

การพยากรณ์การจราจรข้อมูลเครือข่ายโดยใช้การเรียนรู้ของเครื่องและการ
เลื่อนหน้าต่าง



วิทยานิพนธ์นี้สำหรับการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคมและคอมพิวเตอร์
มหาวิทยาลัยเทคโนโลยีสุรนารี
ปีการศึกษา 2564

FORECASTING NETWORK DATA TRAFFIC USING MACHINE
LEARNING METHODS AND SLIDING WINDOW



A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Telecommunication and Computer Engineering

Suranaree University of Technology

Academic Year 2021

การพยากรณ์การจราจรข้อมูลเครือข่ายโดยใช้การเรียนรู้ของเครื่องและการเลื่อน
หน้าต่าง

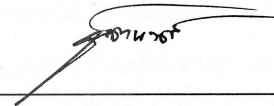
มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้บัณฑิตวิทยาลัยฉบับนี้เป็นส่วนหนึ่งของการศึกษา
ตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

คณะกรรมการสอบวิทยานิพนธ์



(อ. ดร.วรวัฒน์ ลวนนท์)

ประธานกรรมการ



(ผศ. ดร.ชิตพงษ์ เวชโธสงค์)

กรรมการ (อาจารย์ที่ปรึกษาวิทยานิพนธ์)



(อ. ดร.ปธิกร อันซีน)

กรรมการ



(รศ. ดร.ฉัตรชัย โชติษฐียงกูร)

รองอธิการบดีฝ่ายวิชาการและประกันคุณภาพ



(รศ. ดร.พรศิริ จงกล)

คณบดีสำนักวิชาวิศวกรรมศาสตร์

ศุภกานต์ ประจํา : การพยากรณ์การจราจรข้อมูลเครือข่ายโดยใช้การเรียนรู้ของเครื่องและการเลื่อนหน้าต่าง (FORECASTING NETWORK DATA TRAFFIC USING MACHINE LEARNING METHODS AND SLIDING WINDOW) อาจารย์ที่ปรึกษา : ผู้ช่วยศาสตราจารย์ ดร.ชิตพงศ์ เวชโรสงค์ , 90 หน้า

คำสำคัญ : การพยากรณ์/โครงข่ายประสาทเทียม/หน่วยความจำระยะสั้นยาว/ค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติ/ค่าเฉลี่ยเคลื่อนที่อย่างง่าย/การเลื่อนหน้าต่าง

ทุกวันนี้ผู้คนต่างใช้งานอินเทอร์เน็ตกันมากขึ้นและมีแนวโน้มว่าจะเพิ่มมากขึ้นเรื่อยๆ เนื่องจากอินเทอร์เน็ตมีส่วนสำคัญต่อกิจวัตรประจำวันของเราไม่ว่าจะเป็นการทำงาน การประชุม การทำธุรกรรมต่าง ๆ การสตรีมหรือไลฟ์สดที่เป็นการใช้งานแบบเรียลไทม์ เป็นต้น ซึ่งทั้งหมดที่กล่าวมาจำเป็นที่จะต้องอาศัยอินเทอร์เน็ตทั้งสิ้น เมื่อมีจำนวนผู้ใช้งานอินเทอร์เน็ตที่เพิ่มมากขึ้นก็จะส่งผลให้มีจำนวนปริมาณการจราจรบนระบบเครือข่ายที่เพิ่มมากขึ้นตามไปด้วย แต่เนื่องจากทรัพยากรบนระบบเครือข่ายที่มีอยู่ค่อนข้างจำกัดจึงทำให้ไม่สามารถรองรับต่อจำนวนผู้ใช้งานบางรายได้ ทำให้ผู้ใช้บริการได้รับประสบการณ์ในการใช้งานที่ไม่ค่อยดี เช่น อินเทอร์เน็ตมีความล่าช้ากว่าแพ็กเก็ตที่ได้รับ ระบบเครือข่ายล่มหรืออินเทอร์เน็ตขัดข้อง เป็นต้น สิ่งเหล่านี้อาจจะส่งผลกระทบต่อผู้ให้บริการเครือข่ายไม่ว่าจะเป็นด้านความน่าเชื่อถือหรือการบริการ ทำให้ผู้ให้บริการเครือข่ายเกิดความกังวลเกี่ยวกับการให้บริการต่อลูกค้า ฉะนั้นจำเป็นที่จะต้องหาวิธีการแก้ไขปัญหาดังกล่าว ดังนั้นในงานวิจัยนี้จึงได้เสนอวิธีการพยากรณ์ข้อมูลปริมาณการจราจรบนระบบเครือข่ายล่วงหน้า เพื่อที่จะสามารถทราบปริมาณการจราจรบนระบบเครือข่ายเพิ่มขึ้นหรือลดลง ซึ่งจะทำให้สามารถจัดสรรทรัพยากรให้มีประสิทธิภาพและเพียงพอต่อความต้องการของผู้ใช้งาน ณ ขณะเวลานั้นได้ โดยในงานวิจัยนี้จะพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่องเปรียบเทียบกับการพยากรณ์ด้วยวิธีการทางสถิติและจะนำวิธีการเลื่อนหน้าต่างเข้ามาใช้งานควบคู่ไปด้วย จากนั้นจะนำผลที่ได้จากการพยากรณ์มาคำนวณเพื่อหาแบบจำลองที่มีความแม่นยำและมีค่าความผิดพลาดต่ำด้วยเครื่องมือวัดประสิทธิภาพของแบบจำลองเพื่อให้ได้มาซึ่งแบบจำลองที่น่าเชื่อถือและมีประสิทธิภาพ

สาขาวิชาวิศวกรรมโทรคมนาคม
ปีการศึกษา 2564

ลายมือชื่อนักศึกษา ศุภกานต์ ประจํา
ลายมือชื่ออาจารย์ที่ปรึกษา ชิตพงศ์ เวชโรสงค์

SUPAKARN PRAJAM : FORECASTING NETWORK DATA TRAFFIC USING MACHINE
LEARNING METHODS AND SLIDING WINDOW: ASST. PROF. CHITAPONG
WECHTAISONG, Ph.D., 90 PP.

Keyword : Forecasting/ Artificial Neural Network/ Long Short-Term Memory/
Autoregressive Integrated Moving Average/ Simple Moving Average/ Sliding
Window

Today, people are increasingly using the internet, and this trend is expected to continue. This is because the internet is an integral part of our daily lives, whether it's for work, meetings, transactions, streaming, or real-time usage. From all of the above activities need to rely on the internet. As the number of people using the internet grows, so does the amount of network traffic. Some users, however, cannot be accommodated due to the restricted number of network resources available. As a result, the user receives a poor user experience, such as the internet being slower than the received packet, the network collapsing, or the internet being disrupted, and so on. These could have a significant impact on the network operator's reliability and service, which makes customers concerned about network operators' ability to provide service. Thereupon as a result, a solution to the problem is necessary. Therefore, in this research, a method for forecasting the traffic volume on the network in advance is proposed. To figure out whether network traffic is increasing or decreasing. This will allow for the efficient and sufficient allocation of resources to satisfy the needs of users at the time. In this research, forecasting using machine learning methods is compared to forecasting using statistics and will bring the method of the sliding window into use together. Then, the outcome of the forecast will then be calculated to discover accurate, low-error models using model performance measurement tools, resulting in an efficient model.

School of Telecommunication Engineering
Academic Year 2021

Student's Signature ศุภกานต์ ประจักษ์
Advisor's Signature ชิตพงษ์ เวชไตสง

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงด้วยดี เนื่องจากได้รับความช่วยเหลืออย่างยิ่ง ทั้งด้านวิชาการและด้านดำเนินงานวิจัย ทั้งคอยให้คำปรึกษาและเป็นกำลังใจจากบุคคลและกลุ่มบุคคลต่าง ๆ ได้แก่

ขอกราบขอบพระคุณผู้ช่วยศาสตราจารย์ ดร.ชิตพงศ์ เวชไธสงค์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ให้คำแนะนำ ให้ความรู้ ให้โอกาสทางการศึกษา ให้คำปรึกษาช่วยแก้ปัญหาและคอยให้กำลังใจด้วยความเอาใจใส่เป็นอย่างดีแก่ผู้วิจัยมาโดยตลอด รวมทั้งช่วยตรวจทาน และแก้ไขวิทยานิพนธ์เล่มนี้จนเสร็จสมบูรณ์

ขอกราบขอบพระคุณคณาจารย์สาขาวิชาวิศวกรรมโทรคมนาคม สำนักวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารีทุกท่าน ที่ได้ถ่ายทอดความรู้ทางวิชาการ ให้คำแนะนำในการดำเนินงาน แนวทางแก้ไขและให้โอกาสในการศึกษา

ขอขอบคุณมหาวิทยาลัยเทคโนโลยีสุรนารีที่ให้การสนับสนุนทุนการศึกษากิตติบัณฑิต

ขอขอบคุณเพื่อน และพี่น้องบัณฑิตศึกษาทุกท่าน ที่คอยให้ความช่วยเหลือ ให้คำปรึกษาด้านวิชาการและคอยให้กำลังใจในการทำวิทยานิพนธ์และขอบคุณคุณวีรินทร์ อาจหาญ เลขานุการสาขาวิชาวิศวกรรมโทรคมนาคมที่ให้คำปรึกษาและจัดการด้านเอกสารตลอดระยะเวลาการศึกษา

สุดท้ายนี้ ผู้วิจัยขอขอบคุณอาจารย์ผู้สอนทุกท่านที่ประสิทธิประสาทความรู้ด้านต่าง ๆ ทั้งในอดีตและปัจจุบัน และขอกราบขอบพระคุณ บิดา มารดา พี่ชาย รวมถึงญาติของผู้วิจัยทุกท่านที่ให้การอบรมเลี้ยงดู คอยเป็นกำลังใจและให้การสนับสนุนทางการศึกษาเป็นอย่างดีมาโดยตลอด ทำให้ผู้วิจัยประสบความสำเร็จในชีวิตเรื่อยมา สำหรับคุณงามความดีอันใดที่เกิดจากวิทยานิพนธ์เล่มนี้ ผู้วิจัยขอมอบให้กับบิดา มารดา พี่ชาย รวมถึงญาติซึ่งเป็นที่รักและเคารพยิ่งตลอดจนครูอาจารย์ผู้สอนที่เคารพทุกท่านที่ได้ถ่ายทอดประสบการณ์ที่ดีให้แก่ผู้วิจัยทั้งในอดีตและปัจจุบัน จนสำเร็จการศึกษาไปได้ด้วยดี

ศุภกานต์ ประจักษ์

สารบัญ

หน้า

บทคัดย่อ (ภาษาไทย).....	ก
บทคัดย่อ (ภาษาอังกฤษ).....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ช
สารบัญรูป.....	ฉ

บทที่

1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์ของการวิจัย	2
1.3 สมมติฐานของการวิจัย	3
1.4 ขอบเขตของการวิจัย	3
1.5 วิธีดำเนินการวิจัย	3
1.6 ประโยชน์ที่คาดว่าจะได้รับ	4
1.7 ส่วนประกอบของวิทยานิพนธ์	4
2 ปรัชญาวรรณกรรมและทฤษฎีที่เกี่ยวข้อง	5
2.1 กล่าวนำ	5
2.2 การเรียนรู้ของเครื่อง	5
2.3 ประเภทของการเรียนรู้ของเครื่อง.....	6
2.3.1 การเรียนรู้แบบมีการสอน	7
2.3.2 การเรียนรู้แบบไม่มีการสอน	9
2.3.3 การเรียนรู้แบบเสริมแรง	10
2.4 การเรียนรู้เชิงลึก	11
2.4.1 โครงข่ายประสาทเทียม.....	12
2.4.2 โครงสร้างของโครงข่ายประสาทเทียม.....	15

สารบัญ (ต่อ)

	หน้า
2.4.3 หน่วยความจำระยะสั้นยาว.....	16
2.5 ฟังก์ชันการตัดสินใจ	24
2.6 การพยากรณ์ด้วยวิธีการทางสถิติ.....	27
2.6.1 ค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติ	27
2.6.2 ค่าเฉลี่ยเคลื่อนที่อย่างง่าย.....	30
2.7 เครื่องมือวัดประสิทธิภาพแบบจำลอง.....	30
2.7.1 ค่าเฉลี่ยของค่าผิดพลาดสัมบูรณ์ (MAE).....	30
2.7.2 ค่าเฉลี่ยความผิดพลาดร้อยละสัมบูรณ์ (MAPE)	31
2.7.3 ค่าเฉลี่ยของค่าผิดพลาดกำลังสอง (Mean Squared Error: MSE).....	31
2.7.4 ค่ารากที่สองของความคลาดเคลื่อนกำลังสองเฉลี่ย (RMSE).....	31
2.7.5 ค่าสัมประสิทธิ์ R-Squared (R^2)	31
2.8 ภาษาโปรแกรมมิ่งไพธอน (Python).....	32
2.9 ปรัชญาวิศวกรรมและงานวิจัยที่เกี่ยวข้อง	32
2.10 กล่าวสรุป	37
3 วิธีดำเนินการวิจัย.....	38
3.1 กล่าวนำ	38
3.2 ชุดข้อมูล	38
3.3 วิธีดำเนินการทดลอง	42
3.3.1 การพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่อง	42
3.3.2 การพยากรณ์ด้วยค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติ	49
3.3.3 ค่าเฉลี่ยเคลื่อนที่อย่างง่าย.....	52
3.4 กล่าวสรุป.....	53
4 ผลการวิจัยและอภิปรายผล	54
4.1 กล่าวนำ	54
4.2 ผลการทดลองของปริมาณการจราจรข้อมูลใน 11 เมืองของยุโรป	54

สารบัญ (ต่อ)

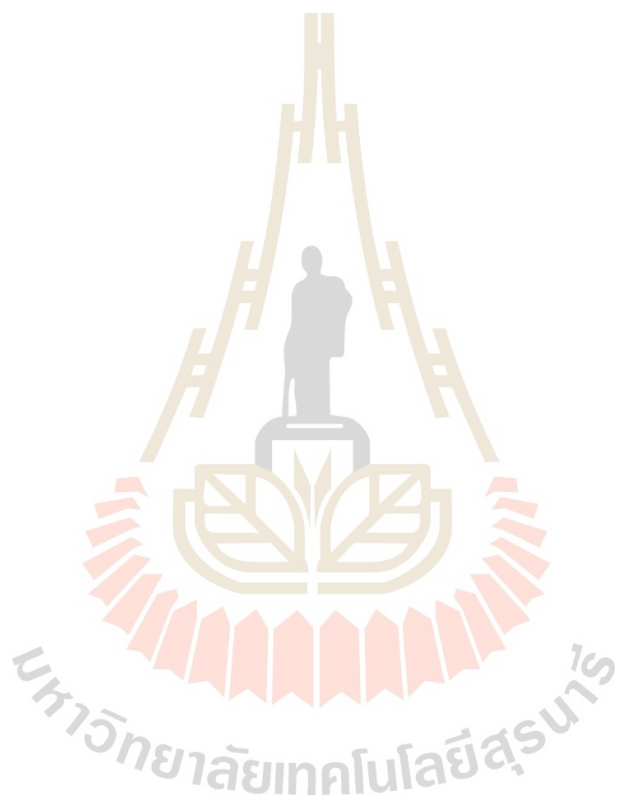
	หน้า
4.2.1 ผลการทดลองของข้อมูล 5 นาที.....	54
4.2.2 ผลการทดลองของข้อมูล 30 นาที.....	55
4.3 ผลการทดลองข้อมูลปริมาณการจราจรเฉลี่ยขาเข้าภายในระบบเครือข่าย มหาวิทยาลัย	57
4.3.1 ผลการทดลองของข้อมูล 5 นาที.....	57
4.3.2 ผลการทดลองของข้อมูล 30 นาที.....	58
4.4 ผลการทดลองข้อมูลปริมาณการจราจรเฉลี่ยขาออกภายในระบบเครือข่าย มหาวิทยาลัย	60
4.4.1 ผลการทดลองของข้อมูล 5 นาที.....	60
4.4.2 ผลการทดลองของข้อมูล 30 นาที.....	61
4.5 อภิปรายผลการทดลอง.....	63
4.6 กล่าวสรุป.....	64
5 สรุปการวิจัยและข้อเสนอแนะ	65
5.1 สรุปเนื้อหาของวิทยานิพนธ์.....	65
5.2 ข้อเสนอแนะและแนวทางการพัฒนา.....	66
รายการอ้างอิง	67
ภาคผนวก ก.....	70
บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างการศึกษา	70
รายชื่อบทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างศึกษา	71
ประวัติผู้เขียน	90

สารบัญตาราง

ตารางที่	หน้า
2.1 ตัวอย่างการจัดชุดข้อมูลก่อนฝึกฝนสำหรับการเรียนรู้แบบจำแนกประเภท.....	8
2.2 ตัวอย่างการจัดชุดข้อมูลก่อนฝึกฝนสำหรับการเรียนรู้แบบถดถอย.....	9
2.3 ตัวอย่างการจัดชุดข้อมูลก่อนฝึกฝนสำหรับการเรียนรู้แบบไม่มีการสอน.....	10
2.4 งานวิจัยที่เกี่ยวข้อง.....	33
3.1 พารามิเตอร์ที่กำหนดสำหรับการฝึกฝนชุดข้อมูล 5 นาทีและ 30 นาทีของปริมาณการใช้..... งานเครือข่ายจาก 11 เมืองในยุโรป.....	46
3.2 พารามิเตอร์ที่กำหนดสำหรับการฝึกฝนชุดข้อมูล 5 นาทีและ 30 นาทีของปริมาณการใช้งาน เครือข่ายมหาวิทยาลัยของข้อมูลเฉลี่ยขาเข้า.....	48
3.3 พารามิเตอร์ที่กำหนดสำหรับการฝึกฝนชุดข้อมูล 5 นาทีและ 30 นาทีของปริมาณการใช้งาน เครือข่ายมหาวิทยาลัยของข้อมูลเฉลี่ยขาออก.....	48
3.4 พารามิเตอร์ของแบบจำลองปริมาณการใช้งานบนระบบเครือข่ายของ 11 เมือง ในยุโรป.....	51
3.5 พารามิเตอร์ของแบบจำลองปริมาณการใช้งานบนระบบเครือข่ายภายใน มหาวิทยาลัย.....	52
4.1 ประสิทธิภาพของแบบจำลองในการพยากรณ์ระบบเครือข่าย 5 นาทีของ 11 เมือง ในยุโรป.....	56
4.2 ประสิทธิภาพของแบบจำลองในการพยากรณ์ระบบเครือข่าย 30 นาทีของ 11 เมือง ในยุโรป.....	56
4.3 ประสิทธิภาพของแบบจำลองในการพยากรณ์ระบบเครือข่ายเฉลี่ยขาเข้า 5 นาทีภายใน มหาวิทยาลัย.....	58
4.4 ประสิทธิภาพของแบบจำลองในการพยากรณ์ระบบเครือข่ายเฉลี่ยขาเข้า 30 นาทีภายใน มหาวิทยาลัย.....	59
4.5 ประสิทธิภาพของแบบจำลองในการพยากรณ์ระบบเครือข่ายเฉลี่ยขาออก 5 นาทีภายใน มหาวิทยาลัย.....	61

สารบัญตาราง (ต่อ)

ตารางที่	หน้า
4.6 ประสิทธิภาพของแบบจำลองในการพยากรณ์ระบบเครือข่ายเฉลี่ยขาออก 30 นาทีภายในมหาวิทยาลัย.....	62



สารบัญรูป

รูปที่	หน้า
2.1	หลักการทํางานของโปรแกรมแบบดั้งเดิม..... 5
2.2	หลักการทํางานและวิธีนําไปใช้งานของการเรียนรู้ของเครื่อง..... 6
2.3	ประเภทของการเรียนรู้ของเครื่อง 7
2.4	การเดินหมากล้อมของ AlphaGo ที่ชนะมนุษย์ด้วยการเรียนรู้แบบเสริมแรง..... 11
2.5	ลักษณะการทํางานระหว่างการเรียนรู้ของเครื่องและการเรียนรู้เชิงลึก..... 12
2.6	โครงสร้างของเซลล์ประสาท 13
2.7	โครงสร้างของเพอเซปตรอน 15
2.8	ลักษณะโครงสร้างของโครงข่ายประสาทเทียม 16
2.9	สถาปัตยกรรมโครงข่ายประสาทเทียมแบบกําเรียบ 17
2.10	โครงสร้างภายในโครงข่ายประสาทแบบกําเรียบ..... 18
2.11	แสดงลักษณะโครงสร้างภายในของเกต (Gate)..... 19
2.12	ลักษณะโครงสร้างภายในของหน่วยความจําระยะสั้นยาว 20
2.13	ลักษณะการทํางานของชั้นเกตลิ้ม 21
2.14	ลักษณะการทํางานของชั้นเกตอินพุตและเกตมอดูเลตอินพุต 22
2.15	ลักษณะการทํางานของการอัปเดตสถานะเซลล์ 23
2.16	ลักษณะการทํางานของชั้นเกตเอาต์พุต 24
2.17	ลักษณะกราฟของฟังก์ชันการตัดสินใจแบบเชิงเส้น..... 25
2.18	ลักษณะกราฟของฟังก์ชันการตัดสินใจแบบซิกมอยด์ 25
2.19	ลักษณะกราฟของฟังก์ชันการตัดสินใจแบบไฮเพอร์โบลิกแทนเจนต์..... 26
2.20	ลักษณะกราฟของฟังก์ชันการตัดสินใจแบบ ReLU 27
2.21	ลักษณะกราฟของข้อมูลแบบคงที่และข้อมูลที่ไม่คงที่ (K. Sultan et al., 2018) 29
3.1	ชุดข้อมูลดิบของการจําจบบนระบบเครือข่ายในยุโรป 5 นาที 39
3.2	ชุดข้อมูลการจําจบบนระบบเครือข่ายในยุโรป 5 นาทีที่ถูกจัดเรียงเรียบร้อยแล้ว 39

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.3 ชุดข้อมูลการจราจรบนระบบเครือข่ายในยุโรป 30 นาทีที่ถูกจัดเรียงเรียบร้อยแล้ว.....	40
3.4 ชุดข้อมูลดิบของการทำงานของเครือข่ายภายในมหาวิทยาลัย.....	41
3.5 ข้อมูลการทำงานของเครือข่ายมหาวิทยาลัย 5 นาทีที่ได้รับการจัดเรียงแล้ว.....	41
3.6 ข้อมูลการทำงานของเครือข่ายมหาวิทยาลัย 30 นาทีที่ได้รับการจัดเรียงแล้ว.....	41
3.7 ผลงานแสดงขั้นตอนสำหรับการพยากรณ์ด้วยแบบจำลอง ANN และ LSTM.....	44
3.8 ผลงานแสดงขั้นตอนสำหรับการพยากรณ์ด้วยแบบจำลอง ARIMA.....	50
3.9 แผนภาพแสดงขั้นตอนสำหรับการพยากรณ์ด้วยแบบจำลอง SMA.....	52
4.1 กราฟแสดงการเปรียบเทียบการพยากรณ์ระบบเครือข่าย 5 นาทีของ 11 เมืองในยุโรป.....	56
4.2 กราฟแสดงการเปรียบเทียบการพยากรณ์ระบบเครือข่าย 30 นาทีของ 11 เมืองในยุโรป.....	57
4.3 กราฟแสดงการเปรียบเทียบการพยากรณ์ระบบเครือข่ายเฉลี่ยขาเข้า 5 นาทีภายใน มหาวิทยาลัย.....	59
4.4 กราฟแสดงการเปรียบเทียบการพยากรณ์ระบบเครือข่ายเฉลี่ยขาเข้า 30 นาทีภายใน มหาวิทยาลัย.....	60
4.5 กราฟแสดงการเปรียบเทียบการพยากรณ์ระบบเครือข่ายเฉลี่ยขาออก 5 นาทีภายใน มหาวิทยาลัย.....	62
4.6 กราฟแสดงการเปรียบเทียบการพยากรณ์ระบบเครือข่ายเฉลี่ยขาออก 30 นาทีภายใน มหาวิทยาลัย.....	63

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ชีวิตและความเป็นอยู่ของสังคมในยุคปัจจุบันนี้เป็นยุคที่เรียกว่ายุคดิจิทัล ส่วนใหญ่มักจะขับเคลื่อนด้วยเทคโนโลยี ทำให้การใช้งานอินเทอร์เน็ตเข้ามามีบทบาทที่สำคัญเป็นอย่างมากกับชีวิตประจำวัน ไม่ว่าจะเป็นการประชุมออนไลน์ การสตรีมมิ่ง โดยถ่ายทอดสดผ่านเฟซบุ๊กหรือการแข่งขันเกมส์ การทำธุรกรรมทางโลกออนไลน์ และอื่น ๆ อีกมากมาย ดังนั้นเมื่อทุกคนต่างก็ใช้งานอินเทอร์เน็ตจึงส่งผลให้มีปริมาณการจราจรบนระบบเครือข่ายเพิ่มมากขึ้นเรื่อย ๆ อย่างไม่สิ้นสุด หากจะมองให้เห็นภาพได้ง่ายขึ้นอาจเปรียบเทียบการใช้งานอินเทอร์เน็ตดังรถยนต์ และสายส่งสัญญาณเปรียบเสมือนถนนเส้นหนึ่ง ในทุก ๆ วันมีการเดินทางด้วยรถยนต์บนถนนเส้นนั้นเป็นประจำ การจราจรคล่องตัว แต่ถ้าหากวันหนึ่งถนนเส้นนี้มีรถยนต์ที่เข้ามาใช้งานมากขึ้นจนทำให้ถนนไม่สามารถรองรับต่อจำนวนรถยนต์ได้ ส่งผลให้รถยนต์ล้นถนนออกมา เกิดการจราจรติดขัดและเป็นปัญหาคอขวด ซึ่งก็เหมือนกับกรณีจำนวนผู้ใช้งานอินเทอร์เน็ตที่เพิ่มมากขึ้น ณ เวลาและสถานที่เดียวกัน ส่งผลให้ระบบเครือข่ายไม่สามารถรองรับต่อจำนวนผู้ใช้งานที่เพิ่มขึ้น ทำให้ประสบการณ์ในการใช้งานอินเทอร์เน็ตมีความล่าช้าหรือเครือข่ายล่ม ซึ่งผู้ให้บริการเครือข่ายจึงต้องดำเนินการหาวิธีการจัดสรรระบบเครือข่ายให้เพียงพอต่อลูกค้าหรือผู้ใช้งานได้ทันทั่วทั้ง แต่เนื่องจากทรัพยากรที่มีอยู่นั้นค่อนข้างจำกัด ฉะนั้นจึงจำเป็นต้องคาดการณ์ปริมาณการจราจรบนระบบเครือข่ายให้เพียงพอต่อผู้ใช้บริการในอนาคตและจะส่งผลให้ผู้ให้บริการเครือข่ายมีความน่าเชื่อถือและไว้วางใจมากยิ่งขึ้น

จากปัญหาดังกล่าวได้มีการศึกษาและดำเนินการวิจัยเกี่ยวกับการพยากรณ์การจราจรบนระบบเครือข่าย โดยลักษณะของข้อมูลการจราจรจะมีรูปแบบที่ไม่เป็นเชิงเส้น เช่น ข้อมูลปริมาณการใช้งานเครือข่ายมือถือ ข้อมูลปริมาณการใช้งานเครือข่ายอินเทอร์เน็ตของมหาวิทยาลัยหรือจากผู้ให้บริการระบบเครือข่าย ซึ่งแต่ละข้อมูลเหล่านี้อาจจะแบ่งเป็นรายนาที รายชั่วโมง หรือรายวัน ฉะนั้นจึงทำให้นักวิจัยหลายท่านได้พยากรณ์ข้อมูลไว้ล่วงหน้าโดยใช้อัลกอริทึมที่แตกต่างกัน และจะวัดความสามารถของแต่ละแบบจำลองเมื่อพยากรณ์เสร็จสิ้น ซึ่งอัลกอริทึมที่นำมาทดสอบส่วนใหญ่จะเป็นการเปรียบเทียบการพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่องกับวิธีการทางสถิติ และนำผลการพยากรณ์ที่ได้มาทดสอบกับเครื่องมือวัดประสิทธิภาพของแบบจำลอง เช่น ค่าเฉลี่ยของค่าผิดพลาดสัมบูรณ์ (Mean Absolute Error: MAE), ค่ารากที่สองของความคลาดเคลื่อนกำลังสองเฉลี่ย

(Root Mean Square Error: RMSE), ค่าสัมประสิทธิ์ R-Squared (R^2), ค่าเฉลี่ยความผิดพลาดร้อยละสัมบูรณ์ (Mean Absolute Percentage Error: MAPE) เป็นต้น โดยข้อสรุปที่ได้จากการทดลองส่วนใหญ่พบว่าพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่องมีความแม่นยำมากกว่าวิธีการทางสถิติ แต่จากงานวิจัยที่ได้ศึกษามานี้ส่วนใหญ่จะเป็นการพยากรณ์ข้อมูลที่ไม่ใช่แบบเรียลไทม์ อาจจะทำให้ไม่สามารถตอบสนองต่อความต้องการในการใช้งานของผู้ใช้งาน ณ ขณะเวลานั้นได้

เนื่องด้วยงานวิจัยนี้เป็นการพยากรณ์ข้อมูลปริมาณการจราจรบนระบบเครือข่าย เพื่อนำไปใช้งานหรือเป็นแบบจำลองการเรียนรู้ให้กับผู้ให้บริการเครือข่าย จึงได้ทำการสอบถามถึงความต้องการที่จะได้รับมาพอสังเขป ข้อมูลที่ได้รับจากผู้ให้บริการระบบเครือข่ายคือต้องการพยากรณ์ปริมาณการใช้งานกราฟฟิคล่วงหน้าเมื่อมีผู้ใช้งานเป็นจำนวนมาก ณ สถานที่และเวลาเดียวกัน โดยต้องการรู้ล่วงหน้าอย่างน้อยประมาณ 30 นาทีเพื่อที่จะสามารถปรับพารามิเตอร์ต่าง ๆ ให้ทันต่อความต้องการของผู้ใช้งานในอนาคต และในการพยากรณ์กราฟฟิคจำเป็นต้องดูปัจจัยภายนอกต่าง ๆ ที่นอกเหนือจากข้อมูลปริมาณการใช้งานกราฟฟิคก่อนหน้า เช่น จำนวนผู้ใช้งานก่อนหน้านั้นมาตัดสินใจร่วมด้วยขณะพยากรณ์

ดังนั้นงานวิจัยนี้จึงต้องการพยากรณ์ข้อมูลปริมาณการจราจรบนระบบเครือข่ายแบบเรียลไทม์ โดยจะใช้วิธีการเลื่อนหน้าต่าง (Sliding Window) เข้ามาช่วยในการพยากรณ์ เพื่อที่จะสามารถรองรับต่อการใช้งานของผู้ใช้งานได้ทันทั่วทั้ง สำหรับวิธีที่จะใช้ในการพยากรณ์ประกอบไปด้วย วิธีการเรียนรู้ของเครื่องคือโครงข่ายประสาทเทียม (Artificial Neural Network: ANN) และหน่วยความจำระยะสั้นยาว (Long Short-Term Memory: LSTM) กับวิธีการทางสถิติคือค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติ (Autoregressive Integrated Moving Average: ARIMA) และค่าเฉลี่ยเคลื่อนที่อย่างง่าย (Simple Moving Average: SMA) จากนั้นจะดำเนินการเปรียบเทียบประสิทธิภาพความสามารถในการพยากรณ์ของแต่ละแบบจำลอง ซึ่งจะทำการทดสอบกับเครื่องมือวัดประสิทธิภาพในลำดับถัดไปเพื่อให้ได้มาซึ่งวิธีการและแบบจำลองที่ดี

1.2 วัตถุประสงค์ของการวิจัย

1.2.1 เพื่อให้ได้มาซึ่งวิธีการและแบบจำลองที่ใช้ในการพยากรณ์ปริมาณการจราจรบนระบบเครือข่ายที่มีความถูกต้องและแม่นยำ

1.2.2 เพื่อต้องการพยากรณ์การจราจรบนระบบเครือข่ายแบบเรียลไทม์ เนื่องจากความต้องการที่จะรองรับต่อการแก้ไขและจัดสรรทรัพยากรให้แก่ผู้ใช้งานได้ทันที

1.3 สมมติฐานของการวิจัย

1.3.1 การพยากรณ์ด้วยการใช้วิธีการเรียนรู้ของเครื่องมีประสิทธิภาพในการพยากรณ์ได้ดีกว่าวิธีการเรียนรู้ทางสถิติ

1.3.2 การนำวิธีการเลื่อนหน้าต่างเข้ามาใช้ควบคู่กับการพยากรณ์ สามารถช่วยให้ทราบถึงข้อมูลปริมาณการจราจรบนระบบเครือข่ายได้ล่วงหน้า เพื่อที่จะได้วางแผนและแก้ไขระบบเครือข่ายให้รองรับต่อการใช้งานแบบเรียลไทม์

1.4 ขอบเขตของการวิจัย

1.4.1 จำลองแบบจำลองในการพยากรณ์ข้อมูลปริมาณการจราจรบนระบบเครือข่ายด้วยโปรแกรม Google Colab และโปรแกรม Excel สำหรับภาษาโปรแกรมมิ่งที่ใช้เขียนคือภาษาไพธอน (Python)

1.4.2 ข้อมูลปริมาณการจราจรบนระบบเครือข่ายแบ่งเป็น 2 ข้อมูลคือ ข้อมูลที่ได้รับการเผยแพร่แบบสาธารณะมาจากการวิจัยของต่างประเทศและข้อมูลการใช้งานอินเทอร์เน็ตที่ได้รับความอนุเคราะห์จากศูนย์คอมพิวเตอร์ของมหาวิทยาลัยเทคโนโลยีสุรนารี

1.5 วิธีดำเนินการวิจัย

1.5.1 แนวทางการดำเนินงาน

1. สำรวจบริษัทที่สนับสนุนกรรมและงานวิจัยที่เกี่ยวข้องกับวิทยานิพนธ์
2. ศึกษาวิธีการพยากรณ์ข้อมูลการจราจรบนระบบเครือข่ายด้วยการเรียนรู้ของเครื่องและวิธีการทางสถิติ
3. ออกแบบและสร้างแบบจำลองเพื่อดำเนินการพยากรณ์
4. วัดประสิทธิภาพในการพยากรณ์ของแต่ละแบบจำลอง

1.5.2 สถานที่ทำการวิจัย

ห้องวิจัยและปฏิบัติการสื่อสารโทรคมนาคม อาคารเครื่องมือ 11 มหาวิทยาลัยเทคโนโลยีสุรนารี 111 ถนนมหาวิทยาลัย ต.สุรนารี อ.เมือง จ.นครราชสีมา 30000

1.5.3 เครื่องมือที่ใช้ในการวิจัย

1. เครื่องคอมพิวเตอร์ส่วนบุคคล
2. โปรแกรม Google Colab
3. โปรแกรม Excel

1.6 ประโยชน์ที่คาดว่าจะได้รับ

1.6.1 ได้แบบจำลองที่มีประสิทธิภาพ เพื่อสามารถนำไปใช้พยากรณ์ข้อมูลเครือข่ายอินเทอร์เน็ตแบบเรียลไทม์

1.6.2 สามารถช่วยผู้ให้บริการเครือข่ายดำเนินการจัดสรรทรัพยากรและวางแผนระบบเครือข่ายให้เพียงพอต่อผู้ใช้งานในอนาคตได้

1.7 ส่วนประกอบของวิทยานิพนธ์

สำหรับเนื้อหาของวิทยานิพนธ์ฉบับนี้ นำเสนอการศึกษาค้นคว้า รวบรวมข้อมูล การสร้างแบบจำลอง วิเคราะห์และสรุปผล โดยประกอบไปด้วยเนื้อหาทั้ง 5 ดังนี้

บทที่ 1 เป็นการกล่าวถึงความเป็นมาและความสำคัญของปัญหา วัตถุประสงค์ของการวิจัย สมมติฐานของการวิจัย ของเขตการวิจัย วิธีดำเนินการวิจัย และประโยชน์ที่คาดว่าจะได้รับ

บทที่ 2 กล่าวถึงปริทัศน์วรรณกรรมและทฤษฎีที่เกี่ยวข้องกับงานวิจัยในการพยากรณ์ข้อมูล การจราจรบนระบบเครือข่ายด้วยวิธีต่าง ๆ และเครื่องมือในการวัดประสิทธิภาพของแบบจำลอง

บทที่ 3 กล่าวถึงวิธีการสร้างและการทดสอบแบบจำลองที่ใช้สำหรับการพยากรณ์ข้อมูล

บทที่ 4 กล่าวถึงการวิเคราะห์ข้อมูลและอภิปรายผลประสิทธิภาพที่ได้จากการเปรียบเทียบของแต่ละแบบจำลองทั้งวิธีการเรียนรู้ของเครื่องและวิธีการทางสถิติ เพื่อแสดงให้เห็นถึงความแม่นยำในการพยากรณ์

บทที่ 5 กล่าวถึงการสรุปผลของการวิจัย ข้อเสนอแนะและแนวทางการพัฒนาต่อในอนาคต

บทที่ 2

ปริทัศน์วรรณกรรมและทฤษฎีที่เกี่ยวข้อง

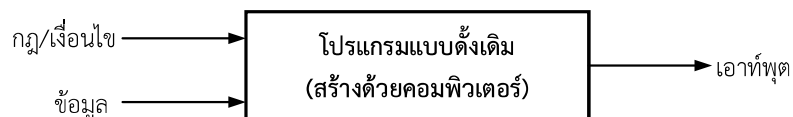
2.1 กล่าวนำ

ในบทนี้จะกล่าวถึงทฤษฎีและงานวิจัยที่เกี่ยวข้องกับวิทยานิพนธ์ฉบับนี้ ซึ่งการพยากรณ์ข้อมูลปริมาณจราจรบนระบบเครือข่ายถือเป็นสิ่งสำคัญอย่างมากต่อผู้ให้บริการระบบเครือข่าย เนื่องจากปัญหาที่ก่อนหน้านี้ทางผู้ให้บริการไม่สามารถวางแผนในการปรับระบบเครือข่ายให้รองรับต่อจำนวนผู้ใช้งานที่เพิ่มมากขึ้นได้ทันท่วงที จึงส่งผลให้ผู้ใช้งานได้รับประสบการณ์ในการใช้งานที่ไม่มีประสิทธิภาพ ส่งผลให้ความน่าเชื่อถือและความประทับใจในการให้บริการของผู้ให้บริการลดลง ดังนั้นงานวิจัยนี้จึงสนใจศึกษาวิธีการพยากรณ์ปริมาณการจราจรบนระบบเครือข่ายที่จะเกิดขึ้นในอนาคตเพื่อให้สามารถรองรับต่อการใช้งานเมื่อมีจำนวนผู้ใช้งานที่เพิ่มมากขึ้น โดยจะเปรียบเทียบการพยากรณ์ระหว่างวิธีการเรียนรู้ของเครื่องกับวิธีการทางสถิติ วัดความสามารถในการพยากรณ์และความแม่นยำด้วยเครื่องมือวัดประสิทธิภาพแบบจำลอง โดยทฤษฎีและงานวิจัยที่เกี่ยวข้องจะกล่าวในลำดับถัดไป

