บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของงานวิจัย

ด้วยการพัฒนาที่รวดเร็วของเทคโนโลยีอินเตอร์เน็ต ทำให้ข้อมูลข่าวสารบนเว็บเติบโตมากขึ้น ในทุกวันจำนวนข้อมูลของหน้าเว็บที่เพิ่มขึ้นนั้นจะมีเนื้อหาสาระใหม่ๆ เพื่อให้เว็บนั้นได้รับความสนใจ ซึ่งหน้าเว็บเหล่านี้เต็มไปด้วยข้อมูลสารสนเทศ และที่ยิ่งไปกว่านั้นองค์ประกอบของหน้าเว็บเหล่านั้นทำให้ไม่สามารถค้นหาได้โดยง่าย จากการสืบค้นข้อมูลโดยเสิร์ชเอ็นจิน (Search Engine) ที่มีอยู่ในปัจจุบัน ซึ่งได้ผลลัพธ์ที่ไม่ตรงกับความต้องการของผู้ใช้งาน จากความต้องการและความสนใจที่หลากหลายของผู้ใช้งาน ทำให้เป็นปัญหาต่อเสิร์ชเอ็นจินที่มีอยู่ในปัจจุบัน เช่น กูเกิล (Google) ยาฮู (Yahoo) เอ็มเอสเอ็น (MSN) และ อื่นๆ นั้นแสดงผลลัพธ์ในการค้นหาแต่ละครั้งได้ผลลัพธ์จำนวนมาก ทั้งข้อมูลที่ต้องการและไม่ต้องการ ผู้ใช้งานจะทำการตัดสินใจเลือกเอาเฉพาะข้อมูลที่ต้องการใหม่อีกครั้งหรือหลายๆ ครั้งจนกระทั่งได้ข้อมูลที่ตรงตามความต้องการจริง จึงทำให้เสียเวลาในการคัดเลือกเอกสารนั้นมาก จากงานวิจัยของ Arul Prakas Asirvatham และคณะ (2001) กล่าวว่ามันเป็นเรื่องยากมากสำหรับผู้ใช้งานที่จะค้นหาเอกสารที่ต้องการโดยมีความสัมพันธ์กับข้อมูลเพียงบางส่วนที่เขากำลังค้นหาอยู่ จะมีประโยชน์ยิ่งถ้าหน้าเว็บสามารถใช้ประโยชน์ได้อย่างเต็มศักยภาพ หากมีวิธีที่ทำให้การจำแนกข้อมูลที่มีขนาดใหญ่มากนั้นมีความถูกต้องและมีประสิทธิผล จึงทำให้มีแนวคิดที่จะนำมาช่วยแก้ปัญหาเหล่านี้

ที่ผ่านมาการจำแนกหน้าเว็บจะทำด้วยแรงงานคนโดยผู้เชี่ยวชาญเฉพาะด้านนั้นๆ แต่การจัดหมวดหมู่หน้าเว็บด้วยความคิดของคนจากการเห็นในครั้งแรกนั้นมักจะอยู่ในสองสามหมวดหมู่อย่างกว้างๆ แต่ไม่ได้ดูจากเนื้อหาของหน้าเว็บนั้นจริงๆ ซึ่งใช้คุณสมบัติอื่น เช่นโครงสร้างของหน้าเว็บ รูปภาพ ลิงก์ที่มีอยู่ในหน้าเว็บ การจัดตำแหน่ง เป็นต้น แต่ล่าสุดการจำแนกสามารถทำโดยกึ่งอัตโนมัติหรืออัตโนมัติ บางงานวิจัยนั้นใช้การจำแนกข้อความ (Text Categorization) และการเรียนรู้ของเครื่อง (Machine Learning) ซึ่งวิธีการเรียนรู้นั้นจะใช้โครงสร้างลำดับชั้น (Hierarchy Structure) ทำให้จำแนกจากเนื้อหาหน้าเว็บตามโครงสร้างลำดับชั้นได้ง่าย โดยมีเนื้อหาบนหน้าเว็บทั้งข้อความ รูปภาพ ภาพเคลื่อนไหว เพลง ซอฟต์แวร์ แผนที่ ข้อมูลบุคคล กลุ่ม

ข่าว และอื่นๆ โดยที่เนื้อหาและโครงสร้างของเอกสารที่มีข้อมูลมากมายเหล่านั้นสามารถนำมาช่วยในการจำแนกหน้าเว็บได้เช่นกัน

ปัจจุบันเทคนิคในการสืบค้นของเสิร์ชเอ็นจินในการจำแนกที่มีอยู่นั้นอาศัยเพียงเนื้อหาของข้อความในการจำแนกเท่านั้น โดยขึ้นอยู่กับการค้นหาคำที่ผู้ใช้ต้องการสืบค้นจากคำที่ตรงหรือสอดคล้องกับคำสำคัญ (Keyword) แม้ว่าหน้าเว็บเขียนด้วยภาษาเอชทีเอ็มแอล (HTML) จะมีแท็กเมตา (Meta Tag) สำหรับใส่คำหลักไว้ในแต่ละเอกสารโดยเจ้าของเว็บไซต์ เพื่อต้องการให้สืบค้น แต่คำหลักที่ถูกใส่ในแต่ละเอกสาร อาจจะใส่คำหลักที่เหมือนกันแต่มีความหมายแตกต่างกันออกไป โดยไม่ได้ระบุถึงความหมายของคำ ความคล้ายคลึงกันของคำ หรือความสัมพันธ์ของคำแต่ละคำ ทำให้เสิร์ชเอ็นจินค้นหาเอกสารที่ตรงกับแนวความคิด (Concept) และความต้องการของผู้สืบค้นได้ยากยิ่งขึ้น

เนื่องจากวิธีการวัดความคล้ายกัน (Similarity) ของหน้าเว็บจากคำสำคัญที่กำหนดไว้ที่หน้าเว็บนั้นๆ โดยการเปรียบเทียบจากคำว่าเหมือนกันหรือไม่ แต่ประเด็นสำคัญที่น่าจะนำมาพิจารณาอย่างยิ่งคือ ปัญหาเรื่องคำและความหมายของคำที่มีความเกี่ยวข้อง และนำเอาความรู้จากการนิยามคำศัพท์แนวคิด (Ontology) ว่าด้วยการจัดกลุ่มสรรพสิ่งต่างๆ บนโลกนี้ โดยใช้วิธีการแยกความแตกต่าง แล้วนำมาจัดกลุ่ม เพื่อง่ายต่อการสืบหา และอ้างอิง เช่น การกล่าวถึงแจ๊ส (Jazz) คือแนวดนตรี ก็จะจัดกลุ่มให้แจ๊สเป็นหนึ่งในสาขาต่างๆ ของดนตรี แล้วถ้ากล่าวว่าดิกซีแลนด์ (Dixieland) คือแจ๊สแนวหนึ่ง ดิกซีแลนด์ก็จะเป็นหนึ่งในสาขาต่างๆ ของแจ๊สอีกที หากกำหนดว่าดิกซีแลนด์คือเพลงแจ๊สยุคแรก กำเนิดที่นิวออร์ลีน ทำให้ได้คุณสมบัติ (Properties) ของดิกซีแลนด์ 2 ประการคือ ช่วงเวลาที่เกิด และสถานที่เกิด การจัดเช่นนี้ทำให้เกิดความสัมพันธ์ลำดับชั้นของข้อมูล (Hierarchy) ที่สามารถนำไปใช้ประโยชน์ในการบริหารความรู้ด้วยคอมพิวเตอร์ได้ ศาสตร์วิชานี้ได้นำแนวคิดการนิยามคำศัพท์แนวคิดซึ่งเป็นปรัชญาทางภาษาศาสตร์ แต่ที่แตกต่างกันคือ นักภาษาศาสตร์ในปัจจุบันมุ่งเน้นไปที่การทำให้คำ กลุ่มคำ คุณลักษณะต่างๆ มีความคลุมเครือมากขึ้น ทำให้หาคำจำกัดความและการจัดกลุ่มได้ยาก และนี่คือปัญหาของเว็บเชิงความหมาย เพราะคำต่างๆ ในภาษาล้วนมีการเปลี่ยนแปลงในด้านความหมาย และแต่ละคำนั้นจะขึ้นอยู่กับการใช้ของแต่ละคนที่ต้องการค้นหา

ผู้วิจัยเห็นว่า หากจะอธิบายถึงโครงสร้างของวัตถุนั้น สามารถนำกราฟมาใช้ในการแสดงข้อมูลที่มีความสัมพันธ์กัน ซึ่งกราฟมีประโยชน์มากสำหรับงานด้านวิทยาศาสตร์ และวิศวกรรม มีโปรแกรมประยุกต์ เช่น การรู้จำแบบ (Pattern Recognition), การเรียนรู้ของเครื่อง และ การค้นคืนข้อมูลสารสนเทศ (Information Retrieval) ซึ่งใช้วัดความเหมือนกันของวัตถุ

(object) โดยการเทียบความเหมือนกันของกราฟ มีบางแนวคิดที่เกี่ยวกับเครื่องมือวัดความเหมือนกันของกราฟ หรือ สมสัณฐานของกราฟ (Graph Isomorphism) คือ กราฟสองกราฟที่มีโครงสร้างที่เหมือนกันทุกประการ นอกจากนั้นคือสมสัณฐานกราฟย่อย (Subgraph Isomorphism) คือกราฟหนึ่งกราฟมีบางส่วนที่อยู่ในกราฟอีกอันหนึ่ง โดยการหากราฟย่อยที่ใหญ่ที่สุดของกราฟนั้น (Maximum Common Subgraph) เป็นอัลกอริธึมที่ใช้การค้นหาแบบย้อนรอย (Backtrack) ซึ่งเป็นปัญหาระดับเอ็นพีบริบูรณ์ (NP-Complete)

จากปัญหาดังกล่าวมาข้างต้น จึงมีนักวิจัยหลายท่านคิดค้นวิธีการสืบค้นข้อมูลได้ตรงกับความต้องการของผู้ใช้โดยการเพิ่มส่วนอธิบายความหมายเข้าไปในเว็บไซต์ แต่วิธีดังกล่าวไม่สามารถนำมาใช้ร่วมกันได้ เนื่องจากต่างคนต่างพัฒนาโครงสร้างข้อมูลของตนเอง ทำให้ขาดความเข้ากันได้ของข้อมูล (Interoperability) ภายหลังองค์กร W3C (WWW Consortium) ได้พัฒนาภาษามาตรฐานเอ็กซ์เอ็มแอล (XML หรือ eXtensive Markup Language) ขึ้น เพื่อใช้เป็นมาตรฐานที่สื่อความหมายของข้อมูลและแลกเปลี่ยนข้อมูลผ่านอินเตอร์เน็ต โดยมีแท็กแสดงความหมายของข้อมูลในแท็กนั้น เช่น <sport>Basketball</sport> สำหรับอธิบายว่า Basketball เป็นกีฬาชนิดหนึ่ง แต่ยังไม่แสดงถึงความหมายได้อย่างชัดเจน ต่อมาองค์กร W3C ได้พัฒนามาตรฐานใหม่ที่เรียกว่าเว็บเชิงความหมาย เพื่อสามารถเชื่อมโยงเครือข่ายของข้อมูลที่อยู่บนเว็บไซต์โดยช่วยให้ค้นหาข้อมูลอย่างมีประสิทธิภาพมากขึ้น และยังสามารถสร้างความสัมพันธ์ใหม่กับข้อมูลที่มาจากแหล่งข้อมูลที่ต่างกันซึ่งก่อให้เกิดเป็นฐานข้อมูลที่ถูกเชื่อมโยงกันทั่วโลก ทั้งยังขยายกลไกบนเว็บให้ไปในแนวทางที่ทำให้เอกสารมีความเหมือนของข้อมูลมากยิ่งขึ้น แนวทางแก้ปัญหาดังกล่าวจะมีกลไกที่สำคัญคือ การอธิบายข้อมูลด้วยเมตาดาตา (Metadata) โดยข้อความในแต่ละส่วนของโครงสร้างเมตาดาตาจะต้องทำการตกลงกันสำหรับสร้างข้อมูลในความหมาย (Context) ที่ตรงกับความสนใจ สำหรับการสืบค้นข้อมูลจากเว็บเชิงความหมายจะสามารถนำไปใช้วิเคราะห์ เพื่อให้ได้ผลลัพธ์ที่ตรงกับความต้องการของผู้สืบค้น และภาษาที่เข้ามาช่วยในการพัฒนาหน้าเว็บให้สามารถใช้งานเชิงความหมายคือ RDF (Resource Description Framework) และ OWL (Web Ontology Language)

ในงานศึกษาวิจัยของภูวดล เรืองกิจภิญโญ (2549) นั้นได้เสนอการใช้ภาษาข้อมูลเอกสาร OWL (OWL Instance) มาใช้เพื่อแสดงการนิยามของคำศัพท์แนวคิด และนำมารวมกับข้อมูลใน OWL ซึ่งเป็นข้อมูลที่อยู่ในรูปแบบโครงสร้างความสัมพันธ์ของข้อมูลโดยแสดงในรูปแบบกราฟ เพื่อค้นหาข้อมูลที่อยู่ในกลุ่มความหมายเดียวกันหรือใกล้เคียงกัน โดยการหาความสมสัณฐานของกราฟ (Isomorphism) แต่พบปัญหาว่าการค้นหาหน้าเว็บจากโครงสร้างของเว็บไซต์

เมื่อเทียบกับวิธีการค้นหาโดยการนับจำนวนคำนั้นไม่มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติ และอีกอุปสรรคหนึ่งคือปัญหาของเว็บเชิงความหมายนั้นยังไม่เป็นที่แพร่หลายเนื่องจากข้อมูลของเว็บเชิงความหมายนั้นจำเป็นต้องให้ผู้เชี่ยวชาญในความรู้เฉพาะด้านนั้นเป็นผู้กำหนดขึ้นมา

วิทยานิพนธ์นี้จึงนำแนวคิดและปัญหาของงานวิจัยข้างต้นมาประยุกต์ใช้และเพิ่มประสิทธิภาพในการค้นหาความเหมือนกันของหน้าเว็บ จากโครงสร้างข้อมูลภาษาที่สามารถอธิบายความหมายของข้อมูลและความสัมพันธ์ของคำจากพจนานุกรมเชิงความหมายเวิร์ดเน็ต (WordNet) ซึ่งแหล่งรวบรวมคำศัพท์ทั้งความหมายของคำศัพท์ ที่มีความสัมพันธ์ของคำในเชิงความหมาย เช่น คำแม่กลุ่ม (Hypernym), คำร่วมกลุ่ม (Hyponym) และลักษณะความหมาย (Sense key) ดังนั้นจึงเสนอการสร้างความสัมพันธ์ของคำสำคัญที่เป็นคำนามตามแนวความคิดเพื่อนำมากำหนดให้เป็นกราฟความสัมพันธ์ของคำศัพท์แนวคิด (Relation Graph) ของข้อมูลอยู่ในรูปแบบของโครงสร้างความสัมพันธ์ เพื่อที่จะให้คอมพิวเตอร์นั้นสามารถประมวลผลจากข้อมูลนั้นได้ โดยการนำโปรแกรมตรรกะเชิงอุปนัย (Inductive Logic Programming) มาเป็นเครื่องมือในการเรียนรู้เพื่อจำแนกข้อมูลและหาความสัมพันธ์ของข้อมูลที่มีความเหมือนหรือคล้ายคลึงกัน โดยการเรียนรู้จากตัวอย่างบวก (Positive Example) และตัวอย่างลบ (Negative Example) โดยใช้ความรู้ภูมิหลัง (Background Knowledge) มาเป็นข้อมูลในการเรียนรู้ และสร้างกฎในการแยกข้อมูลตัวอย่างออกจากกัน

1.2 วัตถุประสงค์

1.2.1 เพื่อเสนอการนำข้อมูบนเว็บมาอธิบายข้อมูลด้วยภาษาโปรล็อก

1.2.2 เพื่อศึกษาการจำแนกข้อมูลจากเว็บโดยใช้โปรแกรมตรรกะเชิงอุปนัย

1.2.3 เพื่อศึกษาการจำแนกข้อมูลจากเว็บเชิงความหมายโดยสืบค้นจากการพิจารณาโครงสร้างตามความหมายของข้อมูลที่เหมือนกัน โดยใช้วิธีการแก้ปัญหาความสมสัณฐานของกราฟย่อย

1.2.4 เปรียบเทียบผลลัพธ์ของความถูกต้องในการจำแนกข้อมูลหน้าเว็บ

1.3 ขอบเขตของงานวิจัย

วิทยานิพนธ์นี้เสนอวิธีการในการจำแนกหน้าเว็บด้วยการนำโปรแกรมตรรกะเชิงอุปนัยมาเป็นเครื่องมือในการเรียนรู้ โดยมีขอบเขตของงานวิจัยดังต่อไปนี้

1.3.1 ชุดข้อมูลที่ใช้ในการทดลอง มาเขียนด้วยกราฟความสัมพันธ์คำศัพท์แนวคิดและเพิ่มความสัมพันธ์กันในเชิงความหมายของคำ จากพจนานุกรมเชิงความหมายเวิร์ดเน็ต (WordNet)

1.3.2 งานวิจัยทดลองบนเครื่องคอมพิวเตอร์ที่มีหน่วยประมวลผลที่ความเร็ว 1.6 GHz ขนาดของหน่วยความจำ 1 GB

1.3.3 ระบบปฏิบัติการที่ใช้เป็น Microsoft Windows XP professional SP2

1.3.4 เครื่องมือในการสร้างกฎด้วยวิธีการโปรแกรมตรรกะเชิงอุปนัย ALEPH

1.3.5 นำข้อมูลจากกราฟคำศัพท์แนวความคิดแปลงอยู่ในรูปแบบของความรู้ภูมิหลังเพื่อค้นหาความเหมือน และความใกล้เคียงกันของข้อมูล

1.3.6 เปรียบเทียบประสิทธิภาพ โปรแกรมตรรกะเชิงอุปนัยกับการหาความสมสัณฐานระหว่างกราฟย่อย

1.3.7 การวัดความถูกต้องโดยจะทำการแบ่งข้อมูลตัวอย่างเป็นส่วนๆ แล้วจะนำมาทำการเรียนรู้ทีละรอบ จากนั้นจะทำการบันทึกผลที่ได้ทั้งหมดมาเปรียบเทียบกับการเรียนรู้กับเซตข้อมูลทั้งหมด

1.3.8 ใช้วิธีการทดสอบความถูกต้องแบบไขว้ข้ามสิบกลุ่ม (Ten-fold Cross-Validation) ในการประมาณค่าความผิดพลาด

1.4 ผลที่คาดว่าจะได้รับ

วิทยานิพนธ์นี้ได้เสนอผลของการวิจัยว่าจะสามารถจำแนกหน้าเว็บโดยโปรแกรมตรรกะเชิงอุปนัยอย่างถูกต้องแม่นยำ เพิ่มประสิทธิภาพและลดเวลาในการค้นหาข้อมูลที่ตรงตามความต้องการของผู้ใช้ได้

1.5 รายละเอียดของวิทยานิพนธ์

ในส่วนของวิทยานิพนธ์แบ่งออกเป็นส่วนต่างๆ ดังนี้

บทที่ 1 บทนำ ประกอบด้วย ความเป็นมาและความสำคัญของงานวิจัยนี้ วัตถุประสงค์ ขอบเขตของงานวิจัย ผลที่คาดว่าจะได้รับ และรายละเอียดของวิทยานิพนธ์

บทที่ 2 ทฤษฎี และงานวิจัยที่เกี่ยวข้อง ประกอบด้วย ความรู้พื้นฐานที่เกี่ยวข้องกับวิทยานิพนธ์ฉบับนี้ และงานวิจัยที่เกี่ยวกับการจัดการกฎของโปรแกรมตรรกะเชิงอุปนัย

บทที่ 3 การดำเนินการวิจัย ประกอบไปด้วย โครงสร้างของระบบ วิธีการทดลองเปรียบเทียบที่ใช้ในวิทยานิพนธ์ฉบับนี้ และ ข้อมูลที่ใช้เป็นตัวอย่างในการทดลองต่างๆ

บทที่ 4 ผลการทดลอง ประกอบด้วยวิธีการทำการทดลอง ชุดข้อมูลที่ทำการทดลอง และผลที่ได้จาการทดลอง

บทที่ 5 สรุปผลการศึกษาและข้อเสนอแนะ

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

จากที่ได้กล่าวถึงไว้ในบทนำ วิทยานิพนธ์ฉบับนี้เกี่ยวข้องกับการจำแนกหน้าเว็บเพื่อหาหน้าเว็บที่คล้ายคลึงกัน ซึ่งมีเนื้อหาที่เกี่ยวข้องหลายส่วนด้วยกัน โดยบทนี้จะกล่าวถึงทฤษฎีที่เกี่ยวข้องและงานวิจัยที่เกี่ยวข้องในวิทยานิพนธ์ฉบับนี้

2.1 ทฤษฎีที่เกี่ยวข้อง (Theory)

วิทยานิพนธ์นี้นำเสนอการจำแนกหน้าเว็บโดยการเพิ่มคุณลักษณะพิเศษของความสัมพันธ์เชิงความหมาย ซึ่งมีทฤษฎีที่เกี่ยวข้องในวิทยานิพนธ์ฉบับนี้ ครอบคลุมเนื้อหาได้แก่

1) การสืบค้นข้อมูลบนเว็บ (Search Engine)
2) สถาปัตยกรรมของการสืบค้นข้อมูลบนเว็บ (Search Engine Architecture)
3) เว็บเชิงความหมาย (Semantic Web)
4) การทำเหมืองเอกสาร (Document Mining)
5) โปรแกรมตรรกะเชิงอุปนัย (Inductive Logic Programming)
6) ทฤษฎีกราฟ และสมสัณฐานของกราฟย่อย (Isomorphism Subgraph)
7) ฐานความรู้อภิธานศัพท์เวิร์ดเน็ต (WordNet)

2.1.1 การสืบค้นข้อมูลบนเว็บ

เสิร์ชเอ็นจิน คือเครื่องมือการค้นหาข้อมูลผ่านอินเตอร์เน็ตโดยระบุคำสำคัญ จากนั้นข้อมูลที่ต้องการสืบค้นจะแสดงออกมาเป็นรายการของลิงก์หน้าเว็บที่มีการจัดอันดับ เพื่อให้เลือกข้อมูลที่ตรงตามความต้องการและสนใจมากที่สุด

วิธีการสืบค้นข้อมูลบนเว็บแบ่งออกได้เป็น 3 แบบดังนี้

2.1.1.1 เสิร์ชเอ็นจินแบบไต่ตรวจหา

เสิร์ชเอ็นจินแบบไต่ตรวจหา (Crawler Based Search Engines) เป็นระบบซอฟต์แวร์หุ่นยนต์ (Robot Software) ในบางครั้งเรียกว่าแมงมุม (Spider) หรือ เครื่องมือรวบรวมหน้าเว็บ (Crawler) ทำหน้าที่ในการท่องไปยังเว็บไซต์ต่างๆ บนอินเตอร์เน็ต เพื่อรวบรวมเอกสารบนเว็บเช่น ข้อมูลที่แท็กหัวเรื่อง (Title Tag), แท็กเมตา หรือคำในส่วนแรกๆ ที่ปรากฏบนหน้าเว็บ แล้วนำมาสร้างเป็นฐานข้อมูลดัชนีสำหรับการสืบค้นเอกสารบนเว็บโดยอัตโนมัติ จึงทำให้สืบค้นได้อย่างรวดเร็วและมีบทบาทต่อการสืบค้นมากที่สุดในปัจจุบัน ดังนั้นกระบวนการจัดเตรียมดัชนีจึงเป็นสิ่งสำคัญสำหรับการสืบค้นข้อมูลบนเว็บ แต่การเตรียมข้อมูลดัชนีในการสืบค้นเป็นขั้นตอนการทำงานที่กระทำโดยระบบ ทำให้สามารถได้ข้อมูลที่ทันสมัยอยู่ตลอดเวลา โดยสามารถส่งซอฟต์แวร์หุ่นยนต์ออกไปรวบรวมเอกสารบนเว็บได้ตลอดเวลา ตัวอย่างของเสิร์ชเอ็นจินที่ได้รับความนิยมในปัจจุบันคือ กูเกิล (http://www.google.com), MSN (http://www.msn.com) และ Lycos (http://www.lycos.com)

