

การพัฒนาระบบจำแนกประเภทปัญหาจาก Log Files ด้วยเทคนิค Machine Learning เพื่อสนับสนุนการวิเคราะห์และการส่งต่อปัญหาในองค์กร Publisher เกม

นายพงศ์ภีระ ชัยพงศ์ภัสสร

การค้นคว้าอิสระนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรมหาบัณฑิต
สาขาวิชาระบบสารสนเทศเพื่อการจัดการ
บัณฑิตวิทยาลัย มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
ปีการศึกษา 2568
ลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

การพัฒนาระบบจำแนกประเภทปัญหาจาก Log Files ด้วยเทคนิค Machine Learning เพื่อสนับสนุนการวิเคราะห์และการส่งต่อปัญหาในองค์กร Publisher เกม

นายพงศ์ภีระ ชัยพงศ์ภัสสร



การค้นคว้าอิสระนี้เป็นส่วนหนึ่งของการศึกษาหลักสูตร
วิทยาศาสตรมหาบัณฑิต
สาขาวิชาระบบสารสนเทศเพื่อการจัดการ
บัณฑิตวิทยาลัย มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
ปีการศึกษา 2568
ลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ



ใบรับรองการค้นคว้าอิสระ

บัณฑิตวิทยาลัย มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

เรื่อง การพัฒนาระบบจำแนกประเภทปัญหาจาก Log Files ด้วยเทคนิค Machine Learning เพื่อสนับสนุนการวิเคราะห์และการส่งต่อปัญหาในองค์กร Publisher เกม

โดย นายพงศ์ภีระ ชัยพงศ์ภัสสร

ได้รับอนุมัติให้นับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาระบบสารสนเทศเพื่อการจัดการ

.....
(ผู้ช่วยศาสตราจารย์ ดร.สุนันฑา สดสี)
คณบดีคณะเทคโนโลยีสารสนเทศและนวัตกรรมดิจิทัล

คณะกรรมการสอบการค้นคว้าอิสระ

.....
(รองศาสตราจารย์ ดร.พณณา ตั้งวรรณวิทย์)
ประธาน

.....
(ผู้ช่วยศาสตราจารย์ ดร.ณัฐวี อุตกฤษฎ์)
อาจารย์ที่ปรึกษา

.....
(ผู้ช่วยศาสตราจารย์ ดร.ธนพล เจนสุทธิเวชกุล)
กรรมการ

ชื่อ : นายพงศ์ภีระ ชัยพงศ์ภัสสร
ชื่อการค้นคว้าอิสระ : การพัฒนาระบบจำแนกประเภทปัญหาจาก Log Files ด้วยเทคนิค Machine Learning เพื่อสนับสนุนการวิเคราะห์และการส่งต่อปัญหาในองค์กร Publisher เกม
สาขาวิชา : ระบบสารสนเทศเพื่อการจัดการ
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
อาจารย์ที่ปรึกษาการค้นคว้าอิสระหลัก : ผู้ช่วยศาสตราจารย์ ดร.ณัฐวี อุตกฤษฎ์
ปีการศึกษา : 2568

บทคัดย่อ

ปัจจุบันธุรกิจเกมออนไลน์มีการแข่งขันสูง ความสำเร็จของเกมมิได้ขึ้นอยู่เพียงคุณภาพของตัวเกม แต่ยังขึ้นอยู่กับเสถียรภาพของระบบสนับสนุนต่าง ๆ ปัญหาที่มักพบคือเมื่อระบบเกิดข้อผิดพลาด องค์กรมักรับทราบล่าช้าเนื่องจากต้องรอผู้เล่นแจ้ง ประกอบกับการวิเคราะห์หาสาเหตุต้องตรวจสอบ Log File จำนวนมากด้วยแรงงานมนุษย์ ส่งผลให้การแก้ไขปัญหาล่าช้าและกระทบต่อประสบการณ์ของผู้ใช้งาน

งานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาระบบช่วยจำแนกประเภทปัญหาจาก Log File โดยประยุกต์ใช้เทคนิค Machine Learning เพื่อให้ระบบสามารถวิเคราะห์และจำแนกประเภทปัญหาได้โดยอัตโนมัติ ช่วยให้ทีมงานที่เกี่ยวข้องสามารถเข้าดำเนินการแก้ไขได้อย่างรวดเร็วและตรงจุด

กระบวนการวิจัยเริ่มจากการรวบรวม Log File จากเว็บไซต์และระบบบัญชีผู้ใช้ มาฝึกสอนโมเดล Machine Learning และเปรียบเทียบประสิทธิภาพการจำแนกอย่างน้อยสามวิธี เพื่อคัดเลือกวิธีที่แม่นยำที่สุดนำไปพัฒนาระบบต้นแบบ ภายหลังผู้ใช้สามารถอัปโหลด Log File เข้าสู่ระบบเพื่อให้แสดงผลการจำแนกประเภทปัญหา พร้อมทั้งสรุปสถิติในรูปแบบกราฟเพื่อสนับสนุนการวิเคราะห์เชิงภาพได้อย่างรวดเร็ว

(มีจำนวนทั้งสิ้น 59 หน้า)

คำสำคัญ : เกมออนไลน์, การจำแนกประเภท, ระบบสนับสนุนการตัดสินใจ

_____ อาจารย์ที่ปรึกษาการค้นคว้าอิสระหลัก

Name : Mr. Pongpeera Chaipongpatsorn
Independent Study Title : Development of a Machine Learning-Based Log File Classification System for Problem Analysis and Escalation Support in a Game Publisher Organization
Major Field : Management Information Systems
King Mongkut's University of Technology North Bangkok
Independent Study Advisor : Assistant Professor Dr. Nattavee Utakrit
Academic Year : 2025

ABSTRACT

The online gaming industry faces continuous competition, where success depends on both game quality and supporting system stability. When errors occur, organizations often detect them late due to reliance on user complaints. Additionally, manual examination of massive log files to identify root causes delays problem resolution and negatively impacts user experience.

This study aims to develop an automated problem classification system using Machine Learning. By analyzing log files and automatically categorizing issues, the system enables responsible teams to resolve problems accurately and promptly.

The research involves collecting log files from websites and account systems to train Machine Learning models. After evaluating at least three classification techniques, the most accurate method is selected to build a prototype. Ultimately, the user-friendly system allows users to upload log files for immediate analysis, presenting classification results and graphical statistical summaries to support rapid visual analysis and decision-making.

(Total 59 Pages)

Keywords: Online gaming, Classification, Decision support system

_____ Advisor

กิตติกรรมประกาศ

การศึกษาวิจัยฉบับนี้สำเร็จลุล่วงได้ด้วยความกรุณาจาก ผู้ช่วยศาสตราจารย์ ดร. ณัฐวี อุตกฤษฏ์ อาจารย์ที่ปรึกษาการค้นคว้าอิสระ ที่ได้สละเวลาให้คำปรึกษาและคำแนะนำอันเป็นประโยชน์ยิ่ง ผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้

ขอบพระคุณคณาจารย์ภาควิชาการจัดการเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ ทุกท่านที่ได้ประสิทธิประสาทวิชาความรู้ตลอดหลักสูตร

ขอขอบพระคุณ ธานิน ภิรมย์หวาด ประธานเจ้าหน้าที่บริหาร และ บริษัท อิเลคทรอนิคส์ เอ็กซ์ตรีม ที่สนับสนุนข้อมูลในการทำวิจัย ตลอดจนสนับสนุนด้านทุนการศึกษาซึ่งเป็นปัจจัยสำคัญที่ทำให้การศึกษาครั้งนี้สำเร็จลุล่วง รวมถึงเพื่อนร่วมงานทุกท่านที่คอยช่วยเหลือและให้กำลังใจ

ขอกราบขอบพระคุณ คุณพ่อและคุณแม่ ผู้เป็นกำลังใจสำคัญที่สุดที่คอยสนับสนุนและอยู่เคียงข้างผู้วิจัยเสมอมา สุดท้ายนี้ ข้าพเจ้าขออำนาจคุณพระศรีรัตนตรัยและสิ่งศักดิ์สิทธิ์ทั้งหลาย โปรดดลบันดาลให้ทุกท่านที่มีนามข้างต้น จงประสบแต่ความสุข ความสำเร็จ และมีสุขภาพร่างกายที่แข็งแรงตลอดไป

พงศ์ภีระ ชัยพงศ์ภัสสร

สารบัญ

		หน้า
บทคัดย่อภาษาไทย		ก
บทคัดย่อภาษาอังกฤษ		ข
กิตติกรรมประกาศ		ค
สารบัญ		ง
สารบัญตาราง (ถ้ามี)		ฉ
สารบัญรูปภาพ (ถ้ามี)		ช
บทที่ 1 บทนำ		1
1.1	ความเป็นมาและความสำคัญของปัญหา	1
1.2	วัตถุประสงค์ของการวิจัย	1
1.3	สมมติฐานของการวิจัย	2
1.4	นิยามศัพท์เฉพาะ	2
1.5	ขอบเขตของการวิจัย	3
1.6	ประโยชน์ที่คาดว่าจะได้รับ	4
บทที่ 2 เอกสารและงานวิจัยที่เกี่ยวข้อง		6
2.1	แนวคิดและทฤษฎีพื้นฐาน Theoretical Framework	6
2.2	อัลกอริทึมสำหรับการจำแนกประเภทปัญหา	9
2.3	การประเมินประสิทธิภาพและผลกระทบทางธุรกิจ	12
2.4	งานวิจัยที่เกี่ยวข้อง	15
บทที่ 3 วิธีดำเนินการวิจัย		20
3.1	กรอบแนวคิดของงานวิจัย	20
3.2	การเก็บรวบรวมและเตรียมข้อมูล	25
3.3	การสร้างและเปรียบเทียบประสิทธิภาพอัลกอริทึม	27
3.4	การพัฒนาระบบและการแสดงผล	29

3.5 การประเมินประสิทธิภาพและความพึงพอใจ 30

สารบัญ (ต่อ)

หน้า

3.6 สภาพแวดล้อมและเครื่องมือที่ใช้พัฒนาระบบ 31

บทที่ 4 ผลการวิจัย 33

4.1 ผลการเตรียมข้อมูลและการสกัดคุณลักษณะ 33

4.2 ผลการเปรียบเทียบประสิทธิภาพของอัลกอริทึม 34

4.3 ผลการพัฒนาระบบแสดงผล 36

4.4 ผลการประเมินความเที่ยงตรงโดยผู้เชี่ยวชาญ 40

4.5 ผลการประเมินความพึงพอใจของผู้ใช้งาน 41

บทที่ 5 สรุปผล อภิปรายผล และข้อเสนอแนะ 43

5.1 สรุปผล 43

5.2 อภิปรายผล 44

5.3 ข้อจำกัดของการวิจัย 45

5.4 ข้อเสนอแนะ 46

บรรณานุกรม 47

ภาคผนวก ก: แบบฟอร์มการประเมินโดยผู้เชี่ยวชาญ (IOC) 50

ภาคผนวก ข: แบบประเมินความพึงพอใจของผู้ใช้งานระบบ 54

ประวัติผู้เขียน 56

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

สารบัญตาราง

ตารางที่		หน้า
2-1	สรุปงานวิจัยที่เกี่ยวข้อง	17
4-1	การกระจายตัวของหมวดหมู่เหตุการณ์จากข้อมูล Log ช่วง Peak Hour	36
4-2	เปรียบเทียบค่าความถูกต้อง (Accuracy) ของแต่ละอัลกอริทึม	38
4-3	ตาราง Confusion Matrix แสดงผลการทำนายประเภทปัญหาของอัลกอริทึม Random Forest จากชุดข้อมูลทดสอบ	39
4-4	ผลการประเมินดัชนีความสอดคล้อง (IOC) จากผู้เชี่ยวชาญ	44
4-5	ผลการประเมินความพึงพอใจของผู้ใช้งาน	45

พระจอมเกล้าพระนครเหนือ มหาวิทยาลัยเทคโนโลยี

สารบัญรูปภาพ

ภาพที่		หน้า
3-1	แบบจำลอง Input-Process-Output (IPO)	22
4-1	ภาพหน้าจอ (Screenshot) ส่วนการอัปโหลดไฟล์ของระบบ	41
4-2	ภาพหน้าจอ Dashboard แสดงผลการวิเคราะห์สัดส่วนปัญหา (Pie Chart)	42
4-3	ภาพหน้าจอ Dashboard แสดงแนวโน้มปัญหาตามช่วงเวลา (Timeline Graph)	43



บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในอุตสาหกรรมเกมดิจิทัลประเภท Publisher ซึ่งมีการแข่งขันสูงและขับเคลื่อนด้วยรูปแบบธุรกิจที่ให้บริการแบบเรียลไทม์ ความเสถียร ความพร้อมใช้งาน และความต่อเนื่องของระบบถือเป็นปัจจัยเชิงกลยุทธ์ที่ส่งผลโดยตรงต่อประสบการณ์ของผู้เล่น ความเชื่อมั่นต่อแบรนด์ และรายได้ขององค์กร โดยเฉพาะระบบที่มีบทบาทสำคัญต่อวงจรชีวิตผู้เล่น ได้แก่ ระบบบัญชีผู้เล่น (Account System) และระบบการชำระเงิน (Payment System) ซึ่งเป็นโครงสร้างพื้นฐานหลักของการยืนยันตัวตน การจัดการสิทธิ์การเข้าใช้งาน ตลอดจนการทำธุรกรรมทางการเงินภายในเกม หากเกิดความผิดพลาดในระบบดังกล่าว ไม่เพียงส่งผลกระทบต่อผู้เล่นจำนวนมากในทันที แต่ยังอาจสร้างความเสียหายเชิงรายได้และภาพลักษณ์ขององค์กรในวงกว้าง

แม้ปัจจุบันองค์กรจะมีระบบเฝ้าระวังและเครื่องมือแจ้งเตือนในระดับหนึ่ง แต่ปัญหาสำคัญที่ยังคงปรากฏคือความล่าช้าในการตรวจจับความผิดปกติ โดยในหลายกรณีองค์กรจะรับรู้ปัญหาเมื่อผู้เล่นได้รับผลกระทบแล้วและแจ้งเรื่องผ่านช่องทางบริการลูกค้า สถานการณ์ดังกล่าวสะท้อนให้เห็นถึงช่องว่างระหว่างข้อมูลเชิงเทคนิคที่ถูกบันทึกไว้ใน Log Files กับกระบวนการวิเคราะห์เพื่อการตัดสินใจ เนื่องจากกระบวนการวิเคราะห์หาสาเหตุจำเป็นต้องตรวจสอบไฟล์บันทึกเหตุการณ์ (Log Files) จำนวนมากด้วยแรงงานมนุษย์ ซึ่งเป็นกระบวนการที่ใช้เวลานานและมีความเสี่ยงต่อความผิดพลาดสูง นำไปสู่ความล่าช้าในการแก้ไขปัญหา (Feldman & Sanger, 2007) อีกทั้งในสถาปัตยกรรมแบบ Microservices ซึ่งมีมากกว่า 60 บริการ การเฝ้าระวังทั้งหมดพร้อมกันด้วยแรงงานคนจึงเป็นภาระงานที่ซับซ้อน สิ้นเปลืองทรัพยากร และมีข้อจำกัดด้านความรวดเร็วในการตอบสนอง

ด้วยเหตุนี้ งานวิจัยฉบับนี้จึงมุ่งพัฒนาระบบจำแนกประเภทปัญหาอัตโนมัติจาก Log Files โดยประยุกต์ใช้เทคนิค Machine Learning เพื่อยกระดับกระบวนการจัดการเหตุการณ์ผิดปกติจากรูปแบบเชิงรับ (Reactive) ไปสู่แนวทางที่ขับเคลื่อนด้วยข้อมูลอย่างเป็นระบบ โดยจำกัดขอบเขตการศึกษาเฉพาะระบบที่มีความสำคัญเชิงกลยุทธ์สูงสุด ได้แก่ ระบบบัญชีผู้เล่นและระบบการชำระเงิน ซึ่งจัดเป็นระบบวิกฤตที่หากเกิดปัญหาจะส่งผลกระทบในทันที การพัฒนาระบบดังกล่าวจะช่วยให้สามารถจำแนกประเภทความผิดพลาด ระบุสาเหตุเบื้องต้น และส่งต่อปัญหาไปยังทีมที่เกี่ยวข้องได้อย่างรวดเร็วและแม่นยำ อันจะนำไปสู่การลดระยะเวลาเฉลี่ยในการแก้ไขปัญหา (MTTR) เพิ่มประสิทธิภาพการดำเนินงาน และลดความเสี่ยงทางธุรกิจได้อย่างมีนัยสำคัญ ทั้งยังเป็นการบูรณาการองค์ความรู้ด้าน Machine Learning และการบริหารจัดการบริการเทคโนโลยีสารสนเทศเข้ากับบริบทของอุตสาหกรรมเกมดิจิทัลอย่างเป็นรูปธรรม

1.2 วัตถุประสงค์ของการวิจัย

การวิจัยครั้งนี้มีวัตถุประสงค์เพื่อพัฒนาและประเมินประสิทธิภาพของระบบจำแนกประเภทปัญหาจากข้อมูล Log Files ในบริบทขององค์กร Publisher เกม โดยมุ่งเน้นระบบที่มี

ความสำคัญเชิงกลยุทธ์ ได้แก่ ระบบบัญชีผู้เล่น (Account System) และระบบการชำระเงิน (Payment System) ทั้งนี้มีวัตถุประสงค์เฉพาะดังต่อไปนี้

1.2.1 เพื่อพัฒนาระบบจำแนกประเภทของปัญหาจาก Log Files ของระบบบัญชีผู้เล่นและระบบการชำระเงิน โดยใช้เทคนิค Machine Learning

1.2.2 เพื่อทดสอบและเปรียบเทียบประสิทธิภาพความแม่นยำของอัลกอริทึม Machine Learning ในการจำแนกประเภทปัญหา

1.2.3 เพื่อประเมินความพึงพอใจของผู้ใช้งานที่มีต่อระบบจำแนกประเภทปัญหาจาก Log Files

วัตถุประสงค์ดังกล่าวสะท้อนทั้งมิติด้านการพัฒนาเชิงเทคนิค การประเมินเชิงประสิทธิภาพ และการตรวจสอบเชิงประยุกต์ เพื่อให้ระบบที่พัฒนาขึ้นมีความสมบูรณ์ครอบคลุม และสามารถสร้างคุณค่าต่อการบริหารจัดการเหตุการณ์ผิดปกติในองค์กรได้อย่างเป็นรูปธรรม

1.3 สมมติฐานของการวิจัย

การวิจัยครั้งนี้กำหนดสมมติฐานเพื่อทดสอบประสิทธิภาพเชิงเทคนิคของระบบที่พัฒนาขึ้น และเพื่อประเมินการยอมรับในเชิงการใช้งานจริง โดยมีรายละเอียดดังต่อไปนี้

1.3.1 สมมุติฐานที่ 1 (H1) ระบบจำแนกประเภทปัญหาจาก Log Files ที่พัฒนาขึ้นโดยใช้เทคนิค Machine Learning จะมีประสิทธิภาพในการจำแนกประเภทปัญหาอยู่ในระดับที่ยอมรับได้ โดยมีค่าความแม่นยำ (Accuracy) ไม่น้อยกว่าร้อยละ 80 ของชุดข้อมูลทดสอบ ซึ่งสะท้อนถึงความสามารถของแบบจำลองในการนำไปประยุกต์ใช้ในบริบทการปฏิบัติงานจริงได้อย่างมีประสิทธิผล

1.3.2 สมมุติฐานที่ 2 (H2) ผู้ใช้งานระบบมีความพึงพอใจต่อระบบจำแนกประเภทปัญหาที่พัฒนาขึ้นอยู่ในระดับมาก โดยมีค่าเฉลี่ยคะแนนความพึงพอใจไม่น้อยกว่า 3.51 จากระดับคะแนนเต็ม 5.00 ซึ่งสะท้อนถึงการยอมรับในด้านความสะดวกในการใช้งาน ความถูกต้องของผลลัพธ์ และประโยชน์ต่อการปฏิบัติงาน

$$H_0 = \mu \leq 3.51$$

$$H_1 = \mu > 3.51$$

สมมติฐานดังกล่าวครอบคลุมทั้งมิติด้านประสิทธิภาพเชิงอัลกอริทึมและมิติด้านการประยุกต์ใช้ในเชิงองค์กร เพื่อให้การประเมินผลการวิจัยมีความสมบูรณ์ทั้งเชิงเทคนิคและเชิงปฏิบัติการ

1.4 นิยามศัพท์เฉพาะ

องค์กรผู้ให้บริการเกม (Game Service Provider): หมายถึง บริษัทที่ดำเนินธุรกิจให้บริการเกมออนไลน์ในระดับภูมิภาค ซึ่งมีการบริหารจัดการระบบโครงสร้างพื้นฐาน (Infrastructure) ระบบบัญชีผู้ใช้งาน (Account System) และระบบชำระเงิน (Payment System) ของตนเอง เพื่อรองรับผู้เล่นจำนวนมาก

ลักษณะข้อมูล (Data Characteristics): หมายถึง ข้อมูลบันทึกเหตุการณ์ (Log Files) ที่ถูกจัดเก็บในรูปแบบโครงสร้าง (Structured Data) จากเว็บเซิร์ฟเวอร์ Nginx ของระบบบัญชีและระบบชำระเงิน ประกอบด้วยฟิลด์ข้อมูลสำคัญ ได้แก่ หมายเลขไอพี (IP Address), ที่อยู่ทรัพยากร (URI), รหัสสถานะตอบกลับ (HTTP Status Code) และระยะเวลาการประมวลผล (Response Time) โดยมุ่งเน้นข้อมูลในช่วงเวลาที่มีภาระงานสูง (Peak Hour)

1.5 ขอบเขตของการวิจัย

งานวิจัยนี้มุ่งเน้นการพัฒนาระบบจำแนกประเภทปัญหาจาก Log Files เพื่อสนับสนุนการวิเคราะห์และการส่งต่อปัญหาในองค์กรผู้ให้บริการเกม (Game Publisher) โดยกำหนดขอบเขตการศึกษาในแต่ละมิติอย่างชัดเจน ดังนี้

1.5.1 ระบบที่ศึกษา การศึกษานี้ครอบคลุมเฉพาะข้อมูล Log Files จากระบบบัญชีผู้เล่น (Account System) และระบบการชำระเงิน (Payment System) ซึ่งเป็นระบบวิกฤตที่ส่งผลกระทบโดยตรงต่อประสบการณ์ผู้เล่นและรายได้ขององค์กร ข้อมูลทั้งหมดถูกดึงผ่านแพลตฟอร์ม Elasticsearch ซึ่งทำหน้าที่เป็นศูนย์กลางในการจัดเก็บ ค้นหา และเรียกดูข้อมูล Log ในลักษณะกึ่งโครงสร้าง (Semi-structured Data) เพื่อให้สามารถสกัดข้อมูลที่เกี่ยวข้องกับเหตุการณ์ผิดปกติได้อย่างมีประสิทธิภาพ

1.5.2 การเตรียมข้อมูล กระบวนการเตรียมข้อมูลครอบคลุมทั้ง Application Logs และ Access Logs ซึ่งสะท้อนเหตุการณ์การทำงานภายในระบบและการร้องขอจากผู้ใช้งานภายนอก โดยจะทำการสุ่มข้อมูลในช่วงเวลาการใช้งานสูงสุด (Peak Hour) ระหว่างเวลา 18:00–22:00 น. เนื่องจากเป็นช่วงที่มีปริมาณธุรกรรมและความหนาแน่นของเหตุการณ์ผิดปกติสูง การเลือกช่วงเวลาดังกล่าวช่วยให้ชุดข้อมูลมีความเข้มข้นของปัญหา (Problem Density) และสอดคล้องกับบริบทการใช้งานจริงมากที่สุด ทั้งนี้กระบวนการเตรียมข้อมูลจะรวมถึงขั้นตอนการทำความสะอาดข้อมูล (Data Cleaning) การตัดคำ (Tokenization) และการแปลงข้อความให้อยู่ในรูปแบบที่เหมาะสมต่อการวิเคราะห์ด้วย Machine Learning

1.5.3 การจัดการข้อมูลละเอียดอ่อน เพื่อให้สอดคล้องกับหลักความมั่นคงปลอดภัยของสารสนเทศและมาตรฐานการคุ้มครองข้อมูลส่วนบุคคล งานวิจัยนี้จะดำเนินการทำนิรนามข้อมูล (Anonymization) สำหรับข้อมูลที่สามารถระบุตัวบุคคลได้ เช่น Cookie Session, IP Address บางส่วน หรือรหัสผู้ใช้งาน โดยจะทำการ Masking หรือ Hashing ข้อมูลดังกล่าวก่อนนำเข้าสู่กระบวนการวิเคราะห์ ทั้งนี้จะใช้เฉพาะข้อมูลที่จำเป็นต่อการจำแนกรูปแบบปัญหาเชิงระบบเท่านั้น

