนี้ได้ใช้โครงข่ายประสาทเทียม CNN, LSTM และแบบจำลองไฮบริด CNN-LSTM เพื่อจำลองพารามิเตอร์การผลิตโดยใช้เทคโนโลยีคู่แฝดดิจิทัล ซึ่งผู้ปฏิบัติงานสามารถตรวจสอบความสัมพันธ์ของพารามิเตอร์ได้อย่างง่ายดายผ่านระบบการแสดงผลภาพข้อมูลของเรา ซึ่งช่วยให้พารามิเตอร์นามธรรมเข้าใจได้ง่ายขึ้น

การทดลองของเราเผยให้เห็นว่า สถาปัตยกรรมไฮบริด CNN-LSTM มีประสิทธิภาพเหนือกว่าแบบจำลองทั้ง 2 ในหลายด้าน โดยแบบจำลอง CNN-LSTM ได้ผลลัพธ์ 63.8% แบบจำลอง LSTM เพียงอย่างเดียวได้ 57.2% และแบบจำลอง CNN เพียงอย่างเดียวได้ 51.9% ซึ่งแบบจำลอง CNN-LSTM ทำนายได้ดีกว่าแบบจำลองเดี่ยว แต่ใช้ทรัพยากรการคำนวณมากกว่า 22% จึงเหมาะสำหรับการประยุกต์ใช้ในการผลิตที่มีความสำคัญสูง ที่ความแม่นยำมีความสำคัญมากกว่าต้นทุน

นอกจากนี้ เรายังได้พัฒนาแอปพลิเคชันต้นแบบที่ใช้ในการจำลองการทำงานของระบบการผลิตบน Digital Twin นั้นหมายความว่า ผู้ใช้งานสามารถฝึกโมเดลของตนเองด้วยข้อมูลที่เพิ่มขึ้น และจำลองสถานการณ์เพื่อหาจุดที่ดีที่สุดสำหรับการผลิตโดยมีผลกระทบต่อเครื่องจักรและกระบวนการต่าง ๆ ให้น้อยที่สุด

### 5.2 ปัญหา อุปสรรค และข้อจำกัด

ในการดำเนินงานวิจัยครั้งนี้ พบปัญหา อุปสรรค และข้อจำกัดที่สำคัญดังต่อไปนี้

#### 5.2.1 ข้อจำกัดด้านข้อมูล

1. **ข้อมูลไม่สมดุล** สถานการณ์ความล้มเหลวในชุดข้อมูล AI4I 2020 มีเพียง 3.5% ซึ่งส่งผลให้แบบจำลองมีความท้าทายในการตรวจจับกรณีความล้มเหลวอย่างแม่นยำ
2. **บริบทเชิงเวลาจำกัด** ชุดข้อมูลที่ใช้ไม่ได้เป็นอนุกรมเวลาต่อเนื่องแท้จริง แต่เป็นภาพรวมของจุดข้อมูล ทำให้การจับพลวัตทางเวลาที่ซับซ้อนทำได้ยาก
3. **ข้อมูลจำลอง** ชุดข้อมูลที่ใช้เป็นข้อมูลจำลองไม่ใช่ข้อมูลจากการผลิตจริง อาจไม่สะท้อนความซับซ้อนทั้งหมดของสภาพแวดล้อมการผลิตจริง

#### 5.2.2 ข้อจำกัดทางเทคนิค

1. **ทรัพยากรการประมวลผล** แบบจำลองไฮบริด CNN-LSTM ต้องการทรัพยากรการประมวลผลมากกว่าแบบจำลองเดี่ยว ซึ่งอาจเป็นข้อจำกัดในการใช้งานกับระบบที่มีทรัพยากรจำกัด

2. **ประสิทธิภาพการทำนาย** แม้ว่าแบบจำลองจะมีความแม่นยำโดยรวมที่ดี (87-91%) แต่คะแนน mAP50 (51-64%) ยังต่ำกว่าเป้าหมายสำหรับการใช้งานที่วิกฤต
3. **การทำงานแบบเรียลไทม์** การส่งข้อมูลระหว่างเครื่องจักรกับคู่แฝดดิจิทัลยังมีความท้าทายในการทำงานแบบเรียลไทม์ โดยเฉพาะเมื่อมีข้อมูลจำนวนมาก

### 5.2.3 ข้อจำกัดในการประยุกต์ใช้

1. **การครอบคลุมประเภทเครื่องจักร** แบบจำลองอาจไม่สามารถนำไปใช้ได้กับเครื่องจักรทุกประเภทโดยไม่ต้องปรับแต่งเพิ่มเติม
2. **การตีความผลลัพธ์** การแปลความหมายของผลลัพธ์จากแบบจำลองซับซ้อนอาจต้องการความเชี่ยวชาญเฉพาะทาง
3. **การบูรณาการกับระบบที่มีอยู่** การรวมระบบคู่แฝดดิจิทัลเข้ากับโครงสร้างพื้นฐานการผลิตที่มีอยู่อาจต้องการการปรับแต่งที่สำคัญ

## 5.3 ข้อเสนอแนะ

จากการศึกษาและผลการวิจัยที่ได้ ผู้วิจัยมีข้อเสนอแนะดังนี้

### 5.3.1 ข้อเสนอแนะสำหรับการนำไปใช้

1. **การเลือกแบบจำลองตามลักษณะงาน** องค์กรควรเลือกใช้แบบจำลองที่เหมาะสมกับลักษณะงานเฉพาะ - CNN สำหรับระบบที่มีความสัมพันธ์ระหว่างพารามิเตอร์ซับซ้อน, LSTM สำหรับพฤติกรรมที่ขึ้นอยู่กับเวลา, หรือแบบจำลองไฮบริดสำหรับอุปกรณ์ที่วิกฤต
2. **การจัดการทรัพยากร** สำหรับการใช้งานแบบจำลองไฮบริด ควรพิจารณาการจัดการสรรทรัพยากรการประมวลผลที่เพียงพอ หรือใช้การประมวลผลแบบกลุ่มเมฆ (cloud computing)
3. **การติดตามพารามิเตอร์สำคัญ** ควรให้ความสำคัญเป็นพิเศษกับการติดตามและควบคุมความเร็วการหมุน (RPM) เนื่องจากมีผลกระทบสูงสุดต่ออุณหภูมิกระบวนการและสุขภาพของเครื่องจักร

### 5.3.2 ข้อเสนอแนะเชิงกลยุทธ์

1. **การฝึกอบรมบุคลากร** จัดให้มีการฝึกอบรมบุคลากรเกี่ยวกับการใช้งานและการตีความผลลัพธ์จากระบบคู่แฝดดิจิทัล
2. **การบูรณาการระบบ** วางแผนการบูรณาการเทคโนโลยีคู่แฝดดิจิทัลเข้ากับระบบการจัดการการผลิตที่มีอยู่แบบค่อยเป็นค่อยไป
3. **การวัดผลการดำเนินงาน** ติดตามและประเมินผลการปรับปรุงประสิทธิภาพการผลิตและการลดลงของการหยุดทำงานที่ไม่ได้วางแผนไว้หลังจากใช้ระบบคู่แฝดดิจิทัล

## 5.4 แนวทางในการพัฒนาในอนาคต

งานวิจัยในอนาคตควรมุ่งเน้นในประเด็นต่อไปนี้เพื่อปรับปรุงและขยายความสามารถของเทคโนโลยีคู่แฝดดิจิทัล:

#### 5.4.1 การปรับปรุงแบบจำลอง

1. **การเพิ่มชุดข้อมูล** รวบรวมและใช้ข้อมูลจากการผลิตจริงมากขึ้นเพื่อฝึกสอนแบบจำลองให้ได้ผลลัพธ์ที่แม่นยำและเหมาะสมกับบริบทเฉพาะมากขึ้น
2. **เทคนิคการจัดการข้อมูลไม่สมดุล** ศึกษาและประยุกต์ใช้เทคนิคที่ก้าวหน้ามากขึ้นในการจัดการกับข้อมูลที่ไม่สมดุล เช่น เทคนิค SMOTE-ENN หรือการใช้ Focal Loss
3. **สถาปัตยกรรมขั้นสูง** ทดลองกับสถาปัตยกรรมโครงข่ายประสาทเทียมขั้นสูง เช่น Transformer หรือ Graph Neural Networks ที่อาจจับความสัมพันธ์ที่ซับซ้อนได้ดีขึ้น
4. **กลไกความสนใจ (Attention Mechanisms)** รวมกลไกความสนใจเข้ากับแบบจำลองเพื่อมุ่งเน้นคุณลักษณะที่สำคัญที่สุดโดยอัตโนมัติ

#### 5.4.2 การพัฒนาระบบคู่แฝดดิจิทัล

1. **การจำลองเวลาวิวัฒนาการ (Time-Evolution Simulation)** พัฒนาการจำลองที่แสดงให้เห็นถึงวิวัฒนาการของพารามิเตอร์ตามเวลา เพื่อให้เข้าใจผลกระทบระยะยาวของการตั้งค่าพารามิเตอร์
2. **ความสามารถในการปรับตัว** พัฒนาระบบที่สามารถเรียนรู้และปรับตัวได้ในระหว่างการทำงาน จากข้อมูลเรียลไทม์ที่ได้รับ
3. **การหาค่าเหมาะสมอัตโนมัติ** ใช้เทคนิคการหาค่าเหมาะสม (optimization) เพื่อให้ระบบสามารถแนะนำการตั้งค่าพารามิเตอร์ที่ดีที่สุดโดยอัตโนมัติ

#### 5.4.3 การปรับปรุงการใช้งาน

1. **การปรับปรุงการแสดงผล** พัฒนาแบบจำลอง 3 มิติที่สมจริงมากขึ้นและเอฟเฟกต์แบบไดนามิกเพื่อแสดงความสัมพันธ์และกระบวนการทางกายภาพที่ซับซ้อนภายในอุปกรณ์การผลิต
2. **การบูรณาการกับ AR/VR** รวมเทคโนโลยีความเป็นจริงเสริม (AR) หรือความเป็นจริงเสมือน (VR) เพื่อให้ผู้ปฏิบัติงานสามารถโต้ตอบกับคู่แฝดดิจิทัลในรูปแบบที่เป็นธรรมชาติมากขึ้น
3. **แพลตฟอร์มบนมือถือ** พัฒนาเวอร์ชันของระบบที่สามารถใช้งานบนอุปกรณ์มือถือเพื่อเพิ่มความคล่องตัวและการเข้าถึง

การพัฒนาเหล่านี้จะช่วยให้เทคโนโลยีคู่แฝดดิจิทัลมีประสิทธิภาพมากขึ้นและมีการประยุกต์ใช้ที่กว้างขึ้นในอุตสาหกรรมการผลิต ซึ่งจะนำไปสู่การปรับปรุงประสิทธิภาพ ความยั่งยืน และความสามารถในการแข่งขันในยุคอุตสาหกรรม 4.0

