การพัฒนาระบบการค้นหาความสัมพันธ์แบบหลายลำดับชั้นโดยใช้ Cumulate Algorithm

Mining Multi-Level Association Rules Discovery using Cumulate Algorithm

โดย

นาย เอกภูมิ อารีรัตน์

รหัส 45066074

อาจารย์ที่ปรึกษา

ผศ.ดร. วรพจน์ กรีสุระเดช

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 2 ปีการศึกษา 2546
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อหัวข้อ	การพัฒนาระบบการค้นหาความสัมพันธ์แบบหลายลำดับชั้นโดยใช้ Cumulate Algorithm
นักศึกษา	นาย เอกภูมิ อารีรัตน์
อาจารย์ที่ปรึกษา	ดร.วรพจน์ กรีสุระเดช
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2546

บทคัดย่อ

ในโครงงานพัฒนาระบบงานนี้ จะทำการศึกษาการทำ data mining ในการวิเคราะห์ค้นหาความสัมพันธ์ของข้อมูล ซึ่งเทคนิคที่จะทำการศึกษา มีความสามารถที่จะหาความสัมพันธ์ของข้อมูลได้แบบหลายลำดับชั้น โดยใช้ Cumulate Algorithm ในการประมวลผล ซึ่ง Cumulate algorithm นี้เป็น algorithm พื้นฐานของเทคนิคโดยจะศึกษาถึงขั้นตอนการทำงานของ Algorithm และ นำมาพัฒนาเป็นระบบเพื่อช่วยในการค้นหาความสัมพันธ์ ของข้อมูล ซึ่งสามารถนำไปใช้หาความสัมพันธ์ของข้อมูลที่ถูกจัดเก็บเอาไว้ภายในฐานข้อมูล และ นำมาใช้ให้ เกิดประโยชน์ได้

Title	Mining Multi-Level Association Rules Discovery using Cumulate Algorithm
Student	Mr. Eakapoom Areerat
Advisor	Dr. Worapoj Keesuredej
Level of Study	Master of Science in Information Technology
Major	Information Science
Academic Year	2003

ABSTRACT

The system development project studies the data mining in order to analyze the Association Discovery. Therefore, the Cumulate Algorithm is used in the process as a basic technique that shows the Multi-Level Association Rules Discovery by providing the Algorithm procedure. As a result, the Cumulate Algorithm also can be developed as a system in order to provide the Data Association stored in the Data Based System.

กิตติกรรมประกาศ

ข้าพเจ้าขอขอบคุณ ดร.วรพจน์ กรีสุระเดช อาจารย์ที่ปรึกษาวิชาโครงการพัฒนาระบบงาน ที่ได้กรุณาสละเวลา ให้ความรู้ ให้คำปรึกษา ให้ความเอาใจใส่ และให้คำแนะนำต่างๆ อันเป็นประโยชน์ต่อการพัฒนาระบบ ของข้าพเจ้าเป็นอย่างมาก

นอกจากนี้ข้าพเจ้าขอกราบขอบพระคุณบุพการี ที่ได้ให้กำลังใจในการทำโครงการนี้ตลอดมา ตลอดจนขอขอบคุณเพื่อนๆ IS13.2 ที่มีส่วนให้ความช่วยเหลือ เป็นกำลังใจ และสนับสนุนให้ผลงานนี้สำเร็จลุล่วงด้วยดี

ขอบคุณครับ

เอกภูมิ อารีรัตน์

สารบัญ

หน้า

สารบัญตาราง

หน้า

ตารางที่

สารบัญภาพ

หน้า

ภาพที่

บทที่ 1

บทนำ

ในปัจจุบันโลกเข้าสู่ยุคของข้อมูลข่าวสาร สารสนเทศ(Information) จึงเป็นสิ่งที่มีความจำเป็นอย่างยิ่งที่ทางภาครัฐและเอกชนจะต้องอาศัยสารสนเทศนี้เข้ามาช่วยในการตัดสินใจ (Make decision) การตัดสินใจที่ล่าช้าอาจจะก่อให้เกิดผลเสียตามมามากมาย อาจจะทำให้เกิดการสูญเสียโอกาสในการแข่งขันทางธุรกิจได้ เทคโนโลยีทางด้านข่าวสารข้อมูลได้มีการพัฒนาให้ก้าวหน้ามากขึ้นนอกเหนือจากการดำเนินงานยังช่วยในการจัดการธุรกิจ เช่น ธุรกิจธนาคาร บริษัทประกันภัย บริษัทท่องเที่ยว โรงงานอุตสาหกรรม โรงพยาบาล โรงแรม และ สถาบันการศึกษา เป็นต้น และยังช่วยในเรื่องของการแข่งขันทางการตลาดโดยการใช้เทคโนโลยีสมัยใหม่ในการดึงดูดลูกค้า

ท่ามกลางการแข่งขันที่ดุเดือด ในโลกของธุรกิจในปัจจุบันนั้น สารสนเทศจึงมีบทบาทเข้ามาช่วยในการดำเนินธุรกิจขององค์กรต่างๆ เพื่อช่วยสนับสนุนการตัดสินใจ (Decision Support) เช่น การประเมินความเสี่ยงของบริษัท การประเมินสถานการณ์ล่วงหน้า การพยากรณ์ทางธุรกิจ เป็นต้น โดยนำข้อมูลที่ถูกจัดเก็บในคลังข้อมูล (Data Warehousing) มาวิเคราะห์หาความสัมพันธ์ที่ซ่อนอยู่ระหว่างข้อมูลที่ถูกจัดเก็บไว้ในคลังข้อมูล และดึงสารสนเทศที่ได้นั้นมาช่วยในการตั้งเป้าหมายทางธุรกิจให้ดีขึ้น รวมทั้งช่วยในการทำนายแนวโน้มและพฤติกรรมของข้อมูลในอนาคต จึงได้นำเอาเทคนิคของดาต้าไมนิ่ง (Data Mining) เข้ามาช่วยในการวิเคราะห์ข้อมูลเพื่อให้ทราบถึงความสัมพันธ์ในรูปแบบต่างๆ ที่ซ่อนอยู่ในคลังข้อมูล ออกมาใช้ให้เกิดประโยชน์

1.1. วัตถุประสงค์

- เพื่อให้เข้าใจถึงวิธีการและขั้นตอนการทำงานของเทคนิคดาต้าไมนิ่ง
- เพื่อให้เข้าใจถึงวิธีการวิเคราะห์หาความสัมพันธ์ของข้อมูลที่ซ่อน อยู่ในคลังข้อมูล
- เพื่อให้เข้าใจถึงการทำงานของ Cumulate Algorithm ที่ใช้หาความสัมพันธ์ของข้อมูล

1.2 ขอบเขตการดำเนินงาน

โครงการนี้เป็นการศึกษาถึงการนำเอาเทคนิคของดาต้าไมนิ่งมาใช้หาความสัมพันธ์ของข้อมูล โดยใช้ Link Analysis Model ซึ่งมีการใช้เทคนิคที่เรียกว่า Multi-Level Association Discovery ในการวิเคราะห์หาความสัมพันธ์ของข้อมูล ซึ่งสามารถหาความสัมพันธ์ของข้อมูลได้หลายลำดับชั้น

1.3 เป้าหมายของการพัฒนาระบบงาน

- เข้าใจถึงวิธีการของ Data Mining
- เข้าใจถึงวิธีการหาความสัมพันธ์ของข้อมูลแบบหลายลำดับชั้น
- โปรแกรมสามารถค้นหาและรายงาน ความสัมพันธ์ของข้อมูลที่เกิดขึ้นแบบหลายลำดับชั้นได้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- เข้าใจถึงหลักการและวิธีการ ร่วมไปถึงขั้นตอนการทำงานของ Data Mining
- นำข้อมูลที่ถูกเก็บไว้ในคลังข้อมูล มาหาความสัมพันธ์ที่ซ่อนอยู่ และนำไปใช้ให้เกิดประโยชน์ขึ้นได้
- เข้าใจถึงเทคนิคการหาความสัมพันธ์ของข้อมูลแบบหลายลำดับชั้น
- สามารถนำการพัฒนาระบบที่ทำการศึกษานี้ ไปประยุกต์ใช้ให้ก่อเกิดประโยชน์ ในองค์กรทางด้านธุรกิจ

บทที่ 2

Data Minimg

Data Mining คือขบวนการทำงานที่เรียกว่า Process ที่สกัดข้อมูล (Extract data) จากฐานข้อมูลขนาดใหญ่ (Large Information) เพื่อให้ได้สารสนเทศ (Usefull Information) ที่เรายังไม่รู้ (UnKnown data) โดยเป็นสารสนเทศที่มีเหตุผล (Valid) และสามารถนำไปใช้ได้(Actionble) ซึ่งเป็นสิ่งสำคัญในการที่จะช่วยการตัดสินใจในการทำธุรกิจ Data Mining เป็นโปรเซสที่สำคัญในการทำ Knowledge Discovery in Database

2.1 Data Mining

จากตอนต้น Data Mining เป็นขบวนการที่สำคัญที่จะทำการสกัดส่วนที่เป็นนัยของข้อมูลในฐานข้อมูล ที่เราไม่ทราบ มาใช้ให้เกิดประโยชน์ กระบวนการค้นหาหรือสกัดสารสนเทศจากฐานข้อมูลขนาดใหญ่นี้ จำเป็นที่จะต้องผ่านกระบวนการจัดเตรียมข้อมูล(Preprocess Data) การค้นหาและจัดรูปแบบ (Search for pattern) จนกระทั่งได้ข้อมูลตามต้องการ โดยวิธีการค้นหานี้ ทำได้โดย

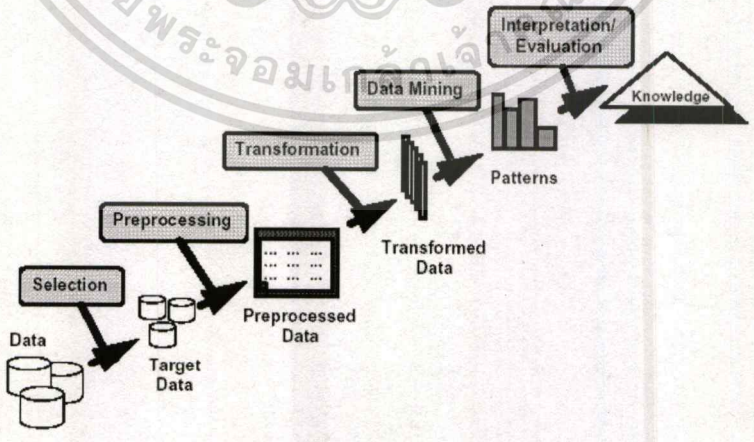
- ผู้ใช้เป็นผู้กำหนดคำถาม และ ระบบจะเป็นผู้ตอบคำถามเหล่านั้น เช่น อาจใช้การซักถาม (Query) และการรายงาน (Reporting) ซึ่งวิธีการนี้มีข้อบกพร่องจากการค้นหาคือ ผู้ใช้ไม่ได้คิดถึงสิ่งที่สัมพันธ์กันหรือสิ่งที่ต้องการถามได้อย่างครอบคลุมทั้งหมด ทำให้ข้อมูลส่วนที่สำคัญหลายส่วนอาจไม่ได้ถูกคัดเลือก
- โปรแกรมทางด้าน Data Mining จะค้นหาข้อมูลโดยอัตโนมัติโดยโปรแกรมจะคิดคำถามที่น่าสนใจด้วยตัวเอง เมื่อพบข่าวสารแล้วจะแสดงในรูปแบบที่เหมาะสม เช่น รายงาน กราฟ

Data Mining เป็นเทคโนโลยีในการค้นหาความรู้ในฐานข้อมูลโดยไม่ต้องตั้งสมมติฐานไว้ล่วงหน้า แต่เป็นการนำความรู้ที่ได้มาทดสอบสมมติฐานภายหลัง สารสนเทศที่ได้มาจากการทำดาต้าไมนิ่งต้องมีลักษณะไม่รู้มาก่อนล่วงหน้า(Unknown) เป็นข้อมูลที่มีความถูกต้อง(Valid) และสามารถนำไปใช้ประโยชน์ได้จริง(Actionable)

- **Unknown** เป็นข้อมูลที่ผู้ใช้เคยไม่รู้มาก่อน ไม่ชัดเจน และ ไม่สามารถตั้งสมมติฐานล่วงหน้าว่าควรเป็นเช่นใด เช่น เจ้าของห้างสรรพสินค้าแห่งหนึ่งค้นพบพฤติกรรมของผู้บริโภคว่าผู้บริโภค ที่เป็นพ่อบ้านส่วนใหญ่มักจะซื้อเบียร์และผ้าอ้อมด้วยกันในวันศุกร์ช่วงเวลาเลิกงาน จากข้อมูลที่ค้นพบนี้ จะเป็นประโยชน์แก่เจ้าของห้างสรรพสินค้า เพื่อทำการจัดเตรียมสินค้าไว้จำหน่ายในเย็นวันศุกร์ โดยในขณะเดียวกันห้างสรรพสินค้าอื่นๆ ที่เป็นคู่แข่งอาจยังไม่รู้พฤติกรรมของผู้บริโภคเลยก็เป็นได้
- **Valid** เป็นข้อมูลที่มีความถูกต้อง เนื่องจากผู้ใช้จะค้นพบสิ่งที่น่าสนใจตลอดเวลา แต่ต้องทำการพิจารณาด้วยว่าสิ่งนั้นถูกต้องหรือไม่ เช่น ผู้ใช้มักพบว่าเมื่อจำนวนความหลากหลายของสินค้ามากขึ้นจะมีความสัมพันธ์ของการซื้อของ 2 สิ่งเสมอ แต่ไม่ได้หมายความว่าจะต้องให้ห้างสรรพสินค้าเก็บสินค้ามากขึ้น เพราะข้อมูลที่ได้อาจเกิดจากความคลาดเคลื่อน
- **Actionable** เป็นข้อมูลที่สามารถนำไปใช้ประโยชน์ได้จริง จะต้องถูกแปลงออกมาและนำมาตัดสินใจให้เป็นความได้เปรียบเชิงธุรกิจ บางครั้งข้อมูลที่เราค้นพบเป็นสิ่งที่คู่แข่งได้ทำไปแล้วหรือเป็นสิ่งที่ผิดกฎหมาย ข้อมูลดังกล่าวจะไม่มีประโยชน์อะไร ดังนั้น จะเป็นต้องใช้วิจารณญานในการเลือกใช้ข้อมูลด้วย

