การพยากรณ์ปริมาณสินค้าคงคลังขาเข้า (CBM) หลายช่วงเวลา สำหรับสินค้าไอที โดยใช้เทคนิคการเรียนรู้ของเครื่อง

อานนท์ณัฏฐ์ เยาว์ธานี

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
บัณฑิตศึกษา สาขาเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีไทย-ญี่ปุ่น
ปีการศึกษา 2568

MULTI-HORIZON FORECASTING OF INBOUND INVENTORY VOLUME (CBM) FOR IT PRODUCTS USING MACHINE LEARNING

Arnonnut Yaothanee

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Information Technology
Graduate Studies
Thai-Nichi Institute of Technology
Academic Year 2025

หัวข้อวิทยานิพนธ์	การพยากรณ์ปริมาณสินค้าคงคลังขาเข้า (CBM) หลายช่วงเวลา สำหรับสินค้าไอที โดยใช้เทคนิคการเรียนรู้ของเครื่อง
โดย	อานนท์ณัฏฐ์ เยาว์ธานี
สาขาวิชา	เทคโนโลยีสารสนเทศ
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ดร.ณปภัช วิชัยดิษฐ์

บัณฑิตศึกษา สถาบันเทคโนโลยีไทย-ญี่ปุ่น อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

.....รองอธิการบดีฝ่ายวิชาการ
(รองศาสตราจารย์ ดร.วรากร ศรีเชวงทรัพย์)
วันที่.....เดือน.....พ.ศ.....

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ
(ดร.สรมย์พร เจริญพิทย์)

.....กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.ฐิติพร เลิศรัตน์เดชากุล)

.....กรรมการ
(ดร.ธันยพร กณิกนันต์)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์
(ดร.ณปภัช วิชัยดิษฐ์)

อานนท์ณัฏฐ์ เยาว์ธานี : การพยากรณ์ปริมาณสินค้าคงคลังขาเข้า (CBM) หลายช่วงเวลาสำหรับสินค้าไอที โดยใช้เทคนิคการเรียนรู้ของเครื่อง. อาจารย์ที่ปรึกษา : ดร.ณปภัช วิชัยดิษฐ์, 55 หน้า.

การบริหารจัดการคลังสินค้าของบริษัทผู้จัดจำหน่ายสินค้าไอทีจำเป็นต้องวางแผนพื้นที่จัดเก็บและทรัพยากรหน้างานให้สอดคล้องกับปริมาตรสินค้าขาเข้า (Inbound) ที่มีความผันผวนตามช่วงเวลา อย่างไรก็ตาม การตัดสินใจเชิงปฏิบัติการมักเผชิญข้อจำกัดจากความไม่แน่นอนของปริมาณงานล่วงหน้า งานวิจัยนี้จึงมุ่งพัฒนาแบบจำลองพยากรณ์เชิงอนุกรมเวลาเพื่อคาดการณ์ ปริมาตรสินค้าขาเข้า (CBM) รายเดือน และขยายขอบเขตไปสู่การพยากรณ์ขนาดบรรจุภัณฑ์ (Forecast Dimension : ความกว้าง ความยาว ความสูง) เพื่อเพิ่มความครบถ้วนของข้อมูลเชิงคาดการณ์ที่สนับสนุนการจัดสรรพื้นที่และการเตรียมความพร้อมของคลังสินค้าในมิติที่สามารถนำไปใช้ได้จริง

ข้อมูลที่ใช้ในการศึกษาเป็นข้อมูลธุรกรรมการรับสินค้าเข้ารายวัน ครอบคลุมช่วง เดือนมกราคม พ.ศ. 2563 ถึงเดือนธันวาคม พ.ศ. 2567 และถูกสรุปผลเป็นรายเดือน (Monthly Aggregation) จากนั้นแบ่งชุดข้อมูลด้วยแนวทาง Time-Ordered Hold-out Split ได้แก่ Train: ม.ค.2563-ธ.ค.2565, Validation: ม.ค.2566-ธ.ค.2566 และ Test: ม.ค.2567-ธ.ค.2567 เพื่อรักษาลำดับเวลาและลดความเสี่ยงของการรั่วไหลของข้อมูล (Data Leakage) งานวิจัยออกแบบตัวแปรอธิบายโดยผสานฟีเจอร์หน่วงเวลา (Lag Features) และสถิติย้อนหลังแบบหน้าต่างเลื่อน (Rolling Statistics) เพื่อสะท้อนความต่อเนื่องของปริมาณงาน ร่วมกับตัวแปรปฏิทิน (Calendar features) เพื่อสะท้อนฤดูกาลและรูปแบบการทำงานของคลังสินค้า ทั้งนี้ได้ทำการเปรียบเทียบแบบจำลองในกลุ่ม Machine Learning และ Deep Learning ได้แก่ Random Forest, XGBoost, RNN, LSTM, LSTM with Attention และ GRU ภายใต้กรอบการพยากรณ์ล่วงหน้า t+1, t+3 และ t+6 เดือน โดยปรับแต่งไฮเปอร์พารามิเตอร์ด้วย Grid Search และประเมินผลด้วย MAE, MSE, RMSE และ MAPE (กำหนด MAPE เป็นตัวชี้วัดหลัก) เพื่อสะท้อนความเหมาะสมของแบบจำลองในแต่ละระยะเวลาการพยากรณ์อย่างเป็นระบบและสอดคล้องกับการใช้งานจริงในคลังสินค้า

บัณฑิตศึกษา ลายมือชื่อนักศึกษา .....

สาขาวิชา เทคโนโลยีสารสนเทศ ลายมือชื่ออาจารย์ที่ปรึกษา .....

ปีการศึกษา 2568

ARNONNUT YAOTHANEE : MULTI-HORIZON FORECASTING OF INBOUND INVENTORY VOLUME (CBM) FOR IT PRODUCTS USING MACHINE LEARNING. ADVISOR : DR.NAPAPHAT VICHAIDIS, 55 PP.

Warehouse operations in IT distribution companies require storage-space planning and on-floor resource allocation that align with inbound volume, which often fluctuates over time. However, operational decisions are frequently constrained by uncertainty in future workload. Therefore, this study aims to develop time-series forecasting models to predict monthly inbound volume in cubic meters (CBM) and to extend the scope to forecasting package dimensions (Forecast Dimension: width, length, and height). This extension is intended to provide more comprehensive predictive information to support storage-space allocation and warehouse readiness in a manner that is practically applicable.

The dataset used in this study consists of daily inbound transaction records covering the period from January 2020 to December 2024, which were aggregated to a monthly level (monthly aggregation). The data were then split using a time-ordered hold-out approach: the training set spans January 2020–December 2022, the validation set spans January 2023–December 2023, and the test set spans January 2024–December 2024, in order to preserve temporal order and reduce the risk of data leakage. Feature engineering combines lag features and rolling-window statistics to capture continuity in workload, together with calendar features to reflect seasonality and warehouse working patterns. The study compares models from both Machine Learning and Deep Learning families, including Random Forest, XGBoost, RNN, LSTM, LSTM with Attention, and GRU, under three forecasting horizons: t+1, t+3, and t+6 months. Hyperparameters for all models are tuned via grid search, and performance is evaluated using MAE, MSE, RMSE, and MAPE, with MAPE designated as the primary metric to systematically assess model suitability across forecasting horizons and to ensure alignment with real-world warehouse operations.

Graduate Studies Student's Signature.....

Field of Study Information Technology Advisor's Signature.....

Academic Year 2025

กิตติกรรมประกาศ

การทำวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดีด้วยความกรุณา และการช่วยเหลือสนับสนุนอย่างยิ่งจาก ดร.ณปภัช วิชัยดิษฐ์ ที่ให้คำปรึกษาแนะนำที่เป็นประโยชน์ในการทำวิจัยและให้คำแนะนำในการปรับปรุงแก้ไขข้อบกพร่องต่างๆ ด้วยความเมตตา ด้วยความเอาใจใส่อย่างยิ่งตลอดระยะเวลาในการทำวิทยานิพนธ์ฉบับนี้ และขอขอบคุณบริษัทผู้จัดจำหน่ายสินค้าไอทีรายใหญ่ในประเทศไทยแห่งหนึ่งที่ให้ความอนุเคราะห์ข้อมูลเพื่อนำมาใช้ในการวิจัยครั้งนี้

อีกทั้งผู้วิจัยตระหนักถึงความตั้งใจจริงและความทุ่มเทของคณะกรรมการทุกท่าน ได้แก่ ดร.ศรายุทธ นนท์ศิริ, อาจารย์กานดา ทิวัฑฒานนท์ และอาจารย์ที่ปรึกษา ดร.ณปภัช วิชัยดิษฐ์ ที่กรุณาให้ข้อเสนอแนะที่เป็นประโยชน์ต่อการทำวิทยานิพนธ์ฉบับนี้ พร้อมทั้งแนะนำแนวทางในการปรับปรุงแก้ไขข้อบกพร่องต่างๆ ด้วยความดูแลเอาใจใส่เป็นอย่างดี จนสามารถทำให้วิทยานิพนธ์ฉบับนี้เสร็จสิ้นสมบูรณ์

นอกจากนี้ทางผู้วิจัยขอขอบคุณแรงสนับสนุนจากครอบครัว ส่งเสริมให้ความรัก ความห่วงใยเป็นกำลังใจให้ตลอดมา อาจารย์ทุกท่านที่ได้ให้ความรู้และให้คำปรึกษาตลอดระยะเวลาในการศึกษา เพื่อนร่วมชั้นเรียน และบุคคลรอบตัวที่ประคับประคอง สนับสนุน ให้กำลังใจ จนสามารถจัดทำวิทยานิพนธ์ฉบับนี้จนเสร็จสิ้น

ผู้วิจัยหวังเป็นอย่างยิ่งว่า วิทยานิพนธ์ฉบับนี้จะก่อให้เกิดประโยชน์แก่ผู้ที่สนใจ และสามารถนำไปต่อยอดเพื่อประยุกต์ใช้กับองค์ความรู้ที่เกี่ยวข้อง อีกทั้งสามารถนำไปใช้ในการศึกษาหรือเป็นแนวทางในการนำไปเพิ่มโอกาสในการทำการตลาดของธุรกิจให้ประสบความสำเร็จมากยิ่งขึ้น

อานนท์ณัฏฐ์ เยาว์ธานี

สารบัญ

สารบัญ (ต่อ)

สารบัญตาราง

สารบัญรูป

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันคลังสินค้าเป็นส่วนสำคัญของห่วงโซ่อุปทาน (Supply Chain) ทำหน้าที่รับเข้า เก็บรักษา และกระจายสินค้าให้ทันตามความต้องการของลูกค้า ซึ่งความต้องการจะเปลี่ยนแปลงไปตามฤดูกาล โปรโมชัน รอบเปลี่ยนรุ่นสินค้าหรือพฤติกรรมการฝากสินค้าไว้ที่คลัง หากการคาดการณ์ปริมาตรสินค้ารับเข้า (Inbound) ไม่แม่นยำจะส่งผลให้เกิดปัญหากับพื้นที่จัดเก็บ เสี่ยงภาวะคลังแออัด (Overstock) ทำให้ต้นทุนแฝงสูงขึ้น เวลาหมุนเวียนงานเพิ่มขึ้น และระดับการให้บริการลูกค้าลดลงไปด้วย [1]

เทคโนโลยี AI (Artificial Intelligence) และ การเรียนรู้ของเครื่อง (Machine Learning) เข้ามาช่วยยกระดับความแม่นยำของการพยากรณ์ โดยสามารถเรียนรู้ลักษณะของอนุกรมเวลาได้ เช่น ฤดูกาล, รอบโปรโมชัน, การเปิดตัวสินค้าใหม่ ซึ่งงานวิจัยนี้ใช้การเรียนรู้ของเครื่องเป็นกรอบหลัก โดยมีการทบทวนทั้งวิธีการพื้นฐานเชิงสถิติ เพื่อเป็น Baseline โดยโมเดลที่นิยมใช้ได้แก่โมเดลอนุกรมเวลา เช่น ARIMA และ Exponential Smoothing โมเดลต้นไม้ตัดสินใจที่เหมาะกับข้อมูลเชิงตาราง เช่น Random Forest และ XGBoost โมเดลโครงข่ายประสาทเทียมสำหรับข้อมูลอนุกรมเวลา เช่น LSTM (Long Short-Term Memory), โมเดล GRU (Gated Recurrent Unit) และโมเดลกลไกความสนใจ (Attention) ที่ทำให้การทำนายมีประสิทธิภาพมากยิ่งขึ้น [2]

บริษัทผู้จัดจำหน่าย (Distributor) ผลิตภัณฑ์ไอทีรายใหญ่ ซึ่งจัดจำหน่ายสินค้าหลากหลายกลุ่มสินค้าไอที ให้แก่ตัวแทนจำหน่าย (Dealer) โดยจะรับสินค้าจากทางผู้ผลิต (Vendor) มาจัดเก็บไว้ในคลังสินค้าก่อนส่งมอบให้กับทางลูกค้า ซึ่งมีปริมาณสินค้าหมุนเวียนเป็นจำนวนมากและต้องใช้พื้นที่จัดเก็บมากเช่นกัน โดยมีการใช้บริการภายนอก (Outsource) เข้ามาบริหาร เนื่องจากมีสินค้าเข้ามาเป็นจำนวนมาก ส่งผลให้พื้นที่คลังเต็ม รถยกต้องวิ่งอ้อมเพื่อเข้าถึงตำแหน่งจัดเก็บ เนื่องจากสินค้าไม่มีที่จัดเก็บถูกวางไว้ที่พื้น และเสี่ยงต่อการเกิดความเสียหาย เพื่อให้เห็นภาพเชิงข้อมูล ทางผู้วิจัยได้ทำการวิเคราะห์นำข้อมูลย้อนหลัง 5 ปี (พ.ศ. 2563 - 2567) พบว่าปริมาณการรับเข้า (Inbound) มีความผันผวนสูง (High Volatility) อ้างอิงจากข้อมูลที่ทำความสะอาดเรียบร้อยแล้ว (Data Cleansing) โดยจะมีค่าเฉลี่ยรายเดือนเท่ากับ 1,302.10 CBM และมีค่าเบี่ยงเบนมาตรฐาน (Coefficient of Variation) อยู่ที่ 0.38 ซึ่งตัวเลขดังกล่าวจะสะท้อนให้เห็นถึงข้อมูลที่มีการกระจายตัวสูงและมีลักษณะ “ความต้องการไม่สม่ำเสมอ” (Intermittent Demand) ส่งผลให้การพยากรณ์แบบดั้งเดิมหรือการใช้ค่าเฉลี่ยไม่สามารถคาดการณ์ช่วงเวลาที่สินค้าเข้ามาเป็นจำนวนมาก (Peak Period) ได้อย่างแม่นยำ ซึ่งตอกย้ำความจำเป็นในการพัฒนาโมเดล เพื่อเตรียมบริหารจัดการภายในคลัง [3]

จากการทบทวนงานวิจัยที่ผ่านมา พบว่างานวิจัยจำนวนมากยังเน้นวิธีเชิงสถิติ เช่น ARIMA และ Exponential Smoothing [1], [4] ซึ่งอธิบายแนวโน้มและฤดูกาลได้ดี แต่ยังมีข้อจำกัดเมื่อพบความต้องการแบบไม่สม่ำเสมอ หรือกรณีมีความกระจายตัวของข้อมูลที่เอียงไปทิศทางใด ทิศทางหนึ่ง (Deposit Behavior) ของคู่ค้าหรือลูกค้า จะนำไปสู่ความคลาดเคลื่อนของการพยากรณ์อย่างสูง ดังนั้นยังมีช่องว่างในการศึกษาการประยุกต์ใช้เทคนิคการเรียนรู้ของเครื่อง (Machine Learning) และการเรียนรู้เชิงลึก (Deep Learning) เพื่อรองรับความซับซ้อนของข้อมูลในธุรกิจจัดจำหน่ายสินค้าไอทีรายใหญ่ในประเทศไทย

จากบริบทของผู้จัดจำหน่ายสินค้าไอทีรายใหญ่ พบว่าการรับสินค้าเข้าคลัง (Inbound) นั้นส่งผลต่อภาระงานในคลังมากที่สุด เนื่องจากหากมีปริมาตรสินค้าเข้ามาอย่างเฉียบพลันโดยจะส่งผลต่อการจัดสรรพื้นที่ในการจัดเก็บสินค้า ซึ่งจะส่งผลกระทบต่อการจัดเก็บสินค้า ตำแหน่งการวาง การใช้งานรถ หากปริมาตรสินค้าเข้ามามากกว่าความสามารถในการจัดเก็บภายในคลัง จะนำไปสู่ปัญหาที่พบเช่น การวางสินค้าไว้ที่พื้นชั่วคราวไม่นำขึ้นชั้นวาง ซึ่งอาจก่อให้เกิดความเสียหายต่อสินค้า แม้ว่าการส่งออกสินค้า (Outbound) จะเป็นอีกกิจกรรมที่มีความสำคัญต่อห่วงโซ่อุปทาน แต่อย่างไรก็ตามการส่งสินค้าออกนั้นมีการวางแผนและถูกควบคุมไว้ล่วงหน้าแล้วจากคำสั่งซื้อจากลูกค้า (Customer Order) ในขณะที่ Inbound มีความไม่แน่นอนสูงกว่า เนื่องจากมีอิทธิพลมาจากปัจจัยภายนอก เช่น ช่วงเวลาในการรับสินค้าในวันทำงานปกติ แผนการผลิตของผู้ผลิตสินค้า (Vendor) ส่งผลให้ Inbound มีความผันผวนและมีลักษณะความต้องการที่ไม่สม่ำเสมอ (Intermittent Demand) ดังนั้นผู้วิจัยจึงเล็งเห็นว่าการเลือกใช้ Inbound เป็นตัวแปรหลักในงานวิจัยนี้มีความเหมาะสมทั้งในเชิงข้อมูลและเชิงประยุกต์ใช้งาน

1.2 วัตถุประสงค์ของการวิจัย

1.2.1 เพื่อพัฒนาแบบจำลองการพยากรณ์ปริมาตรสินค้าคงคลังของกลุ่มสินค้าไอที โดยใช้เทคนิคการเรียนรู้ของเครื่อง

1.2.2 เพื่อเปรียบเทียบประสิทธิภาพโมเดลโดยใช้เกณฑ์วัดได้แก่ MAE, MSE, RMSE, MAPE ภายใต้ Time-Ordered Hold-out Split

1.3 คำถามวิจัย

1.3.1 เพื่อให้สอดคล้องกับวัตถุประสงค์งานวิจัยจึงกำหนดคำถามงานวิจัยดังนี้

(1) โมเดลในกลุ่ม Machine Learning และ Deep Learning มีความแตกต่างกันในด้านความแม่นยำของการพยากรณ์ปริมาตรสินค้าคงคลังขาเข้าหรือไม่ และโมเดลใดมีความเหมาะสมที่สุดภายใต้เงื่อนไขข้อมูลที่ศึกษา

(2) ระยะเวลาการพยากรณ์ (Forecast Horizon) ที่แตกต่างกัน ได้แก่ t+1, t+3 และ t+6 เดือน ส่งผลต่อระดับความแม่นยำของโมเดลอย่างมีนัยสำคัญหรือไม่ และลักษณะของผลกระทบดังกล่าวเป็นไปในทิศทางใด

1.4 ขอบเขตของงานวิจัย

1.4.1 ขอบเขตด้านชุดข้อมูล

(1) ขอบเขตด้านช่วงเวลาใช้ข้อมูลย้อนหลังระหว่างปี พ.ศ. 2563-2567

(2) ขอบเขตด้านประเภทข้อมูลใช้ข้อมูลการรับสินค้าเข้าสู่คลัง (Inbound) และการจ่ายสินค้าออก (Outbound) ของบริษัทผู้จัดจำหน่ายสินค้าไอทีรายใหญ่

(3) ขอบเขตด้านกลุ่มสินค้า พิจารณาเฉพาะกลุ่มสินค้าไอทีเท่านั้น

(4) ขอบเขตด้านตัวแปรที่ใช้ในการวิเคราะห์ข้อมูลที่ผ่านกระบวนการทำความสะอาด (Data Cleansing) แล้วประกอบด้วย ปริมาณสินค้า (Quantity), ขนาดบรรจุภัณฑ์ (Dimension) ได้แก่ ความกว้าง (Width), ความยาว (Length), และความสูง (Height) เพื่อนำไปคำนวณปริมาตรในหน่วยลูกบาศก์เมตร (Cubic Meter: CBM)

(5) ขอบเขตด้านปริมาณข้อมูล มีจำนวนรายการรับเข้าสินค้า (Inbound Transactions) รวมทั้งสิ้น 146,149 รายการ

(6) ขอบเขตด้านระดับความละเอียดของข้อมูล ข้อมูลดิบเป็นข้อมูลรายวัน (Daily Level) และถูกนำมารวมผลเป็นรายเดือน (Monthly Aggregation) เพื่อใช้ในการวิเคราะห์เชิงอนุกรมเวลา

(7) ขอบเขตด้านการนำข้อมูล ใช้เป็นชุดข้อมูลหลักสำหรับการฝึก (Training) การตรวจสอบ (Validation) และการทดสอบ (Testing) แบบจำลองการพยากรณ์

1.4.2 ขอบเขตด้านเทคนิคและวิธีการ

(1) ศึกษาและพัฒนาแบบจำลองในกลุ่มการเรียนรู้ของเครื่อง (Machine Learning) และการเรียนรู้เชิงลึก (Deep Learning) สำหรับงานพยากรณ์อนุกรมเวลา

(2) ดำเนินการประมวลผลและพัฒนาแบบจำลองด้วยภาษาโปรแกรม Python โดยใช้ไลบรารีที่รองรับงานวิเคราะห์ข้อมูลและการเรียนรู้ของเครื่อง

(3) ประเมินและเปรียบเทียบประสิทธิภาพของแบบจำลองภายใต้กรอบการแบ่งข้อมูลตามลำดับเวลา (Time-Ordered Hold-out Split)

(4) ปรับแต่งค่าพารามิเตอร์ของแต่ละโมเดลด้วยเทคนิค Grid Search ภายใต้ชุดข้อมูลตรวจสอบ (Validation Set)

(5) ใช้ตัวชี้วัดมาตรฐาน ได้แก่ MAE, MSE, RMSE และ MAPE ในการวัดระดับความแม่นยำของโมเดล

1.5 นิยามศัพท์เฉพาะ

1.5.1 **สินค้าคงคลัง (Inventory)** สินค้าที่ถูกจัดเก็บอยู่ภายในคลังสินค้าในช่วงเวลาหนึ่ง เพื่อรองรับกระบวนการรับเข้าและจ่ายออก

1.5.2 **สินค้าขาเข้า (Inbound)** กระบวนการและข้อมูลที่เกี่ยวข้องกับการรับสินค้าเข้าสู่คลัง ซึ่งสะท้อนปริมาณภาระงานและความต้องการใช้พื้นที่จัดเก็บ

1.5.3 **ขนาดบรรจุภัณฑ์ (Dimension)** ลักษณะทางกายภาพของบรรจุภัณฑ์สินค้า ได้แก่ ความกว้าง ความยาว และความสูง ที่ใช้เป็นฐานในการคำนวณปริมาตร

1.5.4 **ปริมาตรลูกบาศก์เมตร (Cubic Meter: CBM)** หน่วยวัดปริมาตรที่ใช้แทนความต้องการใช้พื้นที่จัดเก็บสินค้า โดยคำนวณจากขนาดบรรจุภัณฑ์

1.5.5 **อนุกรมเวลา (Time Series)** ข้อมูลที่จัดเรียงตามลำดับเวลาและมีความต่อเนื่องของค่าในแต่ละช่วงเวลา

1.5.6 **การพยากรณ์ (Forecasting)** กระบวนการคาดการณ์ค่าที่จะเกิดขึ้นในอนาคตจากข้อมูลในอดีต เพื่อสนับสนุนการวางแผนและการตัดสินใจ