2.2 การเรียนรู้ของเครื่อง

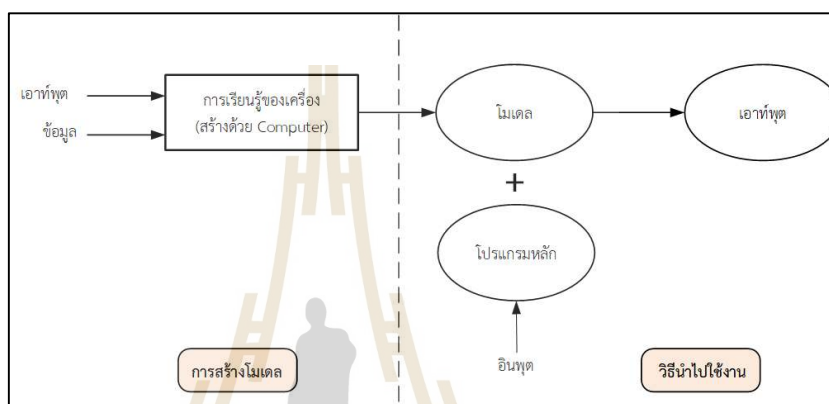
การเรียนรู้ของเครื่อง (Machine Learning: ML) เป็นการป้อนข้อมูลหรือชุดตัวอย่างเข้าไปในระบบคอมพิวเตอร์ เพื่อให้คอมพิวเตอร์ได้เกิดการเรียนรู้หรือประสบการณ์จากข้อมูลชุดนั้นโดยอาศัยวิธีการทางคณิตศาสตร์และสถิติในการสร้างแบบจำลองขึ้นมา และเมื่อคอมพิวเตอร์เกิดการเรียนรู้ก็จะนำความรู้ที่ได้มาใช้งาน เช่น การพยากรณ์ การตัดสินใจ หรือการวิเคราะห์ผล

หลักการในการสร้างการเรียนรู้ของเครื่องจะแตกต่างจากการเขียนโปรแกรมแบบดั้งเดิม โดยการทำงานของโปรแกรมแบบดั้งเดิมจะดำเนินการกำหนดกฎหรือเงื่อนไขขึ้นมาที่จะกระทำกับข้อมูลแล้วจึงได้ผลลัพธ์ออกมา ดังแสดงในรูปที่ 2.1



รูปที่ 2.1 หลักการทำงานของโปรแกรมแบบดั้งเดิม

แต่วิธีการเรียนรู้ของเครื่องนั้นอันดับแรกจะนำผลลัพธ์และข้อมูลใส่เข้าไปในคอมพิวเตอร์ให้ทำการเรียนรู้หรือที่เรียกว่า train เพื่อดำเนินการคำนวณและประมวลผลให้ได้มาซึ่งกฎหรือเงื่อนไขของโปรแกรมซึ่งนั่นก็คือแบบจำลอง และเมื่อนำไปใช้งานตัวผู้ใช้งานจะต้องดำเนินการเขียนโปรแกรมหลักเพิ่มขึ้นมาก่อน แล้วจึงใส่ข้อมูลอินพุตเข้าไปที่โปรแกรมหลักแล้วจึงทำการประมวลผลด้วยตัวแบบจำลองเพื่อให้ได้มาซึ่งข้อมูลเอาต์พุต ดังแสดงในรูปที่ 2.2

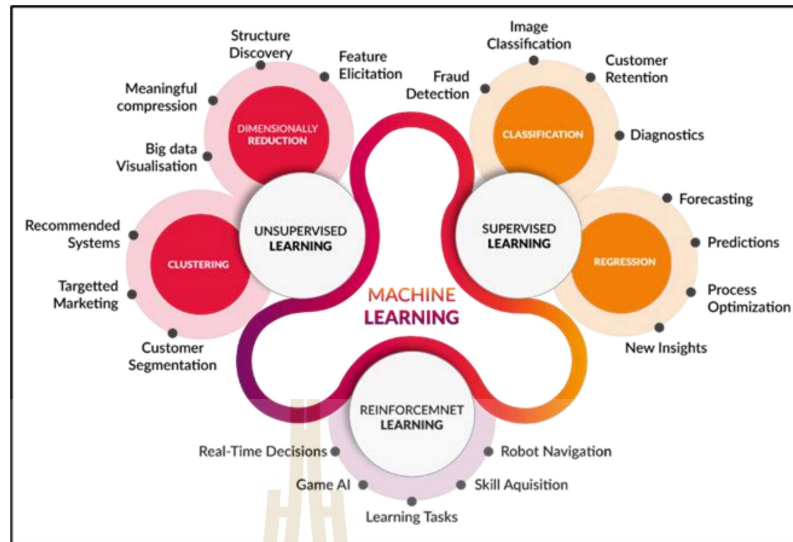


รูปที่ 2.2 หลักการทำงานและวิธีนำไปใช้งานของการเรียนรู้ของเครื่อง

ดังนั้นจำเป็นที่จะต้องเลือกงานที่เหมาะสมกับการเรียนรู้ของเครื่อง เพื่อให้ได้แบบจำลองและผลลัพธ์ที่มีประสิทธิภาพ ซึ่งวิธีการเรียนรู้ของเครื่องจะเหมาะกับงานในลักษณะต่าง ๆ เช่น งานที่มีความซับซ้อนมากจนมนุษย์ไม่สามารถแยกแยะหรือจัดการได้ด้วยตัวเอง งานที่ต้องมีการวิเคราะห์เนื่องจากต้องการหาความสัมพันธ์ของข้อมูลที่มีจำนวนมาก หรือแม้กระทั่งงานที่ไม่สามารถออกแบบวิธีการหรือพยากรณ์ได้ล่วงหน้า เนื่องจากปัจจัยที่แตกต่างไปตามสถานการณ์ เป็นต้น

2.3 ประเภทของการเรียนรู้ของเครื่อง

วัตถุประสงค์หลักของการเรียนรู้ของเครื่องนั้นต้องการให้คอมพิวเตอร์ได้เรียนรู้ชุดข้อมูลอินพุตที่ถูกป้อนเข้ามา แล้วเกิดการเรียนรู้ที่เรียกว่า training และสร้างเป็นแบบจำลองที่มีประสิทธิภาพ ยิ่งตัวแบบจำลองมีการเรียนรู้มากเท่าไรก็จะทำให้แบบจำลองนั้นมีความแม่นยำมากยิ่งขึ้น ซึ่งคล้ายกับมนุษย์ที่ยังมีประสบการณ์ในการใช้ชีวิตมากเท่าไรก็สามารถทำให้คาดเดาได้แม่นยำเท่านั้น เช่น การคาดเดาว่าวันนี้ฝนจะตกหรือไม่ โดยดูจากสภาพอากาศหรือการใช้ชีวิตของสิ่งมีชีวิต ดังนั้นการเรียนรู้ของเครื่องจึงแบ่งประเภทของการเรียนรู้ออกเป็น 3 ประเภท ดังแสดงในภาพที่ 2.3



รูปที่ 2.3 ประเภทของการเรียนรู้ของเครื่อง (<https://www.coghub.com>)

2.3.1 การเรียนรู้แบบมีการสอน

การเรียนรู้แบบมีการสอน (Supervised Learning) เป็นการเรียนรู้แขนงหนึ่งที่อยู่ภายใต้การเรียนรู้ของเครื่องโดยต้องมีการป้อนข้อมูลตัวอย่างและคำตอบเข้าไปให้คอมพิวเตอร์ได้เรียนรู้ (Training Data) เพื่อให้ได้มาซึ่งแบบจำลองที่จะนำมาหาคำตอบของปัญหานั้นได้ด้วยตนเอง หากจะมองให้เห็นภาพได้ง่ายยิ่งขึ้น ก็อาจจะเหมือนกับการสอนเด็กให้แยกแยะระหว่างสุนัขและแมว โดยสอนให้เด็กดูรูปสุนัขและรูปแมวหลาย ๆ รูป และบอกเด็ก ๆ ว่ารูปแบบนี้คือสุนัขและรูปแบบนี้คือแมว เมื่อเด็กเกิดการเรียนรู้มากขึ้นเรื่อย ๆ เวลาที่ไปเจอสุนัขก็จะสามารถแยกแยะได้ว่าเป็นสุนัข และสามารถแยกแยะได้ว่าเป็นแมว ซึ่งการสอนในรูปแบบนี้จะเรียกว่าเป็นการสอนแบบการแยกประเภท (Classification) ดังนั้นในการเรียนรู้แบบมีการสอนจะสามารถแบ่งย่อยออกได้ 2 กลุ่มคือการจำแนกแบบการแยกประเภทและแบบการถดถอย (Regression) ตามลำดับ

1) การเรียนรู้แบบแยกประเภท: เป็นการเรียนรู้แบบคัดแยก จำแนกประเภทออกมาให้เป็นกลุ่มหรือหมวดหมู่ที่ได้กำหนดไว้ล่วงหน้าแล้ว สำหรับการจำแนกประเภทจะอาศัยอัลกอริทึมที่หลากหลายในการจัดประเภทชุดข้อมูลซึ่งจะนำข้อมูลอินพุตมาฝึกฝน เรียนรู้ เพื่อคาดการณ์โอกาสที่จะเกิดขึ้น โดยการเรียนรู้แบบแยกประเภทจะใช้กับงานในลักษณะต่าง ๆ เช่น ระบบคัดแยกสแปมเมลเพื่อแยกว่าอีเมลไหนที่ไม่ควรเข้ามาอยู่ในกล่องจดหมายของผู้ใช้งาน, การแยกแยะระหว่างเพศชายหรือเพศหญิง หรือการคาดการณ์สถานการณ์และจัดหมวดหมู่ออกมาเป็นกลุ่มของลาเบล (Label) หรือกลุ่มของชุดคำตอบ (Class) เป็นต้น โดยลักษณะข้อมูลที่ต้องจัดก่อนนำมาฝึกฝนกับวิธีการเรียนรู้

แบบแยกประเภท เช่น ในการจำแนกประเภทกลุ่มผลลัพธ์จะต้องจัดข้อมูลให้มีคอลัมน์เป้าหมาย (Target) หรือผลลัพธ์ให้เป็นแบบใดแบบหนึ่ง อาทิ ผนตกหรือไม่ตก, ผู้ชายหรือผู้หญิง, ลาเบล 0/1 เป็นต้น ดังแสดงในตารางที่ 2.1 เป็นการจัดข้อมูลที่ตัดสินใจว่าจะเล่นเทนนิสหรือไม่ด้วยการตรวจสอบจากสภาพอากาศ

ตารางที่ 2.1 ตัวอย่างการจัดชุดข้อมูลก่อนฝึกฝนสำหรับการเรียนรู้แบบจำแนกประเภท

แนวโน้ม	อุณหภูมิ	ความชื้น	ลม	เล่น/ไม่เล่น
มีแดดมาก	ร้อน	สูง	ลมกำลังอ่อน	ไม่เล่น
มีแดดมาก	ร้อน	สูง	ลมกำลังแรง	ไม่เล่น
มีเมฆมาก	ร้อน	สูง	ลมกำลังแรง	เล่น
มีฝน	อบอุ่น	สูง	ลมกำลังอ่อน	เล่น
มีฝน	เย็น	ปกติ	ลมกำลังอ่อน	เล่น
มีฝน	เย็น	ปกติ	ลมกำลังแรง	ไม่เล่น
มีเมฆมาก	เย็น	ปกติ	ลมกำลังแรง	เล่น
มีแดดมาก	อบอุ่น	สูง	ลมกำลังอ่อน	ไม่เล่น
มีแดดมาก	เย็น	ปกติ	ลมกำลังอ่อน	เล่น
มีฝน	อบอุ่น	ปกติ	ลมกำลังอ่อน	เล่น

2) การเรียนรู้แบบถดถอย: เป็นการเรียนรู้ที่ให้ผลลัพธ์มาในรูปแบบของค่าตัวเลข ไม่สามารถจำแนกออกมาเป็นกลุ่มได้ โดยหลักการจะเป็นการหาความสัมพันธ์ระหว่างตัวแปรที่จะประเมินว่าตัวแปรหนึ่งมีผลต่อตัวแปรอื่นอย่างไรบ้าง ซึ่งจะนำข้อมูลอินพุตและเป้าหมายมาฝึกฝนให้เครื่องเกิดการเรียนรู้และค่าเป้าหมายจะเป็นข้อมูลที่มีค่าต่อเนื่อง เช่น ค่าตัวเลข 1-100 การเรียนรู้ประเภทนี้ส่วนใหญ่จะนำไปใช้เกี่ยวกับการทำนาย พยากรณ์ค่าต่าง ๆ ไม่ว่าจะเป็นการทำนายการใช้ปริมาณกระแสไฟฟ้าในอนาคต, การทำนายการขึ้นลงของราคาหุ้น, การทำนายผลกำไร เป็นต้น สำหรับการเตรียมข้อมูลก่อนที่จะเข้าไปเรียนรู้นั้น ข้อมูลที่นำมาใช้จะต้องมีคอลัมน์เป้าหมายหรือผลลัพธ์ที่เป็นค่าตัวเลขแบบต่อเนื่อง ดังแสดงในตารางที่ 2.2 เป็นข้อมูลเกี่ยวกับรถยนต์แต่ละรุ่นและผลลัพธ์ที่แสดงปริมาณการปล่อยก๊าซคาร์บอนไดออกไซด์ (CO₂)

ตารางที่ 2.2 ตัวอย่างการจัดชุดข้อมูลก่อนฝึกฝนสำหรับการเรียนรู้แบบถดถอย

ยี่ห้อรถยนต์	รุ่น	เครื่องยนต์	น้ำหนัก	ปริมาณ CO2
โตโยต้า	Aygo	1000	790	99
มิตซูบิชิ	Space Star	1200	1160	95
มินิ	Cooper	1500	1140	105
เมอร์เซเดส-เบนซ์	A-Class	1500	1365	92
ฟอร์ด	Fiesta	1500	1112	98
อวดี	A1	1600	1150	99
ฮุนได	I20	1100	980	99
ซูซูกิ	Swift	1300	990	101
ฟอร์ด	Fiesta	1000	1112	99
โพล์สวาเกน	UP!	1000	929	105

2.3.2 การเรียนรู้แบบไม่มีการสอน

การเรียนรู้แบบไม่มีการสอน (Unsupervised learning) เป็นการสอนให้เครื่องหรือคอมพิวเตอร์เรียนรู้ข้อมูลโดยปราศจากคำตอบซึ่งก็คือข้อมูลเป้าหมายหรือลาเบลเพื่อให้คอมพิวเตอร์หาความสัมพันธ์ของข้อมูลเอง ลักษณะการทำงานของวิธีการเรียนรู้แบบไม่มีการสอนนั้นจะเป็นการป้อนข้อมูลอินพุตที่ต้องการทำนายเข้าไปภายในตัวแบบจำลอง แล้วจากนั้นระบบที่อยู่ภายในแบบจำลองจะดำเนินการประมวลผลหาโครงสร้างของข้อมูลออกมา ในส่วนของข้อมูลที่จะนำเข้าไปฝึกฝนภายในแบบจำลองนั้นจะต้องไม่มีคอลัมน์ที่เป็นข้อมูลเป้าหมายหรือลาเบลซึ่งจะแตกต่างจากการเรียนรู้แบบมีการสอนจะแสดงดังตารางที่ 2.3 ภายในตารางจะประกอบไปด้วยลำดับ, อายุ, น้ำหนักและส่วนสูง สำหรับการเรียนรู้แบบไม่มีการสอนสามารถแบ่งการใช้งานออกเป็น 2 ประเภท ดังนี้

1) การจัดกลุ่มข้อมูล (Clustering): เป็นการจัดข้อมูลตามคุณลักษณะให้เป็นกลุ่มก้อน โดยจะจัดกลุ่มของข้อมูลจากความคล้ายคลึงกัน เช่นการจัดกลุ่มด้านพัฒนาการของเด็กกว่ามีพัฒนาการปกติ, เร็ว, ช้า หรือแม้แต่การจัดกลุ่มผลิตภัณฑ์ทางการค้าโดยสังเกตจากพฤติกรรมในการซื้อของลูกค้าว่าสินค้าประเภทไหนเป็นที่ต้องการมากและสินค้าตัวไหนที่ไม่ได้รับความนิยม เพื่อที่จะได้นำข้อมูลเหล่านี้ไปใช้งานต่อไป เป็นต้น

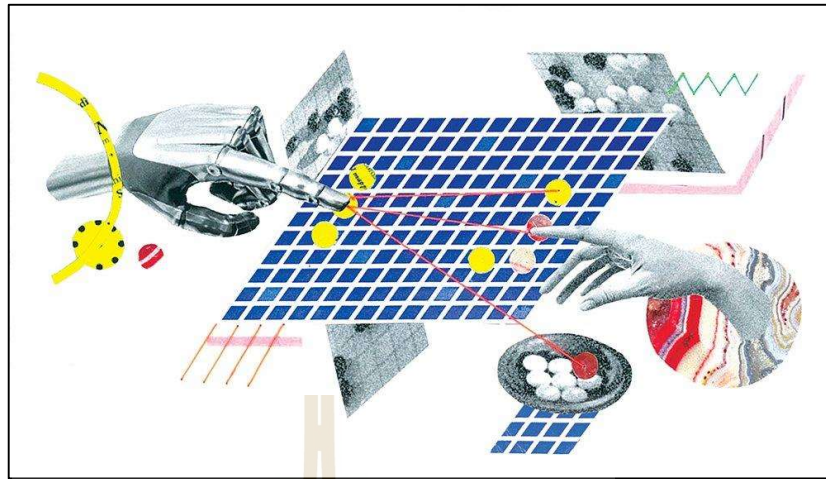
2) การลดมิติของข้อมูล (Dimensionality Reduction): วิธีการนี้เป็นการลดมิติของข้อมูลลงและลบคุณสมบัติ (Feature) ที่ไม่จำเป็นออก เพื่อแสดงผลออกมาให้อ่านค่าได้ง่ายขึ้น อาจจะแสดงในลักษณะของรูปแบบของกราฟ ซึ่งในการลดมิติของข้อมูลนั้นมีข้อดีคือ ทำให้คอมพิวเตอร์สามารถที่จะประหยัดเวลาในการฝึกฝนและเรียนรู้ข้อมูลได้เร็วยิ่งขึ้น สามารถประหยัดทรัพยากรเพื่อลดพื้นที่ในการจัดเก็บข้อมูลและช่วยลดข้อมูลรบกวน (Noise) ที่เกิดจากการซ้ำของข้อมูลได้ เป็นต้น

ตารางที่ 2.3 ตัวอย่างการจัดชุดข้อมูลก่อนฝึกฝนสำหรับการเรียนรู้แบบไม่มีการสอน

ลำดับ	อายุ	น้ำหนัก	ส่วนสูง
1	41	61	159
2	53	75	177
3	45	64	165
4	33	63	157
5	26	85	163

2.3.3 การเรียนรู้แบบเสริมแรง

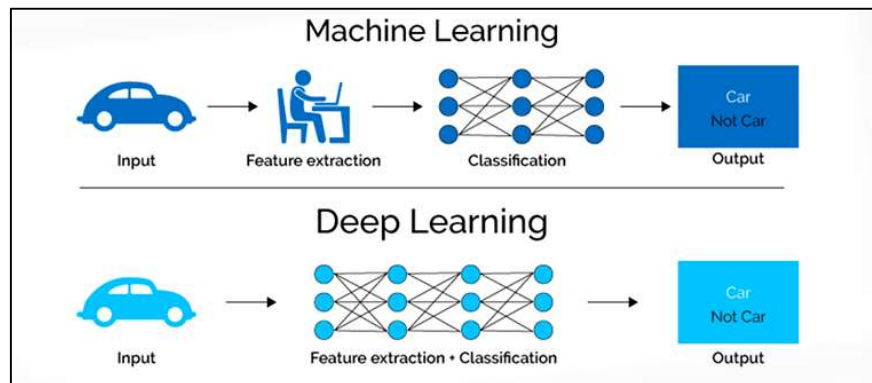
การเรียนรู้แบบเสริมแรง (Reinforcement Learning) เป็นการเรียนรู้ด้วยการไม่ใส่ข้อมูลตัวอย่างเข้าไปสำหรับการสร้างแบบจำลอง แต่จะให้เรียนรู้ไปกับสภาพแวดล้อมที่เป็นอยู่ ณ ขณะนั้นโดยอาศัยวิธีการป้อนกลับให้กับระบบ จากนั้นระบบจะทำการปรับปรุงตัวมันเอง ถ้าเปรียบเทียบการเรียนรู้กับการเรียนรู้ของมนุษย์ก็เปรียบได้กับการลองผิดลองถูกเพื่อทดสอบว่าการกระทำใดเป็นสิ่งที่ดีที่สุดใน หากจะมองให้เข้าใจภาพได้ง่ายยิ่งขึ้นก็เหมือนกับการฝึกเดินตอนเด็ก ๆ อาจจะมีหกล้ม ยืนไม่ตรงหรือเดินไม่ตรงบ้าง แต่เมื่อมีผู้สอนว่าต้องทำท่าทางอย่างไรถึงจะเดินได้คล่อง เด็กก็จะฝึกฝนไปเรื่อย ๆ แล้วเริ่มทำท่าทางได้ถูกต้องผู้สอนก็จะให้รางวัลกับเด็ก เด็กก็จะพบว่าสิ่งที่เขาคำนึงถึงและจดจำ จึงเกิดการซ้ำไปเรื่อย ๆ จนสามารถฝึกเดินเองได้ ฉะนั้นจากหลักการหรือแนวคิดที่กล่าวมานั้นการเรียนรู้แบบเสริมแรงจึงสามารถนำไปใช้กับงานต่างๆ ได้ เช่น การฝึกโปรแกรมให้เดินหมากล้อมองของ AlphaGo ดังรูปที่ 2.4 หรือการทำระบบการทรงตัวของหุ่นยนต์ เป็นต้น



รูปที่ 2.4 การเดินหมากล้อมของ AlphaGo ที่ชนะมนุษย์ด้วยการเรียนรู้แบบเสริมแรง
(<https://minimore.com>)

2.4 การเรียนรู้เชิงลึก

การเรียนรู้เชิงลึก (Deep Learning) เป็นศาสตร์แขนงย่อยที่แตกออกมาจากการเรียนรู้ของเครื่องที่เรียนรู้แบบอัตโนมัติด้วยการเลียนแบบการทำงานของโครงข่ายประสาทจริงทางชีววิทยา (Biological Neural Network) ของมนุษย์ ซึ่งทางเทคนิคจะนำระบบโครงข่ายประสาทเทียมที่มีโครงสร้างของเซลล์ประสาทหรือนิวรอน (Neuron) หลายๆอันมาต่อหรือซ้อนกันและมีความลึกหลายชั้น โดยหลักการทำงานของการเรียนรู้เชิงลึกมีแนวคิดคือ เมื่อมีการป้อนข้อมูลอินพุตเข้าไปภายในโครงข่าย จากนั้นการเรียนรู้เชิงลึกจะดำเนินการหาคุณลักษณะเด่นของข้อมูลออกมาด้วยตนเอง ต่อมาจะดำเนินการสร้างแบบจำลองและทำนายข้อมูลในอนาคต ซึ่งลักษณะเด่นของการเรียนรู้เชิงลึกจะมีความแตกต่างจากการเรียนรู้ของเครื่องคือ สามารถที่จะคัดคุณลักษณะเด่นของข้อมูลออกมาได้เองโดยที่มนุษย์ไม่ต้องเป็นคนลงมือที่สามารถแสดงการทำงานดังรูปภาพที่ 2.5 ดังนี้



รูปที่ 2.5 ลักษณะการทำงานระหว่างการเรียนรู้ของเครื่องและการเรียนรู้เชิงลึก
(<https://blog.thinkwik.com>)

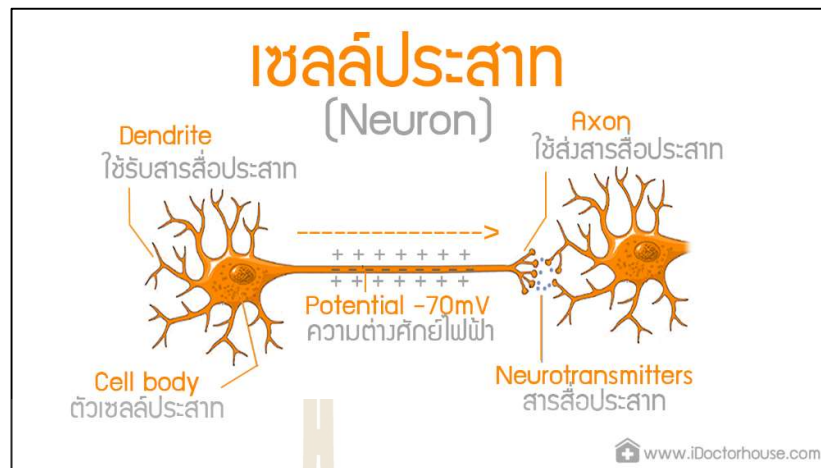
2.4.1 โครงข่ายประสาทเทียม

โครงข่ายประสาทเทียมเป็นการจำลองให้คอมพิวเตอร์มีลักษณะการทำงานที่คล้ายหรือเลียนแบบการทำงานของเซลล์ประสาทภายในสมองของมนุษย์โดยการเขียนโปรแกรมขึ้นมาภายในสมองของมนุษย์จะมีการประมวลผลต่อสิ่งเร้าจากธรรมชาติ ไม่ว่าจะเป็นการรับรู้ความรู้สึก, ความหนาวและความคิดต่างๆ เป็นต้น ทั้งหมดนี้จะเป็นการประมวลผลจากเซลล์ประสาทหรือนิวรอนซึ่งภายในสมองของมนุษย์จะประกอบด้วยนิวรอนจำนวนมากและเชื่อมต่อกันเป็นระบบโครงข่ายที่มีความซับซ้อนซึ่งสามารถแสดงโครงสร้างของเซลล์ประสาทได้ดังรูปที่ 2.5 และเซลล์ประสาทยังมีลักษณะการทำงานดังต่อไปนี้

1) ส่วนการรับข้อมูล ณ ตำแหน่งที่เรียกว่าเดนไดรต์ (Dendrite) หรือเส้นใยประสาทนำเข้าจะทำหน้าที่รับข้อมูลจากสภาพแวดล้อมภายนอกและภายในร่างกายเข้าสู่เซลล์ประสาท

2) ส่วนของการประมวลผล ณ ที่ตัวเซลล์ (Cell Body) ภายในจะประกอบด้วยนิวเคลียส (Nucleus) ซึ่งจะทำหน้าที่ในการตัดสินใจว่าจะกระตุ้นเซลล์อื่นๆ ที่อยู่ภายในเซลล์ประสาทหรือไม่ ถ้าหากมีค่าสัญญาณอินพุตที่แรงมากพอก็จะดำเนินการกระตุ้นและจะส่งสัญญาณกระตุ้นที่เป็นไฟฟ้าเคมีไปยังแอกซ์ซอน

3) ส่วนการส่งข้อมูลออกจะอยู่ที่ตำแหน่งแอกซอน (Axon) จะเป็นตัวส่งข้อมูลแอกซ์ซอนออกหรือเป็นตัวส่งสัญญาณแอกซ์ซอนออกไปยังหน่วยการรับข้อมูลเข้าลำดับถัดไป



รูปที่ 2.6 โครงสร้างของเซลล์ประสาท (<http://www.idoctorhouse.com>)

จากที่กล่าวถึงลักษณะการทำงานภายในสมองของมนุษย์ข้างต้นนั้น ในสมองของมนุษย์จะมีเซลล์ประสาทที่คอยทำหน้าที่รับข้อมูล, ประมวลผลข้อมูลและส่งข้อมูลออก ดังนั้นโครงข่ายประสาทเทียมที่ได้โปรแกรมขึ้นมาให้มีการทำงานคล้ายกับสมองก็จะมีเพอร์เซปตรอน (Perceptron) ที่คอยทำหน้าที่เหมือนเซลล์ประสาทเช่นกัน โดยองค์ประกอบของเพอร์เซปตรอนมีการทำงานดังนี้และแสดงในรูปที่ 2.7

1) ส่วนการรับข้อมูลอินพุต input: $x_1, x_2, x_3, \dots, x_n$ เป็นการรับข้อมูลที่ถูกป้อนเข้ามาภายในโครงข่ายประสาทเทียม เช่น เพศ, ส่วนสูง, น้ำหนัก, ข้อมูลอนุกรมเวลา เป็นต้น ซึ่งจะมีค่าน้ำหนักตัวคูณ (Weight: w_i) อยู่กับข้อมูลอินพุตแต่ละตัว เพื่อที่จะเป็นตัวกำหนดความสำคัญของอินพุตว่าอินพุตตัวไหนมีความสำคัญมากและความสำคัญน้อย จากนั้นจะนำข้อมูลอินพุต, น้ำหนักตัวคูณและค่า w_0 เรียกว่าไบแอส (Bias) จะเป็นตัวช่วยในการปรับการคำนวณตัดสินมารวมเข้าด้วยกันในส่วนของการรวมผล (Sum)

2) ส่วนของการประมวลผลและตัดสินใจจะมีหน้าที่อยู่ 2 ส่วนคือ ส่วนของการรวมผลและส่วนของฟังก์ชันการตัดสินใจ (Activation Function) อันดับแรกในส่วนของการรวมผลจะทำหน้าที่ในการรวมค่าอินพุตคูณกับค่าน้ำหนักตัวคูณที่ประจำแต่ละอินพุต เมื่อรวมผลลัพธ์เข้าด้วยกันแล้วต่อไปจะนำไปรวมเข้ากับค่าไบแอส โดยสมการที่จะใช้ในการรวมผลจะมีรากฐานมาจากสมการเส้นตรงทั่วไปคือ

$$y = mx + c \quad (2.1)$$

โดยที่ y, x คือ ตัวแปร y และตัวแปร x
 m คือ ค่าความชัน
 c คือ ค่าคงที่

แต่ถ้าหากมีข้อมูลอินพุตตั้งแต่ 2 อินพุตขึ้นไป ฉะนั้นในการคำนวณของการรวมผลจะนำข้อมูลอินพุตมารวมเข้ากับตัวคูณแต่ละตัวและรวมเข้ากับค่าไบแอสจะได้สมการดังนี้

$$s = w_0 + (w_1 * x_1) + (w_2 * x_2) + \dots + (w_i * x_n) \quad (2.2)$$

และสามารถเขียนเป็นสมการทางคณิตศาสตร์ได้ดังนี้

$$s = \sum_{i=1}^n w_i * x_n + w_0 \quad (2.3)$$

โดยที่ s คือ ค่าผลรวมระหว่างอินพุต, w_i คือ น้ำหนักตัวคูณและ w_0 คือ ไบแอส

เมื่อรวมผลเรียบร้อยแล้วก็จะนำผลลัพธ์ที่ได้ส่งไปยังส่วนถัดไปคือฟังก์ชันการตัดสินใจ โดยในกระบวนการนี้จะเป็นการประมวลผลที่ใช้ค่าตัวเลขเป็นเกณฑ์ในการตัดสินใจ (Threshold) สามารถกำหนดค่าเป็นเท่าไรก็ได้หรือนำฟังก์ชันการตัดสินใจต่าง ๆ มาช่วยในการประมวลผลที่เหมาะสมกับรูปแบบของงานได้ (ซึ่งจะกล่าวในหัวข้อที่ 2.5) สำหรับลักษณะการทำงานในขั้นตอนนี้จะนำผลรวมที่ได้มาเปรียบเทียบกับเกณฑ์ตัดสินใจ ถ้าผลรวมที่ได้มีค่ามากกว่าเกณฑ์ตัดสินใจก็จะแสดงผลลัพธ์ที่เท่ากับ 1 แต่ถ้าหากผลรวมน้อยกว่าเกณฑ์ตัดสินใจก็จะให้ผลลัพธ์ที่เท่ากับ 0 ซึ่งสามารถเขียนเป็นสมการอย่างง่ายได้ดังนี้

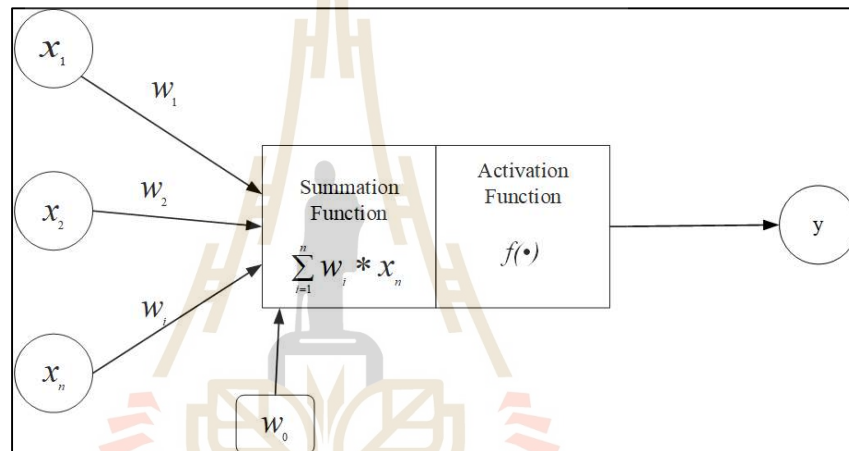
$$s \geq t, \quad y = 1 \quad (2.4)$$

โดยที่ s คือ ค่าผลรวม
 t คือ ค่าเกณฑ์ในการตัดสินใจ
 y คือ เอาต์พุต

3) ส่วนของการส่งออกข้อมูลหรือเอาต์พุต (y) เป็นส่วนที่จะส่งข้อมูลเอาต์พุตออกไปยังเพอร์เซปตรอนถัดไปหรือส่งออกไปแสดงผล สามารถแสดงได้ดังสมการที่ 2.5

$$y = f\left(\sum_{i=1}^n w_i * x_n + w_0\right) \quad (2.5)$$

โดยที่ y คือ ค่าเอาต์พุต, w_i คือ น้ำหนักตัวคูณ, x_n คือ ข้อมูลอินพุต, w_0 คือ ไบแอส และ $f(\bullet)$ คือ ฟังก์ชันการตัดสินใจ



รูปที่ 2.7 โครงสร้างของเพอเซปตรอน

2.4.2 โครงสร้างของโครงข่ายประสาทเทียม

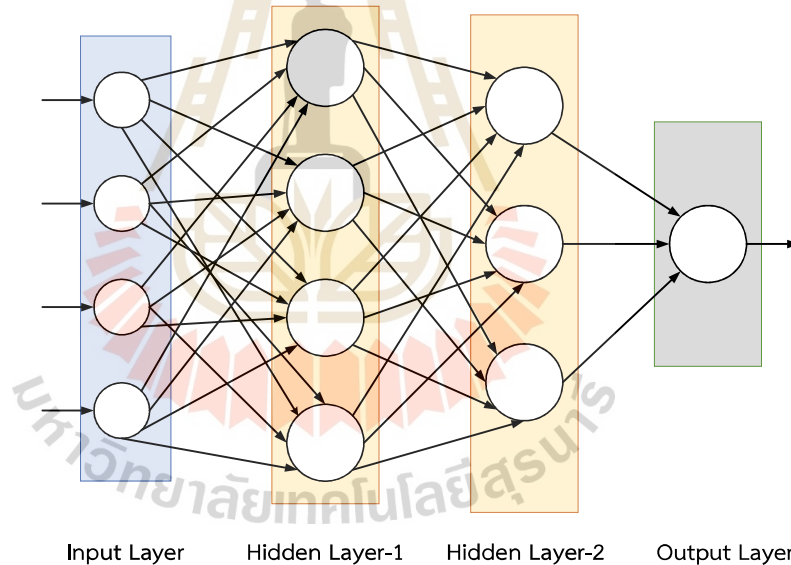
โครงข่ายประสาทเทียมเป็นการเรียนรู้ที่สามารถวิเคราะห์ข้อมูลที่มีลักษณะไม่เป็นเชิงเส้นและข้อมูลตัวเลขได้ เช่น ข้อมูลอนุกรมเวลา เพื่อที่จะสามารถนำไปทำนายหรือพยากรณ์สิ่งต่างๆ ในอนาคต โดยโครงข่ายประสาทเทียมนี้มีคุณสมบัติในการเรียนรู้ข้อมูลที่ถูกป้อนเข้ามาในโครงข่ายและคำนวณด้วยระบบคอมพิวเตอร์จากแบบจำลองทางคณิตศาสตร์ ส่วนประกอบภายในแบบจำลองของโครงข่ายประสาทเทียมจะประกอบไปด้วย 3 ชั้น (Layers) ซึ่งมีหน้าที่แตกต่างกันดังนี้

1) ชั้นอินพุต (Input Layer) : เป็นชั้นที่รับข้อมูลอินพุตจากภายนอกหรือจากเอาต์พุตชั้นก่อนหน้าเข้ามาในแบบจำลองแล้วส่งต่อไปประมวลผลยังชั้นถัดไป ซึ่งภายในชั้นนี้แต่ละอินพุตจะมีค่าน้ำหนักตัวคูณของตัวเองเพื่อระบุถึงความสำคัญของข้อมูล และชั้นอินพุตจะไม่มีฟังก์ชันการตัดสินใจ

2) ชั้นซ่อน (Hidden Layer) : เป็นชั้นที่อยู่กึ่งกลางระหว่างชั้นอินพุตและชั้นเอาต์พุต ซึ่งชั้นนี้จะรับข้อมูลมาจากชั้นอินพุตแล้วนำข้อมูลมาดำเนินการประมวลผลโดยการรวมผลแล้วส่งต่อไปที่ฟังก์ชันการตัดสินใจเพื่อให้ได้ผลลัพธ์ออกมา แต่ผลลัพธ์ที่ได้นั้นจะถูกส่งต่อไปยังชั้นเอาต์พุตเพื่อดำเนินการประมวลผลอีกที และภายในชั้นซ่อนสามารถมีจำนวนโหนดได้มากกว่าหนึ่งโหนดแต่การเพิ่มหรือลดจำนวนโหนดก็จะส่งผลต่อประสิทธิภาพในการทำงานของแบบจำลองได้

