

การทำนายระดับประจุแบตเตอรี่ของกลุ่มแบตเตอรี่ลิเทียมไอออนด้วยวิธี
ปัญญาประดิษฐ์โดยพิจารณาค่าความต้านทานภายในแบตเตอรี่

SOC ESTIMATION OF LITHIUM-ION BATTERY PACK USING ARTIFICIAL
INTELLIGENCE CONSIDERING BATTERY INTERNAL RESISTANCE



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ (หลักสูตรสหวิทยาการ)
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ.2566

KMITL-2023-EN-M-027-167

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SOC ESTIMATION OF LITHIUM-ION BATTERY PACK USING ARTIFICIAL
INTELLIGENCE CONSIDERING BATTERY INTERNAL RESISTANCE



A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF ENGINEERING IN ELECTRICAL AND COMPUTER ENGINEERING
(MULTI-DISCIPLINARY PROGRAM)
SCHOOL OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2023

KMITL-2023-EN-M-027-167

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2023

SCHOOL OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์

การทำนายระดับประจุแบตเตอรี่ของกลุ่มแบตเตอรี่ลิเทียมไอออน
ด้วยวิธีปัญญาประดิษฐ์โดยพิจารณาค่าความต้านทานภายใน
แบตเตอรี่

นักศึกษา

นายพศุทธิ์ กรุดสอน

รหัสประจำตัว

64601232

ปริญญา

วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชา

วิศวกรรมไฟฟ้าและคอมพิวเตอร์ (หลักสูตรสหวิทยาการ)

พ.ศ.

2566

อาจารย์ที่ปรึกษาวิทยานิพนธ์

ผศ.ดร.เปี่ยมภูมิ สฤกพฤกษ์

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้นำเสนอการทำนายระดับประจุของกลุ่มแบตเตอรี่ลิเทียมไอออนด้วยวิธีปัญญาประดิษฐ์โดยพิจารณาค่าความต้านทานภายในด้วยข้อมูลจากระบบจัดการแบตเตอรี่ภายใต้การใช้งานที่เป็นการคายประจุผ่านภาระทางไฟฟ้าชนิดกระแสคงที่และชนิดกำลังไฟฟ้าคงที่ การอัดประจุชนิดกระแสคงที่แรงดันคงที่ ที่อุณหภูมิห้อง โดยใช้โมเดลปัญญาประดิษฐ์ชนิดมีการป้อนกลับ 3 ชนิด ได้แก่ 1) Recurrent Neural Network (RNN) 2) Long Short-term Memory (LSTM) 3) Gated Recurrent Unit (GRU) และโมเดลปัญญาประดิษฐ์ชนิดไม่มีการป้อนกลับ 1 ชนิด ได้แก่ 1) Neural Network (NN) ผลลัพธ์ที่ได้สามารถสรุปได้ว่าการพิจารณาค่าความต้านทานภายในร่วมด้วยสามารถลดความผิดพลาดในการทำนายระดับประจุแบตเตอรี่ลงได้แต่จะเป็นการเพิ่มระยะเวลาที่ใช้ในการสอนและทำนายในทั้ง 4 โมเดล เมื่อพิจารณาค่าความผิดพลาดและระยะเวลาจะสรุปได้ว่าโมเดลปัญญาประดิษฐ์ชนิด Long Short-term Memory ที่พิจารณาค่าความต้านทานภายในร่วม มีความเหมาะสมที่สุดมีค่าความผิดพลาดต่ำกว่า 1.8% เมื่อคำนวณแบบค่าเฉลี่ยสัมบูรณ์ และต่ำกว่า 2.5% เมื่อคำนวณแบบค่ารากที่สองของกำลังสองเฉลี่ย

Thesis	SoC Estimation of Lithium-ion Battery Pack using Artificial Intelligence Considering Battery Internal Resistance
Student	Mr. Pasut Groudsorn
Student ID.	64601232
Degree	Master of Engineering
Program	Electrical and Computer Engineering (Multi-Disciplinary Program)
Year	2023
Thesis Advisor	Assoc. Prof. Dr. Piampoom Sarikpruek

ABSTRACT

This thesis proposes the State of Charge (SoC) estimation of the Lithium-ion Battery Pack using artificial intelligence considering battery internal resistance based on data from the battery management system. The tests are composed of discharging test with constant current and constant power loads as well as charging test with constant current and constant voltage (CC/CV) at ambient temperature. Three types of artificial intelligence with feedback are evaluated: 1) Recurrent Neural Network (RNN), 2) Long Short-term Memory (LSTM), and 3) Gated Recurrent Unit (GRU) and one type of artificial intelligence without feedback is evaluated: 1) Neural Network (NN). As the results, the models with internal resistance can reduce error for SoC estimation but the predicting time for all models are increased. In conclusion, considering both error and time, Long Short-term Memory with internal resistance model is the most suitable model. This model achieves an error rate lower than 1.8% calculated by Mean Absolute Error (MAE) and lower than 2.5% calculated by Root Mean Squared Error (RMSE)

กิตติกรรมประกาศ

วิทยานิพนธ์เล่มนี้สำเร็จลุล่วงได้ด้วยดีจากความกรุณา ความช่วยเหลือ คำปรึกษา คำแนะนำ ความรู้ และมอบประสบการณ์ที่มีคุณค่ายิ่ง จาก ผศ.ดร.เปี่ยมภูมิ สฤกพฤกษ์ และ รศ.พิชิต ถ้ายอง อาจารย์ผู้ควบคุมวิทยานิพนธ์ฉบับนี้ อันเป็นประโยชน์ในการจัดทำวิทยานิพนธ์ฉบับเล่มนี้และต่อผู้เขียนอย่างยิ่ง ผู้เขียนจึงขอขอบพระคุณเป็นอย่างสูงในความอนุเคราะห์จากท่านด้วยความเคารพอย่างสูง

ขอขอบคุณห้องปฏิบัติการวิจัยเครื่องจักรกลไฟฟ้า (Electrical Machine Research Laboratory : EMRL) ภาควิชาวิศวกรรมไฟฟ้า สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เป็นสถานที่ทำงานวิจัย เรียนรู้ และมอบประสบการณ์ที่มีคุณค่าแก่ผู้เขียน

ขอขอบคุณทุนวิจัยส่งเสริมส่วนงานวิชาการ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่ให้การสนับสนุนงานวิจัยนี้

สุดท้าย ขอขอบพระคุณบิดา มารดา และครอบครัวที่เป็นกำลังใจให้ผู้เขียนและสนับสนุนในทุกๆ อย่างเสมอมา

พศุตม์ กรุดสอน

สารบัญ

	หน้า
บทคัดย่อ.....	i
ABSTRACT.....	ii
กิตติกรรมประกาศ.....	iii
สารบัญ.....	iv
สารบัญตาราง.....	vii
สารบัญรูป.....	viii
บทที่ 1 บทนำ.....	1
1.1. ความเป็นมาและความสำคัญของปัญหา.....	1
1.2. ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	3
1.3. ขอบเขตการวิจัย.....	3
1.4. แนวความคิดที่ใช้ในการวิจัย.....	4
1.5. ขั้นตอนการศึกษา.....	4
1.6. ประโยชน์ที่คาดว่าจะได้รับ.....	5
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....	6
2.1. งานวิจัยที่เกี่ยวข้อง.....	6
2.2. แบตเตอรี่ลิเทียมไอออน.....	8
2.2.1. ประเภทของแบตเตอรี่ลิเทียมไอออน.....	8
2.2.1.1. แบตเตอรี่ลิเทียมโคบอลต์ออกไซด์.....	8
2.2.1.2. แบตเตอรี่ลิเทียมแมงกานีสออกไซด์.....	9
2.2.1.3. แบตเตอรี่ลิเทียมไอออนฟอสเฟต.....	10
2.2.1.4. แบตเตอรี่ลิเทียมนิกเกิลแมงกานีสโคบอลต์ออกไซด์.....	10
2.2.1.5. แบตเตอรี่ลิเทียมนิกเกิลโคบอลต์อะลูมิเนียมออกไซด์.....	11
2.2.1.6. แบตเตอรี่ลิเทียมไททาเนต.....	11
2.3. ระบบจัดการแบตเตอรี่.....	13
2.4. ระดับประจุแบตเตอรี่.....	13
2.4.1. การทำนายระดับประจุของกลุ่มแบตเตอรี่.....	14
2.4.1.1. การทำนายแต่ละเซลล์.....	14
2.4.1.2. การทำนายโดยรวมเซลล์.....	14
2.4.1.3. การทำนายโดยเลือกเซลล์อ้างอิง.....	14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1.4. การทำนายโดยเซลล์เฉลี่ยและค่าความแตกต่าง.....	15
2.5. ค่าความต้านทานภายในแบตเตอรี่.....	15
2.6. โมเดลปัญญาประดิษฐ์.....	16
2.6.1. Recurrent Neural Network.....	16
2.6.2. Long Short-term Memory.....	17
2.6.3. Gated Recurrent Unit.....	17
2.6.4. การหาค่าความผิดพลาด.....	18
2.7. การสื่อสารข้อมูลผ่านพอร์ตอนุกรมแบบอะซิงโครนัส.....	19
2.8. การเตรียมข้อมูล.....	19
2.8.1. การแบ่งข้อมูล.....	19
2.8.2. การทำให้อยู่ในบรรทัดฐานเดียวกัน.....	20
2.8.3. การคัดเลือกคุณลักษณะ.....	20
บทที่ 3 วิธีการดำเนินงานวิจัย.....	21
3.1. การออกแบบการเก็บรับ ส่ง และเตรียมข้อมูล.....	21
3.1.1. วิธีการเก็บรับและส่งข้อมูล.....	21
3.1.2. การเตรียมข้อมูล.....	23
3.2. การออกแบบโมเดลปัญญาประดิษฐ์.....	28
3.2.1. การหาค่าพารามิเตอร์ของโมเดลปัญญาประดิษฐ์.....	28
3.2.2. การทำงานของโมเดลปัญญาประดิษฐ์.....	29
3.2.3. การวิเคราะห์ผล.....	29
3.2.3.1. การวิเคราะห์ผลจากค่าความผิดพลาด.....	30
3.2.3.2. การวิเคราะห์ระยะเวลาที่ใช้ในการสอนและทดสอบโมเดลปัญญาประดิษฐ์.....	30
3.3. การออกแบบการทดสอบแบตเตอรี่.....	30
3.3.1. การทดสอบการอัดประจุ.....	32
3.3.2. การทดสอบการคายประจุ.....	33
บทที่ 4 ผลการดำเนินงานวิจัย.....	34
4.1. ผลการปรับแต่งพารามิเตอร์ของโมเดลปัญญาประดิษฐ์.....	34
4.1.1. ค่าพารามิเตอร์โมเดล Neural Network.....	34
4.1.2. ค่าพารามิเตอร์โมเดล Recurrent Neural Network.....	37
4.1.3. ค่าพารามิเตอร์โมเดล Long Short-term Memory.....	40
4.1.4. ค่าพารามิเตอร์โมเดล Gated Recurrent Unit.....	43
4.2. ผลการทำนายและระยะเวลาในการสอนและทำนายระดับประจุแบตเตอรี่.....	46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1. ผลจากการสอนและทำนายโดยโมเดล Neural Network.....	47
4.2.1.1. ผลการทำนายระดับประจูปแบบเตอรีโดยโมเดล Neural Network.....	47
4.2.1.2. ระยะเวลาในการสอนและทำนายผลโดยโมเดล Neural Network.....	49
4.2.2. ผลจากการสอนและทำนายโดยโมเดล Recurrent Neural Network.....	50
4.2.2.1. ผลการทำนายระดับประจูปแบบเตอรีโดยโมเดล Recurrent Neural Network	50
4.2.2.2. ระยะเวลาในการสอนและทำนายผลโดยโมเดล Recurrent Neural Network	53
4.2.3. ผลและระยะเวลาการทำนายโดยโมเดล Long Short-term Memory.....	53
4.2.3.1. ผลการทำนายระดับประจูปแบบเตอรีโดยโมเดล Long Short-term Memory.	54
4.2.3.2. ระยะเวลาในการสอนและทำนายผลโดยโมเดล Long Short-term Memory	57
4.2.4. ผลและระยะเวลาการทำนายโดยโมเดล Gated Recurrent Unit.....	57
4.2.4.1. ผลการทำนายระดับประจูปแบบเตอรีโดยโมเดล Gated Recurrent Unit.....	57
4.2.4.2. ระยะเวลาในการสอนและทำนายผลโดยโมเดล Gated Recurrent Unit.....	60
4.3. เปรียบเทียบผลของโมเดลปัญญาประดิษฐ์ที่ต่างชนิด	61
4.3.1. เปรียบเทียบค่าความผิดพลาดในการทำนายระดับประจูปแบบเตอรี	61
4.3.2. เปรียบเทียบระยะเวลาการสอนและการทำนายระดับประจูปแบบเตอรี	65
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ	67
5.1. สรุปผลการวิจัย.....	67
5.2. ข้อเสนอแนะ	69
เอกสารอ้างอิง	70
ภาคผนวก.....	73
ภาคผนวก ก บทความวิชาการ	74
ภาคผนวก ข โปรแกรม Arduino รับส่งข้อมูล.....	79
ภาคผนวก ค โปรแกรม Jupyter Lab โมเดลปัญญาประดิษฐ์	85
ประวัติผู้เขียน.....	95

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 เปรียบเทียบประสิทธิภาพและการใช้งานของแบตเตอรี่ลิเทียมไอออนต่างชนิด [18].....	12
3.1 ระดับประจุแบตเตอรี่และแรงดันเปิดวงจร.....	27
4.1 ค่าความผิดพลาดเพี้ยนบนชุดข้อมูลตรวจสอบของโมเดลชนิด Neural Network.....	37
4.2 ค่าพารามิเตอร์โมเดลปัญญาประดิษฐ์ชนิด Neural Network.....	37
4.3 ค่าความผิดพลาดเพี้ยนบนชุดข้อมูลตรวจสอบของโมเดลชนิด Recurrent Neural Network.....	40
4.4 ค่าพารามิเตอร์โมเดลปัญญาประดิษฐ์ชนิด Recurrent Neural Network.....	40
4.5 ค่าความผิดพลาดเพี้ยนบนชุดข้อมูลตรวจสอบของโมเดลชนิด Long Short-term Memory.....	43
4.6 ค่าพารามิเตอร์โมเดลปัญญาประดิษฐ์ชนิด Long Short-term Memory.....	43
4.7 ค่าความผิดพลาดเพี้ยนบนชุดข้อมูลตรวจสอบของโมเดลชนิด Gated Recurrent Unit.....	46
4.8 ค่าพารามิเตอร์โมเดลปัญญาประดิษฐ์ชนิด Gated Recurrent Unit.....	46
4.9 ค่าความผิดพลาดเพี้ยนบนชุดข้อมูลสอนของโมเดล Neural Network.....	48
4.10 ค่าความผิดพลาดเพี้ยนบนชุดข้อมูลทดสอบของโมเดล Neural Network.....	49
4.11 ระยะเวลาการสอนและทำนายของโมเดล Neural Network.....	50
4.12 ค่าความผิดพลาดเพี้ยนบนชุดข้อมูลสอนของโมเดล Recurrent Neural Network.....	51
4.13 ค่าความผิดพลาดเพี้ยนบนชุดข้อมูลทดสอบของโมเดล Recurrent Neural Network.....	52
4.14 ระยะเวลาการสอนและทำนายของโมเดล Recurrent Neural Network.....	53
4.15 ค่าความผิดพลาดเพี้ยนบนชุดข้อมูลสอนของโมเดล Long Short-term Memory.....	55
4.16 ค่าความผิดพลาดเพี้ยนบนชุดข้อมูลทดสอบของโมเดล Long Short-term Memory.....	56
4.17 ระยะเวลาการสอนและทำนายของโมเดล Long Short-term Memory.....	57
4.18 ค่าความผิดพลาดเพี้ยนบนชุดข้อมูลสอนของโมเดล Gated Recurrent Unit.....	58
4.19 ค่าความผิดพลาดเพี้ยนบนชุดข้อมูลทดสอบของโมเดล Gated Recurrent Unit.....	59
4.20 ระยะเวลาการสอนและทำนายของโมเดล Gated Recurrent Unit.....	60
4.21 ค่าความผิดพลาดเพี้ยนระหว่างโมเดลปัญญาประดิษฐ์ต่างชนิดบนชุดข้อมูลสอนและทดสอบ.....	61
4.22 ระยะเวลาระหว่างโมเดลปัญญาประดิษฐ์ต่างชนิดในการสอนและทดสอบบนชุดข้อมูลสอนและทดสอบ.....	66
5.1 ตัวอย่างที่แนะนำในการใช้งานโมเดลปัญญาประดิษฐ์ในงานวิจัยนี้.....	68

สารบัญรูป

รูปที่	หน้า
1.1 ปริมาณการผลิตพลังงานไฟฟ้าจากพลังงานหมุนเวียนในประเทศไทย	1
1.2 การคาดการณ์ปริมาณการใช้งานแบตเตอรี่ลิเทียมไอออนทั่วโลก [2].....	2
2.1 สมรรถนะของแบตเตอรี่ลิเทียมโคบอลต์ออกไซด์ [18]	9
2.2 สมรรถนะแบตเตอรี่ลิเทียมแมงกานีสออกไซด์ [18]	9
2.3 สมรรถนะแบตเตอรี่ลิเทียมไอรอนฟอสเฟต [18].....	10
2.4 สมรรถนะแบตเตอรี่ลิเทียมนิกเกิลแมงกานีสโคบอลต์ออกไซด์ [18]	11
2.5 สมรรถนะแบตเตอรี่ลิเทียมนิกเกิลโคบอลต์อะลูมิเนียมออกไซด์ [18]	11
2.6 สมรรถนะแบตเตอรี่ลิเทียมไททานเนต [18].....	12
2.7 การทำนายระดับประจุกลุ่มแบตเตอรี่แบบแต่ละเซลล์ [19]	14
2.8 การทำนายระดับประจุกลุ่มแบตเตอรี่แบบรวมเซลล์ [19]	14
2.9 การทำนายระดับประจุกลุ่มแบตเตอรี่โดยเลือกเซลล์อ้างอิง [19].....	15
2.10 การทำนายระดับประจุกลุ่มแบตเตอรี่โดยเซลล์เฉลี่ยและความแตกต่าง [19]	15
2.11 Thevenin model ของแบตเตอรี่ [7].....	16
2.12 โครงสร้าง Long Short-term Memory [15].....	17
2.13 โครงสร้าง Gated Recurrent Unit [16]	18
3.1 การเชื่อมต่อการสื่อสารชนิด UART	21
3.2 ข้อมูลฐาน 16 จาก BMS.....	22
3.3 ตัวอย่างข้อมูลในงานวิจัยที่ถูกส่งขึ้น Google Sheet	23
3.4 ตัวอย่างข้อมูลในงานวิจัยก่อนทำให้อยู่ในบรรทัดฐานเดียวกัน	23
3.5 ตัวอย่างข้อมูลในงานวิจัยหลังทำให้อยู่ในบรรทัดฐานเดียวกัน	24
3.6 ค่าสัมประสิทธิ์ Pearson correlation ในงานวิจัย	24
3.7 การแบ่งชุดข้อมูลตรวจสอบแบบ TimeSeriesSplit [22].....	25
3.8 การแบ่งชุดข้อมูลตรวจสอบในงานวิจัย	26
3.9 กราฟความสัมพันธ์ระหว่างแรงดันเปิดวงจรกับระดับประจุแบตเตอรี่.....	28
3.10 วงจรไฟฟ้าขณะทำการอัดประจุ.....	31
3.11 วงจรไฟฟ้าขณะทำการอัดประจุที่ทดลองจริง	31
3.12 วงจรไฟฟ้าขณะทำการคายประจุ	32
3.13 วงจรไฟฟ้าขณะทำการคายประจุจริง	32
3.14 กราฟคุณลักษณะการอัดประจุของข้อมูลที่ใช้ในงานวิจัย.....	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.15 กราฟคุณลักษณะการคายประจุของข้อมูลที่ใช้ในงานวิจัย.....	33
4.1 กราฟความสัมพันธ์ระหว่างความผิดพลาดสอนความผิดพลาดที่เรียนรู้ตรวจสอบกับจำนวนรอบการ สอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 1 ของโมเดลชนิด Neural Network.....	35
4.2 กราฟความสัมพันธ์ระหว่างความผิดพลาดสอนความผิดพลาดที่เรียนรู้ตรวจสอบกับจำนวนรอบการ สอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 2 ของโมเดลชนิด Neural Network.....	35
4.3 กราฟความสัมพันธ์ระหว่างความผิดพลาดสอนความผิดพลาดที่เรียนรู้ตรวจสอบกับจำนวนรอบการ สอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 3 ของโมเดลชนิด Neural Network.....	36
4.4 กราฟความสัมพันธ์ระหว่างความผิดพลาดสอนความผิดพลาดที่เรียนรู้ตรวจสอบกับจำนวนรอบการ สอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 4 ของโมเดลชนิด Neural Network.....	36
4.5 กราฟความสัมพันธ์ระหว่างความผิดพลาดสอนความผิดพลาดที่เรียนรู้ตรวจสอบกับจำนวนรอบการ สอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 1 ของโมเดลชนิด Recurrent Neural Network	38
4.6 กราฟความสัมพันธ์ระหว่างความผิดพลาดสอนความผิดพลาดที่เรียนรู้ตรวจสอบกับจำนวนรอบการ สอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 2 ของโมเดลชนิด Recurrent Neural Network	38
4.7 กราฟความสัมพันธ์ระหว่างความผิดพลาดสอนความผิดพลาดที่เรียนรู้ตรวจสอบกับจำนวนรอบการ สอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 3 ของโมเดลชนิด Recurrent Neural Network	39
4.8 กราฟความสัมพันธ์ระหว่างความผิดพลาดสอนความผิดพลาดที่เรียนรู้ตรวจสอบกับจำนวนรอบการ สอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 4 ของโมเดลชนิด Recurrent Neural Network	39
4.9 กราฟความสัมพันธ์ระหว่างความผิดพลาดสอนความผิดพลาดที่เรียนรู้ตรวจสอบกับจำนวนรอบการ สอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 1 ของโมเดลชนิด Long Short-term Memory	41
4.10 กราฟความสัมพันธ์ระหว่างความผิดพลาดสอนความผิดพลาดที่เรียนรู้ตรวจสอบกับจำนวนรอบการ สอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 2 ของโมเดลชนิด Long Short-term Memory	41
4.11 กราฟความสัมพันธ์ระหว่างความผิดพลาดสอนความผิดพลาดที่เรียนรู้ตรวจสอบกับจำนวนรอบการ สอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 3 ของโมเดลชนิด Long Short-term Memory	42

4.12 กราฟความสัมพันธ์ระหว่างความผิดพลาดที่เพิ่มขึ้นของชุดสอนความผิดพลาดที่เพิ่มขึ้นชุดตรวจสอบกับจำนวนรอบการ สอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 4 ของโมเดลชนิด Long Short-term Memory	42
4.13 กราฟความสัมพันธ์ระหว่างความผิดพลาดที่เพิ่มขึ้นของชุดสอนความผิดพลาดที่เพิ่มขึ้นชุดตรวจสอบกับจำนวนรอบการ สอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 1 ของโมเดลชนิด Gated Recurrent Unit	44
4.14 กราฟความสัมพันธ์ระหว่างความผิดพลาดที่เพิ่มขึ้นของชุดสอนความผิดพลาดที่เพิ่มขึ้นชุดตรวจสอบกับจำนวนรอบการ สอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 2 ของโมเดลชนิด Gated Recurrent Unit	44
4.15 กราฟความสัมพันธ์ระหว่างความผิดพลาดที่เพิ่มขึ้นของชุดสอนความผิดพลาดที่เพิ่มขึ้นชุดตรวจสอบกับจำนวนรอบการ สอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 3 ของโมเดลชนิด Gated Recurrent Unit	45
4.16 กราฟความสัมพันธ์ระหว่างความผิดพลาดที่เพิ่มขึ้นของชุดสอนความผิดพลาดที่เพิ่มขึ้นชุดตรวจสอบกับจำนวนรอบการ สอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 4 ของโมเดลชนิด Gated Recurrent Unit	45
4.17 ระดับประจูปฏิบัติการที่ทำงานโดยโมเดล Neural Network บนชุดข้อมูลสอน	47
4.18 ภาพขยายระดับประจูปฏิบัติการที่ทำงานโดยโมเดล Neural Network บนชุดข้อมูลสอน	47
4.19 ระดับประจูปฏิบัติการที่ทำงานโดยโมเดล Neural Network บนชุดข้อมูลทดสอบ	48
4.20 ภาพขยายระดับประจูปฏิบัติการที่ทำงานโดยโมเดล Neural Network บนชุดข้อมูลทดสอบ	48
4.21 สัมประสิทธิ์ความสัมพันธ์ระหว่างค่าความต้านทานภายในและระดับประจูปฏิบัติการของโมเดล ปัญญาประดิษฐ์ชนิด Neural Network บนชุดข้อมูลทดสอบ	49
4.22 ระดับประจูปฏิบัติการที่ทำงานโดยโมเดล Recurrent Neural Network บนชุดข้อมูลสอน	50
4.23 ภาพขยายระดับประจูปฏิบัติการที่ทำงานโดยโมเดล Recurrent Neural Network บนชุดข้อมูล สอน	51
4.24 ระดับประจูปฏิบัติการที่ทำงานโดยโมเดล Recurrent Neural Network บนชุดข้อมูลทดสอบ	52
4.25 ภาพขยายระดับประจูปฏิบัติการที่ทำงานโดยโมเดล Recurrent Neural Network บนชุดข้อมูล ทดสอบ	52
4.26 สัมประสิทธิ์ความสัมพันธ์ระหว่างค่าความต้านทานภายในและระดับประจูปฏิบัติการของโมเดล ปัญญาประดิษฐ์ชนิด Recurrent Neural Network บนชุดข้อมูลทดสอบ	53
4.27 ระดับประจูปฏิบัติการที่ทำงานโดยโมเดล Long Short-term Memory บนชุดข้อมูลสอน	54
4.28 ภาพขยายระดับประจูปฏิบัติการที่ทำงานโดยโมเดล Long Short-term Memory บนชุดข้อมูล สอน	54
4.29 ระดับประจูปฏิบัติการที่ทำงานโดยโมเดล Long Short-term Memory บนชุดข้อมูลทดสอบ	55
4.30 ภาพขยายระดับประจูปฏิบัติการที่ทำงานโดยโมเดล Long Short-term Memory บนชุดข้อมูล ทดสอบ	56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.31 สัมประสิทธิ์ความสัมพันธ์ระหว่างค่าความต้านทานภายในและระดับประจุแบตเตอรี่ของโมเดล ปัญญาประดิษฐ์ชนิด Long Short-term Memory บนชุดข้อมูลทดสอบ.....	57
4.32 ระดับประจุแบตเตอรี่ทำนายโดยโมเดล Gated Recurrent Unit บนชุดข้อมูลสอน.....	58
4.33 ภาพขยายระดับประจุแบตเตอรี่ทำนายโดยโมเดล Gated Recurrent Unit บนชุดข้อมูลสอน.....	58
4.34 ระดับประจุแบตเตอรี่ทำนายโดยโมเดล Gated Recurrent Unit บนชุดข้อมูลทดสอบ.....	59
4.35 ภาพขยายระดับประจุแบตเตอรี่ทำนายโดยโมเดล Gated Recurrent Unit บนชุดข้อมูลทดสอบ	59
4.36 สัมประสิทธิ์ความสัมพันธ์ระหว่างค่าความต้านทานภายในและระดับประจุแบตเตอรี่ของโมเดล ปัญญาประดิษฐ์ชนิด Gated Recurrent Unit บนชุดข้อมูลทดสอบ	60
4.37 เปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลสอน เมื่อไม่พิจารณาค่าความต้านทานภายใน	62
4.38 ภาพขยายเปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อทำนายระดับประจุแบตเตอรี่บนชุด ข้อมูลสอนเมื่อพิจารณาค่าความต้านทานภายใน	62
4.39 ภาพขยายเปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อทำนายระดับประจุแบตเตอรี่บนชุด ข้อมูลสอนเมื่อไม่พิจารณาค่าความต้านทานภายใน	63
4.40 ภาพขยายเปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อทำนายระดับประจุแบตเตอรี่บนชุด ข้อมูลสอนเมื่อพิจารณาค่าความต้านทานภายใน	63
4.41 เปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อทำนายระดับประจุแบตเตอรี่บนชุดข้อมูล ทดสอบเมื่อไม่พิจารณาค่าความต้านทานภายใน.....	64
4.42 เปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อทำนายระดับประจุแบตเตอรี่บนชุดข้อมูล ทดสอบเมื่อพิจารณาค่าความต้านทานภายใน.....	64
4.43 ภาพขยายเปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อทำนายระดับประจุแบตเตอรี่บนชุด ข้อมูลทดสอบเมื่อไม่พิจารณาค่าความต้านทานภายใน	65
4.44 ภาพขยายเปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อทำนายระดับประจุแบตเตอรี่บนชุด ข้อมูลทดสอบเมื่อพิจารณาค่าความต้านทานภายใน	65
4.45 ระยะเวลาระหว่างโมเดลปัญญาประดิษฐ์ต่างชนิดในการสอนและทดสอบบนชุดข้อมูลสอนและ ทดสอบ	66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

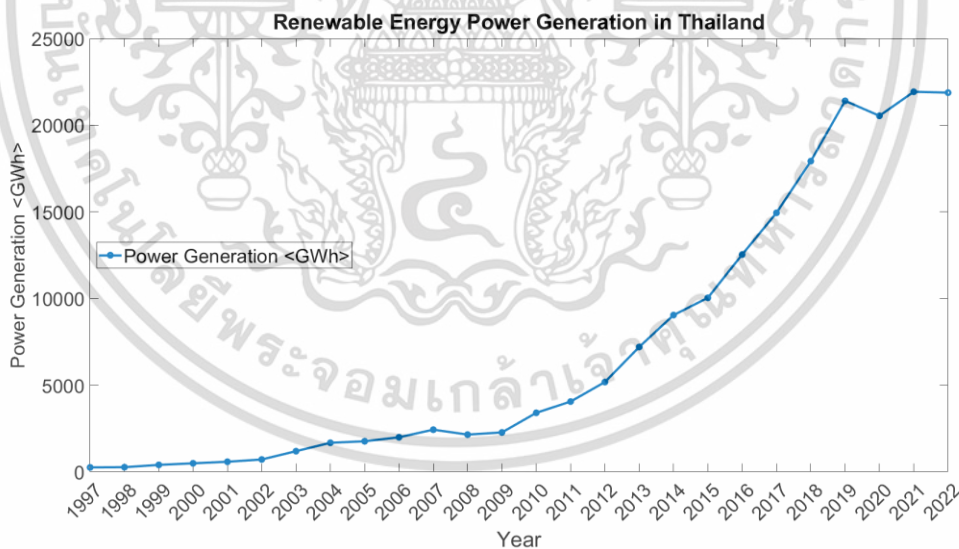
บทที่ 1

บทนำ

1.1. ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันพลังงานไฟฟ้าหมุนเวียนเป็นรูปแบบของพลังงานไฟฟ้าที่กำลังเป็นที่ใช้งานอย่างแพร่หลายมากขึ้นเนื่องมาจากการต้องการพลังงานไฟฟ้าที่มีความยั่งยืน และความกังวลจากการผลิตพลังงานไฟฟ้าในรูปแบบเดิมที่อาศัยเชื้อเพลิงฟอสซิลในการผลิต ซึ่งก่อให้เกิดมลภาวะทางสิ่งแวดล้อม โดยพลังงานไฟฟ้าหมุนเวียนทำให้การปลดปล่อยก๊าซเรือนกระจกลดลง ลดการนำเข้าเชื้อเพลิงฟอสซิล และอีกปัจจัยที่ทำให้พลังงานไฟฟ้าหมุนเวียนเป็นที่นิยมคือการเข้าถึงเทคโนโลยีด้านพลังงานไฟฟ้าหมุนเวียนที่สะดวกมากยิ่งขึ้น

ข้อมูลรายงานสถิติพลังงาน จากกระทรวงพลังงาน แสดงให้เห็นว่าปริมาณการผลิตพลังงานไฟฟ้าจากพลังงานหมุนเวียนในประเทศไทยมีทิศทางที่เพิ่มขึ้นโดยในปี 2022 มีปริมาณการผลิตพลังงานไฟฟ้าจากพลังงานหมุนเวียนสูงถึง 21,876.09 GWh ดังรูปที่ 1.1 โดยคิดเป็นสัดส่วน 10% จากจำนวนการผลิตไฟฟ้าทั้งประเทศไทยเป็นอันดับที่ 4 รองมาจาก ก๊าซธรรมชาติ ลิกไนต์/ถ่านหิน และการนำเข้า [1]

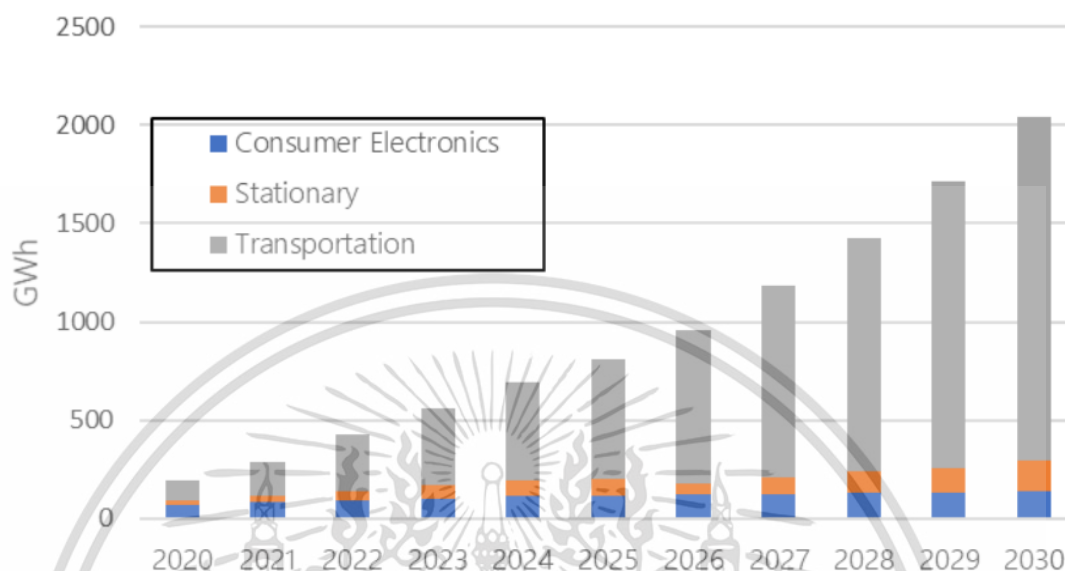


รูปที่ 1.1 ปริมาณการผลิตพลังงานไฟฟ้าจากพลังงานหมุนเวียนในประเทศไทย

เมื่อมีการใช้งานพลังงานไฟฟ้าหมุนเวียนที่มากขึ้น ทำให้แบตเตอรี่ก็เริ่มมีการใช้งานอย่างแพร่หลายมากขึ้นเช่นกัน โดยเฉพาะแบตเตอรี่ชนิดลิเทียมไอออน (Lithium – ion Battery) ที่มีอัตราการเติบโตอย่างรวดเร็วที่สุดในหมวดหมู่แบตเตอรี่ที่สามารถอัดประจุใหม่ได้ เป็นที่นิยมในการใช้งานร่วมกับระบบพลังงานไฟฟ้าหมุนเวียน โดยมีการใช้งานทั้งในภาคครัวเรือนและภาคอุตสาหกรรม มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานในอุปกรณ์ต่างๆ เช่น โทรศัพท์มือถือ คอมพิวเตอร์ และยานยนต์ไฟฟ้า โดยมีการคาดการณ์ว่าจะได้รับความนิยมมากขึ้นเรื่อย ๆ คาดการณ์ว่าจะมีปริมาณถึง 2 TWh ในปี 2030 ดังรูปที่ 1.2 [2] ซึ่งแสดงให้เห็นว่าในอนาคตแบตเตอรี่ชนิดลิเทียมไอออนจะมีอิทธิพลในการใช้ชีวิตเพิ่มขึ้นเรื่อย ๆ



รูปที่ 1.2 การคาดการณ์ปริมาณการใช้งานแบตเตอรี่ลิเทียมไอออนทั่วโลก [2]

เมื่อแบตเตอรี่ชนิดลิเทียมไอออนมีบทบาทมากขึ้น ส่งผลให้การใช้งานแบตเตอรี่อย่างมีประสิทธิภาพนั้นเป็นสิ่งสำคัญ เพื่อก่อให้เกิดการใช้ประโยชน์อย่างสูงสุดและมีประสิทธิภาพ การให้ความสำคัญกับการคำนวณหาค่าระดับประจุแบตเตอรี่ (State of Charge: SoC) ค่าสุขภาพแบตเตอรี่ (State of Health: SoH) หรือ อายุการใช้งานที่เหลือของแบตเตอรี่ (Remaining Useful Life: RUL) ล้วนส่งผลต่อประสิทธิภาพแบตเตอรี่ได้ เมื่อค่าเหล่านี้ถูกคำนวณอย่างแม่นยำจะทำให้เราสามารถวางแผนการใช้งานแบตเตอรี่ได้อย่างมีประสิทธิภาพได้และลดความเสียหายได้

การทำนายระดับประจุแบตเตอรี่ วิธีที่ง่ายที่สุดคือการวัดแรงดันเปิดวงจรหรือวัดกระแสไฟฟ้าโดยตรงแต่วิธีที่ซับซ้อนมากกว่ามักจะถูกนำมาใช้เพื่อรับมือกับความผิดพลาดของเซนเซอร์และความรู้ด้านโมเดลแบตเตอรี่ที่ยังไม่แน่นอน เทคโนโลยีปัญญาประดิษฐ์ (Artificial Intelligence: AI) เป็นหนึ่งในวิธีการทำนายระดับประจุแบตเตอรี่โดยอาศัยข้อมูลเช่น แรงดัน กระแส และอุณหภูมิ ก็สามารถทำนายระดับประจุแบตเตอรี่ได้ หรือการรวมค่าความต้านทานภายในแบตเตอรี่เข้าไปก็สามารถช่วยลดค่าความผิดพลาดลงได้ ในการทำนายนั้นมีปัจจัยสำคัญที่ส่งผลต่อความแม่นยำได้แก่ชนิดแบตเตอรี่ชนิดของโมเดลปัญญาประดิษฐ์ รวมไปถึงค่าพารามิเตอร์ของโมเดลปัญญาประดิษฐ์ที่ต้องการปรับให้เหมาะสมกับแต่ละชุดข้อมูล และเงื่อนไขในชุดข้อมูลที่นำมาใช้ในการสอนและทดสอบโมเดลปัญญาประดิษฐ์ก็มีผลต่อความแม่นยำ โดยโมเดลปัญญาประดิษฐ์ชนิดที่มีการป้อนกลับนั้นจะให้ค่าความผิดพลาดที่ต่ำ [3]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากข้อสรุปข้างต้นนั้น ในการทำนายหาค่าระดับประจุแบตเตอรี่โดยวิธีปัญญาประดิษฐ์นั้น มีประสิทธิภาพและความสามารถที่แม่นยำสำหรับทั้งชุดข้อมูลที่มีการเปลี่ยนแปลงภาระทางไฟฟ้าและชุดข้อมูลที่ภาระทางไฟฟ้าคงที่รวมถึงชุดข้อมูลที่อุณหภูมิแตกต่างกัน อย่างไรก็ตามงานวิจัยส่วนมากจะมุ่งศึกษาถึงชนิดของโมเดลปัญญาประดิษฐ์ไม่ได้ให้ความสำคัญกับพารามิเตอร์ความต้านทานภายในแบตเตอรี่เท่าที่ควรและส่วนมากจะทำการทดสอบกับแบตเตอรี่จำนวน 1 เซลล์ ซึ่งไม่ได้ตอบโจทย์ในการใช้งานจริง ดังนั้นการทำนายหาค่าระดับประจุแบตเตอรี่โดยวิธีปัญญาประดิษฐ์ในวิทยานิพนธ์นี้นั้นจะเป็นการศึกษาถึงการทำนายค่าระดับประจุแบตเตอรี่ภายใต้สภาวะการใช้งานจริงที่ใช้ข้อมูลจากระบบจัดการแบตเตอรี่ และมีการพิจารณาค่าความต้านทานภายในร่วมด้วย เพื่อหาโมเดลปัญญาประดิษฐ์ที่มีความเหมาะสมที่สุด

1.2. ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

จากความเป็นมาและความสำคัญของปัญหา วิทยานิพนธ์เล่มนี้มีจุดมุ่งหมายคือ ออกแบบโมเดลปัญญาประดิษฐ์ที่สามารถทำนายค่าระดับประจุแบตเตอรี่ โดยใช้ข้อมูลจากระบบจัดการแบตเตอรี่ภายใต้การใช้งานที่เป็นการคายประจุผ่านภาระทางไฟฟ้าชนิดกระแสคงที่และชนิดกำลังไฟฟ้าคงที่ การอัดประจุชนิดกระแสคงที่แรงดันคงที่ (Constant Current/Constant Voltage: CC/CV) ที่อุณหภูมิห้อง เปรียบเทียบโมเดลปัญญาประดิษฐ์ที่ออกแบบระหว่างโมเดลที่มีการพิจารณาค่าความต้านทานภายในแบตเตอรี่กับโมเดลปัญญาประดิษฐ์ที่ไม่มีการพิจารณาค่าความต้านทานภายใน และเปรียบเทียบโมเดลปัญญาประดิษฐ์ชนิดแตกต่างกันในด้านค่าความผิดพลาดและระยะเวลาการสอนและทำนาย เพื่อหาโมเดลปัญญาประดิษฐ์ที่สามารถทำนายระดับประจุแบตเตอรี่ได้อย่างเหมาะสมที่สุด

1.3. ขอบเขตการวิจัย

งานวิจัยนี้ออกแบบและทดลองขึ้นในห้องปฏิบัติการวิจัยเครื่องจักรกลไฟฟ้า (Electrical Machine Research Laboratory: EMRL) คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ทำการศึกษาโดยใช้แบตเตอรี่ชนิดลิเทียมไอออน รุ่น NCR18650B ยี่ห้อ Panasonic ขนาดความจุ 3,350 mAh แรงดันมาตรฐาน 3.6 V จำนวน 7 เซลล์ต่ออนุกรมกับระบบจัดการแบตเตอรี่ รุ่น JBD-DP24S002 โดยจะทำการสร้างโมเดลปัญญาประดิษฐ์ทั้งแบบพิจารณาและไม่พิจารณาค่าความต้านทานภายใน และโมเดลปัญญาประดิษฐ์ชนิดแตกต่างกันโดยโปรแกรมคอมพิวเตอร์ภาษา Python เขียนบน Jupyter Lab เพื่อเปรียบเทียบหาโมเดลปัญญาประดิษฐ์ที่ดีที่สุดจากชุดข้อมูลอัดประจุแบบกระแสคงที่แรงดันคงที่และชุดข้อมูลคายประจุชนิดกระแสคงที่และกำลังไฟฟ้าคงที่จากระบบจัดการแบตเตอรี่ ที่ได้ทดลองภายใต้สภาวะอุณหภูมิห้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4. แนวความคิดที่ใช้ในการวิจัย

โมเดลปัญญาประดิษฐ์ที่สร้างขึ้นเพื่อศึกษาในวิทยานิพนธ์เล่มนี้จะเป็นโมเดลปัญญาประดิษฐ์ชนิดการเรียนรู้แบบมีผู้สอน (Supervised Learning) ที่มีการป้อนกลับ โดยจะทำการศึกษากับโมเดลปัญญาประดิษฐ์ชนิด Recurrent Neural Network (RNN), Long Short-term Memory (LSTM) และ Gated Recurrent Unit (GRU) เนื่องจากมีความเหมาะสมที่สุดสำหรับการทำนายระดับประจุแบตเตอรี่ที่ข้อมูลเป็นชนิดขึ้นกับเวลา และมีการเปรียบเทียบกับโมเดลปัญญาประดิษฐ์ที่ไม่มีการป้อนกลับชนิด Neural Network (NN) ทำการเปรียบเทียบโมเดลระหว่างที่พิจารณาและไม่พิจารณาค่าความต้านทานภายใน หาประสิทธิภาพของโมเดลปัญญาประดิษฐ์โดยเปรียบเทียบกับค่าความผิดพลาดชนิดค่าสัมบูรณ์เฉลี่ยของค่าความผิดพลาด (Mean Absolute Error: MAE) และค่ารากที่สองของค่าเฉลี่ยกำลังสองของค่าความผิดพลาด (Root Mean Squared Error: RMSE)

ข้อมูลที่ใช้มาจากระบบจัดการแบตเตอรี่จะต่ออยู่กับแบตเตอรี่ชนิดลิเทียมไอออนจำนวน 7 เซลล์ แหล่งจ่ายไฟกระแสตรง (DC Power Supply) ใช้สำหรับการอัดประจุโดยการอัดประจุจะเป็นชนิดกระแสคงที่แรงดันคงที่และต่อกับค่าความต้านทานปรับค่าได้ (Variable Resistor) เพื่อทำการคายประจุที่เป็นชนิดกระแสคงที่และกำลังไฟฟ้าคงที่ โดยทั้งการอัดและคายประจุจะอยู่ภายใต้อุณหภูมิห้อง

ข้อมูลที่ใช้สำหรับการสอนโมเดลปัญญาประดิษฐ์มาจากระบบจัดการแบตเตอรี่ผ่านการอ่านค่า แปลงค่า และส่งค่าไปเก็บโดยไมโครคอนโทรลเลอร์ชนิด ESP32 เนื่องจากสามารถเชื่อมต่อกับอินเทอร์เน็ตได้เพื่อทำการส่งข้อมูล โดยค่าพารามิเตอร์ที่อ่านจากระบบจัดการแบตเตอรี่จะถูกเตรียมและทำให้อยู่ในบรรทัดฐานเดียวกันก่อนที่จะทำการคัดเลือกคุณลักษณะของข้อมูลโดยวิธี Pearson Correlation Coefficient ก่อนนำไปใช้กับโมเดลปัญญาประดิษฐ์ ส่วนค่าความต้านทานภายในแบตเตอรี่ได้มาจากการคำนวณระหว่างการทำนายระดับประจุแบตเตอรี่