1.5.7 **การเรียนรู้ของเครื่อง (Machine Learning)** วิธีการสร้างแบบจำลองทางสถิติที่ให้ระบบเรียนรู้รูปแบบจากข้อมูล เพื่อใช้ในการทำนายค่าตัวแปรเป้าหมาย

1.5.8 **การเรียนรู้เชิงลึก (Deep Learning)** แนวทางหนึ่งของการเรียนรู้ของเครื่องที่ใช้โครงข่ายประสาทเทียมหลายชั้นในการเรียนรู้ความสัมพันธ์ที่ซับซ้อนของข้อมูล

1.5.9 **ระยะเวลาการพยากรณ์ (Forecast Horizon)** ช่วงเวลาที่ต้องการคาดการณ์ล่วงหน้า เช่น 1 เดือน 3 เดือน หรือ 6 เดือน

1.6 ประโยชน์ที่คาดว่าจะได้รับ

1.6.1 สามารถคาดการณ์แนวโน้มช่วงเวลาที่กลุ่มสินค้าไอที จะเข้ามาที่คลัง เพื่อให้หน่วยงานที่เกี่ยวข้องเตรียมความพร้อม

1.6.2 ได้โมเดลที่สร้างขึ้นด้วยเทคนิคการเรียนรู้ของเครื่องที่มีความแม่นยำในการคาดการณ์ช่วงเวลาที่สินค้ากลุ่มสินค้าไอที จะเข้ามาใน 1, 3 และ 6 เดือนข้างหน้า

1.6.3 ผลการพยากรณ์ CBM สามารถใช้เป็นฐานในการวางแผนพื้นที่และทรัพยากรคลังสินค้าล่วงหน้า สำหรับช่วง t+1, t+3 และ t+6 เดือน

1.6.4 ได้เกณฑ์การวัดเชิงปริมาณของโมเดล เช่น MAE, MSE, RMSE, MAPE สำหรับเปรียบเทียบประสิทธิภาพโมเดล

1.6.5 ลดต้นทุนการจัดเก็บ โดยลดความเสี่ยงของ Overstock โดยจะใช้พื้นที่อย่างเหมาะสมและการบริหารจัดการคลัง

1.6.6 ข้อมูลเชิงคาดการณ์ที่พัฒนาขึ้นในงานวิจัยนี้สามารถใช้เป็นฐานข้อมูลเชิงวิเคราะห์เพื่อสนับสนุนการกำหนดกลยุทธ์ด้านการบริหารพื้นที่และทรัพยากรคลังสินค้าในระดับผู้บริหาร

บทที่ 2
ทฤษฎีและเทคโนโลยีที่ใช้ในการปฏิบัติงาน

2.1 ทฤษฎีที่เกี่ยวข้องกับการจัดการสินค้าคงคลัง

2.1.1 การจัดการสินค้าคงคลัง (Inventory Management)

เป็นกระบวนการที่สำคัญในห่วงโซ่อุปทานที่เกี่ยวข้องกับการวางแผน ควบคุม และตรวจสอบปริมาณสินค้าที่จัดเก็บอยู่ในคลังสินค้า มีวัตถุประสงค์เพื่อให้มีสินค้าเพียงพอต่อความต้องการของลูกค้า และการลดต้นทุนจากการจัดเก็บสินค้าส่วนเกิน (Overstock) ในมุมมองของ ผู้จัดจำหน่าย (Distributor) ซึ่งไม่ได้ผลิตสินค้าเอง แต่ทำหน้าที่ซื้อสินค้าจาก Vendor และขายต่อให้ Dealer การจัดการสินค้าคงคลังมีความสำคัญอย่างมาก เนื่องจากสินค้ามีจำนวนมาก หากไม่มีการบริหารที่เหมาะสม อาจจะเกิดปัญหาภายในคลังเช่น คลังเต็ม รถชนสินค้าเสียหาย เสียเวลาในการจัดการ และต้นทุนที่สูงขึ้นจากการจัดเก็บที่ไม่มีประสิทธิภาพในคลังสินค้าของผู้จัดจำหน่ายรายใหญ่ จะมีฤดูกาล (Seasonality) และเหตุการณ์พิเศษ เช่น โปรโมชัน ที่ทำให้รูปแบบเวลาไม่เป็นเชิงเส้น ทำให้การใช้วิธีดั้งเดิมแบบ ES/ARIMA ใช้เป็น Baseline ที่ดี แต่หากมีสัญญาณที่ซับซ้อนมากขึ้น จึงมีการใช้เทคนิค Deep Learning ร่วมกับ Attention/Transformer เพื่อโฟกัสช่วงเวลาที่สำคัญและจับ Long-Range Dependency ได้ดีกว่า นอกจากนี้ภายในคลังยังพบ SKU ที่ความต้องการสินค้าไม่สม่ำเสมอ หรือมาเป็นช่วง (Intermittent/Slow-Moving) ทำให้การใช้วิธี ES ให้ความคลาดเคลื่อนสูง ส่งผลให้งานวิจัยทางเทคนิค Croston และอนุพันธ์ ที่ลดอคติและปรับปรุงความแม่นยำได้อย่างชัดเจนสำหรับความต้องการประเภทนี้ [5], [6]

Intermittent Demand คือความต้องการสินค้าที่เกิดขึ้นแบบไม่สม่ำเสมอ บางช่วงที่ปริมาณมาก แต่บางช่วงเป็นศูนย์ยาวนาน ทำให้วิธีการพยากรณ์มาตรฐานอย่าง ARIMA หรือ ES ให้ความคลาดเคลื่อนสูง โมเดลกลุ่มนี้ถูกพัฒนามาเพื่อจัดการ SKU ประเภท Slow-Moving หรือ Lumpy Demand เช่น อะไหล่ โดยแกนหลักคือการออกแบบสมการพยากรณ์ที่ “ลดผลกระทบจากข้อมูลที่เป็นศูนย์จำนวนมาก เพราะทำให้ค่าเฉลี่ยผิดเพี้ยน” และรองรับ Obsolescence (การเลิกขายหรือสินค้าตกรุ่น) ได้ดีกว่า และโมเดลที่ใช้กันอย่างแพร่หลายและถูกพัฒนาต่อยอดดังนี้

(1) Croston’s Method จะแยกการพยากรณ์ออกเป็น 2 ส่วนคือ ขนาดความต้องการเมื่อเกิด และ ช่วงเวลาระหว่าง Demand แล้วจะรวมกันเป็นอัตราคาดหมาย ข้อดีคือใช้งานง่ายและเป็น Baseline ที่ดีในข้อมูลที่ศูนย์บ่อย แต่จะมีข้อเสียคือ มีอคติบวก (Positive Bias) และไม่อัปเดตเมื่อมีข้อมูลศูนย์ต่อเนื่อง ทำให้การคาดการณ์สูงเกินจริงถ้าสินค้าเริ่มตกรุ่น [7]

(2) Syntetos-Boylan Approximation (SBA) ซึ่งถูกพัฒนาขึ้นเพื่อแก้อคติของ Croston โดยจะเพิ่มตัวปรับแก้ (Debiasing Factor) ทำให้ค่าพยากรณ์มีความแม่นยำมากขึ้น และงานภาคสนาม

พบว่าเหมาะสมกว่า Croston ในหลายอุตสาหกรรม จุดแข็งคือมีความแม่นยำกว่าโดยไม่ซับซ้อนเพิ่มมากนัก แต่ข้อเสียคือ “ไม่ปรับลดค่าคาดหมาย” หากศูนย์ต่อเนื่องยาว จึงจะไม่ดีต่อกรณี Obsolescence [7], [8]

Teunter-Syntetos-Babai (TSB) ซึ่งถูกพัฒนาเพื่อแก้ข้อจำกัดใหญ่ของ Croston และ SBA โดยจะทำการเพิ่ม “การอัปเดตความน่าจะเป็น” ที่จะเกิดจาก Demand ทุกคาบเส้นเวลา ทำให้ค่าพยากรณ์ “ลดลงอัตโนมัติ” เมื่อพบข้อมูลที่มีค่าเป็นศูนย์ต่อเนื่องกัน จุดแข็งคือรองรับ Obsolescence ได้ดีกว่า Croston และ SBA ในส่วนของข้อเสียคือในบางกรณีที่ข้อมูลศูนย์ไม่มาก SBA อาจจะส่งผลให้ Error ต่ำกว่าเล็กน้อย [8]

2.1.2 แนวคิดการพยากรณ์

การพยากรณ์ (Forecasting) คือการทำนายเหตุการณ์ที่จะเกิดขึ้นในอนาคต โดยใช้ข้อมูลในอดีตและปัจจัยที่เกี่ยวข้องเพื่อสนับสนุนการตัดสินใจ ทำให้มีบทบาทสำคัญต่อการจัดการสินค้าที่อยู่ในคลัง เช่น การวางแผนการจัดเก็บ การสั่งซื้อ และลดความเสี่ยงที่จะเก็บสินค้าไว้ในระยะยาวจนสินค้าตกรุ่น โดยวิธีการดั้งเดิมจะใช้ Moving Average, Exponential Smoothing และ ARIMA [9] ซึ่งการประยุกต์ใช้ Machine Learning และ Big Data Analytics สามารถเพิ่มความแม่นยำและยืดหยุ่นกว่าวิธีการเดิม โดยแสดงให้เห็นว่าการใช้โมเดล เชิงสถิติร่วมกับ Machine Learning เช่น Decision Trees, Random Forests และ Neural Networks สามารถจัดการข้อมูลที่มีปริมาณมากและซับซ้อนได้ดี [10] รวมทั้งยังลดความเสี่ยงจากสินค้าเต็มคลัง (Overstock) และสินค้าขาดแคลน (Stock-out) และยังมีโมเดลสำหรับลำดับยาวและความสัมพันธ์ซับซ้อน (Deep และ Attention/Transformer) วิธี RNN/LSTM ร่วมกับ Attention ช่วยให้โมเดลโฟกัสช่วงเวลาที่สำคัญเช่น เดือนพีค, โปรโมชัน และจับ Long-range Dependency ได้ดีกว่า Baseline [11] ขณะที่โมเดล Transformer ถูกออกแบบให้รองรับลำดับที่ยาวและตัวแปรหลายมิติเช่น ปริมาณสินค้ารับเข้า-จ่ายออก, ข้อมูลขนาดสินค้า, ปฏิทินที่ทราบล่วงหน้าอย่างมีประสิทธิภาพ ซึ่งเพื่อให้ผลลัพธ์ตรงตามที่ต้องการ ควรกำหนดขอบเขตการพยากรณ์เช่น 3-12 เดือนข้างหน้าและกรอบประเมินผล โดยใช้ RMSE และ MAPE [12] เปรียบเทียบระหว่าง Baseline (ES/ARIMA) กับโมเดลสมัยใหม่เช่น LSTM, Attention/Transformer

2.1.3 แบบจำลองเชิงสถิติ (Statistical Forecasting)

2.1.3.1 Baseline Statistical Models

คือแบบจำลองเชิงสถิติ (Statistical Forecasting) ที่ถูกใช้มาอย่างยาวนาน มักถูกกำหนดให้เป็น Baseline เพื่อเปรียบเทียบกับวิธีการสมัยใหม่อย่าง Machine Learning หรือ Deep Learning ซึ่ง Baseline จะช่วยให้เห็นความซับซ้อนของโมเดลให้ “คุณค่าเพิ่มขึ้น” จริงหรือไม่

และยังเหมาะกับกรณีที่ต้องการพยากรณ์แบบสั้น-กลาง ระยะโดยจะอาศัยโครงสร้างของอนุกรมเวลา (ระดับ-แนวโน้ม-ฤดูกาล) เป็นหลัก

(1) Moving Average (MA) คือการหาค่าเฉลี่ยเคลื่อนที่ เป็นเทคนิคพื้นฐานที่ใช้ในการลดความผันผวนของอนุกรมเวลา โดยจะคำนวณค่าเฉลี่ยข้อมูลในช่วงที่หน้าต่าง K จุดล่าสุดเพื่อใช้เป็นค่าพยากรณ์ถัดไป เหมาะสำหรับอนุกรมเวลาที่ไม่มีแนวโน้มหรือฤดูกาล และจะใช้เป็นเส้นอ้างอิงเบื้องต้น (Baseline) ได้อย่างมีประสิทธิภาพ ในงานวิจัยนี้ช่วงรายสัปดาห์สามารถใช้เป็นสัญญาณนำเพื่อเตรียมทรัพยากรการรับเข้า (Receiving Capacity) แต่จะมีข้อจำกัดคือผลของการพยากรณ์จะมีความหน่วง (Lag) เมื่อเกิดการเปลี่ยนแปลงฉับพลันเช่น ช่วงโปรโมชันหรือเดือนที่มีความต้องการมาก

(2) Exponential Smoothing (ES/ETS) คือ การให้ค่าน้ำหนักกับข้อมูลล่าสุดและลดทอนตามเวลา ทำให้ตอบสนองต่อความเปลี่ยนแปลงได้ไวกว่า MA และขยายได้เป็น Holt (มีแนวโน้ม) และ Holt-Winters/ETS (มีแนวโน้มและฤดูกาล) จึงเหมาะกับการพยากรณ์ระยะสั้นถึงกลางในอนุกรมเวลาที่มีโครงสร้างระดับ แนวโน้ม และฤดูกาลชัดเจน [9], [10] สำหรับคลังสินค้า ES/ETS สามารถจับฤดูกาลรายปีของ Inbound เพื่อวางแผนพื้นที่และกำลังคนล่วงหน้าได้ อย่างไรก็ตามวิธีการนี้ไม่ได้ถูกออกแบบมาเพื่อจัดการกับปัจจัยภายนอกหลายมิติที่มีอิทธิพลพร้อมกัน

(3) ARIMA/SARIMA/SARIMAX คือการผสานองค์ประกอบของการถดถอยอัตโนมัติ (AR) การทำให้ข้อมูลนิ่งด้วยผลต่าง (I) และส่วนของเฉลี่ยเคลื่อนที่ของส่วนคงเหลือ (MA) ในส่วนของ SARIMA จะเพิ่มพารามิเตอร์ฤดูกาล และ SARIMAX จะอนุญาตให้ตัวแปรภายนอก (Exogenous) เพื่อสะท้อนถึงอิทธิพลของปฏิทิน วันหยุด หรือการกำหนดส่งของ Supplier จุดแข็งคือกรอบแบบจำลองที่เป็นระบบ เหมาะสมกับอนุกรมที่มี Autocorrelation และฤดูกาลที่สม่ำเสมอ ในบริบทคลังสามารถตั้งค่า Seasonal = 12 (รายเดือน) และจะเพิ่มตัวแปรปฏิทินใน SARIMAX เพื่อยกระดับความแม่นยำ ข้อจำกัดคือการเลือกพารามิเตอร์ซับซ้อนและประสิทธิภาพอาจจะด้อยลงเมื่อข้อมูลมีความไม่เชิงเส้นสูงหรือความสัมพันธ์ข้ามตัวแปรจำนวนมาก [11]

(4) Theta Method คือวิธีการที่ Theta จะแยกอนุกรมเวลาออกเป็นองค์ประกอบ “เส้น Theta” หลายเส้นผ่านการปรับความโค้งของแนวโน้ม แล้วรวมผลพยากรณ์กลับเข้าด้วยกัน เป็น Baseline ที่แข็งแรงมากในงานแข่งขันการพยากรณ์ (M-Competitions) จุดเด่นคือเรียบง่าย แต่จับแนวโน้มของฤดูกาลได้ดี เหมาะกับข้อมูลที่ Inbound มีทั้งเทรนด์ระยะยาวและฤดูกาล เช่น ขาขึ้นทั้งปีหรือความต้องการสูงแค่บางเดือน สำหรับคลังสินค้าใช้ Theta เพื่อได้ตัวตั้งที่เสถียรและคำนวณเร็วก่อนถึงจะค่อยเทียบกับ Machine Learning และ Deep Learning ขั้นสูง แต่จะมีข้อจำกัดคือไม่รองรับตัวแปรภายนอกและความสัมพันธ์หลายมิติภายในตัวแบบ [12]

2.1.4 การเรียนรู้ของเครื่อง (Machine Learning)

การเรียนรู้ของเครื่อง (Machine Learning) คือเทคนิคที่ทำให้คอมพิวเตอร์สามารถเรียนรู้ได้ด้วยตนเอง เป็นสาขาย่อยของปัญญาประดิษฐ์ (Artificial Intelligence : AI) ซึ่งจะมุ่งเน้นในการสร้างแบบจำลอง จากความรู้ความสัมพันธ์ต่างๆ ของข้อมูล เพื่อนำไปแก้ปัญหาที่มาจากชุดข้อมูลที่ใช้ในการเรียนรู้ หรือศาสตร์ที่ว่าด้วย การศึกษาและการสร้างอัลกอริทึม (Algorithm) ที่สามารถเรียนรู้ และพยากรณ์ข้อมูลได้ โดยเรียนรู้จากโมเดล (Model) ของข้อมูลตัวอย่าง ซึ่งใน Machine Learning ถูกใช้งานอย่างแพร่หลาย เช่น ธนาคารและการเงิน คอมพิวเตอร์วิทัศน์ อุตสาหกรรมการผลิต รวมถึงสื่อสารและบันเทิงเป็นต้น โดยการเรียนรู้ของเครื่องแบ่งออกเป็น 3 ประเภทคือ การเรียนรู้แบบมีผู้สอน (Supervised Learning), การเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) และการเรียนรู้แบบเสริมกำลัง (Reinforcement Learning)

การเรียนรู้แบบมีผู้สอน (Supervised Learning) คือการเรียนรู้จากข้อมูลโครงสร้างเป็นกลุ่มของอัลกอริทึม (Algorithm) ที่ใช้ชุดข้อมูลมีตัวแปรนำเข้า (Input) และผลลัพธ์ (Output) ที่ชัดเจนเพื่อฝึกฝนโมเดลเพื่อใช้ในการแก้ปัญหา

การเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) คือการเรียนรู้โดยใช้ชุดข้อมูลตัวแปรนำเข้า (input) แต่ไม่มีข้อมูลหรือผลลัพธ์ให้กับคอมพิวเตอร์ โดยจะใช้คอมพิวเตอร์วิเคราะห์และค้นหาความสัมพันธ์ของข้อมูลด้วยตนเอง โดยนิยมใช้ 3 รูปแบบ คือ การจัดกลุ่ม (Clustering) การลดมิติ (Dimensionality) และการจับกลุ่มความสัมพันธ์ (Association)

การเรียนรู้แบบเสริมกำลัง (Reinforcement Learning) คือการเรียนรู้แบบลองผิดลองถูก (Trial and Error) เพื่อให้บรรลุเป้าหมาย โดยจะมีการให้รางวัล (Reward) หากดำเนินการตามเป้าหมายได้ และการลงโทษ (Penalty) ในการกระทำบางอย่าง เพื่อปรับปรุงการตัดสินใจของคอมพิวเตอร์ เช่น การควบคุมหุ่นยนต์เป็นต้น

ซึ่งงานวิจัยนี้ทางผู้วิจัยจะใช้การเรียนรู้แบบมีผู้สอน (Supervised Learning) ในการทำพยากรณ์ ซึ่งแบ่งออกเป็น 2 ประเภท ได้แก่

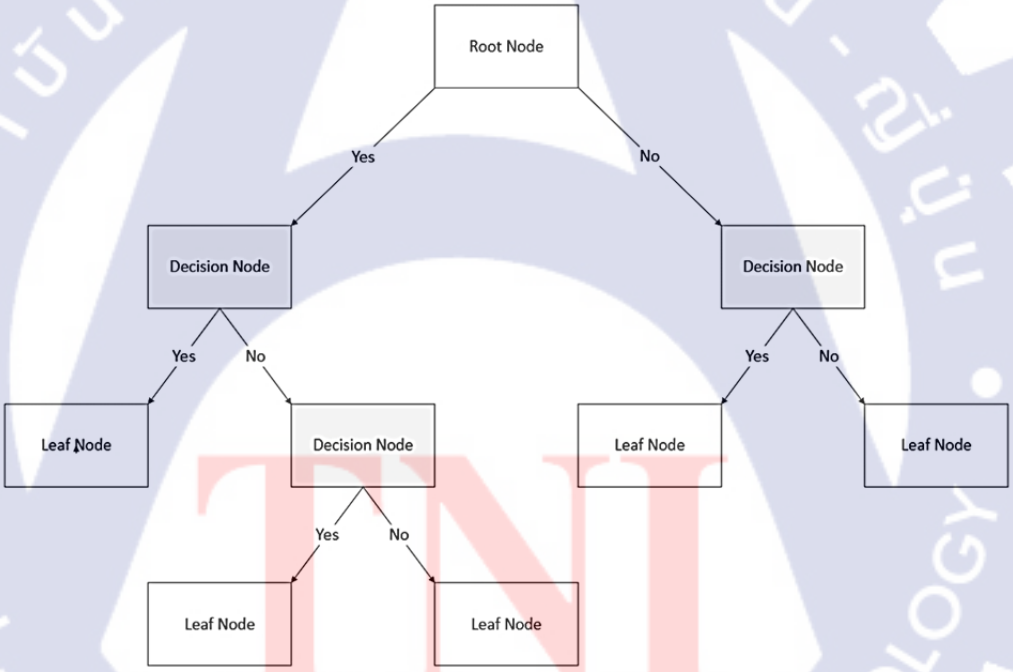
Classification คือการทำนายแบบแบ่ง “กลุ่ม” หรือ “ประเภท” เช่น การจำแนกอีเมลว่าเป็นสแปมหรือไม่ การทำนายลูกค้ามีแนวโน้มว่าจะซื้อสินค้าหรือไม่ หรือการวิเคราะห์สุขภาพจากข้อมูลเพื่อทำนายผู้ป่วยมีความเสี่ยงที่จะเป็นโรคใด

2.1.4.1 Decision Tree ต้นไม้การตัดสินใจ (Decision Tree) เป็นโมเดลที่ใช้กันอย่างแพร่หลาย มีลักษณะคล้ายกับต้นไม้ ใช้สำหรับการเรียนรู้แบบมีผู้สอน (Supervised Learning) โดยจะแบ่งออกเป็นแบบ การจำแนก (Classification) และการถดถอย (Regression) โครงสร้างต้นไม้จะประกอบไปด้วย โหนดราก (Root Node) คือตัวแปรหรือคุณลักษณะแรกที่ใช้ในการแบ่งข้อมูล กิ่ง (Branch) คือเส้นทางที่ข้อมูลจะไหลไปยังโหนดถัดไป โหนดตัดสินใจ (Decision Node) จุดที่จะเกิดการแบ่งข้อมูลตาม

เงื่อนไข และโหนดใบ (Leaf Node) คือผลลัพธ์สุดท้ายที่จะใช้ในการตัดสินใจ เช่น ค่าพยากรณ์ปริมาตรสินค้าในคลังของเดือนหรือปีถัดไป การแบ่ง Decision Tree สามารถจำแนกเป็น 2 ประเภทหลักดังนี้

(1) Decision Tree สำหรับจำแนกหมวดหมู่ (Classification Tree) คือใช้สำหรับปัญหาที่ผลลัพธ์เป็นกลุ่ม หรือเป็นหมวดหมู่เช่น การจำแนกสินค้าที่เป็น “สินค้าหมุนเวียนเร็ว” หรือ “สินค้าที่หมุนเวียนช้า” โดยใช้หลักการแบ่งข้อมูลเช่น Gini Impurity หรือ Entropy เพื่อวัดความไม่บริสุทธิ์ของกลุ่มข้อมูล โดยยิ่ง Gini Impurity มีค่าน้อย แสดงว่าการแบ่งข้อมูลนั้นมีความบริสุทธิ์สูง [13]

