การใช้เทคนิค Data Cleansing เพื่อปรับปรุงคุณภาพข้อมูลทะเบียนนักศึกษาและประเมินผลโดยระบบต้นแบบ

โดย
นายเอนก รุ่งนาไร่

การค้นคว้าอิสระนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรมหาบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศและนวัตกรรมดิจิทัล แผน 2 แบบวิชาชีพ
ภาควิชาคอมพิวเตอร์
มหาวิทยาลัยศิลปากร
ปีการศึกษา 2568

การใช้เทคนิค Data Cleansing เพื่อปรับปรุงคุณภาพข้อมูลทะเบียนนักศึกษาและประเมินผลโดยระบบต้นแบบ

โดย
นายเอนก รุ่งนาไร่

การค้นคว้าอิสระนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรมหาบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศและนวัตกรรมดิจิทัล แผน 2 แบบวิชาชีพ
ภาควิชาคอมพิวเตอร์
มหาวิทยาลัยศิลปากร
ปีการศึกษา 2568

IMPROVING STUDENT REGISTRATION DATA QUALITY USING DATA CLEANSING TECHNIQUES INCOMPLETE AND INCORRECT DATA: A PROTOTYPE-BASED EVALUATION

By

Mr. Anek RUNGNARAI

An Independent Study Submitted in Partial Fulfillment of the Requirements
for Master of Science INFORMATION TECHNOLOGY AND DIGITAL INNOVATION
Department of COMPUTER SCIENCE
Academic Year 2025

หัวข้อ	การใช้เทคนิค Data Cleansing เพื่อปรับปรุงคุณภาพข้อมูลทะเบียนนักศึกษาและประเมินผลโดยระบบต้นแบบ
โดย	นายเอนก รุ่งนาไร่
สาขาวิชา	เทคโนโลยีสารสนเทศและนวัตกรรมดิจิทัล แผน 2 แบบวิชาชีพ
อาจารย์ที่ปรึกษาหลัก	ผู้ช่วยศาสตราจารย์ ดร. อรวรรณ เชาวลิต

คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร ได้รับพิจารณาอนุมัติให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรมหาบัณฑิต

.....คณบดีคณะวิทยาศาสตร์
(ผู้ช่วยศาสตราจารย์ ดร. กรกช ชั้นจิรกุล)

พิจารณาเห็นชอบโดย

.....ประธานกรรมการ
(รองศาสตราจารย์ ดร. โอม ศรนิล)

.....อาจารย์ที่ปรึกษาหลัก
(ผู้ช่วยศาสตราจารย์ ดร. อรวรรณ เชาวลิต)

.....ผู้ทรงคุณวุฒิภายใน
(ผู้ช่วยศาสตราจารย์ ดร. สัจจาภรณ์ ไวจรรยา)

660720067 : เทคโนโลยีสารสนเทศและนวัตกรรมดิจิทัล แผน 2 แบบวิชาชีพ

นาย เอนก รุ่งนาไร่: การใช้เทคนิค Data Cleansing เพื่อปรับปรุงคุณภาพข้อมูลทะเบียนนักศึกษาและประเมินผลโดยระบบต้นแบบ อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก : ผู้ช่วยศาสตราจารย์ ดร. อรวรรณ เชาวลิต

งานวิจัยนี้มีวัตถุประสงค์เพื่อแก้ไขปัญหาคุณภาพข้อมูลในระบบทะเบียนนักศึกษาของมหาวิทยาลัยเอกชน ซึ่งพบว่ามีข้อบกพร่องหลายประการ เช่น ข้อมูลซ้ำซ้อน ข้อมูลขาดหาย ข้อมูลไม่เป็นไปตามรูปแบบที่กำหนด และความไม่สอดคล้องกันระหว่างตารางข้อมูล ปัญหาเหล่านี้ส่งผลกระทบโดยตรงต่อกระบวนการบริหารจัดการและการตัดสินใจเชิงนโยบาย อีกทั้งยังสร้างความเสี่ยงต่อความถูกต้องของข้อมูลเชิงวิชาการและการรายงานต่อหน่วยงานกำกับดูแล งานวิจัยนี้จึงมุ่งเน้นการศึกษาและพัฒนากระบวนการทำความสะอาดข้อมูล (Data Cleansing) เพื่อยกระดับคุณภาพข้อมูลให้อยู่ในเกณฑ์มาตรฐานที่สามารถนำไปใช้ประโยชน์ได้จริง

แนวทางการวิจัยประกอบด้วย 4 ระดับหลัก ได้แก่ (1) การทำความสะอาดข้อมูลด้วยกฎเชิงตรรกะ (Rule-Based Cleaning) เพื่อกำหนดเงื่อนไขที่ชัดเจนในการตรวจสอบและแก้ไขข้อมูล (2) การใช้เครื่องมือซอฟต์แวร์ เช่น OpenRefine ในการจัดกลุ่มข้อความ (Text Clustering) และการแก้ไขข้อมูลผิดรูปแบบ (3) การประยุกต์ใช้เทคนิคเชิงการเรียนรู้ของเครื่อง (Machine Learning-Based Cleaning) โดยเฉพาะการตรวจจับค่าผิดปกติด้วยอัลกอริทึม เช่น One-Class SVM เพื่อค้นหาข้อมูลที่แตกต่างจากรูปแบบปกติ และ (4) การตรวจสอบและปรับปรุงข้อมูลด้วยวิธีการเชิงมนุษย์ (Manual Cleaning) สำหรับกรณีข้อมูลจำนวนน้อยหรือมีความซับซ้อนสูงที่ระบบอัตโนมัติไม่สามารถจัดการได้

กระบวนการดำเนินงานวิจัยใช้แนวคิดเชิง ETL (Extract–Transform–Load) โดยเริ่มจากการดึงข้อมูลจากฐานข้อมูลทะเบียนเดิมจำนวน 21 ตาราง มาทำการวิเคราะห์คุณภาพเบื้องต้น (Data Profiling) เพื่อตรวจสอบความสมบูรณ์ ความถูกต้อง และความสอดคล้องของข้อมูล จากนั้นจึงนำเทคนิคการทำความสะอาดที่เลือกใช้มาใช้แก้ไขข้อมูล และทำการตรวจสอบซ้ำหลังการปรับปรุงเพื่อติดตามผลลัพธ์ ผลการวิจัยพบว่าหลังการดำเนินการ Data Cleansing ค่าความครบถ้วน (Completeness) ของข้อมูลเพิ่มขึ้นจากร้อยละ 66.4 เป็น 100 ความถูกต้อง (Accuracy) และความเป็นไปตามรูปแบบ (Validity) ได้รับการปรับปรุงให้สมบูรณ์ทั้งหมด อีกทั้งยังสามารถลดความซ้ำซ้อน (Uniqueness) และความไม่สอดคล้องกัน (Consistency) ได้อย่างมีนัยสำคัญ

ผลลัพธ์ชี้ให้เห็นว่ากระบวนการ Data Cleansing ที่ออกแบบขึ้นสามารถปรับปรุงคุณภาพ

ข้อมูลทะเบียนนักศึกษาได้อย่างเป็นระบบ และสามารถพัฒนาเป็นต้นแบบสำหรับการจัดการข้อมูลในมหาวิทยาลัยอื่น ๆ ได้ในอนาคต ทั้งนี้ ประโยชน์สำคัญที่ได้รับไม่เพียงแต่ช่วยลดความผิดพลาดในการจัดการข้อมูล แต่ยังช่วยเพิ่มประสิทธิภาพในการให้บริการนักศึกษาและบุคลากร อีกทั้งยังเสริมสร้างความเชื่อมั่นในการนำข้อมูลไปใช้ในการวางแผนและการตัดสินใจเชิงกลยุทธ์ของผู้บริหาร

660720067 : Major INFORMATION TECHNOLOGY AND DIGITAL INNOVATION

Mr. Anek RUNGNARAI : Improving Student Registration Data Quality Using Data Cleansing Techniques Incomplete and Incorrect Data: A PROTOTYPE-BASED EVALUATION Thesis advisor : Assistant Professor Dr. Orawan Chaowalit

This research aims to address data quality problems in the student registration system of a private university, which include duplicate records, missing values, invalid formats, and inconsistencies across tables. Such issues directly affect administrative processes, policy decision-making, and academic reporting to regulatory authorities, creating risks in the accuracy and reliability of academic information. Therefore, this study focuses on designing and developing a systematic data cleansing process to improve the quality of student registration data and ensure that it meets practical and institutional standards.

The research methodology is structured into four main levels: (1) Rule-Based Cleaning, where explicit logical rules are applied to detect and correct erroneous values; (2) Software-Assisted Cleaning, using tools such as OpenRefine to perform text clustering, format validation, and semi-automated corrections; (3) Machine Learning-Based Cleaning, particularly anomaly detection with algorithms such as One-Class SVM to identify abnormal or outlier data; and (4) Manual Cleaning, reserved for small-scale or complex cases that automated methods cannot fully handle.

The overall process adopts an ETL (Extract–Transform–Load) approach. Data were extracted from 21 tables in the legacy registration database and underwent data profiling to assess completeness, accuracy, and consistency. Cleansing techniques were then applied systematically, followed by post-cleansing evaluation. The results demonstrate that data completeness increased from 66.4% to 100%, while accuracy and validity improved significantly to meet required standards. Duplicates and inconsistencies were also reduced to a substantial extent.

กิตติกรรมประกาศ

งานวิจัยฉบับนี้สำเร็จลุล่วงได้ด้วยความกรุณา ความเมตตา และการสนับสนุนจากหลายฝ่าย ข้าพเจ้าขอกราบขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร. อรวรรณ เชาวลิต อาจารย์ที่ปรึกษาหลัก ผู้ได้กรุณาให้คำแนะนำอย่างใกล้ชิด ตั้งแต่การกำหนดแนวทางการวิจัย การวางโครงสร้าง ไปจนถึงการเขียนรายงานวิจัย ให้ข้อคิดเห็นอย่างตรงไปตรงมาและสร้างสรรค์ ตลอดจนเป็นกำลังใจสำคัญที่ทำให้งานวิจัยฉบับนี้สำเร็จสมบูรณ์

ขอกราบขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร. นรงค์ ฉิมพาลี คณบดีคณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร ที่ได้ให้การสนับสนุนด้านการเรียนการสอนและทรัพยากรทางวิชาการ รวมทั้งเอื้อเฟื้อโอกาสในการศึกษาค้นคว้า

ขอกราบขอบพระคุณ รองศาสตราจารย์ ดร. โอม ศรนิล ประธานกรรมการสอบ และ อาจารย์ ดร. สัจจาภรณ์ ไวจรรยา กรรมการผู้ทรงคุณวุฒิภายใน ที่ได้กรุณาให้คำแนะนำ ข้อคิดเห็น และข้อเสนอแนะที่เป็นประโยชน์อย่างยิ่ง ซึ่งช่วยเสริมให้งานวิจัยมีความสมบูรณ์ทั้งในด้านแนวคิด ทฤษฎี และวิธีการดำเนินงาน

ขอขอบคุณ คณาจารย์ทุกท่านในหลักสูตรวิทยาศาสตรมหาบัณฑิต ที่ได้ถ่ายทอดความรู้และประสบการณ์อันมีค่า ทั้งทางวิชาการและการประยุกต์ใช้ในทางปฏิบัติ ซึ่งเป็นรากฐานสำคัญของการทำวิจัยในครั้งนี้

นอกจากนี้ ข้าพเจ้าขอขอบคุณ เจ้าหน้าที่ประจำหน่วยงานที่ให้ข้อมูล และ ผู้ให้ความร่วมมือทุกท่าน ที่สละเวลาอันมีค่า ตลอดจนให้ความร่วมมือในการตอบแบบสอบถามและการสัมภาษณ์ ซึ่งมีส่วนสำคัญในการเก็บรวบรวมข้อมูลที่จำเป็นต่อการวิเคราะห์และการสรุปผลการวิจัย

ขอขอบคุณ เพื่อนนักศึกษาร่วมรุ่นและเพื่อนร่วมงาน ที่คอยช่วยเหลือ แบ่งปันความคิดเห็น ให้คำปรึกษา และสร้างบรรยากาศการเรียนรู้ที่ดีงาม รวมถึงให้กำลังใจอย่างต่อเนื่องในทุกช่วงเวลาของการศึกษา

ท้ายที่สุด ข้าพเจ้าขอแสดงความซาบซึ้งอย่างสุดซึ้งต่อ ครอบครัว ที่เป็นแรงสนับสนุนสำคัญในทุกด้าน ทั้งความรัก ความเข้าใจ การเสียสละ และกำลังใจที่มั่นคง ซึ่งช่วยให้ข้าพเจ้ามีกำลังใจในการก้าวผ่านอุปสรรค จนสามารถทำงานวิจัยฉบับนี้ได้สำเร็จลุล่วงตามเป้าหมาย

ความสำเร็จในครั้งนี้ขอมอบแด่ทุกท่านที่ได้มีส่วนเกี่ยวข้องด้วยความเคารพและสำนึกในพระคุณอย่างหาที่สุดมิได้

เอนก รุ่งนาไร่

สารบัญ

หน้า

สารบัญตาราง

หน้า

สารบัญภาพ

หน้า

ทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในยุคปัจจุบัน เทคโนโลยีสารสนเทศ (Information Technology: IT) ได้เข้ามามีบทบาทสำคัญในการบริหารจัดการของสถาบันอุดมศึกษา โดยเฉพาะในการจัดเก็บ วิเคราะห์ และใช้ข้อมูลเพื่อประกอบการตัดสินใจเชิงกลยุทธ์ ข้อมูลต่าง ๆ เช่น ข้อมูลนักศึกษา รายวิชา การลงทะเบียน และผลการเรียน หากได้รับการจัดการอย่างถูกต้องและมีคุณภาพ ย่อมส่งผลต่อประสิทธิภาพของการดำเนินงานในระดับองค์กร และสามารถนำไปใช้ในการวางแผนพัฒนาอย่างเป็นระบบในระยะยาว

อย่างไรก็ตามในหลายมหาวิทยาลัย ยังคงประสบปัญหาการใช้ระบบฐานข้อมูลเดิม (Legacy Systems) ที่พัฒนาขึ้นตั้งแต่ยุคก่อน โดยระบบเหล่านี้มักไม่รองรับเทคโนโลยีสมัยใหม่ เช่น การจัดการ Unicode หรือการบูรณาการกับระบบสารสนเทศอื่น ๆ ได้อย่างยืดหยุ่น ส่งผลให้การขยายตัวของข้อมูลในปัจจุบันเกิดข้อจำกัด ทั้งในด้านโครงสร้างข้อมูล การบำรุงรักษา และประสิทธิภาพในการใช้งาน ขณะเดียวกัน บุคลากรรุ่นใหม่มักไม่มีความรู้เชิงลึกในการดูแลระบบเก่าเหล่านี้ ทำให้ข้อมูลที่สะสมอยู่มีปัญหาด้านคุณภาพ เช่น ข้อมูลซ้ำซ้อน ข้อมูลที่ขาดหาย หรือมีรูปแบบไม่สอดคล้องกับมาตรฐานที่กำหนด

กรณีศึกษาของมหาวิทยาลัยคริสเตียน จังหวัดนครปฐม พบว่า ระบบทะเบียนนักศึกษาที่ใช้งานอยู่มีข้อจำกัดจากการพึ่งพาระบบฐานข้อมูลเชิงสัมพันธ์รุ่นเก่า (Pervasive PSQL v11) ซึ่งทำให้ข้อมูลที่จัดเก็บมีลักษณะไม่สมบูรณ์ และยากต่อการนำไปประยุกต์ใช้ในการพัฒนาระบบใหม่ที่ต้องการความถูกต้องแม่นยำสูง ปัญหาเหล่านี้จึงสะท้อนถึงความจำเป็นในการยกระดับคุณภาพข้อมูล (Data Quality) และวางรากฐานระบบใหม่ที่สามารถรองรับข้อมูลสะอาด พร้อมใช้งานในระดับองค์กรได้อย่างมีประสิทธิภาพ

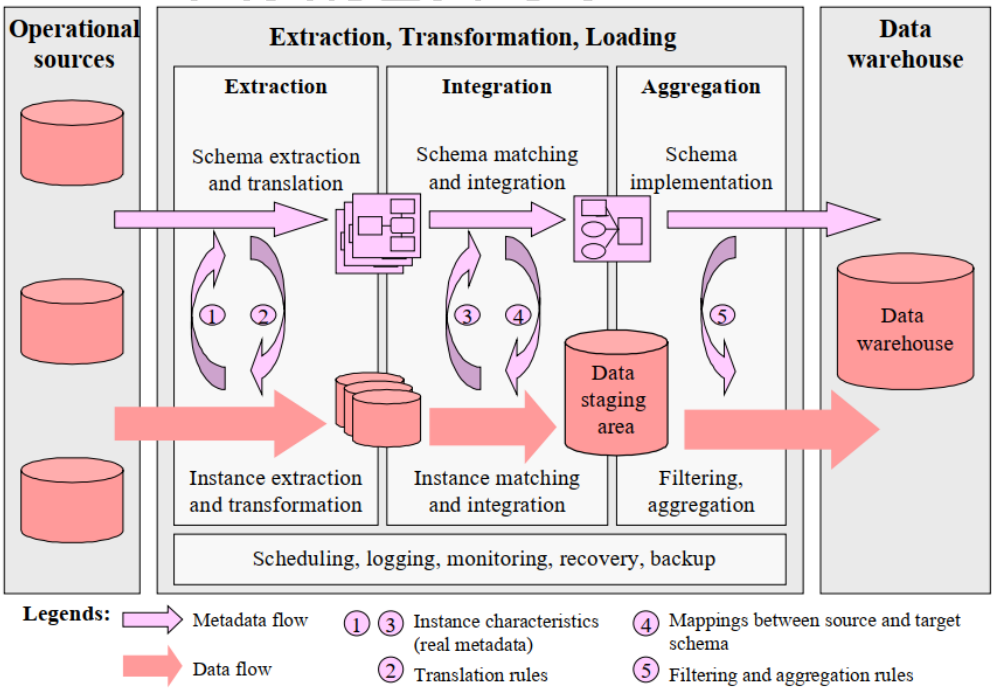
แนวคิดเรื่องคุณภาพข้อมูลที่นำมาใช้ในงานวิจัยนี้อ้างอิงจาก Wang & Strong (1996) ซึ่งได้นิยามว่าข้อมูลที่มีคุณภาพควรมีลักษณะสำคัญ เช่น ความน่าเชื่อถือ (Believability) ความถูกต้อง (Accuracy) ความเป็นกลาง (Objectivity) และแหล่งที่มาที่เชื่อถือได้ (Reputation) ซึ่งส่งผลโดยตรงต่อความแม่นยำในการวิเคราะห์และการตัดสินใจ

นอกจากนี้ DAMA International (2017) ยังได้เสนอกรอบมิติของคุณภาพข้อมูล (Data Quality Dimensions) ซึ่งครอบคลุมถึง ความสมบูรณ์ (Completeness), ความสอดคล้องกัน (Consistency), ความทันเวลา (Timeliness) และความสามารถในการเข้าถึง (Accessibility) โดยเฉพาะอย่างยิ่ง DAMA DMBOK 2 ยังได้เน้นถึงมิติของ Validity หรือความถูกต้องของรูปแบบ

ข้อมูล เช่น การจัดรูปแบบรหัสนักศึกษา หรือวันที่ให้เป็นไปตามข้อกำหนด เพื่อหลีกเลี่ยงข้อผิดพลาดในการใช้งานจริง

เพื่อปรับปรุงคุณภาพข้อมูลให้เป็นไปตามมาตรฐานข้างต้น การทำ Data Cleansing จึงมีบทบาทสำคัญ โดยมีเป้าหมายในการตรวจสอบ แก้ไข และปรับข้อมูลให้พร้อมใช้งานได้อย่างถูกต้องและครบถ้วน ซึ่ง Batini et al. (2009) ได้เสนอว่า Data Cleansing เป็นขั้นตอนพื้นฐานที่ช่วยลดความเสี่ยงจากข้อมูลผิดพลาด และเพิ่มความน่าเชื่อถือของระบบสารสนเทศในเชิงการบริหารจัดการ

แนวทางของ DAMA DMBOK 2 ยังได้แสดงขั้นตอนของกระบวนการ Data Cleansing ที่สามารถประยุกต์ใช้ได้จริง ได้แก่ การตรวจสอบความถูกต้องของข้อมูล (Data Validation), การจัดการข้อมูลที่ขาดหายไป (Handling Missing Data), การลบข้อมูลซ้ำซ้อน (Duplicate Removal), การตรวจสอบความสอดคล้องกันของข้อมูล (Consistency Check) และการปรับปรุงข้อมูลให้เป็นมาตรฐาน (Data Standardization) ทั้งนี้ กระบวนการเหล่านี้สามารถผนวกรวมเข้ากับขั้นตอนของกระบวนการ ETL (Extraction, Transformation, Loading) ซึ่งมักถูกใช้ในงานด้านคลังข้อมูล เพื่อจัดการกับข้อมูลจำนวนมากอย่างเป็นระบบ



ภาพที่ 1 กระบวนการ ETL สำหรับการเตรียมข้อมูล

ที่มา: Rahm and Do (2000)

จากแนวคิดข้างต้น งานวิจัยนี้จึงมีเป้าหมายเพื่อออกแบบและพัฒนาระบบทะเบียนนักศึกษาใหม่ ที่สามารถรองรับข้อมูลที่ผ่านการทำความสะอาดแล้ว โดยแยกออกจากระบบเดิม พร้อมทั้งใช้ระบบต้นแบบในการทดสอบความถูกต้องของข้อมูลที่ได้รับการปรับปรุง การเชื่อมโยงแนวคิด Data Quality, กระบวนการ ETL และเทคนิค Data Cleansing จะช่วยให้สามารถแก้ไขปัญหาข้อมูลผิดพลาดที่มีอยู่เดิม และวางรากฐานสำหรับการจัดการข้อมูลที่ยั่งยืนในอนาคต

1.2 วัตถุประสงค์ของการวิจัย

1.2.1 เพื่อศึกษาและวิเคราะห์ปัญหาคุณภาพของข้อมูลจากระบบทะเบียนนักศึกษาเดิม รวมถึงแนวคิดและหลักการที่เกี่ยวข้องกับ Data Quality และกระบวนการ Data Cleansing

1.2.2 เพื่อออกแบบกระบวนการ Data Cleansing โดยเฉพาะในบริบทของข้อมูลทะเบียนนักศึกษา ที่ครอบคลุมตั้งแต่การตรวจสอบ แก้ไข และปรับมาตรฐานข้อมูล ภายใต้กรอบของกระบวนการ ETL

1.2.3 เพื่อพัฒนาระบบต้นแบบทะเบียนนักศึกษาใหม่ที่สามารถรองรับข้อมูลที่ผ่านการ Cleansing แล้ว และมีความยืดหยุ่นต่อการบูรณาการข้อมูลในอนาคต

1.2.4 เพื่อประเมินประสิทธิภาพของกระบวนการ Data Cleansing และระบบต้นแบบ โดยใช้เมตริกด้านคุณภาพข้อมูล (เช่น Accuracy, Completeness, Consistency) ในการเปรียบเทียบข้อมูลก่อนและหลังการ Cleansing รวมถึงประเมินผลการใช้งานจากมุมมองของผู้ใช้ระบบต้นแบบ

1.3 ขอบเขตงานวิจัย

1.3.1 งานวิจัยนี้ศึกษาข้อมูลจากระบบทะเบียนนักศึกษาเดิม (Legacy Student Database) ของมหาวิทยาลัยคริสเตียน จังหวัดนครปฐม โดยข้อมูลถูกจัดเก็บในระบบฐานข้อมูล Pervasive PSQL v11 ซึ่งยังคงใช้งานในปัจจุบัน

1.3.2 ออกแบบกระบวนการ Data Cleansing ที่สามารถจัดการกับข้อมูลที่มีปัญหา ได้แก่

1.3.2.1 ข้อมูลซ้ำซ้อน (Duplicate Data) เช่น นักศึกษาคนเดียวกันมีหลายระเบียน

1.3.2.2 ข้อมูลที่ขาดหายไป (Missing Data) เช่น ไม่มีเลขบัตรประชาชน ไม่มีที่อยู่ หรือไม่มีเบอร์โทรศัพท์

1.3.2.3 ข้อมูลไม่ถูกต้อง (Incorrect Data) เช่น วันเกิดไม่ตรงกับเลขบัตร, คณะไม่ตรงกับสาขา

โดยใช้แนวคิด ETL และประยุกต์ใช้เทคนิคที่หลากหลาย ได้แก่ Rule-Based Data Cleaning, Software-Based Data Cleaning ด้วย OpenRefine และ Machine Learning-Based Data Cleaning โดยใช้โมเดล SDCM ร่วมกับ K-Nearest Neighbors

ทั้งนี้ ผู้วิจัยจะพิจารณาเลือกเฉพาะฟิลด์ข้อมูลที่มีปัญหาชัดเจนและมีผลกระทบสูงต่อระบบ เช่น ฟิลด์ข้อความและตัวเลขที่เกี่ยวข้องกับอัตลักษณ์นักศึกษา เพื่อให้การดำเนินงานอยู่ในขอบเขตที่สามารถทดลองได้จริงภายในระยะเวลาที่กำหนด

1.3.3 พัฒนาระบบทะเบียนนักศึกษาใหม่ในรูปแบบ Prototype ที่สามารถรองรับข้อมูลที่ผ่านการ Cleansing แล้ว โดยออกแบบตามแนวทาง MVC, Clean Architecture เพื่อให้ระบบมีความยืดหยุ่นและสามารถต่อยอดได้ในอนาคต

1.3.4 ประเมินประสิทธิภาพของทั้งกระบวนการ Cleansing และระบบต้นแบบ โดยใช้เมตริกด้านคุณภาพข้อมูล เช่น Accuracy, Completeness, Consistency เพื่อเปรียบเทียบข้อมูลก่อนและหลังการ Cleansing และเพิ่มเติมด้วยการเก็บข้อมูลความพึงพอใจของผู้ใช้งานจริงผ่านแบบสอบถาม

1.4 ขั้นตอนการดำเนินการวิจัย

การดำเนินงานวิจัยในครั้งนี้ได้ออกแบบให้สอดคล้องกับแนวคิดการจัดการข้อมูลตามกระบวนการ ETL (Extraction, Transformation, Loading) ผนวกกับเทคนิคการทำ Data Cleansing อย่างเป็นระบบ โดยมีขั้นตอนดังต่อไปนี้

1.4.1 ศึกษาระบบทะเบียนนักศึกษาปัจจุบัน

วิเคราะห์โครงสร้างการทำงานและรูปแบบการใช้งานของระบบทะเบียนนักศึกษาที่มีอยู่ในปัจจุบัน โดยมีเป้าหมายเพื่อออกแบบระบบทะเบียนนักศึกษาใหม่ในรูปแบบต้นแบบ (Prototype) ที่สามารถรองรับข้อมูลที่ผ่านการทำ Data Cleansing แล้ว การศึกษาระบบปัจจุบันจะช่วยระบุข้อมูลที่จำเป็นต้องดำเนินการทำความสะอาด และหลีกเลี่ยงการนำข้อมูลที่ไม่เกี่ยวข้องมาใช้งาน

1.4.2 ดำเนินการ Extract ข้อมูลจากฐานข้อมูลเดิม

ทำการส่งออกข้อมูลนักศึกษาจากระบบฐานข้อมูล Pervasive PSQL v11 ให้อยู่ในรูปแบบที่สามารถนำมาใช้งานได้ เช่น CSV หรือ Excel โดยยังไม่ปรับเปลี่ยนค่าภายใน

1.4.3 วิเคราะห์และจัดเตรียมข้อมูลเบื้องต้น (Data Profiling & Preprocessing)

ตรวจสอบคุณภาพข้อมูลเบื้องต้น เพื่อแยกข้อมูลที่มีปัญหาออกมา ได้แก่ ข้อมูลซ้ำซ้อน ข้อมูลขาดหาย และข้อมูลไม่ถูกต้อง พร้อมจัดเตรียมข้อมูลให้อยู่ในรูปแบบพร้อมสำหรับการปรับปรุง

1.4.4 ศึกษาและพัฒนาเทคนิคการทำ Data Cleansing

1.4.4.1 Rule-Based Data Cleaning

ใช้กฎและเงื่อนไขที่กำหนดไว้ล่วงหน้าเพื่อตรวจสอบความถูกต้องของข้อมูล โดยอาศัยตรรกะและกฎ if-else ซึ่งเหมาะสำหรับข้อมูลที่มีโครงสร้างชัดเจน เช่น ข้อมูลทะเบียนนักศึกษา

1.4.4.2 Software-Based Cleaning

ใช้ซอฟต์แวร์สำหรับการทำ Data Cleansing เพื่อเพิ่มประสิทธิภาพของกระบวนการ โดยเลือกใช้ OpenRefine ในการตรวจสอบและทำความสะอาดข้อมูล

1.4.4.3 Machine Learning-Based Data Cleaning

ใช้เทคนิค Machine Learning เพื่อช่วยตรวจจับข้อผิดพลาดของข้อมูลโดยอัตโนมัติ โดยเลือกใช้โมเดล SDCM (Supervised Dataset Cleaning Model) และอัลกอริทึม K-Nearest Neighbors (KNN) ในการระบุค่าที่ผิดพลาด

1.4.5 พัฒนา Prototype ระบบทะเบียนนักศึกษาใหม่

สร้างระบบต้นแบบที่สามารถนำเข้าข้อมูลหลังผ่านกระบวนการ Cleansing ได้ และออกแบบให้รองรับการตรวจสอบ แก้ไข และค้นหาข้อมูลอย่างมีประสิทธิภาพ โดยใช้แนวคิด MVC, Clean Architecture ในการออกแบบระบบ

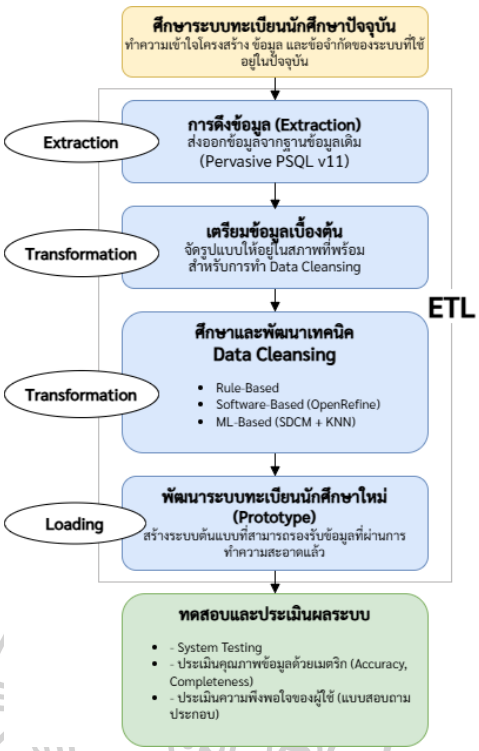
1.4.6 การทดสอบระบบ ประเมินผล และสรุปผลการดำเนินการ

ขั้นตอนสุดท้ายเป็นการทดสอบและประเมินประสิทธิภาพของระบบต้นแบบ พร้อมทั้งสรุปผลการดำเนินการ โดยแบ่งเป็น 3 กิจกรรมหลัก ได้แก่

1.4.6.1 ทดสอบระบบ (System Testing) เพื่อตรวจสอบว่าระบบสามารถทำงานได้ตามที่ออกแบบและรองรับข้อมูลที่ผ่านการ Cleansing ได้อย่างถูกต้อง

1.4.6.2 เปรียบเปรียบเทียบข้อมูลก่อนและหลังการทำ Data Cleansing โดยใช้เมตริกด้านคุณภาพข้อมูล เช่น Accuracy, Completeness และ Consistency เพื่อวัดประสิทธิผลของกระบวนการที่นำมาใช้

1.4.6.3 ประเมินความพึงพอใจของผู้ใช้งาน (User Satisfaction Evaluation) โดยใช้แบบสอบถามกับเจ้าหน้าที่ฝ่ายทะเบียน เพื่อเก็บข้อมูลเสริมจากประสบการณ์ใช้งานจริงในด้านความสะดวก ความเร็ว และการลดภาระงาน ซึ่งใช้ประกอบกับผลการประเมินเชิงปริมาณจากเมตริกด้านคุณภาพข้อมูล



ภาพที่ 2 กระบวนการดำเนินงานวิจัยตามแนวคิด ETL และ Data Cleansing

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 ด้านการจัดการข้อมูล

1.5.1.1 ข้อมูลทะเบียนนักศึกษามีคุณภาพตามมาตรฐาน สามารถนำไปใช้งานได้อย่างมีประสิทธิภาพ

1.5.1.2 ช่วยลดภาระของเจ้าหน้าที่ฝ่ายทะเบียนในการตรวจสอบและแก้ไขข้อมูลด้วยตนเอง

1.5.1.3 รองรับการโอนย้ายข้อมูลเข้าสู่ระบบใหม่ได้อย่างราบรื่น ลดข้อผิดพลาดที่อาจเกิดขึ้นระหว่างกระบวนการย้ายข้อมูล

1.5.1.4 สนับสนุนการตัดสินใจของผู้บริหารให้แม่นยำยิ่งขึ้น เนื่องจากสามารถเข้าถึงข้อมูลที่ถูกต้องและเป็นปัจจุบัน

1.5.2 ด้านระบบทะเบียนนักศึกษาต้นแบบ (Prototype)

1.5.2.1 ระบบต้นแบบที่พัฒนาขึ้นใช้สำหรับการทดสอบความพร้อมของข้อมูลที่ผ่านการ Cleansing แล้ว และสามารถนำไปต่อยอดเป็นระบบทะเบียนนักศึกษาที่สมบูรณ์และพร้อมใช้งานจริง รวมถึงขยายผลไปยังระบบงานอื่นที่เกี่ยวข้องในอนาคต

1.5.2.2 เป็นแนวทางให้มหาวิทยาลัยสามารถออกแบบมาตรฐานการจัดเก็บและบันทึกข้อมูลที่มีประสิทธิภาพ

1.5.3 ด้านองค์ความรู้และการศึกษาด้าน Data Cleansing

1.5.3.1 สามารถใช้เป็นกรณีศึกษาเพื่อเป็นต้นแบบสำหรับมหาวิทยาลัยหรือองค์กรอื่น ๆ ที่ต้องการพัฒนาระบบจัดการข้อมูลขนาดใหญ่

1.5.3.2 ช่วยส่งเสริมองค์ความรู้และแนวทางปฏิบัติในการปรับปรุงคุณภาพข้อมูลและการบริหารจัดการข้อมูลทะเบียนนักศึกษา

1.5.3.3 เป็นตัวอย่างเชิงเทคนิคของการเลือกใช้ Data Cleansing ที่หลากหลายและเหมาะสมกับประเภทของปัญหาข้อมูล เช่น การใช้ Rule-Based กับเบอร์โทรศัพท์ และ Reference Matching กับชื่อเขต/อำเภอ

1.6 นิยามศัพท์เฉพาะ

1.6.1 Data Quality (คุณภาพข้อมูล)

หมายถึง ลักษณะของข้อมูลที่เหมาะสมต่อการนำไปใช้งานอย่างมีประสิทธิภาพ โดยประกอบด้วยมิติต่าง ๆ เช่น ความถูกต้อง (Accuracy), ความสมบูรณ์ (Completeness), ความสอดคล้องกัน (Consistency), ความทันสมัย (Timeliness) และความสามารถในการเข้าถึง (Accessibility) Wang & Strong (1996) DAMA International (2017)

1.6.2 Data Cleansing (การทำความสะอาดข้อมูล)

หมายถึง กระบวนการตรวจสอบ แก้ไข หรือปรับปรุงข้อมูลที่ผิดพลาด ซ้ำซ้อน ขาดหาย หรือไม่เป็นไปตามมาตรฐาน เพื่อให้ข้อมูลมีความถูกต้องและพร้อมใช้งาน Batini et al. (2009)

1.6.3 Duplicate Data (ข้อมูลซ้ำซ้อน)

หมายถึง ข้อมูลที่ซ้ำกันในระบบ เช่น นักศึกษาคนเดียวมีการบันทึกข้อมูลไว้หลายระเบียน ซึ่งอาจเกิดจากการป้อนข้อมูลผิดพลาดหรือไม่มีการตรวจสอบซ้ำ

1.6.4 Missing Data (ข้อมูลที่ขาดหาย)

หมายถึง ข้อมูลที่จำเป็นแต่ไม่มีการระบุไว้ในระบบ เช่น ขาดเลขประจำตัวประชาชน ที่อยู่ หรือเบอร์โทรศัพท์

1.6.5 Inconsistent Data (ข้อมูลไม่สอดคล้องกัน)

หมายถึง ข้อมูลที่มีความขัดแย้งกันหรือใช้รูปแบบไม่เป็นมาตรฐาน เช่น รูปแบบวันที่ที่ต่างกันในระบบเดียวกัน

1.6.6 Rule-Based Data Cleaning (การทำความสะอาดข้อมูลแบบใช้กฎ)

หมายถึง เทคนิคการทำ Data Cleansing โดยใช้เงื่อนไขหรือกฎ (เช่น if–else) ที่กำหนดล่วงหน้าเพื่อตรวจสอบความถูกต้องของข้อมูล ซึ่งเหมาะกับข้อมูลที่มีโครงสร้างชัดเจน

