# วิธีเชิงความหมายสำหรับสร้างโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติ

นนท์ แสนประสิทธิ์

วิทยานิพนธ์เสนอบัณฑิตวิทยาลัย มหาวิทยาลัยนเรศวร
เพื่อเป็นส่วนหนึ่งของการศึกษา หลักสูตรปรัชญาดุษฎีบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศ
ปีการศึกษา 2565

# วิธีเชิงความหมายสำหรับสร้างโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติ

นนท์ แสนประสิทธิ์

วิทยานิพนธ์เสนอบัณฑิตวิทยาลัย มหาวิทยาลัยนเรศวร
เพื่อเป็นส่วนหนึ่งของการศึกษา หลักสูตรปรัชญาดุษฎีบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศ
ปีการศึกษา 2565

วิทยานิพนธ์ เรื่อง "วิธีเชิงความหมายสำหรับสร้างโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติ"
ของ นนท์ แสนประสิทธิ์
ได้รับการพิจารณาให้นับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาปรัชญาดุษฎีบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

**คณะกรรมการสอบวิทยานิพนธ์**

..... ประธานกรรมการสอบวิทยานิพนธ์
(ดร.มารุต บูรณรัช)

..... ประธานที่ปรึกษาวิทยานิพนธ์
(รองศาสตราจารย์ ดร.ไกรศักดิ์ เกษร)

..... กรรมการที่ปรึกษาวิทยานิพนธ์
(รองศาสตราจารย์ ดร.เกตุจันทร์ จำปาไชยศรี)

..... กรรมการผู้ทรงคุณวุฒิภายใน
(ผู้ช่วยศาสตราจารย์ ดร.ดวงเดือน อัศวสุธีรกุล)

..... กรรมการผู้ทรงคุณวุฒิภายใน
(ผู้ช่วยศาสตราจารย์ ดร.สุธาสินี จิตต์อนันต์)

**อนุมัติ**

.....
(รองศาสตราจารย์ ดร.กรองกาญจน์ ชูทิพย์)
คณบดีบัณฑิตวิทยาลัย

**ชื่อเรื่อง** วิธีเชิงความหมายสำหรับสร้างโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติ
**ผู้วิจัย** นนท์ แสนประสิทธิ์
**ประธานที่ปรึกษา** รองศาสตราจารย์ ดร.ไกรศักดิ์ เกษร
**กรรมการที่ปรึกษา** รองศาสตราจารย์ ดร.เกตุจันทร์ จำปาไชยศรี
**ประเภทสารนิพนธ์** วิทยานิพนธ์ ปร.ด. เทคโนโลยีสารสนเทศ, มหาวิทยาลัยนเรศวร, 2565
**คำสำคัญ** คลังข้อมูล, โครงสร้างข้อมูลแบบหลายมิติ, ออนโทโลยี

## บทคัดย่อ

คลังข้อมูล (Data warehouse) เป็นเทคโนโลยีที่มีความสามารถในการสนับสนุนการตัดสินใจซึ่งมีโครงสร้างที่เหมาะสมสำหรับการวิเคราะห์ข้อมูลเพื่อสนับสนุนการตัดสินใจเชิงกลยุทธ์สำหรับผู้ที่มีหน้าที่กำหนดนโยบายในฝ่ายต่าง ๆ ของหน่วยงาน และยังสามารถทำงานร่วมกับเทคนิคเหมืองข้อมูล (Data mining) เพื่อการพยากรณ์แนวโน้มจากข้อมูลที่อยู่ในคลังข้อมูลได้ คลังข้อมูลมีการเก็บข้อมูลในรูปแบบโครงสร้างข้อมูลแบบหลายมิติ (Multidimensional schema) ซึ่งมีโครงสร้างที่ซับซ้อนกว่าโครงสร้างในฐานข้อมูลเชิงสัมพันธ์ ส่งผลให้กระบวนการในการสร้างคลังข้อมูลต้องใช้ระยะเวลานาน มีค่าใช้จ่ายที่สูง และต้องใช้ผู้ที่มีความเชี่ยวชาญในการสร้าง ดังนั้นงานวิจัยนี้จึงเสนอกรอบแนวคิดในการนำฐานความรู้ที่อยู่ในรูปแบบออนโทโลยี (Ontology) มาช่วยในการสร้างโครงสร้างแบบดาว (Star schema) กระบวนการทำงานของงานวิจัยนี้ประกอบ 3 ส่วนได้แก่ 1) การสกัดและวิเคราะห์ข้อมูล 2) การสร้างโครงสร้างแบบดาว และ 3) การสกัดและโหลดข้อมูล เพื่อสนับสนุนการประมวลผลเชิงวิเคราะห์ออนไลน์ (Online analytical processing) ผู้วิจัยเสนอการสร้างโครงสร้างข้อมูลแบบหลายมิติจากข้อมูลแบบกึ่งโครงสร้างในรูปแบบไฟล์ .CSV ความท้าทายของโครงสร้างข้อมูลประเภทนี้คือข้อมูลไม่ได้มีโครงสร้างที่ชัดเจน ไม่มีการระบุคีย์หลัก (Primary key) คีย์นอก (Foreign key) และไม่ได้ระบุความสัมพันธ์ระหว่างตารางไว้ซึ่งสำคัญสำหรับการสร้างโครงสร้างแบบดาว ผู้วิจัยได้เสนอเทคนิคการอนุมานชื่อคอลัมน์โดยใช้ฟังก์ชันความหนาแน่นของความน่าจะเป็น (Probability density function) และการเข้ารหัสเลขคณิต (Arithmetic coding) ในกรณีที่ชื่อคอลัมน์ไม่ปรากฏในแหล่งข้อมูลและเทคนิคการอนุมานชนิดข้อมูลด้วยออนโทโลยี การประเมินประสิทธิภาพของกรอบแนวคิดได้นำข้อมูลใน 3 โดเมนที่ต่างกันมาเปรียบเทียบกัน คือโดเมนทางการแพทย์ (ข้อมูลการระบาดของโรคไข้เลือดออก) โดเมนการเกษตร (ข้อมูลผลผลิตข้าว) และโดเมนธุรกิจ (ข้อมูลการขาย) ผลการวิจัยพบว่ากรอบแนวคิดนี้สามารถสร้างโครงสร้างแบบดาวและอนุมาน

ชนิดข้อมูลและชื่อคอลัมน์ได้อย่างมีประสิทธิภาพ

**Title** A SEMANTIC APPROACH TO AUTOMATE MULTIDIMENSIONAL SCHEMA CONSTRUCTION
**Author** Non Sanprasit
**Advisor** Associate Professor Kraisak Kesorn, Ph.D.
**Co-Advisor** Associate Professor Katechan Jampachaisri, Ph.D.
**Academic Paper** Ph.D. Dissertation in Information Technology, Naresuan University, 2022
**Keywords** Data warehouse, Multidimensional schema, Ontology

## ABSTRACT

Data warehouse (DW) is a leading technology for Decision Support Systems, providing data structures that are useable for data analytics to support strategic decision-making by policymakers in various domains. DWs can be integrated with data mining techniques for forecasting trends based on the data in the DW. However, DWs usually store data in the form of a multidimensional schema, which is a significantly more complex data structure than in the traditional Relational schema. As a consequence, it is a time-consuming and high-cost designing process to develop a DW, even by experts. In this research, a framework is proposed that exploits a knowledgebase model that uses an ontology to assist the development of the framework used for generating a DW star schema. The main contributions of this research include 1) Attribute metadata extraction and analysis 2) Multidimensional schema construction, and 3) Data extraction and loading phase, tables that will be used to support Online Analytical Processing for decision-making. The current version of the presented framework will support the generation of a multidimensional schema from semi-structured data e.g., .CSV file. The main challenge of these data structures is they do not explicitly provide structural data or semantics that identify the primary key, foreign keys, or relationships between tables, which are important for a star schema in a DW. We first introduce the use of the Probability Density Function and Arithmetic coding to handle the uncertainty when the column names in the data source are missing and data Type Inference Techniques with Ontology.

Our proposed approach has been validated by comparison using data from three different domains: the medical domain (dengue fever epidemiology data), the agricultural domain (rice production data), and the business domain (sales information). The results show that our framework can efficiently construct a star schema and effectively predict the missing column names and data types.

# ประกาศคุณูปการ

ผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูงในความกรุณาของ รศ.ดร.ไกรศักดิ์ เกษร ประธาน ที่ปรึกษาวิทยานิพนธ์ รศ.ดร.เกตุจันทร์ จำปาไชยศรี อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม ที่ได้อุตส่าห์สละเวลาอันมีค่ามาเป็นที่ปรึกษา พร้อมทั้งให้คำแนะนำตลอดระยะเวลาในการทำวิทยานิพนธ์ฉบับนี้ และกรรมการผู้ทรงคุณวุฒิทุกท่าน ที่ได้กรุณาให้คำแนะนำตลอดจนแก้ไขข้อบกพร่องของวิทยานิพนธ์ด้วยความเอาใจใส่ จนทำให้วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้อย่างสมบูรณ์และทรงคุณค่า

กราบขอบพระคุณ ดร.มารุต บูรณรัช ที่ได้กรุณาเสียสละเวลาอันมีค่า ให้คำปรึกษาและแนะนำในการสอบวิทยานิพนธ์จนสำเร็จลุล่วง

เหนือสิ่งอื่นใดขอกราบขอบพระคุณ บิดา มารดา ของผู้วิจัยที่ให้กำลังใจและให้การสนับสนุนในทุก ๆ ด้านอย่างดีที่สุดเสมอมา

คุณค่าและคุณประโยชน์อันพึงจะมีจากวิทยานิพนธ์ฉบับนี้ ผู้วิจัยขอมอบและอุทิศแด่ผู้มีพระคุณทุก ๆ ท่าน ผู้วิจัยหวังเป็นอย่างยิ่งว่า งานวิจัยนี้จะเป็นประโยชน์ต่อการศึกษาด้านการสร้างโครงสร้างข้อมูลแบบหลายมิติไม่มากก็น้อย หากมีข้อผิดพลาดประการใดผู้วิจัยขออภัยไว้ ณ ที่นี้ และจะพยายามปรับปรุงแก้ไขให้ถูกต้องตามสมควรทุกประการ

นนท์ แสนประสิทธิ์

# สารบัญ

**หน้า**

# สารบัญตาราง

**หน้า**

# สารบัญภาพ

หน้า

# บทที่ 1

# บทนำ

**ความเป็นมาของปัญหา**

เทคโนโลยีการบันทึกและจัดเก็บข้อมูลในคอมพิวเตอร์นั้นเป็นสิ่งสำคัญสำหรับบริษัทหรือหน่วยงานต่าง ๆ การจัดเก็บข้อมูลจากแหล่งข้อมูลที่แตกต่างกัน มักใช้รูปแบบ (Format) ในการจัดเก็บที่ต่างกัน เช่น รูปแบบไฟล์ตารางคำนวณ (Spreadsheet) รูปแบบไฟล์ .CSV หรือรูปแบบฐานข้อมูลเชิงสัมพันธ์ (Relational database) เป็นต้น ซึ่งการจัดเก็บในรูปแบบฐานข้อมูลเชิงสัมพันธ์เป็นการจัดเก็บข้อมูลเป็นแถว (Row) และคอลัมน์ (Column) อยู่ในลักษณะของตารางสองมิติ ซึ่งแต่ละตารางมีความสัมพันธ์กัน ในการออกแบบฐานข้อมูลต้องจัดให้อยู่ในรูปแบบบรรทัดฐาน (Normalization) เป็นกระบวนการออกแบบฐานข้อมูลที่ช่วยแก้ปัญหาความซ้ำซ้อนของข้อมูล โดยการกระจายรีเลชั่น (Relation) ที่มีโครงสร้างซับซ้อนออกเป็นรีเลชั่นย่อย ๆ ส่งผลให้จำนวนตารางในฐานข้อมูลมีเป็นจำนวนมากและมีความซับซ้อน การจัดทำรายงานจากฐานข้อมูลที่ซับซ้อนต้องมีการกรองข้อมูลที่เกี่ยวข้องจากตารางในฐานข้อมูลเพื่อนำมาจัดทำรายงาน กระบวนการดังกล่าวต้องใช้เวลามากส่งผลต่อความล่าช้าในการจัดทำรายงาน

ปัจจุบันองค์กรส่วนใหญ่ได้นำคลังข้อมูล (Data warehouse) มาใช้เพื่อจัดเก็บข้อมูลที่จำเป็นสำหรับการวิเคราะห์ข้อมูล เพื่อตอบสนองความต้องการของผู้ใช้งานได้อย่างรวดเร็ว สามารถวิเคราะห์ข้อมูลในมุมมองต่าง ๆ จากข้อมูลที่มีอยู่ การหาแนวโน้มในอนาคตหรือการหาองค์ความรู้ใหม่ (Knowledge discovery) เพื่อสนับสนุนการตัดสินใจสำหรับผู้บริหาร คลังข้อมูลมีการใช้กันอย่างแพร่หลายในหลายภาคธุรกิจ เช่น ธุรกิจการตลาด การผลิต การศึกษา และการแพทย์ เป็นต้น (Usman, Pears & Fong, 2012)ข้อมูลในคลังข้อมูลจะถูกรวบรวมจากแหล่งข้อมูลที่แตกต่างกันหลายแหล่ง จัดให้อยู่ในรูปแบบโครงสร้างข้อมูลแบบหลายมิติ (Multidimensional schema) ประกอบด้วยตารางข้อเท็จจริง (Fact table) เป็นศูนย์รวมของข้อมูลและล้อมรอบไปด้วยตารางมิติ (Dimension table) ที่มีรายละเอียดของรหัสที่อยู่ในตารางข้อเท็จจริง ซึ่งตารางมิติจะมีจำนวนเท่าใดก็ได้และต้องมีคีย์ที่สัมพันธ์ไปยังตารางข้อเท็จจริง เพื่อสามารถแสดงผลได้หลากหลายมุมมอง (Selma et al., 2012) ข้อมูลที่ได้สามารถนำมาวิเคราะห์เพื่อกำหนดกลยุทธ์ หาทิศทางการดำเนินงานและเพิ่มประสิทธิภาพในการตัดสินใจของผู้มีอำนาจตัดสินใจ เนื่องจากคลังข้อมูลมาจากแหล่งข้อมูลที่มีความสอดคล้องกันและวิเคราะห์ข้อมูลตามประเด็นที่ผู้ใช้ต้องการ สะดวกและรวดเร็วในการค้นหาข้อมูลอีกทั้งยังสามารถนำมาใช้วิเคราะห์และช่วยในการตัดสินใจด้วยการประมวลผลเชิงวิเคราะห์แบบ

ออนไลน์ (OLAP) โดยผู้ใช้งานสามารถแสดงผลข้อมูลในรูปแบบลูกบาศก์ที่มีหลายมิติ (Cube) กระบวนการดังกล่าวทำให้ผู้ใช้สามารถแสดงมุมมองที่แตกต่างกันของข้อมูลได้โดยง่ายและมีประสิทธิภาพ (Romero, Simitsis & Abelló, 2011)

กระบวนการวิเคราะห์และออกแบบคลังข้อมูลต้องใช้ผู้เชี่ยวชาญด้านการออกแบบคลังข้อมูล มีต้นทุนสูงและใช้ระยะเวลานานในการสร้างและการออกแบบคลังข้อมูล หากผู้สร้างไม่มีความเชี่ยวชาญจะส่งผลให้การออกแบบโครงสร้างข้อมูลแบบหลายมิติไม่มีความสมบูรณ์และขาดประสิทธิภาพ (Pardillo & Mazón, 2011) จากข้อจำกัดเหล่านี้ผู้วิจัยจึงสามารถสรุปปัญหาได้ดังนี้

1. การสร้างโครงสร้างข้อมูลแบบหลายมิติมีความยุ่งยากซับซ้อน จำเป็นต้องใช้ผู้เชี่ยวชาญในการวิเคราะห์และออกแบบคลังข้อมูลส่งผลต่อต้นทุนและระยะเวลาในการสร้าง

2. แหล่งข้อมูลที่ใช้ในการสร้างโครงสร้างข้อมูลแบบหลายมิติในหลายหน่วยงานไม่ได้จัดเก็บในรูปแบบฐานข้อมูลที่มีโครงสร้างชัดเจน แต่มีการจัดเก็บข้อมูลแบบกึ่งโครงสร้างที่อยู่ในรูปแบบไฟล์ .CSV หรือข้อมูลแบบมีโครงสร้างในรูปแบบไฟล์ตารางคำนวณ

3. แหล่งข้อมูลที่อยู่ในรูปแบบกึ่งโครงสร้างไม่มีการระบุชนิดข้อมูลที่ชัดเจนและข้อมูลไม่มีความสมบูรณ์จากการสูญหายของชื่อคอลัมน์ส่งผลต่อการสร้างโครงสร้างแบบหลายมิติ

ดังนั้น งานวิจัยนี้จึงมุ่งเน้นที่การแก้ข้อจำกัดดังกล่าวโดยเสนอกรอบแนวคิดในการออกแบบและสร้างโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติ โดยใช้องค์ความรู้มาช่วยในการวิเคราะห์และออกแบบคลังข้อมูล องค์ความรู้ที่ใช้เก็บอยู่ในรูปแบบของออนโทโลยี ซึ่งสามารถกำหนดหรือนิยามความหมายของข้อมูลและสามารถแสดงถึงความสัมพันธ์กันของข้อมูลต่าง ๆ ได้ ออนโทโลยีจึงถือเป็นการจัดกลุ่มทางความหมาย (Semantic domain) ของระบบสารสนเทศ (Meersman, 1999) โดยมีจุดประสงค์เพื่อช่วยให้ผู้ที่ไม่ใช่ผู้เชี่ยวชาญด้านการออกแบบคลังข้อมูลสามารถสร้างคลังข้อมูลในรูปแบบโครงสร้างแบบดาวจากข้อมูลแบบกึ่งโครงสร้างที่อยู่ในรูปแบบไฟล์ .CSV หรือข้อมูลแบบมีโครงสร้างในรูปแบบไฟล์ตารางคำนวณได้โดยอัตโนมัติ

**จุดมุ่งหมายของการศึกษา**

1. เพื่อพัฒนาเทคนิคการสร้างโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติจากข้อมูลแบบกึ่งโครงสร้าง

2. เพื่อพัฒนาเทคนิคการอนุมานชื่อคอลัมน์จากข้อมูลชนิดตัวอักษรและตัวเลข ในกรณีที่ชื่อคอลัมน์ไม่ปรากฏมากับชุดข้อมูล

3. เพื่อพัฒนาเทคนิคการอนุมานชนิดข้อมูลด้วยออนโทโลยี

**ขอบเขตของงานวิจัย**

**1. ขอบเขตด้านข้อมูล**

ข้อมูลที่นำมาใช้ในการวิจัยเพื่อตรวจสอบความถูกต้องของวิธีเชิงความหมายสำหรับสร้างโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติได้รวบรวมจากแหล่งข้อมูลที่น่าเชื่อถือในประเทศไทยและฐานข้อมูลตัวอย่าง AdventureWorks ประกอบด้วยชุดข้อมูลดังนี้

1.1 ข้อมูลการระบาดของโรคไข้เลือดออกของประเทศไทยในปี พ.ศ. 2546 ถึง พ.ศ. 2560 จากกรมควบคุมโรคและหน่วยวิจัยชีววิทยาและแมลงพาหะนำโรค จุฬาลงกรณ์มหาวิทยาลัย จำนวน 13,764 ระเบียน

1.2 ข้อมูลผลผลิตข้าวนาปีของประเทศไทยในปี พ.ศ. 2552 ถึง พ.ศ. 2560 จากสำนักงานเศรษฐกิจการเกษตรจำนวน 4,544 ระเบียน

1.3 ข้อมูลการขายจากฐานข้อมูลตัวอย่าง AdventureWorks ของบริษัทไมโครซอฟท์ จำนวน 60,855 ระเบียน

**2. ขอบเขตด้านการทำงานของระบบ**

2.1 ระบบสามารถสร้างโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติจากข้อมูลแบบกึ่งโครงสร้างได้

2.1.1 สามารถระบุเมเชอร์จากข้อกำหนดของผู้ใช้ได้

2.1.2 สามารถระบุมิติของข้อมูลจากข้อกำหนดของผู้ใช้ร่วมกับออนโทโลยีได้

2.1.3 สามารถสร้างความสัมพันธ์ของตารางได้

2.2 สามารถอนุมานชื่อคอลัมน์ในกรณีที่ชื่อคอลัมน์ไม่ปรากฏมากับชุดข้อมูลในเอกสารรูปแบบไฟล์ .CSV หรือข้อมูลแบบมีโครงสร้างในรูปแบบไฟล์ตารางคำนวณ

2.3 สามารถอนุมานชนิดข้อมูลด้วยออนโทโลยีได้

2.4 สามารถสร้างรายงานในรูปแบบ OLAP (Online Analytical Processing) จากข้อกำหนดของผู้ใช้ได้

**นิยามศัพท์เฉพาะ**

**คลังข้อมูล (Data Warehouse)** หมายถึง ระบบฐานข้อมูลที่ถูกออกแบบมาเพื่อใช้งานด้านการวิเคราะห์ ซึ่งต่างจากระบบฐานข้อมูลโดยปกติที่ใช้กับงานระบบปฏิบัติการ โดยข้อมูลในคลังข้อมูลจะถูกนำมาใช้เพื่อสนับสนุนการตัดสินใจหรือรายงานผู้บริหาร

**โครงสร้างแบบดาว (Star Schema)** หมายถึง รูปแบบเชิงโครงสร้างทางตรรกะที่ประกอบด้วยตารางข้อเท็จจริงนิยมแสดงไว้ที่จุดกึ่งกลางและล้อมด้วยตารางมิติ ซึ่งตารางมิติจะมีจำนวนเท่าใดก็ได้ และจะมีคีย์ที่สัมพันธ์ไปยังตารางข้อเท็จจริงเท่านั้น

**การประมวลผลเชิงวิเคราะห์ออนไลน์ (Online Analytical Processing)** หมายถึง เทคโนโลยีที่ใช้ข้อมูลจากคลังข้อมูลเพื่อนำไปใช้ในการวิเคราะห์และตัดสินใจทางได้อย่างมีประสิทธิภาพ สามารถค้นหาคำตอบที่ต้องการ และสามารถแก้ปัญหาที่มีความซับซ้อนได้

**เมเชอร์ (Measure)** หมายถึง ประเภทของข้อมูลที่เป็นตัวเลข ทำหน้าที่เก็บจำนวน หรือปริมาณที่เกิดขึ้นของทรานเซคชั่น เช่น จำนวนผู้ป่วยจากโรคไข้เลือดออกหรือจำนวนผู้เสียชีวิต เป็นต้น

**มิติ (Dimension)** หมายถึง ข้อมูลที่เป็นมุมมองให้แก่เมเชอร์ เพื่อประโยชน์ในการวิเคราะห์ข้อมูล

**สมมติฐานของการวิจัย**

1. องค์ความรู้จากออนโทโลยีสามารถนำมาใช้ในการวิเคราะห์และออกแบบโครงสร้างข้อมูลแบบหลายมิติได้

2. ชื่อคอลัมน์ที่สูญหายสามารถอนุมานโดยใช้ฟังก์ชันความหนาแน่นของความน่าจะเป็น (Probability Density Function) และการเข้ารหัสเลขคณิต (Arithmetic coding)

3. องค์ความรู้จากออนโทโลยีสามารถนำมาใช้อนุมานชนิดข้อมูลที่เหมาะสมที่สุดสำหรับข้อมูลแต่ละคอลัมน์ได้

# บทที่ 2

# เอกสารและงานวิจัยที่เกี่ยวข้อง

วิทยานิพนธ์นี้นำเสนอวิธีเชิงความหมายสำหรับสร้างโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติ เพื่อช่วยลดขั้นตอนในการสร้างคลังข้อมูลและการนำข้อมูลไปใช้ โดยนำเสนอแนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้องดังต่อไปนี้

## งานวิจัยที่เกี่ยวข้อง

ในส่วนนี้ผู้วิจัยได้ศึกษางานวิจัยที่เกี่ยวข้องกับการสร้างโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติ ผู้วิจัยได้ทำการรวบรวมและวิเคราะห์บทความมากมายที่เกี่ยวข้องกับการสร้างคลังข้อมูลแบบอัตโนมัติ และการสร้างโครงสร้างแบบดาวสามารถแบ่งตามประเภทของข้อมูลนำเข้าเป็น 3 ประเภท คือ ข้อมูลแบบไม่มีโครงสร้าง ข้อมูลแบบกึ่งโครงสร้าง และข้อมูลแบบมีโครงสร้าง (Siriyasatien, Chadsuthi, Jampachaisri & Kesorn, 2018) ผู้วิจัยได้ศึกษางานวิจัยที่เกี่ยวข้องดังต่อไปนี้

**การสร้างโครงสร้างข้อมูลแบบหลายมิติจากข้อมูลแบบไม่มีโครงสร้างหรือกึ่งโครงสร้าง**

ข้อมูลแบบไม่มีโครงสร้าง คือ ข้อมูลที่ไม่ได้มีการกำหนดรูปแบบข้อมูลไว้ล่วงหน้า ข้อมูลที่ไม่มีโครงสร้างโดยทั่วไปจะแสดงเป็นข้อความอิสระที่เกิดจากการพิมพ์ เช่น เอกสารที่อยู่ในรูปแบบ word และ pdf เป็นต้น การประมวลผลต้องใช้ระบบประมวลผลที่มีความซับซ้อน เช่น การประมวลผลภาษาธรรมชาติ (NLP) Lumbantoruan, Sibarani, Sitorus, Mindari & Sinaga (2014) เสนอวิธีการสร้างฐานข้อมูลแบบดาว (Star schema) จากความต้องการของผู้ใช้ในรูปแบบภาษาธรรมชาติ ข้อมูลอยู่ในลักษณะของประโยคโดยทำการวิเคราะห์จากโครงสร้างของประโยค ความหมาย และชนิดของคำในประโยค ข้อมูลที่สกัดได้จะถูกนำมาสร้างเป็นโครงสร้างข้อมูลแบบหลายมิติ (Multidimensional structure)

ข้อมูลแบบกึ่งโครงสร้าง (Semi structured data) เป็นข้อมูลที่มีโครงสร้างเฉพาะ ไม่ใช่โครงสร้างข้อมูลแบบในฐานข้อมูล จะมีการกำหนดแท็ก (Tags) หรือเครื่องหมายพิเศษ (Maker) เพื่อแบ่งแยกรายละเอียด หรือองค์ประกอบแต่ละส่วนของข้อมูล เช่น เอกสาร XML JSON และ CSV เป็นต้น Hansen, Jensen, Tarp & Thomsen (2017) เสนอวิธีการสร้างโครงสร้างแบบดาวจากไฟล์ .CSV ซึ่งรูปแบบไฟล์ประเภทนี้ไม่มีการระบุชนิดของข้อมูลและความสัมพันธ์ของข้อมูลไว้อย่างชัดเจน ดังนั้นจึงจำเป็นต้องอนุมานชนิดข้อมูลและความสัมพันธ์ของข้อมูลจากข้อมูลที่มีอยู่ซึ่ง

ต้องใช้รูปแบบการวิเคราะห์ทางสถิติระหว่างกระบวนการสร้าง ข้อจำกัดของงานวิจัยนี้คือคลังข้อมูลที่ได้ไม่สามารถระบุระดับชั้นของข้อมูลได้ซึ่งมีความสำคัญสำหรับการวิเคราะห์ OLAP

ภาษามาร์กอัป (markup languages) ถูกนำมาใช้ตั้งแต่ปี 1960 เป็นภาษาที่ใช้ในการกำหนดโครงสร้างของข้อมูลและเอกสาร ข้อมูลที่มีอยู่จะถูกอธิบายความหมายด้วยการกำกับ (Markup) ด้วยแท็ก ยกตัวอย่างของภาษามาร์กอัป เช่น XML RDF และ OWL เป็นต้น ประโยชน์ของการจัดเก็บข้อมูลรูปแบบนี้คือการเข้าถึงได้โดยซอฟต์แวร์ที่หลากหลาย Nebot, Berlanga, Pérez, Aramburu & Pedersen (2009) เสนอวิธีการออกแบบ คลังข้อมูลเชิงความหมาย และใช้ MIO (Multidimensional integrated ontology) เพื่อแปลข้อมูลอินสแตนซ์ลงใน OLAP สำหรับการวิเคราะห์ ข้อจำกัดของวิธีการนี้คือข้อมูลอาจล้าสมัยเมื่อระยะเวลาผ่านไป และวิธีการที่เสนออาจไม่ยืดหยุ่นสำหรับการเปลี่ยนแปลงตามความต้องการของผู้ใช้ (Nebot et al., 2009) ได้เสนอวิธีการสร้างคลังข้อมูลแบบกึ่งอัตโนมัติโดยใช้ข้อมูลจากเว็บเชิงความหมายที่แสดงในรูปแบบ RDF และ OWL สำหรับโดเมนชีวการแพทย์ วิธีการที่เสนอจะวิเคราะห์และสำรวจข้อมูลเชิงความหมายแบบไดนามิกโดยใช้การรวม การนำทาง และการรายงานด้วย OLAP แต่ไม่สามารถสร้างลำดับชั้นหรือระดับของมิติข้อมูลได้

การออกแบบโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติโดยใช้ออนโทโลยีซึ่งเสนอโดย (Romero & Abelló, 2010) ช่วยให้สร้างคลังข้อมูลจากข้อกำหนดของผู้ใช้ได้ ข้อดีของวิธีนี้ คือการสร้างโครงร่างแบบหลายมิติได้โดยไม่ต้องทราบถึงข้อกำหนดของผู้ใช้มาก่อน สามารถขยายแนวคิดของคลังข้อมูล และ OLAP ไปสู่แหล่งข้อมูลอื่น ๆ ด้วยการผสานข้อมูลจากเว็บเชิงความหมายกับคลังข้อมูล เพื่อให้ข้อมูลเป็นปัจจุบันสำหรับโดเมนธุรกิจของตน อย่างไรก็ตามยังต้องมีการระบุเมเชอร์ (Measure) และตารางข้อเท็จจริง (Fact table) ด้วยตนเอง

ผู้วิจัยจึงได้พัฒนาวิธีการอนุมานชื่อคอลัมน์และความสัมพันธ์ระหว่างตาราง เพื่อสร้างฐานข้อมูลแบบดาว แต่งานนี้ยังมีข้อจำกัดคือ ไม่สามารถระบุระดับชั้นของข้อมูล นอกจากนี้การอนุมานชื่อคอลัมน์ยังไม่มีความถูกต้องเพียงพอ

**การสร้างโครงสร้างข้อมูลแบบหลายมิติจากข้อมูลแบบมีโครงสร้าง**

ข้อมูลแบบมีโครงสร้าง (Structured data) เป็นการจัดเก็บข้อมูลในโครงสร้างที่ชัดเจน สามารถนำข้อมูลไปใช้ได้ทันทีและเข้าถึงได้ง่าย รูปแบบข้อมูลแบบมีโครงสร้างที่ใช้ในปัจจุบัน เช่น ฐานข้อมูลเชิงสัมพันธ์ ซึ่งข้อมูลจะถูกจัดเก็บในรูปแบบของตารางที่ประกอบด้วยแถวและคอลัมน์ มีการจัดเก็บข้อมูลในลักษณะที่เป็นกลุ่มของข้อมูลที่มีความสัมพันธ์กัน รูปแบบฐานข้อมูลเชิงสัมพันธ์ได้มีผู้วิจัยนำมาสร้างฐานข้อมูลที่มีโครงสร้างแบบดาว เพื่อนำข้อมูลที่ได้มาใช้วิเคราะห์ให้มีประสิทธิภาพมากขึ้น SAMSTAR (Song, Khare & Dai, 2007) เป็นเฟรมเวิร์คในการสร้างฐานข้อมูลแบบดาว จากฐานข้อมูลเชิงสัมพันธ์ โดยหาตารางข้อเท็จจริง และตารางมิติ (Dimension table) จากการวิเคราะห์

ความสัมพันธ์ของตารางในฐานข้อมูลจึงเป็นเรื่องยากสำหรับ ผู้ที่ไม่ใช่ผู้เชี่ยวชาญในการค้นหาและเลือกตารางข้อเท็จจริงจากตัวเลือกจำนวนมากที่สร้างโดยเฟรมเวิร์กนี้

Sehgal & Ranga (2016) และ Phipps & Davis (2002) เสนอการสร้างฐานข้อมูลที่มีโครงสร้างแบบดาว โดยวิเคราะห์จาก อีอาร์โมเดล (ER Model) ของฐานข้อมูลต้นทาง (Jensen, Holmgren & Pedersen, 2004) เสนอกระบวนการสร้างฐานข้อมูลแบบเกล็ดหิมะ (Snowflake schema) โดยอัตโนมัติจากฐานข้อมูลการประมวลผลธุรกรรมออนไลน์เชิงสัมพันธ์ (OLTP) (Abdalaziz Ahmedl & Mohamed Ahmed, 2014) เสนอการสร้างคลังข้อมูลโดยอัตโนมัติจากฐานข้อมูลแบบ OLTP และได้พัฒนาให้ผู้ใช้สามารถปรับโครงสร้างเพื่อให้เหมาะสมกับความต้องการมากที่สุด อย่างไรก็ตามเทคนิคเหล่านี้มีการทำงานแบบกึ่งอัตโนมัติ เนื่องจากผู้ใช้ต้องเลือกโครงร่างที่ถูกต้องและปรับแต่งโครงร่างที่สร้างขึ้นเพื่อให้สอดคล้องกับข้อกำหนดของของผู้ใช้ระบบ

ปัญหาหลักในการสร้างฐานข้อมูลแบบดาว จากฐานข้อมูลเชิงสัมพันธ์ คือ ฐานข้อมูลเหล่านี้ต้องผ่านการนอร์มัลไลเซชัน (Normalization) ที่เหมาะสมและและมีความสัมพันธ์ระหว่างตารางที่ชัดเจน นอกจากนี้ยังไม่มีการวิเคราะห์ในเชิงความหมายของชื่อคอลัมน์และชื่อตารางในฐานข้อมูล สำหรับข้อมูลที่ไม่มีโครงสร้างและกึ่งโครงสร้างข้อมูลเหล่านี้โดยปรกติจะไม่มีความสัมพันธ์ของข้อมูลและชนิดข้อมูล เป็นข้อจำกัดหนึ่งที่ต้องหาวิธีเพื่อใช้อนุมานข้อมูลเหล่านี้

**การสร้างโครงสร้างข้อมูลแบบหลายมิติจากฐานความรู้**

เทคนิคสำหรับการออกแบบโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติสามารถแบ่งออกเป็น 3 กลุ่ม คือ ขับเคลื่อนด้วยอุปทาน (Supply-driven) ขับเคลื่อนด้วยความต้องการ (Demand-driven) และแนวทางไฮบริด (Hybrid approaches) สำหรับการสร้างโครงสร้างข้อมูลแบบหลายมิติโดยขับเคลื่อนด้วยอุปทาน เป็นการสร้างตามแหล่งข้อมูลซึ่งอาจไม่ตรงกับความต้องการที่แท้จริงของผู้ใช้คลังข้อมูล แตกต่างกับการสร้างโดยการขับเคลื่อนด้วยความต้องการ ที่สามารถสร้างจากข้อกำหนดของผู้ใช้แต่อาจขาดแนวคิดที่น่าสนใจที่ได้จากแหล่งข้อมูล (Thenmozhi & Vivekanandan, 2012) ด้วยข้อเสียเหล่านี้แนวทางไฮบริดจึงเกิดขึ้น ซึ่งข้อกำหนดของผู้ใช้และแหล่งข้อมูลได้นำมาใช้ในขั้นตอนการออกแบบ ซึ่งช่วยให้ระบบสามารถดึงข้อมูลที่น่าสนใจสำหรับการวิเคราะห์ออกมา ดังนั้นผู้วิจัยจึงเน้นศึกษาแนวทางไฮบริดเพื่อใช้ออกแบบโครงสร้างข้อมูลแบบหลายมิติ

Khouri et al. (2012) เสนอวิธีการรวมออนโทโลยีจากแหล่งที่ต่างกันลงในฐานข้อมูลเดียวกันไปไว้ในฐานข้อมูลเดียวกันเรียกว่าฐานข้อมูลออนโทโลยี (Ontology based database) และสร้างคลังข้อมูลจากออนโทโลยีนี้ ซึ่งระบบนี้สามารถรองรับองค์กรขนาดใหญ่ได้ Liu and Iftikhar (2013) เสนออัลกอริทึมการออกแบบคลังข้อมูลทีมีมิติข้อมูลขนาดใหญ่จากออนโทโลยีที่สร้างขึ้น Liu & Iftikhar (2013) เพื่ออธิบายข้อกำหนดของผู้ใช้ ข้อมูลจะถูกแบ่งแยกออกในแนวตั้ง