2.1.1.2 ไดเร็กทอรี

ไดเร็กทอรี (Directory) เป็นการแบ่งแยกหมวดหมู่ของเอกสารโดยมนุษย์ ซึ่งแตกต่างจากเสิร์ชเอ็นจิน ทำได้โดยเจ้าของเว็บไซต์ต้องติดต่อผู้ดูแลไดเร็กทอรีเพื่อให้เพิ่มรายชื่อในไดเร็กทอรี จากนั้นทำการจำแนกหมวดหมู่ของเอกสารของเว็บนั้นให้อยู่ในหมวดหมู่ที่เหมาะสม การพิจารณาจำแนกและระบุหมวดหมู่แบบไดเร็กทอรีนี้เป็นการจัดเรียงข้อมูลบนฐานข้อมูลด้วยมนุษย์ ทำให้การค้นหาเอกสารนั้นได้ผลลัพธ์ของการสืบค้นข้อมูลที่ถูกต้องตามหมวดหมู่ข้อมูลที่มีความเกี่ยวข้องกันในปริมาณมากตรงกับความต้องการมากกว่าเสิร์ชเอ็นจินที่จำแนกด้วยคอมพิวเตอร์ แต่เนื่องจากการสร้างไดเร็กทอรีมีการจัดเอกสารด้วยมนุษย์ทำให้ใช้เวลานาน และมีจำนวนเอกสารในไดเร็กทอรี น้อยกว่าเอกสารที่มีอยู่ในเสิร์ชเอนจิน ตัวอย่างของไดเร็กทอรีในปัจจุบันคือ ยาฮู (http://www.yahoo.com) และ DMOZ (http://www.dmoz.org) หรือ ODP (Open Directory Project) สำหรับเว็บไทยที่รู้จักกันคือ สนุก (http://www.sanook.com) เป็นต้น แต่ด้วยข้อจำกัดของทั้งสองวิธีที่กล่าวมานั้น ทำให้เกิดวิธีการใหม่โดยการรวมวิธีทั้งสองเข้าด้วยกัน เรียกว่า ไฮบริดเสิร์ชเอ็นจิน (Hybrid Search Engine) ซึ่งก็คือ เสิร์ชเอ็นจินที่มีไดเร็กทอรีผสมอยู่ด้วย

2.1.1.3 **เมตาเสิร์ชเอ็นจิน**

เมตาเสิร์ชเอ็นจิน (Meta Search Engine) เป็นการใช้หลักการค้นหาโดยอาศัยจากแท็กเมตาของภาษาเอชทีเอ็มแอล ซึ่งจะถูกประกาศไว้ในรูปแบบข้อความ เช่น ชื่อผู้พัฒนา คำสำคัญหลัก เจ้าของเว็บไซต์ หรือรายละเอียดอื่น ผลการค้นหาวิธีนี้จะไม่แม่นยำ เนื่องจากผู้ผลิตเว็บไซต์อาจจะระบุข้อความไว้มากมาย เพื่อให้สามารถค้นหาและพบเว็บ จึงทำให้ผลการค้นหาข้อมูลดังกล่าวไม่ถูกต้อง เช่น DogPile (http://www.dogpile.com) และ Metacrawler (http://www.metacrawler.com)

2.1.2 **สถาปัตยกรรมของการสืบค้นข้อมูลบนเว็บ**

การสืบค้นข้อมูลบนเว็บ มีการทำงานหลัก 4 ส่วนดังนี้คือ

2.1.2.1 **เครื่องมือรวบรวมหน้าเว็บ**

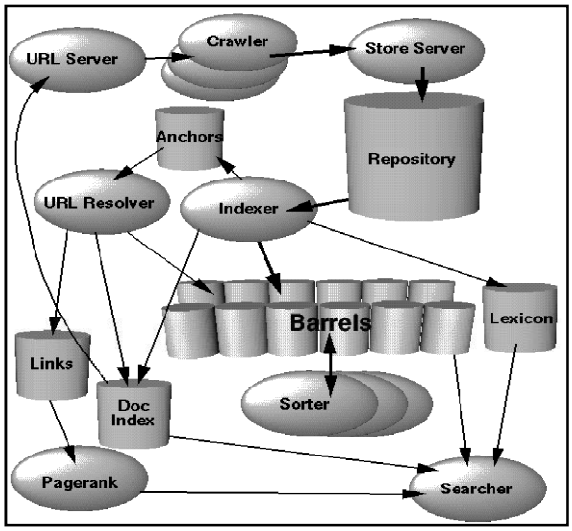
เครื่องมือรวบรวมหน้าเว็บ หรือแมงมุมเป็นโปรแกรมหุ่นยนต์ (Robot) ทำหน้าที่ไปดึงหน้าเว็บต่างๆ มา โดยปกติแล้วข้อมูลที่แสดงในหน้าเว็บไซต์นั้นที่จริงแล้วเป็นเพียงข้อมูลตัวอักษร (Text File) เท่านั้น เมื่อหุ่นยนต์นี้ได้ข้อมูลมาแล้ว จะทำการแยกข้อมูลและลิงก์ (Link) ที่ไปยังหน้าอื่นออกมา แล้วเรียงคิวลิงก์เพื่อจะไล่ไปทีละลิงก์ตามลำดับ และไปเก็บข้อมูลของหน้าเว็บนั้นมาทำอย่างนี้เรื่อยไป เสมือนว่าหุ่นยนต์นี้จะคืบคลานออกจากจุดเริ่มต้นไปเรื่อยๆ เริ่มแรกหุ่นยนต์นี้จะทำการเก็บข้อมูลเฉพาะชื่อเว็บ และข้อมูลภายในส่วนหัวของหน้าเว็บ ที่อยู่ภายในแท็กเมตาเท่านั้น แต่ภายหลังหุ่นยนต์นี้ได้ทำการเก็บรวบรวมทั้งลิงก์ ข้อความ และทุกอย่างที่ปรากฏบนหน้าเว็บด้วย

2.1.2.2 **ตัวทำดัชนี**

ตัวทำดัชนี (Indexer) เป็นส่วนที่ทำหน้าที่ตัดสินใจในการจัดดัชนีของเอกสารต่างๆ ว่าเอกสารนี้ควรรวบรวมอยู่ในดัชนีเมตาดาตา คำสำคัญในการนำมาจัดดัชนีคือคำหลักในส่วนหัวของเอกสาร จากแท็กเมตา แท็กหัวเรื่อง และแท็กอักษรพิเศษ ตัวหนา ตัวเอน หรือตัวขยาย วิธีการ

สร้างดัชนีสำหรับการค้นหา เช่น ปริภูมิเวกเตอร์ (Vector space), เอ็น-แกรม (N-Gram) และ ดัชนีย้อนกลับ (Inverted Index)

ภาพที่ 2.1
แสดงสถาปัตยกรรมของการสืบค้นข้อมูล (บรินและเพจ, 1998)



2.1.2.3 ตัวสืบค้น

ตัวสืบค้น (Searcher) เป็นส่วนของตัวสืบค้นข้อมูล (Query) จากคำสำคัญของผู้ใช้งานมาค้นหาเอกสารที่เกี่ยวข้องในฐานข้อมูลดัชนี

2.1.2.4 การจัดอันดับ

การจัดอันดับ (Page Rank) เป็นค่าตัวเลขที่แสดงค่าความสำคัญของหน้าเว็บ โดยพิจารณาจากหน้าเว็บที่มีหน้าเว็บอื่นอ้างอิงมาถึงมากน้อยเพียงใด ทฤษฎีนี้มีการนำแนวความคิดจากการอ้างอิงเกี่ยวกับผลงานทางวิชาการ ในการอ้างอิงถึงทฤษฎีของนักวิชาการท่านอื่น ดังนั้นผลงานทางวิชาการใดที่ได้รับการอ้างถึง (Citation) บ่อย จากนักวิชาการท่านอื่นๆ แสดงให้เห็นว่าผลงานทางวิชาการนั้นได้รับการยอมรับอย่างแน่นอน ผลงานที่มีการถูกอ้างอิงมากน้อยนั้นมีดัชนีอ้างอิงเรียกว่า ดัชนีอ้างอิง (Citation Index)

2.1.3 **เว็บเชิงความหมาย** (Semantic Web)

เว็บเชิงความหมาย เป็นเว็บที่มีการอธิบายความหมายของข้อมูล เพื่อให้ข้อมูลนั้นสามารถถูกอ่านโดยซอฟต์แวร์ และเครื่องคอมพิวเตอร์แล้วเข้าใจ (Machine Readable & Understandable) โดยกำหนดโครงสร้างในการอธิบายข้อมูลให้มีมาตรฐานเดียวกัน ซึ่งการพัฒนาเว็บเชิงความหมายนั้น จะเน้นการพัฒนาโดยให้คอมพิวเตอร์ติดต่อสื่อสารข้อมูลระหว่างคอมพิวเตอร์ด้วยกัน และสามารถเข้าใจข้อมูลได้อย่างถูกต้อง โดยแนวคิดในการพัฒนาการติดต่อสื่อสารกันระหว่างคอมพิวเตอร์มีนำมาใช้บ้างแล้ว ตัวอย่างเช่น เทคโนโลยีเอ็กซ์เอ็มแอล หรือเว็บเซอร์วิส (Web Services) เทคโนโลยีเหล่านี้ยังไม่แก้ปัญหาข้างต้นได้ดีพอ เนื่องจากเทคโนโลยีเหล่านี้ยังไม่ได้กล่าวถึงความหมาย หรือความสัมพันธ์ของข้อมูล เพียงแค่สามารถแลกเปลี่ยนข้อมูลระหว่างคอมพิวเตอร์ได้เท่านั้น แต่เทคโนโลยีของเว็บเชิงความหมาย จะถูกนำมาช่วยให้การติดต่อสื่อสารระหว่างเครื่องคอมพิวเตอร์ก้าวหน้าไปกว่าเดิม เพราะเว็บเชิงความหมายมีรูปแบบในการอธิบายข้อมูลด้วยเมตาดาตา ที่ทำให้คอมพิวเตอร์สามารถวิเคราะห์โครงสร้างของเมตาดาตาแล้วสืบค้นข้อมูลที่ผู้ใช้ต้องการค้นหาได้อย่างตรงตามความต้องการ (Relevant) เทคโนโลยีที่จะนำมาใช้ในเว็บเชิงความหมายนั้น จะกล่าวถึงต่อไปดังนี้

2.1.3.1 **เอ็กซ์เอ็มแอล**

เอ็กซ์เอ็มแอล (XML หรือ eXtensive Markup Language) เป็นภาษาที่ใช้สำหรับการเขียนเอกสารพิเศษ (Markup Document) โดยมีเมตาดาตา หรือ แท็ก (Tag) สำหรับกำหนดรูปแบบและประเภทของข้อมูลในส่วนต่างๆ ของเอกสารนั้นได้ชัดเจน ในการเพิ่มเมตาดาตาในเอกสารนั้นจะทำให้โครงสร้างของเอกสารชัดเจนยิ่งขึ้นและสามารถประมวลผลเอกสารได้ง่ายและไม่จำเป็นต้องอาศัยมนุษย์ในการตีความเอกสารนั้นอีก

ด้วยเทคโนโลยีเอ็กซ์เอ็มแอลถูกกำหนดให้เป็นภาษามาตรฐานจากองค์กร W3C สำหรับการกระจายข้อมูลและแลกเปลี่ยนข้อมูลผ่านระบบคอมพิวเตอร์ ทั้งยังเป็นภาษาที่มีความยืดหยุ่น จึงทำให้ผู้ใช้สามารถกำหนดและตั้งค่าเมตาดาตาเฉพาะกับเอกสารเพื่อสื่อความหมายให้เหมาะกับความต้องการใช้งานได้ ตัวอย่างเช่น การเขียนโน้ตจากเจนถึงโทนี เป็นดังนี้

ภาพที่ 2.2

แสดงข้อความในรูปแบบเอกสารเอ็กซ์เอ็มแอล

```
<note>
<to>Tony</to>
<from>Jane</from>
<subject>Reminder</subject>
<body>Meeting On Wednesday</body>
</note>
```

การสร้างประโยค (Syntax) โครงสร้างของเอกสารเอ็กซ์เอ็มแอล แบ่งออกเป็น 2 ส่วน คือ

1. โปรล็อก (Prolog) เป็นโครงสร้างส่วนแรก ประกอบด้วย

1) การประกาศเอกสารเอ็กซ์เอ็มแอล (XML Declaration) เป็นการระบุเวอร์ชัน (Version) ของเอ็กซ์เอ็มแอลจึงควรจะประกาศไว้เสมอ และต้องมีส่วนประกาศรหัสภาษา (Encoding Declaration) เพื่อระบุวิธีการเข้ารหัสตัวอักษรเช่น UTF-8 เป็นการระบุแบบแผนการเข้ารหัสตัวอักษร (Default Character Encoding Scheme) กล่าวคือเป็นตัวแทนของตัวอักษรส่วนใหญ่ในภาษาอังกฤษรวมถึงภาษาไทยด้วย

ภาพที่ 2.3

แสดง XML Declaration

```
<?xml version="1.0" encoding="UTF-8"?>
```

2) รายละเอียดของเนื้อหาเอกสาร (Document Type Declaration หรือ DTD) ประกอบด้วยรหัสพิเศษ (Markup Code) เพื่อกำหนดกฎการเขียนรายละเอียดของเนื้อหาเอกสาร

2. องค์ประกอบของเอกสาร (Document Element) เป็นส่วนแสดงข้อความภายใน (Content) องค์ประกอบเอกสารนั้นเป็น องค์ประกอบเดี่ยว (Single Element) ที่สามารถประกอบด้วยองค์ประกอบย่อย (Sub elements) และ แหล่งข้อมูลภายนอก (External Entities) ที่

ไม่จำกัด กล่าวอีกนัยหนึ่งองค์ประกอบของเอกสารคือ องค์ประกอบหลัก (Root Element) ของเอกสารนั้น

ภาพที่ 2.4

แสดงเอกสาร XML

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE note SYSTEM "note.dtd">
<note>
<to>Tony</to>
<from>Jane</from>
<subject>Reminder</subject>
<body>Meeting On Wednesday</body>
</note>
```

2.1.3.2 Uniform Resource Identifier

Uniform Resource Identifier หรือ URI เป็นวิธีการหนึ่งในการอ้างถึงหรือการระบุตัวตนให้กับตำแหน่งของเนื้อหาอื่น ซึ่งอาจจะเป็นหน้าเว็บข้อความ วีดีโอ เสียง ภาพ ภาพเคลื่อนไหว หรือโปรแกรม เป็นต้น รูปแบบร่วมกันของ URI สำหรับระบุตำแหน่งของหน้าเว็บซึ่งเป็นรูปแบบเฉพาะหรือกลุ่มย่อย ที่รู้จักกันคือ URL (Uniform Resource Locator) ซึ่ง URL ช่วยให้สามารถเข้าถึง (อ้างถึง) เว็บไซต์ได้ เช่น http://www.w3.org/xml เป็นต้น เห็นได้ว่าสามารถค้นหาแหล่งที่มาของข้อมูล (Resource) ที่ชี้เฉพาะได้ และสามารถระบุตัวตน (Identifiers) รวมถึงระบุตำแหน่ง (Locate) ของเว็บไซต์นั้นๆ แต่สำหรับ URI สามารถระบุตัวตนได้แต่จะไม่สามารถระบุตำแหน่งได้

ตัวอย่างของ URI เช่น http://www.w3.org/Icons/www/w3c.main.gif เป็นการระบุไฟล์ที่สามารถเข้าถึงโดยเว็บโปรโตคอล (http://)

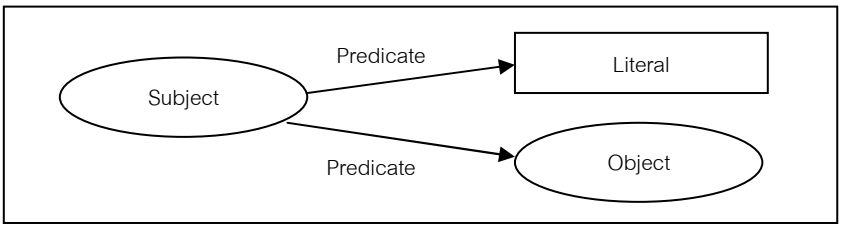
2.1.3.3 Resource Description Framework

Resource Description Framework หรือ RDF เป็นภาษามาตรฐานที่ถูกพัฒนาขึ้นมาโดย W3C โดยมีจุดมุ่งหมายเพื่อนำมาใช้สำหรับอธิบายโครงสร้างในการกำหนดและแลกเปลี่ยนข้อมูลเมตาดาตา ซึ่งประกอบด้วย 3 ส่วน (Triple) ดังนี้คือ

1) Subject คือแหล่งข้อมูลที่เกี่ยวข้อง โดยระบุเป็น URI ใช้สัญลักษณ์แทนด้วยวงรี
2) Predicate คือแหล่งข้อมูลที่มีชื่อเฉพาะและมีคุณสมบัติ เช่น ผู้แต่ง หรือ หัวเรื่อง ใช้สัญลักษณ์แทนด้วยลูกศรที่ชี้ไปยังส่วนของ Object
3) Object คือค่าของคุณสมบัติที่แทนด้วย URI ใช้สัญลักษณ์แทนด้วยวงรี กรณีที่เป็นค่าของคุณสมบัตินั้น จะเรียกว่า สัญพจน์ (Literal) ใช้สัญลักษณ์แทนด้วยสี่เหลี่ยม

ภาพที่ 2.5

แสดงโครงสร้างของ Triple



การแทนข้อมูลให้อยู่ในรูปแบบ RDF นั้นจะแทนข้อมูลด้วยตัวอย่างดังนี้คือ กำหนดให้มีข้อมูล “Tony is the author of the resource http://www.myweb.com/tony.html“ ดังนั้นจะสามารถแทนข้อมูลตามโครงสร้างการอธิบายข้อมูล RDF ดังนี้

Subject แทน http://www.myweb.com/tony.html

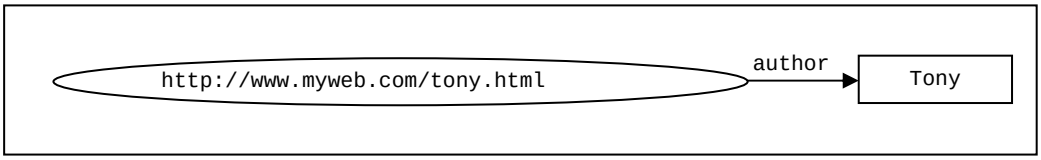
Predicate แทน author

Object แทน Tony

นำข้อมูลข้างต้นมาแสดงเป็นโครงสร้างกราฟรูปแบบ RDF ได้แผนภาพดังนี้

ภาพที่ 2.6

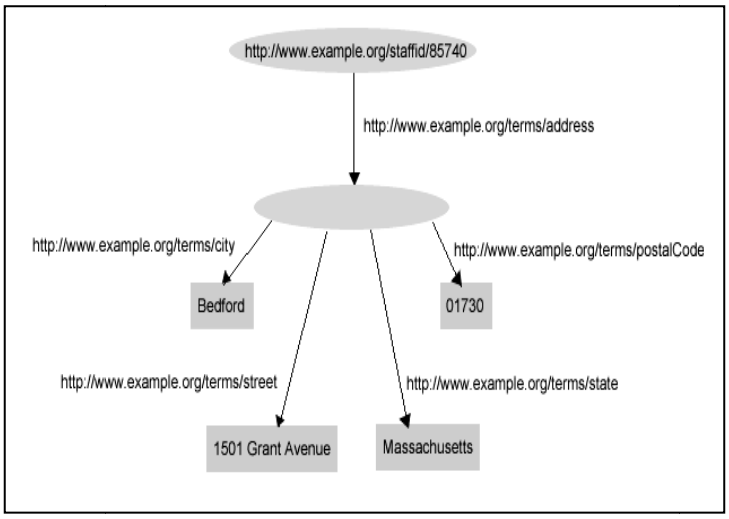
แสดงการแทนข้อมูลในโครงสร้าง RDF (Triple)



การแสดงเอกสาร RDF โดยการแทนด้วย Triple นั้น แต่สำหรับในทางการปฏิบัติจริงแล้วข้อมูลที่นำมาใช้งานจริงจะมีความซับซ้อนและมีจำนวนมากกว่าหนึ่ง Triple ซึ่งจะสามารถเขียนให้อยู่ในรูป N-Triple ดังแผนภาพดังนี้

ภาพที่ 2.7

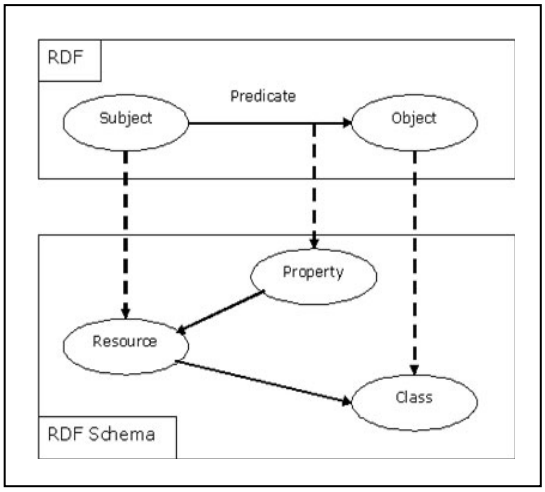
แสดงการแทนข้อมูลในโครงสร้าง RDF (N-Triple)



สำหรับเอกสาร RDF นั้นยังมีเทคนิคในการอธิบายโครงสร้างของเมตาดาตาที่เรียกว่า RDF Schema (RDFS) ซึ่งจะใช้อธิบายแหล่งข้อมูล โดยการกำหนดคำศัพท์ (Vocabulary) ซึ่งประกอบด้วย คุณสมบัติ และค่าของคุณสมบัติ โดยเพิ่มการอธิบายให้อยู่ในรูปของโหนดที่มีคุณสมบัติ และคลาส เพื่อใช้ในการอธิบายโครงสร้างของเมตาดาตา โดยสามารถอธิบายได้ดังแผนภาพที่ 2.8 ซึ่งจากแผนภาพนั้นแสดงการนำโครงสร้างเมตาดาตามาอธิบายข้อมูลประกอบด้วยคุณสมบัติและค่าของคุณสมบัติมากำหนดแทนด้วยโหนดสำหรับอธิบายแหล่งข้อมูลสำหรับภาคแสดงเป็นคุณสมบัติมากำหนดแทนด้วยโหนดคุณสมบัติ และส่วนของวัตถุเป็นค่าคุณสมบัติที่จะถูกกำหนดแทนด้วยโหนดคลาส เนื่องด้วยทั้งโหนดคุณสมบัติ และโหนดคลาสจะสามารถสืบทอดเป็นคุณสมบัติย่อย (Sub Property) และคลาสย่อย (Subclass) ตามลำดับ

RDF Schema มีมาตรฐานคำที่สามารถนำมาใช้ร่วมกันที่จะมารองรับการช่วยกำหนดในการเพิ่มการอธิบายความหมายของข้อมูลเพื่อให้เข้าใจมากขึ้น

ภาพที่ 2.8
แสดงการอธิบายโครงสร้างของเมตาดาตา



W3C ได้มีการกำหนดคำเฉพาะไว้สำหรับ RDF Classes มีดังนี้ rdfs:Resource rdfs:Literal rdf:XMLLiteral rdfs:Class rdf:Property rdfs:Datatype rdf:Statement rdf:Bag rdf:Seq rdf:Alt rdfs:Container rdfs:ContainerMembershipProperty rdf:List เป็นต้น

และส่วน RDF Properties เช่น rdf:type rdfs:subClassOf rdfs:subPropertyOf rdfs:domain rdfs:range rdfs:label rdfs:comment rdfs:member rdf:first rdf:rest rdfs:seeAlso rdfs:isDefinedBy rdf:value rdf:subject rdf:predicate rdf:object

2.1.3.4 Web Ontology Language

Web Ontology Language หรือ OWL เป็นภาษามาตรฐานที่ใช้สำหรับเก็บเอกสารที่มีข้อมูล (Information) ที่ต้องการให้คอมพิวเตอร์ หรือโปรแกรมอื่นๆ สามารถนำไปประมวลผลข้อมูลนี้ได้ (Machine Processable) โดย OWL สามารถนำเสนอความหมาย และความสัมพันธ์ของคำจากคำศัพท์ได้อย่างชัดเจน ทำให้การนำเสนอข้อมูลแบบนี้เรียกว่า การนิยามคำศัพท์แนวคิด OWL ถูกคิดค้นขึ้นมาเพื่อสนับสนุนให้มีการติดต่อสื่อสาร แลกเปลี่ยนข้อมูลซึ่งกันและกันได้บนเว็บไซต์ และมีประสิทธิภาพมากกว่าภาษาที่กล่าวไว้ก่อนหน้านี้ เช่น XML, RDF และ RDFS มีการเตรียมเทคนิคสำหรับการเพิ่มคำศัพท์ที่มีรูปแบบความหมายที่เป็นรูปนัย (Formal Semantic)

ซึ่ง OWL มีระดับแบบแผนในการอธิบายความหมายสามระดับคือ OWL Lite, OWL DL และ OWL Full

ทั้งยังมีคุณลักษณะมาตรฐานเพื่อช่วยในการเพิ่มการอธิบายความหมายของข้อมูลเพื่อให้เข้าใจตามรูปนัย (Formal) มากยิ่งขึ้นโดยจะเพิ่มจากคุณสมบัติเดิมของ RDFS โดยสามารถนำมาใช้ได้ สำหรับส่วนที่เพิ่มขึ้นมาใน OWL มีดังนี้ ส่วน OWL Lite เช่น owl:ObjectProperty owl:DatatypeProperty owl:inverseOf owl:equivalentClass owl:equivalentProperty owl:sameAs owl:differentFrom owl:AllDifferent owl:distinctMembers เป็นต้น สำหรับส่วนที่เพิ่มใน OWL DL และ OWL Full เช่น owl:minCardinaltily owl:maxCardinality owl:cardinality เป็นต้น

ภาพที่ 2.9

แสดง OWL Document ของ Music Ontology