2.2. กระบวนการทำงานของดาต้าไมนิ่ง

กระบวนการ Data Mining เป็นกระบวนการของการสร้างแบบจำลอง (Model) ประกอบด้วย 5 ขั้นตอนดังแสดงในภาพที่ 2.1



ภาพที่ 2.1 กระบวนการทำ Data Mining

กระบวนการ Data Mining นั้นประกอบไปด้วย 5 ขั้นตอน และแต่ละขั้นตอนอาจมีการทำงานมากกว่า 1 ครั้ง และ อาจจะมีการวนกลับมาทำงานใหม่อีกครั้ง รูปที่ 2 จะแสดงเปอร์เซ็นต์การทำงานของแต่ละขั้นตอนในการทำ Data Mining ซึ่งแต่ละขั้นตอนจะมีรายละเอียดในการทำงานดังนี้

2.2.1 Business Objective Determination

การกำหนดจุดประสงค์ทางธุรกิจ จะต้องกำหนดปัญหาและเป้าหมายให้ชัดเจน ซึ่งในส่วนนี้จะประกอบด้วยการวิเคราห์ทางธุรกิจ และการวิเคราะห์ข้อมูลเบื้องต้นว่าเรามีข้อมูลใดอยู่บ้างและต้องการอะไรจากข้อมูล ซึ่งเป้าหมายทางธุรกิจนี้จะนำไปสู่การสร้าง Model ที่เหมาะสม ซึ่งModel ที่สร้างขึ้นจะแตกต่างกัน โดยขึ้นอยู่กับเป้าหมายทางธุรกิจ

2.2.2 Data Preparation

การจัดเตรียมข้อมูล เป็นขั้นตอนที่สำคัญที่สุด และเป็นช่วงที่ใช้เวลามากที่สุด โดยปกติแล้วต้องการเวลาประมาณ 60% ของเวลาทั้งหมดในการเตรียมข้อมูล ซึ่งต้องใช้เวลาและความพยายามมากกว่าขั้นตอนอื่นๆ ทั้งหมด เนื่องจากจะต้องมีการพิจารณาข้อมูลเกือบทั้งหมด และบางครั้งอาจมีการนำข้อมูลจากหลายแหล่งมารวมกันเพื่อดูความสัมพันธ์ของข้อมูล โดยจะต้องมีการย้อนกลับมาทำซ้ำในขั้นตอนการเตรียมข้อมูล และขั้นตอนการสร้าง Model เนื่องจากการเรียนรู้บางสิ่งจาก Model อาจจำไปสู่การแก้ไขข้อมูล ข้อมูลที่ได้จากขั้นตอนนี้จะต้องมีความชัดเจน ถูกต้อง ซึ่งขั้นตอนการเตรียมข้อมูลประกอบไปด้วย 3 ขั้นตอนดังนี้

2.2.2.1 Data selection

คือ การเลือกข้อมูลที่ต้องการและข้อมูลที่ไม่ต้องการออกไปซึ่งเป็นการเริ่มต้นของการเตรียมการไมนิ่ง การเลือกข้อมูลนั้นจะแตกต่างไปตามวัตถุประสงค์ของแต่ละธุรกิจที่ได้กำหนดไว้ตอนต้น ซึ่งการเลือกข้อมูลจำเป็นที่จะต้องมีความเข้าใจกับชนิดของข้อมูล และประเภทของข้อมูลที่จะต้องนำมาใช้ด้วย โดยประเภทขอข้อมูลแบ่งได้ 2 ลักษณะ คือ

- **Categorical**
  - Nominal : เป็นตัวแปรที่ลำดับไม่มีความสำคัญ(ลำดับไม่มีผลกับค่า) เช่น สถานะ ก แต่งงาน(โสด ,แต่งงาน ,หม้าย)
  - Ordinal : เป็นตัวแปรที่ลำดับมีความสำคัญ(ลำดับมีผลกับค่า) เช่น เช่น ลำดับของสินค้า (A ,B, C, D, F)
- **Quantitative**
  - Continuous : จะเก็บค่าตัวเลขที่เป็นจำนวนจริง (Real number) เช่น ข้อมูลจำนวนพนักงาน
  - Discrete : จะเก็บค่าตัวเลขที่เป็นจำนวนเต็ม (Integer) เช่น จำนวนพนักงานในบริษัท

ในการเลือกข้อมูลต้องคำนึงถึงอายุของข้อมูลด้วย เช่น ข้อมูลอาชีพของลูกค้า ซึ่งจะมีการเปลี่ยนแปลงบ่อยเมื่อเวลาผ่านไปเพราะฉะนั้นการนำเอาข้อมูลอาชีพของลูกค้ามาใช้นั้นต้องตรวจสอบให้แน่ชัดว่าข้อมูลนั้นถูกต้องหรือไม่

2.2.2.2 Data Preprocessing

คือ ตรวจสอบข้อมูล เพื่อทำให้มั่นใจว่าคุณภาพของข้อมูลที่ถูกเลือกนั้นถูกต้อง และ เหมาะสมที่จะนำมาทำดาต้าไมนิ่ง เนื่องจากข้อมูลที่ถูกเลือกมาจากกระบวนการ Data Selection อาจมีข้อมูลที่ไม่ถูกต้อง ดังนั้นในขั้นตอนนี้มีประเด็นที่จะต้องพิจารณาเพิ่มเติม 2 ประเด็นคือ

- **Noisy Data :** เป็นข้อมูลที่มีลักษณะแตกต่างจากข้อมูลที่คาดการณ์เอาไว้ หรือ ค่าของข้อมูลอาจจะผิดไปจากที่ควรจะเป็น ซึ่งอาจจะเกิดจากการป้อนข้อมูลผิด เช่น ใส่อายุเป็น 650 ปีหรือป้อนรายได้เป็นข้อมูลติดลบ เป็นต้น ซึ่งข้อมูลที่ผิดนี้อาจไปรบกวนการวิเคราะห์ จึงต้องทำการกำจัดข้อมูลที่ผิดนี้ออกไป
- **Missing Data :** เป็นค่าของข้อมูลที่ไม่ได้ถูกเลือกมาจากขั้นตอน Data Selection หรืออาจจะเป็นค่าที่ไม่สมบูรณ์ ที่เราทำการลบออกไประหว่างการทำ Noise Detection ค่าอาจหายไปเพราะเกิดจากความเลินเล่อของมนุษย์ สามารถแก้ไขได้โดยทำการตัดข้อมูลนั้นทิ้งทั้งรายการ หรือทำการบันทึกส่วนที่ขาดหายไปด้วยค่าเฉลี่ย (Mean) หรือค่าที่ปรากฏบ่อย (Mode) สำหรับข้อมูลประเภท Quantitative ส่วนข้อมูลประเภท Categorical อาจบันทึกด้วยค่าที่ปรากฏบ่อย (Mode) หรือบันทึกเป็น "Unknown"

2.2.2.3 Data Transformation

เป็นการแปลงข้อมูลให้อยู่ในรูปแบบของข้อมูลที่พร้อมจะนำไปวิเคราะห์กับอัลกอลิทึม ที่ใช้กับเทคนิคต่างๆของ Data Mining เช่นการแปลงตัวเลขให้เป็นช่วงๆ เพื่อใช้กับอัลกอลิทึมของ Decision Tree หรือการปรับอัตราส่วนตัวเลขให้อยู่ในช่วง 0-1 เพื่อใช้กับ Neural Network เป็นต้น

2.2.3 Data Mining

เป็นการประมวณผลข้อมูลตามอัลกอริทึมที่ได้กำหนดเอาไว้ ซึ่งในขั้นตอนนี้จะมีความสัมพันธ์กับการวิเคราะห์ข้อมูลและขั้นตอน Data Transformation ที่ผ่านมา โดยเมื่อทำในส่วนของดาต้าไมนิ่งแล้วอาจต้องย้อนกลับไปทำในขั้นตอนของการเตรียมข้อมูลใหม่ ในการพัฒนาในส่วนของดาต้าไมนิ่ง จะเกี่ยวข้องกับการใช้ อัลกอริทึมหลายๆแบบ ซึ่งแต่ละแบบมีข้อดีและข้อเสียที่แตกต่างกัน

Data Mining ประกอบด้วย 4 โมเดลหลักที่ใช้สำหรับงานทางธุรกิจ ดังนี้

2.2.3.1 Predictive Modeling

เป็นโมเดลที่ใช้ในการสร้างแบบจำลองพยากรณ์ มีลักษณะคล้ายการเรียนรู้ของมนุษย์ คือจะต้องเข้าใจลักษณะของสิ่งที่จะศึกษาอย่างแท้จริง เราจะใช้โมเดลนี้ใจการวิเคราะห์ฐานข้อมูลที่มีอยู่ เพื่อกำหนดคุณสมบัติที่สำคัญของข้อมูล เพราะฉะนั้นข้อมูลที่มีอยู่ต้องเป็นข้อมูลที่สมบูรณ์ แบบจำลองจึงจะให้คำทำนายที่ถูกต้อง การพัฒนาแบบจำลองพยากรณ์จะนำเอาข้อมูลในอดีตมาสร้างเป็นแบบจำลองโดยแบ่งออกเป็น 2 ขั้นตอนคือ

- **Training Phase :** ช่วงการเรียนรู้ เป็นการสร้างโมเดลโดยการใช้ข้อมูลในอดีตและมีจำนวนข้อมูลจำนวนมาก

- **Testing Phase :** เป็นช่วงตรวจสอบความน่าเชื่อถือและประสิทธิภาพของโมเดลที่สร้างขึ้นมาว่ามีความเหมาะสมหรือไม่ โดยใช้กับข้อมูลที่ถูกแบ่งเอาไว้สำหรับการทดสอบ ซึ่งเป็นข้อมูลที่มีจำนวนไม่มากนัก

Predictive Modeling สามารถแบ่งได้เป็น 2 เทคนิคคือ

- Classification เป็นการทำนายกลุ่มของรายการจากข้อมูลที่สนใจ ซึ่งกลุ่มต่างๆได้มีการกำหนดไว้ล่วงหน้าแล้ว เช่น ในการจัดกลุ่มของลูกค้าเพื่อทำนายลักษณะของลูกค้าที่เปลี่ยนไปใช้บริการของคู่แข่ง เป็นต้น
- Value Prediction เป็นการทำนายค่าที่เป็นตัวเลข ใช้เพื่อทำนายเหตุการณ์ในอนาคต เช่น การทำนายราคาหุ้น หรือการทำนายภาษีที่จะจัดเก็บในปีหน้า เป็นต้น

2.2.3.2 Link Analysis

เป็นการวิเคราะห์หาความสัมพันธ์ระหว่างข้อมูล ว่าข้อมูลแต่ละรายการ มีความสัมพันธ์กันหรือไม่ อย่างไร ความสัมพันธ์นี้เรียกว่า "Association" เช่น เก็บข้อมูลการซื้อสินค้าแต่ละครั้งของลูกค้าเพื่อศึกษาพฤติกรรมการซื้อสินค้า เพื่อนำมาทำนายการส่งเสริมการขายและการจัดชั้นวางสินค้าให้เหมาะสม การวิเคราะห์หาความสัมพันธ์นี้สามารถแบ่งได้เป็น 3 ลักษณะ ตามการวิเคราะห์จากข้อมูลดิบ ดังนี้

- **Association Discovery**

  เป็นการวิเคราะห์ข้อมูลที่เกิดขึ้นพร้อมกันภายในกลุ่มข้อมูลเดียวกัน เป็นเทคนิคที่นิยมมากชนิดหนึ่ง มักจะใช้ในการวิเคราะห์ถึงพฤติกรรมการซื้อของผู้บริโภค จึงเป็นเทคนิคที่มีอีกชื่อหนึ่งที่เรียกว่า "Market Basket Analysis

- **Sequential Pattern Discovery**

  เป็นการศึกษาความสัมพันธ์ระหว่างข้อมูล โดยเทียบข้อมูลกับเวลา ซึ่งเป็นการศึกษาพฤติกรรมในระยะยาว (Long Term Behavior)

- **Similar Time Sequence Discovery**

  เป็นพฤติกรรมของข้อมูลที่เกิดขึ้นทั้งหมดหรือเกิดขึ้นในช่วงเวลาเดียวกัน เพื่อหาความสัมพันธ์ระหว่างกลุ่มของข้อมูลเหล่านี้

2.2.3.2 Deviation Detection

เป็นการตรวจสอบค่าเบี่ยงเบน เป็นโมเดลที่จะใช้เทคนิคทางสถิติ(Statistics) และการทำให้เห็นภาพ(Visualization) ซึ่งเป็นการสรุปข้อมูลให้ออกมาในรูปแบบการแสดงผลกราฟฟิก(Graphic) มักจะนำไปใช้งานทางด้านตรวจสอบ จับผิด เช่น ตรวจลายเซ็นปลอมของบัตรเครดิต หรือ ตรวจหาจุดบกพร่องของชิ้นงานในโรงงานอุตสาหกรรม เป็นต้น