3) ชั้นเอาต์พุต (Output Layer) : เป็นชั้นที่รับข้อมูลมาจากชั้นซ่อน โดยจะนำข้อมูลที่ได้นั้นมาทำการรวมผลและตัดสินใจอีกครั้งหนึ่งแล้วจึงแสดงผลที่แท้จริงออกมา ภายในชั้นเอาต์พุตสามารถมีมากกว่าหนึ่งโหนดได้ แล้วแต่การนำไปประยุกต์ใช้กับแต่ละงาน

สามารถแสดงโครงสร้างของโครงข่ายประสาทเทียมได้ดังรูปที่ 2.8 จากรูปจะประกอบไปด้วย 1 ชั้นอินพุตที่มีด้วยกัน 4 โหนด, 2 ชั้นซ่อน โดยชั้นซ่อนแรกจะมี 4 โหนด และชั้นซ่อนที่สองมี 3 โหนด และ 1 ชั้นเอาต์พุตที่ประกอบด้วย 1 โหนด



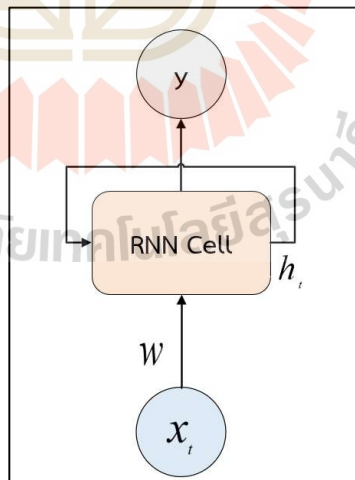
รูปที่ 2.8 ลักษณะโครงสร้างของโครงข่ายประสาทเทียม

2.4.3 หน่วยความจำระยะสั้นยาว

หน่วยความจำระยะสั้นยาวเป็นโครงข่ายประสาทเทียมแบบหนึ่งที่จำลองรูปแบบความจำของมนุษย์ที่สามารถบรรจุความทรงจำได้อย่างจำกัด เนื่องจากหากมีเหตุการณ์ใหม่เข้ามาในสมอง สมองก็จะเลือกจำเหตุการณ์ที่สำคัญ และลบเหตุการณ์ที่ไม่สำคัญออกไป โดยหน่วยความจำ

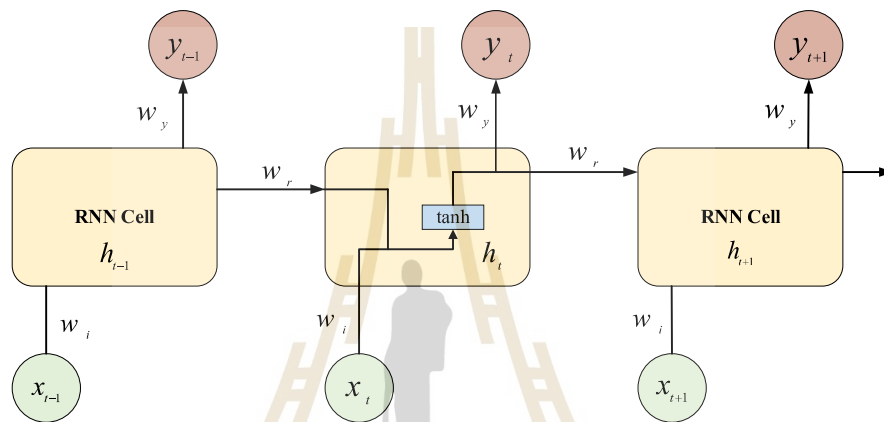
ระยะสั้นยาวออกมาแบบมาให้ประมวลผลข้อมูลเป็นลำดับ (Sequence) จะเหมาะกับข้อมูลที่มีความสัมพันธ์กันกับลำดับก่อนหน้าและหลัง ซึ่งหน่วยความจำสั้นระยะยาวนี้ได้ถูกพัฒนามาจากโครงข่ายประสาทเทียมแบบก้ำเริบ (Recurrent Neural Network: RNN) เพื่อลดจุดบกพร่องของโครงข่ายประสาทแบบก้ำเริบบางจุดโดยจะกล่าวในลำดับถัดไป ดังนั้นในหัวข้อนี้จึงจะขอเกริ่นนำถึงลักษณะของโครงข่ายประสาทแบบก้ำเริบเสียก่อน

โครงข่ายประสาทแบบก้ำเริบเป็นโครงข่ายประสาทเทียมรูปแบบหนึ่งที่มีหน่วยความจำ (Memory) โดยมีแนวคิดคือสามารถเชื่อมโยงข้อมูลก่อนหน้ากับข้อมูลปัจจุบันได้ ซึ่งลักษณะการทำงานจะมีการเก็บสถานะ (State) ข้อมูลก่อนหน้าเอาไว้เพื่อนำเข้าไปประมวลผลกับข้อมูลที่อยู่ในลำดับถัดไป โดยที่โครงข่ายประสาทเทียมแบบก้ำเริบมีหลักการทำงานคือ จะดำเนินการประมวลผลข้อมูลที่ละขั้นตอนแล้วจากนั้นจะเก็บสถานะของข้อมูลที่เป็นประโยชน์ไว้ที่สถานะซ่อน (Hidden State) เพื่อนำไปใช้ประมวลผลในครั้งถัดไป เช่น ข้อมูลที่ถูกป้อนเป็นคำว่า “ฉันกำลังทำงาน” ตัวโครงข่ายประสาทแบบก้ำเริบจะค่อยๆอ่านข้อมูลที่ป้อนเข้ามาจากซ้ายไปขวาทีละคำ โดยข้อมูลลำดับแรกี่อ่านคือคำว่า “ฉัน” ข้อมูลลำดับที่สองคือคำว่า “กำลัง” และค่อยๆอ่านไปที่ละคำจนถึงคำว่า “งาน” จะทำทีละขั้นตอนไปเรื่อยๆ จบครบ ซึ่งจะไม่ป้อนข้อมูลเข้าไปเป็นก้อนเป็นต้น ในกระบวนการดังกล่าวสามารถแสดงสถาปัตยกรรมของโครงข่ายประสาทแบบก้ำเริบได้ดังรูปที่ 2.9 ดังนี้



รูปที่ 2.9 สถาปัตยกรรมโครงข่ายประสาทเทียมแบบก้ำเริบ

โดยสถาปัตยกรรมของโครงข่ายประสาทเทียมแบบกำเริบจะมีลักษณะการทำงานแบบลูป (Loop) เพื่อให้ข้อมูลก่อนหน้ายังคงอยู่ จากรูปด้านบนโครงข่ายประสาทเทียมแบบกำเริบจะประกอบไปด้วย ชั้นของอินพุต (x_t), เอาต์พุต (y_t) และชั้นซ่อนคือที่ RNN Cell ในโครงข่ายประสาทเทียมแบบกำเริบ จะมีส่วนที่เรียกว่าสถานะซ่อน ซึ่งก็คือ h_t จากที่กล่าวไปแล้วข้างต้นเพิ่มเข้ามา โดยจะเป็นส่วนที่ต้องการหาค่าสถานะปัจจุบันของข้อมูลจากการคำนวณค่าของสถานะก่อนหน้าร่วมกับข้อมูลอินพุต ณ เวลาปัจจุบัน เมื่อนำส่วน RNN cell มาคลี่ดังรูปที่ 2.10



รูปที่ 2.10 โครงสร้างภายในโครงข่ายประสาทแบบกำเริบ

จากโครงสร้างภายในจะพบว่าภายในโครงข่ายประสาทแบบกำเริบจะออกแบบมาเพื่อให้โครงสร้างภายในมีแค่ชั้นเดียว เช่นมีฟังก์ชัน \tanh หนึ่งชั้นและมีการส่งป้อนข้อมูลกลับเพียงแค่หนึ่งค่า ดังแสดงในรูปที่ 2.10 ซึ่ง ณ จุดนี้จะทำให้โครงข่ายประสาทแบบกำเริบเกิดข้อเสียขึ้นมา เช่น เมื่อป้อนข้อมูลเข้าไปในโครงข่ายมากๆ อาจจะเป็นข้อมูลตัวเลข 1-100 ตัวโครงข่ายจะเริ่มมีปัญหาเกี่ยวกับชุดข้อมูลที่มีจำนวนข้อมูลก่อนหน้ามากๆ ส่งผลให้ข้อมูลในชุดแรก อาทิ ช่วงของเลข 1,2,3 เกิดการไม่อัปเดตน้ำหนักตัวคุณ จึงเป็นปัญหาที่เรียกว่า Vanishing Gradient ขึ้น แต่ปัญหา ณ จุดนี้จึงได้เกิดการพัฒนาอัลกอริทึมขึ้นมาใหม่คือหน่วยความจำระยะสั้นยาวเพื่อมาแก้ปัญหา โดยจะกล่าวในลำดับถัดไปในโครงข่ายประสาทแบบกำเริบจะสามารถหาค่าสถานะซ่อนได้ดังสมการที่ 2.6 ดังนี้

$$h_t = f_h(w_i \cdot x_t + w_r \cdot h_{t-1} + b_h) \quad (2.6)$$

โดยที่ h_t คือ ค่าสถานะซ่อน ณ เวลาปัจจุบัน

f_h คือ ฟังก์ชันในการประมวลผลหรือฟังก์ชันการตัดสินใจที่สถานะซ่อน

h_{t-1} คือ ค่าสถานะซ่อนของสถานะก่อนหน้า ($t-1$)

x_t คือ ข้อมูลอินพุตที่เวลาปัจจุบัน

w_t คือ ค่าน้ำหนักตัวคูณที่นำมาคูณกับค่าอินพุต

w_r คือ ค่าน้ำหนักตัวคูณของสถานะซ่อน

b_h คือ ค่าไบแอสของโครงข่ายประสาทแบบกำเร็บ

และในส่วนของการประมวลผลผลลัพธ์เพื่อหาค่าเอาต์พุตสุดท้ายหรือเอาต์พุตที่มาจากโหนดก่อนหน้าจะหาได้จากสมการที่ 2.7 คือ

$$y_t = f_y(w_y \cdot h_t + b_y) \quad (2.7)$$

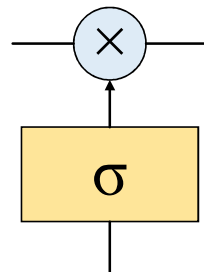
โดยที่ y_t คือ ค่าของเอาต์พุต

f_y คือ ฟังก์ชันในการประมวลผลหรือฟังก์ชันการตัดสินใจที่ชั้นเอาต์พุต

w_y คือ น้ำหนักตัวคูณของโหนดเอาต์พุต

b_y คือ ค่าไบแอสของโหนดเอาต์พุต

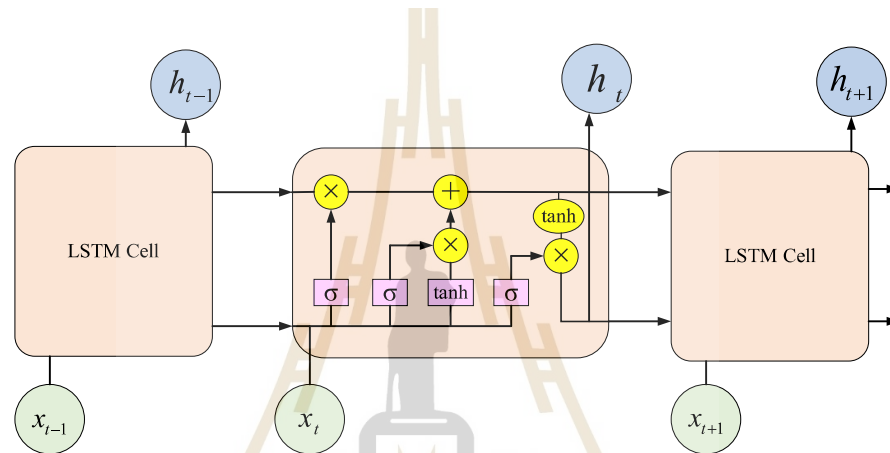
สำหรับหน่วยความจำสั้นระยะยาวที่ถูกพัฒนาเพื่อแก้ปัญหาเรื่อง Vanishing Gradient นั้น จุดเด่นของหน่วยความจำระยะสั้นยาวจะอยู่ที่ สถานะเซลล์ (Cell State) ซึ่งจะมีหน้าที่ในการเก็บข้อมูลที่เป็นประโยชน์หรือลืม (Forget) ข้อมูลนั้นทิ้งไป จะทำหน้าที่คล้ายกับสายพานที่คอยลำเลียงข้อมูลที่จะนำข้อมูลนั้นไปประมวลผลต่อหรือไม่ และความสามารถอีกอย่างหนึ่งของหน่วยความจำระยะสั้นยาวคือ ประตูหรือเกต (Gate) ที่เป็นส่วนที่ตัดสินใจว่าจะเพิ่มข้อมูลนั้นเข้าไปยังสถานะของเซลล์หรือลบข้อมูลนั้นทิ้งไป โดยภายในเกตจะประกอบไปด้วยฟังก์ชันซิกมอยด์ (Sigmoid) และการคูณแบบจุดดังรูปที่ 2.11



รูปที่ 2.11 แสดงลักษณะโครงสร้างภายในของเกต (Gate)

โดยที่ฟังก์ชันซิกมอยด์จะแสดงผลตัวเลขแค่ 0 กับ 1 ซึ่งจะสามารถบอกได้ว่าแต่ละข้อมูลที่เข้ามานั้นผ่านเข้าไปได้มาน้อยเพียงใด โดยหากเป็น 0 คือไม่ปล่อยให้ข้อมูลนั้นผ่านไป แต่หากเป็น 1 จะแสดงว่าปล่อยให้ข้อมูลทุกอย่างที่เข้ามานั้นผ่านไป

ลักษณะโครงสร้างและการทำงานภายในหน่วยความจำระยะสั้นยาวจะประกอบไปด้วย 4 ส่วนคือ ชั้นเกตลืม (Forget Gate Layer), ชั้นเกตอินพุต (Input Gate Layer), อัปเดตสถานะเซลล์ (Update Cell State) และชั้นเกตเอาต์พุต (Output Gate Layer) สามารถแสดงโครงสร้างโดยรวมดังรูปที่ 2.12 และจะอธิบายลักษณะการทำงานเป็นขั้นตอนในลำดับถัดไป



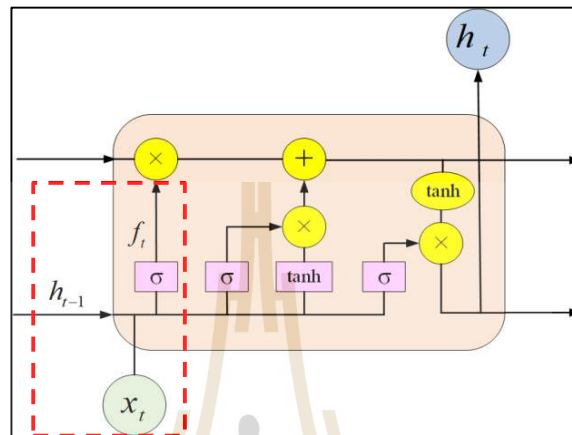
รูปที่ 2.12 ลักษณะโครงสร้างภายในของหน่วยความจำระยะสั้นยาว

ลักษณะการทำงานภายในหน่วยความจำระยะสั้นยาวสามารถแบ่งการทำงานเป็นขั้นตอนได้ดังนี้

ขั้นตอนที่หนึ่ง: ที่ชั้นเกตลืมจะเป็นเกตที่คอยควบคุมว่าข้อมูลที่เข้ามานั้นควรที่จะทิ้งหรือเก็บไว้ในสถานะเซลล์หรือไม่ โดยในการตัดสินใจนี้จะดำเนินการผ่านฟังก์ชันซิกมอยด์ที่จะนำข้อมูลอินพุตที่เข้ามาประกอบกับเมทริกซ์ของสถานะก่อนหน้า ถ้าหากผลที่ได้ออกมาเป็น 1 จะหมายถึงเก็บข้อมูลที่เข้ามานั้นไว้ แต่ถ้าหากผลออกมาเป็น 0 จะหมายถึงให้ลบข้อมูลนั้นทิ้งไป สามารถแสดงสมการของชั้นเกตลืมดังสมการที่ 2.8 และลักษณะการทำงานดังรูปที่ 2.13 ดังนี้

$$f_t = \sigma(w_f \cdot [h_{t-1}, x_t] + b_f) \quad (2.8)$$

โดยที่ f_t คือ ค่าที่ชั้นเกตลิ้ม, σ คือ ฟังก์ชันซิกมอยด์, w_f คือ น้ำหนักตัวคูณของชั้นเกตลิ้ม, h_{t-1} คือ ค่าของสถานะซ่อนก่อนหน้า, x_t คือ ค่าของข้อมูลอินพุต และ b_f คือ ค่าไบแอสที่ชั้นเกตลิ้ม



รูปที่ 2.13 ลักษณะการทำงานของชั้นเกตลิ้ม

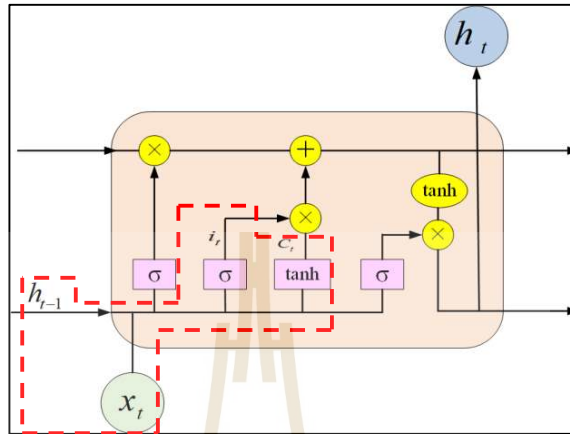
ขั้นตอนที่สอง: เป็นการตัดสินใจว่าจะเพิ่มข้อมูลใหม่ที่สำคัญอะไรเข้าไปเก็บในสถานะเซลล์ซึ่งในขั้นตอนนี้จะประกอบไปด้วย 2 ส่วน ในส่วนแรกคือ ข้อมูลอินพุตที่เข้ามาใหม่สามารถนำมาอัปเดตสถานะเซลล์ได้หรือไม่ ในส่วนนี้จะถูกควบคุมด้วยชั้นเกตอินพุตโดยจะนำข้อมูลอินพุตที่เข้ามาประกอบกับสถานะซ่อนก่อนหน้าเข้าไปประมวลผลผ่านฟังก์ชันซิกมอยด์ที่เป็นตัวตัดสินใจว่าจะดำเนินการอัปเดตสถานะเซลล์หรือไม่ ดังแสดงในสมการที่ 2.9 จากนั้นถ้าหากตัดสินใจว่าให้อัปเดตข้อมูลก็จะมาดำเนินการทำในส่วนที่สองคือเกตมอดูเลตอินพุต (Input Modulation Gate) ในเกตนี้จะประมวลผลผ่านฟังก์ชันไฮเพอร์โบลิกแทนเจนต์ (Hyperbolic Tangent: $\tanh()$) ดังสมการที่ 2.10 เพื่อหาว่าควรจะนำค่าข้อมูลใดมาดำเนินการอัปเดต และลักษณะการทำงานในขั้นตอนที่สองสามารถแสดงได้ดังรูปที่ 2.14

$$i_t = \sigma(w_i \cdot [h_{t-1}, x_t] + b_i) \quad (2.9)$$

$$\tilde{C}_t = \tanh(w_c \cdot [h_{t-1}, x_t] + b_c) \quad (2.10)$$

โดยที่ i_t คือ ค่าของชั้นเกตอินพุต, w_i คือ น้ำหนักตัวคูณของชั้นเกตอินพุต, b_i คือ ค่าไบแอสของชั้นเกตอินพุต, \tilde{C}_t คือ ค่าของเกตมอดูเลตอินพุต, $\tanh()$ คือ ฟังก์ชันไฮเพอร์

โบลิกแทนเจนต์, w_c คือ น้ำหนักตัวคูณของเกตมอดูเลตอินพุต และ b_c คือ ค่าไบแอสของเกตมอดูเลตอินพุต



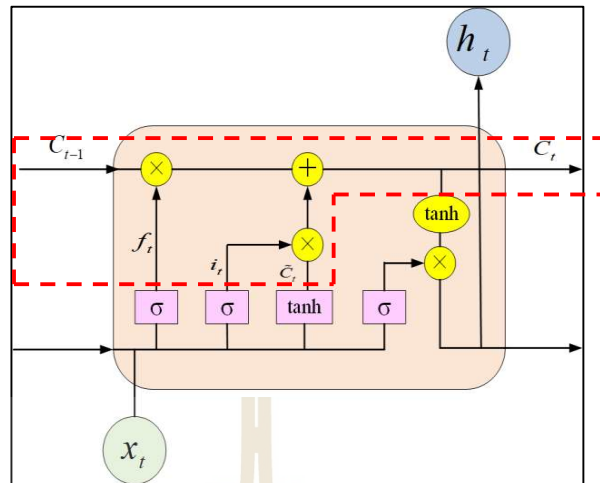
รูปที่ 2.14 ลักษณะการทำงานของชั้นเกตอินพุตและเกตมอดูเลตอินพุต

ขั้นตอนที่สาม: เป็นขั้นตอนในการอัปเดตสถานะเซลล์เก่าเป็นสถานะเซลล์ใหม่ ที่เรียกว่าการอัปเดตสถานะเซลล์ ซึ่งจะรวมข้อมูลที่ได้จากชั้นเกตลืม, ชั้นเกตอินพุตและชั้นเกตมอดูเลตอินพุตเข้าด้วยกันดังสมการที่ 2.11 ดังนี้

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (2.11)$$

โดยที่ C_t คือ สถานะเซลล์ปัจจุบันและ C_{t-1} คือ สถานะเซลล์ก่อนหน้า

จากสมการในการอัปเดตสถานะเซลล์พบว่าหากต้องการอัปเดตสถานะเซลล์ใหม่นั้นจะมีเงื่อนไขด้วยกันอยู่ 2 ข้อคือ เงื่อนไขที่หนึ่ง f_t จะต้องมีค่าเท่ากับ 1 เพราะเมื่อนำมาคูณกับ C_{t-1} ซึ่งหมายความว่า จะเก็บสถานะเซลล์ก่อนหน้าไว้ แต่ถ้า f_t มีค่าเท่ากับ 0 ก็จะลบสถานะเซลล์ก่อนหน้าที่ทิ้งไป และเงื่อนไขที่สอง ณ ที่ i_t จะต้องมีค่าเท่ากับ 1 เช่นกันเพราะเมื่อนำไปคูณกับ \tilde{C}_t จะหมายความว่า ข้อมูลที่ใช้อัปเดตสถานะเซลล์จากข้อมูลใหม่ยังคงอยู่ และเมื่อนำผลที่ได้จากสมการมารวมกันก็จะมี การอัปเดตสถานะเซลล์ใหม่ โดยลักษณะการทำงานสามารถแสดงได้ดังรูปที่ 2.15 ดังนี้



รูปที่ 2.15 ลักษณะการทำงานของเซลล์สถานะเซลล์

ขั้นตอนที่สี่: ขั้นตอนนี้เป็นขั้นตอนสุดท้ายที่ทำหน้าที่รวบรวมข้อมูลและกรองผลลัพธ์ที่ได้เพื่อดำเนินการตัดสินใจว่าจะส่งผลลัพธ์หรือจะผลิตสถานะซ่อนที่เวลาปัจจุบันออกไป ประมวลผลในเซลล์ถัดไปหรือไม่ ซึ่งขั้นตอนนี้เรียกว่าชั้นเกตเอาต์พุต ก่อนที่จะรู้ว่าสามารถผลิตสถานะซ่อนออกไปได้หรือไม่ จะพิจารณาค่าของชั้นเกตเอาต์พุตก่อน โดยในขั้นนี้จะนำค่าของสถานะซ่อนก่อนหน้ากับข้อมูลอินพุตไปประมวลผลผ่านฟังก์ชันซิกมอยด์ดังสมการที่ 2.12 ผลลัพธ์ที่ได้จะแสดงค่าแค่ 0 กับ 1

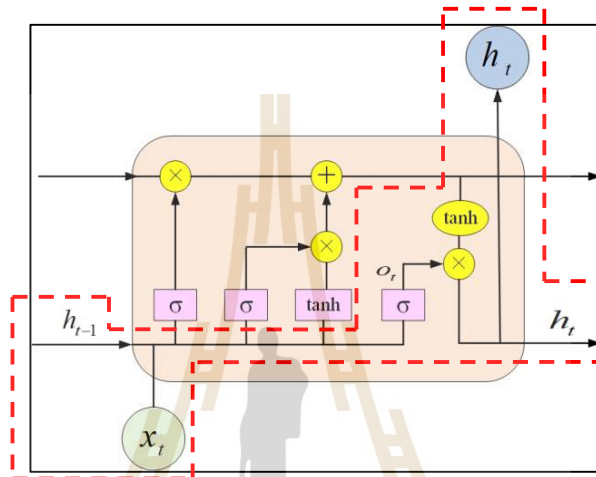
$$o_t = \sigma(w_o[h_{t-1}, x_t] + b_o) \quad (2.12)$$

โดยที่ o_t คือ ค่าของชั้นเกตเอาต์พุต, w_o คือ น้ำหนักตัวคูณของชั้นเกตเอาต์พุต และ b_o คือ ไบแอสของชั้นเกตเอาต์พุต

จากนั้นจะนำค่าที่ผ่านการประมวลผลจากชั้นเกตเอาต์พุตมาหาว่าควรที่จะส่งค่าสถานะซ่อนออกไปหรือไม่ดังสมการที่ 2.13

$$h_t = o_t * \tanh(c_t) \quad (2.13)$$

เมื่อพิจารณาที่สมการด้านขวาพบว่าได้นำค่าสถานะเซลล์ใหม่มาประมวลผลในฟังก์ชัน $\tanh()$ แล้วคูณกับค่าของชั้นเกตเอาต์พุต ถ้าหากค่า o_t เท่ากับ 0 ค่าของสถานะซ่อนก็จะเท่ากับ 0 แสดงว่าไม่มีการส่งค่าของสถานะซ่อนออกไป แต่ถ้าหาก o_t มีค่าเท่ากับ 1 ค่าสถานะซ่อนก็จะถูกคำนวณและถูกส่งออกไปเป็นผลลัพธ์หรือประมวลผลในเซลล์ถัดไปและในขั้นตอนที่สี่นั้นสามารถแสดงลักษณะการทำงานดังรูปที่ 2.16 ดังนี้

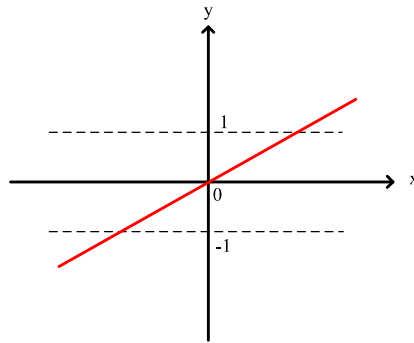


รูปที่ 2.16 ลักษณะการทำงานของชั้นเกตเอาต์พุต

2.5 ฟังก์ชันการตัดสินใจ

ฟังก์ชันการตัดสินใจเป็นส่วนการประมวลผลที่ถัดมาจากการรวมผล เมื่อดำเนินการรวมผลเสร็จเรียบร้อยแล้วก็จะส่งต่อข้อมูลมายังฟังก์ชันการตัดสินใจเพื่อกำหนดว่าจะแสดงผลลัพธ์อะไรออกมา ซึ่งฟังก์ชันการตัดสินใจมีดังนี้

1) ฟังก์ชันการตัดสินใจแบบเชิงเส้น (Linear Activation Function): มีลักษณะการทำงานเป็นเชิงเส้นตามสมการเส้นตรงที่ 2.1 โดยที่ค่าเอาต์พุต y เป็นคำตอบที่ได้จากฟังก์ชันซึ่งจะหาจากค่าความชันหรือค่าใดค่าหนึ่ง (m) แล้วนำมาคูณกับค่าอินพุต (x) ที่ถูกป้อนเข้ามา ผลลัพธ์ที่ได้จะมีค่าระหว่าง $[-\infty, \infty]$ (อนุสุรา, 2020) และสามารถแสดงกราฟได้ดังรูปที่ 2.17 ดังนี้



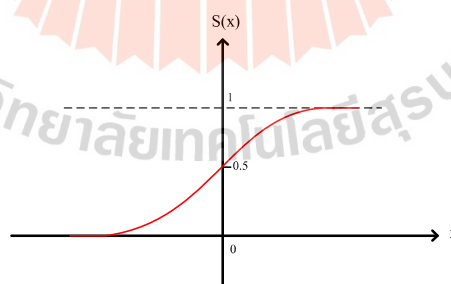
รูปที่ 2.17 ลักษณะกราฟของฟังก์ชันการตัดสินใจแบบเชิงเส้น

2) ฟังก์ชันการตัดสินใจแบบซิกมอยด์ (Sigmoid Function): เป็นฟังก์ชันที่มีลักษณะของกราฟเป็นรูปตัวเอส (S) ดังรูปที่ 2.18 ซึ่งมีหน้าที่ในการแปลงค่าข้อมูลให้อยู่ในช่วงตัวเลขระหว่าง (0,1) โดยฟังก์ชันการตัดสินใจแบบซิกมอยด์จะเหมาะกับการใช้งานที่ใช้ในการจำแนกสิ่งของหรือความน่าจะเป็น โดยสมการซิกมอยด์สามารถหาค่าได้ดังสมการที่ 2.14

$$S(x) = \frac{1}{1 + e^{-x}}$$

(2.14)

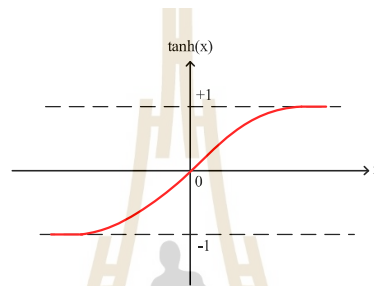
โดยที่ e คือ ค่าคงที่ของ Euler's number



รูปที่ 2.18 ลักษณะกราฟของฟังก์ชันการตัดสินใจแบบซิกมอยด์

3) ฟังก์ชันการตัดสินใจแบบไฮเพอร์โบลิกแทนเจนต์ (Hyperbolic Tangent Activation Function: TanH): เป็นฟังก์ชันที่มีลักษณะกราฟเป็นรูปตัวเอส (S) คล้ายกับฟังก์ชันซิกมอยด์รูปที่ 2.19 แต่มีระยะช่วงของผลลัพธ์ที่กว้างกว่าโดยค่าเอาต์พุตจะมีช่วงระหว่าง $(-1,1)$ ซึ่งเหมาะกับงานที่ใช้จำแนกสิ่งของระหว่างของสองสิ่ง โดยมีแสดงเป็นสมการได้ดังสมการที่ 2.15

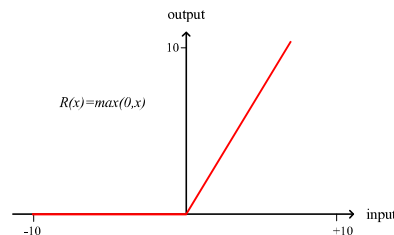
$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.15)$$



รูปที่ 2.19 ลักษณะกราฟของฟังก์ชันการตัดสินใจแบบไฮเพอร์โบลิกแทนเจนต์

4) ฟังก์ชันการตัดสินใจแบบ ReLU (Rectified Linear Unit): เป็นฟังก์ชันที่นิยมนำมาใช้งานกับการเรียนรู้เชิงลึกเนื่องจากถ้าหากพิจารณารูปที่ 2.20 จะพบว่าฟังก์ชันการตัดสินใจนี้จะให้ผลลัพธ์ที่มีค่าเท่ากับ x หรืออินพุตถ้าหากค่าอินพุตที่เข้ามามีค่ามากกว่าศูนย์ แต่ถ้าหากค่าอินพุตที่เข้ามามีค่าติดลบ จะส่งผลให้ค่าเอาต์พุตมีค่าเท่ากับศูนย์ดังสมการที่ 2.16 และข้อดีของฟังก์ชันการตัดสินใจนี้จะทำให้สามารถประมวลผลได้เร็วยิ่งขึ้น อีกทั้งยังใช้พื้นที่ในหน่วยความจำน้อย

$$R(x) = \begin{cases} 0, & \text{if } x < 0 \\ x, & \text{otherwise} \end{cases} \quad (2.16)$$



รูปที่ 2.20 ลักษณะกราฟของฟังก์ชันการตัดสินใจแบบ ReLU

2.6 การพยากรณ์ด้วยวิธีการทางสถิติ

สำหรับการพยากรณ์ข้อมูลปริมาณการจราจรบนระบบเครือข่ายด้วยวิธีการทางสถิติ ในงานวิจัยนี้จะพยากรณ์ข้อมูลด้วยวิธีการหาค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติกับวิธีการพยากรณ์โดยใช้ค่าเฉลี่ยเคลื่อนที่อย่างง่าย โดยจะกล่าวถึงเนื้อหาในลำดับถัดไป

2.6.1 ค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติ

การพยากรณ์ข้อมูลในอนาคตโดยการใช้ข้อมูลในอดีตมาดำเนินการวิเคราะห์ อีกทั้งยังสามารถวิเคราะห์ข้อมูลที่เป็นอนุกรมเวลาได้นั้นตัวแบบที่ได้รับความนิยมมากที่สุดในการทำงานใช้งานคือ ARIMA ที่ถูกเสนอโดย จอร์จ อี.พี. บ็อกซ์ และ กวิลิม เอ็ม. เจนกินส์ (George E.P. Box & Gwilym M. Jenkins) ในปี พ.ศ. 2513 โดยที่หลักการการทำงานของ ARIMA มีจุดเด่นคือการนำข้อมูลที่ได้จากการสังเกตหรือเก็บรวบรวมจากในอดีตมาพยากรณ์เพื่อให้ได้ข้อมูลหรือแนวโน้มที่จะเกิดขึ้นในอนาคต โดยจะให้น้ำหนักความสำคัญกับข้อมูลทุกข้อมูลที่นำมาใช้ในการพยากรณ์ โดยข้อมูลอนุกรมเวลาที่นำมาใช้ในการพยากรณ์ต้องเป็นข้อมูลที่คงที่ (Stationary) ไม่ขึ้นอยู่กับเวลาและฤดูกาล สำหรับการพยากรณ์โดยตัวแบบ ARIMA นั้นจะเป็นการรวมกันของสามองค์ประกอบหรือออเดอร์ (Order) ซึ่งจะประกอบไปด้วย ตัวแบบการถดถอยในตัว (Autoregressive: AR), ตัวแบบบูรณาการ (Integrated: I) และตัวแบบค่าเฉลี่ยเคลื่อนที่ (Moving Average: MA) (Yaffee and McGee, 2000) โดยที่รายละเอียดของทั้งสามองค์ประกอบมีดังนี้

1) ตัวแบบ AR: เป็นตัวแบบการถดถอยที่หาความสัมพันธ์โดยใช้ค่าอนุกรมเวลาของตัวเองในอดีตเพื่อทำนายข้อมูลที่จะเกิดขึ้นในอนาคต โดยตัวแบบการถดถอยนี้จะแสดงด้วยพหุนามดีกรี p บ่งชี้ถึงจำนวนอันดับในอดีต ดังนั้นตัวแบบการถดถอย AR สามารถแสดงอยู่ในรูปแบบของสมการที่ 2.17 (Vijay and Bala, 2019) ดังนี้

$$y_t = l + \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \dots + \alpha_p y_{t-p} + e \quad (2.17)$$

โดยที่ l คือ ระดับในชุดข้อมูล, α คือ ค่าสัมประสิทธิ์ถ่วงน้ำหนักที่ต้องเรียนรู้จากชุดข้อมูลและ e คือ สัญญาณรบกวน

หรือสามารถหาตัวแบบการถดถอยได้อีกหนึ่งวิธีคือหาจากสมการฟังก์ชันสหสัมพันธ์บางส่วน (Partial Autocorrelation Function: PACF) เป็นการหาค่าความสัมพันธ์ของข้อมูลบางส่วนและความล่าช้า (Lag Time) ของตัวเอง โดยการหาความสัมพันธ์ระหว่างช่วงของข้อมูลเวลาในปัจจุบันกับช่วงของข้อมูลเวลา 1 และ 2 วันก่อนหน้า เช่นวัดความสัมพันธ์ระหว่างราคาน้ำมัน ณ

เวลาปัจจุบัน (t) กับราคาน้ำมันของเมื่อวาน ($t-1$) แต่เมื่อพิจารณาราคาน้ำมัน ณ เวลาปัจจุบัน (t) พบว่ามีความสัมพันธ์กับราคาน้ำมันในเวลาสองวันก่อนหน้า ($t-2$) ดังสมการที่ 2.18 (Oduro-Gyimah et al., 2018) ดังนี้

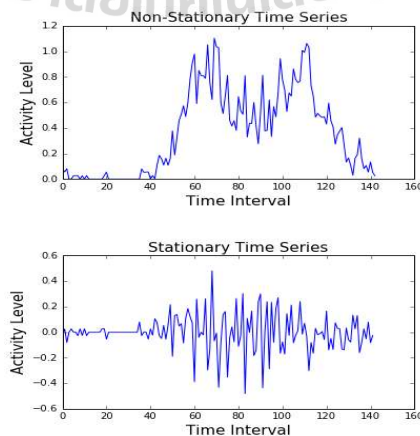
$$PACF = \frac{Covariance(X_t, X_{t-2} | X_{t-1})}{\sqrt{Variance(X_t | X_{t-1})Variance(X_{t-2} | X_{t-1})}} \quad (2.18)$$

โดยที่ X_t คือ ข้อมูลช่วงเวลาปัจจุบัน, X_{t-1} คือ ข้อมูลช่วงเวลาหนึ่งวันก่อนหน้า และ X_{t-2} คือ ข้อมูลช่วงเวลาสองวันก่อนหน้า