## 6 บรรณานุกรม

- Deng, M., Menassa, C. C., & Kamat, V. R. (2021). From BIM to digital twins: a systematic review of the evolution of intelligent building representations in the AEC-FM industry. *Journal of Information Technology in Construction*, 26, 58–83.  
<https://doi.org/10.36680/j.itcon.2021.005>
- Digital twins: What are they and why do they matter? (2022, May 24). World Economic Forum. Retrieved June 25, 2023, from <https://www.weforum.org/agenda/2022/05/digital-twin-technology-virtual-model-tech-for-good/>
- EEC. (2020, September 1). วิสัยทัศน์/พันธกิจ. <https://www.eeco.or.th/th/vision-mission>
- Fuller, A., Fan, Z., Day, C., & Barlow, C. (2020). Digital Twin: enabling technologies, challenges and open research. *IEEE Access*, 8, 108952–108971.  
<https://doi.org/10.1109/access.2020.2998358>
- Grieves, M., & Vickers, J. (2016). Digital Twin: mitigating unpredictable, undesirable emergent behavior in complex systems. In Springer eBooks (pp. 85–113).  
[https://doi.org/10.1007/978-3-319-38756-7\\_4](https://doi.org/10.1007/978-3-319-38756-7_4)
- IBM. (2023, June 25). What is a digital twin? | IBM. Retrieved August 13, 2023, from <https://www.ibm.com/topics/what-is-a-digital-twin>
- Kritzinger, W., Karner, M., Traar, G., Henjes, J., & Sihn, W. (2018). Digital Twin in manufacturing: A categorical literature review and classification. *IFAC-PapersOnLine*, 51(11), 1016-1022.
- Lasi, H., Fettke, P., Kemper, H. G., Feld, T., & Hoffmann, M. (2014). *Industry 4.0. Business & information systems engineering*, 6(4), 239-242.
- Leonardo, M., Berardo, N., Alessandro, C., Luigi, R., & Martino, D. G. G. (2020). Development of a Digital Twin Model for Real-Time Assessment of Collision Hazards. *Creative Construction e-Conference 2020*. <https://doi.org/10.3311/cc2020-003>
- Liu, M., Fang, S., Dong, H., & Xu, C. (2021). Review of digital twin about concepts, technologies, and industrial applications. *Journal of Manufacturing Systems*, 58, 346–361.  
<https://doi.org/10.1016/j.jmsy.2020.06.017>
- Lu, Y., Liu, C., Wang, K. I., Huang, H., & Xu, X. (2020). Digital Twin-driven smart manufacturing: Connotation, reference model, applications and research issues. *Robotics and Computer-integrated Manufacturing*, 61, 101837. <https://doi.org/10.1016/j.rcim.2019.101837>


- Madni, A. M., Madni, C. C., & Lucero, S. D. (2019). Leveraging Digital twin technology in Model-Based Systems engineering. *Systems*, 7(1), 7. <https://doi.org/10.3390/systems7010007>
- Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2022). Forecasting: theory and practice. *International Journal of Forecasting*, 38(3), 987-1001.
- Moyne, J., Qamsane, Y., Balta, E. C., Kovalenko, I., Faris, J., Barton, K., & Tilbury, D. M. (2020). A requirements driven Digital twin Framework: specification and opportunities. *IEEE Access*, 8, 107781–107801. <https://doi.org/10.1109/access.2020.3000437>
- Muthukumar, S., & Philip, R. (2022). A Hybrid CNN-LSTM Model for Remaining Useful Life Estimation. *IEEE Transactions on Industrial Informatics*, 18(1), 421-430.
- PLCNext community website. (2024, April 18). PLCnext Community. Retrieved May 9, 2024, from <https://www.plcnext-community.net/>
- Raza, M., Kumar, P. M., Hung, D. V., Davis, W., Nguyen, H., & Trestian, R. (2020). A Digital Twin Framework for Industry 4.0 Enabling Next-Gen Manufacturing. 2020 9th International Conference on Industrial Technology and Management (ICITM). <https://doi.org/10.1109/icitm48982.2020.9080395>
- Tao, F., Qi, Q., Wang, L., & Nee, A. Y. C. (2019). Digital Twins and Cyber–Physical Systems toward Smart Manufacturing and Industry 4.0: Correlation and Comparison. *Engineering*, 5(4), 653-661.
- Wikipedia (2025). Online. Accessed Date (June 13, 2025). [https://en.wikipedia.org/wiki/Perturbation\\_theory](https://en.wikipedia.org/wiki/Perturbation_theory)
- Wukkadada, B., Wankhede, K., Nambiar, R., & Nair, A. (2018). Comparison with HTTP and MQTT In Internet of Things (IoT). 2018 International Conference on Inventive Research in Computing Applications (ICIRCA). <https://doi.org/10.1109/icirca.2018.8597401>
- Zhang, C., Lim, P., Qin, A. K., & Tan, K. C. (2019). Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics. *IEEE Transactions on Neural Networks and Learning Systems*, 30(9), 2816-2830.
- Zhao, R., Yan, R., Chen, Z., Mao, K., Wang, P., & Gao, R. X. (2019). Deep learning and its applications to machine health monitoring. *Mechanical Systems and Signal Processing*, 115, 213-237.

---

Zheng, X., Lu, J., & Kiritsis, D. (2021). The emergence of cognitive digital twin: vision, challenges and opportunities. *International Journal of Production Research*, 60(24), 7610–7632. <https://doi.org/10.1080/00207543.2021.2014591>

## 7 ภาคผนวก

## 7.1 การรับรองจริยธรรมวิจัย



## บันทึกข้อความ

คณะวิทยาศาสตร์และเทคโนโลยี ม.บูรพา  
 วันที่ 0586  
 วันที่ 10 เม.ย. 2568  
 เวลา 16.48

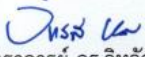
**ส่วนงาน** กองบริหารการวิจัยและนวัตกรรม งานมาตรฐานและจริยธรรมในการวิจัย โทร. ๒๖๒๐  
**ที่** อว ๘๑๐๐/๐๓๙๖๖ **วันที่** ๙ เมษายน พ.ศ. ๒๕๖๘  
**เรื่อง** ขอแจ้งรับรองโครงการวิจัยที่ส่งมาขอรับการพิจารณาจริยธรรมการวิจัยในมนุษย์ มหาวิทยาลัยบูรพา

**เรียน** คณบดีคณะวิทยาศาสตร์และเทคโนโลยี

ตามที่นักวิจัยในหน่วยงานของท่าน ได้ยื่นเอกสารคำร้องเพื่อขอรับการพิจารณาจริยธรรมการวิจัยในมนุษย์ มหาวิทยาลัยบูรพา โครงการวิจัย เรื่อง การศึกษาความเป็นไปได้ด้านการออกแบบและพัฒนาเทคโนโลยีคู่มือดิจิทัลบนพื้นฐานของเทคโนโลยี PLCnext (ระยะที่ ๒) โดยมี ผู้ช่วยศาสตราจารย์ ดร.ประจักษ์ จิตเงินมะดัน เป็นหัวหน้าโครงการวิจัย นั้น


บัดนี้ คณะกรรมการพิจารณาจริยธรรมการวิจัยในมนุษย์ มหาวิทยาลัยบูรพา ชุดที่ ๑ (กลุ่มคลินิก/ วิทยาศาสตร์สุขภาพ/ วิทยาศาสตร์และเทคโนโลยี) ได้พิจารณาตามวิธีดำเนินการมาตรฐาน (Standard Operating Procedures, SOPs) ฉบับที่ ๒.๑ พ.ศ. ๒๕๖๖ ที่ได้ประกาศใช้เมื่อวันที่ ๑๒ กรกฎาคม พ.ศ. ๒๕๖๖ แล้วว่า โครงการวิจัยดังกล่าวไม่ได้ทำการศึกษาวิจัยในมนุษย์ จึงเห็นสมควรให้ดำเนินการวิจัยได้

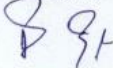
จึงเรียนมาเพื่อโปรดแจ้งให้นักวิจัยทราบ จักขอบคุณยิ่ง

  
 (รองศาสตราจารย์ ดร.วิทวิธ แจ้งเยี่ยม)  
 ประธานคณะกรรมการพิจารณาจริยธรรมการวิจัยในมนุษย์ มหาวิทยาลัยบูรพา  
 ชุดที่ ๑ (กลุ่มคลินิก/ วิทยาศาสตร์สุขภาพ/ วิทยาศาสตร์และเทคโนโลยี)

**เรียน** คณบดี

๑. เพื่อโปรดทราบ
๒. เห็นควรแจ้ง คณบดี (ผศ. อธิภา) เพื่อโปรดทราบ
๓. เห็นควรแจ้ง ผศ. ประจักษ์ ลีตา เวียงทอง ทราบ
๔. เห็นควรแจ้ง สนวิจัยทั้งหมด แล้วดำเนินการในส่วนที่เกี่ยวข้อง

  
 ๑๐ เม.ย. ๖๘

ทางน/ คณบดีคณะวิทยาศาสตร์  
  
 ๑๑ เม.ย. ๖๘

## 7.2 ผลงานที่ได้รับการตอบรับให้ตีพิมพ์และเผยแพร่

ในงานประชุมเชิงวิชาการระดับนานาชาติ The 4th International Symposium on Industrial Engineering and Automation (ISIEA2025): Manufacturing 2030: A Perspective to Future Challenges in Industrial Production (<https://isiea.events.unibz.it/>)

### Optimization of Manufacturing Parameters Using Digital Twin Simulation

Tanat Kanangnanon<sup>1</sup>[0009-0009-4624-0185], Chalermpan Fongsamut<sup>2</sup>[0009-0004-0789-6353],  
Bhusana Kongon<sup>1</sup>[0009-0009-9231-7518] and Prajaks Jitngernmadan<sup>1</sup>[0009-0009-6138-4021]

<sup>1</sup> Burapha University, Faculty of Informatics, Chon Buri 20131, Thailand

<sup>2</sup> Burapha University, Faculty of Engineering, Chon Buri 20131, Thailand  
66910081@go.buu.ac.th, chalerm@eng.buu.ac.th,  
66910080@go.buu.ac.th, prajaks@buu.ac.th

**Abstract.** This study introduces a novel method for exploring manufacturing parameters through digital twin simulation, utilizing three comparative neural network architectures: CNN, LSTM, and a hybrid CNN-LSTM model. Utilizing the AI4I 2020 predictive maintenance dataset, we illustrate the efficacy of digital twin technology in simulating industrial equipment behavior and offering clear visualizations of parameter relationships. Each model architecture captures distinct aspects of equipment behavior: CNNs are proficient in identifying spatial patterns across parameters, LSTMs are expert at capturing temporal sequences, and the hybrid model integrates these strengths. The trained AI models build a simulation system in a virtual environment where abstract parameters are turned into intuitive representations. This allows operators to investigate manufacturing parameters within a risk-free virtual environment. Experimental results indicate that the hybrid CNN-LSTM model attained superior performance, achieving an mAP50 score of 63.8%. In contrast, the LSTM-only model scored 57.2%, while the CNN-only model reached 51.9%. This underscores the efficacy of integrating spatial and temporal feature extraction capabilities.