(2) Decision Tree สำหรับการถดถอย (Regression Tree) คือใช้กับปัญหาที่ผลลัพธ์มีค่าที่เป็นตัวเลขต่อเนื่อง เช่น การพยากรณ์ ปริมาณสินค้าคงคลังในเดือนถัดไป โดยเริ่มจากโหนดรากและแบ่งข้อมูลออกเป็น 2 ส่วน โดยการทำแบบซ้ำซ้อน (Recursive Binary Split) ตามตัวแปรที่สามารถลดความแปรปรวนของค่าผลลัพธ์ได้มากที่สุด โดยนิยมใช้ Mean Squared Error (MSE) เป็นเกณฑ์การตัดสินใจดังรูปที่ 2.1 การทำงานของ Decision Tree



รูปที่ 2.1 การทำงานของ Decision Tree

2.1.4.2 Regression หรือการถดถอย เป็นแนวทางการสร้างแบบจำลองเพื่อทำนายค่าตัวเลขต่อเนื่อง โดยเรียนรู้ความสัมพันธ์ระหว่างตัวแปรอธิบาย (Features) กับตัวแปรเป้าหมาย (Target) ภายใต้ข้อมูลในอดีต งานวิจัยนี้จัดอยู่ในปัญหา Regression เนื่องจากมีเป้าหมายเพื่อพยากรณ์ ปริมาตรสินค้าขาเข้า (Inbound) ในหน่วยลูกบาศก์เมตร: CBM ซึ่งเป็นค่าปริมาณที่เปลี่ยนแปลงตามเวลาและ

ต้องการผลลัพธ์เป็นตัวเลขสำหรับใช้วางแผนการจัดสรรพื้นที่คลังสินค้า ในเชิงแบบจำลอง การพยากรณ์สามารถอธิบายได้ด้วยรูปแบบทั่วไปดังสมการ

$$\hat{y}_{t+h} = f(x_t)$$

โดย $\hat{y}_{t+h}$ คือ ค่าปริมาตร CBM ที่แบบจำลองพยากรณ์ล่วงหน้า ณ ระยะเวลา h เดือน (เช่น h = 1, 3, 6) x_t

t = เวลา ณ ปัจจุบัน (รายเดือน)

h = ระยะเวลาการพยากรณ์ล่วงหน้า (t+1, t+3, t+6)

$\hat{y}_{t+h}$ = ค่าพยากรณ์ปริมาตรสินค้าขาเข้า (Inbound CBM)

x_t = ตัวแปรอธิบาย ณ เดือน

f (x) = แบบจำลอง Regressi*on* ที่ใช้ในการพยากรณ์

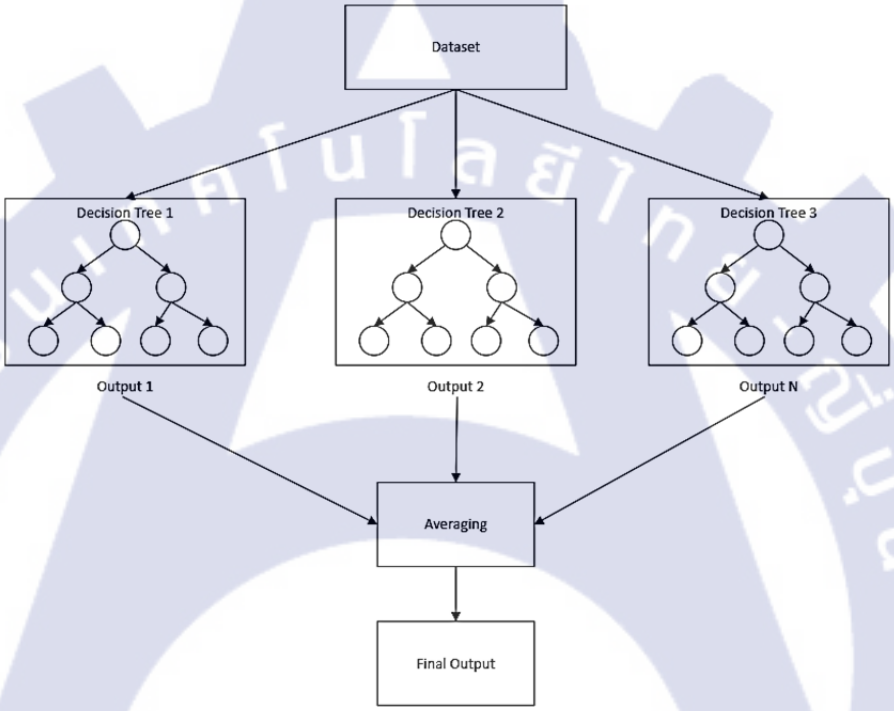
Regression สำหรับการพยากรณ์อนุกรมเวลาสามารถแบ่งเชิงการใช้งานที่เหมาะกับงานนี้ได้ 3 กลุ่มหลัก ดังนี้

(1) Linear Regression (Regression เชิงเส้น) เป็นแบบจำลองพื้นฐานที่อธิบายความสัมพันธ์เชิงเส้นระหว่างตัวแปรอธิบายกับ CBM โดยเหมาะสำหรับใช้เป็น “ฐานอ้างอิง” (Baseline) และช่วยให้ตีความผลได้ง่าย ทั้งนี้ในงานอนุกรมเวลามักต้องสร้างตัวแปรจากอดีต (เช่น Lag/Rolling) เพื่อให้สมการสามารถสะท้อนความต่อเนื่องของข้อมูลตามเวลาได้

(2) Tree-Based Regression (Regression แบบต้นไม้และแบบรวมกลุ่ม) เป็นกลุ่มแบบจำลองที่เรียนรู้ความสัมพันธ์ไม่เชิงเส้นได้ดี และรองรับปฏิสัมพันธ์ระหว่างตัวแปรหลายรูปแบบ โดยมักให้ผลที่เสถียรเมื่อใช้กับข้อมูลเชิงตารางที่สร้างฟีเจอร์ไว้แล้ว เช่น Lag/Rolling และฟีเจอร์ปฏิทิน ตัวอย่างที่สอดคล้องกับงานวิจัยนี้ ได้แก่ Random Forest Regression และ XGBoost Regression

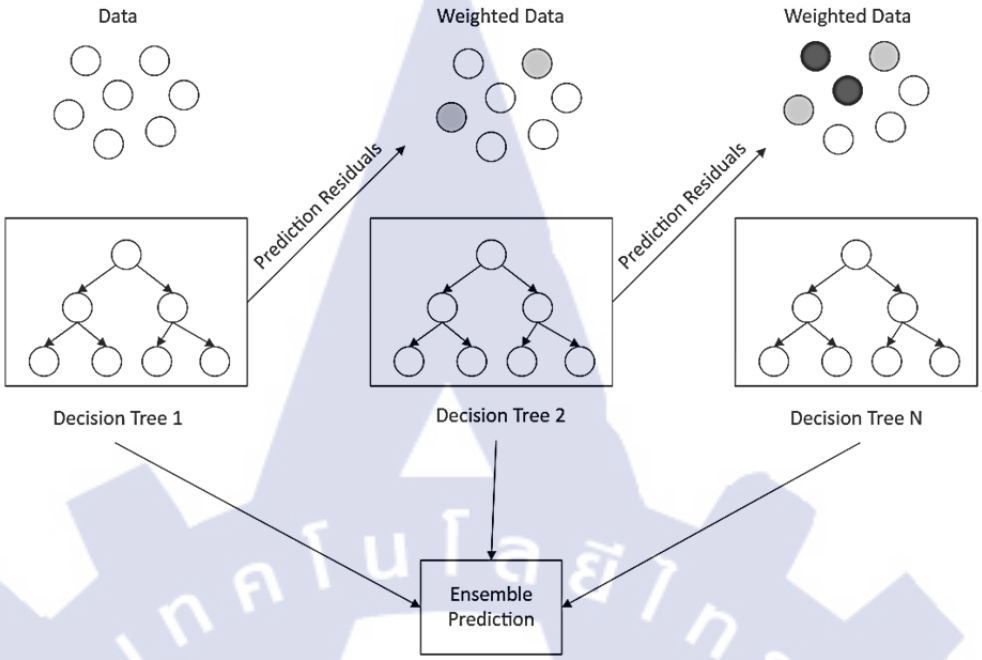
(3) Neural Network Regression สำหรับข้อมูลลำดับเวลา เป็น Regression ที่ใช้โครงข่ายประสาทเทียมเพื่อเรียนรู้รูปแบบเชิงลำดับจากข้อมูลย้อนหลังโดยตรง เหมาะเมื่อข้อมูลมีรูปแบบซับซ้อนหรือมีความสัมพันธ์ระยะยาว ตัวอย่างที่ใช้ในงานวิจัยแนวนี้ได้แก่ RNN, LSTM และ GRU ซึ่งสามารถออกแบบให้ทำนาย CBM ล่วงหน้าหลายระยะเวลาได้ ทั้งนี้ประสิทธิภาพจะขึ้นกับระยะเวลาการพยากรณ์ (t+1, t+3, t+6) และความเพียงพอของข้อมูลที่ใช้ฝึกสอน

2.1.4.3 Random Forest Random Forest เป็นโมเดลที่พัฒนาต่อยอดมาจาก Decision Tree จะเกิดจากการที่นำชุดข้อมูลมาแบ่งออกเป็นหลายๆ ชุด (Bootstrapping) และคุณลักษณะ (Feature Randomization) แล้วจะรวมผลลัพธ์ผ่านการโหวตหรือค่าเฉลี่ยทำให้ลดปัญหาของ Overfitting และเพิ่มความแม่นยำได้ดี ปัญหาการคาดการณ์สินค้าคงคลัง เช่น การพยากรณ์ความเสี่ยงของสินค้าเกินความจุคลังจะพบว่าสามารถจัดการกับข้อมูลมิติและไม่สมดุลได้อย่างมีประสิทธิภาพ [14] ดังรูปที่ 2.2 การทำงานของ Random Forest



รูปที่ 2.2 การทำงานของ Random Forest

2.1.4.4 Extreme Gradient Boosting (XGBoost) XGBoost เป็นอัลกอริทึมการเรียนรู้แบบ Ensemble ที่พัฒนาต่อยอดจาก Gradient Boosted Decision Trees โดยทำการสร้างต้นไม้การตัดสินใจเพื่อใช้ในการเรียนรู้ และนำมาฝึกสอนต่อกันหลายต้น โดยแต่ละต้นจะเรียนรู้จากค่าความผิดพลาดก่อนหน้า ซึ่งส่งผลให้ค่าความแม่นยำในการพยากรณ์จะมากขึ้นเรื่อยๆ โดยจะสร้างออกมาเป็นในรูปแบบของระดับชั้น (Level-Wise) เมื่อการเรียนรู้ต่อเนื่องจนมีความลึกมากพอที่จะมีค่าความผิดพลาดเป็นที่น่าพอใจ [15]



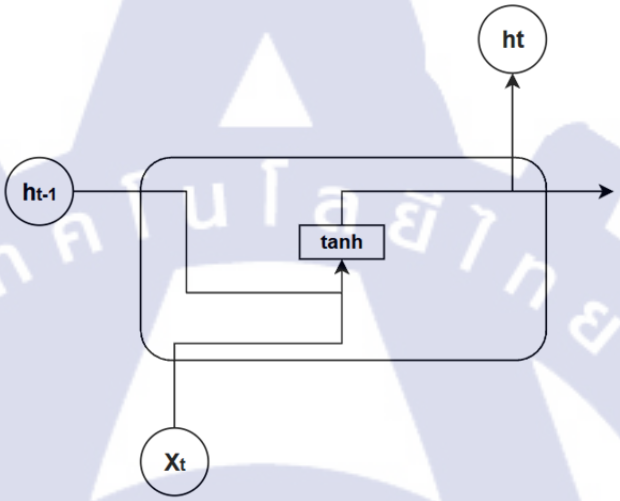
รูปที่ 2.3 การทำงานของ Extreme Gradient Boosting (XGBoost)

2.1.5 Deep Learning

Deep Learning (การเรียนรู้เชิงลึก) เป็นการพัฒนาต่อยอดมาจาก Neural Networks (โครงข่ายประสาทเทียม) โดยเพิ่มจำนวนชั้นการเรียนรู้ให้มากขึ้น ทำให้สามารถเรียนรู้ความสัมพันธ์ที่ซับซ้อนของข้อมูลได้ดีกว่าวิธีการดั้งเดิม สำหรับข้อมูลอนุกรมเวลา เช่น ปริมาณการรับ-จ่ายสินค้าออกจากคลัง โมเดล Deep Learning นั้นสามารถจับรูปแบบที่ไม่เชิงเส้น (Non-Linear) ความสัมพันธ์ระยะยาว (Long-Term Dependency) และผลกระทบหลายตัวแปร (Multivariate) ได้พร้อมกัน จึงได้รับความนิยมอย่างมากสำหรับการพยากรณ์ความต้องการสินค้า เมื่อเปรียบเทียบกับโมเดลเชิงเส้น (ARIMA, ES) โดยจะเหมาะกับข้อมูลที่มีโครงสร้างเชิงเส้นชัดเจน ดังนั้น Deep Learning สามารถปรับตัวเข้ากับข้อมูลที่มีความผันผวนสูง เช่น ช่วงโปรโมชัน หรือเดือนที่มีความต้องการสูงกว่าปกติ จึงจะช่วยให้การพยากรณ์มีความแม่นยำมากขึ้นและรองรับสถานการณ์จริงได้ อย่างไรก็ตามข้อจำกัดของ Deep Learning คือต้องมีข้อมูลที่เพียงพอ การตรวจข้อมูลจึงจำเป็นต้องรอบคอบ เช่น การทำ Normalization, การสร้าง Window/Lag Features, ป้องกันข้อมูลรั่ว ดังนั้นการฝึกสอนการใช้ทรัพยากรค่อนข้างสูง รวมถึงมีความเสี่ยงที่จะเกิด Overfitting หากไม่ได้กำหนดขอบเขตการทดลองและตัวชี้วัด (RMSE, MAPE, Accuracy) อย่างเป็นระบบ [16]

2.1.5.1 Recurrent Neural Network (RNN)

RNN เป็นโครงข่ายประสาทเทียมแบบวนซ้ำ ถูกออกแบบมาเพื่อจะจัดการข้อมูลลำดับเวลา (Sequential Data) โดยการประมวลผลย่อยเรียกว่าโหนด (Node) ซึ่งเป็นการจำลองลักษณะการทำงานของเซลล์การส่งสัญญาณของมนุษย์ ซึ่งแต่ละโหนดจะรับข้อมูลจากค่าปัจจุบันและสามารถเรียนรู้ความสัมพันธ์ของอนุกรมเวลาได้ [17]



รูปที่ 2.4 การทำงานของ Recurrent Neural Network

ในระหว่างที่โหนดเชื่อมกันอยู่จะถูกแบ่งออกเป็น 3 ส่วนดังนี้

Input Layer เป็นชั้นรับข้อมูลเข้ามาแบบทีละชั้น เช่น ค่าหนึ่งค่าในชุดข้อมูลของเวลา

Hidden Layer โหนดประสาทภายในซึ่งจะมีการเชื่อมโยงกับตัวเอง ช่วยให้โมเดล สามารถจดจำและใช้งานข้อมูลจากการคำนวณรอบก่อนหน้านี้

Output Layer ชั้นที่ส่งออกข้อมูลผลลัพธ์ลำดับสุดท้ายที่ถูกประมวลผลเรียบร้อยแล้ว

2.1.5.2 Long Short-Term Memory (LSTM)

LSTM คือโครงข่ายประสาทเทียมแบบวนซ้ำ (Recurrent Neural Network) ถูกออกแบบมาเพื่อแก้ไขปัญหาที่มีข้อมูล ลำดับยาว (Long Sequences) ซึ่งโมเดล RNN ปกติไม่สามารถที่จะจดจำได้ดี เนื่องจากเกิดปัญหา Gradient Vanishing ทำให้ข้อมูลในอดีตถูกลืมเมื่อเวลาผ่าน

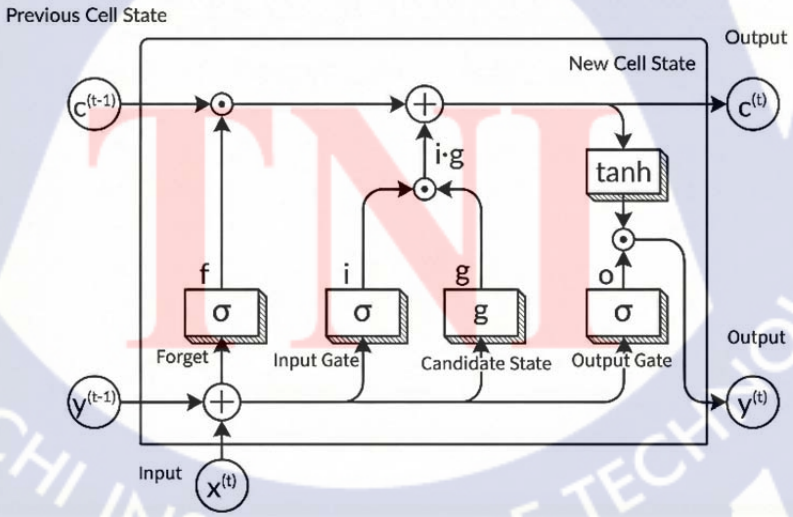
ไป กลไกหลักของ LSTM คือ Memory Cell ที่จะสามารถ “จำ” หรือ “ลืม” ข้อมูลบางส่วนตามเงื่อนไข [18] โดยใช้โครงสร้างหลัก 3 ส่วนคือ

Forget Gate จะทำหน้าที่กำหนดว่าข้อมูลจากสถานะก่อนหน้า (Previous Hidden State) จะถูกเก็บไว้มากน้อยขนาดไหน โดยจะใช้ค่าที่ได้จากการคำนวณ Sigmoid Function โดยจะให้ค่าอยู่ระหว่าง 0 ถึง 1 ซึ่งจะบ่งบอกถึงอัตราส่วนที่มีข้อมูลจะถูกลืมหรือจะยังคงไว้

Input Gate ทำหน้าที่กำหนดว่าข้อมูลที่เข้ามา ควรจะถูกเก็บไว้ในหน่วยความจำมากน้อยขนาดไหน โดยใช้การคำนวณ Sigmoid Function และ Tanh Function เพื่อทำการควบคุมข้อมูลที่จะถูกบันทึกลงในหน่วยความจำ

Output Gate ทำหน้าที่เลือกข้อมูลจากหน่วยความทรงจำจะถูกส่งออกมาเป็นผลลัพธ์ และ Hidden State ที่จะถูกส่งไปยังอีก Cell โดยจะผสมข้อมูลจากสถานะปัจจุบัน (Cell State) ผ่านการคำนวณ Tanh Function และใช้สถานะก่อนหน้า (Previous Hidden State) ผ่านการคำนวณ Sigmoid

ทั้งสามโครงสร้างนี้ส่งผลทำให้สามารถเก็บความสัมพันธ์ระยะยาวเช่น ฤดูกาล แนวโน้ม สามารถรับข้อมูลได้หลายมิติ อย่างไรก็ตามโมเดลต้องการข้อมูลจำนวนมากเช่นเดียวกับ Deep Learning การปรับพารามิเตอร์มีผลต่อความเสถียร และเมื่อข้อมูลมีศูนย์บ่อยหรือความต้องการเป็นช่วง (Intermittent) ประสิทธิภาพอาจจะด้อยกว่าวิธีเฉพาะทางอย่าง Croston, SBA และ TSB แต่ว่าจะเหมาะกับ SKU/High-Level Series ที่มีสัญญาณต่อเนื่องหรือฤดูกาลชัดเจนมากกว่า [19]



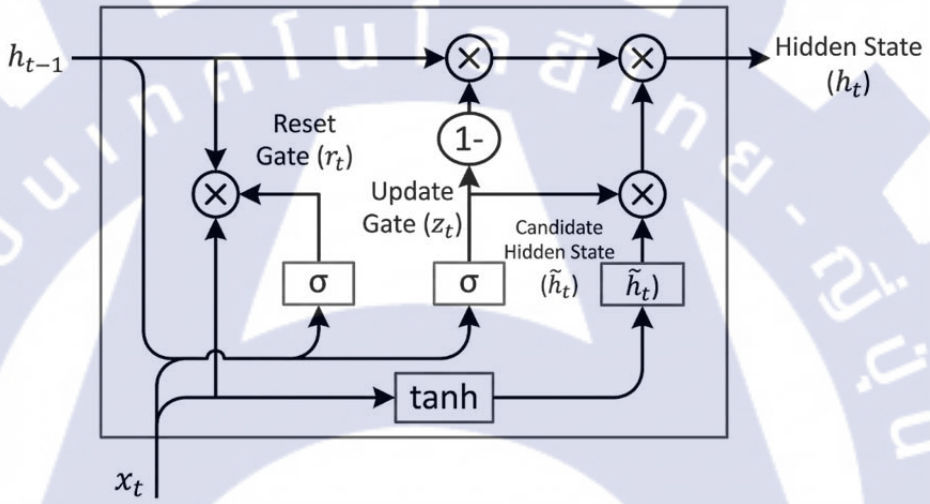
รูปที่ 2.5 การทำงานของ Long Short-Term Memory (LSTM)

2.1.5.3 Gated Recurrent Unit (GRU)

GRU เป็นโครงข่ายประสาทเทียมที่ถูกออกแบบมาสำหรับการประมวลผลแบบลำดับ (Sequence) ที่จะมาช่วยในการแก้ปัญหา Vanishing Gradient หรือ Exploding โดยจะทำเช่นเดียวกับ LSTM [20] แต่โครงสร้างการทำงานของ GRU จะประกอบเพียงแค่ 2 ส่วนคือ

Reset Gate ที่จะคอยทำหน้าที่ควบคุมว่าข้อมูลจากสถานะก่อนหน้านี้จะล้างค่าก่อนส่งไปยังสถานะปัจจุบัน

Update Gate ทำหน้าที่ควบคุมข้อมูลจากสถานะก่อนหน้าที่ควรใช้เป็นค่าสถานะใหม่เท่าไหร่



รูปที่ 2.6 การทำงานของ GRU

2.1.5.4 LSTM with Attention Mechanism

การผสาน Attention เข้ากับ LSTM ช่วยให้น้ำหนักของ “ช่วงเวลาสำคัญ” ที่มีผลต่อการพยากรณ์ เช่น เดือนที่มีโปรโมชันหรือวันหยุด ทำให้โมเดล โฟกัสช่วงสัญญาณที่เกี่ยวข้อง ลดผลรบกวนจาก Noise และช่วยตีความได้ จุดแข็งคือรับมือกับความผันผวนของ Non-Stationarity ได้ดีขึ้น โดยเฉพาะเมื่อมีตัวแปรภายนอกหลายตัวและรูปแบบฤดูกาลที่เปลี่ยนตามเหตุการณ์ ส่วนจุดอ่อนคือเป็นโมเดลที่มีสถาปัตยกรรมซับซ้อนขึ้น ต้องจูนพารามิเตอร์มากขึ้นและใช้ทรัพยากรสูงกว่า LSTM ปกติ รวมทั้งต้องระวังเรื่องข้อมูลรั่วเมื่อสร้างฟีเจอร์เหตุการณ์ล่วงหน้างานวิจัยของ Qin et al. [21] ได้เสนอ Dual-Stage Attention RNN (DA-RNN) ซึ่งใช้ทั้ง Input Attention และ Temporal Attention เพื่อเลือกคุณลักษณะและช่วงเวลาที่สำคัญสำหรับการพยากรณ์ เพื่อจัดการความผันผวนและปัจจัยภายนอกได้อย่างมีประสิทธิภาพ นอกจากนี้ยังได้พัฒนาแนวทางที่ผสาน LSTM เข้ากับ Attention Mechanism และ

มีกรอบการเรียนรู้แบบ Fuzzy Cognitive Map เพื่อยกระดับการจับความสัมพันธ์ที่ซับซ้อนของข้อมูลอนุกรมเวลาทำให้สามารถเพิ่มความแม่นยำในการพยากรณ์ [22]

