วิทยานิพนธ์

การเพิ่มประสิทธิภาพเทคนิคการจำแนกประเภทข้อมูลโดยใช้หลายกฎความสัมพันธ์แบบกระชับสมบูรณ์

Improving Associative Classification Technique by Using Multiple Essential Class-Association Rules

นายวีระพล หาญโชติช่วง

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

พ.ศ. 2549

ใบรับรองวิทยานิพนธ์

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

วิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมคอมพิวเตอร์)

ปริญญา

วิศวกรรมคอมพิวเตอร์	วิศวกรรมคอมพิวเตอร์
สาขา	**ภาควิชา**

เรื่อง การเพิ่มประสิทธิภาพเทคนิคการจำแนกประเภทข้อมูลโดยใช้หลายกฎความสัมพันธ์แบบกระชับสมบูรณ์

Improving Associative Classification Technique by Using Multiple Essential Class-Association Rules

นามผู้วิจัย นายวีระพล หาญโชติช่วง

ได้พิจารณาเห็นชอบโดย

ประธานกรรมการ .....

(ผู้ช่วยศาสตราจารย์กฤษณะ ไวยมัย, Ph.D.)

กรรมการ .....

(ผู้ช่วยศาสตราจารย์อานนท์ รุ่งสว่าง, Ph.D.)

กรรมการ .....

(อาจารย์พีรวัฒน์ วัฒนพงศ์, Ph.D.)

หัวหน้าภาควิชา .....

(อาจารย์พีรวัฒน์ วัฒนพงศ์, Ph.D.)

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์รับรองแล้ว

.....

(รองศาสตราจารย์วินัย อาจคงหาญ, M.A.)

คณบดีบัณฑิตวิทยาลัย

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

วิทยานิพนธ์

เรื่อง

การเพิ่มประสิทธิภาพเทคนิคการจำแนกประเภทข้อมูลโดยใช้หลายกฎความสัมพันธ์แบบกระชับสมบูรณ์

Improving Associative Classification Technique by Using Multiple Essential Class-Association Rules

โดย

นายวีระพล หาญโชติช่วง

เสนอ

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์
เพื่อความสมบูรณ์แห่งปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมคอมพิวเตอร์)
พ.ศ. 2549

ISBN 974-16-2186-8

วีระพล หาญโชติช่วง 2549: การเพิ่มประสิทธิภาพเทคนิคการจำแนกประเภทข้อมูลโดยใช้หลายกฎความสัมพันธ์แบบกระชับสมบูรณ์ ปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมคอมพิวเตอร์) สาขาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ ประธานกรรมการที่ปรึกษา: ผู้ช่วยศาสตราจารย์กฤษณะ ไวยมัย, Ph.D. 67 หน้า
ISBN 974-16-2186-8

เทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ (Associative Classification) เป็นเทคนิคหนึ่งในสาขาวิชาทางด้านดาตาไมนิ่ง ซึ่งได้รวมเทคนิคการสืบค้นกฎความสัมพันธ์เข้าไว้ด้วยกันกับเทคนิคการจำแนกประเภทข้อมูล โดยที่เทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์สามารถที่จะให้ผลลัพธ์ในการทำนายข้อมูลที่มีประสิทธิภาพและความแม่นยำสูงกว่าเทคนิคการจำแนกประเภทข้อมูลที่ผ่านมาก่อนหน้านี้ ยิ่งไปกว่านั้น การที่นำกฎความสัมพันธ์แบบมีคลาสมาใช้ในการสร้างโมเดลในการทำนายข้อมูล จะให้ความสมบูรณ์ของข้อมูลมากกว่าเทคนิคการจำแนกประเภทข้อมูลที่ใช้หลักทางสถิติหรือความน่าจะเป็น โดยในการเพิ่มประสิทธิภาพและความแม่นยำสำหรับเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ ก็ได้มีผู้นำเสนออัลกอริทึมต่างๆ ไม่ว่าจะเป็น CBA, CMAR และ CPAR ซึ่งอัลกอริทึมเหล่านั้น เน้นปรับปรุงในส่วนของการสร้างโมเดลในการทำนายเป็นหลัก ซึ่งส่งผลให้ในส่วนของการสร้างกฎความสัมพันธ์มีการสร้างกฎที่มีขนาดใหญ่และมีความซับซ้อนออกมาเป็นจำนวนมาก ถึงแม้ว่าจะมีวิธีการในการจัดเรียงกฎรวมไปถึงการกำจัดกฎที่ไม่มีประโยชน์ออกไป แต่กฎจำนวนมากที่มีความซ้ำซ้อนกันก็ยังคงมีอยู่

ดังนั้นในวิทยานิพนธ์เล่มนี้จึงได้เสนอวิธีการในการกำจัดกฎที่ซ้ำซ้อน เพื่อทำให้จำนวนกฎความสัมพันธ์แบบมีคลาสมีจำนวนลดลง และเสนอวิธีการจัดเรียงกฎเพื่อเพิ่มประสิทธิภาพความแม่นยำให้กับโมเดลที่ใช้ในการทำนาย โดยได้เสนออัลกอริทึมใหม่สำหรับเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ เรียกว่า CBEAR (Classification Based on Essential Class-Association Rules) แทนที่จะใช้กฎความสัมพันธ์แบบมีคลาสทั้งหมด แต่ CBEAR ใช้เฉพาะกฎความสัมพันธ์ที่เรียกว่า ECAR (Essential Class-Association Rules) สำหรับการสร้างโมเดลในการทำนาย และในขั้นตอนของการสร้างโมเดลในการทำนายนั้น CBEAR จะมีวิธีการจัดเรียงกฎ รวมถึงวิธีการเลือกกฎที่จะนำไปใช้ในการทำนาย โดยจะพิจารณาเฉพาะกฎที่ยาวที่สุดก่อน ในการเปรียบเทียบประสิทธิภาพความแม่นยำกับอัลกอริทึม C4.5, CBA และ CMAR นั้นได้ใช้ฐานข้อมูลมาตรฐานจาก UCI machine learning database repository ซึ่งจากผลการทดลองพบว่า อัลกอริทึม CBEAR ให้ประสิทธิภาพในการจำแนกประเภทข้อมูลได้แม่นยำมากกว่าอัลกอริทึมตัวอื่นๆ ซึ่งเป็นเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ที่ได้รับความนิยม

_____ / / _____ / / _____

ลายมือชื่อนิสิต ลายมือชื่อประธานกรรมการ

Vearapon Hanchodchung 2006: Improving Associative Classification Technique by Using Multiple Essential Class-Association Rules. Master of Engineering (Computer Engineering), Major Field: Computer Engineering, Department of Computer Engineering. Thesis Advisor: Assistant Professor Kitsana Waiyamai, Ph.D. 67 pages. ISBN 974-16-2186-8

Associative classification is a data mining technique that integrates classification with association rule mining. Associative classification can produce more efficient and accurate classifiers than traditional classification techniques. Moreover, generated classifiers in the form of class-association rules (CARs) are more comprehensive than statistical classifiers. A few accurate and effective classifiers based on associative classification have been presented recently, such as CBA, CMAR and CPAR. With the focus on classifier phase, these algorithms generate very large and complex rule sets during the rule generator phase. Despite strategies for sorting and pruning unuseful rules, large number of redundant rules still exists.

The objective of this research is to propose pruning methods for minimizing and reducing number of class-association rules, and sorting strategies for improving accuracy of the output classifier. We propose a new algorithm, CBEAR (Classification Based on Essential Class-Association Rules) for associative classification. Instead of using a complete rule sets, CBEAR uses only ECAR (Essential Class-Association Rules) for building prediction model. At the classifier phase, CBEAR considers only maximal frequent itemsets for sorting and selecting rules. To compare its accuracy with C4.5, CBA, and CMAR, we use standard datasets from UCI machine learning database repository. Experimental results show that CBEAR yields better classification accuracy compared with other associative classification techniques.

_____ _____ ___ / ___ / ___

Student's signature Thesis Advisor's signature

กิตติกรรมประกาศ

ข้าพเจ้าขอกราบขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร.กฤษณะ ไวยมัย ประธานกรรมการที่ปรึกษา ที่ได้ช่วยเหลือในการวางแผนงานวิจัยในวิทยานิพนธ์ฉบับนี้ ตลอดจนการให้คำปรึกษา พร้อมทั้งให้แนวทางและความรู้เกี่ยวกับทฤษฎีต่างๆ มากมายในการทำวิจัย รวมถึงข้อเสนอแนะที่เป็นประโยชน์ต่อการทำวิทยานิพนธ์ฉบับนี้ รวมถึงการตรวจแก้ไขข้อบกพร่องต่างๆ และขอกราบขอบพระคุณผู้ช่วยศาสตราจารย์ ดร.อานนท์ รุ่งสว่าง กรรมการที่ปรึกษาวิชาเอก และ ดร.พีรวัฒน์ วัฒนพงศ์ กรรมการที่ปรึกษาวิชารอง และผู้ช่วยศาสตราจารย์ อุมาพร ศิรธรานนท์ อาจารย์ผู้แทนบัณฑิตวิทยาลัย ที่กรุณาให้คำปรึกษาแนะนำและได้ให้ข้อเสนอแนะดีๆ ในการทำวิทยานิพนธ์ให้สำเร็จลุล่วงไปด้วยดี

ขอขอบคุณอาจารย์ ธนาวินท์ รักธรรมานนท์ ที่ให้คำปรึกษาที่ดีมาโดยตลอด ไม่ว่าจะเป็นทิศทางของงานวิจัย ความรู้ต่างๆ มากมาย และขอขอบคุณพี่ธนภัทร ฆังคะจิตร ที่คอยช่วยให้คำปรึกษาไม่ว่าจะเป็นเรื่องงานและเรื่องอื่นๆมากมาย และสุดท้ายขอขอบคุณสมาชิกห้องปฏิบัติการ DAKDL คุณท่านที่คอยให้คำแนะนำและกำลังใจที่ดีเสมอมา

ขอขอบคุณพี่จุ๊ เจ้าหน้าที่ธุรการโครงการปริญญาโทที่ช่วยเหลือในการประสานงานและงานด้านเอกสารต่างๆ ให้งานเป็นไปอย่างสะดวกลุล่วงไปด้วยดี รวมถึงขอขอบคุณเจ้าหน้าที่ธุรการภาควิชาวิศวกรรมคอมพิวเตอร์มหาวิทยาลัยเกษตรศาสตร์ทุกท่าน

คุณงามความดี หรือประโยชน์อันใดที่เกิดจากวิทยานิพนธ์ฉบับนี้ ขออุทิศให้แก่ บิดา มารดา บุพการี และผู้มีพระคุณทุกท่าน

วีระพล หาญโชติช่วง

เมษายน 2549

สารบัญ

สารบัญตาราง

สารบัญตาราง (ต่อ)

สารบัญภาพ

คำอธิบายสัญลักษณ์และคำย่อ

CARs	=	Class-Association Rules
CBA	=	Classification Based on Associations
CBA-RG	=	Classification Based on Associations (Rule generator phase)
CBA-CB	=	Classification Based on Associations (Classifier builder phase)
CMAR	=	Classification Based on Multiple Class-Association Rules
CPAR	=	Classification Based on Predictive Association Rules
FP-tree	=	Frequent Pattern tree
CBEAR	=	Classification Based on Essential Class-Association Rules
ECARs	=	Essential Class-Association Rules

การเพิ่มประสิทธิภาพเทคนิคการจำแนกประเภทข้อมูลโดยใช้หลายกฎความสัมพันธ์แบบกระชับสมบูรณ์

Improving Associative Classification Technique by Using Multiple Essential Class-Association Rules

คำนำ

ในปัจจุบันนี้เทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ เป็นเทคนิคหนึ่งที่สำคัญในการในการสืบค้นความรู้บนฐานข้อมูลขนาดใหญ่ โดยการนำข้อมูลที่สืบค้นได้เหล่านั้นมาผ่านกระบวนการเพื่อที่จะได้โมเดลต้นแบบออกมา ซึ่งโมเดลที่ได้นั้นจะนำไปใช้ในการทำนายข้อมูลต่อไป โดยเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์นี้ ได้นำ 2 เทคนิคสำคัญได้ในสาขาวิชาด้าน Data mining นั่นก็คือ การจำแนกประเภทข้อมูล (Data classification) และ การสืบค้นกฎความสัมพันธ์ (Association rule discovery) เข้ามาร่วมไว้ด้วยกัน โดยแบ่งส่วนการทำงานออกเป็น 2 ส่วนหลักๆ คือส่วนการสร้างกฎความสัมพันธ์ (Rule generator) และส่วนในการทำนายข้อมูล (Classifier builder) โดยข้อดีของเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์เมื่อเปรียบเทียบกับเทคนิคการจำแนกข้อมูลโดยทั่วไป เช่นเทคนิคของ C4.5 (Quinlan, 1993) นั่นก็คือ มีความแม่นยำในการทำนายสูง ถึงแม้ว่าข้อมูลที่นำมาใช้จะมีความซับซ้อนและมีหลากหลายคลาสก็ตาม รวมถึงสามารถที่จะให้เหตุผลในการทำนายข้อมูลได้โดยดูจากกฎความสัมพันธ์ที่ใช้ในการทำนาย ดังนั้นผู้ใช้สามารถทำความเข้าใจกับผลของการทำนายได้ง่าย ส่วนเหตุผลที่ทำให้เทคนิคดังกล่าวมีประสิทธิภาพและความแม่นยำในการทำนายสูงนั้น ขึ้นอยู่กับปัจจัยทั้ง 2 ส่วนคือในส่วนของการสร้างกฎความสัมพันธ์ และในส่วนของการสร้างโมเดลในการทำนายข้อมูล

หลังจากเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์เริ่มได้รับความนิยม ก็ได้มีการพัฒนาอัลกอริทึมให้มีประสิทธิภาพมากยิ่งขึ้น ทั้งในด้านของความแม่นยำในการทำนาย ความเร็วของระบบและจำนวนของกฎความสัมพันธ์ที่ได้ แต่เนื่องจากเทคนิคที่มีผู้นำเสนอมานั้นก็ยังคงมีจุดอ่อนอยู่ ทั้งในส่วนของการสร้างกฎความสัมพันธ์ที่มีการสร้างกฎความสัมพันธ์ออกมาเป็นจำนวนมาก และบางส่วนก็ไม่ได้ถูกนำไปใช้ประโยชน์แต่อย่างใด ซึ่งส่งผลต่อประสิทธิภาพของระบบไม่ว่าจะเป็นเรื่องการเสียเวลาในการคำนวณหากฎความสัมพันธ์ เสียเวลาในการประมวลผลข้อมูลที่ไม่มีความจำเป็นต่อระบบ และในส่วนของการสร้างโมเดลในการทำนาย มีการ

ใช้การพิจารณากฎความสัมพันธ์เพียงกฎเดียวในการทำนายข้อมูล จนถึงการนำทุกกฎความสัมพันธ์มาพิจารณา ซึ่งส่งผลให้การทำนายเกิดข้อผิดพลาดได้

ในวิทยานิพนธ์เล่มนี้ได้เสนอวิธีการพัฒนา เพื่อเพิ่มประสิทธิภาพให้กับเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ ซึ่งมีการปรับปรุงใน 2 ส่วนคือ ในส่วนของการสร้างกฎความสัมพันธ์ ได้เสนอวิธีการในการกำจัดกฎความสัมพันธ์แบบมีคลาสที่ไม่มีประโยชน์ และกฎความสัมพันธ์แบบมีคลาสที่ซ้ำซ้อนออกไปเพื่อให้เหลือแต่กฎที่มีประโยชน์ เรียกว่า ECAR (Essential class-association rule) เพื่อนำไปใช้ในการทำนายข้อมูล และในส่วนการสร้างโมเดลในการทำนายข้อมูลได้เสนอวิธีการในการจัดเรียงกฎแบบใหม่ รวมถึงวิธีการเลือกกฎความสัมพันธ์แบบมีคลาสไปใช้ในการคำนวณ โดยจะทำการพิจารณากลุ่มของกฎความสัมพันธ์มีแบบคลาสเฉพาะในระดับของกฎที่ยาวที่สุดก่อน (Maximal frequent itemsets) รวมถึงได้เสนอสูตรการคำนวณค่ากลุ่มของกฎความสัมพันธ์แบบมีคลาสเพื่อใช้ในการทำนายข้อมูลอีกด้วย

วัตถุประสงค์และขั้นตอนการวิจัย

วัตถุประสงค์ของการวิจัย

พัฒนาเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ให้มีประสิทธิภาพและความแม่นยำในการทำนายข้อมูลได้สูง โดย

1) ในส่วนของการสร้างกฎความสัมพันธ์ ได้เสนอวิธีการใหม่ในการกำจัดกฎความสัมพันธ์แบบมีคลาสที่ซ้ำซ้อนออกทั้งหมด

2) ในส่วนของการสร้างโมเดลในการทำนายข้อมูล ได้เสนอวิธีการใหม่ในการจัดเรียงและพิจารณากฎความสัมพันธ์แบบมีคลาส รวมถึงเสนอสูตรที่ใช้ในการทำนายข้อมูล

ขั้นตอนการวิจัย

1. ศึกษาทฤษฎีต่างๆของเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ รวมถึงศึกษาทฤษฎีที่เกี่ยวข้อง เพื่อที่จะนำความรู้ที่ได้มาใช้ในการวิจัย
2. รวบรวมฐานข้อมูลมาตรฐานเพื่อที่จะนำมาใช้ในการทดสอบเพื่อศึกษาผลการทำนายที่ได้และหาสาเหตุของปัญหาที่เกิดขึ้น
3. ศึกษาผลที่ได้จากการทดลอง เพื่อวิเคราะห์ปัญหาและรวบรวมข้อดีและข้อด้อยต่างๆของงานก่อนหน้านี้ เพื่อนำมาเป็นข้อมูลในการพัฒนาเทคนิคและอัลกอริทึมในการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์
4. พัฒนาเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์
5. ทดสอบและวัดผลของเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ที่ได้พัฒนาขึ้น
6. สรุปผลการวิจัยและประโยชน์ที่ได้รับ

การตรวจเอกสาร

ความรู้พื้นฐานของเทคนิคดาต้าไมนิ่ง

การสืบค้นกฎความสัมพันธ์ (Association Rule Discovery)

การสืบค้นกฎความสัมพันธ์ (Association Rule Discovery) เป็นหนึ่งในเทคนิคของ Data mining ที่มีความสำคัญ โดยวิธีการสืบค้นกฎความสัมพันธ์นี้ จะเปรียบเสมือนกับการค้นหาทองจากฐานข้อมูลขนาดใหญ่ ซึ่งทองที่ได้กล่าวถึงนั้นก็คือ กฎ ที่มีความน่าสนใจ ที่บ่งบอกถึงลักษณะเฉพาะหรือคุณสมบัติเด่นของฐานข้อมูลนั้นๆ โดยที่เราไม่สามารถที่จะค้นหาได้มาก่อน โดยหลักการทำงานของเทคนิคนี้คือ การค้นหากฎความสัมพันธ์ของข้อมูลจากข้อมูลขนาดใหญ่ที่มีอยู่ เพื่อนำกฎที่ได้เหล่านั้นไปวิเคราะห์เพื่อใช้ช่วยในการตัดสินใจ โดยส่วนใหญ่แล้วจะนำไปใช้ทางด้านธุรกิจ (Business decision making) เช่น การนำเทคนิคนี้ไปวิเคราะห์พฤติกรรมของลูกค้าที่ซื้อสินค้าในซุปเปอร์มาเก็ต (Market basket analysis) โดยดูว่าลูกค้ามักจะซื้นสินค้าอะไรด้วยกัน เพื่อที่จะนำข้อมูลการซื้อสินค้าของลูกค้าเหล่านั้นมาช่วยในการวางแผนทางการตลาด เช่น การจัดวางสินค้าที่มักจะถูกซื้อด้วยกันไว้ใกล้ๆ กันหรือการจัดโปรโมชั่นให้กับสินค้า เป็นต้น

โดยหนึ่งในอัลกอริทึมในการสืบค้นกฎความสัมพันธ์ที่รู้จักกันดี นั่นคือ Apriori อัลกอริทึม (Agrawal and Srikant, 1994; Agrawal et al., 1993) ซึ่งหลักการคือ จะทำการคำนวณหาความสัมพันธ์ของ Itemsets ที่มักจะเกิดขึ้นพร้อมๆกันในฐานข้อมูล โดยความสัมพันธ์ของ Itemsets นั้นเรียกว่า กฎความสัมพันธ์ (Association rule) ซึ่งจะอยู่ในรูปแบบดังต่อไปนี้

{item1, item2} => {item3} [ค่า Support%, ค่า Confidence%]

ภาพที่ 1 ตัวอย่างกฎความสัมพันธ์

โดยอัลกอริทึม Apriori จะต้องมีการกำหนดค่าสนับสนุนขั้นต่ำ (Minimum support) และค่าความมั่นใจขั้นต่ำ (Minimum confidence) ด้วยซึ่งในการกำหนดค่าขั้นต่ำทั้งสองค่านี้ จะขึ้นอยู่กับผู้ใช้ระบบเป็นผู้กำหนดเอง หรือจะใช้ผู้เชี่ยวชาญ (Expert user) เป็นผู้กำหนดให้ก็ได้ โดยกฎความสัมพันธ์ที่ได้นั้นจะต้องมีค่าสนับสนุน (Support) และค่าความมั่นใจ (Confidence) ไม่น้อยกว่าค่าขั้นต่ำที่ได้กำหนดเอาไว้ข้างต้น โดยที่ค่าสนันสนุน (Support) คือ เปอร์เซ็นต์ของจำนวน Itemsets ทั้งหมดที่เกิดขึ้นในฐานข้อมูล ส่วนค่าความมั่นใจ (Confidence) คือ เปอร์เซ็นต์ของจำนวน Itemsets ทั้งหมดที่เกิดขึ้นในฐานข้อมูล ต่อ จำนวน Itemsets ที่เกิดขึ้นทางด้านซ้ายมือ

ตัวอย่างการใช้เทคนิค Association Rule Discovery ในการค้นหากฎความสัมพันธ์ของข้อมูลโดยในตารางที่ 1 คือ ตัวอย่างชุดข้อมูลการซื้อสินค้า ซึ่งคอลัมน์ TID เปรียบเสมือนตะกร้าที่ใส่สินค้าที่ซื้อในครั้งหนึ่งๆ และคอลัมน์ Items คือรายการสินค้าที่ซื้อพร้อมกันใน TID ใดๆ และตัวอักษร A, B, C, D, และ E แทนชื่อสินค้าแต่ละชนิด โดยที่กำหนดค่าสนับสนุนขั้นต่ำ (Minimum support) เท่ากับ 50% และค่าความมั่นใจขั้นต่ำ (Minimum confidence) เท่ากับ 70%