1.5. ขั้นตอนการศึกษา

วิทยานิพนธ์นี้ได้จัดเรียงเนื้อหาและแบ่งขั้นตอนรายละเอียดการศึกษาตามหัวข้อดังนี้

บทที่ 2 จะกล่าวถึง การทบทวนวรรณกรรม ทฤษฎีที่เกี่ยวข้องกับงานวิจัยนี้ทั้งโมเดลปัญญาประดิษฐ์ชนิดต่างๆ ทฤษฎีของอุปกรณ์ที่ใช้เป็นหลักในงานวิจัย และวิธีการประเมินผลของโมเดลปัญญาประดิษฐ์

บทที่ 3 จะกล่าวถึง การออกแบบการรับ ส่ง และเตรียมข้อมูล การออกแบบโมเดลปัญญาประดิษฐ์ และการออกแบบการทดสอบแบตเตอรี่

บทที่ 4 จะทำการอภิปรายผลการปรับแต่งค่าพารามิเตอร์ของโมเดลปัญญาประดิษฐ์ ผลการทำนายและระยะเวลาในการสอนและทำนายระดับประจุแบตเตอรี่ และเปรียบเทียบผลของโมเดลปัญญาประดิษฐ์ที่ต่างชนิดในด้านค่าความผิดพลาดและระยะเวลาการสอนและทำนาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5 จะกล่าวถึงการสรุปผลของงานวิจัย ข้อเสนอแนะ และแนวทางการพัฒนางานวิจัยต่อไปในอนาคต

1.6. ประโยชน์ที่คาดว่าจะได้รับ

- 1) ผลจากการทำนายระดับประจุแบตเตอรี่ด้วยวิธีโมเดลปัญญาประดิษฐ์เมื่อพิจารณาค่าความต้านทานภายในแบตเตอรี่
- 2) โมเดลปัญญาประดิษฐ์ที่สามารถทำนายระดับประจุแบตเตอรี่ได้อย่างแม่นยำ
- 3) ใช้เป็นแนวทางสำหรับการทำนายระดับประจุแบตเตอรี่ในระบบจัดการแบตเตอรี่



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1. งานวิจัยที่เกี่ยวข้อง

ในระบบจัดการแบตเตอรี่ (Battery Management System: BMS) ค่าระดับประจุแบตเตอรี่ (State of Charge: SoC) เป็นหนึ่งในตัวชี้วัดที่มีความสำคัญต่อแบตเตอรี่และเป็นองค์ประกอบพื้นฐานสำหรับค่าอื่นๆ เช่น ระดับสุขภาพแบตเตอรี่ (State of Health: SoH) การทำนายค่าระดับประจุแบตเตอรี่ที่แม่นยำ สามารถยืดอายุการใช้งานของแบตเตอรี่ เพิ่มประสิทธิภาพของแบตเตอรี่ และสร้างระบบไฟฟ้าที่เชื่อถือ ในการทำนายระดับประจุแบตเตอรี่สามารถแบ่งได้เป็น 3 วิธีได้แก่ 1) วิธีโดยตรง 2) สร้างโมเดลแบตเตอรี่จากคุณสมบัติทางไฟฟ้าหรือเคมี 3) สร้างโมเดลจากข้อมูลเช่น แรงดัน กระแส และอุณหภูมิ [4,5]

วิธีโดยตรงสามารถแบ่งได้เป็น วิธีแรงดันเปิดวงจร (Open Circuit Voltage: OCV) คือ การสร้างกราฟความสัมพันธ์ระหว่างแรงดันเปิดวงจรกับระดับประจุแบตเตอรี่ ในระหว่างการอัดประจุและคายประจุในทุกช่วงของระดับประจุแบตเตอรี่ โดยสมการที่ได้ จะนำมาใช้ร่วมกับโมเดลแบตเตอรี่เพื่อหาค่าระดับประจุแบตเตอรี่ โดยได้มีการใช้งานร่วมกันระหว่าง Low Current OCV Test กับ Incremental OCV Test เพื่อหาสมการระหว่างแรงดันเปิดวงจรกับระดับประจุแบตเตอรี่ ประกอบกับ Adaptive Extended Kalman Filter (AEKF) เพื่อหาระดับประจุแบตเตอรี่แสดงในงานวิจัยที่ [6] วิธีโดยตรงอีกวิธีคือ การนับกระแส (Coulomb Counting Method) เป็นการคำนวณซึ่งใช้หลักการการรวมกันของกระแสระหว่างอัดประจุหรือคายประจุ คำนวณร่วมกับค่าระดับประจุแบตเตอรี่ตอนเริ่มต้น ตัวอย่างในงานวิจัย [7] ได้ใช้หลักการนับกระแสโดยมีการปรับปรุงค่าระดับประจุแบตเตอรี่ตอนเริ่มต้นโดยใช้ค่าความต้านทานภายในประกอบทำให้มีความแม่นยำที่มากขึ้น

สำหรับการสร้างโมเดลแบตเตอรี่จากคุณสมบัติทางไฟฟ้าหรือเคมี ในงานวิจัย [8] เป็นการสร้างโมเดลแบตเตอรี่จากคุณสมบัติทางไฟฟ้า (Equivalent Circuit Model: ECM) โดยเป็นการสร้างวงจรทางไฟฟ้าของแบตเตอรี่ซึ่งประกอบไปด้วยตัวต้านทาน ตัวเก็บประจุ และแหล่งจ่ายไฟกระแสตรง แล้วนำมาใช้กับ Extended Kalman Filter (EKF) เพื่อหาค่าระดับประจุแบตเตอรี่ในการสร้างโมเดลแบตเตอรี่จากคุณสมบัติทางเคมี (Electrochemical Model: EM) เป็นการหาค่าระดับประจุแบตเตอรี่จากความเข้มข้นในขั้วบวกและลบ เป็นการสร้างโมเดลจากปฏิกิริยาเคมีของแบตเตอรี่และทำการทำนายระดับประจุแบตเตอรี่โดยใช้อัลกอริทึมตัวกรอง ตัวอย่างโมเดลเคมี Electrochemical Single Particle (SP Model) ร่วมกับอัลกอริทึมตัวกรอง EKF แสดงใน [9]

แต่เนื่องจากโมเดลแบตเตอรี่มีความซับซ้อนและมีความไม่เป็นเชิงเส้น ดังนั้นการทำนายระดับประจุแบตเตอรี่โดยการสร้างโมเดลจากข้อมูล โดยมักจะใช้กระบวนการเรียนรู้ของเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Machine Learning: ML) เพื่อหาความสัมพันธ์ในข้อมูล เป็นที่นิยมในการแก้ปัญหาที่ได้กล่าวข้างต้น ซึ่งสามารถสร้างได้ง่ายโดยใช้ข้อมูลแรงดัน กระแส และอุณหภูมิ ทำให้ไม่ต้องใช้ข้อมูลเกี่ยวกับพารามิเตอร์แบตเตอรี่และข้อมูลปฏิกิริยาเคมีของแบตเตอรี่ [4,5]

โมเดลปัญญาประดิษฐ์ชนิด Neural Network ถูกนำมาใช้ทำนายระดับประจุแบตเตอรี่ ในงานวิจัย [10] ได้ใช้อัลกอริทึม Back-Propagation Neural Network (BPNN) ซึ่งจะมีการคำนวณค่าความผิดพลาดของผลลัพธ์ในขั้นก่อนหน้าย้อนกลับมาเพื่อปรับปรุงน้ำหนักของโมเดลปัญญาประดิษฐ์ เช่นเดียวกับ [11] ที่ใช้ Recurrent Neural Network (RNN) ที่มีการคำนวณกลับ ซึ่งให้ความแม่นยำที่สูงกว่าโมเดล Neural network ทั่วไป แต่เนื่องจากระดับประจุแบตเตอรี่เป็นข้อมูลที่เกี่ยวข้องกับเวลาเพราะฉะนั้นโมเดลปัญญาประดิษฐ์ชนิด Long Short-Term Memory (LSTM) หรือ Gate Recurrent Unit (GRU) จะมีความสามารถในด้านนี้ [5] ในงานวิจัยที่ [12] มีการใช้โมเดล LSTM ซึ่งเป็นโมเดลปัญญาประดิษฐ์ที่พัฒนาต่อมาจากโมเดลปัญญาประดิษฐ์ RNN ที่กำจัดปัญหาในการเรียนรู้แบบระยะยาวได้มาทำนายระดับประจุแบตเตอรี่ หรือโมเดลชนิด GRU ที่เป็นการลดความซับซ้อนของโมเดล LSTM แสดงในงานวิจัยที่ [13] ซึ่งมีความแม่นยำที่ใกล้เคียง LSTM ขึ้นอยู่กับการปรับแต่งค่าพารามิเตอร์ แต่เนื่องจากโมเดล GRU มีความซับซ้อนที่น้อยกว่าโมเดล LSTM จึงทำให้ใช้เวลาที่น้อยกว่าในการทำนาย ในงานวิจัย [14] มีการใช้งานโมเดล Bidirectional Gated Recurrent Unit (Bi-GRU) คือการรวม GRU 2 อันสำหรับข้อมูลในอดีตและข้อมูลอนาคตเข้าด้วยกันเพื่อเพิ่มความสามารถของโมเดล และเพื่อเพิ่มความแม่นยำขึ้นไปสามารถทำการรวมโมเดลเข้าด้วยกันได้โดยในงานวิจัยที่ [15] ที่ทำการรวมโมเดลชนิด Convolution Neural Network (CNN) เข้ากับ LSTM และงานวิจัยที่ [16] ที่ทำการรวมโมเดล CNN เข้ากับ GRU โดยการเพิ่มโมเดล CNN เข้าไปจะเป็นการเพิ่มความซับซ้อนให้คุณลักษณะของข้อมูลป้อนเข้า

ตัวแปรที่นอกเหนือจากแรงดัน กระแส หรือ อุณหภูมิ คือ ความต้านทานภายในแบตเตอรี่ ที่สามารถส่งผลต่อระดับประจุแบตเตอรี่ แสดงในงานวิจัยที่ [17] ด้วยโมเดล Adaptive Unscented Kalman Filter (AUKF) ที่เป็นการรวมกันระหว่างโมเดล Unscented Kalman Filter (UKF) ที่ใช้ทำนายระดับประจุแบตเตอรี่และโมเดล Extended Kalman Filter (EKF) ที่ใช้หาค่าความต้านทานภายในแบตเตอรี่ ซึ่งให้ความแม่นยำที่สูงกว่าโมเดล UKF อย่างเดียว

จากการศึกษางานวิจัยที่เกี่ยวข้องการทำนายระดับประจุแบตเตอรี่โดยโมเดลปัญญาประดิษฐ์สามารถใช้งานได้ง่าย งานวิจัยส่วนใหญ่อาศัยข้อมูลที่เป็นข้อมูลในห้องทดลองภายใต้สภาวะควบคุมมาใช้สอนโมเดลปัญญาประดิษฐ์ ทำการทดสอบกับแบตเตอรี่เพียง 1 ก้อน ทำให้โมเดลปัญญาประดิษฐ์นั้นยังไม่ได้พัฒนาสำหรับการใช้งานจริง ที่อยู่ภายใต้การใช้งานแบตเตอรี่ในรูปแบบกลุ่มแบตเตอรี่ที่มีการใช้งานร่วมกับระบบจัดการแบตเตอรี่ ดังนั้นจุดประสงค์ของวิทยานิพนธ์เล่มนี้คือ

- 1) การสร้างโมเดลปัญญาประดิษฐ์ที่สามารถทำนายระดับประจุแบตเตอรี่ของกลุ่มแบตเตอรี่

2) การพิจารณาค่าความต้านทานภายในแบตเตอรี่ร่วมด้วยโดยใช้ข้อมูลภายใต้สภาวะการใช้งานจริงที่เป็นภาระไฟฟ้าชนิดกระแสคงที่และกำลังคงที่ในอุณหภูมิห้อง

3) เปรียบเทียบประสิทธิภาพของโมเดลปัญญาประดิษฐ์ชนิดแตกต่างกันเพื่อหาโมเดลปัญญาประดิษฐ์ที่เหมาะสมกับการทำนายระดับประจุแบตเตอรี่ของกลุ่มแบตเตอรี่

2.2. แบตเตอรี่ลิเทียมไอออน

แบตเตอรี่ลิเทียมไอออน (Lithium – ion Battery) เป็นแบตเตอรี่ชนิดที่สามารถอัดประจุใหม่ได้ ส่วนใหญ่ถูกนำไปใช้ในอุปกรณ์อิเล็กทรอนิกส์ รถยนต์ไฟฟ้า และใช้เพื่อกักเก็บพลังงานในระบบไฟฟ้า จุดเด่นเหนือแบตเตอรี่แบบอัดประจุได้ชนิดอื่นคือมีความหนาแน่นพลังงานที่สูง ไม่ได้รับผลกระทบจากอัดประจุเมื่อคายประจุไปบางส่วนซึ่งทำให้แบตเตอรี่สูญเสียความจุสูงสุดไป (Memory Effect) และอัตราการคายประจุเองต่ำ เนื่องมาจากการใช้แกรไฟต์เป็นขั้วแอโนดทำให้ขั้วแคโทดสามารถสร้างจากวัสดุหลากหลายชนิดได้ซึ่งจะใช้เป็นตัวแบ่งแยกชนิดแบตเตอรี่ลิเทียมไอออน จุดเด่นของแบตเตอรี่ลิเทียมไอออนมีอัตราพลังงานไฟฟ้าต่อน้ำหนักและกำลังไฟฟ้าต่อน้ำหนักที่สูง อายุการใช้งานยาวนาน และมีประสิทธิภาพในการอัดและคายประจุ ส่วนจุดด้อยได้แก่ ราคา ต้องมีอุปกรณ์อิเล็กทรอนิกส์สำหรับป้องกันในระหว่างการอัดและคายประจุ และก๊าซเรือนกระจกที่ถูกปลดปล่อยระหว่างการผลิตและการกำจัด

แบตเตอรี่ลิเทียมไอออนประกอบไปด้วยส่วนประกอบหลัก 4 ส่วนได้แก่ แคโทด แอโนด อิเล็กโทรไลต์ (Electrolyte) และ ตัวกั้น (Seperator) ขั้วแคโทดจากผง Lithium-metal-oxide ขั้วแอโนดจากผงแกรไฟต์ โดยวางเป็นชั้นบนตัวรับกระแสลุ่มนิเยียมและทองแดงตามลำดับ อิเล็กโทรไลต์ประกอบด้วยเกลือลิเทียมและตัวทำละลายอินทรีย์ มีหน้าที่เคลื่อนย้ายลิเทียมไอออนระหว่างแคโทดและแอโนด ส่วนตัวกั้นเป็นเมมเบรนที่มีรูพรุนขนาดเล็กป้องกันการลัดวงจรระหว่างขั้วแคโทดและแอโนดที่ยอมให้แคลิเทียมไอออนผ่านได้เท่านั้น

2.2.1. ประเภทของแบตเตอรี่ลิเทียมไอออน

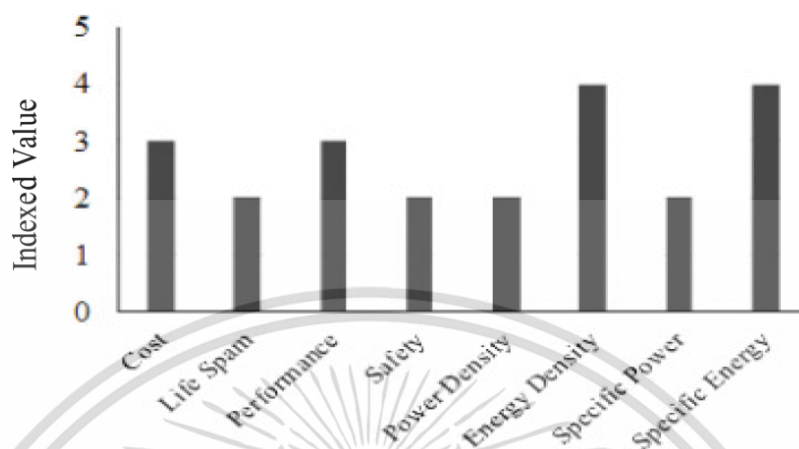
แหล่งลิเทียมไอออนหลักในแบตเตอรี่มาจากขั้วบวกของตัวนำหรือขั้วแคโทดเพื่อให้บรรลุความจุที่สูงลิเทียมจำนวนมากจึงอยู่ในนั้นนอกจากนี้วัสดุแคโทดยังสามารถย้อนกลับเพื่อแลกเปลี่ยนลิเทียมด้วยการปรับเปลี่ยนโครงสร้างวัสดุเล็กน้อยซึ่งประกอบด้วยวัสดุแคโทดชนิดต่างๆ ดังนี้ Lithium Cobalt Oxide (LiCoO_2), Lithium Manganese Oxide (LiMn_2O_4), Lithium Iron Phosphate (LiFePO_4), Lithium Nickel–Manganese–Cobalt Oxide (LiNiMnCoO_2), Lithium Nickel Cobalt Aluminium Oxide (LiNiCoAlO_2) และ Lithium Titanate ($\text{Li}_4\text{Ti}_5\text{O}_{12}$)

2.2.1.1. แบตเตอรี่ลิเทียมโคบอลต์ออกไซด์

แบตเตอรี่ลิเทียมโคบอลต์ออกไซด์ Lithium Cobalt Oxide (LiCoO_2) ถูกสร้างในปี 1991

เนื่องจากมีพลังงานไฟฟ้าที่สูงจึงถูกเลือกให้ใช้ในโน้ตบุ๊ก แท็บเล็ต และกล้อง แต่มีขีดจำกัดในการเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

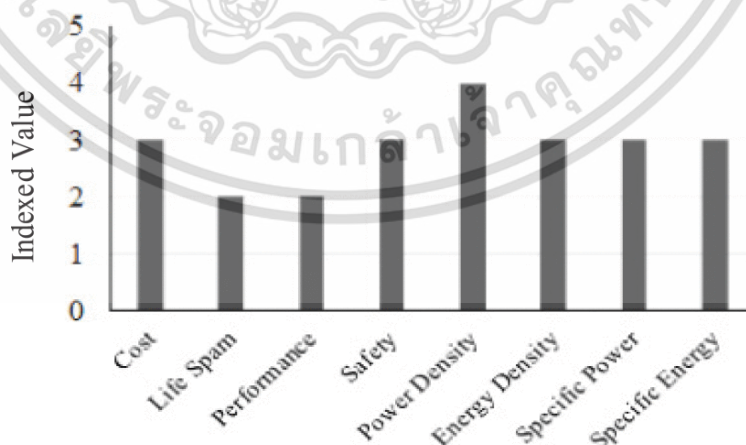
เข้าถึงโคบอลต์ทำให้มีราคาสูง มีอายุการใช้งานต่ำ ความจุที่จำกัด ไม่สามารถอัดหรือคายประจุที่กระแสออกช่วงได้ ต้องการอุปกรณ์ป้องกันอุณหภูมิเกินเมื่อทำการอัดประจุอย่างรวดเร็ว โดยอัตราการอัดและคายประจุถูกจำกัดในช่วงประมาณ 1C



รูปที่ 2.1 สมรรถนะของแบตเตอรี่ลิเทียมโคบอลต์ออกไซด์ [18]

2.2.1.2. แบตเตอรี่ลิเทียมแมงกานีสออกไซด์

แบตเตอรี่ลิเทียมแมงกานีสออกไซด์ Lithium Manganese Oxide (LiMn_2O_4) ถูกสร้างในปี 1994 มีโครงสร้างเป็นรูปเพชรทำให้มีความสามารถด้านกระแสที่สูงขึ้นและความต้านทานภายในที่ต่ำลงทำให้การอัดและคายประจุได้อย่างรวดเร็ว มีความสามารถในการทนต่อความร้อนสูงและปลอดภัย แต่อย่างไรก็ตามมีอายุการใช้งานที่ต่ำ มีความจุแบตเตอรี่ที่น้อยกว่าแบบโคบอลต์ถึง 33% แต่ให้พลังงานไฟฟ้าที่สูงกว่าแบบนิกเกิลประมาณ 50% มีขีดจำกัดอยู่ที่ 100 – 120 mAh/g มีการสูญเสียความจุแบตเตอรี่ระหว่างการอัดและคายประจุเพราะว่าแมงกานีสเกิดการสลายตัวของแมงกานีสอย่างมากในอิเล็กโทรไลต์เมื่ออุณหภูมิสูงขึ้น

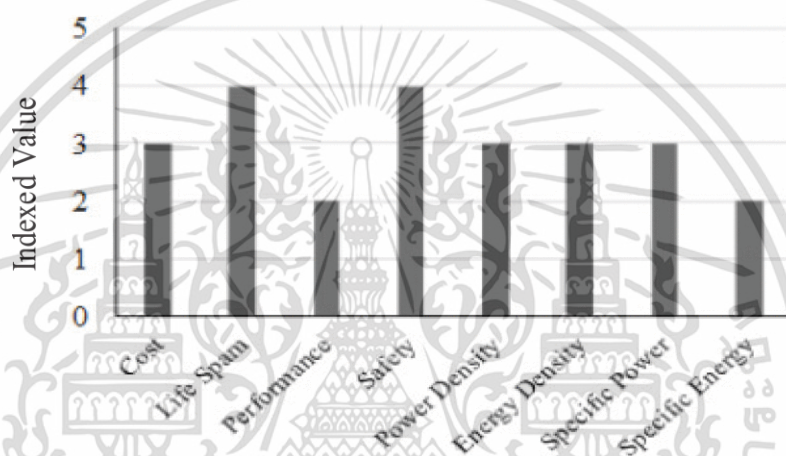


รูปที่ 2.2 สมรรถนะแบตเตอรี่ลิเทียมแมงกานีสออกไซด์ [18]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1.3. แบตเตอรี่ลิเทียมไอรอนฟอสเฟต

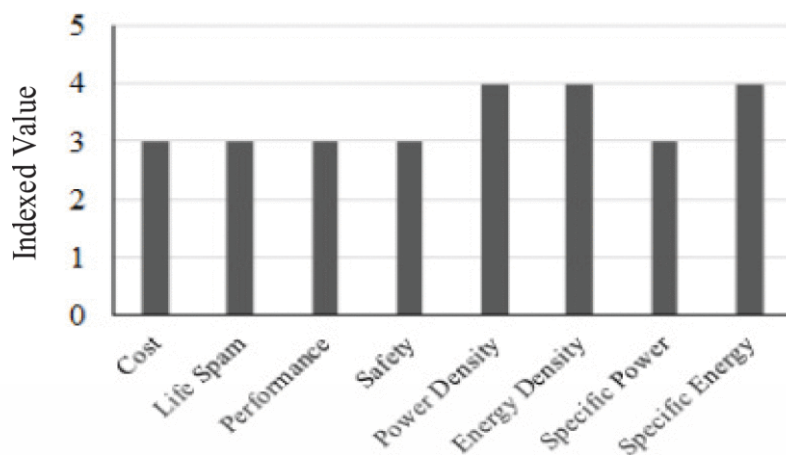
แบตเตอรี่ลิเทียมไอรอนฟอสเฟต Lithium Iron Phosphate (LiFePO_4) ถูกค้นพบในปี 1996 ชั่วแคว่โตจากฟอสเฟตคงที่ในสถานะอัดประจุเกิน (Overcharge) และสามารถทนต่ออุณหภูมิได้โดยไม่พังทลาย ทำให้เชื่อถือได้และปลอดภัยกว่าชั่วแคว่โตอื่นๆ สามารถใช้งานได้ภายใต้อุณหภูมิตั้งแต่ -30°C ถึง $+60^\circ\text{C}$ ที่ขจัดปัญหาด้านอุณหภูมิและป้องกันการเกิดไฟลุกไหม้ได้ โดยรวมแล้วมีค่าความต้านทานต่ำ อายุการใช้งานสูง สามารถรับมือกับภาระทางไฟฟ้าที่สูงได้ มีความปลอดภัย อุณหภูมิคงที่ ไม่มีผลกระทบต่อพิษ และราคาถูก ได้รับกระทบน้อยเมื่อทำการอัดประจุหรือคายประจุมากเกินไป มีความสามารถในการให้พลังงานไฟฟ้าและแรงดันที่ 160 mAh/g และ 3.40 ตามลำดับ



รูปที่ 2.3 สมรรถนะแบตเตอรี่ลิเทียมไอรอนฟอสเฟต [18]

2.2.1.4. แบตเตอรี่ลิเทียมนิกเกิลแมงกานีสโคบอลต์ออกไซด์

แบตเตอรี่ลิเทียมนิกเกิลแมงกานีสโคบอลต์ออกไซด์ Lithium Nickel Manganese Cobalt Oxide (LiNiMnCoO_2) ปัจจุบันบริษัทแบตเตอรี่มุ่งเน้นในการทำชั่วแคว่โตจากการผสมนิกเกิลแมงกานีสและโคบอลต์ (NMC) ถูกผลิตเพื่อการใช้งานแบบพลังงานไฟฟ้าสูงหรือกำลังไฟฟ้าสูง ในการผสมนิกเกิล 33% แมงกานีส 33% และโคบอลต์ 34% เป็นการผสมแบบใหม่ที่ทำให้ราคาของวัสดุดิบต่ำลงเพราะการใช้โคบอลต์ที่น้อยลง แบตเตอรี่ชนิดนี้กำลังเป็นที่ต้องการสำหรับการใช้ในยานยนต์ไฟฟ้าเนื่องมาจากการที่มีระดับพลังงานไฟฟ้าที่สูงและอัตราการปล่อยความร้อนต่ำ



รูปที่ 2.4 สมรรถนะแบตเตอรี่ลิเทียมนิกเกิลแมงกานีสโคบอลต์ออกไซด์ [18]

2.2.1.5. แบตเตอรี่ลิเทียมนิกเกิลโคบอลต์อะลูมิเนียมออกไซด์

แบตเตอรี่ลิเทียมนิกเกิลโคบอลต์อะลูมิเนียมออกไซด์ Lithium Nickel Cobalt Aluminium Oxide (LiNiCoAlO₂) หรือ NCA เป็นส่วนน้อยในส่วนแบ่งตลาดโลก อุตสาหกรรมยานยนต์กำลังเน้นการผลิตแบตเตอรี่ชนิดนี้เพราะว่าคุณสมบัติที่สูงเช่นระดับพลังงานไฟฟ้าและกำลังไฟฟ้าที่สูง อายุการใช้งานสูงเมื่อพิจารณาราคาและความปลอดภัย

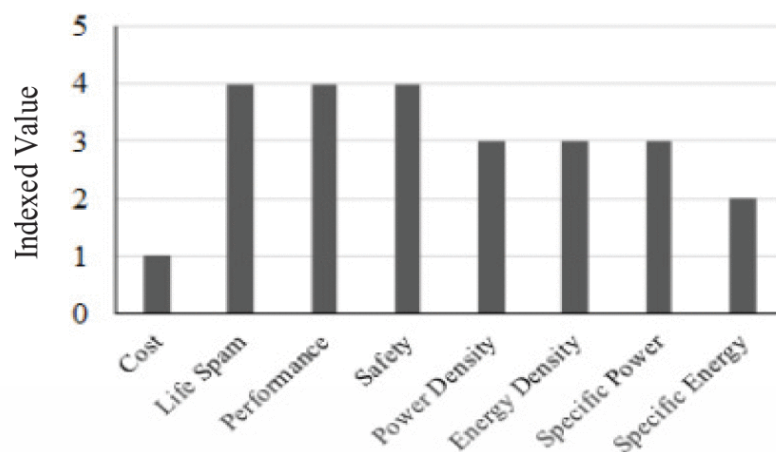


รูปที่ 2.5 สมรรถนะแบตเตอรี่ลิเทียมนิกเกิลโคบอลต์อะลูมิเนียมออกไซด์ [18]

2.2.1.6. แบตเตอรี่ลิเทียมไททาเนต

แบตเตอรี่ลิเทียมไททาเนต Lithium Titanate (Li₄Ti₅O₁₂) มีการใช้งานตั้งแต่ช่วงยุคปี 1980 เป็นการใช้ขั้วแอโนดจากไททาเนตแทนแกรไฟต์ มีแรงดันมาตรฐาน 2.40 V มีความจุแบตเตอรี่สูง อัตราการอัดและคายประจุสูง อายุการใช้งานสูง สามารถใช้งานได้อย่างปลอดภัยและมีคุณสมบัติที่มหัศจรรย์ที่อุณหภูมิต่ำเนื่องจากระดับพลังงานไฟฟ้าที่ไม่สูง การพัฒนาและวิจัยมุ่งพัฒนาระดับพลังงานไฟฟ้าและลดราคาลงมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 สมรรถนะแบตเตอรี่ลิเทียมไททาเนต [18]

เมื่อเปรียบเทียบแบตเตอรี่ลิเทียมไอออนด้วยกันสามารถสรุปได้ว่าแบตเตอรี่ลิเทียมไททาเนต ดีที่สุดในด้านของความปลอดภัย ประสิทธิภาพ อายุการใช้งาน และราคา แต่ยังมีจุดด้อยในด้านความจุแบตเตอรี่และในกลุ่มของกำลังไฟฟ้าและพลังงานไฟฟ้า

ตารางที่ 2.1 เปรียบเทียบประสิทธิภาพและการใช้งานของแบตเตอรี่ลิเทียมไอออนต่างชนิด [18]

ชนิดแบตเตอรี่ลิเทียมไอออน	จุดเด่น	จุดด้อย	การใช้งาน
LiCoO ₂	พลังงานไฟฟ้าสูง	อายุการใช้งานและความปลอดภัยต่ำ	โทรศัพท์ กล้อง
LiMn ₂ O ₄	กำลังไฟฟ้า ความปลอดภัย อายุการใช้งาน	ประสิทธิภาพโดยรวมปานกลาง	ยานยนต์ไฟฟ้า การแพทย์
LiFePO ₄	ทนต่ออุณหภูมิ ปลอดภัย อายุการใช้งาน	ประสิทธิภาพต่ำในอุณหภูมิ ต่ำ แรงดันไฟฟ้าต่ำ	ยานยนต์ไฟฟ้า อุปกรณ์พกพา
LiNiMnCoO ₂	ประสิทธิภาพโดยรวมดี โดยเฉพาะพลังงานไฟฟ้า	ราคาสูง	ยานยนต์ไฟฟ้า แหล่งกักเก็บพลังงาน
LiNiCoAlO ₂	ความหนาแน่นพลังงานและกำลังไฟฟ้าสูง อายุการใช้งานยาว	ราคาสูง ความปลอดภัยต่ำ	ยานยนต์ไฟฟ้า รถไฟ
Li ₄ Ti ₅ O ₁₂	ช่วงอุณหภูมิการใช้งานสูง อายุการใช้งานและความปลอดภัยสูง	แรงดันต่ำ 2.4 V/เซลล์ ความหนาแน่นพลังงานไฟฟ้าต่ำ ยากต่อการผลิต	เทคโนโลยีนาโนขั้นสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3. ระบบจัดการแบตเตอรี่

ระบบจัดการแบตเตอรี่ (Battery Management System) หรือ BMS คืออุปกรณ์อิเล็กทรอนิกส์ที่มีหน้าที่ควบคุมการอัดและคายประจุแบตเตอรี่ ทำการปรับระดับให้แรงดันแต่ละเซลล์มีความสมดุล ฝ้าสังเกตกระแส แรงดัน อุณหภูมิ ป้องกันแบตเตอรี่จากข้อผิดพลาดต่างๆ ทั้งการอัดประจุเกิน คายประจุเกิน แรงดันเกิน กระแสเกิน หรือ การลัดวงจร และรวบรวมข้อมูล ในระหว่างการใช้งาน แบตเตอรี่อาจมีการประพุดิตัวที่แตกต่างกัน ดังนั้นจึงจำเป็นต้องฝ้าสังเกตตลอด ผลจากการฝ้าสังเกตสถานะของแบตเตอรี่อาจช่วยให้ระบบมีประสิทธิภาพจากการจัดการ ป้องกัน และการควบคุม สามารถสื่อสารสำหรับการเก็บข้อมูลได้ การอัดและคายประจุหลายๆรอบ อาจทำให้เกิดความไม่เท่ากันของแรงดันแบตเตอรี่ในแต่ละเซลล์ เพราะคุณลักษณะทางกายภาพของแบตเตอรี่ที่เปลี่ยนไป เนื่องจากการผลิต อุณหภูมิ และอายุของเซลล์ ระดับแรงดันที่ไม่เท่ากันอาจทำให้ประสิทธิภาพและความทนทานของระบบก็เก็บพลังงานไฟฟ้าลดลง การอัดประจุที่มากเกินไปอาจทำให้เซลล์แบตเตอรี่ระเบิดได้ในทางตรงกันข้ามการอัดประจุที่ต่ำเกินไปอาจทำความเสียหายต่อคุณสมบัติทางเคมีของแบตเตอรี่และทำให้อายุการใช้สั้นลง [18]

2.4. ระดับประจุแบตเตอรี่

ระดับประจุแบตเตอรี่ (State of Charge) หรือ SoC คือตัวแปรที่อธิบายถึงพลังงานไฟฟ้าที่เหลืออยู่ในแบตเตอรี่ค่าระดับประจุแบตเตอรี่ไม่สามารถวัดได้โดยตรง แต่จะประมาณได้จากการวัดค่าที่มีความสัมพันธ์เช่น แรงดัน กระแส หรืออุณหภูมิ โดยส่วนมากค่าระดับประจุแบตเตอรี่จะอยู่ในรูปของเปอร์เซ็นต์ โดยเป็นอัตราส่วนระหว่างความจุไฟฟ้าที่เหลืออยู่เทียบกับความจุไฟฟ้าสูงสุดของแบตเตอรี่ตามสมการที่ (2.1) [5]

$$SOC = \frac{Q_{available}}{Q_{rated}} \quad (2.1)$$

โดยที่	SOC	คือ	ระดับประจุแบตเตอรี่
	$Q_{available}$	คือ	ความจุไฟฟ้าที่เหลืออยู่
	Q_{rated}	คือ	ความจุไฟฟ้าสูงสุด

ระดับประจุแบตเตอรี่สามารถเขียนเป็นสมการของเวลาและกระแสได้ตามสมการที่(2.2) [12]

$$SOC(t) = SOC(0) - \frac{1}{C_n} \int_0^t I(t) dt \quad (2.2)$$

โดยที่	$SOC(t)$	คือ	ระดับประจุแบตเตอรี่ที่เวลา t
	$SOC(0)$	คือ	ระดับประจุแบตเตอรี่เริ่มต้น
	C_n	คือ	ความจุไฟฟ้าที่ปกติ
	$I(t)$	คือ	กระแสไฟฟ้าที่เวลา t

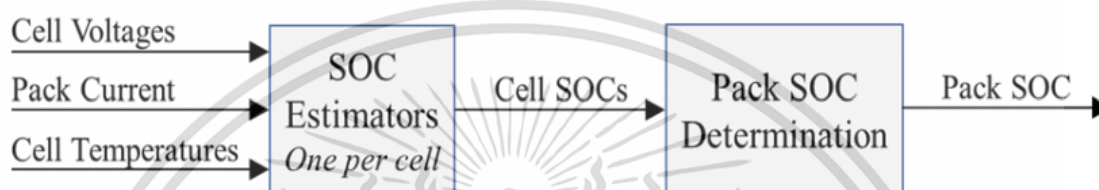
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1. การทำนายระดับประจุของกลุ่มแบตเตอรี่

การทำนายระดับประจุของกลุ่มแบตเตอรี่ต้องพิจารณาค่าต่างๆของกลุ่มแบตเตอรี่โดยที่ไม่ใช่เป็นการเพิ่มภาระให้กับการคำนวณใน BMS และเพิ่มความแม่นยำโดยการรวมเซลล์เข้าด้วยกัน

2.4.1.1. การทำนายแต่ละเซลล์

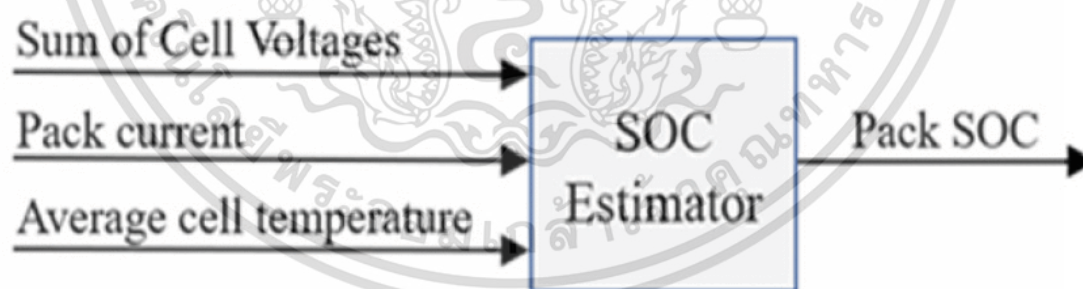
ระดับประจุแบตเตอรี่ของการทำนายแต่ละเซลล์โดยค่าระดับประจุต่ำสุดที่สุดจะใช้แทนระดับประจุของกลุ่มแบตเตอรี่ระหว่างทำการคายประจุและค่าประจุของเซลล์สูงสุดจะใช้แทนระดับประจุของกลุ่มแบตเตอรี่ระหว่างทำการอัดประจุ แต่เนื่องจากการทำนายระดับประจุของทุกเซลล์จะใช้ทรัพยากรในการคำนวณมากจึงไม่ค่อยเหมาะกับการใช้งานจริง



รูปที่ 2.7 การทำนายระดับประจุกลุ่มแบตเตอรี่แบบแต่ละเซลล์ [19]

2.4.1.2. การทำนายโดยรวมเซลล์

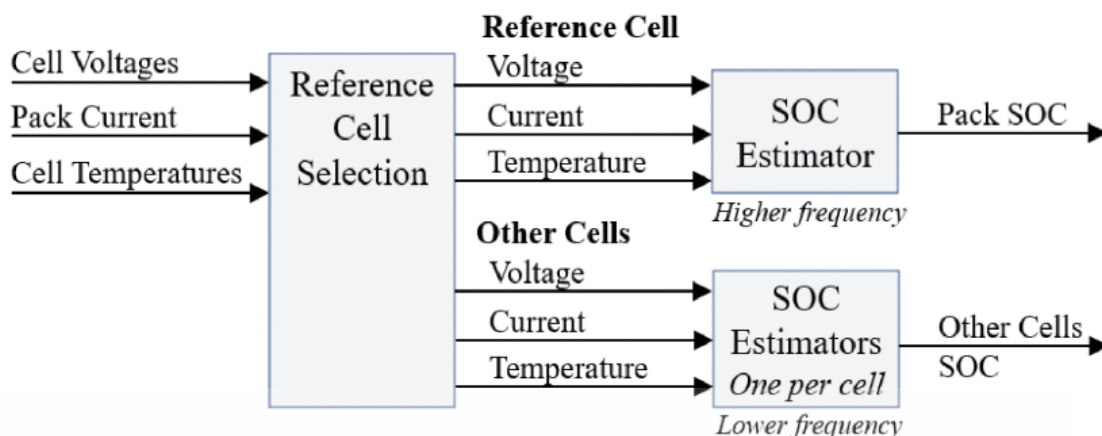
ถ้าเซลล์ของกลุ่มแบตเตอรี่มีคุณลักษณะที่เหมือนกันจึงเหมาะสมที่จะคิดว่ากลุ่มแบตเตอรี่เป็นก้อนเดียวกันโดยทำนายระดับประจุแบตเตอรี่จากแรงดันของกลุ่มแบตเตอรี่และกระแสของกลุ่มแบตเตอรี่ข้อดีคือความง่ายและเหมาะกับระดับประจุแบตเตอรี่ที่แต่ละเซลล์มีความแตกต่างกันไม่มาก แต่ทำให้เซลล์ที่ต่ำสุดมีอายุการใช้งานที่ต่ำลงได้และการทำนายที่ผิดพลาดถ้าคุณลักษณะของเซลล์แตกต่างกันมาก



รูปที่ 2.8 การทำนายระดับประจุกลุ่มแบตเตอรี่แบบรวมเซลล์ [19]

2.4.1.3. การทำนายโดยเลือกเซลล์อ้างอิง

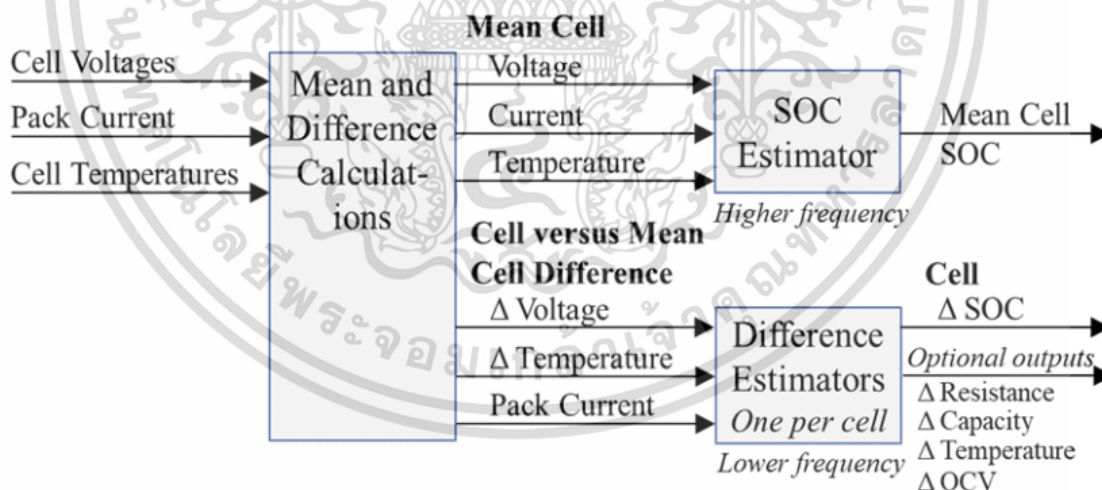
การทำนายโดยเลือกเซลล์อ้างอิงจะการทำนายที่ใช้ความถี่ของข้อมูลสูงกว่าเพื่อความแม่นยำที่สูงกว่าสำหรับเซลล์ที่เลือกและเซลล์ที่เหลือใช้ความถี่ของข้อมูลที่น้อยกว่า เพื่อให้การทำนายระดับประจุกลุ่มแบตเตอรี่ไม่จำเป็นต้องใช้ประสิทธิภาพสูง โดยปกติเซลล์ที่ถูกใช้อ้างอิงจะเป็นเซลล์ที่แรงดันต่ำสุดเวลาคายประจุและเซลล์ที่สูงสุดเวลาอัดประจุ



รูปที่ 2.9 การทำนายระดับประจุกลุ่มแบตเตอรี่โดยเลือกเซลล์อ้างอิง [19]

2.4.1.4. การทำนายโดยเซลล์เฉลี่ยและค่าความแตกต่าง

เซลล์เฉลี่ยทำนายโดยค่าเฉลี่ยของแรงดันทั้งหมดและความแตกต่างของระดับประจุแบตเตอรี่ของแต่ละเซลล์เทียบกับค่าเฉลี่ยได้มาจากความแตกต่างของแรงดันกับแรงดันเฉลี่ยและอุณหภูมิกับอุณหภูมิเฉลี่ย โดยเซลล์เฉลี่ยจะใช้ความถี่ของข้อมูลที่สูงและความถี่ของข้อมูลต่ำจะใช้สำหรับค่าความแตกต่างจากนั้นหาค่าระดับประจุแบตเตอรี่ของกลุ่มแบตเตอรี่จากค่าเฉลี่ยและค่าความแตกต่างโดยค่าระดับประจุต่ำสุดที่สุดจะใช้แทนระดับประจุของกลุ่มแบตเตอรี่ระหว่างทำการคายประจุและค่าประจุของเซลล์สูงสุดจะใช้แทนระดับประจุของกลุ่มแบตเตอรี่ระหว่างทำการอัดประจุ สามารถเพิ่มความแตกต่างของความต้านทานภายในและแรงดันเปิดวงจร



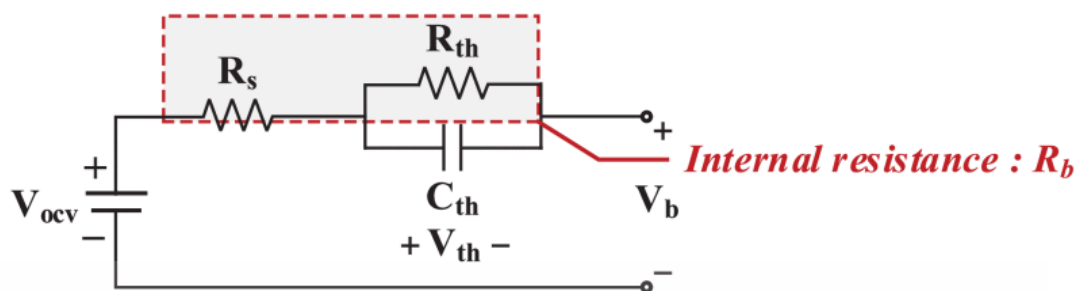
รูปที่ 2.10 การทำนายระดับประจุกลุ่มแบตเตอรี่โดยเซลล์เฉลี่ยและความแตกต่าง [19]

2.5. ค่าความต้านทานภายในแบตเตอรี่

ความต้านทานภายในแบตเตอรี่ ถูกแสดงโดยโมเดล Thevenin ดังรูปที่ 2.11 ประกอบด้วยแรงดันเปิดวงจร (V_{ocv}) แรงดันที่ขั้วแบตเตอรี่ (V_b) ตัวเก็บประจุ (C_{th}) และความต้านทานภายในแบตเตอรี่ (R_b) ซึ่งประกอบไปด้วย Ohmic Resistance (R_s) และ Polarization Resistance (R_{th})