2.1.6 การประเมินประสิทธิภาพการทำงานของโมเดล

การประเมินประสิทธิภาพโมเดลที่สร้างขึ้นด้วยเทคนิคการเรียนรู้ของเครื่อง ในการพยากรณ์สินค้าคงคลัง การเลือกใช้ตัวชี้วัด (Evaluation Metrics) มีความสำคัญอย่างยิ่ง เพราะจะช่วยในเรื่องของความแม่นยำในการพยากรณ์ และยังช่วยบ่งชี้ความเสี่ยงเชิงธุรกิจ เช่น สินค้าล้น หรือการขาดแคลนสินค้า การประเมินประสิทธิภาพจึงเลือกใช้หลายตัวชี้วัดดังนี้

(1) ค่าเฉลี่ยความคลาดเคลื่อนสัมบูรณ์ (Mean Absolute Error: MAE) ใช้วัดค่าความคลาดเคลื่อนโดยเฉลี่ยในรูปแบบเชิงเส้น เหมาะกับการตีความเชิงปริมาณสินค้า เช่น คลาดเคลื่อนเฉลี่ยกี่หน่วยของสินค้า

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i - y_i|$$

N = จำนวนข้อมูลทั้งหมด

y_i = ค่าจริงตำแหน่ง i

$\hat{y}_i$ = ค่าพยากรณ์ที่ได้ในตำแหน่ง i

(2) ค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error: MSE) คือค่าความผิดพลาดกำลังสองระหว่างค่าที่พยากรณ์ได้กับค่าจริง ทำให้เมื่อมีข้อผิดพลาดขนาดใหญ่ ค่าจะถูกขยายขึ้น เหมาะกับการใช้เป็นฟังก์ชันการสูญเสีย (Loss Function) ในการฝึกโมเดล

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

N = จำนวนข้อมูลทั้งหมด

y_i = ค่าจริงตำแหน่ง i

$\hat{y}_i$ = ค่าพยากรณ์ที่ได้ในตำแหน่ง i

(3) รากที่สองของค่าความคลาดเคลื่อนกำลังสองเฉลี่ย (Root Mean Square Error: RMSE) เน้นลงโทษค่าคลาดเคลื่อนที่มากผิดปกติ (Outlier) ซึ่งในงานคลังสินค้า ความคลาดเคลื่อนที่สูงอาจส่งผลให้เกิด Overstock อย่างรุนแรง

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}$$

N = จำนวนข้อมูลทั้งหมด

y_i = ค่าจริงตำแหน่ง i

$\hat{y}_i$ = ค่าพยากรณ์ที่ได้ในตำแหน่ง i

(4) ค่าความคลาดเคลื่อนเฉลี่ยเชิงเปอร์เซ็นต์ (Mean Absolute Percentage Error: MAPE) ค่าความคลาดเคลื่อนเฉลี่ยเชิงเปอร์เซ็นต์ (MAPE) เป็นการวัดความคลาดเคลื่อนในรูปแบบเปอร์เซ็นต์ ทำให้สามารถตีความได้ง่ายในเชิงธุรกิจ

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100$$

N = จำนวนข้อมูลทั้งหมด

y_i = ค่าจริงตำแหน่ง i

$\hat{y}_i$ = ค่าพยากรณ์ที่ได้ในตำแหน่ง i

2.2 งานวิจัยที่เกี่ยวข้อง

2.2.1 ตัวแปรในการพยากรณ์สินค้าคงคลัง

งานวิจัยด้านการพยากรณ์ สินค้าคงคลังชี้ว่าความแม่นยำของแบบจำลองขึ้นอยู่กับการคัดเลือกตัวแปรอิสระ (Covariates) ที่จะสะท้อนพฤติกรรมอุปสงค์และอุปทานโดยที่โครงสร้างเวลาอย่างเหมาะสม [23] สำหรับบริบทของผู้จัดจำหน่ายสินค้าไอทีที่มี Intermittent Inbound การกำหนดชุดตัวแปรให้ครอบคลุมเช่น ฤดูกาล, โปรโมชัน และฝั่ง Supply Chain เช่น Lead Time, รอบสั่งซื้อ เป็นสิ่งสำคัญ [5]-[6]

จากการทบทวนวรรณกรรม พบว่ากลุ่มตัวแปรที่ถูกนำมาใช้อย่างแพร่หลายในการพยากรณ์สินค้าคงคลัง สามารถจำแนกได้ดังนี้

(1) ปฏิทิน/ฤดูกาล เช่น เดือน, ไตรมาส, วันหยุด, เทศกาล เพื่อสะท้อนความต้องการที่เกิดซ้ำและแนวโน้มตามฤดูกาล [3]

(2) วงจรสินค้า (Product Lifecycle) เช่นการเปิดตัวของสินค้า (Launch) และการยุติจำหน่าย (EOL) ซึ่งมีผลโดยตรงต่อปริมาณ Inbound ก่อน-หลังการเปลี่ยนรุ่น [3]

(3) ราคาและโปรโมชัน (Price & Promotion) รวมถึงกิจกรรมการตลาดที่จะสร้างความผันผวนและช่วงพีคชั่วคราวของปริมาณเข้า-ออก

(4) Supply & Logistics รอบการสั่งซื้อแบบ Lead Time Backlog/BO ที่มีผลต่อจังหวะและปริมาณเข้า-ออกของสินค้า [23]

(5) พฤติกรรมลูกค้า (Customer/Dealer Behavior) ความถี่ที่สินค้านำเข้าเป็นจำนวนมากซึ่งสัมพันธ์โดยตรงกับลักษณะ Intermittent Demand [5], [6]

(6) ฟีเจอร์อนุกรมเวลา (Time Series Features) Lag ของปริมาณ Inbound ค่าสถิติ Rolling หรือค่า EWMA เพื่อสะท้อนแรงเฉื่อยและแนวโน้มของข้อมูล [9], [18]

(7) คุณลักษณะสินค้า/บรรจุภัณฑ์ (Product & Packaging Attributes) ขนาดสินค้า ปริมาตรกล่อง จำนวนหน่วยต่อพาเลต เพื่อเชื่อมโยงผล [11], [23]

ในกรณีของผู้จัดจำหน่ายสินค้าไอทีที่มีความต้องการแบบ Intermittent Demand และมีลักษณะพฤติกรรมการซื้อของลูกค้า การพยากรณ์ด้วยวิธีการดั้งเดิม Croston, SBA และ TSB ถือเป็นเครื่องมือที่เหมาะสมในการจัดการปัญหาความไม่ต่อเนื่องของอุปสงค์ [5], [7], [8] อย่างไรก็ตาม วิธีการเหล่านี้ยังมีข้อจำกัดเมื่อต้องเผชิญกับความซับซ้อน

จากงานในหลายงานวิจัยการพยากรณ์สินค้าคงคลังได้ใช้ตัวแปรตามตารางที่ 2.1

ตารางที่ 2.1 รายชื่องานวิจัยและตัวแปรที่ใช้ในการพยากรณ์สินค้าคงคลัง

ปีอ้างอิง	การเรียนรู้ของเครื่อง/สถิติ	ตัวแปร	ช่วงเวลาในการพยากรณ์	ข้อมูลที่พยากรณ์
2021 [5]	Intermittent Demand	- intermittent - ขนาดความต้องการ	รายสัปดาห์, รายเดือน	ปริมาณดีมานด์
2013 [7]	Neural Networks (Intermittent)	- Lags - intermittent	รายสัปดาห์	ปริมาณดีมานด์

ตารางที่ 2.1 รายชื่องานวิจัยและตัวแปรที่ใช้ในการพยากรณ์สินค้าคงคลัง (ต่อ)

ปีอ้างอิง	การเรียนรู้ของเครื่อง/สถิติ	ตัวแปร	ช่วงเวลาในการพยากรณ์	ข้อมูลที่พยากรณ์
2017 [8]	NN Single hidden Layer (Intermittent)	- Lag - Rolling - ความถี่ศูนย์	รายสัปดาห์	ปริมาณดีมานด์
2020 [9]	ES-RNN (Hybrid)	- Lag - แนวโน้ม - ฤดูกาล	รายเดือน	ปริมาณดีมานด์
2016 [15]	XGBoost	- ฤดูกาล - โปรโมชัน	รายเดือน	ปริมาณดีมานด์
1997 [18]	LSTM (RNN-based)	- Lag	รายเดือน	ปริมาณดีมานด์

จากตารางที่ 2.1 พบว่าตัวแปรที่ถูกใช้บ่อยที่สุดจะเป็นปฏิทิน, ฤดูกาล และฟีเจอร์อนุกรมเวลา (Lag/Rolling) เนื่องจากเป็นตัวแปรพื้นฐานที่จะสะท้อนโครงสร้างอนุกรมเวลาได้ดี ขณะเดียวกันตัวแปรด้าน พฤติกรรมการสั่งซื้อ ก็มีบทบาทสำคัญโดยเฉพาะกับงานวิจัยที่ศึกษาดีมานด์แบบเว้นช่วง [5], [7], [8] ส่วนงานเชิงลึกเช่น LSTM และ Transformer มักใช้กับตัวแปรภายนอกหลายมิติ เช่น ราคา โปรโมชัน และ Lead Time [11], [15], [18] เพื่อรองรับการพยากรณ์แบบ Multi-Horizon การเลือกตัวแปรไม่เพียงแค่ช่วยเพิ่มความแม่นยำเท่านั้น แต่ยังช่วยให้ผลลัพธ์สามารถนำไปใช้วางแผนพื้นที่คลังได้ตรงกับสภาพจริงของผู้จัดจำหน่ายสินค้าไอที ในกรณีของผู้จัดจำหน่ายสินค้าไอทีที่มีความต้องการแบบ Intermittent Demand และมีลักษณะพฤติกรรมของลูกค้า การพยากรณ์ด้วยวิธีการดั้งเดิมเช่น Croston, SBA และ TSB ถือเป็นเครื่องมือที่เหมาะสมในการจัดการปัญหาความไม่ต่อเนื่องของอุปสงค์ [5], [7], [8] อย่างไรก็ตาม วิธีการเหล่านี้ยังมีข้อจำกัดเมื่อต้องเผชิญกับความซับซ้อนของปัจจัยภายนอกเช่น ฤดูกาล, โปรโมชัน และพฤติกรรมลูกค้า ดังนั้นงานวิจัยสมัยใหม่จึงได้มีการพัฒนา Hybrid Models ที่ผสานวิธีการดั้งเดิมให้เข้ากับ Deep Learning (เช่น LSTM, GRU และ RNN) เพื่อเพิ่มความแม่นยำในการพยากรณ์โดยเฉพาะในบริบทที่ความซับซ้อน ดังนั้นโมเดลที่เป็นพื้นฐานสมการยังไม่เพียงพอต่อการพยากรณ์ แต่สามารถใช้เป็นแนวทางในการเชื่อมโยง ทฤษฎี Inventory Forecasting เข้ากับกรณีศึกษาจริงของผู้จัดจำหน่ายสินค้าไอทีที่ต้องเจอกับปัญหา Intermittent Demand และพฤติกรรมของลูกค้าได้อย่างเหมาะสม

2.2.2 การพยากรณ์ด้วยการเรียนรู้ของเครื่อง

การเรียนรู้ของเครื่อง (Machine Learning) ได้รับความนิยมอย่างมากในการพยากรณ์อนุกรมเวลาและการจัดการสินค้าคงคลัง เนื่องจากสามารถจัดการกับข้อมูลที่มีความซับซ้อน มีตัวแปรหลายมิติ และความสัมพันธ์ที่ไม่เป็นเชิงเส้นได้ดีกว่าวิธีการเชิงสถิติแบบดั้งเดิม โดยเฉพาะกลุ่มของ Tree-Based เช่น Random Forest และ XGBoost [14], [15] รวมถึงโมเดลเชิงลึกอย่าง RNN, LSTM และ GRU [18], [20] ที่สามารถจัดการกับข้อมูลที่มีลำดับเวลาและความสัมพันธ์ซับซ้อนได้ นอกจากนี้ยังมีการพัฒนา Hybrid Models เช่น ES-RNN [9] ที่ผสานวิธีสถิติกับโครงข่ายประสาทเทียมที่ใช้กลไก Attention เพื่อรองรับตัวแปรแบบหลายมิติและการพยากรณ์หลายช่วงเวลา (Multi-horizon) งานวิจัยที่ผ่านมาได้แสดงให้เห็นว่าโมเดล “การเรียนรู้ของเครื่อง” มีความสามารถในการพยากรณ์ที่แม่นยำกว่าวิธีเชิงสถิติ โดยเฉพาะในกรณีที่มีตัวแปรภายนอกจำนวนมากเช่น ฤดูกาล โปรโมชัน และตัวแปรด้าน Supply Chain ซึ่งสอดคล้องกับบริบทของงานวิจัยนี้ที่มุ่งเน้นการพยากรณ์ Inbound Inventory เพื่อการจัดการพื้นที่สินค้าคงคลังอย่างมีประสิทธิภาพ จากทั้งหมดที่กล่าวมาโมเดลการเรียนรู้ของเครื่องนำไปใช้ในการพยากรณ์หลากหลายรูปแบบในตารางที่ 2.2 งานวิจัยที่ใช้เทคนิคการเรียนรู้ของเครื่อง

ตารางที่ 2.2 งานวิจัยที่ใช้เทคนิคการเรียนรู้ของเครื่อง

ปี	เทคนิคใช้ในการวิจัย	จุดประสงค์ของงานวิจัย	ค่าที่ใช้ในการประเมินโมเดล	ผลลัพธ์	ข้อเสนอแนะงานวิจัย
2025 [24]	Hybrid Stacking (LSTM, XGBoost, RF, LR) โดยใช้ Linear Regression รวมผล	พยากรณ์โหลดไฟฟ้าอาคารพาณิชย์/อาคารเรียน โดยใช้จำนวนผู้ใช้อาคารแบบเรียลไทม์	RMSE, MAE, MSE, NRMSE	อาคารพาณิชย์: RMSE=2.99 kW, MAE=2.18 kW, อาคารเรียน RMSE=4.48 kW, MAE = 2.85 kW ลดต้นทุน 17.42%, 33.40%	ยังไม่ได้ปรับตารางการใช้พลังงานช่วง Off-hours ให้เหมาะสมที่สุด
2024 [25]	Ensemble (RNN-LSTM) เทียบกับ Prophet, SVR, CNN, GRU	พยากรณ์ความต้องการ Steel wire mesh เพื่อบริหารสินค้าคงคลัง	RMSE, MAPE	โมเดลดีที่สุด (RNN-LSTM) RMSE=45,931, MAPE=1.20%ลดต้นทุน	ใช้ข้อมูลแบบ ตัวแปรเดียว (Univariate) ไม่รวมปัจจัยภายนอก เช่น ราคาเหล็ก หรือ GDP
2023 [26]	LSTM-Attention-LSTM (Encoder–Decoder + Attention)	พยากรณ์อนุกรมเวลาหลายตัวแปร เช่น คุณภาพอากาศ/ไฟฟ้า/หุ้น/ทองคำ	RMSE, MAPE	RMSE=0.013, MAPE=0.76%, Gold: RMSE=0.005, MAPE=0.65%, Air: RMSE=0.091, MAPE=16.76%	ประสิทธิภาพลดลงมากเมื่อข้อมูลมี Missing values สูง (เช่นชุด Air) สะท้อนความไวต่อคุณภาพข้อมูล
2022 [27]	XGBoost เทียบกับ 9 วิธี ML และ 3 วิธีดั้งเดิม (เช่น SARIMA, ES)	พยากรณ์ยอดขายสินค้าพืชสวน (ดอกทิวลิป/ไม้ตัดดอก/ไม้กระถาง)	RMSE, MAPE, sMAPE	XGBoost ให้ค่าความคลาดเคลื่อนต่ำกว่าแนวทางอื่นส่วนใหญ่ โดยในบางกรณี MAPE อยู่ที่ 23.98% และ 52.87% ตามประเภทสินค้า	ยอดขายไม่สะท้อนดีมานด์จริงทั้งหมด (เช่น สินค้าหมดสต็อกไม่ได้ถูกบันทึก) และจับช่วงพุ่งของโควิด-19 ได้ยาก

ตารางที่ 2.2 งานวิจัยที่ใช้เทคนิคการเรียนรู้ของเครื่อง (ต่อ)

ปี	เทคนิคใช้ในการวิจัย	จุดประสงค์ของงานวิจัย	ค่าที่ใช้ในการประเมินโมเดล	ผลลัพธ์	ข้อเสนอแนะงานวิจัย
2022 [28]	Seq2seq trimmed และ Transformer (ดัดแปลงสำหรับ time series)	พยากรณ์ยอดขายร้านของชำ Corporación Favorita รายวัน/รายสาขา/รายสินค้า	RMSLE, RMSWLE, MALE	ให้ค่า RMSLE อยู่ในช่วง 0.538–0.542 ซึ่งต่ำกว่าค่า baseline อย่างมีนัยสำคัญ	Transformer มีความผันผวนของ error ในบางวันของช่วงพยากรณ์ และข้อมูลขาดตัวแปรสินค้าคงคลัง ทำให้แยกไม่ออกระหว่าง “ไม่มีคนซื้อ” กับ “ของหมด”
2020 [29]	Stacked LSTM (Multi-layer) + Grid Search	พยากรณ์ความต้องการสินค้าบริษัทเฟอร์นิเจอร์	RMSE, SMAPE	ให้ค่า SMAPE = 10.85% ซึ่งต่ำกว่า SVM (11.22%) แสดงถึงความสามารถในการจับรูปแบบข้อมูลเชิงลำดับ	อนุกรมเวลามีความผันผวนสูง ทำให้ความแม่นยำในระยะยาวมีข้อจำกัด

จากการทบทวนงานวิจัยที่ผ่านมา พบว่าแบบจำลองเชิงเส้นสถิติเช่น ARIMA หรือ Exponential Smoothing แม้ว่าจะสามารถอธิบายแนวโน้มและฤดูกาลได้ดี แต่มีข้อจำกัดเมื่อเจอข้อมูลที่มีความต้องการไม่สม่ำเสมอ (intermittent Demand) และไม่สามารถรองรับตัวแปรภายนอก เช่น ปริมาตรการจัดเก็บ (CBM) งานวิจัยนี้จึงเลือกใช้และเปรียบเทียบแบบจำลองที่มีศักยภาพสูงในกลุ่ม Machine Learning และ Deep Learning ได้แก่

(1) Random Forest เน้นการใช้ฟีเจอร์อนุกรมเวลา (Lag, Rolling) และตัวปัจจัยภายนอก (CBM, Seasonality, Proxy Promotion) ช่วยในการพยากรณ์และลด Overfitting ได้

(2) XGBoost คล้ายกับ Random Forest แต่จะใช้วิธีการ Boosting ปรับปรุง Error ของต้นไม้ก่อนหน้า ทำให้มีความแม่นยำที่สูง

(3) Recurrent Neural Network (RNN) โมเดลพื้นฐานสำหรับข้อมูลของอนุกรมเวลา สามารถเรียนรู้ลำดับและความสัมพันธ์ต่อเนื่องของข้อมูล

(4) Long Short-Term (LSTM) พัฒนาต่อจาก RNN เพื่อแก้ปัญหา Vanishing Gradient สามารถจดจำความสัมพันธ์ระยะยาวได้ดี

(5) LSTM + Attention เพิ่มกลไกล Attention เข้าไปใน LSTM ทำให้โมเดลสามารถโฟกัสช่วง เวลาที่สำคัญต่อการพยากรณ์ เช่น สิ้นเดือนหรือสิ้นไตรมาสที่ยอดขายจะพุ่งสูง

(6) Gated Recurrent Unit (GRU) เป็นโครงสร้างแบบ RNN ที่คล้าย LSTM แต่มีโครงสร้างง่ายกว่าและมีพารามิเตอร์น้อยกว่า จึงฝึกฝนได้เร็วกว่า

ซึ่งการเลือกโมเดลดังกล่าวทำให้สามารถตอบโจทย์ทั้งในเชิงวิชาการ (เปรียบเทียบประสิทธิภาพ ML/DL กับสถิติ/Intermittent) และในเชิงธุรกิจ (ลด Overstock, วางแผนพื้นที่ CBM) ซึ่งสอดคล้องกับบริบทของบริษัทผู้จัดจำหน่ายสินค้าไอทีที่ศึกษา

บทที่ 3
แผนงานการปฏิบัติงานและขั้นตอนการดำเนินงาน

จากการศึกษางานวิจัย แนวคิด และทฤษฎีที่เกี่ยวข้องผู้ศึกษาได้นำความรู้มาประยุกต์ใช้กับการพยากรณ์ปริมาตรสินค้าคงคลังขาเข้ากลุ่มสินค้าไอที โดยใช้เทคนิคการเรียนรู้ของเครื่อง โดยมีขั้นตอนการดำเนินการดังนี้

3.1 การรวบรวมข้อมูล

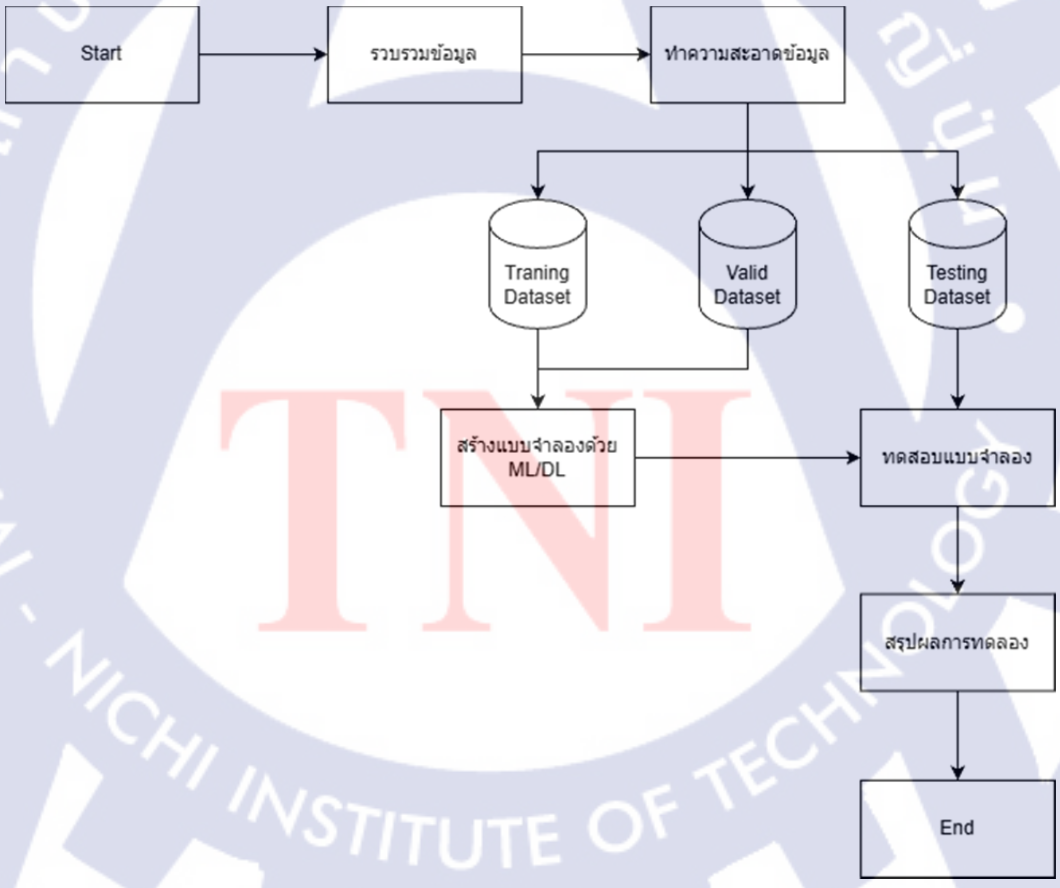
3.2 การทำความสะอาดข้อมูล (Data Cleansing) และเตรียมข้อมูล