และแนวนอน ช่วยลดระยะเวลาในการค้นหาข้อมูล นอกจากนี้ยังมีความยืดหยุ่นในการรองรับการเปลี่ยนแปลงข้อกำหนดของผู้ใช้ได้ด้วยการสร้างโครงสร้างขึ้นมาใหม่โดยใช้อัลกอริทึม แต่ระบบยังรองรับเฉพาะการแสดงผลข้อมูลเท่านั้นและขาดความสามารถในการสรุป Gulić (2013)นำเสนอเฟรมเวิร์กกึ่งอัตโนมัติสำหรับการแปลง OWL เป็นคลังข้อมูล ผู้ใช้จำเป็นต้องเลือกข้อเท็จจริงและเมเชอร์ในออนโทโลยีก่อนทำการเปลี่ยนแปลงเป็นคลังข้อมูล Thenmozhi & Vivekanandan (2012) เสนอวิธีไฮบริดสำหรับการออกแบบโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติ ข้อกำหนดของผู้ใช้จะได้รับการประมวลผลโดยอัตโนมัติด้วยใช้อัลกอริทึมการประมวลผลภาษาธรรมชาติ แต่ละกระบวนการทางธุรกิจถูกจับคู่กับแนวคิดในออนโทโลยีเพื่อคำนวณความคล้ายคลึงกันโดยใช้เทคนิค Levenshteinv (Euzenat & Shvaiko, 2013) และเทคนิค Resnik (Otero-Cerdeira, Rodríguez-Martínez & Gómez-Rodríguez, 2015) เสนอวิธีการกำหนดเอนทิตีในตารางมิติแบบอัตโนมัติโดยใช้ความสามารถของออนโทโลยีในการให้เหตุผล งานวิจัยเหล่านี้ระบุเมเชอร์โดยใช้แนวคิดที่มีอัตราส่วนของแอตทริบิวต์ที่เป็นตัวเลขหรือสูงกว่าเกณฑ์ที่กำหนดโดยผู้ออกแบบ ดังนั้นคุณลักษณะที่เป็นตัวเลขเหล่านี้จึงเป็นเมเชอร์ในตารางข้อเท็จจริง อย่างไรก็ตามแนวทางนี้ไม่สามารถนำไปใช้ได้เสมอไปหรือมีความยืดหยุ่นไม่เพียงพอ จึงถือเป็นข้อจำกัดของวิธีนี้

วิธีการแบบไฮบริดที่ผสมผสานแบบจำลองเชิงความหมายเข้าไว้ด้วยกันถือเป็นเทคนิคการออกแบบคลังข้อมูลที่มีประสิทธิภาพ Elamin & Feki (2014)และ Elamin, Alzaidi & Feki (2018)นำเสนอแนวทางไฮบริดแบบใหม่ซึ่งประกอบด้วยสามขั้นตอนหลัก คือ การสร้างโครงสร้างแบบดาวที่ผู้ใช้ต้องการ การสร้าง โครงสร้างแบบดาวของแหล่งข้อมูล และการจับคู่โครงสร้าง เริ่มจากเมเชอร์และข้อเท็จจริงจะถูกดึงโดยอัตโนมัติจากข้อกำหนดทางธุรกิจที่แสดงในรูปของภาษาธรรมชาติโดยใช้วิธีการวิเคราะห์รูปแบบทางภาษาศาสตร์ ขั้นที่สองโครงสร้างแบบดาวถูกสร้างขึ้นตามความสัมพันธ์ของฐานข้อมูล เอนทิตีใช้ในการออกแบบมิติ ความสัมพันธ์ใช้เพื่อสร้างข้อเท็จจริง ในที่สุดโครงสร้างจะถูกจับคู่และรับรองโดยผู้เชี่ยวชาญ Chakiri, El Mohajir & Assem (2020) เสนอการทำงานแบบอัตโนมัติสำหรับการออกแบบ คลังข้อมูล วิธีการแบบผสมถูกสร้างขึ้นคล้ายกับงานของ Elamin et al. (2018) ซึ่งไม่ได้ดำเนินการโดยอัตโนมัติอย่างสมบูรณ์และจำเป็นต้องมีการกำหนดโดยมนุษย์ ข้อจำกัดหลักของ วิธีการที่นำเสนอจำเป็นต้องมีแหล่งข้อมูลเชิงสัมพันธ์ มิฉะนั้นจะไม่สามารถสร้างโครงสร้างได้ นักออกแบบยังต้องปรับแต่งและอนุมัติแบบจำลอง ยังขาดการระบุเมเชอร์สำหรับตารางข้อเท็จจริงที่มีประสิทธิภาพ และจัดการกับความไม่แน่นอนเมื่อไม่มีชื่อแอตทริบิวต์ ส่งผลต่อการสร้างโครงสร้างแบบหลายมิติแบบอัตโนมัติ

จากการทบทวนวรรณกรรมและศึกษางานวิจัยที่เกี่ยวข้องสามารถเปรียบเทียบวิธีการต่าง ๆ จากแหล่งข้อมูลที่แตกต่างกัน ดังตาราง 1

**ตาราง 1 แสดงเปรียบเทียบวิธีการสร้างคลังข้อมูล**

| Methodology | Heterogeneous attribute name recognition | Measure inferencing | Data type inferencing | Attribute name inferencing | Data hierarchy defining | Data warehouse evolutions | Dynamic to business key changes | Supported data representation |
|---|---|---|---|---|---|---|---|---|
| Romero & Abelló (2010) | × | ✓ | × | × | × | ✓ | × | OWL |
| Lumbantoruan et al. (2014) | × | ✓ | × | × | × | × | ✓ | Natural language |
| Hansen et al. (2017) | × | ✓ | ✓ | ✓ | × | × | × | CSV |
| Liu & Iftikhar (2013) | × | × | × | × | × | × | ✓ | OWL |
| Song et al. (2007) | ✓ | ✓ | × | × | × | × | × | Database |
| Sehgal & Ranga (2016) | × | × | × | × | ✓ | × | × | Database |
| Bentayeb et al. (2013) | × | × | × | × | ✓ | ✓ | × | XML |

จากการศึกษางานวิจัยที่เกี่ยวข้องกับการสร้างโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติ พบว่าระบบที่มีอยู่ในปัจจุบันยังมีข้อจำกัดสามารถสรุปได้ดังต่อไปนี้

1. ระบบส่วนใหญ่ไม่รองรับความไม่แน่นอนของชื่อคอลัมน์ที่ใช้คำที่มีความหมายเหมือนกัน ตรงกัน หรือคล้ายกันมาก แต่เขียนและออกเสียงต่างกัน ที่เรียกว่า "คำพ้อง" (Bentayeb et al., 2013; Hansen et al., 2017; Liu & Iftikhar, 2013; Lumbantoruan et al., 2014; Romero & Abelló, 2010; Sehgal & Ranga, 2016) หรือการอนุมาณชื่อคอลัมน์ในกรณีที่ชื่อคอลัมน์สูญหาย (Bentayeb et al., 2013; Liu & Iftikhar, 2013; Lumbantoruan et al., 2014; Romero & Abelló, 2010; Sehgal & Ranga, 2016; Song et al., 2007) ส่งผลให้ไม่สามารถสร้างโครงสร้างข้อมูลแบบหลายมิติได้

2. งานวิจัยหลายงานไม่สามารถกำหนดระดับลำดับชั้นของข้อมูล (Hansen et al., 2017; Liu & Iftikhar, 2013; Lumbantoruan et al., 2014; Romero & Abelló, 2010; Song et al., 2007) เนื่องจากขาดข้อมูลที่แสดงถึงความสัมพันธ์ของข้อมูล ยกตัวอย่างเช่น ลำดับชั้นของสถานที่ สามารถจัดเป็น ประเทศ ภูมิภาค จังหวัด อำเภอ และตำบล เป็นต้น ซึ่งออนโทโลยีสามารถช่วยกำหนดระดับข้อมูลเหล่านี้ซึ่งแสดงอยู่ในโครงสร้างแบบลำดับชั้น ช่วยอำนวยความสะดวกในการจัดระดับข้อมูลได้ง่ายและมีประสิทธิภาพ

3. งานวิจัยบางงานไม่ได้จัดเตรียมฟังก์ชันการระบุเมเชอร์ (Bentayeb et al., 2013; Liu & Iftikhar, 2013; Sehgal & Ranga, 2016) ส่งผลให้ไม่สามารถสร้างตารางข้อเท็จจริงและไม่

สามารถสร้างโครงสร้างข้อมูลแบบหลายมิติได้ ต่างจากงานวิจัยของ (Hansen et al. (2017); Lumbantoruan et al. (2014); Romero & Abelló (2010); Song et al. (2007))ที่มีกระบวนการดังกล่าว

4. งานวิจัยหลายงานไม่มีกระบวนการในการอนุมานชนิดข้อมูล เป็นขั้นตอนที่สำคัญสำหรับข้อมูลแบบไม่มีโครงสร้างหรือข้อมูลแบบกึ่งโครงสร้างที่ไม่มีการกำหนดชนิดข้อมูลไว้ การวิเคราะห์และกำหนดชนิดข้อมูลที่มีขนาดของข้อมูลที่เหมาะสมเป็นสิ่งจำเป็นสำหรับการสร้างตารางและส่งผลต่อขนาดของคลังข้อมูล

**ทฤษฎีที่เกี่ยวข้อง**

จากการศึกษางานวิจัยที่เกี่ยวข้องผู้วิจัยจึงมีแนวคิดการนำวิธีเชิงความหมายสำหรับสร้างโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติ โดยได้ศึกษาทฤษฎีต่าง ๆ ซึ่งทฤษฎีที่สำคัญมีดังต่อไปนี้

1. แนวคิดและทฤษฎีเกี่ยวกับออนโทโลยี
2. แนวคิดและทฤษฎีเกี่ยวกับคลังข้อมูล
3. แนวคิดและทฤษฎีการวิเคราะห์และประมวลผลออนไลน์
4. การอนุมานชื่อคอลัมน์
5. การอนุมานชนิดข้อมูล

**แนวคิดและทฤษฎีเกี่ยวกับออนโทโลยี**

ออนโทโลยี (Ontology) เป็นการกำหนดรูปแบบโครงสร้าง ถูกนำมาใช้แทนความรู้เฉพาะด้านเพื่ออธิบายในขอบเขตของข้อมูลที่สนใจ มีการนำมาประยุกต์ใช้กับระบบงานต่าง ๆ เช่น ด้านปัญญาประดิษฐ์ (AI) วิศวกรรมความรู้ (Knowledge Engineering) ประยุกต์ใช้ในการจัดการภาษาธรรมชาติ (Natural Language Processing: NLP) รวมไปถึงระบบงานต่าง ๆ ที่พัฒนาขึ้นบนเว็บเชิงความหมายเพื่อช่วยในการจัดเก็บและค้นคืนความรู้ โดยมีผู้นิยามความหมายของออนโทโลยีไว้หลากหลายแง่มุมโดย Gruber (1993) ให้นิยามว่าออนโทโลยี คือ ข้อกำหนด (Specification) เพื่อแสดงมโนภาพ (Conceptualization) ของสิ่งต่าง ๆ ที่มีอยู่ในโดเมน (Domain) โดยข้อกำหนดนั้นได้ถูกอธิบายขึ้นเพื่อสร้างความเข้าใจในการใช้งานข้อมูลร่วมกัน (Information Sharing) ระหว่างผู้ที่มีส่วนเกี่ยวข้องในโดเมน Noy & McGuinness (2001) ให้นิยามว่าออนโทโลยีคือรายละเอียดที่ประกาศอย่างเป็นทางการ ซึ่งรายละเอียดในโดเมนนั้นถูกบรรยายด้วยคลาส (Class) สล็อต (Slot) และเงื่อนไข (Restriction) ของสล็อตที่เรียกว่าฟาเซตส์ (Facet) Guarino (1998) ให้นิยามว่าออนโทโลยีคือทฤษฎีทางโลจิกที่ใช้กำหนดความหมายของการอธิบายคำศัพท์ (Vocabulary) ซึ่งมีการกำหนดการอธิบายความหมายนั้นด้วยรูปแบบอย่างเป็นทางการ

โครงสร้างออนโทโลยีประกอบด้วย แนวคิด (Concepts) คุณสมบัติ (Properties) ความสัมพันธ์ (Relationships) ข้อกำหนดในการสร้างความสัมพันธ์ (Axiom) และคำศัพท์หรือข้อมูล (Instances) (มาลี กาบมาลา, ลำปาง แม่นมาตย์ และครรชิต มาลัยวงศ์, 2556) ได้เสนอองค์ประกอบของออนโทโลยีประกอบด้วยส่วนต่าง ๆ ดังต่อไปนี้

1. แนวคิด (Concepts) เป็นขอบเขตความรู้เรื่องใดเรื่องหนึ่งที่สนใจ ซึ่งจะเป็นสิ่งที่ต้องการกล่าวถึง และอธิบายรายละเอียดได้ เช่น งาน (Task) หน้าที่ (Function) การกระทำ (Action) กลยุทธ์ (Strategy) และกระบวนการอย่างมีเหตุผล (Reasoning Process) และอธิบายได้ว่าคลาสต่าง ๆ บรรจุอะไรไว้ในโดเมน ซึ่งคลาสเป็นส่วนที่จะต้องพิจารณาอย่างละเอียดในการพัฒนาออนโทโลยี

2. คุณสมบัติ (Properties) เป็นคุณสมบัติต่าง ๆ ที่เกี่ยวข้องสัมพันธ์กับแนวคิด เพื่อนำมาใช้อธิบายแนวคิด

3. ความสัมพันธ์ (Relationships) เป็นการกำหนดความสัมพันธ์หรือคุณลักษณะของแนวคิด เพื่อเชื่อมโยงความพันธ์ระหว่างแนวคิด โดยมีการกำหนดความสัมพันธ์ในลักษณะต่าง ๆ ดังนี้

3.1 ความสัมพันธ์แบบลำดับชั้น (Subclass of หรือ is-a hierarchy) คือ ความสัมพันธ์ที่มีคุณสมบัติการถ่ายทอด คุณสมบัติของแนวคิดแม่ไปยังแนวคิดลูก

3.2 ความสัมพันธ์แบบเป็นส่วนหนึ่ง (Part-of) คือ ความสัมพันธ์ที่หมายถึงการเป็นส่วนประกอบ

3.3 ความสัมพันธ์เชิงความหมาย (Syn-of) คือ ความสัมพันธ์ที่แสดงถึงแนวคิดที่มีความเหมือนเชิงความหมายต่อกัน

3.4 ความสัมพันธ์การเป็นตัวแทน (Instance-of) คือ ความสัมพันธ์ที่แสดงถึงการเป็นตัวแทน หรือสมาชิกของแนวคิด

4. ข้อกำหนดในการสร้างความสัมพันธ์ (Axiom) เป็นเงื่อนไขหรือข้อกำหนดเฉพาะ หรือตรรกะในการแปลงความสัมพันธ์ระหว่างแนวคิดกับคุณสมบัติแนวคิดกับแนวคิด เพื่อให้แปลงความหมายได้ถูกต้อง

5. คำศัพท์หรือข้อมูล (Instances) เป็นการอธิบายรายละเอียดของข้อมูลซึ่งใช้ข้อมูลเค้าร่างเป็นแม่แบบในการอธิบาย

**ประเภทของออนโทโลยี**

ออนโทโลยีสามารถแบ่งประเภทตามวัตถุประสงค์ของการใช้งานโดยสามารถจำแนกได้ดังต่อไปนี้

1. ออนโทโลยีระดับบน (Top-level Ontology หรือ Upper Ontology) เป็นออนโทโลยีที่ประกอบด้วยเบสคลาส (Base Class) และกำหนดคุณสมบัติเพื่ออธิบายคลาสหรือกำหนดความสัมพันธ์ระหว่างคลาสโดยสามารถนาไปใช้งานได้ในโดเมนทั่วไป (Generic domain)

2. ออนโทโลยีสำหรับกิจกรรม (Task Ontology) เป็นออนโทโลยีที่พัฒนาขึ้นเพื่อตอบสนองการทำงานของกิจกรรมย่อย ๆ โดยอาศัยการถ่ายทอดคุณลักษณะเฉพาะของกิจกรรมจากออนโทโลยีระดับบน

3. ออนโทโลยีสำหรับโดเมน (Domain Ontology) เป็นออนโทโลยีที่ตอบสนองต่อโดเมนโดยอาศัยการถ่ายทอดคุณลักษณะเฉพาะของโดเมนจากออนโทโลยีระดับบน ที่อยู่บนเงื่อนไขของโครงสร้างและเนื้อหาของขอบเขตความรู้โดยมีรายละเอียดครอบคลุมในระบบงานหนึ่ง ๆ

4. ออนโทโลยีระดับโลคอล (Local Ontology) เป็นออนโทโลยีที่ถูกจำกัดการใช้งานในโดเมนที่มีความจำเพาะเจาะจง (Specific domain) ประกอบด้วยคำนิยามวิธีการและมีการระบุหน้าที่ (Task-Specifics) ซึ่งออนโทโลยีประเภทนี้ต้องการโมเดลความรู้สำหรับแอพพลิเคชันต่าง ๆ โดยผสมผสานแนวคิดที่ได้จากออนโทโลยีโดเมนและออนโทโลยีทั่วไป

5. ออนโทโลยีทั่วไป (General Ontology) เป็นออนโทโลยีที่คล้ายกับออนโทโลยีโดเมนแต่จะให้ความสำคัญกับการนำออนโทโลยีกลับมาใช้ใหม่ โดยทั่วไปจะต้องกำหนดแนวคิดเหตุการณ์กระบวนการการกระทำและองค์ประกอบต่าง ๆ ซึ่งถูกกำหนดให้เป็นรายละเอียดของแนวคิด

**ขั้นตอนการพัฒนาออนโทโลยี**

ในการสร้างออนโทโลยี ต้องอาศัยความรู้และความเข้าใจความสัมพันธ์ของสิ่งต่าง ๆ เป็นอย่างดี Antoniou, Groth, Harmelen, Hoekstra & Yu (2012) ได้กล่าวถึงขั้นตอนการพัฒนาออนโทโลยี โดยมีแนวทางและขั้นตอน ดังต่อไปนี้

1. กำหนดขอบเขต (Determine Scope) เป็นการกำหนดกรอบหรือขอบเขตข้อมูลของออนโทโลยีโดยพิจารณาองค์ความรู้ว่าครอบคลุมเรื่องใดบ้าง และวัตถุประสงค์ในการนำออนโทโลยีไปใช้ประโยชน์เพื่ออะไร

2. การเลือกใช้ออนโทโลยีที่มีอยู่ (Consider Reuse) เป็นการพิจารณาการนำออนโทโลยีที่มีอยู่มาปรับใช้ให้เหมาะสมกับขอบเขตออนโทโลยีที่สร้างใหม่ ซึ่งจะช่วยลดระยะเวลาในการสร้างออนโทโลยี

3. กำหนดคำศัพท์ (Enumerate Terms) เป็นการพิจารณาถึงสิ่งที่เกี่ยวข้องหรือคำศัพท์ทั้งหมดที่ต้องการสื่อ ที่อยู่ภายใต้ขอบเขตเรื่องที่สนใจศึกษา โดยระบุคำศัพท์ที่เป็นไปได้

4. การกำหนดคลาส และความสัมพันธ์ของคลาส (Define Taxonomy) เป็นการจัดหมวดหมู่ให้กับคำศัพท์ที่เกี่ยวข้องกัน โดยกำหนดให้คำศัพท์ที่มีลักษณะเหมือนกันไว้เป็นคลาสเดียวกัน และกำหนดความสัมพันธ์ให้กับคลาส หรือการกำหนดลำดับชั้น วิธีการกำหนดลำดับชั้นมี

3 แบบ คือแบบบนลงล่าง (Top-down) แบบล่างขึ้นบน (Bottom-up) และแบบผสม (Combination)

5. กำหนดคุณสมบัติของข้อมูล (Define Properties) เป็นการกำหนดคุณสมบัติโดยพิจารณาว่าเป็นคุณสมบัติแบบทั่วไป (Generality) หรือคุณสมบัติแบบเฉพาะเจาะจง (Specificity)

6. กำหนดมุมมอง และข้อจำกัดของคุณสมบัติ (Define Constraints / Facet of Slots) เป็นการกำหนดเงื่อนไขให้กับคุณสมบัติ โดยพิจารณาจากค่าที่เป็นไปได้ของข้อมูลและความสัมพันธ์ระหว่างคุณสมบัติ

7. กำหนดข้อมูลในคลาส (Define Instances) เป็นการสร้างสมาชิกหรือข้อมูลให้กับคลาสของออนโทโลยี

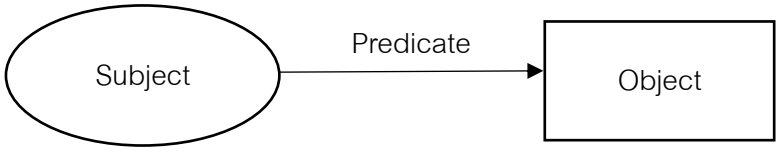
8. ตรวจสอบความผิดปกติ (Check for Anomalies) เป็นขั้นตอนการตรวจสอบและตรวจทานความถูกต้องในการกำหนดความหมายของสิ่งต่าง ๆ ที่อธิบายในโดเมน และแก้ไขให้มีความถูกต้อง

**ภาษาที่ใช้อธิบายข้อมูลในเชิงความหมาย**

ออนโทโลยีจะถูกนำเสนอในสองรูปแบบ คือ อาร์ดีเอฟ (Resource Definition Framework: RDF) และเอาล์ (Ontology Web Language: OWL)

**ภาษาอาร์ดีเอฟ (Resource Definition Framework)**

Antoniou et al. (2012) ได้กล่าวถึงภาษาอาร์ดีเอฟไว้ว่าเป็นภาษาที่ใช้อธิบายลักษณะและความสัมพันธ์ของข้อมูล ที่ใช้สำหรับอธิบายข้อมูลที่อยู่ภายใต้ออนโทโลยี อาร์ดีเอฟถือได้ว่าเป็นตัวกลางที่จะช่วยให้เครื่องคอมพิวเตอร์สามารถเข้าใจข้อมูลที่ถูกบรรยายด้วยภาษาอาร์ดีเอฟร่วมกันได้ และสามารถตีความข้อมูลตามที่ต้องการได้ โดยภาษาอาร์ดีเอฟประกอบด้วย 3 ส่วนที่เรียกว่าทริปเปิล (Triple) คือ ทรัพยากร (Subject) คุณสมบัติของทรัพยากร (Predicate) และค่าของคุณสมบัติ (Object) แสดงดังภาพ 1



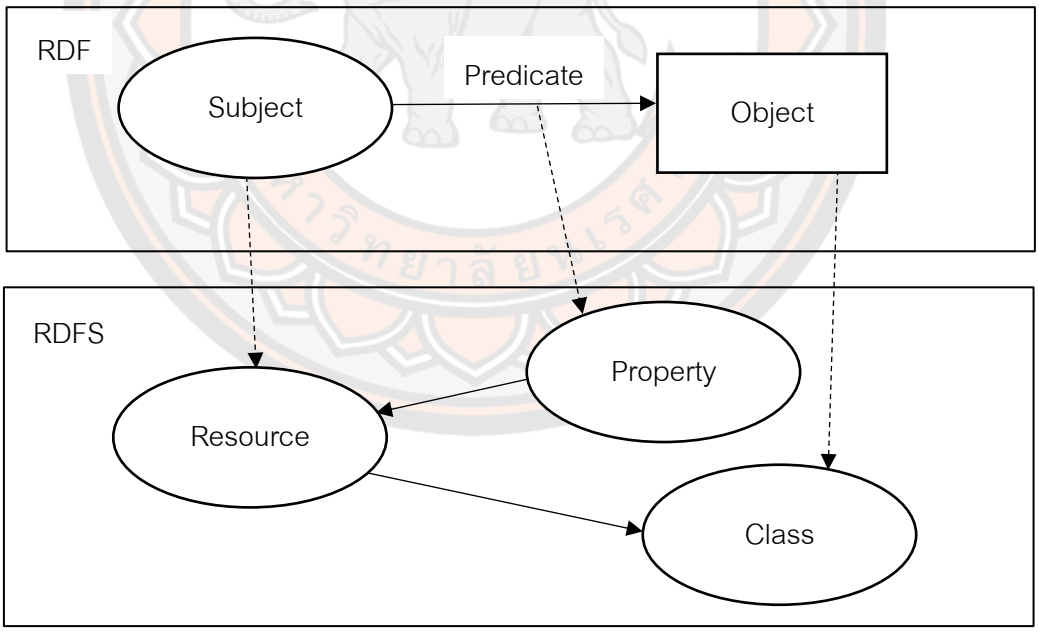

**ภาพ 1 แสดงแบบจำลองโครงสร้างข้อมูลของ RDF**

1. Subject คือ ทรัพยากรหรือสิ่งที่สนใจ เช่น คน สัตว์ สิ่งของ เป็นต้น ใช้สัญลักษณ์แทนด้วยโหนดที่เป็นวงรี โดยที่ทรัพยากรเหล่านี้จะมีการระบุที่อยู่ด้วย URI (Universal Resource Identifier) เพื่อระบุว่าเป็นทรัพยากรที่ไม่ซ้ำกัน

2. Predicate คือ คุณสมบัติของทรัพยากร ใช้สัญลักษณ์แทนด้วยลูกศรชี้จากส่วน Subject ชี้ไปยังส่วน Object หรือ Literal ซึ่งแทนด้วย URI

3. Object คือ ค่าของคุณสมบัติ ใช้สัญลักษณ์แทนด้วยโหนดที่เป็นวงรี ซึ่งแทนด้วย URI แต่ถ้าหากเป็นค่าของอักขระจะเรียกว่า Literal ซึ่งจะใช้สัญลักษณ์แทนด้วยโหนดที่เป็นสี่เหลี่ยม

ภาษาอาร์ดีเอฟมีข้อจำกัดในการอธิบายข้อมูล คือ ไม่สามารถอธิบายเงื่อนไขหรือความหมายของข้อมูลได้อย่างละเอียดทาง W3C จึงได้นำเสนอ RDF Schema (RDFS) เป็นมาตรฐานที่ใช้ในการกำหนดคำนิยามหรือกำหนดโครงสร้างของภาษาอาร์ดีเอฟ Antoniou et al. (2012) ได้ให้ความหมายของอาร์ดีเอฟเอส (Resource Description Framework Schema: RDFS) เป็นภาษาที่พัฒนาเพิ่มจากอาร์ดีเอฟ ที่ทำให้สามารถอธิบายโครงสร้างข้อมูล โดยเป็นการอธิบายโครงสร้างของข้อมูลที่เกิดจากการจัดหมวดหมู่ให้กับข้อมูลให้อยู่ในรูปคลาส (Class) และคุณสมบัติ (Property) มีการจัดลำดับชั้นให้กับข้อมูลดังภาพ 2




**ภาพ 2 แสดงแบบจำลองโครงสร้างข้อมูลของ RDFS**

จากภาพ 2 เป็นการนำโครงสร้างเมทาดาตามาใช้อธิบายข้อมูล ซึ่งประกอบไปด้วย คุณสมบัติ และค่าของคุณสมบัติ มากำหนดให้เป็นโหนดเพื่ออธิบาย Resource โดยในส่วนของ Predicate ซึ่งเป็นคุณสมบัติจะถูกกำหนดเป็นโหนด Property และส่วนของ Object ซึ่งเป็นค่าของคุณสมบัติจะถูกกำหนดให้เป็นโหนด Class โดยทั้งโหนด Property และโหนด Class จะสามารถสืบทอดเป็น SubProperty และ SubClass ได้ตามลำดับ

**ภาษาเว็บออนโทโลยี (Web Ontology Language)**

Antoniou et al. (2012) ได้กล่าวถึงภาษาเว็บออนโทโลยีเป็นภาษาที่สร้างขึ้นเพื่อรองรับการสร้างออนโทโลยี มีวัตถุประสงค์เพื่อใช้เป็นมาตรฐานและได้รับการยอมรับ OWL จัดเป็นองค์ประกอบหนึ่งในงานเว็บเชิงความหมาย (Semantic Web) ที่ใช้ในการบรรยายข้อมูลเชิงความหมาย ถูกออกแบบไวยากรณ์และการสื่อความหมาย มีการสนับสนุนการให้เหตุผลได้อย่างมีประสิทธิภาพ

McGuinness & Harmelen (2004) ได้กล่าวถึงภาษาเว็บออนโทโลยีว่าเป็นการพัฒนาจากภาษา RDF (Resource Description Framework) เพื่อแก้ไขข้อจำกัดในการระบุเงื่อนไขให้กับความสัมพันธ์ระหว่างคลาสได้ โดยภาษาเว็บออนโทโลยีเป็นส่วนประกอบหนึ่งของเว็บเชิงความหมายที่ใช้บรรยายข้อมูลเชิงความหมาย สามารถกำหนดโครงสร้างข้อมูลลำดับชั้นได้

ภาษา OWL เป็นภาษาที่ใช้สำหรับการอธิบายออนโทโลยีและกำหนดความสัมพันธ์ระหว่างข้อมูลตามขอบเขตที่สนใจ ซึ่งพัฒนาต่อมาจากภาษา RDF และสืบทอดมาจากภาษา DAML (DARPA Agent MarkupLanguage) +OIL (Ontology Interchange Language) โดยภาษา OWL ได้นำเอาคลาสและคุณสมบัติของคลาสจาก RDF มาใช้ โดยเพิ่มส่วนของการกำหนดชนิดข้อมูล การบรรยายข้อมูลเชิงตรรกะ และการกำหนดขนาดข้อมูลเข้าไปทำให้ข้อมูลที่ถูกแทนที่มีความหมายมากยิ่งขึ้น ลักษณะการบรรยายจะอยู่ในรูปคลาส คุณสมบัติของคลาสและความสัมพันธ์ระหว่างคลาส เพื่ออธิบายเอ็นทิตี้ (Entity) และความสำพันธ์ต่าง ๆ ที่เกิดขึ้น ภาษา OWL มีประสิทธิภาพอย่างมากในการอธิบายเนื้อหาต่าง ๆ ตามขอบเขตซึ่งคอมพิวเตอร์สามารถอ่านค่าและเข้าใจความหมายของข้อมูล ซึ่งทางองค์กร W3C ได้แบ่งภาษา OWL ออกเป็น 3 กลุ่ม คือ OWL LITE, OWL DL และ OWL FULL โดยแต่ละประเภทถูกออกแบบมาให้เหมาะสมกับการใช้งานตามกลุ่มการใช้งานดังนี้

1. OWL LITE ออกแบบมาเพื่อสนับสนุนการใช้งานเบื้องต้น จะมีการกำหนดโครงสร้างในรูปแบบลำดับชั้นและมีการบังคับใช้คุณสมบัติพื้นฐานในการกำหนดโครงสร้างข้อมูล ถูกออกแบบมาให้ง่ายต่อการพัฒนาและมีการเตรียมฟังก์ชันการใช้งานต่างๆ สำหรับเริ่มใช้งานในการเขียนโอดับเบิ้ลยูแอลได้

2. OWL DL ออกแบบมาเพื่อสนับสนุนการอธิบาย Logic Business Segment โดยใน OWL DL จัดให้มีคุณสมบัติที่เหมาะกับการใช้งานด้านฐานข้อมูล และการแทนความรู้ที่ตั้งอยู่บน

พื้นฐานของการอธิบายด้วยเหตุผลทางตรรกะ OWL DL สามารถบรรยายข้อมูลและโครงสร้างข้อมูลในรูปแบบโครงสร้างภาษาโอดับเบิ้ลยูแอลด้วยการกำหนดข้อจำกัดของคลาสและคุณสมบัติของคลาสได้

3. OWL FULL ออกแบบมาเพื่อสนับสนุนผู้ใช้งานที่ต้องการความครบถ้วนและมีโครงสร้างภาษาที่สมบูรณ์แบบ โดย OWL FULL จะมีการผสมผสานกันระหว่างภาษาโอดับเบิ้ลยูแอลและอาร์ดีเอฟเอส ผู้ใช้งานสามารถบรรยายข้อมูลในรูปแบบอาร์ดีเอฟเอสได้อย่างอิสระ โดยไม่มีการบังคับในส่วนการแบ่งคลาส การกำหนดคุณสมบัติและค่าของข้อมูล

**โครงสร้างของภาษาโอดับเบิ้ลยูแอล**

ภาษาโอดับเบิ้ลยูแอลมีการบรรยายข้อมูลแบบผสมผสานกันระหว่างการใช้ไวยกรณ์ของอาร์ดีเอฟ, อาร์ดีเอฟเอส และเอ็กซ์เอ็มแอลซึ่งแบ่งตามประเภทของการใช้งาน ผลลัพธ์ที่ได้จะอยู่ภายใต้รูปแบบของ RDF Triples จะประกอบด้วยกลุ่มของข้อมูล Namespace, Ontology Header, Class, Property และรายละเอียดต่างๆ ของข้อมูล โดยไฟล์นามสกุลที่ใช้สำหรับการสร้างเอกสารเป็นไฟล์นามสกุล .owl หรือ .rdf (Anon (2012)) อธิบายรายละเอียดส่วนประกอบต่างๆ ได้ดังนี้

1. เนมสเปซ (Namespace) จะประกาศไว้ที่ส่วนเริ่มต้นของเอกสาร เพื่อเป็นการกำหนดกลุ่มในการอ้างอิงข้อมูล เอกสาร OWL ที่ถูกสร้างจะขึ้นอยู่กับโครงสร้างที่ถูกนิยามด้วย RDF, RDFS และชนิดข้อมูลของ XML Schema การเขียนเนมสเปซจะประกาศไว้ภายใต้คำสั่งของ rdf:RDF syntax จากภาพ 3 แสดงตัวอย่างเนมสเปซของออนโทโลยีโรคไข้เลือดออก

```
<rdf:RDF
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
        xmlns:owl="http://www.w3.org/2002/07/owl#"
        xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
        xmlns="http://www.hozo.jp/owl/Dengue.owl#"
        xml:base="http://www.hozo.jp/owl/Dengue.owl#">
```

**ภาพ 3 แสดงตัวอย่างเนมสเปซออนโทโลยีโรคไข้เลือดออก**

2. Ontology Headers แสดงการอธิบายรายละเอียดเบื้องต้นของออนโทโลยีว่าเป็นโครงสร้างข้อมูลเกี่ยวกับอะไรภายใต้ภายใต้ขอบเขตอะไร ประกอบด้วยคำสั่งดังนี้

อธิบายว่าเป็นโครงสร้างข้อมูลเกี่ยวกับอะไรภายใต้ขอบเขตอะไร ใช้คำสั่ง

<owl: Ontology rdf:about="">

อธิบายรุ่นของข้อมูลที่สร้างใช้คำสั่ง

<owl:versionInfo>

อธิบายหมายเหตุของข้อมูล

<rdfs:comment>

อธิบายการอ้างอิงแหล่งข้อมูลว่ามาจากไหน

<owl:imports rdf:resource=" ">

3. การกำหนดคลาส ในการอธิบายคลาสในออนโทโลยี จะมีคลาสเริ่มต้นคือ owl:Class โดยกำหนดให้ owl:Class เป็นคลาสใหญ่ที่สามารถ ครอบคลุมทุกคลาสข้อมูลได้ ดังนั้นไม่ว่าผู้ใช้งานกลุ่มใดสร้างคลาสขึ้นมาจะเสมือนว่าเป็นสมาชิกอยู่ภายใต้คลาส owl:Class เช่น กำหนดคลาส “โรคไข้เลือดออก” เป็นคลาสย่อย (Subclass) ของคลาส “โรคที่มียุงเป็นพาหะ” ซึ่งอธิบายได้ว่าโรคไข้เลือดออกเป็นโรคที่มียุงเป็นพาหะ

4. การกำหนดคุณสมบัติ (Property) การกำหนดคุณสมบัติของคลาสใน OWL สามารถกำหนดได้ 2 ประเภทคือ

4.1 การกำหนดคุณสมบัติด้วย owl:DatatypeProperty เพื่อกำหนดการอธิบายคุณสมบัติของคลาสที่เป็นค่าชนิดพื้นฐาน เช่น การอธิบายข้อมูลจำนวนผู้ป่วยไข้เลือดออก

4.2 การกำหนดคุณสมบัติด้วย owl:ObjectProperty เพื่อกำหนดการอธิบายข้อมูลซึ่งต้องการอธิบายคุณสมบัติของคลาส ซึ่งเป็น Resource หรือการกำหนดการเชื่อมโยงความสัมพันธ์ระหว่างคลาส 2 คลาส

**เครื่องมือที่ใช้ในการพัฒนาออนโทโลยี**

การออกแบบและสร้างออนโทโลยีมีเครื่องมือเป็นจำนวนมากซึ่งแต่ละเครื่องมือจะมีรูปแบบการสนับสนุนการทำงานที่แตกต่างกัน ดังนั้นผู้สร้างออนโทโลยีต้องเลือกเครื่องมือที่มีความเหมาะสม โดยตัวอย่างของเครื่องมือที่ใช้ในการพัฒนาออนโทโลยีมีดังนี้

1. ควอน (KAON) เป็นโปรแกรมที่ทำงานแบบมัลติยูสเซอร์ (Multi-User) ถูกพัฒนาขึ้นโดยมหาวิทยาลัยคาร์ลสรูห์ (Karlsruhe University) ประเทศเยอรมนี โปรแกรมมีการทำงานที่ง่ายต่อการสร้างและค้นหาออนโทโลยีผ่านเว็บบราวเซอร์ ผู้ใช้งานสามารถเข้าใจการเปลี่ยนแปลงที่เกิดขึ้นในออนโทโลยีอันเนื่องมาจากการกระทำที่เกิดขึ้นได้ แต่ไม่อาจทราบได้ว่าใครเป็นผู้เปลี่ยนแปลง

2. โฮโซ (Hozo) เป็นโปรแกรมที่พัฒนาขึ้นโดยมหาวิทยาลัยโอซากา (Osaka University) ประเทศญี่ปุ่น มีส่วนติดต่อผู้ใช้งานเป็นแบบกราฟิก (Graphical User Interface: GUI) ประกอบด้วย 4 ฟังก์ชัน ได้แก่ Ontology Editor, Ontology Manager, Ontology Server และ Onto-studio โดยมีกฎความสัมพันธ์พื้นฐาน คือ Is-a, Past-of และ Attribute-of เป็นเครื่องมือสนับสนุนการพัฒนาออนโทโลยี (Ontology Editor) ที่ได้รับความนิยมในปัจจุบัน มีขั้นตอนการทำงานที่ไม่ซับซ้อนสะดวกต่อการสร้างและนำเสนอออนโทโลยีในรูปแบบแผนผัง ช่วยให้ผู้สร้างสามารถเห็นภาพรวมของออนโทโลยี