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการหาความต้านทานภายในแบตเตอรี่สามารถหาได้จากการวัดแรงดันเปิดวงจรกับแรงดันขณะจ่ายโหลดและกระแส นำมาคำนวณจะได้ความต้านทานภายในตามสมการที่ (2.3) [7]



รูปที่ 2.11 Thevenin model ของแบตเตอรี่ [7]

$$V_b = V_{ocv} \pm I_b \times R_b \quad (2.3)$$

โดยที่	V_b	คือ	แรงดันที่ขั้วแบตเตอรี่
	V_{ocv}	คือ	แรงดันเปิดวงจร
	I_b	คือ	กระแสอัดหรือคายประจุ
	R_b	คือ	ความต้านทานภายในแบตเตอรี่

2.6. โมเดลปัญญาประดิษฐ์

ปัญญาประดิษฐ์ คือระบบที่สามารถรับรู้ วิเคราะห์ สรุป และจำลองข้อมูล โดยใช้เครื่องประมวลผล เป็นการรวมกันของศาสตร์ทางด้าน คณิตศาสตร์, การหาค่าเหมาะสมที่สุด (Optimization) และลอจิก โดยมีรากฐานมาจาก สถิติ ความน่าจะเป็น และอื่นๆ โดยระบบจะมีการเรียนรู้ที่เรียกว่า การเรียนรู้ของเครื่อง (Machine Learning) เป็นโมเดลที่สร้างมาจากข้อมูลตัวอย่างที่เรียกว่า ข้อมูลฝึกสอน (Training Data) เพื่อทำการทำนายหรือตัดสินใจ โดยสามารถแบ่งเป็น 1) การเรียนรู้แบบมีผู้สอน (Supervised Learning) หมายถึง คอมพิวเตอร์จะถูกสอนด้วยข้อมูลตัวอย่างและผลลัพธ์ มีจุดประสงค์เพื่อหากฎหรือความเกี่ยวข้องกันระหว่างข้อมูลขาเข้าและผลลัพธ์ 2) การเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) ข้อมูลที่ไม่ได้ถูกกำหนดถูกส่งให้คอมพิวเตอร์เรียนรู้ด้วยตัวเองเพื่อหาความเกี่ยวข้องกันของข้อมูลขาเข้า และ 3) การเรียนรู้ด้วยตัวเอง (Reinforcement Learning) เป็นการเรียนรู้ที่คอมพิวเตอร์ทำการโต้ตอบกับระบบต่างๆ ในสิ่งแวดล้อมเพื่อบรรลุเป้าหมายบางอย่าง เช่น การขับรถ หรือ เล่นเกมแข่งกับคู่ต่อสู้ เพื่อทำลายปัญหาต่างๆ ไปเรื่อยๆ จนได้วิธีที่ดีที่สุด

2.6.1. Recurrent Neural Network

Recurrent Neural Network คือ โมเดลที่มีการป้อนข้อมูล 2 ทิศทางหรือ Bi-directional หมายความว่าผลลัพธ์ของโหนดใดๆจะส่งผลข้อมูลขาเข้าของโหนดนั้นๆ เป็นโมเดลที่โดดเด่นเพราะมีหน่วยความจำ เหมาะกับการทำนายข้อมูลที่เป็นลำดับหรือเวลา แต่มีปัญหาที่ เมื่อโมเดลไม่มีการเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรียนรู้ Vanishing Gradient เมื่อโมเดลเรียนรู้ไปเรื่อยๆแล้ว gradient หรือ การปรับน้ำหนักเทียบต่อค่าความผิดพลาด เล็กเกินไปทำให้ น้ำหนักในโมเดลเปลี่ยนแปลงจนไม่มีนัยสำคัญ กับ โมเดลไม่สามารถหาค่าได้ Exploding Gradient เมื่อ gradient ใหญ่เกินไปทำให้โมเดลมีขนาดน้ำหนักที่มากเกินไปจนไม่สามารถหาค่าได้

2.6.2. Long Short-term Memory

Long Short-term Memory คือ โมเดล RNN ที่แก้ปัญหา Vanishing Gradient และ Exploding Gradient เมื่อผ่านไปเป็นระยะเวลาอันยาวนาน มีโครงสร้างตามรูปที่ 2.12 และตามสมการที่ (2.4)-(2.8) โดยที่ x_k คือข้อมูลเข้า h_k คือข้อมูลขาออก c_k คือหน่วยความจำ i_k, f_k , และ o_k คือ Activation Vector ของ Input Gate, Forget Gate และ Output Gate ตามลำดับ σ_g, σ_c และ σ_h Activation Function หรือการคำนวณข้อมูลออกจากโหนด W, U, b คือน้ำหนัก และ \odot คือ Hadamard product โดยโมเดลจะทำการปรับ Forget, Input และ Output Gate ทำให้โมเดลไม่มีปัญหาเมื่อทำการทำนายในระยะยาว

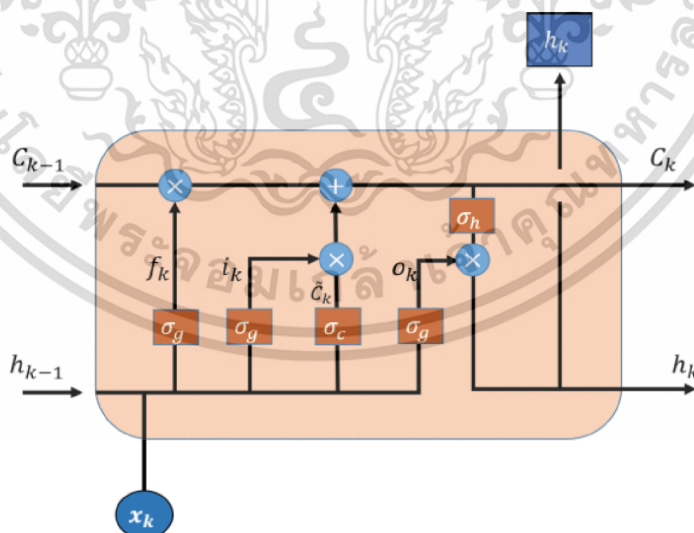
$$f_k = \sigma_g (W_f x_k + U_f h_{k-1} + b_f) \quad (2.4)$$

$$i_k = \sigma_g (W_i x_k + U_i h_{k-1} + b_i) \quad (2.5)$$

$$o_k = \sigma_g (W_o x_k + U_o h_{k-1} + b_o) \quad (2.6)$$

$$c_k = f_k \odot c_{k-1} + i_k \odot \sigma_c (W_c x_k + U_c h_{k-1} + b_c) \quad (2.7)$$

$$h_k = o_k \odot \sigma_h (c_k) \quad (2.8)$$



รูปที่ 2.12 โครงสร้าง Long Short-term Memory [15]

2.6.3. Gated Recurrent Unit

Gated Recurrent Unit คือ LSTM ที่มีตัวแปรน้อยกว่าคือไม่มี Output Gate โดยมีโครงสร้างตามรูปที่ 2.13 และตามสมการที่ (2.9)-(2.12) z_t คือ Update Gate ทำหน้าที่กำหนดข้อมูลเอกสารนี้เป็นเอกสารที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

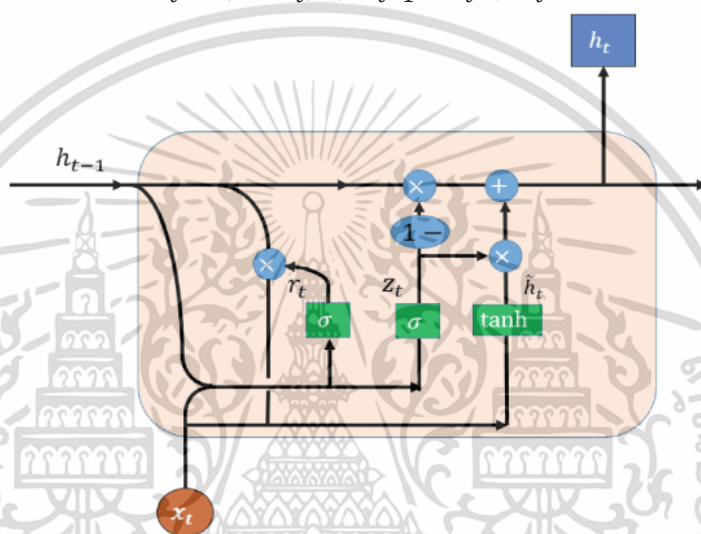
ที่จะทำการอัปเดต r_t คือ Reset Gate ทำหน้าที่กำหนดจำนวนข้อมูลที่จะทำการลืมนำไป เมื่อรวมกับผลลัพธ์จากขั้นตอนก่อนผ่าน \tanh activation function จะได้ตัวแทน h_t vector แล้วนำมารวมกับ Update Gate ของข้อมูลขั้นก่อนหน้าเพื่อกำหนดว่าจะใช้หรือจะลืมนำไป เมื่อรวมกันทั้งหมดจะได้ข้อมูลผลลัพธ์ออกมา

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \quad (2.9)$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \quad (2.10)$$

$$\hat{h}_t = \tanh(W_h x_t + U_h (r_t \odot h_{t-1}) + b_h) \quad (2.11)$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \hat{h}_t \quad (2.12)$$



รูปที่ 2.13 โครงสร้าง Gated Recurrent Unit [16]

2.6.4. การหาค่าความผิดพลาด

การหาค่าความผิดพลาด (Error) มีไว้เพื่อประเมินความแม่นยำของโมเดลปัญญาประดิษฐ์ที่ได้ทำการสร้างและสามารถหาความผิดพลาดของข้อมูลที่น่ามาทดสอบได้โดยสำหรับการหาค่าผิดพลาดด้วยวิธีค่าสัมบูรณ์เฉลี่ยของค่าความผิดพลาด (Mean absolute error) หรือ MAE ตามสมการที่(2.13) และค่ารากที่สองของค่าเฉลี่ยกำลังสองของค่าความผิดพลาด (Root mean squared error) หรือ RMSE ตามสมการที่ (2.14) มีความสะดวกในการตีความเพราะค่าความผิดพลาดไม่มีเลขยกกำลัง 2

$$MAE = \frac{\sum_{i=1}^n |Y_i - \hat{Y}_i|}{n} \quad (2.13)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{n}} \quad (2.14)$$

โดยที่ Y_i	คือ	ค่าจริงของตัวแปร
\hat{Y}_i	คือ	ค่าทำนายของตัวแปร
n	คือ	จำนวนตัวแปร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7. การสื่อสารข้อมูลผ่านพอร์ตอนุกรมแบบอะซิงโครนัส

การสื่อสารข้อมูลผ่านพอร์ตอนุกรมแบบอะซิงโครนัส หรือ Universal Asynchronous Receiver and Transmitter (UART) คือการสื่อสารชนิดหนึ่งที่จะทำการส่งข้อมูลทีละบิต (bit) จากหลักน้อยที่สุดไปมากที่สุด โดยจะมีบิตที่กำหนดจุดเริ่มต้นและจุดสิ้นสุด โดยที่ตัวส่งจะนำข้อมูลหลายไบต์มาส่งทีละบิต ที่ตัวรับข้อมูลจะประกอบบิตต่างๆเข้ากันเป็นไบต์ โดยการสื่อสารสามารถแบ่งได้ 3 แบบ 1) แบบทางเดียว (Simplex) 2) อุปกรณ์ทั้ง 2 ส่งและรับพร้อมกัน (Full Duplex) 3) ผลัดกันรับและส่ง (Half Duplex) ในการใช้งาน UART ทั้งตัวรับและส่งต้องมีค่าเหล่านี้ที่เหมือนกันคือ อัตราบอด (Baud Rate) แพริตีบิตหรือบิตตรวจสอบ (Parity Bit) ขนาดบิตข้อมูล และขนาดบิตตัวหยุด โดยทั่วไปจะมีค่าเหล่านี้อยู่ที่ ข้อมูลยาว 8 บิต ไม่มี Parity Bit และ บิตหยุด 1 บิต หรือเรียกว่า 8N1 องค์ประกอบของข้อมูล 1 ชุดประกอบไปด้วย 1) บิตที่ไม่ได้ใช้งาน (Idle) มีแรงดันไฟฟ้าสูง (Logic High) 2) บิตเริ่มต้น มีแรงดันไฟฟ้าต่ำ (Logic Low) คือบิตที่บอกว่าเป็นจุดเริ่มต้น 3) บิตข้อมูล มีความยาว 5 – 9 บิต 4) Parity Bit ใช้ตรวจสอบว่ามีการเปลี่ยนแปลงของข้อมูลหรือไม่ โดยสามารถอธิบายได้โดยตัวเลขเป็นเลขคู่หรือคี่ และ 5) บิตหยุดเป็น Logic High เป็นสัญญาณบอกว่าข้อมูลที่ส่งหมดแล้ว เนื่องจากบิตเริ่มเป็น Logic Low และ บิตหยุดเป็น Logic High แสดงว่าทุกๆการส่งข้อมูล จะมีการเปลี่ยนสัญญาณ 2 ครั้งซึ่งสามารถนำมาใช้ตรวจสอบว่า เมื่อ Logic Low แสดงนานกว่าเวลา ความยาวของข้อมูลแสดงว่าไม่มีการส่งข้อมูล (Break Condition) ที่สามารถตรวจจับได้

2.8. การเตรียมข้อมูล

การเตรียมข้อมูล (Data Preprocessing) คือกระบวนการจัดการข้อมูลที่สำคัญต่อการใช้งานกับระบบปัญญาประดิษฐ์เพื่อเพิ่มประสิทธิภาพความแม่นยำ เป็นขั้นตอนที่ใช้ในตอนแรกสุดของการใช้งานกับระบบปัญญาประดิษฐ์มีเครื่องมือและวิธีที่หลากหลายในการเตรียมข้อมูลเช่นการ แบ่งข้อมูล แปลงข้อมูล ลบข้อมูลที่ไม่ต้องการ เพิ่มข้อมูลที่ตกหล่น การทำให้อยู่ในบรรทัดฐานเดียวกัน หรือ การคัดเลือกลักษณะ

2.8.1. การแบ่งข้อมูล

ในการสร้างโมเดลปัญญาประดิษฐ์โดยปกติจะทำการแบ่งชุดข้อมูลเป็น 3 ชุด 1) ชุดข้อมูลการ สอน (Training) คือชุดข้อมูลที่ใช้ในการสอนโมเดลปัญญาประดิษฐ์ประกอบด้วยข้อมูลที่ใช้สอนและ ผลลัพธ์โดยโมเดลปัญญาประดิษฐ์จะทำการปรับค่าน้ำหนักเพื่อให้ได้ความสัมพันธ์ตามชุดข้อมูลสอน 2) เมื่อโมเดลปัญญาประดิษฐ์ทำการสอนแล้วจะใช้ชุดข้อมูลตรวจสอบ (Validation) เพื่อทำการ สังเกตผลลัพธ์ที่ได้จากการทำนายว่าเกิดการเรียนรู้ที่มากเกินไป (Over-fitting) หรือน้อยเกินไป (Under-fitting) และปรับแต่งค่าพารามิเตอร์ของโมเดลปัญญาประดิษฐ์ (Hyperparameter) และ 3) ชุด ข้อมูลตรวจสอบ (Test) ใช้ประเมินผลลัพธ์จากโมเดลปัญญาประดิษฐ์ที่ได้ทำการปรับแต่งสุดท้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.2. การทำให้อยู่ในบรรทัดฐานเดียวกัน

การทำให้อยู่ในบรรทัดฐานเดียวกัน (Normalization) คือการปรับค่าของข้อมูลที่อยู่ในช่วงมาตรฐานที่แตกต่างกันให้มาอยู่ที่ช่วงมาตรฐานเดียวกันโดยไม่ทำให้เกิดการบิดเบือนของข้อมูลหรือสูญเสียข้อมูล ทำให้โมเดลปัญญาประดิษฐ์สามารถเรียนรู้ความสัมพันธ์ระหว่างตัวแปรแตกต่างกันได้ สำหรับวิธีทำให้อยู่ในบรรทัดฐานเดียวกันตามช่วงที่ต้องการ (Min-Max Scaling) เป็นไปตามสมการที่ (2.15) และสมการที่ (2.16) [21]

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (2.15)$$

$$x_{scaled} = (x_{norm} \times (max - min)) + min \quad (2.16)$$

โดยที่	x	คือ	ค่าก่อนทำให้อยู่ในบรรทัดฐานใหม่
	x_{norm}	คือ	ค่าที่อยู่ในบรรทัดฐานใหม่
	x_{min}	คือ	ค่าต่ำสุดของข้อมูล
	x_{max}	คือ	ค่าสูงสุดของข้อมูล
	x_{scaled}	คือ	ค่าที่อยู่ในบรรทัดฐานใหม่ในช่วงที่ต้องการ
	max	คือ	ค่าสูงสุดของช่วงบรรทัดฐานที่ต้องการ
	min	คือ	ค่าต่ำสุดของช่วงบรรทัดฐานที่ต้องการ

2.8.3. การคัดเลือกคุณลักษณะ

การคัดเลือกคุณลักษณะ (Feature Selection) คือกระบวนการคัดเลือกคุณลักษณะที่ใช้ในการสร้างโมเดลปัญญาประดิษฐ์โดยเป็นการคัดเลือกคุณลักษณะที่มีความสำคัญ กำจัดคุณลักษณะที่ซ้ำซ้อนและไม่เกี่ยวข้อง เพื่อลดความซับซ้อน ลดระยะเวลาการสอน และเพิ่มความแม่นยำของโมเดลปัญญาประดิษฐ์ โดยวิธี Pearson correlation coefficient เป็นหนึ่งในการวิธีการคัดเลือกคุณลักษณะเพื่อหาสัมประสิทธิ์ของความสัมพันธ์ระหว่าง คุณลักษณะ (Feature) และ ผลลัพธ์ (Output) มีค่าอยู่ระหว่าง -1 ถึง 1 ค่าที่ยิ่งเข้าใกล้ 1 หมายความว่าคุณลักษณะและผลลัพธ์มีความสัมพันธ์ไปในทิศทางเดียวกัน ค่าที่ยิ่งเข้าใกล้ -1 หมายความว่าคุณลักษณะและผลลัพธ์มีความสัมพันธ์ไปในทิศทางตรงข้ามกัน สามารถหาได้โดยสมการที่ (2.17)

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}} \quad (2.17)$$

โดยที่	r	คือ	ค่าสัมประสิทธิ์ของความสัมพันธ์
	x_i	คือ	ค่าของตัวแปร Feature
	\bar{x}	คือ	ค่าเฉลี่ยของตัวแปร Feature
	y_i	คือ	ค่าของตัวแปร Output
	\bar{y}	คือ	ค่าเฉลี่ยของตัวแปร Output

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

วิธีการดำเนินงานวิจัย

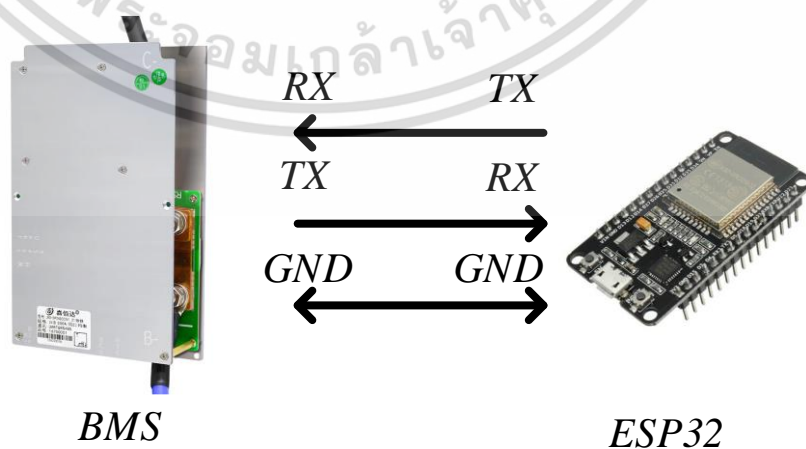
ในบทนี้จะนำเสนอการออกแบบงานวิจัยเพื่อการทำนายระดับประจุแบตเตอรี่โดยจะแบ่งการออกแบบเป็นการออกแบบการเก็บข้อมูลและรับส่งข้อมูล การออกแบบโมเดลปัญญาประดิษฐ์ และการออกแบบการทดสอบแบตเตอรี่

3.1. การออกแบบการเก็บรับ ส่ง และเตรียมข้อมูล

เนื่องจากงานวิจัยนี้การทำนายระดับประจุแบตเตอรี่กับกลุ่มแบตเตอรี่ ดังนั้นจึงต้องมีการเก็บรับและส่งข้อมูลจากระบบจัดการแบตเตอรี่ (Battery Management System: BMS) แต่เนื่องจากระบบจัดการแบตเตอรี่ไม่สามารถส่งข้อมูลค่าพารามิเตอร์ต่างๆ ออกมาเพื่อเก็บข้อมูลได้โดยตรง จึงต้องมีการออกแบบโปรแกรมคอมพิวเตอร์เพื่อทำการรับ ส่ง เก็บข้อมูลออกมา และทำขั้นตอนการเตรียมข้อมูลเพื่อให้ได้ถึงข้อมูลค่าพารามิเตอร์ที่จะนำไปการทำนายระดับประจุแบตเตอรี่

3.1.1. วิธีการเก็บรับและส่งข้อมูล

ในการรับ ส่ง และเก็บข้อมูลจะใช้ไมโครคอนโทรลเลอร์ชนิด ESP 32 ซึ่งมีความสามารถในการเชื่อมต่ออินเทอร์เน็ตเพื่อให้มีการส่งข้อมูลไปเก็บไว้บนออนไลน์ได้และมีหน้าที่ในการรับ ส่ง แปลงค่าไปมา ระบบจัดการแบตเตอรี่ที่ใช้ในงานวิจัยนี้เป็นรุ่น JBD-DP24S002 มีช่องทางการสื่อสาร UART สามารถต่อโดยตรงกับไมโครคอนโทรลเลอร์ได้ซึ่ง UART จะมีพอร์ตทั้งหมด 4 พอร์ต พอร์ต 1 เป็นช่องจ่ายแรงดันไฟฟ้า พอร์ต 2 เป็นช่องส่งข้อมูล (Transmit) หรือ Tx พอร์ต 3 เป็นช่องรับข้อมูล (Receive) หรือ Rx พอร์ต 4 เป็นกราวด์โดยจะใช้เพียงพอร์ตที่ 2, 3 และ 4 ไมโครคอนโทรลเลอร์จะส่งสัญญาณไปหาระบบจัดการแบตเตอรี่ ทาง Tx ที่ระดับแรงดัน 3.2 V และรับข้อมูลจากระบบจัดการแบตเตอรี่ทาง RX ที่ระดับแรงดัน 5 V โดยการต่อวงจรการรับส่งข้อมูลเป็นไปดังรูปที่ 3.1



รูปที่ 3.1 การเชื่อมต่อการสื่อสารชนิด UART

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับการอ่าน แปลง และส่งค่าทั้งหมดทำบนโปรแกรม Arduino IDE มีไว้เขียนโปรแกรม คอมพิวเตอร์สำหรับไมโครคอนโทรลเลอร์ ซึ่งใช้ภาษา C ในการเขียน ในงานวิจัยนี้โปรแกรม คอมพิวเตอร์เริ่มจากการเชื่อมต่ออินเทอร์เน็ตมีไว้เพื่อบันทึกวัน เวลา ต่อมาเป็นการส่งข้อมูลให้ BMS เพื่อให้ BMS ส่งข้อมูลพารามิเตอร์กลับมาแล้วจึงแปลงค่ารวบรวมข้อมูลพารามิเตอร์ทั้งหมดแล้วส่งขึ้น Google Sheet เพื่อไปเก็บบนออนไลน์

สำหรับการรับข้อมูลค่าพารามิเตอร์จาก BMS นั้นตัวไมโครคอนโทรลเลอร์จำเป็นต้องส่งข้อมูล ไปหา BMS เพื่อให้ BMS ตอบรับและส่งข้อมูลกลับมาโดยอ้างอิงจากข้อมูลผู้ผลิต BMS ที่ใช้ในงานวิจัยนี้ [20] ไมโครคอนโทรลเลอร์จะต้องส่งข้อความที่เป็นฐาน 16 โดยมีย่อยเป็น 2 ชุด ชุดที่ 1 เพื่อรับข้อมูลพื้นฐานทั่วไปทั้งหมด และชุดที่ 2 เพื่อรับข้อมูลของแต่ละเซลล์

โดยสำหรับชุดที่ 1 ต้องส่งข้อความว่า DD A5 03 00 FF FD 77 แล้ว BMS จะส่งข้อมูลกลับมาเป็นฐาน 16 เช่นกันทำให้ต้องแปลงค่ากลับมาเป็นฐาน 10 เพื่อให้สามารถอ่านเข้าใจได้ ซึ่งสำหรับชุดที่ 1 นี้ ข้อมูลที่ส่งกลับมาจะขึ้นต้นด้วย DD 03 มีตัวแปรได้แก่ แรงดันของกลุ่มแบตเตอรี่, กระแส, ค่าความเก็บประจุที่เหลือ, ค่าความเก็บประจุที่ระบุของแบตเตอรี่, จำนวนรอบการใช้งาน, สถานะการบาลานซ์แรงดัน, ขนาดประจุแบตเตอรี่ที่เหลือเป็นเปอร์เซ็นต์, จำนวนเซลล์ และอุณหภูมิ

ส่วนข้อมูลชุดที่ 2 ต้องส่งข้อความว่า DD A5 04 00 FF FC 77 แล้ว BMS จะส่งข้อมูลกลับมาเป็นฐาน 16 ขึ้นต้นด้วย DD 04 มีตัวแปรคือ แรงดันแต่ละเซลล์ของแบตเตอรี่ ข้อมูลฐาน 16 จาก BMS แสดงดังรูปที่ 3.2

```
Pack Data : DD 3 0 26 B 48 0 0 1 4F 1 4F 0 6 2D 1D 0 24 0 0 0 0 53 64 1 7 3 C 34 B E1 B E5 0 0 0 1 4F 1 4F 3 E3 FA F 77
Cell Data : DD 4 0 E 10 27 10 28 10 21 10 25 10 27 F E8 10 27 FD AA 77 1 0 0 0 62 28 0 0 2D 0 0 0 8C A4 D 80 D0 25 FB 3F 58 47 FC 3F
```

รูปที่ 3.2 ข้อมูลฐาน 16 จาก BMS

เมื่อได้รับข้อมูลที่แปลงเสร็จแล้วก็ส่งขึ้น Google Sheet โดยข้อมูลที่ส่งขึ้นไปได้แก่ วัน, เวลา, ค่าความจุเป็นเปอร์เซ็นต์, กระแส, แรงดัน, อุณหภูมิที่ 1 เป็นอุณหภูมิของ BMS, อุณหภูมิที่ 2 เป็นอุณหภูมิแบตเตอรี่, อุณหภูมิที่ 3 เป็นอุณหภูมิของสภาพแวดล้อม, แรงดันเซลล์ที่ 1, แรงดันเซลล์ที่ 2, แรงดันเซลล์ที่ 3, แรงดันเซลล์ที่ 4, แรงดันเซลล์ที่ 5, แรงดันเซลล์ที่ 6, แรงดันเซลล์ที่ 7, แรงดันเซลล์ที่สูงสุด, แรงดันเซลล์ที่ต่ำสุด, แรงดันเฉลี่ย, แรงดันที่แตกต่างกันของเซลล์สูงสุดและต่ำสุด, สถานะการบาลานซ์เซลล์ที่ 1, สถานะการบาลานซ์เซลล์ที่ 2, สถานะการบาลานซ์เซลล์ที่ 3, สถานะการบาลานซ์เซลล์ที่ 4, สถานะการบาลานซ์เซลล์ที่ 5, สถานะการบาลานซ์เซลล์ที่ 6, สถานะการบาลานซ์เซลล์ที่ 7, ค่าความเก็บประจุที่ระบุของแบตเตอรี่ และค่าความเก็บประจุที่เหลือ โดยชุดข้อมูลจะถูกส่งขึ้นมาทุกๆประมาณ 5 วินาที ดังรูปที่ 3.3

Timestamp	Date	Time	Capacity	Current	Voltage	Temp1 (Onboard)	Temp2 (Battery)	Temp3 (Ambient)	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Cell 7
2/18/2023 20:54:18	Saturday-Febru	20:54:18	12	1660	23.64	31.9	29.9	30.2	3375	3376	3381	3376	3372	3392	3372
2/18/2023 20:54:22	Saturday-Febru	20:54:22	12	1540	23.68	31.9	30.2	30	3380	3384	3388	3384	3377	3397	3380
2/18/2023 20:54:26	Saturday-Febru	20:54:26	12	1540	23.72	31.6	30.1	30.2	3387	3391	3394	3391	3384	3404	3387
2/18/2023 20:54:34	Saturday-Febru	20:54:30	12	1660	23.77	31.8	30.1	30.1	3392	3396	3401	3398	3389	3409	3392
2/18/2023 20:54:38	Saturday-Febru	20:54:34	12	1660	23.8	31.7	30	30	3397	3401	3403	3401	3394	3416	3397
2/18/2023 20:54:42	Saturday-Febru	20:54:38	12	1540	23.83	32	30.2	30	3401	3406	3408	3406	3399	3421	3402
2/18/2023 20:54:46	Saturday-Febru	20:54:42	12	1540	23.87	31.9	30	30	3406	3411	3414	3411	3404	3426	3407
2/18/2023 20:54:50	Saturday-Febru	20:54:46	12	1540	23.9	31.8	29.9	30.2	3411	3416	3418	3416	3409	3431	3412
2/18/2023 20:54:54	Saturday-Febru	20:54:50	12	1660	23.93	32	30.1	30	3416	3418	3423	3421	3414	3436	3417
2/18/2023 20:54:58	Saturday-Febru	20:54:54	12	1660	23.96	31.8	29.9	30.2	3421	3423	3428	3423	3419	3438	3420
2/18/2023 20:55:02	Saturday-Febru	20:54:58	13	1540	23.99	31.9	29.9	30.2	3426	3428	3433	3428	3422	3443	3424
2/18/2023 20:55:06	Saturday-Febru	20:55:03	13	1660	24.03	32	30.1	30.1	3431	3433	3438	3433	3429	3448	3431
2/18/2023 20:55:10	Saturday-Febru	20:55:06	13	1660	24.06	31.7	30	30.2	3433	3438	3440	3438	3431	3451	3434
2/18/2023 20:55:15	Saturday-Febru	20:55:11	13	1540	24.09	31.8	30.3	30.2	3438	3440	3445	3440	3436	3456	3439
2/18/2023 20:55:19	Saturday-Febru	20:55:15	13	1660	24.11	31.8	30.2	30.1	3441	3445	3446	3445	3441	3468	3441
2/18/2023 20:55:23	Saturday-Febru	20:55:19	13	1540	24.14	31.9	30.2	30	3443	3448	3453	3450	3444	3463	3446
2/18/2023 20:55:27	Saturday-Febru	20:55:23	13	1660	24.16	31.7	30.3	30	3448	3453	3453	3453	3446	3466	3449
2/18/2023 20:55:32	Saturday-Febru	20:55:27	13	1540	24.18	32	30	30	3451	3453	3458	3455	3451	3470	3451
2/18/2023 20:55:36	Saturday-Febru	20:55:32	13	1660	24.21	31.7	30.1	30.1	3453	3458	3460	3460	3454	3473	3454

รูปที่ 3.3 ตัวอย่างข้อมูลในงานวิจัยที่ถูกส่งขึ้น Google Sheet

3.1.2. การเตรียมข้อมูล

ก่อนทำการป้อนข้อมูลให้โมเดลปัญญาประดิษฐ์ทำการเรียนรู้หรือทำการทำนาย ต้องมีการเตรียมข้อมูลเริ่มจากการทำความสะอาดข้อมูล (Data Cleansing) เป็นการตัดข้อมูลที่ไมใช่งานออก แล้วจึงนำชุดข้อมูลที่ได้ทำความสะอาดแล้วมาเตรียมข้อมูล (Data Preprocessing) โดยเป็นการนำมาทำให้อยู่ในบรรทัดฐานเดียวกัน (Normalization) ตามสมการที่ (3.1) [14] แล้วทำให้อยู่ในช่วงที่ต้องการตามสมการที่ (3.2) [21] โดยในงานวิจัยนี้ให้อยู่ในช่วง 0 ถึง 1

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3.1}$$

$$x_{scaled} = (x_{norm} \times (max - min)) + min \tag{3.2}$$

- โดยที่ x คือ ค่าก่อนทำให้อยู่ในบรรทัดฐานใหม่
- x_{norm} คือ ค่าที่อยู่ในบรรทัดฐานใหม่
- x_{min} คือ ค่าต่ำสุดของข้อมูล
- x_{max} คือ ค่าสูงสุดของข้อมูล
- x_{scaled} คือ ค่าที่อยู่ในบรรทัดฐานใหม่ในช่วงที่ต้องการ
- max คือ ค่าสูงสุดของช่วงบรรทัดฐานที่ต้องการ
- min คือ ค่าต่ำสุดของช่วงบรรทัดฐานที่ต้องการ

ตัวอย่างชุดข้อมูลก่อนและหลังการทำให้อยู่ในบรรทัดฐานเดียวกันในงานวิจัยนี้แสดงดังรูปที่

รูปที่ 3.4 และ รูปที่ 3.5 ตามลำดับ

Time	Current	Voltage	Temp1 (Onboard Temp)	Temp2 (Battery temp)	Temp3 (Ambient temp)	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Cell 7
2023-02-20 19:56:01	1020.0	22.91	32.0	30.8	31.0	3281.0	3288.0	3293.0	3293.0	3294.0	3306.0	3277.0
2023-02-20 19:56:05	1540.0	23.38	32.0	30.9	30.9	3331.0	3339.0	3347.0	3342.0	3343.0	3363.0	3328.0
2023-02-20 19:56:09	1660.0	23.44	32.2	30.7	31.1	3339.0	3347.0	3357.0	3352.0	3350.0	3373.0	3335.0
2023-02-20 19:56:14	1660.0	23.50	32.0	31.1	30.9	3348.0	3357.0	3366.0	3362.0	3360.0	3380.0	3345.0
2023-02-20 19:56:18	1660.0	23.56	32.1	30.9	31.1	3358.0	3366.0	3374.0	3369.0	3367.0	3388.0	3355.0

รูปที่ 3.4 ตัวอย่างข้อมูลในงานวิจัยก่อนทำให้อยู่ในบรรทัดฐานเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

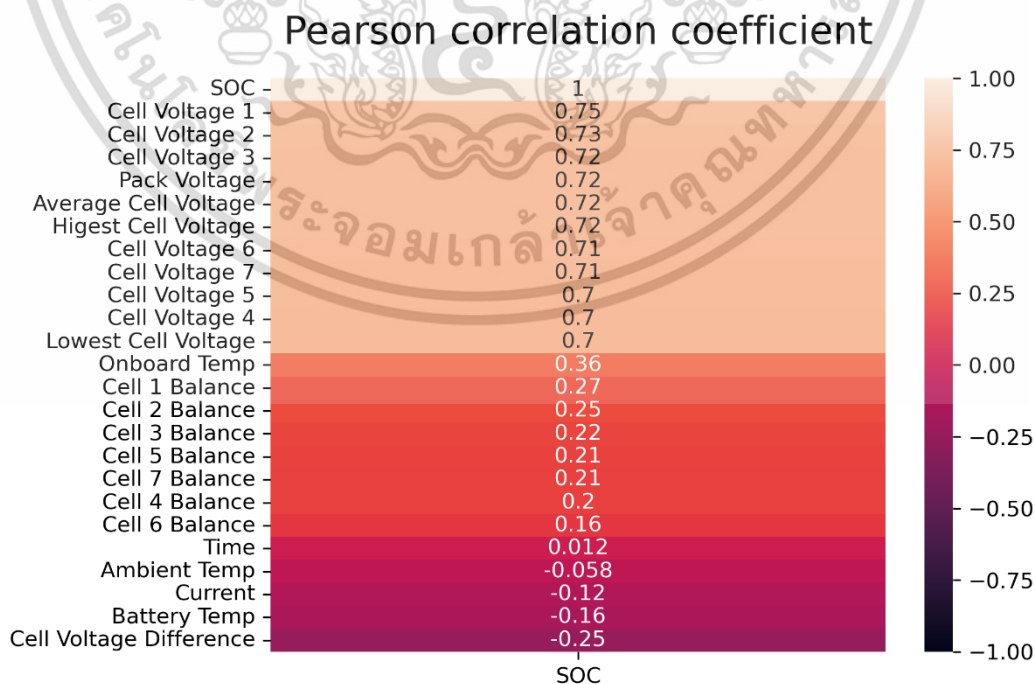
Time	Current	Voltage	Temp1 (Onboard Temp)	Temp2 (Battery temp)	Temp3 (Ambient temp)	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Cell 7
2023-02-20 19:56:01	0.887770	0.368979	0.116022	0.132743	0.455882	0.339689	0.346583	0.381464	0.362439	0.258929	0.399333	0.403061
2023-02-20 19:56:05	0.962590	0.414258	0.116022	0.137168	0.441176	0.375000	0.382148	0.417730	0.396396	0.283234	0.437333	0.435587
2023-02-20 19:56:09	0.979856	0.420039	0.127072	0.128319	0.470588	0.380650	0.387727	0.424446	0.403326	0.286706	0.444000	0.440051
2023-02-20 19:56:14	0.979856	0.425819	0.116022	0.146018	0.441176	0.387006	0.394700	0.430490	0.410256	0.291667	0.448667	0.446429
2023-02-20 19:56:18	0.979856	0.431599	0.121547	0.137168	0.470588	0.394068	0.400976	0.435863	0.415107	0.295139	0.454000	0.452806

รูปที่ 3.5 ตัวอย่างข้อมูลในงานวิจัยหลังทำให้อยู่ในบรรทัดฐานเดียวกัน

หลังจากข้อมูลถูกทำความสะอาดและอยู่ในบรรทัดฐานเดียวกันแล้ว จึงนำมาสู่การเลือกคุณลักษณะของข้อมูล (Feature Selection) โดยใช้วิธี Pearson correlation coefficient เพื่อหาสัมประสิทธิ์ของความสัมพันธ์ระหว่าง Feature และ Output สามารถหาได้โดยสมการที่ (3.3) มีค่าอยู่ระหว่าง -1 ถึง 1 ค่าที่ยิ่งเข้าใกล้ 1 หมายความว่า Feature และ Output มีความสัมพันธ์ไปในทิศทางเดียวกัน ในทางตรงกันข้าม ค่าที่ยิ่งเข้าใกล้ -1 หมายความว่า Feature และ Output มีความสัมพันธ์ไปในทิศทางตรงกันข้ามโดยค่า Pearson correlation coefficient ของงานวิจัยนี้แสดงดังรูปที่ 3.6

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}} \tag{3.3}$$

- โดยที่
- r คือ ค่าสัมประสิทธิ์ของความสัมพันธ์
 - x_i คือ ค่าของตัวแปร Feature
 - \bar{x} คือ ค่าเฉลี่ยของตัวแปร Feature
 - y_i คือ ค่าของตัวแปร Output
 - \bar{y} คือ ค่าเฉลี่ยของตัวแปร Output

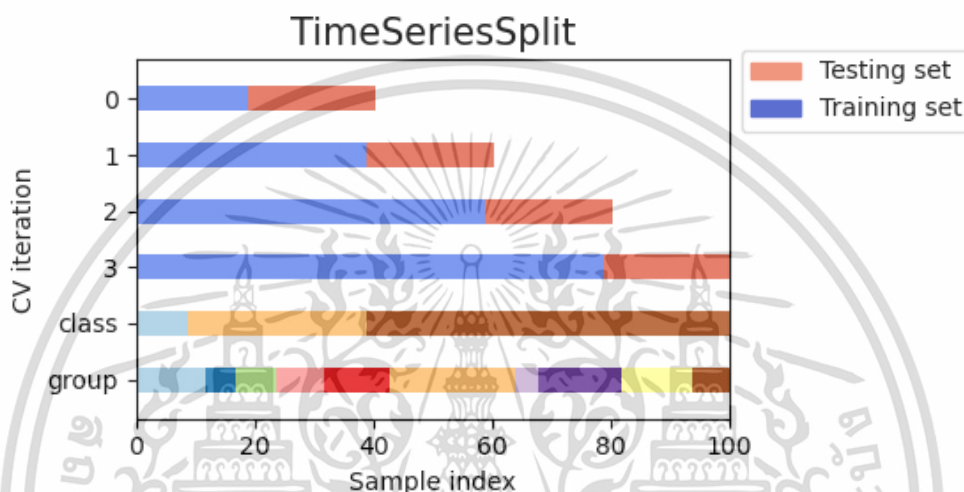


รูปที่ 3.6 ค่าสัมประสิทธิ์ Pearson correlation ในงานวิจัย

เอกสารนี้เป็นเอกสารที่สงวนไว้เพื่อใช้ในงานวิจัยเท่านั้น เมื่อผู้เห็นได้เห็นไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในงานวิจัยนี้ได้เลือกใช้ทุกคุณลักษณะยกเว้นตัวแปรเวลา Time ถึงแม้ว่าคุณลักษณะอื่นจะมีค่าสัมประสิทธิ์ที่ต่ำแต่ก็สามารถนำมาใช้เสริมกับคุณลักษณะหลักได้

หลังจากที่ได้ทำการเตรียมข้อมูลด้วยวิธีที่ได้กล่าวข้างต้นจะได้ชุดข้อมูลทั้งหมด 55,307 ข้อมูล ที่พร้อมใช้ในการสอนและทดสอบโมเดลปัญญาประดิษฐ์ โดยทำการแบ่งข้อมูลเป็นข้อมูลชุดสอน (Train Set) และชุดทดสอบ (Test Set) ในอัตราส่วน 7:3 และในส่วนของชุดสอนจะแบ่งเป็นข้อมูลชุดสอนและชุดข้อมูลตรวจสอบ (Validation Set) เนื่องจากชุดข้อมูลที่ใช้เป็นชุดข้อมูลที่ขึ้นกับเวลาดังนั้นการสร้างชุดข้อมูลตรวจสอบจึงใช้การแบ่งแบบ TimeSeriesSplit ตามรูปที่ 3.7



รูปที่ 3.7 การแบ่งชุดข้อมูลตรวจสอบแบบ TimeSeriesSplit [22]

โดยในงานวิจัยนี้ทำการแบ่งชุดข้อมูลตรวจสอบเป็น 4 ชุด ในทุกชุดจะมีข้อมูลตรวจสอบคิดเป็น 20% ของข้อมูลชุดสอนทั้งหมด ส่วนชุดข้อมูลสอนจะค่อยๆเพิ่มทีละ 20% ของชุดข้อมูลสอนทั้งหมดไปเรื่อยๆ จนถึง 80% ของชุดข้อมูลสอนทั้งหมด จะได้ชุดข้อมูลทดสอบดังนี้ จากชุดข้อมูลสอนที่คิดเป็น 70% ของข้อมูลทั้งหมดจะได้ 38,714 ข้อมูล นำมาแบ่งเป็นชุดข้อมูลตรวจสอบ 4 ชุด ได้ดังนี้

ชุดที่ 1 ชุดสอนคือข้อมูลตั้งแต่แถวที่ 0 ถึง 7,745 ชุดทดสอบตั้งแต่แถวที่ 7,746 ถึง 15,487

ชุดที่ 2 ชุดสอนคือข้อมูลตั้งแต่แถวที่ 0 ถึง 15,487 ชุดทดสอบตั้งแต่แถวที่ 15,488 ถึง 23,229

ชุดที่ 3 ชุดสอนคือข้อมูลตั้งแต่แถวที่ 0 ถึง 23,229 ชุดทดสอบตั้งแต่แถวที่ 23,230 ถึง 30,971

ชุดที่ 4 ชุดสอนคือข้อมูลตั้งแต่แถวที่ 0 ถึง 30,971 ชุดทดสอบตั้งแต่แถวที่ 30,972 ถึง 38,713 แสดงดังรูปที่ 3.8

```

Validation Data Set 1
Train index = [ 0 1 2 ... 7743 7744 7745]
Validation index = [ 7746 7747 7748 ... 15485 15486 15487]
Validation Data Set 2
Train index = [ 0 1 2 ... 15485 15486 15487]
Validation index = [15488 15489 15490 ... 23227 23228 23229]
Validation Data Set 3
Train index = [ 0 1 2 ... 23227 23228 23229]
Validation index = [23230 23231 23232 ... 30969 30970 30971]
Validation Data Set 4
Train index = [ 0 1 2 ... 30969 30970 30971]
Validation index = [30972 30973 30974 ... 38711 38712 38713]
    
```

รูปที่ 3.8 การแบ่งชุดข้อมูลตรวจสอบในงานวิจัย

โมเดลปัญญาประดิษฐ์จะทำการเรียนและปรับแต่งค่าพารามิเตอร์ของโมเดลปัญญาประดิษฐ์บนชุดข้อมูล Train Set และ Validate Set เมื่อได้โมเดลปัญญาประดิษฐ์ที่สมบูรณ์แล้วจะทำการทดสอบโมเดลปัญญาประดิษฐ์กับชุดข้อมูลที่โมเดลปัญญาประดิษฐ์ไม่เคยเห็นมาก่อนหรือ Test Set เพื่อประเมินประสิทธิภาพโมเดลปัญญาประดิษฐ์และทำการเปรียบเทียบประสิทธิภาพระหว่างโมเดลปัญญาประดิษฐ์ต่างชนิดกัน

