การพัฒนาระบบดาต้าไมน์นิ่งแบบ Decision Trees โดยใช้ SQL Server

System Development of Building Decision Trees Model Application Using SQL Server

โดย

จตุรงค์ จิตติยพล

รหัสประจำตัว 46066742

อาจารย์ที่ปรึกษา

ผศ.ดร.วรพจน์ กรีสุระเดช

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 1 ปีการศึกษา 2548
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อหัวข้อ	การพัฒนาระบบดาต้าไมน์นิ่งแบบ Decision Trees โดยใช้ SQL Server
นักศึกษา	นายจตุรงค์ จิตติยพล
อาจารย์ที่ปรึกษา	ผศ.ดร.วรพจน์ กรีสุระเดช
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2548

บทคัดย่อ

เทคโนโลยีที่ใช้ในการวิเคราะห์ข้อมูลที่มีปริมาณมากให้ได้ข้อมูลที่มีประโยชน์สูงสุดเพื่อใช้ในการสนับสนุน การตัดสินใจตามกลยุทธ์ทางธุรกิจ โดยการใช้ Data Mining นั้นจะอาศัย Data Mining Model ในการนำเสนอเกี่ยวกับการจัดกลุ่ม (Grouping) และ การคาดเดา (Prediction) ผ่านเครื่องมือที่สนับสนุนการสร้าง Model ซึ่งสนับสนุน Algorithms Microsoft Decision Trees อาทิ Microsoft SQL Server เพื่อประโยชน์ในการได้มาซึ่งข้อมูลที่สามารถใช้ในการสนับสนุนการตัดสินใจได้ต่อไป

Title	System Development of Building Decision Trees Model Application Using SQL Server.
Student	Mr. Jaturong Chittiyaphol
Advisor	Asst. Prof. Dr. Worapoj Kreesuradej
Level of Study	Master of Science in Information
Major	Information Science
Academic	2005

ABSTRACT

Technology of analyzing the large amount of data for the most useful information which supports the decision making compatible with business strategies is data mining. This technology will require the data mining model for representing the grouping and the prediction via a tool which can build the model providing the algorithms Microsoft decision trees such as Microsoft SQL Server. The model will be very useful to obtain the information that supports the decision making.

กิตติกรรมประกาศ

โครงการพัฒนาระบบฉบับนี้ ได้รับความช่วยเหลือทั้งทางด้านความรู้ แนวทางปฏิบัติ และกำลังใจจากหลายท่าน ต้องขอขอบบุคคลดังต่อไปนี้

- คุณพ่อและคุณแม่ ที่อบรมสั่งสอนดูแลทั้งด้านการทำงาน การเรียน และการใช้ชีวิต
- ผศ.ดร.วรพจน์ กรีสุระเดช อาจารย์ที่ปรึกษาโครงการ ได้ให้ความกรุณาดูแลการทำงานและให้คำปรึกษาอันเป็นประโยชน์อย่างยิ่งต่อการพัฒนาโครงการจนแล้วเสร็จ
- คุณแม่อารี เทพกิจอารีกุล ที่อุปการะค่าเล่าเรียนในครั้งนี้
- เพื่อนๆน้องๆทุกคนที่คอยให้กำลังใจ และพิเศษสำหรับเพื่อน ๆ กลุ่มจตุรเทพ
- รุ่นพี่ที่คอยให้คำปรึกษาในการพัฒนาโปรแกรมโดยตลอด
- ทุก ๆ กำลังใจที่ไม่ได้เอ่ยนาม

ท้ายนี้ต้องขอขอบคุณ สถาบัน คณะ และคณาจารย์ทุกท่านที่ได้ให้ความกรุณาประสิทธิประสาทวิชาความรู้ จนสามารถพัฒนาโครงการพัฒนาระบบจนสำเร็จ

จตุรงค์ จิตติยพล

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

หน้า

ตารางที่

สารบัญภาพ

หน้า

รูปที่

สารบัญภาพ (ต่อ)

หน้า

รูปที่

บทที่ 1
บทนำ

1.1 ความเป็นมาของปัญหา

ในปัจจุบันมีเทคนิคและเทคโนโลยีที่สามารถเก็บข้อมูลที่มีปริมาณมากไว้ได้ แต่ก็ยังมีการนำข้อมูลเหล่านั้นมาใช้ให้เกิดประโยชน์ได้น้อย ดังนั้นจึงมีแนวความคิดที่จะนำข้อมูลที่มี มาใช้ให้เกิดประโยชน์สูงสุดเพื่อใช้ในการสนับสนุน การตัดสินใจตามกลยุทธ์ทางธุรกิจโดยการใช้ Data Mining ซึ่งจะอาศัย Data Mining Model ในการนำเสนอเกี่ยวกับการจัดกลุ่ม (Grouping) และ การคาดเดา (Prediction) และเพื่อเป็นการใช้ทรัพยากรที่มีอยู่ให้เหมาะสมและมีประสิทธิภาพสูงสุด ผ่านเครื่องมือที่สนับสนุนการสร้าง Model ซึ่งสนับสนุน Algorithms Microsoft Decision Trees อาทิ Microsoft SQL Server มาประยุกต์ใช้เพื่อประโยชน์ในการที่ได้มาซึ่งข้อมูลที่สามารถใช้ในการสนับสนุนการตัดสินใจได้ต่อไป (สมพร จิวรสกุล.2545:15)

อีกประเด็นหนึ่งคือ การนำเสนอข้อมูลแบบเดิม มักจะใช้วิธีการนำเสนอในรูปแบบที่เป็นรายงานที่มีรูปแบบที่แน่นอน ที่เรียกว่า Fixed Format ซึ่งบางครั้งผู้บริหารที่ได้รับข้อมูลนั้น อาจต้องการปรับเปลี่ยนรูปแบบ หรือมุมมองอย่างอื่น ในกรณีนี้ต้องให้ผู้ทำข้อมูลเป็นผู้ทำการปรับเปลี่ยนรูปแบบให้ โดยผู้ที่ใช้ข้อมูลไม่สามารถเปลี่ยนรูปแบบได้เอง ทำให้ไม่สะดวกและเป็นสาเหตุหนึ่งของการขาดประสิทธิภาพในการนำข้อมูลไปใช้งาน

กระบวนการที่สามารถนำเอาข้อมูลมานำเสนอให้มีประสิทธิภาพนั้น จะต้องพึ่งกระบวนการในการทำ Mining เสียก่อน ซึ่งจะทำให้สามารถเรียกดูข้อมูลได้เร็ว ซึ่งในโครงงานนี้เป็นการจัดทำโปรแกรมเพื่อช่วยในการจัดทำโมเดลในการทำ Data Mining บน SQL Server ผ่านเครื่องมือ Analysis Service ซึ่งจะสามารถทำให้นำเอาข้อมูลมาใช้ประโยชน์ได้อย่างสูงสุด

1.1.1 การนำเสนอข้อมูลด้วยโปรแกรมประยุกต์

เมื่อทำการพัฒนา Application ที่จะใช้ในการสร้างโมเดลและทำการวิเคราะห์ข้อมูลจนเสร็จเรียบร้อยแล้ว มักจะได้ข้อมูลที่อยู่ในมุมมองลูกบาศก์ (Cubes)อยู่แล้ว สำหรับผู้ใช้ที่มีความชำนาญทางด้านคอมพิวเตอร์ จะสามารถใช้งานการพิจารณาข้อมูลจาก Analysis Service ได้ แต่สำหรับผู้บริหารหรือผู้ที่ยังไม่มีความชำนาญในการใช้โปรแกรม Analysis Service นั้น มีความจำเป็นที่ต้องมีเครื่องมือที่ง่ายต่อการนำเสนอและใช้งาน ดังนั้นจึงได้พัฒนา โปรแกรมประยุกต์สำหรับงานเฉพาะด้าน ซึ่งพัฒนาจากโปรแกรม Microsoft Visual Basic 6 ขึ้นมาในการใช้งาน

1.1.2 ปัญหาที่พบในระบบงาน

จากการศึกษาการทำงานของการทำงานของ Analysis Server นั้นพบว่าปัญหาที่เกิดขึ้นนั้นเกิดขึ้นจากข้อจำกัดของการทำงานของตัวโปรแกรมส่งผลให้กระทบต่อการวิเคราะห์ข้อมูลผ่านตัวโมเดลที่ถูกสร้างขึ้น เนื่องจากการทำงานนั้นจะแยกส่วนของการทำงานระหว่างข้อมูลที่อยู่ใน SQL Server และ ข้อมูลที่เป็นแหล่งข้อมูลที่ใช้ในการสร้างโมเดล ที่อยู่ใน Analysis Server อีกทั้งการทำงานในส่วนของการวิเคราะห์ OLAP ก็ยังแยกส่วนของการทำงานอีกด้วย ทำให้การใช้งานนั้นมีความยุ่งยากในการใช้งานเพราะจะต้องเข้าไปจัดการกับงานหลายๆอย่าง เพื่อให้ได้ข้อมูลที่ต้องการจากการวิเคราะห์ และหากผู้ใช้งานไม่มีความชำนาญแล้วกอาจทำให้เกิดความเสียหายกับข้อมูลจริงได้ อาทิเช่น Case Study ของโครงงานในครั้งนี้คือ User คือผู้บริหารของบริษัทที่เป็นตัวกลางรับจัดทำและสมัครบัตรเครดิตระหว่างผู้สมัครกับผู้ให้บริการบัตร ต้องการทราบว่าปัจจัยใดที่มีผลต่อการเลือกทำบัตรเครดิตประเภทต่าง ๆ เพื่อที่จะได้ทำการเจาะกลุ่มลูกค้าในแต่ละกลุ่มได้อย่างถูกต้อง จึงต้องใช้ Data Mining เข้ามาช่วยในการวิเคราะห์ข้อมูลจากฐานข้อมูลของบริษัทที่มีอยู่เดิมอยู่แล้วโดยเลือกใช้เฉพาะข้อมูลที่ต้องการนำมาใช้

แนวทางในการแก้ไขปัญหา คือการพัฒนา Application ขึ้นมาเพื่อสนับสนุนการทำงานในส่วนของ SQL Server และ ในส่วน Analysis Server เข้ามาไว้ด้วยกันเพื่อความสะดวกในการทำงานทั้งการสร้างโมเดลในแบบ Decision Trees Model และการวิเคราะห์ข้อมูล OLAP อีกทั้งการสร้าง Virtual Cube ให้สามารถทำงานได้บน Application เดียวกัน

ดังนั้นการพัฒนาระบบดังกล่าว จะสามารถรองรับการทำงานในส่วนต่าง ๆที่แยกกันอยู่เพื่อให้สามารถทำงานร่วมกันได้บน Application เดียวกันได้ อีกทั้งมีความยืดหยุ่นในการเลือกเงื่อนไขในการวิเคราะห์ข้อมูลที่มากขึ้นเพราะเงื่อนไขบ้างอย่างหากไม่ใส่ไว้ในโมเดลแล้วก็จะไม่สามารถทำการวิเคราะห์ได้

1.2 วัตถุประสงค์ของระบบงาน

การจัดทำโปรแกรมเพื่อช่วยในการสร้างโมเดลในการทำ Data Mining นั้นเกิดขึ้นเนื่องจากความประสงค์ที่จะช่วยลดปัญหาของการใช้โปรแกรม Analysis Service บน SQL Server เนื่องจากจะต้องใช้ความชำนาญในการใช้งาน ซึ่งอาจทำให้เสียเวลาในการวิเคราะห์ข้อมูลที่มีอยู่เป็นจำนวนมาก หรืออาจทำให้ผลการวิเคราะห์นั้นผิดพลาดคลาดเคลื่อนไปจากความเป็นจริง โดยมีจุดประสงค์ดังนี้

1.2.1 เพื่อนำความรู้และเทคนิคที่ศึกษาไปประยุกต์ใช้กับการวิเคราะห์ข้อมูลผ่าน Data Mining Model

1.2.2 เพื่อเป็นการใช้ทรัพยากรที่มีอยู่อย่างเหมาะสมและมีประสิทธิภาพสูงสุด

1.2.3 เพื่อให้ได้ข้อมูลที่สามารถนำมาช่วยในการตัดสินใจได้อย่างถูกต้อง และรวดเร็ว

1.2.4 เพื่อเพิ่มความสามารถในการเรียกใช้ข้อมูลสำหรับผู้บริหาร

1.3 ขอบเขตการดำเนินงาน

ระบบงานที่ทำการศึกษานี้ จะเป็นการศึกษาและพัฒนาระบบงานโดยศึกษาเทคนิค Data Mining โดยทำการเลือกเทคนิค Model แบบ Decision Trees Model จากเครื่องมือ SQL Server มาทำการประยุกต์ใช้ผ่าน Application ในการพัฒนาระบบงานในครั้งนี้ โปรแกรมที่ทำการพัฒนาขึ้นจะเป็นเครื่องมือที่ช่วยให้การทำงานในการนำข้อมูลที่มีอยู่มากมายมาทำการวิเคราะห์หาผลลัพธ์ได้อย่างมีประสิทธิภาพมากขึ้น โดยโปรแกรมที่พัฒนาขึ้นมานี้ จะมุ่งเน้นในการนำข้อมูลจาก Cubes ที่ถูกสร้างไว้ใน SQL Server มาใช้ให้เกิดประโยชน์ และง่ายต่อการใช้งาน

นอกจากนี้ขอบเขตการทำงานของโครงงานนี้ยังจะกล่าวถึงกระบวนการนำเอาข้อมูลที่ได้ทำการสร้างเป็นโมเดลเอาไว้แล้วออกมานำเสนอในรูปแบบต่าง ๆ ด้วยรูปแบบที่สนับสนุนการติดต่อผู้ใช้งานผ่านทางหน้าจอบนระบบปฏิบัติการ Windows ได้อย่างง่ายดายตามความต้องการของผู้ใช้งาน ซึ่งจะครอบคลุมการทำงานดังต่อไปนี้

- ทำการสร้างโมเดลด้วย Algorithm Microsoft_Decision_Trees บน SQL Server ผ่านการทำงานโปรแกรม Application
- ทำการนำข้อมูลที่อยู่ในรูปแบบ Cubes ออกมานำเสนอในรูปแบบต่าง ๆ เช่นรูปแบบของตัวเลข หรือ ในรูปแบบของแผนภูมิเพื่อความเข้าใจและนำไปใช้งานได้ง่าย

1.4 เทคนิคและวิธีการที่ใช้ในการพัฒนาระบบงาน

ในการศึกษาและพัฒนาระบบจะใช้เทคนิคของ Data Mining โดยเลือกใช้ Decision Trees Model ผ่าน SQL Server ซึ่งสนับสนุน Algorithms Microsoft Decision Trees เพื่อให้เรียกใช้ผ่านทาง Application ได้

1.5 องค์ประกอบของการพัฒนาระบบงาน

1.5.1 คอมพิวเตอร์ COMPAQ รุ่น Presario X1000

1.5.2 Microsoft Visual Basic 6.0

1.5.3 Microsoft SQL Server

1.5.4 ระบบปฏิบัติการ Windows 2000 Server

1.6 ขั้นตอนและวิธีการดำเนินงาน

1.6.1 ศึกษาเกี่ยวกับข้อมูลและเทคนิคการเชื่อมต่อของ OLE DB for Data Mining

1.6.2 ศึกษาเทคนิคและวิธีการของ Data Mining

1.6.3 คัดเลือกเทคนิคและ Algorithms ที่เหมาะสมกับข้อมูลที่จะทำการศึกษา

1.6.4 รวบรวมและจัดการข้อมูลที่จะนำมาใช้ในการพัฒนาระบบ

1.6.5 พัฒนาระบบงาน

1.6.6 ตรวจสอบและทำการแก้ไขของระบบงานให้มีความถูกต้อง

1.6.7 จัดทำเอกสารประกอบ

1.7 ประโยชน์ที่คาดว่าจะได้รับ

1.7.1 ได้ความรู้และเทคนิคที่ศึกษาที่สามารถประยุกต์ใช้ในการวิเคราะห์ข้อมูลผ่าน Data Mining Model

1.7.2 เพื่อเป็นการใช้ทรัพยากรที่มีอยู่อย่างมีประสิทธิภาพ

1.7.3 เพื่อช่วยให้การทำงานสามารถทำได้เร็วขึ้น

จากรายละเอียดที่กล่าวมาข้างต้นเป็นหลักการที่ใช้ในการพัฒนาระบบ รวมทั้งประโยชน์ที่ได้รับจากการศึกษาโครงการในครั้งนี้ ในรายละเอียดของบทต่อไปจะเป็นการศึกษาและทำความเข้าใจในข้อมูลและมาตรฐานในการเชื่อมต่อที่จะนำใช้ในการพัฒนาระบบ โดยเป็นความรู้เบื้องต้นที่เกี่ยวกับ OLE DB for Data Mining Specification Versions 1.0 Microsoft Corporation เพื่อทราบถึงแนวทางในการที่จะทำการเชื่อมต่อการทำงานระหว่าง Model กับ Application

1.8 รายละเอียดของข้อมูลในบทต่าง ๆ

- **บทที่ 2** จะกล่าวถึงแนวความคิดของไมโครซอฟท์ที่มีต่อ Business Intelligence และมาตรฐานการเชื่อมโยงฐานข้อมูลในรูปแบบต่าง ๆ
- **บทที่ 3** จะกล่าวถึงทฤษฎีที่เกี่ยวข้องในส่วนของ Data Mining เทคนิคต่าง ๆที่สามารถนำมาทำ Data Mining ได้ รวมไปถึงรูปแบบต่าง ๆที่ใช้ในการทำโมเดลในการทำ Data Mining
- **บทที่ 4** กล่าวถึงความหมายของ Decision Trees และอัลกอริธึมต่างๆ ที่ใช้ในการสร้าง Decision Trees ร่วมไปถึงเครื่องมือที่ใช้ในการทำ Data Mining
- **บทที่ 5** กล่าวถึงผลของการทดสอบ Application ที่ได้ทำการสร้างขึ้นมาเพื่ออธิบายขั้นตอนการทำงานในแต่ละขั้น

ความเป็นมาของโครงการทั้งหมดซึ่งประกอบไปด้วย ความเป็นมาของโครงการ วัตถุประสงค์ ขอบเขตการทำงาน และการวางแผน รวมไปถึงประโยชน์ที่จะได้รับจากการพัฒนาโครงการ ซึ่งจะแสดงให้เห็นถึงการพัฒนาระบบที่เป็นไปอย่างมีแบบแผน และเป็นขั้นตอน

บทที่ 2

OLE DB for Data Mining

2.1 แนวความคิดด้านแอปพลิเคชั่น

แนวคิดของ ไมโครซอฟท์ที่มีต่อ Business Intelligence คือ การออกแบบให้ระบบการจัดการคลังข้อมูลเป็นเรื่องง่าย ซึ่ง SQL Server ได้มีสิ่งจำเป็นทั้งหมด ต่อการสร้างแอปพลิเคชั่น ซึ่งประกอบด้วย

- ฐานข้อมูลเชิงสัมพันธ์สำหรับจัดเก็บข้อมูลเชิงสัมพันธ์ หรือคลังข้อมูล (ทำงานโดย SQL Server เอง)
- Data Transformation Services (DTS) ซึ่งเป็นซอฟต์แวร์ สำหรับคัดแยก แปลงรูปแบบ และโหลดข้อมูล (ETL) จากระบบปฏิบัติการสู่หน่วยจัดเก็บข้อมูลเชิงสัมพันธ์
- ซอฟต์แวร์บริหารระบบ ใช้ในการจัดการทั้งฐานข้อมูลเชิงสัมพันธ์ และกลไก OLAP
- ความสามารถในการจัดทำ Data mining
- บริการ Meta Data
- การบริหารระบบแบบกราฟิก และสนับสนุนการเชื่อมต่ออย่างกว้างขวาง
- กลไก OLAP (Analysis Server)

2.2 มาตรฐานการเชื่อมโยงข้อมูล(Cabena.1998:125)

ทางไมโครซอฟต์ ได้ประกาศใช้อินเทอร์เฟสและมาตรฐานแบบเปิด สำหรับการทำงานร่วมกับผลิตภัณฑ์จากผู้ผลิตรายอื่นๆ ซึ่งไมโครซอฟต์ได้ชี้ถึงวัตถุประสงค์ในหลายทางด้วยกัน ทั้ง SQL Server และ Analysis Server เผยให้เห็นถึงออปเจ็คโมเดล และเปิดโอกาสให้เชื่อมต่อผ่านทางโปรแกรมโดยตรง โดยข้ามขั้นตอนของอินเทอร์เฟส โดยการจัดการของไมโครซอฟต์ เช่น

- SQL Distributed Management Object (SQL-DMO) เป็นออปเจ็คโมเดลสำหรับ SQL Server
- DSO (Decision Support Objects) เป็นออปเจ็คโมเดลของ Analysis Manager
- ActiveX Data Objects Multidimensional (ADO MD) เป็นออปเจ็คโมเดลสำหรับ PivotTable Services ซึ่งเป็นส่วนประกอบด้านไคลเอนท์ของ Analysis Services

2.3 การเชื่อมต่อแบบ OLE DB

การเชื่อมต่อ OLE DB เป็นการเชื่อมต่อในระดับต่ำไปยังข้อมูลของทั้งองค์กร ในขณะที่มาตรฐานเฉพาะฐานข้อมูล เช่น ODBC ได้รับการออกแบบให้ครอบคลุมฐานข้อมูลเชิงสัมพันธ์ทั้งหมด ส่วนมาตรฐาน OLE DB ได้รับการออกแบบให้เรียกใช้ได้ทั้งแหล่งข้อมูลแบบสัมพันธ์ และไม่ใช่แบบสัมพันธ์ จึงให้ความสามารถในการทำงานร่วมกันได้ในขอบเขตที่กว้างกว่า มาตรฐาน OLE DB เป็นข้อกำหนดแบบเปิด ที่ได้รับการพัฒนาร่วมกันจากหลายองค์กรในอุตสาหกรรมและเปิดให้นักพัฒนาทั่วโลกได้ใช้งานอย่างทั่วถึง Analysis Services ได้ขยายมาตรฐานด้าน OLE DB ด้วยข้อกำหนดเพิ่มเติม 3 ประการ ได้แก่