3. โปรทีเจ (Protege) เป็นเครื่องมือที่สนับสนุนการพัฒนาออนโทโลยี ที่ถูกพัฒนาขึ้นโดยมหาวิทยาลัยสแตนฟอร์ด (Stanford University) ประเทศสหรัฐอเมริกา เป็นโปรแกรมแบบเปิด (Open Source) และไม่เสียค่าใช้จ่ายในการใช้งาน มีส่วนติดต่อผู้ใช้งานเป็นแบบกราฟิก รองรับการทำงานแบบมัลติยูสเซอร์ โดยจัดเก็บออนโทโลยีในรูปแบบแฟ้มข้อมูลและฐานข้อมูลเชิงสัมพันธ์ อีกทั้งยังมีเครื่องมือสำหรับสร้างโดเมนของออนโทโลยีและรูปแบบข้อมูลที่สะดวกในการป้อนข้อมูล โดยยอมให้ผู้ใช้ทำงานพร้อมกันบนคลาสหรืออินสแตนซ์ใหม่ได้

ในงานวิจัยนี้ผู้วิจัยเลือกใช้โปรแกรมโฮโซเป็นเครื่องมือในการพัฒนาออนโทโลยี เนื่องจากโปรแกรม สามารถใช้งานได้ง่าย มีส่วนการติดต่อผู้ใช้งานเป็นแบบกราฟิก และมีคุณสมบัติที่เหมาะสมในการทำงาน

**แนวคิดและทฤษฎีเกี่ยวกับคลังข้อมูล**

คลังข้อมูล (Data Warehouse) เป็นฐานข้อมูลที่ออกแบบมาเพื่อสนับสนุนการตัดสินใจในองค์กรต่างๆ มีการปรับปรุงข้อมูลและมีโครงสร้างข้อมูลสำหรับการสืบค้นแบบออนไลน์ได้อย่างรวดเร็ว โดยลักษณะของคลังข้อมูลเป็นการการรวมกันเป็นหนึ่ง (Integrated) แบ่งโครงสร้างตามเนื้อหา (Subject oriented) แปรผันตามมิติของเวลา (Time variant) และข้อมูลมีความคงสภาพ (Nonvolatile) เป็นแหล่งจัดเก็บข้อมูลสารสนเทศที่ได้มาจากการประมวลผลข้อมูลจากฐานข้อมูลเชิงปฏิบัติการ (Operational Databases) รวมไปถึงมูลภายนอกองค์กร โดยทำการคัดลอกข้อมูลสารสนเทศเหล่านั้นมาจัดเก็บในแหล่งใหม่ โดยมีข้อมูลสรุปที่อยู่ในกรอบความสนใจเพิ่มเติมเข้ามา และยังถูกจัดเก็บข้อมูลในรูปแบบฐานข้อมูลเชิงสัมพันธ์ (Power, 2002) เป็นกระบวนการในการรวบรวมข้อมูลจากหน่วยงานต่าง ๆ โดยการแปลงข้อมูลให้อยู่ในรูปแบบโครงสร้างที่สอดคล้องกันและจัดเก็บไว้ในแหล่งข้อมูลเดียวกัน ช่วยในการวิเคราะห์ข้อมูล เพื่อสนับสนุนการตัดสินใจให้การบริหารงานในองค์กรเป็นไปอย่างมีประสิทธิภาพ (กิตติพงศ์ กลมกล่อม, 2552)

จากที่กล่าวมาสรุปได้ว่าคลังข้อมูล (Data Warehouse) หมายถึง การจัดเก็บข้อมูลขององค์กร ซึ่งได้จากการรวบรวมข้อมูลจากฐานข้อมูลของระบบปฏิบัติงานประจำวัน (Operational Database) หรือจากแหล่งข้อมูลอื่น แล้วนำมาจัดให้อยู่ในรูปแบบที่เหมาะสมเป็นมาตรฐานเดียวกัน ซึ่งคลังข้อมูลจะเก็บข้อมูลย้อนหลังที่จำเป็นสำหรับการวิเคราะห์ข้อมูลขององค์กร ช่วยให้ผู้ใช้สามารถสืบค้นข้อมูลสำหรับสร้างรายงานและทำการวิเคราะห์ข้อมูลได้สะดวกมากขึ้น สามารถนำมาใช้สนับสนุนการตัดสินใจเพื่อวางแผนได้ตรงตามความต้องการของผู้ใช้

**ลักษณะของระบบคลังข้อมูล**

ลักษณะของระบบคลังข้อมูลมีความแตกต่างจากฐานข้อมูลซึ่งประกอบด้วย 4 ประการดังนี้

1. การจัดกลุ่มตามเนื้อหาของข้อมูล (Subject oriented) คลังข้อมูลสามารถนำมาใช้ในการวิเคราะห์เรื่องใดเรื่องหนึ่ง การจัดกลุ่มข้อมูลตามระบบสารสนเทศดังกล่าวเป็นการแบ่งข้อมูลตามกิจกรรมขององค์กรที่แตกต่างกัน ในคลังข้อมูลข้อมูลจะถูกจัดกลุ่มตามเนื้อหาของข้อมูลโดยการพิจารณาจากข้อมูลในทุก ๆ ระบบ ข้อมูลที่มีความสัมพันธ์กันจะถูกจัดอยู่ในกลุ่มเดียวกันได้

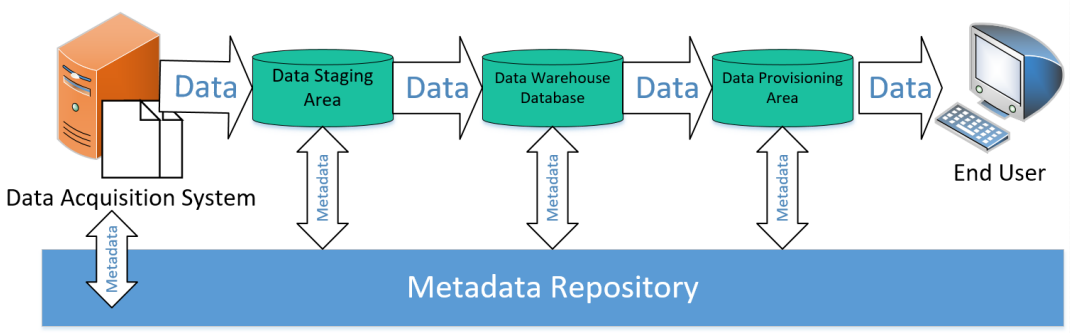
2. การรวมข้อมูลเป็นหนึ่งเดียว (Integrated) ข้อมูลที่เข้าสู่คลังข้อมูลนั้นมาจากหลายแหล่งข้อมูลที่มีความหมายเดียวกันไม่ว่าจะเป็นข้อมูลจากระบบปฏิบัติงานประจำวันขององค์กร ข้อมูลจากระบบสารสนเทศต่าง ๆ ในองค์กร หรือจากแหล่งข้อมูลภายนอก ดังนั้นเมื่อแหล่งข้อมูลมีหลากหลายจึงอาจเกิดความซ้ำซ้อนของข้อมูลขึ้น คลังข้อมูลจึงมีหน้าที่กำจัดความซ้ำซ้อนของข้อมูล ทำให้ข้อมูลจากหลายแหล่งที่มีชนิดของข้อมูลที่แตกต่างกันรวมเป็นชนิดเดียวกัน เพื่อไม่ให้เกิดความสับสนในการวิเคราะห์ข้อมูล

3. ความสัมพันธ์กับเวลา (Time variant) ข้อมูลในคลังข้อมูลแตกต่างจากข้อมูลในระบบปฏิบัติการตรงที่ระบบคลังข้อมูลจะมุ่งเน้นไปที่การเก็บข้อมูลเพื่อการวิเคราะห์ ข้อมูลตามช่วงเวลา ดังนั้นระบบคลังข้อมูลต้องมีการจัดเก็บข้อมูลทั้งในอดีตและปัจจุบันขององค์กร เนื่องจากการตัดสินใจจำเป็นต้องใช้ข้อมูลในอดีตเพื่อใช้เปรียบเทียบในแต่ละช่วงเวลา หรือต้องใช้ข้อมูลในอดีตเพื่อคาดการณ์เหตุการณ์ในอนาคต

4. ความคงสภาพของข้อมูล (Nonvolatile) เพื่อให้การวิเคราะห์ข้อมูลแบบช่วงเวลาให้ผลลัพธ์ที่มีประสิทธิภาพ ข้อมูลในคลังข้อมูลจึงเป็นข้อมูลที่คงอยู่ตลอดไป ไม่เปลี่ยนแปลงแม้ว่าข้อมูลจะเก่าเพียงใด ซึ่งเหมาะกับการวิเคราะห์ข้อมูลย้อนหลัง

**ส่วนประกอบของคลังข้อมูล**

คลังข้อมูล (กิตติพงศ์ กลมกล่อม, 2552) ได้แบ่งองค์ประกอบและความสัมพันธ์ระหว่างองค์ประกอบดังภาพ 4 ไว้ดังนี้




**ภาพ 4 แสดงโครงสร้างของคลังข้อมูล**

1. ระบบรวบรวมข้อมูล (Data Acquisition System) ทำหน้าที่รับข้อมูลจากฐานข้อมูลข้อมูลภายในองค์กรและภายนอกองค์กร โดยทำการตรวจสอบความถูกต้องขั้นต้นก่อนที่จะส่งไปยังส่วนอื่น ๆ การรับข้อมูลนั้นอาจจะได้รับมาโดยผ่านทางระบบเครือข่ายอินเทอร์เน็ต หรือใช้โปรแกรมคอมพิวเตอร์ในการป้อนข้อมูล ข้อมูลที่รับเข้ามาจะได้รับการตรวจสอบความถูกต้องหรือข้อผิดพลาดในการรับข้อมูลแล้วทำการแจ้งผลลัพธ์กลับไปให้ผู้ให้ข้อมูล

2. พื้นที่พักข้อมูล (Data Staging Area) เป็นที่พักและตรวจสอบรายละเอียดของข้อมูลก่อนที่จะนำไปเก็บไว้ใน Data warehouse ได้แก่ การแปลงข้อมูลให้อยู่ในรูปแบบเดียวกัน การตรวจสอบความถูกต้องของข้อมูล การสำรองข้อมูลเบื้องต้น (Temporary Backup) และกระบวนการทำความสะอาดข้อมูล (Data Cleansing) เพื่อทำให้ข้อมูลนั้น พร้อมสำหรับการนำไปเก็บไว้ในระบบคลังข้อมูล ข้อมูลที่พักอยู่จะมีกระบวนการหลายอย่างโดยกระบวนการหลักที่เกิดขึ้นได้แก่

2.1 ตรวจสอบความถูกต้องของข้อมูล

ก่อนนำข้อมูลเข้าสู่คลังข้อมูล กระบวนการตรวจความถูกต้องจะเป็นการตรวจสอบความถูกต้องของการมีค่าของข้อมูล (Data Consistency) ซึ่งกระบวนการนี้จะทำหน้าที่ตรวจสอบข้อมูลที่เข้ามาว่ามีค่าเป็นไปได้ (Cardinality) ของของฟิลด์ต่าง ๆ ตรงตามที่กำหนดไว้หรือไม่

ความถูกต้องของค่าที่เป็นไปได้ของข้อมูล (Possible Values) คือข้อจำกัดของค่าที่จะมีอยู่ในฟิลด์ใดฟิลด์หนึ่งของข้อมูล เช่น ข้อมูลที่มีชนิดข้อมูลเป็นเดือนจะต้องมีมีค่าเป็น 1-12 หรือมกราคม ถึง ธันวาคม เป็นต้น

ความถูกต้องของความสัมพันธ์ของข้อมูล (Data Relationship) ซึ่งข้อมูลจะต้องมีความสัมพันธ์กับข้อมูลอื่น ๆ ตามที่ออกแบบไว้ในแบบจำลองข้อมูล โดยภาษาที่ใช้ในการจัดการข้อมูลในฐานข้อมูลนั้นอาจจะใช้ ภาษา SQL (Structured Query Language) ช่วยในการจัดการข้อมูล โดยภาษา SQL เป็นภาษาทางด้านฐานข้อมูล แบ่งออกเป็น 3 กลุ่ม คือ ภาษาที่ใช้สำหรับนิยาม

ข้อมูล (Data Definition Language: DDL) ภาษาที่ใช้ในการจัดการข้อมูล (Data Manipulation Language: DML) และภาษาที่ใช้ในการควบคุม (Data Control Language: DCL)

2.2 เป็นพื้นที่สำรองข้อมูลชั่วคราว (Temporary Backup) เนื่องจากในขั้นตอนของการตรวจสอบความถูกต้องของข้อมูลนั้น เป็นขั้นตอนที่ต้องใช้เวลานานในการตรวจสอบ ดังนั้นในระหว่างที่กำลังดำเนินการอยู่นั้น จึงต้องมีการสำรองข้อมูลชั่วคราวไว้ด้วย เพื่อป้องกันการเกิดเหตุการณ์ไม่คาดคิดที่จะสร้างความเสียหายต่อข้อมูล เพราะถ้าหากเกิดความผิดพลาดของกระบวนการนำเข้าข้อมูล ยังสามารถนำข้อมูลที่สำรองไว้เข้าสู่กระบวนการตรวจสอบข้อมูลใหม่อีกครั้ง

2.3 กระบวนการ ETL Data Staging Area เป็นการเคลื่อนย้ายข้อมูลจากที่หนึ่ง (Source) ไปยังอีกที่หนึ่ง (Destination) ซึ่งเรียกว่ากระบวนการ Extract-Transform-Load (ETL) ประกอบด้วย 3 ขั้นตอน ได้แก่

ขั้นตอนที่ 1 การสกัดข้อมูล (Extract) คือ กระบวนการดึงข้อมูลออกจากแหล่งข้อมูล

ขั้นตอนที่ 2 การแปลงข้อมูล (Transform) คือ การแปลงข้อมูลจากโครงสร้างเดิมที่กำหนดไว้ในแหล่งข้อมูลให้อยู่ในรูปแบบโครงสร้างข้อมูลตามที่กำหนดไว้

ขั้นตอนที่ 3 การโหลดข้อมูล (Load) คือการนำข้อมูลที่แปลงรูปแบบแล้วเข้าสู่ระบบปลายทาง

3. ฐานข้อมูลของคลังข้อมูล (Data Warehouse Database) เป็นที่เก็บบันทึกข้อมูลต่าง ๆ ที่จำเป็นสำหรับการวิเคราะห์ข้อมูลขององค์กร ข้อมูลที่เก็บไวในฐานข้อมูลของระบบคลังข้อมูลมีความแตกต่างจากระบบสารสนเทศทั่ว ๆ ไปซึ่งในระบบสารสนเทศทั่วไปนั้น ข้อมูลที่ทำการจัดเก็บสามารถถูกเปลี่ยนแปลงปรับเปลี่ยนได้ แต่ข้อมูลในคลังข้อมูลนั้น จะมีลักษณะของการเก็บแบบตลอดไป ไม่แก้ไขข้อมูลหากไม่จำเป็น แต่ถ้ามีการเปลี่ยนแปลงแก้ไข ข้อมูลนั้นจะถูกเพิ่มเข้าไป และให้ข้อมูลเดิมเปลี่ยนเป็นข้อมูลในอดีตของข้อมูลปัจจุบันแทน ข้อมูลในคลังข้อมูลจะไม่ถูกลบออกไป

4. พื้นที่ของข้อมูลที่จะใช้วิเคราะห์ (Data Provisioning Area) ทำหน้าที่บันทึกข้อมูลและผลลัพธ์ต่าง ๆ ที่จำเป็นสำหรับการวิเคราะห์ข้อมูล ซึ่งข้อมูลจากฐานข้อมูลของระบบคลังข้อมูลจะถูกดึงและประมวลผล แล้วนำผลลัพธ์ที่ได้เก็บไว้ใน Data Provisioning Area หรือ Data Mart ซึ่งโครงสร้างข้อมูลนั้นอาจมีลักษณะที่คล้ายคลึงกับฐานข้อมูลของระบบคลังข้อมูล หรือมีโครงสร้างข้อมูลที่เหมาะสมสำหรับการนำข้อมูลไปใช้งาน

Data Mart คือ การตัดเอาบางส่วนของฐานข้อมูลของระบบคลังข้อมูลมาใช้งาน และจัดเตรียมรูปแบบที่ง่ายในการเข้าถึงข้อมูลเพื่อนำไปใช้งานต่อไป โดยโครงสร้างมักจะอยู่ในรูปของ

Cube เพื่อตอบสนองจุดประสงค์ของการใช้งานอย่างใดอย่างหนึ่งหรือเพื่อกลุ่มงานใดกลุ่มงานหนึ่งหรือหลาย ๆ กลุ่มงานในองค์กร

5. การใช้งานและนำเสนอข้อมูล (End User Terminal) เป็นผู้ที่ทำหน้าที่ดึงเอาข้อมูลที่ได้ถูกเตรียมไว้ใน Data Mart หรือฐานข้อมูลของระบบคลังข้อมูลไปนำเสนอข้อมูล โดยใช้ระบบหรือเครื่องเมือเพื่อช่วยในการออกรายงาน

6. ข้อมูลเพื่อการควบคุมการทำงานและควบคุมข้อมูลของคลังข้อมูล (Metadata Repository) พื้นที่ที่เก็บข้อมูลต่าง ๆ ที่จำเป็นสำหรับการควบคุมการทำงานและควบคุมข้อมูล ซึ่งเรียกว่า “Metadata” โดยจะมีข้อมูลที่เกี่ยวข้องกับข้อมูลต่าง ๆ ของระบบเพื่ออธิบายความหมายของคำจำกัดความของข้อมูล และเพื่อใช้เป็นข้อมูลสำหรับการดำเนินการต่าง ๆ กับข้อมูลตามกระบวนการของคลังข้อมูล

**โครงสร้างของคลังข้อมูล**

การออกแบบระบบสารสนเทศเพื่อการประมวลผลเชิงวิเคราะห์ออนไลน์ (Online Analytical Processing) จะใช้แบบจำลองข้อมูลที่เรียกว่าแบบจำลองฐานข้อมูลแบบมัลติไดเมนชั่น (Multidimensional Database Model) แบบจำลองของการประมวลผลเชิงวิเคราะห์ออนไลน์มักจะทำในลักษณะของคิวบ์ (Cube) ซึ่งเปรียบเหมือนกับรูปลูกบาศก์ที่มีมุมมองหลากหลาย แต่ละมุมมองจะทำให้เกิดการสืบค้นข้อมูลได้หลากหลายมิติ คิวบ์ประกอบด้วยส่วนประกอบพื้นฐานที่สำคัญ 2 ส่วน คือ ตารางมิติ และตารางข้อเท็จจริง การผสมผสานของมิติต่าง ๆ ของคิวบ์จะช่วยสร้างผลลัพธ์หลากหลายรูปแบบได้

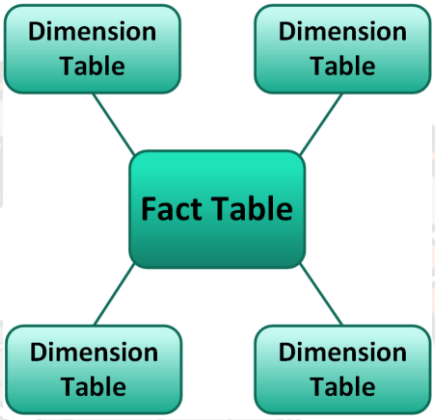
ตารางข้อเท็จจริง เป็นตารางที่เก็บข้อมูลที่ต้องการจะนำมาวิเคราะห์ เป็นข้อมูลที่สามารถใช้ตอบคำถามที่ต้องการ เช่น ข้อมูลการระบาดของโรคไข้เลือดออกจะมีการจัดเก็บข้อมูลจำนวนผู้ป่วย และมีรายละเอียดเกี่ยวกับเวลาที่ระบาด สถานที่ระบาด เป็นต้น ข้อมูลในตารางนี้จะไม่มีการแก้ไขหากไม่จำเป็น โดยปรกติจะใช้การเพิ่มข้อมูลเข้าไปในตาราง

ตารางมิติ เป็นตารางที่เก็บมิติของข้อมูลเพื่อใช้ตอบคำถาม โดยแสดงข้อมูลที่เป็นความหมายของรหัสที่อยู่ในตารางข้อเท็จจริง เพื่อช่วยในการสืบค้นหาคำอธิบายของรหัสที่ใช้ในตารางข้อเท็จจริง เช่น ข้อมูลการระบาดของโรคจะมีตารางสถานที่เพื่อใช้เก็บข้อมูล ภูมิภาค จังหวัด และอำเภอ เป็นต้น

การจำแนกลักษณะของแบบจำลองฐานข้อมูลแบบมัลติไดเมนชั่น แบ่งออกเป็น 2 แบบคือ โครงสร้างแบบดาว และ โครงสร้างแบบเกล็ดหิมะ (Snowflake Schema) ในการออกแบบแบบจำลองคลังข้อมูลนั้นยังคงกำหนดความสัมพันธ์ของตารางทั้ง 2 ประเภทนี้เป็นแบบหนึ่งต่อหนึ่งหรือแบบหนึ่งต่อกลุ่มตามหลักการออกแบบฐานข้อมูล

**1. โครงสร้างแบบดาว**

โครงร้างแบบดาวเป็นการแสดงรูปแบบเชิงโครงสร้างทางตรรกะที่ประกอบด้วยข้อมูลที่เป็นข้อเท็จจริงจัดเก็บอยู่ในตารางข้อเท็จจริง นิยมแสดงไว้ที่จุดกึ่งกลางและล้อมด้วยข้อมูลมิติซึ่งจัดเก็บอยู่ในตารางมิติ ซึ่งตารางมิติจะมีจำนวนเท่าใดก็ได้ และจะมีคีย์ที่สัมพันธ์ไปยังตารางข้อเท็จจริงเท่านั้น โครงสร้างชนิดนี้จะช่วยเพิ่มความเร็วในการสืบค้นข้อมูลเนื่องจากความสัมพันธ์ระหว่างตารางไม่ซับซ้อนแสดงดังภาพ 5



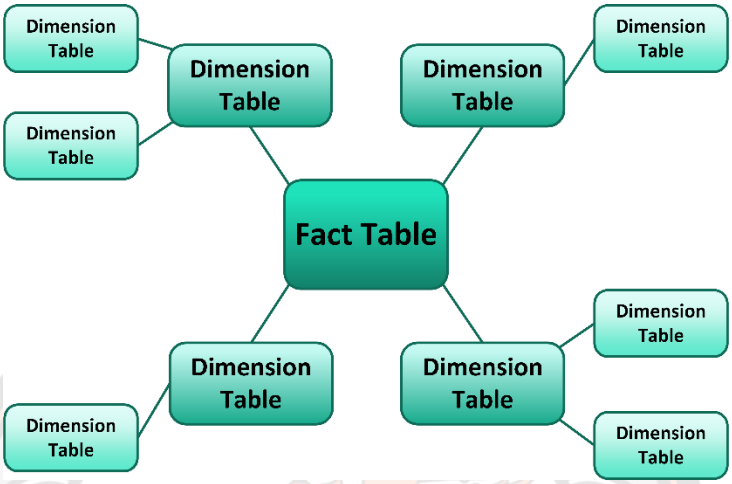

**ภาพ 5 แสดงตัวอย่างโครงสร้างแบบดาว**

ตารางข้อเท็จจริงใช้จัดเก็บข้อมูลที่เกิดขึ้นจริงในอดีตจนถึงปัจจุบัน และจัดเก็บข้อมูลส่วนสำคัญของคลังข้อมูล เช่น จำนวนผู้ป่วย ยอดขาย ผลผลิตทางการเกษตร เป็นต้น ซึ่งมีการจัดเก็บข้อมูลไว้เป็นเวลานานเพื่อสามารถนำข้อมูลกลับมาวิเคราะห์ใหม่ได้ เช่น จัดเก็บข้อมูลจำนวนผู้ป่วยไว้เป็นระยะ 5 ปี เป็นต้น ส่วนตารางมิติเป็นคุณลักษณะที่ใช้อธิบายข้อมูลในตารางข้อเท็จจริง เช่น ช่วงเวลา สถานที่ ช่วงอายุ เป็นต้น เพื่อใช้ในการออกรายงาน เช่น การวิเคราะห์ข้อมูลผู้ป่วยตามฤดูกาล การวิเคราะห์จำนวนผู้ป่วยตามสถานที่ที่พบผู้ป่วย เป็นต้น โครงสร้างแบบดาวนี้จะช่วยเพิ่มความสามารถในการสอบถามข้อมูลโดยลดปริมาณข้อมูลที่ต้องอ่านจากแหล่งเก็บข้อมูลที่มีจำนวนมากจากจำนวนตารางที่น้อยทำให้ลดจำนวนข้อมูลที่ต้องตรวจสอบ จึงสมารถสอบถามข้อมูลได้ง่ายและรวดเร็ว

**2. โครงสร้างแบบเกล็ดหิมะ**

โครงสร้างแบบเกล็ดหิมะ (Snowflake Schema) เป็นการแสดงรูปแบบเชิงโครงสร้างทางตรรกะอีกแบบหนึ่งที่ประกอบด้วยข้อมูลที่เป็นตารางข้อเท็จจริงและข้อมูลตารางมิติเช่นเดียวกับโครงสร้างแบบดาว แต่ตารางมิติจะมีความซับซ้อนมากขึ้นจึงต้องจัดเก็บข้อมูลเป็นลำดับชั้น

(Hierarchy) คือจะมีตารางมิติที่ไม่ได้เชื่อมต่อโดยตรงกับตารางข้อเท็จจริงแต่จะมีความสัมพันธ์กับตารางมิติอื่น ดังภาพ 6



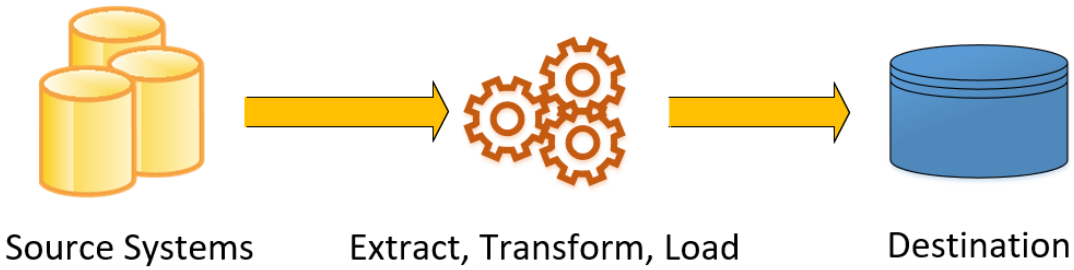

**ภาพ 6 แสดงตัวอย่างโครงสร้างแบบเกล็ดหิมะ**

โครงสร้างแบบเกล็ดหิมะตารางมิติจะมีการเก็บอยู่ในรูปแบบบรรทัดฐาน (Normal form) เนื่องจากขึ้นอยู่กับลักษณะของธุรกิจ หรือไม่สามารถออกแบบให้มีโครงสร้างแบบดาวได้ ส่งผลให้ประสิทธิภาพในการคิวรี่ข้อมูลลดลงเมื่อเปรียบเทียบกับโครงสร้างแบบดาว

เมื่อผู้ออกแบบคลังข้อมูลรวบรวมความต้องการใช้คลังข้อมูลจากผู้ใช้ระบบแล้ว ผู้ออกแบบคลังข้อมูลต้องสามารถวิเคราะห์ความต้องการของผู้ใช้ และแสดงออกมาในรูปแบบของรายงานต่าง ๆ โดยเลือกโครงสร้างที่เหมาะสมเพื่อนำไปใช้ในการกำหนดโครงสร้างของคลังข้อมูลและดาตามาร์ทในระบบคลังข้อมูลขององค์กรต่อไป

**กระบวนการนำเข้าข้อมูลสู่คลังข้อมูล (Extraction, Transformation and Load)**

กระบวนการนำเข้าข้อมูลสู่คลังข้อมูล คือ กระบวนการรวบรวมข้อมูลจากแหล่งข้อมูลต่าง ๆ เข้าสู่คลังข้อมูล โดยมีการแปลงข้อมูลจากหลายแหล่งข้อมูลให้อยู่ในรูปแบบที่สามารถใช้งานร่วมกันได้ และมีมาตรฐานเดียวกัน โดยขั้นตอนการทำงานสามารถแบ่งออกเป็น 4 ขั้นตอน ดังนี้




**ภาพ 7 แสดงกระบวนการทำงานในการนำข้อมูลเข้าสู่คลังข้อมูล**

1. ออกแบบกระบวนการทำงาน (Designing ETL Process) คือขั้นตอนของการออกแบบโครงสร้างของคลังข้อมูล กำหนดฐานข้อมูลที่จะทำการนำเข้าข้อมูล และกำหนดโครงร่างของข้อมูล เพื่อนำไปเป็นข้อกำหนดในการโอนถ่ายข้อมูลที่จำเป็นเพื่อนำเข้าสู่คลังข้อมูล ขั้นตอนการออกแบบนี้จะส่งผลต่อการนำข้อมูลในคลังข้อมูลไปใช้งาน

2. การดึงข้อมูล (Extract) คือขั้นตอนของการรวบรวม และดึงข้อมูลจากแหล่งข้อมูลต่าง ๆ ทั้งข้อมูลที่เป็นปัจจุบันและข้อมูลที่เกิดขึ้นในอดีตเข้ามาเก็บไว้ในพื้นที่พักข้อมูล สามารถดึงข้อมูลที่อยู่ในรูปแบบของฐานข้อมูลต่างชนิดกัน หรือรูปแบบที่ไม่ใช่ฐานข้อมูลซึ่งอาจจะเป็นระบบไฟล์ข้อมูลธรรมดา (Flat Files) หรือ ในอีกกรณี คือ เป็นข้อมูลในฐานข้อมูลที่ไม่ใช่รูปแบบ RDBMS (Relational Database Management System) เป้าหมายของกระบวนการนี้ คือ ดึงข้อมูลเข้ามาสู่รูปแบบมาตรฐานเดียวกัน เพื่อให้เหมาะสมต่อการแปลงรูปแบบข้อมูลในขั้นตอนถัดไป

3. การแปลงรูปแบบข้อมูล (Transform) คือ ขั้นตอนของการนำเอาข้อมูลที่ได้มาจากขั้นตอนของการดึงข้อมูลนั้นมาทำการจัดรูปแบบให้ถูกต้อง สอดคล้องกัน โดยจะมีกระบวนการย่อยที่เกิดขึ้นหลายกระบวนการดังนี้

3.1 Aggregation คือ การรวมกันของข้อมูล หรือการเชื่อมต่อกันของข้อมูล โดยจะเป็นการนำเอาข้อมูลในหลาย ๆ ส่วน มาวิเคราะห์เพื่อให้ได้ข้อมูลใหม่

3.2 Filtering คือ การกรองข้อมูลที่ไม่ได้อยู่ในเงื่อนไข หรือข้อมูลที่มีค่าของข้อมูลนอกขอบเขตข้อมูลที่เป็นไปได้ ทิ้งไป

3.3 Data mapping คือ การกำหนด และปรับเปลี่ยนรูปแบบของข้อมูลที่มีความหมายเดียวกัน ให้อยู่ในรูปแบบเดียวกัน เพื่อให้ข้อมูลที่มาจากแหล่งข้อมูลที่แตกต่างกันนั้นสามารถใช้งานร่วมกันได้

4. การนำเข้าข้อมูล (Load) คือ ขั้นตอนของการนำข้อมูลที่ผ่านการแปลงรูปแบบข้อมูลมาแล้ว นำเข้าสู่คลังข้อมูล

**แนวคิดและทฤษฎีการวิเคราะห์และประมวลผลออนไลน์**

การวิเคราะห์และประมวลผลออนไลน์ (Online Analytical Processing) เป็นเทคโนโลยีที่ช่วยดึงข้อมูลและนำข้อมูลมาเสนอในรูปแบบหลายมิติ (Multidimensional) สามารถแสดงผลได้หลากหลายมุมมองโดย OLAP ได้รับการออกแบบมาสำหรับหน่วยงาน หรือองค์กรที่ต้องการวิเคราะห์ข้อมูลเพื่อใช้ประกอบการตัดสินใจในระดับสูง (Chaudhuri & Dayal, 1997) นอกจากนี้ OLAP ยังเป็นเครื่องมือที่มีประสิทธิภาพสำหรับการสรุปผลในภาพรวม สามารถวิเคราะห์ เปรียบเทียบและนำเสนอข้อมูลตามรูปแบบที่ผู้ใช้ต้องการ

การดำเนินการของ OLAP แบ่งได้เป็น 4 ลักษณะ คือ

1. Roll up เป็นการปรับระดับความละเอียดของการพิจารณาข้อมูลจากระดับที่มีความละเอียดมากมาเป็นข้อมูลสรุป

2. Drill Down เป็นการปรับระดับความละเอียดของการพิจารณาข้อมูลจากข้อมูลสรุปมาเป็นข้อมูลที่แสดงถึงรายละเอียด

3. Slice เป็นการเลือกพิจารณาผลลัพธ์บางส่วนที่สนใจ โดยเลือกเฉพาะค่าที่ถูกกำกับด้วยข้อมูลบางค่าของแต่ละมิติเท่านั้น

4. Dice เป็นกระบวนการพลิกแกนหรือมิติของข้อมูล ให้ได้มุมมองที่แตกต่างออกไปและตรงตามความต้องการของผู้ใช้คลังข้อมูล

**การอนุมานชื่อคอลัมน์**

การสร้างโครงสร้างแบบหลายมิติจากข้อมูลแบบกึ่งโครงสร้างจากรูปแบบไฟล์ .CSV หรือข้อมูลแบบมีโครงสร้างในรูปแบบไฟล์ตารางคำนวณ ออนโทโลยีมีบทบาทสำคัญในการพิจารณาว่าคอลัมน์ใดควรจะอยู่ในตารางเดียวกัน โดยพิจารณาจากความสัมพันธ์ของข้อมูลในออนโทโลยีนั้น แต่ในบางกรณีข้อมูลแบบกึ่งโครงสร้างไม่มีการระบุชื่อคอลัมน์มา หรือข้อมูลชื่อคอลัมน์มีการสูญหาย จำเป็นต้องมีเครื่องมือเพื่อช่วยในการพิจารณาว่าข้อมูลที่อยู่ในคอลัมน์นั้นคือข้อมูลอะไร งานวิจัยนี้เสนอขั้นตอนการอนุมานชื่อคอลัมน์โดยใช้ฟังก์ชันความหนาแน่นของความน่าจะเป็น (Probability density function) และการเข้ารหัสเลขคณิต (Arithmetic encoding function)

**ฟังก์ชันความหนาแน่นของความน่าจะเป็น (Probability Density Function)**

เป็นการประมาณค่าความหนาแน่นในทางสถิติและทางคณิตศาสตร์ การประมาณค่าความหนาแน่น มีการนำมาใช้อย่างแพร่หลาย จากข้อมูลกำหนดให้ $x_1, x_2, ..., x_n$ เป็นตัวอย่างที่อิสระและมีการแจกแจงเหมือนกันที่มาจากฟังก์ชันความหนาแน่น $f$ ดังนั้น ค่าประมาณความหนาแน่นของความน่าจะเป็น $f$ แสดงดังสมการ (1)

$$f_h(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{\bar{x} - x_i}{h}\right) \qquad (1)$$

จากสมการกำหนดให้

$h$ คือ ค่าพารามิเตอร์ปรับเรียบหรือแบนวิดจ์ (Smoother parameter)

$n$ คือ จำนวนข้อมูลทั้งหมด

$\bar{x}$ คือ ค่าเฉลี่ยของตัวแปรสุ่ม

$x_i$ คือ ค่าของตัวแปรสุ่ม $x$ โดยที่ $i$ = 1, 2, .., $n$

$K$ คือ ฟังก์ชันความหนาแน่นของความน่าจะเป็น

ฟังก์ชันความหนาแน่นที่ใช้กันทั่วไปมีหลายรูปแบบ เช่น Uniform, Triangular, Epanechnikov, Quartic, Triweight, Gaussian และ Cosine ฯลฯ

**ฟังก์ชันความหนาแน่นที่ใช้ในการวิจัย**

ฟังก์ชันความหนาแน่นแบบเกาส์เซียน (Gaussian) เป็นรูปแบบการแจกแจงข้อมูลตามความน่าจะเป็นของตัวแปรที่สนใจ ซึ่งเป็นพื้นฐานที่สำคัญทางสถิติ จึงมักถูกนำมาใช้บ่อยครั้ง ดังแสดงในสมการ (2)

$$K(a) = \frac{1}{\sqrt{2\pi}} e^{\left(-\frac{a^2}{2}\right)} \qquad (2)$$

จึงได้ฟังก์ชันความหนาแน่นของความน่าจะเป็น (Probability Density Function) มาตรฐานปกติ โดยที่ $a = \frac{\bar{x} - x_i}{h}$ ดังสมการที่ (3)

$$K(a) = \frac{1}{h\sqrt{2\pi}} e^{-\frac{(\bar{x} - x_i)^2}{2h^2}} \qquad (3)$$

จากสมการกำหนดให้

$h$ คือ ค่าพารามิเตอร์ปรับเรียบหรือแบนวิดจ์

$\bar{x}$ คือ ค่าเฉลี่ยของตัวแปรสุ่ม

$x_i$ คือ ค่าของตัวแปรสุ่ม $x$ โดยที่ $i$ = 1, 2, .., $n$

และจากการศึกษาของ Bowman & Azzalini (1997) นำเสนอการประมาณค่าพารามิเตอร์จากขนาดของข้อมูลตัวอย่างดังสมการ (4) ดังนี้

$$h = \sigma \times \left(\frac{4}{3 \times n}\right)^{\frac{1}{5}} \qquad (4)$$

จากสมการกำหนดให้

$\sigma$ คือ ส่วนเบี่ยงเบนมาตรฐาน

$n$ คือ จำนวนตัวอย่าง

**การเข้ารหัสเลขคณิต (Arithmetic Coding)**

กระบวนการแปลงข้อมูลตัวอย่างไปเป็นข้อมูลในรูปแบบอื่น ที่ทำให้ขนาดของข้อมูลลดลง แบ่งออกเป็น 2 ประเภทดังนี้

