วิทยานิพนธ์

อี-สตรีม: เทคนิคการแบ่งกลุ่มกระแสข้อมูลเชิงวิวัฒนาการ

E-STREAM: EVOLUTION-BASED TECHNIQUE FOR STREAM CLUSTERING

นายคมกริช อุดมมณีธนกิจ

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

พ.ศ. 2550

ใบรับรองวิทยานิพนธ์

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

วิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมคอมพิวเตอร์)

ปริญญา

วิศวกรรมคอมพิวเตอร์	วิศวกรรมคอมพิวเตอร์
สาขา	**ภาควิชา**

เรื่อง อี-สตรีม: เทคนิคการแบ่งกลุ่มกระแสข้อมูลเชิงวิวัฒนาการ

E-Stream: Evolution-based Technique for Stream Clustering

นามผู้วิจัย นายคมกริช อุดมมณีธนกิจ

ได้พิจารณาเห็นชอบโดย

ประธานกรรมการ .....

(รองศาสตราจารย์กฤษณะ ไวยมัย, Ph.D.)

กรรมการ .....

(ผู้ช่วยศาสตราจารย์จิตร์ทัศน์ ฝักเจริญผล, Ph.D.)

กรรมการ .....

(อาจารย์ยอดเยี่ยม ทิพย์สุวรรณ์, Ph.D.)

หัวหน้าภาควิชา .....

(ผู้ช่วยศาสตราจารย์เขมะฑัต วิภาตะวนิช, Ph.D.)

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์รับรองแล้ว

.....

(รองศาสตราจารย์วินัย อาจคงหาญ, M.A.)

คณบดีบัณฑิตวิทยาลัย

วันที่ ..... **เดือน** ..... **พ.ศ.** .....

วิทยานิพนธ์

เรื่อง

อี-สตรีม: เทคนิคการแบ่งกลุ่มกระแสข้อมูลเชิงวิวัฒนาการ

E-Stream: Evolution-based Technique for Stream Clustering

โดย

นายคมกริช อุดมมณีธนกิจ

เสนอ

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์
เพื่อความสมบูรณ์แห่งปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมคอมพิวเตอร์)
พ.ศ. 2550

คมกริช อุดมมณีธนกิจ 2550: อี-สตรีม: เทคนิคการแบ่งกลุ่มกระแสข้อมูลเชิงวิวัฒนาการ ปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมคอมพิวเตอร์) สาขาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ ประธานกรรมการที่ปรึกษา: รองศาสตราจารย์กฤษณะ ไวยมัย, Ph.D. 62 หน้า

การแบ่งกลุ่มกระแสข้อมูล เป็นเทคนิคการวิเคราะห์ข้อมูลซึ่งมีลักษณะเป็นกระแส มีประโยชน์ในการวิเคราะห์ข้อมูลที่มีการเปลี่ยนแปลงตามเวลา อีกทั้งสามารถติดตามผลลัพธ์ของกลุ่มได้ตลอดเวลา ดังนั้นงานวิจัยนี้จึงได้เสนอ เทคนิคใหม่ในการแบ่งกลุ่มกระแสข้อมูล E-Stream (Evolution-based Stream Clustering) ซึ่งประกอบด้วยโครงสร้างการทำงานใหม่ซึ่งสามารถรองรับการเปลี่ยนแปลงพฤติกรรมต่างๆ ของกลุ่มข้อมูลได้ โดยแบ่งลักษณะการเปลี่ยนแปลงออกเป็น 5 ประเภท คือ การเกิดขึ้นของกลุ่มข้อมูล การหายไปของกลุ่มข้อมูล การเลื่อนที่ของกลุ่มข้อมูล การรวมตัวกันของกลุ่มข้อมูล และการแยกตัวของกลุ่มข้อมูลได้ นอกจากนี้ยังได้เสนอการเก็บตัวแทนของกลุ่มข้อมูล และการหาค่าระยะห่างที่เหมาะสมกับโครงสร้างการทำงานนี้ด้วย จากผลการวิจัยพบว่า สำหรับข้อมูลสังเคราะห์ที่มีการเปลี่ยนแปลงอยู่ใน 5 ประเภทดังกล่าว เทคนิคที่เสนอสามารถให้ผลลัพธ์ที่มีคุณภาพมากขึ้น เมื่อเทียบกับเทคนิคที่มีอยู่

_____ / ___ / ___

ลายมือชื่อนิสิต ลายมือชื่อประธานกรรมการ

Komkrit Udommanetanakit 2007: E-Stream: Evolution-based Technique for Stream Clustering. Master of Engineering (Computer Engineering), Major Field: Computer Engineering, Department of Computer Engineering. Thesis Advisor: Associate Professor Kitsana Waiyamai, Ph.D. 62 pages.

Stream clustering is a technique that performs cluster analysis over data stream and able to monitor the results in real time. In this research, we propose a new stream clustering technique called E-Stream (Evolution-based Stream Clustering) which can support 5 evolutions of data stream. These evolutions are appearance, disappearance, self evolution, merge and split. Also propose the suitable cluster representation and distance function for this framework. For the dataset which has these evolutions, the experimental results show that our technique yields better cluster quality than the existing techniques.

_____ _____ ___ / ___ / ___

Student's signature Thesis Advisor's signature

กิติกรรมประกาศ

ขอกราบขอบพระคุณ รศ. ดร. กฤษณะ ไวยมัย อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่คอยชี้แนะให้คำปรึกษา อบรมสั่งสอน และสนับสนุนข้าพเจ้าเสมอมา คำชี้แนะของอาจารย์ทำให้แนวทางของงานวิจัยชิ้นนี้ชัดเจนขึ้น จนกระทั่งสำเร็จลุล่วงมาได้

ขอขอบพระคุณ อาจารย์ ธนาวินทร์ รักธรรมานนท์ ที่คอยช่วยเสนอแนวคิด ช่วยแก้ปัญหา ทำให้งานวิจัยชิ้นนี้สมบูรณ์ขึ้นมาได้

ขอขอบคุณพี่ ธนภัทร ฆังคะจิตร ที่คอยให้คำชี้แนะในฐานะรุ่นพี่ และตักเตือนข้าพเจ้าด้วยความหวังดี

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ในแลบ DAKDL ที่ทำให้มีบรรยากาศดีๆในแลบ คอยรับฟังปัญหาและคอยช่วยเหลือกันอยู่เสมอ

สุดท้ายนี้ขอขอบพระคุณ คุณพ่อ คุณแม่ ที่มีความเชื่อมั่น สนับสนุน และห่วงใยข้าพเจ้าตลอดมา

คมกริช อุดมมณีธนกิจ

กุมภาพันธ์ 2550

สารบัญ

หน้า

สารบัญตาราง

สารบัญภาพ

สารบัญภาพ (ต่อ)

ภาพที่ **หน้า**

ภาพผนวกที่

คำอธิบายและสัญลักษณ์คำย่อ

CF	=	Clustering Feature
E-Stream	=	Evolution-based Stream Clustering
FC	=	Fading Cluster Structure
FCH	=	Fading Cluster Structure with Histogram

อี-สตรีม: เทคนิคการแบ่งกลุ่มกระแสข้อมูลเชิงวิวัฒนาการ

E-Stream: Evolution-based Technique for Stream Clustering

คำนำ

ในปัจจุบันความสามารถในการวิเคราะห์ข้อมูลที่มีอยู่นั้น ถือเป็นสิ่งที่มีประโยชน์และความสำคัญอย่างยิ่ง จนมีศาสตร์แขนงใหม่เกี่ยวกับการสืบค้นความรู้หรือรูปแบบที่น่าสนใจจากข้อมูล (Data Mining) เกิดขึ้น เทคนิคซึ่งใช้ในการสืบค้นความรู้ที่จะกล่าวถึงในที่นี้คือ เทคนิคการแบ่งกลุ่มข้อมูล (Data Clustering) ซึ่งถือเป็นเทคนิคหนึ่งที่ได้รับความสนใจ มีประโยชน์มากในการวิเคราะห์พฤติกรรมและการจับกลุ่มกันของข้อมูล โดยไม่ต้องอาศัยตัวอย่างในการเรียนรู้ (Unsupervised Learning) เทคนิคนี้จะทำการแบ่งข้อมูลออกเป็นกลุ่มย่อยๆ แต่ละกลุ่มเรียกว่าคลัสเตอร์ (Cluster) ข้อมูลที่มีลักษณะคล้ายคลึงกันจะถูกจัดให้อยู่ในกลุ่มเดียวกัน และข้อมูลที่มีลักษณะแตกต่างกันจะถูกจัดไว้ต่างกลุ่มกัน เป็นเทคนิคที่ใช้สำหรับพิจารณาลักษณะภายในของข้อมูล

ข้อมูลที่ต้องการวิเคราะห์ มีหลายประเภทที่อยู่ในรูปแบบของกระแสข้อมูล (Data Stream) คือ ข้อมูลที่เกิดขึ้นอย่างต่อเนื่องตลอดเวลา อาจพิจารณาได้ว่ามีจำนวนข้อมูลเป็นอนันต์ หรือเป็นกระแสที่ไม่สิ้นสุด ตัวอย่างของกระแสข้อมูล เช่น ข้อมูลการเชื่อมต่อระบบเครือข่าย ข้อมูลการซื้อขายหุ้นในตลาดหุ้น ข้อมูลการใช้เครดิตการ์ด เป็นต้น ข้อมูลเหล่านี้เกิดขึ้นอย่างรวดเร็วและมีจำนวนมหาศาล หลายๆ แอพพลิเคชั่น ซึ่งเดิมทีพิจารณาเป็นข้อมูลซึ่งมีจำนวนคงที่อยู่ในฐานข้อมูล แต่ความเป็นจริงนั้นข้อมูลเกิดขึ้นใหม่อยู่ตลอดเวลา อีกทั้งพฤติกรรมยังมีการเปลี่ยนแปลงพฤติกรรมได้ เป็นเหตุให้นักวิเคราะห์มีความต้องการที่จะวิเคราะห์ในลักษณะของกระแสข้อมูลมากกว่าข้อมูลที่คงที่ เช่น การวิเคราะห์การเปลี่ยนแปลงของข้อมูลเปรียบเทียบระหว่างช่วงเวลา การตรวจจับข้อมูลผิดปกติ (Anomaly Detection) แบบทันเวลา สิ่งที่ต้องการเหล่านี้สามารถทำได้ด้วยการใช้เทคนิคการแบ่งกลุ่มกระแสข้อมูล (Stream Clustering)

เทคนิคการแบ่งกลุ่มข้อมูลโดยปกติจะเก็บข้อมูลทั้งหมดในหน่วยความจำเพื่อใช้ประมวลผล ข้อมูลทุกตัวสามารถนำกลับมาวิเคราะห์ใหม่ได้เมื่อต้องการ แต่สำหรับกระแสข้อมูลเนื่องจากข้อมูลมีจำนวนเป็นอนันต์ ไม่อาจเก็บรายละเอียดของข้อมูลทุกตัวลงในหน่วยความจำ ข้อมูลแต่ละตัวสามารถใช้ประมวลผลได้เพียงชั่วคราว ทำให้ต้องประมวลผลในเวลาจำกัด และไม่

สามารถนำกลับมาวิเคราะห์ซ้ำได้ ดังนั้นเทคนิคการแบ่งกลุ่มกระแสข้อมูลจึงต้องมีความรวดเร็วในการประมวลผลข้อมูลที่เกิดขึ้นอย่างต่อเนื่อง และใช้หน่วยความจำได้อย่างมีประสิทธิภาพ

เทคนิคการแบ่งกลุ่มกระแสข้อมูลโดยทั่วไปใช้การเก็บตัวแทนของคลัสเตอร์ (Cluster Representation) แทนการเก็บข้อมูลทั้งหมด เพื่อใช้ประมวลผลต่อไป ดังนั้นจึงสามารถกำจัดปัญหาทางด้านหน่วยความจำได้ แต่หลายเทคนิคไม่ได้คำนึงถึงการเปลี่ยนแปลงพฤติกรรมของข้อมูลตามเวลา และบางเทคนิคยังทำได้ไม่ดี ในงานวิจัยนี้เราจึงสนใจปัญหาการแบ่งกลุ่มกระแสข้อมูลซึ่งสามารถปรับปรุงตัวเองได้ เมื่อพฤติกรรมของข้อมูลเปลี่ยนไป เพื่อให้ได้ลักษณะของกลุ่มข้อมูลที่ถูกต้องและทันสมัยตลอดเวลา

วัตถุประสงค์

พัฒนาเทคนิคการแบ่งกลุ่มกระแสข้อมูล เพื่อเพิ่มคุณภาพของกลุ่มข้อมูลผลลัพธ์ โดยการ

1 พัฒนาโครงสร้างการทำงานให้สามารถรองรับรูปแบบการเปลี่ยนแปลงพฤติกรรมของข้อมูลได้มากขึ้น

3 พัฒนาการหาตัวแทนของกลุ่มข้อมูลที่เหมาะสม

2 พัฒนาการหาค่าระยะห่างที่เหมาะสม

การตรวจเอกสาร

ในหัวข้อนี้จะกล่าวถึงการแบ่งกลุ่มข้อมูล การแบ่งกลุ่มกระแสข้อมูล งานวิจัยต่างๆ ที่เกี่ยวข้อง สถิติพื้นฐานที่ควรทราบ และวิธีการวัดคุณภาพผลของการแบ่งกลุ่มข้อมูล

การสืบค้นความรู้จากฐานข้อมูลขนาดใหญ่

ปัจจุบันการใช้งานคอมพิวเตอร์มีแนวโน้มเพิ่มขึ้นอย่างรวดเร็ว ทุกบริษัทหรือองค์กรต่างๆ ล้วนใช้คอมพิวเตอร์เพื่อช่วยประมวลผลและจัดเก็บข้อมูลแทบทั้งสิ้น ข้อมูลต่างๆที่ได้จากการทำงานในแต่ละวันจะถูกจัดเก็บไว้ในรูปของข้อมูลดิบ ซึ่งมีปริมาณมหาศาล ดังนั้นจึงมีแนวคิดการนำข้อมูลที่ถูกจัดเก็บไว้นี้มาสกัดเอาความรู้หรือข้อมูลที่มีประโยชน์มาจากข้อมูลปริมาณมหาศาลนี้ จนเกิดเป็นศาสตร์แขนงใหม่ว่าด้วยการสืบค้นความรู้หรือรูปแบบที่น่าสนใจจากฐานข้อมูลขนาดใหญ่ เรียกว่า เดต้าไมน์นิ่ง (Data Mining)

ศาสตร์แห่งเดต้าไมน์นิ่งนี้ ถูกศึกษาและพัฒนามาอย่างต่อเนื่อง แตกออกเป็นแขนงย่อยต่างๆมากมาย ซึ่งสามารถแบ่งออกเป็น 3 ส่วนหลักๆ คือ การสืบค้นกฎความสัมพันธ์ (Association Rules Discovery) การจัดกลุ่มข้อมูลและการทำนาย (Data Classification and Prediction) และการแบ่งกลุ่มข้อมูล (Data Clustering) โดยเทคนิคที่เราสนใจและจะกล่าวถึงต่อไป คือ การแบ่งกลุ่มข้อมูล

การแบ่งกลุ่มข้อมูล

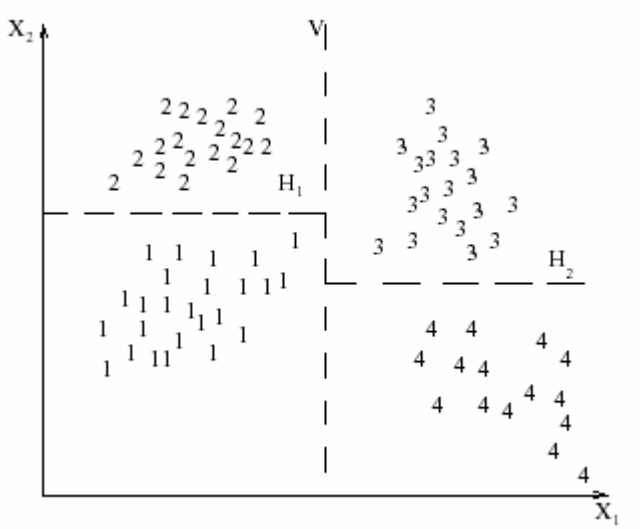
การแบ่งกลุ่มข้อมูล (Data Clustering) เป็นแขนงหนึ่งของเดต้าไมน์นิ่ง ซึ่งไม่ต้องอาศัยตัวอย่างในการเรียนรู้ (Unsupervised Learning) เทคนิคนี้จะทำการแบ่งข้อมูลทั้งหมดจะถูกแบ่งออกเป็นกลุ่มย่อยๆ เรียกว่าคลัสเตอร์ (Cluster) โดยใช้ความคล้ายคลึงกันของข้อมูลเป็นเกณฑ์ ข้อมูลที่มีลักษณะคล้ายคลึงกันจะถูกจัดให้อยู่ในกลุ่มเดียวกัน และข้อมูลที่มีลักษณะแตกต่างกันจะถูกจัดไว้ต่างกลุ่มกัน มีประโยชน์มากในการวิเคราะห์ความสัมพันธ์กันของชุดข้อมูล สามารถนำไปประยุกต์ใช้ได้หลากหลายกับงานประเภทต่างๆ เช่น ใช้ช่วยฝ่ายการตลาดในการแบ่งกลุ่มลูกค้าตามพฤติกรรมการซื้อเพื่อที่จะพัฒนาการขายให้ตรงเป้าหมายหรือตรงตามความต้องการในแต่ละกลุ่ม

ใช้ในการแบ่งเอกสารให้เป็นหมวดหมู่ ใช้ในการแบ่งกลุ่มเพื่อตรวจจับข้อมูลผิดปกติ (Outlier Detection)

เทคนิคการแบ่งกลุ่มถูกพัฒนาขึ้นมาเป็นจำนวนมาก แต่ละเทคนิคมีลักษณะเฉพาะที่แตกต่างกัน เทคนิคที่จะกล่าวถึงในที่นี้เป็นเทคนิคพื้นฐานที่มีการนำไปใช้อย่างแพร่หลาย คือ การแบ่งกลุ่มแบบมีลำดับชั้น (Hierarchical Based) การแบ่งกลุ่มแบบพาร์ติชั่น (Partition Based) การแบ่งกลุ่มแบบใช้ความหนาแน่น (Density Based) การแบ่งกลุ่มโดยใช้กริด (Grid Based) และการแบ่งกลุ่มเพื่อตรวจจับข้อมูลผิดปกติ

1. วิธีการแบ่งกลุ่มแบบพาร์ติชัน (Partitioning methods)

วิธีนี้จะแบ่งกลุ่มข้อมูลให้มีจำนวนเท่ากับจำนวนกลุ่มที่ต้องการตั้งแต่ในขั้นตอนแรก โดยที่ทุกกลุ่มข้อมูลไม่มีการซ้อนทับกัน โดยทั่วไปจะปรับให้ได้ค่าฟังก์ชันวัดคุณภาพ (Criterion Function) สูงสุด ซึ่งการวัดคำนวณหาผลคำตอบที่ดีที่สุดโดยวัดและหาค่าจากทุกความเป็นไปได้นั้นใช้เวลาและการคำนวณมหาศาล โดยทั่วไปแล้วนิยมสุ่มการทำงานหลายๆครั้งแล้วเลือกเอาผลที่ดีที่สุด ข้อดีของวิธีนี้คือ เป็นวิธีที่ง่าย และรวดเร็ว ส่วนข้อเสียคือ ต้องกำหนดจำนวนกลุ่มที่ต้องการแบ่ง



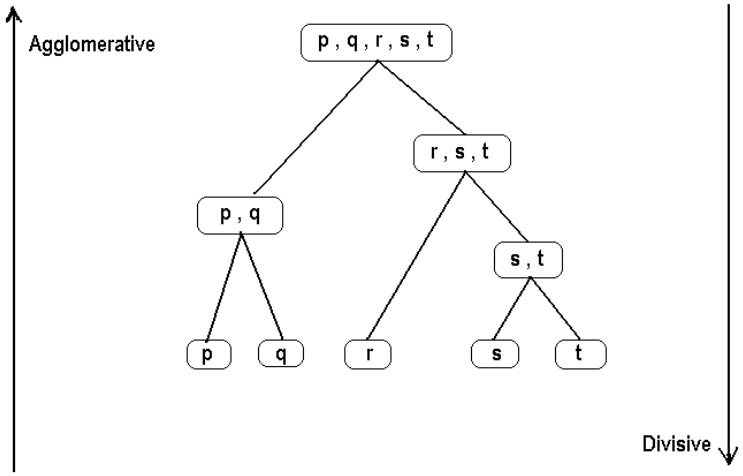
ภาพที่ 1 แสดงลักษณะการแบ่งกลุ่มแบบพาร์ทิชัน

2. วิธีการแบ่งกลุ่มแบบลำดับชั้น (Hierarchical methods)

วิธีนี้มีลักษณะการทำงานคล้ายโครงสร้างต้นไม้ ซึ่งแบ่งออกเป็นสองประเภทตามลำดับการทำงานคือ การแบ่งกลุ่มแบบล่างขึ้นบน (Agglomerative Hierarchical Clustering) และการแบ่งกลุ่มจากบนลงล่าง (Divisive Hierarchical Clustering)

การแบ่งกลุ่มแบบล่างขึ้นบน(Agglomerative Hierarchical Clustering) เริ่มด้วยการให้ข้อมูลทุกตัวอยู่ต่างกลุ่มกัน ในแต่ละลำดับชั้นจะทำการรวมกลุ่มที่มีความคล้ายคลึงกันมากที่สุดเข้าด้วยกันเกิดเป็นกลุ่มข้อมูลที่ใหญ่ขึ้นเรื่อยๆ

การแบ่งกลุ่มจากบนลงล่าง(Divisive Hierarchical Clustering) มีแนวทางสลับกับแบบล่างขึ้นบน โดยเริ่มจากการให้ข้อมูลทุกตัวอยู่ในกลุ่มเดียวกัน จากนั้นจึงหาคู่ของข้อมูลที่มีความแตกต่างกันมากที่สุดภายในกลุ่ม แล้วจึงแยกข้อมูลทั้งคู่ให้อยู่ต่างกลุ่มกัน ข้อมูลที่เหลือจะถูกจัดให้อยู่ในกลุ่มที่ให้ความคล้ายคลึงมากที่สุด กลุ่มข้อมูลที่ได้ในแต่ละลำดับชั้นจะมีขนาดเล็กลงเรื่อยๆ

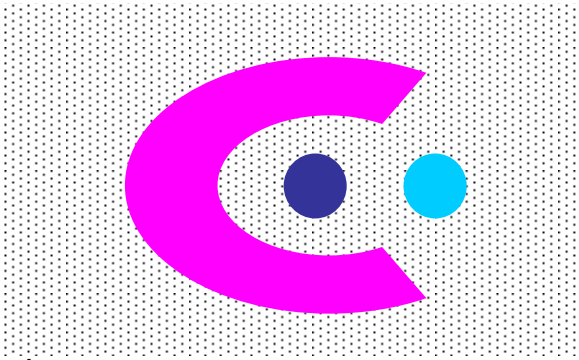


ภาพที่ 2 แสดงลักษณะการแบ่งกลุ่มแบบลำดับชั้น

ข้อดีของการแบ่งกลุ่มด้วยแบบลำดับชั้นคือ ไม่ต้องกำหนดจำนวนกลุ่มที่ตั้งแต่เริ่มทำงาน เราสามารถหยุดการทำงานได้เมื่อเห็นว่าได้คุณภาพ หรือจำนวนกลุ่มตามที่ต้องการแล้ว และยังให้ความถูกต้องสูงเนื่องจากในแยกกลุ่มหรือรวมกลุ่มจะต้องมีการคำนวณค่าความคล้ายคลึงระหว่างกลุ่มย่อยทุกคู่ที่เป็นไปได้ แล้วจึงเลือกคู่ของกลุ่มที่ดีที่สุดในการรวมหรือแยกกลุ่ม แต่มีผลทำให้ใช้เวลาในการทำงานสูงขึ้นตามไปด้วย

3. วิธีการแบ่งกลุ่มแบบใช้ความหนาแน่น (Density-based methods)