1.5.4 การจำแนกประเภทปัญหา งานวิจัยนี้กำหนดหมวดหมู่ปัญหาที่ครอบคลุมพฤติกรรมหลักของระบบจำนวน 5 ประเภท ได้แก่

1.5.4.1 Success คือ การตอบสนองที่สำเร็จตามปกติ

1.5.4.2 Client_Error คือ ความผิดพลาดจากฝั่งผู้ใช้งาน เช่น การส่งคำขอไม่ถูกต้อง

1.5.4.3 Server_Error คือ ความผิดพลาดที่เกิดจากฝั่งเซิร์ฟเวอร์

1.5.4.4 Client_Timeout (Status 499) คือ การยกเลิกการเชื่อมต่อจากฝั่งผู้ใช้ก่อนเซิร์ฟเวอร์ตอบกลับ

1.5.4.5 Security_Block (Status 444) คือ การถูกบล็อกจากมาตรการด้านความปลอดภัยของระบบ หมวดหมู่ดังกล่าวถูกออกแบบให้สะท้อนสถานการณ์ที่เกิดขึ้นจริงในสภาพแวดล้อมการให้บริการเกมออนไลน์ และรองรับการส่งต่อปัญหาไปยังทีมที่เกี่ยวข้องได้อย่างเหมาะสม

1.5.5 เทคนิคที่ใช้ การพัฒนาระบบจะใช้ภาษา Python เป็นหลัก โดยประยุกต์ใช้ไลบรารี Scikit-learn สำหรับการสร้างและประเมินโมเดล Machine Learning และใช้ไลบรารี Streamlit
ในการพัฒนาส่วนต่อประสานกับผู้ใช้ในรูปแบบ Web Application เพื่อให้ทีมปฏิบัติการสามารถอัปโหลด Log หรือเชื่อมต่อข้อมูลเพื่อวิเคราะห์ได้อย่างสะดวก

1.5.6 ผลลัพธ์ของระบบ ผลลัพธ์ของงานวิจัยคือเว็บแอปพลิเคชันที่สามารถวิเคราะห์และจำแนกประเภท Log ได้แบบทันที (Real-time Prediction) พร้อมทั้งแสดงผลผ่านแดชบอร์ดเชิงภาพ (Visual Dashboard) เพื่อสรุปสถิติแนวโน้มของปัญหา เช่น สัดส่วนประเภทความผิดพลาด ความถี่ของเหตุการณ์ และการกระจายตัวตามช่วงเวลา ซึ่งช่วยสนับสนุนการตัดสินใจของทีมปฏิบัติการได้อย่างมีประสิทธิภาพ

1.5.7 ข้อจำกัดของงานวิจัย งานวิจัยนี้จำกัดขอบเขตอยู่ที่การจำแนกประเภทของปัญหาและการแสดงผลเชิงวิเคราะห์เท่านั้น โดยไม่ครอบคลุมถึงการดำเนินการแก้ไขปัญหาโดยอัตโนมัติ (Automated Remediation) หรือการควบคุมระบบโดยตรง ทั้งนี้เพื่อมุ่งเน้นการพัฒนาและประเมินประสิทธิภาพของกระบวนการจำแนกประเภทปัญหาเป็นหลักก่อนการต่อยอดสู่ระบบอัตโนมัติในอนาคต

1.6 ประโยชน์ที่คาดว่าจะได้รับ

จากการพัฒนาระบบจำแนกประเภทปัญหาจาก Log Files ด้วยเทคนิค Machine Learning งานวิจัยนี้คาดว่าจะก่อให้เกิดประโยชน์ทั้งในเชิงปฏิบัติการ เชิงการบริหารจัดการ และเชิงกลยุทธ์ขององค์กร Publisher เกม โดยผลลัพธ์ของระบบมิได้จำกัดเพียงการเพิ่มความแม่นยำในการวิเคราะห์ข้อมูลเท่านั้น หากยังช่วยยกระดับกระบวนการบริหารจัดการเหตุการณ์ผิดปกติ (Incident Management) ให้มีความเป็นระบบ รวดเร็ว และขับเคลื่อนด้วยข้อมูล (Data-Driven Operations) มากยิ่งขึ้น ทั้งนี้สามารถสรุปประโยชน์ที่คาดว่าจะได้รับดังต่อไปนี้

1.6.1 ลดระยะเวลาในการตรวจจับและวิเคราะห์ประเภทของปัญหาที่เกิดขึ้นในระบบสำคัญ โดยช่วยให้สามารถระบุลักษณะของเหตุการณ์ผิดปกติได้อย่างรวดเร็ว ลดภาระการวิเคราะห์ด้วยมือ และลดค่าเฉลี่ยระยะเวลาในการแก้ไขปัญหา (MTTR)

1.6.2 เพิ่มประสิทธิภาพในการส่งต่อปัญหา (Incident Escalation) ไปยังทีมพัฒนาที่รับผิดชอบได้อย่างถูกต้องและรวดเร็ว เนื่องจากระบบสามารถจำแนกหมวดหมู่ของปัญหาได้

ชัดเจน ส่งผลให้ลดความผิดพลาดในการประสานงานและลดความซ้ำซ้อนของกระบวนการตรวจสอบเบื้องต้น

1.6.3 ช่วยให้ผู้บริหารสามารถมองเห็นแนวโน้ม รูปแบบ และสถิติของปัญหาผ่านแดชบอร์ดเชิงภาพ ทำให้สามารถนำข้อมูลไปใช้ในการวางแผนทรัพยากร การจัดลำดับความสำคัญของงานพัฒนา และการบริหารความเสี่ยงด้านเทคโนโลยีในอนาคตได้อย่างมีประสิทธิภาพ

โดยสรุป งานวิจัยนี้คาดว่าจะสร้างคุณค่าเชิงระบบให้แก่องค์กร ทั้งในมิติของการเพิ่มประสิทธิภาพการดำเนินงาน ลดความสูญเสียจากความล่าช้า และเสริมสร้างความสามารถในการแข่งขันในอุตสาหกรรมเกมดิจิทัลที่มีพลวัตสูง.

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

บทที่ 2
เอกสารและงานวิจัยที่เกี่ยวข้อง

การวิจัยครั้งนี้มุ่งเน้นการบูรณาการเทคโนโลยีสารสนเทศเพื่อแก้ปัญหาเชิงปฏิบัติการในอุตสาหกรรมเกม ผู้วิจัยได้ดำเนินการทบทวนวรรณกรรมเพื่อสร้างกรอบแนวคิดเชิงทฤษฎี โดยครอบคลุมตั้งแต่พื้นฐานการเรียนรู้ของเครื่อง การประมวลผลภาษาธรรมชาติ ไปจนถึงเทคโนโลยีการจัดการข้อมูลขนาดใหญ่ ซึ่งเป็นองค์ประกอบสำคัญในการพัฒนาระบบจำแนกประเภทปัญหาจาก Log Files ได้มีการศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้อง เพื่อนำแนวคิดมาประยุกต์ใช้กับงานวิจัย ดังนี้

2.1 แนวคิดและทฤษฎีพื้นฐาน Theoretical Framework

2.2 อัลกอริทึมสำหรับการจำแนกประเภทปัญหา

2.3 การประเมินประสิทธิภาพและผลกระทบทางธุรกิจ

2.4 การสังเคราะห์งานวิจัยที่เกี่ยวข้อง

2.5 งานวิจัยที่เกี่ยวข้อง

2.1 แนวคิดและทฤษฎีพื้นฐาน Theoretical Framework

การพัฒนาระบบจำแนกประเภทปัญหาจาก Log Files ในบริบทขององค์กร Publisher เกม จำเป็นต้องอาศัยการบูรณาการองค์ความรู้จากหลายสาขา ทั้งด้านการเรียนรู้ของเครื่อง การประมวลผลภาษาธรรมชาติ การวิเคราะห์ข้อมูลเชิงเหตุการณ์ และเทคโนโลยีการจัดการข้อมูลสมัยใหม่ เนื่องจากข้อมูล Log มีลักษณะเป็นข้อความกึ่งโครงสร้าง ปริมาณมาก และเกิดขึ้นอย่างต่อเนื่องในสภาพแวดล้อมแบบ Distributed Systems การสร้างแบบจำลองที่มีประสิทธิภาพจึงต้องตั้งอยู่บนกรอบแนวคิดเชิงทฤษฎีที่ชัดเจนและสอดคล้องกับลักษณะของข้อมูล

ในหัวข้อนี้ ผู้วิจัยนำเสนอแนวคิดและทฤษฎีพื้นฐานที่ใช้เป็นรากฐานในการออกแบบกระบวนการวิเคราะห์ ตั้งแต่ขั้นตอนการเรียนรู้ของแบบจำลอง การแปลงข้อมูลข้อความให้อยู่ในรูปแบบเชิงตัวเลข ไปจนถึงเทคโนโลยีที่รองรับการจัดเก็บและแสดงผลข้อมูล ทั้งนี้เพื่อให้การพัฒนาระบบมีความถูกต้องเชิงวิชาการและสามารถประยุกต์ใช้ได้จริงในบริบทองค์กร

2.1.1 การเรียนรู้ของเครื่อง (Machine Learning)

การเรียนรู้ของเครื่อง (Machine Learning) เป็นแขนงหนึ่งของเทคโนโลยีปัญญาประดิษฐ์ (Artificial Intelligence: AI) ที่มุ่งเน้นการสร้างและพัฒนาแบบจำลองหรืออัลกอริทึมให้ระบบคอมพิวเตอร์สามารถเรียนรู้รูปแบบ (Patterns) และความสัมพันธ์จากข้อมูลได้ด้วยตนเอง รวมถึงสามารถปรับปรุงประสิทธิภาพของตนเองได้จากประสบการณ์ โดยไม่จำเป็นต้องอาศัยการเขียนโปรแกรมหรือระบุกฎเกณฑ์คำสั่งไว้ล่วงหน้าอย่างชัดเจนในทุกกรณี เทคนิคเหมืองข้อมูลและการเรียนรู้ของเครื่องสามารถประยุกต์ใช้กับข้อมูลขนาดใหญ่เพื่อสนับสนุนการตัดสินใจได้อย่างมีประสิทธิภาพ และถือเป็นพื้นฐานสำคัญของการวิเคราะห์ข้อมูลสมัยใหม่ (Han, Pei, & Kamber, 2011)

งานวิจัยนี้เลือกใช้การเรียนรู้แบบมีผู้สอน (Supervised Learning) ซึ่งเป็นกระบวนการที่แบบจำลองเรียนรู้ความสัมพันธ์ระหว่างชุดข้อมูลนำเข้า (Input Features) และคำตอบที่ถูกต้อง (Labels) ที่ผ่านการกำหนดหมวดหมู่ไว้ล่วงหน้า เช่น Success, Client_Error และ Server_Error กระบวนการดังกล่าวประกอบด้วยการแบ่งข้อมูลออกเป็นชุดฝึกสอน (Training Set) และชุดทดสอบ (Testing Set) เพื่อประเมินความสามารถของแบบจำลองในการทำนายข้อมูลที่ไม่เคยพบมาก่อน

แนวคิดสำคัญของ Supervised Learning คือการลดค่าความผิดพลาด (Loss Function) ระหว่างค่าที่แบบจำลองทำนายกับค่าจริง ผ่านกระบวนการปรับพารามิเตอร์ (Model Optimization) จนได้แบบจำลองที่สามารถแยกแยะรูปแบบความผิดปกติของระบบบัญชีผู้เล่นและระบบชำระเงินได้อย่างมีประสิทธิภาพ

2.1.2 การประมวลผลภาษาธรรมชาติ (Natural Language Processing: NLP)

เนื่องจาก Log Files มีลักษณะเป็นข้อความกึ่งโครงสร้าง (Semi-structured Text) ที่ประกอบด้วยข้อความ ตัวเลข และสัญลักษณ์ผสมกัน การเตรียมข้อมูล (Data Preprocessing) จึงเป็นขั้นตอนที่มีความสำคัญอย่างยิ่งต่อคุณภาพของแบบจำลอง

2.1.2.1 Tokenization และ Word Segmentation เป็นกระบวนการแบ่งข้อมูลดิบออกเป็นหน่วยย่อยที่มีความหมาย (Tokens) เช่น การแยกคำ แยกสัญลักษณ์ หรือแยกส่วนประกอบของบรรทัด Log สำหรับ Access Log ของ Nginx ขั้นตอนนี้รวมถึงการสกัดสถานะ HTTP (Status Code) เช่น 200, 404, 500 ตลอดจนพิกัดทรัพยากร (Uniform Resource Identifier: URI) ออกมาเป็นคุณลักษณะ (Features) ที่สามารถนำไปวิเคราะห์เชิงสถิติได้

2.1.2.2 TF-IDF Vectorization Term Frequency–Inverse Document Frequency (TF-IDF) เป็นเทคนิคในการแปลงข้อมูลข้อความให้อยู่ในรูปแบบเวกเตอร์เชิงตัวเลข โดยพิจารณาจากความถี่ของการปรากฏของคำในเอกสารหนึ่ง (Term Frequency: TF) ควบคู่กับความสำคัญของคำนั้นเมื่อเปรียบเทียบกับเอกสารทั้งหมดในชุดข้อมูล (Inverse Document Frequency: IDF) ปัจจุบัน TF-IDF เป็นเทคนิคยอดนิยมในการประเมินความสำคัญของคำในเอกสาร เพื่อแปลงข้อความเป็นเวกเตอร์ตัวเลขสำหรับการประมวลผลของคอมพิวเตอร์ (สถาบันส่งเสริมการวิเคราะห์และบริหารข้อมูลขนาดใหญ่ภาครัฐ, 2563)

หลักการทำงานที่สำคัญของเทคนิค TF-IDF คือการให้ค่าน้ำหนัก (Weight) แก่คำศัพท์ โดยคำที่มีความถี่สูงในเอกสารใดเอกสารหนึ่ง แต่ไม่ค่อยปรากฏในเอกสารอื่น ๆ ในระบบ จะได้รับค่าน้ำหนักที่สูงขึ้น ซึ่งบ่งชี้ว่าคำนั้นเป็นคำที่มีเอกลักษณ์และมีความสำคัญในการจำแนกเอกสาร ในทางตรงกันข้าม หากคำใดปรากฏบ่อยในทุก ๆ เอกสาร (เช่น คำเชื่อม หรือข้อความทั่วไปของระบบ) เทคนิคนี้จะทำการลดทอนน้ำหนักของคำนั้นลงโดยอัตโนมัติ เนื่องจากถือว่าเป็นคำที่ไม่มีนัยสำคัญในการแยกแยะความแตกต่างระหว่างกลุ่มข้อมูล

ในบริบทของงานวิจัยฉบับนี้ การใช้เทคนิค TF-IDF มีความเหมาะสมอย่างยิ่งในขั้นตอนการสกัดคุณลักษณะ (Feature Extraction) ของไฟล์บันทึกเหตุการณ์ (Log Files) ซึ่งมีลักษณะเป็นข้อความกึ่งโครงสร้าง (Semi-structured Text) เทคนิคนี้ช่วยลดน้ำหนักของข้อความปกติทั่วไปที่ระบบบันทึกไว้ซ้ำ ๆ และไปเพิ่มน้ำหนักให้กับคำหรือรูปแบบ (Patterns) ที่

สะท้อนลักษณะเฉพาะของข้อผิดพลาด เช่น "Timeout", "Connection Refused", หรือ รหัส "Error 500" กระบวนการนี้จะช่วยสร้างตัวแทนเอกสาร (Document Term Matrix) ที่มีประสิทธิภาพ ส่งผลให้แบบจำลองการเรียนรู้ของเครื่อง (Machine Learning Model) สามารถเรียนรู้คำสำคัญที่บ่งชี้ประเภทของเหตุการณ์ผิดปกติของระบบบัญชีผู้เล่นและระบบการชำระเงินได้อย่างแม่นยำยิ่งขึ้น

2.1.3 เล็อกไฟล์และการวิเคราะห์เชิงลึก (Log Analysis)

Log Files เปรียบเสมือนร่องรอยการทำงานของระบบในระดับเหตุการณ์จริง (Real-time Operational Footprints) ซึ่งสะท้อนทั้งพฤติกรรมของผู้ใช้งานและการตอบสนองของระบบ การวิเคราะห์ Log ในงานวิจัยนี้มุ่งเน้นสององค์ประกอบหลัก ได้แก่

2.1.3.1 Access Logs เป็นบันทึกการร้องขอทรัพยากรผ่านโปรโตคอล HTTP ซึ่งสามารถสะท้อนปริมาณการใช้งาน รูปแบบคำร้องขอ และระยะเวลาการตอบสนองของระบบ การวิเคราะห์ Access Logs ช่วยให้สามารถตรวจจับรูปแบบที่ผิดปกติ เช่น การร้องขอซ้ำจำนวนมาก หรือการเข้าถึงทรัพยากรที่ล้มเหลวอย่างต่อเนื่อง

2.1.3.2 Error Indicators การใช้รหัสสถานะ HTTP (Status Codes) เป็นตัวบ่งชี้สำคัญในการวิเคราะห์สาเหตุของปัญหา เช่น 499 (Client Closed Request) สะท้อนปัญหาด้านประสิทธิภาพหรือความล่าช้าในการตอบสนอง 444 (No Response) มักเกี่ยวข้องกับมาตรการด้านความปลอดภัยการวิเคราะห์ตัวชี้วัดเหล่านี้ช่วยสนับสนุนการระบุ Root Cause ทั้งในมิติ Performance และ Security

2.1.4 เทคโนโลยีการบริหารจัดการข้อมูลสมัยใหม่ (Data Tech Stack)

การพัฒนาระบบในงานวิจัยนี้อาศัยเทคโนโลยีที่รองรับการจัดการข้อมูลขนาดใหญ่และการแสดงผลแบบโต้ตอบ

2.1.4.1 Elasticsearch (The Search Engine) Elasticsearch เป็นระบบสืบค้นและจัดเก็บข้อมูลที่พัฒนาบนพื้นฐานของ Apache Lucene รองรับการทำดัชนี (Indexing) ข้อมูลขนาดใหญ่และการค้นหาแบบรวดเร็ว (Near Real-time Search) ทำให้สามารถดึงข้อมูลย้อนหลังในช่วงเวลา Peak Hour (18:00–22:00 น.) ได้อย่างมีประสิทธิภาพ

2.1.4.2 Streamlit (The Interface Framework) Streamlit เป็นเฟรมเวิร์กสำหรับพัฒนาเว็บแอปพลิเคชันด้วยภาษา Python ที่เน้นความเรียบง่ายและการทำงานแบบ Reactive ช่วยให้นักวิทยาศาสตร์ข้อมูลสามารถสร้างแดชบอร์ดแสดงผลการวิเคราะห์ได้อย่างรวดเร็ว รองรับการโต้ตอบกับผู้ใช้งานแบบ Real-time และสนับสนุนการนำแบบจำลองไปประยุกต์ใช้ในสภาพแวดล้อมการทำงานจริง

โดยสรุป แนวคิดและทฤษฎีพื้นฐานในหัวข้อนี้เป็นรากฐานสำคัญที่เชื่อมโยงองค์ความรู้ด้าน Machine Learning, NLP, Log Analysis และ Data Technology เข้าด้วยกันอย่างเป็นระบบ เพื่อรองรับการพัฒนาและประเมินประสิทธิภาพของระบบจำแนกประเภทปัญหาจาก Log Files ในบทถัดไปอย่างมีหลักวิชาการและความสมบูรณ์เชิงโครงสร้าง

2.2 อัลกอริทึมสำหรับการจำแนกประเภทปัญหา

การเลือกอัลกอริทึมสำหรับการจำแนกประเภทปัญหาจาก Log Files จำเป็นต้องพิจารณาทั้งในมิติทางทฤษฎีและเชิงปฏิบัติ เนื่องจากข้อมูล Log มีลักษณะเป็นข้อความกึ่งโครงสร้างที่ผ่านกระบวนการแปลงเป็นเวกเตอร์เชิงตัวเลข (เช่น TF-IDF) ซึ่งมักมีจำนวนคุณลักษณะ (High-dimensional Feature Space) สูง และอาจมีความไม่สมดุลของข้อมูล (Class Imbalance) ระหว่างหมวดหมู่ต่าง ๆ ดังนั้น ผู้วิจัยจึงเลือกเปรียบเทียบอัลกอริทึมที่มีพื้นฐานแนวคิดแตกต่างกัน 3 กลุ่ม ได้แก่ แบบจำลองเชิงความน่าจะเป็น (Probabilistic Model) แบบจำลองเชิงเรขาคณิต (Geometric Margin-based Model) และแบบจำลองเชิงกลุ่ม (Ensemble-based Model) เพื่อประเมินประสิทธิภาพและความเหมาะสมกับบริบทของข้อมูลจริง

2.2.1 Naïve Bayes

พัฒนาขึ้นโดยอาศัยพื้นฐานของทฤษฎีบทของเบย์ (Bayes' Theorem) โดยมีข้อสมมติฐานที่สำคัญคือ คุณลักษณะ (Features) แต่ละตัวมีความเป็นอิสระต่อกัน (Independence Assumption)

ซึ่งหมายถึงการปรากฏของคำศัพท์หนึ่งในข้อความจะไม่มีความเกี่ยวข้องหรือส่งผลต่อการปรากฏของคำศัพท์อื่น แม้ว่าในทางภาษาศาสตร์ (NLP) คำศัพท์มักจะมีความสัมพันธ์กัน แต่ Naïve Bayes ยังคงให้ประสิทธิภาพสูงในการจำแนกข้อความ เนื่องจากมีโครงสร้างที่ไม่ซับซ้อน ประมวลผลได้รวดเร็ว และใช้ทรัพยากรระบบต่ำ จึงนิยมใช้เป็นแบบจำลองพื้นฐาน (Baseline Model) สำหรับการคัดกรองข้อมูลจากสื่อสังคมออนไลน์ที่มีปริมาณมหาศาล โดยมีสมการพื้นฐานดังนี้

$$P(C|X) = \frac{(P(X|C) \cdot P(C))}{P(X)} \quad (2\text{-}1)$$

โดยที่ P(c|x) คือ ความน่าจะเป็นที่ข้อความ x จะอยู่ในคลาส c (Posterior Probability)

P(x|c) คือ ความน่าจะเป็นที่จะพบข้อความ x ในคลาส c (Likelihood)

P(c) คือ ความน่าจะเป็นของคลาส c (Class Prior Probability)

P(x) คือ ความน่าจะเป็นของข้อความ x (Predictor Prior Probability)

ในการประยุกต์กับงานข้อความ มักใช้ Multinomial Naïve Bayes ซึ่งเหมาะกับข้อมูลแบบ TF-IDF หรือ Term Frequency โดยแบบจำลองจะคำนวณค่า Posterior ของทุกคลาส และเลือกคลาสที่ให้ค่ามากที่สุด (Maximum A Posteriori: MAP)

แม้ว่าสมมติฐานความเป็นอิสระจะไม่สอดคล้องกับธรรมชาติของภาษาโดยสมบูรณ์ แต่ในทางปฏิบัติ Naïve Bayes มักให้ผลลัพธ์ที่ดีในงาน Text Classification เนื่องจากมีความเรียบง่าย คำนวณรวดเร็ว และใช้ทรัพยากรต่ำ จึงเหมาะสำหรับใช้เป็น Baseline Model ในการเปรียบเทียบประสิทธิภาพกับแบบจำลองที่ซับซ้อนกว่า

2.2.2 Support Vector Machine (SVM)

Support Vector Machine (SVM) เป็นอัลกอริทึมที่มีรากฐานจากทฤษฎีการเรียนรู้เชิงสถิติ (Statistical Learning Theory) มุ่งเน้นการสร้างระนาบแบ่ง (Hyperplane) ที่เหมาะสมที่สุดเพื่อแยกข้อมูลแต่ละคลาสออกจากกัน โดยใช้หลักการระยะขอบสูงสุด (Maximum Margin Principle) เพื่อให้ระยะห่างระหว่างระนาบแบ่งกับจุดข้อมูลที่ใกล้ที่สุดมีค่ามากที่สุด

ในงานวิจัยนี้เลือกใช้ Linear SVM (LinearSVC) เนื่องจากข้อมูลข้อความที่แปลงเป็นเวกเตอร์ด้วย TF-IDF มักมีจำนวนคุณลักษณะสูงกว่าจำนวนตัวอย่าง ทำให้ข้อมูลมีแนวโน้มที่จะแยกออกจากกันได้ด้วยระนาบเชิงเส้น (Linearly Separable) ในปริภูมิหลายมิติ อีกทั้งยังมีความรวดเร็วในการฝึกฝนและทำนายมากกว่าเคอร์เนลรูปแบบอื่น โดยมีสมการพื้นฐานของระนาบแบ่งดังนี้

$$w \cdot x + b = 0 \quad (2\text{-}2)$$

โดยที่ w คือ เวกเตอร์น้ำหนัก (Weight Vector)

x คือเ วกเตอร์คุณลักษณะ

b คือ ค่าคงที่ (Bias)