1. Lossless Compression เป็นอัลกอริทึมในการบีบอัดข้อมูลที่ไม่ทำให้ข้อมูลสูญหายในการบีบอัด ข้อมูลจึงมีความสมบูรณ์เหมือนต้นฉบับ ใช้สำหรับข้อมูลที่ไม่ต้องการสูญเสียรายละเอียดไป เช่น ข้อมูลรูปภาพ หรือข้อมูลทางการแพทย์

2. Lossy Compression เป็นอัลกอริทึมในการบีบอัดข้อมูลที่มีการตัดข้อมูลบางส่วนออกไปเพื่อลดขนาดของไฟล์ เมื่อทำการแปลงกลับข้อมูลที่ได้จะไม่เหมือนกับข้อมูลเดิม แต่มีอัตราส่วนการบีบอัดข้อมูลมากกว่าแบบ Lossless compression

การเข้ารหัสเลขคณิต (Arithmetic Coding) เป็นการเข้ารหัสที่มีประสิทธิภาพด้านค่าอัตราส่วนการบีบอัด (Compression Ratio: CR) สูง และเป็นการเข้ารหัสหรือการบีบอัดที่ไม่มีการสูญเสีย (Lossless Compression) (Howard & Vitter, 1992; Witten, Neal & Cleary, 1987) จึงได้รับความนิยมในการบีบอัดไฟล์ข้อมูลที่เป็นทั้งตัวหนังสือ และรูปภาพ โดยกำหนดช่วงของจำนวนจริงที่อยู่ระหว่าง 0 ถึง 1 เริ่มจากการกำหนดช่วงของข้อมูลดังสมการที่ (5)

$$\Phi_k(S) = |b, l\rangle \quad k = 0, 1, 2,...N \qquad (5)$$

จากสมการกำหนดให้

$S$ คือ ข้อมูลชนิดสตริง

$b$ คือ ค่าเริ่มต้น

$l$ คือ ช่วงของข้อมูล

จากข้อมูลตัวอย่างเมื่อทำการกำหนดช่วงของข้อมูลจะได้ดังสมการ (6) และ (7)

$$\Phi_0(S) = |b_0, l_0\rangle = |0,1\rangle \quad (6)$$

$$\Phi_k(S) = |b_k, l_k\rangle = |b_{k-1} + c(s_k)l_{k-1},\ p(s_k)l_{k-1}\rangle \quad (7)$$

จากสมการกำหนดให้

$c$ คือ ค่าการกระจายตัว

$p$ คือ ความน่าจะเป็นของข้อมูลชนิดสตริง ($S$)

โดยข้อมูลที่ได้จะอยู่ในช่วง $0 \leq b_k \leq b_{k+1} < 1$ และ $0 \leq l_{k+1} < l_k \leq 1$ งานวิจัยนี้ได้นำการเข้ารหัสเลขคณิตมาใช้เพื่อทำการแปลงข้อมูลที่อยู่ในรูปแบบตัวอักษรไปเป็นตัวเลขที่อยู่ระหว่าง 0 ถึง 1 ทำให้ได้ค่าตัวเลขที่ใช้แทนข้อมูลตัวอักษร ตัวเลขที่ได้จะนำมาใช้อนุมานชื่อคอลัมน์ด้วยฟังก์ชันความหนาแน่นของความน่าจะเป็น

**ระยะทางแบบยูคลิด (Euclidean Distance)**

ระยะทางแบบยูคลิด เป็นมาตรวัดระยะพื้นฐานใช้สำหรับหาระยะทางระหว่างจุดสองจุด คำนวณได้ตามสมการ (8)

$$d(p,q) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2} \quad (8)$$

จากสมการกำหนดให้

$d(p,q)$ คือ ระยะทางจากจุด $p$ ไปยังจุด $q$

$q$ , $p$ คือ จุดใด ๆ

$n$ คือ จำนวนมิติของข้อมูล

ค่าที่ได้จากฟังก์ชันความหนาแน่นของความน่าจะเป็นจากชุดทดสอบและชุดการสอนจะถูกนำมาคำนวณเพื่อหาความคล้ายคลึงกันโดยใช้ระยะทางแบบยุคลิด

**การอนุมานชนิดข้อมูล**

การอนุมานชนิดของข้อมูล (Data type) และขนาด (Size) ของแอตทริบิวต์เป็นอีกกระบวนการหนึ่งที่มีความสำคัญสำหรับการออกแบบคลังข้อมูลใน Microsoft SQL Server 2019 จากแนวคิดของ Hansen et al. (2017) และArmbrust et al. (2015) มีแนวคิดว่าชนิดข้อมูลที่เหมาะสมที่สุดของคอลัมน์ใดคอลัมน์หนึ่ง คือชนิดข้อมูลในโหนดที่ต่ำที่สุดของออนโทโลยีที่เข้ากันได้ (Fit) กับข้อมูลที่มีขนาดใหญ่ที่สุดในคอลัมน์นั้นๆ ของข้อมูลตัวอย่าง งานวิจัยนี้เสนอจึงได้เสนอขั้นตอนการอนุมานชนิดข้อมูลโดยใช้ฐานความรู้ออนโทโลยี โดยสร้างออนโทโลยีจากชนิดข้อมูลที่

ฐานข้อมูล Microsoft SQL Server 2019 รองรับ แบ่งออกเป็น 4 กลุ่มข้อมูลได้แก่ ข้อมูลที่เป็นตัวเลข ข้อมูลที่เป็นตัวอักษร ข้อมูลที่เป็นวันที่และเวลา และข้อมูลแบบบูลีน แสดงดังตาราง 2 ถึง 5

**ตาราง 2 แสดงชนิดข้อมูลประเภทตัวเลข**

| ชนิดข้อมูล | รายละเอียด | |
|---|---|---|
| | ตั้งแต่ | ถึง |
| bigint | -9,223,372,036,854,775,808 | 9,223,372,036,854,775,807 |
| int | -2,147,483,648 | 2,147,483,647 |
| smallint | -32,768 | 32,767 |
| tinyint | 0 | 255 |
| decimal | -10^38 +1 | 10^38 -1 |
| float | -1.79E + 308 | 1.79E + 308 |
| real | -3.40E + 38 | 3.40E + 38 |

**ตาราง 3 แสดงชนิดข้อมูลประเภทตัวอักษร**

| ชนิดข้อมูล | รายละเอียด |
|---|---|
| **Non Unicode** | |
| char | สูงสุด 8,000 ตัวอักษร ( จำนวนตัวอักษรต้องเท่ากัน ) |
| varchar | สูงสุด 8,000 ตัวอักษร ( จำนวนตัวอักษรต้องไม่เกิน ) |
| text | สูงสุด 2,147,483,647 ตัวอักษร ( จำนวนตัวอักษรต้องไม่เกิน) |
| **Unicode** | |
| nchar | สูงสุด 4,000 ตัวอักษร ( จำนวนตัวอักษรต้องเท่ากัน) |
| nvarchar | สูงสุด 4,000 ตัวอักษร ( จำนวนตัวอักษรต้องไม่เกิน ) |
| ntext | สูงสุด 1,073,741,823 ตัวอักษร ( จำนวนตัวอักษรต้องไม่เกิน) |

**ตาราง 4 แสดงชนิดข้อมูลประเภทวันที่และเวลา**

| ชนิดข้อมูล | รายละเอียด | |
|---|---|---|
| | ตั้งแต่ | ถึง |
| datetime | Jan 1, 1753 | Dec 31, 9999 |
| smalldatetime | Jan 1, 1900 | Jun 6, 2079 |

**ตาราง 5 แสดงชนิดข้อมูลประเภทบูลีน**

| ชนิดข้อมูล | รายละเอียด |
|---|---|
| Bit | ค่า 0 กับ 1 |

จากรายละเอียดชนิดข้อมูลสามารถแสดงคำศัพท์ที่เกี่ยวข้องและคำศัพท์ในแต่ละหัวข้อจะถูกกำหนดให้เป็นคลาสในออนโทโลยี ผู้วิจัยได้ทำการสร้างให้อยู่ในรูปแบบของออนโทโลยี แสดงถึงระดับของชนิดข้อมูลตามขนาดของชนิดข้อมูล โดยชนิดข้อมูลที่มีขนาดใหญ่กว่าจะเป็นคลาสแม่และมีข้อมูลที่มีขนาดเล็กกว่าเป็นคลาสลูก

# บทที่ 3

# วิธีดำเนินการวิจัย

การทำวิจัยในครั้งนี้ผู้วิจัยได้ทำการพัฒนาเทคนิคและกระบวนการสำหรับสร้างโครงสร้างแบบหลายมิติโดยอัตโนมัติ โดยมีรายละเอียดวิธีดำเนินการวิจัยดังนี้

1. การเก็บรวบรวมข้อมูล
2. เครื่องมือที่ใช้ในการพัฒนา
3. การออกแบบและพัฒนาออนโทโลยี
4. กรอบแนวคิดของงานวิจัย
5. การประเมินผล

**การเก็บรวบรวมข้อมูล**

จากการศึกษาเอกสารและงานวิจัยที่เกี่ยวข้องกับการสร้างโครงสร้างแบบหลายมิติโดยอัตโนมัติ พบว่ารูปแบบของข้อมูลตัวอย่างที่นำมาใช้มี 3 รูปแบบ คือ ข้อมูลที่มีโครงสร้าง (Structured data) แบบกึ่งโครงสร้าง (Semi-structure data) และแบบไร้โครงสร้าง (Unstructured data) ข้อมูลแบบไร้โครงสร้างต้องใช้เทคนิคทาง Natural Language Processing (NLP) มาช่วยจึงทำให้มีความซับซ้อนสูง แต่ในงานวิจัยนี้จะเน้นไปที่การใช้ข้อมูลแบบกึ่งโครงสร้างในรูปแบบไฟล์ .CSV ซึ่งเป็นข้อมูลที่มีโครงสร้างเฉพาะ ไม่ใช่โครงสร้างข้อมูลแบบในฐานข้อมูล ออนโทโลยีจึงมีบทบาทสำคัญในการพิจารณาว่าในแต่ละตารางควรมีคอลัมน์ใดบ้าง โดยพิจารณาจากความสัมพันธ์ของข้อมูลในออนโทโลยี โดยข้อมูลที่ใช้ในการศึกษานี้ประกอบด้วยข้อมูลจาก 3 โดเมน คือ โดเมนทางการแพทย์ โดเมนทางการเกษตร และโดเมนทางธุรกิจ

โดเมนทางการแพทย์เป็นข้อมูลการระบาดของโรคไข้เลือดออกในประเทศไทย ในปี พ.ศ. 2546 ถึง พ.ศ. 2560 จำนวน 23 แอตทริบิวต์ 13,764 ระเบียน จากกรมควบคุมโรคและหน่วยวิจัยชีววิทยาและแมลงพาหะนำโรค จุฬาลงกรณ์มหาวิทยาลัย แสดงตัวอย่างข้อมูลดังภาพ 8

| | A | B | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|---|
| 1 | ID | Year | SeasonName | MonthName | RegionKey | Region | Province | Age | Case |
| 2 | 1 | 2560 | winter | January | 3 | Northeastern | Amnat Charoen | 1 | 0 |
| 3 | 2 | 2560 | winter | January | 3 | Northeastern | Amnat Charoen | 1 | 1 |
| 4 | 3 | 2560 | winter | January | 3 | Northeastern | Amnat Charoen | 1 | 0 |
| 5 | 4 | 2560 | winter | January | 3 | Northeastern | Amnat Charoen | 2 | 0 |
| 6 | 5 | 2560 | winter | January | 3 | Northeastern | Amnat Charoen | 3 | 1 |
| 7 | 6 | 2560 | winter | January | 3 | Northeastern | Amnat Charoen | 4 | 5 |
| 8 | 7 | 2560 | winter | January | 3 | Northeastern | Amnat Charoen | 5 | 6 |
| 9 | 8 | 2560 | winter | January | 3 | Northeastern | Amnat Charoen | 6 | 4 |
| 10 | 9 | 2560 | winter | January | 3 | Northeastern | Amnat Charoen | 7-9 | 7 |
| 11 | 10 | 2560 | winter | January | 3 | Northeastern | Amnat Charoen | 10-14 | 23 |
| 12 | 11 | 2560 | winter | January | 3 | Northeastern | Amnat Charoen | 15-24 | 16 |
| 13 | 12 | 2560 | winter | January | 3 | Northeastern | Amnat Charoen | 25-34 | 8 |

**ภาพ 8 แสดงตัวอย่างข้อมูลโดเมนทางการแพทย์**

โดเมนทางการเกษตรเป็นข้อมูลผลผลิตข้าว ในปี พ.ศ. 2552 ถึง พ.ศ. 2560 จำนวน 6 แอตทริบิวต์ 4,544 ระเบียน จากสำนักงานเศรษฐกิจการเกษตร แสดงตัวอย่างข้อมูลดังภาพ 9

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | ID | Year | Region | Province | RiceVariety | RiceYield |
| 2 | 1 | 2560 | Southern | Krabi | Native rice | 474 |
| 3 | 2 | 2560 | Southern | Krabi | Khao Dawk Mali 105 | 119 |
| 4 | 3 | 2560 | Southern | Krabi | Photoperiod sensitivity Rice | 1,095 |
| 5 | 4 | 2560 | Southern | Krabi | Pathum Thani 1 | 41 |
| 6 | 5 | 2560 | Central | Bangkok | Non-photoperiod sensitivity Rice | 42,140 |
| 7 | 6 | 2560 | Central | Bangkok | Pathum Thani 1 | 11,714 |
| 8 | 7 | 2560 | Central | Bangkok | Suphan Buri 1 | 4,377 |
| 9 | 8 | 2560 | Western | Kanchanaburi | Native rice | 11,343 |
| 10 | 9 | 2560 | Western | Kanchanaburi | Khao Dawk Mali 105 | 33,968 |
| 11 | 10 | 2560 | Western | Kanchanaburi | Photoperiod sensitivity Rice | 2,944 |
| 12 | 11 | 2560 | Western | Kanchanaburi | Non-photoperiod sensitivity Rice | 46,853 |

**ภาพ 9 แสดงตัวอย่างข้อมูลโดเมนทางการเกษตร**

และโดเมนทางธุรกิจเป็นข้อมูลการขายจากฐานข้อมูล AdventureWorks จำนวน 12 แอตทริบิวต์ 60,855 ระเบียน ของบริษัทไมโครซอฟท์ แสดงตัวอย่างข้อมูลดังภาพ 10

| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 1 | ID | Year | Month | ResellerKey | Reseller | Country | State | City |
| 2 | 1 | 2553 | December | 676 | Better Bike Shop | United States | Georgia | Austell |
| 3 | 2 | 2553 | December | 676 | Better Bike Shop | United States | Georgia | Austell |
| 4 | 3 | 2553 | December | 676 | Better Bike Shop | United States | Georgia | Austell |
| 5 | 4 | 2553 | December | 676 | Better Bike Shop | United States | Georgia | Austell |
| 6 | 5 | 2553 | December | 676 | Better Bike Shop | United States | Georgia | Austell |
| 7 | 6 | 2553 | December | 676 | Better Bike Shop | United States | Georgia | Austell |
| 8 | 7 | 2553 | December | 676 | Better Bike Shop | United States | Georgia | Austell |
| 9 | 8 | 2553 | December | 676 | Better Bike Shop | United States | Georgia | Austell |
| 10 | 9 | 2553 | December | 676 | Better Bike Shop | United States | Georgia | Austell |
| 11 | 10 | 2553 | December | 676 | Better Bike Shop | United States | Georgia | Austell |
| 12 | 11 | 2553 | December | 676 | Better Bike Shop | United States | Georgia | Austell |
| 13 | 12 | 2553 | December | 676 | Better Bike Shop | United States | Georgia | Austell |
| 14 | 13 | 2553 | December | 117 | Pedals Warehouse | United States | Georgia | Suwanee |
| 15 | 14 | 2553 | December | 117 | Pedals Warehouse | United States | Georgia | Suwanee |
| 16 | 15 | 2553 | December | 442 | Original Bicycle Supply Company | Canada | Ontario | Toronto |
| 17 | 16 | 2553 | December | 442 | Original Bicycle Supply Company | Canada | Ontario | Toronto |
| 18 | 17 | 2553 | December | 442 | Original Bicycle Supply Company | Canada | Ontario | Toronto |
| 19 | 18 | 2553 | December | 442 | Original Bicycle Supply Company | Canada | Ontario | Toronto |
| 20 | 19 | 2553 | December | 442 | Original Bicycle Supply Company | Canada | Ontario | Toronto |

**ภาพ 10 แสดงตัวอย่างข้อมูลโดเมนทางธุรกิจ**

**เครื่องมือที่ใช้ในการพัฒนา**

เครื่องมือที่ใช้ในการพัฒนาการสร้างโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติ แบ่งเป็น 3 ประเภท ได้แก่ ซอฟต์แวร์ ฮาร์ดแวร์ และระบบปฏิบัติการ โดยมีรายละเอียดดังนี้

1. ซอฟต์แวร์
   1.1 โปรแกรมฐานข้อมูล: Microsoft SQL Server 2019
   1.2 โปรแกรมพัฒนาออนโทโลยี: Hozo
   1.3 ภาษาพัฒนาระบบ: Python
2. ฮาร์ดแวร์
   2.1 หน่วยประมวลผล (CPU) Intel Core i3-4005U 1.7 GHz
   2.2 หน่วยความจำหลัก (RAM) DDR 6 GB
3. ระบบปฏิบัติการ
   ระบบปฏิบัติการ: Microsoft Windows 10 64-bit

**การออกแบบและพัฒนาออนโทโลยี**

การพัฒนาฐานความรู้ออนโทโลยีจะอ้างอิงตามทฤษฎีกระบวนการพัฒนาออนโทโลยี โดยนำหลักการพัฒนาฐานความรู้ออนโทโลยีจากบนลงล่าง (Top-Down) มาใช้สำหรับการสร้างต้นแบบออนโทโลยีโดยมีขั้นตอนดังนี้

1. การกำหนดขอบเขตข้อมูล
2. การกำหนดคำศัพท์ภายใต้ขอบเขตข้อมูล
3. การกำหนดโครงสร้างออนโทโลยี
4. การกำหนดคุณสมบัติของคลาสในโครงสร้างออนโทโลยี

โดยมีรายละเอียดของแต่ละขั้นตอนดังนี้

**1. การกำหนดขอบเขตฐานความรู้ออนโทโลยี**

การกำหนดขอบเขตของฐานความรู้ออนโทโลยี ผู้วิจัยใช้โปรแกรม Hozo เป็นเครื่องมือในการสร้างออนโทโลยี โดยแบ่งออกเป็น 4 ออนโทโลยี คือ โดเมนออนโทโลยีการระบาดของโรคไข้เลือดออก โดเมนออนโทโลยีผลผลิตข้าวนาปี โดเมนออนโทโลยีการขาย และออนโทโลยีชนิดข้อมูล

**2. การกำหนดคำศัพท์ภายใต้ขอบเขตข้อมูล**

การกำหนดคำศัพท์เป็นการพิจารณาถึงสิ่งที่เกี่ยวข้อง ที่อยู่ภายใต้ขอบเขตของข้อมูลที่นำมาสร้างออนโทโลยี จากการพิจารณาข้อมูลจากทั้งสามโดเมนพบว่าทุกโดเมนประกอบด้วยประเด็นหลักเหมือนกัน เช่น สถานที่ และเวลา เป็นต้น ดังนั้นการกำหนดคำศัพท์ที่เกี่ยวข้องที่อยู่ภายใต้ขอบเขตของข้อมูลจะพิจารณาจากข้อมูลที่ได้จากเอกสารที่เกี่ยวข้อง โดยจำแนกออกเป็น 2 ลักษณะ คือ หัวข้อที่ประกอบด้วยหัวข้อย่อย และหัวข้อที่ไม่มีหัวข้อย่อย จากการจำแนกลักษณะจะนำมาสร้างเป็นคลาสดังแสดงคำศัพท์ที่เกี่ยวข้องในตาราง 6 ถึง 9 คำศัพท์ในแต่ละหัวข้อจะกำหนดให้เป็นคลาส

**ตาราง 6 แสดงคำศัพท์ที่เกี่ยวข้องกับข้อมูลโรคไข้เลือดออก**

| ข้อมูล | คำศัพท์ |
|---|---|
| สถานที่ | Location |
| ภูมิภาค | Region |
| ภาคเหนือ | Northern |
| ภาคกลาง | Central |
| ภาคตะวันออกเฉียงเหนือ | Northeastern |

**ตาราง 6 (ต่อ)**

| ข้อมูล | คำศัพท์ |
|---|---|
| ภาคตะวันตก | Western |
| ภาคตะวันออก | Eastern |
| ภาคใต้ | Southern |
| จังหวัด | Province |
| เวลา | Time |
| ปี | Year |
| ฤดูกาล | Season |
| เดือน | Month |
| อายุ | Age |
| การระบาดของโรคไข้เลือดออก | DengueFever |

**ตาราง *7* แสดงคำศัพท์ที่เกี่ยวข้องกับข้อมูลผลผลิตข้าว**

| ข้อมูล | คำศัพท์ |
|---|---|
| สถานที่ | Location |
| ภูมิภาค | Region |
| ภาคเหนือ | Northern |
| ภาคกลาง | Central |
| ภาคตะวันออกเฉียงเหนือ | Northeastern |
| ภาคตะวันตก | Western |
| ภาคตะวันออก | Eastern |
| ภาคใต้ | Southern |
| จังหวัด | Province |
| เวลา | Time |
| ปี | Year |
| พันธุ์ข้าว | RiceVariety |
| พันธุ์พื้นเมือง | Native rice |

**ตาราง 7 (ต่อ)**

| ข้อมูล | คำศัพท์ |
|---|---|
| กข.6 | RD6 |
| กข.15 | RD15 |
| ขาวดอกมะลิ 105 | Khao Dawk Mali 105 |
| สุพรรณบุรี 1 | Suphan Buri 1 |
| สุพรรณบุรี 60,90 | Suphan Buri 60,90 |
| ราชการไวต่อแสง | Photoperiod sensitivity Rice |
| ราชการไม่ไวต่อแสง | Non-photoperiod sensitivity Rice |
| ชัยนาท 1 | Chai Nat 1 |
| หอมสุพรรณบุรี | Khao Jow Hawm Suphan Buri |
| ปทุมธานี 1 | Pathum Thani 1 |
| คลองหลวง 1 | Khlong Luang 1 |
| ผลผลิตข้าว | Rice Yield |

**ตาราง 8 แสดงคำศัพท์ที่เกี่ยวข้องกับข้อมูลการขาย**

| ข้อมูล | คำศัพท์ |
|---|---|
| Location | Location |
| Country | Country |
| Australia | Australia |
| Canada | Canada |
| France | France |
| Germany | Germany |
| United Kingdom | United Kingdom |
| United States | United States |
| State | State |
| City | City |
| Time | Time |

**ตาราง 8 (ต่อ)**

| ข้อมูล | คำศัพท์ |
|---|---|
| Year | Year |
| Month | Month |
| Product | Product |
| ProductCategory | ProductCategory |
| ProductSubcategory | ProductSubcategory |
| ProductName | ProductName |
| Reseller | Reseller |
| Sale | Sale |

**ตาราง 9 แสดงคำศัพท์ที่เกี่ยวข้องกับชนิดข้อมูล**

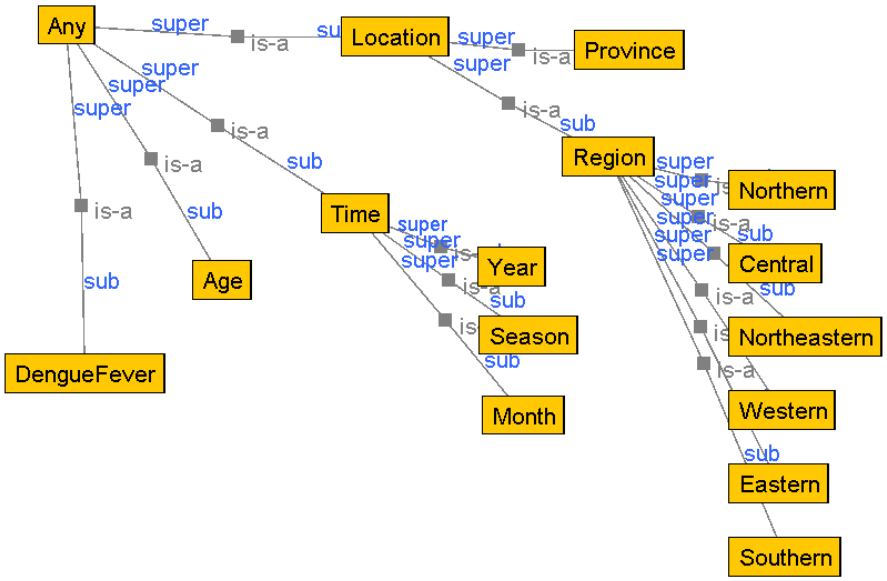
| ข้อมูล | คำศัพท์ |
|---|---|
| String | String |
| Boolean | Boolean |
| DateTime | DateTime |
| Decimal | Decimal |
| bigint | bigint |
| int | int |
| smallint | smallint |
| tinyint | tinyint |
| decimal | decimal |
| float | float |
| real | real |
| Non-Unicode | Non-Unicode |
| char | char |
| varchar | varchar |
| text | text |

**ตาราง 9 (ต่อ)**

| ข้อมูล | คำศัพท์ |
|---|---|
| Unicode | Unicode |
| nchar | nchar |
| nvarchar | nvarchar |
| ntext | ntext |
| datetime | datetime |
| smalldatetime | smalldatetime |
| Bit | Bit |

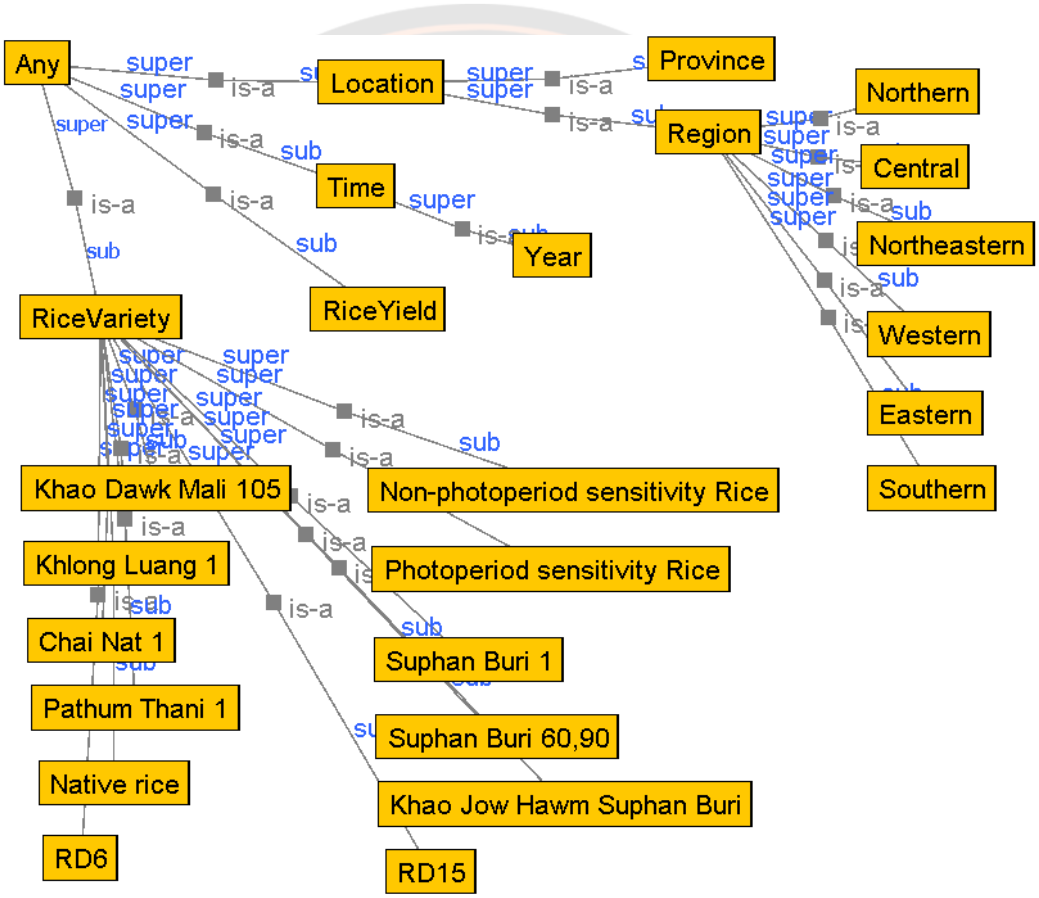
**3. การกำหนดโครงสร้างออนโทโลยี**

จากตารางคำศัพท์ที่เกี่ยวข้องกับข้อมูลโรคไข้เลือดออกพบว่าคำศัพท์ที่มีความสัมพันธ์ในลักษณะลำดับชั้น คือ คลาสสถานที่ (Location) แบ่งเป็น ภูมิภาค (Region) และจังหวัด (Province) คลาสภูมิภาค (Region) แบ่งเป็น ภาคเหนือ (Northern) ภาคกลาง (Central) ภาคตะวันออกเฉียงเหนือ (Northeastern) ภาคตะวันตก (Western) ภาคตะวันออก (Eastern) และภาคใต้ (Southern) คลาสเวลา (Time) แบ่งเป็น ปี (Year) ฤดูกาล (Season) และเดือน (Month) การกำหนดคลาสของการระบาดของโรคไข้เลือดออกแบ่งออกเป็นคลาสได้ทั้งหมด 15 คลาสดังภาพ 11




**ภาพ 11 แสดงคลาสการระบาดของโรคไข้เลือดออก**

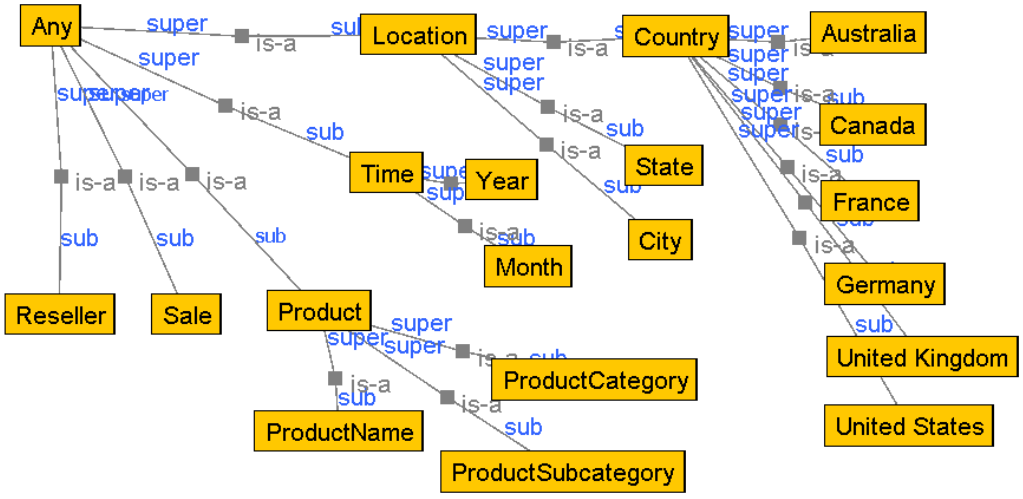
จากตารางคำศัพท์ที่เกี่ยวข้องกับข้อมูลผลผลิตข้าวนาปีพบว่าคำศัพท์ที่มีความสัมพันธ์ในลักษณะลำดับชั้น คือ คลาสสถานที่ (Location) แบ่งเป็น ภูมิภาค (Region) และจังหวัด (Province) คลาสภูมิภาค (Region) แบ่งเป็น ภาคเหนือ (Northern) ภาคกลาง (Central) ภาคตะวันออกเฉียงเหนือ (Northeastern) ภาคตะวันตก (Western) ภาคตะวันออก (Eastern) และภาคใต้ (Southern) คลาสเวลา (Time) แบ่งเป็น ปี (Year) คลาสพันธ์ข้าว (RiceVariety) แบ่งเป็นพันธ์ข้าวทั้งหมด 12 สายพันธ์ การกำหนดคลาสของข้อมูลผลผลิตข้าวนาปีแบ่งออกเป็นคลาสได้ทั้งหมด 25 คลาสแสดงดังแสดงในภาพ 12




**ภาพ 12 แสดงคลาสข้อมูลผลผลิตข้าว**

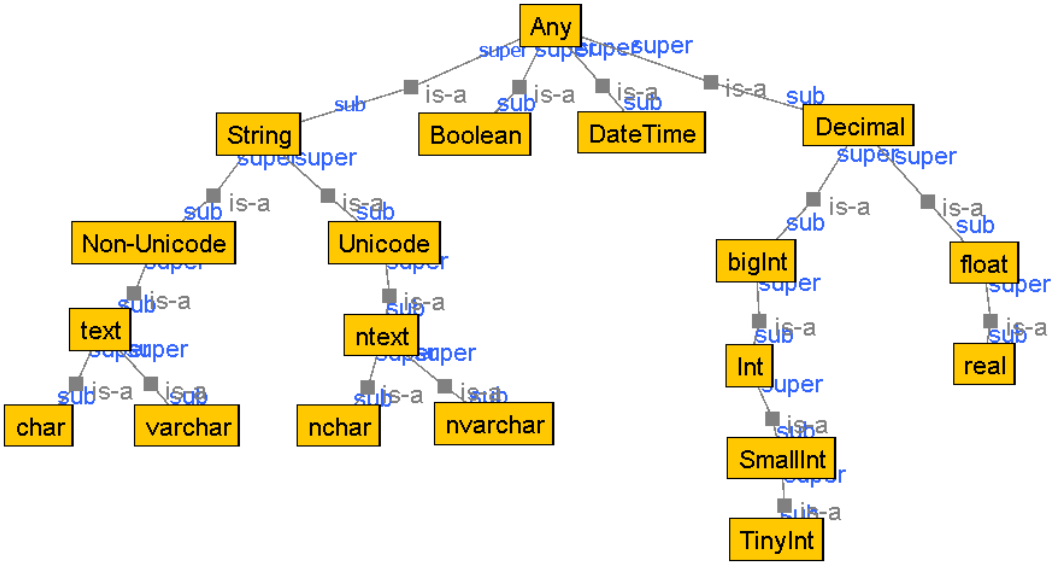
จากตารางคำศัพท์ที่เกี่ยวข้องกับข้อมูลการขายพบว่าคำศัพท์ที่มีความสัมพันธ์ในลักษณะลำดับชั้น คือ คลาส Location แบ่งเป็น Country, State และ City คลาส Country แบ่งเป็น Australia, Canada, France, Germany, United Kingdom และ United States คลาส Product แบ่งเป็น ProductCategory, ProductSubcategory และ ProductName คลาส Time

แบ่งเป็น Year และ Month การกำหนดคลาสของข้อมูลการขายแบ่งออกเป็นคลาสได้ทั้งหมด 19 คลาสแสดงดังภาพ 13




**ภาพ 13 แสดงคลาสข้อมูลการขาย**

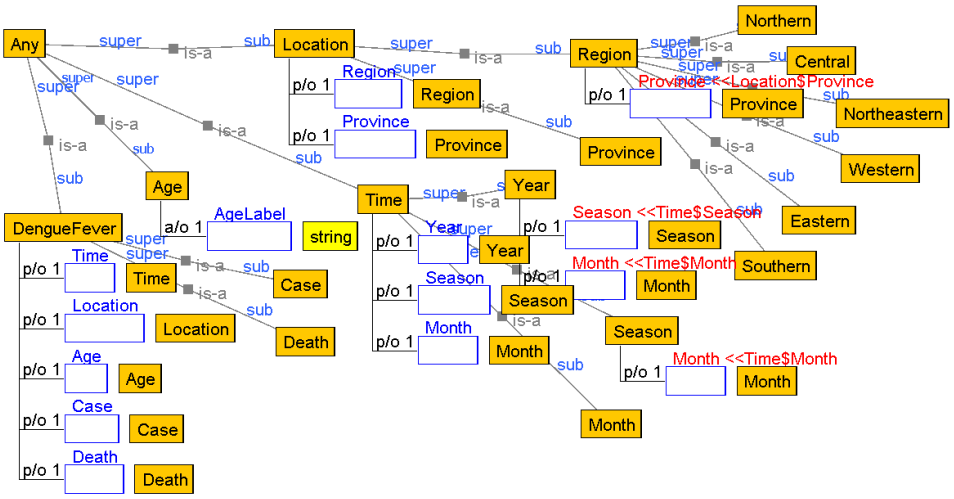
จากตารางคำศัพท์ที่เกี่ยวข้องกับรายละเอียดชนิดข้อมูล ผู้วิจัยได้ทำการสร้างให้อยู่ในรูปแบบของออนโทโลยี แสดงถึงระดับของชนิดข้อมูลตามขนาดของชนิดข้อมูล โดยชนิดข้อมูลที่มีขนาดใหญ่กว่าจะเป็นคลาสแม่และมีข้อมูลที่มีขนาดเล็กกว่าเป็นคลาสลูกดังภาพ 14