1.6.7 Software-Based Data Cleaning (การทำความสะอาดข้อมูลด้วยซอฟต์แวร์)

หมายถึง การใช้โปรแกรมเฉพาะทางในการช่วยทำ Data Cleansing เช่น OpenRefine เพื่อเพิ่มความแม่นยำและลดภาระงานด้วยมือ

1.6.8 Machine Learning-Based Data Cleaning (การทำความสะอาดข้อมูลด้วยการเรียนรู้ของเครื่อง)

หมายถึง การใช้เทคนิค Machine Learning เพื่อช่วยวิเคราะห์ ตรวจจับ และแก้ไขข้อมูลที่ผิดพลาดโดยอัตโนมัติ เช่น การใช้โมเดล SDCM และอัลกอริทึม K-Nearest Neighbors (KNN)

1.6.9 Prototype (ต้นแบบระบบ)

หมายถึง ระบบที่ได้รับการพัฒนาขึ้นในรูปแบบจำลองเพื่อตรวจสอบการทำงานและทดลองใช้งานก่อนพัฒนาเป็นระบบจริง

1.6.10 Clean Architecture

หมายถึง แนวทางการออกแบบซอฟต์แวร์ที่เน้นการแยกส่วนของระบบอย่างเป็นระบบ เพื่อความยืดหยุ่นในการปรับปรุง แก้ไข และบำรุงรักษา

1.6.11 MVC (Model–View–Controller)

หมายถึง รูปแบบการออกแบบซอฟต์แวร์ (Software Design Pattern) ที่ใช้สำหรับแยกความรับผิดชอบของส่วนประกอบต่าง ๆ ภายในระบบออกจากกันอย่างชัดเจน

1.6.12 Pervasive PSQL v11

หมายถึง ซอฟต์แวร์ระบบฐานข้อมูลเชิงสัมพันธ์ที่ใช้ในระบบทะเบียนนักศึกษาเดิมของมหาวิทยาลัย เป็นระบบที่มีการใช้งานมานานและมีข้อจำกัดในการรองรับเทคโนโลยีใหม่

บทที่ 2
เอกสารและงานวิจัยที่เกี่ยวข้อง

2.1 ปัญหาคุณภาพข้อมูลในระบบทะเบียนนักศึกษา

จากการศึกษาระบบทะเบียนนักศึกษาของมหาวิทยาลัยคริสเตียน จังหวัดนครปฐม พบว่าระบบทะเบียนนักศึกษาเดิม (Legacy Systems) มีข้อจำกัดในการจัดเก็บและบริหารจัดการข้อมูล ซึ่งส่งผลให้เกิดปัญหาด้านคุณภาพข้อมูล เช่น ข้อมูลซ้ำซ้อน (ส่งผลต่อ Consistency), ข้อมูลขาดหาย (Completeness) และข้อมูลที่ไม่เป็นมาตรฐาน (Validity) ซึ่งล้วนเป็นองค์ประกอบสำคัญของมิติคุณภาพข้อมูลตามแนวทางของ Wang & Strong (1996) และ DAMA International (2017)

ปัญหาเหล่านี้ส่งผลกระทบโดยตรงต่อความถูกต้องของข้อมูล การวิเคราะห์ และการตัดสินใจของฝ่ายบริหาร งานวิจัยหลายฉบับ เช่น Batini et al. (2009) และ Woo et al. (2019) ได้เสนอแนวทางการปรับปรุงคุณภาพข้อมูลผ่านกระบวนการ Data Cleansing เพื่อลดข้อผิดพลาด และเพิ่มความน่าเชื่อถือของข้อมูลในการใช้งานเชิงระบบ

2.2 แนวทางการปรับปรุงคุณภาพข้อมูลด้วยกระบวนการ Data Cleansing

2.2.1 การทำความสะอาดข้อมูลด้วยโมเดล SDCM Al-Madi et al. (2023) ได้นำเสนอแนวทางการทำความสะอาดข้อมูลด้วยโมเดล Supervised Dataset Cleaning Model (SDCM) ซึ่งออกแบบมาเพื่อลดข้อผิดพลาดในข้อมูลประเภท Dirty Data โดยอาศัยการเรียนรู้ของเครื่องแบบมีผู้สอน (Supervised Learning) โดยใช้ชุดข้อมูลที่มีการระบุข้อผิดพลาดล่วงหน้า (Labeled Data) ในการฝึกโมเดล จากนั้นนำโมเดลมาวิเคราะห์ข้อมูลใหม่ที่ยังไม่มีป้ายกำกับ เพื่อระบุและแก้ไขข้อผิดพลาดที่เกิดขึ้น

ผู้วิจัยใช้เทคนิค K-Nearest Neighbors (KNN) ในการประเมินความคล้ายของข้อมูล เพื่อตรวจจับ outlier เติมข้อมูลที่ขาดหาย และกำจัดค่าผิดปกติและข้อมูลซ้ำซ้อน ผลการทดลองแสดงว่าโมเดลดังกล่าวสามารถเพิ่มความถูกต้องและความสมบูรณ์ของข้อมูลได้อย่างมีนัยสำคัญ ซึ่งอาจช่วยเพิ่มความแม่นยำในการประมวลผลและการตัดสินใจของระบบ

2.2.2 การจัดกลุ่มข้อความและการแปลงค่าโดยใช้ OpenRefine Woo et al. (2019) ได้เสนอแนวทางการทำความสะอาดข้อมูลกึ่งโครงสร้างในรายงานทางการแพทย์ โดยใช้เทคนิค Text Clustering และ Value Conversion ผ่านเครื่องมือ OpenRefine 2.7 เพื่อจัดการกับข้อผิดพลาดจากข้อมูลซ้ำที่มีรูปแบบแตกต่างกัน ข้อผิดพลาดจากการพิมพ์ และความไม่สอดคล้องกันของข้อมูล

กระบวนการทำงานแบ่งเป็น 4 ขั้นตอน ได้แก่

(1) การเตรียมข้อมูลเบื้องต้นด้วย STATA และ Regular Expressions (Regex)

(2) การทำ Text Faceting เพื่อวิเคราะห์คำที่ใช้บ่อย

(3) การใช้ Text Clustering และ Value Merging ใน OpenRefine เพื่อรวมกลุ่มข้อมูลที่คล้ายกัน

(4) การตรวจสอบแบบแมนนวลเพิ่มเติม ผลการทดลองของผู้วิจัยพบว่าวิธีการดังกล่าวสามารถแก้ไขข้อผิดพลาดได้สูงถึง 98.61% โดยเฉพาะข้อผิดพลาดทางการพิมพ์ และสามารถปรับรูปแบบข้อมูลได้อย่างแม่นยำถึง 97.78%

2.2.3 การประยุกต์ใช้ AI กับข้อมูลเชิงสัมพันธ์ (Relational Data) Zhu et al. (2024) ได้ศึกษาแนวทางการทำ Data Cleansing ในฐานข้อมูลเชิงสัมพันธ์ (Relational Database) โดยใช้เทคโนโลยีปัญญาประดิษฐ์ (AI) เพื่อเพิ่มความยืดหยุ่นและความสามารถในการจัดการข้อมูลที่ซับซ้อน วิธีที่เสนอแบ่งออกเป็น 3 กลุ่ม ได้แก่ การตรวจจับข้อผิดพลาด (Error Detection), การซ่อมแซมข้อมูล (Data Repairing), และการเติมข้อมูลที่ขาดหาย (Data Imputation)

แนวทางดั้งเดิมจะมีข้อดีในด้านความเข้าใจง่ายและต้นทุนต่ำ แต่ผู้วิจัยรายงานว่า AI มีประสิทธิภาพสูงกว่าในการจัดการข้อมูลขนาดใหญ่ โดยเฉพาะเมื่อใช้เทคนิค Deep Learning อย่างไรก็ตาม งานวิจัยยังระบุข้อจำกัดของ AI ได้แก่ ความต้องการข้อมูลฝึกจำนวนมาก และความยากในการตีความผลลัพธ์ของโมเดล

2.2.4 การทำความสะอาดข้อมูลด้วยแนวทาง Workflow-Based Cleaning Guo et al. (2023) ได้นำเสนอแนวทางการทำ Data Cleaning แบบ Workflow-Based โดยเน้นการระบุข้อผิดพลาด แยกประเภทปัญหา และดำเนินการตามขั้นตอนที่ชัดเจนตามตรรกะและกฎที่กำหนดไว้ล่วงหน้า (Rule-Based Approaches)

แนวทางนี้สามารถผสานร่วมกับกระบวนการอื่น เช่น การเติมข้อมูลหรือการตรวจสอบความสอดคล้อง ผู้วิจัยเสนอว่าแนวทางดังกล่าวเหมาะกับข้อมูลที่มีรูปแบบไม่ซับซ้อน เช่น ข้อมูลทะเบียน หรือข้อมูลการบริหารจัดการในองค์กรที่ไม่สามารถนำ Machine Learning มาใช้ได้อย่างเต็มประสิทธิภาพ

2.2.5 แนวทางการจัดการข้อมูลขนาดใหญ่ (Big Data Cleansing) Hosseinzadeh et al. (2023) ได้ศึกษาแนวทางการทำความสะอาดข้อมูลในบริบทของ Big Data โดยมุ่งเน้นกลไกที่เหมาะสมสำหรับจัดการข้อมูลที่มีปริมาณมาก ความหลากหลายสูง และต้องประมวลผลแบบ Real-Time

ผู้วิจัยเสนอแนวทางที่หลากหลาย เช่น Machine Learning-Based Cleaning, Rule-Based Cleaning และ Sample-Based Cleaning โดยเน้นให้เลือกใช้ตามลักษณะของข้อมูลและข้อจำกัดของระบบ

2.2.6 ซอฟต์แวร์ DataAssist และการใช้ ML เพื่อเตรียมข้อมูลอัตโนมัติ Goyle et al. (2024) ได้พัฒนาซอฟต์แวร์ชื่อ DataAssist ซึ่งใช้ Machine Learning เพื่อช่วยในกระบวนการเตรียมข้อมูล (Data Preparation) และการทำ Data Cleaning แบบกึ่งอัตโนมัติ โดยมีฟังก์ชันในการตรวจสอบคุณภาพ วิเคราะห์ค่าผิดปกติ เติมข้อมูลที่ขาดหาย และปรับรูปแบบข้อมูลให้สอดคล้องกัน

ผู้วิจัยรายงานว่าการใช้ DataAssist สามารถลดระยะเวลาการทำ Data Cleaning ได้มากกว่า 50% และเพิ่มความแม่นยำของข้อมูล ทั้งนี้ซอฟต์แวร์ดังกล่าวได้รับการออกแบบมาเพื่อรองรับผู้ใช้งานที่ไม่มีความเชี่ยวชาญด้านการเขียนโปรแกรม

2.2.7 การประเมินวิธีการ Supervised และ Unsupervised ในการจัดการ Label ผิดพลาด Khamket et al. (2025) ได้ศึกษาการปรับปรุงคุณภาพข้อมูลในบริบทของการวิเคราะห์ข้อความ (Sentiment Classification) โดยมุ่งเน้นที่ปัญหาความคลาดเคลื่อนของป้ายกำกับ (Label Noise) ซึ่งเป็นหนึ่งในรูปแบบของข้อมูลผิดพลาดในชุดข้อมูลจริง งานวิจัยเปรียบเทียบประสิทธิภาพระหว่างวิธีแบบ Supervised และ Unsupervised ในการตรวจจับและจัดการ Label ที่ผิดพลาดในชุดข้อมูลขนาดใหญ่

ในเชิงเทคนิค วิธีแบบ Supervised ที่นำมาใช้ ได้แก่ Support Vector Machine (SVM) และ Convolutional Neural Network (CNN) ส่วนวิธีแบบ Unsupervised ใช้อัลกอริทึมที่ไม่ต้องการข้อมูลป้ายกำกับ โดยใช้หลักการตรวจจับความผิดปกติเชิงสถิติ ผลการทดลองพบว่าวิธีแบบ Supervised ให้ความแม่นยำสูงกว่าอย่างชัดเจน โดยสามารถลดข้อผิดพลาดของป้ายกำกับได้อย่างมีนัยสำคัญ

แม้ว่างานวิจัยนี้จะอยู่ในบริบทของข้อมูลประเภทข้อความ (Textual Data) แต่แนวคิดเรื่องการประเมินวิธีการเรียนรู้เพื่อจัดการข้อมูลที่มีความผิดพลาดในระดับป้ายกำกับ สามารถประยุกต์ใช้กับบริบทของการทำ Data Cleansing ในข้อมูลเชิงโครงสร้างได้เช่นกัน โดยเฉพาะเมื่อมีความจำเป็นต้องตรวจจับข้อมูลที่อาจมีค่าผิดหรือค่าขัดแย้งกับบริบทที่ควรจะเป็น

2.3 สรุปงานวิจัยที่เกี่ยวข้อง

จากการศึกษางานวิจัยต่าง ๆ พบว่าแนวทางในการดำเนินการ Data Cleansing มีความหลากหลายทั้งในเชิงเทคนิคและเครื่องมือ โดยในเชิงเทคนิคมีการนำเสนอวิธีการ เช่น Rule-Based, Text Clustering, Workflow-Based และ Machine Learning-Based ในขณะที่เชิงเครื่องมือมีการใช้ซอฟต์แวร์เฉพาะทาง เช่น OpenRefine และ DataAssist เพื่อสนับสนุนกระบวนการทำความสะอาดข้อมูลให้มีประสิทธิภาพยิ่งขึ้น

การเลือกใช้แนวทางใดขึ้นอยู่กับลักษณะของข้อมูล โครงสร้างระบบสารสนเทศที่เกี่ยวข้อง และทรัพยากรขององค์กร งานวิจัยที่ได้ศึกษาชี้ให้เห็นว่าการจัดการข้อมูลที่มีปัญหา เช่น ข้อมูลซ้ำซ้อน

ข้อมูลขาดหาย หรือข้อมูลที่ไม่เป็นมาตรฐานนั้นจำเป็นต้องใช้เทคนิคที่หลากหลายและเหมาะสมกับปัญหาเฉพาะของแต่ละบริบท เพื่อให้สามารถยกระดับคุณภาพข้อมูลได้อย่างครอบคลุม

ตาราง 1 สรุปงานวิจัยที่เกี่ยวข้อง

ลำดับ	ผู้วิจัย (ปี)	เทคนิคที่นำเสนอ	ประเด็นที่เกี่ยวข้องกับการ Cleansing
1	Al-Madi et al. (2023)	SDCM + K-Nearest Neighbors	Supervised Learning สำหรับตรวจจับและแก้ Dirty Data แบบอัตโนมัติ
2	Woo et al. (2019)	OpenRefine + Text Clustering	การจัดกลุ่มข้อมูลข้อความและตรวจสอบความคล้ายของค่า
3	Zhu et al. (2024)	AI-Based Cleansing for Relational DB	ใช้ AI และ Deep Learning จัดการข้อผิดพลาดในฐานข้อมูลเชิงสัมพันธ์
4	Guo et al. (2023)	Workflow-Based Rule Cleaning	การออกแบบขั้นตอนทำ Cleansing แบบตรรกะชัดเจน เหมาะกับข้อมูลไม่ซับซ้อน
5	Hosseinzadeh et al. (2023)	Big Data Cleansing Framework	แนวทางจัดการข้อมูลปริมาณมหาศาลโดยเลือกเทคนิคตามลักษณะข้อมูล
6	Goyle et al. (2024)	DataAssist (ML-based Cleaning Tool)	ซอฟต์แวร์ช่วยเตรียมข้อมูลแบบกึ่งอัตโนมัติโดยไม่ต้องใช้โค้ด
7	Khamket et al. (2025)	Supervised vs Unsupervised Label Repair	เปรียบเทียบวิธีจัดการ Label ผิดพลาดเพื่อใช้ในบริบทข้อมูลเชิงโครงสร้าง

บทที่ 3
ทฤษฎีที่เกี่ยวข้อง

3.1 แนวคิดคุณภาพข้อมูล (Data Quality)

คุณภาพของข้อมูล (Data Quality) หมายถึงระดับที่ข้อมูลสามารถตอบสนองต่อความต้องการของผู้ใช้งานได้อย่างมีประสิทธิภาพ ถูกต้อง และเชื่อถือได้ Wang & Strong (1996) ได้นำเสนอกรอบแนวคิดที่แบ่งมิติของคุณภาพข้อมูลออกเป็น 15 มิติ จำแนกเป็น 4 กลุ่ม ได้แก่

3.1.1 คุณสมบัติของข้อมูลในตัวมันเอง (Intrinsic Data Quality)

3.1.1.1 ความถูกต้อง (Accuracy)

3.1.1.2 ความเป็นกลาง (Objectivity)

3.1.1.3 ความน่าเชื่อถือ (Believability)

3.1.1.4 ความน่าเชื่อถือของแหล่งข้อมูล (Reputation)

3.1.2 ความเหมาะสมของข้อมูลต่อบริบทที่ใช้งาน (Contextual Data Quality)

3.1.2.1 ความเกี่ยวข้อง (Relevancy)

3.1.2.2 มีคุณค่าในการใช้งาน (Value-added)

3.1.2.3 ความเป็นปัจจุบัน (Timeliness)

3.1.2.4 ความสมบูรณ์ (Completeness)

3.1.2.5 ปริมาณข้อมูลที่เหมาะสม (Appropriate Amount of Data)

3.1.3 ความเข้าใจและการแสดงผลข้อมูล (Representational Data Quality)

3.1.3.1 ตีความได้ (Interpretability)

3.1.3.2 เข้าใจง่าย (Ease of understanding)

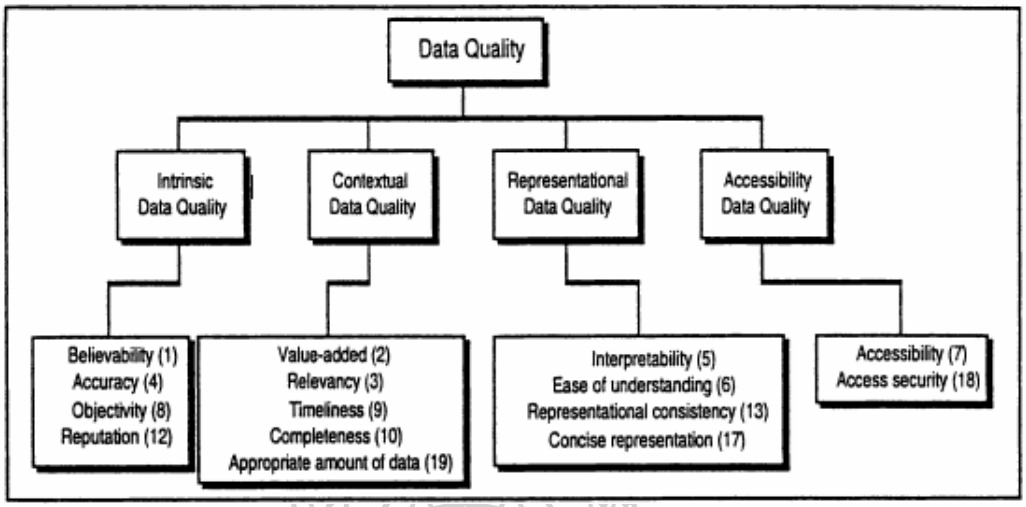
3.1.3.3 การแสดงผลกระชับ (Concise representation)

3.1.3.4 การแสดงผลสอดคล้องกัน (Consistent representation)

3.1.4 ความสามารถในการเข้าถึง (Accessibility Data Quality)

3.1.4.1 Accessibility (เข้าถึงได้ง่าย)

3.1.4.2 Security (มีความปลอดภัยในการเข้าถึง)



ภาพที่ 3 มิติของคุณภาพข้อมูลจำแนกตามกลุ่ม

ที่มา Wang & Strong (1996) หน้า 20

Batini et al. (2009) ได้เสนอเพิ่มเติมว่าคุณภาพของข้อมูลสามารถประเมินได้ทั้งในเชิงเทคนิคและเชิงกระบวนการ โดยเน้นว่าการประเมินคุณภาพข้อมูล (Data Quality Assessment) เป็นกระบวนการที่จำเป็นในการควบคุมและดูแลข้อมูล โดยเฉพาะในบริบทที่ข้อมูลถูกนำไปใช้เพื่อสนับสนุนการตัดสินใจขององค์กร

ในงานวิจัยนี้จะมุ่งเน้นมิติที่เกี่ยวข้องกับ ความถูกต้อง (Accuracy), ความสมบูรณ์ (Completeness), ความสอดคล้องกัน (Consistency) และ ความเป็นปัจจุบัน (Timeliness) ซึ่งเป็นเกณฑ์หลักที่ใช้ในการประเมินคุณภาพข้อมูลก่อนและหลังกระบวนการทำ Data Cleansing

3.2 กระบวนการทำความสะอาดข้อมูล (Data Cleansing)

Rahm and Do (2000) ได้เสนอแนวทางการจัดการข้อมูลที่ผิดพลาดหรือไม่สอดคล้องกันผ่านกระบวนการที่เรียกว่า *Data Cleaning* ซึ่งเป็นส่วนสำคัญของการเตรียมข้อมูล (*Data Preparation*) โดยเฉพาะเมื่อข้อมูลมาจากหลายแหล่งที่มีโครงสร้างและมาตรฐานต่างกัน โดยกระบวนการดังกล่าวประกอบด้วย 5 ขั้นตอนหลัก ดังนี้

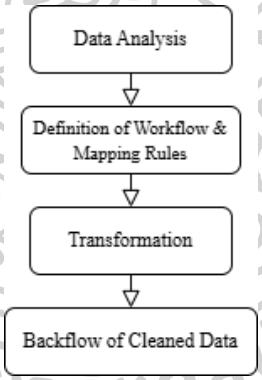
3.2.1 การวิเคราะห์ข้อมูล (Data Analysis) ขั้นตอนแรกของการทำ Data Cleaning คือการทำ Data Profiling และการวิเคราะห์ข้อมูลเพื่อระบุปัญหา เช่น ความไม่สมบูรณ์ ความไม่สอดคล้องหรือข้อผิดพลาดในการป้อนข้อมูล กระบวนการนี้ช่วยให้สามารถออกแบบกฎการแปลงข้อมูล (Transformation Rules) ได้แม่นยำยิ่งขึ้น

3.2.2 การกำหนดเวิร์กโฟลว์และกฎการแปลงข้อมูล (Definition of Transformation Workflow and Mapping Rules) ในขั้นตอนนี้จะมีการกำหนดลำดับของการประมวลผลข้อมูล (Workflow) และสร้าง Mapping Rules ที่จำเป็นต่อการแปลงข้อมูลจากรูปแบบเดิมให้เป็นมาตรฐาน โดยกฎที่ใช้ควรอยู่ในรูปแบบที่สามารถนำกลับมาใช้ซ้ำได้ (Reusable)

3.2.3 การตรวจสอบความถูกต้อง (Verification) ก่อนนำกฎไปใช้จริง ระบบจะต้องมีการทดสอบและประเมินคุณภาพของ Transformation Workflow ที่สร้างขึ้นเพื่อให้มั่นใจว่าข้อมูลที่ถูกแปลงจะไม่สูญเสียความหมายและไม่มีข้อผิดพลาดเกิดขึ้นในกระบวนการ

3.2.4 การดำเนินการแปลงข้อมูล (Transformation) หลังจากผ่านการตรวจสอบ กฎที่สร้างไว้จะถูกนำไปใช้จริงกับชุดข้อมูลที่มีปัญหา โดยจะมีการดำเนินการแก้ไข ปรับปรุง และแปลงข้อมูลให้เป็นไปตามรูปแบบที่กำหนด

3.2.5 การส่งข้อมูลที่สะอาดกลับเข้าสู่ระบบ (Backflow of Cleaned Data) เมื่อข้อมูลได้รับการทำความสะอาดแล้ว จะมีการนำข้อมูลที่ผ่านกระบวนการนี้กลับเข้าสู่ระบบหลักหรือจัดเก็บไว้ในระบบสำรอง เพื่อให้สามารถนำไปใช้งานในขั้นตอนถัดไปของระบบสารสนเทศได้



ภาพที่ 4 ขั้นตอนการทำ Data Cleaning

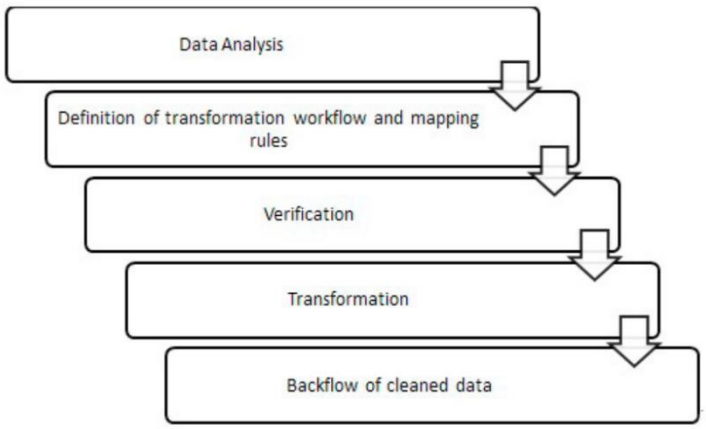
ที่มา: สร้างขึ้นจากแนวคิดใน Rahm and Do (2000)

นอกจากแนวทางของ Rahm and Do (2000) แล้ว Ridzuan and Wan Zainon (2019) ยังได้เสนอขั้นตอนการทำ Data Cleansing ที่เน้นลำดับขั้นตอนเชิงกระบวนการ ซึ่งเหมาะกับการจัดการข้อมูลขนาดใหญ่และสามารถนำมาประยุกต์ใช้ในระบบฐานข้อมูลที่มีความซับซ้อน โดยจำแนกเป็น 5 ขั้นตอน ได้แก่

1. การวิเคราะห์ข้อมูลเบื้องต้น (Data Analysis)
2. การกำหนดกฎการแปลงและ mapping rules (Define Transformation Workflow and Mapping Rule)

3. การตรวจสอบกฎและความถูกต้อง (Verification)
4. การแปลงข้อมูล (Transformation)
5. การส่งข้อมูลที่ผ่านการ Cleansing กลับเข้าสู่ระบบ (Backflow of Cleaned Data)

เมื่อเปรียบเทียบกับแนวทางของ Rahm and Do (2000) ที่มุ่งเน้นประเด็นด้านเทคนิคของการจัดการข้อมูลจากหลายแหล่ง Ridzuan and Wan Zainon (2019) ได้ขยายมุมมองให้ครอบคลุมการควบคุม workflow เชิงธุรกิจ และการนำข้อมูลที่สะอาดกลับเข้าสู่ระบบเพื่อใช้งานจริง ซึ่งเป็นประเด็นที่สำคัญในบริบทของการปรับปรุงคุณภาพข้อมูลภายในองค์กร



ภาพที่ 5 ขั้นตอนการทำ Data Cleaning

ที่มา: Ridzuan and Wan Zainon (2019) หน้า 733

3.3 แนวทาง Rule-Based Data Cleaning

สำหรับข้อมูลที่มีโครงสร้างแน่นอน เช่น ข้อมูลทะเบียนนักศึกษา การใช้แนวทาง Rule-Based Data Cleaning ถือเป็นทางเลือกที่เหมาะสมและสามารถควบคุมคุณภาพข้อมูลได้อย่างเป็นระบบ โดยอาศัยกฎหรือเงื่อนไขที่กำหนดไว้ล่วงหน้าเพื่อตรวจสอบความถูกต้อง ความสมเหตุสมผล และความสอดคล้องของข้อมูล

Borkar et al. (2001) ได้นำเสนอแนวคิดในการจัดการกับข้อมูลข้อความโดยอัตโนมัติ โดยการแบ่งข้อความออกเป็นหน่วยข้อมูลที่มีโครงสร้างผ่านเทคนิคการจัดกลุ่มข้อความและกฎเชิงตรรกะ ซึ่งสามารถนำไปใช้ในการจัดการข้อมูลที่มีแบบแผนแน่นอน เช่น ข้อมูลจากฟอร์ม หรือเอกสารโครงสร้างคงที่ จากแนวคิดนี้สามารถนำมาประยุกต์ใช้กับระบบทะเบียนนักศึกษา ซึ่งเป็นระบบที่ต้องอาศัยข้อมูลที่ถูกต้องและมีความสอดคล้องกันในหลายมิติ โดยสามารถออกแบบกฎการตรวจสอบได้หลากหลายรูปแบบ เช่น

3.3.1 การตรวจสอบความถูกต้องของข้อมูล (Data Validation) การกำหนดกฎเพื่อระบุว่าข้อมูลที่ป้อนเข้ามามีรูปแบบที่ถูกต้อง เช่น เบอร์โทรศัพท์ต้องมี 10 หลัก หรือวันที่ต้องอยู่ในรูปแบบ yyyy-mm-dd

3.3.2 การตรวจสอบความสอดคล้องของข้อมูล (Consistency Check) การระบุความสัมพันธ์ระหว่างข้อมูล เช่น นักศึกษาที่ลงทะเบียนเรียนในรายวิชาหนึ่งจะต้องมีรหัสรายวิชานั้นในฐานข้อมูลรายวิชา

3.3.3 การจัดรูปแบบข้อมูลให้เป็นมาตรฐาน (Standardization) เช่น การแปลงชื่อจังหวัดให้มีรูปแบบเดียวกัน หรือการปรับชื่อย่อให้อยู่ในรูปแบบเต็มที่เหมือนกัน

แนวทาง Rule-Based ดังกล่าวช่วยลดข้อผิดพลาดที่เกิดจากการป้อนข้อมูลผิด ลดภาระของเจ้าหน้าที่ในการตรวจสอบข้อมูลด้วยตนเอง และสามารถประยุกต์ใช้ร่วมกับระบบตรวจสอบอัตโนมัติเพื่อเพิ่มความแม่นยำในการจัดการข้อมูลได้อย่างมีประสิทธิภาพ

3.4 การจัดกลุ่มข้อความ (Text Clustering)

การจัดกลุ่มข้อความ (Text Clustering) เป็นเทคนิคหนึ่งในการปรับปรุงคุณภาพของข้อมูล โดยเฉพาะข้อมูลที่มีลักษณะเป็นข้อความที่ไม่มีโครงสร้างแน่นอน หรือข้อมูลกึ่งโครงสร้าง เช่น ชื่อบุคคล ที่อยู่ หรือชื่อหน่วยงาน ซึ่งอาจมีการสะกดหรือเขียนไม่ตรงกัน แม้จะหมายถึงสิ่งเดียวกัน

Woo et al. (2019) ได้นำเสนอการประยุกต์ใช้ Text Clustering ร่วมกับกระบวนการ Value Conversion ผ่านเครื่องมือ OpenRefine 2.7 เพื่อทำความสะอาดรายงานทางการแพทย์ขนาดใหญ่ที่อยู่ในรูปแบบกึ่งโครงสร้าง โดยมีเป้าหมายในการลดข้อผิดพลาดที่เกิดจากความหลากหลายของการป้อนข้อมูล เช่น คำสะกดที่แตกต่างกัน การใช้คำย่อ หรือความไม่สอดคล้องของรูปแบบข้อความ

ในงานวิจัยดังกล่าว มีการใช้เทคนิคหลัก 2 แนวทาง คือ

3.4.1 Key Collision การจัดกลุ่มข้อความโดยใช้อัลกอริทึมเปรียบเทียบเสียงหรือโครงสร้างของคำ เช่น Metaphone และ Soundex

3.4.2 Nearest Neighbor การเปรียบเทียบข้อความด้วยระยะห่างของ string (string distance) เพื่อตรวจจับความคล้ายและจัดกลุ่มข้อความที่เกี่ยวข้อง

3.4.2.1 กระบวนการทำความสะอาดประกอบด้วย 4 ขั้นตอน

3.4.2.1 การเตรียมข้อมูลเบื้องต้น ด้วย STATA และ Regular Expressions (Regex)

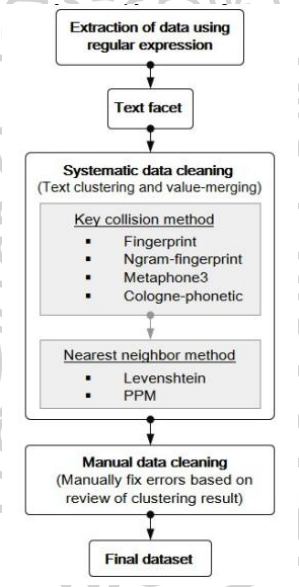
3.4.2.1 การวิเคราะห์ข้อความ (Text Faceting) เพื่อระบุข้อความที่มีลักษณะซ้ำหรือใกล้เคียง

3.4.2.1 การจัดกลุ่มและแปลงค่า (Text Clustering and Value Merging) ด้วย OpenRefine

3.4.2.1 การปรับปรุงข้อมูลแบบแมนนวล (Manual Cleaning) เพื่อเสริมความถูกต้องในขั้นสุดท้าย

จากผลการทดลองพบว่า วิธีการดังกล่าวสามารถปรับปรุงความถูกต้องของข้อมูลได้อย่างมีประสิทธิภาพ โดยมีอัตราการแก้ไขข้อผิดพลาดทางการพิมพ์สูงถึง 98.61% และสามารถปรับปรุงรูปแบบข้อมูลผิดพลาดได้ถึง 97.78%

สำหรับการประยุกต์ใช้ในระบบทะเบียนนักศึกษา เทคนิค Text Clustering สามารถช่วยรวมค่าข้อมูลที่คล้ายกัน เช่น ชื่อบุคคลที่สะกดแตกต่างกัน หรือชื่อจังหวัดที่มีทั้งแบบย่อและแบบเต็ม ให้มีรูปแบบที่สอดคล้องกัน ช่วยลดความคลาดเคลื่อนของข้อมูล และเพิ่มความแม่นยำของระบบโดยรวม



ภาพที่ 6 กระบวนการทำความสะอาดข้อมูลโดยใช้ Text Clustering

ที่มา: Woo et al. (2019)

3.5 การประยุกต์ใช้ Machine Learning ใน Data Cleansing

การประยุกต์ใช้เทคนิค Machine Learning (ML) ในการทำความสะอาดข้อมูล (Data Cleansing) ได้รับความสนใจมากขึ้นในช่วงหลัง เนื่องจากความสามารถของ ML ในการจัดการข้อมูลขนาดใหญ่และตรวจจับความผิดปกติที่ซับซ้อนซึ่งอาจไม่สามารถมองเห็นได้จากการวิเคราะห์แบบเดิม

Al-Madi et al. (2023) ได้เสนอโมเดลที่มีชื่อว่า Supervised Dataset Cleaning Model (SDCM) ซึ่งใช้การเรียนรู้แบบมีผู้สอน (Supervised Learning) โดยอาศัยชุดข้อมูลที่มีการระบุ

ข้อผิดพลาดไว้แล้วในการฝึกโมเดล จากนั้นนำโมเดลไปใช้กับข้อมูลใหม่เพื่อค้นหาและแก้ไขข้อผิดพลาด โมเดลนี้ใช้ K-Nearest Neighbors (KNN) ในการวิเคราะห์ความใกล้เคียงของข้อมูล เพื่อระบุค่าที่ผิดปกติ (outliers) เติมค่าที่ขาดหายไป และตรวจจับข้อผิดพลาดเชิงโครงสร้างในข้อมูล

Zhu et al. (2024) ได้เสนอกรอบแนวคิดที่รวมแนวทางการใช้ AI และ ML ในการทำ Data Cleansing โดยแบ่งการประยุกต์ใช้ ML ออกเป็น 3 ลักษณะหลัก ได้แก่ (1) การตรวจจับข้อผิดพลาด (Error Detection), (2) การซ่อมแซมข้อมูล (Data Repairing), และ (3) การเติมข้อมูลที่ขาดหาย (Data Imputation) แนวคิดนี้สามารถใช้ได้กับข้อมูลที่มีความซับซ้อน เช่น ฐานข้อมูลเชิงสัมพันธ์ (Relational Data) ที่มีข้อผิดพลาดในระดับระหว่างตารางหรือระดับโครงสร้าง

Goyle et al. (2024) ได้นำเสนอซอฟต์แวร์ DataAssist ซึ่งใช้เทคนิค ML ในการเตรียมและทำความสะอาดข้อมูลอย่างกึ่งอัตโนมัติ โดยมีฟังก์ชัน เช่น การวิเคราะห์ข้อมูลเบื้องต้น (Exploratory Data Analysis: EDA), การเติมค่าที่ขาดหายไป (Missing Value Imputation), การตรวจจับและลบค่าผิดปกติ (Outlier Detection & Removal), และการลบข้อมูลซ้ำซ้อน (Duplicate Removal) จุดเด่นของ DataAssist คือช่วยลดระยะเวลาในการทำ Data Cleansing ได้มากกว่า 50% เมื่อเทียบกับวิธีแบบแมนนวล และสามารถปรับรูปแบบข้อมูลให้สอดคล้องกันโดยอัตโนมัติ