Support Vector Machine (SVM) จะเลือกเส้นแบ่งที่ทำให้ระยะห่างระหว่างระนาบกับจุดข้อมูลที่ใกล้ที่สุด (Support Vectors) มีค่ามากที่สุด ซึ่งช่วยเพิ่มความสามารถในการทั่วไป (Generalization Ability) กับข้อมูลใหม่

ในงานวิจัยนี้เลือกใช้ Linear SVM (LinearSVC) เนื่องจากข้อมูล TF-IDF มักมีมิติสูงมาก และในปริภูมิหลายมิติ ข้อมูลมักมีแนวโน้มแยกได้ด้วยเส้นตรง อีกทั้ง Linear SVM มีความเร็ว

ในการฝึกและทำนายสูงกว่า Support Vector Machine (SVM) แบบ Kernel อื่น ๆ และเหมาะสมกับข้อมูลขนาดใหญ่ของระบบ Log ในองค์กร

2.2.3 Random Forest (The Chosen Model)

Random Forest เป็นอัลกอริทึมแบบ Ensemble Learning ที่พัฒนาเพื่อแก้ข้อจำกัดของ Decision Tree ซึ่งมักเกิดปัญหา Overfitting เมื่อฝึกกับข้อมูลฝึกมากเกินไป แนวคิดหลักของ Random Forest คือการสร้างต้นไม้ตัดสินใจจำนวนมาก และรวมผลการตัดสินใจของต้นไม้เหล่านั้นเข้าด้วยกัน กลไกสำคัญประกอบด้วย

2.4.3.1 Bootstrap Aggregating หรือ Bagging เป็นกระบวนการสุ่มตัวอย่างข้อมูลจากชุดฝึกแบบมีการคืนตัวอย่าง (Sampling with Replacement) เพื่อสร้างชุดข้อมูลย่อยหลายชุด ทำให้ต้นไม้แต่ละต้นถูกฝึกด้วยข้อมูลที่แตกต่างกัน ช่วยลดความแปรปรวนของแบบจำลอง

2.4.3.2 Feature Randomness ในแต่ละโหนดของต้นไม้ จะสุ่มเลือกเพียงบางส่วนของคุณลักษณะเพื่อใช้พิจารณาการแบ่งข้อมูล แทนที่จะใช้คุณลักษณะทั้งหมด วิธีนี้ช่วยลดความสัมพันธ์ระหว่างต้นไม้ (De-correlation) และเพิ่มความหลากหลายของแบบจำลอง

2.4.3.3 Majority Voting เมื่อทำการทำนาย ต้นไม้แต่ละต้นจะให้ผลลัพธ์เป็นคลาส และผลลัพธ์สุดท้ายจะได้จากการลงคะแนนเสียงข้างมาก (Majority Voting)

กระบวนการตัดสินใจสามารถอธิบายด้วยสมการ

$$\hat{Y} = \text{mode}\{h_{1(x)}, h_{2(x)}, \dots, h_{n(x)}\} \quad (2\text{-}6)$$

โดยที่ $\hat{Y}$ คือ คลาสที่ถูกทำนาย และ

$h_{i(x)}$ คือ ผลการทำนายจากต้นไม้ต้นที่ i จากจำนวนต้นไม้ทั้งหมด

n คือ จำนวนต้นไม้ทั้งหมด

เหตุผลที่เลือก Random Forest เป็นแบบจำลองหลัก (The Chosen Model) เนื่องจากสามารถจัดการกับข้อมูลที่มีความซับซ้อน ไม่เชิงเส้น และมีปฏิสัมพันธ์ระหว่างคุณลักษณะได้ดี อีกทั้งยังมีความทนทานต่อ Noise และ Class Imbalance เมื่อมีการปรับค่า Hyperparameters อย่างเหมาะสม จึงให้ความแม่นยำและความเสถียรสูงกว่าสองแบบจำลองแรกในการทดลองกับข้อมูล Log จริงขององค์กร

2.3 การประเมินประสิทธิภาพและผลกระทบทางธุรกิจ

การประเมินผลของงานวิจัยนี้มิได้จำกัดอยู่เพียงตัวชี้วัดเชิงเทคนิคของแบบจำลอง (Model Performance Metrics) เช่น Accuracy, Precision, Recall และ F1-Score เท่านั้น แต่ยังครอบคลุมถึงผลกระทบเชิงปฏิบัติการและเชิงธุรกิจ (Operational & Business Impact) เพื่อสะท้อนคุณค่าที่แท้จริงของระบบในบริบทองค์กร Publisher เกม ซึ่งต้องให้ความสำคัญกับความต่อเนื่องของบริการ (Service Continuity) และประสบการณ์ผู้เล่น (Player Experience)

2.3.1 ตัวชี้วัดเชิงประสิทธิภาพของแบบจำลอง (Model Performance Metrics) การประเมินประสิทธิภาพของแบบจำลอง (Model Evaluation) เป็นขั้นตอนสำคัญเพื่อยืนยันว่าแบบจำลองที่พัฒนาขึ้นมีความแม่นยำและน่าเชื่อถือก่อนนำไปใช้งานจริง การวัดประสิทธิภาพของแบบจำลอง Classification นิยมใช้ตารางเมทริกซ์สับสน (Confusion Matrix) เพื่อคำนวณหา

ค่าความถูกต้อง ความแม่นยำ ค่าระลึก และค่าความถ่วงดุล (Soontranon, 2023) ตาราง Confusion Matrix จะทำการจัดแบ่งผลการทำนายออกเป็น 4 ประเภทพื้นฐาน ได้แก่:

- True Positive (TP): จำนวนรายการที่แบบจำลองทำนายว่าอยู่ในคลาสที่พิจารณา (เช่น เป็นปัญหาข้อผิดพลาด) และความเป็นจริงข้อมูลนั้นก็เป็นคลาสนั้นอย่างถูกต้อง

- True Negative (TN): จำนวนรายการที่แบบจำลองทำนายว่าไม่อยู่ในคลาสที่พิจารณา (เช่น เป็นสถานะปกติ) และความเป็นจริงข้อมูลนั้นก็ถูกต้องตามที่ทาย

- False Positive (FP): จำนวนรายการที่แบบจำลองทำนายว่าอยู่ในคลาสที่พิจารณา แต่ความเป็นจริงข้อมูลนั้นไม่ใช่ (ทำนายผิดประเภท หรือผลบวกเทียม)

- False Negative (FN): จำนวนรายการที่แบบจำลองทำนายว่าไม่อยู่ในคลาสที่พิจารณา แต่ความเป็นจริงข้อมูลนั้นกลับอยู่ในคลาสที่พิจารณา (ทำนายผิดพลาด หรือผลลบเทียม)

จากค่าในตาราง Confusion Matrix สามารถนำมาคำนวณตัวชี้วัดประสิทธิภาพของแบบจำลองเพื่อการประเมินผลเชิงลึก (Powers, 2011) ได้ดังนี้:

2.3.1.1 ค่าความถูกต้อง (Accuracy) คือสัดส่วนของจำนวนตัวอย่างที่แบบจำลองทำนายได้ถูกต้องทั้งหมด (ทั้งการทายถูกว่าใช่และทายถูกว่าไม่ใช่) เมื่อเทียบกับจำนวนข้อมูลทั้งหมดในชุดทดสอบ ตัวชี้วัดนี้สะท้อนความแม่นยำโดยรวมของแบบจำลอง อย่างไรก็ตาม ในบริบทของไฟล์บันทึกเหตุการณ์ (Log Files) ที่ชุดข้อมูลมักมีความไม่สมดุล (Class Imbalance) สูง ค่าความถูกต้องเพียงอย่างเดียวอาจไม่เพียงพอต่อการสะท้อนประสิทธิภาพที่แท้จริง

2.3.1.2 Precision คือความถูกต้องของการทำนายในแต่ละคลาส คำนวณจากสัดส่วนของจำนวนตัวอย่างที่ทำนายถูกต้องในคลาสนั้น เทียบกับจำนวนตัวอย่างทั้งหมดที่โมเดลทายว่าเป็นคลาสดังกล่าว ตัวชี้วัดนี้มีความสำคัญอย่างยิ่งในสถานการณ์ที่ต้องการลดการเกิดผลบวกเทียม (False Positive) เพื่อป้องกันการส่งแจ้งเตือนเหตุการณ์ผิดปกติไปยังทีมปฏิบัติการทั้งที่ระบบยังคงทำงานเป็นปกติ

2.3.1.3 Recall (Sensitivity) คือความสามารถของแบบจำลองในการตรวจจับเหตุการณ์ที่เกิดขึ้นจริง คำนวณจากสัดส่วนของจำนวนตัวอย่างที่ทำนายถูกต้องในคลาสนั้นเทียบกับจำนวนตัวอย่างจริงทั้งหมดของคลาสดังกล่าว ค่าระลึกมีความสำคัญอย่างยิ่งเมื่อองค์กรต้องการลดผลลบเทียม (False Negative) เนื่องจากหากระบบพลาดการตรวจจับเหตุการณ์วิกฤตที่แฝงอยู่ใน Log Files ย่อมส่งผลกระทบรุนแรงต่อธุรกิจ

2.3.1.4 F1-Score คือ คือค่าเฉลี่ยเชิงฮาร์มอนิก (Harmonic Mean) ระหว่างค่าความแม่นยำ (Precision) และค่าระลึก (Recall) ตัวชี้วัดนี้ถูกนำมาใช้เพื่อสะท้อนสมดุลของแบบจำลองในการแยกแยะข้อมูล ซึ่งมีความเหมาะสมอย่างยิ่งสำหรับชุดข้อมูลที่ไม่มีความสมดุลระหว่างหมวดหมู่ เนื่องจากสามารถแสดงประสิทธิภาพการทำงานที่แท้จริงของแบบจำลองได้ครอบคลุมกว่าการพิจารณาเพียงค่าความถูกต้องโดยรวม

2.3.2 MTTR (Mean Time To Recovery) นอกจากตัวชี้วัดเชิงอัลกอริทึมแล้ว งานวิจัยนี้ยังประเมินผลในเชิงปฏิบัติการผ่านตัวชี้วัด **MTTR (Mean Time To Recovery)** ซึ่งเป็นค่าเฉลี่ยระยะเวลาที่ใช้ตั้งแต่เกิดเหตุการณ์ผิดปกติจนถึงระบบสามารถกลับมาทำงานได้ตามปกติ ในบริบทขององค์กรเกมออนไลน์ MTTR ที่ลดลงหมายถึง ลด Downtime ของระบบเติมเงินหรือระบบล็อกอิน ลดผลกระทบต่อรายได้จากการหยุดชะงักของบริการ ลดความไม่พึงพอใจของผู้เล่น

ระบบจำแนกประเภทปัญหาอัตโนมัติช่วยให้ Incident ถูกส่งต่อ (Escalate) ไปยังทีมที่เกี่ยวข้องได้รวดเร็วและถูกต้องตั้งแต่ต้นทาง จึงช่วยลดเวลาการวิเคราะห์ Root Cause และลด MTTR โดยรวมขององค์กร ซึ่งสะท้อนผลลัพธ์เชิงธุรกิจได้อย่างชัดเจน

2.3.3 IOC (Index of Item-Objective Congruence) เพื่อรับรองความเที่ยงตรงเชิงเนื้อหา (Content Validity) ของระบบ ผู้วิจัยใช้ดัชนีความสอดคล้องระหว่างรายการกับวัตถุประสงค์ หรือ **IOC (Index of Item-Objective Congruence)** โดยอาศัยการประเมินจากผู้เชี่ยวชาญจำนวน 3 ท่าน ที่มีประสบการณ์ด้านระบบสารสนเทศและการบริหารจัดการ Incident ในองค์กร โดยทั่วไป รายการที่มีค่า IOC $\geq$ 0.50 ถือว่ามีความสอดคล้องในระดับยอมรับได้ และหากค่า IOC $\geq$ 0.67 จะถือว่ามีความเหมาะสมในระดับดี การใช้ IOC ในงานวิจัยนี้มีวัตถุประสงค์เพื่อ ตรวจสอบว่าระบบ จำแนกประเภทปัญหาได้ตรงกับบริบทการทำงานจริง

ยืนยันว่าการจัดหมวดหมู่ปัญหาสอดคล้องกับโครงสร้างทีมพัฒนาในองค์กร และสนับสนุนแนวคิด Human-in-the-loop ซึ่งทำงานร่วมกับมนุษย์ ไม่ได้ทดแทนทั้งหมด

2.4 งานวิจัยที่เกี่ยวข้อง

จากการศึกษาวรรณกรรมและงานวิจัยที่เกี่ยวข้องกับเทคนิคเหมืองข้อความ (Text Mining) และการเรียนรู้ของเครื่อง (Machine Learning) เพื่อจำแนกความคิดเห็นและวิเคราะห์ข้อความภาษาไทยในบริบทที่หลากหลาย สามารถสรุปสาระสำคัญได้ดังนี้

วสวัตติ์ อินทร์แปลง และ จารี ทองคำ (2563) ได้ทำการศึกษาวิจัย เรื่อง การวิเคราะห์ความคิดเห็นต่อเกมมือถือพับจีด้วยเหมืองข้อความ พบว่า อัลกอริทึม **K-Nearest Neighbor มีประสิทธิภาพดีที่สุดในการวิเคราะห์ความคิดเห็น** โดยมีค่าความถูกต้องสูงสุด **99.87%** ตามมาด้วยค่าความแม่นยำ **99.75%** และค่าความระลึก **100%** ซึ่งสะท้อนถึงความโดดเด่นของ KNN ในการจัดกลุ่มความคิดเห็นเชิงบวกและลบบนฐานข้อมูลข้อความที่ผ่านการวิเคราะห์ด้วยเหมืองข้อมูลข้อความ (Text Mining)

Han และคณะ (2011) ศึกษาเรื่อง *Data Mining: Concepts and Techniques* ซึ่งเป็นตำราหลักด้านการทำเหมืองข้อมูล โดยอธิบายกระบวนการตั้งแต่การเตรียมข้อมูล การสร้างแบบจำลอง การจำแนกประเภท (Classification) การจัดกลุ่ม (Clustering) ตลอดจนการประเมินประสิทธิภาพของโมเดล ผลงานนี้สรุปว่าเทคนิคเหมืองข้อมูลสามารถประยุกต์ใช้กับข้อมูลขนาดใหญ่เพื่อสนับสนุนการตัดสินใจได้อย่างมีประสิทธิภาพ และถือเป็นพื้นฐานสำคัญของงาน Machine Learning สมัยใหม่

Feldman และ Sanger (2007) นำเสนอหนังสือ *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data* ซึ่งมุ่งเน้นการวิเคราะห์ข้อมูลข้อความที่ไม่มีโครงสร้าง (Unstructured Data) โดยเสนอเทคนิคการแปลงข้อความเป็นคุณลักษณะเชิงตัวเลข (Feature Representation) และวิธีการจำแนกข้อความ ผลการศึกษาชี้ให้เห็นว่า Text Mining สามารถสกัดสารสนเทศที่มีคุณค่าออกจากข้อมูลจำนวนมหาศาลได้อย่างเป็นระบบ

Patil และ Patil (2013) ศึกษาการสร้างระบบวิเคราะห์อารมณ์อัตโนมัติบนเครือข่ายสังคมออนไลน์ด้วยเทคนิค Text Mining โดยพัฒนาแบบจำลองเพื่อจำแนกข้อความตามอารมณ์ เช่น ดีใจ โกรธ หรือเศร้า ผลการทดลองพบว่าวิธีการทำเหมืองข้อความสามารถจำแนกอารมณ์ได้อย่างมีประสิทธิภาพ และช่วยเพิ่มความเข้าใจพฤติกรรมผู้ใช้งานบนแพลตฟอร์มออนไลน์

Murthy et al. (2020) วิจัยเรื่องการวิเคราะห์ความรู้สึกจากข้อความโดยใช้โมเดล Long Short-Term Memory (LSTM) ซึ่งเป็นเทคนิค Deep Learning สำหรับข้อมูลลำดับ ผลการทดลองพบว่า LSTM ให้ความแม่นยำสูงกว่าวิธี Machine Learning แบบดั้งเดิมในบางชุดข้อมูล โดยเฉพาะเมื่อข้อมูลมีบริบทต่อเนื่องและความซับซ้อนทางภาษา

Chowdhury และ Alzarrad (2023) ศึกษาการประยุกต์ใช้ Text Mining กับข้อมูลโครงสร้างพื้นฐานด้านคมนาคม โดยใช้ข้อมูลจากแหล่ง Big Data และสื่อสังคมออนไลน์ ผลการศึกษาพบว่าเทคนิคดังกล่าวสามารถช่วยวิเคราะห์แนวโน้มปัญหาและสนับสนุนการวางแผนนโยบายด้านการคมนาคมได้อย่างมีประสิทธิภาพ

Pratana และคณะ (2019) ทำการเปรียบเทียบอัลกอริทึม Support Vector Machine (SVM) และ Naive Bayes ในการวิเคราะห์ความรู้สึกจากข้อความบน Twitter เกี่ยวกับองค์กรตำรวจอินโดนีเซีย ผลการทดลองพบว่า SVM ให้ค่าความแม่นยำสูงกว่า Naive Bayes ในกรณีศึกษานี้

Lazard, Saffer และ Wilcox (2016) ศึกษาการทำเหมืองข้อความเกี่ยวกับบุหรี่ไฟฟ้าบน Twitter โดยวิเคราะห์ทั้งข้อความด้านการตลาดและการสนทนาของผู้บริโภค ผลการศึกษาพบว่าการใช้ Text Mining สามารถระบุรูปแบบการสื่อสารเชิงการตลาดและแนวโน้มพฤติกรรมผู้บริโภคได้อย่างชัดเจน

กิตติศักดิ์ ขำจิตร และคณะ (2565) ศึกษาประสิทธิภาพของเทคนิคเหมืองข้อมูลในการพยากรณ์การเกิดโรคหลอดเลือดในสมอง โดยเปรียบเทียบหลายอัลกอริทึม ผลการวิจัยพบว่าเทคนิคแบบต้นไม้ตัดสินใจและวิธี Ensemble ให้ผลลัพธ์ความแม่นยำสูง และสามารถนำไปใช้สนับสนุนการวินิจฉัยทางการแพทย์ได้

จิรรัฐ ศรีโชค และมณเฑียร รัตนศิริวงศ์วุฒิ (2564) พัฒนาระบบธุรกิจอัจฉริยะเพื่อวิเคราะห์ข้อมูลจากระบบรับแจ้งเหตุขัดข้อง โดยใช้เทคนิคเหมืองข้อมูลในการจำแนกประเภทเหตุการณ์ ผลการศึกษาพบว่าระบบสามารถลดเวลาในการวิเคราะห์ข้อมูลและช่วยให้ผู้บริหารตัดสินใจได้รวดเร็วขึ้น

ดำรงพล สุวรรณชาตรี (2564) ศึกษาการวิเคราะห์ความคิดเห็นเกี่ยวกับรถยนต์ไฟฟ้าบนสื่อสังคมออนไลน์ด้วยเทคนิคเหมืองข้อความ โดยใช้กระบวนการเตรียมข้อมูลและจำแนกความรู้สึก ผลการวิจัยพบว่าสามารถประเมินทัศนคติของผู้บริโภคได้อย่างมีประสิทธิภาพ และสะท้อนแนวโน้มความสนใจต่อเทคโนโลยียานยนต์ไฟฟ้าในสังคมออนไลน์

กัญญารัตน์ พูลเพิ่ม และแกมกาญจน์ สมประเสริฐศรี (2565) ได้ทำการศึกษาวิจัยเรื่องการจำแนกโครงการตามแผนแม่บทภายใต้ยุทธศาสตร์ชาติ โดยใช้เทคนิคเหมืองข้อความ เพื่อเปรียบเทียบประสิทธิภาพอัลกอริทึมในการจำแนกแบบจำลองโครงการ โดยใช้ชุดข้อมูลจากระบบติดตามและประเมินผลแห่งชาติ (eMenscr) จำนวน 1,150 โครงการ ทำการวิเคราะห์ผ่านอัลกอริทึม Support Vector Machine, Random Forest, Naïve Bayes และ K-Nearest Neighbor ผลการวิจัยพบว่า อัลกอริทึม Random Forest (ป่าสุ่ม) มีประสิทธิภาพสูงสุดในการจำแนกโครงการ โดยมีค่าความถูกต้อง (Accuracy) ร้อยละ 82.50 ค่าความแม่นยำร้อยละ 83.30 และค่าความระลึกร้อยละ 82.50 รวมถึงสามารถจำแนกโครงการใหม่ได้ถูกต้องแม่นยำถึงร้อยละ 90.91

ผู้วิจัยได้ทำการสรุปงานวิจัยที่เกี่ยวข้องแสดงได้ดังตารางที่ 2-1

ตารางที่ 2-1 สรุปงานวิจัยที่เกี่ยวข้อง

ชื่องานวิจัย	วิธีที่ใช้	ผลที่ได้มา
การวิเคราะห์ความคิดเห็นต่อเกมมือถือพับจีด้วยเหมืองข้อความ	- Naïve Bayes - Support Vector Machine - K-Nearest Neighbor - Decision Tree C4.5 - Random Forest	อัลกอริทึม K-Nearest Neighbor ให้ประสิทธิภาพดีที่สุดในการวิเคราะห์ความคิดเห็น โดยมีค่าความถูกต้อง 99.87% ค่าความแม่นยำ 99.75% และค่าความระลึก 100%
การวิเคราะห์ความคิดเห็นเกี่ยวกับรถยนต์ไฟฟ้าบนสื่อสังคมออนไลน์ด้วยเทคนิคเหมืองข้อความ	- Random Forest - Support Vector Machine - Naïve Bayes - Long Short-Term Memory - K-Nearest Neighbor	อัลกอริทึม Random Forest ให้ประสิทธิภาพดีที่สุด โดยมีความถูกต้อง 88.00% ค่าความแม่นยำ 95.00% และค่าความระลึก 80.00%
การจำแนกความรู้สึกในข้อความภาษาไทยจากผู้ใช้ชาวไทยในแพลตฟอร์มสื่อสังคมออนไลน์	- Naïve Bayes - Random Forest - Logistic Regression - Support Vector Machine	อัลกอริทึม Support Vector Machine ให้ประสิทธิภาพดีที่สุด โดยมีค่าความถูกต้อง 80.39% รองลงมาคือ Logistic Regression ได้ค่าความถูกต้อง 79.57%

ตารางที่ 2-1 (ต่อ)

ชื่องานวิจัย	วิธีที่ใช้	ผลที่ได้มา
การพัฒนาแบบจำลองในการตรวจจับข้อความภาษาไทยที่เป็นการกลั่นแกล้งทางไซเบอร์	- Support Vector Machine - K-Nearest Neighbor - Naïve Bayes	อัลกอริทึม Support Vector Machine ให้ประสิทธิภาพในการจำแนกข้อความมากที่สุด โดยมีค่าความถูกต้อง 83.91% ค่าความแม่นยำ 84.23% และค่าความระลึก 83.80%
การวิเคราะห์เหมืองความคิดเห็นโดยใช้เทคนิคการสกัดคำ	- Naïve Bayes - K-Nearest Neighbor - Decision Tree	อัลกอริทึม Naïve Bayes ให้ประสิทธิภาพสูงสุด โดยมีค่าความถูกต้อง 87.97% อัลกอริทึม K-Nearest Neighbor มีค่าความถูกต้อง 83.80%
การประยุกต์ใช้อัลกอริทึมทำเหมืองข้อความเพื่อสนับสนุนการตัดสินใจเลือกใช้ Food Delivery Application	- Naïve Bayes - Support Vector Machine - K-Nearest Neighbor	อัลกอริทึม Support Vector Machine ให้ประสิทธิภาพดีที่สุดในการจำแนกข้อความด้านบวกและด้านลบ โดยมีค่าความถูกต้อง 80.14% ค่าความระลึก 86.36% และค่าความแม่นยำ 79.36%
การพัฒนาระบบธุรกิจอัจฉริยะเพื่อการวิเคราะห์ข้อมูลจากระบบการรับแจ้งเหตุขัดข้องโดยใช้เทคนิคเหมืองข้อมูล	- Support Vector Machine - K-Nearest Neighbor - Naïve Bayes - Ensemble Learning	อัลกอริทึม K-Nearest Neighbor ให้ค่าความถูกต้องสูงสุด โดยมีค่าเท่ากับ 99.28% ค่าความแม่นยำ 99.28% และค่าความระลึก 99.28%
การจำแนกความคิดเห็นการใช้บริการร้านอาหารบนเครือข่ายสังคมออนไลน์ในจังหวัดอุบลราชธานี	- Support Vector Machine - Naïve Bayes - Decision Tree - K-Nearest Neighbor - LSTM	อัลกอริทึม Decision Tree ให้ค่าความถูกต้องสูงสุดที่ 89.00% แต่หากประเมินภาพรวม (F1-Score, Precision) อัลกอริทึม Support Vector Machine ให้ผลดีที่สุดที่ 90.00%