วิธีนี้แบ่งกลุ่มตามความหนาแน่นและความต่อเนื่องของข้อมูล พื้นที่ที่ข้อมูลมีความหนาแน่นและต่อเนื่องกันจะถูกเชื่อมต่อกันเป็นพื้นที่ที่ใหญ่ขึ้น เนื่องจากใช้วิธีการเชื่อมต่อกันทำให้รูปร่างของกลุ่มสามารถขยายได้ในทุกทิศทาง และสามารถเกิดเป็นรูปร่างใดๆ ได้ ข้อมูลที่ไม่อยู่ในส่วนหนาแน่นจะถูกพิจารณาเป็นข้อมูลผิดปกติ (Outlier) และจะไม่ถูกนำมาพิจารณาในการแบ่งกลุ่ม ข้อดีของวิธีนี้คือ รูปร่างของกลุ่มไม่จำเป็นต้องเป็นทรงกลม (เนื่องจากวิธีแบ่งกลุ่มแบบพาติชันและแบบลำดับชั้นโดยทั่วไปใช้ค่าระยะห่างจากจุดศูนย์กลาง จะมีขอบเขตของกลุ่มอยู่ที่รัศมีค่าหนึ่งจากจุดศูนย์กลาง ทำให้ได้กลุ่มที่แบ่งได้มีลักษณะเหมือนเป็นทรงกลม) และยังสามารถจัดการกับข้อมูลผิดปกติได้ดีอีกด้วย (ข้อมูลผิดปกติ อาจเกิดจากความผิดพลาดของระบบที่สร้างข้อมูล หรืออาจเกิดจากพฤติกรรมที่ผิดปกติของผู้ใช้ ซึ่งถ้าไม่มีวิธีการจัดการกับข้อมูลเหล่านี้จะทำให้กลุ่มที่แบ่งได้เกิดความผิดเพี้ยน)



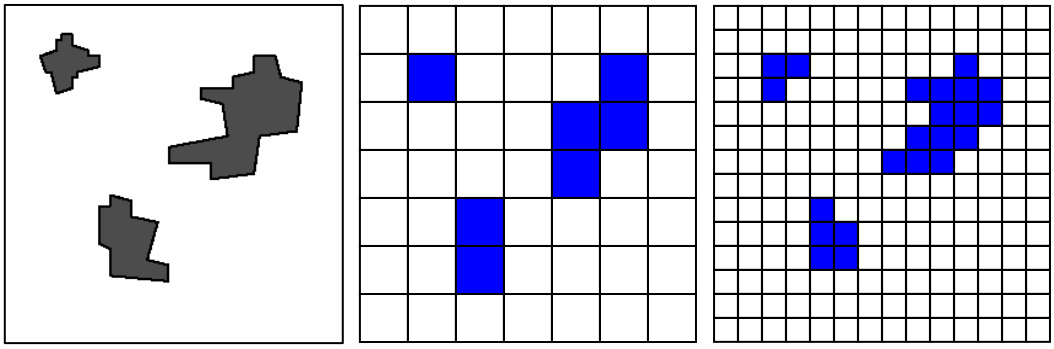
ภาพที่ 3 แสดงลักษณะการแบ่งกลุ่มแบบใช้ความหนาแน่น

จากภาพที่ 3 จะเห็นว่ากลุ่มที่แบ่งสามารถมีได้หลายรูปร่างตามพื้นที่ที่มีการเชื่อมต่อความหนาแน่น และข้อมูลที่มีอยู่กระจัดกระจายจะถูกพิจารณาเป็นข้อมูลผิดปกติ

4. วิธีการแบ่งกลุ่มโดยใช้กริด (Grid-based methods)

วิธีนี้จะทำการแบ่งข้อมูลในแต่ละมิติออกเป็นช่วงๆ การแบ่งนี้ทำให้เกิดช่องเล็กๆมากมาย เราเรียกแต่ละช่องที่แบ่งได้นี้ว่า เซลล์ โดยที่ในแต่ละเซลล์จะเก็บจำนวนข้อมูลที่อยู่ภายในเซลล์นั้นไว้ การแบ่งกลุ่มทำได้โดยการเชื่อมต่อเซลล์มีจำนวนข้อมูล (อาจเรียกว่าค่าสนับสนุน) มากเข้า

ด้วยกัน ในการแบ่งกลุ่มจะพิจารณาในระดับเซลล์ซึ่งหยาบกว่าระดับข้อมูล เซลล์ที่จำนวนข้อมูลไม่ผ่านเกณฑ์จะไม่ถูกนำมาพิจารณา ดังภาพที่ 4 จะเห็นว่าถ้าแบ่งข้อมูลหยาบเกินไปจะทำให้มีข้อมูลบางส่วนหายไป และถ้าแบ่งข้อมูลละเอียดขึ้นจะได้ผลลัพธ์ที่ละเอียดมากขึ้น โดยภาพรวมแล้วข้อดีของวิธีนี้จะเหมือนกับวิธีใช้ความหนาแน่น แต่มีความเร็วในการทำงานมากกว่า แต่มีข้อเสียคือ ใช้หน่วยความจำจำนวนมากในการเก็บข้อมูลแต่ละเซลล์ ถ้าแบ่งเซลล์ละเอียดมากจะเปลืองหน่วยความจำมาก แต่ถ้าแบ่งเซลล์หยาบจะทำให้สูญเสียรายละเอียดของข้อมูล



ก) ข) ค)

ภาพที่ 4 ก) ข้อมูลที่แบ่งกลุ่ม ข) คลัสเตอร์ที่ได้ ค) คลัสเตอร์ที่ได้เมื่อความละเอียดมากขึ้น

การแบ่งกลุ่มกระแสข้อมูล

กระแสข้อมูล เกิดจากการเกิดขึ้นอย่างต่อเนื่องของข้อมูล ซึ่งมีจำนวนไม่จำกัด และสามารถเปลี่ยนแปลงพฤติกรรมได้ตลอดเวลา (Gaber *et al.*, 2005) มีหลายแอพพลิเคชั่นที่ให้กำเนิดข้อมูลในรูปแบบของกระแส เช่น ข้อมูลการเชื่อมต่อระบบเครือข่าย ข้อมูลการซื้อขายหุ้นในตลาดหุ้น ข้อมูลการใช้เครดิตการ์ด เป็นต้น เราสามารถให้คำจำกัดความของ กระแสข้อมูล ได้ดังนี้

คำจำกัดความ 1 กระแสข้อมูล คือ ชุดของข้อมูลจำนวนอนันต์ $\overline{X}_1 ... \overline{X}_k ...$ ซึ่งเกิดขึ้นที่เวลา $T_1 ... T_k ...$ ตามลำดับ โดยที่ข้อมูลแต่ละตัวประกอบด้วย d มิติ สามารถเขียนได้เป็น $\overline{X}_i = (x_i^1 ... x_i^d)$

การแบ่งกลุ่มกระแสข้อมูล เกิดจากความต้องการวิเคราะห์ข้อมูลที่มีลักษณะเป็นกระแส ซึ่งเกิดขึ้นในหลายแอพพลิเคชั่น การใช้เทคนิคการแบ่งกลุ่มข้อมูลทั่วไปไม่สามารถรองรับข้อมูลจำนวนมหาศาลที่เกิดขึ้นอยู่ตลอดเวลานี้ได้ จากที่กล่าวไว้เบื้องต้น เทคนิคการแบ่งกลุ่มข้อมูลโดยทั่วไปจะเก็บข้อมูลทั้งหมดในหน่วยความจำ ทำให้ข้อมูลทุกตัวสามารถนำกลับมาวิเคราะห์ซ้ำ

ได้ แต่สำหรับกระแสข้อมูล ข้อมูลหนึ่งๆ สามารถเก็บในหน่วยความจำได้เพียงชั่วคราว ดังนั้น วิธีการแบ่งกลุ่มกระแสข้อมูลจึงต้อง ต้องใช้การอ่านผ่านข้อมูลเพียงรอบเดียว และมีความรวดเร็วในการประมวลผลข้อมูลที่เกิดขึ้นอย่างต่อเนื่อง และใช้หน่วยความจำได้อย่างมีประสิทธิภาพอีกด้วย (Barbara, 2000)

งานวิจัยเกี่ยวกับการแบ่งกลุ่มกระแสข้อมูลได้ถูกพัฒนามาอย่างต่อเนื่องและหลากหลาย ในที่นี้ เราสนใจเฉพาะงานวิจัยที่เกี่ยวกับเทคนิคการแบ่งกลุ่มกระแสข้อมูลโดยทั่วไป ซึ่งพิจารณาเฉพาะคุณสมบัติของข้อมูลเชิงปริมาณ (Quantitative Attribute) ซึ่งจะอธิบายรายละเอียดของแต่ละงานต่อไป

1. BIRCH

BIRCH (Zhang *et al.*, 2000) เป็นเทคนิคที่ถูกเสนอในการจัดการกับข้อมูลขนาดใหญ่ และสามารถจัดการกับข้อมูลที่เพิ่มเติมได้ด้วย โดย BIRCH ใช้การเก็บตัวแทนของคลัสเตอร์ในรูปของ Clustering Feature (CF) ประกอบด้วย

N คือ จำนวนข้อมูลภายในกลุ่ม

LS คือ ค่าผลรวมของข้อมูลทุกตัวภายในกลุ่ม แยกแต่ละมิติ

$$LS = \sum_{i=1}^{N} \left(\overline{X_i} \right) \qquad (1)$$

SS คือ ค่าผลรวมของข้อมูลทุกตัวภายในกลุ่มยกกำลังสอง แยกแต่ละมิติ

$$SS = \sum_{i=1}^{N} \left(\overline{X_i}^2 \right) \qquad (2)$$

CF เป็นค่าคุณสมบัติต่างๆที่ใช้หาตัวแทนของกลุ่มข้อมูล ซึ่งใช้ในการหาจุดศูนย์กลางของกลุ่ม และค่าเบี่ยงเบนมาตรฐานได้ และสามารถจัดเก็บแบบเพิ่มเติมภายหลัง (Incremental) ได้อีกด้วย ทำให้เราสามารถหาค่าจุดศูนย์กลาง และค่าเบี่ยงเบนมาตรฐานของกลุ่มข้อมูลใดๆ โดยการอ่านผ่านข้อมูลเพียงรอบเดียว

BIRCH จะเก็บข้อมูลทั้งหมดที่อ่านในรูปแบบของ CF ซึ่งถือเป็นคลัสเตอร์ย่อย โดยมีเงื่อนไขให้ภายใน CF เดียวกันข้อมูลมีค่าความห่างกันไม่เกินค่าเกณฑ์ที่กำหนด CF ทั้งหมดจะถูกจัดลำดับและจัดเก็บลงในโครงสร้างแบบต้นไม้ชื่อ CF-Tree เมื่อมีข้อมูลใหม่เข้ามาจะทำการหากลุ่มคลัสเตอร์ย่อยที่ใกล้เคียงที่สุดโดยการไล่หาไปตาม CF-Tree ข้อมูลจะถูกรวมในคลัสเตอร์ย่อยที่เจอ ถ้าคลัสเตอร์ย่อยนั้นห่างจากข้อมูลตัวใหม่ไม่มากกว่าค่าที่กำหนด แต่ถ้าไม่สามารถรวมได้ก็จะสร้างเป็นคลัสเตอร์ย่อยตัวใหม่ BIRCH สามารถแบ่งกลุ่มได้โดยการอ่านข้อมูลจากข้อมูลเพียงรอบเดียวและเก็บข้อมูลที่อ่านทั้งหมดในรูปของ CF เป็นเทคนิคหนึ่งที่สามารถนำมาประยุกต์ใช้กับกระแสข้อมูลได้ แต่การนำ BIRCH มาใช้กับกระแสข้อมูลโดยตรงนั้นให้ผลที่ยังไม่น่าพอใจ

2. STREAM

STREAM (Guha *et al.*, 2000) เป็นเทคนิคที่ใช้วิธีการแบ่งข้อมูลออกเป็นส่วนย่อยๆ ก่อนจากนั้นจะมองปัญหาการแบ่งกลุ่มให้อยู่ในรูปของปัญหาการหาตำแหน่งโรงงาน (Facility Location) ซึ่งเป็นปัญหาทางอัลกอริทึม โดยจุดมุ่งหมายของปัญหาคือจะหาตำแหน่งของโรงงานที่ทำให้เสียค่าใช้จ่ายน้อยที่สุด เมื่อได้แบ่งกลุ่มในข้อมูลแต่ละส่วนแล้ว จากนั้นจึงใช้เฉพาะตัวแทนคลัสเตอร์ที่เก็บไว้ คือ จุดศูนย์กลางและจำนวนข้อมูลภายในกลุ่ม ของคลัสเตอร์ที่ได้จากทุกส่วนมาเพื่อแบ่งกลุ่มในลำดับชั้นที่สูงขึ้นจนได้จำนวนกลุ่มที่ต้องการ

3. CluStream

CluStream (Aggarwal *et al.*, 2003) เป็นเทคนิคที่ใช้แนวคิดในการเก็บข้อมูลในรูปแบบของคลัสเตอร์ย่อย ซึ่งใช้หลักการเก็บตัวแทนของคลัสเตอร์เช่นเดียวกับ CF ใน BIRCH โดยแบ่งการทำงานออกเป็น 2 ส่วน ส่วนแรกเป็นส่วนที่รับข้อมูลจากสตรีมเข้ามาและเก็บไว้ในรูปแบบของคลัสเตอร์ย่อยจำนวนมาก ซึ่งชุดของคลัสเตอร์ย่อยที่เวลาหนึ่งๆ (Cluster Snapshot) จะถูกเก็บเพื่อใช้ในการคลัสเตอร์ในช่วงเวลาที่ต้องการ โดยความถี่ในการเก็บชุดของคลัสเตอร์นั้นจะให้ความสำคัญกับเวลาที่ข้อมูลเข้ามาด้วย โดยจะให้ความสำคัญกับข้อมูลปัจจุบันมากกว่าข้อมูลที่ผ่านมานานแล้ว นั่นคือที่เวลาย้อนหลังมากๆ จะมีการลบชุดของคลัสเตอร์ที่บางเวลาออก เพื่อลดการใช้หน่วยความจำ และส่วนที่สอง เป็นส่วนของการนำคลัสเตอร์ย่อยจากช่วงเวลาใดๆ มาทำการคลัสเตอร์เพื่อหาคลัสเตอร์ที่เป็นผลลัพธ์

4. HPStream

HPStream (Aggarwal *et al.*, 2004) เป็นเทคนิคที่ถูกพัฒนาต่อมาจาก CluStream โดยใช้รูปแบบการเก็บตัวแทนของคลัสเตอร์คล้ายเดิม สิ่งที่แตกต่างคือ ในส่วนของการให้ความสำคัญกับเวลาที่เข้ามาของข้อมูล จะถูกกำหนดโดย ฟังก์ชันการเลือนหาย (Fading Function) ซึ่งเป็นฟังก์ชันที่มีค่าลดลงเมื่อเวลาผ่านไป โดยที่ข้อมูลแต่ละตัวจะถูกให้ความสำคัญ (Weight) ด้วยค่าฟังก์ชันนี้ โดยเรียกการเก็บตัวแทนคลัสเตอร์นี้ว่า FC (Fading Cluster Structure)

ให้ t เป็นเวลาปัจจุบัน $C = \{\overline{X}_1 ... \overline{X}_N\}$ เป็นสมาชิกของข้อมูลขนาด d มิติ ภายในคลัสเตอร์ ที่เกิดขึ้นที่เวลา $T_1 ... T_N$ ตามลำดับ จะได้ตัวแทนคลัสเตอร์ที่เกิดจาก C และ t เป็น

$$FC(C,t) = (\overline{FC1(C,t)}, \overline{FC2(C,t)}, W(t), \overline{H(C,t)}) \qquad (3)$$

โดยที่

$$\overline{FC1(C,t)} = (FC1^1 ... FC1^d) \qquad (4)$$

และ

$$\overline{FC2(C,t)} = (FC2^1 ... FC2^d) \qquad (5)$$

$FC1^j(t)$ เป็นผลรวมถ่วงน้ำหนักด้วยค่าการเลือนหายของข้อมูลทั้งหมดของคลัสเตอร์ในมิติที่ j

$$FC1^j(C,t) = \sum_{i=1}^{N} f(t - T_i) \cdot (x_i^j) \qquad (6)$$

$FC2^j(t)$ เป็นผลรวมถ่วงน้ำหนักด้วยค่าการเลือนหายของข้อมูลยกกำลังสองของข้อมูลทั้งหมดของคลัสเตอร์ในมิติที่ j

$$FC2^j(C,t) = \sum_{i=1}^{N} f(t - T_i) \cdot (x_i^j)^2 \qquad (7)$$

W(t) เป็นผลรวมของค่าการเลือนหายของข้อมูลทั้งหมดในคลัสเตอร์ ซึ่งเราเรียกค่านี้ว่า ค่าความสำคัญ

$$W(t) = \sum_{i=1}^{N} f(t - T_i) \qquad (8)$$

นอกจากนี้เทคนิคนี้ยังได้เสนอวิธีการโปรเจกต์มิติ (Dimension Projection) เพื่อหามิติที่เป็นตัวแทนของคลัสเตอร์นั้น และตัดมิติที่ไม่เกี่ยวข้องออกไป ด้วยแนวคิดที่ว่าในคลัสเตอร์ใดๆ ไม่จำเป็นต้องหาความเหมือนระหว่างข้อมูลกับคลัสเตอร์ใดๆจากทุกมิติ หรือทุกคุณสมบัติของข้อมูล แต่ใช้แค่บางตัวที่บ่งบอกถึงความเป็นคลัสเตอร์นั้นในการหาความเหมือน ซึ่งมีผลต่อความถูกต้องเมื่อข้อมูลมีมิติจำนวนมาก และยังมีผลทำให้คลัสเตอร์ผลลัพธ์ที่ได้ดีขึ้น เป็นเทคนิคหนึ่งที่ได้รับการยอมรับในปัจจุบัน

5. An Improvement of HPStream

Udommanetanakit *et al.* (2006) ได้เสนอเทคนิคซึ่งปรับปรุงการหาค่าระยะห่างต่อจาก HPStream โดยประยุกต์การหาค่าระยะห่างจาก DataBubbles (Breunig *et al.*, 2001) มาใช้กับข้อมูลหลายมิติ ซึ่งทำให้ผลลัพธ์ที่ได้จาก HPStream ดีขึ้น

6. Gaussian Mixture Models for Online Data Stream Clustering

Song and Wang (2005) ได้เสนอการประยุกต์ใช้โมเดล Gaussian Mixture Models (GMM) กับกระแสข้อมูล โดยจะเก็บข้อมูลเก่าไว้ในรูปแบบของ GMM ชุดหนึ่ง และพิจารณาชุดของข้อมูลที่เกิดขึ้นใหม่เป็น GMM อีกชุดหนึ่ง จากนั้นจึงเสนอวิธีการรวม GMM ทั้งสองชุดเข้าด้วยกัน

7. Intrusion Detection based on Clustering a Data Stream

Oh *et al.* (2005) ได้เสนอเทคนิคการแบ่งกลุ่มกระแสข้อมูลเพื่อตรวจจับการบุกรุก (Intrusion Detection) โดยเสนอว่าพฤติกรรมของข้อมูลสามารถมีการเปลี่ยนแปลงได้ คลัสเตอร์ที่มีอยู่สามารถรวมตัว หรือแตกตัวได้ โดยทดลองรวมสองคลัสเตอร์ที่พิจารณาเข้าด้วยกัน ถ้าค่า

เบี่ยงเบนมาตรฐานของคลัสเตอร์ที่เกิดจากการรวมมีค่าไม่เกินค่าที่กำหนดจึงทำการรวมเข้าด้วยกัน และพิจารณาการแตกตัวโดยพิจารณาค่าเบี่ยงเบนมาตรฐานของทุกคลัสเตอร์ ถ้าคลัสเตอร์ใดมีค่าเกินกว่าที่กำหนดจึงทำการแตกคลัสเตอร์นั้นออกเป็นสองส่วน โดยการแตกนั้นจะแตกส่วนที่ทำให้มีค่าเกินออกเท่านั้น เทคนิคนี้เก็บข้อมูลทั้งหมดลงในกริดสำหรับแต่ละมิติ และเก็บตัวแทนของคลัสเตอร์ในรูปของ จุดศูนย์กลาง ค่าความแปรปรวนและเซลล์ของกริดซึ่งเป็นสมาชิกของคลัสเตอร์นั้น

สถิติพื้นฐานที่ควรทราบ

ในงานวิจัยนี้ได้อ้างอิงค่าทางสถิติต่างๆ หลายประเภท ในหัวข้อนี้เราจะอธิบายถึงความหมายของคำต่างๆ ที่ใช้ เพื่อเป็นประโยชน์ในการอธิบายหัวข้อถัดไป

1. ค่าเบี่ยงเบนมาตรฐาน (Standard Deviation) คือ ค่าการกระจายตัวภายในของชุดข้อมูล ให้ x_i เป็นข้อมูลใดๆภายในกลุ่ม $\bar{x}$ เป็นค่าเฉลี่ยของข้อมูลทุกตัวภายในกลุ่ม และ N เป็นจำนวนข้อมูลภายในกลุ่ม เราสามารถหาค่าเบี่ยงเบนมาตรฐานของกลุ่มข้อมูลได้จากสมการ

$$SD = \sqrt{\frac{\sum_{i=1}^{N}\left(x_i - \bar{x}\right)^2}{N}} \qquad (9)$$

2. การทดสอบไคสแควร์ (Chi-Square Test) เป็นการทดสอบสมมติฐานว่า ค่าที่คาดหวังต่างกับค่าที่เกิดขึ้นจริง ของตัวแปรที่พิจารณาเหมือนหรือต่างกัน ด้วยระดับความเชื่อมั่นที่กำหนดหรือไม่

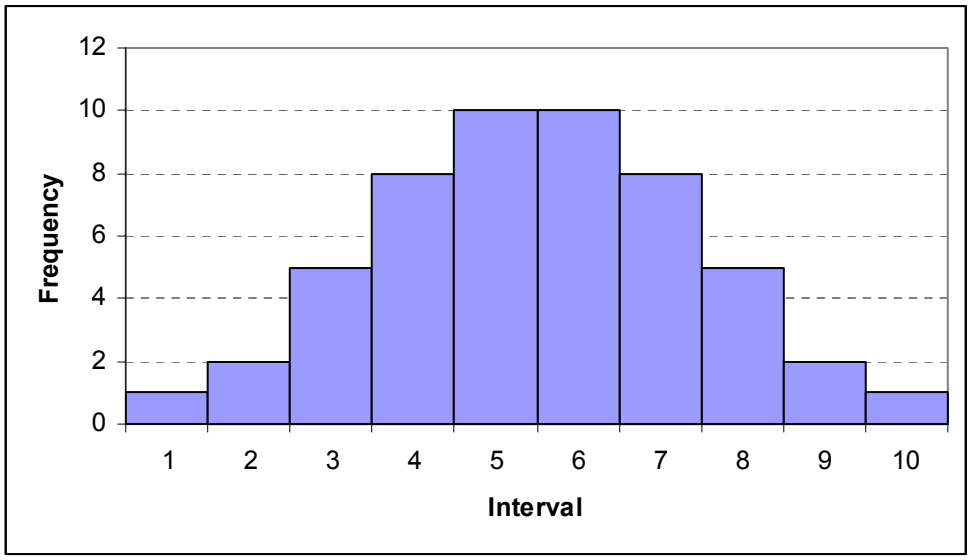
$$\chi^2 = \frac{\left(Actual - Expected\right)^2}{Expected} \qquad (10)$$

เมื่อได้ค่า χ^2 มาแล้วจากนั้นจึงนำค่ามาเทียบจากตาราง ถ้าค่าที่ได้มีค่ามากกว่าในตารางก็จะถือว่า ค่าที่ได้ถูกยอมรับในระดับนั้น เช่น ที่องศาความเป็นอิสระ (Degree of Freedom) 1 และค่าความเชื่อมั่นที่ 95 เปอร์เซนต์ จะต้องมีค่าไคสแควร์มากกว่า 3.84 เป็นต้น