2) ตัวแบบบูรณาการ: เนื่องจากข้อมูลอนุกรมเวลาที่นำมาพยากรณ์ส่วนใหญ่จะมีองค์ประกอบของความแปรปรวนและฤดูกาลทำให้ข้อมูลอนุกรมเวลาเป็นข้อมูลแบบไม่คงที่ (Non-Stationary) ส่งผลให้ไม่สามารถนำมาพยากรณ์ได้เพราะจะทำให้เกิดความผิดพลาด ดังนั้นจึงต้องใช้ตัวแบบบูรณาการที่กำจัดองค์ประกอบของความแปรปรวนและฤดูกาลออกจากข้อมูลอนุกรมเวลาเสียก่อนเพื่อให้ได้ข้อมูลอนุกรมเวลาแบบคงที่ด้วยวิธีการหาค่าผลต่าง (Difference) โดยการคำนวณความเปลี่ยนแปลงระหว่างจุดข้อมูลสองจุดที่อยู่ต่อเนื่องกันเพื่อให้ได้ความแตกต่างของลำดับ d ดังสมการที่ 2.19 (Lu, S et al., 2021) และรูปแบบของกราฟของข้อมูลอนุกรมเวลาแบบไม่คงที่และแบบคงที่จะแสดงดังรูปที่ 2.21 ดังนี้

$$\Delta^d y_t = \Delta^{d-1} y_t - \Delta^{d-1} y_{t-1} \quad (2.19)$$

โดยที่ d คือ ลำดับของความแตกต่างและ Δ^d คือ ตัวดำเนินการความแตกต่างของคำสั่ง d



รูปที่ 2.21 ลักษณะกราฟของข้อมูลแบบคงที่และข้อมูลที่ไม่คงที่ (K. Sultan et al., 2018)

3) ตัวแบบ MA: เป็นตัวแบบที่ใช้ค่าความผิดพลาดจากการพยากรณ์ข้อมูลในอดีตมาทำนายข้อมูลในอนาคต โดยตัวแบบ MA จะแสดงอันดับด้วยพารามิเตอร์ q ที่เป็นตัวบ่งชี้ถึงจำนวนค่าความคลาดเคลื่อนจากการพยากรณ์ข้อมูลในอดีต ในการหาค่าตัวแบบ MA สามารถพิจารณาได้จากสมการที่ 2.20 (Vijay and Bala, 2019)

$$y_t = I + e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \dots + \theta_q e_{t-q} \quad (2.20)$$

โดยที่ I คือ ค่าคงที่ (Constant), e_t คือ สัญญาณรบกวนสีขาว (White Noise) และ θ คือ ค่าเฉลี่ยเคลื่อนที่ถ่วงน้ำหนักของข้อผิดพลาด q ที่ผ่านมา

ในการหาค่าตัวแบบ MA ยังสามารถหาได้จากอีกหนึ่งวิธีคือหาจากฟังก์ชันสหสัมพันธ์อัตโนมัติ (Autocorrelation Function: ACF) เป็นการหาความสัมพันธ์ระหว่างข้อมูลในปัจจุบันและข้อมูลความล่าช้าของตัวเองในอดีตจำนวน k ข้อมูลดังแสดงในสมการที่ 2.21 (Oduro-Gyimah et al., 2018) ดังนี้

$$P_k = \frac{\sum_{t=k+1}^T (y_t - \bar{y})(y_{t-k} - \bar{y}_{t-k}) / (t-k)}{\sum_{t=1}^T (y_t - \bar{y})^2 / T} \quad (2.21)$$

$$\text{เมื่อ } \bar{y}_{t-k} = \sum y_{t-k} / (T-k)$$

โดยที่ k คือ ค่าความล่าช้า, T คือ ความยาวของข้อมูลอนุกรมเวลา, y คือ ชุดข้อมูลอนุกรมเวลาและ \bar{y} คือ ค่าเฉลี่ยของชุดข้อมูลอนุกรมเวลา

จากนั้นจึงนำตัวแบบทั้งสามตัวมารวมกันเพื่อให้ได้แบบจำลอง ARIMA ที่สามารถเขียนในรูปแบบของสัญลักษณ์ทั่วไปคือ ARIMA(p,d,q) โดยจากที่กล่าวมาข้างต้นนั้นพารามิเตอร์ p เป็นค่าสัมประสิทธิ์ของตัวแบบ AR, d เป็นจำนวนครั้งที่มีการหาค่าผลต่างของตัวแบบบูรณาการ และ q เป็นสัมประสิทธิ์ของตัวแบบ MA (Carlberg, 2013) สามารถแสดงสมการของแบบจำลอง ARIMA ดังสมการที่ 2.22 ดังนี้

$$y'_t = I + \alpha_1 y'_{t-1} + \alpha_2 y'_{t-2} + \dots + \alpha_p y'_{t-p} + e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \dots + \theta_q e_{t-q} \quad (2.22)$$

2.6.2 ค่าเฉลี่ยเคลื่อนที่อย่างง่าย

ค่าเฉลี่ยเคลื่อนที่อย่างง่ายเป็นส่วนหนึ่งของค่าเฉลี่ยเคลื่อนที่ ซึ่งการพยากรณ์ด้วยวิธีการใช้ค่าเฉลี่ยเคลื่อนที่เพื่อที่จะพยากรณ์ค่าที่จะเกิดขึ้นในอนาคต จะเหมาะกับข้อมูลอนุกรมเวลาที่มีความผิดปกติและเกิดการเปลี่ยนแปลงไม่มากนัก โดยหลักการทำงานของค่าเฉลี่ยเคลื่อนที่อย่างง่ายอันดับแรกจะหาค่าเคลื่อนไหวโดยเลือกข้อมูลอนุกรมเวลาในอดีตมา k จำนวน แล้วจากนั้นจะนำค่าของข้อมูลในอดีต k ค่ามาบวกรวมกันแล้วหาค่าเฉลี่ยออกมาดังสมการที่ 2.23 เมื่อหาค่าเฉลี่ยเรียบร้อยแล้วจะดำเนินการพยากรณ์ข้อมูลดังสมการที่ 2.24 (Chiulli, 1999) ดังสมการด้านล่าง และในกระบวนการทำงานนั้นข้อมูลทุกๆข้อมูลในอดีตจะมีการให้ค่าน้ำหนักของข้อมูลเท่ากันทุกข้อมูล (Yaffee and McGee, 2000)

$$MA(k)_t = \frac{1}{k} \sum_{i=t-k+1}^t y_i \quad (2.23)$$

$$\hat{Y}_{t+1} = MA(k)_t \quad (2.24)$$

โดยที่ k คือ จำนวนข้อมูลหรืออันดับในอดีตเป็นจำนวนเต็มบวก, y_i คือ ค่าข้อมูลในอดีต, $MA(k)_t$ คือ ค่าเคลื่อนไหว ณ เวลา t และ \hat{Y}_{t+1} คือ ค่าพยากรณ์ ณ เวลา $t+1$

2.7 เครื่องมือวัดประสิทธิภาพแบบจำลอง

สำหรับเครื่องมือที่ใช้ในการวัดประสิทธิภาพการพยากรณ์ปริมาณการจราจรบนระบบเครือข่ายของแต่ละแบบจำลองประกอบด้วยดังนี้

2.7.1 ค่าเฉลี่ยของค่าผิดพลาดสัมบูรณ์ (MAE)

เป็นการหาค่าเฉลี่ยของความผิดพลาดจากข้อมูลที่คาดการณ์เปรียบเทียบกับข้อมูลจริงด้วยการหาผลต่างสัมบูรณ์ว่ามีความแตกต่างมากหรือน้อย ถ้าหากแบบจำลองมีค่า MAE น้อยถือว่าแบบจำลองนั้นมีประสิทธิภาพ และสามารถวัดค่าเฉลี่ยของค่าผิดพลาดสัมบูรณ์ได้ดังสมการที่ 2.25

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - x_i| \quad (2.25)$$

โดยที่ y_i คือ ข้อมูลที่ได้จากการทำนาย, x_i คือ ข้อมูลจริง และ n คือ จำนวนตัวอย่างทั้งหมด

2.7.2 ค่าเฉลี่ยความผิดพลาดร้อยละสัมบูรณ์ (MAPE)

เป็นการวัดความแม่นยำหรือความผิดพลาดของแบบจำลองในรูปแบบของเปอร์เซ็นต์สัมบูรณ์ ยังมีค่า MAPE น้อยยิ่งมีประสิทธิภาพดี สามารถหาค่า MAPE ดังสมการที่ 2.26

$$MAPE = \frac{100\%}{n} \sum_{i=1}^n \left| \frac{y_i - x_i}{x_i} \right| \quad (2.26)$$

2.7.3 ค่าเฉลี่ยของค่าผิดพลาดกำลังสอง (Mean Squared Error: MSE)

จากสมการที่ 2.27 เป็นการวัดค่าความผิดพลาดเฉลี่ยกำลังสองระหว่างข้อมูลที่ได้จากการพยากรณ์กับข้อมูลจริง ยิ่งมีค่าน้อยแสดงว่าแบบจำลองมีความแม่นยำ

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - x_i)^2 \quad (2.27)$$

2.7.4 ค่ารากที่สองของความคลาดเคลื่อนกำลังสองเฉลี่ย (RMSE)

เป็นการวัดขนาดเฉลี่ยข้อผิดพลาดโดยการใส่รากที่สองเพื่อวัดระหว่างข้อมูลที่ได้จากการพยากรณ์และข้อมูลจริง ซึ่งผลที่ได้หาค่า RMSE ยิ่งมีค่าน้อยยิ่งดี แสดงได้ดังสมการที่ 2.28

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - x_i)^2} \quad (2.28)$$

2.7.5 ค่าสัมประสิทธิ์ R-Squared (R^2)

เป็นการวัดความสามารถหรือความแม่นยำในการพยากรณ์ของแบบจำลองว่ามีความน่าเชื่อถือมากน้อยเพียงใด โดยจะวัดระหว่างข้อมูลที่ได้จากการพยากรณ์และข้อมูลจริงดังแสดงในสมการที่ 2.29 ถ้า R^2 มีค่าเข้าใกล้ 1 หรือ 100% แสดงว่าแบบจำลองนั้นมีประสิทธิภาพ

$$R^2 = 1 - \frac{\sum_{i=1}^n (x_i - y_i)^2}{\sum_{i=1}^n (x_i - \bar{y})^2} \quad (2.29)$$

โดยที่ $\bar{y} = \frac{1}{n} \sum_{i=1}^n x_i$ คือ ค่าเฉลี่ยของข้อมูลจริง

2.8 ภาษาโปรแกรมมิ่งไพธอน (Python)

สำหรับภาษาที่ใช้ในการเขียนโปรแกรมของงานวิจัยนี้จะใช้ภาษาไพธอน ที่ถูกสร้างโดยนักโปรแกรมเมอร์ชาวดัตช์ที่มีชื่อว่า Guido van Rossum ที่ถูกสร้างขึ้นในปี 1991 ซึ่งไพธอนเป็นภาษาโปรแกรมมิ่งระดับสูงที่ถูกนำมาใช้งานอย่างแพร่หลายเนื่องจากเป็นภาษาที่ง่ายต่อการเรียนรู้, มีการทำงานที่รวดเร็ว และมีความซับซ้อนน้อยกว่าภาษาอื่นเพราะมีโครงสร้างที่เข้าใจได้ง่าย คุณสมบัติหลักของไพธอนคือเป็นภาษาโปรแกรมมิ่งที่ไม่ยึดติดกับแพลตฟอร์มใดหรือเรียกว่าเป็นภาษาเสรี (สุชาติ, 2020) ทำให้นักโปรแกรมเมอร์ต่างๆ สามารถเข้าไปปรับปรุงและพัฒนาไพธอนให้มีประสิทธิภาพที่ดีขึ้นเรื่อย ๆ โดยภาษาไพธอนมีข้อดีอาทิเช่น

- 1) เป็นภาษาที่มีความสามารถในการจัดการกับพื้นที่ในหน่วยความจำได้อย่างมีประสิทธิภาพ ซึ่งเป็นการจัดการแบบอัตโนมัติ
- 2) มีลักษณะการทำงานที่เรียกว่าอินเทอร์พรีเตอร์ (Interpreter) คือลักษณะการทำงานที่มีการประมวลผลทีละบรรทัด ทำให้ใช้เวลาน้อยสำหรับการประมวลผลคำสั่ง
- 3) ชุดไลบรารีของภาษาไพธอน (Library) เหมาะแก่การประมวลผลทางด้านวิศวกรรมศาสตร์และวิทยาศาสตร์
- 4) โครงสร้างตัวแปรของภาษาไพธอนมีความยืดหยุ่น ซึ่งจะทำให้สามารถพัฒนาโปรแกรมได้ง่ายยิ่งขึ้น เช่นข้อมูลแบบลิสต์ (List), ข้อมูลแบบดิกชันนารี (Dictionary) เป็นต้น
- 5) ในภาษาไพธอนมีชุดของไลบรารีที่สนับสนุนเกี่ยวกับงานด้านปัญญาประดิษฐ์ (Artificial Intelligent)

เนื่องจากภาษาไพธอนมีความสามารถในการเขียนโปรแกรมได้หลากหลายและเป็นที่ยอมรับในสมัยนี้ จึงได้ถูกนำไปใช้กับลักษณะงานในรูปแบบต่าง ๆ ไม่ว่าจะเป็น การประยุกต์ใช้กับอินเทอร์เน็ตของสรรพสิ่ง (Internet of Things: IoT), การทำ Chatbot ที่เกี่ยวกับธุรกิจซึ่งเป็นการตอบกลับข้อความอัตโนมัติ, การพยากรณ์ข้อมูลต่างๆ , การสร้างหุ่นยนต์ ที่ต้องใช้ภาษาไพธอนมาเขียนโปรแกรม เป็นต้น

2.9 ปรัชญาบรรณกรรมและงานวิจัยที่เกี่ยวข้อง

จากทฤษฎีที่ได้ดำเนินการศึกษาข้างต้น ในงานวิจัยนี้ยังได้ดำเนินการสำรวจและศึกษา งานวิจัยอื่นๆ เพื่อประกอบกับการทำวิจัยมาพอสังเขป ซึ่งสามารถสรุปได้ดังตารางที่ 2.4 ดังนี้

ตารางที่ 2.4 งานวิจัยที่เกี่ยวข้อง

ชื่อผู้แต่ง	ปีที่ตีพิมพ์	ชื่อบทความ	วิธีดำเนินงานและผลลัพธ์
Aldhyani, T. H. and Joshi, M. R.	2017	Enhancement of Single Moving Average Time Series Model Using Rough k-Means for Prediction of Network Traffic	-เป็นการศึกษาโดยการผสมแบบจำลองระหว่าง RKM (Rough K-mean Approach) เข้ากับ SMA เพื่อคาดการณ์การรับส่งข้อมูลบนระบบเครือข่ายออนไลน์ทุกๆ หนึ่งชั่วโมง - จากการทดลองพบว่าเมื่อผสมทั้งสองแบบจำลองเข้าด้วยกันส่งผลให้ปรับปรุงความแม่นยำในการทำนายให้มีประสิทธิภาพมากยิ่งขึ้น
Oduro-Gyimah, F. K. and Boateng, K. O.	2018	A Comparative Analysis of Telecommunication Network Traffic Forecasting: A Three Model Approach	- ศึกษาการพยากรณ์ข้อมูลการจราจรบนอัปลิงค์ (Uplink) ในเครือข่าย 3G โดยต้องการพิสูจน์ว่า ARIMA, ENN (Elman Neural Network), MLP (Multilayer Perceptron) ว่าสามารถคาดการณ์ข้อมูลที่มีลักษณะไม่เป็นเชิงเส้นได้หรือไม่ - ผลที่ได้จากการทดลองพบว่าทั้ง 3 อัลกอริทึมสามารถคาดการณ์ข้อมูลที่ไม่เป็นเชิงเส้นอย่างข้อมูล 3G ได้อย่างแม่นยำ
Purnawansyah, P.; Haviluddin, H.; Alfred, R. and Gaffar, A. F. O.	2018	Network Traffic Time Series Performance Analysis using Statistical Methods	- เสนอวิธีการทางสถิติคือ Decomposition, Winter's Exponential Smoothing และ ARIMA เพื่อเปรียบเทียบประสิทธิภาพในการทำนายระบบเครือข่ายว่าวิธีการใดที่ดีที่สุด - จากการทดลองพบว่าวิธีการทำนายด้วย ARIMA สามารถทำนายได้แม่นยำที่สุดเนื่องจากมีค่า MSD (Mean Square Deviation) น้อยที่สุด

ตารางที่ 2.4 งานวิจัยที่เกี่ยวข้อง (ต่อ)

ชื่อผู้แต่ง	ปีที่ตีพิมพ์	ชื่อบทความ	วิธีดำเนินงานและผลลัพธ์
Lv, T.; Wu, Y. and Zhang, L.	2021	A Traffic Interval Prediction Method Based on ARIMA	<ul style="list-style-type: none"> - ต้องการสร้างแบบจำลอง ARIMA เพื่อคาดการณ์ข้อมูลเครือข่ายอัปลิงค์และดาวน์โหลด และเปรียบเทียบอัตราความผิดพลาดในการคาดการณ์ระหว่าง LSTM และ Linear Regression - ผลที่ได้พบว่าแบบจำลอง ARIMA มีอัตราความผิดพลาดของการทำนายต่ำกว่า LSTM และ Linear Regression จากการวัดด้วยวิธี MIE (Mean Interval Error)
Azari, A.; Papapetrou, P.; Denic, S. and Peters, G.	2019	Cellular Traffic Prediction and Classification: a comparative evaluation of LSTM and ARIMA	<ul style="list-style-type: none"> - ศึกษาถึงปัญหาและการทำนายข้อมูลในเครือข่ายมือถือ โดยเปรียบเทียบระหว่างแบบจำลอง LSTM และ ARIMA - ผลลัพธ์ที่ได้พบว่า เมื่อข้อมูลที่ใช้ฝึกฝนในแบบจำลองมีขนาดใหญ่และความละเอียดที่เพียงพอ จึงทำให้ LSTM มีประสิทธิภาพในการพยากรณ์ที่เหนือกว่า ARIMA
Zhou, K.; Wang, W. Y.; Hu, T. and Wu, C. H.	2020	Comparison of Time Series Forecasting Based on Statistical ARIMA Model and LSTM with Attention Mechanism	<ul style="list-style-type: none"> - ต้องการเปรียบเทียบประสิทธิภาพในการคาดการณ์ชุดข้อมูลการเข้าใช้งานของเว็บไซต์ Kaggle ระหว่าง ARIMA กับ LSTM ที่รวมกับ Attention Mechanism - ผลการทดลองพบว่า LSTM มีประสิทธิภาพในการคาดการณ์ดีกว่า

ตารางที่ 2.4 งานวิจัยที่เกี่ยวข้อง (ต่อ)

ชื่อผู้แต่ง	ปีที่ตีพิมพ์	ชื่อบทความ	วิธีดำเนินงานและผลลัพธ์
Oliveira, T. P.; Barbar, J. S. and Soares, A. S.	2016	Computer network traffic prediction: a comparison between traditional and deep learning neural networks	<ul style="list-style-type: none"> - ศึกษาการพยากรณ์การรับส่งข้อมูลในเครือข่าย TCP/IP จากผู้ให้บริการ 2 แห่ง โดยแบ่งเป็นข้อมูลรายนาฬิกา, ชั่วโมง และรายวัน ซึ่งจะเปรียบเทียบการพยากรณ์ระหว่าง MLP-BP (Multilayer Perceptron Using Backpropagation), MLP-RP (Multilayer Perceptron with Resilient Backpropagation), RNN, และ SAE (Deep Learning Stacked Autoencoder) - ผลลัพธ์ที่ได้พบว่าการพยากรณ์ด้วยแบบจำลอง RNN มีประสิทธิภาพที่ดีกว่าแบบจำลองอื่นๆ เนื่องจากมีค่า NRMSE (Normalized root mean squared error) ที่เล็กและสามารถพยากรณ์ออกมาได้อย่างรวดเร็ว
Do, Q. H.; Doan, T. T. H.; Nguyen, T. V. A.; Duong, N. T. and Linh, V. V.	2020	Prediction of Data Traffic in Telecom Networks based on Deep Neural Networks	<ul style="list-style-type: none"> - ศึกษาการคาดการณ์ข้อมูลการใช้งานเครือข่ายโทรคมนาคมล่วงหน้า 1 ชั่วโมง ซึ่งจะเปรียบเทียบประสิทธิภาพการคาดการณ์ระหว่าง LSTM, GRU (Gates Recurrent Units), ANFIS (Adaptive Neuro-Fuzzy Inference System), ANN และ GMDH (Group Method of Data Handling) - ผลการคาดการณ์ที่ได้พบว่า LSTM และ GRU มีความแม่นยำมากกว่าวิธีอื่นเนื่องจากมีค่า RMSE, MAPE, MAE ที่มีขนาดเล็กมีค่า Correlation Coefficient

ตารางที่ 2.4 งานวิจัยที่เกี่ยวข้อง (ต่อ)

ชื่อผู้แต่ง	ปีที่ตีพิมพ์	ชื่อบทความ	วิธีดำเนินงานและผลลัพธ์
			หรือค่า R ใกล้เคียง 1 และมีค่า Theil's U-statistic ใกล้เคียง 0
Kirmaz, A.; Michalopoulos, D. S.; Balan, I. and Gerstacker, W.	2020	Mobile Network Traffic Forecasting Using Artificial Neural Networks	<ul style="list-style-type: none"> - ศึกษาการคาดการณ์การใช้งานบนเครือข่ายมือถือ โดยเปรียบเทียบความแม่นยำในการคาดการณ์ระหว่าง ANN และ Linear regression โดยพิจารณาถึงความยาวของหน้าต่างการสังเกต - ผลการทดสอบพบว่าแบบจำลอง ANN สามารถคาดการณ์ได้แม่นยำกว่าและต้องการความยาวของหน้าต่างการสังเกตที่สั้นกว่า
Ciptaningtyas, H. T.; Fatichah, C. and Sabila, A.	2017	Network Traffic Anomaly Prediction Using Artificial Neural Network	<ul style="list-style-type: none"> - เสนอการใช้แบบจำลอง ANN ในการตรวจจับความผิดปกติของการรับส่งข้อมูลบนระบบเครือข่ายโดยการพยากรณ์ เพื่อจะได้ทราบว่าถูกโจมตีหรือไม่ - ผลการทดลองพบว่าแบบจำลอง ANN สามารถพยากรณ์ความผิดปกติของกราฟฟิกับระบบเครือข่ายได้ดี

จากการศึกษาปริทัศน์วรรณกรรมและงานวิจัยที่เกี่ยวข้องทำให้ทราบว่า การพยากรณ์ด้วยการเรียนรู้ของเครื่องส่วนใหญ่มีความสามารถและความแม่นยำในการพยากรณ์ข้อมูลที่มีลักษณะที่ไม่เป็นเชิงเส้นในที่นี้คือข้อมูลปริมาณการใช้งานบนระบบเครือข่าย และการพยากรณ์ด้วยวิธีทางสถิติก็สามารถที่จะนำไปใช้พยากรณ์ข้อมูลที่ไม่เป็นเชิงเส้นได้ด้วยเช่นกัน แต่ในงานที่ได้ศึกษาและกล่าวมาข้างต้นพบว่าส่วนใหญ่จะเป็นการพยากรณ์ข้อมูลที่ไม่เป็นแบบเรียลไทม์ ทำให้อาจที่จะไม่สามารถรองรับต่อจำนวนผู้ใช้งานที่มากขึ้น ณ ขณะเวลานั้น ดังนั้นในงานวิจัยนี้จึงได้นำความรู้ที่ได้จากการศึกษามาเป็นความรู้เพื่อช่วยในการสร้างแบบจำลองที่ดีและพยากรณ์ข้อมูลออกมาให้เป็นข้อมูลแบบเรียลไทม์

2.10 กล่าวสรุป

จากทฤษฎีและงานวิจัยที่เกี่ยวข้องในบทที่ 2 ได้ศึกษาเกี่ยวกับข้อมูลการเรียนรู้ของเครื่อง, การเรียนรู้เชิงลึกที่ประกอบไปด้วยโครงสร้างและหลักการทำงานของโครงข่ายประสาทเทียมและหน่วยความจำระยะสั้นยาว การพยากรณ์ด้วยวิธีทางสถิติที่อธิบายถึงความหมายและหลักการทำงานของค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติและค่าเฉลี่ยเคลื่อนที่อย่างง่าย รวมถึงพารามิเตอร์ต่างๆที่ใช้ในการปรับปรุงและประเมินประสิทธิภาพของแบบจำลอง ล้วนเป็นความรู้ที่ต้องนำมาใช้ในการพยากรณ์ข้อมูลการจราจรบนระบบเครือข่ายทั้งสิ้น ดังนั้นในงานวิจัยนี้จึงศึกษาการพยากรณ์ข้อมูลบนระบบเครือข่ายแบบเรียลไทม์ (Real Time) ด้วยการทำงานร่วมกับวิธีการเลื่อนหน้าต่างและจะเปรียบเทียบประสิทธิภาพในการพยากรณ์ของแต่ละแบบจำลอง



บทที่ 3

วิธีดำเนินการวิจัย

3.1 กล่าวนำ

เนื่องจากในปัจจุบันนี้ผู้คนล้วนต่างใช้งานอินเทอร์เน็ตกันมากขึ้น จึงทำให้การจัดสรรทรัพยากรบนระบบเครือข่ายต่อผู้ใช้งานไม่ทันทางที่ และจากการสอบถามข้อมูลเกี่ยวกับการพยากรณ์จากผู้ให้บริการเครือข่ายรวมทั้งการศึกษาปริทัศน์วรรณกรรมและงานวิจัยที่เกี่ยวข้องมาแล้วนั้น งานวิจัยนี้จึงได้ตัดสินใจที่จะที่จะพยากรณ์ข้อมูลการจราจรบนระบบเครือข่ายแบบเรียลไทม์ประกอบกับการใช้วิธีการเลื่อนหน้าต่างควบคู่ไปด้วย ฉะนั้นในบทนี้จะกล่าวถึงขั้นตอนวิธีการดำเนินงานวิจัยสำหรับการพยากรณ์ข้อมูลการจราจรบนระบบเครือข่ายที่จะเกิดขึ้นในอนาคตก่อนเวลา 5 นาทีและ 30 นาทีซึ่งอาศัยการเรียนรู้ข้อมูลในอดีต จากนั้นจะกล่าวถึงรายละเอียดของข้อมูลที่จะนำมาใช้ในการพยากรณ์ เครื่องมือหรือโปรแกรมที่ใช้รวมถึงวิธีการพยากรณ์ของแต่ละแบบจำลอง เมื่อพยากรณ์ข้อมูลเสร็จสิ้นจะดำเนินการประเมินประสิทธิภาพความแม่นยำหรือข้อผิดพลาดของแต่ละแบบจำลองด้วยเครื่องมือวัดประสิทธิภาพแบบจำลองเพื่อให้ได้แบบจำลองที่ดีที่สุดที่จะนำมาใช้งานต่อไป

3.2 ชุดข้อมูล

ชุดข้อมูลปริมาณการจราจรบนระบบเครือข่ายที่นำมาวิเคราะห์ในงานวิจัยนี้ประกอบไปด้วย 2 ชุดข้อมูลที่จะนำมาศึกษาและพยากรณ์ข้อมูลในอนาคต โดยข้อมูลชุดที่ 1 มาจากงานวิจัยของ (Cortez, P. et.al., 2012) สามารถเข้าถึงชุดข้อมูลได้ โดยเป็นข้อมูลสาธารณะและดาวน์โหลดได้จากเว็บ <http://www3.dsi.uminho.pt/pcortez/data/ittraffic.html> ซึ่งรายละเอียดของชุดข้อมูลจะเป็นข้อมูลที่เก็บรวบรวมจากผู้ให้บริการเครือข่ายอินเทอร์เน็ตส่วนตัว (Private Internet Service Provider) ซึ่งเป็นศูนย์กลางของ 11 เมืองในยุโรป ซึ่งชุดข้อมูลจะเริ่มเก็บตั้งแต่วันที่ 7 มิถุนายน 2548 เวลา 06.57 น. ไปจนถึงวันที่ 28 กรกฎาคม 2548 เวลา 11.17 น. โดยงานวิจัยนี้ได้นำข้อมูลการจราจรบนระบบเครือข่าย (หน่วยเป็น bit per sec (bps)) ที่ถูกเก็บรวบรวมทุกๆ 5 นาทีที่มีข้อมูลรวมทั้งสิ้น 14772 ข้อมูลดังรูปที่ 3.1 และจากรูปที่แสดงจะพบว่าข้อมูลที่ดาวน์โหลดมาจะแสดงแค่ค่าปริมาณการจราจรบนระบบเครือข่ายอย่างเดียว ซึ่งหากจะนำชุดข้อมูลไปดำเนินการฝึกฝนกับแบบจำลองจำเป็นที่จะต้องจัดเรียงชุดข้อมูลใหม่ ดังนั้นผู้วิจัยจึงได้จัดเรียงข้อมูลใหม่จะประกอบไปด้วยข้อมูลราย 5 นาที และ 30 นาทีซึ่งข้อมูล 30 นาทีมาจากการหาค่าเฉลี่ยข้อมูลจำนวน 6 ค่าของข้อมูล 5 นาที และจากนั้นจึงจัดข้อมูลให้สมบูรณ์ยิ่งขึ้นโดยใส่วันที่ เวลา ที่อ้างอิงจากการเก็บข้อมูล

ประกอบด้วยข้อมูลปริมาณการใช้งานบนระบบเครือข่าย ดังรูปที่ 3.2 ที่แสดงการจัดข้อมูลราย 5 นาที และรูปที่ 3.3 เป็นการจัดข้อมูลราย 30 นาที

	A
1	1.06868E+11
2	1.11306E+11
3	1.16324E+11
4	1.16291E+11
5	1.37476E+11
6	1.5049E+11
7	1.56075E+11
8	1.62318E+11
9	1.62242E+11
10	1.67941E+11
11	1.58046E+11
12	1.56179E+11
13	1.66629E+11
14	1.82154E+11
15	1.96002E+11

รูปที่ 3.1 ชุดข้อมูลดิบของการจราจรบนระบบเครือข่ายในยุโรป 5 นาที

Date/Month/Yea	Time	Traffic (bps)
7/6/2005	6:57 AM	1.06868E+11
7/6/2005	7:02 AM	1.11306E+11
7/6/2005	7:07 AM	1.16324E+11
7/6/2005	7:12 AM	1.16291E+11
7/6/2005	7:17 AM	1.37476E+11
7/6/2005	7:22 AM	1.5049E+11
7/6/2005	7:27 AM	1.56075E+11
7/6/2005	7:32 AM	1.62318E+11
7/6/2005	7:37 AM	1.62242E+11
7/6/2005	7:42 AM	1.67941E+11
7/6/2005	7:47 AM	1.58046E+11
7/6/2005	7:52 AM	1.56179E+11
7/6/2005	7:57 AM	1.66629E+11
7/6/2005	8:02 AM	1.82154E+11

รูปที่ 3.2 ชุดข้อมูลการจราจรบนระบบเครือข่ายในยุโรป 5 นาทีที่ถูกจัดเรียงเรียบร้อยแล้ว

Date	30m_Traffic (bps)
7/6/2005-7:22 AM	1.23126E+11
7/6/2005-7:52 AM	1.60467E+11
7/6/2005-8:22 AM	1.8714E+11
7/6/2005-8:52 AM	1.9514E+11
7/6/2005-9:22 AM	2.03972E+11
7/6/2005-9:52 AM	2.08698E+11
7/6/2005-10:22 AM	2.20775E+11
7/6/2005-10:52 AM	2.2321E+11
7/6/2005-11:22 AM	2.29158E+11
7/6/2005-11:52 AM	2.22081E+11
7/6/2005-12:22 PM	2.20436E+11
7/6/2005-12:52 PM	2.30294E+11

รูปที่ 3.3 ชุดข้อมูลการจราจรบนระบบเครือข่ายในยุโรป 30 นาทีที่ถูกจัดเรียงเรียบร้อยแล้ว

สำหรับข้อมูลชุดที่ 2 ได้รับความอนุเคราะห์จากทางศูนย์คอมพิวเตอร์ของมหาวิทยาลัยเทคโนโลยีสุรนารีที่ได้เก็บรวบรวมข้อมูลปริมาณการใช้งานในเครือข่ายภายในมหาวิทยาลัย โดยข้อมูลปริมาณการจราจรบนระบบเครือข่ายถูกเก็บรวบรวมทุกๆ 5 นาที (มีหน่วยเป็น byte per sec: Bps) ข้อมูลจะเริ่มเก็บตั้งแต่วันที่ 15 มิถุนายน 2564 เวลา 00.00 น. จนถึงวันที่ 15 กรกฎาคม 2564 เวลา 03.35 น. เป็นระยะเวลา 1 เดือน โดยลักษณะการจัดเก็บข้อมูลจะดำเนินการเก็บเป็นรายวันและเก็บไว้ในรูปแบบของไฟล์ .log ดังรูปที่ 3.4 จากรูปเป็นการเก็บข้อมูลของวันที่ 15 มิถุนายน 2564 ซึ่งแต่ละคอลัมน์จะประกอบไปด้วยตราเวลา (Timestamp), ข้อมูลเฉลี่ยขาเข้า (Average Incoming), ข้อมูลเฉลี่ยขาออก (Average Outgoing), ค่าข้อมูลขาเข้าสูงสุด (Maximum Incoming), ค่าข้อมูลขาออกสูงสุด (Maximum Outgoing) และจำนวนข้อมูลผู้ใช้งาน (Client) สามารถอ้างอิงได้จากเว็บไซต์ <https://oss.oetiker.ch/mrtg/doc/mrtg-logfile.en.html> จึงได้ดำเนินการจัดเรียงข้อมูลใหม่ด้วยโปรแกรม Excel จะแบ่งเป็นข้อมูลราย 5 นาทีและ 30 นาที ซึ่งข้อมูล 30 นาทีจะมาจากการค่าเฉลี่ยจำนวน 6 ค่าของข้อมูล 5 นาที จากนั้นจะแปลงตราเวลาเป็นวันที่และเวลา และข้อมูลปริมาณการจราจรบนระบบเครือข่ายที่มีหน่วยเป็น Bps จะถูกแปลงเป็นหน่วย bps และจากรูปที่ 3.5 จะเป็นการจัดเรียงข้อมูลราย 5 นาที และรูปที่ 3.6 จะเป็นการจัดเรียงข้อมูลราย 30 นาที สุดท้ายข้อมูลที่จะนำมาพยากรณ์งานวิจัยนี้จะประกอบไปด้วยวันที่, เวลา, ค่าเฉลี่ยขาออกและขาเข้าเท่านั้น

```

*15.06.64.log - Notepad
File Edit View

1623728701 37786741 392403052 37786741 392403052
1623728401 45064592 390475875 45064592 390475875
1623728400 45010572 390255327 45064592 390475875
1623728100 36973505 357258520 38744407 357393802
1623727800 38743392 337295432 38744407 366183504
1623727500 38604212 366168015 42193050 366183504
1623727200 42170862 361431613 42193050 361537091
1623726900 35531123 329968318 35536793 352199735
1623726600 33848621 352258830 37648763 369928342
1623726300 37646888 369936784 37648763 372461155
1623726000 37095035 372261323 39717401 372461155

```

รูปที่ 3.4 ชุดข้อมูลดิบของการใช้งานเครือข่ายภายในมหาวิทยาลัย

Date	Average_Incoming (bps)	Average_Outgoing (bps)
15/06/2564 12:00 AM	85055328	525795288
15/06/2564 12:05 AM	104251248	520796400
15/06/2564 12:10 AM	77693160	536243808
15/06/2564 12:15 AM	57460104	583787192
15/06/2564 12:20 AM	55738808	647429376
15/06/2564 12:25 AM	79576544	779159592
15/06/2564 12:30 AM	77873432	729827736
15/06/2564 12:35 AM	107474888	735731528
15/06/2564 12:40 AM	90481168	782560144
15/06/2564 12:45 AM	113911608	834472016
15/06/2564 12:50 AM	130945136	952066440
15/06/2564 12:55 AM	105503304	1024022672
15/06/2564 1:00 AM	106542296	948385200
15/06/2564 1:05 AM	190110760	1189263160

รูปที่ 3.5 ข้อมูลการใช้งานเครือข่ายมหาวิทยาลัย 5 นาทีที่ได้รับการจัดเรียงแล้ว

Date	30m_Average_Incoming (bps)	30m_Average_Outgoing (bps)
15/06/2564 12:25 AM	354896.1389	2839169.111
15/06/2564 12:55 AM	354896.3056	2839170.444
15/06/2564 1:25 AM	354896.4722	2839171.778
15/06/2564 1:55 AM	354896.6389	2839173.111
15/06/2564 2:25 AM	354896.8056	2839174.444
15/06/2564 2:55 AM	354896.9722	2839175.778
15/06/2564 3:25 AM	354897.1389	2839177.111
15/06/2564 3:50 AM	354897.2778	2839178.222
15/06/2564 4:20 AM	354897.4444	2839179.556
15/06/2564 4:50 AM	354897.6111	2839180.889
15/06/2564 5:20 AM	354897.7778	2839182.222
15/06/2564 5:50 AM	354897.9444	2839183.556
15/06/2564 6:20 AM	354898.1111	2839184.889

รูปที่ 3.6 ข้อมูลการใช้งานเครือข่ายมหาวิทยาลัย 30 นาทีที่ได้รับการจัดเรียงแล้ว

3.3 วิธีดำเนินการทดลอง

สำหรับวิธีการทดลองจะเป็นออกเป็น 2 หัวข้อใหญ่คือ การพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่องและวิธีการทางสถิติ ซึ่งโปรแกรมที่ใช้ในการพยากรณ์ของแบบจำลอง ANN, LSTM และ ARIMA จะนำเสนอด้วยการเขียนโปรแกรมคอมพิวเตอร์ด้วยภาษาไพธอนด้วย Google Colab ซึ่งเป็นบริการจาก Google บนคลาวด์ (Cloud) ซึ่งเป็นแพลตฟอร์มที่สามารถเขียนภาษาโปรแกรมมิ่งโดยภาษาหลักคือภาษาไพธอนและเหมาะกับงานด้านการเรียนรู้ของเครื่อง การใช้งานของโปรแกรมจะใช้เพียงแค่บัญชี Google และอินเทอร์เน็ตเท่านั้น และข้อดีของโปรแกรมคือจะให้ใช้งานหน่วยประมวลผล (CPU) และหน่วยประมวลผลกราฟิกส์ (GPU) ได้ฟรี เหมาะกับการฝึกฝนข้อมูลที่ต้องใช้เวลานาน ส่วนการพยากรณ์ด้วยแบบจำลอง SMA จะดำเนินการพยากรณ์ด้วยโปรแกรม Excel เมื่อพยากรณ์เสร็จสิ้นจะประเมินความแม่นยำของแต่ละแบบจำลองด้วยเครื่องมือวัดประสิทธิภาพแบบจำลอง