ตารางที่ 2-1 (ต่อ)

ชื่องานวิจัย	วิธีที่ใช้	ผลที่ได้มา
การใช้ Social Listening Tools ในการพยากรณ์ความคิดเห็นของคนไทยเกี่ยวกับการซื้อสินค้าออนไลน์	- K-Nearest Neighbor - Random Forest - Support Vector Machine - Ada Boosting- Bagging - LSTM, CNN ฯลฯ	อัลกอริทึม Ada Boosting ให้ประสิทธิภาพดีที่สุดในการพยากรณ์ โดยมีค่าความถูกต้องเฉลี่ย 99.12% รองลงมาคือ Gradient Boosting มีค่าความถูกต้อง 99.11%
การวิเคราะห์ความคิดเห็นของผู้ใช้งานทวิตเตอร์ที่มีต่อองค์กร กรณีศึกษา บริษัทไปรษณีย์ไทย จำกัด	- Bi-Directional LSTM - Long Short -Term Memory - Gated Recurrent Unit	อัลกอริทึม Bi-Directional LSTM ให้ประสิทธิภาพสูงที่สุด โดยมีค่าความถูกต้อง 94.87% ค่าความแม่นยำ 95.00% และค่าความระลึก 95.00%

บทที่ 3
วิธีดำเนินการวิจัย

การพัฒนาและประเมินประสิทธิภาพของระบบจำแนกประเภทปัญหาจาก Log Files แบบอัตโนมัติ โดยมีวัตถุประสงค์เพื่ออธิบายกระบวนการดำเนินงานอย่างเป็นระบบ ตั้งแต่การกำหนดกรอบแนวคิดของงานวิจัย การเก็บรวบรวมและเตรียมข้อมูล การพัฒนาและเปรียบเทียบแบบจำลองการเรียนรู้ของเครื่อง ตลอดจนการพัฒนาระบบแสดงผลและการประเมินผลการใช้งานจริง

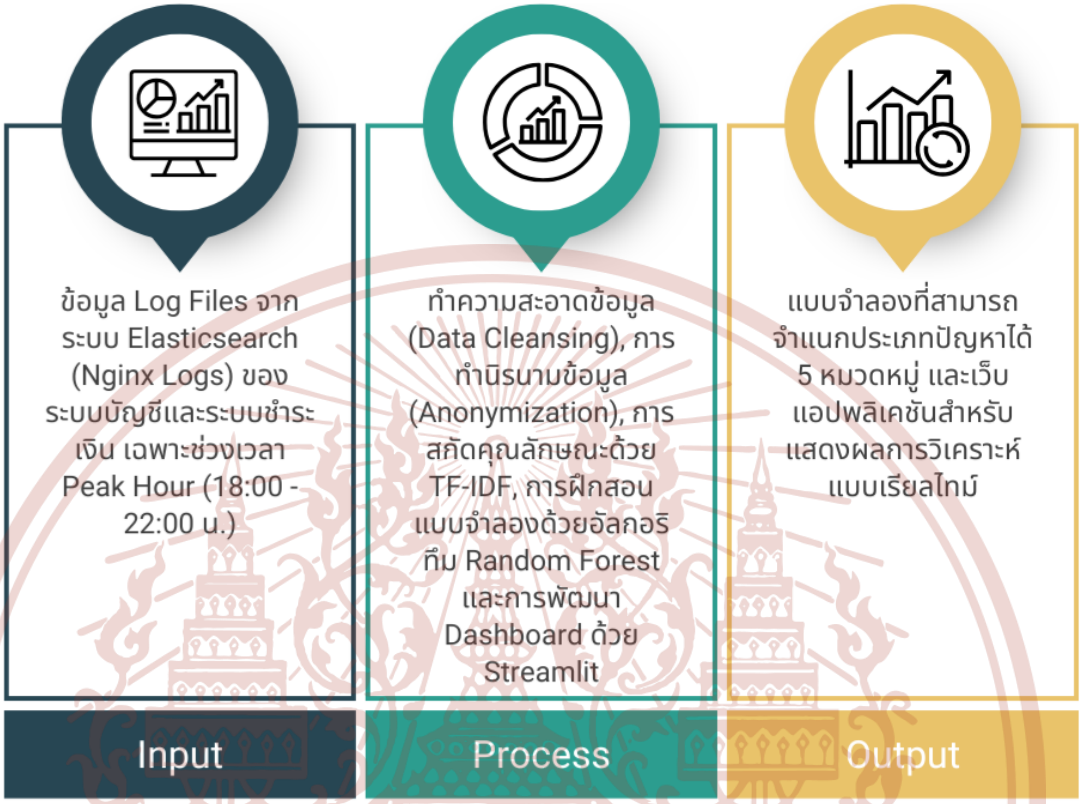
กระบวนการวิจัยถูกออกแบบภายใต้แนวคิดการวิจัยเชิงพัฒนา (Developmental Research) ที่มุ่งเน้นการสร้างต้นแบบระบบ (Prototype System) ควบคู่กับการทดสอบประสิทธิภาพเชิงปริมาณและการประเมินเชิงคุณภาพจากผู้เชี่ยวชาญและผู้ใช้งาน เพื่อให้มั่นใจว่าระบบที่พัฒนาขึ้นมีทั้งความถูกต้องเชิงเทคนิคและสามารถนำไปประยุกต์ใช้ในบริบทการปฏิบัติงานจริงได้อย่างมีประสิทธิภาพ ผู้วิจัยได้ดำเนินการวิจัยตามขั้นตอน ดังนี้

3.1 กรอบแนวคิดของงานวิจัย
3.2 การเก็บรวบรวมและเตรียมข้อมูล
3.3 การสร้างและเปรียบเทียบประสิทธิภาพอัลกอริทึม
3.4 การพัฒนาระบบและการแสดงผล
3.5 การประเมินประสิทธิภาพและความพึงพอใจ
3.6 สภาพแวดล้อมและเครื่องมือที่ใช้พัฒนาระบบ

3.1 กรอบแนวคิดของงานวิจัย

เพื่อให้เห็นภาพรวมของระบบที่พัฒนาขึ้น ผู้วิจัยได้ประยุกต์ใช้แบบจำลอง Input-Process-Output (IPO) ในการออกแบบโครงสร้างการวิจัย ดังนี้:

- ตัวป้อน (Input): ข้อมูล Log Files จากระบบ Elasticsearch (Nginx Logs) ของระบบบัญชีและระบบชำระเงิน เฉพาะช่วงเวลา Peak Hour (18:00 - 22:00 น.)
- กระบวนการ (Process): ประกอบด้วยการทำความสะอาดข้อมูล (Data Cleansing), การทำนิรนามข้อมูล (Anonymization), การสกัดคุณลักษณะด้วย TF-IDF, การฝึกสอนแบบจำลองด้วยอัลกอริทึม Random Forest และการพัฒนา Dashboard ด้วย Streamlit
- ผลลัพธ์ (Output): แบบจำลองที่สามารถจำแนกประเภทปัญหาได้ 5 หมวดหมู่ และเว็บแอปพลิเคชันสำหรับแสดงผลการวิเคราะห์แบบเรียลไทม์



ภาพที่ 3-1 แบบจำลอง Input-Process-Output (IPO)

การวิจัยครั้งนี้เป็นการวิจัยเชิงพัฒนา (Developmental Research) มีเป้าหมายเพื่อออกแบบ พัฒนา และประเมินประสิทธิภาพของระบบจำแนกประเภทปัญหาจาก Log Files แบบอัตโนมัติ โดยประยุกต์ใช้เทคนิคการเรียนรู้ของเครื่อง (Machine Learning) เพื่อสนับสนุนการวิเคราะห์ปัญหาเชิงปฏิบัติการของระบบสารสนเทศในองค์กรอย่างเป็นระบบและแม่นยำ

กรอบแนวคิดของการวิจัยได้รับการออกแบบตามโครงสร้างของกระบวนการพัฒนาแบบจำลองข้อมูล (Machine Learning Pipeline) ตั้งแต่ต้นน้ำถึงปลายน้ำ ครอบคลุมกระบวนการจัดการข้อมูล การสร้างแบบจำลอง การประเมินผล และการนำไปใช้งานจริง โดยแบ่งออกเป็น 3 ระยะหลัก ดังนี้

3.1.1 ระยะที่ 1 การจัดการข้อมูล (Data Management Phase)

ระยะนี้ถือเป็นขั้นตอนต้นน้ำของการพัฒนาแบบจำลอง ซึ่งมุ่งเน้นการเตรียมข้อมูลให้มีคุณภาพและเหมาะสมต่อการวิเคราะห์ โดยผู้วิจัยได้เชื่อมต่อและดึงข้อมูลจากระบบจัดเก็บ Log ส่วนกลางขององค์กรผ่าน Elasticsearch API ซึ่งเป็นแหล่งข้อมูลเชิงปฏิบัติการที่สะท้อนพฤติกรรมการทำงานของระบบแบบ Real-time

เพื่อให้ข้อมูลสะท้อนสถานการณ์ที่มีความซับซ้อนและภาระงานสูง ผู้วิจัยกำหนดช่วงเวลา Peak Hour ระหว่าง 18:00–22:00 น. ซึ่งเป็นช่วงที่มีปริมาณการใช้งานสูงสุด (High Traffic Load) และมีแนวโน้มเกิดความผิดปกติของระบบมากที่สุด จากนั้นดำเนินการทำความสะอาด

ข้อมูล (Data Cleansing) การจัดการข้อมูลละเอียดอ่อน (Sensitive Data Handling) และการกำจัดค่าที่ไม่สมบูรณ์ เพื่อเพิ่มความถูกต้องและความน่าเชื่อถือของข้อมูล

ในขั้นตอนการติดป้ายกำกับ (Labeling) ผู้วิจัยกำหนดกรอบการจำแนกประเภทปัญหาโดยอิงจากมาตรฐานรหัสสถานะ HTTP และความเห็นของทีมปฏิบัติการ (Operation Team) เพื่อให้สอดคล้องกับบริบทการใช้งานจริง โดยแบ่งออกเป็น 5 หมวดหมู่หลัก ได้แก่

3.1.1.1 สถานะการทำงานปกติ (Success) คือหมวดหมู่นี้ครอบคลุมรหัสสถานะที่บ่งชี้ว่าระบบสามารถตอบสนองคำขอได้อย่างสมบูรณ์ เช่น 200 (OK) และ 304 (Not Modified) ซึ่งสะท้อนถึงการทำงานปกติของระบบ ทั้งในด้านความถูกต้องของข้อมูลและประสิทธิภาพการประมวลผล การกำหนดหมวดหมู่นี้มีความสำคัญในการสร้างฐานอ้างอิง (Baseline) สำหรับเปรียบเทียบกับเหตุการณ์ผิดปกติ และช่วยให้โมเดลสามารถแยกแยะระหว่างเหตุการณ์ปกติกับเหตุการณ์ที่ต้องได้รับการตรวจสอบได้อย่างชัดเจน

3.1.1.2 ความผิดพลาดจากฝั่งผู้ใช้ (Client_Error) คือ หมวดหมู่นี้ครอบคลุมรหัสสถานะในกลุ่ม 4xx ที่เกิดจากความผิดพลาดในการร้องขอข้อมูลของผู้ใช้ เช่น 400 (Bad Request), 403 (Forbidden) และ 405 (Method Not Allowed) ซึ่งโดยทั่วไปไม่ได้เกิดจากความล้มเหลวของระบบหลัก แต่เกิดจากการกำหนดคำขอที่ไม่ถูกต้อง การขาดสิทธิ์ หรือการเรียกใช้วิธีการที่ไม่รองรับ การแยกหมวดหมู่นี้ออกจาก Server_Error มีความสำคัญเชิงการบริหารจัดการ เนื่องจากแนวทางการแก้ไขปัญหามีลักษณะแตกต่างกันอย่างชัดเจน

3.1.1.3 ความผิดพลาดจากฝั่งเซิร์ฟเวอร์ (Server_Error) คือ หมวดหมู่นี้ครอบคลุมรหัสสถานะในกลุ่ม 5xx เช่น 500 (Internal Server Error), 502 (Bad Gateway) และ 504 (Gateway Timeout) ซึ่งสะท้อนถึงความล้มเหลวของระบบประมวลผลฝั่งเซิร์ฟเวอร์หรือโครงสร้างพื้นฐาน (Infrastructure) ปัญหาในกลุ่มนี้มักมีผลกระทบต่อผู้ใช้งานจำนวนมากและส่งผลโดยตรงต่อความน่าเชื่อถือของระบบ การจำแนกหมวดหมู่นี้อย่างแม่นยำจึงมีความสำคัญเชิงกลยุทธ์ในการจัดลำดับความเร่งด่วนของการแก้ไขปัญหา

3.1.1.4 การยกเลิกคำขอโดยผู้ใช้หลังใช้เวลานาน (Client_Timeout (499)) คือรหัสสถานะ 499 เป็นรหัสเฉพาะของ Nginx ที่บ่งชี้ว่าผู้ใช้ยกเลิกการเชื่อมต่อก่อนที่เซิร์ฟเวอร์จะตอบสนองเสร็จสมบูรณ์ ในการวิจัยนี้ ผู้วิจัยกำหนดเงื่อนไขเพิ่มเติมโดยพิจารณาค่า request_time มากกว่า 3.0 วินาที เพื่อสะท้อนถึงเหตุการณ์ที่อาจเกิดจากความล่าช้าของระบบ แม้ต้นเหตุจะปรากฏเป็นการยกเลิกจากฝั่งผู้ใช้ก็ตาม การกำหนดเกณฑ์เชิงเวลา (Time Threshold) ดังกล่าวช่วยเพิ่มความละเอียดในการจำแนก และลดความคลาดเคลื่อนของการตีความข้อมูล

3.1.1.5 การปฏิเสธการเชื่อมต่อเพื่อเหตุผลด้านความปลอดภัย (Security_Block (444)) คือ รหัสสถานะ 444 เป็นรหัสที่ใช้ใน Nginx เพื่อยุติการเชื่อมต่อโดยไม่ส่งข้อมูลตอบกลับ มักเกิดจากการตรวจจับพฤติกรรมต้องสงสัย เช่น การโจมตีแบบ Bot หรือการเข้าถึงที่ละเมิดนโยบายความปลอดภัย การจัดหมวดหมู่นี้แยกต่างหากจาก Client_Error ช่วยให้สามารถติดตามแนวโน้มด้านความมั่นคงปลอดภัยไซเบอร์ (Cybersecurity Monitoring) ได้อย่างชัดเจน และสนับสนุนการวางแผนป้องกันเชิงรุกในอนาคต

โดยสรุป กระบวนการติดป้ายกำกับในงานวิจัยนี้มิได้เป็นเพียงการจัดกลุ่มตามรหัสสถานะเท่านั้น แต่เป็นการออกแบบโครงสร้างการจำแนกที่สอดคล้องกับมิติทางเทคนิค มิติการปฏิบัติงาน และมิติการบริหารจัดการเหตุการณ์ (Incident Management) ซึ่งมีบทบาทสำคัญต่อความแม่นยำของแบบจำลองและประสิทธิภาพของระบบสนับสนุนการตัดสินใจที่พัฒนาขึ้นในลำดับถัดไป

3.1.2 ระยะที่ 2 การพัฒนาและเปรียบเทียบโมเดล (Modeling & Comparison) ในระยะนี้ ผู้วิจัยมุ่งเน้นการพัฒนาแบบจำลองการเรียนรู้ของเครื่องเพื่อจำแนกประเภทปัญหาจาก Log Files อย่างมีประสิทธิภาพ โดยดำเนินการภายใต้แนวคิดของกระบวนการ Machine Learning Pipeline ที่เป็นระบบและสามารถทำซ้ำได้ (Reproducible Framework) ในระยะนี้ ผู้วิจัยมุ่งเน้นการพัฒนาแบบจำลองการเรียนรู้ของเครื่องเพื่อจำแนกประเภทปัญหาจาก Log Files อย่างมีประสิทธิภาพ โดยดำเนินการภายใต้แนวคิดของกระบวนการ Machine Learning Pipeline ที่เป็นระบบและสามารถทำซ้ำได้ (Reproducible Framework) หลังจากจัดเตรียมข้อมูลในรูปแบบข้อความแล้ว ผู้วิจัยดำเนินการแปลงข้อมูลให้อยู่ในรูปเวกเตอร์เชิงตัวเลขด้วยเทคนิค TF-IDF (Term Frequency–Inverse Document Frequency) ซึ่งเป็นวิธีการถ่วงน้ำหนักคำที่สะท้อนความสำคัญของคำในแต่ละเอกสารเมื่อเทียบกับทั้งชุดข้อมูล

หลักการของ TF-IDF ช่วยเพิ่มน้ำหนักให้กับคำที่ปรากฏเฉพาะในบริบทของปัญหาบางประเภท เช่น คำที่เกี่ยวข้องกับ error endpoint หรือคำที่บ่งชี้ความล้มเหลว และลดอิทธิพลของคำทั่วไปที่ปรากฏบ่อยแต่ไม่มีนัยสำคัญเชิงจำแนก การเลือกใช้ TF-IDF จึงเหมาะสมกับข้อมูลประเภท Log ที่มีลักษณะเป็นข้อความกึ่งโครงสร้าง (Semi-Structured Text) และมีคำซ้ำจำนวนมาก ผู้วิจัยได้คัดเลือกอัลกอริทึมที่มีหลักการทำงานแตกต่างกัน 3 ประเภท ได้แก่

3.1.2.1 **Naïve Bayes (MultinomialNB)** เป็นโมเดลเชิงความน่าจะเป็นที่ตั้งอยู่บนทฤษฎีบทของเบย์ (Bayes' Theorem) และสมมติฐานความเป็นอิสระของคุณลักษณะ (Conditional Independence Assumption) แม้สมมติฐานดังกล่าวอาจไม่สอดคล้องกับข้อมูลจริงทั้งหมด แต่โมเดลนี้มีข้อดีด้านความเร็วในการประมวลผลและความเหมาะสมกับข้อมูลข้อความที่ผ่านการถ่วงน้ำหนักแบบ TF-IDF การเลือก MultinomialNB ในงานวิจัยนี้มีวัตถุประสงค์เพื่อเป็น Baseline Model สำหรับเปรียบเทียบประสิทธิภาพกับโมเดลที่มีความซับซ้อนมากขึ้น

3.1.2.2 **Random Forest** เป็นโมเดลในกลุ่ม Ensemble Learning ที่สร้างต้นไม้ตัดสินใจหลายต้น (Decision Trees) และรวมผลลัพธ์ผ่านกลไกการโหวต (Majority Voting) เพื่อลดความแปรปรวน (Variance) และเพิ่มความเสถียรของการทำนาย ข้อได้เปรียบของ Random Forest คือความสามารถในการจัดการข้อมูลที่มีความสัมพันธ์ไม่เชิงเส้น (Non-linear Relationship) และลดปัญหา Overfitting เมื่อเทียบกับ Decision Tree เพียงต้นเดียว การนำโมเดลนี้มาใช้จึงช่วยทดสอบว่าการจำแนกปัญหาจาก Log Files มีลักษณะเชิงเส้นหรือไม่เชิงเส้น และเหมาะสมกับโมเดลประเภทใดมากกว่า

3.1.2.3 **Support Vector Machine (SVM)** เป็นโมเดลในกลุ่ม Ensemble Learning ที่สร้างต้นไม้ตัดสินใจหลายต้น (Decision Trees) และรวมผลลัพธ์ผ่านกลไกการโหวต (Majority Voting) เพื่อลดความแปรปรวน (Variance) และเพิ่มความเสถียรของการทำนาย

ข้อได้เปรียบของ Random Forest คือความสามารถในการจัดการข้อมูลที่มีความสัมพันธ์ไม่เชิงเส้น (Non-linear Relationship) และลดปัญหา Overfitting เมื่อเทียบกับ Decision Tree เพียงต้นเดียว การนำโมเดลนี้มาใช้จึงช่วยทดสอบว่าการจำแนกปัญหาจาก Log Files มีลักษณะเชิงเส้นหรือไม่เชิงเส้น และเหมาะสมกับโมเดลประเภทใดมากกว่า การทดลองใช้วิธีแบ่งข้อมูลเป็น Training Set (80%) และ Testing Set (20%) พร้อมกำหนดค่า random_state เพื่อความสามารถในการทำซ้ำผลการทดลอง (Reproducibility) และประเมินผลด้วยตัวชี้วัดมาตรฐาน ได้แก่ Accuracy, Precision, Recall และ F1-Score การเปรียบเทียบดังกล่าวช่วยให้สามารถคัดเลือกโมเดลที่มีสมรรถนะสูงสุดสำหรับนำไปใช้งานจริงในระยะต่อไป

การเปรียบเทียบโมเดลทั้งสามประเภท ซึ่งมีพื้นฐานทางทฤษฎีแตกต่างกัน ได้แก่ โมเดลเชิงความน่าจะเป็น โมเดลแบบ Ensemble และโมเดลเชิงเรขาคณิต ช่วยให้การวิจัยมีความครอบคลุมเชิงวิธีวิทยา และสามารถคัดเลือกโมเดลที่มีสมรรถนะสูงสุดทั้งในด้านความแม่นยำ ความเสถียร และความเหมาะสมกับบริบทการใช้งานจริง ผลจากระยะนี้จะถูกนำไปใช้เป็นพื้นฐานในการพัฒนาระบบต้นแบบในระยะถัดไป เพื่อยกระดับการวิเคราะห์ Log Files จากกระบวนการตรวจสอบแบบดั้งเดิม ไปสู่ระบบวิเคราะห์เชิงอัตโนมัติที่สนับสนุนการตัดสินใจอย่างมีประสิทธิภาพ

3.1.3 ระยะที่ 3 การพัฒนาระบบแสดงผลและการประเมิน (System Deployment & Evaluation) ภายหลังจากการคัดเลือกแบบจำลองที่มีประสิทธิภาพสูงสุดจากระยะการทดลอง ผู้วิจัยได้ดำเนินการพัฒนาระบบต้นแบบ (Prototype System) เพื่อประยุกต์ใช้แบบจำลองในบริบทการปฏิบัติงานจริง โดยมีวัตถุประสงค์เพื่อทดสอบความสามารถในการใช้งานเชิงปฏิบัติ ความรวดเร็วในการประมวลผล และความถูกต้องของผลลัพธ์ที่ได้จากการทำนาย

3.1.3.1 การพัฒนา Web Application เพื่อรองรับการวิเคราะห์แบบทันที (Real-time Inference) ผู้วิจัยเลือกใช้ไลบรารี Streamlit บนภาษา Python ในการพัฒนา Web Application เนื่องจากเป็นเครื่องมือที่เหมาะสมสำหรับการสร้างระบบวิเคราะห์ข้อมูลและโมเดล Machine Learning ในรูปแบบ Interactive Dashboard ที่สามารถพัฒนาได้รวดเร็วและรองรับการประมวลผลแบบทันที

3.1.3.2 การประเมินความเที่ยงตรงของระบบ (System Validation) เพื่อยืนยันความน่าเชื่อถือของระบบในบริบทการใช้งานจริง ผู้วิจัยดำเนินการประเมินความเที่ยงตรงเชิงเนื้อหา (Content Validity) โดยใช้วิธี Index of Item-Objective Congruence (IOC)

เพื่อยืนยันความน่าเชื่อถือของระบบในบริบทการใช้งานจริง ผู้วิจัยได้ประเมินความเที่ยงตรงเชิงเนื้อหา (Content Validity) ด้วยวิธี Index of Item-Objective Congruence (IOC) โดยเชิญผู้เชี่ยวชาญด้านระบบสารสนเทศและเหมืองข้อความพิจารณาความสอดคล้องระหว่างองค์ประกอบของระบบกับวัตถุประสงค์การวิจัย ผลการประเมินสะท้อนว่าระบบมีความสอดคล้องในระดับเหมาะสม สามารถนำไปประยุกต์ใช้เพื่อวิเคราะห์ความคิดเห็นรถยนต์ไฟฟ้าบนสื่อสังคมออนไลน์ได้อย่างมีประสิทธิภาพในสถานการณ์จริง

3.2 การเก็บรวบรวมและเตรียมข้อมูล

การเก็บรวบรวมและเตรียมข้อมูลนับเป็นขั้นตอนสำคัญที่ส่งผลโดยตรงต่อความแม่นยำและความน่าเชื่อถือของแบบจำลอง เนื่องจากข้อมูล Log ของระบบสารสนเทศมีลักษณะเป็นข้อมูลกึ่งโครงสร้าง (Semi-structured Data) ที่มีปริมาณมาก ความหลากหลายสูง และอาจแฝงด้วยข้อมูลละเอียดอ่อน การดำเนินการในระยะนี้จึงมุ่งเน้นทั้งด้านความครบถ้วนของข้อมูล ความสอดคล้องกับวัตถุประสงค์การวิจัย และการกำกับดูแลข้อมูลตามหลักความมั่นคงปลอดภัยและการคุ้มครองข้อมูลส่วนบุคคล กระบวนการประกอบด้วย 3 ขั้นตอนหลัก ได้แก่