ตารางที่ 1 แสดงตัวอย่างค่าไคสแควร์ที่ความเชื่อมั่นในระดับต่างๆ

องศาความเป็นอิสระ	ความเชื่อมั่น 95 %	ความเชื่อมั่น 99 %	ความเชื่อมั่น 99.9 %
1	3.84	6.64	10.83
2	5.99	9.21	13.82
3	7.82	11.35	16.27
4	9.49	13.28	18.47
5	11.07	15.09	20.52

3. ฮิสโทแกรม เป็นการแสดงผลของชุดข้อมูลในรูปของแท่งความถี่ โดยข้อมูลทั้งหมดจะถูกแบ่งเป็นช่วงๆ แสดงเป็นแผนภูมิแท่งซึ่งแสดงความถี่ของข้อมูลในช่วงนั้น ดังตัวอย่างในรูปที่ ... ข้อมูลแบ่งออกเป็น 10 ช่วง และมีความถี่ 1, 2, 5, 8, 10, 10, 8, 5, 2, 1 ตามลำดับ



ภาพที่ 5 แสดงตัวอย่างฮิสโทแกรมของข้อมูลตัวอย่าง

วิธีการวัดคุณภาพผลการแบ่งกลุ่มข้อมูล

การวัดผลการแบ่งกลุ่มโดยทั่วไปสามารถแบ่งตามเทคนิคการวัดผลได้ 2 วิธี คือ การวัดผลแบบภายใน (Internal quality) และการวัดผลแบบภายนอก (External quality)

1. การวัดผลแบบภายใน

การวัดผลแบบภายในไม่มีการนำเอาความรู้จากภายนอกมาใช้ในการวัดผล แต่เป็นการวัดผลด้วยการเปรียบเทียบหาความแตกต่างระหว่าง คลัสเตอร์ที่เกิดขึ้น ตัวอย่างของการวัดผลด้วยวิธีนี้ คือ เช่นการวัดค่าความเบี่ยงเบนจากค่าเฉลี่ยมาตรฐาน (Mean Square Error)

2. การวัดผลแบบภายนอก

การวัดผลแบบภายนอก เป็นการวัดผลเพื่อตรวจสอบว่าการแบ่งกลุ่มที่เกิดขึ้นเป็นการแบ่งกลุ่มที่ดีหรือไม่ โดยการเปรียบเทียบกับกลุ่มข้อมูลที่ทราบผลเฉลยอยู่แล้ว ตัวอย่างของการวัดผลโดยวิธีนี้คือการวัดค่าเอฟเมเชอร์ (F-measure) และการวัดค่าความบริสุทธิ์ของกลุ่มข้อมูล (Purity) ซึ่งในงานวิจัยนี้เราใช้ทั้งสองค่าในการวัดผล

2.1 ค่าเอฟเมเชอร์ เป็นการวัดผลแบบภายนอกอีกวิธีหนึ่ง ซึ่งเป็นการรวมกันระหว่างการวัดผลโดยใช้ค่าความถูกต้องและค่าความครบถ้วน การวัดผลนี้จะมองแต่ละคลัสเตอร์ที่แบ่งได้เปรียบเทียบกับกลุ่มข้อมูลผลเฉลย ค่าเอฟเมเชอร์สามารถหาได้จากสมการ

$$F(i,j) = \frac{2 * recall(i,j) * precision(i,j)}{recall(i,j) + precision(i,j)} \qquad (11)$$

โดยค่าความครบถ้วน (Recall) ของกลุ่มข้อมูลที่ i ในคลัสเตอร์ j คิดได้จากอัตราส่วนจำนวนข้อมูลของกลุ่ม i ซึ่งอยู่ในคลัสเตอร์ j ต่อจำนวนข้อมูลของกลุ่ม i ทั้งหมด ค่าความครบถ้วนสามารถหาได้จากสมการ

$$recall(i,j) = \frac{n_{ij}}{n_i} \qquad (12)$$

และค่าความถูกต้อง (Precision) ของกลุ่มข้อมูลที่ i ในคลัสเตอร์ j คิดได้จากอัตราส่วนจำนวนข้อมูลของกลุ่ม i ซึ่งอยู่ในคลัสเตอร์ j ต่อจำนวนข้อมูลของคลัสเตอร์ j ทั้งหมด ค่าความถูกต้องสามารถหาได้จากสมการ

$$precision(i,j) = \frac{n_{ij}}{n_j} \qquad (13)$$

และค่าเอฟเมเชอร์รวมสามารถหาได้จากสมการ

$$F = \sum_i \frac{n_i}{n} \max\{F(i,j)\} \qquad (14)$$

2.2 การวัดค่าความบริสุทธิ์ เป็นการวัดผลที่พิจารณาว่าในแต่ละกลุ่ม สามารถแยกข้อมูลได้อย่างชัดเจนมากน้อยเพียงใด โดยการคิดค่ารวมแบบถ่วงน้ำหนักของความถูกต้องที่มากที่สุดของกลุ่มข้อมูลในทุกกลุ่ม (ในคลัสเตอร์ j ใดๆ จะเลือกค่าความถูกต้องที่มากที่สุดสำหรับข้อมูลในกลุ่มที่ i จากนั้นจะนำมาคิดผลรวมแบบถ่วงน้ำหนักของความถูกต้องเหล่านั้น) การหาค่าความบริสุทธิ์มีสมการดังนี้

$$purity = \sum_j \frac{|n_j|}{n} \max\{precision(i,j)\} \qquad (15)$$

อุปกรณ์และวิธีการ

อุปกรณ์

1. เครื่องคอมพิวเตอร์ Pentium IV 2 GHz RAM 512 MB Hard disk 80 GB
2. ภาษา C/C++ ของไมโครซอฟต์เวอร์ชัน 2005

วิธีการ

จากที่กล่าวมาพบว่างานวิจัยสำหรับการแบ่งกลุ่มกระแสข้อมูลที่ผ่านมานั้นได้ถูกพัฒนาขึ้นเพื่อให้สามารถทำงานได้กับสภาวะของกระแสข้อมูลเท่านั้น แต่ไม่ได้พิจารณาถึงความเป็นไปได้ในการเปลี่ยนแปลงพฤติรรมของข้อมูลเท่าที่ควร โดยเฉพาะเมื่อเกิดการเปลี่ยนแปลงในด้านจำนวนกลุ่ม เช่น มีกลุ่มข้อมูลหายไป หรือ กลุ่มข้อมูลเกิดขึ้น จะทำให้คุณภาพของคลัสเตอร์ที่ได้ลดลง ดังนั้นเราจึงต้องการเสนอ เทคนิคการแบ่งกลุ่มกระแสข้อมูลที่สามารถรองรับการเปลี่ยนแปลงพฤติกรรมต่างๆ ของข้อมูลได้ดีขึ้น โดยปรับในแง่ของโครงสร้างการทำงาน และปรับตัวแทนของกลุ่มข้อมูล และการหาค่าระยะห่าง เพื่อให้เหมาะสมต่อโครงสร้างการทำงานนี้

1. แนวคิดการทำงาน

สำหรับกระแสข้อมูลซึ่งพฤติกรรมของข้อมูลสามารถเปลี่ยนแปลงได้ตลอดเวลา เราแบ่งประเภทของการเปลี่ยนแปลงออกได้เป็น 5 ประเภท คือ การเกิดขึ้นของกลุ่มข้อมูล การหายไปของกลุ่มข้อมูล การเปลี่ยนแปลงตัวเองของกลุ่มข้อมูล การรวมตัวกันของกลุ่มข้อมูล และการแยกตัวของกลุ่มข้อมูล ซึ่งงานวิจัยชิ้นนี้ถูกพัฒนาเพื่อรองรับการเปลี่ยนแปลงทั้งหมดที่กล่าวมา

1.1 การเกิดขึ้นของกลุ่มข้อมูล : กลุ่มข้อมูลใหม่สามารถเกิดขึ้นได้ ถ้ามีชุดของข้อมูลเกิดขึ้นในพื้นที่ใกล้กันเป็นจำนวนหนึ่งและทำให้เกิดการจับกลุ่มของข้อมูลในบริเวณนั้นมากกว่าบริเวณอื่นอื่น การเกิดกลุ่มข้อมูลใหม่ในช่วงแรกจึงอาจถูกมองว่าเป็นกลุ่มข้อมูลผิดปกติเนื่องจากข้อมูลมีจำนวนน้อย แต่เมื่อเวลาผ่านไปข้อมูลมีการจับกลุ่มกันชัดเจนจึงถือให้สามารถตั้งเป็นกลุ่มได้

1.2 การหายไปของกลุ่มข้อมูล : กลุ่มข้อมูลที่มีอยู่เดิมสามารถหายไปได้ เนื่องจากข้อมูลเก่าจะถูกพิจารณาว่ามีความสำคัญลดลงเรื่อยๆ ตามเวลา เมื่อเวลาผ่านไปกลุ่มข้อมูลใดๆ จะถูกลดค่าความสำคัญลงเรื่อยๆ ถ้ากลุ่มข้อมูลนั้นไม่มีสมาชิกใหม่เพิ่มขึ้น จนค่าความสำคัญน้อยกว่าเกณฑ์ที่กำหนดก็จะพิจารณาว่ากลุ่มข้อมูลนั้นหายไป

1.3 การเปลี่ยนแปลงตัวเองของกลุ่มข้อมูล : กลุ่มข้อมูลที่มีการเปลี่ยนแปลงตัวเอง เกิดจากการที่กลุ่มข้อมูลนั้นค่อยๆ เปลี่ยนแปลงพฤติกรรมไปทีละน้อย มีผลทำให้กลุ่มข้อมูล มีขนาดใหญ่ขึ้น เล็กลง หรือ เลื่อนตำแหน่งไปได้ การปรับให้เข้ากับการเปลี่ยนแปลงเหล่านี้สามารถทำได้รวดเร็วขึ้น ถ้ามีการลดค่าความสำคัญของข้อมูลเก่าตามเวลา

1.4 การรวมตัวกันของกลุ่มข้อมูล : กลุ่มข้อมูลสามารถรวมตัวกับกลุ่มข้อมูลอื่นได้ ถ้ากลุ่มข้อมูลมีการเปลี่ยนแปลงตัวเอง และเข้าใกล้กับกลุ่มข้อมูลอื่นจนมีพฤติกรรมส่วนใหญ่ใกล้เคียงกัน กลุ่มข้อมูลทั้งสองกลุ่มจะถูกรวมกันเป็นกลุ่มข้อมูลที่ใหญ่ขึ้นและครอบคลุมพฤติกรรมมากขึ้น

1.5 การแยกตัวของกลุ่มข้อมูล : กลุ่มข้อมูลใดๆสามารถแยกตัวออกจากกันได้ ถ้าพิจารณาข้อมูลภายในกลุ่มแล้วพบว่าเกิดเป็นกลุ่มย่อยภายในที่แตกแยกกันอย่างชัดเจน การแยกตัวอาจเกิดได้จากการรวมกลุ่มที่ผิดพลาดในขณะที่ข้อมูลมีจำนวนน้อย หรือการแยกตัวโดยพฤติกรรมของข้อมูลเอง

1.6 โครงสร้างการทำงานโดยรวม : แนวคิดการทำงานทั้งหมดเป็นดังนี้ ในการแบ่งกลุ่มกระแสข้อมูลสามารถเริ่มได้จากศูนย์ โดยเริ่มพิจารณาข้อมูลทุกตัวที่เกิดขึ้นเป็นข้อมูลโดดเดี่ยวและเก็บไว้เพื่อพิจารณาต่อไป เมื่อข้อมูลโดดเดี่ยวมากขึ้นและเกิดการจับกลุ่มขึ้นจำนวนหนึ่งจึงพิจารณาให้เป็นการเกิดขึ้นของกลุ่มข้อมูล ซึ่งเมื่อเป็นกลุ่มข้อมูลแล้วจึงสามารถรับข้อมูลใหม่เข้าเป็นสมาชิกได้ ส่วนข้อมูลใหม่ที่ไม่มีความคล้ายคลึงกับกลุ่มข้อมูลที่มีอยู่ก็จะพิจารณาว่าเป็นข้อมูลโดดเดี่ยวต่อไป จนกว่าจะมีการจับกลุ่มของข้อมูลโดดเดี่ยวขึ้นอีกจึงสามารถตั้งเป็นกลุ่มข้อมูลใหม่ได้ เมื่อเวลาผ่านไปข้อมูลทั้งหมดจะถูกลดความสำคัญลงเรื่อยๆ ทำให้กลุ่มข้อมูลมีการเปลี่ยนแปลงตัวเองตามข้อมูลใหม่ได้รวดเร็วขึ้น และยังทำให้กลุ่มข้อมูลที่ล้าสมัยหายไปได้อีกด้วย เมื่อมีกลุ่มข้อมูลเกิดขึ้นจำนวนหนึ่งอาจมีบางกลุ่มที่มีความคล้ายคลึงกันมากเกิดขึ้นจึงสามารถมีการรวมตัวกันของกลุ่มข้อมูลได้ หรือเมื่อเวลาผ่านไปกลุ่มข้อมูลมีลักษณะแตกแยกกันมากขึ้นก็ควรแยกข้อมูลกลุ่มนั้นออกจากกัน

2. คำจำกัดความต่างๆ

2.1 ข้อมูลที่เก็บไว้ในระบบ คือ ข้อมูลทั้งหมดที่ระบบเก็บไว้ในหน่วยความจำเพื่อใช้ประมวลผล เราจะแบ่งข้อมูลที่เก็บไว้ในระบบออกเป็น 2 ส่วน คือ ข้อมูลโดดเดี่ยว (Isolate Data Point) กลุ่มข้อมูลไม่ใช้ไม่ใช้งาน (Inactive Cluster) และ กลุ่มข้อมูลใช้งาน (Active Cluster)

2.2 ข้อมูลโดดเดี่ยว คือ ข้อมูลหนึ่งตัวที่ไม่ถูกจัดเข้าอยู่ในกลุ่มข้อมูลใด ถูกเหลือทิ้งไว้ในระบบเพื่อใช้พิจารณาต่อไป อาจเกาะกลุ่มกันและเกิดเป็นกลุ่มใหม่ หรือถูกลดค่าความสำคัญจนหายไปในที่สุด

2.3 กลุ่มข้อมูลไม่ใช้งาน คือ ข้อมูลโดดเดี่ยวหรือ กลุ่มของข้อมูลโดดเดี่ยวที่เริ่มมีการจับกลุ่มกัน แต่ยังไม่สามารถตั้งตัวเป็นกลุ่มข้อมูลใช้งานได้ เนื่องจากค่าความสำคัญไม่เพียงพอ

2.4 กลุ่มข้อมูลใช้งาน คือ กลุ่มข้อมูลที่พิจารณาเป็นโมเดลของระบบ สามารถรับข้อมูลใหม่เป็นสมาชิกได้ถ้ามีความคล้ายคลึงกับข้อมูลภายในกลุ่ม

2.5 กลุ่มข้อมูล คือ ข้อมูลโดดเดี่ยว กลุ่มข้อมูลไม่ใช้งาน หรือ กลุ่มข้อมูลใช้งาน ที่มีค่าเก็บไว้ในระบบ ในความเป็นจริง ข้อมูลทั้ง 3 แบบถูกเก็บไว้ในรูปแบบตัวแทนคลัสเตอร์ (Cluster Representation) เหมือนกัน ดังนั้นเมื่อกล่าวถึงคำว่า กลุ่มข้อมูล จะไม่ได้เจาะจงว่าเป็นข้อมูลโดดเดี่ยว กลุ่มข้อมูลไม่ใช้งาน หรือ กลุ่มข้อมูลใช้งาน

2.6 การเลือนหายของข้อมูล คือ การลดค่าความสำคัญของข้อมูลเมื่อเวลาผ่านไป เนื่องจากสำหรับกระแสข้อมูล รูปแบบของกลุ่มข้อมูลสามารถมีการเปลี่ยนแปลงได้ตลอดเวลา เราจึงควรให้ความสำคัญกับข้อมูลใหม่มากกว่าข้อมูลเก่า โดยการลดความสำคัญของข้อมูลเก่าลงเรื่อยๆตามเวลา เพื่อให้ระบบสามารถปรับตัวเข้ากับการเปลี่ยนแปลงได้ เราใช้ฟังก์ชันการเลือนหายเพื่อลดค่าความสำคัญของข้อมูล ให้ λ คืออัตราการเลือนหาย และ t คือเวลาที่ผ่านไป ฟังก์ชันการเลือนหายมีสมการดังนี้

$$f(t) = 2^{-\lambda t} \qquad (16)$$

2.7 ค่าความสำคัญ คือ จำนวนสมาชิกเสมือนของกลุ่มข้อมูลในขณะนั้น เหตุที่เรียกว่าเสมือนเนื่องจากจำนวนข้อมูลจะเสมือนว่ามีการลดลงตามเวลา เนื่องจากมีการเลือนหาย เมื่อเริ่มต้นข้อมูลแต่ละตัวมีค่าความสำคัญเป็น 1 แต่เมื่อเวลาผ่านไปจะมีค่าลดลงเรื่อยๆ กลุ่มข้อมูลใดๆ จะมีความสำคัญเพิ่มขึ้นได้ก็ต่อเมื่อกลุ่มข้อมูลนั้นรับสมาชิกเพิ่ม ซึ่งเป็นไปได้สองแบบ คือ รวมกับกลุ่มข้อมูลอื่น หรือรับข้อมูลเข้าใหม่เป็นสมาชิก

3. การเก็บตัวแทนคลัสเตอร์ (Cluster Representation)

การเก็บข้อมูลในระบบทั้ง ข้อมูลโดดเดี่ยว กลุ่มข้อมูลไม่ใช้งาน และกลุ่มข้อมูลใช้งานทั้งหมดจะเก็บในรูปแบบของตัวแทนคลัสเตอร์ที่เรียกว่า Fading Cluster Structure (FC) (Aggarwal *et al.*, 2004) เนื่องจากสามารถรองรับการเลือนหายของข้อมูลได้ และได้เพิ่มการเก็บฮิสโทแกรมของคลัสเตอร์ α แท่ง ในแต่ละมิติ เพื่อใช้ในการพิจารณาการแยกตัวของกลุ่มข้อมูล และเรียกตัวแทนใหม่นี้ว่า Fading Cluster Structure with Histogram (FCH)

ให้ t เป็นเวลาปัจจุบัน $C = \{\overline{X}_1 ... \overline{X}_N\}$ เป็นสมาชิกของข้อมูลขนาด d มิติ ภายในคลัสเตอร์ ที่เกิดขึ้นที่เวลา $T_1 ... T_N$ ตามลำดับ จะได้ตัวแทนคลัสเตอร์ที่เกิดจาก C และ t เป็น

$$FCH(C,t) = (\overline{FC1(C,t)}, \overline{FC2(C,t)}, W(t), \overline{H(C,t)}) \qquad (17)$$

โดยที่

$$\overline{FC1(C,t)} = (FC1^1 ... FC1^d) \qquad (18)$$

และ

$$\overline{FC2(C,t)} = (FC2^1 ... FC2^d) \qquad (19)$$

$FC1^j(C,t)$ เป็นผลรวมถ่วงน้ำหนักด้วยค่าการเลือนหายของข้อมูลทั้งหมดของคลัสเตอร์ในมิติที่ j

$$FC1^j(C,t) = \sum_{i=1}^{N} f(t - T_i) \cdot (x_i^j) \qquad (20)$$

$FC2^j(C,t)$ เป็นผลรวมถ่วงน้ำหนักด้วยค่าการเลือนหายของข้อมูลยกกำลังสองของข้อมูลทั้งหมดของคลัสเตอร์ในมิติที่ j

$$FC2^j(C,t) = \sum_{i=1}^{N} f(t - T_i) \cdot (x_i^j)^2 \qquad (21)$$

W(t) เป็นผลรวมของค่าการเลือนหายของข้อมูลทั้งหมดในคลัสเตอร์ ซึ่งเราเรียกค่านี้ว่า ค่าความสำคัญ

$$W(t) = \sum_{i=1}^{N} f(t - T_i) \qquad (22)$$

และ $H^j(C,t)$ เป็นฮิสโทแกรม 10 แท่งของข้อมูลทั้งหมดในคลัสเตอร์ในมิติที่ j

$$H^j(C,t) = (h^{j1}(C,t) \ldots h^{j10}(C,t)) \qquad (23)$$

โดยที่ $H_l^j(C,t)$ เป็น

$$H_l^j(t) = \sum_{i=1}^{N} f(t - T_i) \cdot (x_i^j) \cdot (y_{il}^j) \qquad (24)$$

y_{il} เป็นค่าน้ำหนักของข้อมูล x_i ในฮิสโทแกรมช่วงที่ l

$$y_{il} = \begin{cases} 1 & if \quad l \cdot b + left \le x_i \le (l+1) \cdot b + left \\ 0 & otherwise \end{cases} \qquad (25)$$

left เป็นค่าต่ำสุดของข้อมูลภายในกลุ่ม

$$left = \min(x_i^j) \qquad (26)$$

right เป็นค่าต่ำสุดของข้อมูลภายในกลุ่ม

$$right = \max(x_i^j) \qquad (27)$$

และ *b* เป็นความกว้างของแต่ละช่วงของฮิสโทแกรม

$$b = \frac{left + right}{\alpha} \qquad (28)$$

4. การจัดการกับฮิสโทแกรม

ในการเก็บตัวแทนคลัสเตอร์ด้วย FC เราไม่สามารถบอกถึงความแตกแยกภายในกลุ่มข้อมูลได้ ดังนั้นเราจึงเพิ่มการเก็บฮิสโทแกรมภายในกลุ่ม เพื่อใช้พิจารณาความแตกแยกของข้อมูลภายในกลุ่ม และมีข้อมูลเพียงพอที่สามารถสร้างกลุ่มข้อมูลใหม่ที่แยกออกจากกันได้

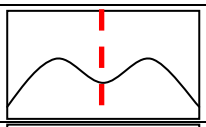
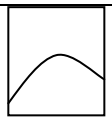
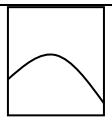
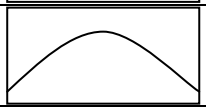
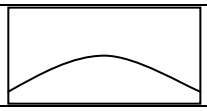
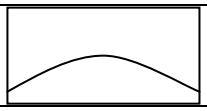
เราเก็บฮิสโทแกรม α แท่ง ในแต่ละมิติ ของในแต่ละกลุ่มข้อมูล การสร้างฮิสโทแกรมทำได้โดยพิจารณาค่าต่ำสุด และค่าสูงสุดของข้อมูลภายในกลุ่ม และแบ่งช่วงระหว่างค่าสูงสุดและค่าต่ำสุดออกเป็น α ช่วง เท่าๆกัน เมื่อมีการเปลี่ยนแปลงค่าสูงสุด หรือค่าต่ำสุด ซึ่งทำให้ช่วงภายในมีการเปลี่ยนแปลง และมีการเหลื่อมกันระหว่างช่วงเก่า และช่วงใหม่ เราจะแตกฮิสโทแกรมช่วงเก่าแต่ละช่วงออกและถ่วงน้ำหนักตามค่าความเหลื่อมเพื่อนำไปสร้างเป็นช่วงใหม่ ฮิสโทรแกรมจะถูกสร้างขึ้นและเก็บไว้ตั้งแต่ยังเป็นข้อมูลโดดเดี่ยว แต่จะถูกใช้ขณะที่เป็นกลุ่มข้อมูลใช้งานแล้วเท่านั้น เนื่องจากกลุ่มข้อมูลใช้งานเท่านั้นที่สามารถรับสมาชิกใหม่ได้และมีผลต่อความถูกต้องของระบบ

การพิจารณาการแยกตัวของกลุ่มข้อมูลจะถูกพิจารณาจากฮิสโทแกรมที่เก็บไว้ (Milenova and Campos, 2003) จะพิจารณาว่าจุดแบ่งของข้อมูลเกิดจากส่วนที่มีลักษณะเป็นหุบเขา (valley) อยู่ระหว่างจุดยอด (peek) 2 จุด ของฮิสโทแกรม โดยที่หุบเขาซึ่งเป็นจุดแบ่งนั้น จะต้องมีความแตกต่างกับจุดยอดที่ต่ำกว่าอย่างมีนัยสำคัญ โดยพิจารณาจากค่า ไคสแควร์ (Chi-Square) จุดแบ่งที่ดีที่สุดคือหุบเขาที่ต่ำที่สุดซึ่งมีนัยสำคัญนั่นเอง เมื่อได้จุดแบ่งที่ดีที่สุดแล้วเราจะแบ่งฮิสโทแกรมของมิตินั้น