3.3.1 การพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่อง

สำหรับการพยากรณ์ปริมาณการจราจรข้อมูลบนระบบเครือข่ายด้วยวิธีการเรียนรู้ของเครื่องจะพยากรณ์ด้วยแบบจำลองโครงข่ายประสาทเทียมและหน่วยความจำระยะสั้นยาว โดยชุดข้อมูลในการพยากรณ์จะแบ่งออกเป็น 2 ชุดใหญ่ ซึ่งชุดข้อมูลชุดที่ 1 ถูกเก็บรวบรวมจากผู้ให้บริการอินเทอร์เน็ตส่วนตัวใน 11 เมืองของยุโรปประกอบไปด้วยชุดข้อมูล 5 นาทีและ 30 นาที และข้อมูลชุดที่ 2 ถูกเก็บรวบรวมจากศูนย์คอมพิวเตอร์ของมหาวิทยาลัยเทคโนโลยีสุรนารีประกอบไปด้วยชุดข้อมูล 5 นาทีและ 30 นาทีเช่นกัน

ขั้นตอนในการพยากรณ์ข้อมูลด้วยโครงข่ายประสาทเทียมและหน่วยความจำระยะสั้นยาวจะมีลักษณะขั้นตอนในการทำงานที่เหมือนกันแต่จะแตกต่างกันที่พารามิเตอร์ที่ใช้สำหรับสร้างแบบจำลอง ในขั้นแรกก่อนที่จะพยากรณ์ข้อมูลจะต้องดำเนินการติดตั้งหรือเรียกใช้งานไลบรารี (Library) ที่จำเป็นเสียก่อน โดยไลบรารีอาจเปรียบได้กับโปรแกรมสำเร็จรูปที่จะทำหน้าที่ในการเก็บฟังก์ชันการทำงานเฉพาะทางต่าง ๆ ฉะนั้นในการพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่องจะเรียกใช้ฟังก์ชันหรือติดตั้งไลบรารีหลัก ๆ ดังนี้

1) Pandas เป็นไลบรารีที่มีความสามารถในการจัดการกับข้อมูลต่างๆ เช่น การโหลดไฟล์ข้อมูลประเภท csv และแสดงผลในลักษณะของตาราง (Table) คือการแบ่งแถวกับคอลัมน์ ซึ่งเรียกว่า Data Frame

2) Numpy เป็นไลบรารีที่ใช้ในการคำนวณทางคณิตศาสตร์ และมีความสามารถในการจัดการกับข้อมูลชุด (Array) ที่มีขนาดใหญ่

3) Matplotlib เป็นไลบรารีพื้นฐานที่มีความสามารถในการสร้างกราฟ

4) Tensorflow เป็นไลบรารีแบบครบวงจรที่เหมาะสมสำหรับสร้างแบบจำลองวิธีการเรียนรู้ของเครื่องและการเรียนรู้เชิงลึก เป็นแพลตฟอร์มที่เปิดให้ใช้งานฟรีจาก Google และรองรับภาษาไพธอนทุกเวอร์ชัน

5) Keras เป็นไลบรารีสำหรับภาษาไพธอน เหมาะสำหรับงานเกี่ยวกับโครงข่ายประสาทและเครือข่ายเชิงลึกที่ทำให้การฝึกฝนแบบจำลองมีความง่ายและในปัจจุบันนี้ไลบรารี Keras แทนที่จะกลายเป็นแพ็คเกจหนึ่งของ tensorflow อีกทั้งภายในไลบรารี Keras สามารถที่จะนำเข้า (Import) โมดูลที่จำเป็น เช่นนำเข้าหน่วยความจำระยะสั้นยาว (LSTM) เพื่อมาสร้างแบบจำลองในการพยากรณ์

6) Scikit-learn เป็นไลบรารีที่เหมาะสมกับวิธีการเรียนรู้ของเครื่อง โดยภายในไลบรารีจะมีอัลกอริทึมที่เหมาะสมกับการฝึกฝนและตรวจสอบความแม่นยำของแบบจำลอง

7) Sequential เป็นการสร้างสแต็กเชิงเส้นของชั้นในแต่ละแบบจำลอง

8) Adam หรือ Adaptive Moment Estimation เป็นอัลกอริทึมที่มีความสามารถในการช่วยเพิ่มประสิทธิภาพแบบจำลอง ซึ่ง Adam มีความสามารถในการฝึกฝนการเรียนรู้เชิงลึก เมื่อฝึกฝนแบบจำลองเรียบร้อยแล้วจะให้ผลลัพธ์ที่มีความแม่นยำและช่วยให้การฝึกฝนข้อมูลมีความรวดเร็ว

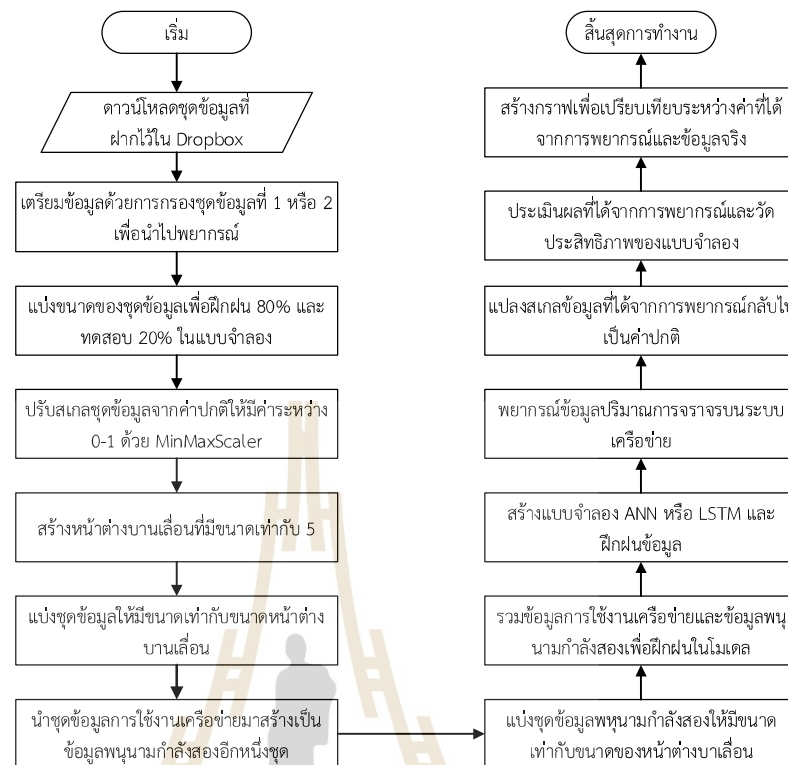
9) Dense เป็นพารามิเตอร์ที่ใช้ในการเชื่อมต่อโหนดทุกโหนดที่อยู่ภายในชั้นเดียวกันและเชื่อมต่อโหนดที่อยู่คนละชั้นเข้าด้วยกัน

10) MinMaxScaler เป็นฟีเจอร์ที่ช่วยในการปรับช่วงของชุดข้อมูลให้อยู่ในช่วงข้อมูลเดียวกัน ซึ่งสามารถที่จะกำหนดค่าต่ำสุดและสูงสุดของข้อมูล

11) PolynomialFeatures เป็นฟีเจอร์ที่ช่วยนำข้อมูลมาสร้างพหุนาม สำหรับการสร้างกำลังของพหุนามสามารถที่จะกำหนดเองได้

12) Math เป็นโมดูลที่เก็บอัลกอริทึมทางคณิตศาสตร์ไว้ เหมาะสำหรับการคำนวณ

เมื่อติดตั้งไลบรารีที่จำเป็นสำหรับการพยากรณ์ด้วยโครงข่ายประสาทเทียมและหน่วยความจำระยะสั้นยาวเสร็จเรียบร้อยแล้ววิธีการดำเนินการพยากรณ์สำหรับชุดข้อมูลแต่ละชุดสามารถอธิบายลักษณะการทำงานด้วยผังงาน (Flowchart) ดังรูปที่ 3.7 ดังแสดงด้านล่าง



รูปที่ 3.7 ผังงานแสดงขั้นตอนสำหรับการพยากรณ์ด้วยแบบจำลอง ANN และ LSTM

สำหรับชุดข้อมูลที่ 1 เป็นชุดข้อมูลปริมาณการจราจรบนระบบเครือข่ายที่ถูกเก็บจากผู้ให้บริการเครือข่ายส่วนตัวของ 11 เมืองในยุโรปประกอบด้วยข้อมูล 5 นาทีและ 30 นาที ซึ่งข้อมูลที่จะนำมาใช้ในการพยากรณ์จะประกอบไปด้วยข้อมูลปริมาณการจราจรบนระบบเครือข่ายและข้อมูลการจราจรบนระบบเครือข่ายที่แปลงเป็นพหุนามยกกำลังสอง สำหรับขั้นตอนในการพยากรณ์มีดังนี้

- 1) โหลดชุดข้อมูลที่ฝากไว้บนคลาวด์คือ Dropbox ด้วยไลบรารี Pandas
- 2) ดำเนินการกรองข้อมูลปริมาณการจราจรบนระบบเครือข่าย ในที่นี้คือข้อมูลปริมาณการใช้งานเครือข่ายที่เก็บรวบรวมจาก 11 เมืองในยุโรป
- 3) จากนั้นจะดำเนินการแบ่งชุดข้อมูลเพื่อใช้สำหรับการฝึกฝนข้อมูลออกเป็น 80 เปอร์เซ็นต์ และข้อมูลที่ใช้ในการทดสอบออกเป็น 20 เปอร์เซ็นต์
- 4) ต่อมาจะดำเนินการแปลงค่าของชุดข้อมูลให้อยู่ในช่วงหรือสเกลเดียวกันด้วยการใช้พีเจอร์ MinMaxScaler ซึ่งจะปรับช่วงของข้อมูลให้อยู่ระหว่าง 0-1

5) เมื่อปรับสเกลของชุดข้อมูลเรียบร้อยแล้ว ขั้นตอนต่อมาคือการสร้างการเลื่อนหน้าต่างให้มีขนาดเท่ากับ 5 จะหมายความว่านำค่าย้อนหลังในอดีต 5 จำนวนเพื่อทำนายค่าในอนาคตคือค่าที่ 6

6) เมื่อสร้างการเลื่อนหน้าต่างเรียบร้อยแล้วจะนำข้อมูลปริมาณการจราจรบนระบบเครือข่ายที่ปรับสเกลแล้วมาแบ่งให้เท่ากับขนาดของการเลื่อนหน้าต่าง ซึ่งจะได้ข้อมูล input#1 มาก่อน

7) จากนั้นจะนำข้อมูลปริมาณการจราจรบนระบบเครือข่ายที่ถูกปรับสเกลแล้วมาแปลงให้เป็นพหุนามยกกำลังสองด้วยพีเจอร์ PolynomialFeatures

8) ถัดไปจะแบ่งข้อมูลพหุนามกำลังสองของปริมาณการจราจรบนระบบเครือข่ายให้มีขนาดเท่ากับขนาดของการเลื่อนหน้าต่างก็จะได้ input#2

9) ขั้นตอนถัดมาจะนำ input#1 กับ input#2 มาต่อเข้ากัน ซึ่งจะได้ input total ที่เป็นชุดข้อมูลที่จะนำไปฝึกฝนและพยากรณ์ในแบบจำลอง

10) เมื่อได้ชุดข้อมูลที่จะนำไปฝึกฝนและพยากรณ์แล้วขั้นตอนถัดไปจะดำเนินการสร้างแบบจำลองของโครงข่ายประสาทเทียมและหน่วยความจำระยะสั้นยาวด้วยไลบรารี Tensorflow และ Keras โดยค่าพารามิเตอร์ที่กำหนดสำหรับการสร้างและฝึกฝนชุดข้อมูล 5 นาที และ 30 นาทีจะแสดงดังตารางที่ 3.1

11) จากนั้นนำข้อมูล input total ที่แบ่งไว้ 80% เข้าไปฝึกฝนกับแบบจำลอง เมื่อฝึกฝนข้อมูลเสร็จเรียบร้อยแล้วจะนำข้อมูล input total ที่แบ่งไว้ 20% ที่เหลือมาทดสอบแบบจำลองและพยากรณ์ค่าข้อมูลในอนาคตออกมา

12) ขั้นตอนต่อไปจะนำข้อมูลที่ได้ออกมาจากการพยากรณ์มาแปลงค่ากลับ จากเดิมข้อมูลจะอยู่ในช่วงระหว่าง 0-1 จึงดำเนินการแปลงค่าให้กลับมาเป็นค่าปกติเพื่อให้อ่านค่าได้ง่ายยิ่งขึ้น

13) จากนั้นจะประเมินผลการทดลองและประสิทธิภาพของแบบจำลองด้วยอัลกอริทึม MAE, MSE, RMSE, MAPE และ R-square ตามลำดับ ซึ่งจะใช้ไลบรารี Scikit-learn และ Math

14) ขั้นตอนสุดท้ายจะแสดงผลด้วยการสร้างกราฟด้วยไลบรารี Matplotlib เพื่อเปรียบเทียบค่าที่ได้จากการพยากรณ์กับค่าปริมาณการจราจรบนเครือข่ายจริง

ตารางที่ 3.1 พารามิเตอร์ที่กำหนดสำหรับการฝึกฝนชุดข้อมูล 5 นาทีและ 30 นาทีของปริมาณการใช้
งานเครือข่ายจาก 11 เมืองในยุโรป

พารามิเตอร์	ปริมาณการใช้งานเครือข่ายของ 11 เมืองในยุโรป			
	ข้อมูล 5 นาที		ข้อมูล 30 นาที	
แบบจำลอง	ANN	LSTM	ANN	LSTM
กำลังพหุนามของข้อมูลปริมาณการจราจร	กำลังสอง		กำลังสอง	
จำนวนชั้นของชั้นซ่อน	2 ชั้น		2 ชั้น	
จำนวนของนิวรอนภายในชั้นซ่อน	128		64	
ฟังก์ชันกระตุ้นที่ทำงานในชั้นซ่อน	relu		relu	
จำนวนชั้นของชั้นเอาต์พุต	1 ชั้น		1 ชั้น	
จำนวนของนิวรอนภายในชั้นเอาต์พุต	1		1	
ฟังก์ชันกระตุ้นที่ทำงานในชั้นเอาต์พุต	linear		linear	
จำนวนผู้ใช้งาน	ไม่นำมาประกอบการพยากรณ์		ไม่นำมาประกอบการพยากรณ์	
จำนวนรอบในการฝึกฝนข้อมูล	200	100	100	

ชุดข้อมูลที่ 2 เป็นชุดข้อมูลปริมาณการจราจรบนระบบเครือข่ายภายในมหาวิทยาลัยเทคโนโลยีสุรนารีประกอบด้วยข้อมูล 5 นาทีและ 30 นาที แต่ข้อมูลปริมาณการจราจรบนระบบเครือข่ายภายในมหาวิทยาลัยจะมีชุดข้อมูลที่แตกต่างกับข้อมูลชุดที่ 1 ซึ่งจะประกอบไปด้วยข้อมูลปริมาณการจราจรบนระบบเครือข่ายที่แบ่งเป็นข้อมูลเฉลี่ยขาเข้า, ข้อมูลเฉลี่ยขาออกและปริมาณจำนวนผู้ใช้งาน จึงทำให้การพยากรณ์ถูกแบ่งออกเป็น 2 กรณี โดยที่กรณีแรกเป็นการพยากรณ์ข้อมูลเฉลี่ยขาเข้าและกรณีที่สองจะเป็นการพยากรณ์ข้อมูลเฉลี่ยขาออก ในกรณีแรกการเตรียมข้อมูลอินพุตเพื่อฝึกฝนและพยากรณ์ในแบบจำลองจะมีขั้นตอนดังรูปที่ 3.7 ดังนี้

- 1) โหลดชุดข้อมูลที่ฝากไว้บนคลาวด์คือ Dropbox ด้วยไลบรารี Pandas
- 2) ดำเนินการกรองข้อมูลปริมาณการจราจรบนระบบเครือข่าย ในที่นี้จะป็นข้อมูลอินพุตของข้อมูลเฉลี่ยขาเข้าซึ่งจะประกอบด้วยข้อมูลสามอินพุตคือ input#1: ปริมาณการจราจรบนระบบเครือข่ายของข้อมูลเฉลี่ยขาเข้า, input#2: พหุนามยกกำลังสองของข้อมูลเฉลี่ยขาเข้าและ input#3: ปริมาณจำนวนผู้ใช้งาน
- 3) จากนั้นจะดำเนินการแบ่งชุดข้อมูลเพื่อใช้สำหรับการฝึกฝนข้อมูลออกเป็น 80 เปอร์เซ็นต์ และข้อมูลที่ใช้ในการทดสอบออกเป็น 20 เปอร์เซ็นต์

4) ต่อมาจะดำเนินการแปลงค่าของข้อมูลอินพุตทั้ง 3 อินพุตให้อยู่ในช่วงหรือสเกลเดียวกันด้วยการใช้พีเจอร์ MinMaxScaler ซึ่งจะปรับช่วงของข้อมูลให้อยู่ระหว่าง 0-1

5) เมื่อปรับสเกลของชุดข้อมูลเรียบร้อยแล้ว ขั้นตอนต่อมาคือการสร้างการเลื่อนหน้าต่างให้มีขนาดเท่ากับ 5 จะหมายความว่านำค่าย้อนหลังในอดีต 5 จำนวนเพื่อทำนายค่าในอนาคตคือค่าที่ 6

6) เมื่อสร้างการเลื่อนหน้าต่างเรียบร้อยแล้วจะนำข้อมูลปริมาณการจราจรบนระบบเครือข่ายเฉลี่ยขาเข้าที่ปรับสเกลแล้วมาแบ่งให้เท่ากับขนาดของการเลื่อนหน้าต่าง ซึ่งจะได้อินพุต#1 มาก่อน

7) จากนั้นจะนำข้อมูลปริมาณการจราจรบนระบบเครือข่ายที่ถูกปรับสเกลแล้วมาแปลงให้เป็นพหุนามยกกำลังสองด้วยพีเจอร์ PolynomialFeatures และจะแบ่งข้อมูลพหุนามกำลังสองของปริมาณการจราจรบนระบบเครือข่ายให้มีขนาดเท่ากับขนาดของการเลื่อนหน้าต่างก็จะได้อินพุต#2

8) ถัดไปจะแบ่งชุดข้อมูลของจำนวนผู้ใช้งานให้มีขนาดเท่ากับขนาดของการเลื่อนหน้าต่างก็จะได้อินพุต#3

9) จากนั้นจะนำ input#1, input#2 และ input#3 มาต่อเข้ากัน ซึ่งจะได้ input total ที่เป็นชุดข้อมูลที่จะเข้าไปฝึกฝนและพยากรณ์ในแบบจำลอง

10) และขั้นตอนสำหรับการฝึกฝนชุดข้อมูล, การพยากรณ์ข้อมูล, การประเมินผล และแบบจำลอง รวมไปถึงการแสดงผลด้วยกราฟจะมีขั้นตอนเหมือนข้อที่ 10-14 ในชุดข้อมูลที่ 1 และพารามิเตอร์ที่กำหนดเพื่อใช้ในการฝึกฝนข้อมูลและพยากรณ์ของข้อมูล 5 นาทีและ 30 นาทีของข้อมูลเฉลี่ยขาเข้าจะแสดงดังตารางที่ 3.2

สำหรับกรณีที่ 2 ข้อมูลอินพุตที่ถูกป้อนเข้าไปในแบบจำลองเพื่อเรียนรู้คือ input#1: ข้อมูลปริมาณการจราจรบนระบบเครือข่ายเฉลี่ยขาออก, input#2: พหุนามยกกำลังสองของข้อมูลเฉลี่ยขาออกและ input#3: ปริมาณจำนวนผู้ใช้งาน ซึ่งขั้นตอนในการเตรียมชุดข้อมูล, ฝึกฝนและพยากรณ์จะเหมือนกับชุดข้อมูลในกรณีแรกซึ่งก็คือปริมาณข้อมูลการจราจรบนระบบเครือข่ายเฉลี่ยขาเข้าหรือมีลักษณะการทำงานดังรูปที่ 3.7 และพารามิเตอร์ที่กำหนดเพื่อใช้ในการฝึกฝนข้อมูลและพยากรณ์ของข้อมูล 5 นาทีและ 30 นาทีของข้อมูลเฉลี่ยขาออกจะแสดงดังตารางที่ 3.3

ตารางที่ 3.2 พารามิเตอร์ที่กำหนดสำหรับการฝึกฝนชุดข้อมูล 5 นาทีและ 30 นาทีของปริมาณการใช้
งานเครือข่ายมหาวิทยาลัยของข้อมูลเฉลี่ยขาเข้า

พารามิเตอร์	ปริมาณการใช้งานเครือข่ายภายในมหาวิทยาลัย ของข้อมูลเฉลี่ยขาเข้า			
	ข้อมูล 5 นาที		ข้อมูล 30 นาที	
แบบจำลอง	ANN	LSTM	ANN	LSTM
กำลังพหุนามของข้อมูลปริมาณการจราจร	กำลังสอง		กำลังสอง	
จำนวนชั้นของชั้นซ่อน	2 ชั้น		2 ชั้น	1 ชั้น
จำนวนของนิวรอนภายในชั้นซ่อน	128		64	
ฟังก์ชันกระตุ้นที่ทำงานในชั้นซ่อน	relu		relu	
จำนวนชั้นของชั้นเอาต์พุต	1 ชั้น		1 ชั้น	
จำนวนของนิวรอนภายในชั้นเอาต์พุต	1		1	
ฟังก์ชันกระตุ้นที่ทำงานในชั้นเอาต์พุต	linear		linear	
จำนวนผู้ใช้งาน	นำมาประกอบการ พยากรณ์		นำมาประกอบการ พยากรณ์	
จำนวนรอบในการฝึกฝนข้อมูล	200	100	100	

ตารางที่ 3.3 พารามิเตอร์ที่กำหนดสำหรับการฝึกฝนชุดข้อมูล 5 นาทีและ 30 นาทีของปริมาณการใช้
งานเครือข่ายมหาวิทยาลัยของข้อมูลเฉลี่ยขาออก

พารามิเตอร์	ปริมาณการใช้งานเครือข่ายภายในมหาวิทยาลัย ของข้อมูลเฉลี่ยขาออก			
	ข้อมูล 5 นาที		ข้อมูล 30 นาที	
แบบจำลอง	ANN	LSTM	ANN	LSTM
กำลังพหุนามของข้อมูลปริมาณการจราจร	กำลังสอง		กำลังสอง	
จำนวนชั้นของชั้นซ่อน	2 ชั้น		1 ชั้น	2 ชั้น
จำนวนของนิวรอนภายในชั้นซ่อน	128		64	
ฟังก์ชันกระตุ้นที่ทำงานในชั้นซ่อน	relu		relu	
จำนวนชั้นของชั้นเอาต์พุต	1 ชั้น		1 ชั้น	
จำนวนของนิวรอนภายในชั้นเอาต์พุต	1		1	
ฟังก์ชันกระตุ้นที่ทำงานในชั้นเอาต์พุต	linear		linear	

ตารางที่ 3.3 พารามิเตอร์ที่กำหนดสำหรับการฝึกฝนชุดข้อมูล 5 นาทีและ 30 นาทีของปริมาณการใช้ งานเครือข่ายมหาวิทยาลัยของข้อมูลเฉลี่ยขาออก (ต่อ)

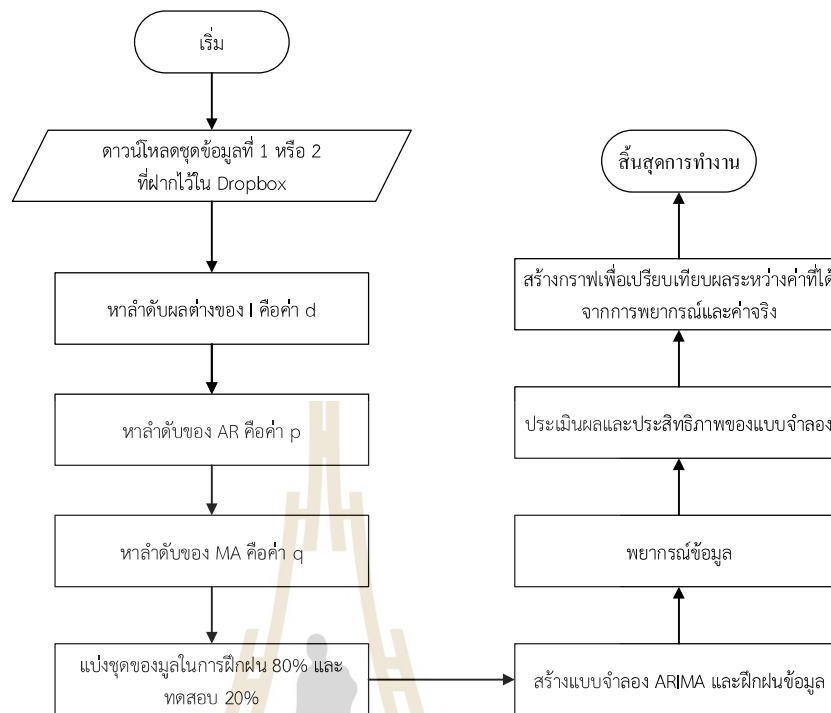
พารามิเตอร์	ปริมาณการใช้งานเครือข่ายภายในมหาวิทยาลัย ของข้อมูลเฉลี่ยขาออก			
	ข้อมูล 5 นาที		ข้อมูล 30 นาที	
แบบจำลอง	ANN	LSTM	ANN	LSTM
จำนวนผู้ใช้งาน	นำมาประกอบการ พยากรณ์		นำมาประกอบการ พยากรณ์	
จำนวนรอบในการฝึกฝนข้อมูล	200	100	200	

3.3.2 การพยากรณ์ด้วยค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติ

สำหรับการพยากรณ์ปริมาณการจราจรบนระบบเครือข่ายด้วยวิธีทางสถิติแรกคือ การพยากรณ์ด้วยค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัตินั้น ข้อมูลที่นำมาพยากรณ์จะแบ่ง ออกเป็น 2 ชุดข้อมูลใหญ่ที่ประกอบไปด้วยข้อมูลชุดที่ 1 ถูกเก็บรวบรวมจากผู้ให้บริการอินเทอร์เน็ต ส่วนตัวใน 11 เมืองของยุโรปซึ่งมีชุดข้อมูลของข้อมูล 5 นาทีและ 30 นาที และข้อมูลชุดที่ 2 ถูกเก็บ รวบรวมจากศูนย์คอมพิวเตอร์ของมหาวิทยาลัยเทคโนโลยีสุรนารีที่มีชุดข้อมูล 5 นาทีและ 30 นาที เช่นกัน ก่อนที่จะเตรียมชุดข้อมูลและพยากรณ์ข้อมูลนั้น จะดำเนินการติดตั้งไลบรารีที่จำเป็นต้องใช้ งานก่อน โดยไลบรารีหลักๆ มีดังนี้

- 1) ไลบรารี Pandas, Numpy, Matplotlib, Math และ Scikit-learn โดยได้อธิบาย ความหมายไว้แล้ว โดยอยู่ในหัวข้อการพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่อง
- 2) Statsmodels เป็นไลบรารีในภาษาไพธอนที่สามารถใช้สร้างแบบจำลองและ วิเคราะห์ข้อมูลทางสถิติซึ่งจะมีฟังก์ชันสำหรับการประมาณค่าของแบบจำลองทางสถิติต่างๆ อีกทั้งยังสามารถเรียกใช้อัลกอริทึมทางสถิติอย่าง ARIMA ได้

เมื่อติดตั้งไลบรารีที่จำเป็นสำหรับการพยากรณ์ด้วยค่าเฉลี่ยเคลื่อนที่แบบบูรณาการ ถดถอยอัตโนมัติเสร็จเรียบร้อยแล้ว ขั้นตอนถัดมาจะเป็นการเตรียมชุดข้อมูล, การสร้างแบบจำลอง เพื่อที่จะพยากรณ์ข้อมูล ซึ่งขั้นตอนการทำงานจะแสดงดังผังงานในรูปที่ 3.8 ดังนี้



รูปที่ 3.8 ผังงานแสดงขั้นตอนสำหรับการพยากรณ์ด้วยแบบจำลอง ARIMA

หลักการเตรียมข้อมูล, การฝึกฝนข้อมูล, การพยากรณ์ข้อมูลและการวัดประสิทธิภาพในการพยากรณ์ รวมถึงการประเมินประสิทธิภาพของแบบจำลองจะใช้หลักการเดียวกัน โดยในการพยากรณ์นี้จะใช้แค่ข้อมูลอินพุตที่เป็นข้อมูลปริมาณการจราจรบนระบบเครือข่ายเท่านั้น ซึ่งจะมีขั้นตอนในการพยากรณ์ดังนี้

1) ขั้นตอนแรกจะโหลดข้อมูลปริมาณการจราจรบนระบบเครือข่ายจากคลาวด์ซึ่งก็คือ Dropbox ด้วยการใช้ไลบรารี pandas โดยในข้อมูลชุดที่ 1 จะกรองข้อมูลปริมาณการใช้งานบนระบบเครือข่ายจาก 11 เมืองในยุโรป หรือข้อมูลชุดที่ 2 จะกรองข้อมูลการจราจรคือข้อมูลเฉลี่ยขาเข้าและข้อมูลเฉลี่ยขาออก

2) จากนั้นจะดำเนินการหา order (p, d, q) ก่อนเพื่อที่จะนำไปสร้างแบบจำลอง โดยอันดับแรกจะหาค่า I หรือลำดับ d ก่อนเพื่อตรวจสอบชุดข้อมูลติดก่อนว่าเป็นข้อมูลที่คงที่หรือไม่ขึ้นอยู่กับเวลาหรือฤดูกาลหรือไม่ ซึ่งจะทดสอบโดยใช้พีเจอร์ Adfuller (Augmented Dickey-Fuller test) ที่เรียกใช้งานโดยไลบรารี Statsmodels หากทดสอบแล้วผลที่ได้มีค่าน้อยกว่า 0.05 แสดงว่าข้อมูลนี้คงที่แล้ว ไม่จำเป็นต้องหาค่าผลต่าง แต่ถ้าหากผลที่ได้มากกว่า 0.05 จำเป็นที่จะต้องหาค่าผลต่างเพื่อทำชุดข้อมูลให้เป็นข้อมูลที่คงที่ก่อน

3) ขั้นตอนถัดไปจะดำเนินการหาค่า AR หรือลำดับ p เพื่อต้องการวัดค่าความสัมพันธ์ของชุดข้อมูล ซึ่งอาจจะอ่านผลจากการสร้างกราฟ PACF ซึ่งจะใช้วิธีการที่เรียกว่า OLS (Ordinary Least Square) เป็นการวิเคราะห์แนวโน้มที่ประมาณค่าสัมประสิทธิ์ด้วยวิธีกำลังสองน้อยที่สุด หรือสามารถที่จะคำนวณหาค่า p จากจุดตัดของกราฟที่ได้สร้างไว้แล้วด้วยการหาค่าจุดตัดตามขีดจำกัดความมั่นใจ (Confidence Interval) ที่ 95% โดยจุดตัดจะมีค่าอยู่ที่ 1.96

4) ต่อมาจะหาลำดับสุดท้ายคือหาค่า MA หรือลำดับ q เพื่อต้องการที่จะวัดความสัมพันธ์ของข้อมูลโดยอาจจะอ่านค่าจากการสร้างกราฟ ACF ขึ้นมาหรือสามารถที่จะคำนวณหาค่า q จากจุดตัดของกราฟที่ได้สร้างไว้แล้วด้วยการหาค่าจุดตัดตามขีดจำกัดความมั่นใจที่ 95% ซึ่งก็มีค่าที่ 1.96 เช่นกัน

5) เมื่อหาค่าลำดับของ p, d, q เรียบร้อยแล้ว ต่อมาจะแบ่งชุดข้อมูลเพื่อทำการฝึกฝนในแบบจำลองออกเป็น 80% และพยากรณ์ 20%

6) ดำเนินการสร้างแบบจำลองด้วยค่าลำดับที่คำนวณมาได้ ด้วยการเรียกใช้อัลกอริทึม ARIMA จากไลบรารี Statsmodels ซึ่งลำดับที่ใช้ในการฝึกฝนและพยากรณ์ข้อมูลปริมาณการใช้งานบนระบบเครือข่ายของ 11 เมืองในยุโรปที่ประกอบด้วยข้อมูล 5 นาทีและ 30 นาทีจะแสดงดังตารางที่ 3.4 และลำดับที่ใช้ในการฝึกฝนและพยากรณ์ข้อมูลปริมาณการใช้งานภายในเครือข่ายมหาลัยที่แบ่งเป็นข้อมูลเฉลี่ยขาเข้าและข้อมูลเฉลี่ยขาออกที่ประกอบด้วยข้อมูล 5 นาทีและ 30 นาทีจะแสดงดังตารางที่ 3.5

7) จากนั้นจะดำเนินการประเมินผลการทดลองและประสิทธิภาพการพยากรณ์ด้วยอัลกอริทึม MAE, MAPE, MSE, RMSE และ R-square ตามลำดับ

8) ขั้นตอนสุดท้ายจะสร้างกราฟเพื่อแสดงผลการเปรียบเทียบข้อมูลระหว่างข้อมูลจริงกับข้อมูลที่ได้จากการพยากรณ์

ตารางที่ 3.4 พารามิเตอร์ของแบบจำลองปริมาณการใช้งานบนระบบเครือข่ายของ 11 เมืองในยุโรป

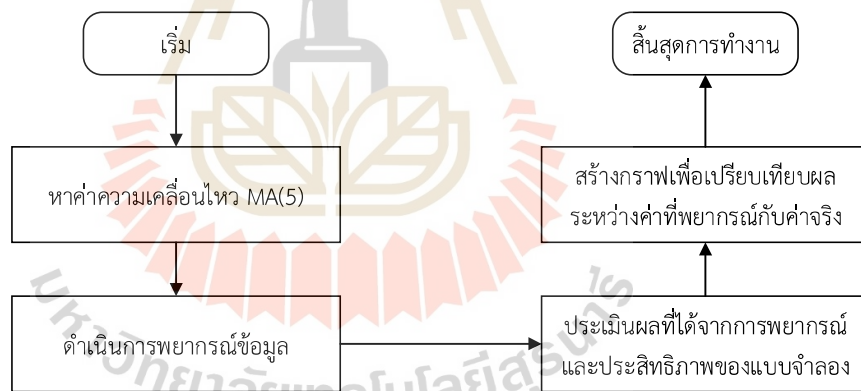
พารามิเตอร์	ข้อมูลปริมาณการใช้งานบนระบบเครือข่ายของ 11 เมืองในยุโรป	
	ข้อมูล 5 นาที	ข้อมูล 30 นาที
Order (p,d,q)	(2,0,0)	(1,0,3)

ตารางที่ 3.5 พารามิเตอร์ของแบบจำลองปริมาณการใช้งานบนระบบเครือข่ายภายในมหาวิทยาลัย

พารามิเตอร์	ข้อมูลปริมาณการใช้งานบนระบบเครือข่ายภายในมหาวิทยาลัย			
	ข้อมูลเฉลี่ยขาเข้า		ข้อมูลเฉลี่ยขาออก	
	ข้อมูล 5 นาที	ข้อมูล 30 นาที	ข้อมูล 5 นาที	ข้อมูล 30 นาที
Order (p,d,q)	(3,0,1)	(1,0,1)	(3,0,0)	(1,0,1)

3.3.3 ค่าเฉลี่ยเคลื่อนที่อย่างง่าย

สำหรับการพยากรณ์ข้อมูลด้วยวิธีการทางสถิติโดยใช้แบบจำลองค่าเฉลี่ยเคลื่อนที่อย่างง่ายนั้นจะคำนวณโดยใช้โปรแกรม Excel ซึ่งข้อมูลที่นำมาพยากรณ์คือข้อมูลปริมาณการจราจรบนระบบเครือข่ายของ 11 เมืองในยุโรปประกอบด้วยข้อมูล 5 นาทีและ 30 นาที และข้อมูลปริมาณการจราจรภายในเครือข่ายมหาวิทยาลัยได้แก่ข้อมูลเฉลี่ยขาเข้าและข้อมูลเฉลี่ยขาออกประกอบด้วยข้อมูล 5 นาทีและ 30 นาที โดยมีแผนภาพที่แสดงขั้นตอนการทำงานดังรูปที่ 3.9 ดังนี้



รูปที่ 3.9 แผนภาพแสดงขั้นตอนสำหรับการพยากรณ์ด้วยแบบจำลอง SMA

จากแผนภาพสามารถอธิบายขั้นตอนการทำงานได้ดังนี้

- 1) ขั้นตอนแรกจะดำเนินการหาค่าความเคลื่อนไหว $MA(k)$, ดังสมการที่ (2.23) โดยที่ k มีค่าเท่ากับ 5 โดยที่จะนำค่าในอดีตจำนวน 5 ค่ามาบวกรวมกันแล้วหารด้วย k จำนวน
- 2) เมื่อได้ค่าเคลื่อนไหวมาแล้วจะดำเนินการพยากรณ์ข้อมูล \hat{Y}_{t+1} ตามสมการที่ (2.24) ซึ่งค่าเคลื่อนไหวที่ได้คำนวณมาแล้วผลที่ได้จะเป็นค่าที่พยากรณ์ในอนาคต ดังเช่นหาค่าเคลื่อนไหวจาก 25 นาทีที่แล้ว ผลลัพธ์ที่ได้จาก 25 นาทีที่แล้วจะเป็นค่าที่พยากรณ์ในนาทีที่ 30

3) จากนั้นจะประเมินผลที่ได้จากการพยากรณ์และวัดประสิทธิภาพของแบบจำลองด้วยอัลกอริทึม MAE, MAPE, MSE, RMSE และ R-square ตามลำดับ

4) สุดท้ายจะสร้างกราฟเพื่อเปรียบเทียบผลระหว่างข้อมูลปริมาณการจราจรบนระบบเครือข่ายจริงกับข้อมูลที่ได้จากการพยากรณ์