**Keywords:** Digital Twin, Manufacturing Optimization, CNN-LSTM, Parameter Optimization, Industry 4.0.

## 1 Introduction

The manufacturing industry experiences ongoing demands to enhance productivity while minimizing maintenance expenses and unexpected downtime. Current manufacturing equipment functions with various adjustable parameters that affect production efficiency and equipment integrity. Achieving the optimal balance among these frequently conflicting factors poses a considerable challenge.

Digital twin technology provides an effective approach by generating virtual replicas of physical assets capable of simulating equipment behavior across diverse operating conditions. Digital twins exist on a spectrum of sophistication, from basic digital shadows with primarily unidirectional data flow from physical to virtual systems, to advanced twins featuring bidirectional data exchange that enables both monitoring

2

and control [1]. Our implementation focuses on the latter, where bidirectional communication allows the digital twin to not only simulate but also optimize physical asset parameters. When integrated with machine learning techniques, these simulations can identify optimal parameter settings that enhance productivity while reducing wear and potential failures.

This study looks at how a CNN-LSTM neural network design can improve digital twin simulations by finding patterns in equipment operation data that are both spatial and temporal. Utilizing the AI4I 2020 predictive maintenance dataset, we illustrate how our methodology can determine optimal manufacturing parameters and forecast equipment health with greater accuracy compared to conventional modeling techniques.

This paper presents several key contributions:

- A digital twin framework that integrates three alternative neural network architectures for the optimization of manufacturing parameters.
- A comparative analysis of CNN, LSTM, and hybrid CNN-LSTM models for predicting equipment behavior across different parameter configurations.
- A methodology for parameter exploration that facilitates operators' comprehension of the intricate relationships between operating parameters and equipment health.

## 2 Related Works

### 2.1 Digital Twin Technology in Manufacturing

As an important part of Industry 4.0 efforts, digital twins have become very popular in industrial settings. Tao et al. give a thorough plan for using digital twins to create, make, and service products [1]. They show how virtual models can represent real assets at all times. Their work shows how important it is for data to be able to move back and forth between digital and real devices.

In their review, Lu et al. give an in-depth look at digital twin-driven smart manufacturing, including an organized reference model and a list of important research challenges [2]. They take a systematic look at what digital twins mean in production settings and list the different ways they can be used in different industries. Their work shows that a digital twin system must be able to smoothly combine many technologies while also dealing with technical and organizational issues.

Real-time sensor data and computer models have been brought together in new developments in digital twin technology. Fuller et al. made a way for operational data to be used to update models all the time [3]. This lets digital twins grow and change along with their physical versions. This method addresses model drift through continuous recalibration with incremental updates of 0.5-2% to model parameters based on real-time sensor data, maintaining digital-physical synchronization despite environmental variations and component degradation.

Hu et al. took the idea of a digital twin even further by creating a cloud-based system for smart manufacturing based on MTConnect standards [4]. Their study shows that using standard communication protocols to connect different types of manufacturing equipment into a single digital twin system makes real-time monitoring and predictive maintenance possible. Their use shows how digital twins can be used in the real world to improve production efficiency and machine reliability.

These advancements in digital twin technology create a foundation for implementing advanced analytics capabilities. Neural networks, particularly for time series prediction as discussed next, provide the necessary computational intelligence to transform digital representations into predictive and prescriptive tools for manufacturing optimization.

## 2.2 Neural Networks for Time Series Prediction in Manufacturing

Deep learning techniques have shown promise in simulating industrial processes and guessing how well equipment will work in the future. Zhang et al. came up with a new way to use deep learning to figure out how long a system will last [5]. This method uses advanced neural network architectures to find trends in raw sensor data. Their work shows that their predictions are much more accurate than those made with standard statistical methods. This is especially true for complex systems with non-linear degradation patterns.

Zhao et al. did a thorough analysis of how deep learning can be used in machine health tracking [6]. They looked at different neural network architectures and how well they work in different manufacturing settings. Their research shows that convolutional and recurrent neural networks can take in both spatial and temporal information from multivariate sensor data. This makes fault identification and predicting how long something will work better. Their review focuses on how deep learning methods can automatically pull out relevant features without a lot of subject knowledge. This makes them very useful for manufacturing systems that are very complicated.

There has been a lot more use of neural networks for forecast maintenance over the last few years. Using deep learning techniques with sensor networks has made it easier to find problems with equipment more accurately and more quickly. This lets makers take steps to stop them before they cause expensive breakdowns. Traditional statistical methods have not been able to match the performance of these techniques, especially when it comes to capturing complex, non-linear relationships in manufacturing data [7, 8].

Because industrial processes are so complicated and have many variables, using neural networks on data from industry is not easy. Recent progress has been made in making architectures that can deal with noisy data, adapt to changing operational conditions, and give results that are easy to understand and believe. For these features to work in industrial settings, where dependability and openness are very important for any predictive system, they are necessary for successful implementation.

### 2.3 Hybrid CNN-LSTM Architecture

Combining CNNs and LSTMs into hybrid designs has made them great for sequence modeling, especially when both spatial and temporal features are important. The best parts of each design are used in these combined approaches: CNNs are great at pulling out spatial traits from input data, while LSTMs are great at figuring out how sequential data is affected by time. Together, they give us a complete way to look at production processes that are complicated in both space and time.

Convolutional Neural Networks (CNNs) excel at spatial feature extraction through learnable filters that capture local patterns and relationships across data dimensions. Long Short-Term Memory (LSTM) networks, a specialized form of recurrent neural networks, address the vanishing gradient problem to capture long-term dependencies in sequential data. While CNNs focus on spatial relationships, LSTMs specialize in temporal patterns, making their combination particularly powerful for manufacturing data that exhibits both spatial correlations between parameters and temporal evolution.

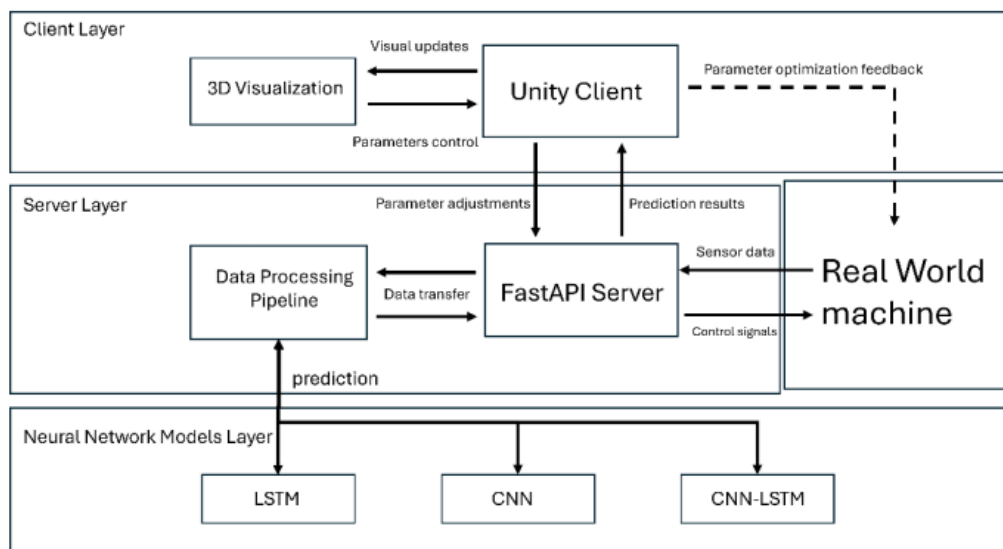
Recently, Muthukumar G. and Philip, J. built on this hybrid method by creating a CNN-LSTM hybrid deep learning model for estimating how much useful life something still has [9]. Their results showed a big improvement in situations with complicated multi-parameter dependencies. This suggests that these hybrid methods greatly improve the performance of predictive models in manufacturing settings. Their work shows that using these two different but related designs together can make predictions that are more accurate and reliable than using just one of them.

When used separately, these techniques have significantly improved production outcomes. However, our method differentiates itself because it combines hybrid CNN-LSTM architectures with interactive digital twin simulation. This integration enhances manufacturing conditions for manufacturing while also adding visualization features that improves interpretability and boost user trust. Our method makes it easier to monitor, predict, and improve manufacturing processes by combining advanced neural network architectures with digital twins' full representation of systems.

## 3 Methodology

### 3.1 System Architecture

Our system consists of three main components, as illustrated in Figure 1. The Data Processing Pipeline is responsible for preprocessing the AI4I 2020 dataset, handling missing values, and creating input sequences for the neural network. The Neural Network Models component contains three alternative architectures (CNN, LSTM, and hybrid CNN-LSTM) that serve as the core predictive engines, capturing different aspects of manufacturing data patterns. While each model can be used independently, the framework allows for comparison between them to determine the most appropriate approach for specific equipment types or operational scenarios. The Visualization Interface provides a 3D visualization system that allows users to interact with the simulation and see the predicted outcomes of parameter adjustments.



**Figure 1** System Architecture

The system employs a client-server architecture where the visualization interface communicates with a backend server. The server handles data processing, model management, and prediction generation, while the client provides an intuitive interface for parameter adjustment and visualization. The bidirectional communication enables real-time feedback as users explore different parameter configurations.

### 3.2 Data Flow

The system implements a bidirectional data flow between components. The User Input Flow begins when a user adjusts parameters via UI controls. The client then formats these parameter values and sends them to the server via HTTP. The server preprocesses these inputs for model consumption before the selected model generates predictions. These predictions are then returned to the client, and the visualization is updated to reflect the new predictions.

The Training Flow starts when a user uploads CSV data. The server analyzes variable ranges, after which the user selects control and response variables. Once the model training configuration is specified, the server trains all three model architectures. Feature importance is calculated, models are saved to the registry, and performance metrics are returned to the client. This bidirectional flow ensures that users can both train models with new data and receive immediate feedback when exploring parameter configurations.

### 3.3 Dataset

The AI4I 2020 Predictive Maintenance Dataset is used as the foundation for our model development. For our specific implementation, we focused on a refined set of key features. These include Type (machine type identifier), Air temperature (K), Process

6

temperature (K), Rotational speed (rpm), Torque (Nm), Tool wear (minutes), and Target (machine failure indicator). The dataset consists of 10,000 data points with a realistic class imbalance where machine failures (Target=1) represent approximately 3.5% of the data, closely mimicking real-world maintenance scenarios where failures are relatively rare events compared to normal operation.