2.2.4 Analysis of Results

ขั้นตอนที่ทำความเข้าใจกับแบบจำลอง ซึ่งจะทำการวิเคราะห์ผลของการประมวลผล ซึ่งจะทำการแปลความหมายและประเมินผลที่ได้จากขั้นตอนการทำไมนิ่งว่าสามารถนำไปใช้บรรลุวัตถุประสงค์ที่ต้องการได้หรือไม่รวมทั้งเป็นการประเมินถึงความถูกต้องของผลที่ได้จากการทำเพราะบางครั้งผลที่ได้จากการทำไมนิ่ง อาจจะมีข้อผิดผลาดได้ เพราะฉะนั้นการทำงานในส่วนนี้จึงจำเป็นที่จะต้องใช้ทักษะในการวิเคราะห์ข้อมูลและการวิเคราะห์ทางธุรกิจเข้ามาช่วย เครื่องมือทางด้าน Graphical Visualization จะช่วยวิเคราะห์ข้อมูลได้อย่างสะดวกและรวดเร็วขึ้น

2.2.5 Assimilation of Knowledge

ขั้นตอนปรับความรู้ที่ได้เข้ากับธุรกิจ เป็นการรวบรวมความเข้าใจทางธุรกิจที่เป็นผลมาจากขั้นตอน Analysis of Results มารวมเข้ากับส่วนความรู้เพื่อนำไปใช้ในโอกาสต่อไป ในขั้นตอนนี้มีหลักอยู่ 2 ประการ คือ

- การนำเสอนแนวคิดทางธุรกิจที่ค้นพบใหม่
- หาแนวทางที่จะใช้กฎเกณฑ์ใหม่ที่ค้นพบเพื่อให้เกิดประโยชน์สูงสุด

บทที่ 3

Link Analysis

เป็นโมเดลที่ให้ค้นหาความสัมพันธ์ของข้อมูลโดยมุ่งเน้นการทำงานบนเรคอร์ดที่สนใจ เพื่อค้นหาความสัมพันธ์ หรือ ความเกี่ยวโยงกันระหว่างเรคอร์ด หรือกลุ่มของเรคอร์ด เช่น ค้นหาความสัมพันธ์ของสินค้าผลิตภัณฑ์ หรือบริการที่ลูกค้าสนใจในเวลาหนึ่งๆ หลักการสำคัญของ Link Analysis มีอยู่ 3 ประการ ดังนี้

1. Association Discovery
2. Sequential Pattern Discovery
3. Similar Time Sequence Discovery

3.1 Association Discovery

ใช้วิเคราะห์หาความสัมพันธ์ที่ซ้อนอยู่ของสินค้า(item) ที่เกิดขึ้นในรายการเดียวกัน ที่มีแนวโน้มว่าสินค้ามักจะถูกซื้อควบคู่กันไปในรายการเดียวกัน การวิเคราะห์ในลักษณะนี้เรียกว่า "Market Basket Analysis" (MBA) ซึ่งจะนำไปใช้วิเคราะห์การขายของสินค้า เพื่อวิเคราะห์หากลุ่มของสินค้า(item set) ที่มักจะถูกซื้อควบคู่กันในแต่ละ Transaction ทำให้ผู้ประกอบธุรกิจหรือผู้วิเคราะห์สามารถนำ MBA นี้ไปช่วยในการวางแผนธุรกิจได้ เช่น

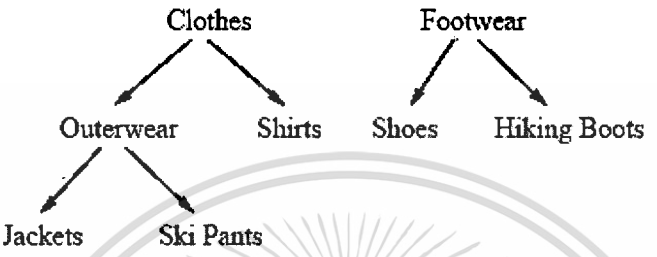
- คูปองและการลดราคา ถ้ามีการลดราคาเบียร์และผ้าอ้อมในเวลาเดียวกัน คงจะไม่ดีแน่ เพราะลูกค้ามีแนวโน้มที่จะซื้อสินค้าคู่นี้พร้อมกันอยู่แล้ว จึงควรที่จะลดราคาสินค้าตัวใดตัวหนึ่ง เพื่อดึงยอดขายของสินค้าอีกตัว
- การวางตำแหน่งของสินค้า การจัดวางสินค้าที่มีความสัมพันธ์กันเอาไว้ใกล้กัน ย่อมทำให้ลูกค้าหยิบสินค้าได้สะดวกขึ้น

3.1.1. Association Rule

การหากฎความสัมพันธ์ จะใช้อัลกอริทึมในการทำงานซึ่งมีอยู่ 3 ขั้นตอน หลักๆ ได้แก่

(1) การเลือกชุดข้อมูลที่ถูกต้อง

ขั้นตอนนี้เป็นการพิจารณาว่าจะนำข้อมูลระดับใดมาใช้ โดยจะพิจารณาจากวัตถุประสงค์ซึ่งปกติ ข้อมูลจะถูกจัดเก็บอยู่ในรูป Transaction ซึ่งข้อมูลจะอยู่ในระดับที่ละเอียด อาจจะถูกจัดเก็บมาจากจุดขายสินค้า จึงจำเป็นที่จะต้องมาผ่านการเลือกชุดข้อมูลก่อน



ภาพที่ 3.1 Taxonomy

ระดับของข้อมูลจะถูกกำหนดโดย Taxonomy ดังแสดงที่ในภาพที่ 3.1 ซึ่งระดับของข้อมูลที่สนใจ สามารถที่จะปรับเปลี่ยนได้ตลอดเวลา เมื่อพบว่าการเกิดขึ้นของข้อมูลระดับสรุปนั้นไม่เพียงพอต่อการวิเคราะห์ เช่น ห้างสรรพสินค้าแห่งหนึ่งมีการขายสินค้าประเภท Outer wear เพียงอย่างเดียว โดยมีสินค้าในหมวด อยู่ 2 แบบคือ Jackets และ Ski Pants จุดประสงค์เดิมของร้าน คือ ค้นหาความสัมพันธ์ของสินค้า 2 ตัวนี้เท่านั้น แต่เวลาต่อมา ทางร้านได้มีนโยบาย ให้นำสินค้าประเภท Shirts มาขายด้วย จะเห็นได้ว่าวัตถุประสงค์ของทางร้านได้เปลี่ยนไป จาก Taxonomy จะเห็นได้ว่าสินค้าประเภท Outerwear และ Shirts อยู่ในระดับ(Level) เดียวกัน จึงต้องปรับเปลี่ยนมาพิจารณาในระดับที่สูงขึ้น เนื่องจากความถี่ในการซื้อสินค้า Jackets และ Ski Pants ในระดับล่างนั้น อาจเป็นส่วนน้อยเมื่อนำมาพิจารณาในระดับที่สูงขึ้น แต่เนื่องด้วยสินค้าทั้งสองชนิดนี้ เป็นสินค้าที่จัดอยู่ในประเภทเดียวกันคือประเภท Outerwear ดังนั้น เมื่อพิจารณาในระดับ Outerwear ความถี่ในการซื้อสินค้าในระดับนี้หมายถึงความถี่ในการซื้อ Jackets และ Ski Pants มารวมกัน(Grouping) ทำให้ค่าความถี่ในการซื้อ Outerwear มีค่าที่มาก และนำไปหาความสัมพันธ์ กับสินค้าประเภท Shirts ที่นำมาขายใหม่ได้ จะเห็นได้ว่าระดับของข้อมูลจะต้องสามารถปรับเปลี่ยนได้ตามจุดประสงค์

จาก Taxonomy จะเห็นได้ว่าข้อมูลสินค้ามีระดับที่หลากหลาย โดยกลุ่มของสินค้าเป็นตัวกำหนดระดับของ Taxonomy โดยปกติแล้ว การวิเคราะห์หาความสัมพันธ์ควรที่จะวิเคราะห์ประเภทของสินค้าที่อยู่ในระดับสูงก่อน เช่นในระดับ Clothes, Footwear เนื่องจากจำนวนความถี่ที่เกิดจากสินค้าประเภทนี้เกิดขึ้นมาจาก การรวมกันของความถี่ที่เกิดขึ้นของ

สินค้าแต่ละชนิดของสินค้าประเภทนี้ในระดับล่างลงมา ที่เกิดใน Transaction ของการซื้อสินค้า ซึ่งถ้าหากเราหาความสัมพันธ์ในระดับล่างก่อน เช่น Jackets และ Ski Pants เมื่อพิจารณาจาก Taxonomy จะพบว่าเป็นสินค้าในระดับล่างสุด และเมื่อนำจำนวนความถี่ที่เกิดจากการซื้อสินค้าไปเทียบกับความถี่ของสินค้าในระดับที่สูงขึ้น จะเห็นได้ว่ามีค่าความถี่ที่น้อยมาก จนอาจจะนำมาพิจารณาไม่ได้ อีกทั้งยังต้องเสียเวลาในการคำนวนหาความสัมพันธ์อีกด้วย

จะเห็นได้ว่า Taxonomy ช่วยในการจัดกลุ่มประเภทของสินค้าอยู่ระดับต่างๆ ทำให้รายการขายสินค้ามีความถี่ที่มากขึ้น ซึ่งช่วยแก้ปัญหาของสินค้าที่ไม่ค่อยเกิดรายการ การขายขึ้น

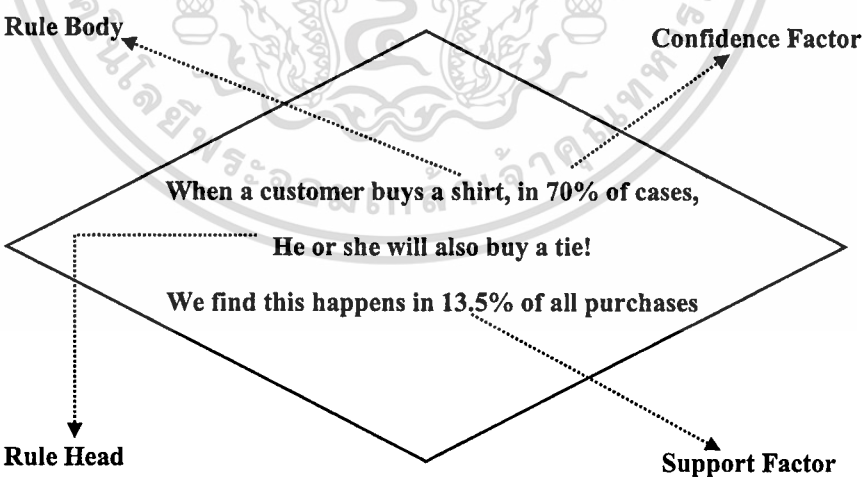
2) นำรายการมาสร้างเป็นกฎ

เมื่อได้ข้อมูลที่สนใจมาเรียบร้อยแล้ว ก็นำมาทำการ Combination ซึ่งรูปแบบของกฎความสัมพันธ์ จะอยู่ในลักษณะ " if X Then Y " หรือ " IF Condition1 THEN Condition2 " โดยที่ทั้ง X และ Y จะเกิดขึ้นพร้อมกันใน Transaction เดียวกัน

เรียก X หรือ Condition1 ว่า Rule Body หรือ Left-hand side, Antecedent

เรียก Y หรือ Condition2 ว่า Rule Head หรือ Right-hand side, Consequent

ตัวอย่างของกฎความสัมพันธ์แสดงได้ดังภาพที่ 3.2



ภาพที่ 3.2 รูปแบบความสัมพันธ์ของ Association Discovery

ผลที่ได้จากกฎจะมีความสัมคัญ ที่เป็นตัววัดหลักๆอยู่ 2 ตัว คือ

- Support Factor (ค่าความถี่ของเหตุการณ์ที่สนับสนุนให้เกิดความสัมพันธ์)
- Confidence Factor (ค่าความมั่นที่จะเกิดความสัมพันธ์ของเหตุการณ์นั้นขึ้น)

สามารถหาค่าต่างๆได้จากตัวอย่างการซื้อสินค้าต่อไปนี้

- รายการซื้อสินค้าทั้งหมด 500,000 รายการ
- เป็นรายการซื้อผ้าอ้อม 20,000 รายการ(4%ของรายการทั้งหมด)
- เป็นรายการซื้อเบียร์ 30,000 รายการ(6%ของรายการทั้งหมด)
- เป็นรายการซื้อทั้งผ้าอ้อมและเบียร์ 10,000 รายการ(2%ของรายการทั้งหมด)

- Support (Prevalence)

เป็นค่าของสัดส่วนของจำนวนเหตุการณ์ซื้อผ้าอ้อมกับเบียร์คู่กัน เทียบกับจำนวนเหตุการณ์ขายสินค้าทั้งหมด

$$\text{Support} = \frac{\text{จำนวนชุดข้อมูลที่มีรายการผ้าอ้อมและเบียร์คู่กัน}}{\text{จำนวนชุดข้อมูลทั้งหมด}} = \frac{10,000}{500,000} = 2\%$$

- Confidence (Predictability)