3.4 กล่าวสรุป

จากเนื้อหาที่กล่าวมาข้างต้นในบทนี้ได้กล่าวถึงวิธีการดำเนินการวิจัย โดยที่กล่าวถึงชุดข้อมูลที่ได้นำมาใช้ การจัดเตรียมข้อมูลก่อนทำการทดลอง การสร้างแบบจำลองเพื่อฝึกฝนชุดข้อมูลและพยากรณ์ข้อมูลโดยใช้ภาษาโปรแกรมมิ่งไพธอนสำหรับแบบจำลองโครงข่ายประสาทเทียม หน่วยความจำระยะสั้นยาวและค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติ ซึ่งได้กล่าวถึงไลบรารีที่จำเป็นตลอดจนถึงขั้นตอนการทดลองและประเมินผล ส่วนการพยากรณ์ด้วยแบบจำลองเคลื่อนที่อย่างง่ายจะใช้โปรแกรม Excel ในการดำเนินการทดลอง โดยผลที่ได้จากการทดลองจะถูกนำไปอภิปรายผลในบทถัดไป



บทที่ 4

ผลการวิจัยและอภิปรายผล

4.1 กล่าวนำ

จากวิธีการดำเนินการวิจัยในบทที่ผ่านมา เนื้อหาในบทนี้จะกล่าวถึงผลที่ได้จากการพยากรณ์ของแต่ละแบบจำลอง โดยจะนำผลที่ได้จากแต่ละแบบจำลองมาเปรียบเทียบเพื่อแสดงถึงประสิทธิภาพในการพยากรณ์ว่ามีความแม่นยำมากน้อยเพียงใด โดยพารามิเตอร์ที่จะใช้วัดประสิทธิภาพของแบบจำลองจะวัดจากค่าเฉลี่ยของค่าผิดพลาดสัมบูรณ์, ค่าเฉลี่ยความผิดพลาดร้อยละสัมบูรณ์, ค่าเฉลี่ยของค่าผิดพลาดกำลังสอง, ค่ารากที่สองของความคลาดเคลื่อนกำลังสองเฉลี่ย และค่าสัมประสิทธิ์ R-Squared จากนั้นจะนำผลที่ได้มาดำเนินการวิเคราะห์และอภิปรายผลในลำดับถัดไป

4.2 ผลการทดลองของปริมาณการจราจรข้อมูลใน 11 เมืองของยุโรป

4.2.1 ผลการทดลองของข้อมูล 5 นาที

จากการพยากรณ์ข้อมูลปริมาณการจราจรบนระบบเครือข่ายในบทที่แล้ว ในหัวข้อนี้จะเป็นการเก็บผลและประเมินผลการทดลองของข้อมูล 5 นาที ซึ่งผลที่ได้จากการทดลองด้วยวิธีการเรียนรู้ของเครื่องและวิธีการทางสถิติจากแบบจำลองทั้งสี่แบบจะถูกแสดงในตารางที่ 4.1 และกราฟเปรียบเทียบผลการพยากรณ์เทียบกับข้อมูลการใช้งานจริงจะแสดงดังรูปที่ 4.1 โดยที่ภายในกราฟจะประกอบไปด้วยแนวแกนตั้งที่แสดงถึงปริมาณข้อมูลการจราจรบนระบบเครือข่าย (หน่วยเป็น bps) และแนวแกนนอนที่แสดงวันที่และเวลา (หน่วยเป็นวินาที) โดยที่การทดลองจากโครงข่ายประสาทเทียมได้ฝึกฝนข้อมูลเป็นจำนวน 200 รอบซึ่งกำหนดพารามิเตอร์ในชั้นซ่อน 2 ชั้นแต่ละชั้นมี 128 นิวรอนและใช้ฟังก์ชันการตัดสินใจคือ ReLU ส่วนที่ชั้นเอาต์พุตจะใช้เพียง 1 ชั้นภายในชั้นมี 1 นิวรอนและฟังก์ชันการตัดสินใจคือ linear จากผลการทดลองที่ได้เมื่อนำมาคำนวณถึงประสิทธิภาพของการพยากรณ์และข้อผิดพลาดด้วยเครื่องมือวัดประสิทธิภาพพบว่าแบบจำลองโครงข่ายประสาทเทียมมีค่า $MAE=2.94 \times 10^9$, ค่า $MAPE=3.19$, ค่า $MSE=1.53 \times 10^{19}$, ค่า $RMSE=3.91 \times 10^9$ และค่า $R^2=99.60\%$ ต่อมาเป็นผลการพยากรณ์ที่ได้จากหน่วยความจำระยะสั้นยาวที่ฝึกฝนข้อมูลเป็นจำนวน 100 รอบและโครงสร้างประกอบไปด้วยชั้นซ่อน 2 ชั้นแต่ละชั้นมี 128 นิวรอนซึ่งใช้ฟังก์ชัน ReLU เป็นฟังก์ชันในการตัดสินใจ ส่วนชั้นเอาต์พุตมีเพียง 1 ชั้นภายในชั้นมี 1 นิวรอนและใช้ฟังก์ชัน linear ซึ่งจะเป็นฟังก์ชันที่ใช้สำหรับการตัดสินใจ ผลที่ได้พบว่ามีค่า $MAE=2.93 \times 10^9$, ค่า $MAPE=3.15$,

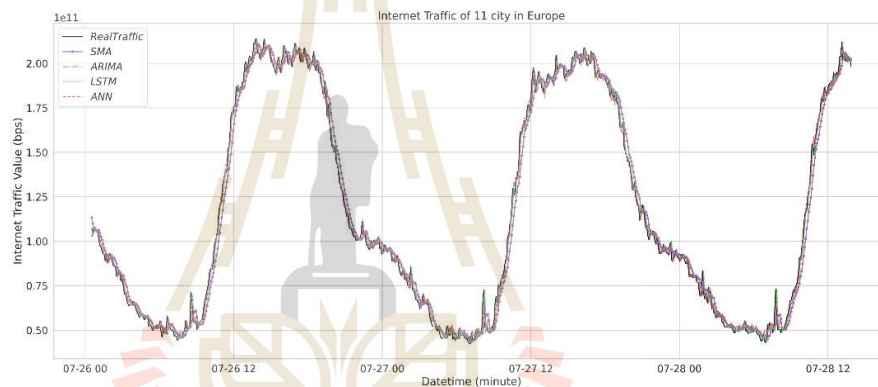
ค่า $MSE=1.54 \times 10^{19}$, ค่า $RMSE=3.93 \times 10^9$ และค่า $R^2=99.59\%$ ส่วนการพยากรณ์ด้วยวิธีทางสถิติ โดยวิธีการแรกเป็นการพยากรณ์ด้วยค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติจะฝึกฝนและพยากรณ์ซึ่งมีจำนวนรอบในการฝึกฝนตามจำนวนข้อมูลที่ได้แบ่งไว้ โดยใช้ order (2,0,0) และผลที่ได้มีค่า $MAE=2.89 \times 10^9$, ค่า $MAPE=76.12$, ค่า $MSE=1.62 \times 10^{19}$, ค่า $RMSE=4.03 \times 10^9$ และค่า $R^2=99.57\%$ และการพยากรณ์สุดท้ายคือค่าเฉลี่ยเคลื่อนที่อย่างง่ายผลที่ได้มีค่า $MAE=4.56 \times 10^9$, ค่า $MAPE=4.45$, ค่า $MSE=4.18 \times 10^{19}$, ค่า $RMSE=6.47 \times 10^9$ และ $R^2=99\%$ เมื่อดำเนินการตรวจสอบผลที่ได้จากการพยากรณ์ข้อมูล 5 นาทีของทั้งสี่แบบจำลองแล้วนั้นพบว่าพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่องมีค่าความผิดพลาดที่น้อยกว่าและมีประสิทธิภาพในการพยากรณ์ดีกว่าแบบวิธีการทางสถิติ อีกทั้งผลที่ได้เมื่อพิจารณาที่รูปของกราฟเห็นว่าการพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่องผลที่ได้มีค่าเข้าใกล้กับการใช้งานของข้อมูลจริง

4.2.2 ผลการทดลองของข้อมูล 30 นาที

ผลที่ได้จากการพยากรณ์ข้อมูล 30 นาทีระหว่างวิธีการเรียนรู้ของเครื่องและวิธีการทางสถิติจะแสดงดังตารางที่ 4.2 และเปรียบเทียบผลโดยการสร้างกราฟดังรูปที่ 4.2 ดังด้านล่าง โดยวิธีการเรียนรู้ของเครื่องที่พยากรณ์ด้วยโครงข่ายประสาทเทียมและหน่วยความจำระยะสั้นยาวที่ฝึกฝนข้อมูล 100 รอบซึ่งที่ชั้นซ่อนจะใช้ฟังก์ชันการตัดสินใจคือ ReLU ภายในประกอบไปด้วย 2 ชั้น และแต่ละชั้นมี 64 นิวรอน และที่ชั้นเอาต์พุตจะใช้ฟังก์ชันการตัดสินใจคือ linear มีเพียง 1 ชั้นและมี 1 นิวรอน ผลที่ได้จากการพยากรณ์ด้วยโครงข่ายประสาทเทียมพบว่ามีค่า $MAE=3.89 \times 10^9$, ค่า $MAPE=4.22$, ค่า $MSE=2.79 \times 10^{19}$, ค่า $RMSE=5.28 \times 10^9$ และค่า $R^2=99.15\%$ และผลที่ได้จากการพยากรณ์ด้วยหน่วยความจำระยะสั้นยาวพบว่ามีค่า $MAE=4.28 \times 10^9$, ค่า $MAPE=5.23$, ค่า $MSE=3.10 \times 10^{19}$, ค่า $RMSE=5.57 \times 10^9$ และค่า $R^2=99.05\%$ สำหรับการพยากรณ์ด้วยวิธีทางสถิติด้วยค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติซึ่งจะฝึกฝนข้อมูลและพยากรณ์ด้วยจำนวนรอบตามข้อมูลที่ได้ถูกแบ่งไว้ โดยใช้ order(1,0,3) ผลที่ได้พบว่ามีค่า $MAE=4.86 \times 10^9$, ค่า $MAPE=72.02$, ค่า $MSE=4.21 \times 10^{19}$, ค่า $RMSE=6.49 \times 10^9$ และค่า $R^2=98.71\%$ และการพยากรณ์ข้อมูลด้วยค่าเฉลี่ยเคลื่อนที่อย่างง่ายผลที่ได้พบว่ามีค่า $MAE=1.79 \times 10^{10}$, ค่า $MAPE=16.65$, ค่า $MSE=6.93 \times 10^{20}$, ค่า $RMSE=2.63 \times 10^{10}$ และค่า $R^2=82.54\%$ เมื่อพิจารณาผลลัพธ์และกราฟเปรียบเทียบผลพบว่าการพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่องมีประสิทธิภาพแล้ว ข้อผิดพลาดน้อยกว่าวิธีการทางสถิติ และจากรูปของกราฟผลที่ได้มีค่าเข้าใกล้ข้อมูลจริง

ตารางที่ 4.1 ประสิทธิภาพของแบบจำลองในการพยากรณ์ระบบเครือข่าย 5 นาทีของ 11 เมืองในยุโรป

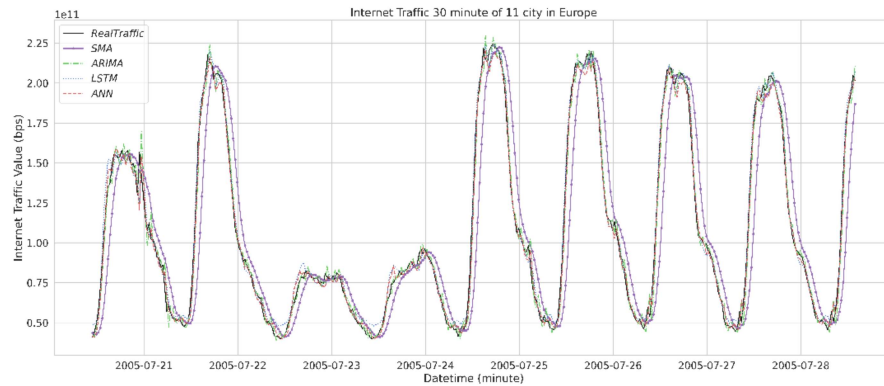
Model	MAE	MAPE	MSE	RMSE	R ²
SMA	4.59×10^9	4.45	4.18×10^{19}	6.47×10^9	99%
ARIMA	2.89×10^9	76.12	1.62×10^{19}	4.03×10^9	99.57%
LSTM	2.93×10^9	3.15	1.54×10^{19}	3.93×10^9	99.59%
ANN	2.94×10^9	3.19	1.53×10^{19}	3.91×10^9	99.60%



รูปที่ 4.1 กราฟแสดงการเปรียบเทียบการพยากรณ์ระบบเครือข่าย 5 นาทีของ 11 เมืองในยุโรป

ตารางที่ 4.2 ประสิทธิภาพของแบบจำลองในการพยากรณ์ระบบเครือข่าย 30 นาทีของ 11 เมืองในยุโรป

Model	MAE	MAPE	MSE	RMSE	R ²
SMA	1.79×10^{10}	16.65	6.93×10^{20}	2.63×10^{10}	82.54%
ARIMA	4.86×10^9	72.02	4.21×10^{19}	6.49×10^9	98.71%
LSTM	4.28×10^9	5.23	3.10×10^{19}	5.57×10^9	99.05%
ANN	3.89×10^9	4.22	2.79×10^{19}	5.28×10^9	99.15%



รูปที่ 4.2 กราฟแสดงการเปรียบเทียบการพยากรณ์ระบบเครือข่าย 30 นาทีของ 11 เมืองในยุโรป

4.3 ผลการทดลองข้อมูลปริมาณการจราจรเฉลี่ยขาเข้าภายในระบบเครือข่ายมหาวิทยาลัย

4.3.1 ผลการทดลองของข้อมูล 5 นาที

ผลที่ได้จากการพยากรณ์ข้อมูล 5 นาทีของปริมาณการจราจรเฉลี่ยขาเข้าภายในมหาวิทยาลัยโดยที่ผลการประเมินประสิทธิภาพแบบจำลองจะแสดงดังตารางที่ 4.3 และกราฟแสดงการเปรียบเทียบแต่ละแบบจำลองจะแสดงดังรูปที่ 4.3 ดังรูปด้านล่าง โดยการพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่องระหว่างโครงข่ายประสาทเทียมจะฝึกฝนข้อมูล 200 รอบและหน่วยความจำระยะสั้นยาวจะฝึกฝนข้อมูล 100 รอบ ซึ่งทั้งสองแบบจำลองจะมีโครงสร้างที่ใช้ฝึกฝนและพยากรณ์ที่มีชั้นซ่อน 2 ชั้นแต่ละชั้นมี 128 นิวรอนและใช้ ReLU เป็นฟังก์ชันในการตัดสินใจ ส่วนที่ชั้นเอาต์พุตจะมี 1 ชั้นภายในชั้นมี 1 นิวรอนและใช้ linear เป็นฟังก์ชันในการตัดสินใจ ผลที่ได้จากโครงข่ายประสาทเทียมพบว่ามีค่า MAE=3.59×10⁶, ค่า MAPE=15.11, ค่า MSE=3.06×10¹³, ค่า RMSE=5.53×10⁶ และค่า R²=81.17% และผลที่ได้จากหน่วยความจำระยะสั้นยาวพบว่ามีค่า MAE=3.63×10⁶, ค่า MAPE=15.37, ค่า MSE=3.45×10¹³, ค่า RMSE=5.87×10⁶ และค่า R²=78.78% สำหรับการพยากรณ์ด้วยวิธีทางสถิติด้วยค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติจะฝึกฝนและพยากรณ์ตามข้อมูลที่ได้ถูกแบ่งไว้ซึ่งจะดำเนินการด้วย order(3,0,1) ผลที่ได้มีค่า MAE=3.67×10⁶, ค่า MAPE=79.52, ค่า MSE=3.29×10¹³, ค่า RMSE=5.74×10⁶ และค่า R²=79.80% และการพยากรณ์ด้วยค่าเฉลี่ยเคลื่อนที่อย่างง่ายผลที่ได้พบว่ามีค่า MAE=4.83×10⁶, ค่า MAPE=16.53, ค่า MSE=5.49×10¹³, ค่า RMSE=7.41×10⁶ และค่า R²=78.38% เมื่อพิจารณาจากผลการประเมินประสิทธิภาพและกราฟการเปรียบเทียบของแต่ละแบบจำลองพบว่าวิธีการเรียนรู้ของเครื่องยังประสบ

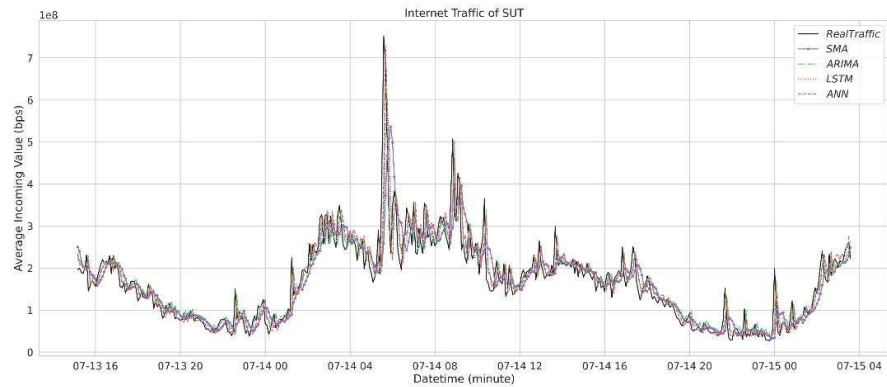
ความสำเร็จในการพยากรณ์ที่ดีกว่าเนื่องจากที่ค่าความผิดพลาดที่น้อยกว่าวิธีการทางสถิติ และเมื่อพิจารณาที่กราฟการเปรียบเทียบพบว่าวิธีการเรียนรู้ของเครื่องมีค่าเข้าใกล้ข้อมูลการใช้งานจริง

4.3.2 ผลการทดลองของข้อมูล 30 นาที

สำหรับการพยากรณ์ข้อมูล 30 นาทีของค่าเฉลี่ยขาเข้าที่ผ่านมาจากวิธีการเรียนรู้ของเครื่องและวิธีการทางสถิติ ผลที่การประเมินแบบจำลองจะแสดงดังตารางที่ 4.4 และกราฟแสดงผลเปรียบเทียบจะแสดงดังรูปที่ 4.4 ซึ่งการพยากรณ์ด้วยโครงข่ายประสาทเทียมจะฝึกฝนข้อมูล 200 รอบ มีจำนวนชั้นซ่อน 1 ชั้นซึ่งมี 64 นิวรอนและนำฟังก์ชัน ReLU เป็นฟังก์ชันตัดสินใจ และที่ชั้นเอาต์พุตประกอบด้วย 1 ชั้นซึ่งมี 1 นิวรอนและใช้ linear เป็นฟังก์ชันการตัดสินใจ ผลที่ได้พบว่ามีค่า MAE=4.19×10⁶, ค่า MAPE=17.46, ค่า MSE=3.85×10¹³, ค่า RMSE=6.20×10⁶ และ R²=72% ส่วนการพยากรณ์ด้วยหน่วยความจำระยะสั้นยาวที่ฝึกฝนข้อมูล 100 รอบ มีจำนวนชั้นซ่อน 2 ชั้นแต่ละชั้นมี 64 นิวรอนและนำ ReLU เป็นฟังก์ชันการตัดสินใจ ณ ที่ชั้นเอาต์พุตประกอบด้วย 1 ชั้นซึ่งมี 1 นิวรอนและใช้ linear เป็นฟังก์ชันการตัดสินใจ ผลจากการทดลองพบว่ามีค่า MAE=3.97×10⁶, ค่า MAPE=18.15, ค่า MSE=3.34×10¹³, ค่า RMSE=5.78×10⁶ และ R²=75.41% ต่อมาเป็นการพยากรณ์ด้วยค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติจะฝึกฝนและพยากรณ์ด้วย order(1,0,1) ผลจากการทดลองมีค่า MAE=4.84×10⁶, ค่า MAPE=74.03, ค่า MSE=5.38×10¹³, ค่า RMSE=7.33×10⁶ และค่า R²=65% และผลที่ได้จากการพยากรณ์ค่าเฉลี่ยเคลื่อนที่อย่างง่ายมีค่า MAE=7.07×10⁶, ค่า MAPE=25.51, ค่า MSE=9.45×10¹³, ค่า RMSE=9.72×10⁶ และค่า R²=58.14% เมื่อพิจารณาจากผลการทดลองและกราฟเปรียบเทียบว่าการพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่องมีความสามารถที่ดีกว่าและมีข้อผิดพลาดที่น้อยกว่าวิธีการทางสถิติ

ตารางที่ 4.3 ประสิทธิภาพของแบบจำลองในการพยากรณ์ระบบเครือข่ายเฉลี่ยขาเข้า 5 นาที
ภายในมหาวิทยาลัย

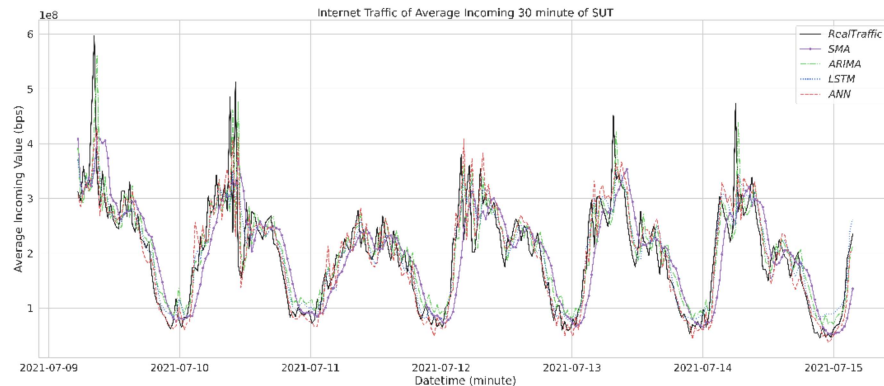
Model	Average Incoming_5 min				
	MAE	MAPE	MSE	RMSE	R ²
SMA	4.83×10 ⁶	16.53	5.49×10 ¹³	7.41×10 ⁶	78.38%
ARIMA	3.67×10 ⁶	79.52	3.29×10 ¹³	5.74×10 ⁶	79.80%
LSTM	3.63×10 ⁶	15.37	3.45×10 ¹³	5.87×10 ⁶	78.78%
ANN	3.59×10 ⁶	15.11	3.06×10 ¹³	5.53×10 ⁶	81.17%



รูปที่ 4.3 กราฟแสดงการเปรียบเทียบการพยากรณ์ระบบเครือข่ายเฉลี่ยขาเข้า 5 นาที
ภายในมหาวิทยาลัย

ตารางที่ 4.4 ประสิทธิภาพของแบบจำลองในการพยากรณ์ระบบเครือข่ายเฉลี่ยขาเข้า 30 นาที
ภายในมหาวิทยาลัย

Model	Average Incoming_30 min				
	MAE	MAPE	MSE	RMSE	R ²
SMA	7.07×10^6	25.51	9.45×10^{13}	9.72×10^6	58.14%
ARIMA	4.84×10^6	74.03	5.38×10^{13}	7.33×10^6	65%
LSTM	3.97×10^6	18.15	3.34×10^{13}	5.78×10^6	75.41%
ANN	4.19×10^6	17.46	3.85×10^{13}	6.20×10^6	72%



รูปที่ 4.4 กราฟแสดงการเปรียบเทียบการพยากรณ์ระบบเครือข่ายเฉลี่ยขาเข้า 30 นาที ภายในมหาวิทยาลัย

4.4 ผลการทดลองข้อมูลปริมาณการจราจรเฉลี่ยขาออกภายในระบบเครือข่าย มหาวิทยาลัย

4.4.1 ผลการทดลองของข้อมูล 5 นาที

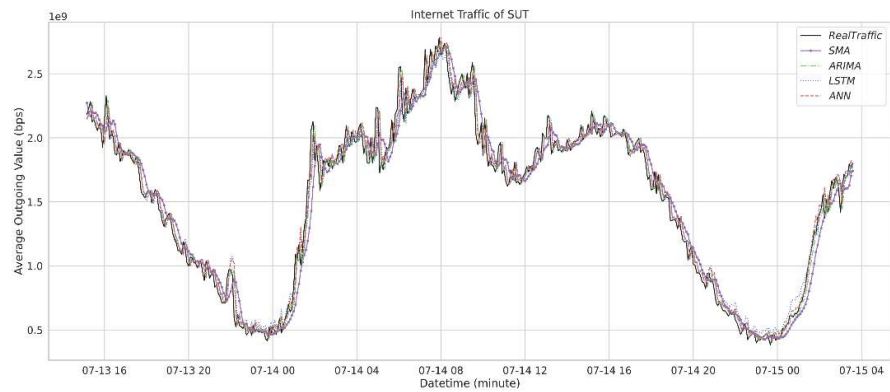
ผลการทดลองจากวิธีการเรียนรู้ของเครื่องและวิธีการทางสถิติซึ่งประเมินถึงประสิทธิภาพและข้อผิดพลาดของแบบจำลองจะแสดงดังตารางที่ 4.5 และกราฟการเปรียบเทียบผล จะแสดงดังรูปที่ 4.5 ด้านล่าง โดยที่การพยากรณ์ด้วยโครงข่ายประสาทเทียมจะฝึกฝนข้อมูล 200 รอบและการพยากรณ์ด้วยหน่วยความจำระยะสั้นยาวจะฝึกฝนข้อมูล 100 รอบ ซึ่งทั้งสองแบบจำลองจะมีชั้นซ่อน 2 ชั้นแต่ละชั้นมี 128 นิวรอนและใช้ ReLU เป็นฟังก์ชันในการตัดสินใจ ส่วนที่ชั้นเอาต์พุตจะมีเพียง 1 ชั้นละ 1 นิวรอนโดยที่ใช้ linear เป็นฟังก์ชันการตัดสินใจ ผลที่ได้จากการพยากรณ์ด้วยโครงข่ายประสาทเทียมพบว่ามีค่า MAE=9.03×10⁶, ค่า MAPE=4.76, ค่า MSE=1.53×10¹⁴, ค่า RMSE=1.24×10⁷ และค่า R²=98% และผลที่ได้จากหน่วยความจำระยะสั้นยาวมีค่า MAE=9.83×10⁶, ค่า MAPE=5.53, ค่า MSE=1.78×10¹⁴, ค่า RMSE=1.33×10⁷ และค่า R²=97.67% สำหรับการพยากรณ์ด้วยค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติจะฝึกฝนและพยากรณ์ด้วย order(3,0,0) ซึ่งผลที่ได้พบว่ามีค่า MAE=9.11×10⁶, ค่า MAPE=68.32, ค่า MSE=1.56×10¹⁴, ค่า RMSE=1.25×10⁷ และค่า R²=97.95% แลการพยากรณ์ด้วยค่าเฉลี่ยเคลื่อนที่อย่างง่ายผลที่ได้พบว่ามีค่า MAE=1.46×10⁷, ค่า MAPE=6.43, ค่า MSE=3.72×10¹⁴, ค่า RMSE=1.93×10⁷ และค่า R²=97.23% ซึ่งเมื่อพิจารณาผลจากการประเมินประสิทธิภาพและข้อผิดพลาด รวมทั้งกราฟเปรียบเทียบผลพบว่าการพยากรณ์ด้วยการเรียนรู้ของเครื่องมีประสิทธิภาพที่ดีกว่าและผลออกมามีค่าใกล้เคียงกับข้อมูลการใช้งานจริง

4.4.2 ผลการทดลองของข้อมูล 30 นาที

ผลการทดลองข้อมูล 30 นาทีของค่าเฉลี่ยขาออกด้วยการพยากรณ์วิธีการเรียนรู้ของเครื่องและวิธีการทางสถิติจะแสดงผลการประเมินแบบจำลองและข้อผิดพลาดของแบบจำลองดังตารางที่ 4.6 อีกทั้งแสดงกราฟเพื่อเปรียบเทียบผลของแต่ละแบบจำลองดังรูปที่ 4.6 ด้านล่าง ซึ่งการพยากรณ์ด้วยโครงข่ายประสาทเทียมจะฝึกฝนข้อมูลจำนวน 200 รอบมีชั้นซ่อน 2 ชั้นแต่ละชั้นมี 64 นิวรอนและใช้ ReLU เป็นฟังก์ชันตัดสินใจ ส่วนที่ชั้นเอาต์พุตมี 1 ชั้นภายในมี 1 นิวรอนและใช้ linear เป็นฟังก์ชันตัดสินใจ ผลที่ได้พบว่ามีค่า MAE= 1.22×10^7 , ค่า MAPE=6.33, ค่า MSE= 2.7×10^{14} , ค่า RMSE= 1.64×10^7 และค่า $R^2=96.43\%$ สำหรับการพยากรณ์ด้วยหน่วยความจำระยะสั้นยาวด้วยการฝึกฝนข้อมูลจำนวน 100 รอบ ที่ชั้นซ่อนมี 1 ชั้นภายในชั้นมี 64 นิวรอนและมีฟังก์ชัน ReLU เป็นฟังก์ชันตัดสินใจ ส่วนที่ชั้นเอาต์พุตมี 1 ชั้นภายในชั้นมี 1 นิวรอนและมีฟังก์ชัน linear เป็นฟังก์ชันตัดสินใจ ผลที่ได้พบว่ามีค่า MAE= 1.31×10^7 , ค่า MAPE=6.6, ค่า MSE= 3.34×10^{14} , ค่า RMSE= 1.83×10^7 และค่า $R^2=95.6\%$ สำหรับการพยากรณ์ด้วยค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติจะฝึกฝนและพยากรณ์ด้วย order(1,0,1) ผลที่ได้มีค่า MAE= 1.47×10^7 , ค่า MAPE=67.34, ค่า MSE= 3.60×10^{14} , ค่า RMSE= 1.89×10^7 และค่า $R^2=95.2\%$ และการพยากรณ์ด้วยค่าเฉลี่ยเคลื่อนที่อย่างง่ายผลที่ได้พบว่ามีค่า MAE= 4.86×10^7 , ค่า MAPE=23.86, ค่า MSE= 3.38×10^{15} , ค่า RMSE= 5.81×10^7 และค่า $R^2=74.60\%$ เมื่อพิจารณาจากผลที่ได้พบว่าการพยากรณ์ด้วยการเรียนรู้ของเครื่องก็ยังมีประสิทธิภาพในการพยากรณ์ที่ดีกว่าและผลที่พยากรณ์ได้มีค่าเข้าใกล้หรือใกล้เคียงกับข้อมูลการใช้งานจริง

ตารางที่ 4.5 ประสิทธิภาพของแบบจำลองในการพยากรณ์ระบบเครือข่ายเฉลี่ยขาออก 5 นาทีภายในมหาวิทยาลัย

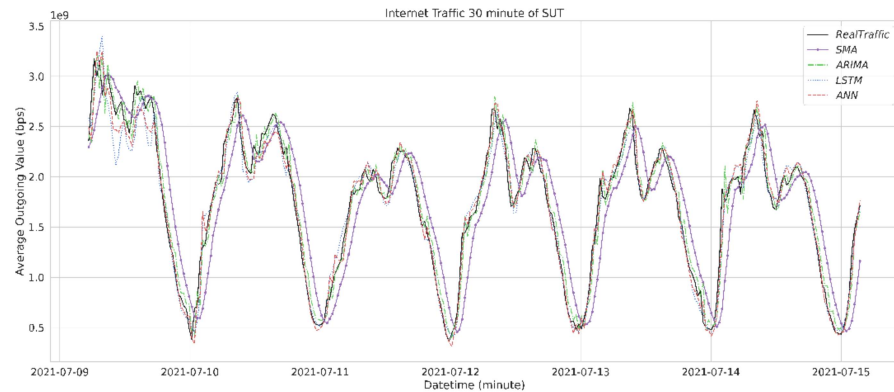
Model	Average Outgoing_5 min				
	MAE	MAPE	MSE	RMSE	R ²
SMA	1.46×10^7	6.43	3.72×10^{14}	1.93×10^7	97.23%
ARIMA	9.11×10^6	68.32	1.56×10^{14}	1.25×10^7	97.95%
LSTM	9.83×10^6	5.35	1.78×10^{14}	1.33×10^7	97.67%
ANN	9.03×10^6	4.76	1.53×10^{14}	1.24×10^7	98%



รูปที่ 4.5 กราฟแสดงการเปรียบเทียบการพยากรณ์ระบบเครือข่ายเฉลี่ยขาออก 5 นาที
ภายในมหาวิทยาลัย

ตารางที่ 4.6 ประสิทธิภาพของแบบจำลองในการพยากรณ์ระบบเครือข่ายเฉลี่ยขาออก 30 นาที
ภายในมหาวิทยาลัย

Model	Average Outgoing_30 min				
	MAE	MAPE	MSE	RMSE	R ²
SMA	4.86×10^7	23.86	3.38×10^{15}	5.81×10^7	74.60%
ARIMA	1.47×10^7	67.34	3.60×10^{14}	1.89×10^7	95.2%
LSTM	1.31×10^7	6.6	3.34×10^{14}	1.83×10^7	95.6%
ANN	1.22×10^7	6.33	2.7×10^{14}	1.64×10^7	96.43%



รูปที่ 4.6 กราฟแสดงการเปรียบเทียบการพยากรณ์ระบบเครือข่ายเฉลี่ยขาออก 30 นาที ภายในมหาวิทยาลัย

4.5 อภิปรายผลการทดลอง

จากการพยากรณ์ชุดข้อมูลทั้งหมดจากทั้งสี่แบบจำลองระหว่างที่ดำเนินการศึกษาและพยากรณ์พบว่าขณะที่พยากรณ์ข้อมูลด้วยวิธีการเรียนรู้ของเครื่องผลของจำนวนนิรอน, จำนวนรอบในการฝึกฝนและจำนวนของชั้นซ่อน จะส่งผลต่อแบบจำลองซึ่งหากเพิ่มหรือลดจำนวนมากเกินไปอาจทำให้เกิด Overfitting คือการที่แบบจำลองสามารถเรียนรู้จากชุดฝึกฝนและให้ผลออกมาได้ถูกต้องและแม่นยำ แต่เมื่อนำไปทดสอบกับชุดทดสอบกลับแสดงผลค่าความถูกต้องและความแม่นยำต่ำหรืออาจส่งผลให้ผลที่ได้มีค่าความผิดพลาดมากยิ่งขึ้น ฉะนั้นจึงจำเป็นต้องปรับพารามิเตอร์แต่ละตัวให้พอดีกับชุดข้อมูลและแบบจำลอง และจากการพยากรณ์พบว่าวิธีการเรียนรู้ของเครื่องมีความสามารถในการพยากรณ์ที่ดีกว่าวิธีการทางสถิติเนื่องจากได้มีการนำชุดข้อมูลมาฝึกฝนหรือนำข้อมูลก่อนหน้ามาเรียนรู้ซ้ำเพื่อให้ได้มาซึ่งแบบจำลองที่ดี จากผลการทดลองพบว่ามีค่าความผิดพลาดที่ต่ำกว่าและที่ R^2 มีค่าเข้าใกล้ 1 หรือ 100% จะหมายความว่าแบบจำลองมีประสิทธิภาพในการพยากรณ์ที่แม่นยำ อีกทั้งเมื่อความยาวของชุดข้อมูลมีขนาดใหญ่และข้อมูลมีความละเอียดก็ทำให้สามารถพยากรณ์ข้อมูลออกมาได้ดีกว่าวิธีการทางสถิติซึ่งสอดคล้องกับงานวิจัยของ (Azari, A. et al., 2019) และเมื่อพิจารณาที่โครงข่ายประสาทเทียมแบบจำลองจะเหมาะกับชุดข้อมูลที่เป็นแบบรูป เช่น ชุดข้อมูลการใช้งานเครือข่ายที่ช่วงวันจันทร์ถึงวันศุกร์จะมีปริมาณการจราจรค่อนข้างมาก แต่ถ้าหากหลัง 16.00 น. หรือวันเสาร์, อาทิตย์จะมีปริมาณการใช้งานค่อนข้างน้อย เป็นต้น และสำหรับจำนวนรอบในการฝึกฝนข้อมูลส่วนใหญ่ที่ส่งผลให้แสดงผลลัพธ์การพยากรณ์ที่ดีจะมีจำนวนรอบอยู่ที่ประมาณ 100-200 รอบ ซึ่งสอดคล้องกับการศึกษาของ (Odim, Mba Obasi et al., 2016) สำหรับหน่วยความจำระยะสั้นยาวจะเหมาะกับชุดข้อมูลที่ไม่มีรูปแบบตายตัวหรือไม่คงที่ เช่น ข้อมูลที่

มีการแกว่ง และจำนวนรอบที่ฝึกฝนข้อมูลที่ให้ผลดีจะอยู่ที่ 100 รอบ สำหรับการพยากรณ์ด้วยวิธีทางสถิติพบว่าแบบจำลองเหมาะกับการพยากรณ์ชุดข้อมูลที่มีขนาดเล็กซึ่งจะทำให้แบบจำลองมีประสิทธิภาพในการพยากรณ์ที่ดีกว่า และเมื่อพิจารณาเกี่ยวกับเวลาที่ใช้ในการพยากรณ์ของแต่ละแบบจำลองพบว่าแบบจำลองโครงข่ายประสาทเทียมใช้เวลาในการพยากรณ์ประมาณ 1-5 นาที, หน่วยความจำระยะสั้นยาวใช้เวลาในการพยากรณ์อยู่ที่ 2-40 นาที, ค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติใช้เวลาในการพยากรณ์อยู่ที่ 2 นาที-1 ชั่วโมง และการพยากรณ์ด้วยค่าเฉลี่ยเคลื่อนที่อย่างง่ายจะไม่มีเวลาในการพยากรณ์เนื่องจากทดสอบด้วยโปรแกรม Excel ซึ่งใช้สูตรในการคำนวณ ทั้งนี้จำนวนเวลาที่ใช้ในการพยากรณ์จะขึ้นอยู่กับปริมาณจำนวนข้อมูลที่ใช้, ความซับซ้อนของแต่ละแบบจำลอง และพบว่าโครงข่ายประสาทเทียมมีข้อได้เปรียบในเรื่องของเวลาที่ใช้ในการพยากรณ์ที่น้อยกว่าหน่วยความจำระยะสั้นยาว ณ จุดนี้โครงข่ายประสาทเทียมจึงเหมาะกับการพยากรณ์ข้อมูลบนระบบเครือข่ายเนื่องจากสามารถตอบสนองต่อการพยากรณ์แบบเรียลไทม์ได้ ซึ่งถ้าหากนำไปใช้กับระบบเครือข่ายจริงจำเป็นที่จะต้องใช้ความรวดเร็วในการพยากรณ์เพื่อที่จะได้จัดสรรระบบเครือข่ายให้ทันเวลาต่อความต้องการของผู้ใช้บริการ