ค่าระดับประจุแบตเตอรี่ที่ใช้ในการสอนหรือทดสอบโมเดลปัญญาประดิษฐ์มาจากวิธีการคำนวณด้วยวิธีการนับกระแส (Coulomb Counting Method) เป็นการคำนวณซึ่งใช้หลักการการรวมกันของกระแสระหว่างอัดประจุหรือคายประจุ คำนวณร่วมกับค่าระดับประจุแบตเตอรี่ตอนเริ่มต้นตามสมการที่ (3.4) [12] ส่วนค่าระดับประจุแบตเตอรี่ตอนเริ่มต้นสามารถหาได้จากกราฟข้อมูลแบตเตอรี่จากผู้ผลิต

$$SOC(t) = SOC(0) - \frac{1}{C_n} \int_0^t I(t)dt \tag{3.4}$$

- โดยที่ $SOC(t)$ คือ ระดับประจุแบตเตอรี่ที่เวลา t
- $SOC(0)$ คือ ระดับประจุแบตเตอรี่เริ่มต้น
- C_n คือ ความจุไฟฟ้าที่ปกติ
- $I(t)$ คือ กระแสไฟฟ้าที่เวลา t

ค่าความต้านทานภายในแบตเตอรี่สามารถหาได้จากสมการที่ (3.5) [7] ซึ่งจะเป็นการคำนวณโดยโปรแกรมคอมพิวเตอร์ระหว่างการทำนายระดับประจุแบตเตอรี่

$$V_b = V_{ocv} \pm I_b \times R_b \tag{3.5}$$

- โดยที่ V_b คือ แรงดันที่ขั้วแบตเตอรี่
- V_{ocv} คือ แรงดันเปิดวงจร
- I_b คือ กระแสอัดหรือคายประจุ
- R_b คือ ความต้านทานภายในแบตเตอรี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดสอบหาค่าแรงดันเปิดวงจรสำหรับแบตเตอรี่ Panasonic NCR18650B เป็นไปตามตามตารางที่ 3.1 [23]

ตารางที่ 3.1 ระดับประจุแบตเตอรี่และแรงดันเปิดวงจร

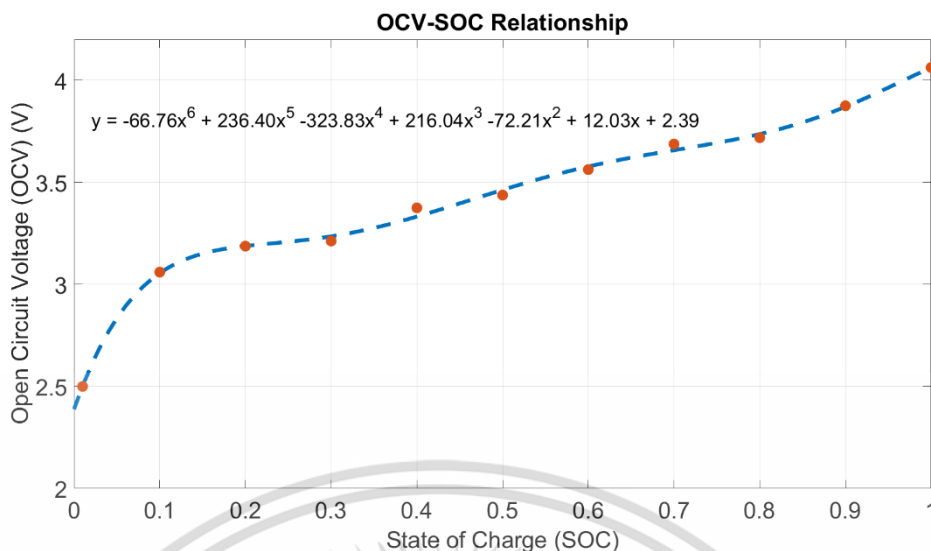
State of Charge	V_{oc}
0.01	2.50
0.1	3.06
0.2	3.19
0.3	3.21
0.4	3.38
0.5	3.44
0.6	3.56
0.7	3.69
0.8	3.72
0.9	3.88
1.0	4.06

จากนั้นนำค่าที่ได้จากตารางที่ 3.1 มาหาสมการด้วยวิธีการ Curve Fitting แบบ Polynomial โดยให้เลขกำลังสูงสุดที่ 6 ได้ดังสมการที่ (3.6)

$$V_{ocv} = 2.39 + 12.03(SOC) - 72.21(SOC)^2 + 216.04(SOC)^3 - 323.83(SOC)^4 + 236.40(SOC)^5 - 66.76(SOC)^6 \quad (3.6)$$

โดยที่ V_{ocv} คือ แรงดันที่ขั้วแบตเตอรี่
 SOC คือ ระดับประจุแบตเตอรี่

เมื่อนำสมการที่ (3.6) มาวาดกราฟเทียบกับค่าที่วัดได้จริงตามตารางที่ 3.1 จะได้ดังรูปที่ 3.9



รูปที่ 3.9 กราฟความสัมพันธ์ระหว่างแรงดันเปิดวงจรกับระดับประจุแบตเตอรี่

3.2. การออกแบบโมเดลปัญญาประดิษฐ์

สำหรับงานวิจัยนี้จะทำการทำนายระดับประจุแบตเตอรี่ซึ่งเป็นข้อมูลที่เกี่ยวข้องกับเวลาดังนั้นการเลือกใช้โมเดลปัญญาประดิษฐ์ที่เหมาะสมกับชุดข้อมูลเป็นสิ่งสำคัญ จึงได้เลือกโมเดลปัญญาประดิษฐ์ชนิด Recurrent Neural Network (RNN), Long Short-term Memory (LSTM) และ Gated Recurrent Unit (GRU) ซึ่งสามารถจัดการกับข้อมูลที่ขึ้นเก็บเวลาได้ และโมเดลปัญญาประดิษฐ์ชนิด Neural Network (NN) เพื่อดูข้อแตกต่าง โมเดลปัญญาประดิษฐ์ทั้งหมดถูกเขียนโดยโปรแกรมคอมพิวเตอร์ภาษา Python และเขียนบน Jupyter Lab

3.2.1. การหาค่าพารามิเตอร์ของโมเดลปัญญาประดิษฐ์

โมเดลปัญญาประดิษฐ์ที่มีการทำนายระดับประจุแบตเตอรี่ได้อย่างแม่นยำต้องมีการปรับค่าพารามิเตอร์ให้เหมาะสมโดยโมเดลปัญญาประดิษฐ์มีค่าพารามิเตอร์ดังนี้

- 1) ช่วงการแบ่งข้อมูล หรือ Timestep
- 2) จำนวนโหนด หรือ Neuron
- 3) จำนวนรอบการสอนโมเดลปัญญาประดิษฐ์ หรือ Epoch
- 4) จำนวนชุดข้อมูล หรือ Batch Size
- 5) ค่าที่โมเดลปัญญาประดิษฐ์จะทำการลด หรือ Loss function
- 6) อัลกอริทึมที่ใช้ปรับแต่งน้ำหนักของโมเดลปัญญาประดิษฐ์ หรือ Optimizer
- 7) ฟังก์ชันที่ใช้คำนวณผลลัพธ์ในแต่ละโหนด หรือ Activation Function
- 8) อัตราการตัดข้อมูล หรือ Dropout
- 9) อัตราการเรียนรู้ หรือ Learning Rate

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการหาค่าพารามิเตอร์ที่เหมาะสมของโมเดลปัญญาประดิษฐ์ทำได้โดยทำการสอนโมเดลปัญญาประดิษฐ์ด้วยชุดสอน (Train Set) และทดสอบบนชุดข้อมูลตรวจสอบ (Validation Set) ให้ครบตามที่ได้แบ่งไว้ แล้วตรวจสอบโมเดลปัญญาประดิษฐ์โดยทำการสร้างกราฟของค่าผิดพลาดด้วยชุดสอน (Train Loss) กับค่าผิดพลาดด้วยชุดตรวจสอบ (Validation Loss) โดยแกนนอนคือจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์ (Epoch) กับแกนตั้งเป็นค่าความผิดพลาด (Error) พิจารณาร่วมกับค่าความผิดพลาด เพื่อตรวจสอบว่าโมเดลปัญญาประดิษฐ์เกิดการจดจำข้อมูลการสอนมากเกินไป (Overfitting) หรือ น้อยเกินไป (Underfitting) เมื่อรวมทั้งหมดจะทราบแนวโน้มในการปรับค่าพารามิเตอร์โมเดลปัญญาประดิษฐ์เพื่อให้ได้โมเดลปัญญาประดิษฐ์ที่มีความแม่นยำ แต่ในส่วนของ การหาจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์ที่เหมาะสมที่สุดจะได้มาจากการหาค่าเฉลี่ยของความผิดพลาดด้วยชุดตรวจสอบทุกชุดในแต่ละรอบการสอนโมเดลปัญญาประดิษฐ์ โดยจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์ที่ให้ค่าเฉลี่ยของความผิดพลาดต่ำที่สุดคือจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์ที่เหมาะสมที่สุด

3.2.2. การทำงานของโมเดลปัญญาประดิษฐ์

ในงานวิจัยนี้การสอนและทดสอบโมเดลปัญญาประดิษฐ์ที่พิจารณาค่าความต้านทานภายใน จะมีความซับซ้อนเนื่องมาจากขั้นตอนในการหาค่าความต้านทานภายในแบตเตอรี่ซึ่งอาศัยค่าแรงดันเปิดวงจรแบตเตอรี่ที่ไม่มีมากับข้อมูลจากระบบจัดการแบตเตอรี่ สามารถอธิบายได้เป็นขั้นตอนดังนี้

ขั้นตอนการสอนโมเดลปัญญาประดิษฐ์

- 1) โมเดลปัญญาประดิษฐ์ทำการเรียนรู้และทำนายค่าแรงดันเปิดวงจร
- 2) คำนวณค่าความต้านทานภายใน
- 3) โมเดลปัญญาประดิษฐ์ทำการเรียนรู้และทำนายค่าระดับประจุแบตเตอรี่

ขั้นตอนการทดสอบโมเดลปัญญาประดิษฐ์

- 1) โมเดลปัญญาประดิษฐ์ทำการทำนายค่าแรงดันเปิดวงจร
- 2) คำนวณค่าความต้านทานภายใน
- 3) โมเดลปัญญาประดิษฐ์ทำการทำนายค่าระดับประจุแบตเตอรี่

สำหรับโมเดลปัญญาประดิษฐ์ชนิดที่ไม่พิจารณาค่าความต้านทานภายในมีขั้นตอนการทำงานที่แตกต่างเล็กน้อยจากที่ได้กล่าวไปข้างต้นโดยตัดขั้นตอนที่ 1 และ 2 ออกจากขั้นตอนการสอนและทดสอบโมเดลปัญญาประดิษฐ์

3.2.3. การวิเคราะห์ผล

ในการวิเคราะห์ผลเพื่อทำการเปรียบเทียบและประเมินประสิทธิภาพของโมเดลปัญญาประดิษฐ์สามารถแบ่งได้เป็น 2 ส่วน 1) วิเคราะห์ผลจากค่าความผิดพลาด 2) วิเคราะห์ระยะเวลาที่ใช้ในการสอนและทดสอบโมเดลปัญญาประดิษฐ์

3.2.3.1. การวิเคราะห์ผลจากค่าความผิดพลาด

การวิเคราะห์จากค่าความผิดพลาด (Error) สามารถหาได้ด้วยวิธีค่าสัมบูรณ์เฉลี่ยของค่าความผิดพลาด (Mean absolute error) หรือ MAE ตามสมการที่ (3.7) และค่ารากที่สองของค่าเฉลี่ยกำลังสองของค่าความผิดพลาด (Root mean squared error) หรือ RMSE ตามสมการที่ (3.8) เพื่อทำการวิเคราะห์ถึงความแม่นยำในการทำนายระดับประจุแบตเตอรี่และเปรียบเทียบความแม่นยำระหว่างโมเดลปัญญาประดิษฐ์ที่มีการคำนึงถึงค่าความต้านภายในและไม่มีค่าความต้านทานภายใน รวมไปถึงการเปรียบเทียบระหว่างโมเดลปัญญาประดิษฐ์ชนิดแตกต่างกัน

$$MAE = \frac{\sum_{i=1}^n |Y_i - \hat{Y}_i|}{n} \quad (3.7)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{n}} \quad (3.8)$$

โดยที่	Y_i	คือ	ค่าจริงของตัวแปร
	\hat{Y}_i	คือ	ค่าทำนายของตัวแปร
	n	คือ	จำนวนตัวแปร

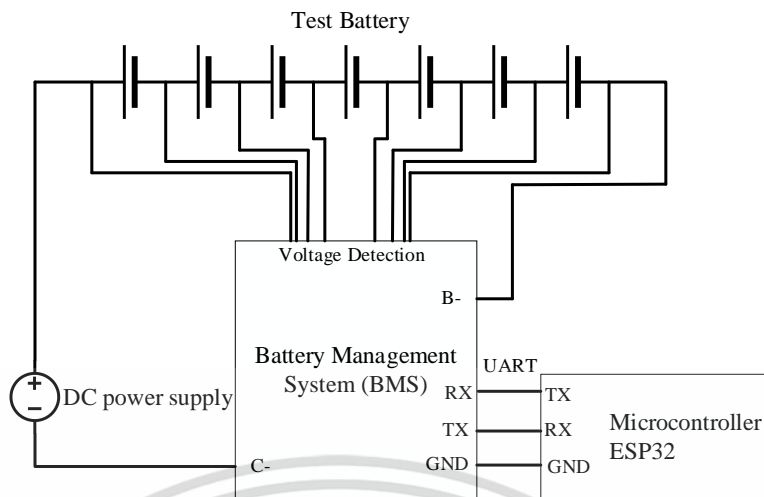
3.2.3.2. การวิเคราะห์ระยะเวลาที่ใช้ในการสอนและทดสอบโมเดลปัญญาประดิษฐ์

ในการวิเคราะห์ระยะเวลาที่ใช้ในการสอนและทดสอบโมเดลปัญญาประดิษฐ์สามารถทำได้โดยใช้ฟังก์ชัน time ใน python จับเวลาก่อนและหลังการสอนหรือทดสอบ นำเวลาหลังมาลบเวลาก่อนจะได้เวลาที่ใช่ไป ซึ่งนำมาใช้เปรียบเทียบและประเมินประสิทธิภาพของโมเดลปัญญาประดิษฐ์

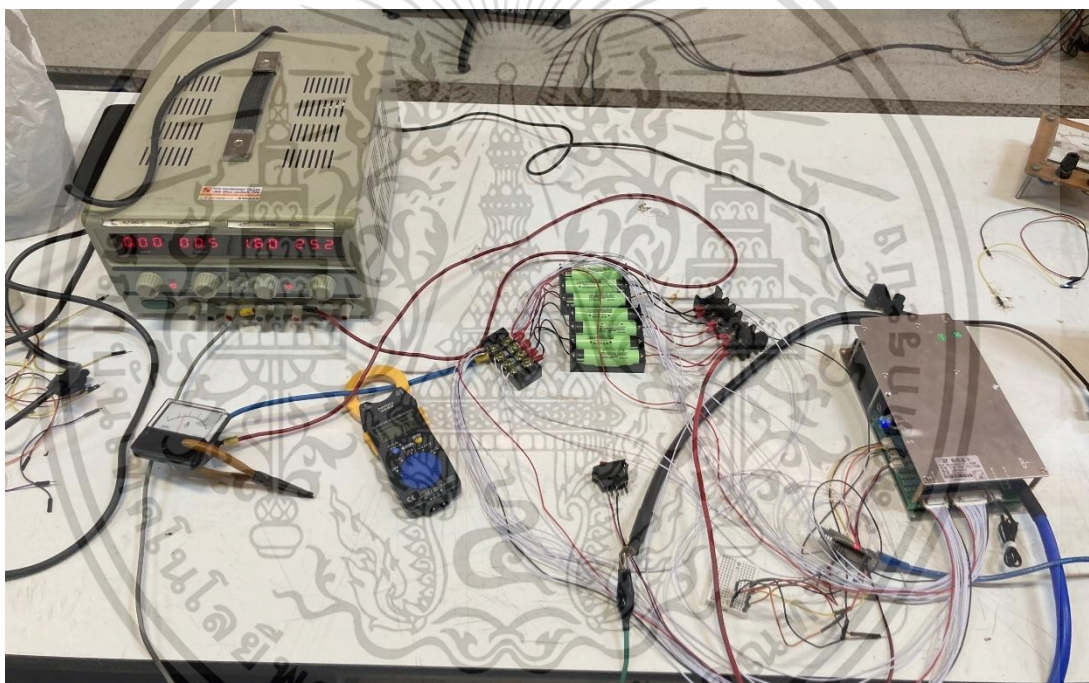
3.3. การออกแบบการทดสอบแบตเตอรี่

งานวิจัยนี้ได้ทำการทดสอบกับแบตเตอรี่ชนิดลิเทียมไอออน Panasonic NCR18650B ซึ่งเป็นแบตเตอรี่ลิเทียมนิเกิลโคบอลต์อะลูมิเนียมออกไซด์ Lithium Nickel Cobalt Aluminium Oxide หรือ NCA มีจุดเด่นด้านความหนาแน่นกำลังไฟฟ้า (Power Density) และความหนาแน่นพลังงานไฟฟ้า (Energy Density) สูงแต่มีราคาที่สูงเหมาะกับการใช้งานในยานยนต์ไฟฟ้าและระบบกักเก็บพลังงานไฟฟ้า มีค่าความจุ (Rated Capacity) 3,350 mAh แรงดันไฟฟ้าที่ระบุ (Nominal Voltage) 3.6 V จำนวน 7 ก้อนต่ออนุกรม ทำการอัดประจุผ่านแหล่งจ่ายไฟกระแสตรง (DC Power Supply) แบบกระแสคงที่แรงดันคงที่ CC/CV และทำการคายประจุผ่านตัวต้านทานปรับค่าได้ (Variable Resistor) แบบกระแสคงที่และกำลังคงที่ ทำการทดสอบที่ระดับแรงดันและกระแสเป็นไปตามขอบเขตของผู้ผลิตแบตเตอรี่ โดยมีวงจรการทดสอบขณะอัดประจุตามรูปที่ 3.10 และรูปที่ 3.11 และวงจรการทดสอบขณะคายประจุตามรูปที่ 3.12 และรูปที่ 3.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

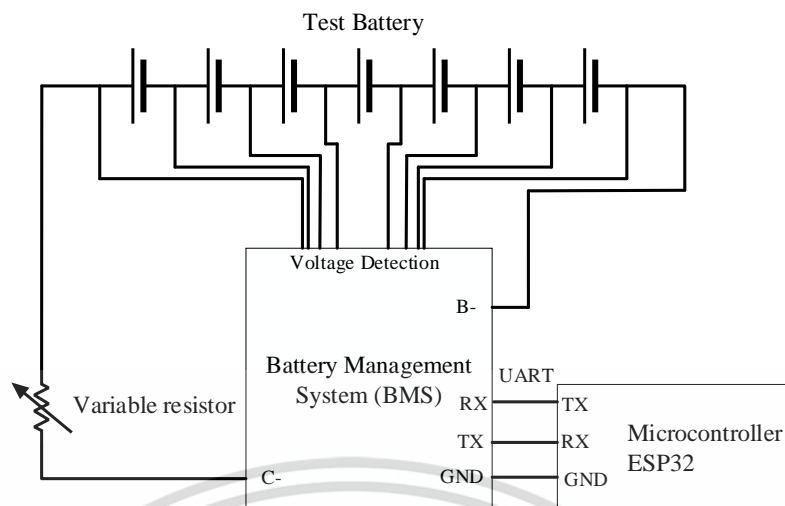


รูปที่ 3.10 วงจรไฟฟ้าขณะทำการอัดประจุ

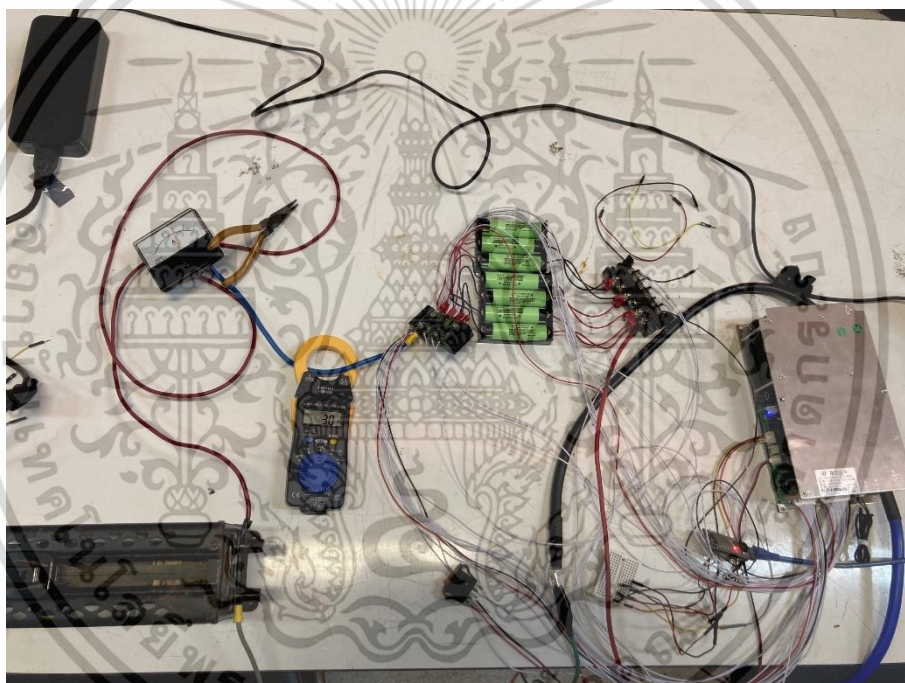


รูปที่ 3.11 วงจรไฟฟ้าขณะทำการอัดประจุที่ทดลองจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 วงจรไฟฟ้าขณะทำการคายประจุ

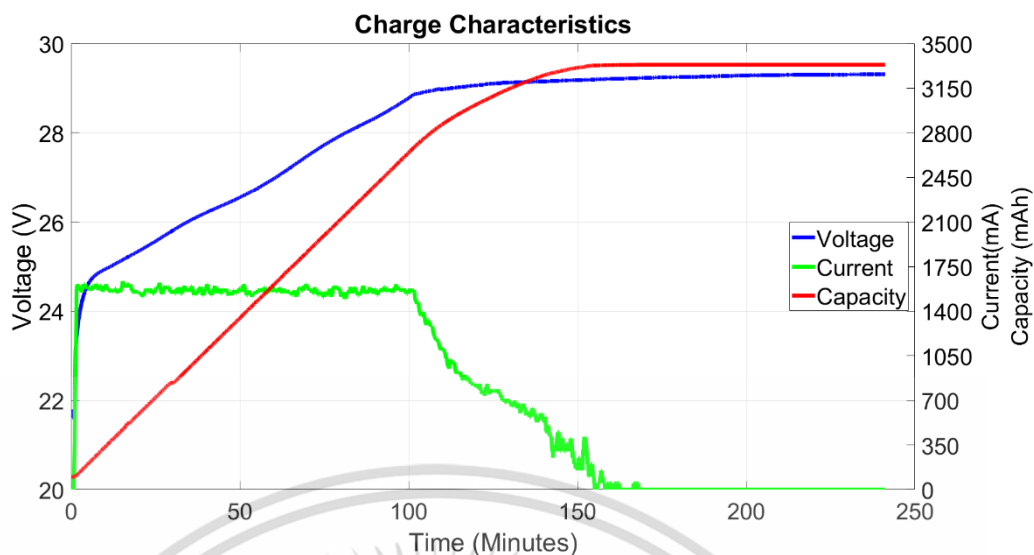


รูปที่ 3.13 วงจรไฟฟ้าขณะทำการคายประจุจริง

3.3.1. การทดสอบการอัดประจุ

ในการทดสอบการอัดประจุแบตเตอรี่ทำการอัดประจุโดยแหล่งจ่ายไฟกระแสตรงแบบกระแสคงที่แรงดันคงที่ หรือ CC/CV ที่กระแส 0.5C (1.65A) แรงดัน 29.4V มีความสัมพันธ์ระหว่างแรงดันกระแส และประจุแบตเตอรี่กับเวลาตามรูปที่ 3.14

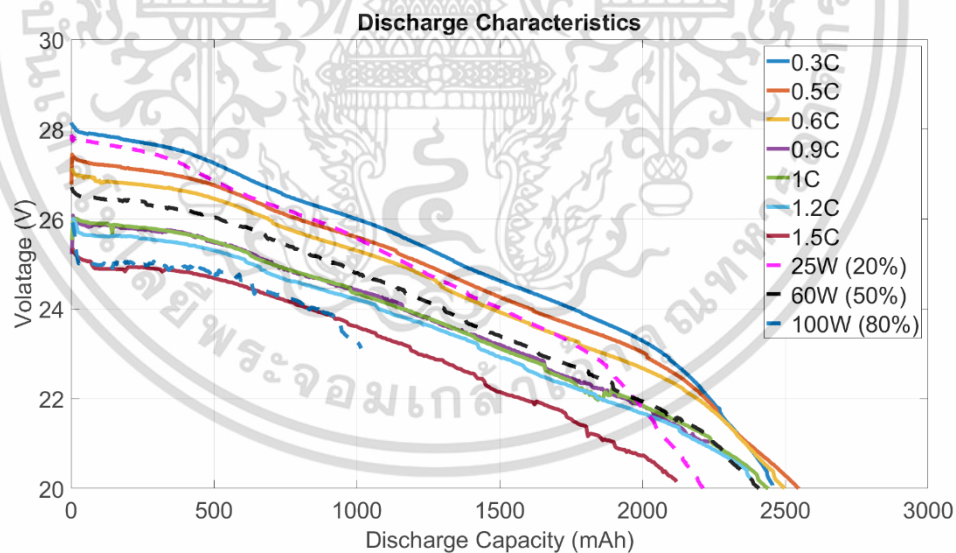
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 กราฟคุณลักษณะการอัดประจุของข้อมูลที่ใช้ในงานวิจัย

3.3.2. การทดสอบการคายประจุ

ในการทดสอบการคายประจุแบตเตอรี่ทำการคายประจุโดยตัวต้านทานปรับค่าได้ แบบกระแสคงที่ขนาด 1A (0.3C), 1.6A (0.5C), 2A (0.6C), 3A (0.9C), 3.2A (1C), 4A (1.2C) และ 5A (1.5C) แบบกำลังคงที่ขนาด 25W (20%) 60W (50%) และ 85W (80%) มีความสัมพันธ์ระหว่างแรงดัน และจำนวนคายประจุแบตเตอรี่ตามรูปที่ 3.15



รูปที่ 3.15 กราฟคุณลักษณะการคายประจุของข้อมูลที่ใช้ในงานวิจัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการดำเนินงานวิจัย

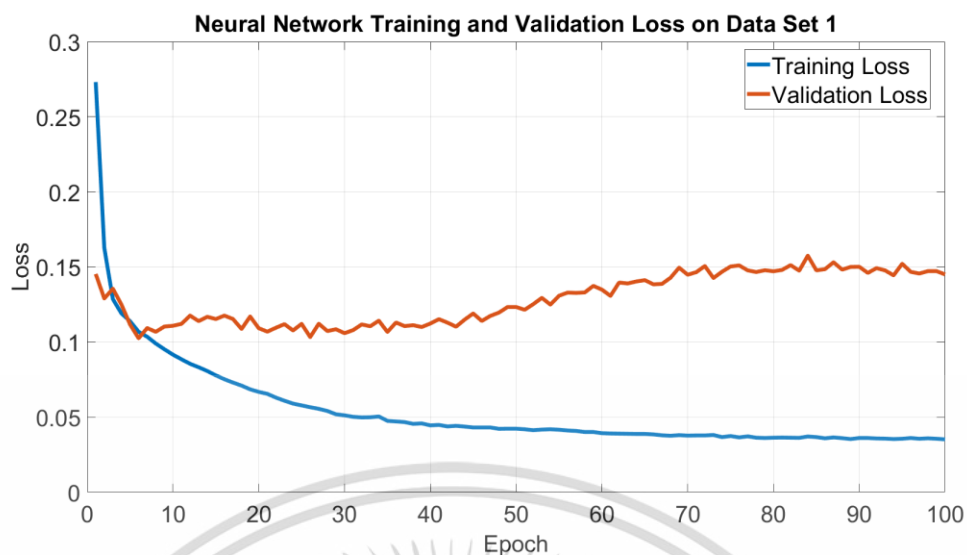
ในบทนี้จะทำการแสดงผลจากการดำเนินงานวิจัยที่ได้กล่าวในบทก่อนหน้า โดยจะแบ่งเป็นผลการปรับแต่งพารามิเตอร์ของโมเดลปัญญาประดิษฐ์ ผลการทำนายระดับประจวบเตอรีระยะเวลาในการทำสอนและทำนายของโมเดลปัญญาประดิษฐ์ที่พิจารณาความต้านทานภายในและไม่พิจารณาความต้านทานภายใน และท้ายสุดเปรียบเทียบผลของโมเดลปัญญาประดิษฐ์ที่ต่างชนิดกันในด้านความผิดพลาดและระยะเวลา

4.1. ผลการปรับแต่งพารามิเตอร์ของโมเดลปัญญาประดิษฐ์

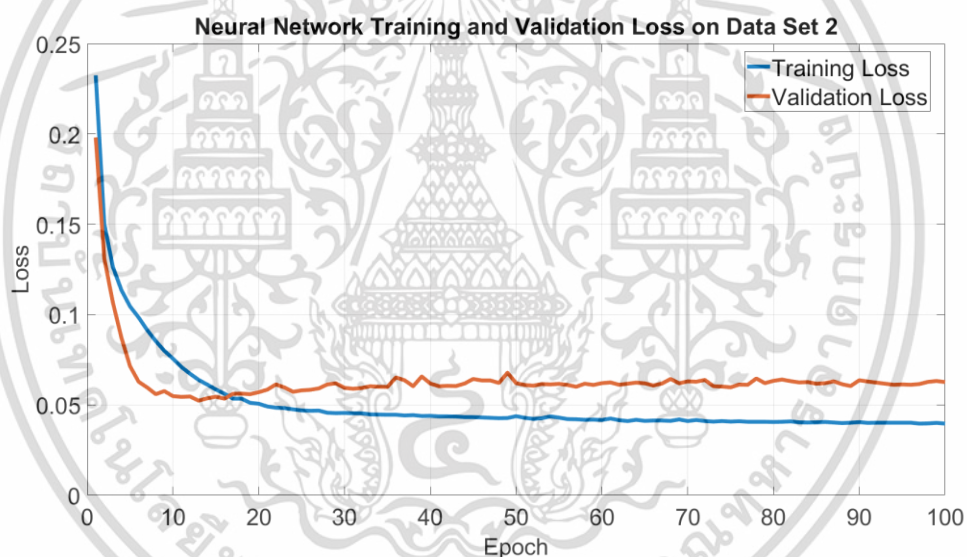
โมเดลปัญญาประดิษฐ์ที่ได้นำมาใช้ในงานวิจัยได้มีการปรับแต่งค่าพารามิเตอร์โดยใช้ข้อมูลสอน (Train Set) และข้อมูลตรวจสอบ (Validation Set) 4 ชุดที่ได้ทำการแบ่งในการปรับแต่งค่าพารามิเตอร์ ในงานวิจัยนี้ได้ใช้โมเดลปัญญาประดิษฐ์ชนิด Neural Network (RNN) , Recurrent Neural Network (RNN), Long Short-term Memory (LSTM) และ Gated Recurrent Unit (GRU) โดยในการปรับแต่งค่าพารามิเตอร์จะอาศัยการวาดกราฟระหว่างความผิดพลาดชุดสอน (Training Loss) ความผิดพลาดชุดตรวจสอบ (Validation Loss) กับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์ (Epoch) พิจารณาร่วมกับความผิดพลาดบนแต่ละชุดตรวจสอบเพื่อทำการปรับแต่งค่าพารามิเตอร์ให้เหมาะสมกับชุดข้อมูล และทำการปรับแต่งค่าพารามิเตอร์แยกแต่ละชนิดโมเดลปัญญาประดิษฐ์โดยให้ผลการปรับแต่งค่าพารามิเตอร์ดังนี้

4.1.1. ค่าพารามิเตอร์โมเดล Neural Network

โมเดลปัญญาประดิษฐ์ชนิด Neural Network (NN) ที่ได้ทำการปรับแต่งค่าพารามิเตอร์มีกราฟความสัมพันธ์ระหว่างความผิดพลาดชุดสอน (Train Loss) ความผิดพลาดชุดตรวจสอบ (Validation Loss) กับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์ (Epoch) บนชุดตรวจสอบ 4 ชุด ดังรูปที่ 4.1 ถึง รูปที่ 4.4 ซึ่งมีลักษณะของกราฟที่ค่าความผิดพลาดชุดตรวจสอบสูงกว่าความผิดพลาดชุดสอนเล็กน้อยตลอดช่วงรอบการสอนโมเดลปัญญาประดิษฐ์เป็นลักษณะที่ บ่งบอกถึงการเรียนรู้ของโมเดลปัญญาประดิษฐ์ที่ไม่มากเกินไป (Overfitting) หรือน้อยเกินไป (Underfitting) ตลอดช่วงจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนทุกชุดข้อมูลยกเว้นชุดที่ 1 ที่เริ่มเกิดการเรียนรู้ของโมเดลปัญญาประดิษฐ์ที่ Overfitting ในช่วง ตั้งแต่ประมาณ Epoch 50 เป็นต้นไป สังเกตจากการที่ Validation Loss เริ่มลู่ออกจาก Training Loss

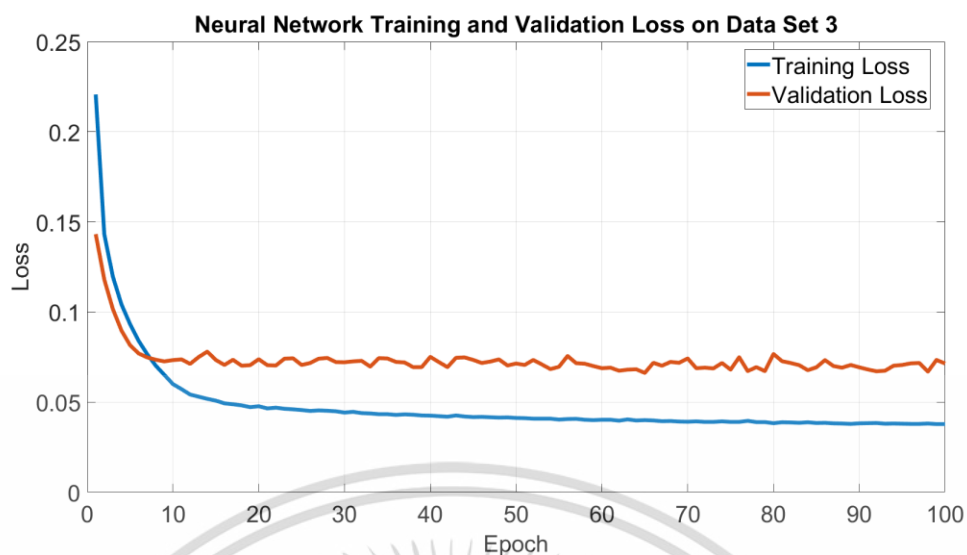


รูปที่ 4.1 กราฟความสัมพันธ์ระหว่างความผิดพลาดบนชุดสอนความผิดพลาดบนชุดตรวจสอบกับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 1 ของโมเดลชนิด Neural Network

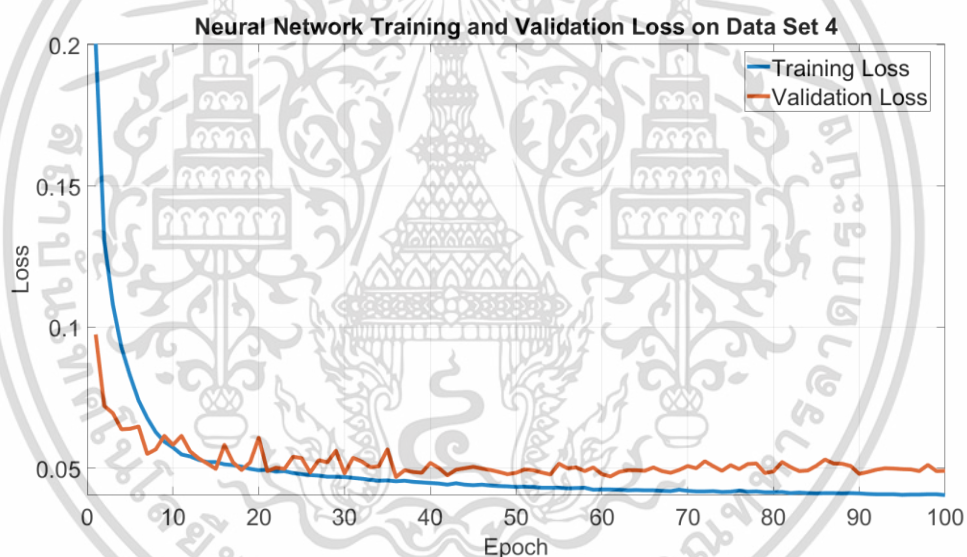


รูปที่ 4.2 กราฟความสัมพันธ์ระหว่างความผิดพลาดบนชุดสอนความผิดพลาดบนชุดตรวจสอบกับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 2 ของโมเดลชนิด Neural Network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 กราฟความสัมพันธ์ระหว่างความผิดพลาดบนชุดสอนความผิดพลาดบนชุดตรวจสอบกับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 3 ของโมเดลชนิด Neural Network



รูปที่ 4.4 กราฟความสัมพันธ์ระหว่างความผิดพลาดบนชุดสอนความผิดพลาดบนชุดตรวจสอบกับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 4 ของโมเดลชนิด Neural Network

ค่าสัมบูรณ์เฉลี่ยของค่าความผิดพลาด (Mean absolute error) หรือ MAE และค่ารากที่สองของค่าเฉลี่ยกำลังสองของค่าความผิดพลาด (Root mean squared error) หรือ RMSE บนชุดข้อมูลตรวจสอบทั้ง 4 ชุด แสดงตามตารางที่ 4.1

ตารางที่ 4.1 ค่าความผิดพลาดเพี้ยนบนชุดข้อมูลตรวจสอบของโมเดลชนิด Neural Network

	MAE Validation Loss (%)	RMSE Validation Loss (%)
Data Set 1	13.1929	15.8684
Data Set 2	5.6992	7.5179
Data Set 3	6.4974	10.0414
Data Set 4	4.4731	7.1410

เมื่อทำการหาค่าเฉลี่ยของความผิดพลาดเพี้ยนชุดตรวจสอบทั้ง 4 ชุดในแต่ละรอบการสอนโมเดล ปัญญาประดิษฐ์พบว่ารอบการสอนที่ 26 ให้ค่าความผิดพลาดเพี้ยนบนชุดตรวจสอบต่ำสุดที่ 0.0705 ซึ่งจะ เป็นรอบการสอนที่จะนำไปใช้บนโมเดลปัญญาประดิษฐ์ที่ทำการทดสอบกับข้อมูลทดสอบ (Test Set)

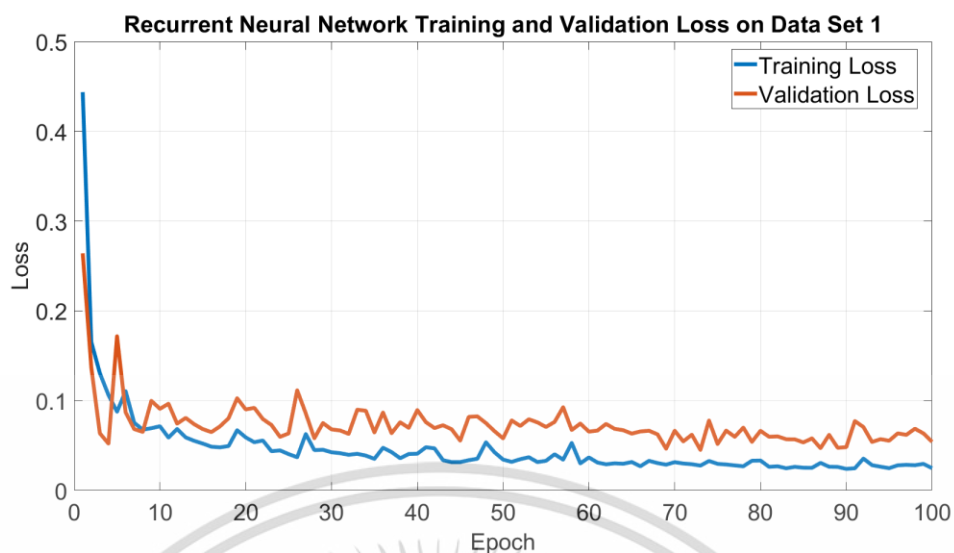
ค่าพารามิเตอร์ที่ได้ทำการปรับแต่งทั้งหมดบนโมเดลปัญญาประดิษฐ์ชนิด Neural Network แสดงตามตารางที่ 4.2

ตารางที่ 4.2 ค่าพารามิเตอร์โมเดลปัญญาประดิษฐ์ชนิด Neural Network

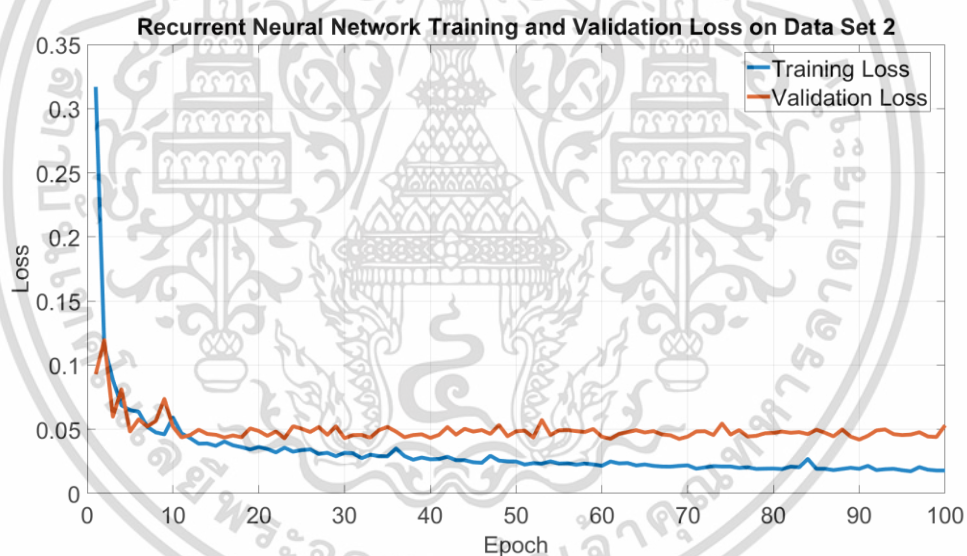
Hyperparameter	Value
Neuron	225
Epoch	26
Batch Size	512
Loss Function	Mae
Optimizer	Adam
Activation Function	Relu
Dropout	0.1
Learning Rate	0.0003

4.1.2. ค่าพารามิเตอร์โมเดล Recurrent Neural Network

โมเดลปัญญาประดิษฐ์ชนิด Recurrent Neural Network (RNN) ที่ได้ทำการปรับแต่ง ค่าพารามิเตอร์มีกราฟความสัมพันธ์ระหว่างความผิดพลาดเพี้ยนชุดสอน (Train Loss) ความผิดพลาดเพี้ยนชุด ตรวจสอบ (Validation Loss) กับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์ (Epoch) บนชุด ตรวจสอบ 4 ชุด ดังรูปที่ 4.5 ถึง รูปที่ 4.8 ซึ่งมีลักษณะของกราฟที่ค่าความผิดพลาดเพี้ยนชุดตรวจสอบสูง กว่าความผิดพลาดเพี้ยนชุดสอนเล็กน้อยตลอดช่วงรอบการสอนโมเดลปัญญาประดิษฐ์เป็นลักษณะที่ บ่ง บอกลถึงการเรียนรู้ของโมเดลปัญญาประดิษฐ์ที่ไม่มากเกินไป (Overfitting) หรือน้อยเกินไป (Underfitting) ตลอดช่วงจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์

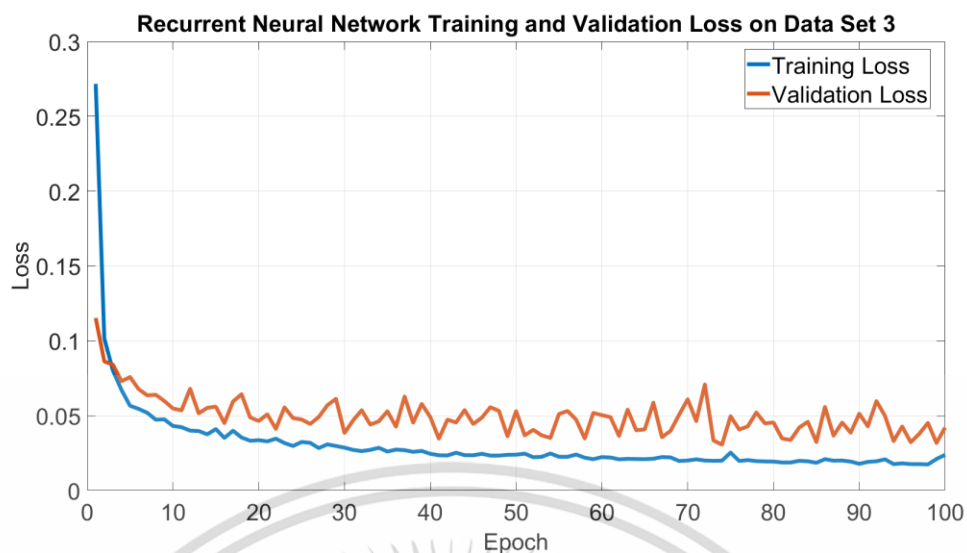


รูปที่ 4.5 กราฟความสัมพันธ์ระหว่างความผิดพลาดบนชุดสอนความผิดพลาดบนชุดตรวจสอบกับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 1 ของโมเดลชนิด Recurrent Neural Network