เป็นค่าสัดส่วนของจำนวนเหตุการณ์ซื้อผ้าอ้อมคู่กับเบียร์คู่กัน เทียบกับจำนวนของเหตุการณ์ซื้อผ้าอ้อมเพียงอย่างเดียว

$$\text{Confidence} = \frac{\text{จำนวนชุดข้อมูลที่มีรายการผ้าอ้อมและเบียร์คู่กัน}}{\text{จำนวนชุดข้อมูลที่มีรายการซื้อผ้าอ้อม}} = \frac{10,000}{200,000} = 50\%$$

ทั้งหมดสามารถนำมาแปลงเป็นกฎได้ว่า "50% ของลูกค้าที่ซื้อผ้าอ้อมมักจะซื้อเบียร์ด้วย" และในทางกลับกัน (Reverse Rule) จะได้กฎอีกกฎคือ "ลูกค้าที่ซื้อเบียร์ จะมีโอกาสที่จะซื้อผ้าอ้อมด้วย 33.33%" แต่จะสังเกตุได้ว่ากฎทั้งสองข้อนั้น มีค่า Support ที่เท่ากันคือ 2 %

โดยปกติ กฎที่น่าสนใจ คือกฎที่มีค่า Confidence ที่สูง เนื่องจากมีโอกาสที่จะเกิดขึ้นสูงตามด้วย และนอกจากตัววัดคือ Support และ Confidence สองตัวนี้แล้ว จากข้อมูลที่

เกิดขึ้นยังพบว่า มีตัววัดค่าของเหตุการณ์ที่เกิดขึ้นของแต่ละรายการของสินค้า เรียกว่า "Expected Confidence" และ "Lift"

- Expected Confidence

เป็นค่าสัดส่วนของจำนวนรายการซื้อแต่ละสินค้า เทียบกับจำนวนรายการขายสินค้าทั้งหมด จากตัวอย่างข้างต้น ค่า Expected Confidence ของ "การซื้อผ้าอ้อม" คือ 4% และ "การซื้อเบียร์" คือ 6%

สรุปได้ว่า 4% ของเหตุการณ์ที่เกิดขึ้นทั้งหมดมีการซื้อผ้าอ้อม และ 6% ของรายการทั้งหมดมีการซื้อเบียร์

- Lift

เป็นค่าที่แสดงความน่าเชื่อถือของความสัมพันธ์ระหว่างเหตุการณ์ ยิ่งค่า Lift มีค่าที่สูงเท่าไร ก็จะน่าเชื่อถือมากขึ้นเท่านั้น โดยหาได้จากสัดส่วนระหว่างค่า Confidence กับค่า Expected Confidence

จากตัวอย่าง Confidence มีค่า 50% ค่า Expected Confidence ของการซื้อเบียร์ คือ 6 %

$$\text{Lift} = \frac{\text{Confidence}}{\text{Expected Confidence}} = \frac{50\%}{6\%} = 8.33$$

ค่า Lift ที่ได้ จะแสดงถึงความสำคัญของความสัมพันธ์ หรือ เหตุการณ์ที่ว่ามีมากน้อยแค่ไหน จากข้อมูลที่ได้ ทำให้ได้กฎ "คนซื้อผ้าอ้อมมักจะซื้อเบียร์ด้วยในอัตรา 8%" ซึ่งถ้าหากว่าค่า Confidence มีค่ามากกว่า 50% จะทำให้ค่า Lift มีค่ามากขึ้นกว่า 8.33% ทำให้ความสัมพันธ์ระหว่างเหตุการณ์มีความน่าเชื่อถือมากยิ่งขึ้น

ในกรณีที่ ค่า Lift มีค่าติดลบ หรือน้อยกว่า 1 หมายถึงเหตุการณ์เหล่านั้นไม่มีทางที่จะเกิดขึ้นได้เลย และในกรณีที่ค่า Lift มากหรือน้อยเกินไป อาจพิจารณาได้ว่า กฎนั้นไม่เป็นความจริง ซึ่งปกตินักวิเคราะห์ข้อมูลมักจะสนใจกฎที่มีค่า Lift ในระดับที่สูงและระดับที่ต่ำ เนื่องจากทำให้หาความสัมพันธ์ได้ง่าย

3) จำกัดจำนวนที่เกิดขึ้นโดยเลือกเฉพาะชุดข้อมูลที่เป็นไปได้

ผลที่ได้จากการทำงานของ Association Rule จะทำให้กฎที่เกิดขึ้น มีจำนวนมาก จึงนำเทคนิคที่เรียกว่า "Pruning" มาช่วยในการกำจัดหรือคัดกฎที่ไม่น่าสนในออกเพื่อเป็นการลดกฎที่เกิดจากเหตุการณ์ที่มีโอกาสที่จะเกิดขึ้นน้อยออกไป ซึ่งสามารถทำได้โดยกำหนดค่า 2 ค่าคือ ค่า "Minimum Support" และ ค่า "Minimum Confidence" ซึ่งถ้าหากว่าค่า Support และ Confidence ต่ำกว่าค่า Minimum ให้กำจัดออกได้ โดยไม่ต้องนำมาพิจารณา ทำให้เวลาที่ใช้ในการคำนวณหากฎความสัมพันธ์นั้นสั้นลง

เมื่อทำการพิจารณากฎที่มีความสัมพันธ์ระหว่าง 4 Item ได้แก่

IF A,B, and C ,Then D

และมีรายการที่เกิดขึ้นทั้งหมด 1,000,000 รายการ ในการหากฎความสัมพันธ์ระหว่าง 4 Item ที่ได้ตั้งค่า "Minimum Support = 1%" นั้นหมายความว่า กฎที่จะนำมาพิจารณานั้น ทุกๆ Item ต้องมีการเกิดของเหตุการณ์ อย่างน้อยที่สุด 10,000 ครั้ง ของจำนวนรายการของเหตุการณ์ที่เกิดขึ้นทั้งหมด ถ้าหากว่า Item ใดมีการเกิดน้อยกว่า 10,000 ครั้ง ก็จะถูกตัดทิ้งไม่นำมาพิจารณา

ในการทำงานจริงนั้น การกำหนดค่า Minimum Support ขึ้นอยู่กับข้อมูลและสถานการณ์ ซึ่งสามารถปรับเปลี่ยนได้ในแต่ละระดับของการทำงานดังที่กล่าวไปตอนต้น เช่น ถ้าหากระบุค่า Minimum Support ให้มีค่าต่ำ จะทำให้รายการหรือเหตุการณ์ที่ไม่เกิดขึ้นบ่อยครั้งปรากฏออกมา หรือให้ทางกลับกัน ถ้าหากเรากำหนดค่า Minimum Support ให้มีค่ามากขึ้น ก็จะปรากฏเหตุการณ์ที่เกิดขึ้นเป็นประจำๆเท่านั้น

ข้อดีของ Association Rule

1) ผู้ใช้สามารถควบคุมจำนวนผลลัพธ์ได้ โดยระบุค่า Minimum Support และ Minimum Confidence

2) สามารถทำงานได้ดี กับข้อมูลขนาดใหญ่ ในขณะที่เทคนิคอื่นๆ จะมีปัญหา กับการทำงานกับข้อมูลปริมาณมากๆ

3) ในกรณีที่มีข้อมูลไม่สมบูรณ์ ก็สามารถทำการไมนิ่งกับข้อมูลบางส่วนได้

4) เทคนิคอื่นๆ อย่างเช่น Decision Trees, Neural Networks จะระบุขอบเขตของกลุ่มข้อมูล ทำให้มีการจำกัดข้อมูล ทำให้ข้อมูลที่เลือกมาทำการไมนิ่ง อาจจะไม่ใช่ตัวแทนที่แท้จริงของกลุ่มข้อมูล

5) สามารถจัดเก็บข้อมูลที่อยู่ในรูปแบบที่ต่างกันได้ ซึ่งจะไม่สูญเสียสารสนเทศในขณะที่เทคนิคอื่นๆ จะมีการจำกัดรูปแบบและความยาวของข้อมูล

6) มีการแสดงผลด้วยสัญลักษณ์ ทำให้ง่ายต่อการทำความเข้าใจกว่าผลลัพท์ที่ได้จากเทคนิคอื่นๆ

ข้อเสียของ Association Rule

1) ในการกำหนดค่า Minimum Support และ Minimum Confidence เพื่อจำกัดจำนวนของกฎที่ถูกสร้างขึ้นจำนวนมากนั้น อาจทำให้กฎที่ได้เกิดความผิดเพี้ยนจากความเป็นจริง เนื่องจากผู้ใช้กำหนดค่าสูงหรือต่ำจนเกินไป

2) กฎที่ได้มานั้น อาจเป็นกฎที่เกิดขึ้นบ่อยๆ และกฎที่เกิดขึ้นจากความบังเอิญ ทำให้มีความยากในการบอกความแตกต่างของกฎที่ได้มาค่อนข้างยาก

3) กฎที่ได้มาสามารถบอกได้เพียงแค่แนวโน้มที่จะเกิดขึ้นด้วยกัน ไม่ได้บอกเรื่องของความเป็นเหตุเป็นผลของกฎ

3.2 Sequential Pattern Discovery

ค้นหาพฤติกรรมของการซื้อสินค้าของลูกค้าในระยะยาว ใช้ระบุความเกี่ยวเนื่องของการซื้อสินค้า เช่น ซื้อสินค้าอย่างหนึ่ง และในระยะเวลาต่อมาจะมาซื้อสินค้าอีกอย่างหนึ่งตาม วิธีการคล้ายกับ Association Discovery ต่างกันที่วิธีการคำนวณหาค่า Support ที่ต่างออกไป โดยค่า Support ที่คำนวณนั้น ได้มาจากอัตราส่วนจำนวนของลูกค้าที่มีข้อมูลการซื้อสินค้าเป็นลำดับต่อจำนวนลูกค้าทั้งหมด

อัลกอริทึมในการทำงาน จะนับจำนวนความถี่ที่เกิดขึ้นของ Transaction ของลูกค้าซึ่งมีการเรียงลำดับของเหตุการณ์เอาไว้เรียบร้อยแล้ว โดยจะแสดงผลเฉพาะคู่ของเหตุการณ์ที่มีค่ามากกว่าค่า Minimum Support

ข้อดีและข้อเสียของเทคนิคนี้คล้ายกับ Association Discovery แต่มีจุดที่ต้องพิจารณาเพิ่มเติมคือ

1) จะต้องมี filed พิเศษ เพื่อใช้เป็นตัวแทนของลูกค้า ซึ่งโดยปกติแล้วในฐานข้อมูลการขายสินค้าทั่วไป มักจะไม่มีการเก็บรหัสของลูกค้าเอาไว้ใน Transaction

2) ต้องมีการเก็บเรียงลำดับของเหตุการณ์ที่เกิดขึ้นของลูกค้าแต่ละราย เพื่อที่จะทำให้เทคนิคนี้ทำงานได้ดี

3.3 Similar Time Sequence

เทคนิคนี้มักใช้สำหรับดูแนวโน้มของยอดขาย เพื่อพิจารณาการตัดสต๊อกของสินค้า โดยค้นหาความสัมพันธ์ที่มีความเกี่ยวเนื่องกันระหว่างกลุ่มของข้อมูล 2 กลุ่ม ซึ่งมีการขึ้นต่อกัน

ทางด้านเวลา ซึ่งมีรูปแบบการเคลื่อนไหวที่เหมือนกัน มีการแทนข้อมูลในแนวแกน X ด้วยค่าของเวลา และแทนตัวแปรที่เราสนใจ ในแกน Y ทำให้สามารถดูข้อมูลได้ว่า ณ.ช่วงเวลาแต่ละวัน หรือแต่ละสัปดาห์ ว่ามียอดขายของสินค้าใดที่มีค่าใกล้เคียงกัน อัลกอริทึมนี้จะมีการแสดงผลลัพท์ในรูปแบบของกราฟ

ข้อดีของอัลกอริทึมนี้คือ เห็นการเคลื่อนไหวของข้อมูล เช่นการเคลื่อนไหวของราคาสินค้า การเคลื่อนไหวของสต๊อก การเปลี่ยนแปลงของยอดขาย

บทที่ 4

Single–level Association Rules

ในบทนี้ จะหากฎความสัมพันธ์ของข้อมูลโดยใช้เทคนิคและอัลกอริทึมของ Apriori ซึ่งเป็นอัลกอริทึมพื้นฐาน สำหรับหา frequent itemset เพื่อนำมาสร้างกฎความสัมพันธ์ ในระดับ Single-level Association Rules

4.1 Apriori Algorithm

เทคนิคและการทำงานของอัลกอริทึมในการจัดการกับข้อมูลของ Apriori มีหลักการทำงานอยู่ 2 ขั้นตอน คือ

1.) การหาชุดของข้อมูล หรือ Itemset ที่เกิดขึ้นในรายการซื้อ ว่ามีค่าของความถี่ที่เกิดขึ้นจากการขายทั้งหมด มากกว่าหรือเท่ากับค่า Minimum Support ที่กำหนดเอาไว้

2.) นำ Frequent itemset มาสร้างเป็นกฎความสัมพันธ์ บนพื้นฐานที่ว่า A,B,C,D เป็น item และ ABCD,AB เป็น Frequent Itemset แล้วสามารถสร้างกฎ AB → CD โดยคำนวนค่า Confidence ได้จาก

$$\text{Confidence} = \frac{\text{Support(ABCD)}}{\text{Support(AB)}}$$

ซึ่งกฎที่ได้นั้นจะต้องมีค่า Confidence มากกว่าหรือเท่ากับค่า Minimum Confidence

4.1.1 สัญลักษณ์ที่ใช้ใน Apriori Algorithm

ตัวแปรที่จะต้องพิจารณา คือ

D	คือ Database ที่เก็บรายการของ Transaction ทั้งหมด <TID,Items>
TID	คือ ตัวเลขที่ใช้ระบุแต่ละรายการของ Transaction
Size	คือ จำนวนของ item ที่อยู่ในเซตของข้อมูล
K-itemset	คือ เซตของข้อมูลที่มีจำนวนสมาชิกจำนวน k ตัว