ตารางที่ 1 ตัวอย่างข้อมูลรายการซื้อสินค้า

TID	Items
1	A C D
2	B C E
3	A B C E
4	B E
5	A B C E

จากข้อมูลในตารางที่ 1 ก็จะถูกนำเข้าสู่กระบวนการสร้างกฎความสัมพันธ์ โดยขั้นตอนวิธีการสร้างกฎความสัมพันธ์ สามารถดูได้จาก ตารางที่ 2 ถึง ตารางที่ 4 และกฎความสัมพันธ์ที่ได้ทั้งหมดสามารถดูได้จากตารางที่ 5

ตารางที่ 2 อัลกอริทึม Apriori ทำการค้นหา Frequent itemsets รอบแรก

1-itemsets C_1	1-itemsets C_1	Count	%	Large 1-itemsets L_1	Count	%
{A}	{A}	3	60	{A}	3	60
{B}	{B}	4	80	{B}	4	80
{C}	{C}	4	80	{C}	4	80
{D}	{D}	1	20			
{E}	{E}	4	80	{E}	4	80
a) Generage phase	b1) Count phase			b2) Select phase		

ตารางที่ 3 อัลกอริทึม Apriori ทำการค้นหา Frequent itemsets รอบที่สอง

2-itemsets C_2	2-itemsets C_2	Count	%	Large 2-itemsets L_2	Count	%
{A, B}	{A, B}	2	40			
{A, C}	{A, C}	3	60	{A, C}	3	60
{A, E}	{A, E}	2	40			
{B, C}	{B, C}	3	60	{B, C}	3	60
{B, E}	{B, E}	4	80	{B, E}	4	80
{C, E}	{C, E}	3	60	{C, E}	3	60
a) Generage phase	b1) Count phase			b2) Select phase		

ตารางที่ 4 อัลกอริทึม Apriori ทำการค้นหา Frequent itemsets รอบที่สาม

3-itemsets C_3	3-itemsets C_3	Count	%	Large 3-itemsets L_3	Count	%
{B, C, E}	{B, C, E}	3	60	{B, C, E}	3	60
a) Generage phase	b1) Count phase			b2) Select phase		

ตารางที่ 5 กฎความสัมพันธ์ทั้งหมดที่ถูกสร้างโดย อัลกอริทึม Apriori

กฎความสัมพันธ์	ค่าสนับสนุน (%)	ค่าความมั่นใจ (%)
{BC} => {E}	60	3/3 = 100
{CE} => {B}	60	3/3 = 100
{BE} => {C}	60	3/4 = 75
{B} => {CE}	60	3/4 = 75
{C} => {BE}	60	3/4 = 75
{E} => {BC}	60	3/4 = 75
{A} => {C}	60	3/3 = 100
{B} => {C}	60	3/4 = 75
{B} => {E}	80	4/4 = 100
{C} => {E}	60	3/4 = 75

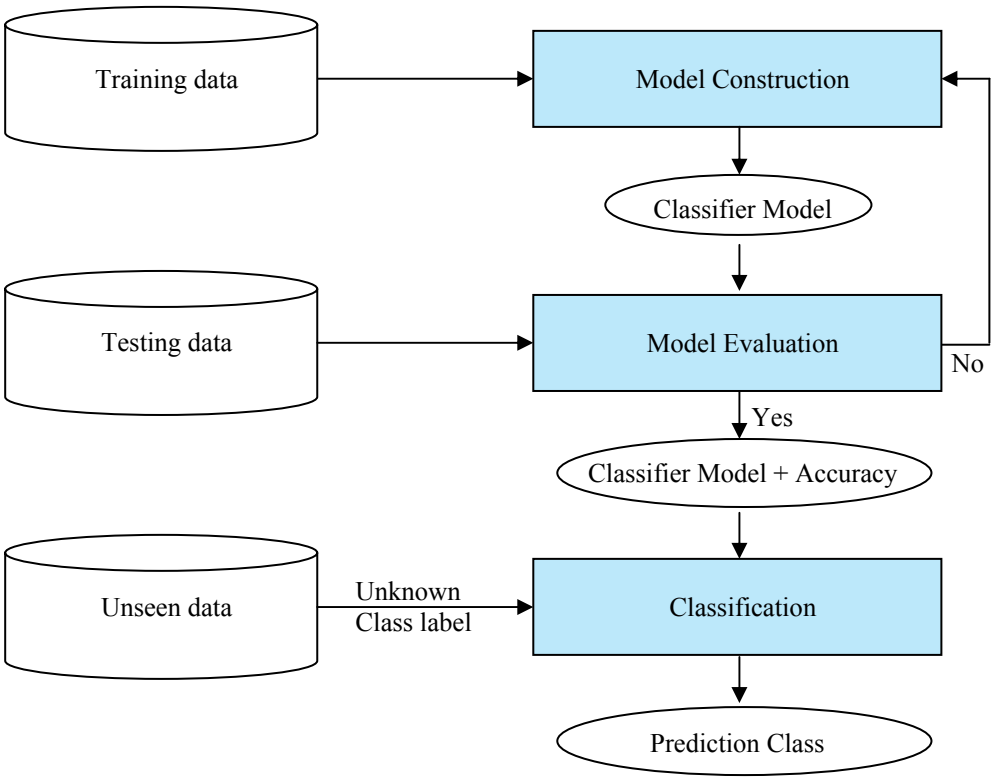
จากตารางที่ 5 แสดงกฎความสัมพันธ์ทั้งหมดที่ถูกสร้างโดย อัลกอริทึม Apriori ซึ่งข้อมูลในตารางจะประกอบไปด้วย กฎความสัมพันธ์ ค่าสนับสนุนของกฎความสัมพันธ์ (Support) และค่าความมั่นใจของกฎความสัมพันธ์ (Confidence)

จากตัวอย่างกฎความสัมพันธ์ในตารางที่ 5 กฎความสัมพันธ์ {BC} => {E} มีค่าสนับสนุนเท่ากับ 60% และค่าความมั่นใจ เท่ากับ 100% หมายความว่า จำนวนครั้งการซื้อสินค้าที่มีการซื้อ B, C, และ E พร้อมกันมีจำนวน 3 ครั้ง จากจำนวนรายการทั้งหมด และความน่าจะเป็นที่เมื่อมีการซื้อสินค้า B และ C พร้อมกันแล้ว จะซื้อสินค้า E ด้วยเสมอ คิดเป็น 100%

<u>การจำแนกประเภทข้อมูล (Data Classification)</u>

การจำแนกประเภทข้อมูล(Data Classification) เป็นอีกหนึ่งเทคนิคใน Data Mining ซึ่งทำหน้าที่สืบค้นความรู้เพื่อสรุปหาแบบจำลองหรือโมเดลของฐานข้อมูลนั้นๆ (Quinlan, 1993; Wang et al., 2000) เพื่อใช้ในการทำนายข้อมูลใหม่ (unseen data) โดยเทคนิคนี้จะหาความสัมพันธ์ของข้อมูลจากฐานข้อมูลขนาดใหญ่ เพื่อนำมาสร้างโมเดลเพื่อใช้ในการจำแนกประเภทข้อมูล ซึ่งจะสามารถนำไปจำแนกประเภทข้อมูลใหม่ๆ ที่ยังไม่ทราบประเภทได้ (Unknown class label)

เนื่องจากการจำแนกประเภทข้อมูลเป็นเทคนิคแบบ Supervise learning นั่นคือ การจะสร้างโมเดลออกมาได้นั้นจะต้องทำการสอนระบบเสียก่อน ดังนั้นเราจำเป็นจะต้องทราบจำนวนคลาสปลายทาง (Class label) และจำนวนแอตทริบิวต์ (Attribute) ที่แน่นอน และส่วนของข้อมูลจะต้องแบ่งออกเป็นสองส่วน ส่วนหนึ่งใช้สอนระบบ (Training data) อีกส่วนหนึ่งใช้ทดสอบความแม่นยำของโมเดลที่ถูกสร้างออกมา (Testing data) โดยปรกติสัดส่วนระหว่าง Training กับ Testing จะอยู่ที่ประมาณ 80 ต่อ 20 โดยในการที่จะสร้างโมเดลออกมาเพื่อใช้สำหรับทำนายข้อมูลได้นั้น จะต้องผ่านขั้นตอนดังต่อไปนี้ เริ่มจากการนำข้อมูลสอนระบบ (Training data) เข้ามาสู่กระบวนการสร้างโมเดลจำแนกประเภทข้อมูล (Model construction) เพื่อให้ได้โมเดลจำแนกประเภทข้อมูล (Classifier model) ออกมา และหลังจากได้โมเดลจำแนกประเภทข้อมูลแล้ว วิธีการทดสอบว่าโมเดลที่ถูกสร้างขึ้นมามีความแม่นยำมากเพียงพอที่จะนำไปใช้ได้หรือไม่นั้น จะใช้ข้อมูลทดสอบระบบ หรือ Testing data เพื่อทดสอบความแม่นยำของโมเดลที่ถูกสร้างขึ้นมา (Model evaluation) ถ้าโมเดลที่สร้างขึ้นมามีความแม่นยำไม่ผ่านเกณฑ์ที่ต้องการ ก็จะต้องกลับไปปรับปรุงในส่วนของกระบวนการสร้างโมเดลจำแนกประเภทข้อมูลเสียก่อน แต่ถ้าโมเดลที่สร้างขึ้นมามีความแม่นยำผ่านเกณฑ์ที่ต้องการแล้ว ก็สามารถที่จะนำโมเดลที่สร้างมานั้นไปประยุกต์ใช้เพื่อทำนายประเภทข้อมูลใหม่ (Unseen data) ที่ไม่ทราบประเภทของข้อมูล (Unknown class label) ต่อไปได้ โดยภาพรวมขั้นตอนทั้งหมดของเทคนิคการจำแนกประเภทข้อมูลที่อธิบายไว้ข้างต้น สามารถดูได้จากภาพที่ 2



ภาพที่ 2 ขั้นตอนของเทคนิคการจำแนกประเภทข้อมูล

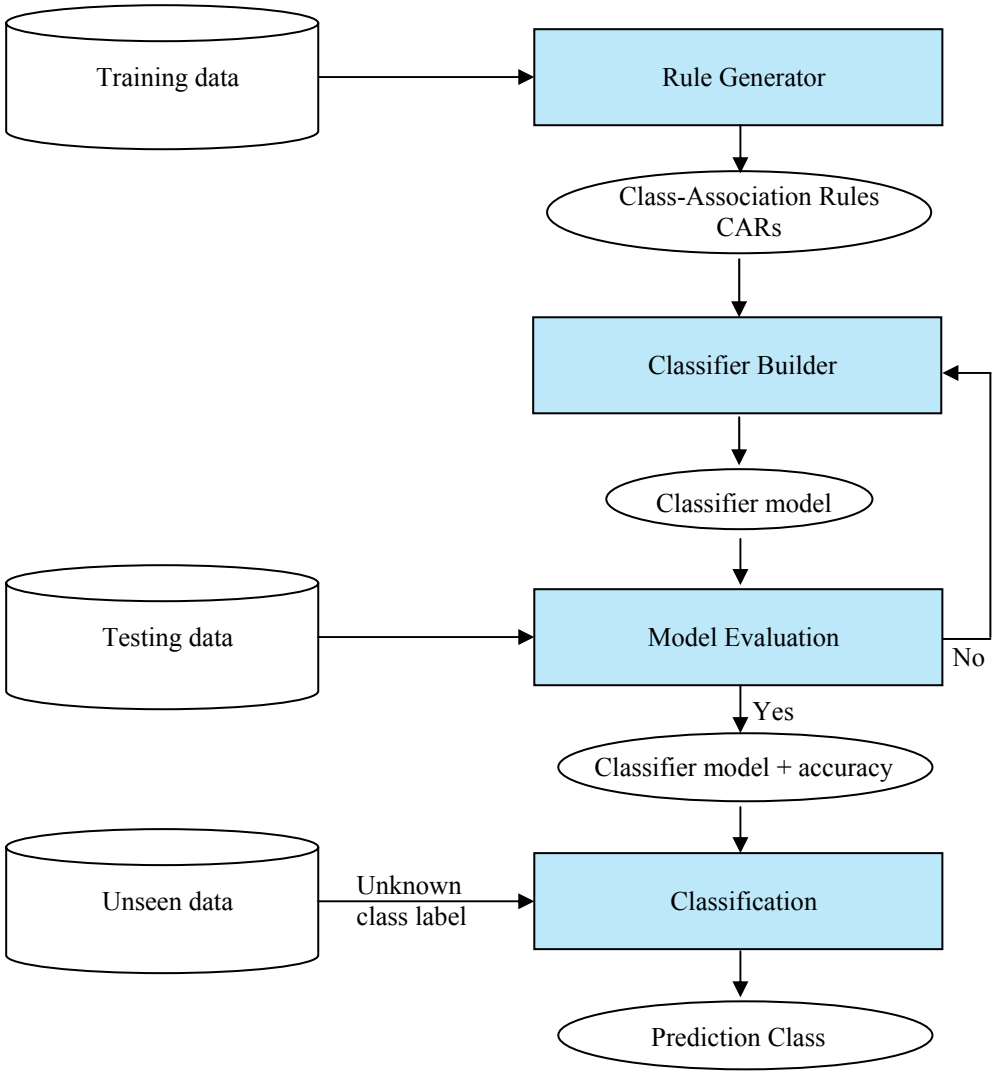
<u>การจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ (Associative Classification)</u>

การจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ (Associative classification) เป็นเทคนิคที่เกิดจากการรวมกันระหว่าง 2 เทคนิค (Liu et al., 1998) ที่ได้กล่าวในหัวข้อข้างต้น นั่นคือ การจำแนกประเภทข้อมูล (Data classification) และ การสืบค้นกฎความสัมพันธ์ (Association rule discovery) โดยที่จุดประสงค์ของเทคนิคการจำแนกประเภทข้อมูลคือ เพื่อค้นหาโมเดลหรือเซตที่เล็กที่สุดของกฎในฐานข้อมูล เพื่อสร้างโมเดลจำแนกประเภทข้อมูลที่มีความถูกต้องแม่นยำมากที่สุด และจุดประสงค์ของเทคนิคการสืบค้นกฎความสัมพันธ์คือ เพื่อค้นหากฎความสัมพันธ์ทั้งหมดที่มีความสำคัญและบ่งบอกถึงคุณลักษณะของฐานข้อมูล โดยที่กฎเหล่านั้นจะต้องผ่านค่าสนับสนุนและค่าความมั่นใจขั้นต่ำด้วย โดยเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ (Associative classification) นี้ ได้แบ่งออกเป็น 2 ส่วนหลักๆ คือ ส่วนที่ใช้ในการสร้างกฎความสัมพันธ์ (Rule generator phase) และส่วนที่นำกฎความสัมพันธ์ไปสร้างโมเดลเพื่อใช้ทำนายข้อมูล (Classifier builder phase)

โดยในส่วนของการสร้างกฎความสัมพันธ์ (Rule generator phase) นั้นจะใช้หลักการหรือวิธีการเดียวกันกับเทคนิค Association rule discovery เกือบทั้งหมด ยกเว้นกฎที่ถูกสร้างจากกระบวนการสร้างกฎความสัมพันธ์นั้นจะต้องเป็นกฎเฉพาะที่เรียกว่า กฎความสัมพันธ์แบบมีคลาส หรือ CARs (Class-Association Rules) นั่นคือกฎความสัมพันธ์ที่สับเซตของกฎทางด้านขวามือจะต้องเป็นแอตทริบิวต์ Class เท่านั้น เช่น {A, B, C $\rightarrow$ **Class**} โดยอัลกอริทึมการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ที่มีอยู่ (Dong et al., 1999; Liu et al., 1998) จะถูกดัดแปลงเพื่อค้นหา CARs ทั้งหมดที่ผ่านค่าสนับสนุนขั้นต่ำ (Minimum support) และค่าความมั่นใจขั้นต่ำ (Minimum confidence)

ในส่วนของการสร้างโมเดลในการทำนาย (Classifier builder phase) จะนำกฎความสัมพันธ์ที่ได้จากส่วนการสร้างกฎมาใช้เพื่อสร้างโมเดลในการทำนายข้อมูล โดยในการทำนายข้อมูลนั้น จะมีการพิจารณาแบ่งออกเป็น 2 วิธี วิธีที่1 จะทำการพิจารณากฎความสัมพันธ์ทีละกฎ (Single rule) โดยวิธีการพิจารณาแบบนี้ จะต้องทำการเรียงลำดับกฎความสัมพันธ์ก่อน โดยทั่วไปแล้วจะเรียงลำดับกฎความสัมพันธ์ตามค่าความมั่นใจ (Confidence) ก่อน แต่ถ้าค่าความมั่นใจของกฎความสัมพันธ์เท่ากัน ก็จะเรียงลำดับของกฎความสัมพันธ์ตามค่าสนับสนุน (Support) แต่ถ้าทั้ง

ค่าความมั่นใจ และ ค่าสนับสนุนของกฎเกิดเท่ากันอีก ก็จะเรียงลำดับกฎโดยดูจาก กฎไหนถูกสร้างมาก่อน ก็จะเรียงกฎนั้นก่อนตามลำดับ หลังจากเรียงลำดับกฎความสัมพันธ์เป็นที่เรียบร้อยแล้วก็พร้อมที่จะทำนายข้อมูล โดยการทำนายข้อมูลนั้นจะทำนายตาม class ของกฎที่มีศักย์ (Precedence) สูงที่สุด ส่วนวิธีที่2 จะทำการพิจารณากฎความสัมพันธ์ทีละหลายๆกฎพร้อมกัน (Multiple rules) โดยในการทำนายข้อมูลนั้นจะนำกลุ่มของกฎที่มีอยู่ในคลาสเดียวกัน มาคำนวณผ่านสูตรที่ได้กำหนดเอาไว้แล้วดูว่าคลาสไหนที่ให้ค่ามากที่สุดคลาสนั้นก็จะเป็นคำตอบ โดยที่ภาพรวมขั้นตอนทั้งหมดของเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ สามารถดูได้จากภาพที่ 3



<u>ภาพที่ 3</u> ขั้นตอนของเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์

งานวิจัยที่เกี่ยวข้อง

ในวิทยานิพนธ์เล่มนี้จะนำเสนองานวิจัยที่เกี่ยวข้องกับเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ (Association Classification) โดยจะนำเสนออัลกอริทึม CBA (Liu et al., 1998) ซึ่งเป็นต้นแบบของเทคนิคดังกล่าว รวมถึงนำเสนองานวิจัยที่ได้พัฒนาให้เทคนิคดังกล่าวมีประสิทธิภาพมากยิ่งขึ้น นั่นก็คือ อัลกอริทึม CMAR (Wenmin, 2001; Wenmin et al., 2001) ซึ่งเป็นอัลกอริทึมที่ให้ความแม่นยำในการทำนายสูงที่สุด โดยในการนำเสนอ จะอธิบายรายละเอียดพร้อมตัวอย่างของแต่ละอัลกอริทึมด้วย

CBA อัลกอริทึม

Classification Based on Associations (CBA) เป็นงานแรกในการเสนอวิธีการรวมเทคนิคการสืบค้นกฎความสัมพันธ์เข้ากับเทคนิคการจำแนกประเภทข้อมูล (Liu et al., 1998) โดยวิธีการแบบนี้ถูกเรียกว่า การจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ (Associative Classification) ซึ่งอัลกอริทึม CBA ประกอบด้วย 2 ส่วนนั่นคือ ในคือในส่วนการสร้างกฎความสัมพันธ์ เรียกว่า CBA-RG ซึ่งอัลกอริทึมในส่วนแรกนี้ก็จะอ้างอิงกับอัลกอริทึม Apriori (Agrawal and Srikant, 1994; Agrawal et al., 1993) และในส่วนของการสร้างโมเดลในการทำนาย เรียกว่า CBA-CB

ก่อนที่จะทำความเข้าใจอัลกอริทึมในส่วนของ CBA-RG นั้น จะต้องทราบเกี่ยวกับแนวคิดเบื้องต้นที่ใช้ในอัลกอริทึมในส่วนของ CBA-RG เสียก่อน โดยจุดมุ่งหมายในส่วนของ CBA-RG นั่นก็คือการค้นหากฎความสัมพันธ์ทั้งหมดที่มีค่าสนับสนุนมากกว่าค่าสนับสนุนขั้นต่ำ โดยที่กฎความสัมพันธ์เหล่านั้นจะอยู่ในรูปของ

<condset, y>

เมื่อ condset คือเซตของไอเท็ม (set of items) และ y คือ คลาสปลายทาง (class label) โดยที่ condsupCount คือ จำนวนครั้งของ Transaction ที่เกิด condset ขึ้นในฐานข้อมูล (D) และ rulesupCount คือ จำนวนครั้งของ Transaction ที่เกิด condset และคลาสปลายทางเป็น y พร้อมกัน โดยกฎความสัมพันธ์ที่ได้จะประกอบด้วยค่าต่อไปนี้

condset → y, ค่าสนับสนุน, ค่าความมั่นใจ

โดยที่ ค่าสนับสนุน (Support) หาได้จาก (rulesupCount ÷ |D|) × 100% โดยที่ |D| คือ ขนาดของฐานข้อมูลหรือจำนวนแถวของข้อมูล (Transaction) และ ค่าความมั่นใจ (Confidence) หาได้จาก (rulesupCount ÷ condsupCount) × 100%

กฎความสัมพันธ์ที่ผ่านค่าสนับสนุนขั้นต่ำ (Minimum support) จะถูกเรียกว่า Frequent ruleitems ส่วนกฎความสัมพันธ์ที่เหลือจะเรียกว่า Infrequent ruleitems

สำหรับกรณีที่มีกฎความสัมพันธ์มากกว่า 1 กฎที่มี condset เหมือนกันนั้น จะทำการเลือกกฎความสัมพันธ์ที่มีค่าความมั่นใจสูงที่สุด เรียกว่า Possible rule (PR) เป็นตัวแทนกฎความสัมพันธ์ที่เหลือ และถ้ากฎความสัมพันธ์นั้นมีค่าความมั่นใจมากกว่าค่าความมั่นใจขั้นต่ำ เราจะบอกว่ากฎเหล่านั้นมีความถูกต้อง (Accurate) ดังนั้นเซตที่เรียกว่า Class-Association Rules (CARs) จะประกอบไปด้วยกฎที่เรียกว่า Frequent และ Accurate

<u>ขั้นตอนในการสร้างกฎความสัมพันธ์ CBA-RG</u>

1. ทำการค้นหา Frequent ruleitems ทั้งหมด (Frequent ruleitem คือ ruleitems ที่มีการ Support สูงกว่า Minimum support) โดยขั้นตอนนี้จะเหมือนกับ Apriori ซึ่งผลลัพธ์ที่ได้นั้นจะเลือกเฉพาะกฎที่เป็น CARs

   <condset,y>, เมื่อ condset เป็น set ของ item และ y เป็น class label

2. สร้าง Class-Association Rule (CARs) จาก Frequent ruleitems และสำหรับทุกๆ ruleitems ที่มี condset เหมือนกัน เราจะเลือกเพียงกฎเดียว โดยกฎที่มีค่า Confidence สูงที่สุด จะถูกเลือก