- OLE DB for OLAP เป็นซอฟต์แวร์สำหรับการเรียกค้นและจัดทำรายงานนั้นมีให้เลือกใช้เป็นจำนวนมาก และไม่มีซอฟต์แวร์ใดที่สามารถตอบสนองความต้องการของธุรกิจได้ทุกประเภท ดังนั้นธุรกิจหนึ่ง ๆ จึงอาจมีมาตรฐานของซอฟต์แวร์เรียกค้นและทำรายงานได้มากกว่า 1 แบบ ซึ่ง OLE DB for OLAP จะทำหน้าที่ในการเชื่อมต่อกับแหล่งข้อมูลแบบหลายมิติ และยอมให้แหล่งข่อมูล OLAP เชื่อมกับแหล่งข้อมูลหลากหลายประเภท
- OLE DB for Data Mining ได้รับการพัฒนาจากความร่วมมืออย่างกว้างขวางในวงการ Data Mining ทำให้ผลิตภัณฑ์ Data Mining ที่มีอยู่อย่างกระจัดกระจาย สามารถแลกเปลี่ยนทั้งข้อมูลและผลลัพธ์ระหว่างกันได้ ด้วยข้อกำหนดของ OLE DB for Data Mining บริษัทต่าง ๆ สามารถใช้ความสามารถ Data Mining ที่มีอยู่กับ Analysis Services ทำการรวมอัลกอริธึมพิเศษหรือแอปพลิเคชั่นเข้าไปเป็นส่วนหนึ่งของแอปพลิเคชั่น Analysis Services หรือใช้ Analysis Services cube ได้ ในฐานะของแหล่งข้อมูลสำหรับแอปพลิเคชั่น Data Mining
- XML for Analysis ในข้อกำหนดของทั้ง OLE DB for OLAP และ OLE DB for Data Mining ต่างใช้องค์ประกอบที่ไคลเอนท์ต้องขึ้นอยู่กับเซิร์ฟเวอร์อย่างมาก รูปแบบเช่นนี้ใช้งานได้ดีในสภาพแวดล้อมหลายแบบ แต่ไม่เหมาะกับไคลเอนท์ความสามารถต่ำ ที่ทำงานบนเว็บ เช่น ไคลเอนท์ ประเภทบราวเซอร์

ข้อกำหนดของ XML for Analysis สนับสนุนการทำงานแบบไม่ผูกติดและเชื่อมโยงแบบไม่มีสถานะในสภาพแวดล้อมของเว็บ จึงเอื้อต่อการเรียกใช้แหล่งข้อมูลวิเคราะห์ทุกรูปแบบ (OPAP และ Data Mining) ผ่านเว็บ ข้อกำหนดนี้ถูกออกแบบให้ทำงานได้ดีที่สุดผ่านเว็บ โดยลดการส่งข้อมูลไปมา และลดปริมาณการใช้เครือข่ายลงช่วยให้เกิดสภาพแวดล้อมแบบผสมผสานจริง ๆ สำหรับการเรียกใช้ข้อมูลจากแพล็ตฟอร์มใดก็ได้ เนื่องจากว่าใน

ข้อกำหนดนั้นใช้ XML สำหรับการแลกเปลี่ยนข้อมูล และแอปพลิเคชั่น ที่นำข้อกำหนดนี้ไปใช้ จึงสามารถเขียนขึ้นด้วยภาษาใด บนแพล็ตฟอร์มใดก็ได้

2.4 OLE DB for Data Mining Version 1.0 (Microsoft Corporation.2000:87)

2.4.1 OLE DB for DM Specification

เป็นมาตรฐานที่ทำให้ผลิตภัณฑ์ Data Mining ที่มีอยู่ สามารถแลกเปลี่ยนทั้งข้อมูลและผลลัพธ์ระหว่างกันได้ และสามารถเข้าถึงข้อมูลในโมเดลได้ ด้วยข้อกำหนดของ OLE DB for Data Mining อาทิเช่น

- การสร้าง OLE DB เพื่อเป็นกลไกมาตรฐานในการเชื่อมต่อข้อมูล
- การสร้าง Data Mining Model โดยสามารถใช้คำสั่งในการสร้างที่มีรูปแบบคำสั่งคล้ายกับคำสั่งในการสร้างตารางคือ CREATE TABLE โดยใช้ CREATE MINING MODEL
- สามารถใส่ข้อมูลที่ต้องการทดสอบเข้าไปในโมเดลคล้ายกับคำสั่ง INSERT INTO
- สามารถใช้ผลลัพธ์จาก DMM มาช่วยในการตัดสินใจโดย เลือกข้อมูลที่ต้องการร่วมกับโมเดล

2.4.2 รูปแบบคำสั่ง

เป็นรูปแบบคำสั่งในการสร้างโมเดลโดยการใช้ Command ในการสร้างโมเดลที่ต้องการ รวมไปถึงกระทั้งการใส่ข้อมูลเข้าไปในโมเดล เพื่อทำการวิเคราะห์ข้อมูลให้ได้ผลลัพธ์ที่ถูกต้องออกมาใช้ร่วมในการตัดสินใจ โดยมีการแบ่งรูปแบบคำสั่งดังนี้

- **CREATE MINING MODEL**

เป็นรูปแบบคำสั่งที่ใช้ในการสร้างโมเดลที่มีความสัมพันธ์กับคำสั่ง SQL ที่เป็นคำสั่งพื้นฐาน

```
CREATE MINING MODEL <model>
(
    <column definition list>
)
  USING <algorithm> [(<parameter list>)]
```

ตารางที่ 2.1 อธิบายพารามิเตอร์คำสั่ง CREATE MINING MODEL ได้ว่า

<model>	ระบุชื่อโมเดลที่ต้องการสร้าง เช่น [Age Prediction]
<column definition list>	ระบุชื่อฟิลด์ที่จะสร้าง เช่น [Customer ID]
<algorithm>	ระบุชื่ออัลกอริธึมที่ใช้ เช่น [Decision Trees]
<parameter list>	(ข้อกำหนดเพิ่มเติม)

โดยในส่วนของ column นั้นยังมีรูปแบบที่กำหนดเอาไว้คือ

<column name><type>

ตารางที่ 2.2 อธิบายพารามิเตอร์ column ได้ว่า

<column name>	ระบุชื่อคอลัมน์ที่ต้องการ
<type>	ระบุชนิดข้อมูลที่จะใช้ LONG, DOUBLE, DATE, TEXT,

ตัวอย่างการใช้คำสั่ง

```
CREATE MINING MODEL [Age Prediction]
(
  [Customer ID]       LONG    KEY,
  [Gender]            TEXT    DISCRETE,
  [Product Purchases] TABLE
   (
    [Product Name]    TEXT   KEY,
    [Quantity]        DOUBLE  NORMAL _&
                      CONTINUOUS,
    [Product Type]    TEXT  DISCRETE RELATED TO
                [Product Name]
)
)
```

USING [Microsoft_Decision_ Trees]

- **INSERT INTO**

เป็นรูปแบบคำสั่งที่ใช้ในการเพิ่มข้อมูลเข้าไปในโมเดล เพื่อใช้ในการวิเคราะห์ข้อมูล

INSERT INTO <model> (<mapped model columns>)

<source data query>

ตารางที่ 2.3 อธิบายพารามิเตอร์คำสั่ง INSERT INTO ได้ว่า

<model>	ระบุชื่อโมเดลที่จะใส่ข้อมูลเข้าไป
<mapped model columns>	ระบุชื่อคอลัมน์โดยมีเครื่องหมายจุลภาคคั่น
<source data query>	รูปแบบการสอบถามข้อมูล

ตัวอย่างการใช้คำสั่ง

INSERT INTO [Age Prediction]

(

[Customer ID], [Gender], [Age],

[Product Purchases](SKIP, [Product Name], [Quantity], [Product Type])

)

- **SELECT**

เป็นรูปแบบคำสั่งที่ใช้ในการเลือกข้อมูลจากโมเดล ที่ต้องการข้อมูล

SELECT * INTO <new model>

USING <algorithm> [(<parameter list>)]

FROM <existing model>

ตารางที่ 2.4 อธิบายพารามิเตอร์คำสั่ง SELECT ได้ว่า

<new model>	ชื่อโมเดลที่ถูกสร้างขึ้น
<algorithm>	ชื่ออัลกอริธึมที่เรียกใช้
<parameter list>	ข้อกำหนดเพิ่มเติม
<existing model>	ชื่อของโมเดลที่ถูกคัดลอก

ตัวอย่างการใช้คำสั่ง

SELECT t.[Customer ID], [Age Prediction].[Age]

FROM [Age Prediction]

- **DELETE**

เป็นรูปแบบคำสั่งที่ใช้ในการลบข้อมูลจากโมเดล ที่ต้องการ

DELETE * FROM <model>[.CONTENT]

ตารางที่ 2.5 อธิบายพารามิเตอร์คำสั่ง DELETE ได้ว่า

<model>	ระบุชื่อโมเดลที่ประกาศไว้

- **DROP**

เป็นรูปแบบคำสั่งที่ใช้ในการหยุดการทำงานจากโมเดล ที่ต้องการ

DROP MINING MODEL <model>

ตารางที่ 2.6 อธิบายพารามิเตอร์คำสั่ง DROP ได้ว่า

<model>	ระบุชื่อโมเดลที่ประกาศไว้

2.5 สรุป

ทั้ง SQL Server และ Analysis Services เป็นเครื่องมือที่มีประสิทธิภาพและเหมาะสมที่จะนำมาใช้ในการจัดการคลังข้อมูล และสามารถใช้ประโยชน์จากการวิเคราะห์ข้อมูลได้อย่างลึกซึ้ง และมีประสิทธิภาพในการนำข้อมูลมาใช้ในการช่วยตัดสินใจสูงสุด

บทที่ 3
ดาต้าไมนิ่ง

Data Mining เป็นกระบวนการที่ใช้ Data Analysis และ Model หลากหลายเพื่อค้นหารูปแบบและความสัมพันธ์ของข้อมูลที่มีอยู่ สามารถนำไปใช้ในการทำนายและตัดสินใจทางธุรกิจได้ รูปแบบการทำงานของดาต้าไมน์นิ่งมี 2 วิธีคือ วิธีการดาต้าไมน์นิ่งแบบทางตรง (Directed Data Mining) และวิธีการดาต้าไมน์นิ่งแบบทางอ้อม (Undirected Data Mining) โดยวิธีแบบทางตรงนั้นเป็นการทำงานแบบ Top-down จะใช้วิธีการนี้เมื่อรู้ผลลัพธ์คร่าวๆ ว่าต้องการค้นหาอะไรหรือต้องการค้นหาอะไร ซึ่งรูปแบบการทำนายนั้นจะใช้ประสบการณ์ที่หาผลลัพธ์ที่เป็นไปได้ในอนาคต ส่วนวิธีการแบบทางอ้อมนั้นเป็นการทำนายแบบ Bottom-up รูปแบบในการค้นหาข้อมูลขึ้นกับการตัดสินใจของผู้ใช้ และยังมีเทคนิคทาง Data Mining เพื่อช่วยในการตัดสินใจเช่น Classification ที่ใช้สำหรับสร้างแบบจำลองการพยากรณ์ (Prediction Model)โดยจะทำการสร้างแบบจำลองจากกลุ่มข้อมูลตัวอย่างที่เลือกมาจากฐานข้อมูลขนาดใหญ่ และแบบจำลองนั้นสามารถพยากรณ์ผลลัพธ์ของข้อมูลที่ไม่เคยพบเห็นมาก่อนได้

3.1 ดาต้าไมนิ่ง (Data Mining)

ดาต้าไมน์นิ่ง(Data Mining) คือกระบวนการค้นหาความสัมพันธ์และรูปแบบทั้งหมดซึ่งมีอยู่จริงในฐานข้อมูล (Data base) แต่ได้ถูกซ่อนไว้ภายในข้อมูลจำนวนมากซึ่งจะช่วยในการเพิ่มมูลค่า (Value added)ให้กับข้อมูลที่องค์กรที่มีอยู่ โดยเป็นการทำงานร่วมกันระหว่างมนุษย์กับคอมพิวเตอร์ โดยมนุษย์จะทำหน้าที่ในการออกแบบฐานข้อมูล กำหนดจุดประสงค์หรือเป้าหมายในการทำงาน และคอมพิวเตอร์จะทำหน้าที่ในการคำนวณหรือค้นหาความสัมพันธ์ ของข้อมูลตามเป้าหมายที่ได้กำหนดไว้ โดยจะนำข้อมูลทั้งระดับปฏิบัติการ (Transaction data) ฐานข้อมูล (Data base) หรือคลังข้อมูล (Data warehouse) มาผ่านขบวนการทำงานที่เรียกว่า process ที่สกัดข้อมูล Extract data จากฐานข้อมูลขนาดใหญ่ Large Information เพื่อให้ได้สารสนเทศ Usefull Information ที่เรายังไม่รู้ Unknown data โดยเป็นสารสนเทศที่มีเหตุผล Valid และสามารถนำไปใช้ได้ Actionable ซึ่งเป็นสิ่งสำคัญในการที่จะช่วยการตัดสินใจในการทำธุรกิจ ดาต้าไมน์นิ่ง เป็นโปรเซสที่สำคัญในการทำ Knowledge Discovery in Database ที่เราเรียกสั้น ๆ ว่า KDD ส่วนดาต้าไมน์นิ่ง สามารถเรียกสั้น ๆ ว่า DM

ดาต้าไมน์นิ่ง บางครั้งถูกมองว่าเป็นชุด Software วิเคราะห์ข้อมูลที่ได้ถูกออกแบบมา เพื่อระบบสนับสนุนการตัดสินใจของผู้ใช้ ดาต้าไมน์นิ่ง เป็น Software ที่สมบูรณ์ทั้งเรื่องการค้นหา การทำรายงาน และโปรแกรมในการจัดการ ซึ่งเราคุ้นเคยดีกับคำว่า Executive information system (EIS) หรือระบบข้อมูลสำหรับการตัดสินใจในการบริหาร Decision Support System คือทำอย่างไรให้ข้อมูลที่เรามีอยู่กลายเป็นความรู้อันมีค่าได้(joshi.1997:37)

สำหรับ Philippe Nieuwbourg (CXP Information)กล่าวไว้ว่า ดาต้าไมน์นิ่งคือ เทคนิคที่ผู้ใช้สามารถปฏิบัติการได้โดยอัตโนมัติ กับข้อมูลที่ไม่รู้จัก ซึ่งเป็นการเพิ่มคุณค่า ให้กับข้อมูลที่มีอัตโนมัติหมายถึง กระบวนการทำงานของดาต้าไมน์นิ่งจะเป็นผู้ทำงานเองไม่ใช่ผู้ใช้ กระบวนการจะไม่ให้คำตอบกับปัญหาที่มี แต่จะเป็นศูนย์กลางของข้อมูล

ข้อมูลที่ไม่รู้จักหมายถึง เครื่องมือในการค้นหาข้อมูลของดาต้าไมน์นิ่ง จะไม่ค้นหาเฉพาะข้อมูลเก่าและข้อมูลที่ผู้ใช้ป้อนเท่านั้น แต่จะค้นหาข้อมูลใหม่ ๆ ให้ด้วย

เพิ่มคุณค่าหมายถึง ผู้ใช้ไม่ได้เป็นเพียงนักสถิติเท่านั้น แต่เป็นได้ถึงระดับการตัดสินใจ

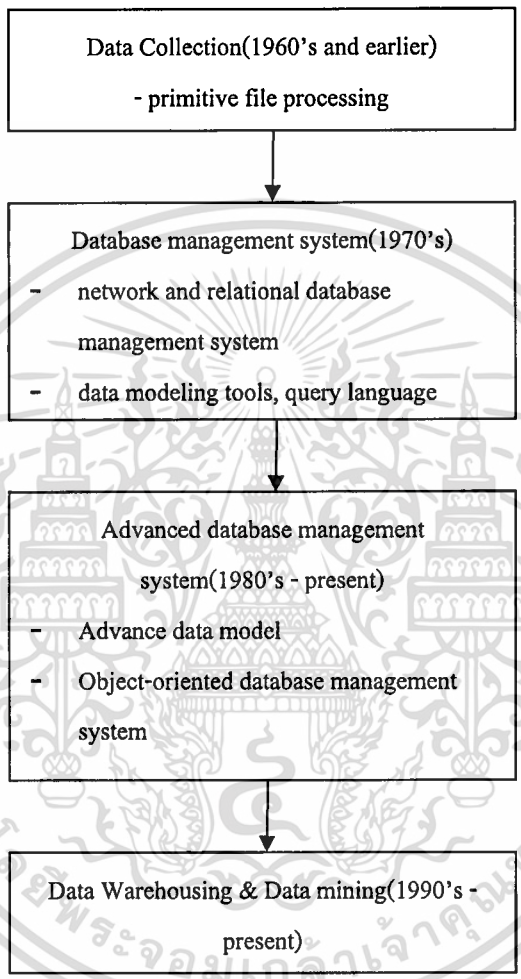
รูปที่ 3.1 แสดงข้อมูลสู่การตัดสินใจและปฏิบัติ

3.2 ที่มาของดาต้าไมน์นิ่ง

ดาต้าไมน์นิ่งได้มีการนำแนวคิดและวิธีการต่าง ๆ ของฐานข้อมูล สถิติ และการเรียนรู้ของเครื่องจักร(Machine learning)โดยจะเน้นไปที่ข้อมูลที่มีขนาดใหญ่ดังรายละเอียดต่าง ๆ ดังนี้มาใช้ในการทำไมน์นิ่ง

- ฐานข้อมูล (Database Technology) การทำงานที่ดาต้าไมน์นิ่ง นำมาจากฐานข้อมูลคือการเก็บข้อมูล การคำนวณ เกี่ยวกับการทำงานที่ทำซ้ำ ๆ และให้ข้อมูลเฉพาะในส่วนที่เก็บไว้เท่านั้น
- สถิติ(Statistics) คือ การรวบรวมข้อมูลที่มีอยู่แล้ว นำทฤษฎีทางสถิติ มาวิเคราะห์เพื่อที่จะบอกถึงค่าความเป็นได้ต่างๆ ที่เกิดขึ้น ซึ่งตรงส่วนนี้ดาต้าไมน์นิ่ง ได้นำแนวความคิดนี้ไปใช้ต่างๆ มาจากสถิติ นำเครื่องจักรมาช่วยในการคำนวณที่เป็นข้อมูลขนาดใหญ่ นำการนำเสนอข้อมูลมาช่วยสำหรับการเตรียมข้อมูลที่จะใช้วิเคราะห์ให้เข้าใจ

ได้ง่ายขึ้น จากอดีตจนถึงปัจจุบัน ได้มีการพัฒนาการข้อมูลมาเรื่อยๆจนกระทั้งมีการพัฒนาการใช้ ดาต้าไมน์นิ่งขึ้น(Peter Cabena.1998:205)



รูปที่ 3.2 แสดงวิวัฒนาการเทคโนโลยีถึงดาต้าไมน์นิ่ง

3.3 วัฏจักรขั้นตอนการทำงานของดาต้าไมน์นิ่ง (Virtuous cycle of data mining)

วัฏจักรขั้นตอนการทำงานของดาต้าไมน์นิ่งประกอบไปด้วย 4 ขั้นตอนหลัก ๆ ดังนี้

1. การระบุโอกาสทางธุรกิจหรือการระบุปัญหาที่เกิด ขึ้นกับธุรกิจ คือ จะเป็นการระบุขอบเขตของข้อมูลที่จะนำมาทำการวิเคราะห์ เพื่อหาความได้เปรียบทางการตลาดหรือเพื่อนำมาแก้ปัญหา
2. ส่วนของดาต้าไมน์นิ่ง จะเป็นการนำเทคนิคของ ดาต้าไมน์นิ่ง ไปใช้การถ่ายทอดหรือทำการเปลี่ยนแปลงข้อมูลดิบให้อยู่ในรูปของข้อมูลที่สามารถนำไปปฏิบัติได้จริงในทางธุรกิจ
3. การปฏิบัติตามข้อมูล (Act on the information) เราจะนำข้อมูล ที่เป็นผลลัพธ์ของส่วนดาต้าไมน์นิ่งมาลองปฏิบัติจริงกับธุรกิจ
4. การวัดประสิทธิภาพจากผลลัพธ์ (Measure the results) จะทำการวัดประสิทธิภาพของเทคนิคดาต้าไมน์นิ่ง ที่จะนำมาใช้จากผลลัพธ์ ซึ่งสามารถตรวจสอบได้หลายทางอาจวัดจากส่วนแบ่งทางการตลาด , ปริมาณลูกค้า และวัดจากกำไรสุทธิที่ได้ เป็นต้น

3.4 เทคนิคดาต้าไมนิ่ง (Data Mining Techniques)

สำหรับเทคนิคที่ใช้ใน Classification แบ่งได้เป็น 2 แบบ ได้แก่ Tree Induction และ Neural Induction และเพื่อให้การศึกษาบรรลุวัตถุประสงค์ จึงกำหนดขั้นตอนในการศึกษาโดยอิงตามกระบวนการทำงานของ Data Mining ดังนี้

ขั้นตอนที่ 1 จัดเตรียมข้อมูลในการวิเคราะห์ ในการจัดเตรียมข้อมูลที่นำมาใช้ให้ เป็นข้อมูลที่เหมาะสมกับอัลกอริธึม ได้แบ่งออกเป็นขั้นตอนดังนี้

- การคัดเลือกข้อมูล (Data Selection) โดยคำนึงถึง

วัตถุประสงค์ในการนำข้อมูลมาใช้งาน ซึ่งข้อมูลที่ได้อาจได้มาจากข้อมูลหลายๆแหล่ง โดยจะต้องทำให้ข้อมูลเหล่านั้นอยู่ในรูปแบบเดียวกันเสียก่อน

- การเตรียมข้อมูล (Data Preprocess) เป็นการนำ

ข้อมูลที่ถูกเลือกเข้ามาจากกระบวนการ Data Selection ซึ่งข้อมูลเหล่านี้เราจะต้องนำมาจัดการกับ Noisy Data คือค่าของข้อมูลที่ผิดไปจากค่าที่ควรจะเป็น แก้ไขโดยค้นหาค่าผิดนำมาแก้ไข

- การแปลงข้อมูล (Data Transformation) เป็นการปรับเปลี่ยนรูปแบบของข้อมูลให้เหมาะสมกับอัลกอริธึมที่เลือกใช้ เช่น แปลงข้อมูลตัวเลขให้เป็นช่วงเพื่อใช้กับ ID3 Algorithm เป็นต้น