```
<mlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
  <owl:Ontology rdf:about="">
    <rdfs:label>Music Ontology</rdfs:label>
  </owl:Ontology>
  <owl:Class rdf:ID="Group"/>
  <owl:Class rdf:ID="Person"/>
  <owl:Class rdf:ID="Instrument"/>
  <Instrument rdf:ID="Guitar"/>
  <Instrument rdf:ID="Bass"/>
  <Instrument rdf:ID="Drums"/>
  <owl:Class rdf:ID="Band">
    <rdfs:subClassOf rdf:resource="#Group"/>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#member"/>
        <owl:allValuesFrom rdf:resource="#Musician"/>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Musician">
    <rdfs:subClassOf rdf:resource="#Person"/>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#plays"/>
        <owl:minCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:minCardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
```

```
  <owl:Class rdf:ID="Guitar">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Guitar</rdfs:comment>
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Strings"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Delta_Blues2">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Regional_Blues"/>
    </rdfs:subClassOf>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Delta Blues</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="Singer_Songwriters6">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Live_Albums10"/>
    </rdfs:subClassOf>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Singer-Songwriters</rdfs:comment>
  </owl:Class>
```

จากภาพที่ 2.9 เป็นภาพแสดง OWL Document และ ข้อมูลของเอกสาร (Instance) ที่มีการแสดงข้อมูลโดยนิยามคำศัพท์แนวคิดเกี่ยวกับเพลง จากงานวิจัย (ภูวดล เรืองกิจภิญโญ, 2549) เสนอว่าเอกสารใดๆ แม้ว่าจะต่างนิยามคำศัพท์ตามแนวความคิดกัน แล้วหากมีบางส่วนของข้อมูลที่มีนิยามคำศัพท์ตามแนวความคิดเกี่ยวเนื่องกัน จะต้องมีโครงสร้างข้อมูลคล้ายคลึงกัน โดยเปรียบเทียบเฉพาะตัวอย่าง OWL โดยแสดงให้อยู่ในรูปแบบกราฟ แล้วใช้กราฟย่อยมาตรวจสอบความสมสัณฐานของกราฟย่อยเพื่อหาความเหมือนกันของเอกสาร OWL

2.1.4 **การทำเหมืองข้อมูลเอกสาร** (Document Mining)

เอกสาร (Documents) เป็นไฟล์เอกสารข้อความ (Text) ที่คอมพิวเตอร์สามารถอ่านได้ และบางครั้งอยู่ในรูปแบบอื่นเช่น ตาราง (Tables) ภาพ (Images) กราฟิกส์ (Graphics) เสียง (Audios) และภาพเคลื่อนไหว (Videos) ซึ่งเอกสารที่ถูกอ้างถึงบนเว็บนั้น จะเรียกว่าหน้าเว็บ (Web page) หรือ เว็บไซต์ (Web site) สำหรับการทำเหมืองข้อมูลเอกสาร (Data Mining) นั้นเป็นการค้นหาและข้อมูลที่มีประโยชน์จากเอกสารโดยอัตโนมัติ โดยการทำเหมืองข้อมูลเอกสารนั้นจะแก้ปัญหาโดยการรวมข้อมูลที่มีอยู่อย่างมากเกินไป ตัวอย่างวิธีการคือ การจัดกลุ่มเอกสาร (Document Clustering) และการจำแนกเอกสาร (Document Classification) จุดประสงค์คือจัดหมวดหมู่ของเอกสารที่เหมือนกันของความหมาย หรือมีความหมายเชื่อมโยงกัน (Topology) ซึ่งการทำเหมืองข้อมูลเอกสารโดยใช้หลักการวิเคราะห์ความสำคัญของการกำหนดความหมายของเอกสาร หรือเซตของเอกสาร ด้วยการสร้างความเข้าใจในเนื้อหา การให้ความหมาย และเป็นไปตามความต้องการของผู้ใช้บางคน โดยแตกต่างจากการทำเหมืองข้อมูลปกติ ด้วยเหตุผลที่รูปแบบ (Pattern) ส่วนใหญ่ถูกคัดสรรมาจากข้อความภาษาตามธรรมชาติ (Natural Language Text) มากกว่าจากโครงสร้างฐานข้อมูลแท้จริง ซึ่งฐานข้อมูลจะถูกออกแบบสำหรับการทำงานโดยอัตโนมัติ แต่ข้อความนั้นถูกเขียนด้วยคนเพื่อให้คนสามารถอ่านได้ ส่วนของหลักการทำเหมืองข้อมูลเอกสารนั้น ประกอบด้วย

2.1.4.1 **แบบจำลองการแปลงข้อความ** (Text Representation Model)

แบบจำลองการแปลงข้อความเป็นการแปลงข้อความให้แสดงในรูปแบบโมเดลสำหรับเอกสารหนึ่งที่มีคำสำคัญเฉพาะ หรือความถี่ของการปรากฏของคำเหล่านั้น ตัวอย่างเช่น ใน

แบบจำลองปริภูมิเวกเตอร์ (Vector Space Model) เอกสารจะถูกแสดงโดยเวกเตอร์ (Vector) ที่แสดงความถี่ของคำที่เป็นไปได้ทั้งหมด (Features) ในเซตของเอกสาร ทั้งคำที่พบไม่บ่อยในแต่ละส่วนของเอกสารโอกาสความเป็นไปได้ของคำ มีค่าตั้งแต่ 0 หรือ ความถี่ต่ำ (Low Frequencies) ดังนั้นคำสำคัญที่ถูกเลือกนำมาแสดงให้ถูกต้องกับเอกสารความสำคัญที่เขียนโดยกฎ เช่น ความถี่ของเอกสารและส่วนกลับความถี่ของเอกสาร (Document Frequency Inverse Document Frequency), อัตราการเพิ่มขึ้นของข้อมูล (Information Gain), สารสนเทศร่วม (Mutual Information), ไคสแควร์ (x^2- test) และ ความเข้มของพจน์ (Term Strength) มากไปกว่านั้นก่อนที่จะทำการเลือกคำสำคัญจะต้องทำการกำจัดคำหยุด (Stop Words) และใช้อัลกอริธึมตัดคำ ตัวอย่างคำหยุดเช่น the, and, a และ for เป็นต้น อัลกอริธึมตัดคำจะแปลงความแตกต่างรูปแบบคำให้อยู่ในรูปแบบปกติที่เหมือนกัน โดยมีสองอัลกอริธึมในการตัดคำที่ถูกนำมาใช้กัน คือ พอร์เตอร์สเต็มเมอร์ (Porter Stemmer) และการค้นหาคำศัพท์จากพจนานุกรม เช่น เวิร์ดเน็ต

แม้บ่อยครั้งการใช้คำศัพท์พื้นฐานที่แสดงในเอกสารนั้น อาจทำให้ขาดความน่าเชื่อถือสำหรับการทำเหมืองข้อมูลเอกสาร ซึ่งพิจารณาเทียบเคียงแล้วเอกสารก็คือ ที่บรรจุของคำ (a bag of words) โดยละเว้นความหมายและความคิดที่ต้องการแสดงออกมาของผู้จัดทำเอกสารนั้น ซึ่งข้อบกพร่องที่เป็นเหตุให้การวัดความเหมือนกันนั้นผิดพลาด นั่นคือแวดล้อมของคำในประโยคโดยยอมรับว่าคำที่เหมือนกันในบทความที่หลากหลายจากประโยคที่ต่างกัน เสมือนว่ามีความหมายเหมือนกันโดยยอมเข้าใจคำอธิบายที่เหมือนกันของข้อความในบทความซึ่งเกิดขึ้นด้วยความหลากหลายของคำที่อยู่ในบทความนั้น หรือเป็นการยอมรับข้อความแวดล้อมที่ต่างกันของข้อความในบทความว่าเป็นความเหมือนกัน เพราะมีความเหมือนกันของคำในบทความ เช่น “Tony eats banana standing beside the tree” และ “The banana tree stands beside Tony’s house” แม้จะมีการใช้คำที่เหมือนกัน แต่ความหมายของทั้งสองประโยคนั้นแตกต่างกัน และสองประโยคที่มีความหมายเหมือนกันแต่ถูกสร้างด้วยเซตของคำที่แตกต่างกัน เช่น “Tony is an intelligent boy” และ “Tony is a brilliant lad”

ด้วยความพยายามในการพัฒนาการแสดงข้อความ นอกจากที่เป็นไปได้โดยการใช้เพียงคำสำคัญ ประกอบกับการใช้เอ็นแกรม สำหรับคัดเลือกคำที่มีสองตัวอักษร และกระจายความสัมพันธ์ของคำที่มีความหมายผ่านสถิติของข้อมูลคำ แล้วใช้ความรู้ภูมิหลังโดยการแทนที่คำด้วยแนวความคิดที่มากกว่าในการนิยามคำศัพท์แนวคิด

2.1.4.2 เครื่องมือวัดความเหมือนกัน

เครื่องมือชี้วัดความเหมือน (Similarity Measures) ถูกนำมาใช้ในการประเมินความคล้ายกันระหว่างเอกสาร หลังจากที่เปลี่ยนรูปจากข้อมูลที่อยู่ในรูปแบบข้อความไปเป็นรูปแบบที่สามารถใช้และเข้าใจได้ ด้วยเทคนิคที่หลากหลายในการชี้วัดความเหมือนกันนั้นขึ้นอยู่กับการเลือกโมเดลที่จะมาแสดงเป็นข้อความ ในแบบจำลองปริภูมิเวกเตอร์สำหรับตัวอย่าง ปริภูมิลักษณะเด่น (Feature Space) นั้นเป็นองค์ประกอบของปริภูมิเรขาคณิต (Geometric Space) โดยที่นำเอกสารมาแสดงเป็นจุดในปริภูมิหลายมิติ (Multidimensional Space) ดังนั้นการชี้วัดความเหมือนสามารถคำนวณด้วยวิธีง่ายๆ ตัวชี้วัดที่ถูกนำมาใช้บ่อยคือ การวัดโคไซน์ (Cosine Measure) และ การวัดแจ็คการ์ด (Jaccard Measure)

การวัดโคไซน์ นิยามโดย $\cos(x, y) = \frac{x.y}{|x||y|}$

โดย (x, y) แทนจุดเชื่อมต่อของ x กับ y

และ $|x|$และ$|y|$ แทนความยาวของเวกเตอร์ x,y ตามลำดับ

การวัดโคไซน์จะให้ค่าความเหมือนสูง กับเอกสารที่มีเซตความเหมือนของคำด้วยความถี่ที่มีเกณฑ์สูง

การวัดแจ็คการ์ด นิยามโดย $\mathrm{sim}(x, y) = \frac{|x \cap y|}{|x \cup y|}$

สำหรับ การซ้อนทับกันระหว่างสองเอกสารนั้นสามารถคำนวณจากจำนวนของคำที่เกิดขึ้นระหว่างสองเอกสาร ซึ่ง การวัดแจ็คการ์ดสามารถทำงานได้ทั้งเวกเตอร์ลักษณะเด่นแบบต่อเนื่องและแบบไบนารี (Binary)

ส่วนฟังก์ชันใกล้เคียงของคลาสอื่นที่รู้จักคือ ระยะทางมินโคว์สกี (Minkowski distance)

นิยามโดย $\left|\left|x - y\right|\right|_p = \sqrt[p]{\sum_{i=1}^{n} |x_i - y_i|^p}$ โดย $x, y \in \Re^2$

ฟังก์ชั่นระยะห่างอธิบายโดยจำนวนอนันต์ของระยะทางที่ชี้โดย p สัณนิษฐานว่ามีค่ามากกว่าหรือเท่ากับ 1

สำหรับค่าที่เกิดขึ้นบางค่าของ p และฟังชันระยะห่างโดยลำดับ คือ

$p = 1$: ระยะทางแมนฮัตตัน (Manhattan Distance) $\left|\left|x - y\right|\right|_1 = \sum_{i=1}^{n} |x_i - y_i|$

$p = 2$: ระยะทางยูคลิด (Euclidean Distance) $\left|\left|x - y\right|\right|_2 = \sqrt{\sum_{i=1}^{n} |x_i - y_i|^2}$

$p = \infty$: ระยะทางเชบิเชฟ (Tschebyshev Distance) $\left|\left|x - y\right|\right|_\infty = \max_{i=1,2,\ldots n} |x_i - y_i|$

ตัวชี้วัดความเหมือนหนึ่งที่ถูกพัฒนาด้วยพื้นฐานการเรียงลำดับคำเพื่อชี้วัดความเหมือนระหว่างเอกสาร โดยพิจารณาจำนวนของลำดับที่ถูกแบ่งระหว่างสองเอกสารตามด้วยความยาว ความถี่ และระดับความสำคัญของแต่ละเอกสาร แม้ว่าเทคนิคถูกสร้างขึ้นมาเพื่อชี้วัดระยะห่างระหว่างเวกเตอร์ของลักษณะเด่นเชิงตัวเลข แต่เนื้อหาที่สืบทอดมาจากคำที่สนใจนั้นยังคงไม่เพียงพอ เนื่องจากไม่มีความสัมพันธ์ร่วมกันระหว่างการเปรียบเทียบคำและการเปรียบเทียบความหมาย เช่นในประโยคตัวอย่างที่แสดงไว้ในหัวข้อด้านบนที่มีค่าความสัมพันธ์ร่วมกันนั้นต่ำมาก นั้นดีกว่าผลลัพธ์ของการทำเหมืองที่ได้มาจาการนับคำที่พบในเอกสาร แต่ทำให้พบว่ามีความหมายของคำเพิ่มขึ้น จึงต้องกำหนดให้หนึ่งประโยคมีความแตกต่างจากประโยคแวดล้อมอื่น

2.1.4.3 ขั้นตอนการทำเหมืองข้อมูล

กระบวนการทำเหมืองข้อมูล (Mining Process) เช่นการแบ่งกลุ่มเอกสาร การจัดหมวดหมู่เอกสาร การค้นคืนข้อมูล (Information Retrieval) การกรองข้อมูล (Information Filtering) และการกระจายข้อมูล (Information Extraction) โดยเฉพาะอาจจะมีความหลากหลายจากความต้องการ ซึ่งเป้าหมายคือการค้นหา (Discovery) และกระจายความรู้จากเอกสาร โดยมีการทำงาน เช่น จัดหมวดหมู่เอกสาร ค้นหาความสัมพันธ์ หรือความเกี่ยวข้องกันระหว่างเอกสาร การค้นหาเอกสารที่สัมพันธ์กันในการสืบค้น ค้นหาเส้นทางของเอกสารที่ผู้ใช้สนใจ และการรวมข้อความไว้กับโครงสร้างข้อมูลอื่น การค้นคืนข้อมูลเกี่ยวข้องกับการค้นหาเอกสารที่สัมพันธ์กันในการแสดงกลับไปให้ผู้ใช้ที่ทำการร้องขอ และทำการจัดเรียงอันดับเอกสารด้วย ปกติแล้วการทำโดยการชี้วัดระยะห่างระหว่างเอกสารกับคำสืบค้นที่อยู่ในรูปแบบของดัชนี (Index) เมื่อการชี้วัดของความสัมพันธ์และความเหมือนกันโดยการแปลงผลลัพธ์ (Transform) เอกสารกลับไปยังผู้ใช้หรือเซตของผู้ใช้ที่มีความสนใจ ส่วนการกรองข้อมูลนั้นนำมาใช้ในการยอมรับหรือตอบปฏิเสธเอกสารที่รับเข้ามา เช่นในการกรองอีเมล์ (Email) ซึ่งจะพยายามกรองเมล์ขยะ (Junk Mail) ก่อน ส่วนเป้าหมายของการกระจายข้อมูล คือการหาที่อยู่เฉพาะของข้อมูลและสร้างรูปแบบโครงสร้างจากเอกสารที่ไม่มีโครงสร้าง หรือกึ่งโครงสร้าง ผลลัพธ์ของระบบการกระจายข้อมูลคือการจัดเรียงตามปกติ หรือลักษณะที่มีการปรับแต่งรูปแบบ และมีการเพิ่มเติมด้วยข้อมูลที่ไม่มีความกำกวม (Unambiguous Data) ทำโดยการวิเคราะห์ข้อมูลที่สัมพันธ์กันแต่ละส่วนของเอกสาร โดยการกำหนดโดเมนนำร่อง ซึ่งชี้เฉพาะชนิดของข้อมูลที่ต้องการค้นหา หลักการจัดหมวดหมู่เอกสารคือ

การกำหนดให้เอกสารใหม่นั้นอยู่ในหมวดหมู่ (Class) ที่เหมาะสมซึ่งสามารถทำได้ด้วยมือ เพื่อมาตรฐานการเรียนรู้แบบมีผู้สอน (Supervised Machine Learning) ว่ากำหนดเอกสารให้อยู่หมวดหมู่ที่ถูกต้องหรือไม่นั้น จึงต้องจัดหมวดหมู่ที่จะเรียนรู้กับเซตของเอกสารว่า ถ้าหมวดหมู่หรือกลุ่มของเอกสารนั้นมีอยู่แล้ว จะค้นหากลุ่มของเอกสารจากเซตที่มีความเหมือนของเอกสารภายในกลุ่มมาก และจะต้องมีความเหมือนกันระหว่างกลุ่มน้อย

กระบวนการเปรียบเทียบความเหมือนกันนั้น จะใช้โมเดลนำมาแสดงข้อความ และตัวชี้วัดความเหมือนกันที่ทำกับเฉพาะงานนั้นๆ สำหรับข้อความที่เป็นประโยชน์ ตามด้วยตัวชี้วัดความแม่นยำของความเหมือนกัน จะเพิ่มความแน่นอนกับผลลัพธ์ การจัดกลุ่มเอกสารถูกนำมาใช้ในกรณีศึกษาที่จำลองการทำงานและประสิทธิผลของการเข้าถึงเอกสารเหมืองข้อมูลที่ถูกนำเสนอการตรวจดูรายละเอียดที่เพิ่มขึ้น เกี่ยวกับการให้ส่วนย่อย (Subsection) ที่ติดตามมาด้วย

2.1.4.4 **การจัดกลุ่มเอกสาร**

หลักการของการจัดกลุ่มเอกสารโดยการแบ่งเอกสารเข้าไปไว้ในกลุ่มตามความเหมือนกันของเนื้อหาของเอกสารกลุ่ม (Cluster) นั้นประกอบด้วยเอกสารที่มีความคล้ายคลึงกันระหว่างเอกสารทั้งหมดที่มีความเหมือนกันภายในกลุ่มสูง (High Intra Cluster Similarity) และเอกสารที่มีความแตกต่างกันจากกลุ่มอื่น คือมีความเหมือนกันระหว่างกลุ่มน้อย (Low Inter Cluster Similarity) เอกสารจัดกลุ่มแล้วนั้นถูกพิจารณาด้วยงานที่ไม่มีการตรวจสอบว่าสามารถจัดหมวดหมู่เอกสารโดยการค้นหาภายใต้รูปแบบ กระบวนการเรียนรู้คือไม่มีการตรวจตรา ซึ่งหมายถึงว่าไม่จำเป็นต้องกำหนดผลลัพธ์ที่ถูกต้อง เช่น ผลลัพธ์ที่เกิดขึ้นจริงอยู่ที่การวางแผนก่อนการนำเข้าแต่ละครั้ง

การจัดกลุ่มเอกสารใช้กับผลลัพธ์ที่ไม่สามารถแยกความแตกต่างได้ของระบบค้นคืนเอกสารโดยแสดงผลลัพธ์ไว้ในหัวข้อเฉพาะ หลีกเลี่ยงจากการสร้างภาพของการค้นหาผลลัพธ์ที่ใช้ในการออกแบบหมวดหมู่ และการค้นหาความเหมือนกันที่มีการจัดหัวข้อหมวดหมู่ไว้ เช่น ยาฮูและ ไดเร็กทอรี่เปิด (Open Directory: dmoz.org) ถูกสร้างด้วยมือ แต่ขั้นตอนนี้สามารถช่วยจัดหมวดหมู่ของเอกสารตัวอย่างที่ใหญ่มากได้ การจัดกลุ่มสามารถช่วยให้การค้นหาที่เหมือนกันได้เร็วขึ้น ซึ่งเอกสารใกล้เคียงนั้นจะถูกค้นคืนมาที่เพิ่มขึ้นมาคือการจัดกลุ่มข้อความตามประโยคทำให้เป็นปัจจัยหลักสำหรับการเพิ่มประสิทธิภาพของระบบการรับรู้ด้วยเสียงอัตโนมัติ

มีหลายเทคนิคที่นำมาใช้สำหรับจัดหากลุ่มในข้อมูล เช่น อัลกอริธึมเคมีน (K-mean Algorithm), อัลกอริธึมที่คำนวณค่าคาดหวังที่มากที่สุด (Expectation Maximization Algorithm) และ เทคนิคการแบ่งกลุ่มข้อมูลแบบตามลำดับชั้น (Hierarchical Clustering)

2.1.4.5 วิธีการประเมินผล

วิธีการประเมินผล (Evaluation Methods) นั้นเป็นการเปรียบเทียบประสิทธิภาพของการค้นคืนผลลัพธ์จากการทำเหมืองข้อมูลเอกสาร โดยวัดจาก ค่าความแม่นยำ (Precision) และ ค่าความระลึก (Recall)

ค่าความแม่นยำ (P) เป็นอัตราส่วนของการค้นพบเอกสารที่ถูกต้องจากจำนวนเอกสารทั้งหมดที่ทำการค้นคืนมาได้ สมการดังนี้

$$\mathrm{P} = \frac{\text{จำนวนเอกสารที่ถูกต้องที่ค้นคืนได้}}{\text{จำนวนเอกสารทั้งหมดที่ค้นคืนออกมาได้}}$$

ค่าความระลึก (R) เป็นอัตราส่วนของการค้นพบเอกสารที่ถูกต้องจากจำนวนเอกสารที่ถูกต้องทั้งหมด สมการดังนี้

$$\mathrm{R} = \frac{\text{จำนวนเอกสารที่ถูกต้องที่ค้นคืนได้}}{\text{จำนวนเอกสารที่ถูกต้องทั้งหมด}}$$

2.1.5 โปรแกรมตรรกะเชิงอุปนัย

2.1.5.1 วิธีการโปรแกรมตรรกะเชิงอุปนัย

การโปรแกรมตรรกะเชิงอุปนัยเป็นการนำวิธีการเรียนรู้ของเครื่องมาใช้ร่วมกับวิธีการโปรแกรมเชิงตรรกะ โดยพยายามสร้างทฤษฎีที่ครอบคลุมตัวอย่างบวก (E^+) แต่ไม่ครอบคลุมตัวอย่างลบ (E^-) การโปรแกรมตรรกะเชิงอุปนัย มีความแตกต่างจากการเรียนรู้ของเครื่องแบบอื่นในส่วนที่ได้มีการนำเอาโปรแกรมเชิงตรรกะมาใช้ในการอธิบายตัวอย่างและสามารถสร้างความรู้ภูมิหลัง (B) ในรูปของตรรกะลำดับที่หนึ่งได้ (First Order Logic) ระบบจะทำการเรียนรู้เพื่อสร้างทฤษฎีในรูปแบบของโปรแกรมเชิงตรรกะจากตัวอย่างและความรู้ภูมิหลังที่ได้รับ โดยทฤษฎีที่ได้นี้สามารถนำไปใช้ในการจำแนกตัวอย่างใหม่ที่ระบบยังไม่เคยเห็นหรือไม่ได้ใช้ในการสอนได้ ซึ่งการ

ใช้โปรแกรมตรรกะเชิงอุปนัย ในการอธิบายสมมติฐาน (H) และตัวอย่าง ทำให้มีข้อดีคือ ความรู้ส่วนใหญ่ของมนุษย์สามารถอธิบายได้ในรูปของตรรกะอันดับที่หนึ่ง ในขณะที่วิธีการเรียนรู้ของเครื่องส่วนใหญ่จะอธิบายในรูปของตรรกศาสตร์ประพจน์ ทำให้อธิบายความรู้ของมนุษย์ได้ไม่ดีเท่าที่ควร และการใช้โปรแกรมตรรกะในการอธิบายแนวคิดที่ได้จากการเรียนรู้ยังสามารถอธิบายแนวคิดได้ซับซ้อนมากกว่าการอธิบายด้วยตรรกศาสตร์ประพจน์ และสามารถใช้ความรู้ภูมิหลังในการเรียนรู้ได้ง่ายกว่าการเรียนรู้ของเครื่องแบบอื่นที่ไม่ได้ใช้การโปรแกรมเชิงตรรกะ

สมการการเรียนรู้ของโปรแกรมตรรกะเชิงอุปนัย แสดงได้ดังนี้

$$\forall e \text{ in } E^{+} \ ((B \wedge H) \models e) \qquad \text{สมการที่ (1)}$$

$$\forall e \text{ in } E^{-} \ ((B \wedge H) \not\models e) \qquad \text{สมการที่ (2)}$$

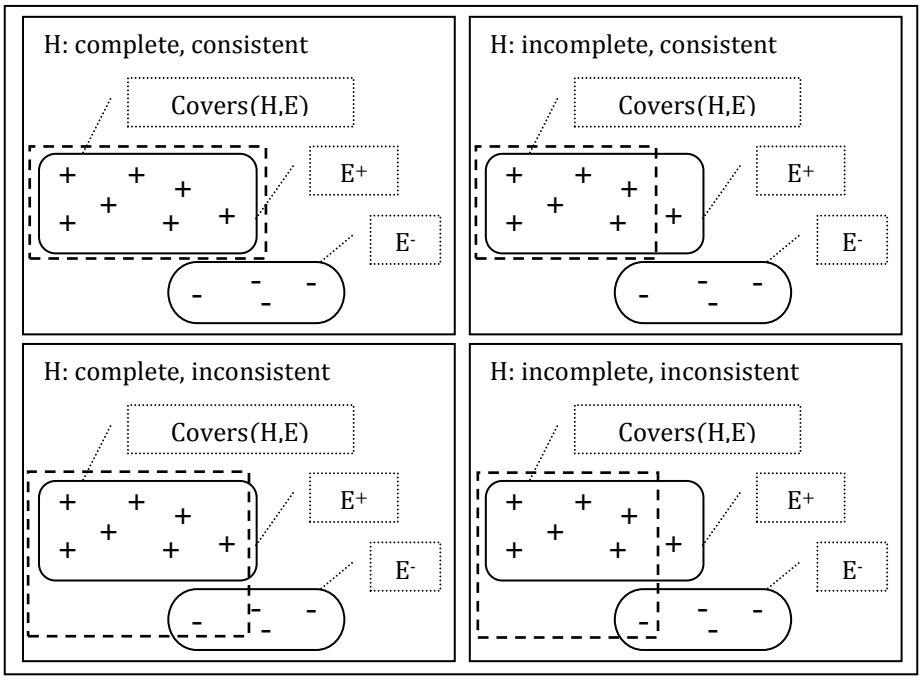
จากสมการข้างต้นนั้นสามารถอธิบายได้ว่าการเรียนรู้ของโปรแกรมตรรกะเชิงอุปนัยจะใช้วิธีการสร้างสมมติฐานโดยเริ่มต้นจากการนำชุดข้อมูลตัวอย่างบวก E^{+}, ชุดข้อมูลตัวอย่างลบ E^{-} และความรู้ภูมิหลัง B ที่อยู่ในรูปแบบของกฎลำดับที่หนึ่ง โดยพยายามค้นหาสมมติฐาน H ที่สามารถจำแนกตัวอย่าง E^{+}, E^{-} ได้ โดยจะทำการค้นหาแล้วนำความสัมพันธ์มาต่อท้ายเพื่อเพิ่มความยาวให้กับสมมติฐานนั้น เพื่อที่จะสามารถค้นหาสมมติฐานที่จำแนกตัวอย่างบวก E^{+} และตัวอย่างลบ E^{-} ได้ ทำให้ได้กฎ (Rules) ที่จำแนกตัวอย่างบวกและลบออกจากกันได้ โดยกฎที่ค้นหามาได้นั้นจะต้องสอดคล้องกับสมการข้างต้น กล่าวคือ สมมติฐานที่จะสามารถนำมาสร้างกฎได้นั้นจะต้องครอบคลุมตัวอย่างบวกทุกตัวอย่าง ตามเงื่อนไขของสมการที่ 1 และต้องไม่ครอบคลุมตัวอย่างลบเลย ตามเงื่อนไขของสมการที่ 2 ดังนั้น H จะสมบูรณ์และสอดคล้องกันกับความรู้ภูมิหลัง B และตัวอย่าง E เมื่อมีความรู้ภูมิหลัง สามารถอธิบายได้ดังนี้

<u>นิยาม</u> ความสมบูรณ์ (Complete) คือ สมมติฐาน H จะสมบูรณ์กับความรู้ภูมิหลัง B และตัวอย่าง E โดยที่ตัวอย่างบวกทั้งหมดถูกครอบคลุม

<u>นิยาม</u> ความสอดคล้องกัน (Consistent) คือ สมมติฐาน H จะสอดคล้องกันกับความรู้ภูมิหลัง B และตัวอย่าง E โดยที่ตัวอย่างลบทั้งหมดไม่ถูกครอบคลุม

ภาพที่ 2.10

แสดงความสมบูรณ์และความสอดคล้องกันของสมมติฐาน



ที่มา: (Lavrač and Džeroski, 1994)

ตามสมมติฐานของอนุประโยคของฮอร์น (Horn Clause) นั้น สามารถที่จะสร้างกฎขึ้นมาตามความสัมพันธ์ของความรู้ภูมิหลังและการเรียนรู้ ซึ่งจะมีรูปแบบดังนี้