L_k คือ เซตของ Frequent K-itemset และทุกเซตจะต้องมีความถี่ในการเกิดมากกว่าหรือเท่ากับค่า Minimum Support ซึ่งสมาชิกในเซ็ตจะประกอบด้วย 2 ฟิลด์ คือ 1) Itemset 2) Support Count

C_k คือ เซตของ Candidate K-itemset ซึ่งเป็นเซตที่ถูกเลือกมาจาก L_k สมาชิกในเซตจะประกอบด้วย 2 ฟิลด์ คือ 1) Itemset 2) Support Count

4.2 การทำงานของ Apriori Algorithm

การทำงานของอัลกอริทึม เริ่มจากการหา Frequent 1-itemset หรือ L_1 โดยการนับค่า Support ที่เกิดขึ้นใน Database ของแต่ละ itemset โดยเลือกเฉพาะ itemset ที่มีค่า Support มากกว่าค่า Minimum Support จากนั้นจึงเข้าสู่ K-Loop การทำงานขั้นตอนนี้ของอัลกอริทึมจะทำการสร้าง C_k ขึ้น โดย apriori-gen ซึ่งมีขั้นตอนการทำงานดังนี้

ขั้นที่ 1: Join Step

L_{k-1} จะนำไปใช้สร้าง Candidate itemset หรือ C_k โดยการใช้ apriori candidate generation โดยการนำแต่ละ itemset ของ L_{k-1} มาทำการ Join กัน

ขั้นที่ 2: Prune Step

นำ C_k ที่เกิดจากการ Join Step มาทำการกำจัด Candidate itemset ของ k-1 subset ของมันที่ไม่ได้อยู่ใน L_{k-1}

หลังจากอัลกอริทึมนับค่า Support ที่เกิดขึ้นใน Database แต่ละ k-1 itemset ใน C_k โดยเลือกมาเฉพาะ k-1 itemset ที่มีค่ามากกว่าค่า Minimum Support มาสร้างเป็น L_k เรียบร้อยแล้ว ก็จะวนลูปโดยทำการเพิ่มค่า k ขึ้น 1 ค่า และทำตามขั้นตอนเดิมใน k-Loop ต่อไปจนไม่สามารถหา frequent itemset หรือ L_k ได้ต่ออีกแล้ว จึงสิ้นสุดการ

ภาพที่ 4.1 เป็นข้อมูลของ Transaction ของการซื้อสินค้าที่เกิดขึ้นใน Database ซึ่งสามารถใช้ Apriori Algorithm หา frequent itemset ใน D ได้ตามขั้นตอนดังต่อไปนี้

1.ขั้นแรกเพียงแค่นำข้อมูลจาก Database มานับจำนวนของ Item แต่ละรายการจาก Transaction ทั้งหมด จะได้ C_1 ออกมา

2.ให้พิจารณาค่าความถี่ หรือ ค่าSupport ที่เกิดขึ้นของแต่ละ itemset ถ้ามีค่า Support ต่ำกว่าค่า Minimum Support ซึ่งมีการกำหนดไว้เท่ากับ 2 ก็จะถูกกำจัดออก และนำค่าที่มากกว่า มาสร้างเป็น L_1

3. ค้นหา itemset ของ L_2 โดยขั้นตอนการ Join Step ของ apriori-gen โดยทำการ Join ระหว่าง L_1 กับ L_1 และผ่านขั้นตอน Prune Step ซึ่งจะได้ C_2 ออกมา

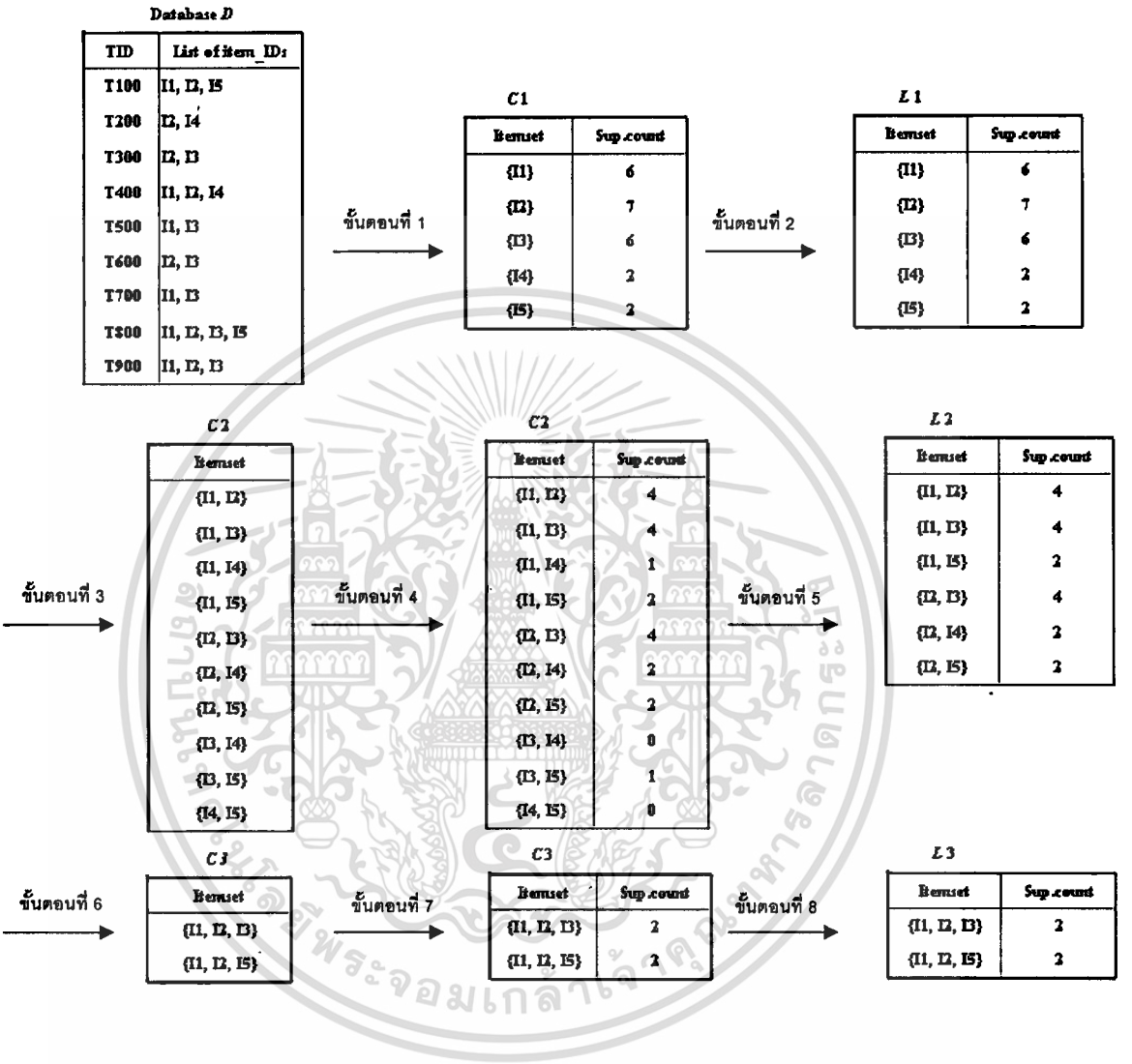
4.จากนั้นกำจัดบาง Candidate ใน C_2 ที่มีค่า Support น้อยกว่าค่า Minimum Support ออก ซึ่งจะได้ L_2 ออกมา

5. สร้าง L_3 ขึ้นมา โดยนำ L_2 มา Join เข้าด้วยกัน โดยทำการพิจารณาจากคุณสมบัติของ Apriori ที่ว่า subset ทั้งหมดของแต่ละ itemset ของ C_k นั้นจะต้องปรากฏอยู่ใน L_2 ทั้งหมด เช่น subset ของ C_3 ทั้งหมดที่ได้มาจากการ Join ต้องปรากฏอยู่ใน itemset ในตาราง L_2 ด้วย แต่ถ้าไม่ปรากฏให้ทำการกำจัด itemset นั้นออก

6. นับค่า Support ของแต่ละ itemset ใน C_3 จาก D

7. เพราะฉะนั้น L_3 จะได้จาก itemset ทั้งหมดของ C_3 ที่มีค่า Support มากกว่าค่า Minimum Support ที่กำหนดไว้

8. หลังจากนั้น อัลกอริทึมจะทำการวนลูป โดยทำการ Join L_3 กับ L_3 เพื่อสร้างเป็น 4 – itemset โดย C_4 จะได้ผลจากการ Join คือ {{ I1,I2,I3,I5}} ในขั้นตอนการ Prune Step เมื่อตรวจสอบพบว่า subset {{I2,I3,I5}} ไม่ปรากฏใน L_3 จึงกำจัด itemset ที่ได้จากการ Join ออก ทำให้ $C_4 = 0$ ซึ่งตรงกับเงื่อนไขที่ให้ออกจาก k-Loop ของอัลกอริทึม และให้ทำเงื่อนไขสุดท้าย คือให้แสดงผลลัพท์ของทุก frequent itemset หรือทุก L_k นั้นเอง



ภาพที่ 4.1 Generation of candidate

L_1 = {frequent 1-itemsets}
For ($k = 2$; $L_{k-1} \neq \phi$; $k++$) do begin
    C_k = apriori-gen(L_{k-1}, min_sup);
    forall transaction $t \in D$ do begin
        C_t = subset(C_k, t)
        forall candidates $c \in C_t$ do
            $c.count++$;
        end
    end
    $L_k = \{ c \in C_k | c.count \geq minsup \}$
end
$Answer = \bigcup_k L_k$;

ภาพที่ 4.2 Apriori Algorithm ซึ่งมี

Input : Database, D และค่า Minimum Support

Output : L, frequent item ใน D

insert into C_k
select $p.item_1, p.item_2, p.item_{k-1}, q.item_{k-1}$
from $L_{k-1}p, L_{k-1}q$
where $p.item_1 = q.item_1, \ldots, p.item_{k-2} = q.item_{k-2}, p.item_{k-1} < q.item_{k-1}$

ภาพที่ 4.3 Apriori – gen Algorithm (Join-Step)

forall *itemsets* $c \in C_k$ do
    forall *(k-1)-subsets s of c* do
        if ($s \notin L_{k-1}$) then
            delete *c from* C_k

ภาพที่ 4.4 Apriori – gen Algorithm (Prune Step)

4.3 การสร้างกฎความสัมพันธ์

การสร้างกฎความสัมพันธ์ . สามารถทำได้โดยนำ Frequent itemset ที่ได้จาก Apriori Algorithm ในหัวข้อ 4.1.2 โดยนำค่า frequent itemset ตั้งแต่ L_2 มาคำนวณหา subset โดยนำ subset เหล่านั้นมาสร้างเป็นกฎความสัมพันธ์ สามารถสร้างเป็นกฎความสัมพันธ์ได้จากอัลกอริทึมในภาพที่ 5.6

บทที่ 5

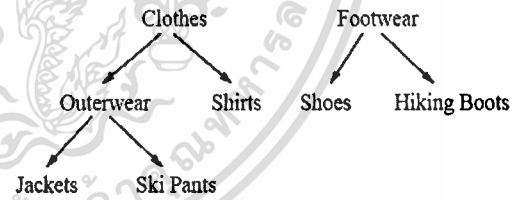
Multi–level Association Rules

การค้นหาความสัมพันธ์ที่เกิดขึ้นของ itemในระดับ Low level นั้นเป็นสิ่งที่ยาก เนื่องจากปกติ ความถี่ (Support) ที่เกิดขึ้นของแต่ละ item ในระดับล่างนั้น จะเกิดขึ้นเพียงเล็กน้อยเท่านั้นเมื่อนำไปเปรียบเทียบกับค่า Support ที่เกิดขึ้นในระดับบน ซึ่งค่า Support ในระดับบนจะได้มาจากการรวมกันของค่า Support ในระดับล่าง จะเห็นได้ว่าการหาความสัมพันธ์แบบ Single-level Association Rule นั้นใช้วิธีหากฎความสัมพันธ์โดยนำค่า Support ในระดับล่างของแต่ละ item มารวมกัน(Grouping) ทำให้สามารถค้นหาความสัมพันธ์ของ item เหล่านี้ได้ เนื่องจากมีค่า Support มากกว่าค่า Minimum Support ที่กำหนดไว้ แต่จะไม่มีทางทราบได้เลยว่าความสัมพันธ์ของแต่ละ item ในระดับ Low level นั้นเกิดความสัมพันธ์ใดขึ้นบ้าง

5.1 Generalized Association Rules

โดยปกติข้อมูลแต่ละ Transaction ที่เก็บไว้ใน Database (ภาพที่5.1) จะเกิดขึ้นจากเซตของ item ซึ่งมีการแบ่งหมวดหมู่ (Taxonomy) เอาไว้เป็นลำดับชั้น (hierarchy) ดังแสดงในภาพที่ 5.2

Transaction	Items Bought
100	Shirt
200	Jacket, Hiking Boots
300	Ski Pants, Hiking Boots
400	Shoes
500	Shoes
600	Jacket