3.2.1 การสกัดข้อมูลจากแหล่งกำเนิด (Data Extraction)

ผู้วิจัยดำเนินการสกัดข้อมูลจากระบบจัดเก็บ Log ส่วนกลางขององค์กรผ่านแพลตฟอร์ม Elasticsearch ซึ่งเป็นระบบจัดการและค้นคืนข้อมูลแบบกระจายศูนย์ (Distributed Search and Analytics Engine) ที่รองรับข้อมูลขนาดใหญ่ (Big Data) และสามารถสืบค้นเชิงเวลาจริง (Near Real-time Search) ได้อย่างมีประสิทธิภาพ การเลือกใช้แพลตฟอร์มดังกล่าวช่วยให้สามารถกำหนดเงื่อนไขการคัดกรองข้อมูลเชิงเวลาและเชิงโครงสร้างได้อย่างแม่นยำ สอดคล้องกับวัตถุประสงค์ของการวิเคราะห์พฤติกรรมและเหตุการณ์ผิดปกติของระบบ

3.2.1.1 Index Pattern เลือกศึกษาจาก web34-access-* ซึ่งเป็นชุดข้อมูล Access Log ของระบบบัญชีและระบบชำระเงิน อันเป็นระบบที่มีความสำคัญเชิงภารกิจ (Mission-Critical System) และมีปริมาณธุรกรรมสูง ส่งผลให้มีความหลากหลายของเหตุการณ์ทั้งปกติและผิดปกติ เหมาะสมต่อการนำมาพัฒนาแบบจำลองจำแนกประเภทเหตุการณ์

3.2.1.2 **Sampling Strategy** ใช้วิธีการสุ่มข้อมูลรายวัน (Daily Sampling) โดยเจาะจงช่วงเวลา Peak Hour ระหว่าง 18:00–22:00 น. (เวลาประเทศไทย) ย้อนหลัง 7 วัน เพื่อสะท้อนพฤติกรรมการใช้งานในสภาวะที่มีภาระงานสูง (High Traffic Load) ซึ่งเป็นช่วงเวลาที่มีความหนาแน่นของคำร้องขอ (Request) มากที่สุด และมีโอกาสเกิดความผิดพลาดหรือความล่าช้าของระบบได้สูง การกำหนดกรอบเวลาที่ชัดเจนช่วยลดอคติจากช่วงเวลาที่มีภาระงานต่ำและเพิ่มความเข้มข้นของข้อมูลเชิงเหตุการณ์

3.2.1.3 **Data Format:** ข้อมูลถูกดึงผ่าน Python API และบันทึกในรูปแบบ CSV เพื่อความสะดวกในการประมวลผลและนำเข้าสู่กระบวนการวิเคราะห์ขั้นถัดไป โดยประกอบด้วยฟิลด์สำคัญ ได้แก่ @timestamp (เวลาเกิดเหตุการณ์), http_host (ปลายทางของคำร้องขอ), request_uri (เส้นทางทรัพยากรที่ถูกเรียกใช้), status (HTTP Status Code), upstream_status (สถานะจากระบบปลายทาง), และ request_time (ระยะเวลาในการตอบสนอง) ซึ่งเป็นตัวแปรหลักที่สะท้อนบริบทการทำงานและประสิทธิภาพของระบบอย่างครอบคลุม

3.2.2 การทำความสะอาดและการจัดการข้อมูลละเอียดอ่อน

เนื่องจากข้อมูลประเภท Access Log อาจมีองค์ประกอบที่สามารถระบุตัวบุคคลได้โดยตรงหรือโดยอ้อม ผู้วิจัยจึงให้ความสำคัญกับการกำกับดูแลข้อมูล (Data Governance) ควบคู่กับการเตรียมข้อมูลเชิงเทคนิค เพื่อให้กระบวนการวิจัยสอดคล้องกับหลักจริยธรรมและข้อกำหนดด้านการคุ้มครองข้อมูลส่วนบุคคล

3.2.2.1 Sensitive Data Handling ดำเนินการลบข้อมูลในส่วนของ cookie, auth_token และพารามิเตอร์ที่มีลักษณะระบุตัวตนผู้ใช้ซึ่งอาจติดมากับ request_uri ออกก่อนเข้าสู่กระบวนการวิเคราะห์ โดยใช้วิธีการกรอง (Filtering) และการทำให้ข้อมูลไม่สามารถระบุตัวบุคคลได้ (De-identification) เพื่อให้สอดคล้องกับหลักการคุ้มครองข้อมูลส่วนบุคคลตามพระราชบัญญัติคุ้มครองข้อมูลส่วนบุคคล (PDPA) และแนวปฏิบัติด้านความมั่นคงปลอดภัยสารสนเทศขององค์กร

3.2.2.2 Data Cleansing ในเชิงเทคนิค ผู้วิจัยได้ดำเนินการกำจัดอักขระพิเศษ (Special Characters) และรูปแบบข้อความที่ไม่เป็นมาตรฐาน ซึ่งอาจส่งผลต่อกระบวนการแปลงข้อมูลเป็นคุณลักษณะ (Feature Extraction) ในขั้นตอนถัดไป นอกจากนี้ ได้จัดการค่าที่หายไป (Null Values) ในฟิลด์ upstream_status โดยแทนที่ด้วยค่าสถานะที่เหมาะสมหรือค่ากลางที่กำหนดตามบริบทของระบบ เพื่อป้องกันความคลาดเคลื่อนในการฝึกแบบจำลอง และลดความเสี่ยงของ Bias ที่เกิดจากข้อมูลไม่สมบูรณ์

กระบวนการดังกล่าวช่วยเพิ่มความถูกต้อง ความสอดคล้อง และความปลอดภัยของชุดข้อมูล ก่อนนำเข้าสู่ขั้นตอนการติดป้ายกำกับและการสกัดคุณลักษณะในลำดับถัดไป

3.2.3 การติดป้ายกำกับและการสกัดคุณลักษณะ (Labeling & Feature Engineering) ภายหลังจากการทำความสะอาดข้อมูล ผู้วิจัยดำเนินการติดป้ายกำกับ (Labeling) และออกแบบคุณลักษณะ (Feature Engineering) เพื่อเตรียมข้อมูลสำหรับการพัฒนาแบบจำลองจำแนกประเภทเหตุการณ์ โดยมุ่งให้ข้อมูลสะท้อนบริบทเชิงเทคนิคของระบบได้อย่างครบถ้วนและสามารถเรียนรู้รูปแบบ (Pattern) ที่สัมพันธ์กับปัญหาได้อย่างมีประสิทธิภาพ

3.2.3.1 Categorization การกำหนดหมวดหมู่ปัญหาอ้างอิงทั้งหลักเกณฑ์มาตรฐานของ HTTP Status Code และความเห็นเชิงปฏิบัติการของทีม Operation เพื่อให้สอดคล้องกับบริบทการใช้งานจริงขององค์กร โดยแบ่งออกเป็น 5 กลุ่ม ได้แก่

3.2.3.1.1 **Success:** Status 200, 304 ซึ่งสะท้อนการตอบสนองของระบบที่สมบูรณ์และเป็นไปตามเงื่อนไขปกติ

3.2.3.1.2 **Client_Error:** Status 400, 403, 405 แสดงถึงข้อผิดพลาดที่มีสาเหตุจากคำร้องขอของผู้ใช้งาน

3.2.3.1.3 **Server_Error:** Status 500, 502, 504 บ่งชี้ถึงความผิดพลาดภายในระบบหรือบริการปลายทาง

3.2.3.1.4 **Client_Timeout:** Status 499 และมีค่า request_time มากกว่า 3.0 วินาที ซึ่งสะท้อนกรณีที่ผู้ใช้ยกเลิกคำร้องขอภายหลังรอเกินเกณฑ์ที่กำหนด

3.2.3.1.5 **Security_Block:** Status 444 (Nginx No Response) ซึ่งเป็นกลไกการปฏิเสธการเชื่อมต่อของเว็บเซิร์ฟเวอร์ Nginx

แนวทางดังกล่าวช่วยให้ป้ายกำกับครอบคลุมมิติด้านประสิทธิภาพ (Performance), ความผิดพลาดของระบบ (Error Handling) และความมั่นคงปลอดภัย (Security) ในเชิงปฏิบัติการ ส่งผลให้ชุดข้อมูลมีความสอดคล้องกับสถานการณ์ใช้งานจริง และเหมาะสมต่อการนำไปพัฒนาแบบจำลองจำแนกประเภทเหตุการณ์ในลำดับถัดไป

3.2.3.2 Data Fusion ผู้วิจัยดำเนินการรวมคุณลักษณะ (Feature Fusion) ระหว่าง request_uri, status และ Time Tag ให้อยู่ในรูปข้อความเดียว เพื่อเพิ่มบริบทของเหตุการณ์ (Contextual Representation) แทนการพิจารณาตัวแปรแบบแยกส่วน วิธีการนี้ช่วยให้แบบจำลองสามารถเรียนรู้ความสัมพันธ์เชิงผสม (Interaction Effects) ระหว่างเส้นทางการเรียกใช้งาน สถานะการตอบสนอง และช่วงเวลาได้อย่างมีประสิทธิภาพมากขึ้น

3.2.3.3 Vectorization ข้อความที่ผ่านการรวมคุณลักษณะถูกแปลงให้อยู่ในรูปเวกเตอร์เชิงตัวเลขด้วยเทคนิค TF-IDF (Term Frequency–Inverse Document Frequency) ซึ่งเป็นวิธีถ่วงน้ำหนักคำตามความสำคัญเชิงสถิติภายในชุดข้อมูล เทคนิคนี้ช่วยเน้นคำหรือรูปแบบที่มีศักยภาพในการจำแนกประเภทเหตุการณ์ และลดอิทธิพลของคำที่ปรากฏบ่อยแต่ไม่ช่วยในการแยกกลุ่ม ส่งผลให้ข้อมูลมีความเหมาะสมต่อการนำไปพัฒนาและฝึกแบบจำลองการเรียนรู้ของเครื่องในขั้นตอนถัดไป

3.2.4 การจัดการข้อมูลที่ไม่สมดุล (Imbalanced Data Handling) เนื่องจากชุดข้อมูลไฟล์บันทึกเหตุการณ์ (Log Files) ที่รวบรวมมามีลักษณะความไม่สมดุลระหว่างหมวดหมู่ (Class Imbalance) อย่างชัดเจน โดยหมวดหมู่ความผิดปกติ (เช่น Server_Error หรือ Client_Timeout) มักจะเกิดขึ้นน้อยกว่าสถานะการทำงานปกติ (Success) อย่างมีนัยสำคัญ ผู้วิจัยจึงดำเนินการจัดการข้อมูลด้วยเทคนิคการปรับสมดุล เช่น การสุ่มเพิ่มข้อมูลคลาสกลุ่มน้อย (Over-sampling) หรือเทคนิค SMOTE (Synthetic Minority Over-sampling Technique) โดยจะนำมาปรับใช้กับชุดข้อมูลสำหรับฝึกสอน (Training Data) เท่านั้น การดำเนินการดังกล่าวมีวัตถุประสงค์เพื่อป้องกันไม่ให้แบบจำลองเกิดความเอนเอียง (Bias) ไปทำนายเฉพาะคลาสส่วนใหญ่ และช่วยให้แบบจำลองสามารถเรียนรู้รูปแบบของปัญหาที่เกิดขึ้นไม่บ่อยได้อย่างมีประสิทธิภาพก่อนนำเข้าสู่กระบวนการฝึกสอนแบบจำลอง

3.3 การสร้างและเปรียบเทียบประสิทธิภาพอัลกอริทึม

ภายหลังจากการเตรียมข้อมูลและแปลงคุณลักษณะเป็นเวกเตอร์เชิงตัวเลขแล้ว ผู้วิจัยดำเนินการพัฒนาแบบจำลองการเรียนรู้ของเครื่อง (Machine Learning Models) เพื่อจำแนกประเภทเหตุการณ์จากข้อมูล Access Log และทำการเปรียบเทียบประสิทธิภาพของอัลกอริทึมที่มีหลักการทำงานแตกต่างกัน ทั้งในเชิงสถิติ เชิงโครงสร้างต้นไม้ และเชิงเรขาคณิตของพื้นที่ข้อมูล เพื่อประเมินความเหมาะสมต่อบริบทข้อมูลขององค์กร

3.3.1 อัลกอริทึมที่ใช้ในการทดลอง

ผู้วิจัยคัดเลือกอัลกอริทึมจำนวน 3 วิธี ซึ่งเป็นตัวแทนของแนวคิดการเรียนรู้ที่แตกต่างกัน ทั้งเชิงความน่าจะเป็น เชิงการรวมแบบจำลอง และเชิงเรขาคณิตของพื้นที่ข้อมูล เพื่อให้สามารถเปรียบเทียบประสิทธิภาพได้อย่างรอบด้าน ดังนี้

3.3.1.1 Naïve Bayes (MultinomialNB) เป็นโมเดลเชิงความน่าจะเป็น (Probabilistic Model) ที่อาศัยทฤษฎีบทของเบย์ภายใต้สมมติฐานว่าคุณลักษณะมีความเป็นอิสระต่อกันแบบมีเงื่อนไข (Conditional Independence) เหมาะสำหรับข้อมูลเชิงข้อความที่ผ่านการแปลงด้วยเทคนิค TF-IDF เนื่องจากสามารถจัดการข้อมูลมิติสูงได้อย่างมีประสิทธิภาพ มี

ความซับซ้อนเชิงคำนวณต่ำ และให้ผลลัพธ์ที่รวดเร็ว เหมาะเป็นโมเดลพื้นฐาน (Baseline Model) สำหรับการเปรียบเทียบ

3.3.1.2 Random Forest (RF) เป็นโมเดลในกลุ่ม Ensemble Learning ที่ผสานผลลัพธ์จากต้นไม้ตัดสินใจหลายต้น (Decision Trees) ผ่านกระบวนการสุ่มตัวอย่างข้อมูลและคุณลักษณะ (Bootstrap Aggregation: Bagging) เพื่อลดความแปรปรวนของโมเดลและเพิ่มความสามารถในการทั่วไป (Generalization) ในการศึกษานี้กำหนดจำนวนต้นไม้ (n_estimators) เท่ากับ 100 เพื่อสร้างสมดุลระหว่างความแม่นยำและเวลาในการประมวลผล ทั้งยังช่วยลดปัญหา Overfitting เมื่อเทียบกับต้นไม้เดี่ยว

3.3.1.3 Support Vector Machine (SVM) เป็นโมเดลที่อาศัยหลักการหาขอบเขตการแบ่งข้อมูล (Optimal Hyperplane) ที่สามารถแยกกลุ่มข้อมูลได้ด้วยระยะขอบ (Margin) สูงสุดในพื้นที่เวกเตอร์ การเลือกใช้ Linear Kernel มีความเหมาะสมกับข้อมูลที่มีมิติสูงจาก TF-IDF และช่วยควบคุมความซับซ้อนของแบบจำลอง ลดภาระการคำนวณเมื่อเทียบกับ Kernel แบบไม่เชิงเส้น

การเลือกอัลกอริทึมทั้งสามวิธีช่วยให้สามารถเปรียบเทียบประสิทธิภาพระหว่างโมเดลเชิงสถิติ โมเดลแบบรวมกลุ่ม (Ensemble) และโมเดลเชิงเรขาคณิตได้อย่างเป็นระบบ

3.3.2 ขั้นตอนการทดสอบ (Model Training & Testing)

ในการประเมินประสิทธิภาพของแบบจำลอง ผู้วิจัยกำหนดกระบวนการทดลองอย่างเป็นระบบเพื่อให้ผลลัพธ์มีความน่าเชื่อถือ สามารถตรวจสอบซ้ำได้ (Reproducibility) และสะท้อนสมรรถนะที่แท้จริงของโมเดลเมื่อใช้งานกับข้อมูลที่ไม่เคยพบมาก่อน (Unseen Data)

3.3.2.1 Data Splitting: ข้อมูลทั้งหมดถูกแบ่งออกเป็น 2 ส่วน ได้แก่ ชุดฝึกสอน (Training Set) ร้อยละ 80 สำหรับใช้ในการเรียนรู้พารามิเตอร์ของโมเดล และชุดทดสอบ (Testing Set) ร้อยละ 20 สำหรับประเมินผลลัพธ์ การแบ่งข้อมูลในอัตราส่วน 80:20 เป็นมาตรฐานที่เหมาะสมสำหรับชุดข้อมูลขนาดกลางถึงใหญ่ เนื่องจากชุดข้อมูลสำหรับเรียนรู้ (Training Data) ร้อยละ 80 มีปริมาณเพียงพอให้แบบจำลองสามารถเรียนรู้ความสัมพันธ์ของตัวแปรได้อย่างซับซ้อน ในขณะที่ชุดข้อมูลทดสอบ (Testing Data) ร้อยละ 20 มีขนาดใหญ่พอที่จะสะท้อนประสิทธิภาพและความแม่นยำที่แท้จริงของแบบจำลองต่อข้อมูลที่ไม่เคยพบมาก่อน (Unseen Data) ซึ่งสอดคล้องกับงานวิจัยด้านการจำแนกข้อความภาษาไทยที่ได้รับการยอมรับ นอกจากนี้ ผู้วิจัยได้กำหนดค่า random_state = 42 เพื่อควบคุมกระบวนการสุ่มตัวอย่างให้สามารถทำซ้ำการทดลองได้ (Reproducibility) และลดความคลาดเคลื่อนที่อาจเกิดจากการสุ่มข้อมูลต่างกันในแต่ละครั้ง

3.3.2.2 Evaluation Metrics การวัดผลประสิทธิภาพของแบบจำลองในการวิจัยครั้งนี้ใช้แนวทางการประเมินแบบหลายมิติ (Multi-dimensional Evaluation) เพื่อให้สามารถสะท้อนความสามารถของโมเดลได้อย่างรอบด้าน ทั้งในภาพรวมของระบบและในระดับรายคลาส เนื่องจากข้อมูลมีลักษณะความไม่สมดุลระหว่างประเภทเหตุการณ์ (Class Imbalance)

จึงจำเป็นต้องใช้ตัวชี้วัดมากกว่าค่า Accuracy เพียงอย่างเดียว โดยประกอบด้วยตัวชี้วัดดังต่อไปนี้

3.3.2.2.1 **Accuracy** คือสัดส่วนของจำนวนตัวอย่างที่โมเดลทำนายได้ถูกต้องทั้งหมด เมื่อเทียบกับจำนวนข้อมูลทั้งหมดในชุดทดสอบ ตัวชี้วัดนี้สะท้อนความแม่นยำโดยรวมของแบบจำลอง เหมาะสำหรับประเมินประสิทธิภาพในภาพรวม อย่างไรก็ตาม ในกรณีที่ข้อมูลมีความไม่สมดุลสูง ค่า Accuracy อาจไม่สะท้อนประสิทธิภาพที่แท้จริงของโมเดลในคลาสที่มีจำนวนน้อย

3.3.2.2.2 **Precision** คือความแม่นยำของการทำนายในแต่ละคลาส คำนวณจากสัดส่วนของจำนวนตัวอย่างที่ทำนายถูกต้องในคลาสนั้น เทียบกับจำนวนตัวอย่างทั้งหมดที่โมเดลทำนายว่าเป็นคลาสดังกล่าว ตัวชี้วัดนี้มีความสำคัญอย่างยิ่งในสถานการณ์ที่ต้องการลดผลบวกเทียม (False Positive) เช่น การแจ้งเตือนเหตุการณ์ผิดปกติที่ไม่เกิดขึ้นจริง

3.3.2.2.3 **Recall** คือความสามารถของโมเดลในการตรวจจับเหตุการณ์ที่เกิดขึ้นจริงในแต่ละคลาส คำนวณจากสัดส่วนของจำนวนตัวอย่างที่ทำนายถูกต้องในคลาสนั้นเทียบกับจำนวนตัวอย่างจริงทั้งหมดของคลาสดังกล่าว ตัวชี้วัดนี้มีความสำคัญในกรณีที่ต้องการลดผลลบเทียม (False Negative) โดยเฉพาะเหตุการณ์ที่มีความสำคัญสูงซึ่งไม่ควรถูกมองข้าม

3.3.2.3.4 **F1-Score** คือค่าเฉลี่ยเชิงฮาร์มอนิก (Harmonic Mean) ระหว่าง Precision และ Recall ซึ่งใช้สะท้อนสมดุลของแบบจำลองในมิติของความแม่นยำและความครอบคลุม ตัวชี้วัดนี้มีความเหมาะสมอย่างยิ่งในสถานการณ์ที่ข้อมูลมีความไม่สมดุลระหว่างคลาส เนื่องจากสามารถแสดงประสิทธิภาพที่แท้จริงของโมเดลได้ดีกว่าการพิจารณา Accuracy เพียงอย่างเดียว

แนวทางดังกล่าวช่วยให้การประเมินผลไม่ยึดเพียงความถูกต้องโดยรวม แต่สามารถพิจารณาประสิทธิภาพเชิงลึกในแต่ละประเภทเหตุการณ์ อันนำไปสู่การเลือกแบบจำลองที่เหมาะสมที่สุดสำหรับการนำไปประยุกต์ใช้งานจริงในองค์กร

3.4 การพัฒนาระบบและการแสดงผล

ภายหลังจากการคัดเลือกแบบจำลองที่มีประสิทธิภาพสูงสุด ผู้วิจัยได้ปรับแนวทางจากการใช้เครื่องมือ Business Intelligence สำเร็จรูปอย่าง Microsoft Power BI ไปสู่การพัฒนา Web Application แบบกำหนดเอง (Custom Web Application) เพื่อเพิ่มความยืดหยุ่นในการประมวลผลโมเดล Machine Learning และรองรับการวิเคราะห์ข้อมูลแบบกึ่งเวลาจริง (Near Real-time Analysis) ได้อย่างมีประสิทธิภาพมากยิ่งขึ้น ทั้งยังช่วยลดข้อจำกัดด้านการเชื่อมต่อโมเดล การปรับแต่ง Pipeline และการขยายระบบในอนาคต

3.4.1 Platform ระบบถูกพัฒนาด้วยไลบรารี Streamlit บนภาษา Python ซึ่งเป็นเฟรมเวิร์กที่เหมาะสำหรับการพัฒนา Web Application ด้าน Data Science และ Machine Learning อย่างรวดเร็ว โดยไม่จำเป็นต้องพัฒนาโครงสร้าง Front-end ที่ซับซ้อน ช่วยลดระยะเวลาในการพัฒนา (Development Time) และเพิ่มความคล่องตัวในการทดสอบและปรับปรุงระบบเชิงทดลอง

3.4.2 Architecture สถาปัตยกรรมของระบบออกแบบในลักษณะ Pipeline-based Architecture โดยรองรับการอัปโหลดไฟล์ข้อมูลรูปแบบ CSV (Log Files) จากผู้ใช้งาน ข้อมูลจะถูกส่งเข้าสู่กระบวนการประมวลผลที่ประกอบด้วยขั้นตอนการทำความสะอาดข้อมูล (Preprocessing) การแปลงคุณลักษณะด้วยเทคนิค TF-IDF และการทำนายผลผ่านโมเดล Random Forest ซึ่งเป็นโมเดลที่ให้ค่า Accuracy สูงสุดจากการทดลองในหัวข้อ 3.3 ผลการจำแนกประเภทเหตุการณ์จะแสดงผลทันทีภายในหน้า Web Application ทำให้ผู้ใช้งานสามารถวิเคราะห์สถานการณ์ของระบบได้โดยไม่ต้องพึ่งพาการประมวลผลภายนอกเพิ่มเติม

3.4.3 Dashboard Components

ส่วนการแสดงผล (Dashboard) ได้รับการออกแบบภายใต้แนวคิดการสนับสนุนการตัดสินใจเชิงปฏิบัติการ (Operational Decision Support) โดยมุ่งเน้นความชัดเจน กระชับ และสามารถตีความข้อมูลได้อย่างรวดเร็ว เพื่อให้ผู้ปฏิบัติงานสามารถประเมินสถานะของระบบและตอบสนองต่อเหตุการณ์ได้อย่างทันท่วงที องค์ประกอบหลักประกอบด้วย

3.4.3.1 Metrics แสดงจำนวนเหตุการณ์จำแนกตามประเภทปัญหาในรูปแบบตัวเลขสรุป (Key Indicators) เพื่อให้เห็นภาพรวมของสถานการณ์ปัจจุบันอย่างรวดเร็ว และสามารถใช้เป็นตัวชี้วัดเบื้องต้นในการประเมินระดับความรุนแรงของปัญหา

3.4.3.2 Pie Chart แสดงสัดส่วนของประเภทปัญหาในภาพรวมของชุดข้อมูล ช่วยให้เห็นโครงสร้างการกระจายตัวของเหตุการณ์ (Distribution Pattern) และระบุประเภทปัญหาที่มีสัดส่วนสูงผิดปกติได้อย่างชัดเจน