3. ทำการ Prune ruleitems ที่มีค่า Confidence น้อยกว่า Minimum confidence

หลังจากได้เซตของกฎความสัมพันธ์ (CARs) ทั้งหมดแล้ว ก็จะถึงส่วนถัดไปนั่นก็คือ การสร้างโมเดลในการทำนาย (Classifier builder) โดยในส่วนนี้จะกล่าวถึงหลักการในการเรียงกฎ และขั้นตอนในการสร้างโมเดลที่ใช้ในการทำนายข้อมูล

<u>แนวคิดหลักการเบื้องต้นที่ใช้ในการเรียงกฎ มีดังต่อไปนี้</u>

กำหนดให้ 2 rules นั่นคือ r_i และ r_j , เราสามารถกำหนดว่า $r_i \succ r_j$ (r_i มีศักย์สูงกว่า r_j) ได้ก็ต่อเมื่อ

1. ค่าความมั่นใจ (Confidence) ของ r_i มากกว่า r_j หรือ
2. ค่าความมั่นใจ (Confidence) เท่ากัน แต่ค่าสนับสนุน (Support) ของ r_i มากกว่า r_j หรือ
3. ทั้งค่าความมั่นใจ (Confidence) และค่าสนับสนุน (Support) เท่ากัน แต่ r_i ถูกสร้างมาก่อน r_j

<u>ขั้นตอนในการสร้างโมเดลในการทำนาย CBA-CB</u>

1. ทำการเรียงลำดับของกฎทั้งหมด โดยใช้หลักการในการเรียงกฎที่กล่าวมาแล้ว
2. จะทำการวนลูปกฎ โดยจะนำกฎแต่ละกฎไปเช็คกับตัวข้อมูล (โดยตัวข้อมูลที่ใช้นั้นเป็นข้อมูลที่เราทราบกลุ่มแล้วเรียกว่า Training data) จากนั้น ถ้ากฎสามารถที่จะรองรับ (satisfy) ตัวข้อมูลทั้งค่าทางด้านซ้าย (condset) และค่าทางด้านขวา (class) แล้ว จะนำกฎนั้นไปเก็บไว้ในเซตของกฎ (Set C) แล้วทำการหา Default class เพื่อใช้ในการทำนายข้อมูลในกรณีที่ไม่มีกฎไหนรองรับข้อมูลนั้นเลย จากนั้นจะเซตของกฎนั้นมาคำนวณหาเปอร์เซนต์ข้อผิดพลาด
3. นำเซตของกฎทุกๆเซตมาทำการเปรียบเทียบกัน แล้วเลือกเซตของกฎที่มีเปอร์เซนต์ความผิดพลาดน้อยที่สุด เพื่อที่จะนำเซตของกฎนั้นไปเป็นโมเดลเพื่อใช้ในการทำนายข้อมูลที่เราไม่ทราบกลุ่ม (Unseen data)

ตัวอย่างการทำงานของอัลกอริทึม CBA ตั้งแต่การค้นหากฎความสัมพันธ์ (CBA-RG) และการสร้างโมเดลในการทำนายข้อมูล (CBA-CB) โดยในตารางที่ 6 คือ ตัวอย่างชุดข้อมูลของคนไข้ ซึ่งประกอบไปด้วย 6 คอลัมน์ โดยมีคลาสปลายทาง คือ Status หรือสถานะของคนไข้นั่นเอง

โดยที่กำหนดค่าสนับสนุนขั้นต่ำ (Minimum support) เท่ากับ 30% และค่าความมั่นใจขั้นต่ำ (Minimum confidence) เท่ากับ 60%

ตารางที่ 6 ตัวอย่างข้อมูลคนไข้

TID	Sex	Cholesterol (250)	Blood sugar <120	Vassel color number	Status
1	Male	High	False	1	Sick
2	Male	Low	True	3	Well
3	Female	Low	False	3	Sick
4	Female	High	True	2	Well
5	Female	High	False	3	Well
6	Male	Low	True	1	Sick
7	Female	Low	False	3	Sick
8	Female	High	True	3	Well
9	Male	Low	False	1	Few
10	Female	Low	True	1	Well

จากตัวอย่างข้อมูลคนไข้ในตารางที่ 6 ก็จะถูกนำเข้าสู่กระบวนในส่วนของการสร้างกฎความสัมพันธ์ (CBA-RG) โดยขั้นตอนวิธีการสร้างกฎความสัมพันธ์ สามารถดูได้จาก ตารางที่ 7 ถึง ตารางที่ 9 และกฎความสัมพันธ์ที่ได้ทั้งหมดที่ผ่านค่าสนับสนุนขั้นต่ำสามารถดูได้จากตารางที่ 10 ส่วนตารางที่ 11 จะแสดงกฎความสัมพันธ์ที่ผ่านค่าสนับสนุนขั้นต่ำและค่าความมั่นใจขั้นต่ำ ซึ่งหลังจากนั้น อัลกอริทึม CBA-RG จะทำการจัดเรียงกฎความสัมพันธ์แบบมีคลาสใหม่ตามค่าความมั่นใจ โดยดูจากตารางที่ 12

ตารางที่ 7 อัลกอริทึม CBA-RG ทำการค้นหา Frequent itemsets รอบแรก

1-itemsets C_1	1-itemsets C_1	Count	%	Large 1-itemsets L_1	Count	%
{Male}	{Male}	4	40	{Male}	4	40
{Female}	{Female}	6	60	{Female}	6	60
{High}	{High}	4	40	{High}	4	40
{Low}	{Low}	6	60	{Low}	6	60
{True}	{True}	5	50	{True}	5	50
{False}	False	5	50	False	5	50
{1}	{1}	4	40	{1}	4	40
{2}	{2}	1	10			
{3}	{3}	5	50	{3}	5	50
{Sick}	{Sick}	4	40	{Sick}	4	40
{Few}	{Few}	1	10			
{Well}	{Well}	5	50	{Well}	5	50
a) Generage phase	b1) Count phase			b2) Select phase		

ตารางที่ 7 แสดงขั้นตอนของอัลกอริทึม CBA-RG ทำการค้นหาไอเท็มเซต ที่มีค่าสนับสนุนมากกว่า ค่าสนับสนุนขั้นต่ำ ซึ่งได้กำหนดไว้คือ 30 เปอร์เซ็นต์ โดยไอเท็มเซตที่ผ่านค่าสนับสนุนขั้นต่ำ จะถูกนำไปใช้ในการหาไอเท็มเซตในระดับที่สอง ถัดไป

ตารางที่ 8 อัลกอริทึม CBA-RG ทำการค้นหา Frequent itemsets รอบที่สอง

2-itemsets C_2	2-itemsets C_2	Count	%	Large 2-itemsets L_2	Count	%
{Male, Sick}	{Male, Sick}	2	20			
{Male, Well}	{Male, Well}	1	10			
{Female, Sick}	{Female, Sick}	2	20			
{Female, Well}	{Female, Well}	4	40	{Female, Well}	4	40
{High, Sick}	{High, Sick}	1	10			
{High, Well}	{High, Well}	3	30	{High, Well}	3	30
{Low, Sick}	{Low, Sick}	3	30	{Low, Sick}	3	30
{Low, Well}	{Low, Well}	2	20			
{True, Sick}	{True, Sick}	1	10			
{True, Well}	{True, Well}	4	40	{True, Well}	4	40
{False, Sick}	{False, Sick}	3	30	{False, Sick}	3	30
{False, Well}	{False, Well}	1	10			
{1, Sick}	{1, Sick}	2	20			
{1, Well}	{1, Well}	1	10			
{3, Sick}	{3, Sick}	2	20			
{3, Well}	{3, Well}	3	30	{3, Well}	3	30
a) Generage phase	b1) Count phase			b2) Select phase		

ตารางที่ 8 แสดงขั้นตอนของอัลกอริทึม CBA-RG ทำการค้นหาไอเท็มเซตในระดับที่สอง ที่มีค่าสนับสนุนมากกว่า ค่าสนับสนุนขั้นต่ำ โดยไอเท็มเซตที่ผ่านค่าสนับสนุนขั้นต่ำ จะถูกนำไปใช้ในการหาไอเท็มเซตในระดับที่สาม ถัดไป

ตารางที่ 9 อัลกอริทึม CBA-RG ทำการค้นหา Frequent itemsets รอบที่สาม

3-itemsets C_3	3-itemsets C_3	Count	%	Large 3-itemsets L_3	Count	%
{Female, High, Well}	{Female, High, Well}	3	30	{Female, High, Well}	3	30
{Female, Low, Well}	{Female, Low, Well}	1	10			
{Low, True, Sick}	{Low, True, Sick}	1	10			
{Low, False, Sick}	{Low, False, Sick}	2	20			
{False, 3, Sick}	{False, 3, Sick}	2	20			
a) Generage phase	b1) Count phase			b2) Select phase		

จากตารางที่ 9 ซึ่งเป็นขั้นตอนในการหาไอเท็มเซตในระดับที่สาม จะเห็นได้ว่า ในระดับนี้จะเป็นระดับสุดท้าย เนื่องจากไม่สามารถสร้างไอเท็มเซตในระดับที่สี่ได้ เพราะเหลือไอเท็มเซตที่ผ่านค่าสนับสนุนในระดับที่สามเพียงตัวเดียวเท่านั้น

ตารางที่ 10 CBA-RG สร้าง Class-Association Rules (CARs) ที่ผ่านค่าสนับสนุน

CARs	ค่าสนับสนุน (%)	ค่าความมั่นใจ (%)
{Female} => {Well}	40	4/6 = 66.7
{High} => {Well}	30	3/4 = 75
{Low} => {Sick}	30	3/6 = 50
{True} => {Well}	40	4/5 = 80
{False} => {Sick}	30	3/5 = 60
{3} => {Well}	30	3/5 = 60
{Female, High} => {Well}	30	3/3 = 100

จากตารางที่ 10 แสดงกฎความสัมพันธ์แบบมีคลาส (CARs) ในทุกๆ ระดับของไอเท็มเซตที่ผ่านค่าสนับสนุนขั้นต่ำ

ตารางที่ 11 CBA-RG สร้าง Class-Association Rules (CARs) ที่ผ่านค่าความมั่นใจ

CARs	ค่าสนับสนุน (%)	ค่าความมั่นใจ (%)
{Female} => {Well}	40	4/6 = 66.7
{High} => {Well}	30	3/4 = 75
{True} => {Well}	40	4/5 = 80
{False} => {Sick}	30	3/5 = 60
{3} => {Well}	30	3/5 = 60
{Female, High} => {Well}	30	3/3 = 100

จากตารางที่ 11 แสดงกฎความสัมพันธ์แบบมีคลาส (CARs) ในทุกๆ ระดับของไอเท็มเซตที่ผ่านค่าสนับสนุนขั้นต่ำ และค่าความมั่นใจขั้นต่ำ

หลังจากตารางที่ 11 จะเป็นการจบในส่วนของการสร้างกฎความสัมพันธ์ (CBA-RG) ซึ่งในขั้นตอนถัดไปนั้นจะเป็นขั้นตอนในส่วนการสร้างโมเดลในการทำนาย (CBA-CB)

ตารางที่ 12 CBA-CB เรียงกฎตามค่าความมั่นใจ

Rid	CARs	ค่าสนับสนุน (%)	ค่าความมั่นใจ (%)
r1	{Female, High} => {Well}	30	3/3 = 100
r2	{True} => {Well}	40	4/5 = 80
r3	{High} => {Well}	30	3/4 = 75
r4	{Female} => {Well}	40	4/6 = 66.7
r5	{False} => {Sick}	30	3/5 = 60
r6	{3} => {Well}	30	3/5 = 60

ในตารางที่ 12 แสดงกฎความสัมพันธ์แบบมีคลาส (CARs) ที่ถูกจัดเรียงใหม่ตามค่าความมั่นใจเพื่อที่จะนำกฎเหล่านี้ไปใช้ในการสร้างโมเดลในการทำนายต่อไป

ตารางที่ 13 CBA-CB สร้างโมเดลในการทำนาย

รอบการทำงาน	โมเดลในการทำนาย	ความถูกต้อง (%)
1	R1 = {r1, Default class = Sick}	70
2	R2 = {r1, r2, Default class = Sick}	80
3	R3 = {r1, r2, Default class = Sick}	80
4	R4 = {r1, r2, Default class = Sick}	80
5	R5 = {{r1, r2, r5, Default class = Sick or Few or Well}}	80

จากตารางที่ 13 จะเห็นว่าอัลกอริทึม CBA-CB หรือส่วนในการสร้างโมเดลในการทำนาย จะมีการสร้างโมเดลออกมาจำนวนหลายชุด ดังนั้นจะต้องทำการเลือกชุดโมเดลที่ดีที่สุด โดยดูจากค่าเปอร์เซนต์ความถูกต้องของโมเดลชุดนั้นๆ แต่ถ้ามีหลายชุดโมเดลที่มีความถูกต้องเท่ากัน ก็จะพิจารณาตามหลักเกณฑ์ที่ว่า จะเลือกโมเดลที่มีเซตของกฎความสัมพันธ์แบบมีคลาส (CARs) ที่สั้นที่สุด ดังนั้นโมเดลที่สั้นที่สุดและมีความถูกต้องมากที่สุด นั่นคือโมเดลชุดที่ R2

CMAR อัลกอริทึม

Classification Based on Multiple Class-Association Rules (CMAR) เป็นอีกอัลกอริทึมหนึ่งที่ได้รับความนิยมสูง (Wenmin, 2001; Wenmin et al., 2001) เนื่องจากเป็นเทคนิคแรกที่เสนอให้พิจารณากฎความสัมพันธ์หลายๆกฎความสัมพันธ์พร้อมกันในการทำนายข้อมูล ซึ่งทำให้มีประสิทธิภาพความแม่นยำในการทำนายสูงกว่าทุกๆอัลกอริทึมที่ผ่านมา ของเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ (Janssens et al., 2003; Liu et al., 1998; Liu et al., 2001; Yin and Han, 2003) โดยจะแบ่งขั้นตอนออกเป็น 2 ส่วน ดังนี้

ส่วนที่ 1 ส่วนสืบค้นกฎความสัมพันธ์ที่มีคลาส

ในส่วนนี้จะมีขั้นตอนการทำงานคือ

1. มีชุดข้อมูลจำนวนหนึ่ง (Data Set) เป็นชุดข้อมูลที่เราสนใจจะวิเคราะห์ เราจะแบ่งข้อมูลชุดนี้ออกเป็น 2 ส่วน โดยส่วนแรกเรียกว่าเทรนนิ่งดาต้า (Training Data) ซึ่งมีไว้สอนให้

ระบบได้เรียนรู้ก่อนที่จะทำนายข้อมูล และ ส่วนที่ 2 เรียกว่าเทสติ้งดาต้า (Testing Data) ซึ่งมีไว้ทดสอบความแม่นยำ และหาค่าความแม่นยำของระบบ ซึ่งส่วนใหญ่แล้วจะแบ่งเป็นเทรนนิ่งดาต้าประมาณ 80% และเทสติ้งดาต้าอีกประมาณ 20%

2. นำเทรนนิ่งดาต้ามาเข้าส่วนการสืบค้นกฎความสัมพันธ์แบบมีคลาส ซึ่งจะแบ่งเป็น 3 กระบวนการดังนี้

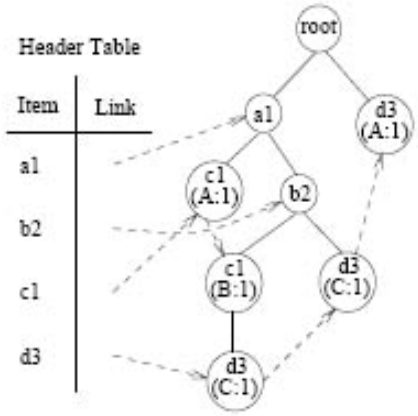
2.1 กระบวนการแรก ทำหน้าที่สร้างกฎความสัมพันธ์ที่มีคลาสทั้งหมดที่เป็นไปได้ออกมา ใช้เทคนิคเอฟพี-โกร์ท (FP-Growth) (Han et al., 2000) เทคนิคนี้จะเริ่มด้วยการสร้างต้นไม้ (ในเทคนิคนี้จะเรียกว่าเอฟพี-ทรี (FP-Tree)) ที่โหนดใบจะเก็บคลาสไว้ด้วย และเฮดเดอร์เทเบิ้ล (Header Table) เอาไว้เก็บตำแหน่งของแต่ละไอเท็มข้อมูล

สมมุติให้ข้อมูลประกอบด้วยดังตารางที่ 14

ตารางที่ 14 ตัวอย่างข้อมูลที่รับเข้ามาเพื่อทำ FP-Tree

Row-id	A	B	C	D	Class label
1	a1	b1	c1	d1	A
2	a1	b2	c1	d2	B
3	a2	b3	c2	d3	A
4	a1	b2	c3	d3	C
5	a1	b2	c1	d3	C

จะสามารถสร้างเอฟพี-ทรีได้ดังรูปที่ 4



ภาพที่ 4 ตัวอย่าง FP-Tree และ Header Table

จากนี้จะเข้าสู่กระบวนการไมน์นิ่งเพื่อหากฎความสัมพันธ์แบบมีคลาสออกมา โดยจะใช้วิธีรีเคอร์ซีฟเข้าไปในต้นไม้ซึ่งจะแบ่งเป็นกรณีตามลำดับของไอเท็มในเฮดเดอร์เทเบิ้ลจากล่างขึ้นบนจนครบ การไมน์นิ่งนี้จะตัดกฎที่ไม่สามารถผ่านค่าสนับสนุนขั้นต่ำ (Minimum support) และค่าความมั่นใจขั้นต่ำ (Minimum confidence) ออกไป โดยค่าสนับสนุนของกฎคือความถี่ของกฎความสัมพันธ์ที่พบ และค่าความมั่นใจของกฎคือค่าสนับสนุนของไอเท็มข้อมูลของกฎ หารด้วยค่าสนับสนุนทางซ้ายมือของกฎนั้นและคูณด้วย 100 เพื่อทำเป็นเปอร์เซ็นต์

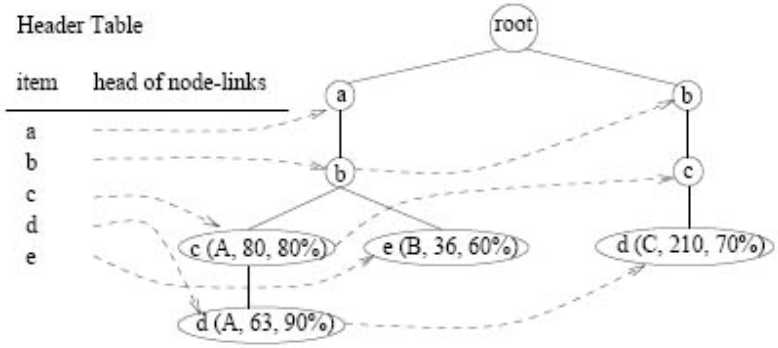
2.2 หลังจากที่ได้กฎความสัมพันธ์แบบมีคลาสมาแล้ว จะเข้าสู่กระบวนการที่ 2 ซึ่งจะมีโครงสร้างข้อมูลที่เรียกว่า ซีอาร์-ทรี (CR-Tree) เพื่อใช้ในการเก็บกฎความสัมพันธ์ที่ได้ไว้

สมมุติว่าได้กฎมา 4 กฎ ดังตารางที่ 15

ตารางที่ 15 ตัวอย่างกฎความสัมพันธ์ที่มีคลาส

Rule-id	Rule	Support	Confidence
1	abc -> A	80	80%
2	abcd -> A	63	90%
3	abe -> B	36	60%
4	bcd -> C	210	70%

จะสามารถเก็บลงซีอาร์-ทรีได้ดังรูปที่ 5



<u>ภาพที่ 5</u> ตัวอย่าง CR-Tree และ Header Table

2.3 สุดท้ายกระบวนการที่ 3 จะทำการตัดกฎความสัมพันธ์ที่อยู่ในซีอาร์-ทรี โดยจะตัดกฎความสัมพันธ์ที่ไม่มีประโยชน์ หรือกฎที่ไม่สามารถทำให้การทำนายข้อมูลดีขึ้นได้ออกไป

<u>ส่วนที่ 2 ส่วนจำแนกประเภทข้อมูลโดยใช้ค่า Chi-Square</u>

หลังจากที่ได้กฎความสัมพันธ์แบบมีคลาสทั้งหมดแล้ว จะเข้าสู่ส่วนที่ 2 ซึ่งในส่วนนี้จะนำเทสติ้งดาต้าเข้ามา และทำการจำแนกประเภทข้อมูล โดยจะนำกฎความสัมพันธ์ที่สามารถจำแนกข้อมูลชุดนี้ได้ออกมาทั้งหมดโดยจะจัดกลุ่มตามคลาส และคำนวณค่า Chi-Square ของแต่ละกลุ่มออกมา ถ้ากลุ่มไหนมีค่า Chi-Square มากที่สุด ก็จะทำนายคลาสของกลุ่มนั้น

<u>ตัวอย่างการคำนวณค่า Chi-Square</u>

การที่จะคำนวณค่า Chi-Square ของกลุ่มได้ เราจะต้องมีค่า Chi-Square ของแต่ละกฎความสัมพันธ์ในกลุ่มนั้นก่อนโดยจะคำนวณได้ดังนี้

สมมุติว่ามีกฎความสัมพันธ์ R: job = no => rejected มีค่าสนับสนุน (support) = 18 และค่าความมั่นใจ (confidence) = 60 % และจากที่หาได้จากเทรนนิ่งดาต้าที่มีจำนวน 500 เรคคอร์ด มีดังตารางที่16

ตารางที่ 16 The observed contingency of rule R.

R	Approved	rejected	Total
job = yes	438	32	470
job = no	12	18	30
Total	450	50	500

จากข้อมูลในตารางที่ 16 สามารถสร้างเป็นตารางใหม่ได้ ดังตารางที่17

ตารางที่ 17 The expected contingency of rule R.