ภาพที่ 5.1 Database *D*

ภาพที่ 5.2 Taxonomy

จากฐานข้อมูล *D* เมื่อนำไปหาความสัมพันธ์ โดย Apriori Algorithm ซึ่งมีค่า Minimum Support = 2 จะเห็นได้ว่าไม่สามารถสามารถหาความสัมพันธ์ของ item ได้ เนื่องจาก Transactionในฐานข้อมูล *D* นั้นมีเพียง Transaction ที่ 100 และ 200 ที่เกิดความสัมพันธ์ระหว่าง 2 item ขึ้นคือ {Jacket, Hiking Boots} และ {Ski Pants, Hiking Boots} ตามลำดับ ซึ่งแต่ละ itemset จะมีค่า Support เพียงแค่ 1 เท่านั้น จึงไม่ถึงค่า Minimum Support ที่กำหนดเอาไว้ ทำให้กฎเหล่านี้ถูกกำจัดออกไป แต่เมื่อพิจารณาจาก Taxonomy จะเห็นได้ว่าทั้ง Jackets และ Ski Pants ต่างก็เป็น item ชนิด

เดียวกันคือ Outerwear กล่าวคือ ทั้ง Jackets และ Ski Pants ต่างก็เป็นทายาท(descendant)ของ Outerwear และ Outerwear ก็เป็น Parent ของทั้ง Jackets และ Ski Pants และกล่าวต่อไปได้อีกว่าใน เมื่อ Outerwear มี Clothes เป็น Parent เพราะฉะนั้นทั้ง Jackets และ Ski Pants ก็จะต้องมี Clothes เป็นบรรพบุรุษ(ancestor) ของ item ทั้ง 2 ด้วย เป็นการสืบทอด จึงทำให้กฎที่บอกว่า "เมื่อลูกค้าซื้อ Jacket แล้วจะซื้อ Hiking Boot ด้วย" (Jacket → Hiking Boots) และ กฎที่ว่า "เมื่อลูกค้าซื้อ Outerwear แล้วจะซื้อ Hiking Boots ด้วย" (Outerwear → Hiking Boots) มีความหมายเดียวกัน จึงสามารถสรุปได้ว่า Transaction ที่ 100,200 นั้น เกิดความสัมพันธ์ (Outerwear → Hiking Boots) ในระดับบนที่เหมือนกัน เมื่อพิจารณาในระดับบนจะเห็นได้ว่าค่า Support ในฐานข้อมูล *D* มีค่าเท่ากับ 2 ซึ่งเท่ากับค่า Minimum Support จึงทำให้สามารถหาความสัมพันธ์ในฐานข้อมูล *D* นี้ได้

5.2 Cumulate Algorithm

ในการค้นหากฎความสัมพันธ์แบบหลายลำดับชั้นนั้นค่า Support และ Confidence จำเป็นจะต้องมีค่าที่มากกว่าหรือเท่ากับค่า Minimum Support และ Minimum Confidence ที่ผู้ใช้งานได้กำหนดไว้ ซึ่งการทำงานของ Cumulate Algorithm มีการทำงานอยู่บนพื้นฐานของ Apriori Algorithm โดยเพิ่มขั้นตอนการกำจัด itemset ที่ซ้ำซ้อน (redundant) ซึ่งเกิดจากการทำงานของอัลกอริทึม เนื่องมากจากมีการเพิ่ม ancestor ของแต่ละ item เข้าไปในทุก Transaction ของฐานข้อมูล โดย Cumulate Algorithm นี้จะมีการค้นหากฎความสัมพันธ์ของข้อมูลแบบ Multi-level ในรูปแบบของ Generalized Association Rules

5.2.1 Algorithms Basic

ในการค้นหาความสัมพันธ์ของ Generalized Association Rules นั้น จะใช้อัลกอริทึมที่มีการทำงานพื้นฐานอยู่บน Apriori Algorithm ต่างกันตรงที่ เพิ่มเติมขั้นตอนการเพิ่ม ancestor หรือหมวดหมู่ของแต่ละ item ที่กำหนดไว้ใน Texonomy เข้าไปในแต่ละ Transaction ในฐานข้อมูล *D* ซึ่งจะทำให้ตารางฐานข้อมูล *D* กลายเป็นตาราง T*

การทำงานในส่วนแรก(1^{st}-Pass) นั้นเพียงแค่หา frequent 1-itemset หรือ L_1 โดยทำการนับความถี่(Support)ที่เกิดขึ้นของแต่ละ item ใน T* ซึ่งทุก item ใน itemset นั้นสามารถที่จะนำมาจากทุกลำดับชั้น(level) ของ Taxonomy หรือ interior node ได้ทั้งหมด ไม่จำเป็นที่จะต้องนำมาจากระดับล่าง(Low level) ระดับเดียว

ในส่วนถัดมา จะเป็นการทำงานของ K-Pass ซึ่งประกอบด้วย 2 ขั้นตอนการทำงานคือ

1. ใช้ Apriori Candidate Generations สร้าง frequent itemset L_{k-1} ใน(k-1)th Pass ซึ่งได้มาจาก Candidate itemsets C_k โดยที่แต่ละ item ใน Candidate itemsets นั้นสามารถที่จะนำมาได้ จากทุกลำดับชั้นใน Taxonomy
2. นับค่า Support ของแต่ละ Candidate ใน C_k โดยนับจากตาราง T* จากนั้นให้ทำการกำจัด บาง Candidate ออก (Prune) จึงได้ L_k ออกมา

$L_1 = \{frequent\ 1\text{-}itemsets\}$
For $(k = 2; L_{k-1} \neq \phi; k++)$ do begin
  C_k = apriori-gen$(L_{k-1}, \min_sup)$;
  forall transaction $t \in D$ do begin
    t = add-ancestor(t, T)
    C_t = subset(C_k, t)
    forall candidates $c \in C_t$ do
      $c.count++$;
  end
  $L_k = \{c \in C_k | c.count \geq minsup\}$
end
$Answer = \bigcup_k L_k$;

ภาพที่ 5.3 Algorithm Basic

5.2.2 Algorithm Cumulate

จาก Basic Algorithm นั้นสามารถพัฒนาโดยการเพิ่ม Optimization เข้าไปในขั้นตอนการทำงานของ Algorithm ทำให้ใช้เวลาในการทำงานสั้นลง และเรียก Algorithm ที่มีประสิทธิภาพนี้ว่า "Cumulate Algorithm" โดย Optimization ที่ว่ามีอยู่ 3 ขั้นตอนดังนี้

1) Filtering the ancestor added to transaction

ในการทำงานของ Cumulate เพียงแค่ทำการเพิ่ม ancestor ของ item ที่ปรากฏอยู่ใน Candidate itemset ของ C_k ที่กำลังนับอยู่ใน Pass ปัจจุบัน แทนการเพิ่มทุก ancestor ของแต่

ละ item ใน Transaction t และถ้า Original item ไม่ได้อยู่ใน itemset ของ Candidate itemset (C_k) ก็ให้กำจัดออกจาก Transaction

2) Pre-computing ancestors

ทำการ Pre-computing หา ancestor ของแต่ละ item โดยทำการสำรวจ Taxonomy graph หลังจากนั้นให้ทำการกำจัด ancestor ที่ไม่ได้อยู่ใน Candidate itemset ออกไป

3) Pruning itemsets containing an item and it ancestor

เมื่อพิจารณาค่า Support ของ itemset X ซึ่งประกอบด้วย item x และ ancestor ของ item x($\hat{x}$) จะมีค่า Support เท่ากับ itemset ที่ประกอบด้วย item x - $\hat{x}$ เนื่องจาก x - $\hat{x}$ ⊂ x

ถ้า L_k ไม่ปรากฏ itemset ที่ประกอบด้วย item และ ancestor ของมัน itemset ของ Candidate C_{k+1} ที่กำเนิดโดย Candidate Generation Procedure จะไม่ปรากฏ itemset ที่ประกอบด้วย item และ ancestor ของมัน

Compute T^*, the set of ancestors of each item, *from* T // Optimiz.2
$L_1 = \{frequent\ 1\text{-}itemsets\}$
For $(k = 2; L_{k-1} \neq \phi; k++)$ do begin
    C_k = apriori-gen$(L_{k-1}, \text{min_sup})$;
    if (k = 2) then prune(C_2) // Optimiz 3
    T^* = remove-unnecessary(T^*, C_k) // Optimiz 1
    forall transaction $t \in D$ do begin
        t = add-ancestor(t, T^*)
        C_t = subset(C_k, t)
        forall candidates $c \in C_t$ do
            $c.count++$;
        end
    end
    $L_k = \{c \in C_k | c.count \geq minsup\}$
end
$Answer = \bigcup_k L_k$;

ภาพที่ 5.4 Cumulate Algorithm

TID	Item Bought
T100	Shirt
T200	Jacket, Hiking, Boots
T300	Ski Pants, Hiking Boots
T400	Shoes
T500	Shoes
T600	Jacket

ตารางที่ 5.1 ตาราง database *D*

Item	Ancestors Set
Shirt	{Clothes}
Jacket	{Outerwear, Clothes}
Hiking Boots	{Footwear}
Ski Pants	{Outerwear, Clothes}
Shoes	{Footwear}

ตารางที่ 5.2 ตาราง Taxonomy

TID	Item Bought
T100	Shirt, *Clothes*
T200	Jacket, Hiking, Boots, *Outerwear, Clothe, Footwear*
T300	Ski Pants, Hiking Boots, *Outerwear, Clothe, Footwear*
T400	Shoes, *Footwear*
T500	Shoes, *Footwear*
T600	Shoes, *Outerwear, Clothe*

ตารางที่ 5.3 ตาราง T*

C1

Itemset	Sup.count
{Shirt}	1
{Jacket}	2
{Hiking Boots}	2
{Ski Pants}	1
{Shoes}	2
{Clothes}	4
{Outerwear}	3
{Footwear}	4

ตารางที่ 5.4 ตาราง C1

L 1

Itemset	Sup.count
{Jacket}	2
{Hiking Boots}	2
{Shoes}	2
{Clothes}	4
{Outerwear}	3
{Footwear}	4

ตารางที่ 5.5 ตาราง L1

C 2

Itemset
{Jacket, Hiking Boots}
{Jacket, Shoes}
{Jacket, Footwear}
{Hiking Boots, Clothes}
{Hiking Boots, Outerwear}
{Shoes, Clothes}
{Shoes, Outerwear}
{Clothes, Footwear}
{Outerwear, Footwear}

ตารางที่ 5.6 ตาราง C2

C 2

Itemset	Support
{Jacket, Hiking Boots}	1
{Jacket, Shoes}	0
{Jacket, Footwear}	1
{Hiking Boots, Clothes}	2
{Hiking Boots, Outerwear}	2
{Shoes, Clothes}	0
{Shoes, Outerwear}	0
{Clothes, Footwear}	2
{Outerwear, Footwear}	2

ตารางที่ 5.7 ตาราง C2

L_2

Itemset	Support
{Hiking Boots, Clothes}	2
{Hiking Boots, Outerwear}	2
{Clothes, Footwear}	2
{Outerwear, Footwear}	2

ตารางที่ 5.8 ตาราง L2

$U_k L_k$

Itemset	Sup.count
{Jacket}	2
{Hiking Boots}	2
{Shoes}	2
{Clothes}	4
{Outerwear}	3
{Footwear}	4
{Hiking Boots, Clothes}	2
{Hiking Boots, Outerwear}	2
{Clothes, Footwear}	2
{Outerwear, Footwear}	2

ตารางที่ 5.9 ตาราง Uk *L*k

Rule	Support	Conf.
Hiking Boots --> Clothes	33%	100%
Hiking Boots --> Outerwear	33%	100%
Clothes --> Footwear	33%	50%
Outerwear --> Footwear	33%	66%
Clothes --> Hiking Boots	33%	50%
Outerwear --> Hiking Boots	33%	66%
Footwear -->Clothes	33%	50%
Footwear -->Outerwear	33%	50%

ตารางที่ 5.10 ตารางกฎความสัมพันธ์ที่ได้จากตาราง Uk *Lk*

Rule	Support	Conf.
Hiking Boots --> Clothes	33%	100%
Hiking Boots --> Outerwear	33%	100%
Outerwear --> Footwear	33%	66%
Outerwear --> Hiking Boots	33%	66%

ตารางที่ 5.11 ตารางกฎความสัมพันธ์ที่ได้จาก Uk Lk ที่ผ่านค่า Min_Sup

5.2.3 การทำงานของ Cumulate Algorithm

ตารางที่ 5.1 ถึงตารางที่ 5.9 เป็นการทำงานของ Cumulate Algorithm ในการหาค้นหากฎความสัมพันธ์ของสินค้าจาก Transaction ของการซื้อสินค้าที่จัดเก็บไว้ในฐานข้อมูล ซึ่งมีการทำงานอยู่บนพื้นฐานของ Apriori Algorithm สามารถที่จะหา frequent itemset และความสัมพันธ์ของสินค้าใน D ได้ตามขั้นตอนดังต่อไปนี้

1.ขั้นแรกเพียงแค่นำข้อมูลจากตาราง Database D มาพิจารณาว่า แต่ละ item มี ancestors หรืออยู่ในหมวดหมู่อะไรบ้าง โดยพิจารณาจากตาราง Taxonomy ของ item

2.ทำการเพิ่ม ancestors ของแต่ละ item เข้าไปในตาราง D โดยมีเงื่อนไขที่ว่า แต่ละ Transaction นั้นสามารถที่จะมี ancestors ชนิดเดียวกันได้เพียงจำนวนหนึ่งตัวเท่านั้น ถ้ามี ancestors เกิดขึ้นซ้ำกันจำนวนหลายตัว ให้ทำการกำจัด ancestors ที่เหลือออก