แม้ว่าการใช้ Machine Learning จะมีข้อได้เปรียบด้านความแม่นยำ ความเร็ว และความสามารถในการเรียนรู้จากข้อมูลขนาดใหญ่ แต่ก็มีข้อจำกัด เช่น ความต้องการชุดข้อมูลสำหรับฝึกที่มีคุณภาพสูง ความซับซ้อนของการตั้งค่าโมเดล และความยากในการตีความผลลัพธ์ของบางโมเดล เช่น Deep Learning สำหรับระบบทะเบียนนักศึกษา การนำ Machine Learning มาใช้กับกระบวนการทำ Data Cleansing อาจเหมาะสมกับบางประเภทของข้อมูล เช่น การเติมค่าที่ขาดหายของช่องทางการติดต่อ หรือการตรวจจับค่าผิดปกติจากข้อมูลที่มีรูปแบบซ้ำ ๆ อย่างไรก็ตาม ในกรณีที่ต้องการความแม่นยำสูงและมีโครงสร้างข้อมูลชัดเจน แนวทาง Rule-Based หรือ Text Clustering อาจยังคงเหมาะสมกว่าในเชิงการควบคุม

ดังนั้น การประยุกต์ใช้ Machine Learning ในงาน Data Cleansing ควรอยู่ภายใต้การพิจารณาด้านทรัพยากร ความซับซ้อนของข้อมูล และเป้าหมายของการใช้งานจริง ในกรณีของข้อมูลทะเบียนนักศึกษาที่มีโครงสร้างชัดเจนและความต้องการควบคุมผลลัพธ์อย่างแม่นยำ แนวทางการใช้ Rule-Based หรือ Text Clustering อาจเหมาะสมกว่า อย่างไรก็ตาม การนำ ML มาใช้ในบางส่วนของกระบวนการ เช่น การเติมค่าที่ขาดหายหรือการวิเคราะห์ข้อมูลเบื้องต้น อาจช่วยเสริมประสิทธิภาพของระบบโดยรวมได้อย่างมีนัยสำคัญ

3.6 สรุปการประยุกต์ใช้ทฤษฎีในงานวิจัยนี้

จากการศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้องในบทนี้ พบว่าแต่ละแนวคิดมีจุดแข็งที่สามารถนำมาประยุกต์ใช้ร่วมกันเพื่อเพิ่มประสิทธิภาพของกระบวนการทำ Data Cleansing ให้เหมาะสมกับบริบทของข้อมูลทะเบียนนักศึกษาในระบบสารสนเทศของมหาวิทยาลัย โดยสรุปแนวทางที่นำมาใช้ในงานวิจัยนี้ มีดังนี้

3.6.1 แนวคิดคุณภาพข้อมูล (Data Quality) อ้างอิงจาก Wang & Strong (1996) และ Batini et al. (2009) ซึ่งเสนอกรอบการประเมินคุณภาพข้อมูลใน 4 มิติหลัก ได้แก่ ความถูกต้อง (Accuracy), ความสมบูรณ์ (Completeness), ความสอดคล้องกัน (Consistency) และความเป็นปัจจุบัน (Timeliness) ซึ่งจะถูกใช้เป็นเกณฑ์ในการวิเคราะห์คุณภาพข้อมูลก่อนและหลังการทำ Data Cleansing

3.6.2 กระบวนการทำ Data Cleansing ยึดแนวทางของ Rahm and Do (2000) ซึ่งประกอบด้วย 5 ขั้นตอนหลัก ได้แก่ Data Profiling, Rule Definition, Verification, Transformation และ Backflow โดยขั้นตอนเหล่านี้ช่วยให้การทำความสะอาดข้อมูลมีโครงสร้างชัดเจนและสามารถตรวจสอบย้อนกลับได้

3.6.3 เทคนิค Rule-Based Data Cleaning ใช้เป็นแนวทางหลักตามแนวคิดของ Borkar et al. (2001) ซึ่งเหมาะสมกับข้อมูลที่มีโครงสร้างแน่นอน เช่น ข้อมูลทะเบียนนักศึกษา สามารถกำหนดกฎตรรกะเพื่อตรวจสอบและจัดรูปแบบข้อมูลให้สอดคล้องกัน

3.6.4 เทคนิค Text Clustering เสริมการทำงานด้วยแนวทางของ Borkar et al. (2001) โดยใช้สำหรับข้อมูลกึ่งโครงสร้าง เช่น ชื่อบุคคลหรือสถานที่ที่อาจสะกดต่างกัน เพื่อจัดกลุ่มและปรับข้อมูลให้มีความสอดคล้องมากขึ้น

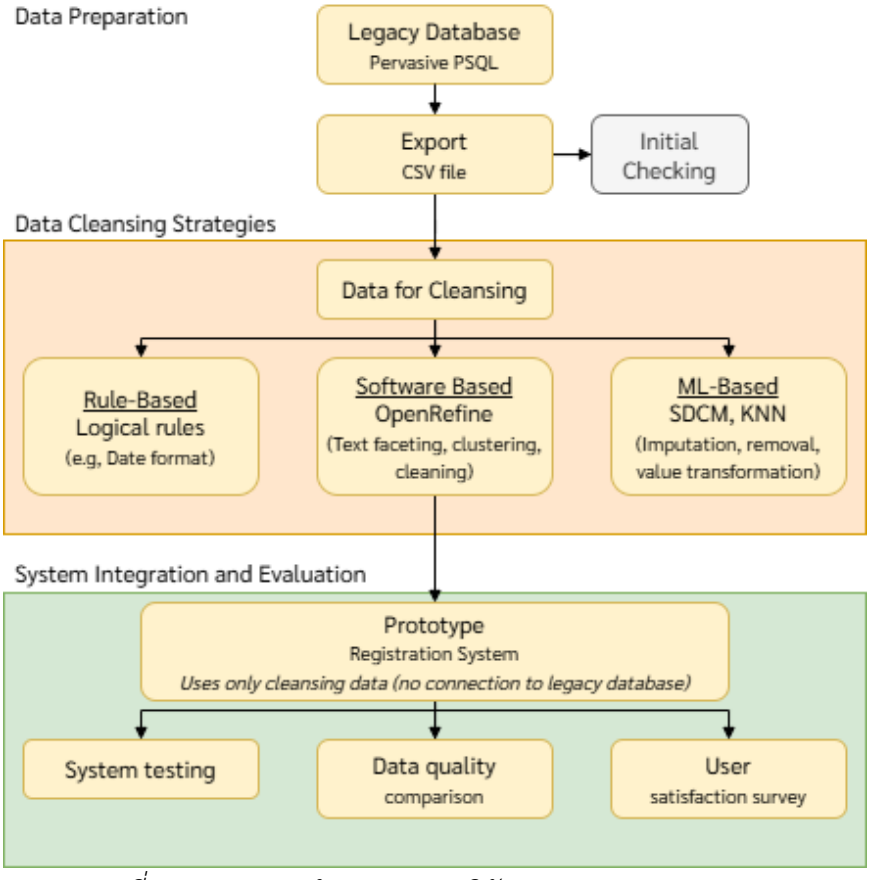
3.6.5 การประยุกต์ใช้ Machine Learning ศึกษาความเป็นไปได้ในการนำแนวทางจาก Al-Madi et al. (2023), Zhu et al. (2024) และ Goyle et al. (2024) มาใช้ เพื่อจัดการกับข้อมูลที่มีความซับซ้อน อย่างไรก็ตาม เนื่องจากข้อมูลทะเบียนมีลักษณะเฉพาะและต้องการการควบคุมอย่างแม่นยำ จึงเลือกใช้ Machine Learning ในบางกรณี เช่น การเติมข้อมูลที่ขาดหาย หรือการตรวจจับค่าผิดปกติ โดยไม่ใช้เป็นแนวทางหลัก

โดยสรุป งานวิจัยนี้เลือกใช้แนวทาง Rule-Based และ Text Clustering เป็นแกนหลักในการทำ Data Cleansing และประยุกต์ใช้ Machine Learning เฉพาะในบริบทที่เหมาะสม เพื่อให้สามารถจัดการข้อมูลทะเบียนนักศึกษาได้อย่างมีประสิทธิภาพภายใต้ข้อจำกัดของระบบเดิม

บทที่ 4
วิธีการดำเนินงานวิจัย

บทนี้นำเสนอขั้นตอนการดำเนินงานของงานวิจัย ตั้งแต่การเตรียมข้อมูลดิบจากระบบฐานข้อมูลเดิม การออกแบบโครงสร้างข้อมูลใหม่และการจัดทำ Source-to-Target Mapping การดำเนินกระบวนการ Data Cleansing การออกแบบและพัฒนาระบบต้นแบบ ไปจนถึงการทดสอบและประเมินผล โดยมีวัตถุประสงค์เพื่อแสดงให้เห็นว่ากระบวนการทั้งหมดที่ออกแบบขึ้นสามารถยกระดับคุณภาพข้อมูลและรองรับการใช้งานจริงได้

เพื่อให้เห็นภาพรวมของลำดับขั้นตอนการดำเนินงานทั้งหมด งานวิจัยได้นำเสนอ System Framework ดังแสดงในรูปภาพ 7 ซึ่งสรุปกระบวนการวิจัยตั้งแต่การ Extract ข้อมูล > การ Mapping และ Cleansing > การพัฒนาระบบต้นแบบ > การประเมินผลลัพธ์



ภาพที่ 7 ระบวนการทำงานของงานวิจัย (Systsem Framework)

4.1 การเตรียมข้อมูลที่ใช้ในงานวิจัย

การวิจัยนี้ใช้ข้อมูลจากระบบฐานข้อมูลทะเบียนนักศึกษาเดิมของมหาวิทยาลัยคริสเตียน จังหวัดนครปฐม ซึ่งพัฒนาโดยใช้ระบบฐานข้อมูล Pervasive PSQL v11 ที่มีการใช้งานต่อเนื่องมาเป็นเวลานาน ระบบดังกล่าวมีข้อจำกัดหลายประการ เช่น ข้อมูลซ้ำซ้อน ข้อมูลไม่สมบูรณ์ หรือมีรูปแบบไม่เป็นมาตรฐาน จึงจำเป็นต้องนำข้อมูลที่มีอยู่มาผ่านกระบวนการตรวจสอบและปรับปรุงคุณภาพ (Data Cleansing) ก่อนนำไปใช้งานในระบบต้นแบบที่พัฒนาขึ้นใหม่ ในการดำเนินงาน ผู้วิจัยได้ทำการส่งออกข้อมูลจากฐานข้อมูลเดิมให้อยู่ในรูปแบบ CSV (Comma-Separated Values) โดยใช้เครื่องมือ Pervasive Control Center (PCC) จากนั้นตรวจสอบเบื้องต้นผ่านโปรแกรม Microsoft Excel เพื่อประเมินความครบถ้วน ความถูกต้องของฟิลด์ข้อมูล และจัดเตรียมสำหรับการดำเนินการ Cleansing อย่างเป็นระบบ

ก่อนการ Cleansing ผู้วิจัยได้ทำการศึกษาโครงสร้างความสัมพันธ์ของตารางข้อมูล (Entity Relationship Diagram: ERR) และจัดทำ Source-to-Target Mapping เพื่อกำหนดการแมปปิงระหว่างคอลัมน์ของระบบเดิมและระบบใหม่ ซึ่งเป็นพื้นฐานสำคัญสำหรับการถ่ายโอนข้อมูลอย่างถูกต้อง นอกจากนี้ ผู้วิจัยได้พัฒนาเครื่องมือ apply_mapping_gui.py สำหรับอ่านไฟล์ Source-to-Target Mapping (Excel) และแปลงข้อมูลจาก CSV เดิมไปเป็น CSV ใหม่ตามโครงสร้างที่ออกแบบใหม่โดยอัตโนมัติ เพื่อลดความผิดพลาดจากการทำงานด้วยมือ และทำให้สามารถทำซ้ำได้เมื่อจำเป็นต้องปรับปรุงข้อมูลใหม่

ทั้งนี้ ข้อมูลที่ใช้ในการวิจัยถูกแยกออกจากระบบปฏิบัติงานจริง (Production) อย่างชัดเจน และมีการจัดการด้านความปลอดภัยของข้อมูล (Data Security) และการปกป้องข้อมูลส่วนบุคคล (Data Privacy) เพื่อหลีกเลี่ยงผลกระทบต่อการดำเนินงานปกติขององค์กร การวิจัยนี้กำหนดขอบเขตการดำเนิน Data Cleansing เฉพาะข้อมูลที่จัดส่งให้หน่วยงานภายนอก เพื่อให้ได้ข้อมูลที่ถูกต้องครบถ้วน และมีมาตรฐานเดียวกัน อันจะเพิ่มประสิทธิภาพการประมวลผลและวิเคราะห์ของหน่วยงานปลายทาง จากการวิเคราะห์โครงสร้างฐานข้อมูลระบบทะเบียนนักศึกษา พบตารางข้อมูลที่เกี่ยวข้อง 21 ตาราง ครอบคลุมข้อมูลประวัติส่วนบุคคล ข้อมูลการศึกษา ข้อมูลการลงทะเบียนเรียน และข้อมูลอ้างอิง ซึ่งมีความเชื่อมโยงกันภายในระบบ

เพื่อกำหนดแนวทางการดำเนินการ Data Cleansing ที่เหมาะสม ผู้วิจัยได้จำแนกข้อมูลออกเป็น 2 มิติ คือ

การจำแนกประเภทข้อมูลตามความอ่อนไหว

1. ข้อมูลอ่อนไหว (Sensitive Data) ประกอบด้วยตาราง REG_STUDENT ที่มีข้อมูลประจำตัว การติดต่อ และสถานะการศึกษาของนักศึกษา ซึ่งมีความเสี่ยงต่อการระบุตัวบุคคล

2. ข้อมูลทั่วไป (Non-Sensitive Data) ประกอบด้วยตาราง REF_SUBJECT และ REG_COURSE_IN_GROUP ที่มีข้อมูลรหัสวิชา รายวิชา และหมวดวิชา ซึ่งไม่สามารถระบุตัวบุคคลได้

การจำแนกประเภทข้อมูลตามหน้าที่การใช้งาน

1. ข้อมูลอ้างอิง (Reference Data) ข้อมูลพื้นฐานที่กำหนดค่ามาตรฐานและเชื่อมโยงข้อมูลในระบบ เช่น REF_COURSE_GROUP, REF_PROVINCE, REF_FACULTY, REF_MAJOR, REF_GENDER เป็นต้น

2. ข้อมูลโครงสร้างหลักสูตร (Curriculum Structure Data) ข้อมูลที่กำหนดและเชื่อมโยงรายละเอียดหลักสูตรและรายวิชา เช่น REG_COURSE_IN_GROUP, REG_CURRICULUM, REG_CURRICULUM_VERSION

3. ข้อมูลนักศึกษา (Student Data) ข้อมูลส่วนบุคคลที่มีความอ่อนไหวสูง เช่น REG_STUDENT, REG_STUDENT_ADDRESS

4. ข้อมูลการลงทะเบียนเรียน (Student Registration Data) ข้อมูลการลงทะเบียนและผลการเรียน เช่น REG_STUDENT_REGISTRATION การจำแนกดังกล่าวเป็นพื้นฐานสำคัญในการกำหนดแนวทาง Data Cleansing ที่เหมาะสมกับลักษณะข้อมูลแต่ละประเภท โดยเฉพาะข้อมูลอ่อนไหวที่ต้องใช้มาตรการคุ้มครองข้อมูลส่วนบุคคลอย่างเคร่งครัด

ในการปรับปรุงคุณภาพข้อมูลและออกแบบฐานข้อมูลใหม่ ผู้วิจัยได้ดำเนินการ Source-to-Target Mapping (STM) โดยเชื่อมโยงคอลัมน์จากระบบเดิมไปยังระบบใหม่ ตัวอย่างการแมปตารางสำคัญแสดงในตารางด้านล่าง ทั้งนี้ การใช้เครื่องมือ apply_mapping_gui.py มีวัตถุประสงค์เพื่อแปลงไฟล์ข้อมูลจากระบบเดิมให้อยู่ในรูปแบบ CSV ใหม่ที่มีโครงสร้างสอดคล้องกับตารางที่ออกแบบขึ้นตาม Source-to-Target Mapping โดยผลลัพธ์ในขั้นตอนนี้ยังคงอยู่ในรูปแบบไฟล์ CSV มิใช่การบันทึกข้อมูลเข้าสู่ฐานข้อมูล PostgreSQL โดยตรง ข้อมูล CSV ที่ได้จะถูกนำไปผ่านกระบวนการ Data Cleansing ในหัวข้อถัดไป เพื่อปรับปรุงคุณภาพข้อมูลให้มีความถูกต้องและสอดคล้อง ก่อนจะถูกโหลดเข้าสู่ฐานข้อมูลใหม่และใช้ในการทดสอบระบบต้นแบบต่อไป

ตาราง 2 Source-to-Target Mapping ข้อมูลนักศึกษา

Source Table: REG_STUDENT		Target Table: reg_student		
Source Column	Source Type	Target Column	Target Type	Transformation Rule / Note
STUDENT_ID	-	student_id	character(10)	
YEAR_ENTRY	-	year_entry	character(4)	
SEMESTER_ENTRY	-	semester_entry	character(1)	
STUDENT_TYPE	-	student_type_code	character(2)	ref_student_type

TITLE_THI	-	title_code	character(3	ref_prefix
NAME_ENG	-	name_eng	character50)	
-	-	middle_name_eng	character50)	
SURNAME_ENG	-	surname_eng	character50)	
NAME_THI	-	name_tha	character50)	
-	-	middle_name_tha	character50)	
SURNAME_THI	-	surname_tha	character50)	
SEX	-	gender_code	character(1	ref_gender
BIRTHDATE	-	birthdate	date	แปลง YYYYMMDD; พ.ศ. > ค.ศ.
ADMIT_DATE	-	admit_date	date	แปลง YYYYMMDD; พ.ศ. > ค.ศ.
GRADUATION_DATE	-	graduation_date	date	แปลง YYYYMMDD; พ.ศ. > ค.ศ.
LEAVE_DATE	-	leave_date	date	แปลง YYYYMMDD; พ.ศ. > ค.ศ.
PERSONAL_ID	-	personal_id	character(20)	
HIEGHT	-	height	integer	
WEIGHT	-	weight	integer	
RACE	-	race_code	character(3)	ref_race
NATION_CODE	-	nationality_code	character(3)	ref_nationality
RELIGION_CODE	-	religion_code	character(3)	ref_religion
FACULTYZ	-	faculty_code	character(3)	ref_faculty
MAJORZ	-	major_code	character(3)	ref_major
STRUC_CURRICULUM_COD	-	curriculum_id	character(15)	
-	-	curriculum_version	character(15)	reg_curriculum_version
CREDIT_REGISTER_TOTA	-	credit_register_total	integer	
CG_P_A	-	cgpa	real	
STUDENT_STATUS_CODE	-	status_code	character(2)	ref_student_status
CONSULT_TEACHER	-	consult_teacher	character(10)	
CONFIRM_TEACHER	-	confirm_teacher	character(10)	
FORM_FEE_REGISTER_ST	-	form_fee_register	text	
PASSWORDZ	-	password	character(255)	
COMMENTZ	-	comment	text	
CLASSZ	-	grade_level	integer	

ตาราง 3 Source-to-Target Mapping ข้อมูลการลงทะเบียนและผลการศึกษา

Source Table: REG_REGISTER_SUBJECT		Target Table: reg_student_registration		
Source Column	**Source Type**	**Target Column**	**Target Type**	**Transformation Rule / Note**
YEAR_REGISTER_ZZ		reg_year	character(4)	
SEMESTER_REGISTER_ZZ		reg_semester	character(1)	
YEAR_REAL		academic_year	character(4)	
SEMESTER_REAL		academic_semester	character(1)	
REG_STUDENT_ID		student_id	character(10)	
NOZ		seq_no	smallint	

REG_SUBJECT_CODE		subject_code	character(10)	
REG_SUBJECT_TYPE		subject_type	character(3)	
CREDIT_REGISTER		credit_reg	numeric(4,1)	
CREDIT_REGISTER_OLD		credit_reg_old	numeric(4,1)	
CREDIT_EARN		credit_earned	numeric(4,1)	
SUBJECT_GROUP_THEORY		group_theory	character(2)	
SUBJECT_GROUP_LAB		group_lab	character(2)	
GRADEZ		grade	character(2)	
GRADE_DOUBLECHECK		grade_doublecheck	character(2)	
CONFIRMZ		grade_confirmed	character(2)	
REGISTER_COMMENT		comment_other	character(255)	
REGISTER_FEE_AMT		fee_register	numeric(10,2)	
LAB_FEE_AMT		fee_lab	numeric(10,2)	
INTENSIVE_FEE_AMOUNT		fee_intensive	numeric(10,2)	
MAINT_FEE_AMT		fee_maintenance	numeric(10,2)	
REGISTER_STATUS		register_status	character(1)	
TRANSFER_GRADE_FLG		flg_transfer_grade	character(1)	
REGISTER_ALLOWED_FLG		flg_allow_register	character(1)	
STUDY_OVELAPTIME_FLG		flg_overlap_study	character(1)	
MID_OVLT_FLG		flg_overlap_mid	character(1)	
FINAL_OVLT_FLG		flg_overlap_final	character(1)	
CANCEL_FLAG		flg_cancel	character(1)	
CHECKCO_STUDYFLG		flg_check_overlap	character(1)	
APPROVED_BY		approved_by	character(10)	
GRADE_ENTRYBY		grade_entry_by	character(10)	
GRADE_ENTRYDATE		grade_entry_date	date	แปลง YYYYMMDD; พ.ศ. > ค.ศ.
GRADE_ENTRYTIME		grade_entry_time	time	
CONSULT_TEACHERID		consult_teacher_id	character(10)	
COUNT_GRADE		count_grade	smallint	
COUNT_CREDIT		count_credit	numeric(4,1)	
PART_OF_CREDIT		part_credit	numeric(4,1)	
CONTACT_SEQREGSUB_AA		contact_seq_a	smallint	
CONTACT_SEQREQSUB_BB		contact_seq_b	smallint	
FLAG_PAID		flag_paid	character(1)	
DATE_REGIS		date_register	date	แปลง YYYYMMDD; พ.ศ. > ค.ศ.

ตาราง 4 Source-to-Target Mapping ข้อมูลหลักสูตร

Source Table: REG_STRUC_LAKSUD		Target Table: reg_curriculum_version		
Source Column	**Source Type**	**Target Column**	**Target Type**	**Transformation Rule / Note**
-		curriculum_version	character(15)	PK ex. B000001-01
CODE_STUC_CURRICULUM		curriculum_id	character(15)	รหัสหลักสูตรเดิม

-		version_no	character(6)	
CURRICULUM_CODE		curriculum_code	smallint	Code ชื่อหลักสูตร (B1, B2, ..)
FACULTY_CODE		faculty_code	character(3)	
MAJOR_CODE		major_code	character(4)	
LEARNING_YEAR		learning_year	smallint	
TOTAL_CREDIT		total_credit	smallint	
ESTIMATE_CREDIT_GRAD		est_credit_grad	smallint	
YEAR_BEGIN		year_begin	smallint	
YEAR_END		year_end	smallint	
CLOSEZ		close_flag	smallint	
SUB_TYPE		sub_type	smallint	
-		created_at	timestamp	อัปเดต
-		updated_at	timestamp	อัปเดต

4.2 กรอบแนวคิดและขั้นตอนการทำ Data Cleansing

จากการจัดทำ Source-to-Target Mapping และการสร้างไฟล์ CSV ใหม่ที่มีโครงสร้างสอดคล้องกับตารางฐานข้อมูลที่ออกแบบไว้ ข้อมูลดังกล่าวถูกนำมาเข้าสู่กระบวนการ Data Cleansing เพื่อปรับปรุงคุณภาพให้ถูกต้อง ครบถ้วน และสอดคล้องกับกฎเกณฑ์ทางธุรกิจของระบบทะเบียนนักศึกษา โดยงานวิจัยนี้กำหนดกรอบแนวคิด วิธีการ และขั้นตอนการดำเนินการ Data Cleansing อย่างชัดเจน เพื่อให้ได้ข้อมูลที่สามารถนำไปใช้ประเมินคุณภาพและทดสอบการทำงานของระบบต้นแบบได้อย่างมีประสิทธิภาพ

4.2.1 กรอบแนวคิด (Framework)

การดำเนินการ Data Cleansing ในงานวิจัยนี้อ้างอิงแนวคิด ETL (Extraction–Transformation–Loading) โดยมุ่งเน้นการปรับปรุงคุณภาพข้อมูลให้สามารถใช้งานได้อย่างถูกต้องและสอดคล้องกับโครงสร้างที่ออกแบบใหม่ ภายใต้เทคนิคหลัก 3 กลุ่ม ได้แก่

1. Rule-Based Data Cleaning การใช้กฎเชิงตรรกะ (Business Rules) ที่นิยามไว้อย่างชัดเจน เช่น ตรวจสอบรูปแบบรหัสนักศึกษา, การตรวจสอบค่า Null, และการแมปกับตารางอ้างอิง (Reference Table)

2. Software-Based Cleaning ใช้เครื่องมือช่วย เช่น OpenRefine และเทคนิค Text Clustering (เช่น key-collision/nearest-neighbor) เพื่อรวมกลุ่มและแก้ไขค่าที่สะกดไม่ตรงมาตรฐานในข้อมูลข้อความ (เช่น ชื่อเขต อำเภอ จังหวัด)

งานนี้ประยุกต์ใช้ PyThaiNLP (NLP toolkit ภาษาไทยแบบสำเร็จรูป) ในขั้น Software-Based Cleaning เพื่อ ตรวจ/แนะนำการสะกด สำหรับคอลัมน์ชื่อวิชา (เช่น NAME_THAI_LINE1)

โดยอาศัยตัวตรวจแบบ Norvig ร่วมกับคลังคำ thai_words และ Custom Dictionary ของหน่วยงาน ทำให้พร้อมใช้งาน โดยไม่ต้องฝึกโมเดล และลดภาระแมนนวลก่อนเข้าสู่ขั้นตอนถัดไป

3. Machine Learning-Based Cleaning งานวิจัยนี้พัฒนาเครื่องมือ Spell Audit GUI โดยใช้เทคนิค One-Class Support Vector Machine (OCSVM) ร่วมกับ TF–IDF character n-grams สำหรับตรวจจับและระบุค่าข้อมูลที่ผิดปกติ ทั้งภาษาไทยและภาษาอังกฤษ เครื่องมือสามารถทำการ Auto-tuning เพื่อเลือกพารามิเตอร์ที่เหมาะสมที่สุด (เช่น ค่า nu, ช่วง ngram, และ max_features) พร้อมทั้งรองรับการตรวจสอบแบบเรียลไทม์ผ่าน GUI และบันทึกรายงานผลการตรวจสอบ (spellcheck_report.csv) เพื่อนำมาใช้ในการแก้ไขข้อมูล การใช้วิธีนี้ทำให้สามารถระบุค่าที่สะกดผิดและค่าที่ไม่เป็นไปตามรูปแบบที่กำหนดได้อย่างมีประสิทธิภาพ

ทั้งนี้ สำหรับข้อมูลจำนวนน้อยที่ไม่คุ้มค่าต่อการสร้างกฎหรือโมเดล ผู้วิจัยเลือกใช้ Manual Cleaning (Microsoft Excel) เป็นเครื่องมือเสริม โดยมีการบันทึกหลักฐานการแก้ไขไว้ใน Cleaning Log เพื่อให้ตรวจสอบย้อนหลังได้

4.2.2 ขั้นตอนการดำเนินการ (Process Pipeline)

กระบวนการ Data Cleansing ดำเนินการอย่างเป็นลำดับขั้น ดังนี้

1. Data Profiling เบื้องต้น ทำการวิเคราะห์ภาพรวมของข้อมูลเพื่อตรวจสอบคุณภาพและระบุปัญหาที่พบ เช่น ค่า Missing Values, Duplicate Values, Invalid Formats และ Business Rules Violations พร้อมทั้งจัดทำ รายงานคุณภาพข้อมูลเบื้องต้น (Initial Data Quality Report) เพื่อใช้เป็นฐานข้อมูลอ้างอิงก่อนการดำเนินการ Cleansing

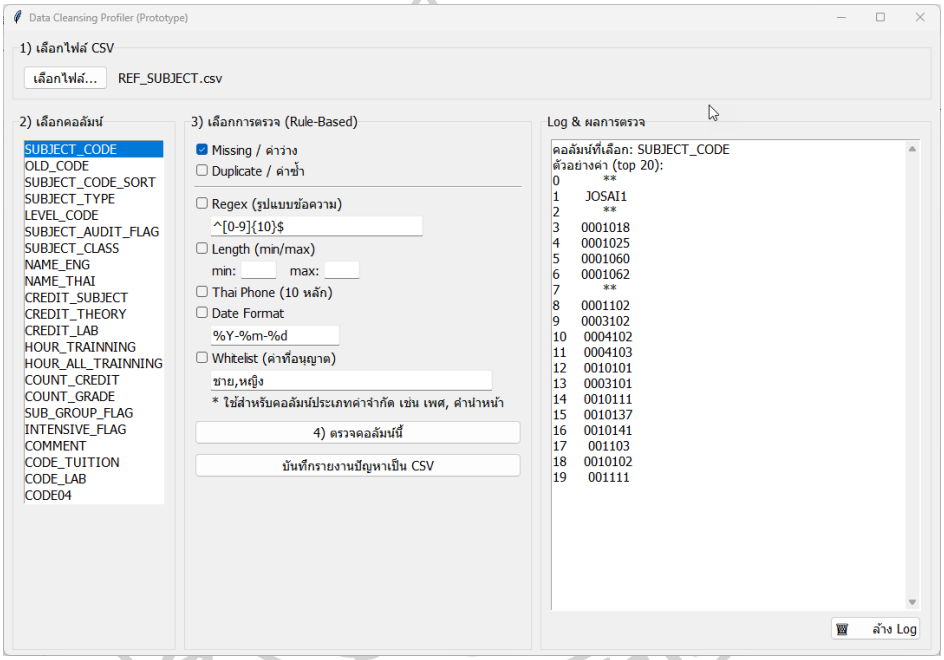
2. การเลือกวิธีการ Cleansing การเลือกวิธีการ Cleansing พิจารณาลักษณะของข้อมูลและประเภทของปัญหาที่ตรวจพบในแต่ละคอลัมน์ แล้วเลือกวิธีที่เหมาะสมตามลำดับความสำคัญ เช่น Rule-Based สำหรับข้อมูลเชิงโครงสร้าง, OpenRefine/Text Clustering และ PyThaiNLP (ตัวตรวจแบบ Norvig + คลังคำ thai_words + Custom Dictionary) สำหรับข้อมูลข้อความ (เช่น ชื่อวิชา NAME_THAI_LINE1), หรือ Machine Learning (เช่น SDCM, KNN) สำหรับกรณีที่ซับซ้อน

3. การดำเนินการ Cleansing ปรับปรุงและแก้ไขข้อมูลด้วยเครื่องมือที่เหมาะสม เช่น Python Script, OpenRefine, SQL Commands หรือ Microsoft Excel (สำหรับข้อมูลจำนวนน้อยหรือกรณีที่ต้องตรวจสอบเชิงบริบทโดยตรง) โดยบันทึกเหตุผลในการเลือกวิธีการและผลการแก้ไขไว้ทุกครั้ง

4. การตรวจสอบหลัง Cleansing ทำ Data Profiling ซ้ำ เพื่อประเมินว่าปัญหาที่ตรวจพบได้รับการแก้ไขครบถ้วน และตรวจสอบความถูกต้อง/สอดคล้องของข้อมูล พร้อมทั้งบันทึกการดำเนินงานใน Data Cleaning Log เพื่อรองรับการตรวจสอบย้อนกลับ

5. การสรุปผลและบันทึก จัดทำตารางสรุปผลการทำ Data Cleansing ครอบคลุมชื่อของตารางที่ตรวจสอบ คอลัมน์ที่มีปัญหา ปัญหาที่พบ วิธีการที่เลือกใช้ เหตุผลประกอบ และผลลัพธ์หลังการแก้ไข เพื่อประเมินประสิทธิภาพของกระบวนการ

เพื่อให้การวิเคราะห์คุณภาพข้อมูลทำได้สะดวกและมีหลักฐานตรวจสอบได้ งานวิจัยนี้ได้พัฒนา Data Profiling Tool แบบ GUI สำหรับใช้สำรวจและวิเคราะห์ภาพรวมของข้อมูล เช่น การหาค่าที่ว่าง ค่าซ้ำ ค่าที่ไม่ถูกต้องตามรูปแบบ และการตรวจสอบข้อผิดพลาดเชิงกฎธุรกิจ โดยผลลัพธ์จะแสดงเป็นรายงานเชิงสถิติและกราฟเบื้องต้น เพื่อใช้ประกอบการตัดสินใจเลือกวิธีการ Cleansing ในขั้นตอนต่อไป



ภาพที่ 8 Data Profiling Tool สำหรับวิเคราะห์ภาพรวมข้อมูลเพื่อระบุปัญหา

4.2.3 ตาราง Mapping ปัญหา-วิธีการ-ผลลัพธ์)

เพื่อแสดงให้เห็นถึงความเชื่อมโยงระหว่างปัญหาที่ตรวจพบ วิธีการหรือกฎที่เลือกใช้ในการแก้ไข และผลลัพธ์ที่เกิดขึ้นหลังการดำเนินการ งานวิจัยจึงได้จัดทำตาราง Mapping สำหรับตารางข้อมูลหลักที่เกี่ยวข้อง โดยตารางเหล่านี้สรุปสาระสำคัญของกระบวนการ Data Cleansing และใช้เป็นหลักฐานประกอบการประเมินคุณภาพข้อมูล ทั้งนี้ รายละเอียดเชิงปริมาณและผลการตรวจสอบทุกตารางได้จัดเก็บไว้ในภาคผนวก ก เพื่อความสมบูรณ์และตรวจสอบได้

ตาราง 5 สรุปกระบวนการ Data Cleansing ตารางข้อมูลนักศึกษา (REG_STUDENT)

คอลัมน์	ปัญหา	วิธี/กฎการแก้ไข	ผลลัพธ์/สถานะ
REG_YEAR	พบค่า null จำนวนเล็กน้อย	Manual Cleaning	เคลียร์ค่า null ตามหลักฐานใน Log
STUDENT_ID	พบค่า null บางกรณี	Manual Cleaning	เติม/ยืนยันค่า > ลด null เป็นศูนย์ตามตารางผลลัพธ์ในเอกสาร
SUBJECT_CODE	ค่าว่าง/ไม่ครบถ้วน	Manual Cleaning ตรวจ cross-table / อ้างอิงรหัสวิชา	ปรับปรุงความถูกต้อง (รายละเอียดตามตารางผลในไฟล์)

ตาราง 6 สรุปกระบวนการ Data Cleansing ตารางข้อมูลการลงทะเบียน (REG_STUDENT_REGISTRATION)

คอลัมน์	ปัญหา	วิธี/กฎการแก้ไข	ผลลัพธ์/สถานะ
REG_YEAR	พบค่า null จำนวนเล็กน้อย	Manual Cleaning	เคลียร์ค่า null ตามหลักฐานใน Log
STUDENT_ID	พบค่า null บางกรณี	Manual Cleaning	เติม/ยืนยันค่า > ลด null เป็นศูนย์ตามตารางผลลัพธ์ในเอกสาร
SUBJECT_CODE	ค่าว่าง/ไม่ครบถ้วน	Manual Cleaning ตรวจ cross-table / อ้างอิงรหัสวิชา	ปรับปรุงความถูกต้อง (รายละเอียดตามตารางผลในไฟล์)

ตาราง 7 สรุปกระบวนการ Data Cleansing ตารางข้อมูลหลักสูตร (REG_CURRICULUM_VERSION)

คอลัมน์	ปัญหา	วิธี/กฎการแก้ไข	ผลลัพธ์/สถานะ
CURRICULUM_VERSION / CURRICULUM_ID / CURRICULUM_CODE / FACULTY_CODE	ตรวจคุณภาพ/ความสอดคล้อง	Manual Cleaning	ยืนยันค่าถูกต้องตามโครงสร้าง
MAJOR_CODE	พบค่า null บางรายการ	Manual Cleaning และส่งให้หน่วยงานที่เกี่ยวข้องตรวจสอบ/ยืนยัน	แก้เฉพาะรายการที่ยืนยันได้; รายการที่ยังไม่ยืนยันคงสถานะรอตรวจสอบ

ตาราง 8 สรุปกระบวนการ Data Cleansing ตารางข้อมูลหลักสูตร (REF_MAJOR)

คอลัมน์	ปัญหา	วิธี/กฎการแก้ไข	ผลลัพธ์/สถานะ
NAME_EN_LONG	ค่าว่างบางส่วน	Manual Cleaning + Machine Learning + ส่งหน่วยงานตรวจสอบ	ค่าว่างคงเหลือเท่าที่จำเป็นจนกว่าจะยืนยัน; ใช้มาตรฐานการตั้งชื่อเดียวกันทั้งระบบ
NAME_TH_LONG	ไม่พบปัญหา	-	ใช้เป็นมาตรฐานอ้างอิงตามปกติ
MAJOR_CODE	ไม่พบปัญหา	Manual ตรวจยืนยัน	ใช้เป็นรหัสอ้างอิงตรงไปตรงมา

ตาราง 9 สรุปกระบวนการ Data Cleansing ตารางข้อมูลหลักสูตร (REF_FACULTY)