R	Approved	rejected	Total
job = yes	423	47	470
job = no	27	3	30
Total	450	50	500

ซึ่งตารางที่ 17 แต่ละช่องได้มาจากการคำนวณจากสูตรดังต่อไปนี้

Expected [i, j] = (Row total[i] × Column total[j]) ÷ Total รวมทั้งหมด

และค่า Chi-Square ได้มาจากการคำนวณจากสูตร

$$\textbf{Chi-Square} = (\textbf{Observed} - \textbf{Expected})^2 \div \textbf{Expected}$$

ขั้นตอนการคำนวณมีดังนี้

เริ่มจาก แถวที่ 1, คอลัมภ์ที่ 1

Observed value (O) = 438

Expected value (E) = (Row total × Column total) ÷ Grand total

E = (470 ×450) ÷ 500 = 423

Chi-Square = $(O - E)^2 \div E$

Chi-Square = $((438 - 423)^2) \div 423$

Chi-Square = 0.531914893617021

Total Chi-Square now = 0.531914893617021

แถวที่ 1, คอลัมภ์ที่ 2

Observed value (O) = 32

Expected value (E) = (Row total × Column total) ÷ Grand total

E = (470 × 50) ÷ 500 = 47

Chi-Square = $(O - E)^2 \div E$

Chi-Square = $((32 - 47)^2) \div 47$

Chi-Square = 4.78723404255319

Total Chi-Square now = 5.31914893617021

แถวที่ 2, คอลัมภ์ที่ 1

Observed value (O) = 12

Expected value (E) = (Row total × Column total) ÷ Grand total

E = (30 × 450) ÷ 500 = 27

Chi-Square = $(O - E)^2 \div E$

Chi-Square = $((12 - 27)^2) \div 27$

Chi-Square = 8.33333333333333

Total Chi-Square now = 13.6524822695035

แถวที่ 2, คอลัมภ์ที่ 2

Observed value (O) = 18

Expected value (E) = (Row total × Column total) ÷ Grand total

E = (30 × 50) ÷ 500 = 3

Chi-Square = $(O - E)^2 \div E$

Chi-Square = $((18 - 3)^2) \div 3$

Chi-Square = 75

Total Chi-Square now = 88.6524822695036

ดังนั้นจะได้ค่า Chi-Square ของกฎ R = 88.6524822695036

หลังจากที่คำนวณของแต่ละกฎความสัมพันธ์หมดทุกกฎในกลุ่มแล้วก็จะนำแต่ละกฎมาเข้าสูตรดังต่อไปนี้

$$\mathbf{max\ X^2 = (min\{sup(P), sup(C)\} - sup(P)sup(C) \div |T|)^2|T|e}$$

โดยกฎแต่ละกฎจะมองในรูป R: P => C ซึ่ง X^2 แทนค่า Chi-Square

โดยที่ e = (1 ÷ (sup(P)sup(C))) + (1 ÷ ((sup(P) (|T| - sup(C)))) + (1 ÷ ((|T| - sup(P))(sup(C)))) + (1 ÷ ((|T| - sup(P))(|T| - sup(C))))

ค่า sup(P) คือ ค่าสนับสนุนของไอเท็มข้อมูลทางซ้ายของกฎความสัมพันธ์

ค่า sup(C) คือ ค่าสนันสนุนของไอเท็มรวมคลาสปลายทางของกฎความสัมพันธ์

ค่า |T| คือ จำนวนเรคคอร์ดในเทรนนิ่งดาต้า

จากสูตร $\mathbf{max\ X^2}$ จะเป็นการคำนวณค่า Chi-Square ของแต่ละกฎนั้นอีกทีเพื่อเป็นการถ่วงค่าน้ำหนัก และหลังจากนั้น จะนำค่าที่คำนวณได้มารวมกันโดยใช้สูตรดังต่อไปนี้

$$\mathbf{Chi\text{-}Square\ of\ Group} = \sum \mathbf{(X^2X^2) \div maxX^2}$$

หลังจากที่แต่ละกลุ่มของคลาสได้ทำการคำนวณตามสูตรที่ได้กล่าวมาแล้ว คลาสที่จะทำนายก็คือ คลาสที่มีค่าผลรวมของค่า **Chi-Square of Group** มากที่สุดนั่นเอง

หลังจากทราบวิธีการและขั้นตอนของงานวิจัยที่เกี่ยวข้องแล้ว ในส่วนนี้จะทำการสรุปและเปรียบเทียบ ข้อดีและข้อเสียของแต่ละอัลกอริทึม

สรุปข้อดีและข้อเสียของอัลกอริทึมที่เกี่ยวข้องกับงานวิจัย

อัลกอริทึม CBA (Liu et al., 1998) เป็นงานวิจัยแรก ที่เสนอเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ โดยมีส่วนการทำงานที่สำคัญอยู่ 2 ส่วน คือ ในส่วนของการสร้างกฎความสัมพันธ์ (Rule generator phase) และในส่วนของการสร้างโมเดลในการทำนาย (Classifier builder phase โดยอัลกอริทึมดังกล่าวมีข้อดีและข้อเสีย ดังต่อไปนี้

ข้อดีของอัลกอริทึม CBA

- เป็นอัลกอริทึมที่สามารถศึกษาทำความเข้าใจได้ง่าย
- มีการสร้างโมเดลของกฎความสัมพันธ์แบบมีคลาสเอาไว้ก่อน เมื่อเวลามีข้อมูลที่ต้องการจะทำนาย จึงสามารถทำนายคลาสได้อย่างรวดเร็ว

ข้อเสียของอัลกอริทึม CBA

- ใช้วิธีการกำจัดกฎโดยใช้ค่าความมั่นใจ ซึ่งกฎที่ไม่มีประโยชน์ยังคงไม่สามารถกำจัดออกไปได้ทั้งหมด
- การเรียงกฎโดยใช้ค่าความมั่นใจขั้นต่ำเพียงค่าเดียว ยังมีข้อผิดพลาดอยู่
- ในส่วนของการทำนายข้อมูล ใช้กฎเพียงกฎเดียวในการทำนายข้อมูล
- เนื่องจากมีการสร้างโมเดลของกฎความสัมพันธ์แบบมีคลาสเอาไว้ก่อน เมื่อมีการเพิ่มขึ้นของข้อมูล จึงต้องทำการปรับปรุงในส่วนของกฎ และโมเดลใหม่ทุกครั้ง ดังนั้นจึงไม่เหมาะกับ Incremental data

อัลกอริทึม CMAR (Wenmin, 2001; Wenmin et al., 2001) เป็นงานวิจัยที่พัฒนาเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ที่มีประสิทธิภาพและความแม่นยำ โดยพัฒนาขึ้นจากอัลกอริทึม CBA ซึ่งมีข้อดีและข้อเสีย ดังต่อไปนี้

ข้อดีของอัลกอริทึม CMAR

- เป็นอัลกอริทึมที่ให้ความแม่นยำในการทำนายข้อมูลสูง
- ใช้อัลกอริทึม FP-growth ซึ่งมีประสิทธิภาพในการสร้างกฎ
- มีการใช้โครงสร้างข้อมูล FP-tree ที่มีประสิทธิภาพในการเก็บข้อมูล ซึ่งทำให้ประหยัดหน่วยความจำลงได้
- เนื่องจากไม่ต้องมีการสร้างโมเดลของกฎความสัมพันธ์แบบมีคลาสเอาไว้ก่อน เมื่อมีการเพิ่มขึ้นของข้อมูล จึงไม่ต้องมีการปรับปรุงตัวโมเดล เพียงแต่ปรับปรุงในส่วนของกฎเท่านั้น ดังนั้นจึงเหมาะกับ Incremental data

ข้อเสียของอัลกอริทึม CMAR

- เป็นอัลกอริทึมที่มีความซับซ้อน ยากต่อการศึกษาทำความเข้าใจ
- ในการกำจัดกฎที่ไม่มีประโยชน์ ต้องมีการกำหนดค่าขั้นต่ำ ซึ่งในการกำหนดค่าขั้นต่ำนี้ ไม่มีตัวเลขที่กำหนดแน่นอนตายตัว ดังนั้นจึงยากในการกำหนดค่าขั้นต่ำให้ได้ผลที่ดีที่สุด
- ในส่วนของการทำนายข้อมูล ใช้ทุกๆ กฎที่สอดคล้องกับข้อมูลนั้นๆ มาเข้าสูตรในการคำนวณ โดยให้ทุกๆกฎมีความสำคัญเท่ากัน ซึ่งที่จริงแล้วกฎที่มีความยาวมาก น่าจะมีความสำคัญกว่ากฎที่สั้นๆ
- ไม่มีการสร้างโมเดลของกฎความสัมพันธ์แบบมีคลาสเอาไว้ก่อน เมื่อเวลามีข้อมูลที่ต้องการจะทำนาย จึงค่อยนำกฎที่สอดคล้องกับข้อมูลนั้นมาเข้าสูตรในการคำนวณก่อน แล้วจึงค่อยทำนายคลาสออกไป ซึ่งจะเสียเวลาในส่วนนี้

อุปกรณ์และวิธีการ

อุปกรณ์

ฮาร์ดแวร์

1. เครื่องคอมพิวเตอร์โน๊ตบุค 1 เครื่อง ประกอบด้วยอุปกรณ์ดังต่อไปนี้
   - ซีพียู (CPU) เพนเทียม IV ความเร็ว1.7 GHz
   - หน่วยความจำหลัก 1 GB
   - ฮาร์ดดิกส์ขนาด 80 GB
   - การ์ดแลน

ซอฟต์แวร์

1. ระบบปฏิบัติการ Windows XP Professional
2. ระบบปฏิบัติการ Linux
3. g++ คอมไพเลอร์
4. Editor C++
5. Putty (File transfer protocol)

วิธีการ

ภาพรวมของระบบ

ดังที่กล่าวมาแล้วในส่วนของความรู้พื้นฐานว่า เทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ (Associative classification) แบ่งส่วนการทำงานออกเป็น 2 ส่วน คือ ส่วนของการสร้างกฎความสัมพันธ์ กับ ส่วนของการสร้างโมเดลในการทำนาย ซึ่งจากการทดลองพบว่า ปัญหาของระบบเกิดขึ้นทั้งในส่วนการสร้างกฎ และในส่วนการสร้างโมเดลในการทำนาย โดยปัญหาในส่วนของการสร้างกฎความสัมพันธ์ คือการสร้างกฎความสัมพันธ์ที่มีจำนวนมากจนเกินไป ส่งผลให้เกิดกฎที่ไม่มีประโยชน์ และกฎที่ซ้ำซอน ซึ่งทำให้ประสิทธิภาพของระบบเสียไป ทั้งด้านความแม่นยำของโมเดลที่จะต้องใช้กฎเหล่านั้นในการสร้างรวมถึงจะต้องเสียเวลาในการประมวลผล และยังต้องเสียพื้นที่ในการเก็บกฎที่ไม่มีประโยชน์เหล่านั้นอีกด้วย ดังนั้นในการเพิ่มประสิทธิภาพใน

ส่วนของการสร้างกฎความสัมพันธ์ ในวิทยานิพนธ์เล่มนี้จะนำเสนอวิธีใหม่ (Hanchodchung et al., 2006) โดยใช้วิธีการกำจัดกฎความสัมพันธ์แบบมีคลาสที่ซ้ำซ้อนออกไป และสร้างเฉพาะกฎความสัมพันธ์ที่เรียกว่า Essential Class-Association Rules หรือ ECARs เท่านั้น ซึ่งในการกำจัดกฎความสัมพันธ์แบบมีคลาสที่ซ้ำซ้อนและไม่มีประโยชน์เหล่านั้นออกไป ก็เพื่อที่จะให้เหลือแต่กฎที่มีคุณภาพซึ่งจะต้องนำไปใช้ในส่วนการสร้างโมเดลให้มีประสิทธิภาพดีต่อไป แต่เนื่องจากในส่วนการสร้างโมเดลก็มีปัญหาอยู่เช่นกัน ซึ่งก็คือปัญหาในการพิจารณากฎเพื่อทำนายข้อมูล ซึ่งในอัลกอริทึม CBA (Liu et al., 1998) จะมีวิธีการพิจารณากฎความสัมพันธ์ที่ละกฎเท่านั้น (Single rule prediction) ซึ่งในการทำนายก็มักจะเกิดข้อผิดพลาดได้ ยกตัวอย่างเช่น ถ้ามีกฎความสัมพันธ์ CARs ตามลำดับต่อไปนี้

ตารางที่ 18 ตัวอย่างกฎความสัมพันธ์ที่สอดคล้อง (match) กับข้อมูลทดสอบระบบ

โมเดลในการทำนายข้อมูล		
กฎความสัมพันธ์	ค่าสนับสนุน	ค่าความมั่นใจ
A, C => X	1%	80%
A, D => Y	90%	75%
A, E => Y	60%	75%

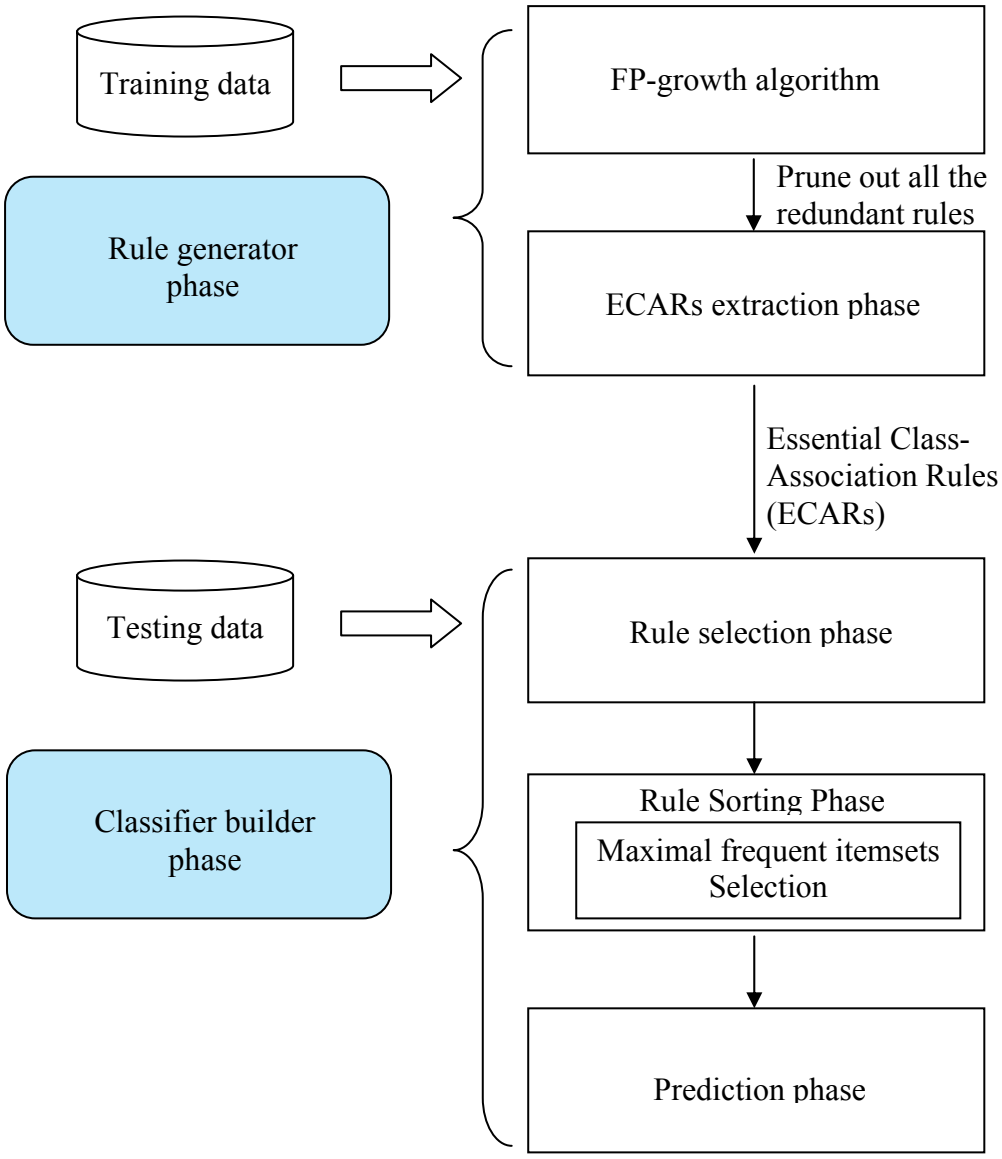
จากตารางที่ 18 แสดงถึงกฎความสัมพันธ์ (CARs) ที่สอดคล้องกับข้อมูลทดสอบระบบ (Testing data) นั่นคือ ACDE -> Y ซึ่งหมายความว่า กฎที่สอดคล้องเหล่านั้นคือ กฎที่เป็นเซต หรือซับเซตของข้อมูลทดสอบระบบนั่นเอง ซึ่งหลังจากได้กฎทั้งหมดที่สอดคล้องกับข้อมูลทดสอบระบบแล้ว จึงนำกฎเหล่านั้นมาทำการจัดเรียงตามค่าความมั่นใจ (Confidence) ก่อน จากนั้นระบบจะทำนาย class X เนื่องจากกว่า กฎที่มีลำดับสูงที่สุดอยู่ใน Class X ซึ่งถ้าดูจากคำตอบที่ถูกต้องแล้วจะต้องตอบคลาส Y ดังนั้นโมเดลจึงตอบผิด ซึ่งถ้าทำการพิจารณาดูดีๆแล้วจะพบว่ากฎในระดับรองลงมา กลับมีความน่าสนใจกว่า เพราะว่ากฎแรกมีค่า ความมั่นใจ 80% แต่มีค่าสนับสนุนเพียงแค่ 1% เท่านั้นเอง ส่วนกฎที่สองนั้นมีค่าความมั่นใจ 75% ซึ่งน้อยกว่าไม่มาก แต่มีค่าสนับสนุนสูงถึง 90% ดังนั้นการใช้วิธีการพิจารณากฎโดยยึดหน่วยวัดใดหน่วยวัดหนึ่งคงไม่ใช่วิธีการที่ดีนัก ดังนั้นจึงมีผู้เสนอ อัลกอริทึม CMAR (Wenmin, 2001; Wenmin et al., 2001) ซึ่งเสนอให้ใช้วิธีการพิจารณากฎความสัมพันธ์หลายๆกฎพร้อมกัน (Multiple rules prediction) เพื่อช่วยแก้ปัญหาดังกล่าว แต่ก็ยังไม่สามารถแก้ปัญหาได้หมด เนื่องจากการพิจารณากฎหลายๆกฎพร้อมกันนั้นได้ให้

ความสำคัญกับกฎทุกกฎเท่ากัน ซึ่งในความเป็นจริงแล้วกฎที่ประกอบด้วยไอเท็มเซตที่ยาวๆ น่าจะเป็นกฎที่มีความน่าสนใจมากกว่ากฎที่ประกอบด้วยไอเท็มเซตสั้นๆ ดังนั้นในแก้ปัญหาดังกล่าวในส่วนของการสร้างโมเดลในการทำนาย จึงได้เสนอวิธีการพิจารณากฎความสัมพันธ์แบบใหม่ (วีระพล และคณะ, 2548; Hanchodchung et al., 2006) โดยเสนอใช้วิธีการพิจารณากฎโดยให้ความสำคัญกับกฎที่เรียกว่า Maximal frequent itemsets เป็นอันดับแรกก่อน ซึ่งในรายละเอียดจะอธิบายในหัวข้อวิธีการในส่วนการสร้างโมเดลในการทำนายต่อไป

ก่อนที่จะทำความเข้าใจในส่วนรายละเอียดนั้น สามารถที่จะดูภาพรวมของระบบได้จากภาพที่ 6 โดยระบบนี้ใช้ชื่อว่า Classification Based on Essential Class-Association Rules (CBEAR) (Hanchodchung et al., 2006)

โดยในส่วนของการสร้างกฎความสัมพันธ์ (Rule generator phase) จะใช้ข้อมูลที่เรียกว่าข้อมูลสอนระบบ (Training data) ซึ่งอัลกอริทึมที่ใช้ในการสร้างกฎความสัมพันธ์ คือ อัลกอริทึม FP-growth (Han et al., 2000) และขั้นตอนที่เพิ่มขึ้นมาในส่วนนี้ก็คือ การกำจัดกฎความสัมพันธ์แบบมีคลาสที่ซ้ำซ้อน (ECARs extraction phase) ซึ่งกฎความสัมพันธ์แบบมีคลาส ที่ผ่านส่วนนี้แล้วจะเรียกว่า ECAR (Essential Class-Association Rules)

ในส่วนของการสร้างโมเดลในการทำนาย (Classifier builder phase) จะใช้ข้อมูลอีกส่วนที่เรียกว่า ข้อมูลทดสอบระบบ (Testing data) ซึ่งในส่วนนี้จะประกอบไปด้วยขั้นตอนต่อไปนี้คือ ขั้นตอนในการเลือกกฎความสัมพันธ์แบบมีคลาส (Rule selection phase) ซึ่งจะทำการเลือกเฉพาะกฎที่สอดคล้องหรือเป็นสับเซตของข้อมูลทดสอบระบบเท่านั้น ขั้นตอนถัดมาคือ การจัดเรียงกฎความสัมพันธ์แบบมีคลาส (Rule sorting phase) ซึ่งในส่วนนี้จะนำกฎความสัมพันธ์มาจัดเรียงใหม่ โดยจะใช้หลักการของ Maximal frequent itemsets (Pasquier et al., 1999a; Pasquier et al., 1999b; Pei et al., 2000) ในการเลือกกฎความสัมพันธ์แบบมีคลาสที่ยาวที่สุด ไปใช้ในการทำนายข้อมูลซึ่งอยู่ในขั้นตอนถัดไป (Prediction phase)



ภาพที่ 6 ภาพรวมของระบบการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ CBEAR

การเตรียมข้อมูล (Data Preprocessing)

ฐานข้อมูลที่นำมาใช้ ได้มากจาก UCI Machine Learning Repository (Blake and Merz, 1998) ซึ่งเป็นข้อมูลมาตรฐานที่ใช้ทดสอบความแม่นยำของระบบการจำแนกประเภทข้อมูลโดยเฉพาะ แต่เนื่องจากข้อมูลเหล่านั้น ประกอบไปด้วยข้อมูลหลายรูปแบบ ดังตารางที่ 19

ตารางที่ 19 ชนิดรูปแบบของข้อมูลต่างๆ

ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง
Categorical	ข้อมูลที่เป็นช่วงๆ แบ่งแยกอย่างชัดเจน	ข้อมูล สี เช่น สีแดง, สีดำ, สีขาว เป็นต้น ข้อมูล รสชาติ เช่น รสเปรี้ยว, รสหวาน, รสขม เป็นต้น
Binary	ข้อมูลที่ประกอบด้วยค่า 2 ค่าเท่านั้น	ข้อมูล เพศ เพศชาย = 0, เพศหญิง = 1 ข้อมูล ใช่- ไม่ใช่ ใช่ = 0 , ไม่ใช่ = 1
Continuous	ข้อมูลที่เป็นค่าต่อเนื่องรวมจุดทศนิยม	ข้อมูล น้ำหนัก-ส่วนสูง 65 กิโลกรัม, 65.5 กิโลกรัม, 68.5 กิโลกรัม, … , 70.5 กิโลกรัม 170 เซนติเมตร, 170 ½ เซนติเมตร, 171 เซนติเมตร
Numerical	ข้อมูลที่เป็นค่าต่อเนื่องไม่รวมจุดทศนิยม	สินค้าจำนวน (ชิ้น) 1 ชิ้น, 2 ชิ้น, 3 ชิ้น, … N ชิ้น ไม่มี 1 ½ ชิ้น