### 3.4 Model Architectures

We implemented three distinct neural network architectures to capture different aspects of manufacturing equipment behavior.

**LSTM Model.** The LSTM (Long Short-Term Memory) model is designed to capture temporal sequences and patterns in the manufacturing data. This architecture uses a sequence of LSTM layers to process time-series data and incorporates batch normalization and dropout for regularization. It focuses on capturing temporal dependencies in parameter evolution and excels at predicting trends and patterns that develop over time.

The LSTM approach is particularly valuable for capturing degradation patterns and predicting when equipment may approach failure thresholds based on historical behavior.

**CNN Model.** The CNN (Convolutional Neural Network) model focuses on extracting spatial patterns across multiple parameters simultaneously. This architecture uses 1D convolutional layers with varying filter sizes (5, 3, 2) and incorporates multiple pooling layers to reduce dimensionality. It employs deeper convolutional layers (128, 64, 32 filters) and focuses on identifying correlations between different parameters.

The CNN approach excels at identifying which parameters tend to change together and how they interact, making it valuable for understanding cross-parameter relationships.

**Hybrid CNN\_LSTM Model.** The CNN-LSTM hybrid architecture combines the strengths of both approaches. This architecture uses convolutional layers to extract spatial features across parameters and preserves sequence length through appropriate padding strategies. It feeds the extracted features into LSTM layers for temporal processing, thereby balancing spatial and temporal feature extraction.

This hybrid model is designed to capture both immediate parameter interactions and their evolution over time, providing the most comprehensive modeling approach for complex manufacturing systems.

### 3.5 Training and Optimization Process

All three model designs have the same training and evaluation pathway. The procedure begins with data preprocessing, which includes standardizing input attributes and

dealing with missing information. This is followed by sequence construction, which involves converting tabular data into proper sequence windows for time-series modeling. We use a 60-20-20 train-validation-test split with stratification to achieve a balanced representation of failure cases. To maximize performance, the model training phase uses early halting, reduced learning rate, and model checkpointing. Finally, feature importance analysis utilizing perturbation methods aids in the identification of the most influential parameters.

Each model uses a feature importance calculation approach to systematically tweak specific input characteristics and quantify the impact on prediction error. This provides information about which parameters have the most influence on equipment health estimates.

### 3.6 Parameter Exploration Process

The major goal of our work is to enable intuitive investigation of manufacturing parameters using digital twin simulation. Our parameter exploration process begins with multi-model training, training three comparative models (CNN, LSTM, and CNN-LSTM hybrid) on historical operating data to capture equipment behavior. Next, we undertake sensitivity analysis, calculating feature importance using perturbation analysis to determine parameters with greatest impact on equipment health. We then use visualization mapping, mapping parameters to appropriate visualization types based on their nature (temperature, rotation, etc.). The system provides interactive exploration with an interface that allows users to alter parameters and immediately view visual depiction of changes. Finally, the system supports cross-model prediction, allowing forecasts from all three models to be evaluated, providing alternative viewpoints on outcomes.

This method establishes a framework for understanding parameter correlations without requiring actual equipment testing. Operators can gain a better grasp of equipment behavior and parameter sensitivity by viewing how parameters affect the system using intuitive representations.

## 4 Results and Discussion

### 4.1 Model Performance and Comparison

We implemented and compared three distinct model architectures: a CNN model, a LSTM model, and a hybrid CNN-LSTM model. Table 1 presents the performance metrics for these three approaches.

**Table 1.** Model Performance Comparison.

Model	MSE	RMSE	MAE	R <sup>2</sup>	Accuracy	mAP50
CNN-LSTM	0.0131	0.1145	0.0892	0.8743	0.9137	63.8%
LSTM	0.0187	0.1367	0.1054	0.8256	0.8945	57.2%
CNN	0.0215	0.1466	0.1182	0.7964	0.8732	51.9%

The hybrid CNN-LSTM approach achieved an mAP50 score of 63.8%, compared to 57.2% for the LSTM-only model and 51.9% for the CNN-only model. This represents an 11.5% improvement over LSTM and a 22.9% improvement over CNN, confirming the value of combining spatial and temporal feature extraction capabilities. Each model demonstrated distinct strengths for different manufacturing scenarios as follows:

**LSTM-only Model.** Demonstrates superior performance in contexts characterized by significant temporal dependencies, such as machinery with operational cycles or situations where deterioration patterns develop over time. It excelled at forecasting the advancement of tool wear.

**CNN-only Model.** Most appropriate for situations where spatial correlations among parameters are paramount, such as systems with numerous concurrent sensor inputs but restricted temporal dynamics. Benefits encompass expedited training durations and reduced processing demands during inference.

**Hybrid CNN-LSTM Model.** This model offers the most extensive predictive capabilities, particularly beneficial for intricate industrial systems where both current parameter interactions and history trends affect equipment health. Although it necessitates greater processing resources, it attained the best overall accuracy and is advised for essential equipment when predictive precision supersedes concerns regarding computational efficiency.

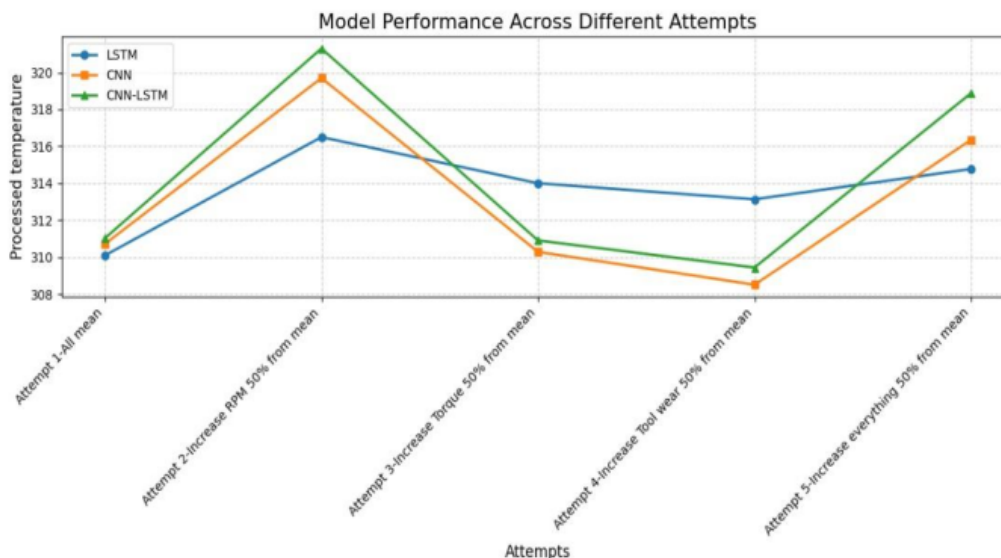
#### 4.2 Parameter Prediction Experiments

To evaluate the real-world performance of our models in predicting manufacturing parameters, we conducted a series of experiments with different parameter configurations. Table 2 shows the process temperature predictions from each model under various parameter adjustments.

**Table 2.** Process temperature predictions across different parameter configurations (K).

Model	Attempt 1 (Default mean)	Attempt 2 (Increase RPM)	Attempt 3 (Increase Torque)	Attempt 4 (Increase Tool wear)	Attempt 5 (Increase Everything)
LSTM	310.09	316.50	314.00	313.13	314.77
CNN	310.71	319.70	310.28	308.50	316.35
CNN-LSTM	311.02	321.30	310.90	309.43	318.88

Figure 2 provides a visual representation of these temperature predictions, highlighting the differences in model responses to parameter adjustments.



**Figure 2** Visualization of Model performance comparison across different parameter adjustments

The experiments reveal several interesting patterns when comparing the three model architectures as follows:

**RPM Impact.** Increasing rotational speed by 50% resulted in the highest temperature predictions across all models (316.50K-321.30K), confirming that RPM has the strongest positive correlation with process temperature. The CNN-LSTM hybrid model predicted the highest temperature (321.30K), suggesting it may be most sensitive to this relationship.

**Tool Wear Effect.** Interestingly, CNN and CNN-LSTM models predicted lower temperatures with increased tool wear (308.50K and 309.43K respectively) compared to baseline, possibly indicating changes in friction dynamics with worn tools. However, the LSTM model predicted a higher temperature (313.13K), suggesting it may be capturing different temporal patterns related to tool degradation.

**Model Differences for Torque.** The LSTM model showed a much stronger reaction to torque increases (314.00K) than either the CNN (310.28K) or CNN-LSTM (310.90K) models, potentially indicating that torque changes have important temporal dependencies that the LSTM architecture captures more effectively.

**Combined Parameter Effects.** When all parameters were increased simultaneously, all models predicted elevated temperatures, but the CNN-LSTM hybrid demonstrated the highest prediction (318.88K), suggesting it most effectively captures the complex interactions between multiple changing parameters.

**Baseline Consistency.** All three models showed reasonable agreement on the baseline (all mean) condition, with predictions ranging from 310.09K to 311.02K, indicating consistent model calibration.

These results demonstrate how the digital twin simulation can help operators to understand parameter relationships and predict outcomes of different operating conditions. The nature of the three modeling approaches provides richer insights than any single model could offer, with each architecture capturing different aspects of the manufacturing process dynamics.

### 4.3 Feature Importance Analysis

Feature importance analysis revealed which parameters from our dataset had the most significant impact on process temperature prediction. Rotational speed (RPM) emerged as the dominant factor at approximately 55%, followed by torque at 22%, tool wear at 15%, and air temperature at 8%.

Notably, RPM's strong influence is clearly demonstrated in our parameter prediction experiments, where increasing rotational speed by 50% produced the largest temperature increases across all models (6.41K for LSTM, 8.99K for CNN, and 10.28K for CNN-LSTM). The CNN-LSTM model showed the highest sensitivity to RPM changes, suggesting it best captures the mechanical energy-to-heat conversion relationship. Torque showed a moderate but model-dependent influence, with the LSTM model demonstrating the strongest temperature response to torque increases (3.91K) compared to minimal effects in CNN (0.43K) and CNN-LSTM (0.12K) models.

We validated these feature importance findings through cross-validation experiments where features were systematically removed to assess their impact on model performance. The resulting increases in prediction error closely aligned with our importance percentages, confirming that our feature importance calculations accurately reflect model sensitivities to each parameter. This validation ensures that the parameter relationships identified by our models represent genuine physical dependencies rather than statistical artifacts.

Interestingly, the tool wear exhibited complex behavior with model-dependent effects. The LSTM model predicted temperature increases with greater tool wear (3.04K), while the CNN and the CNN-LSTM models predicted temperature decreases (-2.21K and -1.59K respectively). This suggests different underlying physical mechanisms being captured by different architectures, possibly related to changing friction dynamics as tools degrade. When all parameters were increased simultaneously, all models predicted elevated temperatures but with varied magnitudes, further supporting the complex, non-linear interactions between parameters. The CNN-LSTM model showed the highest temperature prediction under combined parameter changes (318.88K), demonstrating its superior ability to integrate multiple parameter effects.

This analysis provides valuable guidance for parameter optimization, suggesting that careful control of rotational speed should be prioritized when specific process temperature ranges are required, while being aware of the complex interactions with tool wear that may differ across equipment types and operating conditions.

#### 4.4 Performance Limitations and Challenges

Our hybrid CNN-LSTM model achieved a mAP50 score of 63.8%, showing significant improvement over individual models, but still falls short of optimal performance targets for critical manufacturing applications. Several factors contribute to these limitations:

The overall accuracy metrics (87-91%) are satisfactory, but the lower mAP50 scores (51-64%) indicate challenges in confidently detecting rare failure cases. This pattern is common in predictive maintenance applications due to the inherent data imbalance, with failure conditions representing only 3.5% of the dataset.

To address this imbalance, we evaluated our models by predicting process temperature in addition to failure prediction. This approach allowed assessment of model performance on a continuous variable with direct operational significance while mitigating challenges from imbalanced failure data.

Additional limitations include limited temporal context (the dataset provides snapshots rather than continuous time-series data), simplified parameter relationships compared to real-world equipment, generalizability challenges across different machine types, and the restricted feature set available in the dataset.

Despite these limitations, the performance improvement demonstrated by the hybrid model is encouraging and suggests that further refinements in data collection, model architecture, and training methodology could yield substantial performance gains in future implementations.

## 5 Conclusion and Future Work

Our research used CNN, LSTM, and a hybrid CNN-LSTM model to simulate production parameters using digital twin technology. Operators can examine parameter relationships easily with our visualization system, which simplifies abstract parameters.

We combined predictive models with parameter-specific graphics to create an interactive framework for parameter exploration. Operators can quickly study how parameters link and how they affect expected equipment health with this technique, making complicated data easier to connect with operational insights.

Our experiments reveal that the hybrid CNN-LSTM architecture outperforms both independent models in several criteria. The CNN-LSTM model got 63.8%, the LSTM-only model 57.2%, and the CNN-only model 51.9%. The CNN-LSTM model predicts better than individual models but uses 22% more computational resources. It is ideal for critical manufacturing applications where accuracy is more crucial than computational cost.

Our future work will concentrate on several significant areas of improvement. We are enthusiastic about using a time-evolution simulation to demonstrate how factors influence outcomes. To cultivate automated optimization capabilities that enhance equipment longevity, reduce energy consumption, and balance out production throughput with failure risk.

We are eager to enhance visualization, too. Our visualization system requires more realistic models and dynamic effects to accurately depict complex interconnections

12

and physical processes within manufacturing equipment. Enhancing model performance will help in exceeding current limitations. Utilization of more balanced and comprehensive training data, enhanced oversampling techniques for the minority class, attention mechanisms to highlight the most pertinent features, and transfer learning from analogous manufacturing domains will be implemented.

## Acknowledgment

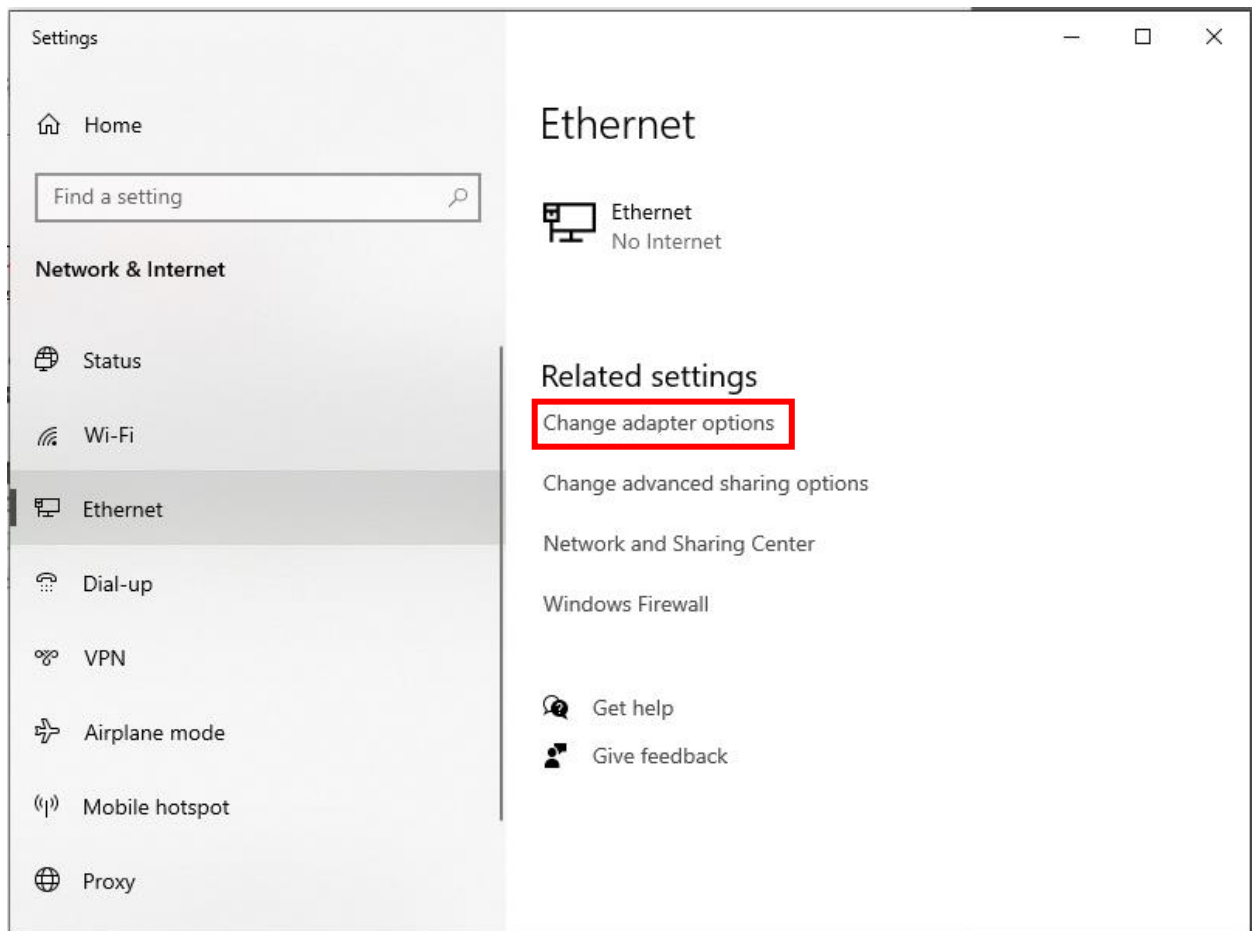
This research was conducted at the Digital Media and Interaction Research Laboratory (DMI), Faculty of Informatics, Burapha University. We express our sincere gratitude for the grant and the support provided.

## References

1. Tao, F., Zhang, H., Liu, A., Nee, A.Y.C.: Digital Twin in Industry. State-of-the-Art. *IEEE Transactions on Industrial Informatics* 15, 2405–2415 (2019)
2. Lu, Y., Liu, C., Wang, K.I.-K., Huang, H., Xu, X.: Digital Twin-Driven Smart Manufacturing. Connotation, Reference Model, Applications and Research Issues. *Robotics and Computer-Integrated Manufacturing* 61, 101837 (2020)
3. Fuller, A., Fan, Z., Day, C., Barlow, C.: Digital Twin. Enabling Technologies, Challenges and Open Research. *IEEE Access* 8, 108952–108971 (2020)
4. Hu, L., Nguyen, N.-T., Tao, W., Leu, M.C., Liu, X.F., Shahriar, M.R., Al Sunny, S.M.N.: Modeling of Cloud-Based Digital Twins for Smart Manufacturing with MT Connect. *Procedia Manufacturing* 26, 1193–1203 (2019)
5. Zhang, J., Wang, P., Yan, R., Gao, R.X.: Deep Learning for Improved System Remaining Life Prediction. *Procedia CIRP* 81, 305–310 (2019)
6. Zhao, R., Yan, R., Chen, Z., Mao, K., Wang, P., Gao, R.X.: Deep Learning and Its Applications to Machine Health Monitoring. *Mechanical Systems and Signal Processing* 115, 213–237 (2019)
7. Zope, K., Singh, K., Nistala, S.H., Basak, A., Rathore, P., Runkana, V.: Anomaly detection and diagnosis in manufacturing systems. A comparative study of statistical, machine learning and deep learning techniques. In: *Annu. Conf. PHM Soc*, 11 (2019)
8. Makridakis, S., Spiliotis, E., Assimakopoulos, V., Semenoglou, A.-A., Mulder, G., Nikolopoulos, K.: Statistical, Machine Learning and Deep Learning Forecasting Methods. Comparisons and Ways Forward. *Journal of the Operational Research Society* 74, 840–859 (2022)
9. Muthukumar, G., Philip, J.: CNN-LSTM Hybrid Deep Learning Model for Remaining Useful Life Estimation. *International Journal of Computational Intelligence and Applications* 21, 2250005 (2022)

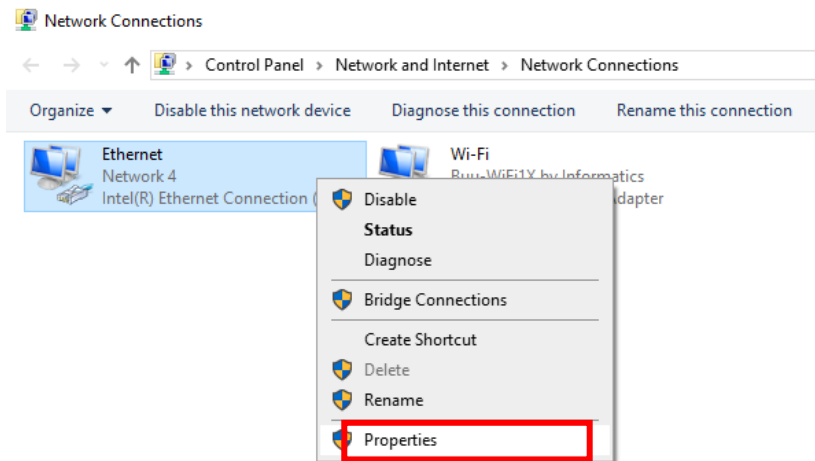
### 7.3 ขั้นตอนการเชื่อมต่อ PLCnext กับ PLCnext engineer

1. เชื่อมต่อเครื่อง PLC เข้ากับคอมพิวเตอร์ ด้วยสาย LAN
2. เปิด Network & Internet เลือกหัวข้อ Ethernet และคลิกที่ Change adapter options



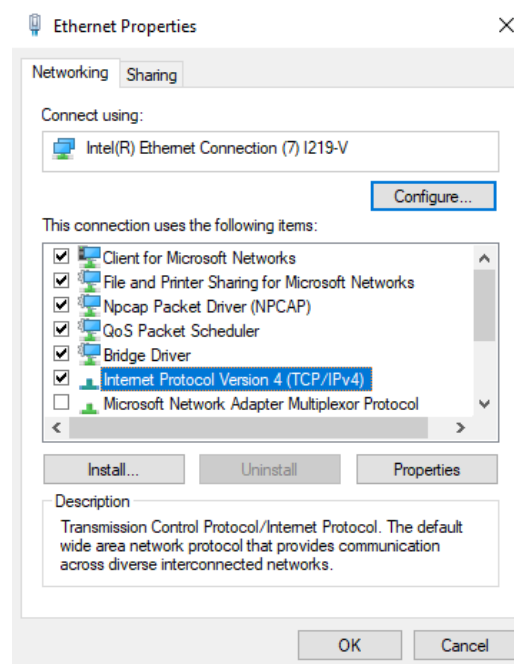
ภาพที่ 9 หน้าต่าง Network & Internet

3. คลิกขวาที่ Ethernet จากนั้นเลือก Properties



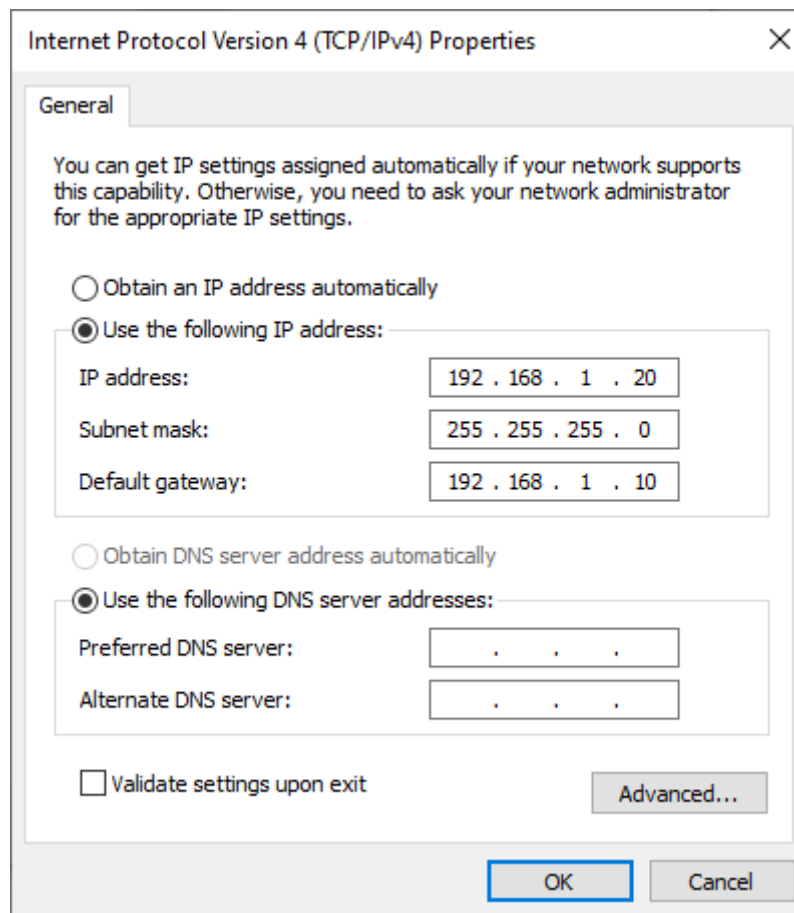
ภาพที่ 10 กด Properties ของ Ethernet

#### 4. ดับเบิลคลิกที่ Internet Protocol Version 4 (TCP/IPv4)



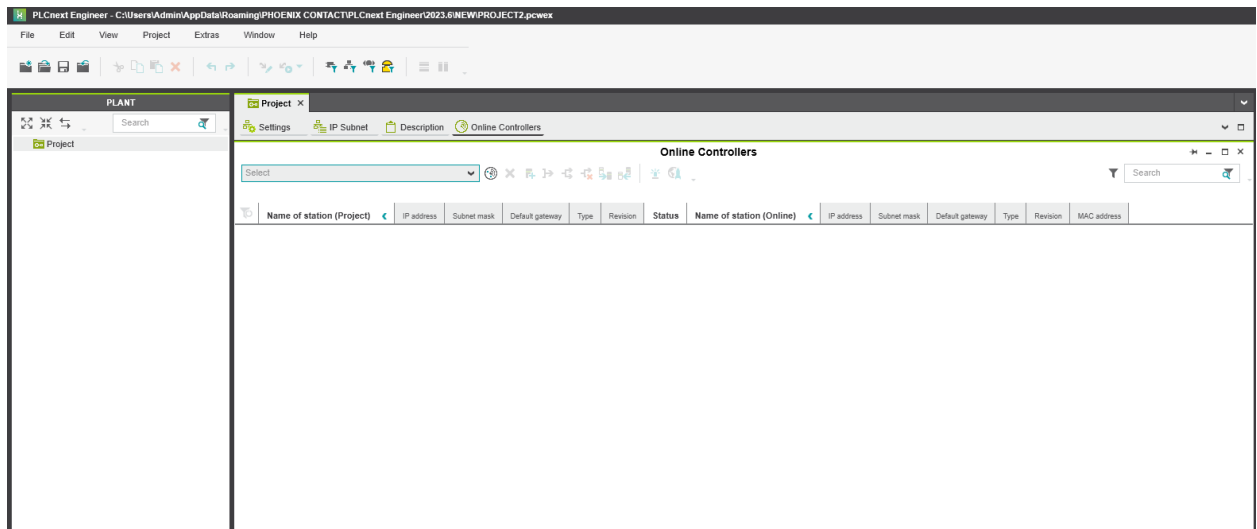
ภาพที่ 11 การเลือก IPv4

#### 5. กำหนด IP address อยู่ในวงแลนเดียวกัน และห้ามทับกับ Default gateway ของ PLC



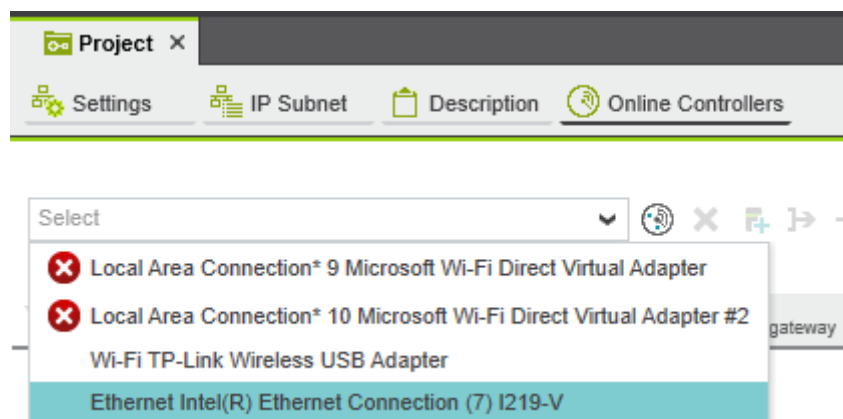
ภาพที่ 12 การกำหนดให้อยู่วง LAN เดียวกัน

6. เปิดโปรแกรม PLCnext engineer และเลือก Project ทางด้านซ้ายมือ และเลือกหัวข้อ Online Controllers