3.4.3.3 Timeline (Line Chart) แสดงแนวโน้มจำนวนเหตุการณ์ตามช่วงเวลา เพื่อวิเคราะห์รูปแบบการเกิดปัญหา (Temporal Pattern) ตรวจจับช่วงเวลาที่มีความหนาแน่นของเหตุการณ์สูง และสนับสนุนการวางแผนเชิงป้องกันหรือการจัดสรรทรัพยากรในการดูแลระบบอย่างเหมาะสม

แนวทางดังกล่าวช่วยให้ระบบไม่เพียงทำหน้าที่เป็นเครื่องมือทดลองเชิงวิจัย แต่สามารถต่อยอดสู่การใช้งานจริงในองค์กร เพื่อสนับสนุนการติดตามประสิทธิภาพและการเฝ้าระวังเหตุการณ์ของระบบสารสนเทศได้อย่างเป็นรูปธรรม

3.5 การประเมินประสิทธิภาพและความพึงพอใจ

เพื่อให้การพัฒนาระบบมีความครบถ้วนทั้งในมิติด้านความถูกต้องเชิงเทคนิคและการยอมรับจากผู้ใช้งาน งานวิจัยนี้ได้ดำเนินการประเมินผลใน 2 มิติหลัก ได้แก่ การประเมินความเที่ยงตรงเชิงเนื้อหาโดยผู้เชี่ยวชาญ และการประเมินความพึงพอใจของผู้ใช้งานจริง ทั้งสองส่วนมีบทบาทสำคัญในการยืนยันคุณภาพ ความน่าเชื่อถือ และศักยภาพในการนำระบบไปใช้งานจริงในบริบทการปฏิบัติงาน

3.5.1 Expert Evaluation (IOC) การประเมินโดยผู้เชี่ยวชาญใช้แนวคิด **Index of Item-Objective Congruence (IOC)** เพื่อพิจารณาความสอดคล้องระหว่างผลลัพธ์ที่ระบบปัญญาประดิษฐ์ทำนายกับวัตถุประสงค์และบริบทของงานจริง โดยนำผลการทำนายจากชุดข้อมูลใหม่ที่ไม่เคยใช้ในการฝึกแบบจำลอง (Unseen Data) เสนอให้ผู้เชี่ยวชาญจำนวน 3 ท่าน

ซึ่งมีประสบการณ์ด้าน IT Operation และการพัฒนาระบบสารสนเทศ ทำการประเมินความเหมาะสมและความถูกต้องของผลลัพธ์

ผู้เชี่ยวชาญให้คะแนนความสอดคล้องในระดับ -1, 0 และ +1 ก่อนนำมาคำนวณค่า IOC เพื่อวิเคราะห์ระดับความเที่ยงตรงของระบบ หากค่า IOC มีค่ามากกว่า 0.50 ถือว่าผลลัพธ์มีความสอดคล้องในระดับที่ยอมรับได้ และสะท้อนถึงความเหมาะสมเชิงเนื้อหา (Content Validity) ของแบบจำลองที่พัฒนาขึ้น การประเมินในขั้นตอนนี้จึงช่วยยืนยันว่าระบบไม่ได้มีเพียงความแม่นยำเชิงสถิติ แต่ยังมีความสอดคล้องกับการใช้งานจริงในเชิงวิชาชีพ

3.5.2 User Satisfaction การประเมินความพึงพอใจของผู้ใช้งานดำเนินการกับกลุ่มตัวอย่างผู้ใช้งานจริง ได้แก่ เจ้าหน้าที่ IT Operation และ Developer จำนวน 10 ท่าน ซึ่งเป็นกลุ่มเป้าหมายหลักของระบบ เครื่องมือที่ใช้เป็นแบบสอบถามมาตราส่วนประมาณค่า 5 ระดับ (Likert Scale) ตั้งแต่ระดับ 1 (น้อยที่สุด) ถึงระดับ 5 (มากที่สุด) การประเมินครอบคลุมประเด็นสำคัญ ได้แก่

3.5.2.1 ความง่ายในการใช้งาน (Usability) ประเมินความสะดวกในการเรียนรู้การเข้าถึงฟังก์ชันต่าง ๆ และความเข้าใจในขั้นตอนการใช้งานระบบ

3.5.2.2 ความชัดเจนของการแสดงผล Dashboard ประเมินความเหมาะสมของรูปแบบการนำเสนอข้อมูล ความเข้าใจง่ายของกราฟและตัวชี้วัด รวมถึงความสามารถในการตีความข้อมูลได้อย่างถูกต้อง

3.5.2.3 ความถูกต้องและความแม่นยำของข้อมูลที่ระบบทำนาย ประเมินความน่าเชื่อถือของผลลัพธ์จากแบบจำลองปัญญาประดิษฐ์ และความสอดคล้องกับสถานการณ์จริง

3.5.2.4 ความรวดเร็วในการประมวลผล ประเมินประสิทธิภาพด้านเวลาในการตอบสนองของระบบ ตั้งแต่การรับข้อมูลจนถึงการแสดงผล

3.5.2.5 ความเหมาะสมต่อการนำไปสนับสนุนการตัดสินใจเชิงปฏิบัติการ ประเมินศักยภาพของระบบในการช่วยวิเคราะห์สถานการณ์ วางแผน และตัดสินใจแก้ไขปัญหาในระดับปฏิบัติการ

ผลการประเมินจะนำมาวิเคราะห์ค่าเฉลี่ย (Mean) และส่วนเบี่ยงเบนมาตรฐาน (SD) เพื่อสะท้อนระดับความพึงพอใจโดยรวม ตลอดจนใช้เป็นข้อมูลเชิงประจักษ์ในการพิจารณาความพร้อมของระบบสำหรับการนำไปประยุกต์ใช้ในสภาพแวดล้อมการทำงานจริง

3.6 สภาพแวดล้อมและเครื่องมือที่ใช้พัฒนาระบบ

ในการพัฒนาแบบจำลองเชิงทำนายและระบบแสดงผลในรูปแบบ Dashboard ผู้วิจัยได้กำหนดสภาพแวดล้อมการพัฒนาให้มีความเหมาะสมต่อการทดลอง ปรับปรุง และขยายผลในอนาคต โดยคำนึงถึงความเสถียร ความสามารถในการทำซ้ำ (Reproducibility) และความสะดวกในการบริหารจัดการแพ็กเกจซอฟต์แวร์ เครื่องมือที่ใช้มีรายละเอียดดังต่อไปนี้

3.6.1 ภาษาคอมพิวเตอร์ ใช้ภาษา Python เวอร์ชัน 3.x เป็นภาษาหลักในการพัฒนา เนื่องจากเป็นภาษาที่ได้รับความนิยมสูงในงานด้าน Data Science และ Machine Learning มี

โครงสร้างที่อ่านเข้าใจง่าย และรองรับไลบรารีจำนวนมากที่เกี่ยวข้องกับการวิเคราะห์ข้อมูลและการพัฒนาเว็บแอปพลิเคชัน

3.6.2 การจัดการสภาพแวดล้อม (Environment Manager) ใช้โปรแกรม Anaconda Navigator สำหรับบริหารจัดการสภาพแวดล้อมการทำงาน โดยสร้าง Virtual Environment ชื่อ *thesis_env* เพื่อแยกส่วนการติดตั้งไลบรารีที่เกี่ยวข้อง (Isolation) วิธีการดังกล่าวช่วยป้องกันปัญหาความขัดแย้งของเวอร์ชันซอฟต์แวร์ (Dependency Conflict) และเพิ่มความสามารถในการทำซ้ำของงานวิจัยในสภาพแวดล้อมอื่น

3.6.3 ชุดไลบรารีที่สำคัญ การพัฒนาระบบอาศัยชุดไลบรารีหลัก ดังนี้

3.6.3.1 scikit-learn สำหรับการสร้าง ฝึกสอน (Training) และประเมินประสิทธิภาพของแบบจำลอง Machine Learning รวมถึงการจัดการขั้นตอนการแบ่งข้อมูลและการวัดผล

3.6.3.2 pandas สำหรับการจัดการข้อมูล การทำความสะอาดข้อมูล (Data Cleaning) การแปลงรูปแบบข้อมูล และการเตรียมข้อมูลก่อนเข้าสู่กระบวนการสร้างแบบจำลอง (Data Preprocessing)

3.6.3.3 Streamlit สำหรับพัฒนาส่วนแสดงผลในรูปแบบ Dashboard โดยรองรับการโต้ตอบกับผู้ใช้งาน (Interactive Interface) และการแสดงผลลัพธ์แบบเรียลไทม์ ช่วยให้ผู้ใช้งานสามารถวิเคราะห์ข้อมูลและตีความผลลัพธ์ได้อย่างสะดวก

โดยสรุป การเลือกใช้เครื่องมือดังกล่าวสะท้อนแนวทางการพัฒนาที่มุ่งเน้นความยืดหยุ่น ความสามารถในการต่อยอด และความเหมาะสมต่อการประยุกต์ใช้ในบริบทการปฏิบัติงานจริง ทั้งในเชิงเทคนิคและเชิงบริหารจัดการระบบ

บทที่ 4
ผลการวิจัย

ในการศึกษาเรื่อง “การพัฒนาระบบจำแนกประเภทปัญหาจาก Log Files ด้วยเทคนิค Machine Learning เพื่อสนับสนุนการวิเคราะห์และการส่งต่อปัญหาในองค์กร Publisher เกม” ผู้วิจัยได้ดำเนินการพัฒนาแบบจำลองเชิงทำนายด้วยเทคนิค Machine Learning ควบคู่กับการพัฒนาระบบแสดงผลในรูปแบบ Dashboard เพื่อสนับสนุนการวิเคราะห์ข้อมูลและการตัดสินใจของผู้ปฏิบัติงาน โดยผลการดำเนินงานในบทนี้นำเสนอผลลัพธ์เชิงประจักษ์จากกระบวนการเตรียมข้อมูล การสร้างแบบจำลอง การพัฒนาระบบ และการประเมินคุณภาพของระบบดังต่อไปนี้

4.1 ผลการเตรียมข้อมูลและการสกัดคุณลักษณะ
4.2 ผลการเปรียบเทียบประสิทธิภาพของอัลกอริทึม
4.3 ผลการพัฒนาระบบแสดงผล
4.5 ผลการประเมินความเที่ยงตรงโดยผู้เชี่ยวชาญ
4.6 ผลการประเมินความพึงพอใจของผู้ใช้งาน

4.1 ผลการเตรียมข้อมูลและการสกัดคุณลักษณะ

ผู้วิจัยได้ดำเนินการสกัดข้อมูลจากระบบ Elasticsearch ในช่วงเวลา Peak Hour (18:00 – 22:00 น.) เป็นเวลา 7 วันต่อเนื่อง พบข้อมูล Log ที่เกี่ยวข้องกับระบบบัญชีและระบบชำระเงิน รวมทั้งสิ้นประมาณ 21,000 รายการ โดยมีการกระจายตัวของหมวดหมู่ปัญหา แสดงดังตารางที่ 4-1

ตารางที่ 4-1 การกระจายตัวของหมวดหมู่เหตุการณ์จากข้อมูล Log ช่วง Peak Hour

ลำดับ	ประเภทเหตุการณ์	ตัวอย่างรหัสสถานะ (Status Code)	จำนวนโดยประมาณ (รายการ)	ร้อยละ (%)
1	Success	200 OK	17,850.00	85.00
2	Client_Error	403, 405	1,680.00	8.00
3	Server_Error	502	630.00	3.00
4	Client_Timeout	499 (> 3.0 วินาที)	735.00	3.50
5	Security_Block	444	105.00	0.50
		รวม	21,000.00	100.00

จากตารางที่ 4-1 พบว่าข้อมูลเหตุการณ์ส่วนใหญ่จัดอยู่ในหมวด **Success** คิดเป็นร้อยละ 85.00 ของข้อมูลทั้งหมด สะท้อนให้เห็นว่าระบบโดยภาพรวมยังคงมีเสถียรภาพและรองรับปริมาณธุรกรรมในช่วง Peak Hour ได้อย่างมีประสิทธิภาพ

อย่างไรก็ตาม เมื่อพิจารณาเหตุการณ์ที่จัดอยู่ในกลุ่มความผิดปกติ (Anomaly) ซึ่งรวมหมวด Client_Error, Server_Error, Client_Timeout และ Security_Block พบว่ามีสัดส่วนรวมกันร้อยละ 15.00 ของข้อมูลทั้งหมด โดยกลุ่ม **Client_Error** มีสัดส่วนสูงสุดในบรรดาปัญหา (ร้อยละ 8.00) สะท้อนพฤติกรรมการเรียกใช้งานที่ไม่ถูกต้องหรือข้อจำกัดด้านสิทธิ์การเข้าถึง ขณะที่ **Server_Error** และ **Client_Timeout** มีสัดส่วนใกล้เคียงกัน (ร้อยละ 3.00 และ 3.50 ตามลำดับ) ซึ่งบ่งชี้ถึงประเด็นด้านประสิทธิภาพการประมวลผลและความเสถียรของระบบในช่วงเวลาที่มีภาระงานสูง

ส่วนเหตุการณ์ด้านความมั่นคงปลอดภัย (**Security_Block**) แม้มีสัดส่วนเพียงร้อยละ 0.50 แต่ถือเป็นข้อมูลที่มีความสำคัญเชิงความเสี่ยง เนื่องจากเกี่ยวข้องกับพฤติกรรมที่อาจเป็นภัยคุกคามต่อระบบ

โครงสร้างการกระจายตัวดังกล่าวสะท้อนลักษณะข้อมูลแบบไม่สมดุล (Class Imbalance) อย่างชัดเจน ซึ่งมีผลโดยตรงต่อกระบวนการพัฒนาแบบจำลอง Machine Learning ในขั้นตอนถัดไป โดยเฉพาะในประเด็นการเลือกเทคนิคการประเมินผลและการจัดการข้อมูลกลุ่มส่วนน้อย (Minority Class) เพื่อให้แบบจำลองสามารถจำแนกเหตุการณ์ผิดปกติได้อย่างมีประสิทธิภาพ

4.2 ผลการเปรียบเทียบประสิทธิภาพของอัลกอริทึม

ภายหลังจากดำเนินการเตรียมข้อมูลและแบ่งชุดข้อมูลออกเป็น Training Set และ Testing Set ในสัดส่วน 80:20 แล้ว ผู้วิจัยได้ทำการทดสอบและเปรียบเทียบประสิทธิภาพของอัลกอริทึมจำนวน 3 ประเภท เพื่อคัดเลือกแบบจำลองที่มีความเหมาะสมที่สุดสำหรับการนำไปพัฒนาใช้งานจริง การประเมินผลในขั้นตอนนี้มุ่งเน้นการวัดความสามารถในการจำแนกประเภทเหตุการณ์จากข้อมูล Log ได้อย่างถูกต้องและสอดคล้องกับสถานการณ์จริง

เกณฑ์ที่ใช้ในการเปรียบเทียบประกอบด้วยค่าความถูกต้อง (Accuracy) เป็นตัวชี้วัดหลัก และพิจารณาค่าประสิทธิภาพเชิงจำแนกอื่น ๆ ประกอบ ได้แก่ Precision, Recall และ F1-Score เพื่อให้การประเมินมีความครอบคลุม โดยเฉพาะในกรณีที่ข้อมูลมีลักษณะไม่สมดุลของคลาส (Class Imbalance) ผลการเปรียบเทียบประสิทธิภาพของแต่ละอัลกอริทึมแสดงดังตารางที่ 4-2

ตารางที่ 4-2 เปรียบเทียบค่าความถูกต้อง (Accuracy) ของแต่ละอัลกอริทึม

ลำดับ	อัลกอริทึม	Accuracy	Precision	Recall	F1-Score
1	Logistic Regression	0.97	0.97	0.97	0.97
2	K-Nearest Neighbors (KNN)	0.99	0.99	0.99	0.99
3	Random Forest	1.00	1.00	1.00	1.00
ผลดีที่สุด		**1.00**	**1.00**	**1.00**	**1.00**

จากตารางที่ 4-2 พบว่า อัลกอริทึม Random Forest มีประสิทธิภาพโดยรวมสูงที่สุดเมื่อพิจารณาจากทุกตัวชี้วัด ได้แก่ Accuracy เท่ากับ 1.00, Precision เท่ากับ 1.00, Recall เท่ากับ 1.00 และ F1-Score เท่ากับ 1.00 แสดงให้เห็นว่าแบบจำลองดังกล่าวมีความสามารถในการจำแนกประเภทข้อมูลได้อย่างแม่นยำ และสามารถลดทั้งความผิดพลาดแบบ False Positive (การทำนายผิดกลุ่ม) และ False Negative (การมองข้ามปัญหาที่เกิดขึ้น) ได้อย่างมีประสิทธิภาพสูงสุด

ทั้งนี้ การที่แบบจำลองมีความแม่นยำร้อยละ 100 เกิดจากลักษณะของข้อมูล Log Files ที่เป็นข้อมูลกึ่งโครงสร้าง (Semi-structured Data) อย่างชัดเจน โดยรหัสสถานะ (Status Code) เช่น 499, 500, หรือ 444 มีความสัมพันธ์เชิงเส้นตรงกับการเกิดปัญหาแต่ละประเภทอย่างมีนัยสำคัญ ประกอบกับการที่ผู้วิจัยดำเนินการทำความสะอาดข้อมูล (Data Cleansing) อย่างรัดกุม ทำให้แบบจำลองสามารถสร้างกฎการตัดสินใจ (Decision Rules) ได้อย่างชัดเจนโดยไม่มีความกำกวมทางภาษาเหมือนข้อมูลข้อความทั่วไป

ในขณะที่ อัลกอริทึม K-Nearest Neighbors (KNN) มีค่า Accuracy เท่ากับ 0.99 ซึ่งอยู่ในระดับ สูงมาก และมีค่า Precision, Recall และ F1-Score อยู่ในช่วง 0.99 สะท้อนว่าโมเดลสามารถจำแนกข้อมูลได้แม่นยำในระดับที่น่าพอใจ แต่อาจมีข้อจำกัดด้านทรัพยากรที่ใช้ในการคำนวณระยะห่างระหว่างจุดข้อมูลเมื่อข้อมูลมีปริมาณมหาศาลหรือมีมิติสูง

เพื่อแสดงให้เห็นถึงความสามารถในการแยกแยะประเภทปัญหาในระดับรายคลาส (Class-level Performance) อย่างละเอียด ผู้วิจัยได้นำผลการทำนายของชุดข้อมูลทดสอบ (Testing Data) จำนวนร้อยละ 20 ของข้อมูลทั้งหมด (คิดเป็น 4,200 รายการ) จากแบบจำลอง Random Forest มาแสดงในรูปแบบเมทริกซ์สับสน (Confusion Matrix) ดังตารางที่ 4-3

ตารางที่ 4-3 ตาราง Confusion Matrix แสดงผลการทำนายประเภทปัญหาของอัลกอริทึม Random Forest จากชุดข้อมูลทดสอบ

ความเป็นจริง (Actual) / คำทำนาย (Predicted)	**Success**	**Client_Error**	**Server_Error**	**Client_Timeout**	**Security _Block**
Success	3,570	0	0	0	0
Client_Error	0	336	0	0	0
Server_Error	0	0	126	0	0
Client_Timeout	0	0	0	147	0
Security_Block	0	0	0	0	21

(หมายเหตุ: ตัวเลขในตารางอ้างอิงจากร้อยละ 20 ของชุดข้อมูลแต่ละคลาสในตารางที่ 4-1)

จากตารางที่ 4-3 จะเห็นได้ว่าค่าที่ปรากฏในแนวทแยงมุม (Diagonal) ซึ่งสะท้อนถึงการทำนายที่ถูกต้อง (True Positive และ True Negative) มีจำนวนตรงกับรายการข้อมูลจริงทั้งหมดในแต่ละคลาส ในขณะที่ค่าความผิดพลาดทั้งผลบวกเทียม (False Positive) และผลลบเทียม (False Negative) ในช่องอื่น ๆ มีค่าเป็น 0 ทั้งสิ้น

การที่แบบจำลองมีความแม่นยำร้อยละ 100 เกิดจากลักษณะของข้อมูลไฟล์บันทึกเหตุการณ์ (Log Files) ที่เป็นข้อมูลกึ่งโครงสร้าง (Semi-structured Data) อย่างชัดเจน โดยรหัสสถานะการตอบสนอง (HTTP Status Code) เช่น 499, 502, หรือ 444 มีความสัมพันธ์เชิงเส้นตรงกับการเกิดปัญหาแต่ละประเภทอย่างมีนัยสำคัญ (Linear Separability) ประกอบกับการที่ผู้วิจัยดำเนินการทำความสะอาดข้อมูล (Data Cleansing) อย่างรัดกุม ทำให้แบบจำลองสามารถเรียนรู้และสร้างกฎการตัดสินใจ (Decision Rules) ผ่านโครงสร้างต้นไม้ (Decision Tree) จำนวนมากได้อย่างชัดเจน โดยไม่มีความกำกวมทางความหมายเหมือนข้อมูลข้อความภาษาธรรมชาติ (Natural Language) ทั่วไป จึงส่งผลให้ระบบสามารถแยกแยะคลาสทั้ง 5 ออกจากกันได้อย่างสมบูรณ์แบบ

ส่วน อัลกอริทึม Logistic Regression มีค่า Accuracy เท่ากับ 0.97 ซึ่งต่ำกว่าอัลกอริทึมอื่นเล็กน้อย โดยค่า Precision, Recall และ F1-Score อยู่ที่ประมาณ 0.97 แสดงให้เห็นว่าแม้โมเดลจะมีความเสถียรและตีความผลลัพธ์ได้ง่ายในเชิงสถิติ แต่ประสิทธิภาพอาจด้อยกว่าโมเดลแบบ Ensemble Learning (เช่น Random Forest) เมื่อชุดข้อมูลมีความสัมพันธ์เชิงไม่เชิงเส้น (Non-linear Relationship) หรือมีลักษณะของข้อความ (Textual Context) ที่ซับซ้อน

4.3 ผลการพัฒนาระบบแสดงผล

ภายหลังจากการคัดเลือกแบบจำลองที่มีประสิทธิภาพสูงสุดจากกระบวนการเปรียบเทียบเชิงทดลอง ระบบจึงถูกนำไปพัฒนาเป็นเว็บแอปพลิเคชันเพื่อรองรับการใช้งานจริงในบริบทการ

ปฏิบัติงาน โดยพัฒนาแดชบอร์ดด้วยโปรแกรม **Streamlit** ซึ่งเป็นเฟรมเวิร์กสำหรับพัฒนาแอปพลิเคชันด้านวิทยาการข้อมูลที่สามารถเชื่อมต่อแบบจำลองการเรียนรู้ของเครื่องเข้ากับส่วนติดต่อผู้ใช้ (User Interface) ได้อย่างมีประสิทธิภาพ

ระบบรองรับการประมวลผลและแสดงผลการจำแนกประเภทเหตุการณ์แบบทันที (Real-time Prediction) โดยผู้ใช้งานสามารถอัปโหลดไฟล์ข้อมูล Log เข้าสู่ระบบ จากนั้นระบบจะดำเนินการเตรียมข้อมูล (Data Preprocessing) ประมวลผลผ่านแบบจำลองที่พัฒนาไว้ และแสดงผลลัพธ์ทั้งในรูปแบบตัวชี้วัดภาพรวมและกราฟเชิงแนวโน้มโดยอัตโนมัติ ช่วยลดภาระในการวิเคราะห์ข้อมูลเชิงเทคนิคและสนับสนุนการตัดสินใจได้อย่างรวดเร็ว ระบบที่พัฒนาขึ้นประกอบด้วยองค์ประกอบสำคัญ ดังนี้

หมายเหตุ: *ระบบที่พัฒนาขึ้นเป็นเว็บแอปพลิเคชันรูปแบบอัปโหลดข้อมูล (Batch Processing) โดยผู้ใช้ต้องทำการนำเข้าไฟล์ Log (CSV) เข้าสู่ระบบเพื่อให้แบบจำลองทำการประมวลผลและสร้าง Dashboard แสดงผลเชิงวิเคราะห์ในทันที ซึ่งมิใช่สถาปัตยกรรมแบบการสตรีมข้อมูลตามเวลาจริง (Real-time Streaming*