3.ให้พิจารณาค่าความถี่ หรือ ค่าSupport ที่เกิดขึ้นของแต่ละ itemset ของตาราง T* ซึ่งขั้นตอนนี้จะเหมือน Apriori Algorithm แตกต่างกันเพียงที่ Cumulate Algorithm นั้นยอมให้ ancestor ของแต่ละ item มารวมอยู่ในแต่ละ itemset ในตาราง C1 ได้

4. ถ้า itemset ใด ในตาราง C1 นั้นมีค่า Support ที่ต่ำกว่าค่า Minimum Support ซึ่งกำหนดไว้เท่ากับ 2 Itemset นั้นจะถูกกำจัดออกไป โดยจะนำเฉพาะ item set ที่มีค่า Support มากกว่าค่า Minimum Support มาสร้างเป็นตาราง L_1

5. เข้าสู่ Loop การสร้างตาราง L_K ซึ่งมีการกำหนดค่าเริ่มต้นคือ k=2 โดยมีเงื่อนไขกำหนดไว้ว่า ถ้าหากว่า $L_K = L_2$ จะต้องเข้าสู่ขั้นตอนการ Join Step ซึ่งมีการหลักการทำงานอยู่บน apriori-gen ของ Apriori Algorithm โดยจะทำการ Join ระหว่าง L_k กับ L_k ก็คือ L_1 กับ L_1 และผ่านขั้นตอน Prune Step ซึ่งจะได้ C_k หรือ C_2 ออกมา และเนื่องจากค่าเริ่มต้นคือ k=2 จึงเข้าขั้นตอนการทำงานของ Cumulate Algorithm โดยจะทำการ Delete บาง candidate ใน C_2 ที่ประกอบด้วย itemset และ ancestors ของตัว itemset เองออก จาก C_2 โดยถ้าหาก $k > 2$ ก็ไม่จำเป็นที่จะต้องทำซ้ำในขั้นตอนนี้อีก เนื่องจากหลังจากที่ทำขั้นตอนที่ k=2 แล้วจะไม่มี itemset และ ancestors ของตัว itemset เอง ปรากฏอยู่ใน candidate ของ C_k ที่ $k > 2$ อีกต่อไป หลังจากนั้นให้ทำการ Delete บาง ancestor ใน T* ที่ไม่ได้อยู่ใน itemset ของ C_2 ออก

6.ทำการนับค่า Support ของแต่ละ itemset ของ C_2 โดยมีวิธีการทำงานเหมือนกับ Apriori Algorithm โดยค่า Support ที่ได้นั้น เกิดจากจำนวน item set ที่ปรากฏอยู่ในตาราง T*

7.จากนั้นให้กำจัดบาง Candidate ใน C_2 ที่มีค่า Support น้อยกว่าค่า Minimum Support ออก ซึ่งจะได้ L_2 ออกมา

8.ทำการวนลูปไปที่ K-Loop ของ C_k ใหม่อีกครั้ง ซึ่ง k จะมีค่าเพิ่มขึ้นหนึ่งค่า ซึ่งเท่ากับ 3 และทำตามขั้นตอนเดิม และทำการสร้างตาราง L_3 ขึ้นมาโดยนำแต่ละ item set ของตาราง L_2 มา Join เข้าด้วยกัน หลังจากนั้นให้พิจารณาจากพื้นฐานคุณสมบัติของ Apriori ที่ว่า subset ทั้งหมดของแต่ละ itemset ของ C_k นั้นจะต้องปรากฏอยู่ใน L_2 ทั้งหมด เช่น subset ของ C_3 ทั้งหมดที่ได้มาจากการ Join ต้องปรากฏอยู่ใน itemset ในตาราง L_2 ทั้งหมดด้วย ซึ่งถ้าหากไม่ปรากฏอยู่ใน L_2 ทั้งหมดแล้วละก็ให้ทำการกำจัด itemset นั้นออกไป หลังจากได้ C_3 แล้ว ก็ให้ทำการวนลูปต่อไปจนกว่า $C_k = 0$ ซึ่งจะทำให้ $L_k = 0$ ด้วย ซึ่งจะตรงกับเงื่อนไขที่จะต้องออกจาก k-Loop ของอัลกอริทึม และให้ไปทำเงื่อนไขสุดท้าย คือให้แสดงผลลัพท์ของทุก frequent itemset หรือทุก L_k นั้นเอง

5.2.4 การสร้างกฎความสัมพันธ์

การสร้างกฎความสัมพันธ์ สามารถทำได้โดยนำทุกๆ itemset ที่ได้จากการทำงานของ Cumulate Algorithm ในหัวข้อ 5.2.3 โดยนำทุกๆ itemset ตั้งแต่ตาราง L_2 มาคำนวณหาค่า Support และค่า Confidance และนำมาแสดงผลเป็นกฎความสัมพันธ์ ในตารางที่ 5.11 จะเห็นได้ว่าผลที่ได้ออกมานั้นจะมีการแสดงความสัมพันธ์ของแต่ละ item ในลักษณะความสัมพันธ์แบบหลายลำดับชั้น ซึ่งจะเห็นได้ว่ากฎความสัมพันธ์บางกฎที่ไม่สามารถปรากฏใน การค้นหาความสัมพันธ์แบบ Single-level ได้นั้น จะสามารถปรากฏออกมาให้ทำการวิเคราะห์หาความสัมพันธ์ได้

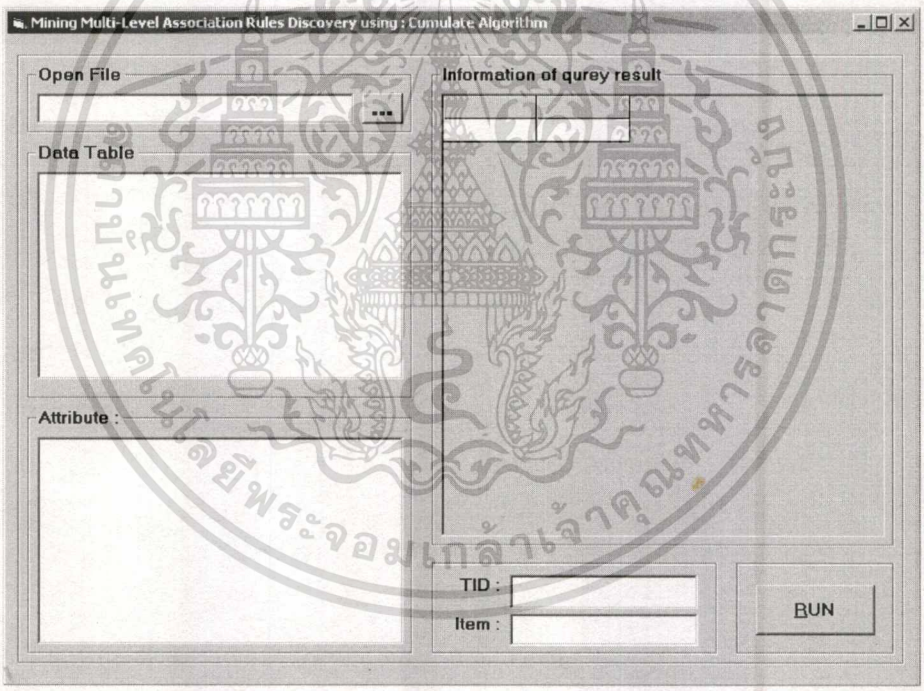
บทที่ 6

การประยุกต์ใช้ดาต้าไมนิ่งเพื่อวิเคราะห์หาความสัมพันธ์แบบหลายลำดับชั้นของข้อมูล

เพื่อให้การศึกษาถึงการนำ Cumulate Algorithm มาใช้ค้นหาความสัมพันธ์แบบ Multi-level Association Rules ของ Data Mining บรรลุตามวัตถุประสงค์ที่กำหนด จึงได้พัฒนาโปรแกรมที่ใช้ค้นหากฎความสัมพันธ์ของ ข้อมูล ซึ่งจะใช้ Cumulate Algorithm ดังนั้นในบทนี้จะกล่าวถึงวิธีการใช้งานโปรแกรมที่พัฒนาขึ้น

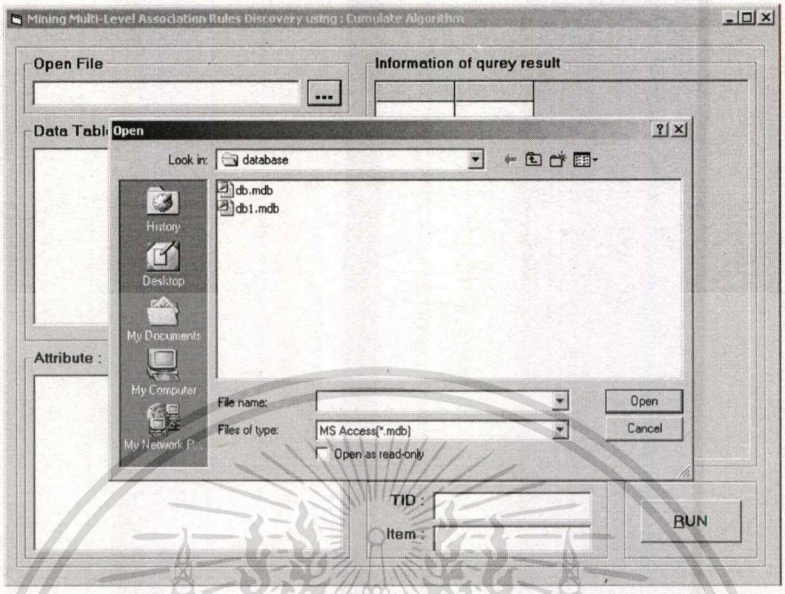
6.1 การติดต่อกับฐานข้อมูลที่ต้องการวิเคราะห์

เมื่อเข้าสู่โปรแกรมจะปรากฏหน้าจอหลัก ดังภาพที่ 6.1



ภาพที่ 6.1 หน้าจอหลักของโปรแกรม

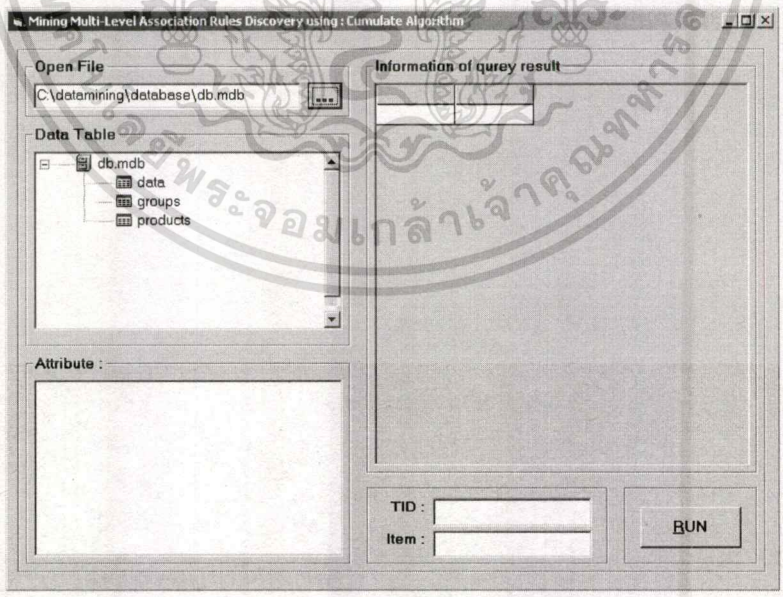
ทำการ Open File ฐานข้อมูล Microsoft Access ที่ต้องการติดต่อ ดังภาพที่ 6.2



ภาพที่ 6.2 หน้าจอแสดงการเลือกฐานข้อมูลที่ต้องการ

6.2 การเลือกตารางที่ต้องการนำมาวิเคราะห์

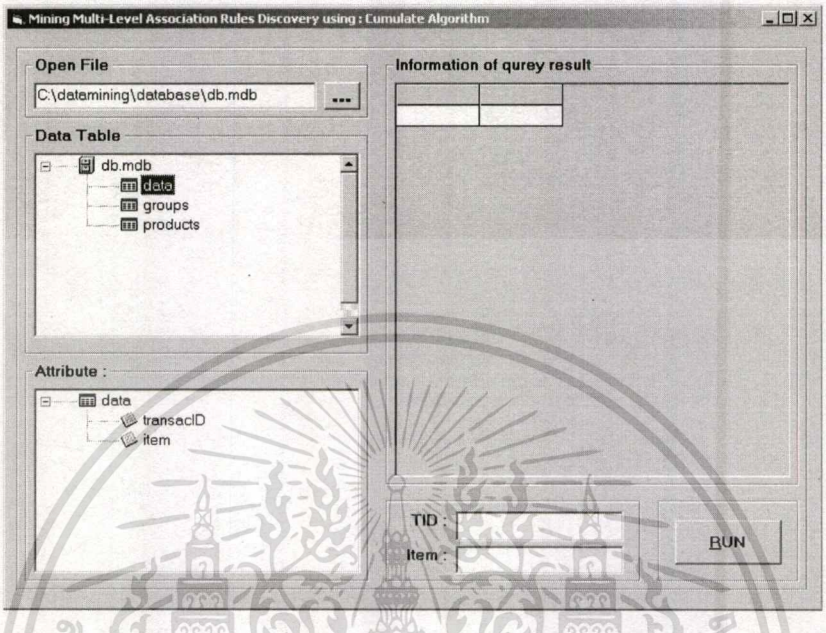
เมื่อติดต่อกับฐานข้อมูลเรียบร้อยแล้ว ชื่อตารางต่างๆ ในฐานข้อมูลจะปรากฏออกมาดังภาพที่ 6.3