คอลัมน์	ปัญหา	วิธี/กฎการแก้ไข	ผลลัพธ์/สถานะ

NAME_TH_LONG	พบสะกดผิดบางรายการ	Manual Cleaning + Machine Learning	เคลียร์คำสะกดผิด; ใช้รูปแบบมาตรฐานเดียวกัน
NAME_EN_LONG	ตรวจสอบมาตรฐานการสะกด	Manual Cleaning + Machine Learning + ส่งหน่วยงานตรวจสอบ	ยืนยันความสอดคล้องคู่ไทย/อังกฤษ
FACULTY_CODE	ไม่พบปัญหา	Manual ตรวจยืนยัน	ใช้เป็นรหัสอ้างอิงตรงไปตรงมา

ตาราง 10 สรุปกระบวนการ Data Cleansing ตารางข้อมูลหลักสูตร (REF_SUBJECT)

คอลัมน์	ปัญหา	วิธี/กฎการแก้ไข	ผลลัพธ์/สถานะ
SUBJECT_CODE	ว่าง/ซ้ำ; ต้องสัมพันธ์กับ SUBJECT_TYPE/OLD_CODE	Rule-Based ตรวจหาว่าง/ซ้ำ + cross-check คอลัมน์ที่เกี่ยวข้อง; ส่งหน่วยงานยืนยันก่อนบันทึกเปลี่ยนแปลง	ลดซ้ำ/ค่าว่าง; รักษาความสอดคล้องของรหัส
NAME_THA	สะกดไม่มาตรฐาน	OpenRefine Text-Clustering > ตรวจสะกดด้วย ML > ส่งผู้เชี่ยวชาญยืนยัน	มาตรฐานชื่อวิชา (ไทย) สอดคล้องทั่วทั้งระบบ
NAME_ENG	ว่าง/มีไทยปน/สะกดผิด	ตรวจสะกดด้วย ML	ลด invalid format / ปรับเป็นมาตรฐานอังกฤษ

จากการสรุปปัญหา วิธีการ และผลการปรับปรุงคุณภาพข้อมูลในหัวข้อ 4.2 จะเห็นได้ว่ากระบวนการ Data Cleansing ถูกออกแบบและดำเนินการอย่างเป็นระบบ อย่างไรก็ตาม รายละเอียดเชิงลึกของการดำเนินการจริง รวมถึงตัวอย่างการ Cleansing ของแต่ละตาราง จะนำเสนอในหัวข้อ 4.3 ต่อไป

4.3 การดำเนินการและผลการทำ Data Cleansing

การดำเนินการ Data Cleansing ครอบคลุมตารางข้อมูลทั้งหมด 21 ตาราง โดยผู้วิจัยได้เลือกนำเสนอกรณีศึกษาของตารางข้อมูลสำคัญ เพื่อแสดงรายละเอียดขั้นตอน วิธีการ และผลลัพธ์ที่ได้อย่างเป็นลำดับ ตลอดจนสะท้อนให้เห็นปัญหาและอุปสรรคที่เกิดขึ้นจริงระหว่างการดำเนินงาน

โครงสร้างการนำเสนอผลการทำ Data Cleansing

ในแต่ละตารางจะนำเสนอเนื้อหาในรูปแบบเดียวกันเพื่อความเป็นระบบ ดังนี้

1. ภาพรวมตารางและโครงสร้างข้อมูล อธิบายลักษณะข้อมูล ปริมาณคอลัมน์ จำนวนแถว และความเชื่อมโยงกับตารางอื่นในระบบ

2. ผล Data Profiling เบื้องต้น แสดงปัญหาที่ตรวจพบ เช่น ค่าขาดหาย (Missing Values), ค่าซ้ำซ้อน (Duplicate Values), ค่าผิดรูปแบบ (Invalid Formats) และค่าที่ไม่สอดคล้องกับกฎเกณฑ์ (Business Rules Violations) ในขั้นตอนนี้ผู้วิจัยใช้เครื่องมือ Data Profiling ที่พัฒนาด้วยภาษา Python ซึ่งสามารถเลือกไฟล์และคอลัมน์ที่ต้องการตรวจสอบได้ พร้อมบันทึกผลเป็นรายงานเพื่อใช้เปรียบเทียบก่อนและหลังการ Cleansing

3. วิธีการ Cleansing และเหตุผล ระบุวิธีการที่ใช้ตามลำดับความสำคัญ (Machine Learning-Based, Software-Based, Rule-Based, Manual) พร้อมเหตุผลการเลือกใช้ เช่น ความเหมาะสมกับปริมาณข้อมูล ความซับซ้อนของปัญหา และความแม่นยำที่ต้องการ

4. ผล Data Profiling หลัง Cleansing แสดงผลการตรวจสอบซ้ำหลังการแก้ไข เพื่อยืนยันว่าปัญหาทั้งหมดได้รับการแก้ไขแล้ว

การจัดโครงสร้างผลลัพธ์ในรูปแบบนี้ช่วยให้สามารถติดตามความคืบหน้าและประเมินประสิทธิภาพของกระบวนการ Data Cleansing ได้อย่างเป็นระบบ รวมทั้งสามารถใช้เป็นต้นแบบในการดำเนินการกับข้อมูลชุดอื่นในอนาคต

ตารางที่ 1 REF_SUBJECT

1. ภาพรวมตารางและโครงสร้างข้อมูล

ตาราง REF_SUBJECT เป็นตารางข้อมูลอ้างอิงรายวิชา ซึ่งใช้เป็นข้อมูลหลักสำหรับการลงทะเบียนเรียนและการจัดการหลักสูตรภายในมหาวิทยาลัย ข้อมูลในตารางนี้มีความเชื่อมโยงกับตาราง REG_COURSE_IN_GROUP และ REG_CURRICULUM_VERSION เพื่อกำหนดความสัมพันธ์ระหว่างรายวิชากับหมวดวิชาและหลักสูตร ข้อมูลประกอบด้วย 29 คอลัมน์ รวมทั้งสิ้น 7,556 แถว ครอบคลุมรายละเอียด เช่น รหัสวิชา (SUBJECT_CODE) ชื่อวิชา (NAME_THA / NAME_ENG) หน่วยกิต (CREDIT) การจัดหมวดวิชา และข้อมูลกำกับอื่น ๆ

2. ผลการทำ Data Profiling เบื้องต้น

การตรวจสอบข้อมูลตารางนี้ใช้เครื่องมือ Data Profiling ที่ผู้วิจัยพัฒนาด้วยภาษา Python โดยเครื่องมือมีความสามารถในการ

1. เลือกไฟล์ข้อมูล (CSV)
2. เลือกคอลัมน์ที่ต้องการตรวจสอบ
3. เลือกประเภทการตรวจสอบ เช่น ค่าขาดหาย (Missing Values), ค่าซ้ำซ้อน (Duplicate Values), ค่าผิดรูปแบบ (Invalid Formats), และการตรวจสอบค่ากับข้อมูลอ้างอิง (Reference Data Validation)
4. แสดงผลการตรวจสอบในรูปแบบตารางบนหน้าจอ พร้อมบันทึกเป็นรายงานเพื่อใช้เปรียบเทียบก่อนและหลังการ Cleansing

สำหรับตาราง REF_SUBJECT ผู้วิจัยได้เลือกตรวจสอบคอลัมน์ที่เกี่ยวข้องกับข้อมูลหลักของรายวิชา ได้แก่ **SUBJECT_CODE, SUBJECT_TYPE, NAME_THA, NAME_ENG, CREDIT_SUBJECT, CREDIT_THEORY, CREDIT_LAB** เนื่องจากเป็นข้อมูลที่มีบทบาทสำคัญต่อการเชื่อมโยงกับตารางอื่น และเป็นข้อมูลที่ต้องจัดส่งให้หน่วยงานภายนอก

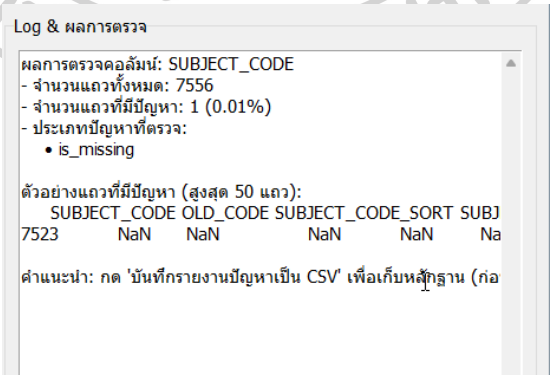
ส่วนคอลัมน์ที่ไม่ถูกนำมาตรวจสอบ เช่น **OLD_CODE, SUBJECT_CODE_SORT, SUBJECT_AUDIT_FLAG** จัดอยู่ในกลุ่มข้อมูลกำกับ (System Control Fields) ซึ่งใช้เพื่อการจัดการภายในระบบ และไม่มีผลต่อการแลกเปลี่ยนข้อมูลภายนอกหรือการวิเคราะห์คุณภาพข้อมูลในครั้งนี้ จึงคงค่าเดิมจากระบบต้นทางโดยไม่ดำเนินการตรวจสอบเชิงเนื้อหา

จากผลการทำ Data Profiling โดยใช้เครื่องมือ Data Profiling Tool ที่ผู้วิจัยพัฒนาด้วยภาษา Python (ไฟล์ apply_clearsing_profiler.py) พบผลการตรวจสอบดังตารางสรุป

ตาราง 11 ผลการตรวจสอบข้อมูลตาราง REF_SUBJECT

คอลัมน์	เงื่อนไขการตรวจสอบ	ผลการตรวจสอบ	หมายเหตุ
SUBJECT_CODE	ห้ามว่าง, ห้ามซ้ำ	ค่าว่าง 1 (0.01%), ซ้ำ 360 (4.76%)	ต้องตรวจสอบ SUBJECT_TYPE, OLD_CODE ประกอบ
NAME_THA	ห้ามว่าง	ค่าว่าง 44 (0.58%) มีคำที่คาดว่าสะกดผิด 348 (4.61%)	-
NAME_ENG	ห้ามว่าง, ห้ามมีภาษาไทยประสม	ค่าว่าง 279 (3.69%), มีภาษาไทยปน 17 (0.22%) มีคำที่คาดว่าสะกดผิด 35 (0.46%)	-
CREDIT_SUBJECT	ห้ามว่าง	ค่าว่าง 0 (0.00%)	-
CREDIT_THEORY	ห้ามว่าง	ค่าว่าง 0 (0.00%)	-
CREDIT_LAB	ห้ามว่าง	ค่าว่าง 0 (0.00%)	-

จากผลการตรวจสอบจะเห็นว่าปัญหาหลักของตารางนี้คือ ค่าซ้ำ (Duplicate Values) และ ค่าขาดหาย (Missing Values) ในหลายคอลัมน์ โดยเฉพาะ SUBJECT_CODE และ LEVEL_CODE ซึ่งมีผลต่อการเชื่อมโยงข้อมูลและความถูกต้องของการอ้างอิงข้ามตาราง การแก้ไขจึงต้องเลือกวิธีการ Cleansing ให้สอดคล้องกับประเภทปัญหาและระดับความสำคัญของข้อมูล



ภาพที่ 9 ตัวอย่าง Log & ผลการตรวจของ Data Profiling Tool

3. วิธีการ Cleansing และเหตุผล

จากผลการทำ Data Profiling ผู้วิจัยได้กำหนดแนวทางการปรับปรุงข้อมูลของตาราง REF_SUBJECT โดยพิจารณาตามลักษณะปัญหาของแต่ละคอลัมน์และเลือกใช้วิธีการที่เหมาะสม ดังนี้

1. SUBJECT_CODE

ใช้วิธีการแบบ Rule-Based เนื่องจากสามารถกำหนดเงื่อนไขการตรวจสอบรหัสวิชาได้อย่างชัดเจน เช่น การตรวจหาค่าที่ซ้ำหรือว่าง และการเปรียบเทียบกับคอลัมน์อื่นที่เกี่ยวข้อง (เช่น SUBJECT_TYPE, OLD_CODE) เพื่อยืนยันความถูกต้อง ก่อนส่งให้ส่วนงานที่เกี่ยวข้องตรวจสอบเพิ่มเติม วิธีนี้ช่วยป้องกันการซ้ำซ้อนในตารางหลักและคงความสอดคล้องของรหัสวิชาในระบบ

2. NAME_THA

ใช้วิธีการแบบผสมผสาน โดยเริ่มจากการใช้ OpenRefine เพื่อจัดกลุ่มคำที่สะกดใกล้เคียงกันและช่วยตรวจหาคำที่ผิดรูปแบบ จากนั้นใช้โมเดล Machine Learning ในการตรวจสอบการสะกดคำภาษาไทยเพิ่มเติม สุดท้ายจึงนำคำที่ยังไม่ชัดเจนส่งให้ผู้เชี่ยวชาญยืนยัน วิธีนี้ช่วยให้มั่นใจได้ว่าชื่อวิชาภาษาไทยมีความถูกต้องและเป็นมาตรฐาน ลดความเสี่ยงจากการสะกดผิดหรือการใช้คำไม่สอดคล้องกัน

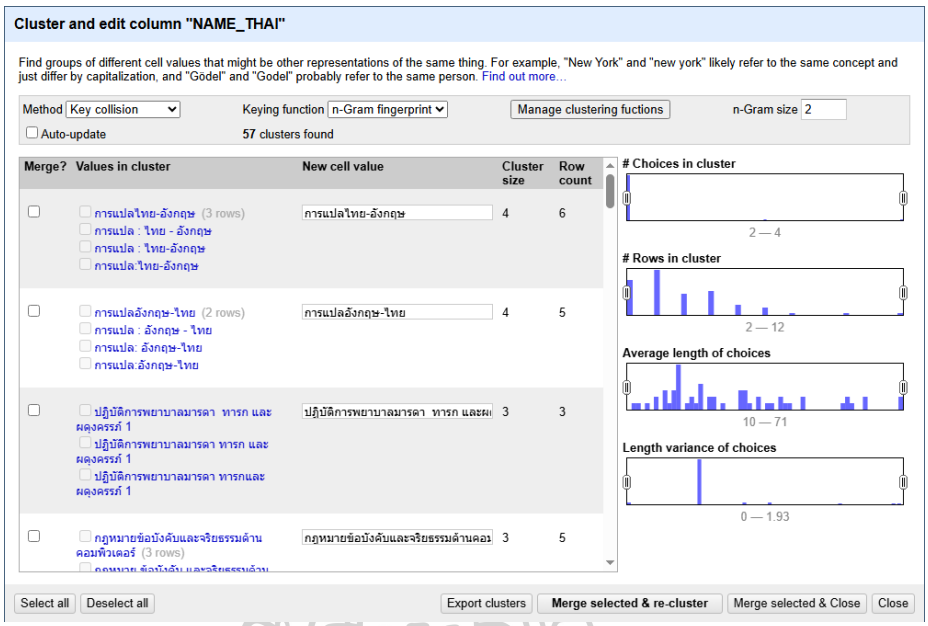
3. NAME_ENG

ใช้วิธีการแบบ Rule-Based ควบคู่กับ Machine Learning โดยเริ่มจากการตรวจจับรูปแบบที่ไม่ถูกต้อง เช่น การมีตัวอักษรไทยปนในข้อความภาษาอังกฤษหรือการใช้รูปแบบการสะกดที่ผิด จากนั้นใช้โมเดล Machine Learning เพื่อตรวจสอบการสะกดคำภาษาอังกฤษเพิ่มเติม และส่งผลที่ได้ให้ผู้เชี่ยวชาญตรวจสอบซ้ำเพื่อป้องกันความผิดพลาด วิธีนี้ช่วยให้ข้อมูลภาษาอังกฤษมีความถูกต้องตามมาตรฐานสากลและลดปัญหาการปนเปื้อนของข้อมูล

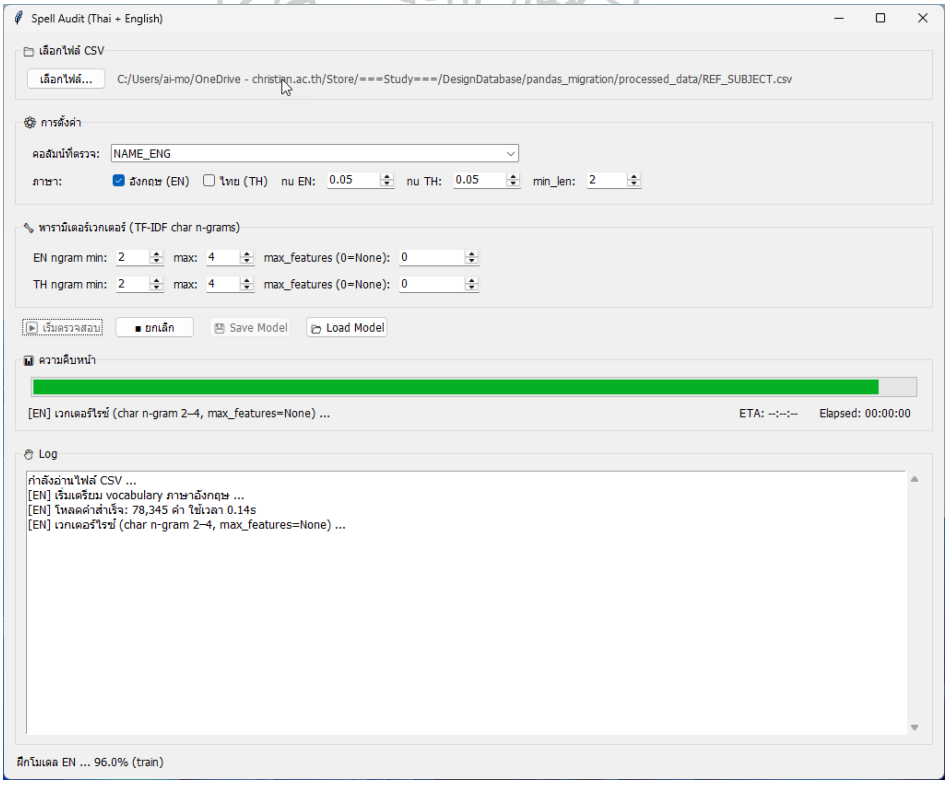
4. CREDIT_SUBJECT, CREDIT_THEORY และ CREDIT_LAB

ข้อมูลในคอลัมน์เหล่านี้มีความสมบูรณ์ ไม่พบปัญหาจากการทำ Data Profiling จึงไม่จำเป็นต้องดำเนินการปรับปรุงเพิ่มเติม

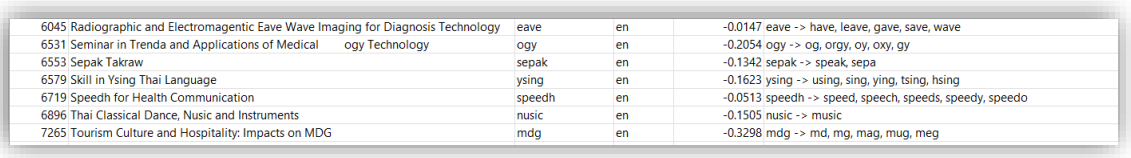
โดยรวมแล้ว การเลือกใช้วิธีการ Cleansing ของตาราง REF_SUBJECT มุ่งเน้นไปที่การใช้ Rule-Based สำหรับข้อมูลเชิงโครงสร้าง (เช่น รหัสวิชา) และการใช้ OpenRefine และ Machine Learning สำหรับข้อมูลข้อความ (เช่น ชื่อวิชาภาษาไทยและอังกฤษ) ร่วมกับการตรวจสอบโดยผู้เชี่ยวชาญเพื่อสร้างความมั่นใจสูงสุดว่าข้อมูลที่ได้จะมีมาตรฐานและสามารถนำไปใช้ต่อได้อย่างถูกต้อง



ภาพที่ 10 OpenRefine ในการจัดกลุ่มและแก้ไขคำผิด



ภาพที่ 11 ตัวอย่าง ML GUI Tool : อัลกอริทึม One-Class SVM (OCSVM)



6045	Radiographic and Electromagentic Eave Wave Imaging for Diagnosis Technology	eave	en	-0.0147	eave -> have, leave, gave, save, wave
6531	Seminar in Trenda and Applications of Medical ogy Technology	ogy	en	-0.2054	ogy -> og, orgy, oy, oxy, gy
6553	Sepak Takraw	sepak	en	-0.1342	sepak -> speak, sepa
6579	Skill in Ysing Thai Language	ysing	en	-0.1623	ysing -> using, sing, ying, tsing, hsing
6719	Speedh for Health Communication	speedh	en	-0.0513	speedh -> speed, speech, speeds, speedy, speedo
6896	Thai Classical Dance, Nusic and Instruments	nusic	en	-0.1505	nusic -> music
7265	Tourism Culture and Hospitality: Impacts on MDG	mdg	en	-0.3298	mdg -> md, mg, mag, mug, meg

ภาพที่ 12 รายการข้อมูลที่คาดว่าจะพิมพ์ผิด จาก ML GUI Tool

การประยุกต์ใช้ Machine Learning

สำหรับการตรวจจับค่าผิดปกติ ผู้วิจัยเลือกใช้อัลกอริทึม One-Class Support Vector Machine (OCSVM) โดยมีการดำเนินงานดังนี้

1. การเตรียมข้อมูลฝึก (Training Data)

ภาษาไทย : ใช้ชุดคำมาตรฐานจาก PyThaiNLP corpus (thai_words) ซึ่งเป็นคลังคำที่ผ่านการตรวจสอบโดยผู้เชี่ยวชาญด้านภาษาไทยและมีการใช้งานแพร่หลายในงานประมวลผลภาษาธรรมชาติ (NLP) จึงมั่นใจได้ว่ามีความถูกต้องและครอบคลุมคำศัพท์พื้นฐาน

ภาษาอังกฤษ : ใช้ชุดคำจาก wordfreq ซึ่งเป็นฐานข้อมูลความถี่คำสากลที่ได้จากข้อความหลายโดเมน (เช่น เว็บไซต์ หนังสือ สิ่งพิมพ์) ทำให้สะท้อนการใช้งานจริงของภาษาอังกฤษได้ดีกว่าการใช้พจนานุกรมเพียงอย่างเดียว

2. การแปลงข้อมูล (Feature Extraction)

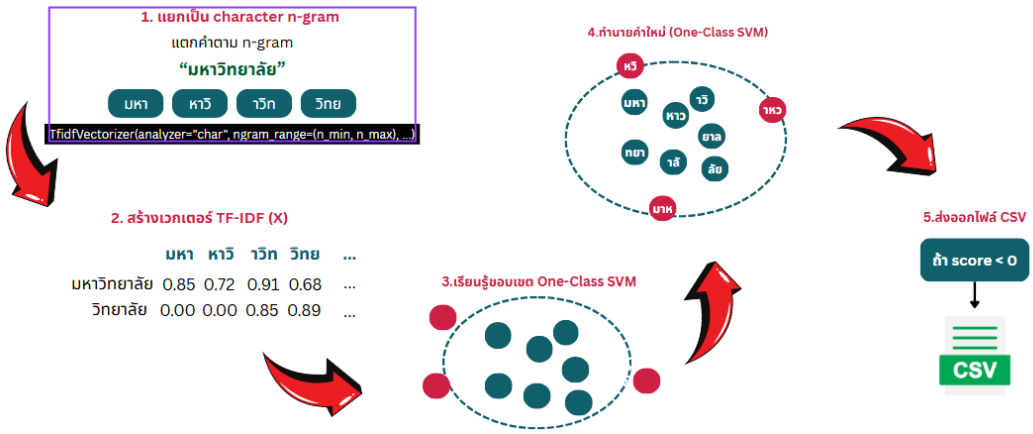
ใช้ TF-IDF Vectorization ในการแปลงข้อความเป็นเวกเตอร์เชิงตัวเลข เพื่อให้โมเดลเข้าใจความถี่และความสำคัญของคำ

3. การฝึกโมเดล (Model Training)

นำเวกเตอร์จากชุดข้อมูลมาตรฐานเข้าสู่ OCSVM เพื่อสร้างขอบเขตของ “ข้อมูลปกติ” ข้อมูลที่เบี่ยงเบนจากรูปแบบนี้จะถูกจัดเป็น outlier

4. การทดสอบและประเมินผล (Testing & Evaluation)

ใช้ข้อมูลใหม่ที่ไม่เคยฝึกมาก่อนเพื่อทดสอบ ประเมินด้วยตัวชี้วัด เช่น AUROC, AUPRC, Precision, Recall, F1-score



ภาพที่ 13 แผนภาพกระบวนการตรวจจับคำผิดด้วย Character n-gram และ One-Class SVM

จากการเปรียบเทียบผลการประเมินโมเดลทั้งสองแบบ พบว่า Model ที่มีผลการประเมินดีที่สุดมีค่าตัวชี้วัดทางสถิติ (AUROC, AUPRC, F1-score) สูงกว่าเล็กน้อย แสดงถึงศักยภาพเชิงทฤษฎีในการจำแนกข้อมูลปกติและคำผิดได้ดี อย่างไรก็ตาม เมื่อทดสอบกับข้อมูลจริง Model ที่ได้ข้อมูลถูกต้องที่สุด กลับสามารถตรวจพบคำผิดได้มากกว่าและตรงกับความเป็นจริงมากกว่า จึงเหมาะสมสำหรับการนำไปใช้งานจริง แม้ว่าค่าตัวเลขประเมินจะต่ำกว่าเล็กน้อยก็ตาม

4. ผล Data Profiling ก่อนและหลังการทำ Data Cleansing

ตาราง 12 ผลการตรวจสอบข้อมูลก่อนและหลังการทำ Data Cleansing ของ REF_SUBJECT

คอลัมน์	ก่อน Cleansing จำนวน (ร้อยละ)	หลัง Cleansing จำนวน (ร้อยละ)	หมายเหตุ
SUBJECT_CODE	ค่าว่าง 1 (0.01%), ซ้ำ 360 (4.76%)	ค่าว่าง 0 (0.00%) ซ้ำ 0 (0.00%)	
NAME_THA	ค่าว่าง 44 (0.58%)	ค่าว่าง 38 (0.50%) มีคำที่คาดว่าสะกดผิด 0 (0%)	เจ้าของข้อมูลไม่สามารถระบุข้อมูลได้
NAME_ENG	ค่าว่าง 279 (3.69%), มีภาษาไทยปน 17 (0.22%)	ค่าว่าง 271 (3.59%) มีภาษาไทยปน: 13 (0.17%) มีคำที่คาดว่าสะกดผิด 0 (0%)	เจ้าของข้อมูลไม่สามารถระบุข้อมูลได้

ตารางที่ 2 REG_STUDENT_ADDRESS

1. ภาพรวมตารางและโครงสร้างข้อมูล

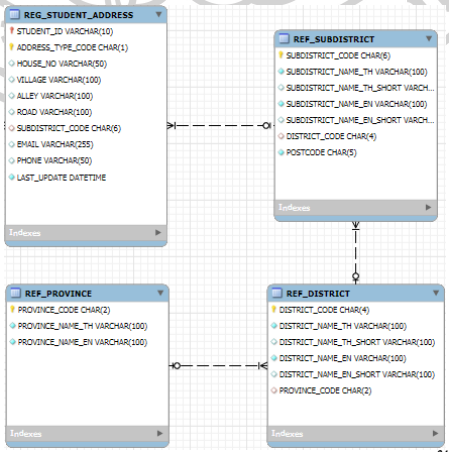
ตาราง REG_STUDENT_ADDRESS เป็นตารางจัดเก็บข้อมูลที่อยู่ของนักศึกษา ซึ่งมีบทบาทสำคัญในการบ่งชี้ที่อยู่ตามทะเบียนบ้าน เพื่อใช้ในการติดต่อสื่อสาร การจัดส่งเอกสาร และการยืนยันตัวตนของนักศึกษา ข้อมูลในตารางนี้เชื่อมโยงกับตารางอ้างอิง REF_SUBDISTRICT, REF_DISTRICT และ REF_PROVINCE เพื่อกำหนดรหัสและชื่อภูมิศาสตร์อย่างเป็นมาตรฐาน ลดความซ้ำซ้อนและความคลาดเคลื่อนของข้อมูลที่อยู่

ตาราง REG_STUDENT_ADDRESS ประกอบด้วย 12 คอลัมน์ รวมทั้งสิ้น 18,177 แถว ครอบคลุมรายละเอียดสำคัญ ได้แก่ รหัสนักศึกษา (STUDENT_ID) รหัสและชื่อจังหวัด อำเภอ ตำบล ตลอดจนข้อมูลประกอบอื่น ๆ โดยมีวัตถุประสงค์เพื่อให้มั่นใจว่าข้อมูลที่อยู่มีความถูกต้อง ครบถ้วน และสอดคล้องกับมาตรฐานการแลกเปลี่ยนข้อมูลทั้งภายในและภายนอกมหาวิทยาลัย

ในการออกแบบโครงสร้างข้อมูลใหม่ ผู้วิจัยพบว่าระบบเดิมจัดเก็บข้อมูลตำบล อำเภอ และจังหวัดในรูปแบบข้อความ (text) ซึ่งก่อให้เกิดข้อจำกัดในการตรวจสอบและนำไปใช้งานต่อ อีกทั้งยังพบปัญหาการสะกดผิดจำนวนมาก เนื่องจากเป็นการกรอกข้อมูลโดยตรงโดยไม่อ้างอิงชุดข้อมูลมาตรฐาน เพื่อแก้ไขปัญหาดังกล่าว ผู้วิจัยจึงได้ออกแบบโครงสร้างของตาราง REG_STUDENT_ADDRESS ใหม่ให้จัดเก็บเฉพาะ SUBDISTRICT_CODE และเชื่อมโยงไปยังตารางอ้างอิง (reference tables) เพื่อดึงค่าตำบล อำเภอ และจังหวัดที่ถูกต้องและเป็นมาตรฐาน

จากการตรวจสอบเบื้องต้นก่อนการทำ Data Cleansing พบว่าข้อมูลเดิมสามารถนำไปแมปกับตารางอ้างอิง (reference master) เพื่อหาค่า SUBDISTRICT_CODE ได้เพียง 241 แถว จากทั้งหมด 18,177 แถว คิดเป็นเพียง 1.33% ของข้อมูลทั้งหมด สะท้อนให้เห็นถึงความจำเป็นที่ต้องดำเนินการ Cleansing และปรับโครงสร้างข้อมูลอย่างจริงจัง เพื่อให้สามารถเชื่อมโยงกับตารางอ้างอิงได้อย่างมีประสิทธิภาพในขั้นตอนต่อไป

ในขั้นตอนแรกของการดำเนินงาน ผู้วิจัยจำเป็นต้องทำการ Data Cleansing ข้อมูลตำบล อำเภอ และจังหวัดจากฐานข้อมูลเดิม เพื่อให้ได้ข้อมูลที่ถูกต้องและสมบูรณ์มากที่สุด จากนั้นจึงนำข้อมูลดังกล่าวไปทำการแมปกับ SUBDISTRICT_CODE ที่ถูกต้องตามตารางอ้างอิงต่อไป



ภาพที่ 14 แผนภาพที่ใช้แสดงความสัมพันธ์ระหว่างเอนทิตี้ ของที่อยู่นักศึกษา

2. ผลการทำ Data Profiling เบื้องต้น

การตรวจสอบข้อมูลของตาราง REG_STUDENT_ADDRESS ดำเนินการโดยใช้เครื่องมือ Data Profiling Tool ร่วมกับการประยุกต์ใช้เทคนิค Fuzzy Matching เพื่อช่วยในการตรวจสอบและปรับปรุงคุณภาพข้อมูลเชิงข้อความ (Textual Data Cleansing) โดยมุ่งเน้นการตรวจสอบเฉพาะคอลัมน์ที่มีบทบาทสำคัญต่อการเชื่อมโยงและการแลกเปลี่ยนข้อมูลกับหน่วยงานภายนอก ได้แก่ STUDENT_ID, SUBDISTRICT_CODE, DISTRICT และ PROVINCE

ตาราง 13 ผลการตรวจสอบข้อมูลตาราง REG_STUDENT_ADDRESS

คอลัมน์	เงื่อนไขการตรวจสอบ	ผลการตรวจสอบ	หมายเหตุ
STUDENT_ID	ห้ามว่าง	ค่าว่าง 1 (0.01%)	-
SUBDISTRICT_CODE	ห้ามว่าง	ค่าว่าง 2,453 (13.50%) คาดว่าไม่ถูกต้อง 502 (2.76%)	-
DISTRICT	ห้ามว่าง	ค่าว่าง 2,102 (11.56%) คาดว่าไม่ถูกต้อง 3,505 (19.28%)	-
PROVINCE	ห้ามว่าง	ค่าว่าง 2,002 (11.01%) คาดว่าไม่ถูกต้อง 181 (0.99%)	-

จากผลการตรวจสอบพบว่าปัญหาหลักของตารางนี้ ได้แก่ ค่าขาดหาย (Missing Values) และค่าที่ผิดรูปแบบหรือไม่สอดคล้องกับมาตรฐานอ้างอิง (Invalid / Non-standardized Values) โดยเฉพาะในคอลัมน์ SUBDISTRICT_CODE, DISTRICT และ PROVINCE ซึ่งสะท้อนถึงข้อจำกัดของระบบเดิมที่อนุญาตให้บันทึกข้อมูลในรูปแบบข้อความอิสระ (Free-text Input) ทำให้เกิดความหลากหลายและข้อผิดพลาดในการสะกดคำ ผู้วิจัยจึงได้ใช้เทคนิค Fuzzy Matching ควบคู่กับ Data Profiling เพื่อช่วยระบุรายการที่มีความน่าจะเป็นสูงว่าเป็นข้อมูลที่ผิดพลาด และเตรียมสำหรับการ Cleansing ต่อไป

3. วิธีการ Cleansing และเหตุผล (ส่วนตรรกะเชิงลำดับชั้นสำหรับภูมิศาสตร์)

เพื่อแก้ไขปัญหาค่าขาดหายหรือค่าที่ไม่เป็นมาตรฐานในข้อมูลภูมิศาสตร์ ผู้วิจัยเลือกใช้แนวทางแบบ Hierarchical Matching ร่วมกับเทคนิค Fuzzy Matching โดยมีขั้นตอนหลักดังนี้

จังหวัด (Province)

1. หากมีค่า ทำการปรับมาตรฐาน/แก้ไขการสะกดให้ตรงกับตารางอ้างอิง

2. หากว่าง อ้างอิงจากข้อมูล “ตำบล” และ/หรือ “อำเภอ” เพื่อตรวจหาจังหวัดที่ถูกต้อง (หากไม่พบข้อมูลทั้งสอง ให้จัดเก็บสถานะเป็น Pending)

อำเภอ (District)

หากว่างหรือผิด ทำการค้นหาเฉพาะ ภายในจังหวัดที่ยืนยันแล้ว โดยใช้น้ำหนักจากค่าของ “ตำบล” เพื่อช่วยระบุชื่ออำเภอที่ถูกต้องที่สุด

ตำบล (SUBDISTRICT)

หากผิด ทำการค้นหาเฉพาะ ภายในคู่ (จังหวัด, อำเภอ) ที่ยืนยันแล้ว เพื่อลดความคลาดเคลื่อน

ตรวจสอบซ้ำ (Validation)

ผลลัพธ์ทั้ง 3 ระดับ (ตำบล-อำเภอ-จังหวัด) จะถูกส่งไปตรวจทานกับ ตารางอ้างอิงมาตรฐาน (Reference Master) เพื่อยืนยันว่า ชื่อและรหัสมีอยู่จริงและสัมพันธ์กัน ตามโครงสร้างการปกครอง

การใช้แนวทางเชิงลำดับชั้นช่วย ลดพื้นที่ค้นหา (Search Space) และลดโอกาสจับคู่ผิดพลาด ขณะที่ Fuzzy Matching ช่วยรองรับความคลาดเคลื่อนด้านการสะกด การเว้นวรรค และตัวสะกดแบบไม่มาตรฐานในระบบเดิม เมื่อรวมกับกระบวนการ Validation จึงทำให้ผลลัพธ์มี ความถูกต้องทั้งเชิงโครงสร้างและเชิงภูมิศาสตร์

สุดท้าย ผู้วิจัยได้นำผลการ Cleansing ที่มีข้อมูลครบทั้งสามระดับ (ตำบล-อำเภอ-จังหวัด) ไปทำการแมปกับ Reference Master เพื่อให้ได้ SUBDISTRICT_CODE ที่ถูกต้อง และสามารถนำไปใช้งานในโครงสร้างฐานข้อมูลใหม่ได้อย่างเป็นมาตรฐาน

4. ผล Data Profiling ก่อนและหลังการทำ Data Cleansing

ตาราง 14 ผลการตรวจสอบข้อมูลก่อนและหลังการทำ Data Cleansing ของ REG_STUDENT_ADDRESS

คอลัมน์	ก่อน Cleansing จำนวน (ร้อยละ)	หลัง Cleansing จำนวน (ร้อยละ)	**หมายเหตุ**
STUDENT_ID	ค่าว่าง 1 (0.01%)	ค่าว่าง 0 (0.00%)	
SUBDISTRICT_CODE	ค่าว่าง 2,453 (13.50%) คาดว่าไม่ถูกต้อง 502 (2.76%)	ค่าว่าง 3,078 (17.05%) คาดว่าไม่ถูกต้อง 0 (0.00%)	
DISTRICT	ค่าว่าง 2,102 (11.56%) คาดว่าไม่ถูกต้อง 3,505 (19.28%)	ค่าว่าง 2,184 (12.10%) คาดว่าไม่ถูกต้อง 0 (0.00%)	
PROVINCE	ค่าว่าง 2,002 (11.01%) คาดว่าไม่ถูกต้อง 181 (0.99%)	ค่าว่าง 1,893 (10.48%) คาดว่าไม่ถูกต้อง 0 (0.00%)	

จากผลการเปรียบเทียบก่อนและหลังการทำ Data Cleansing พบว่า จำนวนค่าที่ขาดหาย (Missing Values) ในบางคอลัมน์เพิ่มขึ้น โดยเฉพาะ SUBDISTRICT_CODE และ DISTRICT สาเหตุเนื่องจากการตรวจสอบพบว่าข้อมูลเดิมที่บันทึกไว้ (เช่น ชื่อตำบลหรืออำเภอที่สะกดผิด หรือไม่มีอยู่จริงในแฟ้มอ้างอิง) ไม่สามารถแมปเข้ากับรหัสมาตรฐานได้ จึงถูกจัดเป็นค่าที่ว่างเพื่อรอการตรวจสอบและเติมเต็มในขั้นตอนถัดไป อย่างไรก็ตาม จำนวนค่าที่ไม่ถูกต้อง (Invalid) ได้รับการแก้ไขจนหมด

ทำให้มั่นใจได้ว่าข้อมูลที่เหลืออยู่ทั้งหมดเป็นไปตามรูปแบบมาตรฐาน แม้จะยังมีบางส่วนที่ต้องอาศัยการตรวจสอบเพิ่มเติมจากหน่วยงานที่เกี่ยวข้อง

ทั้งนี้ จากจำนวนข้อมูลทั้งหมด 18,177 แถว พบว่าสามารถแมป SUBDISTRICT_CODE ได้สำเร็จจำนวน 14,979 แถว คิดเป็นร้อยละ 82.42 ของข้อมูลทั้งหมด แสดงให้เห็นว่ากระบวนการ Cleansing และการแมปกับตารางอ้างอิงสามารถยกระดับความถูกต้องของข้อมูลได้อย่างมีนัยสำคัญ แม้ยังมีบางส่วนที่ต้องดำเนินการเสริมต่อไป

ตารางที่ 3 REG_STUDENT

1. ภาพรวมตารางและโครงสร้างข้อมูล

ตาราง REG_STUDENT เป็นตารางหลักของระบบทะเบียนนักศึกษา ใช้สำหรับจัดเก็บข้อมูลประจำตัวและสถานะทางการศึกษาของนักศึกษาแต่ละราย โดยมีความเชื่อมโยงกับตารางสำคัญอื่น ๆ เช่น FACULTY, MAJOR และ CURRICULUM ข้อมูลในตารางนี้ครอบคลุมทั้งด้านอัตลักษณ์ของนักศึกษา ข้อมูลการเข้าศึกษา ข้อมูลด้านหลักสูตร และสถานะการศึกษา ซึ่งล้วนมีบทบาทสำคัญต่อการบริหารจัดการข้อมูลในระบบทะเบียน

โดยตาราง REG_STUDENT มีจำนวนข้อมูลประกอบด้วย 34 คอลัมน์ รวมทั้งสิ้น 17,952 แถว ครอบคลุมข้อมูลทั้งในเชิงอัตลักษณ์และเชิงวิชาการ โดยมี STUDENT_ID เป็น Primary Key ข้อมูลที่บันทึกไว้ในตารางนี้ถูกนำไปใช้งานต่อทั้งในการรายงานภายในมหาวิทยาลัย การเชื่อมโยงกับระบบสารสนเทศอื่น และการจัดส่งให้หน่วยงานภายนอก เช่น สำนักงานการอุดมศึกษา จึงจำเป็นต้องมีการตรวจสอบและปรับปรุงคุณภาพข้อมูลให้ถูกต้องและครบถ้วนอยู่เสมอ

2. ผลการทำ Data Profiling เบื้องต้น

ในการตรวจสอบคุณภาพข้อมูลของตาราง REG_STUDENT ผู้วิจัยได้ทำการสำรวจค่าที่ขาดหาย ความถูกต้องของรูปแบบข้อมูล และความครบถ้วนของข้อมูลในคอลัมน์ที่สำคัญต่อการระบุตัวตนและการจัดการด้านการศึกษา ผลการตรวจสอบดังแสดงในตารางสรุป

ตาราง 15 ผลการตรวจสอบข้อมูลตาราง REG_STUDENT

คอลัมน์	เงื่อนไขการตรวจสอบ	ผลการตรวจสอบ	หมายเหตุ
STUDENT_ID	ห้ามว่าง, จำนวน 6 หลัก	ค่าว่าง 1 (0.01%) จำนวนไม่อยู่ในกำหนด 122 (0.68%)	-
YEAR_ENTRY	ห้ามว่าง	ค่าว่าง 529 (2.95%)	-
SEMESTER_ENTRY	ห้ามว่าง	ค่าว่าง 613 (3.41%)	-
STUDENT_TYPE_CODE		ค่าว่าง 528 (2.94%)	-
TITLE_THI	ห้ามว่าง	ค่าว่าง 69 (0.38%)	-
NAME_THA	ห้ามว่าง	ค่าว่าง 70 (0.39%)	-

GENDER_CODE	ห้ามว่าง	ค่าว่าง 558 (3.11%)	-
BIRTHDATE	ห้ามว่าง	ค่าว่าง 1 (0.01%)	-
ADMIT_DATE	ห้ามว่าง	ค่าว่าง 0 (0.00%)	-
PERSONAL_ID	ห้ามว่าง	ค่าว่าง 1,427 (7.95%)	-
RACE_CODE	ห้ามว่าง	ค่าว่าง 1,438 (8.01%)	-
NATIONALITY_CODE	ห้ามว่าง	ค่าว่าง 1,436 (8.00%)	-
RELIGION_CODE	ห้ามว่าง	ค่าว่าง 1,462 (8.14%)	-
FACULTY_CODE	ห้ามว่าง	ค่าว่าง 528 (2.94%)	-
MAJOR_CODE	ห้ามว่าง	ค่าว่าง 858 (4.78%)	-
CURRICULUM_ID	ห้ามว่าง	ค่าว่าง 6,132 (34.16%)	-
CGPA	ห้ามว่าง	ค่าว่าง 0 (0.00%)	-
STATUS_CODE	ห้ามว่าง	ค่าว่าง 527 (2.94%)	-

จากผลการตรวจสอบพบว่า ปัญหาหลักของตารางนี้คือการมีค่าที่ขาดหายจำนวนมากในบางคอลัมน์ เช่น รหัสหลักสูตร (CURRICULUM_ID) และข้อมูลอัตลักษณ์บางส่วน เช่น PERSONAL_ID, RACE_CODE, และ RELIGION_CODE ข้อมูลเหล่านี้มีความสำคัญต่อการเชื่อมโยงกับตารางอื่นและการรายงานต่อหน่วยงานภายนอก ดังนั้นจึงจำเป็นต้องมีการปรับปรุงและดำเนินการ Cleansing เพื่อให้ได้ข้อมูลที่ครบถ้วนและถูกต้องมากยิ่งขึ้น

3. วิธีการ Cleansing และเหตุผล

ในการปรับปรุงคุณภาพข้อมูลตาราง REG_STUDENT ผู้วิจัยได้เลือกใช้วิธีการ Cleansing ที่แตกต่างกันตามลักษณะของข้อมูลแต่ละคอลัมน์ โดยพิจารณาจากระดับความซับซ้อนของปัญหา ปริมาณข้อมูลที่ต้องแก้ไข และความเสี่ยงต่อการเกิดข้อผิดพลาด ซึ่งสามารถสรุปแนวทางได้ดังนี้

1. STUDENT_ID

ดำเนินการแก้ไขแบบ Manual โดยใช้ Excel เนื่องจากจำนวนข้อมูลที่ผิดพลาดมีไม่มาก และการแก้ไขจำเป็นต้องอาศัยการตรวจสอบเชิงบริบทเพื่อความถูกต้อง

2. YEAR_ENTRY

ใช้ Rule-Based กำหนดเงื่อนไขเติมค่าจากสองหลักแรกของ STUDENT_ID (เช่น 580037 > ปีการศึกษา 2558) เฉพาะในกรณีที่ YEAR_ENTRY ว่างเท่านั้น เพื่อลดการสูญเสียข้อมูลและรักษาความถูกต้องของข้อมูลเดิมที่มีอยู่แล้ว

3. SEMESTER_ENTRY

ไม่สามารถ Cleansing ได้ เนื่องจากไม่มีข้อมูลอ้างอิงที่แน่นอนและเจ้าของข้อมูลไม่สามารถให้ข้อมูลเพิ่มเติมได้

4. STUDENT_TYPE_CODE

ได้รับการปรับปรุงโดยอัตโนมัติหลังจากแก้ไข STUDENT_ID ที่ไม่ถูกต้อง ทำให้ค่าที่ผิดพลาดได้รับการแก้ไขไปพร้อมกัน

5. TITLE_THA

ใช้ Rule-Based โดยการแมปกับตารางอ้างอิง (Reference Table) เพื่อให้ได้ TITLE_CODE ที่ถูกต้องและมาตรฐาน

6. NAME_THA

แก้ไขด้วย Manual ผ่าน Excel เนื่องจากเป็นข้อมูลข้อความอิสระที่ต้องการการตรวจสอบด้วยสายตาและความเข้าใจภาษา

7. GENDER_CODE

แก้ไขด้วย Manual โดยเปรียบเทียบกับคำนำหน้า (TITLE_THA) ซึ่งช่วยลดจำนวนข้อมูลที่ว่างลงอย่างมาก ทำให้สามารถแก้ไขได้ด้วยทรัพยากรที่น้อย

8. PERSONAL_ID, RACE_CODE, NATIONALITY_CODE, RELIGION_CODE, MAJOR_CODE

ไม่สามารถเติมค่าที่ว่างได้เนื่องจากเป็นข้อมูลเฉพาะที่ต้องยืนยันกับผู้เกี่ยวข้อง ผู้วิจัยจึงออกรายงานสรุปข้อมูลที่ผิดพลาดและส่งต่อให้หน่วยงานที่รับผิดชอบตรวจสอบ นอกจากนี้ ข้อมูลที่มีอยู่และถูกต้องถูกนำเข้า Rule-Based Matching ผ่านเครื่องมือ Python GUI ที่พัฒนาขึ้นเพื่อทำการแมปกับตารางรหัสมาตรฐานสำหรับใช้งานต่อไป

9. FACULTY_CODE, STATUS_CODE

ปัญหาค่าว่างได้รับการแก้ไขโดยอัตโนมัติหลังจาก STUDENT_ID ถูกปรับปรุง ทำให้ไม่จำเป็นต้องใช้วิธีการเพิ่มเติม

10 CURRICULUM_ID

พบว่ามีค่าว่างสูงถึง 31.94% ของข้อมูลทั้งหมด และจากการทดลองใช้ Rule-Based Matching (FACULTY_CODE + MAJOR_CODE + YEAR_ENTRY กับตาราง REG_CURRICULUM_VERSION) พบว่ามีข้อผิดพลาดถึง 10.66% ของข้อมูลที่ตรวจสอบ เทียบกับข้อมูลเดิมที่มีอยู่แล้ว ผลลัพธ์ดังกล่าวสะท้อนว่า Rule-Based มีความเสี่ยงสูงต่อการแมปข้อมูลผิดพลาด จึงไม่เหมาะสมที่จะใช้เป็นแนวทางการแก้ไขอัตโนมัติ ผู้วิจัยจึงเลือกที่จะ ไม่ใช้ Rule-Based และดำเนินการส่งข้อมูลดังกล่าวให้ ส่วนงานที่เกี่ยวข้องตรวจสอบทั้งหมด เพื่อให้มั่นใจว่าข้อมูล CURRICULUM_ID มีความถูกต้องสมบูรณ์ในระดับที่ใช้งานได้จริง

จากการเลือกใช้วิธีการที่แตกต่างกันตามลักษณะของข้อมูล พบว่า Rule-Based เหมาะสำหรับการแก้ไขข้อมูลที่มีรูปแบบชัดเจนและสามารถอ้างอิงกับรหัสมาตรฐานได้ ขณะที่ข้อมูลเชิงข้อความ

อิสระหรือข้อมูลที่ต้องใช้การตีความจำเป็นต้องอาศัย Manual Correction ส่วนข้อมูลเชิงอัตลักษณ์ที่ว่างจำนวนมากและไม่สามารถสรุปได้ด้วยกฎเกณฑ์ จึงถูกจัดให้อยู่ในกลุ่มที่ ต้องส่งตรวจสอบโดยเจ้าของข้อมูล การผสมผสานวิธีการดังกล่าวทำให้สามารถรักษาความถูกต้อง ความสมบูรณ์ และความน่าเชื่อถือของข้อมูลได้อย่างเหมาะสม

ตาราง 16 ผลการตรวจสอบข้อมูลก่อนและหลังการทำ Data Cleansing ของ REG_STUDENT

คอลัมน์	ก่อน Cleansing จำนวน (ร้อยละ)	หลัง Cleansing จำนวน (ร้อยละ)	**หมายเหตุ**
STUDENT_ID	ค่าว่าง 1 (0.01%) จำนวนไม่อยู่ในกำหนด 122 (0.68%)	ค่าว่าง 0 (0.00%) จำนวนไม่อยู่ในกำหนด 0 (0.00%)	
YEAR_ENTRY	ค่าว่าง 529 (2.95%)	ค่าว่าง 0 (0.00%)	
SEMESTER_ENTRY	ค่าว่าง 613 (3.41%)	ค่าว่าง 613 (3.41%)	ไม่มีมูลอ้างอิงเพื่อแก้ไข
STUDENT_TYPE_CODE	ค่าว่าง 528 (2.94%)	ค่าว่าง 0 (0.00%)	
TITLE_THI	ค่าว่าง 69 (0.38%)	ค่าว่าง 4 (0.02%)	
NAME_THA	ค่าว่าง 70 (0.39%)	ค่าว่าง 5 (0.03%)	
GENDER_CODE	ค่าว่าง 558 (3.11%)	ค่าว่าง 0 (0.00%)	
BIRTHDATE	ค่าว่าง 1 (0.01%)	ค่าว่าง 0 (0.00%)	
PERSONAL_ID	ค่าว่าง 1,427 (7.95%)	ค่าว่าง 1,427 (7.95%)	
RACE_CODE	ค่าว่าง 1,438 (8.01%)	ค่าว่าง 859 (4.95%)	
NATIONALITY_CODE	ค่าว่าง 1,436 (8.00%)	ค่าว่าง 856 (4.93%)	
RELIGION_CODE	ค่าว่าง 1,462 (8.14%)	ค่าว่าง 884 (5.09%)	
FACULTY_CODE	ค่าว่าง 528 (2.94%)	ค่าว่าง 0 (0.00%)	
MAJOR_CODE	ค่าว่าง 858 (4.78%)	ค่าว่าง 316 (1.82%)	
CURRICULUM_ID	ค่าว่าง 6,132 (34.16%)	ค่าว่าง 5,546 (31.94%)	
STATUS_CODE	ค่าว่าง 527 (2.94%)	ค่าว่าง 2 (0.01%)	

ข้อจำกัดของการทำ Data Cleansing แม้ว่างานวิจัยนี้จะประสบความสำเร็จในการปรับปรุงคุณภาพข้อมูล แต่ยังมีข้อจำกัดที่ต้องพิจารณา ได้แก่

1. **ข้อจำกัดด้านข้อมูลอ้างอิง**: ข้อมูลบางฟิลด์ไม่มีแหล่งอ้างอิงมาตรฐานที่เพียงพอ จึงไม่สามารถตรวจสอบหรือแก้ไขได้ครบถ้วน เช่น SEMESTER_ENTRY
2. **ข้อจำกัดด้านทรัพยากร**: การใช้ Machine Learning และ Text Clustering ต้องอาศัยเวลาประมวลผลและพลังงานคอมพิวเตอร์สูง โดยเฉพาะเมื่อข้อมูลมีปริมาณมาก
3. **ข้อจำกัดด้านความถูกต้องของข้อมูลต้นทาง**: แม้จะมีการ Cleansing แล้ว แต่หากข้อมูลที่บันทึกในระบบเดิมผิดพลาดโดยเนื้อแท้ (เช่น กรอกค่าผิดจริง) ก็ไม่สามารถแก้ไขได้โดยอัตโนมัติ จำเป็นต้องอาศัยการยืนยันจากหน่วยงานที่เกี่ยวข้อง

4. ข้อจำกัดด้านการใช้ Manual Cleaning: แม้จะช่วยแก้ไขปัญหาเฉพาะหน้าได้ แต่การพึ่งพาการแก้ไขแบบ Manual ไม่สามารถรองรับปริมาณข้อมูลขนาดใหญ่ และอาจเสี่ยงต่อความผิดพลาดจากมนุษย์ (Human Error)

4.4 การออกแบบและพัฒนาระบบต้นแบบ (Prototype System Design)

แม้ว่าการทำ Data Cleansing จะยังมีข้อจำกัดบางประการดังที่กล่าวไว้ในหัวข้อ 4.3 แต่ข้อมูลที่ผ่านการตรวจสอบและปรับปรุงแล้วมีคุณภาพเพียงพอที่จะนำไปใช้ในการออกแบบและพัฒนาระบบต้นแบบได้ เพื่อแสดงให้เห็นถึงความเป็นไปได้ของการใช้งานจริง งานวิจัยนี้จึงได้ออกแบบระบบต้นแบบ (Prototype) ที่อ้างอิงโครงสร้างข้อมูลใหม่ และทดสอบด้วยข้อมูลที่ผ่านการ Cleansing แล้ว โดยระบบต้นแบบนี้มีวัตถุประสงค์เพื่อประเมินความถูกต้อง ความสมบูรณ์ และความสอดคล้องของข้อมูล รวมทั้งทดสอบความสามารถของระบบในการสนับสนุนการปฏิบัติงานในอนาคต

4.4.1 กรอบแนวคิดของระบบต้นแบบ (System Framework)

ระบบต้นแบบที่พัฒนาขึ้นในงานวิจัยนี้มีวัตถุประสงค์หลักเพื่อใช้เป็นกลไกทดสอบและประเมินคุณภาพของข้อมูลหลังการ Cleansing โดยมุ่งเน้นไปที่สามประเด็นสำคัญ ได้แก่

1. การแสดงรายงานข้อมูลนักศึกษา ที่มีความถูกต้องและครบถ้วนมากขึ้น เมื่อเปรียบเทียบกับระบบฐานข้อมูลเดิม
2. การเปรียบเทียบความถูกต้องกับระบบเก่า เพื่อยืนยันว่าการ Cleansing ช่วยลดความซ้ำซ้อนและปรับข้อมูลให้เป็นมาตรฐาน
3. การประเมินระบบโดยผู้รับผิดชอบ (เช่น เจ้าหน้าที่ทะเบียน) เพื่อสะท้อนความถูกต้อง ความสะดวก และความเป็นไปได้ในการใช้งานจริง

ระบบต้นแบบนี้ถูกออกแบบโดยอิงแนวคิด Clean Architecture สำหรับ Backend และ Model–View–Controller (MVC) สำหรับ Frontend เพื่อแยกหน้าที่และเพิ่มความยืดหยุ่นในการพัฒนาและบำรุงรักษา

Backend: Clean Architecture แบ่งเป็น 4 ชั้น คือ

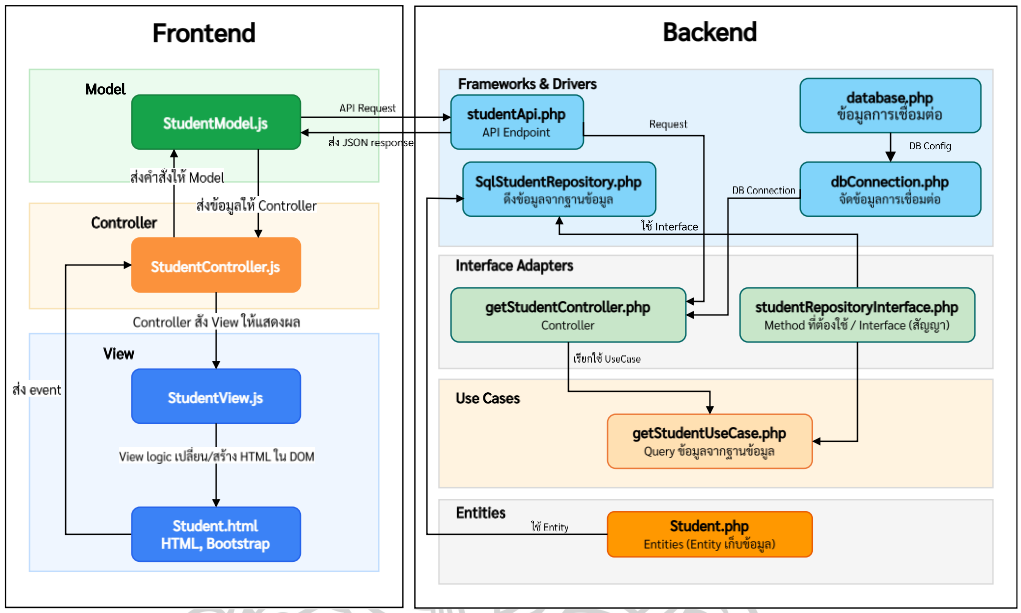
1. **Entities (Domain)** โครงสร้างข้อมูลและกฎธุรกิจ เช่น นักศึกษา รายวิชา หลักสูตร
2. **Use Cases (Application)** กระบวนการหลัก เช่น การค้นหานักศึกษา การออกรายงาน
3. **Interface Adapters** ตัวกลางเชื่อม Use Cases กับ Database และ API ภายนอก
4. **Frameworks & Drivers** เครื่องมือ เช่น PostgreSQL, Web Framework, UI Library

Frontend: MVC แยกการทำงานออกเป็น 3

1. Model จัดการข้อมูลและตรรกะ เช่น ดึงข้อมูลจาก API และเตรียมก่อนแสดงผล

2. View แสดงผลผู้ใช้ (UI) โดยใช้ HTML/CSS/Bootstrap และ DataTables

3. Controller รับคำสั่งจากผู้ใช้ (เช่น การค้นหา คลิกเมนู) ประสานงานกับ Model และส่งต่อข้อมูลไปยัง View



ภาพที่ 15 ออกแบบระบบต้นแบบด้วย Layer-Based Design

4.4.2 การออกแบบระบบเชิงกระบวนการ (Workflow-Based System Design)

เพื่อแสดงลำดับการทำงานของระบบต้นแบบในกรณีใช้งานจริง งานวิจัยเลือกนำเสนอกระบวนการ “ค้นหาและแสดงข้อมูลนักศึกษา” ซึ่งเป็นฟังก์ชันหลักที่อาศัยชุดข้อมูลที่ผ่านการทำ Data Cleansing แล้ว โดยระบบจะดึงข้อมูลที่ตรงกับเงื่อนไขผ่าน API ไปยังฐานข้อมูลใหม่ และแสดงผลรายชื่อนักศึกษาที่ถูกต้องมากขึ้น (ลดความซ้ำซ้อน/ความผิดรูปแบบ) พร้อมความสามารถในการออกรายงานสำหรับใช้งานต่อไปได้โดยตรง (เช่น PDF หรือ Excel)

1. การยืนยันตัวตนและเข้าใช้ระบบ: เจ้าหน้าที่ผู้รับผิดชอบเข้าสู่ระบบด้วยบัญชีที่ได้รับสิทธิ์ เพื่อใช้ฟังก์ชันค้นหา/รายงานของต้นแบบ (สิทธิ์เฉพาะเพื่อความปลอดภัยของข้อมูล)

2. กำหนดเงื่อนไขการค้นหา: ผู้ใช้ระบุเกณฑ์ค้นหา (เช่น รหัสนักศึกษา/ชื่อ/คณะ ฯลฯ) จากนั้น Controller จะประสานกับ Model/Use Case เพื่อเรียก API ไปยังฐานข้อมูลที่ผ่านการ Cleansing แล้ว

3. ประมวลผลและดึงข้อมูลที่ถูกต้อง: ระบบประมวลผลตามกฎธุรกิจและโครงสร้างข้อมูลใหม่ เพื่อคืนผลลัพธ์ที่ “สะอาด” มากขึ้น เช่น ไม่มีค่าซ้ำซ้อน/สะกดผิด/ค่าว่างที่ผิดรูปแบบ ก่อนส่งให้ส่วนแสดงผล (View) นำเสนอแก่ผู้ใช้

4. นำเสนอผลลัพธ์และออกรายงาน: แสดงรายชื่อนักศึกษาที่ตรงเงื่อนไข และรองรับ การออกรายงาน เพื่อนำไปใช้งานในงานเอกสารหรือส่งต่อหน่วยงานภายนอกได้ทันที (รูปแบบ PDF/Excel)

ภาพที่ 16 ออกแบบระบบต้นแบบด้วย Workflow-Based Design

แสดง Workflow-Based Design สำหรับกระบวนการค้นหาและแสดงข้อมูลนักศึกษาในระบบต้นแบบ

4.4.3 แผนภาพกรณีการใช้งานของระบบ (Use Case Diagram)

เพื่อแสดงความสัมพันธ์ระหว่างผู้ใช้งานและฟังก์ชันของระบบต้นแบบ งานวิจัยนี้ได้จัดทำ Use Case Diagram เพื่อกำหนดบทบาทของผู้ใช้งานหลัก และกิจกรรมที่สามารถดำเนินการได้ภายในระบบ

Actor หลักของระบบต้นแบบ

1. เจ้าหน้าที่ทะเบียน (Registrar Officer) – มีสิทธิ์เข้าสู่ระบบเพื่อค้นหาข้อมูลนักศึกษา ออกรายงาน และตรวจสอบคุณภาพข้อมูลที่ผ่านการ Cleansin

2. ผู้ดูแลระบบ (System Administrator) – ดูแลการกำหนดสิทธิ์ผู้ใช้งานและการจัดการระบบเบื้องต้น (ในขอบเขตของ Prototype กำหนดเฉพาะเพื่อรองรับการทดลองใช้งาน)

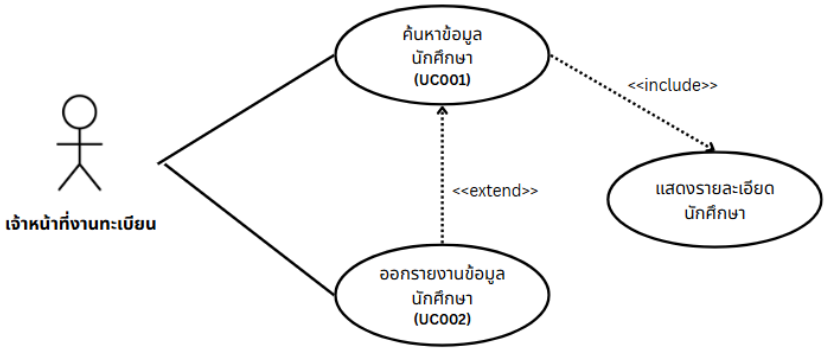
Use Case หลัก

1. เข้าสู่ระบบ (Login) – ยืนยันสิทธิ์ผู้ใช้งาน

2. ค้นหาข้อมูลนักศึกษา (Search Student Information) – ระบุเงื่อนไขการค้นหา เช่น รหัสนักศึกษา ชื่อ–สกุล คณะ หรือหลักสูตร

3. แสดงผลข้อมูลที่ถูกต้อง (Display Cleansed Student Data) – แสดงข้อมูลที่ผ่านการ Cleansing แล้ว ลดปัญหาค่าว่าง ข้อมูลซ้ำซ้อน หรือการสะกดผิด

4. ออกรายงาน (Export Report) – ส่งออกผลการค้นหาในรูปแบบ PDF หรือ Excel เพื่อใช้งานจริงหรือส่งต่อหน่วยงานภายนอก



ภาพที่ 17 ออกแบบระบบต้นแบบด้วย Use Case Diagram

แสดง Use Case Diagram สำหรับระบบต้นแบบการค้นหาและแสดงข้อมูลนักศึกษา โดยแสดงความสัมพันธ์ระหว่างผู้ใช้งาน (เจ้าหน้าที่ทะเบียน) กับฟังก์ชันหลักของระบบ

4.4.4 คำอธิบายกรณีการใช้งาน (Use Case Description)

เพื่อให้เห็นรายละเอียดการทำงานของระบบต้นแบบอย่างเป็นลำดับ งานวิจัยนี้ได้จัดทำคำอธิบายกรณีการใช้งาน (Use Case Description) โดยอธิบาย Actor ที่เกี่ยวข้อง, เงื่อนไขก่อนหน้า (Precondition), ขั้นตอนการทำงานหลัก (Flow of Events), เงื่อนไขหลังการทำงาน (Postcondition), และข้อยกเว้น (Exception) ของแต่ละกรณี

ตาราง 17 Use Case Description

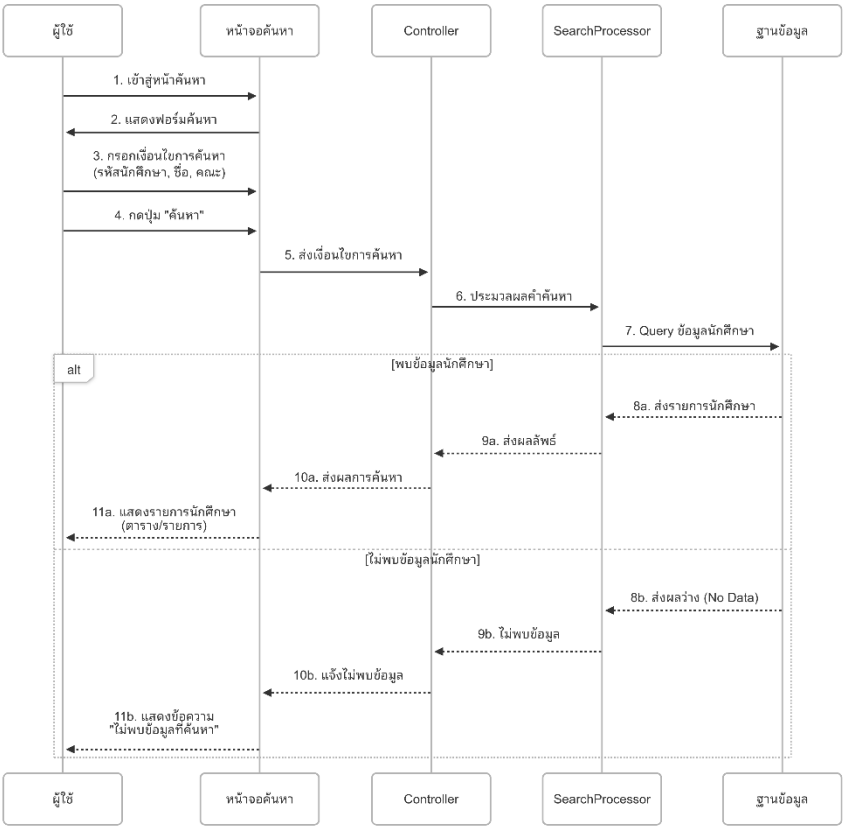
USE CASE	Use Case Name: **ค้นหาข้อมูลนักศึกษา**
Actor:	เจ้าหน้าที่ทะเบียน
Goal:	ให้เจ้าหน้าที่สามารถค้นหาข้อมูลนักศึกษาจากฐานข้อมูลที่ผ่านการทำ Data

	Cleansing แล้ว เพื่อให้แสดงผลข้อมูลได้อย่างถูกต้องและครบถ้วน เช่น การแสดงชื่อ จังหวัด คณะ และรายละเอียดที่เกี่ยวข้อง
Precondition:	a. เจ้าหน้าที่ต้องเข้าสู่ระบบด้วยบัญชีที่ได้รับสิทธิ์ b. ฐานข้อมูลที่ใช้แสดงผลต้องเป็นข้อมูลที่ผ่านการ Cleansing แล้ว
Trigger:	เจ้าหน้าที่ต้องการตรวจสอบ/ค้นหาข้อมูลของนักศึกษาผ่านระบบต้นแบบ
Main Success Scenario (Basic Flow)	
1. เจ้าหน้าที่เข้าสู่ระบบสำเร็จ 2. เลือกเมนู “ค้นหาข้อมูลนักศึกษา” 3. กรอกเงื่อนไขการค้นหา เช่น รหัสนักศึกษา ชื่อ หรือจังหวัด 4. ระบบประมวลผลคำค้นโดยดึงข้อมูลจากฐานข้อมูลที่ผ่านการ Cleansing 5. ระบบแสดงรายการนักศึกษาที่ตรงตามเงื่อนไข 6. เจ้าหน้าที่สามารถกดดูรายละเอียดเพิ่มเติมของแต่ละคน หรือออกรายงานรายชื่อได้	
Alternative Flow (Alternative Scenario)	
กรณี 1: ไม่พบข้อมูลตรงเงื่อนไข 1. ระบบแจ้งว่า “ไม่พบข้อมูล” 2. เจ้าหน้าที่สามารถกลับไปกรอกเงื่อนไขใหม่ **กรณี 2: กรอกเงื่อนไขไม่ครบหรือไม่ถูกต้อง** 1. ระบบแจ้งเตือน “กรุณากรอกเงื่อนไขให้ครบถ้วน” **2.** เจ้าหน้าที่สามารถแก้ไขและส่งคำค้นหาใหม่ได้	
Postcondition:	1. ระบบแสดงข้อมูลนักศึกษาที่ผ่านการ Cleansing แล้วอย่างถูกต้อง 2. เจ้าหน้าที่สามารถดาวน์โหลดหรือออกรายงานได้
Exceptions:	1. ระบบไม่สามารถประมวลผลการค้นหาได้เนื่องจากคำค้นมีอักขระพิเศษหรือรูปแบบไม่ถูกต้อง 2. เซิร์ฟเวอร์ตอบสนองล่าช้าหรือไม่พร้อมใช้งานชั่วคราว
Assumptions:	1. เจ้าหน้าที่มีสิทธิ์เข้าถึงข้อมูลทั้งหมด 2. ระบบทำงานกับฐานข้อมูลที่ผ่านการ Cleansing และเป็นข้อมูลล่าสุด
Related Use Case:	1. แสดงรายละเอียดนักศึกษา (Include) 2. ออกรายงานรายชื่อนักศึกษา (Extend)
USE CASE	**Use Case Name: ออกรายงานข้อมูลนักศึกษา**
Actor:	เจ้าหน้าที่ทะเบียน
Goal:	ให้เจ้าหน้าที่สามารถออกรายงานรายชื่อนักศึกษาที่ตรงตามเงื่อนไขการค้นหาในรูปแบบ PDF หรือ Excel ได้อย่างถูกต้อง โดยใช้ข้อมูลที่ผ่านการ Cleansing

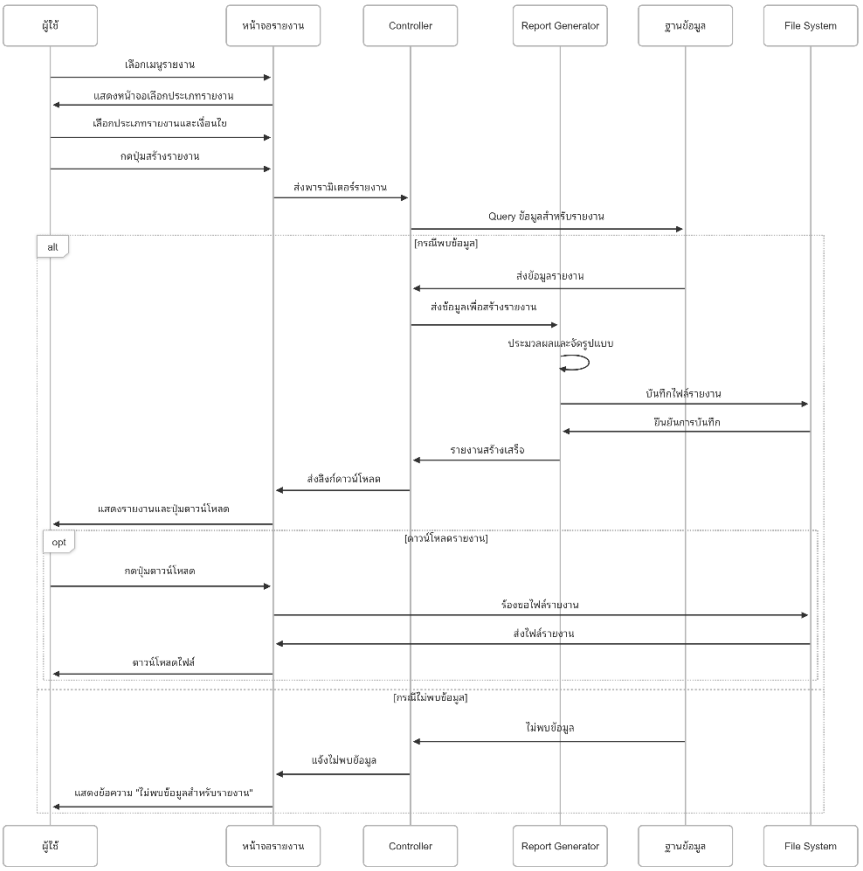
	แล้ว
Precondition:	1. เจ้าหน้าที่ได้เข้าสู่ระบบแล้ว 2. มีการค้นหาข้อมูลนักศึกษาเสร็จสิ้น และมีรายการแสดงบนหน้าจอ 3. ระบบสามารถเชื่อมต่อกับบริการสร้างเอกสารหรือระบบรายงานได้
Trigger:	เจ้าหน้าที่ต้องการออกรายงานจากผลการค้นหาข้อมูลนักศึกษาที่ปรากฏในหน้าจอ
Main Success Scenario (Basic Flow)	
1. เจ้าหน้าที่เข้าสู่ระบบ และดำเนินการค้นหาข้อมูลนักศึกษา 2. ระบบแสดงผลลัพธ์รายการนักศึกษาที่ตรงเงื่อนไข 3. เจ้าหน้าที่คลิกปุ่ม “ออกรายงาน” และเลือกประเภทของรายงาน (PDF หรือ Excel) 4. ระบบสร้างไฟล์รายงานโดยใช้ข้อมูลที่แสดงผลอยู่ **5.** ระบบดาวน์โหลดไฟล์ให้เจ้าหน้าที่ หรือแสดงลิงก์ให้ดาวน์โหลด	
Alternative Flow (Alternative Scenario)	
กรณี 1: ไม่มีข้อมูลให้รายงาน 1. หากไม่มีข้อมูลในผลการค้นหา ระบบจะแจ้งว่า "ไม่มีข้อมูลให้ออกรายงาน" 2. กลับไปค้นหาใหม่ **กรณี 2: เจ้าหน้าที่ยกเลิกการออกรายงาน** หากยกเลิกก่อนการดาวน์โหลด ระบบจะไม่สร้างไฟล์ และกลับสู่หน้ารายการตามปกติ	
Postcondition:	1. เจ้าหน้าที่ได้รับไฟล์รายงานสำเร็จ 2. ระบบบันทึก log การออกรายงานไว้ (ถ้ามี)
Exceptions:	1. การสร้างไฟล์รายงานล้มเหลว (เช่น library ไม่พร้อมใช้งาน) 2. ขนาดข้อมูลมากเกินระบบรองรับ ระบบแจ้งให้กรองเงื่อนไขเพิ่มเติมก่อนออกรายงาน 3. ปัญหาการเชื่อมต่อ เช่น ไม่สามารถโหลดไฟล์ได้
Assumptions:	1. ข้อมูลที่แสดงในหน้าจอผลลัพธ์ได้ผ่านการ Cleansing แล้ว 2. ระบบต้นแบบมีบริการสร้างรายงาน (PDF/Excel) พร้อมใช้งาน 3. เจ้าหน้าที่มีสิทธิ์ในการเข้าถึงฟีเจอร์นี้
Related Use Case:	ค้นหาข้อมูลนักศึกษา (Include)

4.4.5 แผนภาพลำดับขั้นตอนการทำงานของระบบ (Sequence Diagram)

เพื่อแสดงลำดับการทำงานภายในของระบบต้นแบบอย่างละเอียด งานวิจัยนี้ได้จัดทำ Sequence Diagram สำหรับอธิบายการทำงานของฟังก์ชันหลัก โดยเน้นไปที่กรณี “ค้นหาข้อมูลนักศึกษาและแสดงผล” ซึ่งถือเป็นกระบวนการสำคัญที่สะท้อนคุณภาพข้อมูลหลังการ Cleansing



ภาพที่ 18 ลำดับขั้นตอนการทำงานของระบบต้นแบบ
แสดง Sequence Diagram สำหรับกรณีการค้นหาข้อมูลนักศึกษา



ภาพที่ 19 ลำดับขั้นตอนการทำงานของระบบต้นแบบ

แสดง Sequence Diagram สำหรับกรณีการออกรายงานข้อมูลนักศึกษา

4.4.6 การออกแบบฐานข้อมูลใหม่ (ER Diagram)

การออกแบบฐานข้อมูลใหม่เป็นขั้นตอนสำคัญที่เกิดจากการ Source-to-Target Mapping (STM) และกระบวนการ Data Cleansing โดยมุ่งเน้นให้โครงสร้างตารางรองรับการจัดเก็บข้อมูลที่ถูกต้อง ครบถ้วน และเป็นมาตรฐานเดียวกัน ทั้งนี้ยังแก้ไขข้อจำกัดของระบบฐานข้อมูลเดิม เช่น การซ้ำซ้อนของข้อมูล การใช้รหัสที่ไม่สอดคล้อง และรูปแบบข้อมูลที่ไม่เป็นมาตรฐาน

หลักการออกแบบ

1. ใช้แนวทาง Normalization เพื่อลดความซ้ำซ้อนของข้อมูล
2. ใช้ Primary Key / Foreign Key ชัดเจน เพื่อสร้างความสัมพันธ์ระหว่างตาราง
3. รองรับการขยายตัวในอนาคต (Scalability) และการเชื่อมโยงกับระบบอื่นได้ง่ายขึ้น

แบ่งกลุ่มตารางออกเป็น 4 หมวดใหญ่ ได้แก่

1. ตารางข้อมูลนักศึกษา (Student Data) เช่น REG_STUDENT, REG_STUDENT_ADDRESS

2. ตารางโครงสร้างหลักสูตร (Curriculum Structure Data) เช่น REG_CURRICULUM_VERSION, REG_COURSE_IN_GROUP

3. ตารางการลงทะเบียนเรียน (Student Registration Data) เช่น REG_STUDENT_REGISTRATION

4. ตารางข้อมูลอ้างอิง (Reference Data) เช่น REF_FACULTY, REF_MAJOR, REF_SUBJECT, REF_PREFIX, REF_PROVINCE

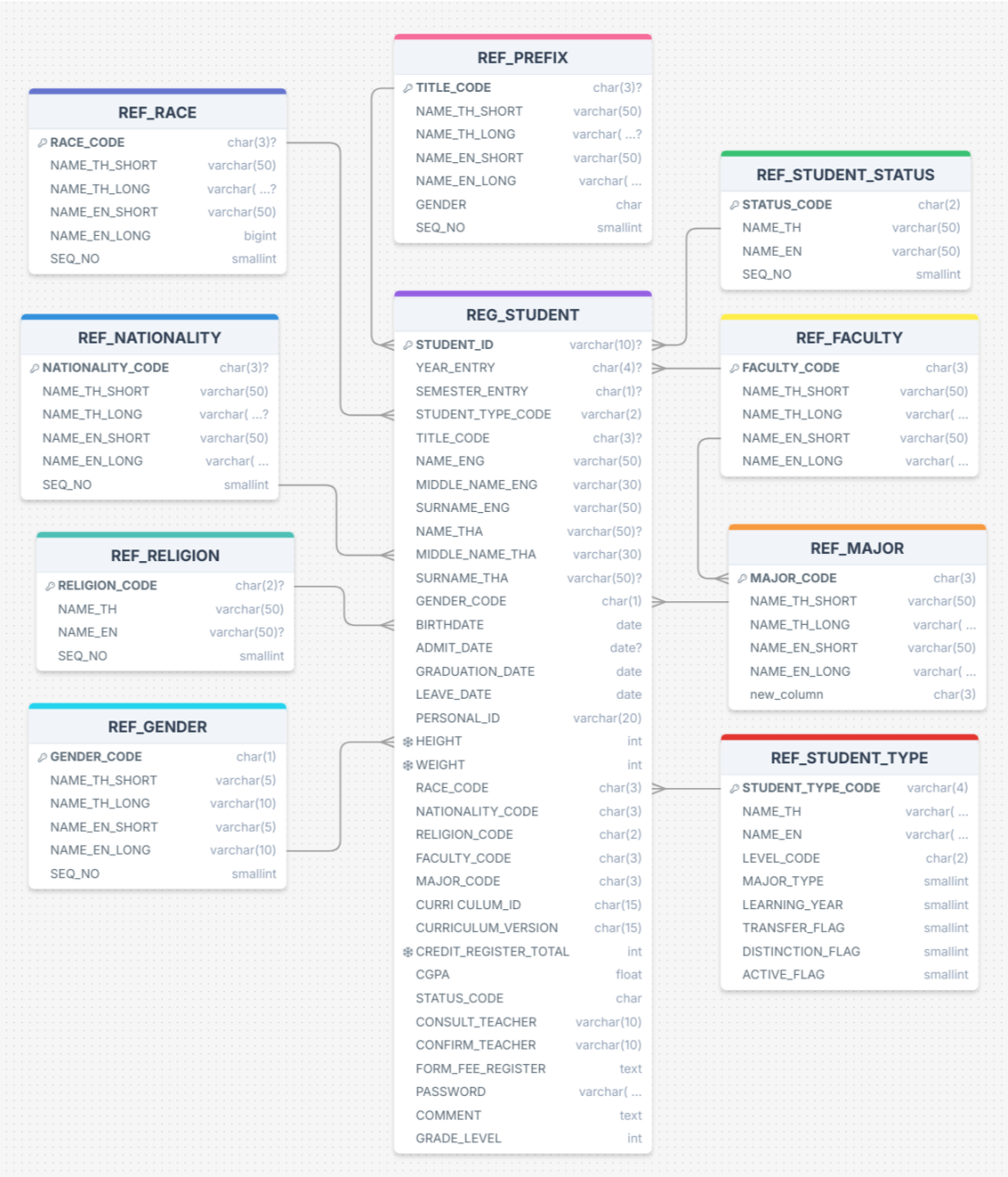
ผลลัพธ์การออกแบบ

1. จากการปรับโครงสร้าง พบว่า ER Diagram ใหม่ สามารถแก้ปัญหาของระบบเดิมและรองรับ Prototype ได้ดีกว่า โดยเฉพาะในมิติ

2. Data Integrity: ความสัมพันธ์นักศึกษา–หลักสูตร–รายวิชาเชื่อมโยงชัดเจน

3. Data Quality: ตารางใหม่ถูกสร้างขึ้นให้รองรับข้อมูลที่ผ่านการ Cleansing แล้ว ลดการซ้ำและค่า Null

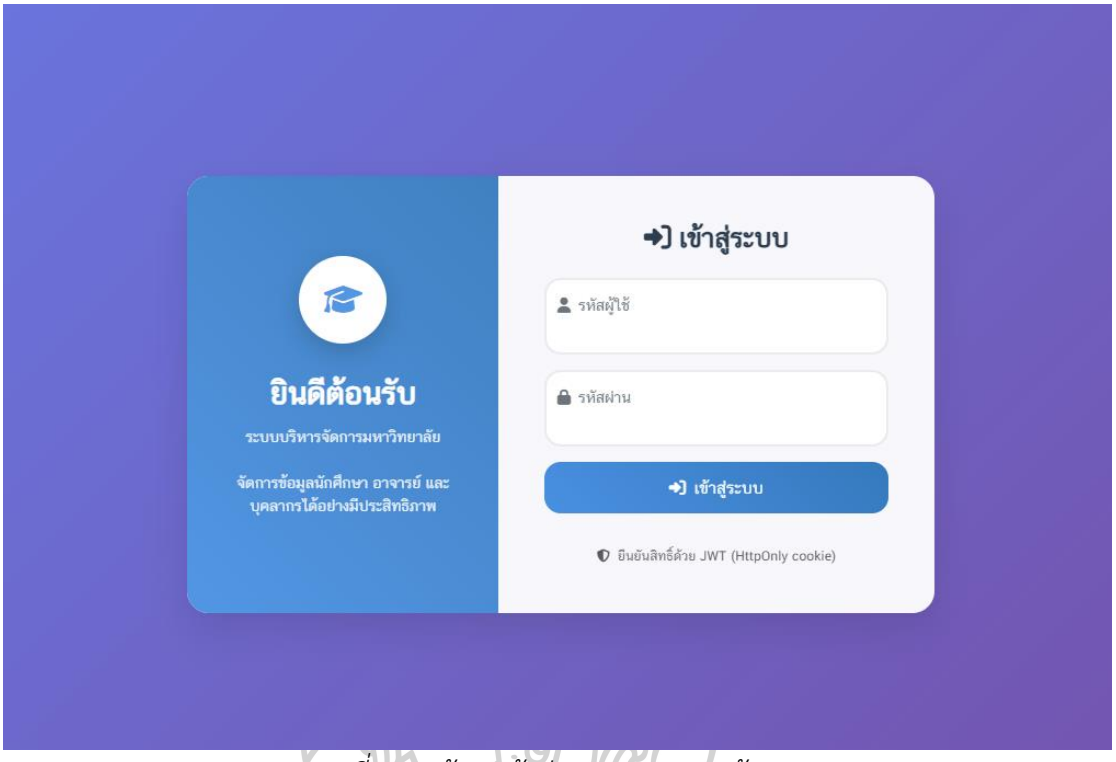
4. Data Usability: รองรับการสร้างรายงานได้ทันที และใช้เป็นฐานสำหรับการพัฒนาโมดูลอื่น ๆ ต่อไป



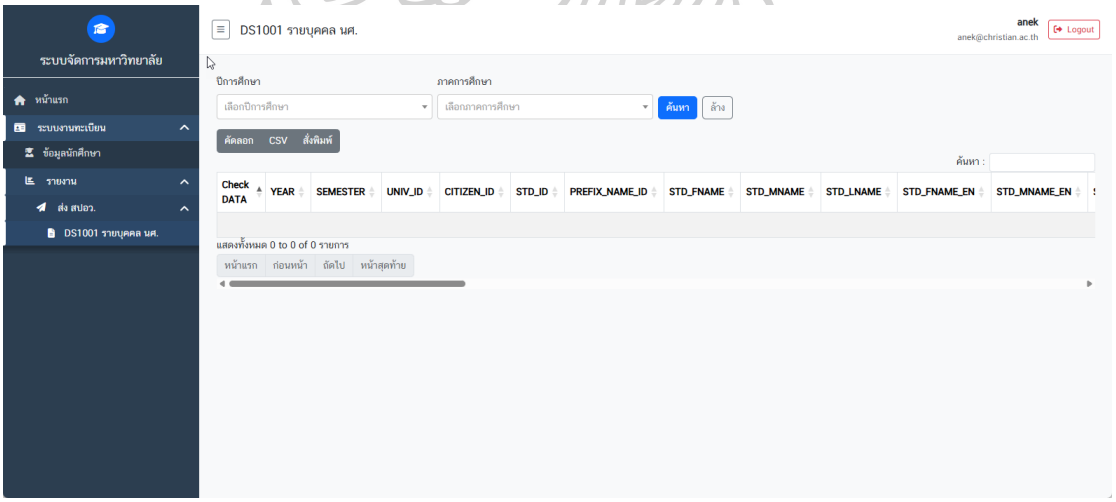
ภาพที่ 20 การออกแบบฐานข้อมูลใหม่ (ER Diagram)

แสดงความสัมพันธ์ระหว่างเอนทิตีหลัก ได้แก่ นักศึกษา หลักสูตร รายวิชา การลงทะเบียนเรียน และตารางอ้างอิง **(ดูทั้งหมดได้ที่ภาคผนวก)**

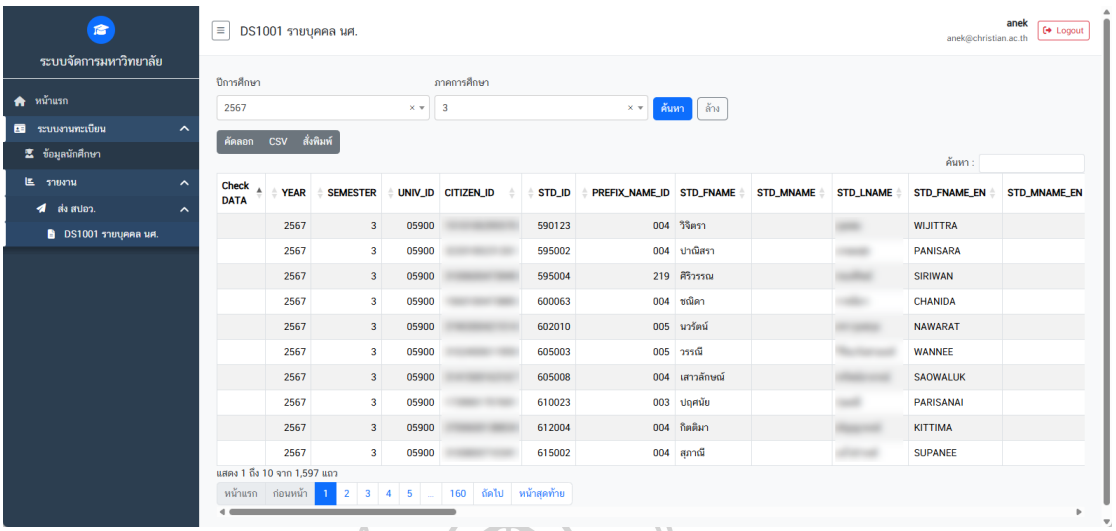
4.4.7 ตัวอย่างส่วนติดต่อผู้ใช้ของระบบต้นแบบ (Prototype UI Screens)



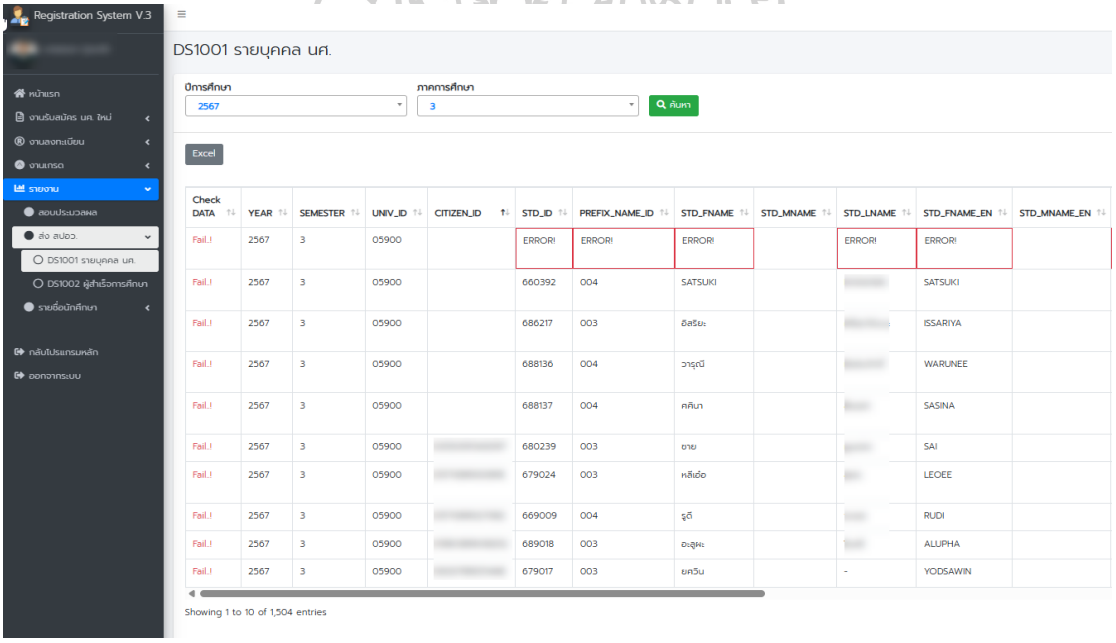
ภาพที่ 21 หน้าจอเข้าสู่ระบบของระบบต้นแบบ



ภาพที่ 22 หน้าจอค้นหาข้อมูลนักศึกษา (Search Form)



ภาพที่ 23 หน้าจอผลการค้นหา (Result Table)



ภาพที่ 24 หน้าจอผลการค้นหา ระบบเดิม (Result Table - Legacy system)

บทที่ 5
ผลการดำเนินงานวิจัย

บทนี้นำเสนอผลการดำเนินงานวิจัยตามขั้นตอนที่ได้กล่าวไว้ในบทที่ 4 โดยมุ่งเน้นการแสดงผลลัพธ์ของการทำ Data Cleansing และการประเมินการทำงานของระบบต้นแบบที่ถูกออกแบบและพัฒนาขึ้นเพื่อรองรับข้อมูลที่ผ่านการปรับปรุงแล้ว การนำเสนอผลการดำเนินงานในบทนี้ประกอบด้วย 3 ส่วนหลัก ได้แก่

1. ผลการทำ Data Cleansing แสดงการเปรียบเทียบคุณภาพของข้อมูลก่อนและหลังการ Cleansing ครอบคลุมประเด็น Missing Values, Duplicate Records, Invalid Formats และ Business Rule Violations พร้อมตารางและสถิติประกอบเพื่อให้เห็นความแตกต่างอย่างชัดเจน

2. ผลการทำงานของระบบต้นแบบ (Prototype Results) นำเสนอผลการทำงานของระบบต้นแบบที่ถูกออกแบบใหม่ โดยแสดงตัวอย่างการค้นหาและรายงานข้อมูลนักศึกษาในรูปแบบที่ถูกต้อง ครบถ้วน และเป็นมาตรฐาน พร้อมการเปรียบเทียบกับข้อมูลที่ได้จากระบบฐานข้อมูลเดิม

3. ผลการประเมินโดยผู้ใช้งาน (User Evaluation) แสดงผลจากการทดสอบระบบต้นแบบโดยเจ้าหน้าที่ที่เกี่ยวข้อง เพื่อประเมินความถูกต้อง ความสะดวกในการใช้งาน และความพึงพอใจตลอดจนข้อเสนอแนะที่ได้รับจากผู้ใช้งานจริง

5.1 ผลการทำ Data Cleansing

การทำ Data Cleansing ครอบคลุมตารางข้อมูลที่เกี่ยวข้องทั้งหมด 21 ตาราง โดยมีเป้าหมายเพื่อแก้ไขปัญหาคุณภาพข้อมูลที่พบในระบบเดิม เช่น ค่า Null, ข้อมูลซ้ำซ้อน, การสะกดที่ไม่ถูกต้อง, รูปแบบข้อมูลไม่สอดคล้อง และความไม่สอดคล้องกับกฎเกณฑ์ทางธุรกิจ (Business Rules) หลังจากดำเนินการตามขั้นตอนที่ได้กล่าวในบทที่ 4 ผลการ Cleansing พบว่า คุณภาพข้อมูลดีขึ้นอย่างมีนัยสำคัญ ทั้งในมิติ Accuracy, Completeness, Consistency และ Validity

5.1.1 สรุปผลการ Cleansing ตามปัญหาที่พบ

จากการตรวจสอบและปรับปรุงคุณภาพข้อมูล สามารถจำแนกผลการแก้ไขได้เป็น 4 กลุ่มปัญหาหลัก ดังนี้

1. Missing Values: มีการเติมค่าจากแหล่งอ้างอิง เช่น YEAR_ENTRY เติมจาก STUDENT_ID, TITLE_THA เติมจาก REF_PREFIX

2. Duplicate Records: มีการลบหรือรวมแถวที่ซ้ำ โดยใช้กฎการเลือกข้อมูลล่าสุด (Latest Record Wins)

3. Invalid Formats: มีการปรับรูปแบบให้เป็นมาตรฐาน เช่น DATE > YYYYMMDD, รหัสวิชา > หลัก

4. Business Rules Violations: ตรวจสอบและแก้ไขค่าให้ตรงตามตารางอ้างอิง เช่น STUDENT_TYPE ต้องตรงกับ REF_STUDENT_TYPE

5.1.2 ตัวอย่างผลการปรับปรุงคุณภาพข้อมูล

ตารางด้านล่างแสดงการเปรียบเทียบคุณภาพข้อมูลก่อนและหลังการทำ Data Cleansing ในตารางสำคัญ

ตาราง 18 เปรียบเทียบคุณภาพข้อมูลก่อนและหลังการทำ Data Cleansing

ฟิลด์ข้อมูล	สถานะ	ก่อน Cleansing (จำนวน / %)	หลัง Cleansing (จำนวน / %)
STUDENT_ID	ค่าว่าง	1 (0.01%)	0 (0.00%)
SUBDISTRICT_CODE	ค่าว่าง	2,453 (13.50%)	3,078 (17.05%)
	คาดว่าไม่ถูกต้อง	502 (2.76%)	0 (0.00%)
DISTRICT	ค่าว่าง	2,102 (11.56%)	2,184 (12.10%)
	คาดว่าไม่ถูกต้อง	3,505 (19.28%)	0 (0.00%)
PROVINCE	ค่าว่าง	2,002 (11.01%)	1,893 (10.48%)
	คาดว่าไม่ถูกต้อง	181 (0.99%)	0 (0.00%)

จากตารางแสดงให้เห็นว่า การทำ Data Cleansing สามารถแก้ไขปัญหาค่าข้อมูลที่ไม่ถูกต้อง (Invalid) ได้อย่างสมบูรณ์ โดยจำนวนค่าที่คาดว่าไม่ถูกต้องในฟิลด์ SUBDISTRICT_CODE, DISTRICT และ PROVINCE ลดลงจากเดิมที่พบร้อยละ 2.76, 19.28 และ 0.99 ตามลำดับ เหลือ 0.00 หลังการ Cleansing ซึ่งสะท้อนถึงความถูกต้อง (Accuracy) และความสอดคล้อง (Consistency) ที่เพิ่มขึ้นอย่างมีนัยสำคัญ ในขณะที่จำนวนค่าที่ว่าง (Missing Values) ของบางฟิลด์ เช่น SUBDISTRICT_CODE และ DISTRICT มีการเพิ่มขึ้นเล็กน้อย เนื่องจากข้อมูลบางส่วนไม่สามารถแมปเข้ากับรหัสมาตรฐานได้โดยอัตโนมัติ และจำเป็นต้องรอการตรวจสอบเพิ่มเติมจากหน่วยงานที่เกี่ยวข้อง อย่างไรก็ตาม ในภาพรวมกระบวนการ Cleansing ช่วยให้คุณภาพข้อมูลดีขึ้นทั้งด้านความครบถ้วน (Completeness) ความถูกต้อง (Accuracy) และความเป็นมาตรฐาน (Validity) ทำให้ข้อมูลที่เหลือมีความน่าเชื่อถือมากขึ้นสำหรับนำไปใช้งานในระบบต้นแบบ

5.2 ผลการทำงานของระบบต้นแบบ (Prototype Results)

ระบบต้นแบบที่พัฒนาขึ้นถูกออกแบบให้ทำงานบนฐานข้อมูลใหม่ที่ผ่านการทำ Data Cleansing แล้ว โดยมีหน้าที่หลักในการแสดงรายงานข้อมูลนักศึกษาในรูปแบบที่ถูกต้อง ครบถ้วน และเป็นมาตรฐานมากขึ้น พร้อมทั้งรองรับการส่งออกข้อมูลในรูปแบบ PDF และ Excel เพื่อนำไปใช้งานจริง ทั้งนี้การทำงานของระบบต้นแบบได้ถูกนำไปทดสอบกับผู้ใช้ที่เกี่ยวข้อง และผลการทำงานสามารถสรุปได้ดังนี้

5.2.1 ความถูกต้องของข้อมูลที่แสดงผล

เมื่อเปรียบเทียบผลการแสดงข้อมูลนักศึกษาระหว่าง ระบบต้นแบบ และ ระบบฐานข้อมูลเดิม (Legacy System) พบว่า

1. ข้อมูลซ้ำซ้อน (Duplicate Records) ที่ปรากฏในระบบเดิมถูกกำจัดออกไปในระบบต้นแบบ

2. ข้อมูลการสะกดที่ไม่เป็นมาตรฐาน เช่น ชื่ออำเภอและจังหวัดที่สะกดต่างกัน ถูกทำให้สอดคล้องกันตาม Reference Table ที่ใช้ใน Data Cleansing

3. ค่าที่ไม่ถูกต้อง (Invalid Values) เช่น Grade ผิดรูปแบบ หรือค่าว่างที่ผิดกฎเกณฑ์ ไม่ปรากฏในผลลัพธ์ของระบบต้นแบบ

ผลการทดสอบนี้ยืนยันว่าการใช้ข้อมูลหลังการ Cleansing ทำให้รายงานที่สร้างขึ้นมีความถูกต้องและเชื่อถือได้มากกว่า

5.2.2 ความรวดเร็วในการทำงาน

จากการทดลองใช้งานจริง พบว่า เวลาที่ใช้ในการสืบค้นและแสดงผลรายงาน ของระบบต้นแบบสั้นลงเมื่อเทียบกับระบบเดิม ทั้งนี้เนื่องจาก

1. การออกแบบโครงสร้างฐานข้อมูลใหม่ตาม ER Diagram ทำให้การ Query มีประสิทธิภาพมากขึ้น

2 .การลดข้อมูลซ้ำซ้อนและปรับรูปแบบข้อมูลให้อยู่ในมาตรฐานเดียวกัน ช่วยลดภาระการประมวลผลในขั้นตอนการดึงข้อมูล

5.2.3 การแสดงผลรายงานและการส่งออก (Reporting & Export)

ระบบต้นแบบสามารถสร้างรายงานได้หลากหลายรูปแบบ เช่น รายงานรายชื่อนักศึกษาตามคณะ/หลักสูตร ผลการศึกษา และที่อยู่ ผู้ใช้งานสามารถเลือกส่งออกรายงานเป็น PDF หรือ Excel เพื่อใช้งานจริงหรือส่งต่อให้หน่วยงานภายนอกได้ทันที

5.2.4 การประเมินโดยผู้ใช้งาน

จากการทดสอบโดย เจ้าหน้าที่ทะเบียน ซึ่งเป็นผู้ใช้งานจริง พบว่า

1. ผู้ใช้สามารถเข้าถึงข้อมูลที่ถูกต้องได้สะดวกขึ้นเมื่อเทียบกับระบบเดิม
2. ระบบต้นแบบช่วยลดเวลาในการตรวจสอบข้อมูลก่อนออกรายงาน
3. ผู้ใช้งานให้ข้อเสนอแนะว่าระบบควรมี Dashboard สรุปคุณภาพข้อมูล เพิ่มเติม เพื่อช่วยติดตามปัญหาที่อาจเกิดขึ้นในอนาคต

สรุปผลการทำงานของระบบต้นแบบแสดงให้เห็นว่า การทำ Data Cleansing มีผลโดยตรงต่อความถูกต้องและความน่าเชื่อถือของรายงานที่ได้ ระบบสามารถทำงานได้อย่างมีประสิทธิภาพมากขึ้นและได้รับการยอมรับจากผู้ใช้งานจริงว่าเหมาะสมต่อการนำไปใช้สนับสนุนงานทะเบียนในอนาคต

5.2.3 การเปรียบเทียบการทำงานระหว่างระบบเดิมและระบบต้นแบบ

เพื่อยืนยันประสิทธิภาพของระบบต้นแบบ ได้ทำการทดสอบกับข้อมูลจำนวน 1,119 แถว โดยเปรียบเทียบกับระบบฐานข้อมูลเดิม ผลการทดสอบสามารถสรุปได้ดังตาราง

ตาราง 19 การเปรียบเทียบผลการตรวจสอบข้อมูลระหว่างระบบเดิมและระบบต้นแบบ

รายการเปรียบเทียบ	ระบบเดิม	ระบบต้นแบบ
จำนวนแถวทั้งหมด	1,119	1,119
จำนวนแถวที่ Fail..!	301 (26.9%)	359 (32.1%)
จำนวนคอลัมน์ที่มีปัญหา	11	8
คอลัมน์ที่พบปัญหาสำคัญ	SUB_DISTRICT_ID (236), NATIONALITY_ID (80), CURR_ID (107), BIRTHDAY (58), ADMIT_YEAR (1), CITIZEN_ID (3), PREFIX_NAME_ID (1), GENDER_ID (1), STD_FNAME_EN (1), STD_LNAME_EN (1), FAC_ID (8)	STD_STATUS_ID (254), SUB_DISTRICT_ID (103), NATIONALITY_ID (68), CURR_ID (68), BIRTHDAY (60), FAC_ID (8), CITIZEN_ID (5), STD_FNAME_EN (1)
เวลาในการประมวลผล	2.4 นาที	10.34 วินาที

การวิเคราะห์ผล

1. จำนวนคอลัมน์ที่มีปัญหา ระบบต้นแบบตรวจพบปัญหาเพียง 8 คอลัมน์ เทียบกับระบบเดิมที่พบถึง 11 คอลัมน์ แสดงให้เห็นว่าการทำ Data Cleansing และการออกแบบโครงสร้างใหม่ช่วยลดจุดบกพร่องลงได้

2. จำนวนแถวที่ Fail..! ระบบต้นแบบตรวจพบแถวที่มีปัญหามากขึ้น (359 แถว เทียบกับ 301 แถวในระบบเดิม) เนื่องจากมีการตรวจสอบที่เข้มงวดขึ้น ทำให้สามารถสะท้อนปัญหาคุณภาพข้อมูลที่ระบบเดิมตรวจไม่พบมาก่อน

3. ความเร็วในการทำงาน ระบบต้นแบบใช้เวลาเพียง 10.34 วินาที ในการประมวลผลข้อมูลทั้งหมด 1,119 แถว ขณะที่ระบบเดิมใช้เวลา 2.4 นาที แสดงให้เห็นถึงประสิทธิภาพเชิงประสิทธิผล (Efficiency) ที่เพิ่มขึ้นอย่างชัดเจน

5.3 ผลการประเมินโดยผู้ใช้งาน (User Evaluation)

การประเมินระบบต้นแบบครั้งนี้ได้ดำเนินการโดยใช้ แบบประเมินความพึงพอใจผู้ใช้งานระบบต้นแบบ ซึ่งแบ่งออกเป็น 3 หมวด โดยได้ผมดังนี้

ตาราง 20 ตารางสรุปผลการประเมินโดยผู้ใช้งาน

หมวดที่ 1 คุณภาพข้อมูล (ตอบเป็น %)	ค่าเฉลี่ย(%)	SD
1. ความถูกต้องของข้อมูล (Accuracy)	100.00	0.00
2. ความครบถ้วนของข้อมูล (Completeness)	66.44	0.00
3. ความเป็นไปตามรูปแบบ (Validity/Conformance)	100.00	0.00
4. ความสอดคล้องกันของข้อมูล (Consistency)	100.00	0.00
5. ความไม่ซ้ำซ้อนของข้อมูล (Uniqueness)	100.00	0.00
6. ความทันเวลาในการอัปเดต (Timeliness)	100.00	0.00
เฉลี่ยรวม = 94.41%, SD (ของผู้ใช้) = 0.00		
หมวดที่ 2: ความสะดวกในการใช้งาน (ระดับ 1–5)	**ค่าเฉลี่ย**	**SD**
7. ความง่ายในการใช้งานโดยรวม	5.00	0.00
8. ความชัดเจนของรายงาน/ผลลัพธ์	5.00	0.00
เฉลี่ยรวม = 5.00, SD = 0.00		
หมวดที่ 3: ประโยชน์ต่อการปฏิบัติงาน (ระดับ 1–5)	**ค่าเฉลี่ย**	**SD**
9. ระบบช่วยลดเวลา/ขั้นตอนการทำงาน	5.00	0.00
10. สามารถนำผลจากระบบไปใช้ประโยชน์จริง	5.00	0.00
เฉลี่ยรวม = 5.00, SD = 0.00		

สรุปผลการประเมินโดยผู้ใช้งาน

จากการประเมินระบบต้นแบบโดยผู้ใช้งานจำนวน 4 คน พบว่า

หมวดที่ 1 คุณภาพข้อมูล ได้ค่าเฉลี่ยรวม 94.41% โดยมี SD = 0.00 สะท้อนว่าผู้ประเมินเห็นตรงกันว่าระบบสามารถแสดงข้อมูลที่ถูกต้อง ครบถ้วน และตรงตามรูปแบบได้เป็นอย่างดี แม้จะมีบางมิติ (Completeness) ที่ยังมีค่าร้อยละต่ำกว่ามิติอื่น แต่โดยรวมถือว่าอยู่ในระดับที่น่าพึงพอใจมาก

หมวดที่ 2 ความสะดวกในการใช้งาน ได้ค่าเฉลี่ยรวม 5.00 (เต็ม 5) และ SD = 0.00 แสดงให้เห็นว่าผู้ใช้งานมีความพึงพอใจมากที่สุดต่อความง่ายในการใช้งานและความชัดเจนของรายงานผลลัพธ์

หมวดที่ 3 ประโยชน์ต่อการปฏิบัติงาน ได้ค่าเฉลี่ยรวม 5.00 (เต็ม 5) และ SD = 0.00 สะท้อนว่าผู้ใช้งานเห็นตรงกันว่าระบบช่วยลดเวลาและขั้นตอนการทำงาน และสามารถนำผลไปใช้ประโยชน์จริงได้อย่างมีประสิทธิภาพ

โดยสรุป ผลการประเมินจากผู้ใช้งานสะท้อนให้เห็นว่า ระบบต้นแบบที่พัฒนาขึ้นสามารถตอบโจทย์ทั้งด้านคุณภาพข้อมูล ความสะดวกในการใช้งาน และการนำไปใช้ประโยชน์จริงได้อย่างครบถ้วน ซึ่งเป็นหลักฐานยืนยันถึงความสำเร็จของการพัฒนาระบบต้นแบบในการปรับปรุงคุณภาพข้อมูลทะเบียนนักศึกษา

บทที่ 6
สรุปผลการวิจัยและข้อเสนอแนะ

บทนี้นำเสนอการสรุปผลการวิจัยจากการดำเนินการตามขั้นตอนที่กล่าวไว้ในบทที่ 4 และผลลัพธ์ที่ปรากฏในบทที่ 5 โดยสรุปสาระสำคัญของงานวิจัยทั้งในด้านการปรับปรุงคุณภาพข้อมูล (Data Cleansing) และการพัฒนาระบบต้นแบบ (Prototype) ที่ถูกนำไปทดสอบใช้งานจริง ตลอดจนสะท้อนผลการประเมินโดยผู้ใช้งานที่เกี่ยวข้อง เพื่อนำไปสู่ข้อสรุปเชิงวิชาการและข้อเสนอแนะสำหรับการพัฒนาในอนาคต

เนื้อหาของบทนี้แบ่งออกเป็น 3 ส่วนหลัก ได้แก่ (1) การสรุปผลการวิจัยที่ได้จากการทดลองและการประเมิน (2) การเปรียบเทียบผลลัพธ์กับงานวิจัยที่เกี่ยวข้อง เพื่อชี้ให้เห็นความสอดคล้องและความแตกต่าง และ (3) การนำเสนอแนวทางการวิจัยและพัฒนาระบบในอนาคต อันจะเป็นประโยชน์ต่อการยกระดับคุณภาพข้อมูลและการจัดการระบบทะเบียนนักศึกษาในระดับสถาบันอุดมศึกษา

6.1 สรุปผลการวิจัย

จากการดำเนินการวิจัยเรื่อง “การใช้เทคนิค Data Cleansing เพื่อปรับปรุงคุณภาพข้อมูลทะเบียนนักศึกษา และประเมินผลโดยระบบต้นแบบ” สามารถสรุปผลที่สำคัญได้ดังนี้

6.1.1 ด้านการปรับปรุงคุณภาพข้อมูล (Data Cleansing)

1. การทำ Data Profiling เบื้องต้นช่วยระบุปัญหาคุณภาพข้อมูล เช่น Missing Values, Duplicate Records, Invalid Formats และ Business Rules Violations ได้อย่างชัดเจน

2. การประยุกต์ใช้วิธีการที่หลากหลาย ทั้ง Rule-Based Cleaning, Software-Based Cleaning (OpenRefine, Microsoft Excel), และ Machine Learning-Based Cleaning (เช่น การตรวจคำสะกดด้วย OCSVM + Spell Checking) ทำให้สามารถปรับปรุงข้อมูลให้ถูกต้องและครบถ้วนมากขึ้น

ผลลัพธ์แสดงให้เห็นว่าค่าที่ไม่ถูกต้อง (Invalid Values) ลดลงเหลือศูนย์ ข้อมูลที่สะกดไม่เป็นมาตรฐานถูกปรับให้อยู่ในรูปแบบเดียวกัน และความครบถ้วนของข้อมูลเพิ่มขึ้นอย่างมีนัยสำคัญ

6.1.2 ด้านการพัฒนาระบบต้นแบบ (Prototype Development)

1. ระบบต้นแบบที่สร้างขึ้นสามารถแสดงรายงานข้อมูลนักศึกษาได้อย่างถูกต้องและเป็นมาตรฐานมากกว่าระบบเดิม

2. การออกแบบโดยใช้สถาปัตยกรรม Clean Architecture และแนวคิด MVC ช่วยให้ระบบมีความยืดหยุ่น และสามารถขยายผลไปยังโมดูลอื่นได้ในอนาคต

3. Prototype รองรับฟังก์ชันการค้นหาและออกรายงาน (PDF/Excel) ที่ใช้งานง่ายและตอบสนองเร็วขึ้นเมื่อเทียบกับระบบฐานข้อมูลเดิม

6.1.3 ด้านผลการประเมินโดยผู้ใช้งาน (User Evaluation)

1. จากการประเมินโดยเจ้าหน้าที่ที่เกี่ยวข้องจำนวน 4 คน พบว่า ผู้ใช้งานเห็นว่าระบบช่วยให้เข้าถึงข้อมูลที่ถูกต้องและครบถ้วนได้สะดวกขึ้น ลดเวลาในการตรวจสอบข้อมูลก่อนออกรายงาน และสามารถนำไปใช้ประโยชน์จริงได้

2. ผู้ใช้งานเสนอแนะให้มีการพัฒนา Dashboard สำหรับสรุปคุณภาพข้อมูลเพิ่มเติม เพื่อสนับสนุนการติดตามและตรวจสอบในระยะยาว

กล่าวโดยสรุป งานวิจัยนี้พิสูจน์ให้เห็นว่า การดำเนินการ Data Cleansing อย่างเป็นระบบและการออกแบบฐานข้อมูลใหม่ที่มีมาตรฐาน สามารถยกระดับคุณภาพข้อมูลทะเบียนนักศึกษาได้จริง และการพัฒนาระบบต้นแบบช่วยยืนยันความเป็นไปได้ในการนำข้อมูลที่ Cleansing แล้วไปใช้งานจริงได้อย่างมีประสิทธิภาพ

6.2 การเปรียบเทียบกับงานวิจัยที่เกี่ยวข้อง

ผลการวิจัยในครั้งนี้มีความสอดคล้องและแตกต่างจากงานวิจัยที่เกี่ยวข้องดังนี้

1. สอดคล้องกับแนวทาง Rule-Based และ Text Clustering งานวิจัยนี้เลือกใช้ Rule-Based Cleaning และ Text Clustering เป็นแนวทางหลักในการปรับปรุงข้อมูล ซึ่งมีความสอดคล้องกับแนวคิดของ Woo et al. (2019) ที่ใช้การจัดกลุ่มข้อความและการแปลงค่าใน OpenRefine เพื่อแก้ปัญหาการสะกดผิดและค่าที่ไม่สอดคล้องกันในข้อมูลกึ่งโครงสร้าง ทั้งสองงานต่างชี้ว่าการใช้ Text Clustering สามารถช่วยลดข้อผิดพลาดเชิงการสะกดและทำให้ข้อมูลมีความสอดคล้องมากขึ้น

2. สอดคล้องบางส่วนกับแนวทาง Machine Learning-Based Cleaning งานวิจัยของ Al-Madi et al. (2023) และ Zhu et al. (2024) ได้เสนอการใช้ Machine Learning สำหรับตรวจจับข้อผิดพลาดและเติมค่าที่หายไป ซึ่งมีประสิทธิภาพสูงในข้อมูลซับซ้อน อย่างไรก็ตาม งานวิจัยนี้เลือกใช้ Machine Learning เฉพาะบางกรณี เช่น การตรวจคำสะกด (Spell Checking) และการตรวจจับค่าผิดปกติ เนื่องจากข้อมูลทะเบียนต้องการการควบคุมที่แม่นยำและอธิบายผลลัพธ์ได้

3. แตกต่างจากแนวทาง Workflow-Based และ Big Data Cleansing Guo et al. (2023) เน้น Workflow-Based Cleaning ที่ใช้ Rule-Based เป็นแกนหลัก เหมาะกับข้อมูลไม่ซับซ้อน และ Hosseinzadeh et al. (2023) มุ่งเน้นกลไก Big Data Cleansing สำหรับข้อมูลขนาดใหญ่และ real-time ซึ่งต่างจากงานวิจัยนี้ที่ไม่ได้เน้น Big Data แต่เน้นข้อมูลทะเบียนที่มีโครงสร้างชัดเจนและปริมาณอยู่ในระดับที่สามารถจัดการได้ด้วย Rule-Based และ Text Clustering เป็นหลัก

4. เปรียบเทียบกับเครื่องมือกึ่งอัตโนมัติ งานของ Goyle et al. (2024) พัฒนา DataAssist ซึ่งช่วยทำ Data Cleansing กึ่งอัตโนมัติและลดเวลาได้มากกว่า 50% ในขณะที่งานวิจัยนี้เลือกใช้เครื่องมือ OpenRefine, Microsoft Excel และ Python Script ควบคู่กับการกำหนดกฎเองเพื่อให้การ Cleansing สอดคล้องกับบริบทข้อมูลทะเบียนที่ต้องการความถูกต้องสูงสุด แม้จะใช้เวลามากกว่า แต่ได้ผลลัพธ์ที่สามารถตรวจสอบย้อนกลับได้ง่ายและเหมาะสมกับข้อจำกัดของสถาบัน

โดยสรุป งานวิจัยนี้แสดงให้เห็นว่าการผสมผสานแนวทาง Rule-Based, Text Clustering และการใช้ Machine Learning เฉพาะกรณี สามารถสร้างผลลัพธ์ที่สอดคล้องกับงานวิจัยก่อนหน้า แต่ปรับให้เหมาะสมกับลักษณะเฉพาะของข้อมูลทะเบียนนักศึกษาและข้อจำกัดของระบบงานจริง

6.3 แนวทางการวิจัยถัดไป (Future Work)

จากผลการดำเนินงานวิจัยครั้งนี้ สามารถเสนอแนวทางสำหรับการต่อยอดและพัฒนาในอนาคตได้ดังนี้

1. การพัฒนาระบบติดตามคุณภาพข้อมูลแบบ Real-time (Data Quality Monitoring) เพิ่มกลไกการตรวจสอบคุณภาพข้อมูลอย่างต่อเนื่องในขณะที่มีการบันทึกหรือปรับปรุงข้อมูล เพื่อป้องกันไม่ให้เกิดปัญหาซ้ำ เช่น Missing Values หรือ Invalid Formats โดยอาจใช้ Dashboard แสดงสถานะคุณภาพข้อมูล (Data Quality Dashboard) ให้ผู้รับผิดชอบติดตามได้ทันที

2. การขยายระบบต้นแบบไปยังโมดูลอื่นที่เกี่ยวข้อง งานวิจัยนี้เน้นที่ข้อมูลทะเบียนนักศึกษา แต่ในอนาคตควรขยายการใช้งานไปยังโมดูลอื่น เช่น ระบบการเงินการศึกษา ระบบการลงทะเบียนเรียน หรือระบบติดตามผลการเรียน เพื่อให้ครอบคลุมกระบวนการจัดการข้อมูลทั้งหมดของสถาบัน

3. การบูรณาการกับระบบเชิงวิเคราะห์ (Analytical Systems) ควรพัฒนาระบบต่อยอดให้สามารถเชื่อมต่อกับ Data Warehouse หรือ Business Intelligence (BI) Dashboard เพื่อสนับสนุนการวิเคราะห์เชิงสถิติและการตัดสินใจของผู้บริหาร โดยใช้ข้อมูลที่ผ่านการ Cleansing แล้วเป็นฐาน

4. การใช้เทคโนโลยี Machine Learning/AI ขั้นสูงมากขึ้น แม้ในงานวิจัยนี้ได้ประยุกต์ใช้ Machine Learning บางส่วน เช่น การตรวจคำสะกดและตรวจหาค่าผิดปกติ แต่ในอนาคตอาจพัฒนาโมเดลขั้นสูง เช่น Deep Learning หรือ Large Language Models (LLMs) สำหรับงาน Cleansing ที่ซับซ้อน เช่น การจัดการข้อมูลข้อความอิสระ (Free-text Fields) หรือการทำ Entity Resolution

5. การพัฒนากลไกการประเมินเชิงผู้ใช้ (User-Centered Evaluation)งานวิจัยนี้มีการประเมินโดยผู้ใช้งานจริงจำนวนหนึ่ง แต่ในอนาคตควรเพิ่มจำนวนผู้ประเมินจากหลายหน่วยงาน เพื่อ

สะท้อนความต้องการและมุมมองที่หลากหลายมากขึ้น รวมทั้งอาจใช้วิธี Usability Testing หรือการวัดผลเชิงปริมาณที่ละเอียดขึ้น

จากการดำเนินงานวิจัยทั้งหมด สามารถสรุปได้ว่า กระบวนการ Data Cleansing ที่เป็นระบบและการออกแบบฐานข้อมูลใหม่ที่ได้มาตรฐาน ช่วยยกระดับคุณภาพข้อมูลทะเบียนนักศึกษาให้ถูกต้อง ครบถ้วน และมีความน่าเชื่อถือมากขึ้น ขณะเดียวกัน การพัฒนาระบบต้นแบบยังทำให้เห็นถึงความเป็นไปได้ในการนำข้อมูลที่ผ่านการ Cleansing ไปใช้งานจริงอย่างมีประสิทธิภาพ ทั้งในด้านการแสดงผล การจัดทำรายงาน และการตอบสนองต่อความต้องการของผู้ใช้งาน

นอกจากนี้ การเปรียบเทียบผลการวิจัยกับงานที่ผ่านมาแสดงให้เห็นถึงความสอดคล้องกับแนวทางปัจจุบัน และยังชี้ให้เห็นโอกาสในการต่อยอดเพื่อพัฒนาระบบให้รองรับการตรวจสอบคุณภาพข้อมูลแบบ Real-time การขยายไปยังโมดูลอื่น และการบูรณาการกับเครื่องมือวิเคราะห์เชิงลึก งานวิจัยนี้จึงไม่เพียงแต่ช่วยแก้ปัญหาเชิงปฏิบัติในสถาบัน แต่ยังวางรากฐานสำหรับการพัฒนางานวิจัยและระบบสารสนเทศด้านการจัดการข้อมูลในอนาคต

รายการอ้างอิง

มหาวิทยาลัยศิลปากร

ประวัติผู้เขียน

ชื่อ-สกุล นายเอนก รุ่งนาไร่

วุฒิการศึกษา สำเร็จการศึกษาปริญญาวิทยาศาสตร์บัณฑิต สาขาเทคโนโลยีสารสนเทศ มหาวิทยาลัยราชภัฏนครปฐม พ.ศ.2552

Al-Madi, M. A., Abdel-Wahab, A., AlShanty, M., Bawazeer, S., & AlZahrani, M. (2023, 5-6 Feb. 2023). A Perceptual Data Cleansing Model (SDCM) for Reducing the Dirty Data. 2023 International Conference on Smart Computing and Application (ICSCA),

Batini, C., Cappiello, C., Francalanci, C., & Maurino, A. (2009). Methodologies for data quality assessment and improvement. *ACM Comput. Surv.*, 41(3), Article 16. https://doi.org/10.1145/1541880.1541883

Borkar, V., Deshmukh, K., & Sarawagi, S. (2001). Automatic segmentation of text into structured records. *SIGMOD Rec.*, 30(2), 175-186. https://doi.org/10.1145/376284.375682

DAMA International. (2017). *DAMA-DMBOK: Data Management Body of Knowledge*. Technics Publications.

Goyle, K., Xie, Q., & Goyle, V. (2024, 2024//). DataAssist: A Machine Learning Approach to Data Cleaning and Preparation. Intelligent Systems and Applications, Cham.

Guo, M., Wang, Y., Yang, Q., Li, R., Zhao, Y., Li, C., Zhu, M., Cui, Y., Jiang, X., Sheng, S., Li, Q., & Gao, R. (2023). Normal Workflow and Key Strategies for Data Cleaning Toward Real-World Data: Viewpoint. *Interact J Med Res*, 12, e44310. https://doi.org/10.2196/44310

Hosseinzadeh, M., Azhir, E., Ahmed, O. H., Ghafour, M. Y., Ahmed, S. H., Rahmani, A. M., & Vo, B. (2023). Data cleansing mechanisms and approaches for big data analytics: a systematic study. *Journal of Ambient Intelligence and Humanized Computing*, 14(1), 99-111. https://doi.org/10.1007/s12652-021-03590-2

Khamket, T., Polpinij, J., Rojarath, A., & Luaphol, B. (2025). Enhancing Sentiment Classification: A Comparative Analysis of Supervised and Unsupervised Methods for Improving Training Data Quality. *ICIC Express Letters, Part B: Applications*, 16(5), 471-479. https://doi.org/10.24507/icicelb.16.05.471

Rahm, E., & Do, H. (2000). Data Cleaning: Problems and Current Approaches. *IEEE Data Eng. Bull.*, 23, 3-13.

Ridzuan, F., & Wan Zainon, W. M. N. (2019). A Review on Data Cleansing Methods for Big Data. *Procedia Computer Science*, 161, 731-738. https://doi.org/https://doi.org/10.1016/j.procs.2019.11.177

Wang, R. Y., & Strong, D. M. (1996). Beyond Accuracy: What Data Quality Means to Data Consumers. *Journal of Management Information Systems*, 12(4), 5-33. https://doi.org/10.1080/07421222.1996.11518099

Woo, H., Kim, K., Cha, K., Lee, J.-Y., Mun, H., Cho, S. J., Chung, J. I., Pyo, J. H., Lee, K.-C., & Kang, M. (2019). Application of Efficient Data Cleaning Using Text Clustering for Semistructured Medical Reports to Large-Scale Stool Examination Reports: Methodology Study. *J Med Internet Res*, 21(1), e10013. https://doi.org/10.2196/10013

Zhu, J., Zhao, X., Sun, Y., Song, S., & Yuan, X. (2024). Relational Data Cleaning Meets Artificial Intelligence: A Survey. *Data Science and Engineering*. https://doi.org/10.1007/s41019-024-00266-7

ภาคผนวก ก

ผลการตรวจสอบข้อมูลก่อนและหลังการทำ Data Cleansing

ผู้วิจัยแสดงเฉพาะคอลัมน์ที่ทำการ Cleansing

ตาราง 21 ผลการ Cleansing ตาราง REG_STUDENT_REGISTRATION จำนวน 740,368 แถว

คอลัมน์	ก่อน Cleansing จำนวน (ร้อยละ)	หลัง Cleansing จำนวน (ร้อยละ)	**วิธี Cleansing**	**หมายเหตุ**
REG_YEAR	ค่าว่าง 1 (0.00%)	ค่าว่าง 0 (0.00%)	Manual Cleaning	
STUDENT_ID	ค่าว่าง 12 (0.00%)	ค่าว่าง 0 (0.00%)	Manual Cleaning	
SUBJECT_CODE	ค่าว่าง 3 (0.00%)	ค่าว่าง 2,184 (12.10%)	Manual Cleaning	
GRADE_DOUBLECHECK	ไม่ถูกรูปแบบ 0 (0.00%)	-	-	รูปแบบ A,B+,B,C+,C,D+,D, F,S,W,*,U,I,AU

ตาราง 22 ผลการ Cleansing ตาราง REF_MAJOR จำนวน 145 แถว

คอลัมน์	ก่อน Cleansing จำนวน (ร้อยละ)	หลัง Cleansing จำนวน (ร้อยละ)	**วิธี Cleansing**	**หมายเหตุ**
MAJOR_CODE	ค่าว่าง 0 (0.00%)	-	Manual Cleaning	
NAME_TH_LONG	ค่าว่าง 0 (0.00%) คำอาจสะกดผิด 0 (0.00%)	-	Manual Cleaning Machine Learning	
NAME_EN_LONG	ค่าว่าง 8 (5.52%) คำอาจสะกดผิด 0 (0.00%)	ค่าว่าง 8 (5.52%)	Manual Cleaning Machine Learning	ส่งส่วนงานที่เกี่ยวข้องตรวจสอบ

ตาราง 23 ผลการ Cleansing ตาราง REF_FACULTY จำนวน 28 แถว

คอลัมน์	ก่อน Cleansing จำนวน (ร้อยละ)	หลัง Cleansing จำนวน (ร้อยละ)	**วิธี Cleansing**	**หมายเหตุ**
FACULTY _CODE	ค่าว่าง 0 (0.00%)	-	Manual Cleaning	
NAME_TH_LONG	ค่าว่าง 0 (0.00%) คำอาจสะกดผิด 1 (3.57%)	คำอาจสะกดผิด 0 (0.00%)	Manual Cleaning Machine Learning	
NAME_EN_LONG	ค่าว่าง 0 (0.00%) คำอาจสะกดผิด 0 (0.00%)	-	Manual Cleaning Machine Learning	ส่งส่วนงานที่เกี่ยวข้องตรวจสอบ

ตาราง 24 ผลการ Cleansing ตาราง REF_STUDENT_STATUS จำนวน 18 แถว

คอลัมน์	ก่อน Cleansing จำนวน (ร้อยละ)	หลัง Cleansing จำนวน (ร้อยละ)	**วิธี Cleansing**	**หมายเหตุ**
STATUS_CODE	ค่าว่าง 0 (0.00%)	-	Manual Cleaning	
NAME_TH	ค่าว่าง 0 (0.00%) คำอาจสะกดผิด 0 (0.00%)	-	Manual Cleaning Machine Learning	

ตาราง 25 ผลการ Cleansing ตาราง REG_STUDENT_TYPE จำนวน 13 แถว

คอลัมน์	ก่อน Cleansing จำนวน (ร้อยละ)	หลัง Cleansing จำนวน (ร้อยละ)	**วิธี Cleansing**	**หมายเหตุ**
STUDENT_TYPE_CODE	ค่าว่าง 0 (0.00%)	-	Manual Cleaning	

NAME_TH	ค่าว่าง 0 (0.00%) คำอาจสะกดผิด 0 (0.00%)	-	Manual Cleaning Machine Learning	
LEVEL_CODE	ค่าว่าง 0 (0.00%)	-	Manual Cleaning	
LEARNING_YEAR	ค่าว่าง 0 (0.00%)		Manual Cleaning	

ตาราง 26 ผลการ Cleansing ตาราง REF_STUDENT_LEVEL จำนวน 18 แถว

คอลัมน์	ก่อน Cleansing จำนวน (ร้อยละ)	หลัง Cleansing จำนวน (ร้อยละ)	**วิธี Cleansing**	**หมายเหตุ**
LEVEL_CODE	ค่าว่าง 0 (0.00%)	-	Manual Cleaning	
NAME_TH_LONG	ค่าว่าง 0 (0.00%) คำอาจสะกดผิด 0 (0.00%)	-	Manual Cleaning Machine Learning	
NAME_EN_LONG	ค่าว่าง 13 (72.22%) คำอาจสะกดผิด 0 (0.00%)	-	Manual Cleaning	ส่งส่วนงานที่เกี่ยวข้องตรวจสอบ

ตาราง 27 ผลการ Cleansing ตาราง REG_CURRICULUM จำนวน 23 แถว

คอลัมน์	ก่อน Cleansing จำนวน (ร้อยละ)	หลัง Cleansing จำนวน (ร้อยละ)	**วิธี Cleansing**	**หมายเหตุ**
CURRICULUM_CODE	ค่าว่าง 0 (0.00%)	-	Manual Cleaning	
NAME_TH	ค่าว่าง 0 (0.00%) คำอาจสะกดผิด 1 (4.35%)	คำอาจสะกดผิด 0 (0.00%)	Manual Cleaning Machine Learning	
NAME_EN	ค่าว่าง 0 (0.00%) คำอาจสะกดผิด 0 (0.00%)	-	Manual Cleaning Machine Learning	
LEVEL_CODE	ค่าว่าง 0 (0.00%)	-		

ตาราง 28 ผลการ Cleansing ตาราง REG_CURRICULUM_VERSION จำนวน 231 แถว

คอลัมน์	ก่อน Cleansing จำนวน (ร้อยละ)	หลัง Cleansing จำนวน (ร้อยละ)	**วิธี Cleansing**	**หมายเหตุ**
CURRICULUM_VERSION	ค่าว่าง 0 (0.00%)	-	Manual Cleaning	
CURRICULUM_ID	ค่าว่าง 0 (0.00%)	-	Manual Cleaning	
CURRICULUM_CODE	ค่าว่าง 0 (0.00%)	-	Manual Cleaning	
FACULTY_CODE	ค่าว่าง 0 (0.00%)	-	Manual Cleaning	
MAJOR_CODE	ค่าว่าง 1 (0.43%)	ค่าว่าง 1 (0.43%)	Manual Cleaning	ส่งส่วนงานที่เกี่ยวข้องตรวจสอบ

ตาราง 29 ผลการ Cleansing ตาราง REF_COURSE_GROUP จำนวน 1,591 แถว

คอลัมน์	ก่อน Cleansing จำนวน (ร้อยละ)	หลัง Cleansing จำนวน (ร้อยละ)	**วิธี Cleansing**	**หมายเหตุ**
CURRICULUM_VERSION	ค่าว่าง 119 (7.48%)	ค่าว่าง 0 (0.00%)	Manual Cleaning	
GROUP_CODE	ค่าว่าง 119 (7.48%)	ค่าว่าง 0 (0.00%)	Manual Cleaning	
NAME_TH	ค่าว่าง 0 (0.00%) คำอาจสะกดผิด 7 (0.44%)	คำอาจสะกดผิด 0 (0.00%)	Manual Cleaning Machine Learning	
TOTAL_CREDIT	ค่าว่าง 0 (0.00%)	-	Manual Cleaning	

ตาราง 30 ผลการ Cleansing ตาราง REG_COURSE_IN_GROUP จำนวน 8,252 แถว

คอลัมน์	ก่อน Cleansing จำนวน (ร้อยละ)	หลัง Cleansing จำนวน (ร้อยละ)	**วิธี Cleansing**	**หมายเหตุ**
CURRICULUM_VERSION	ค่าว่าง 932 (11.29%)	ค่าว่าง 0 (0.00%)	Manual Cleaning	
GROUP_CODE	ค่าว่าง 932 (11.29%)	ค่าว่าง 0 (0.00%)	Manual Cleaning	
SUBJECT_CODE	ค่าว่าง 14 (0.17%)	ค่าว่าง 0 (0.00%)	Manual Cleaning	

ภาคผนวก ข

Source-to-Target Mapping

ตาราง 31 Source Table: REG_STUDENT

Source Table: REG_STUDENT		Target Table: reg_student		
Source Column	Source Type	Target Column	Target Type	Transformation Rule / Note
STUDENT_ID	-	student_id	character(10)	
YEAR_ENTRY	-	year_entry	character(4)	
SEMESTER_ENTRY	-	semester_entry	character(1)	
STUDENT_TYPE	-	student_type_code	character(2)	ref_student_type
TITLE_THI	-	title_code	character(3	ref_prefix
TITLE_ENG	-	-	-	
NAME_ENG	-	name_eng	character50)	
-	-	middle_name_eng	character50)	
SURNAME_ENG	-	surname_eng	character50)	
NAME_THI	-	name_tha	character50)	
-	-	middle_name_tha	character50)	
SURNAME_THI	-	surname_tha	character50)	
SEX	-	gender_code	character(1	ref_gender
BIRTHDATE	-	birthdate	date	แปลง YYYYMMDD; พ.ศ. > ค.ศ.
ADMIT_DATE	-	admit_date	date	แปลง YYYYMMDD; พ.ศ. > ค.ศ.
GRADUATION_DATE	-	graduation_date	date	แปลง YYYYMMDD; พ.ศ. > ค.ศ.
LEAVE_DATE	-	leave_date	date	แปลง YYYYMMDD; พ.ศ. > ค.ศ.
PERSONAL_ID	-	personal_id	character(20)	
HIEGHT	-	height	integer	
WEIGHT	-	weight	integer	
RACE	-	race_code	character(3)	ref_race
NATION_CODE	-	nationality_code	character(3)	ref_nationality
RELIGION_CODE	-	religion_code	character(3)	ref_religion
FACULTYZ	-	faculty_code	character(3)	ref_faculty
MAJORZ	-	major_code	character(3)	ref_major
STRUC_CURRICULUM_COD	-	curriculum_id	character(15)	
-	-	curriculum_version	character(15)	reg_curriculum_version
CREDIT_REGISTER_TOTA	-	credit_register_total	integer	
CG_P_A	-	cgpa	real	
STUDENT_STATUS_CODE	-	status_code	character(2)	ref_student_status
CONSULT_TEACHER	-	consult_teacher	character(10)	
CONFIRM_TEACHER	-	confirm_teacher	character(10)	
FORM_FEE_REGISTER_ST	-	form_fee_register	text	
PASSWORDZ	-	password	character(255)	

COMMENTZ	-	comment	text	
IN_TAKE	-	-	-	
GROUPZ	-	-	-	
FIRST_DIGIT_NAME	-	-	-	
FULL_NAME_THAI	-	-	-	
FULL_NAME_ENG	-	-	-	
GRADUATION_SEMESTER	-	-	-	
LEAVE_SEMESTER	-	-	-	
AGE	-	-	-	
PACE_OF_BIRTH	-	-	-	
PROVINCE_OF_BIRTH	-	-	-	
COUNTRY_OF_BIRTH	-	-	-	
DEPARTZ	-	-	-	
MINOR	-	-	-	
STUDENT_PLAN_CODE	-	-	-	
DEGREE_CODE	-	-	-	
GRADE_AVERAGE_LAST_S	-	-	-	
GRADE_POINT_PASSED_T	-	-	-	
CREDIT_PASSED_TOTAL	-	-	-	
HONOR_SEQ	-	-	-	
VUT_NOT_USE	-	-	-	
STUDENT_MAJOR_TYPE	-	-	-	
LEARNING_YEAR	-	-	-	
STUDENT_YEAR	-	-	-	
REGISTER_AMOUNT	-	-	-	
NOT_PAY_AMOUNT	-	-	-	
PAYMENT_STATUS_FLAG	-	-	-	
CAPITAL_CODE	-	-	-	
BANK_CODE	-	-	-	
BANK_ACOUNT_NO	-	-	-	
LOAN_STATUS_FLAG	-	-	-	
FORM_FEE_OTHER_STUDE	-	-	-	
FORM_SUBJECT_STUDENT	-	-	-	
TRANSFER_FROM	-	-	-	
COUNTRY	-	-	-	
LINK_CODE	-	-	-	
CLASSZ	-	grade_level	integer	
FILENAME	-	-	-	
CENTER	-	-	-	
RESERVEZ	-	-	-	

ตาราง 32 Source Table: REG_REGISTER_SUBJECT

Source Table: REG_REGISTER_SUBJECT		Target Table: reg_student		
Source Column	Source Type	Target Column	Target Type	Transformation Rule / Note
REG_SUBJECT_CODE	-	subject_code	character(10)	
REG_SUBJECT_TYPE	-	subject_type	character(3)	
CREDIT_REGISTER	-	credit_reg	numeric(4,1)	
CREDIT_REGISTER_OLD	-	credit_reg_old	numeric(4,1)	
CREDIT_EARN	-	credit_earned	numeric(4,1)	
SUBJECT_GROUP_THEORY	-	group_theory	character(2)	
SUBJECT_GROUP_LAB	-	group_lab	character(2)	
GRADEZ	-	grade	character(2)	
GRADE_DOUBLECHECK	-	grade_doublecheck	character(2)	
CONFIRMZ	-	grade_confirmed	character(2)	
REGISTER_COMMENT	-	comment_other	character(255)	
REGISTER_FEE_AMT	-	fee_register	numeric(10,2)	
LAB_FEE_AMT	-	fee_lab	numeric(10,2)	
INTENSIVE_FEE_AMOUNT	-	fee_intensive	numeric(10,2)	
MAINT_FEE_AMT	-	fee_maintenance	numeric(10,2)	
REGISTER_STATUS	-	register_status	character(1)	
TRANSFER_GRADE_FLG	-	flg_transfer_grade	character(1)	
REGISTER_ALLOWED_FLG	-	flg_allow_register	character(1)	
STUDY_OVELAPTIME_FLG	-	flg_overlap_study	character(1)	
MID_OVLT_FLG	-	flg_overlap_mid	character(1)	
FINAL_OVLT_FLG	-	flg_overlap_final	character(1)	
CANCEL_FLAG	-	flg_cancel	character(1)	
CHECKCO_STUDYFLG	-	flg_check_overlap	character(1)	
APPROVED_BY	-	approved_by	character(10)	
GRADE_ENTRYBY	-	grade_entry_by	character(10)	
GRADE_ENTRYDATE	-	grade_entry_date	date	
GRADE_ENTRYTIME	-	grade_entry_time	time	
CONSULT_TEACHERID	-	consult_teacher_id	character(10)	
COUNT_GRADE	-	count_grade	smallint	
COUNT_CREDIT	-	count_credit	numeric(4,1)	
PART_OF_CREDIT	-	part_credit	numeric(4,1)	
CONTACT_SEQREGSUB_AA	-	contact_seq_a	smallint	
CONTACT_SEQREQSUB_BB	-	contact_seq_b	smallint	
TB1027_CL040	-	-	-	
COMMENTZ	-	-	-	
FLAG_PAID	-	flag_paid	character(1)	
DATE_REGIS	-	date_register	date	

ตาราง 33 Source Table: REG_MAJOR

Source Table: REG_MAJOR		Target Table: ref_major		
Source Column	Source Type	Target Column	Target Type	Transformation Rule / Note
MAJOR_CODE	-	major_code	character(3)	
MAJOR_NAMETHAI	-	name_th_long	character(255)	
MAJOR_NAMEENG	-	name_en_long	character(255)	
FLAG_GROUP	-	-	-	
FLAG_SHOW	-	-	-	
-	-	name_th_short	character(50)	
-	-	name_en_short	character(50)	
-	-	faculty_code	character(3)	
-	-	note	text	

ตาราง 34 Source Table: REF_FACULTY

Source Table: REF_FACULTY		Target Table: ref_faculty		
Source Column	Source Type	Target Column	Target Type	Transformation Rule / Note
FACULTY_CODE	-	faculty_code	character(3)	
FACULTY_NAMEFULLTHI	-	name_th_long	character(255)	
FACULTY_NAMEFULLENG	-	name_en_long	character(255)	
FACULTY_NAMESHORTT	-	name_th_short	character(50)	
FACULTY_NAMESHORTE	-	name_en_short	character(50)	

ตาราง 35 Source Table: REG_STUDENT_STATUS

Source Table: REG_STUDENT_STATUS		Target Table: ref_student_status		
Source Column	Source Type	Target Column	Target Type	Transformation Rule / Note
STATUS_CODE	-	status_code	character(2)	
STATUS_SEQ	-	seq_no	smallint	
STUDENT_STATUS_NAME	-	name_th	character(50)	
-	-	name_en	character(50)	
NEXT_SEMESTER_REGIST	-	-	-	
PRE_EXPIRE_FLAG	-	-	-	
EXPIRE_FLAG	-	-	-	
GRADUATION_FLAG	-	-	-	
NOT_RIGHT_STUDY	-	-	-	
NOT_RIGHT_EXAM	-	-	-	
NOT_IN_INSTITUTE	-	-	-	
LEAVE	-	-	-	
CAL_FROM_GPA	-	-	-	
RESERVEZ	-	-	-	
COMMENTZ	-	-	-	

ตาราง 36 Source Table: REG_STUDENT_TYPE

Source Table: REG_STUDENT_TYPE		Target Table: ref_student_type		
Source Column	Source Type	Target Column	Target Type	Transformation Rule / Note
STUDENT_TYPE_CODE	-	student_type_code	Character(4)	
STUDENT_TYPE_NAME	-	name_th	Character(120)	
-	-	name_en	Character(120)	
STUDENT_VUT	-	level_code	Character(2)	
STUDENT_MAJOR_TYPE	-	major_type	smallint	
LEARNING_YEAR	-	learning_year	smallint	
TRANSFER_STUDENT_FLA	-	transfer_flag	smallint	
DISTINCION_FLAG	-	distinction_flag	smallint	
-	-	active_flag	smallint	

ตาราง 37 Source Table: HRS_PNDEGREE

Source Table: HRS_PNDEGREE		Target Table: ref_student_level		
Source Column	Source Type	Target Column	Target Type	Transformation Rule / Note
SEQUEN	-	level_code	Character(2)	
DEGREE_NAME_THI_L		name_th_long	character(100)	
DEGREE_NAME_ENG_S	-	name_en_short	character(50)	
DEGREE_NAME_ENG_L		name_en_long	character(100)	
-	-	name_en_short	character(50)	
-	-	default_yr	smallint	

ตาราง 38 Source Table: REG_PLACE_CONTACT

Source Table: REG_PLACE_CONTACT		Target Table: ref_student_level		
Source Column	Source Type	Target Column	Target Type	Transformation Rule / Note
STUDENT_ID	-	STUDENT_ID	character(10)	
-	-	ADDRESS_TYPE_CODE	character(1)	
TB0965_CL002	-	house_no	character(255)	
TB0965_CL003	-	village	character(100)	
TB0965_CL004	-	alley	character(100)	
TB0965_CL005	-	road	character(100)	
TB0965_CL006	-	subdistrict_code	character(6)	
TB0965_CL007	-	-	-	
TB0965_CL008	-	-	-	
TB0965_CL009	-	-	-	
TB0965_CL010	-	-	-	
TB0965_CL011	-	-	-	
TB0965_CL012	-	email	character(255)	
TB0965_CL013	-	phone	character(50)	
TB0965_CL014	-	-	-	

TB0965_CL015	-	-	-	
TB0965_CL016	-	-	-	
TB0965_CL017	-	-	-	
TB0965_CL018	-	-	-	
TB0965_CL019	-	-	-	
TB0965_CL020	-	-	-	
TB0965_CL021	-	-	-	
TB0965_CL022	-	-	-	
TB0965_CL023	-	-	-	
TB0965_CL024	-	-	-	
TB0965_CL025	-	-	-	
TB0965_CL026	-	-	-	
TB0965_CL027	-	-	-	
TB0965_CL028	-	-	-	
TB0965_CL029	-	-	-	
TB0965_CL030	-	-	-	
TB0965_CL031	-	-	-	
TB0965_CL032	-	-	-	
TB0965_CL033	-	-	-	
TB0965_CL034	-	-	-	
TB0965_CL035	-	-	-	
TB0965_CL036	-	-	-	
TB0965_CL037	-	-	-	
TB0965_CL038	-	-	-	
TB0965_CL039	-	-	-	
Lastupdate	-	last_update	timestamp	