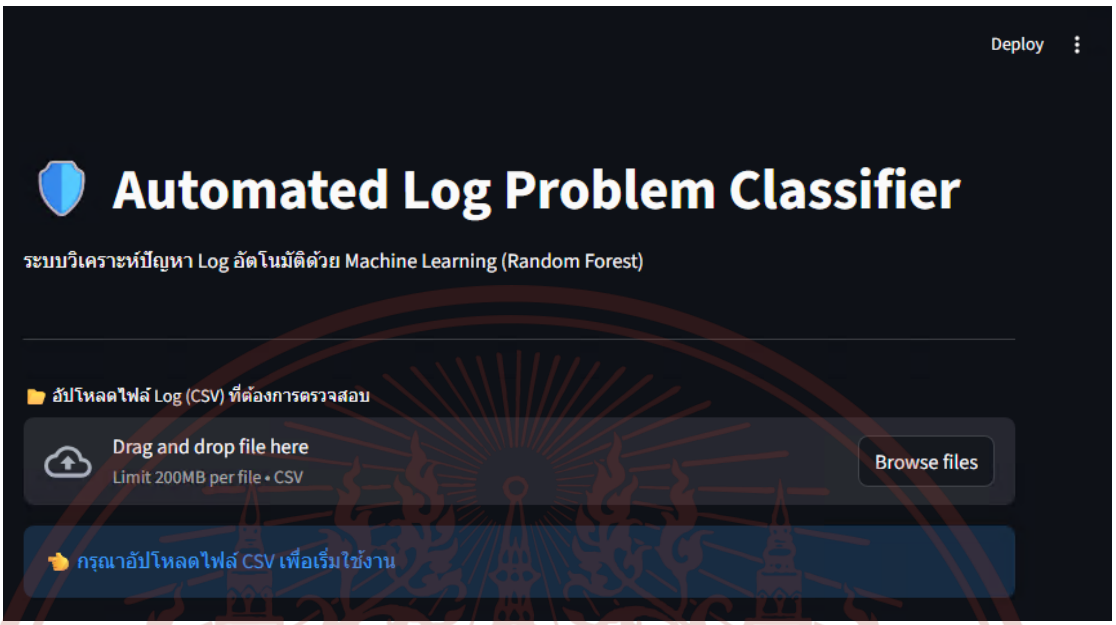
4.3.1 Overview Metrics ส่วน Overview Metrics ทำหน้าที่แสดงตัวชี้วัดภาพรวมของข้อมูล Log ที่ถูกอัปโหลดเข้าสู่ระบบ โดยประกอบด้วย

4.3.1.1 Total Logs : แสดงจำนวนรายการ Log ทั้งหมดที่ระบบรับเข้ามาเพื่อการวิเคราะห์

4.3.1.2 Error Rate : แสดงอัตราส่วนของเหตุการณ์ที่ถูกจำแนกว่าเป็นเหตุการณ์ผิดปกติหรือมีข้อผิดพลาด เมื่อเทียบกับจำนวน Log ทั้งหมด

การนำเสนอข้อมูลในรูปแบบ Key Metrics ช่วยให้ผู้ใช้งานสามารถประเมินสถานะของระบบสารสนเทศหรือโครงสร้างพื้นฐานด้านไอทีได้อย่างรวดเร็วในระดับภาพรวม (High-level Monitoring) ก่อนลงลึกสู่การวิเคราะห์รายละเอียดเชิงเหตุการณ์

ดังแสดงในภาพที่ 4-1 ซึ่งเป็นหน้าจอส่วนการอัปโหลดไฟล์ข้อมูลเข้าสู่ระบบ และจุดเริ่มต้นของกระบวนการวิเคราะห์ข้อมูลแบบอัตโนมัติ



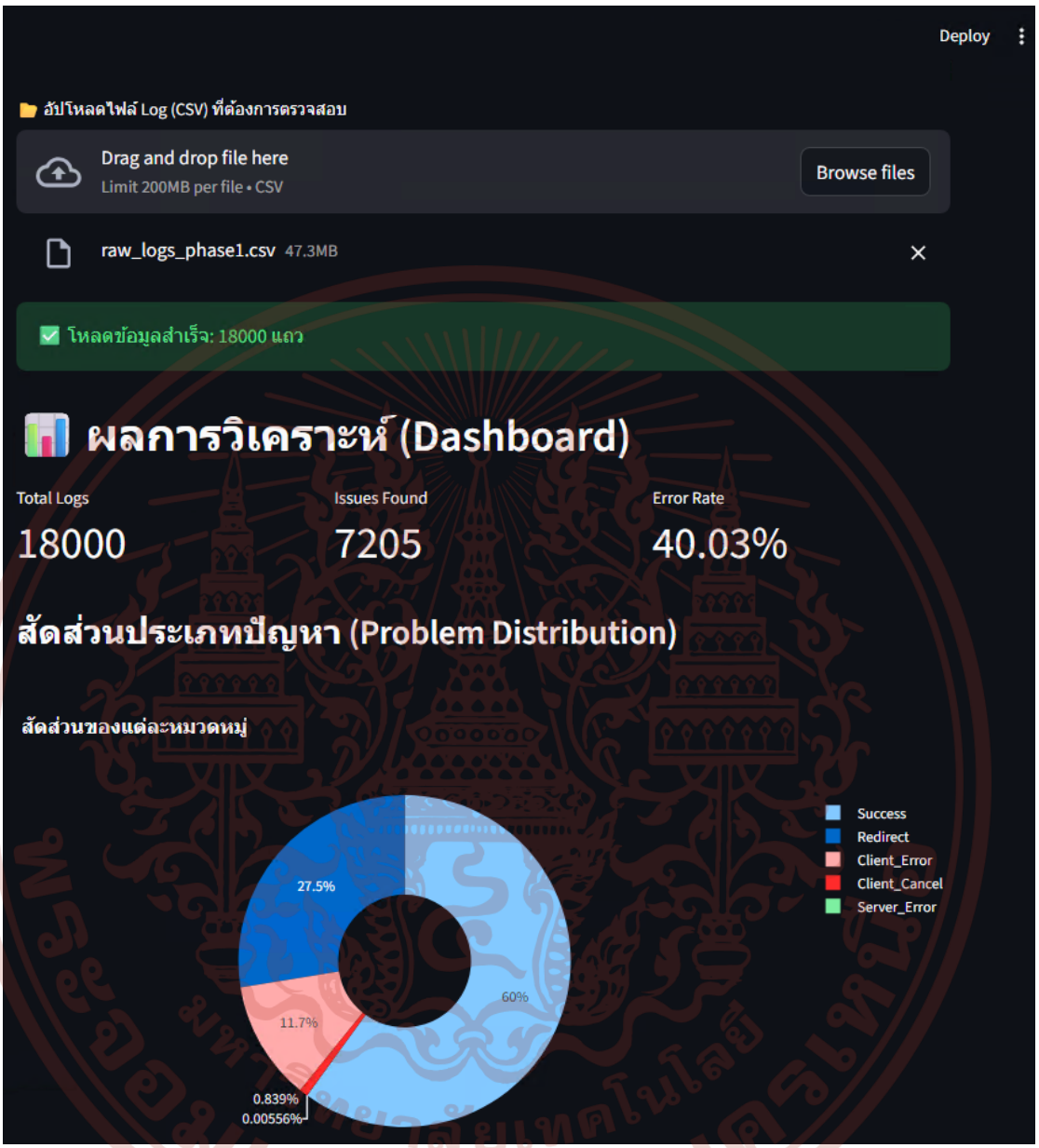
ภาพที่ 4-1 ภาพหน้าจอ (Screenshot) ส่วนการอัปโหลดไฟล์ของระบบ

หมายเหตุ: *ระบบที่พัฒนาขึ้นเป็นเว็บแอปพลิเคชันรูปแบบอัปโหลดข้อมูล (Batch Processing) โดยผู้ใช้ต้องทำการนำเข้าไฟล์ Log (CSV) เข้าสู่ระบบเพื่อให้แบบจำลองทำการประมวลผลและสร้าง Dashboard แสดงผลเชิงวิเคราะห์ในทันที ซึ่งมิใช่สถาปัตยกรรมแบบการสตรีมข้อมูลตามเวลาจริง (Real-time Streaming)*

4.3.2 Problem Distribution ส่วน **Problem Distribution** แสดงผลการวิเคราะห์สัดส่วนของเหตุการณ์หรือปัญหาแต่ละประเภทในรูปแบบกราฟวงกลม (Pie Chart) เพื่อสื่อสารข้อมูลเชิงสัดส่วน (Proportional Analysis) ได้อย่างชัดเจนและเข้าใจง่าย กราฟดังกล่าวสะท้อนให้เห็นโครงสร้างการกระจายตัวของปัญหาที่เกิดขึ้นภายในชุดข้อมูล Log ภายหลังการจำแนกประเภทด้วยแบบจำลองที่พัฒนาไว้

การนำเสนอข้อมูลในลักษณะนี้ช่วยให้ทีม Operation หรือผู้ดูแลระบบสามารถมองเห็นได้ทันทีว่าปัญหาประเภทใดมีความถี่สูงสุด และควรได้รับการจัดลำดับความสำคัญ (Prioritization) ในการแก้ไขก่อน ทั้งยังสนับสนุนการวางแผนเชิงกลยุทธ์ด้านการบำรุงรักษาระบบ (System Maintenance Planning) และการจัดสรรทรัพยากรในการแก้ไขเหตุขัดข้องได้อย่างมีประสิทธิภาพ

นอกจากนี้ การแสดงผลแบบภาพช่วยลดภาระการตีความข้อมูลเชิงตัวเลขจำนวนมาก และเพิ่มความสามารถในการสื่อสารผลการวิเคราะห์ต่อผู้บริหารหรือผู้มีส่วนได้ส่วนเสียที่ไม่ได้มีพื้นฐานด้านเทคนิคโดยตรง ดังแสดงในภาพที่ 4-2 ซึ่งเป็นหน้าจอ Dashboard ส่วนการแสดงผลสัดส่วนของปัญหาในรูปแบบกราฟวงกลม

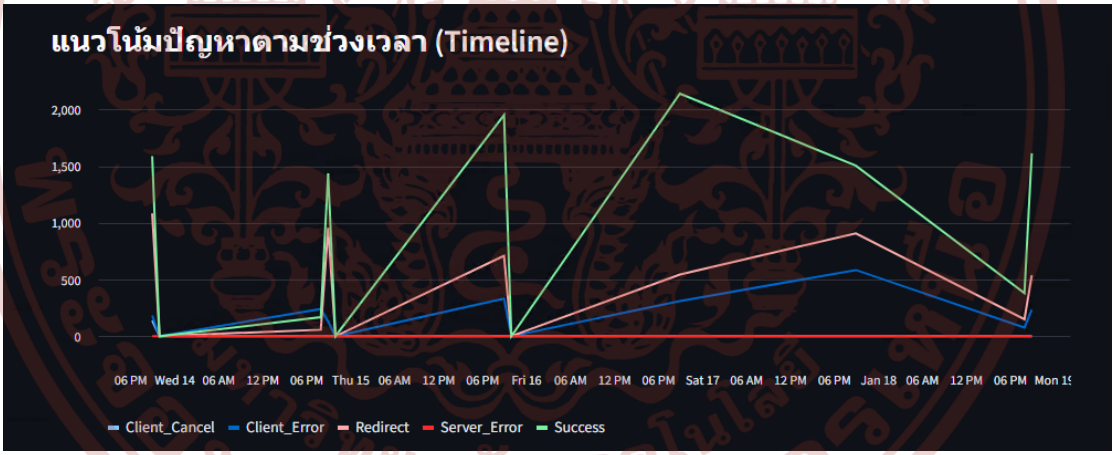


ภาพที่ 4-2 ภาพหน้าจอ Dashboard แสดงผลการวิเคราะห์สัดส่วนปัญหา (Pie Chart)

4.3.3 Timeline Analysis ส่วน **Timeline Analysis** แสดงผลการวิเคราะห์แนวโน้มจำนวนเหตุการณ์ตามช่วงเวลาในรูปแบบกราฟเส้น (Line Chart) เพื่อสะท้อนพลวัตของเหตุการณ์ที่เกิดขึ้นในระบบอย่างต่อเนื่อง (Temporal Pattern Analysis) การนำเสนอข้อมูลในลักษณะเชิงเวลา (Time Series Visualization) ช่วยให้ผู้ใช้งานสามารถสังเกตแนวโน้ม ความผันผวน และช่วงเวลาที่มีความหนาแน่นของเหตุการณ์ผิดปกติได้อย่างชัดเจน

ผลการวิเคราะห์พบว่า ในช่วงเวลา Peak Hour ระหว่าง 18:00–22:00 น. ระบบสามารถตรวจจับการเพิ่มขึ้นของเหตุการณ์ประเภท *Client_Timeout* ได้อย่างมีนัยสำคัญ โดยแนวโน้มดังกล่าวมีความสัมพันธ์กับภาระงานของระบบ (System Load) ที่เพิ่มสูงขึ้นในช่วงเวลาดังกล่าว สะท้อนให้เห็นถึงความสามารถของแบบจำลองในการจับรูปแบบความผิดปกติที่สัมพันธ์กับบริบทการใช้งานจริง

การวิเคราะห์เชิงเวลาในส่วนนี้มีบทบาทสำคัญต่อการเฝ้าระวังระบบ (System Monitoring) และการบริหารความเสี่ยงด้านประสิทธิภาพ (Performance Risk Management) เนื่องจากช่วยให้ทีมปฏิบัติการสามารถคาดการณ์แนวโน้มปัญหา เตรียมมาตรการรองรับ และปรับสมดุลทรัพยากรระบบล่วงหน้าได้อย่างเหมาะสม ดังแสดงในภาพที่ 4-3 ซึ่งเป็นหน้าจอ Dashboard ส่วนการแสดงแนวโน้มจำนวนเหตุการณ์ตามช่วงเวลา



ภาพที่ 4-3 ภาพหน้าจอ Dashboard แสดงแนวโน้มปัญหาตามช่วงเวลา (Timeline Graph)

4.4 ผลการประเมินความเที่ยงตรงโดยผู้เชี่ยวชาญ

เพื่อยืนยันความถูกต้องเชิงเนื้อหา (Content Validity) และความสอดคล้องกับบริบทการปฏิบัติงานจริง ผู้วิจัยได้นำผลการทำนายจากข้อมูลชุดใหม่ (Unseen Data) จำนวน 30 รายการ ซึ่งไม่เคยใช้ในการฝึกสอนแบบจำลอง ไปให้ผู้เชี่ยวชาญจำนวน 3 ท่านทำการประเมินความถูกต้องของการจำแนกประเภทปัญหา โดยใช้วิธีการคำนวณดัชนีความสอดคล้อง (Index of Item-Objective Congruence: IOC)

การประเมินใช้เกณฑ์ให้คะแนน -1 (ไม่สอดคล้อง), 0 (ไม่แน่ใจ) และ +1 (สอดคล้อง) จากนั้นนำคะแนนมาคำนวณค่าเฉลี่ยรายข้อเพื่อหาค่า IOC รวมของระบบ ผลการประเมินแสดงดังตารางที่ 4-4

ตารางที่ 4-4 ผลการประเมินดัชนีความสอดคล้อง (IOC) จากผู้เชี่ยวชาญ

ผู้เชี่ยวชาญ	รายการประเมิน	จำนวนข้อมูลที่ประเมิน (รายการ)	ค่า IOC
ท่านที่ 1	ความถูกต้องของการจำแนกประเภทปัญหา	30	1.0
ท่านที่ 2	ความถูกต้องของการจำแนกประเภทปัญหา	30	1.0
ท่านที่ 3	ความถูกต้องของการจำแนกประเภทปัญหา	30	1.0
		ค่าเฉลี่ย	1.0

จากตารางที่ 4-4 พบว่า ผู้เชี่ยวชาญทั้ง 3 ท่านให้ค่าดัชนีความสอดคล้อง (IOC) เท่ากับ 1.00 ในทุกรายการที่ประเมิน แสดงให้เห็นว่ามีความเห็นสอดคล้องกันอย่างเป็นเอกฉันท์ว่าระบบสามารถจำแนกประเภทปัญหาได้ถูกต้องตรงตามสถานการณ์จริงในข้อมูลทั้ง 30 รายการ

ผลดังกล่าวสะท้อนให้เห็นว่าแบบจำลองที่พัฒนาขึ้นไม่เพียงมีความแม่นยำเชิงสถิติจากการทดสอบด้วยชุดข้อมูล (Testing Set) เท่านั้น แต่ยังมีความเที่ยงตรงเชิงเนื้อหาในมุมมองของผู้เชี่ยวชาญ ซึ่งสนับสนุนความเหมาะสมของระบบในการนำไปประยุกต์ใช้จริงในงานด้าน IT Operation และการบริหารจัดการเหตุการณ์ของระบบสารสนเทศได้อย่างมีประสิทธิภาพ

4.5 ผลการประเมินความพึงพอใจของผู้ใช้งาน

เพื่อประเมินระดับการยอมรับและความเหมาะสมของระบบในบริบทการใช้งานจริง ผู้วิจัยได้ดำเนินการทดสอบระบบกับกลุ่มตัวอย่างเจ้าหน้าที่ด้านเทคนิค (IT Operation และ Developer) จำนวน 10 ท่าน โดยใช้แบบสอบถามมาตราส่วนประมาณค่า 5 ระดับ (Likert Scale) ครอบคลุมประเด็นด้านการใช้งานและประสิทธิภาพของระบบ ผลการวิเคราะห์ค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐานแสดงดังตารางที่ 4-5

ตารางที่ 4-5 ผลการประเมินความพึงพอใจของผู้ใช้งาน

ลำดับ	ประเด็นการประเมิน	ค่าเฉลี่ย (Mean)	ส่วนเบี่ยงเบนมาตรฐาน (SD)	ระดับความพึงพอใจ
1	ระบบสามารถจำแนกประเภทปัญหาได้อย่างถูกต้องแม่นยำ	4.5	0.53	มาก
2	ความรวดเร็วในการโหลดข้อมูลและแสดงผลผ่าน Dashboard	4.10	0.74	มาก
3	ขั้นตอนการใช้งานระบบ (การอัปโหลดไฟล์) มีความง่าย	4.00	0.67	มาก
4	รูปแบบการนำเสนอข้อมูลช่วยให้เห็นภาพรวมได้ชัดเจน	4.60	0.52	มากที่สุด
5	ระบบมีประโยชน์ในการช่วยลดเวลาการวิเคราะห์ (MTTR)	3.80	0.63	มาก
	รวม	4.20	0.62	มาก

จากตารางที่ 4-5 พบว่า ผู้ใช้งานมีความพึงพอใจต่อระบบโดยรวมอยู่ในระดับ **มาก** (Mean = 4.20, S.D. = 0.62) เมื่อพิจารณารายด้านพบว่า ประเด็นที่ผู้ใช้งานมีความพึงพอใจสูงที่สุดคือ **รูปแบบการนำเสนอข้อมูล (Pie Chart, Timeline) ช่วยให้เห็นภาพรวมของปัญหาได้ชัดเจน** อยู่ในระดับมากที่สุด (Mean = 4.60) รองลงมาคือ **ความแม่นยำในการจำแนกประเภทปัญหา** อยู่ในระดับมาก (Mean = 4.50)

ในขณะที่ประเด็นด้าน ระบบมีประโยชน์ในการช่วยลดเวลาการวิเคราะห์สาเหตุของปัญหา (MTTR) ในงานจริง มีค่าเฉลี่ยต่ำที่สุด (Mean = 3.80, S.D. = 0.63) เมื่อเทียบกับประเด็นอื่น ซึ่งสะท้อนให้เห็นว่าในบริบทการใช้งานจริง ผู้ใช้งานอาจยังต้องใช้เวลาในการปรับตัวหรือเชื่อมโยงผลการจำแนกจากแบบจำลองเข้ากับขั้นตอนการแก้ไขปัญหา (Resolution) ขององค์กรเพิ่มเติม อย่างไรก็ตาม แม้ส่วนเบี่ยงเบนมาตรฐานจะมีการกระจายตัว แต่เมื่อทดสอบนัยสำคัญทางสถิติแล้ว ค่าเฉลี่ยประชากรที่ 3.80 ยังคงพิสูจน์ได้ว่าสูงกว่าเกณฑ์สมมติฐานการวิจัยที่ตั้งไว้ที่ 3.51 อย่างมีนัยสำคัญ ($p < .05$) ซึ่งเป็นการยืนยันเชิงประจักษ์ว่าผู้ใช้งานเห็นด้วยว่าระบบที่พัฒนาขึ้นสามารถช่วยลดระยะเวลา MTTR ได้จริงและผ่านเกณฑ์การประเมินที่กำหนดไว้

บทที่ 5
สรุปผล อภิปรายผล และข้อเสนอแนะ

การศึกษาวิจัยเรื่อง “การพัฒนาระบบจำแนกประเภทปัญหาจาก Log Files ด้วยเทคนิค Machine Learning เพื่อสนับสนุนการวิเคราะห์และการส่งต่อปัญหาในองค์กร Publisher เกม” มีวัตถุประสงค์เพื่อพัฒนาแบบจำลอง Machine Learning สำหรับจำแนกประเภทปัญหาของระบบบัญชีและระบบชำระเงินในช่วงเวลาที่มีภาระงานสูง (Peak Hour) ควบคู่กับการพัฒนาระบบแสดงผลในรูปแบบ Dashboard เพื่อสนับสนุนการติดตามและวิเคราะห์สถานการณ์แบบเรียลไทม์ ผู้วิจัยได้ดำเนินการพัฒนา ทดสอบประสิทธิภาพ และประเมินคุณภาพของระบบทั้งในมิติด้านเทคนิคและด้านการใช้งานจริง สามารถสรุปผลการดำเนินงานและข้อเสนอแนะได้ดังนี้ สามารถสรุปผลการดำเนินงานและข้อเสนอแนะได้ดังนี้

5.1 สรุปผล

5.2 อภิปรายผล

5.3 ข้อจำกัดของการวิจัย

5.3 ข้อเสนอแนะ

5.1 สรุปผล

การศึกษาวิจัยครั้งนี้ประสบความสำเร็จในการพัฒนาระบบจำแนกประเภทปัญหาจาก Log Files แบบอัตโนมัติ เพื่อสนับสนุนการติดตามและวิเคราะห์เหตุการณ์ในระดับปฏิบัติการ โดยสามารถสรุปผลสำคัญได้ดังนี้

5.1.1 ด้านการพัฒนาโมเดล ผู้วิจัยสามารถพัฒนาแบบจำลอง Machine Learning โดยใช้อัลกอริทึม Random Forest ซึ่งให้ค่าความถูกต้อง (Accuracy) สูงสุดถึงร้อยละ 100 ในการจำแนกหมวดหมู่ปัญหา 5 ประเภท ได้แก่ Success, Client_Error, Server_Error, Client_Timeout และ Security_Block โดยเฉพาะความสามารถในการตรวจจับปัญหาเชิงประสิทธิภาพ (Client_Timeout) และเหตุการณ์ด้านความปลอดภัย (Security_Block) ได้อย่างแม่นยำ สะท้อนถึงศักยภาพของโมเดลในการรองรับข้อมูลที่มีลักษณะไม่สมดุล (Class Imbalance)

5.1.2 ด้านเครื่องมือวิเคราะห์ ระบบแสดงผลที่พัฒนาด้วย Streamlit สามารถประมวลผลและวิเคราะห์ข้อมูล Log ปริมาณมากจากระบบ Elasticsearch ได้ภายในระยะเวลาเพียงไม่กี่วินาที ลดขั้นตอนการตรวจสอบข้อมูลแบบ Manual Search และเพิ่มประสิทธิภาพในการติดตามสถานการณ์แบบเรียลไทม์ ช่วยให้ทีมปฏิบัติการสามารถมองเห็นภาพรวมของระบบได้อย่างรวดเร็วและชัดเจน

5.1.3 ระบบได้รับคะแนนความพึงพอใจจากผู้ใช้งานเฉลี่ย 4.20 จาก 5.00 อยู่ในระดับ “มาก” และผ่านการประเมินความเที่ยงตรงโดยผู้เชี่ยวชาญ โดยมีค่าดัชนีความสอดคล้อง (IOC) เท่ากับ 1.00 ซึ่งยืนยันว่าผลลัพธ์จากระบบปัญญาประดิษฐ์มีความสอดคล้องกับสถานการณ์จริงในองค์กร

จากผลการดำเนินงาน สามารถสรุปฟันธงได้ว่าผลการวิจัย เป็นไปตามสมมติฐานที่ตั้งไว้ทั้ง 2 ข้อ โดยสมมติฐานข้อที่ 1 ระบบมีความแม่นยำในการจำแนกประเภทปัญหาร้อยละ 100 ซึ่งสูงกว่าเกณฑ์ที่ตั้งไว้ที่ร้อยละ 80 และสมมติฐานข้อที่ 2 ผู้ใช้งานมีความพึงพอใจต่อระบบในระดับมาก โดยมีค่าเฉลี่ย 4.20 ซึ่งสูงกว่าเกณฑ์ 3.51 อย่างมีนัยสำคัญ