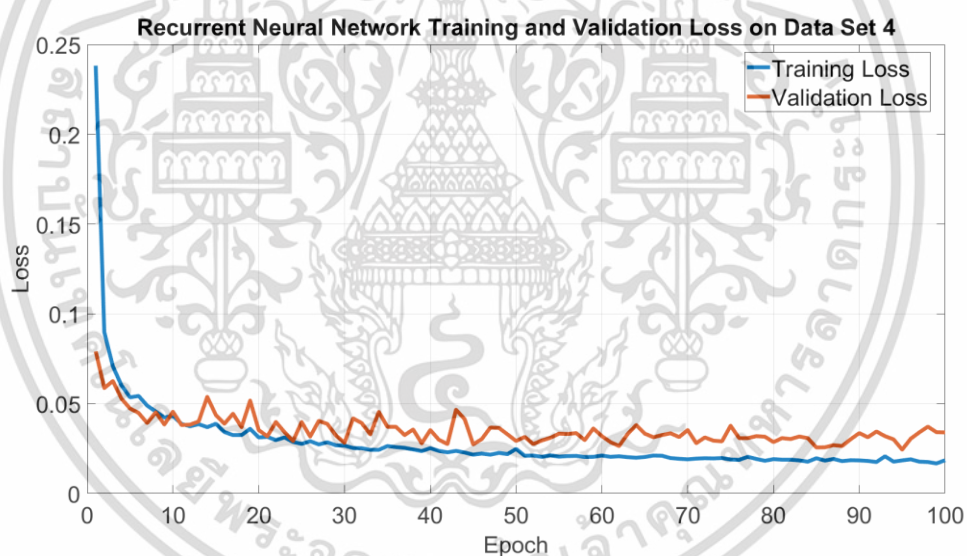


รูปที่ 4.6 กราฟความสัมพันธ์ระหว่างความผิดพลาดบนชุดสอนความผิดพลาดบนชุดตรวจสอบกับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 2 ของโมเดลชนิด Recurrent Neural Network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 กราฟความสัมพันธ์ระหว่างความผิดพลาดบนชุดสอนความผิดพลาดบนชุดตรวจสอบกับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 3 ของโมเดลชนิด Recurrent Neural Network



รูปที่ 4.8 กราฟความสัมพันธ์ระหว่างความผิดพลาดบนชุดสอนความผิดพลาดบนชุดตรวจสอบกับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 4 ของโมเดลชนิด Recurrent Neural Network

ค่าสัมบูรณ์เฉลี่ยของค่าความผิดพลาด (Mean absolute error) หรือ MAE และค่ารากที่สองของค่าเฉลี่ยกำลังสองของค่าความผิดพลาด (Root mean squared error) หรือ RMSE บนชุดข้อมูลตรวจสอบทั้ง 4 ชุด แสดงตามตารางที่ 4.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 ค่าความผิดพลาดเพี้ยนบนชุดข้อมูลตรวจสอบของโมเดลชนิด Recurrent Neural Network

	MAE Validation Loss (%)	RMSE Validation Loss (%)
Data Set 1	4.9337	6.1318
Data Set 2	4.8349	6.2978
Data Set 3	3.8248	6.0110
Data Set 4	3.0907	3.8724

เมื่อทำการหาค่าเฉลี่ยของความผิดพลาดเพี้ยนชุดตรวจสอบทั้ง 4 ชุดในแต่ละรอบการสอนโมเดล ปัญญาประดิษฐ์พบว่ารอบการสอนที่ 73 ให้ค่าความผิดพลาดเพี้ยนบนชุดตรวจสอบต่ำสุดที่ 0.0384 ซึ่งจะ เป็นรอบการสอนที่จะนำไปใช้บนโมเดลปัญญาประดิษฐ์ที่ทำการทดสอบกับข้อมูลทดสอบ (Test Set)

ค่าพารามิเตอร์ที่ได้ทำการปรับแต่งทั้งหมดบนโมเดลปัญญาประดิษฐ์ชนิด Recurrent Neural Network แสดงตามตารางที่ 4.4

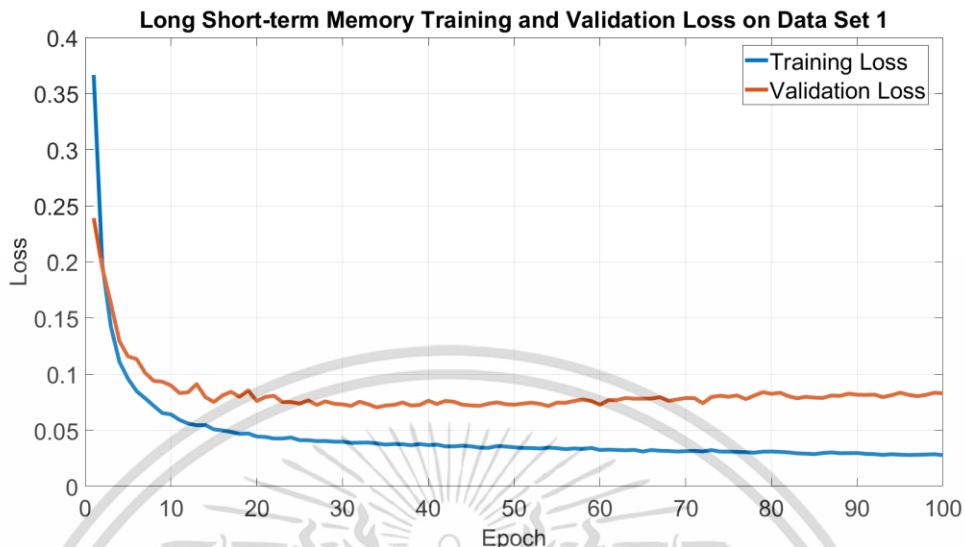
ตารางที่ 4.4 ค่าพารามิเตอร์โมเดลปัญญาประดิษฐ์ชนิด Recurrent Neural Network

Hyperparameter	Value
Timestep	30
Neuron	300
Epoch	73
Batch Size	512
Loss Function	Mae
Optimizer	Adam
Activation Function	Relu
Dropout	0.1
Learning Rate	0.0005

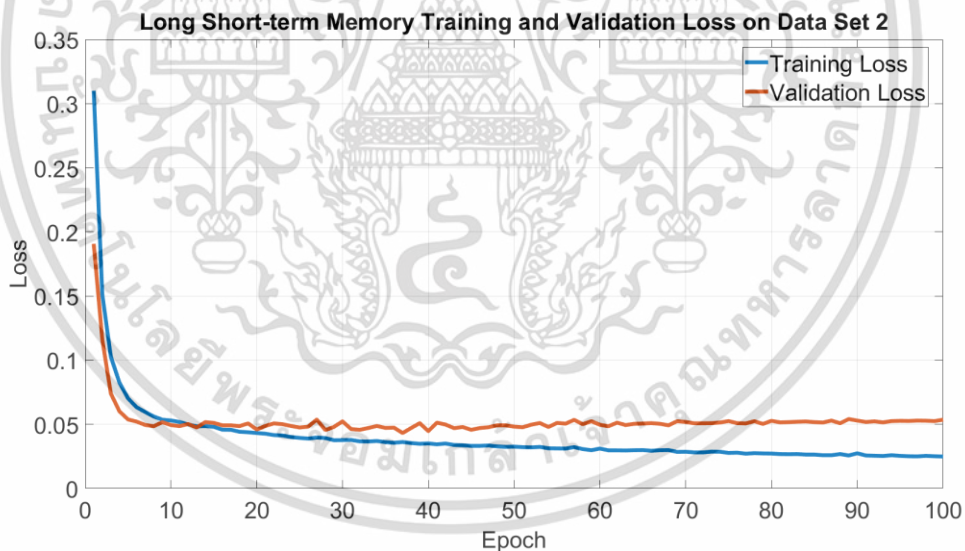
4.1.3. ค่าพารามิเตอร์โมเดล Long Short-term Memory

โมเดลปัญญาประดิษฐ์ชนิด Long Short-term Memory (LSTM) ที่ได้ทำการปรับแต่ง ค่าพารามิเตอร์มีกราฟความสัมพันธ์ระหว่างความผิดพลาดเพี้ยนชุดสอน (Train Loss) ความผิดพลาดเพี้ยนชุด ตรวจสอบ (Validation Loss) กับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์ (Epoch) บนชุด ตรวจสอบ 4 ชุด ดังรูปที่ 4.9 ถึง รูปที่ 4.12 ซึ่งสามารถสังเกตจากรูปที่ 4.9 และ รูปที่ 4.10 ได้ว่า โมเดลปัญญาประดิษฐ์เริ่มเกิด Overfitting ในช่วง Epoch ท้ายๆ สังเกตจากการที่ Validation Loss เริ่มลู่ออกจาก Training Loss ในส่วนของรูปที่ 4.11 โมเดลปัญญาประดิษฐ์มีการเรียนรู้ที่ดี

ตลอดช่วง Epoch และรูปที่ 4.12 โมเดลปัญญาประดิษฐ์เกิด Underfitting ในช่วง Epoch แรกๆ สังเกตจากการที่ Validation Loss มีค่าที่น้อยกว่า Training Loss

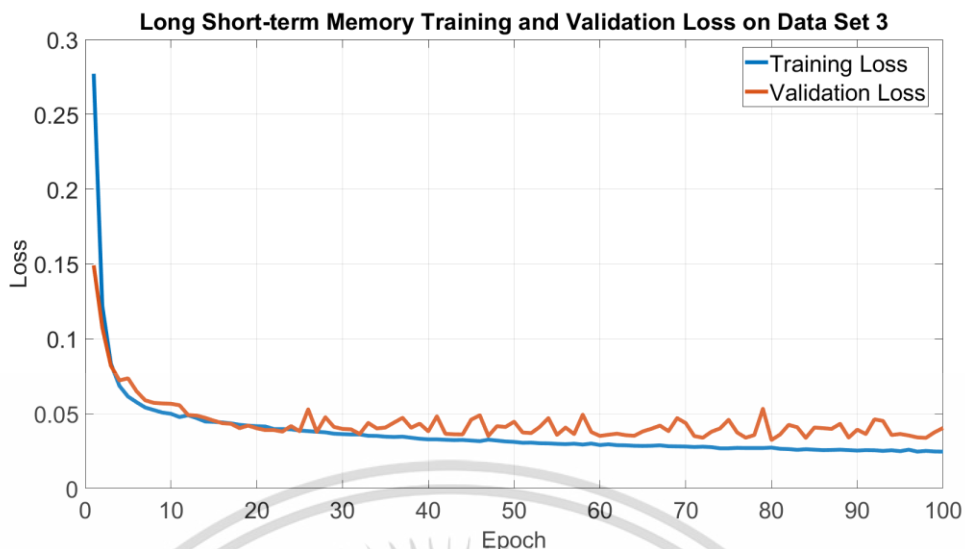


รูปที่ 4.9 กราฟความสัมพันธ์ระหว่างความผิดพลาดในการสอนความผิดพลาดในการตรวจสอบกับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 1 ของโมเดลชนิด Long Short-term Memory

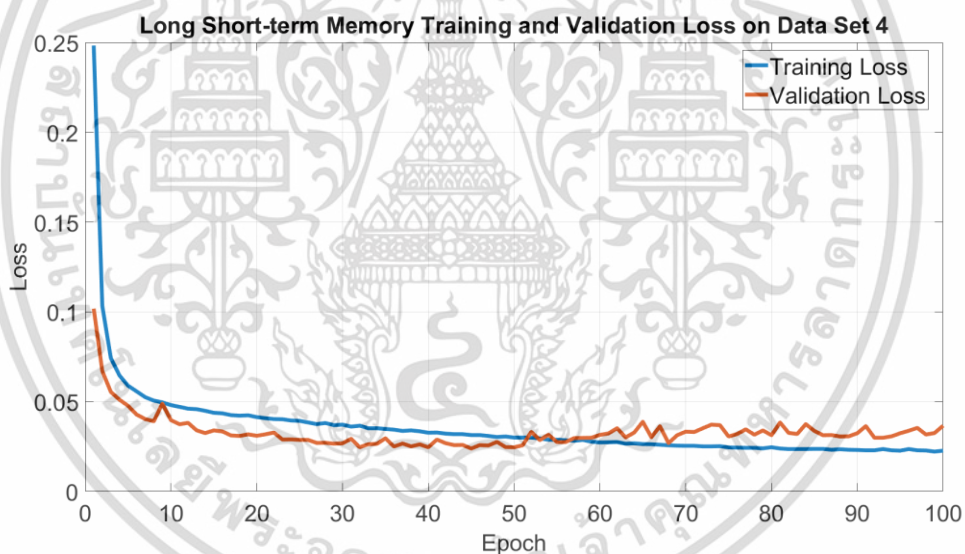


รูปที่ 4.10 กราฟความสัมพันธ์ระหว่างความผิดพลาดในการสอนความผิดพลาดในการตรวจสอบกับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 2 ของโมเดลชนิด Long Short-term Memory

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 กราฟความสัมพันธ์ระหว่างความผิดพลาดขั้นตอนความผิดพลาดขั้นตอนตรวจสอบกับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 3 ของโมเดลชนิด Long Short-term Memory



รูปที่ 4.12 กราฟความสัมพันธ์ระหว่างความผิดพลาดขั้นตอนความผิดพลาดขั้นตอนตรวจสอบกับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 4 ของโมเดลชนิด Long Short-term Memory

ค่าสัมบูรณ์เฉลี่ยของค่าความผิดพลาด (Mean absolute error) หรือ MAE และค่ารากที่สองของค่าเฉลี่ยกำลังสองของค่าความผิดพลาด (Root mean squared error) หรือ RMSE บนชุดข้อมูลตรวจสอบทั้ง 4 ชุด แสดงตามตารางที่ 4.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.5 ค่าความผิดพลาดเพี้ยนบนชุดข้อมูลตรวจสอบของโมเดลชนิด Long Short-term Memory

	MAE Validation Loss (%)	RMSE Validation Loss (%)
Data Set 1	7.5467	9.6316
Data Set 2	4.8779	5.9981
Data Set 3	3.6811	5.6298
Data Set 4	3.3316	4.0420

เมื่อทำการหาค่าเฉลี่ยของความผิดพลาดเพี้ยนชุดตรวจสอบทั้ง 4 ชุดในแต่ละรอบการสอนโมเดล ปัญญาประดิษฐ์พบว่ารอบการสอนที่ 47 ให้ค่าความผิดพลาดเพี้ยนบนชุดตรวจสอบต่ำสุดที่ 0.0455 ซึ่งจะ เป็นรอบการสอนที่จะนำไปใช้บนโมเดลปัญญาประดิษฐ์ที่ทำการทดสอบกับข้อมูลทดสอบ (Test Set)

ค่าพารามิเตอร์ที่ได้ทำการปรับแต่งทั้งหมดบนโมเดลปัญญาประดิษฐ์ชนิด Long Short-term Memory แสดงตามตารางที่ 4.6

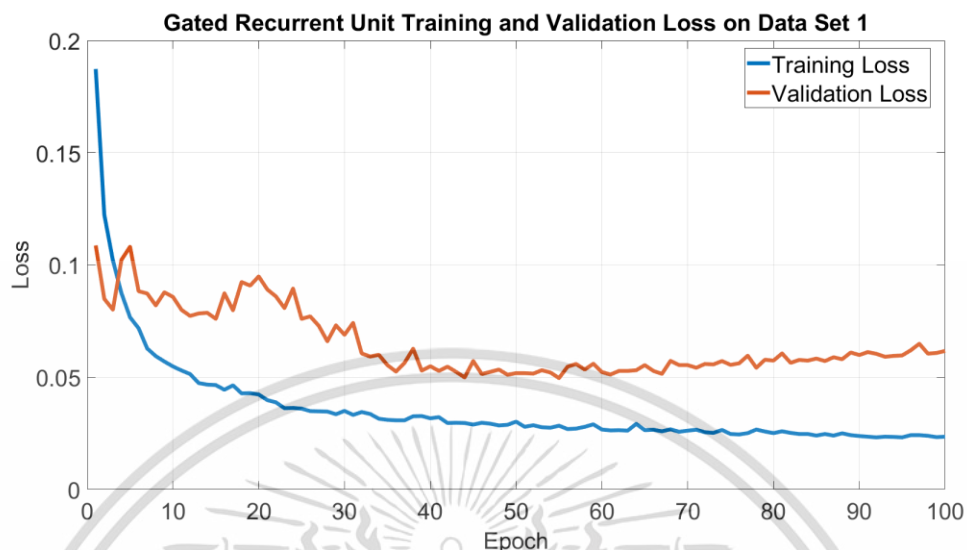
ตารางที่ 4.6 ค่าพารามิเตอร์โมเดลปัญญาประดิษฐ์ชนิด Long Short-term Memory

Hyperparameter	Value
Timestep	10
Neuron	50
Epoch	47
Batch Size	512
Loss Function	Mae
Optimizer	Adam
Activation Function	Relu
Dropout	0.2
Learning Rate	0.0005

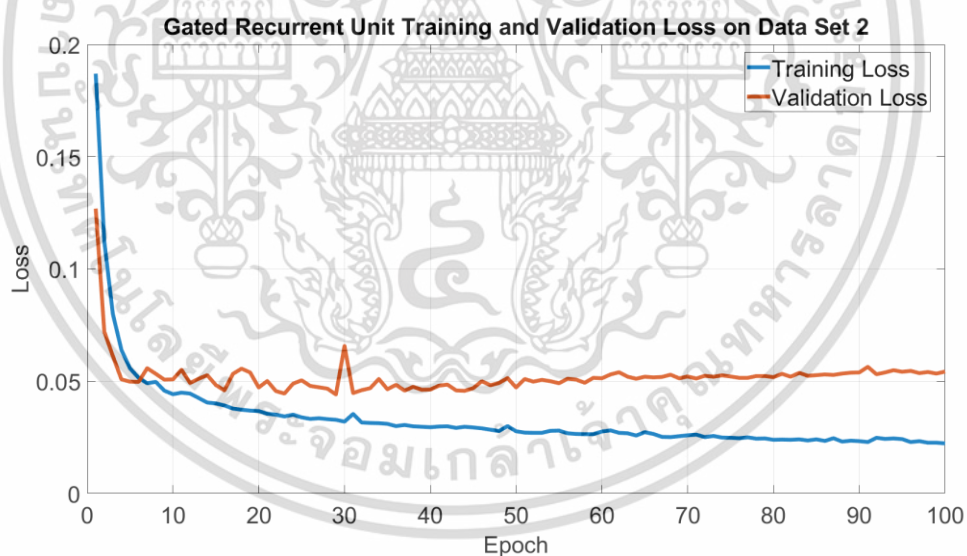
4.1.4. ค่าพารามิเตอร์โมเดล Gated Recurrent Unit

โมเดลปัญญาประดิษฐ์ชนิด Gated Recurrent Unit (GRU) ที่ได้ทำการปรับแต่ง ค่าพารามิเตอร์มีกราฟความสัมพันธ์ระหว่างความผิดพลาดเพี้ยนชุดสอน (Train Loss) ความผิดพลาดเพี้ยนชุด ตรวจสอบ (Validation Loss) กับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์ (Epoch) บนชุด ตรวจสอบ 4 ชุด ดังรูปที่ 4.13 ถึง รูปที่ 4.16 ซึ่งสามารถสังเกตจากรูปที่ 4.13 และ รูปที่ 4.14 ได้ว่า โมเดลปัญญาประดิษฐ์เริ่มเกิด Overfitting ในช่วง ตั้งแต่ประมาณ Epoch 50 เป็นต้นไป สังเกตจาก การที่ Validation Loss เริ่มลู่ออกจาก Training Loss ในส่วนของ รูปที่ 4.15 โมเดลปัญญาประดิษฐ์

มีการเรียนรู้ที่ติดตลอดช่วง Epoch และรูปที่ 4.16 โมเดลปัญญาประดิษฐ์เกิด Underfitting ในช่วง Epoch แรกๆ สังเกตจากการที่ Validation Loss มีค่าที่น้อยกว่า Training Loss

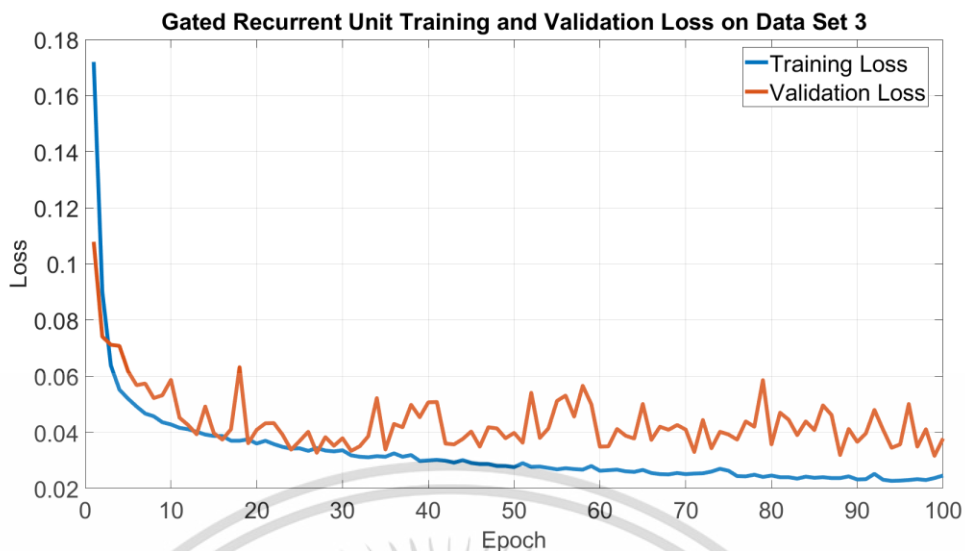


รูปที่ 4.13 กราฟความสัมพันธ์ระหว่างความผิดพลาดสอนความผิดพลาดตรวจสอบกับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 1 ของโมเดลชนิด Gated Recurrent Unit

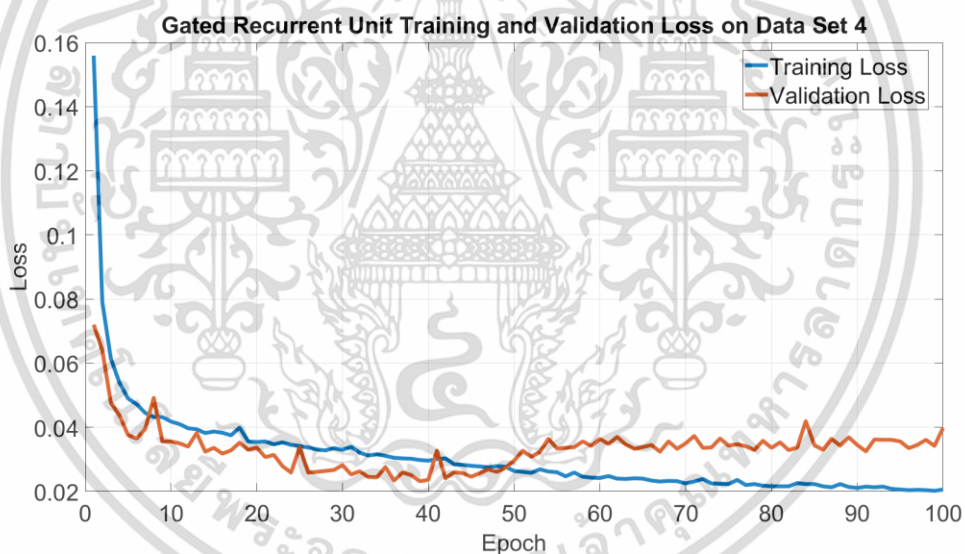


รูปที่ 4.14 กราฟความสัมพันธ์ระหว่างความผิดพลาดสอนความผิดพลาดตรวจสอบกับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 2 ของโมเดลชนิด Gated Recurrent Unit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.15 กราฟความสัมพันธ์ระหว่างความผิดพลาดสอนความผิดพลาดตรวจสอบกับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 3 ของโมเดลชนิด Gated Recurrent Unit



รูปที่ 4.16 กราฟความสัมพันธ์ระหว่างความผิดพลาดสอนความผิดพลาดตรวจสอบกับจำนวนรอบการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลตรวจสอบที่ 4 ของโมเดลชนิด Gated Recurrent Unit

ค่าสัมบูรณ์เฉลี่ยของค่าความผิดพลาด (Mean absolute error) หรือ MAE และค่ารากที่สองของค่าเฉลี่ยกำลังสองของค่าความผิดพลาด (Root mean squared error) หรือ RMSE บนชุดข้อมูลตรวจสอบทั้ง 4 ชุด แสดงตามตารางที่ 4.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.7 ค่าความผิดพลาดเพี้ยนบนชุดข้อมูลตรวจสอบของโมเดลชนิด Gated Recurrent Unit

	MAE Validation Loss (%)	RMSE Validation Loss (%)
Data Set 1	5.6175	7.6355
Data Set 2	4.9432	6.0487
Data Set 3	3.4436	4.9896
Data Set 4	3.6219	4.5327

เมื่อทำการหาค่าเฉลี่ยของความผิดพลาดเพี้ยนบนชุดตรวจสอบทั้ง 4 ชุดในแต่ละรอบการสอนโมเดล ปัญญาประดิษฐ์พบว่ารอบการสอนที่ 44 ให้ค่าความผิดพลาดเพี้ยนบนชุดตรวจสอบต่ำสุดที่ 0.0397 ซึ่งจะ เป็นรอบการสอนที่จะนำไปใช้บนโมเดลปัญญาประดิษฐ์ที่ทำการทดสอบกับข้อมูลทดสอบ (Test Set)

ค่าพารามิเตอร์ที่ได้ทำการปรับแต่งทั้งหมดบนโมเดลปัญญาประดิษฐ์ชนิด Gated Recurrent Unit แสดงตามตารางที่ 4.8

ตารางที่ 4.8 ค่าพารามิเตอร์โมเดลปัญญาประดิษฐ์ชนิด Gated Recurrent Unit

Hyperparameter	Value
Timestep	10
Neuron	125
Epoch	44
Batch Size	512
Loss Function	Mae
Optimizer	Adam
Activation Function	Relu
Dropout	0.2
Learning Rate	0.0005

4.2. ผลการทำนายและระยะเวลาในการสอนและทำนายระดับประจูปฏิบัติการ

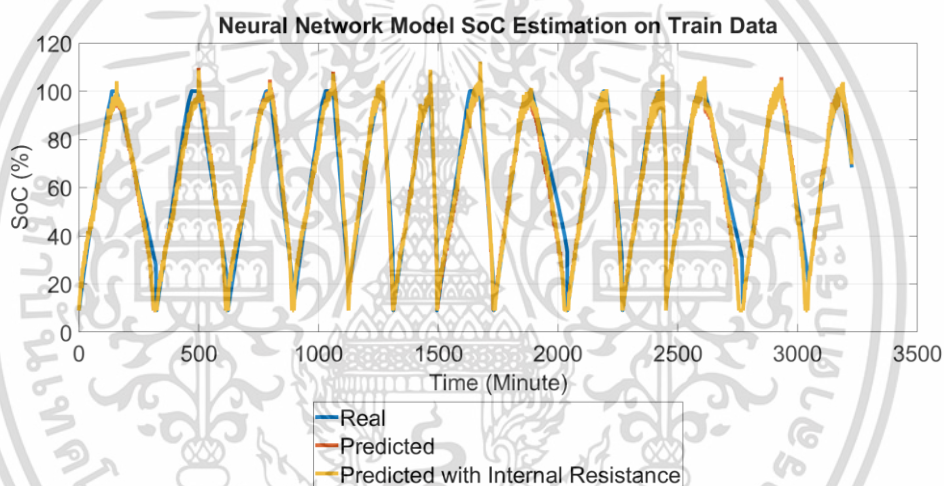
หลังจากโมเดลปัญญาประดิษฐ์มีการปรับแต่งค่าพารามิเตอร์ จะทำการทำนายระดับประจูปฏิบัติการโดยใช้ข้อมูลสอน (Train Set) และ ข้อมูลทดสอบ (Test Set) ซึ่งข้อมูลทดสอบเป็นชุดข้อมูลใหม่ที่แบ่งไว้ เพื่อทำการประเมินประสิทธิภาพของโมเดลปัญญาประดิษฐ์ในด้านความแม่นยำและระยะเวลาในการสอนและทำนายระดับประจูปฏิบัติการ ซึ่งโมเดลปัญญาประดิษฐ์ทั้ง 4 ชนิดที่ใช้ในงานวิจัยให้ผลความผิดพลาดเพี้ยนและระยะเวลาในการทำสอนและทำนายระดับประจูปฏิบัติการดังนี้

4.2.1. ผลจากการสอนและทำนายโดยโมเดล Neural Network

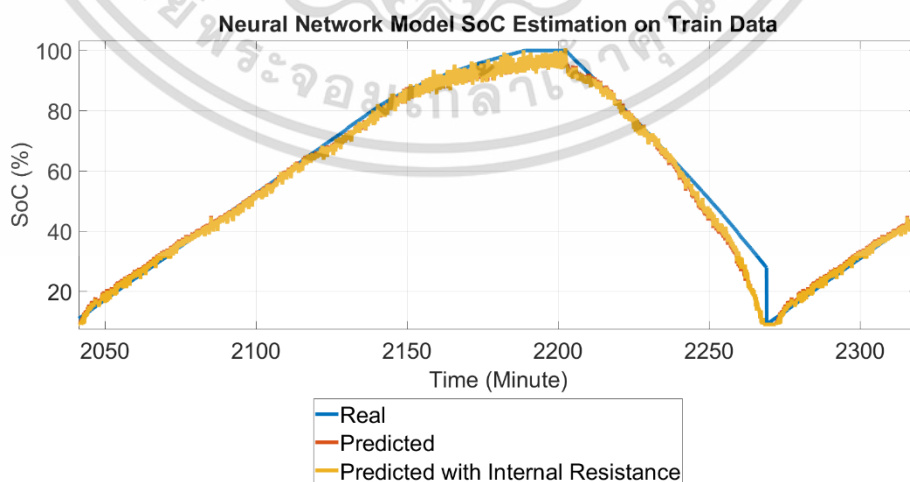
ผลจากการสอนและทำนายโดยโมเดล Neural Network สามารถแบ่งเป็น 2 ส่วนได้แก่ 1) ผลการทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลสอนและข้อมูลทดสอบ 2) ระยะเวลาในการสอนบนชุดข้อมูลสอนและระยะเวลาในการทำนายบนชุดข้อมูลสอนและข้อมูลทดสอบ

4.2.1.1. ผลการทำนายระดับประจุแบตเตอรี่โดยโมเดล Neural Network

ผลการทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลสอนโดยโมเดล Neural Network เมื่อพิจารณาค่าความต้านทานภายในและไม่พิจารณาค่าความต้านทานภายในแสดงดังรูปที่ 4.17 เมื่อทำการขยายดังรูปที่ 4.18 มีค่าความผิดพลาดตามตารางที่ 4.9 ซึ่งแสดงให้เห็นว่าโมเดลปัญญาประดิษฐ์ชนิด Neural Network มีความสามารถในการทำนายระดับประจุแบตเตอรี่ที่มีความไม่เป็นเชิงเส้นได้ โดยในกรณีที่พิจารณาค่าความต้านทานภายในมีค่าความผิดพลาดที่ต่ำกว่าแบบไม่พิจารณาค่าความต้านทานภายในตามตารางที่ 4.9



รูปที่ 4.17 ระดับประจุแบตเตอรี่ทำนายโดยโมเดล Neural Network บนชุดข้อมูลสอน



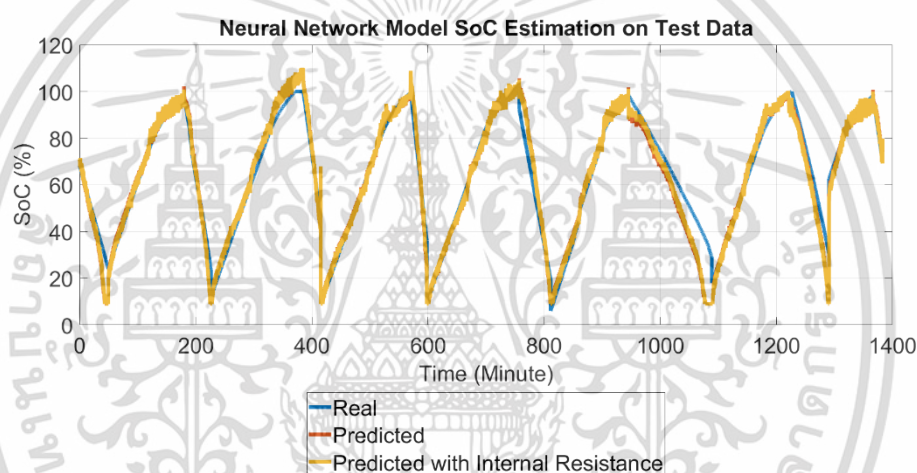
รูปที่ 4.18 ภาพขยายระดับประจุแบตเตอรี่ทำนายโดยโมเดล Neural Network บนชุดข้อมูลสอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

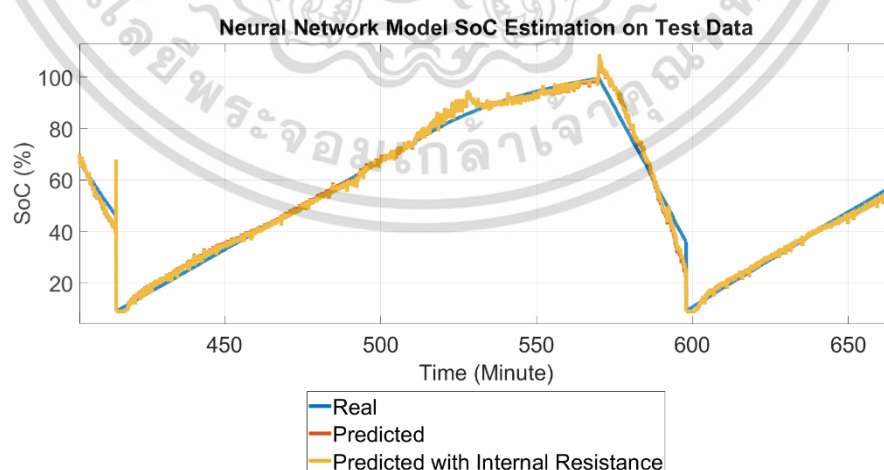
ตารางที่ 4.9 ค่าความผิดพลาดบนชุดข้อมูลสอนของโมเดล Neural Network

	MAE (%)	RMSE (%)
NN	3.9707	5.9647
NN with Internal Resistance	3.8561	5.8375

ผลการทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลทดสอบโดยโมเดล Neural Network เมื่อพิจารณาค่าความผันผวนภายในและไม่พิจารณาค่าความผันผวนภายในแสดงดังรูปที่ 4.19 เมื่อทำการขยายดังรูปที่ 4.20 เช่นเดียวกับกรณีข้อมูลสอน ในชุดข้อมูลทดสอบโมเดลปัญญาประดิษฐ์ชนิด Neural Network มีความสามารถในการทำนายระดับประจุแบตเตอรี่ที่มีความไม่เป็นเชิงเส้นได้ โดยในกรณีที่พิจารณาค่าความผันผวนภายในมีค่าความผิดพลาดที่ต่ำกว่าแบบไม่พิจารณาผันผวนภายในตามตารางที่ 4.10



รูปที่ 4.19 ระดับประจุแบตเตอรี่ทำนายโดยโมเดล Neural Network บนชุดข้อมูลทดสอบ



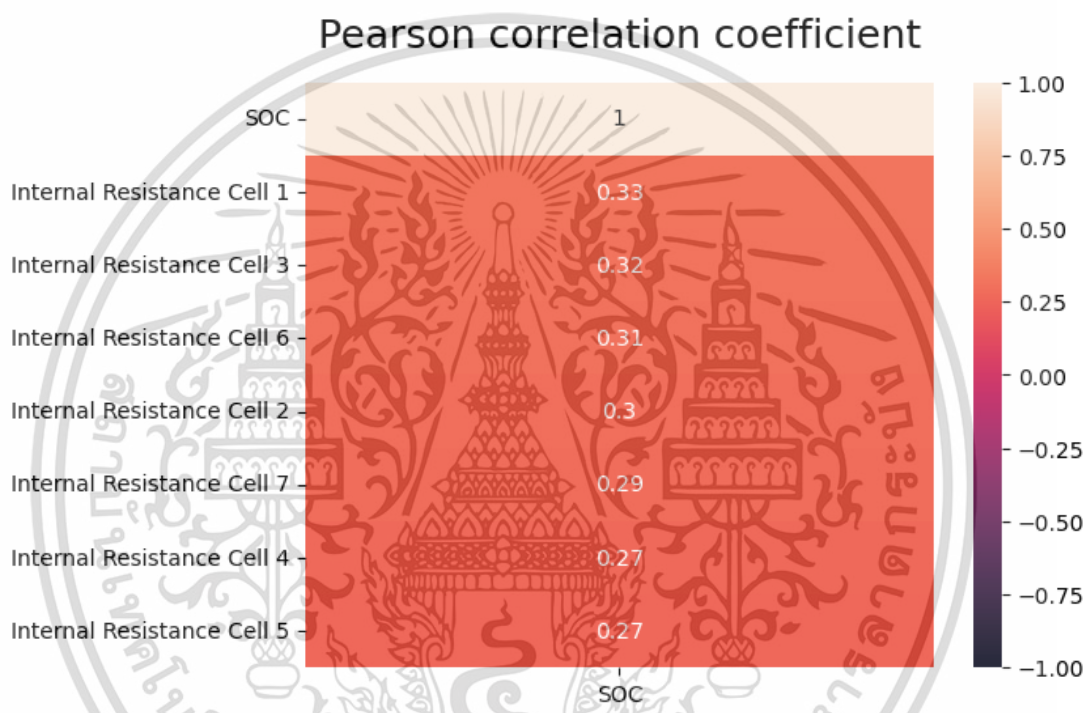
รูปที่ 4.20 ภาพขยายระดับประจุแบตเตอรี่ทำนายโดยโมเดล Neural Network บนชุดข้อมูลทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.10 ค่าความผิดพลาดบนชุดข้อมูลทดสอบของโมเดล Neural Network

	MAE (%)	RMSE (%)
NN	3.7486	5.6508
NN with Internal Resistance	3.6545	5.5442

ผลของสัมประสิทธิ์ความสัมพันธ์ระหว่างค่าความต้านทานภายในและระดับประจุแบตเตอรี่ของโมเดลปัญญาประดิษฐ์ชนิด Neural Network บนชุดข้อมูลทดสอบหาด้วยวิธี Pearson Correlation Coefficient มีผลดังรูปที่ 4.21



รูปที่ 4.21 สัมประสิทธิ์ความสัมพันธ์ระหว่างค่าความต้านทานภายในและระดับประจุแบตเตอรี่ของโมเดลปัญญาประดิษฐ์ชนิด Neural Network บนชุดข้อมูลทดสอบ

4.2.1.2. ระยะเวลาในการสอนและทำนายผลโดยโมเดล Neural Network

เมื่อพิจารณาระยะเวลาที่ใช้ในการสอนโมเดลปัญญาประดิษฐ์ Neural Network และทำนายบนชุดข้อมูลสอนและชุดข้อมูลทดสอบเป็นไปตามตารางที่ 4.11 ทำการทำนายบนชุดสอนทั้งหมด 38,714 ข้อมูล และทำการทำนายบนชุดทดสอบทั้งหมด 16,593 ข้อมูล เนื่องจากโมเดลปัญญาประดิษฐ์ที่พิจารณาค่าความต้านทานภายในต้องทำการคำนวณค่าความต้านทานภายในจากการทำนายค่าแรงดันเปิดวงจรก่อนแล้วจึงทำนายระดับประจุแบตเตอรี่เพราะฉะนั้นจึงใช้เวลามากกว่าโมเดลปัญญาประดิษฐ์ที่ไม่พิจารณาค่าความต้านทานภายในประมาณ 2 เท่า

ตารางที่ 4.11 ระยะเวลาการสอนและทำนายของโมเดล Neural Network

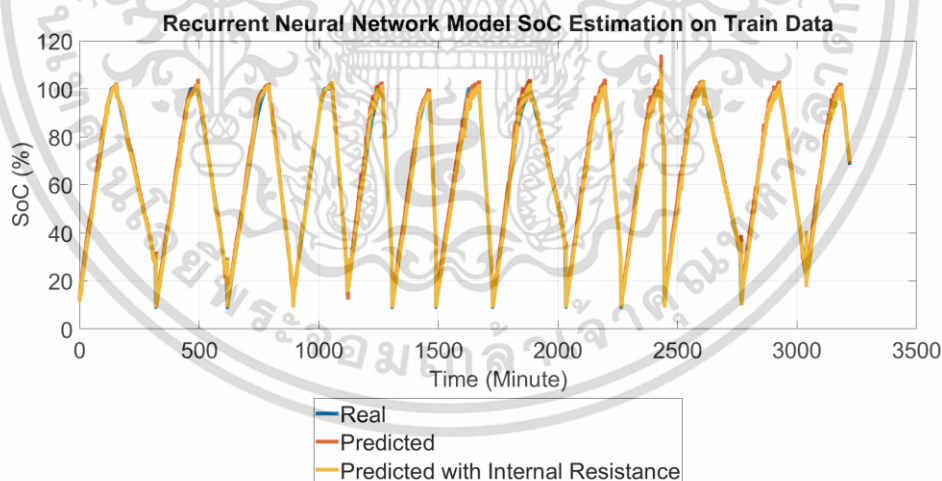
	Training Time (s)	Testing on Train Set Time (s)	Testing on Test Set Time (s)
NN	10.16	4.66	2.04
NN with Internal Resistance	19.79	6.83	3.10

4.2.2. ผลจากการสอนและทำนายโดยโมเดล Recurrent Neural Network

ผลจากการสอนและทำนายโดยโมเดล Recurrent Neural Network สามารถแบ่งเป็น 2 ส่วนได้แก่ 1) ผลการทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลสอนและข้อมูลทดสอบ 2) ระยะเวลาในการสอนบนชุดข้อมูลสอนและระยะเวลาในการทำนายบนชุดข้อมูลสอนและข้อมูลทดสอบ

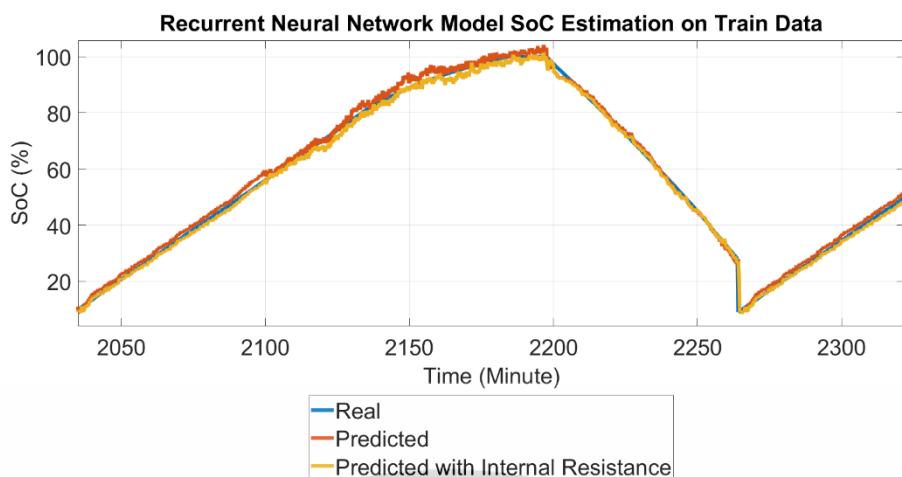
4.2.2.1. ผลการทำนายระดับประจุแบตเตอรี่โดยโมเดล Recurrent Neural Network

ผลการทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลสอนโดยโมเดล Recurrent Neural Network เมื่อพิจารณาค่าความต้านทานภายในและไม่พิจารณาค่าความต้านทานภายในแสดงดังรูปที่ 4.22 เมื่อทำการขยายดังรูปที่ 4.23 มีค่าความผิดพลาดตามตารางที่ 4.12 ซึ่งแสดงให้เห็นว่าโมเดลปัญญาประดิษฐ์ชนิด Recurrent Neural Network มีความสามารถในการทำนายระดับประจุแบตเตอรี่ที่มีความไม่เป็นเชิงเส้นได้ โดยในกรณีที่พิจารณาค่าความต้านทานภายในมีค่าความผิดพลาดที่ต่ำกว่าแบบไม่พิจารณาค่าความต้านทานภายในตามตารางที่ 4.12



รูปที่ 4.22 ระดับประจุแบตเตอรี่ทำนายโดยโมเดล Recurrent Neural Network บนชุดข้อมูลสอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

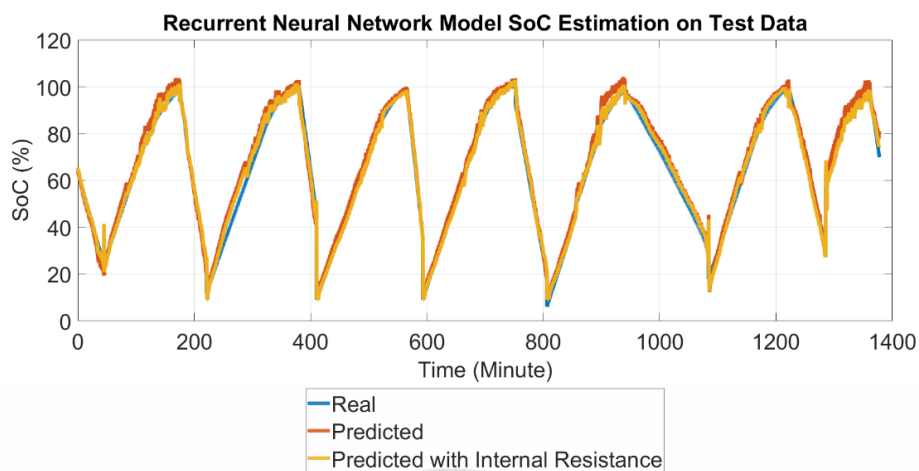


รูปที่ 4.23 ภาพขยายระดับประจุแบตเตอรี่ที่ทำนายโดยโมเดล Recurrent Neural Network บนชุดข้อมูลสอน