3.3 ขั้นตอนการแบ่งชุดข้อมูลและการฝึกฝนโมเดล

3.4 การประเมินผล

3.5 เกณฑ์การประเมินผลการพยากรณ์ (Evaluation Criteria and KPI)

3.6 สรุปผลการทดลอง



รูปที่ 3.1 ขั้นตอนการดำเนินการวิจัย

ตารางที่ 3.1 ระยะเวลาการดำเนินการวิจัย

การดำเนินการ	ระยะเวลา								
	ปี 2568							ปี 2569	
	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.
ขั้นตอนที่ 1 การรวบรวมข้อมูล									
ขั้นตอนที่ 2 การทำความสะอาดข้อมูล (Data Cleansing) และเตรียมข้อมูล									
ขั้นตอนที่ 3 ดำเนินการฝึกฝนโมเดล									
ขั้นตอนที่ 4 เกณฑ์การประเมินผลการพยากรณ์									
ขั้นตอนที่ 5 สรุปผลการทดลอง									

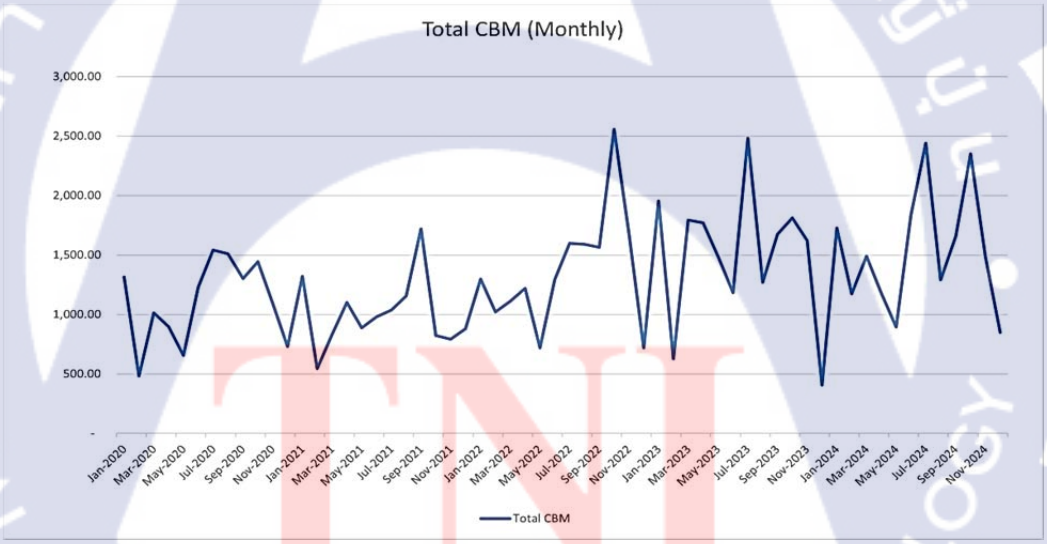
3.1 การรวบรวมข้อมูล

งานวิจัยนี้ใช้ข้อมูลจากคลังสินค้าของผู้จัดจำหน่ายสินค้าไอทีรายใหญ่ในประเทศไทย โดยจะรายงานในเอกสารจะไม่เปิดเผยข้อมูลชื่อแบรนด์และบริษัท ซึ่งทางผู้วิจัยได้มีการเซ็นเอกสารข้อตกลงไม่เปิดเผยข้อมูลหรือ NDA (Non-Disclosure Agreement) เพื่อคุ้มครองข้อมูลทางธุรกิจ จึงขอเรียกโดยย่อว่า บริษัทผู้จัดจำหน่าย และแบรนด์ ที่ผู้วิจัยนำมาใช้เพื่อพยากรณ์โมเดลที่ถูกสร้างขึ้นนั้นจะสอดคล้องกับปัจจัยที่เกี่ยวข้องมากที่สุด ซึ่งข้อมูลที่รวบรวมมานั้นจะอยู่ในช่วง พ.ศ. 2563-2567 โดยแต่ข้อมูลมีแหล่งที่มาจากปริมาณรับสินค้า (Inbound) ซึ่งได้ผ่านการคัดเลือกและตรวจสอบความถูกต้องแล้ว โดยมีตัวแปรหลักได้แก่ QTY และ Dimension สินค้าที่เป็น CBM (ปริมาตรรวม) ไฟล์ต้นฉบับอยู่ในรูปแบบไฟล์ xlsx ตามที่แนบไว้ โดยงานวิจัยนี้กำหนดกลยุทธ์การสุ่มตัวอย่าง (Sampling Strategy) โดยเลือกใช้ข้อมูลทั้งหมด 146,149 รายการ เนื่องจากข้อมูลมีลักษณะความไม่ต่อเนื่อง (Intermittent Demand) การใช้ข้อมูลครบถ้วนจะช่วยสะท้อนความหลากหลายของรูปแบบความต้องการได้ดีกว่าการเลือกเฉพาะบางส่วน ทั้งนี้ เพื่อความถูกต้องในการประเมินแบบจำลอง ข้อมูลจะถูกแบ่งออกเป็น ชุด (Training Set), ชุดตรวจสอบ (Validation Set) และชุดทดสอบ (Test set) ตามลำดับเวลา โดยจะใช้สัดส่วน ปี 2563-

2565 สำหรับ Training, ปี 2566 สำหรับ Validation และ ปี 2567 สำหรับ Testing โดยจะยึดตามหลักการ Time-Ordered Hold-out Split เพื่อป้องกันการรั่วไหลของข้อมูลในอนาคต (Data Leakage) และเพิ่มความน่าเชื่อถือของผลการพยากรณ์

จากข้อมูลที่ถูกรวบรวมและแสดงในรูปที่ 3.1 สามารถสังเกตได้ว่าปริมาณการรับเข้า CBM ไม่สมดุลกันในหลายช่วงเวลา โดยบางเดือนมีการรับเข้าสินค้ามากจนก่อให้เกิดความเสี่ยงต่อ Overstock ภายในคลังสินค้า นอกจากนี้ยังพบว่ามีการเกิดจุดพีคตามฤดูกาลเช่น การรับเข้าที่สูงในช่วงปลายปีจากการนำเข้าสินค้ารุ่นใหม่ ข้อมูล CBM ยังชี้ให้เห็นถึงความท้าทายด้านการบริหารจัดการพื้นที่ ซึ่งการรวบรวมข้อมูลในขั้นตอนนี้จึงเป็นรากฐานสำคัญของการวิจัย โดยจะถูกนำไปใช้ในการทำความสะอาดและการเตรียมข้อมูลในหัวข้อถัดไป และถูกประมวลผลด้วยแบบจำลองการพยากรณ์โดยโมเดลสมัยใหม่ที่ใช้เทคนิคการเรียนรู้ของเครื่องและการเรียนรู้เชิงลึก เพื่อเปรียบเทียบและประเมินประสิทธิภาพการพยากรณ์ในช่วง 1, 3 และ 6 เดือนล่วงหน้า

3.2 การทำความสะอาดข้อมูล (Data Cleansing) และเตรียมข้อมูล



รูปที่ 3.2 ปริมาตรสินค้าแบรนด์หนึ่งเข้าคลัง

3.2.1 การทำความสะอาดและจัดเตรียมข้อมูล

ข้อมูลที่ใช้ในการวิจัยนี้เป็นข้อมูลดิบระดับรายการรายวัน (Daily Transaction) ซึ่งได้รับจากบริษัทผู้จัดจำหน่ายสินค้าไอทีรายใหญ่ในประเทศไทย ครอบคลุมช่วงเวลา เดือนมกราคม พ.ศ. 2563 ถึงเดือนธันวาคม พ.ศ. 2567 โดยมุ่งเน้นเฉพาะข้อมูลการรับสินค้าเข้า (Inbound) เพื่อใช้เป็นฐาน

ในการพยากรณ์ปริมาตรสินค้าขาเข้าในหน่วยลูกบาศก์เมตร (CBM) ข้อมูลดังกล่าวประกอบด้วยรายละเอียดสำคัญ ได้แก่ วันที่รับสินค้า รหัสสินค้า (Part) ปริมาณสินค้า (Quantity) และปริมาตรรวมของรายการ (TotalM3) อย่างไรก็ตาม ข้อมูลดิบยังปรากฏรายการบางประเภทที่มิได้สะท้อนภาระงานรับเข้าที่เกิดขึ้นจากการไหลเข้าของสินค้าในเชิงธุรกิจโดยตรง เช่น รายการโอนย้ายภายในคลัง (Internal Transfer) ซึ่งหากนำมารวมในการสร้างแบบจำลองจะทำให้สัญญาณของข้อมูลเป้าหมายคลาดเคลื่อนจากความเป็นจริงและเพิ่มความผันผวนที่ไม่จำเป็น ดังนั้น ผู้วิจัยจึงดำเนินการคัดกรองและตัดรายการที่ไม่เกี่ยวข้องออก เพื่อคงไว้เฉพาะธุรกรรมที่สะท้อนการเคลื่อนไหวของสินค้าขาเข้าที่แท้จริง ก่อนเข้าสู่กระบวนการทำความสะอาดข้อมูลและการเตรียมข้อมูลสำหรับการสร้างแบบจำลอง

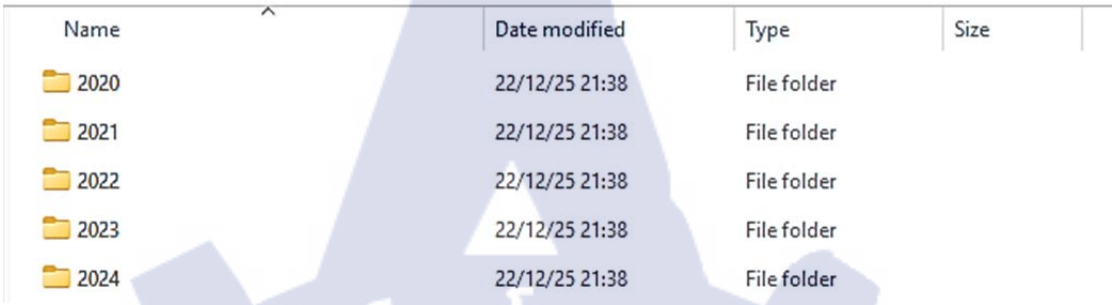
เนื่องจากบริษัทจัดเก็บไฟล์ข้อมูลในลักษณะ แยกย่อยเป็นหลายโฟลเดอร์ย่อย (Subfolder) และไม่มีรูปแบบการจัดเรียงที่เป็นมาตรฐานเดียวกัน

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	Goods Receive Date	Goods Receive No	PO NO	Type	Vendor Name	Part	Qty	Request By	Segment	Sign [Vendor] Time	Receive in Systems	SUM	KPI On time	Width[CM]	Length[CM]	Hight[CM]	Total[M3]	Fee[THB]	Remark
2	18/1/2024 9:12	1		W/R	Vendor1	Part1	1	Employee A	Segment A					57	18	66	0.07		
3	ToTal	1		W/R				Employee A	Segment A	9:00	9:12	0:12	1				0.07	100.00	
4	18/1/2024 9:12	2		W/R	Vendor1	Part2	6	Employee B	Segment B					35	25	0.1	0.00		
5	18/1/2024 9:12	2		W/R	Vendor1	Part2	4	Employee B	Segment B					35	25	0.1	0.00		
6	ToTal	2		W/R				Employee B	Segment B	9:10	9:12	0:02	1				0.00	100.00	
7	18/1/2024 9:15	3		W/R	Vendor2	Part3	1	Employee C	Segment C					35	25	0.1	0.00		
8																			
9																			

รูปที่ 3.3 ตัวอย่างข้อมูลการรับเข้า

ผู้วิจัยจึงดำเนินการจัดระเบียบไฟล์ข้อมูล Handling-IN ใหม่ให้เป็นโครงสร้างที่สืบค้นและอ้างอิงได้สะดวก เพื่อรองรับการดึงข้อมูล การตรวจสอบความครบถ้วน และการรวมผลในขั้นตอนถัดไป กระบวนการดึงและรวมข้อมูลเลือกใช้ภาษาโปรแกรม Python เนื่องจากปริมาณข้อมูลมีขนาดค่อนข้างมาก กล่าวคือ ไฟล์ข้อมูล Excel หนึ่งไฟล์มีจำนวนแถวมากกว่า 1,000 แถว และโดยเฉลี่ยมีประมาณ 25 ไฟล์ต่อเดือน ส่งผลให้ในหนึ่งปีมีไฟล์ข้อมูลราว 300 ไฟล์ และมีจำนวนระเบียนรวมมากกว่า 300,000 แถวต่อปี นอกจากนี้ ชื่อแผ่นงาน (Sheet) ภายในไฟล์ Excel ยังมีการตั้งชื่อไม่เป็นรูปแบบเดียวกัน โดยมักระบุวันเดือนปีร่วมกับรหัสเอกสาร เช่น “200121A Handling-IN” จึงทำให้การรวมข้อมูลด้วยวิธีการดำเนินงานแบบแมนนวลมีความเสี่ยงต่อความผิดพลาดและการตกหล่นของข้อมูลได้ง่าย ดังนั้น การใช้ Python เพื่ออ่านไฟล์แบบหลายแฟ้ม (Batch Processing) และสกัดข้อมูลจากหลายแผ่นงานภายใต้ชื่อที่แตกต่างกัน พร้อมรวมให้อยู่ในโครงสร้างข้อมูลเดียวกันจึงมีความเหมาะสม โดยช่วยลดภาระงานซ้ำซ้อน เพิ่มความรวดเร็วในการเตรียมข้อมูล และทำให้ขั้นตอนการประมวลผลสามารถทำซ้ำได้อย่างเป็นระบบ ทั้งนี้ โครงสร้างโฟลเดอร์ที่ใช้จัดเก็บไฟล์รายปีแสดงไว้ในรูปที่ 3.4 เพื่อใช้เป็นกรอบอ้างอิงในการจัดการข้อมูลตลอดทั้งงานวิจัย ซึ่งการจัดโครงสร้างไฟล์ข้อมูลและการประมวลผล

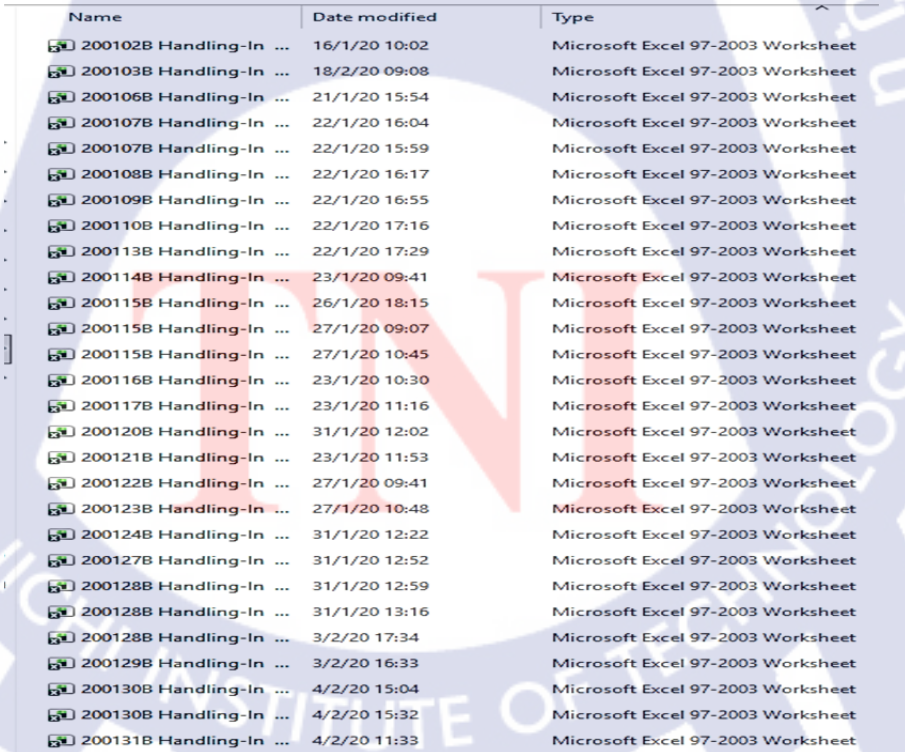
ด้วย Python มีส่วนช่วยลดความเสี่ยงจากการตกหล่นของข้อมูล เพิ่มความสอดคล้องของกระบวนการเตรียมข้อมูล และสนับสนุนความสามารถในการทำซ้ำของงานวิจัย (reproducibility) ภายใต้เงื่อนไขชุดข้อมูลเดียวกัน



Name	Date modified	Type	Size
2020	22/12/25 21:38	File folder	
2021	22/12/25 21:38	File folder	
2022	22/12/25 21:38	File folder	
2023	22/12/25 21:38	File folder	
2024	22/12/25 21:38	File folder	

รูปที่ 3.4 โฟลเดอร์ที่จัดเก็บไฟล์แต่ละปี

ซึ่งภายในโฟลเดอร์นั้นจะถูกเก็บข้อมูลการรับเข้า (Inbound) โดยข้อมูล 1 ไฟล์จะเป็นรายงานการรับเข้าต่อ 1 วันดังรูปที่ 3.5 ไฟล์การรับเข้า (Handling Inbound)



Name	Date modified	Type
200102B Handling-In ...	16/1/20 10:02	Microsoft Excel 97-2003 Worksheet
200103B Handling-In ...	18/2/20 09:08	Microsoft Excel 97-2003 Worksheet
200106B Handling-In ...	21/1/20 15:54	Microsoft Excel 97-2003 Worksheet
200107B Handling-In ...	22/1/20 16:04	Microsoft Excel 97-2003 Worksheet
200107B Handling-In ...	22/1/20 15:59	Microsoft Excel 97-2003 Worksheet
200108B Handling-In ...	22/1/20 16:17	Microsoft Excel 97-2003 Worksheet
200109B Handling-In ...	22/1/20 16:55	Microsoft Excel 97-2003 Worksheet
200110B Handling-In ...	22/1/20 17:16	Microsoft Excel 97-2003 Worksheet
200113B Handling-In ...	22/1/20 17:29	Microsoft Excel 97-2003 Worksheet
200114B Handling-In ...	23/1/20 09:41	Microsoft Excel 97-2003 Worksheet
200115B Handling-In ...	26/1/20 18:15	Microsoft Excel 97-2003 Worksheet
200115B Handling-In ...	27/1/20 09:07	Microsoft Excel 97-2003 Worksheet
200115B Handling-In ...	27/1/20 10:45	Microsoft Excel 97-2003 Worksheet
200116B Handling-In ...	23/1/20 10:30	Microsoft Excel 97-2003 Worksheet
200117B Handling-In ...	23/1/20 11:16	Microsoft Excel 97-2003 Worksheet
200120B Handling-In ...	31/1/20 12:02	Microsoft Excel 97-2003 Worksheet
200121B Handling-In ...	23/1/20 11:53	Microsoft Excel 97-2003 Worksheet
200122B Handling-In ...	27/1/20 09:41	Microsoft Excel 97-2003 Worksheet
200123B Handling-In ...	27/1/20 10:48	Microsoft Excel 97-2003 Worksheet
200124B Handling-In ...	31/1/20 12:22	Microsoft Excel 97-2003 Worksheet
200127B Handling-In ...	31/1/20 12:52	Microsoft Excel 97-2003 Worksheet
200128B Handling-In ...	31/1/20 12:59	Microsoft Excel 97-2003 Worksheet
200128B Handling-In ...	31/1/20 13:16	Microsoft Excel 97-2003 Worksheet
200128B Handling-In ...	3/2/20 17:34	Microsoft Excel 97-2003 Worksheet
200129B Handling-In ...	3/2/20 16:33	Microsoft Excel 97-2003 Worksheet
200130B Handling-In ...	4/2/20 15:04	Microsoft Excel 97-2003 Worksheet
200130B Handling-In ...	4/2/20 15:32	Microsoft Excel 97-2003 Worksheet
200131B Handling-In ...	4/2/20 11:33	Microsoft Excel 97-2003 Worksheet

รูปที่ 3.5 ไฟล์การรับเข้า (Handling Inbound)

เพื่อให้เข้าถึงข้อมูลปริมาณมาก (Big Data) เป็นไปได้อย่างมีประสิทธิภาพ ผู้วิจัยได้ดำเนินการขั้นตอนทางเทคนิคดังนี้

(1) Storage Restructuring ทำการปรับเปลี่ยนโครงสร้างโฟลเดอร์ (Folder Hierarchy) ใหม่ เพื่อสนับสนุนการเขียน Query ให้สามารถเข้าถึงข้อมูลในทุก Sub Folder ได้อย่างมีประสิทธิภาพ

(2) การจัดการโครงสร้าง Metadata และ Header ทางผู้วิจัยได้กำหนด Work Sheet ที่มีความเฉพาะเจาะจงเช่น “YYMMDDA Handling-IN” และทำการ Header Skipping ที่ไม่เกี่ยวข้องออกไป เพื่อให้สามารถเข้าถึงข้อมูลดิบได้อย่างถูกต้อง

(3) Data Purging โดยชุดคำสั่ง Python จะเลือกเฉพาะ Column ที่เป็นตัวแปรสำคัญสำหรับการพยากรณ์ (Feature Selection) ได้แก่วันที่รับเข้า รหัสสินค้า (Part), ปริมาณสินค้า (QTY), และปริมาตรสินค้า โดยทำการกรองข้อมูลเฉพาะกลุ่มสินค้าเป้าหมาย เพื่อให้ได้ชุดข้อมูลที่สะอาดและตรงประเด็นสำหรับการพยากรณ์

(4) Cross-System Validations (การตรวจสอบความถูกต้องข้ามระบบ) เพื่อให้มั่นใจในความแม่นยำ (Data Integrity) ของชุดคำสั่ง ผู้วิจัยได้ทำการตรวจสอบความถูกต้องด้วยการเปรียบเทียบผลลัพธ์กับระบบ Excel โดยใช้ SUMIFS เพื่อตรวจสอบผลรวมของปริมาตรสินค้า (Dimension) ในแต่ละเดือน และมีการวิเคราะห์แนวโน้มผ่านกราฟ (Trend Analysis) เพื่อยืนยันข้อมูลที่ได้มาจากการ Query มีความสอดคล้องกับพฤติกรรมข้อมูลก่อนที่จะนำไปฝึกโมเดล

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Date	Qty	Width(CM	Length(Cl	Hight(CM	Total(M	Total Dimens	Day	Mon(	Yea	ayOfW	WeekendFlag	DoubleDay	LastMont
2	02/01/2020	1.00	49.00	20.00	48.00	0.05	47,040.00	2	1	2020	3	0	0	0
3	02/01/2020	1.00	45.00	7.00	30.00	0.01	9,450.00	2	1	2020	3	0	0	0
4	02/01/2020	1.00	47.00	19.00	34.00	0.03	30,362.00	2	1	2020	3	0	0	0
5	02/01/2020	1.00	48.00	7.00	30.00	0.01	10,080.00	2	1	2020	3	0	0	0
6	02/01/2020	1.00	48.00	8.00	35.00	0.01	13,440.00	2	1	2020	3	0	0	0
7	02/01/2020	1.00	11.00	36.00	14.00	0.01	5,544.00	2	1	2020	3	0	0	0
8	02/01/2020	1.00	42.00	30.00	6.00	0.01	7,560.00	2	1	2020	3	0	0	0
9	02/01/2020	1.00	7.00	55.00	30.00	0.01	11,550.00	2	1	2020	3	0	0	0
10	02/01/2020	140.00	25.00	35.00	0.10	0.01	12,250.00	2	1	2020	3	0	0	0
11	02/01/2020	140.00	25.00	35.00	0.10	0.01	12,250.00	2	1	2020	3	0	0	0
12	02/01/2020	20.00	29.00	40.00	50.00	1.16	1,160,000.00	2	1	2020	3	0	0	0
13	02/01/2020	5.00	48.00	7.00	32.00	0.05	53,760.00	2	1	2020	3	0	0	0
14	02/01/2020	5.00	29.00	40.00	50.00	0.29	290,000.00	2	1	2020	3	0	0	0
15	02/01/2020	50.00	39.00	29.00	50.00	2.83	2,827,500.00	2	1	2020	3	0	0	0
16	02/01/2020	50.00	39.00	29.00	50.00	2.83	2,827,500.00	2	1	2020	3	0	0	0
17	02/01/2020	100.00	39.00	29.00	50.00	5.66	5,655,000.00	2	1	2020	3	0	0	0
18	02/01/2020	267.00	38.00	26.00	1.00	0.26	263,796.00	2	1	2020	3	0	0	0
19	02/01/2020	219.00	36.30	8.30	8.60	0.57	567,449.59	2	1	2020	3	0	0	0
20	02/01/2020	4.00	48.00	7.00	32.00	0.04	43,008.00	2	1	2020	3	0	0	0
21	02/01/2020	5.00	29.00	40.00	50.00	0.29	290,000.00	2	1	2020	3	0	0	0
22	02/01/2020	10.00	48.00	14.00	32.00	0.22	215,040.00	2	1	2020	3	0	0	0
23	02/01/2020	150.00	39.00	50.00	29.00	8.48	8,482,500.00	2	1	2020	3	0	0	0
24	02/01/2020	1.00	56.00	51.00	15.00	0.04	42,840.00	2	1	2020	3	0	0	0