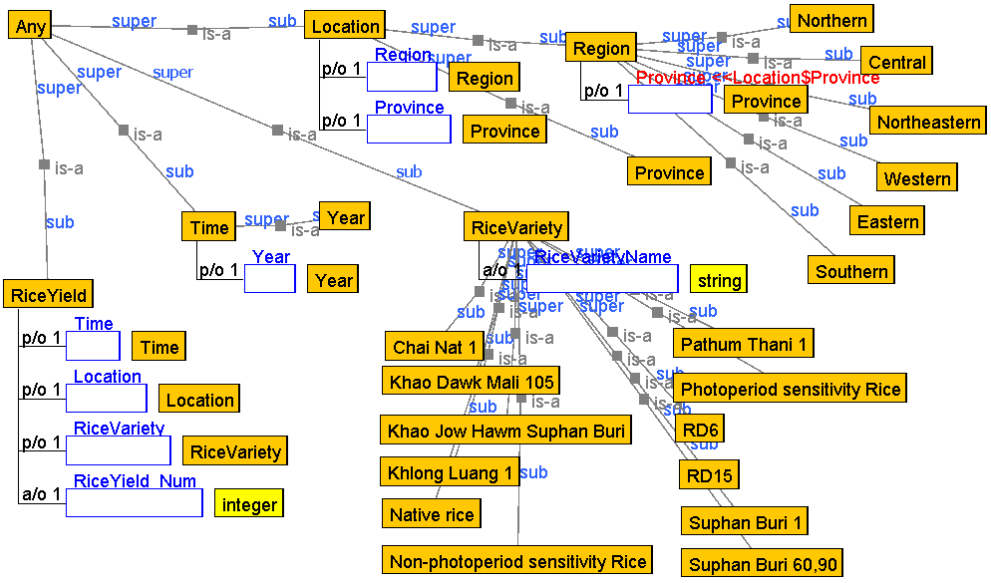

**ภาพ 14 แสดงคลาสชนิดข้อมูล**

**4. การกำหนดคุณสมบัติของคลาสในโครงสร้างออนโทโลยี**

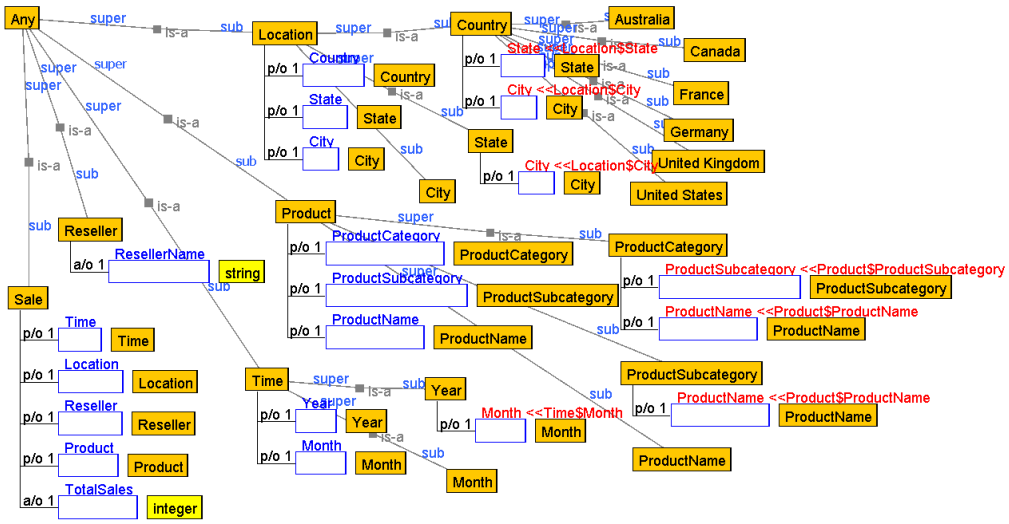
คุณสมบัติสามารถแบ่งออกเป็น 2 แบบ คือคุณสมบัติของวัตถุ (Object Property) และคุณสมบัติของข้อมูล (Data Property) สำหรับคุณสมบัติของวัตถุเป็นความสัมพันธ์ที่เชื่อมโยงระหว่างคลาส 2 คลาส หรือระหว่างวัตถุกับวัตถุ และคุณสมบัติของข้อมูลเป็นการอธิบายคุณลักษณะเพิ่มเติมของแต่ละวัตถุ เมื่อทำการกำหนดโครงสร้างและคุณสมบัติ จะได้ออนโทโลยีการระบาดของโรคไข้เลือดออก ออนโทโลยีผลผลิตข้าวนาปี ออนโทโลยีการขาย และออนโทโลยีชนิดข้อมูล แสดงดังภาพ 15-18



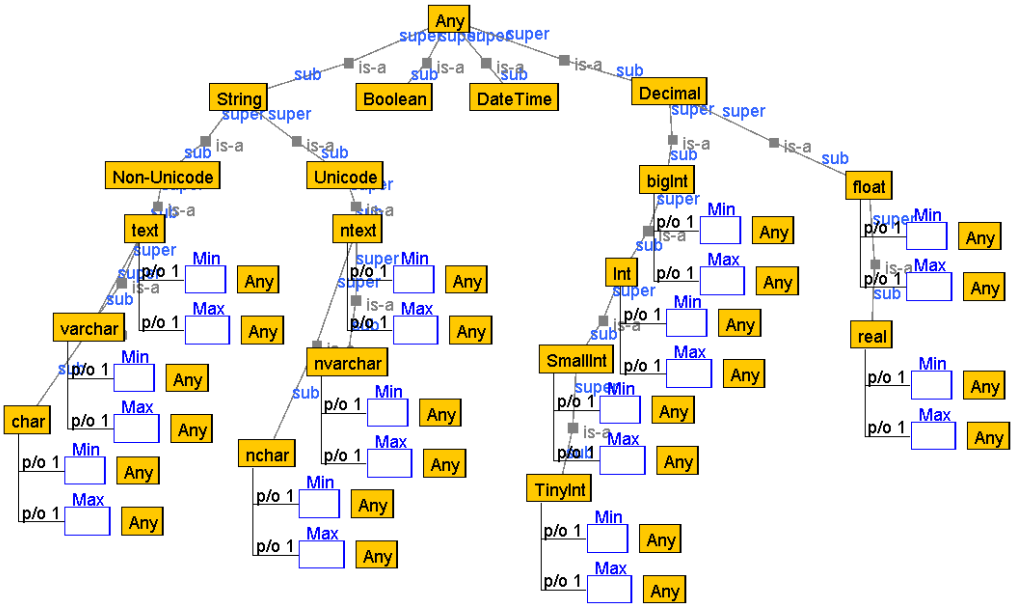

**ภาพ 15 แสดงออนโทโลยีการระบาดของโรคไข้เลือดออก**




**ภาพ 16 แสดงออนโทโลยีผลผลิตข้าว**



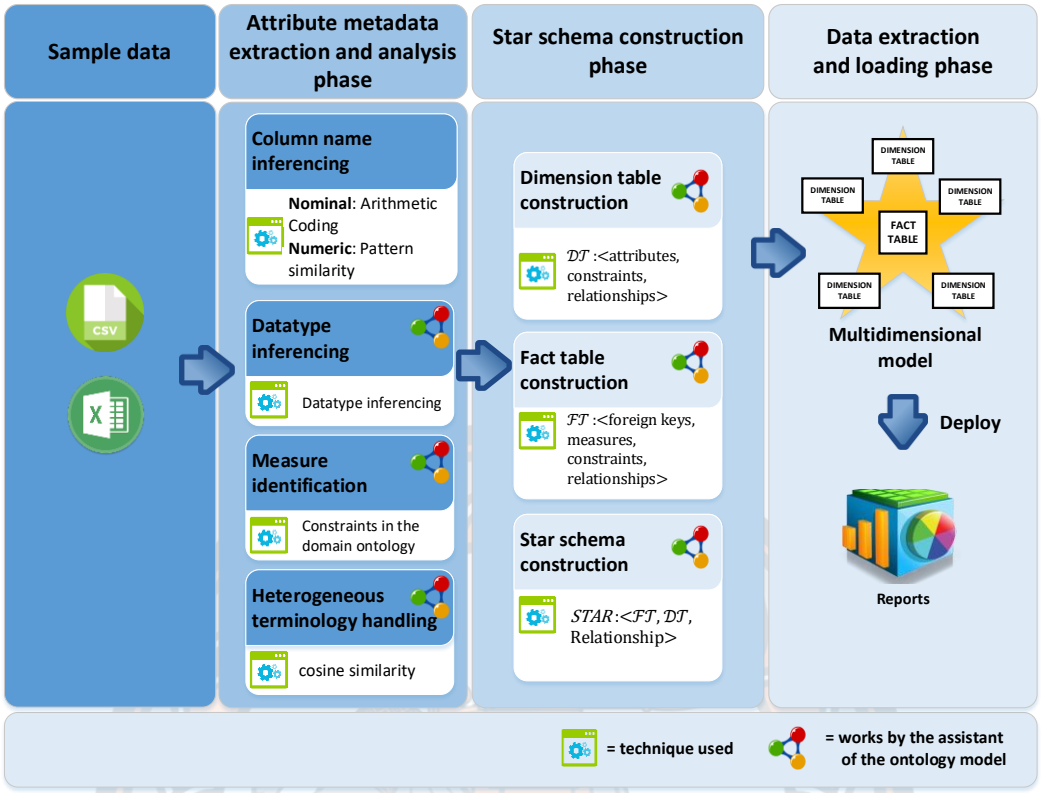

**ภาพ 17 แสดงออนโทโลยีข้อมูลการขาย**




**ภาพ 18 แสดงงออนโทโลยีชนิดข้อมูล**

## กรอบแนวคิดของงานวิจัย

กรอบแนวคิดของการพัฒนาเทคนิคการสร้างโครงสร้างแบบหลายมิติโดยอัตโนมัติ ดังภาพ 19แบ่งการทำงานออกเป็น 3 ส่วนได้แก่ การสกัดและวิเคราะห์ข้อมูล การสร้างโครงสร้างแบบดาว และการสกัดและโหลดข้อมูล โดยมีรายละเอียดดังนี้




**ภาพ 19 แสดงสถาปัตยกรรมการทำงานของระบบ**

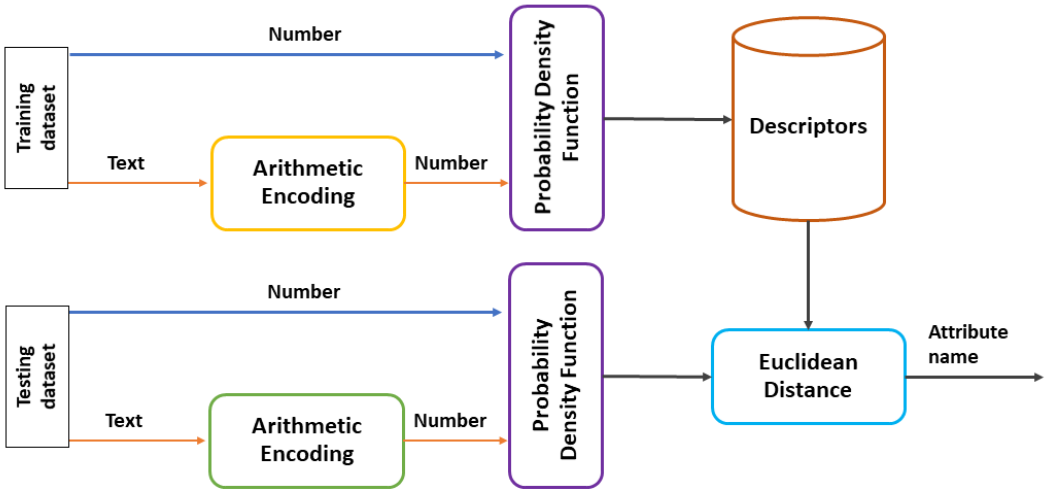
### 1. การสกัดและวิเคราะห์ข้อมูล (Attribute metadata extraction and analysis)

ขั้นตอนการสกัดและวิเคราะห์ข้อมูลนี้มีหน้าที่ในการวิเคราะห์และสกัดข้อมูลจากแหล่งข้อมูล กระบวนการดังกล่าวช่วยในการตรวจสอบความถูกต้องของข้อมูล ช่วยแก้ไขข้อมูลที่ผิดพลาด และสกัดข้อมูลที่ต้องการจากแหล่งข้อมูล แบ่งเป็น 4 กระบวนการ ได้แก่ การอนุมานชื่อคอลัมน์ การอนุมานชนิดข้อมูล การระบุเมเซอร์ และการตรวจสอบคำศัพท์

#### 1.1 อนุมานชื่อคอลัมน์ (Column name determining)

เป็นการอนุมานชื่อคอลัมน์จากข้อมูลในรูปแบบไฟล์ .CSV หรือข้อมูลแบบมีโครงสร้างในรูปแบบไฟล์ตารางคำนวณ หากแหล่งข้อมูลที่นำเข้าสู่ระบบไม่มีความสมบูรณ์ คือ ชื่อคอลัมน์ไม่ปรากฏในเอกสาร หากข้อมูลดังกล่าวเป็นข้อมูลที่มีความสำคัญก็จะส่งผลให้การสร้างคลังข้อมูลไม่สมบูรณ์ การแสดงผลข้อมูลอาจไม่มีความครบถ้วนได้ ฉะนั้นจึงต้องมีการอนุมานชื่อคอลัมน์ โดยนำข้อมูลในคอลัมน์นั้นมาทำการวิเคราะห์หาชื่อคอลัมน์ จากงานวิจัยที่ผ่านมามีการใช้การวิเคราะห์จากรูปแบบของคำ (Regular expressions) หรือเรียกว่า regex ซึ่งเป็นการกำหนดรูปแบบ (Pattern) ของคำเพื่อตรวจสอบรูปแบบของข้อมูลว่าเป็นรูปแบบที่ถูกต้องหรือไม่ วิธีนี้สามารถทำงานได้ดีกับข้อมูลที่มีรูปแบบที่ชัดเจน ยกตัวอย่างเช่น ข้อมูลอีเมล หมายเลขโทรศัพท์ และ

เลขบัตรประชาชน เป็นต้น แต่ชื่อคอลัมน์ในเอกสารไม่ได้มีรูปแบบที่ชัดเจนตายตัวยกตัวอย่าง เช่น ข้อมูลอุณหภูมิและจำนวนผู้ป่วย เป็นต้น ดังนั้นเทคนิค regex อาจทำงานได้ไม่ดีกับข้อมูลรูปแบบนี้ ผู้วิจัยจึงเสนอวิธีการอนุมานชื่อคอลัมน์โดยใช้เทคนิคที่เรียกว่าฟังก์ชันความหนาแน่นของความน่าจะเป็นและการเข้ารหัสเลขคณิตมาใช้อนุมานชื่อคอลัมน์ โดยมีแนวคิดในการอนุมานชื่อคอลัมน์แสดงดังภาพ 20

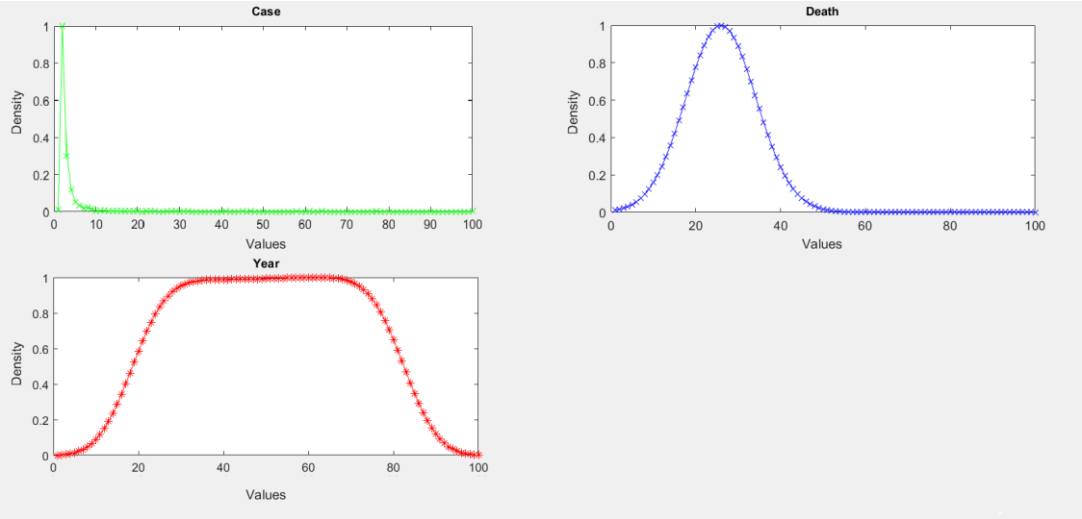



**ภาพ 20 แสดงแนวคิดการอนุมานชื่อคอลัมน์สำหรับข้อมูลตัวอักษรและตัวเลข**

ข้อมูลที่นำมาทดสอบการอนุมานชื่อคอลัมน์เป็นข้อมูลจาก 3 โดเมน คือ ข้อมูลการระบาดของโรคไข้เลือดออกในประเทศไทย ข้อมูลผลผลิตข้าวนาปี และข้อมูลการขายจากฐานข้อมูล AdventureWorks รวม 25 คอลัมน์ โดยได้แบ่งข้อมูลในแต่ละคอลัมน์ออกเป็นชุดการสอน (Training set) จำนวน 70 เปอร์เซ็นต์ของข้อมูลทั้งหมดและชุดทดสอบ (Testing set) จำนวน 30 เปอร์เซ็นต์ แนวคิดดังกล่าวสามารถอนุมานชื่อคอลัมน์ได้จากรูปแบบข้อมูล 2 รูปแบบ คือ ข้อมูลที่อยู่ในรูปแบบตัวเลขและข้อมูลที่อยู่ในรูปแบบตัวอักษร โดยข้อมูลที่อยู่ในรูปแบบตัวเลขสามารถใช้ฟังก์ชันความหนาแน่นของความน่าจะเป็นในการสอนระบบและทดสอบระบบได้เลย แต่หากข้อมูลในคอลัมน์เป็นข้อมูลชนิดตัวอักษรต้องทำการแปลงข้อมูลตัวอักษรให้อยู่ในรูปแบบตัวเลขโดยใช้การเข้ารหัสเลขคณิตก่อนเข้าสู่ฟังก์ชันความหนาแน่นของความน่าจะเป็น เนื่องจากการอนุมานชื่อคอลัมน์โดยใช้ฟังก์ชันความหนาแน่นของความน่าจะเป็นข้อมูลที่นำเข้าต้องเป็นข้อมูลที่อยู่ในรูปแบบตัวเลขเท่านั้น

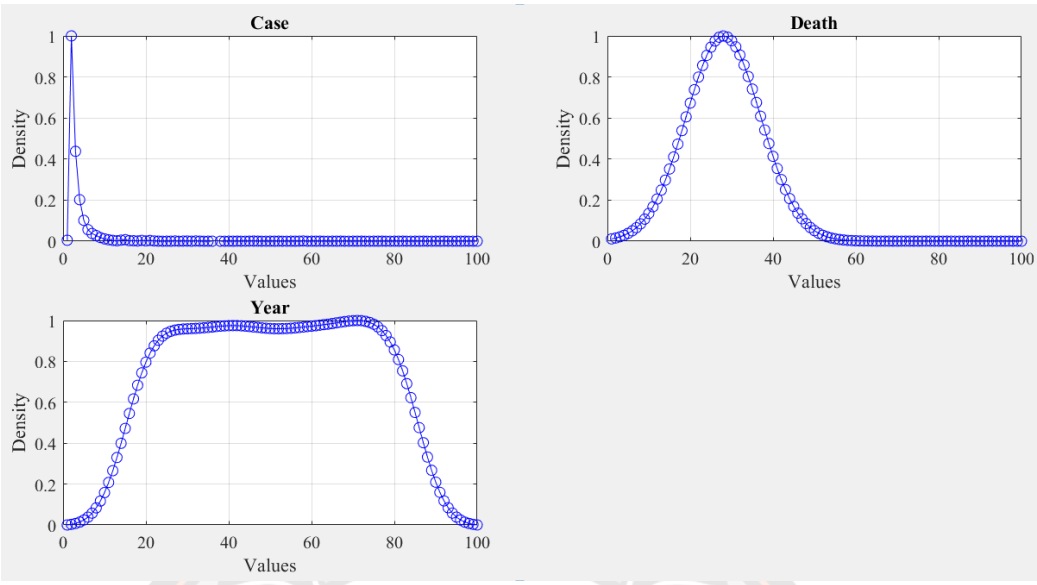
### 1.1.1 ข้อมูลที่อยู่ในรูปแบบตัวเลข

ข้อมูลที่อยู่ในรูปแบบตัวเลขผู้วิจัยใช้ฟังก์ชันความหนาแน่นของความน่าจะเป็นในการอนุมานชื่อคอลัมน์โดยจะได้รูปแบบฟังก์ชันความหนาแน่นของความน่าจะเป็นจากชุดการสอนของแต่ละคอลัมน์ แสดงตัวอย่างดังภาพ 21



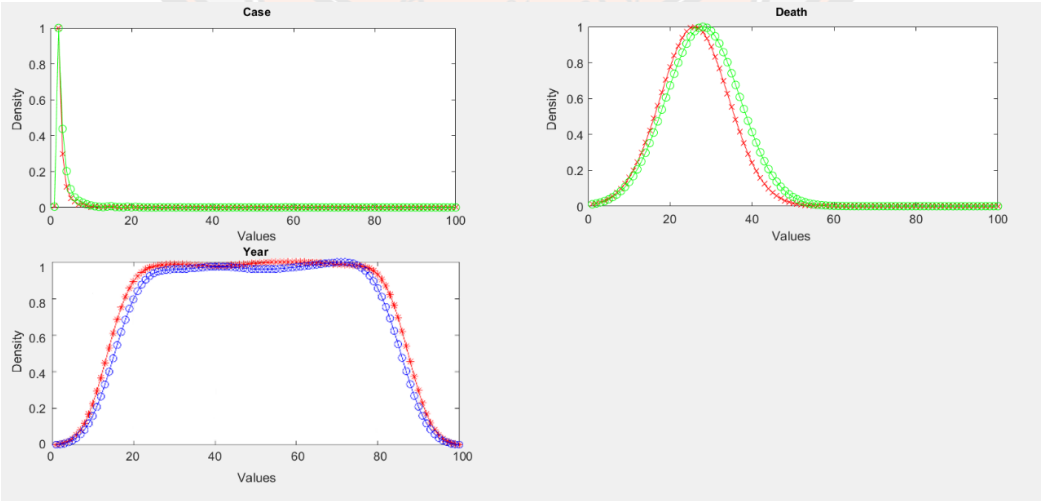

**ภาพ 21 แสดงกราฟฟังก์ชันความหนาแน่นของความน่าจะเป็นจากชุดการสอนของข้อมูลที่อยู่ในรูปแบบตัวเลข**

ขั้นตอนต่อมาจะใช้กระบวนการทำงานเดียวกันนี้กับชุดทดสอบจะได้รูปแบบฟังก์ชันความหนาแน่นของความน่าจะเป็นจากชุดทดสอบของแต่ละคอลัมน์ แสดงตัวอย่างดังภาพ 22




**ภาพ 22 แสดงกราฟฟังก์ชันความหนาแน่นของความน่าจะเป็นจากชุดทดสอบของข้อมูลที่อยู่ในรูปแบบตัวเลข**

จากนั้นนำค่าที่ได้จากจากชุดทดสอบและชุดการสอนมาคำนวณเพื่อหาความคล้ายคลึงกัน โดยใช้ระยะทางแบบยุคลิด หากทั้งสองชุดมีความคล้ายคลึงกันจะได้ค่าระยะทางแบบยุคลิดที่น้อยที่สุด ถือว่าเป็นข้อมูลที่อยู่ในชื่อคอลัมน์เดียวกัน เมื่อนำรูปแบบของฟังก์ชันทั้งสองชุดข้อมูลมาเปรียบเทียบกันพบว่าข้อมูลจากแอตทริบิวต์เดียวกันจะมีรูปแบบความหนาแน่นคล้ายกัน ดังภาพ 23




**ภาพ 23 แสดงกราฟเปรียบเทียบความหนาแน่นของความน่าจะเป็นจากชุดการสอนและชุดทดสอบของข้อมูลที่อยู่ในรูปแบบตัวเลข**

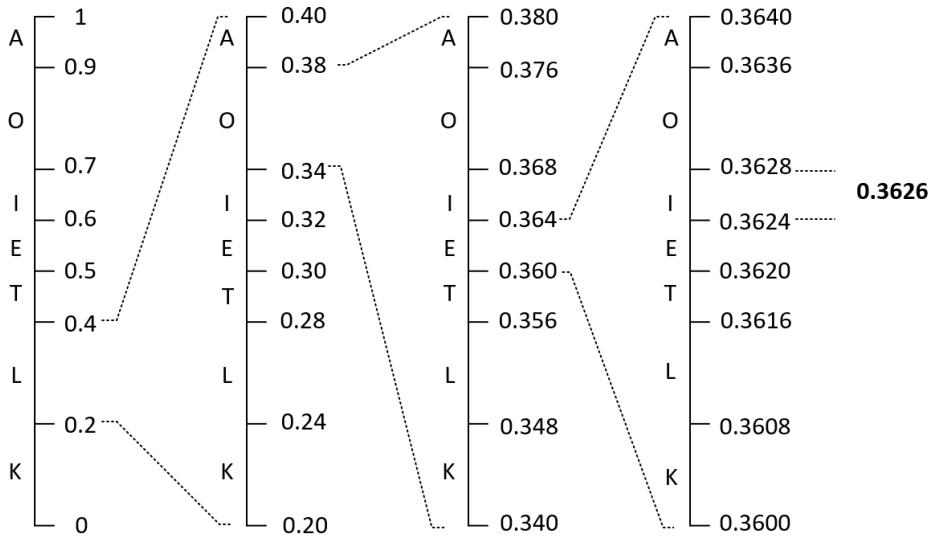
1.1.2 ข้อมูลที่อยู่ในรูปแบบตัวอักษร

ข้อมูลที่เป็นตัวอักษรไม่สามารถนำมาคำนวณด้วยฟังก์ชันความหนาแน่นของความน่าจะเป็นได้ เนื่องจากข้อมูลนำเข้าฟังก์ชันความหนาแน่นของความน่าจะเป็นต้องอยู่ในรูปแบบของตัวเลข ผู้วิจัยจึงได้นำการเข้ารหัสข้อมูลที่เรียกว่าการเข้ารหัสเลขคณิต (Witten, et al., 1987; Howard and Vitter, 1992) มาทำการเข้ารหัสข้อมูลที่อยู่ในรูปแบบตัวอักษร ข้อความจะถูกแทนด้วยเลขจำนวนจริงที่อยู่ในช่วง [0,1) หากข้อความมีจำนวนตัวอักษรมากจำนวนจริงที่ใช้แทนข้อความจะมีค่าน้อย และหากข้อความมีจำนวนตัวอักษรน้อยจำนวนจริงที่ใช้แทนข้อความจะมีค่ามาก ยกตัวอย่างการเข้ารหัสเลขคณิตของข้อมูลที่ประกอบไปด้วยสองจังหวัดคือ ตาก (TAK) และเลย (เลย) ตัวอักษรทั้งหมด คือ (A, O, I, E, T, L,K) นำตัวอักษรทั้งหมดที่ปรากฏในคอลัมน์มาหาความน่าจะเป็นของแต่ละตัวอักษรได้ดังตาราง 10

**ตาราง 10 แสดงตัวอย่างค่าความน่าจะเป็นของแต่ละตัวอักษร**

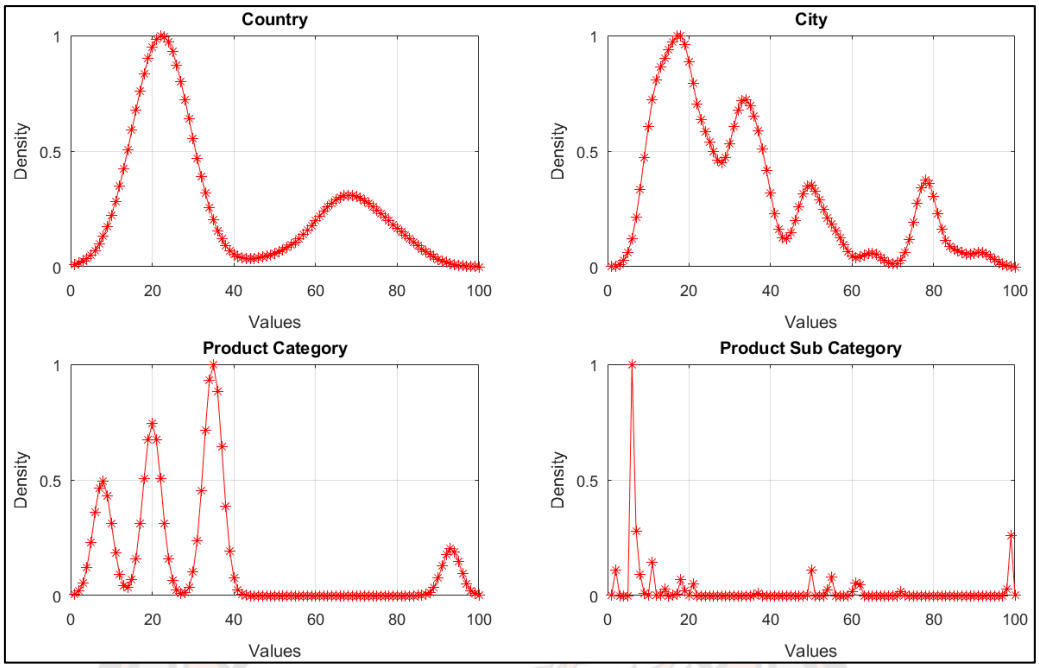
| ตัวอักษร | ความน่าจะเป็น | ช่วง |
|---|---|---|
| A | 0.1 | [0,0.1) |
| O | 0.2 | [0.1,0.3) |
| I | 0.1 | [0.3,0.4) |
| E | 0.1 | [0.4,0.5) |
| T | 0.1 | [0.5,0.6) |
| L | 0.2 | [0.6,0.8) |
| K | 0.2 | [0.8,1) |

จากตาราง 10 เป็นการหาค่าความน่าจะเป็นของแต่ละตัวอักษรที่ต้องการเข้ารหัส จะได้ค่าความน่าจะเป็นของแต่ละตัวอักษร เมื่อได้ค่าความน่าจะเป็นแล้วจะทำการแบ่งช่วงของข้อมูลให้อยู่ในช่วง 0 ถึง 1 เช่น ตัวอักษร “A” อยู่ในช่วง [0,0.1) ตัวอักษร “O” อยู่ในช่วง [0.1,0.3) ตัวอักษร “I” อยู่ในช่วง [0.3,0.4) ตัวอักษร “E” อยู่ในช่วง [0.4,0.5) ตัวอักษร “T” อยู่ในช่วง [0.5,0.6) ตัวอักษร “L” อยู่ในช่วง [0.6,0.8) จนถึงตัวอักษรตัวสุดท้าย คือ ตัวอักษร “K” อยู่ในช่วง [0.8,1) จากค่าความน่าจะเป็นของตัวอักษรจะนำมาใช้เข้ารหัสของคำต่าง ๆ ที่อยู่ในคอลัมน์ ยกตัวอย่างเช่น การเข้ารหัสคำว่า “LOEI” สามารถแสดงขั้นตอนการเข้ารหัสดังภาพ 24



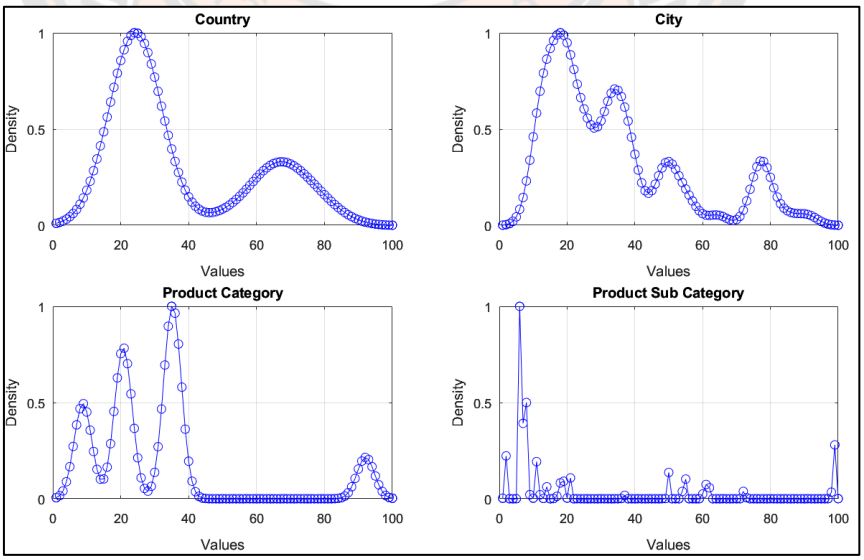

**ภาพ 24 แสดงขั้นตอนการเข้ารหัสคำว่า “LOEI”**

จากภาพเป็นการเลือกช่วงย่อยตามลำดับของสัญลักษณ์ คือ L, O, E, I ตามลำดับ เริ่มจากอักษรตัวแรกคือ “L” การเข้ารหัสจะแคบลงเป็นช่วง [0.2,0.4) ทำการแบ่งช่วงย่อยแล้วเลือกตัวอักษรตัวที่สองคือ “O” จะทำการแบ่งช่วงย่อยให้แคบลงจะอยู่ในช่วง [0.34,0.38) ตัวอักษรตัวที่สามคือ “E” จะทำการแบ่งช่วงย่อยให้แคบลงจะอยู่ในช่วง [0.360,0.364) ตัวอักษรตัวสุดท้ายคือ “I” จะได้ช่วงข้อมูลอยู่ในช่วง [0.3624,0.3628) ขั้นตอนสุดท้ายจะนำค่าที่ได้มาทำการหาจุดกึ่งกลางจะได้ค่าเท่ากับ 0.03626 ซึ่งเป็นตัวเลขที่ใช้แทนข้อความ “LOEI” การเข้ารหัสเลขคณิตช่วยให้สามารถแปลงข้อมูลที่เป็นตัวอักษรให้เป็นตัวเลขได้ เมื่อทำการแปลงข้อมูลทั้งหมดแล้วจะนำข้อมูลที่ได้เข้าสู่ฟังก์ชันความหนาแน่นของความน่าจะเป็นต่อไป ตัวอย่างรูปแบบของฟังก์ชันจากชุดการสอนของข้อมูลที่อยู่ในรูปแบบตัวอักษรดังแสดงดังภาพ 25



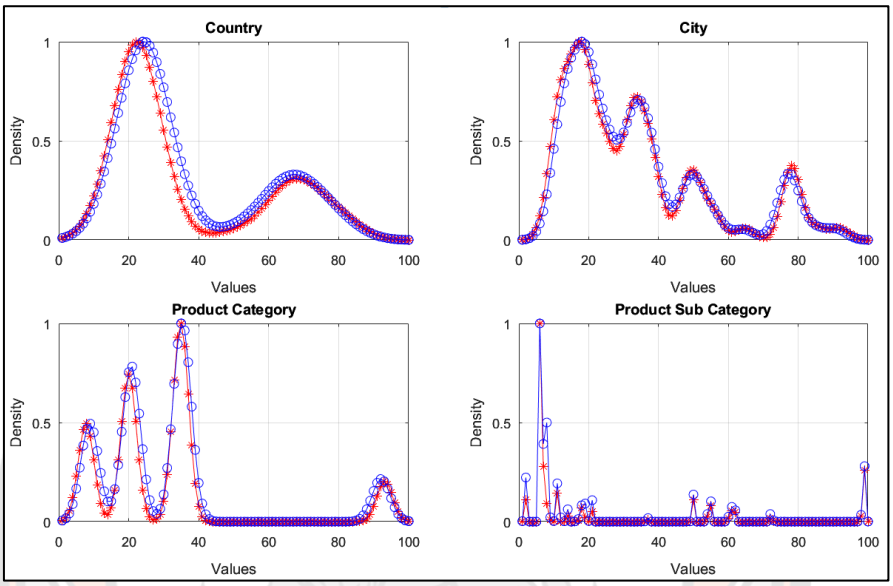

**ภาพ 25 แสดงกราฟฟังก์ชันความหนาแน่นของความน่าจะเป็นจากชุดการสอนของข้อมูลที่อยู่ในรูปแบบตัวอักษร**

ขั้นตอนต่อมาจะใช้กระบวนการทำงานเดียวกันนี้กับชุดทดสอบ จะได้รูปแบบฟังก์ชันความหนาแน่นของความน่าจะเป็นจากชุดทดสอบของแต่ละคอลัมน์ แสดงตัวอย่างดังภาพ 26




**ภาพ 26 แสดงกราฟฟังก์ชันความหนาแน่นของความน่าจะเป็นจากชุดทดสอบของข้อมูลที่อยู่ในรูปแบบตัวอักษร**

จากนั้นนำค่าที่ได้จากจากชุดทดสอบและชุดการสอนมาคำนวณเพื่อหาความคล้ายคลึงกัน โดยใช้ระยะทางแบบยุคลิด หากทั้งสองชุดมีความคล้ายคลึงกันจะได้ค่าระยะทางแบบยุคลิดที่น้อยที่สุด ถือว่าเป็นข้อมูลที่อยู่ในชื่อคอลัมน์เดียวกัน เมื่อนำรูปแบบของฟังก์ชันทั้งสองชุดข้อมูลมาเปรียบเทียบกันพบว่าข้อมูลจากแอตทริบิวต์เดียวกันจะมีรูปแบบความหนาแน่นคล้ายกัน ดังภาพ 27




**ภาพ 27 แสดงกราฟเปรียบเทียบความหนาแน่นของความน่าจะเป็นจากชุดการสอน และชุดทดสอบของข้อมูลที่อยู่ในรูปแบบตัวอักษร**

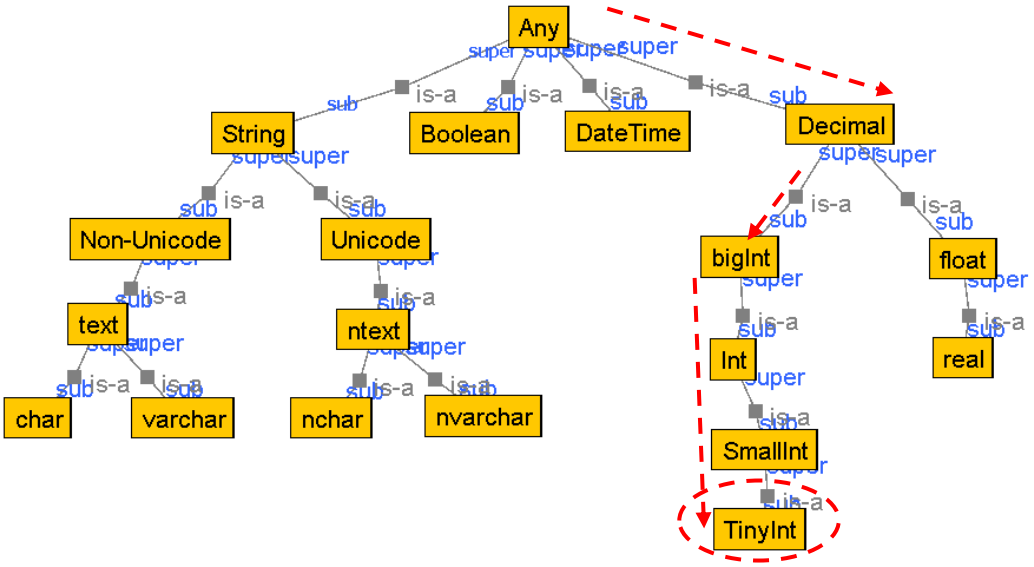
### 1.2 อนุมานชนิดข้อมูล (Datatype determining)

เป็นการอนุมานชนิดของข้อมูลและขนาดของแอตทริบิวต์ ซึ่งเป็นอีกกระบวนการหนึ่งที่มีความสำคัญสำหรับการสร้างโครงสร้างข้อมูลแบบหลายมิติ จากแนวคิดของ Hansen, et al. (2017) และ (Armbrust et al., 2015) มีแนวคิดว่าชนิดข้อมูลที่เหมาะสมที่สุดของคอลัมน์ใดคอลัมน์หนึ่ง คือ ชนิดข้อมูลในโหนดที่ต่ำที่สุดของออนโทโลยีและชนิดข้อมูลนั้นต้องมีขนาดใหญ่กว่าข้อมูลที่มีขนาดใหญ่ที่สุดในคอลัมน์นั้นๆ ผู้วิจัยจึงได้ทำการอนุมานชนิดข้อมูลของแต่ละคอลัมน์ โดยวิเคราะห์จากข้อมูลในคอลัมน์กับโครงสร้างของออนโทโลยีชนิดข้อมูลที่มีรูปแบบเป็นลำดับชั้น ตาราง 11 แสดงตัวอย่างข้อมูลจำนวนผู้ป่วยโรคไข้เลือดออกใน 5 จังหวัดประกอบไปด้วย Amnat Charoen, Ang Thong, Bangkok, Bungkan และ Buri Ram

**ตาราง 11 แสดงจำนวนผู้ป่วยโรคไข้เลือดออก**

| FactID | Province | Case |
|---|---|---|
| 1 | Amnat Charoen | 1 |
| 2 | Ang Thong | 6 |
| 3 | Bangkok | 200 |
| 4 | Bungkan | 0 |
| 5 | Buri Ram | 6 |

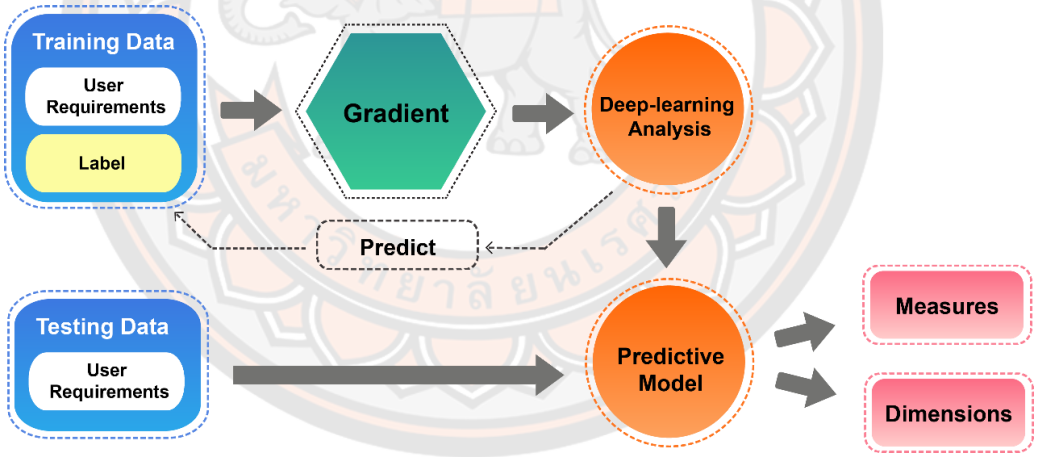
จากข้อมูลนำมาเปรียบเทียบกับออนโทโลยีชนิดข้อมูลโดยมีขั้นตอนดังนี้ เริ่มจากพิจารณาชนิดของข้อมูลจากข้อมูลจำนวนผู้ป่วย พบว่าเป็นข้อมูลชนิดตัวเลขและเป็นตัวเลขที่เป็นจำนวนเต็มจึงพิจารณาออนโทโลยีในโหนด Decimal ลำดับถัดมาพิจารณาตามขนาดของข้อมูลที่มีขนาดใหญ่ที่สุด คือ 200 ระบบจำทำการพิจารณาขนาดตามดำดับ คือ BigInt, int, Smallint และ TinyInt ชนิด TinyInt เป็นชนิดข้อมูลที่เหมาะสมที่สุดเนื่องจากข้อมูลชนิดนี้สามารถเก็บข้อมูลได้สูงสุด 255 ซึ่งข้อมูลที่มากที่สุดในคอลัมน์มีขนาดของข้อมูลไม่เกินขนาดของข้อมูลที่สามารถเก็บได้สูงสุดของข้อมูลชนิดนี้ดังภาพ 29




**ภาพ 28 แสดงการอนุมานชนิดข้อมูล**

### 1.3 ระบุเมเซอร์ (Measure identification)

การระบุเมเซอร์เป็นกระบวนการที่สำคัญอีกกระบวนการหนึ่งในการสร้างโครงสร้างข้อมูลแบบหลายมิติ เนื่องจากการระบุเมเซอร์โดยอัตโนมัติเป็นกระบวนการที่ทำได้ยาก โดยส่วนใหญ่จะใช้การระบุเมเซอร์โดยผู้ใช้หรือการระบุจากแหล่งข้อมูลแบบมีโครงสร้างที่มีโครงสร้างชัดเจน (Phipps & Davis, 2002; Romero & Abelló, 2010) แต่สำหรับข้อมูลที่ไม่มีโครงสร้างหรือแบบกึ่งโครงสร้างในการระบุเมเซอร์จึงเป็นสิ่งที่ทำได้ยาก สำหรับขั้นตอนนี้เป็นการระบุเมเซอร์โดยอัตโนมัติจากข้อมูลแบบกึ่งโครงสร้าง โดยมีขั้นตอนดังนี้ ขั้นแรกในการระบุเมเซอร์ คือ การพิจารณาจากข้อกำหนดของผู้ใช้ซึ่งอยู่ในรูปแบบภาษาธรรมชาติ เช่น “Analyze dengue cases by location and time.” หรือ “Display dengue cases according to location and patient age group.” เป็นต้น ในโดเมนไข้เลือดออกผู้ใช้อาจต้องการทราบจำนวนผู้ป่วยหรือผู้เสียชีวิตโดยแบ่งตามช่วงเวลาหรือสถานที่ที่ที่มีการระบาดของโรค ในกระบวนการในการพิจารณาข้อกำหนดของผู้ใช้ผู้วิจัยได้ใช้ spaCy ซึ่งเป็นซอฟต์แวร์เสรี (Free software) ที่มีความสามารถในการสกัดนิพจน์เฉพาะหรือชื่อเฉพาะในประโยค โดยมีกระบวนการทำงานของ spaCy ดังภาพ 29




**ภาพ 29 แสดงกระบวนการทำงานของ spaCy**

กระบวนการทำงานของ spaCy เริ่มจากการการติดลาเบลของคำหรือชื่อเฉพาะ เช่น เมเซอร์ และมิติ โดยแบบจำลองทางสถิติของ spaCy ได้รับการสอนให้รู้จักชื่อของ เมเซอร์ และมิติ เช่น จำนวนผู้ป่วยและผู้เสียชีวิตเป็นเมเซอร์ หรือสถานที่และเวลาเป็นมิติ เป็นต้น ผู้วิจัยได้ทำการสอน spaCy ให้เรียนรู้ชื่อเฉพาะจากการรวบรวมข้อกำหนดของผู้ใช้โดยแบ่งประโยคออกเป็น 3 กลุ่มดังนี้ 1) โครงสร้างอย่างง่าย หมายถึงข้อกำหนดที่มีโครงสร้างพื้นฐานในประโยคภาษาอังกฤษเช่น “Analyze dengue cases by location, time, and age.” 2) โครงสร้างที่ซับซ้อน หมายถึง

ข้อกำหนดที่มีโครงสร้างที่ซับซ้อนมากขึ้น คือมีคำที่ใช้ขยายความ หรืออธิบายเพิ่มเติมในประโยค เช่น “Analyze the top 5 serious dengue cases by location, time, and age.” และ 3) โครงสร้างที่ซับซ้อนร่วมกับการใช้คำศัพท์ที่แตกต่างกัน หมายถึงประโยคที่มีคำที่ใช้ขยายความและแทนคำบางคำด้วยคำที่มีความหมายเหมือนกันแต่เขียนต่างกันหรือคำพ้องความหมาย เช่น “Analyze the top 5 serious dengue patients by different periods, positions, and generations.” โดยประโยคในแต่ละกลุ่มได้แบ่งออกเป็นชุดการสอน 70% และชุดการทดสอบ 30% ชุดการสอนเป็นชุดข้อมูลที่มีการระบุลาเบลให้กับคำ เพื่อระบุว่าคำใดเป็นเมเชอร์คำใดเป็นมิติ แสดงดังภาพ 30

| | A | B | C |
|---|---|---|---|
| 1 | **requirement** | **measure** | **dimension** |
| 2 | Evaluate dengue deaths using time, age and location | deaths | time, age, location |
| 3 | Judge dengue deaths employing location and time | deaths | location, time |
| 4 | Subdivision of dengue deaths in terms of age and time | deaths | age, time |
| 5 | Description of dengue deaths according to age, time and location | deaths | age, time, location |
| 6 | Arranging dengue cases according to age, time and location | cases | age, time, location |
| 7 | Judge dengue cases by time, location and age | cases | time, location, age |
| 8 | To distribute dengue cases according to time and location | cases | time, location |
| 9 | View dengue deaths utilizing location and age | deaths | location, age |
| 10 | Indication of dengue cases with location and age | cases | location, age |
| 11 | Evaluating dengue cases in terms of age and time | cases | age, time |
| 12 | Investigate dengue cases by age, time and location | cases | age, time, location |
| 13 | Present dengue deaths in terms of location, age and time | deaths | location, age, time |
| 14 | Exploration of dengue deaths according to location and time | deaths | location, time |
| 15 | To manifest dengue deaths employing time and age | deaths | time, age |

**ภาพ 30 แสดงตัวอย่างข้อมูลชุดการสอน**

เมื่อทำการสอนให้กับ spaCy จะได้โมเดลสำหรับการพยากรณ์ เพื่อใช้ระบุเมเชอร์และมิติให้กับประโยคที่เป็นข้อกำหนดของผู้ใช้ ซึ่งระบบสามารถระบุว่าคำใดเป็นเมเชอร์และคำใดเป็นมิติ อย่างไรก็ตามคุณลักษณะที่เป็นตัวเลขบางอย่างเป็นตัวเลขที่ไม่ใช่เมเชอร์ จึงต้องพิจารณาจากแอตทริบิวต์ที่มีคุณสมบัติดังนี้ คือ เป็นข้อมูลชนิดจำนวนจริง เป็นฟิลด์ที่ไม่ใช่เป็นคีย์หลัก (Non-Key) และเป็นสมาชิกของรายชื่อของเมเชอร์ $\{M\}$ นิยามของเมเชอร์อธิบายได้ดังสมการที่ 10

$$\mathrm{M} = \left|\{att_i \mid att_i \in \mathrm{AT},\ att_i.dtp \in \mathbf{R},\ att_i \text{ is non-key, } and\ att_i \in \{M\}\}\right| \quad (10)$$

จากสมการกำหนดให้

$att_i$ คือ แอทริบิวต์ต่าง ๆ ในแอตทริบิวต์ทั้งหมด ($\mathrm{AT}$)

$dtp$ คือ ชนิดของข้อมูลของ $att_i$

$\{M\}$ คือ รายชื่อของเมเชอร์

**1.4 การจัดการคำศัพท์ที่แตกต่างกัน (Heterogeneous terminology handling)**

การจัดการคำศัพท์ที่แตกต่างกันเป็นการตรวจสอบชื่อของคอลัมน์ เนื่องจากแหล่งข้อมูลที่แตกต่างกันอาจใช้ชื่อคอลัมน์ที่แตกต่างกับและอาจแตกต่างจากที่กำหนดไว้ในออนโทโลยี ยกตัวอย่างชื่อคอลัมน์ที่แตกต่างกัน เช่น คอลัมน์ชื่อจังหวัดอาจใช้คำว่า State หรือ Province ซึ่งเป็นคำที่มีความหมายเหมือนกันแต่มีการเขียนที่แตกต่างกันหรือเรียกว่าคำพ้องความหมาย (Synonym) ผู้วิจัยจึงใช้คลังคำศัพท์ WordNet (Miller, 1995) ในการตรวจสอบคำพ้องความหมายเพื่อให้ระบบสามารถจับคู่คำศัพท์ที่มีความหมายตรงกันได้ แสดงดังอัลกอริทึมในภาพ 31

**Algorithm 1: Synonym and word sense disambiguation.**
Input: Column names (q) which do not match with ontology's terms (Anon)
Output: A list of similarity words(Anon)
1. LOOK UP all synonym words (Anon) in WordNet for a column name (q);
2. FOR each keyword pairs (Abdalaziz Ahmedl & Mohamed Ahmed, 2014) where ki ∈ K
3. IF (ki match with q) THEN
4. RETURN ki;
5. ELSE
6. Compute similarity score (si) of (ki,q);
7. END FOR
8. SORT (Anon) according to their si;
9. RETURN ki; //Select ki that has the highest similarity value of word sense.

**ภาพ 31 แสดงอัลกอริทึมตรวจสอบคำพ้องความหมายและการสะกดคำ**

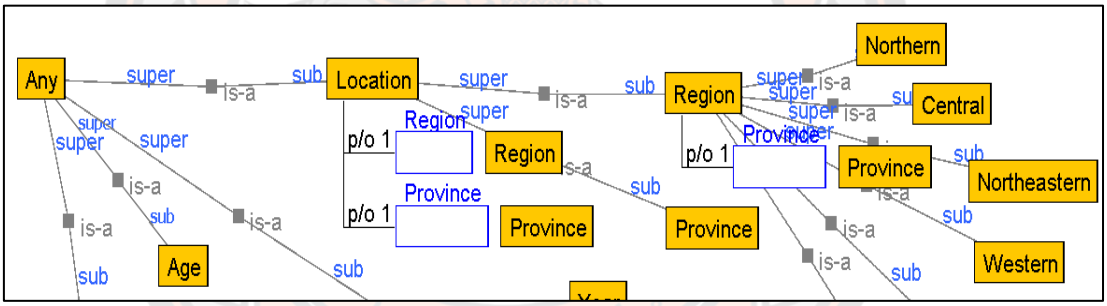
**2. การสร้างโครงสร้างข้อมูลแบบหลายมิติ (Multidimensional schema construction)**

การสร้างโครงสร้างข้อมูลแบบหลายมิติเป็นการนำข้อมูลที่ได้จากกระบวนการสกัดและวิเคราะห์ข้อมูลมาสร้างโครงสร้างข้อมูลแบบหลายมิติ (Multidimensional schema) โดยการแมพข้อมูลกับคลาสในออนโลโลยี แล้วทำการสร้างตารางมิติ (Dimension table) ตารางข้อเท็จจริง

(Fact table) และสร้างโครงสร้างแบบดาว (Star schema) เพื่อนำมาวิเคราะห์ข้อมูลโดยใช้ OLAP ในขั้นตอนนี้แบ่งเป็น 3 กระบวนการได้แก่ การสร้างตารางมิติ การสร้างตารางข้อเท็จจริง และการสร้างโครงสร้างแบบดาว

**2.1 การสร้างตารางมิติ (Dimension table generation)**

การสร้างตารางมิติเป็นการสร้างแอตทิบิวต์ต่าง ๆ ที่เป็นคุณลักษณะของตารางมิติ โดยทั่วไปแอตทิบิวต์ในตารางมิติจะอยู่ในรูปแบบข้อมูลที่เป็นข้อความหรือข้อมูลตัวเลขที่ไม่ใช้ในการนำมาคำนวณ แอตทริบิวต์เหล่านี้ใช้เพื่อการสืบค้นหรือกรองข้อมูล โดยคีย์หลักของตารางจะถูกสร้างโดยอัตโนมัติด้วยชนิดข้อมูลแบบ Integer โดยมีการนำออนโทโลยีมาช่วยในการค้นหาแอตทริบิวต์ที่เกี่ยวข้องตารางมิติ แอตทริบิวต์ที่เกี่ยวข้องกันจะจัดให้อยู่ในตารางมิติเดียวกัน และจัดลำดับตามระดับชั้นของข้อมูลตามความสัมพันธ์ในโครงสร้างออนโทโลยี ยกตัวอย่างเช่น ภูมิภาค (Region) และจังหวัด (Province) เป็นคลาสที่อยู่ภายใต้คลาสสถานที่ (Location) และจังหวัดจัดเป็นคุณสมบัติของภูมิภาค แสดงดังภาพ 32




**ภาพ 32 แสดงออนโทโลยีแสดงความสัมพันธ์ของคลาสสถานที่ ภูมิภาค และจังหวัด**

ตารางสถานที่จึงประกอบไปด้วยแอตทริบิวต์ภูมิภาคและจังหวัด แสดงให้เห็นว่าออนโทโลยีสามารถช่วย Normalization และ De-normalization ตารางมิติที่มีรูปแบบการเก็บข้อมูลตามลำดับชั้น ซึ่งจะเป็นประโยชน์ในการใช้งาน OLAP แสดงดังตาราง 12

**ตาราง 12 แสดงสถานที่**

| LocationID | region | province |
|---|---|---|
| 1 | Central | Ang Thong |
| 2 | Central | Bangkok |
| 3 | Central | Chai Nat |
| 4 | Central | Kamphaeng Phet |
| 5 | Central | Lop Buri |

**ตาราง 12 (ต่อ)**

| LocationID | region | province |
|---|---|---|
| 6 | Central | Nakhon Nayok |
| 7 | Central | Nakhon Pathom |
| 8 | Central | Nakhon Sawan |
| 9 | Central | Nonthaburi |
| 10 | Central | P.Nakhon S.Ayutthaya |
| 11 | Central | Pathum Thani |
| 12 | Central | Phetchabun |

นอกจากนี้ตารางที่สร้างมีเซอโรเกทคีย์ (Surrogate key) เป็นแอตตริบิวต์ที่ใช้เพื่อเชื่อมข้อมูลกับฟอร์เรนคีย์ (Foreign key) ในตารางข้อเท็จจริง และในบางแอตทริบิวต์จะถูกนำมาวิเคราะห์เพื่อสร้างแอตทริบิวต์ใหม่เพื่อให้ง่ายต่อการนำข้อมูลมาวิเคราะห์ เช่น ข้อมูลวันที่ “08/08/2018” สามารถสร้างข้อมูลเป็น ชื่อวัน วันที่ และ ชื่อเดือน เป็นต้น ผลลัพธ์ที่ได้คือตารางมิติที่มีความสัมพันธ์กับตารางข้อเท็จจริง ดังสมการ (11) แสดงถึงตารามิติที่ประกอบด้วยชุดของข้อมูลที่ใช้อธิบายเมเซอร์ที่อยู่ในตารางข้อเท็จจริง

$$DT = \{AT,\ CST,\ RS \mid att_i \in AT,\ i = 1,2,3,...,n\} \qquad (11)$$

จากสมการกำหนดให้

$\mathrm{AT}$ คือ ชุดของคอลัมน์ในตารางหรือแอตทริบิวต์ (Attributes)

$CST$ คือ ข้อจำกัด (Constraint)

$RS$ คือ ความสัมพันธ์ (Relationship)

ตารางมิติ ( $DT$ ) สร้างมาจากชื่อคลาสในออนโทโลยี ประกอบไปด้วยแอตทริบิวต์ ( $\mathrm{AT}$ ) คือ ชุดของคอลัมน์ในตาราง ข้อจำกัด ( $CST$ ) ของตารางมิติใช้อธิบายลักษณะของแอตทริบิวต์ เช่น เป็นคีย์หลัก (Primary keys) และไม่เป็นค่าว่าง (Not null) เป็นต้น ความสัมพันธ์ ( $RS$ ) ของตารางมิติใช้อธิบายว่าตารางข้อเท็จจริงมีความสัมพันธ์กับตารางมิติอย่างไรดังกำหนดในสมการที่ 12

$$RS = \{(att_{\dim_1}, att_{fact}), ..., (att_{\dim_n}, att_{fact}), cardinality \mid cardinality \in \{1:M\}\} \qquad (12)$$

จากสมการกำหนดให้

$att_{\dim_n}$ คือ แอตทริบิวต์ของตารางมิติ

$att_{fact}$ คือ แอตทริบิวต์ของตารางข้อเท็จจริง

$cardinality$ คือ ลักษณะความสัมพันธ์ของข้อมูล

**2.2 การสร้างตารางข้อเท็จจริง (Fact table construction)**

เป็นกระบวนการสร้างตารางข้อเท็จจริงที่มีฟอร์เรนคีย์ (Foreign key) เพื่ออ้างถึงคีย์หลัก (Primary key) ในตารางมิติ โดยตารางข้อเท็จจริงเป็นตารางที่อยู่ตรงกลางของฐานข้อมูลแบบดาว แอตทริบิวต์ในตารางข้อเท็จจริงโดยทั่วไปจะเป็นข้อมูลในเชิงปริมาณที่เรียกว่าเมเชอร์ (Measure) กำหนดไว้ดังสมการที่ (13)

$$FT = \{FK, \mathrm{M}, CST, RS\} \qquad (13)$$

จากสมการกำหนดให้

$FK$ คือ ฟอร์เรนคีย์ (Foreign key)

$\mathrm{M}$ คือ เมเชอร์ (Measure)

$CST$ คือ ข้อจำกัด (Constraint)

$RS$ คือ ความสัมพันธ์ (Relationship)

**2.3 การสร้างโครงสร้างแบบดาว (Star schema construction)**

กระบวนการสร้างโครงสร้างแบบดาวต้องประกอบไปด้วยตารางข้อเท็จจริง ตารางมิติและความสัมพันธ์ของตารางในฐานข้อมูลดังสมการที่ (14)

$$STAR = \{FT, DT, RS\} \qquad (14)$$

จากสมการกำหนดให้

$FT$ คือ ชุดของตารางข้อเท็จจริง

$DT$ คือ ชุดของตารางมิติ

$RS$ คือ ความสัมพันธ์ (Relationship)

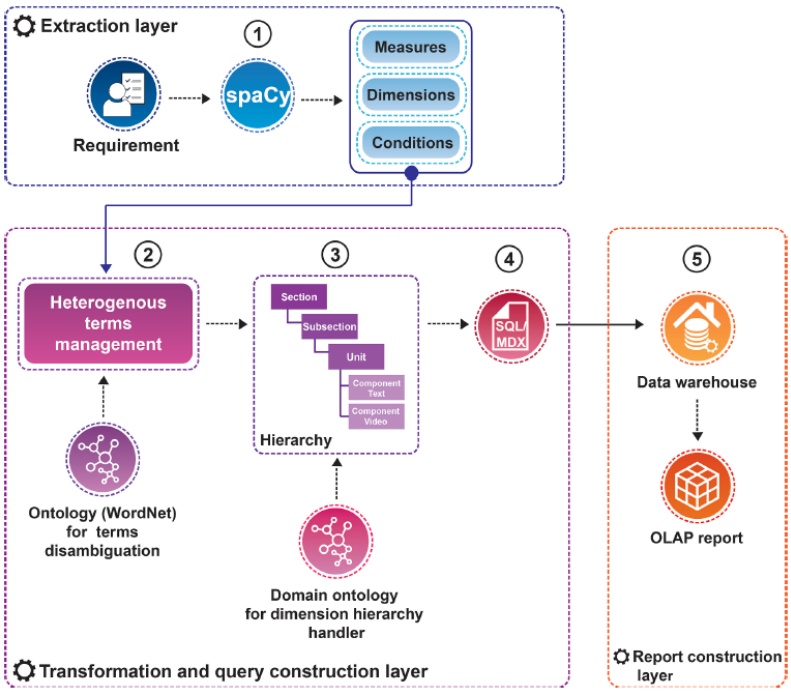
เมื่อได้ตารางข้อเท็จจริงและตารางมิติแล้ว ระบบจะทำการเชื่อมความสัมพันธ์ระหว่างตารางข้อเท็จจริงและตารางมิติ จะได้โครงสร้างแบบดาว

**3. การสกัดและโหลดข้อมูล (Data extraction and loading phase)**

ขั้นตอนการสกัดและโหลดข้อมูลเป็นขั้นตอนการดึงข้อมูลทั้งหมดจากแหล่งข้อมูลแบบกึ่งโครงสร้างเพื่อเติมข้อมูลลงไปในตารางข้อเท็จจริงและตารางมิติ ซึ่งจะทำการถ่ายโอนข้อมูลจากแหล่งข้อมูลไปยังตารางที่ถูกต้องในคลังข้อมูล ข้อมูลที่ซ้ำกันทั้งหมดถูกลบออกก่อนที่จะโหลดลงในตารางมิติและตารางข้อเท็จจริง

**4. การสร้างรายงานในรูปแบบ OLAP (OLAP report generation)**

ทุกองค์กรมีความต้องการระบบสารสนเทศเพื่อช่วยในการตัดสินใจทางธุรกิจ รายงานที่อยู่ในรูปแบบ OLAP สามารถแสดงข้อมูลเชิงลึกจากชุดข้อมูลขนาดใหญ่โดยสรุปข้อมูลและแสดงผลในรูปแบบที่เข้าใจง่าย เพื่อสนับสนุนผู้บริหารได้อย่างมีประสิทธิภาพ ผู้วิจัยจึงได้เสนอวิธีการสร้างรายงาน OLAP โดยอัตโนมัติ แสดงดังภาพ 33




**ภาพ 33 แสดงกระบวนการสร้างรายงานในรูปแบบ OLAP**

จากภาพกระบวนการสร้างรายงานประกอบด้วยสามขั้นตอน คือ การสกัดข้อมูล (Extraction Layer) การแปลงข้อมูลและสร้างแบบสอบถาม (Transformation and query construction layer) และการสร้างรายงาน (Report construction layer) การสกัดข้อมูลเป็นการประมวลผลภาษาธรรมชาติโดยใช้ spaCy จากข้อกำหนดของผู้ใช้ที่อยู่ในรูปแบบประโยคจะได้เมเชอร์ มิติ และเงื่อนไขของรายงาน ขั้นตอนที่สองเป็นการแปลงข้อมูลและสร้างแบบสอบถาม โดยการตรวจสอบคำที่มีความหมายเหมือนกันแต่เขียนในรูปแบบที่แตกต่างกันเพื่อให้สามารถเชื่อมโยงกับคำที่อยู่ในคลังข้อมูลได้ และทำการจัดลำดับชั้นหรือระดับของมิติ เนื่องจากข้อมูลในตารางมิติมีลำดับชั้นของข้อมูล เช่น Region ⟶ Province ⟶ District หรือ Year ⟶ Season ⟶ Month เป็นต้น ซึ่งการใช้โดเมนออนโทโลยีช่วยให้สามารถจัดลำดับชั้นของข้อมูลเหล่านี้ได้อย่างถูกต้อง หลังจากการสร้างลำดับชั้นข้อมูลเสร็จสมบูรณ์แล้ว ข้อมูลลำดับชั้นนี้จะถูกป้อนเข้าสู่กระบวนการสร้างภาษาสอบถามเชิงโครงสร้าง ซึ่งจะแปลคำค้นหาของผู้ใช้ที่อยู่ในรูปแบบภาษาธรรมชาติให้เป็นภาษาสอบถามเชิงโครงสร้าง เพื่อใช้สำหรับการสอบถามข้อมูลจากโครงสร้างแบบดาวในคลังข้อมูล ขั้นตอนสุดท้ายการสร้างรายงานเป็นการสร้างรายงานในรูปแบบ OLAP ตามภาษาสอบถามเชิงโครงสร้างโดยสามารถแสดงผลบนโปรแกรม Microsoft SQL Server 2019

**การประเมินประสิทธิภาพ**

การพัฒนาวิธีเชิงความหมายสำหรับสร้างโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติใช้การประเมินประสิทธิภาพจากค่าความถูกต้อง (Accuracy) โดยมีวิธีการคำนวณดังสมการ (15)

$$accuracy(\%) = \frac{no.\ of\ \ data\ correctly\ predicted}{no.\ of\ \ all\ data} \times 100 \quad (15)$$

จากสมการกำหนดให้

$no.\ of\ \ data\ correctly\ predicted$ คือ จำนวนข้อมูลที่อนุมานได้ถูกต้อง

$no.\ of\ \ all\ data$ คือ จำนวนข้อมูลทั้งหมด

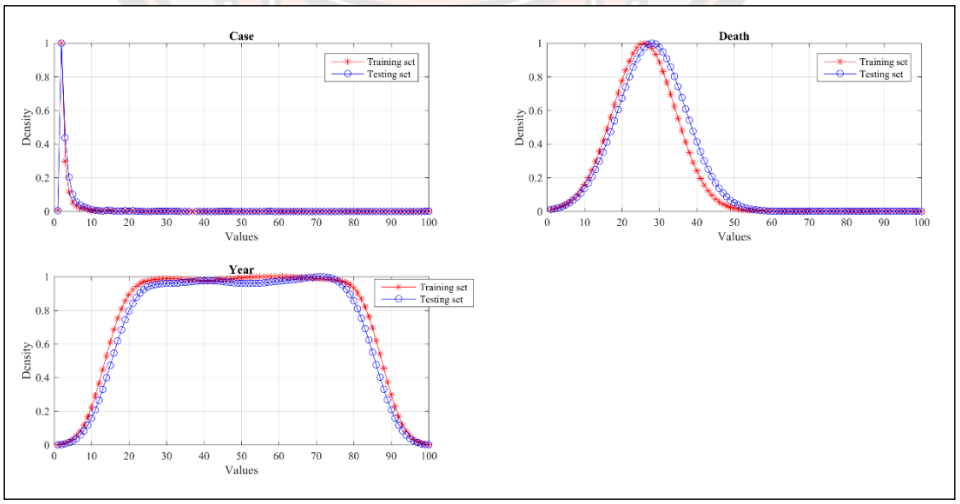
# บทที่ 4

# ผลการวิจัย

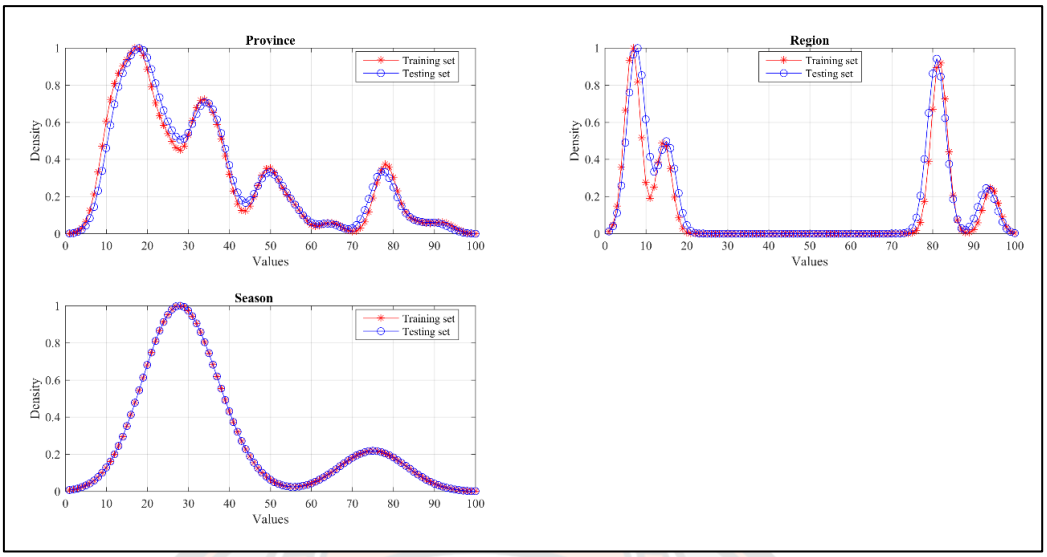
ผลการดำเนินการวิจัยการพัฒนาเทคนิคการสร้างโครงสร้างแบบหลายมิติโดยอัตโนมัติ ประกอบไปด้วย ผลการทดลองเทคนิคการอนุมานชื่อคอลัมน์ ผลการประเมินประสิทธิภาพการอนุมานชนิดข้อมูล ผลการประเมินระยะเวลาในการสร้างโครงสร้างแบบดาว ผลการสร้างโครงสร้างแบบดาว และผลการสร้างรายงานในรูปแบบ OLAP

**ผลการทดลองเทคนิคการอนุมานชื่อคอลัมน์**

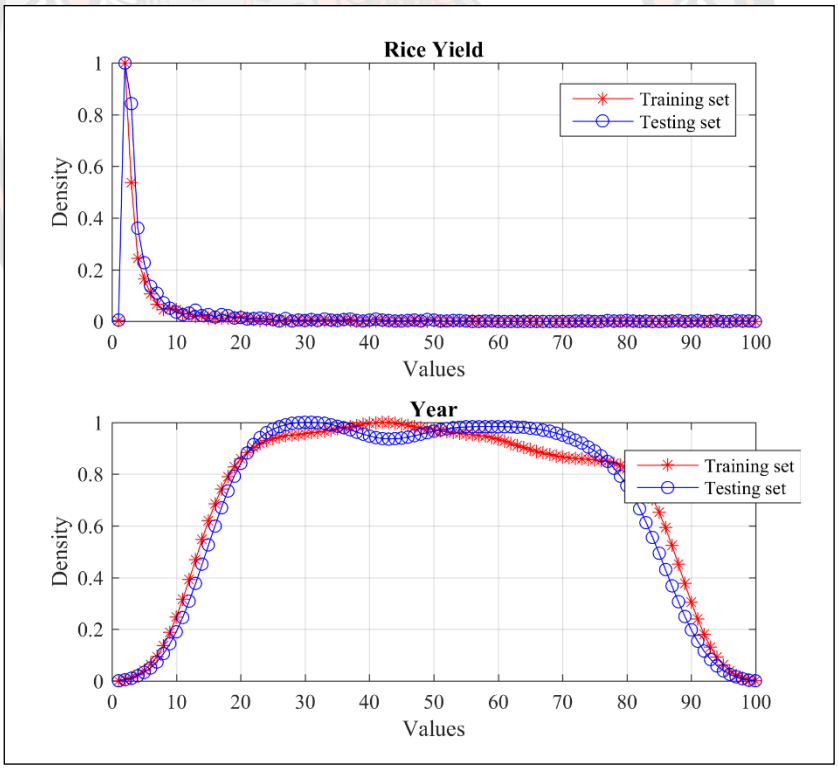
การทดลองนี้เป็นการประเมินประสิทธิภาพในการอนุมานชื่อคอลัมน์โดยใช้เทคนิคฟังก์ชันความหนาแน่นของความน่าจะเป็น (Probability density function) ร่วมกับการเข้ารหัสเลขคณิต (Arithmetic coding) ข้อมูลที่นำมาทำการทดลองประกอบด้วย 3 โดเมน คือ ข้อมูลการระบาดของโรคไข้เลือดออกในประเทศไทย ในปี พ.ศ. 2546 ถึงปี พ.ศ. 2560 จำนวน 13,764 ระเบียน ข้อมูลผลผลิตข้าวนาปี ในปี พ.ศ. 2552 ถึงปี พ.ศ. 2560 จำนวน 4,544 ระเบียน และข้อมูลการขายจากฐานข้อมูล AdventureWorks จำนวน 60,855 ระเบียน ซึ่งในแต่ละโดเมนได้แบ่งข้อมูลออกเป็น 2 ชุด คือ ชุดการสอนจำนวน 70% ของข้อมูล และชุดทดสอบจำนวน 30% ของข้อมูล การวัดประสิทธิภาพใช้วิธี 5-fold cross-validation ตัวอย่างผลการอนุมานชื่อคอลัมน์จากฟังก์ชันความหนาแน่นของความน่าจะเป็นในทั้ง 3 โดเมนของข้อมูลประเภทตัวเลขและตัวอักษรแสดงดังภาพ 34 – 39



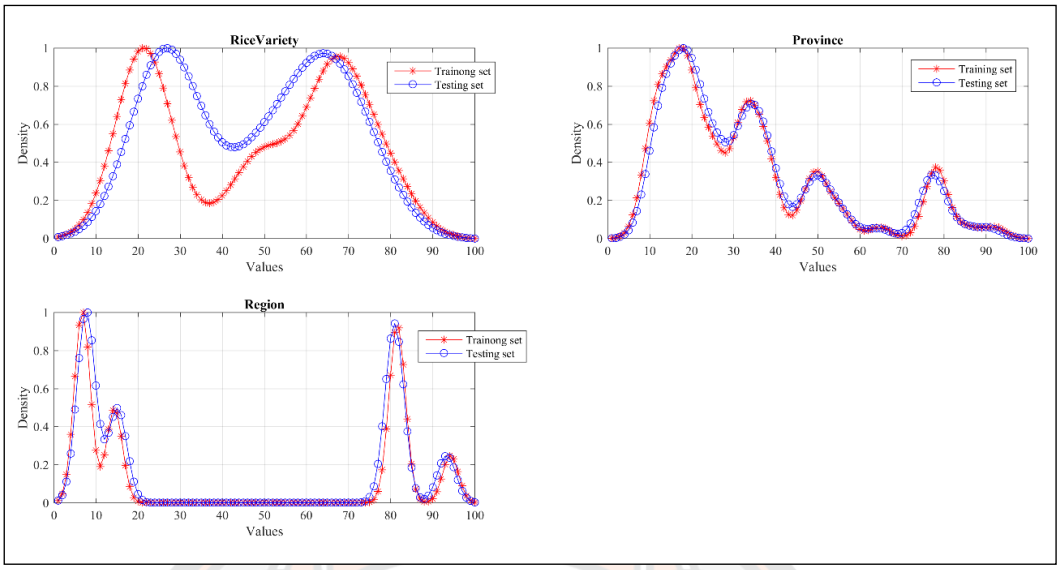

**ภาพ 34 แสดงกราฟฟังก์ชันความหนาแน่นของความน่าจะเป็นของข้อมูลประเภทตัวเลขในโดเมนทางการแพทย์ของแอททริบิวต์ Case, Death และ Year**



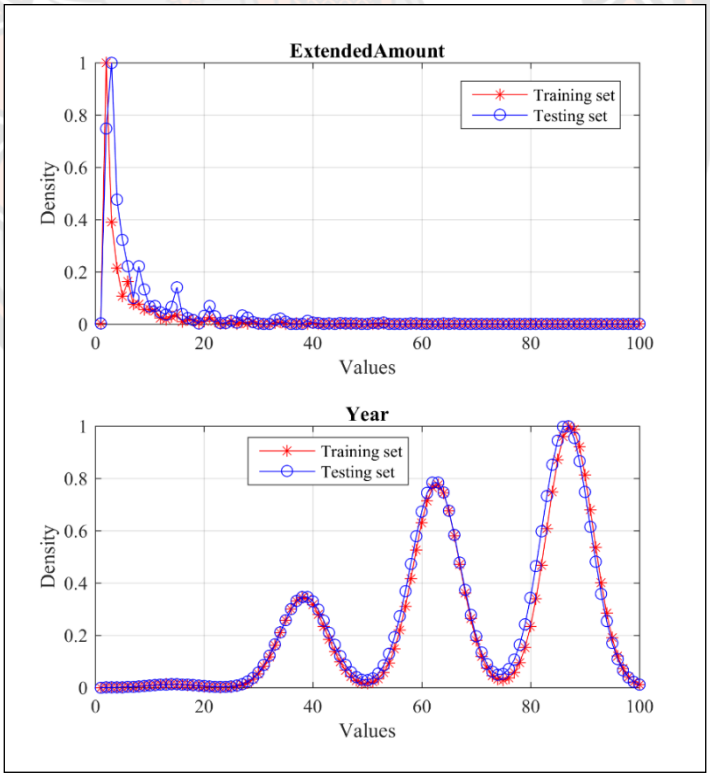

ภาพ 35 แสดงกราฟฟังก์ชันความหนาแน่นของความน่าจะเป็นของข้อมูลประเภทตัวอักษรในโดเมนทางการแพทย์ของแอททริบิวต์ Province, Region และ Season



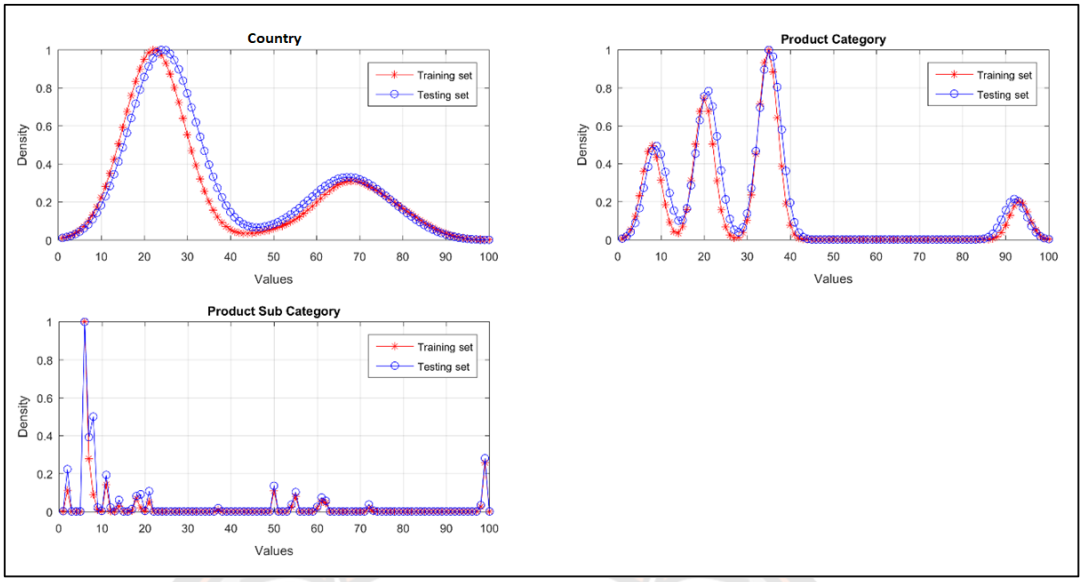

ภาพ 36 แสดงกราฟฟังก์ชันความหนาแน่นของความน่าจะเป็นของข้อมูลประเภทตัวเลขในโดเมนทางการเกษตรของแอททริบิวต์ Rice Yield และ Year




**ภาพ 37 แสดงกราฟฟังก์ชันความหนาแน่นของความน่าจะเป็นของข้อมูลประเภทตัวอักษรในโดเมนทางการเกษตรของแอททริบิวต์ RiceVariety, Province และ Region**




**ภาพ 38 แสดงกราฟฟังก์ชันความหนาแน่นของความน่าจะเป็นของข้อมูลประเภทตัวเลขในโดเมนทางธุรกิจของแอททริบิวต์ ExtendedAmount และ Year**




**ภาพ 39 แสดงกราฟฟังก์ชันความหนาแน่นของความน่าจะเป็นของข้อมูลประเภทตัวอักษรในโดเมนทางธุรกิจของแอททริบิวต์ Country, Product Category และ Product Sub Category**

จากภาพแสดงกราฟฟังก์ชันความหนาแน่นของความน่าจะเป็น โดยแกน X แสดงจำนวนข้อมูล ส่วนแกน Y แสดงค่าความหนาแน่นของความน่าจะเป็น จะเห็นได้ว่าข้อมูลในกลุ่มเดียวกันจะมีรูปแบบความหนาแน่นที่คล้ายคลึงกัน

ค่าที่ได้จากจากชุดทดสอบและชุดการสอนจะถูกนำมาคำนวณเพื่อหาความคล้ายคลึงกันโดยใช้ระยะทางแบบยุคลิด ดังแสดงในสมการ (8) หากค่าระยะทางแบบยุคลิดมีค่าน้อยที่สุดแสดงว่าชุดข้อมูลนั้นมีความคล้ายคลึงกันมากที่สุด ผลการประเมินประสิทธิภาพได้นำมาเปรียบเทียบกับเทคนิคอื่น ๆ อีก 3 เทคนิค ได้แก่ การจำแนกแบบนาอีฟเบย์ (Naïve Bayes Classifier) วิธีต้นไม้ตัดสินใจ (Decision Tree) และโครงข่ายประสาทเทียม (Artificial Neural Networks) จึงได้ผลการประเมินประสิทธิภาพของการอนุมานชื่อคอลัมน์ของทั้ง 3 โดเมนดังต่อไปนี้

**1. ผลการประเมินประสิทธิภาพการอนุมานชื่อคอลัมน์ของโดเมนทางการแพทย์**

การอนุมานชื่อคอลัมน์โดยใช้เทคนิคฟังก์ชันความหนาแน่นของความน่าจะเป็นร่วมกับการเข้ารหัสเลขคณิตของโดเมนทางการแพทย์ ผลการประเมินประสิทธิภาพแสดงดังตาราง 13

**ตาราง 13 แสดงผลการประเมินประสิทธิภาพของการอนุมานชื่อคอลัมน์โดเมนทางการแพทย์**

| Approaches | Accuracy (%) | | |
|---|---|---|---|
| | Nominal | Numerical | Average |
| Naïve Bayes (NB) | 95.75 | 35.92 | 65.84 |
| Decision Tree (DT) | 95.63 | 77.67 | 86.65 |
| Artificial Neural Network (ANN) | 95.12 | 27.18 | 61.15 |
| Our presented approach (PD) | **96.31** | **96.67** | **96.49** |

จากตารางแสดงการเปรียบเทียบการอนุมานชื่อคอลัมน์โดยใช้ฟังก์ชันความหนาแน่นของความน่าจะเป็นกับเทคนิคอื่น ๆ อีก 3 เทคนิค ได้แก่ การจำแนกแบบนาอีฟเบย์ วิธีต้นไม้ตัดสินใจ และโครงข่ายประสาทเทียม จะเห็นได้ว่าฟังก์ชันความหนาแน่นของความน่าจะเป็นสามารถช่วยเพิ่มประสิทธิภาพในการอนุมานชื่อคอลัมน์ได้ โดยมีค่าความถูกต้องเฉลี่ยเท่ากับ 96.49 % ซึ่งมีค่าความถูกต้องสูงที่สุดเมื่อเปรียบเทียบกับการจำแนกแบบนาอีฟเบย์ วิธีต้นไม้ตัดสินใจ และโครงข่ายประสาทเทียม ที่มีค่าความถูกต้องเฉลี่ยเท่ากับ 65.84% 86.65% และ 61.15% ตามลำดับ

เมื่อพิจารณาตามประเภทข้อมูลพบว่าการอนุมานชื่อคอลัมน์จากข้อมูลประเภทตัวอักษร ระบบสามารถอนุมานชื่อคอลัมน์โดยมีค่าความถูกต้องเท่ากับ 96.31 % ซึ่งมีค่าความถูกต้องสูงที่สุดเมื่อเปรียบเทียบกับการจำแนกแบบนาอีฟเบย์ วิธีต้นไม้ตัดสินใจ และโครงข่ายประสาทเทียม ที่มีค่าความถูกต้องเท่ากับ 95.75% 95.63% และ 95.12% ตามลำดับ

การอนุมานชื่อคอลัมน์จากข้อมูลประเภทตัวเลขระบบสามารถอนุมานชื่อคอลัมน์โดยมีค่าความถูกต้องเท่ากับ 96.67 % ซึ่งมีค่าความถูกต้องสูงที่สุด เมื่อเปรียบเทียบกับการจำแนกแบบนาอีฟเบย์ วิธีต้นไม้ตัดสินใจ และโครงข่ายประสาทเทียม ที่มีค่าความถูกต้องเท่ากับ 35.92% 77.67% และ 27.18% ตามลำดับ ซึ่งพบว่าเทคนิคอื่นได้ค่าความถูกต้องน้อยเนื่องจากเทคนิคเหล่านี้ทำงานได้ไม่ดีกับข้อมูลแบบต่อเนื่อง (Continuous Data) เมื่อนำค่าที่ได้มาหาค่าเฉลี่ยพบว่าการอนุมานชื่อคอลัมน์โดยใช้ฟังก์ชันความหนาแน่นของความน่าจะเป็นมีค่าความถูกต้องสูงที่สุดเมื่อเทียบกับเทคนิคอื่นที่นำมาเปรียบเทียบ

**2. ผลการประเมินประสิทธิภาพการอนุมานชื่อคอลัมน์ของโดเมนทางการเกษตร**

การอนุมานชื่อคอลัมน์โดยใช้เทคนิคฟังก์ชันความหนาแน่นของความน่าจะเป็นร่วมกับการเข้ารหัสเลขคณิตของโดเมนทางการเกษตร ผลการประเมินประสิทธิภาพดังตาราง 14

**ตาราง 14 ผลการประเมินประสิทธิภาพของการอนุมานชื่อคอลัมน์โดเมนทางการเกษตร**

| Approaches | Accuracy (%) | | |
|---|---|---|---|
| | Nominal | Numerical | Average |
| Naïve Bayes (NB) | 96.67 | 38.62 | 67.65 |
| Decision Tree (DT) | 96.18 | 80.84 | 88.51 |
| Artificial Neural Network (ANN) | 95.92 | 31.65 | 63.79 |
| Our presented approach (PD) | **81.33** | **98.50** | **89.92** |

จากตารางพบว่าการอนุมานชื่อคอลัมน์โดยใช้ฟังก์ชันความหนาแน่นของความน่าจะเป็นมีค่าความถูกต้องเฉลี่ยเท่ากับ 89.92 % ซึ่งมีค่าความถูกต้องสูงที่สุดเมื่อเทียบกับการจำแนกแบบนาอีฟเบย์ วิธีต้นไม้ตัดสินใจ และโครงข่ายประสาทเทียม ที่มีค่าความถูกต้องเฉลี่ยเท่ากับ 67.65% 88.51% และ 63.79% ตามลำดับ

เมื่อพิจารณาตามประเภทข้อมูลพบว่าการอนุมานชื่อคอลัมน์จากข้อมูลประเภทตัวอักษรสามารถอนุมานชื่อคอลัมน์โดยมีค่าความถูกต้องเท่ากับ 81.33 % ซึ่งมีค่าความถูกต้องน้อยที่สุดเมื่อเปรียบเทียบกับการจำแนกแบบนาอีฟเบย์ วิธีต้นไม้ตัดสินใจ และโครงข่ายประสาทเทียม ที่มีค่าความถูกต้องเท่ากับ 96.67% 96.18% และ 95.92% ตามลำดับ เนื่องจากข้อมูลประเภทตัวอักษรในโดเมนทางการเกษตรข้อมูลที่อยู่ต่างคอลัมน์กันมีคำที่เขียนใกล้เคียงกันยกตัวอย่าง เช่น คอลัมน์พันธุ์ข้าวมีพันธ์ข้าวที่มีชื่อใกล้เคียงกับชื่อจังหวัด คือ “ปทุมธานี 1” ซึ่งเขียนใกล้เคียงกับจังหวัด “ปทุมธานี” และข้อมูลที่อยู่ในคอลัมน์เดียวกันบางคำเขียนใกล้เคียงกันยกตัวอย่าง เช่น คอลัมน์พันธุ์ข้าวมีชื่อพันธ์ข้าวที่ใกล้เคียงกัน คือ “RD6” เขียนใกล้เคียงกับ “RD15” ทำให้การแปลงข้อมูลโดยใช้การเข้ารหัสเลขคณิตได้ค่าที่ใกล้เคียงกันส่งผลให้การอนุมานชื่อคอลัมน์โดยใช้ฟังก์ชันความหนาแน่นของความน่าจะเป็นได้ค่าความถูกต้องน้อยที่สุด

การอนุมานชื่อคอลัมน์จากข้อมูลประเภทตัวเลข สามารถอนุมานชื่อคอลัมน์โดยมีค่าความถูกต้องเท่ากับ 98.50 % ซึ่งพบว่ามีค่าความถูกต้องสูงที่สุด เมื่อเปรียบเทียบกับการจำแนกแบบนาอีฟเบย์ วิธีต้นไม้ตัดสินใจ และโครงข่ายประสาทเทียม ที่มีค่าความถูกต้องเท่ากับ 38.62% 80.84% และ 31.65% ตามลำดับ ซึ่งพบว่าเทคนิคอื่นได้ค่าความถูกต้องน้อยเนื่องจากเทคนิคเหล่านี้

ทำงานได้ไม่ดีกับข้อมูลแบบต่อเนื่อง เมื่อนำค่าที่ได้มาหาค่าเฉลี่ยพบว่าการอนุมานชื่อคอลัมน์โดยใช้ฟังก์ชันความหนาแน่นของความน่าจะเป็นมีค่าความถูกต้องสูงที่สุดเมื่อเทียบกับเทคนิคอื่น

**3. ผลการประเมินประสิทธิภาพการอนุมานชื่อคอลัมน์ของโดเมนทางธุรกิจ**

การอนุมานชื่อคอลัมน์โดยใช้เทคนิคฟังก์ชันความหนาแน่นของความน่าจะเป็นร่วมกับการเข้ารหัสเลขคณิตของโดเมนทางธุรกิจ ผลการประเมินประสิทธิภาพดังตาราง 15

**ตาราง 15 แสดงผลการประเมินประสิทธิภาพของการอนุมานชื่อคอลัมน์โดเมนทางธุรกิจ**

| Approaches | Accuracy (%) | | |
|---|---|---|---|
| | Nominal | Numerical | Average |
| Naïve Bayes (NB) | 95.73 | 39.65 | 67.69 |
| Decision Tree (DT) | 95.52 | 82.97 | 89.25 |
| Artificial Neural Network (ANN) | 95.19 | 32.79 | 63.99 |
| Our presented approach (PD) | **86.75** | **96.00** | **91.38** |

จากตารางพบว่าการอนุมานชื่อคอลัมน์โดยใช้ฟังก์ชันความหนาแน่นของความน่าจะเป็นมีค่าความถูกต้องเฉลี่ยเท่ากับ 91.38 % ซึ่งมีค่าความถูกต้องสูงที่สุดเมื่อเทียบกับการจำแนกแบบนาอีฟเบย์ วิธีต้นไม้ตัดสินใจ และโครงข่ายประสาทเทียม ที่มีค่าความถูกต้องเฉลี่ยเท่ากับ 67.69% 89.25% และ 63.99% ตามลำดับ

เมื่อพิจารณาตามประเภทข้อมูลพบว่าการอนุมานชื่อคอลัมน์จากข้อมูลประเภทตัวอักษรสามารถอนุมานชื่อคอลัมน์โดยมีค่าความถูกต้องเท่ากับ 86.75 % ซึ่งมีค่าความถูกต้องน้อยที่สุดเมื่อเปรียบเทียบกับการจำแนกแบบนาอีฟเบย์ วิธีต้นไม้ตัดสินใจ และโครงข่ายประสาทเทียม ที่มีค่าความถูกต้องเท่ากับ 95.73% 95.52% และ 95.19% ตามลำดับ เนื่องจากข้อมูลประเภทตัวอักษรในโดเมนทางธุรกิจ ข้อมูลที่อยู่ในคอลัมน์เดียวกันเขียนใกล้เคียงกันยกตัวอย่าง เช่น หมวดหมู่ย่อยของสินค้า “Mountain Frames” เขียนใกล้เคียงกับ “Mountain Bikes” ทำให้การแปลงข้อมูลโดยใช้การเข้ารหัสเลขคณิตได้ค่าที่ใกล้เคียงกัน การอนุมานชื่อคอลัมน์โดยใช้ฟังก์ชันความหนาแน่นของความน่าจะเป็นจึงได้ค่าความถูกต้องน้อยที่สุด

การอนุมานชื่อคอลัมน์จากข้อมูลประเภทตัวเลข สามารถอนุมานชื่อคอลัมน์โดยมีค่าความถูกต้องเท่ากับ 96.00 % ซึ่งพบว่ามีค่าความถูกต้องสูงที่สุดเมื่อเปรียบเทียบกับการจำแนกแบบนาอีฟเบย์ วิธีต้นไม้ตัดสินใจ และโครงข่ายประสาทเทียม ที่มีค่าความถูกต้องเท่ากับ 39.65% 82.97% และ 32.79% ตามลำดับ ซึ่งพบว่าเทคนิคอื่นได้ค่าความถูกต้องน้อยเนื่องจากเทคนิคเหล่านี้ทำงานได้ไม่ดีกับข้อมูลแบบต่อเนื่อง เมื่อนำค่าที่ได้มาหาค่าเฉลี่ยพบว่าการอนุมานชื่อคอลัมน์โดยใช้ฟังก์ชันความหนาแน่นของความน่าจะเป็นมีค่าความถูกต้องสูงที่สุดเมื่อเทียบกับเทคนิคอื่น

**ผลการประเมินประสิทธิภาพการอนุมานชนิดข้อมูล**

การทดลองนี้เป็นการอนุมานชนิดข้อมูลด้วยออนโทโลยี โดยเปรียบเทียบกับการระบุชนิดข้อมูลจากโปรแกรม Microsoft SQL Server 2019 จากข้อมูลทั้งหมด 23 คอลัมน์ ข้อมูลรูปแบบตัวอักษร 13 คอลัมน์ และข้อมูลรูแบบตัวเลข 10 คอลัมน์ แต่ละกลุ่มแบ่งการประเมินออกเป็น 3 ระดับ คือ ไม่มีการเบี่ยงเบน (No deviation) คือ การอนุมานประเภทข้อมูลและขนาดของข้อมูลตรงกับที่ผู้เชี่ยวชาญกำหนด การเบี่ยงเบนเล็กน้อย (Minor deviation) คือ การอนุมานชนิดข้อมูลหรือขนาดของข้อมูลไม่ตรงกับที่ผู้เชี่ยวชาญกำหนดแต่ไม่มีผลต่อการนำไปประมวลผล และการเบี่ยงเบนมาก (Major deviation) คือ การอนุมานชนิดข้อมูลหรือขนาดของข้อมูลไม่ตรงกับที่ผู้เชี่ยวชาญกำหนดและส่งผลต่อการประมวลผลข้อมูล

การอนุมานชนิดข้อมูลถ้ามีการกำหนดขนาดของข้อมูลที่น้อยเกินไปก็จะส่งผลต่อข้อมูลที่จัดเก็บ ทำให้ข้อมูลถูกตัดทอนออกไป แต่ถ้ากำหนดขนาดของข้อมูลใหญ่เกินไปก็จะส่งผลต่อพื้นที่ในการจัดเก็บข้อมูลที่ต้องใช้เพิ่มมากขึ้น การกำหนดชนิดข้อมูลที่มีขนาดของข้อมูลที่เหมาะสมจึงเป็นสิ่งจำเป็นสำหรับการสร้างตารางและส่งผลต่อขนาดของคลังข้อมูล ผลการประเมินประสิทธิภาพของการอนุมานชนิดข้อมูลดังแสดงในตาราง 16 และ 17

**ตาราง 16 แสดงผลการประเมินประสิทธิภาพการอนุมานชนิดข้อมูลของข้อมูลประเภทตัวอักษร**

| Approaches | Accuracy (%) | | |
|---|---|---|---|
| | No-dev | Min-dev | Maj-dev |
| **Microsoft SQL Server 2019** | | | |
| Datatype | 38.46 | 0.00 | 61.54 |
| Attribute size | 20.00 | 80.00 | 0.00 |

**ตาราง 16 (ต่อ)**

| Approaches | Accuracy (%) | | |
|---|---|---|---|
| | No-dev | Min-dev | Maj-dev |
| **SSD** | | | |
| Datatype | 84.62 | 0.00 | 15.38 |
| Attribute size | 90.91 | 9.09 | 0.00 |

จากตาราง 16 ผลการประเมินประสิทธิภาพการอนุมานชนิดข้อมูลของข้อมูลประเภทตัวอักษรได้ทำการทดสอบกับ 3 โดเมนรวม 13 คอลัมน์ ผลการอนุมานชนิดข้อมูลโดยใช้ออนโทโลยีพบว่า ระบบสามารถอนุมานชนิดข้อมูลได้ตรงกับที่ผู้เชี่ยวชาญกำหนดมีความถูกต้องเท่ากับ 84.62 % ซึ่งมีความถูกต้องมากกว่าการอนุมานด้วยโปรแกรม Microsoft SQL Server 2019 ซึ่งมีค่าความถูกต้องเท่ากับ 38.46 %

ผลการอนุมานขนาดของข้อมูลโดยใช้ออนโทโลยีพบว่า ระบบสามารถอนุมานขนาดของข้อมูลได้ตรงกับที่ผู้เชี่ยวชาญกำหนดมีค่าความถูกต้องเท่ากับ 90.91 % ซึ่งมีความถูกต้องมากกว่าการอนุมานด้วยโปรแกรม Microsoft SQL Server 2019 ซึ่งมีค่าความถูกต้องเท่ากับ 20.00%

จากผลการประเมินประสิทธิภาพของระบบพบว่า การอนุมานชนิดข้อมูลไม่ตรงกับที่ผู้เชี่ยวชาญกำหนดและส่งผลต่อการประมวลผลข้อมูลมี 2 คอลัมน์ คือ ข้อมูลตัวเลขที่ใช้แทนเดือนและตัวเลขรหัสจังหวัด ซึ่งข้อมูลดังกล่าวเป็นตัวเลขที่ไม่สามารถนำไปคำนวณได้ยกตัวอย่างเช่น ข้อมูลตัวเลขที่ใช้ระบุเดือนจะประกอบไปด้วยตัวเลข 1 ถึง 12 และตัวเลขที่ใช้ระบุจังหวัดมีตัวเลขไม่เกิน 99 เป็นต้น โดยผู้เชี่ยวชาญกำหนดให้เป็น Char(2) แต่ระบบได้กำหนดให้เป็นประเภท TinyInt เนื่องจากข้อมูลเป็นชนิดจำนวนเต็ม การอนุมานขนาดข้อมูลพบว่ามีความเบี่ยงเบนเล็กน้อย จากการกำหนดขนาดข้อมูลไม่ตรงกับผู้เชี่ยวชาญกำหนด เนื่องจากระบบกำหนดขนาดข้อมูลตามความยาวของข้อมูลจริงที่มีขนาดใหญ่ที่สุดแต่ผู้เชี่ยวชาญมองว่าข้อมูลบางข้อมูลอาจมีขนาดที่ใหญ่ขึ้นในอนาคต

โปรแกรม Microsoft SQL Server 2019 กำหนดชนิดข้อมูลได้ตรงกับที่ผู้เชี่ยวชาญกำหนดมีค่าความถูกต้องเท่ากับ 38.46 % คือมีความถูกต้อง 5 คอลัมน์จากทั้งหมด 13 คอลัมน์ ข้อผิดพลาดส่วนใหญ่เกิดจากข้อมูลที่เป็นตัวเลขที่ไม่ได้ใช้ในการคำนวณ เช่น ตัวเลขที่ใช้แทนเดือนและตัวเลขรหัสจังหวัด เป็นต้น โปรแกรม Microsoft SQL Server 2019 กำหนดชนิดข้อมูลที่เป็นตัวเลขทั้งหมดเป็นประเภท Float ส่งผลให้มีการเบี่ยงเบนมากอยู่ที่ 61.54 % สำหรับการกำหนดขนาดของข้อมูลมีความถูกต้อง 20.00% มีความเบี่ยงเบนเล็กน้อยอยู่ที่ 80.00 % เนื่องจากการกำหนดขนาดของข้อมูล

โปรแกรม Microsoft SQL Server 2019 กำหนดตามค่าสูงสุดของข้อมูลชนิดนั้น ส่งผลให้ขนาดของคลังข้อมูลมีขนาดที่ใหญ่ขึ้น

**ตาราง 17 แสดงผลการประเมินประสิทธิภาพการอนุมานชนิดข้อมูลของข้อมูลประเภทตัวเลข**

| Approaches | Accuracy (%) | | |
|---|---|---|---|
| | No-dev | Min-dev | Maj-dev |
| **Microsoft SQL Server 2019** | | | |
| Datatype | 0.00 | 40.00 | 60.00 |
| Attribute size | n/a | n/a | n/a |
| **SSD** | | | |
| Datatype | 90.00 | 10.00 | 0.00 |
| Attribute size | n/a | n/a | n/a |

จากตาราง 17 ผลการประเมินประสิทธิภาพการอนุมานชนิดข้อมูลของข้อมูลประเภทตัวเลขได้ทำการทดสอบกับ 3 โดเมนรวม 10 คอลัมน์ ผลการอนุมานชนิดข้อมูลโดยใช้ออนโทโลยีพบว่า ระบบสามารถอนุมานชนิดข้อมูลได้ตรงกับที่ผู้เชี่ยวชาญกำหนดมีค่าความถูกต้องเท่ากับ 90.00 % ซึ่งมีความถูกต้องมากกว่าการอนุมานด้วยโปรแกรม Microsoft SQL Server 2019 และมีความเบี่ยงเบนเล็กน้อยอยู่ที่ 10.00 % เนื่องจากพบว่าแอตทริบิวต์ที่ถูกอนุมานอย่างไม่ถูกต้องคือแอตทริบิวต์ของจำนวนผู้ป่วย เนื่องจากมีการใช้เครื่องหมายจุลภาค เช่น 1,000 ในค่า ดังนั้นระบบจึงถือว่าเป็นชนิดข้อมูล varchar ในขณะที่ประเภทข้อมูลที่ถูกต้องควรเป็น SmallInt

การอนุมานชนิดข้อมูลด้วยโปรแกรม Microsoft SQL Server 2019 ไม่สามารถระบุชนิดข้อมูลได้ตรงกับผู้เชี่ยวชาญกำหนด เนื่องจากโปรแกรม Microsoft SQL Server 2019 ระบุชนิดข้อมูลเป็น Float ทั้งหมดสำหรับข้อมูลที่เป็นตัวเลข ซึ่งข้อมูลที่เป็นตัวเลขบางคอลัมน์ไม่ใช่ข้อมูลที่อยู่ในรูปแบบทศนิยม เช่น จำนวนผู้ป่วย จำนวนสินค้า เป็นต้น

จากการอนุมานชื่อคอลัมน์จากทั้งสองประเภทข้อมูลจะเห็นได้ว่าแนวทางการอนุมานชนิดข้อมูลด้วยออนโทโลยีมีประสิทธิภาพมากกว่าเมื่อเปรียบเทียบกับโปรแกรม Microsoft SQL Server 2019 ซึ่งไม่มีองค์ความรู้ที่จะช่วยในการอนุมาน

**ผลการประเมินประสิทธิภาพการระบุเมเชอร์**

การระบุเมเชอร์ผู้วิจัยได้ใช้เครื่องมือ spaCy ในการประมวลผลภาษาธรรมชาติ จากรูปแบบประโยคที่เป็นข้อกำหนดของผู้ใช้ซึ่งได้แบ่งออกเป็น 3 กลุ่มดังนี้ 1) โครงสร้างอย่างง่าย หมายถึง ข้อกำหนดที่มีโครงสร้างพื้นฐานในประโยคภาษาอังกฤษเช่น “Analyze dengue cases by location, time, and age.” 2) โครงสร้างที่ซับซ้อน หมายถึงข้อกำหนดที่มีโครงสร้างที่ซับซ้อนมากขึ้น คือมีคำที่ใช้ขยายความ หรืออธิบายเพิ่มเติมในประโยค เช่น “Analyze the top 5 serious dengue cases by location, time, and age.” และ 3) โครงสร้างที่ซับซ้อนร่วมกับการใช้คำศัพท์ที่แตกต่างกัน หมายถึงประโยคที่มีคำที่ใช้ขยายความและแทนคำบางคำด้วยคำที่มีความหมายเหมือนกันแต่เขียนต่างกันหรือคำพ้องความหมาย เช่น “Analyze the top 5 serious dengue patients by different periods, positions, and generations.” โดยประโยคในแต่ละกลุ่มได้แบ่งออกเป็นชุดการสอน 70% และชุดการทดสอบ 30% ประสิทธิภาพของ spaCy สำหรับการระบุเมเชอร์และมิติออกจากข้อกำหนดของผู้ใช้ แสดงดังตาราง 18

**ตาราง 18 แสดงประสิทธิภาพของ spaCy สำหรับการระบุเมเชอร์และมิติ**

| Class | Accuracy (%) | | |
|---|---|---|---|
| | Simple structure | Complex structure | Complex + heterogeneous terminology |
| Measures | 100.00 | 100.00 | 100.00 |
| Dimensions | 100.00 | 94.73 | 66.67 |
| Average | 100.00 | 97.50 | 83.34 |

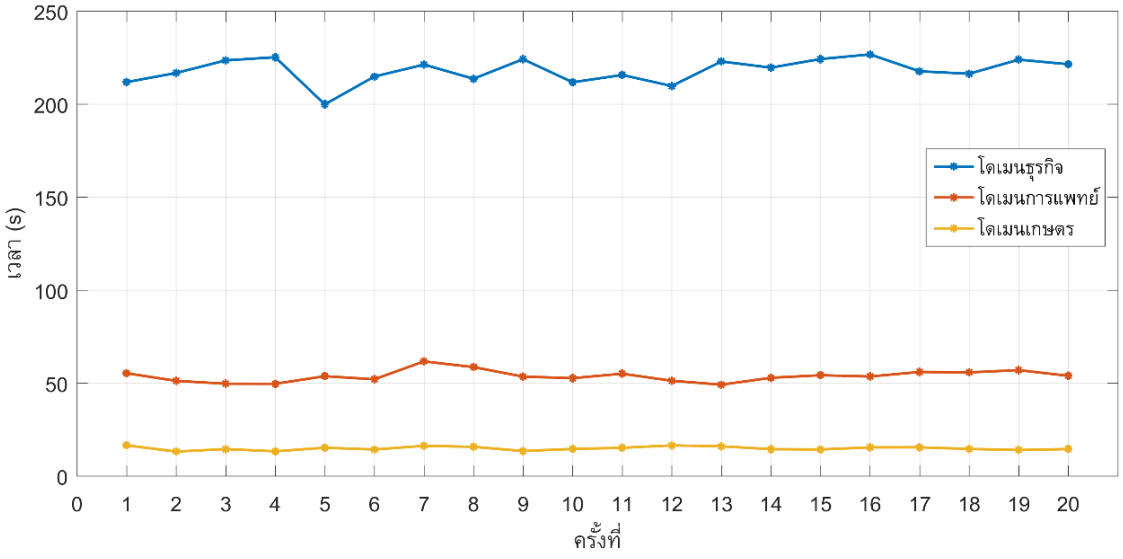
จากตารางพบว่า spaCy สามารถระบุเมเชอร์และมิติจากข้อกำหนดของผู้ใช้ในกรณีที่เป็นโครงสร้างประโยคอย่างง่ายได้ถูกต้อง 100.00 % เนื่องจาก spaCy ใช้กระบวนการเรียนรู้เชิงลึก (Deep learning) ร่วมกับโครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional neural network) เพื่อเรียนรู้รูปแบบของภาษาธรรมชาติจากความต้องการของผู้ใช้ ดังนั้นความถูกต้องของการระบุเมเชอร์และการระบุมิติในชุดข้อมูลทั้งสองจึงมีความถูกต้องสูง

เมื่อทดลองกับประโยคที่มีโครงสร้างที่ซับซ้อนพบว่าระบบสามารถระบุเมเชอร์ได้ถูกต้อง 100.00 % สามารถระบุมิติได้ถูกต้อง 94.73 % เนื่องจากศัพท์เฉพาะสำหรับเมเชอร์มีจำนวนน้อยเมื่อเทียบกับศัพท์เฉพาะที่ใช้กับมิติที่มีจำนวนมากและมีโครงสร้างที่ซับซ้อนมากขึ้น การระบุเมเชอร์และมิติมีค่าความถูกต้องเฉลี่ยอยู่ที่ 97.59 %

เมื่อทดลองกับประโยคที่มีโครงสร้างที่ซับซ้อนร่วมกับการใช้คำศัพท์ที่แตกต่างกันพบว่าระบบสามารถระบุเมเชอร์ได้ถูกต้อง 100.00 % สามารถระบุมิติได้ถูกต้อง 66.67 % ระบบระบุมิติได้ความถูกต้องน้อยเนื่องจากการเปลี่ยนคำศัพท์ในประโยคโดยใช้คำพ้องความหมายทำให้ spaCy ไม่สามารถเรียนรู้ความหมายของคำศัพท์และรูปแบบประโยคจากชุดข้อมูลการฝึกอบรมได้อย่างมีประสิทธิภาพ

**ผลการประเมินระยะเวลาในการสร้างโครงสร้างแบบดาว**

การสร้างโครสร้างแบบดาว ผู้วิจัยได้ใช้ข้อมูลตัวอย่างจากข้อมูลใน 3 โดเมนเป็นการสร้างโครงสร้างแบบดาวโดยอัตโนมัติ ผู้วิจัยได้ทำการทดลองโดยใช้ชุดข้อมูลจากหน่วยงานของรัฐบาลสองหน่วยงาน ได้แก่ ชุดข้อมูลการผลิตข้าวที่มีข้อมูล 4,544 ระเบียนจากสำนักงานเศรษฐกิจการเกษตร และชุดข้อมูลอุบัติการณ์โรคไข้เลือดออกจำนวน 13,764 ระเบียนจากกรมควบคุมโรค ชุดข้อมูลที่สามได้แก่ ชุดข้อมูล AdventureWorksDW ที่มี 60,855 ระเบียนจากไมโครซอฟท์ ในทั้งสามโดเมนนี้ได้ทำการทดลอง 20 ครั้งเพื่อประเมินระยะเวลาในการสร้างโครงสร้างแบบดาวแสดงดังภาพ 40




**ภาพ 40 แสดงกราฟระยะเวลาในการสร้างโครงสร้างแบบดาว**

เมื่อนำมาหาค่าเฉลี่ยของระยะเวลาในการสร้างโครงสร้างแบบดาวโดยแบ่งตามขั้นตอนการทำงานของระบบเป็น 3 ขั้นตอนคือ การสกัดและวิเคราะห์ข้อมูล (data extraction and analysis phase) การสร้างโครงสร้างแบบดาว (star schema construction phase) และการนำเข้าข้อมูล (loading phase) แสดงดังตาราง 19

**ตาราง 19 แสดงระยะเวลาการสร้างโครงสร้างแบบดาว**

| โดเมน | จำนวนคอลัมน์ | จำนวนระเบียน | เวลา (วินาที) | | | |
|---|---|---|---|---|---|---|
| | | | Extract | construction | Load | รวม |
| ทางการเกษตร | 6 | 4,544 | 4.31 | 1.23 | 9.66 | 15.20 |
| ทางการแพทย์ | 23 | 13,764 | 25.59 | 1.18 | 25.29 | 52.06 |
| ทางธุรกิจ | 12 | 60,855 | 20.56 | 1.30 | 189.97 | 211.83 |

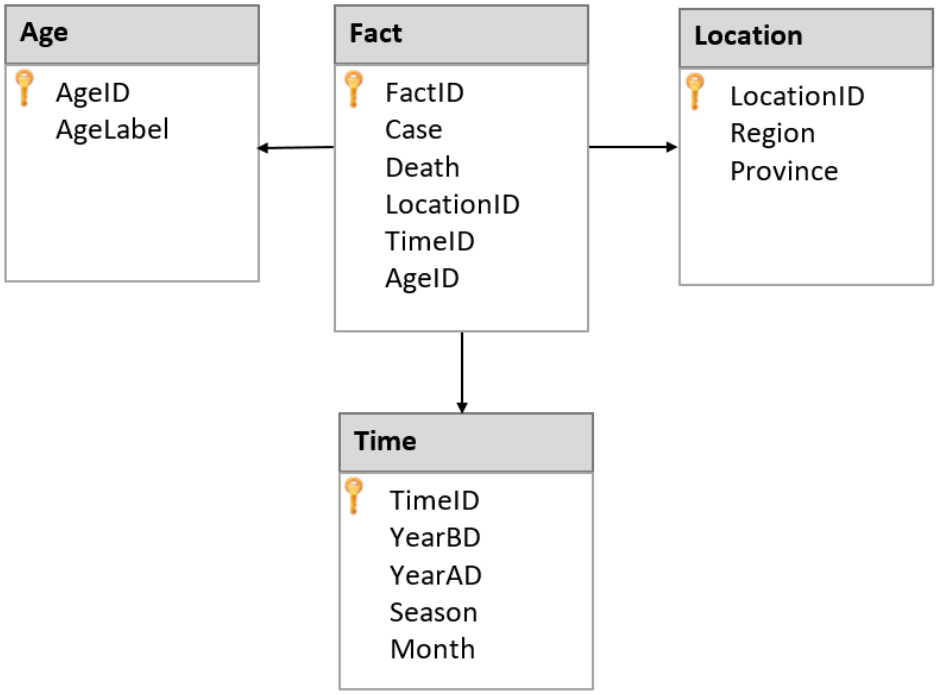
จากตารางแสดงระยะเวลาในการสร้างโครงสร้างแบบดาวพบว่า โดเมนทางการเกษตรระบบใช้เวลาในการสร้างโครงสร้างแบบดาวน้อยที่สุดอยู่ที่ 15.20 วินาที โดเมนทางการแพทย์ใช้เวลา 52.06 วินาที และโดเมนทางธุรกิจใช้เวลา 211.83 วินาที เมื่อแบ่งตามขั้นตอนการทำงานพบว่ากระบวนการสกัดและวิเคราะห์ข้อมูลโดเมนทางการเกษตรใช้เวลาน้อยที่สุดอยู่ที่ 4.31 วินาที โดเมนทางธุรกิจใช้เวลา 20.56 วินาที และโดเมนทางการแพทย์ใช้เวลา 25.59 วินาที จากผลการทดลองสังเกตุได้ว่าระยะเวลาในการสกัดและวิเคราะห์ข้อมูลขึ้นอยู่กับจำนวนคอลัมน์ที่นำมาวิเคราะห์ โดเมนทางการเกษตรมีจำนวนคอลัมน์น้อยที่สุดทำให้ใช้เวลาในการสกัดและวิเคราะห์ข้อมูลน้อยที่สุด ส่วนโดเมนทางการแพทย์มีจำนวนคอลัมน์มากที่สุดทำให้ใช้เวลามากที่สุด

ขั้นตอนการสร้างโครงสร้างแบบดาวพบว่าเวลาที่ใช้สำหรับแต่ละโดเมนไม่แตกต่างกัน เนื่องจากตารางมิติที่สร้างในแต่ละโดเมนมีจำนวนเท่ากัน อย่างไรก็ตามจำนวนคอลัมน์ในตารางมิติส่งผลต่อเวลาในการสร้างโครงสร้างแบบดาวสำหรับแต่ละโดเมน หากมีจำนวนคอลัมน์และความสัมพันธ์ระหว่างตารางมากจะต้องใช้เวลาในการสร้างโครงสร้างแบบดาวมากขึ้น ดังนั้นโดเมนทางการแพทย์มีจำนวนคอลัมน์ที่สร้างน้อยที่สุดจึงใช้เวลาในการสร้างน้อยที่สุดที่ 1.18 วินาที รองลงมาคือโดเมนการเกษตร 1.23 วินาที และโดเมนธุรกิจซึ่งมีจำนวนคอลัมน์ที่สร้างมากที่สุดจึงใช้เวลาในการสร้างมากที่สุดที่ 1.30 วินาที

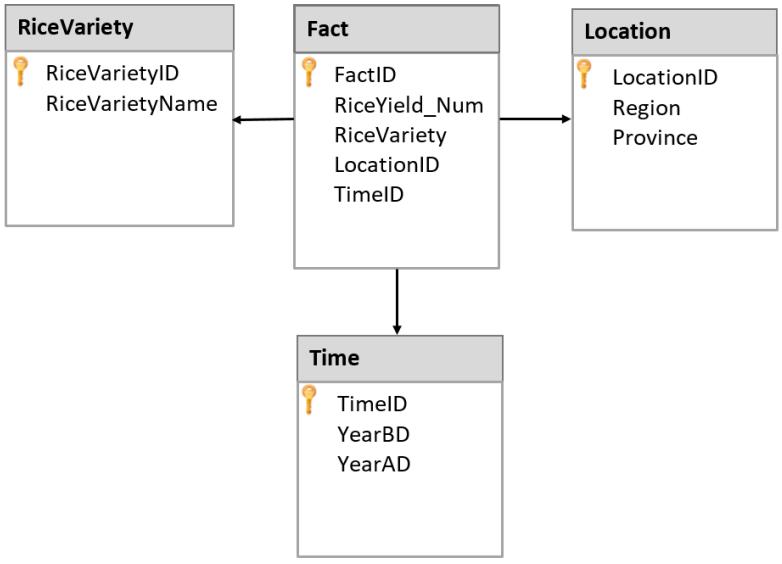
ขั้นตอนการนำเข้าข้อมูลพบว่าโดเมนทางการเกษตรใช้เวลาน้อยที่สุดอยู่ที่ 9.66 วินาที โดเมนทางการแพทย์ใช้เวลา 25.29 วินาที และโดเมนทางธุรกิจใช้เวลา 189.97 วินาที จากผลการทดลองสังเกตได้ว่า ระยะเวลาการนำเข้าข้อมูลของระบบขึ้นอยู่กับปริมาณข้อมูลที่นำเข้า ระบบต้องทำการสกัดข้อมูลจากเอกสารข้อมูลแบบกึ่งโครงสร้าง และตัดข้อมูลที่ซ้ำกันออกก่อนทำการถ่ายโอนข้อมูลเข้าสู่ตารางที่สร้างขึ้นในโปรแกรม Microsoft SQL Server 2019 โดเมนทางการเกษตรมีจำนวนข้อมูลนำเข้าน้อยที่สุดทำให้ใช้เวลานำเข้าข้อมูลน้อยที่สุดจำนวน 4,544 ระเบียน ส่วนโดเมนทางธุรกิจใช้เวลามากที่สุดเนื่องจากมีจำนวนข้อมูลนำเข้ามากที่สุดจำนวน 60,855 ระเบียน ดังนั้นแสดงให้เห็นว่าระยะเวลาการทำงานของระบบขึ้นอยู่กับปริมาณข้อมูลที่นำเข้าสู่ตาราง

**ผลการสร้างโครงสร้างแบบดาว**

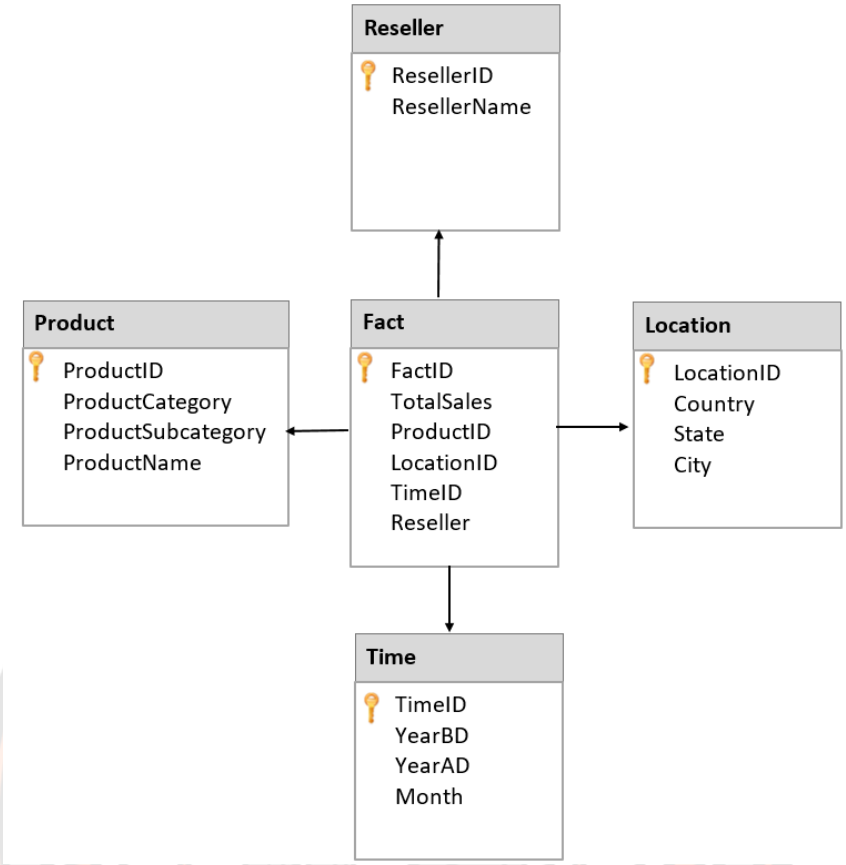
ระบบสามารถสร้างโครงสร้างแบบดาวที่รองรับการรายงานผลในรูปแบบ OLAP ในฐานข้อมูล Microsoft SQL Server 2019 ดังภาพ 41 ถึง 43




**ภาพ 41 แสดงโครงสร้างแบบดาวโดเมนทางการแพทย์**



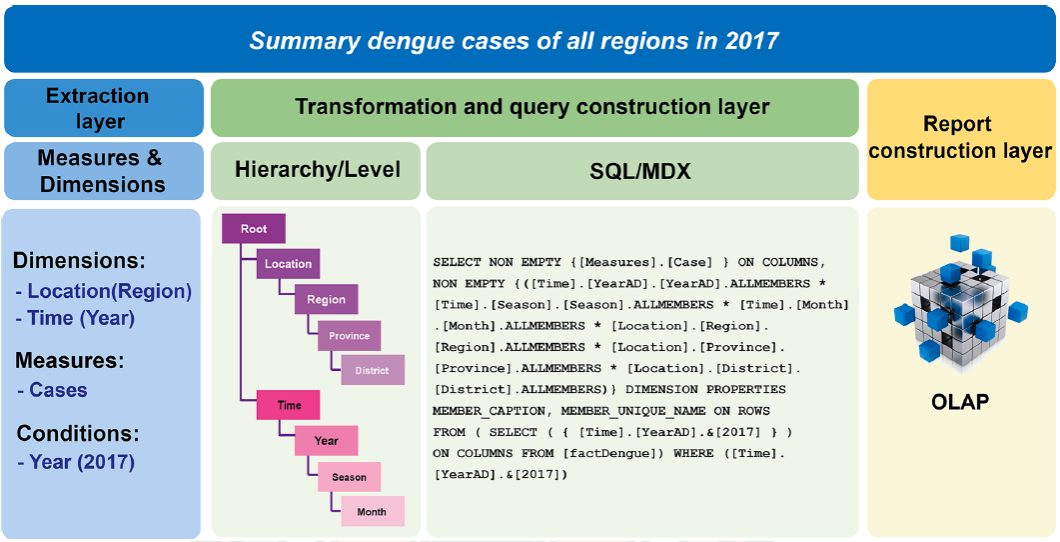

**ภาพ 42 แสดงโครงสร้างแบบดาวโดเมนทางการเกษตร**




**ภาพ 43 แสดงโครงสร้างแบบดาวโดเมนทางธุรกิจ**

**ผลการสร้างรายงานในรูปแบบ OLAP**

การสร้างรายงานในรูปแบบ OLAP จากข้อกำหนดของผู้ใช้ระบบ ผู้วิจัยได้ทำการทดลองจาก 3 โดเมนโดยใช้ข้อกำหนดของผู้ใช้ที่อยู่ในรูปแบบของประโยคจำนวน 2,000 รายการ ข้อมูลจะถูกระบุเมเชอร์และมิติด้วย spaCy และแปลเป็นภาษาสอบถามเชิงโครงสร้างโดยใช้ออนโทโลยีแสดงขั้นตอนการสร้างดังภาพ 44




**ภาพ 44 แสดงขั้นตอนในการสร้างภาษาสอบถามเชิงโครงสร้างที่สนับสนุนการรายงานแบบ OLAP**

ระบบจะทำการสร้างรายงานจากข้อกำหนดของผู้ใช้ที่อยู่ในรูปแบบประโยคจะได้เมเซอร์มิติ และเงื่อนไขของรายงาน และทำการจัดลำดับชั้นหรือระดับของมิติจากโครงสร้างของออนโทโลยี ข้อมูลเหล่านี้จะถูกป้อนเข้าสู่กระบวนการสร้างภาษาสอบถามเชิงโครงสร้าง เพื่อใช้สำหรับการสอบถามข้อมูลจากโครงสร้างแบบดาวในคลังข้อมูล ขั้นตอนสุดท้ายคือการสร้างรายงาน เป็นการสร้างรายงานในรูปแบบ OLAP ตามภาษาสอบถามเชิงโครงสร้างโดยเป็นรูปแบบที่สามารถแสดงผลบนโปรแกรม Microsoft SQL Server 2019 ผลการสร้างรายงานในรูปแบบ OLAP แสดงดังภาพ 45

| | | 2017 | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | rainy | | | | | summer | winter |
| | | August | July | June | October | September | | |
| Central | | 1,624 | 1,551 | 920 | 1,065 | 1,239 | 944 | 2,691 |
| East | | 367 | 404 | 356 | 140 | 213 | 338 | 409 |
| North | Chiang Mai | 256 | 413 | 232 | 117 | 131 | 178 | 183 |
| | Chiang Rai | 316 | 328 | 271 | 158 | 238 | 72 | 221 |
| | Lampang | 36 | 66 | 40 | 14 | 24 | 54 | 30 |
| | Lamphun | 56 | 50 | 16 | 12 | 12 | 30 | 20 |
| | Mae Hong Son | 80 | 66 | 61 | 31 | 47 | 49 | 50 |
| | Nan | 33 | 58 | 63 | 18 | 17 | 82 | 33 |
| | Phayao | 5 | 13 | 11 | 6 | 8 | 16 | 2 |
| | Phrae | 5 | 9 | 6 | 0 | 5 | 9 | 4 |
| | Uttaradit | 52 | 47 | 30 | 11 | 22 | 33 | 15 |
| Northeastern | | 1,045 | 1,336 | 1,363 | 293 | 530 | 943 | 474 |
| South | | 505 | 545 | 719 | 446 | 429 | 1,584 | 2,563 |
| West | | 432 | 482 | 323 | 173 | 264 | 176 | 386 |

**ภาพ 45 แสดงรายงานในรูปแบบ OLAP ข้อมูลการระบาดของโรคไข้เลือดออก**

จากภาพแสดงตัวอย่างรายงานในรูปแบบ OLAP ที่ได้จากคลังข้อมูลการระบาดของโรคไข้เลือดออก รายงานนี้สอดคล้องกับข้อกำหนดของผู้ใช้ โดยแสดงจำนวนผู้ป่วยโรคไข้เลือดออกแบ่งตามสถานที่และเวลา ผู้ใช้สามารถแสดงข้อมูลแบบเจาะลึก (drill down) หรือแสดงข้อมูลแบบสรุป (roll up) เพื่อแสดงรายละเอียดข้อมูลได้ตรงตามความต้องการและช่วยสนับสนุนการตัดสินใจได้

# บทที่ 5

# บทสรุป

## สรุปผลการวิจัย

งานวิจัยนี้ได้ทำการศึกษาวิธีเชิงความหมายสำหรับสร้างโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติ โดยแบ่งการทำงานออกเป็น 3 ขั้นตอน ได้แก่ การสกัดและวิเคราะห์ข้อมูล การสร้างโครงสร้างข้อมูลแบบหลายมิติ และการสกัดและโหลดข้อมูล เมื่อเปรียบเทียบกับงานวิจัยอื่นๆ ที่เป็นการสร้างโครงสร้างแบบดาวโดยใช้ออนโทโลยีเช่นเดียวกัน เทคนิคที่นำเสนอนี้สามารถสร้างโครงสร้างแบบดาวได้จากข้อกำหนดของผู้ใช้ที่อยู่ในรูปแบบภาษาธรรมชาติ ช่วยให้ระบบมีความยืดหยุ่นสามารถสร้างโครงสร้างแบบดาวขึ้นใหม่ให้ตรงกับข้อกำหนดใหม่ได้

ขั้นตอนการสกัดและวิเคราะห์ข้อมูล ระบบสามารถอนุมานชื่อคอลัมน์ในกรณีที่ไม่พบชื่อคอลัมน์หรือชื่อคอลัมน์สูญหายโดยใช้ฟังก์ชันความหนาแน่นของความน่าจะเป็น จากการเปรียบเทียบกับเทคนิคอื่นๆ อีก 3 เทคนิคได้แก่ การจำแนกแบบนาอีฟเบย์ วิธีต้นไม้ตัดสินใจ และโครงข่ายประสาทเทียม พบว่าระบบที่พัฒนาขึ้นมีค่าความถูกต้องสูงที่สุดเมื่อเทียบกับเทคนิคอื่น การอนุมานชนิดข้อมูลและขนาดของข้อมูลระบบสามารถอนุมานได้ตรงกับที่ผู้เชี่ยวชาญกำหนด มีค่าความถูกต้องที่สูงกว่าการอนุมานด้วยโปรแกรม Microsoft SQL Server 2019 การระบุเมเซอร์ด้วย spaCy พบว่าจากรูปแบบประโยคที่มีโครงสร้างอย่างง่าย ประโยคที่มีโครงสร้างที่ซับซ้อน และประโยคที่มีโครงสร้างที่ซับซ้อนร่วมกับการใช้คำศัพท์ที่แตกต่างกันพบว่า ระบบสามารถระบุเมเซอร์ได้ถูกต้อง 100.00% ทั้งสามรูปแบบ และระบบสามารถระบุมิติได้ถูกต้อง 100.00% 94.73% และ 66.67% ตามลำดับ จากขั้นตอนการสกัดและวิเคราะห์ข้อมูลพบว่า ระยะเวลาของกระบวนการสกัดและวิเคราะห์ข้อมูล โดเมนทางการเกษตรระบบใช้เวลาในการสร้างโครงสร้างแบบดาวน้อยที่สุดอยู่ที่ 15.20 วินาที

ขั้นตอนการสร้างโครงสร้างแบบหลายมิติและการโหลดข้อมูลเข้าสู่คลังข้อมูล ผลลัพธ์ที่ได้ในขั้นตอนนี้คือโครงสร้างแบบดาวที่รองรับการประมวลผลเชิงวิเคราะห์แบบออนไลน์ (OLAP) เพื่อนำเสนอข้อมูลในหลายมิติได้พบว่า ระยะเวลาการสร้างโครงสร้างแบบดาว โดเมนทางการแพทย์ใช้เวลาในการสร้างน้อยที่สุดอยู่ที่ 1.18 วินาที และระยะเวลาการนำเข้าข้อมูลพบว่าโดเมนทางการเกษตรใช้เวลาน้อยที่สุดอยู่ที่ 9.66 วินาที

ขั้นตอนการการสร้างรายงานในรูปแบบ OLAP จากข้อกำหนดของผู้ใช้ ระบบสามารถแสดงรายงานในรูปแบบ OLAP โดยผู้ใช้สามารถแสดงข้อมูลแบบเจาะลึก (Drill down) หรือแสดงข้อมูลแบบสรุป (Roll up) เพื่อแสดงรายละเอียดข้อมูลได้ตามความต้องการ และช่วยสนับสนุนการตัดสินใจ

จากกระบวนการทั้งหมดระบบสามารถช่วยลดระยะเวลาในการสร้างโครงสร้างแบบดาว และคลังข้อมูลที่ได้มีความถูกต้องและสามารถแสดงผลในรูปแบบ OLAP ได้

**อภิปรายผล**

ผลจากการสร้างโครงสร้างข้อมูลแบบหลายมิติโดยอัตโนมัติ ขั้นตอนการสกัดและวิเคราะห์ข้อมูลระบบสามารถอนุมานชื่อคอลัมน์มีค่าความถูกต้องสูงที่สุดเมื่อเทียบกับเทคนิคการจำแนกแบบนาอีฟเบย์ วิธีต้นไม้ตัดสินใจ และโครงข่ายประสาทเทียม แต่ระบบยังไม่รองรับการกระบวนการตรวจสอบความถูกต้องของข้อมูล หากเพิ่มขั้นตอนการทำความสะอาดข้อมูล (Data cleansing) โดยอัตโนมัติ ก่อนที่จะโหลดลงในคลังข้อมูล เพื่อให้แน่ใจว่าข้อมูลนำเข้าไม่มีความผิดพลาดยกตัวอย่าง เช่น การตรวจสอบข้อมูลที่สูญหายโดยอัตโนมัติและการแทนค่าข้อมูล (Kamkhad, et al., 2020) เพื่อปรับปรุงคุณภาพของข้อมูลนำเข้าและไม่ส่งผลเสียต่อความถูกต้องของกระบวนการรายงานผลในรูปแบบ OLAP การอนุมานชนิดข้อมูลและขนาดของข้อมูล ระบบสามารถอนุมานชนิดข้อมูลมีค่าความถูกต้องสูงกว่าการกำหนดด้วยโปรแกรม Microsoft SQL Server 2019 ซึ่งการกำหนดขนาดของข้อมูลที่เป็นตัวเลขในบางข้อมูลไม่ตรงกับผู้เชี่ยวชาญกำหนด เนื่องจากระบบกำหนดขนาดข้อมูลตามความยาวของข้อมูลจริงที่มีขนาดใหญ่ที่สุด แต่ผู้เชี่ยวชาญมองว่าข้อมูลบางข้อมูลอาจมีขนาดที่ใหญ่ขึ้นในอนาคตจึงกำหนดขนาดของข้อมูลใหญ่กว่าขนาดของข้อมูลจริง การระบุเมเชอร์ด้วย spaCy พบว่าระบบระบุมิติได้ค่าความถูกต้องลดลงเมื่อโครงสร้างประโยคมีความซับซ้อนมากขึ้นเนื่องจากคำศัพท์ที่เป็นมิติมีหลากหลายคำและ spaCy ไม่สามารถเรียนรู้ความหมายของคำศัพท์ที่เป็นคำพ้องความหมายได้ หากระบบสามารถเรียนรู้คำพ้องความหมายหรือทำการสอนระบบด้วยรูปแบบของประโยคที่มีความหลากหลายมากขึ้นจะช่วยให้ระบบสามารถระบุมิติได้มีความถูกต้องมากขึ้น จากขั้นตอนการสกัดและวิเคราะห์ข้อมูล ระยะเวลาการสกัดวิเคราะห์ข้อมูลขึ้นอยู่กับจำนวนคอลัมน์ที่นำมาวิเคราะห์ เนื่องจากระบบจะทำการอ่านข้อมูลจากคอลัมน์ทั้งหมดในตารางข้อมูล เมื่อจำนวนคอลัมน์มีจำนวนมากระบบจะใช้เวลาในการวิเคราะห์มากกว่าตารางที่มีจำนวนคอลัมน์ที่น้อยกว่า ซึ่งหากมีกระบวนการตัดบางคอลัมน์ที่ไม่เกี่ยวข้องกับการสร้างโครงแบบหลายมิติจะช่วยให้ระยะเวลาของการสร้างทำได้รวดเร็วยิ่งขึ้น

ขั้นตอนการสร้างโครงสร้างแบบหลายมิติและการโหลดข้อมูลเข้าสู่คลังข้อมูล กระบวนการสร้างโครงสร้างแบบดาว ระบบจะทำการสร้างตารางมิติ สร้างตารางตารางข้อเท็จจริง และสร้างความสัมพันธ์ระหว่างตารางมิติและตารางตารางข้อเท็จจริง หากมีจำนวนคอลัมน์และความสัมพันธ์ระหว่างตารางมากระบบจะต้องใช้ระยะเวลาในการสร้างโครงสร้างแบบดาวมากขึ้น กระบวนการนำเข้าข้อมูล ระบบจะทำการสกัดข้อมูลจากตารางข้อมูลและตรวจสอบข้อมูลที่ซ้ำกันหากมีข้อมูลที่ซ้ำกันระบบจะทำการตัดข้อมูลที่ซ้ำกันออกก่อนทำการถ่ายโอนข้อมูลเข้าสู่ตารางในคลังข้อมูลตามชื่อ

คอลัมน์ที่ตรงกัน ซึ่งระบบจะทำการอ่านข้อมูลในทุกระเบียนหากข้อมูลมีปริมาณมากจะส่งผลทำให้ใช้ระยะเวลามากขึ้นตามปริมาณของข้อมูลในตารางข้อมูล ดังนั้นระยะเวลาการสร้างโครงสร้างแบบดาวในภาพรวมพบว่า โดเมนทางการเกษตรระบบใช้เวลาในการสร้างโครงสร้างแบบดาวน้อยที่สุด

ขั้นตอนการการสร้างรายงานในรูปแบบ OLAP ระบบสามารถแสดงรายงานในรูปแบบ OLAP โดยมีรายละเอียดข้อมูลตรงตามข้อกำหนดของผู้ใช้ หากการแสดงผลระบบสามารถแสดงผลข้อมูลในรูปแบบอื่น ๆ เช่น แผนภูมิและแผนที่ เป็นต้น จะช่วยให้ผู้ใช้สามารถวิเคราะห์ข้อมูลที่มีจำนวนมากและทำการตัดสินใจจากข้อมูลได้มีประสิทธิภาพมากขึ้น

**ข้อเสนอแนะ**

1. การทำความสะอาดข้อมูล (Data cleansing) โดยอัตโนมัติ กระบวนการตรวจสอบความถูกต้องของข้อมูลก่อนที่จะโหลดลงในคลังข้อมูล เพื่อให้แน่ใจว่าข้อมูลนำเข้าไม่มีข้อผิดพลาดที่จะส่งผลเสียต่อความถูกต้องของกระบวนการรายงานในรูปแบบ OLAP เช่น การตรวจสอบข้อมูลที่สูญหายไปโดยอัตโนมัติและการแทนค่าข้อมูล (Kamkhad, Jampachaisri, Siriyasatien & Kesorn, 2020) เพื่อปรับปรุงคุณภาพของข้อมูลนำเข้า

2. การแสดงผลข้อมูลในรูปแบบอื่น ๆ ที่ช่วยให้ผู้ใช้สามารถวิเคราะห์ข้อมูลที่มีจำนวนมากและทำการตัดสินใจจากข้อมูลได้มีประสิทธิภาพมากขึ้น เช่น แผนภูมิและแผนที่ เป็นต้น ซึ่งรูปแบบการแสดงผลสารสนเทศที่เหมาะสมช่วยให้ทราบถึงแนวโน้ม ค่าที่ผิดปกติ และลักษณะของข้อมูลจึงมีความจำเป็นสำหรับผู้มีอำนาจตัดสินใจ

3. การทำงานกับข้อมูลที่ไม่มีโครงสร้าง ปัจจุบันระบบรองรับรูปแบบข้อมูลแบบกึ่งโครงสร้างเท่านั้น ดังนั้นหากสามารถรองรับข้อมูลที่ไม่มีโครงสร้าง ระบบจะสามารถประมวลผลข้อมูลที่มีขนาดใหญ่มากขึ้น เนื่องจากปัจจุบันมีข้อมูลจำนวนมากที่อยู่ในรูปแบบที่ไม่มีโครงสร้าง

4. การใช้ออนโทโลยีทั่วไปที่เป็นข้อมูลพื้นฐานที่เกี่ยวข้องกับโดเมนที่สนใจ เช่น ข้อมูลสถานที่ (Location) ข้อมูลเวลา (Time) ที่สามารถใช้ร่วมกับโดเมนออนโทโลยีที่หลากหลายโดเมนได้ ช่วยให้การสร้างคลังข้อมูลที่เป็นข้อมูลทั่วไปสามารถทำได้รวดเร็วขึ้น และออนโทโลยีที่สร้างขึ้นสามารถนำกลับมาใช้ใหม่กับโดเมนที่สนใจได้

5. การสร้างโครงสร้างแบบเกล็ดหิมะและโครงสร้างแบบกาแล็กซี่ ปัจจุบันระบบรองรับเฉพาะการสร้างโครงสร้างแบบดาวเท่านั้น ซึ่งโครงสร้างแบบดาวเป็นชุดของตารางข้อเท็จจริงที่มีความสัมพันธ์กับตารางมิติแบบหนึ่งต่อกลุ่ม แต่โครงสร้างแบบเกล็ดหิมะจะขยายความสัมพันธ์แบบหนึ่งต่อกลุ่มระหว่างตารางมิติ และโครงสร้างแบบกาแล็กซี่จะใช้ตารางข้อเท็จจริงสองตารางที่ใช้ตารางมิติร่วมกัน ดังนั้นรูปแบบดังกล่าวในบางสถานการณ์สามารถนำมาใช้ได้เหมาะสมและมีประสิทธภาพมากกว่าโครงสร้างแบบดาว

# บรรณานุกรม


กิตติพงศ์ กลมกล่อม. (2552). *การออกแบบและพัฒนาคลังข้อมูล (data warehouse)*. กรุงเทพฯ: เคที พี คอมพ์ แอนด์ คอนซัลท์.

มาลี กาบมาลา, ลำปาง แม่นมาตย์, และครรชิต มาลัยวงศ์. (2556). ออนโทโลยี: แนวคิดการพัฒนา. *วารสารสารสนเทศศาสตร์*, *31*(1), 108-140.

Abdalaziz Ahmedl, R., & Mohamed Ahmed, T. (2014). Generating data warehouse schema. *International Journal in Foundations of Computer Science & Technology, 4*(1), 1-16. doi: 10.5121/ijfcst.2014.4101

Anon. (2012). *Owl - semantic web standards.* Retrieved December 7, 2021, from https://www.w3.org/2001/sw/wiki/OWL

Antoniou, G., Groth, P., Harmelen, F. V., Hoekstra, R., & Yu, E. (2012). *A semantic web primer* (3rd ed.). Cambridge, Mass: MIT Press.

Armbrust, M., Xin, R. S., Lian, C., Huai, Y., Liu, D., Bradley, J. K., . . . Zaharia, M. (2015). Spark sql: Relational data processing in spark. In *2015 ACM SIGMOD International Conference on Management of Data* (pp. 1383–1394). Melbourne, Victoria, Australia: Association for Computing Machinery.

Bentayeb, F., Maïz, N., Mahboubi, H., Favre, C., Loudcher, S., Harbi, N., . . . Darmont, J. (2013). Innovative approaches for efficiently warehousing complex data from the web. In M. Khosrow-Pour (Ed.), In *Data mining: Concepts, methodologies, tools, and applications* (pp. 1422-1448). Pennsylvania, PA: IGI Global.

Bowman, A., & Azzalini, A. (1997). *Applied smoothing techniques for data analysis: The kernel approach with s-plus illustrations* (Vol. 94). Oxford: OUP Oxford.

Chakiri, H., El Mohajir, M., & Assem, N. (2020). A data warehouse hybrid design framework using domain ontologies for local good-governance assessment. *Transforming Government: People, Process and Policy, 14*(2), 171-203. doi: 10.1108/TG-04-2019-0025

Chaudhuri, S., & Dayal, U. (1997). An overview of data warehousing and olap technology. *SIGMOD Rec.,* 26, 65-74.

Elamin, E., Alzaidi, A., & Feki, J. (2018). A semantic resource based approach for star

schemas matching. *International Journal of Database Management Systems, 10(*6), 15-28. doi: 10.5121/IJDMS.2018.10602

Elamin, E., & Feki, J. (2014). Toward an ontology based approach for data warehousing state of the art and proposal. In *The international Arab conference on information technology (ACIT2014)* (pp. 170-179). Zarqa: The International Arab Conference on Information Technology

Euzenat, J., & Shvaiko, P. (2013). *Ontology matching* (2). Berlin Heidelberg: Springer-Verlag.

Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition, 5(*2), 199-220. doi: 10.1006/knac.1993.1008

Guarino, N. (1998). *Formal ontology in information systems: Proceedings of the* 1*st international conference june* 6-8*,* 1998*, trento, italy* (1st). NLD: IOS Press.

Gulić, M. (2013). Transformation of owl ontology sources into data warehouse. In *2013 36th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (pp. 1143-1148). Opatija, Croatia: IEEE.

Hansen, J. B., Jensen, S., Tarp, M., & Thomsen, C. (2017). *Dwstar - automated star schema generation* (Master's Thesis). Denmark: AALBORG university.

Howard, P. G., & Vitter, J. S. (1992). Analysis of arithmetic coding for data compression. *Information Processing & Management, 28*(6), 749-763. doi: 10.1016/0306-4573(92)90066-9

Jensen, M. R., Holmgren, T., & Pedersen, T. B. (2004). Discovering multidimensional structure in relational data. In *6th International Conference, DaWaK 2004* (pp. 138-148). Zaragoza: Springer Berlin Heidelberg.

Kamkhad, N., Jampachaisri, K., Siriyasatien, P., & Kesorn, K. (2020). Toward semantic data imputation for a dengue dataset. *Knowledge-Based Systems,* 196, 105803. doi: 10.1016/j.knosys.2020.105803

Khouri, S., Boukhari, I., Bellatreche, L., Sardet, E., Jean, S., & Baron, M. (2012). Ontology-based structured web data warehouses for sustainable interoperability: Requirement modeling, design methodology and tool. *Computers in industry, 63*(8), 799-812. doi: 10.1016/j.compind.2012.08.001

Liu, X., & Iftikhar, N. (2013). Ontology-based big dimension modeling in data warehouse

schema design. In *16th International Conference* (pp. 75-87). Heidelberg: Springer Berlin Heidelberg.

Lumbantoruan, R., Sibarani, E., Sitorus, M., Mindari, A., & Sinaga, S. (2014). An approach for automatically generate star schema from natural language. *TELKOMNIKA (Telecommunication Computing Electronics and Control), 12*(2), 501-510. doi: 10.12928/telkomnika.v12i2.63

McGuinness, D. L., & Van Harmelen, F. (2004). *OWL web ontology language overview. W3C recommendation, 10*(10), 2004. https://static.twoday.net/71desa1bif/files/W3C-OWL-Overview.pdf

Meersman, R. A. (1999). Semantic ontology tools in is design. In Z. W. Raś & A. Skowron (Eds.), *Lecture notes in computer science* (pp. 30-45). Berlin: Springer.

Miller, G. A. (1995). Wordnet: A lexical database for english. *Communications of the ACM, 38*(11), 39-41. doi: 10.1145/219717.219748

Nebot, V., Berlanga, R., Pérez, J. M., Aramburu, M. J., & Pedersen, T. B. (2009). Multidimensional integrated ontologies: A framework for designing semantic data warehouses. In S. Spaccapietra, E. Zimányi & I. Song (Eds.), *Lecture notes in computer science* (Vol. 5530, pp. 1-36). Berlin: Springer, Berlin, Heidelberg.

Noy, N. F., & McGuinness, D. L. (2001). Ontology development 101: A guide to creating your first ontology.

Otero-Cerdeira, L., Rodríguez-Martínez, F. J., & Gómez-Rodríguez, A. (2015). Ontology matching: A literature review. *Expert Systems with Applications, 42*(2), 949-971. doi: 10.1016/j.eswa.2014.08.032

Pardillo, J., & Mazón, J.-N. (2011). Using ontologies for the design of data warehouses. *International Journal of Database Management Systems, 3*(2), 73-87. doi: 10.5121/ijdms.2011.3205

Phipps, C., & Davis, K. C. (2002). Automating data warehouse conceptual schema design and evaluation. In *4th International Workshop on Design and Management of Data Warehouses* (pp. 23-32). Toronto: CEUR-WS.org.

Power, D. J. (2002). *Decision support systems: Concepts and resources for managers*. Westport, CT: Greenwood Publishing Group.

Romero, O., & Abelló, A. (2010). A framework for multidimensional design of data warehouses from ontologies. *Data & Knowledge Engineering, 69*(11), 1138-1157. doi: 10.1016/j.datak.2010.07.007

Romero, O., Simitsis, A., & Abelló, A. (2011). *Gem: Requirement-driven generation of etl and multidimensional conceptual designs*. In *Data Warehousing and Knowledge Discovery* (pp. 80-95). Berlin: Springer.

Sehgal, S., & Ranga, K. K. (2016). Translation of entity relational model to dimensional model. *International Journal of Computer Science and Mobile Computing, 5*(5), 439-447.

Selma, K., Ilyès, B., Ladjel, B., Eric, S., Stéphane, J., & Michael, B. (2012). Ontology-based structured web data warehouses for sustainable interoperability: Requirement modeling, design methodology and tool. *Computers in industry, 63*(8), 799-812.

Siriyasatien, P., Chadsuthi, S., Jampachaisri, K., & Kesorn, K. (2018). Dengue epidemics prediction: A survey of the state-of-the-art based on data science processes. *IEEE Access,* 6, 53757-53795. doi: 10.1109/ACCESS.2018.2871241

Song, I. Y., Khare, R., & Dai, B. (2007). Samstar: A semi-automated lexical method for generating star schemas from an entity-relationship diagram. In *Proceedings of the ACM tenth international workshop on Data warehousing and OLAP* (pp. 9-16). New York: Association for Computing Machinery.

Thenmozhi, M., & Vivekanandan, K. (2012). An ontology based hybrid approach to derive multidimensional schema for data warehouse. *International Journal of Computer Applications, 54*(8), 36-42.

Usman, M., Pears, R., & Fong, A. C. M. (2012). Data guided approach to generate multi-dimensional schema for targeted knowledge discovery. In *10th Australasian Data Mining Conference (AusDM 2012)* (pp. 229-240). Sydney: Information Technology.

Witten, I. H., Neal, R. M., & Cleary, J. G. (1987). Arithmetic coding for data compression. *Communications of the ACM, 30*(6), 520-540. doi: 10.1145/214762.214771