ขั้นตอนที่ 2 กระบวนการจัดการข้อมูล การจัดหมวดหมู่ของข้อมูล เป็นการแบ่งประเภทข้อมูลแต่ละหน่วยออกตาม Class ที่ได้กำหนดไว้ก่อนแล้ว คือข้อมูลที่เข้ามารียกว่า Training data ประกอบด้วยหลายๆ Object แต่ละ Object มีหลาย Attributes ซึ่งแต่ละ Attributes ก็จะมีชื่อเฉพาะแต่ละ Attribute (Class Label) งานของการทำ Classification คือการวิเคราะห์ Training data และพัฒนาเป็น Model ของแต่ละ Class เพื่อแสดงถึงลักษณะของข้อมูล

3.4.1 Tree Induction

เทคนิค tree induction จะสร้างตัวอย่างในการทำนายในรูปแบบของ Decision Trees) เป็นที่นิยมกันมากเนื่องจากเป็นลักษณะที่คนจำนวนมากคุ้นเคย ทำให้เข้าใจได้ง่ายมีลักษณะเหมือนแผนภูมิองค์กร โดยที่แต่ละโหนดจะแสดง attribute แต่ละกิ่งแสดงผลในการทดสอบและลีฟโหนดแสดงคลาสที่กำหนดไว้ อัลกอริธึมส์จะตัดสินใจในเงื่อนไขอย่างอัตโนมัติ โดยค่าที่ถูกจัดกลุ่มจะเป็นหัวข้อที่ตัดสินใจ เรียกว่า (Nodes) และหากโหนดนั้นเป็นโหนดสุดท้ายที่ถูกจัดกลุ่มจะเรียกว่า Leave

3.4.2 Neural Induction

เทคนิค Neural Induction จะนำเสนอตัวอย่างที่เป็นโครงสร้างของโหนดและน้ำหนัก (Weight) ของความสัมพันธ์ในการเชื่อมตัวนั้น ซึ่งเป็นพื้นฐานของเครือข่ายศูนย์กลางประสาท Neural Induction ซึ่งเป็นการร่วมกลุ่มของการเชื่อมต่อโหนดที่ได้ Input Output และการประมวลผลแต่ละโหนด ระหว่างค่า Input Output มีชั้นการประมวลผลที่ถูกซ่อนไว้ (Hidden processing layers)

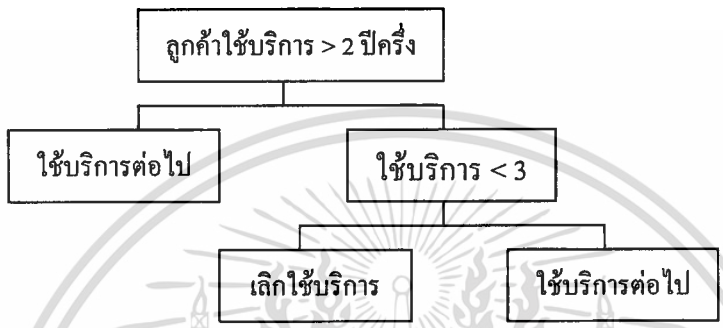
3.5 โอเปอเรชั่น ดาต้าไมนิ่ง (Data Mining Operations)

เป็นโอเปอเรชั่นใหญ่ ๆ ที่สนับสนุนโปรแกรมประยุกต์ทางธุรกิจและมีความสัมพันธ์กับดาต้าไมนิ่งคือ Predictive Model , Database Segmentation , Link Analysis และ Deviation Detection ดังนี้

3.5.1 Predictive Modeling

Predictive Modeling คล้ายกับการเรียนรู้จากประสบการณ์ของมนุษย์ การสังเกตรูปแบบพื้นฐานทั่ว ๆ ไป ภายใต้ลักษณะพิเศษของปรากฏการณ์ธรรมชาติต่าง ๆ เช่นเด็กเล็กจะเรียนรู้ลักษณะพิเศษ ความแตกต่างของสุนัข เพื่อชี้เฉพาะว่าสัตว์ที่พบใหม่เป็นสุนัข

ในดาต้าไมนิ่งเราใช้ Predictive Modeling ในการวิเคราะห์ข้อมูลที่มีอยู่จริงบนพื้นฐานของการตัดสินใจจากหลาย ๆ ลักษณะพิเศษของปัจจัยพื้นฐานเกี่ยวกับข้อมูล และแน่นอนว่าข้อมูลนั้นจะต้องมีความสมบูรณ์อย่างยิ่ง ตัวอย่างจะต้องบอกคำตอบที่ถูกต้อง และมีการแก้ปัญหาเรียบร้อยก่อนที่จะเริ่มสังเกตใหม่ต่อไป รูปแบบเป็นกลุ่มของกฎ IF THEN ในแบบอย่างที่ถูกต้อง เช่น บล็อกของ SQL หรือการจัดแบ่ง source code ของภาษาซี

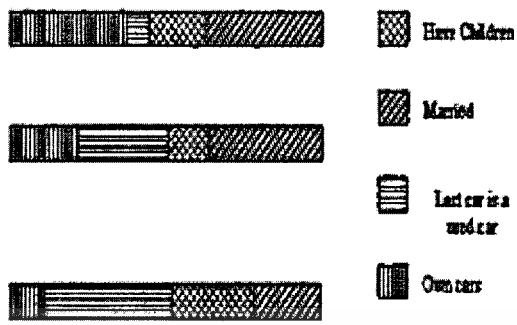


รูปที่ 3.3 แสดงตัวอย่าง Predictive Modeling

3.5.2 Database Segmentation

เป็นเทคนิคการลดขนาดของข้อมูลด้วยการรวมกลุ่มตัวแปรที่มีลักษณะเดียวกันไว้ด้วยกัน โดยหนังสือบางเล่มอาจใช้คำว่า Clustering เป็นธรรมดาว่าจะพบว่ามันเป็นกลุ่มย่อยในกลุ่มตัวอย่างที่เราสนใจในฐานข้อมูลลูกค้าในการปรับปรุงข้อมูลให้ละเอียดยิ่งขึ้น เมื่อฐานข้อมูลมีข้อมูลมากขึ้นจากกลุ่มตัวอย่างที่มีความหลากหลายของข้อมูล ซึ่งจำเป็นมากในการแบ่งกลุ่มและรวมกลุ่มของข้อมูลที่สัมพันธ์กันในแต่ละฐานข้อมูล ตัวอย่างเช่น บริษัทจำหน่ายรถยนต์ได้แยกกลุ่มลูกค้าออกเป็น 3 กลุ่ม ด้วยกันคือ

- กลุ่มผู้มีรายได้สูง (> 80,000)
- กลุ่มผู้มีรายได้ปานกลาง (25,000 to 80,000)
- กลุ่มผู้มีรายได้ต่ำ (less then 25,000)
-



รูปที่ 3.4 แสดงตัวอย่าง Database Segmentation

3.5.3 Link Analysis

Link Analysis จะพยายามค้นหาความสัมพันธ์ที่แน่นอนระหว่างเรคคอร์ดแต่ละอันหรือกลุ่มของเรคคอร์ดในฐานข้อมูล ความสัมพันธ์นี้เรียกว่า Associations โปรแกรมประยุกต์ที่ดีของ Link Analysis คือการค้นพบความสัมพันธ์กันระหว่างผลิตภัณฑ์ หรือบริการที่ลูกค้าได้รับมาพร้อมกับการซื้อสินค้า ตัวอย่างอื่น ๆ ที่โปรแกรมประยุกต์ทางธุรกิจรองรับเช่น การขายสินค้าเป็นแพ็กเก็ต

3.5.4 Deviation Detection

เป็นกรรมวิธีในการหาค่าที่แตกต่างไปจากค่ามาตรฐาน หรือค่าที่คาดคิดไว้ว่าต่างไปมากน้อยเพียงใด โดยทั่วไปมักใช้วิธีการทางสถิติ หรือการแสดงให้เห็นภาพ (Visualization) สำหรับเทคนิคนี้ใช้ในการตรวจสอบ ลายเซ็นปลอม หรือบัตรเครดิตปลอม รวมทั้งการตรวจหาจุดบกพร่องของชิ้นงานในโรงงานอุตสาหกรรม

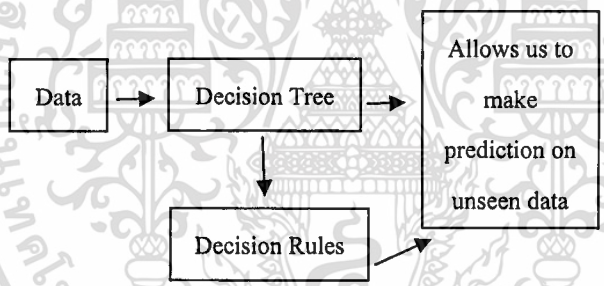
3.6 สรุป

จากรายละเอียดที่กล่าวมาข้างต้นเป็นการทำความเข้าใจเกี่ยวกับทฤษฎีดาต้าไมนิ่ง ทำให้สามารถเลือกเทคนิคและวิธีการที่เหมาะสมที่จะนำไปประยุกต์ใช้ในการพัฒนาระบบงาน คือ เทคนิคคลาสซิฟิเคชั่น โดยใช้แนวทางดิซิชั่นทรี ในการทำงาน

บทที่ 4
ดิซิชั่นทรี

4.1 ความหมายของดิซิชั่นทรี

Decision trees เป็น Flow-chart ที่มีโครงสร้างเป็น Tree ซึ่งแต่ละ 1 Internal node จะแสดงถึง Attribute ที่จะใช้ Test แต่ละกิ่ง (Branch) ของ Tree ซึ่งจะแสดงถึงผลของการทดสอบและ Left node จะแสดงถึง Class ส่วน Node สูงสุดของ Tree คือ Root node ในแต่ละกิ่งหรือโหนดจะมีทางเลือกให้ตัดสินใจ และแต่ละโหนดจะแสดง ข้อมูลที่แบ่งแยกเป็นประเภทหรือข้อมูลที่ได้ตัดสินแล้วเช่น เรามักจะใช้สถานการณ์ทางการเงินตัดสินใจในการให้กู้เงิน



รูปที่ 4.1 แสดงการทำงานของดิซิชั่นทรี

ดิซิชั่นทรีจะทำงานโดยสร้างกฎ ในรูปแบบโครงสร้างต้นไม้ โดยเงื่อนไขบนสุด (Root Node) จะถูกเปรียบเทียบก่อน คำตอบที่ได้จะอ้างถึงหนึ่งในหลายซับโหนดของต้นไม้ แต่ละการทดสอบจะระบุเงื่อนไขออกมาได้ การประมวลผลของการทดสอบเงื่อนไข และการอ้างถึงซับโหนดถูกวนซ้ำ จนกระทั่งเข้าถึงโหนดใบ (left node) ของต้นไม้ เพราะว่าการสร้างต้นไม้มีจุดมุ่งหมายคือ โหนดใบที่จะแสดงถึงประเภทของคลาสที่แบ่งแยกได้ ซึ่งคือความรู้หรือผลลัพธ์ที่ได้นั่นเอง(Uregina.1999:37)

4.2 การสร้างดิซิชั่นทรี (Decision Tree Building)

การสร้างดิซิชั่นทรีทำได้โดยใช้อัลกอริธึมที่อ้างอิงจาก Top-down Induction of Decision Trees (TDIDT) ใช้ induction เพราะความรู้ได้มาจากวิธีการเฉพาะจากกลุ่มข้อมูล ซึ่งอยู่ในรูปแบบของ top-down กฎที่ถูกเลือกกฎแรกให้เป็นโหนดเดี่ยวบนสุด (root node) และจากนั้นก็จะมีการวนทำซ้ำในส่วนนั้นๆ ต่อไปการแบ่งแยกจะสิ้นสุดถ้าสมาชิก ทุกตัวของกลุ่มจัดอยู่ในประเภทเดียวกันแล้ว หรือไม่มีเกณฑ์ที่จะลงทางซ้ายสุดแล้ว ในบางอัลกอริธึมจะหยุดการแบ่งแยกโดยอ้างอิงถึงการ pre pruning ถ้าประเภทข้อมูลนั้น ได้รับการปรับปรุงแล้วดูเหมือนไม่มีความหมาย อัลกอริธึมอื่นๆ จะแทนที่ซับทรีที่ไม่มีความหมายโดยโหนดใบหลังจากการประมวลผล ซึ่งเรียกว่า post pruning การนำข้อมูลมาสร้าง Decision Tree มีขั้นตอนพื้นฐานคือ

- หา Attribute ที่สำคัญที่สุดมาแบ่งข้อมูลโดย Attribute นี้ จะถูกนำมาสร้างเป็น Root node
- นำค่าที่เป็นไปได้ใน Attribute ที่ถูกเลือกแตกออกมาเป็นกลุ่ม
- แบ่งข้อมูลทั้งหมดตามกลุ่มที่แตกออกจาก Root node
- นำข้อมูลแต่ละกลุ่มมาทำซ้ำขั้นตอนแรกคือหา Attribute ที่สำคัญที่สุด

4.3 อัลกอริธึม Decision Trees

4.3.1 อัลกอริธึม C4.5

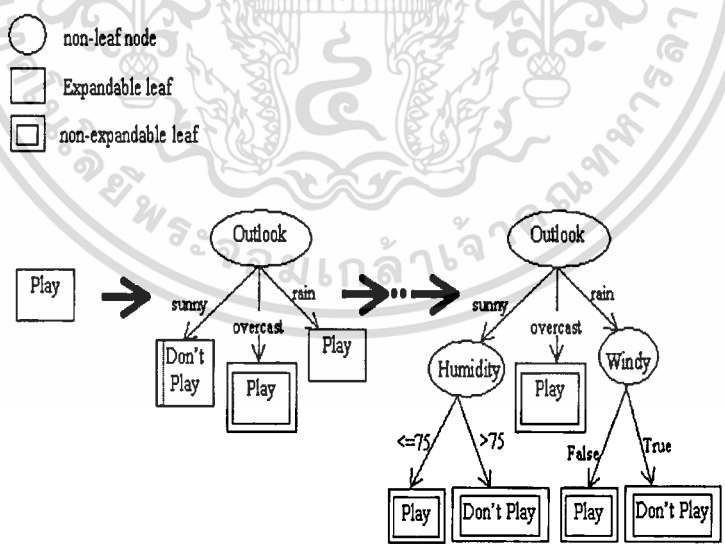
อัลกอริธึมนี้เกิดขึ้นในปี 1993 โดย Quinlan เป็นการพัฒนาเพิ่มจากอัลกอริธึม ID 3 โดยอ้างอิงเทคนิค classification-decision เพื่อเรียกซ้ำสำหรับ data-set เพื่อใช้จัดการข้อมูลที่ ID3 ไม่สามารถจัดการได้ เช่น

- หลีกเลี่ยงข้อมูลที่ Overfitting Determining how deeply to grow a decision tree
- ลดการพรุนนิ่งผิดพลาด (error pruning)
- เลือกแอททริบิวต์ที่เหมาะสมมาเป็นเครื่องมือในการคัดเลือก
- ตรวจจับการเทรนนิ่งข้อมูลด้วยค่าที่ไม่ถูกต้อง
- ตรวจจับ Attribute ที่เป็นข้อมูลต่อเนื่อง เช่น อุณหภูมิ
- ตรวจจับแอททริบิวต์ด้วยค่าที่แตกต่างกัน
- ปรับปรุงประสิทธิภาพการประมวลผลโดยอัลกอริธึม

ตารางที่ 4.1 แสดงตารางข้อมูลศึกษาการทำงาน C4.5

Outlook	Temp(°F)	Humidity(%)	Windy	Class
Sunny	75	70	True	Play
Sunny	80	90	True	Don't play
Sunny	85	85	False	Don't play
Sunny	72	95	False	Don't play
Sunny	69	70	False	Play
Overcast	72	90	True	Play
Overcast	83	78	False	Play
Overcast	64	65	True	Play
Overcast	81	75	False	Play
Rain	71	80	True	Don't play
Rain	65	70	True	Don't play
Rain	75	80	False	Play
Rain	68	80	False	Play
Rain	70	96	False	Play

จากรูปที่ 2 จะเห็นว่าประกอบด้วย 2 Class คือ Play และ Don't play โดยข้อมูลจำนวน 9 record อยู่ใน class Play และ 5 record อยู่ใน class Don't play



รูปที่ 4.2 แสดงขบวนการทำงานในลักษณะทรี

C4.5 จะสร้างดิซิชั่นทรีที่มีการจัดประเภทให้กลุ่มของข้อมูลได้วนแบบต่อเนื่องเพื่อแบ่งแยกข้อมูล โดยใช้วิธีการที่เรียกว่า Depth-first ซึ่งอัลกอริธึมนี้จะพิจารณาทดสอบความเป็นไปได้ในการแบ่งกลุ่มข้อมูลและเลือกทดสอบข้อมูลที่ดีที่สุด โดยในแต่ละแอตทริบิวต์ ผลการทดสอบจะได้ตัวเลขมากมาย ค่าที่มีความแตกต่างจะถูกนำมาพิจารณา อัลกอริธึม C4.5 ได้เพิ่ม Feature ที่อัลกอริธึม ID3 ไม่มี ดังนี้

- Gain ratio criterion พัฒนาขึ้นเพื่อแก้ปัญหาของ Gain Criterion กรณีที่ Attribute มีค่าที่ unique การแบ่งข้อมูลโดยใช้ Attribute นี้อาจทำให้เกิด Subset จำนวนมากซึ่งแต่ละ Subset จะประกอบไปด้วยข้อมูลเพียง 1 record เท่านั้น ทำให้ $info_x(T) = 0$ ซึ่งจะมีผลให้ค่า information Gain ของ Attribute นี้มีค่าสูงมากและการแบ่งข้อมูลโดยใช้ Attribute นี้ไม่ก่อให้เกิดประโยชน์ใดๆ ต่อการทำนาย C4.5 แก้ไขโดยใช้ค่า Gain ratio ซึ่งคำนวณโดยใช้ split info(x) และ gain ratio(x) โดย split info(x) เป็นค่า information ที่ได้จากการแบ่ง T ออกเป็น n subset

$$\text{Split Info}(x) = -\sum_{i=1}^{n} \frac{|T_i|}{|T|} \log_2 \frac{|T_i|}{|T|}$$

gain ratio(x) เป็นการวัดว่า การแบ่งข้อมูลโดยใช้ Attribute นั้น ก่อให้เกิดประโยชน์ต่อการทำนายหรือไม่

gain ratio(x) = gain(x)/split info(x)

ซึ่งการใช้ gain ratio ทำให้ tree ที่ได้มีขนาดเล็กกว่าการใช้ gain criterion

- Unknown attribute values

  - หาค่า Info(T) และ $Info_x(T)$ โดยพิจารณาจากข้อมูลที่รู้ค่าของ A

  - การหาค่า gain(x) โดยgain(x) = probability A is know x(info(T) – $info_x(T)$)

  - หาค่า split info(x) โดยพิจารณาจากลุ่มของข้อมูลที่ไม่รู้ค่าของ A เป็นอีก 1 subset เช่น ถ้า attribute ที่จะนำมาทดสอบมีค่าที่เป็นไปได้ n ค่า split info(x) จะถูกคำนวณโดยแบ่งข้อมูลออกเป็น n+1 subset

- การแบ่ง Training set สมมุติ attribute ที่เลือกจากขั้นตอนแรก มีค่าที่เป็นไปได้คือ O1,O2,...,On เมื่อข้อมูล 1 record ใน T ซึ่งมีค่า O_i ถูกกำหนดให้ subset T_i ค่าความน่าจะเป็นที่ข้อมูลนี้อยู่ใน subset T_i เท่ากับ 1 และความน่าจะเป็นที่ข้อมูลนี้อยู่ใน subset อื่น ๆ เท่ากับ 0 แต่ถ้าใน attribute ไม่ทราบค่า ความน่าจะเป็นจะมีค่าน้อยลง สำหรับข้อมูลแต่ละ record ในแต่ละ subset T_i weight จะเท่ากับความน่าจะเป็นของ O_i ที่จุดนั้น ๆ ทำให้ T_i เป็นผลรวมของค่า weight w ซึ่งค่าใน attribute ไม่ทราบค่าจะถูกกำหนดให้แต่ละ subset T_i ด้วย weight

W x probability of outcome O_i

โดยความน่าจะเป็นคือ ผลรวมของ weight ของข้อมูลทั้งหมดใน T ซึ่งมีค่า O_i หารด้วยผลรวมของ weight ของข้อมูลทั้งหมดใน T ซึ่งค่าใน Attribute เป็นค่าที่ทราบมาก่อน

- การใช้ decision tree ที่ได้มาทำนายกลุ่มของข้อมูล ในกรณีที่ค่าใน attribute ที่จะทดสอบที่ decision node เป็นค่าที่ไม่ทราบค่า ทำให้ไม่สามารถแบ่งข้อมูลได้ กรณีนี้ระบบจะสำรวจทุกเส้นทางที่เป็นไปได้ และรวมผลที่ได้จากการ classification ด้วยวิธีการทางคณิตศาสตร์ โดยผลที่ได้จะเกิดได้จากหลายเส้นทางจาก root ของ tree ไปยัง left node และ class ที่ได้จากการทำนายจะเป็น class ที่มีความน่าจะเป็นสูงสุด(Quinlan.2000:97)

ต่อไปจะนำเสนอตัวอย่างโดยใช้รูปที่ 2 เพื่อหา Attribute เพื่อใช้แบ่งข้อมูล

- การหา Attribute เพื่อใช้แบ่งข้อมูล สมมุติว่าค่าใน Attribute outlook ใน record ที่ 6 เป็นค่าที่ไม่ทราบค่า ซึ่งแทนโดย "?" ซึ่งเราจะพิจารณาเฉพาะข้อมูล 13 record ที่เหลือจะได้ความถี่ดังแสดงในรูปที่ 2 ทำการคำนวณค่าต่าง ๆ โดยพิจารณา Attribute Outlook ดังนี้

ตารางที่ 4.2 แสดงความถี่ของข้อมูล

	Play	Don't Play	Total
Outlook = sunny	2	3	5
Overcast	3	0	3
Rain	3	2	5
Total	8	5	23