ตาราง 39 Source Table: ThepExcel-Thailand-Tambon

Source Table: ThepExcel-Thailand-Tambon		Target Table: ref_subdistrict		
Source Column	Source Type	Target Column	Target Type	Transformation Rule / Note
TambonID	-	subdistrict_code	character(6)	
TambonThai	-	subdistrict_name_th	character(100)	
TambonThaiShort	-	subdistrict_name_th_short	character(100)	
TambonEng	-	subdistrict_name_en	character(100)	
TambonEngShort	-	subdistrict_name_en_short	character(100)	
DistrictID	-	district_code	character(4)	
PostCodeMain	-	postcode	character(5)	

ตาราง 40 Source Table: REG_SUBJECT

Source Table: REG_SUBJECT		Target Table: ref_subject		
Source Column	Source Type	Target Column	Target Type	Transformation Rule / Note
-	-	subject_id	bigint	
SUBJECT_CODE	-	subject_code	character(10)	
OLD_CODE	-	old_code	character(10)	
SUBJECT_CODE_SORT	-	subject_code_sort	character(10)	
SUBJECT_TYPE	-	subject_type	character(3)	
VUT_CODE	-	level_code	numeric(5,0)	
SUBJECT_AUDIT_FLAG	-	subject_audit_flag	character(1)	
SUBJECT_CLASS	-	subject_class	character(2)	
NAME_ENG_LINE1	-	-	-	
NAME_ENG_LINE2	-	-	-	
NAME_ENG_LINE	-	name_eng	character(255)	
NAME_THAI_LINE1	-	-	-	
NAME_THAI_LINE2	-	-	-	
NAME_THAI_LINE	-	name_thai	character(255)	
CREDIT_SUBJECT	-	credit_subject	smallint	
CREDIT_THEORY	-	credit_theory	numeric(3,1)	
CREDIT_LAB	-	credit_lab	numeric(3,1)	
CREDIT_TRAINNING	-	-	-	
HOUR_TRAINNING	-	hour_trainning	numeric(4,1)	
HOUR_ALL_TRAINNING	-	hour_all_trainning	numeric(5,1)	
CONSTRUC_AMOUNT	-	-	-	
TB0931_CL019	-	-	-	
MAINTANANCE_AMOUNT	-	-	-	
CHECK_STATUS_FSUBJ	-	-	-	
COUNT_CREDIT		count_credit	smallint	
COUNT_GRADE		count_grade	smallint	
SUB_GROUP_FLAG		sub_group_flag	character(1)	
INTENSIVE_FLAG		intensive_flag	character(1)	
COMMENTZ		comment	character(255)	
CODE_TUITION		code_tuition	character(10)	
TUITION_AMOUNT	-	-	-	
CODE_LAB	-	code_lab	character(10)	
LAB_AMOUNT	-	-	-	
CODE03	-	-	-	
AMOUNT_03	-	-	-	
CODE04		code04	character(10)	
AMOUNT_04	-	-	-	
CODE05	-	-	-	
AMOUNT_05	-	-	-	

CODE06	-	-	-	
AMOUNT_06	-	-	-	
CODE07	-	-	-	
AMOUNT_07	-	-	-	
CODE08	-	-	-	
AMOUNT_08	-	-	-	
CODE09	-	-	-	
AMOUNT_09	-	-	-	
CODE10	-	-	-	
AMOUNT_10	-	-	-	
INTENSIVE_AMOUNT	-	-	-	
FLAG_PART_CREDITS		flag_part_credits	smallint	
TYPE_ACCESS		type_access	smallint	
CANCEL		cancel	smallint	
CO_LAB		co_lab	smallint	
FACULTYZ		faculty_code	character(3)	
MAJORZ		major_code	character(3)	
TRAIN_YOURSELF		train_yourself	numeric	

ตาราง 41 Source Table: REG_LAKSUD

Source Table: REG_LAKSUD		Target Table: reg_curriculum		
Source Column	Source Type	Target Column	Target Type	Transformation Rule / Note
CURRICULUM_CODE		curriculum_code	character(6)	
CURRICULUM_NAME_TH		name_th	character(200)	
CURRICULUM_NAME_EN		name_en	character(200)	
TYPE_CURRICULUM		curriculum_type	character(1)	
VUT_CODE		level_code	character(2)	

ตาราง 42 Source Table: REG_STRUC_LAKSUD

Source Table: REG_STRUC_LAKSUD		Target Table: reg_curriculum_version		
Source Column	Source Type	Target Column	Target Type	Transformation Rule / Note
-	-	curriculum_version	character(15)	
CODE_STUC_CURRICULUM	-	curriculum_id	character(15)	
CURRICULUM_CODE	-	curriculum_code	character(6)	
-	-	version_no	smallint	
FACULTY_CODE	-	faculty_code	character(3)	
MAJOR_CODE	-	major_code	character(4)	
DEPARTZ	-	-	-	
LEARNING_YEAR	-	learning_year	smallint	
TOTAL_CREDIT	-	total_credit	smallint	
VUD_CODE_NOT_USE	-	-	-	
DEGREE_CODE	-	-	-	

ESTIMATE_CREDIT_GRAD	-	est_credit_grad	smallint	
SHORT_NAME_NORMAL	-	-	-	
SHORT_NAME_ABNORMAL	-	-	-	
RUNNING_NORMAL	-	-	-	
RUNNING_ABNORMAL	-	-	-	
ESTIMATE_COST	-	-	-	
TYPE_RUN_APPLICATION	-	-	-	
CURRICULUM_OF_ACCOUN	-	-	-	
YEAR_BEGIN	-	year_begin	smallint	
YEAR_END	-	year_end	smallint	
CLOSEZ	-	close_flag	smallint	
SUB_TYPE	-	sub_type	smallint	
-	-	created_at	timestamp	
-	-	updated_at	timestamp	

ตาราง 43 Source Table: REG_GROUP_IN_STRUC

Source Table: REG_GROUP_IN_STRUC		Target Table: ref_course_group		
Source Column	Source Type	Target Column	Target Type	Transformation Rule / Note
-	-	curriculum_version	character(15)	
STRUC_CURRICULUMCODE	-	-	-	
SUB_TYPE	-	-	-	
GROUPZ	-	group_code	character(4)	
-	-	parent_group_code	character(4)	
GROUP_NAME	-	name_th	character(150)	
-	-	name_en	character(150)	
PLANZ	-	-	-	
CREDIT_TYPE	-	credit_type	smallint	
TOTAL_CREDIT	-	total_credit	smallint	
TOTAL_TRANING_HOUR	-	training_hour	smallint	
-	-	created_at	timestamp	
-	-	updated_at	timestamp	
ESTIMATE_CURRENCY	-	-	-	

ตาราง 44 Source Table: REG_SUBJECT_IN_GROUP

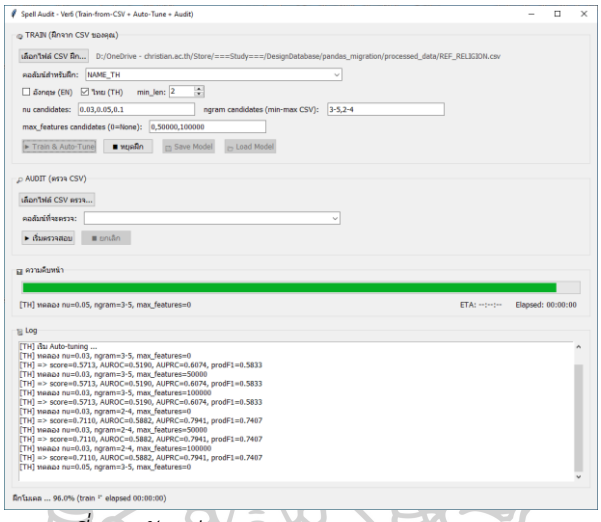
Source Table: REG_SUBJECT_IN_GROUP		Target Table: reg_course_in_group		
Source Column	Source Type	Target Column	Target Type	Transformation Rule / Note
-	-	curriculum_version	character(15)	
STRUC_CURRICULUMCODE	-	-	-	
SUB_TYPE	-	-	-	
STRUC_CURRGROUP		group_code	character(4)	
PLANZ	-	-	-	
SUBJECT_CODE	-	subject_code	character(10)	

SUBJECT_TYPE	-	**subject_type**	character(3)	
-	-	created_at	timestamp	
-	-	**updated_at**	timestamp	

ภาคผนวก ค

คู่มือการใช้งาน GUI ที่ใช้ในการ Cleansing

1. คู่มือใช้งานเครื่องมือ Spell Audit (Thai + English)



ภาพที่ 25 ตัวอย่าง Spell Audit (Thai + English)

จุดประสงค์ ตรวจจับสตริง ผิดปกติ/สะกดผิด (ทั้งไทยและอังกฤษ) เพื่อใช้ในขั้นตอน Data Cleansing ก่อนนำเข้าฐานข้อมูล โดยเฉพาะคอลัมน์ประเภทชื่อเฉพาะ เช่น ชื่อหลักสูตร/สาขา/หน่วยงาน ฯลฯ

ภาพรวมการทำงาน (Method)

1. **แนวคิดหลัก:** ตรวจจับคำ/สตริงที่ “ผิดปกติ” เมื่อเทียบกับรูปแบบภาษาในชุดข้อมูลอ้างอิง (เรียนรู้จาก CSV ฝึก หรือโหลดโมเดลเดิม)
2. **เวคเตอร์ไรซ์** แปลงสตริงเป็นเวคเตอร์ด้วย TF-IDF character n-grams
3. **โมเดล One-Class SVM (OCSVM)** แบบกึ่งกำกับ (unsupervised anomaly detection) พารามิเตอร์สำคัญ nu ≈ สัดส่วนสูงสุดของ outliers ที่คาดว่าจะมีในข้อมูลฝึก
4. **ตัวกรองความยาว** min_len กรองสตริงที่สั้นเกินไป (เช่น 1 ตัวอักษร) ซึ่งมักเป็นสัญลักษณ์หรือรหัสสั้น ๆ
5. **ผลลัพธ์** ได้ชุดรายการที่ “มีแนวโน้มผิดปกติ” เพื่อนำไปตรวจทาน/แก้ไขต่อด้วยกฎเชิงธุรกิจหรือเครื่องมืออื่น

ความต้องการระบบ (Environment)

1. **Pandas**: จัดการตารางข้อมูล/อ่าน-เขียน CSV, คัดกรองคอลัมน์, ทำความสะอาดเบื้องต้น

2. **Numpy**: โครงสร้างอาร์เรย์/เวกเตอร์เชิงตัวเลขที่ sklearn ใช้ต่อ, คำนวณเชิงเส้นพื้นฐาน

3. **scikit-learn**: เครื่องมือ ML หลัก: TfidfVectorizer (char n-grams), OneClassSVM, Pipeline/Model utils

4. **joblib**: บันทึก/โหลดโมเดลและเวคเตอร์ไรซ์เป็นไฟล์ .pkl เพื่อใช้งานซ้ำ (Save/Load Model)

ขั้นตอนการใช้งาน (Workflow)

โหมด A : TRAIN (ฝึกจาก CSV ที่ต้องการ)

1. กด “เลือกไฟล์ CSV ฝึก...” ⟶ เลือก คอลัมน์สำหรับฝึก
2. เลือกภาษา: EN, TH และตั้ง min_len (ความยาว token ขั้นต่ำ แนะนำ 2)
3. ตั้ง ชุดค่าที่จะ Auto-tune
   - nu candidates: เช่น 0.03,0.05,0.1
   - ngram candidates (min-max): เช่น 3-5,2-4
   - max_features candidates (0=None): เช่น 0,50000,100000
4. Train & Auto-Tune

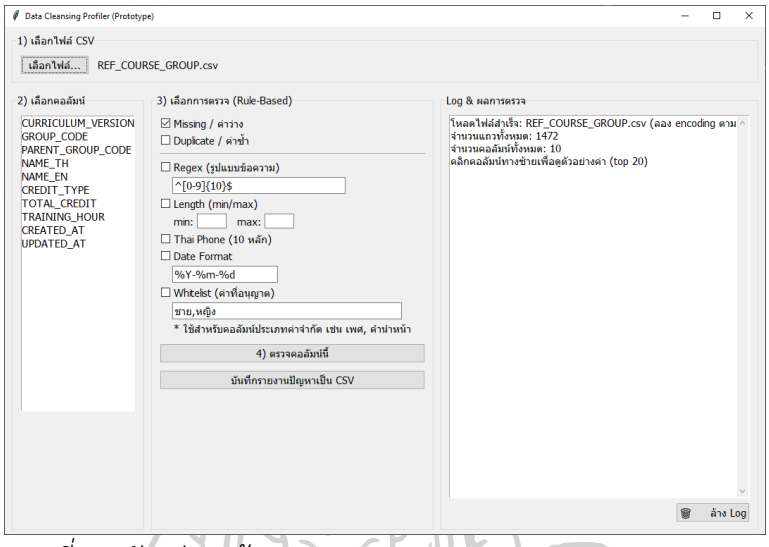
   - ระหว่างฝึกจะแสดง เปอร์เซ็นต์, สถานะ (ป้องกันเข้าใจว่าโปรแกรมค้าง)
5. สามารถกด หยุดฝึก ได้ทุกเมื่อ
6. เสร็จแล้วกด Save Model (.pkl) หรือ Load Model เพื่อเรียกใช้ภายหลัง

โหมด B: AUDIT (ตรวจ CSV)

ต้องมีโมเดลอย่างน้อย 1 ภาษา (จาก Train หรือ Load) ก่อน

1. กด “เลือกไฟล์ CSV ตรวจ...” ⟶ เลือก คอลัมน์ที่จะตรวจ
2. กด เริ่มตรวจสอบ (มี เปอร์เซ็นต์ + ETA + Log)
3. เสร็จแล้วโปรแกรมจะบันทึก spellcheck_report.csv ในโฟลเดอร์เดียวกับไฟล์อินพุต

2. คู่มือใช้งานเครื่องมือ Data Cleansing Profiler (Rule-Based)



ภาพที่ 26 ตัวอย่างหน้าจอ Data Cleansing Profiler (Rule-Based)

จุดประสงค์ เครื่องมือนี้ใช้ “ตรวจสอบเชิงกฎ (Rule-Based)” สำหรับคอลัมน์ที่เลือกในไฟล์ CSV เช่น ค่าว่าง ค่าซ้ำ รูปแบบรหัส ความยาว วันเวลา เบอร์โทร และชุดค่าอนุญาต (Whitelist) เพื่อจัดทำรายงานปัญหาก่อนนำไป Cleansing/แก้ไข

ภาพรวมการทำงาน (Method)

1. อ่าน CSV > เลือกคอลัมน์ > ตั้งค่ากฎตรวจ > กดตรวจ > แสดงผลใน Log และบันทึก “รายงานปัญหา” เป็น CSV
2. รองรับกฎตรวจหลัก
   - **Missing ค่าว่าง**
   - **Duplicate ค่าซ้ำ** (ภายในคอลัมน์ที่เลือก)
   - **Regex** (ตรวจรูปแบบด้วยนิพจน์ประจำ)
   - **Length** (min/max) ความยาวสตริง
   - **Thai Phone** (10 หลัก) ตรวจหมายเลขไทย 10 หลัก
   - **Date Format** ตรวจรูปแบบวันที่ตาม %Y-%m-%d ฯลฯ
   - **Whitelist** กำหนด “ค่าที่อนุญาต” แบบคั่นด้วยจุลภาค

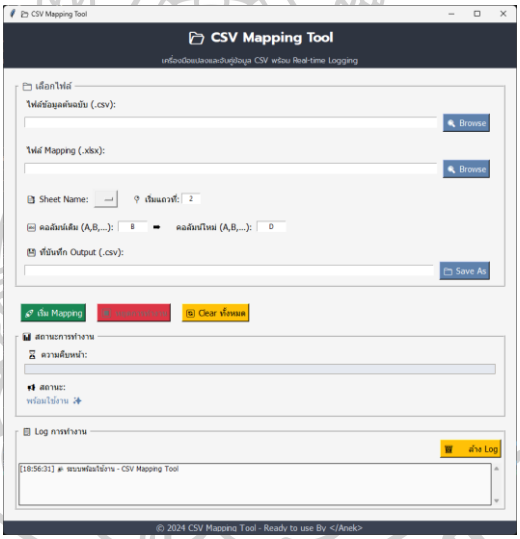
ความต้องการระบบ (Environment)

1. **Pandas**: จัดการตารางข้อมูล/อ่าน-เขียน CSV, คัดกรองคอลัมน์
2. **Numpy:** ช่วยงานเชิงอาร์เรย์ (ขึ้นกับเวอร์ชันสคริปต์)

ขั้นตอนการใช้งาน (Workflow)

1. เลือกไฟล์ CSV
2. เลือกคอลัมน์ ทางซ้าย (คลิก คอลัมน์ต่อการตรวจครั้งหนึ่ง)
3. ตั้งค่ากฎตรวจ ในกล่อง “เลือกการตรวจ (Rule-Based)” ติ๊กเฉพาะกฎที่ต้องการ แล้วกรอกรายละเอียด (เช่น regex, ช่วงความยาว, รูปแบบวันที่, รายการ whitelist)
4. กด “ตรวจคอลัมน์นี้” ผลจะขึ้นใน Log ด้านขวา (แสดงตัวอย่าง Top-20)
5. ถ้าต้องการเก็บผลทั้งหมด กด “บันทึกรายงานปัญหาเป็น CSV”

3. คู่มือใช้งานเครื่องมือ CSV Mapping Tool



ภาพที่ 27 ตัวอย่างหน้าจอ CSV Mapping Tool

	A	B	C	D	E	F
1		REG_STUDENT_STATUS (เดิม)		REF_STUDENT_STATUS (ใหม่)		
2	No.	Columns	datatype	Columns	datatype	Transformation Rule / Note
3	1	STATUS_CODE	TBD	STATUS_CODE	CHAR(2)	คัดลอกตรง (‘N’, ‘T’ ...)
4	2	STATUS_SEQ	TBD	SEQ_NO	SMALLINT UNSIGNED	trim ตัวเลข
5	3	STUDENT_STATUS_NAME	TBD	NAME_TH	VARCHAR(50)	คัดลอก (ปกติ, พ้นสภาพ, ...)
6	4	-	-	NAME_EN	VARCHAR(50)	เติมค่าแปล EN (เช่น “Active”, “Dismissed”) หรือใส่ NULL

ภาพที่ 28 ตัวอย่างไฟล์ Mapping (.xlsx)

จุดประสงค์ เครื่องมือนี้ใช้ “แปลง/จัดเรียง/เปลี่ยนชื่อคอลัมน์” ของไฟล์ CSV ตามกติกาที่กำหนดไว้ในไฟล์ Mapping (.xlsx) พร้อมบันทึกผลเป็น CSV ใหม่ และมี Real-time Logging สำหรับตรวจสอบความคืบหน้า/ข้อผิดพลาด

ภาพรวมการทำงาน (Method)

1. ผู้ใช้เลือก ไฟล์ข้อมูลต้นฉบับ (.csv) และ ไฟล์ Mapping (.xlsx)
2. ระบุ Sheet Name (ถ้ามีหลายชีต) และ เริ่มแถวที่ (ปกติ 2 = ข้าม header ของไฟล์ mapping)
3. ตั้งค่า คู่คอลัมน์ เสริมแบบเร็ว (เช่น B > D) ได้จากช่อง “คอลัมน์เดิม > คอลัมน์ใหม่”
4. ระบบอ่าน mapping แล้วทำการ เปลี่ยนชื่อ / จัดเรียง / คัดเลือกคอลัมน์ตามลำดับ
5. บันทึกผลเป็น ไฟล์ CSV ใหม่ ตาม path ที่กำหนด พร้อม Log รายละเอียด

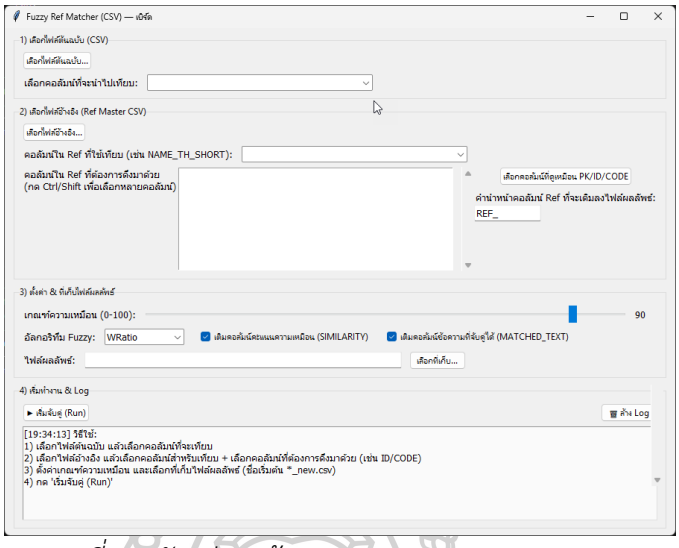
ความต้องการระบบ (Environment)

1. **Pandas**: จัดการตารางข้อมูล/อ่าน-เขียน CSV, คัดกรองคอลัมน์
2. **Openpyxl**: อ่านไฟล์ Excel (.xlsx) ที่เก็บกติกา mapping

ขั้นตอนการใช้งาน (Workflow)

1. เลือกไฟล์ข้อมูลต้นฉบับ (.csv)
2. เลือกไฟล์ Mapping (.xlsx)
3. เลือก Sheet Name
4. กำหนด เริ่มแถวที่ (ปกติเป็น 2)
5. กรอกคู่คอลัมน์ เช่น คอลัมน์เดิม (B) > คอลัมน์ใหม่ (D)
6. เลือก ที่บันทึก Output (.csv) ด้วยปุ่ม Save As
7. กด เริ่ม Mapping
8. สถานะจะขึ้นว่า “พร้อมใช้งาน/กำลังทำงาน” และมี Log รายงานความคืบหน้า
9. ปุ่ม หยุดการทำงาน เมื่อต้องการหยุดหรือยกเลิก

4. คู่มือใช้งานเครื่องมือ Fuzzy Ref Matcher (CSV)



ภาพที่ 29 ตัวอย่างหน้าจอ Fuzzy Ref Matcher (CSV)

จุดประสงค์ จับคู่ข้อความในคอลัมน์ของไฟล์งาน (CSV ต้นฉบับ) กับ “ตารางอ้างอิง (Ref Master CSV)” โดยใช้ Fuzzy String Matching เพื่อดึงค่า มาตรฐาน (เช่น CODE/ชื่อสะอาด) กลับมาเติมในไฟล์งาน เหมาะกับงานแมตช์ชื่อสาขา/หลักสูตร/เขต/ตำบล ฯลฯ

ภาพรวมการทำงาน (Method)

1. อ่าน ไฟล์ต้นฉบับ และ ไฟล์อ้างอิง
2. ใช้ อัลกอริทึม Fuzzy (ค่าเริ่มต้น WRatio) คำนวณ คะแนนความเหมือน (0–100) ระหว่างค่าที่จะเทียบกับคอลัมน์อ้างอิง
3. เลือก คำตอบที่คล้ายที่สุด (Best Match) ถ้าคะแนน ≥ เกณฑ์ (Threshold) ที่กำหนด
4. นำคอลัมน์ที่เลือกจาก Ref (เช่น ID, CODE, NAME_EN) มาเพิ่มคอลลัมน์ท้ายผล โดยใส่คำนำหน้า เช่น REF_
5. บันทึกเป็นไฟล์ใหม่ (โดยทั่วไป “ชื่อไฟล์เดิม + _new.csv”) พร้อมคอลัมน์ช่วยคือ SIMILARITY และ MATCHED_TEXT (ถ้าเลือกให้บันทึก)

ความต้องการระบบ (Environment)

1. **Pandas**: จัดการตารางข้อมูล/อ่าน-เขียน CSV, คัดกรองคอลัมน์
2. **Rapidfuzz:** คำนวณความเหมือน (WRatio/Token/Partial)

ขั้นตอนการใช้งาน (Workflow)

1. เลือกไฟล์ต้นฉบับ (CSV)

- กด เลือกไฟล์ต้นฉบับ... แล้วเลือก CSV งานจริง

- เลือก คอลัมน์ที่ต้องนำไปเทียบ จากดรอปดาวน์ (เช่น NAME_TH_SHORT)

2. เลือกไฟล์อ้างอิง (Ref Master CSV)

- กด เลือกไฟล์อ้างอิง... แล้วเลือก CSV มาตรฐาน (ข้อมูลสะอาด/มี CODE/ID)
- ช่อง คอลัมน์ใน Ref ที่ใช้เทียบ ระบุคอลัมน์ใน Ref ที่เป็น “ชื่อมาตรฐาน” สำหรับจับคู่ (เช่น NAME_TH_SHORT)
- กล่องรายการ คอลัมน์ใน Ref ที่ต้องการดึงมาด้วย เลือกหนึ่งถึงหลายคอลัมน์ (กด Ctrl/Shift เลือกหลายรายการ) เช่น ID, CODE, NAME_EN
- ช่อง คำนำหน้าคอลัมน์ Ref ที่จะเติมลงไฟล์ผลลัพธ์: แนะนำ REF_ (ผลลัพธ์จะเป็น REF_CODE, REF_ID, ...)

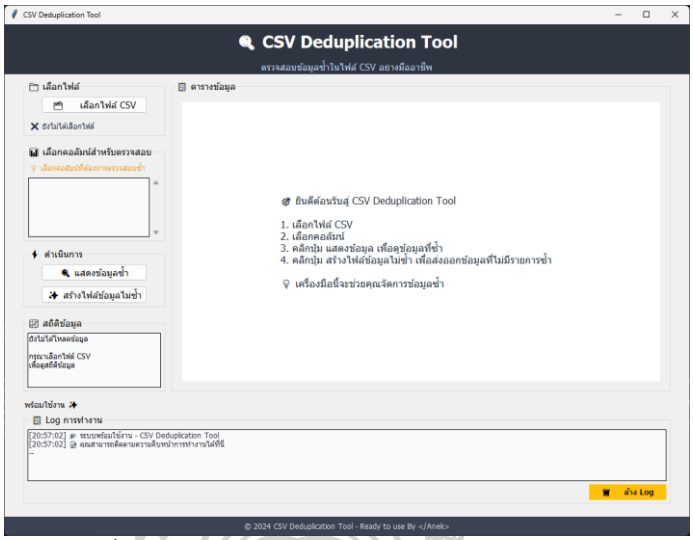
3. ตั้งค่า & ที่เก็บไฟล์ผลลัพธ์

- เกณฑ์ความเหมือน (0–100) ค่าแนะนำ ≥ 90 = แมตช์ความเชื่อมั่นสูง (อัตโนมัติได้)
- อัลกอริทึม Fuzzy ค่าเริ่มต้น WRatio (ปรับได้ตามลักษณะข้อมูล)
- ติ๊ก บันทึกคะแนนความเหมือน (SIMILARITY) และ บันทึกข้อความที่จับคู่ได้ (MATCHED_TEXT) ตามต้องการ
- เลือกที่จัดเก็บ ไฟล์ผลลัพธ์ ด้วยปุ่ม เลือกที่เก็บ...

4. เริ่มงาน & Log

- กด เริ่มรัน (Run) เพื่อเริ่มแมตช์ ระบบจะอัปเดต คู่มือใช้งาน แบบย่อในกล่อง Log และบันทึกความคืบหน้า
- ปุ่ม ล้าง Log เคลียร์หน้าต่างข้อความ

5 คู่มือใช้งานเครื่องมือ CSV Deduplication Tool



ภาพที่ 30 ตัวอย่างหน้าจอ CSV Deduplication Tool

จุดประสงค์ ตรวจหาข้อมูลซ้ำในไฟล์ CSV ตามคอลัมน์ที่กำหนด และสร้างไฟล์ **ข้อมูลไม่ซ้ำ** (ลบแถวซ้ำออก) เพื่อเตรียมข้อมูลก่อนนำเข้า/ย้ายระบบ

ภาพรวมการทำงาน (Method)

1. อ่านไฟล์ CSV > ผู้ใช้เลือก คอลัมน์สำหรับตรวจซ้ำ (เลือกได้หลายคอลัมน์)
2. ระบบหากลุ่มที่มีค่า เหมือนกันทุกคอลัมน์ที่เลือก = “ซ้ำ”
3. แสดงตัวอย่างรายการซ้ำในตาราง พร้อม สถิติข้อมูล (จำนวนแถวทั้งหมด, จำนวนแถวซ้ำ, กลุ่มซ้ำ)
4. ผู้ใช้กด สร้างไฟล์ข้อมูลไม่ซ้ำ เพื่อบันทึก CSV

ความต้องการระบบ (Environment)

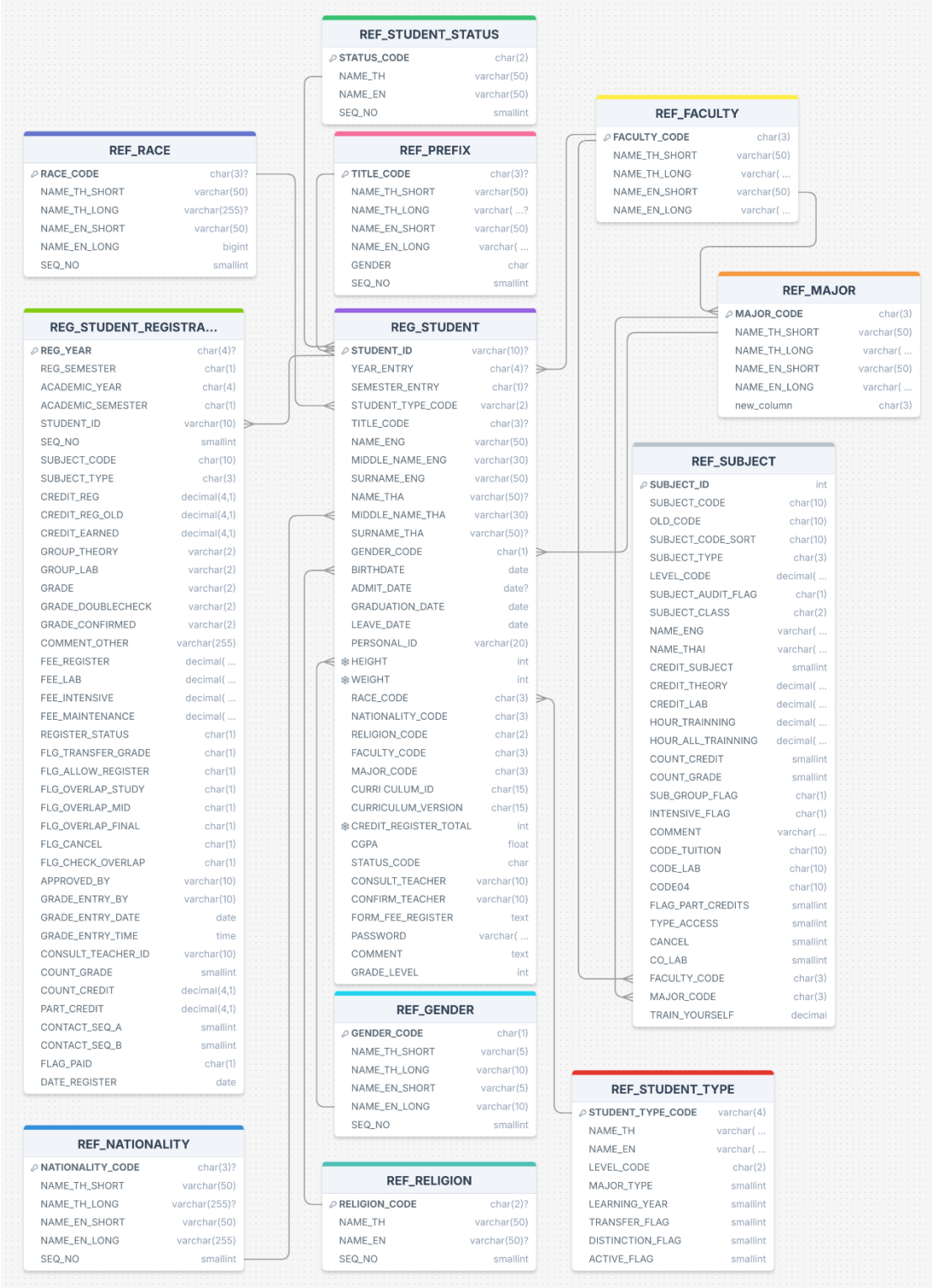
1. **Pandas**: จัดการตารางข้อมูล/อ่าน-เขียน CSV, คัดกรองคอลัมน์

ขั้นตอนการใช้งาน (Workflow)

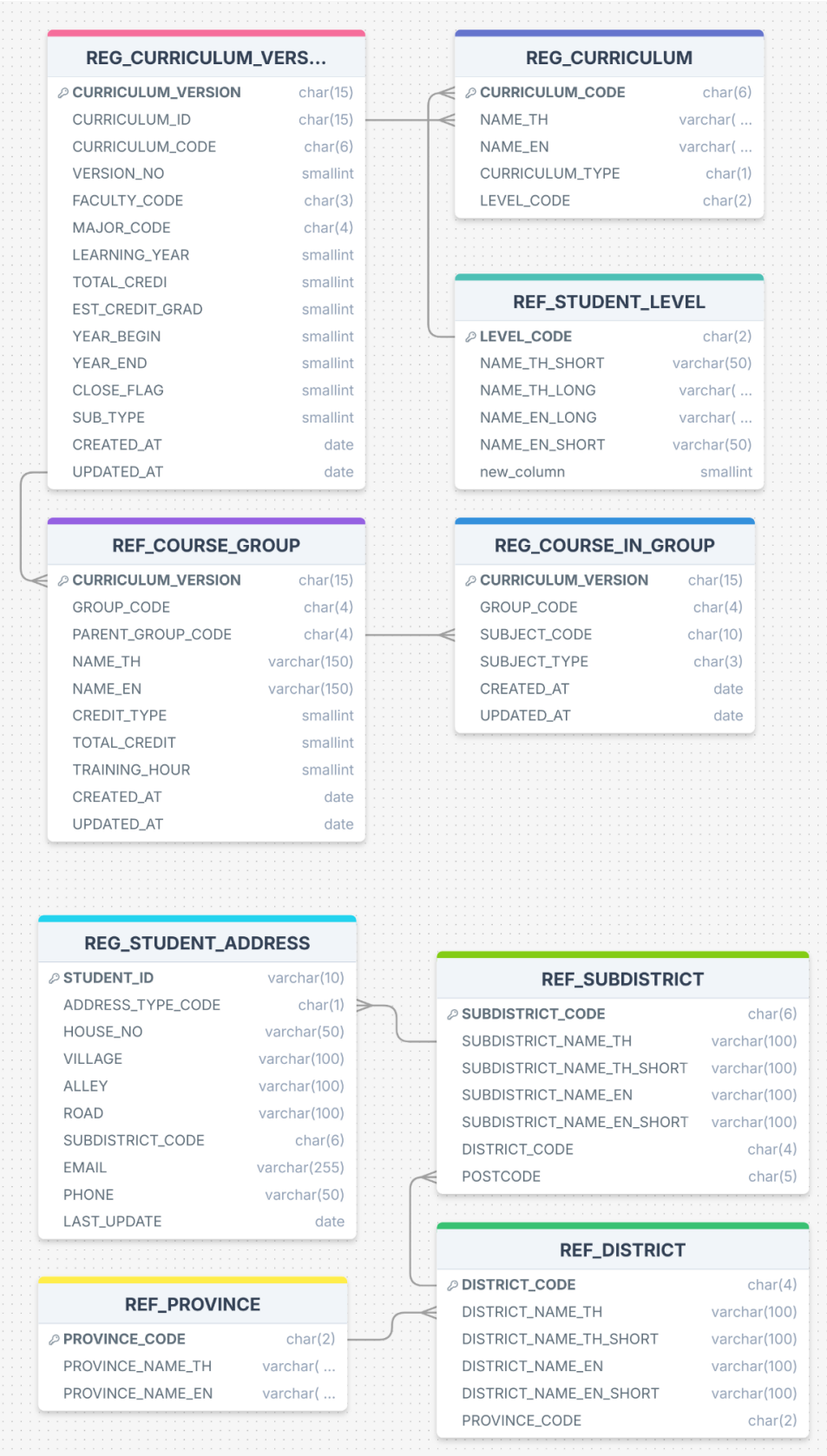
1. คลิก เลือกไฟล์ CSV แล้วเลือกไฟล์ต้นฉบับ
2. ในกล่อง “เลือกคอลัมน์สำหรับตรวจสอบ” ให้ กด Ctrl/Shift เพื่อเลือกหลายคอลัมน์ (เช่น STUDENT_ID, REG_YEAR, REG_SEMESTER)
3. คลิก แสดงข้อมูลซ้ำ เพื่อดูตัวอย่างรายการซ้ำในตาราง และตรวจดู สถิติข้อมูล
4. เมื่อตรวจสอบแล้ว คลิก สร้างไฟล์ข้อมูลไม่ซ้ำ > เลือกที่จัดเก็บและชื่อไฟล์ผลลัพธ์ (เช่น *_nodup.csv)
5. เปิดไฟล์ผลลัพธ์เพื่อตรวจทานก่อนนำไปใช้ต่อ

ภาคผนวก ง

Entity-Relationship Diagram (ERD) ระบบต้นแบบ



ภาพที่ 31 ภาพ Entity-Relationship Diagram 1



ภาพที่ 32 ภาพ Entity-Relationship Diagram 1