ภาพที่ 13 หน้าต่าง UI ของ PLCnext engineer

7. คลิกที่ drop down icon ให้เลือก Ethernet



ภาพที่ 14 เลือกพอร์ตที่เชื่อมต่อกับ PLC

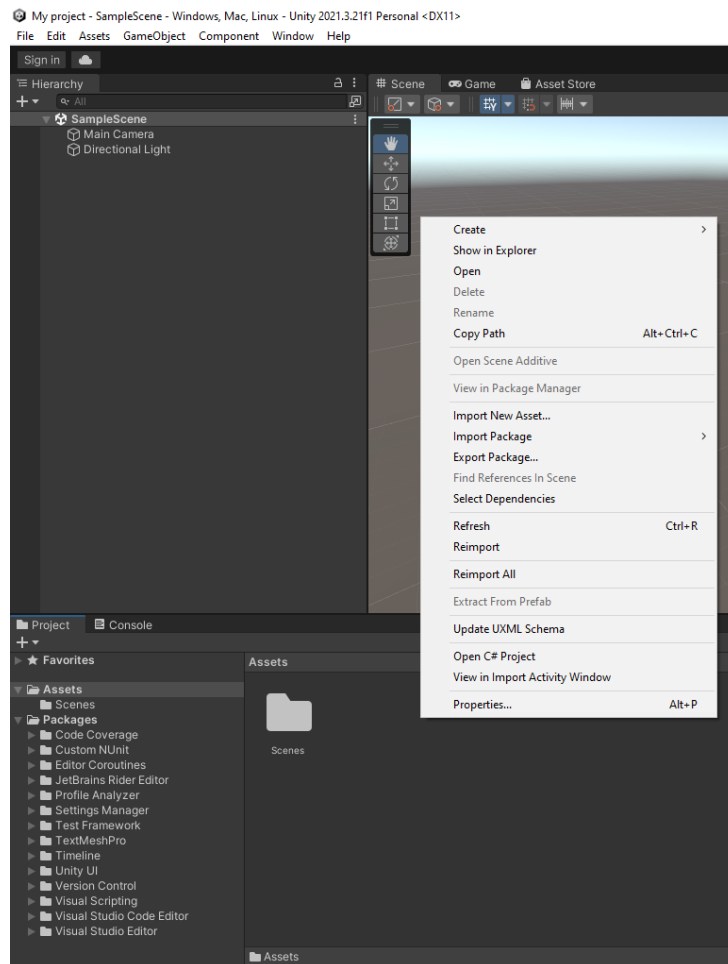
8. คลิกที่ปุ่ม Scan the network



ภาพที่ 15 การกด Scan network เพื่อ ค้นหา PLC

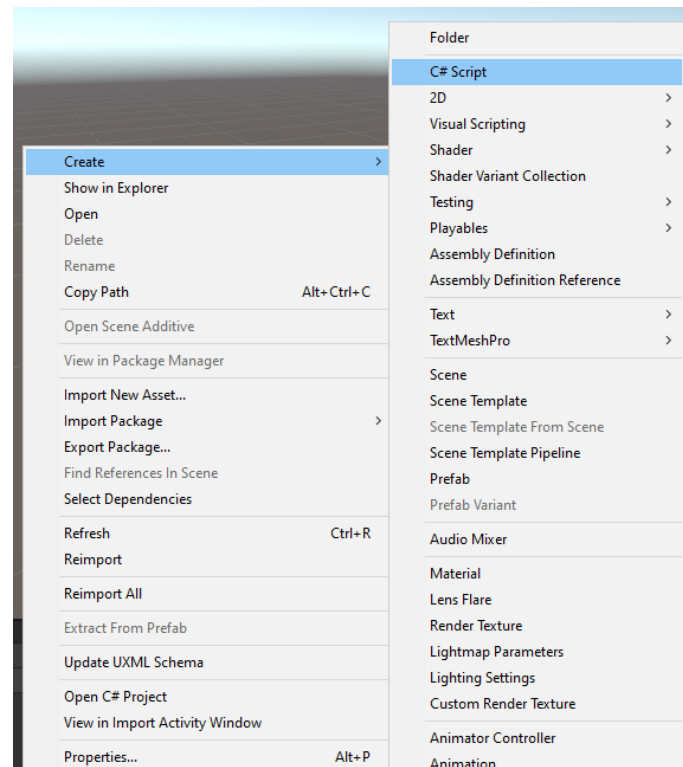
## 7.4 การสร้างและใช้งานสคริปต์ใน Unity

### 1. คลิกขวาที่พื้นที่ว่างในช่องของ Assets จากนั้นเลือก Create



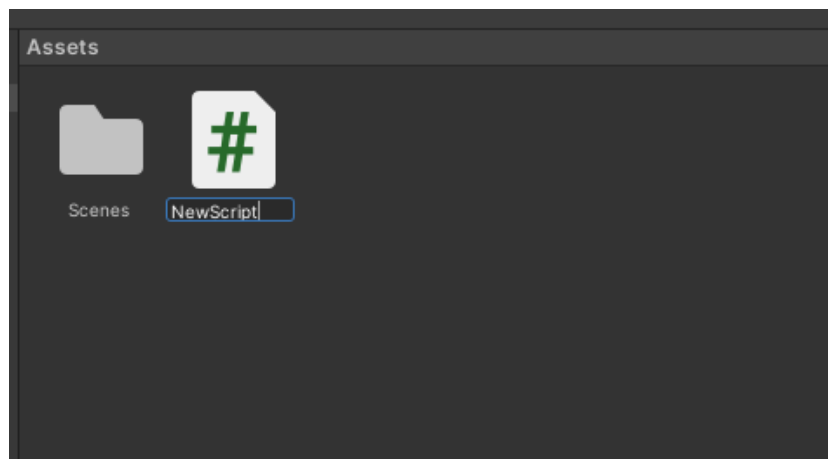
ภาพที่ 16 การสร้าง Script ใน Unity

### 2. หลังจากเลือก Create แล้ว ให้เลือก C# Script



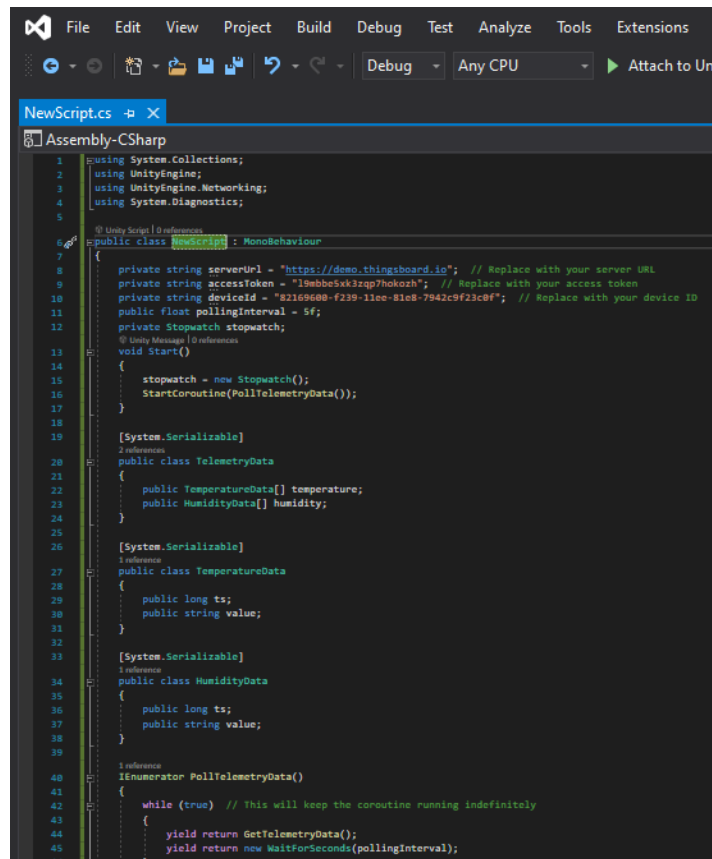
ภาพที่ 17 การสร้าง Script ใน Unity (ต่อ)

### 3. ตั้งชื่อสคริปต์ที่สร้างตามต้องการ



ภาพที่ 18 การตั้งชื่อ Script ใน Unity

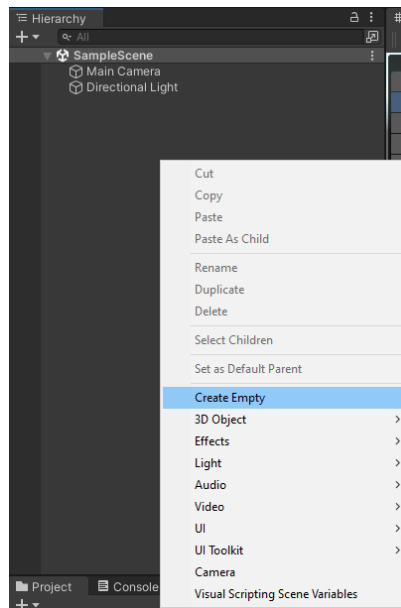
### 4. ทำการ Coding ตามฟังก์ชันที่ใช้



```
1 using System.Collections;
2 using UnityEngine;
3 using UnityEngine.Networking;
4 using System.Diagnostics;
5
6 public class NewScript : MonoBehaviour
7 {
8     private string serverUrl = "https://demo.thingsboard.io"; // Replace with your server URL
9     private string accessToken = "1smbbe5kk3zq7hokozh"; // Replace with your access token
10    private string deviceId = "822169680-f239-11ee-81e8-7942c9f23c0f"; // Replace with your device ID
11    public float pollingInterval = 5f;
12    private Stopwatch stopwatch;
13
14    void Start()
15    {
16        stopwatch = new Stopwatch();
17        StartCoroutine(PollTelemetryData());
18    }
19
20    [System.Serializable]
21    public class TelemetryData
22    {
23        public TemperatureData[] temperature;
24        public HumidityData[] humidity;
25    }
26
27    [System.Serializable]
28    public class TemperatureData
29    {
30        public long ts;
31        public string value;
32    }
33
34    [System.Serializable]
35    public class HumidityData
36    {
37        public long ts;
38        public string value;
39    }
40
41    IEnumerator PollTelemetryData()
42    {
43        while (true) // This will keep the coroutine running indefinitely
44        {
45            yield return GetTelemetryData();
46            yield return new WaitForSeconds(pollingInterval);
47        }
48    }
49 }
```

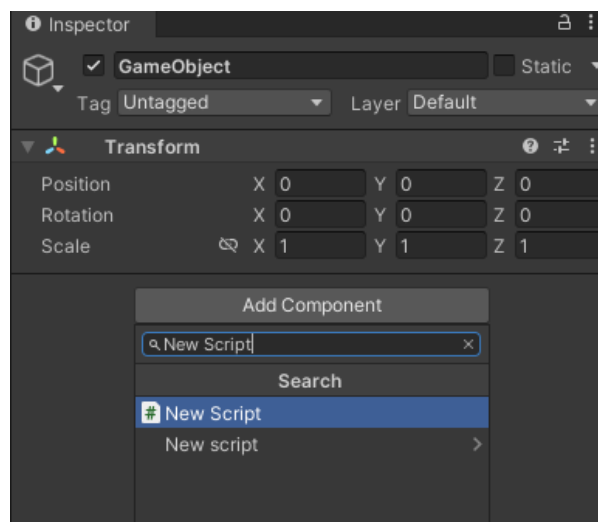
ภาพที่ 19 การเขียนโค้ดลง Script

5. คลิกขวาที่ช่อง Hierarchy เลือก Create Empty จะเป็นการสร้าง Game Object เพื่อใช้เป็นที่ใส่ Script ที่เราสร้างไว้ให้ใช้งานได้ ใน Scene



ภาพที่ 20 การสร้าง Game Object

6. เลือก Game Object ที่เราสร้าง เลือก Add component ให้เราใส่ Script ที่เขียนโค้ดเสร็จแล้ว



ภาพที่ 21 การใส่ Script ลงไปใน Game Object

## 7.5 อธิบายโค้ดที่ใช้เบื้องต้น

1. นำเข้า Library

```
using System.Collections;
using UnityEngine;
using UnityEngine.Networking;
using System.Diagnostics;
```

ภาพที่ 22 โค้ดการนำเข้า Library

## 2. ประกาศและแทนค่าตัวแปร

```
private string serverUrl = "https://demo.thingsboard.io";
private string accessToken = "19mbbe5xk3zqp7hokozh";
private string deviceId = "82169600-f239-11ee-81e8-7942c9f23c0f";
public float pollingInterval = 5f;
private Stopwatch stopwatch;
```

ภาพที่ 23 โค้ดการประกาศตัวแปร

## 3. ให้มีการดึงข้อมูลตลอดเวลาตาม interval ที่ตั้งค่าไว้

```
IEnumerator PollTelemetryData()
{
    while (true)
    {
        yield return GetTelemetryData();
        yield return new WaitForSeconds(pollingInterval);
    }
}
```

ภาพที่ 24 โค้ดกำหนดให้โค้ดรันเป็น Loop เป็นรอบวินาทีที่กำหนด

## 4. ทำการดึงข้อมูลจากเซิร์ฟเวอร์ โดยเริ่มจับเวลา ส่งคำขอดึงข้อมูล และหยุดจับเวลาเมื่อคำขอเสร็จสิ้น

```

IEnumerator GetTelemetryData()
{
    stopwatch.Restart();
    string url = $"{serverUrl}/api/plugins/telemetry/DEVICE/{deviceId}/values/timeseries";

    UnityWebRequest request = UnityWebRequest.Get(url);
    request.SetRequestHeader("Authorization", "Bearer " + accessToken);
    request.SetRequestHeader("Accept", "application/json");

    yield return request.SendWebRequest();
    stopwatch.Stop();

    if (request.result == UnityWebRequest.Result.Success)
    {
        string jsonResponse = request.downloadHandler.text;
        UnityEngine.Debug.Log("Telemetry Data: " + jsonResponse);
        TelemetryData telemetryData = JsonUtility.FromJson<TelemetryData>(jsonResponse);
    }
    else
    {
    }
    UnityEngine.Debug.Log("Time taken: " + stopwatch.Elapsed.TotalMilliseconds + " ms");
    yield return null;
}