รูปที่ 3.6 ตัวอย่างข้อมูลหลังการทำความสะอาด

3.2.2 การแบ่งส่วนข้อมูลสำหรับการพยากรณ์

หลังจากการทำความสะอาดข้อมูลเรียบร้อยแล้ว ข้อมูลจะถูกแบ่งออกเป็นตามลำดับเวลาด้วยวิธีการ Time-Ordered Hold-out Split เพื่อหลีกเลี่ยงปัญหา Data Leakage ซึ่งเป็นหลักสำคัญของการพยากรณ์ ความแตกต่างจาก Random Split คือข้อมูลในอนาคตจะต้องไม่ถูกนำไปใช้ใน

การฝึกฝนโมเดล โดยสัดส่วนของงานวิจัยนี้จะแบ่งข้อมูลออกเป็น 3 ส่วนได้แก่ ปี 2563- 2565 สำหรับ Training Set, ปี 2566 สำหรับ Validation Set และ ปี 2567 สำหรับ Testing Set อย่างไรก็ตาม เนื่องจากข้อมูลอนุกรมเวลามีลักษณะตามเดือน (Monthly Time Series) การแบ่งข้อมูลจะใช้วิธี ตัดตามสิ้นเดือน (Monthly Closure) เพื่อป้องกันการรั่วไหลของข้อมูลภายในเดือนเดียวกัน ดังนั้นการแบ่งสัดส่วน จะถูกใช้เป็นแนวคิดในการแบ่งข้อมูล แต่จำนวนสัดส่วนจริงอาจมีการคาดเคลื่อนเล็กน้อยตามข้อมูลจริง ทั้งนี้ยังคงรักษาลำดับตามเวลาเพื่อความถูกต้องของการพยากรณ์ และในส่วนของ Testing Set ถูกกำหนดให้เป็น 12 เดือนสุดท้ายของชุดข้อมูลทั้งหมด เพื่อการใช้งานจริง โดยงานวิจัยนี้ไม่ใช้การแบ่งข้อมูลแบบสุ่ม (Random Split) เนื่องจากข้อมูลมีลักษณะเป็นอนุกรมเวลา (Time Series) ซึ่งการสุ่มอาจจะทำให้เกิดปัญหาข้อมูลรั่วไหล (Data Leakage) และไม่สะท้อนรูปแบบการใช้งานจริงในธุรกิจคลังสินค้า ผู้วิจัยจึงเลือกใช้การแบ่งข้อมูลตามลำดับเวลา (Time-Ordered Hold-out Split) เพื่อให้การฝึก ทดสอบ และประเมินโมเดลสอดคล้องกับสถานการณ์จริงที่ใช้ข้อมูลในอดีตในการพยากรณ์อนาคต

3.2.3 กลยุทธ์การพยากรณ์หลายระยะเวลา (Multi-Horizon Forecasting Strategy)

งานวิจัยนี้ใช้แนวทาง Direct Multi-Horizon Forecasting โดยทำการฝึกโมเดลแยกกันสำหรับแต่ละระยะเวลาการพยากรณ์ (t+1, t+3, t+6) ซึ่งตัวแปรเป้าหมายจะถูกกำหนดตามระยะเวลาที่ต้องการพยากรณ์โดยตรง วิธีดังกล่าวช่วยลดปัญหาการสะสมความคลาดเคลื่อนที่อาจเกิดขึ้นในวิธี Recursive Forecasting และทำให้การประเมินผลในแต่ละช่วงเวลามีความเป็นอิสระต่อกัน

3.3 ขั้นตอนการแบ่งชุดข้อมูลและการฝึกฝนโมเดล

3.3.1 การแบ่งชุดข้อมูล

ข้อมูลดิบของงานวิจัยครอบคลุมช่วงเวลา เดือนมกราคม พ.ศ. 2563 ถึงเดือนธันวาคม พ.ศ. 2567 โดยข้อมูลเริ่มต้นอยู่ในระดับรายวัน (Daily Level) และถูกสรุปผลเป็นรายเดือน (Monthly Aggregation) เพื่อให้สอดคล้องกับลักษณะข้อมูลอนุกรมเวลาและกรอบการพยากรณ์แบบหลายระยะเวลา ทั้งนี้ การกำหนดขอบเขตของแต่ละชุดข้อมูลจะยึด “ช่วงเดือนเต็ม” เป็นหน่วยอ้างอิง เพื่อให้การคำนวณตัวแปรเชิงเวลาและการประเมินผลมีความสอดคล้องกันตลอดการทดลอง

การแบ่งชุดข้อมูลใช้แนวทาง Time-Ordered Hold-out Split โดยรักษาลำดับเวลาอย่างเคร่งครัด แบ่งออกเป็น 3 ส่วน ดังนี้

(1) Training Set: ครอบคลุมช่วง เดือนมกราคม พ.ศ. 2563 ถึงเดือนธันวาคม พ.ศ. 2565 ใช้สำหรับฝึกสอนแบบจำลองให้เรียนรู้รูปแบบแนวโน้มและฤดูกาลจากข้อมูลอดีต

(2) Validation Set: ครอบคลุมช่วง เดือนมกราคม พ.ศ. 2566 ถึงเดือนธันวาคม พ.ศ. 2566 ใช้สำหรับตรวจสอบประสิทธิภาพและเลือกค่าพารามิเตอร์ที่เหมาะสมของแบบจำลองภายใต้ข้อมูลที่ไม่ถูกใช้ในการฝึก

(3) Testing Set: ครอบคลุมช่วง เดือนมกราคม พ.ศ. 2567 ถึงเดือนธันวาคม พ.ศ. 2567 ใช้สำหรับประเมินสมรรถนะของแบบจำลองในข้อมูลที่ไม่เคยถูกเห็นมาก่อน เพื่อสะท้อนการใช้งานจริงเชิงธุรกิจ

แนวทางการแบ่งข้อมูลดังกล่าวมีจุดมุ่งหมายเพื่อจำลองสถานการณ์การพยากรณ์ในบริบทคลังสินค้าจริง โดยหลีกเลี่ยงการสลับลำดับเวลา และลดความเสี่ยงของการรั่วไหลของข้อมูล (Data Leakage) ที่อาจทำให้ผลการประเมินสูงกว่าความเป็นจริง

3.3.2 เครื่องมืองานวิจัย

ในการดำเนินการประมวลผลและพัฒนาแบบจำลองพยากรณ์ งานวิจัยนี้กำหนดสภาพแวดล้อมการทดลองโดยอาศัยทั้งซอฟต์แวร์และฮาร์ดแวร์ร่วมกัน เพื่อให้กระบวนการทดลองมีความเป็นระบบ ตรวจสอบย้อนกลับได้ และลดความคลาดเคลื่อนที่อาจเกิดจากความแตกต่างของเครื่องมือที่ใช้ ในส่วนของซอฟต์แวร์ งานวิจัยใช้ภาษา Python 3.11.9 เป็นเครื่องมือหลักตลอดกระบวนการ ตั้งแต่การรวบรวมและจัดรูปข้อมูล การสร้างตัวแปรเชิงเวลา การฝึกสอนแบบจำลอง ไปจนถึงการประเมินผล โดยเลือกใช้ไลบรารีตามบทบาทของงานอย่างเหมาะสม ได้แก่ Pandas และ NumPy สำหรับการจัดการข้อมูลเชิงตารางและการคำนวณเชิงตัวเลข, Scikit-learn สำหรับการพัฒนาแบบจำลองในกลุ่ม Machine Learning รวมถึงขั้นตอนการประเมินผลและการปรับแต่งไฮเปอร์พารามิเตอร์ และใช้เฟรมเวิร์ก Tenso rFlow/ Keras และ PyTorch สำหรับการสร้างและฝึกสอนแบบจำลองในกลุ่ม Deep Learning ที่ต้องอาศัยการคำนวณเชิงเมทริกซ์จำนวนมาก สำหรับการตรวจสอบแนวโน้มของข้อมูลและการนำเสนอผลในรูปกราฟ

ในส่วนของฮาร์ดแวร์ การทดลองทั้งหมดถูกดำเนินการบนเครื่องคอมพิวเตอร์ที่ประกอบด้วยหน่วยประมวลผลกลาง Intel Core i7-12700F (12th Gen Alder Lake) ความเร็วสูงสุด 4.8 GHz แคช 25 MB จำนวน 12 คอร์ (8 Performance-cores และ 4 Efficient-cores) 20 เธรด, หน่วยความจำ 32 GB DDR4 3200 MHz (16 GB × 2 แบบ Dual Channel) การ์ดประมวลผลกราฟิก NVIDIA GeForce RTX 3060 Ti ขนาด 8 GB GDDR6, และหน่วยจัดเก็บข้อมูลแบบ WD Black NVMe SSD ซึ่งเอื้อต่อการประมวลผลไฟล์ข้อมูลจำนวนมาก การรวมผลข้อมูลรายเดือน และการฝึกสอนแบบจำลองที่มีการทดลองซ้ำหลายรอบ โดยเฉพาะในกรณีของแบบจำลองเชิงลึกที่สามารถใช้ GPU เพื่อเร่งการคำนวณได้ ทั้งนี้การระบุสภาพแวดล้อมการทดลองอย่างชัดเจนช่วยให้การทดลองสามารถทำซ้ำภายใต้เงื่อนไขเดียวกันได้ และเพิ่มความน่าเชื่อถือของผลการเปรียบเทียบแบบจำลองในเชิงระเบียบวิธี

3.3.3 การฝึกฝนโมเดล

ในการพัฒนาแบบจำลองพยากรณ์ ผู้วิจัยมุ่งเน้นไปที่เทคนิคการเรียนรู้ของเครื่อง (Machine Learning) และการเรียนรู้เชิงลึก (Deep Learning) ที่ได้รับการยอมรับว่ามีประสิทธิภาพสำหรับงานอนุกรมเวลา โดยโมเดลที่เลือกมาทดลองประกอบไปด้วย Random Forest, XGBoost, Recurrent Neural Network (RNN), Long Short-Term (LSTM), LSTM + Attention และ Gated Recurrent Unit (GRU) ด้วยภาษาโปรแกรมคอมพิวเตอร์ Python

ก่อนการฝึกฝนโมเดล ข้อมูลทั้ง Training Set และ Testing Set จะทำการปรับขนาด (Scaling) เพื่อให้โมเดลเชิงลึกสามารถเรียนรู้ได้อย่างมีประสิทธิภาพ โดยใช้เทคนิค Min-Max Scaling ซึ่งจะทำให้ค่าของตัวแปรถูกแปลงให้อยู่ในช่วง [0, 1] ในชุดคำสั่งของ Python เพื่อลดระยะเวลาในการฝึกฝนโมเดลด้วยเทคนิคที่ได้กล่าวมา

$$X_{new} = \frac{X_i - X_{min}}{X_{max} - X_{min}}$$

X_i คือ ค่าปัจจุบันที่ถูกแปลง

X_{max} คือ ค่าสูงสุดในชุดข้อมูล

X_{min} คือ ค่าต่ำสุดในชุดข้อมูล

และจะมีการใช้เทคนิค Log Transformation เนื่องจากข้อมูลปริมาณรับเข้ามีความผันผวนสูง (High Volatility) ผู้วิจัยจึงใช้เทคนิคดังกล่าว เพื่อลดความแปรปรวน (Variance Stabilization) และลดอิทธิพลของค่าผิดปกติ (Outliers) ซึ่งจะช่วยให้แบบจำลองเรียนรู้ทิศทางหลักของข้อมูลได้ดียิ่งขึ้น ซึ่งกระบวนการฝึกฝนและปรับแต่งพารามิเตอร์ (Model Training and Hyperparameter Tuning) ผู้วิจัยได้แบ่งชุดข้อมูลตามลำดับเวลา (Chronological Splitting) โดยจะทำการแบ่งชุดข้อมูลออกเป็น 3 ส่วน ได้ Training (ปี 2563-2565), Validation (ปี 2566), Testing (ปี 2567) และยังได้ใช้เทคนิค Early Stopping มาใช้ระหว่างการฝึกฝนโมเดล เพื่อหยุดการเรียนรู้ทันทีเมื่อค่าความสูญเสียในชุดของ Validation ไม่มีการปรับปรุงที่ดีขึ้น

3.4 การประเมินผล

ผู้วิจัยจะนำโมเดลที่ได้จากการฝึกฝนด้วยเทคนิค Random Forest, XGBoost, Recurrent Neural Network (RNN), Long Short-Term (LSTM), LSTM with Attention และ Gated Recurrent Unit (GRU) มาทดสอบชุดที่ 2 เพื่อเปรียบเทียบประสิทธิภาพในการพยากรณ์ โดยการประเมินผลจะทำ

การหาค่าความผิดพลาดสัมบูรณ์ (MAE) ค่าความผิดพลาดกำลังสองเฉลี่ย (MSE) ความผิดพลาดกำลังสองเฉลี่ย (RMSE) และค่าเฉลี่ยของเปอร์เซ็นต์ความผิดพลาดสัมบูรณ์ (MAPE) ในแต่ละเทคนิคด้วยข้อมูลระหว่าง 2563-2567 เพื่อหาความแม่นยำเชิงปริมาณและเชิงเปอร์เซ็นต์ของการพยากรณ์ โดยจะพิจารณาว่าโมเดลที่มีค่า MAPE ต่ำที่สุดจะถือว่ามีความแม่นยำสูงที่สุด ขณะเดียวกันผู้วิจัยยังได้ปรับแต่งค่า Hyperparameter Tuning ของแต่ละโมเดลด้วยเทคนิค Grid Search เพื่อให้ได้ค่าที่เหมาะสมที่สุดสำหรับชุดข้อมูลนี้ ผลการทดสอบทั้งหมดจะถูกนำมาเปรียบเทียบในเชิงคุณภาพและปริมาณ เพื่อระบุว่าเทคนิคใดสามารถให้ผลลัพธ์การพยากรณ์ที่แม่นยำและมีความเหมาะสมต่อการนำไปประยุกต์ใช้กับสินค้าคงคลังมากที่สุด แม้ว่า MAE, MSE และ RMSE จะสะท้อนความคลาดเคลื่อนเชิงปริมาณได้ดี แต่งานวิจัยนี้ให้ความสำคัญกับ MAPE เป็นหลัก เนื่องจากสามารถตีความผลลัพธ์ในเชิงธุรกิจได้ง่าย โดยผู้บริหารสามารถเข้าใจระดับความคลาดเคลื่อนของการพยากรณ์ในรูปเปอร์เซ็นต์ และนำไปใช้กำหนดเกณฑ์การตัดสินใจด้านการวางแผนพื้นที่และทรัพยากรได้โดยตรง

3.5 เกณฑ์การประเมินผลการพยากรณ์ (Evaluation Criteria and KPI)

เนื่องจากปริมาตรสินค้าขาเข้ามีความผันผวนสูง และมีความเบี่ยงเบนจากค่าเฉลี่ยในระดับที่มาก งานวิจัยนี้จึงกำหนดเกณฑ์การประเมิน (Key Performance Indicators: KPI) เพื่อใช้ในการตีความถึงความเหมาะสมของโมเดลสำหรับการนำไปใช้งานเพื่อเปรียบเทียบประสิทธิภาพของโมเดล โดยมุ่งเน้นที่ความสามารถในการสนับสนุนการวางแผนมากกว่าการวัดความแม่นยำเชิงตัวเลขเพียงอย่างเดียวโดยผู้วิจัยกำหนด ซึ่งเกณฑ์การประเมินดังกล่าวได้รับการกำหนดขึ้นจากมุมมองของผู้บริหาร โดยรายละเอียดของ KPI ดังตารางที่ 3.2

ตารางที่ 3.2 การกำหนดค่า KPI

ค่า MAPE (%)	ระดับประสิทธิภาพ	การนำไปใช้งาน
< 20	ดีเยี่ยม	วางแผนทรัพยากรได้อย่างดีเยี่ยม
20 - < 30	ดี/ใช้งานได้จริง	ใช้กำหนดกำลังคน
30 - < 40	พอใช้	ใช้ดูแนวโน้มและเตือนช่วงงานสูง
≥ 40	ต้องปรับปรุง	ไม่เหมาะใช้เชิงปฏิบัติการ

จากการวิเคราะห์ค่าเบี่ยงเบนของปริมาตร CBM รายวันเทียบเท่ากับค่าเฉลี่ย 1,302.10 CBM ซึ่งจากการตรวจสอบพบว่าข้อมูลมีความผันผวนสูง โดยมีค่าความคลาดเคลื่อนอยู่ที่ -69% ถึง +96% สะท้อนให้เห็นว่าปริมาณงานในแต่ละวันไม่สม่ำเสมออย่างมีนัยสำคัญ ความผันผวนนี้ส่งผลโดยตรงต่อการวางแผนการจัดการกำลังคน, พื้นที่จัดเก็บและอุปกรณ์ที่ต้องใช้ภายในคลัง

อย่างไรก็ตามจากการปรึกษากับผู้เชี่ยวชาญและผู้บริหารหน้างาน พบว่าการประเมินความแม่นยำของการพยากรณ์ในบริบทคลังสินค้าสามารถพิจารณาความคลาดเคลื่อนน้อยกว่าหรือเท่ากับ 30% ได้ เนื่องจากไม่ส่งผลกระทบต่อดำเนินที่หน้างานจริง ด้วยเหตุนี้ทางผู้วิจัยจึงนำข้อมูลความผันผวนในอดีตมาประกอบการกำหนดเกณฑ์การประเมินผล เพื่อให้สอดคล้องกับสภาพการใช้งานจริงของคลังสินค้า

ในการประเมินประสิทธิภาพของโมเดลพยากรณ์ งานวิจัยนี้ใช้ค่า Mean Absolute Percentage Error (MAPE) เป็นตัวชี้วัดหลัก เนื่องจากเป็นตัวชี้วัดที่สามารถแปลความหมายในเชิงร้อยละได้โดยตรงและเหมาะสมสำหรับการเปรียบเทียบโมเดลหลายรูปแบบ งานวิจัยอ้างอิงแนวทางการตีความค่า MAPE ตาม Lewis [30] ซึ่งแบ่งระดับความแม่นยำออกเป็นช่วงต่างๆ โดยค่า MAPE ระหว่าง 21-50% จัดอยู่ในระดับที่สามารถยอมรับได้ (Reasonable Forecast)

อย่างไรก็ตาม เพื่อให้สอดคล้องกับบริบทการบริหารพื้นที่จัดเก็บในคลังสินค้า งานวิจัยนี้ได้กำหนดเกณฑ์ KPI เชิงปฏิบัติการเพิ่มเติม โดยกำหนดให้ค่า MAPE ต่ำกว่า 20% อยู่ในระดับดีเยี่ยม และค่า MAPE ระหว่าง 20-30% อยู่ในระดับที่สามารถนำไปใช้งานจริงได้ ทั้งนี้การกำหนดเกณฑ์ดังกล่าวอ้างอิงจากระดับความผันผวนของข้อมูลย้อนหลังและลักษณะการจัดเก็บสินค้าในระดับ Pallet และ Zone

3.6 สรุปผลการทดลอง

ผู้วิจัยได้ทำการแสดงค่าไฮเปอร์พารามิเตอร์ที่เหมาะสมของแต่ละเทคนิคได้แก่ Random Forest, XGBoost, RNN, LSTM, LSTM with Attention และ GRU โดยได้ทำการปรับค่าไฮเปอร์พารามิเตอร์ (Hyperparameter Tuning) ด้วยเทคนิค Grid Search เพื่อลดปัญหาการเกิด Overfitting/Underfitting สำหรับการเปรียบเทียบโมเดล

การประเมินผลลัพธ์จะใช้ตัวชี้วัด MAE, MSE, RMSE และ MAPE เพื่อสะท้อนถึงความแม่นยำทั้งในเชิงปริมาณและเชิงเปอร์เซ็นต์ โดยมีการพิจารณาค่า MAPE ต่ำที่สุด เป็นเกณฑ์หลักในการระบุว่าโมเดลใด ที่ความแม่นยำสูงที่สุด อย่างไรก็ตาม การเลือกโมเดลที่เหมาะสมไม่ได้พิจารณาเฉพาะค่าความผิดพลาดทางสถิติเท่านั้น แต่ยังรวมถึงความเหมาะสมเชิงธุรกิจเช่น ความสามารถในการจัดการกับลักษณะความต้องการที่ไม่สม่ำเสมอ (Intermittent Demand) ซึ่งเป็นปัจจัยหลักของบริษัทผู้จัดจำหน่ายสินค้าไอที

นอกจากนี้แต่ละโมเดลได้มีการวิเคราะห์ ความสำคัญของคุณลักษณะ (Feature Importance) เช่น ฤดูกาล, รอบการสั่งซื้อ, และปริมาตรการจัดเก็บ (CBM) เพื่อระบุว่าปัจจัยใดที่มีผลต่อการพยากรณ์มากที่สุด ผลลัพธ์จากการพยากรณ์จะช่วยในการสนับสนุนการตัดสินใจของผู้บริหารในการบริหารสินค้าคงคลัง เพื่อลดปัญหา Overstock ซึ่งจะช่วยในการเพิ่มประสิทธิภาพการให้บริการลูกค้า

บทที่ 4
ผลการดำเนินงาน การวิเคราะห์และสรุปผลต่างๆ

การศึกษาวิจัยการพยากรณ์ปริมาตรสินค้าคงคลังขาเข้ากลุ่มสินค้าไอที โดยใช้เทคนิคการเรียนรู้ของเครื่องโดยมีวัตถุประสงค์เพื่อสร้างโมเดลสำหรับการพยากรณ์ปริมาตรสินค้าคงคลังขาเข้า และทำการเปรียบเทียบประสิทธิภาพของโมเดลโดยใช้เทคนิคการเรียนรู้ของเครื่อง โดยผลลัพธ์จากการทดลองมีดังต่อไปนี้

4.1 การตั้งค่าการทดลอง (Experimental Setup)

ผู้วิจัยได้กำหนดข้อมูลที่ใช้ในการทดลอง โดยจะใช้ข้อมูลการรับเข้าสินค้า (Inbound) โดยใช้ตัวแปรเป้าหมาย (Target) เป็นปริมาตรรับเข้า (Dimension) โดยจะใช้หน่วยเป็น CBM และมีการรวมข้อมูลจากรายวันเป็นรายเดือน (Monthly Aggregation) เพื่อให้สอดคล้องกับ Horizon แบบ t+1, t+3, t+6 เดือน