เนื่องจากข้อมูลที่สามารถจะนำมาใช้ในเทคนิค การจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์นั้น จะต้องเป็นข้อมูลที่สามารถที่จะแยกแยะเป็นประเภทได้ ซึ่งจากตารางที่ 19 จะเห็นว่า ข้อมูลที่สามารถที่จะแยกแยะได้นั่นคือ ข้อมูลชนิด Categorical และข้อมูลชนิด Binary นั่นเอง ส่วนข้อมูลที่เป็น Continuous และ Numerical จะต้องผ่านกระบวนการแปลงข้อมูลให้เป็นค่าไม่ต่อเนื่อง หรือที่เรียกว่า Discretization เสียก่อน (Liu et al., 1998) โดยวิธีการแปลงข้อมูลให้เป็นค่าไม่ต่อเนื่องก็มีหลายวิธี (Dougherty et al., 1995, Fayyad and Irani, 1993; Kohavi and Sahami, 1996) เช่น วิธี Density method ซึ่งเป็นวิธีการแปลงข้อมูลที่เป็นค่าต่อเนื่อง ให้เปลี่ยนเป็นค่าไม่ต่อเนื่อง (Discrete) โดยดูที่ความหนาแน่นของข้อมูล โดยการทำงานจะเริ่มโดย การรับข้อมูลและทำการจัดเรียงลำดับข้อมูลเหล่านั้นใหม่โดยจัดเรียงให้เป็นค่าต่อเนื่องกัน จากนั้นจะทำการแบ่งข้อมูลออกเป็นช่วงๆ โดยจะมีจำนวนข้อมูลเท่ากันในแต่ละช่วง ซึ่งการหาตำแหน่งของจุดแบ่งสามารถหาได้จากสูตร (Dougherty et al., 1995, Fayyad and Irani, 1993) ต่อไปนี้

$$\text{ตำแหน่งของจุดแบ่ง} = \left[| S_f | \times \frac{i}{k+1} \right]$$

โดยที่ S_f คือ ชุดของข้อมูลที่เป็นค่าต่อเนื่อง

i คือ ลำดับของการหาค่าของจุดแบ่ง โดยจุดแรก i จะมีค่าเป็น 1 และจุดต่อไปก็จะเพิ่มค่า i ขึ้นทีละ 1 ตามลำดับ

k คือ จำนวนของจุดแบ่งที่ต้องการแบ่ง

ตัวอย่างเช่น กำหนดให้ข้อมูลทั้งหมดที่ทำการจัดเรียงแล้วคือ S_f = 1, 2, 4, 6, 7, 8, 9, 11, 12 โดยต้องการแบ่งข้อมูลออกเป็น 3 ช่วงนั่นคือจะมีจุดแบ่งทั้งหมด 2 จุดนั่นคือ

$$\text{ตำแหน่งของจุดแบ่งจุดแรก} = 9 \times 1 / (2 + 1) = 3$$

ดังนั้น จุดแบ่งจุดแรกคือ ค่าลำดับที่ 3 ใน S_f นั่นคือ 4

$$\text{ตำแหน่งของจุดแบ่งจุดที่สอง} = 9 \times 2 / (2 + 1) = 6$$

ดังนั้น จุดแบ่งจุดที่สองคือ ค่าลำดับที่ 6 ใน S_f นั่นคือ 8

สุดท้ายข้อมูลจะถูกแบ่งออกเป็น 3 ช่วงคือ ช่วงตั้งแต่ 1 ถึง 4 ช่วงที่มีค่ามากกว่า 4 ถึง 8 และช่วงที่มีค่ามากกว่า 8 ถึง 12

นอกจากนี้แล้วยังมีข้อมูลที่เรียกว่า Missing value ซึ่งใช้สัญลักษณ์ " ? " ปรากฏอยู่ในฐานข้อมูลอีกด้วย ซึ่งวิธีในการจัดการกับข้อมูลประเภทนี้ก็มีอยู่หลายวิธี (Blake and Merz, 1998) ซึ่งในวิทยานิพนธ์เล่มนี้ใช้วิธีการแบบง่าย คือการตัดแถวที่ปรากฏ Missing value นั้นออก เหตุผลที่ใช้วิธีนี้เนื่องจากว่า ฐานข้อมูลที่นำมาใช้นั้นมีแถวที่ปรากฏ Missing value อยู่ไม่ถึง 5% ดังนั้นวิธีการตัดแถวที่มี Missing value ออกจึงไม่ส่งผลกระทบต่อฐานข้อมูลนั้นๆ

<u>วิธีการในส่วนการสร้างกฎความสัมพันธ์</u>

เนื่องจากในการสร้างกฎความสัมพันธ์นั้น ยังคงใช้อัลกอริทึม FP-growth (Han et al., 2000) อยู่ซึ่งก็เหมือนกับอัลกอริทึม CMAR (Wenmin, 2001; Wenmin et al., 2001) ที่ได้กล่าวรายละเอียดเอาไว้ในส่วนของงานวิจัยที่เกี่ยวข้อง เพียงแต่มีการเพิ่มส่วนที่เรียกว่าการบีบอัดกฎ (Rule compression) เข้ามาเพื่อช่วยกำจัดกฎที่ซ้ำซ้อนและไม่มีประโยชน์ออกไป (Baralis and Chiusano, 2004; Baralis et al., 2004) โดยวิธีการกำจัดกฎที่ซ้ำซ้อนนี้จะใช้วิธีตามนิยามที่ 1 ซึ่งจะกล่าวต่อไป โดยหลังจากกำจัดกฎความสัมพันธ์ที่ซ้ำซ้อนออกไปแล้ว ก็จะเรียกกฎความสัมพันธ์แบบมีคลาสที่เหลืออยู่ว่า Essential Class-Association Rules (ECARs) (Hanchodchung et al., 2006)

<u>ความหมายของ Essential Class-Association Rules</u>

กฎความสัมพันธ์แบบมีคลาสที่เรียกว่า Essential Class-Association Rules (ECARs) คือกฎความสัมพันธ์แบบมีคลาสที่ไม่มีความซ้ำซ้อนเหลืออยู่ จากการกำจัดกฎความสัมพันธ์ที่ซ้ำซ้อน (Redundant rule) ทำให้ขนาดของกฎความสัมพันธ์มีขนาดลดลง และยังสามารถที่จะนำกลุ่มของ Essential Class-Association Rules มาใช้แทนกฎความสัมพันธ์แบบเดิมได้โดยไม่ทำให้ขาดความสมบูรณ์ของข้อมูลแต่อย่างใด แถมยังเป็นการกำจัดความซ้ำซ้อนของข้อมูล ซึ่งจะส่งผลให้ลดความผิดพลาดในการสร้างโมเดลในการทำนาย ทำให้โมเดลที่ได้มีความแม่นยำเพิ่มมากขึ้นอีกด้วย

นิยามที่ 1 (กฎความสัมพันธ์ที่ซ้ำซ้อน)

กำหนดให้ r_i เป็นกฎความสัมพันธ์แบบมีคลาส ซึ่งมี X เป็น itemsets ทางฝั่งซ้ายมือของกฎ และมีคลาสปลายทางเป็น c_i และ r_j เป็นกฎความสัมพันธ์แบบมีคลาส ซึ่งมี Y เป็น itemsets ทางฝั่งซ้ายมือของกฎ และมีคลาสปลายทางเป็น c_j จะบอกได้ว่า r_i เป็นกฎความสัมพันธ์แบบมีคลาสที่ซ้ำซ้อนของ r_j ก็ต่อเมื่อตรงตามเงื่องไขดังต่อไปนี้ (1) Y เป็น subset ของ X (2) คลาส c_i เท่ากับคลาส c_j และ (3) ค่าสนับสนุนของ X เท่ากับ ค่าสนับสนุนของ Y และค่าสนับสนุนของ X ที่มีคลาสปลายทางเป็น c_i เท่ากับค่าสนับสนุนของ Y ที่มีคลาสปลายทางเป็น c_j

กลุ่มของกฎความสัมพันธ์แบบมีคลาสหลังจากที่กำจัดกฎที่ซ้ำซ้อนออกหมดแล้วเราจะเรียกกลุ่มของกฎนั้นว่า Essential class-association rule (ECARs) ซึ่งสามารถดูจากตัวอย่างต่อไปนี้

ตัวอย่าง การกำจัดกฎความสัมพันธ์แบบมีคลาสที่ซ้ำซ้อน

ตารางที่ 20 ตัวอย่างกฎความสัมพันธ์แบบมีคลาส ก่อนกำจัดกฎที่ซ้ำซ้อน

Rid	กฎความสัมพันธ์	ค่าสนับสนุนของ Itemsets (%)	ค่าสนับสนุนของกฎ (%)
r1	{A, B, C, D} => {X}	{A, B, C, D} = 70	{A, B, C, D, X} = 60
r2	{A, B, C} => {X}	{A, B, C} = 70	{A, B, C, X} = 60
r3	{A, B} => {X}	{A, B} = 60	{A, B, X} = 55
r4	{A} => {X}	{A} = 65	{A, X} = 50
r5	{B, D, E} => {Y}	{B, D, E} = 60	{B, D, E, Y} = 60
r6	{B, D} => {Y}	{B, D} = 60	{B, D, Y} = 40
r7	{B, E} => {Y}	{B, E} = 60	{B, E, Y} = 60
r8	{A, C, E} => {Z}	{A, C, E} = 30	{A, C, E, Z} = 30
r9	{A, E} => {Z}	{A, E} = 40	{A, E, Z} = 40
r10	{E} => {Z}	{E} = 40	{E, Z} = 40

ตารางที่ 21 ตัวอย่างกฎความสัมพันธ์แบบมีคลาส หลังกำจัดกฎที่ซ้ำซ้อน

Rid	กฎความสัมพันธ์	ค่าสนับสนุนของ Itemsets (%)	ค่าสนับสนุนของกฎ (%)
r2	{A, B, C} => {X}	{A, B, C} = 70	{A, B, C, X} = 60
r3	{A, B} => {X}	{A, B} = 60	{A, B, X} = 55
r4	{A} => {X}	{A} = 65	{A, X} = 50
r6	{B, D} => {Y}	{B, D} = 60	{B, D, Y} = 40
r7	{B, E} => {Y}	{B, E} = 60	{B, E, Y} = 60
r8	{A, C, E} => {Z}	{A, C, E} = 30	{A, C, E, Z} = 30
r10	{E} => {Z}	{E} = 40	{E, Z} = 40

จากตารางที่21 จะเห็นว่ากฎความสัมพันธ์แบบมีคลาสที่ถูกกำจัดออกไป คือ กฎที่ r1, r5 และ r9 เนื่องจากกฎเหล่านั้นเป็นกฎที่ซ้ำซ้อนกับกฎความสัมพันธ์อื่น คือ กฎ r1 ซ้ำซ้อนกับกฎ r2 ตามนิยามที่1 และกฎ r5 ซ้ำซ้อนกับกฎ r7 ตามนิยามที่1 และสุดท้ายกฎ r9 ซ้ำซ้อนกับกฎ r10 ตามนิยามที่1

วิธีการในส่วนการสร้างโมเดลในการทำนาย

ในส่วนของการสร้างโมเดลในการทำนายนั้น เราจะใช้วิธีการพิจารณากฎความสัมพันธ์แบบมีคลาสโดยจะทำเลือกเฉพาะกฎความสัมพันธ์ที่เป็น Maximal frequent itemset (Pasquier et al., 1999a; Pasquier et al., 1999b; Pei et al., 2000) ก่อนเพื่อใช้ในการทำนายข้อมูล โดยความหมายของ Maximal frequent itemsets จะกล่าวในนิยามที่ 2 ต่อไป ซึ่งหลังจากนั้นก็จะนำกลุ่มของกฎที่ได้ไปเข้าสูตรในการคำนวณ เพื่อที่จะทำนายคลาสต่อไป ซึ่งขั้นตอนวิธีในส่วนของการสร้างโมเดลในการทำนายนั้น จะเป็นส่วนที่มีการคิดขึ้นมาใหม่ทั้งหมด ตั้งแต่การเลือกพิจารณาเฉพาะกฎที่ยาวที่สุดก่อน รวมถึงสูตรที่ใช้ในการคำนวณ โดยในรายละเอียดหัวข้อนี้ จะอธิบายถึงความหมายของ Maximal frequent itemsets รวมถึงนิยาม และขั้นตอนวิธีการจัดเรียงกฎความสัมพันธ์แบบมีคลาส

เพื่อนำไปใช้ในการทำนาย และสุดท้ายคือการนำเสนอสูตรที่ใช้คำนวณกลุ่มของกฎความสัมพันธ์แบบมีคลาสในการทำนายข้อมูล

ความหมายของ Maximal Frequent Itemsets

เนื่องจากกฎความสัมพันธ์แบบมีคลาสที่ถูกสร้างขึ้นมาก็มีหลากหลายระดับความยาวของกฎ ดังนั้นกฎในแต่ละระดับความยาว น่าจะมีความสำคัญไม่เท่ากัน กฎที่ความยาวมากที่สุด น่าจะมีความสำคัญมากที่สุดด้วย เนื่องจากว่ากฎนั้นเป็นกฎที่มีความใกล้เคียงหรือมีความเหมือนกับชุดคำถามหรือข้อมูลทดสอบระบบมากที่สุด ดังนั้นจึงเรียกกฎที่มีความยาวมากที่สุดว่า Maximal frequent itemsets โดยกฎที่มีความยาวรองๆลงมาก็น่าจะมีความสำคัญลดหลั่นกันลงมาด้วย

นิยามที่ 2 (Maximal frequent itemset)

กำหนดให้ r_i เป็นกฎความสัมพันธ์แบบมีคลาส ซึ่งมี X เป็น itemsets ทางฝั่งซ้ายมือของกฎ และมีคลาสปลายทางเป็น c_i และ r_j เป็นกลุ่มของกฎความสัมพันธ์แบบมีคลาส ซึ่งมี Y เป็น itemsets ทางฝั่งซ้ายมือของกฎเหล่านั้น และมีคลาสปลายทางเป็น c_j จะบอกได้ว่า r_i เป็น Maximal frequent itemsets ของ r_j ก็ต่อเมื่อ X ไม่เป็นซับเซตของ Y ใดๆ

จากนิยามสรุปได้ว่า Frequent itemset หนึ่งๆจะเป็น Maximal ก็ต่อเมื่อ มันไม่เป็น Subset ของ frequent itemset ตัวอื่น เช่น กำหนดให้เซตของกฎเป็นดังนี้ {A, B, C, AB, BC, AC, ABC} กฎที่เป็น Maximal คือ ABC เพราะว่าไม่มี Super set ของ ABC ที่ผ่านค่าสนับสนุน หรือ ABC เป็นกฎที่ยาวที่สุดนั่นเอง

ขั้นตอนการจัดเรียงกฎความสัมพันธ์

ในส่วนของการสร้างโมเดลในการทำนาย (Classifier builder) นี้ได้เพิ่มส่วนของการเรียงลำดับกฎความสัมพันธ์ (Rule sorting) เพื่อพิจารณาเฉพาะกฎที่เป็น Maximal เท่านั้น เนื่องจากแนวคิดที่ได้เคยกล่าวเอาไว้แล้วว่า กฎที่ยาวที่สุดจะเป็นกฎที่มีความเหมือนหรือคล้ายคลึงกับตัวข้อมูลทดสอบระบบมากที่สุด เพราะฉะนั้นในการเรียงลำดับกฎความสัมพันธ์แบบมีคลาส จะเรียงตามความยาวของกฎจากมากไปน้อย ซึ่งความยาวของกฎนั้นสามารถจะดูได้จาก Itemsets ทางฝั่ง

ซ้ายมือของกฎนั่นเอง โดยหลังจากได้จัดเรียงกฎตามความยาวของ Itemsets แล้ว ในแต่ละความยาวของ Itemsets ก็จะจัดเรียงกฎตามคลาสปลายทางอีกทีนึง เพื่อที่จะได้ง่ายในการเรียกดูและค้นหาและนำกฎเหล่านั้นไปใช้ โดยตัวอย่างการจัดเรียงกฎ สามารถดูได้จากตารางที่ 22

<u>ตารางที่ 22</u> ตัวอย่างการจัดเรียงกฎความสัมพันธ์

คลาส / ระดับความยาวของกฎ	X	Y	Z
3	A,B,C -> X A,B,D -> X	B,D,E -> Y B,D,F -> Y B,E,F -> Y	A,E,F -> Z
2	A,B -> X A,C -> X A,D -> X B,C -> X B,D -> X	B,D -> Y B,E -> Y B,F -> Y D,E -> Y D,F -> Y E,F -> Y	A,E -> Z A,F -> Z E,F -> Z
1	A -> X B -> X C -> X D -> X	B -> Y D -> Y E -> Y F -> Y	A -> Z E -> Z F -> Z

จากตารางที่ 22 จะเห็นได้ว่ากฎความสัมพันธ์แบบมีคลาสที่มีความยาวของ Itemsets มากที่สุดนั่นคือ ความยาว 3 ดังนั้นจะถูกจัดเรียงให้อยู่ส่วนบนทั้งหมด และยังจัดเรียงกฎเหล่านั้นให้อยู่ตามคลาสปลายทางอีกด้วย ทั้งนี้เพื่อให้ง่ายในการที่จะสืบค้นเพื่อนำไปใช้ในการคำนวณต่อได้ จากนั้นกฎที่มีความยาวของ Itemsets รองลงมาก็จะถูกจัดเรียงให้อยู่ถัดลงมาตามลำดับความยาวของ Itemsets นั้นๆ

<u>สูตรการคำนวณกลุ่มของกฎในการทำนายข้อมูล</u>

หลังจากที่มีการเรียงลำดับกฎตามความยาวและตามคลาสแล้ว ในวิทยานิพนธ์เล่มนี้ได้นำเสนอสูตรการคำนวณ จำนวน 4 วิธีดังต่อไปนี้

สูตรที่ 1 Max $\sum_{i=1}^{n} Count\_rule(r_i)$

สูตรที่ 2 Max $\sum_{i=1}^{n} Support(r_i) \div n$

สูตรที่ 3 Max $\sum_{i=1}^{n} Confidence(r_i) \div n$

เมื่อ r_i คือสมาชิกแต่ละตัวใน Class นั้นๆ

n คือจำนวนสมาชิกทั้งหมดของ Class นั้นๆ

$Count\_rule(r_i)$ คือการนับจำนวนกฎในแต่ละ Class

$Support(r_i)$ คือ การหาค่าสนับสนุนของแต่ละกฎใน Class นั้นๆ

$Confidence(r_i)$ คือ การหาค่าความมั่นใจของแต่ละกฎใน Class นั้นๆ

สูตรที่ 4 Max

$$\sum_{i=1}^{c}\left(\left(\left(\sum_{j=1}^{n}\left(Confidence(r_{ij}) \times Support(r_{ij})\right)\right) \div \sum_{j=1}^{n}(Support(r_{ij})\right) \times Weight_i\right) \div \left(\sum_{i=1}^{c} Weight_i\right)$$

เมื่อ r_{ij} คือสมาชิกแต่ละตัวใน Class นั้นๆ

n คือ จำนวนสมาชิกทั้งหมดของ Class นั้นๆ

j คือ ค่าความยาวของ Attribute หรือความยาวของกฎ แต่ละระดับ

c คือระดับความยาวของ Attribute ทั้งหมด

$Weight_i$ คือค่าถ่วงน้ำหนักของกฎในแต่ละระดับความยาวของกฎ

โดยตัวอย่างการคำนวณของแต่ละสูตรจะแสดงในตารางต่อไป โดยตารางที่ 23 จะเป็นตัวอย่างกฎความสัมพันธ์แบบมีคลาสที่จะนำมาใช้ในการคำนวณในสูตรที่ 1- 3 และตารางที่27 จะเป็นตัวอย่างของกฎความสัมพันธ์แบบมีคลาสที่จะนำมาใช้ในการคำนวณกับสูตรที่ 4

ตารางที่ 23 ตัวอย่างกฎความสัมพันธ์แบบมีคลาสที่นำไปใช้ในการคำนวณในสูตรที่ 1-3

Rid	Rule	Support%	Confidence%
1	A, B, C => X	30	60
2	A, B, D => X	20	60
3	B, D, E => X	25	50
4	B, D, F => Y	40	80
5	B, E, F => Y	15	60
6	A, E, F => Z	25	95

ตารางที่ 24 ผลที่ได้จากการคำนวณโดยใช้สูตรที่ 1

คลาส X	คลาส Y	คลาส Z
1+1+1 = 3	1+1 = 2	= 1

จากตารางที่ 24 ซึ่งใช้สูตรที่ 1 ในการคำนวณ จะเห็นว่าคลาส X มีสมาชิกของกฎความสัมพันธ์แบบมีคลาสอยู่มากที่สุด เพราะฉะนั้น คลาสที่จะทำการทำนาย คือคลาส X

ตารางที่ 25 ผลที่ได้จากการคำนวณโดยใช้สูตรที่ 2

คลาส X	คลาส Y	คลาส Z
(30 + 20 + 25) ÷ 3 = 25%	(40 + 15) ÷ 2 = 27.5%	= 25%

จากตารางที่ 25 ซึ่งใช้สูตรที่ 2 ในการคำนวณ จะเห็นว่าคลาส Y มีค่าสนับสนุนเฉลี่ยของกฎความสัมพันธ์แบบมีคลาสอยู่มากที่สุด เพราะฉะนั้น คลาสที่จะทำการทำนาย คือคลาส Y

ตารางที่ 26 ผลที่ได้จากการคำนวณโดยใช้สูตรที่ 3

คลาส X	คลาส Y	คลาส Z
(60 + 60 + 50) ÷ 3 = 56.67%	(80 + 60) ÷ 2 = 70%	= 95%

จากตารางที่ 26 ซึ่งใช้สูตรที่ 3 ในการคำนวณ จะเห็นว่าคลาส Z มีค่าความมั่นใจเฉลี่ยของกฎความสัมพันธ์แบบมีคลาสอยู่มากที่สุด เพราะฉะนั้น คลาสที่จะทำการทำนาย คือคลาส Z

ตารางที่ 27 ตัวอย่างกฎความสัมพันธ์แบบมีคลาสที่นำไปใช้ในการคำนวณในสูตรที่ 4

Rid	Rule	Support%	Confidence%
1	A, B, C => X	30	60
2	A, B, D => X	20	60
3	B, D, E => X	25	50
4	B, D, F => Y	40	80
5	B, E, F => Y	15	60
6	A, E, F => Z	25	70
7	A, B => X	40	40
8	B, D => Y	30	35
9	B, E => Y	45	45
10	E, F => Z	35	90
11	A => X	60	30
12	A => Z	60	30

ตารางที่ 27 เป็นตัวอย่างข้อมูลที่จะนำมาใช้คำนวณในสูตรที่4 เนื่องจากสูตรที่4 จะต้องใช้หน่วยวัดทั้งค่าสนับสนุน (Support) และค่าความมั่นใจ (Confidence) ด้วย

<u>ตารางที่ 28</u> ผลของการคำนวณในทุกระดับ โดยใช้สูตรที่ 4

ระดับของกฎ \ คลาส	คลาส X	คลาส Y	คลาส Z
3	(0.6 × 0.3) + (0.6 ×0.2) + (0.5 × 0.25) ÷ 0.75 = 56.7%	(0.8 × 0.4) + (0.6 × 0.15) ÷ 0.55 = 74.5%	(0.7 × 0.25) ÷ 0.25 = 70%
2	(0.4 × 0.4) ÷ 0.4 = 40%	(0.35 × 0.3) + (0.45 × 0.45) ÷ 0.75 = 41%	(0.9 × 0.35) ÷ 0.35 = 90%
1	(0.3 × 0.6) ÷ 0.6 = 30%	-	(0.3 × 0.6) ÷ 0.6 = 30%