$Info(T) = -8/13 x \log_2(8/13) - 5/13 x \log_2(5/13)$

$= 0.9691$

$Info_x(T) = 5/13 x(-2/5 x \log_2(2/5) - 3/5 x \log_2(3/5))$

$+3/13 x(-3/3 x \log_2(3/3) - 0/3 x \log_2(0/3))$

$+5/13 x(-3/5 \log_2(3/5) - 2/5 x \log_2(2/5))$

$= 0.747$

$gain(x) = 13/14 x(0.961 - 0.747)$

$= 0.199$

- Continuous attribute values สมมุติ A เป็น attribute

ชนิด continuous numeric value การทดสอบค่าที่ attribute นี้จะแบ่งเป็น $A \leq Z$ และ $A > Z$ โดยทำการเปรียบเทียบค่าของ A กับค่า Threshold value Z โดยการหาค่า Threshold ที่เหมาะสมมีขั้นตอนดังนี้

- เรียงลำดับ Training set ด้วยค่าใน Attribute A จากน้อยไปมาก และเลือกเฉพาะค่าไม่ซ้ำกันมาพิจารณาจะได้ { v1 , v2 ,..., vn }

- หาค่า Threshold ใดๆ ซึ่งค่า Threshold ใดๆ จะอยู่ระหว่าง V1 และ Vi+1 โดยคำนวณจาก Midpoint ของแต่ละช่วงดังนี้ Vi+Vi+1/2 โดย C4.5 จะเลือกค่าที่มากที่สุดใน Attribute A แต่ต้องไม่เกินค่า Midpoint นั้น ๆ จาก training Set เป็นค่า Threshold ของแต่ละช่วง เพื่อที่ว่าค่า Threshold ทั้งหมดที่ปรากฏอยู่ใน Tree หรือ Rule จะเป็นค่าที่เกิดขึ้นจริงในข้อมูล

- หาค่า Threshold ที่เหมาะสม โดยพิจารณาจากค่า Threshold ที่มีค่า Information Gain สูงสุด

- Pruning decision tree การแบ่งข้อมูลใน Training set

เพื่อสร้าง Decision Tree จะทำไปจนกระทั้งข้อมูล ในแต่ละ Subset อยู่ใน class เดียวกัน ซึ่งผลลัพธ์ที่ได้อาจทำให้ Tree มีความซับซ้อนมากเกินไปที่เรียกว่า "Overfits the data" ซึ่งปัญหานี้ สามารถทำการแก้ไขได้โดยทำการ Pruning จะทำให้แต่ละ left node ที่ได้ไม่จำเป็นที่จะต้องประกอบด้วยข้อมูลที่อยู่ใน class เดียวกันทั้งหมด โดยแต่ left node จะมีการระบุการกระจายของข้อมูลแต่ละ class ไว้ ซึ่งจะบอกถึงความน่าจะเป็นที่ข้อมูลจะอยู่ใน class นั้นๆ อัลกอริธึมของ C4.5 จะทำการ Pruning โดยการตัด Sub tree นั้นด้วย left node โดยเทคนิคนี้จะใช้เพียงข้อมูลใน training set ที่ใช้ในการสร้าง tree เท่านั้น และการคำนวณความผิดพลาดที่เกิดจากการทำนาย ของแต่ละ left node และ sub tree จะทำโดยสมมุติว่า จะทำการแบ่งกลุ่ม set ของข้อมูลที่ไม่เคยพบมาก่อนที่มีขนาดเท่ากับ training set โดยการคำนวณจะใช้ function ทางสถิติซึ่งอยู่บนพื้นฐานของการกระจายแบบ Binomial จำนวน Error ที่เกิดขึ้น เมื่อข้อมูลมีขนาดเท่ากับ N

$= N \times U_{CF}(E,N)$

โดย N แทน ขนาดของข้อมูลที่ left node ใดๆ

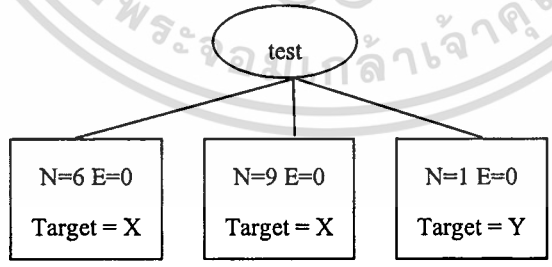
E แทน จำนวนของ error ที่เกิดขึ้นใน set ของข้อมูล

ที่ left node ใดๆ

$U_{CF}(E,N)$ แทน ความน่าจะเป็นสูงสุดที่จะเกิด error

และ C4.5ใช้ Confidence level เท่ากับ 0.25 หรือ 25%

ต่อไปจะอธิบายการ pruning โดยพิจารณา sub tree ดังรูป



รูปที่ 4.3 แสดง Sub tree ก่อนทำการ Pruning

จากรูปจะพบว่าค่าที่เป็นไปได้ที่เกิดจากกการทดสอบมี 3 ค่า คือ A , B และ C และ Target attribute มี 2 ค่าคือ X และ Y ซึ่งในกรณีไม่พบ error ที่เกิดขึ้นใน Training set ใน left node ที่ 1 พบว่า N = 6 และ e=0 ดังนั้น

$U_{25\%}(0,6) = 0.206$

ถ้าเราใช้ left node นี้ในการแบ่งข้อมูลจำนวน 6 record จำนวน error ที่เกิดขึ้นในการทำนายจะเท่ากับ 6 x 0.026 สำหรับ left node ที่ 2 และ 3 จะได้ $U_{25\%}(0,9) = 0.143$ และ $U_{25\%}(0,1) = 0.750$ ตามลำดับ ดังนั้นจำนวน error ที่เกิดจากการทำนายของ sub tree นี้เท่ากับ

$6x0.206+9x0.143+1x0.750 = 3.273$

ถ้าทำการแทนที่ sub tree นี้ด้วย left node ที่มี Target = X เมื่อ X เป็นค่าที่มีความถี่มากที่สุดของ Target Attribute ของ Training subset จำนวน 16 record จะเกิด error และจำนวน error ที่เกิดจากการทำนายเท่ากับ

$16xU_{25\%}(1,16) = 16x0.157 = 2.512$

จะพบว่า sub tree นี้มีจำนวนของ error ที่เกิดจากการทำนายสูงกว่า ดังนั้นจึงทำการ Pruning โดยแทนที่ด้วย left node(Quinlan,1993:103)

4.3.2 อัลกอริธึมส์ ID3

ผู้พัฒนาอัลกอริธึมส์ ID3 คือ Ross Quinlan ในปี 1986 เป็นดิซิชั่นทรี ที่สร้างอัลกอริธึมส์ที่จัดกลุ่มและตัดสิน ประเภทของอ๊อปเจ็ค โดยการทดสอบค่าคุณสมบัติของอ๊อปเจ็คเหล่านั้น ซึ่งเป็นการสร้างทรีแบบบนลงล่าง (Top-down approach) การทำงานเริ่มที่อ๊อปเจ็ค กลุ่มที่กำหนดและระบุรายละเอียดของคุณสมบัติ ในแต่ละโหนดของต้นไม้นั้นคุณสมบัติจะถูกทดสอบและใช้ผลลัพธ์ที่ได้ในการแบ่งส่วนกลุ่มของอ๊อปเจ็ค การประมวลผลจะวนกลับทำใหม่จนกระทั้งไดซับทรีที่ประกอบขึ้นจากเกณฑ์เดียวกัน จนในที่สุดจะได้โหนดใบออกมา ซึ่งแต่ละโหนดนี้คุณสมบัติที่ถูกเลือกมาทดสอบจะอยู่บนพื้นฐานของข้อมูลที่เป็นเหตุเป็นผลกัน ซึ่งนำไปสู่ข้อมูลที่มีประโยชน์สูงสุด และมี entropy น้อยสุด

บทที่ 5
เครื่องมือและวิธีการที่ใช้ในการพัฒนาระบบ Data Mining

5.1 เครื่องมือในการทำดาต้าไมน์นิ่ง (Data Mining Tools)

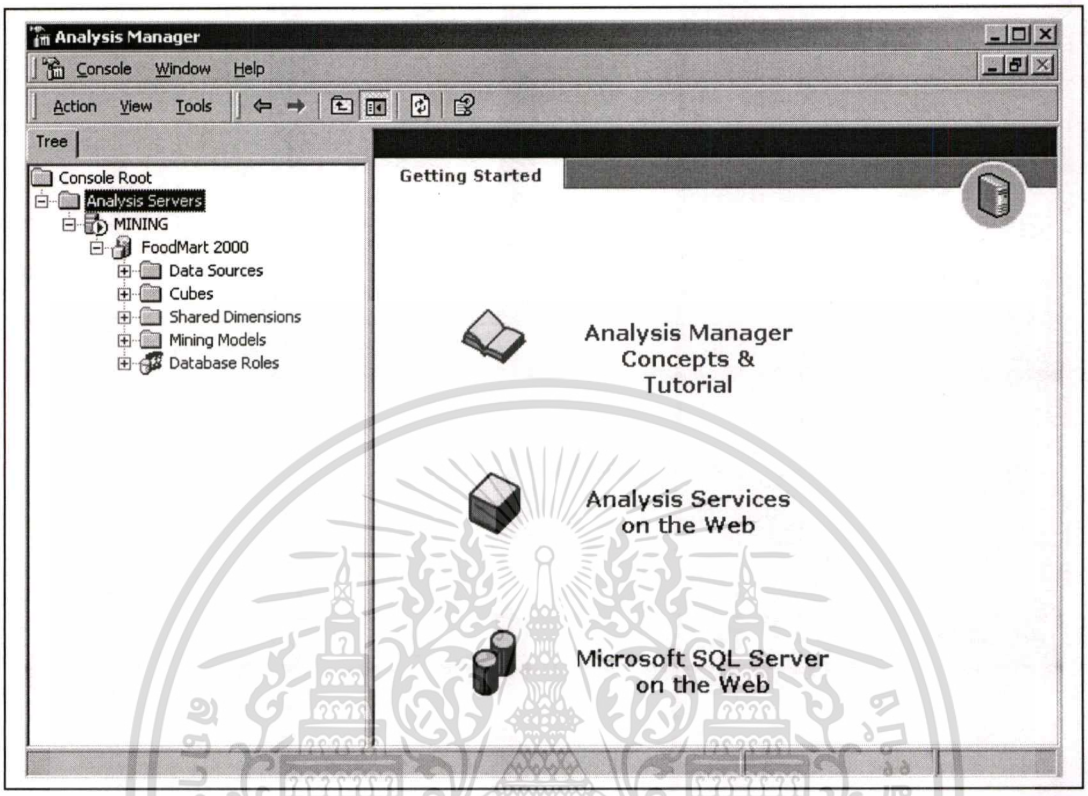
ในปัจจุบันได้มีการพัฒนาเครื่องมือต่าง ๆ มาใช้ในการทำ ดาต้าไมน์นิ่ง ขึ้นมาอย่างมากมายในโครงการฉบับนี้จะกล่าวอ้างอิงถึงเครื่องมือตัวหนึ่งที่ถูกพัฒนา ขึ้นมาจากรุ่นเดิม ซึ่งสามารถจัดการกับฐานข้อมูลขนาดใหญ่ ได้เท่านั้น ให้มีความสามารถในการทำไมน์นิ่งได้ และกำลังเริ่มเป็นที่นิยมเป็นที่แพร่หลายในปัจจุบัน เครื่องมือตัวนั้นก็คือ Microsoft SQL Server 2000 เป็นรุ่นถัดมาของ SQL Server เป็นระบบฐานข้อมูลที่แข็งแกร่ง ซึ่งนอกจากความสามารถทางด้านของ RDBMS (Relational Database Management System) ตามปกติแล้วยังสามารถสอบถาม (Query) วิเคราะห์ ตลอดจนจัดการข้อมูลผ่านเว็บ ด้วยการสนับสนุน XML ช่วยให้การจัดการข้อมูลทั้งแบบ OLTP (Online Transaction Processing) และได้สร้างโซลูชั่น OLAP (Online Analytical Processing) คลังข้อมูล และ Data Mining เพื่อใช้ประโยชน์จากฐานข้อมูลได้อย่างง่ายดาย และมีประสิทธิภาพสูงสุด(สมพร จิวรสกุล.2545:4)

ซึ่งเมื่อพิจารณาความเกี่ยวข้องของเครื่องมือกับรูปแบบโครงสร้างสถาปัตยกรรมแล้วสามารถแบ่งขั้นการทำงานได้ดังนี้

- ขั้นการจัดการกับข้อมูลในการประมวลผลเชิงออนไลน์
- ขั้นตอนในส่วนการแสดงต่อผู้ใช้งาน

5.1.1 ขั้นการจัดการกับข้อมูลในการประมวลผลเชิงออนไลน์

การพัฒนาระบบนั้นได้ใช้เครื่องมือที่มีมากับ Microsoft SQL Server 2000 เป็นตัวจัดการกับข้อมูลที่จะนำมาทำการวิเคราะห์ เพื่อให้อยู่ในรูปแบบ Cube ดังนั้นเครื่องมือที่สำคัญอย่างแรกคือ Analysis Manager



รูปที่ 5.1 เครื่องมือในการพัฒนาระบบ Analysis Manager

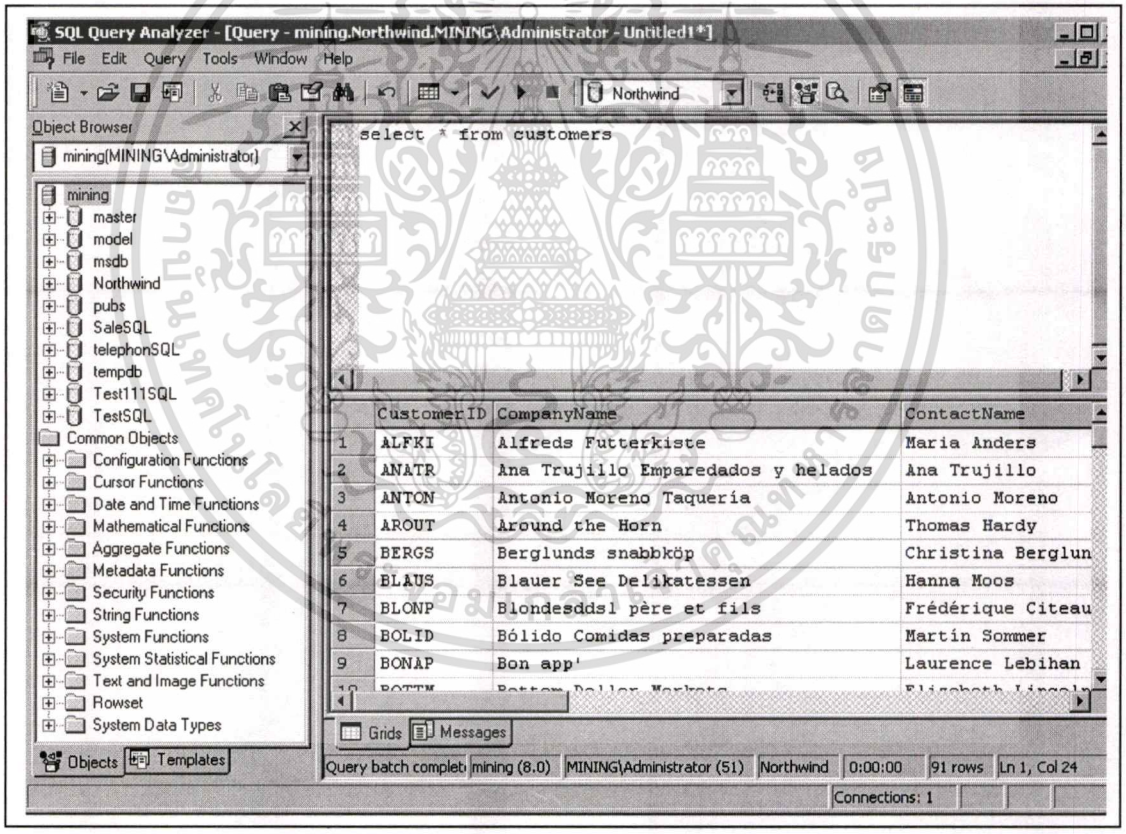
ในส่วนนี้ทำหน้าที่เป็นตัวหลักในการจัดการกับข้อมูล โดยมีงานหลักที่เกี่ยวข้องกับการพัฒนาระบบดังนี้

- Server เป็นส่วนที่ใช้ในการจัดการสร้างฐานข้อมูล ที่จะนำข้อมูลมาทำการวิเคราะห์ หลังจากทำการสร้างฐานข้อมูลเสร็จก็จะมีเครื่องมือต่าง ๆที่จะช่วยในการวิเคราะห์ข้อมูลเพิ่มเข้ามา อาทิ
  - Data Sources
  - Cubes
  - Shared Dimensions
  - Mining Models

ซึ่งเป็นโปรแกรมส่วนประกอบเสริมของ Microsoft SQL Server 2000 สำหรับ Analysis Services จะมี Analysis Manager เป็นเครื่องมือทำหน้าที่ในการสร้างลูกบาศก์เพื่อใช้ในการวิเคราะห์ข้อมูล ซึ่งขั้นตอนในการทำงานโดยสรุปมีดังนี้ ในขั้นตอนแรกต้องทำการสร้างฐานข้อมูลลูกบาศก์ และทำการสร้าง Data Source เพื่อติดต่อระหว่างฐานข้อมูลคลังข้อมูล กับฐานข้อมูล

ลูกบาศก์ จากนั้นจะทำการสร้างลูกบาศก์ และมุมมอง เพื่อวิเคราะห์ข้อมูลภายหลังจากการสร้างลูกบาศก์เสร็จสิ้นแล้วต้องทำการสร้างที่เก็บข้อมูลของลูกบาศก์ โดยในที่นี้ได้เลือกรูปแบบเป็นลักษณะ MOLAP เพื่อให้สามารถที่จะตอบสนองต่อการทำงานได้อย่างรวดเร็ว ซึ่งเมื่อเสร็จสิ้นขั้นตอนนี้แล้วก็จะสามารถดูผลวิเคราะห์ข้อมูลต่างๆได้

จากที่กล่าวมานั้นเป็นเครื่องมือในการพัฒนาระบบที่อยู่ใน Analysis Manager นอกจากนั้นแล้วยังมี SQL Query Analyzer ซึ่งเป็นเครื่องมือที่สำคัญที่อยู่ใน SQL Server ใช้สำหรับในการทดสอบคำสั่งในการทำงานซึ่งเป็นคำสั่งภาษา SQL (Structure Query Language) ซึ่งเป็นภาษาที่ใช้ในการเรียกดูข้อมูลจากฐานข้อมูลเชิงสัมพันธ์ ทำให้สามารถดึงข้อมูลในการถ่ายโอนข้อมูลระหว่างฐานข้อมูลได้(พรพิมล อนันควานิช.2545:203)



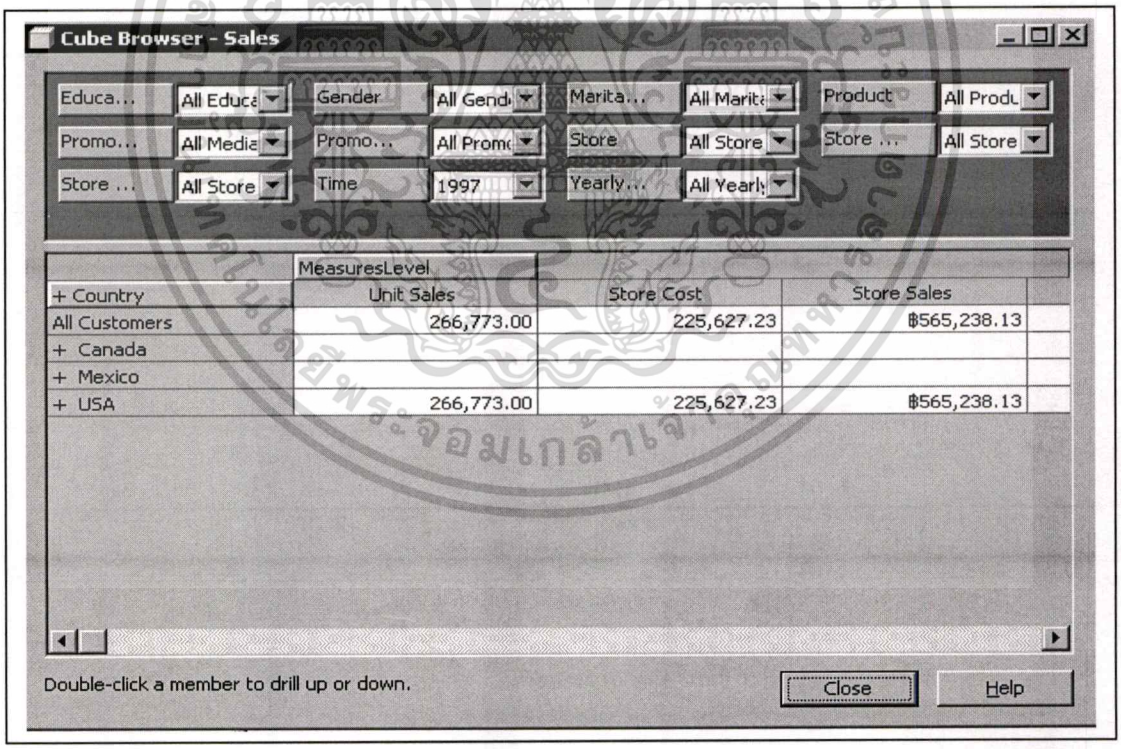
รูปที่ 5.2 เครื่องมือในการพัฒนาระบบ SQL Query Analyzer

นอกจากนั้นไมโครซอฟท์ยังมีแนวความคิดอยู่ว่า จะทำให้การออกแบบระบบการจัดการคลังข้อมูล การทำ Data Mining เป็นเรื่องง่ายใน SQL Server โดยเพิ่มกระบวนการทำงานต่าง ๆ เข้ามาผ่านทาง Analysis Service อาทิเช่น

- **OLAP Cube** จากความสัมพันธ์ของตาราง Fact Table และ ตาราง Dimension Table แบบ Star Schema นั้นสามารถนำไปสร้างเป็น OLAP Cube ได้ซึ่งการสร้าง Cube ก็คือการรวบรวมข้อมูลกลุ่มเดียวกันเข้าด้วยกัน แล้วเก็บผลลัพธ์ที่ได้เอาไว้ ทำให้ลดเวลาในการ Query ข้อมูลเนื่องจากไม่ต้องเสียเวลาคำนวณใหม่ ซึ่งภายใน OLAP Cube ประกอบไปด้วย
  - Dimension คือโครงสร้างของ Cube ถ้ามองง่าย ๆ ก็คือด้านต่าง ๆ ของ Cube นั้นเอง โดยจำนวน Dimension ของ Cube อาจจะเท่ากับจำนวน table ของ Star Schema หรือไม่ก็ได้ ทั้งนี้ขึ้นอยู่กับการออกแบบ Cube
  - Member คือสมาชิกของ Dimension ซึ่งชื่อของ Member จะเป็นข้อมูลที่อยู่ภายใน Dimension Table ใน Star Schema และ Member เป็นส่วนสำคัญในการเข้าถึงข้อมูลภายใน Cube
  - Level เป็นการแบ่ง Member ออกเป็นลำดับขั้น ภายในแต่ละ Dimension นั้น ๆ
  - Hierarchy คือความสัมพันธ์แบบ Parent – Child ภายในแต่ละ Dimension
  - Measurement เป็นข้อมูลประเภทตัวเลขที่จะนำมาวิเคราะห์ซึ่ง Measurement นี้จะอยู่ใน Fact Table ใน Star Schema ถ้ามองในมุมมองของ Cube แล้ว Measurement ก็คือค่าต่าง ๆ ที่ถูกบรรจุอยู่ภายในช่องต่าง ๆ ภายใน Cube นั้นเอง(สมพร จิวรสกุล. 2545:703)
- **การสร้างและจัดการ Cube** วิธีหนึ่งที่ทำให้ระบบ OLAP ประสบความสำเร็จในเรื่องประสิทธิภาพของการเรียกค้นที่ซับซ้อนก็คือ การออกแบบกลุ่มข้อมูลไว้ล่วงหน้าจำนวนหนึ่ง ซึ่งในการปรับแต่งประสิทธิภาพในการเรียกค้น จำเป็นต้องจัดกลุ่มข้อมูลเหล่านี้ก่อน การจัดกลุ่มข้อมูลนั้น ที่จริงก็คือการเพิ่มข้อมูลลงไปใน Cube ซึ่งอาจจะไปถึงจุดที่กลุ่มข้อมูลมีปริมาณมากกว่าข้อมูลเริ่มต้น ปัญหานี้เรียกว่า การระเบิดของข้อมูล (Data Explosion) ซึ่งการจัดการกลุ่มเหล่านี้จะต้องมีการคิดอย่างรอบคอบ

Analysis Manager ทำให้การจัดการกลุ่มข้อมูลที่ชาญฉลาด โดยทำให้กระบวนการจัดการระบบ OLAP ง่ายขึ้น ลดการจัดเก็บข้อมูลและกลุ่มโดย

- ให้การจัดกลุ่มข้อมูลล่วงหน้าที่ชาญฉลาด ในการเลือกทำเฉพาะกลุ่มที่มี ความสำคัญสูงสุด เช่น กลุ่มซึ่งเป็นฐานในการคำนวณให้กับกลุ่มข้อมูลอื่น
- ติดตามการเรียกค้นซึ่งทำให้ผู้บริหารระบบสามารถปรับแต่งการจัดกลุ่มข้อมูลได้ตรงกับการเรียกค้นที่เกิดขึ้นกับ Cube อย่างแท้จริง
- ช่วยบ่งชี้ให้เห็นถึงผลดีผลเสียระหว่างปริมาณเนื้อที่จัดเก็บกลุ่มข้อมูลกับประสิทธิภาพที่ได้รับ และยอมให้ผู้บริหารระบบเลือกและปรับแต่งให้สมดุลได้ซึ่งส่วนใหญ่จะมีจุดที่ทำให้เกิดประสิทธิภาพสูงสุดโดยไม่เพิ่มเนื้อที่ในการจัดเก็บซึ่งหากอยู่ในระดับที่ต่ำกว่านี้ จะแสดงให้เก็บว่ากำลังเพิกเฉยต่อการปรับปรุงประสิทธิภาพ และ ณ ระดับที่สูงกว่า ก็หมายถึงกำลังใช้เนื้อที่จัดเก็บข้อมูลมากเกินไป เพื่อให้ได้มาซึ่งประสิทธิภาพที่เพิ่มขึ้นเพียงเล็กน้อย



รูปที่5.3 แสดงมุมมองลูกบาศก์ที่ได้จากการประมวลผล

- **MDX – Multidimensional Expressions Language** MDX เป็นคำสั่ง Query Language ประเภท DML – Data Manipulation language ซึ่งเป็นส่วนหนึ่งใน Microsoft SQL Server 2000 Analysis Service มีโครงสร้างของคำสั่งและหน้าที่การทำงานคล้ายกับ SQL แต่ส่วนที่ต่างกัน คือ MDX สามารถดึงข้อมูลมาจาก Multidimensional Database หรือ OLAP Cube ได้ทันที เพราะมีโครงสร้างของภาษาที่ออกแบบมาให้ทำหน้าที่แบบนี้อยู่แล้ว และข้อมูลที่ได้มาอาจยังคงอยู่ในสภาพที่เป็นลักษณะหลายมิติก็ได้ ในขณะที่ข้อมูลที่ได้จาก SQL จะมีเพียงสองมิติ คือ Column และ Row เท่านั้น ดังนั้นในคำสั่ง MDX จึงต้องมีการระบุ Dimension ให้กับข้อมูลที่ได้รับมาด้วยเช่น Row หรือ Column หรือ Page

  รูปแบบคำสั่ง MDX มีโครงสร้างคล้ายกับ SQL โดยมีลักษณะดังนี้

  SELECT [<axis_specification> [,<axis_specification>…]]

  FROM [<cube_specification>]

  [WHERE [<slicer_specification>]]

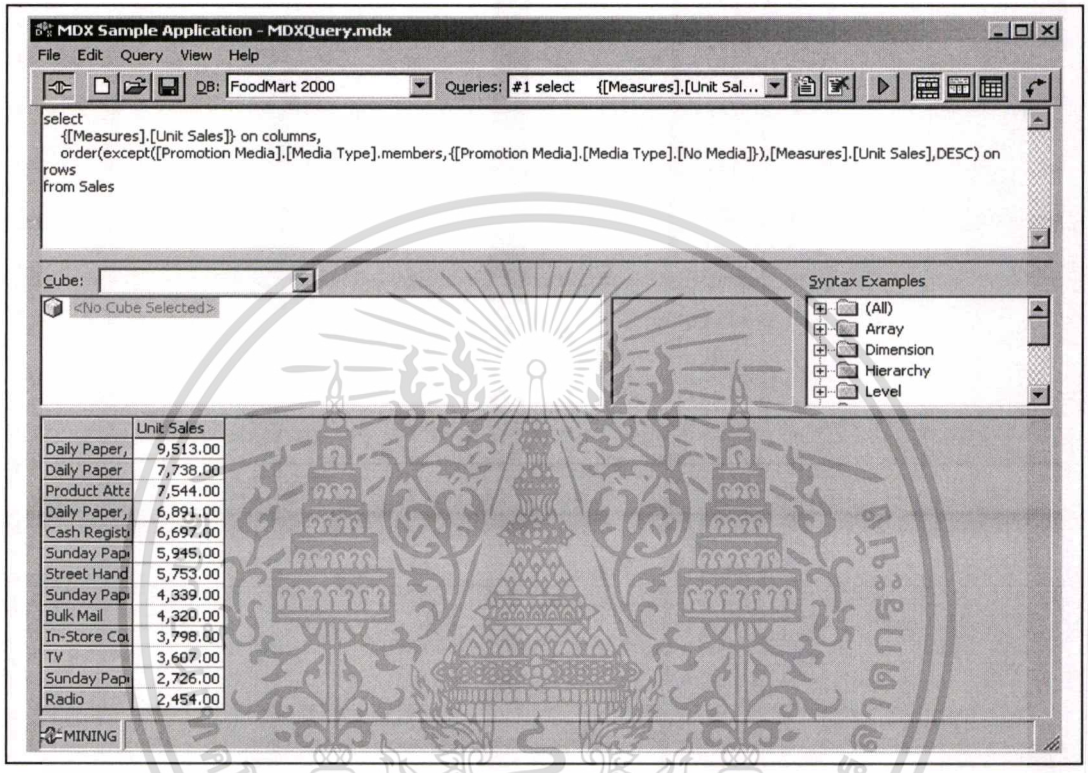
  โดยที่

  - axis_specification เป็นการเลือก member หรือเซตของ member ซึ่งตามที่ได้กล่าวไปแล้วว่าจะต้องระบุ Dimension ให้กับ member ที่เลือกด้วย ดังนั้น รูปแบบของ axis_specification จะต้องกำหนดด้วยว่า on row หรือ on column ไว้ด้วยเสมอ เช่น [Product].[Food] on column เป็นต้น ดังนั้น axis_specification จะมีรูปแบบคือ <set> on <axis_name> ซึ่ง set ก็คือเซตของ member และ axis_specification สามารถระบุโดยชื่อ ได้แก่ column , row , page , sections , chapter หรือระบุเป็น index เช่น axis(0) , axis(1) , axis(2) ไปเรื่อย ๆ ก็ได้ ซึ่ง column คือ axis(0) , row คือ axis(1) และ page คือ axis(2) นั้นเอง ใน MDX นี้ยอมให้มี axis ได้ทั้งหมด 64 axis แต่สามารถระบุแบบชื่อได้ถึง chapter เท่านั้น หลังจากนั้นต้องระบุเป็นแบบ index และในการเขียนควรเรียงจาก index น้อยไปหามากเพื่อป้องกันการสับสน

  - cube_specification คือชื่อของ cube ที่ต้องการ เหมือนกับการระบุชื่อ table ในคำสั่ง SQL

  - slicer _specification เป็นการระบุเงื่อนไขในการเลือกข้อมูล เหตุที่ใช้คำว่า slicer เนื่องจากโครงสร้าง cube เป็นลักษณะ multidimensional เมื่อระบุเงื่อนไขใน dimension หนึ่งแล้ว ก็เหมือนกับว่าข้อมูลที่ได้มานี้จะอยู่ภายใต้เงื่อนไขของ dimension นี้เท่านั้น ทั้งนี้ dimension ถูกระบุใน axis_specificational แล้วจะไม่สามารถนำมาใช้ใน slicer_specification ได้อีก หรือกล่าวอีกนัยหนึ่งคือ axis_specificational และ slicer_specification จะต้องเป็น dimension ที่ต่างกันเสมอ

คำสั่ง MDX มีความสำคัญสำหรับการพัฒนาโปรแกรมสำหรับวิเคราะห์ข้อมูลทาง Data Mining เนื่องจากโปรแกรมดังกล่าวจะมีการนำสิ่งต่าง ๆ ที่ผู้ใช้ต้องการ มาสร้างเป็นคำสั่ง MDX เพื่อส่งไปขอข้อมูลจาก OLAP Server แล้วนำข้อมูลมาแสดงในโปรแกรมต่อไป

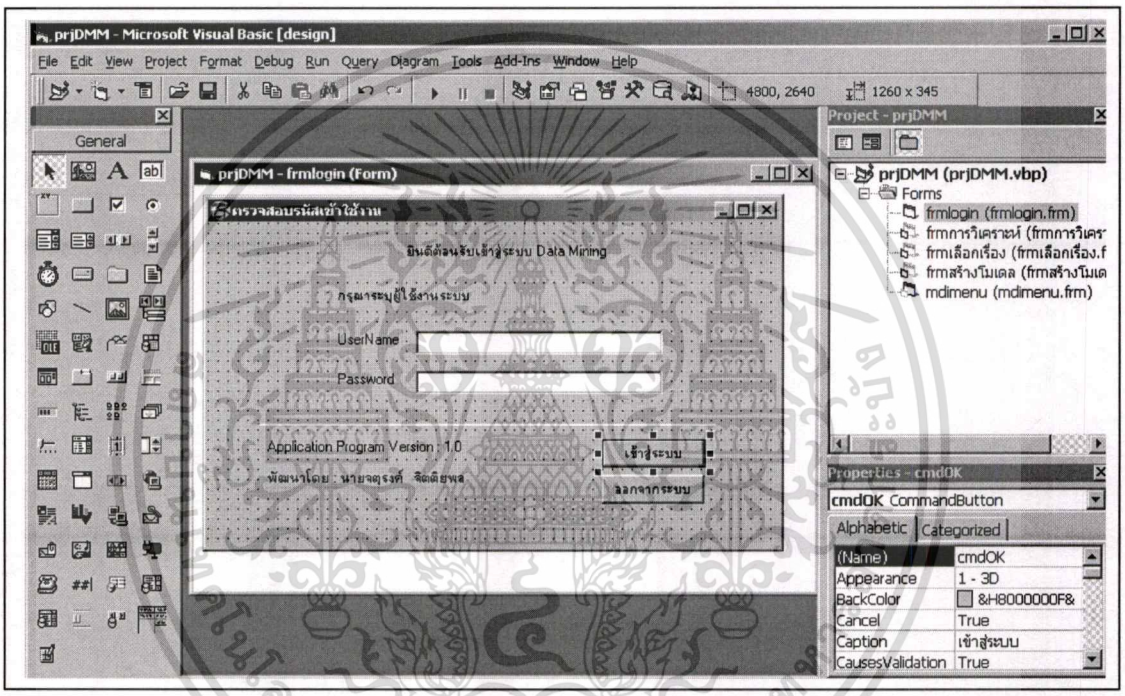


รูปที่5.4 เครื่องมือในการพัฒนาระบบ MDX Sample Application

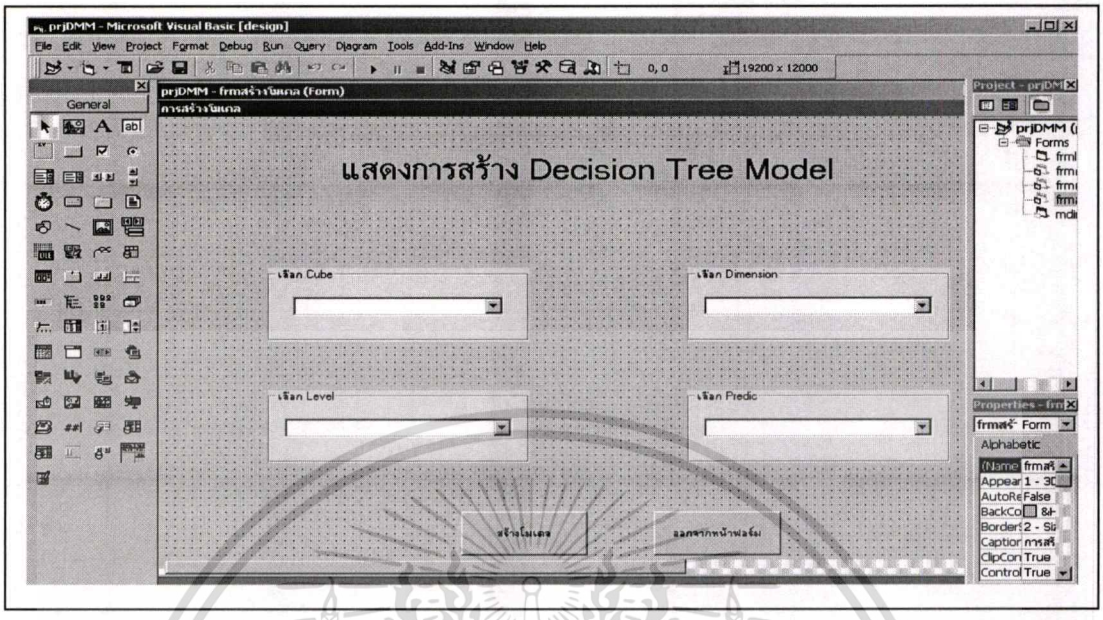
- **ADO-MD Object Library** ADO-MD (ActiveX Data Object Multidimensional) เป็น Object Library สำหรับใช้ติดต่อกับ OLAP Server ของ Microsoft ® SQL Server™ 2000 Analysis Service ซึ่งมีประโยชน์มากสำหรับการพัฒนา Application ในเชิงวิเคราะห์ เนื่องจากสามารถดึงข้อมูลจาก OLAP Cube ได้โดยตรงไม่ว่าจะเป็นโครงสร้างของ Cube (Cube Schema) หรือข้อมูลภายใน Cube (Cube Data)

5.1.2 ขั้นตอนในส่วนการแสดงต่อผู้ใช้งาน

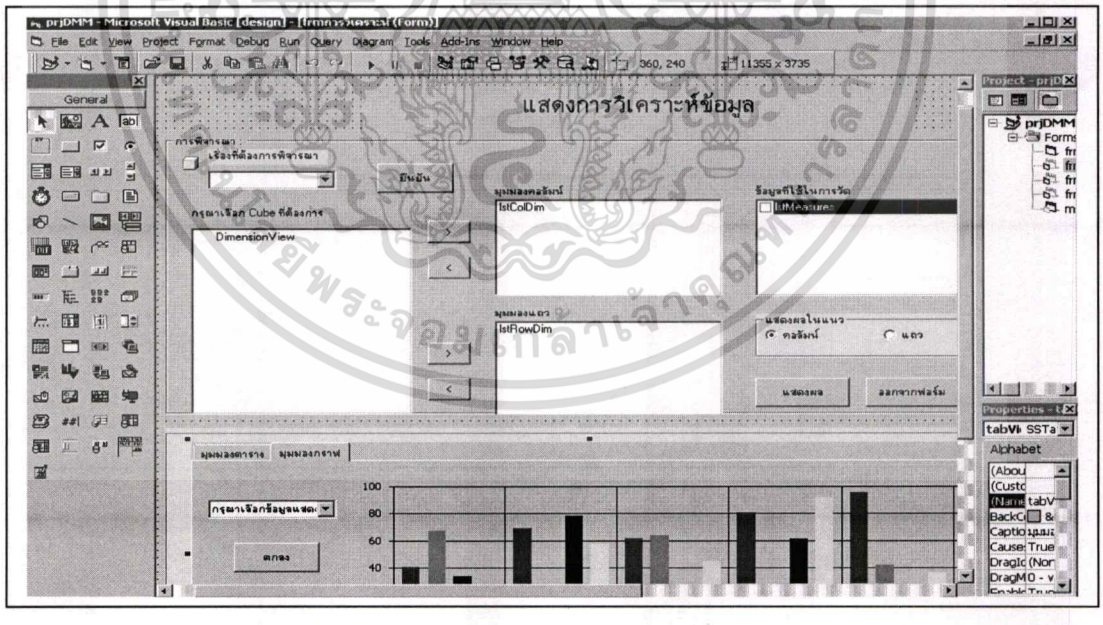
สำหรับส่วนแสดงผลข้อมูลผู้ใช้งานนั้น ได้ทำการพัฒนาด้วยโปรแกรม Microsoft Visual Basic 6 โดยกำหนดมีลักษณะการนำเสนอข้อมูลที่ผ่านการประมวลผลในรูปของตาราง และแผนภูมิแท่ง นอกจากนั้นยังได้เพิ่มความสามารถโดยการส่งข้อมูลที่พิจารณาให้อยู่ในรูปของเอกสารตารางทำการ Microsoft Excel อีกด้วย



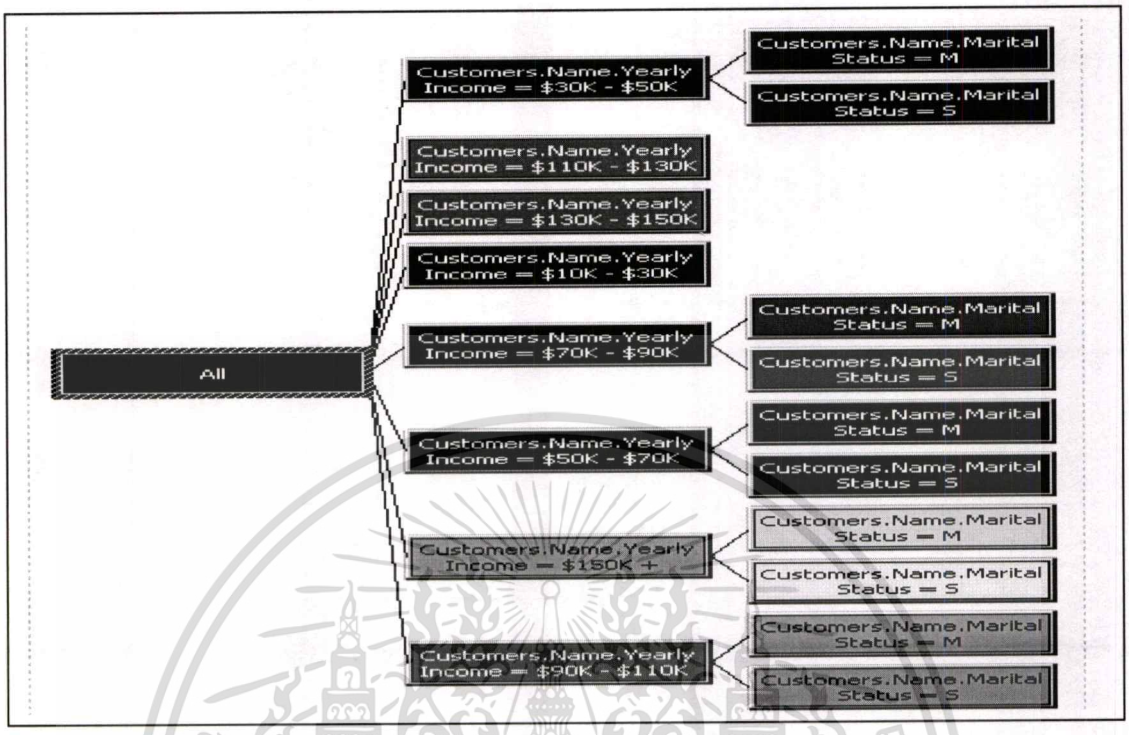
รูปที่5.5 เครื่องมือในการพัฒนาระบบ Microsoft Visual Basic 6 การ Login



รูปที่5.6 เครื่องมือในการพัฒนาระบบ Microsoft Visual Basic 6 การสร้างโมเดล



รูปที่5.7 เครื่องมือในการพัฒนาระบบ Microsoft Visual Basic 6 การประมวลผล



รูปที่ 5.8 แสดงลักษณะโมเดลที่ถูกสร้างใน Analysis Manager

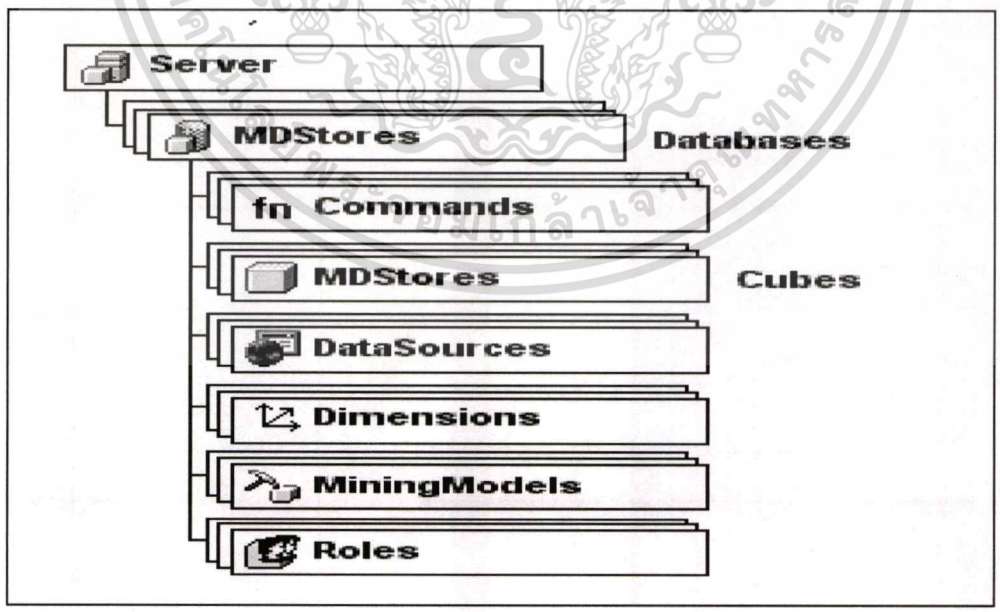
และในการสร้างโปรแกรมประยุกต์นั้น ได้อาศัยโปรแกรม MDX Sample Application ในการทำการทดสอบคำสั่งในการเรียกใช้ข้อมูลจากมุมมองลูกบาศก์ ซึ่งต้องใช้ MDX (Multidimensional Expressions Language)ที่เป็นเครื่องมือ ส่วนหนึ่งในMicrosoft SQL Server 2000 Analysis Services ที่มีโครงสร้างคล้ายภาษา SQL ในการดึงข้อมูลออกมาจาก Cubes

บทที่ 6
การวิเคราะห์และออกแบบโปรแกรม

6.1 การวิเคราะห์และออกแบบโปรแกรม

ในการวิเคราะห์การทำงานโครงงานในครั้งนี้ได้มองถึงกระบวนการทำงานอยู่ 3 กระบวนการคือ

- กระบวนการศึกษาการทำงานของอัลกอริธึมส์และการเชื่อมต่อกับ Server Analysis ในการศึกษาขั้นตอนนี้ได้พบกับปัญหาคือ อัลกอริธึมส์ C4.5 ไม่สมารถทำงานในการสร้างโมเดลบน Analysis Server ที่ทำงานอยู่ SQL Server ไม่ได้ เนื่องจาก Server Analysis นั้นได้มีอัลกอริธึมส์ที่ทำงานในการสร้างโมเดลแบบ Decision Trees เอาไว้แล้วคือ Microsoft Decision Trees Algorithms และในการเชื่อมต่อนั้นไม่สามารถติดต่อผ่าน Ado แบบปกติได้ จะต้องทำการติดต่อกับ Server แบบ DSO Object เท่านั้น จึงจะสามารถติดต่อ Server Analysis เข้าไปใช้ข้อมูลใน Cube และ model ได้



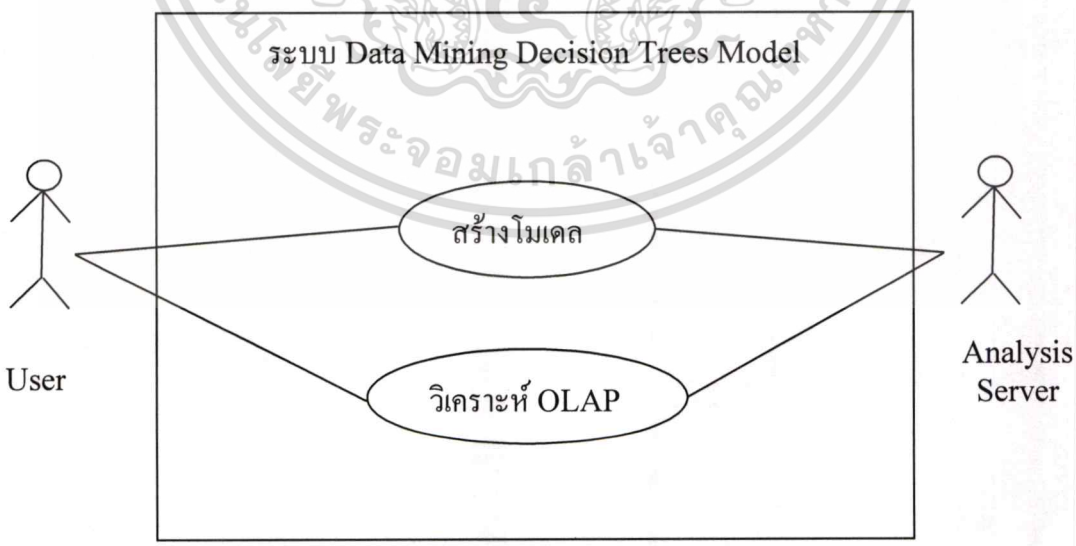
รูปที่ 6.1 แสดง Decision Support Objects Architecture

- กระบวนการสร้างโมเดล ในการศึกษาขั้นตอนนี้ได้พบกับปัญหาคือในการสร้างโมเดลนั้นหากมีการใส่ข้อมูลในส่วนของ Cube , Dimension , Property , Predic ไม่ถูกต้องจะทำให้การสร้างโมเดลนั้นไม่สามารถทำงานได้ หรือแม้แต่กระทั้งหาก Cube ที่ใช้ในการสร้างโมเดลนั้นถูกสร้างมาแบบผิดๆ ก็จะทำให้โมเดลที่ถูกสร้างมานั้นวิเคราะห์ข้อมูลให้เราผิด ดังนั้นในการสร้าง Application ขึ้นมาจะต้องทำการบังคับให้ผู้ใช้ใส่ข้อมูลให้ครบ

- กระบวนการในการวิเคราะห์ข้อมูลที่ถูกเก็บเอาไว้โมเดลที่สร้างเอาไว้ ในการศึกษาขั้นตอนนี้ปัญหาที่พบก็คือ การที่เราจะรวมการทำงานระหว่าง SQL Server และ Analysis Service ให้สามารถทำงานร่วมกันได้บน Application ที่สร้างขึ้นมานั้นประสบปัญหาเป็นอย่างมากเนื่องจากจะต้องทำการสร้าง Virtual Cube ขึ้นมาให้พร้อมกับ Model ที่สร้างขึ้นมา และจะต้องให้สามารถเลือกเงื่อนไขในการแสดงผลได้ทั้ง Cube ที่อยู่ใน Analysis และ Model ให้ได้บน Application เดียวกันได้

6.2 Use Case Diagram

ในการออกแบบในครั้งนี้ได้ใช้ Use Case ในการออกแบบระบบเพื่อให้ User มองเห็นลักษณะการทำงานของระบบ Data Mining Decision Trees ที่ได้ทำการสร้าง Application เอาไว้



รูปที่ 6.2 แสดง Use Case ในการทำงาน

จาก Use Case ในส่วนของ Actor User นั้นจะทำการส่งเงื่อนไขต่าง ๆ ในการสร้างโมเดลเข้าไปใน Use Case สร้างโมเดล เพื่อทำการตรวจสอบเงื่อนไขต่าง ๆ ที่เลือกเข้ามาว่าสามารถทำการสร้างโมเดลได้หรือไม่ เมื่อทำการตรวจสอบเป็นที่เรียบร้อยแล้วก็จะไปเรียกใช้ Actor Analysis Server ให้ทำการสร้างโมเดลให้ตามที่ร้องขอ

งานอีกส่วนหนึ่งก็คือ Actor User สามารถที่จะทำการส่งเงื่อนไข หรือ คำถามต่าง ๆ เพื่อส่งให้โมเดลทำการวิเคราะห์หาคำตอบออกมาให้โดยเลือกที่จะติดต่อกับโมเดลตัวใดและทำการส่งเงื่อนไขหรือคำถามต่าง ๆ เข้าไปที่ Use Case วิเคราะห์ OLAP และตัว Use Case ก็จะทำการเรียกใช้ Actor Analysis Server ให้ทำการประมวลผลตามที่ร้องขอและแสดงข้อมูลที่ได้ทำการวิเคราะห์ออกมาแสดง

6.2.1 Use Case Description

ตารางที่ 6.1 REQ-1-1 สร้างโมเดล

Actors	User
Purpose	เมื่อ User ส่งเงื่อนไขความต้องการในการสร้างโมเดล ระบบช่วยทำการตรวจสอบความถูกต้องของข้อมูลที่จะนำมาสร้างโมเดล

Typical Course of Events

ACTOR	SYSTEM	DESCRIPTION
1		Actor ส่งเงื่อนไขในการสร้างโมเดลเข้าสู่ระบบ
	2	ระบบจะทำการตรวจสอบข้อมูลที่เลือกในการสร้างโมเดลและระบบจะทำการส่งคำสั่งเพื่อไปสั่งให้ Analysis Server ทำการสร้างโมเดลให้ตามที่ร้องขอ
	3	ระบบจะทำการแจ้ง ข้อความในการสร้างโมเดลให้ทราบหากทำงานเสร็จเรียบร้อย

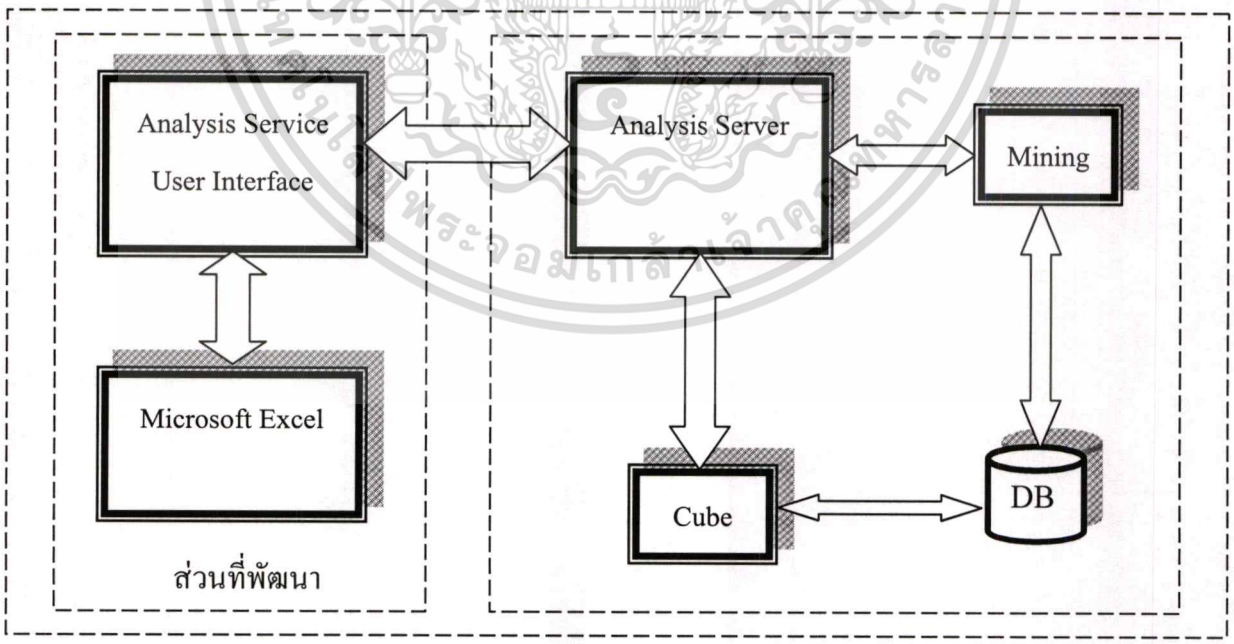
ตารางที่ 6.2 REQ-1-2 การวิเคราะห์ข้อมูล

Actors	User
Purpose	เมื่อ User ส่งเงื่อนไขความต้องการในการสอบถามหรือวิเคราะห์ข้อมูล ระบบช่วยทำการตรวจสอบความถูกต้องของข้อมูลที่จะนำมาวิเคราะห์

Typical Course of Events

ACTOR	SYSTEM	DESCRIPTION
1		Actor ส่งเงื่อนไขในการวิเคราะห์ข้อมูลเข้าสู่ระบบ
	2	ระบบจะทำการตรวจสอบข้อมูลที่เลือกหากถูกต้องก็จะทำการส่งคำสั่งเพื่อไปสั่งให้ Analysis Server ทำการวิเคราะห์ข้อมูลจากโมเดลให้ตามที่ร้องขอ
	3	ระบบจะทำการแจ้ง ข้อมูลในการวิเคราะห์ข้อมูลให้ทราบหากทำงานเสร็จเรียบร้อย

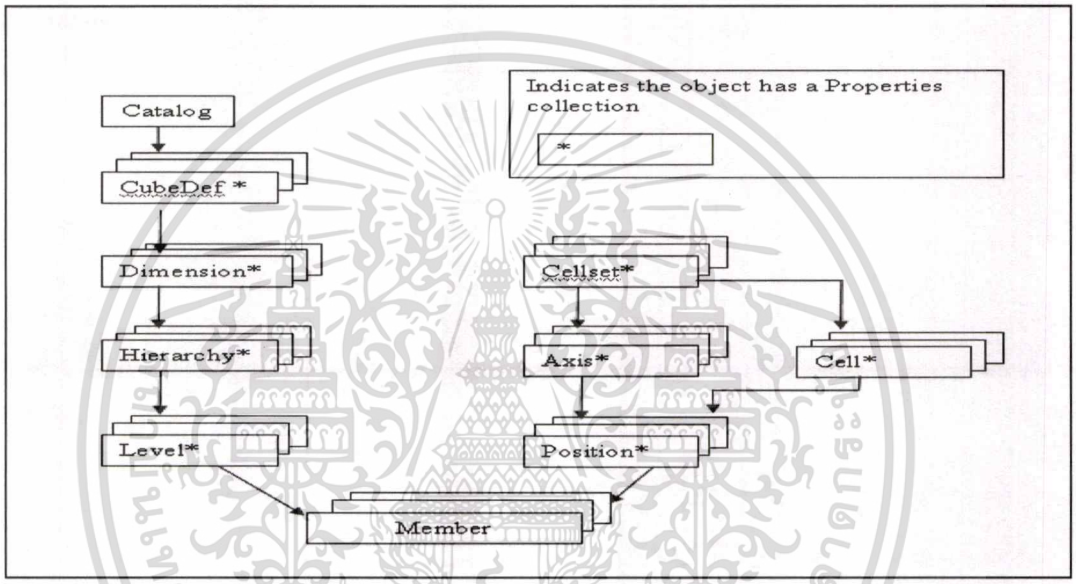
จากระบบการทำงานนั้นสามารถนำมาเขียนเป็นสถาปัตยกรรมเพื่อสรุปขอบเขตการทำงานของ Application ที่สร้างขึ้นมาได้ดังภาพ



รูปที่ 6.3 แสดงสถาปัตยกรรมของระบบทำงาน

6.2.2 โครงสร้างการติดต่อสื่อสารในโปรแกรมตาม ADO/MD Object hierarchy

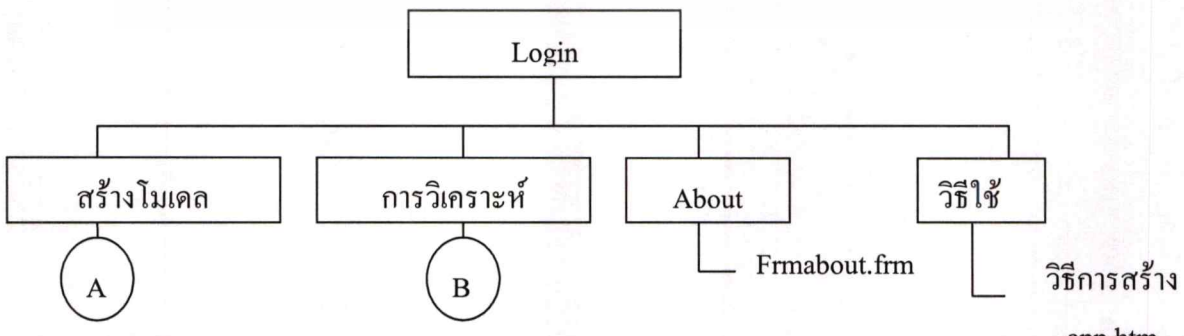
ในการทำการติดต่อกับแหล่งข้อมูลต่าง ๆ ใน Analysis Server นั้นจำเป็นจะต้องทำการติดต่อเข้าไปเป็นลำดับชั้นตามผังของ ADO/MD Object hierarchy เช่นต้องการข้อมูลจาก Cube ที่อยู่ใน Analysis Server นั้นจะต้องเข้าไปใน Catalog หรือติดต่อให้ได้เสียก่อน โดยผังภาพการติดต่อข้อมูลใน Analysis Server นั้นได้แสดงไว้ดังภาพด้านล่าง



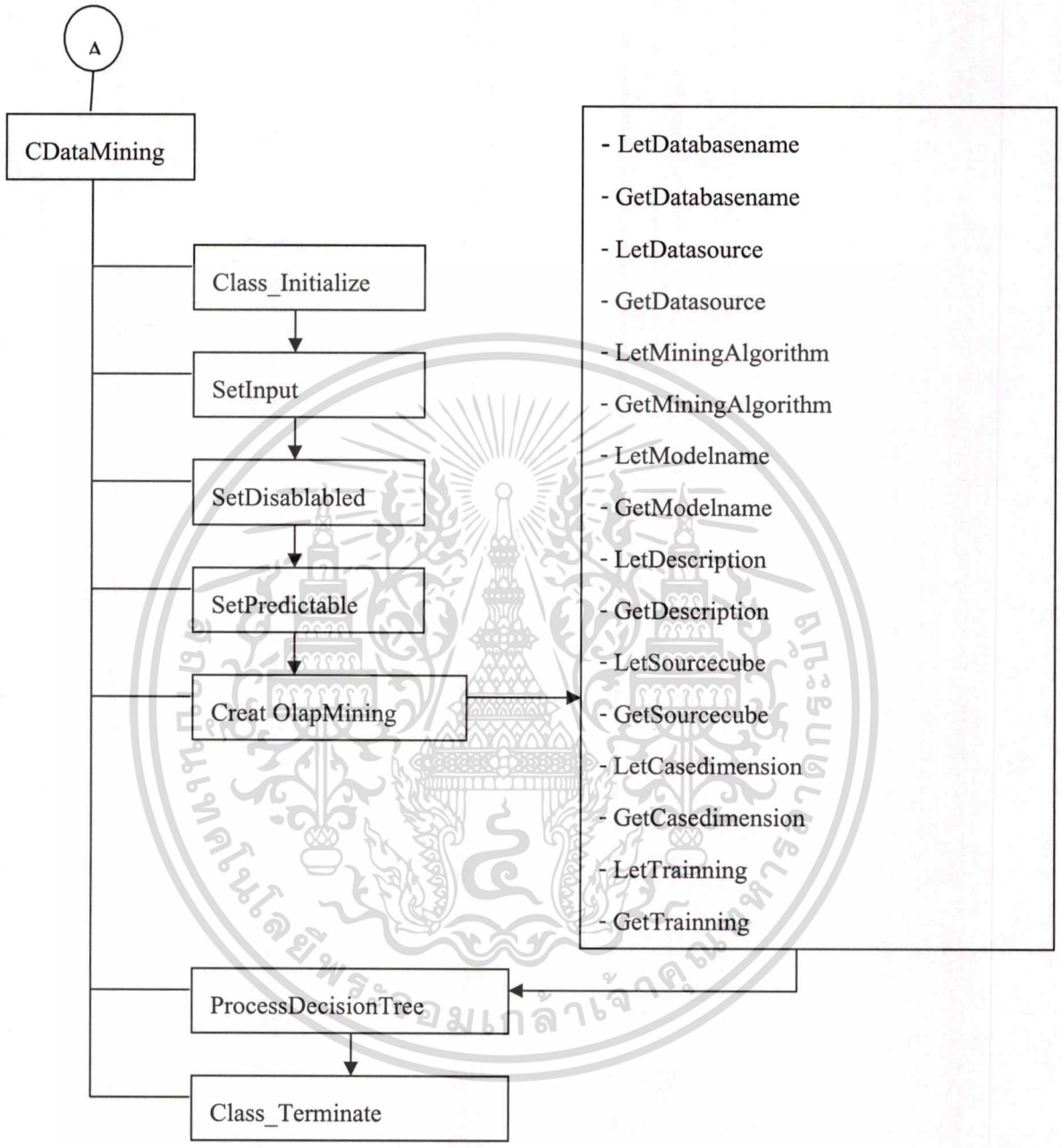
รูปที่ 6.4 แสดง ADO/MD Object hierarchy

6.2.3 โครงสร้างโปรแกรม และ ส่วนประกอบต่าง ๆ ของโปรแกรม

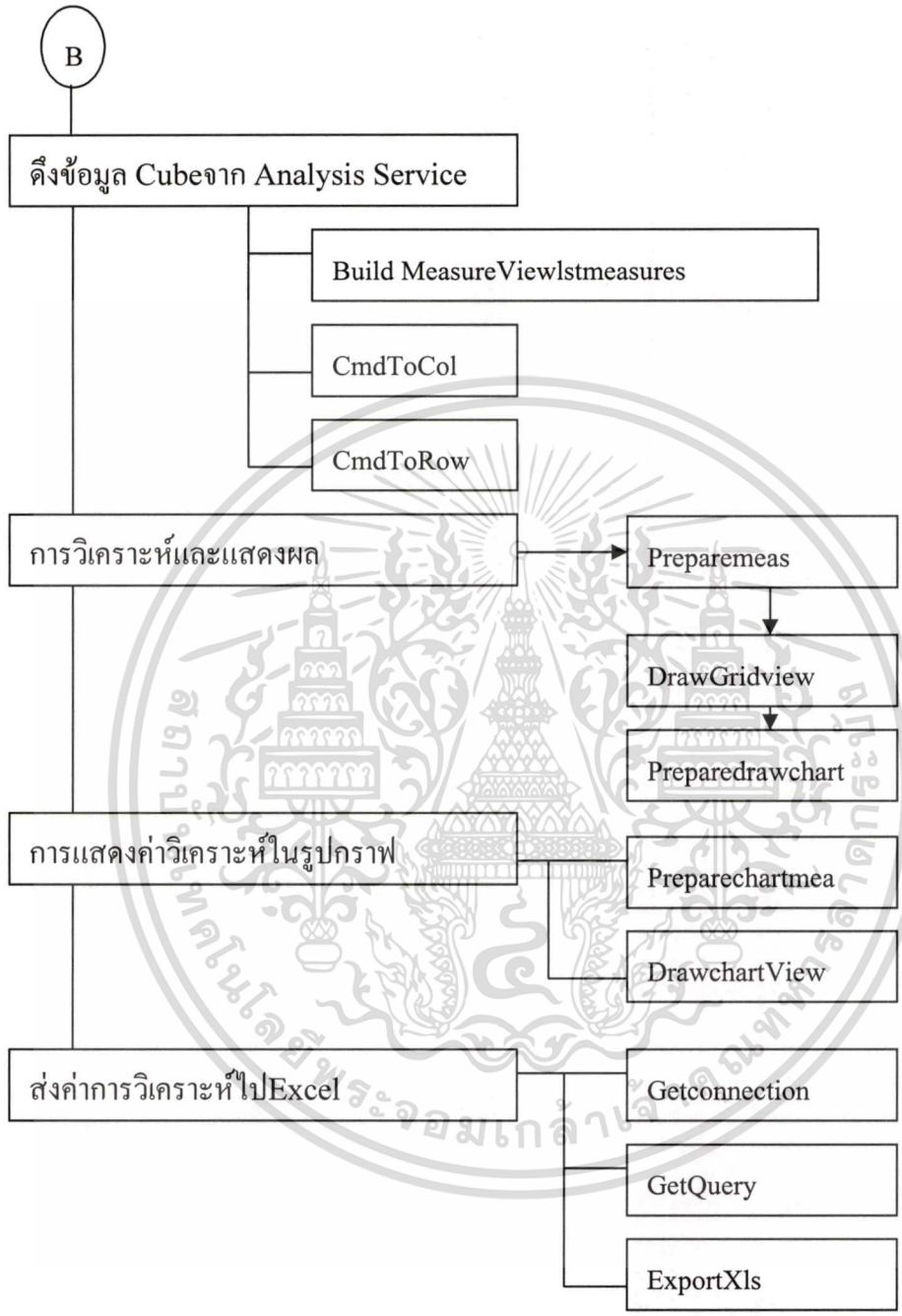
เป็นโครงสร้างของการเรียกใช้โปรแกรมที่สร้างขึ้นมาเพื่อทำการสร้างโมเดลและทำการวิเคราะห์ข้อมูลต่าง ๆ ตามที่ User ต้องการดังนี้



