การพัฒนาโปรแกรมดาต้าไมน์นิ่งแบบ Decision Tree โดยใช้อัลกอริทึม CART

SYSTEM DEVELOPMENT OF BUILDING DECISION TREE MODEL APPLICATION USING CART ALGRITHM

โดย

นภนันท์ อมแก้ว

NAPANUN OMKAEW

อาจารย์ที่ปรึกษา

รศ. ดร. วรพจน์ กรีสุระเดช

รายงานฉบับนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน

หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 2 ปีการศึกษา 2550

SYSTEM DEVELOPMENT OF BUILDING DECISION TREE MODEL APPLICATION USING CART ALGRITHM

NAPANUN OMKAEW

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/ 2007

หัวข้อ	การพัฒนาโปรแกรมคาด้าไมน์นิ่งแบบ Decision Tree โดยใช้อัลกอริทึ่ม CART
นักศึกษา	นางสาวนภนันท์ อมแก้ว
รหัสนักศึกษา	47066231
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2550
อาจารย์ที่ปรึกษา	รศ. ดร. วรพจน์ กรีสุระเดช

บทคัดย่อ

การจัดแบ่งออกเป็นหมวดหมู่ (classification) เป็นกระบวนการสร้างแบบจำลอง (Model) สำหรับกำหนดกลุ่มของข้อมูลให้กับแต่ละแถวในฐานข้อมูล เพื่อแสดงให้เห็นความแตกต่างระหว่างกลุ่มของข้อมูล โดยในระบบที่พัฒนานี้จะนำเสนอการสร้างโมเดลของแผนภูมิต้นไม้โดยอาศัยอัลกอริทึ่ม CART ซึ่งจะเรียกว่าต้นไม้ช่วยในการตัดสินใจ (decision tree) โดยจะสามารถนำไปใช้สำหรับการพยากรณ์ซึ่งจะมีประโยชน์ในการนำไปประยุกต์ใช้ในการตัดสินใจในการดำเนินธุรกิจได้

Title	System Devlopment of Building Decesion Tree Model Using CART Algorithm
Student	Miss. Napanun Omkaew
Student ID.	47066231
Degree	Master of Science
Program	Information Science
Academic Year	2007
Advisor	Asst.Prof. Dr.Worapoj Kreesuradej

ABSTRACT

Classification is a process that uses to builds a model to classify data for each record in database ,it represents different among subset of data. This project introduces by build model of tree diagram base on classification and regression tree algorithm (CART) that called decision tree. Classification can be used to prediction,it has many benefit that used information from prediction to decision making in business.

กิตติกรรมประกาศ

ในโครงการพัฒนาระบบฉบับนี้ จะสำเร็จไม่ได้เลยหากไม่ได้รับความช่วยเหลือจากท่านทั้งหลายเหล่านี้ ซึ่งผู้เขียนใคร่ขอแสดงความระลึกถึงบุคคลสำคัญผู้อยู่เบื้องหลังดังต่อไปนี้

- คุณพ่อ และคุณแม่สำหรับทุนการศึกษาและกำลังใจที่เต็มเปี่ยมและทุกสิ่งทุกอย่าง
- นายวฐดล อมแก้ว น้องชายผู้น่ารักสำหรับคำปรึกษาทางด้านเทคโนโลยี Java
- อาจารย์วรพจน์ กรีสุระเดช อาจารย์ที่ปรึกษาโครงการ ผู้ให้คำปรึกษา ให้กำลังใจ และคำแนะนำต่าง ๆ ที่เป็นประโยชน์ยิ่งจนทำให้โครงการนี้สำเร็จลุล่วงไปได้ด้วยดี
- เพื่อนๆ IS17.2 ทุกคนที่คอยให้กำลังใจ และความช่วยเหลือทุกอย่าง

ท้ายนี้ต้องขอขอบคุณ สถาบัน คณะ และคณาจารย์ทุกท่านที่ได้ให้ความกรุณาประสิทธิประสาทวิชา ความรู้ จนสามารถพัฒนาโครงการพัฒนาระบบจนสำเร็จ

นภนันท์ อมแก้ว

สารบัญ

สารบัญ(ต่อ)

หน้า

สารบัญตาราง

สารบัญตาราง(ต่อ)

สารบัญภาพ

สารบัญภาพ(ต่อ)

บทที่ 1
บทนำ

1.1 ความเป็นมา

การขุดค้นข้อมูล หรือ การทำเหมืองข้อมูล (data mining) เป็นเทคโนโลยีใหม่ของการประยุกต์ใช้ข้อมูลที่เก็บอยู่ในฐานข้อมูลให้เกิดประโยชน์สูงสุดแก่หน่วยงานที่เป็นเจ้าของข้อมูล การประยุกต์ใช้ข้อมูลที่กล่าวถึงนี้มีได้หลายแนวทาง แต่โดยทั่วไปมักจะเป็นการสรุปภาพรวมของข้อมูลในฐานข้อมูล, การวิเคราะห์แนวโน้มการเปลี่ยนแปลงของข้อมูล หรือ การค้นหาความสัมพันธ์ที่ซ่อนอยู่ภายในกลุ่มของข้อมูล

หัวใจสำคัญของกระบวนการดาต้าไมนิ่ง คือส่วนของโปรแกรมที่ทำหน้าที่สังเคราะห์ความรู้ขึ้นมาจากข้อมูลจำนวนมากในฐานข้อมูล ส่วนสังเคราะห์ความรู้นี้เรียกว่า learning algorithm ซึ่งมีผู้เสนอและพัฒนาอัลกอริทึมส่วนนี้ขึ้นเป็นจำนวนมาก ได้แก่ อัลกอริทึมที่ใช้หลักการของการสร้างต้นไม้ตัดสินใจ (decision-tree induction algorithm) ตัวอย่างเช่น ID3, C4.5,CART เป็นต้น โครงการวิจัยนี้จึงเสนอขึ้นเพื่อศึกษาลักษณะของอัลกอริทึมสังเคราะห์ความรู้โดยจะอาศัย อัลกอริทึม CART ในการ Implement และนำเทคโนโลยี Java ที่เป็นที่นิยมใช้สำหรับนำมาใช้ในการพัฒนา Object Oriented Application มาช่วยในการพัฒนาทำให้เกิดการพัฒนาได้อย่างรวดเร็ว มีมาตรฐาน และบำรุงรักษา Application ได้ง่าย

1.2 วัตถุประสงค์ของโครงการ

การศึกษาในครั้งนี้เพื่อทำการจัดทำโปรแกรมเพื่อช่วยในการสร้างโมเดลในการทำดาต้าไมนิ่ง ซึ่งมีวัตถุประสงค์ดังต่อไปนี้

1. เพื่อนำความรู้และเทคนิคที่ศึกษาไปประยุกต์ใช้กับการวิเคราะห์ข้อมูลผ่านดาต้าไมนิ่งโมเดล
2. เพื่อเป็นการใช้ทรัพยากรที่มีอยู่อย่างเหมาะสมและมีประสิทธิภาพสูงสุด
3. เพื่อให้ได้ข้อมูลที่สามารถนำมาช่วยในการตัดสินใจได้อย่างถูกต้อง และรวดเร็ว
4. เพื่อเพิ่มความสามารถในการเรียกใช้ข้อมูลสำหรับผู้บริหาร

1.3 ขอบเขตของโครงการ

ระบบที่ทำการศึกษานี้ จะเป็นการศึกษาและพัฒนาแอพลิเคชั่นทางด้านดาต้าไมนิ่งโดยทำการเลือกเทคนิคโมเดลแบบ Decision Trees Model โดยใช้อัลกอริทึ่ม CART ในการพัฒนา โดยมีขอบเขตการศึกษาดังต่อไปนี้

1. สามารถสร้างโมเดลด้วยอัลกอริทึ่ม CART ผ่านการทำงานโปรแกรมเว็บแอพลิเคชั่น
2. สามารถสร้างโมเดลต้นไม้สำหรับการตัดสินใจจากข้อมูลที่ผ่านการเปลี่ยนรูปแบบของข้อมูลให้อยู่ในรูปแบบของข้อมูลสอนระบบเป็นที่เรียบร้อยแล้วได้
3. สามารถวิเคราะห์ข้อมูลและทำการแยกประเภทข้อมูลด้วยโมเดลที่สร้างขึ้นและเป้าหมายของการแยกประเภทข้อมูลที่ได้กำหนดไว้แล้วได้
4. ระบบจะสามารถรองรับได้เฉพาะฐานข้อมูลออราเคิลเท่านั้น

1.4 ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ที่คาดว่าจะได้รับจากการพัฒนาระบบงานครั้งนี้มีดังต่อไปนี้

1. ได้ความรู้และเทคนิคที่ศึกษาที่สามารถประยุกต์ใช้ในการวิเคราะห์ข้อมูลผ่านดาต้าไมนิ่งโมเดลโดยการใช้อัลกอริทึ่ม CART
2. เพื่อเป็นการใช้ทรัพยากรที่มีอยู่อย่างมีประสิทธิภาพ
3. เพื่อช่วยให้การทำงานสามารถทำได้เร็วขึ้น

1.5 ขั้นตอนในการพัฒนาระบบงาน

ขั้นตอนในการการพัฒนาระบบงานจะประกอบด้วยขั้นตอนดังต่อไปนี้

1. ศึกษาเทคนิคและวิธีการของดาต้าไมนิ่ง
2. คัดเลือกเทคนิคและอัลกอริทึ่มที่เหมาะสมที่จะทำการศึกษา
3. รวบรวมและจัดการข้อมูลที่จะนำมาใช้ในการพัฒนา
4. ศึกษารายละเอียดของการพัฒนาเว็บแอพพลิเคชัน
5. พัฒนาระบบงาน
6. ทดสอบการใช้งานและปรับปรุงแก้ไขระบบที่พัฒนาแล้วให้มีความถูกต้อง
7. สรุปผลการทดสอบจากการใช้งานที่เกิดขึ้น
8. จัดทำเอกสารคู่มือระบบ

1.6 รายละเอียดในบทต่าง ๆ

รายละเอียดต่างๆในการพัฒนาระบบงานมีดังต่อไปนี้

- **บทที่2** จะกล่าวถึงทฤษฎีและหลักการที่เกี่ยวข้องในการพัฒนาระบบงาน รวมถึงรายละเอียดของเครื่องมือและโปรแกรมต่างๆ ที่ใช้ในการพัฒนาระบบงาน
- **บทที่3** จะกล่าวถึงทฤษฎีที่เกี่ยวข้องในส่วนของดาต้าไมนิ่งเทคนิคต่างๆ ที่สามารถนำมาทำดาต้าไมนิ่งเกี่ยวกับการพยากรณ์ได้ รวมไปถึงรูปแบบต่างๆที่ใช้ในการทำโมเดลในการทำดาต้าไมนิ่ง
- **บทที่4** จะกล่าวถึงความหมายของ Decision Trees และอัลกอริทึ่มต่างๆที่ใช้ในการสร้าง Decision Trees
- **บทที่5** จะกล่าวถึงขั้นตอนการวิเคราะห์และออกแบบโปรแกรม
- **บทที่6** จะกล่าวถึงการสร้างและทดสอบโปรแกรม
- **บทที่7** จะกล่าวสรุปผลของการศึกษา ปัญหาที่พบและข้อเสนอแนะต่างๆ

บทที่ 2
ทฤษฎีและหลักการที่ใช้ในการพัฒนาแอพพลิเคชัน

ในบทนี้จะกล่าวถึงทฤษฎีและหลักการต่าง ๆ รวมทั้งเครื่องมือที่ใช้ในการพัฒนาระบบงาน ซึ่งการพัฒนานั้นจะนำเทคโนโลยี Java ที่เป็นที่นิยมใช้สำหรับนำมาใช้ในการพัฒนา Object Oriented Java Application มาช่วยในการพัฒนาทำให้เกิดการพัฒนาได้อย่างรวดเร็ว มีมาตรฐาน และบำรุงรักษา Application ได้ง่าย โดยทฤษฎีที่เกี่ยวข้อง รายละเอียดของโปรแกรมและเครื่องมือต่าง ๆ ที่ใช้ในการพัฒนาแอพพลิเคชันมีดังต่อไปนี้

2.1 Java

Java เป็นภาษาสำหรับการโปรแกรมภาษาหนึ่งที่กำลังได้รับความนิยม ภาษาจาวาถูกพัฒนาขึ้นโดยบริษัท ซันไมโครซิสเต็มส์ (Sun Microsystems Inc.) เป็นภาษาสำหรับเขียนโปรแกรมภาษาหนึ่ง มีลักษณะสนับสนุนการเขียนโปรแกรมเชิงวัตถุ (OOP : Object-Oriented Programming) ที่ชัดเจน โปรแกรมต่าง ๆ ถูกสร้างภายใน class โปรแกรมเหล่านั้นถูกเรียกว่า method หรือ behavior โดยปกติจะเรียกแต่ละ class ว่า object โดยแต่ละ object มีพฤติกรรมมากมาย โปรแกรมที่สมบูรณ์จะเกิดจากหลาย object หรือหลาย class มารวมกัน โดยแต่ละ class จะมี method หรือ behavior แตกต่างกันไป

ภาษาจาวา เป็นภาษา Object-Oriented Programming สมบูรณ์แบบ ที่มีคำสั่งพื้นฐานคล้ายกับภาษา C++ ดังนั้นเราต้องมาทำความเข้าใจถึงหลักการเบื้องต้นของภาษาในแบบ OOP กันเสียก่อนหลักการของการพัฒนาซอฟท์แวร์ด้วย Object-Oriented Programming คือการแบ่งซอฟท์แวร์ออกเป็นส่วนๆ เรียกว่า class โดยการนิยาม class และ object ทั้งนี้เพื่อทำให้สามารถนำส่วนของซอฟท์แวร์นั้นกลับมาเรียกใช้ได้อีก ลดความซ้ำซ้อนและเวลาลงได้ การทำงานของ class จะถูกกำหนดโดยส่วนอินเตอร์เฟสของ method ส่วนการทำงานของส่วนที่เป็นโค้ด จะไม่ถูกคำนึงถึงในการออกแบบ. ภาษา OOP สนใจเฉพาะข้อมูลที่จะถูกประมวลผลมากกว่าฟังก์ชั่นที่ทำการประมวลข้อมูลนั้น(กิตติ ภักดีวัฒนะกุล และ ศิริวรรณ อัมพรดนัย. 2544.)

หลักสำคัญบางประการของ OOP

- Class and Subclass
- Encapsulate
- Inheritance
- Polymorphism
- Abstract Data Type

เหตุผลที่ OOP มีบทบาทมากขึ้น

- ง่าย และรวดเร็ว ทำให้ลดเวลาในการพัฒนาลงไปได้
- เพิ่มปริมาณงานที่ได้ และมีความน่าเชื่อถือมากกว่า
- สามารถนำ code กลับมาใช้ได้อีก (เรียกใช้ class)
- ทำต้นแบบ (Prototyping) ได้รวดเร็วกว่า
- ลดต้นทุนในการสร้าง และบำรุงรักษาซอฟท์แวร์
- การเปลี่ยนแปลงแก้ไข ไม่ทำให้เกิดผลกระทบไปยังภายนอก class

Class คือกลุ่ม (category) ของ objects ที่มีคุณสมบัติและพฤติกรรมที่เหมือนกัน โดย class จะต้องประกอบไปด้วย data, behavior และ interface Class คือต้นแบบ (prototype) หรือพิมพ์เขียว ที่กำหนดตัวแปรและวิธีการ เพื่อนำไปใช้ได้ในทุก object ของ class

Object คือ สิ่งใดๆก็ตาม ซึ่งมีคุณลักษณะ (State) บ่งบอกถึงความเป็นตัวของมันเองในขณะนั้น และสามารถแสดงพฤติกรรม (Behavior) ของตัวเองออกมาได้ เช่น รถยนต์สีน้ำเงิน : มีความหมายคือ วัตถุประเภทรถยนต์ มีคุณลักษณะของสีเป็นสีน้ำเงิน และมีพฤติกรรมที่แสดงถึงการเคลื่อนที่ และหยุดได้ หรือกล่าวได้ว่า object ก็คือข้อมูลของ class (เป็นentities ของ class) ซึ่งทุกๆอย่างจะจัดเป็น objects โดยต้องประกอบไปด้วย

- ชื่อ (Identity)
- สถานะ (State) คุณสมบัติ หรือค่าของข้อมูล ซึ่งแทนด้วย value
- พฤติกรรม (Behavior) ที่ระบุว่าสามารถทำอะไรได้บ้าง ซึ่งแทนด้วย method

Method คือ function ที่บ่งบอกพฤติกรรมของ object ว่าทำอะไรได้บ้าง กำหนดไว้ใน class โดยต้องประกอบด้วย ชื่อของ method เรียกว่า Identifier ตามด้วยเครื่องหมายวงเล็บ () โดยในวงเล็บอาจมี parameter list อยู่หรือไม่ก็ได้ เช่น

- getBalance()
- raiseSalary(float Salary, float Percent)

Constructor Method คือเมธอดที่ใช้สำหรับสร้าง instance object ของคลาสนั้นๆ โดยที่ชื่อเมธอดนี้ต้องเหมือนกับชื่อคลาส และใช้สำหรับ initialize ข้อมูลให้กับ instance variable โดยจะไม่มีการถ่ายทอดให้กับ subclass และไม่มีการ return ค่า

Message คือคำสั่งหรือข้อความที่จะให้ข้อมูลหรือตัวแปรใดทำงาน ก็คือ patameter ในภาษาอื่นที่ไม่ใช่ OOP คือใช้เพื่อนำส่งค่าข้อมูลระหว่าง object โดยใน message นั้นต้องประกอบด้วย

- Destination ก็คือชื่อของ object
- Method
- Parameters

Accessibility

- Parameters
- private : เข้าถึงได้เฉพาะภายใน class เท่านั้น ไม่รวม sub class
- protected : เข้าถึงได้เฉพาะภายใน class และ sub class ที่สืบทอดกันมา (Inherit)
- default : ถ้าไม่ระบุ จะเข้าถึงข้อมูลภายใน class และอยู่เพ็กเกจเดียวกัน

Encapsulate คือการปิดบัง หรือจำกัดการเข้าถึงข้อมูลบางอย่าง (Information hiding) ที่ไม่จำเป็นต้องให้ส่วนอื่นรับรู้ ยกตัวอย่างเช่น เราจะไม่สนใจหรือมองเห็นได้ว่า เครื่องเล่น CD จะแปลงสัญญาณดิจิตอล ออกมาเป็นเพลงได้อย่างไร เราใช้และติดต่อกับเครื่องแค่ควบคุมการทำงานผ่านแผงควบคุม เช่น เปิด-ปิด เล่น เร่งเสียง เปลี่ยนแทร็กไปข้างหน้า ย้อนกลับ เป็นต้น โดยเราต้องออกแบบควบคุมกฎเกณฑ์ต่างๆ ของซอฟท์แวร์ให้สอดคล้องกับความเป็นจริง Encapsulate เป็นคุณสมบัติของ object ซึ่งมีลักษณะดังนี้

- กำหนดขอบเขตที่ชัดเจนให้กับ object
- กำหนดอินเตอร์เฟสว่าจะติดต่อกับ object อื่นๆ อย่างไร
- ส่วนอิมพลิเมนต์ไม่สามารถเข้าถึงได้ภายนอกขอบเขตของ class

Inheritance คือการถ่อยทอดข้อมูล (ซึ่งก็คือ state และ behavior) จาก class ลำดับที่สูงกว่า (super class หรือ parent class) ไปยังลำดับที่ต่ำกว่า (subclass) โดยที่ subclass นั้นสามารถเปลี่ยนแปลง หรือแทนที่ข้อมูล (override) ที่ได้รับการถ่ายทอดมานั้นได้

Polymorphisim คือ การทำให้ message อันหนึ่งสามารถส่งให้ object แต่ละตัวใน class และ subclass ตอบสนองต่อ message อันเดียวกัน ในลักษณะที่เหมาะสมกับ class ของตัวเอง ยกตัวอย่างเช่น method print นี้สามารถส่งให้ทุก object ของ class และ subclass ที่ทำให้ object นั้นรู้จัก method print และแต่ละ object ที่ต่างกันจะตอบสนองต่อ message นี้ต่างกันออกไป ตามความสามารถในการใช้

Abstract Data Type (ADT) คือ รูปแบบชนิดของข้อมูลที่ผู้พัฒนาเป็นผู้กำหนดขึ้นมาเอง

2.2 Oracle Database Server

Oracle Database Server คือ ฐานข้อมูลเชิงสัมพันธ์(RDBMS) ตัวหนึ่งของบริษัทออราเคิล และยังเป็น RDBMS เชิงพาณิชย์ตัวแรกของโลกด้วย ออราเคิลเซิร์ฟเวอร์มีจุดเด่นที่ มีความเชื่อถือได้สูงและมีให้เลือกใช้ได้เกือบทุก Platform ตั้งแต่บนเครื่องเมนเฟรม,มินิคอมพิวเตอร์และพีซี บนระบบปฏิบัติการตั้งแต่ Windows 9x,Windows NT,Windows 2000 Server,Windows ME,UNIX, โซลาริส,Linux โดยที่ในทุกพอร์ตมีโครงสร้างกลางเหมือนกันหมด คำสั่งที่ใช้ก็เป็นแบบเดียวกัน สามารถทำงานร่วมกันได้ สามารถนำข้อมูลจากพอร์ตหนึ่งไปยังพอร์ตอื่นได้อย่างไม่มีปัญหาเหมาะแก่การทำระบบต้นแบบ (Prototype) เช่น นักพัฒนาสามารถเขียน,ทดสอบ,พัฒนาระบบบนเครื่อง

Desktop ได้ โดยไม่ต้องสนใจว่าจะนำไปใช้ที่ Platform ไหนเพราะสามารถทำงานได้บนหลาย Platform

บทที่ 3
ดาต้าไมนิ่งเกี่ยวกับการพยากรณ์

ปัจจุบันธุรกิจต่างๆ มีการแข่งขันกันสูงมากทำให้องค์กรต่างๆ จำเป็นต้องปรับตัวและมีการนำเอาเทคโนโลยีสารสนเทศเข้ามาช่วยในการดำเนินงานขององค์กรกันอย่างแพร่หลาย องค์กรเหล่านี้บางแห่งมีการจัดเก็บข้อมูลในการดำเนินงานของตน บางแห่งมีการจัดเก็บข้อมูลสำหรับให้บริการแก่องค์กรอื่น ปัญหาขององค์กรต่าง ๆ ก็คือ ทำอย่างไรจึงจะประมวลผลข้อมูลที่มีอยู่ให้ได้สารสนเทศที่สามารถนำไปใช้ในรูปแบบที่เหมาะสมและมีคุณค่าต่อการนำไปใช้สนับสนุนในการบริหาร การวางแผน และการตัดสินใจในการดำเนินงาน ขององค์กรต่างๆ ได้ ซึ่ง Data Mining ก็เป็นเทคโนโลยีสารสนเทศที่สามารถตอบสนองต่อความต้องการเหล่านี้ได้ โดย Data Mining เป็นสารสนเทศที่ช่วยในการประมวลผลและวิเคราะห์ข้อมูลเพื่อให้ได้ข้อมูลที่มีประโยชน์ และสามารถนำข้อมูลเหล่านั้นมาใช้ประกอบในการตัดสินใจในการดำเนินธุรกิจขององค์กรต่างๆในอนาคตได้

3.1 Data Mining และการพยากรณ์ด้วย Data Mining คืออะไร

Data Mining คือการขุดค้นข้อมูล หรือ การทำเหมืองข้อมูล (Data Mining) เป็นเทคโนโลยีใหม่ของการประยุกต์ใช้ข้อมูลที่เก็บอยู่ในฐานข้อมูลให้เกิดประโยชน์สูงสุดแก่หน่วยงานที่เป็นเจ้าของข้อมูล การประยุกต์ใช้ข้อมูลที่กล่าวถึงนี้มีได้หลายแนวทาง แต่โดยทั่วไปมักจะเป็นการสรุปภาพรวมของข้อมูลในฐานข้อมูล, การวิเคราะห์แนวโน้มการเปลี่ยนแปลงของข้อมูล หรือ การค้นหาความสัมพันธ์ที่ซ่อนอยู่ภายในกลุ่มของข้อมูล(Cabena et al. 1998)

การทำ Data Mining เพื่อการพยากรณ์เป็นการนำความรู้ที่เรียนรู้มาจากข้อมูลที่มีอยู่เพื่อประโยชน์ในการพยากรณ์ข้อมูลใหม่ที่จะเกิดขึ้นในอนาคต ในการทำ Data Mining จะทำการเรียนรู้จากข้อมูลในฐานข้อมูลที่มีปริมาณมากๆ และค้นหาโมเดลที่สามารถใช้อธิบายลักษณะของข้อมูลเหล่านั้น จากโมเดลที่ได้นี้สามารถนำไปใช้ในการพยากรณ์ข้อมูลใหม่ๆ ที่จะเกิดขึ้นในอนาคตได้

3.2 กระบวนการที่ใช้ในการทำ Data Mining

โดยทั่วไปแล้ว หากพูดถึงการทำ Data Mining แล้วส่วนใหญ่จะให้ความสำคัญกับการทำ Mining และการค้นหารูปแบบที่ซ่อนอยู่ของกลุ่มข้อมูล แต่ที่จริงแล้วการทำ Mining ข้อมูลเป็นเพียงกระบวนการหนึ่งที่ใช้ในการทำ Data Mining เท่านั้น

กระบวนการในการทำ Data Mining มีอยู่หลายขั้นตอน ได้แก่ การกำหนดจุดมุ่งหมายของธุรกิจ, การเตรียมข้อมูล , การ Mining ข้อมูล , การวิเคราะห์ผลลัพธ์และการนำความรู้ที่ได้ไปประยุกต์ใช้ ซึ่งแม้ว่าการทำงานจะมีการทำงานต่อเนื่องเป็นขั้นตอน แต่บางขั้นตอนก็สามารถทำงานซ้ำๆ หรือมีการวนกลับไปยังขั้นตอนที่ได้ผ่านมาแล้วและในกระบวนการแต่ละขั้นตอนก็จะใช้เวลาและความพยายามในการทำงานไม่เท่ากัน(Cabena et al. 1998)

3.2.1 การกำหนดจุดมุ่งหมายของธุรกิจ (Business Objective Determination)

ขั้นตอนนี้จะเป็นการเข้าใจปัญหาทางธุรกิจและสามารถระบุถึงความต้องการหรือเป้าหมายขององค์กรได้ และเมื่อรู้ความต้องการและเป้าหมายขององค์กร ก็จะสามารถระบุถึงเป้าหมายของการ Mining ข้อมูลได้ ซึ่งจะเป็นแนวทางในการระบุถึง Algorithms ของ Data Mining และฐานข้อมูลที่สัมพันธ์กับจุดมุ่งหมายของธุรกิจได้

3.2.2 การเตรียมข้อมูล (Data Preparation)

ขั้นตอนนี้จะเป็นขั้นตอนการแปลงข้อมูลดิบหรือข้อมูลที่จัดเก็บอยู่ในที่เก็บข้อมูลต่างๆ เช่น Relational Database , Data warehouse เป็นต้น ให้อยู่ในรูปแบบมาตรฐาน (Standard Form) เพื่อนำไปทำ Data Mining ต่อไป

3.2.2.1 การเลือกข้อมูล (Data Selection)

มีจุดมุ่งหมายเพื่อระบุถึงข้อมูลที่มีอยู่ และทำการเลือกเฉพาะข้อมูลที่ต้องการ โดยจะต้องรู้ความหมายของข้อมูล ชนิดของข้อมูล ค่าที่เป็นไปได้ รูปแบบของข้อมูล และลักษณะอื่นๆ ของข้อมูลนั้น ซึ่งสามารถแบ่งชนิดการแบ่งข้อมูลออกได้เป็น 2 ลักษณะคือ

1) แบ่งตามประเภท (Catagorical) ค่าที่เป็นไปได้มีขอบเขตที่แน่นอน และมีความแตกต่างในชนิดนั้นๆ แบ่งออกได้ 2 ประเภท คือ
   a) Norminal ข้อมูลจะบ่งบอกถึงชนิดของ Object ที่อ้างอิงถึง แต่จะไม่มีลำดับของค่าที่เป็นไปได้ เช่น สถานการณ์แต่งงาน (โสด , สมรส , หย่า , ไม่ระบุ) เป็นต้น
   b) Ordinal มีลำดับของค่าที่เป็นไปได้ เช่น สถานะเครดิตของลูกค้า (ดี , ปานกลาง , แย่) เป็นต้น
2) แบ่งตามปริมาณ (Quantitative) จะมีความแตกต่างระหว่างค่าที่เป็นไปได้ แบ่งออกได้ 2 ประเภท คือ
   a) Continuous ค่าที่เก็บเป็นเลขจำนวนจริง (Real Number) เช่น ค่าเฉลี่ยต่างๆ เป็นต้น
   b) Discrete ค่าที่เก็บเป็นเลขจำนวนเต็ม (Integer) เช่น จำนวนของคนหรือสิ่งของ เป็นต้น

ข้อมูลที่ถูกเลือกเพื่อนำไปทำ Data Mining นี้ จะถูกเรียกว่า Active variable เนื่องจากข้อมูลเหล่านี้จะถูกนำไปใช้ในการแบ่งประเภท , ใช้ในการพยากรณ์ และนำไปใช้ในวิธีการอื่นๆ ซึ่งสิ่งที่ต้องคำนึงถึงอย่างมากในการเลือกข้อมูลเหล่านี้คือ ความทันสมัยของข้อมูล เนื่องจากข้อมูลอาจมีการเปลี่ยนแปลงได้เมื่อเวลาผ่านไป

3.2.2.2 การประมวลผลข้อมูลเบื้องต้น (Data Preprocessing)

ข้อมูลในโลกความเป็นจริง มักจะเป็นข้อมูลที่สกปรก , ไม่สมบูรณ์ และมีความขัดแย้งของข้อมูล เทคนิคการประมวลผลเบื้องต้นสามารถเพิ่มคุณภาพของข้อมูล ซึ่งจะช่วยเพิ่มความถูกต้องแม่นยำและประสิทธิภาพของกระบวนการทำ Mining การประมวลผลข้อมูลเบื้องต้นเป็นขั้นตอนที่สำคัญในกระบวนการค้นพบความรู้ , ป้องกันและแก้ไขข้อผิดปกติและสามารถลดขนาดข้อมูลซึ่งจะเป็นการวิเคราะห์ที่ให้ผลคุ้มค่าสำหรับการนำไปใช้ในการตัดสินใจ

3.2.2.2.1 การทำความสะอาดข้อมูล (Data Cleaning)

กระบวนการ Data Cleaning จะเป็นกระบวนการขจัดข้อมูลที่ไม่สมบูรณ์ , ข้อมูลผิดปกติซึ่งมีดังต่อไปนี้

- Missing Value ข้อมูลที่ขาดหายไป กระบวนการ Data cleaning จะทำการเติมหรือคาดคะเนข้อมูลที่ขาดหายไป
- Noisy Data ข้อมูลผิดพลาดที่อาจเกิดขึ้นหรือเป็นข้อมูลที่มีลักษณะแตกต่างจากค่าของข้อมูลที่ได้คาดการณ์ไว้
- Inconsistent Data ข้อมูลที่มีความซ้ำซ้อนหรือไม่สอดคล้องกัน

3.2.2.2.2 การรวมข้อมูลและการแปลงข้อมูล (Data Integration and Transformation)

การทำ Data Mining บ่อยครั้งต้องการการรวมกันของข้อมูลจากหลายๆ ที่เก็บ (Data stores) และบางครั้งข้อมูลอาจต้องทำการแปลงไปอยู่ในรูปแบบที่เหมาะสมสำหรับการทำ Mining

1) Data Integration ในกระบวนการนี้จะเป็นงานที่ทำการรวบรวมข้อมูลจากหลายๆ แหล่งรวมเป็นที่เก็บเดียวกัน เช่น Data warehouse ซึ่งอาจจะเป็นการรวมของข้อมูลจากหลาย Database , Data cubes หรือ Flat File เป็นต้น
2) Data Transformation ในกระบวนการนี้จะเป็นการแปลงหรือรวบรวมข้อมูลให้อยู่ในรูปแบบที่เหมาะสมสำหรับการ Mining

3.2.2.2.3 การลดขนาดข้อมูล (Data Reduction)

ในกระบวนการนี้จะเป็นเทคนิคที่สามารถใช้ลดขนาดแสดงผลของเซตข้อมูลให้มีขนาดเล็กลงกว่าเก่ามาก แต่ยังคงรักษาการคงสภาพของข้อมูลต้นฉบับ ซึ่งจะช่วยประหยัดเวลาในกระบวนการ Mining และช่วยกลั่นกรองข้อมูลสำหรับวิธีการพยากรณ์ (Prediction Method)

3.2.3 การ Mining ข้อมูล (Data Mining)

จะเป็นขั้นตอนของกระบวนการในการทำ Data Mining จริงๆ วัตถุประสงค์เพื่อประยุกต์ใช้ Algorithm ของการทำ Data Mining ที่เลือกกับข้อมูลที่ผ่านการประมวลผลเบื้องต้น ซึ่งการทำงานในขั้นตอนนี้จะแยกกันไม่ชัดเจนกับขั้นตอนการวิเคราะห์ผลลัพธ์ เนื่องจากการทำงานทั้ง 2 ขั้นตอนนี้มีความเชื่อมโยงกัน และบางครั้งอาจมีการทำซ้ำของ 2 ขั้นตอนระหว่างกระบวนการทำ Data Mining แต่จริงๆ การทำซ้ำบ่อยครั้งก็ต้องการทำย้อนกลับไปในขั้นตอนของการทำ Data Preparation แต่ขั้นตอนในการทำ Data Mining ขั้นตอนนี้จริงๆ จะหมายถึงการทำงานของ Algorithm เพื่อทำความเข้าใจในขอบเขตของข้อมูลที่วิเคราะห์ ซึ่งสิ่งที่เกิดขึ้นระหว่างกระบวนการนี้จะแตกต่างขึ้นกับชนิดของ Application ที่อยู่ภายใต้การพัฒนา เช่น ในกรณีของ Database Segmentation จะมีการทำงานของหนึ่งหรือสอง Algorithm ก็เพียงพอสำหรับขั้นตอนนี้และย้ายไปในขั้นตอนการวิเคราะห์ผลลัพธ์อย่างไรก็ตามถ้าการวิเคราะห์คือการพัฒนา Model ที่เกี่ยวกับการพยากรณ์ ก็จะเป็นกระบวนการที่สำคัญมาก ซึ่งในการพัฒนา Data Mining โดยทั่วไปจะเกี่ยวข้องกับการใช้หลายๆ Algorithm ซึ่งแต่ละ Algorithm ก็จะมีข้อดีและข้อเสียตามแต่ที่ใช้

3.2.4 การวิเคราะห์ผลลัพธ์ (Analysis of Results)

ขั้นตอนการวิเคราะห์ผลลัพธ์นี้เป็นขั้นตอนที่มีความสำคัญมาก การทำงานในขั้นตอนนี้จะต้องใช้ทักษะในการวิเคราะห์ข้อมูลร่วมกับทักษะในการวิเคราะห์เชิงธุรกิจ เพื่อทำการแปลความหมายและประเมินผลลัพธ์ที่ได้มาจากขั้นตอนการ Mining ข้อมูล ซึ่งการทำงานในขั้นตอนนี้จะขึ้นอยู่กับ Application ที่ใช้พัฒนา

3.2.5 การนำความรู้ที่ได้ไปประยุกต์ใช้ (Assimilation of Knowledge)

เป็นการรวบรวมความเข้าใจเชิงธุรกิจซึ่งได้มาจากขั้นตอนการวิเคราะห์ผลลัพธ์เพื่อนำมาประยุกต์ใช้ให้เข้ากับการดำเนินธุรกิจขององค์กร และระบบสารสนเทศ การทำงานในขั้นตอนนี้มีสิ่งที่ต้องคำนึงถึงเป็นหลักอยู่ 2 ประการ คือ

- จะต้องแสดงให้เห็นถึงแนวคิดใหม่ในเชิงธุรกิจ
  จะต้องใช้วิธีการอย่างไรในการที่จะนำความรู้ที่ได้ค้นพบใหม่นี้ไปใช้ให้เกิดประโยชน์สูงสุด

3.3 วิธีการของการทำ Data Mining และเทคนิคที่ใช้

วิธีการ และเทคนิคที่ใช้ในการทำ Data Mining สามารถแบ่งออกได้เป็น 4 ประเภทใหญ่ๆ (Cabena et al. 1998) คือ

1. Predictive Modeling เป็นการนำข้อมูลที่มีอยู่มาใช้ในการกำหนดรูปแบบที่สำคัญของข้อมูลนั้นๆ แบ่งเป็น 2 ประเภท คือ
   1.1. Classification เป็นกระบวนการสร้าง model สำหรับกำหนดกลุ่มของข้อมูลให้กับแต่ละ record ในฐานข้อมูล เพื่อแสดงให้เห็นความแตกต่างระหว่าง class หรือกลุ่มของข้อมูลได้ โดยจะใช้สำหรับพยากรณ์ข้อมูลที่ไม่ต่อเนื่อง ซึ่ง model ที่ใช้จำแนกข้อมูลออกเป็นกลุ่มตามที่ได้กำหนดไว้สามารถใช้เพื่อเข้าใจข้อมูลเก่าที่มีอยู่และพยากรณ์ข้อมูลใหม่ที่จะเกิดขึ้น เทคนิคที่ใช้เช่น Trees Induction , Neural Nets เป็นต้น
   1.2. Value Prediction หรือ Regression จะใช้สำหรับพยากรณ์ค่าข้อมูลที่ต่อเนื่องหรือเรียงลำดับโดยจะใช้ค่าข้อมูลเก่าที่มีอยู่สำหรับพยากรณ์ค่าข้อมูลอื่นที่จะเกิดขึ้น เทคนิคที่นิยมใช้สำหรับ Value Prediction ก็คือเทคนิคทางสถิติ เช่น linear regression และ nonlinear regression: Neural Nets เป็นต้น

สำหรับ Model ชนิดเดียวกันบ่อยครั้งสามารถใช้ได้ทั้งวิธีแบบ Regression และ classification เช่น Tree Induction algorithm สามารถใช้สร้างทั้ง classcification trees (แยกประเภทข้อมูล response variables) และ regression trees (ทำนายค่า response variables) Neural Nets ก็เช่นเดียวกัน สามารถสร้างทั้ง classification และ regression models

2. Database Segmentation เป็นการแบ่งข้อมูลใน ฐานข้อมูลออกเป็นส่วนย่อยๆ ตามกลุ่มของ Record ที่คล้ายคลึงกัน แบ่งเป็น 2 ประเภท คือ
   2.1. Demographic Clustering จะแบ่งข้อมูลโดยการสร้างส่วนย่อยๆของ Record ที่คล้ายคลึงกันจะเรียกว่า Segment ซึ่งจะเปรียบเทียบแต่ละ Record กับทุก Segment ที่ได้ถูกสร้างจากการทำ Mining ข้อมูล และทำให้ Record กลายไปเป็น segment ของข้อมูลดังนั้น Segment ใหม่จะถูกสร้างจากกระบวนการนี้โดย Record ที่อยู่ต่าง Segment กันจะมีความแตกต่างกัน ส่วน Record ใน Segment เดียวกันจะมีความใกล้เคียงกัน เทคนิคนี้เหมาะสำหรับการเปรียบเทียบ record ที่ข้อมูลเป็นแบบลำดับชั้น หรือแบ่งแยกออกเป็นประเภทๆ (categorical data)
   2.2. Neural Clustering เป็นการแบ่งข้อมูลโดยวิธีการสร้างบน Neural Networks จะใช้วิธี Kohenen feature maps เทคนิคการวัดความแตกต่างระหว่าง Record ว่ามีความคล้ายคลึงกันหรือแตกต่างกัน จะใช้หลัก Euclidean distance และ Segment ผลลัพธ์ที่ได้ จะจัดเรียงเป็นลำดับชั้น (Hierarchy) โดย Segment ที่เหมือนกันมากที่สุดจะตั้งอยู่ใกล้กัน เทคนิคนี้เหมาะสำหรับการเปรียบเทียบ Record ที่ข้อมูลเป็นตัวเลข (Numerical data)

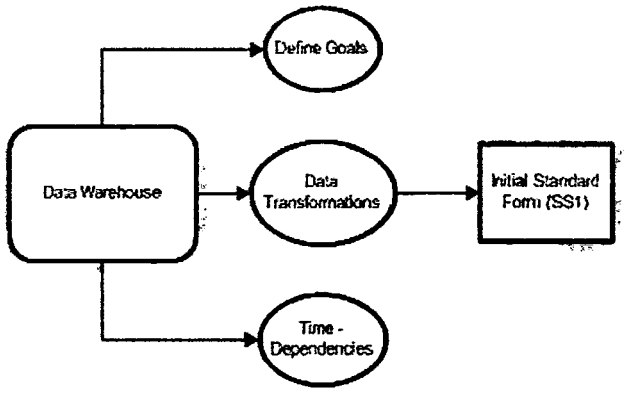
3.3 วิธีการของการทำ Data Mining และเทคนิคที่ใช้

วิธีการ และเทคนิคที่ใช้ในการทำ Data Mining สามารถแบ่งออกได้เป็น 4 ประเภทใหญ่ๆ (Cabena et al. 1998) คือ

1. Predictive Modeling เป็นการนำข้อมูลที่มีอยู่มาใช้ในการกำหนดรูปแบบที่สำคัญของข้อมูลนั้นๆ แบ่งเป็น 2 ประเภท คือ
   1.1. Classification เป็นกระบวนการสร้าง model สำหรับกำหนดกลุ่มของข้อมูลให้กับแต่ละ record ในฐานข้อมูล เพื่อแสดงให้เห็นความแตกต่างระหว่าง class หรือกลุ่มของข้อมูลได้ โดยจะใช้สำหรับพยากรณ์ข้อมูลที่ไม่ต่อเนื่อง ซึ่ง model ที่ใช้จำแนกข้อมูลออกเป็นกลุ่มตามที่ได้กำหนดไว้สามารถใช้เพื่อเข้าใจข้อมูลเก่าที่มีอยู่และพยากรณ์ข้อมูลใหม่ที่จะเกิดขึ้น เทคนิคที่ใช้เช่น Trees Induction , Neural Nets เป็นต้น
   1.2. Value Prediction หรือ Regression จะใช้สำหรับพยากรณ์ค่าข้อมูลที่ต่อเนื่องหรือเรียงลำดับโดยจะใช้ค่าข้อมูลเก่าที่มีอยู่สำหรับพยากรณ์ค่าข้อมูลอื่นที่จะเกิดขึ้น เทคนิคที่นิยมใช้สำหรับ Value Prediction ก็คือเทคนิคทางสถิติ เช่น linear regression และ nonlinear regression: Neural Nets เป็นต้น

สำหรับ Model ชนิดเดียวกันบ่อยครั้งสามารถใช้ได้ทั้งวิธีแบบ Regression และ classification เช่น Tree Induction algorithm สามารถใช้สร้างทั้ง classcification trees (แยกประเภทข้อมูล response variables) และ regression trees (ทำนายค่า response variables) Neural Nets ก็เช่นเดียวกัน สามารถสร้างทั้ง classification และ regression models

2. Database Segmentation เป็นการแบ่งข้อมูลใน ฐานข้อมูลออกเป็นส่วนย่อยๆ ตามกลุ่มของ Record ที่คล้ายคลึงกัน แบ่งเป็น 2 ประเภท คือ
   2.1. Demographic Clustering จะแบ่งข้อมูลโดยการสร้างส่วนย่อยๆของ Record ที่คล้ายคลึงกันจะเรียกว่า Segment ซึ่งจะเปรียบเทียบแต่ละ Record กับทุก Segment ที่ได้ถูกสร้างจากการทำ Mining ข้อมูล และทำให้ Record กลายไปเป็น segment ของข้อมูลดังนั้น Segment ใหม่จะถูกสร้างจากกระบวนการนี้โดย Record ที่อยู่ต่าง Segment กันจะมีความแตกต่างกัน ส่วน Record ใน Segment เดียวกันจะมีความใกล้เคียงกัน เทคนิคนี้เหมาะสำหรับการเปรียบเทียบ record ที่ข้อมูลเป็นแบบลำดับชั้น หรือแบ่งแยกออกเป็นประเภทๆ (categorical data)
   2.2. Neural Clustering เป็นการแบ่งข้อมูลโดยวิธีการสร้างบน Neural Networks จะใช้วิธี Kohenen feature maps เทคนิคการวัดความแตกต่างระหว่าง Record ว่ามีความคล้ายคลึงกันหรือแตกต่างกัน จะใช้หลัก Euclidean distance และ Segment ผลลัพธ์ที่ได้ จะจัดเรียงเป็นลำดับชั้น (Hierarchy) โดย Segment ที่เหมือนกันมากที่สุดจะตั้งอยู่ใกล้กัน เทคนิคนี้เหมาะสำหรับการเปรียบเทียบ Record ที่ข้อมูลเป็นตัวเลข (Numerical data)



ภาพที่ 3.1 การเตรียมข้อมูล

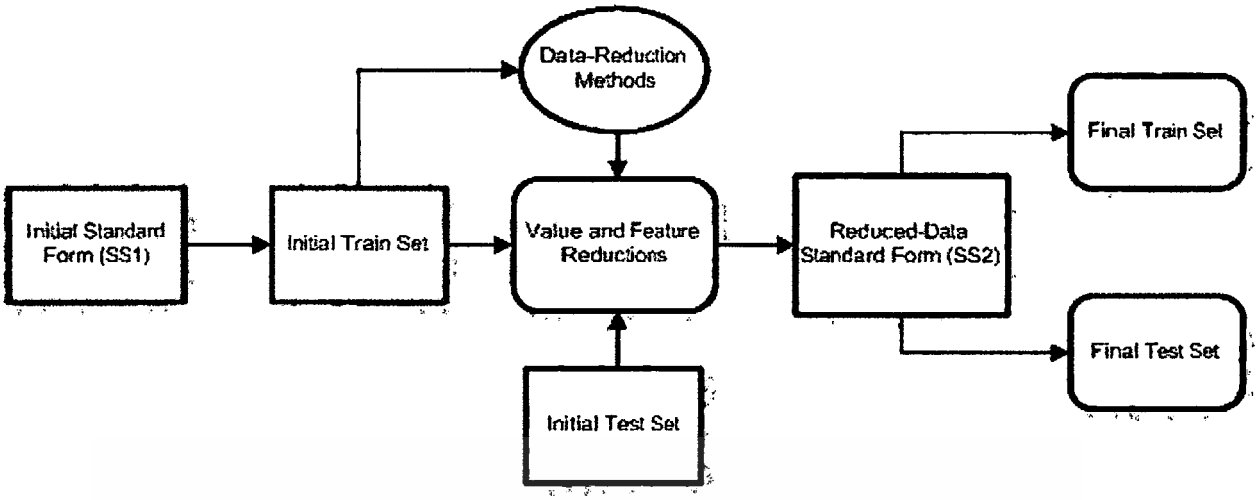
➢ ขั้นตอนที่ 2 การลดข้อมูล (Data Reduction)

การลดขนาดของข้อมูลลง เพื่อไม่ให้ข้อมูลมีขนาดใหญ่เกินไปสำหรับความจุของโปรแกรมสำหรับการพยากรณ์ หรืออาจใช้เวลาในกระบวนการประมวลผลข้อมูลและหาวิธีการในการแก้ปัญหานานมากเกินไป ซึ่งเพื่อให้ได้มาของวิธีของการแก้ปัญหาอาจทำให้ต้องมีการทำการทดลองวนซ้ำๆ ในขั้นตอนนี้จะใช้การลดขนาดข้อมูล 2 วิธีคือ

1. คุณลักษณะหรือเทคนิคการลดค่า (Feature or value - reduction techniques)
2. การแบ่งข้อมูลสำหรับการเรียนรู้และการทดสอบ (Division of data training and Testing)

ในภาพที่ 3.2 แสดงให้เห็นถึงการลดขนาดข้อมูลโดยข้อมูลถูกแบ่งเป็น 2 ส่วนคือ ข้อมูลในกลุ่มการเรียนรู้ (Train set) และข้อมูลในกลุ่มของการทดสอบ (Test set) ซึ่งข้อมูลการเรียนรู้ต้องการผ่านการลดขนาดข้อมูลเพื่อให้ได้ข้อมูลที่ถูกต้องแต่ข้อมูลการทดสอบไม่จำเป็นต้องใช้ข้อมูลที่ถูกต้องจึงไม่ต้องผ่านกระบวนการดังกล่าว จากนั้นข้อมูลทั้ง 2 กลุ่มก็เข้าสู่ขั้นตอนการลดค่าและลักษณะเพื่อให้ได้รูปแบบมาตรฐานของการลดขนาดข้อมูล ผลลัพธ์ที่ได้คือข้อมูลการเรียนรู้และข้อมูลการทดสอบที่สมบูรณ์

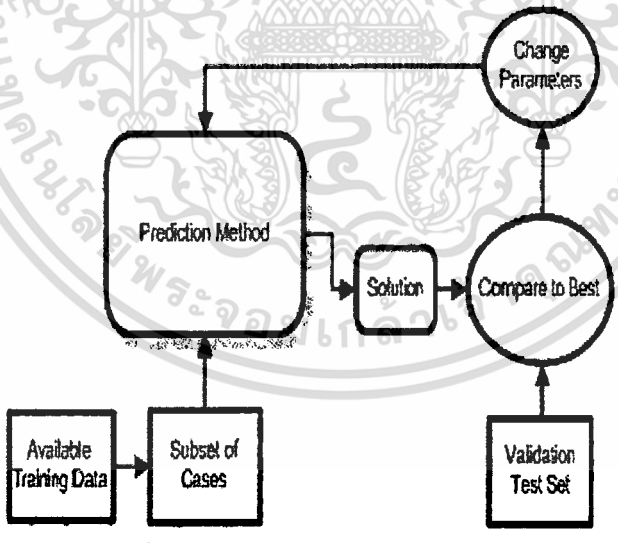
การลดขนาดข้อมูลของ Data Mining มีวัตถุประสงค์เพื่อให้เราสามารถควบคุมข้อมูลได้ง่ายยิ่งขึ้น



ภาพที่ 3.2 การลดข้อมูล

➢ **ขั้นตอนที่ 3** แบบจำลองข้อมูลและการพยากรณ์ (Data Modeling and Prediction)

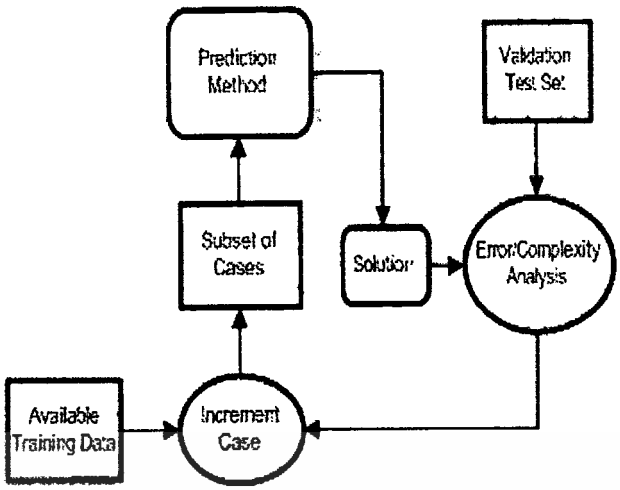
คือแบบจำลองของการเลือกข้อสรุปของข้อมูล และการอธิบายการแก้ปัญหาและวิธีการพยากรณ์เป็นการประยุกต์และปรับเปลี่ยนพารามิเตอร์เพื่อให้ได้วิธีการพยากรณ์ที่ดีที่สุด ดังแสดงไว้ในภาพที่ 3.3



ภาพที่ 3.3 แบบจำลองข้อมูลและการพยากรณ์

➢ **ขั้นตอนที่ 4** กรณีและการวิเคราะห์การแก้ปัญหา (Case and Solution Analysis)

เป็นการวิเคราะห์การแก้ปัญหาและศึกษาถึงกรณีที่เกิดข้อผิดพลาดหรือมีความซับซ้อนแล้วนำมาเพิ่มกรณีของปัญหาใหม่เพื่อทำให้การพยากรณ์นั้นถูกต้องและแม่นยำยิ่งขึ้น ดังแสดงไว้ในภาพที่ 3.4



ภาพที่ 3.4 กรณีและการวิเคราะห์การแก้ปัญหา

3.5 วิธีการที่ใช้ในการพยากรณ์ (Prediction Methods)

ในส่วนนี้จะแสดงถึงวิธีการที่ใช้ในการพยากรณ์ ซึ่งจะแบ่งเป็นกลุ่มขึ้นอยู่กับชนิดของการแก้ปัญหา ซึ่งจะมีอยู่ 3 ชนิด คือ วิธีการทางคณิตศาสตร์ (Math Solutions) , วิธีการทางระยะทาง (Distance Solutions) , วิธีการทางตรรกวิทยา (Logic Solutions) (Weiss and Indurkhya. 1998)

3.5.1 วิธีการทางคณิตศาสตร์ (Math Solution)

3.5.1.1 Linear Scoring

จะให้น้ำหนักแก่การรวมกันทางเชิงเส้นของลักษณะเฉพาะของข้อมูล ซึ่งน้ำหนักของ score ที่ได้นั้นจะเป็นผลลัพธ์ของการทำนายค่า

สมการของ Linear Scoring คือ

$$y = w_1 f_1 + w_2 f_2 + \ldots + w_m f_m + w_0 \qquad (3.1)$$

จากสมการ ค่าของลักษณะเฉพาะของข้อมูล (f_i) จะถูกคูณด้วยน้ำหนักของข้อมูล (w_i) ซึ่งจะได้คำตอบเป็น y โดยการรวมผลคูณดังกล่าวและค่าน้ำหนักที่คงที่ (w_0)

ในวิธีการนี้เราสามารถจะใช้ข้อมูลที่อยู่ในรูปแบบมาตรฐานในการประมวลผลได้อย่างง่ายดาย แต่วิธีการทางเชิงเส้นก็เหมือนกับวิธีการทางคณิตศาสตร์ ซึ่งจะอ่อนไหวได้ง่าย ทำให้เกิดปัญหา 2 ประการ คือ

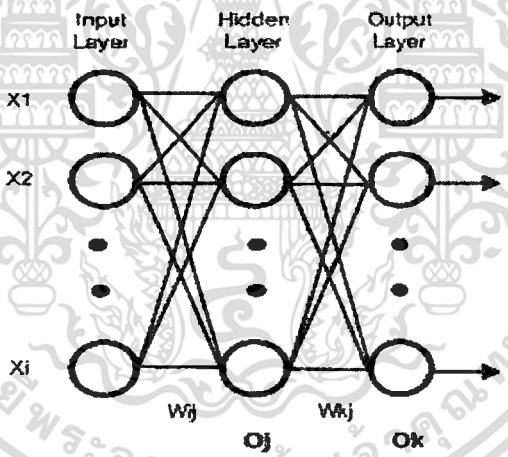
1. คุณสมบัติการแบ่งประเภท (Categorical feature)วิธีการแบบเชิงเส้นต้องการค่าที่ต่อเนื่อง ต้องแปลงคุณสมบัติการแบ่งประเภท (Categorical feature) ให้อยู่ในรูปของตัวเลขก่อนเพื่อเป็นการจำลองคุณสมบัติที่ต่อเนื่องขึ้นมา
2. ข้อมูลขาดหายไป (Missing Value) วิธีการทางคณิตศาสตร์มักจะมีปัญหาเมื่อมีข้อมูลขาดหายไป ซึ่งจะสามารถแก้ปัญหานี้ได้โดยการลดขนาดของข้อมูลบางชนิดลงเพื่อช่วยลดจำนวนของข้อมูลที่ขาดหายไป

วิธีการนี้จะใช้เวลาในการทำงานน้อยกว่าวิธีการในการพยากรณ์วิธีอื่นๆ แต่ก็จะมีปัญหาในกรณีที่วิธีการแก้ปัญหามีความซับซ้อนมาก

3.5.1.2 Nonlinear Scoring : Neural Nets

สำหรับวิธีการแก้ปัญหาแบบจำนวนไม่เป็นเชิงเส้น (Nonlinear Scoring) จะเป็นวิธีการทางคณิตศาสตร์ซึ่งไม่มีข้อจำกัดและซับซ้อน และในกลุ่มของวิธีการแก้ปัญหาแบบ Nonlinear จะเรียกว่า Neural Nets โดยวิธีการ Neural Nets สามารถนำไปประยุกต์ใช้ในการสร้าง Model สำหรับการพยากรณ์ได้ทั้ง 2 ประเภทคือ การแบ่งกลุ่มข้อมูล (Classification) และการพยากรณ์ค่า (Value prediction)

Neural Network เป็นการเลียนแบบระบบประสาทของสิ่งมีชีวิตในส่วนที่ใช้ในการพัฒนาและประมวลผล โดยการนำพื้นฐานการคำนวณของแบบจำลองทางคณิตศาสตร์มาใช้ในการสร้างให้โครงข่ายเกิดการเรียนรู้ โดยที่ในแบบจำลองของ Neural Network นั้นสามารถพิจารณาได้ในลักษณะเป็นลำดับชั้น (Layer) ดังแสดงในภาพที่ 3.5 มีส่วนประกอบที่สำคัญคือ Input Layer , Hidden Layer และ Output Layer ซึ่งในแต่ละ Layer นั้นจะมี node (บางครั้งอาจจะเรียกว่า neuron หรือ unit) ในบางครั้งส่วน Hidden Layer จะมีชั้นเดียวเรียกว่า โครงข่ายชั้นเดียว (Single Layer)



ภาพที่ 3.5 แสดงลักษณะโครงข่ายของแบบจำลอง Neural Network

และอาจจะมีหลาย Layer ได้ใน Hidden Layer เรียกว่า โครงข่ายหลายชั้น (Multilayer) ซึ่งไม่มีข้อจำกัดตายตัวในการกำหนดจำนวน Hidden Layer โดยในแต่ละ Layer นั้นจะติดต่อได้ระหว่าง node ที่อยู่คนละ Layer เท่านั้นและการติดต่อนั้นจะมีเส้นที่ใช้เชื่อมกันระหว่าง node เรียกว่า link ซึ่งจะมีค่า weight เพื่อใช้บอกระดับความสำคัญของ input โดยทั่วไปนั้น Neuron Network ประกอบด้วย

1. Processing Units ในแต่ละ Unit นั้นจะใช้แทนข้อมูลที่ได้รับเข้ามาและส่งต่อออกไป ซึ่งเราสามารถแบ่งกลุ่มของ Unit ได้ดังต่อไปนี้

Input Unit เป็นส่วนที่รับข้อมูลเข้า ข้อมูลที่ป้อนเข้าเป็นข้อมูลจากภายนอกซึ่งอาจเป็นเลขจำนวนจริงหรืออาจเป็นเลขที่ถูกปรับให้อยู่ระหว่างช่วงของ [0,1] หรือ [-1,1] ขึ้นอยู่กับรูปแบบของ Neural Network

Hidden Unit เป็นทั้งตัวรับและส่งข้อมูลภายใน network ข้อมูลที่ป้อนเข้าอาจเป็นข้อมูลจากภายนอก หรืออาจเป็นข้อมูลที่ได้จากการกระตุ้นจาก node อื่น ในกรณีที่เป็น Single Layer นั้นจะไม่มี Layer นี้

Output Unit เป็นส่วนที่ส่งข้อมูลออกนอก Network ซึ่งก็คือผลลัพธ์ที่ได้จากการทำงานของ Neural Network นั่นเอง

2. Weight (W) เป็นค่าน้ำหนักที่ให้กับ Link ที่เชื่อมระหว่าง node ที่ใช้ในการบอกว่าข้อมูลที่ป้อนเข้ามานั้นมีความสำคัญมากน้อยแค่ไหน ถ้า Weight มีค่ามากแสดงว่าข้อมูลมีความสำคัญมาก ซึ่งในขั้นตอนการเรียนรู้ (learning phase) ให้โครงข่ายเกิดการเรียนรู้นั้น Network จะทำการปรับน้ำหนักให้สามารถทำนายกลุ่มหรือค่าข้อมูลที่ถูกต้องได้
3. Activation Level เป็นค่าที่ได้จากการรวมผลคูณของค่า Weight กับ Input บวกกับค่า threshold หรือค่า Bias (θ_j)

$$net_j = \sum_{i=1}^{N} Wji\ Ii + \theta_j \qquad (3.2)$$

4. Activation Function คือฟังก์ชั่นที่ใช้ในการเปลี่ยนค่าระดับการกระตุ้นของ node นั้นๆ ซึ่ง Activate Function ที่ได้รับความนิยมนั้นได้แก่ Sigmoid Function โดยค่าที่ได้จากการผ่านฟังก์ชั่นนี้จะอยู่ระหว่าง 0 และ 1 ซึ่งสมการได้แก่

$$O_j = \frac{1}{1+e^{-net_j}} \qquad (3.3)$$

อย่างไรก็ตามถึงแม้ว่าแบบจำลอง Neural Network นี้จะมีข้อดี ตรงที่มีความยืดหยุ่นให้กับข้อมูลที่มีลักษณะผิดปกติไปจากกลุ่มได้มากและมีความสามารถในการสร้างแบบจำลองจากข้อมูลที่ไม่เคยมีการเรียนรู้มาก่อนได้เป็นอย่างดี แต่ก็ยังมีข้อด้อยคือ ต้องใช้เวลามากในการสอนให้โครงข่ายเกิดการเรียนรู้ ก่อนที่จะนำแบบจำลองไปใช้ในการพยากรณ์ค่าข้อมูลได้ และนอกจากนี้ยังไม่สามารถอธิบายวิธีการสร้างแบบจำลองของโครงข่ายได้อย่างชัดเจน

นอกจากวิธีการทางคณิศาสตร์ที่กล่าวมาข้างต้นแล้วยังมีวิธีการอื่นๆ อีกที่สามารถใช้ในการพยากรณ์

3.5.2 วิธีการทางระยะห่าง (Distance Solutions)

ในขณะที่การวิเคราะห์ข้อมูล เพื่อกรองรูปแบบออกมาซึ่งจะมีการทิ้งข้อมูลบางส่วนไป แต่สำหรับวิธีการนี้แล้วข้อมูลจะถูกเก็บได้ไว้เพื่อใช้สำหรับการ matching กับข้อมูลเก่า ในการวัดระยะห่าง (distance measures) จะเป็นการใช้ในการหาปริมาณของระยะห่างหรือความแตกต่างของชุดข้อมูลเก่าที่ถูกเก็บไว้ว่ามีความคล้ายคลึงกันมากที่สุดกับชุดข้อมูลใหม่ที่เข้ามาเพียงใด เช่น

วิธีการ k-nearest-neighbor (k-NN) ซึ่งชุดของข้อมูลจะเก็บไว้ เพื่อเปรียบเทียบกับข้อมูลใหม่ที่เข้ามา โดยเมื่อข้อมูลใหม่เข้ามาก็จะพบความใกล้เคียงและความคล้ายคลึงกันของชุดข้อมูล และส่วนที่เหมือนกันมากที่สุด (nearest neighbors) จะถูกระบุไว้ในชุดเดียวกัน

k-NN model จะสามารถเข้าใจได้ง่ายเมื่อตัวแปรในการพยากรณ์ไม่มาก และสามารถใช้สร้าง model ที่ประกอบด้วยชนิดของข้อมูลที่ไม่เป็นมาตรฐาน (non-standard data types) เช่น ข้อความตัวอักษร (Text)

3.5.3 วิธีการทางตรรกวิทยา (Logic Solutions)

3.5.3.1 Trees Induction

สำหรับ Trees Induction จะมีลักษณะเป็น Flow-chart ที่มีโครงสร้างคล้ายต้นไม้ ที่มีส่วนประกอบดังต่อไปนี้

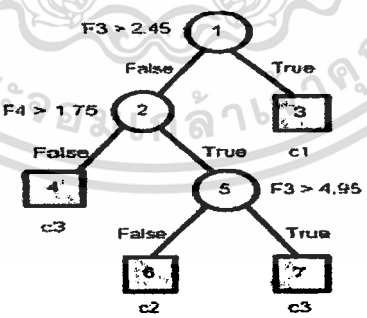
1.Internal Node (nonleaf) จะใช้แสดงการทดสอบข้อมูล

2.Branch เส้นที่แตกสาขาออกมาจาก Internal Node จะเป็นผลลัพธ์จากการทดสอบ

3.Leaf Node จะแทนกลุ่มของข้อมูล

ส่วนที่อยู่บนสุดของ Tree จะเรียกว่า Root Node โดยในการแบ่งชนิดของข้อมูลที่ไม่รู้จักมาก่อน จะนำค่าของข้อมูลนั้นไปทำการทดสอบโดยการนำไปเปรียบเทียบกับ Tree Induction เส้นทางที่ผ่านจาก Root ไปยัง Leaf Node จะเป็นชนิดของข้อมูลนั้นซึ่งได้มาจากการทำนาย

ในวิธีการนี้จะมีประสิทธิภาพสูงในเรื่องเวลาที่ใช้ในการประมวลผลข้อมูลเบื้องต้น และการจัดหาวิธีการใหม่ๆในการหาผลลัพธ์ แต่ Tree Induction ก็มีข้อเสียหลายอย่าง เช่น ในบาง Tree Induction อาจทำให้เกิดการเข้าใจผิด หรือไม่สามารถนำไปใช้กับข้อมูลบางอย่างได้ และอาจทำให้เกิดปัญหาจากการประมวลผลค่าที่มีความต่อเนื่องกัน



ภาพที่ 3.6 แสดงลักษณะ Tree Induction

3.5.3.2 Rules Induction

วิธี Trees Induction และ Rules จะเป็นการแสดงความสัมพันธ์ในเชิง Logic โดยความสัมพันธ์เชิง Logic มักจะถูกนำเสนอในรูปของกฎเกณฑ์ หรือ Rules โดยประเภทที่ง่ายที่สุดจะแสดงออกมาในรูปเงื่อนไข (Condition) หรือ ความสัมพันธ์ร่วมกัน เช่น If *condition1* then *condition2* ซึ่ง If-then เป็นวิธีการ (routine) สำหรับแสดงการทำการตัดสินใจ (decision Making)

โดยเงื่อนไขหรือข้อแม้ของกฎ (rule) จะเป็น Boolean expression ในปัญหาที่จะต้องพิสูจน์ (proposition form) ผลสรุปสุดท้ายจะเป็นการให้ค่าหรือกำหนด class ใน Rules Induction จะไม่เหมือนเส้นทางของโครงสร้างต้นไม้ (tree paths) โดยบางวิธีการจะเป็นการรวมกันของกฎหลายกฎ หรือเลือกกฎเดียว แต่ละ Rule Induction จะเป็นการรวมกัน (Conjunction (AND)) ในเทอมของ true-or-false และกฎหลายกฎจะสอดคล้องสัมพันธ์ซึ่งกันกับกฎอื่น ๆ แต่ละกฎโดย (Disjunction (OR))

$$F3 \leq 2.45 \rightarrow Class=1$$
$$F3 \leq 4.95 \wedge F4 \leq 1.7 \rightarrow Class=2$$
$$Otherwise \rightarrow Class=3 \qquad (3.4)$$

จากตัวอย่างข้างต้นจะเป็นการแสดง Rule-Induction Solution

Rules Induction สามารถค้นพบหลักการทั่วไปมากมาย ที่ซึ่งเกี่ยวข้องกับทั้งข้อมูลที่เป็นตัวเลขและไม่เป็นตัวเลขและยังสามารถรวมหลักการเงื่อนไขและความสัมพันธ์ไว้ร่วมกันได้ Rule Induction บ่อยครั้งมักพบว่ามีประสิทธิภาพมากกว่า Trees Induction และมีสามารถแสดงได้กระชับมากกว่า ข้อดีสำหรับ Rules Induction ส่วนใหญ่เมื่อเจอปัญหาที่ไม่ซับซ้อนมากจะเป็นการพยากรณ์ที่ดีที่สุด

Data Mining คือการขุดค้นข้อมูล หรือ การทำเหมืองข้อมูล (Data Mining) เป็นเทคโนโลยีใหม่ของการประยุกต์ใช้ข้อมูลที่เก็บอยู่ในฐานข้อมูลให้เกิดประโยชน์สูงสุดแก่หน่วยงานที่เป็นเจ้าของข้อมูล การประยุกต์ใช้ข้อมูลที่กล่าวถึงนี้มีได้หลายแนวทาง แต่โดยทั่วไปมักจะเป็นการสรุปภาพรวมของข้อมูลในฐานข้อมูล , การวิเคราะห์แนวโน้มการเปลี่ยนแปลงของข้อมูล หรือ การค้นหาความสัมพันธ์ที่ซ่อนอยู่ภายในกลุ่มของข้อมูล

การทำ Data Mining เกี่ยวกับการพยากรณ์ (Predictive Data Mining) เป็นการนำความรู้ที่เรียนรู้มาจากข้อมูลที่มีอยู่เพื่อประโยชน์ในการพยากรณ์ข้อมูลใหม่ที่จะเกิดขึ้นในอนาคต ในการทำ Data Mining จะทำการเรียนรู้จากข้อมูลในฐานข้อมูลที่มีปริมาณมากๆ และค้นหาโมเดลที่สามารถใช้อธิบายลักษณะของข้อมูลเหล่านั้น จากโมเดลที่ได้นี้สามารถนำไปใช้ในการพยากรณ์ข้อมูลใหม่ๆ ที่จะเกิดขึ้นในอนาคตได้

งานของการพยากรณ์จะมี 4 ขั้นตอน คือ การเตรียมข้อมูล , การลดข้อมูล , การหาแบบจำลองข้อมูลและการพยากรณ์ และกรณีและการวิเคราะห์การแก้ปัญหา ซึ่งในขั้นตอนของการหาแบบจำลองของการพยากรณ์จะมีวิธีการที่ใช้ในการพยากรณ์(Prediction Methods)ได้หลายวิธีมีดังนี้

วิธีการทางคณิตศาสตร์ , วิธีการทางระยะทาง , วิธีการทางตรรกวิทยา ซึ่งในแต่ละวิธีการก็ยังมีเทคนิคอีกหลายวิธีให้เลือกใช้ วิธีการทำงานแต่ละแบบก็จะมีข้อดีและข้อเสียแตกต่างกันไป การเลือกวิธีการให้เหมาะสมกับข้อมูลที่มีอยู่จึงเป็นสิ่งสำคัญ

บทที่ 4
การจัดแบ่งออกเป็นหมวดหมู่ด้วยแผนภาพต้นไม้ตัดสินใจ

Data mining เป็นกระบวนการที่ใช้ในการวิเคราะห์ข้อมูล (data analysis) กลั่นกรองและแยกแยะประเภทข้อมูล (data classification) ที่มีปริมาณมหาศาล เพื่อค้นหารูปแบบและความสัมพันธ์ในข้อมูลหรือข้อมูลที่มีประโยชน์ และนำข้อมูลนั้นมาใช้เป็นฐานความรู้ เพื่อนำไปใช้สนับสนุนการบริหาร การวางแผน และการตัดสินใจในการดำเนินงานขององค์กรต่างๆได้ การจัดแบ่งข้อมูลออกเป็นหมวดหมู่หรือแยกแยะประเภทข้อมูล (classification) ก็เป็นเทคนิคหนึ่งใน Data Mining ที่ใช้สำหรับสร้างแบบจำลองการพยากรณ์ (predictive modeling) โดยจะทำการสร้างแบบจำลองจากกลุ่มข้อมูลตัวอย่างที่เลือกมาจากฐานข้อมูล ที่เรียกว่าข้อมูลสอนระบบ (trainning data) และแบบจำลองนั้นสามารถพยากรณ์ผลลัพธ์ของข้อมูลที่ไม่เคยพบมาก่อน บนพื้นฐานความสัมพันธ์ของกลุ่มข้อมูลเดิม และแบบจำลองลักษณะนี้เรียกว่า Supervised learning ซึ่งสำหรับเทคนิคที่ใช้ในการ classification แบ่งได้เป็น 2 แบบ คือ Tree Induction และ Neural Induction โดยในที่นี้จะนำเสนอเทคนิคของ Tree Induction

4.1 Classification

การจัดแบ่งข้อมูลออกเป็นหมวดหมู่หรือแยกแยะประเภทข้อมูล เป็นกระบวนการสร้างแบบจำลอง (Model)สำหรับกำหนดกลุ่มของข้อมูลให้กับแต่ละ record ในฐานข้อมูลซึ่งจะถูกเรียกว่ากลุ่มของข้อมูลสอนระบบ (training data set) โดยแต่ละ record ประกอบด้วยหลายๆ แอตทริบิวต์ โดยกลุ่มของประเภทหนึ่งๆของแอตทริบิวต์ (categorical attributes) จะถูกเรียกว่า คลาส label ซึ่งจะใช้ในการระบุคลาส ที่แต่ละ record เป็นส่วนหนึ่งของคลาสนั้น เพื่อแสดงให้เห็นความแตกต่างระหว่างคลาสหรือกลุ่มของข้อมูลได้ โดยจะใช้สำหรับพยากรณ์ข้อมูลที่ไม่ต่อเนื่อง ซึ่งแบบจำลอง (model) ที่ใช้จำแนกข้อมูลออกเป็นกลุ่มตามที่ได้กำหนดไว้สามารถใช้เพื่อเข้าใจข้อมูลเก่าที่มีอยู่และพยากรณ์ข้อมูลใหม่ที่จะเกิดขึ้น ซึ่งมีอัลกอริทึมที่นิยมคือ Tree Induction และ Neural Induction

- Tree Induction

การนำเอาข้อมูลมาสร้างแบบจำลองในรูปแบบของต้นไม้ตัดสินใจ (decision tree) ซึ่งมีการทำงานแบบการเรียนรู้ขั้นสูง คือ สามารถสร้างแบบจำลองการจัดหมวดหมู่ได้จากข้อมูลตัวอย่างที่ได้กำหนดไว้ล่วงหน้าแล้ว และสามารถพยากรณ์กลุ่มของข้อมูล ที่ยังไม่เคยนำมาจัดหมวดหมู่ได้อีก ด้วยรูปแบบของต้นไม้มีส่วนประกอบดังต่อไปนี้

1.Internal Node (nonleaf) จะใช้แสดงถึงการทดสอบข้อมูลบน Attribute

2.Branch เส้นที่แตกสาขาออกมาจาก Internal Node จะเป็นผลลัพธ์จากการทดสอบ

3.Leaf Node จะแทนกลุ่มของข้อมูล class

ส่วนที่อยู่บนสุดของต้นไม้ (tree) จะเรียกว่า Root Node โดยในการแบ่งชนิดของข้อมูลที่ไม่รู้จักมาก่อน จะนำค่าของข้อมูลนั้นไปทำการทดสอบโดยการนำไปเปรียบเทียบกับ Tree Induction เส้นทางที่ผ่านจาก Root ไปยัง Leaf Node จะเป็นกลุ่มของข้อมูลนั้น (class) ซึ่งได้มาจากการทำนาย

- Neural Induction

แบบจำลองที่มีโครงสร้างที่เป็นการเลียนแบบการทำงานของระบบประสาทของสิ่งมีชีวิตในส่วนที่ใช้ในการพัฒนาและประมวลผล โดยการนำพื้นฐานการคำนวณของแบบจำลองทางคณิตศาสตร์มาใช้ในการสร้างให้โครงข่ายเกิดการเรียนรู้ โดยที่ในแบบจำลองของ neural นั้นสามารถพิจารณาได้ในลักษณะเป็นลำดับชั้น (layer) มีส่วนประกอบที่สำคัญคือ Input Layer , Hidden Layer และ Output Layer ซึ่งในแต่ละ layer นั้นจะมี node (บางครั้งอาจจะเรียกว่า neuron หรือ unit) ในบางครั้งส่วน Hidden Layer จะมีชั้นเดียวเรียกว่า โครงข่ายชั้นเดียว (single layer) และอาจจะมีหลาย layer ได้ใน Hidden Layer เรียกว่า โครงข่ายหลายชั้น (multilayer) โดยในแต่ละ layer นั้นจะติดต่อได้ระหว่าง node ที่อยู่คนละ layer เท่านั้นและการติดต่อนั้นจะมีเส้นที่ใช้เชื่อมกันระหว่าง node เรียกว่าลิงค์ link ซึ่งจะมีค่าถ่วงน้ำหนัก (weight) ของแต่ละลิงค์ เพื่อใช้บอกระดับความสำคัญของ input

4.2 การจัดแบ่งออกเป็นหมวดหมู่ด้วยแผนภาพต้นไม้ตัดสินใจ

เป้าหมายของการจัดแบ่งออกเป็นหมวดหมู่ (classification) จะเป็นการใช้กลุ่มของข้อมูลสอนระบบ (training data) สร้างโมเดลของคลาส label หรือจัดการข้อมูลให้อยู่ในกลุ่มที่กำหนดมาให้และยังสามารถใช้เพื่อแบ่งแยกข้อมูลใหม่ที่ไม่รู้ คลาส label ของมัน โดยการสร้างกฎเพื่อช่วยในการตัดสินใจจากข้อมูลที่มีอยู่ เพื่อใช้ทำนายแนวโน้มการเกิดขึ้นของข้อมูลที่ยังไม่เกิดขึ้น ในหลายๆชนิดของโมเดลเคยถูกสร้างขึ้นสำหรับการแบ่งแยกประเภทของข้อมูล เช่น โครงข่ายประสาทเทียม (neural network) ,โมเดลทางสถิติ(statistical model) ,และโมเดลต้นไม้ตัดสินใจ (decision tree model) เป็นต้นจะพบว่าโมเดลต้นไม้ตัดสินใจจะใช้ประโยชน์มากที่สุดในขอบเขตของ data mining เพราะมันมีความถูกต้องและค่อนข้างไม่ยุ่งยากในการคำนวณ โดยต้นไม้ตัดสินใจ (decision tree) จะเป็นโครงสร้างที่ใช้แสดงกฎ ที่ได้จากเทคนิคการจัดแบ่งข้อมูลออกเป็นหมวดหมู่ (classification) หรือการแยกแยะประเภทข้อมูลนั่นเอง และต้นไม้ตัดสินใจได้เคยถูกนำไปประยุกต์ใช้ในหลายแอพลิเคชั่นตั้งแต่ทางการแพทย์ไปจนถึงหลักการของเกมส์และเรื่องเกี่ยวกับธุรกิจ เป็นต้น (Han, Jiawei ,Kander and Micheline. 2001)

ในการดำเนินการแบ่งแยกประเภทข้อมูลจะกระทำใน 2 ขั้นตอน คือ ขั้นตอนการสร้างต้นไม้ตัดสินใจ (Tree Building) และขั้นตอนการตัดทอนหรือปรับแต่งต้นไม้ตัดสินใจ (Tree

Pruning) ในโมเดลการสร้างต้นไม้ตัดสินใจจะสร้างโดยการแบ่งแยกข้อมูลสอนระบบ (training data) แบบวนซ้ำ (recursive) อยู่บนพื้นฐานของเกณฑ์ที่เฉพาะเจาะจงที่ดีที่สุด และวิธีการในการปรับปรุงต้นไม้ตัดสินใจจะใช้การตัดโหนดหรือสาขาที่มีความน่าเชื่อถือสำหรับการแบ่งแยกข้อมูลน้อยที่สุดออก

อัลกอริทึมพื้นฐานของการสร้างต้นไม้ตัดสินใจ จะสร้างต้นไม้จากบนลงล่างแบบวนซ้ำด้วยวิธีการแบ่งปัญหาใหญ่เป็นปัญหาย่อย โดยการสร้างทรีจากรูทโหนดและจะแตกกิ่งย่อยออกไปตามโหนดปลายทาง(leaf node) ดังตารางที่ 4.1

ตารางที่4.1 อัลกอริทึมพื้นฐานในการสร้างต้นไม้ตัดสินใจจากตัวอย่างข้อมูลสอนระบบ

Algorithm: Generate_decision_tree.Generate a decision tree from the given training data.

Input : The training samples,samples,represented by discrete-valued attributes;the set of candidate attributes,attribute-list.

Output: A decision tree

Methode:

(1) create a node *N*;

(2) if *samples* are all of the same class, *C* then

(3) return *N* as a leaf node labeled with the class *C*;

(4) if *attribute-list* is empty then

(5) return *N* as a leaf node labeled with the most common class in samples; // majority voting

(6) select *test-attribute*,the attribute among *attribute-list* with the highest information gain;

(7) label node *N* with *test-attribute*;

(8) for each known value a_i of *test-attribute* //partition the samples

(9) grow a branch from node *N* for the condition *test-attribute* = a_i ;

(10) let s_i be the set of samples in *samples* for which *test-attribute* = a_i ; // a partition

(11) if s_i is empty then

(12) attach a leaf labeled with the most common class in *samples*;

(13) else attach the node returned by Generate_decision_tree (s_i , *attribute-list-test-attribute*);

4.2.1 การสร้างต้นไม้ตัดสินใจ Decision Tree Induction [1]

อัลกอริทึมพื้นฐานสำหรับสร้างต้นไม้ตัดสินใจเป็นอัลกอริทึมที่สร้างต้นไม้ตัดสินใจในลักษณะวิธีการตัดแบ่งออกเป็นส่วนย่อยๆ ทำซ้ำจากบนลงล่าง (top-down recursive divide-and-conquer) ซึ่งอัลกอริทึมในตารางที่4.1 จะเป็น version ของ ID3 โดยต้นไม้ตัดสินใจ จะสร้างขึ้นโดยการเรียนรู้จากข้อมูลสอนระบบเป็นหลัก จากตารางที่4.1 อธิบายอัลกอริทึมได้ดังนี้

- ต้นไม้เริ่มสร้างโหนดแรก N จากข้อมูลสอนระบบ(บรรทัดที่ 1)

- ถ้าข้อมูลตัวอย่างเป็น คลาสเดียวกันทั้งหมด class C ให้โหนด N กลายเป็นโหนดปลายหรือคลาสปลายทาง กำหนดเป็นคลาส C (บรรทัดที่ 2 และ 3)

- อัลกอริทึมใช้เครื่องมือวัดพื้นฐาน (entropy-based) รู้ว่าค่า information gain ซึ่งเป็นตัวที่ช่วยสำหรับเลือกคุณลักษณะ (attribute) ที่ดีที่สุดในการแยกแยะข้อมูลตัวอย่าง(samples) ไปเป็นคลาสเดี่ยวๆ (บรรทัดที่ 6) คุณลักษณะ (attribute) ที่ได้นี้จะกลายเป็นคุณลักษณะ(attribute) ทดสอบหรือตัดสินใจที่โหนด (บรรทัดที่ 7) คุณลักษณะ(attribute)ทั้งหมดในอัลกอริทึมนี้เป็นค่าไม่ต่อเนื่องดังนั้นต้องแปลงค่าคุณลักษณะ(attribute) ที่มีค่าเป็นตัวเลขต่อเนื่องให้เป็นค่าที่ไม่ต่อเนื่อง

- กิ่งหรือสาขาจะเป็นการสร้างสำหรับแต่ละค่าที่รู้ของคุณลักษณะที่จะทดสอบ(test attribute) และข้อมูลทดสอบจะถูกแบ่งแยกอย่างสอดคล้องตามลำดับ (บรรทัดที่ 8-10)

- อัลกอริทึมใช้กระบวนการเดิมทำวนซ้ำ เพื่อที่จะสร้างตัดไม้ตัดสินใจสำหรับข้อมูลตัวอย่างที่แต่ละส่วนแบ่ง (บรรทัดที่ 7)

- ส่วนของการเรียกซ้ำจะหยุด ก็ต่อเมื่อหนึ่งในเงื่อนไขข้างล่างเป็นจริง

(a) ทุกข้อมูลตัวอย่างสำหรับโหนดที่ให้มากลายเป็นคลาสเดียวกัน(บรรทัดที่ 2 และ 3) หรือ

(b) ไม่มีคุณลักษณะ(attribute) หรือข้อมูลตัวอย่างที่อาจจะถูกแบ่งแยกต่อไปอีกได้(บรรทัดที่ 4)ในกรณีนี้พิจารณาจากข้อมูลส่วนใหญ่ (majority voting) (บรรทัดที่ 5) และรวมถึงการแปลงโหนดที่ให้มาไปเป็นโหนดปลายทางและกำหนดเป็นคลาสในข้อมูลตัวอย่างส่วนใหญ่

(c) ไม่มีข้อมูลตัวอย่างสำหรับกิ่งหรือสาขา *test-attribute* = a_i (บรรทัดที่ 11) ในกรณีนี้โหนดปลายทางจะถูกสร้างด้วยคลาสส่วนใหญ่ในข้อมูลตัวอย่าง *samples* (บรรทัดที่ 12)

4.2.1.1การเลือกคุณลักษณะที่ใช้วัด Attribute Selection Measure

การเลือกคุณลักษณะที่ใช้ในการวัดหรือการหาตัวแบ่งแยกข้อมูล (split) ที่ดีที่สุดสำหรับแต่ละโหนดบนต้นไม้ (tree) เราจะหาค่า information gain ที่หาได้สำหรับแต่ละโหนดบนต้นไม้ (tree) โดยค่า information gain ที่มีค่าสูงที่สุด(ที่มีค่า entropy ที่ดีที่สุด) จะถูกใช้เพื่อเลือกคุณลักษณะ (Attribute) ที่จะใช้เป็นแอตทริบิวต์ทดสอบ(test attribute) สำหรับโหนดปัจจุบันที่แต่ละโหนด แอตทริบิวต์นี้จะลดสารสนเทศ (Information) ที่ต้องการเพื่อแบ่งแยกข้อมูลตัวอย่างในส่วนของ

ผลลัพธ์และสะท้อนให้เห็นค่าสุ่มที่น้อยที่สุดหรือค่าที่ไม่บริสุทธิ์ (Impurity) ในส่วนของผลลัพธ์ที่แบ่งนี้

ให้ S เป็นเซตประกอบด้วย s ข้อมูลตัวอย่างให้แอตทริบิวต์คลาส label มีจำนวน m ค่าไม่ซ้ำกัน กำหนดเป็น m คลาสไม่ซ้ำกัน C_i (สำหรับ i = 1,...,m) และให้ s_i เป็นจำนวนข้อมูลตัวอย่างของ S ในคลาส C_i ค่า สารสนเทศที่คาดหวัง(expected information) ที่ใช้สำหรับแบ่งแยกข้อมูลตัวอย่างที่ให้มา จะหาได้จาก

$$I(s_1, s_2, \dots, s_m) = -\sum_{i=1}^{m} p_i \log_2(p_i), \qquad (4.1)$$

เมื่อค่า p_i เป็นค่าความน่าจะเป็นที่ตัดสินข้อมูลตัวอย่างว่าเป็นส่วนหนึ่งของคลาส C_i ถูกประมาณโดย s_i/s

ให้ค่าแอตทริบิวต์ A มี v ค่าไม่ซ้ำกัน $\{a_1, a_2, \dots, a_v\}$ แอตทริบิวต์ A สามารถใช้เพื่อแบ่ง S ไปเป็นจำนวน v สับเซต $\{S_1, S_2, \dots, S_v\}$ ที่ S_j ประกอบด้วยข้อมูลตัวอย่างใน S ที่มีค่า a_j ของ A ถ้า A ถูกเลือกเป็นแอตทริบิวต์ทดสอบ (แอตทริบิวต์ที่ดีที่สุดที่ใช้สำหรับแบ่งแยกข้อมูล (split)) ดังนั้นสับเซตนี้จะมีค่าตรงกับกิ่งหรือสาขาที่โตขึ้นจากโหนดที่ประกอบด้วยเซต S ให้ s_{ij} เป็นจำนวนของข้อมูลตัวอย่างของคลาส C_i ในสับเซต S_j ค่า entropy หรือข้อมูลสารสนเทศที่คาดหวัง(expected information) อยู่บนพื้นฐานการแบ่งไปเป็นสับเซตโดย A ซึ่งจะหาได้จาก

$$E(A) = \sum_{j=1}^{v} \frac{s_{1j} + \dots + s_{mj}}{s} I(s_{1j}, \dots, s_{mj}). \qquad (4.2)$$

ในเทอมของ $\frac{s_{1j} + \dots + s_{mj}}{s}$ จะแสดงน้ำหนักของ j สับเซตและเป็นจำนวนของข้อมูลตัวอย่างในสับเซตหารด้วยจำนวนของข้อมูลตัวอย่างทั้งหมดใน S

$$I(s_{1j}, s_{2j}, \dots, s_{mj}) = -\sum_{i=1}^{m} p_{ij} \log_2(p_{ij}) \qquad (4.3)$$

เมื่อ $p_{ij} = \frac{s_{ij}}{|S_j|}$ และเป็นค่าความน่าจะเป็นที่ข้อมูลตัวอย่างใน s_j เป็นส่วนหนึ่งของคลาส C_i การเข้ารหัสสารสนเทศจะได้จากการแตกสาขาบน A

การเลือกคุณลักษณะหรือแอตทริบิวต์ เพื่อใช้ในการแบ่งแยกข้อมูลจะเลือกแอตทริบิวต์ที่มีค่า Information gain สูงสุด จากสูตร

$$Gain(A) = I(s_1, s_2, \dots, s_m) - E(A). \qquad (4.4)$$

โดยที่ A เป็นแอตทริบิวต์ที่จะพิจารณา และ s_i เป็นจำนวนข้อมูล S ในคลาส C_i

อัลกอริทึมจะคำนวณค่า Information Gain สำหรับแต่ละแอตทริบิวต์ แอตทริบิวต์ที่มีค่า Information Gain สูงสุดจะถูกเลือกให้เป็นแอตทริบิวต์ทดสอบในเซต S จากนั่นจะสร้างโหนด และกิ่งที่แสดงค่าในแอตทริบิวต์ และแบ่งข้อมูลต่อไปตามลำดับ(Han, Jiawei,Kander and Micheline. 2001)

ตารางที่ 4.2 ข้อมูลสอนระบบ

RID	age	income	student	credit_rating	Class:buys_computer
1	<=30	high	no	fair	no
2	<=30	high	no	excellent	no
3	31...40	high	no	fair	yes
4	>40	medium	no	fair	yes
5	>40	low	yes	fair	yes
6	>40	low	yes	excellent	no
7	31...40	low	yes	excellent	yes
8	<=30	medium	no	fair	no
9	<=30	low	yes	fair	yes
10	>40	medium	yes	fair	yes
11	<=30	medium	yes	excellent	yes
12	31...40	medium	no	excellent	yes
13	31...40	high	yes	fair	yes
14	>40	medium	no	excellent	no

ต่อไปนี้จะเป็นตัวอย่างของการสร้างต้นไม้ตัดสินใจ โดยการใช้ข้อมูลตัวอย่างจากตารางที่ 4.1 ซึ่งจะแสดงเซตของข้อมูลสอนระบบที่มาจากฐานข้อมูล AllElectronics customer โดยใน แอตทริบิวต์คลาส label : *buys_computer* จะมี 2 ค่าที่ไม่ซ้ำกัน(namely,{yes,no}) ดังนั้นจะมี 2 คลาสที่ไม่ซ้ำกัน (m=2) ให้ คลาส C_1 มีค่าตรงกับ yes และคลาส C_2 มีค่าตรงกับ no จะมี 9 ข้อมูลตัวอย่างของคลาส yes และ 5 ข้อมูลตัวอย่างของคลาส no ในการคำนวณหาค่า information gain ของแต่ละแอตทริบิวต์เราจะใช้สมการที่ (1) ในการคำนวณหาค่าสารสนเทศที่คาดหวัง (expected information) เพื่อใช้ในการแบ่งแยกข้อมูลตัวอย่างที่ให้มาโดย

$$I(s_1, s_2)=I(9,5) = -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14} = 0.940$$

จากนั้นเราจะคำนวณหาค่า entropy ของแต่ละแอตทริบิวต์เริ่มจากแอตทริบิวต์ age เราจะดูที่การกระจายตัวของ yes และ no สำหรับแต่ละค่าของ age เราจะคำนวณหาค่าสารสนเทศที่คาดหวังสำหรับแต่ละการกระจ่ายตัวนี้

สำหรับ age = "<=30":

สำหรับ age = “31…40”:

$s_{12} = 4$ $s_{22} = 0$ $I(s_{12}, s_{22}) = 0$

สำหรับ age = “>40”:

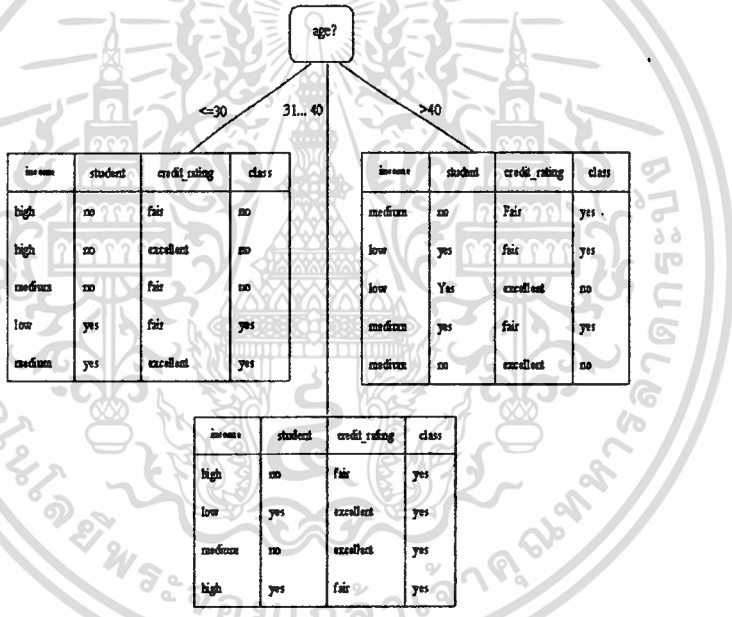
$s_{13} = 3$ $s_{23} = 2$ $I(s_{13}, s_{23}) = 0.971$

จากนั้นใช้สมการที่ 2 หาค่าสารสนเทศที่คาดหวังเพื่อใช้แบ่งแยกข้อมูลตัวอย่างที่ให้มา ถ้าข้อมูลตัวอย่างถูกแบ่งแยกโดยขึ้นกับ age คือ

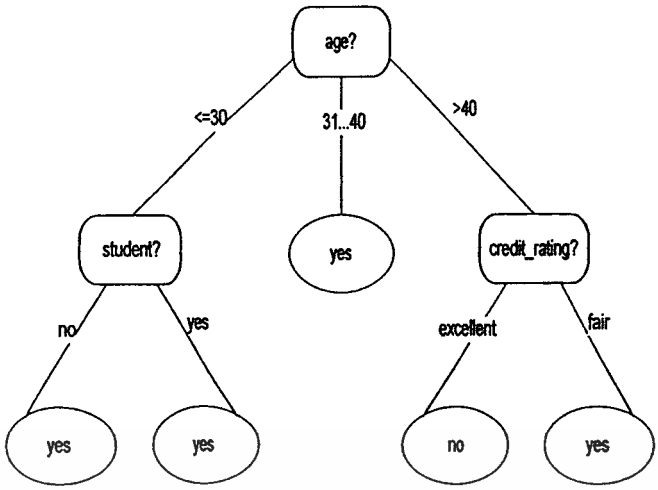
$$E(age) = \frac{5}{14} I(s_{11}, s_{21}) + \frac{4}{14} I(s_{12}, s_{22}) + \frac{5}{14} I(s_{13}, s_{23}) = 0.694$$

ดังนั้นค่า information gain ของ age จะเป็น

$Gain(age) = I(s_1, s_2) - E(age) = 0.246$



ภาพที่4.1 แสดงแอตทริบิวต์ทดสอบ age และแต่ค่าของแอตทริบิวต์



ภาพที่4.2 Decision tree สำหรับ buys computer

ในทำนองเดียวกันเราสามารถคำนวณ Gain(income)=0.029 , Gain(student)=0.151 ,และ Gain(credit_rate)=0.048 จะได้ว่า age จะมีค่า information gain มากที่สุดระหว่างแอตทริบิวต์ทั้งหมด ดังนั้น age จะถูกเลือกเป็นแอตทริบิวต์ทดสอบ (test attribute) และโหนดจะถูกสร้างขึ้นและถูกระบุ(labeled) ด้วย age จากนั้นกิ่งหรือสาขาจะเติบโตขึ้นสำหรับแต่ละค่าของแอตทริบิวต์ ตัวอย่างการแบ่งแยกจะแสดงในภาพที่ 4.1 จากภาพที่ 4.1 จะเห็นว่าแอตทริบิวต์ age จะมีค่า information gain สูงสุดและจะกลายเป็นแอตทริบิวต์ทดสอบที่โหนด root กิ่งหรือสาขาจะโตขึ้นสำหรับแต่ละค่าของแอตทริบิวต์ age และจากข้อมูลตัวอย่างจะเห็นว่าที่ส่วนของ age = "31...40" ทั้งหมดจะขึ้นอยู่ภายในคลาสเดียวกัน ดังนั้นทั้งหมดจะขึ้นอยู่กับคลาส yes ดังนั้นจึงควรสร้างโหนดปลายทางที่กิ่งหรือสาขานี้และระบุ(label) ด้วย yes และสุดท้ายอัลกอริทึมนี้จะให้ต้นไม้ตัดสินใจ (decision tree) ที่ถูกสร้างเสร็จแสดงในภาพที่ 4.2 จากภาพที่ 4.2 จะแสดงวแนวคิดในการเลือกซื้อ computer ที่แต่ละโหนดที่ไม่ใช่โหนดปลายทาง (nonleaf) จะแสดงแอตทริบิวต์ทดสอบ และที่แต่ละโหนดปลายทางจะแสดงคลาส (*buys_computer = yes* หรือ *buys_computer = no*)

4.2.1.2 Classification and Regression Tree (CART)

CART มาจาก Classification and Regression Tree ซึ่งพัฒนาโดย Leo Breiman, Jerome H.Friedman,Richard A.Olshen และ Charles J.Stone ในปี ค.ศ. 1984 ซึ่งเป็นวิธีในการสร้างแผนภูมิต้นไม้เพื่อช่วยในการตัดสินใจ(Decision Tree) โดย Decision Tree ที่ถูกสร้างขึ้นจากอัลกอริทึม CART จะอยู่ในรูปแบบของ Binary Tree ที่ประกอบด้วย 2 กิ่ง(Branch) ของแต่ละโหนดการตัดสินใจ โดยอัลกอริทึมนี้จะใช้กฎในการจำแนกข้อมูลที่จะทำการเรียนรู้ ซึ่งในการเลือกคุณสมบัติหรือแอตทริบิวต์ที่มีความสำคัญเพื่อใช้เป็นกฎแรกในการจำแนกข้อมูล(Breiman,Friedman,Olshen and Stone. 1984.)

4.2.1.2.1 Tree Building

ในขั้นตอนแรกจะทำการแบ่ง Training Data ออกเป็น 2 ส่วน โดยพิจารณาจากแอตทริบิวต์ที่มีความสำคัญมากที่สุดก่อน และทำซ้ำไปเรื่อยๆ จนกว่าตัวอย่างข้อมูลในแต่ละส่วนจะขึ้นกับคลาสใดคลาสหนึ่ง

ในการพิจารณาแอตทริบิวต์ในการสร้าง Decision Tree ของ CART Algorithm มีหลักเกณฑ์ในการแตกกิ่งด้วยกัน 2 วิธี คือ

- Gini Criterion
- Twoing Criterion

โดยทั้ง 2 วิธีมีขั้นตอนการทำงานดังนี้

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);

Gini Criterion

การแตกกิ่งด้วย Gini Criterion สูตรที่นำมาใช้คือ

$$\text{Gini (S)} = 1 - \sum_{j=1}^{n} p^2 j \qquad (4.5)$$

โดย S เป็นชุดของข้อมูลที่เก็บตัวอย่างจาก N คลาส

P_j เป็นความถี่สัมพันธ์ของคลาส j ใน S

$$\text{Gini}_{\text{split}} (S) = N_1 / N \text{ gini } (S_1) + N_2 / N \text{ gini } (S_2) \qquad (4.6)$$

โดยจะมีขั้นตอนในการหาจุดแบ่ง (Split Point) ในการสร้าง Tree ดังนี้

1. พิจารณาแอตทริบิวต์ที่จะเป็น Test Node โดยแอตทริบิวต์ที่เหมาะสมที่จะใช้เป็น Split Point ในการวิเคราะห์ข้อมูลโดยใช้สูตรของ Gini Index ที่กล่าวมาข้างต้น ซึ่งจะเลือกแอตทริบิวต์ที่มีค่า $\text{Gini}_{\text{sprit}}$ น้อยที่สุด

2. เมื่อได้แอตทริบิวต์ที่นำมาใช้เป็นจุดแบ่งแล้วจะมีการกำหนดจุดแบ่งนั้นตามประเภทของข้อมูล

   — การแบ่งข้อมูลแอตทริบิวต์ที่เป็นประเภทตัวเลข(Numeric/Continuous Attribute) เป็นการแบ่งที่มีลักษณะเป็น 2 ทาง(Binary Split) จากตัวอย่างการให้เครดิตลูกค้าของแอตทริบิวต์ Incomes จะได้ Incomes <= v โดย v เป็นตัวเลขที่เป็นไปได้ของแอตทริบิวต์ Incomes ที่ถูกเรียงลำดับแล้ว ซึ่งอยู่ในรูปแบบ $v_1, v_2, \ldots, v_n$ เมื่อได้ค่า Best Split Point ที่ v_1 ให้นำค่า $v_1 + v_2/2$ เป็น Best Split Point

   — การแบ่งข้อมูลข้อมูลแอตทริบิวต์ที่เป็นประเภทจัดหมวดหมู่(Categorical Attribute) โดยให้ S(A) เป็นเซตของค่าที่เป็นไปได้ของแอตทริบิวต์ A เมื่อ X $\subset$ Domain(A) ดังนั้นจำนวนซับเซตที่เป็นไปได้เท่ากับ 2^n

Twoing Criterion

การแตกกิ่งด้วย Twoing Criterion จะพิจารณาแอตทริบิวต์ที่ดีที่สุดจากสูตร

$$\Phi(s|t) = 2P_L P_R - \sum_{j=1}^{\#classes} |P(j|t_L) - P(j|t_R)| \qquad (4.7)$$

โดย t_L = โหนดลูกทางด้านซ้ายที่โหนด t

t_R = โหนดลูกทางด้านขวาที่โหนด t

P_L = จำนวนเรคคอร์ดของโหนดลูกทางซ้าย / จำนวนเรคคอร์ดของข้อมูลที่ใช้

P_R = จำนวนเรคคอร์ดของโหนดลูกทางขวา / จำนวนเรคคอร์ดของข้อมูลที่ใช้

$P(j/t_R)$ = จำนวนของคลาส j ที่เรคคอร์ด t_R / จำนวนเรคคอร์ดที่ t

$P(j/t_L)$ = จำนวนของคลาส j ที่เรคคอร์ด t_L / จำนวนเรคคอร์ดที่ t

โดยในการพิจารณาการแตกกิ่งของ Twoing Criterion นี้จะเลือกแอตทริบิวต์ที่มีค่ามากที่สุดมาใช้ในการตัดสินใจ ซึ่งในสัมมนานี้จะยกตัวอย่างการแตกกิ่งด้วย Twoing Criterion

ตารางที่ 4.3 ตัวอย่างข้อมูลการให้เครดิตลูกค้า

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

จากข้อมูลตัวอย่างการให้เครดิตลูกค้าโดยพิจารณาจากคุณสมบัติเงินออม(Savings), สินทรัพย์(Assets), รายได้(Incomes) เพื่อใช้ในการให้เครดิตลูกค้าสามารถนำมาใช้ในการจำแนกประเภทลูกค้าได้ 2 แบบคือ ลูกค้าที่มีเครดิตดีและลูกค้าที่มีเครดิตไม่ดี โดยเมื่อนำข้อมูลดังกล่าวมาสร้างเป็น Decision Tree ด้วยอัลกอริทึม CART ซึ่งจากแบบจำลองที่ได้จะเป็น Binary Tree จึงต้องมีการแบ่งกรณีที่ได้ของแต่ละคุณสมบัติออกเป็น 2 กรณีย่อยที่จะเกิดได้ดังต่อไปนี้

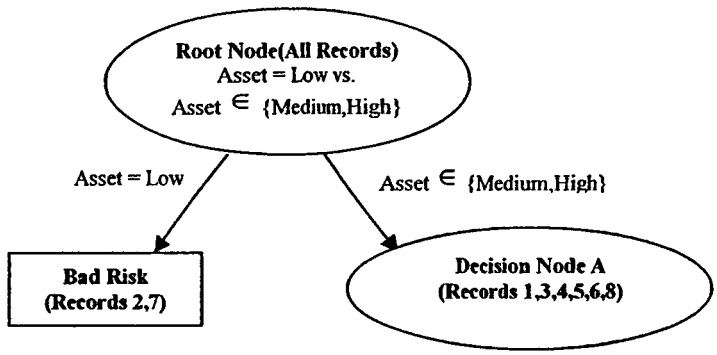
Candidate Split	Left Child Node t_L	Right Child Node t_R
1	Savings = low	Saving ∈ {medium,high}
2	Savings = Medium	Saving ∈ {low,high}
3	Savings = High	Saving ∈ {low,medium}
4	Assets = low	Assets ∈ {medium,high}
5	Assets = Medium	Assets ∈ {low,high}
6	Assets = High	Assets ∈ {low,medium}
7	Incomes <=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

จากการแทนสูตร $\Phi(s|t)$ จะได้ค่าดังต่อไปนี้

ตารางที่ 4.4 ผลลัพธ์ที่ได้จากการแทนสูตร $\Phi(s|t)$

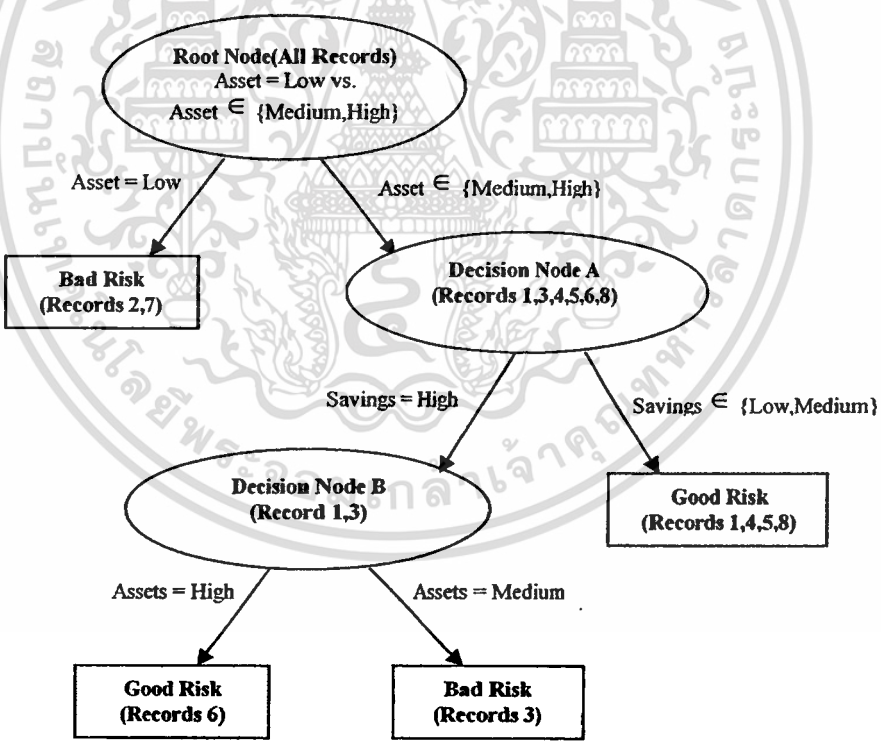
Split	P_L	P_R	$P(j\|t_L)$	$P(j\|t_R)$	$2P_L P_R$	$\Phi(s\|t)$
1	0.375	0.625	G: .333	G: .8	0.46875	0.4378
			B: .667	B: .2		
2	0.375	0.625	G: 1	G: .4	0.46875	0.5625
			B: .0	B: .6		
3	0.25	0.75	G: .5	G: .667	0.375	0.1235
			B: .5	B: .333		
4	0.25	0.75	G: 0	G: .833	0.375	0.6248
			B: .1	B: .167		
5	0.5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

นำคุณสมบัติที่ได้ค่า $\Phi(s|t)$ มากที่สุดมาใช้เป็นโหนดในการตัดสินใจโหนดแรกโดยใน Candidate Split ที่ 4 มีค่ามากที่สุดจะนำมาใช้เป็นโหนดเริ่มต้น ซึ่งใช้แอตทริบิวต์สินทรัพย์เพื่อนำมาใช้ในการพิจารณาการให้เครดิตลูกค้า ดังรูปที่ 4.3



ภาพที่4.3 โหนดแรกในการตัดสินใจของ Decision Tree

หลังจากเลือกแอตทริบิวต์แรกมาเป็น Root Note ได้แล้วก็ทำซ้ำไปเรื่อยๆ จนกว่าจะได้โหนดสุดท้ายที่เป็น แอตทริบิวต์เป้าหมาย(Target Attribute) โดยในการคำนวณจะไม่นำ Candidate Split ที่ 4 มาพิจารณาอีก ซึ่งสุดท้ายแล้วจะได้ Decision Tree เพื่อใช้ในการให้เครดิตลูกค้า ดังรูปที่ 4.4



ภาพที่4.4 Decision Tree เพื่อใช้ในการให้เครดิตลูกค้า

บทที่ 5
การวิเคราะห์และออกแบบ

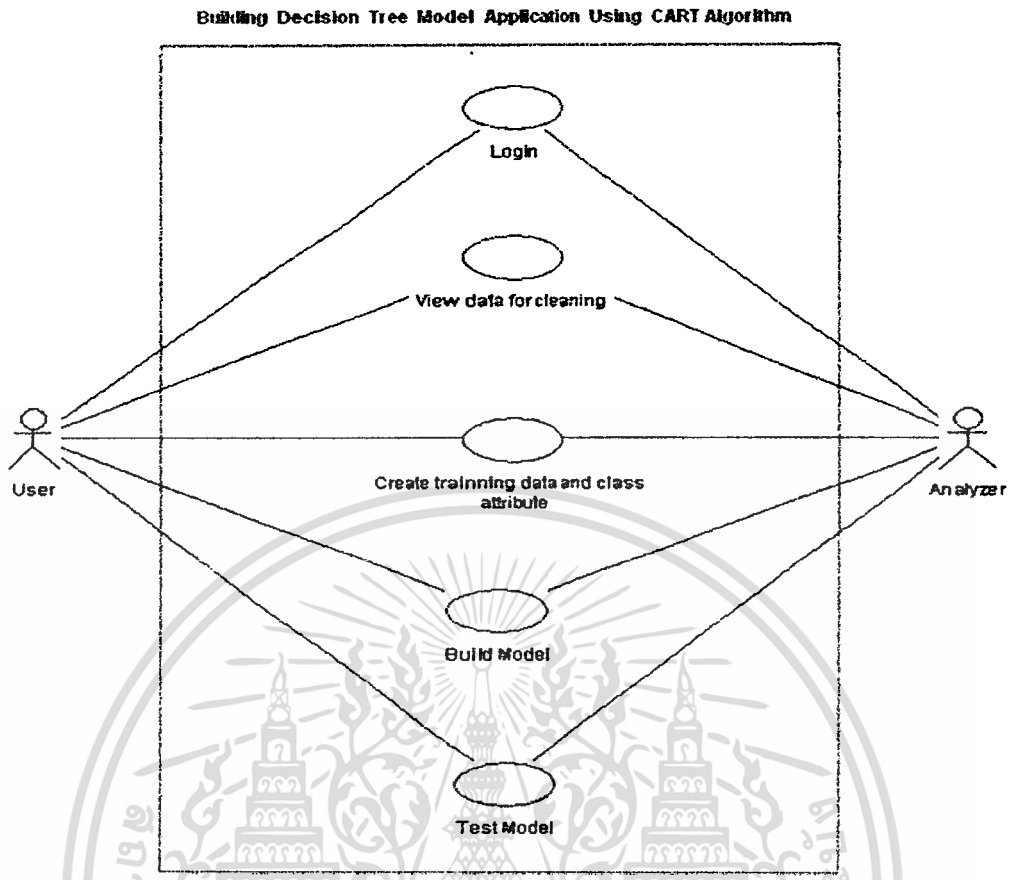
หลังจากทำการศึกษาวิธีการในการทำดาต้าไมนิ่งเกี่ยวกับการพยากรณ์ และการสร้างแบบจำลองสำหรับการพยากรณ์โดยใช้อัลกอริทึ่ม CART แล้วนั้น จากนั้นจึงได้ทำการวิเคราะห์และออกแบบโปรแกรมสำหรับใช้ในการสร้างแบบจำลองในการจัดแบ่งข้อมูลออกเป็นหมวดหมู่หรือแยกแยะประเภทข้อมูล แสดงออกมาในลักษณะเป็นโครงสร้างต้นไม้สำหรับใช้ในการตัดสินใจต่อไป

5.1 การวิเคราะห์และออกแบบโปรแกรม

การวิเคราะห์และออกแบบแบบจำลองเชิงแนวคิดของระบบงาน จะทำโดยใช้ภาษายูเอ็มแอล ซึ่งจะแสดงด้วยไดอะแกรมแบบต่างๆ โดยในการวิเคราะห์และออกแบบแบบจำลองเชิงแนวคิดของระบบงานนี้จะแสดงด้วย ยูสเคสไดอะแกรม แอ็กทิวิตีไดอะแกรม คลาสไดอะแกรม ซีและเควนซ์ไดอะแกรม

5.2 ยูสเคสไดอะแกรม

ยูสเคสไดอะแกรมเป็นแบบจำลองของระบบในมุมมองของผู้ใช้งานระบบ ซึ่งช่วยให้นักวิเคราะห์ระบบกับผู้ใช้งานระบบสื่อสารเข้าใจตรงกันว่าผู้ใช้งานระบบจะนำระบบไปใช้งานอะไร โดยมีองค์ประกอบ 2 ส่วน คือ แอ็กเตอร์ และยูสเคส โดยที่ยูสเคสจะแสดงถึงขอบเขตของระบบงาน ส่วนแอ็กเตอร์คือสิ่งที่อยู่นอกระบบซึ่งจะเป็นทั้งผู้กระตุ้นให้ระบบเกิดการทำงาน หรือรับผลลัพธ์จากการกระทำของระบบด้วยก็ได้ (สุนทริน วงศ์ศิริกุล. 2543 : 52)



ภาพที่ 5.1 ยูสเคสของระบบ

ยูสเคสไดอะแกรมของการพัฒนาโปรแกรมดาต้าไมน์นิ่งแบบ Decision Tree โดยใช้อัลกอริทึ่ม CART ประกอบด้วย แอ็กเตอร์ 2 แอ็กเตอร์ และยูสเคส 5 ยูสเคส ซึ่งมีรายละเอียดดังนี้คือ

แอ็กเตอร์ที่เกี่ยวข้องกับระบบ มี 2 แอ็กเตอร์ ดังนี้ คือ

1. Analyser คือ ผู้ที่ทำหน้าที่วิเคราะห์ก็คือระบบ
2. User คือ ผู้ใช้งานระบบ

ยูสเคสที่เกี่ยวข้องกับระบบ มี 5 ยูสเคส ดังนี้ คือ

1. Login เป็นฟังก์ชันที่ใช้ตรวจสอบการเข้าใช้งานระบบ
2. View data for cleaning เป็นฟังก์ชันที่ให้ผู้ใช้ทำการตรวจสอบข้อมูลก่อนทำการแก้ไขหรือเพิ่มเติมข้อมูลใหม่เพื่อใช้ในการทำ Classification
3. Create training data and class attribute เป็นฟังก์ชันที่ใช้ในการสร้างข้อมูลที่สอนระบบที่ได้ทำจัดรูปแบบข้อมูลใหม่เรียบร้อยแล้ว
4. Build Model เป็นฟังก์ชันที่ใช้ในการสร้างทรีหรือกฎ
5. Test Model ประกอบด้วยฟังก์ชันที่ใช้ในการทดสอบโมเดล และ Classification เป็นฟังก์ชันที่ใช้ในการแยกประเภทข้อมูลโดยใช้กฎที่ได้จากการสร้างทรี

5.3 ยูสเคสดิสคริพฌัน

ยูสเคสดิสคริพฌันเป็นส่วนที่ใช้อธิบายรายละเอียดการทำงานของแต่ละฟังก์ชันว่าฟังก์ชันมีการทำงานอย่างไรและมีแอ็กเตอร์อะไรหรือฟังก์ชันใดที่สัมพันธ์กันกับฟังก์ชันนี้บ้าง โดยรายละเอียดของยูสเคสดิสคริพฌันสามารถแสดงได้ดังตารางที่ 5.1-5.5

ตารางที่ 5.1 อธิบายยูสเคสดิสคริพฌันของการ Login

Use case name: Login	ID: 1	Importance level: High
Primary actor: User	Use case type: Detail,essential	
Stakeholders and interest: User – ผู้ใช้งานระบบ Analyzer-ระบบที่พัฒนา		
Brief description: use case นี้ใช้เพื่ออธิบายว่า วิธีการที่ เข้าใช้งานระบบจะต้องมีการ Login ป้อนข้อมูล Username,Password,ชื่อ Database ทุกครั้ง		
Trigger: User ต้องขอ Username,password,ชื่อ Database จากผู้ดูแลระบบเพื่อขอเข้ามาใช้งาน Type: External		
Relationship: Association: User,Analyzer Include: - Extend: - Generalization: -		
Normal flow of events: User เข้ามาในระบบและระบบจะตรวจสอบว่ามีสิทธิใช้งานระบบหรือไม่ ถ้าไม่ก็จะต้องให้ผู้ดูแลระบบทำการป้อนข้อมูลรายละเอียดต่างๆของ User เอง และ User ก็จะต้องกำหนด Username,Password และชื่อ Database เพื่อ login เข้ามาในระบบในภายหลัง มีขั้นตอนดังนี้ 1. ผู้ใช้ป้อน Username ,Password 2. ผู้ใช้ป้อนชื่อ Database ที่ต้องการซึ่งระบบจะทำการติดต่อได้แค่ฐานข้อมูลออราเคิลเท่านั้น 3. ผู้ใช้กดปุ่ม ตกลง 4. ระบบเช็คสิทธิการเข้าใช้ 5. ผู้ใช้เข้าใช้งานระบบ		
Subflows:		
Alternate/exceptional flows: 2.1 ถ้าไม่มีสิทธิก็ไม่สามารถเข้ามาใช้งานระบบได้ ต้องไปติดต่อกับผู้ดูแลระบบก่อน		

ตารางที่ 5.2 อธิบายยูสเคสดิสคริพฌันของการ view data for cleaning

Use case name: View data for cleaning	ID: 2	Importance level: High
Primary actor: User,Analyzer	Use case type: Detail,essential	
Stakeholders and interest: User – ผู้ใช้งานระบบ Analyzer – โปรแกรมที่ทำการพัฒนา		
Brief description: use case นี้ใช้เพื่ออธิบายว่า ข้อมูลที่ใช้ในระบบจะต้องทำการให้ ผู้ใช้ตรวจสอบความถูกต้องและทำการเพิ่มเติมหรือแก้ไขข้อมูลที่ต้องการใช้เองเสียก่อนที่จะนำมาใช้ในระบบต่อไป		
Trigger: Type: External		
Relationship: Association: - User , Analyzer Include: - Extend: - Generalization: -		
Normal flow of events:ข้อมูลที่ใช้ในระบบนั้นจะต้องมีข้อมูลที่ถูกต้องจะทำการให้ผู้ใช้ทำการตรวจสอบความถูกต้องของข้อมูลก่อน มีขั้นตอนดังนี้ 1. ผู้ใช้งานเลือกเทเบิลที่ต้องการนำมาสร้างโมเดล 2. ระบบจะแสดงแอตทริบิวต์ที่มีทั้งหมดของเทเบิลที่เลือก 3. ผู้ใช้ดูคุณสมบัติและตรวจสอบข้อมูลขยะของแอตทริบิวต์นั้นในเทเบิล 4. ถ้ามีข้อมูลขยะจะให้ผู้ใช้เลือกว่าจะทำการ ลบ หรือทับค่าข้อมูลที่เป็นขยะด้วยค่าที่มากที่สุดกรณีที่เป็นตัวอักษรแต่ถ้าเป็นตัวเลขก็จะทับด้วยค่าเฉลี่ยของข้อมูลทั้งหมด 5. ระบบจะทำการบันทึกข้อมูลเข้าในระบบ		
Subflows:		
Alternate/exceptional flows: 3.1 ถ้าไม่มีข้อมูลขยะก็จะทำขั้นตอนต่อไปตามลำดับ		

ตารางที่ 5.3 อธิบายยูสเคสดิสคริพฌันของการ Create training data and class attribute

Use case name: Create trainig data and class attribute	ID: 3	Importance level: High
Primary actor: User,Analyzer	Use case type: Detail,essential	
Stakeholders and interest: User – ผู้ใช้งานระบบ Analyzer-ระบบที่พัฒนา		
Brief description: use case นี้ใช้เพื่ออธิบายว่าสำหรับการใช้การสร้างโมเดลจะใช้ข้อมูลที่ได้ทำการผ่านการเปลี่ยนรูปแบบข้อมูลให้เหมาะสมเพื่อใช้ในการสอนระบบ โดยในยูสเคสนี้จะเป็นการสร้างข้อมูลให้อยู่ในรูปข้อมูลสอนระบบเข้าสู่ระบบ และ class attribute ที่ต้องการแยกประเภทข้อมูลเข้าสู่ระบบงาน		
Trigger: ข้อมูลจะต้องมีการทำความสะอาด และจัดการข้อมูลที่ผิดปกติเป็นที่เรียบร้อยแล้ว Type: External		
Relationship: Association: - User,Analyzer Include: - Extend: Register Generalization: -		
Normal flow of events:ในการสร้างโมเดลจำเป็นจะต้องมีข้อมูลที่จัดรูปแบบใหม่เพื่อใช้ในการสร้างโมเดลเรียกว่าข้อมูลสอนระบบ training data โดยในยูสเคสนี้จะเป็นการที่ผู้ใช้สร้างข้อมูลสอนระบบที่ได้ผ่านการจัดรูปแบบข้อมูลให้อยู่ในรูปแบบที่เหมาะสมเข้ามาในระบบ มีขั้นตอนดังนี้ 1. ผู้ใช้ป้อนชื่อของข้อมูลสอนระบบที่ต้องการจะสร้าง 2. ผู้ใช้เลือกเทเบิลของข้อมูลที่ต้องการจะสร้างข้อมูลสอนระบบ 3. ระบบแสดงแอตทริบิวต์ทั้งหมด 4. ผู้ใช้เลือกแอตทริบิวต์ที่ต้องการ 5. ผู้ใช้ป้อนเปอร์เซ็นต์ของข้อมูลที่ต้องการนำมาใช้ทดสอบและสร้างโมเดลเป็นคนละชุดข้อมูลกัน 6. ผู้ใช้กดปุ่ม ตกลง 7. ระบบจะทำการสร้างข้อมูลสอนระบบให้และระบบแสดงผลลัพธ์ของการสร้างชุดข้อมูลสอนระบบ trainning set		
Subflows:		

ตารางที่ 5.3 (ต่อ)

Alternate/exceptional flows: 5.1 หากระบบไม่สามารถสร้างข้อมูลสอนระบบได้ระบบจะแจ้งข้อความว่า ไม่สามารถสร้างข้อมูลสอนระบบ 5.2 ในหน้าจอผลลัพธ์ระบบจะไม่แสดงผลลัพธ์ของข้อมูลสอนระบบ

ตารางที่ 5.4 อธิบายยูสเคสดิสคริพฌันของการสร้างแบบจำลอง

Use case name: Build Model	ID: 4	Importance level: High
Primary actor: User,Analyzer	Use case type: Detail,essential	
Stakeholders and interest: User – ผู้ใช้งานระบบ Analyzer – โปรแกรมที่ทำการพัฒนา		
Brief description: use case นี้ใช้เพื่ออธิบายว่า หลังจากทำการกำหนด class attribute และได้ข้อมูลสอนระบบเป็นที่เรียบร้อยแล้วจะทำการสร้างโมเดลที่ใช้ในการพยากรณ์ต่อไป		
Trigger: จะต้องมีข้อมูลสอนระบบเป็นที่เรียบร้อย และจะต้องกำหนด Class Attribute ก่อน Type: External		
Relationship: Association: - User,Analyzer Include: - Extend: - Generalization: -		

ตารางที่ 5.4 (ต่อ)

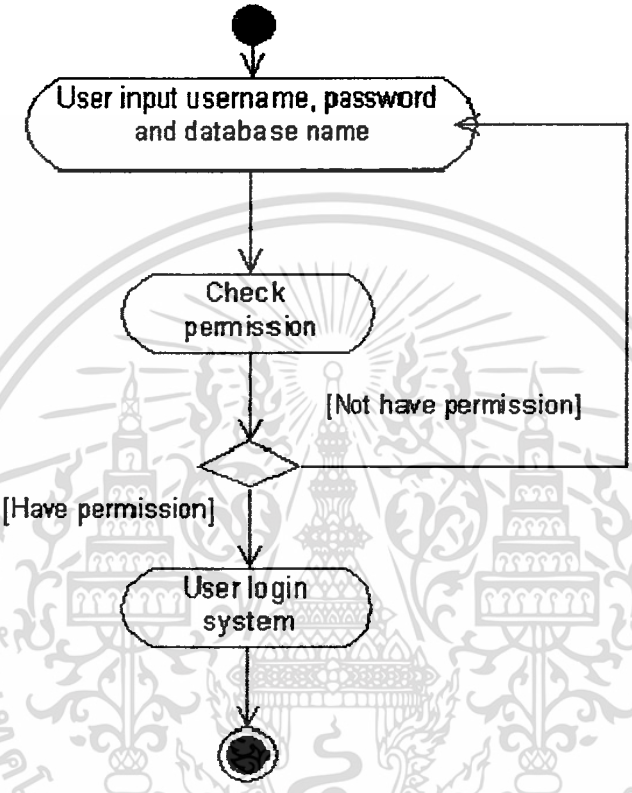
Normal flow of events:หลังจากได้ข้อมูลสอนระบบและ class attribute เป็นที่เรียบร้อยแล้ว หลังจากนั้นจะมาทำการสร้างแบบจำลองหรือกฎเพื่อนำไปใช้ในการวิเคราะห์ข้อมูลใหม่ๆที่เข้ามา มีขั้นตอนดังนี้ 1. ผู้ใช้ป้อนข้อมูลชื่อโมเดลที่ต้องการจะสร้าง 2. ผู้ใช้เลือกข้อมูลสอนระบบที่ได้สร้างมาจากขั้นตอนการสร้างข้อมูลสอนระบบ 3. ผู้ใช้ป้อนชื่อคลาสที่ต้องการ 4. ระบบจะทำการอ่านข้อมูลจากชุดข้อมูลสอนระบบ 5. ระบบอ่านชุดข้อมูลของคลาสที่กำหนด 6. ระบบหาแอตทริบิวต์ที่จะเป็นแอตทริบิวต์ทดสอบ 7. ระบบสร้างโหนดของต้นไม้ 8. ระบบเปรียบเทียบค่าในข้อมูลสอนระบบ 9. ระบบเช็คว่าทุกค่าข้อมูลไม่ได้อยู่ในคลาสเดียวกันและไม่มีค่าว่าง 10. ระบบคำนวณค่า gini ของแต่ละแอตทริบิวต์สำหรับแตกต้นไม้ 11. ระบบเลือกค่าแอตทริบิวต์ทดสอบในลิสต์ของแอตทริบิวต์ที่มีค่า gini มากที่สุด 12. ระบบกำหนดโหนดด้วยค่าแอตทริบิวต์ทดสอบที่มีค่า gini มากที่สุด 13. ระบบหาค่าข้อมูล ของแอตทริบิวต์ทดสอบ และสร้างกิ่งของต้นไม้ด้วยค่านั้น 14. ระบบหากลุ่มข้อมูลของข้อมูลสอนระบบที่อยู่ภายใต้ค่าของแอตทริบิวต์ทดสอบนั้น 15. ระบบเช็คกลุ่มของข้อมูลสอนระบบนั้นว่ายังมีค่าอยู่ก็จะเข้ากระบวนการสร้างต้นไม้ต่อไปจนกระทั่งไม่มีแอตทริบิวต์ทดสอบหรือทุกค่าอยู่ในคลาสเดียวกัน
Subflows:
Alternate/exceptional flows: 8.1. ทุกค่าของข้อมูลอยู่ในคลาสเดียวกัน 8.2. กำหนดโหนดของต้นไม้นั้นด้วยคลาสนั้น

ตารางที่ 5.5 อธิบายยูสเคสดิสคริพฌันของการ Test Model

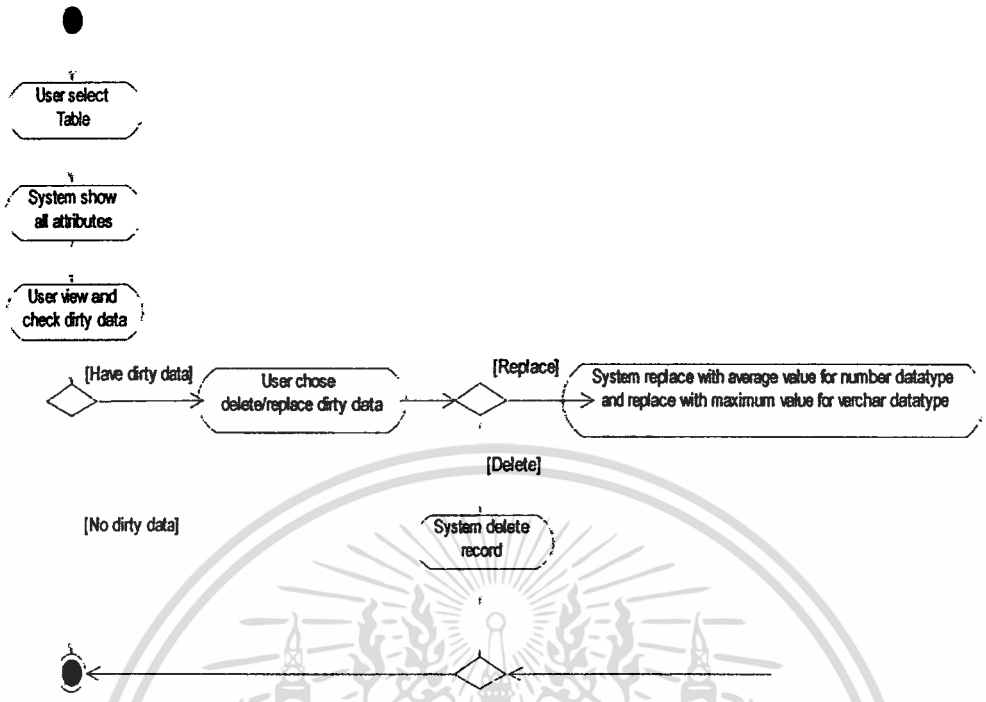
Use case name: Test Model	ID: 5	Importance level: High
Primary actor: User,Analyzer	Use case type: Detail,essential	
Stakeholders and interest: User – ผู้ใช้งานระบบ Analyzer – โปรแกรมที่ทำการพัฒนา		
Brief description: use case นี้ใช้เพื่ออธิบายว่า เมื่อทำการสร้างโมเดลเป็นที่เรียบร้อยแล้วจะให้ทำการทดสอบความถูกต้องของโมเดลโดยการใช้ข้อมูลคนละชุดกับที่ใช้ในการสร้างข้อมูลสอนระบบนำมาใช้ทดสอบ รวมถึงสามารถพยากรณ์ข้อมูลที่ผู้ใช้ได้ทำการกรอกข้อมูลใหม่เข้าไปโดยใช้โมเดลที่เลือก		
Trigger: ผ่านขั้นตอนการสร้างโมเดลเสียก่อน Type: External		
Relationship: Association: - User , Analyzer Include: - Extend: - Generalization: -		
Normal flow of events: เมื่อได้โมเดลจากขั้นตอนการสร้างโมเดล จะให้ผู้ใช้เลือกโมเดลและข้อมูลอีกชุดมาทำการทดสอบโมเดลที่สร้าง มีขั้นตอนดังนี้ 1. ผู้ใช้เลือกชื่อโมเดลที่ต้องการทดสอบ 2. ผู้ใช้เลือกข้อมูลที่ต้องการนำมาทดสอบโมเดล 3. ผู้ใช้กดปุ่ม ตกลง 4. ระบบจะทำการแม๊พข้อมูลที่ต้องการทดสอบกับข้อมูลของโมเดล 5. ระบบจะแสดงผลของการทดสอบโมเดล		
Subflows:		
Alternate/exceptional flows: 3.1. หากข้อมูลไม่แม๊พระบบจะแจ้งข้อความว่าข้อมูลไม่แม๊พกับข้อมูลของโมเดล		

แอ็กทิวิตีไดอะแกรม

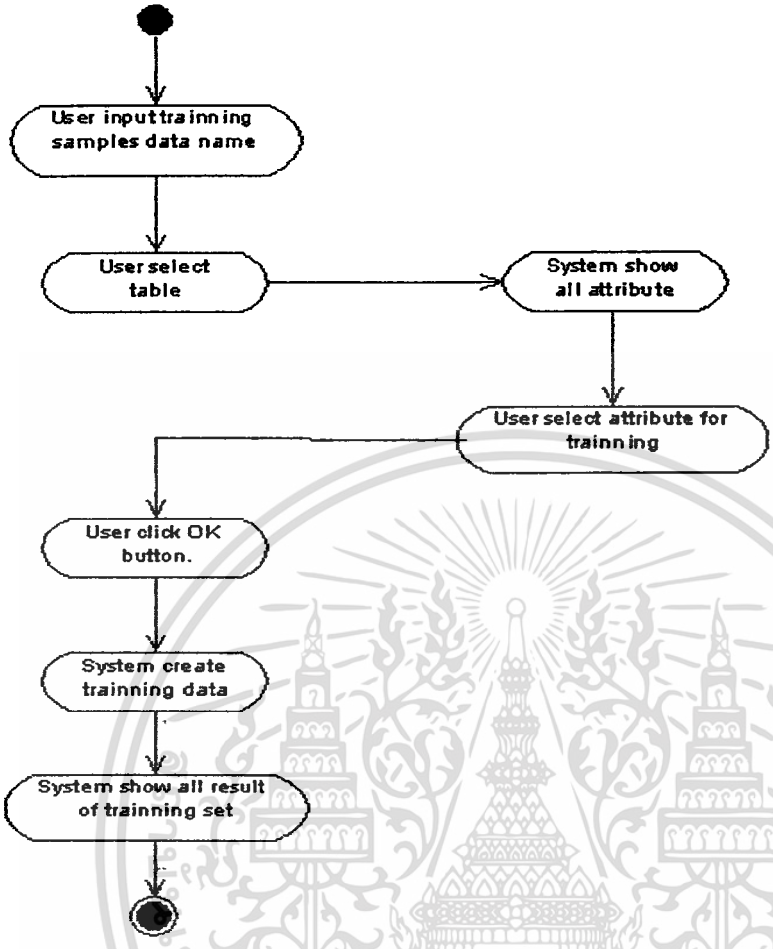
แอ็กทิวิตีไดอะแกรมใช้อธิบายการรายละเอียดขั้นตอนการทำงานของระบบในแต่ละยูสเคส โดยที่ขั้นตอนการทำงานในแต่ละขั้นตอนจะเรียกว่า แอ็กทิวิตี (สุนทริน วงศ์ศิริกุล. 2543 : 87) โดยรายละเอียดของแอ็กทิวิตีไดอะแกรมสามารถแสดงได้ดังภาพที่ 5.2-5.6



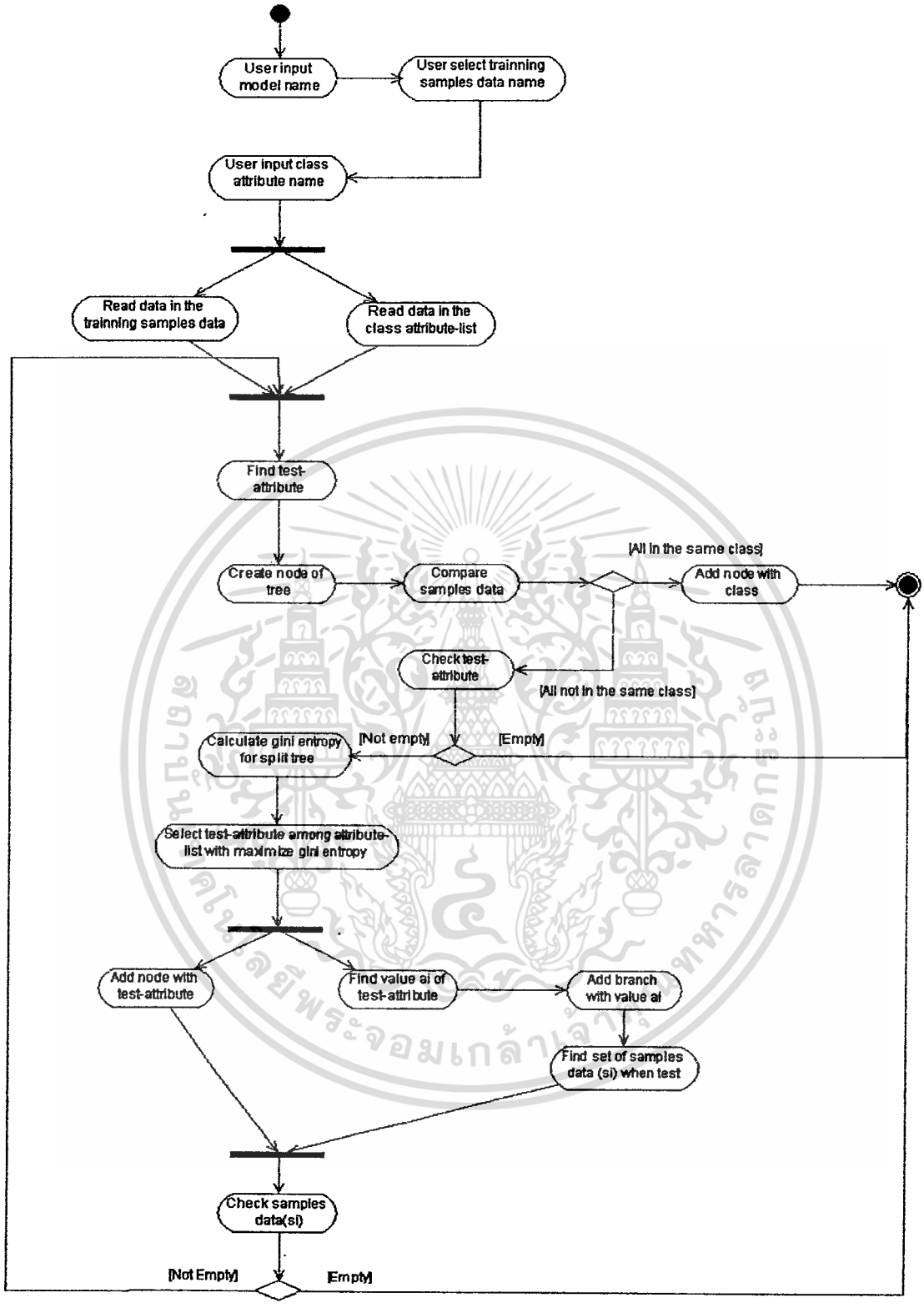
ภาพที่ 5.2 แอ็กทิวิตีไดอะแกรมของยูสเคส Login



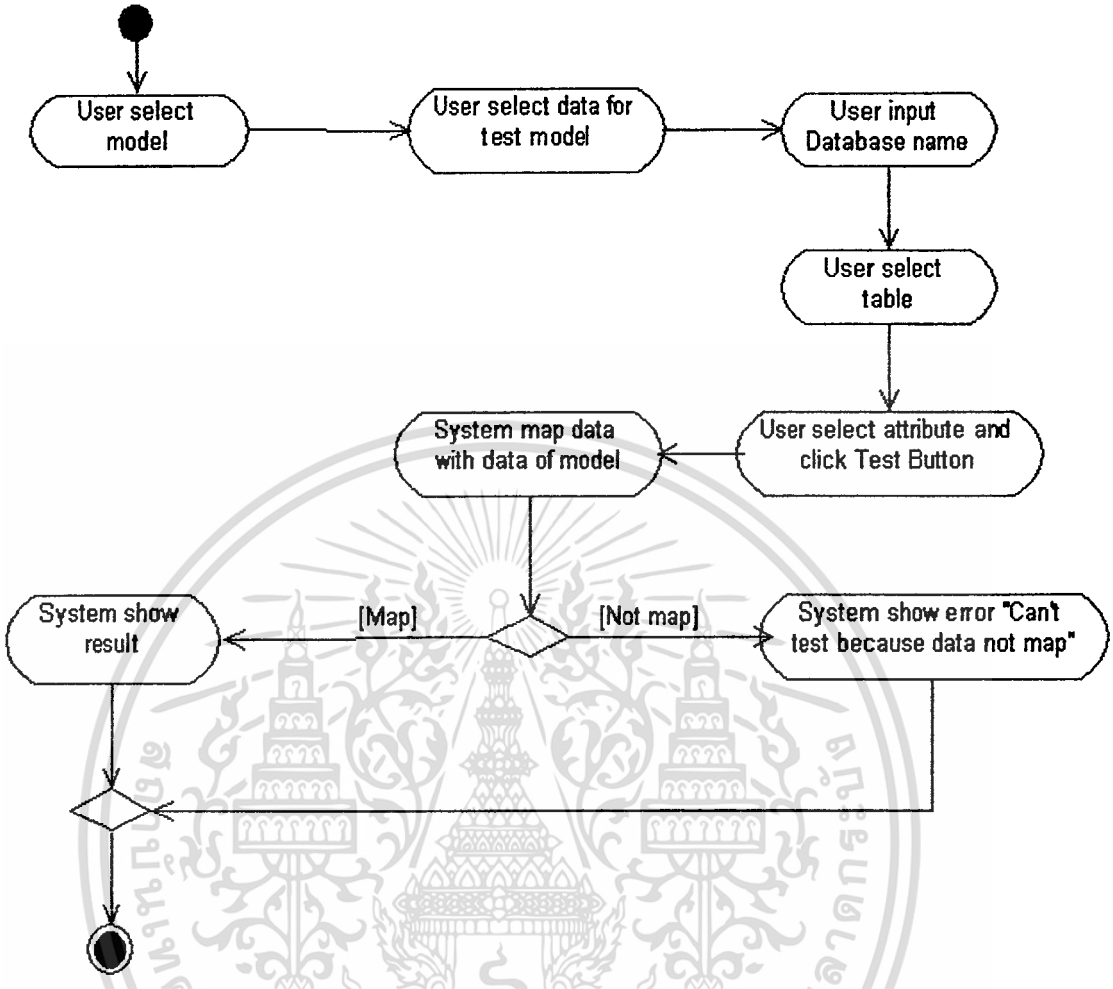
ภาพที่ 5.3 แอ็กทิวิตีไดอะแกรมของยูสเคส View data for cleaning



ภาพที่ 5.4 แอ็กทิวิตี้ไดอะแกรมของยูสเคส Create samples data and class attribute



ภาพที่ 5.5 แอ็กทิวิตีไดอะแกรมของยูสเคส Build Model



ภาพที่ 5.6 แอ็กทิวิตีไดอะแกรมของยูสเคส Test Model

5.4 ซีเควนซ์ไดอะแกรม

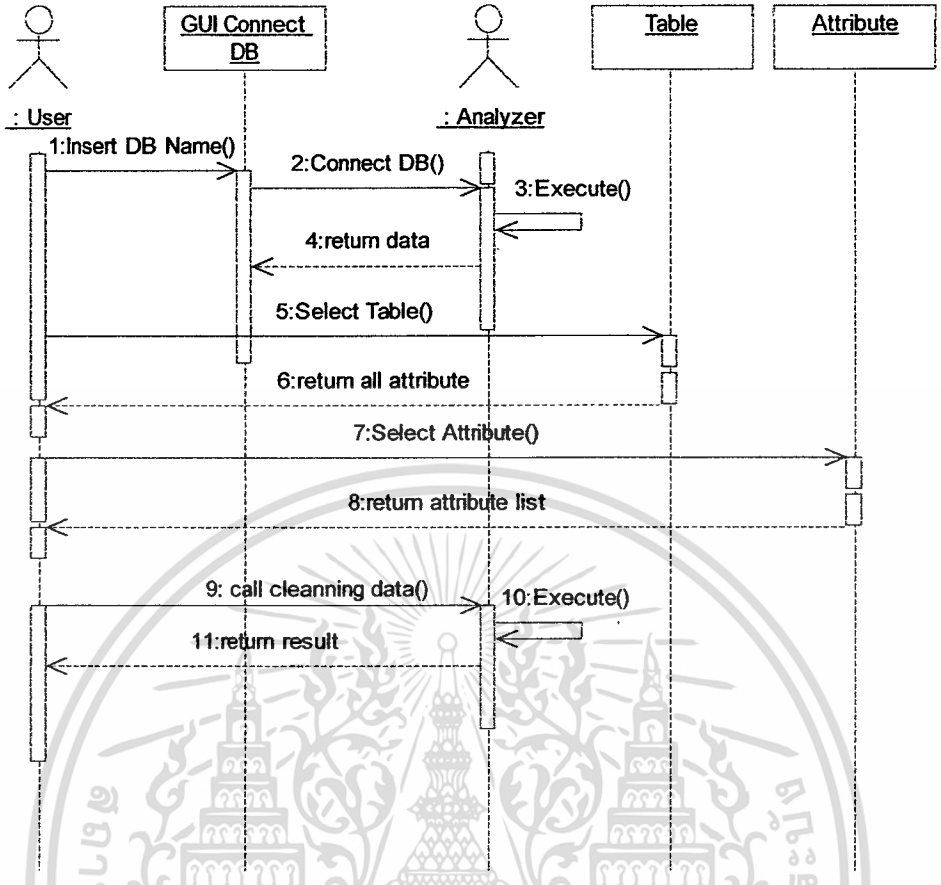
ในส่วนของซีเควนซ์ไดอะแกรม จะถูกสร้างขึ้นหลังจากได้ทำการออกแบบยูสเคสไดอะแกรมของระบบแล้ว เพื่อเป็นแผนภาพแสดงถึงลำดับขั้นตอนการทำงานของระบบซึ่งเป็นไปตามลำดับของการเกิดเหตุการณ์ (Scenario) เพื่ออธิบายความสัมพันธ์ของอ็อบเจกต์ เมื่อมีการส่งข้อความตามเหตุการณ์ที่เกิดขึ้นระหว่างอ็อบเจกต์ โดยซีเควนซ์ไดอะแกรม นี้จะประกอบด้วย

- เส้นในแนวตั้ง แสดงถึงอ็อบเจกต์ โดยจะมีชื่อของแต่ละอ็อบเจกต์อยู่ด้านบนของเส้น
- เส้นในแนวนอน แสดงถึงข้อความที่ส่งผ่านกันระหว่าง อ็อบเจกต์

โดยในส่วนนี้จะนำมาใช้อธิบายขั้นตอนการส่งข้อความถึงกันระหว่างอ็อบเจกต์ในการทำงานของการสร้างโมเดลและแยกแยะประเภทข้อมูลแสดงดังภาพที่ 5.7 – 5.10

Sequence Diagram ของการทำความสะอาดข้อมูลก่อนสร้างโมเดล

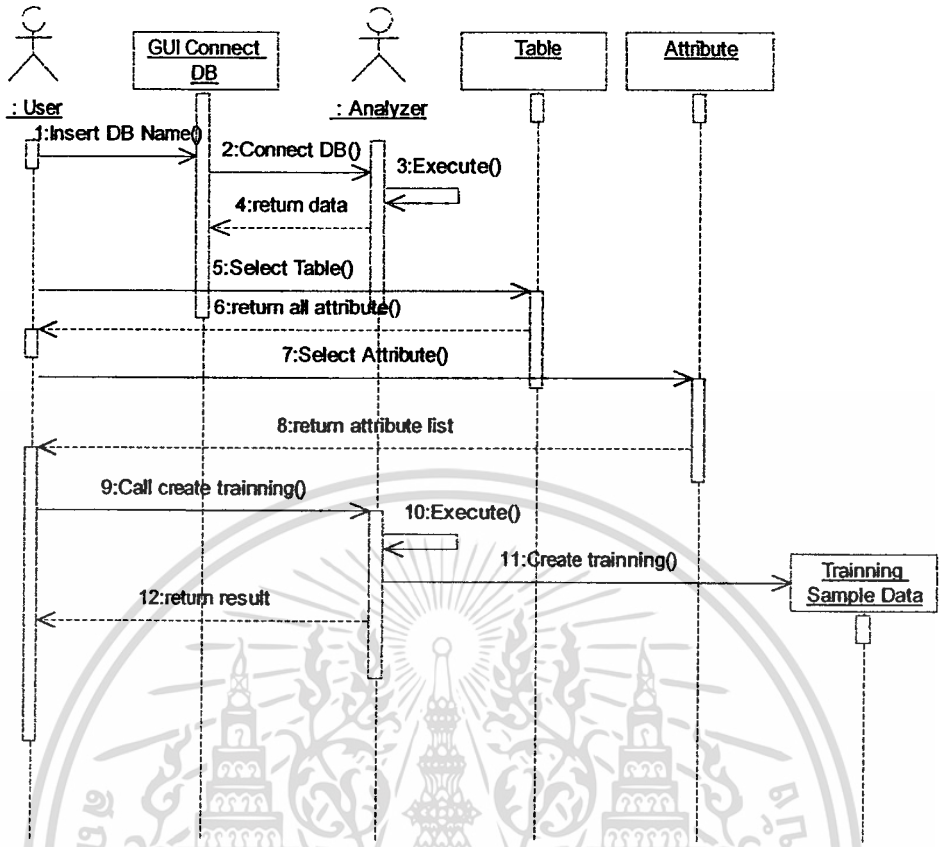
- ผู้ใช้ป้อนชื่อฐานข้อมูลที่ต้องการจะติดต่อ
- ระบบจะทำการติดต่อฐานข้อมูลที่ผู้ใช้ต้องการและระบบจะแสดงดึงข้อมูลของฐานข้อมูลนั้นมาแสดงที่หน้าจอ
- ผู้ใช้เลือกตารางที่ต้องการนำมาสร้างข้อมูลสอนระบบ
- ระบบจะแสดงแอตทริบิวต์ที่มีทั้งหมดของตารางนั้น และแสดงค่าของข้อมูลของแอตทริบิวต์ที่เลือกดังกล่าวว่ามีค่าที่เป็นค่าว่างหรือไม่
- และระบบจะให้ผู้ใช้สามารถเลือกการทำความสะอาดได้ 2 แบบคือการลบเรคคอร์ทที่มีค่าว่างหรือแทนที่ข้อมูลด้วยค่าที่มีจำนวนเรคคอร์ทมากสุดกรณีที่แอตทริบิวต์นั้นมีค่าเป็นตัวอักษรหรือแทนที่ข้อมูลด้วยค่าเฉลี่ยของข้อมูลเหล่านั้นกรณีที่แอตทริบิวต์นั้นมีค่าเป็นตัวเลข
- และระบบจะทำการแสดงผลของการทำความสะอาดที่หน้าจอใหม่อีกครั้ง



ภาพที่ 5.7 ซีเควนซ์ไดอะแกรมของการทำความสะอาดข้อมูล

Sequence Diagram ของการสร้างข้อมูลสอนระบบ

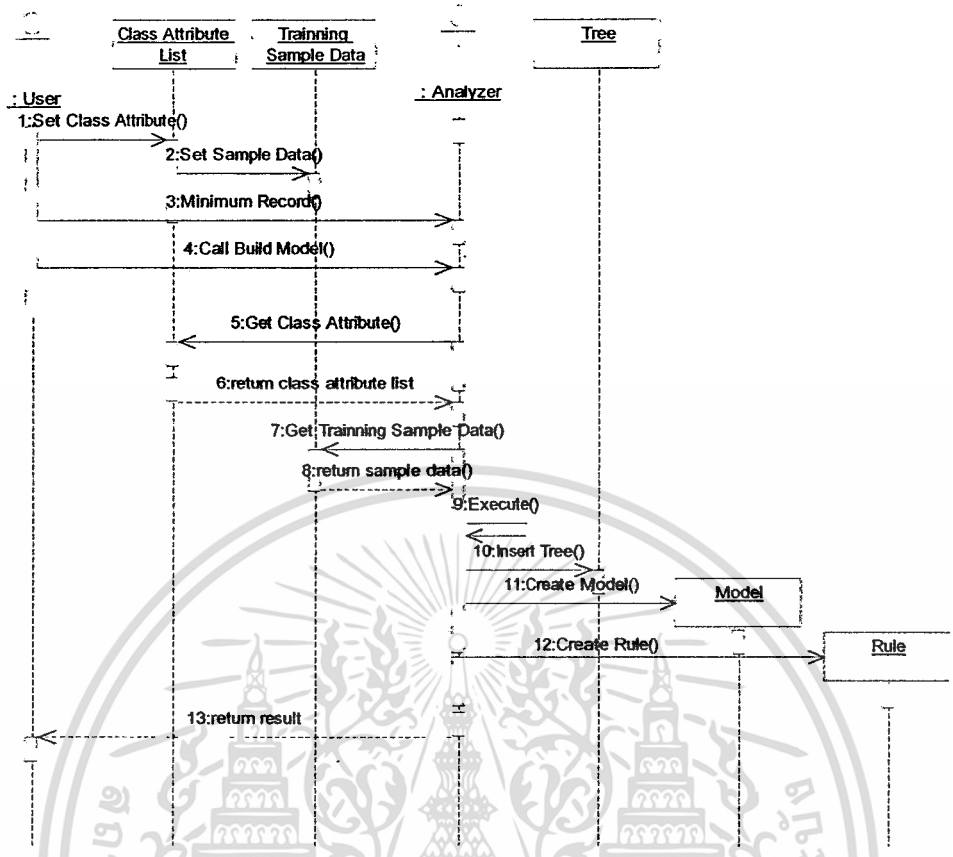
- ผู้ใช้ป้อนชื่อฐานข้อมูลที่ต้องการจะติดต่อ
- ระบบจะทำการติดต่อฐานข้อมูลที่ผู้ใช้ต้องการและระบบจะแสดงดึงข้อมูลของฐานข้อมูลนั้นมาแสดงที่หน้าจอ
- ผู้ใช้ป้อนชื่อข้อมูลสอนระบบที่ต้องการจะสร้าง
- ผู้ใช้เลือกตารางและแอตทริบิวต์ของตารางที่เลือก
- จากนั้นระบบจะให้ผู้ใช้ทำการกดปุ่ม ตกลงเพื่อทำการส่งแอตทริบิวต์ทั้งหมดที่เลือกให้ระบบทำการประมวลผลต่อไป
- เมื่อระบบประมวลผลเสร็จสิ้นก็จะได้ข้อมูลสอนระบบที่ต้องการ



ภาพที่ 5.8 ซีเควนซ์ไดอะแกรมของการสร้างข้อมูลสอนระบบ

Sequence Diagram ของการสร้างแบบจำลอง

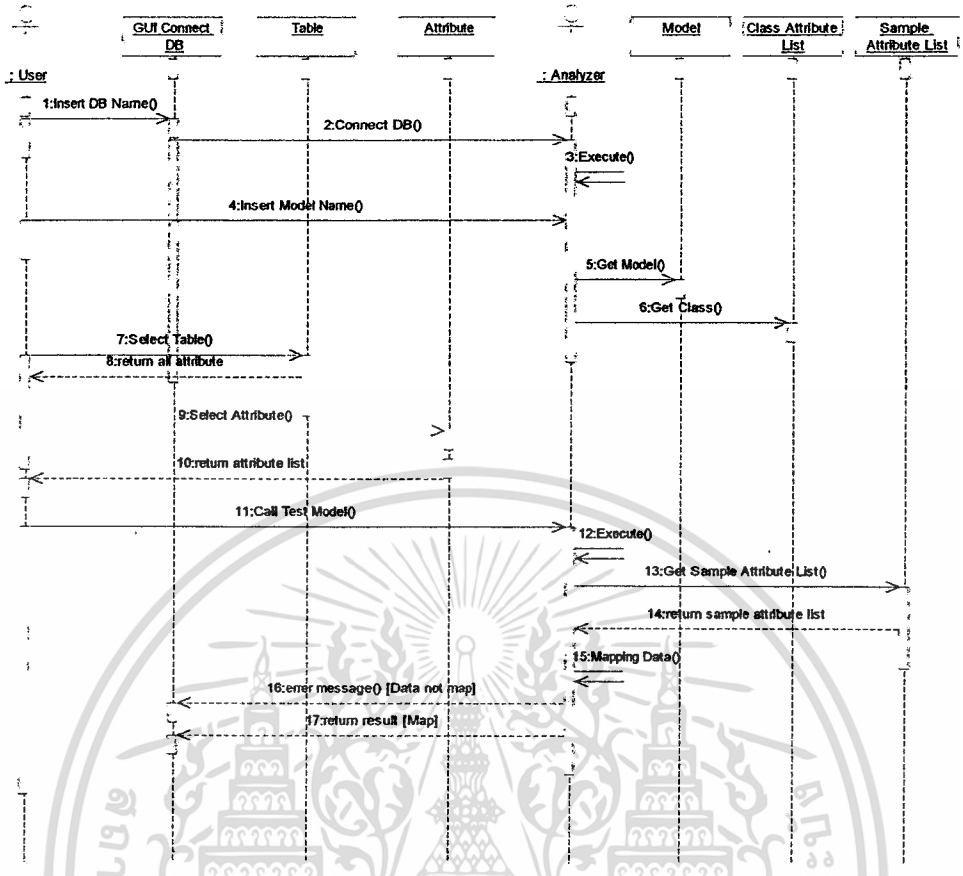
- ผู้ใช้ป้อนชื่อโมเดลที่ต้องการจะสร้าง
- ผู้ใช้ป้อนคลาสที่ต้องการจะนำไปวิเคราะห์เพื่อแบ่งโครงสร้างต้นไม้
- ผู้ใช้ป้อนข้อมูลสอนระบบที่นำไปวิเคราะห์เพื่อสร้างโมเดล
- ผู้ใช้ทำการป้อนข้อมูลจำนวนเรคคอร์ดที่น้อยที่สุดที่จะสามารถแบ่งโครงสร้างต้นไม้ได้
- ผู้ใช้ทำการกดปุ่ม ตกลง เพื่อให้ระบบทำการประมวลผล
- จากนั้นระบบจะทำการดึงข้อมูลคลาส ดึงข้อมูลสอนระบบ และทำการประมวลผล
- และสุดท้ายระบบจะทำการสร้างโครงสร้างต้นไม้และทำการสร้างกฎของโมเดลนั้น
- ระบบจะแสดงผลลัพธ์ที่ได้ให้ผู้ใช้ที่หน้าจอ



ภาพที่ 5.9 ซีเควนซ์ไดอะแกรมของการสร้างแบบจำลอง

Sequence Diagram ของการทดสอบโมเดล

- ผู้ใช้ป้อนชื่อโมเดลที่ต้องการนำมาทดสอบ
- ระบบจะทำการดึงข้อมูลโมเดล ดึงคลาสของโมเดล
- ผู้ใช้ป้อนข้อมูลที่ต้องการนำมาทดสอบโมเดลซึ่งจะเป็นข้อมูลคนละชุดกับข้อมูลที่นำมาสร้างข้อมูลสอนระบบ
- ระบบจะทำการดึงคุณสมบัติของข้อมูลสอนระบบมาทำการแม็พกับข้อมูลที่ผู้ใช้ป้อนเข้ามา หากเป็นข้อมูลที่เหมือนกันระบบก็จะทำการประมวลผลต่อไป
- ระบบจะทำการประมวลผลและแสดงผลลัพธ์ที่ได้จากการประมวลผลที่หน้าจอ



ภาพที่ 5.10 ซีเควนซ์ไดอะแกรมของการทดสอบโมเดล

5.5 คลาสไดอะแกรม

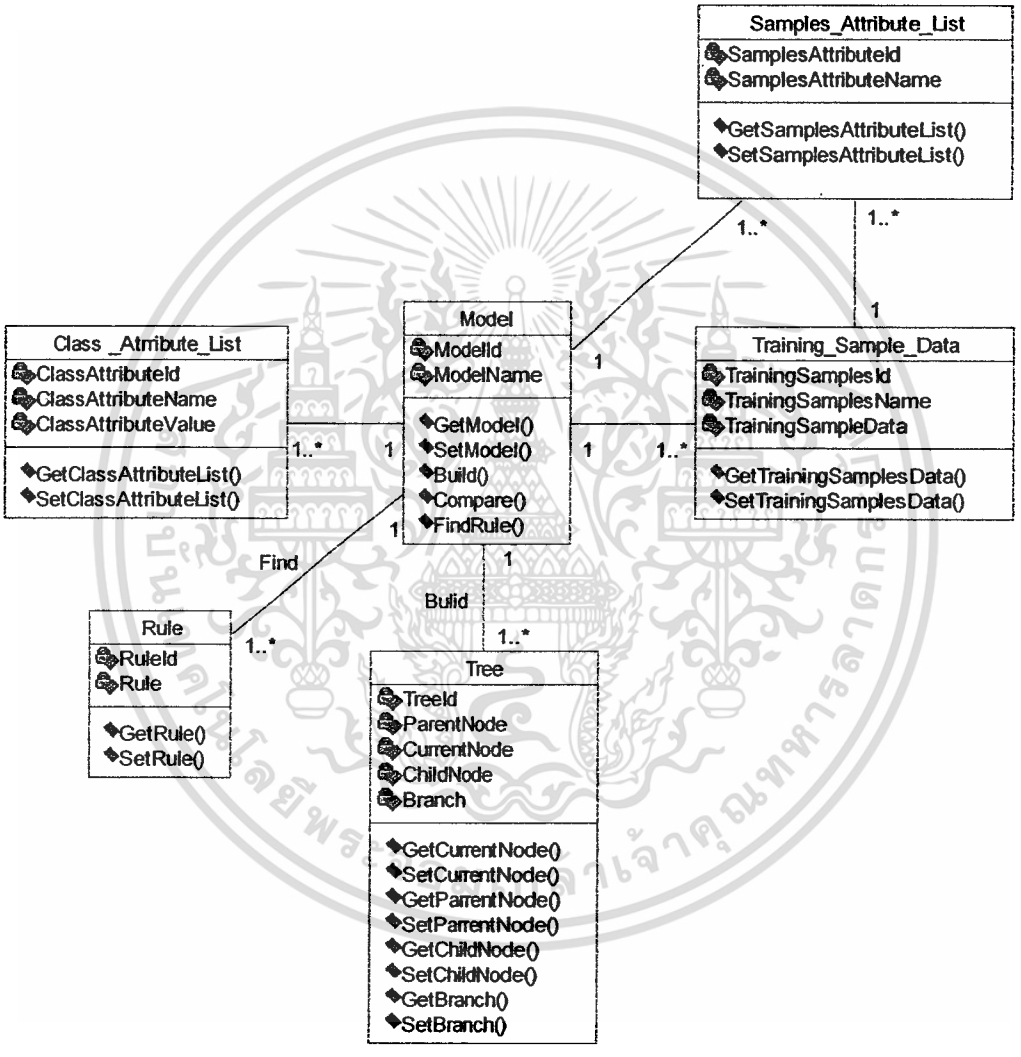
คลาสไดอะแกรมเป็นคลาสเป็นองค์ประกอบที่สำคัญสำหรับระบบงานเชิงวัตถุ โดยคลาสเป็นการนำเอากลุ่มของออบเจ็กต์มาอธิบายความหมาย ออบเจ็กต์ซึ่งถูกจัดให้อยู่ในคลาสเดียวกันจะมี แอตทริบิวต์ โอเปอร์เรชัน รีเลชัน และความหมายบางอย่างเหมือนกัน (สุนทริน วงศ์ศิริกุล. 2543 : 62)

จากขั้นตอนการวิเคราะห์และออกแบบระบบที่ผ่านมาทำให้ได้คลาสต่างๆที่จำเป็น มีดังนี้

1. คลาส Class_Attribute_List เป็นคลาสของคลาสที่ต้องการจำแนก
2. คลาส Training_Samples_Data เป็นคลาสของข้อมูลสอนระบบ
3. คลาส Samples_Attribute_List เป็นคลาสแอทริบิวต์ของข้อมูลสอนระบบ
4. คลาส Tree เป็นคลาสโครงสร้างต้นไม้
5. คลาส Rule เป็นคลาสกฎที่ได้
6. คลาส Model เป็นคลาสข้อมูลโมเดลที่สร้าง

โดยแต่ละคลาสจะประกอบด้วยแอตทริบิวต์และเมธอดต่างๆที่สามารถทำงานกับระบบได้ ซึ่งแต่ละอ็อบเจกต์ที่ถูกสร้างขึ้นมาจากคลาสเหล่านี้ก็จะได้รับแอตทริบิวต์และเมธอดทั้งหมดของคลาสมาด้วย

จากคลาสที่กล่าวถึงข้างต้นนี้สามารถนำมาสร้างเป็นคลาสไดอะแกรม ซึ่งแสดงความสัมพันธ์ระหว่างคลาสต่างๆได้ดังภาพที่ 5.11

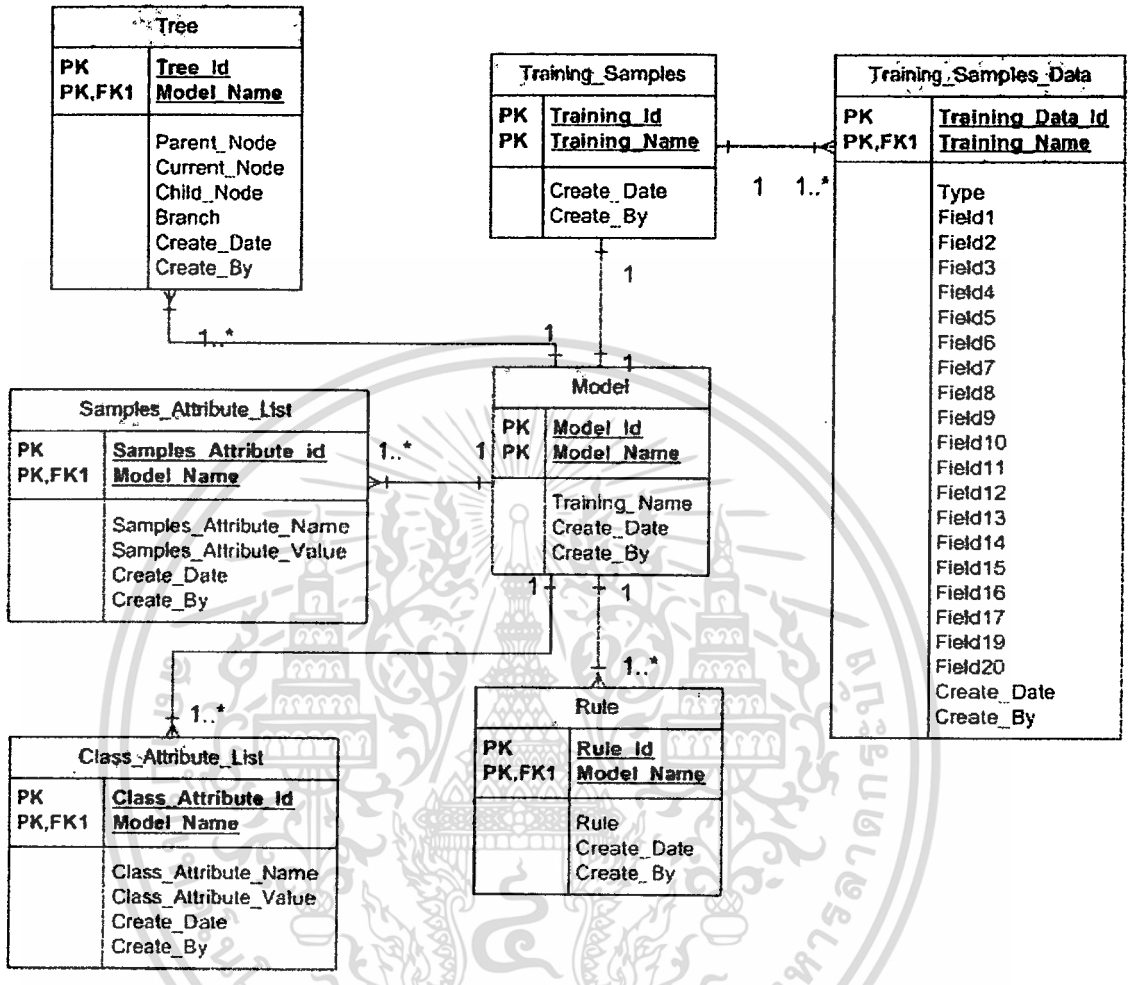


ภาพที่ 5.11 คลาสไดอะแกรมของการระบบ

5.6 การออกแบบระบบงานโดยการจำลองแบบข้อมูล

ในหัวข้อนี้จะอธิบายการออกแบบระบบงานเกี่ยวกับกลุ่มของข้อมูลที่สัมพันธ์กัน ด้วยแบบจำลองข้อมูล สำหรับเครื่องมือที่จะนำมาใช้ในการวิเคราะห์คือแผนภาพแสดงความสัมพันธ์

ระหว่างเอนทิตี (Entity Relationship Diagram) โดยระบบงานจะมีแผนภาพแสดงความสัมพันธ์ระหว่างเอนทิตี ดังภาพที่ 5.12



ภาพที่ 5.12 แบบจำลองความสัมพันธ์ระหว่างเอนทิตี (แผนภาพอีอาร์)

รายละเอียดของเอนทิตีทั้งหมดมีดังตารางที่ 5.6

ตารางที่ 5.6 รายละเอียดของแต่ละเอนทิตี

Table Name	Table Description
Model	เก็บข้อมูลเกี่ยวชื่อโมเดลที่สร้าง
Traning_Samples	เก็บข้อมูลเกี่ยวกับชื่อของข้อมูลสอนระบบที่ใช้ (Training Data)
Training_Samples_Data	เก็บข้อมูลของข้อมูลสอนระบบทั้งหมด
Class_Attribute_List	เก็บข้อมูลของคลาสแอทริบิวต์ทั้งหมด
Samples_Attribute_List	เก็บข้อมูลเกี่ยวกับแอทริบิวต์ของข้อมูลสอนระบบ
Tree	เก็บข้อมูลของทรีที่ได้จากการสร้างโมเดลผ่านข้อมูลสอนระบบที่เลือก
Rule	เก็บข้อมูลของกฎที่ได้จากการสร้างโมเดลผ่านข้อมูลสอนระบบที่เลือก

พจนานุกรมข้อมูล

หลังจากที่ได้ทำการวิเคราะห์และออกแบบฐานข้อมูลโดยวิธี Data Modeling แล้ว สามารถกำหนดคุณลักษณะของแอททริบิวต์ในแต่ละเอนทิตีได้ดังตารางที่ 5.7 – 5.13 โดยข้อความในคอลัมน์ Key มีความหมายดังนี้

PK หมายถึง คีย์หลักของตาราง(Primary Key)

FK หมายถึง คีย์นอกของตาราง(Foreign Key)

1. เอนทิตี Model

เก็บรายละเอียดของชื่อโมเดลที่ทำการสร้าง รายละเอียดดังตารางที่ 5.7

ตารางที่ 5.7 ดาต้าดิกชันนารีของเอนทิตี Model

Attribute	Type	Detail	Key	Format
Model_Id	Number	รหัสโมเดล	PK	9999
Model_Name	Varchar (50)	ชื่อโมเดล	PK	xxxxxxxxx
Training _Name	Varchar (50)	ชื่อข้อมูลสอนระบบ	FK	xxxxxxxxx
Create_date	Date	วันที่สร้างข้อมูล		dd/mm/yyyy hh:ss
Create_By	Varchar (50)	สร้างโดย		xxxxxxxxx

2. เอนทิตี Traing_Samples

เก็บชื่อของชื่อของข้อมูลสอนระบบที่ใช้ รายละเอียดดังตารางที่ 5.8

ตารางที่ 5.8 คาต้าดิกชันนารีของเอนทิตี Training_Samples

Attribute	Type	Detail	Key	Format
Training_Id	Number	รหัสข้อมูลสอนระบบ	PK	9999
Training _Name	Varchar (50)	ชื่อข้อมูลสอนระบบ	PK	xxxxxxxxx
Create_date	Date	วันที่สร้างข้อมูล		dd/mm/yyyy hh:ss
Create_By	Varchar (50)	สร้างโดย		xxxxxxxxx

3. เอนทิตี Training_Samples_Data

เก็บข้อมูลของข้อมูลสอนระบบที่ใช้ รายละเอียดดังตารางที่ 5.9

ตารางที่ 5.9 คาต้าดิกชันนารีของเอนทิตี Training_Samples_Data

Attribute	Type	Detail	Key	Format
Training_Data_Id	Number	รหัส	PK	9999
Training_Name	Varchar (50)	ชื่อข้อมูลสอนระบบ	PK,FK	xxxxxxxxx
Type	Varchar (50)	ประเภทของข้อมูล		xxxxxxxxx
Field1	Varchar (50)	ข้อมูลแอทริบิวต์ที่1		xxxxxxxxx
Field2	Varchar (50)	ข้อมูลแอทริบิวต์ที่2		xxxxxxxxx
Field3	Varchar (50)	ข้อมูลแอทริบิวต์ที่3		xxxxxxxxx
Field4	Varchar (50)	ข้อมูลแอทริบิวต์ที่4		xxxxxxxxx
Field5	Varchar (50)	ข้อมูลแอทริบิวต์ที่5		xxxxxxxxx
Field6	Varchar (50)	ข้อมูลแอทริบิวต์ที่6		xxxxxxxxx
Field7	Varchar (50)	ข้อมูลแอทริบิวต์ที่7		xxxxxxxxx
Field8	Varchar (50)	ข้อมูลแอทริบิวต์ที่8		xxxxxxxxx
Field9	Varchar (50)	ข้อมูลแอทริบิวต์ที่9		xxxxxxxxx
Field10	Varchar (50)	ข้อมูลแอทริบิวต์ที่10		xxxxxxxxx
Field11	Varchar (50)	ข้อมูลแอทริบิวต์ที่11		xxxxxxxxx
Field12	Varchar (50)	ข้อมูลแอทริบิวต์ที่12		xxxxxxxxx
Field13	Varchar (50)	ข้อมูลแอทริบิวต์ที่13		xxxxxxxxx

ตารางที่ 5.9 (ต่อ)

Attribute	Type	Detail	Key	Format
Field14	Varchar (50)	ข้อมูลแอทริบิวต์ที่14		xxxxxxxxx
Field15	Varchar (50)	ข้อมูลแอทริบิวต์ที่15		xxxxxxxxx
Field16	Varchar (50)	ข้อมูลแอทริบิวต์ที่16		xxxxxxxxx
Field17	Varchar (50)	ข้อมูลแอทริบิวต์ที่17		xxxxxxxxx
Field18	Varchar (50)	ข้อมูลแอทริบิวต์ที่18		xxxxxxxxx
Field19	Varchar (50)	ข้อมูลแอทริบิวต์ที่19		xxxxxxxxx
Field20	Varchar (50)	ข้อมูลแอทริบิวต์ที่20		xxxxxxxxx
Create_date	Date	วันที่สร้างข้อมูล		dd/mm/yyyy hh:ss
Create_By	Varchar (50)	สร้างโดย		xxxxxxxxx

4. เอนทิตี Class_Attribute_List

เก็บข้อมูลของคลาสแอทริบิวต์ รายละเอียดดังตารางที่ 5.10

ตารางที่ 5.10 คาดิกชันนารีของเอนทิตี Class_Attribute_List

Attribute	Type	Detail	Key	Format
Class_Attribute_Id	Number	รหัสคลาส	PK	9999
Model_Name	Varchar (50)	ชื่อโมเดล	PK	xxxxxxxxx
Class_Attribute_Name	Varchar (50)	ชื่อคลาส	PK	xxxxxxxxx
Class_Attribute_Value	Varchar (50)	ค่าข้อมูลในคลาส		xxxxxxxxx
Create_date	Date	วันที่สร้างข้อมูล		dd/mm/yyyy hh:ss
Create_By	Varchar (50)	สร้างโดย		xxxxxxxxx

5. เอนทิตี Samples_Attribute_List

เก็บข้อมูลของแอทริบิวต์ของข้อมูลตัวอย่าง รายละเอียดดังตารางที่ 5.11

ตารางที่ 5.11 คาต้าดิกชันนารีของเอนทิตี Samples_Attribute_List

Attribute	Type	Detail	Key	Format
Samples_Attribute_Id	Number	รหัสข้อมูลตัวอย่าง	PK	9999
Model_Name	Varchar (50)	ชื่อโมเดล	PK	xxxxxxxxx
Samples_Attribute_Name	Varchar (50)	ชื่อข้อมูลตัวอย่าง		xxxxxxxxx
Samples_Attribute_Value	Varchar (50)	ค่าข้อมูลตัวอย่าง		xxxxxxxxx
Create_date	Date	วันที่สร้างข้อมูล		dd/mm/yyyy hh:ss
Create_By	Varchar (50)	สร้างโดย		xxxxxxxxx

6. เอนทิตี Tree

เก็บข้อมูลของทรีที่ได้จากการสร้างโมเดลผ่านข้อมูลสอนระบบ รายละเอียดดังตารางที่ 5.12

ตารางที่ 5.12 คาต้าดิกชันนารีของเอนทิตี Tree

Attribute	Type	Detail	Key	Format
Tree_Id	Number	รหัสทรี	PK	9999
Model_Name	Varchar (50)	ชื่อโมเดล	PK	xxxxxxxxx
Parent_Node	Varchar (50)	โหนดแม่		xxxxxxxxx
Current_Node	Varchar (50)	ชื่อโหนดปัจจุบัน		xxxxxxxxx
Child_Node	Varchar (50)	โหนดลูก		xxxxxxxxx
Branch	Varchar (50)	ค่าของสาขา		xxxxxxxxx
Create_date	Date	วันที่สร้างข้อมูล		dd/mm/yyyy hh:ss
Create_By	Varchar (50)	สร้างโดย		xxxxxxxxx

7. เอนทิตี Rule

เก็บข้อมูลของกฎที่ได้จากการสร้างโมเดลผ่านข้อมูลสอนระบบ รายละเอียดดังตารางที่ 5.13

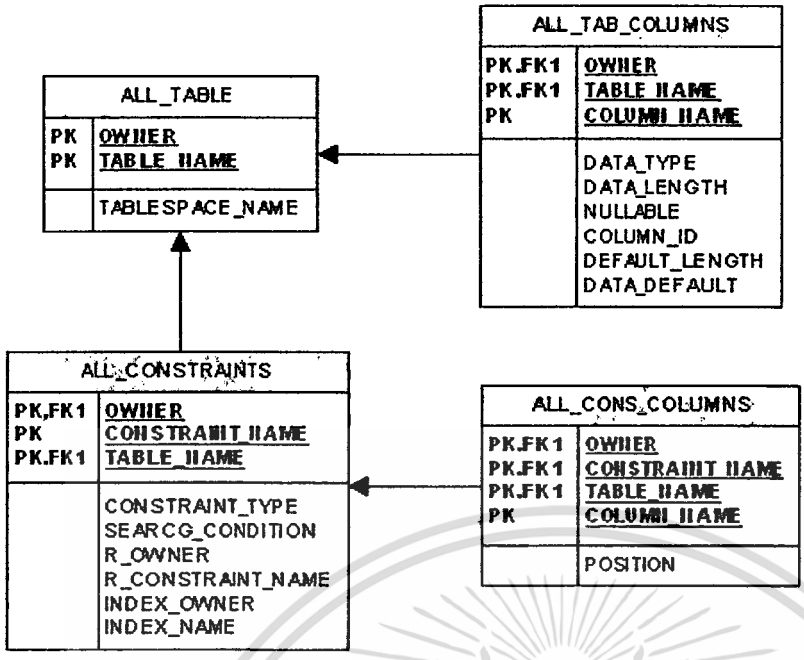
ตารางที่ 5.13 คาต้าดิกชันนารีของเอนทิตี Rule

Attribute	Type	Detail	Key	Format
Rule_Id	Number	รหัสกฎ	PK	9999
Model_Name	Varchar (50)	ชื่อโมเดล	PK	xxxxxxxxx
Rule_Value	Varchar (250)	กฎ		xxxxxxxxx
Create_date	Date	วันที่สร้างข้อมูล		dd/mm/yyyy hh:ss
Create_By	Varchar (50)	สร้างโดย		xxxxxxxxx

ตารางข้อมูลของฐานข้อมูลออราเคิลที่ใช้ในการศึกษา

สำหรับฐานข้อมูลออราเคิลนั้นในส่วนที่เป็นข้อมูลเกี่ยวกับข้อจำกัดของแต่ละตารางข้อมูลนั้น ฐานข้อมูลออราเคิลจะไม่อนุญาตให้ผู้ใช้งานเข้าถึงตารางข้อมูลเกี่ยวข้องกับตารางข้อมูลข้อจำกัดโดยตรง แต่จะอนุญาตให้ผู้ใช้อ่านข้อมูลผ่านมุมมองของตารางข้อมูล (View) โดยรายละเอียดของตารางข้อมูล และมุมมองของตารางข้อมูลที่นำมาช่วยในการพัฒนาระบบ มีดังนี้

1. ตารางข้อมูลของตารางทั้งหมดภายในฐานข้อมูล (ALL_TABLE) เป็นตารางข้อมูลที่เก็บตารางข้อมูลทั้งหมดของฐานข้อมูลออราเคิล ซึ่งระบบที่ทำการพัฒนานำมาใช้ในส่วนที่ให้ผู้ใช้เลือกในการที่จะทำการสร้างโมเดล
2. ตารางข้อมูลรายละเอียดของตารางข้อมูล (ALL_TAB_COLUMNS) เป็นตารางข้อมูลที่เก็บรายละเอียด เช่น คอลัมน์ ประเภทข้อมูลของตารางข้อมูล ซึ่งระบบที่ทำการพัฒนาใช้ตารางข้อมูลนี้ช่วยในการแสดงข้อมูล และดึงประเภทข้อมูล
3. มุมมองตารางข้อมูลข้อจำกัด (ALL_CONSTRAINTS) เป็นมุมมองตารางข้อมูลที่ใช้ในการเก็บข้อจำกัดต่างๆของตารางข้อมูลทั้งหมดในระบบฐานข้อมูลของออราเคิล
4. มุมมองตารางข้อมูลรายละเอียดข้อจำกัด (ALL_CONS_COLUMNS) เป็นมุมมองตารางข้อมูลที่ใช้ในการเก็บรายละเอียดข้อจำกัด ว่าข้อจำกัดนี้ประกอบไปด้วยข้อมูลคอลัมน์ใด และมี key ของตารางมีอะไรบ้าง



ภาพที่ 5.13 แบบจำลองความสัมพันธ์ระหว่างเอนทิตีของตารางข้อมูลต่างๆในฐานข้อมูล

1. เอนทิตี ALL_TABLE

เก็บรายละเอียดของตารางทั้งหมดภายในฐานข้อมูล รายละเอียดดังตารางที่ 5.14

ตารางที่ 5.14 คาต้าดิกชันนารีของเอนทิตี ALL_TABLE

Attribute	Type	Detail	Key	Format
Owner	Varchar (50)	เจ้าของตาราง	PK	xxxxxxxxx
Table_Name	Varchar (50)	ชื่อตาราง	PK	xxxxxxxxx
Tablespace_Name	Varchar (50)	ชื่อ Tablespace		xxxxxxxxx

2. เอนทิตี ALL_TAB_COLUMNS

เก็บรายละเอียดของตารางข้อมูล รายละเอียดดังตารางที่ 5.15

ตารางที่ 5.15 คาต้าดิกชันนารีของเอนทิตี ALL_TAB_COLUMNS

Attribute	Type	Detail	Key	Format
Owner	Varchar (50)	เจ้าของตาราง	PK	xxxxxxxxx
Table_Name	Varchar (50)	ชื่อตาราง	PK	xxxxxxxxx
Column_Name	Varchar (50)	ชื่อคอลัมน์	PK	xxxxxxxxx
Data_Type	Varchar (250)	ประเภทข้อมูล		xxxxxxxxx

ตารางที่ 5.15 (ต่อ)

Attribute	Type	Detail	Key	Format
Data_Length	Number	ความยาวของข้อมูล		
Nullable	Varchar(1)	สามารถมีค่าเป็น Null		
Column_Id	Number	รหัสคอลัมน์		
Default_Length	Number	ความยาวเริ่มต้น		
Data_Default	Long	ข้อมูลเริ่มต้น		

3. เอนทิตี ALL_CONSTRAINTS

เก็บข้อมูลข้อจำกัดของตาราง รายละเอียดดังตารางที่ 5.16

ตารางที่ 5.16 คาต้าดิกชันนารีของเอนทิตี ALL_CONSTRAINTS

Attribute	Type	Detail	Key	Format
Owner	Varchar (50)	เจ้าของตาราง	PK	xxxxxxxxx
Constraint_Name	Varchar (50)	ชื่อข้อจำกัด	PK	xxxxxxxxx
Constraint_Type	Varchar (1)	ประเภทข้อจำกัด		P-Primary F-Foreign C-Check U-Unique
Table_Name	Varchar (250)	ชื่อตารางข้อมูล	PK	xxxxxxxxx
Search_Condition	Long	เงื่อนไขในการตรวจสอบ		
R_Owner	Varchar(50)	เจ้าของตารางที่มีความสัมพันธ์		
R_Constraint_Name	Varchar(50)	ชื่อข้อจำกัดที่มีความสัมพันธ์		
Index_Owner	Varchar(50)	ชื่อเจ้าของอินเด็กซ์		
Index_Name	Varchar(50)	ชื่ออินเด็กซ์		

4. เอนทิตี ALL_CONS_COLUMNS

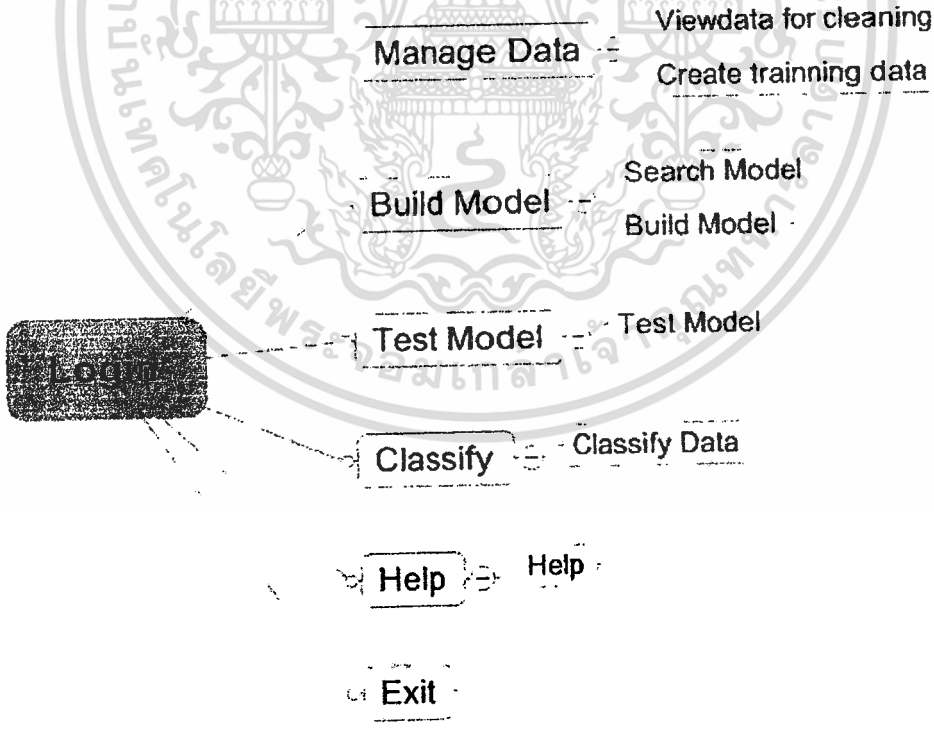
เก็บรายละเอียดของข้อมูลข้อจำกัดของตาราง รายละเอียดดังตารางที่ 5.17

ตารางที่ 5.17 ดาต้าดิกชันนารีของเอนทิตี ALL_CONS_COLUMNS

Attribute	Type	Detail	Key	Format
Owner	Varchar (50)	เจ้าของตาราง	PK	xxxxxxxxx
Constraint_Name	Varchar (50)	ชื่อข้อจำกัด	PK	xxxxxxxxx
Table_Name	Varchar (50)	ชื่อตารางข้อมูล	PK	xxxxxxxxx
Column_Name	Varchar (250)	ชื่อคอลัมน์	PK	xxxxxxxxx
Position	Number	ตำแหน่ง		

5.7 โครงสร้างโปรแกรมและส่วนประกอบต่างๆ ของโปรแกรม

ในส่วนแรกของการออกแบบส่วนติดต่อกับผู้ใช้งาน จะออกแบบแผนที่เว็บไซต์ เพื่อแสดงโครงสร้างระบบเนวิเกชั่นของเว็บไซต์ว่าประกอบด้วยเมนูหลัก เมนูย่อยอะไรบ้าง โดยแผนที่เว็บไซต์ของระบบ แสดงไว้ดังภาพที่ 5.14



ภาพที่ 5.14 แผนที่เว็บไซต์ของระบบ

บทที่ 6
การสร้างและทดสอบโปรแกรม

ในการประยุกต์ใช้กับกรณีศึกษาในโครงงานนี้ได้ทำการคัดเลือกข้อมูลจากฐานข้อมูลการสั่งซื้อสินค้า ซึ่งเป็นการนำข้อมูลการซื้อสินค้าของลูกค้ามาเป็นข้อมูลตัวอย่างในการศึกษาซึ่งหากเจ้าของบริษัทต้องการทราบปัจจัยที่มีผลต่อการสั่งซื้อสินค้าของลูกค้าแล้ว จะทำให้สามารถวางแผนทางการตลาดได้ถูกต้อง

6.1 เทคนิคของดาต้าไมนิ่ง

จากปัญหาที่เกิดขึ้นและทรัพยากรที่ทางบริษัทมีสามารถที่จะทำการวิเคราะห์ข้อมูลที่ผู้บริหารต้องการได้โดยใช้เทคนิคของดาต้าไมน์นิ่งแบบ Decision Trees Model มาช่วยในการวิเคราะห์ข้อมูลที่มีอยู่ โดยการคัดเลือกข้อมูลจากฐานข้อมูลมาทำการสร้าง Traning Data เพื่อนำไปใช้ในการสร้างแบบจำลอง

6.2 การทำงานของโปรแกรม

เป็นส่วนที่ใช้ประมวลผลในการวิเคราะห์รายงาน ซึ่งจะแยกแต่ละเงื่อนไขของการวิเคราะห์ในแต่ละหน้าจอ แต่เนื่องจากมีการออกแบบหน้าจอที่มีความสอดคล้องกัน โดยมีลักษณะโครงสร้างที่เหมือนกัน ดังนั้นหน้าจอจึงคล้ายกัน และแตกต่างกันบางจุดเท่านั้นเช่น ปุ่มคำสั่ง ข้อมูลในตารางมาจากแหล่งที่ต่างกัน โดยมีรายละเอียดดังนี้

- การประยุกต์ใช้งานกับกรณีศึกษา

จะนำข้อมูลการซื้อสินค้าของลูกค้ามาเป็นข้อมูลตัวอย่างในการศึกษาซึ่งหากเจ้าของบริษัทต้องการทราบว่ามีปัจจัยใดบ้างที่มีผลต่อการสั่งซื้อสินค้า ,ซื้อสินค้าแต่ละประเภทของลูกค้าและประเภทการใช้จ่าย

ตารางที่ 6.1 รายละเอียดของแต่ละเอนทิตี

Table Name	Table Description
Customer	เก็บข้อมูลลูกค้าและการซื้อสินค้า

➢ พจนานุกรมข้อมูล

ข้อมูลของการซื้อสินค้าของลูกค้าจะกำหนดคุณลักษณะของแอททริบิวต์ได้ดังตารางที่ 6.2 โดยข้อความในคอลัมน์ Key มีความหมายดังนี้

PK หมายถึง คีย์หลักของตาราง(Primary Key)

เก็บรายละเอียดของการซื้อสินค้าของลูกค้า รายละเอียดดังตารางที่ 6.2

ตารางที่ 6.2 คาต้าดิกชันนารีของเอนทิตี Customer

Attribute	Type	Detail	Key	Format
Customer_Id	Number	รหัสลูกค้า	PK	9999
Customer_Name	Varchar (250)	ชื่อลูกค้า		xxxxxxxxx
Age	Number	อายุ		999
Education	Varchar(250)	การศึกษา		xxxxxxxxx
Sex	Varchar(1)	เพศ		F/M
Marital_Status	Varchar(20)	สถานะสมรส		xxxxxxxxx
Have_Child	Varchar(20)	จำนวนบุตร		Yes/No
Job	Varchar(250)	อาชีพ		xxxxxxxxx
Income	Number	รายได้		99999999999
Experience	Number	ประสบการณ์ทำงาน		99
Province	Varchar(250)	จังหวัด		xxxxxxxxx
Product_Type	Varchar(250)	ประเภทสินค้า		xxxxxxxxx
Purchase_Type	Varchar(250)	ประเภทการซื้อ		xxxxxxxxx

ตารางที่ 6.3 รายการการศึกษา

ค่าที่เป็นไปได้	ความหมาย
PRIMARY OR LOWER	มัธยมศึกษาหรือต่ำกว่า
DIPLOMA	ปวส./อนุปริญญา
BACHELOR	ปริญญาตรี
MASTER	ปริญญาโท
SEMASTER	ปริญญาเอก

ตารางที่ 6.4 รายการสถานะสมรส

ค่าที่เป็นไปได้	ความหมาย
SINGLE	โสด
MARRIED	สมรส
DIVORCED	หย่าร้าง
WIDOW	หม้าย

ตารางที่ 6.5 รายการอาชีพ

ค่าที่เป็นไปได้	ความหมาย
BUSINESS	ธุระกิจส่วนตัว
GOVERNMENT	ข้าราชการ
OFFICER	พนักงานบริษัท/ลูกจ้าง
STATE	พนักงานรัฐวิสาหกิจ

ตารางที่ 6.6 รายการที่อยู่

ค่าที่เป็นไปได้	ความหมาย
BANGKOK	กรุงเทพฯและปริมณฑล
OTHERS	ต่างจังหวัด

ตารางที่ 6.7 รายการประเภทสินค้า

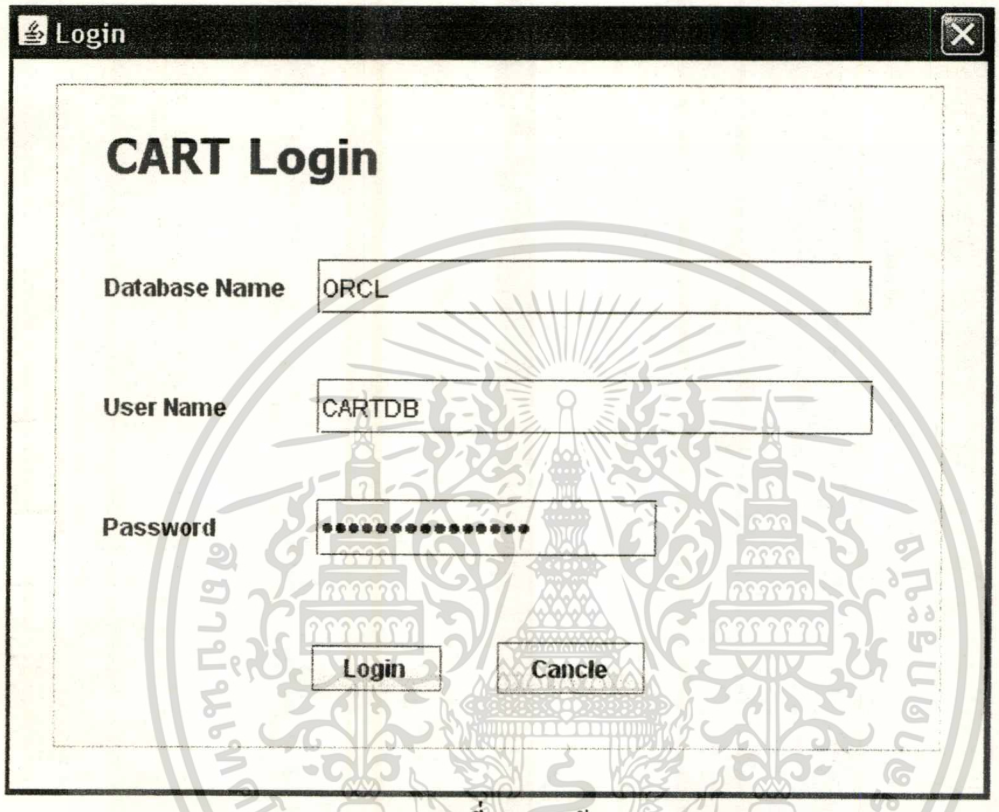
ค่าที่เป็นไปได้	ความหมาย
ELECTRONIC	สินค้าอิเล็กทรอนิกส์ เครื่องใช้ไฟฟ้า
IT	สินค้า IT,Computer
FOOD	อาหาร
CLOTHES	เสื้อผ้า
JEWELRY	เครื่องประดับ
FERNITURE	เฟอร์นิเจอร์

ตารางที่ 6.8 รายการประเภทการซื้อ

ค่าที่เป็นไปได้	ความหมาย
CASH	เงินสด
CREDIT	เครดิต

➢ หน้าจอการ Login

เป็นหน้าจอแรกก่อนเข้าสู่ระบบจะต้องมีการเช็คสิทธิการเข้าใช้งานระบบโดยจะต้องมีการป้อน Database Name,Username,Password ของผู้ใช้เสียก่อน

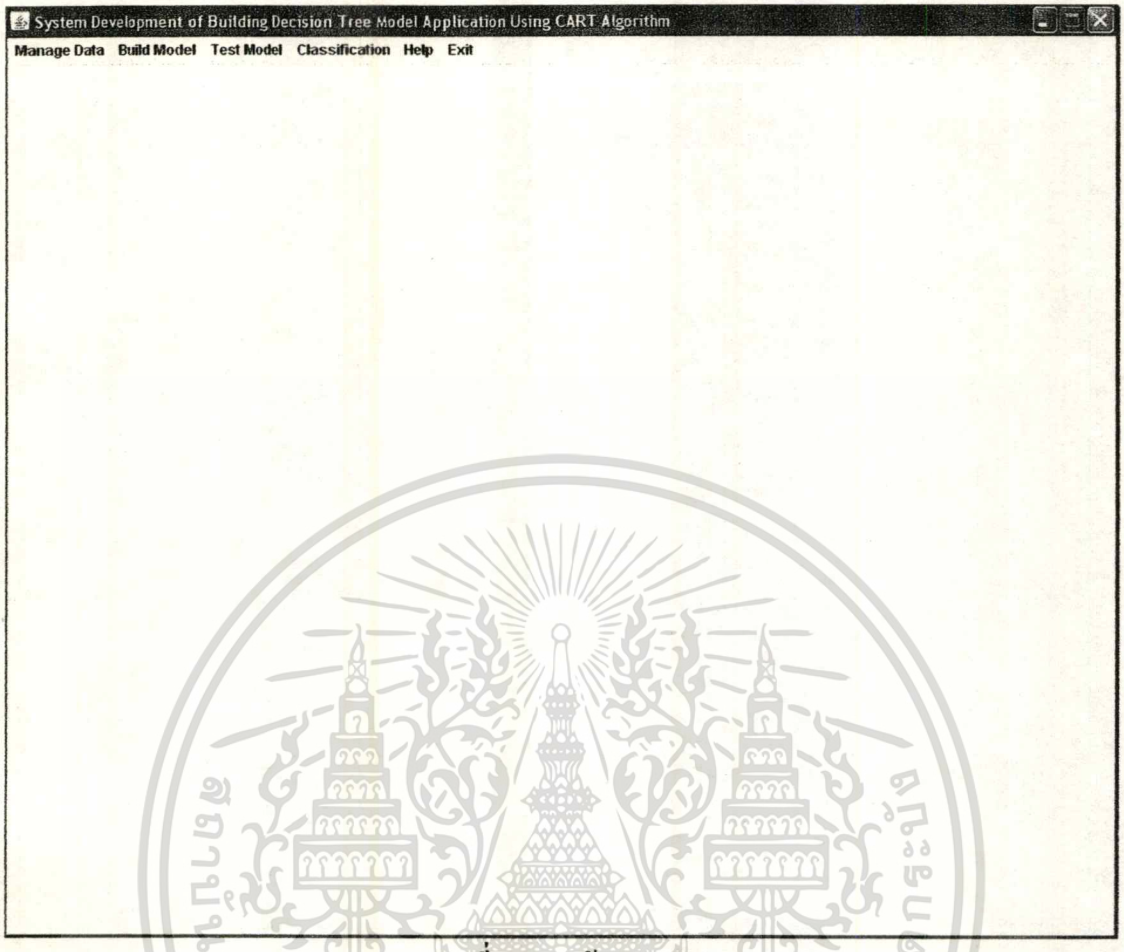


ภาพที่ 6.1 หน้าจอ Login

➢ ส่วนเมนูหลัก

เป็นหน้าจอหลักที่ใช้ในการทำงานในระบบ โดยจะแบ่งออกตามฟังก์ชันการทำงานต่างๆ ที่ได้ออกแบบไว้ ซึ่งจะประกอบด้วยเมนูต่างๆ และปุ่มเครื่องมือการทำงานดังภาพที่ 6.2 ซึ่งประกอบด้วย 5 เมนูย่อยดังนี้คือ

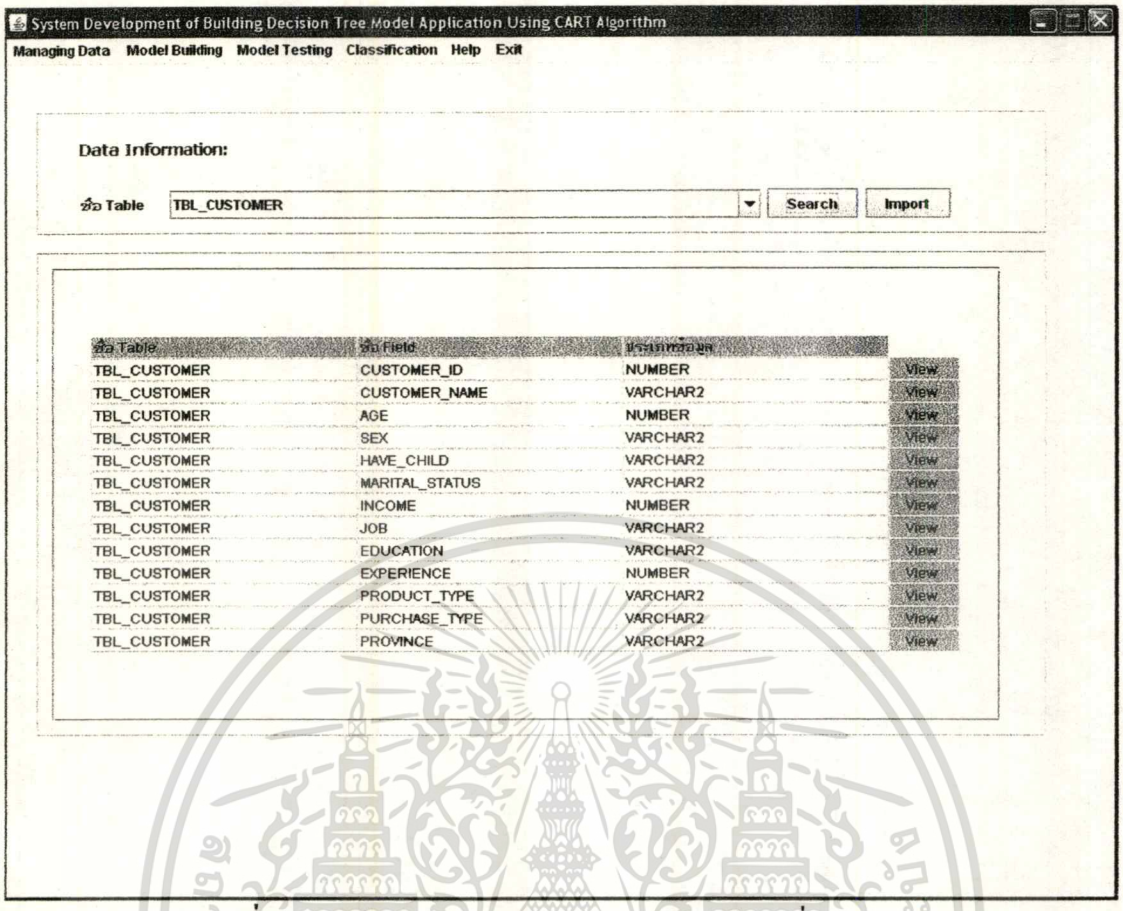
1. Manage data คือ เมนูที่ใช้ในการจัดการข้อมูลสำหรับสร้างโมเดลประกอบด้วย 2 เมนูย่อยคือ View Data For Cleaning และ Create Training Data
2. Build Model คือ เมนูที่ใช้ในการสร้างโมเดลประกอบด้วย 2 เมนูย่อยคือ Inquiry และ Build Model
3. Test Model คือ เมนูที่ใช้ในการทดสอบโมเดล
4. Classification คือ เมนูที่ใช้ในการพยากรณ์ข้อมูลจากการป้อนข้อมูลของผู้ใช้เอง
5. Help คือ เมนูที่แสดงคู่มือการใช้งานระบบ



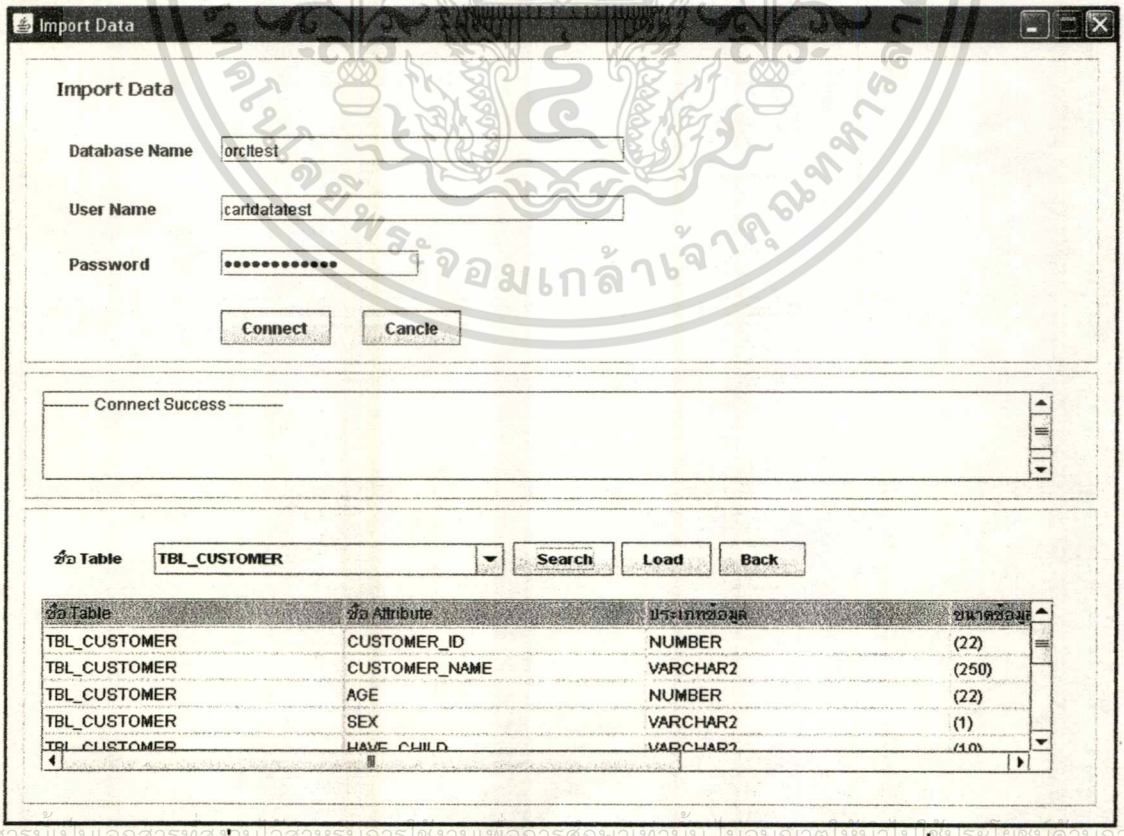
ภาพที่ 6.2 หน้าเมนูหลัก

➢ ส่วนแสดงรายการข้อมูลที่ใช้ในระบบ

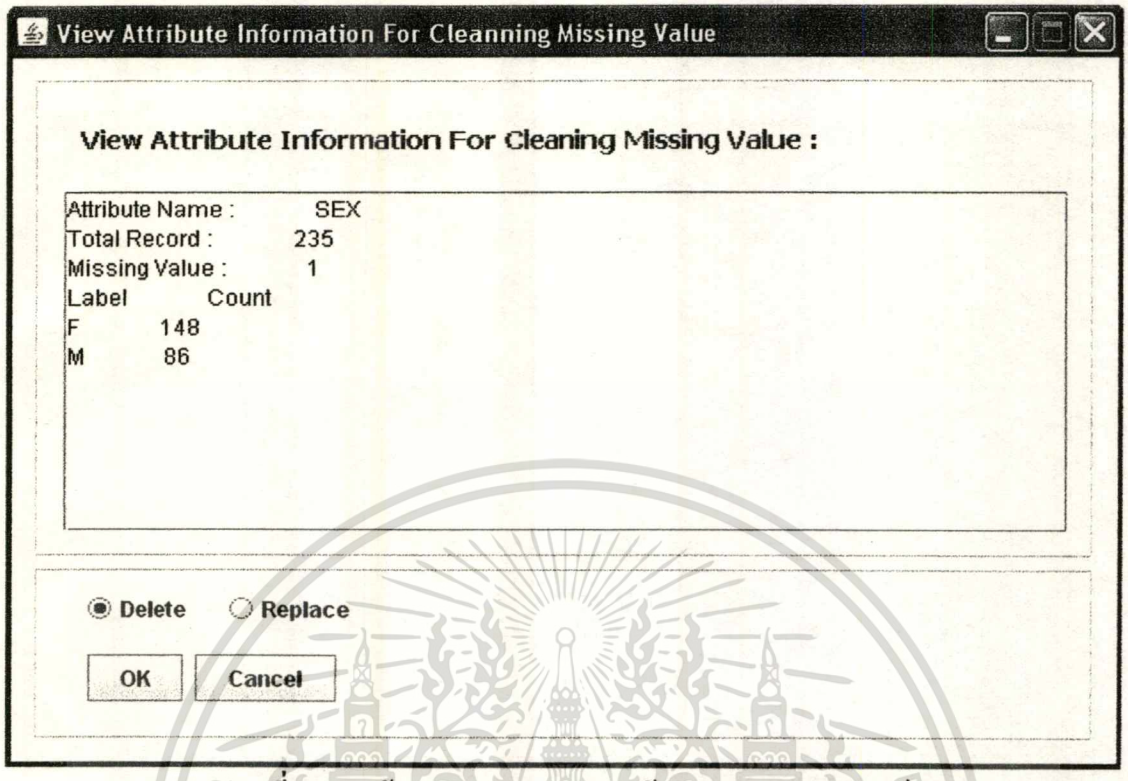
ข้อมูลที่ใช้ในระบบนั้นจะมีส่วนที่คอยจัดการ โดยการให้ผู้ใช้ทำการตรวจสอบข้อมูลในระบบโดยการตรวจสอบเทเบิลข้อมูลที่มีในระบบจะแสดงดังภาพที่ 6.3 หน้าจอแสดงรายการของการแสดงข้อมูลที่มีในระบบ ภาพที่ 6.4 หน้าจอแสดงการนำเข้าข้อมูลสอนระบบจากฐานข้อมูลอื่น และภาพที่ 6.5 หน้าจอแสดงรายการของข้อมูลแต่ละแอตทริบิวต์



ภาพที่ 6.3 หน้าจอแสดงรายการของการแสดงข้อมูลที่มีในระบบ



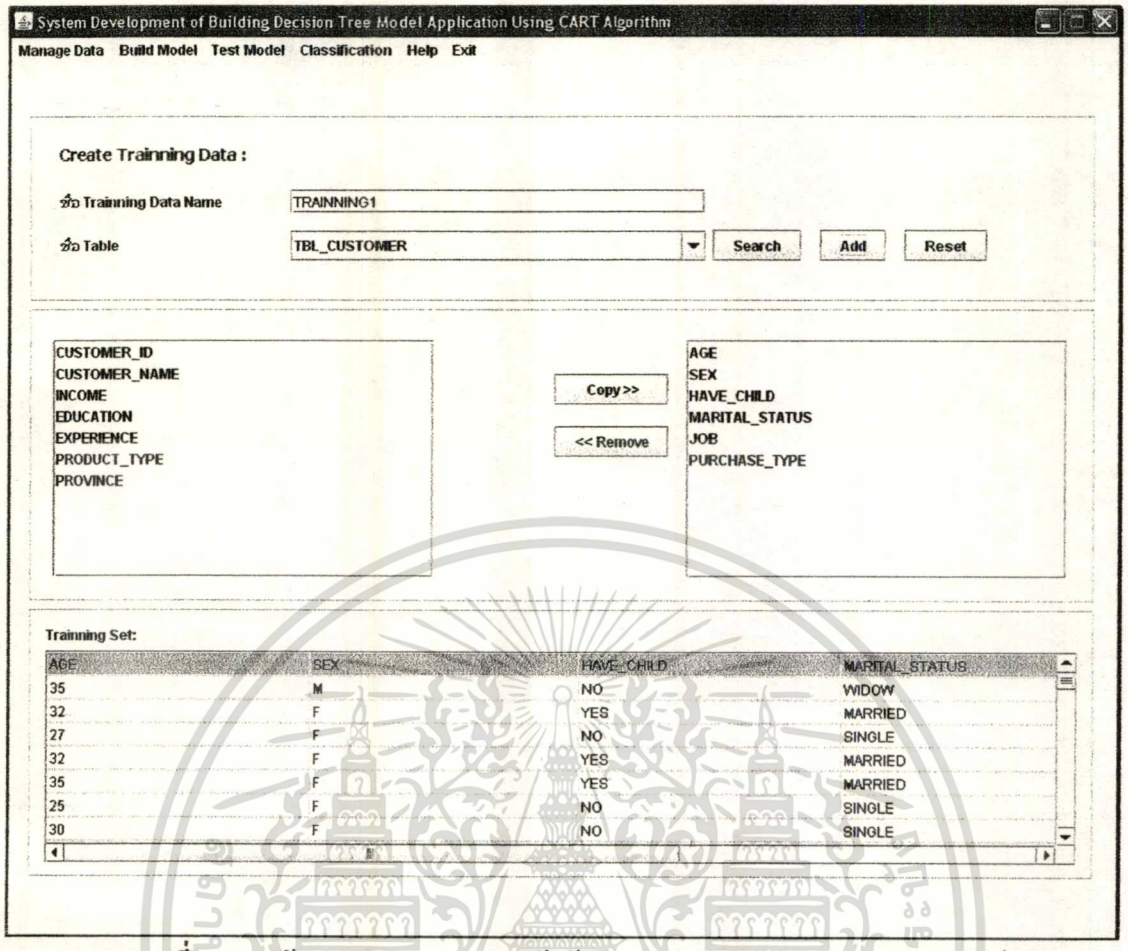
ภาพที่ 6.4 หน้าจอแสดงการนำเข้าข้อมูลสอนระบบจากฐานข้อมูลอื่น



ภาพที่ 6.5 หน้าจอแสดงรายการของข้อมูลแต่ละแอตทริบิวต์

➢ ส่วนแสดงรายการสร้างข้อมูลสอนระบบและคลาสแอตทริบิวต์

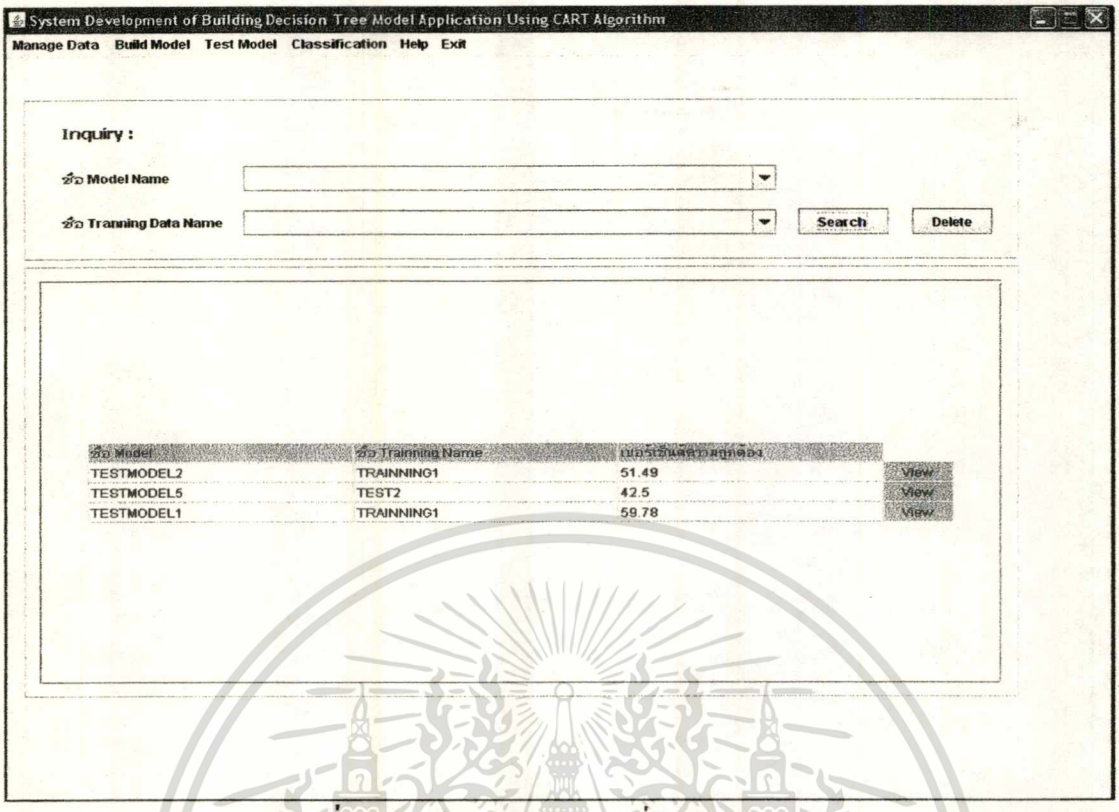
ข้อมูลที่ใช้ในการสร้างโมเดลจะมีข้อมูลสอนระบบและคลาสแอตทริบิวต์โดยระบบนี้จะไม่สามารถสร้างข้อมูลสอนระบบได้จะให้ผู้ใช้นำข้อมูลที่ผ่านการเปลี่ยนรูปแบบข้อมูลให้อยู่ในรูปแบบที่สามารถใช้สอนระบบได้เข้ามาใช้สร้างโมเดล จะแสดงดังภาพที่ 6.6 ส่วนแสดงรายการสร้างข้อมูลสอนระบบและคลาสแอตทริบิวต์



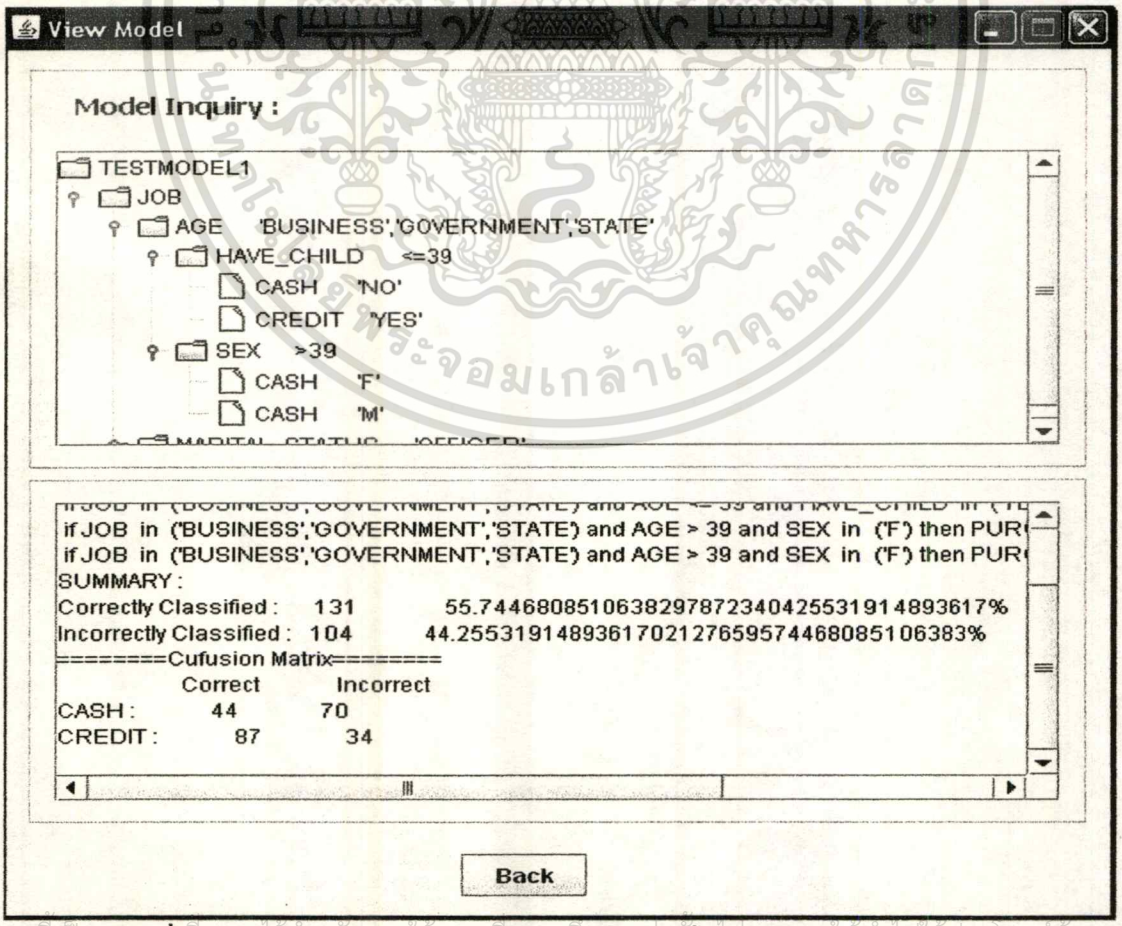
ภาพที่ 6.6 หน้าจอแสดงรายการนำเข้าข้อมูลสอนระบบและคลาสแอตทริบิวต์

➢ ส่วนแสดงการสร้างโมเดลในระบบงาน

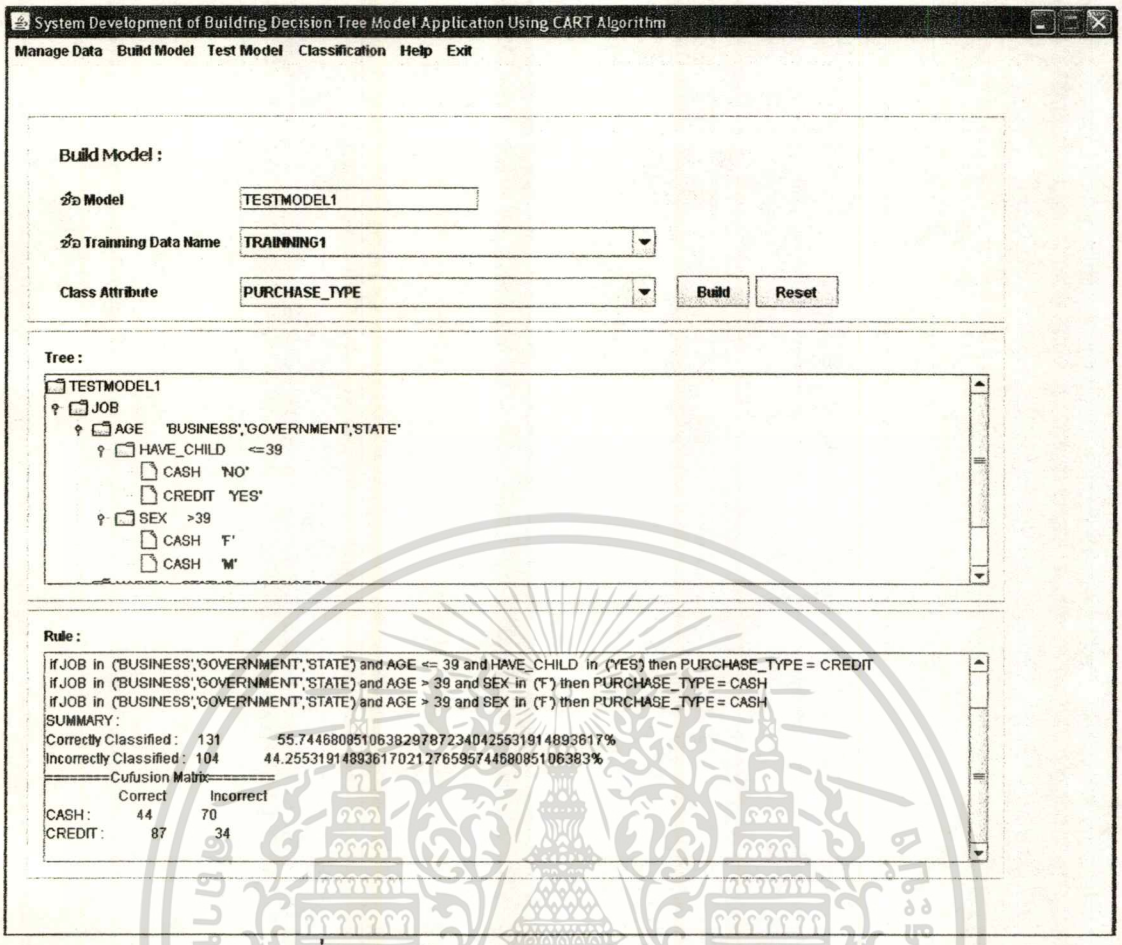
เมื่อได้ข้อมูลสอนระบบและกำหนดคลาสแอตทริบิวต์เป็นที่เรียบร้อยแล้วก็จะเป็นขั้นตอนของการสร้างโมเดลจากข้อมูลเหล่านั้น จะแสดงดังภาพที่ 6.7 หน้าจอแสดงโมเดลที่มีอยู่ในระบบงาน ภาพที่ 6.8 หน้าจอแสดงการสอบถามข้อมูลโมเดลที่เคยถูกสร้างมาแล้วในระบบ และดังภาพที่ 6.9 หน้าจอแสดงการสร้างโมเดลในระบบงาน



ภาพที่ 6.7 หน้าจอแสดงโมเดลที่มีอยู่ในระบบงาน



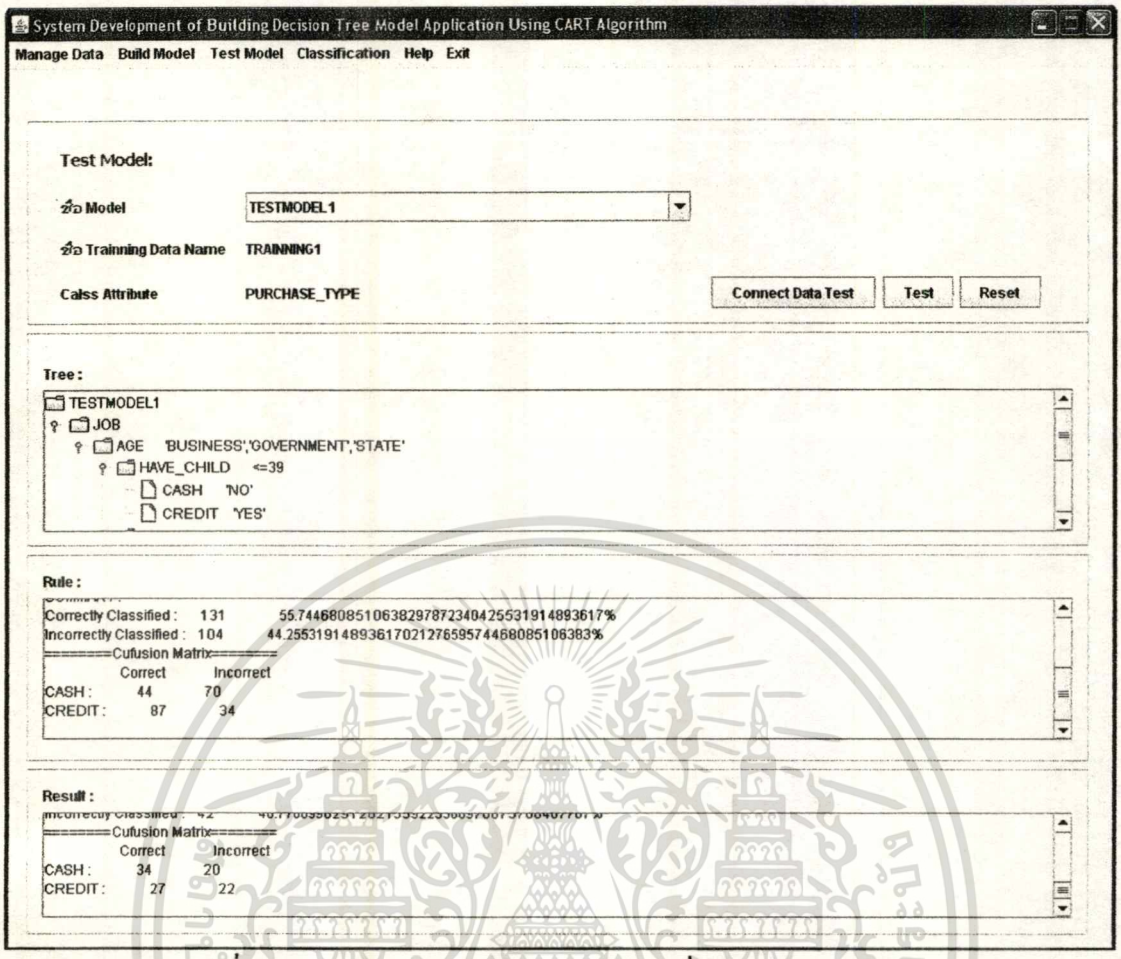
ภาพที่ 6.8 หน้าจอแสดงการสอบถามข้อมูลโมเดลที่เคยถูกสร้างมาแล้วในระบบ



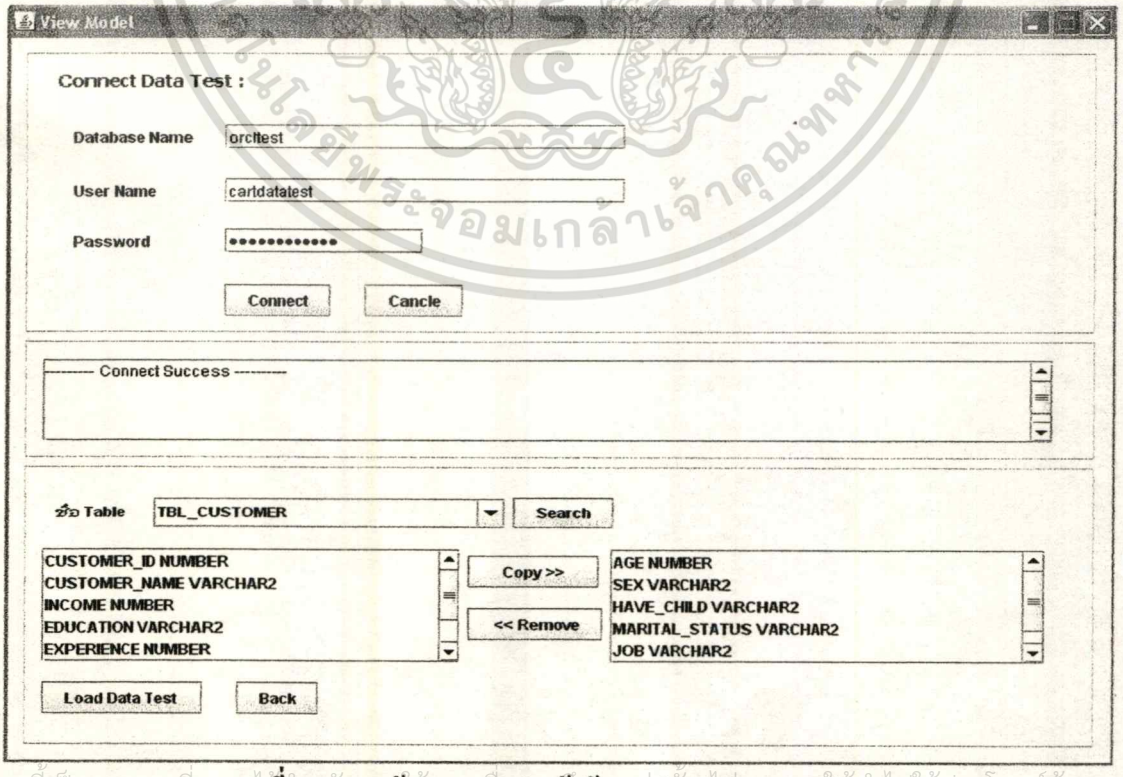
ภาพที่ 6.9 หน้าจอแสดงการสร้างโมเดลในระบบงาน

➢ ส่วนแสดงการนำโมเดลที่ได้ไปทดสอบ

เมื่อได้โมเดลแล้วจากนั้นจะเป็นการนำโมเดลที่ได้ไปทดสอบ จะแสดงดังภาพที่ 6.10 หน้าจอแสดงการทดสอบโมเดลที่ได้จากการสร้างโมเดล และภาพที่ 6.11 หน้าจอแสดงนำเข้าข้อมูลสำหรับทดสอบโมเดล



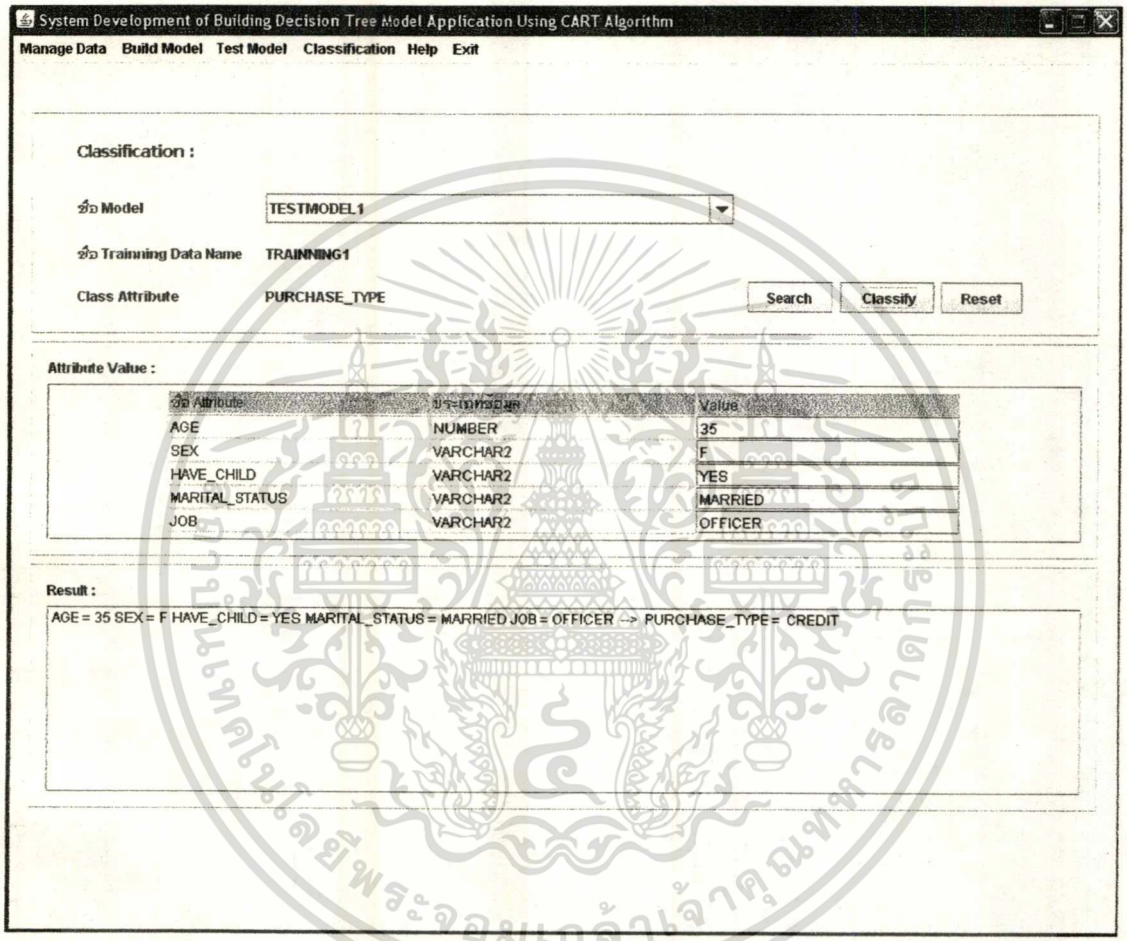
ภาพที่ 6.10 หน้าจอแสดงการทดสอบโมเดลที่ได้จากการสร้างโมเดล



ภาพที่ 6.11 หน้าจอแสดงนำเข้าข้อมูลสำหรับทดสอบโมเดล

➢ ส่วนแสดงการพยากรณ์ข้อมูล

เมื่อได้โมเดลแล้วจากนั้นจะเป็นการนำโมเดลที่ได้ไปประยุกต์ใช้งาน จะแสดงดังภาพที่ 6.12 หน้าจอแสดงนำโมเดลที่ได้ไปแยกประเภทข้อมูลหรือพยากรณ์ข้อมูลในระบบงาน



ภาพที่ 6.12 หน้าจอแสดงนำโมเดลที่ได้ไปแยกประเภทข้อมูลในระบบงาน

บทที่ 7

บทสรุปผลการศึกษา และข้อเสนอแนะ

โครงการพัฒนาระบบดาต้าไมน์นิ่งแบบ Decision Trees โดยใช้อัลกอริทึ่ม CART ในการสร้างโมเดลสำหรับการพยากรณ์ จัดทำเพื่อให้สามารถนำข้อมูลที่มีอยู่นำมาใช้ให้มีประสิทธิภาพเพิ่มมากขึ้น เพิ่มประโยชน์ให้กับข้อมูลที่มีอยู่ และสามารถนำโมเดลมาใช้ในการพยากรณ์ข้อมูลได้ ซึ่งจะทำให้สามารถวิเคราะห์ข้อมูลได้โดยง่ายผ่านทางแอพพลิเคชั่น

7.1 สรุปผลการพัฒนาโปรแกรม

สำหรับการพัฒนาระบบใหม่นี้ มีวัตถุประสงค์เพื่อศึกษาการสร้างโมเดลด้วยอัลกอริทึ่ม CART โดยผ่านข้อมูลที่มีการสอนระบบเป็นที่เรียบร้อยแล้ว และนำโมเดลที่ได้มาใช้ในการพยากรณ์ข้อมูลที่ทำการศึกษาได้ ซึ่งตรงกับวัตถุประสงค์ของการพัฒนาระบบงาน ปัจจัยที่มีความสำคัญก็คือ ข้อมูลที่นำมาใช้ในการสร้างโมเดล นั้นข้อมูลจะต้องมีความสัมพันธ์กันโดยตัวโปรแกรมไม่สามารถที่จะทราบได้ว่าข้อมูลมีความสันพันธ์กันจริงหรือไม่

7.2 ประโยชน์ของการพัฒนาโปรแกรม

- ด้านผู้พัฒนาโปรแกรม
  - เพื่อศึกษาและทำความเข้าใจถึงหลักการทำดาต้าไมน์นิ่งแบบ Decision Trees โดยใช้อัลกอริทึ่ม CART
- ด้านการจัดการข้อมูล
  - เพื่อสามารถจำแนกแยกแยะกลุ่มของข้อมูลที่ต้องการทำการศึกษาได้

7.3 ข้อจำกัดของโปรแกรมที่พัฒนาขึ้น

- ด้านการทำงานของตัวโปรแกรม
  - โปรแกรมที่ได้ทำการพัฒนาขึ้นนั้นในการสร้างโครงสร้างต้นไม้ในแต่ละครั้งผู้ใช้สามารถที่จะทำการเลือกตารางข้อมูลได้เพียงตารางเดียวต่อการสร้างโครงสร้างต้นไม้หนึ่งครั้ง

7.4 ปัญหาและอุปสรรคระหว่างการพัฒนาโปรแกรม

- ปัญหาด้านข้อจำกัดทางด้านเวลา

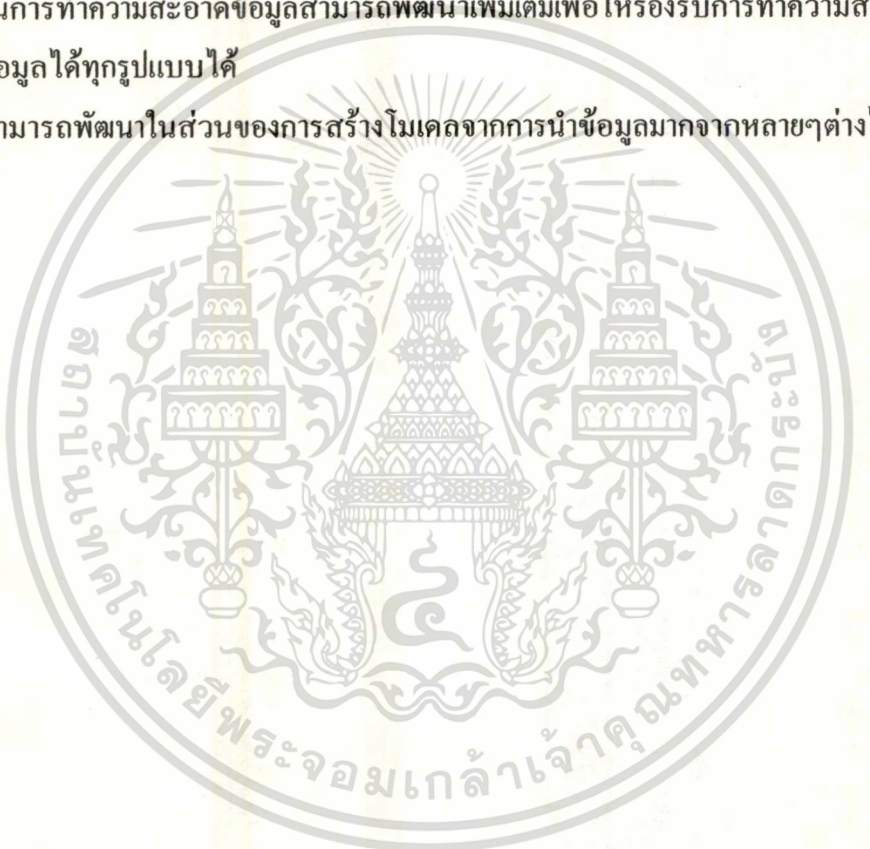
เนื่องจากผู้เขียนต้องทำงานไปด้วยและเรียนไปด้วย และในระหว่างที่ผู้เขียนได้ทำการพัฒนาโปรแกรมนี้ งานประจำที่ที่ผู้เขียนดูแลอยู่มีเพิ่มมาก ทำให้มีเวลาในการพัฒนาโปรแกรม

น้อยลง ประกอบกับความเหนื่อยล้าจากงานประจำที่ทำอยู่ ส่งผลทำให้ไม่สามารถพัฒนาโปรแกรมให้เสร็จได้ทันตามที่ได้ตั้งใจไว้

7.5 ข้อเสนอแนะ

โปรแกรมที่พัฒนาขึ้นใหม่นี้ถือได้ว่าเสร็จสมบูรณ์ในระดับที่น่าพอใจ แต่ก็ยังมีบางส่วนที่สามารถพัฒนาต่อเพื่อให้โปรแกรมมีประสิทธิภาพมากขึ้นได้ ตัวอย่างเช่น

- สามารถมีการพัฒนาในส่วนของการเตรียมข้อมูลสอนระบบได้ ให้รับจำนวนคอลัมน์หรือ Samples Attribute ได้จำนวนไม่จำกัด
- ในการทำความสะอาดข้อมูลสามารถพัฒนาเพิ่มเติมเพื่อให้รองรับการทำความสะอาดข้อมูลได้ทุกรูปแบบได้
- สามารถพัฒนาในส่วนของการสร้างโมเดลจากการนำข้อมูลมากจากหลายๆต่างได้



บรรณานุกรม

กิตติ ภักดีวัฒนะกุล และ ศิริวรรณ อัมพรดนัย. 2544. **Object-Oriented ฉบับพื้นฐาน.** กรุงเทพฯ : บริษัท เคทีพี คอมแอนด์ คอนซัลท์ จำกัด.

สุนทริน วงศ์ศิริกุล. 2543. **พัฒนาโมเดลยุคใหม่ UML มาตรฐานการสร้างโมเดลระบบงาน.** กรุงเทพฯ : ซัคเซสมีเดีย.

Peter Cabena et al. 1998. **Discovering Data Mining.** New Jersey : Prentice Hall.

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

Han, Jiawei ,Kander and Micheline. 2001. **Data Mining Concepts and Techniques**. San Francisco : Morgan Kaufmann Publishers.

L.Breiman,J.H. Friedman,R.A. Olshen and C.J. Stone. 1984. **Classification Regression Trees**. California : Wasdworth International. Group.

Weiss, Sholom M. and Indurkhya, Nitin. 1998. **Predictive Data Mining : A Practical Guide.** San Francisco : Morgan Kaufmann Publishers.

ประวัติผู้เขียน

ชื่อผู้เขียน	นางสาว นภนันท์ อมแก้ว
สถานที่เกิด	จังหวัดนครศรีธรรมราช
ระดับประถมศึกษา	โรงเรียนอนุบาลนครศรีธรรมราช “ณ นครอุทิศ” จังหวัดนครศรีธรรมราช
ระดับมัธยมศึกษาตอนต้น	โรงเรียนสุวรรณสุทธารามวิทยา จังหวัดกรุงเทพฯ
ระดับมัธยมศึกษาตอนปลาย	โรงเรียนสาธิตมัธยมสถาบันราชภัฏสวนสุนันทา จังหวัดกรุงเทพฯ
วุฒิการศึกษาระดับปริญญาตรี	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ประสบการณ์การทำงาน	บริษัท ซอร์ฟเวนเจอร์ จำกัด (มหาชน) บริษัท สกายบลิทซ์ (1995) จำกัด บริษัท โมทีฟ เทคโนโลยี จำกัด (มหาชน)