```
daughter(X, Y) :- female(X),  parent(Y,X).
```

สำหรับกฎที่ได้จากการสร้างของไอแอลพีนั้น จะต้องมีความต้องกันและความสมบูรณ์ หลังจากกำหนดความรู้ภูมิหลังและตัวอย่างการเรียนรู้ การให้ความรู้ภูมิหลังที่แตกต่างกันทำให้มีผลกระทบต่อกฎที่ถูกสร้างขึ้นได้ ทำให้ความรู้ภูมิหลังจากความสัมพันธ์ของ `mother` และ `father` แทนที่จะกำหนดเป็น `parent` ดังนั้นกฎที่ได้อาจจะแตกต่างกันออกไป ดังนี้

```
daughter(X, Y) :- female(X),  mother(Y,X).
daughter(X, Y) :- female(X),  father(Y,X).
```

ตัวอย่างปัญหาไอแอลพี เช่น การหาความสัมพันธ์ `daughter(X,Y)` ซึ่งกำหนดให้ X คือลูกสาวของ Y ส่วนความรู้ภูมิหลังที่อยู่ในรูปแบบของความสัมพันธ์ `female` และ `parent` ดังตารางที่ 2.1 ที่มีการแสดงตัวอย่างบวกและตัวอย่างลบ

ตารางที่ 2.1

แสดงโครงสร้างครอบครัวในการเรียนรู้

ตัวอย่างเรียนรู้	ความรู้ภูมิหลัง	
daughter(mary,ann). (+)	parent(ann,mary).	female(ann).
daughter(eve,tom). (+)	parent(ann,tom).	female(mary).
daughter(tom,ann). (-)	parent(tom,eve).	female(eve).
daughter(eve,ann). (-)	parent(tom,ian).	

ที่มา: `(Lavrač and Džeroski, 1994)`

การใช้งานโปรแกรมตรรกะเชิงอุปนัยในการสร้างสมมติฐานที่ครอบคลุมตัวอย่างแบบไม่เจาะจงโดยมีวิธีค้นหาสมมติฐานโดยเริ่มจากสมมติฐานที่มีความเฉพาะเจาะจงมากที่สุดก่อนและจึงขยายสมมติฐานนั้นด้วยการลดทอนเงื่อนไขความสัมพันธ์หรือการเพิ่มสมมติฐานใหม่เข้าไปเนื่องด้วยการลดเงื่อนไขในแต่ละครั้งนั้นต้องทำการตรวจสอบสมมติฐานใหม่ว่าจะต้องไม่ครอบคลุมตัวอย่างลบ และในงานอีกด้านหนึ่งจะมุ่งเน้นในการค้นหาสมมติฐานที่มีความเฉพาะเจาะจงสูงเพื่อนำมาใช้อธิบายตัวอย่างที่ต้องการ โดยเริ่มจากสมมติฐานที่ไม่เฉพาะเจาะจงเลย แล้วจึงทำการเพิ่มเงื่อนไขหรือเพิ่มกฎที่เฉพาะเจาะจงมากขึ้นเรื่อยๆ

แต่เนื่องจากวิธีการโปรแกรมตรรกะเชิงอุปนัยจะต้องทำการสร้างสมมติฐานขึ้นมาเพื่อทดสอบกับตัวอย่างจำนวนมาก ดังนั้นหากจะทำการทดสอบให้ครบทุกกรณีที่เป็นไปได้นั้น อาจจะไม่เหมาะสมในการนำไปปฏิบัติจริง เนื่องด้วยวิธีการที่ใช้ในการค้นหาของวิธีการเรียนรู้โปรแกรมตรรกะเชิงอุปนัยมีส่วนสำคัญในการเลือกและค้นหาสมมติฐานที่เหมาะสม โดยหลักการนั้น จากงานวิจัย ของ S. Muggleton และคณะ (1994) สามารถนิยามวิธีการอุปนัย (Induction) เป็นส่วนกลับของวิธีการนิรนัย (Deduction) ดังนี้

สมมติฐาน G จะมีความเป็นทั่วไป (General) มากกว่าสมมติฐาน S เมื่อ $G \models S$ และกล่าวได้คือ สมมติฐาน S มีความเฉพาะเจาะจง (Specific) มากกว่าสมมติฐาน G

กฎการอนุมานแบบอุปนัย (Inductive Inference Rule) r โดยทาบกลุ่มรวมของอนุประโยคในสมมติฐาน G ลงบนกลุ่มรวมของอนุประโยคในสมมติฐาน S เพื่อให้ได้ $G \models S$ จะเรียกได้ว่า r เป็นกฎที่ทำให้มีความเป็นกลางมากขึ้น

กฎการอนุมานแบบนิรนัย (Deductive Inference Rule) r โดยทาบกลุ่มรวมของอนุประโยคในสมมติฐาน G ลงบนกลุ่มรวมของอนุประโยคในสมมติฐาน S เพื่อให้ได้ $G \models S$ จะเรียกได้ว่า r เป็นกฎที่ทำให้เฉพาะเจาะจงมากขึ้น

2.1.5.2 **เทคนิคไอแอลพีพื้นฐาน** (Basic ILP Techniques)

ในการค้นหาปริภูมิสมมติฐานสามารถกระทำได้ทั้งจากล่างขึ้นบน (Bottom-Up) หรือบนลงล่าง (Top-Down) เทคนิคการวางนัยทั่วไป (Generalization Technique) จะค้นหาปริภูมิสมมติฐานในลักษณะจากล่างขึ้นบน คือเริ่มจากตัวอย่างเรียนรู้ (สมมติฐานที่เฉพาะเจาะจงมากที่สุด) และค้นหาปริภูมิสมมติฐานโดยใช้ตัวกระทำการวางนัยทั่วไป (Generalization Operator) ซึ่งเทคนิคนี้เหมาะสำหรับการเรียนรู้แบบโต้ตอบ (Interactive Learning) และการเรียนรู้แบบเพิ่ม (Incremental Learning) จากตัวอย่างน้อยๆ แต่เทคนิคการแจงจำเพาะ (Specialization Technique) จะค้นหาปริภูมิสมมติฐานจากบนลงล่าง จากกรณีทั่วไป จนถึงกรณีเฉพาะเจาะจง โดยการใช้ตัวกระทำการแจงจำเพาะ (Specialization Operator) เทคนิคนี้เหมาะสำหรับการเรียนรู้เชิงประสบการณ์ (Empirical Learning) ที่มีสัญญาณรบกวนเล็กน้อย เนื่องจากการค้นหาแบบบนลงล่างนั้นสามารถกระทำได้ง่ายขึ้นด้วยวิธีการฮิวริสติก

1. <u>เทคนิคการวางนัยทั่วไป</u> (Generalization Techniques) โดยทั่วไปจะทำการค้นหาปริภูมิสมมติฐานในลักษณะล่างขึ้นบน เริ่มจากอนุประโยคที่เฉพาะเจาะจงมากที่สุดที่ครอบคลุมตัวอย่างที่ให้มา แล้วทำให้อนุประโยคนั้นมีลักษณะทั่วไปมากขึ้นเรื่อยๆ จนไม่สามารถกระทำให้มีลักษณะทั่วไปได้มากขึ้น โดยจะต้องไม่ครอบคลุมตัวอย่างลบด้วย

สำหรับการสร้างสมมติฐานจากล่างขึ้นบน นัยทั่วไป c' ของ $c(c' < c)$ ได้จากการใช้ตัวกระทำการวางนัยทั่วไปโดยวิธี θ-subsumption

<u>นิยาม</u> ตัวกระทำการวางนัยทั่วไป กำหนดให้ L คืออคติทางภาษา ตัวกระทำการวางนัยทั่วไป p จะจับคู่อนุประโยค c กับเซตของอนุประโยค $p(c)$ ซึ่งคือนัยทั่วไปของ c

$$p(c) = \{c' \mid c' \in L, c' < c\}$$

ตัวกระทำการวางนัยทั่วไปจะทำ 2 วากยสัมพันธ์พื้นฐานกับอนุประโยคคือ

1. การแทนค่าแบบย้อนกลับ (inverse substitution) กับอนุประโยค

2. การกำจัดสัญพจน์ออกจากส่วนลำตัวของอนุประโยค

<u>เทคนิคอาร์แอลจีจี</u> (Relative Least General Generalization)

แอลจีจี (Plotkin, 1970, p. 153-163) มีความสำคัญกับไอแอลพี เนื่องจากมันเป็นรูปแบบพื้นฐานของอัลกอริธึมการวางนัยทั่วไป ซึ่งจะทำการค้นหาแบบล่างขึ้นบน ของ θ-subsumption การวางนัยทั่วไปจะสมมติว่า ถ้า 2 อนุประโยค c_1 และ c_2 เป็นจริง $\mathrm{lgg}(c_1,c_2)$ จะเป็นจริงด้วย

จากนิยามของ lgg ของ 2 อนุประโยค c และ c' จะได้ว่า $\mathrm{lgg}(c,c')$ คือขอบเขตบน (Upper Bound) ของ c และ c' ใน θ-subsumption หากจะคำนวณ lgg ของ 2 อนุประโยค lgg ของพจน์ อะตอม และสัญพจน์ จะต้องถูกกำหนดขึ้นมาก่อนดังนี้

ก. แอลจีจีของพจน์ $\mathrm{lgg}(t1,t2)$

1. $\mathrm{lgg}(t,t) = t$,
2. $\mathrm{lgg}(f(s_1,....,s_n), f(t_1,....,t_n)) = f(\mathrm{lgg}(s_1,t_1),...,(\mathrm{lgg}(s_n,t_n))$,
3. $\mathrm{lgg}(f(s_1,....s_m), g(t_1,....t_n)) = V$, โดยที่ $f \neq g$ และ V คือตัวแปรซึ่งแสดงใน $\mathrm{lgg}(f(s_1,....s_m), g(t_1,....t_n))$,
4. $\mathrm{lgg}(s,t) = V$, โดยที่ $s \neq t$ และ ใน s และ t มีอย่างน้อยที่สุด 1 ตัวแปร ในกรณีนี้ V คือตัวแปรที่แทน $\mathrm{lgg}(s,t)$

ข. แอลจีจีของอะตอม $\mathrm{lgg}(A_1,A_2)$

1. $\mathrm{lgg}(p(s_1,....,s_n), p(t_1,....,t_n)) = p(\mathrm{lgg}(s_1,t_1),...,\mathrm{lgg}(s_n,t_n))$, ถ้าอะตอมมีสัญลักษณ์เพรดิเคต p เหมือนกัน
2. $\mathrm{lgg}(p(s_1,....,s_m),q(t_1,....,t_n))$ จะไม่นิยาม ถ้า $p \neq q$

ค. แอลจีจีของสัญพจน์ $\mathrm{lgg}(L_1,L_2)$

1. ถ้า L_1 และ L_2 คืออะตอม แล้ว $\mathrm{lgg}(L_1,L_2)$ จะถูกคำนวณคล้ายข้างต้น
2. ถ้าทั้ง L_1 และ L_2 คือสัญพจน์ลบ $L_1 = \bar{A}_1$ และ $L_1 = \bar{A}_2$ แล้ว $\mathrm{lgg}(L_1,L_2) = \mathrm{lgg}(\bar{A}_1,\bar{A}_2) = \overline{\mathrm{lgg}(A_1,A_2)}$,
3. ถ้า L_1 คือสัญพจน์บวก และ L_2 คือสัญพจน์ลบ หรือในทางกลับกัน $\mathrm{lgg}(L_1,L_2)$ จะไม่นิยาม

ง. แอลจีจีของอนุประโยค $\mathrm{lgg}(c_1,c_2)$

ให้ $c_1 = \{L_1,...,L_n\}$ และ $c_2 = \{K_1,...,K_m\}$ แล้ว $lgg(c_1,c_2) = \{M_{ij} = lgg(L_i,K_j) \mid Li \in c_1, Kj \in c_2$ และ $lgg(L_i,K_i)$ จะต้องถูกกำหนด}

นิยาม อาร์แอลจีจี (rlgg) ของ 2 อนุประโยค c_1 และ c_2 คือ แอลจีจีของ c_1 และ c_2 $lgg(c_1,c_2)$) ที่มีความสัมพันธ์กับความรู้ภูมิหลัง B

ถ้าความรู้ภูมิหลังประกอบด้วยความจริงพื้นฐาน และ K คือความจริงพื้นฐานทั้งหมดเหล่านี้ที่เชื่อมต่อกัน อาร์แอลจีจีของ 2 อะตอมพื้นฐาน A_1 และ A_2 (ตัวอย่างบวก) ที่มีความสัมพันธ์กับความรู้ภูมิหลัง K จะถูกนิยามได้ดังนี้

$$rlgg(A_1,A_2) = lgg((A_1 \leftarrow K), A_2 \leftarrow K))$$

ตัวอย่างเช่น ให้ตัวอย่างบวก $e_1 = daughter(mary,ann)$ และ $e_2 = daughter(eve,tom)$ และความรู้ภูมิหลัง B ดังตารางที่ 2.1 แอลจีจีของ $e1$ และ $e2$ ที่สัมพันธ์กับ B คือ

$$rlgg(e_1,e_2) = lgg((e_1 \leftarrow K), (e_2 \leftarrow K))$$

โดยที่ K คือการเชื่อมต่อกันของสัญพจน์ parent(ann,mary), parent(ann,tom), parent(tom,eve), parent(tom,ian), female(ann),female(mary), และ female(eve) ผลลัพธ์ที่ได้คือ

$$rlgg(e_1,e_2) = daughter(X,Y) \leftarrow female(X), parent(Y,X)$$

2. เทคนิคการแจงจำเพาะ (Specialization Techniques)

เทคนิคการแจงจำเพาะจะค้นหาปริภูมิสมมติฐานในลักษณะบนลงล่าง จากสมมติฐานกว้างๆ ลงมาหาสมมติฐานแคบๆ โดยการใช้ θ-subsumption ด้วยตัวกระทำการแจงจำเพาะหรือตัวกระทำรีไฟน์เมนต์ (refinement operator)

นิยาม ตัวกระทำรีไฟน์เมนต์กำหนดให้ L คืออคติทางภาษา ตัวกระทำรีไฟน์เมนต์ p จะจับคู่อนุประโยค c กับเซตของอนุประโยค $p(c)$ ซึ่งก็คือรีไฟน์เมนต์ของ c

$$p(c) = \{c' \mid c' \in L, c < c'\}$$

ตัวกระทำรีไฟน์เมนต์จะทำ 2 วากยสัมพันธ์พื้นฐานกับอนุประโยคคือ

1. แทนค่ากับอนุประโยค และ
2. เพิ่มสัญพจน์ในส่วนลำตัวของอนุประโยค

เทคนิคไอแอลพีแบบเฉพาะเจาะจงพื้นฐานคือการค้นหาแบบบนลงล่างของกราฟรีไฟน์เมนต์ (Top-Down Search of Refinement Graph) ตัวเรียนรู้จะเริ่มเรียนรู้จากอนุประโยคที่มีลักษณะทั่วไปมากที่สุดแล้วค่อยๆ ทำให้มีลักษณะเฉพาะเจาะจงลงมาเรื่อยๆ จนกระทั่งไม่ครอบคลุมตัวอย่างลบ โดยระหว่างที่ทำการค้นหาจะต้องมีอย่างน้อยที่สุดหนึ่งตัวอย่างบวกที่ตรงตามอนุประโยคนั้นๆ

กราฟรีไฟน์เมนต์คือ กราฟที่มีทิศทางและไม่มีวงวน โดยที่โหนด (Node) แต่ละโหนดแทนอนุประโยคโปรแกรม และกิ่งของกราฟแทนตัวกระทำรีไฟน์เมนต์พื้นฐาน เช่นการแทนค่าตัวแปรด้วยพจน์และการเพิ่มสัญพจน์เข้าไปในส่วนลำตัวของอนุประโยค

ตัวอย่างการค้นหากราฟรีไฟน์เมนต์จากตัวอย่างความสัมพันธ์ของครอบครัว เริ่มแรกเรียนรู้จะได้รับตัวอย่างบวก $e_1 = \text{daughter(mary,ann)}$ โดยปกติโหนดบนสุดของกราฟรีไฟน์เมนต์จะมีลักษณะทั่วไปมากที่สุด การค้นหาเริ่มจากลักษณะทั่วไปมากที่สุดของเพรดิเคต daughter