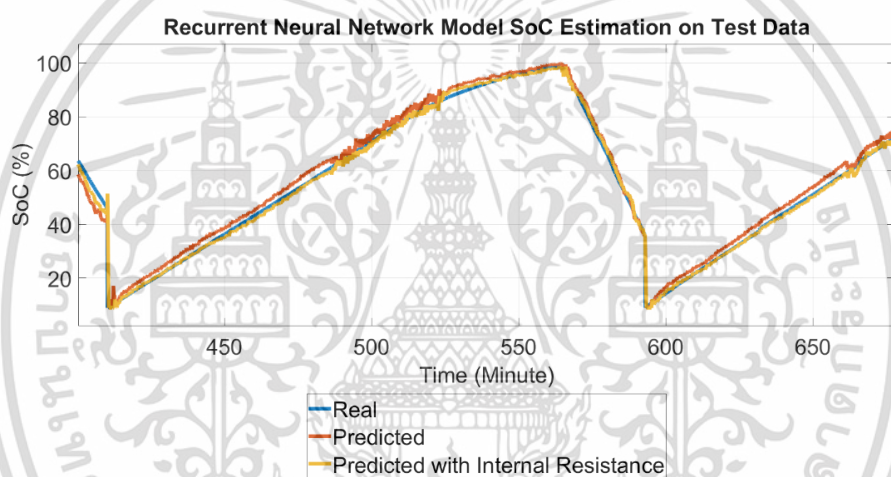
ตารางที่ 4.12 ค่าความผิดพลาดบนชุดข้อมูลสอนของโมเดล Recurrent Neural Network

	MAE (%)	RMSE (%)
RNN	1.6832	2.2456
RNN with Internal Resistance	1.0866	1.6145

ผลการทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลทดสอบโดยโมเดล Recurrent Neural Network เมื่อพิจารณาค่าความต้านทานภายในและไม่พิจารณาค่าความต้านทานภายในแสดงดังรูปที่ 4.24 เมื่อทำการขยายดังรูปที่ 4.25 เช่นเดียวกับกรณีข้อมูลสอน ในชุดข้อมูลทดสอบโมเดลปัญญาประดิษฐ์ชนิด Recurrent Neural Network มีความสามารถในการทำนายระดับประจุแบตเตอรี่ที่มีความไม่เป็นเชิงเส้นได้ โดยในกรณีที่พิจารณาค่าความต้านทานภายในมีค่าความผิดพลาดที่ต่ำกว่าแบบไม่พิจารณาต้านทานภายในตามตารางที่ 4.13



รูปที่ 4.24 ระดับประจุแบตเตอรี่ทำนายโดยโมเดล Recurrent Neural Network บนชุดข้อมูลทดสอบ



รูปที่ 4.25 ภาพขยายระดับประจุแบตเตอรี่ทำนายโดยโมเดล Recurrent Neural Network บนชุดข้อมูลทดสอบ

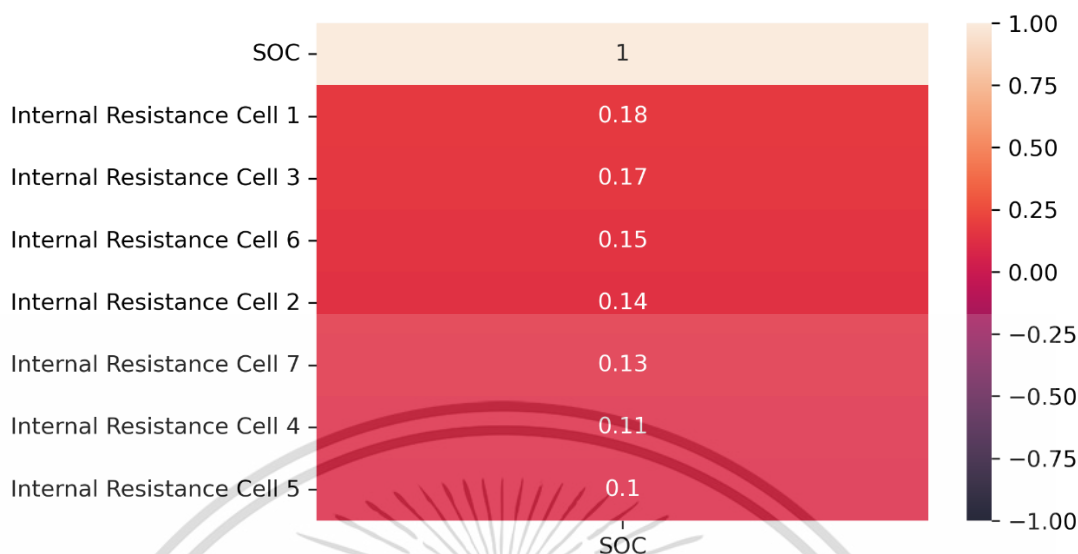
ตารางที่ 4.13 ค่าความผิดพลาดบนชุดข้อมูลทดสอบของโมเดล Recurrent Neural Network

	MAE (%)	RMSE (%)
RNN	3.1601	3.7954
RNN with Internal Resistance	1.7919	2.4589

ผลของสัมประสิทธิ์ความสัมพันธ์ระหว่างค่าความต้านทานภายในและระดับประจุแบตเตอรี่ของโมเดลปัญญาประดิษฐ์ชนิด Recurrent Neural Network บนชุดข้อมูลทดสอบหาด้วยวิธี Pearson Correlation Coefficient มีผลดังรูปที่ 4.26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pearson correlation coefficient



รูปที่ 4.26 สัมประสิทธิ์ความสัมพันธ์ระหว่างค่าความต้านทานภายในและระดับประจุแบตเตอรี่ของโมเดลปัญญาประดิษฐ์ชนิด Recurrent Neural Network บนชุดข้อมูลทดสอบ

4.2.2.2. ระยะเวลาในการสอนและทำนายผลโดยโมเดล Recurrent Neural Network

เมื่อพิจารณาระยะเวลาที่ใช้ในการสอนโมเดลปัญญาประดิษฐ์ Recurrent Neural Network และทำนายบนชุดข้อมูลสอนและชุดข้อมูลทดสอบเป็นไปตามตารางที่ 4.14 ทำการสอนบนชุดสอนทั้งหมด 38,714 ข้อมูล และทำการทำนายบนชุดทดสอบทั้งหมด 16,593 ข้อมูล เนื่องจากโมเดลปัญญาประดิษฐ์ที่พิจารณาค่าความต้านทานภายในต้องทำการคำนวณค่าความต้านทานภายในจากการทำนายค่าแรงดันเปิดวงจรก่อนแล้วจึงทำนายระดับประจุแบตเตอรี่เพราะฉะนั้นจึงใช้เวลามากกว่าโมเดลปัญญาประดิษฐ์ที่ไม่พิจารณาค่าความต้านทานภายในประมาณ 2 เท่า

ตารางที่ 4.14 ระยะเวลาการสอนและทำนายของโมเดล Recurrent Neural Network

	Training Time (s)	Testing on Train Set Time (s)	Testing on Test Set Time (s)
RNN	344.10	7.99	3.58
RNN with Internal Resistance	715.15	15.39	6.73

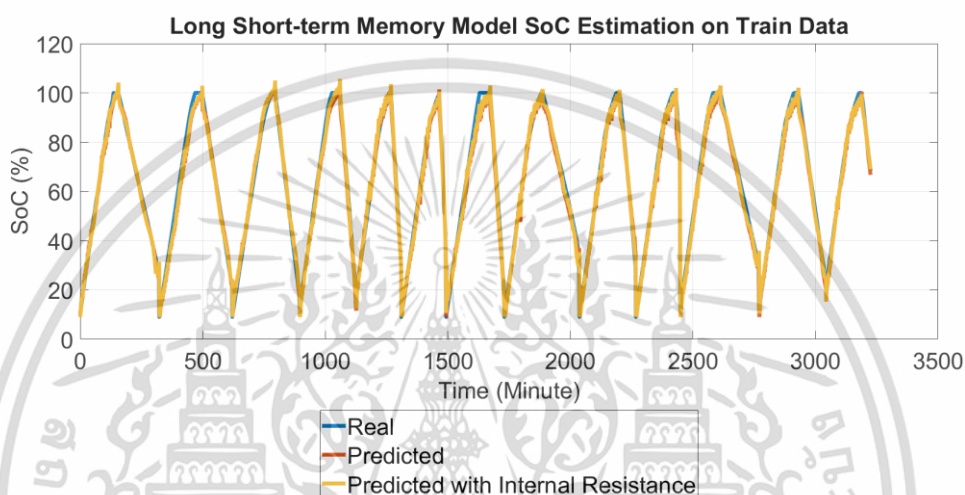
4.2.3. ผลและระยะเวลาการทำนายโดยโมเดล Long Short-term Memory

ผลจากการสอนและทำนายโดยโมเดล Long Short-term Memory สามารถแบ่งเป็น 2 ส่วนได้แก่ 1) ผลการทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลสอนและข้อมูลทดสอบ 2) ระยะเวลาในการสอนบนชุดข้อมูลสอนและระยะเวลาในการทำนายบนชุดข้อมูลสอนและข้อมูลทดสอบ

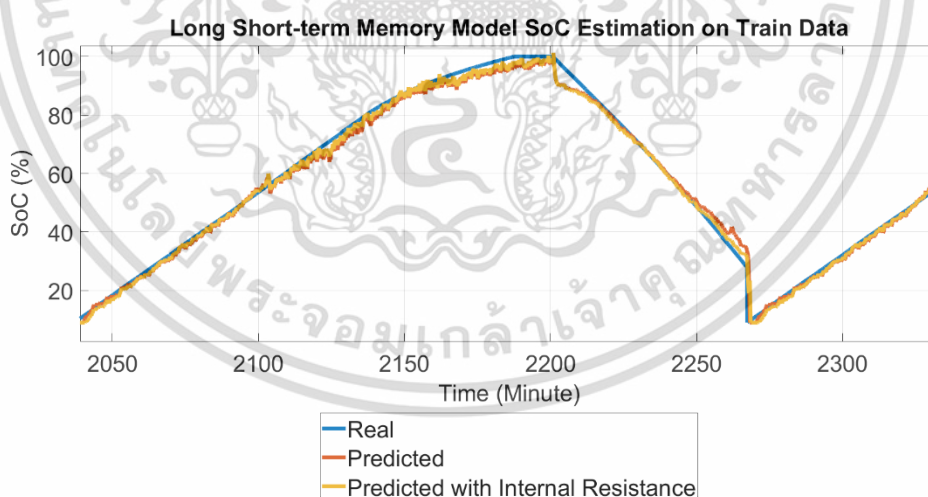
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.3.1. ผลการทำนายระดับประจุแบตเตอรี่โดยโมเดล Long Short-term Memory

ผลการทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลสอนโดยโมเดล Long Short-term Memory เมื่อพิจารณาค่าความต้านทานภายในและไม่พิจารณาค่าความต้านทานภายในแสดงดังรูปที่ 4.27 เมื่อทำการขยายดังรูปที่ 4.28 มีค่าความผิดพลาดตามตารางที่ 4.15 ซึ่งแสดงให้เห็นว่าโมเดลปัญญาประดิษฐ์ชนิด Long Short-term Memory มีความสามารถในการทำนายระดับประจุแบตเตอรี่ที่มีความไม่เป็นเชิงเส้นได้ โดยในกรณีที่พิจารณาค่าความต้านทานภายในมีค่าความผิดพลาดที่ต่ำกว่าแบบไม่พิจารณาค่าความต้านทานภายในตามตารางที่ 4.15



รูปที่ 4.27 ระดับประจุแบตเตอรี่ทำนายโดยโมเดล Long Short-term Memory บนชุดข้อมูลสอน



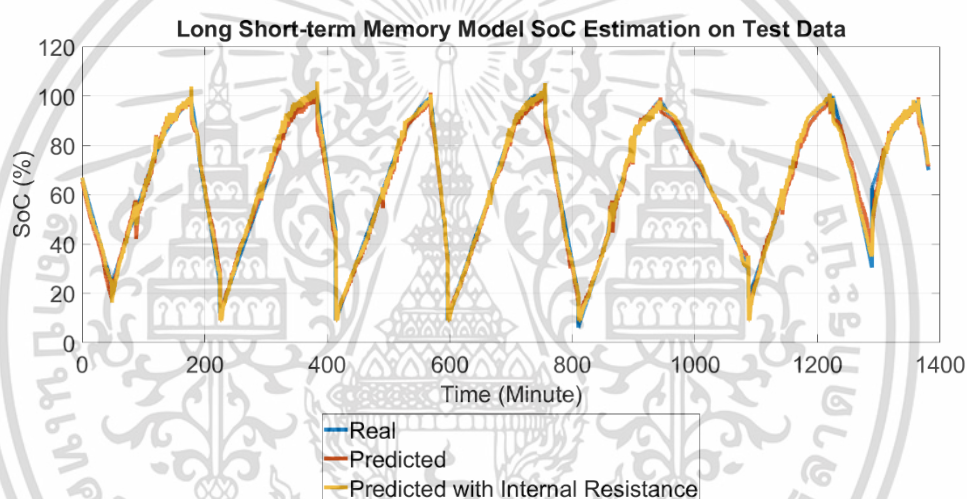
รูปที่ 4.28 ภาพขยายระดับประจุแบตเตอรี่ทำนายโดยโมเดล Long Short-term Memory บนชุดข้อมูลสอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

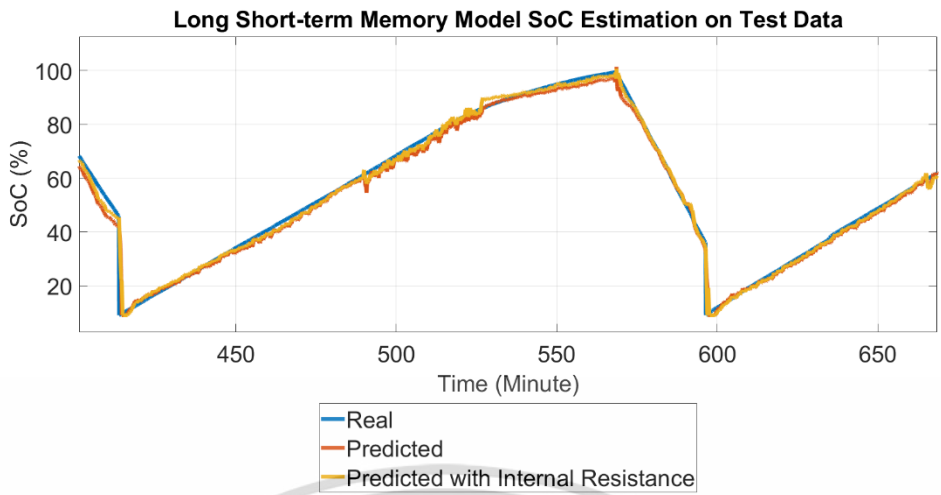
ตารางที่ 4.15 ค่าความผิดพลาดบนชุดข้อมูลสอนของโมเดล Long Short-term Memory

	MAE (%)	RMSE (%)
LSTM	2.0186	2.6539
LSTM with Internal Resistance	1.6163	2.2190

ผลการทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลทดสอบโดยโมเดล Long Short-term Memory เมื่อพิจารณาค่าความผันผวนภายในและไม่พิจารณาค่าความผันผวนภายในแสดงดังรูปที่ 4.29 เมื่อทำการขยายดังรูปที่ 4.30 เช่นเดียวกับกรณีข้อมูลสอน ในชุดข้อมูลทดสอบโมเดล ปัญญาประดิษฐ์ชนิด Long Short-term Memory มีความสามารถในการทำนายระดับประจุแบตเตอรี่ที่มีความไม่เป็นเชิงเส้นได้ โดยในกรณีที่พิจารณาค่าความผันผวนภายในมีค่าความผิดพลาดที่ต่ำกว่าแบบไม่พิจารณาผันผวนภายในตามตารางที่ 4.16



รูปที่ 4.29 ระดับประจุแบตเตอรี่ทำนายโดยโมเดล Long Short-term Memory บนชุดข้อมูลทดสอบ

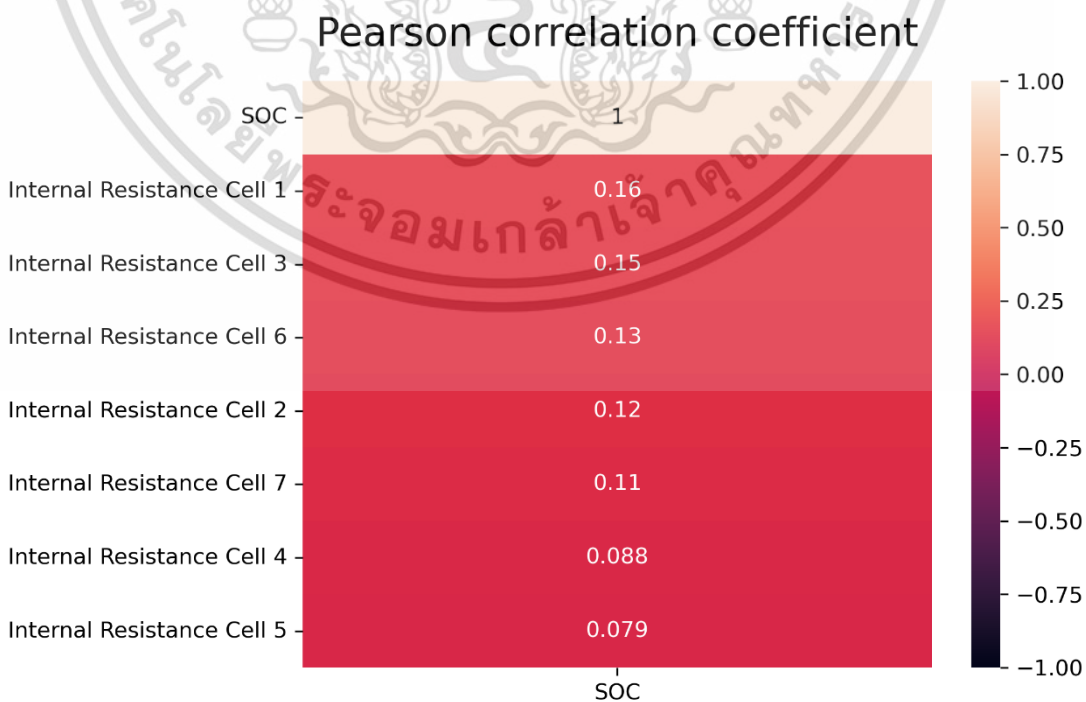


รูปที่ 4.30 ภาพขยายระดับประจุแบตเตอรี่ทำนายโดยโมเดล Long Short-term Memory บนชุดข้อมูลทดสอบ

ตารางที่ 4.16 ค่าความผิดพลาดบนชุดข้อมูลทดสอบของโมเดล Long Short-term Memory

	MAE (%)	RMSE (%)
LSTM	1.9776	2.8309
LSTM with Internal Resistance	1.7345	2.4789

ผลของสัมประสิทธิ์ความสัมพันธ์ระหว่างค่าความต้านทานภายในและระดับประจุแบตเตอรี่ของโมเดลปัญญาประดิษฐ์ชนิด Long Short-term Memory บนชุดข้อมูลทดสอบหาด้วยวิธี Pearson Correlation Coefficient มีผลดังรูปที่ 4.31



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.31 สัมประสิทธิ์ความสัมพันธ์ระหว่างค่าความต้านทานภายในและระดับประจุแบตเตอรี่ของโมเดลปัญญาประดิษฐ์ชนิด Long Short-term Memory บนชุดข้อมูลทดสอบ

4.2.3.2. ระยะเวลาในการสอนและทำนายผลโดยโมเดล Long Short-term Memory

เมื่อพิจารณาระยะเวลาที่ใช้ในการสอนโมเดลปัญญาประดิษฐ์ Long Short-term Memory และทำนายบนชุดข้อมูลสอนและชุดข้อมูลทดสอบเป็นไปตามตารางที่ 4.17 ทำการสอนบนชุดสอนทั้งหมด 38,714 ข้อมูล และทำการทำนายบนชุดทดสอบทั้งหมด 16,593 ข้อมูล เนื่องจากโมเดลปัญญาประดิษฐ์ที่พิจารณาค่าความต้านทานภายในต้องทำการคำนวณค่าความต้านทานภายในจากการทำนายค่าแรงดันเปิดวงจรก่อนแล้วจึงทำนายระดับประจุแบตเตอรี่เพราะฉะนั้นจึงใช้เวลามากกว่าโมเดลปัญญาประดิษฐ์ที่ไม่พิจารณาค่าความต้านทานภายในประมาณ 2 เท่า

ตารางที่ 4.17 ระยะเวลาการสอนและทำนายของโมเดล Long Short-term Memory

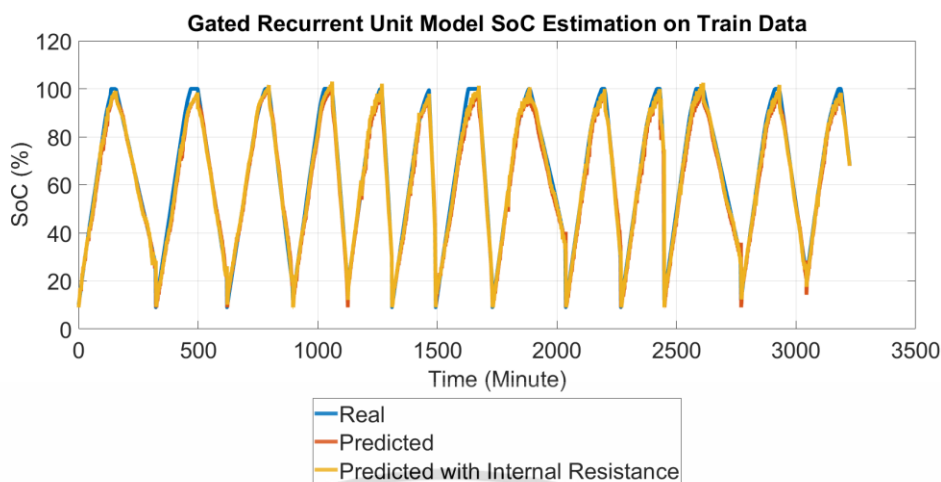
	Training Time (s)	Testing on Train Set Time (s)	Testing on Test Set Time (s)
LSTM	24.60	5.38	2.47
LSTM with Internal Resistance	50.62	8.58	3.96

4.2.4. ผลและระยะเวลาการทำนายโดยโมเดล Gated Recurrent Unit

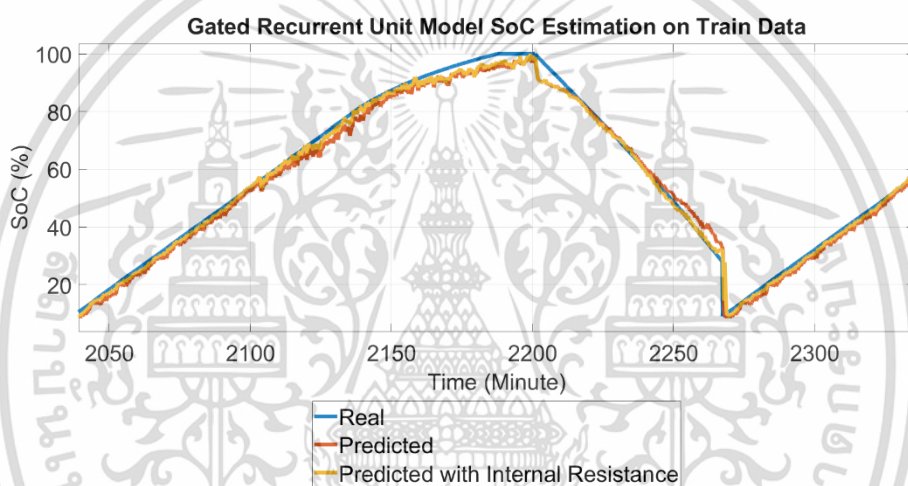
ผลจากการสอนและทำนายโดยโมเดล Gated Recurrent Unit สามารถแบ่งเป็น 2 ส่วน ได้แก่ 1) ผลการทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลสอนและข้อมูลทดสอบ 2) ระยะเวลาในการสอนบนชุดข้อมูลสอนและระยะเวลาในการทำนายบนชุดข้อมูลสอนและข้อมูลทดสอบ

4.2.4.1. ผลการทำนายระดับประจุแบตเตอรี่โดยโมเดล Gated Recurrent Unit

ผลการทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลสอนโดยโมเดล Gated Recurrent Unit เมื่อพิจารณาค่าความต้านทานภายในและไม่พิจารณาค่าความต้านทานภายในแสดงดังรูปที่ 4.32 เมื่อทำการขยายดังรูปที่ 4.33 มีค่าความผิดพลาดเพียงตามตารางที่ 4.18 ซึ่งแสดงให้เห็นว่าโมเดลปัญญาประดิษฐ์ชนิด Gated Recurrent Unit มีความสามารถในการทำนายระดับประจุแบตเตอรี่ที่มีความไม่เป็นเชิงเส้นได้ โดยในกรณีที่พิจารณาค่าความต้านทานภายในมีค่าความผิดพลาดที่ต่ำกว่าแบบไม่พิจารณาค่าความต้านทานภายในตามตารางที่ 4.18



รูปที่ 4.32 ระดับประจุแบตเตอรี่ที่ทำนายโดยโมเดล Gated Recurrent Unit บนชุดข้อมูลสอน



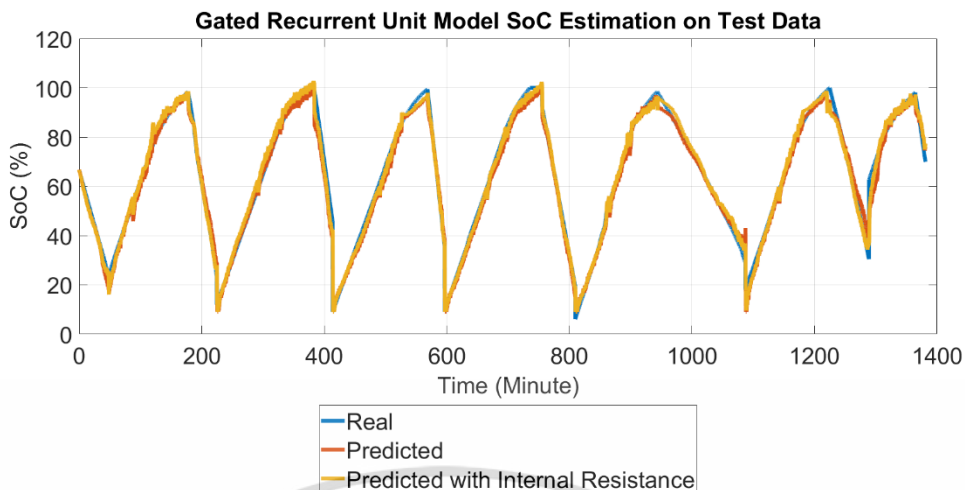
รูปที่ 4.33 ภาพขยายระดับประจุแบตเตอรี่ที่ทำนายโดยโมเดล Gated Recurrent Unit บนชุดข้อมูลสอน

ตารางที่ 4.18 ค่าความผิดพลาดบนชุดข้อมูลสอนของโมเดล Gated Recurrent Unit

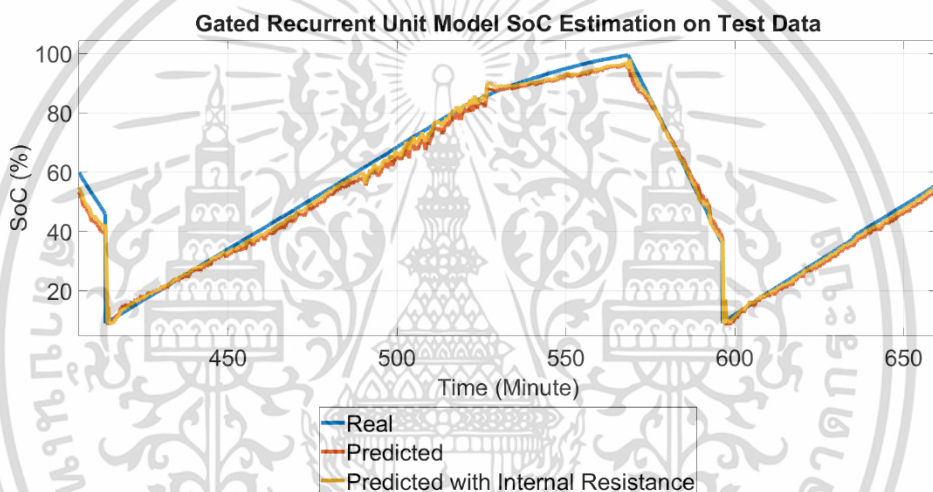
	MAE (%)	RMSE (%)
GRU	2.5386	3.2151
GRU with Internal Resistance	1.7755	2.4277

ผลการทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลทดสอบโดยโมเดล Gated Recurrent Unit เมื่อพิจารณาค่าความผันผวนภายในและไม่พิจารณาค่าความผันผวนภายในแสดงดังรูปที่ 4.34 เมื่อทำการขยายดังรูปที่ 4.35 เช่นเดียวกับกรณีข้อมูลสอน ในชุดข้อมูลทดสอบโมเดลปัญญาประดิษฐ์ชนิด Gated Recurrent Unit มีความสามารถในการทำนายระดับประจุแบตเตอรี่ที่มีความไม่เป็นเชิงเส้นได้ โดยในกรณีที่พิจารณาค่าความผันผวนภายในมีค่าความผิดพลาดที่ต่ำกว่าแบบไม่พิจารณาผันผวนภายในตามตารางที่ 4.19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.34 ระดับประจุแบตเตอรี่ทำนายโดยโมเดล Gated Recurrent Unit บนชุดข้อมูลทดสอบ



รูปที่ 4.35 ภาพขยายระดับประจุแบตเตอรี่ทำนายโดยโมเดล Gated Recurrent Unit บนชุดข้อมูลทดสอบ

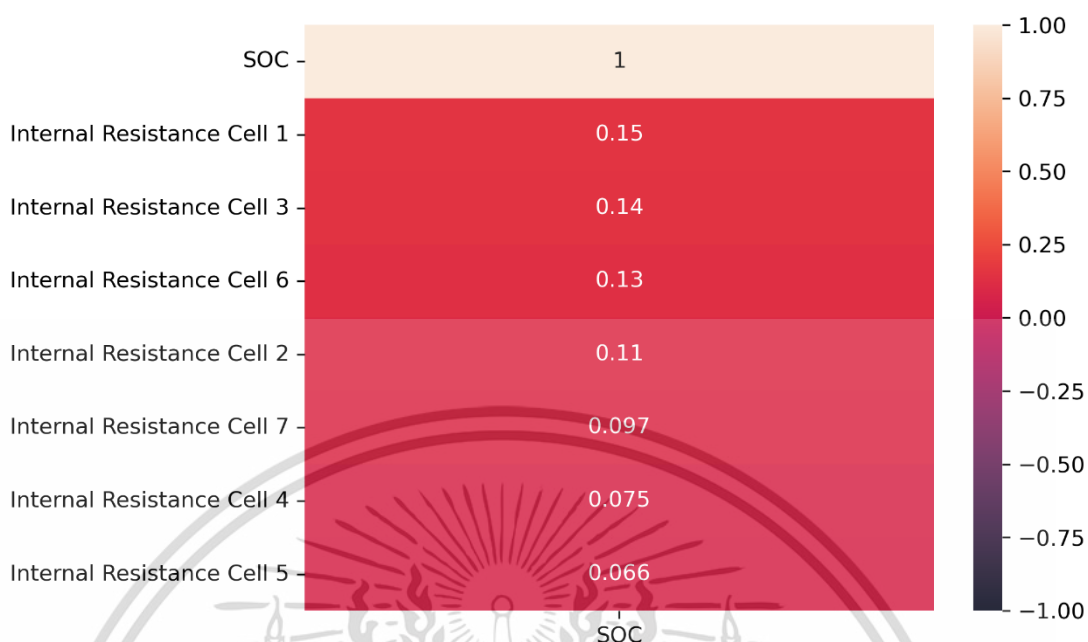
ตารางที่ 4.19 ค่าความผิดพลาดบนชุดข้อมูลทดสอบของโมเดล Gated Recurrent Unit

	MAE (%)	RMSE (%)
GRU	2.1906	3.0416
GRU with Internal Resistance	1.7726	2.4609

ผลของสัมประสิทธิ์ความสัมพันธ์ระหว่างค่าความต้านทานภายในและระดับประจุแบตเตอรี่ของโมเดลปัญญาประดิษฐ์ชนิด Gated Recurrent Unit บนชุดข้อมูลทดสอบหาด้วยวิธี Pearson Correlation Coefficient มีผลดังรูปที่ 4.36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pearson correlation coefficient



รูปที่ 4.36 สัมประสิทธิ์ความสัมพันธ์ระหว่างค่าความต้านทานภายในและระดับประจุแบตเตอรี่ของโมเดลปัญญาประดิษฐ์ชนิด Gated Recurrent Unit บนชุดข้อมูลทดสอบ

4.2.4.2. ระยะเวลาในการสอนและทำนายผลโดยโมเดล Gated Recurrent Unit

เมื่อพิจารณาระยะเวลาที่ใช้ในการสอนโมเดลปัญญาประดิษฐ์ Gated Recurrent Unit และทำนายบนชุดข้อมูลสอนและชุดข้อมูลทดสอบเป็นไปตามตารางที่ 4.20 ทำการสอนบนชุดสอนทั้งหมด 38,714 ข้อมูล และทำการทำนายบนชุดทดสอบทั้งหมด 16,593 ข้อมูล เนื่องจากโมเดลปัญญาประดิษฐ์ที่พิจารณาค่าความต้านทานภายในต้องทำการคำนวณค่าความต้านทานภายในจากการทำนายค่าแรงดันเปิดวงจรก่อนแล้วจึงทำนายระดับประจุแบตเตอรี่เพราะฉะนั้นจึงใช้เวลามากกว่าโมเดลปัญญาประดิษฐ์ที่ไม่พิจารณาค่าความต้านทานภายในประมาณ 2 เท่า

ตารางที่ 4.20 ระยะเวลาการสอนและทำนายของโมเดล Gated Recurrent Unit

	Training Time (s)	Testing on Train Set Time (s)	Testing on Test Set Time (s)
GRU	23.44	5.12	2.42
GRU with Internal Resistance	48.04	7.52	3.71

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3. เปรียบเทียบผลของโมเดลปัญญาประดิษฐ์ที่ต่างชนิด

การเปรียบเทียบประสิทธิภาพของโมเดลปัญญาประดิษฐ์ต่างชนิดในงานวิจัยนี้สามารถเปรียบเทียบได้ด้วยค่าความผิดพลาดในการทำนายระดับประจุแบตเตอรี่และระยะเวลาการสอนและทำนายระดับประจุแบตเตอรี่

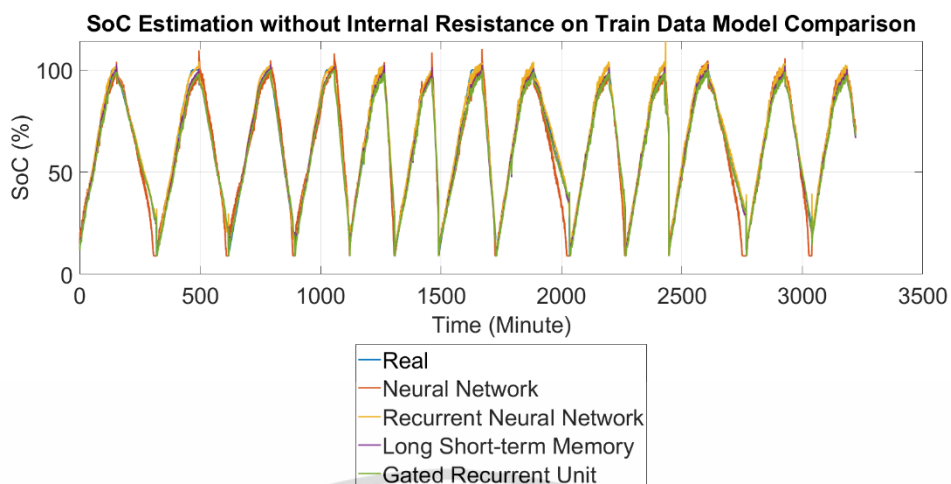
4.3.1. เปรียบเทียบค่าความผิดพลาดในการทำนายระดับประจุแบตเตอรี่

ค่าความผิดพลาดในการทำนายระดับประจุแบตเตอรี่ในงานวิจัยนี้เป็นไปตามตารางที่ 4.21 ในทุกโมเดลในงานวิจัยที่พิจารณาค่าความต้านทานภายในในร่วมด้วยจะสามารถลดความผิดพลาดในการทำนายประจุได้ทั้งบนชุดข้อมูลสอนและข้อมูลทดสอบโดยโมเดล Recurrent Neural Network ที่พิจารณาค่าความต้านทานภายในให้ค่าความผิดพลาดต่ำสุดบนชุดข้อมูลสอน ส่วนความผิดพลาดบนชุดทดสอบ ทั้ง 3 โมเดลที่มีการป้อนกลับได้แก่ Recurrent Neural Network, Long Short-term Memory และ Gated Recurrent Unit ที่พิจารณาค่าความต้านทานภายในในงานวิจัยให้ค่าความผิดพลาดที่ใกล้เคียงกัน โดยโมเดล Recurrent Neural Network ที่พิจารณาค่าความต้านทานภายในให้ความผิดพลาดต่ำสุดบนชุดทดสอบเมื่อคำนวณโดย RMSE และโมเดล Long Short-term Memory ที่พิจารณาค่าความต้านทานภายใน ให้ค่าความผิดพลาดบนชุดทดสอบต่ำสุดเมื่อคำนวณโดย MAE

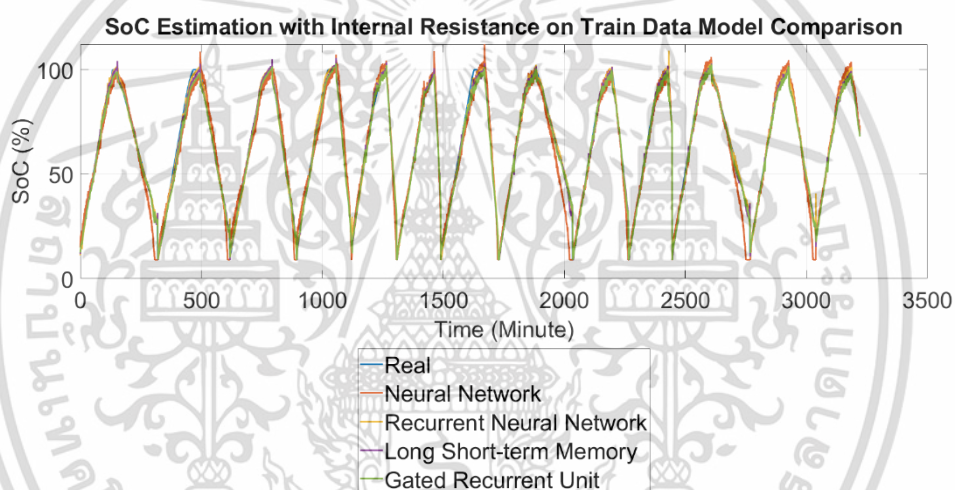
ตารางที่ 4.21 ค่าความผิดพลาดระหว่างโมเดลปัญญาประดิษฐ์ต่างชนิดบนชุดข้อมูลสอนและทดสอบ

	Train Error		Test Error	
	MAE (%)	RMSE (%)	MAE (%)	RMSE (%)
NN	3.9707	5.9647	3.7486	5.6508
NN with Internal Resistance	3.8561	5.8375	3.6545	5.5442
RNN	1.6832	2.2456	3.1601	3.7954
RNN with Internal Resistance	1.0866	1.6145	1.7919	2.4589
LSTM	2.0186	2.6539	1.9776	2.8309
LSTM with Internal Resistance	1.6163	2.2190	1.7345	2.4789
GRU	2.5386	3.2151	2.1906	3.0416
GRU with Internal Resistance	1.7755	2.4277	1.7726	2.4609

ซึ่งสามารถเปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลสอนได้ด้วยรูปดังรูปที่ 4.37 ถึง รูปที่ 4.40 และสามารถเปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลทดสอบได้ด้วยรูปดังรูปที่ 4.41 ถึง รูปที่ 4.44

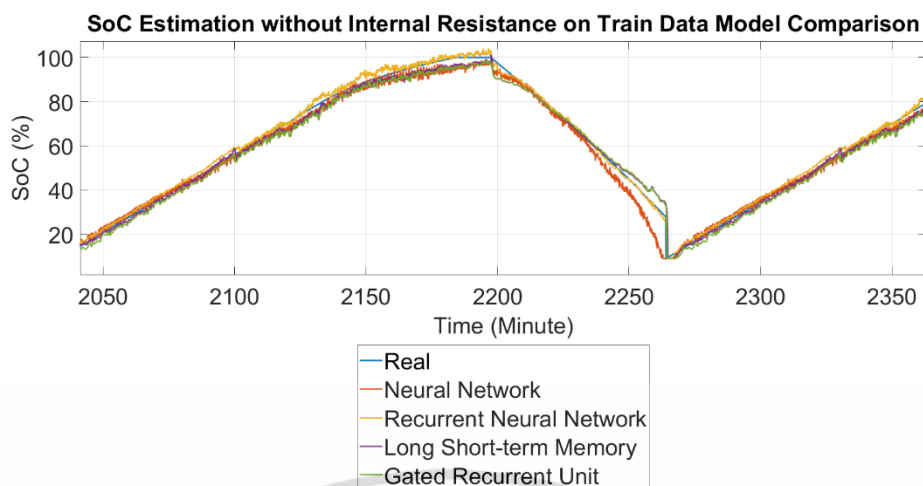


รูปที่ 4.37 เปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลสอนเมื่อไม่พิจารณาค่าความต้านทานภายใน

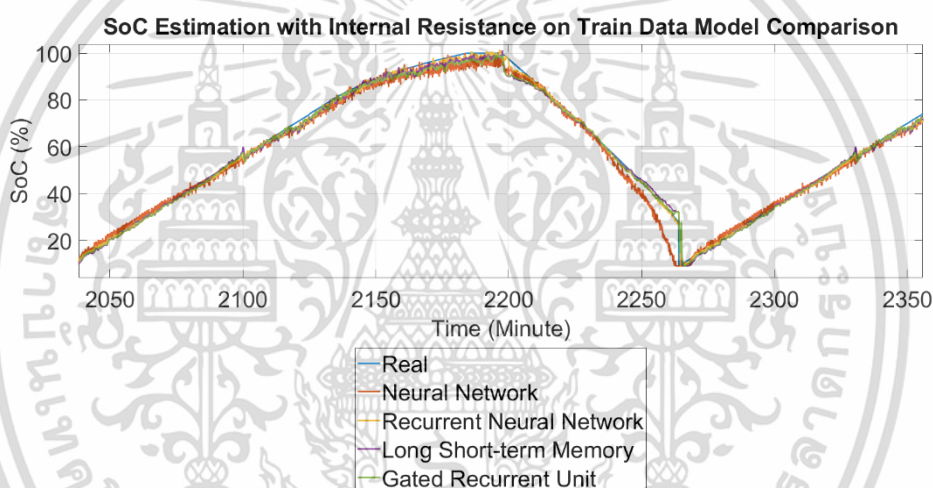


รูปที่ 4.38 ภาพขยายเปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลสอนเมื่อพิจารณาค่าความต้านทานภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

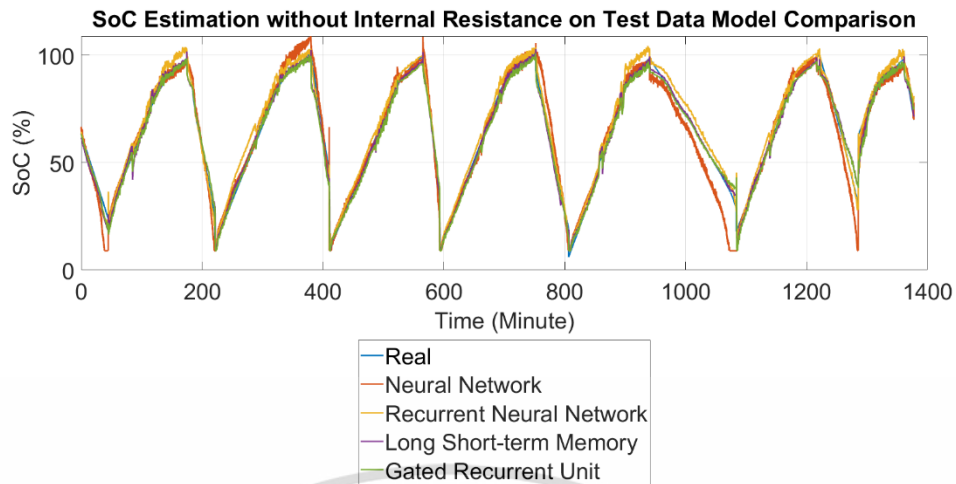


รูปที่ 4.39 ภาพขยายเปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลสอนเมื่อไม่พิจารณาค่าความต้านทานภายใน

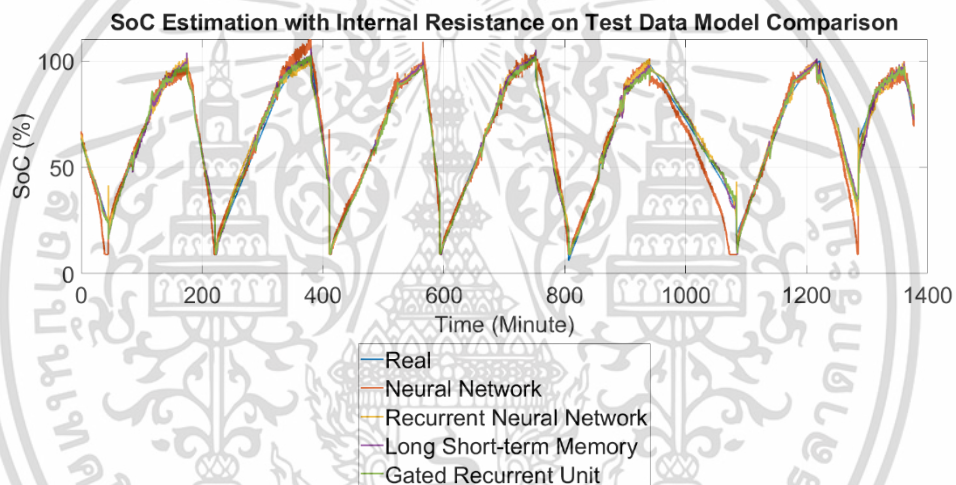


รูปที่ 4.40 ภาพขยายเปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลสอนเมื่อพิจารณาค่าความต้านทานภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