ออกจากกัน ส่วนมิติอื่นเราจะถ่วงน้ำหนักแล้วเฉลี่ยฮิสโทแกรมออกไปให้มิติอื่นๆ ตามค่าน้ำหนักนั้น ส่วนค่า FC1, FC2 , W สามารถสร้างใหม่จากฮิสโทแกรมที่แบ่งแล้วได้

ลักษณะการแบ่งจะแสดงไว้ดังตารางที่ 2 ให้กลุ่มข้อมูลหนึ่งมีฮิสโทแกรมดังช่อง ฮิสโทแกรมก่อนแบ่ง โดยมิติที่ 1 มีลักษณะการแยกตัว เมื่อทำการแยกแล้วจะได้ข้อมูล 2 กลุ่ม สำหรับมิติแรก ฮิสโทแกรมจะถูกแบ่งออกเป็นสองส่วนให้กับทั้งสอง กลุ่มข้อมูล ส่วนมิติที่ 2 ฮิสโทแกรมทั้งหมดจะถูกเฉลี่ยให้กับทั้งสองกลุ่ม

ตารางที่ 2 แสดงลักษณะการแบ่งฮิสโทแกรม

มิติ	ฮิสโทแกรมก่อนแบ่ง	ฮิสโทแกรมหลังการแบ่งส่วนแรก	ฮิสโทแกรมหลังการแบ่งส่วนหลัง
มิติที่ 1			
มิติที่อื่น			

5. การหาค่าระยะห่าง

เราแบ่งการหาค่าระยะห่างออกเป็น 2 กรณี คือ การหาค่าระยะห่างระหว่างข้อมูลเกิดใหม่กับกลุ่มข้อมูล และการหาค่าระยะห่างระหว่างกลุ่มข้อมูลกับกลุ่มข้อมูลด้วยกัน โดยการหาค่าระยะห่างทุกกรณีจะพิจารณาแยกกันในแต่ละมิติ หลังจากนั้นจึงนำใช้ค่าเฉลี่ยเพื่อหาผลลัพธ์จากทุกมิติ

5.1 การหาค่าระยะห่างระหว่างข้อมูลเกิดใหม่กับกลุ่มข้อมูล (Cluster-Point Distance) ค่าระยะห่างนี้ใช้เมื่อต้องการหาว่าข้อมูลที่เกิดใหม่ควรจะเป็นสมาชิกของกลุ่มใด (กลุ่มที่พิจารณาต้องเป็นกลุ่มข้อมูลใช้งานเท่านั้น) โดยอ้างอิงจากกลุ่มข้อมูลเป็นหลัก กล่าวคือจะวัดค่าระยะห่างระหว่างจุดศูนย์กลางของกลุ่มกับข้อมูลเกิดใหม่เป็นจำนวนเท่าของรัศมีของกลุ่มข้อมูลนั้น ซึ่งรัศมีของกลุ่มข้อมูมีค่าเท่ากับค่าเบี่ยงเบนมาตรฐานของกลุ่มข้อมูลนั้น ดังนั้นสำหรับความแตกต่างระหว่างจุดศูยน์กลางของกลุ่มกับข้อมูลใหม่ที่เท่ากัน กลุ่มที่มีรัศมีกว้างจะให้ค่าระยะห่างน้อยกว่า

กลุ่มที่มีรัศมีแคบ ให้ C เป็นกลุ่มข้อมูล และ x เป็นข้อมูลเกิดใหม่ ค่าระยะห่างระหว่าง C และ x สามารถหาได้จากสมการ

$$dist(C,x) = \frac{1}{d} \cdot \sum_{j=1}^{d} \left| \frac{center_C^j - x^j}{radius_C^j} \right| \qquad (25)$$

5.2 การหาค่าระยะห่างระหว่างกลุ่มข้อมูลและกลุ่มข้อมูล (Cluster-Cluster Distance) ค่าระยะห่างนี้ถูกใช้เมื่อต้องการหาค่าความคล้ายคลึงของกลุ่มข้อมูลสองกลุ่มใดๆ (อาจเป็นข้อมูลโดดเดี่ยว กลุ่มข้อมูลไม่ใช้งาน และกลุ่มข้อมูลใช้งานก็ได้) หรือใช้เมื่อต้องการหาคู่ของกลุ่มข้อมูลที่มีความคล้ายคลึงมากที่สุด สามารถหาได้จากระยะห่างระหว่างจุดศูนย์กลางของกลุ่มข้อมูลทั้งสอง

$$dist(C_a, C_b) = \frac{1}{d} \sum_{j=1}^{d} \left| center_{C_a}^j - center_{C_b}^j \right| \qquad (26)$$

6. อัลกอริทึม E-Stream

ในหัวข้อนี้เราจะกล่าวถึงรายละเอียดของอัลกอริทึมที่เรานำเสนอ และเราตั้งชื่อให้อัลกอริทึมนี้ว่า E-Stream ซึ่งเป็นอัลกอริทึมที่สามารถรองรับการเปลี่ยนแปลงทั้ง 5 ประเภท ที่กล่าวไว้เบื้องต้นได้ ในที่นี้จะใช้โค้ดจำลอง (Pseudo-Code) เป็นตัวอธิบายการทำงาน ซึ่งภายในโค้ดมีตัวแปรต่างๆ ที่ควรทราบดังตารางที่ 3

ตารางที่ 3 ความหมายของตัวแปรต่างๆ ที่ใช้ในโค้ดจำลอง

ตัวแปร	ความหมาย
|FCH|	จำนวนกลุ่มข้อมูลใช้งานในปัจจุบัน
FCH_i	กลุ่มข้อมูลใช้งานที่ i
FCH_i.W	ค่านำหนักของกลุ่มใช้งานที่ i
FCH_i.sd	ค่าเบี่ยงเบนมาตรฐานของกลุ่มใช้งานที่ i
S	เซ็ตของคู่ลำดับของดัชนีของกลุ่มใช้งานที่มีการแตกตัวในรอบการทำงานนั้น
#isolate	จำนวนข้อมูลโดดเดี่ยวในปัจจุบัน

6.1 E-Stream เป็นอัลกอริทึมหลักในการแบ่งกลุ่มกระแสข้อมูลโดยในบรรทัดที่ 1 เมื่อได้รับข้อมูลใหม่อัลกอริทึมจึงเริ่มทำงาน ในบรรทัดที่ 2 เราจะทำการเลือนกลุ่มข้อมูลเก่าทั้งหมดเพื่อลดความสำคัญลง และลบทิ้งถ้ามีค่าความคัญน้อยกว่าที่กำหนด บรรทัดที่ 3 พิจารณาการแตกแยกภายในกลุ่ม โดยพิจารณาจากฮิสโทแกรม ถ้าพบว่ามีการแตกแยกก็จะทำการแยกข้อมูลกลุ่มนั้นออกจากกัน บรรทัดที่ 4 พิจารณาว่าถ้ากลุ่มข้อมูลใดมีการเหลื่อล้ำกัน ก็จะรวมทั้งสองกลุ่มนั้นเข้าด้วยกัน บรรทัดที่ 5 ถ้าจำนวนกลุ่มข้อมูลในขณะนั้นมากกว่าจำนวนกลุ่มสูงสุด ก็จะรวมกลุ่มข้อมูลที่ใกล้กันที่สุดเข้าด้วยกันจนกระทั้งได้จำนวนกลุ่มไม่เกินที่กำหนด บรรทัดที่ 6 จากนั้นเราจะตรวจสอบว่ากลุ่มข้อมูลใช้งานมีการเกิดขึ้นหรือหายไปหรือไม่ บรรทัดที่ 7-10 เมื่อได้กลุ่มข้อมูลใช้งานในขณะนั้นแล้วจึงนำข้อมูลเข้าใหม่มาพิจารณาว่าควรจะเป็นสมาชิกของกลุ่มข้อมูลใช้งานใดมากที่สุด หรือควรถูกเหลือไว้เป็นข้อมูลโดดเดี่ยว ถ้ามีค่าระยะห่างไม่เกินค่าเกณฑ์รัศมีที่ยอมรับได้ (radius_factor) ซึ่งกำหนดเป็นอินพุทของระบบ เมื่อสามารถจัดการกับข้อมูลใหม่ได้แล้วจึงจบการพิจารณา และรอข้อมูลใหม่ตัวถัดไป

Algorithm E-Stream

1 retrieve new data $X_{\neg i}$
2 FadingAll
3 CheckSplit
4 MergeOverlapCluster
5 LimitMaximumCluster
6 FlagActiveCluster
7 (minDistance, index) ← FindClosestCluster
8 if minDistance < radius_factor
9 add X_i to FCH_{index}
10 else
11 create new FCH from X_i
12 waiting for new data

ภาพที่ 6 แสดงอัลกอริทึม E-Stream

6.2 FadingAll เป็นอัลกอริทึมในการเลือนข้อมูลทั้งหมดในระบบ และจะลบข้อมูลนั้นทิ้งเมื่อมีค่าความสำคัญน้อยกว่าค่าเกณฑ์การหาย (remove_threshold) ซึ่งกำหนดเป็นอินพุทของระบบ

6.3 CheckSplit เป็นอัลกอริทึมในการพิจารณาฮิสโทแกรมว่ามีการแยกตัวออกจากกันภายในกลุ่มหรือไม่ ถ้ามีจะทำการแยกออกจากกัน เมื่อมีการแยกกันแล้วเราจะเก็บดัชนีของคู่ที่แยกกันไว้เพื่อป้องกันการรวมตัวในภายหลัง

6.5 MergeOverlapCluster เป็นอัลกอริทึมในการรวมคลัสเตอร์ที่มีพื้นร่วมกันเข้าด้วยกัน โดยการตรวจสอบทุกคู่ของคลัสเตอร์ที่เป็นไปได้ ถ้าคู่ใดมีระยะห่างระหว่างศูนย์กลางน้อยกว่าค่าเกณฑ์การรวม (merge_thershold) ซึ่งกำหนดเป็นอินพุทของระบบ ถือว่ามีพื้นที่ร่วมกันและให้รวมตัวกันทันที

6.5 LimitMaximumCluster เป็นอัลกอริทึมในการจำกัดจำนวนของกลุ่มข้อมูลในระบบ โดยจะตรวจสอบว่าจำนวนกลุ่มมากกว่าค่าเกณฑ์จำนวนกลุ่มมากที่สุด (maximum_cluster) ซึ่งกำหนดเป็นอินพุทของระบบ หรือไม่ ถ้ามากกว่าจะทำการหาค่าระยะห่างของกลุ่มข้อมูลทุกคู่ที่เป็นไปได้ และวนรอบกลุ่มที่ใกล้ที่สุดเข้าด้วยกันจนกว่า จำนวนกลุ่มจะไม่เกินค่าเกณฑ์นี้

6.6 FlagActiveCluster เป็นอัลกอริทึมในการตรวจสอบว่าในขณะนั้นกลุ่มข้อมูลใดถือเป็นกลุ่มใช้งานบ้าง โดยการพิจารณาค่าความสำคัญของทุกกลุ่ม ถ้ากลุ่มใดมีค่าความสำคัญไม่น้อยกว่าค่าเกณฑ์น้ำหนักของคลัสเตอร์ใช้งาน (active_threshold) ซึ่งกำหนดเป็นอินพุทของระบบ จะกำหนดให้เป็นกลุ่มข้อมูลใช้งาน แต่ถ้ามีค่าน้อยกว่าที่กำหนดจะไม่กำหนดให้เป็นกลุ่มข้อมูลใช้งาน

6.7 FindClosestCluster เป็นอัลกอริทึมในการหากลุ่มข้อมูลใช้งานที่ใกล้ที่สุดสำหรับข้อมูลเกิดใหม่ โดยจะพิจารณากลุ่มข้อมูลทุกกลุ่ม ตรวจสอบว่าเป็นกลุ่มข้อมูลใช้งานหรือไม่ ถ้าเป็นให้หาค่าระยะห่างระหว่างข้อมูลเกิดใหม่และกลุ่มข้อมูลนั้น อัลกอริทึมจะคืนค่าระยะห่าง และดัชนีของกลุ่มที่ให้ค่าระยะห่างที่ต่ำที่สุดออกมา

Algorithm FadingAll

for i ← 1 to |FCH|

fading FCH_i

if FCH_i.W < fade_threshold

delete FCH_i

ภาพที่ 7 แสดงอัลกอริทึม FadingAll

Algorithm CheckSplit

for i ⟵ 1 to |FCH|

    for j ⟵ 1 to d

        if FCH_{ij} have split point

            split FCH_i

            S ⟵ S U {(i, |FCH|)}

ภาพที่ 8 แสดงอัลกอริทึม CheckSplit

Algorithm MergeOverlapCluster

for i ⟵ 1 to |FCH|

    for j ⟵ i+1 to |FCH|

        overlap[i,j] ⟵ dist(FCH_i, FCH_j) – merge_threshold*(FCH_i.sd + FCH_j.sd)

        if overlap[i,j] > 0

            if (i,j) not in S

                merge(FCH_i, FCH_j)

ภาพที่ 9 แสดงอัลกอริทึม MergeOverlapCluster

Algorithm LimitMaximumCluster

while |FCH| > maximum_cluster or #isolate > maximum_isolate

    for i ← i+1 to |FCH|

        for j ← 1 to |FCH|

            dist[i,j] ← dist(FCH_i, FCH_j)

    (first, second) ← $\text{argmin}_{(i,j)}$(dist[i,j])

    merge(FC_{first}, FC_{second})

ภาพที่ 10 แสดงอัลกอริทึม LimitMaximumCluster

```
Algorithm FlagActiveCluster
for i ← 1 to |FCH|
        if FCH_i.W > active_threshold
                flag FCH_i as active cluster
        else
                remove active flag from FCH_i
```

ภาพที่ 11 แสดงอัลกอริทึม FlagActiveCluster

```
Algorithm FindClosestCluster
for i ← 1 to |FCH|
        dist[i] ← dist(FCH_i, x_i)
 (minDistance,i) ← min(dist[i])
return (minDistance, i)
```

ภาพที่ 12 แสดงอัลกอริทึม FindClosestCluster

ผลและวิจารณ์ผลการทดลอง

1. ข้อมูลสำหรับทำการทดลอง

1.1 ชุดข้อมูลสังเคราะห์ (Synthetic Dataset) เราได้สร้างชุดข้อมูลสังเคราะห์ขนาด 2 มิติ จำนวน 8,000 ตัว โดยในแต่ละช่วงของเวลารูปแบบของข้อมูลจะมีการเปลี่ยนแปลงไปเรื่อยๆ สามารถแบ่งได้ 8 ช่วง ดังนี้

1.1 เริ่มต้นมีกลุ่มข้อมูล 4 กลุ่มซึ่งอยู่ในภาวะสมดุล

1.2 เกิดกลุ่มข้อมูลใหม่ขึ้น

1.3 กลุ่มข้อมูลที่ 1 หายไป คงเหลือกลุ่มข้อมูล

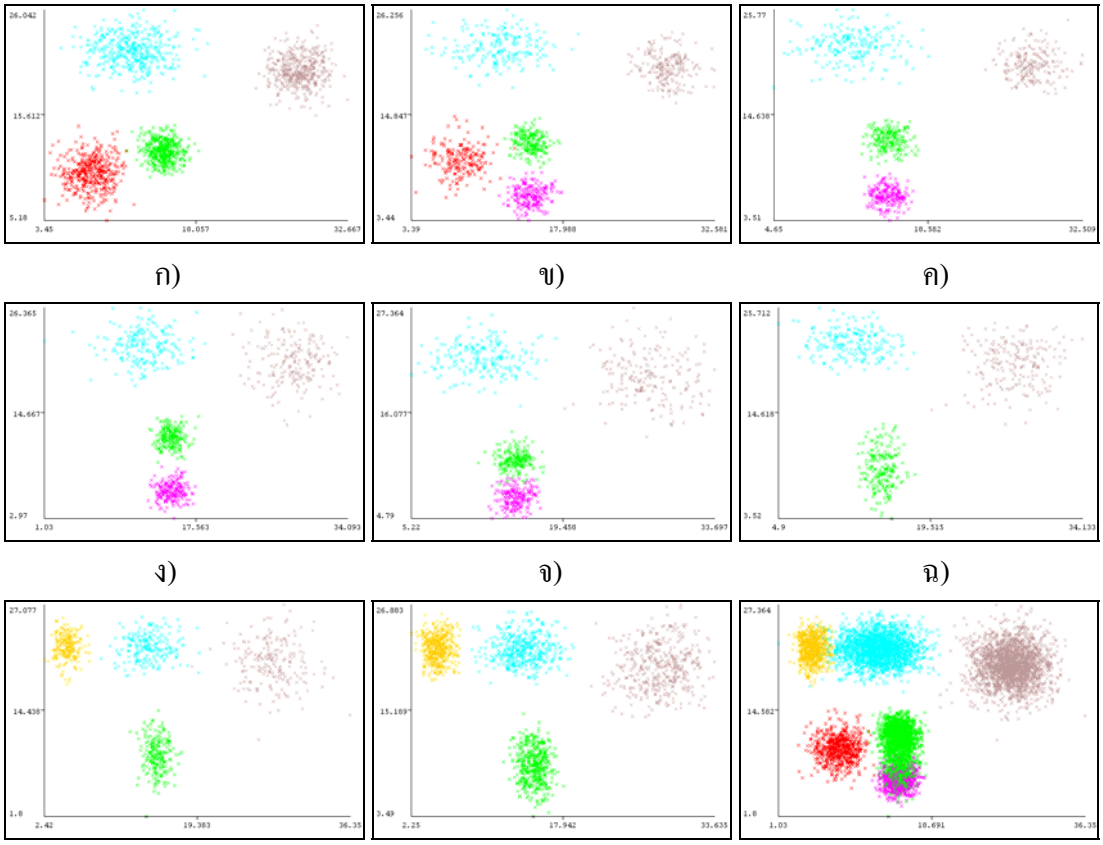
1.4 ข้อมูลกลุ่มที่ 4 มีการขยายขอบเขตของรัศมี

1.5 ข้อมูลกลุ่มที่ 2 และกลุ่มที่ 5 เลื่อนต่ำแหน่งเข้าใกล้กัน

1.6 ข้อมูลกลุ่มที่ 2 ได้รวมเข้ากับ ข้อมูลกลุ่มที่ 5 กลายเป็นกลุ่มที่มีขนาดใหญ่ขึ้น

1.7 ข้อมูลกลุ่มที่ 6 ได้แยกตัวออกจากข้อมูลกลุ่มที่ 3

1.8 กลุ่มข้อมูลทั้งหมดอยู่ในภาวะสมดุล



ก) ข) ค)

ง) จ) ฉ)

ช) ซ) ญ)

ภาพที่ 13 แสดงข้อมูลในแต่ละช่วงของสตรีม (ก-ซ) และแสดงข้อมูลทุกช่วง (ญ)

จากภาพที่ 13 (ญ) ซึ่งแสดงข้อมูลทุกช่วงของสตรีมลงบนภาพเดียวกัน พบว่าการมองข้อมูลทั้งหมดของสตีมพร้อมกันเป็นกลุ่มข้อมูลที่คงที่นั้นทำให้ลักษณะข้อมูลสับสนและไม่สามารถแบ่งแยกได้ แต่เมื่อมองแบ่งข้อมูลเป็นแต่ละช่วง (ก-ซ) จะพบว่าข้อมูลมีพฤติกรรมเปลี่ยนไปตามช่วงเวลา ดังนั้นเทคนิคการแบ่งกลุ่มกระแสข้อมูลที่ดีต้องไม่พิจารณาข้อมูลทั้งหมดคงที่ แต่จำเป็นต้องพิจารณาข้อมูลเป็นช่วงๆ และควรให้ผลลัพธ์ที่ทันสมัยตลอดเวลา

1.2 ชุดข้อมูลจริง เราใช้ชุดข้อมูลจาก KDDCup 1999 ชุดข้อมูลการตรวจจับการบุกรุกเครือข่าย ขนาด 34 มิติ จำนวน 494200 ตัว เนื่องจากเป็นชุดข้อมูลที่งานวิจัยทางการแบ่งกลุ่มกระแสข้อมูลทั่วไปนิยมใช้

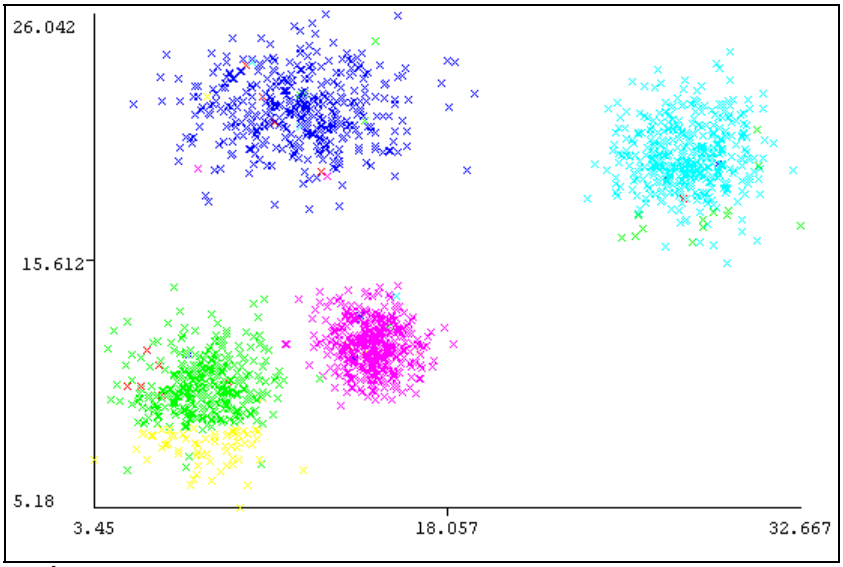
2. การวัดคุณภาพของคลัสเตอร์ที่ได้ในแต่ละช่วงเวลา

สำหรับการทดลองเรานี้ได้ทำการวัดคุณภาพของคลัสเตอร์ที่ได้จากอัลกอริทึมที่เรานำเสนอ E-Stream เทียบกับ HPStream ซึ่งเป็นอัลกอริทึมที่มีการยอมรับ โดยเราได้ปรับค่าพารามิเตอร์ต่างๆ ดังตารางที่ 1 เนื่องจาก E-Stream ไม่ได้กำหนดจำนวนกลุ่มคงที่ แต่กำหนดจำนวนกลุ่มที่มากที่สุดที่ยอมรับได้ ดังนั้นจึงตั้งค่าจำนวนกลุ่มสูงสุดไว้มากกว่าปกติ แต่สำหรับ HPStream นั้นจำนวนกลุ่มที่กำหนดจะคงที่ในทุกๆช่วงเวลา ดังนั้นเราจึงตั้งค่าไว้ที่ 5 เนื่องจากชุดข้อมูลสังเคราะห์ของเรามีจำนวนกลุ่มไม่เกิน 5 ในทุกๆช่วงของสตรีม ถือเป็นการทำให้ HPStream ได้เปรียบกว่า E-Stream เนื่องจากรู้จำนวนกลุ่มแล้ว และ HPStream ยังต้องการข้อมูลจำนวนหนึ่งเพื่อนำไปหากลุ่มตั้งต้นก่อนจะเริ่มทำงานได้จริง เราจึงกำหนดไว้จำนวน 100 ตัว

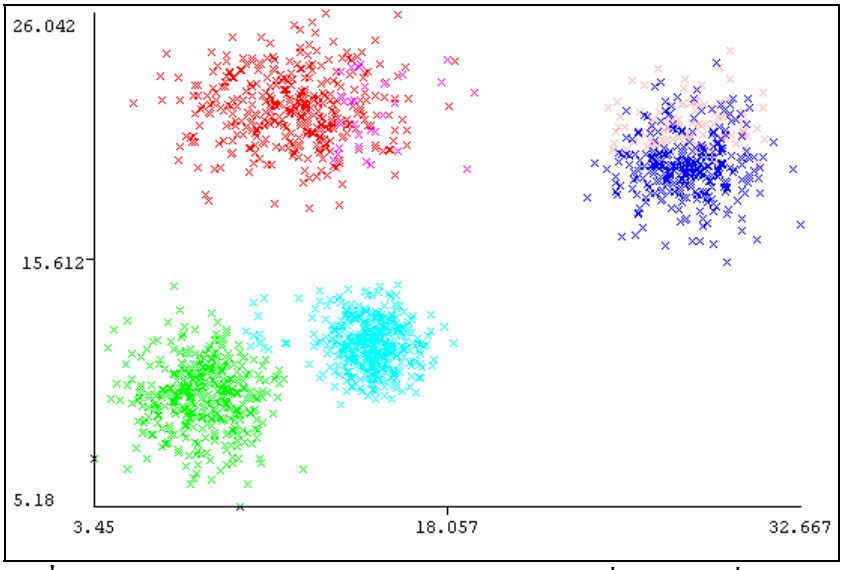
ตารางที่ 4 ค่าพารามิเตอร์ต่างๆ ของทั้งสองอัลกอริทึม