```

ภาพที่ 25 ส่งคำขอถึงข้อมูลและการจับเวลา

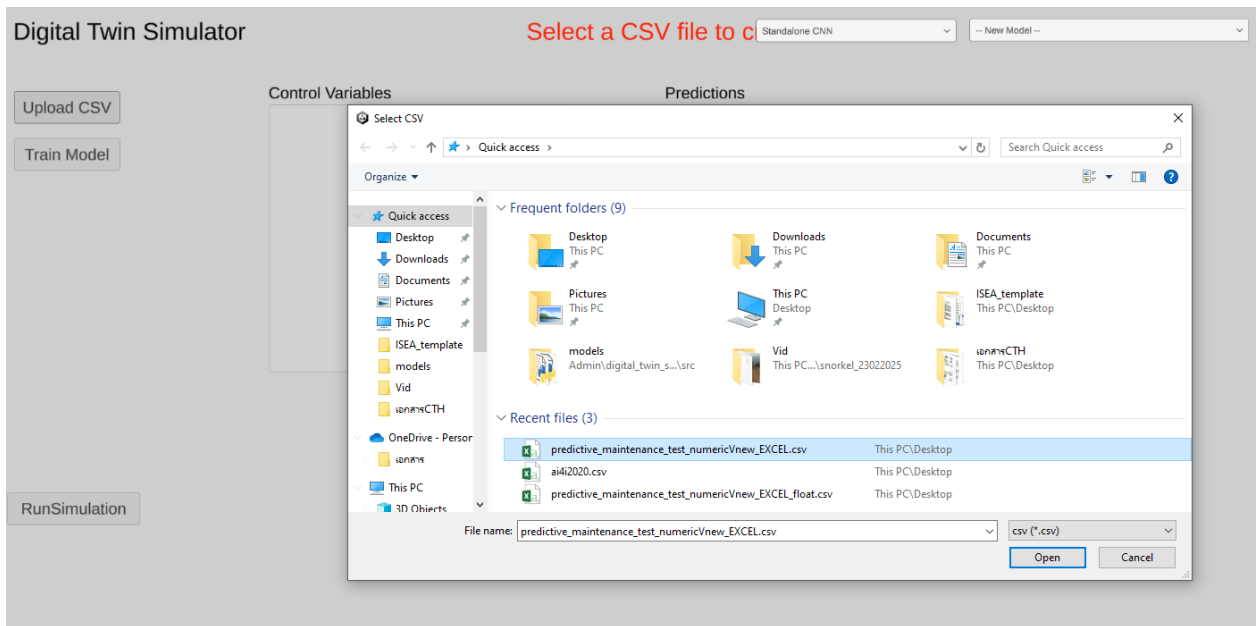
## 7.6 ขั้นตอนการใช้แอปพลิเคชันคู่มือดิจิทัล

1. เลือกโมเดลที่ต้องใช้สำหรับการทำนาย



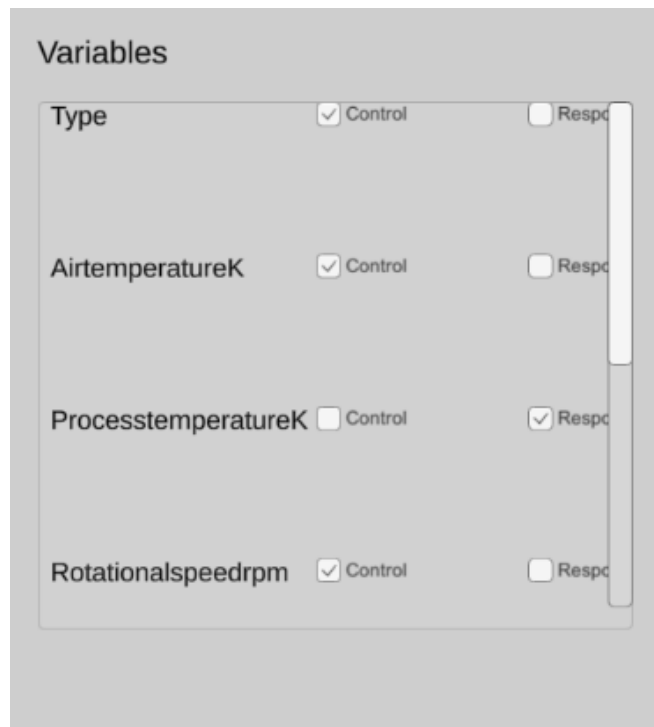
ภาพที่ 26 การเลือกโมเดลสำหรับการทำนาย

2. เลือกชุดข้อมูลที่ต้องการนำมาทำนาย



ภาพที่ 27 การเลือกชุดข้อมูล

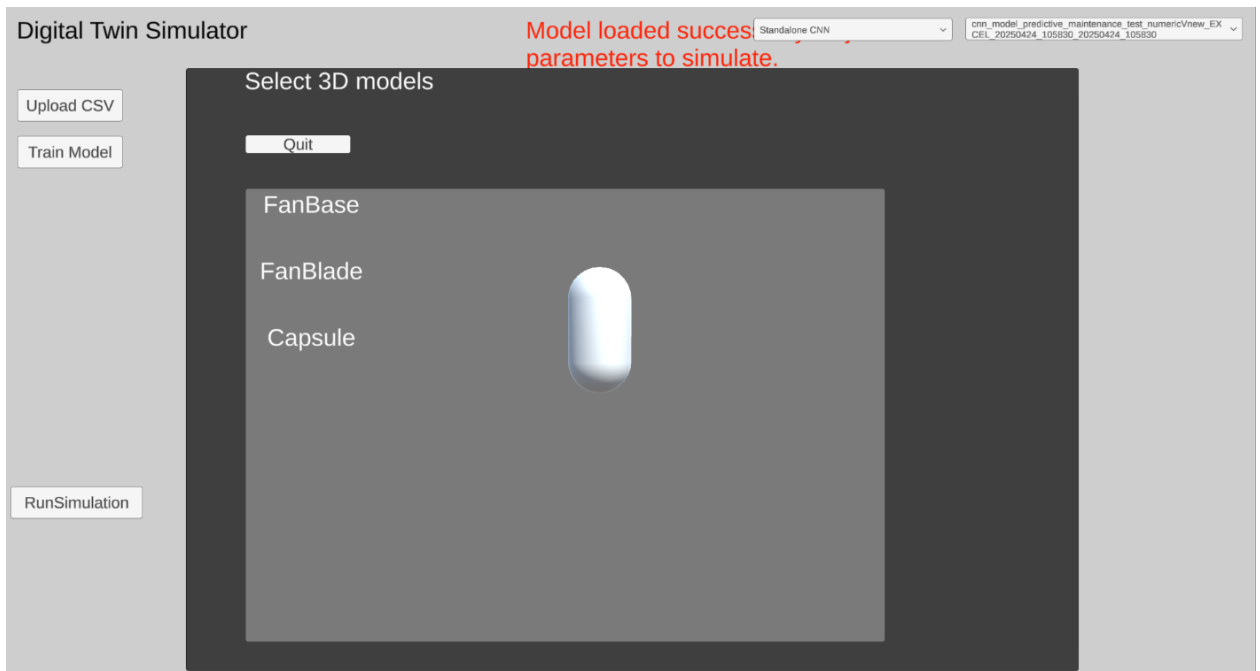
3.เลือกตัวแปรที่ต้องการทำนาย และ เลือกตัวแปรที่คาดว่าจะส่งผลต่อตัวแปรนั้น ๆ



ภาพที่ 28 การเลือกตัวแปรที่ต้องการทำนาย

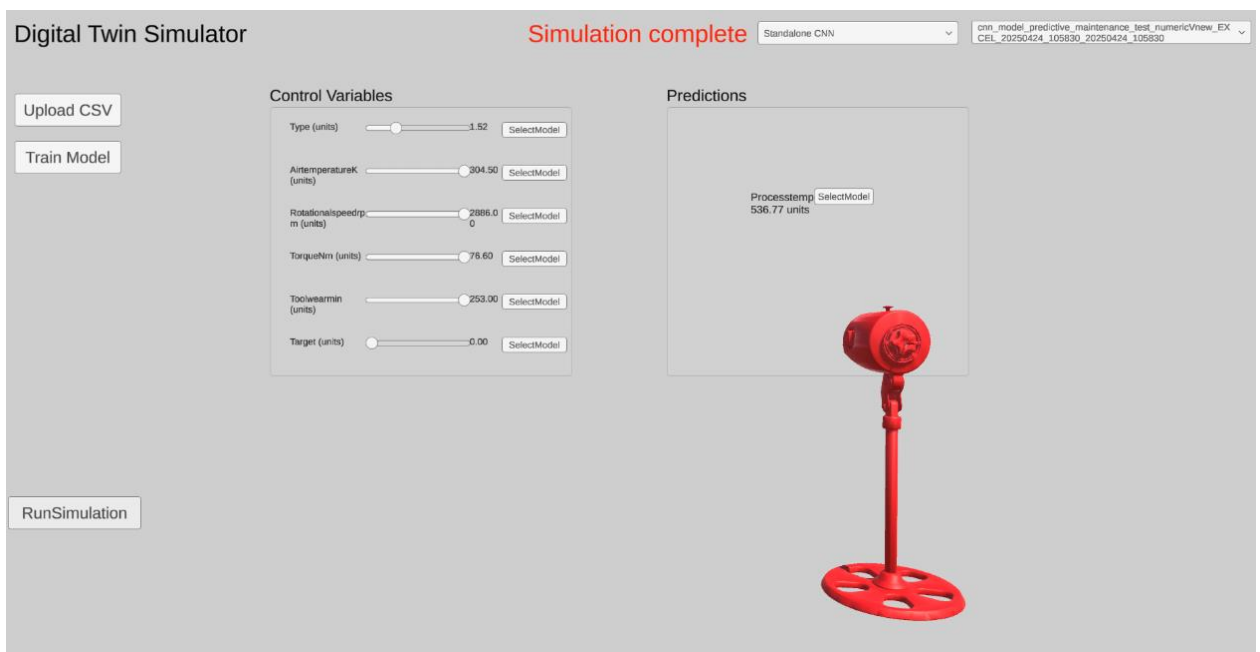
4.รอโมเดลเทรนเสร็จ





ภาพที่ 31 เลือกโมเดลที่สอดคล้องกับชุดข้อมูลที่มี

7.สามารถเริ่มใช้งานการปรับพารามิเตอร์ และสามารถสังเกตผลลัพธ์ที่จะเกิดขึ้นกับอุปกรณ์แบบเรียลไทม์



ภาพที่ 32 ปรับพารามิเตอร์และทำการเปลี่ยนแปลงของโมเดลแบบเรียลไทม์