ภาพที่ 6.3 หน้าจอแสดง Table ในฐานข้อมูล

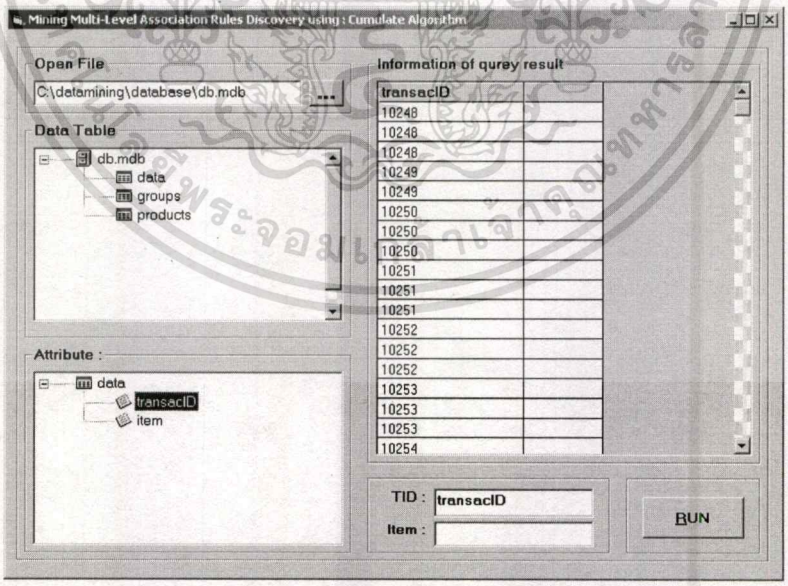
6.3 การเลือก Attribute ที่จะนำมาวิเคราะห์

เลือกตารางที่ต้องการ จะปรากฏชื่อของ Attribute ต่างๆในตารางนั้นขึ้นมา ดังภาพที่ 6.4

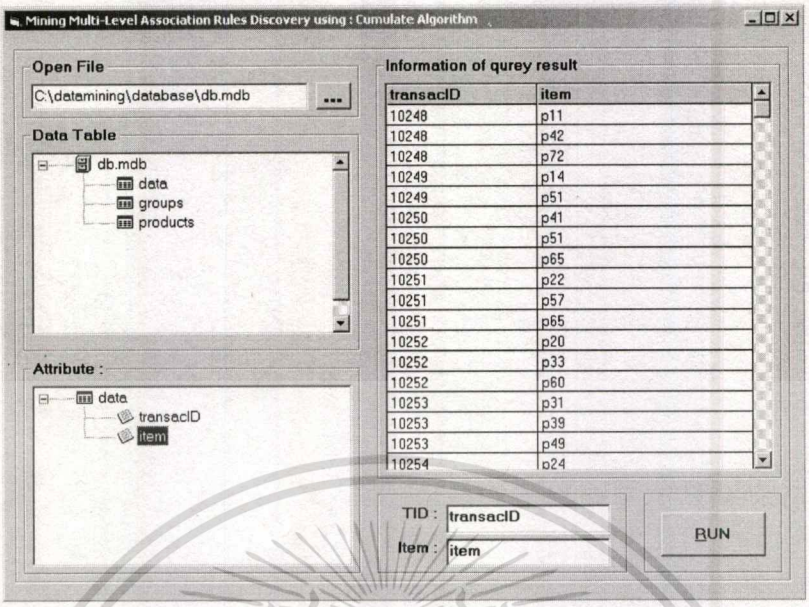


ภาพที่ 6.4 หน้าจอแสดง Attribute ที่อยู่ในตาราง

จากนั้นทำการเลือก Attribute ที่ต้องการจะนำมาวิเคราะห์หาความสัมพันธ์ ดังภาพที่ 6.5 และ 6.6



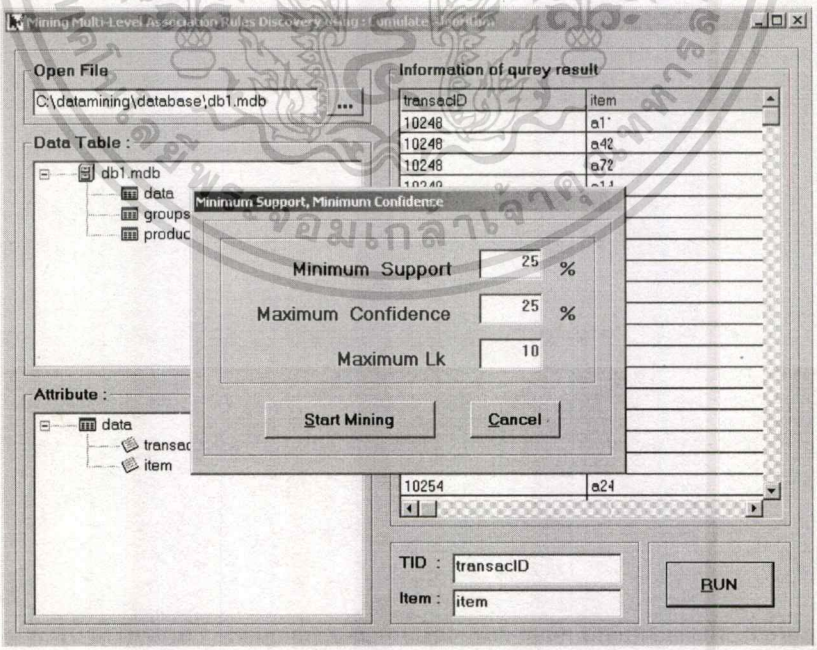
ภาพที่ 6.5 หน้าจอแสดงข้อมูลที่อยู่ใน Attribute TID



ภาพที่ 6.6 หน้าจอแสดงค่าต่างๆที่อยู่ใน Attribute Item

6.4 การค้นหากฎความสัมพันธ์

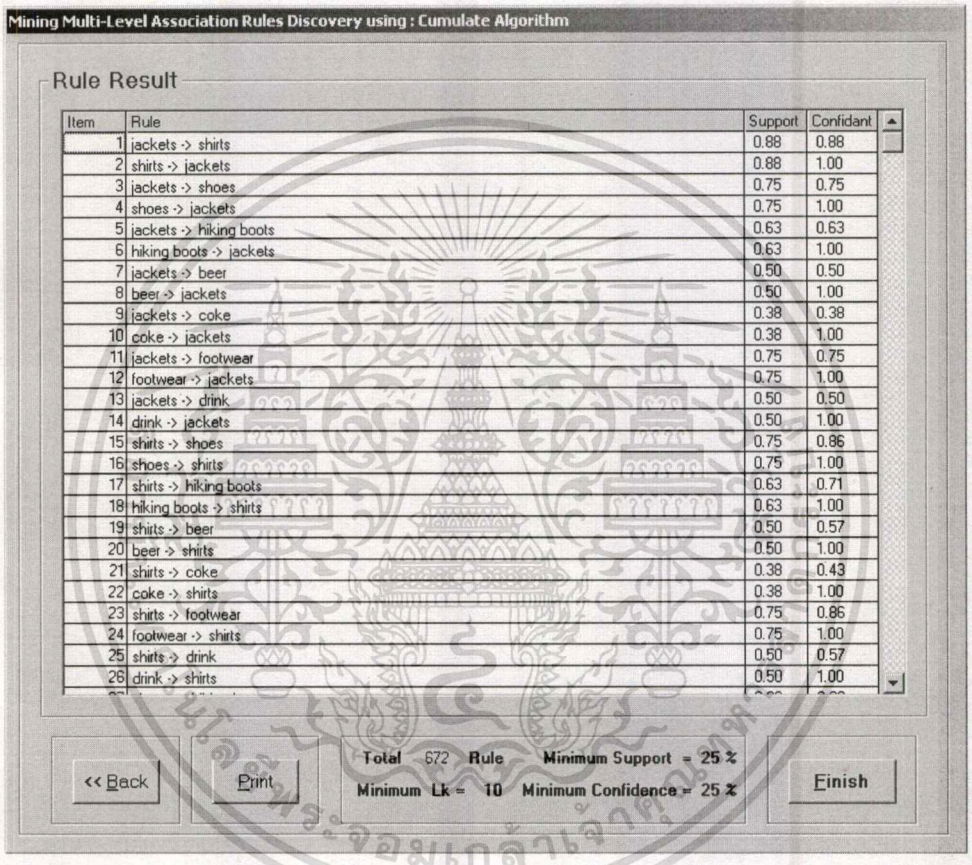
หลังจากพิจารณา เลือกข้อมูล และ Attribute ที่ต้องการเรียบร้อยแล้ว จากนั้นทำการ Run โปรแกรมซึ่งจะต้องกำหนดค่า Minimum Support และ Minimum Confidence ก่อนที่จะทำการ Mining ดังภาพที่ 6.6



ภาพที่ 6.7 หน้าจอแสดง การรับค่า Minimum Support และค่า Minimum Confidence

6.5 ผลที่ได้จากการวิเคราะห์ หากฎความสัมพันธ์

หลังจากโปรแกรม ทำการค้นหาความสัมพันธ์เรียบร้อยแล้ว จะแสดงผลลัพท์ออกมาเป็นกฎความสัมพันธ์ ดังในภาพที่ 6.7



ภาพที่ 6.8 หน้าจอแสดงผลลัพธ์เป็นกฎความสัมพันธ์ที่ได้จากการทำ Mining

บทที่ 7

สรุป

โครงการนี้จัดทำขึ้นมาเพื่อนำเสนอให้เห็นถึงการนำทฤษฎี Data Mining มาประยุกต์ใช้กับธุรกิจ ในการหาความสัมพันธ์ในรูปแบบต่างๆของข้อมูลที่มีอยู่ เพื่อนำผลลัพธ์ที่ได้ไปใช้ในการสร้างแผนกลยุทธ์ทางการตลาดเพื่อทำให้ธุรกิจได้รับกำไรสูงสุด

7.1 สรุปผลการดำเนินงาน

โครงการนี้เป็นการพัฒนาโปรแกรมเพื่อค้นหา Association Rule แบบ Mulit-level โดยใช้ Cumulate Algorithm ในการค้นหาความสัมพันธ์แบบหลายลำดับชั้น ซึ่งโปรแกรมที่พัฒนานี้สามารถติดต่อข้อมูลที่เป็น Relational Database โดยผู้ใช้งานสามารถกำหนดได้ว่าต้องการวิเคราะห์ข้อมูลอะไร ซึ่งผลลัพท์ที่ได้จะอยู่ในรูปแบบของกฎความสัมพันธ์แบบหลายลำดับชั้น และผู้ใช้ไม่จำเป็นต้องทำขั้นตอนการ Grouping ข้อมูลที่เกิดความสัมพันธ์จำนวนน้อยครั้ง จึงทำให้โปรแกรมนี้มีขั้นตอนการทำงานที่ไม่ยุ่งยาก ง่ายต่อการใช้งาน และสามารถนำไปประยุกต์ใช้ในการกำหนดทิศทาง ทางการตลาดได้อย่างมีประสิทธ์ภาพ

7.2 ข้อเสนอแนะ

ระบบนี้สามารถที่ทำงานได้อย่างมีประสิทธิภาพ ด้วยการใช้ข้อมูลจาก DataWarehouse เนื่องจากข้อมูลจะได้รับการทำความสะอาดเรียบร้อยแล้ว สามารถนำมาทำการ Mining ได้ทันที และระบบที่พัฒนาขึ้นมานี้สามารถที่จะใช้วิเคราะห์กับข้อมูลในธุรกิจต่างๆได้หลายรูปแบบ ทั้งสินค้า และบริการ โดยผู้ใช้สามารถเป็นคนกำหนดได้ว่าต้องการที่จะวิเคราะห์ข้อมูลอะไร จากตารางไหน

ภาคผนวก

ก. ความต้องการของระบบ

1) Microsoft windows 98 ,2000

2) Visual Basic Run Time 6.0 และ Service Pack 5

ข. โครงสร้างตารางที่ใช้ในระบบ

ในระบบได้มีการใช้งานตารางภายในระบบเอง ซึ่งเก็บอยู่ในฐานข้อมูล Microsoft Access ชื่อ DB1.mdb ซึ่งมีรายละเอียดดังนี้

- **ตาราง Data**

Transac_ID	Text	ลำดับของ Transaction.
Item	Text	Data Item

- **ตาราง Group**

Group_ID	Text	รหัสกลุ่มสินค้า
Group_Name	Text	ชื่อกลุ่มสินค้า

- **ตาราง Products**

Group_ID	Text	รหัสกลุ่มสินค้า
Product_ID	Text	รหัสสินค้า
Product_Name	Text	ชื่อสินค้า

บรรณานุกรม

Jiawei Han, Micheline Kamber. 2000. **Data Mining: Concept and Techniqies**. Morgan Kaufmann Publishers.

Jiawei Han, Yongjian Fu. 1999. **Mining Multiple-Level Association Rules in Large Databases.** IEEE Computer Society Member.

Ramakrishnan Srikant Rakesh Agrawal. 1995. **Mining Generalized Association Rules.** IBM Research Division Almaden Research Center.

Rakesh Agrawal and Ramakrishnan Srikant. 1994 .**Fast algorithms for mining association rules** In Proc. of the VLDB Conference, Santiago, Chile.

ประวัติผู้เขียน

นาย เอกภูมิ อารีรัตน์

เกิดวันที่ 28 เดือน เมษายน ปี พ.ศ. 2519

สถานที่เกิด จ.ราชบุรี

ประวัติการศึกษา

1. ระดับปริญญาตรี จาก คณะวิศวกรรมศาสตร์ สาขาเทคโนโลยีอีเล็กทรอนิกส์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง