การพัฒนาระบบดาต้าไมน์นิ่งโดยใช้คาร์ทอัลกอริทึม

DEVELOPMENT OF DATA MINING SYSTEM USING CART ALGORITHM

โดย

ณัฐสุดา สิทธิโชค

NUTSUDA SITHEECHOKE

อาจารย์ที่ปรึกษา

รศ.ดร.วรพจน์ กรีสุระเดช

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ภาคเรียนที่ 2 ปีการศึกษา 2551

DEVELOPMENT OF DATA MINING SYSTEM USING CART ALGORITHM

NUTSUDA SITHEECHOKE

A SYSTEM DEVELOPMENT PROJECT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY

FACULTY OF INFORMATION TECNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2/ 2008

หัวข้อ	การพัฒนาระบบดาต้าไมน์นิ่งโดยใช้คาร์ทอัลกอริทึม
นักศึกษา	นางสาวณัฐสุดา สิทธิโชค
รหัสนักศึกษา	42061197
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2551
อาจารย์ที่ปรึกษา	รศ.ดร.วรพจน์ กรีสุระเดช

บทคัดย่อ

ดาต้าไมน์นิ่งเป็นกระบวนการค้นหาข้อมูลที่มีความหมาย และเป็นการค้นพบข้อมูลใหม่ในฐานข้อมูลขนาดใหญ่ โดยจะนำดาต้าไมน์นิ่งมาใช้ในการวิเคราะห์ข้อมูลในฐานข้อมูลได้อย่างมีประสิทธิภาพ เพราะสามารถวิเคราะห์ข้อมูลและหาข้อสรุปซึ่งนำไปสู่ข้อมูลที่เป็นประโยชน์เพื่อใช้ในการตัดสินใจ โดยเทคนิคที่ใช้ในการจัดกลุ่มข้อมูลและใช้ในการทำนายค่าของข้อมูลเพื่อช่วยในการตัดสินใจ คือดิซิชั่นทรี (Decision Tree) ซึ่งแสดงผลอยู่ในรูปแบบแผนภูมิต้นไม้ โดยใช้คาร์ทอัลกอริทึม (Classification and Regression Tree) ซึ่งเป็นอัลกอริทึมหนึ่งในการสร้างโครงสร้างแผนภูมิต้นไม้แบบไบนารี่ทรี เพื่อประโยชน์ในการได้มาซึ่งข้อมูลที่สามารถใช้ในการสนับสนุนการตัดสินใจต่อไป

Title	Development of Data Mining using CART Algorithm
Student	Miss. Nutsuda Sitheechoke
Student ID.	48066417
Degree	Master of Science
Programme	Information Science
Academic Year	2008
Advisor	Assoc. Prof. Dr. Worapoj Kreesuradej

ABSTRACT

Data Mining Process is the way to discovery into useful information and knowledge which has meaning and to find new pattern or information from a large database. Data Mining is used effective to analysis data because it can analyze data to find and to retrieve appropriate information conclusion for appropriate decision. A decision Tree is a predictive modeling technique used in classification task that present in Decision Tree form. One of the techniques used to build a binary tree model is CART Algorithm (Classification and Regression Tree). The model will be very useful to obtain the information that supports the decision making.

กิตติกรรมประกาศ

ในการศึกษาและพัฒนาโครงการพัฒนาระบบดาต้าไมน์นิ่งโดยใช้คาร์ทอัลกอริทึม ได้รับการสนับสนุนและความช่วยเหลือทั้งทางด้านความรู้ แนวทางปฏิบัติ และกำลังใจจากหลายท่าน เพื่อให้โครงการนี้สำเร็จลุล่วง จึงขอขอบพระคุณบุคคลดังต่อไปนี้

บิดา มารดาที่ให้โอกาสทางการศึกษาและเป็นกำลังใจในการทำงานครั้งนี้

รศ.ดร.วรพจน์ กรีสุระเดช อาจารย์ที่ปรึกษาโครงการ ที่ให้ความกรุณาดูแลการทำงานและให้คำปรึกษา ข้อเสนอแนะ ตลอดทั้งแนะนำหนังสือเทคนิควิธีการแก้ไขปัญหาต่างๆอันเป็นประโยชน์ยิ่งต่อการพัฒนาโครงการจนแล้วเสร็จ

เพื่อนๆ ทุกคนที่คอยให้คำปรึกษาและคอยเป็นกำลังใจ รวมทั้งคณาจารย์คณะเทคโนโลยีสารสนเทศที่ได้ประสิทธิประสาทวิชาความรู้ จนสามารถพัฒนาโครงการนี้จนสำเร็จ

ณัฐสุดา สิทธิโชค

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

สารบัญภาพ

รูปที่ หน้า

สารบัญภาพ (ต่อ)

รูปที่ หน้า

บทที่ 1
บทนำ

1.1 ความเป็นมาของปัญหา

ในปัจจุบันองค์กรต่างๆ ได้มีการนำเอาระบบสารสนเทศเข้ามาใช้งานเพื่อเพิ่มประสิทธิภาพในการดำเนินงานขององค์กรเพิ่มมากขึ้น จึงทำให้มีข้อมูลจำนวนมากเกิดขึ้นภายในระบบ โดยข้อมูลที่เกิดขึ้นนั้นได้ถูกจัดเก็บลงฐานข้อมูลที่มีความแตกต่างกันไป ขึ้นอยู่กับความต้องการของแต่ละองค์กรซึ่ง Microsoft SQL Server ก็เป็น Relational Database Management Systems (RDBMS) ตัวหนึ่งที่ได้รับความนิยมเป็นอย่างมากในการจัดเก็บและดูแลรักษาข้อมูลขององค์กรต่างๆ โดยข้อมูลต่างๆ ที่เกิดขึ้นในระบบนั้นหากแต่ละองค์กรรู้จักวิธีการที่จะจัดการกับข้อมูลที่ถูกต้องก็จะเป็นข้อมูลที่มีประโยชน์อย่างมาก แต่ในความเป็นจริงแล้วในหลายองค์กรยังไม่สามารถที่จะนำข้อมูลที่มีประโยชน์เหล่านี้มาใช้งานได้อย่างคุ้มค่าเท่าที่ควร ดังนั้นจึงได้มีแนวคิดเกี่ยวกับการนำกระบวนการทางด้านดาต้าไมน์นิ่ง (Data Mining) ซึ่งเป็นกระบวนการในการเพิ่มประสิทธิภาพของระบบข้อมูลข่าวสารที่มีอยู่ให้มีประโยชน์สูงสุด มาใช้กับข้อมูลต่างๆ ที่เกิดขึ้นภายในองค์กร โดยโครงการนี้ได้นำเอาเทคนิคการจัดกลุ่มของข้อมูล (Classification) แบบโครงสร้างต้นไม้ (Decision Tree) มาประยุกต์ใช้ในการวิเคราะห์ข้อมูลที่เกิดขึ้นของแต่ละองค์กร โดยจะแสดงผลลัพธ์ที่ได้ออกมาในรูปแบบของโครงสร้างต้นไม้ที่มีความสะดวกและง่ายต่อการเข้าใจมากกว่าการนำเสนอข้อมูลหรือผลลัพธ์ในรูปแบบอื่นที่ออกมาเป็นในรูปแบบของข้อมูลทั่วไป ไม่ได้ผ่านกระบวนการจัดการข้อมูลที่ดี ดังนั้นในแต่ละองค์กรจึงจำเป็นที่จะต้องจัดหาเครื่องมือ (Tools) ที่มีความสามารถเกี่ยวกับการจัดการบริหารข้อมูลภายในองค์กรให้มีประสิทธิภาพและเกิดประโยชน์สูงสุด

1.2 วัตถุประสงค์ของโครงการ

1) เพื่อศึกษากระบวนการทางด้านดาต้าไมน์นิ่ง

2) เพื่อเรียนรู้วิธีการจำแนกข้อมูลด้วยดิซิชั่นทรี (Decision Tree) โดยใช้คาร์ทอัลกอริทึม (CART Algorithm) มาใช้ประโยชน์เพื่อเพิ่มประสิทธิภาพของข้อมูลที่มีอยู่ให้เกิดประโยชน์สูงสุด

3) สร้าง Software Tools ที่สามารถจัดการข้อมูลเพื่อให้ได้ข้อมูลที่สามารถนำมาช่วยในการตัดสินใจได้อย่างรวดเร็ว

1.3 ขอบเขตการศึกษา

โครงการนี้เป็นการศึกษาและพัฒนาระบบงานเกี่ยวกับทางด้านดาต้าไมน์นิ่งโดยจะนำหลักการและเทคนิคในการจัดกลุ่มของข้อมูล โดยใช้รูปแบบของ โครงสร้างต้นไม้ มาใช้ในการบรรยายผลลัพธ์ของข้อมูลที่เกิดขึ้นหลังจากที่ข้อมูลนั้นได้ผ่านกระบวนการทางด้านดาต้าไมน์นิ่งแล้ว โดยข้อมูลจะถูกจัดเก็บอยู่ใน Relational Database Management คือ Microsoft SQL Server ซึ่งจะศึกษาเกี่ยวกับวิธีใช้หลักการทำงานต่างๆ ของ Microsoft SQL Server ตลอดจนภาษาที่ใช้ในการสื่อสารกับฐานข้อมูลซึ่งก็คือ Structure Query Language หรือเรียกง่ายๆว่าภาษา SQL Statement

โดยขอบเขตของระบบจะเป็นการดำเนินการตามกระบวนการทางด้านดาต้าไมน์นิ่ง โดยเริ่มจากเลือกข้อมูลที่ได้ผ่านการจัดเตรียมข้อมูล (Data Preparation) ไว้แล้วให้ตรงกับความต้องการและทำการวิเคราะห์ข้อมูลผ่านกระบวนการทางดาต้าไมน์นิ่งให้ได้รูปแบบของข้อมูลที่เราต้องการเพื่อนำไปใช้ในการทดสอบกับข้อมูลอื่นต่อไป

1.4 ขั้นตอนและวิธีการดำเนินโครงการ

1) ศึกษากระบวนการทางด้านดาต้าไมน์นิ่ง โดยเทคนิคที่เลือกใช้กับโครงการคือ Decision Tree รวมทั้งศึกษา CART Algorithm ซึ่งเป็นอัลกอริทึมที่ใช้ในการสร้าง Decision Tree

2) ศึกษาการทำงานของ Microsoft SQL Server

3) ศึกษา Structure Query Language และการติดต่อฐานข้อมูลของ Microsoft Visual Studio .NET

4) ออกแบบและพัฒนาระบบงานตามหลักการและวิธีที่ได้ศึกษา

5) ทดสอบระบบงานและตรวจสอบข้อผิดพลาดต่างๆ เพื่อทำการปรับปรุงและแก้ไข แล้วพัฒนาให้สมบูรณ์

6) สรุปผลการศึกษาและจัดทำเอกสาร

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1) เพิ่มความรวดเร็วในการทำงานเกี่ยวกับการจัดการกับข้อมูลที่มีอยู่

2) เพิ่มประสิทธิภาพการใช้ข้อมูลให้เกิดประโยชน์สูงสุด

3) นำความรู้และเทคนิคที่ได้ศึกษาเกี่ยวกับกระบวนการต่างๆ ไปประยุกต์ใช้ในการวิเคราะห์ข้อมูลเพื่อเพิ่มประสิทธิภาพในการดำเนินงานขององค์กร

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

ปัจจุบันการดำเนินธุรกิจมีการแข่งขันสูง การรวบรวมข้อมูลข่าวสารจึงมีความจำเป็น และมีบทบาทสำคัญในการนำมาใช้ประโยชน์ประกอบการตัดสินใจขององค์กร เพื่อพัฒนาองค์กรให้ประสบความสำเร็จ และสร้างความได้เปรียบในการดำเนินธุรกิจ การวิเคราะห์ข้อมูลที่มีอยู่เพื่อแปลงให้เป็นข่าวสารเพื่อช่วยในการตัดสินใจ โดยการวิเคราะห์ข้อมูลที่มีขนาดใหญ่โดยไม่มีเครื่องมือใดช่วยสามารถทำได้ยาก จึงได้มีการนำระบบเทคโนโลยีสารสนเทศมาประยุกต์ใช้ เพื่อช่วยในการทำงาน ซึ่งได้แก่ การทำดาต้าไมน์นิ่ง (Data Mining) เพื่อใช้ในการวิเคราะห์ข้อมูลจากข้อมูลจำนวนมากที่ถูกจัดเก็บในฐานข้อมูลอย่างเป็นระเบียบ เพื่อหารูปแบบและความสัมพันธ์ของข้อมูล ซึ่งสามารถนำผลที่ได้มาใช้ประโยชน์ประกอบการตัดสินใจ

2.1 ดาต้าไมน์นิ่ง (Data Mining)

ดาต้าไมน์นิ่ง คือ ขั้นตอนหรือกระบวนการที่ใช้ในการนำข้อมูลที่ไม่ทราบมาก่อนออกจากฐานข้อมูลขนาดใหญ่ ซึ่งข้อมูลนั้นจะมีความถูกต้อง และสามารถนำไปใช้งานเพื่อช่วยในการตัดสินใจขององค์กร ดาต้าไมน์นิ่งเป็นส่วนหนึ่งของกระบวนการค้นพบความรู้ในฐานข้อมูล (Knowledge Discovery in Database: KDD) ซึ่งเป็นการนำแนวโน้มของข้อมูลและสารสนเทศที่ซ่อนอยู่มาใช้ประโยชน์เพื่อเพิ่มคุณค่าให้กับฐานข้อมูลที่มีอยู่ จึงช่วยให้เกิดศักยภาพในการใช้ข้อมูลในฐานข้อมูล

การทำดาต้าไมน์นิ่ง แบ่งเป็น 4 ประเภท ได้แก่

1) Predictive Modeling เป็นการสร้างแบบจำลองเพื่อใช้ในการทำนาย ซึ่งใช้การสังเกตจากรูปแบบข้อมูลที่มีอยู่ โดยแบ่งเป็น 2 ลักษณะ คือ

1.1 Value Predictive หรือ Forecasting เป็นการทำนายออกมาเป็นค่า เช่น การพยากรณ์อากาศ การทำนายหุ้น เป็นต้น

1.2 Classification เป็นการจำแนกประเภทข้อมูล โดยจัดกลุ่มข้อมูลว่าควรอยู่ในหมวดหมู่ใด โดยแบ่งชนิดตามกลุ่มข้อมูลที่ควรเป็น

2) Database Segmentation เป็นการจัดกลุ่มข้อมูลที่มีลักษณะคล้ายกันหรือมีคุณสมบัติใกล้เคียงกันให้เป็นข้อมูลกลุ่มเดียวกัน ซึ่งช่วยในการหากลุ่มเป้าหมาย เพื่อพิจารณาว่าในแต่ละกลุ่มมีพฤติกรรมหรือลักษณะอย่างไร

3) Link Analysis เป็นการวิเคราะห์ความสัมพันธ์ของข้อมูลแต่ละรายการว่ามีความสัมพันธ์กันหรือไม่ อย่างไร โดยใช้เทคนิคต่างๆ ได้แก่

3.1 Association discovery เป็นหลักการค้นหาสิ่งที่มีความสัมพันธ์กัน

3.2 Sequential Pattern Discovery เป็นการศึกษาว่าหากเกิดเหตุการณ์ขึ้นแล้วเหตุการณ์ใดจะเกิดตามมา

3.3 Similar Time Sequence Discovery เป็นการศึกษาพฤติกรรมของข้อมูลที่เกิดขึ้นทั้งหมด ในช่วงเวลาเดียวกัน

4) Deviation Detection การตรวจสอบค่าเบี่ยงเบน เป็นการวิเคราะห์ความแตกต่าง ว่าข้อมูลใดมีความแตกต่างไปจากข้อมูลอื่นหรือไม่ ซึ่งนิยมนำไปใช้ในการตรวจจับความผิดปกติ โดยมักนำเสนอออกมาในรูปแบบกราฟิก เช่น แผนภูมิ เพื่อให้สามารถเข้าใจได้ง่าย

2.2 ความเป็นมาของดาต้าไมน์นิ่ง

ดาต้าไมน์นิ่ง พัฒนามาจากเทคนิคทางสถิติ ฐานข้อมูล และการเรียนรู้ของเครื่องจักรกล (Machine Learning) โดยจะเน้นที่ข้อมูลที่มีขนาดใหญ่ เพื่อสร้างแบบจำลองหรือตัวแบบที่ใช้สำหรับทำนายพฤติกรรมของข้อมูล ซึ่งในปัจจุบันเทคโนโลยีทำให้กระบวนการดาต้าไมน์นิ่งเป็นไปอย่างอัตโนมัติ มีการรวมเข้ากับคลังข้อมูล (Data Warehouse) ซึ่งเป็นที่รวบรวมข้อมูลทั้งข้อมูลปัจจุบันและข้อมูลเก่าที่รวบรวมมาจากส่วนต่างๆ ขององค์กร และนำเสนอผลลัพธ์ในได้หลากหลายรูปแบบตามที่ผู้ใช้ต้องการได้อย่างสะดวกมากขึ้น จากการพัฒนาดาต้าไมน์นิ่งด้วย 3 เทคนิคข้างต้นนั้นจะมีแนวทางที่ดาต้าไมน์นิ่งนำหลักการมาใช้แตกต่างกันไป คือ

1) ฐานข้อมูล (Database Technology) การทำงานที่ดาต้าไมน์นิ่งนำมาจากฐานข้อมูลคือการเก็บรวบรวบรวมข้อมูล การคำนวณเกี่ยวกับการทำงานที่ทำซ้ำๆ ซึ่งอาจใช้ข้อมูลเฉพาะบางส่วนที่ต้องการในการวิเคราะห์จากฐานข้อมูลที่เก็บไว้เท่านั้น

2) สถิติ (Statistics) คือการรวบรวมข้อมูลที่มีอยู่แล้วโดยนำทฤษฎีทางสถิติมาวิเคราะห์เพื่อบอกถึงค่าความเป็นไปได้ต่างๆ ที่อาจจะเกิดขึ้น

3) การเรียนรู้ของเครื่องจักรกล (Machine Learning) โดยในการทำนายค่าต่างๆ ของดาต้าไมน์นิ่งได้นำความรู้จากวิทยาศาสตร์ทางคอมพิวเตอร์ที่ได้คิดค้นแนวทางที่ทำให้เครื่องจักรเรียนรู้ข้อมูลเพื่อช่วยเพิ่มความสามารถในฐานความรู้ขององค์กรที่มีข้อมูลจำนวนมากหรือมีความซับซ้อนมากเกินไปสำหรับมนุษย์โดยเฉพาะเมื่อต้องวิเคราะห์ข้อมูลนั้นให้เสร็จภายในระยะเวลาอันสั้นซึ่งการนำเสนอผลลัพธ์โดยรูปแบบโครงสร้างหรือแบบจำลองที่สามารถใช้ในการวิเคราะห์ข้อมูลได้โดยรูปแบบขึ้นอยู่เทคนิคที่ใช้

ในปัจจุบันกระบวนการทางดาต้าไมน์นิ่งนั้นจะนำมาใช้สำหรับแก้ปัญหาเกี่ยวการดำเนินธุรกิจที่เกิดขึ้น ซึ่งปัญหาทางธุรกิจที่ได้นำเอาดาต้าไมน์นิ่งมาประยุกต์ใช้ เช่น

1) การเพิ่มขึ้นของคู่แข่งและความเสี่ยงทางธุรกิจ มีความขึ้นโดยเมื่อพิจารณาแล้วพบว่าแนวโน้มของสินค้าจะมีการซื้อขายกันทั่วโลก โดยผ่านทางอินเตอร์เน็ตทำให้ยากที่จะเก็บข้อมูลต่างอีกทั้งแนวโน้มการเปลี่ยนแปลงของลูกค้าเป็นไปอย่างรวดเร็วทำให้การดำเนินธุรกิจเกิดความเสี่ยง

2) การเพิ่มขึ้นของสินค้า โดยสินค้าและบริการหลายชนิดมีการเพิ่มที่แตกต่างจากสายของสินค้าชนิดนั้นๆ เพราะว่ามีตลาดกลุ่มใหม่เกิดขึ้นตลอดเวลา เนื่องจากตลาดของผู้บริโภคมีความชอบที่แตกต่างกันไป

3) เวลา ถูกให้ความสำคัญมากขึ้นเพราะว่าคู่แข่งขันในตลาดมีมากขึ้นอย่างรวดเร็ว

4) วงจรชีวิตของผลิตภัณฑ์สั้นลง เนื่องจากผลิตภัณฑ์ต่างๆ สามารถหาซื้อได้ในตลาดอย่างรวดเร็วและพฤติกรรมของผู้บริโภคมีการเปลี่ยนแปลงอย่างรวดเร็วมาก ทำให้อายุของสินค้ามีช่วงชีวิตที่สั้นลง

5) รูปแบบพฤติกรรมของผู้บริโภค พฤติกรรมของผู้บริโภคมีการเปลี่ยนแปลงไป ซึ่งจะเกิดจากการปรับเปลี่ยนตามสภาพแวดล้อมหรือสภาพของเศรษฐกิจ อีกทั้งลูกค้ายังมีความต้องการและมีการรับรู้ข่าวสารได้รวดเร็วขึ้น

2.3 กระบวนการทำงานของดาต้าไมน์นิ่ง

การทำงานของดาต้าไมน์นิ่งประกอบด้วยหลายขั้นตอนซึ่งมีการทำซ้ำ หรือต้องมีการวนกลับมาทำใหม่อีกครั้ง โดยสามารถแบ่งการทำงานได้เป็น 4 ขั้นตอน ดังนี้

1) การกำหนดวัตถุประสงค์ทางธุรกิจ (Business Objective Determination) มีการศึกษาความต้องการในการวิเคราะห์ โดยกำหนดปัญหาและวัตถุประสงค์ทางธุรกิจหรือองค์กรให้ชัดเจน เพราะปัญหาวัตถุประสงค์ทางธุรกิจนั้นเป็นตัวกำหนดถึงผลลัพธ์หรือเทคนิคที่ต้องนำมาใช้ในการทำดาต้าไมน์นิ่ง จึงทำให้ต้องเข้าใจถึงปัญหาและความต้องการทางธุรกิจ รวมทั้งการวิเคราะห์ข้อมูลเบื้องต้นว่ามีข้อมูลใดอยู่บ้าง ต้องการข่าวสารอะไรจากแหล่งข้อมูลเหล่านั้นและเป็นการกำหนดว่าเมื่อใดจะใช้ดาต้าไมน์นิ่ง ในการแก้ปัญหา

2) การจัดเตรียมข้อมูล (Data Preparation) เป็นขั้นตอนที่มีความสำคัญและใช้เวลานานที่สุด เนื่องจากต้องมีการคัดเลือกข้อมูลที่เหมาะสมและอยู่ในประเด็นที่ต้องการ โดยประกอบด้วยขั้นตอนย่อย 3 ขั้นตอน

2.1 การคัดเลือกข้อมูล (Data Selection) เป็นการกำหนดรูปแบบข้อมูลที่ต้องการ ระบุลักษณะข้อมูลและเลือกข้อมูลที่ต้องการและนำข้อมูลที่ไม่ต้องการออก เป็นการเริ่มต้นของการเตรียมการไมน์นิ่ง โดยการเลือกข้อมูลขึ้นอยู่กับวัตถุประสงค์ทางธุรกิจ ไม่ว่าจะเป็นการเลือกตัวแปรความสัมพันธ์จำเป็นต้องเข้าใจความหมาย ประเภทข้อมูล และ

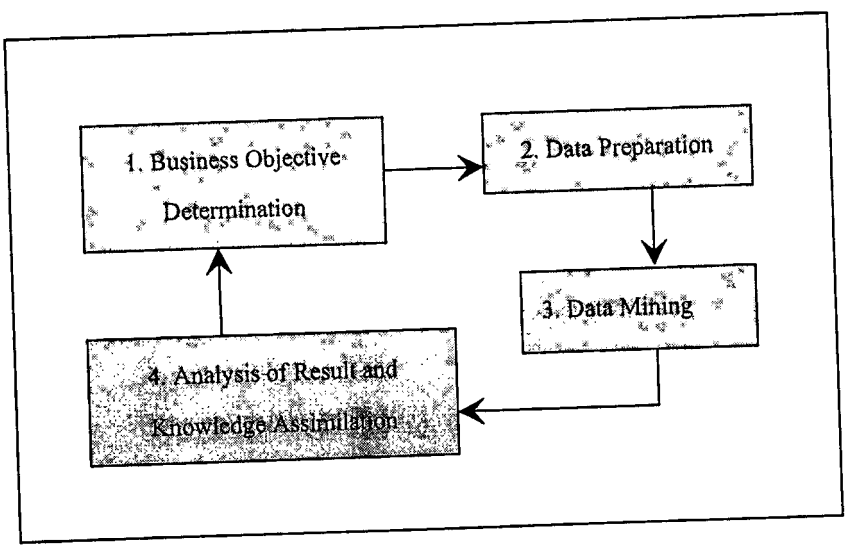
ค่าที่สามารถเป็นไปได้ด้วย การเลือกข้อมูลต้องคำนึงถึงอายุของข้อมูล โดยตัวแปรข้อมูลมี 2 ลักษณะ คือ Categorical ซึ่งเป็นตัวแปรที่ขึ้นกับค่าที่เก็บแบบมีลำดับข้อมูล และ Quantitative ซึ่งเก็บค่าเป็นเลขจำนวนจริง

2.2 การตรวจสอบข้อมูล (Data Preprocess) เป็นขั้นตอนการประมวลผลข้อมูลเบื้องต้นจากเทคนิคที่เลือกไว้แล้ว

2.3 การเปลี่ยนรูปแบบข้อมูล (Data Transformation) เป็นกระบวนการในการปรับขอบเขตข้อมูลให้อยู่ในช่วงที่เหมาะสมต่อการนำไปใช้ในการวิเคราะห์แบบจำลอง โดยปรับให้อยู่ในรูปแบบที่ตรงกับอัลกอริทึมของดาต้าไมน์นิ่งโดยแบ่งตามชนิดของปัญหา

3) ดาต้าไมน์นิ่ง (Data Mining) เลือกวิธีการและอัลกอริทึมที่เหมาะสมมาใช้กับข้อมูลที่ได้เตรียมไว้ เพื่อให้ได้ผลลัพธ์ตามความต้องการของวัตถุประสงค์หรือปัญหาที่ต้องการแก้ไข ซึ่งขั้นตอนนี้มีความสัมพันธ์กับการวิเคราะห์ข้อมูลและขั้นตอนที่ผ่านมา โดยอาจมีการย้อนกลับไปทำขั้นตอนที่ 2 ใหม่ ซึ่งขั้นตอนนี้จะมีความเกี่ยวข้องกับแบบจำลองและอัลกอริทึมหลายๆ แบบ

4) การวิเคราะห์ผลลัพธ์และการรวบรวมข้อมูลเพื่อนำไปใช้ (Analysis of Result and Knowledge Assimilation) เป็นการนำผลลัพธ์ที่ได้จากขั้นตอนดาต้าไมน์นิ่ง มาวิเคราะห์เพื่อนำไปประยุกต์ใช้กับงานต่างๆ ที่ต้องการ โดยการวิเคราะห์นั้นจะต้องมีประสบการณ์และทักษะในเชิงธุรกิจด้วย เพื่อที่จะได้ประโยชน์สูงสุด



รูปที่ 2.1 กระบวนการทำงานของ Data Mining

หลังจากที่ได้ทำทุกขั้นตอนเสร็จแล้ว ควรมีการย้อนกลับไปทำในทุกๆ ขั้นตอนใหม่ เพื่อเป็นการตรวจสอบความถูกต้องของผลลัพธ์ที่ได้ออกมา และยังเพิ่มประสิทธิภาพของผลลัพธ์ ดังรูปที่ 2.1 เพราะถ้าทำเพียง 1 รอบ อาจจะมีข้อผิดพลาดเกิดขึ้นได้ เช่น การกำหนดวัตถุประสงค์ผิดหรือเลือกเทคนิคการทำดาต้าไมน์นิ่ง ที่ไม่เหมาะสมกับวัตถุประสงค์

2.4 การจำแนกประเภทข้อมูล (Classification)

Classification เป็นเทคนิคหนึ่งของดาต้าไมน์นิ่ง (Data Mining) ที่ใช้ใน Predictive Modeling ซึ่งสามารถสร้างแบบจำลองการจำแนกประเภทข้อมูลได้จากกลุ่มตัวอย่างของข้อมูลที่ได้กำหนดไว้ก่อนล่วงหน้า และสามารถพยากรณ์กลุ่มของรายการที่ยังไม่เคยนำมาจำแนกข้อมูลได้ด้วย ซึ่งเป็นเทคนิคการทำนายว่าสิ่งที่เราสนใจจะอยู่ในกลุ่มใด โดยจะเห็นว่าการแบ่งระดับเป็นการจำแนกคลาสของเรคคอร์ดในข้อมูล โดยมีการกำหนดค่าของคลาสไว้ก่อนล่วงหน้า โดยมีขั้นตอนการจำแนกประเภทข้อมูลดังนี้

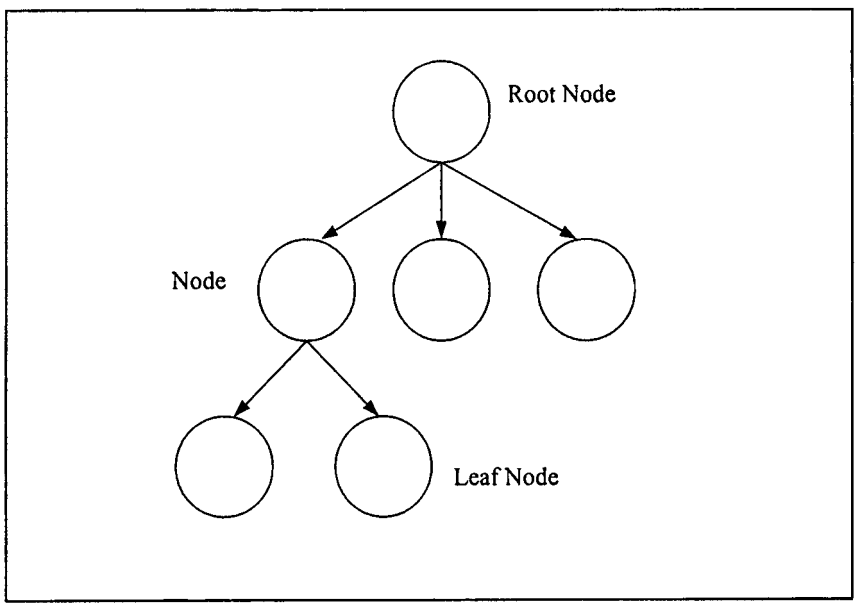
1) Training Phase เป็นขั้นตอนการเรียนรู้ที่นำกลุ่มข้อมูลตัวอย่าง (Training Data Set) ซึ่งเป็นกลุ่มข้อมูลที่ใช้ในการสร้างแบบจำลอง นำมาทำการวิเคราะห์โดยใช้อัลกอริทึมของ Classification เพื่อทำการเปรียบเทียบการเรียนรู้และทำการสร้างแบบจำลองที่จะสามารถอธิบายถึงลักษณะความสัมพันธ์ของกลุ่มข้อมูลที่ซ่อนอยู่ภายในฐานข้อมูล ซึ่งแบบจำลองนี้จะมีลักษณะของกลุ่มข้อมูลที่ถูกแจกแจงออกเป็นคลาสต่างๆ ด้วย Classification Rule และในแต่ละคลาสจะมีลักษณะเฉพาะกลุ่มที่สามารถสรุปออกมาเป็นรูปแบบความสัมพันธ์ได้ โดยแบบจำลองสามารถถูกนำเสนอได้หลากหลายรูปแบบ เช่น ในรูปของกฎการจำแนกประเภทข้อมูลแบบโครงสร้างต้นไม้ (Decision Tree) หรือเป็นสมการทางคณิตศาสตร์

2) Testing Phase เป็นขั้นตอนการทดสอบข้อมูลโดย Test Data จะถูกนำมาทดสอบเพื่อดูความถูกต้องของ Classification Rule ที่ถูกสร้างขึ้นมาจากขั้นตอนการ Training ซึ่งข้อมูลที่นำมาทดสอบส่วนใหญ่จะเป็นข้อมูลที่ทราบผลอยู่แล้วว่าเมื่อได้ทดสอบแล้วจะได้คำตอบอย่างไรเพราะในขั้นตอนนี้จะเป็นการพิจารณาว่า Classification Rule ที่สร้างขึ้นมีความถูกต้องน่าเชื่อถือและเหมาะสมที่จะสามารถนำไปใช้งานได้หรือไม่

สำหรับเทคนิคที่นิยมใช้ใน Classification นั้นมีอยู่หลากหลายเทคนิค ตัวอย่างเช่น Decision Tree, Neural Networks, Bayesian Networks เป็นต้น

2.5 ดิซิชั่นทรี (Decision Tree)

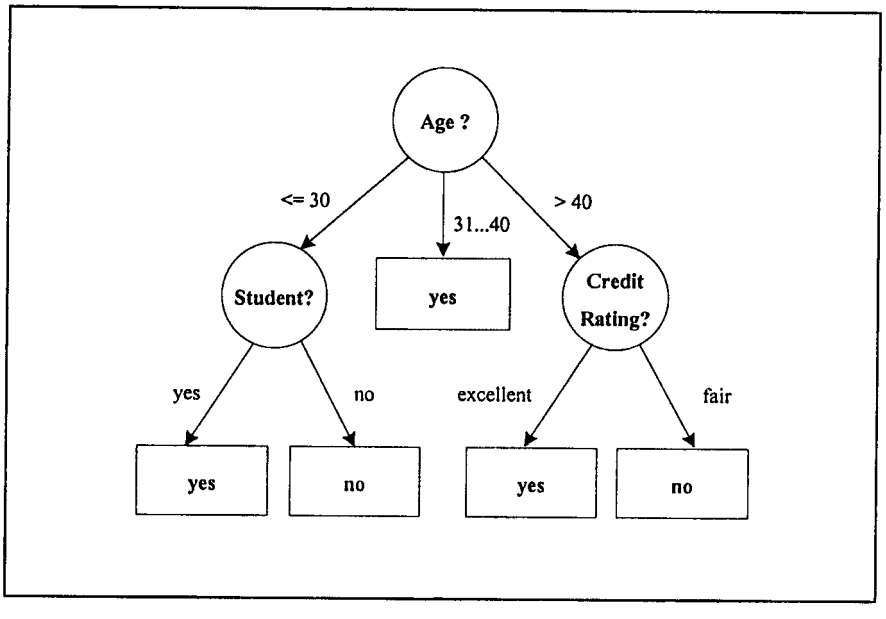
ดิซิชั่นทรี (Decision Tree) หมายถึงโครงสร้างที่แสดงผลอยู่ในรูปแบบของแผนภูติต้นไม้ เพื่อช่วยในการตัดสินใจ โดยโครงสร้างของดิซิชั่นทรีจะประกอบด้วยโหนด (Node) ต่างๆ โดยโหนดแรกสุดจะเรียกว่าโหนดเริ่มต้น (Root Node) แล้วแตกออกไปเป็นโหนดลูก (Child Node) โดยที่โหนดลูกอาจแตกออกเป็นโหนดต่อไปได้อีก โดยโหนดลูกแต่ละระดับอาจมีมากกว่า 2 โหนดได้ ส่วนโหนดที่อยู่ระดับสุดท้ายจะเรียกว่าโหนดปลาย (Leaf Node) โดยแต่ละโหนดภายใน (Internal Node) แทนเอตทริบิวท์ทดสอบ แต่ละกิ่ง (Branch) ของดิซิชั่นทรีจะแสดงถึงผลที่เกิดจากการทดสอบ และโหนดปลายที่อยู่ระดับสุดท้ายจะเป็นกลุ่มของข้อมูลที่ถูกจำแนกจากโหนดเริ่มต้นจนถึงโหนดปลายจะมีเพียงเส้นทางเดียวเท่านั้น ซึ่งเส้นทางที่ได้จะอธิบายถึงกฎที่ใช้สำหรับการจำแนกประเภทข้อมูลแต่ละกลุ่มดังตัวอย่างรูปที่ 2.2



รูปที่ 2.2 ตัวอย่างของดิซิชั่นทรี

ดิซิชั่นทรีจะทำงานโดยสร้างกฎในรูปแบบของแผนภูมิต้นไม้ โดยโหนดเริ่มต้นจะเป็นเงื่อนไขบนสุดที่ถูกเปรียบเทียบก่อน และคำตอบที่ได้จะอ้างถึงโหนดลูกในชั้นถัดไปซึ่งในแต่ละการทดสอบจะระบุเงื่อนไขออกมา โดยการประมวลผลของการทดสอบเงื่อนไขและการอ้างถึงโหนดลูก จะถูกวนซ้ำจนกระทั่งเข้าถึงโหนดปลายที่เป็นเงื่อนไขสุดท้าย ซึ่งโหนดปลายจะแสดงถึงประเภทของคลาสที่แบ่งแยกได้ นั่นก็คือความรู้หรือผลลัพธ์ที่ได้จากการสร้างดิซิชั่นทรี

จากรูปที่ 2.3 แสดงแนวโน้มของการขายคอมพิวเตอร์ ซึ่งจะทำนายว่าลูกค้าน่าจะซื้อคอมพิวเตอร์หรือไม่ จากตัวอย่างจะเห็นว่ามี 2 คลาส คือ "yes" และ "no" ซึ่งเป็นผลลัพธ์สุดท้ายที่ได้จากการทดสอบ เพื่อการแยกตัวอย่างที่เราไม่รู้ ค่าของแอตทริบิวท์ของตัวอย่างจะถูกทดสอบกับดิซิชั่นทรีโดยเส้นทางเริ่มจากโหนดเริ่มต้น ไปยังโหนดปลายซึ่งมีการทำนายคลาสของตัวอย่างนั้นอยู่ ซึ่งในตัวอย่างนี้การทดสอบจะเริ่มจากการพิจารณาอายุก่อน โดยแบ่งเป็นช่วงอายุน้องกว่าหรือเท่ากับ 30 ปี อายุ 31 – 40 ปี และอายุ 40 ปีขึ้นไป ซึ่งหากมีอายุน้อยกว่าหรือเท่ากับ 30 ปี จะพิจารณาต่อไปว่าเป็นนักเรียนหรือไม่ หากใช่ก็มีแนวโน้มว่าต้องการซื้อคอมพิวเตอร์ กลุ่มผู้ที่มีอายุ 31 – 40 ปีมีแนวโน้มว่าต้องการซื้อคอมพิวเตอร์ และกลุ่มผู้ที่มีอายุมากกว่า 40 ปีจะพิจารณาต่อไปว่ามีอัตราเครดิตเป็นอย่างไร ถ้ามีเครดิตดีจะเป็นมีแนวโน้มว่าต้องการซื้อคอมพิวเตอร์และถ้ามีเครดิตในระดับพอใช้จะเป็นมีแนวโน้มว่าไม่ต้องการซื้อคอมพิวเตอร์



รูปที่ 2.3 Decision Tree แนวโน้มการขายคอมพิวเตอร์

นอกจากนี้ดิซิชั่นทรียังสามารถแปลงให้เป็นกฎการจำแนกประเภทข้อมูล IF- THEN ได้ด้วย ซึ่งจาก ตัวอย่างดิซิชั่นทรีแนวโน้มการขายคอมพิวเตอร์ ในรูปที่ 2.3 สามารถแปลงให้อยู่ในรูปของกฎการจำแนกประเภทข้อมูล IF- THEN ได้ดังนี้

IF *Age* = *"< = 30"* AND *Student* = *"no"* THEN *Buys Computer* = *"no"*

IF *Age* = *"< = 30"* AND *Student* = *"yes"* THEN *Buys Computer* = *"yes"*

IF *Age* = *"31...40"* THEN *Buys Computer* = *"yes"*

IF *Age* = *"> 40"* AND *Credit Rating* = *"excellent"* THEN *Buys Computer* = *"no"*

IF *Age* = *"> 40"* AND *Credit Rating* = *"fair"* THEN *Buys Computer* = *"yes"*

การทำงานของดิซิชั่นทรีนั้นสามารถใช้ข้อมูลในรูปแบบมาตรฐานได้โดยไม่จำเป็นต้องทำการ Normalized และสามารถทำงานได้กับปัญหา Missing Value โดยในการทำงานจริงนั้นควรจะมีการลดขนาดของข้อมูลโดยใช้เทคนิคการเลือกข้อมูลเพื่อทำให้ Tree นั้นทำงานได้เร็วมากขึ้น โดยสาเหตุที่ทำให้เกิดความล่าช้าคือการเรียงลำดับค่าของแอตทริบิวท์ต่างๆ ซึ่งการเรียงลำดับข้อมูลและการลดข้อมูลสามารถทำให้การหาค่าดิซิชั่นทรีจะมีประสิทธิภาพในการอธิบายถึงสาเหตุที่มาของคำตอบ แต่ก็ยังมีปัญหาการเข้าถึงข้อมูลที่แตกย่อยมากเกินไปหรือที่เรียกว่า Overfitting ซึ่งหมายถึงเมื่อดิซิชั่นทรี ได้ทำการเรียนรู้ไปสักระยะหนึ่งแล้วจะให้ค่าที่ดีที่สุด แต่ถ้าเลยจุดนั้นไปแล้วอาจทำให้ได้ค่าที่ด้อยกว่าเดิม แต่สามารถแก้ไขได้โดยการใช้วิธีการ Pruning ซึ่งเป็นวิธีการกำจัดกิ่งที่คิดว่าไม่จำเป็นหรือส่วนของข้อมูลที่ไม่เกี่ยวข้องออกไป ซึ่งส่วนนี้เกิดจากข้อมูลที่ใช้ในการสร้างดิซิชั่นทรีบางส่วนมีข้อผิดพลาด เพราะข้อมูลที่ผิดปรกตินี้จะปรากฏให้เห็นหากทำการสร้างดิซิชั่นทรีที่มีขนาดใหญ่เกินไป หลังจากกำจัดกิ่งที่คิดว่าไม่จำเป็นออกแล้วทำการตรวจสอบว่าได้ค่าที่ดีกว่าเดิมหรือไม่ ถ้าดีกว่าก็ให้ตัดกิ่งนั้นออกไป แต่วิธีการนี้จะมีปัญหาหากนำไปใช้กับข้อมูลที่มีจำนวนเรคคอร์ดน้อยเกินไป เพราะอาจจะทำให้มีความน่าเชื่อถือน้อยลง โดยเทคนิคการ Pruning สามารถแบ่งได้ 2 เทคนิคคือ Pre Pruning และ Post Pruning

1) Pre Pruning จะทำการกำจัดกิ่งโดยการหยุดการแตกกิ่งตั้งแต่ในช่วงการสร้างดิซิชั่นทรี ซึ่งจำทำให้โหนดที่จะแตกต่อไปกลายเป็น Leaf Node โดยปกติจะมีการวัดค่าทางสถิติเพื่อบอกถึงความเหมาะสมของการแตกกิ่ง ถ้าพบว่าผลของการเลือกค่าที่ดีที่สุดในแต่ละกิ่งน้อยกว่าค่า Threshold ที่ตั้งไว้ การแตกกิ่งก็จะถูกหยุดลง ซึ่งเป็นเรื่องยากที่จะกำหนดค่า Threshold ให้มีความเหมาะสม

2) Post Pruning จะทำการกำจัดกิ่งออกจากดิซิชั่นทรีที่สร้างเสร็จแล้ว โดยในการกำจัดบางกิ่งออกจะสามารถหาได้จากการคำนวณค่าความผิดพลาด (Expect Error Rate) เมื่อ Sub Tree ใต้โหนดนั้นถูกตัดออกไป ซึ่งถ้าการ Pruning ทำให้ค่าความผิดพลาดสูงขึ้น Sub Tree นั้นจะถูกตัดออกไปแต่หากทำให้เกิดค่าที่ต่ำลงก็จะเก็บกิ่งนั้นเอาไว้ และหลังจากการทำ Pruning แล้ว Test Set ที่ได้จะนำมาหาค่าความแม่นยำ โดย Test Set ของดิซิชั่นทรีที่ได้ค่าความผิดพลาดต่ำสุด หรือ ค่าความแม่นยำสูงสุดก็จะถูกนำมาเลือกใช้

2.6 การสร้างดิซิชั่นทรี

การสร้างดิซิชั่นทรีทำได้โดยการใช้อัลกอริทึมที่อ้างอิงจาก Top-down Induction of Decision Tree (TDIDT) เพราะความรู้ที่ได้มาจากกลุ่มข้อมูลซึ่งอยู่ในรูปแบบของ Top-down กฎหรือเงื่อนไขที่ถูกเลือกกฎแรกจะเป็น Root Node และจากนั้นก็จะมีการวนซ้ำไปเรื่อยๆ จนได้คุณสมบัติสุดท้ายที่ต้องการ (Target Attribute)

องค์ประกอบหลักในการสร้างดิซิชั่นทรี ประกอบด้วย

1) Choosing splitting attribute คือการเลือกแอตทริบิวท์ที่จะ Slit จะมีผลกระทบต่อประสิทธิภาพที่จะนำดิซิชั่นทรีไปใช้ในการจำแนกข้อมูล แต่ละแอตทริบิวท์อาจมีความความสำคัญต่อการจำแนกข้อมูลมากน้อยแตกต่างกันออกไป ซึ่งบางแอตทริบิวท์อาจมีความสำคัญน้อยมาก จึงทำให้ไม่ได้นำมาพิจารณาในการสร้างดิซิชั่นทรี ซึ่งในการคิดหาค่าความสำคัญของแต่ละแอตทริบิวท์จะมีหลักเกณฑ์ในการคิดแตกต่างกันออกไปในแต่ละอัลกอริทึม ซึ่งหลักนี้ถือเป็นจุดเด่นของ Decision Tree

2) Spits เมื่อเลือกแอตทริบิวท์ที่จะใช้แตกได้แล้วก็ทำการแตกกิ่งตามเส้นทางที่ได้

3) Tree Structure ในการแตกกิ่งของแต่ละแอตทริบิวท์อาจมีมากกว่า 2 ทางเลือก โดยในการพิจาณานี้จะขึ้นอยู่กับ โครงสร้างของดิซิชั่นทรีด้วย เช่น หากเป็น Binary Tree จะมีการแตกกิ่งออกเป็น 2 กิ่งเท่านั้น

4) Training data เป็นการเรียนรู้ดิซิชั่นทรีที่สร้างขึ้น เพื่อดูค่าความผิดพลาดที่เกิดขึ้น ซึ่งส่งผลต่อความถูกต้องของดิซิชั่นทรีที่สร้างขึ้น

5) Pruning นำค่าความผิดพลาดไปใช้พิจารณาการกำจัดกิ่งที่ทำให้เกิดความผิดพลาดหรือกิ่งที่ไม่จำเป็นต่อดิซิชั่นทรีออก

ดิซิชั่นทรีมีอัลกอริทึมอยู่หลายแบบให้สามารถเลือกใช้ โดยแต่ละอัลกอริทึมจะแตกต่างกันที่หลักในการสร้างและการเลือกพารามิเตอร์ที่จะทำการแตกกิ่งเพื่อที่จะสร้างดิซิชั่นทรี รวมทั้งหลักการ Pruning ตัวอย่างของอัลกอริทึมที่ใช้ได้แก่

- CHAID : Chi square-Automatic-Interaction-Detection
- CART : Classification and Regression Trees
- ID3 : Induction Decision Tree โดย Quilan
- C4.5 : Decision Tree Induction Algorithm ซึ่งพัฒนาต่อมาจาก ID3
- SPRINT : A Scalable Parallel Classifier for Data Mining
- SLIQ : A Fast Scalable Classifier for Data Mining

2.7 คาร์ทอัลกอริทึม (CART Algorithm)

CART มาจาก Classification and Regression Tree ซึ่งพัฒนาโดย Leo Breiman, Jerome H. Friedman, Richard A. Olshen และ Charles J. Stone ในปี ค.ศ. 1984 ซึ่งเป็นวิธีในการสร้างโครงสร้างต้นไม้เพื่อช่วยในการตัดสินใจ หรือที่เรียกว่า ดิซิชั่นทรี (Decision Tree) โดยดิซิชั่นทรีที่ถูกสร้างขึ้นจากอัลกอริทึม CART จะอยู่ในรูปแบบของ Binary Tree ที่ประกอบด้วย 2 กิ่ง (Branch) ของแต่ละโหนดการตัดสินใจ โดยอัลกอริทึมนี้จะใช้กฎในการจำแนกข้อมูลที่จะทำการเรียนรู้ ซึ่งในการเลือกคุณสมบัติหรือแอตทริบิวท์ที่มีความสำคัญเพื่อใช้เป็นกฎแรกในการจำแนกข้อมูล ซึ่งในการพิจารณาแอตทริบิวท์ในการสร้างดิซิชั่นทรีของอัลกอรึทึม CART มีหลักเกณฑ์ในการแตกกิ่ง (Splitting Rule) ด้วยกัน 2 วิธี คือ Gini Criterion และ Twoing Criterion โดยในการพัฒนาระบบงานนี้จะใช้วิธี Twoing Criterion ที่มีกระบวนการในการสร้างดิซิชั่นทรีโดยพิจารณาจากแอตทริบิวท์ที่มีความสำคัญมากที่สุดก่อน และทำซ้ำไปเรื่อยๆ จนกว่าตัวอย่างข้อมูลในแต่ละส่วนจะขึ้นกับคลาสใดคลาสหนึ่ง ซึ่งขั้นตอนการทำงานสามารถอธิบายได้ดังนี้

Partition (Data)

if (All points in S are of the same classes) **then**

return;

for each attribute A **do**

evaluate splits on attribute A;

Use best split found to partition S into S_1 and S_2

Partition(S_1);

Partition(S_2);

หลักในการแตกกิ่ง (Splitting Rule) ด้วยวิธี Twoing Criterion จะพิจารณาแอตทริบิวท์ที่ดีที่สุดจากการคำนวณจากสูตร (2.1)

$$\Phi(s|t) = 2P_L P_R \sum_{j=1}^{\#\text{ classes}} |P(j|t_L) - P(j|t_R)| \qquad (2.1)$$

โดยในการพิจารณาการแตกกิ่งจะเลือกแอตทริบิวท์ที่มีค่ามากที่สุดมาใช้ในการตัดสินใจ ซึ่งจะใช้สูตรนี้คำนวณซ้ำไปเรื่อยๆ โดยแต่ละครั้งจำนวนเรคคอร์ด (Record) ก็จะเปลี่ยนไปด้วย

โดย t_L = โหนดลูกทางด้านซ้ายที่โหนด t

t_R = โหนดลูกทางด้านขวาที่โหนด t

$$P_L = \frac{\text{จำนวนเรคคอร์ดของโหนดลูกทางซ้าย}}{\text{จำนวนเรคคอร์ดของข้อมูลที่ใช้}}$$

$$P_R = \frac{\text{จำนวนเรคคอร์ดของโหนดลูกทางขวา}}{\text{จำนวนเรคคอร์ดของข้อมูลที่ใช้}}$$

$$P(j \mid t_L) = \frac{\text{จำนวนของคลาส } j \text{ ที่เรคคอร์ด } t_L}{\text{จำนวนเรคคอร์ดที่ } t}$$

$$P(j \mid t_R) = \frac{\text{จำนวนของคลาส } j \text{ ที่เรคคอร์ด } t_R}{\text{จำนวนเรคคอร์ดที่ } t}$$

จากข้อมูลในจากตารางที่ 2.1 เป็นตัวอย่างการให้เครดิตลูกค้าโดยพิจารณาจากคุณสมบัติเงินออม (Savings), สินทรัพย์ (Assets), รายได้ (Incomes) เพื่อใช้ในการให้เครดิตลูกค้าสามารถนำมาใช้ในการจำแนกประเภทลูกค้าได้ 2 แบบ คือ ลูกค้าที่มีเครดิตดีและลูกค้าที่มีเครดิตไม่ดี

ตารางที่ 2.1 แสดงตัวอย่างการให้เครดิตลูกค้า

Customer	Saving	Assets	Income ($1,000s)	Credit Risk
1	Medium	High	75	Good
2	Low	Low	50	Bad
3	High	Medium	25	Bad
4	Medium	Medium	50	Good
5	Low	Medium	100	Good
6	High	High	25	Good
7	Low	Low	25	Bad
8	Medium	Medium	75	Good

ในการนำข้อมูลดังกล่าวมาสร้างเป็นดิซิชั่นทรีด้วยอัลกอริทึม CART หากมีข้อมูลที่เป็นตัวเลขจะต้องจัดเป็นกลุ่มข้อมูลใหม่ก่อน ซึ่งในแอตทริบิวท์รายได้ (Incomes) จะเห็นได้ว่าข้อมูลที่ได้เป็นตัวเลข ซึ่งในการแตกกิ่งจะต้องนำข้อมูลที่เป็นตัวเลขมาจัดเป็นกลุ่มข้อมูลใหม่โดยจากการจัดกลุ่มจะแบ่งออกเป็นช่วงรายได้ นอกจากนี้แบบจำลองที่ได้จากการสร้างเป็นดิซิชั่นทรีด้วยอัลกอริทึม CART จะเป็น Binary Tree จึงต้องมีการแบ่งกรณีที่ได้ของแต่ละคุณสมบัติออกเป็น 2 กรณีย่อยที่จะเกิดขึ้นซึ่งเมื่อนำมาพิจารณาเป็นกรณีแล้วจะได้ Candidate Split ทั้งหมด 9 กรณีดังตารางที่ 2.2

ตารางที่ 2.2 แสดง Candidate Split ในการแตกกิ่งการให้เครดิตลูกค้า

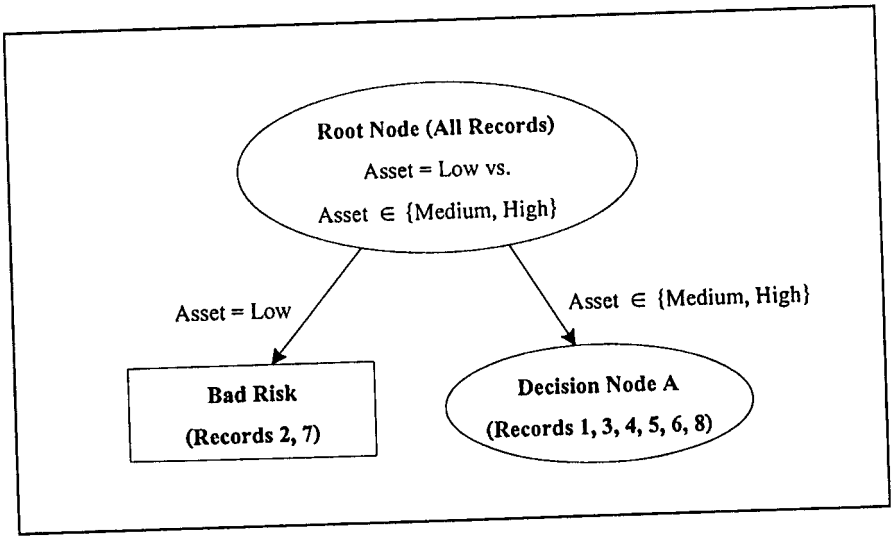
Candidate Split	Left Child Node, t_L	Right Child Node, t_R
1	Savings = low	Savings $\in$ {medium, high}
2	Savings = Medium	Savings $\in$ {low, high}
3	Savings = High	Savings $\in$ {low, medium}
4	Assets = low	Assets $\in$ {medium, high}
5	Assets = Medium	Assets $\in$ {low, high}
6	Assets = High	Assets $\in$ {low, medium}
7	Income $\leq$ \$25,000	Income $>$ \$25,000
8	Income $\leq$ \$50,000	Income $>$ \$50,000
9	Income $\leq$ \$75,000	Income $>$ \$75,000

หลังจาก Candidate Split ในการแตกกิ่งแล้วก็ทำการแทนค่าจากสูตร $\Phi(s|t)$ เพื่อนำไปหาคุณสมบัติที่จะใช้เป็นโหนดในการตัดสินใจโหนดแรก โดยในการเริ่มพิจารณาโหนดแรกจะพิจารณาจากจำนวนเรคคอร์ดทั้งหมด ซึ่งในที่นี้ก็คือ 8 เรคคอร์ด โดยนำไปแทนค่าในสูตร $\Phi(s|t)$ จะได้ค่าดังตารางที่ 2.3

ตารางที่ 2.3 แสดงการคำนวณค่าจากสูตรในการแตกกิ่งครั้งที่ 1

Split	P_L	P_R	$P(j \mid t_L)$		$P(j \mid t_R)$		$2 P_L P_R$	$\Phi(s\|t)$
1	0.375	0.625	G: .333	B: .667	G: .8	B: .2	0.46875	0.4378
2	0.375	0.625	G: 1	B: 0	G: .4	B: .6	0.46875	0.5625
3	0.25	0.75	G: .5	B: .5	G: .667	B: .333	0.375	0.1235
4	0.25	0.75	G: 0	B: 1	G: .833	B: .167	0.375	0.6248
5	0.5	0.5	G: .75	B: .25	G: .5	B: .5	0.5	0.25
6	0.25	0.75	G: 1	B: 0	G: .5	B: .5	0.375	0.375
7	0.375	0.625	G: .333	B: .667	G: .8	B: .2	0.46875	0.4378
8	0.625	0.375	G: .4	B: .6	G: 1	B: 0	0.46875	0.5625
9	0.875	0.125	G: .571	B: .429	G: 1	B: 0	0.21875	0.1877

นำคุณสมบัติที่ได้ค่า $\Phi(s|t)$ มากที่สุดมาใช้เป็นโหนดในการตัดสินใจโหนดแรก โดยใน Candidate Split ที่ 4 มีค่ามากที่สุดจะนำมาใช้เป็นโหนดเริ่มต้น ซึ่งใช้แอตทริบิวท์สินทรัพย์ (Assets) เพื่อนำมาใช้ในการพิจารณาการให้เครดิตลูกค้า ดังรูปที่ 2.4



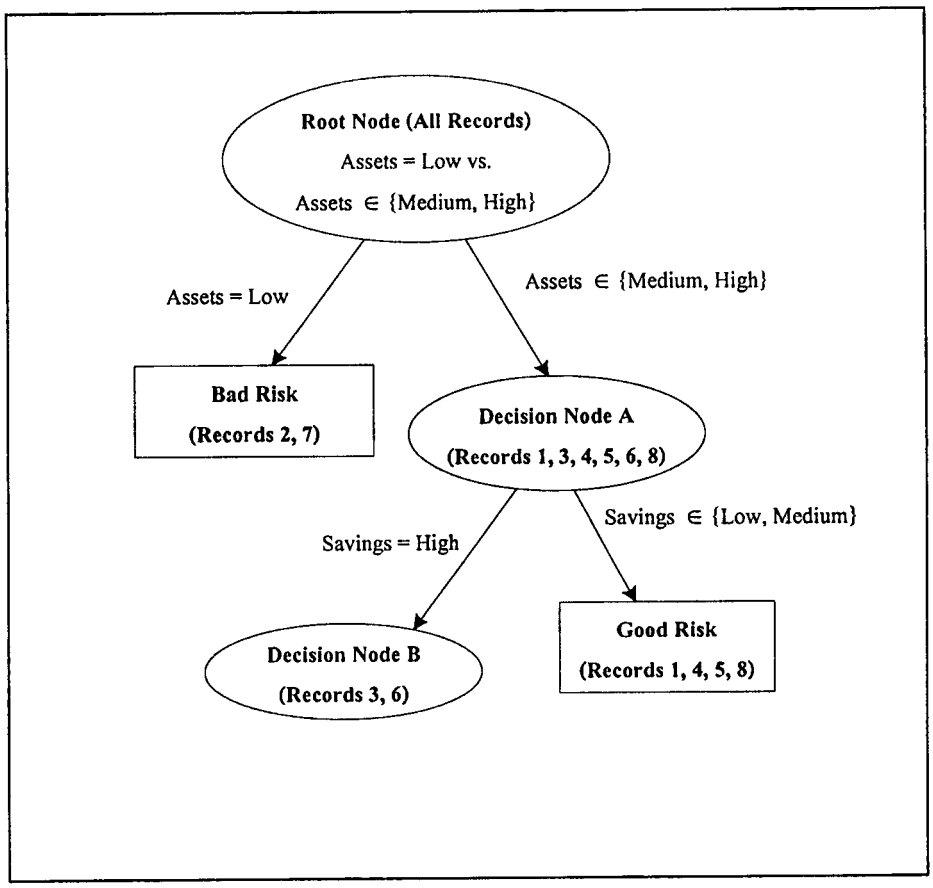
รูปที่ 2.4 การแตกโหนดแรกในการตัดสินใจของดิซิชั่นทรี

หลังจากเลือก Candidate Split ที่ 4 มาเป็น Root Node ได้แล้วก็นำ Candidate Split ที่เหลือมาคำนวณค่าใหม่ โดยตัด Candidate Split ที่นำไปพิจารณาแล้วออก โดยค่าที่ได้จะแตกต่างออกไป เพราะจากการแตกโหนด เรคคอร์ด 2 และ 7 ได้ถูกใช้ไปพิจารณาแล้ว จำนวนเรคคอร์ดที่ใช้ทั้งหมดจึงเปลี่ยนไปจากเดิม คือ 8 เรคคอร์ดเหลือเพียง 6 เรคคอร์ด ซึ่งจากการแทนสูตร $\Phi\ (s|t)$ จะได้ค่าของ Candidate Split ที่เหลือดังต่อไปนี้

ตารางที่ 2.4 แสดงการคำนวณค่าจากสูตรในการแตกกิ่งครั้งที่ 2

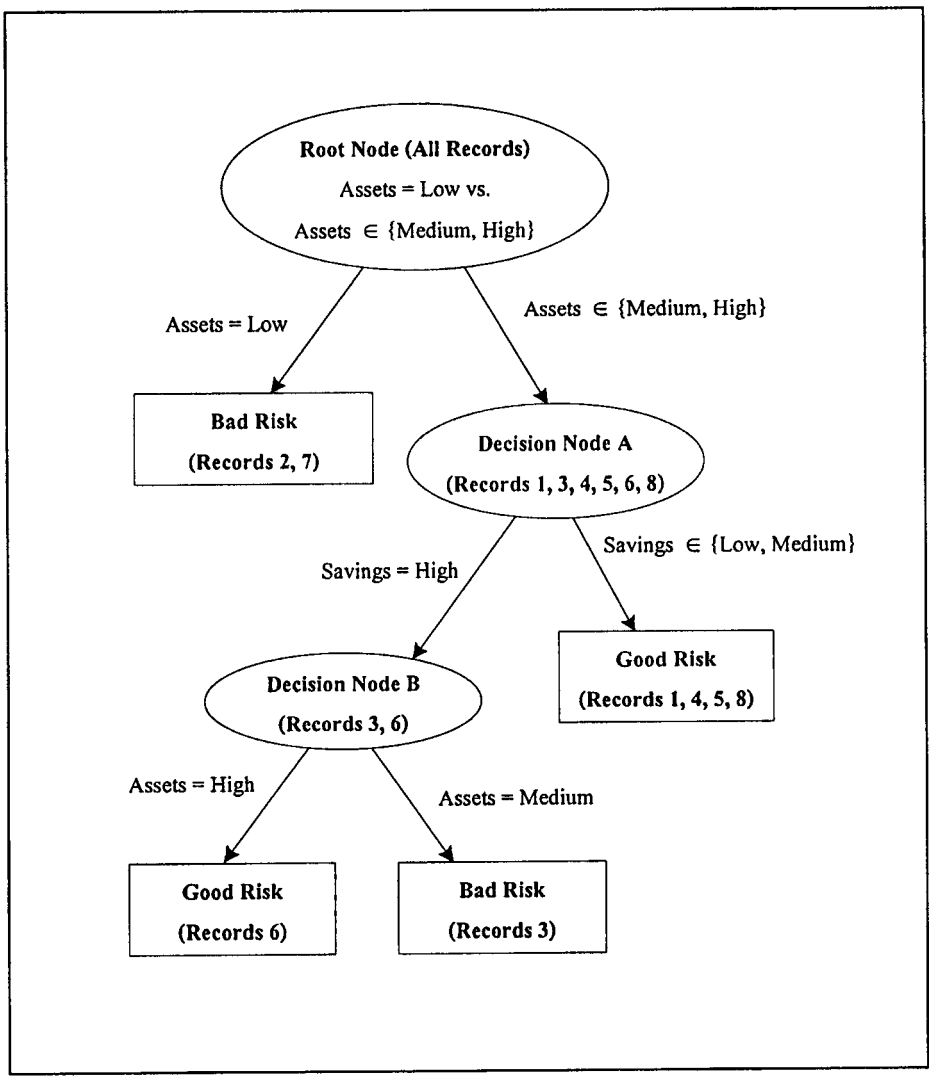
Split	P_L	P_R	$P(j \mid t_L)$		$P(j \mid t_R)$		$2P_LP_R$	$\Phi(s\|t)$
1	0.167	0.833	G: 1	B: 0	G: .8	B: .2	0.2782	0.1112
2	0. 5	0. 5	G: 1	B: 0	G: .667	B: .333	0.5	0.3333
3	0.333	0.667	G: .5	B: .5	G: 1	B: 0	0.4444	0.4444
5	0.667	0.333	G: .75	B: .25	G: 1	B: 0	0.4444	0.2222
6	0.333	0.667	G: 1	B: 0	G: .75	B: .25	0.4444	0.2222
7	0.333	0.667	G: .5	B: .5	G: 1	B: 0	0.4444	0.4444
8	0. 5	0. 5	G: .667	B: .333	G: 1	B: 0	0. 5	0.3333
9	0.167	0.833	G: .8	B: .2	G: 1	B: 0	0.2782	0.1112

นำคุณสมบัติที่ได้ค่า $\Phi\ (s|t)$ มากที่สุดมาใช้เป็นโหนดในการตัดสินใจโดยใน Candidate Split ที่ 3 และ 7 มีค่าเท่ากัน สามารถเลือก Candidate Split ใดก็ได้ จากตัวอย่างเลือก Candidate Split ที่ 3 มาใช้ในการแตกโหนด ซึ่งใช้แอตทริบิวท์เงินออม (Savings) เพื่อนำมาใช้ในการพิจารณาการให้เครดิตลูกค้า ซึ่งจะได้ดิซิชั่นทรีดังรูปที่ 2.5 และเมื่อเลือก Candidate Split ที่ 3 มาใช้ในการพิจารณาแล้วก็จะนำ Candidate Split ที่เหลือมาคำนวณค่าใหม่ โดยค่าที่นำมาคำนวณจะตัดเรคคอร์ดที่ 1 4 5 และ 8 ออกเพราะได้ถูกใช้ไปพิจารณาแล้ว ดังนั้นจำนวนเรคคอร์ดที่ใช้จึงเปลี่ยนไปจากเดิม คือ 6 เรคคอร์ดเหลือเพียง 2 เรคคอร์ด



รูปที่ 2.5 การแตกโหนดในการตัดสินใจของดิซิชั่นทรี

หลังจากนั้นก็ทำการคำนวณค่าซ้ำไปเรื่อยๆ จนกว่าจะได้โหนดสุดท้ายที่เป็นแอตทริบิวท์เป้าหมาย (Target Attribute) โดยในการคำนวณจะไม่นำ Candidate Split ที่ 4 และ 3 มาพิจารณาอีก และในการคำนวณค่าจำนวนเรคคอร์ดที่ใช้ทั้งหมดก็จะเปลี่ยนไปด้วย ซึ่งเมื่อคำนวณไปเรื่อยๆ สุดท้ายแล้วจะได้ดิซิชั่นทรีเพื่อใช้ในการให้เครดิตลูกค้า ดังรูปที่ 2.6



รูปที่ 2.6 ดิซิชั่นทรีเพื่อใช้ในการให้เครดิตลูกค้า

บทที่ 3
วิเคราะห์และออกแบบโปรแกรม

3.1 รายละเอียดของระบบ

ระบบดิซิชั่นทรีที่สร้างขึ้นจะทำงานโดยติดต่อกับ Relational Database Management Systems ซึ่งก็คือ Microsoft SQL Server เพื่อจะทำการเลือก ฐานข้อมูล ตาราง และฟิลด์ต่างๆ ที่ต้องการ นำมาใช้ในการคำนวณเพื่อสร้างโครงสร้างแผนภูมิต้นไม้ โดยอัลกอริทึมที่ใช้ในการคำนวณออกมานั้นคือ อัลกอริทึม CART โดยก่อนที่จะนำค่าต่างๆ ที่ต้องการมาคำนวณจะต้องมีการตรวจสอบข้อมูลว่ามีค่าว่างหรือไม่ และทำการแก้ไข โดยหากข้อมูลนั้นมีชนิดของข้อมูลที่เป็นตัวเลขก็จำต้องแบ่งข้อมูลออกเป็นกลุ่มย่อยแล้วแปลงข้อมูลเป็นตัวอักษร โดยหลังจากที่แปลงข้อมูลเป็นที่เรียบร้อยแล้วก็จะนำข้อมูลที่ได้มาให้คำนวณในการสร้างโครงสร้างแผนภูมิต้นไม้ต่อไป ซึ่งการทำงานของระบบ Decision Tree จะแบ่งออกเป็น 2 ส่วนคือ การสร้างแบบจำลองใหม่ (Model Building) และการทดสอบแบบจำลอง (Model Testing)

3.1.1 การสร้างแบบจำลองใหม่ (Model Building)

เป็นขั้นตอนการเรียนรู้ที่นำกลุ่มข้อมูลตัวอย่าง (Training Data Set) ซึ่งเป็นกลุ่มข้อมูลที่ใช้ในการสร้างแบบจำลองที่ผ่านการตรวจสอบไว้แล้ว (Data Preparation) มาทำการวิเคราะห์โดยใช้อัลกอริทึม CART ในการคำนวณเพื่อทำการเปรียบเทียบการเรียนรู้และทำการสร้างแบบจำลองที่สามารถอธิบายถึงลักษณะความสัมพันธ์ของข้อมูลได้ แล้วจัดเก็บแบบจำลองที่ได้ลงในฐานข้อมูลเพื่อเก็บไว้ใช้ในการทดสอบต่อไป

3.1.2 การทดสอบแบบจำลองที่ได้จากการเรียนรู้ (Model Testing)

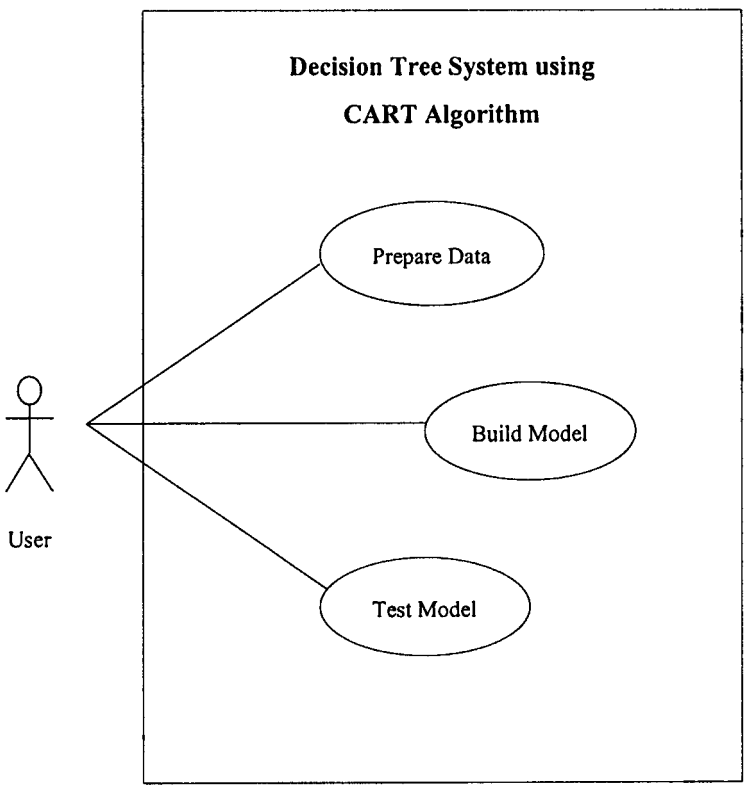
ในการทดสอบแบบจำลองที่ได้นั้นเป็นการประเมินแบบจำลอง โดยจะทำการดึงรูปแบบโครงสร้างของแบบจำลองที่ได้ทำการบันทึกลงไปในฐานข้อมูลกลับขึ้นมาแสดงผล พร้อมทั้งทำการออกแบบการติดต่อกับฐานข้อมูลเพื่อที่จะนำข้อมูลที่ใช้ในการทดสอบมาทำการแม็พกับฐานข้อมูลที่ในการสร้างแบบจำลอง หลังจากนั้นก็ทำการทดสอบและแสดงผลลัพธ์ออกมา

3.2 Process Model

ระบบดิซิชั่นทรีที่สร้างขึ้นออกแบบระบบงานโดยใช้ UML ซึ่งประกอบไปด้วยยูสเคสไดอะแกรม (Use Case Diagram) ซีเควนซ์ไดอะแกรม (Sequence Diagram) และผังการทำงานของระบบ (Flowchart)

3.2.1 Use Case Diagram

Use Case Diagram แสดงถึงผู้ใช้งานระบบและการทำงานหลักของระบบ ซึ่งการทำงานของระบบดิซิชั่นทรีได้แบ่งการทำงานหลักออกเป็น 3 ส่วนด้วยกัน คือ การเตรียมข้อมูล (Prepare Data) การสร้างโปรเจ็กต์เพื่อทำการสร้างแบบจำลอง (Build Model) และการทดสอบแบบจำลองที่ได้ทำการสร้างขึ้นมา (Test Model)



รูปที่ 3.1 Use Case Diagram ของระบบดิซิชั่นทรี

ตารางที่ 3.1 คำอธิบายยูสเคสไดอะแกรมของ Prepare Data

ยูสเคส	Prepare Data
วัตถุประสงค์	เพื่อเตรียมข้อมูลสำหรับการทำดาต้าไมน์นิ่ง
เมื่อทำงานสำเร็จ	ได้ข้อมูลสำหรับการทำดาต้าไมน์นิ่ง
Actor ที่เกี่ยวข้อง	User
อินพุต	ชุดข้อมูล
เอาต์พุต	ชุดข้อมูลที่ถูกแก้ไขแล้ว
รายละเอียด	ผู้ใช้เลือกชุดข้อมูลที่ต้องแก้ไขก่อนนำไปใช้ทำดาต้าไมน์นิ่ง

ตารางที่ 3.2 คำอธิบายยูสเคสไดอะแกรมของ Build Model

ยูสเคส	Build Model
วัตถุประสงค์	เพื่อสร้างแบบจำลองใหม่ให้กับระบบ
เมื่อทำงานสำเร็จ	ได้แบบจำลองใหม่ซึ่งอยู่ในรูปแบบโครงสร้างต้นไม้เพื่อการตัดสินใจ
Actor ที่เกี่ยวข้อง	User
อินพุต	ชุดข้อมูลที่ถูกแก้ไขแล้ว
เอาต์พุต	แบบจำลองพร้อมกับโครงสร้างข้อมูลที่ใช้สร้างแบบจำลอง
รายละเอียด	1. ผู้ใช้เลือกชุดข้อมูลและเงื่อนไขในการสร้างแบบจำลองเข้าสู่ระบบ 2. ระบบทำการแสดงผลลัพธ์ในการสร้างแบบจำลอง 3. ผู้ใช้บันทึกแบบจำลองที่ได้เพื่อใช้ในการทดสอบต่อไป

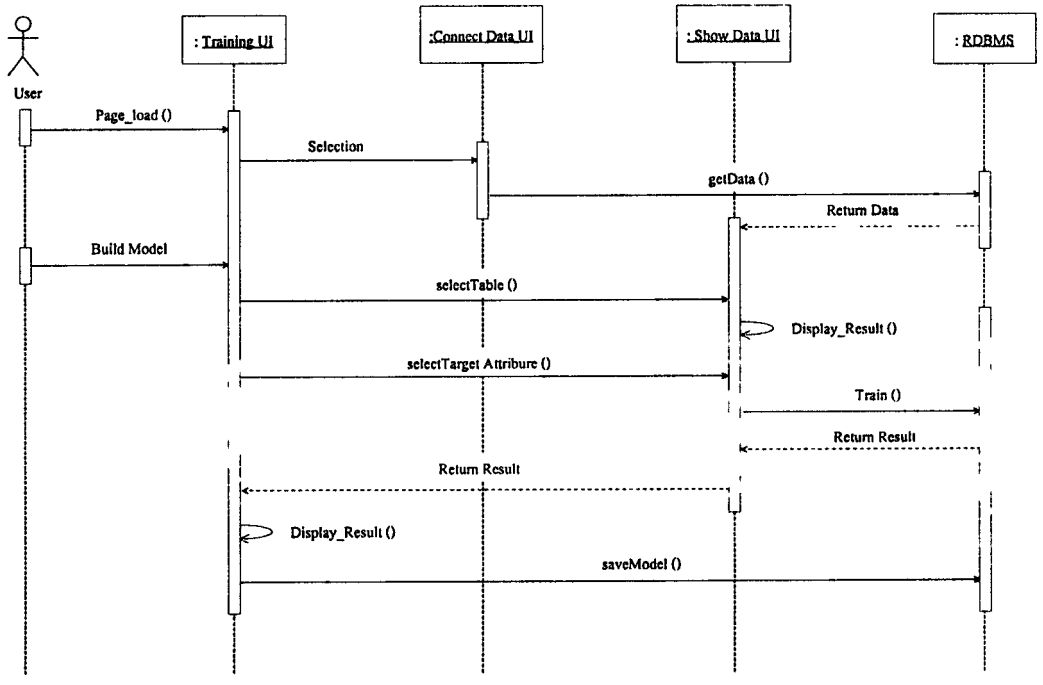
ตารางที่ 3.3 คำอธิบายยูสเคสไดอะแกรมของ Test Model

ยูสเคส	Test Model
วัตถุประสงค์	เพื่อทดสอบแบบจำลองที่มีอยู่
เมื่อทำงานสำเร็จ	แสดงผลลัพธ์ในการทดสอบแบบจำลอง
Actor ที่เกี่ยวข้อง	User
อินพุต	ชุดข้อมูลและแบบจำลองที่ใช้ในการทดสอบ
เอาต์พุต	ผลลัพธ์ของการทดสอบแบบจำลองในรูปแบบของรายงาน
รายละเอียด	1. ผู้ใช้เลือกแบบจำลองและชุดข้อมูลที่ใช้ในการทดสอบแบบจำลอง 2. ระบบทำการแสดงผลลัพธ์ในการทดสอบแบบจำลอง

3.2.2 Sequence Diagram

จากการออกแบบยูสเคสไดอะแกรม สามารถนำมาอธิบายขั้นตอนการทำงานของระบบโดยใช้ Sequence Diagram ซึ่งจะสามารถอธิบายการทำงานเป็นขั้นตอน ดังต่อไปนี้

1) Sequence Diagram ของการสร้างแบบจำลอง



รูปที่ 3.2 Sequence Diagram ของการสร้างแบบจำลอง

จากรูปที่ 3.2 นี้ถูกใช้งานโดยผู้ใช้ระบบ โดยระบบสามารถสร้างแบบจำลองการตัดสินใจโดยเรียนรู้จากข้อมูลที่ใช้ในการสร้างแบบจำลอง โดยมีขั้นตอนการทำงานดังต่อไปนี้

ขั้นตอนที่ 1 ผู้ใช้ติดต่อฐานข้อมูลเพื่อเรียกข้อมูลจากฐานข้อมูล

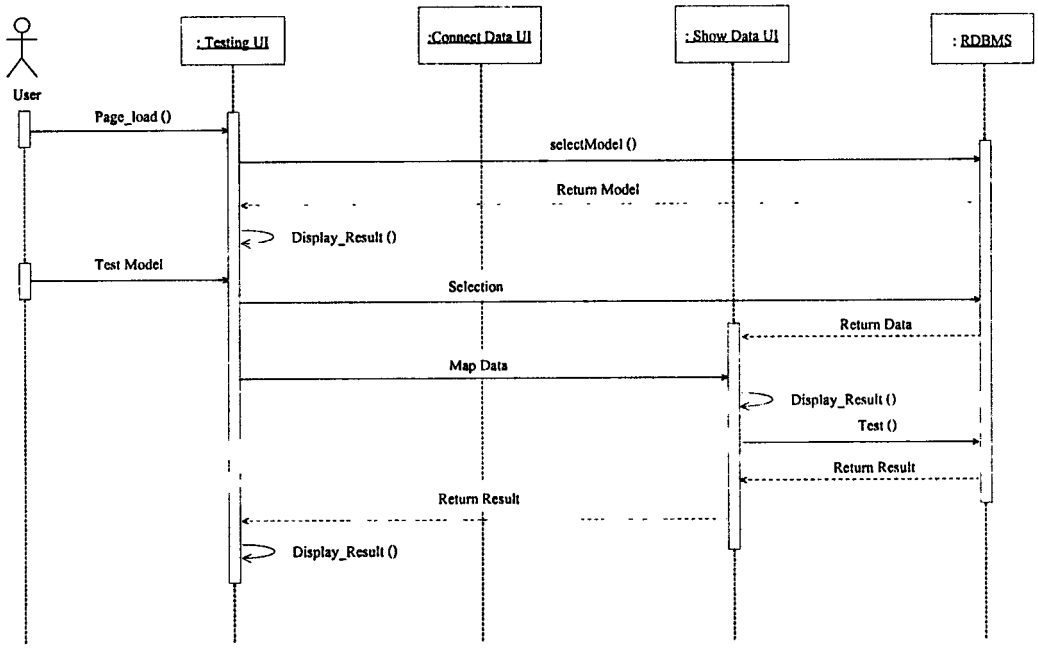
ขั้นตอนที่ 2 ผู้ใช้เลือกตารางและแอตทริบิวท์ ที่จะใช้ในการสร้างแบบจำลอง โดยข้อมูลที่เลือกจะต้องผ่านการเตรียมข้อมูลมาแล้ว โดยการแก้ไขข้อมูล ลบข้อมูลที่เป็น Null รวมทั้งเป็นผ่านการแปลงข้อมูลมาเสร็จเรียบร้อยแล้ว

ขั้นตอนที่ 3 ผู้ใช้เลือกแอตทริบิวท์เป้าหมายที่จะใช้ในการตัดสินใจ

ขั้นตอนที่ 4 ระบบทำการเรียนรู้จากข้อมูลที่ได้ แล้วแสดงผลลัพธ์ของการสร้างแบบจำลอง

ขั้นตอนที่ 5 ผู้ใช้สามารถทำการฝึกสอนจนได้แบบจำลองที่ได้ผลลัพธ์เป็นที่น่าพอใจ ผู้ใช้สามารถทำการบันทึกแบบจำลองได้

2) Sequence Diagram ของการทดสอบแบบจำลอง



รูปที่ 3.3 Sequence Diagram ของการทดสอบแบบจำลอง

จากรูปที่ 3.3 ผู้ใช้ระบบสามารถทำการทดสอบแบบจำลองที่สร้างขึ้นได้โดยใช้ข้อมูลที่ใช้ในการทดสอบ โดยข้อมูลที่ใช้นั้นต้องผ่านการเตรียมไว้สำหรับการทดสอบแบบจำลองแล้ว โดยการทดสอบมีขั้นตอนดังต่อไปนี้

ขั้นตอนที่ 1 ผู้ใช้งานระบบเลือกแบบจำลองที่จะใช้ในการทดสอบ

ขั้นตอนที่ 2 ระบบทำการโหลดแบบจำลองและโครงสร้างของแบบจำลอง

ขั้นตอนที่ 3 ผู้ใช้ทำการเลือกข้อมูลที่จะใช้ทดสอบ

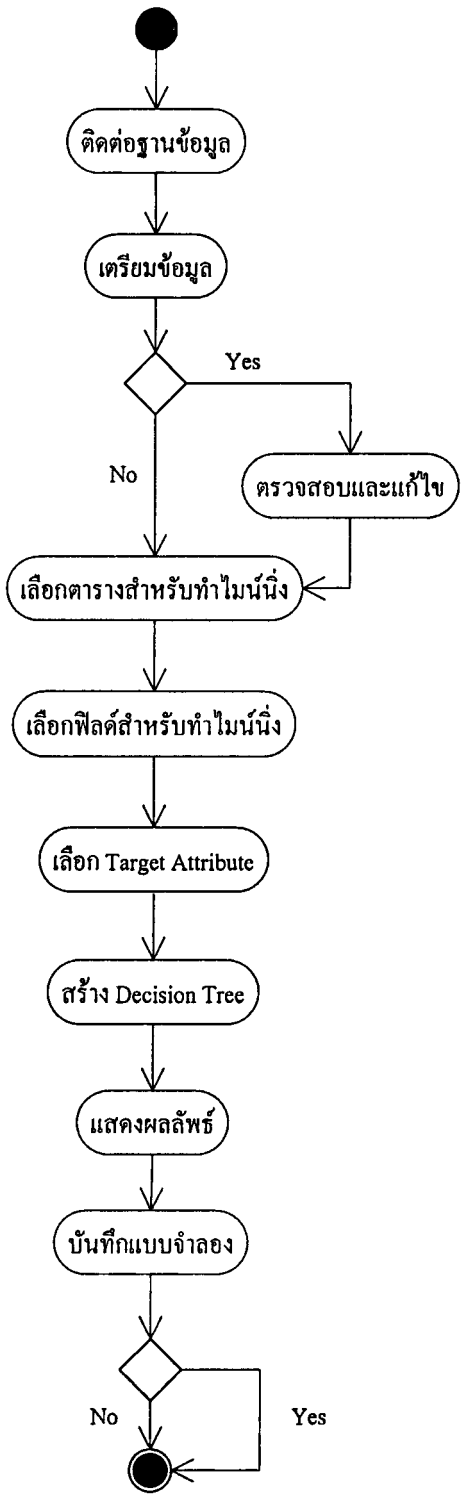
ขั้นตอนที่ 4 ผู้ใช้งานระบบทำการแม็พข้อมูลที่ใช้ในการทดสอบกับข้อมูลของแบบจำลองที่สร้างขึ้น

ขั้นตอนที่ 5 ระบบทำการทดสอบแบบจำลอง และแสดงผลการทดสอบ

3.2.3 Activity Diagram

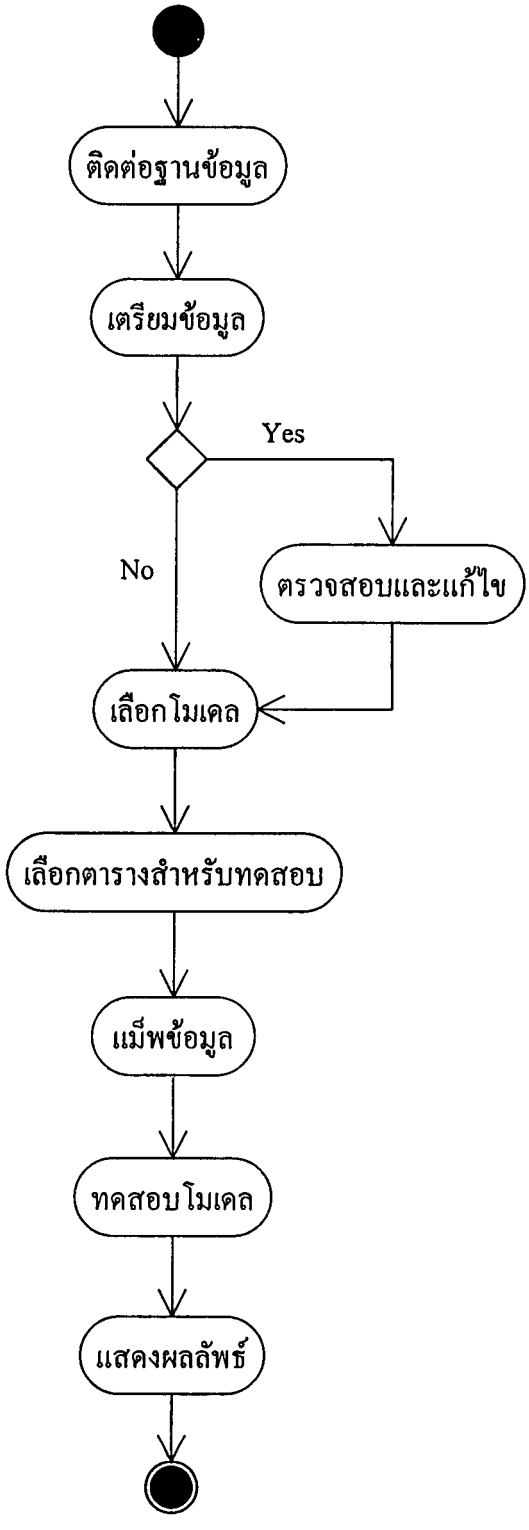
เป็นการแสดงลำดับการทำงานของระบบ Decision Tree โดยประกอบไปด้วยการสร้างแบบจำลอง (Train Model) และการทดสอบแบบจำลองที่สร้างขึ้น (Test Model)

1) Activity Diagram ของการสร้างแบบจำลอง



รูปที่ 3.4 Activity Diagram ของการสร้างแบบจำลอง

2) Activity Diagram ของการทดสอบแบบจำลอง



รูปที่ 3.5 Activity Diagram ของการทดสอบแบบจำลอง

จากรูปที่ 3.4 Activity Diagram ของการสร้างแบบจำลองโครงสร้างต้นไม้โดยใช้อัลกอริทึม CART นั้นมีขั้นตอนการทำงานดังนี้

ขั้นที่ 1 เริ่มต้นการติดต่อฐานข้อมูล

ขั้นที่ 2 เมื่อติดต่อฐานข้อมูลได้แล้วระบุแอตทริบิวท์ที่จะใช้ในการสร้างแบบจำลองเพื่อทำการเตรียมข้อมูลโดยการตรวจสอบและแก้ไขข้อมูล หากมีการเตรียมข้อมูลแล้วก็ข้ามไปยังขั้นที่ 3

ขั้นที่ 3 - 5 จากนั้นเลือกตาราง เลือกฟิลด์และแอตทริบิวท์เป้าหมาย (Target Attribute) ที่จะใช้ในการสร้างแบบจำลอง

ขั้นที่ 6-7 ระบบทำการสร้างแบบจำลองและแสดงผลลัพธ์

ขั้นที่ 8 บันทึกแบบจำละอองที่สร้างขึ้น

และรูปที่ 3.5 Activity Diagram ของการทดสอบแบบจำลองโครงสร้างต้นไม้โดยมีขั้นตอนการทำงานดังนี้

ขั้นที่ 1 เริ่มต้นการติดต่อฐานข้อมูล

ขั้นที่ 2 เมื่อติดต่อฐานข้อมูลได้แล้วระบุแอตทริบิวท์ที่จะใช้ในการสร้างแบบจำลองเพื่อทำการเตรียมข้อมูลโดยการตรวจสอบและแก้ไขข้อมูล หากมีการเตรียมข้อมูลแล้วก็ข้ามไปยังขั้นตอนที่ 3

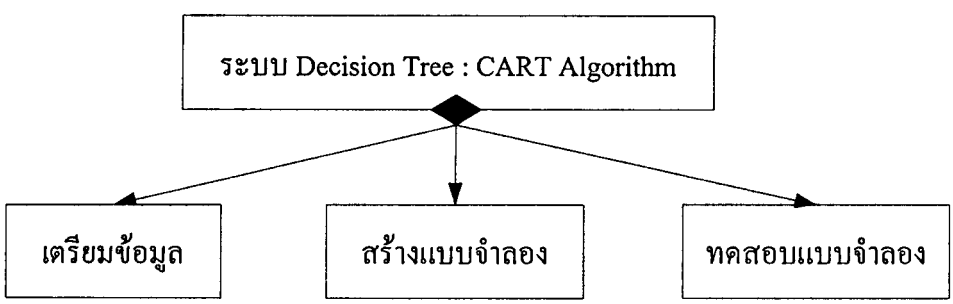
ขั้นที่ 3 จากนั้นเลือกโมเดลที่จะใช้ในการทดสอบแบบจำลอง

ขั้นที่ 4 เลือกตาราง เลือกฟิลด์ที่จะใช้ในการทดสอบแบบจำลอง

ขั้นที่ 5 ทำการแม็พข้อมูลของข้อมูลที่ใช้ในการทดสอบแบบจำลองกับข้อมูลที่ใช้ในการสร้างแบบจำลองให้ตรงกัน

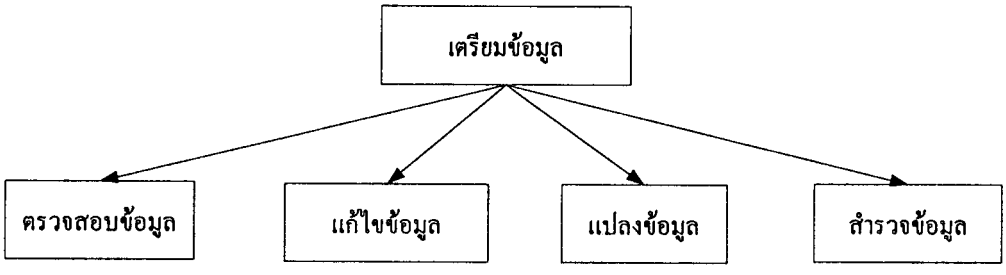
ขั้นที่ 6-7 ระบบทำการทดสอบแบบจำลองที่สร้างขึ้นแล้วแสดงผลลัพธ์

Structure Chart และ Flow – Chart แสดงการทำงาน



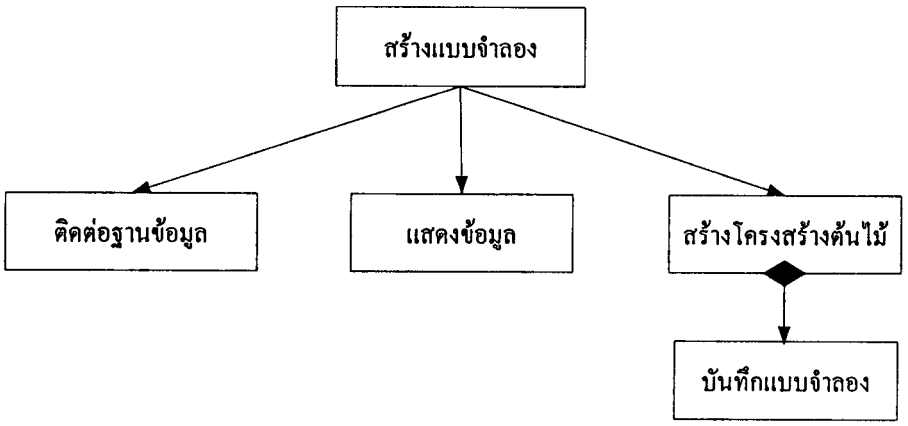
รูปที่ 3.6 Structure Chart แสดงระบบ Decision Tree : CART Algorithm

จากรูปที่ 3.6 แสดงระบบ Decision Tree : CART Algorithm ซึ่งประกอบไปด้วย 3 ส่วน ด้วยกัน คือ ส่วนของการเตรียมข้อมูล การสร้างแบบจำลองและการทดสอบแบบจำลองโดย โดยในการเตรียมข้อมูลจะเป็นการตรวจสอบและแก้ไขข้อมูลเพื่อให้ได้ข้อมูลที่พร้อมจะนำใช้ในการสร้างแบบจำลองหรือใช้ในการทดสอบแบบจำลอง การสร้างแบบจำลองใหม่นั้นจะเป็นการนำข้อมูลที่ใช้ในการเรียนรู้เพื่อใช้สร้างเป็นแบบจำลองขึ้นมาตามความต้องการของผู้ใช้งานและเมื่อสร้างแบบจำลองเสร็จผู้ใช้สามารถที่จะบันทึกแบบจำลองที่สร้างขึ้นเพื่อใช้ในการทดสอบต่อไป ส่วนการทดสอบแบบจำลองนั้นจะเป็นการนำแบบจำลองโครงสร้างต้นไม้ที่ได้ทำการสร้างและบันทึกเอาไว้มาทำการทดสอบกับข้อมูลอื่นๆ



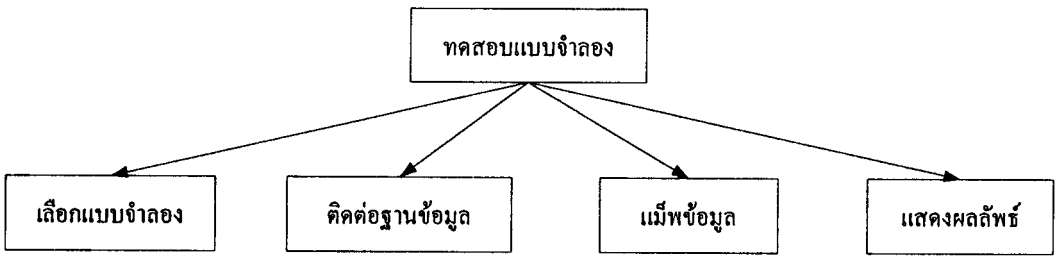
รูปที่ 3.7 Structure Chart แสดงการเตรียมข้อมูล

จากรูปที่ 3.7 กาเตรียมข้อมูลซึ่งประกอบด้วย 4 ส่วนหลัก คือการตรวจสอบข้อมูล การแก้ไขข้อมูล การแปลงข้อมูล เริ่มจากการเลือกข้อมูลที่จะนำไปใช้ในการสร้างหรือทดสอบแบบจำลอง (Data Selection) โดยระบบจะทำการตรวจสอบค่าว่างของแต่ละเรคคอร์ดเพื่อที่จะทำการแก้ไข (Data Cleaning) จากนั้นสามารถนำข้อมูลที่ได้มาปรับเปลี่ยนให้อยู่ในรูปแบบอื่นได้ด้วย (Data Transformation) เช่นการแปลงข้อมูลเป็นตัวเลข หรือการจัดกลุ่มข้อมูล เป็นต้น เสร็จแล้วก็จะเข้าสู่ขั้นตอนการสำรวจข้อมูล (Data Exploration) โดยจะแสดงรายละเอียดต่างๆ ของข้อมูลเพื่อเป็นการตรวจสอบข้อมูลที่ได้ทำการเตรียมไว้ โดยขั้นตอนการเตรียมข้อมูลจะต้องทำก่อนที่จะนำข้อมูลไปใช้ในการสร้างหรือทดสอบแบบจำลอง



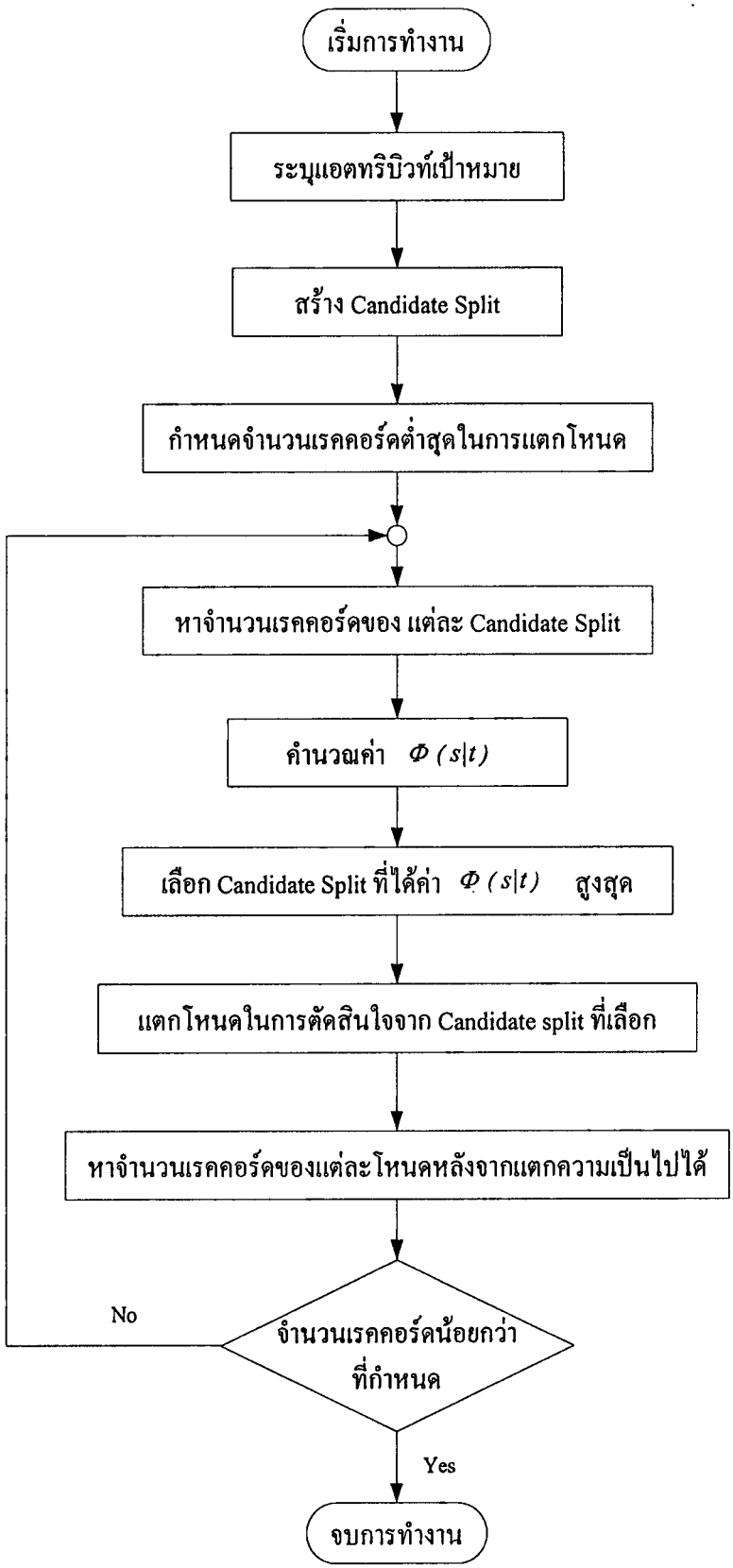
รูปที่ 3.8 Structure Chart แสดงการสร้างแบบจำลอง

ในการสร้างแบบจำลองนั้นจะเริ่มจากให้ผู้ใช้งานทำการติดต่อฐานข้อมูลที่ต้องการนำมาสร้างแบบจำลองโครงสร้างต้นไม้ โดยให้ผู้ใช้ต้องทำการเลือกตารางข้อมูล และเลือกแอตทริบิวท์จากตารางที่ได้ทำการเลือกไว้ หลังจากนั้นก็ทำการสร้างแบบจำลองและผู้ใช้สามารถบันทึกแบบจำลองที่สร้างขึ้นได้



รูปที่ 3.9 Structure Chart แสดงการทดสอบแบบจำลอง

การทดสอบแบบจำลอง ผู้ใช้ทำการเลือกแบบจำลองที่ต้องการจะทำการทดสอบ หลังจากนั้นให้ผู้ใช้ทำการติดต่อฐานข้อมูลที่ต้องการจะนำมาทดสอบกับแบบจำลองที่ได้ทำการเลือกไว้ ทำการแม็พข้อมูลระหว่างข้อมูลของแบบจำลองที่สร้างขึ้นและข้อมูลของฐานข้อมูลที่จะนำมาทดสอบ หลังจากนั้นก็จะแสดงข้อมูลที่ใช้ทดสอบพร้อมทั้งผลลัพธ์ออกมา



รูปที่ 3.10 Flow – Chart แสดงการทำงานของการสร้าง Decision Tree

จาก Flow – Chart รูปที่ 3.10 การสร้างแบบจำลองโครงสร้างต้นไม้โดยใช้อัลกอริทึม CART นั้นมีขั้นตอนการทำงานดังนี้

ขั้นที่ 1 เริ่มต้นการทำงานโดยระบุแอตทริบิวท์เป้าหมาย (Target Attribute) เพื่อใช้เป็นคุณสมบัติในการทำนายข้อมูล

ขั้นที่ 2 เมื่อระบุแอตทริบิวท์เป้าหมายได้แล้วก็จะนำข้อมูลทั้งหมดมาสร้างเป็นกรณีย่อยที่สามารถเกิดขึ้น (Candidate Split) ที่จะใช้ในการแตกโหนดแบบ Binary Tree

ขั้นที่ 3 จากนั้นกำหนดจำนวนเรคคอร์ดต่ำสุดของแต่ละโหนด (Minimum Quantity at Leaf Node) ซึ่งเป็นการกำหนดข้อมูลต่ำสุดของแต่ละโหนดที่สามารถนำแตกเป็นแบบจำลองได้ โดยถ้ามีจำนวนข้อมูล (Records) น้อยกว่าค่าที่กำหนดไว้ระบบจะทำการหยุดแตกโหนดของแบบจำลองโครงสร้างต้นไม้

ขั้นที่ 4 หาค่าจำนวนเรคคอร์ดทั้งหมดที่จะใช้ในการแตกโหนด

ขั้นที่ 5 คำนวณค่าในการพิจารณาเลือก Candidate Split จากสูตร $\Phi(s|t)$

ขั้นที่ 6 เลือก Candidate Slit ที่ได้ค่า $\Phi(s|t)$ สูงสุด ไปใช้ในการแตกโหนด

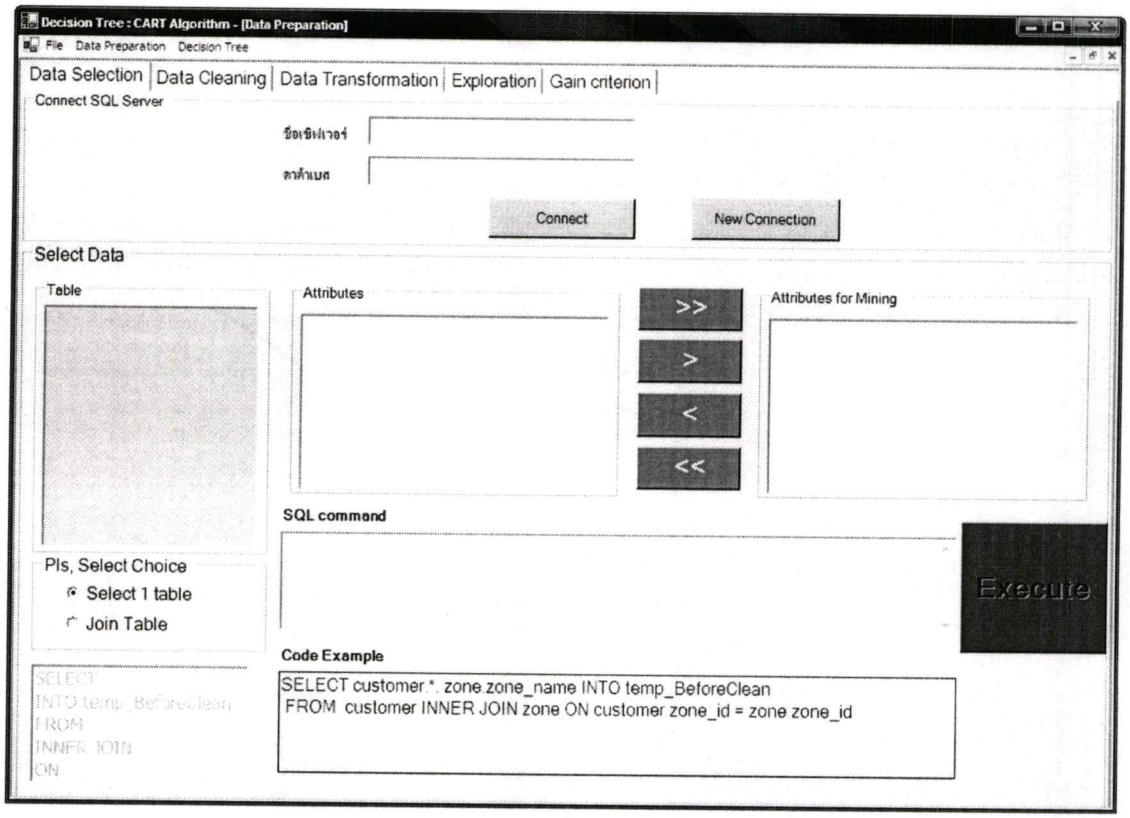
ขั้นที่ 7 ทำการแตกโหนดในการตัดสินใจ

ขั้นที่ 8 หาจำนวนเรคคอร์ดของแต่ละโหนดหลังจากแตกความเป็นไปได้ หากมีจำนวนเรคคอร์ดน้อยกว่าค่าที่กำหนดก็จะหยุดการแตกโหนด แต่หากมีจำนวนเรคคอร์ดมากกว่าที่กำหนดก็จะแตกโหนดต่อไป โดยหาจำนวนเรคคอร์ดของแต่ละ Candidate Split ที่เหลือโดยไม่นำ Candidate Split ที่ใช้แตกโหนดไปแล้วมาพิจารณาอีก จากนั้นคำนวณค่า $\Phi(s|t)$ เพื่อพิจารณาในการแตกโหนดต่อไปเรื่อยๆ จนกว่าจะได้โหนดที่มีจำนวนเรคคอร์ดน้อยกว่าที่กำหนด จึงจะเสร็จสิ้นการสร้างแบบจะลอง

บทที่ 4
การประยุกต์ใช้โปรแกรม

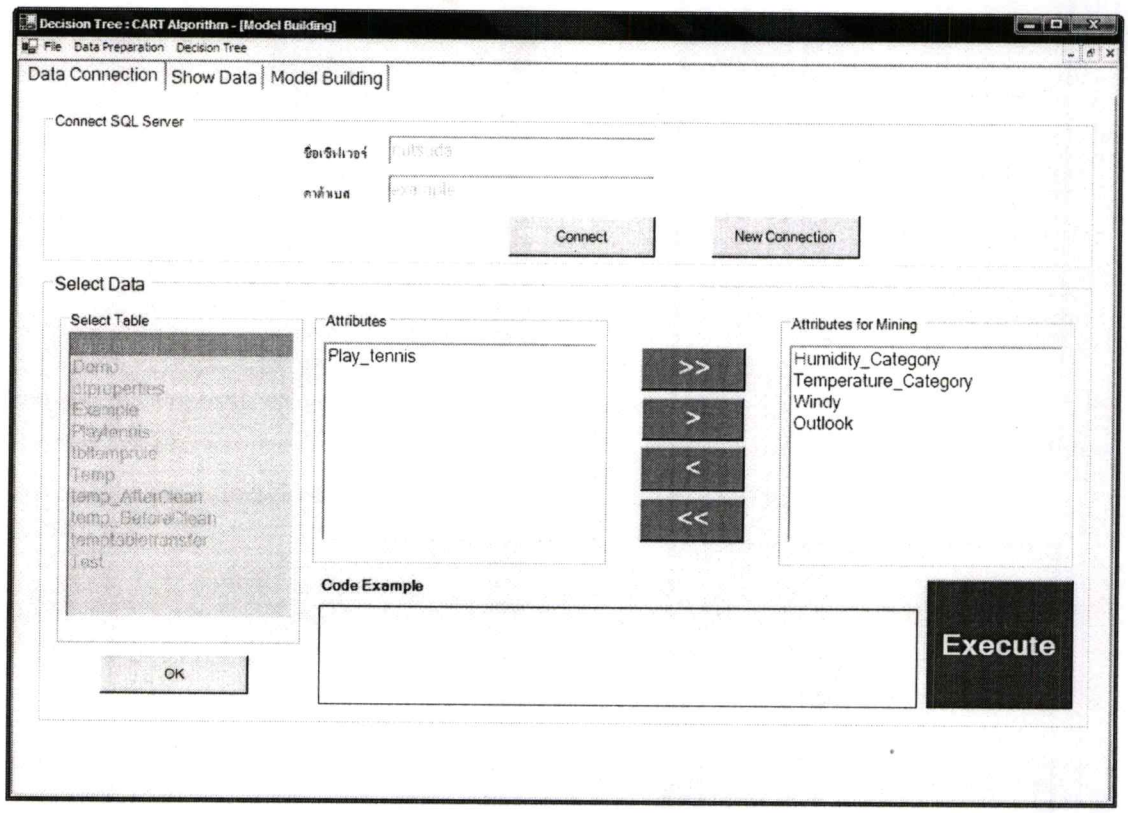
4.1 การเตรียมข้อมูล (Data Preparation)

เมื่อทำการเปิดโปรแกรม Decision Tree: CART Algorithm จะมีเมนูหลักให้เลือกคือ เมนู File, Data Preparation และ Decision Tree โดยในการเริ่มสร้างแบบจำลองนั้นจะต้องมีการเตรียมข้อมูลก่อนที่จะนำข้อมูลที่แก้ไขแล้วไปใช้ในการสร้างแบบจำลอง ซึ่งสามารถเตรียมข้อมูลได้จากเมนู Data Preparation โดยเมื่อเลือกจะได้หน้าจอขึ้นมาดังรูปที่ 4.1 โดยในหน้าจอนี้จะประกอบไปด้วยเมนูย่อยคือ Data Selection, Data Cleaning, Data Transformation, Exploration ซึ่งเป็นการทำงานของการเตรียมข้อมูลในแต่ละขั้นตอน



รูปที่ 4.1 หน้าจอเมนูการเตรียมข้อมูล

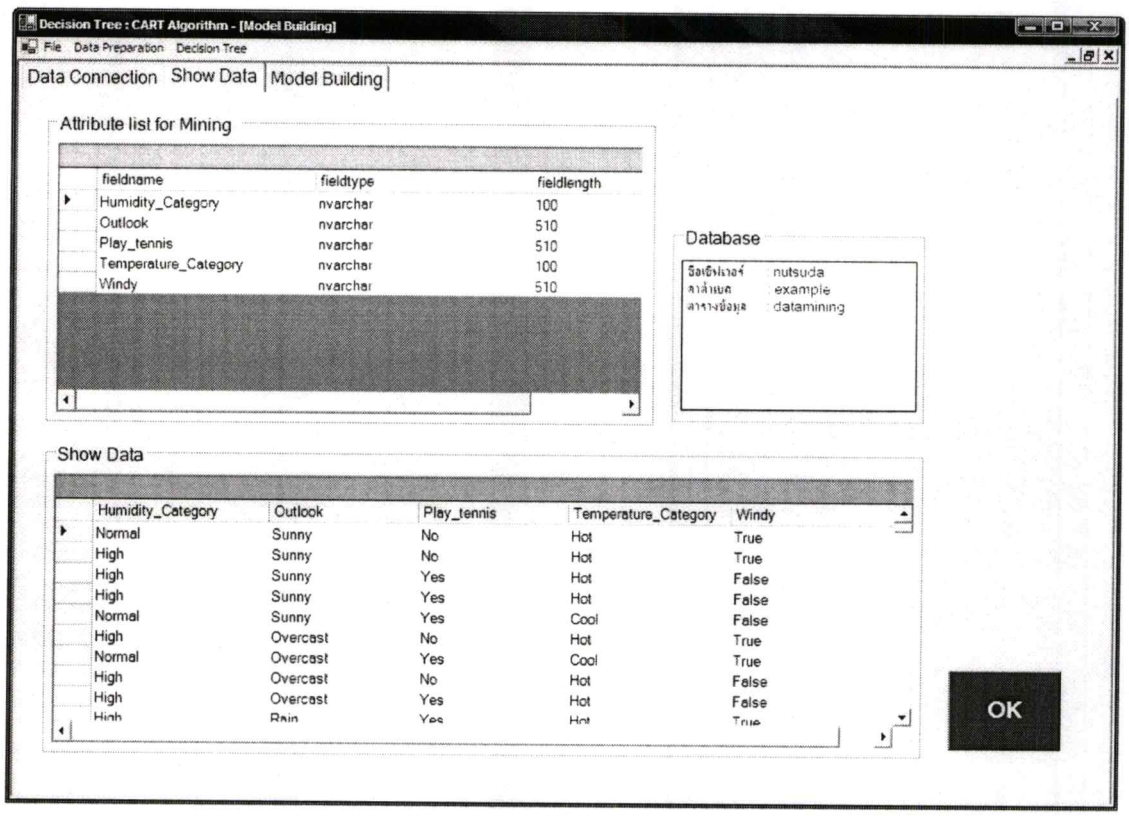
ในการเตรียมข้อมูลเพื่อนำไปใช้ในการสร้างแบบจำลองโครงสร้างต้นไม้นั้นจะเริ่มจากการเลือกข้อมูลในเมนูย่อย Data Selection ซึ่งในหน้าจอนี้จะมีขั้นตอนการทำงานเหมือนกับการติดต่อฐานข้อมูล (Data Connection) ในเมนูการสร้างแบบจำลอง โดยฐานข้อมูลที่จะทำการติดต่อด้วยคือ Microsoft SQL Server โดยผู้ใช้ต้องทำการกรอกข้อมูลชื่อเซิฟเวอร์และดาต้าเบส หลังจากนั้นกดปุ่ม Connect เพื่อเชื่อมต่อฐานข้อมูล เมื่อติดต่อฐานข้อมูลได้จะแสดงตารางของฐานข้อมูลนั้นใน Listbox Table ทางด้านซ้าย โดยผู้ใช้ทำการกดเลือกตารางที่ต้องการใช้ในการทำไมน์นิ่ง เมื่อทำการเลือกแอตทริบิวท์ที่จะเตรียมข้อมูลเสร็จแล้วกดเลือกปุ่ม Execute เพื่อทำความสะอาดข้อมูลต่อไป (Data Cleaning) โดยในขั้นตอนนี้เป็นขั้นตอนในการจัดการข้อมูลที่สมบูรณ์ เช่น การกำจัดเรคคอร์ดที่มีค่าว่าง หลังจากนั้นก็เข้าสู่ขั้นตอนการแปลงข้อมูล (Data Transformation) โดยในการแปลงข้อมูลสำหรับการสร้างแบบจำลองโครงสร้างต้นไม้นั้นจะต้องทำการแปลงข้อมูลจากข้อมูลที่เป็นตัวเลข (Numeric) เป็นกลุ่มข้อมูล (Categorical) ซึ่งจะกำหนดช่วงข้อมูลแล้วทำการจัดกลุ่มของตัวเลขให้แบ่งตามกลุ่มที่สร้างขึ้น หลังจากแปลงข้อมูลเสร็จแล้วก็จะเข้าสู่ขั้นตอนการสำรวจข้อมูล (Data Exploration) โดยจะแสดงรายละเอียดต่างๆ ของข้อมูลเพื่อเป็นการตรวจสอบข้อมูลที่ได้ทำการเตรียมไว้



รูปที่ 4.2 หน้าจอเมนูการติดต่อฐานข้อมูล

4.2 การสร้างแบบจำลองโครงสร้างต้นไม้ (Model Building)

4.2.1 เมื่อทำการเตรียมข้อมูลเพื่อใช้ในการทำไมน์นิ่งเรียบร้อยแล้วก็ถึงขั้นตอนในการสร้างแบบจำลองโดยเลือกไปที่เมนู Decision Tree > Model Building โดยจะปรากฏหน้าจอการทำงานซึ่งประกอบไปด้วยเมนูย่อยคือ Data Connection, Show Data, Model Building ซึ่งในเมนูย่อย Data Connection นั้นจะมีหน้าจอการติดต่อฐานข้อมูล ดังรูป 4.2 โดยกรอกข้อมูลชื่อเซิฟเวอร์และดาต้าเบสหลังจากนั้นกดปุ่ม Connect เมื่อติดต่อฐานข้อมูลได้จะแสดงตารางของฐานข้อมูลนั้นใน Listbox Select Table ทางด้านซ้าย โดยผู้ใช้ทำการเลือกตารางที่ถูกสร้างขึ้นจากการเตรียมข้อมูลแล้วกดเลือก OK



รูปที่ 4.3 หน้าจอเมนูแสดงข้อมูล

จากนั้นระบบจะแสดงแอตทริบิวท์ของตารางนั้นใน Listbox Attributes จากนั้นเลือกแอตทริบิวท์ที่ต้องการใช้ในการทำไมน์นิ่ง แล้วระบบก็จะแสดงแอตทริบิวท์ที่ถูกเลือกใน Listbox Attributes for Mining ทางด้านขวา ซึ่งปุ่มที่ใช้ในการทำงานคือ

ปุ่ม >> ย้ายแอตทริบิวท์จาก Listbox Attributes ไปยัง Listbox Attributes for mining ทั้งหมดทุกแอตทริบิวท์

ปุ่ม > ย้ายแอตทริบิวท์จาก Listbox Attributes ไปยัง Listbox Attributes for mining เฉพาะแอตทริบิวท์ที่เลือกไว้

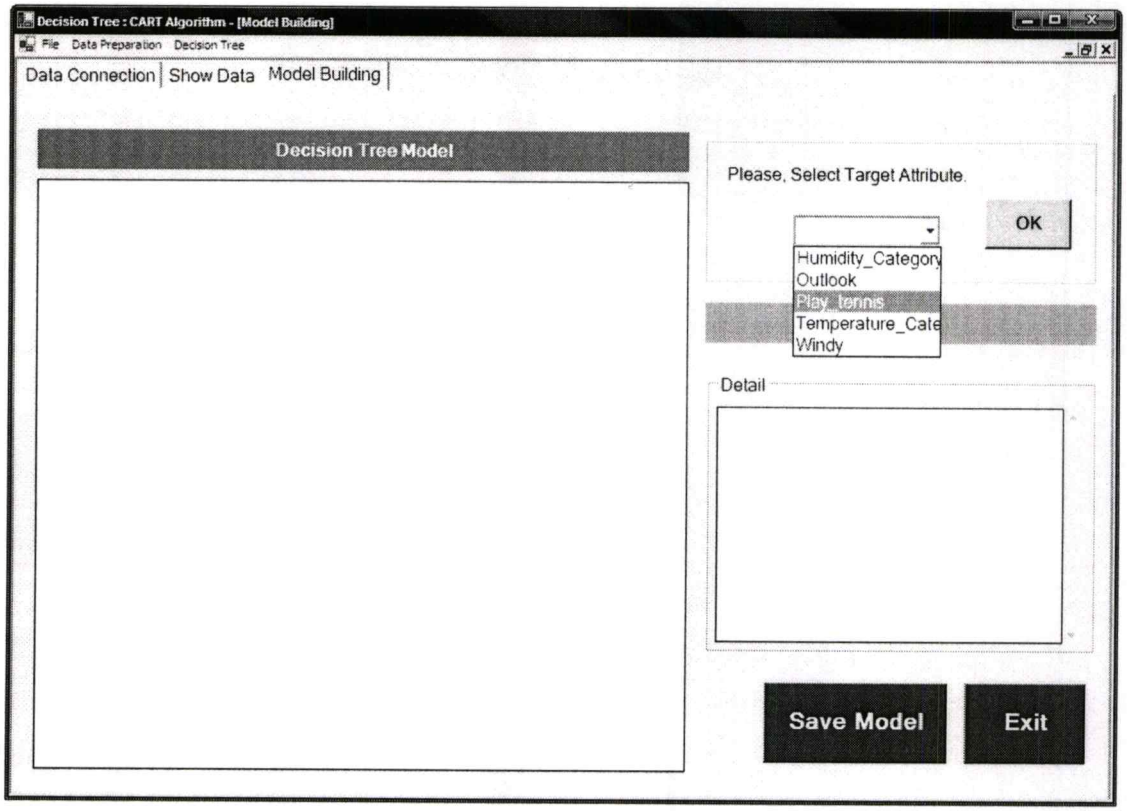
ปุ่ม < ย้ายแอตทริบิวท์จาก Listbox Attributes for mining กลับไปยัง Listbox Attributes เฉพาะแอตทริบิวท์ที่เลือกไว้

ปุ่ม << ย้ายแอตทริบิวท์จาก Listbox Attributes for mining กลับไปยัง Listbox Attributes ทั้งหมดทุกแอตทริบิวท์

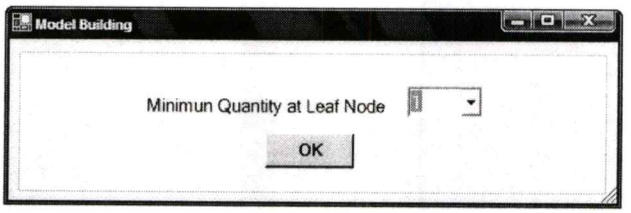
เมื่อทำการเลือกแอตทริบิวท์ที่จะใช้ในการสร้างแบบจำลองโครงสร้างต้นไม้เสร็จแล้วกดเลือกปุ่ม Execute

4.2.2 เมื่อเลือกแอตทริบิวท์ที่ใช้ในการสร้างแบบจำลองเสร็จแล้วจะปรากฏหน้าจอแสดงข้อมูล (Show Data) ดังรูปที่ 4.3 โดยจะแสดงข้อมูลที่ได้เลือกไว้โดยประกอบด้วยรายละเอียดของฐานข้อมูลที่เลือก ชื่อเซิฟเวอร์ ดาต้าเบส และชื่อตารางข้อมูลทางด้านขวามือ และข้อมูลของตารางที่ใช้ในการสร้างแบบจำลองที่แสดงชื่อฟิลด์ (Field Name) ชนิดข้อมูล (Field Type) และขนาดข้อมูล (Field Length) ทางด้านซ้ายมือ ส่วนทางด้านล่างจะแสดงรายละเอียดของแอตทริบิวท์ในตารางว่ามีข้อมูลอะไรบ้าง โดยผู้ใช้สามารถกดปุ่ม OK เพื่อเข้าสู่ขั้นตอนการสร้างแบบจำลอง

4.2.3 เมื่อเข้าสู่หน้าจอการสร้างแบบจำลอง (Model Building) ดังรูปที่ 4.4 ผู้ใช้จะต้องทำการเลือกแอตทริบิวท์เป้าหมาย (Target Attribute) ที่จะใช้ในการทำนายทางด้านขวา หลังจากเลือกนั้นกด ปุ่ม OK ระบบจะแสดงหน้าต่างให้ผู้ใช้งานเลือก Minimum Quantity at Leaf Node ดังรูปที่ 4.5 ซึ่งเป็นการกำหนดข้อมูลต่ำสุดของแต่ละโหนดที่สามารถนำแตกเป็นแบบจำลองได้ โดยถ้ามีจำนวนข้อมูล (Records) น้อยกว่าค่าที่กำหนดไว้ระบบจะทำการหยุดแตกโหนดของแบบจำลองโครงสร้างต้นไม้

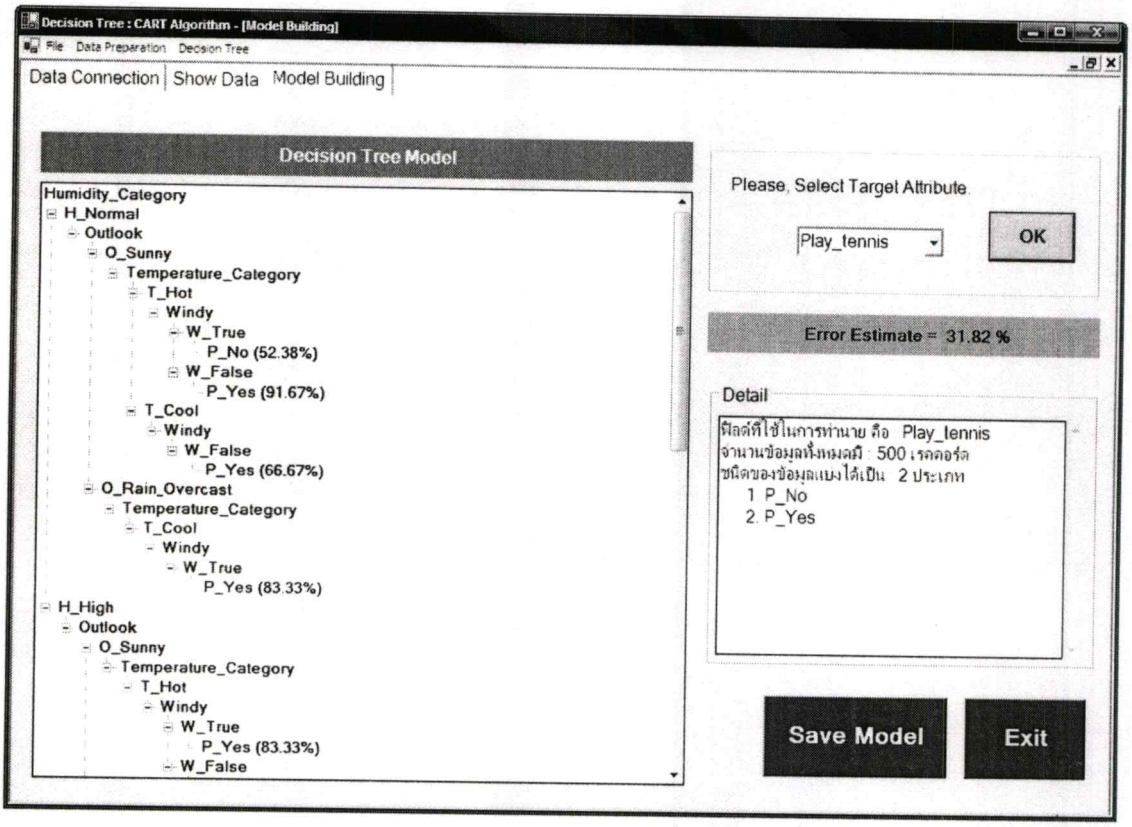


รูปที่ 4.4 หน้าจอแสดงแบบจำลอง



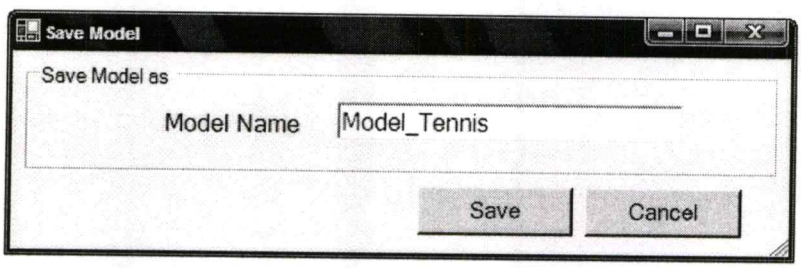
รูปที่ 4.5 หน้าจอกำหนดข้อมูลต่ำสุดของแต่ละโหนดที่สามารถแตกได้

4.2.4 หลังจากระบุค่า Minimum Quantity at Leaf Node แล้วระบบจำแสดงแบบจำลองทางด้านซ้าย รวมทั้งระบุค่าความคาดเคลื่อน (Error Estimate) ของแบบจำลองที่สร้างขึ้น และแสดงรายละเอียดข้อมูลในการสร้างแบบจำลองด้วย ดังรูปที่ 4.6 โดยรายละเอียดนั้นประกอบด้วยแอตทริบิวท์ที่ใช้ในการทำนาย จำนวนข้อมูล (Records) ที่ใช้ในการสร้างแบบจำลอง ประเภทของการทำนายแบ่งออกเป็นกี่ประเภท อะไรบ้าง รวมทั้งแสดงชื่อ ประเภท ขนาดข้อมูล ที่เป็นรายละเอียดของแอตทริบิวท์ที่ใช้ในการทำนายด้วย



รูปที่ 4.6 หน้าจอเมนูแสดงแบบจำลองโครงสร้างต้นไม้

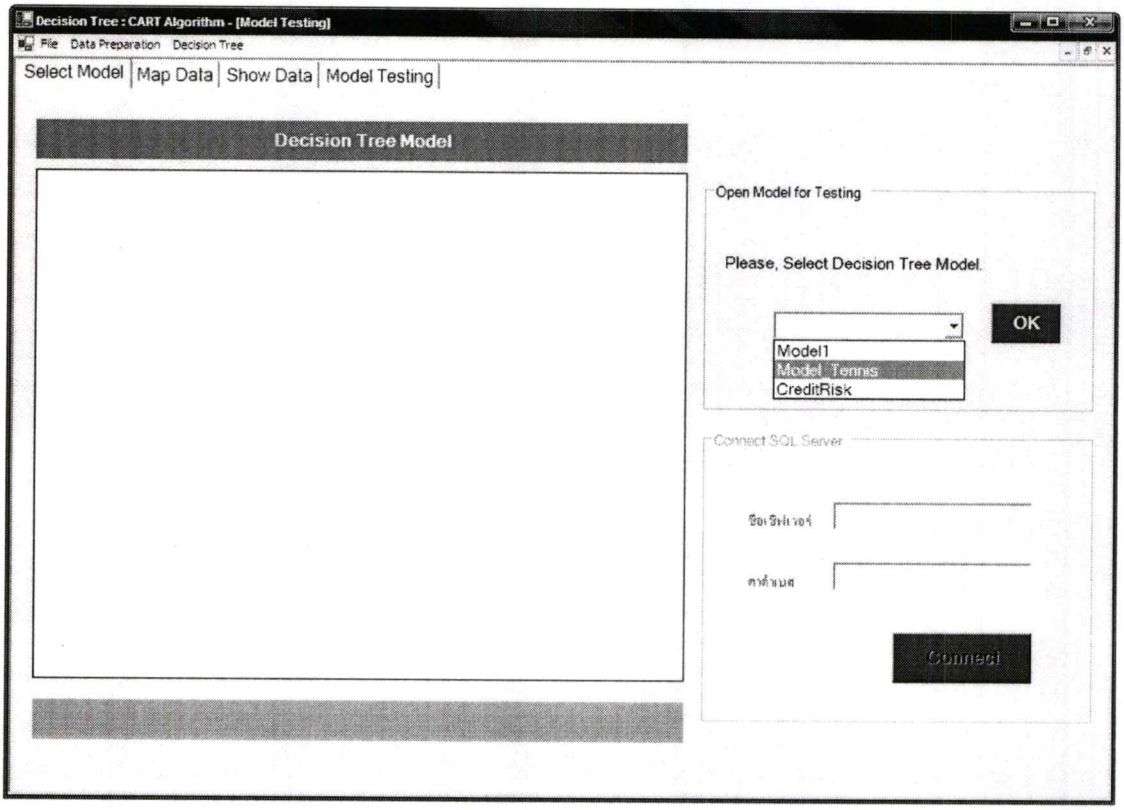
จากนั้นหากผู้ใช้ต้องการบันทึกแบบจำลองที่สร้างขึ้น สามารถกดปุ่ม Save Model โดยเมื่อเลือกแล้วระบบจะแสดงหน้าต่างให้ทำการใส่ชื่อแบบจำลองที่ต้องการบันทึก จากนั้นกด Save ดังรูปที่ 4.7 หรือหากผู้ใช้ไม่ต้องการบันทึกแบบจำลองกดปุ่ม Cancel แล้วเมื่อกลับไปยังหน้าจอแสดงแบบจำลองให้เลือกกดปุ่ม Exit เพื่อออกจากโปรแกรม



รูปที่ 4.7 หน้าจอแสดงการบันทึกแบบจำลอง

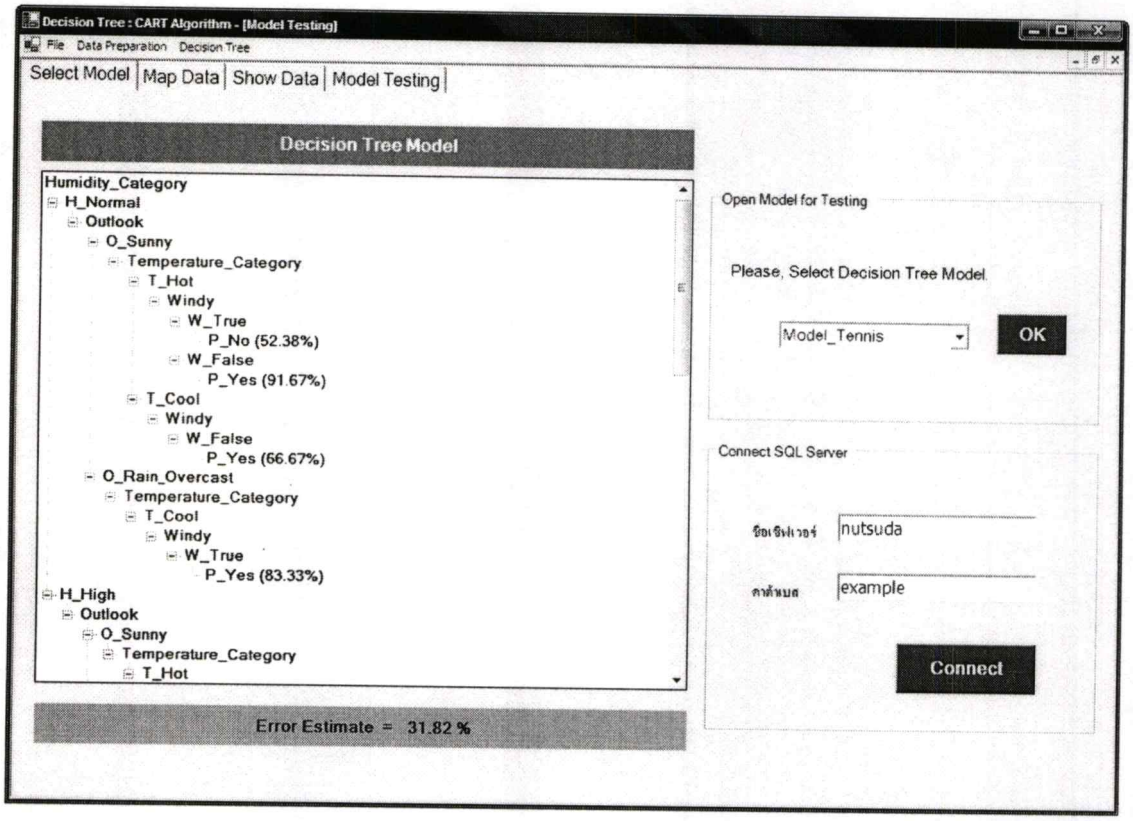
4.3 การทดสอบแบบจำลอง (Model Testing)

ผู้ใช้สามารถทดสอบแบบจำลองที่สร้างขึ้นได้โดยเลือกไปที่เมนู Decision Tree > Model Testing จากนั้นจะปรากฏหน้าจอให้ผู้ใช้เลือกเปิดแบบจำลองที่ต้องการทดสอบดังรูปที่ 4.8



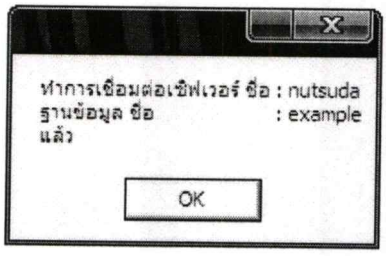
รูปที่ 4.8 หน้าจอเมนูทดสอบแบบจำลอง

4.3.1 จากหน้าจอเมนูทดสอบแบบจำลองในรูปที่ 4.8 ซึ่งแสดงหน้าจอการเลือกแบบจำลอง (Select Model) ผู้ใช้ต้องเลือกแบบจำลองที่จะทดสอบที่ได้สร้างไว้เมื่อเลือกแล้วกดปุ่ม OK ระบบจะแสดงแบบจำลองโครงสร้างต้นไม้ทางด้านซ้าย รวมทั้งแสดงค่าความคลาดเคลื่อน (Error Estimate) ทางด้านล่าง ดังรูปที่ 4.9 จากนั้นทำการติดต่อฐานข้อมูลที่จะใช้ในการทดสอบ โดยฐานข้อมูลที่จะทำการติดต่อด้วยคือ Microsoft SQL Server โดยกรอกข้อมูลชื่อเซิฟเวอร์และดาต้าเบส ทางซ้ายมือด้านล่างหลังจากนั้นกดปุ่ม Connect เพื่อเชื่อมต่อฐานข้อมูล



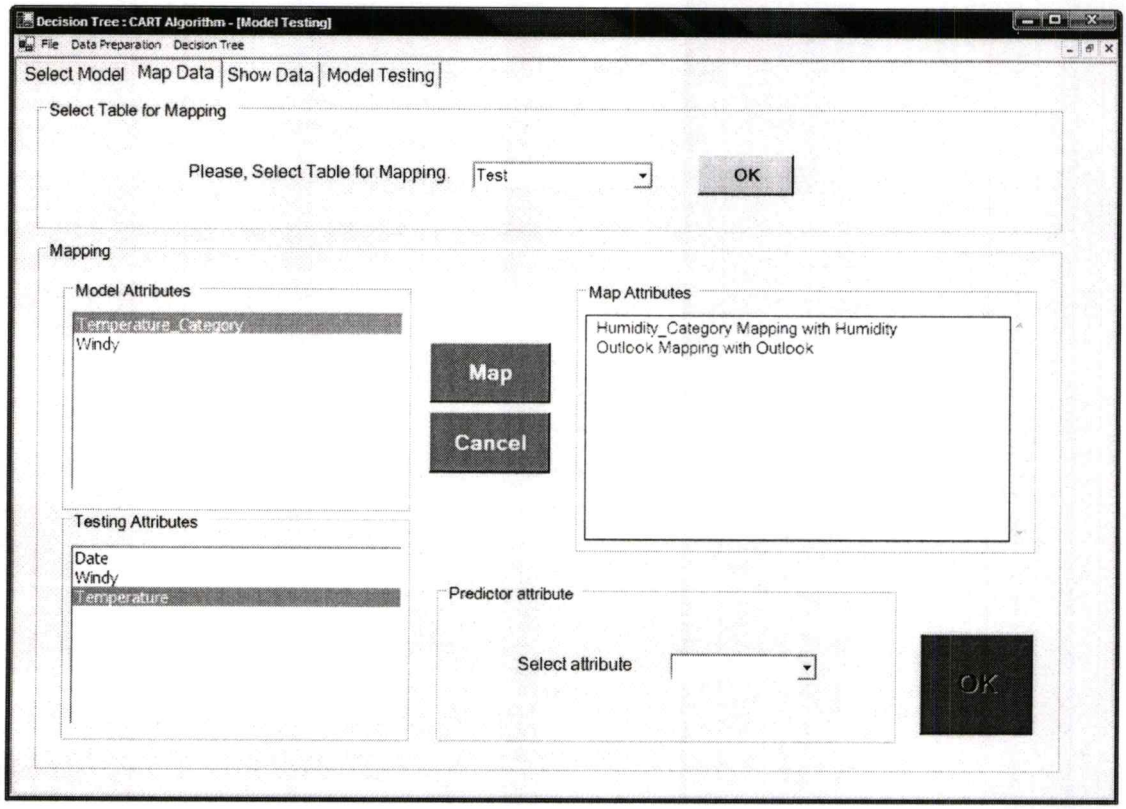
รูปที่ 4.9 หน้าจอเมนูแสดงแบบจำลองที่ใช้ทดสอบ

เมื่อติดต่อกับฐานข้อมูลที่จะใช้ในการทดสอบแบบจำลองได้แล้ว ระบบจะแสดงหน้าจอสถานะในการการติดต่อฐานข้อมูล ดังรูปที่ 4.10 หลังจานั้นกดปุ่ม OK



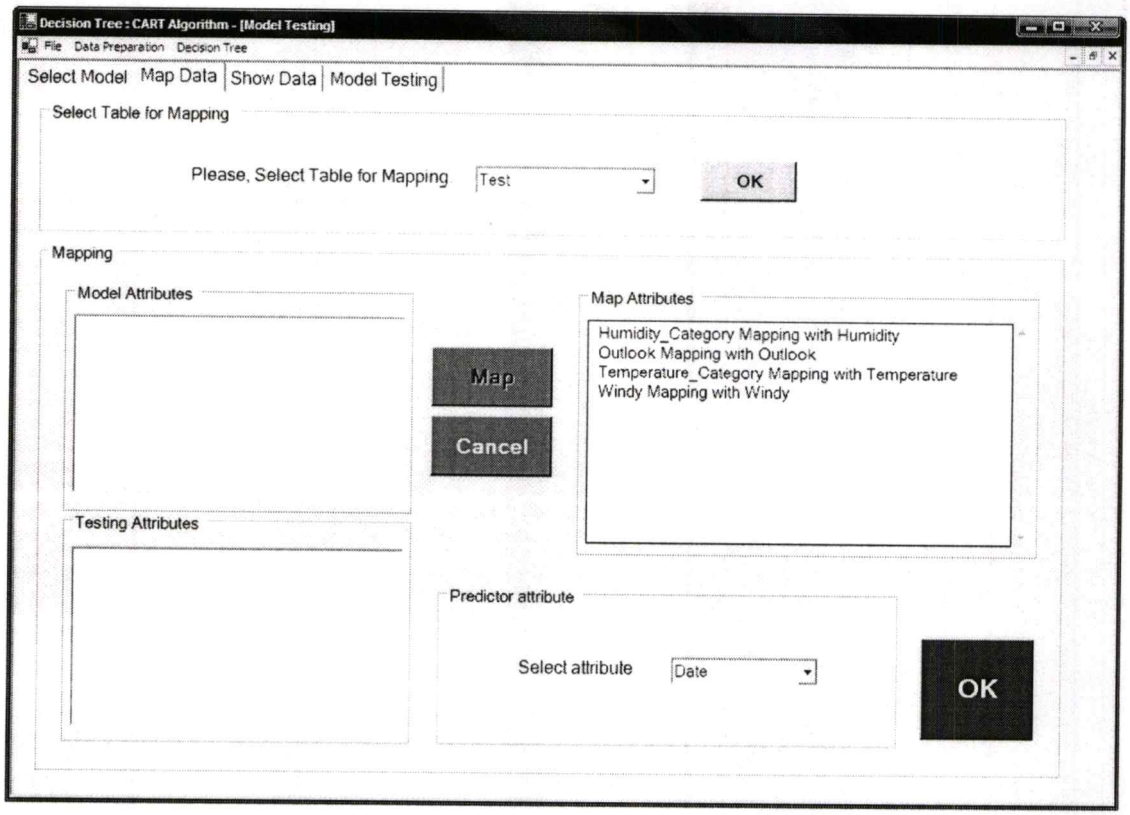
รูปที่ 4.10 หน้าจอแสดงสถานะติดต่อฐานข้อมูล

4.3.2 เมื่อติดต่อฐานข้อมูลที่จะใช้ในการทดสอบแบบจำลองได้แล้วระบบจะแสดงหน้าจอเมนูการแม็พข้อมูลที่จะใช้ในการทดสอบ (Map Data) ดังรูปที่ 4.11



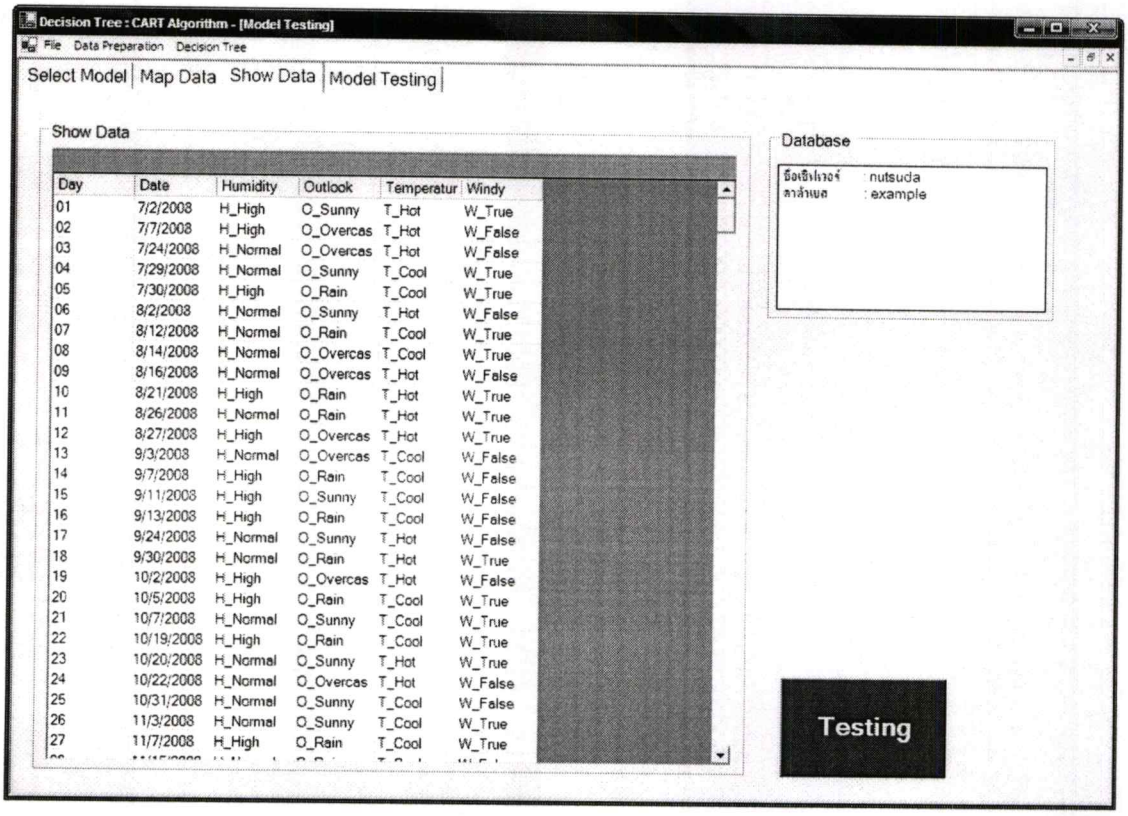
รูปที่ 4.11 หน้าจอเมนูการแม็พข้อมูลที่จะใช้ในการทดสอบ

จากรูปที่ 4.11 หน้าจอเมนูการแม็พข้อมูลที่จะใช้ในการทดสอบ ผู้ใช้จะต้องเลือกตารางที่จะใช้ทดสอบ หลังจากนั้นกดปุ่ม OK เพื่อทำการแม็พข้อมูลที่จะใช้ในการทดสอบ โดยระบบจะแสดงชื่อแอตทริบิวท์ของตารางข้อมูลที่ใช้ทดสอบใน Listbox Testing attributes ทางด้านบน และจะแสดงชื่อแอตทริบิวท์ของตารางที่ใช้ในการสร้างแบบจะลองใน Listbox Model attributes ด้านล่าง โดยให้ผู้ใช้เลือกชื่อแอตทริบิวท์ของทั้ง 2 ตาราง หลังจากนั้นกดปุ่ม Map เพื่อทำการแม็พข้อมูลทั้งสองเข้าด้วยกัน โดยระบบจะแสดงแอตทริบิวท์ที่แม็พกันใน Listbox Map attributes ทางด้านขวา โดยเมื่อทำการแม็พข้อมูลทั้ง 2 ตารางเสร็จเรียบร้อยแล้วผู้ใช้ทำการเลือกแอตทริบิวท์ที่ใช้ทำนายค่าทางด้านล่าง จากนั้นกดปุ่ม OK ดังรูปที่ 4.12



รูปที่ 4.12 หน้าจอเมนูการแม็พข้อมูลที่แสดงแอตทริบิวท์ที่ใช้ทดสอบ

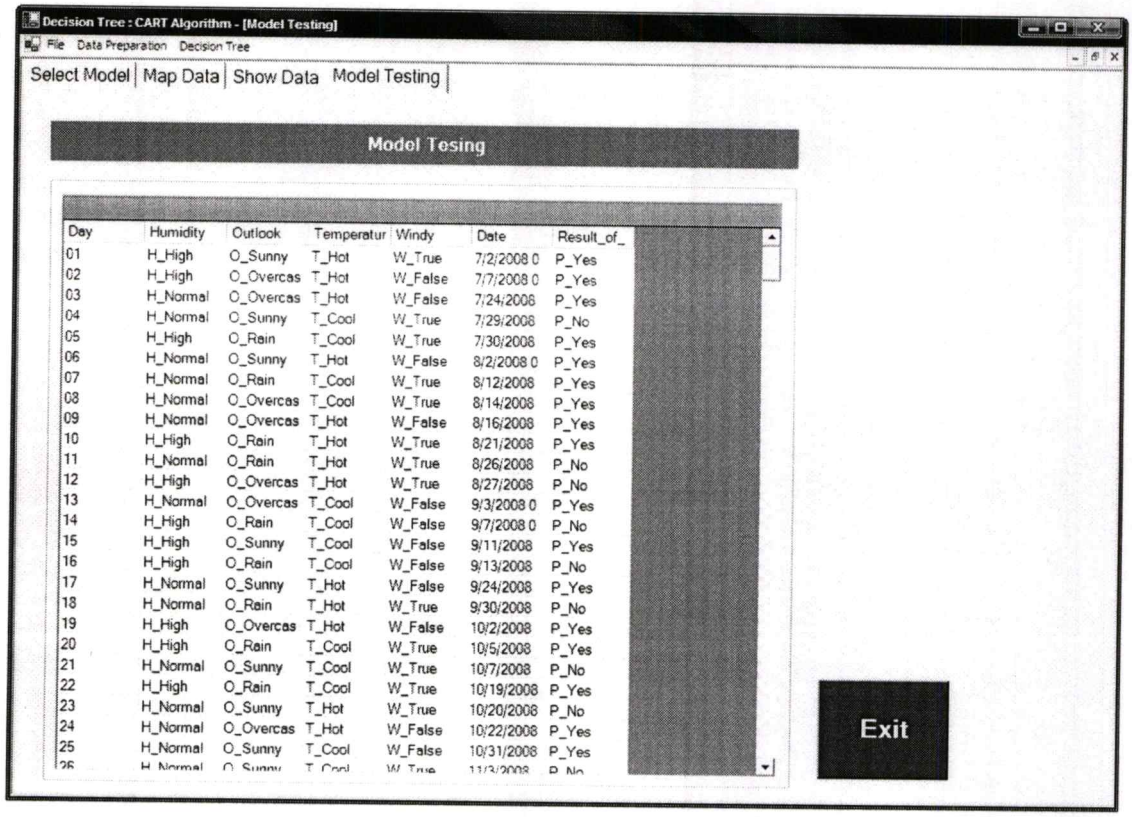
4.3.3 ระบบจะแสดงหน้าจอแสดงข้อมูลที่จะใช้ในการทดสอบ (Show Data) ดังรูปที่ 4.13 โดยจะแสดงข้อมูลของแต่ละแอตทริบิวท์ของตารางที่จะทดสอบ และแสดงรายละเอียดชื่อเซิฟเวอร์ ชื่อดาต้าเบส ชื่อตารางข้อมูลทางด้านขวา หลังจากนั้นผู้ใช้สามารถกดปุ่ม Testing ด้านล่างเพื่อทำการทดสอบแบบจะลอง



Day	Date	Humidity	Outlook	Temperatur	Windy
01	7/2/2008	H_High	O_Sunny	T_Hot	W_True
02	7/7/2008	H_High	O_Overcas	T_Hot	W_False
03	7/24/2008	H_Normal	O_Overcas	T_Hot	W_False
04	7/29/2008	H_Normal	O_Sunny	T_Cool	W_True
05	7/30/2008	H_High	O_Rain	T_Cool	W_True
06	8/2/2008	H_Normal	O_Sunny	T_Hot	W_False
07	8/12/2008	H_Normal	O_Rain	T_Cool	W_True
08	8/14/2008	H_Normal	O_Overcas	T_Cool	W_True
09	8/16/2008	H_Normal	O_Overcas	T_Hot	W_False
10	8/21/2008	H_High	O_Rain	T_Hot	W_True
11	8/26/2008	H_Normal	O_Rain	T_Hot	W_True
12	8/27/2008	H_High	O_Overcas	T_Hot	W_True
13	9/3/2008	H_Normal	O_Overcas	T_Cool	W_False
14	9/7/2008	H_High	O_Rain	T_Cool	W_False
15	9/11/2008	H_High	O_Sunny	T_Cool	W_False
16	9/13/2008	H_High	O_Rain	T_Cool	W_False
17	9/24/2008	H_Normal	O_Sunny	T_Hot	W_False
18	9/30/2008	H_Normal	O_Rain	T_Hot	W_True
19	10/2/2008	H_High	O_Overcas	T_Hot	W_False
20	10/5/2008	H_High	O_Rain	T_Cool	W_True
21	10/7/2008	H_Normal	O_Sunny	T_Cool	W_True
22	10/19/2008	H_High	O_Rain	T_Cool	W_True
23	10/20/2008	H_Normal	O_Sunny	T_Hot	W_True
24	10/22/2008	H_Normal	O_Overcas	T_Hot	W_False
25	10/31/2008	H_Normal	O_Sunny	T_Cool	W_False
26	11/3/2008	H_Normal	O_Sunny	T_Cool	W_True
27	11/7/2008	H_High	O_Rain	T_Cool	W_True

รูปที่ 4.13 หน้าจอเมนูแสดงข้อมูลที่จะใช้ในการทดสอบ

4.3.4 ระบบแสดงหน้าจอแสดงผลการทดสอบแบบจำลอง (Model Tessting) ดังรูปที่ 4.14 ซึ่งจะแสดงรายละเอียดของข้อมูลในตารางที่ใช้ในการทดสอบและแสดงผลลัพธ์ของการทดสอบแบบจำลอง โดยค่าของผลลัพธ์จะอยู่ในแอตทริบิวท์ที่ชื่อว่า Result_of_data



Day	Humidity	Outlook	Temperatur	Windy	Date	Result_of_
01	H_High	O_Sunny	T_Hot	W_True	7/2/2008 0	P_Yes
02	H_High	O_Overcas	T_Hot	W_False	7/7/2008 0	P_Yes
03	H_Normal	O_Overcas	T_Hot	W_False	7/24/2008	P_Yes
04	H_Normal	O_Sunny	T_Cool	W_True	7/29/2008	P_No
05	H_High	O_Rain	T_Cool	W_True	7/30/2008	P_Yes
06	H_Normal	O_Sunny	T_Hot	W_False	8/2/2008 0	P_Yes
07	H_Normal	O_Rain	T_Cool	W_True	8/12/2008	P_Yes
08	H_Normal	O_Overcas	T_Cool	W_True	8/14/2008	P_Yes
09	H_Normal	O_Overcas	T_Hot	W_False	8/16/2008	P_Yes
10	H_High	O_Rain	T_Hot	W_True	8/21/2008	P_Yes
11	H_Normal	O_Rain	T_Hot	W_True	8/26/2008	P_No
12	H_High	O_Overcas	T_Hot	W_True	8/27/2008	P_No
13	H_Normal	O_Overcas	T_Cool	W_False	9/3/2008 0	P_Yes
14	H_High	O_Rain	T_Cool	W_False	9/7/2008 0	P_No
15	H_High	O_Sunny	T_Cool	W_False	9/11/2008	P_Yes
16	H_High	O_Rain	T_Cool	W_False	9/13/2008	P_No
17	H_Normal	O_Sunny	T_Hot	W_False	9/24/2008	P_Yes
18	H_Normal	O_Rain	T_Hot	W_True	9/30/2008	P_No
19	H_High	O_Overcas	T_Hot	W_False	10/2/2008	P_Yes
20	H_High	O_Rain	T_Cool	W_True	10/5/2008	P_Yes
21	H_Normal	O_Sunny	T_Cool	W_True	10/7/2008	P_No
22	H_High	O_Rain	T_Cool	W_True	10/19/2008	P_Yes
23	H_Normal	O_Sunny	T_Hot	W_True	10/20/2008	P_No
24	H_Normal	O_Overcas	T_Hot	W_False	10/22/2008	P_Yes
25	H_Normal	O_Sunny	T_Cool	W_False	10/31/2008	P_Yes
26	H_Normal	O_Sunny	T_Cool	W_True	11/3/2008	P_No

รูปที่ 4.14 หน้าจอแสดงผลการทดสอบแบบจำลอง

บทที่ 5
สรุปผลการศึกษา และ ข้อเสนอแนะ

โครงการพัฒนาระบบดิซิชั่นทรี จัดทำขึ้นเพื่อให้สามารถนำข้อมูลที่มีอยู่นำมาใช้ให้มีประสิทธิภาพเพิ่มมากขึ้น และช่วยเพิ่มประโยชน์ให้กับข้อมูลที่มีอยู่ โดยการนำข้อมูลทั้งหมดที่มีอยู่มาผ่านกระบวนการทางด้านดาต้าไมน์นิ่งโดยใช้รูปแบบของโครงสร้างต้นไม้เพื่อการตัดสินใจ โดยใช้คาร์ทอัลกอริทึมในการสร้างแบบจำลองขึ้นมาเพื่อใช้ประโยชน์ต่อไป

5.1 สรุปผลการดำเนินงาน

โครงการพัฒนาระบบนี้มีวัตถุประสงค์หลักคือ เพื่อศึกษากระบวนการทางด้านดาต้าไมน์นิ่ง และประยุกต์ใช้งานเพื่อเพิ่มประสิทธิภาพของข้อมูลที่มีอยู่ให้เกิดประโยชน์สูงสุด ซึ่งเทคนิคการทำงานของดาต้าไมน์นิ่งนั้นมีอยู่หลากหลายรูปแบบที่สามารถเลือกใช้ได้ ขึ้นอยู่กับความเหมาะสมในแต่ละงาน โดยโครงการนี้ได้ศึกษาเพื่อเรียนรู้เทคนิคการสร้างแบบจำลองเพื่อการทำนาย (Predictive Modeling) และใช้วิธีการจำแนกประเภทข้อมูล (Classification) ซึ่งเป็นวิธีการทำนายว่าสิ่งที่เราสนใจจะอยู่ในกลุ่มใด โดยเสนอแบบจำลองที่อยู่ในรูปแบบโครงสร้างต้นไม้เพื่อการตัดสินใจ (Decision Tree) โดยระบบงานที่พัฒนาขึ้นนี้ได้เลือกใช้อัลกอริทึม CART ซึ่งจะนำเสนอแบบจำลองในรูปแบบ Binary Tree

ในการสร้างแบบจำลองขั้นตอนแรกก็คือการเตรียมข้อมูล โดยระบบดิซิชั่นทรีที่พัฒนาขึ้นเป็นการพัฒนาโครงการต่อจากการเตรียมข้อมูลเพื่อที่จะนำไปใช้ในการทำไมน์นิ่ง (Data Preparation) โดยข้อมูลที่นำมาใช้จะติดต่อกับ Relational Database Management Systems คือ Microsoft SQL Server ดังนั้นในการทำงานของระบบดิซิชั่นทรีที่พัฒนาขึ้นจึงสามารถนำข้อมูลที่ได้เตรียมไว้แล้วนำมาใช้งานได้เลย จึงทำให้การทำงานของระบบแบ่งออกเป็น 2 ส่วนหลักคือการสร้างแบบจำลองและการทดสอบแบบจำลอง

ส่วนแรกเป็นการสร้างแบบจำลอง โดยผู้ใช้สามารถที่จะระบุถึงฐานข้อมูลที่ต้องการที่จะติดต่อซึ่งจะเลือกจากข้อมูลจากที่ได้ถูกเตรียมไว้เรียบร้อยแล้ว โดยในสร้างแบบจำลองโครงสร้างต้นไม้ในแต่ละครั้งสามารถเลือกตารางได้เพียงตารางเดียว จากนั้นเลือกแอตตมริบิวท์ที่ต้องการใช้ในการสร้างแบบจำลอง และระบุแอตทริบิวท์ที่จะใช้ในการทำนายเพื่อให้ระบบสร้างแบบจำลองสร้างโครงสร้างต้นไม้ออกมา จากนั้นผู้ใช้สามารถที่จะทำการบันทึกข้อมูลของแบบจำลองที่ได้สร้างขึ้นเพื่อที่จะสามารถนำมาทดสอบอีกครั้งได้

ส่วนที่สองคือการทดสอบแบบจำลองโดยผู้ใช้เลือกเปิดแบบจำลองที่บันทึกไว้ขึ้นมาเพื่อใช้ทดสอบกับข้อมูลอื่น โดยในการทดสอบผู้ใช้จะต้องทำการแม็พข้อมูลที่จะใช้ในการทำนายให้ตรงกับข้อมูลที่สร้างแบบจำลอง จากนั้นระบบก็จะแสดงผลลัพธ์การทำนายออกมา

5.2 ข้อเสนอแนะ

โปรแกรมที่ได้ทำการพัฒนาขึ้นเพื่อสร้างและทดสอบแบบจำลองโครงสร้างต้นไม้ ซึ่งผู้ใช้สามารถเลือกตารางข้อมูลได้เพียงตารางเดียวต่อการสร้างโครงสร้างต้นไม้หนึ่งครั้ง ดังนั้นควรที่จะสามารถใช้ฟิลด์จากหลายๆ ตารางได้ในการสร้างโครงสร้างต้นไม้ในแต่ละครั้ง นอกจากนี้ในแต่ละครั้งที่การสร้างและทดสอบแบบจำลองโครงสร้างต้นไม้ผู้ใช้งานจะต้องทำการเตรียมข้อมูลให้สมบูรณ์ก่อนที่จะนำมาใช้ในการสร้างและทดสอบแบบจำลอง และสำหรับการทดสอบแบบจำลองโครงสร้างต้นไม้ข้อมูลที่ใช้ในการทดสอบผู้ใช้จะต้องทำการกำหนดคีย์หลักของข้อมูลเพื่อใช้ในการทดสอบด้วย

บรรณานุกรม

Breiman, L. et al. 1984. **Classificationand Regression Trees.** California : Wasdworth Int. Group.

Daniel T. Larose. 2005. **Discovering Knowledge in Data An Introduction to Data Mining.** New Jersey : Wiley Interscience.

Han, J. and Kamber, M. 2001. **Data Mining Concepts and Techniques.** San Francisco: Morgan Kaufmann Publishers.

Margaret H. Dunham. 2003. **Data Mining Introductory and Advanced Topic.** New Jersey : Prentice Hall

Michie, D. et al. 1994. **Machine Learning, Neural and Statistical Classification.** Hertfodshire : Ellis Horwood.

KDnuggets. 2008. **Data Mining Course.** [Online]. Available: http://www.kdnuggets.com/data_mining_course.

Statsoft. 2008. **Classification and Regression Trees (C&RT).** [Online]. Available : http://www.statsoft.com/textbook/stcart.html.

Wikipedia. 2008. **Decision tree.** [Online]. Available : http://en.wikipedia.org/wiki/Decision_tree.

ประวัติผู้เขียน

ชื่อผู้เขียน	นางสาวณัฐสุดา สิทธิโชค
วัน เดือน ปีเกิด	5 ตุลาคม 2525
สถานที่เกิด	กรุงเทพมหานคร
วุฒิการศึกษา	วิทยาศาสตรบัณฑิต สาขาเทคโนโลยีการจัดการ
สถานที่สำเร็จการศึกษา	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีที่สำเร็จการศึกษา	2547

บรรณานุกรม

Breiman, L. et al. 1984. **Classificationand Regression Trees.** California : Wasdworth Int. Group.

Daniel T. Larose. 2005. **Discovering Knowledge in Data An Introduction to Data Mining.** New Jersey : Wiley Interscience.

Han, J. and Kamber, M. 2001. **Data Mining Concepts and Techniques.** San Francisco: Morgan Kaufmann Publishers.

Margaret H. Dunham. 2003. **Data Mining Introductory and Advanced Topic.** New Jersey : Prentice Hall

Michie, D. et al. 1994. **Machine Learning, Neural and Statistical Classification.** Hertfodshire : Ellis Horwood.

KDnuggets. 2008. **Data Mining Course.** [Online]. Available: http://www.kdnuggets.com/data_mining_course.

Statsoft. 2008. **Classification and Regression Trees (C&RT).** [Online]. Available : http://www.statsoft.com/textbook/stcart.html.

Wikipedia. 2008. **Decision tree.** [Online]. Available : http://en.wikipedia.org/wiki/Decision_tree.

ประวัติผู้เขียน

ชื่อผู้เขียน	นางสาวณัฐสุดา สิทธิโชค
วัน เดือน ปีเกิด	5 ตุลาคม 2525
สถานที่เกิด	กรุงเทพมหานคร
วุฒิการศึกษา	วิทยาศาสตรบัณฑิต สาขาเทคโนโลยีการจัดการ
สถานที่สำเร็จการศึกษา	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีที่สำเร็จการศึกษา	2547