<u>ตารางที่ 29</u> ผลรวมของทุกระดับที่ได้จากการคำนวณ โดยใช้สูตรที่ 4

คำนวณผลรวมทั้งหมดในทุกระดับของกฎความสัมพันธ์		
คลาส X	คลาส Y	คลาส Z
(56.7 × 3)+ (40 × 2)+ (30 × 1) ÷ 6 = 46.68	(74.5 × 3) + (41 × 2) ÷ 6 = 50.92	(70 × 3) +(90 × 2) +(30 × 1) ÷ 6 = 70

จากตารางที่ 28 ซึ่งเป็นการคำนวณเบื้องต้นโดยใช้สูตรที่ 4 ในการคำนวณ และหลังจากนั้นจึงทำการคำนวณผลรวมในทุกระดับของแต่ละคลาสออกมาดังตารางที่ 29 จากผลที่ได้จะเห็นว่าคลาส Z มีค่าผมรวมของค่าถ่วงน้ำหนักระหว่างค่าความมั่นใจกับค่าสนับสนุนของกฎความสัมพันธ์แบบมีคลาสอยู่มากที่สุด เพราะฉะนั้น คลาสที่จะทำการทำนาย คือคลาส Z

ในการทำนายข้อมูลตามสูตรทั้ง 4 สูตร จะคำนวณค่าจากกลุ่มของกฎแต่ละคลาสโดยดูว่ากลุ่มของกฎคลาสไหนที่ให้ค่ามากที่สุด ก็จะทำนายคลาสของกลุ่มนั้นไป โดยรายละเอียดของอัลกอริทึมในส่วนของการทำนายข้อมูล อยู่ในภาพที่ 7

```
R_All = ECARs;
R_sat = select (Sat_dataobj);
R_sat = sort(R_sat);
for each level (i = max; i > 0; i--)  /* start from maximal frequent itemsets */
    for each class (j = 0; j < class_number; j++)
        total_j = (Calculate_multiple_rules);  /* use one of the four equations */
    Class_result = Find_max (total);
    if (Class_result.size() = 1)
        Predict = Class_result;
        break;
    end
end
```

<u>ภาพที่ 7</u> อัลกอริทึม CBEAR ในส่วนของการทำนายข้อมูล

จากรูปจะเริ่มจาก ได้เซตของกฎความสัมพันธ์แบบมีคลาสที่เรียกว่า ECARs (Essential Class-Association Rules) มาทั้งหมด ซึ่งหลังจากนี้ระบบก็พร้อมที่จะทำนายข้อมูลแล้ว โดยเมื่อมีข้อมูลทดสอบระบบเข้ามา ระบบจะทำการเลือกเฉพาะกฎความสัมพันธ์แบบมีคลาสที่สอดคล้องกับข้อมูลนั้น จากนั้นก็นำกฎเหล่านั้นมาทำการจัดเรียงให้อยู่ในรูปแบบที่ได้กำหนดไว้ หลังจากนั้นก็จะพิจารณาเฉพาะกฎที่มีความยาวของ Itemsets มากที่สุดก่อน โดยการนำกฎเหล่านั้นไปคำนวณ ซึ่งจะคำนวณเป็นกลุ่มตามคลาสปลายทาง และหลังจากคำนวณเสร็จแล้ว ก็จะนำค่าที่ได้ในแต่ละกลุ่มมาเปรียบเทียบเพื่อหาค่าที่มากที่สุด ถ้ามีเพียงกลุ่มเดียวที่มีค่ามากที่สุด ระบบก็จะทำนายคลาสปลายทางของกลุ่มนั้นออกมา แต่ถ้ามีหลายๆกลุ่มที่มีค่ามากที่สุดเท่ากัน ระบบก็จะทำการพิจารณากลุ่มของกฎที่มีความยาวรองลงมา โดยจะนำกฎเหล่านั้นไปคำนวณตามวิธีที่ได้กล่าวมา เพื่อที่จะทำนายข้อมูล โดยระบบจะทำจนกว่าที่จะมีเพียงกลุ่มเดียวที่มีค่ามากที่สุด จากนั้นระบบก็จะทำนายคลาสปลายทางตามกลุ่มของกฎความสัมพันธ์แบบมีคลาสที่มีค่ามากที่สุด

การเปรียบเทียบคุณลักษณะของแต่ละอัลกอริทึม

เนื่องจากความหลากหลายและความแตกต่างกันของคุณลักษณะในแต่ละอัลกอริทึมที่มีอยู่ ดังนั้นในหัวข้อนี้จึงได้นำเสนอตารางเปรียบเทียบความแตกต่างของคุณลักษณะหลักๆที่มีอยู่ในแต่ละอัลกอริทึม จากตารางที่ 30

ตารางที่ 30 ตารางเปรียบเทียบคุณลักษณะของแต่ละอัลกอริทึม

Algorithms	Rule generator phase		Classifier builder phase	
	Rule pruning	Rule compression	Rule sorting	Prediction method
CBA	✓	✗	✓	Single rule
Improved CBA	✓	✗	✓	Single rule
CMAR	✓	✗	✗	Multiple rule
CPAR	✓	✗	✗	Multiple rule
CBEAR	✓	✓	✓	Multiple rule

จากตารางที่30 ซึ่งเปรียบเทียบให้เห็นถึงคุณลักษณะที่แตกต่างกันของแต่ละอัลกอริทึม โดยมีคุณลักษณะที่สำคัญๆ นั่นคือในส่วนของการสร้างโมเดลในการทำนาย (Rule generator phase) จะประกอบด้วย การกำจัดกฎความสัมพันธ์ที่ไม่มีประโยชน์ (Rule pruning) และ การบีบอัดหรือการกำจัดกฎความสัมพันธ์แบบมีคลาสที่ซ้ำซ้อน (Rule compression)

การกำจัดกฎความสัมพันธ์ที่ไม่มีประโยชน์ (Rule pruning) ถือเป็นวิธีการในการกำจัดกฎความสัมพันธ์ขั้นพื้นฐาน ซึ่งทุกๆ อัลกอริทึมควรจะต้องมี โดยในการกำจัดกฎความสัมพันธ์ดังกล่าวจะมีการกำหนดค่าสนับสนุนขั้นต่ำ เพื่อใช้กำจัดไอเท็มเซตที่ไม่มีความจำเป็นหรือไอเท็มเซตที่มีค่าสนับสนุนน้อยกว่าค่าสนับสนุนขั้นต่ำ และมีการกำหนดค่าความมั่นใจขั้นต่ำ เพื่อกำจัดกฎ

ความสัมพันธ์แบบมีคลาสที่มีค่าความมั่นใจของกฎต่ำกว่าค่าความมั่นใจขั้นต่ำ โดยในการกำหนดค่าขั้นต่ำของหน่วยวัดนั้นๆ จะกระทำโดยผู้เชี่ยวชาญระบบ (Expert user) หรือจะกำหนดตามค่ามาตรฐานของแต่ละระบบก็ได้ โดยในเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ จะกำหนดค่าสนับสนุนขั้นต่ำเท่ากับ 1 เปอร์เซ็นต์ เพื่อป้องกันไม่ให้ไอเท็มเซตที่มีความสำคัญถูกกำจัดทิ้งไป เพราะโมเดลในการทำนายจะมีความแม่นยำมากน้อยเพียงใด ขึ้นอยู่กับกฎความสัมพันธ์แบบมีคลาสที่นำไปใช้ในการสร้างโมเดลในการทำนายด้วย

การบีบอัดหรือการกำจัดกฎความสัมพันธ์แบบมีคลาสที่ซ้ำซ้อน (Rule compression) เป็นอีกคุณลักษณะหนึ่งที่สำคัญในส่วนของการสร้างกฎความสัมพันธ์ ซึ่งอัลกอริทึม CBEAR ได้นำเสนอ โดยในอัลกอริทึมต่างๆ ก่อนหน้านี้ไม่ได้ใช้วิธีดังกล่าว โดยอัลกอริทึม CBEAR ใช้เทคนิคการบีบอัดกฎความสัมพันธ์ในการกำจัดกฎความสัมพันธ์แบบมีคลาสที่ซ้ำซ้อนออกไปทั้งหมด ซึ่งผลที่ได้ก็คือจะทำให้เหลือแต่กฎที่มีความสำคัญจริงๆและไม่ซ้ำซ้อนกับกฎอื่นๆ ซึ่งจะส่งผลให้การนำกฎเหล่านั้นไปใช้ในการสร้างโมเดลเพื่อทำนายมีข้อผิดพลาดลดลง และเป็นการเพิ่มความแม่นยำให้กับอัลกอริทึมที่ได้นำเสนออีกด้วย

ในส่วนของการสร้างโมเดลในการทำนายข้อมูล (Classifier builder phase) จะประกอบไปด้วยคุณลักษณะที่สำคัญนั่นคือ การจัดเรียงกฎความสัมพันธ์ (Rule sorting) และวิธีการในการทำนายข้อมูล (Prediction method)

ในส่วนของวิธีการในการทำนายข้อมูล (Prediction method) อัลกอริทึมจะมีอยู่สองวิธี วิธีแรก คือการทำนายข้อมูลโดยใช้กฎความสัมพันธ์เพียงแบบมีคลาสกฎเดียวในการทำนาย (Single rule) และวิธีที่สอง คือการทำนายข้อมูลโดยใช้กลุ่มของกฎความสัมพันธ์แบบมีคลาสในการทำนายข้อมูล (Multiple rules) ซึ่งอัลกอริทึม CBEAR ใช้วิธีการทำนายแบบที่สอง เนื่องจากงานวิจัยของ (Wenmin, 2001; Wenmin et al., 2001) ได้พิสูจน์จากผลการทดลองแล้วว่าให้ความแม่นยำในการทำนายมากกว่าวิธีแรก ส่วนคุณลักษณะถัดมา คือการจัดเรียงกฎความสัมพันธ์แบบมีคลาส (Rule sorting) ซึ่งมีความสัมพันธ์กันกับวิธีการในการทำนายข้อมูล โดยอัลกอริทึมที่ใช้การทำนายแบบวิธีที่สอง ซึ่งเสนอก่อนหน้านี้นั้นจะไม่มีการจัดเรียงกฎ แต่อัลกอริทึม CBEAR ได้นำเสนอวิธีการจัดเรียงกฎแบบใหม่ (Hanchodchung et al., 2006) โดยให้ความสำคัญในการพิจารณากฎในระดับที่มีความยาวมากที่สุดก่อน เนื่องจากเหตุผลที่ว่ากฎเหล่านั้นน่าจะมีความใกล้เคียงกับข้อมูลคำถามมากที่สุด ดังนั้นจึงนำกฎเหล่านั้นไปใช้ในการทำนายข้อมูล

ผลและวิจารณ์

ผล

วิธีวัดผลการทดลอง

ในการวัดผลการทดลองนี้ จะใช้หน่วยวัด 2 หน่วยคือ วัดค่าความแม่นยำ (Accuracy) และวัดค่าการบีบอัดหรืออัตราร้อยละการลดจำนวนกฎความสัมพันธ์ (Compression factor) โดยสูตรในการคำนวณของแต่ละหน่วยวัด เป็นดังต่อไปนี้

สูตรการหาเปอร์เซนต์ค่าความแม่นยำ คือ

(จำนวนข้อมูลที่ทำนายถูก ÷ จำนวนข้อมูลทั้งหมด) × 100

สูตรการหาเปอร์เซนต์การบีบอัดหรืออัตราร้อยละการลดจำนวนกฎความสัมพันธ์ (Compression factor) คือ

(จำนวนกฎความสัมพันธ์ที่ได้ ÷ จำนวนกฎความสัมพันธ์ทั้งหมด) × 100

ตัวอย่างเช่น ถ้ากฎความสัมพันธ์ที่ได้กระบวนการสร้างกฎความสัมพันธ์มีขนาด 50,000 กฎ แต่เราสามารถสร้างกฎความสัมพันธ์ที่มีขนาดเล็กลงได้ เหลือแค่ 15,000 กฎ ดังนั้นค่าการบีบอัดหรืออัตราร้อยละการลดจำนวนกฎความสัมพันธ์จะได้มาจาก (15,000 ÷ 50,000) × 100 ซึ่งจะได้ค่าเท่ากับ 30% โดยค่าที่ได้จากสูตร ถ้าตัวเลขที่ได้ เข้าใกล้หนึ่ง จะหมายถึง สามารถที่จะลดจำนวนกฎลงได้ แต่ไม่มากนัก แต่ถ้าตัวเลขที่ได้ เข้าใกล้ศูนย์ หมายความว่า สามารถที่จะลดจำนวนกฎลงไปได้มาก

ผลการทดลอง

ตารางที่ 31 รายละเอียดฐานข้อมูล UCI

Dataset	#Attribute	#Transactions	#Classes	Characteristic
Breast	10	699	2	Dense
Cleve	13	303	2	Dense
Diabetes	8	768	2	Sparse
Heart	13	270	2	Dense
Iris	4	150	3	Sparse
Led7	7	3,200	10	Sparse
Pima	8	768	2	Sparse

จากตารางที่31 แสดงให้เห็นถึงรายละเอียดของทั้ง 7 ฐานข้อมูลที่นำมาใช้โดยบอกรายละเอียดดังต่อไปนี้

#Attribute คือ จำนวนของข้อมูลในแต่ละแถว
#Transactions คือ จำนวนแถวของฐานข้อมูล
#Classes คือ จำนวนคลาสปลายทาง ที่ต้องการจะทำนาย
Characteristic คือ ลักษณะของข้อมูล

ลักษณะของข้อมูล จะแบ่งออกเป็น 2 ชนิดคือ ข้อมูลแบบหนาแน่น (Dense dataset) และข้อมูลแบบกระจาย (Sparse dataset) ซึ่งข้อมูลแบบหนาแน่นจะมีลักษณะคือ จะมี Transaction ที่มีความคล้ายคลึงกันอยู่มากๆ ในฐานข้อมูล และข้อมูลแบบกระจาย (Sparse dataset) จะมีลักษณะตรงกันข้ามคือ จำนวน Transaction ที่มีความคล้ายคลึงกันจะมีอยู่ไม่มากนั่นเอง โดยที่ข้อมูลแบบหนาแน่นั้นยังสามารถที่จะเกิดขึ้นในฐานข้อมูลแบบกระจายได้ด้วย นั่นคือมี Transaction ที่คล้ายคลึงกันเกิดขึ้นเป็นกลุ่มๆ ในฐานข้อมูลแบบกระจาย ซึ่งฐานข้อมูลที่ได้นำมาใช้ก็มีลักษณะที่เป็นข้อมูลที่หนาแน่น (Dense) ที่เกิดขึ้นในฐานข้อมูลแบบกระจายนั่นเอง

ในส่วนของการทดลอง จะต้องมีการกำหนดเกณฑ์ขั้นต่ำของหน่วยวัด เพื่อใช้แบ่งแยกข้อมูลที่ไม่มีความจำเป็นออกไป เพื่อจะพิจารณาเฉพาะข้อมูลที่ผ่านเกณฑ์ขั้นต่ำเท่านั้น โดยมีการกำหนดค่าสนับสนุนขั้นต่ำ (Minimum support) เท่ากับ 1 เปอร์เซ็นต์ และกำหนดค่าความมั่นใจขั้นต่ำ (Minimum confidence) เท่ากับ 50 เปอร์เซ็นต์

การวัดขนาดของกฎความสัมพันธ์ โดยใช้ค่า Compression Factor

ตารางที่ 32 แสดงค่า Compression Factor (CF%) ในแต่ละฐานข้อมูล

Datasets	minsup = 1%		
	Rules	ECARs	CF%
Breast	27,997	9,010	32.18
Cleve	66,619	10,353	15.54
Diabetes	1,687	864	51.22
Heart	24,427	3,956	16.20
Iris	128	79	61.72
Led7	1,300	1,144	88.00
Pima	1,858	999	53.77
Average			45.52%

จากตารางที่ 32 แสดงให้เห็นถึง อัตราร้อยละของการบีบอัดหรืออัตราการลดจำนวนกฎความสัมพันธ์แบบมีคลาส (Compression factor)

โดยที่

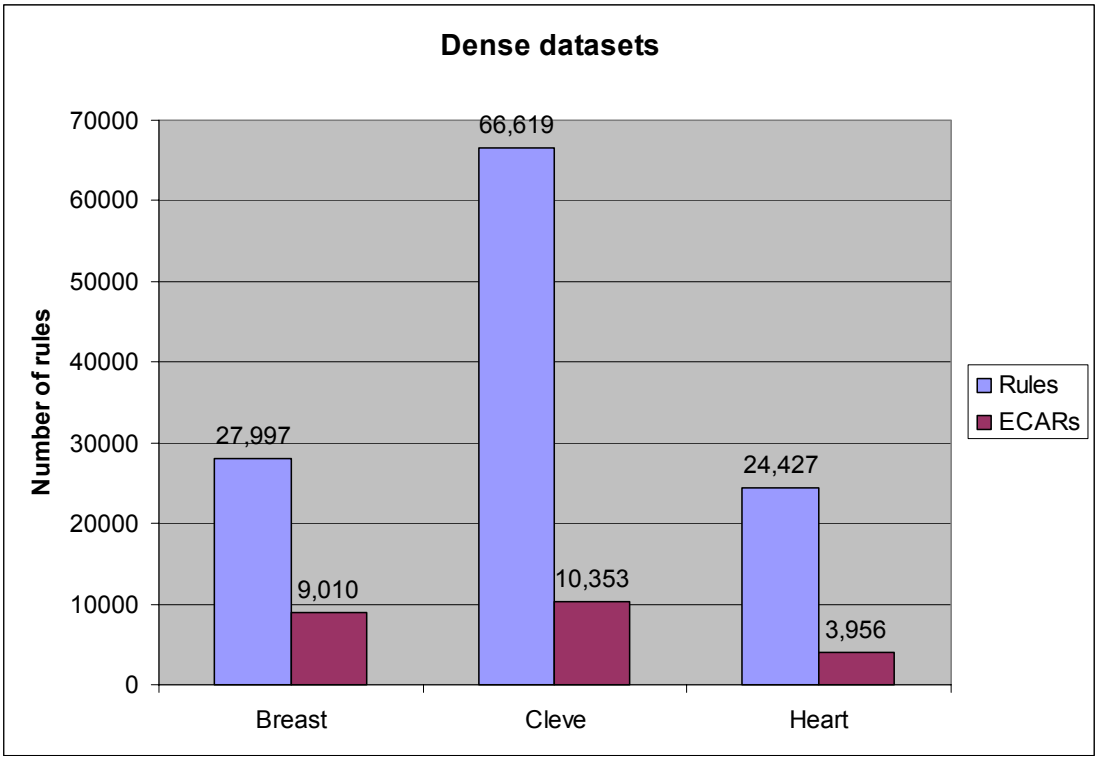
Rules คือ กฎความสัมพันธ์แบบมีคลาสก่อนลดจำนวน (Complete class-association rules)

ECARs คือ กฎความสัมพันธ์แบบมีคลาสหลังลดจำนวนลงแล้ว (Essential class-association rules)

CF% คือ ค่า Compression factor หรือการบีบอัดกฎความสัมพันธ์แบบมีคลาส

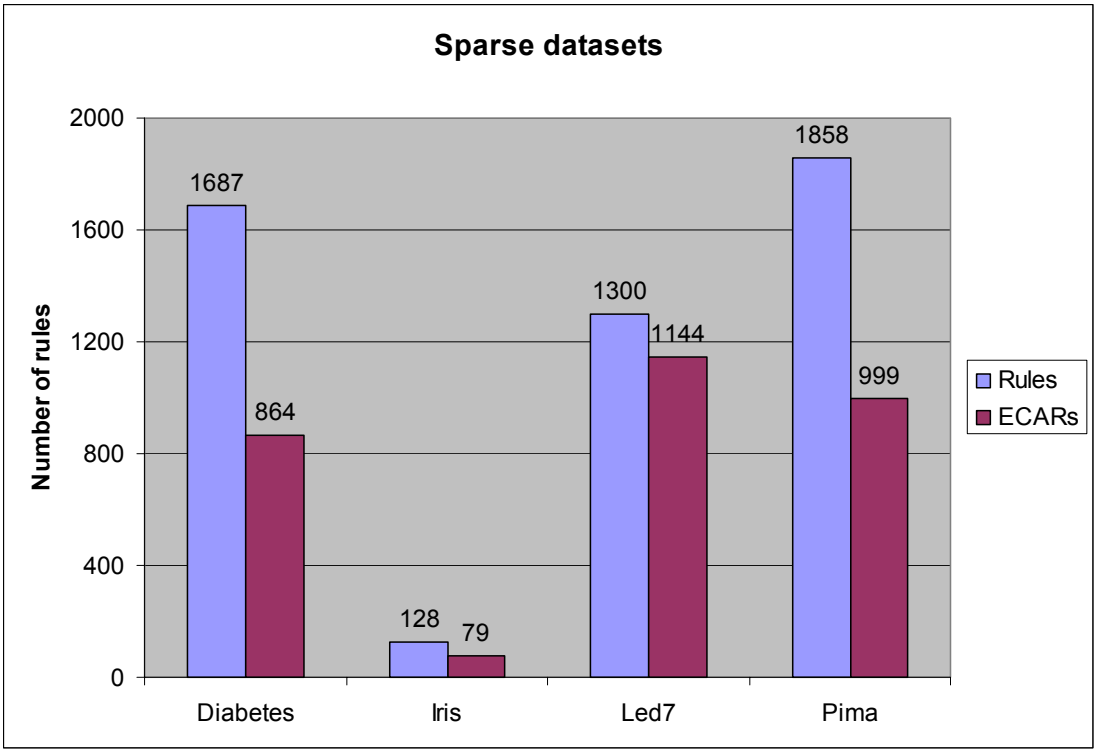
จากผลการทดลองที่ได้จะเห็นว่า วิธีการที่ได้นำเสนอในงานวิจัยเล่มนี้สามารถที่จะลดจำนวนกฎความสัมพันธ์แบบมีคลาสลงไปได้เกินกว่าครึ่งหนึ่ง ซึ่งโดยเฉลี่ยแล้วสามารถลดจำนวนกฎลงได้ประมาณ 54.48เปอร์เซ็นต์ หรือเหลือกฎความสัมพันธ์ประมาณ 45.52 เปอร์เซ็นต์

โดยในส่วนของการเปรียบเทียบจำนวนกฎความสัมพันธ์แบบมีคลาสที่ได้ก่อน และหลังการกำจัดกฎที่ซ้ำซ้อนออกไป ในฐานข้อมูลแบบหนาแน่น และฐานข้อมูลแบบกระจาย จะแสดงในภาพที่ 8 และ 9 ตามลำดับ