รูปที่ 6.5 แสดง โครงสร้างโปรแกรมส่วน1



รูปที่ 6.6 แสดง โครงสร้างโปรแกรมส่วน2



รูปที่ 6.7 แสดง โครงสร้างโปรแกรมส่วน3

บทที่ 7
การประยุกต์ใช้โปรแกรมกับกรณีศึกษา

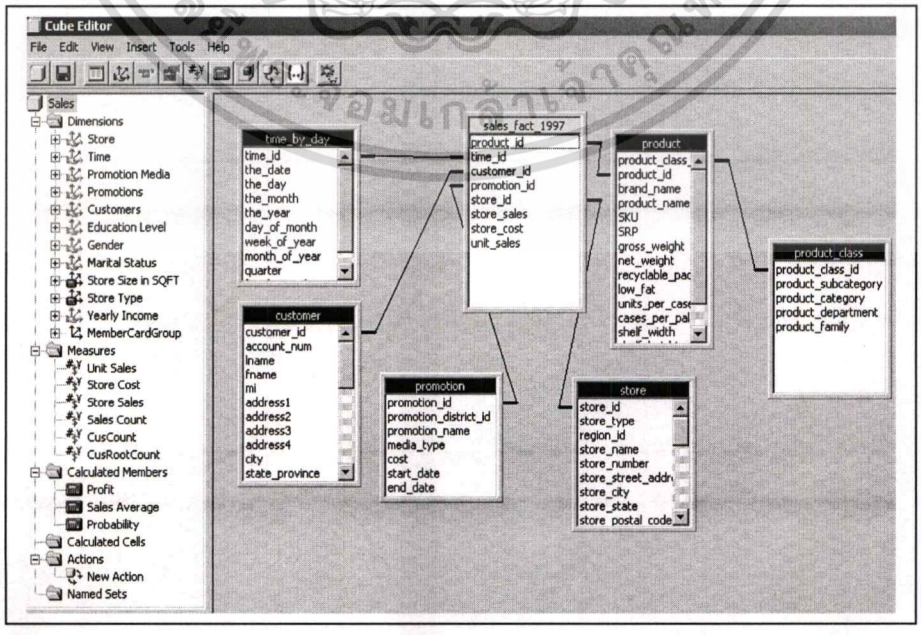
7.1 การระบุโอกาสทางธุรกิจหรือปัญหาที่เกิดขึ้น

บริษัทแห่งหนึ่งมีการเก็บข้อมูลของการใช้บัตรเครดิตประเภทต่าง ๆของลูกค้าในการซื้อสินค้าและบริการ มีความต้องการที่จะทำการจัดตั้งบริษัทลูกที่รับเป็นตัวแทนในการทำบัตรเครดิตให้กับธนาคารต่าง ๆ ดังนั้นจึงต้องการทราบถึงปัจจัยที่ส่งผลต่อการเลือกทำบัตรเครดิตประเภทต่าง ๆของลูกค้าเพื่อที่จะได้ทำการวางแผนทางการตลาดได้ถูกต้องต้องกับกลุ่มเป้าหมาย

7.2 เทคนิคของดาต้าไมน์นิ่ง

จากปัญหาที่เกิดขึ้นและทรัพยากรที่ทางบริษัทมี สามารถที่จะทำการวิเคราะห์ข้อมูลที่ผู้บริหารต้องการได้โดยใช้เทคนิคของดาต้าไมน์นิ่ง แบบ Decision Trees Model มาช่วยในการวิเคราะห์ข้อมูลที่มีอยู่ โดยเริ่มจากกระบวนการคัดเลือกข้อมูลจากฐานข้อมูลมาทำการสร้าง Cube และ Dimension เพื่อให้พร้อมสำหรับการสร้างโมเดล

7.2.1 รายละเอียดตารางระบบงานที่นำมาสร้างเป็น Cube



รูปที่ 7.1 แสดงตารางที่ใช้ในการสร้างCube

จากแผนภาพตารางส่วนประกอบของ Cube ที่นำมาทำ Data Mining นั้นสามารถแสดงรายละเอียดของตารางต่าง ๆ ได้ ดังนี้

ตารางที่ 7.1 อธิบายรายละเอียดตาราง ทั้งหมด ได้ว่า

ลำดับที่	ชื่อตาราง	รายละเอียด
1	Sales_fact_1997	ตารางข้อมูลการซื้อของลูกค้า
2	Customer	ตารางข้อมูลของลูกค้า
3	Product	ตารางข้อมูลสินค้า
4	Product_class	ตารางข้อมูลประเภทสินค้า
5	Promotion	ตารางข้อมูลโปรโมชั่น
6	Store	ตารางที่เก็บสินค้า
7	Time_by_day	ตารางเก็บวันเดือนปี

รายละเอียดตารางทั้ง 7 ตารางแสดงเอาไว้ในตารางที่ 16.2 ถึง ตารางที่ 16.8 โดยข้อความในคอลัมน์ Key มีความหมายดังนี้

PK หมายถึง คีย์หลักของตาราง (Primary Key)

FK หมายถึง คีย์นอกของตาราง (Foreign Key)

ส่วนข้อความในคอลัมน์หมายเหตุ หมายถึง ชื่อตารางที่มีความสัมพันธ์กับค่าคีย์หลักหรือคีย์นอกของตารางที่อ้างถึง (Referenced Table)

ตารางที่ 7.2 อธิบายรายละเอียดตาราง Product_class

ลำดับที่	Attribute	Data Type	Key	หมายเหตุ
1	product_class_id	Number	PK	
2	product_subcategory	Text		
3	product_category	Text		
4	product_department	Text		
5	product_family	Text		

ตารางที่ 7.3 อธิบายรายละเอียดตาราง Promotion

ลำดับที่	Attribute	Data Type	Key	หมายเหตุ
1	promotion_id	Number	PK	
2	promotion_district_id	Number		
3	promotion_name	Text		
4	media_type	Text		
5	cost	Number		
6	start_date	Date/Time		
7	end_date	Date/Time		

ตารางที่ 7.4 อธิบายรายละเอียดตาราง Store

ลำดับที่	Attribute	Data Type	Key	หมายเหตุ
1	store_id	Number	PK	
2	store_type	Text		
3	region_id	Number		
4	store_name	Text		
5	store_number	Number		
6	store_street_address	Text		
7	store_city	Text		
8	store_state	Text		
9	store_postal_code	Text		
10	store_country	Text		
11	store_manager	Text		
12	store_phone	Text		
13	store_fax	Text		

ตารางที่ 7.5 อธิบายรายละเอียดตาราง Customer

ลำดับที่	Attribute	Data Type	Key	หมายเหตุ
1	customer_id	Number	PK	
2	lname	Text		
3	fname	Text		
4	address1	Text		
5	city	Text		
6	state_province	Text		
7	postal_code	Text		
8	country	Text		
9	phone1	Text		
10	birthdate	Date/Time		
11	marital_status	Text		
12	yearly_income	Text		
13	gender	Text		
14	education	Text		
15	member_card	Text		

ตารางที่ 7.6 อธิบายรายละเอียดตาราง Time_by_day

ลำดับที่	Attribute	Data Type	Key	หมายเหตุ
1	time_id	Number	PK	
2	the_date	Date/Time		
3	the_day	Text		
4	the_month	Text		
5	the_year	Number		
6	day_of_month	Number		
7	week_of_year	Number		
8	month_of_year	Number		

ตารางที่ 7.7 อธิบายรายละเอียดตาราง Product

ลำดับที่	Attribute	Data Type	Key	หมายเหตุ
1	product_id	Number	PK	
2	product_class_id	Number	FK	Product_class
3	product_name	Text		
4	SKU	Number		
5	SRP	Currency		
6	gross_weight	Number		

ตารางที่ 7.8 อธิบายรายละเอียดตาราง Sales_fact_1997

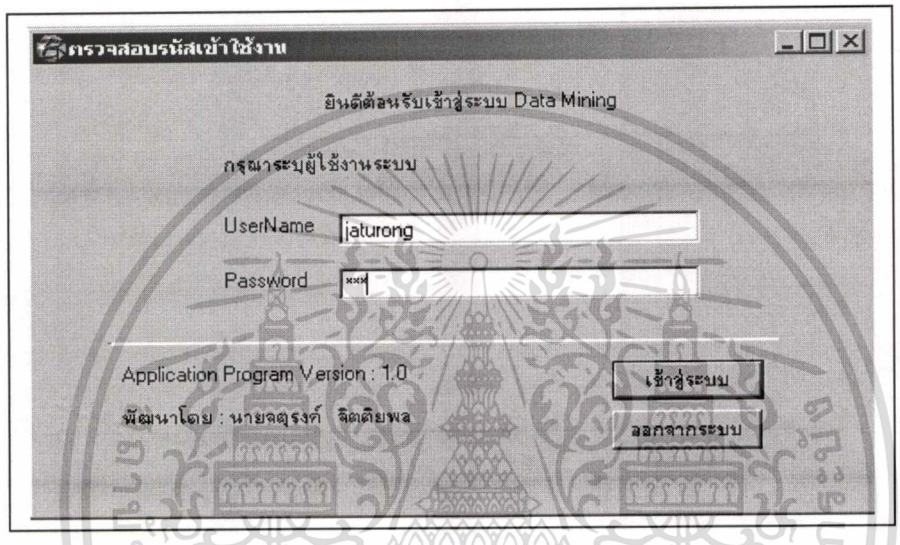
ลำดับที่	Attribute	Data Type	Key	หมายเหตุ
1	product_id	Number	PK	Product
2	time_id	Number	PK	Time_by_day
3	customer_id	Number	PK	Customer
4	promotion_id	Number	PK	Promotion
5	store_id	Number	PK	Store
6	store_sales	Currency		
7	store_cost	Currency		
8	unit_sales	Number		

7.3 ส่วนต่อประสานผู้ใช้

7.3.1 การทำงานของโปรแกรม

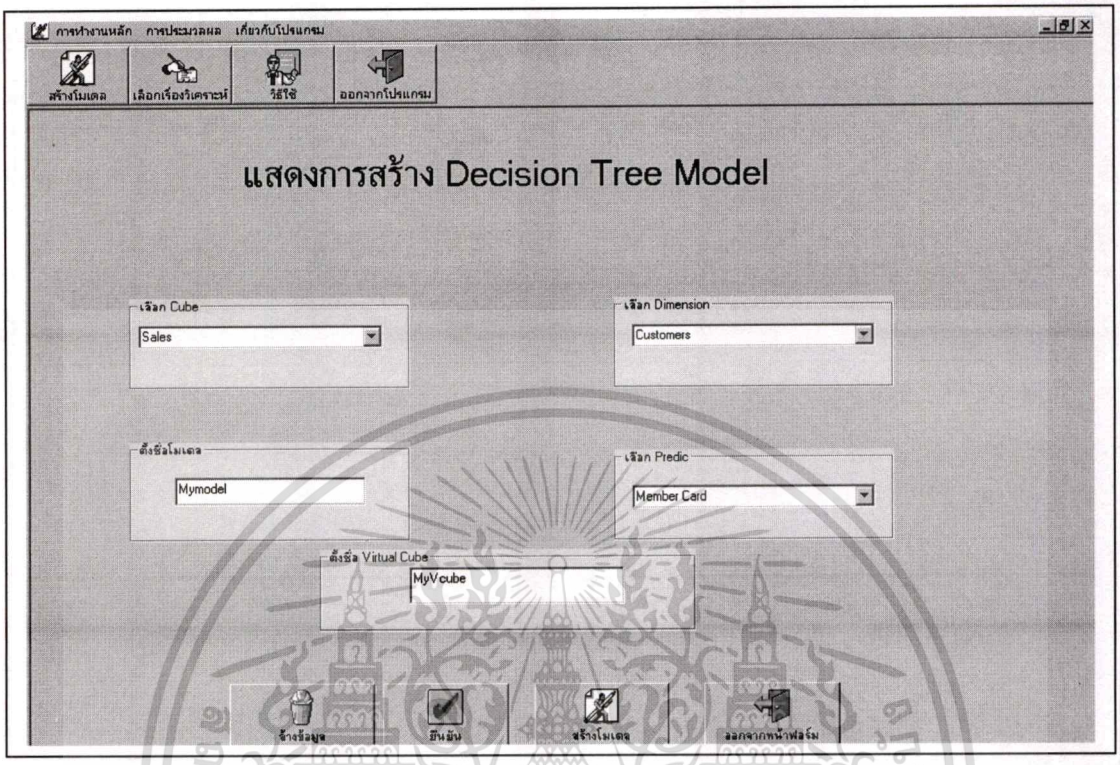
โปรแกรมประยุกต์ที่พัฒนานั้น มีลักษณะการทำงานดังนี้

- ทำการ Login เข้าสู่ระบบเพื่อตรวจสอบสิทธ์ในการเข้าใช้ว่าถูกต้องหรือไม่หากไม่ถูกต้องจะมีข้อความเตือนให้ป้อน user และ password ใหม่

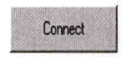


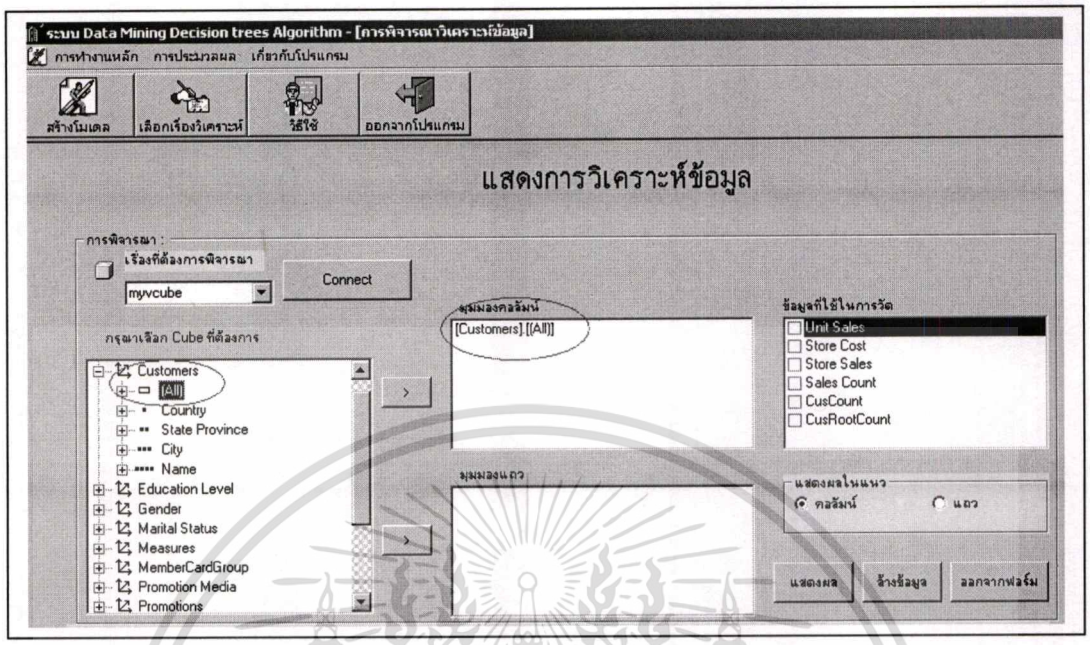
รูปที่ 7.2 แสดงการตรวจสอบชื่อและรหัสเข้าใช้

- คลิกที่ปุ่ม  จากเมนู ในกรณีที่ไม่เคยมีการสร้างโมเดลเอาไว้ก่อนเพื่อทำการสร้างโมเดลในการวิเคราะห์เพื่อหาปัจจัยใดมีผลต่อการเลือกทำบัตรเครดิตประเภทต่าง ๆ เมื่อคลิกแล้วจะได้หน้าจอในการสร้างโมเดล โดย User จะต้องทำการป้อนเงื่อนไขต่าง ๆ ในการที่จะสร้างโมเดลให้ครบ อาทิ เลือก Cube ที่จะนำข้อมูลมาสร้างโมเดล เลือก Dimension , Predict , ตั้งชื่อโมเดล , และตั้งชื่อ Virtual Cube ที่จะถูกสร้างขึ้นมา หากใส่ข้อมูลไม่ครบเมื่อกดปุ่ม  ระบบจะมีข้อความเตือนว่ายังไม่ได้ระบุเงื่อนไขใดและจะไม่สามารถใช้งานปุ่มสร้างโมเดล

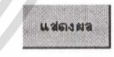


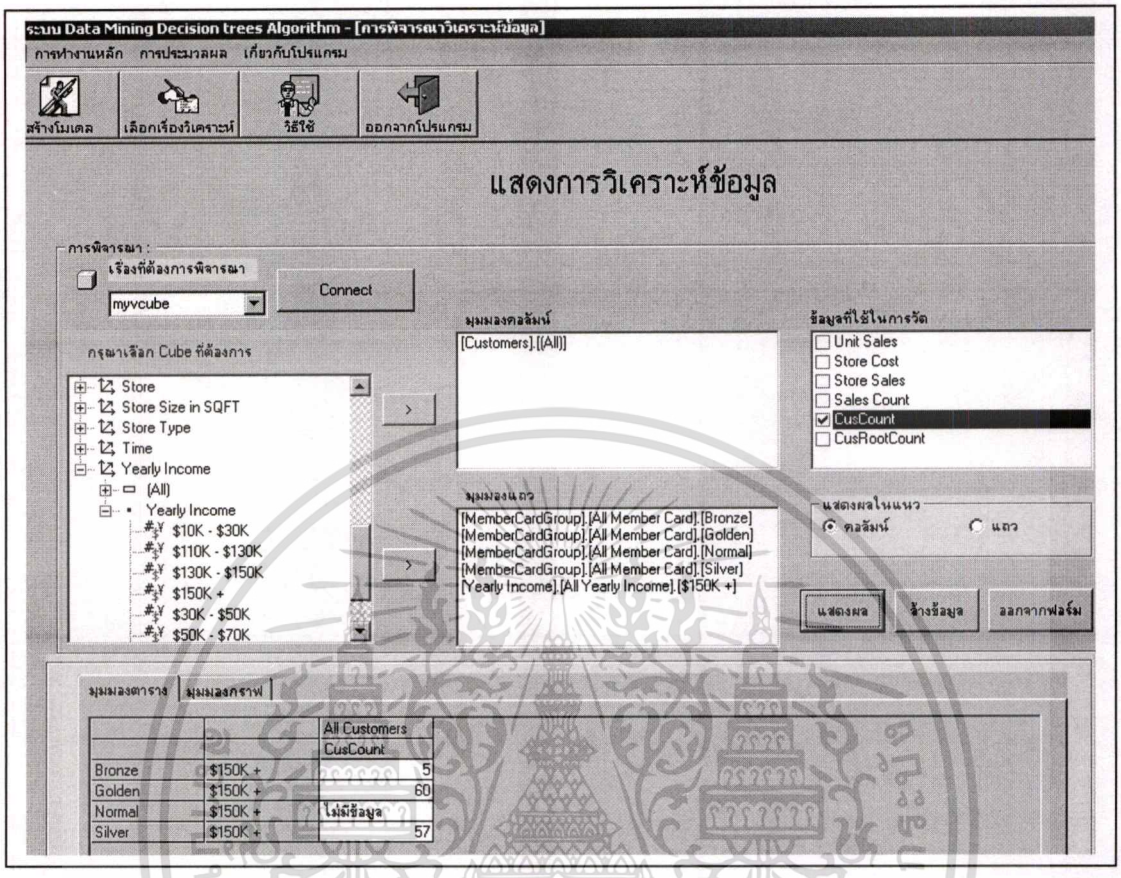
รูปที่ 7.3 แสดงการเลือกเงื่อนไขในการสร้างโมเดล

