การสังเคราะห์อย่างรวดเร็วของคลาสส่วนน้อยโดยใช้เครือข่ายฝ่ายตรงข้ามเชิงกำเนิดสำหรับปัญหาการจำแนกประเภทข้อมูลที่ไม่สมดุล

Fast synthesis of the minority class using generative adversarial networks for imbalanced data classification problems

วฤษายย์ ร่มสายหยุด[1*]
Walisa Romsaiyud[1*]

Received: 21 April 2023 ; **Revised**: 28 June 2023 ; **Accepted**: 17 July 2023

บทคัดย่อ

อัลกอริทึมเครือข่ายฝ่ายตรงข้ามเชิงกำเนิด (แกน) คือ คลาสของเครือข่ายประสาทแบบเชิงลึก ที่สามารถสร้างตัวอย่างข้อมูลในสถานการณ์ข้อมูลที่ไม่สมดุลได้ ซึ่งแกนประกอบด้วย 2 ส่วนที่เกิดขึ้นพร้อมกัน ได้แก่ ส่วนการสร้างแบบจำลองเชิงกำเนิด และส่วนแยกแยะ โดยส่วนการสร้างแบบจำลองเชิงกำเนิดจะสุ่มข้อมูลจากชุดข้อมูลฝึกสอน และส่วนแยกแยะจะดำเนินการแยกแยะข้อมูลจากข้อมูลตัวอย่างที่สร้างขึ้นและจากข้อมูลจริง การวิจัยนี้ศึกษาการถ่ายโอนข้อมูลที่ทับซ้อนกันระหว่างการสร้างแบบจำลองบนข้อมูลแบบสตรีมมิงในเวลาเรียลไทม์ โดยนำเสนอวิธีการสร้างส่วนขยายใหม่บนแกนที่เรียกว่า แกนทูที (GANs2T) บนพื้นฐานของอนุกรมเวลาแบบตาราง เพื่อเพิ่มประสิทธิภาพการทำงานของแบบจำลองและระยะเวลา ซึ่งใช้วิธีการจับข้อมูลโครงสร้างของค่าความแปรปรวนร่วมกันของข้อมูลจากคลาสกลุ่มน้อย และสร้างข้อมูลสังเคราะห์จากความน่าจะเป็น ด้วยการใช้อัลกอริทึมนี้เรียนรู้ข้อมูลบนข้อมูลแบบสตรีมมิง การวัดประสิทธิภาพโดยดำเนินการกับวิธีการเรียนรู้ของข้อมูลที่ไม่สมดุล ทั้งจากไบนารีคลาส และมัลติคลาสจากชุดข้อมูลมาตรฐานจำนวนหลายชุด ซึ่งผลลัพธ์ที่ได้พบว่า GANs2T ร่วมกับอัลกอริทึม XGBoost (GANs2T+XGBoost) มีค่าความถูกต้อง = 84.93%, ค่าความแม่นยำ = 90.48%, ค่าความครบถ้วน = 88.13%, ค่าประสิทธิภาพโดยรวม = 89.53% และค่าเฉลี่ยของเวลาในการฝึกสอนแบบจำลองเท่ากับ 60.20 วินาที

คำสำคัญ: แบบจำลองการเรียนรู้ของเครื่อง, ข้อมูลที่ไม่สมดุล, อัลกอริทึมเครือข่ายฝ่ายตรงข้ามเชิงกำเนิด, วิธีการเพิ่มข้อมูลสังเคราะห์, คลาสส่วนน้อย

Abstract

Generative Adversarial Networks (GANs) are a class of deep neural networks that can be used to generate data examples in imbalanced data situations. GANs consist of two simultaneously trained modes: generative and discriminative modeling. The generative model generates new data as random noise from the training dataset, and the discriminator model distinguishes examples from generated new data and real data. We studied the overlapping data transfer during a generating model in distributed real-time data streaming. This paper proposes a new extension method on GANs called GANs2T based on tabular time series to improve the model performance and execution time.

We use this technique to capture the covariance structure of the minority class and to generate synthetic samples along the probability contours for learning algorithms on streaming data. The experimental testing is performed on binary-class and multi-class imbalanced learning methods from several benchmark datasets. The results validate GANs2T combine with the XGBoost (GANs2T+XGBoost) algorithm for the overall accuracy = 84.93%, precision = 90.48%, recall = 88.13%, F1-score = 89.93 and average training time for training model = 60.20 seconds.

Keywords: Machine learning model, imbalanced data, generative adversarial networks algorithm, synthetic data augmentation approach, minority class

---

[1] รองศาสตราจารย์ สาขาวิชาวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยสุโขทัยธรรมาธิราช นนทบุรี 11120
[1] Associate Professor, School of Science and Technology, Sukhothai Thammathirat Open University, Nonthaburi, 11120
[*] Corresponding author: E-mail: walisa.rom@stou.ac.th

บทนำ

การเรียนรู้ของเครื่อง (machine learning: ML) ได้ถูกนำมาประยุกต์ใช้กับงานด้านต่างๆ จำนวนมาก อาทิการเรียนรู้จำรูปแบบ (pattern recognition) การวิเคราะห์เชิงทำนาย (predictive analytics) หรือการทำงานแบบอัตโนมัติ (automation) ต่างๆ (Theobald, 2021) เช่น การเรียนรู้จำรูปแบบการฉ้อโกงเงินจากบัญชีต่างๆ ของลูกค้า การวิเคราะห์เชิงทำนายการโจมตีเครื่องเซิร์ฟเวอร์ (server) จากผู้ไม่ประสงค์ดี การทำงานแบบอัตโนมัติสำหรับการทำนายผลการออกกลางคัน (dropout) ของนักศึกษา หรือการวิเคราะห์ผลการตรวจหามะเร็งจากภาพการตรวจคลื่นแม่เหล็กไฟฟ้า (magnetic resonance imaging: MRI) เป็นต้น ซึ่งการดำเนินการต่างๆ เหล่านี้อาศัยหลักการจำแนกข้อมูล (data classification) ของการเรียนรู้ของเครื่องแบบมีผู้สอน (supervised machine learning) โดยจะทำงานได้อย่างมีประสิทธิภาพสูงมากยิ่งขึ้น ก็เมื่อข้อมูลฝึกสอน (training data) สำหรับสร้างแบบจำลอง (model) เป็นข้อมูลที่มีความสมดุลกันของข้อมูลภายในคลาสผลลัพธ์ (result class) ตัวอย่างเช่น การทำนายการออกกลางคัน (dropout) ของนักศึกษาระดับปริญญาตรี จำนวน 1,000 คน โดยจะต้องมีข้อมูลการออกกลางคัน และไม่ออกกลางคันจำนวนเท่ากันหรือใกล้เคียงกัน เช่น สัดส่วน 500:500 (ซึ่งในความเป็นจริงนักศึกษาที่ออกกลางคันมีจำนวนน้อย) หรือในกรณีของภาพจากการทำ MRI สำหรับการตรวจหามะเร็งปอดในระยะแรกจะต้องมีภาพที่เป็นปอดปกติ และภาพที่มีเซลล์มะเร็งอย่างละเท่ากัน (ซึ่งในความเป็นจริงภาพเซลล์มะเร็งอาจจะมี 1 ภาพจากทั้งหมด 50,000 ภาพ) จะเห็นได้ว่าในความเป็นจริงเป็นไปได้น้อยมาก เนื่องจากข้อมูลส่วนใหญ่เป็นข้อมูลที่ไม่สมดุล (imbalanced data) ส่งผลให้แบบจำลองการวิเคราะห์เชิงทำนายผลมีประสิทธิภาพลดลง หรือการทำนายผลมีโอกาสเกิดความคลาดเคลื่อนสูง

จากผลงานวิจัยจำนวนมาก และการศึกษาค้นคว้าของหน่วยงานวิจัยต่างๆ ได้ให้ความสำคัญกับการแก้ปัญหาการจำแนกประเภทที่ไม่สมดุล (imbalanced classification problems) เช่น งานวิจัยของ Alberto (2018) นำเสนอวิธีการสุ่มตัวอย่าง (sampling method) ทั้งการสุ่มตัวอย่างแบบน้อย (random under-sampling) และการสุ่มตัวอย่างแบบมาก (random over-sampling) โดยการสุ่มตัวอย่างแบบน้อยจะนำข้อมูลจากคลาสส่วนใหญ่ (majority class) มาลดขนาดลงให้มีขนาดเท่ากับคลาสส่วนน้อย (minority class) ซึ่งอาจทำให้ข้อมูลบางส่วนหายไป และการสุ่มตัวอย่างแบบมากจะทำการเพิ่มขนาดของคลาสส่วนน้อย ให้มีค่าเท่ากับคลาสส่วนใหญ่ ซึ่งจะส่งผลให้แบบจำลองเกิดการฟิตเกินไป (overfit) กับข้อมูลฝึกสอน (training data) เมื่อใช้งานจริงทำนายผลผิดพลาดจากปัญหาดังกล่าวต่อมาได้มีการพัฒนาเทคนิคที่เรียกว่า “SMOTE” หรือ Synthetic Minority Oversampling TEchnique ซึ่งถูกพัฒนาโดย Chawla *et al* (2002) ได้รับความนิยมอย่างมากในการแก้ปัญหาการไม่สมดุลของข้อมูล โดยใช้หลักการของอัลกอริทึม เค-เพื่อนบ้านที่ใกล้ที่สุด (k-Nearest Neighbors) โดยนำมาประยุกต์ใช้ในการคำนวณจากคลาสส่วนน้อยที่ใกล้กันและสังเคราะห์ข้อมูลใหม่ขึ้นมาให้มีสัดส่วนเท่ากับคลาสส่วนใหญ่ แต่วิธีนี้มีข้อจำกัดในกรณีที่เมื่อข้อมูลมีจำนวนน้อยมากอาจจะทำให้ประสิทธิภาพแบบจำลองต่ำ เช่น มีข้อมูล 1 รายการ หรือสัดส่วน 1000000:1 ตัวอย่างกรณีลูกค้าธนาคารที่ปกติเป็นลูกค้าที่ไม่ฉ้อโกง แต่อาจจะมีบางคนหรือส่วนน้อย (He & Ma, 2013) เช่น 1 คนที่ฉ้อโกง หรือกรณีข้อมูลภาพจากฟิลม์เอ็กซ์เรย์ที่ปกติเป็นภาพปอดแบบปกติ แต่มีภาพมะเร็งที่ปอดในตำแหน่งใหม่ (ไม่เคยมีมาก่อน) เพียง 1 ภาพ นอกจากนี้มีการนำหลักการเรียนรู้เชิงลึก (deep learning: DL) มาประยุกต์ใช้ในการสังเคราะห์ข้อมูลใหม่ให้เหมือนกับข้อมูลเดิม เช่น อัลกอริทึมเครือข่ายฝ่ายตรงข้ามเชิงกำเนิด หรือเรียกว่า แกน (generative adversarial networks: GANs) (Jakub & Vladimir, 2019) ที่อาศัยหลักการทำงานของเครือข่ายประสาทเทียม (neural network) ประกอบด้วย 2 ส่วนหลัก โดยส่วนที่ 1 ทำหน้าที่สร้างข้อมูล หรือสังเคราะห์ข้อมูล เรียกว่า ผู้สร้าง (generator) ซึ่งดำเนินการสร้างข้อมูลใหม่จากการเสริมข้อมูล (data augmentation) หรือการเพิ่มข้อมูลขึ้นมา และส่วนที่ 2 ทำหน้าที่แยกแยะข้อมูลใหม่ เรียกว่า ผู้แยกแยะ (discriminator) ซึ่งดำเนินการแยกแยะข้อมูลจริงกับข้อมูลที่สร้างขึ้นมาใหม่ ตัวอย่างการประยุกต์อัลกอริทึม GANs เช่นงานด้านคอมพิวเตอร์วิทัศน์ (computer vision) สำหรับแยกแยะรูปภาพต่างๆ หรือการสร้างภาพใบหน้าด้วยปัญญาประดิษฐ์

ดังนั้น งานวิจัยนี้ ขอนำเสนอการสังเคราะห์คลาสส่วนน้อยอย่างรวดเร็วโดยใช้เครือข่ายฝ่ายตรงข้ามเชิงกำเนิดสำหรับปัญหาการจำแนกประเภทข้อมูลที่ไม่สมดุล ตามหลักการเรียนรู้ของเครื่องแบบมีผู้สอน (supervised machine learning) ที่ข้อมูลมีป้ายกำกับ (labeled data) และสามารถทำได้ทั้งกับข้อมูลที่มีคลาสผลลัพธ์แบบไบนารีคลาสหรือ 2 คลาส (binary-class) และมัลติคลาส (multi-class) หรือหลายคลาส บนข้อมูลแบบต่อเนื่องหรือแบบสตรีมมิง (streaming data) ในเวลาเรียลไทม์ได้ และจากนั้นประเมินประสิทธิภาพแบบจำลองการเรียนรู้ของเครื่อง ในสภาพแวดล้อมจริง

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

เนื้อหาในส่วนนี้ขออธิบายทฤษฎี และงานวิจัยที่เกี่ยวข้อง จำนวน 5 ประเด็น ได้แก่ 1) ชุดข้อมูลและคลาสผลลัพธ์ 2) ข้อมูลไม่สมดุล 3) วิธีการสุ่มตัวอย่าง 4) อัลกอริทึม

เครือข่ายฝ่ายตรงข้ามเชิงกำเนิด และ 5) ข้อมูลแบบสตรีมมิง รายละเอียดดังนี้

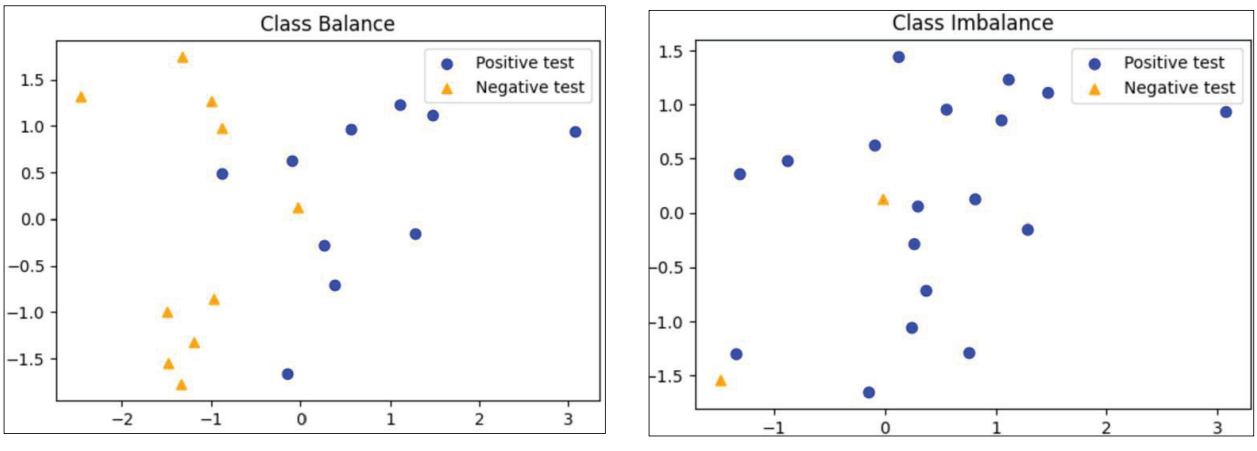
1. ชุดข้อมูลและผลลัพธ์ของคลาส (dataset and class result) โดยชุดข้อมูลคือข้อมูลต่างๆ ที่นำมาใช้ในการสร้างแบบจำลอง เช่น ชุดข้อมูลนักศึกษา ชุดข้อมูลภาพ X-ray ปอด หรือชุดข้อมูลไฟล์วิดีโอการสอบออนไลน์ เป็นต้น และผลลัพธ์ของคลาส เป็นการนำชุดข้อมูลตามคุณลักษณะของข้อมูล และค่าของข้อมูล พร้อมทั้งกำหนดคลาสผลลัพธ์ หรือข้อมูลที่มีป้ายกำกับ (labeled data) สำหรับบ่งบอกผลลัพธ์ของข้อมูลตามคุณลักษณะ และเพื่อให้อัลกอริทึมในกลุ่มการเรียนรู้แบบมีผู้สอน (supervised learning) สามารถเรียนรู้ และเข้าใจรูปแบบของข้อมูลจากคลาสผลลัพธ์ ดัง Table 1

Table 1 The example of Covid-19 dataset and result class

cough	Fever	runny nose	...	result class
Y	**39**	**Y**		**1**
N	37	N		0
Y	37	N		0
N	36	N		0
N	38	N		0

จาก Table 1 ตัวอย่างของชุดข้อมูลโควิด-19 และคลาสผลลัพธ์ ซึ่งประกอบด้วยคุณลักษณะของข้อมูลจำนวนหลายคุณลักษณะ (คอลัมน์) ได้แก่ ไอ (cough) เป็นไข้ (fever) น้ำมูกไหล (runny nose) รวมถึงคุณลักษณะอื่นๆ และคลาสผลลัพธ์ (result class) ซึ่งเป็นคุณลักษณะสุดท้ายของชุดข้อมูลนี้ ที่ประกอบด้วยค่า 2 ค่าคือ 0 หรือ 1 โดยที่ 0 หมายถึงมีผลไม่ได้ติดเชื้อโควิด-19 หรือผลตรวจเป็นลบ (negative test) และ 1 หมายถึงมีผลติดเชื้อโควิด-19 หรือผลตรวจเป็นบวก (positive test) จากความหมายของรายการที่ 1 หรือในแถวที่ 1 ที่คุณลักษณะข้อมูลไอ มีค่าเป็น Y หมายถึง มีการไอ, คุณลักษณะเป็นไข้ มีค่าเป็น 39 หมายถึง ตัวร้อนหรือมีไข้สูง, คุณลักษณะข้อมูลน้ำมูกไหล มีค่าเป็น Y หมายถึงมีน้ำมูกไหล และจากการเก็บข้อมูลผลลัพธ์ของคลาสการติดเชื้อโควิด-19 มีค่าเป็น 1 แสดงว่ามีผลติดเชื้อโควิด ซึ่งที่ผลลัพธ์ของคลาสการติดเชื้อโควิดมีค่าเป็น 0 หรือ 1 จะเรียกคลาสแบบนี้ว่าไบนารีคลาส (binary-class) หรือคลาสที่มีค่า 2 ค่าภายในคลาสผลลัพธ์ เช่น มีค่าเป็น 0 หรือ 1, T หรือ F และ Y หรือ N เป็นต้น และสำหรับคลาสที่มีค่าภายในคลาสผลลัพธ์มากกว่า 2 ค่า เรียกว่ามัลติคลาส (multi-class) เช่น 0 หรือ 1 หรือ 2 และ positive หรือ negative หรือ neutral ซึ่งมีค่า 3 ค่าภายในคลาส เป็นต้น

2. ข้อมูลไม่สมดุล (imbalanced data) หมายถึงชุดข้อมูลที่มีคลาสผลลัพธ์ หรือบางครั้งเรียกว่าผลลัพธ์เป้าหมาย (target result) มีการกระจายของข้อมูลภายในคลาสแบบไม่เท่ากัน กล่าวคือมีคลาสใดคลาสหนึ่งมีจำนวนข้อมูลที่สูงมากกว่าอีกคลาสหนึ่งเป็นจำนวนมาก (Weiss, 2013) ดัง Figure 1



a b

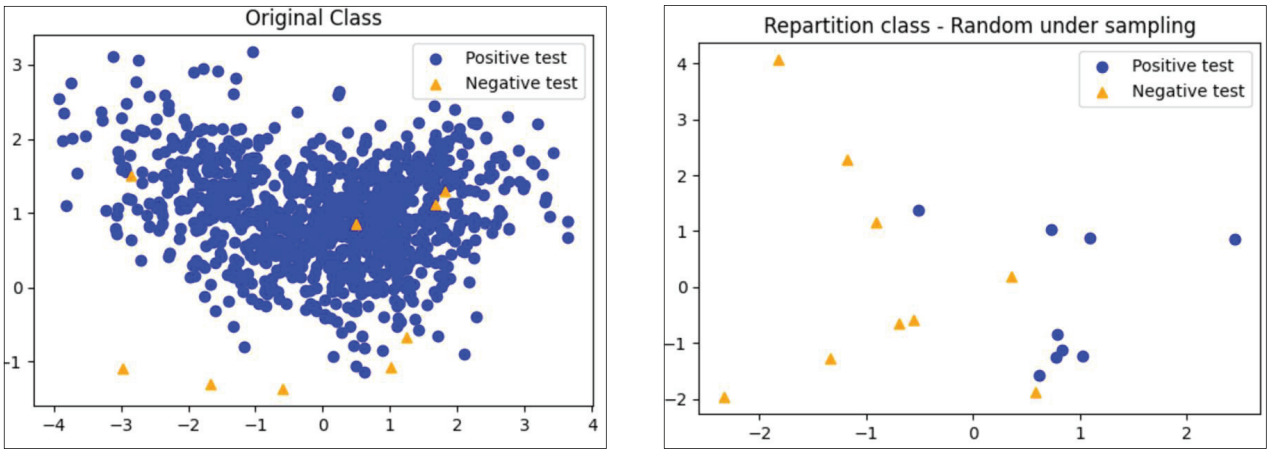
Figure 1 The balanced data and imbalanced data from 20 sample data

จาก Figure 1 แสดงข้อมูลแบบสมดุล และข้อมูลแบบไม่สมดุล ของชุดข้อมูลผู้ป่วยโควิด-19 และคลาสผลลัพธ์ ซึ่งมีคลาส 2 คลาสได้แก่ คลาสผลตรวจเป็นลบ (negative test) และคลาสผลตรวจเป็นบวก (positive test) จาก Figure 1a. แสดงจำนวนคลาสผลตรวจเป็นลบ (เป็นรูปสามเหลี่ยม และสีส้ม) มีจำนวน 10 รายการ และคลาสผลตรวจเป็นบวก (เป็นรูปวงกลม และสีน้ำเงิน) มีจำนวน 10 รายการ เท่ากันทั้ง 2 คลาส (จากข้อมูลตัวอย่างทั้งหมด 20 รายการ) กล่าวคือมีจำนวนผู้ป่วยที่เป็นโควิด-19 และไม่เป็นโควิด-19 มีจำนวนเท่ากัน หรือเรียกว่าข้อมูลมีความสมดุลกัน สำหรับ Figure 1b. แสดงข้อมูลแบบไม่สมดุล ของคลาสการติดเชื้อโควิด ที่มีคลาสผลตรวจเป็นบวก มีจำนวน 18 รายการ และคลาสผลตรวจเป็นลบ มีจำนวน 2 คน ส่งผลให้ข้อมูลเกิดความไม่สมดุลของข้อมูลทั้ง 2 คลาส หรือที่เรียกว่าเกิดปัญหาข้อมูลแบบไม่สมดุล (imbalanced data problem)

ในการสร้างแบบจำลองการเรียนรู้ของเครื่องให้มีประสิทธิภาพ ทั้งในด้านความถูกต้อง (accuracy) ความน่าเชื่อถือ (reliability) และสามารถนำไปใช้งาน (implementation) ได้จริงนั้น กระบวนการเตรียมข้อมูล (data preparation) ซึ่งเป็นหนึ่งในกระบวนการที่สำคัญของกระบวนการเรียนรู้ของเครื่อง (machine learning process) ที่จะต้องจัดเตรียมข้อมูลให้พร้อมในการสร้างแบบจำลอง โดยวิธีหนึ่งที่จะต้องคำนึงถึงคือการทำให้ข้อมูลมีความสมดุล เพื่อให้แบบจำลองสามารถเรียนรู้จำรูปแบบข้อมูลจากข้อมูลในแต่ละคลาส และคุณลักษณะข้อมูลที่เท่ากัน ไม่เอนเอียงไปทางข้อมูลคลาสส่วนใหญ่ ซึ่งจะทำให้แบบจำลองไม่มีประสิทธิภาพกับข้อมูลคลาสส่วนน้อย หรือทำนายข้อมูลใหม่ที่เป็นข้อมูลส่วนน้อยไม่ได้ และมีความคลาดเคลื่อนสูง ส่งผลให้แบบจำลองไม่มีความน่าเชื่อถือ และไม่สามารถใช้งานจริงได้ ดังนั้นจึงจำเป็นต้องมีการจัดการข้อมูลที่ไม่สมดุล เพื่อให้เกิดความสมดุลของข้อมูลในการสร้างแบบจำลอง

3. วิธีการสุ่มตัวอย่าง (sampling method) เป็นวิธีที่ได้รับความนิยมในการแก้ปัญหาข้อมูลไม่สมดุล ซึ่งประกอบด้วย 2 วิธีหลัก (Japkowicz & Stephen, 2002) ได้แก่ การสุ่มตัวอย่างแบบน้อย (random under-sampling) และการสุ่มตัวอย่างแบบมาก (random over-sampling)

3.1 การสุ่มตัวอย่างแบบน้อย (random under-sampling) เป็นวิธีการสุ่มเลือกตัวอย่างจากข้อมูลของคลาสส่วนใหญ่ โดยทำการลบข้อมูลจากคลาสส่วนใหญ่ออกจากชุดข้อมูลฝึกสอน (training dataset) เพื่อให้คลาสส่วนใหญ่มีขนาดเท่ากับคลาสส่วนน้อย ดัง Figure 2



a b

Figure 2 Random under-sampling

จาก Figure 2a. แสดงข้อมูลที่คลาสผลตรวจเป็นบวก (positive test) เป็นคลาสส่วนใหญ่ และคลาสผลตรวจเป็นลบ (negative test) เป็นคลาสส่วนน้อยที่ข้อมูลมีสัดส่วนความไม่สมดุลของคลาสสูง และจาก Figure 2b. ดำเนินการลบข้อมูลจากคลาสส่วนใหญ่ หรือ คลาสผลตรวจเป็นบวกของชุดข้อมูลฝึกสอน ให้มีจำนวนคลาสเท่ากับคลาสส่วนน้อย หรือคลาสผลตรวจเป็นลบ ซึ่งส่งผลให้ทั้ง 2 คลาสมีจำนวนคลาสผลลัพธ์เท่ากัน

ตัวอย่างวิธีการสุ่มตัวอย่างแบบน้อย เช่น Tomek Links เป็นวิธีการหาคลาสที่ตรงกันข้ามในบริเวณใกล้เคียง และลบคลาสที่ตรงกันข้าม หรือคลาสส่วนใหญ่ออก จากงานวิจัยของ Jonathan *et al.* (2020) ประยุกต์ Tomek Links สำหรับทำนายผู้ใช้ที่ขายสินค้าสำหรับผู้หญิงแบบรายวัน ของข้อมูลจากเหมืองข้อมูลที่มีความไม่สมดุลของข้อมูล และงานวิจัยของ Sridhar & Sanagavarapu (2021) และJapkowicz and Stephen (2002) พบว่าวิธีการสุ่มตัวอย่างแบบน้อยเป็นการ

สร้างปัญหาอย่างมาก เนื่องจากเกิดการสูญหายของข้อมูล ทำให้การวิเคราะห์ข้อมูล เกิดขึ้นกับข้อมูลของคลาสส่วนน้อย ซึ่งอาจจะไม่เพียงพอต่อการเรียนรู้จำรูปแบบของข้อมูล ส่งผลให้แบบจำลองการเรียนรู้ของเครื่องมีประสิทธิภาพต่ำ เมื่อทำงานในสภาพแวดล้อมการทำงานจริง

3.2 การสุ่มตัวอย่างแบบมาก (random over-sampling) เป็นการสุ่มเลือกตัวอย่างจากคลาสส่วนน้อย โดยการเสริมข้อมูล หรือเพิ่มข้อมูลไปยังชุดข้อมูลฝึกสอน เพื่อให้มีขนาดเท่ากับข้อมูลของคลาสส่วนใหญ่ ดัง Figure 3

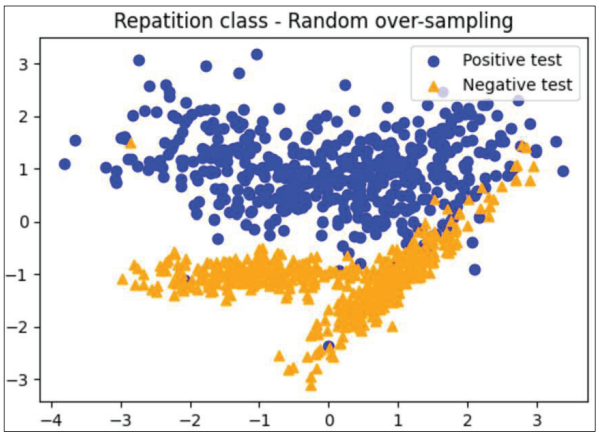


Figure 3 Random over-sampling

จาก Figure 3 การสุ่มตัวอย่างแบบมากดำเนินการโดยสร้างข้อมูลให้สมดุลด้วยวิธีการสุ่มตัวอย่างแบบมาก จากคลาสผลตรวจเป็นลบ (negative test) ซึ่งเป็นคลาสส่วนน้อย จากนั้นทำการเสริมข้อมูล หรือเพิ่มข้อมูลไปยังชุดข้อมูลฝึกสอนของคลาสส่วนน้อย ให้มีจำนวนคลาสเท่ากับคลาสส่วนใหญ่ ซึ่งส่งผลให้ทั้ง 2 คลาส มีจำนวนคลาสผลลัพธ์เท่ากัน

ตัวอย่างวิธีการการสุ่มตัวอย่างแบบมาก เช่น SMOTE มาจากชื่อเต็ม Synthetic Minority Oversampling Technique เป็นวิธีการสังเคราะห์ข้อมูลสำหรับคลาสกลุ่มน้อยในบริเวณใกล้เคียงกับข้อมูลที่มีอยู่แล้ว โดยใช้หลักการของอัลกอริทึม K-Nearest Neighbors (k-NN) ซึ่งเป็นอัลกอริทึมสำหรับจำแนกข้อมูล โดยเป็นการวิเคราะห์ข้อมูลใหม่จากข้อมูลเดิมที่อยู่ในบริเวณใกล้เคียงกันมากที่สุด จำนวน k ตัว หรือกล่าวได้ว่าการกำหนดค่า k คือการกำหนดว่าจะวิเคราะห์ข้อมูลที่ใกล้ข้อมูลที่ต้องการจำแนกที่สุดจำนวนกี่ข้อมูล (พิจารณาจากค่า k เช่น k = 1, 2, 3..., n) จากนั้นทำการหาระยะทางแบบยุคลิด (Euclidean distance) สำหรับหาค่าระยะทางระหว่างจุดสองจุดในแนวเส้นตรง เพื่อสังเคราะห์ข้อมูลของคลาสส่วนน้อยให้มีจำนวนเพิ่มมากขึ้นในบริเวณระหว่างระยะทางของจุด 2 จุด ดังสมการที่ 1 (Maureen *et al*., 2016)

$$X_{new} = X_i + (\hat{X} - X_i) * \delta \quad (1)$$

จากสมการที่ 1 โดย X_{new} คือ ตำแหน่งของข้อมูลจากคลาสส่วนน้อยที่ถูกสังเคราะห์ขึ้นมาใหม่ ระหว่างตำแหน่งของ X_i และ $\hat{X}$ ของคลาสส่วนน้อย และ δ เป็นค่าการสุ่มตัวเลขระหว่าง [0,1] ดัง Figure 4

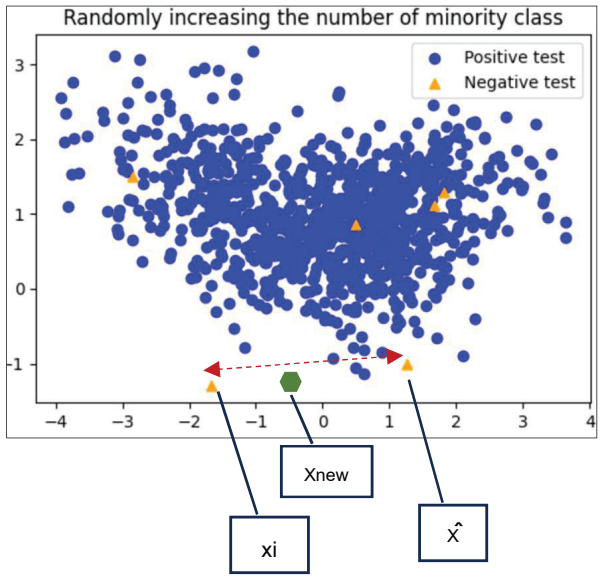


Figure 4 The synthetic data with SMOTE method

จาก Figure 4 ข้อมูลของคลาสส่วนน้อย หรือคลาสผลตรวจเป็นลบ (negative test) ที่แสดงด้วยรูปสามเหลี่ยมสีส้ม จะทำการสังเคราะห์ข้อมูลขึ้นมาใหม่ ตามจำนวนสัดส่วนของคลาสส่วนใหญ่ต่อคลาสส่วนน้อย เพื่อให้เกิดความสมดุลของคลาส ดังนั้นจะมีการสังเคราะห์คลาสส่วนน้อยเพิ่มมากขึ้น ซึ่งมีขั้นตอนดำเนินการดังนี้ 1) คำนวณหาจำนวนตำแหน่งทั้งหมดที่จะสังเคราะห์ข้อมูลใหม่ด้วยอัลกอริทึม k-NN เพื่อหาจำนวนค่า k ตำแหน่งในข้อมูลคลาสส่วนน้อย, 2) เลือกตำแหน่ง X_i และตำแหน่ง $\hat{X}$ จากการกระจายข้อมูลในคลาสกลุ่มน้อย, 3) คำนวณหาระยะทางระหว่างตำแหน่ง X_i และ $\hat{X}$ ด้วยวิธีการหาระยะทางแบบยุคลิด (Euclidean distance) เพื่อหาระยะทางระหว่างตำแหน่งของจุด 2 จุดในแนวเส้นตรงที่ใกล้ที่สุด (ตามแนวเส้นประสีแดง), 4) สร้างเวกเตอร์คุณลักษณะ (feature vector) เพื่อเก็บค่าข้อมูลและตำแหน่งที่สร้าง, 5) นำข้อมูลของ 2 ตำแหน่งมาลบกัน $(\hat{X} - X_i)$ ซึ่งผลลัพธ์ที่ได้จากค่าผลต่างของระยะห่างจะถูกนำมาคูณด้วยตัวเลขสุ่ม δ ซึ่งมีค่าระหว่าง (0,1) เพื่อให้ได้ตำแหน่งข้อมูลที่สังเคราะห์ขึ้นมาใหม่ (Xnew) แสดงด้วยรูปหลายเหลี่ยมสีเขียว ที่อยู่ห่างจากข้อมูลในตำแหน่งที่ X_i และ 6) การระบุตำแหน่งของข้อมูลใหม่ที่ทำการสังเคราะห์ขึ้นมาคือค่า X_{new} ระหว่างตำแหน่ง X_i และ $\hat{X}$ ได้อย่างถูกต้อง

จากงานวิจัยของ วิทยา ปัญญา และ วฤษาย์ ร่มสายหยุด (2565) ประยุกต์ใช้วิธีการ SMOTE ในการแก้ไขปัญหาข้อมูลที่ไม่สมดุลสำหรับการพยากรณ์การหลบหนีของผู้ขอปล่อยชั่วคราวในคดียาเสพติด จากชุดข้อมูลของศาลจังหวัดพะเยา หรืองานวิจัยของ Bao and Yang (2023) ดำเนินการปรับปรุงประสิทธิภาพของ SMOTE ให้มีประสิทธิภาพเพิ่มมากขึ้นในการสังเคราะห์ข้อมูลของเครือข่ายประสาทเทียม (neural network) และนอกจากนี้มีงานวิจัยจำนวนมากดำเนินการเปรียบเทียบประสิทธิภาพของ SMOTE และ ADASYN เช่น งานวิจัยของ พุทธิพร ธนธรรมเมธี และเยาวเรศ ศิริสถิตย์กุล (2561) พัฒนาตัวแบบการพยากรณ์ภาวะข้อเข่าเสื่อมในผู้สูงอายุ โดยข้อมูลกลุ่มน้อย คือข้อมูลที่สนใจ และการจำแนกความผิดพลาดที่เกิดขึ้นได้สูงกว่าข้อมูลกลุ่มมาก ซึ่งข้อมูลชุดนี้มีจำนวนรวมของคลาส 0 และคลาส 1 สูงกว่าคลาส 2 และคลาส 3 เป็นจำนวนมาก จึงเกิดความไม่สมดุลของข้อมูลส่งผลให้การจำแนกข้อมูลผิดพลาดได้ การปรับความไม่สมดุลของข้อมูลทั้งแบบ 2 คลาส และแบบ 3 คลาส มีผลการดำเนินงานทำให้แบบจำลองนี้สามารถนำมาใช้ในแผนส่งเสริมสุขภาพเพื่อวินิจฉัยและบำบัดผู้สูงอายุ นอกจากวิธีของ SMOTE แล้วยังมีวิธีอื่นที่พัฒนาต่อยอดจาก SMOTE เช่น SMOTENC, SMOTEEN, ADASYN, BorderlineSMOTE, KMeansSMOTE หรือ SVMSMOTE เป็นต้น

4. อัลกอริทึมเครือข่ายฝ่ายตรงข้ามเชิงกำเนิด (generative adversarial networks: GANs) หรือเรียกอีกชื่อหนึ่งว่า เครือข่ายแกน (Brownlee, 2019) เป็นเทคนิคที่สร้างข้อมูลใหม่ให้เหมือนข้อมูลจริง โดยประกอบด้วย 2 ส่วนหลัก ได้แก่ ส่วนผู้สร้าง (generator) และส่วนผู้แยกแยะ (discriminator) ดัง Figure 5

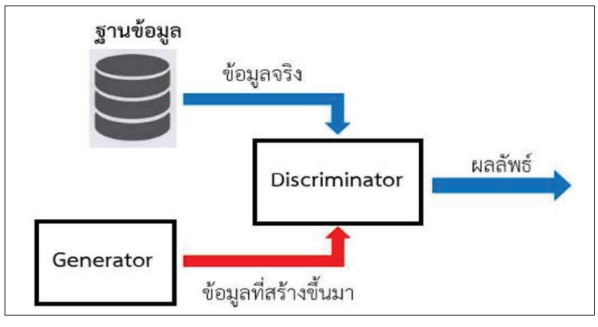


Figure 5 Generative Adversarial Networks: GANs (ภิรมย์ คงเลิศ, 2565)

จาก Figure 5 อัลกอริทึมเครือข่ายฝ่ายตรงข้ามเชิงกำเนิด (generative adversarial networks: GANs) หรือเครือข่ายแกน เป็นอัลกอริทึมที่ดำเนินการหารูปแบบในการเรียนรู้ด้วยตัวเอง โดยแบบจำลองจะทำการสร้างผลลัพธ์จากข้อมูลนำเข้า (input data) และได้ผลลัพธ์ใหม่ จากภาพเริ่มต้นจากการนำชุดข้อมูลจากฐานข้อมูล เช่น ข้อมูลไฟล์ภาพ ซึ่งเป็นข้อมูลจริงจากฐานข้อมูล และส่วนผู้สร้าง (generator) ทำหน้าที่สร้างข้อมูลใหม่ และนำข้อมูลที่สร้างขึ้นมา เข้าสู่ส่วนผู้แยกแยะ (discriminator) ที่ทำหน้าที่ตัดสินหรือแยกแยะว่าข้อมูลที่รับเข้ามานั้น เป็นข้อมูลจริง (จากฐานข้อมูล) หรือข้อมูลที่สร้างใหม่ขึ้นมา (จากส่วนผู้สร้าง) โดยที่ส่วนผู้สร้างจะต้องพยายามสร้างข้อมูลให้เหมือนกับข้อมูลจริงมากที่สุด ส่วนผู้แยกแยะต้องแยกแยะว่าข้อมูลที่สร้างขึ้นมาใหม่นี้เป็นข้อมูลจริง ๆ ทำให้ส่วนผู้แยกแยะจะต้องปรับความสามารถของตนเองเพื่อไม่ให้โดนส่วนผู้สร้างหลอก โดยที่ทั้งสองส่วนนี้จะต้องแข่งขันกันเพื่อปรับปรุงการทำงานของตัวเอง เพื่อสร้างข้อมูลใหม่ที่เหมือนข้อมูลจริง

สมการที่ 2 สมการของเครือข่ายฝ่ายตรงข้ามเชิงกำเนิด หรือเครือข่ายแกน (Goodfellow *et al.*, 2020)

$$G^{min}D^{max}V(D,G) = \sum_{x \sim p\,data}(x)[\log D(x)] + \sum_{x \sim p\,z}(z)[\log(1 - D(G(z)))] \quad (2)$$

โดยที่

- $G^{min}D^{max}V(D,G)$ คือ การกำหนดเป้าหมาย เพื่อให้บรรลุเป้าหมายการทำงานของอัลกอริทึมจะต้องดำเนินการให้ส่วนผู้สร้าง (generator) มีค่าน้อย (minimum) หรือไม่เก่งในการสร้างข้อมูลใหม่ และส่วนผู้แยกแยะ (discriminator) มีค่าเป็นสูงสุด (maximum) มีความเก่งในการแยกแยะข้อมูล
- V คือ ฟังก์ชันมูลค่า (value function) ที่มูลค่าของ action หรือ state ที่มีค่าสูง แสดงว่าทำ action นั้น แล้วจะได้ reward ที่สูงตามมา โดยคำนวณหาจาก (D,G)
- $\sum_{x \sim p\,data}(x)[\log D(x)]$ คือฟังก์ชันการทำงานของส่วนผู้สร้าง ซึ่งมีวัตถุประสงค์เพื่อสร้างภาพขึ้นมาใหม่ ให้มีค่าความเป็นไปได้หรือ D(x) สูงสุดที่ทำให้ผู้แยกแยะไม่สามารถจำแนกได้ (ถ้า D(x)=1 หมายความว่าผู้แยกแยะบอกว่ารูป x เป็นรูปจริง หรือ D(x)=0 หมายความว่าผู้แยกแยะบอกว่ารูป x นั้นมาจากส่วนผู้สร้าง)

- P data(X) คือการแจกแจงความน่าจะเป็นของตัวอย่างข้อมูล
- $\sum_{x \sim p\, z}(z)$ [$\log(1 - D(G(z)))$] คือฟังก์ชันการทำงานของส่วนแยกแยะ ซึ่งมีวัตถุประสงค์เพื่อจำแนกภาพที่สร้างภาพขึ้นมาใหม่ด้วยค่า D(G(z)) กับภาพจริง
- P z(Z) คือการกระจายความน่าจะเป็นของการรบกวน (noise)

จากสมการที่ 2 โดยที่ G คือส่วนผู้สร้าง และ D คือส่วนผู้แยกแยะ ซึ่งจะมีการเรียนรู้จำและแยกแยะรูปภาพจำนวนมาก โดยจะดำเนินการทำงานซ้ำๆ เพื่อฝึกสอนแบบจำลอง ซึ่งในแต่ละรอบการทำงานอาจจะเกิดการสูญเสีย (loss) หรือทำนายไม่ถูกต้อง ดังนั้นจึงต้องมีการปรับอัลกอริทึมให้เหมาะสมเพื่อลดค่าการสูญเสีย และเพื่อให้ผู้แยกแยะสามารถจำแนกสิ่งปลอมได้อย่างถูกต้อง

งานวิจัยของ Maniyar, *et al* (2022) ทำการสังเคราะห์ภาพใบหน้าของบุคคลจากข้อมูลเสียง โดยทั้งภาพและเสียงเป็นแหล่งข้อมูลหลักจากอัลกอริทึม GANs เพื่อสร้างส่วนผู้สร้าง และส่วนผู้แยกแยะ และงานวิจัยของ Strelcenia & Prakoonwit (2022) นำแบบจำลอง GANs มาใช้สร้างข้อมูลสังเคราะห์เพื่อช่วยในการจัดประเภทกิจกรรมต่างๆ ของการฉ้อโกงบัตรเครดิตสำหรับหากิจกรรมที่แท้จริงที่จะเป็นการฉ้อโกง

5. ข้อมูลแบบสตรีมมิง (data streaming) คือข้อมูลถูกสร้างขึ้นอย่างต่อเนื่องจากแหล่งข้อมูลจำนวนมาก ตามลำดับอนุกรมเวลา t และสามารถแสดงเป็น $S_t = \{d_t, d_{t+1},\ldots, d_n\}$ ตามลำดับเวลาที่ต่อเนื่องกัน ซึ่ง S_t คือข้อมูลแบบสตรีมมิงที่เข้ามาแบบต่อเนื่อง และมีลำดับ (sequence) ณ เวลา t ที่ t = 1,2,3,…,n และ d คือขนาดของข้อมูลในแต่ละช่วงเวลา ตัวอย่างแอปพลิเคชันที่เกี่ยวข้องกับการประมวลผลข้อมูลแบบสตรีมมิง เช่น เครื่องจักรที่ทำงานแบบอัตโนมัติในฟาร์มจะส่งข้อมูลไปยังเครื่องเซิร์ฟเวอร์ตลอดเวลา เพื่อแสดงสถานะในการทำงาน การเคลื่อนที่ของโดรนจะมีการระบุตำแหน่งและส่งภาพถ่ายจากโดรนกลับมาที่หน่วยควบคุมกลาง หรือระบบนำทางในรถยนต์ ที่ใช้อุปกรณ์จีพีเอส (GPS) สามารถกำหนดตำแหน่งละติจูด (latitude) และลองติจูด (longitude) ของการเคลื่อนที่ และแสดงผลที่หน้าจอของโปรแกรมกูเกิลแมพ เป็นต้น และงานวิจัยของ Bernardo & Valle (2020) นำเสนอเทคนิคการสุ่มตัวอย่างแบบมากบนข้อมูลแบบสตรีมมิง ซึ่งเรียกว่า VFC-SMOTE โดยใช้หลักการเรียนรู้ของเครื่องแบบสตรีมมิง และประเมินประสิทธิภาพด้านเวลา ด้านการใช้หน่วยความจำ และด้านความเร็วในการกู้คืนข้อมูลที่มีประสิทธิภาพสูง สำหรับงานวิจัยของ Brophy *et al*. (2023) นำเสนอการทบทวนวรรณ (literature review) ที่เกี่ยวข้องกับจำแนกประเภทข้อมูลด้วยอัลกอริทึม GANs และ GANs บนข้อมูลแบบสตรีมมิง รวมถึงงานวิจัยของ Xiaomin Li *et al*., (2022) นำเสนอสถาปัตยกรรมที่เรียกว่า Transformer Time-Series GANs ที่ประยุกต์การสร้างภาพใหม่ ด้วยเทคนิคการลดขนาดภาพ เพื่อสร้างของข้อมูลใหม่ที่มีขนาดเล็กได้อย่างถูกต้อง ในแต่ละลำดับของอนุกรมเวลา ซึ่งในงานวิจัยนี้ได้พัฒนา และต่อยอดจากงานวิจัยของ Brophy *et al*., (2023) และ Li *et al*. (2022)

ดังนั้น ในงานวิจัยนี้ ดำเนินการโดยใช้วิธีการการสุ่มตัวอย่างแบบมาก เพื่อสังเคราะห์คลาสส่วนน้อย และสร้างวิธีการขยาย (extension method) บนอัลกอริทึมเครือข่ายฝ่ายตรงข้ามเชิงกำเนิด สำหรับแก้ไขปัญหาการจำแนกประเภทข้อมูลที่ไม่สมดุล และข้อมูลที่มีป้ายกำกับที่สามารถทำงานได้บนข้อมูลแบบต่อเนื่องหรือแบบสตรีมมิงในเวลาเรียลไทม์ได้ และประเมินประสิทธิภาพแบบจำลองในสถานการณ์จริง

วิธีดำเนินการวิจัย

เนื้อหาในส่วนนี้ประกอบด้วย 5 ประเด็น ได้แก่ 1) สถาปัตยกรรมและกระบวนการทำงาน 2) การสร้างส่วนขยายเพิ่มเติมบนอัลกอริทึมเครือข่ายฝ่ายตรงข้ามเชิงกำเนิดบนข้อมูลแบบสตรีมมิง 3) กลุ่มตัวอย่างข้อมูล และเครื่องมือที่ใช้ในการพัฒนา 4) กระบวนการทดลอง และ 5) การประเมินประสิทธิภาพการทำงาน รายละเอียดดังนี้

1. สถาปัตยกรรมและกระบวนการทำงาน โดยดำเนินการนำข้อมูลที่ไม่สมดุลจากแหล่งข้อมูลต่างๆ (data sources) โดยเป็นข้อมูลที่มีคลาสส่วนใหญ่ มากกว่าคลาสส่วนน้อย จากนั้นทำการสร้างวิธีการขยายบนอัลกอริทึมเครือข่ายฝ่ายตรงข้ามเชิงกำเนิด ดัง Figure 6

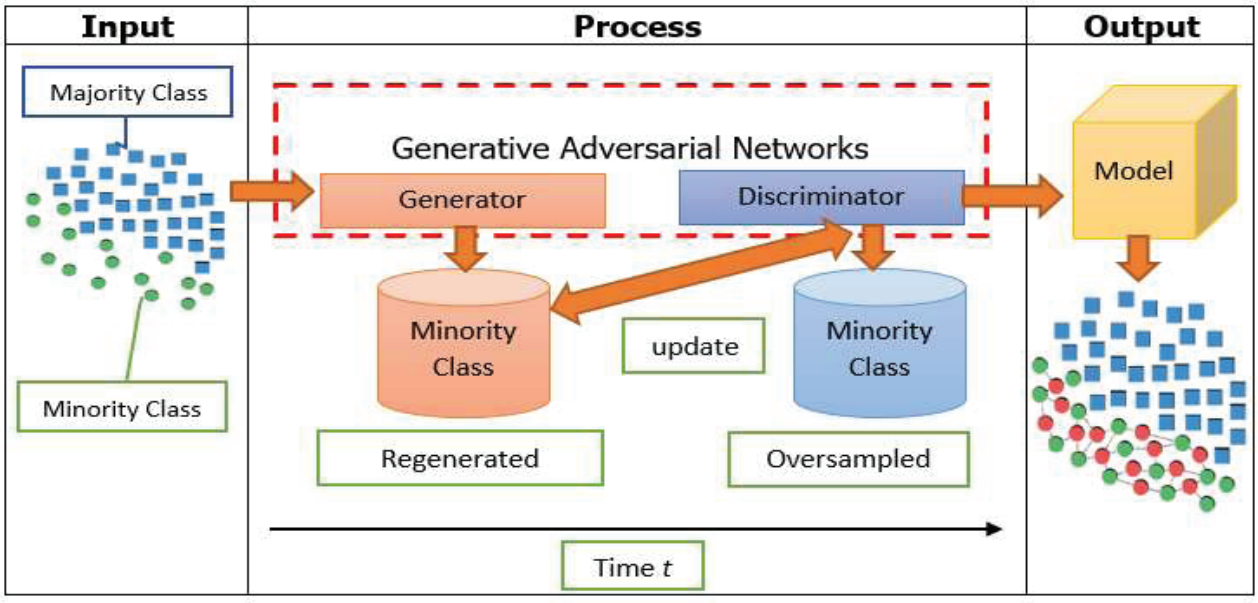


Figure 6 The overall system architecture of Fast Synthetic the Minority Class using Generative Adversarial Networks for Imbalanced Data Classification Problems

จาก Figure 6 ภาพรวมสถาปัตยกรรมการทำงานของการสังเคราะห์คลาสส่วนน้อยโดยใช้อัลกอริทึมเครือข่ายฝ่ายตรงข้ามเชิงกำเนิด หรือแกน สำหรับแก้ปัญหาการจำแนกประเภทข้อมูลที่ไม่สมดุล เริ่มต้นจาก 1) การนำข้อมูลเข้า (Input) โดยนำตัวอย่างชุดข้อมูลเกณฑ์มาตรฐาน (benchmark) จากเว็บไซต์ KEEL-dataset repository สำหรับข้อมูลที่ไม่สมดุล จำนวน 20 ชุดข้อมูล ประกอบด้วยคลาสส่วนใหญ่ (majority class) มากกว่าคลาสส่วนน้อย (minority class) จากภาพคลาสส่วนใหญ่แทนด้วยสี่เหลี่ยมสีน้ำเงิน และคลาสส่วนน้อยแทนด้วยวงกลมสีเขียว ซึ่งมีทั้งจำนวนคุณลักษณะที่แตกต่างกัน และมีจำนวนคลาสผลลัพธ์เป็นแบบไบนารีคลาส และ มัลติคลาส จากนั้นนำข้อมูลเข้าสู่การประมวลผล 2) การประมวลผลข้อมูล (data processing) ด้วยอัลกอริทึมเครือข่ายฝ่ายตรงข้ามเชิงกำเนิดที่ทำการสังเคราะห์ข้อมูลส่วนน้อยขึ้นมาใหม่ ซึ่งแบ่งเป็น 2 ส่วนได้แก่ผู้สร้าง (generator) และผู้แยกแยะ (discriminator) บนข้อมูลคลาสส่วนน้อย โดยดำเนินสร้างใหม่ (regenerated) หรือสร้างข้อมูลใหม่จากการเสริมข้อมูล (data augmentation) ขึ้นมาได้ทั้งแบบที่ไม่มีข้อมูล หรือมีข้อมูลเพียง 1 รายการ ให้ข้อมูลเกิดความสมดุล จากนั้นส่วนผู้แยกแยะจะทำการปรับปรุง (update) ข้อมูลที่ได้สังเคราะห์ขึ้นมาใหม่

โดยทำการปรับปรุงข้อมูลตลอดเวลา t และผลที่ได้จะทำการปรับปรุงข้อมูลอีกครั้งที่คลาสส่วนน้อย เพื่อให้ผู้แยกแยะทำการจำแนกประเภทข้อมูลให้มีความถูกต้องและแม่นยำ จากนั้นนำข้อมูลที่สังเคราะห์ขึ้นมาใหม่ ส่งไปที่กระบวนการส่วนผลลัพธ์ และ 3) ส่วนผลลัพธ์ (output) เป็นการนำข้อมูลที่สังเคราะห์ขึ้นมาใหม่จากการสุ่มตัวอย่างแบบมาก มาดำเนินการสร้างแบบจำลองจากข้อมูลที่สมดุลในเวลา t

2. การสร้างส่วนขยายเพิ่มเติมบนอัลกอริทึมเครือข่ายฝ่ายตรงข้ามเชิงกำเนิดบนข้อมูลแบบสตรีมมิง หรือเรียกว่า แกนทูที (GANs2T) บนพื้นฐานของอนุกรมเวลาแบบตาราง (tabular time series) เพื่อปรับปรุงแบบจำลอง

Algorithm 1: GANs2T extension method
Input:
S: data stream
T = time series-dependent rows
V = vector for building time series dictionary
Gi = Generator for each minority class
r_t = number of synthetic observations
$\hat{r}$ = probability of $\sum_i \hat{r} = 1$
The first phase:
1. S <- $d_1, d_2, \ldots d_n, \ldots, d_i \in D$
2. T <- $t_{i+1}, t_{i+2}, \ldots, t_{i+T}$
3. V <- 0,1,2,…

The second phase:
4. Create sliding window from d_T
   N = len(array) //length of the array
   callarray = sum(arr[:k]) //k-elements
   sumvalue = callarray
   for I in range(n-k):
      callarray = callarray -arr[i]+arr[i+k]
      sumvalue = max(callarray, sumvalue)
5. Optimization the sliding window

The third phase:
6. Data transformation method to tabular time series with GANs
7. Generating Synthetic data which is the number of synthetic observations
   $G_i = \hat{r} * \frac{r_i}{\sum_{i=1}^{n} r_i}$
8. Update the discriminator on GANs
   $\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} log\,(1 - D(G(z^{(i)})))$
9. Use the Binary Cross-Entropy Loss function

Output: GANs2T model

จาก Algorithm 1 การสร้างวิธีการขยาย (extension method) เพิ่มเติมบนอัลกอริทึมเครือข่ายฝ่ายตรงข้ามเชิงกำเนิด หรือเรียกว่าแกนทูที (GANs2T) ซึ่ง 2T มาจากคำว่าอนุกรมเวลาแบบตาราง (tabular time series) ซึ่งนิยมใช้ในการประมวลผลภาษาธรรมชาติ (natural language processing: NLP) มาประยุกต์ใช้ในการเพิ่มประสิทธิภาพแบบจำลองในระหว่างการถ่ายโอนข้อมูลที่ทับซ้อนกันของเวลาในขณะที่มีการสร้างแบบจำลองบนข้อมูลแบบสตรีมมิงในเวลาเรียลไทม์ การดำเนินการแบ่งเป็น 3 ระยะ ได้แก่ ระยะที่ 1: กำหนดตัวแปรหลัก 3 ตัวแปรได้แก่ data stream, time series และ vector ระยะที่ 2: ทำการสร้างวิธีการเลื่อนหน้าต่าง (sliding window method) บนข้อมูลแบบสตรีมมิงเพื่อนำขั้นตอนของเวลา (t) ก่อนหน้ามาใช้สำหรับการทำนายในขั้นตอนต่อไป และทำการปรับวิธีการเลื่อนหน้าต่างให้เหมาะสมกับข้อมูล และระยะที่ 3: ทำการสร้างอนุกรมเวลาแบบตาราง (tabular time series) บนอัลกอริทึมแกน เพื่อกำหนดโครงสร้างแบบลำดับชั้นบนข้อมูล จากนั้นแปลงข้อมูล (data transform) ด้วยการเข้ารหัสแถวแต่ละแถว เพื่อลดขนาดข้อมูลให้ประมวลผลในเวลา t ที่ข้อมูลเข้ามาต่อเนื่องกันได้ ส่งผลให้การเข้ารหัสอนุกรมเวลาแบบตารางนี้สามารถฝึกสอนข้อมูลแบบล่วงหน้าตั้งแต่ข้อมูลเริ่มเข้าสู่ระบบถึงข้อมูลปลายทางได้ จากนั้นทำการสร้างข้อมูลใหม่จากการสังเคราะห์ที่เวลา t และการปรับค่าน้ำหนักให้เป็นมาตรฐานของแต่ละโหนดภายในส่วนผู้แยกแยะของ GANs เพื่อให้สามารถแยกแยะข้อมูลได้ถูกต้องในเวลา t ใดๆ และการสร้างฟังก์ชันสูญเสียสำหรับการหาค่าผลลัพธ์กับค่าจริงว่ามีความแตกต่างกันอย่างไร สุดท้ายได้ผลลัพธ์ตัวจำแนกแยกแยะข้อมูลระหว่างคลาส และปรับคลาสกลุ่มน้อยให้เพิ่มมากขึ้นได้

3. กลุ่มตัวอย่างข้อมูล และเครื่องมือที่ใช้ในการพัฒนา

3.1 กลุ่มตัวอย่างข้อมูล งานวิจัยนี้นำตัวอย่างชุดข้อมูลเกณฑ์มาตรฐาน (benchmark) จากเว็บไซต์ KEEL-dataset repository (KEEL-dataset, 2023) สำหรับข้อมูลที่ไม่สมดุล จำนวน 20 ชุดข้อมูล มีคลาสส่วนใหญ่ มากกว่าคลาสส่วนน้อย ใน 4 รูปแบบ จำแนกตามแต่ละสัดส่วน ได้แก่ 1) 1.5-9, 2) 9.1-40, 3) 41-100 และ 4) มากกว่า 100 และมีจำนวนคุณลักษณะที่แตกต่างกัน รวมถึงมีจำนวนคลาสผลลัพธ์ทั้งแบบไบนารีคลาส และแบบมัลติคลาส ที่ผ่านกระบวนการเตรียมข้อมูล (data preparation) เรียบร้อยแล้ว กล่าวคือเป็นข้อมูลที่พร้อมสำหรับการสร้างแบบจำลอง ซึ่งมีการคัดเลือกคุณลักษณะ (feature selection) เสร็จสิ้น ทำให้ข้อมูลเหล่านี้พร้อมใช้งานจริง และเป็นชุดข้อมูลที่งานวิจัยจำนวนมากยอมรับ และได้รับการตีพิมพ์เผยแพร่ในวารสารชั้นนำระดับนานาชาติสำหรับการทดสอบประสิทธิภาพของอัลกอริทึม

Table 2 The example dataset for 20 items

No.	dataset title	No. of feature	Ratio
	wine	9	1.5
	Iris0	4	2
	Vehicle3	9	3.2
	Glass6	9	6.38
	Glass0	8	8.44
	Yeast-2_vs_4	8	9.08
	Glass2	9	10.29
	yeast-1-2-8-9_vs_7	8	30.57
	Yeast5	8	32.73
	Thyroid	7	36.94
	lymphography	8	40.5
	winequality-white-3_ vs_7	11	44
	winequality-red-8_ vs_6-7	11	46.5
	winequality-white-3-9_ vs_5	11	58.28
	shuttle-2_vs_5	9	66.67
	ecoli	8	71.50
	poker-8_vs_6	10	85.88
	kddcup-rootkit-imap_ vs_back	41	100.14
	pageblocks	10	164
	shuttle	9	853

จาก Table 2 ชุดข้อมูลตัวอย่างจำนวน 20 รายการ โดยสัดส่วน 1000 ตัวอย่าง เช่น ชุดข้อมูลลำดับที่ 1 คือ wine มีจำนวนคุณลักษณะ = 9 คุณลักษณะ (feature) และมีสัดส่วน 1.5 หมายความว่า 1:1.5 หรือจำนวนคลาสส่วนใหญ่ 1500 คลาส ต่อจำนวนคลาสส่วนน้อย 1000 คลาส โดยข้อมูลลำดับที่ 1-5 เป็นข้อมูลที่มีค่าสัดส่วนระหว่าง 1.5-9, ข้อมูลลำดับที่ 6-10 เป็นข้อมูลที่มีค่าสัดส่วนระหว่าง 9.1-40, ข้อมูลลำดับที่ 11-17 เป็นข้อมูลที่มีค่าสัดส่วนระหว่าง 41-100 และข้อมูลลำดับที่ 18-20 เป็นข้อมูลที่มีค่าสัดส่วนมากกว่า 100

Table 3 Setting values for parameters of GANs

Parameter	Generator Value	Discriminator Value
The total neurons per hidden layer	128, 256, 512, 1024	512, 256, 128
Optimizer	Adam (15.00 steps)	Adam (15.00 steps)
Loss Function	BCEWith LogitsLoss	BCEWithLogitsLoss
Learning Rate	Le-4	Le-4
Batch-size for training	128	128

จาก Table 3 การกำหนดค่าพารามิเตอร์ของ GANs ประกอบด้วยพารามิเตอร์จำนวน 5 พารามิเตอร์หลัก ได้แก่ The total neurons per hidden layer, Optimizer, Loss Function, Learning Rate และ Batch-size for training ของค่า Generator และ ค่า Discriminator

3.2 เครื่องมือ (รวมถึงวิธีการสร้าง และการตรวจสอบคุณภาพ) เครื่องมือที่ใช้คือ Google-Colab และใช้ภาษาไพทอน (Python) ในการพัฒนา โดยเพิ่มประสิทธิภาพการประมวลผลด้วยหน่วยประมวลผลกราฟิกส์ (Graphics Processing Unit: GPU) สำหรับการจัดการกระแสข้อมูลด้วย Apache Spark ใช้ไลบรารี PySpark รุ่น 3.0.0 โดยเรียกใช้ไลบรารี (library) ของ GAN_DeepLearning4J ในการสร้างแบบจำลองของ GANs บนกระแสข้อมูล และทำการประเมินผลแบบจำลองด้วยไลบรารีของ pyspark.ml.evaluation model จำนวน 2 โมดูล ได้แก่ Binary ClassificationEvaluator และ MulticlassClassification Evaluator และวัดประสิทธิภาพแบบจำลองจากค่าความถูกต้อง (accuracy) ค่าความแม่นยำ (precision) ค่าความครบถ้วน (recall) ค่าประสิทธิภาพโดยรวม (F-measure) และค่าเฉลี่ยของเวลา (average training time) ในการฝึกสอนแบบจำลอง

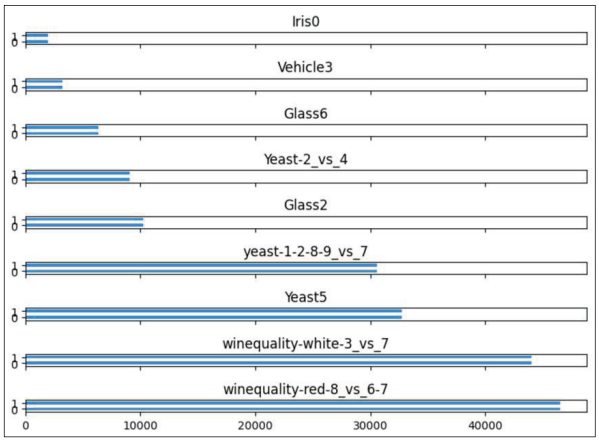
4. กระบวนการทดลอง ดำเนินการใน 3 ประเด็น ได้แก่ การสังเคราะห์ข้อมูลใหม่ด้วย GANs2T ของไบนารีคลาส และแบบมัลติคลาส และ การเปรียบเทียบกระบวนการภายใน GANs ระหว่าง Discriminative และ Predictive กับ GANs2T

4.1 การสังเคราะห์ข้อมูลใหม่ด้วย GANs2T ของไบนารีคลาส และมัลติคลาส ดำเนินการโดยนำวิธีการของ GANs2T มาดำเนินการสังเคราะห์คลาสกลุ่มน้อยดัง Table 4

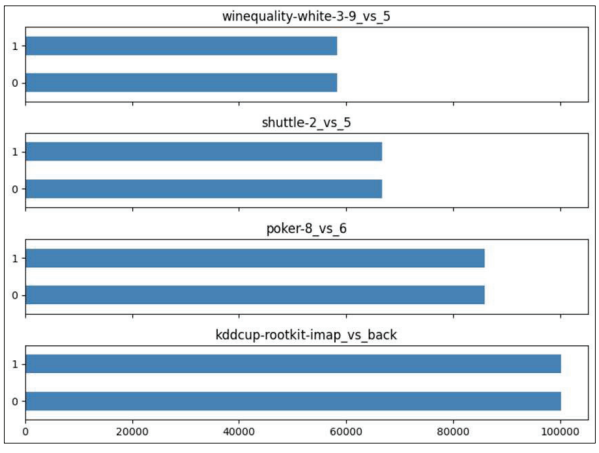
Table 4 Synthesising data with GANs2T for binary-class

No.	Dataset title	Initial class ratio	Ratio GANs2T
	Iris0	1000:2000	2000:2000
	Vehicle3	1000:3200	3200:3200
	Glass6	1000:6380	6380:6380
	Yeast-2_vs_4	1000:9080	9080: 9080
	Glass2	1000:10290	10290:10290
	yeast-1-2-8-9_vs_7	1000:30570	30570:30570
	Yeast5	1000:32730	32730:32730
	winequality-white-3_vs_7	1000:44000	44000:44000
	winequality-red-8_vs_6-7	1000:46500	46500:46500
	winequality-white-3-9_vs_5	1000:58280	**58279:58280**
	shuttle-2_vs_5	1000:66670	**66669:66670**
	poker-8_vs_6	1000:85880	**85878:85880**
	kddcup-rootkit-imap_vs_back	1000:100140	**100138:100140**

จาก Table 4 การสังเคราะห์ข้อมูลใหม่ด้วย GANs2T ของไบนารีคลาส จากทั้งหมด 13 ชุดข้อมูล พบว่าเมื่อสัดส่วนระหว่างคลาสส่วนใหญ่มีค่ามากกว่า 50000 ตัวอย่าง เมื่อใช้วิธีการสังเคราะห์คลาสส่วนน้อยด้วย GANs2T กลุ่มตัวอย่างที่มีการทับซ้อน (overlap) ของคลาสส่วนใหญ่ได้รับความสนใจจากวิธี GANs2T มากที่สุด ทำให้คลาสส่วนน้อยเกิดการสังเคราะห์ที่ผิดปกติได้ (หมายความว่าคลาสส่วนน้อยทำการสังเคราะห์ข้อมูลของคลาสใหม่ได้ไม่สอดคล้องหรือใกล้เคียงกับสัดส่วนของคลาสส่วนใหญ่) และส่งผลให้ประสิทธิภาพในการทำนายผลขอแบบจำลองต่ำลง ดังนั้นหากใช้วิธี GANs2T ในกรณีที่มีกลุ่มตัวอย่างข้อมูลและสัดส่วนมากกว่า 50000 จะต้องมีการหาข้อมูลที่ผิดปกติ (outlier) และลบออกไปก่อนการสังเคราะห์ จะทำให้ค่าสัดส่วนกลับมาเป็นปกติ ดัง Figure 7



a



b

Figure 7 Synthesising data with GANs2T for binary-class and binary-class with noise

จาก Figure 7a. แสดงการสังเคราะห์ข้อมูลใหม่ด้วย GANs2T ของไบนารีคลาส (สัดส่วนที่สังเคราะห์ข้อมูลของคลาสส่วนใหญ่ และคลาสส่วนน้อยมีค่าเท่ากันหรือใกล้เคียงกัน) จากจำนวน 9 ตัวอย่าง ที่ GANs2T สามารถสังเคราะห์คลาสกลุ่มน้อยได้เท่ากับคลาสกลุ่มส่วนใหญ่ และ Figure 7b. แสดงการสังเคราะห์ข้อมูลใหม่ด้วย GANs2T ของไบนารีคลาสแบบมีสัญญาณรบกวน หรือเรียกว่าผิดปกติจำนวน 4 ตัวอย่างที่ GANs2T สามารถสังเคราะห์คลาสกลุ่มน้อยได้ไม่เท่ากับคลาสกลุ่มส่วนใหญ่ (+/- ค่า 1-2)

Table 5 Synthesising data with GANs2T for multi-class

No.	Dataset title	Initial class ratio	Ratio GANs2T
	wine	1000:1500	1500: 1500
	Glass0	1000:8440	8440: 8440
	Thyroid	1000:36940	36940: 36940
	lymphography	1000:40500	40500: 40500
	ecoli	1000:71500	71500: 71500
	pageblocks	1000:164000	164000: 164000
	shuttle	1000:853000	853000: 853000

จาก Table 5 การสังเคราะห์ข้อมูลใหม่ด้วย GANs2T ของมัลติคลาส จากทั้งหมด 7 ชุดข้อมูล พบว่าเมื่อการสังเคราะห์ด้วยวิธี GANs2T ของข้อมูลแบบมัลติคลาส จะได้ผลดีในการสังเคราะห์คลาสกลุ่มน้อยต่อสัดส่วนข้อมูลที่มีแตกต่างระหว่างคลาสส่วน น้อย (ระหว่างค่า 1500 - 853000) และเมื่อข้อมูลมีคลาสส่วนใหญ่มีค่ามากกว่า 800000 ตัวอย่าง การสังเคราะห์คลาสส่วนน้อยก็สามารถคำนวณหาสัดส่วนได้ตามปกติ ดัง Figure 8

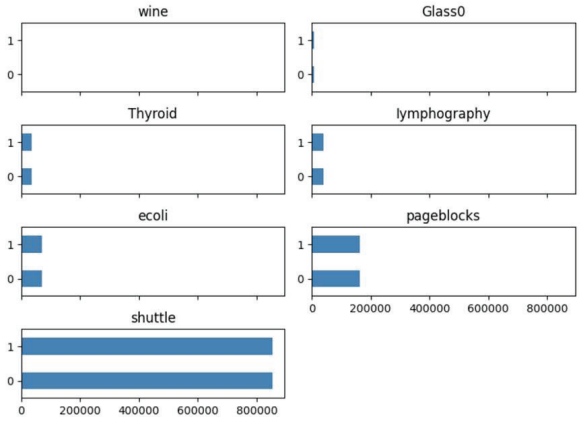


Figure 8 The results of synthetic data with GANs2T for multi-class

4.2 การเปรียบเทียบกระบวนการภายใน GANs ระหว่าง Discriminative และ Predictive กับ GANs2T เพื่อวัดประสิทธิภาพการแยกแยะ และการทำนายข้อมูลระหว่าง GANs และ GANs2T จากคุณลักษณะ (feature) และคลาสผลลัพธ์ของชุดข้อมูล ซึ่งเป็นข้อมูลทั้งแบบตัวเลข (numeric) และข้อมูลเชิงอันดับ (ordinal data) จึงใช้อัลกอริทึม XGBoost หรือ eXtreme Gradient Boosting ในการวิเคราะห์เชิงทำนาย (predictive analytics) ข้อมูลใหม่ ดัง Table 6

Table 6 The performance evaluation of discriminative and predictive measures using GANs and GANs2T models

Metric	Method	Accuracy		
		Training step 0	Training step 5000	Training step 10000
Discriminative	GANs	52.73%	**59.32%**	56.78%
	GANs2T	51.50%	55.52%	**59.24%**
Predictive	GANs	55.70%	56.96%	58.36%
	GANs2T	54.02%	56.98%	**59.44%**

จาก Table 6 การวัดประสิทธิภาพการแยกแยะ และการทำนายข้อมูลโดยใช้แบบจำลอง GANs และ GANs2T โดยวัดค่าความถูกต้อง จากการฝึกสอนข้อมูลซึ่งแบ่งเป็น 3 ขั้นตอนได้แก่ 0, 5000 และ 10000 และใช้อัลกอริทึม XGBoost สำหรับการทำนายข้อมูลใหม่ พบว่าในขั้นตอน Discriminative วิธีการ GANs มีค่าความถูกต้องสูงสุด = 59.32% สำหรับการฝึกสอนข้อมูล 5000 รอบ และในขั้นตอน Predictive วิธีการ GANs2T มีค่าความถูกต้องสูงสุด = 59.44% สำหรับการฝึกสอนข้อมูล 10000 รอบ จะเห็นได้ว่าวิธีการ GANs2T จะทำงานได้อย่างมีประสิทธิภาพบนข้อมูลที่มีการฝึกสอนจำนวนมาก (กล่าวคือทำการฝึกสอนจำนวน 10000 รอบ)

5. การประเมินประสิทธิภาพการทำงาน โดยการเปรียบเทียบประสิทธิภาพกับวิธีการสุ่มตัวอย่างแบบมากจำนวน 5 วิธี ได้แก่ SMOTE, ADASYN, BorderlineSMOTE, GANs และ GANs2T บนข้อมูลแบบสตรีมมิง ดัง Table 6

Table 7 Comparing the performance of random over-sampling with 5 methods

Methods	Overall Accuracy (%)	Average Training Time (s)
SMOTE+XGBoost	82.42	**54.80**
ADASYN+XGBoost	**80.42**	57.45
BorderlineSMOTE +XGBoost	83.96	56.11
GANs+XGBoost	84.79	**75.72**
GANs2T+XGBoost	**84.93**	60.20

จาก Table 7 การเปรียบเทียบประสิทธิภาพกับการสุ่มตัวอย่างแบบมากด้วย 5 วิธี พบว่า GANs2T มีค่า overall accuracy สูงสุด แต่ average training time ใช้เวลาค่อนข้างนานกว่าวิธีการ SMOTE แต่เร็วกว่าวิธีการของ GANs ดัง Figure 9

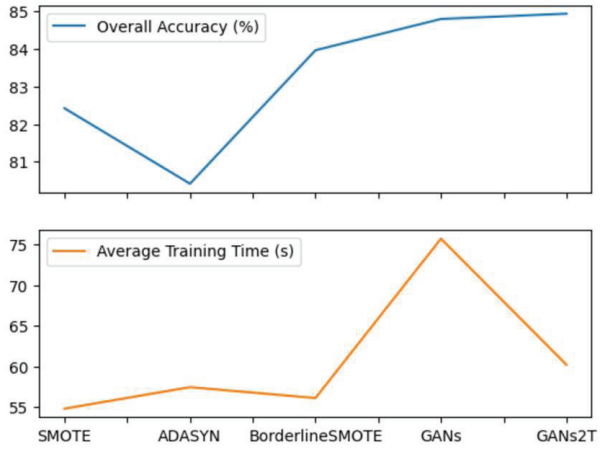


Figure 9 The comparison results with 5 methods on overall accuracy and average training time

จาก Figure 9 แสดงข้อมูลการเปรียบเทียบ 5 วิธีการ กับภาพรวมความถูกต้อง และค่าเฉลี่ยระยะเวลาในการฝึกสอนข้อมูลของข้อมูลที่ไม่สมดุล พบว่าค่าความถูกต้องของวิธีการ GANs2T มีค่าความถูกต้องสูงที่สุด = 84.93 และวิธีการ ADASYN มีค่าความถูกต้องต่ำที่สุด = 80.42 สำหรับค่าเฉลี่ยระยะเวลาในการฝึกสอนข้อมูล วิธีการ SMOTE ใช้เวลาเร็วที่สุด = 54.80 และวิธีการ GANs ใช้เวลาช้าที่สุด = 75.72 โดยรองลงมาคือ GANs2T = 60.20

จากนั้นทำการประเมินประสิทธิภาพด้วย confusion matrix กับอัลกอริทึม GANs2T+XGBoost มีค่าความถูกต้อง (accuracy) = 84.93 ค่าความแม่นยำ (precision) = 90.48 ค่าความครบถ้วน (recall) = 88.13 และค่าประสิทธิภาพโดยรวม (F1-score) = 89.53

สรุปผลการดำเนินงาน และอภิปรายผล

สรุปผลการดำเนินงาน

ผลการศึกษาการสร้างแบบจำลองการสังเคราะห์คลาสส่วนน้อยอย่างรวดเร็วโดยใช้เครือข่ายฝ่ายตรงข้ามเชิงกำเนิดสำหรับปัญหาการจำแนกประเภทข้อมูลที่ไม่สมดุล สรุปได้ดังนี้

1.1 ผลการสร้างแบบจำลองการสังเคราะห์คลาสส่วนน้อยอย่างรวดเร็วโดยใช้เครือข่ายฝ่ายตรงข้ามเชิงกำเนิดสำหรับปัญหาการจำแนกประเภทข้อมูลที่ไม่สมดุล สามารถแก้ไขปัญหาข้อมูลไม่สมดุลจากการสังเคราะห์ข้อมูลใหม่ด้วย GANs2T ได้ โดยไบนารีคลาส พบว่าสามารถสังเคราะห์ข้อมูลใหม่ตรงตามสัดส่วนของคลาสส่วนใหญ่กับข้อมูลของคลาสส่วนใหญ่มีค่าน้อยกว่า 50000 ตัวอย่าง และมัลติคลาส

พบว่าจะได้ผลดีในการสังเคราะห์คลาสกลุ่มน้อยกับสัดส่วนข้อมูลที่มีแตกต่างระหว่างคลาสส่วนน้อย (จำนวน 1500 - 853000 รายการ)

1.2 ผลการประเมินประสิทธิภาพของแบบจำลองการสังเคราะห์คลาสส่วนน้อยอย่างรวดเร็วโดยใช้เครือข่ายฝ่ายตรงข้ามเชิงกำเนิดสำหรับปัญหาการจำแนกประเภทข้อมูลที่ไม่สมดุล ซึ่งมีค่าความถูกต้อง (accuracy) ของการทำนาย (predictive) ระหว่างอัลกอริทึม GANs และ GANs2T มีค่า 58.36% และ 59.44% ซึ่งอัลกอริทึม GANs2T มีประสิทธิภาพสูงกว่าอัลกอริทึม GANs และเมื่อนำ GANs2T มาใช้งานร่วมกับอัลกอริทึม XGBoost จะมีค่าความถูกต้อง 84.93% และค่าเฉลี่ยเวลาการฝึกสอนแบบจำลอง 60.20 วินาที

อภิปรายผล

ผลการศึกษาการสร้างแบบจำลองการสังเคราะห์คลาสส่วนน้อยอย่างรวดเร็วโดยใช้เครือข่ายฝ่ายตรงข้ามเชิงกำเนิดสำหรับปัญหาการจำแนกประเภทข้อมูลที่ไม่สมดุล สามารถอภิปรายผลได้ ดังนี้

จากผลการศึกษาพบว่า การสังเคราะห์คลาสส่วนน้อยโดยใช้เครือข่ายฝ่ายตรงข้ามเชิงกำเนิด ในสถานการณ์ที่ข้อมูลไม่สมดุล และข้อมูลถูกสร้างขึ้นอย่างต่อเนื่องจากแหล่งข้อมูลจำนวนมาก หรือ streaming data จากเว็บไซต์ KEEL-dataset repository สำหรับข้อมูลที่ไม่สมดุล จำนวน 20 ชุดข้อมูล มีคลาสส่วนใหญ่ (majority class) มากกว่าคลาสส่วนน้อย (minority class) มีจำนวนคุณลักษณะ (feature) ที่แตกต่างกัน และมีจำนวนคลาสผลลัพธ์ทั้งแบบไบนารีคลาสและมัลติคลาส เข้าสู่ส่วนการประมวลผลข้อมูล (data processing) ซึ่งได้พัฒนาวิธีการขยาย (extension method) ใหม่ชื่อว่า GANs2T เป็นวิธีการที่นำข้อมูลแบบตาราง (tabular data) จากชุดข้อมูลตัวอย่าง เพื่อให้ส่วนผู้สร้าง (generator) ทำการสร้างข้อมูล และผู้แยกแยะ (discriminator) ทำการหาข้อมูลคลาสส่วนน้อย โดยดำเนินการสร้างข้อมูลใหม่จากการเสริมข้อมูล (data augmentation) บนอัลกอริทึม GANs ซึ่งจะคล้ายกับวิธีการของ Brophy *et al*. (2023) และ Li *et al*. (2022) ที่สามารถใช้อนุกรมเวลาแบบตาราง (tabular time series) มาช่วยแก้ปัญหาการสร้างแบบจำลองอย่างรวดเร็วบนข้อมูลแบบสตรีมมิงได้อย่างมีประสิทธิภาพ

การสร้างวิธีการขยายใหม่ ที่เรียกว่า GANs2T สามารถทำงานได้อัลกอริทึม GANs และสามารถทำงานร่วมกับอัลกอริทึมอื่นได้อย่างมีประสิทธิภาพ โดยการเปรียบเทียบประสิทธิภาพกับวิธีการสุ่มตัวอย่างแบบมาก จำนวน 5 วิธี ได้แก่ SMOTE+XGBoost, ADASYN+XGBoost, BorderlineSMOTE+XGBoost, GANs+XGBoost และ GANs2T+XGBoost บนข้อมูลแบบ streaming data พบว่า GANs2T มีค่า overall accuracy สูงสุด = 84.93%

ข้อเสนอแนะในประเด็นต่อยอดงานวิจัย สามารถใช้อัลกอริทึมการเรียนรู้เชิงลึก (deep learning: DL) เช่น Recurrent neural network: RNN หรือ Long Short-Term Memory: LSTM มาประยุกต์ใช้ในการทำการบนข้อมูลแบบสตรีมมิงในเวลาเรียลไทม์

กิตติกรรมประกาศ

งานวิจัยนี้ได้รับทุนสนับสนุนการวิจัยจากสาขาวิชาวิทยาศาสตร์และเทคโนโลยี และสถาบันวิจัยและพัฒนา มหาวิทยาลัยสุโขทัยธรรมาธิราช ภายใต้ทุนสนับสนุนการวิจัย เลขที่ IRD-15000-11.25/2565

เอกสารอ้างอิง

พุทธิพร ธนธรรมเมธี และเยาวเรศ ศิริสถิตย์กุล. (2561). เทคนิคการจำแนกข้อมูลที่พัฒนาสำหรับชุดข้อมูลที่ไม่สมดุลของภาวะข้อเข่าเสื่อมในผู้สูงอายุ. *วารสารวิทยาศาสตร์และเทคโนโลยี, 27*(6), 1164-1178.

ภิรมย์ คงเลิศ. (2565). หน่วยที่ 9 การเรียนรู้เชิงลึก. ใน *ประมวลสาระชุดวิชาปัญญาประดิษฐ์และการประยุกต์ หน่วยที่ 6-10*. มหาวิทยาลัยสุโขทัยธรรมาธิราช.

วิทยา ปัญญา และ วฤษาย์ ร่มสายหยุด. (2565). วิธีการสร้างแบบจำลองเชิงทำนายพฤติกรรมการผิดเงื่อนไขการปล่อยชั่วคราวของศาลจากชุดข้อมูลที่ไม่สมดุลโดยใช้เทคนิคการเรียนรู้ของเครื่อง. *วารสารวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยมหาสารคาม, 42*(2), 1686-9664.

Alberto, F., Salvador, G., Mikel, G., Ronaldo, C. P., Bartosz, K., & Francisco, H. (2018). Learning from imbalanced data sets. *Springer*. https://doi.org/10.1007/978-3-319-98074-4.

Bernardo, A., & Valle, E. D. (2020). VFC-SMOTE: very fast continuous synthetic minority oversampling for evolving data streams. *Data Mining and Knowledge Discovery, 35*, 2679-2713. https://doi.org/10.1007/s10618-021-00786-0.

Bao, Y. & Yang, S. (2023). Two novel SMOTE methods for solving imbalanced classification problems. *IEEE Access, 11*, 5816-5823. 10.1109/ACCESS.2023.3236794.

Brophy, E., Wang, Z., She, Q., & Ward, T. (2023). Generative adversarial networks in time series: A systematic literature review. *ACM Computing Surveys, 55*(10), 1-31. https://doi.org/10.1145/3559540.

Brownlee, J. (2019). *Generative adversarial networks with python*. https://www.scribd.com/document/473922459/Jason-Brownlee-Generative-Adversarial-Networks-with-Python-2020-pdf

Chawla, N., Bowyer, K., Hall, L. & Kegelmeyer, W. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research, 16*, 321-357.

Goodfellow, L., Pougel-Abadie, J., Mirza,M., Bing X., Warde-Farley, D., Ozair, S., Courville, A. & Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM, 63*(11), 139-144. https://doi.org/10.1145/3422622.

He, H., & Ma, Y. (2013). *Imbalanced learning*. John Wiley & Sons.

Jakub, L., & Vladimir, B. (2019). *GANs in action*. Manning.

Japkowicz, N. & Stephen, S. (2002). The class imbalance problem: A systematic study. *Intelligent Data Analysis, 6*(5), 203-231.

Jonathan, B., Putra, P. H. & Ruldeviyani, Y. (2020). Observation imbalanced data text to predict users selling products on female daily with SMOTE, Tomek, and SMOTE-Tomek. *IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT)* (pp. 81-85). 10.1109/IAICT50021.2020.9172033.

KEEL-dataset. (2023). *Imbalanced data sets*. https://sci2s.ugr.es/keel/imbalanced.php.

Li, X., Metsis, V., Wang, H., Hee, A., & Ngu, H. (2022). *TTS-GAN: A transformer-based time-series generative adversarial network*. AIME 2022. Springer.

Maniyar, H., Budihal, S. V. & Siddamal, S. V. (2022). Persons facial image synthesis from audio with generative adversarial networks. *ECTI-CIT Transactions, 16*(2), 135-141.

Maureen, L. C., Lauron, & Jaderick, P. P. (2016). *Improved sampling techniques for learning an imbalanced data set*. ArXiv abs/1601.04756.

Theobald, O. (2021). *Machine learning for absolute beginners: A plain English introduction* (3rd ed). Independently published.

Sridhar, S. & Sanagavarapu, S. (2021). Handling data imbalance in predictive maintenance for machines using SMOTE-based oversampling. *13th International Conference on Computational Intelligence and Communication Networks (CICN)* (pp. 44-49). 10.1109/CICN51697.2021.9574668.

Strelcenia, E. & Prakoonwit, S. (2022). Comparative analysis of machine learning algorithms using GANs through credit card fraud detection. International Conference on Computing, Networking, *Telecommunications & Engineering Sciences Applications (CoNTESA)* (pp. 1-5). 10.1109/CoNTESA57046.2022.10011268.

Weiss, G. M. (2013). *Foundations of imbalanced learning, imbalanced learning: Foundations, algorithms, and applications*. John Wiley & Sons.