4.6 กล่าวสรุป

เนื้อหาในบทนี้จะเป็นการกล่าวถึงผลที่ได้จากการพยากรณ์และนำไปวัดประสิทธิภาพและค่าความผิดพลาดเพื่อแสดงถึงแบบจำลองที่ดีที่สุดโดยใช้เครื่องมือวัดประสิทธิภาพของแบบจำลองในการประเมิน ผลที่ได้พบว่าวิธีการเรียนรู้ของเครื่องมีความสามารถและมีค่าความผิดพลาดที่น้อยกว่าวิธีทางสถิติ อีกทั้งเวลาที่ใช้ในการฝึกฝนและพยากรณ์ข้อมูลนั้นน้อยกว่า แต่ถ้าหากต้องนำแบบจำลองไปใช้กับระบบเครือข่ายจริงจำเป็นที่จะต้องการแบบจำลองที่มีการประมวลผลไว ซึ่งแบบจำลองโครงข่ายประสาทเทียมมีข้อได้เปรียบสำหรับเรื่องของการประมวลผล

บทที่ 5

สรุปการวิจัยและข้อเสนอแนะ

5.1 สรุปเนื้อหาของวิทยานิพนธ์

วิทยานิพนธ์ฉบับนี้ได้นำเสนอวิธีการพยากรณ์ข้อมูลปริมาณการใช้งานบนระบบเครือข่ายมาพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่องคือโครงข่ายประสาทเทียมและหน่วยความจำระยะสั้นยาวเปรียบเทียบกับการพยากรณ์ด้วยวิธีทางสถิติคือค่าเฉลี่ยเคลื่อนที่แบบบูรณาการถดถอยอัตโนมัติและค่าเฉลี่ยเคลื่อนที่อย่างง่าย ซึ่งในการพยากรณ์จะนำวิธีการเลื่อนหน้าต่างมาใช้ควบคู่ไปด้วยเพื่อรองรับต่อการใช้งานแบบเรียลไทม์ โดยจะนำข้อมูลปริมาณการจราจรบนระบบเครือข่ายมาดำเนินพยากรณ์ในงานวิจัยนี้จะนำข้อมูล 2 ชุดข้อมูลใหญ่มาดำเนินการทดสอบและพยากรณ์คือ ข้อมูลการจราจรบนระบบเครือข่ายที่ถูกรวบรวมจากผู้ให้บริการส่วนตัวจาก 11 เมืองในยุโรปซึ่งเป็นข้อมูลที่เปิดให้ดาวน์โหลดแบบสาธารณะบนอินเทอร์เน็ตและข้อมูลปริมาณการใช้งานระบบเครือข่ายภายในเครือข่ายมหาวิทยาลัยที่ได้รับการอนุเคราะห์จากศูนย์คอมพิวเตอร์ของมหาวิทยาลัยเทคโนโลยีสุรนารี จากนั้นจะนำชุดข้อมูลดังกล่าวมาจัดเตรียมเพื่อที่จะให้แบบจำลองได้ดำเนินการเรียนรู้หรือฝึกฝนแล้วจึงพยากรณ์ผลในอนาคตออกมา เมื่อพยากรณ์ข้อมูลเรียบร้อยแล้วจะนำผลที่ได้มาสร้างกราฟเปรียบเทียบระหว่างผลที่ได้จากการพยากรณ์ของแต่ละแบบจำลองเทียบกับข้อมูลการใช้งานจริง จากนั้นจึงดำเนินการวัดประสิทธิภาพและข้อผิดพลาดของแต่ละแบบจำลองด้วยเครื่องมือวัดประสิทธิภาพแบบจำลองเช่น ค่าเฉลี่ยความผิดพลาดร้อยละสัมบูรณ์, ค่าเฉลี่ยของค่าผิดพลาดกำลังสอง, ค่าสัมประสิทธิ์ R-Squared เป็นต้น เพื่อให้ได้มาซึ่งแบบจำลองที่มีประสิทธิภาพที่จะนำไปใช้ในการพยากรณ์เครือข่ายในอนาคต

จากผลลัพธ์ที่ได้จากการพยากรณ์ของแต่ละแบบจำลอง เมื่อนำมาวัดประสิทธิภาพและความแม่นยำพบว่าการพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่องมีความแม่นยำและมีประสิทธิภาพในการพยากรณ์ได้ดีกว่าแบบจำลองทางสถิติ เนื่องจากมีค่าความผิดพลาดที่ต่ำและมีผลด้านความแม่นยำที่ดีกว่าเมื่อทดสอบด้วยพารามิเตอร์ค่าสัมประสิทธิ์ R-Squared ที่มีค่าเข้าใกล้ 1 หรือ 100 เปอร์เซ็นต์ ซึ่งการพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่องพบว่าส่วนใหญ่การพยากรณ์ด้วยโครงข่ายประสาทเทียมจะมีประสิทธิภาพที่ดีกว่า รองลงมาคือการพยากรณ์ด้วยหน่วยความจำระยะสั้นยาว เนื่องจากการพยากรณ์ด้วยวิธีการเรียนรู้ของเครื่องเหมาะกับชุดข้อมูลที่มีลักษณะไม่เป็นเชิงเส้น แต่เมื่อได้ดำเนินการศึกษาและทดลองในงานวิจัยนี้พบว่าโครงข่ายประสาทเทียมจะเหมาะกับชุดข้อมูลที่ไม่เป็นเชิงเส้นที่เป็นแบบรูป ส่วนหน่วยความจำระยะสั้นยาวจะเหมาะกับข้อมูลที่ไม่เป็นเชิงเส้นที่ไม่เป็นแบบ

รูปตายตัวมากกว่าและเมื่อพิจารณาถึงระยะเวลาที่ใช้ในการฝึกฝนและพยากรณ์ สำหรับงานวิจัยนี้ที่มีวัตถุประสงค์หลักคือต้องการพยากรณ์ข้อมูลออกมาเป็นแบบเรียลไทม์เพื่อให้รองรับต่อการใช้งานของผู้ใช้บริการได้ทันทั่วถึงที่จึงพบว่าพยากรณ์ด้วยการใช้โครงข่ายประสาทเทียมมีความเหมาะสมกว่าเนื่องจากมีความเร็วในการพยากรณ์ที่รวดเร็วกว่าแบบจำลองอื่นๆ เพื่อที่จะได้จัดสรรระบบเครือข่ายหรือทรัพยากรให้ทันต่อความต้องการในการใช้งานของผู้ใช้บริการ และถึงแม้ว่าในการพยากรณ์ครั้งนี้จะเป็นการพยากรณ์ข้อมูลการใช้งานเครือข่ายไวไฟ (Wi-Fi) เป็นส่วนใหญ่ แต่ถ้าหากต้องการพยากรณ์ระบบเครือข่ายมือถือก็พบว่าสามารถนำวิธีการเรียนรู้ของเครื่องมาพยากรณ์ค่าการใช้งานข้อมูลมือถือในอนาคตได้เช่นกัน ซึ่งจะสอดคล้องกับงานวิจัยของ (Kirmaz, A. et al., 2020) และ (Azari, A. et al., 2019) ที่ได้นำเสนอไว้ว่าวิธีการเรียนรู้ของเครื่องสามารถพยากรณ์ข้อมูลการใช้งานมือถือได้อย่างมีประสิทธิภาพและแม่นยำ แต่อย่างไรก็ตามอาจจะต้องปรับพารามิเตอร์ภายในโครงสร้างของแบบจำลองเพื่อให้เหมาะสมกับชุดข้อมูลเพื่อผลลัพธ์ที่ดียิ่งขึ้น

5.2 ข้อเสนอแนะและแนวทางการพัฒนา

สำหรับการวิจัยในวิทยานิพนธ์ฉบับนี้จะเป็นการเปรียบเทียบประสิทธิภาพของแต่ละแบบจำลองเพื่อให้ได้แบบจำลองที่ดีที่สุดที่เหมาะสมต่อการพยากรณ์ข้อมูลปริมาณการจราจรบนระบบเครือข่ายในที่นี้คือแบบจำลองด้วยวิธีการเรียนรู้ของเครื่อง ดังนั้นหากมีการนำไปพัฒนาหรือใช้งานในอนาคตอาจจะต้องมีการเก็บข้อมูลที่เกี่ยวข้องกับการใช้งานบนระบบเครือข่ายเพิ่มเติมเพื่อเพิ่มความแม่นยำของการพยากรณ์ หรือถ้าหากมีข้อมูลชุดใหม่ที่ต้องการพยากรณ์ในแบบจำลอง อาจที่จะต้องปรับพารามิเตอร์ของแบบจำลองเพื่อให้เหมาะสมกับชุดข้อมูล อาทิเช่น จำนวนนิเวศน์ภาพในชั้นซ่อน จำนวนรอบในการฝึกฝนข้อมูลหรือจำนวนชั้นภายในชั้นซ่อน เป็นต้น หรืออาจจะนำแบบจำลองไปใช้ควบคู่กับการควบคุมระบบแบบอัตโนมัติเช่น ใช้งานกับ SDN (Software-defined network) ซึ่งเป็นซอฟต์แวร์ที่ใช้ควบคุมอุปกรณ์ต่างๆในงานด้านโทรคมนาคม อาทิเช่น เราเตอร์ ซึ่งสำหรับการนำไปใช้งานนั้นอาจจะนำแบบจำลองพยากรณ์มาพยากรณ์ข้อมูลการใช้งานเครือข่ายในอนาคตออกมาและดูแนวโน้มการใช้งานในอนาคตว่ามีการใช้งานมากหรือน้อย จากนั้นจึงให้ซอฟต์แวร์เป็นตัวสั่งการหรือควบคุมอุปกรณ์ว่าควรที่จะเปิดใช้งานเพิ่มหรือปิดการใช้งาน เป็นต้น และสามารถนำแบบจำลองไปพัฒนาต่อเพื่อวิเคราะห์ถึงความผิดปกติที่จะเกิดขึ้นในระบบเครือข่ายได้ ซึ่งจะทำให้ผู้ใช้บริการระบบเครือข่ายสามารถจัดสรรทรัพยากรและควบคุมระบบให้มีประสิทธิภาพ อีกทั้งลดการใช้ทรัพยากรที่อาจทำให้เกิดการสิ้นเปลือง และสามารถเรียกความน่าเชื่อถือและความเชื่อมั่นจากลูกค้าหรือผู้ใช้งานในอนาคต

รายการอ้างอิง

- กอบเกียรติ สระอุบล. (2565). เรียนรู้ AI: Deep Learning ด้วย Python (พิมพ์ครั้งที่ 1). กรุงเทพฯ: หสม สำนักพิมพ์ อินเทอร์เน็ตมีเดีย.
- บัญชา ปะสีละเตสัง. (2564). สร้างการเรียนรู้สำหรับ AI ด้วย Python Machine Learning. กรุงเทพฯ: ซีเอ็ดยูเคชั่น.
- Laurence Moroney. (2564). AI and Machine Learning for Coders (วิโรจน์ อัครวงษ์สี) (พิมพ์ครั้งที่ 1). นนทบุรี: หจก.ต๋อแก้วเมชั่น.
- ปริญญา สงวนสัตย์. (2562). Artificial Intelligence with Machine Learning, AI สร้างได้ด้วยแมชชีน เลิร์นนิ่ง Python Edition. นนทบุรี: ไอดีซี พรีเมียร์.
- อนุสรฯ หิรัญวานากุล. (2563). เทคนิคการเพิ่มประสิทธิภาพการคาดการณ์ผลผลิตในอุตสาหกรรมการผลิตฮาร์ดดิสก์ไดรฟ์ (ปริญญาณิพนธ์ปริญาคุณภูษิต). นครราชสีมา: มหาวิทยาลัยเทคโนโลยีสุรนารี.
- กรินทร์ กาญจนานนท์. (2561). การพยากรณ์ทางสถิติ. กรุงเทพฯ: ซีเอ็ดยูเคชั่น.
- Yaffee, R. & McGee, M. (2000). An Introduction to Time Series Analysis and Forecasting. Cambridge: Academic Press.
- Vijay Kotu & Bala Deshpande. (2019). Chapter 12 - Time Series Forecasting. Data Science (Second Edition) (p. 395-445). Morgan Kaufmann.
- Oduro-Gyimah, F. K., & Boateng, K. O. (2018). A Comparative Analysis of Telecommunication Network Traffic Forecasting: A Three Model Approach. 2018 IEEE 7th International Conference on Adaptive Science & Technology (ICAST). 1-6.
- Lu, S., Zhang, Q., Chen, G., & Seng, D. (2021). A combine method for short-term traffic flow prediction based on recurrent neural network. Alexandria Engineering Journal, vol.60, no.1, 87-94.
- K, Sultan., H, Ali., & Z, Zhang. (2018). Call Detail Records Driven Anomaly Detection and Traffic Prediction in Mobile Cellular Networks. in IEEE Access, vol. 6, 41728-41737.
- Carlberg, C. (2013). Predictive Analytics: Microsoft Excel. Indianapolis: QUE.

- Chiulli, R.M. (1999). *Quantitative Analysis : An Introduction*. Amsterdam: Gordon and Breach Science.
- สุชาติ คุ่มมะณี. (2565). การเขียนโปรแกรมไพธอนสำหรับระบบเครือข่ายและความปลอดภัย (พิมพ์ครั้งที่ 1). ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาการสารสนเทศ, มหาวิทยาลัยมหาสารคาม.
- โชติพันธุ์ หล่อเลิศสุนทร และ จิตะพันธุ์ หล่อเลิศสุนทร. (2562). คู่มือเรียน เขียนโปรแกรม Python (ภาคปฏิบัติ) (พิมพ์ครั้งที่ 1). คอร์ฟังก์ชั่น.
- Aldhyani, T. H., & Joshi, M. R. (2017). Enhancement of Single Moving Average Time Series Model Using Rough k-Means for Prediction of Network Traffic. *International Journal of Engineering Research and Application*, vol. 7, 45-51.
- Purnawansyah, P., Haviluddin, H., Alfred, R., & Gaffar, A, F, O. (2018). Network traffic time series performance analysis using statistical methods. *Knowledge Engineering and Data Science (KEDS)*, vol. 1, no. 1, 1-7.
- Lv, T., Wu, Y., & Zhang, L. (2021). A Traffic Interval Prediction Method Based on ARIMA. in *Journal of Physics: Conference Series*, vol. 1880, no. 1, pp. 012031.
- Azari, A., Papapetrou, P., Denic, S., & Peters, G. (2019). Cellular traffic prediction and classification: A comparative evaluation of LSTM and ARIMA. *International Conference on Discovery Science*, 129-144.
- Zhou, K., Wang, W. Y., Hu, T., & Wu, C. H. (2020). Comparison of Time Series Forecasting Based on Statistical ARIMA Model and LSTM with Attention Mechanism. *Journal of Physics: Conference Series*, vol. 1631, no. 1, pp. 012141.
- Oliveira, T. P., Barbar, J. S., & Soares, A. S. (2016). Computer network traffic prediction: a comparison between traditional and deep learning neural networks. *International Journal of Big Data Intelligence*, vol.3, no. 1, 28-37.
- Do, Q. H., Doan, T. T. H., Nguyen, T. V. A., Duong, N. T. & Linh, V. V. (2020). Prediction of Data Traffic in Telecom Networks based on Deep Neural Networks. *Journal of Computer Science*, vol. 16, no. 9, 1268–1277.
- Kirmaz, A., Michalopoulos, D. S., Balan, I., & Gerstacker, W. (2020). Mobile Network Traffic Forecasting Using Artificial Neural Networks. *2020 28th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, 1-7.

- Ciptaningtyas, H. T., Fatichah, C., & Sabila, A. (2017). Network traffic anomaly prediction using Artificial Neural Network. AIP Conference Proceedings, vol. 1818, no. 1, pp. 020010.
- Cortez, P.; Rio, M.; Rocha, M. and Sousa, P. (2012). Multi-scale Internet traffic forecasting using neural networks and time series methods. Expert Systems, vol. 29, no. 2, 143-155.
- Odim, M. O., Gbadeyan, J., & Sadiku, J. (2016). Modelling the Multi-Layer Artificial Neural Network for Internet Traffic Forecasting: The Model Selection Design Issues. *OcRI*.
- COGNUB An Innovation Incubator Company. (2022). COGNITIVE COMPUTING AND MACHINE LEARNING. Retrieved May 16, 2022, from <https://www.cognub.com/index.php/cognitive-platform/>
- Giraffe Magazine. (2016). จับตาเกมโกะหยุดโลก อะไรกันแน่ที่ทำให้ ALPHA GP ชนมนุษย์?!. สืบค้นเมื่อ 16 พฤษภาคม 2565, จาก <https://minimore.com/f/Lab33-745>
- Dhaval Patel. (2018). Insights Of The Machine Learning And The Deep Learning. Retrieved May 16, 2022, from <https://blog.thinkwik.com/insights-of-the-machine-learning-and-the-deep-learning/>
- Library, Physiology. (2016). การทำงานของระบบประสาท. สืบค้นเมื่อ 16 พฤษภาคม 2565, จาก <http://www.idoctorhouse.com/library/physio-neuro/>



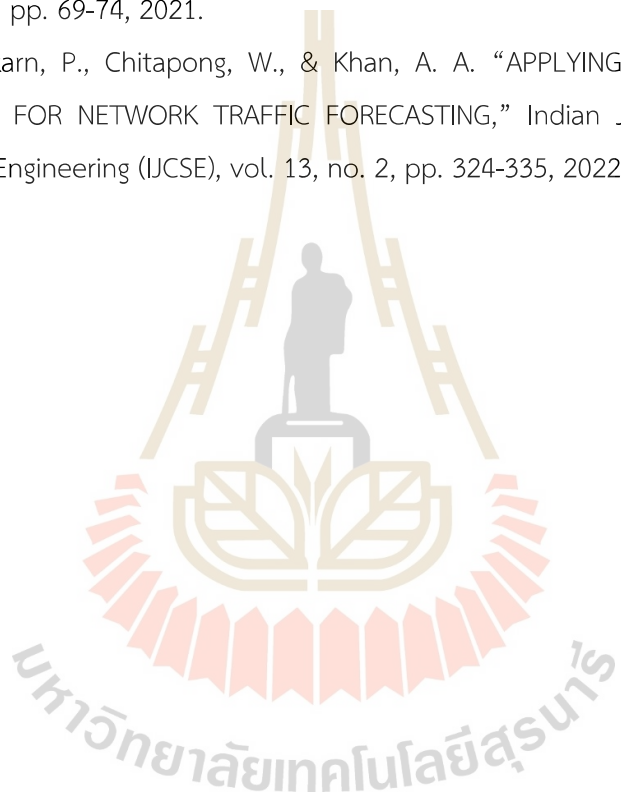
ภาคผนวก ก

บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างการศึกษา

รายชื่อบทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างศึกษา

Supakarn, P., and Chitapong, W. “EVALUATION OF COMPUTER NETWORK DATA TRAFFIC FORECASTING METHODS,” In The 15th South East Asian Technical University Consortium (SEATUC) Symposium 2021: Virtual Online Symposium, Institut Teknologi Bandung(ITB), pp. 69-74, 2021.

Supakarn, P., Chitapong, W., & Khan, A. A. “APPLYING MACHINE LEARNING APPROACHES FOR NETWORK TRAFFIC FORECASTING,” Indian Journal of Computer Science and Engineering (IJCSE), vol. 13, no. 2, pp. 324-335, 2022.



The 15th South East Asian Technical University Consortium (SEATUC 2021)
25th – 26th February 2021, Institut Teknologi Bandung, Indonesia

EVALUATION OF COMPUTER NETWORK DATA TRAFFIC FORECASTING METHODS

*Supakarn Prajam, Chitapong Wechtaisong
School of Telecommunication Engineering, Institute of Engineering, Suranaree
University of Technology, Muang District, Nakhon Ratchasima, 30000, Thailand
*Corresponding author: M6202460@g.sut.ac.th

ABSTRACT

At the present day, people frequently access the internet, and the traffic data are rapidly increased. Due to the resources on the internet are limited, the networks are not efficiently able to respond to applications for service users. In this paper, there is network traffic forecasting to predict future traffic usage by comparing Simple Moving Average (SMA) statistics, Artificial Neural Network (ANN) algorithm, and Long short-term memory (LSTM) algorithm for accurate forecast ability. Five sliding windows are forecasted to predict future values using the One-way ANOVA statistics to measure the effectiveness of the forecast from all three methods. The results showed that ANN was the highest accurate forecast ability.

Keywords: *Forecasting, Data Traffic, Simple Moving Average, Artificial Neural Network, Long short-term memory, One-way ANOVA.*

1. INTRODUCTION

In the digital age, communication through online media is integrated with our daily life, such as using real-time applications on the internet, live on Facebook, streaming games, video conferences, etc. Therefore, they increase the number of people using the internet and affect high network traffic. As the use of traffic on the network increases causing the insufficient allocation of resources to users, causing congestion in data transmission, internet interruption problems, and cannot be used. In order to solve these problems, the internet service provider should have to know the information of the users' internet traffic in the future to allocate bandwidth efficiently by using the method of forecasting traffic on the network.

Several studies have found to study and forecast traffic by using the LSTM algorithm (WANG et al. 2018), which uses a neural network called LSTM to predict the cost of non-linear. Moreover, research that forecasting computer network traffic (Madan and Sarathimangipudi. 2018) using Discrete Wavelet Transform (DWT), Auto Regressive Integrated Moving Average (ARIMA), and Recurrent Neural Network (RNN) as predictive tools. For our research work, we aim to evaluate the predictive ability between a statistic

and neural networks, SMA, LSTM, and ANN, by assessing which model is more accurate.

1.1 LITERATURE SURVEY

There is various research work focus on the forecast by forecasting network traffic which used the LSTM and LSTM-DNN algorithms that optimized from autocorrelation features (WANG et al. 2019), which presented that LSTM had better data prediction performance at different time resolutions, and LSTM-DNN was better at predicting data with a coarse time resolution such as one hour but not suitable for forecasting large volumes of data. Subsequently, computer network traffic was predicted using DWT, ARIMA, and ANN (Madan and Sarathimangipudi. 2018). DWT performed linear and non-linear separation. Then, used ARIMA and RNN for forecasting, the results showed that RNN predicted traffic better than the ARIMA model.

On the other hand, there were not only network traffic forecasts but also stock prices forecasting using ANN algorithm, LSTM algorithm, and ARIMA is a statistical forecast (Dhenuvakonda, Anandan, and Kumar 2020), or even predicting stock prices under different stability. The comparison between LSTM and ARIMA (Qian and Chen 2019) found that results obtained by ANN and LSTM had better forecasting accuracy than ARIMA. A research work used the neural network to predict the number of bus passengers using ANN compared to non-linear regression (Yu et al. 2016), ANN had a better forecasting capability. The stock price forecast was studied, but the statistics used SMA by news classification (Lauren and Harlili 2015). The forecast data divided between 5, 10, 20, and 50 days. The results obtained when combination data with a 5-day and 20-day period showed a positive buy-in response. Thus, there is a high chance that the trader will receive a profit. In addition, the application of SMA in predictions with Linear regression and Multilayer Perceptron (Chantarakasemchit et al. 2020) to calculate historical forex data. It uses an SMA with periods equal to 3, 5, and 7 days and divides the resulting data into four types: forex dataset, forex with factors, forex with SMA, and forex with factors and SMA. As a result, the linear regression predicted forex with factors and SMA better,

and the multilayer perceptron predicts the forex dataset better than the others due to the lower MSE mean. These research works lead us to evaluate traffic forecasting from average incoming and average outgoing traffic direction. We evaluated three prediction schemes, namely, SMA, LSTM, and ANN. Then, we analyzed an appropriate method for the internet traffic.

2. BACKGROUND THEORY

To forecast the internet traffic, we used a statistical forecast that were Simple Moving Average and Machine Learning, including Artificial Neural Network algorithm and Long short-term memory. The forecasting results from the three methods were analyzed for their accuracy. The analysis tool was One-way ANOVA to find an appropriate forecasting method.

2.1 Simple Moving Average

Simple Moving Average (SMA) is the statistical forecasting that is a part of forecasting using a moving average. It is a form of time series forecasting using the number of historical data values to calculate the mean to forecast future values (Chiulli and R.M. 1999). The concept of this principle is the weighting of historical data at each time equally. It depends on giving weight to how much historical data (Yaffee et al., 2000). Therefore, the formula for calculating moving values is shown as the equation below.

$$MA_{(k)} = \frac{1}{k} \sum_{i=t-k+1}^t y_i \quad (1)$$

The values of moving average at t are using forecast the values at the time $t+1$ (\hat{y}_{t+1}) that is

$$\hat{y}_{t+1} = MA(k)_t \quad (2)$$

For choosing the appropriate k parameter in this paper we used trial and error method. We varied the k value from 1 to 5 and found that $k=5$ was the most suitable to forecasting network traffic of Suranaree University of Technology (SUT).

2.2 Long Short-Term Memory

Long Short-Term Memory (LSTM) is the neural network considered one of the Recurrent Neural Network (RNN) with increased memory cells state. Additionally, the LSTM is improved to provide better stability and performance than the traditional RNN in the problem of gradient values is less from the work of back-propagation. Operation of the LSTM is an assessment of situation each node and can to select which data should remember, or disposed of through the forgetting of the state in that the node. The internal architecture of the LSTM has special functions such as reading, writing, updating of data, and forgetting data. These functions like gates or unit cells that control the data coming into each node consist of forget gate, input gate, and output gate. Each gate has a different function. Forget gate is the gate that controls what data should be

stored or should be forgotten by making decisions based on incoming data at that node is combined with the results of the previous node computation through the sigmoid function. Then the results are multiplied to the previous state of the node using the pointwise multiplication operator. The results are 0 and 1, where 0 is that data has been forgotten from memory and 1 is stored in memory. Input gate functions accept the new data input. The output gate is the gate that decides. The output data is ready to exit memory or not. The calculations of gates follow the equations below (Qian and Chen 2019).

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (3)$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (4)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad (5)$$

$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \quad (6)$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \tilde{c}_t \quad (7)$$

$$h_t = o_t \circ \tanh(c_t) \quad (8)$$

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (9)$$

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (10)$$

Where x_t : input vector sends to the memory unit, f_t : forget gate, i_t : input gate, o_t : output gate, h_t : output value filtered by output gate, \tilde{c}_t : state of the memory cell, c_t : state of the current memory cell at time t , W and U : weight matrices, b : bias vector, σ : sigmoid function and LSTM the structure as shown (Le et al. 2019).

2.3 Artificial Neuron Network

Artificial Neuron Network (ANN) is the foundation of Artificial intelligence (AI). The characteristic of ANN is part of a computer system designed to simulate the pattern and structure is similar to that of the human brain. Neural networks are able to learn, performing the tasks assigned by learning through examples, which is supervised learning. The processing of the neural network takes place in the unit cells called the node. Inside the node has the function that defines the output signal that is called the transfer function. In the neural network model, there are three types of neurons: the input layer, output layer, and hidden layer. The connection between the node of the layer has the configuration of weighting values applied to every link. When started, it is configured value to the node input and then send the value and this value to be multiplied by the link weight of the link, and then the next node receives the value. The received value is the value calculated from the previous node. The result calculation such as calculation from the sigmoid function and send the value to the next layer. This kind

of calculation will continuously calculate until the output layer. The total input at the node i can be calculated from the equation below (Yu et al. 2016).

$$u_i = \sum_j w_{ij} x_j \quad (11)$$

$$v_i = u_i + \theta_i \quad (12)$$

Where x_j is the output value from the j knot, and w_{ij} is the weight value on the junction in the direction from node i to junction j , v_i is the adjusted value from the neuron i , θ_i is the threshold of neuron i and model. ANN's items are shown in Yu et al. 2016.

2.4 One-way ANOVA

One-way ANOVA, or one-way analysis of variance, is a test method for analyzing the relationship between the independent and the dependent variable. It tests the difference between the mean obtained from three or more groups. This type of analysis involves two variables: one independent variable, divided into three or more types (k type) and one dependent variable. In analyzing the variance of one factor, the main hypothesis H_0 and the secondary H_1 hypothesis must be made. If the variance analysis rejects H_0 , it needs to check if any of the pairs are different using the Multiple Comparison Test or Post hoc. The test is performed after the variance analysis has been completed. The statistics used in the 1-factor variance test that indicate whether the tested value has a significant difference (F) is shown in the equation below (Howard 2018).

Table 1. Analysis of variance 1 factor

Source of variation	df	Sum of Squares	Mean Square	F
Between groups	$k-1$	SS_b	$MS_b = SS_b / k-1$	$\frac{MS_b}{MS_w}$
Within groups	$N-k$	SS_w	$MS_w = SS_w / N-k$	
Total	$N-1$	SS_t		

3. METHODOLOGY

To forecast internet traffic usage, forecasting is performed from three methods: SMA Statistic, LSTM Algorithm, and ANN Algorithm. The prediction data were average incoming and average outgoing. The first method uses the SMA statistic forecast from equations (1) and (2), which uses $k = 5$ to predict future traffic by excel to calculate. There were forecasting using LSTM and ANN algorithms written in Python language on Google Colaboratory, which ran the program on GPU.

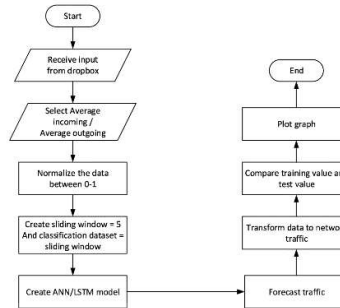


Fig. 1. The LSTM and ANN traffic forecasting workflow

Figure.1 presents the forecasting workflow. Firstly, the program received traffic data from the computer center service of Suranaree University of Technology stored on the cloud called Dropbox. The average incoming or average outgoing column was selected as the traffic data. Then the data was normalized the value between 0-1 before entering the model. After that, the sliding window was created with a value of five to divide the data into the model by five values. The future value was predicted by sending the data in the sliding window into the train according to the number of epochs specified in the LSTM model or ANN model. After that, the model performed a forecast and converted the forecasted data from the normalization format to normalize network traffic. Then, the forecasted value was compared with the actual value, and show the comparison results in the form of a graph.

From these three predictions, the forecasted average incoming and average outgoing values were used to find the percentage of mean error and standard deviation relative to the real value. As a result, the error was applied to statistic investigation using One-way ANOVA to find the most accurate predicting method for internet traffic. The results will be shown in the next section.

4. RESULTS AND DISCUSSION

Table 2. The percentage of average error and standard deviation.

	SMA		ANN		LSTM	
	AVG	SD	AVG	SD	AVG	SD
Average Incoming at w=5	19.21	25.17	17.45	22.23	26.68	28.92
Average Outgoing at w=5	11.83	13.33	9.65	10.23	10.8	11.41

The 15th South East Asian Technical University Consortium (SEATUC 2021)
 25th – 26th February 2021, Institut Teknologi Bandung, Indonesia

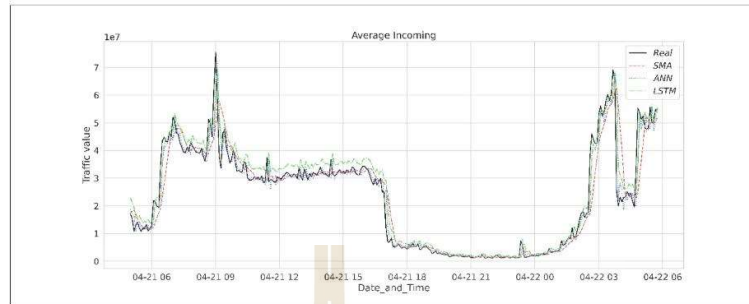


Fig 2. The comparison result of average incoming traffic between Real traffic, SMA, LSTM and ANN

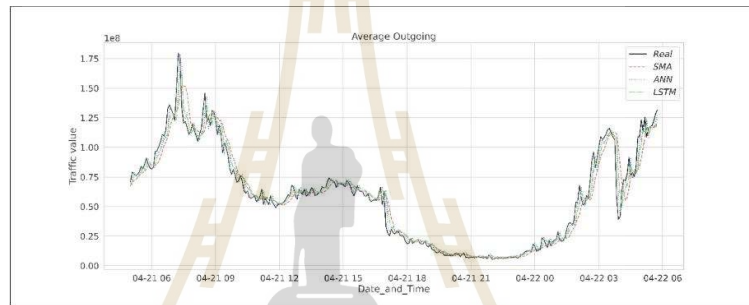


Fig 3. The comparison result of average outgoing traffic between Real traffic, SMA, LSTM and ANN

Table 3. One-way ANOVA of average incoming

Source of variance	SS	df	MS	F	Sig
Between Groups	2.851	2	1.425	21.785	0.000
Within Groups	116.403	1779	0.65		
Total	119.254	1781			

Table 4. One-way ANOVA of average outgoing

Source of variance	SS	df	MS	F	Sig
Between Groups	0.139	2	0.070	5.065	0.006
Within Groups	24.446	1779	0.014		
Total	24.585	1781			

Table 5. Post Hoc of average incoming and average outgoing

(I) Algorithm	(J) Algorithm	Average Incoming		Average Outgoing	
		Mean Difference (I-J)	Sig.	Mean Difference (I-J)	Sig.
SMA	ANN	.01752	.498	-.02163*	.006
	LSTM	-.07471*	.000	.01015	.328
ANN	SMA	-.01752	.498	-.02163*	.006
	LSTM	-.92243*	.000	-.01148	.241
LSTM	SMA	.07471*	.000	-.01054	.328
	ANN	.09224*	.000	.01148	.241

After forecasting internet traffic, using both average incoming and average outgoing direction based on three forecasting methods: SMA statistics, ANN and LSTM algorithm. The input was 25 minutes previous traffic to

predict the next 5 minutes traffic.

Based on the forecast results of average incoming in Figure 2, we plotted the comparison chart from the graph, and the ANN algorithm has a very close graph to the actual usage data. The results were then analyzed to find the percentage of mean error and standard deviation in Table 2. It can be seen that the ANN algorithm has a mean percentage error of 17.45 and a standard deviation of 22.23. Contrast with the forecast results using the SMA statistics and the LSTM algorithm with tolerances percentage: 19.21 and 26.68 and the standard deviation are 25.17 and 28.92, respectively.

The forecasting efficiency of the three methods was compared using a statistical tool, as shown in Table 3. One-way ANOVA was used to analyze the variance to compare the mean difference between the resulting mean error. It considers the values between groups and within the group whether there are significant differences or not. As a result, the statistically significant is .000, which is less than .05. It was found that the methods of forecasting traffic from both statistical and algorithmic differently. There is at least one pair of differences in mean tolerances statistically significant at the 0.05 level. Hence, a pair of comparisons were made to determine which forecasting tools had different mean error values. Table 5 shows the Multiple Comparisons or Post Hoc found that the statistical forecasts such as SMA versus LSTM and ANN versus LSTM were sig of .000. Therefore, the mean of the two groups was statistically significant. When examining the mean difference, it was found that SMA and ANN were more predictable than LSTM because they had a negative mean difference, i.e., there was a mean error less than the LSTM mean error. When considering ANN and SMA, it was found that the mean of the two groups was not statistically significant, or they were the same. When considering the mean difference, it will be negative. The mean ANN error is less than that of the SMA. Therefore, the ANN algorithm predicts average incoming traffic better than the other two methods.

Subsequently, the average outgoing will be considered when comparing the results in a graph in Figure 3. It was found that the results obtained from the ANN algorithm were closer to the actual results than the other two methods. When using the mean percentage error and the standard deviation, the mean percentage error is 9.65 and the standard deviation 10.23, which is better than the other two methods.

The forecasting efficacy of all three methods was compared using the One-way ANOVA statistical tool, as shown in Table 4. When comparing the variance of the mean of the tolerances obtained both between groups and within the group, it was found that the methods of forecasting traffic from both statistical and algorithmic were different. There is at least one pair of differences in mean tolerances statistically significant at the 0.05 level. When comparing the mean results individually, it was found that the SMA as a statistical tool. Against the ANN algorithm or vice versa ANN versus SMA, it can

be concluded that the mean of the two groups was statistically significant. When examining the mean difference, it was found that the ANN algorithm was more effective in forecasting traffic than SMA because of the minus mean difference, meaning that there was a mean error less than the mean discrepancy of the SMA. When comparing each pair between SMA with LSTM and ANN with LSTM, it was found that the mean of the two groups was not statistically significant or the same mean.

5. CONCLUSIONS

From forecasting the usage of data traffic on the network using SMA statistic, LSTM model, and ANN model, the forecast results show that the ANN model predicted traffic better than SMA and LSTM. Consideration by the percentage of mean error from both the average incoming and average outgoing of 17.45 and 9.65, respectively. One-way ANOVA tested the model forecast performance. However, the results from the test will see that there is still a large percentage of error.

6. FUTURE WORK

We will improve the forecasting method using statistical forecasting method integrated with the neural network in future work. For traffic forecasting in this paper, the parameters set are suitable for the trial and error method and simulation applied to the network within SUT. Suppose traffic forecasting was used in other environments; in that case, the forecasting parameters will also be changed accordingly, such as the k parameter was used to predict statistics. In addition to parameters of ANN and LSTM, such as the number of learning cycles, the number of layers, and the number of neural numbers may increase if used traffic increases, which will adjust the parameters according to the environment used.

REFERENCES

- Chiulli, and R.M., *Quantitative Analysis: An Introduction*. Amsterdam: Gordon and Breach Science, pp.234, 2020.
- Chantarakasemchit, Orawan, Siranee Nuchitprasitchai, and Yuenyong Nilsiam, Forex Rates Prediction on EUR/USD with Simple Moving Average Technique and Financial Factors, *ECTI-CON*, 2020.
- Dhenuvakonda, Padmaja, R Anandan, and N Kumar, STOCK PRICE PREDICTION USING ARTIFICIAL NEURAL NETWORKS, *J. of Critical Reviews.*, Vol. 7, 2020.
- Howard J. Seltman, *Experimental Design and Analysis*, pp.176-186, 2018.
- Stefan Lauren, and Dra. Harlili S., M.Sc, Stock Trend Prediction Using Simple Moving Average Supported by News Classification, *International conf on ICAICTA*, 2014.
- Le Xuan Hien, Hung Viet Ho, Giha Lee, and Sungho Jung, *Application of Long Short-Term Memory*

The 15th South East Asian Technical University Consortium (SEATUC 2021)
25th – 26th February 2021, Institut Teknologi Bandung, Indonesia

(LSTM) Neural Network for Flood Forecasting, pp.7, 2019.

Madan, Rishabh, and Partha Sarathimangipudi, Predicting Computer Network Traffic: A Time Series Forecasting Approach Using DWT, ARIMA and RNN, *Conf on IC3*, 2018.