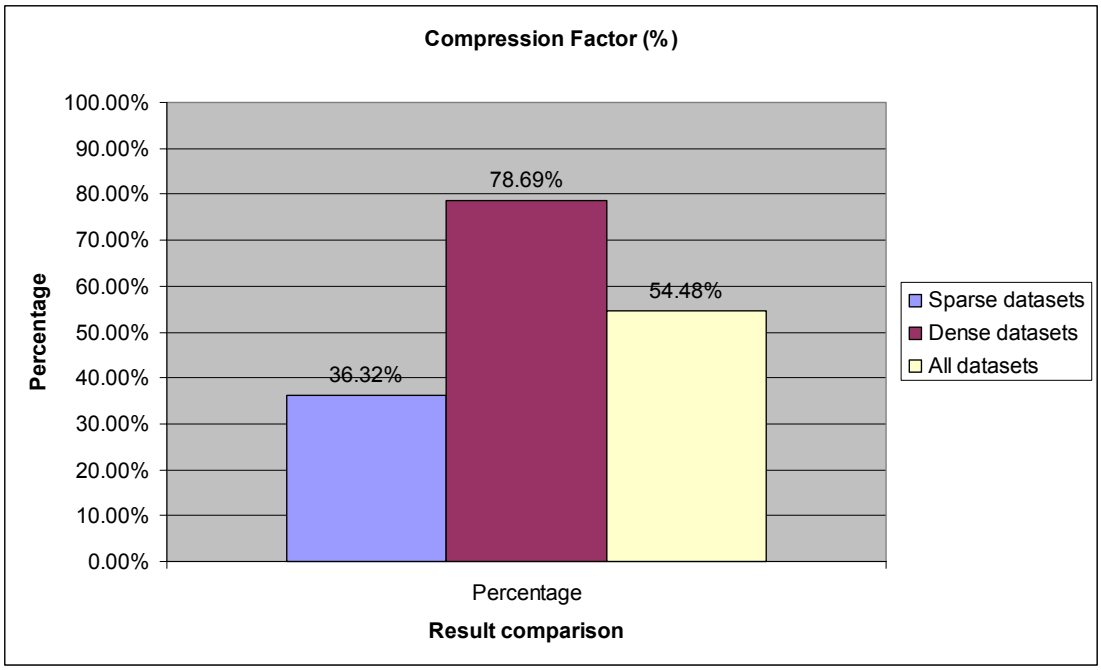
ภาพที่ 8 เปรียบเทียบจำนวนกฎความสัมพันธ์แบบมีคลาสที่ได้ในฐานข้อมูลแบบหนาแน่น

จากภาพที่ 8 แสดงให้เห็นถึงความสามารถในการลดจำนวนกฎความสัมพันธ์แบบมีคลาสในฐานข้อมูลแบบหนาแน่น ซึ่งจากการเปรียบเทียบจะเห็นได้ว่าสามารถลดจำนวนกฎลงไปได้มากหลายเท่าตัว



ภาพที่ 9 เปรียบเทียบจำนวนกฎความสัมพันธ์แบบมีคลาสที่ได้ในฐานข้อมูลแบบกระจาย

จากภาพที่ 9 แสดงให้เห็นถึงความสามารถในการลดจำนวนกฎความสัมพันธ์แบบมีคลาสในฐานข้อมูลแบบกระจาย ซึ่งจากการเปรียบเทียบจะเห็นได้ว่าสามารถลดจำนวนกฎลงได้ประมาณ 1 ใน 3 ของกฎความสัมพันธ์ที่มี



<u>ภาพที่ 10</u> เปรียบเทียบค่า Compression Factor (CF%) ของฐานข้อมูลแต่ละแบบ

จากรูปที่ 10 แสดงให้เห็นถึง อัตราร้อยละของการบีบอัดหรืออัตราการลดจำนวนกฎความสัมพันธ์แบบมีคลาส (Compression factor) ในฐานข้อมูลแต่ละแบบ ซึ่งจะเห็นว่าในฐานข้อมูลแบบกระจาย สามารถที่จะลดจำนวนกฎความสัมพันธ์แบบมีคลาสลงได้มากโดยเฉลี่ยประมาณ 36.32 เปอร์เซ็นต์ แต่ก็ไม่มากเท่ากับในฐานข้อมูลแบบหนาแน่นที่สามารถลดจำนวนกฎความสัมพันธ์แบบมีคลาสลงไปโดยเฉลี่ยได้มากถึง 78.69 เปอร์เซ็นต์ และโดยสรุปแล้วสามารถลดจำนวนกฎความสัมพันธ์แบบมีคลาสกับฐานข้อมูลทั้งหมดโดยเฉลี่ยประมาณ 54.48 เปอร์เซ็นต์ ซึ่งจะเห็นว่าสามารถลดจำนวนกฎลงไปได้มากกว่าขึ้นเลยทีเดียว

จากการทดลองพบว่า วิธีการกำจัดกฎความสัมพันธ์แบบมีคลาสที่ซ้ำซ้อนเมื่อใช้กับฐานข้อมูลแบบหนาแน่น จะให้ผลการบีบอัดหรืออัตราการลดจำนวนกฎความสัมพันธ์แบบมีคลาส (Compression factor) ได้มากกว่าการใช้กับฐานข้อมูลแบบกระจาย

การวัดความแม่นยำในการทำนาย (Accuracy)

ตารางที่ 33 แสดงค่าความแม่นยำของแต่ละอัลกอริทึม

ฐานข้อมูล	อัลกอริทึม						
	C4.5	CBA	CMAR	CBEAR กับสูตรที่ 1	CBEAR กับสูตรที่ 2	CBEAR กับสูตรที่ 3	CBEAR กับสูตรที่ 4
Breast	95	96.3	96.4	98	97.8	98.1	98.4
Cleve	78.2	82.8	82.2	83.5	81.8	83.5	85.8
Diabetes	74.2	74.5	75.8	72.7	72.7	72.8	77
Heart	80.8	81.9	82.2	84.8	82.2	84.4	82.6
Iris	95.3	94.7	94	96.7	96	96.7	96
Led7	73.5	71.9	72.5	74.9	74.9	75	74.7
Pima	75.5	72.9	75.1	75.5	75.4	76.7	76.8
Average	81.8	82.1	82.6	83.7	83.0	83.9	84.5

โดยที่

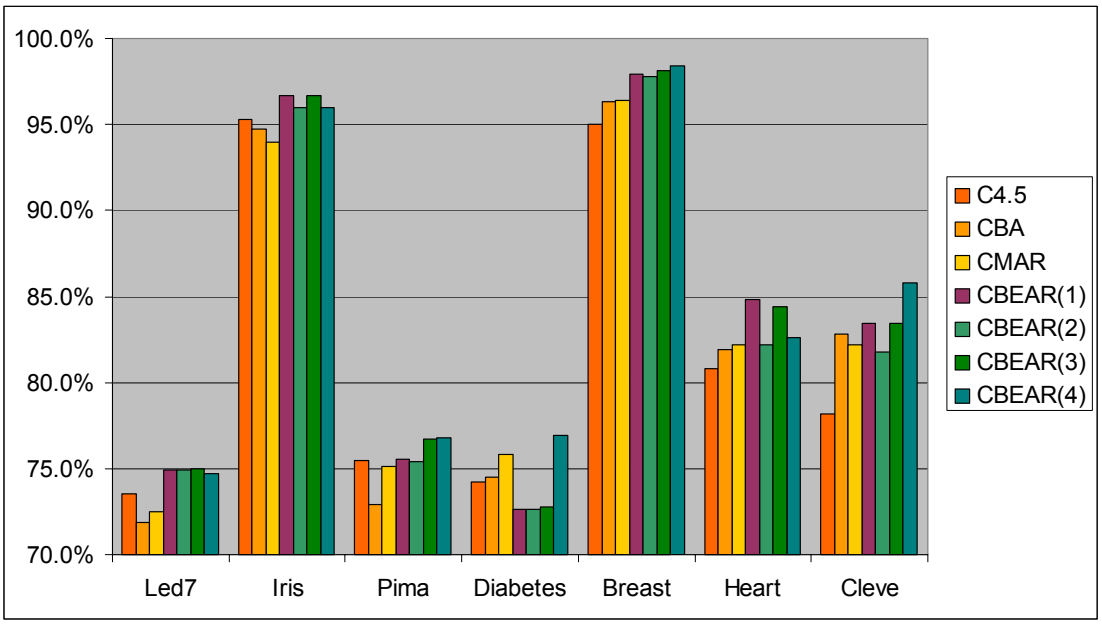
สูตรที่ 1 คือ การหาจำนวนสมาชิกของกฎในแต่ละกลุ่ม

สูตรที่ 2 คือ การหาค่าเฉลี่ยของค่าสนับสนุนในแต่ละกลุ่ม

สูตรที่ 3 คือ การหาค่าเฉลี่ยของค่าความมั่นใจในแต่ละกลุ่ม

สูตรที่ 4 คือ การหาค่าถ่วงน้ำหนักของความมั่นใจกับค่าสนับสนุนของแต่ละกลุ่ม

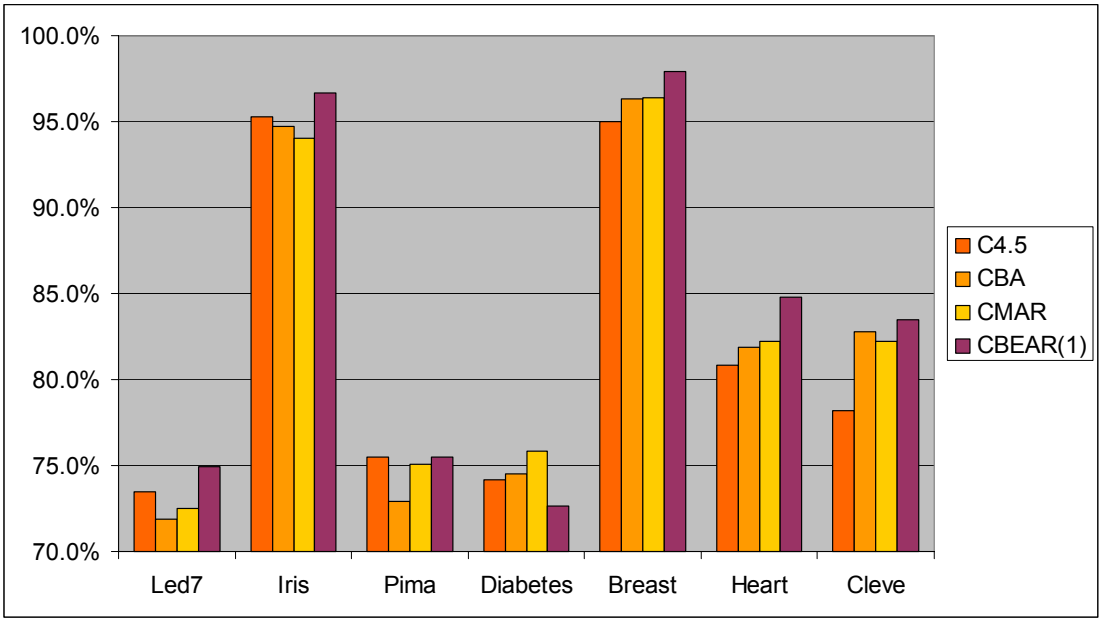
ตารางที่33 แสดงให้เห็นถึงค่าความแม่นยำของอัลกอริทึม CBEAR ที่ได้นำเสนอเปรียบเทียบกับอัลกอริทึมอื่นๆที่เป็นที่นิยมของเทคนิคการจำแนกการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ (Associative Classification) โดยจะเห็นว่าอัลกอริทึม CBEAR มีความแม่นยำมากกว่า อัลกอริทึมที่มีอยู่ในปัจจุบัน (Janssens et al., 2003; Liu et al., 1998; Liu et al., 2001; Yin and Han, 2003)



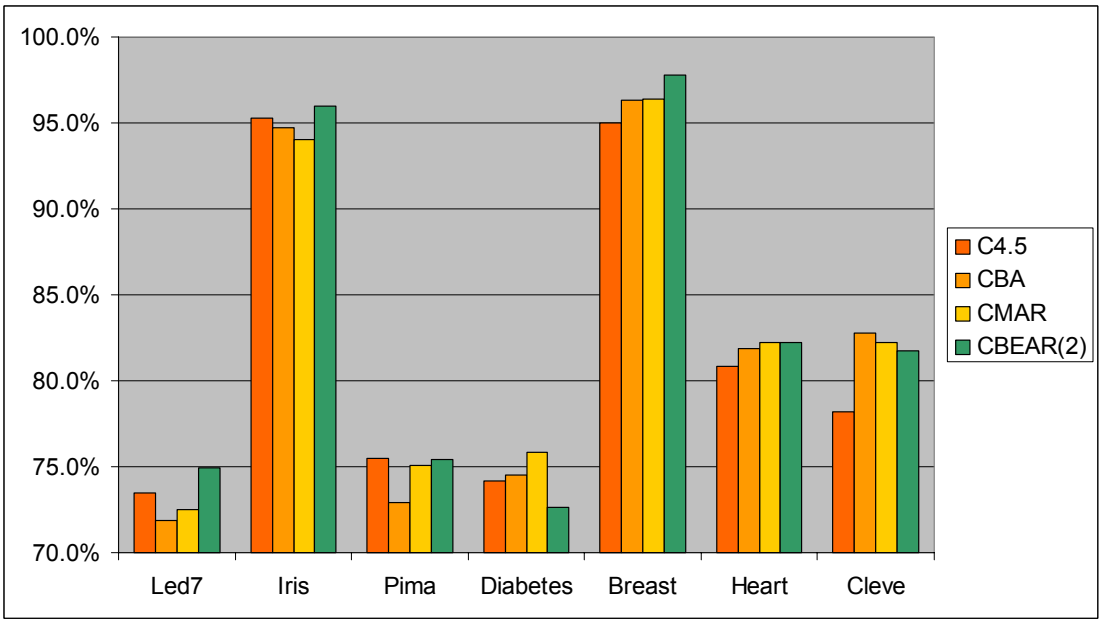
ภาพที่ 11 เปรียบเทียบความแม่นยำของแต่ละอัลกอริทึม

เพื่อความเข้าใจได้ง่ายในการจัดลำดับค่าความแม่นยำ ในภาพที่ 11 จึงแสดงเป็นกราฟให้เห็นถึงค่าความแม่นยำของอัลกอริทึม CBEAR ทั้ง 4 สูตร เปรียบเทียบกับอัลกอริทึมตัวอื่นๆ

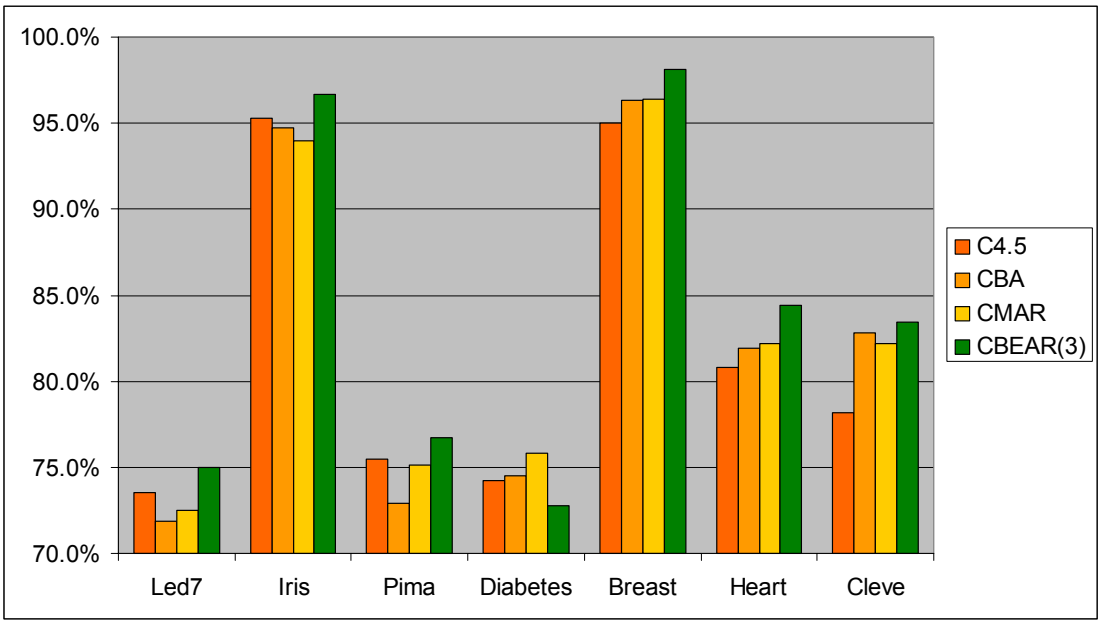
เพื่อความชัดเจนของผลการทดลองยิ่งขึ้น จึงแสดงค่าความแม่นยำของอัลกอริทึม CBEAR ในแต่ละสูตร เพื่อทำการเปรียบเทียบกับอัลกอริทึมตัวอื่นๆ โดยจะแสดงในภาพที่ 12 ถึงภาพที่ 16 ตามลำดับ



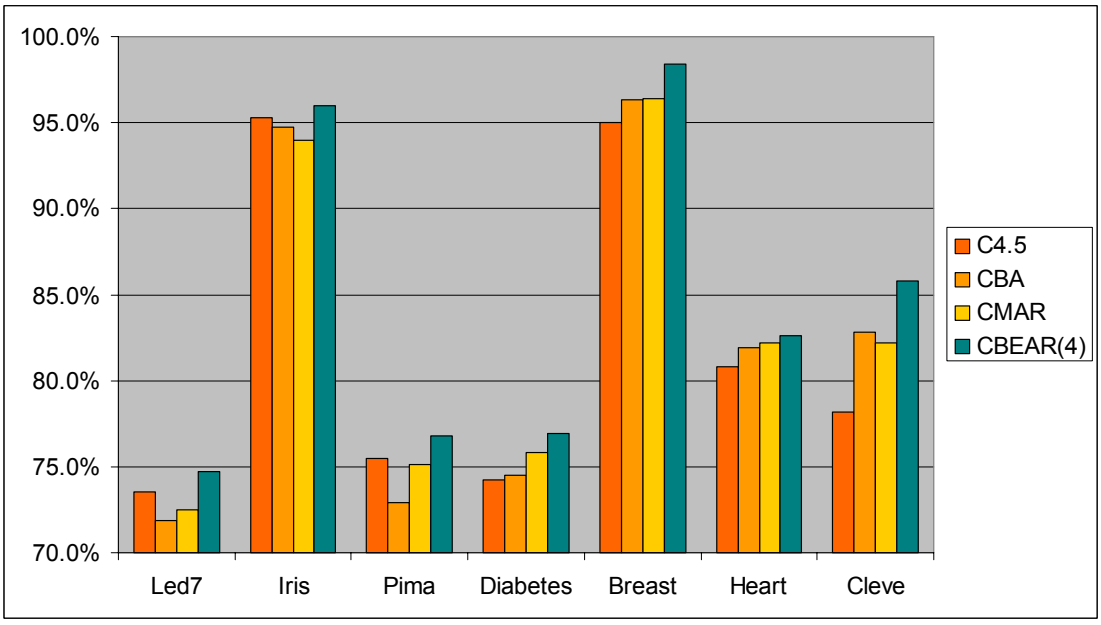
ภาพที่ 12 เปรียบเทียบความแม่นยำของอัลกอริทึม CBEAR(1) กับอัลกอริทึมอื่นๆ



ภาพที่ 13 เปรียบเทียบความแม่นยำของอัลกอริทึม CBEAR(2) กับอัลกอริทึมอื่นๆ



ภาพที่ 14 เปรียบเทียบความแม่นยำของอัลกอริทึม CBEAR(3) กับอัลกอริทึมอื่นๆ

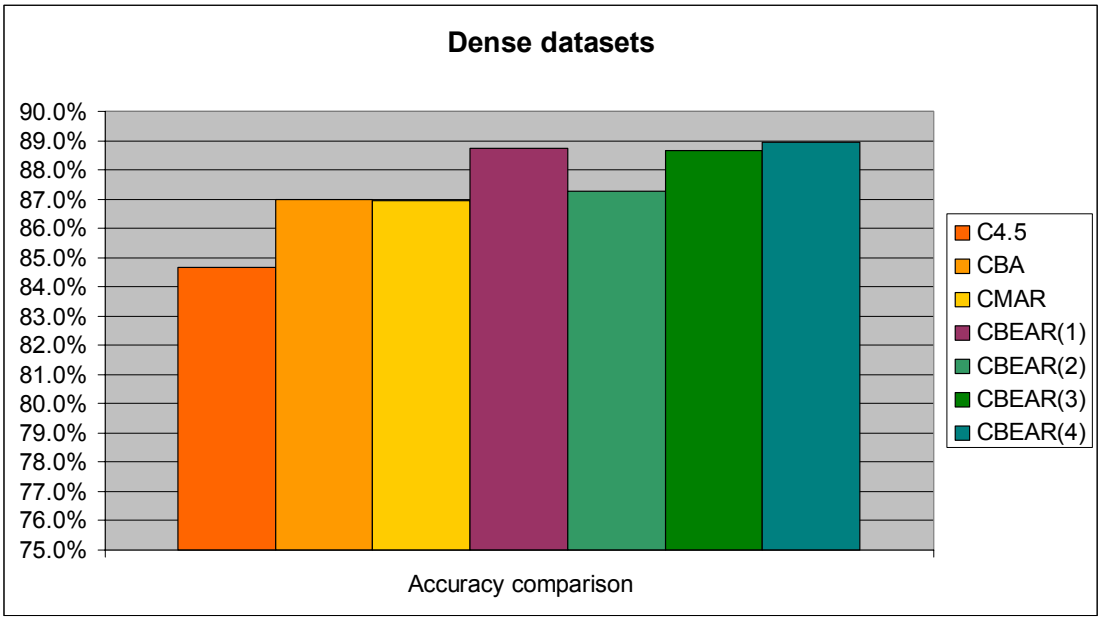


ภาพที่ 15 เปรียบเทียบความแม่นยำของอัลกอริทึม CBEAR(4) กับอัลกอริทึมอื่นๆ

จากผลการทดลองเปรียบเทียบความแม่นยำของอัลกอริทึม CBEAR ทั้ง 4 สูตรกับพบว่า CBEAR(1), CBEAR(2), CBEAR(3) ยังแพ้อัลกอริทึมอื่นๆ ในฐานข้อมูล Diabetes เนื่องจากทั้ง 3