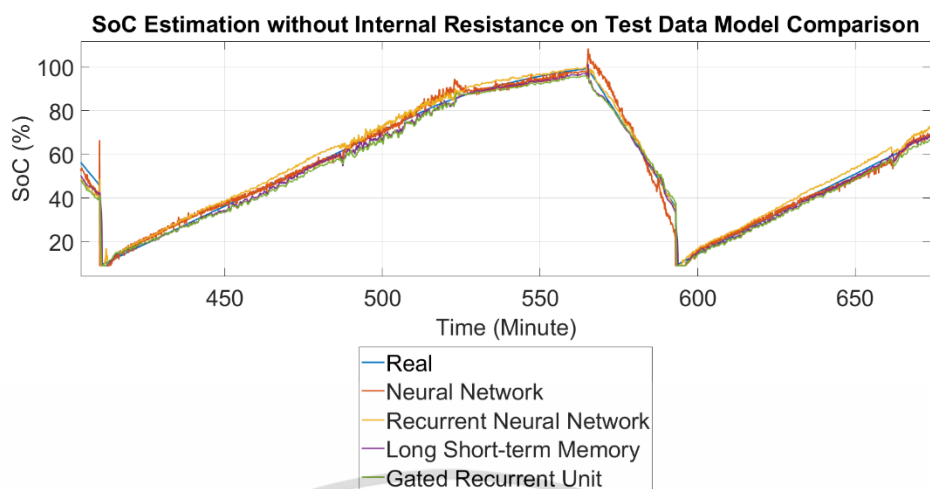


รูปที่ 4.41 เปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อนำมาทำนายระดับประจุแบตเตอรี่รับชุดข้อมูลทดสอบเมื่อไม่พิจารณาค่าความต้านทานภายใน

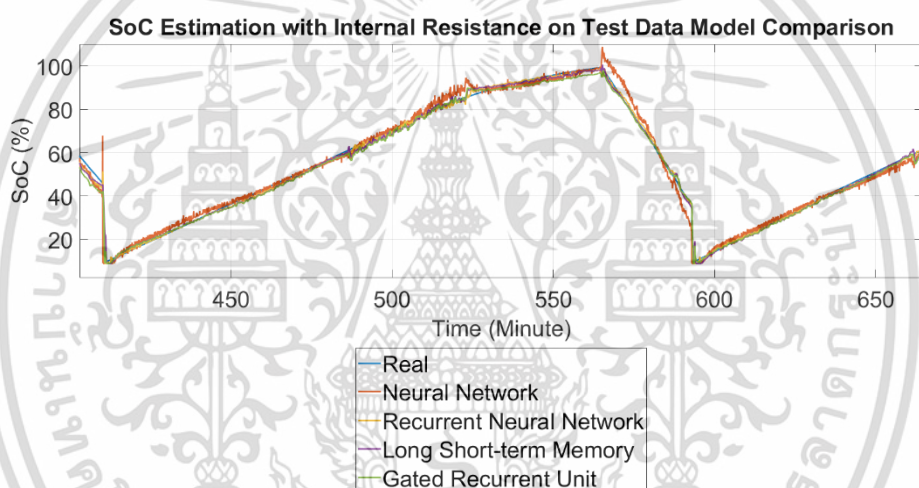


รูปที่ 4.42 เปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อนำมาทำนายระดับประจุแบตเตอรี่รับชุดข้อมูลทดสอบเมื่อพิจารณาค่าความต้านทานภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.43 ภาพขยายเปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อนำมาทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลทดสอบเมื่อไม่พิจารณาค่าความต้านทานภายใน



รูปที่ 4.44 ภาพขยายเปรียบเทียบโมเดลปัญญาประดิษฐ์ต่างชนิดเมื่อนำมาทำนายระดับประจุแบตเตอรี่บนชุดข้อมูลทดสอบเมื่อพิจารณาค่าความต้านทานภายใน

4.3.2. เปรียบเทียบระยะเวลาการสอนและการทำนายระดับประจุแบตเตอรี่

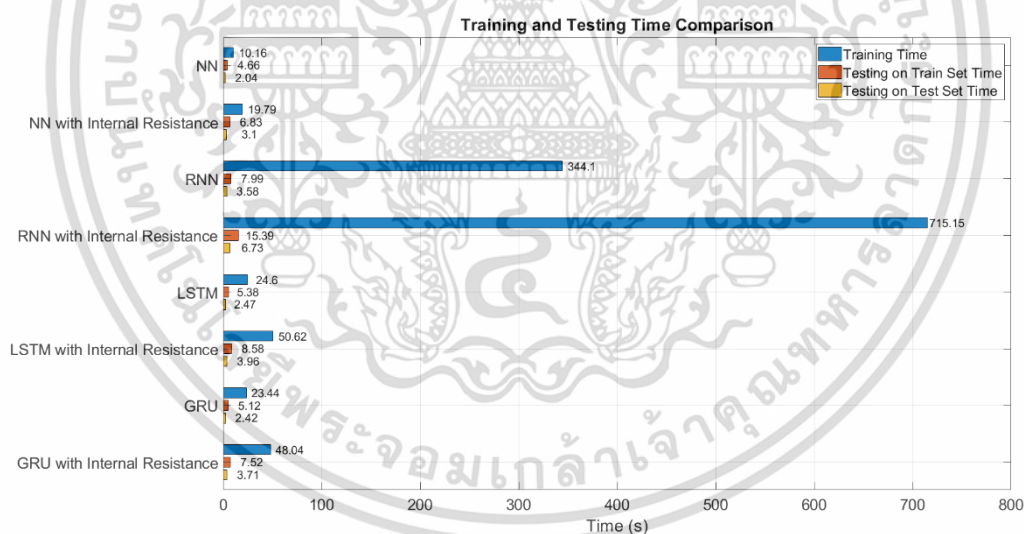
ระยะเวลาการสอนและการทำนายระดับประจุแบตเตอรี่ในงานวิจัยนี้เป็นไปตามตารางที่ 4.22 ซึ่งสามารถแสดงโดยรูปดังรูปที่ 4.45 พบว่าโมเดลปัญญาประดิษฐ์ในงานวิจัยที่พิจารณาค่าความต้านทานภายในร่วมด้วยจะใช้เวลานานกว่าโมเดลปัญญาประดิษฐ์ที่ไม่พิจารณาค่าความต้านทานภายในอยู่ประมาณ 2 เท่าในทุกกรณีเนื่องมาจากการคำนวณค่าความต้านทานภายในต้องทำการทำนายแรงดันเปิดวงจรก่อน โดยรวมโมเดลปัญญาประดิษฐ์ชนิด Neural Network ใช้เวลาน้อยสุดในทุกกรณี โมเดลปัญญาประดิษฐ์ชนิด Gated Recurrent Unit ใช้เวลาน้อยสุดรองลงมา และมีเวลาที่ใกล้เคียงกับโมเดลปัญญาประดิษฐ์ชนิด Long Short-term Memory แต่โมเดลปัญญาประดิษฐ์ชนิด Recurrent Neural Network จะใช้เวลานานต่างกันอย่างมีนัยสำคัญเนื่องมาจากโมเดล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัญญาประดิษฐ์ชนิด Recurrent Neural Network ต้องปรับค่าพารามิเตอร์ให้มีความซับซ้อนสูง เพื่อให้ได้ค่าความผิดพลาดที่ต่ำ

ตารางที่ 4.22 ระยะเวลาระหว่างโมเดลปัญญาประดิษฐ์ต่างชนิดในการสอนและทดสอบบนชุดข้อมูลสอนและทดสอบ

	Training Time (s)	Testing on Train Set Time (s)	Testing on Test Set Time (s)
NN	10.16	4.66	2.04
NN with Internal Resistance	19.79	6.83	3.10
RNN	344.10	7.99	3.58
RNN with Internal Resistance	715.15	15.39	6.73
LSTM	24.60	5.38	2.47
LSTM with Internal Resistance	50.62	8.58	3.96
GRU	23.44	5.12	2.42
GRU with Internal Resistance	48.04	7.52	3.71



รูปที่ 4.45 ระยะเวลาระหว่างโมเดลปัญญาประดิษฐ์ต่างชนิดในการสอนและทดสอบบนชุดข้อมูลสอนและทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1. สรุปผลการวิจัย

งานวิจัยนี้ได้ทำการออกแบบโมเดลปัญญาประดิษฐ์ที่สามารถทำนายค่าระดับประจุแบตเตอรี่ โดยใช้ข้อมูลจากระบบจัดการแบตเตอรี่ภายใต้การใช้งานที่เป็นการคายประจุผ่านภาระทางไฟฟ้าชนิด กระแสคงที่ และชนิดกำลังไฟฟ้าคงที่ การอัดประจุชนิดกระแสคงที่แรงดันคงที่ (Constant Current/Constant Voltage: CC/CV) ที่อุณหภูมิห้อง ศึกษาถึงผลของค่าความต้านทานภายใน แบตเตอรี่ในการทำนายระดับประจุแบตเตอรี่ในด้านความผิดพลาดและระยะเวลาในการสอนและทำนาย โดยทำการเปรียบเทียบโมเดลปัญญาประดิษฐ์ที่ออกแบบระหว่างโมเดลที่มีการพิจารณาค่าความต้านทานภายในแบตเตอรี่กับโมเดลปัญญาประดิษฐ์ที่ไม่มีการพิจารณาค่าความต้านทานภายใน และเปรียบเทียบโมเดลปัญญาประดิษฐ์ที่ชนิดแตกต่างกัน

โมเดลปัญญาประดิษฐ์ชนิดที่มีการป้อนกลับทั้ง 3 ชนิด ที่นำเสนอในงานวิจัยนี้ได้แก่ 1) Recurrent Neural Network (RNN) 2) Long Short-term Memory (LSTM) 3) Gated Recurrent Unit (GRU) มีความสามารถในการทำนายค่าระดับประจุแบตเตอรี่โดยใช้ข้อมูลจากระบบจัดการ แบตเตอรี่ที่สภาวะอัดประจุชนิดกระแสคงที่แรงดันคงที่และคายประจุชนิดกระแสคงที่และชนิด กำลังไฟฟ้าคงที่ที่อุณหภูมิห้องได้ดีกว่าโมเดลปัญญาประดิษฐ์ชนิดที่ไม่มีการป้อนกลับที่นำเสนอใน งานวิจัยนี้ได้แก่ 1) Neural Network (NN)

เมื่อมีการพิจารณาค่าความต้านทานภายในร่วมในการทำนายระดับประจุแบตเตอรี่จะทำให้ โมเดลปัญญาประดิษฐ์ทั้ง 4 ชนิด ที่นำเสนอในงานวิจัยนี้มีค่าความผิดพลาดที่ลดลง ในโมเดล ปัญญาประดิษฐ์ที่มีการป้อนกลับทั้ง 3 ชนิด มีค่าความผิดพลาดที่ใกล้เคียงกันและน้อยกว่าโมเดล ปัญญาประดิษฐ์ที่ไม่มีการป้อนกลับอย่างมีนัยสำคัญ โมเดลปัญญาประดิษฐ์ชนิด Long Short-term Memory ที่พิจารณาค่าความต้านทานภายในให้ค่าความผิดพลาดต่ำสุดเมื่อคำนวณแบบ Mean absolute error (MAE) ที่ 1.7345% บนชุดข้อมูลทดสอบ และโมเดลปัญญาประดิษฐ์ชนิด Recurrent Neural Network ที่พิจารณาค่าความต้านทานภายในให้ค่าความผิดพลาดต่ำสุดเมื่อ คำนวณแบบ Root mean squared error (RMSE) ที่ 2.4589% บนชุดข้อมูลทดสอบ

ในด้านระยะเวลาในการสอนและทำนายพบว่าโมเดลปัญญาประดิษฐ์ทั้ง 4 ชนิด ที่ใช้ใน งานวิจัยนี้เมื่อมีการพิจารณาค่าความต้านทานภายในจะทำให้ใช้ระยะเวลาที่นานขึ้นประมาณ 2 เท่า เนื่องมาจากการคำนวณค่าความต้านทานภายในจำเป็นต้องทำการทำนายค่าแรงดันเปิดวงจรก่อน โดยโมเดลปัญญาประดิษฐ์ชนิด Neural Network ใช้ระยะเวลาน้อยที่สุดในการสอนโมเดล ปัญญาประดิษฐ์ที่มีข้อมูล 38,714 ข้อมูลที่ 10.16 วินาที การทำนายบนชุดข้อมูลสอนที่มีข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

38,714 ข้อมูลที่ 4.66 วินาที และการทำนายบนชุดข้อมูลทดสอบที่มีข้อมูล 16,593 ข้อมูลที่ 2.04 วินาที ซึ่งน้อยกว่าโมเดลปัญญาประดิษฐ์ชนิด Long Short-term Memory และ Gated Recurrent Unit เล็กน้อย แต่น้อยกว่าโมเดลปัญญาประดิษฐ์ชนิด Recurrent Neural Network อย่างมีนัยสำคัญ

สุดท้ายเมื่อพิจารณาค่าความผิดพลาดในการทำนายระดับประจุแบตเตอรี่และระยะเวลาในการสอนและทำนายโมเดลปัญญาประดิษฐ์ทั้ง 4 ชนิดที่นำเสนอในงานวิจัยนี้ โมเดลปัญญาประดิษฐ์ชนิด Long Short-term Memory ที่พิจารณาค่าความต้านทานภายในแบตเตอรี่มีความเหมาะสมที่สุดในการทำนายระดับประจุแบตเตอรี่ภายใต้ชุดข้อมูลจากระบบจัดการแบตเตอรี่ที่เป็นการคายประจุชนิดกระแสคงที่และกำลังไฟฟ้าคงที่ การอัดประจุชนิดกระแสคงที่แรงดันคงที่ ที่อุณหภูมิห้อง

ตัวอย่างที่แนะนำในการใช้งานโมเดลปัญญาประดิษฐ์ในงานวิจัยนี้ที่เหมาะสมกับผลความแม่นยำและระยะเวลาที่ใช้ในการสอนและทำนายโดยโมเดลปัญญาประดิษฐ์ในงานวิจัยนี้เป็นไปตามตารางที่ 5.1

ตารางที่ 5.1 ตัวอย่างที่แนะนำในการใช้งานโมเดลปัญญาประดิษฐ์ในงานวิจัยนี้

	ความแม่นยำ	ระยะเวลา	การใช้งาน
NN	ต่ำ	ดีมาก	อุปกรณ์ที่สามารถอัดประจุใหม่ได้ง่าย เช่น นาฬิกาอัจฉริยะ หรือ ลำโพง
NN with Internal Resistance	ต่ำ	ดีมาก	อุปกรณ์ที่สามารถอัดประจุใหม่ได้ง่าย เช่น นาฬิกาอัจฉริยะ หรือ ลำโพง
RNN	ต่ำ	ต่ำ	-
RNN with Internal Resistance	ดีมาก	ต่ำ	อุปกรณ์ที่ส่งข้อมูลวันละครั้ง เช่น อุปกรณ์ IoT กำลังต่ำ
LSTM	ปานกลาง	ปานกลาง	ยานยนต์ไฟฟ้าระยะทางต่ำ
LSTM with Internal Resistance	ดีมาก	ปานกลาง	ยานยนต์ไฟฟ้าหรือระบบกักเก็บพลังงาน
GRU	ปานกลาง	ปานกลาง	ยานยนต์ไฟฟ้าระยะทางต่ำ
GRU with Internal Resistance	ดีมาก	ปานกลาง	ยานยนต์ไฟฟ้าหรือระบบกักเก็บพลังงาน

โมเดลปัญญาประดิษฐ์ชนิด Neural Network ที่พิจารณาและไม่พิจารณาค่าความต้านทานภายในมีความเหมาะสมกับการใช้งานอุปกรณ์ที่สามารถอัดประจุใหม่ได้ง่ายเพราะต้องการทราบถึงเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยามให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระดับประจุแบตเตอรี่ที่หมดหรือยังไม่หมดเท่านั้น โมเดลปัญญาประดิษฐ์ชนิด Recurrent Neural Network ที่ไม่พิจารณาค่าความต้านทานภายในไม่เหมาะกับการใช้งานใด โมเดลปัญญาประดิษฐ์ชนิด Recurrent Neural Network ที่พิจารณาค่าความต้านทานภายในเหมาะกับการใช้ในอุปกรณ์ที่ทำการส่งข้อมูลวันละครั้งหรือนานๆ ครั้ง โมเดลปัญญาประดิษฐ์ชนิด Long Short-term Memory และ Gated Recurrent Unit ที่ไม่พิจารณาค่าความต้านทานภายในเหมาะกับการใช้งานยานยนต์ไฟฟ้าที่มีระยะทางต่ำเนื่องจากไม่จำเป็นต้องวางแผนการใช้งานที่แม่นยำมากนัก และโมเดลปัญญาประดิษฐ์ชนิด Long Short-term Memory และ Gated Recurrent Unit ที่พิจารณาค่าความต้านทานภายในเหมาะกับการใช้งานยานยนต์ไฟฟ้าและระบบกักเก็บพลังงานเนื่องจากจำเป็นต้องมีการวางแผนการใช้งานซึ่งอาศัยความแม่นยำ

5.2. ข้อเสนอแนะ

งานวิจัยนี้ยังเป็นงานวิจัยที่อยู่บนขอบเขตของแบตเตอรี่ลิเทียมนิกเกิลโคบอลต์อะลูมิเนียมออกไซด์ Lithium Nickel Cobalt Aluminium Oxide หรือ NCA ขนาด 3,350 mAh 7 เซลล์ต่ออนุกรมกัน ดังนั้นเมื่อมีการใช้งานกับแบตเตอรี่ชนิดอื่นหรือขนาดอื่นหรือการใช้งานในรูปแบบอื่นก็อาจจะใช้งานได้ โดยควรสังเกตถึงค่าความผิดพลาดที่เกิดขึ้นและทำการปรับแต่งโมเดลปัญญาประดิษฐ์ให้มีความเหมาะสม

งานวิจัยนี้มีประเด็นที่น่าศึกษาต่อในการพัฒนาต่อไป เช่น การผสมโมเดลปัญญาประดิษฐ์ต่างชนิดเข้าด้วยกัน รวมไปถึงการศึกษาถึงการเพิ่มค่าพารามิเตอร์อื่นของโมเดลปัญญาประดิษฐ์ เช่น เพิ่มจำนวนชั้นซ่อน (Hidden Layer) ร่วมในการสอนและทำนายระดับประจุแบตเตอรี่ เพื่อศึกษาถึงค่าความผิดพลาดในการทำนายระดับประจุแบตเตอรี่ที่อาจจะต่ำลงได้ หรือ ทำการสอนโมเดลปัญญาประดิษฐ์บนชุดข้อมูลแบตเตอรี่ชนิดและอุณหภูมิอื่นๆ เพิ่มเติม เพื่อให้โมเดลปัญญาประดิษฐ์มีความสามารถในการทำนายระดับประจุแบตเตอรี่ที่หลากหลายมากขึ้น หรือ ดำเนินการนำโมเดลปัญญาประดิษฐ์ที่ได้ทำการออกแบบในงานวิจัยนี้ฝังลงในระบบจัดการแบตเตอรี่ เพื่อให้สามารถทำนายระดับประจุแบตเตอรี่ระหว่างใช้งานได้ ซึ่งอาจพัฒนาความสามารถของระบบจัดการแบตเตอรี่ในการทำนายระดับประจุแบตเตอรี่ และอาจนำไปสู่ความก้าวหน้าของเทคโนโลยีระบบจัดการแบตเตอรี่อื่นๆ ในอนาคตได้

เอกสารอ้างอิง

- [1] กระทรวงพลังงาน. 2566 รายงานสถิติพลังงานรายปี 2566
- [2] U.S. Department of Energy's Energy Storage Market Report 2020
- [3] C. Vidal, P. Malysz, P. Kollmeyer and A. Emadi, "Machine Learning Applied to Electrified Vehicle Battery State of Charge and State of Health Estimation: State-of-the-Art," in IEEE Access, vol. 8, pp. 52796-52814, 2020
- [4] M. O. Qays, Y. Buswig, M. L. Hossain and A. Abu-Siada, "Recent progress and future trends on the state of charge estimation methods to improve battery-storage efficiency: A review," CSEE Journal of Power and Energy Systems, vol. 8, no. 1, pp. 105-114, Jan. 2022
- [5] D. N. T. How, M. A. Hannan, M. S. Hossain Lipu and P. J. Ker, "State of Charge Estimation for Lithium-Ion Batteries Using Model-Based and Data-Driven Methods: A Review," in IEEE Access, vol. 7, pp. 136116-136136, 2019
- [6] Y. Li, H. Guo, F. Qi, Z. Guo and M. Li, "Comparative Study of the Influence of Open Circuit Voltage Tests on State of Charge Online Estimation for Lithium-Ion Batteries," in IEEE Access, vol. 8, pp. 17535-17547, 2020
- [7] J. Lee and J. Won, "Enhanced Coulomb Counting Method for SoC and SoH Estimation Based on Coulombic Efficiency," in IEEE Access, vol. 11, pp. 15449-15459, 2023
- [8] F. Naseri, E. Schaltz, D. -I. Stroe, A. Gissero and E. Farjah, "An Enhanced Equivalent Circuit Model With Real-Time Parameter Identification for Battery State-of-Charge Estimation," in IEEE Transactions on Industrial Electronics, vol. 69, no. 4, pp. 3743-3751, April 2022
- [9] Y. Liu et al., "A Nonlinear Observer SOC Estimation Method Based on Electrochemical Model for Lithium-Ion Battery," in IEEE Transactions on Industry Applications, vol. 57, no. 1, pp. 1094-1104, Jan.-Feb. 2021
- [10] M. A. Hannan, M. S. H. Lipu, A. Hussain, M. H. Saad and A. Ayob, "Neural Network Approach for Estimating State of Charge of Lithium-Ion Battery Using Backtracking Search Algorithm," in IEEE Access, vol. 6, pp. 10069-10079, 2018

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [11] H. Chaoui and C. C. Ibe-Ekeocha, "State of Charge and State of Health Estimation for Lithium Batteries Using Recurrent Neural Networks," in IEEE Transactions on Vehicular Technology, vol. 66, no. 10, pp. 8773-8783, Oct. 2017
- [12] F. Yang, X. Song, F. Xu and K. -L. Tsui, "State-of-Charge Estimation of Lithium-Ion Batteries via Long Short-Term Memory Network," in IEEE Access, vol. 7, pp. 53792-53799, 2019
- [13] M. A. Hannan, D. N. T. How, M. B. Mansor, M. S. Hossain Lipu, P. J. Ker and K. M. Muttaqi, "State-of-Charge Estimation of Li-ion Battery Using Gated Recurrent Unit With One-Cycle Learning Rate Policy," in IEEE Transactions on Industry Applications, vol. 57, no. 3, pp. 2964-2971, May-June 2021
- [14] Z. Zhang et al., "An Improved Bidirectional Gated Recurrent Unit Method for Accurate State-of-Charge Estimation," in IEEE Access, vol. 9, pp. 11252-11263, 2021
- [15] X. Song, F. Yang, D. Wang and K. -L. Tsui, "Combined CNN-LSTM Network for State-of-Charge Estimation of Lithium-Ion Batteries," in IEEE Access, vol. 7, pp. 88894-88902, 2019
- [16] Z. Huang, F. Yang, F. Xu, X. Song and K. -L. Tsui, "Convolutional Gated Recurrent Unit–Recurrent Neural Network for State-of-Charge Estimation of Lithium-Ion Batteries," in IEEE Access, vol. 7, pp. 93139-93149, 2019
- [17] Q. Sun, H. Lv, S. Wang, S. Gao and K. Wei, "Optimized State of Charge Estimation of Lithium-Ion Battery in SMES/Battery Hybrid Energy Storage System for Electric Vehicles," in IEEE Transactions on Applied Superconductivity, vol. 31, no. 8, pp. 1-6, Nov. 2021
- [18] M. A. Hannan, M. M. Hoque, A. Hussain, Y. Yusof and P. J. Ker, "State-of-the-Art and Energy Management System of Lithium-Ion Batteries in Electric Vehicle Applications: Issues and Recommendations," in IEEE Access, vol. 6, pp. 19362-19378, 2018
- [19] M. Naguib, P. Kollmeyer and A. Emadi, "Lithium-Ion Battery Pack Robust State of Charge Estimation, Cell Inconsistency, and Balancing: Review," in IEEE Access, vol. 9, pp. 50570-50582, 2021

- [20] Jiabaida. JDB RS485-RS232-UART-Bluetooth-Communication Protocol. [Online] Available:https://cdn.shopifycdn.net/s/files/1/0606/5199/5298/files/JDB_RS485-RS232-UART-Bluetooth-Communication_Protocol.pdf?v=1682577935
- [21] Scikit-learn. Sklearn.preprocessing.MinMaxScaler[Online]Available: <https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html>
- [22] Scikit-learn. Cross-validation: evaluating estimator performance. [Online] Available: https://scikit-learn.org/stable/modules/cross_validation.html
- [23] M. Camas-Náfate, A. Coronado-Mendoza, C. J. Vega-Gómez, and F. Espinosa-Moreno, “Modeling and Simulation of a Commercial Lithium-Ion Battery with Charge Cycle Predictions,” *Sustainability*, vol. 14, no. 21, p. 14035, Oct. 2022



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำนายระดับประจุแบตเตอรี่ของกลุ่มแบตเตอรี่ลิเทียมไอออนด้วยวิธีปัญญาประดิษฐ์โดยพิจารณาค่าความต้านทานภายในแบตเตอรี่

SoC Estimation of Lithium-ion Battery Pack using Artificial Intelligence considering Battery Internal Resistance

พศุตม์ กรุดสอน¹ เปี่ยมภูมิ สฤตฤกษ์¹ และ พิชิต ล้ายอง¹

¹ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 64601232@kmitl.ac.th

บทคัดย่อ

บทความนี้นำเสนอการทำนายระดับประจุแบตเตอรี่ จากกลุ่มของแบตเตอรี่ลิเทียมไอออน ทำการทดสอบการอัดประจุโดยวิธีกระแสคงที่ แรงดันคงที่ และทดสอบการคายประจุที่ภาระทางไฟฟ้าชนิดกระแสคงที่และกำลังไฟฟ้าคงที่ ทำการเปรียบเทียบประสิทธิภาพระหว่างโมเดลปัญญาประดิษฐ์ 3 วิธี ได้แก่ วิธีต้นไม้ตัดสินใจ (Decision Tree), วิธีโครงข่ายประสาทเทียม (Neural Network) และวิธีโครงข่ายความจำระยะสั้นและยาว (LSTM) โดยใช้ข้อมูลที่ได้จากระบบจัดการแบตเตอรี่ (BMS) ประกอบด้วย กระแส แรงดันของกลุ่มแบตเตอรี่ แรงดันของแต่ละเซลล์ อุณหภูมิของระบบจัดการแบตเตอรี่ อุณหภูมิแบตเตอรี่ อุณหภูมิสภาพแวดล้อม และความต้านทานภายในของแต่ละเซลล์ สำหรับการสร้างโมเดลปัญญาประดิษฐ์ ผลการทดลองพบว่าโมเดลชนิด LSTM ร่วมกับค่าความต้านทานภายในให้ค่าความผิดพลาดที่น้อยกว่า 5%

คำสำคัญ: การทำนายระดับประจุแบตเตอรี่ แบตเตอรี่ลิเทียมไอออน ระบบจัดการแบตเตอรี่ ความต้านทานภายใน โมเดลโครงข่ายความจำระยะสั้นและยาว

Abstract

This paper presents a State of Charge (SoC) Estimation of a lithium-ion battery pack composed of charging test with constant current and constant voltage method as well as discharging test with constant current and constant power loads. Three artificial intelligence models are applied to compare the estimation performance including Decision Tree, Neural Network, and Long short-term memory (LSTM) models. The predictive data obtained from a battery management system (BMS) include current, pack voltage, cell voltage, BMS temperature, battery temperature, ambient temperature, and battery internal resistance. As the results LSTM considering internal resistance effectively yields the lowest error as less than 5%

Keywords: SoC Estimation, Lithium-ion Battery, Battery Management System, Internal Resistance, Long short-term memory

1. บทนำ

ในปัจจุบันปริมาณการผลิตไฟฟ้าโดยพลังงานหมุนเวียนมีแนวโน้มที่เพิ่มขึ้นเนื่องมาจากการผลักดันนโยบายด้านพลังงานสะอาดเพื่อลดการปล่อยก๊าซเรือนกระจกและการปล่อยก๊าซคาร์บอนไดออกไซด์เป็นศูนย์ พลังงานหมุนเวียนในระบบที่มีแบตเตอรี่ร่วมด้วย การควบคุมเพื่อให้มีประสิทธิภาพและเสถียรภาพจำเป็นต้องมีการทำนายค่าระดับประจุแบตเตอรี่ (SoC) ให้มีความเที่ยงตรงแม่นยำ

มีงานวิจัยจำนวนมากทำการศึกษาเกี่ยวกับการทำนายระดับประจุแบตเตอรี่แต่ละการทำการศึกษาเกี่ยวกับแบตเตอรี่จำนวนหนึ่งเซลล์และขาดการเปรียบเทียบระหว่างโมเดลปัญญาประดิษฐ์ชนิดต่างๆ [1] การสร้างและทำนายจะใช้ข้อมูลที่เป็นภาระทางไฟฟ้าแบบคงที่ชนิดเดียว [2] ปรากฏการทำนายโดยมีตัวแปรความต้านทานภายในแบตเตอรี่ [3]

งานวิจัยชิ้นนี้สร้างโมเดลปัญญาประดิษฐ์สำหรับการทำนายระดับประจุแบตเตอรี่ในกลุ่มแบตเตอรี่ โดยใช้ชุดข้อมูลที่เก็บมาจากระบบจัดการแบตเตอรี่ (Battery Management System) โดยเป็นการอัดประจุและคายประจุที่สภาวะแตกต่างกัน ทำการเปรียบเทียบประสิทธิภาพระหว่างโมเดลปัญญาประดิษฐ์ชนิดต้นไม้ตัดสินใจ (Decision Tree) โมเดลโครงข่ายประสาทเทียม (Neural Network) และ โมเดลหน่วยความจำระยะสั้นยาว (Long Short-Term Memory) ในกรณีที่มีและไม่มีค่าความต้านทานภายในแบตเตอรี่ (Internal resistance) ด้วยการหาค่าความผิดพลาด (Error) ด้วยค่าสัมบูรณ์เฉลี่ยของค่าความผิดพลาด (MAE) และค่ารากที่สองของค่าเฉลี่ยกำลังสองของค่าความผิดพลาด (RMSE)

2. ทฤษฎีที่เกี่ยวข้อง

2.1 ระบบจัดการแบตเตอรี่

ระบบจัดการแบตเตอรี่ หรือ BMS คืออุปกรณ์อิเล็กทรอนิกส์ที่หน้าที่ควบคุมการอัดคายประจุ ป้องกันไม่ให้อายุสั้น กระแส อุณหภูมิ มากหรือน้อยจากค่าที่ตั้งเอาไว้ จำนวน และรายงานข้อมูลของแบตเตอรี่

2.2 โมเดลต้นไม้ตัดสินใจ

โมเดลต้นไม้ตัดสินใจคือการสร้างอัลกอริทึมคณิตศาสตร์โดยจะทำการแบ่งข้อมูลโดยหาจุดที่มีค่าความผิดพลาดต่ำสุดในการแบ่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 โมเดลโครงข่ายประสาทเทียม

โมเดลโครงข่ายประสาทเทียมแบ่งออกเป็น 3 ชั้นได้แก่ ชั้นรับข้อมูล (Input Layer) จะทำการรับข้อมูลมาตามตัวแปรที่ได้ใส่เข้าไป ชั้นซ่อน (Hidden Layer) เป็นการคำนวณทางคณิตศาสตร์โดยกำหนดค่าน้ำหนักให้ตัวแปรต่างๆจะมีที่ชั้นก็ได้ และชั้นส่งข้อมูล (Output Layer) เป็นชั้นส่งออกข้อมูลมีจำนวนเท่ากับตัวแปรขาออกที่ต้องการ

2.4 โมเดลหน่วยความจำระยะสั้นยาว

โมเดลหน่วยความจำระยะสั้นยาวเป็นโมเดลที่เหมาะสมกับการทำนายข้อมูลที่เป็นระยะเวลา โดยจะเป็นการนำข้อมูลในอดีตมาผ่านประตู (gate) ที่แตกต่างกันเพื่อนำเข้าหรือนำออกข้อมูลต่างๆ เพื่อทำนายข้อมูลถัดไป

2.5 ความต้านทานภายในแบตเตอรี่

ความต้านทานภายในแบตเตอรี่ (Internal Resistance) เป็นตัวแปรที่สำคัญซึ่งบ่งบอกถึงประสิทธิภาพของแบตเตอรี่ในการรับและจ่ายประจุ สามารถหาค่าความต้านทานภายในแบตเตอรี่ดังสมการที่ 1 [4]

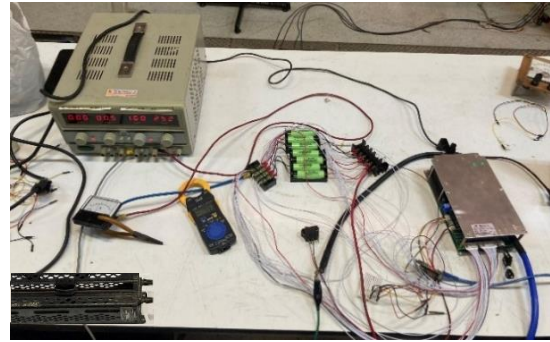
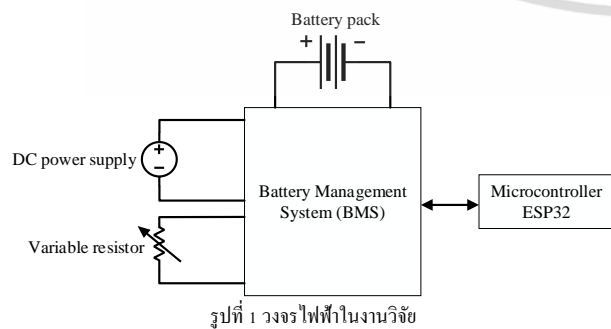
$$R_i = \frac{V_{ocv} - V_L}{i_L} \quad (1)$$

โดยที่ R_i คือ ความต้านทานภายในแบตเตอรี่
 V_{ocv} คือ แรงดันแบตเตอรี่ขณะเปิดวงจร
 V_L คือ แรงดันขณะจ่ายโหลด
 i_L คือ กระแสแบตเตอรี่

3. การออกแบบการทดลอง

3.1 วงจรไฟฟ้า

งานวิจัยนี้ได้ทำการทดลองกับแบตเตอรี่ชนิดลิเทียมไอออน Panasonic NCR18650B 3.6V 3,350mAh จำนวน 7 เซลล์ต่ออนุกรมกันกับ BMS รุ่น JBD-DP24S002 เชื่อมต่อกับไมโครคอนโทรลเลอร์ ESP 32 โดยการสื่อสารข้อมูลผ่านพอร์ตอนุกรมแบบอะซิงโครนัส (UART) เพื่อทำการรับส่งและแปลงข้อมูล ทำการอัดประจุผ่านแหล่งจ่ายไฟกระแสตรง (DC Power Supply) และทำการคายประจุผ่านตัวต้านทานปรับค่าได้ (Variable Resistor) ดังรูปที่ 1



รูปที่ 2 วงจรไฟฟ้าที่ทดลองจริง

จากรูปที่ 1 แบตเตอรี่ทำการอัดประจุแบบกระแสคงที่แรงดันคงที่ CC/CV และทำการคายประจุแบบกระแสคงที่และกำลังคงที่ที่วงจรที่ได้ทำการทดลองจริงเป็นไปตามรูปที่ 2

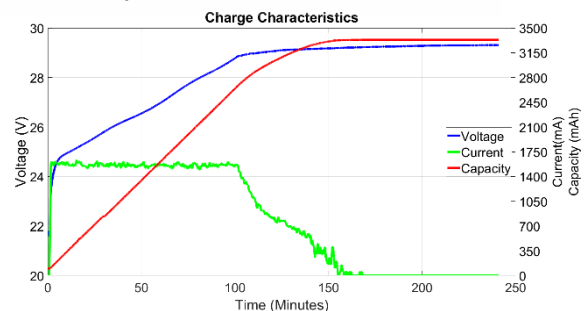
3.2 การจัดการข้อมูล

ข้อมูลที่ได้รับจาก BMS นั้นเป็นข้อมูลฐาน 16 ต้องแปลงค่ากลับมาให้เป็นฐาน 10 โดยข้อมูลจาก BMS ได้แก่ กระแส แรงดันของกลุ่มแบตเตอรี่ แรงดันของแต่ละเซลล์ อุณหภูมิของระบบจัดการแบตเตอรี่ อุณหภูมิแบตเตอรี่ อุณหภูมิสภาพแวดล้อม ส่วนค่าความต้านทานภายในแบตเตอรี่ได้มาจากการคำนวณตามสมการที่ 1 โดยแรงดันแบตเตอรี่ขณะเปิดวงจรได้มาจากการประมาณค่า [5] เมื่อได้ข้อมูลครบแล้วจึงทำให้ข้อมูลอยู่ในบรรทัดฐานเดียวกัน (Normalization) ดังสมการที่ 2

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (2)$$

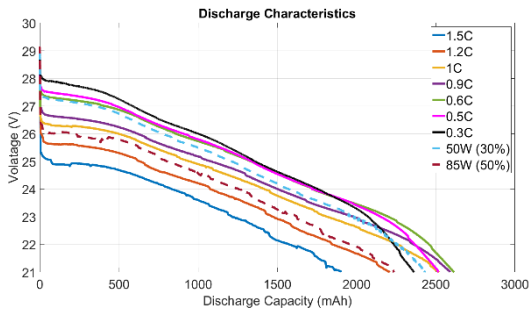
โดยที่ x คือ ค่าก่อนทำให้อยู่ในบรรทัดฐานใหม่
 x_{norm} คือ ค่าที่อยู่ในบรรทัดฐานใหม่
 x_{min} คือ ค่าต่ำสุดของข้อมูล
 x_{max} คือ ค่าสูงสุดของข้อมูล

แบตเตอรี่ทำการอัดประจุโดยแหล่งจ่ายไฟกระแสตรงแบบกระแสคงที่แรงดันคงที่ หรือ CC/CV ที่กระแส 0.5C (1.65A) แรงดัน 29.4V มีความสัมพันธ์ระหว่างแรงดัน กระแส และประจุแบตเตอรี่กับเวลาดังรูปที่ 3 และทำการคายประจุโดยตัวต้านทานปรับค่าได้ แบบกระแสคงที่ขนาด 0.3C, 0.5C, 0.6C, 0.9C, 1C, 1.2C และ 1.5C แบบกำลังคงที่ขนาด 30% (50W) และ 50% (85W) มีความสัมพันธ์ระหว่างแรงดัน และจำนวนคายประจุแบตเตอรี่ดังรูปที่ 4



รูปที่ 3 กราฟคุณลักษณะการอัดประจุของข้อมูลที่ใช้ในงานวิจัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4 กราฟคุณลักษณะการคายประจุของข้อมูลที่ใช้ในงานวิจัย

3.3 โมเดลปัญญาประดิษฐ์

งานวิจัยนี้ได้ทำการสร้างโมเดลปัญญาประดิษฐ์ 3 ชนิดได้แก่ โมเดลต้นไม้ตัดสินใจ โมเดลโครงข่ายประสาทเทียม และโมเดลหน่วยความจำระยะสั้นยาว โดยใช้ข้อมูล กระแส แรงดันของกลุ่มแบตเตอรี่ แรงดันของแต่ละเซลล์ อุณหภูมิของระบบจัดการแบตเตอรี่ อุณหภูมิแบตเตอรี่ อุณหภูมิสภาพแวดล้อม ค่าความต้านทานภายใน ที่ทำให้ข้อมูลอยู่ในบรรทัดฐานเดียวกันแล้วมาสร้างโดยแบ่งข้อมูล 70% สำหรับการสอนโมเดล 30% สำหรับการทดสอบโมเดล และทำการทดสอบประสิทธิภาพของแต่ละโมเดลโดยชุดข้อมูลอื่นๆที่โมเดลไม่เคยเห็นในการทดสอบหาค่าความผิดพลาดโดยค่าสัมบูรณ์เฉลี่ยของค่าความผิดพลาด และ ค่ารากที่สองของค่าเฉลี่ยกำลังสองของค่าความผิดพลาด การเขียนโค้ดทั้งหมดที่กล่าวมาข้างต้นทำบน jupyter notebook ค่าตัวแปรสำหรับโมเดลต้นไม้ตัดสินใจ โมเดลโครงข่ายประสาทเทียม และ โมเดลหน่วยความจำระยะสั้นยาว เป็นไปตามตารางที่ 1 ตารางที่ 2 และตารางที่ 3 ตามลำดับ

ตารางที่ 1 ค่าตัวแปรโมเดลต้นไม้ตัดสินใจ

ตัวแปร	ขนาด
Max depth	10

ตารางที่ 2 ตัวแปร โมเดลโครงข่ายประสาทเทียม

ตัวแปร	ขนาด
Neurons	150
Batch	32
Optimizer	Adam
Active function	Rectified Linear Unit (ReLU)

ตารางที่ 3 ตัวแปรโมเดลหน่วยความจำระยะสั้นยาว

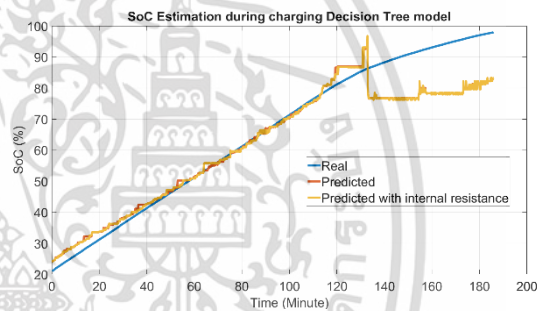
ตัวแปร	ขนาด
Neurons	150
Epoch	30
Batch	32
Optimizer	Adam
Active function	Rectified Linear Unit (ReLU)
Dropout	0.2
Input Data window	20

4. ผลการทดลอง

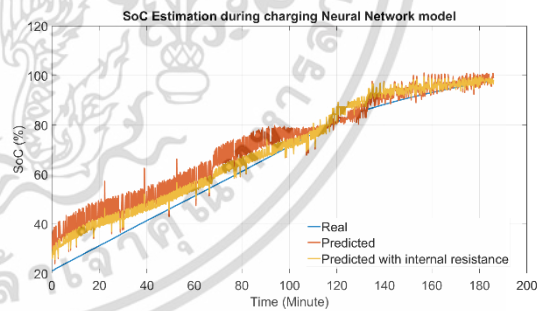
โมเดลปัญญาประดิษฐ์ที่ถูกสร้างจะนำไปทำนายระดับประจุแบตเตอรี่ในการอัดประจุและคายประจุ เปรียบเทียบประสิทธิภาพระหว่างโมเดลต้นไม้ตัดสินใจ โมเดลโครงข่ายประสาทเทียม และโมเดลหน่วยความจำระยะสั้นยาว ทั้งแบบคำนึงถึงความต้านทานภายในแบตเตอรี่และไม่คำนึงถึงค่าความต้านทานภายในแบตเตอรี่โดยการหาค่าความผิดพลาดด้วยวิธีค่าสัมบูรณ์เฉลี่ยของค่าความผิดพลาด และ ค่ารากที่สองของค่าเฉลี่ยกำลังสองของค่าความผิดพลาด มีผลการทดลองดังนี้

4.1 การทำนายระดับประจุแบตเตอรี่ระหว่างอัดประจุ

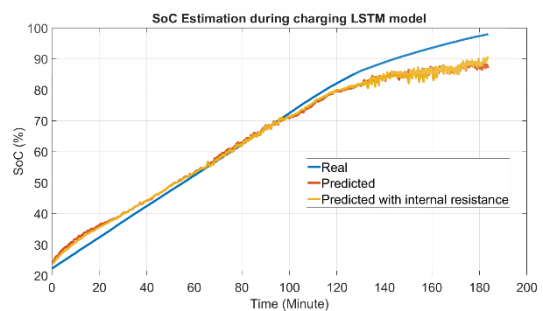
การทำนายระดับประจุแบตเตอรี่ระหว่างอัดประจุสำหรับโมเดลต้นไม้ตัดสินใจได้ผลดังรูปที่ 5 โมเดลโครงข่ายประสาทเทียมได้ผลดังรูปที่ 6 และโมเดลหน่วยความจำระยะสั้นยาวได้ผลดังรูปที่ 7 จะสังเกตเห็นได้ว่าในแต่ละโมเดลในกรณีที่มีค่าความต้านทานภายในแบตเตอรี่ร่วมด้วยจะทำให้โมเดลทำนายระดับประจุแบตเตอรี่ได้แม่นยำกว่าโมเดลที่ไม่มีค่าความต้านทานภายใน