Qian, Fei, and Xianfu Chen, Stock Prediction Based on LSTM under Different Stability, *Conf on ICCCBDA*, 2019.

WANG Shihao, ZHOU Qinzhen, YAN Han, LI Qianmu, and QI Yong, *A Network Traffic Prediction Method Based on LSTM*, China, 2019.

Yu, Shaoqiang, Caiyun Shang, Yang Yu, Shuyuan Zhang, and Wenlong Yu, Prediction of Bus Passenger Trip Flow Based on Artificial Neural Network, *Special Issue Article Advances in Mechanical Engineering*, vol. 8, 2016.

Yaffee, R. and McGee, M, *An Introduction to Time Series Analysis and Forecasting*, pp.18, 2000.

ACKNOWLEDGMENT

The authors wish to express their most sincere gratitude to Center for Computer Services, Suranaree University of Technology who supported the raw internet traffic data for this research.

PHOTOS AND INFORMATION



Supakarn Prajam received the B.Eng. (2017), degrees in Telecommunication Engineering from Suranaree University of Technology. She is currently pursuing the M. Eng. degree in Telecommunication Engineering.



Chitapong Wechtaisong received Ph.D. in the field of Information and Communication Engineering, from Shibaura Institute of Technology, Japan in 2016. Currently he is an assistant professor at the School of Telecommunication Engineering, Institute of Engineering, Suranaree University of Technology, Thailand.

มหาวิทยาลัยเทคโนโลยีสุรนารี

APPLYING MACHINE LEARNING APPROACHES FOR NETWORK TRAFFIC FORECASTING

Supakarn Prajam

Graduate Student, School of Telecommunication Engineering, Suranaree University of Technology, Muang District, Nakhon Ratchasima, 30000, Thailand
M6202460@g.sut.ac.th

Chitapong Wechtaisong

Assistant Professor, School of Telecommunication Engineering, Suranaree University of Technology, Muang District, Nakhon Ratchasima, 30000, Thailand
chitapong@g.sut.ac.th

Arfat Ahmad Khan

Lecturer, School of Manufacturing Engineering, Institute of Engineering, Suranaree University of Technology, Muang District, Nakhon Ratchasima, 30000, Thailand
arafat@sut.ac.th

Abstract

In the era of the digital world. The communication and the use the internet is an important role in today's society. As a result, the number of users networks traffic increases but not enough resources for users causing users to receive inefficient services. Therefore, the network service provider must take action to fix the aforementioned problem. Forecasting is therefore necessary in order to determine the amount of network traffic in order to support the future increase in user numbers. Consequently, this research investigates to assess network traffic forecasts comparing the machine learning: Artificial Neural Network (ANN), Long Short-Term Memory (LSTM), and statistical methods: Autoregressive Integrated Moving Average (ARIMA), Simple Moving Average (SMA). The method of sliding window will be used simultaneously and evaluate the forecast and model performance using the MAE, MAPE, MSE, RMSE and R-square algorithms, respectively. The results show that machine learning forecasting is more effective than statistical forecasting. Because the error value is lower, the model can reliably anticipate data. Therefore, the results of this research are expected to help network service provider to improve their networks quickly and efficiently to accommodate the number of users that may increase in the future.

Keywords: Machine learning; Forecasting; Artificial Neural Network; Long Short-Term Memory; Autoregressive Integrated Moving Average; Simple Moving Average.

1. Introduction

Nowadays, the life and well-being of society are being driven by technology. Communication through the online world has resulted in the increasing use of the internet. Therefore, the amount of traffic on the network has increased even more. Using an internet network are described as simple compared to a car and the transmission line is compared to a road. If one day when the number of cars on one of the roads has increased until the car overflows the road causing a bottleneck and causing traffic congestion. It's like having several internet users at the same place and time. As a result, the network system cannot support the use of the service user. This problem causes users to experience inefficient usage such as internet lag, internet speed that is not according to package, etc. Therefore, network operators need to find revise the problem aforementioned necessary have to be predicted the amount of internet usage on the network in the future. In order to allocate resources available on the network provide efficiency as well as sufficient for future service users and will result in more reliable network service providers.

There are various research was studied forecasting network traffic whether its machine learning forecasting or statistical forecasting. In [1] the integration of a model between RKM and SMA to predict online network traffic was examined. It is noticeable that when the models are combined, they can help in planning and designing networks efficiently. A study in [2] predicts uplink traffic in 3G cellular networks with ARIMA,

ENN, and MLP. The forecast results showed that ARIMA, ENN, and MLP were able to forecast nonlinear 3G data effectively. A study in [3] proposed a statistical method, namely Decomposition, Winter's Exponential Smoothing, and ARIMA, which predicted Mulawarman University internet usage data for each model. In [4] compare to the uplink and downlink network traffic forecasting capabilities with ARIMA, LSTM, and linear regression models. Both results studies show that the ARIMA model had a lower forecast error rate than the other models. However, in paper [5] predict traffic data in the cellular network, and time series forecasting has been studied [6], in two articles compare the forecasting capability between LSTM and ARIMA models. It can be seen LSTM is more efficient in forecasting accuracy. In article [7], traffic data in TCP/IP network from two providers, split by minute, hour, and day, where compared forecasting, are compared with MLP-BP, MLP-RP, RNN, and SAE. It may be considered that the RNN has better forecasting performance because it has a smaller NRMSE value and can predict the data more quickly. Article [8] contrasts the forecasting performance of telecommunication network usage data with LSTM, GRU, ANFIS, ANN, and GMDH models. The illustrate that the LSTM and GRU were more predictable than other models due to the smaller RMSE, MAPE, MAE values, the R is close to 1, and Theil's U-statistic is close to 0. In [9] proposes a forecast of mobile networks by comparing between ANN and linear prediction, ANN considers the length of the observation window. The results showed that the ANN was more predictable and required a shorter observation window length. And [10] proposes the use of a prediction ANN model to detect abnormalities in network traffic that indicates whether the network has been compromised or not. The ANN uses a series of weight updates is Gradient Descent, and Momentum and the results indicate that the ANN can predict network traffic anomalies. This research requires real-time network traffic usage forecasting to be able to support the number of users per minute from studying various research It was found that most of the research has not to forecast the data in real-time. Therefore, this research presents a sliding window method to assist in real-time data forecasting it will be used in conjunction with machine learning methods such as ANN, LSTM, and statistical methods such as SMA, ARIMA, then will evaluate the performance of each model used in forecasting.

2. Background Theory

2.1. Simple Moving Average (SMA)

The SMA, also known as single moving average, is a part of moving average (MA) suitable for forecasting abnormal and a little volatile time-series data. The working principle of SMA is to combine the values of a number of historical data and after that take to find the mean and forecast future data [11]. Which in the process of working will give equal weight to all historical data [12]. The first stage in calculating the SMA is to calculate the moving average, followed by forecasting as stated in the equation below.

$$MA(k)_t = \frac{1}{k} \sum_{i=t-k+1}^t y_i \quad (1)$$

$$\hat{Y}_{t+1} = MA(k)_t \quad (2)$$

where, k : the amount of historical data is a positive integer, y_t : past data value, $MA(k)_t$: moving average at time t , and \hat{Y}_{t+1} : prediction value at time t .

2.2. Autoregressive Integrated Moving Average (ARIMA)

ARIMA is a powerful statistical forecasting tool it analyzes the available data as a time series. It will use historical data to predict future data and give importance to every data both historical data as well as forecasting from moving average. ARIMA is a combination of three components, known as an "order" is Autoregressive (AR), Integrated (I), and Moving Average (MA), also known as ARIMA(p,d,q), where p is the number of coefficients of AR, q is the number of coefficients of MA [13] and d is the number of times the difference is determined. To eliminate seasonality components of variance and make time-series data stationary [14]. The AR(p) model selection was based on the partial autocorrelation function (PACF), and the MA(q) model selection was based on the auto correlation function (ACF), shown in equations 3 and 4 respectively [2].
Autocorrelation:

$$P_k = \frac{\sum_{t=k+1}^T (y_t - \bar{y})(y_{t-k} - \bar{y}_{t-k}) / (t-k)}{\sum_{t=1}^T (y_t - \bar{y})^2 / T} \tag{3}$$

where,

$$\bar{y}_{t-k} = \sum y_{t-k} / (T-k).$$

Partial Autocorrelation:

$$= \frac{\text{Covariance}(X_t, X_{t-2} | X_{t-1})}{\sqrt{\text{Variance}(X_t | X_{t-1}) \text{Variance}(X_{t-2} | X_{t-1})}} \tag{4}$$

To make is time-series data stationary the number of differences d is performed as shown in equation 5 [15].

$$\Delta^d X_t = \Delta^{d-1} X_t - \Delta^{d-1} X_{t-1} \tag{5}$$

Where, d : the order of differences and Δ^d : the command differential operator d .

2.3. Artificial Neural Network (ANN)

ANN or Neural Network is a model for forecasting with machine learning which can analyze and forecast time-series data. The ANN functions similarly to the human brain. This property of ANN can learn the information that feeds into the network and calculate with a computer system from a mathematical model. There are three layers in the ANN model: the input layer, hidden layer, and output layer, each with a distinct number of neurons. The first layer, the function of the input layer is to feed inputs into the model, and the number of neural relies on the inputs or feature that want the model to learn. The second layer, the hidden layer, is in charge of processing and takes the input that combines the preceding layer's weights and sums them with the bias. Then the sum will be calculated using a linear, relu, or sigmoid activation function. This layer can have any number of layers of neurons but increasing or lowering the number can affect the model's performance. The final layer, the output layer, will display the results of the previous layers calculations. If it is a regression model forecast, there is only one neuron in this layer. The building of the ANN model is depicted in Figure 1, while the model is functioning is depicted in equations 6 and 7.

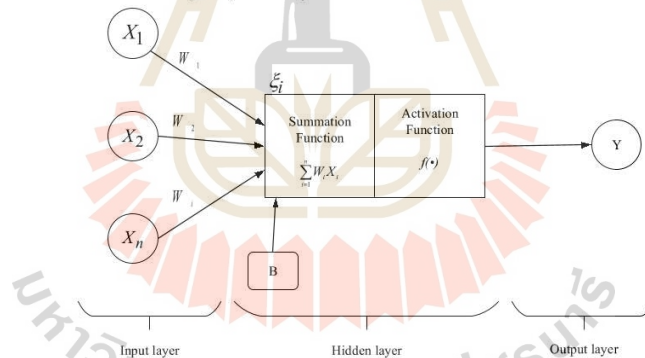


Fig. 1. Structure and concept of operation of the ANN model [16].

can be equation written as [16]:

$$\xi_i = \sum_{i=1}^n W_i X_i + B. \tag{6}$$

$$Y = f(\sum_{i=1}^n W_i X_i + B). \tag{7}$$

Where; ξ_i : sum function between input, weight, and bias, X_i : input data of neuron i , W_i : the weight of each link between the neurons, B : bias value, $f(\bullet)$: activation function, and Y : output node of neuron.

2.4. Long Short-Term Memory (LSTM)

The LSTM is a neural network in Recurrent Neural Network (RNN) family that was developed to correct gradient vanishing of RNN and increase memory cell stage. The LSTM can process or forecast sequential time-series data. The LSTM network consists of four cell gates: an input gate, an input modulation gate, a forget gate, and an output gate. The LSTM function at that forget gate will control the cell stage in memory if 1 preserves the cell stage in memory, but 0 deletes it. The input gate and input modulation gate control the new input data and check whether or not it can be used to update the cell stage. If an update is possible, the input modulation gate will perform the update with function tanh. The output gate decides which data to send out of the memory to get the result. The following equation can be used to define the LSTM is working principle [17][18] and Figure 2 depicts the internal structure of the LSTM:

$$f_t = \sigma(W_f \cdot [h_{t-1}, X_t] + b_f), \tag{8}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, X_t] + b_i), \tag{9}$$

$$c'_t = \tanh(W_c \cdot [h_{t-1}, X_t] + b_c), \tag{10}$$

$$c_t = f_t * c_{t-1} + i_t * c'_t, \tag{11}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, X_t] + b_o), \tag{12}$$

$$h_t = o_t * \tanh(c_t). \tag{13}$$

Where; x_t : input vector, w : weight, σ : sigmoid function, h : hidden stage, b : bias vector, c_t : cell stage, f_t : forget gate, i_t : input gate, c'_t : input modulation gate, and o_t : output gate.

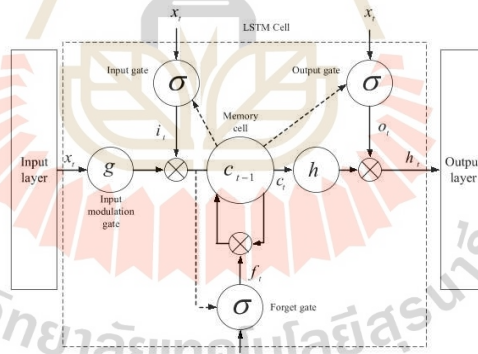


Fig. 2. The LSTM model structure and operation [8].

2.5. Performance tool

The tools used to measure the performance of network traffic forecasting for each model are as follows:

2.5.1. Mean Absolute Error (MAE)

It is a measure of the difference between the forecasted and actual data to evaluate if the discrepancy is greater or lesser. A model is considered efficient if its MAE value is low.

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - x_i|. \quad (14)$$

Where; y_i : data forecasting, x_i : actual data, and n : number of samples total.

2.5.2. Mean Absolute Percentage Error (MAPE)

It is a measure of a model's correctness or mistake in the form of an absolute percentage. The better the performance, the lower the MAPE value.

$$MAPE = \frac{100\%}{n} \sum_{i=1}^n \left| \frac{y_i - x_i}{x_i} \right|. \quad (15)$$

2.5.3. Mean Squared Error (MSE)

It is a metric for calculating the squared average error between forecast and actual data. The model is more accurate if the MSE value is low.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - x_i)^2. \quad (16)$$

2.5.4. Root Mean Squared Error (RMSE)

It is a metric for measure average size of the error between forecasted and actual data. The lesser the RMSE, the better.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - x_i)^2}. \quad (17)$$

2.5.5. R-squared (R^2)

It is a metric for how well a model can predict, by measure between the forecast data and the actual data. The model is performing well if R approaches 1 or 100 percent.

$$R^2 = 1 - \frac{\sum_{i=1}^n (x_i - y_i)^2}{\sum_{i=1}^n (x_i - \bar{y})^2}. \quad (18)$$

where; $\bar{y} = \frac{1}{n} \sum_{i=1}^n x_i$; mean of actual data.

3. Methodology

3.1. Input Dataset

The network traffic data set, which comprises of two sets of data, will be analyzed, and forecast in the future in this research. The first data set from research [19] can access and download from <http://www3.dsi.uminho.pt/pcortez/data/ittraffic.html>. Data collected from private ISP located at centers in 11 European cities from June 7, 2005, at 6:57 AM. to July 28, 2005, at 11:17 AM. was collected every 5 minutes (in bits per sec) to forecast.

The second set of data comes courtesy of the Computer Center of the Suranaree University of Technology. By collecting traffic data on the university network system was collected every 5 minutes (bit per sec) from June 15, 2021, at 00:00 AM. to July 15, 2021, at 03:35 AM. The data consists of date, time, the number of users, average incoming, and average outgoing, and in this research, it will forecast average incoming and average outgoing.

3.2. Experiment

ANN, LSTM, and ARIMA are machine learning forecasting tools and statistical methodologies, respectively it is presented by python computer programming on Google Cloud Platform namely Google Colab. Machine learning forecasting uses the Keras tool, a library in running high-performance models. The statistical forecasting section of SMA performs forecasting using excel. After forecasting future traffic, it will perform forecast evaluation and model performance with the MAE, MAPE, MSE, RMSE, and R-square algorithms, respectively.

3.2.1. Machine learning methods

Data is forecasted by two algorithms, ANN and LSTM, and the dataset is partitioned into two to forecast network traffic using machine learning methods. Figure 3 depicts the forecasting method.

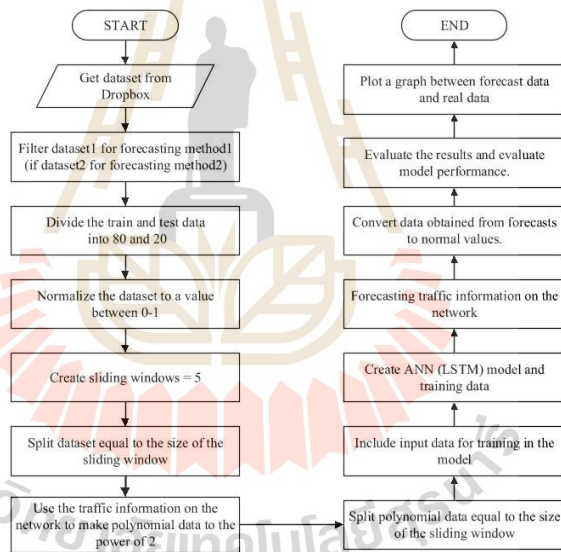


Fig. 3. Shows the forecasting methods of the ANN and LSTM algorithms.

Dataset 1 is a collection of network traffic volumes of 11 European cities. Network traffic data and polynomial network traffic data raised to the power of two are utilized for forecasting. For the forecasting process, it retrieves a cloud-stored dataset called Dropbox and performs filter traffic on the network system. The traffic data was split into two parts, with 80 percent being training datasets and 20 percent being testing datasets. Then, using MinMaxScaler, convert the data values to the same range and normalizing the data to be between 0-1. Then, we will create a sliding window with a value of 5, i.e., take 5 values from the past to forecast the future value, which is 6. (e.g., network traffic volume in the past 25 minutes to predict the future value at the 30 minutes). Once the sliding window has been created, the normalized network systems traffic volume data is divided equally by the sliding windows size, yielding the data for input#1. Then take the normalized traffic volume data to make a polynomial to the power of 2 and divide the data equal to the size of the sliding window, yielding input#2 and then sum input#1 to input#2 which will get input total. Afterward, proceed to model the ANN and LSTM algorithms the parameters for training are shown in Table 1. Following that, the model will be trained using 80 percent of the total input, and the test will use the remaining 20 percent of the data to estimate future data values. And the data obtained from the forecast is converted back from values in the range of 0-1 back to normal values. The experimental data and model performance were then evaluated using the MAE, MAPE, MSE, RMSE, and R-square approaches, respectively. Finally, the results obtained from the forecast are compared with the actual network traffic by plotting graphs.

The second data set is the traffic volume on the Suranaree University of Technology network system unlike data set 1 by dataset divides network traffic volume into average incoming and outgoing traffic, as well as the number of users. In forecasting is divided into two cases. Average incoming and average outgoing are the first and second cases, respectively. In case 1, the inputs for the model to train and the forecast will have three inputs: input#1: network traffic volume of average incoming, input#2: Polynomial to the power of 2 of average incoming, and input#3: the number of users.

In case 2, the model learns from the following input data: input#1: network traffic volume of average outgoing, input#2: Polynomial squared of average outgoing, and input#3: the number of users. Then arrange the input data to equal the size of the sliding window. Once the data is formatted, the inputs are concatenated first and trained on the data within the model. It will then forecast the data and evaluate the model's performance. The forecasting method is the same as in dataset 1 or in Figure 3 experimental procedure diagram.

Parameter	Data from 11 city in Europe		Data from network inside university	
			Average Incoming / Average Outgoing	
	ANN	LSTM	ANN	LSTM
Polynomial of network traffic	The power of 2		The power of 2	
Number of hidden layers	2		2	
Number of neurons in hidden layer	128		128	
Activation function in hidden layer	relu		relu	
Number of output layers	1		1	
Number of neurons in output layer	1		1	
Activation function in output layer	linear		linear	
Clients Count	No		Yes	
Epochs	200	100	200	100

Table 1. Parameter designated for training of ANN and LSTM

3.2.2. ARIMA Model

For network traffic volume forecasting using statistical methods is the ARIMA model, the forecast is divided into two data sets, and the forecasting process is shown in Figure 4.

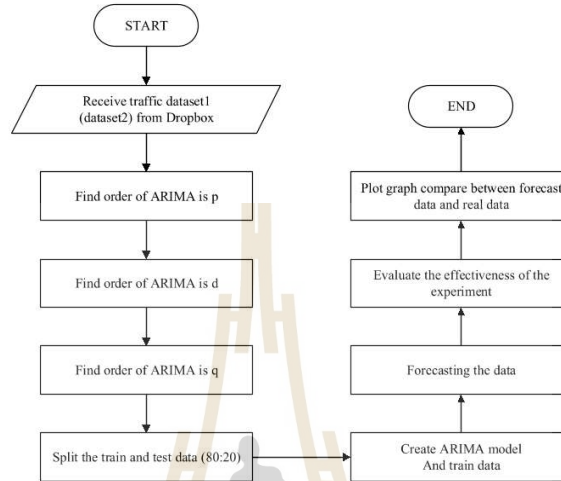


Fig. 4. The forecasting process using ARIMA statistical methods.

Data preparation principles, data training, data forecasting, forecasting performance measurement, and model performance evaluation will use the same method as ANN and LSTM. To use ARIMA to anticipate data, the first step is to extract network traffic volume from Dropbox, a cloud service. The first data set is network traffic volume data from 11 cities in Europe, and the second data set is network traffic volume within the university, divided into average incoming and average outgoing traffic, then carry out find order (p,d,q) used to create a forecasting model.

To find an order, it first it checks the raw data whether the raw data is stationary or not. Which will the Dickey-Fuller Test was used to determine whether the data was stationary, which P-values less than 0.05 indicate that the data is stationary, hence the resulting d-values do not need to make a difference. Then plot the PACF graph to find the AR value or p-value, which will measure the relative value of the data and the method used is "ols.". The p-value is then calculated from the graph using a confidence interval of 95 percent and the graph intersection, which will be at 1.96. And the last order is the MA value or q, which will the measured relationship of the data by plotting the ACF graph. The graph intersection with the 95 percent confidence interval is then used to determine the q-value. And table 2 shows order at used creating an ARIMA model to data forecasting.

The model will then be constructed using the previously discovered order. and divide the training data by 80 percent and testing data by 20 percent. It feed training data into the model so that it can learn and forecast the rest of the test data. After that, once the data set has been forecasted, will experimental evaluation and forecast performance is with MAE, MSE, RMSE, and R-square, respectively, and will plot graphs showing the comparison of the actual data with the forecasting data.

Parameter	Data from 11 city in Europe	Data from network inside university	
		Average Incoming	Average Outgoing
Order for model (p, d, q)	(2,0,0)	(3,0,1)	(3,0,0)

Table 2. Parameter designated for training in ARIMA

3.2.3. SMA Model

For forecasting data using statistical methods using the SMA model, it is calculated which use excel. The forecasted data is the network traffic of 11 European cities and the network traffic within the university network are average incoming and average outgoing. Where taking five past values, $k = 5$ and calculating find the $MA(k)$, according to equation (1), for example, bring five past values, or values from the last 25 minutes, add them together and divide by k . Then, after calculating the movement value, it predicts the next future value, such as the value at time 6 or the value at 30 minutes and calculates find the value of \hat{y}_{t+1} using equation (2). When the data forecast is finished, the forecast performance is check, and the model and plot's efficiency is compared, afterward graph will be showed to compare the actual data and the forecast data.

4. Result

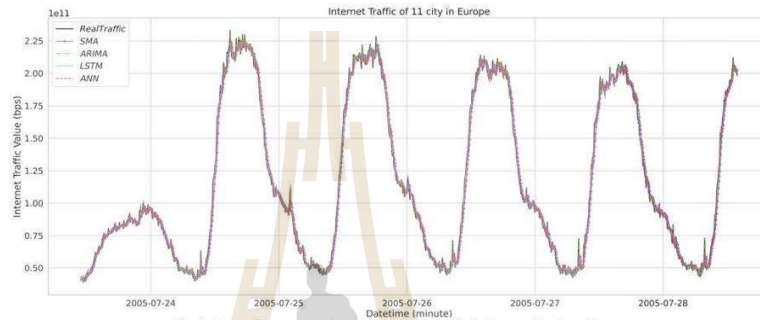


Fig. 5. The graph compares network forecasts of 11 cities in Europe of each model.

Model	MAE	MAPE	MSE	RMSE	R ²
SMA	4.59×10 ⁹	4.45	4.18×10 ¹⁹	6.47×10 ⁹	0.990
ARIMA	2.89×10 ⁹	76.12	1.62×10 ²²	4.03×10 ⁹	0.9957
LSTM	2.93×10 ⁹	3.15	1.54×10 ¹⁹	3.93×10 ⁹	0.9959
ANN	2.94×10 ⁹	3.19	1.53×10 ¹⁹	3.91×10 ⁹	0.9960

Table 3. The performance of each model forecasting data for 11 cities in Europe.

After forecasting the amount of traffic on the network successfully, the results can be displayed as follows. The result of the first set of data, network traffic volumes of 11 European cities, is shown in Figure 5, it can be seen the comparison of the forecasts of each model and comparing it with the actual data. Within the graph contains, the vertical axis represents the network traffic volume (in bps), and the horizontal axis represents the date and time (unit in seconds). The predictive data using machine learning is ANN and LSTM, as demonstrated in red and blue lines, it is more effective in forecasting than statistical forecasting, as shown in the graph, because the graph is nearby to the actual data line (black line). And the predictive performance of each model was tested with MAE, MAPE, MSE, RMSE and R-square as shown in Table 3. The ANN model has forecast efficiency 2.94×10⁹, 3.19, 1.53×10¹⁹, 3.91×10⁹, and 0.9960 respectively. And the LSTM model is effective in forecasting is 2.93×10⁹, 3.15, 1.54×10¹⁹, 3.93×10⁹, and 0.9959 respectively. It can be seen the ANN and LSTM models offer lower prediction errors than the ARIMA and SMA models.

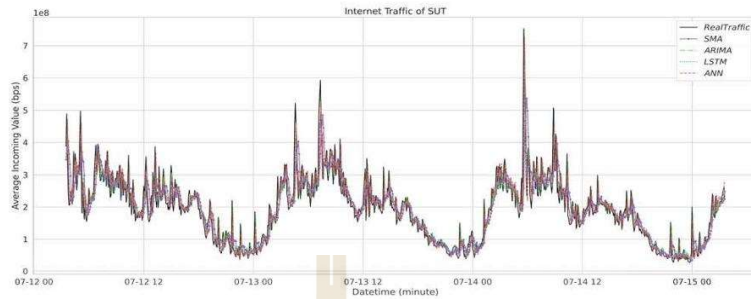


Fig. 6. The graph compares network forecast average incoming of SUT for each model.

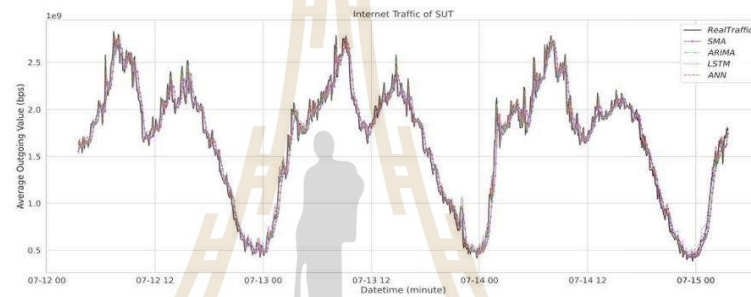


Fig. 7. The graph compares network forecast average outgoing of SUT for each model.

Model	Average Incoming					Average Outgoing				
	MAE	MAPE	MSE	RMSE	R ²	MAE	MAPE	MSE	RMSE	R ²
SMA	4.83×10 ⁶	16.53	5.49×10 ¹³	7.41×10 ⁶	0.7838	1.46×10 ⁷	6.43	3.72×10 ¹⁴	1.93×10 ⁷	0.9723
ARIMA	3.67×10 ⁶	79.52	3.29×10 ¹³	5.74×10 ⁶	0.7980	9.11×10 ⁶	68.32	1.56×10 ¹⁴	1.25×10 ⁷	0.9795
LSTM	3.63×10 ⁶	15.37	3.45×10 ¹³	5.87×10 ⁶	0.7878	9.83×10 ⁶	5.35	1.78×10 ¹⁴	1.33×10 ⁷	0.9767
ANN	3.59×10 ⁶	15.11	3.06×10 ¹³	5.53×10 ⁶	0.8117	9.03×10 ⁶	4.76	1.53×10 ¹⁴	1.24×10 ⁷	0.98

Table 4. The performance of each model forecasting data for SUT network.

The result from the second set of data is the traffic volume on the network system inside the university. Figure 6 indicate a graph comparing the results of the average incoming of each model against the actual data. It is noticeable that the ANN and LSTM model (red and blue lines, respectively) had more predictive results that were closer to the actual data (black lines) than ARIMA and SMA. When the model efficiency was evaluated using the parameters listed in Table 4, it was discovered that both models had lower error levels. Where the model efficiency ANN is 3.59×10⁶, 15.11, 3.06×10¹³, 5.53×10⁶, and 0.8117 respectively. The efficiency of the LSTM model is 3.63×10⁶, 15.37, 3.45×10¹³, 5.87×10⁶, and 0.7878 respectively.

In term of the results from the average outgoing test, it is illustrated in Fig. 7, will comparing the forecast results of each model with the actual data, it is found that the ANN and LSTM models (red and blue lines, respectively) yield results forecast close to actual data (black line). When comparing the efficiency of each model, the ANN and LSTM have lower predicted error values than the ARIMA and SMA, as shown in Table 4. The efficiency of a ANN model is 9.03×10⁶, 4.76, 1.53×10¹⁴, 1.24×10⁷, and 0.98 respectively. And the efficiency of a LSTM model is 9.83×10⁶, 5.35, 1.78×10¹⁴, 1.33×10⁷, 0.9767 respectively.

From the forecasting of both datasets, the ANN and LSTM models had an R-square approach closer to 1 as shown in Table 4, meaning the model had more accurate forecasting performance than ARIMA and SMA. As a result, forecasting using machine learning is proven to be more effective than predicting using statistical methods. For the forecast time of each model, ANN forecast time is about 5 minutes, LSTM forecast time is 30-40 minutes, and ARIMA forecast time is 20 minutes-1hour depends on the amount of data to be forecast that there is a lot or little. As a result, ANN outperforms LSTM in terms of forecasting time but if using LSTM for forecasting, it may not be able to respond to real-time forecasting. Because, when used to a real network system, speed of forecasting is required in order to allocate the network in a timely manner to satisfy the needs of users.

5. Conclusion

Forecasting network traffic volume from both datasets is nonlinear, with machine learning methods and statistical methods and will also incorporate sliding window methods. To forecast data in real time, such as data for the next 30 minutes based on the previous 25 minutes data. Based on the metrics MAE, MAPE, MSE, RMSE, and R-square, it was discovered that forecasting with machine learning methods, ANN, and LSTM model, has superior performance and accuracy, and is also less error-prone than statistical approaches ARIMA and SMA. However, using the machine learning approach used in this study, the ANN model has an advantage over the LSTM model in that it spends less time forecasting. Therefore, the ANN model is more suitable. Because in this research, the need to forecast the data quickly and efficiently. In order to support the use of users in real-time, the internet network service providers can also upgrade and allocate the network system to be sufficient for the number of users that will increase in the future.

Acknowledgments

The author would like to express his gratitude to the Center for Computer Services, Suranaree University of Technology for their assistance with network traffic data in this study.

References

- [1] Aldhyani, T. H. and Joshi, M. R. (2017): Enhancement of Single Moving Average Time Series Model Using Rough k-Means for Prediction of Network Traffic, *International Journal of Engineering Research and Application*, vol. 7, pp.45-51.
- [2] Oduro-Gyimah, F. K. and Boateng, K. O. (2018): A comparative analysis of telecommunication network traffic forecasting: A three model approach, 2018 IEEE 7th International Conference on Adaptive Science & Technology (ICAST), pp. 1-11.
- [3] Purnawansyah, P.; Haviluddin, H.; Alfred, R. and Gaffar, A. F. O. (2018): Network traffic time series performance analysis using statistical methods, *Knowledge Engineering and Data Science (KEDS)*, vol. 1, no. 1, pp. 1-7.
- [4] Lv, T.; Wu, Y. and Zhang, L. (2021): A Traffic Interval Prediction Method Based on ARIMA, in *Journal of Physics: Conference Series*, vol. 1880, no. 1, pp. 012031.
- [5] Azari, A.; Papapanou, P.; Denic, S. and Peters, G. (2019): Cellular traffic prediction and classification: A comparative evaluation of LSTM and ARIMA, *International Conference on Discovery Science*, pp. 129-144.
- [6] Zhou, K.; Wang, W. Y.; Hu, T. and Wu, C. H. (2020): Comparison of Time Series Forecasting Based on Statistical ARIMA Model and LSTM with Attention Mechanism, *Journal of Physics: Conference Series*, vol. 1631, no. 1, pp. 012141.
- [7] Oliveira, T. P.; Barbar, J. S. and Soares, A. S. (2016): Computer network traffic prediction: a comparison between traditional and deep learning neural networks, *International Journal of Big Data Intelligence*, vol.3, no. 1, pp. 28-37.
- [8] Do, Q. H.; Doan, T. T. H.; Nguyen, T. V. A.; Duong, N. T. and Linh, V. V. (2020): Prediction of Data Traffic in Telecom Networks based on Deep Neural Networks, *Journal of Computer Science*, vol. 16, no. 9, pp. 1268-1277.
- [9] Kirmaz, A.; Michalopoulos, D. S.; Balan, I. and Gerstaecker, W. (2020): Mobile Network Traffic Forecasting Using Artificial Neural Networks, 2020 28th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), pp. 1-7.
- [10] Ciptaningtyas, H. T.; Faticah, C. and Sabila, A. (2017): Network traffic anomaly prediction using Artificial Neural Network, *AIP Conference Proceedings*, vol. 1818, no. 1, pp. 020010.
- [11] Chiulih, R.M. (1999). *Quantitative Analysis: An Introduction*, Gordon and Breach Science Publishers, pp.234.
- [12] Yaffee, R. and McGee, M. (2000). *An Introduction to Time Series Analysis and Forecasting*, Cambridge: Academic Press, pp. 18.
- [13] Carlberg, C. (2013). *Predictive Analytics: Microsoft Excel*, Indianapolis: QUE, pp. 243.
- [14] Fan, J. and Yao, Q. (2005). *Nonlinear Time Series: Nonparametric and Parametric Methods*, New York: Springer, pp. 14.
- [15] Lu, S.; Zhang, Q.; Chen, G. and Seng, D. (2021): A combined method for short-term traffic flow prediction based on recurrent neural network, *Alexandria Engineering Journal*, vol. 60, no. 1, pp. 87-94.
- [16] Alaloul, W. S. and Qureshi, A. H. (2020): Data processing using artificial neural network, *Dynamic Data Assimilation-Beating the Uncertainties*.
- [17] Wang, S.; Zhuo, Q.; Yan, H.; Li, Q. and Qi, Y. (2019): A network traffic prediction method based on LSTM, *ZTE Communications*, vol. 17, no. 2, pp. 19-25.
- [18] Androletti, D.; Troia, S.; Musumeci, F.; Giordano, S.; Maier, G. and Tomatore, M. (2019): Network traffic prediction based on diffusion convolutional recurrent neural networks, *IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHOPS)*, pp. 246-251.
- [19] Cortez, P.; Rio, M.; Rocha, M. and Sousa, P. (2012): Multi-scale Internet traffic forecasting using neural networks and time series methods, *Expert Systems*, vol. 29, no. 2, pp. 143-155.

Authors Profile

Supakarn Prajam received the B.Eng. (2017), degrees in Telecommunication Engineering from Suranaree University of Technology. She is currently pursuing the M. Eng. degree in Telecommunication Engineering.



Chitapong Wechtaisong received B.Eng. in Telecommunication Engineering in 2008, M.Eng. in Telecommunication Engineering in 2014 from Suranaree University of Technology, Thailand and Ph.D. in the field of Information and Communication Engineering, from Shibaura Institute of Technology, Japan in 2016. Currently he is an assistant professor at the School of Telecommunication Engineering, Institute of Engineering, Suranaree University of Technology, Thailand. His research interest include wireless network design and optimization, network traffic localization and global engineering education.



Arfat Ahmad Khan received the B.Eng. degree in electrical engineering from the University of Lahore, Pakistan, in 2013, and the M.Eng. degree in electrical engineering from the Government College University Lahore, Pakistan, in 2015, and the Ph.D. degree in Telecommunication and Computer Engineering from the Suranaree University of Technology, Thailand, in 2018. From 2014 to 2016, he was an RF Engineer with Etisalat, UAE. From 2019 to 2020, he worked as a Post-Doctoral Researcher and lecturer at the department of Telecommunication Engineering, Suranaree University of Technology. Currently, he is working as a lecturer in the department of Manufacturing Engineering, Suranaree University of Technology. His research interests include robotics, automation, machine learning, optimization and stochastic processes, channel and the mathematical modeling, wireless sensor networks, ZigBee, green communications, massive MIMO, OFDM, wireless technologies, signal processing, and the advance wireless communications

ประวัติผู้เขียน

นางสาวศุภกานต์ ประจํา เกิดวันที่ 20 เมษายน พุทธศักราช 2539 เริ่มศึกษาชั้นอนุบาลปีที่ 1-2 ที่โรงเรียนบ้านช่องแมว ชั้นประถมศึกษาปีที่ 1-6 ที่โรงเรียนอนุบาลลำปลายมาศ ชั้นมัธยมศึกษาปีที่ 1-6 ได้สำเร็จการศึกษาจากโรงเรียนลำปลายมาศ อำเภอลำปลายมาศ จังหวัดบุรีรัมย์ และสำเร็จการศึกษาระดับปริญญาตรี สาขาวิชาวิศวกรรมโทรคมนาคม สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี ในปีพุทธศักราช 2557

ปีพุทธศักราช 2561 ได้ทำงานเริ่มทำงานที่บริษัทเอกชนเป็นระยะเวลาประมาณ 11 เดือน ต่อมาในปีพุทธศักราช 2562 จึงได้เข้าศึกษาต่อระดับปริญญาโท สาขาวิชาวิศวกรรมโทรคมนาคม หลักสูตรวิศวกรรมโทรคมนาคมและคอมพิวเตอร์ มหาวิทยาลัยเทคโนโลยีสุรนารี โดยขณะศึกษาได้รับทุนสนับสนุนจากทุนกิตติบัณฑิต ระหว่างศึกษาได้ปฏิบัติหน้าที่เป็นผู้ช่วยสอนปฏิบัติการในรายวิชา พื้นฐานและรายวิชาปฏิบัติการเกี่ยวกับวิศวกรรมโทรคมนาคม