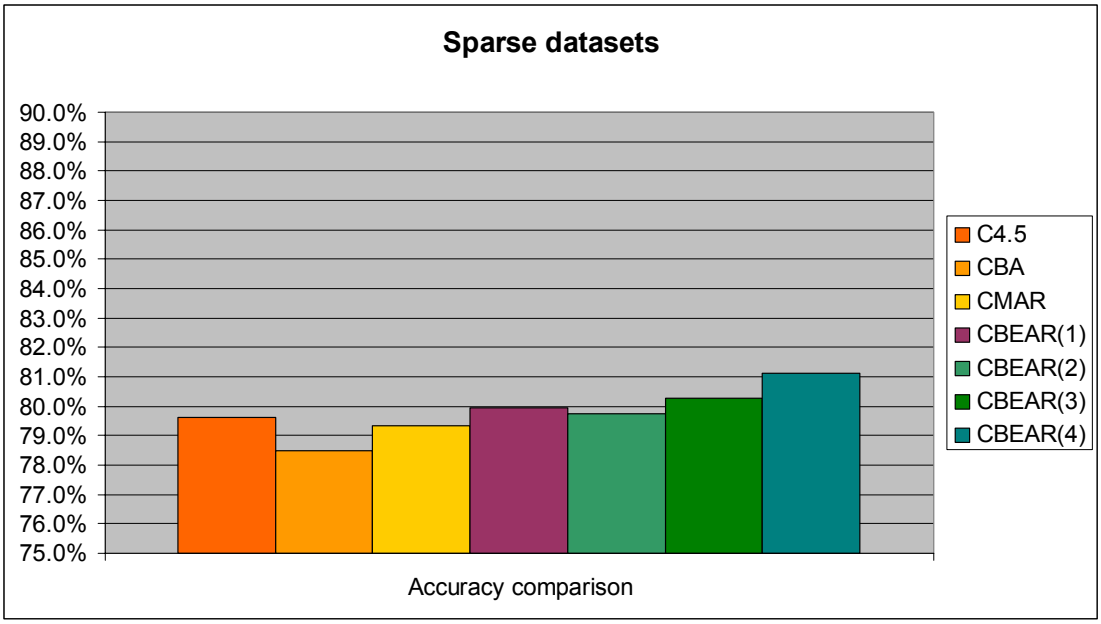
สูตรดังกล่าวได้ใช้หน่วยวัดเดียวในการคำนวณ ซึ่งอาจจะมีข้อผิดพลาดเกิดขึ้นได้ แต่ในอัลกอริทึม CBEAR(4) นั้นมีการใช้หน่วยวัดทั้งค่าสนับสนุน และค่าความมั่นใจในการคำนวณ ซึ่งจากผลการทดลองพบว่าอัลกอริทึม CBEAR(4) ให้ความแม่นยำในการทำนายมากกว่าอัลกอริทึมตัวอื่นๆ ในทุกๆฐานข้อมูล

เพื่อความชัดเจนมากยิ่งขึ้น ในภาพที่ 16 และ 17 จะแสดงให้เห็นถึงการเปรียบเทียบความแม่นยำของแต่ละอัลกอริทึมกับฐานข้อมูลแบบหนาแน่น และฐานข้อมูลแบบกระจาย ตามลำดับ



ภาพที่ 16 เปรียบเทียบความแม่นยำของแต่ละอัลกอริทึมกับฐานข้อมูลแบบหนาแน่น

จากภาพที่ 16 แสดงให้เห็นถึงอัลกอริทึม CBEAR ทั้ง 4 สูตร เปรียบเทียบกับอัลกอริทึมตัวอื่นๆในฐานข้อมูลแบบหนาแน่น โดยจะเห็นว่า อัลกอริทึม CBEAR ทั้ง 4 สูตร ให้ความแม่นยำกว่าอัลกอริทึมตัวอื่นๆ ทั้งหมด

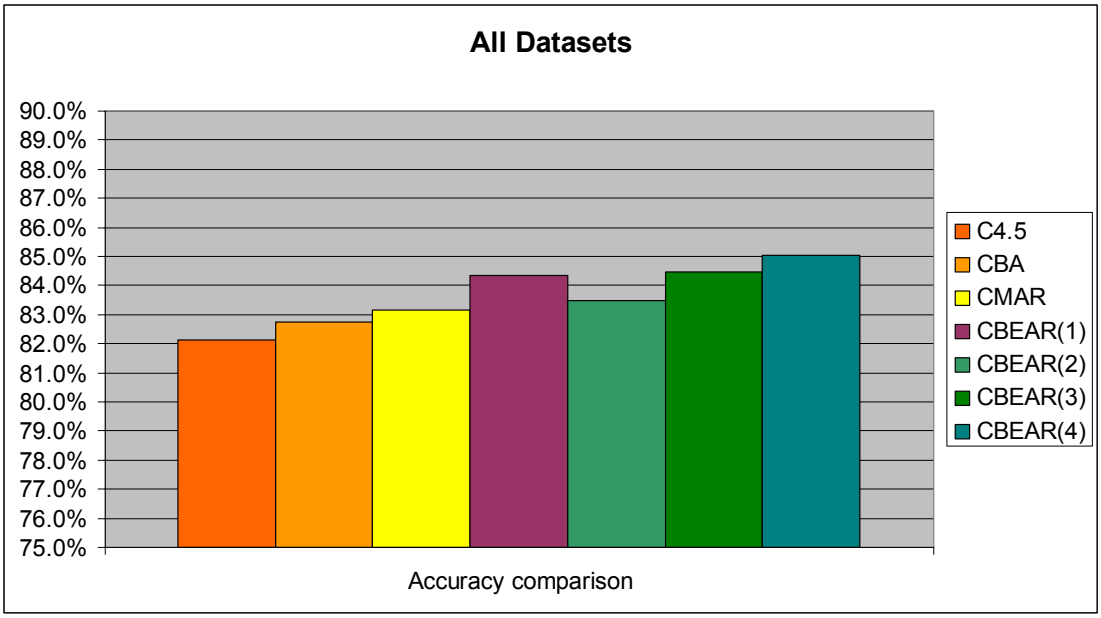


ภาพที่ 17 เปรียบเทียบความแม่นยำของแต่ละอัลกอริทึมกับฐานข้อมูลแบบกระจาย

จากภาพที่ 17 แสดงให้เห็นถึงอัลกอริทึม CBEAR ทั้ง 4 สูตร เปรียบเทียบกับอัลกอริทึมตัวอื่นๆในฐานข้อมูลแบบกระจาย โดยจะเห็นว่า อัลกอริทึม CBEAR ทั้ง 4 สูตร ให้ความแม่นยำกว่าอัลกอริทึมตัวอื่นๆ ทั้งหมด

จากภาพที่ 16 และ 17 จะเห็นว่าความแม่นยำในฐานข้อมูลแบบหนาแน่นจะสูงกว่า ความแม่นยำในฐานข้อมูลแบบกระจาย

ในส่วนของการเปรียบเทียบความแม่นยำของ CBEAR ทั้ง 4 สูตร กับอัลกอริทึมตัวอื่นๆ เมื่อทดสอบกับฐานข้อมูลทั้งหมด จะแสดงในภาพที่ 18



ภาพที่ 18 เปรียบเทียบความแม่นยำของแต่ละอัลกอริทึมกับฐานข้อมูลทั้งหมด

จากภาพที่ 18 แสดงให้เห็นถึงการเปรียบเทียบความแม่นยำของอัลกอริทึม CBEAR ทั้ง 4 สูตร กับอัลกอริทึมตัวอื่นๆ เมื่อทดลองกับฐานข้อมูลทั้งหมดจะเห็นว่า อัลกอริทึม CBEAR(4) ที่ใช้สูตรที่ 4 จะมีความแม่นยำในการทำนายมากที่สุด ส่วนอัลกอริทึมที่มีความแม่นยำรองลงมาคือ CBEAE(3), CBEAR(1) และ CBEAR(2) ตามลำดับ ซึ่งจะเห็นว่า อัลกอริทึม CBEAR ทั้ง 4 สูตร ให้ความแม่นยำในการทำนายข้อมูลได้สูงกว่า อัลกอริทึมอื่นๆ ไม่ว่าจะเป็น CMAR, CBA และ C4.5

วิจารณ์

จากผลการทดลอง แสดงให้เห็นว่าอัลกอริทึม CBEAR (Hanchodchung et al., 2006) มีประสิทธิภาพทั้งในด้านการลดจำนวนกฎความสัมพันธ์แบบมีคลาสลง โดยสามารถที่จะลดกฎลงได้โดยเฉลี่ยประมาณ 54 เปอร์เซ็นต์ ซึ่งถือว่าสามารถลดจำนวนกฎลงไปได้มาก ทำให้ประสิทธิภาพของระบบดีขึ้น โดยเฉพาะอย่างยิ่งกับฐานข้อมูลที่มีจำนวนแอตทริบิวต์และจำนวนแถวของข้อมูลมากๆ ก็จะยิ่งสามารถลดจำนวนกฎลงได้เยอะ เนื่องจากว่า แอตทริบิวต์ยิ่งมาก ก็ยิ่งจะทำให้เกิดซับเซตของ Itemsets มากขึ้นตามไปด้วย ดังนั้นจึงสามารถที่จะกำจัดกฎที่ยาวแต่ไม่มีประโยชน์ออกไปได้มากด้วย

เพราะฉะนั้น สามารถที่จะสรุปได้ว่าวิธีการลดจำนวนกฎแบบที่เสนอนี้ เหมาะสมที่จะนำไปใช้กับชุดข้อมูลที่มีขนาดใหญ่ๆ เนื่องจากสามารถที่จะลดจำนวนกฎลงได้มาก วิธีนี้จึงเหมาะสมกับการนำมาใช้กับข้อมูลในปัจจุบัน เนื่องจากข้อมูลในปัจจุบันนี้ก็มีขนาดและความซับซ้อนมากขึ้นเรื่อยๆ และในส่วนของความแม่นยำในการทำนายข้อมูล ก็ถือว่าอัลกอริทึม CBEAR มีความแม่นยำมากที่สุดโดยเฉพาะอย่างยิ่งในสูตรที่ 4 เนื่องจากมีการใช้หน่วยวัดทั้ง ค่าสนับสนุน และค่าความมั่นใจร่วมกัน ซึ่งจากการใช้หน่วยวัดหลายตัวร่วมกันนี้เองสามารถนำมายืนยันกับสมมุติฐานที่กล่าวไว้ตอนต้นว่า การใช้หน่วยวัดเพียงหน่วยเดียวในการพิจารณากฎเพื่อทำนายข้อมูลไม่ใช่วิธีที่ดีนั่นเอง

สรุปและข้อเสนอแนะ

สรุป

เทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ (Associative classification) เป็นเทคนิคที่ถูกสร้างขึ้นโดยการรวมเทคนิคที่สำคัญของดาต้าไมนิ่งเข้าไว้ด้วยกันนั่นก็คือ เทคนิคการจำแนกประเภทข้อมูล (Data classification) กับเทคนิคการสืบค้นกฎความสัมพันธ์ (Association rule discovery) ซึ่งทั้งสองเทคนิคที่ได้กล่าวมานี้ เป็นเทคนิคที่ได้รับความนิยมและถูกเผยแพร่อย่างกว้างขวางในศาสตร์ทางด้านคอมพิวเตอร์ ซึ่งหลักการของเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์จะแบ่งส่วนการทำงานออกเป็น 2 ส่วนนั่นก็คือ ส่วนที่ใช้ในการสร้างกฎความสัมพันธ์ซึ่งที่เรียกว่า Rule generator และในส่วนของการสร้างโมเดลในการทำนาย ซึ่งเรียกว่า Classifier builder นั่นเอง

โดยจากงานวิจัยก่อนหน้านี้พบว่า เทคนิคดังกล่าวได้ให้ความแม่นยำมากกว่าเทคนิคการจำแนกประเภทข้อมูลแบบเก่าๆ เช่น อัลกอริทึม C4.5 แต่ในช่วงแรกๆของเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์นั้นก็ยังคงมีจุดอ่อนอยู่ ซึ่งก็คือการใช้กฎความสัมพันธ์เพียงกฎเดียวในการพิจารณาเพื่อทำนายข้อมูล จากนั้นจึงมีผู้เสนอให้ใช้กฎความสัมพันธ์หลายๆกฎ หรือการนำกลุ่มของกฎความสัมพันธ์แบบมีคลาสมาใช้พิจารณาเพื่อทำนายข้อมูล ซึ่งจากผลการทดลองก็พบว่ามีความแม่นยำกว่าการใช้กฎเพียงกฎเดียวมาพิจารณาเพื่อทำนายข้อมูล

ในวิทยานิพนธ์เล่มนี้ได้ชี้ให้เห็นถึงปัญหาในการใช้กฎความสัมพันธ์แบบมีคลาส หลายๆกฎมาพิจารณาโดยไม่คำนึงถึงระดับหรือความยาวของกฎเหล่านั้น โดยวิธีการที่เราได้นำเสนอใหม่คือการพิจารณากฎความสัมพันธ์แบบมีคลาส หลายๆกฎพร้อมกันแต่มีเงื่อนไขว่าจะให้ความสำคัญกับกฎที่มีความยาวมากที่สุดก่อน หรือมีการกำหนดความสำคัญของกฎโดยพิจารณาจากความยาวของกฎนั่นเอง ซึ่งกฎที่มีความยาวมากก็น่าจะเป็นกฎที่มีความสำคัญมาก เนื่องจากมีความเหมือนหรือคล้ายกับคำถามมากที่สุด และยิ่งไปกว่านั้นเราได้นำเสนอวิธีการกำจัดกฎความสัมพันธ์แบบมีคลาสที่ซ้ำซ้อนออกไปด้วย โดยอัลกอริทึมที่เราได้เสนอนั้นชื่อว่า Classification Based on Essential Class-Association Rules (CBEAR) และจากการทดลองพบว่าอัลกอริทึม CBEAR มีประสิทธิภาพทั้งในด้านการลดจำนวนกฎความสัมพันธ์ และด้านความแม่นยำในการทำนายข้อมูล ซึ่งในการลดกฎนั้นอัลกอริทึม CBEAR สามารถที่จะลดจำนวนกฎลงได้โดยเฉลี่ยแล้วประมาณ 54 เปอร์เซ็นต์

และสามารถลดจำนวนกฎลงได้สูงสุดถึงประมาณ 85 เปอร์เซ็นต์อีกด้วย ในส่วนของความแม่นยำในการทำนายข้อมูล อัลกอริทึม CBEAR ก็ได้ให้ความแม่นยำสูงที่สุด โดยทำการเปรียบเทียบกับอัลกอริทึม C4.5 ซึ่งเป็นเทคนิคการจำแนกประเภทข้อมูลแบบเก่า และเปรียบเทียบกับอัลกอริทึม CBA และ CMAR ซึ่งเป็นอัลกอริทึมการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ที่ได้รับความนิยมในปัจจุบัน

ข้อเสนอแนะ

เนื่องจากเทคนิคการจำแนกประเภทข้อมูลโดยใช้กฎความสัมพันธ์ ประกอบไปด้วยส่วนที่สำคัญอยู่ 2 ส่วนนั่นคือ ส่วนในการสร้างกฎความสัมพันธ์ และส่วนของการสร้างโมเดลในการทำนาย ซึ่งในการเพิ่มประสิทธิภาพและความแม่นยำของเทคนิคดังกล่าวนั้นจะต้องคำนึงถึงทั้ง 2 ส่วนที่กล่าวมาข้างต้น โดยในส่วนการสร้างกฎความสัมพันธ์ นอกจากวิธีการกำจัดกฎความสัมพันธ์แบบมีคลาสที่ซ้ำซ้อนออกไปทั้งหมดตามที่งานวิจัยเล่มนี้ได้เสนอไปแล้ว ควรจะมีวิธีการในการกรองข้อมูลที่มีประโยชน์และไม่มีประโยชน์ออกจากกันก่อนที่จะนำเข้ามาในส่วนของการสร้างกฎความสัมพันธ์ เนื่องจากเหตุผลที่เคยกล่าวไว้แล้วว่าในส่วนการสร้างกฎความสัมพันธ์จะมีผลต่อความแม่นยำของโมเดลในการทำนายเป็นอย่างมาก ดังนั้น กฎความสัมพันธ์แบบมีคลาสที่ถูกสร้างขึ้นมาจะต้องเป็นกฎที่ความสำคัญเท่านั้น กฎที่ไม่มีความสำคัญหรือไม่มีประโยชน์จะส่งผลเสียต่อประสิทธิภาพและความแม่นยำของระบบ ดังนั้นจึงก็ไม่ควรจะถูกสร้างออกมานั่นเอง โดยที่วิธีในการกรองข้อมูลอาจจะทำได้โดยการตัดแอตทริบิวต์ที่ไม่มีความจำเป็นหรือไม่มีประโยชน์ออกไป หรืออาจจะนำหลายๆแอตทริบิวต์มารวมเข้าไว้ด้วยกัน ให้เกิดเป็นแอตทริบิวต์ใหม่ที่มีประโยชน์ออกมาได้

ในส่วนของการสร้างโมเดลในการทำนาย นอกจากวิธีการที่ได้นำเสนอไว้ในงานวิจัยเล่มนี้แล้ว ไม่ว่าจะเป็นวิธีการจัดเรียงกฎความสัมพันธ์แบบมีคลาส วิธีการกำหนดความสำคัญของกฎโดยจะให้ความสำคัญกับกฎที่มีความยาวมากที่สุดก่อน สิ่งที่ควรจะคำนึงถึงในการเพิ่มความแม่นยำให้กับโมเดลในการทำนายนั่นก็คือ จะต้องมีการเพิ่มส่วนในการพิจารณากฎให้มีความละเอียดมากยิ่งขึ้น โดยจะต้องคำนึงถึงสูตรที่จะนำมาใช้ รวมถึงการนำสูตรต่างๆมาพิจารณาพร้อมกันๆ ซึ่งจะทำให้ผลของการทำนายสูงขึ้นได้ เช่นอาจจะเพิ่มรายละเอียดในส่วนของการพิจารณากฎ โดยให้ความสำคัญกับกฎที่ยาวและมีค่าความมั่นใจสูง มากกว่ากฎที่มีความยาวแต่ค่าความมั่นใจต่ำ เป็นต้น

เอกสารและสิ่งอ้างอิง

วีระพล หาญโชติช่วง, ธนาวินท์ รักธรรมานนท์ และ กฤษณะ ไวยมัย. 2548. การเพิ่มประสิทธิภาพสำหรับเทคนิคการจำแนกข้อมูลโดยใช้กฎความสัมพันธ์. **การประชุมทางวิชาการระดับชาติด้านคอมพิวเตอร์และเทคโนโลยีสารสนเทศ ครั้งที่ 1 (NCCIT 05)** 1: 34.

Agrawal, R., and R. Srikant. 1994. Fast algorithm for mining association rules in large databases. *In* **Proceedings of the 20th International Conference on Very Large Data Bases (VLDB'94)** Santiago, Chile.

Agrawal, R., T. Imielinski., and A. Swami. 1993. Mining association rules between sets of items in large databases. *In* **Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data** Washington,DC.

Baralis, E., and S. Chiusano. 2004. Essential classification rule sets. *In* **ACM Transactions on Database Systems (TODS)**.

Baralis, E., S. Chiusano., and P. Garza. 2004. On support thresholds in associative classification. *In* **2004 ACM Symposium on Applied Computing**.

Blake, C., and C. Merz. 1998. **UCI repository of machine learning database** Department of Information and Computer Science, University of California, Irvine, Irvine, CA. Available Source: http://www.cs.uci.edu/~mlearn/MLRepository.html

Dong, G., X. Zhang., L. Wong., and J. Li. 1999. CAEP: Classification by aggregating emerging patterns. *In* **Proceedings of the 2nd International Conference on Discovery Science (DS'99)** Lecture Notes in Computer Science, vol. 1721. Springer-Verlag (LNCS 1721), New York.

Dougherty, J., R. Kohavi., and M. Sahami. 1995. Supervised and Unsupervised of continuous features. ICML-95

Fayyad, U., and K. Irani. 1993. Multi-interval discretization of continuous-valued attributes for classification learning. *In* **Proceedings of 13th International Joint Conference on Artificial Intelligence (IJCAI-93)** 13 , Morgan Kaufmann, San Francisco, CA.

Han, J., J. Pei., and Y. Yin. 2000. Mining frequent patterns without candidate generation. *In* **Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data (ACM SIGMOD'00)** ACM Press, New York, NY.

Hanchodchung, V., T. Rakthanmanon., K. Waiyamai. 2006. CBEAR: Classification based on essential class-association rules. *In* **The 10th Annual National Symposium on Computational Science and Engineering (ANSCSE10)** 10: 252-257.

Janssens, D., G. Wets., T. Brijs., K. Vanhoof., and G. Chen. 2003. Adapting the CBA-algorithm by means of intensity of implication. *In* **Proceedings of the First International Conference on Fuzzy Information Processing Theories and Application** Beijing (China).

Kohavi, R., and M. Sahami. 1996. Error-based and Entropy-based discretization of continuous features. *In* **Proceedings of the Second International Conference on Knowledge Discovery and Data Mining**.

Liu, B., W. Hsu., and Y. Ma. 1998. Integrating classification and association rule mining. *In* **Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining (KDD'98)** 4: 80-86.

Lui, B., Y. Ma., and CK. Wong. 2001. Classification using association rules: weaknesses and enhancements. *In* **Vipin Kumar,et al,(eds), Data mining for scientific and engineering applications**.

Pasquier, N., Y. Bastide., R. Taouil., and L. Lakhal. 1999a. Closed itemsets discovery of small covers for association rules. *In* **Proceedings of 15mes Jornes Based des Donnes Avances (BDA'99)** (15): 361-381.

Pasquier, N., Y. Bastide., R. Taouil., and L. Lakhal. 1999b. Discovering frequent closed itemsets for association rules. *In* **Proceedings of the 7th International Conference on Databased Theory (ICDT'99)** Lecture Notes in Computer Science, vol 1540. Springer-Verlag, New York 7: 398-416.

Pei, J., J. Han., and R. Mao. 2000. CLOSET: An efficient algorithms for mining frequent closed itemsets. *In* **Proceeding of the ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery (DMKD'00)** ACM Press, New York.

Quinlan, J.R. 1993. C4.5: Programs for machine learning. **Morgan Kaufmann**

Wang, K., S. Zhou., and Y. He. 2000. Growing decision tree on support-less association rules. *In* **Proceedings of the 6th International Conference on Knowledge Discovery and Data Mining (KDD'00)** Boston, MA, Aug.

Wenmin, Li. 2001. **Classificaion based on multiple association rules.** M.S. Thesis, Simon Fraser University.

Wenmin, Li., H. Jiawei., and J. Pei. 2001. CMAR: Accurate and efficient classification based on multiple class-association rules. *In* **Proceedings of the IEEE International Conference on Data Mining (ICDM'01)** IEEE Computer Society Press, Los Alamitos, CA.

Yin, X., and J. Han. 2003. CPAR: Classification based on Predictive Association Rules.

ประวัติการศึกษา และการทำงาน

ชื่อ –นามสกุล	นายวีระพล หาญโชติช่วง
วัน เดือน ปี ที่เกิด	วันที่ 10 มิถุนายน 2524
สถานที่เกิด	กรุงเทพมหานคร
ประวัติการศึกษา	วท.บ. (วิทยาการคอมพิวเตอร์) คณะวิทยาศาสตร์ มหาวิทยาลัยกรุงเทพ (พ.ศ.2545)
ตำแหน่งหน้าที่การงานปัจจุบัน	อาจารย์พิเศษ
สถานที่ทำงานปัจจุบัน	มหาวิทยาลัยกรุงเทพ
ผลงานดีเด่นและรางวัลทางวิชาการ	
ทุนการศึกษาที่ได้รับ	