$c = \text{daughter(X,Y)} \leftarrow$ โดยที่ส่วนลำตัวที่ว่างเปล่าหมายถึงส่วนลำตัวเป็นจริง ดังนั้น c จะครอบคลุม $e1$ กำหนดให้สมมติฐาน H เริ่มต้นคือ $H = \{c\}$ หลังจากนั้นจะรับตัวอย่างที่สอง $e_2 = \text{daughter(eve,tom)}$ เข้ามา ซึ่งเป็นตัวอย่างบวกและถูกครอบคลุมด้วย H ดังนั้นจึงไม่มีการปรับเปลี่ยนสมมติฐาน เมื่อตัวอย่างที่สาม $e_3 = \text{daughter(tom,ann)}$ ถูกรับเข้ามา ซึ่งเป็นตัวอย่างลบ ตัวเรียนรู้จะกำจัดอนุประโยค c ออกจาก H เนื่องจากมันครอบคลุมตัวอย่างลบ ระบบจะพยายามสร้างอนุประโยคใหม่ให้ครอบคลุมตัวอย่างบวกอันแรกนั่นคือ e_1 เนื่องจาก c ครอบคลุมตัวอย่างลบ e_3 ตัวเรียนรู้จังต้องสร้างเซตของการปรับเปลี่ยนของมันใหม่ในรูปดังนี้

$$p(c) = \{\text{daughter(X,Y)} \leftarrow L\}$$

โดยที่ L คือ สัญพจน์ที่มีพารามิเตอร์เป็นตัวแปรจากส่วนหัวของอนุประโยค เช่น $X = Y$, female(X), female(Y), parent(X,X), parent(X,Y), parent(Y,X), และ parent(Y,Y) และ สัญพจน์ที่สร้างตัวแปร Z โดยที่ $Z \neq X$ และ $Z \neq Y$ ขึ้นมาในส่วนลำตัวของอนุประโยค parent(X,Z), parent(Z,X), parent(Y,Z), และ parent(Z,Y) ดังภาพที่ 2.11

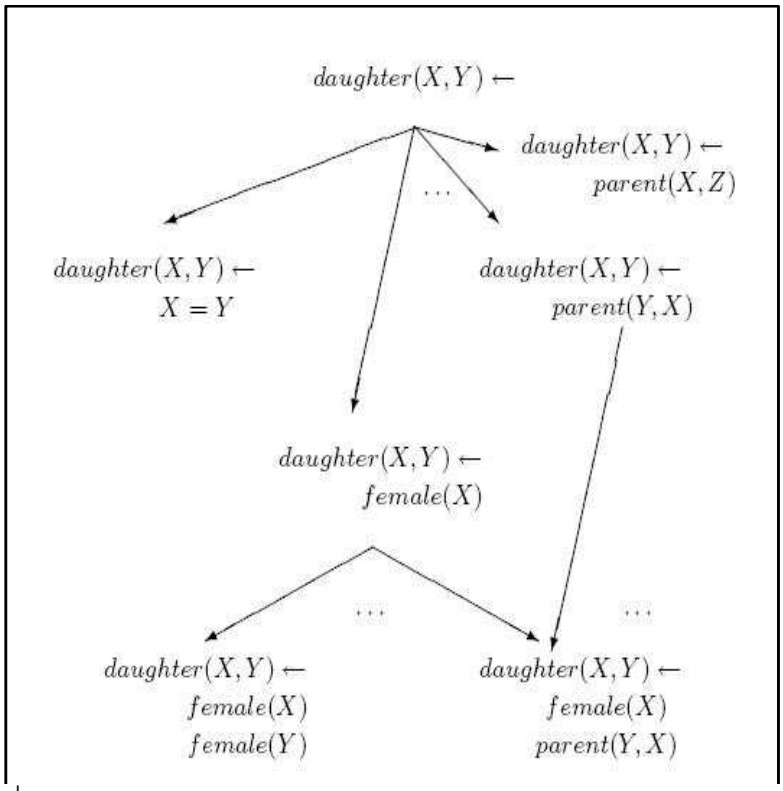
จากกราฟด้านล่าง การปรับ $\text{daughter(X,Y)} \leftarrow X = Y$ จะถูกพิจารณาเป็นอันดับแรก แต่จะถูกกำจัดออกไปเนื่องจากไม่ครอบคลุมตัวอย่างบวก $e1$ การปรับ $c' = \text{daughter(X,Y)} \leftarrow \text{female(X)}$ จะถูกพิจารณาเป็นลำดับถัดไป และพบว่ามันครอบคลุมตัวอย่างบวกแต่ไม่ครอบคลุม

ตัวอย่างลบ ดังนั้นมันจะถูกนำเข้าไปเป็นสมมติฐาน H อย่างไรก็ตามหากมีตัวอย่างลบ e4 = daughter(eve,ann) เข้ามา อนุประโยค c' ก็จะไม่มีความสอดคล้องกับ e4 เนื่องจากครอบคลุม e4 อนุประโยค c' ก็จะถูกกำจัดออกจากสมมติฐาน H ดังนั้น $H = \phi$

กระบวนการสร้างอนุประโยคจะถูกดำเนินการต่อไปเรื่อยๆ และระบบจะพยายามสร้างอนุประโยคที่ครอบคลุม e1 ในกระบวนการนี้ การปรับอนุประโยค c ที่เหลืออยู่จะถูกพิจารณาเป็นอันดับแรก นั่นคือ daughter(X,Y) ← female(Y) และ daughter(X,Y) ← parent(Y,X) จะถูกนำเข้าไปในคิวของการค้นหาทางกว้างก่อน (Breadth-first Search) เนื่องจากมันครอบคลุม e1 แต่ไม่สอดคล้องกับตัวอย่างลบ daughter(X,Y) ← X = Y จึงถูกนำมาพิจารณาต่อไป และจะถูกกำจัดทิ้งเนื่องจากไม่ครอบคลุม e1 ท้ายที่สุด จะค้นพบ c' เป็น daughter(X,Y) ← female(X), parent(Y,X) ซึ่งครอบคลุม e1 และสอดคล้องกับตัวอย่างลบ และอนุประโยคนี้ยังครอบคลุม e2 ดังนั้นตัวเรียนรู้จะหยุดการค้นหา

ภาพที่ 2.11

แสดงส่วนหนึ่งของกราฟรีไฟน์เมนต์ของปัญหาความสัมพันธ์ของครอบครัว



ที่มา: (Lavrač and Džeroski, 1994)

2.1.6 ทฤษฎีกราฟ และสมสัณฐานของกราฟย่อย

กราฟไม่ระบุทิศทาง (Undirected Graph) หรือ กราฟ G คือคู่อันดับ ที่แสดงโดย $G \coloneqq (\mathrm{V}, \mathrm{E})$ ซึ่งมีส่วนประกอบพื้นฐาน คือ เซตของจุดยอด V (Vertices) หรือ จุด (Nodes) กับเซตของคู่ของจุดยอดที่ต่างกัน ซึ่งเรียกว่าเส้นเชื่อม E (Edges) หรือ เส้น (Lines) ซึ่งเส้นเชื่อมทุกเส้นจะมีจุดยอดปลาย (End Vertices) ของเส้นเชื่อม

ส่วนกราฟระบุทิศทาง (Directed Graph) หรือ ไดกราฟ G คือคู่อันดับ ที่แสดงโดย $G \coloneqq (V, A)$ ซึ่งมีส่วนประกอบพื้นฐานคือเซตของจุดยอด V หรือ จุด กับเซตของคู่ลำดับของจุดยอด ซึ่งเรียกว่าเส้นเชื่อมระบุทิศทาง A (Directed Edges), เส้นเชื่อม (Arcs) หรือ ลูกศร (Arrows) โดยที่เส้นเชื่อม $e = (x, y)$ จะแสดงว่าเป็นเส้นเชื่อมจาก x ไป y โดยที่ y เป็น หัว (Head) และ x เป็น หาง (Tail) ของเส้นเชื่อม

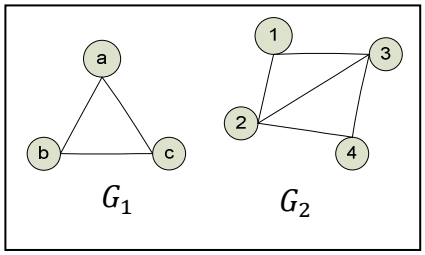
และคุณลักษณะของกราฟ จะกล่าวว่าเส้นเชื่อมสองเส้นประชิด (Adjacent) กัน เมื่อเส้นเชื่อมสองเส้นมีจุดปลายร่วมกัน และจุดยอดสองจุดประชิดกัน เมื่อจุดยอดทั้งสองเป็นจุดปลายของเส้นเชื่อมเดียวกัน ซึ่งเส้นเชื่อมต่อ (Incident) กับจุดของเส้นเชื่อมเสมอ สำหรับกราฟถ่วงน้ำหนัก (Weighted Graph) คือกราฟที่มีการกำหนดค่าให้กับเส้นเชื่อมแต่ละเส้น ซึ่งอาจจะเป็นค่า เช่น ค่าใช้จ่าย ระยะทาง หรือน้ำหนัก เป็นต้น ซึ่งกราฟถ่วงน้ำหนักสามารถนำไปใช้แก้ปัญหาหลายอย่างเช่น ปัญหาการค้นหาเส้นทางที่ดีที่สุด โดยทั่วไปแล้วนั้นจุดยอดของกราฟนั้นจะไม่สามารถถูกแยกแยะหรือพิจารณาได้ว่าแตกต่างกัน ยกเว้นกรณีที่มีจำนวนเส้นเชื่อมที่มีความแตกต่างกัน ซึ่งบางครั้งมีการระบุจุดยอดด้วยการระบุชื่อ (Label) ที่ชัดเจนกำกับ เรียกกราฟชนิดนี้ว่า กราฟระบุชื่อ (Labeled Graph)

กราฟย่อยของกราฟ G คือกราฟที่มีเซตของจุดยอด และเซตของเส้นเชื่อมเป็นเซตย่อยของ G สำหรับความสมสัณฐานของกราฟย่อยนั้นเป็นปัญหาหนึ่งของทฤษฎีกราฟ (สำหรับวิชา Computational Theory) และเป็นปัญหาการตัดสินใจ (Decision Problem) ในระดับเอ็นพีบริบูรณ์ โดยปัญหาความสมสัณฐานของกราฟนั้น จะพิจาณาจากความเหมือนกันของกราฟส่วนมากจะถูกนำไปประยุกต์ใช้ในงานวิจัยโครงสร้างทางเคมี และการออกแบบวงจรไฟฟ้า

Subgraph Isomorphism (G_1, G_2) โดยข้อมูลนำเข้าเป็นกราฟ G_1และ G_2 และเมื่อกราฟ G_1เป็นสมสัณฐานของกราฟ G_2 ต้องมีลักษณะตามสมการ ดังนี้คือ

$F: V(G_1) \rightarrow V(subgraph(G_2))$ ตัวอย่างเช่น ภาพที่ 2.12

ภาพที่ 2.12
แสดงกราฟย่อย G_1 เป็นสมสัณฐานของกราฟ G_2



โดยกำหนดให้ $F(a) = 1, F(b) = 2$ และ $F(c) = 3$ ซึ่งจะสอดคล้องกับทฤษฎีของกราฟย่อยคือ ให้กราฟ $G = (V, E)$ และ กราฟ $G' = (V', E')$

G' เป็นกราฟย่อยของ G ถ้า $V' \subseteq V$ และ $E' \subseteq E$

2.1.7 **ฐานความรู้อภิธานศัพท์เวิร์ดเน็ต** (WordNet)

ฐานความรู้อภิธานเวิร์ดเน็ต ของ Christiane (1998) เป็นคลังข้อมูลภาษาออนไลน์ ซึ่งเป็นที่รู้จักกันอย่างกว้างขวาง พัฒนาโดยห้องปฏิบัติการเกี่ยวกับกระบวนการรับรู้ (Cognitive Science Laboratory) มหาวิทยาลัยพรินซ์ตัน โดย ศาสตราจารย์ จอร์จ เอ มิลเลอร์ (Professor George A. Miller) และคณะ ตั้งแต่ปี ค.ศ. 1985 และพัฒนาอย่างต่อเนื่องจนถึงปัจจุบัน เวิร์ดเน็ตนั้นเป็นการรวมรวมข้อมูลทางภาษาอังกฤษจากตัวบท (Text) ประเภทต่างๆ เช่น ตัวบทที่มีเนื้อหาสำหรับการให้ข้อมูลข่าวสาร (Informative) หรือตัวบทที่เป็นเรื่องแต่ง (Imaginative) โดยเน้นข้อมูลเกี่ยวกับคำ (Words) และความหมายของคำ (Semantics) โดยให้ข้อมูลของคำพ้องความหมาย หรือกล่าวได้อีกนัยหนึ่งคือเวิร์ดเน็ตจะทำการจัดกลุ่มของคำโดยรวบรวมตามความหมาย โดยจะจัดกลุ่มคำที่มีความหมายเดียวกัน หรือคำพ้องความหมาย (Synonyms) ไว้ด้วยกันเรียกว่า กลุ่มของคำศัพท์ (Synset) เช่น คำว่า good, competent, qualified, smart และ excellent เป็นต้น นั้นต่างก็มีความหมายว่า “ดี” ดังนั้นคำกลุ่มนี้จึงถูกจัดอยู่ในกลุ่มของคำศัพท์เดียวกัน หากจะกล่าวอย่างเฉพาะเจาะจงว่าเวิร์ดเน็ตคือคลังข้อมูลทางความหมาย (Semantic Lexicon)

จากข้างต้นนั้นเวิร์ดเน็ตถูกพัฒนาอย่างต่อเนื่อง ในปี ค.ศ. 2005 เวิร์ดเน็ตได้รวบรวมคำและความหมายของคำในภาษาอังกฤษจำนวน 150,000 คำ และมีการจัดกลุ่มของคำศัพท์ไว้

มากกว่า 115,000 กลุ่ม ทั้งนี้ภายในเวิร์ดเน็ตจะมีการจัดประเภทของคำเป็น คำนาม กริยา คุณศัพท์ คำวิเศษณ์ ซึ่งในกลุ่มคำศัพท์แต่ละกลุ่ม จะประกอบด้วยกลุ่มของคำพ้องความหมาย และรูปแบบการปรากฏร่วมกันของคำแต่ละคำกับคำอื่นๆ (Collocation) เช่น คำว่า “car” ปรากฏร่วมกับคำว่า “pool” เมื่อรวมกันแล้วจะได้เป็น “car pool” หมายถึง “กลุ่มของคนที่มาใช้รถร่วมกันเพื่อไปสถานที่เดียวกัน”

ปัจจุบันเวอร์ชันล่าสุดของเวิร์ดเน็ตคือเวอร์ชัน 2.1 โดยใช้ร่วมกับโปรแกรมวินโดว์ (Windows), ยูนิกซ์ (Unix), ลินุกซ์ (Linux) หรือ โซลาริส (Solaris) และสามารถใช้งานออนไลน์ได้ที่ http://wordnet.princeton.edu หรือสามารถดาวน์โหลดโปรแกรมมาใช้ได้

สำหรับสรุปรายละเอียดของจำนวนคำ จำนวนกลุ่มคำศัพท์ และจำนวนความหมายที่ถูกจัดเก็บ และแสดงผลในเวิร์ดเน็ต (เวอร์ชัน 2.1) นั้นจะถูกแสดงไว้ในตารางที่ 2.2 ดังนี้

ตารางที่ 2.2
แสดงจำนวนคำ, จำนวนกลุ่มคำศัพท์, และจำนวนความหมายที่ถูกจัดเก็บ
และแสดงผลใน เวิร์ดเน็ต (เวอร์ชัน 2.1)

ประเภทของคำ (Part of Speech)	จำนวนคำ	จำนวนกลุ่มคำศัพท์	จำนวนความหมาย
คำนาม	117,097	81,426	145,104
คำกริยา	11,488	13,650	24,890
คำคุณศัพท์	22,141	18,877	31,302
คำกริยาวิเศษณ์	4,601	3,644	5,720
จำนวนรวม	155,327	117,597	207,016

จากตารางข้างต้น แสดงให้เห็นว่าจำนวนคำทั้งหมดในเวิร์ดเน็ต (เวอร์ชัน 2.1) มีจำนวนถึง 155,327 คำ แสดงให้เห็นว่าเป็นคลังข้อมูลทางภาษาอังกฤษขนาดใหญ่ที่สามารถนำมาใช้ในการศึกษาภาษาอังกฤษได้เป็นอย่างดี และสามารถกล่าวได้ว่า ด้วยจำนวนของข้อมูลที่มีขนาดใหญ่นี้ จึงเป็นสิ่งที่แสดงให้เห็นว่าข้อมูลเวิร์ดเน็ตเป็นเสมือนตัวแทนทางภาษาอังกฤษได้เป็นอย่างดี

โดยวิธีการสืบค้นนั้นสามารถแสดงผลของข้อมูลที่สำคัญโดยแสดงเฉพาะความหมายหลักของคำ (Sense Keys) ส่วนการสืบค้นข้อมูลจากเวิร์ดเน็ตนั้นสามารถค้นหาข้อมูลเกี่ยวกับคำที่สนใจ โดยมีรายละเอียดต่างๆ ดังนี้

1. ประเภททางไวยกรณ์ของคำที่ค้นหา เช่น เป็นคำนาม (Noun), กริยา (Verb), คุณศัพท์ (Adjective), หรือ กริยาวิเศษณ์ (Adverb)

2. ความหมายของคำ หากคำที่ค้นหามีหลายความหมาย (Polysemy count) เวิร์ดเน็ตจะแสดงความหมายทุกความหมายของคำนั้น

3. ความถี่ของคำ (Frequency score) เป็นการแสดงจำนวนครั้งที่มีคำที่ค้นหาปรากฏในฐานข้อมูล เช่น คำว่า “university” มีปรากฏ 54 ครั้งในฐานข้อมูล

4. ข้อมูลอื่นๆ เช่น คำพ้องความหมายของคำที่ค้นหา (Synonyms) คำที่มีความหมายครอบคลุมความหมายของคำที่ค้นหาหรือคำแม่กลุ่ม (Hypernyms) และคำที่มีความหมายเฉพาะภายใต้ความหมายของคำที่ค้นหา หรือคำร่วมกลุ่ม (Hyponyms)

2.2 งานวิจัยที่เกี่ยวข้อง

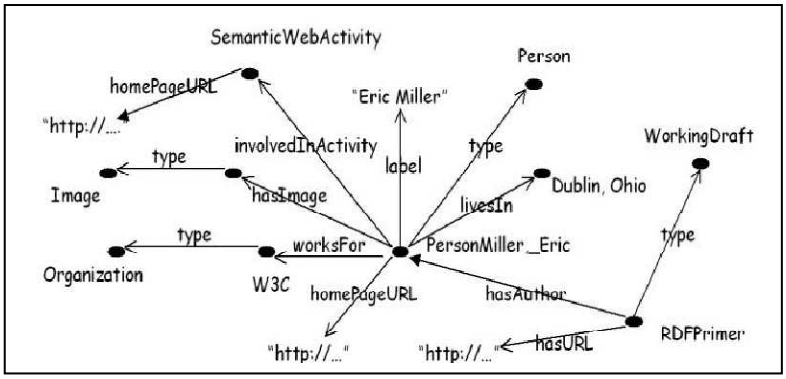
งานวิจัยเรื่อง การโปรแกรมตรรกะเชิงอุปนัยและแบ็กพรอพาเกชันนิวรอลเน็ตเวิร์กในการรู้จำตัวพิมพ์อักษรไทย (สุกรี สินธุภิญโญ, 2541) นำเสนอวิธีการโปรแกรมตรรกะเชิงอุปนัยและแบ็กพรอพาเกชันนิวรอลเน็ตเวิร์กในการรู้จำตัวพิมพ์อักษรไทย โดยการนำโปรแกรมตรรกกะเชิงอุปนัย มารวมกับการประมาณเพื่อเลือกฎที่ใกล้เคียงกันกับตัวอย่างที่มีสัญญาณรบกวนโดยแบ็กพรอพาเกชันนิวรอลเน็ตเวิร์ก ซึ่งการทดลองโดยการใช้จำนวนของสัญพจน์ (Literal) ที่ไม่ตรงกับตัวอย่าง และจำนวนสัญพจน์ที่ตรงกับตัวอย่างเป็นอินพุตเวกเตอร์ ทำให้กระบวนการเรียนรู้สำหรับอัตราการรูจำของแบบแบ็กพรอพาเกชั่นนิวรอลเน็ตเวิร์กมีค่าสูงกว่าอัตราการรู้จำของแบบโปรแกรมตรรกะเชิงอุปนัย แต่การเรียนรู้จำแบบแบ็กพรอพาเกชั่นนิวรอลเน็ตเวิร์กนั้นจะให้ความสำคัญกับทุกสัญพจน์ของแต่ละกฎเทียบเท่ากัน ทำให้ไม่สามารถกำหนดน้ำหนักแก่สัญพจน์ที่มีความสำคัญมากกว่าได้ ด้วยเหตุนี้จึงมีการทดลองโดยใช้ค่าความจริงของสัญพจน์ของกฎแต่ละข้อ เพื่อแทนจำนวนสัญพจน์ที่ตรงกับตัวอย่างและไม่ตรงกับตัวอย่างเป็นอินพุตเวกเตอร์ ทำให้อัตราการรู้จำสูงขึ้น

งานวิจัยเรื่อง การประมาณกฎจากวิธีการโปรแกรมตรรกะเชิงอุปนัยด้วยวิธีการแบ็กพรอพาเกชันนิวรอลเน็ตเวิร์กในการรู้จำตัวพิมพ์อักษรไทย (สุกรี สินธุภิญโญ, 2544) นำเสนอเกี่ยวกับการประมาณกฎจากวิธีการโปรแกรมตรรกะเชิงอุปนัยด้วยวิธีการแบ็กพรอพาเกชั่นนิวรอลเน็ตเวิร์กในการรู้จำตัวพิมพ์อักษรไทย ได้นำเสนอแนวทางในการแก้ปัญหาของการโปรแกรมตรรกะเชิงอุปนัย ซึ่งกฎที่ได้จากไอแอลพีนั้นไม่สามารถจำแนกตัวอย่างใหม่ หรือตัวอย่างที่มีสัญญาณรบกวนได้ ทั้งนี้เนื่องจากไอแอลพีจะทำการเลือกกฎที่ตรงกับตัวอย่างแล้วทำการจำแนกตามกฎนั้น แต่ถ้าไม่มีกฎที่ตรงกับตัวอย่างเลย ไอแอลพีจะไม่สามารถทำการจำแนกตัวอย่างนั้นได้ แนวทางแก้ปัญหาคือวิธีการดึงลักษณะสำคัญ (Feature Extraction Algorithm) มาใช้ร่วมกับแบ็กพรอพาเกชั่นนิวรอลเน็ตเวิร์กในการประมาณกฎที่ใกล้เคียงกับตัวอย่าง จากการทดลองพบว่าเปอร์เซ็นความถูกต้องเพิ่มขึ้นจากการใช้แค่กฎไอแอลพีอย่างเดียวในการจำแนกตัวอย่าง โดยเฉพาะปัญหาที่มีลักษณะเป็นหลายกลุ่ม (Multi-Class Problem) รวมถึงทดลองความทนทานต่อสัญญาณรบกวนในกรณีที่ชุดข้อมูลแบ่งเป็นสองกลุ่ม โดยผลการทดลองพบว่าเปอร์เซ็นต์ความถูกต้องของวิธีการนี้ลดลงช้ากว่ากฎจากไอแอลพี

งานวิจัยของ R. Guha (2003) นั้นศึกษาและพัฒนาโปรแกรมเกี่ยวกับ Semantic Search เพื่อรองรับเทคโนโลยีการค้นหาหน้าเว็บที่หลายหลายมากขึ้น โดยการนำวิธี TAP ซึ่งเป็นโปรแกรมพื้นฐานสำหรับการสร้างการค้นหาเชิงความหมาย (Semantic Search) โดยพัฒนาความสัมพันธ์ของการสืบค้นด้วยการสอบถาม (Search Query) และ เพิ่มประสิทธิภาพในการค้นหาจากเดิม ด้วยแนวความคิดที่ว่าข้อมูลที่มีอยู่บนเว็บเชิงความหมายนั้นจะต้องมีความสัมพันธ์กันและสามารถเชื่อมโยงถึงกันได้ และใช้ RDF ตามมาตรฐานของ W3C มาเพื่ออธิบายคำศัพท์รูปแบบของ RDFS เพื่ออธิบายความหมายสำหรับรายละเอียดแหล่งข้อมูลและระหว่างความสัมพันธ์ โดยที่ความสัมพันธ์ที่มีอยู่ระหว่างเอชทีเอ็มแอลกับเว็บเชิงความหมายเสมือนขยายจากเว็บปัจจุบัน โดยมีกลุ่มที่เชื่อมโยงกันดีจากโหนดในเว็บเชิงความหมายไปยัง เอกสารเอชทีเอ็มแอล ความสัมพันธ์ดังกล่าวเป็นชนิดการเชื่อมต่อด้วยแนวความคิดเว็บเชิงความหมายด้วยหน้าเว็บที่มีความเกี่ยวข้องกันมากที่สุด

ภาพที่ 2.13

กราฟแสดงความหมายและความสัมพันธ์ของข้อมูล ของ Eric Miller



จากภาพข้างต้น แสดงให้เห็นแหล่งข้อมูลที่เกี่ยวข้อกับ Eric Miller เป็นข้อมูลที่เกี่ยวกับบุคคลซึ่งมีคนมากมายที่มีชื่อเดียวกันนี้ แสดงถึง Eric Miller เป็นส่วนหนึ่งของบุคคลที่ใช้ชื่อนี้ เมื่อทำการสืบค้นด้วย “Eric Miller” ทำให้พบแหล่งข้อมูลของ “PersonMiller_Eric” จากนั้นทำการค้นหาคุณสมบัติทั้งหมดก่อนจึงจะแสดงผลการค้นหาออกมา

งานวิจัยของ Li Ding และคณะ (2004) ศึกษาถึงเว็บเชิงความหมาย เนื่องด้วยในปัจจุบันเว็บเชิงความหมายจะมีเอกสารบนเว็บที่ถูกเขียนอยู่ในรูปแบบของ RDF และ OWL เอกสารเว็บเชิงความหมายเป็นเอกสารเพื่อแสดงคำอธิบายประกอบความหมายของคำและความหมายสำหรับการอ้างอิง โดยเฉพาะสำหรับภาษาที่จำเป็นเพื่อให้คนและคอมพิวเตอร์สามารถสื่อสารกันได้เข้าใจ งานวิจัยนี้พัฒนาเสิร์ชเอ็นจินเว็บเชิงความหมาย เรียกว่า Swoogle มาเพื่อประโยชน์ในการค้นหาข้อมูลเชิงความหมาย

งานวิจัยของ Zhenjiang Lin และคณะ (2006) ศึกษาเครื่องมือวัดความเหมือนกันของหน้าเว็บในการสืบค้นบนเว็บเรียกว่า Ageism ซึ่งตรงกันข้ามกับ Simian แต่นำมาใช้กำหนดและอ้างอิงถึง โดย Ageism สามารถวัดความเหมือนกันระหว่างหน้าเว็บใดๆ สองหน้าเว็บ ที่ Simian ไม่สามารถทำได้ในบางกรณี เนื่องจากการค้นหาหน้าเว็บที่มีความเหมือนกันโดยการสืบค้นจากเสิร์ชเอ็นจิน ปัจจุบันมีความหลากหลายในการวัดความเหมือนกันจากการเชื่อมโยงดูจาก ไฮเปอร์ลิงก์ที่มีในหน้าเว็บ โดยใช้อัลกอริธึมเช่น อัลกอริธึมการเปรียบเทียบ (companion algorithm),

อัลกอริธึมการอ้างอิงรวม (co citation algorithm) และ Simian เป็นต้น ในส่วนของงานวิจัยนี้ได้นำวิธี Simian มาแก้ไขในบางจุดที่สองหน้าเว็บเหมือนกันและถูกอ้างอิงถึงโดยหน้าเว็บที่เหมือนกัน (recursive definition)

งานวิจัยของ บังอร กลับบ้านเกาะ (2543) นั้นเสนอเกี่ยวกับการค้นคืนเอกสารออนไลน์โดยประยุกต์ใช้จีเนติกอัลกอริธึมเพื่อเพิ่มประสิทธิภาพในการค้นคืนสารสนเทศ ภายใต้เวกเตอร์สเปซโมเดล ซึ่งการค้นคืนสารสนเทศนั้นขึ้นอยู่กับความคล้ายคลึงระหว่างเอกสารและสืบค้น เอกสารมีความคล้ายคลึงกับคิวรีสูงย่อมแสดงว่าเอกสารนั้นมีความสัมพันธ์กับคิวรีมากกว่าและควรจะได้รับการค้นคืนขึ้นมาก่อน ในขั้นตอนของอัลกอริธึมเชิงพันธุกรรมนั้น คิวรีจะถูกแทนด้วยโครโมโซม ซึ่งโครโมโซมเหล่านี้จะถูกนำเข้าสู่กระบวนการจีเนติกโอเปอเรเตอร์ต่างๆ กัน ได้แก่ การคัดเลือก การครอสโอเวอร์ และการมิวเตชัน จนกระทั้งได้โครโมโซมคิวรีที่เหมาะสมเพื่อนำไปค้นคืนสารสนเทศต่อไป

งานวิจัยของ Raymond J. Mooney และคณะ (2002) การค้นพบการเชื่อมต่อ (Link Discovery) เป็นงานสำคัญในการทำเหมือนข้อมูลสำหรับการนับจำนวนลัทธิก่อการร้ายจากข้อมูลอ้างอิงของเครือข่ายสำนักงานโครงการวิจัยขั้นสูงของกระทรวงกลาโหม (DARPA'Evidence Extraction and Link Discovery) เพื่อค้นหาการเชื่อมโยงหรือแบบแผนในการโจมตีที่อาจจเกิดขึ้นได้จากข้อมูลที่มีความสัมพันธ์กันอยู่เป็นจำนวนมาก โดยการนำไอแอลพีมาค้นหากฎในรูปแบบของกฎลำดับที่หนึ่งจากฐานข้อมูลเชิงสัมพันธ์หลายมิติ

งานวิจัยของ Monika Zakova และคณะ (2006) นั้นเสนอการค้นหาความถี่ของรูปแบบ (Pattern) จากเอกสารออกแบบ (Computer Aided Design หรือ CAD) โดยโครงสร้างและรายละเอียดถูกจัดเรียงโดยใช้ความรู้พื้นฐานแบบมีลำดับชั้น (Hierarchical Background Knowledge) ด้วยการทำเหมืองข้อมูลสัมพันธ์ (Relational Data Mining) โดยนำข้อมูลจาก CAD แปลงให้อยู่ในรูปแบบของ RDF จากนั้นนำมาค้นหากฎความสัมพันธ์ (Associated Rule) ด้วยโปรแกรมตรรกะเชิงอุปนัย แล้วประยุกต์วิธีจากการค้นหากลุ่มย่อย (Subgroup) จากการค้นหาความถี่ในรูปแบบโครงสร้าง (Structural Pattern) จากลักษณะสำคัญของรายละเอียด (Descriptive Features) สำหรับรูปแบบความถี่ที่ถูกสร้างโดยใช้วิธีการค้นหาแบบใหม่โดยการเรียงลำดับ (Sorted Downward Refinement)

งานวิจัยของ Adam Schenker และคณะ (2003) นำเสนอความสัมพันธ์ของเอกสารบนเว็บโดยแสดงบนรูปแบบกราฟแทนการใช้เวกเตอร์โมเดลในแบบเดิม ซึ่งเปรียบเทียบความถูกต้องในการจำแนกแบบเวกเตอร์โดยการใช้อัลกอริธึมเพื่อนบ้านใกล้ที่สุด (k-Nearest Neighbor Algorithm: k-NN) กับวิธีการที่นำเสนอ โดยการใช้กราฟในการแสดงข้อมูลของเอกสารด้วยอัลกอริธึม k-NN โดยวิธีที่นำเสนอนั้นจะประเมินความแตกต่างของเอกสารบนเว็บด้วยการใช้การวัดความถูกต้องในการจำแนกการเข้าใกล้เพื่อวัดความถูกต้องในการจำแนกเว็บ จากผลการทดลองสรุปออกมาว่าการแสดงในรูปแบบกราฟ k-NN นั้นสามารถให้ความถูกต้องมากกว่าการใช้เวกเตอร์ k-NN ทั้งความแม่นยำและเวลาในการประมวลผล

งานวิจัยของ Sofia Stamou และคณะ (2006) นั้นเสนอวิธีโดยการจัดหมวดหมู่หน้าเว็บโดยอัตโนมัติโดยการใช้ลำดับชั้นของหัวข้อใหญ่ (subject hierarchy) ในโครงสร้างไดเรคทอรีสำหรับการจำแนกหน้าเว็บไว้ในแต่ละหมวดหมู่ของหัวข้อสำคัญนั้นจะใช้ใจความคำสำคัญของหน้าเว็บ (thematic word) โดยค้นหาหน้าเว็บที่ใกล้เคียงที่สุดกับหัวข้อโดยการใช้อัลกอริธึมในการจัดหาตำแหน่ง (Rank Algorithm) ของหน้าเว็บ รวมถึงการจัดเรียงอย่างมีลำดับ ซึ่งใช้พจนานุกรมคำศัพท์เวิร์ดเน็ตในการจัดลำดับชั้น โดยการรวมเอา SUMO และ MWND มารวมกัน

บทที่ 3

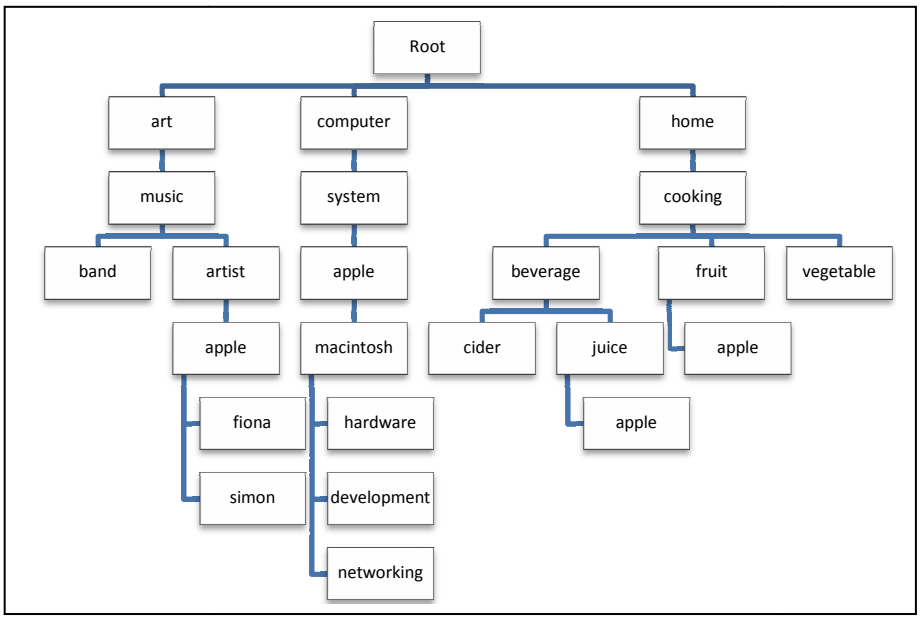
วิธีการดำเนินงานวิจัย

เนื่องจากงานวิจัยนี้มีจุดประสงค์เพื่อนำเสนอวิธีการจัดหมวดหมู่ของหน้าเว็บที่เหมือนกัน ดังนั้นในบทนี้จึงนำเสนอวิธีการดำเนินงานวิจัยที่ประกอบด้วย การออกแบบจำลองสำหรับการสืบค้นข้อมูลบนเว็บ โดยแบบจำลองประกอบด้วย กระบวนการที่สำคัญคือ การพัฒนาฐานความรู้ การจัดหมวดหมู่โดยการหาความเหมือนกันของหน้าเว็บ และการทดลองเพื่อประเมินประสิทธิภาพแบบจำลองที่พัฒนาขึ้น ดังนี้

3.1 กราฟความสัมพันธ์ของคำศัพท์แนวคิด

กราฟความสัมพันธ์ของคำศัพท์แนวคิดเป็นเสมือนฐานความรู้ที่ถูกจัดทำขึ้นตามแนวความคิดตามโครงสร้างภาษาให้อยู่ในรูปแบบความสัมพันธ์ของคำที่ประกอบด้วยคำนาม (Nouns) โดยการสร้างในรูปแบบความสัมพันธ์ลำดับชั้น (Hierarchy Relation) ดังตัวอย่างดังนี้

ภาพที่ 3.1
แสดงลำดับชั้นฐานความรู้ของ Apple



จากภาพที่ 3.1 เป็นลำดับชั้นฐานความรู้โดยยึดพื้นฐานจาก DMOZ ของคำศัพท์ว่า Apple นั้นมีความหมายในหลายโดเมน เช่น Apple ที่อยู่ในโดเมนของผลไม้ หรือ Apple ที่อยู่ในโดเมนเกี่ยวกับคอมพิวเตอร์ เป็นต้น นั้นขึ้นอยู่กับความคิดและความต้องการใช้ ณ เวลานั้นของผู้ใช้ที่ต้องการค้นหา เป้าหมายของขั้นตอนในการสร้างฐานความรู้เสมือน คือ การได้กลุ่มคำหลักของฐานความรู้ที่ต้องการมุ่งเน้น เพื่อทำหน้าที่เป็นตัวแทนเชิงความหมายของแต่ละเอกสาร รวมถึงระดับความสำคัญเชิงความหมายของเอกสาร เพื่อเป็นข้อมูลสำหรับการเปรียบเทียบระดับความเหมือนกัน

3.2 **การตัดคำหลักและสกัดคำหลัก** (Keyword Extraction)

การตัดคำหลักและสกัดคำหลักเป็นขั้นตอนการตัดข้อความโดยเก็บข้อมูลเอกสารบนเว็บจากแท็กเมตา เช่น หัวเรื่อง, คำสำคัญ, รายละเอียด (Description) และ ลิงก์ ออกเป็นกลุ่มของคำ ซึ่งเป็นคำหลักที่มีความหมาย ในที่นี้หมายถึง คำนาม เป้าหมายของการตัดคำคือ การได้มาซึ่งคำ เพื่อใช้เป็นตัวแทนของเอกสาร เพื่อนำไปสืบค้นเชิงความหมาย

ภาพที่ 3.2

แสดงเอกสารบนเว็บ

```
<title>Apple</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8">
<meta http-equiv="pragma" content="no-cache">
<meta name="Author" content="Apple, Inc.">
<meta name="Keywords" content="Apple">
<link rel="home" href="http://www.apple.com/">
```

จากภาพที่ 3.2 นั้นเป็นภาพแสดงเอกสารบนเว็บซึ่งในแต่ละเอกสารประกอบด้วยข้อมูลที่ถูกจัดไว้ในตำแหน่งที่หลากหลายโดยจะเลือกคำที่อยู่ภายใต้แท็กหัวเรื่อง (Title Tag) แท็กเมตา และแท็กลิงค์ (Link Tag)

หลักการและเครื่องมือในการตัดคำในภาษาอังกฤษโดยใช้เวิร์ดเน็ตเป็นเครื่องมือ ที่เป็นฐานข้อมูลคำหลักและเป็นที่รวบรวมคำหลักที่มีความหมายใกล้เคียงกันและความสัมพันธ์ระหว่างคำหลัก การตัดคำเพื่อให้ได้คำนาม ซึ่งถูกสกัดออกจากข้อความ ตามกระบวนการดังนี้

1. การตัดคำ (Word Segmentation) เป็นเทคนิคในการแบ่งตัวอักษรจากข้อความ (String) เพื่อหาขอบเขตของแต่ละหน่วยคำ (Morpheme) ในเอกสารทั้งหมดออกมาโดยให้อยู่ในลักษณะของคำชัดเจน (Distinct Word) ซึ่งจะนำมาใช้เป็นดัชนี หรือลักษณะสำคัญ (Feature) ของเอกสาร

2. ปรับตัวอักษรในเอกสารให้เป็นตัวพิมพ์เล็กทุกตัว (Lowercase Letter) เพื่อลดความแตกต่างของตัวอักษรในการวิเคราะห์คำ

3. การกำจัดคำที่ไม่มีนัยสำคัญหรือคำหยุด (Stop Words Elimination) เป็นการตัดคำที่ไม่ใช่คำหลักออก ซึ่งเป็นกลุ่มคำที่ไม่อาจส่งผล หรือไม่ได้ทำให้การพิจารณาเกิดข้อแตกต่าง เพื่อกรองคำที่ไม่สื่อถึงความสำคัญของเอกสารออกไป โดยการตัดคำด้วยช่องว่าง (Space Character) ตัวอย่างของคำหยุด เช่น is, am, are, a, for, from, who, what, where, do, does, done, on, for, then และ the

4. การกำจัดอักษรพิเศษ (Special Character) เป็นการกำจัดตัวอักษรพิเศษต่างๆ ที่เกิดขึ้นในเอกสารทั่วๆ ไป เช่น $ % * & เป็นต้น โดยทำการตัดคำด้วยช่องว่าง

5. วิเคราะห์หารากศัพท์ของคำ (Stemming) เช่น เซตของคำ {compute, computed, computer, computing} และสำหรับคำที่เป็นพหูพจน์ (Plural) จะทำให้อยู่ในรูปแบบเอกพจน์ (Singular) เมื่อผ่านกระบวนการนี้แล้วจะได้คำคือ {comput} โดยอัลกอริธึมที่ใช้ในการวิเคราะห์หารากศัพท์ของคำ คือ อัลกอริธึมพอร์เตอร์สเต็มมิ่ง (Porter Stemming Algorithm)

6. ความสัมพันธ์เชิงความหมาย เช่น ความสัมพันธ์ของคำที่มีคำแม่กลุ่มเดียวกัน เช่น “Bird”, “Duck” และ “Chicken” เป็นกลุ่มคำลูกกลุ่มหรือคำร่วมกลุ่มที่มีระดับชั้นต่ำลงมาภายใต้คำว่า “Animal” ที่เป็นคำแม่กลุ่ม โดยมีความสัมพันธ์ที่มีลักษณะเป็นลำดับชั้น ซึ่งได้นำคำเหล่านี้มาจากพจนานุกรมเชิงความหมายเวิร์ดเน็ต

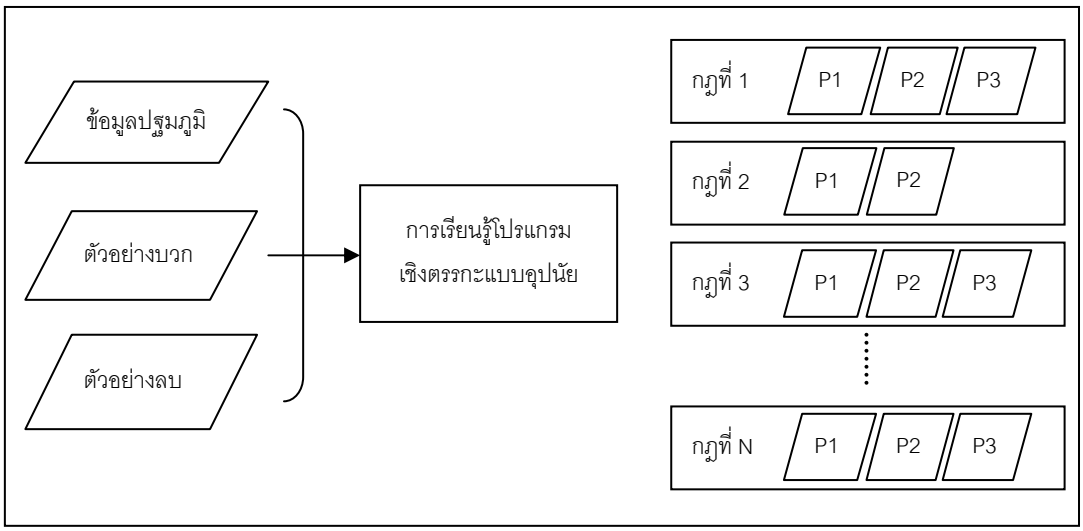
3.3 การค้นหาคำหลัก (Matching)

การค้นหาคำหลักเป็นวิธีการจัดหาคำหลักที่ค้นพบในเอกสาร จากดัชนีคำหลักที่ถูกสร้างไว้ในกราฟความสัมพันธ์ของคำศัพท์แนวคิด เพื่อจัดเตรียมไว้สำหรับการสืบค้นโดยการนำคำที่ได้จากการตัดคำและสกัดคำของแต่ละเอกสาร แล้วมาตรวจหาจากคำหลักในฐานความรู้เสมือน โดยจะเลือกเก็บเฉพาะคำที่ปรากฏในฐานความรู้เสมือนเท่านั้น

3.4 การจำแนกข้อมูลด้วยกฎจากโปรแกรมตรรกะเชิงอุปนัย (Classifying by Inductive Logic Programming)

การเรียนรู้ของโปรแกรมตรรกะเชิงอุปนัยเป็นการเรียนรู้โดยการใช้ความรู้ภูมิหลัง รวมถึงตัวอย่างบวกและตัวอย่างลบที่อยู่ในรูปแบบของกฎลำดับที่หนึ่งมาเพื่อค้นหาสมมติฐาน ที่สามารถแยกตัวอย่างบวกและตัวอย่างลบออกจากกันได้ โดยกฎที่ได้นั้นจะต้องครอบคลุมตัวอย่างบวกทั้งหมดและไม่มีตัวอย่างลบอยู่ด้วย

ภาพที่ 3.3
การเรียนรู้แบบสมบูรณ์ของโปรแกรมตรรกะเชิงอุปนัย



เนื่องจากไอแอลพีเป็นวิธีการเรียนรู้ที่มีข้อดีคือสามารถนำความรู้ภูมิหลังมาใช้ในการเรียนรู้ได้และแนวคิดหรือกฎที่ได้จากการเรียนรู้จะอยู่ในรูปแบบที่มนุษย์เข้าใจได้ง่ายโดยเขียนอยู่ในรูปแบบภาษาโปรล็อก งานวิจัยนี้จึงนำไอแอลพีมาใช้ในการจัดหมวดหมู่เว็บที่เหมือนกัน สำหรับกระบวนการในการเรียนรู้มีดังนี้

3.4.1 การสร้างความรู้ภูมิหลัง (Background Knowledge)

3.4.1.1 กลุ่มโครงสร้างกราฟ

กลุ่มโครงสร้างกราฟเป็นการแปลงกราฟคลังความรู้อยู่ในรูปแบบภาษาโปรล็อก ซึ่งคลังความรู้ (Ontology Corpus-Based) เป็นฐานข้อมูลความรู้ของคำหลักของเว็บที่สนใจในการค้นหา โดยแสดงอยู่ในรูปแบบโครงสร้างของความสัมพันธ์แบบลำดับชั้น ที่มีคำนามแสดงอยู่ในโหนดรูปแบบของข้อความบนกราฟ โดยมีการเชื่อมต่อกัน (Link) ด้วยความสัมพันธ์ของคำ ดังนั้นสามารถแสดงในรูปแบบภาษาโปรล็อกได้ดังนี้

1) *connected(word1,word2) :- haslink(word1,word2).* กล่าวคือ *word1* และ *word2* มีเส้นทางเชื่อมไปถึงกัน เมื่อ *word1* มีเส้นเชื่อมไปยัง *word2*

2) *connected(word1,word2) :- connected(word1,word3), haslink(word3,word2).* กล่าวคือ *word1* และ *word2* มีเส้นทางเชื่อมไปถึงกัน เมื่อ *word1* มีเส้นเชื่อมไปยัง *word3* และ *word3* มีเส้นเชื่อมไปยัง *word2*

3) *word(word1).* กล่าวคือ *word1* เป็นคำที่มีอยู่ในกราฟนิยามศัพท์แนวคิด

4) *sameword(word1,word2) :-word1=word2.* กล่าวคือ *word1* มีความหมายเหมือนกันกับ *word2*

3.4.1.2 กลุ่มความหมายของคำ

กลุ่มความหมายของคำเป็นการแปลงข้อมูลตามความหมายของคำอยู่ในรูปแบบภาษาโปรล็อก โดยการนำคำหลักที่มีอยู่ใน คลังความรู้ของแต่ละเว็บทำการกำหนดความหมายลักษณะคำ ดังนั้นสามารถเขียนให้อยู่ในรูปแบบภาษาโปรล็อกได้ดังนี้

1) *hasbigram(word1,word2).* กล่าวคือ *word1* จะอยู่ก่อนหน้าหรือตามหลัง *word2*

2) *ishyponym(word1,word2).* กล่าวคือ *word1* เป็นคำร่วมกลุ่มของ *word2*

3) *ishypernym(word1,word2).* กล่าวคือ *word1* เป็นคำแม่กลุ่มของ *word2*

4) *hassensekey(word1,word2).* กล่าวคือ *word2* มีลักษณะความหมายคล้ายกับ word1

3.4.1.3 กลุ่มการแปลงหน้าเว็บเป็นกราฟ

กลุ่มการแปลงหน้าเว็บเป็นกราฟ เป็นการแปลงข้อมูลหน้าเว็บในรูปแบบภาษาโปรล็อก ดังนั้นสามารถเขียนให้อยู่ในรูปแบบภาษาโปรล็อกได้ดังนี้

1) *member(word1,list1).* กล่าวคือ *word1* ที่แสดงอยู่ในกราฟอยู่ใน *list1*

2) *hasword(web,[word1,word2,..,word-n])* กล่าวคือ *Web* ประกอบด้วยลิสต์ของคำที่ได้จากการเทียบคำที่มีอยู่ในกราฟนิยามศัพท์แนวคิดของแต่ละโดเมน

3.4.2 การสร้างตัวอย่างบวก (Positive Example) และตัวอย่างลบ (Negative Example)

ในส่วนของตัวอย่างบวกในงานวิจัยนี้ได้มาจากการใช้คนในการเปรียบเทียบความเหมือนกันของเว็บเพื่อตัดสินว่าเว็บคู่ใดมีความเหมือนกันโดยยึดหลักตามการจัดหมวดหมู่ของ ODP สำหรับตัวอย่างลบเป็นคู่เว็บที่ไม่เหมือนกันนอกเหนือจากตัวอย่างบวก โดยสามารถเขียนในรูปแบบภาษาโปรล็อกของคู่เว็บที่คล้ายกันได้ดังนี้

1) *sim(web001,web002).* กล่าวคือ เว็บ *web001* และ เว็บ *web002* เหมือนกัน สำหรับคู่เว็บที่ต่างกันสามารถเขียนได้ดังนี้

2) *sim(web001,web003).* กล่าวคือ เว็บ *web001* และ เว็บ *web003* ต่างกัน

3.4.3 การสอน (Training) และการหากฎ (Induce Rule)

ในงานวิจัยนี้ใช้โปรแกรม Aleph ในการสอนและการหากฎ โดยการนำความรู้ภูมิหลัง ตัวอย่างบวก และตัวอย่างลบนำเข้าไปสอน กับชุดข้อมูลตัวอย่างสอน (Training Set) จนได้กฎที่สามารถอ่านแล้วเข้าใจได้ ซึ่งกฎที่ได้มาจากสมมติฐานที่ตั้งไว้

3.4.4 ค่าความแม่นยำ (Accuracy)

ค่าความแม่นยำหรือค่าความถูกต้องเป็นผลลัพธ์ในการแสดงค่าความแม่นยำในการเรียนรู้

3.4.5 การทดสอบ (Testing)

ภายหลังจากการสอนแล้วจะทำการนำกฎที่ได้มาทำการทดสอบกับชุดทดสอบ (Testing Set) เพื่อหาค่าความแม่นยำ

3.5 วิธีการทดสอบความถูกต้องแบบไขว้ข้าม

วิธีในการทดสอบความถูกต้องแบบไขว้ข้าม (Cross-Validation) เป็นวิธีการหนึ่งที่ใช้ในการประมาณค่าความผิดพลาดในกฎหรือวิธีการในการเรียนรู้ของเครื่อง หลักการของวิธีการนี้คือ การทดสอบความถูกต้องแบบไขว้ข้ามโดยการใช้กลุ่มข้อมูลสำหรับการทดสอบซึ่งได้มาจากกลุ่มข้อมูลที่ใช้ในการทดลอง โดยแยกกลุ่มข้อมูลสำหรับการทดสอบ ออกจากกลุ่มข้อมูลที่ใช้สำหรับเรียนรู้ และจะไม่นำกลุ่มข้อมูลสำหรับทดสอบมาใช้จนกว่าการเรียนรู้ทั้งหมดจะเสร็จสิ้น ซึ่งประสิทธิภาพที่ได้จากการทดสอบเป็นการประมาณค่าผิดพลาดที่ได้ของกฎหรือวิธีการนั้นๆ โดยวิธีการทดสอบความถูกต้องแบบไขว้ข้ามที่เป็นที่นิยมวิธีการหนึ่งคือ วิธีทดสอบความถูกต้องแบบไขว้ข้าม N ชุดข้อมูลย่อย (N-fold Cross-Validation)

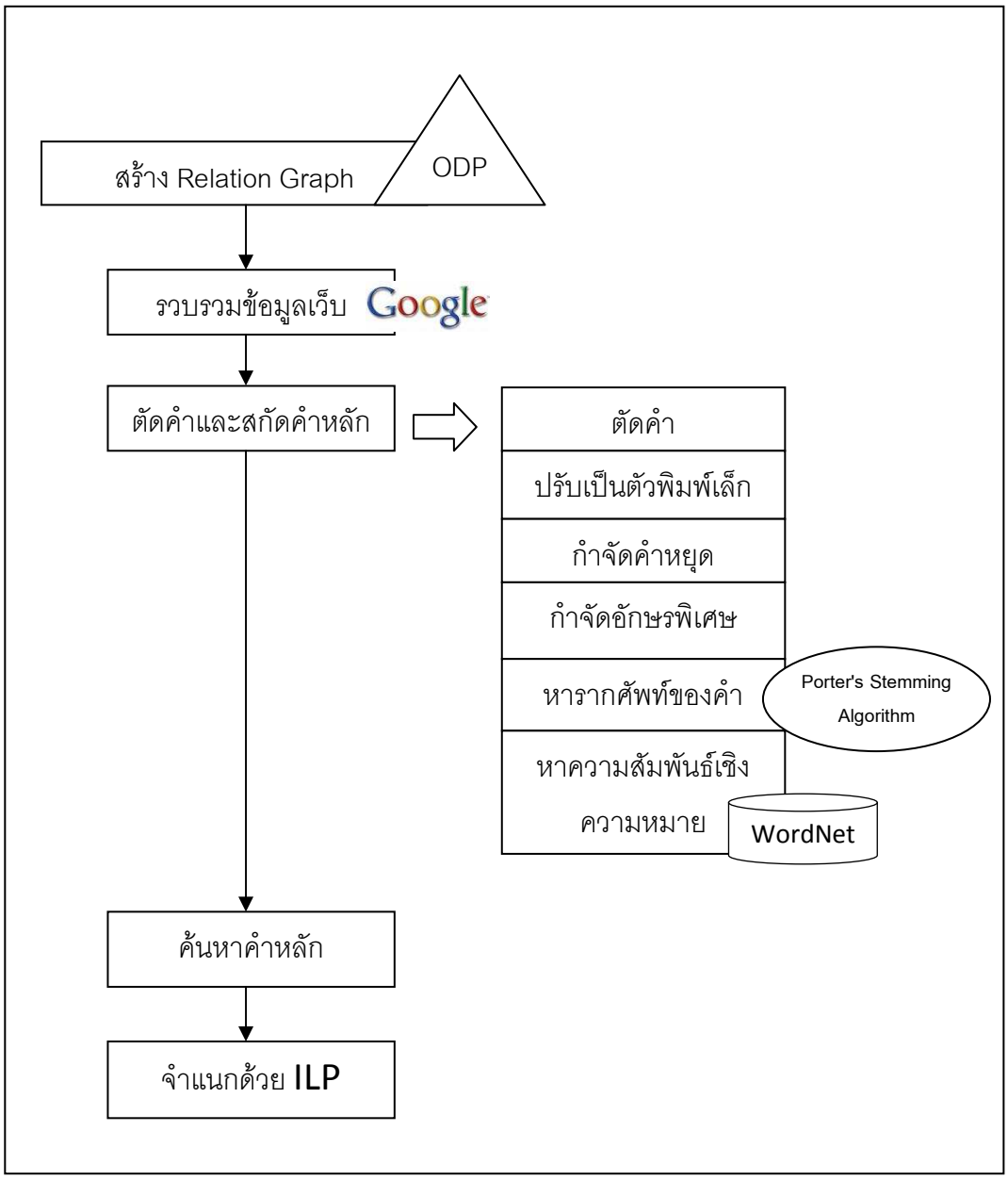
แนวคิดในการทำงานของการทดสอบความถูกต้องแบบไขว้ข้าม N ชุดข้อมูลย่อย โดยการทำซ้ำการทดสอบความถูกต้องโดยใช้เวลาเท่ากับ N ซึ่งทำการแบ่งข้อมูลในการทดลองออกเป็น N ชุดข้อมูลย่อย และในแต่ละชุดข้อมูลย่อยจะประกอบด้วยข้อมูลตัวอย่างที่ได้จากข้อมูลที่กระจายกัน

ในการใช้งานทั่วไปนักวิจัยได้ใช้การทดสอบความถูกต้องแบบไขว้ข้าม โดยใช้จำนวน N ชุดข้อมูลย่อยเท่ากับ 5 หรือ 10 ชุดข้อมูลย่อย ซึ่งจากการค้นคว้าเกี่ยวกับงานวิจัยที่ได้นำวิธีการทดสอบความถูกต้องแบบไขว้ข้ามมาใช้ พบว่าในการใช้จำนวน N ชุดข้อมูลย่อยเท่ากับ 10 หรือเรียกว่า การทดสอบความถูกต้องแบบไขว้ข้าม 10 ชุดข้อมูลย่อย ซึ่งเป็นเทคนิคที่มีประสิทธิภาพในการทดสอบความถูกต้อง จากการนำเสนอเกี่ยวกับระบบคอมพิวเตอร์สำหรับการเรียนรู้ (Weiss และ Kulikowski, 1991) และมีงานวิจัยหลายงานที่ให้การยอมรับและใช้วิธีการทดสอบความถูกต้องแบบไขว้ข้ามจำนวน 10 ชุดข้อมูลย่อย เป็นวิธีการทดสอบมาตรฐานในการทดสอบความถูกต้องทางด้านการเรียนรู้ของเครื่อง

ดังนั้นขั้นตอนการทำงานของงานวิจัยนี้สามารถเขียนเป็นกระบวนการจากที่กล่าวไว้ข้างต้นได้เป็นดังนี้

ภาพที่ 3.4

แสดงกระบวนการทำงานจำแนกเว็บ



บทที่ 4

ผลการทดลอง

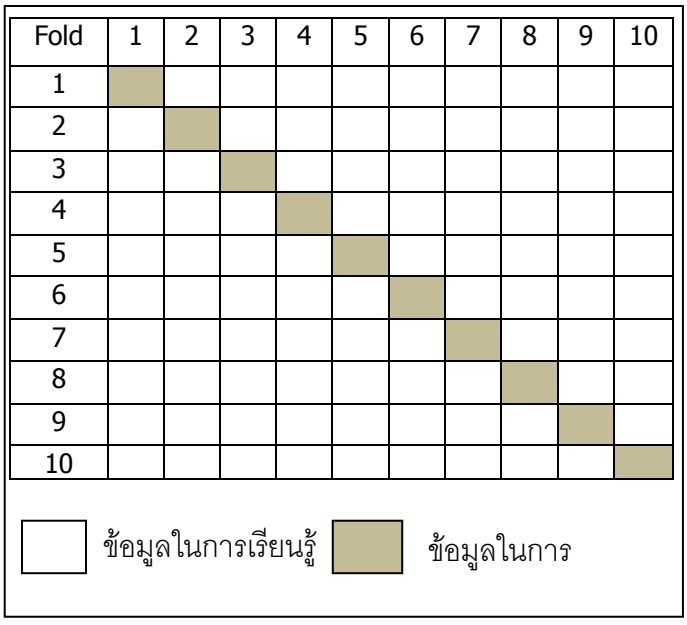
ในงานวิจัยนี้ผู้วิจัยใช้เครื่องมือของโปรแกรมตรรกะเชิงอุปนัย ALEPH ซึ่งขั้นตอนในการทดลองเป็นดังนี้

4.1 วิธีขั้นตอนในการทดลอง

4.1.1 กลุ่มข้อมูลตัวอย่างที่ใช้ในการทดลอง

วิธีการแบ่งกลุ่มข้อมูลทดลองที่ใช้ในวิทยานิพนธ์นี้ ใช้วิธีการแบ่งแบบไขว้ข้าม 10 กลุ่ม โดยแบ่งข้อมูลออกเป็น 10 กลุ่มและใช้ 9 กลุ่มข้อมูลในการเรียนรู้ และ 1 กลุ่มข้อมูลในการทดสอบวัดค่าความถูกต้องแม่นยำ ดังนั้นการทดลองที่เกิดขึ้นจะมีลักษณะจำนวนรอบตามภาพ

ภาพที่ 4.1
แสดงการจัดแบ่งข้อมูลในการทดลอง



4.1.2 ชุดข้อมูลที่ใช้ในการทดลอง

ในแต่ละชุดจะผ่านการเรียนรู้สองแบบ คือ การเรียนรู้โดยโปรแกรมตรรกะเชิงอุปนัยเพื่อจำแนก โดยไม่เพิ่มคุณสมบัติพิเศษ และ การเรียนรู้โดยโปรแกรมตรรกะเชิงอุปนัยเพื่อจำแนกโดยใช้คุณสมบัติพิเศษ

4.1.3 ค่าที่ใช้วัดผลในการทดลอง

ค่าที่เก็บไว้เพื่อเปรียบเทียบผลในการทดลองได้แก่ ค่าความถูกต้องของการจำแนกข้อมูล, เวลาทั้งหมดที่ใช้ในการเรียนรู้ของแต่ละวิธีการเรียนรู้, จำนวนกฎที่เกิดขึ้นในขั้นตอนการเรียนรู้โดยค่าที่ใช้ในการวัดนั้นมีวิธีการดังนี้

4.1.3.1 ค่าความถูกต้องของการจำแนกข้อมูล