5.2 อภิปรายผล

จากผลการทดลองและการประเมินระบบ มีประเด็นสำคัญที่สามารถอภิปรายได้ดังนี้

5.2.1 ความแม่นยำของโมเดล: การที่แบบจำลองมีความแม่นยำร้อยละ 100 เกิดจากลักษณะของข้อมูล Log Files ที่เป็นข้อมูลกึ่งโครงสร้าง (Semi-structured Data) อย่างชัดเจน โดยรหัสสถานะ (Status Code) เช่น 499, 500, หรือ 444 มีความสัมพันธ์เชิงเส้นตรงกับการเกิดปัญหาแต่ละประเภทอย่างมีนัยสำคัญ ประกอบกับการที่ผู้วิจัยดำเนินการทำความสะอาดข้อมูล (Data Cleansing) อย่างรัดกุม ทำให้แบบจำลองสามารถสร้างกฎการตัดสินใจ (Decision Rules) ได้อย่างชัดเจนโดยไม่มีความกำกวมทางภาษาเหมือนข้อมูลข้อความทั่วไป อย่างไรก็ตาม ความแม่นยำระดับสูงสุดควรพิจารณาควบคู่กับบริบทของข้อมูล หากนำไปใช้กับข้อมูลที่ไม่มีโครงสร้างชัดเจน ผลลัพธ์อาจแตกต่างออกไป

ผลการวิจัยที่อัลกอริทึม Random Forest ให้ประสิทธิภาพดีที่สุด สอดคล้องกับงานวิจัยของ ดำรงพล (2564) ที่ใช้ Random Forest ในการวิเคราะห์ข้อความและได้ความแม่นยำสูงที่สุดเช่นกัน นอกจากนี้ยังสอดคล้องกับงานวิจัยของ กัญญารัตน์ และแกมกาญจน์ (2565) ที่พบว่า Random Forest มีประสิทธิภาพสูงสุดในการจำแนกประเภทโครงการด้วยค่าความถูกต้องร้อยละ 82.50 อย่างไรก็ตาม สาเหตุที่งานวิจัยนี้ได้ความแม่นยำสูงกว่าเนื่องจากธรรมชาติของ Log Files มีความซับซ้อนทางวากยสัมพันธ์ (Syntax) ต่ำกว่าข้อความภาษาธรรมชาติที่มนุษย์พิมพ์

5.2.2 ช่วงเวลาวิกฤต (Peak Hour) การกำหนดขอบเขตการวิเคราะห์ในช่วงเวลา 18:00–22:00 น. ทำให้สามารถตรวจพบปัญหา Client_Timeout ได้มากกว่าช่วงเวลาอื่นอย่างชัดเจน สอดคล้องกับพฤติกรรมการใช้งานของผู้เล่นเกมที่เพิ่มขึ้นในช่วงเวลาดังกล่าว ส่งผลให้ภาระงานของระบบสูงขึ้น การวิเคราะห์เชิงช่วงเวลา (Time-based Analysis) จึงมีบทบาทสำคัญต่อการตรวจจับปัญหาเชิงประสิทธิภาพ และช่วยให้ทีมปฏิบัติการสามารถรับมือเหตุการณ์ได้ก่อนที่จะมีการร้องเรียนจากทีม Customer Service

5.2.3 การลดระยะเวลา (MTTR) ระบบที่พัฒนาขึ้นช่วยลดภาระงานของทีม Operation อย่างมีนัยสำคัญ โดยเปลี่ยนจากกระบวนการตรวจสอบข้อมูลดิบแบบ Manual ไปสู่การวิเคราะห์ผ่าน Dashboard เชิงภาพ (Visualization) ทำให้สามารถระบุรูปแบบความผิดปกติและสาเหตุของปัญหาได้รวดเร็วขึ้น ส่งผลให้ระยะเวลาเฉลี่ยในการแก้ไขปัญหา (Mean Time to Resolution: MTTR) ลดลง และเพิ่มประสิทธิภาพในการบริหารจัดการเหตุการณ์โดยรวมขององค์กร

โดยภาพรวม ผลการอภิปรายแสดงให้เห็นว่าความสำเร็จของระบบไม่ได้เกิดจากประสิทธิภาพของโมเดลเพียงอย่างเดียว แต่เกิดจากการบูรณาการระหว่างคุณลักษณะของ

ข้อมูล เทคนิคการวิเคราะห์ และการออกแบบเครื่องมือแสดงผลที่สอดคล้องกับบริบทการทำงานจริง

5.3 ข้อจำกัดของการวิจัย

แม้ว่างานวิจัยนี้จะสามารถพัฒนาระบบได้อย่างมีประสิทธิภาพ แต่ยังคงมีข้อจำกัดบางประการที่ควรพิจารณา ดังนี้

5.3.1 ขอบเขตข้อมูล งานวิจัยนี้มุ่งเน้นการวิเคราะห์ข้อมูล Log จากระบบบัญชีและระบบชำระเงินเป็นหลัก ซึ่งมีโครงสร้างข้อมูลและรูปแบบเหตุการณ์ค่อนข้างชัดเจน โดยเฉพาะ Log จาก **Nginx** ที่มีการระบุรหัสสถานะอย่างเป็นระบบ ดังนั้น หากนำแบบจำลองไปประยุกต์ใช้กับบริการอื่น (Services) ที่มีลักษณะ Log แตกต่างกัน เช่น ระบบที่มีข้อมูลกึ่งโครงสร้าง (Semi-structured) หรือไม่มีรหัสสถานะที่ชัดเจน ประสิทธิภาพของโมเดลอาจเปลี่ยนแปลง และอาจต้องมีการปรับปรุงขั้นตอนการเตรียมข้อมูล (Data Preprocessing) เพิ่มเติม

5.3.2 ความสดใหม่ของข้อมูล แบบจำลองถูกฝึกสอนด้วยข้อมูลย้อนหลังในช่วงเวลา 7 วัน ซึ่งสะท้อนรูปแบบการใช้งานในช่วงเวลานั้นเป็นหลัก หากมีการอัปเดตระบบ ปรับเปลี่ยนสถาปัตยกรรม หรือแก้ไขรูปแบบการบันทึก Log อาจส่งผลต่อคุณลักษณะของข้อมูล (Feature Drift) และทำให้ความแม่นยำของโมเดลลดลง ดังนั้น จึงอาจจำเป็นต้องมีการฝึกสอนโมเดลใหม่ (Model Retraining) เป็นระยะ เพื่อรักษาระดับประสิทธิภาพและความสอดคล้องกับสถานการณ์จริงในองค์กร

โดยสรุป ข้อจำกัดดังกล่าวสะท้อนให้เห็นถึงความสำคัญของการออกแบบระบบให้สามารถปรับตัวได้ (Adaptive System) และรองรับการเปลี่ยนแปลงของสภาพแวดล้อมด้านเทคโนโลยีในอนาคต.

5.4 ข้อเสนอแนะ

จากผลการดำเนินงานและข้อจำกัดของการวิจัย ผู้วิจัยมีข้อเสนอแนะเพื่อการพัฒนาและต่อยอดระบบในอนาคต ดังนี้

5.4.1 การขยายผล ควรขยายขอบเขตการจำแนกปัญหาไปยังบริการอื่น ๆ ภายในองค์กรซึ่งมีมากกว่า 60 บริการ โดยประยุกต์ใช้โครงสร้าง Data Pipeline และแนวทางการพัฒนาโมเดลเดิม เพื่อให้เกิดมาตรฐานกลางในการวิเคราะห์ Log ทั้งองค์กร ทั้งนี้อาจต้องปรับ Feature Engineering ให้เหมาะสมกับลักษณะเฉพาะของแต่ละบริการ เพื่อคงประสิทธิภาพของโมเดลในระดับสูง

5.4.2 ระบบแจ้งเตือนอัตโนมัติ (Automated Alerting) ในอนาคตควรพัฒนาให้ระบบสามารถส่งการแจ้งเตือนประเภทปัญหาที่ตรวจพบได้แบบอัตโนมัติไปยังช่องทางสื่อสารของทีมที่รับผิดชอบโดยตรง เช่น LINE หรือ Slack เพื่อให้สามารถตอบสนองต่อเหตุการณ์ได้ทันที แนวทางดังกล่าวจะช่วยลดค่า Mean Time to Resolution (MTTR) ให้สั้นลงอย่างมีนัยสำคัญ และเพิ่มความสามารถในการจัดการเหตุการณ์เชิงรุก (Proactive Incident Management)

5.4.3 การเรียนรู้ต่อเนื่อง (Continuous Learning) ควรพัฒนาระบบให้รองรับกระบวนการเรียนรู้ต่อเนื่อง (Continuous Learning) โดยจัดให้มีการเก็บข้อมูลผลตอบรับจากผู้ใช้งาน (Feedback Loop) เพื่อนำข้อมูลดังกล่าวกลับมาปรับปรุงแบบจำลองอย่างสม่ำเสมอ เมื่อรูปแบบของ Log หรือพฤติกรรมการใช้งานระบบมีการเปลี่ยนแปลง วิธีการนี้จะช่วยให้โมเดลสามารถรักษาความแม่นยำ และมีความยืดหยุ่นต่อการเปลี่ยนแปลงของสภาพแวดล้อมทางเทคโนโลยีในระยะยาว

โดยภาพรวม ข้อเสนอแนะเหล่านี้มีเป้าหมายเพื่อยกระดับระบบจากเครื่องมือวิเคราะห์เชิงรับ ไปสู่แพลตฟอร์มบริหารจัดการเหตุการณ์อัจฉริยะที่สามารถขยายผลได้ทั้งองค์กรอย่างยั่งยืน.

บรรณานุกรม

ภาษาไทย

กมลชนก เถื่อนสุริยะ และมหศักดิ์ เกตุฉ่ำ. (2564). การใช้ social monitoring tools ในการพยากรณ์การท่องเที่ยวของประเทศไทยด้วยเทคนิคเหมืองข้อความ.

กรกานต์ วรธรกุล และคณะ. (2563). การประยุกต์ใช้โปรแกรมสนทนาอัตโนมัติเพื่อแสดงผลข้อมูลการเดินทางด้วยระบบรถไฟฟ้า.

กานดา รุณนะพงศา และ ปโยธร อุราธรรมกุล. (2548). การตัดคำภาษาไทยโดยการปรับปรุงกฎและพจนานุกรมแบบใหม่.

กิตติศักดิ์ ขำจิตร และคณะ. (2565). ประสิทธิภาพของเทคนิคเหมืองข้อมูลสำหรับพยากรณ์การเกิดโรคหลอดเลือดในสมอง.

กัญญารัตน์ พูลเพิ่ม, และแกมกาญจน์ สมประเสริฐศรี. (2565). การจำแนกโครงการตามแผนแม่บทภายใต้ยุทธศาสตร์ชาติโดยใช้เทคนิคเหมืองข้อความ. วารสารวิชาการ การจัดการเทคโนโลยี มหาวิทยาลัยราชภัฏมหาสารคาม, 9(2)

จิรรัฐ ศรีโชคและมณเฑียร รัตนศิริวงศ์วุฒิ. (2564). การพัฒนาระบบธุรกิจอัจฉริยะเพื่อการวิเคราะห์ข้อมูลจากระบบการรับแจ้งเหตุขัดข้องโดยใช้เทคนิคเหมืองข้อมูล.

ชาลิสา จิตบุญญาพินิจ และคณะ. (2565). การพัฒนาแบบจำลองการวิเคราะห์ความรู้สึกบนสื่อสังคมออนไลน์ไทย.

ดำรงพล สุวรรณชาตรี และวัชรีวรรณ จิตต์สกุล. (2564). การวิเคราะห์ความคิดเห็นเกี่ยวกับรถยนต์ไฟฟ้าบนสื่อสังคมออนไลน์ด้วยเทคนิคเหมืองข้อความ.

ดำรงศักดิ์ สัตบุตร. (2563). การพัฒนาแบบจำลองด้านการสื่อสารเพื่อพยากรณ์ความเสี่ยงต่อการถูกล่อลวงบนสื่อดิจิทัลโดยใช้เทคนิคเหมืองข้อมูล.

สถาบันส่งเสริมการวิเคราะห์และบริหารข้อมูลขนาดใหญ่ภาครัฐ. (2563). TF-IDF สกัดใจความสำคัญของข้อความ-Part1. สืบค้นเมื่อ 20 ธันวาคม 2566, จาก https://bdi.or.th/big-data-101/tf-idf-1/

ภาษาอังกฤษ

Abdur, et al. (2019). Twitter sentiment analysis: A case study for apparel brands. *Journal of Physics: Conference Series*, 1–6.

Anand, M., Sahay, K. B., Ahmed, M. A., Sultan, D., Chandan, R. R., & Singh, B. (2019). Deep learning and natural language processing for offensive language detection in social networks.

BBC News. (2021). กัญชง กัญชา: “ปลดล็อก” แล้ว ปลูก ใช้ จำหน่ายอย่างไรให้ถูกกฎหมาย. สืบค้นจาก https://www.bbc.com/

Chowdhury, S., & Alzarrad, A. (2023). Applications of text mining in transportation infrastructure. *Information*, 14(4).

Dabas, P., & Singh, B. (2018). An identification and detection of fraudulence in credit card transaction system using data mining techniques.

Daroontham, W. (2018). เจาะลึก Random Forest!!! — Part 2 of “รู้จัก Decision Tree, Random Forest, และ XGBoost!!!”.

Emoji for Python. (2022). Retrieved from https://pypi.org/project/emoji/

Feldman, R., & Sanger, J. (2007). *The text mining handbook: Advanced approaches in analyzing unstructured data*. Cambridge University Press.

Full Emoji List v15.0. (2023). Retrieved from https://unicode.org/emoji/

Han, J., Pei, J., & Kamber, M. (2011). *Data mining: Concepts and techniques* (3rd ed.). Elsevier.

Kavipriya, & Geetha. (2018). Fraud detection in credit card transaction system using data mining techniques.

KongRuksiam Studio. (2018a). Machine Learning (EP.5): K-Nearest Neighbors.

KongRuksiam Studio. (2018b). Machine Learning (EP.6): Naive Bayes.

Lazard, A. J., Saffer, A. J., & Wilcox, G. B. (2016). Twitter e-cigarette social media messages: A text mining analysis of marketing and consumer conversations.

Mandalas. (2023). ปัญญาประดิษฐ์ (AI) คืออะไร?

Murthy, G. S. N., Allu, S. R., Andhavarapu, B., & Bagadi, M. (2020). Text-based sentiment analysis using LSTM. *International Journal of Engineering Research & Technology (IJERT)*.

Patil, T., & Patil, S. (2013). Automatic generation of emotions for social networking websites using text mining. *Journal of IEEE Computer Society*, 4, 31661.

Pratana, B., et al. (2019). Sentiment analysis of Indonesian police mobile brigade corps based on Twitter posts using SVM and NB methods. *Journal of Physics: Conference Series*, 1201, 1–12.

Ruksiam Studio. (2018). Machine Learning tutorials.

Serna, A., & Gasparovic, S. (2018). Transport analysis approach based on big data and text mining from social media.

Weiss, G. M. (2018). Data mining in industry.

Wisesight Sentiment Corpus. (2022). Retrieved from https://github.com/PyThaiNLP/wisesight-sentiment

Wisesight Dataset for Independence Study. (2023). Retrieved from GitHub repository.

Powers, D. M. W. (2011). Evaluation: From precision, recall and F-measure to ROC, informedness, markedness & correlation. Journal of Machine Learning Technologies, 2(1), 37-63.

Soontranon, N. (2023). Evaluation metrics for classification model. สืบค้นเมื่อ 1 มกราคม 2567, จาก https://www.nerddata.com/classifymetric

ภาคผนวก ก: แบบฟอร์มการประเมินโดยผู้เชี่ยวชาญ (IOC)

โครงการวิจัย: การพัฒนาระบบจำแนกประเภทปัญหาจาก Log Files ด้วยเทคนิค Machine Learning เพื่อสนับสนุนการวิเคราะห์และการส่งต่อปัญหาในองค์กร Publisher เกม

ส่วนที่ 1: ข้อมูลผู้เชี่ยวชาญ

ชื่อ-นามสกุล: .....

ตำแหน่งงาน: .....

ประสบการณ์ในสายงาน (ปี): ..... ปี

ส่วนที่ 2: คำชี้แจงในการประเมิน

กรุณาพิจารณาตัวอย่างข้อมูล Log Files และผลการจำแนกประเภทที่ระบบ AI (Random Forest) ทำนายออกมา โดยให้คะแนนตามเกณฑ์ดังนี้:

- **+1:** แน่ใจว่าระบบจำแนกประเภทได้ถูกต้องตามความเป็นจริง
- **0:** ไม่แน่ใจว่าระบบจำแนกประเภทได้ถูกต้องหรือไม่
- **-1:** แน่ใจว่าระบบจำแนกประเภทไม่ถูกต้อง

ส่วนที่ 3: ตารางประเมินผลการจำแนกประเภท

ลำดับ	ข้อมูล Log (URL, Status, Time)	ประเภทที่ AI ทำนาย	คะแนนผู้เชี่ยวชาญ		
			+1	0	-1
1	/v1/accounts/login | **200** | 0.12s	**Success**	[ ]	[ ]	[ ]
2	/create | **301** | 16.92s	**Success (Redirect)**	[ ]	[ ]	[ ]
3	/v1/payment/checkout | **499** | 5.50s	**Client_Timeout**	[ ]	[ ]	[ ]
4	/wp-login.php | **444** | 0.00s	**Security_Block**	[ ]	[ ]	[ ]
5	/v2/auth/verify | **502** | 0.45s	**Server_Error**	[ ]	[ ]	[ ]
6	/v1/payment/methods | **200** | 0.08s	**Success**	[ ]	[ ]	[ ]
7	/forgot | **301** | 19.91s	**Success (Redirect)**	[ ]	[ ]	[ ]
8	/v1/accounts/register | **403** | 0.05s	**Client_Error**	[ ]	[ ]	[ ]
9	/api/v1/reports | **504** | 30.00s	**Server_Error**	[ ]	[ ]	[ ]
10	/v1/accounts/sso/callback | **499** | 4.20s	**Client_Timeout**	[ ]	[ ]	[ ]
11	/v1/user/info | **200** | 0.25s	**Success**	[ ]	[ ]	[ ]
12	/games | **301** | 3.51s	**Success (Redirect)**	[ ]	[ ]	[ ]
13	/v1/admin/dashboard | **403** | 0.04s	**Client_Error**	[ ]	[ ]	[ ]
14	/v1/payment/execute | **500** | 1.20s	**Server_Error**	[ ]	[ ]	[ ]

ส่วนที่ 3: ตารางประเมินผลการจำแนกประเภท (ต่อ)

ลำดับ	ข้อมูล Log (URL, Status, Time)	ประเภทที่ AI ทำนาย	คะแนนผู้เชี่ยวชาญ		
			+1	0	-1
15	/v1/payment/gateway/notify | **499** | 6.80s	**Client_Timeout**	[ ]	[ ]	[ ]
16	/api/v1/games/list | **200** | 0.45s	**Success**	[ ]	[ ]	[ ]
17	/create/2?utm_campaign=xx4... | **301** | 4.51s	**Success (Redirect)**	[ ]	[ ]	[ ]
18	.env | **444** | 0.00s	**Security_Block**	[ ]	[ ]	[ ]
19	/v2/auth/token | **502** | 0.15s	**Server_Error**	[ ]	[ ]	[ ]
20	/v1/auth/session | **200** | 0.11s	**Success**	[ ]	[ ]	[ ]
21	/profile | **301** | 12.89s	**Success (Redirect)**	[ ]	[ ]	[ ]
22	/v1/payment/verify | **400** | 0.15s	**Client_Error**	[ ]	[ ]	[ ]
23	/v1/database/sync | **504** | 30.00s	**Server_Error**	[ ]	[ ]	[ ]
24	/v1/accounts/check-email | **200** | 0.09s	**Success**	[ ]	[ ]	[ ]
25	/games/xxxx/connections | **301** | 14.98s	**Success (Redirect)**	[ ]	[ ]	[ ]
26	/api/v1/invalid-path | **404** | 0.02s	**Client_Error**	[ ]	[ ]	[ ]
27	/v1/payment/status | **500** | 0.85s	**Server_Error**	[ ]	[ ]	[ ]
28	/xmlrpc.php | **444** | 0.00s	**Security_Block**	[ ]	[ ]	[ ]
29	/v1/accounts/update-password | **405** | 0.08s	**Client_Error**	[ ]	[ ]	[ ]
30	/v1/payment/cancel | **400** | 0.12s	**Client_Error**	[ ]	[ ]	[ ]

ส่วนที่ 4: การประเมินความเหมาะสมของ Dashboard (Streamlit)

ลำดับ	ประเด็นการประเมิน	คะแนนผู้เชี่ยวชาญ		
		+1	0	-1
1	หมวดหมู่ปัญหาที่กำหนดครอบคลุมปัญหาหลักในองค์กร	[ ]	[ ]	[ ]
2	การแสดงผล Timeline ช่วยให้ระบุช่วงเวลาที่เกิดปัญหาได้ชัดเจน	[ ]	[ ]	[ ]
3	การแสดงผลสัดส่วนปัญหา (Pie Chart) ช่วยในการตัดสินใจได้ดี	[ ]	[ ]	[ ]
4	ระบบสามารถลดภาระงานในการวิเคราะห์ Log ด้วยมือได้จริง	[ ]	[ ]	[ ]

ส่วนที่ 5: ข้อเสนอแนะเพิ่มเติม

.....

.....

.....

ลงชื่อ ..... (ผู้เชี่ยวชาญ)

วันที่ ..... / ..... / .....

ภาคผนวก ข: แบบประเมินความพึงพอใจของผู้ใช้งานระบบ

ส่วนที่ 1: หัวข้อและคำอธิบาย

ชื่อแบบฟอร์ม: แบบสอบถามความพึงพอใจ: ระบบจำแนกประเภทปัญหาจาก Log Files ด้วย Machine Learning

คำอธิบาย: แบบสอบถามนี้เป็นส่วนหนึ่งของงานวิจัยระดับปริญญาโท (MIS) เพื่อประเมินประสิทธิภาพและการใช้งานระบบจำแนกประเภทปัญหาจาก Log Files ด้วยเทคนิค Random Forest และการแสดงผลผ่าน Dashboard (Streamlit) ข้อมูลที่ได้รับจะนำไปใช้เพื่อการวิเคราะห์ในเชิงวิชาการเท่านั้น

ส่วนที่ 2: ข้อมูลทั่วไป

1. **ตำแหน่งงานปัจจุบัน**
   - Developer / Software Engineer
   - IT Operation / SRE / DevOps
   - System Engineer / IT Infrastructure
   - อื่นๆ (โปรดระบุ)
2. **ประสบการณ์การทำงานในสายงาน IT**
   - น้อยกว่า 2 ปี
   - 2 - 5 ปี
   - มากกว่า 5 ปี

พระจอมเกล้าพระนครเหนือ มหาวิทยาลัยเทคโนโลยี

ส่วนที่ 3: การประเมินความพึงพอใจ

คำชี้แจง: โปรดเลือกระดับความเห็นที่ตรงกับความพึงพอใจของท่าน (5 = มากที่สุด, 4 = มาก, 3 = ปานกลาง, 2 = น้อย, 1 = น้อยที่สุด)

คำถาม	**คะแนน**
1. ระบบสามารถจำแนกประเภทปัญหา (เช่น Success, 499, 444) ได้อย่างถูกต้องแม่นยำ	1, 2, 3, 4, 5
2. ความรวดเร็วในการโหลดข้อมูลและการแสดงผลผลลัพธ์ผ่าน Dashboard	1, 2, 3, 4, 5
3. ขั้นตอนการใช้งานระบบ (การอัปโหลดไฟล์) มีความง่ายและไม่ซับซ้อน	1, 2, 3, 4, 5
4. รูปแบบการนำเสนอข้อมูล (Pie Chart, Timeline) ช่วยให้เห็นภาพรวมของปัญหาได้ชัดเจน	1, 2, 3, 4, 5
5. ระบบมีประโยชน์ในการช่วยลดเวลาการวิเคราะห์สาเหตุของปัญหา (MTTR) ในงานจริง	1, 2, 3, 4, 5

ส่วนที่ 4: ข้อเสนอแนะ

คำถาม: ข้อเสนอแนะเพิ่มเติมเพื่อการพัฒนาระบบในอนาคต

ประวัติผู้เขียน

ชื่อ	นายพงศ์ภีระ ชัยพงศ์ภัสสร
ชื่อการค้นคว้าอิสระ	การพัฒนาระบบจำแนกประเภทปัญหาจาก Log Files ด้วยเทคนิคMachine Learning เพื่อสนับสนุนการวิเคราะห์ และการส่งต่อปัญหาในองค์กร Publisher เกม
สาขาวิชา	ระบบสารสนเทศเพื่อการจัดการ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
ประวัติ	ประวัติส่วนตัว เกิดวันที่ 23 ตุลาคม พ.ศ. 2525
	เรียนจบระดับปริญญาตรี สำเร็จการศึกษาระดับปริญญาตรี หลักสูตรบริหารธุรกิจ
	คณะบริหารธุรกิจ สาขาวิชา ระบบสารสนเทศทางคอมพิวเตอร์ มหาวิทยาลัยมหาวิทยาลัยเทคโนโลยีราชมงคลตะวันออก เขตพื้นที่จักรพงษภูวนารถ ปี พ.ศ. 2549
	ปัจจุบันทำงานอยู่ที่ Electronics Extreme Limited