Algorithm E-Stream	Algorithm HPStream
maximum_cluster 10	num_group 5
decay_rate 0.1	decay_rate 0.1
radius_factor 3	radius_factor 3
stream_speed 100	stream_speed 100
fading_threshold 0.1	num_initial 100
merge_threshold 1.25	
active_threshold 5	
maximum_isolate 10	
α 10	

โดยการใช้ชุดข้อมูลที่กล่าวมา เราได้ทำการวัดคุณภาพของคลัสเตอร์ที่ได้ในแต่ละช่วงพฤติกรรมของข้อมูล ทั้ง 8 ช่วง ได้ผลดังนี้

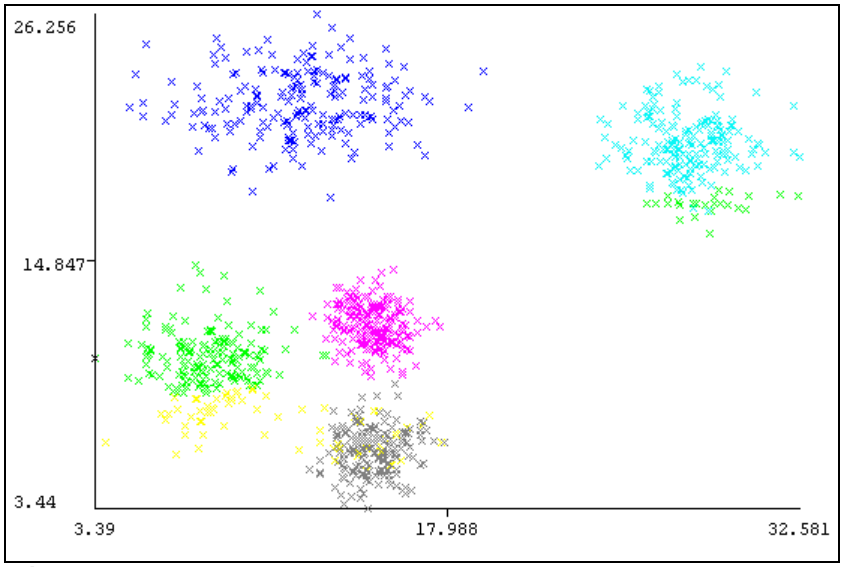


ภาพที่ 14 แสดงผลการแบ่งกลุ่มของ E-Stream ในช่วงที่ 1 (ข้อมูลที่ 1 – 1600)

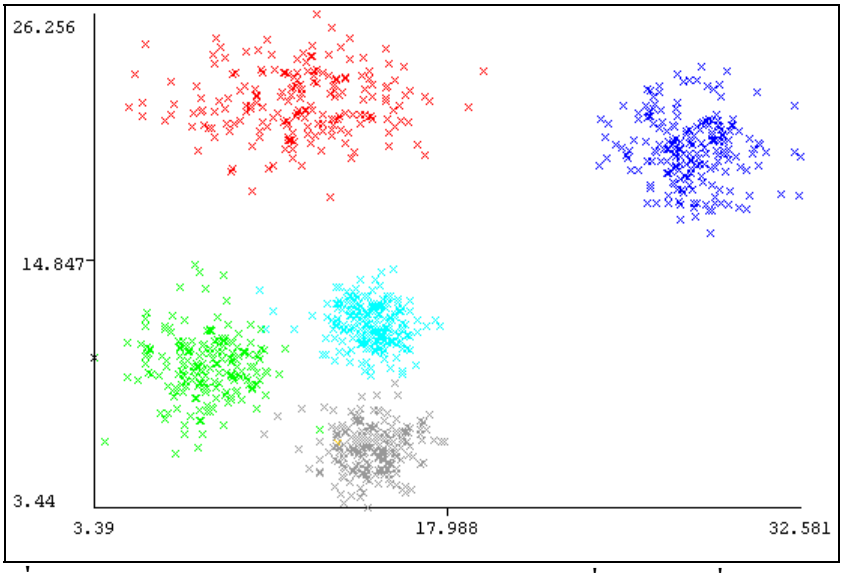


ภาพที่ 15 แสดงผลการแบ่งกลุ่มของ HPStreamในช่วงที่ 1 (ข้อมูลที่ 1 – 1600)

ในช่วงที่ 1 ข้อมูลทั้งหมดมี 4 กลุ่มและมีเสถียรภาพ รูปแบบคลัสเตอร์ของทั้งสองอัลกอริทึมผิดไปจากกลุ่มจริงเล็กน้อย และจำนวนกลุ่มของ E-Stream มีมากกว่าอย่างเห็นได้ชัด เนื่องช่วงเริ่มทำงาน E-Stream ยังไม่มีกลุ่มข้อมูลตั้งตน จึงพิจารณาข้อมูลทุกตัวเป็นข้อมูลที่ผิดปกติ เมื่อผ่านไปซักระยะข้อมูลจึงเกิดการจับกลุ่มกันและเกิดเป็นคลัสเตอร์ขึ้น แต่ HPStream จำเป็นต้องใช้ข้อมูลตั้งต้นที่เก็บมาจำนวนหนึ่ง (ในที่นี้กำหนดให้เป็น 100 ตัว) เพื่อทำคลัสเตอริ่งแบบออฟไลน์หากลุ่มตั้งต้นให้อัลกอริมทึม ทำให้ HPStream ได้เปรียบอยู่เล็กน้อย

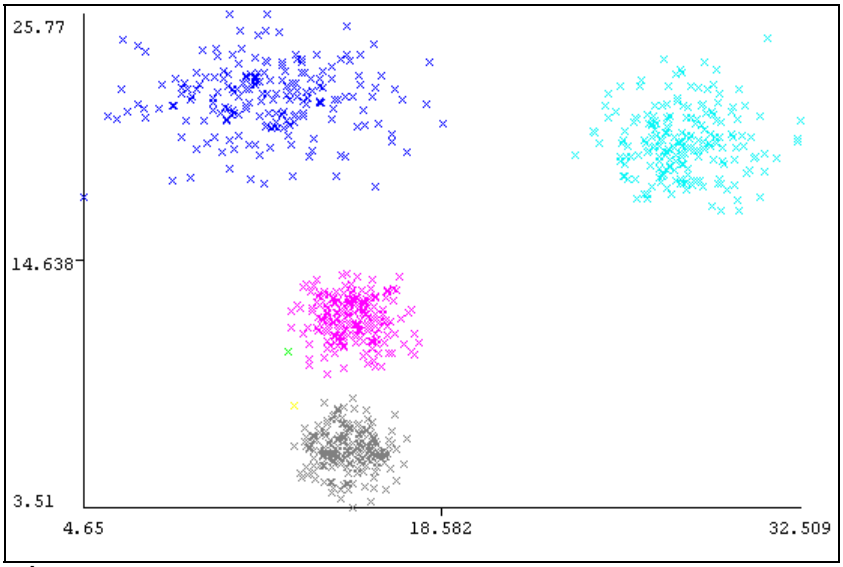


ภาพที่ 16 แสดงผลการแบ่งกลุ่มของ E-Stream ในช่วงที่ 2 (ข้อมูลที่ 1601 – 2600)

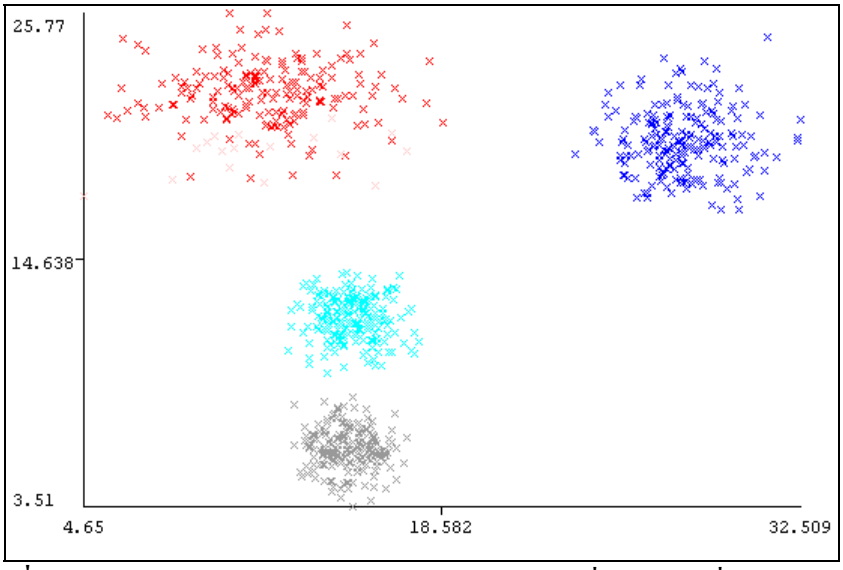


ภาพที่ 17 แสดงผลการแบ่งกลุ่มของ HPStream ในช่วงที่ 2 (ข้อมูลที่ 1601 – 2600)

ในช่วงที่ 2 ซึ่งเกิดข้อมูลกลุ่มใหม่ HPStream ให้คุณภาพที่ดีกว่าเล็กน้อยเนื่องจากสามารถหากลุ่มจริงได้ครบ และกลุ่มข้อมูลในช่วงนี้มี 5 กลุ่ม ตามที่กำหนดไว้ในพารามิเตอร์ของ HPStream พอดี แต่ E-Stream นั้นยังมีการรวมกลุ่มผิดอยู่เล็กน้อย โดยรวมเอากลุ่มที่เกิดใหม่เข้ากับกลุ่มข้อมูลที่สร้างผิดพลาดในช่วงแรก เนื่องจากระหว่างทั้งสองกลุ่มนั้นมีข้อมูลระหว่างกันเล็กน้อย และกลุ่มข้อมูลที่ผิดพลาดนั้นมีลักษณะกระจายทางด้านแนวนอนอยู่แล้ว ทำให้มีแนวโน้มการรวมตัวกับกลุ่มข้อมูลใหม่ที่อยู่ด้านข้างได้ง่าย แต่เมื่อมีข้อมูลมากขึ้นกลุ่มที่ผิดพลาดนั้นจึงแตกตัวออกเป็นกลุ่มใหม่ที่ถูกต้อง

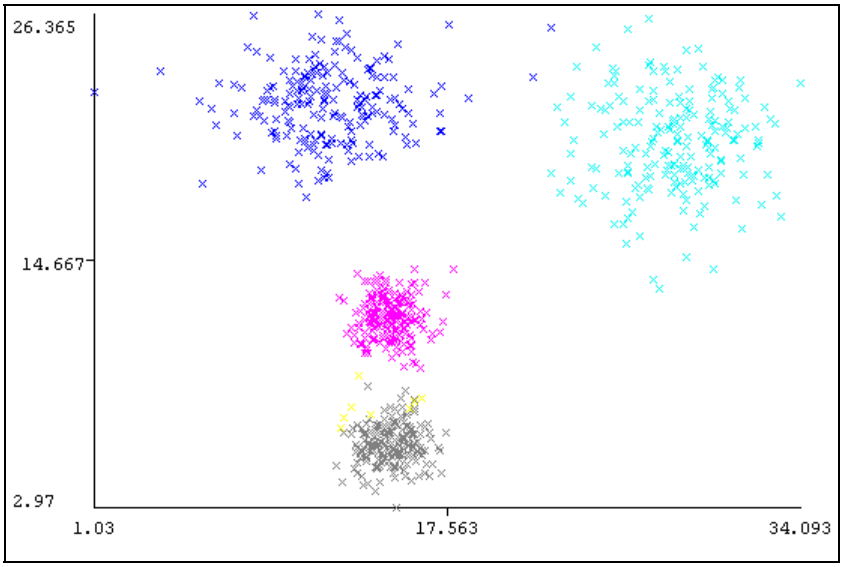


ภาพที่ 18 แสดงผลการแบ่งกลุ่มของ E-Stream ในช่วงที่ 3 (ข้อมูลที่ 2601 – 3400)

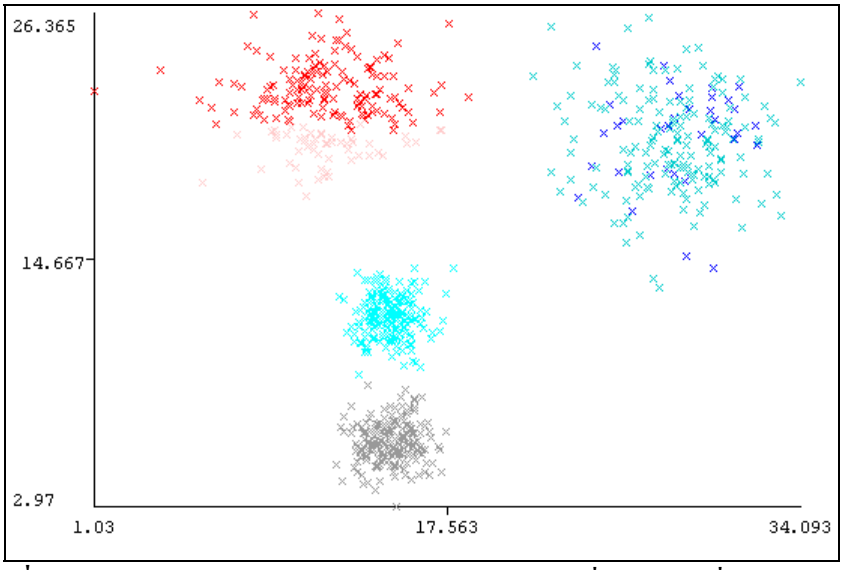


ภาพที่ 19 แสดงผลการแบ่งกลุ่มของ HPStream ในช่วงที่ 3 (ข้อมูลที่ 2601 – 3400)

ในช่วงที่ 3 กลุ่มข้อมูลด้านซ้ายล่างหายไป E-Stream สามารถปรับให้เข้ากับการเปลี่ยนแปลงนี้ได้อย่างดี แต่ HPStream ได้พยายามสร้างกลุ่มข้อมูลให้ครบ 5 กลุ่ม จึงไปสร้างกลุ่มข้อมูลใหม่โดยแยกตัวออกมาจากกลุ่มข้อมูลที่มีความถูกต้องอยู่แล้ว แม้ว่ากลุ่มข้อมูลใหม่ที่เกิดขึ้นจะไม่มีลักษณะแตกแยกจากกลุ่มเก่าอย่างมีนัยสำคัญก็ตาม

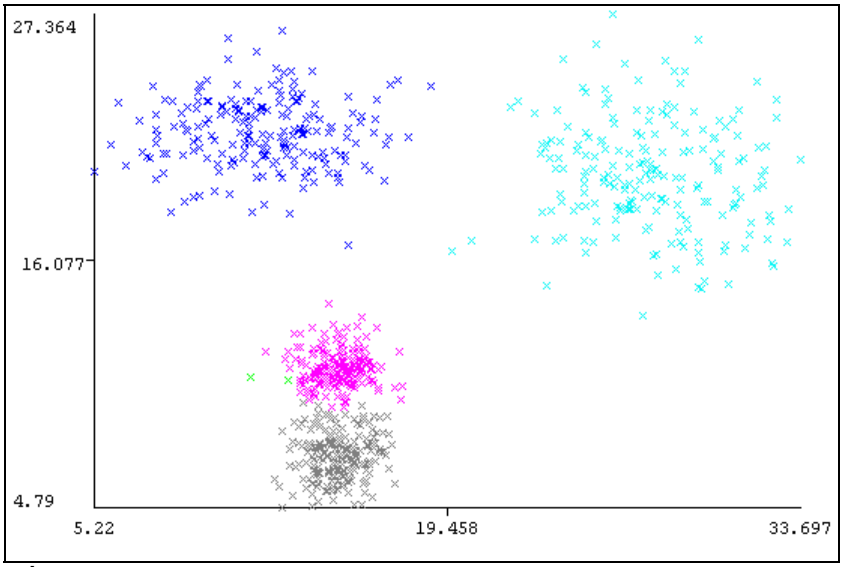


ภาพที่ 20 แสดงผลการแบ่งกลุ่มของ E-Stream ที่เสนอในช่วงที่ 4 (ข้อมูลที่ 3401 – 4200)

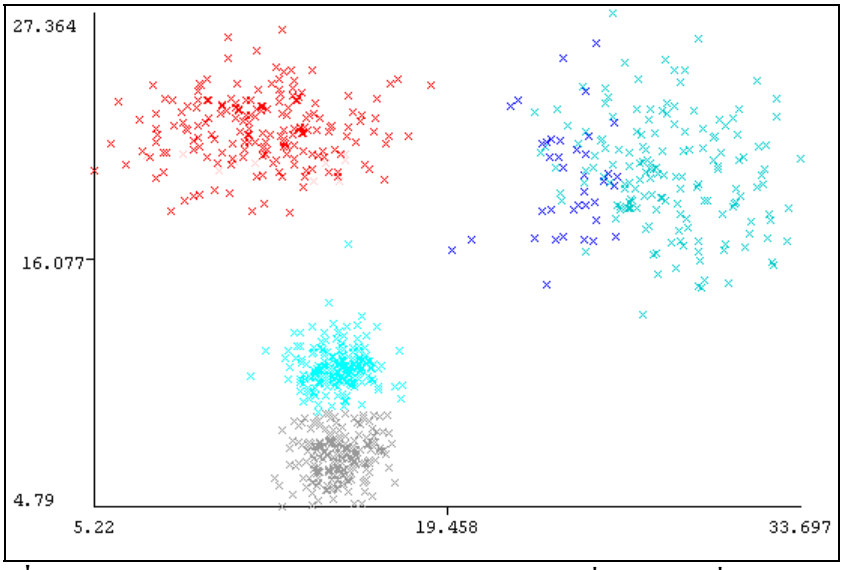


ภาพที่ 21 แสดงผลการแบ่งกลุ่มของ HPStream ในช่วงที่ 4 (ข้อมูลที่ 3401 – 4200)

ในช่วงที่ 4 กลุ่มข้อมูลขยายตัว ทั้งสองอัลกอริทึมสามารถรองรับการขยายตัวของกลุ่มข้อมูลได้ แต่ HPStream ยังคงพยายามสร้างกลุ่มข้อมูลให้ครบ 5 โดยเปลี่ยนกลุ่มข้อมูลใหม่ไปเรื่อยๆ ในขณะที่ E-Stream นั้นค่อนข้างมีเสถียรภาพ

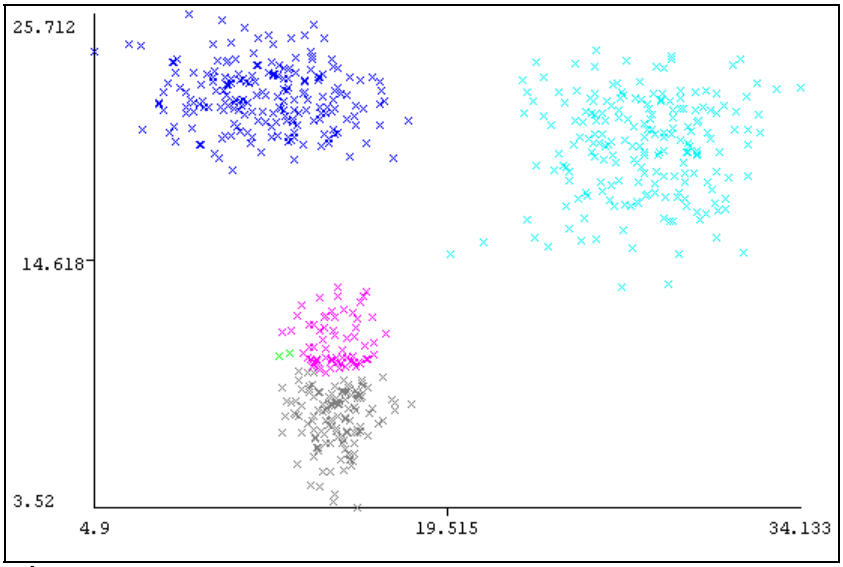


ภาพที่ 22 แสดงผลการแบ่งกลุ่มของ E-Stream ในช่วงที่ 5 (ข้อมูลที่ 4201 – 5000)

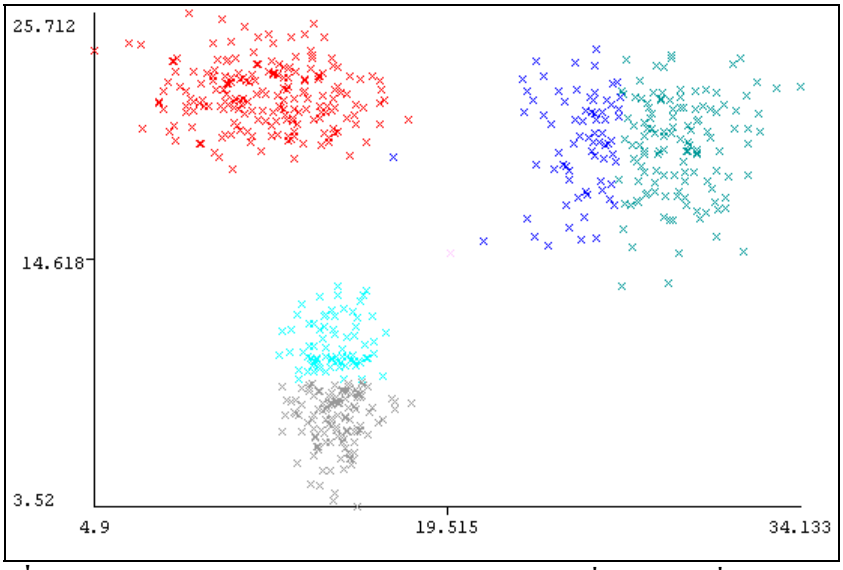


ภาพที่ 23 แสดงผลการแบ่งกลุ่มของ HPStream ในช่วงที่ 5 (ข้อมูลที่ 4201 – 5000)

ในช่วงที่ 5 กลุ่มข้อมูลเลื่อนเข้าหากัน ทั้งสองอัลกอริทึมสามารถรองรับกับการเลื่อนที่ของกลุ่มข้อมูลได้ดี

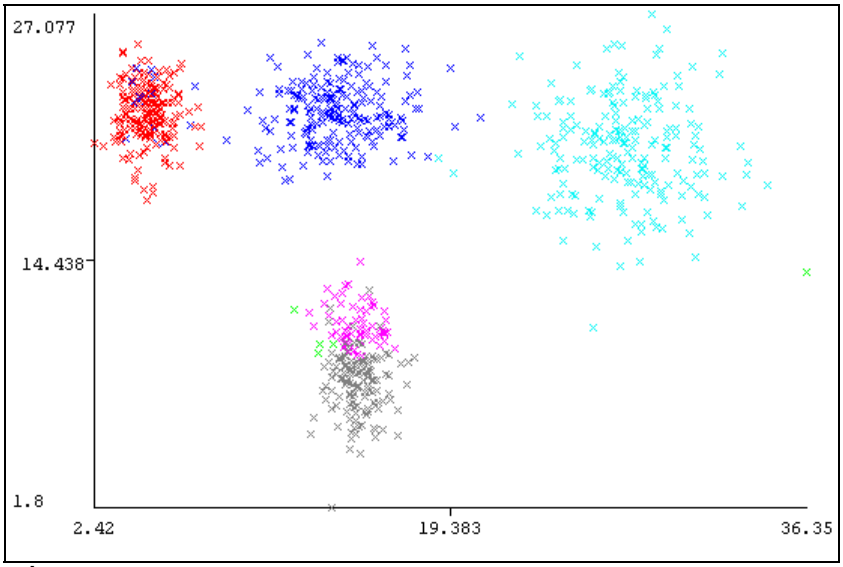


ภาพที่ 24 แสดงผลการแบ่งกลุ่มของ E-Stream ในช่วงที่ 6 (ข้อมูลที่ 5001 – 5600)