- เมื่อทำการยืนยันข้อมูลแล้ว ก็คลิกที่ปุ่ม  หลังจากนั้นระบบจะทำการสร้างโมเดลและ virtual cube ให้ เมื่อทำการสร้างเสร็จจะมีข้อความขึ้นมาบอกว่าได้ทำการสร้างโมเดลตามที่ร้องขอเสร็จเป็นที่เรียบร้อยแล้ว
- จากนั้นก็เข้าสู่หน้าจอของการวิเคราะห์ข้อมูลโดยทำการคลิกที่ปุ่ม  แล้วทำการเลือกเงื่อนไขในการวิเคราะห์ข้อมูลตาม Case ที่ต้องการ เช่น ยากทราบปัจจัยที่มีผลต่อการเลือกทำบัตรเครดิตประเภทต่าง ๆ โดยเลือกเงื่อนไขต่าง ๆ ดังต่อไปนี้
- คลิกเลือก Cube ที่เราสร้างขึ้นมาที่ชื่อ myvcube แล้วคลิกที่ปุ่ม  ระบบก็จะทำการดึงข้อมูลออกจากโมเดลที่เราสร้างเอาไว้ออกมาให้
- จากนั้นก็เลือกข้อมูลในส่วนของ Dimension Customer ไว้ในส่วนของคอลัมน์โดยคลิกเลือก Dimension แล้วคลิกที่ปุ่ม 



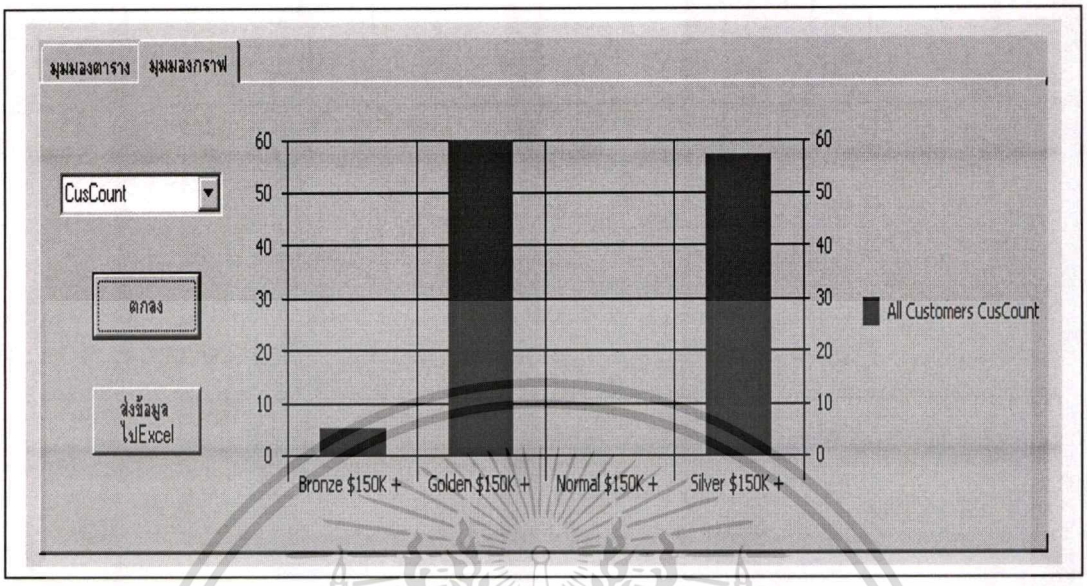
รูปที่7.4 แสดงการเลือกเงื่อนไขมาไว้ในคอลัมน์

- จากนั้นก็ทำการเลือกเงื่อนไขส่วนที่เหลือก็คือ เลือกMemberCardGroup เอาประเภทบัตรทุกประเภทมาไว้ในส่วนของแถว และเลือก Yearly Income ในส่วนที่ต้องการเช่น $150K+
- ขั้นตอนต่อไปคือเลือกวิธีการวัดผลที่ต้องการเช่นวัดที่จำนวนประชากรก็คลิกที่ช่องข้อมูลที่ใช้ในการวัดในส่วน CusCount หลังจากนั้นก็คลิกที่ปุ่ม 

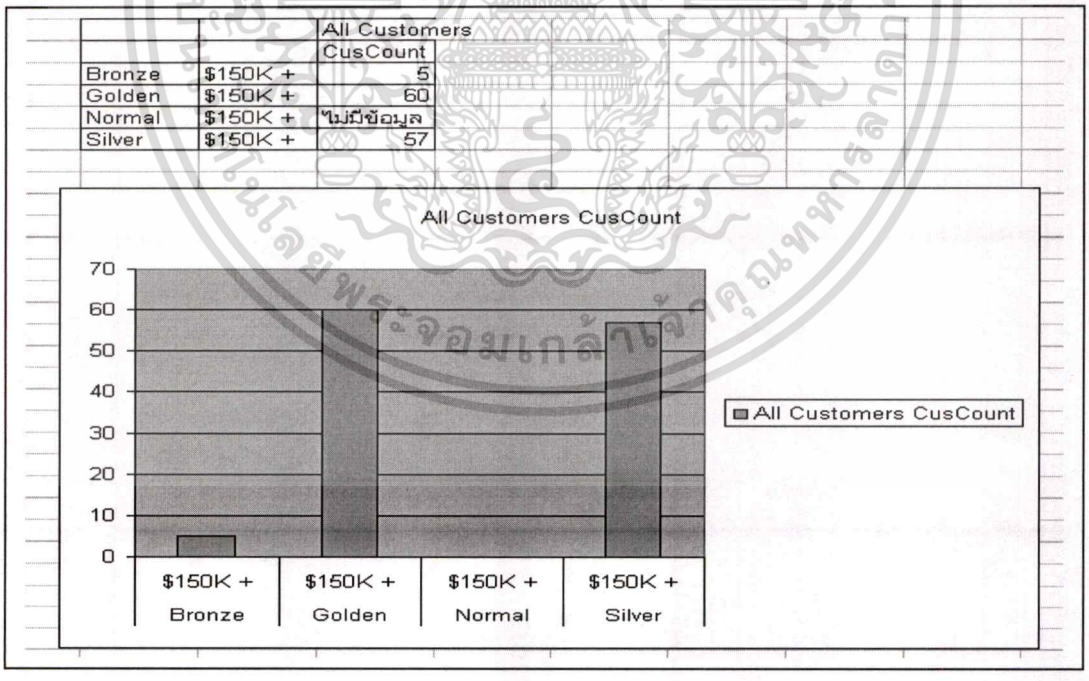


รูปที่ 7.5 แสดงการเลือกเงื่อนไขการวิเคราะห์มาไว้ที่ส่วนแถว

- หลังจากที่คลิกที่ปุ่ม แสดงผล แล้วระบบจะทำการประมวลผลตามที่ร้องขอ และสรุปออกมาเป็นชุดของตัวเลขในลักษณะ PivotTable หากต้องการดูการแสดงผลแบบกราฟก็ให้ทำการคลิกที่แทป มุมมองกราฟ จากนั้นเลือกเงื่อนไขที่ต้องการแสดงผลทางกราฟแล้วคลิกที่ปุ่ม ตกลง
- และถ้าหากต้องการส่งข้อมูลกราฟเอาไปบันทึกเก็บไว้เป็นไฟล์ประเภท Excel ก็ให้คลิกที่ปุ่ม ส่งข้อมูลไปExcel ทางระบบก็จะทำการส่งข้อมูลออกไปอยู่ในรูปเอกสาร Excel ที่เราสามารบันทึกเก็บเอาไว้ได้



รูปที่ 7.6 แสดงผลการวิเคราะห์ในรูปแบบกราฟ



		All Customers CusCount
Bronze	$150K +	5
Golden	$150K +	60
Normal	$150K +	ไม่มีข้อมูล
Silver	$150K +	57

รูปที่ 7.7 แสดงผลการวิเคราะห์ในรูปแบบเอกสาร Excel

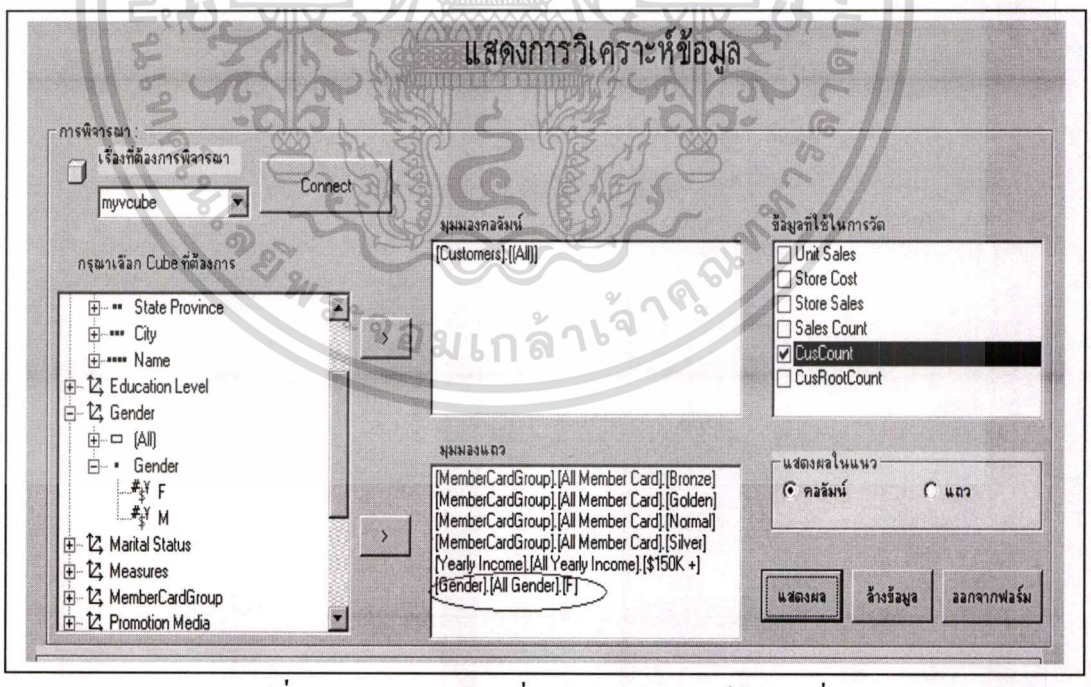
7.3.2 การวิเคราะห์ผลที่ได้จากผลลัพธ์

จากผลการวิเคราะห์สามารถสรุปได้ว่า ปัจจัยที่มีผลต่อการเลือกทำบัตรเครดิตประเภทต่าง ๆ ภายใต้เงื่อนไข Year Income = $150K+ จะมีจำนวนประชากรนิยมทำบัตรเครดิตประเภทต่าง ๆ สรุปจากมากไปหาน้อยได้ดังนี้

Golden	60
Silver	57
Bronze	5
Normal	0

ดังนั้น เห็นควรว่าการลงทุนในส่วนของลูกค้าที่มีรายได้ประมาณ $150K+ นั้น ควรเน้นหนักไปที่ประเภทบัตร Golden รองลงมาคือ Silver

- หากผู้ใช้ต้องการวิเคราะห์ข้อมูลจากโมเดลหรือจาก Cubes ที่มีอยู่แล้วก็ทำการเลือก Cubes ที่ต้องการพิจารณาจากมุมมองลูกบาศก์ที่มีอยู่
- จากนั้นทำการระบุว่าต้องการให้แสดงข้อมูลต่างๆในลักษณะของตาราง ซึ่งจะให้เลือกว่ามุมมองที่ต้องการพิจารณาใดอยู่ในแนวคอลัมน์หรือแนวนอน

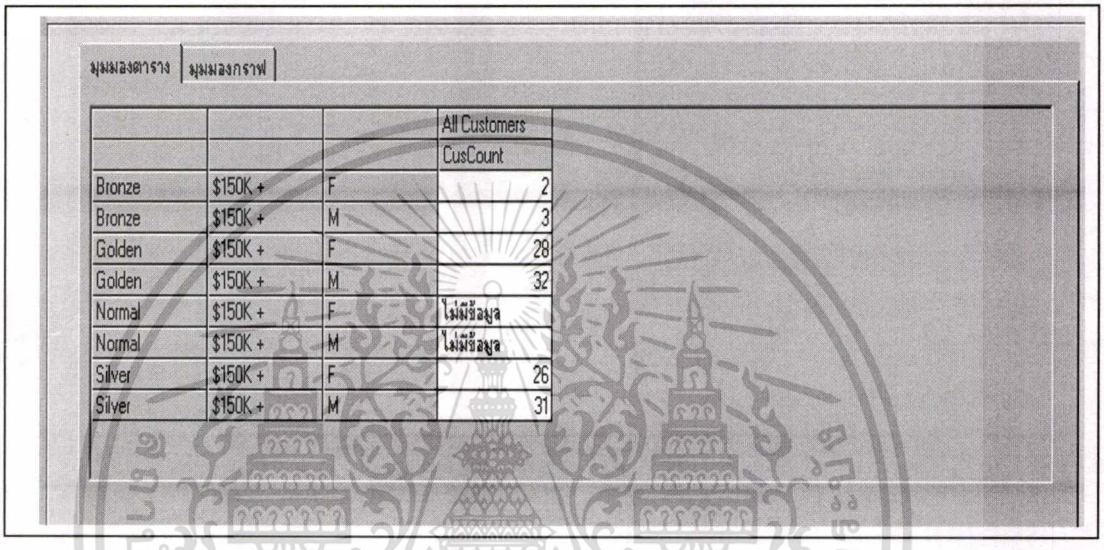


รูปที่ 7.8 แสดงการเลือกเงื่อนไขในวิเคราะห์ข้อมูลเพิ่มเติม

ทั้งนี้ในกรณีที่เป็นมุมมองเดียวกันแต่มีการเลือกในระดับที่แตกต่างกัน โปรแกรมจะเลือกข้อมูลในระดับที่ลึกที่สุดในการนำเสนอ เพื่อให้ตรงกับความต้องการในการพิจารณามากที่สุด

- ภายใต้เงื่อนไขเดิมแต่ต้องการทราบว่า ในส่วนของบัตรประเภท Golden นั้น เพศมีความสนใจมากที่สุด
- เมื่อกดปุ่มแสดงผล โปรแกรมจะแสดงผลในรูปแบบของตารางตามที่ได้เลือกไว้

นอกจากนั้น ยังสามารถระบุให้แสดงผลของข้อมูลที่ใช้ในการวัดได้ทั้งแนวคอลัมน์และแนวนอน



			All Customers
			CusCount
Bronze	$150K +	F	2
Bronze	$150K +	M	3
Golden	$150K +	F	28
Golden	$150K +	M	32
Normal	$150K +	F	ไม่มีข้อมูล
Normal	$150K +	M	ไม่มีข้อมูล
Silver	$150K +	F	26
Silver	$150K +	M	31

รูปที่ 7.9 แสดงข้อมูลจากการวิเคราะห์ในรูปแบบตาราง

- เมื่อเลือกแถบมุมมองกราฟจะสามารถพิจารณาในรูปแบบของแผนภูมิแท่ง ซึ่งอาจจะต้องระบุเลือกข้อมูลที่ใช้ในการวัดแล้วแต่กรณีหากมีค่าที่ใช้วัดมากกว่า 1 อย่าง

จากข้อมูลที่ปรากฏก็สามารถสรุปได้ว่าในกรณีเดียวกันคือปัจจัยที่มีผลต่อการเลือกทำบัตรเครดิตประเภทต่าง ๆ ภายใต้เงื่อนไข Year Income = $150K+ โดยเพิ่มเงื่อนไขว่าเพศใดให้ความนิยมมากที่สุด จากผลการวิเคราะห์สรุปได้ว่า เพศชายนั้นให้ความสนใจในบัตรประเภทนี้มากกว่า

บทที่ 8
สรุปผลการศึกษา และ ข้อเสนอแนะ

โครงการพัฒนาระบบดาต้าไมน์นิ่งแบบ Decision Trees โดยใช้ SQL Server นี้จัดทำขึ้น เพื่อให้สามารถนำข้อมูลที่เป็นข้อมูลเชิงสัมพันธ์มาสร้างเป็นข้อมูลหลายมิติ (Cubes) สำหรับการใช้งานด้านการวิเคราะห์ข้อมูล โดยนำข้อมูลหลายมิติที่ได้นำมาจัดสร้างโมเดลในการวิเคราะห์ข้อมูลและวิเคราะห์ข้อมูลได้โดยง่ายผ่านทาง Application

8.1 สรุปผลการดำเนินงาน

การนำระบบสารสนเทศเข้ามาช่วยในการทำงานและช่วยในการวิเคราะห์ข้อมูล เพื่อให้ได้ข้อมูลที่สามารถสนับสนุนการตัดสินใจได้ ก็จะทำให้เกิดประสิทธิภาพมากยิ่งขึ้น อีกทั้งในระดับผู้บริหารหน่วยงานย่อมต้องการข้อมูลข่าวสารเฉพาะด้านเพื่อการรองรับการตัดสินใจ ซึ่งนอกเหนือจากระบบคอมพิวเตอร์โดยทั่วไปที่ใช้ในการดำเนินงานแล้ว ยังจะต้องมีระบบที่รองรับการตัดสินใจเพิ่มขึ้น ซึ่งในที่นี้ได้แก่ระบบดาต้าไมน์นิ่งที่ได้จัดทำขึ้น จะทำให้ผู้บริหารได้รับข้อมูลที่ตรงความต้องการด้วยความรวดเร็ว

สำหรับในการพัฒนาระบบดาต้าไมน์นิ่งแบบ Decision Trees นี้พบว่าปัจจัยหนึ่งที่ส่งผลต่อการพัฒนาระบบนี้คือ ความถูกต้องครบถ้วนของข้อมูลนำเข้าสู่ฐานข้อมูลเชิงสัมพันธ์หลายมิติ (Cubes) เนื่องจากหากมีข้อมูลที่ไม่ถูกต้องตามหลักการของระบบฐานข้อมูล จะต้องทำการตรวจสอบก่อนที่จะสร้างระบบระบบดาต้าไมน์นิ่งแบบ Decision Trees อีกทั้งสำหรับโปรแกรมประยุกต์ที่ได้พัฒนา หากผู้ใช้งานต้องการพิจารณาข้อมูลในระดับที่ต่างกัน ก็จำเป็นที่จะต้องเลือกมุมมองใหม่อีกครั้งหนึ่ง เพื่อแสดงผล ซึ่งอาจส่งผลให้เกิดความไม่สะดวกในการใช้งานในบางครั้ง อย่างไรก็ตามระบบดาต้าไมน์นิ่งแบบ Decision Trees ก็ยังเป็นประโยชน์ต่อผู้บริหาร เพื่อที่จะได้ใช้ข้อมูลที่มีอยู่ได้อย่างเต็มประสิทธิภาพ สามารถวางแผนจัดการทรัพยากรต่างๆขององค์กร เพื่อตอบสนองต่อความต้องการต่อไป

8.2 ข้อเสนอแนะ

โปรแกรมที่ได้ทำการพัฒนาขึ้นนั้น สามารถนำไปใช้กับ Cubes อื่นๆ ที่อยู่ใน SQL Server ได้ด้วย เนื่องจากในการพัฒนาได้ใช้ MDX เข้ามาเป็นตัวดึงข้อมูลจาก Cubes จากฐานข้อมูลที่ได้เลือกเอาไว้ อีกประการหนึ่งการพัฒนา Application ในครั้งนี้เป็นการพัฒนา Application ที่อยู่ในรูปแบบ Windows Form กล่าวคือสามารถทำงานบน OS Windows เท่านั้น ซึ่งอาจจะทำให้ไม่เกิดความสะดวกหากต้องการนำไปใช้บน OS อื่น

นอกนากนี้ Application ที่ได้พัฒนาขึ้นยังสามารถที่จะพัฒนาเพิ่มเติมให้ทำงานบน Web Application ได้อีกด้วย โดยอาจจะใช้ภาษา ASP ในการพัฒนาเพื่อทำให้การใช้งานเพิ่มประสิทธิภาพมากขึ้นต่อไปได้

บรรณานุกรม

พรพิมล อนันควานิช. 2545. **คัมภีร์นักวิเคราะห์ Microsoft SQL Server 2000 Analysis Service.** กรุงเทพ: สามย่าน.com.

สมพร จิวรสกุล. 2545. **คู่มือการติดตั้งและใช้งาน Microsoft SQL Server 2000.** นนทบุรี: อินโฟเพรส.

Cabena,Peter and Hadjinian,Pablo and Stadler,Rolf and Verhees,Jaap and Zanasi, Alessandro. 1998. **Discovering Data Mining: From Concept to Implementation.** New Jersey. Prentice Hall PTR.

Gunderloy,Mike and Sneath,Tim. 2001. **SQL Server Developer's Guide To OLAP With Analysis Services.** San Francisco. Paris.

Joshi,Karuna Pande. 1997 . **Analysis of Data Mining Algorithms.**[Online]. Available: http://userpages.umbc.edu/~kjoshi1/data-mine/proj_rpt.htm

Quinlan,J.Ross. 1993. **C4.5:PROGRAMS FOR MACHINE LEARNING.** Morgan Kaufmann. San Mate.

Quinlan,Ross. 2003. **See5/C5.0 1.18.**[Online]. Available: http://www.rulequest.com/see5-info.html

Uregina. 1999. **Decision Tree Rules & Pruning.**[Online]. Available:http://www2.cs.uregina.ca/~hamilton/courses/831/notes/ml/dtrees/4_dtrees3.html

ประวัติผู้เขียน

ชื่อ : นายจตุรงค์ จิตติยพล

วันเดือนปีเกิด : 22 พฤศจิกายน 2518

สถานที่เกิด : ศูนย์อนามัยแม่และเด็ก จังหวัด อุดรธานี

ประวัติการศึกษา :

มัธยมต้น : โรงเรียนอุดรพิทยานุกูล

มัธยมปลาย : โรงเรียนอุดรพิทยานุกูล

ปริญญาตรี : มหาวิทยาลัยภาคตะวันออกเฉียงเหนือ