4.1.1 การจัดเตรียมข้อมูลและกำหนด Feature

แปลง Date เป็นชนิดของวันที่ และรวม (Sum) ข้อมูลรายวันให้เป็นรายเดือน ทำ Log Transformation กับ CBM เพื่อช่วยลดความผันผวนของอนุกรมเวลา สร้าง Feature เชิงอนุกรมเวลา (Time-Series Features) และ Feature ปฏิทิน ได้แก่

(1) Lag : Lag_1, Lag_2, Lag_3

(2). Rolling Mean : Roll_3, Roll_6, Roll_12 (อิงข้อมูลในอดีต)

(3) Calendar: Month, Year

(4) Special-Day Flags แบบรวมรายเดือนเช่น Holiday (จำนวนวันหยุด), Weekend Flag (จำนวนวันเสาร์-อาทิตย์), FirstMonth (วันแรกที่ทำงานของเดือน)

4.1.2 Horizon และรูปแบบของชุดข้อมูลสำหรับโมเดล

มีการกำหนดการพยากรณ์แบบหลายระยะ (Multi-Horizon) ดังต่อไปนี้

(1) Target_t1 = t+1 เดือน

(2) Target_t3 = t+3 เดือน

(3) Target_t6 = t+6 เดือน

4.1.3 การแบ่งชุดข้อมูลและการตรวจสอบข้อมูล

งานวิจัยนี้ใช้แนวทางการแบ่งข้อมูลแบบ Training (ปี 2563-2565), Validation (ปี 2566), Testing (ปี 2567) สำหรับ Testing Set โดยเรียงลำดับเวลา เพื่อใช้ในการเปรียบเทียบแต่ละโมเดล

4.1.4 โมเดลที่ใช้ในการเปรียบเทียบ

ผู้วิจัยต้องการเปรียบเทียบโมเดลกลุ่ม Machine Learning และ Deep Learning หลายชนิดโดยจะใช้โมเดลดังต่อไปนี้

(1) Machine Learning

XGBoost

Random Forest

(2) Deep Learning

RNN

LSTM

LSTM with Attention

GRU

4.1.5 ตัวชี้วัดในการประเมินผลการทดลอง

ในการประเมินผลการทดลอง งานวิจัยนี้ใช้ตัวชี้วัดมาตรฐาน ได้แก่ MAE, MSE, RMSE และ MAPE เพื่อวัดความคลาดเคลื่อนของผลการพยากรณ์ โดยกำหนดให้ MAPE เป็นตัวชี้วัดหลักในการเปรียบเทียบความแม่นยำระหว่างแบบจำลอง เนื่องจากสามารถสะท้อนความคลาดเคลื่อนในรูปเปอร์เซ็นต์และเอื้อต่อการเปรียบเทียบข้ามโมเดลได้อย่างเหมาะสม ทั้งนี้ ผลการประเมินจะรายงานแยกตามระยะเวลาการพยากรณ์ล่วงหน้า (Forecast Horizon) จำนวน 3 ระยะ ได้แก่ t+1 เดือน, t+3 เดือน และ t+6 เดือน เพื่อพิจารณาความเหมาะสมของแบบจำลองในมุมมองระยะสั้น ระยะกลาง และระยะยาวอย่างเป็นระบบ

4.2 การปรับแต่งพารามิเตอร์ (Hyperparameter Tuning) โดยใช้เทคนิค Grid Search

ผู้วิจัยได้ทำการกำหนดค่าพารามิเตอร์ที่น่าจะทำให้เกิดโมเดลที่มีประสิทธิภาพสำหรับการพยากรณ์สินค้าคงคลังขาเข้า โดยการกำหนดค่าและปรับแต่งไฮเปอร์พารามิเตอร์โดยใช้เทคนิค Grid Search แต่ละ Horizon จะปรับแต่งไฮเปอร์พารามิเตอร์ดังนี้

ตารางที่ 4.1 การปรับแต่งไฮเปอร์พารามิเตอร์โดยใช้เทคนิค Grid Search t+1

โมเดล	พารามิเตอร์	ช่วงค่าที่ทดสอบ	ค่าที่เลือกใช้
RNN	Hidden Units	{16, 32, 64}	32
	Dropout	{0.1, 0.2, 0.3}	0.3
	Sequence Length	{12}	12
LSTM	Hidden Units	{16, 32, 64}	32
	Dropout	{0.1, 0.2, 0.3}	0.1
	Sequence Length	{12}	12
LSTM with Attention	Hidden Units	{16, 32, 64}	32
	Dropout	{0.1, 0.2, 0.3}	0.3
	Attention Units	{16, 32}	16
	Sequence Length	{12}	12
GRU	Hidden Units	{16, 32, 64}	16
	Dropout	{0.1, 0.2, 0.3}	0.2
	Sequence Length	{12}	12
Random Forest	n_estimators	{100, 200, 300}	100
	max_depth	{3, 5, 7}	7
	min_samples_split	{2, 4}	2
XGBoost	n_estimators	{50, 100, 200, 300}	200
	max_depth	{2, 3, 5}	2
	subsample	{0.8, 1.0}	0.8

ตารางที่ 4.2 การปรับแต่งไฮเปอร์พารามิเตอร์โดยใช้เทคนิค Grid Search t+3

โมเดล	พารามิเตอร์	ช่วงค่าที่ทดสอบ	ค่าที่เลือกใช้
RNN	Hidden Units	{16, 32, 64}	64
	Dropout	{0.1, 0.2, 0.3}	0.3
	Sequence Length	{12}	12
LSTM	Hidden Units	{16, 32, 64}	32
	Dropout	{0.1, 0.2, 0.3}	0.2
	Sequence Length	{12}	12

ตารางที่ 4.2 การปรับแต่งไฮเปอร์พารามิเตอร์โดยใช้เทคนิค Grid Search t+3 (ต่อ)

โมเดล	พารามิเตอร์	ช่วงค่าที่ทดสอบ	ค่าที่เลือกใช้
LSTM with Attention	Hidden Units	{16, 32, 64}	16
	Dropout	{0.1, 0.2, 0.3}	0.1
	Attention Units	{16, 32}	16
	Sequence Length	{12}	12
GRU	Hidden Units	{16, 32, 64}	64
	Dropout	{0.1, 0.2, 0.3}	0.3
	Sequence Length	{12}	12
Random Forest	n_estimators	{100, 200, 300}	100
	max_depth	{3, 5, 7}	3
	min_samples_split	{2, 4}	4
XGBoost	n_estimators	{50, 100, 200, 300}	200
	max_depth	{2, 3, 5}	2
	subsample	{0.8, 1.0}	1.0

ตารางที่ 4.3 การปรับแต่งไฮเปอร์พารามิเตอร์โดยใช้เทคนิค Grid Search t+6

โมเดล	พารามิเตอร์	ช่วงค่าที่ทดสอบ	ค่าที่เลือกใช้
RNN	Hidden Units	{16, 32, 64}	64
	Dropout	{0.1, 0.2, 0.3}	0.3
	Sequence Length	{12}	12
LSTM	Hidden Units	{16, 32, 64}	32
	Dropout	{0.1, 0.2, 0.3}	0.1
	Sequence Length	{12}	12
LSTM with Attention	Hidden Units	{16, 32, 64}	16
	Dropout	{0.1, 0.2, 0.3}	0.3
	Attention Units	{16, 32}	32
	Sequence Length	{12}	12

ตารางที่ 4.3 การปรับแต่งไฮเปอร์พารามิเตอร์โดยใช้เทคนิค Grid Search t+6 (ต่อ)

โมเดล	พารามิเตอร์	ช่วงค่าที่ทดสอบ	ค่าที่เลือกใช้
GRU	Hidden Units	{16, 32, 64}	64
	Dropout	{0.1, 0.2, 0.3}	0.1
	Sequence Length	{12}	12
Random Forest	n_estimators	{100, 200, 300}	300
	max_depth	{3, 5, 7}	5
	min_samples_split	{2, 4}	2
XGBoost	n_estimators	{50, 100, 200, 300}	100
	max_depth	{2, 3, 5}	5
	subsample	{0.8, 1.0}	0.8

ผลการปรับจูนพารามิเตอร์พบว่าโมเดลในแต่ละช่วงเวลาพยากรณ์ (t+1, t+3 และ t+6) ให้ค่าที่เหมาะสมแตกต่างกัน ซึ่งสะท้อนให้เห็นว่าปริมาณข้อมูลฝึกสอนและระดับความยากของการพยากรณ์มีผลโดยตรงต่อความซับซ้อนของแบบจำลอง เมื่อระยะเวลาพยากรณ์เพิ่มขึ้น จำนวนตัวอย่างที่ใช้ฝึกสอนลดลงจากข้อจำกัดของการสร้างลำดับข้อมูล (sequence) ส่งผลให้โมเดลที่มีโครงสร้างซับซ้อนเกินไปมีแนวโน้มเกิด Overfitting จึงจำเป็นต้องเลือกค่าพารามิเตอร์ให้เหมาะสมกับบริบทของข้อมูลในแต่ละกรณี ดังนั้น การที่พารามิเตอร์ที่ดีที่สุดแตกต่างกันในแต่ละ Horizon จึงไม่ใช่ความผิดปกติ แต่เป็นผลลัพธ์ที่สอดคล้องกับหลักการทางสถิติและการเรียนรู้ของเครื่อง ซึ่งแสดงให้เห็นถึงความเหมาะสมของกระบวนการ Grid Search และความถูกต้องเชิงระเบียบวิธีของงานวิจัยนี้

4.2.1 ผลการทดลอง (Experimental Results)

4.2.1.1 ผลสรุปการทดลองโดยจะใช้ผลสรุปของ MAPE (%) ของแต่ละโมเดลที่มี Target t1, t3, t6 โดยจะได้ผลประเมินตามตารางที่ 4.2 ผลการเปรียบเทียบ MAPE (%)

ตารางที่ 4.4 ผลการเปรียบเทียบ MAPE (%)

กลุ่มโมเดล	โมเดล	t+1 (%)	t+3 (%)	t+6 (%)	ค่าเฉลี่ย (%)
Machine Learning	Random Forest	24.72	**24.07**	**29.26**	**26.02**
	XGBoost	**23.31**	29.66	36.51	29.83

ตารางที่ 4.4 ผลการเปรียบเทียบ MAPE (%) (ต่อ)

กลุ่มโมเดล	โมเดล	t+1 (%)	t+3 (%)	t+6 (%)	ค่าเฉลี่ย (%)
Deep Learning	LSTM	31.18	34.54	33.72	33.15
	LSTM with Attention	27.98	28.66	41.44	32.69
	GRU	48.7	27.54	41.49	39.24
	RNN	51.18	41.71	45.55	46.15

สรุปโมเดลที่ได้ผลลัพธ์ดีที่สุดในแต่ละ t+1 เดือน

t+1 เดือน XGBoost (MAPE = 23.31%)

t+3 เดือน Random Forest (MAPE = 24.07%)

t+6 เดือน Random Forest (MAPE = 29.26%)

เฉลี่ยทุกช่วงเวลา Random Forest ให้ผลลัพธ์ที่ดีที่สุด (MAPE = 26.02%)

4.2.1.2 ผลเชิงลึก (MAE/MSE/RMSE) ได้แสดงรายละเอียดข้อมูลค่า MAE, MSE, RMSE จากผลการสรุปได้ดังนี้

ตารางที่ 4.5 ผลการประเมินโมเดลเชิงลึก

โมเดล	Horizon	MAE	MSE	RMSE
LSTM	t+1	462.37	294,719.45	542.88
	t+3	522.06	356,979.31	597.48
	t+6	569.34	420,153.82	648.19
LSTM with Attention	t+1	418.61	269,205.83	518.86
	t+3	431.92	271,133.07	520.75
	t+6	701.82	584,612.55	764.62
GRU	t+1	736.52	623,351.08	789.54
	t+3	401.88	249,823.14	499.82
	t+6	706.47	595,823.11	771.88
RNN	t+1	771.33	667,889.44	817.29
	t+3	633.11	449,984.67	670.73
	t+6	748.53	648,507.92	805.29

ตารางที่ 4.5 ผลการประเมินโมเดลเชิงลึก (ต่อ)

โมเดล	Horizon	MAE	MSE	RMSE
Random Forest	t+1	368.75	215,483.79	464.2
	t+3	387.21	237,511.06	487.35
	t+6	491.4	372,805.36	610.58
XGBoost	t+1	349.47	207,649.02	455.68
	t+3	463.44	303,880.29	551.25
	t+6	624.19	517,955.74	719.78

โดยการประเมินผลได้ทำการเลือกพารามิเตอร์ Grid Search ก่อนนำไปประเมินผลบนชุดทดสอบ สามารถสรุปได้ดังนี้

(1) ช่วงเวลาที่ t+1 โมเดล XGBoost ให้ผลลัพธ์ดีที่สุด โดยมีค่า MAPE = 23.31% และให้ค่า RMSE = 455.68 สะท้อนให้เห็นว่าโมเดลแบบ Tree-based สามารถจับรูปแบบข้อมูลระยะสั้นได้อย่างมีประสิทธิภาพ และให้ความแม่นยำสูงกว่าโมเดล Deep Learning ในชุดข้อมูลนี้

(2) ช่วงเวลาที่ t+3 โมเดล Random Forest ให้ผลลัพธ์ดีที่สุด โดยมีค่า MAPE = 24.07% สะท้อนให้เห็นว่าโมเดลมีความเสถียรในการจับความสัมพันธ์ของข้อมูลในระยะกลางได้ดี และสามารถควบคุมความคลาดเคลื่อนได้เหมาะสม

(3) ช่วงเวลาที่ t+6 โมเดล Random Forest ให้ผลลัพธ์ดีที่สุด โดยมีค่า MAPE = 29.26% แสดงให้เห็นว่าเมื่อระยะเวลาการพยากรณ์ยาวขึ้น โมเดล Tree-based มีแนวโน้มให้ผลลัพธ์ที่มีความเสถียรและความคลาดเคลื่อนต่ำกว่าโมเดลอื่นในงานวิจัยนี้

(4) สำหรับค่าเฉลี่ยทุกช่วงเวลาพบว่าโมเดล Random Forest ให้ค่าเฉลี่ย MAPE ดีที่สุดอยู่ที่ 26.02% แสดงให้เห็นว่าโมเดลมีความเสถียรโดยรวมดีที่สุดเมื่อเปรียบเทียบในทุกช่วงเวลา

4.2.2 ผลการพยากรณ์เปรียบเทียบ Actual และ Predictive

ตารางที่ 4.6 ผลการพยากรณ์ t+1 เดือนมกราคมปี 2567

โมเดล	Actual (CBM)	Predictive (CBM)	Absolute Error (%)
XGBoost	1,726.37	1,721.39	0.29
Random Forest	1,726.37	1,273.97	26.21

ตารางที่ 4.6 ผลการพยากรณ์ t+1 เดือนมกราคมปี 2567 (ต่อ)

โมเดล	Actual (CBM)	Predictive (CBM)	Absolute Error (%)
LSTM with Attention	1,726.37	1,031.75	40.24
LSTM	1,726.37	1,002.62	41.92
GRU	1,726.37	1,371.87	20.53
RNN	1,726.37	813.71	52.87

จากผลการพยากรณ์ระยะ t+1 เดือน ในเดือนมกราคม พ.ศ. 2567 ซึ่งมีค่า Actual เท่ากับ 1,726.37 CBM พบว่าโมเดล XGBoost ให้ค่าความคลาดเคลื่อนต่ำที่สุดเมื่อเทียบกับโมเดลอื่น โดยมีค่า Absolute Error ต่ำมากเมื่อเปรียบเทียบกับโมเดลในกลุ่ม Deep Learning และ Random Forest ขณะที่โมเดล RNN และ LSTM มีค่าความคลาดเคลื่อนสูงกว่าอย่างชัดเจน สะท้อนให้เห็นว่าในระยะการพยากรณ์ระยะสั้น โมเดลประเภท ensemble-based สามารถเรียนรู้รูปแบบของข้อมูลได้อย่างมีประสิทธิภาพและให้ค่าการพยากรณ์ที่ใกล้เคียงกับค่าจริงมากกว่า

ตารางที่ 4.7 ผลการพยากรณ์ t+3 เดือนมีนาคมปี 2567

โมเดล	Actual (CBM)	Predictive (CBM)	Absolute Error (%)
XGBoost	1,490.31	898.53	39.71
Random Forest	1,490.31	1,194.76	19.83
LSTM with Attention	1,490.31	1,073.14	27.99
LSTM	1,490.31	897.04	39.81
GRU	1,490.31	1,237.98	16.93
RNN	1,490.31	704.04	52.76

จากผลการพยากรณ์ระยะ t+3 เดือน โดยพิจารณาตัวอย่างเดือนมีนาคม พ.ศ. 2567 ซึ่งมีค่า Actual เท่ากับ 1,490.31 CBM พบว่าโมเดล GRU ให้ค่าพยากรณ์ใกล้เคียงค่าจริงมากที่สุดในเดือนดังกล่าว โดยมีค่า Absolute Error ต่ำกว่าโมเดลอื่น ขณะที่โมเดล Random Forest ให้ค่าความคลาดเคลื่อนใกล้เคียงกันและมีความเสถียรในการพยากรณ์ในหลายช่วงเวลา เมื่อพิจารณาผลการประเมินโดยรวมจากค่า MAPE เฉลี่ยของระยะ t+3 พบว่า Random Forest มีค่าความคลาดเคลื่อนเฉลี่ยต่ำกว่า

GRU เล็กน้อย แสดงให้เห็นว่าแม้ GRU อาจให้ผลลัพธ์ที่ดีในบางช่วงเวลา แต่โมเดล Random Forest มีแนวโน้มให้ผลการพยากรณ์ที่มีความเสถียรมากกว่าในภาพรวม

ตารางที่ 4.8 ผลการพยากรณ์ t+6 เดือนมิถุนายนปี 2567

โมเดล	Actual (CBM)	Predictive (CBM)	Absolute Error (%)
XGBoost	1,838.69	856.35	53.43
Random Forest	1,838.69	1,090.62	40.69
LSTM with Attention	1,838.69	777.21	57.73
LSTM	1,838.69	1,014.64	44.82
GRU	1,838.69	771.75	58.03
RNN	1,838.69	502.95	72.65

จากผลการพยากรณ์ระยะ t+6 เดือน ในเดือนมิถุนายน พ.ศ. 2567 ซึ่งมีค่า Actual เท่ากับ 1,838.69 CBM พบว่าโมเดล Random Forest ให้ค่าความคลาดเคลื่อนต่ำที่สุดเมื่อเทียบกับโมเดลอื่น ขณะที่โมเดลในกลุ่ม Deep Learning และ XGBoost มีแนวโน้ม underestimate ค่าปริมาณจริงอย่างชัดเจน ส่งผลให้ค่า Absolute Error เพิ่มสูงขึ้น สะท้อนให้เห็นว่าเมื่อระยะเวลาการพยากรณ์ยาวขึ้น ความไม่แน่นอนของข้อมูลส่งผลต่อความแม่นยำของโมเดลมากขึ้น โดยเฉพาะโมเดลที่มีโครงสร้างซับซ้อนและต้องการปริมาณข้อมูลจำนวนมากในการเรียนรู้รูปแบบอนุกรมเวลา

4.3 อภิปรายผล (Discussion)

4.3.1 ระยะสั้น (t+1) พบว่าโมเดล XGBoost ให้ผลลัพธ์ดีที่สุด โดยมีค่า MAPE ต่ำที่สุดเมื่อเทียบกับโมเดลอื่น ผลลัพธ์ดังกล่าวสะท้อนให้เห็นว่าเมื่อ Horizon ยังสั้น ความสัมพันธ์ของข้อมูลย้อนหลังยังไม่ซับซ้อนมาก โมเดลกลุ่ม Tree-based สามารถจับรูปแบบของ Lag และ Rolling Features ได้อย่างมีประสิทธิภาพ ส่งผลให้สามารถลดค่าความคลาดเคลื่อนได้ดีกว่าโมเดล Deep Learning ในชุดข้อมูลนี้

4.3.2 ระยะกลาง (t+3) พบว่าโมเดล Random Forest ให้ผลลัพธ์ดีที่สุด โดยมีค่า MAPE ต่ำที่สุดในช่วงเวลาดังกล่าว แสดงให้เห็นว่าเมื่อระยะเวลาการพยากรณ์เพิ่มขึ้น โมเดลที่มีโครงสร้างไม่ซับซ้อนมากนักยังคงสามารถรักษาเสถียรภาพของการพยากรณ์ได้ดี ทั้งนี้แม้โมเดล Deep Learning บาง

รูปแบบจะสามารถเรียนรู้ความสัมพันธ์เชิงเวลาได้ แต่ในบริบทของข้อมูลชุดนี้ โมเดล Tree-based มีแนวโน้มควบคุมความคลาดเคลื่อนได้ดีกว่าในระยะกลาง

4.3.3 ระยะยาว (t+6) ผลการทดลองยังคงแสดงให้เห็นว่า Random Forest ให้ค่าความคลาดเคลื่อนต่ำที่สุดเมื่อเทียบกับโมเดลอื่น สะท้อนให้เห็นว่าเมื่อระยะเวลาการพยากรณ์ยาวขึ้น ความไม่แน่นอนของข้อมูลมีแนวโน้มเพิ่มสูงขึ้น อย่างไรก็ตาม โมเดล Tree-based ยังคงแสดงความเสถียรในการพยากรณ์ได้ดีกว่าโมเดล Deep Learning ในชุดข้อมูลนี้

4.3.4 เมื่อพิจารณาค่าเฉลี่ย MAPE ทั้ง 3 ช่วงเวลา พบว่าโมเดล Random Forest มีค่าเฉลี่ย MAPE ต่ำที่สุดที่ 26.02% แสดงให้เห็นว่าโมเดลมีความเสถียรโดยรวมดีที่สุดเมื่อเปรียบเทียบในทุก Horizon ทั้งนี้ไม่มีโมเดลใดให้ผลลัพธ์ดีที่สุดในทุกช่วงเวลาอย่างชัดเจน สะท้อนให้เห็นว่าความเหมาะสมของโมเดลขึ้นอยู่กับระยะเวลาการพยากรณ์และลักษณะความผันผวนของข้อมูล ดังนั้น หากพิจารณาในมุมมองเชิงธุรกิจที่ต้องการโมเดลที่ให้ผลลัพธ์สม่ำเสมอในหลาย Horizon โมเดล Random Forest อาจเป็นตัวเลือกที่เหมาะสมในบริบทของงานวิจัยนี้

4.4 การวิเคราะห์ตามกลุ่มของโมเดล (Model-Family Discussion)

4.4.1 Machine Learning กลุ่ม Machine Learning ได้แก่ Random Forest และ XGBoost ซึ่งอาศัยคุณลักษณะจากการทำ Feature Engineering เช่น Lag Features และ Rolling Statistics ในการเรียนรู้รูปแบบของข้อมูลเชิงอนุกรมเวลา จากผลการทดลองพบว่าโมเดลกลุ่มนี้สามารถให้ผลลัพธ์ในระดับที่ใช้งานได้จริงในทุกช่วงเวลา โดยเฉพาะ XGBoost ซึ่งให้ผลดีที่สุดในระยะสั้น (t+1) และ Random Forest ซึ่งให้ผลดีที่สุดในระยะกลางและระยะยาว (t+3 และ t+6) สะท้อนให้เห็นว่าแม้โมเดลกลุ่มนี้จะไม่ได้เรียนรู้โครงสร้างลำดับเวลาโดยตรง แต่การใช้ Lag และ Rolling Features สามารถถ่ายทอดข้อมูลเชิงเวลาให้โมเดลเรียนรู้ได้อย่างมีประสิทธิภาพ นอกจากนี้ ข้อได้เปรียบสำคัญของโมเดลกลุ่ม Machine Learning คือความเรียบง่าย ความเร็วในการฝึกโมเดล และความสามารถในการอธิบายผลลัพธ์ได้ง่าย ทำให้เหมาะสมต่อการนำไปประยุกต์ใช้ในบริบทเชิงธุรกิจที่ต้องการความเสถียรและความโปร่งใสของโมเดล

4.4.2 Deep Learning ได้แก่ RNN, LSTM, LSTM with Attention และ GRU ซึ่งมีความสามารถในการเรียนรู้ความสัมพันธ์เชิงลำดับเวลา (Sequential Dependency) โดยตรง จากผลการทดลองพบว่าโมเดลกลุ่มนี้สามารถให้ผลลัพธ์ในระดับที่ใช้งานได้ แต่ไม่ได้ให้ค่าความคลาดเคลื่อนต่ำที่สุดในทุก Horizon เมื่อเปรียบเทียบกับโมเดลกลุ่ม Machine Learning แม้ว่า LSTM with Attention จะมีโครงสร้างที่ช่วยให้โมเดลสามารถให้น้ำหนักกับช่วงเวลาที่สำคัญได้ แต่ในบริบทของชุดข้อมูลนี้ ความได้เปรียบดังกล่าวไม่ได้ส่งผลให้มีค่าความแม่นยำสูงกว่าโมเดล Tree-based อย่างชัดเจน อย่างไรก็ตาม โมเดลกลุ่ม Deep Learning ยังคงมีศักยภาพในการจัดการข้อมูลที่มีความซับซ้อนสูงหรือมีโครงสร้างระยะยาวที่ชัดเจน ซึ่งอาจแสดงผลลัพธ์ที่แตกต่างในบริบทของข้อมูลประเภทอื่น

4.5 การวิเคราะห์ความสำคัญของตัวแปร (Feature Importance)

ในงานวิจัยนี้ได้ดำเนินการวิเคราะห์ความสำคัญของตัวแปรเพื่อสนับสนุนการตีความผลลัพธ์ของโมเดล โดยผลการทดลองแสดงให้เห็นว่าโมเดลกลุ่ม Machine Learning โดยเฉพาะ Random Forest และ XGBoost ให้ประสิทธิภาพโดดเด่นในหลายช่วงเวลาการพยากรณ์ สะท้อนให้เห็นว่าการออกแบบคุณลักษณะ (Feature Engineering) โดยใช้ตัวแปร Lag และ Rolling Statistics มีบทบาทสำคัญต่อการพยากรณ์ปริมาตรสินค้าขาเข้า

ตัวแปรที่ใช้ในการทดลองประกอบด้วย Lag_1, Lag_2, Lag_3, Roll_3, Roll_6, Roll_12, Month, Year, Holiday, WeekendFlag และ FirstMonth ซึ่งสร้างขึ้นจากข้อมูลอนุกรมเวลา เพื่อสะท้อนทั้งความต่อเนื่องของปริมาณงานย้อนหลัง (Continuity), รูปแบบฤดูกาล (Seasonality) และพฤติกรรมการทำงานของคลังสินค้า

งานวิจัยนี้กำหนดความยาวของลำดับเวลาย้อนหลัง (Sequence Length) เท่ากับ 12 เดือน เพื่อให้โมเดลสามารถเรียนรู้รูปแบบเชิงฤดูกาลครบหนึ่งรอบปี ก่อนนำข้อมูลเข้าสู่โครงสร้างของโมเดลแบบลำดับเวลา ภายหลังการรวมข้อมูลรายวันเป็นรายเดือน และการจัดการค่าที่ขาดหาย ชุดข้อมูลที่นำมาใช้ในการฝึกและประเมินผลแบ่งตามลำดับเวลา โดยใช้ข้อมูลปี 2020–2022 สำหรับการฝึกโมเดล ปี 2023 สำหรับการตรวจสอบความถูกต้อง และปี 2024 สำหรับการทดสอบ ซึ่งสะท้อนสถานการณ์การใช้งานจริงในเชิงธุรกิจ

4.6 การเปรียบเทียบเวลาในการฝึกโมเดลและทรัพยากรที่ใช้ (Training Time & Resources)

ตารางที่ 4.9 การเปรียบเทียบระยะเวลาในการสร้างแบบจำลอง

กลุ่มโมเดล	โมเดล	t+1 (วินาที)	t+3 (วินาที)	t+6 (วินาที)	รวมทั้งหมด (วินาที)
Deep Learning	LSTM	201.71	215.31	231.89	648.91
Deep Learning	GRU	242.35	257.09	260.49	759.93
Deep Learning	RNN	232.79	243.9	259.74	736.43
Deep Learning	LSTM with Attention	368.34	421.47	455.57	1,245.38
Machine Learning	Random Forest	2.9	2.71	2.79	8.4
Machine Learning	XGBoost	2.23	2.73	2.43	7.39

จากตารางที่ 4.9 พบว่าโมเดลกลุ่ม Deep Learning ใช้เวลาในการฝึกสอนสูงกว่าโมเดลกลุ่ม Machine Learning อย่างชัดเจน โดยเฉพาะ LSTM with Attention ซึ่งใช้เวลาฝึกสอนรวมมากกว่า 1,200 วินาที ขณะที่ Random Forest และ XGBoost ใช้เวลาไม่ถึง 10 วินาทีเมื่อรวมทั้ง 3 Horizon

ความแตกต่างดังกล่าวสะท้อนให้เห็นถึงต้นทุนเชิงคำนวณ (Computational Cost) ที่เพิ่มขึ้นตามความซับซ้อนของโครงสร้างโมเดลและจำนวนพารามิเตอร์ ทั้งนี้เมื่อพิจารณาร่วมกับผลการพยากรณ์ พบว่าโมเดลกลุ่ม Machine Learning ไม่เพียงมีข้อได้เปรียบด้านเวลาในการฝึกสอน แต่ยังให้ค่าความแม่นยำได้ดีกว่าในหลายช่วงเวลา

บทที่ 5
บทสรุปและข้อเสนอแนะ

การพยากรณ์ปริมาตรสินค้าคงคลังขาเข้ากลุ่มสินค้าไอที โดยใช้เทคนิคการเรียนรู้ของเครื่อง โดยที่การวิจัยมีวัตถุประสงค์ตามบทที่ 1 สามารถสรุปหัวข้อได้ดังนี้

5.1 สรุปผลการดำเนินงานวิจัย

5.2 ปัญหาและอุปสรรคในการวิจัย

5.3 ข้อเสนอแนะจากการวิจัย

5.1 สรุปผลการดำเนินงานวิจัย

งานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาและเปรียบเทียบประสิทธิภาพของโมเดลพยากรณ์อนุกรมเวลา สำหรับคาดการณ์ปริมาตรสินค้าขาเข้า (Inbound) หน่วยลูกบาศก์เมตร (CBM) โดยใช้ข้อมูลรายวันตั้งแต่ปี พ.ศ. 2563–2567 และทำการรวมข้อมูลเป็นรายเดือนเพื่อพยากรณ์ล่วงหน้าในช่วงเวลา t+1, t+3 และ t+6 เดือน การแบ่งชุดข้อมูลใช้วิธี Time-Ordered Hold-out Split แบบรายปี โดยกำหนดให้ปี 2563–2565 เป็นชุดฝึกสอน ปี 2566 เป็นชุดตรวจสอบความถูกต้อง และปี 2567 เป็นชุดทดสอบ เพื่อสะท้อนสถานการณ์ใช้งานจริงในเชิงธุรกิจ

ภายหลังการปรับแต่งพารามิเตอร์ด้วยวิธี Grid Search บนชุด Validation แล้ว ได้นำโมเดลที่ได้ค่าดีที่สุดในแต่ละ Horizon ไปประเมินผลบนชุด Test ปี 2567 โดยทำการเปรียบเทียบค่าพยากรณ์ (Predictive) กับค่าจริง (Actual) ของแต่ละเดือน และคำนวณตัวชี้วัด MAE, MSE, RMSE และ MAPE โดยกำหนดให้ MAPE เป็นตัวชี้วัดหลักในการประเมินประสิทธิภาพ

ผลการทดลองพบว่า ไม่มีโมเดลใดให้ผลลัพธ์ดีที่สุดในทุกช่วงเวลา โดยในระยะสั้น (t+1 เดือน) โมเดล XGBoost ให้ค่า MAPE ต่ำที่สุดที่ 23.31% ขณะที่ระยะกลาง (t+3 เดือน) และระยะยาว (t+6 เดือน) โมเดล Random Forest ให้ค่า MAPE ต่ำที่สุดที่ 24.07% และ 29.26% ตามลำดับ ทั้งนี้เมื่อพิจารณาค่าเฉลี่ย MAPE ทั้งสามช่วงเวลา พบว่าโมเดล Random Forest มีค่าเฉลี่ยต่ำที่สุดที่ 26.02% ซึ่งอยู่ในช่วง 20-30% ตามเกณฑ์ KPI ที่กำหนดว่าอยู่ในระดับที่สามารถนำไปประยุกต์ใช้ในเชิงปฏิบัติการได้

เมื่อพิจารณาตารางเปรียบเทียบ Actual และ Predictive ของปี 2567 พบว่าโมเดลสามารถสะท้อนแนวโน้มการเปลี่ยนแปลงของปริมาตรสินค้าได้สอดคล้องกับค่าจริง แม้บางเดือนจะมีความคลาดเคลื่อนสูงในช่วงที่ปริมาตรสินค้าเกิดความผันผวนผิดปกติ แต่โดยภาพรวมโมเดลยังคงรักษาความแม่นยำให้อยู่ในระดับที่ยอมรับได้ในบริบทของการบริหารพื้นที่คลังสินค้า ทั้งนี้ระดับความคลาดเคลื่อนในช่วง 20-

30% ไม่ได้ส่งผลกระทบเชิงปฏิบัติการอย่างมีนัยสำคัญ เนื่องจากการจัดเก็บสินค้าในคลังมีความยืดหยุ่นในระดับ Pallet และ Storage Zone

ผลการวิเคราะห์ดังกล่าวแสดงให้เห็นว่า โมเดลกลุ่ม Machine Learning ซึ่งอาศัย Lag Features และ Rolling Statistics สามารถให้ผลลัพธ์ที่มีความเสถียรในหลาย Horizon ขณะที่โมเดลกลุ่ม Deep Learning มีศักยภาพในการเรียนรู้ความสัมพันธ์เชิงลำดับเวลาโดยตรง แต่ในบริบทของข้อมูลชุดนี้ ยังไม่แสดงความได้เปรียบเหนือกว่าโมเดล Tree-based อย่างชัดเจน ซึ่งอาจเป็นผลมาจากข้อจำกัดด้านขนาดข้อมูลและจำนวนตัวอย่างที่ใช้ในการฝึกโมเดล

นอกจากนี้ยังพบว่าความแม่นยำของโมเดลมีแนวโน้มลดลงเมื่อ Forecast Horizon ยาวขึ้น ซึ่งสอดคล้องกับหลักการของงานพยากรณ์อนุกรมเวลา เนื่องจากความไม่แน่นอนของข้อมูลมีแนวโน้มสะสมเพิ่มขึ้นตามระยะเวลา อย่างไรก็ตาม ภายใต้ข้อมูลชุดนี้ โมเดลที่ให้ผลดีที่สุดในแต่ละ Horizon ยังคงสามารถควบคุมระดับความคลาดเคลื่อนไม่ให้เกินกรอบ KPI ที่กำหนดไว้

โดยสรุป งานวิจัยนี้แสดงให้เห็นว่า การออกแบบ Feature Engineering ที่สะท้อนความต่อเนื่องของปริมาณงาน (Lag Features และ Rolling Statistics) รวมถึงปัจจัยด้านฤดูกาล (Calendar Features) มีบทบาทสำคัญต่อความแม่นยำของการพยากรณ์ และผลลัพธ์ที่ได้สามารถนำไปสนับสนุนการตัดสินใจด้านการวางแผนพื้นที่จัดเก็บสินค้าในคลังได้อย่างเหมาะสม

5.2 ปัญหาและอุปสรรคในการวิจัย

5.2.1 ข้อมูลที่ใช้ในการทดลองเป็นข้อมูลรายเดือนภายหลังการรวมข้อมูลรายวัน (Aggregation) ซึ่งทำให้จำนวนตัวอย่างในเชิงลำดับเวลาอยู่ในระดับจำกัด เมื่อกำหนดความยาวลำดับเวลาย้อนหลัง (Sequence Length) เท่ากับ 12 เดือน เพื่อสะท้อนรูปแบบฤดูกาลครบหนึ่งรอบปี จำนวนตัวอย่างที่สามารถนำไปใช้ฝึกโมเดลจึงลดลงตามโครงสร้างของข้อมูลลำดับเวลา ข้อจำกัดดังกล่าวส่งผลให้โมเดลที่มีความซับซ้อนสูง โดยเฉพาะโมเดล Deep Learning มีความเสี่ยงต่อการเกิด Overfitting มากขึ้น จึงจำเป็นต้องใช้เทคนิค Early Stopping และการควบคุมพารามิเตอร์อย่างเหมาะสมเพื่อรักษาเสถียรภาพของการเรียนรู้

5.2.2 ข้อจำกัดด้านโครงสร้างข้อมูล ข้อมูลที่ใช้เป็นข้อมูลรวมเชิงปริมาตร (CBM) ซึ่งไม่สามารถแยกผลกระทบเชิงลึกในระดับสินค้า หรือกลุ่มสินค้า ทำให้โมเดลไม่สามารถเรียนรู้ปัจจัยเชิงโครงสร้างบางประการที่อาจมีผลต่อปริมาตรสินค้าเข้า

5.2.3 ความผันผวนของข้อมูลจากปัจจัยภายนอก เช่น ช่วง peak ทางธุรกิจ วันหยุดยาว หรือเหตุการณ์พิเศษ ซึ่งบางช่วงไม่สามารถอธิบายได้ด้วยฟีเจอร์ที่มีอยู่ ส่งผลให้ค่าความคลาดเคลื่อนของการพยากรณ์เพิ่มสูงขึ้นในบางช่วงเวลา นอกจากนี้ การใช้ทรัพยากรคอมพิวเตอร์แบบเดียวกันในการฝึก

ทุกโมเดล แม้จะช่วยให้การเปรียบเทียบเป็นธรรม แต่ก็เป็นข้อจำกัดต่อการปรับแต่งโมเดล Deep Learning ให้มีความลึกหรือจำนวนพารามิเตอร์มากขึ้น

5.3 ข้อเสนอแนะจากการดำเนินงาน

5.3.1 การเพิ่มข้อมูลและขยายช่วงเวลาเพื่อเสริมความเสถียรของแบบจำลอง เพื่อให้แบบจำลองสามารถเรียนรู้รูปแบบเชิงฤดูกาลและความเปลี่ยนแปลงระยะยาวได้ครบถ้วนมากขึ้น งานวิจัยในอนาคตควรขยายช่วงข้อมูลให้ยาวขึ้น หรือเพิ่มจำนวนรอบปีของข้อมูลรายเดือน โดยเฉพาะเพื่อรองรับการพยากรณ์หลายระยะเวลา (t+1, t+3, t+6) อย่างเป็นธรรม นอกจากนี้ อาจพิจารณาเพิ่มข้อมูลเหตุการณ์พิเศษที่มีผลต่อปริมาณ Inbound (เช่น ช่วงโปรโมชันหรือการเร่งส่งมอบจากผู้ผลิต) เพื่อให้โมเดลรับรู้สัญญาณที่กระทบปริมาณงานอย่างมีนัยสำคัญ และลดความเสี่ยงของความคลาดเคลื่อนในช่วงที่ข้อมูลผันผวนสูง

5.3.2 แม้งานวิจัยนี้ใช้ฟีเจอร์เชิงเวลาและค่าทางสถิติจากอดีตเพื่อสะท้อนความต่อเนื่องของปริมาณงานแล้ว แต่การพยากรณ์ในบริบทคลังสินค้ามักได้รับอิทธิพลจากปัจจัยภายนอกและปัจจัยเชิงปฏิบัติงานที่ไม่สามารถอธิบายได้ด้วยเวลาเพียงอย่างเดียว ดังนั้น งานวิจัยในอนาคตควรเพิ่มฟีเจอร์ที่สะท้อนบริบทเชิงธุรกิจ เช่น ตัวแปรที่เกี่ยวข้องกับคำสั่งซื้อหรือแผนการส่งมอบจากผู้ผลิต ตัวแปรเชิงกลุ่มสินค้า/แบรนด์ที่มีรูปแบบการไหลเข้าแตกต่างกัน รวมถึงตัวแปรเหตุการณ์ที่มีผลต่อปริมาณการนำเข้าในบางช่วง เช่น ช่วงแคมเปญการขายหรือช่วงการส่งมอบล็อตใหญ่ ทั้งนี้การเพิ่มฟีเจอร์ดังกล่าวจะช่วยให้แบบจำลองอธิบายความผันผวนของข้อมูลได้ครบถ้วนขึ้น ลดความคลาดเคลื่อนในช่วงที่เกิดการเปลี่ยนแปลงรูปแบบของอนุกรมเวลา และทำให้ผลพยากรณ์มีความสอดคล้องกับสภาพแวดล้อมการทำงานของคลังสินค้าจริงมากยิ่งขึ้น

5.3.3 ในด้านแบบจำลอง งานวิจัยในอนาคตสามารถขยายแนวทางการพัฒนาไปสู่การใช้แบบจำลองเพิ่มเติมหรือการผสานแบบจำลอง (Hybrid/Ensemble) เพื่อรองรับการพยากรณ์หลายระยะเวลาภายใต้ข้อเท็จจริงที่ว่าไม่มีแบบจำลองใดให้ผลดีที่สุดในทุก Horizon เสมอไป แนวทางการผสานโมเดลอาจช่วยรวมจุดเด่นของโมเดลที่ตอบโจทย์ระยะสั้นเข้ากับโมเดลที่มีความเสถียรในระยะยาว รวมถึงเพิ่มความสามารถในการจัดการความไม่เชิงเส้นและความผันผวนของข้อมูลในบางช่วงเวลา นอกจากนี้ งานวิจัยต่อยอดควรให้ความสำคัญกับการประเมินผลเชิงปฏิบัติการร่วมกับตัวชี้วัดมาตรฐาน โดยเชื่อมโยงความคลาดเคลื่อนของการพยากรณ์กับการตัดสินใจด้านพื้นที่และทรัพยากรของคลังสินค้า เพื่อให้ข้อสรุปของงานวิจัยสะท้อนทั้งมิติทางเทคนิคและความเหมาะสมในการนำไปใช้งานจริง

บรรณานุกรม

บรรณานุกรม

[1] T. E. Goltsos et al., “Inventory-forecasting : Mind the gap,” *European Journal of Operational Research,* vol. 299, no. 2, pp. 397-419, June 2022.

[2] A. M. Khedr, “Enhancing supply chain management with deep learning and machine learning techniques : A review,” *Journal of Open Innovation : Technology, Market, and Complexity,* vol. 10, no. 4, pp. 100,379, December 2024.

[3] M. Seyedan and F. Mafakheri, “Predictive big data analytics for supply chain demand forecasting: methods, applications, and research opportunities,” *Journal of Big Data,* vol. 7, no. 1, pp. 53, July 2020.

[4] L. Ye et al., “Forecasting seasonal demand for retail : A fourier time-varying grey model,” *International Journal of Forecasting,* vol. 40, no. 4, pp. 1,467-1,485, October 2024.

[5] X. Tian and H. Wang, “Forecasting intermittent demand for inventory management by retailers : A new approach,” *Journal of Retailing and Consumer Services,* vol. 62, no. 1, pp. 102,662, September 2021.

[6] Y. Guo et al., “Supply chain resilience : A review from the inventory management perspective,” *Fundamental Research,* vol. 5, no. 4, pp. 102-108, August 2024.

[7] N. Kourentzes, “Intermittent demand forecasts with neural networks,” *International Journal of Production Economics,* vol. 143, no. 1, pp. 198-206, May 2013.

[8] F. Lolli et al., “Single-hidden layer neural networks for forecasting intermittent demand,” *International Journal of Production Economics,* vol. 183, no. 1, pp. 116-128, January 2017.

[9] S. Smyl, “A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting,” *International Journal of Forecasting,* vol. 36, no. 1, pp. 75-85, January 2020.

[10] S. Makridakis et al., “The M4 competition : Results, findings, conclusion and way forward,” *International Journal of Forecasting,* vol. 34, no. 4, pp. 802-808, October 2018.

[11] X. Liu and W. Wang, “Deep time series forecasting models : A comprehensive survey,” *Mathematics,* vol. 12, no. 10, pp. 1,504, January 2024.

[12] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International Journal of Forecasting,* vol. 22, no. 4, pp. 679-688, October 2006.

[13] J. R. Quinlan, "Induction of decision trees," *Machine Learning,* vol. 1, no. 1, pp. 81-106, March 1986.

[14] L. Breiman, "Random forests," *Machine Learning,* vol. 45, no. 1, pp. 5-32, October 2001.

[15] T. Chen and C. Guestrin, "XGBoost : A scalable tree boosting system," *International Conference on Emerging Trends in Electrical, Electronic and Communications Engineering, ELECOM 2020,* York, USA, November 25-27, 2020, pp. 198-203.

[16] Y. LeCun and G. Hinton, "Deep learning," *Nature,* vol. 521, no. 7,553, pp. 436-444, May 2015.

[17] I. D. Mienye et al., "Recurrent neural networks : A comprehensive review of architectures, Variants and Applications," *Journal of Imaging,* vol. 15, no. 9, pp. 517, September 2024.

[18] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation,* vol. 9, no. 8, pp. 1,735-1,780, November 1997.

[19] K. Bandara et al., "LSTM-MSNet : Leveraging forecasts on sets of related time series with multiple seasonal patterns," *International Journal of Computer Science,* vol. 32, no. 4, pp. 1,586-1,599, April 2021.

[20] S. Chavhan et al., "Deep learning approaches for stock price prediction : A comparative study of LSTM, RNN, and GRU models," *International Conference on Advanced Algorithms and Control Engineering, ICAACE 2024,* Shanghai, China, June 1-3, 2024, pp. 1-6.

[21] Y. Qin et al., "A dual-stage attention-based recurrent neural network for time series prediction," *International Conference on Artificial Intelligence,* IJCAI 2017, Melbourne, Australia, August 19-25, 2017, pp. 2,623-2,629.

[22] Y. Teng et al., "Time series prediction based on LSTM and high-order fuzzy cognitive map with attention mechanism," *Neural Processing Letters,* vol. 56, no. 5, pp. 267-274, September 2024.

[23] Y. Liu et al., "A machine learning approach to inventory stockout prediction," *Journal of Digital Economy,* vol. 4, no. 1, pp. 144-155, January 2025.

[24] I. Rafiq et al., "A novel approach for forecasting and scheduling building load through real-time occupant count data," *Arabian Journal for Science and Engineering,* vol. 50, no. 10, pp. 7,375-7,388, May 2025.

[25] N. Sukolkit et al., "An open innovative inventory management based demand forecasting approach for the steel industry," *Journal of Open Innovation : Technology, Market, and Complexity,* vol. 10, no. 4, pp. 100,407, December 2024.

[26] X. Wen and W. Li, "Time series prediction based on LSTM-attention-LSTM model," *Journal of Selected Topics in Applied Earth Observations and Remote Sensing,* vol. 11, no. 1, pp. 48,322-48,331, May 2023.

[27] F. Haselbeck et al., "Machine learning outperforms classical forecasting on horticultural sales predictions," *Machine Learning with Applications,* vol. 7, no. 1, pp. 100,239, March 2022.

[28] I. Vallés-Pérez et al., "Approaching sales forecasting using recurrent neural networks and transformers," *Expert Systems with Applications,* vol. 201, no. 1, pp. 116,993, September 2022.

[29] H. Abbasimehr et al., "An optimized model using LSTM network for demand forecasting," *Computers & Industrial Engineering,* vol. 143, no. 1, pp. 106,435, May 2020.

[30] Scientific Research Publishing, *"Lewis, C.D. Industrial and Business Forecasting Methods,"* [Online]. Available : https://www.scirp.org/reference/ referencespapers?referenceid=1166521. [Accessed : April 25, 2026].