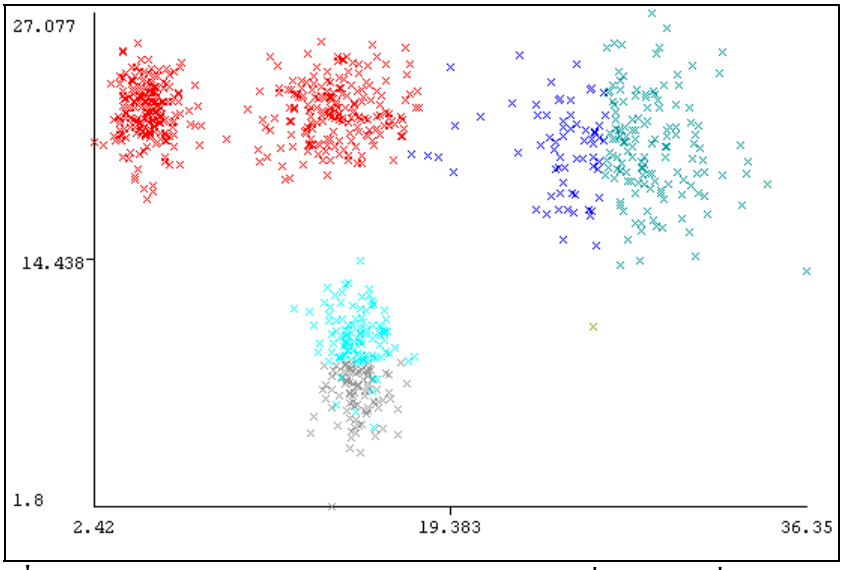


ภาพที่ 25 แสดงผลการแบ่งกลุ่มของ HPStream ในช่วงที่ 6 (ข้อมูลที่ 5001 – 5600)

ในช่วงที่ 6 กลุ่มข้อมูลที่เลื่อนเข้าหากันนั้นถูกรวมเป็นกลุ่มเดียวกัน ทั้งสองอัลกอริทึมยังไม่สามารถรวมกลุ่มข้อมูลที่เข้าใกล้กันได้ แม้ว่าพฤติกรรมจริงของทั้งสองกลุ่มจะใกล้เคียงกันมากก็ตาม

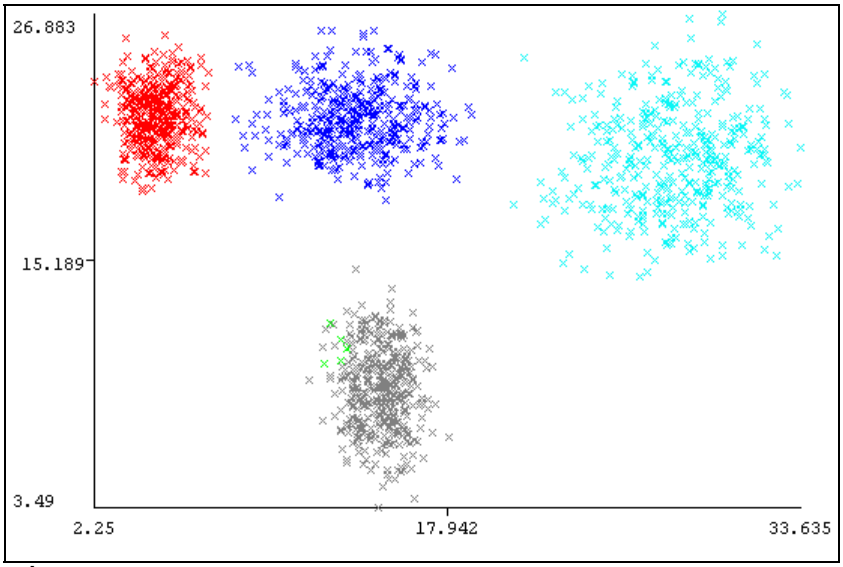


ภาพที่ 26 แสดงผลการแบ่งกลุ่มของ E-Stream ในช่วงที่ 7 (ข้อมูลที่ 5601 – 6400)

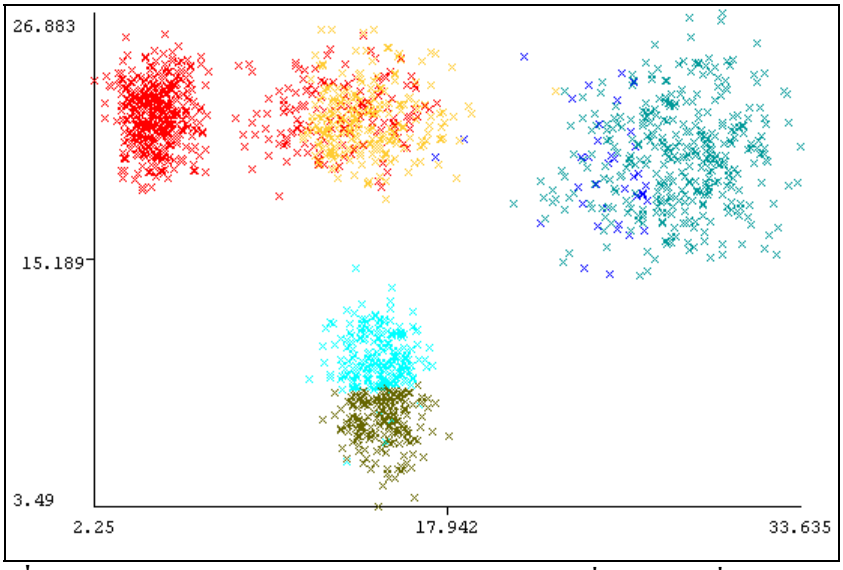


ภาพที่ 27 แสดงผลการแบ่งกลุ่มของ HPStream ในช่วงที่ 7 (ข้อมูลที่ 5601 – 6400)

ในช่วงที่ 7 เกิดกลุ่มข้อมูลใหม่ ซึ่งมีพฤติกรรมแบ่งแยกจากกลุ่มเก่า E-Stream สามารถตรวจพบการเปลี่ยนแปลงนี้และสามารถบ่งบอกกลุ่มใหม่ได้เมื่อได้รับข้อมูลจำนวนหนึ่ง ในขณะที่ HPStream ไม่สามารถแบ่งแยกได้เลย



ภาพที่ 28 แสดงผลการแบ่งกลุ่มของ E-Stream ในช่วงที่ 8 (ข้อมูลที่ 6401 – 8000)



ภาพที่ 29 แสดงผลการแบ่งกลุ่มของ HPStream ในช่วงที่ 8 (ข้อมูลที่ 6401 – 8000)

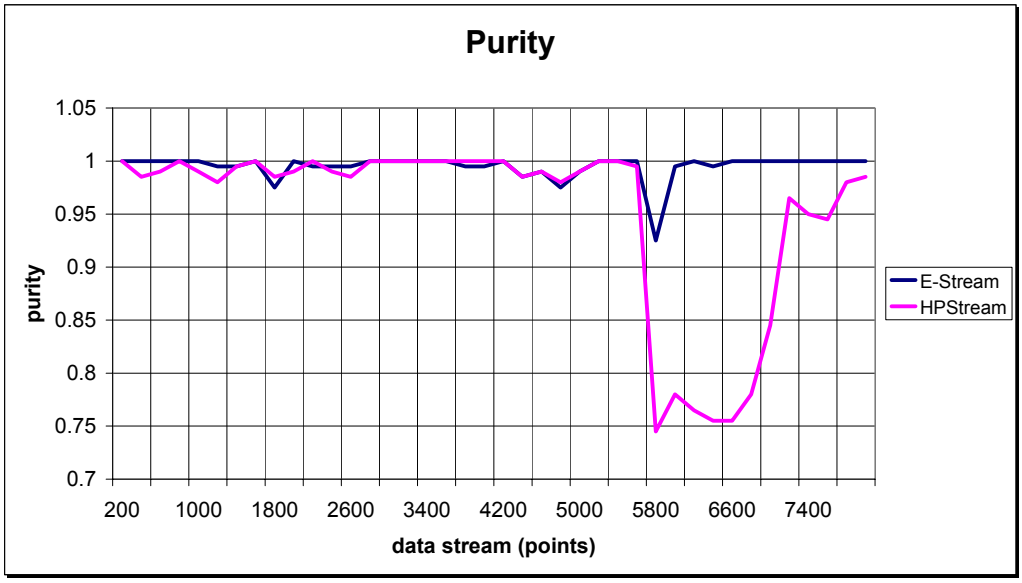
ในช่วงที่ 8 ข้อมูลทั้งหมดมีเสถียรภาพ พบว่า E-Stream นั้นสามารถให้ผลลัพธ์ได้อย่างถูกต้อง กลุ่มข้อมูลทั้งสองกลุ่มที่เลื่อนเข้าหากัน ก็สามารถรวมเข้าเป็นกลุ่มเดียวได้ ถือเป็นการปรับเพื่อเข้ากับการเปลี่ยนแปลงต่างๆได้ในที่สุด ในขณะที่ HPStream ให้ผลลัพธ์ที่ไม่ดีนัก

จากการพิจารณาในแต่ละช่วงสรุปได้ว่า เนื่องจากอัลกอริทึมของ HPStream จำเป็นต้องกำหนดจำนวนกลุ่มให้คงที่ ณ เวลาใดๆ ทำให้ไม่สามารถรองรับการเปลี่ยนแปลงด้านจำนวนกลุ่มได้ และให้ผลลัพธ์ที่ไม่ดีเมื่อจำนวนกลุ่มในขณะนั้นไม่ตรงกับพารามิเตอร์ที่กำหนด ในขณะที่ E-

Stream นั้นสามารถรองรับการเปลี่ยนแปลงทั้ง 5 แบบ ที่กล่าวไว้ตอนต้นได้ แม้ว่าการเปลี่ยนแปลงบางอย่างต้องใช้ข้อมูลจำนวนมากเพื่อตัดสินการเปลี่ยนแปลงนั้น เช่นการรวมกลุ่มที่เกิดขึ้นตั้งแต่ช่วงที่ 6 แต่สามารถรวมได้ถูกต้องในช่วงที่ 8

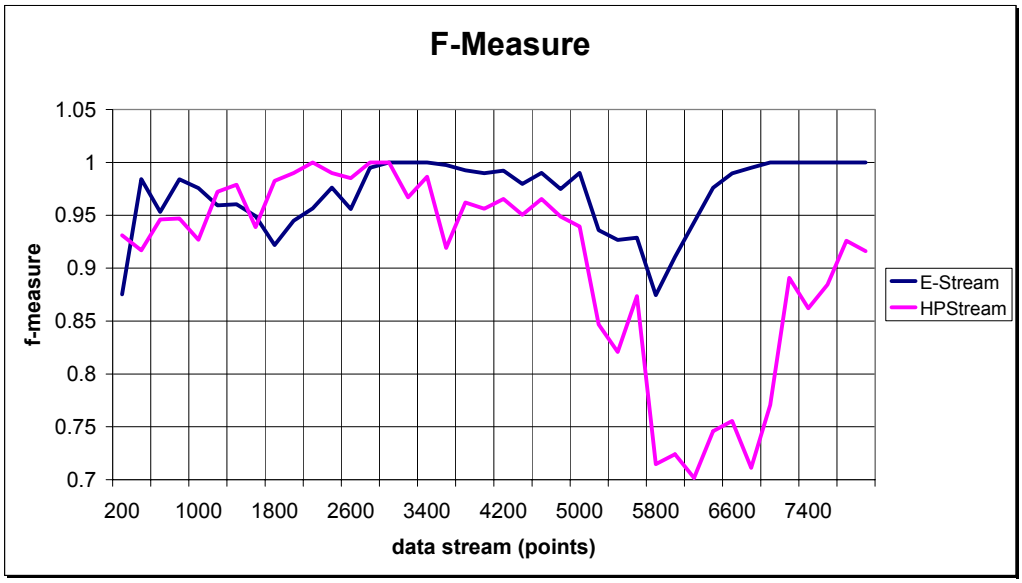
3. การวัดคุณภาพของคลัสเตอร์ที่ได้ด้วยค่าความบริสุทธิ์และค่าเอฟเมเชอร์

3.1 ชุดข้อมูลสังเคราะห์ เราใช้ชุดข้อมูลเดิมจากการทดสอบในการเปลี่ยนแปลงในแต่ละช่วงเวลา โดยใช้ค่าเฉลี่ยทุกๆ 400 ตัว



ภาพที่ 30 กราฟเปรียบเทียบค่าความบริสุทธิ์ของแต่ละอัลกอริทึมโดยใช้ข้อมูลสังเคราะห์

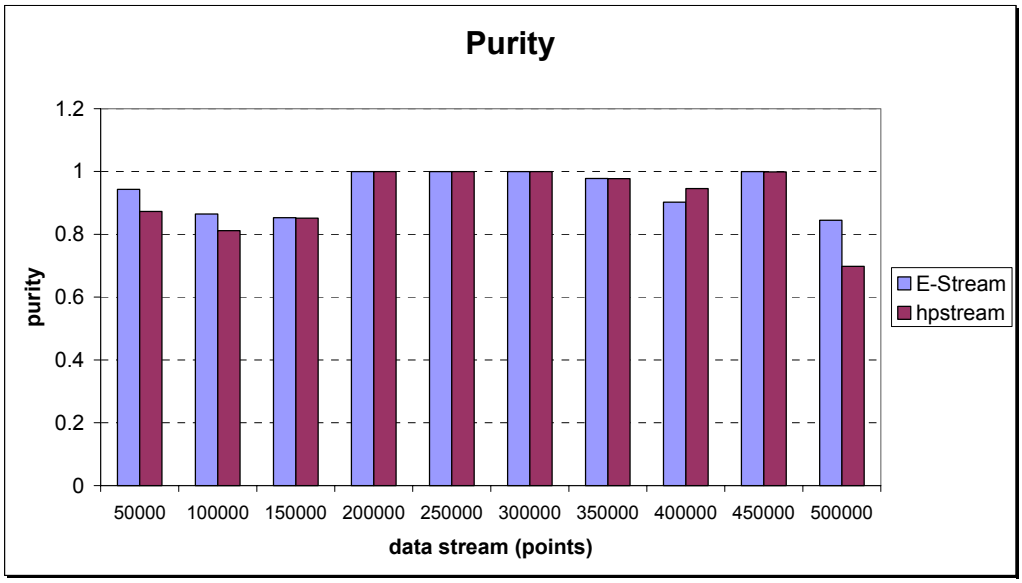
เมื่อพิจารณาค่าความบริสุทธิ์ E-Stream สามารถให้ผลลัพธ์ที่มีคุณภาพที่ดี คือ เกิน 0.9 ตลอดเวลา สามารถกล่าวได้ว่า E-Stream นั้นสามารถรองรับการเปลี่ยนแปลงที่เกิดขึ้นได้ทั้งหมด ในขณะที่ HPStream มีค่าความบริสุทธิ์ตกลงอย่างมากในช่วงการเปลี่ยนแปลงที่ 7 นั่นคือ ช่วงที่ข้อมูลเกิดการแบ่งแยกตัวอย่างชัดเจน และเกิดกลุ่มใหม่ซึ่งแตกแยกกันสองกลุ่ม ซึ่ง HPStream ไม่สามารถแบ่งแยกทั้งสองกลุ่มนั้นได้



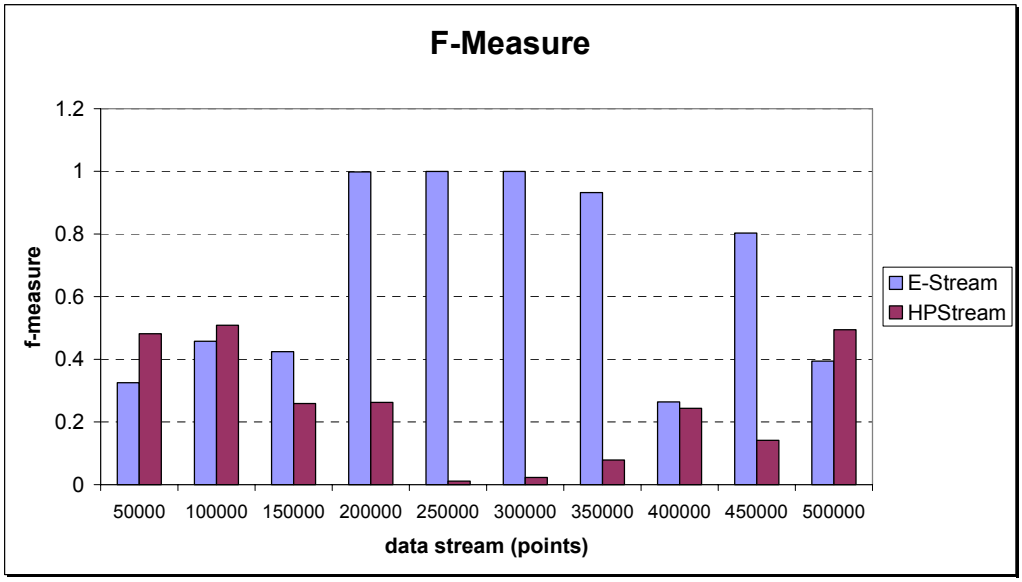
ภาพที่ 31 กราฟเปรียบเทียบค่าเอฟเมเชอร์ของแต่ละอัลกอริทึมโดยใช้ข้อมูลสังเคราะห์

เมื่อพิจารณาค่าเอฟเมเชอร์ E-Stream มีช่วงที่ด้อยกว่า HPStream 2 ช่วง คือ ช่วงเริ่มการทำงาน ซึ่ง E-Stream ต้องอาศัยข้อมูลจำนวนหนึ่งในการสร้างกลุ่มตั้งต้น ในขณะที่ HPStream แม้ว่าจะได้ผลลัพธ์ที่ดีกว่าแต่จำเป็นใช้การทำงานแบบออฟไลน์ และในช่วงการเปลี่ยนแปลงที่ 2 (1601 – 2600) เนื่องจาก E-Stream นั้นมีการรวมกลุ่มที่ผิดพลาดคือรวมกลุ่มใหม่ที่เกิดขึ้นใหม่กับกลุ่มเก่า ส่วนที่ช่วงเวลาอื่นๆ E-Stream ให้ผลลัพธ์ที่ดีกว่า HPStream

3.2 ชุดข้อมูลจริง เราใช้ชุดข้อมูลการตรวจจับการบุกรุกเครือข่าย (KDDCup 1999) โดยจะวัดประสิทธิภาพโดยใช้ค่าเฉลี่ยทุกๆ 50000 ตัว



ภาพที่ 32 กราฟเปรียบเทียบค่าความบริสุทธ์ของแต่ละอัลกอริทึมโดยใช้ข้อมูลจริง

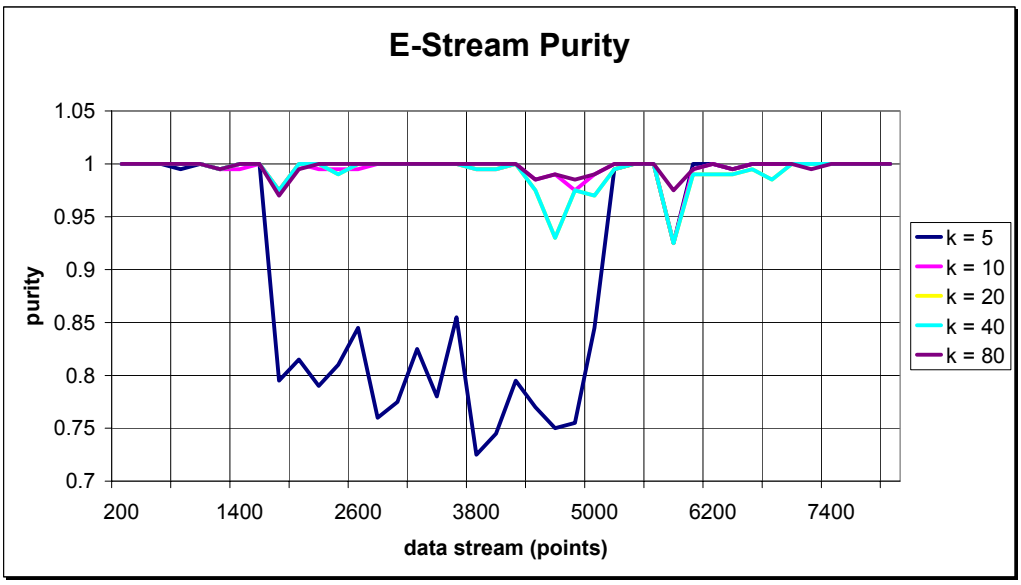


ภาพที่ 33 กราฟเปรียบเทียบค่าเอฟเมเชอร์ของแต่ละอัลกอริทึมโดยใช้ข้อมูลจริง

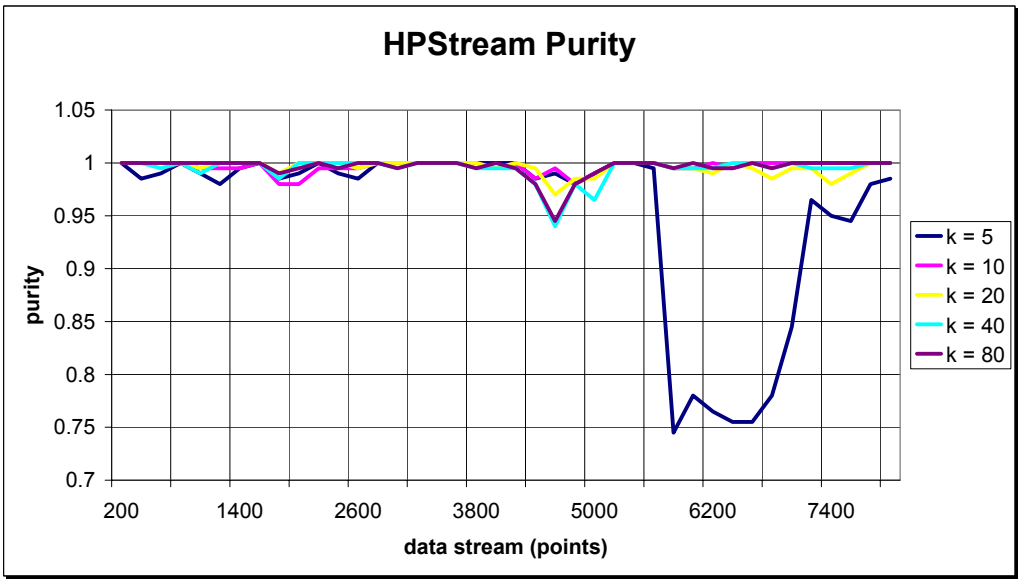
เมื่อเปรียบเทียบโดยใช้ชุดข้อมูลจริง ทั้งสองอัลกอริทึมให้ค่าความบริสุทธ์ที่ไม่ต่างกันมากนัก แต่ค่า F-Measure อัลกอริทึม HPStream ให้ค่าที่ต่ำผิดปกติในช่วง 250000-350000 เนื่องจาก HPStream สร้างคลัสเตอร์ขนาดเล็กเป็นจำนวนมาก ในขณะที่ข้อมูลจริงมีเพียงกลุ่มเดียว ในขณะที่อัลกอริทึม E-Stream ให้ผลลัพธ์ที่ถูกต้อง

4. การวัดความอ่อนไหวต่อพารามิเตอร์จำนวนกลุ่ม

ในการทดลองวัดความอ่อนไหว (Sensitivity) นี้เราได้ใช้ข้อมูลสังเคราะห์ชุดเดิม แต่ได้เปลี่ยนพารามิเตอร์อินพุทจำนวนกลุ่ม เพื่อเปรียบเทียบความอ่อนไหวต่อพารามิเตอร์นี้ โดยได้ปรับค่าเป็น 5, 10, 20, 40, 80 ตามลำดับ

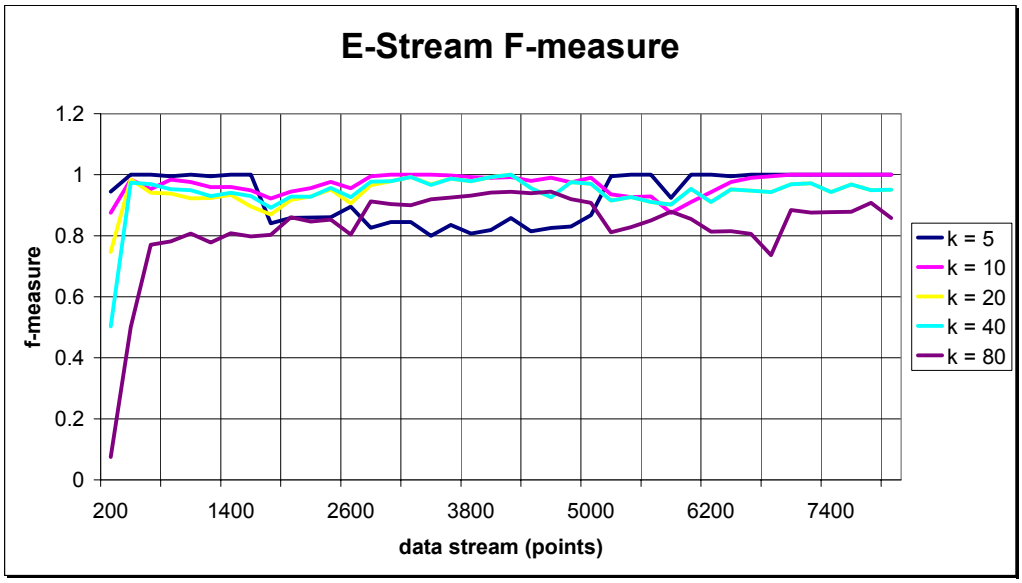


ภาพที่ 34 แสดงผลของพารามิเตอร์จำนวนกลุ่มกับค่าความบริสุทธิ์ของ E-Stream

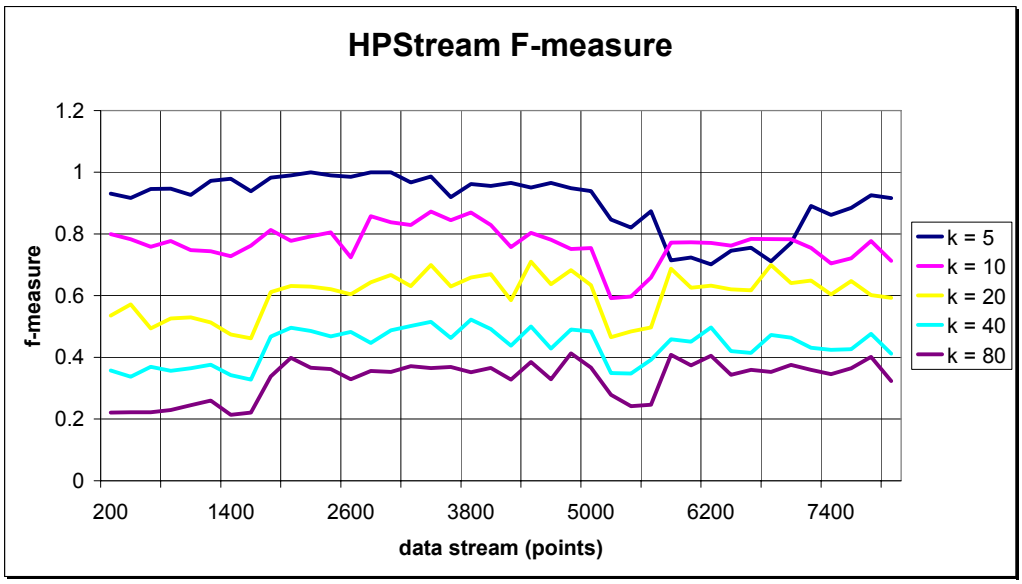


ภาพที่ 35 แสดงผลของพารามิเตอร์จำนวนกลุ่มกับค่าความบริสุทธิ์ของ HPStream

จากภาพที่ 32 และ 33 ทั้งสองอัลกอริทึม ให้ค่าความบริสุทธิ์ตกลงอย่างเห็นได้ชัด ถ้ากำหนดพารามิเตอร์จำนวนกลุ่มน้อยเกินไป แต่จะให้ค่าความบริสุทธิ์ที่ดีถ้ากำหนดพารามิเตอร์ให้มากในระดับเกินจำนวนกลุ่มจริงเล็กน้อย เนื่องจากค่าความบริสุทธิ์มีแนวโน้มมากขึ้นเมื่อเราแบ่งให้จำนวนกลุ่มมากขึ้น แต่ถ้ากำหนดค่าพารามิเตอร์จำนวนกลุ่มให้น้อยเกินไปจะมีผลทำให้การรวมกลุ่มผิดพลาด



ภาพที่ 36 แสดงผลของพารามิเตอร์จำนวนกลุ่มกับค่าเอฟเมเชอร์ของ E-Stream

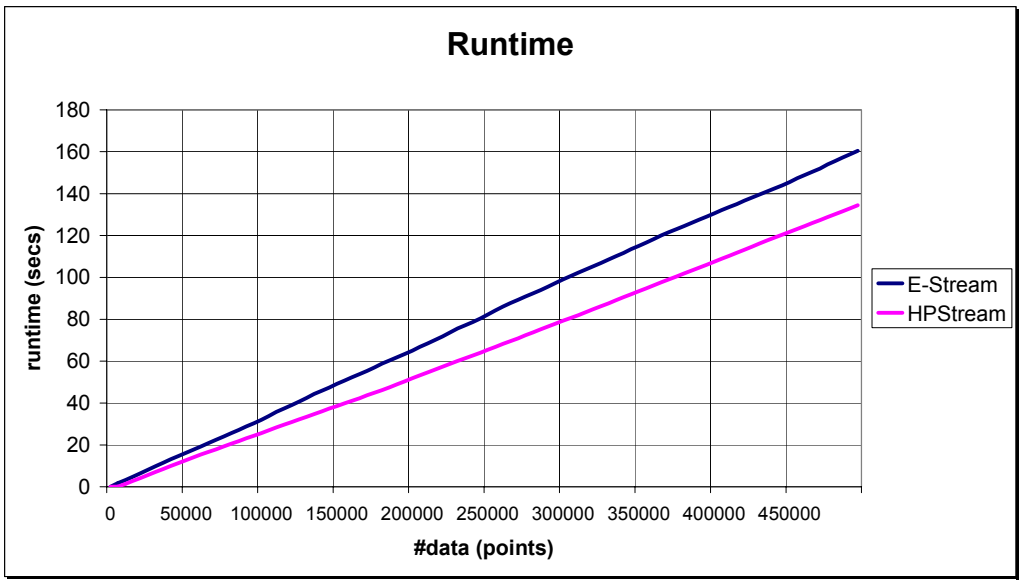


ภาพที่ 37 แสดงผลของพารามิเตอร์จำนวนกลุ่มกับค่าเอฟเมเชอร์ของ HPStream

จากภาพที่ 34 และ 35 ผลการวัดประสิทธิภาพด้วยค่าเอฟเมเชอร์ พบว่าเมื่อกำหนดพารามิเตอร์จำนวนกลุ่ม (k) เกินกว่าจำนวนกลุ่มที่มีอยู่จริงมากขึ้นเรื่อยๆ ประสิทธิภาพการทำงานของ HPStream มีแนวโน้มลดลง ส่วน E-Stream นั้นให้ผลลัพธ์ไม่อ่อนไหวต่อพารามิเตอร์นี้ เนื่องจากการทำงานของ E-Stream ไม่ได้กำหนดจำนวนกลุ่มให้คงที่ตลอดเวลาดัง HPStream แต่จะกำหนดเป็นจำนวนกลุ่มที่มากที่สุดที่ยอมให้เก็บในระบบได้ ซึ่งจำนวนกลุ่มจริงๆ ในขณะหนึ่งของ E-Stream นั้นจะขึ้นกับพฤติกรรมของตัวข้อมูลเอง ถ้ากำหนดพารามิเตอร์นี้ให้มากไว้จะทำให้ E-Stream มีอิสระในการตัดสินการรวมตัว และการแตกตัว และจำนวนกลุ่มที่ได้จะขึ้นกับพฤติกรรมของข้อมูลจริง แต่ถ้ากำหนดพารามิเตอร์นี้น้อยกว่าจำนวนกลุ่มจริงมีผลให้การตัดสินการรวมตัวและแตกตัวจะถูกจำกัดโดยพารามิเตอร์จำนวนกลุ่ม และให้ผลลัพธ์ที่ไม่ดีนัก

5. การเปรียบเทียบเวลาในการทำงานเทียบกับจำนวนข้อมูล

ในการทดลองนี้เราใช้ ข้อมูลจำนวน 2 มิติ ประกอบด้วย 5 กลุ่ม จำนวน 500,000 ตัว

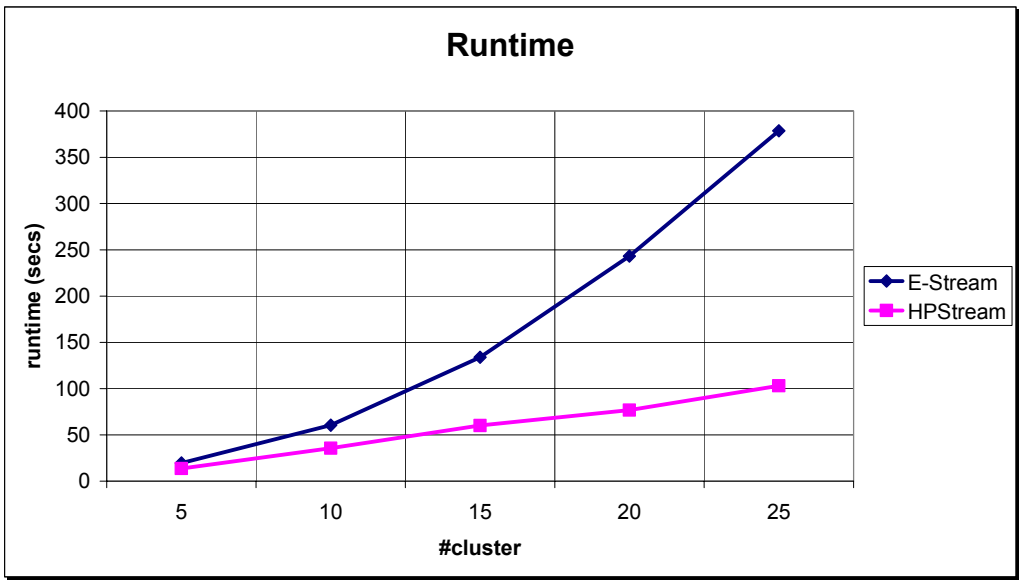


ภาพที่ 38 แสดงเวลาการทำงานเทียบกับจำนวนข้อมูล

การทำงานทั้ง E-Stream และ HPStream ใช้เวลาในการทำงานเป็นเชิงเส้น (linear) เมื่อเทียบกับจำนวนข้อมูล ซึ่งตามเงื่อนไขของสตรีมจำเป็นต้องใช้เวลาในการทำงานเป็นเชิงเส้นเมื่อเทียบกับจำนวนข้อมูล โดยที่ E-Stream ช้ากว่าด้วยค่าคงที่เล็กน้อย

6. การเปรียบเทียบเวลาในการทำงานเทียบกับจำนวนกลุ่มของข้อมูล

สำหรับการวัดความเวลาในการทำงานเมื่อเทียบกับจำนวนกลุ่ม เราใช้ชุดข้อมูลขนาด 2 มิติ จำนวน 100,000 ตัว และทดลองเปลี่ยนแปลงจำนวนกลุ่มของข้อมูลเป็น 5, 10, 15, 20 และ 25 ตามลำดับ

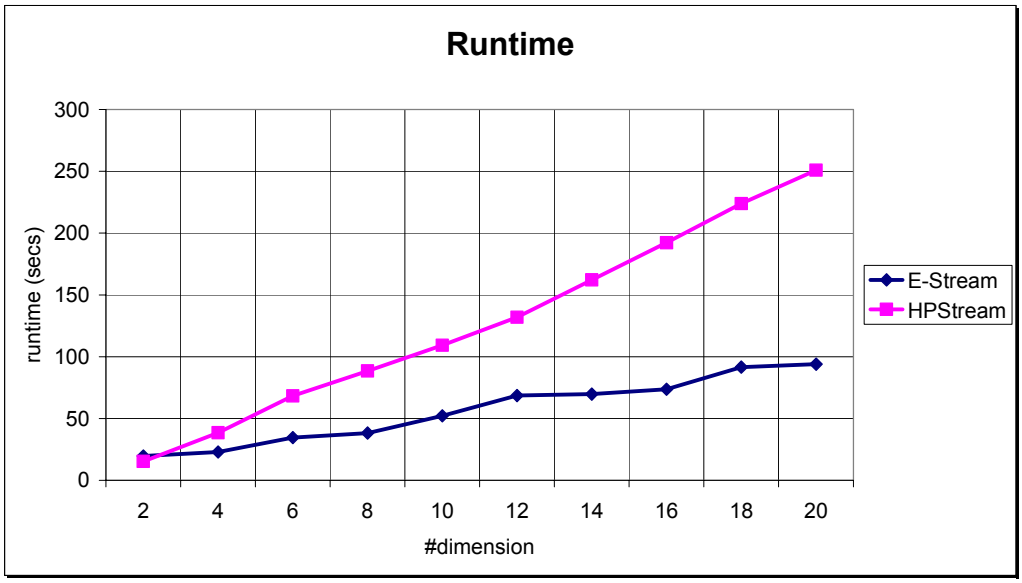


ภาพที่ 39 แสดงเวลาการทำงานเทียบกับจำนวนกลุ่มของข้อมูล

จากการทดลองพบว่า E-Stream ใช้เวลาเป็นโพลิโนเมียลเมื่อเทียบกับจำนวนกลุ่มของข้อมูล ในขณะที่ HPStream ใช้เวลาเป็นเชิงเส้น เนื่องจาก E-Stream ใช้เวลาเป็น $O(k^2)$ เมื่อ k คือ จำนวนคลัสเตอร์ ในการหาคู่ของกลุ่มข้อมูลที่ใกล้ที่สุดเพื่อตรวจสอบการรวมตัวของกลุ่มข้อมูล ในทุกรอบการทำงาน

7. การเปรียบเทียบเวลาในการทำงานเทียบกับจำนวนมิติของข้อมูล

สำหรับการวัดเวลาในการทำงานเมื่อเทียบกับจำนวนมิติ เราใช้ชุดข้อมูลที่มี 5 กลุ่ม จำนวน 100,000 ตัว และทดลองเปลี่ยนแปลงจำนวนมิติของข้อมูลเป็น 2, 4, 6, 8, 10, 12, 14, 16, 18 และ 20 ตามลำดับ



ภาพที่ 40 แสดงเวลาการทำงานเทียบกับจำนวนมิติของข้อมูล

จากการทดลองพบว่าทั้งสองอัลกอริทึมใช้เวลาการทำงานเป็นเชิงเส้น เมื่อเทียบกับจำนวนมิติของข้อมูล โดยที่ HPStream ใช้เวลามากกว่าประมาณ 2 เท่า เนื่องจาก HPStream มีขั้นตอนการคำนวณเพื่อหามิติที่เป็นตัวแทนคลัสเตอร์จึงใช้เวลามากกว่า E-Stream

สรุปและข้อเสนอแนะ

สรุป

งานชิ้นนี้ได้เสนอเทคนิคการแบ่งกลุ่มกระแสข้อมูล E-Stream ซึ่งสามารถรองรับการเปลี่ยนแปลงพฤติกรรมต่างๆของข้อมูล คือ การเกิดขึ้นของกลุ่มข้อมูล การหายไปของกลุ่มข้อมูล การเลื่อนที่ของกลุ่มข้อมูล การรวมตัวกันของกลุ่มข้อมูล และการแยกตัวของกลุ่มข้อมูลได้ ซึ่งทำให้การแบ่งกลุ่มกระแสข้อมูลให้ผลลัพธ์ที่มีคุณภาพมากขึ้น และผลลัพธ์ที่ได้มีความทันสมัยตลอดเวลา โดยเราได้เปรียบเทียบประสิทธิภาพของอัลกอริทึมใน 2 ด้าน คือ

1. ด้านคุณภาพของคลัสเตอร์ สำหรับข้อมูลที่มีการเปลี่ยนแปลง E-Stream ให้คุณภาพของคลัสเตอร์ที่ดีกว่า HPStream อย่างเห็นได้ชัด เนื่องจากสามารถรองรับการเปลี่ยนแปลงต่างๆ ได้ตลอดเวลา อีกทั้งยังไม่อ่อนไหวต่อค่าพารามิเตอร์อินพุทจำนวนกลุ่ม ซึ่งเป็นปัญหาสำหรับอีกหลายๆ อัลกอริทึมอีกด้วย

2. ด้านเวลาการทำงาน การวิเคราะห์เพื่อหาการเปลี่ยนแปลงนั้นจำเป็นต้องใช้การคำนวณที่มากขึ้น โดยได้ใช้เวลามากในการตรวจสอบการแตกตัว และการตรวจสอบการรวมตัว โดยเฉพาะสำหรับการรวมตัว E-Streamใช้เวลาเป็นโพลิโนเมียล $O(k^2)$ เมื่อ k คือจำนวนกลุ่มของข้อมูล เนื่องจากต้องคำนวณหาคู่ของคลัสเตอร์ที่คล้ายคลึงกันเพื่อตรวจสอบการรวมตัวของกลุ่มข้อมูล

ข้อเสนอแนะ

1. ความขัดแย้งของเงื่อนไขการรวมตัวและแยกตัว งานวิจัยนี้มุ่งเน้นไปที่การหาโครงสร้างการทำงานที่สามารถรองรับการเปลี่ยนแปลงของข้อมูลได้ และพบว่าควรมีการตรวจสอบการรวมตัว และแยกตัวกันของกลุ่มข้อมูล ตลอดเวลา จึงได้เสนอวิธีที่ง่ายและเพียงพอต่อการทำงานโดยการรวมตัว จะพิจารณาจากค่าระยะห่างระหว่างกลุ่มข้อมูล แต่การแตกตัวพิจารณาจากฮิสโทแกรม ซึ่งทำให้เกิดปัญหาเงื่อนไขขัดแย้งกันเช่น บางคลัสเตอร์ตรวจพบการแตกตัวและรวมตัวพร้อมกัน เนื่องจากเงื่อนไขการตรวจสอบการรวมตัวและการแยกตัวนี้ไม่สอดคล้องกันเท่าที่ควร

การแก้ไขปัญหานี้อาจทำได้โดย ปรับปรุงให้เงื่อนไข ในการรวมตัว และแตกตัว ให้ใช้ตัววัดค่าแบบเดียวกันเช่น พิจารณาจาก ค่าระยะห่างเหมือนกัน พิจารณาจากฮิสโทแกรมเหมือนกัน หรือประยุกต์รวมทั้งสองวิธีเข้าด้วยกัน ซึ่งต้องการการวิจัยต่อไป

2. แนวทางการวิจัยของการแบ่งกลุ่มกระแสข้อมูลนั้น ได้แบ่งออกเป็น 2 แนว หลักๆ คือ การแบ่งกลุ่มเพื่อกระแสข้อมูล เพื่อพิจารณาลักษณะของกลุ่มข้อมูลที่ได้ และการแบ่งกลุ่มกระแสข้อมูลเพื่อตรวจจับการเปลี่ยนแปลงพฤติกรรมของข้อมูล ซึ่งงานวิจัยชิ้นนี้จัดอยู่ในแนวทางแรก การเปลี่ยนทิศทางงานวิจัยไปเป็นแนวทางหลัง เป็นสิ่งหนึ่งที่สามารถพัฒนาต่อไปได้

เอกสารและสิ่งอ้างอิง

สมพล ตันติคุณ. 2549. **ระบบจำลองการตรวจจับการบุกรุกเครือข่ายด้วยการแบ่งกลุ่มกระแสข้อมูล**. โครงงานวิศวกรรมคอมพิวเตอร์, มหาวิทยาลัยเกษตรศาสตร์

Aggarwal, C.C., J. Han, J. Wang and P.S. Yu. 2003. A Framework for Clustering Evolving Data Streams, *In* **Proceeding of the 29th VLDB conference**.

Aggarwal, C.C., J. Han, J. Wang and P.S. Yu. 2004. A Framework for Projected Clustering of High Dimensional Data Streams, *In* **Proceeding of the 30th VLDB conference**.

Barbara, D. 2002. Requirements for Clustering Data Streams, *In* **ACM SIGKDD Explorations Newsletter**.

Breunig, M.M., H. P. Kriegel, P. Kroger and J. Sander. 2001. Data Bubbles: Quality Preserving Performance Boosting for Hierarchical Clustering, *In* **ACM SIGMOD record**.

Gaber, M.M., A. Zaslavsky and S. Krishnaswmy. 2005. Mining Data Streams: A Review, *In* **SIGMOD Record**.

Guha, S., A. Meyerson, N. Mishra, R. Motwani and L. O'Callaghan. 2003. Clustering Data Streams: Theory and Practice, *In* **IEEE Transactions on Knowledge and Data Engineering**.

Milenova, L.B. and M.M. Campos. 2003. Clustering Large Databases with Numeric and Nominial Values Using Orthogonal Projections, *In* **Proceedings of the 29th VLDB Conference**.

Oh, S.H., J.S. Kang, Y.C. Byun, G.L. Park and S.Y. Byun. 2005. Intrusion Detection based on Clustering a Data Stream, *In* **Proceedings of the 2005 Third ACIS International Conference on Software Engineering Research, Management and Applications**.

Song, M. and H. Wang. 2005. Highly Efficient Incremental Estimation of Gaussian Miture Models for Online Data Stream Clustering, *In* **SPIE Conference on Intelligent Computing: Theory And Application III**.

Udommanetanakit, K., T. Rakthanmanon and K. Waiyamai. 2006. A Novel Approach for Improving Stream Clustering Technique, *In* **10th Annual National Synposium on Computational Science & Engineering**.

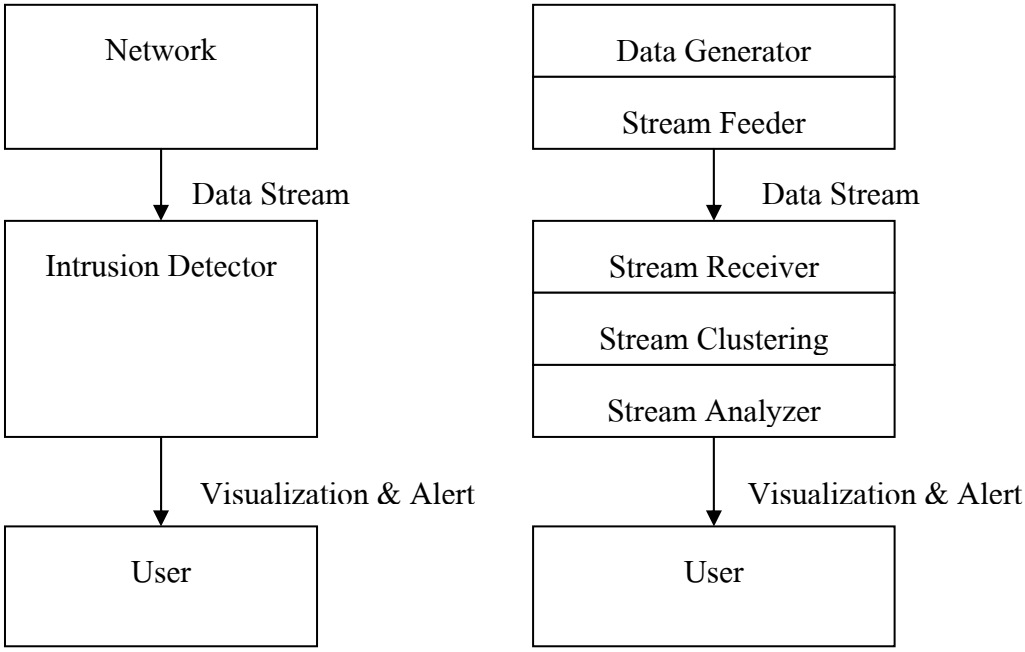
Zhang, T., R. Ramakhrisnan and M. Livny. 1996. BIRCH: An Efficient Data Clustering Method for Very Large Databases, *In* **Proceeding of ACM SIGMOD International Conference on Management of Data**.