รูปที่ 5 กราฟการทำนาย SoC ระหว่างอัดประจุโดยโมเดลต้นไม้ตัดสินใจ



รูปที่ 6 กราฟการทำนาย SoC ระหว่างอัดประจุโดยโมเดลโครงข่ายประสาทเทียม

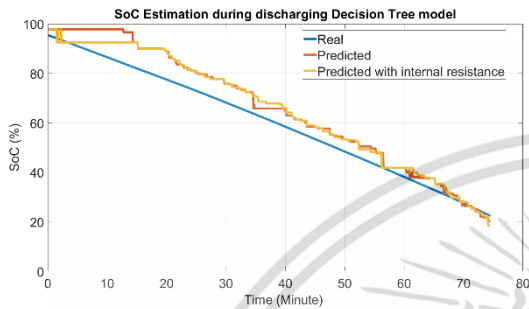


รูปที่ 7 กราฟการทำนาย SoC ระหว่างอัดประจุโดยโมเดลหน่วยความจำระยะสั้นยาว

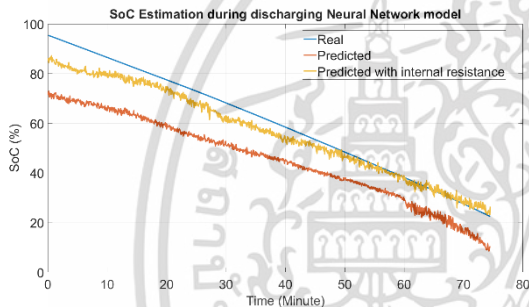
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทำนายระดับประจุแบตเตอรี่ระหว่างคายประจุ

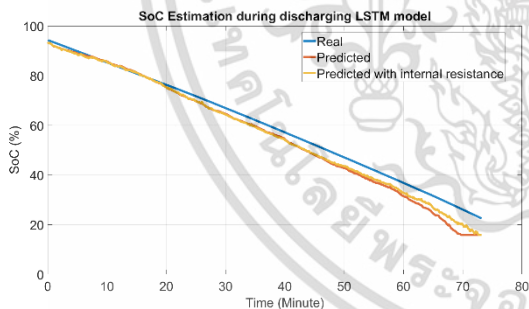
การทำนายระดับประจุแบตเตอรี่ระหว่างคายประจุสำหรับโมเดลชนิดต้นไม้ตัดสินใจได้ผลดังรูปที่ 8 โมเดลโครงข่ายประสาทเทียมได้ผลดังรูปที่ 9 และโมเดลหน่วยความจำระยะสั้นยาวได้ผลดังรูปที่ 10 จะสังเกตได้ว่าในแต่ละโมเดลในกรณีที่มีค่าความต้านทานภายในแบตเตอรี่ร่วมด้วยจะทำให้โมเดลทำนายระดับประจุแบตเตอรี่ได้แม่นยำกว่าโมเดลที่ไม่มีค่าความต้านทานภายใน



รูปที่ 8 กราฟการทำนาย SoC ระหว่างคายประจุโดยโมเดลต้นไม้ตัดสินใจ



รูปที่ 9 กราฟการทำนาย SoC ระหว่างคายประจุโดยโมเดลโครงข่ายประสาทเทียม



รูปที่ 10 กราฟการทำนาย SoC ระหว่างคายประจุโดยโมเดลหน่วยความจำระยะสั้นยาว

4.3 ค่าความผิดพลาด

การหาค่าความผิดพลาดในงานวิจัยนี้โดยค่าสัมบูรณ์เฉลี่ยของค่าความผิดพลาดและค่ารากที่สองของค่าเฉลี่ยกำลังสองของค่าความผิดพลาดเป็นไปตามตารางที่ 4 โดยสำหรับการอัดประจุและคายประจุโมเดลชนิด LSTM ร่วมกับค่าความต้านทานภายในให้ความผิดพลาดต่ำสุดมีค่าความผิดพลาดระหว่างอัดประจุ MAE 3.26% และ RMSE 4.24% ค่าความผิดพลาดระหว่างคายประจุ MAE 3.89% และ RMSE 4.53 %

ตารางที่ 4 ค่าความผิดพลาดระหว่างอัดและคายประจุ

	ค่าความผิดพลาดอัดประจุ		ค่าความผิดพลาดคายประจุ	
	MAE (%)	RMSE (%)	MAE (%)	RMSE (%)
Decision Tree	5.2	7.9	9.12	9.88
Decision Tree with internal resistance	5.19	7.9	8.41	8.95
Neural Network	6.34	8.05	13.75	15.07
Neural Network with internal resistance	4.2	5.07	4.64	5.50
LSTM	3.49	4.41	4.22	5.09
LSTM with internal resistance	3.26	4.24	3.89	4.53

5. สรุปผลการทดลอง

งานวิจัยการทำนายระดับประจุแบตเตอรี่ของกลุ่มแบตเตอรี่ลิเทียมไอออนโดยปัญญาประดิษฐ์ร่วมกับค่าความต้านทานภายในได้ทำการสร้างโมเดลปัญญาประดิษฐ์ชนิดต้นไม้ตัดสินใจ โครงข่ายประสาทเทียม และหน่วยความจำระยะสั้นยาว จากผลการทดลองสรุปได้ว่าโมเดลชนิดหน่วยความจำระยะสั้นยาวร่วมกับค่าความต้านทานภายในสามารถทำนายระดับประจุแบตเตอรี่ได้แม่นยำที่สุดโดยมีค่าความผิดพลาดระหว่างอัดประจุโดย MAE และ RMSE ได้ 3.26 % และ 4.24 % ตามลำดับ ค่าความผิดพลาดระหว่างคายประจุโดย MAE และ RMSE ได้ 3.89 % และ 4.53 %

เอกสารอ้างอิง

- Y. Zhou, M. Dong and J. Wu, "Hyperparameter Optimization for SOC Estimation by LSTM with Internal Resistance," *2021 International Conference on Computer Network, Electronic and Automation (ICCNEA)*, pp. 263-267, 2021.
- J. Lee and I. Lee, "Online Estimation Algorithm of SOC and SOH Using Neural Network for Lithium Battery," *3rd Eurasia Conference on IOT, Communication and Engineering (ECICE)*, pp. 568-571, 2021.
- D. H. C. Lam, J. Wong, Y. S. Lim and L. C. Hau, "State of Charge, State of Health, and Degradation Estimation for Second-Life Batteries for Maximum Demand Reductions," *2022 IEEE International Conference on Power and Energy (PECon)*, pp. 240-245, 2022.
- Y. Hu and Z. Wang, "Study on SOC Estimation of Lithium Battery Based on Improved BP Neural Network," *8th International Symposium on Next Generation Electronics (ISNE)*, pp. 1-3, 2019.
- F. Elmahdi, L. Ismail, and M. Noureddine, "Fitting the OCV-SOC relationship of a battery lithium-ion using genetic algorithm method," *9th The International Conference on Innovation, Modern Applied Science & Environmental Studies (ICIES2020)*, pp.5, 2020.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#include "Arduino.h"
#define BMS_LEN_response 45
typedef struct packCellInfoStruct{
    uint8_t NumOfCells;
    uint16_t CellVoltage[15];
    uint16_t CellLow;
    uint16_t CellHigh;
    uint16_t CellDiff;
    uint16_t CellAvg;
} packCellInfoStruct;
class JbdBms{public:
    JbdBms(HardwareSerial * t_hardwareSerial);
    bool readBmsData();
    bool readPackData();
    float getVoltage();
    float getCurrent();
    float getChargePercentage();
    uint16_t getProtectionState();
    uint16_t getCycle();
    float getTemp1();
    float getTemp2();
    float getTemp3();
    int getnumberofcell();
    int getnumberoftemp();
    int getbalancestatus1to16();
    int getnominalcapacity();
    int getresidualcapacity();
    packCellInfoStruct getPackCellInfo(); private:
    Stream * m_port;
    bool m_hwserial;
    float m_voltage = 0;
    float m_current = 0;
    float m_chargePercentage = 0;
    uint16_t m_protectionState = 0;
    uint16_t m_cycle = 0;
    float m_Temp1 = 0;
    float m_Temp2 = 0;
    float m_Temp3 = 0;
    int m_numberofcell = 0;
    int m_numberoftemp = 0;
    int m_balancestatus1to16 = 0;
    int m_nominalcapacity = 0;
    int m_residualcapacity = 0;
    packCellInfoStruct m_packCellInfo = {0};
    void sendReqBasicMessage();
    void passReqBasicMessage(uint8_t * t_message);
    void sendCellMessage();
    void passReqPackMessage(uint8_t * t_message);
    bool readresponse(uint8_t * t_outMessage);
    bool checkCheckSumRecieve(uint8_t * t_message);
    uint16_t computeCrc16JbdChina(uint8_t * puchMsg, uint8_t usDataLen);
    float converUint32ToFloat(uint32_t number);
    uint16_t convertTwoIntsToUint16(int highbyte, int lowbyte);
    int16_t convertTwoIntsToInt16(int highbyte, int lowbyte);
    bool checkreadbmsdata(uint8_t * t_message);
    bool checkreadpackdata(uint8_t * t_message);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

uint32_t getMaxTimeout();};
JbdBms::JbdBms(HardwareSerial * t_hardwareSerial){
    m_port = t_hardwareSerial;
    m_hwserial = true;
    static_cast<HardwareSerial*>(m_port)->begin(9600);}
bool JbdBms::readBmsData(){
    uint8_t response[BMS_LEN_response];
    sendReqBasicMessage();
    readresponse(response);
    if (checkCheckSumRecieve(response) == true)
        {passReqBasicMessage(response);} else {return false;}return true;}
bool JbdBms::readPackData(){
    uint8_t response[BMS_LEN_response];
    sendCellMessage();
    readresponse(response);
    if (checkCheckSumRecieve(response) == true) {
        if(checkreadpackdata(response) == true)
            {passReqPackMessage(response);}
        else {return false;}return true;}
float JbdBms::getVoltage() {return m_voltage;}
float JbdBms::getCurrent() {return m_current;}
float JbdBms::getChargePercentage(){return m_chargePercentage;}
uint16_t JbdBms::getProtectionState(){return m_protectionState;}
uint16_t JbdBms::getCycle(){return m_cycle;}
float JbdBms::getTemp1() {return m_Temp1;}
float JbdBms::getTemp2() {return m_Temp2;}
float JbdBms::getTemp3() {return m_Temp3;}
int JbdBms::getnumberofcell() {return m_numberofcell;}
int JbdBms::getnumberoftemp() {return m_numberoftemp;}
int JbdBms::getbalancestatus1to16() {return m_balancestatus1to16;}
int JbdBms::getnominalcapacity() {return m_nominalcapacity;}
int JbdBms::getresidualcapacity() {return m_residualcapacity;}
packCellInfoStruct JbdBms::getPackCellInfo() {return m_packCellInfo;}
void JbdBms::sendReqBasicMessage() {
    uint8_t reqMessage[] = { 0xDD, 0xA5, 0x03, 0x00, 0xFF, 0xFD, 0x77 };
    m_port->write(reqMessage, 7);}
void JbdBms::sendCellMessage() {
    uint8_t reqMessage[] = { 0xDD, 0xA5, 0x04, 0x00, 0xFF, 0xFC, 0x77 };
    m_port->write(reqMessage, 7);}
void JbdBms::passReqBasicMessage(uint8_t * t_message) {
    m_voltage = ((float)convertTwoIntsToUint16(t_message[4],
t_message[5])/100;
    m_current = (((float)convertTwoIntsToInt16(t_message[6],
t_message[7])) * 10;
    m_residualcapacity=((int)convertTwoIntsToUint16(t_message[8],
t_message[9]))*10;
    m_nominalcapacity=((int)convertTwoIntsToUint16(t_message[10],
t_message[11]))*10;
    m_cycle = (((float)convertTwoIntsToUint16(t_message[12],
t_message[13]));
    m_balancestatus1to16=convertTwoIntsToUint16(t_message[16],t_message[1
7]);
    m_protectionState = (((float)convertTwoIntsToUint16(t_message[20],
t_message[21]));
    m_chargePercentage = t_message[23];
    m_numberofcell=t_message[25];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    m_numberofTemp=t_message[26];
    m_Temp1 = (((float)convertTwoIntsToUint16(t_message[27],
t_message[28])) - 2731) / 10.00f;
    m_Temp2 = (((float)convertTwoIntsToUint16(t_message[29],
t_message[30])) - 2731) / 10.00f;
    m_Temp3 = (((float)convertTwoIntsToUint16(t_message[31],
t_message[32])) - 2731) / 10.00f;}
void JbdBms::passReqPackMessage(uint8_t * t_message){
    uint16_t _cellSum = 0;
    uint16_t _CellLow = 5000;
    uint16_t _CellHigh = 0;
    m_packCellInfo.NumOfCells = t_message[3] / 2;
    byte offset = 4;
    for (byte i = 0; i < m_packCellInfo.NumOfCells; i++)
    {m_packCellInfo.CellVoltage[i] =
((uint16_t)convertTwoIntsToUint16(t_message[i * 2 + offset],
t_message[i * 2 + 1 + offset]));
    _cellSum += m_packCellInfo.CellVoltage[i];
    if (m_packCellInfo.CellVoltage[i] > _CellHigh){_CellHigh =
m_packCellInfo.CellVoltage[i];}
    if (m_packCellInfo.CellVoltage[i] < _CellLow){ _CellLow =
m_packCellInfo.CellVoltage[i];}
    m_packCellInfo.CellLow = _CellLow;
    m_packCellInfo.CellHigh = _CellHigh;
    m_packCellInfo.CellDiff = _CellHigh - _CellLow;
    m_packCellInfo.CellAvg = _cellSum / m_packCellInfo.NumOfCells;}}
bool JbdBms::readresponse(uint8_t *t_outMessage)
{uint8_t i = 0;
    bool findBeginByte = false;
    uint32_t statrTime = millis();
    while (i <= BMS_LEN_response - 1)
    {if (abs((millis() - statrTime) > getMaxTimeout())){return false;}
    if (m_port->available() > 0) {uint8_t thisByte = m_port->read();
        if (thisByte == 0xDD){findBeginByte = true;}
        if (findBeginByte){t_outMessage[i] = thisByte;i++;}}return
true;}
bool JbdBms::checkChecksumRecieve(uint8_t *t_message)
{uint16_t checkSumCompute;
    uint16_t checkSumAccepted;
    uint8_t lengthData;
    uint8_t startIndexCS;
    if (t_message[2] != 0) {return false;}
    lengthData = t_message[3];
    checkSumCompute = computeCrc16JbdChina(t_message, BMS_LEN_response);
    startIndexCS = lengthData + 4;
    checkSumAccepted = (t_message[startIndexCS] << 8) |
t_message[startIndexCS + 1];
    if (checkSumCompute != checkSumAccepted)
        {return false;}return true;}
uint16_t JbdBms::computeCrc16JbdChina(uint8_t *puchMsg, uint8_t
usDataLen)
{uint8_t lengthData = puchMsg[3];
    uint16_t summa = 0;
    for (int i = 4; i < lengthData + 4; i++)
        summa = summa + puchMsg[i];
    uint16_t checkSum = (summa + lengthData - 1) ^ 0xFFFF;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    return checksum;}
uint32_t JbdBms::getMaxTimeout(){return 100;}
int16_t JbdBms::convertTwoIntsToInt16(int highbyte, int lowbyte) {
    int16_t value = convertTwoIntsToUint16(highbyte, lowbyte);
    if (value & 0b1000000000000000)
        {value = ~ value;value = value * -1;}return value;}
uint16_t JbdBms::convertTwoIntsToUint16(int highbyte, int lowbyte)
{uint16_t a16bitvar = (highbyte);
  a16bitvar <<= 8;
  a16bitvar = (a16bitvar | lowbyte);
  return a16bitvar;}
bool JbdBms::checkreadbmsdata(uint8_t *t_message)
{if (t_message[1] == 0x03){return true;} else
  return false;}
bool JbdBms::checkreadpackdata(uint8_t *t_message)
{if (t_message[1] == 0x04){return true;} else
  return false;}
#include "WiFi.h"
#include <HTTPClient.h>
#include "time.h"
const char* ntpServer = "pool.ntp.org";
const long  gmtOffset_sec = 7*60*60;
const int   daylightOffset_sec = 0;
const char* ssid = "*****";
const char* password = "*****";
String GOOGLE_SCRIPT_ID = "*****";
JbdBms myBms(&Serial2);
void setup()
{WiFi.begin(ssid, password);while (WiFi.status() !=
WL_CONNECTED){delay(500);}
  configTime(gmtOffset_sec, daylightOffset_sec, ntpServer);}
void loop()
{unsigned long starttime,endtime;
  int attempts = 0, cellnum = 0, numberofcell = 0, numberoftemp = 0,
status[16]={0}, nomcap = 0, rescap = 0;
  float cap = 0.0, cur = 0.0, vol = 0.0, temp1 = 0.0, temp2 = 0.0,
temp3 = 0.0, highcell = 0.0, lowcell = 0.0, avgcell = 0.0, diffcell =
0.0, cell[7]={0};
  String urlcell;
  String urlbalance;
  String urlfinal;
  String urltotal;
  if (WiFi.status() == WL_CONNECTED) {
    struct tm timeinfo;
    if (!getLocalTime(&timeinfo)) {return;}
    char date[50];
    char time[50];
    strftime(date, sizeof(date), "%A-%B-%d-%Y", &timeinfo);
    strftime(time, sizeof(time),"%H:%M:%S", &timeinfo);
    String Strdate(date);
    bool hasBmsData = false;
    while (!hasBmsData) {
      if (attempts > 10) break;
      hasBmsData = myBms.readBmsData();
      delay(200);
      attempts++;}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (myBms.readBmsData() == true)
{cap = myBms.getChargePercentage();
  cur = myBms.getCurrent();
  vol = myBms.getVoltage();
  temp1 = myBms.getTemp1();
  temp2 = myBms.getTemp2();
  temp3 = myBms.getTemp3();
  numberofcell = myBms.getnumberofcell();
  numberoftemp = myBms.getnumberoftemp();
  for (int j = 0; j < numberofcell; j++)
{status[j]=bitRead(myBms.getbalancestatus1to16(), j);}
  nomcap = myBms.getnominalcapacity();
  rescap = myBms.getresidualcapacity();} else{}
attempts = 0;
bool hasPackData = false;
while (!hasPackData) {
  if (attempts > 8) break;
  hasPackData = myBms.readPackData();
  delay(200);
  attempts++;}
if (myBms.readPackData()== true)
{packCellInfoStruct packInfo = myBms.getPackCellInfo();
  lowcell = packInfo.CellLow;
  highcell = packInfo.CellHigh;
  diffcell = packInfo.CellDiff;
  avgcell = packInfo.CellAvg;
  for (byte i = 0; i < packInfo.NumOfCells; i++)
  {cell[i]= packInfo.CellVoltage[i];
  cellnum = packInfo.NumOfCells;}}else {}
for (int i = 0; i < cellnum; i++)
  {urlcell = urlcell+"&cell"+(i+1)+"="+cell[i];
  urlbalance = urlbalance+"&cellbal"+(i+1)+"="+status[i];}
urltotal =
"https://script.google.com/macros/s/"+GOOGLE_SCRIPT_ID+"/exec?"+date="
+
Strdate+"&time="+time+"&cap="+cap+"&cur="+cur+"&vol="+vol+"&temp1="+tem
p1+"&temp2="+temp2+"&temp3="+temp3;
urlFinal =
urltotal+urlcell+"&cellhigh="+highcell+"&celllow="+lowcell+"&cellavg="+
avgcell+"&celldiff="+diffcell+urlbalance+"&nomcap="+nomcap+"&rescap="+r
escap;}
HTTPClient http;
http.begin(urlFinal.c_str());
http.setFollowRedirects(HTTPC_STRICT_FOLLOW_REDIRECTS);
int httpCode = http.GET();
String payload;
if (httpCode > 0) {payload = http.getString();}
http.end();}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ค

โปรแกรม Jupyter Lab โมเดลปัญญาประดิษฐ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

import numpy as np
import pandas as pd
import sklearn
from sklearn.model_selection import train_test_split
from sklearn import preprocessing
from sklearn.preprocessing import MinMaxScaler
from keras.models import Sequential
from keras.layers import SimpleRNN,LSTM,GRU,Activation,Dense,Dropout
import tensorflow as tf
from keras.callbacks import ModelCheckpoint
from keras.models import load_model
from numpy.testing import assert_allclose
import keras
from tensorflow import Graph
from keras.layers import TimeDistributed
import time
import os
import numpy as np
import tensorflow as tf
import random as python_random
global_seed = 42
np.random.seed(global_seed)
python_random.seed(global_seed)
tf.random.set_seed(global_seed)
tf.keras.utils.set_random_seed(global_seed)
os.environ['PYTHONHASHSEED'] = str(global_seed)
dateparse = lambda x: datetime.strptime(x, '%Y-%m-%d %H:%M:%S')
Data = pd.read_excel('*****.xlsx',parse_dates=[[ 'Date', 'Time' ]])
modelselect = 'MODEL TYPE'
timestep = NUMBER OF TIMESTEP
neurons = NUMBER OF NEURONS
output_size = 1
epochs = NUMBER OF EPOCHS
batch_size = NUMBER OF BATCH SIZE
loss = 'LOSS FUNCTION'
dropout = NUMBER OF DROPOUT
optimizer = 'OPTIMIZER FUNCTION'
activ_func = 'ACTIVATION FUNCTION'
learning_rate = NUMBER OF LEARNING RATE
drop = 1
dropir = *True OR False*
train_size = TRAIN SIZE RATIO
def find_error(Y, Yhat, TypeOfError):
    if TypeOfError == 'MAE':
        error = find_MAE(Y, Yhat)
    elif TypeOfError == 'RMSE':
        error = find_RMSE(Y, Yhat)
    return error
def find_MAE(Y, Yhat):
    N = Y.shape[0]
    MAE = (np.abs(Y - Yhat)).sum()/N
    return MAE
def find_RMSE(Y, Yhat):
    RMSE = np.sqrt(np.mean((Yhat-Y)**2))
    return RMSE
def prepare_data(X, y, timestep=timestep, train_size = train_size):

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

X_train,X_test,y_train,y_test = train_test_split(X,y, train_size =
train_size, random_state = 7, shuffle = False)
X_trainlstm = extract_timestep(X_train, timestep)
X_testlstm = extract_timestep(X_test, timestep)
y_trainlstm = y_train[timestep:].values
y_testlstm = y_test[timestep:].values
return X_trainlstm, y_trainlstm, X_train, y_train, X_testlstm,
y_testlstm, X_test, y_test
def prepare_test_data(X2, y2, timestep=timestep):
X2 = extract_timestep(X2, timestep)
y2 = y2[timestep:].values
return X2, y2
def extract_timestep(df, timestep=timestep):
timestep_data = []
for idx in range(len(df) - timestep):
tmp = df[idx: (idx + timestep)].copy()
timestep_data.append(tmp.values)
return np.array(window_data)
def prepare_input_data(Data):
df = pd.DataFrame(Data)
df = df.dropna()
if ocv == 0:
df.drop(['Timestamp', 'Unnamed: 1', 'Unnamed: 2', 'TEMP IR 1',
'TEMP IR 2', 'Calculate SoC', 'TEMP IR 3', 'TEMP IR 4', 'TEMP IR 5',
'TEMP IR 6', 'TEMP IR 7', 'Capacity', 'Calculate Capacity', 'Unnamed:
3'],axis=1,inplace=True)
if ocv == 1:
df.drop(['Timestamp', 'Unnamed: 1', 'Unnamed: 2', 'TEMP IR 1',
'TEMP IR 2', 'Open Circuit Voltage', 'TEMP IR 3', 'TEMP IR 4', 'TEMP IR
5', 'TEMP IR 6', 'TEMP IR 7', 'Capacity', 'Calculate Capacity', 'Unnamed:
3'
],axis=1,inplace=True)
df['Date_Time'] = pd.to_datetime(df['Date_Time'])
df.rename(columns={'Date_Time': 'Time'},inplace=True)
df = df.set_index('Time')
scaler = MinMaxScaler(feature_range=(0, 1))
scaler.fit(df)
df=pd.DataFrame(scaler.transform(df), index=df.index, columns =
df.columns)
if ocv == 0:
X=df.drop(columns=['Open Circuit Voltage'])
y=df['Open Circuit Voltage']
if ocv == 1:
X=df.drop(columns=['Calculate SoC'])
y=df['Calculate SoC']
X_trainlstm,y_trainlstm,X_train,
y_train,X_testlstm,y_testlstm,X_test,y_test = prepare_data(X, y,
timestep=timestep,train_size=train_size)
return X_trainlstm,y_trainlstm,X_train,
y_train,X_testlstm,y_testlstm,X_test,y_test,scaler,df.columns
def prepare_vali_data_clean(Data):
df = pd.DataFrame(Data)
df = df.dropna()
if ocv == 0:
df.drop(['Timestamp', 'Unnamed: 1', 'Unnamed: 2', 'TEMP IR 1',
'TEMP IR 2', 'Calculate SoC', 'TEMP IR 3', 'TEMP IR 4', 'TEMP IR 5',

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

'TEMP IR 6', 'TEMP IR 7', 'Capacity', 'Calculate Capacity', 'Unnamed:
3', axis=1, inplace=True)
    if ocv == 1:
        df.drop(['Timestamp', 'Unnamed: 1', 'Unnamed: 2', 'TEMP IR 1',
'TEMP IR 2', 'Open Circuit Voltage', 'TEMP IR 3', 'TEMP IR 4', 'TEMP IR
5', 'TEMP IR 6', 'TEMP IR 7', 'Capacity', 'Calculate Capacity', 'Unnamed:
3'
                ], axis=1, inplace=True)
df['Date_Time'] = pd.to_datetime(df['Date_Time'])
df.rename(columns={'Date_Time': 'Time'}, inplace=True)
df = df.set_index('Time')
if ocv == 0:
    X=df.drop(columns=['Open Circuit Voltage'])
    y=df['Open Circuit Voltage']
if ocv == 1:
    X=df.drop(columns=['Calculate SoC'])
    y=df['Calculate SoC']
X_train,X_test,y_train,y_test = train_test_split(X,y, train_size =
train_size, random_state = 7, shuffle = False)
return X_train,X_test,y_train,y_test
def prepare_vali_data(Data):
    df = Data
    scaler = MinMaxScaler(feature_range=(0, 1))
    scaler.fit(df)
    df=pd.DataFrame(scaler.transform(df), index=df.index, columns =
df.columns)
    if ocv == 0:
        X=df.drop(columns=['Open Circuit Voltage'])
        y=df['Open Circuit Voltage']
    if ocv == 1:
        X=df.drop(columns=['Calculate SoC'])
        y=df['Calculate SoC']
    Xlstm,ylstm = prepare_test_data(X, y, timestep=timestep)
    return Xlstm,ylstm,X,y,scaler, df.columns
def prepare_vali_data_test(Data,scaler):
    df = Data
    df=pd.DataFrame(scaler.transform(df), index=df.index, columns =
df.columns)
    if ocv == 0:
        X=df.drop(columns=['Open Circuit Voltage'])
        y=df['Open Circuit Voltage']
    if ocv == 1:
        X=df.drop(columns=['Calculate SoC'])
        y=df['Calculate SoC']
    Xlstm,ylstm = prepare_test_data(X, y, timestep=timestep)
    return Xlstm,ylstm,X,y
def prepare_input_data_nodrop(Data):
    df = pd.DataFrame(Data)
    df = df.dropna()
    if ocv == 0:
        df.drop(['Timestamp', 'Unnamed: 1', 'Unnamed: 2', 'Calculate SoC',
'Capacity', 'Calculate Capacity', 'Unnamed: 3'
                ], axis=1, inplace=True)
    if ocv == 1:
        df.drop(['Timestamp', 'Unnamed: 1', 'Unnamed: 2', 'Open Circuit
Voltage',

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        'Capacity', 'Calculate Capacity', 'Unnamed: 3'
    ],axis=1,inplace=True)
df['Date_Time'] = pd.to_datetime(df['Date_Time'])
df.rename(columns={'Date_Time': 'Time'},inplace=True)
df = df.set_index('Time')
scaler = MinMaxScaler(feature_range=(0, 1))
scaler.fit(df)
df=pd.DataFrame(scaler.transform(df), index=df.index, columns =
df.columns)
if ocv == 0:
    X=df.drop(columns=['Open Circuit Voltage'])
    y=df['Open Circuit Voltage']
if ocv == 1:
    X=df.drop(columns=['Calculate SoC'])
    y=df['Calculate SoC']
X_trainlstm,y_trainlstm,X_train,
y_train,X_testlstm,y_testlstm,X_test,y_test = prepare_data(X, y,
timestep=timestep,train_size=train_size)
return X_trainlstm,y_trainlstm,X_train,
y_train,X_testlstm,y_testlstm,X_test,y_test,scaler,df.columns
def prepare_input_data_same(Data,scaler):
    df = pd.DataFrame(Data)
    df = df.dropna()
    if ocv == 0:
        df.drop(['Timestamp', 'Unnamed: 1', 'Unnamed: 2', 'TEMP IR 1',
'TEMP IR 2', 'Calculate SoC', 'TEMP IR 3', 'TEMP IR 4', 'TEMP IR 5',
'TEMP IR 6', 'TEMP IR 7', 'Capacity', 'Calculate Capacity', 'Unnamed:
3'],axis=1,inplace=True)
    if ocv == 1:
        df.drop(['Timestamp', 'Unnamed: 1', 'Unnamed: 2', 'TEMP IR 1',
'TEMP IR 2', 'Open Circuit Voltage', 'TEMP IR 3', 'TEMP IR 4', 'TEMP IR
5', 'TEMP IR 6', 'TEMP IR 7', 'Capacity', 'Calculate Capacity', 'Unnamed:
3'
        ],axis=1,inplace=True)
    df['Date_Time'] = pd.to_datetime(df['Date_Time'])
    df.rename(columns={'Date_Time': 'Time'},inplace=True)
    df = df.set_index('Time')
    df=pd.DataFrame(scaler.transform(df), index=df.index, columns =
df.columns)
    if ocv == 0:
        X=df.drop(columns=['Open Circuit Voltage'])
        y=df['Open Circuit Voltage']
    if ocv == 1:
        X=df.drop(columns=['Calculate SoC'])
        y=df['Calculate SoC']
    Xlstm,ylstm = prepare_test_data(X, y, timestep=timestep)
    return Xlstm,ylstm,X,y
def prepare_error_data_same(Data,scaler):
    df = pd.DataFrame(Data)
    df = df.dropna()
    if ocv == 0:
        df.drop(['Timestamp', 'Unnamed: 1', 'Unnamed: 2', 'TEMP IR 1',
'TEMP IR 2', 'Calculate SoC', 'TEMP IR 3', 'TEMP IR 4', 'TEMP IR 5',
'TEMP IR 6', 'TEMP IR 7', 'Capacity', 'Calculate Capacity', 'Unnamed:
3'],axis=1,inplace=True)
    if ocv == 1:

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

df.drop(['Timestamp', 'Unnamed: 1', 'Unnamed: 2', 'TEMP IR 1',
'TEMP IR 2', 'Open Circuit Voltage', 'TEMP IR 3', 'TEMP IR 4', 'TEMP IR
5', 'TEMP IR 6', 'TEMP IR 7', 'Capacity', 'Calculate Capacity', 'Unnamed:
3'
],axis=1,inplace=True)
df['Date_Time'] = pd.to_datetime(df['Date_Time'])
df.rename(columns={'Date_Time': 'Time'},inplace=True)
df = df.set_index('Time')
df=pd.DataFrame(scaler.transform(df), index=df.index, columns =
df.columns)
if ocv == 0:
    X=df.drop(columns=['Open Circuit Voltage'])
    y=df['Open Circuit Voltage']
if ocv == 1:
    X=df.drop(columns=['Calculate SoC'])
    y=df['Calculate SoC']
Xlstm,ylstm = prepare_test_data(X, y, timestep=timestep)
return Xlstm,ylstm,X,y,df.columns
def prepare_input_data_round_same(Data,scaler):
df = pd.DataFrame(Data)
df=pd.DataFrame(scaler.transform(df), index=df.index, columns =
df.columns)
X=df.drop(columns=['Calculate SoC'])
y=df['Calculate SoC']
Xlstm,ylstm = prepare_test_data(X, y, timestep=timestep)
return Xlstm,ylstm,X,y
def prepare_error_data_round_same(Data,scaler):
df = pd.DataFrame(Data)
df = df.dropna()
df=pd.DataFrame(scaler.transform(df), index=df.index, columns =
df.columns)
if ocv == 0:
    X=df.drop(columns=['Open Circuit Voltage'])
    y=df['Open Circuit Voltage']
if ocv == 1:
    X=df.drop(columns=['Calculate SoC'])
    y=df['Calculate SoC']
Xlstm,ylstm = prepare_test_data(X, y, timestep=timestep)
return Xlstm,ylstm,X,y
def prepare_scaler_round(Data):
df = pd.DataFrame(Data)
scaler = MinMaxScaler(feature_range=(0, 1))
scaler.fit(df)
df=pd.DataFrame(scaler.transform(df), index=df.index, columns =
df.columns)
return scaler
def build_model(input_data, output_size, neurons=lstm_neurons,
activ_func=activ_func, dropout=dropout, loss=loss,
optimizer=optimizer,learning_rate = learning_rate):
model = Sequential()
if modelselect == 'RNN':
    model.add(SimpleRNN(units=neurons,input_shape=(input_data.shape
[1], input_data.shape[2])))
elif modelselect == 'LSTM':
    model.add(LSTM(neurons, input_shape=(input_data.shape[1],
input_data.shape[2]), return_sequences = False))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

elif modelselect == 'GRU':
    model.add(GRU(neurons, input_shape=(input_data.shape[1],
input_data.shape[2]), return_sequences = False))
    model.add(Dropout(dropout))
    model.add(Dense(units=output_size))
    model.add(Activation(activ_func))
    optimizer = keras.optimizers.Adam(learning_rate=learning_rate,
beta_1=0.9, beta_2=0.999, epsilon=1e-7)
    model.compile(loss=loss, optimizer=optimizer)
    return model
ocv = 0
X_trainlstm,y_trainlstm,X_train,
y_train,X_testlstm,y_testlstm,X_test,y_test,scale,dfcolumns =
prepare_input_data(Data)
ocv = 1
XX_trainlstm,yy_trainlstm,XX_train,yy_train,XX_testlstm,yy_testlstm,XX_
test,yy_test,scalee,dfcolumnss = prepare_input_data(Data)
ocv = 1
X_trainlstm2,y_trainlstm2,X_train2,y_train2,X_testlstm2,y_testlstm2,X_t
est2,y_test2,scale2,dfcolumns2 = prepare_input_data_nodrop(Data)
model = build_model(X_trainlstm, output_size, neurons=lstm_neurons,
activ_func=activ_func, dropout=dropout, loss=loss, optimizer=optimizer)
if dropir == True:
    model2 = build_model(X_trainlstm, output_size,
neurons=lstm_neurons, activ_func=activ_func, dropout=dropout,
loss=loss, optimizer=optimizer)
else:
    model2 = build_model(X_trainlstm2, output_size,
neurons=lstm_neurons, activ_func=activ_func, dropout=dropout,
loss=loss, optimizer=optimizer)
start = time.time()
if dropir == False:
    model.fit(X_trainlstm,y_trainlstm,epochs=epochs,
batch_size=batch_size, verbose=0, shuffle=True)
    yhattrain = model.predict(X_trainlstm,verbose = 0)
    yhattrain = pd.DataFrame(yhattrain)
    yhattrain = yhattrain.reset_index()
    X_train1 = X_train.iloc[timestep:]
    X_train1 = X_train1.reset_index()
    xyconcat = pd.concat([X_train1,yhattrain],axis='columns')
    xyconcat = xyconcat.drop(columns=['Time','index'])
    xyconcat = scale.inverse_transform(xyconcat)
    xyconcat = pd.DataFrame(xyconcat)
    xyconcat = xyconcat.set_axis(dfcolumns, axis=1)
    cell = ["Cell 1","Cell 2","Cell 3","Cell 4","Cell 5","Cell 6","Cell
7"]
    for vi in range(0,7):
        ir = abs((xyconcat['Open Circuit Voltage']-
xyconcat[cell[vi]])/xyconcat['Current'])
        xyconcat = pd.concat([xyconcat, ir], axis=1)
        xyconcat.rename(columns={0: ("IR Cell " + str(vi+1))}, inplace
= True)
    xyconcatsoc = xyconcat['Open Circuit Voltage']
    xyconcat = xyconcat.drop(columns=['Open Circuit Voltage'])
    xyconcat = pd.concat([xyconcat, xyconcatsoc], axis=1)
    xyconcatir = xyconcat

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

xyconcattrain = pd.concat([XX_train,yy_train],axis='columns')
xyconcattrain = scalee.inverse_transform(xyconcattrain)
xyconcattrain = pd.DataFrame(xyconcattrain)
xyconcattrain = xyconcattrain.set_axis(dfcolumnss, axis=1)
xyconcattrainsoc = xyconcattrain['Calculate SoC']
xyconcattrainsoc = xyconcattrainsoc.iloc[timestep:]
xyconcattrainsoc = xyconcattrainsoc.reset_index()
xyconcatir =
pd.concat([xyconcatir,xyconcattrainsoc],axis='columns')
xyconcatir = xyconcatir.drop(columns=['index','Open Circuit
Voltage'])
else:
xyconcattrain = pd.concat([XX_train,yy_train],axis='columns')
xyconcattrain = scalee.inverse_transform(xyconcattrain)
xyconcattrain = pd.DataFrame(xyconcattrain)
xyconcattrain = xyconcattrain.set_axis(dfcolumnss, axis=1)
xyconcattrain = xyconcattrain.iloc[timestep:]
xyconcatir = xyconcattrain
scale_train2 = prepare_scaler_round(xyconcatir)
X2lstm,y2lstm,X2, y2 =
prepare_input_data_round_same(xyconcatir,scale_train2)
model2.fit(X2lstm,y2lstm,epochs=epochs, batch_size=batch_size,
verbose=0, shuffle=True)
end = time.time()
print("Train time = ",end - start,"s")
start = time.time()
if dropir == False:
yhattrain = model.predict(X_trainlstm,verbose = 0)
yhattrain = pd.DataFrame(yhattrain)
yhattrain = yhattrain.reset_index()
X_train1 = X_train.iloc[timestep:]
X_train1 = X_train1.reset_index()
xyconcat = pd.concat([X_train1,yhattrain],axis='columns')
xyconcat = xyconcat.drop(columns=['Time','index'])
xyconcat = scale.inverse_transform(xyconcat)
xyconcat = pd.DataFrame(xyconcat)
xyconcat = xyconcat.set_axis(dfcolumns, axis=1)
cell = ["Cell 1","Cell 2","Cell 3","Cell 4","Cell 5","Cell 6","Cell
7"]
for vi in range(0,7):
ir = abs((xyconcat['Open Circuit Voltage']-
xyconcat[cell[vi]])/xyconcat['Current'])
xyconcat = pd.concat([xyconcat, ir], axis=1)
xyconcat.rename(columns={0: ("IR Cell " + str(vi+1))}, inplace
= True)
xyconcatsoc = xyconcat['Open Circuit Voltage']
xyconcat = xyconcat.drop(columns=['Open Circuit Voltage'])
xyconcat = pd.concat([xyconcat, xyconcatsoc], axis=1)
xyconcatir = xyconcat
xyconcattrain = pd.concat([XX_train,yy_train],axis='columns')
xyconcattrain = scalee.inverse_transform(xyconcattrain)
xyconcattrain = pd.DataFrame(xyconcattrain)
xyconcattrain = xyconcattrain.set_axis(dfcolumnss, axis=1)
xyconcattrainsoc = xyconcattrain['Calculate SoC']
xyconcattrainsoc = xyconcattrainsoc.iloc[timestep:]
xyconcattrainsoc = xyconcattrainsoc.reset_index()

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

xyconcatir =
pd.concat([xyconcatir,xyconcattrain],axis='columns')
xyconcatir = xyconcatir.drop(columns=['index','Open Circuit
Voltage'])
else:
xyconcattrain = pd.concat([XX_train,yy_train],axis='columns')
xyconcattrain = scalee.inverse_transform(xyconcattrain)
xyconcattrain = pd.DataFrame(xyconcattrain)
xyconcattrain = xyconcattrain.set_axis(dfcolumnss, axis=1)
xyconcattrain = xyconcattrain.iloc[timestep:]
xyconcatir = xyconcattrain
scale_train2 = prepare_scaler_round(xyconcatir)
X21stm,y21stm,X2, y2 =
prepare_input_data_round_same(xyconcatir,scale_train2)
yhat_train = model2.predict(X21stm,verbose = 0)
ylstmindex = y2[timestep:]
yhat_train = pd.DataFrame(yhat_train, index = ylstmindex.index)
invtrain = pd.concat([X2,yhat_train ],axis='columns')
invtrain = scale_train2.inverse_transform(invtrain )
invtrain = pd.DataFrame(invtrain )
invtrain = invtrain .dropna()
a = len(invtrain .columns)
yhat_train = invtrain .iloc[:,a-1].values
X_train1 = XX_train.iloc[timestep:]
X_train1 = X_train1.reset_index()
y_train1 = yy_train.iloc[timestep:]
y_train1 = y_train1.reset_index()
invchecktest = pd.concat([X_train1,y_train1],axis='columns')
invchecktest = invchecktest .drop(columns=['Time'])
invchecktest = scalee.inverse_transform(invchecktest )
invchecktest = pd.DataFrame(invchecktest )
a = len(invchecktest.columns)
invchecktest = invchecktest.iloc[timestep:,a-1].values
errorlstm3 = invchecktest
error_train_mae = find_error(errorlstm3, yhat_train , 'MAE')
error_train_rmse = find_error(errorlstm3, yhat_train , 'RMSE')
end = time.time()
print("Predict on train data time = ",end - start,"s")
start = time.time()
if dropir == False:
yhat = model.predict(X_testlstm,verbose = 0)
yhat = pd.DataFrame(yhat)
yhat = yhat.reset_index()
X_test1 = X_test.iloc[timestep:]
X_test1 = X_test1.reset_index()
errorxyconcat = pd.concat([X_test1,yhat],axis='columns')
errorxyconcat = errorxyconcat.drop(columns=['Time','index'])
errorxyconcat = scale.inverse_transform(errorxyconcat)
errorxyconcat = pd.DataFrame(errorxyconcat)
errorxyconcat = errorxyconcat.set_axis(dfcolumns, axis=1)
cell = ["Cell 1","Cell 2","Cell 3","Cell 4","Cell 5","Cell 6","Cell
7"]
for vi in range(0,7):
ir = abs((errorxyconcat['Open Circuit Voltage']-
errorxyconcat[cell[vi]])/errorxyconcat['Current'])
errorxyconcat = pd.concat([errorxyconcat, ir], axis=1)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        errorxyconcat.rename(columns={0: ("IR Cell " + str(vi+1))},
inplace = True)
        errorxyconcatsoc = errorxyconcat['Open Circuit Voltage']
        errorxyconcatsoc = errorxyconcatsoc.drop(columns=['Open Circuit
Voltage'])
        errorxyconcat = pd.concat([errorxyconcat, errorxyconcatsoc],
axis=1)
        errorxyconcatir = errorxyconcat
        xyconcatteest = pd.concat([XX_test,yy_test],axis='columns')
        xyconcatteest = scalee.inverse_transform(xyconcatteest)
        xyconcatteest = pd.DataFrame(xyconcatteest)
        xyconcatteest = xyconcatteest.set_axis(dfcolumnss, axis=1)
        xyconcatteestsoc = xyconcatteest['Calculate SoC']
        xyconcatteestsoc = xyconcatteestsoc.iloc[timestep:]
        xyconcatteestsoc = xyconcatteestsoc.reset_index()
        errorxyconcatir =
pd.concat([errorxyconcatir,xyconcatteestsoc],axis='columns')
        errorxyconcatir = errorxyconcatir.drop(columns=['index','Open
Circuit Voltage'])
    else:
        xyconcatteest = pd.concat([XX_test,yy_test],axis='columns')
        xyconcatteest = scalee.inverse_transform(xyconcatteest)
        xyconcatteest = pd.DataFrame(xyconcatteest)
        xyconcatteest = xyconcatteest.set_axis(dfcolumnss, axis=1)
        xyconcatteest = xyconcatteest.iloc[timestep:]
        errorxyconcatir = xyconcatteest
        X_test1 = XX_test.iloc[timestep:]
        X_test1 = X_test1.reset_index()
        Xlstm,ylstm,X, y =
prepare_error_data_round_same(errorxyconcatir,scale_train2)
        yhat2 = model2.predict(Xlstm,verbose = 0)
        ylstmindex = y[timestep:]
        yhat2 = pd.DataFrame(yhat2, index = ylstmindex.index)
        invttest = pd.concat([X,yhat2],axis='columns')
        invttest = scale_train2.inverse_transform(invttest)
        invttest = pd.DataFrame(invttest)
        invttest = invttest.dropna()
        a = len(invttest.columns)
        yhat2 = invttest.iloc[:,a-1].values
        y_test1 = yy_test.iloc[timestep:]
        y_test1 = y_test1.reset_index()
        invchecktest = pd.concat([X_test1,y_test1],axis='columns')
        invchecktest = invchecktest.drop(columns=['Time'])
        invchecktest = scalee.inverse_transform(invchecktest)
        invchecktest = pd.DataFrame(invchecktest)
        a = len(invchecktest.columns)
        invchecktest = invchecktest.iloc[timestep:,a-1].values
        errorlstm2 = invchecktest
        error_test_mae = find_error(errorlstm2, yhat2, 'MAE')
        error_test_rmse = find_error(errorlstm2, yhat2, 'RMSE')
        end = time.time()
        print("Predict on test data time = ",end - start,"s")
        print("Train MAE Error = ",error_train_mae)
        print("Train RMSE Error = ",error_train_rmse)
        print("Test MAE Error = ",error_test_mae)
        print("Test RMSE Error = ",error_test_rmse)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อ-นามสกุล นายพศุทธิ์ กรุดสอน
วัน เดือน ปีเกิด 12 มีนาคม 2542
ประวัติการศึกษา 2564 วิศวกรรมศาสตรบัณฑิต วิศวกรรมไฟฟ้า (เกียรตินิยมอันดับ 2)
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้