ภาคผนวก

โปรแกรมจำลองการตรวจจับการบุกรุกเครือข่าย

โปรแกรมนี้ได้ถูกพัฒนาควบคู่ไปกับงานวิจัยชิ้นนี้ โดยตัวโปรแกรมได้ใช้เทคนิคการแบ่งกลุ่มกระแสข้อมูลมาช่วยในการตรวจสอบการบุกรุก (สมพล, 2549) การใช้โปรแกรมนี้ควบคู่ไปกับการพัฒนาการแบ่งกลุ่มกระแสข้อมูลจะทำให้สามารถตรวจสอบ และพัฒนาได้ง่ายขึ้น เนื่องจากมีส่วนแสดงผลซึ่งทำให้เราสามารถเห็นการเปลี่ยนแปลงของรูปแบบของกลุ่มข้อมูลที่ได้ตลอดเวลา

ภาพรวมของระบบ



ภาพผนวกที่ 1 บล็อกไดอะแกรมแสดงโครงสร้างระบบ

ตัวโปรแกรมจะจำลองมาจากสถาณการณ์จริง ซึ่งกระแสข้อมูลจะกำเนิดจากระบบเครือข่าย (Network) และมีระบบตรวจจับการบุกรุกซึ่งนำประมวลผลกระแสข้อมูล (Intrusion Detector) และส่งผลการทำงานไปให้ผู้ใช้ (User) ซึ่งโปรแกรมนี้จะมีส่วนสร้างกระแสข้อมูล (Data Generator) คอยจำลองการกำเนิดข้อมูลจากเครือข่าย และระบบจ่ายข้อมูล (Stream Feeder) เป็นตัวจ่ายข้อมูลส่วน ระบบตรวจจับก็จะจำลองโดยมีระบบรับข้อมูล (Stream Receiver) และระบบแบ่งกลุ่มกระแสข้อมูล (Stream Clustering) เพื่อตรวจสอบการบุกรุก เมื่อระบบ Stream Clustering ทำการจัดกลุ่ม

ข้อมูลเสร็จแล้วจะส่งผลการแบ่งกลุ่มกระแสข้อมูลให้กับระบบวิเคราะห์ข้อมูล (Stream Analyzer) เพื่อทำการวิเคราะห์ข้อมูลและแสดงผลให้ผู้ใช้

โปรแกรมนี้แบ่งออกเป็นระบบย่อย 4 ระบบ คือ ระบบสร้างข้อมูล ระบบจ่ายและรับข้อมูล ระบบแบ่งกลุ่มกระแสข้อมูล และระบบแสดงผลข้อมูล

1. ระบบสร้างข้อมูล (Data Generator) เป็นตัวสร้างกระแสข้อมูลโดยมีสมมติว่าการกระจายของข้อมูลเป็นแบบปกติ (Normal Distribution) มีความสามารถต่างๆ ดังนี้

1.1 สามารถระบุจำนวนกลุ่ม จำนวนมิติ ตำแหน่งของกลุ่ม และค่าเบี่ยงเบนมาตรฐานได้

1.2 สามารถสร้างข้อมูลบุกรุกได้

1.3 สามารถเปลี่ยนลักษณะของข้อมูลได้ เช่น บางกลุ่มเกิดขึ้น บางกลุ่มหายไป เปลี่ยนตำแหน่งของกลุ่ม เปลี่ยนลักษณะการกระจายตัว

2. ระบบจ่ายข้อมูลและระบบรับข้อมูล (Stream Feeder & Stream Receiver) มีไว้เพื่อจำลองการรับและส่งข้อมูลในรูปแบบของกระแส โดยมีความสามารถต่างๆ ดังนี้

2.1 สามารถกำหนดอัตราเร็วของการจ่ายข้อมูลได้

2.2 สามารถแจ้งเตือนได้เมื่อบัฟเฟอร์ของส่วนรับเต็ม เกิดจากระบบแบ่งกลุ่มกระแสข้อมูลไม่สามารถประมวลผลได้ทัน

3. ระบบแบ่งกลุ่มกระแสข้อมูล (Stream Clustering) เป็นตัวร้องขอข้อมูลจาก ระบบรับข้อมูล เพื่อทำการแบ่งกลุ่มกระแสข้อมูล และส่งผลลัพธ์ให้ ระบบวิเคราะห์ข้อมูล ต่อไป

4. ระบบวิเคราะห์ข้อมูล (Stream Analyzer) เป็นตัววิเคราะห์กลุ่มข้อมูลและแสดงผลลัพธ์ให้แก่ผู้ใช้งานซึ่ง

4.1 สามารถแสดงผลแบบสแก๊ตเตอร์พล็อต (Scatter Plot)

4.2 สามารถแสดงผลแบบกลุ่มข้อมูลและรัศมีใน 2 มิติ

4.2 สามารถแจ้งเตือนผู้ใช้เมื่อเกิดข้อมูลบุกรุกขึ้น

หน้าจอสำหรับติดต่อผู้ใช้ (User Interface)

โปรแกรมนี้มีหน้าจอติดต่อกับผู้ใช้ทั้งหมด 3 tab คือ Add group, Visualization และ Config ซึ่งจะอธิบายถึงการใช้งานในแต่ละ tab

1. Add group มีลักษณะดังภาพผนวกที่ 2 มีส่วนประกอบต่างๆ ดังนี้

1.1 Textbox สำหรับใส่จำนวน attribute ที่จะใช้ในการสร้างข้อมูล และวิเคราะห์ข้อมูล

1.2 Button สำหรับตกลงเลือกจำนวน attribute ที่ใส่ใน TextBox 1

1.3 กลุ่มของ Textbox ที่ไว้ใส่ค่าจุดศูนย์กลางและค่า SD ของแต่ละ attribute ใน 1 กลุ่ม

1.4 Button สำหรับทำการเพิ่มกลุ่มข้อมูล ที่ได้ใส่ค่าไว้ในข้อ 3 เมื่อกดปุ่มนี้ 1 ครั้ง จะทำการเพิ่มกลุ่มของข้อมูลเพิ่มขึ้น 1 กลุ่ม และทำการ random ค่าจุดศูนย์กลางและค่า SD ของกลุ่มใหม่ขึ้นมาให้โดยอัตโนมัติ

1.5 กลุ่มของ Textbox ที่ไว้ใส่ค่าจุดศูนย์กลางและค่า SD ของแต่ละ attribute ในกลุ่มที่ต้องการแก้ไข

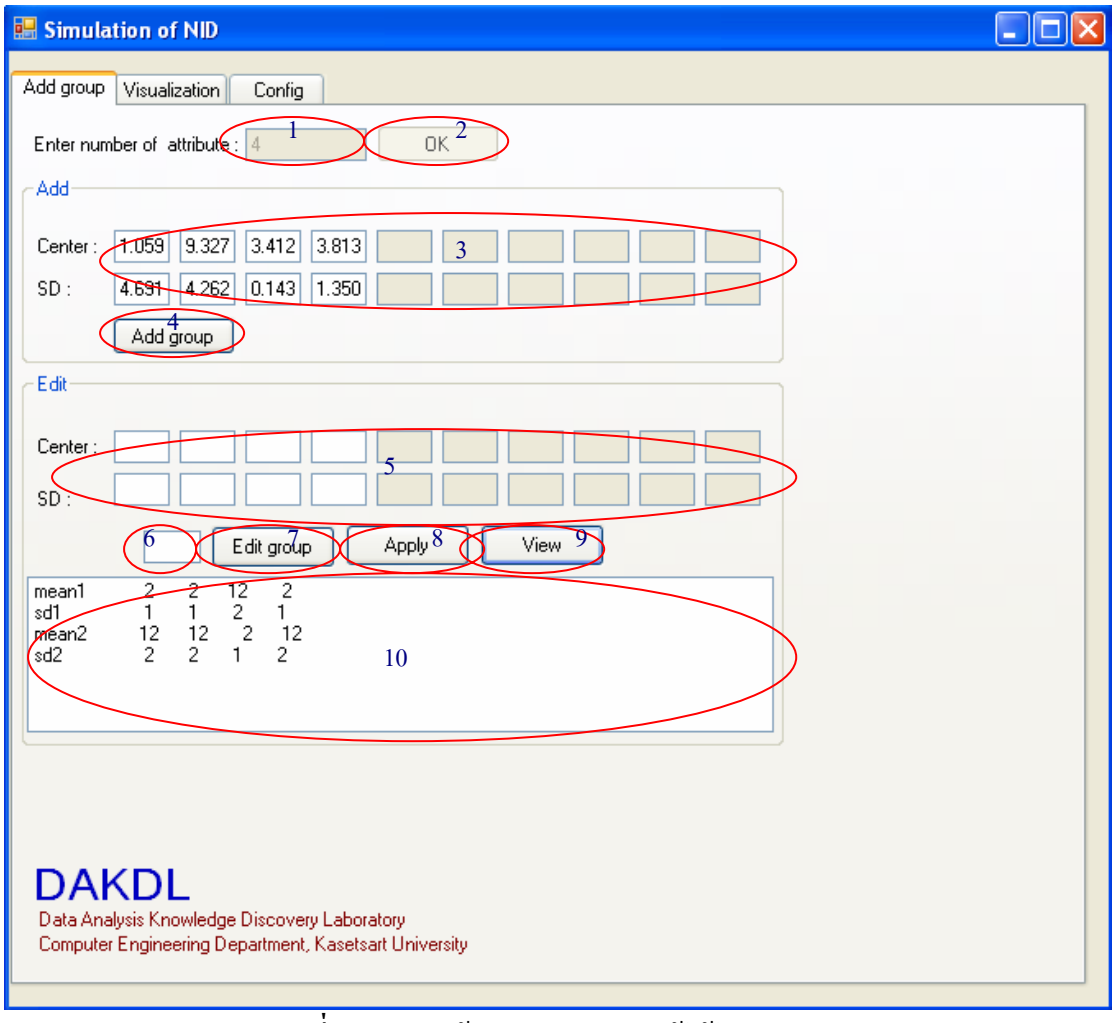
1.6 Texbox ที่ไว้ใส่ค่าตัวเลขของกลุ่มที่ต้องการแก้ไข

1.7 Button สำหรับกดตกลงเลือกแก้ไขกลุ่มที่ใส่ค่าไว้ใน Textbox ในข้อ 6 และแสดงค่าของกลุ่มนั้นในกลุ่มของ Textbox ในข้อ 5

1.8 Button สำหรับกดตกลงแก้ไขกลุ่มข้อมูลตามค่าที่ใส่ไว้ในกลุ่มของ Textbox ในข้อ 5

1.9 Button สำหรับกดเรียกดูข้อมูลที่ได้ทำการเพิ่มไปแล้ว (ข้อมูลแสดงในListbox ในข้อ 10)

1.10 Listbox สำหรับแสดงค่าของกลุ่มข้อมูลที่ได้ทำการเพิ่มไปแล้ว



ภาพผนวกที่ 2 แสดงหน้าจอสำหรับติดต่อผู้ใช้ในส่วน Add group

2. Visualization มีลักษณะดังภาพผนวกที่ 3 มีส่วนประกอบต่างๆ ดังนี้

2.1 Textbox เก็บ path ของ textfile ที่ใช้ในการวิเคราะห์ข้อมูล

2.2 Button สำหรับ Browse หาไฟล์ที่ต้องการ

2.3 Button สำหรับเริ่มทำการวิเคราะห์ข้อมูลจาก textfile ที่ได้เลือกไว้

2.4 Button สำหรับเริ่มทำการวิเคราะห์ข้อมูลจากข้อมูล

2.5 Label แสดงจำนวนบรรทัด ที่ได้ทำการวิเคราะห์ข้อมูลไปแล้ว

2.6 Label แสดงขนาดของคิว (Queue) ณ เวลาขณะนั้น (เป็น 1 แสดงว่าเมื่อระบบ Stream Feeder ส่งข้อมูลลงบัฟเฟอร์แล้ว ระบบ Stream Clustering จะเรียกข้อมูลจาก Stream Feeder ได้เลย ที่เป็น 1 เพราะว่าระบบ Stream Feeder ส่งข้อมูลลงบัฟเฟอร์ตลอดเวลา ทำให้เหลือข้อมูลค้างอยู่ใน quene 1 บรรทัด เสมอ)

2.7 Label บอกถึงสถานะขณะนั้น ว่าโปรแกรมทำอะไรอยู่

2.8 Combobox สำหรับเลือก attribute ที่จะแสดงผลในแกน x

2.9 Combobox สำหรับเลือก attribute ที่จะแสดงผลในแกน y

2.10 กราฟแสดงจุดโคออดิเนตของข้อมูลที่ Feed ไปแล้ว

2.11 กราฟแสดงกลุ่มของข้อมูลที่ได้จากการทำการแบ่งกลุ่มกระแสข้อมูลแล้ว

2.12 listbox แสดงค่าของข้อมูลที่ระบบทำการตรวจจับเป็นข้อมูลบุกรุก

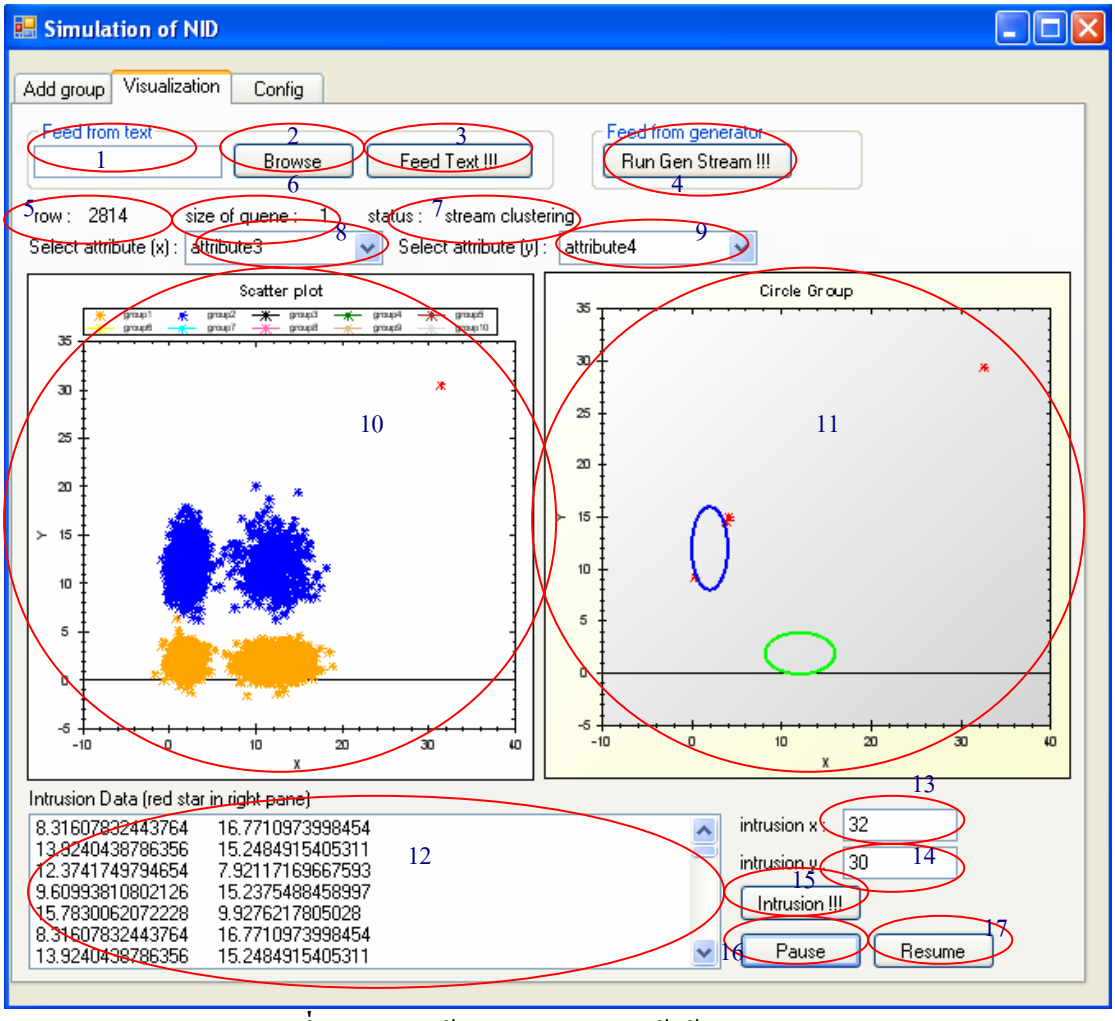
2.13 Textbox สำหรับใส่ค่าของข้อมูลบุกรุกในแกน x

2.14 Textbox สำหรับใส่ค่าของข้อมูลบุกรุกในแกน y

2.15 Button สำหรับสร้างข้อมูลบุกรุกที่ได้ตั้งค่าไว้ใน Textbox ข้อ 13 และ ข้อ 14

2.16 Button สำหรับหยุดการทำงานของโปรแกรมชั่วขณะ

2.17 Button สำหรับสั่งให้โปรแกรมทำงานต่อ



ภาพผนวกที่ 3 แสดงหน้าจอสำหรับติดต่อผู้ใช้ในส่วน Visualization

3. Config มีลักษณะดังภาพผนวกที่ 4 มีส่วนประกอบต่างๆ ดังนี้

3.1Button สำหรับกดตกลงใช้ค่าที่ได้ตั้งค่าไว้ทางด้านซ้าย

3.2 Button สำหรับกลับคืนสู่ค่า default โดยค่า default จะแสดงใน Textbox ด้านซ้ายมือ (ค่าจะเปลี่ยนเป็น default ก็ต่อเมื่อกด Button OK ในข้อ 1)

3.3 Button สำหรับเรียกดูค่า ณ ขณะนั้น โดยค่า ณ ขณะนั้น จะแสดงใน Textbox ด้านซ้ายมือ

3.4 num_data จำนวนบรรทัดของข้อมูลทั้งหมดที่จะทำการวิเคราะห์ (default เป็น 2000)

3.5 num_group จำนวนกลุ่มของข้อมูลที่มากที่สุดที่เป็นไปได้ (default เป็น 10)

3.6 algorithm อัลกอริทึมที่ใช้ในการวิเคราะห์ (default เป็น 2 และในขณะนี้ใช้ได้เพียงอัลกอริทึมเดียวเท่านั้น)

3.7 clustering ค่ากำหนดว่าให้ทำการแบ่งกลุ่มหรือไม่ (default เป็น 1)

3.8 evaluate ค่ากำหนดว่าให้ทำการประเมินผลหรือไม่ (deault เป็น 1)

3.9 num_initial จำนวนข้อมูลการหากลุ่มตั้งต้น (default เป็น 0 และขณะนี้ยังไม่สามารถใช้งานได้)

3.10 decay_rate อัตราการเลือนหายของข้อมูล (deault เป็น 0.1)

3.11 dimension_factor จำนวนมิติที่ใช้ในการแบ่งกลุ่ม (deault เป็น 20 และขณะนี้ยังไม่สามารถใช้งานได้)

3.12 radius_factor จำนวนเท่าของรัศมีที่ยอมรับได้ (deault เป็น 3)

3.13 first_position ตำแหน่งเริ่มต้นในการประเมินผล(deault เป็น 0)

3.14 stream_speed ความเร็วของจำนวนบรรทัดของข้อมูลต่อวินาที(default เป็น 10)

3.14 horizon จำนวนข้อมูลสำหรับการประเมินผลในแต่ละช่วง(deault เป็น 500)

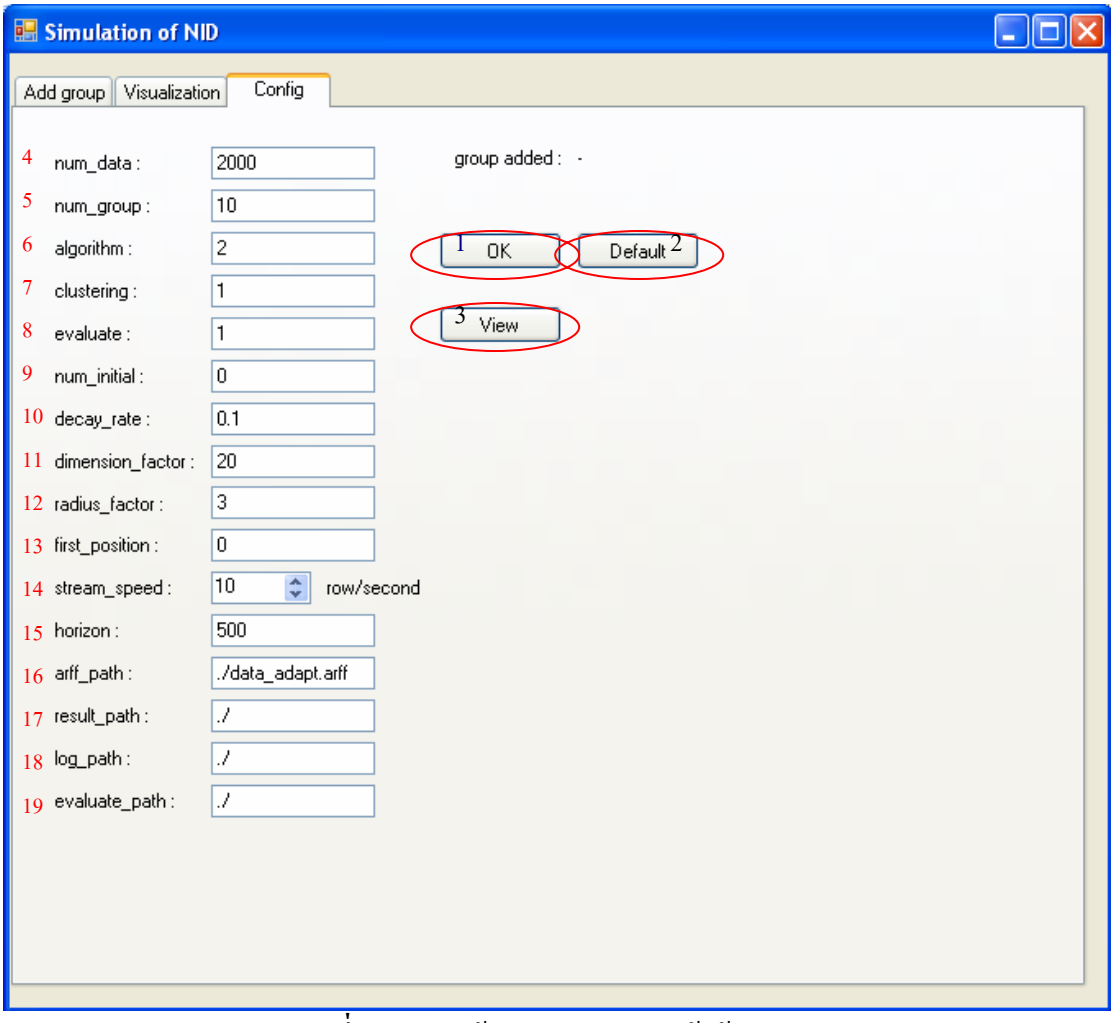
3.15 arff_path พาธที่เก็บข้อมูลอินพุทในรูปแบบของ arff (deault เป็น ./data_adapt.arff)

3.16 result_path พาธที่เก็บของผลลัพธ์การแบ่งกลุ่ม(deault เป็น ./)

3.17 log_path พาธที่เก็บลอกไฟล์ (deault เป็น ./)

3.18 evaluate_path พาธที่เก็บผลลัพธ์การประเมินผล(deault เป็น ./)

3.19 group_added จำนวนกลุ่มที่ได้ทำการเพิ่มไปแล้วในหน้า add group



ภาพผนวกที่ 4 แสดงหน้าจอสำหรับติดต่อผู้ใช้ในส่วน Config

ประวัติการศึกษา และการทำงาน

ชื่อ –นามสกุล	นายคมกริช อุดมมณีธนกิจ
วัน เดือน ปี ที่เกิด	วันที่ 28 ตุลาคม 2524
สถานที่เกิด	ระนอง
ประวัติการศึกษา	วศ.บ. (ไฟฟ้า) มหาวิทยาลัยธรรมศาสตร์
ตำแหน่งหน้าที่การงานปัจจุบัน	
สถานที่ทำงานปัจจุบัน	
ผลงานดีเด่นและรางวัลทางวิชาการ	
ทุนการศึกษาที่ได้รับ	