การเปรียบเทียบว่าการจำแนกข้อมูลนั้นถูกต้องหรือไม่นั้นตามวิธีการเรียนรู้โดยโปรแกรมตรรกะเชิงอุปนัย ความถูกต้องจะหาได้จากคำตอบที่กฎของโปรแกรมตรรกะเชิงอุปนัยตอบออกมา เปรียบเทียบกับกลุ่มที่เป็นคำตอบที่คาดหวังไว้ของข้อมูลตัวอย่างทดสอบ ถ้าเป็นกลุ่มเดียวกันจะถือว่าถูกต้อง โปรแกรมตรรกะเชิงอุปนัย สามารถเก็บข้อมูลเวลาในขั้นตอนของการเรียนรู้โดยโปรแกรมตรรกะเชิงอุปนัยเป็นสำคัญ โดยใช้คำสั่งที่ช่วยในการวัดเวลาในการเรียนรู้ คำสั่งในโปรแกรม ALEPH “time (induce).” โดยตัวอย่างของผลลัพธ์ของการทดลองในการวัดค่าความถูกต้องจากการเรียนรู้ เป็นดังตารางที่ 4.1

4.1.3.2 จำนวนกฎที่เกิดขึ้นในขั้นตอนการเรียนรู้

การเก็บข้อมูลจำนวนกฎที่อยู่ในขั้นตอนการเรียนรู้โดยโปรแกรมตรรกะเชิงอุปนัย จะเก็บค่าจำนวนกฎที่เกิดขึ้นหลังจากขั้นตอนการเรียนรู้ของโปรแกรมตรรกะเชิงอุปนัย

ตารางที่ 4.1
แสดงค่าความถูกต้องที่ได้จากการเรียนรู้ของโปรแกรมตรรกะเชิงอุปนัย

ชุดเรียนรู้	ค่าความถูกต้อง (%)							
	Apple		Jaguar		Mouse		Virus	
	ไอแอลพี	ไอแอลพีคุณลักษณะพิเศษ	ไอแอลพี	ไอแอลพีคุณลักษณะพิเศษ	ไอแอลพี	ไอแอลพีคุณลักษณะพิเศษ	ไอแอลพี	ไอแอลพีคุณลักษณะพิเศษ
1	84.81	93.7	93.33	97.78	87.41	91.3	86.11	92.59
2	84.44	94.44	93.52	97.78	87.78	91.48	86.67	93.7
3	84.63	94.26	93.89	97.41	88.15	90.37	87.41	94.63
4	84.63	94.26	93.89	97.59	87.96	91.48	87.78	94.07
5	84.81	93.52	94.44	97.78	88.33	92.04	87.96	94.81
6	84.81	93.89	93.33	97.22	87.96	91.67	86.67	94.07
7	84.63	93.89	93.89	97.41	87.59	91.3	87.04	93.52
8	84.81	93.7	92.96	97.59	87.04	90.74	86.85	93.15
9	84.81	94.07	93.89	97.78	88.33	91.3	87.78	93.89
10	84.81	94.26	93.89	97.59	88.33	91.67	86.85	93.15

4.2 ข้อมูลคำค้นที่ใช้ในการทดลอง

งานวิจัยนี้ผู้วิจัยใช้คำค้นเพื่อสร้างเป็นตัวอย่างที่ใช้เป็นตัวอย่างในการเรียนรู้ ประกอบด้วยคำว่า “apple”, “jaguar”, “mouse” และ “virus” โดยได้มาจากการค้นหาในเสิร์ชเอ็นจินด้วยคำค้นดังกล่าว แล้วเลือกผลการค้นหาที่ตรงกับคำค้นที่สนใจตามโดเมนที่แตกต่างกัน 5 โดเมน โดเมนละ 5 เว็บ รวม 25 เว็บ นำมาสร้างเป็นตัวอย่างบวกและตัวอย่างลบ โดยเลือกคู่ของเว็บที่อยู่ในโดเมนเดียวกันเป็นตัวอย่างบวก และเลือกคู่ของเว็บที่ต่างโดเมนเป็นตัวอย่างลบ รวมมีตัวอย่างบวกทั้งสิ้น 400 ตัวอย่าง และตัวอย่างลบทั้งสิ้น 2,000 ตัวอย่าง ในการทดสอบนั้นจะทดสอบเพื่อสร้างกฎโดยแยกตามคำค้น

4.3 ผลการทดลอง

ผลของการทดลองจากตารางที่ 4.2 แสดงให้เห็นว่าผลลัพธ์ในการจำแนกข้อมูลด้วยโปรแกรมตรรกะเชิงอุปนัยโดยเพิ่มคุณลักษณะพิเศษนั้นให้ความถูกต้องสูงที่สุดในการเปรียบเทียบค่าความถูกต้องของวิธีการจำแนกทั้ง 3 วิธี เนื่องจากกฎที่ได้จากการจำแนกโดยโปรแกรมตรรกะเชิงอุปนัยแบบเพิ่มคุณลักษณะพิเศษ เมื่อเปรียบเทียบกับกฎจากการเรียนรู้ของโปรแกรมตรรกะเชิงอุปนัยแบบปกติ ทำให้ได้กฎที่หลากหลายมากกว่า โดยสามารถครอบคลุมตัวอย่างได้ดีกว่า และเมื่อนำมาเปรียบเทียบการสมสัณฐานของกราฟย่อยกับโปรแกรมตรรกะเชิงอุปนัยพบว่าโปรแกรมตรรกะเชิงอุปนัยสามารถให้ค่าความถูกต้องในการจำแนกข้อมูลได้ดีกว่าในทุกคำค้น เนื่องจากไอแอลพีสามารถใช้คุณลักษณะพิเศษเชิงความหมายของคำในแต่ละโหนดได้ จากการเปรียบเทียบวิธีการทดลองของแต่ละวิธีที่นำมาใช้ในการทดลองของวิทยานิพนธ์นี้ จะใช้วิธีการวัดค่าทางสถิติที่เรียกว่า การทดสอบค่าที (T-Test Value) ซึ่งเป็นค่าบอกความเชื่อมั่นของข้อมูลที่ได้โดยทำการทดสอบค่าทีกับทั้งสามวิธีการที่นำมาทดลอง

จะเห็นได้ว่าผลการทดสอบค่าทีจากวิธีการที่ทดลองด้วยไอแอลพีโดยเพิ่มคุณลักษณะพิเศษนั้นให้ค่าเปอร์เซ็นต์ความถูกต้องสูงกว่า 88% ในทุกคำค้น ซึ่งให้ผลดีกว่าอย่างมีนัยสำคัญทางสถิติที่ระดับ .01

ตารางที่ 4.2

แสดงผลเปรียบเทียบค่าความแม่นยำ (แสดงในรูปแบบค่าเฉลี่ย±ส่วนเบี่ยงเบนมาตรฐาน)
จากการเรียนรู้ด้วยตัวอย่างโดย *, **, *** และ ****
แสดงการมีระดับนัยสำคัญที่ต่ำกว่า .01 ที่ระดับ .05 ที่ระดับ .1 และที่ระดับ .3 ตามลำดับ

คำค้น	ความถูกต้อง		
	ไอแอลพี	ไอแอลพี คุณลักษณะพิเศษ	การสมสัณฐานของกราฟย่อย
Apple	84.7±1.72**	92.0±2.46*	83.0±1.72
Jaguar	93.7±3.58*	96.8±2.99*	86.8±2.14
Mouse	87.3±3.16**	88.7±3.31*	84.2±2.26
Virus	84.7±3.82****	88.8±3.77*	83.3±2.94

ตารางที่ 4.3

แสดงกฎที่ได้หลังจากการเรียนรู้ คำค้น “jaguar” ในการเรียนรู้ชุดที่ 3 ด้วย minpos=2

ชุดที่ 3	Rule contents
Rule 1	[Pos cover = 20 Neg cover = 0] sim(A, B) :- hasword(B, C), member(D, C), haslink(D, E), member(F, C), hasword(A, G), member(H, G), haslink(H, D), connected(D, D), haslink(F, H).
Rule 2	[Pos cover = 44 Neg cover = 0] sim(A, B) :- hasword(A, C), hasword(B, D), member(E, C), member(F, D), member(E, D), member(G, C), haslink(F, H), haslink(E, G), connected(G, G).
Rule 3	[Pos cover = 17 Neg cover = 0] sim(A, B) :- hasword(B, C), hasword(A, D), member(E, C), connected(E, E), member(F, C), member(E, D), haslink(F, E), member(G, D), haslink(G, F).

ตารางที่ 4.4

แสดงกฎที่ได้หลังจากการเรียนรู้ คำค้น “jaguar” โดยเพิ่มคุณลักษณะพิเศษ ในการเรียนรู้ชุดที่ 3 ด้วย minpos=2

ชุดที่ 3	Rule contents
Rule 1	[Pos cover = 4 Neg cover = 0] sim(A, B) :- hasword(B, C), member(member, C), hasword(A, D), member(atari, D).
Rule 2	[Pos cover = 42 Neg cover = 0] sim(A, B) :- hasword(B, C), member(D, C), member(E, C), hasword(A, F), member(D, F), haslink(D, E), connected(E, E).
Rule 3	[Pos cover = 45 Neg cover = 0] sim(A, B) :- hasword(B, C), member(D, C), member(E, C), hasbigram(D, E), hasword(A, F), member(D, F), member(E, F).
Rule 4	[Pos cover = 4 Neg cover = 0] sim(A, B) :- hasword(B, C), member(sport, C), hasword(A, D), member(nfl, D).
Rule 5	[Pos cover = 4 Neg cover = 0] sim(A, B) :- hasword(B, C), member(atari, C), hasword(A, D), member(member, D).
Rule 6	[Pos cover = 6 Neg cover = 0] sim(A, B) :- hasword(B, C), hasword(A, C).
Rule 7	[Pos cover = 2 Neg cover = 0] sim(A, B) :- hasword(B, C), member(fact, C), hasword(A, D), member(animal, D).

จากการเรียนรู้ด้วยการใช้ตัวอย่างการเรียนรู้โดยการใช้ไอแอลพีด้วยคำค้น Jaguar ที่ไม่มีการเพิ่มคุณลักษณะพิเศษจะได้เซตของกฎดังตัวอย่างตารางที่ 4.3 และเมื่อนำไปทำการทดสอบกับเซตของตัวอย่างสำหรับใช้ทดสอบ พบว่า เซตของกฎนี้ให้ค่าความถูกต้อง 0.938889 หรือเซตของกฎครอบคลุมตัวอย่างบวก 5 ตัวจากทั้งหมด 10 ตัว และไม่ครอบคลุมตัวอย่างลบจากตัวอย่างลบ 10 ตัว แต่เมื่อทำการเพิ่มคุณลักษณะพิเศษทำให้พบว่า มีความถูกต้องสูงกว่า ซึ่งจะได้เซตของกฎที่ได้ดังตัวอย่างที่ 4.4 และนำไปทำการทดสอบกับเซตตัวอย่างที่ใช้ในการทดสอบนั้น พบว่า ให้ค่าความถูกต้อง 0.974074 หรือ เซตของกฎครอบคลุมตัวอย่างบวก 9 ตัว จากทั้งหมด 10 ตัว และไม่ครอบคลุมตัวอย่างลบเลย นั่นแสดงว่าเซตของกฎที่นำเสนอนั้นให้ค่าความถูกต้องจากเซตของกฎดีขึ้น ครอบคลุมตัวอย่างบวกได้มากขึ้น

บทที่ 5

สรุปผลการศึกษาและข้อเสนอแนะ

5.1 สรุปการศึกษา

การจำแนกหน้าเว็บนั้นมีความสำคัญกับหลายงานในการสืบค้นคืนสารสนเทศบนเว็บ เช่น การรวบรวมหน้าเว็บและการจัดการเว็บไดเร็กทอรี เป็นต้น โดยธรรมชาติแล้วนั้นไม่สามารถควบคุมเนื้อหาของเว็บมีเพิ่มขึ้นได้ แต่ด้วยการเติบโตที่รวดเร็วของเว็บนั้นทำให้เกิดปัญหาในการค้นหาสารสนเทศที่ไม่ตรงกับความต้องการ ทางแก้หนึ่งของปัญหานี้ คือการจัดหน้าเว็บเหล่านี้ให้เป็นกลุ่มของความสนใจ วิธีการทั่วไปที่ใช้จัดกลุ่มหน้าเว็บคือการใช้โครงสร้างของหน้าเว็บเหล่านั้น เช่น ใช้ไฮเปอร์ลิงก์ คำสำคัญ และแท็กเมตา เป็นต้น ซึ่งนำมาเป็นลักษณะสำคัญที่ใช้ในการจัดกลุ่ม

วิทยานิพนธ์นี้ได้นำเสนอวิธีการที่นำวิธีการสร้างกฎลำดับที่หนึ่งมาใช้เพื่อจัดกลุ่มเว็บไซต์ แล้วนำกฎจากโปรแกรมตรรกะเชิงอุปนัยนั้นมาใช้เพื่อจำแนกข้อมูลเว็บที่มีความคล้ายคลึงกัน โดยใช้ข้อความจากหน้าเว็บร่วมกันกับพจนานุกรมเชิงความหมายจากเวิร์ดเน็ต ซึ่งนำมาเขียนใหม่เป็นกราฟความสัมพันธ์ของคำศัพท์ที่กำหนดขึ้นโดยผู้วิจัย จากผลการทดลองจะเห็นได้ว่า วิธีที่นำเสนอนั้นสามารถสร้างกฎที่ครอบคลุม และผลการทดลองที่ได้นั้นให้ค่าความถูกต้องสูงกว่าวิธีการจัดกลุ่มเว็บไซต์โดยเทียบกับวิธีการสมสัณฐานของกราฟย่อย แต่เนื่องจากเรียนรู้นั้นใช้เวลานานในการสร้างกฎเพื่อให้ได้กฎที่ครอบคลุม และหลากหลายอาจจะต้องเพิ่มจำนวนของเว็บและคุณลักษณะพิเศษอื่นๆ รวมถึงการกำหนดคำในกราฟแนวคิดที่มีสัมพันธ์อย่างละเอียดจากผู้เชี่ยวชาญของแต่ละโดเมน เพื่อให้การเรียนรู้นั้นได้ผลที่เป็นที่น่ายอมรับและสามารถการจำแนกหน้าเว็บให้มีความถูกต้องยิ่งขึ้น

สำหรับข้อดีของวิธีที่นำเสนอนั้นคือ สามารถจำแนกหน้าเว็บที่เหมือนกันจากกราฟความสัมพันธ์ ซึ่งไอแอลพีสามารถเพิ่มคุณลักษณะความสำคัญของโหนด และสามารถแสดงลักษณะความสัมพันธ์ของกราฟโดยที่กราฟนั้นมีเพียงความสัมพันธ์กันจากโหนดสู่โหนด โดยที่มีเพียงเส้นทางจากโหนดถึงโหนด แต่ไม่จำเป็นต้องมีการเชื่อมต่อถึงกันทุกจุดภายในกราฟ

สำหรับข้อเสียของงานวิจัยนี้หน้าเว็บบางเว็บยังไม่ได้กำหนดคำสำคัญของแต่ละหน้าเว็บทำให้อาจจะยังไม่สามารถจำแนกได้ดีเท่าที่ควร

5.2 ข้อเสนอแนะ

งานวิจัยนี้พิจารณาความสัมพันธ์เชิงความหมาย แบบ Hypernym, Hyponym และ Sense key ซึ่งหากพิจารณาความสัมพันธ์แบบอื่นๆ เช่น Synonym, Antonym, Meronym, Holonym และ Troponym เพิ่มด้วยจะทำให้สามารถจำแนกหน้าเว็บได้ครอบคลุมมากขึ้น และส่วนกราฟแสดงโครงสร้างความสัมพันธ์หากนำความสัมพันธ์ของคำศัพท์จากพจนานุกรมเชิงความหมายเวิร์ดเน็ตมาสร้างเป็นกราฟความสัมพันธ์อาจจะทำให้ได้ข้อมูลโครงสร้างกราฟที่สมบูรณ์ได้ ผู้วิจัยคาดหวังว่าอาจจะสามารถข้อมูลจากเอกสาร RDF และ OWL เพื่อใช้เป็นข้อมูลในการค้นหาเว็บแทนการกำหนดในแท็กเมตาคำหลัก เนื่องจากเอกสารเหล่านี้ถูกออกแบบมาสำหรับเว็บเชิงความหมาย แต่ปัจจุบันยังไม่นิยมนำมาใช้กันอย่างแพร่หลาย

ภาคผนวก

ภาคผนวก ก.

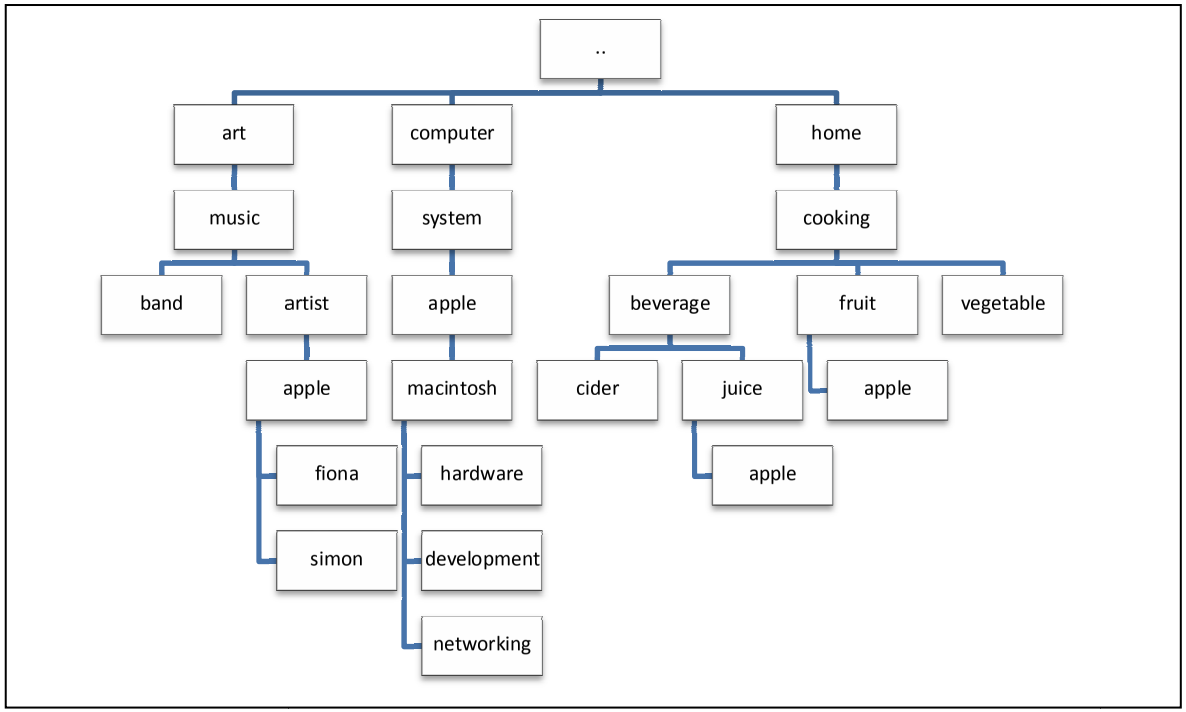
ตัวอย่างชุดข้อมูลเรียนรู้และกราฟแสดงความสัมพันธ์คำศัพท์แนวคิด

ตารางที่ ก.1
แสดงคำค้นจำแนกโดเมนที่ใช้ในการทดลอง

คำค้น	โดเมน
Apple	• Art/Music • Computer/Company • Computer/Hardware • Home/Cooking/Fruit • Home/Cooking/Beverage/Juice
Jaguar	• Arts/Music • Game/Video /Console • Home/Consumer /Automobiles/Purchasing • Science/Biology/Animal/Mammal • Sport/Football/American/NFL
Mouse	• Arts/Music • Kid/Teen/Entertainment/Animation/Cartoon • Computer/Software/Shareware/Windows/Utility • Computer/Programming/Language • Science/Biology/Genetics/Eukaryotic/Animal/Mammal/Rodent
Virus	• Computer/Security/Malicious • Computer/Software/Shareware/Windows/Security/Anti-Virus • Health/Conditions/Diseases/Infectious/Diseases/Viral/West/Nile • Health/Pharmacy/Drugs/Medication/I/Influenza/Vaccine • Society/Folklore/Literature/Urban/Legend/Computer/Hoax

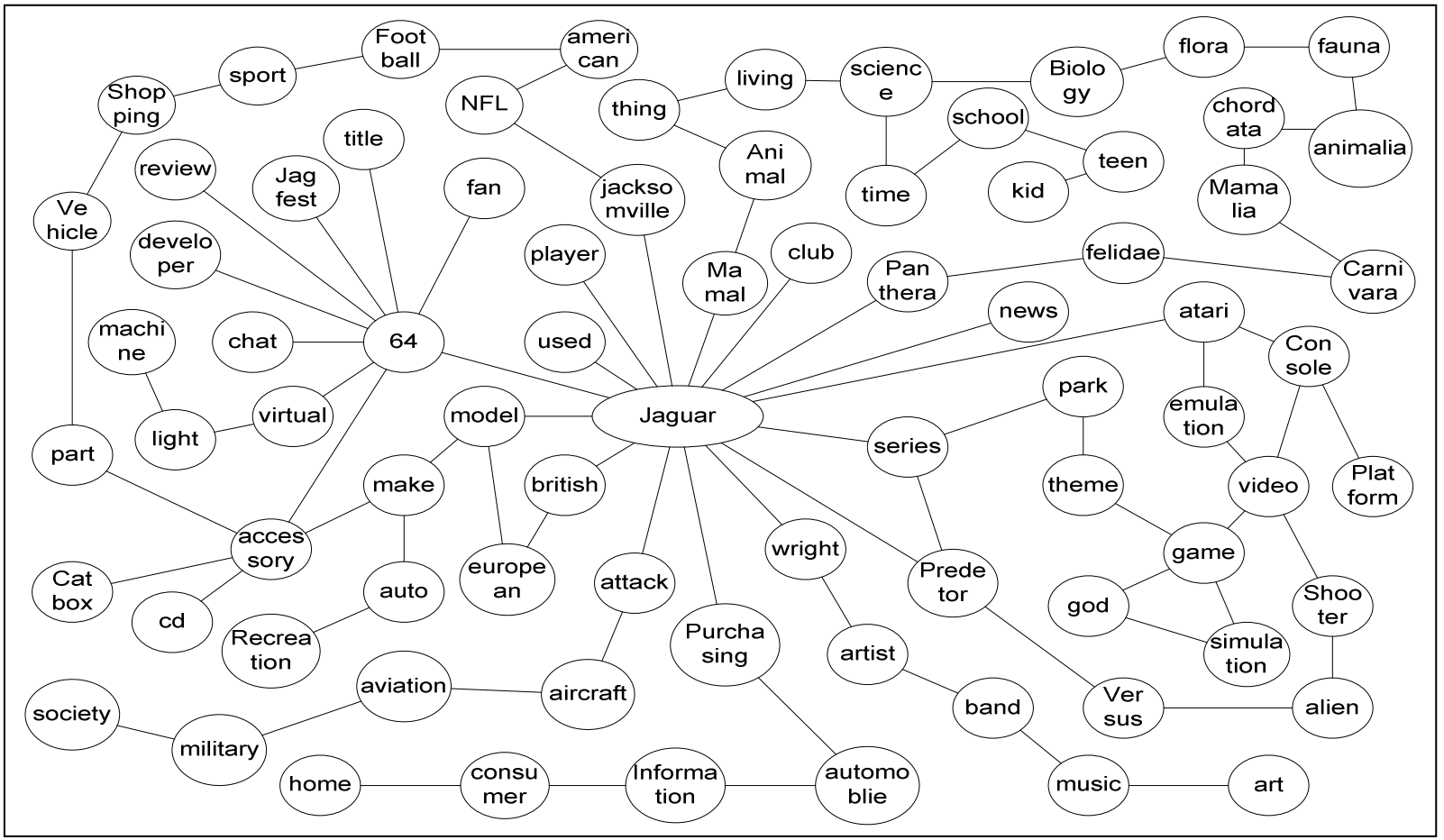
ภาพที่ ก.1

แสดงความสัมพันธ์ของคำค้น Apple



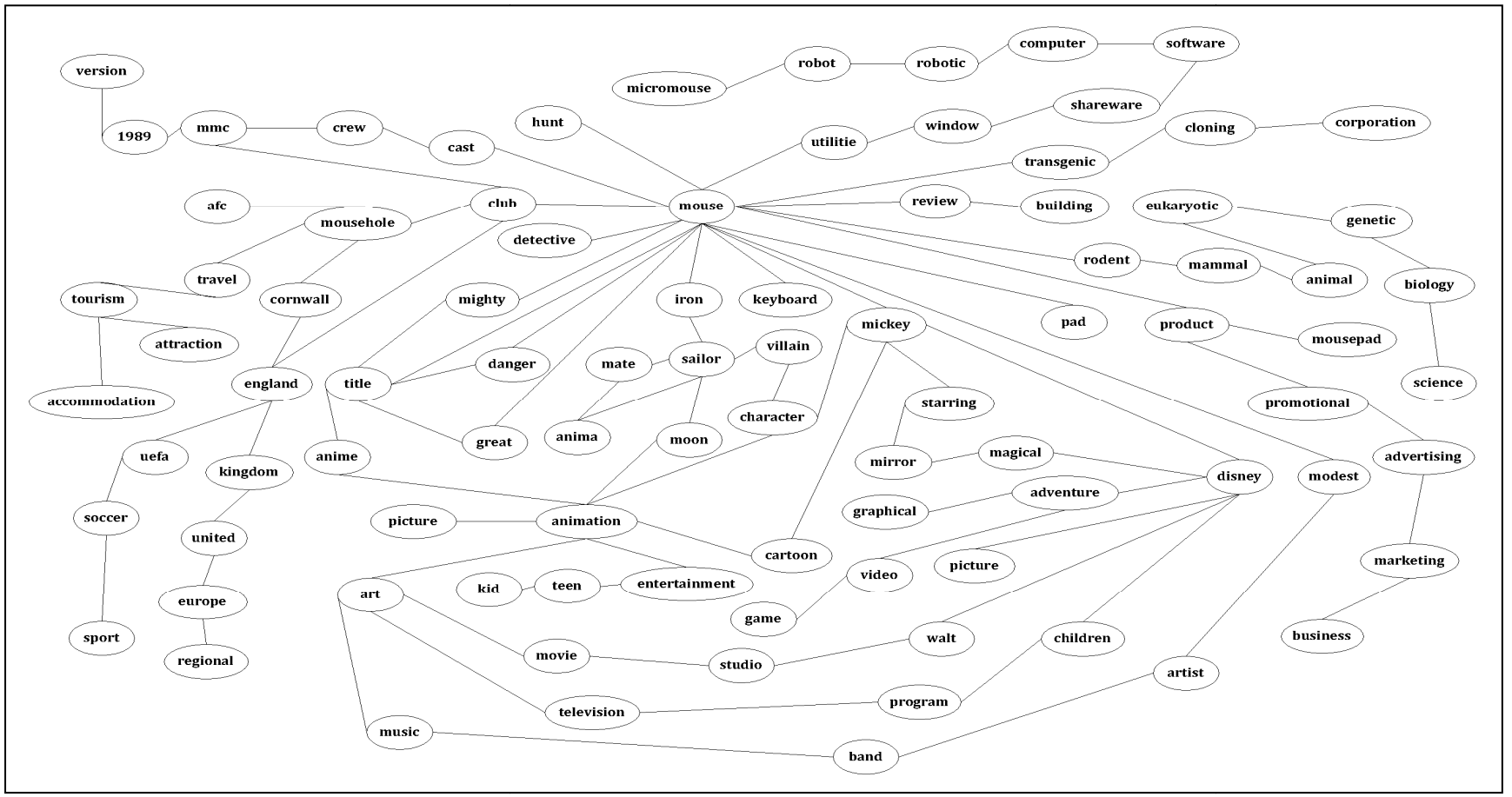
ภาพที่ ก.2

แสดงความสัมพันธ์ของคำค้น Jaguar



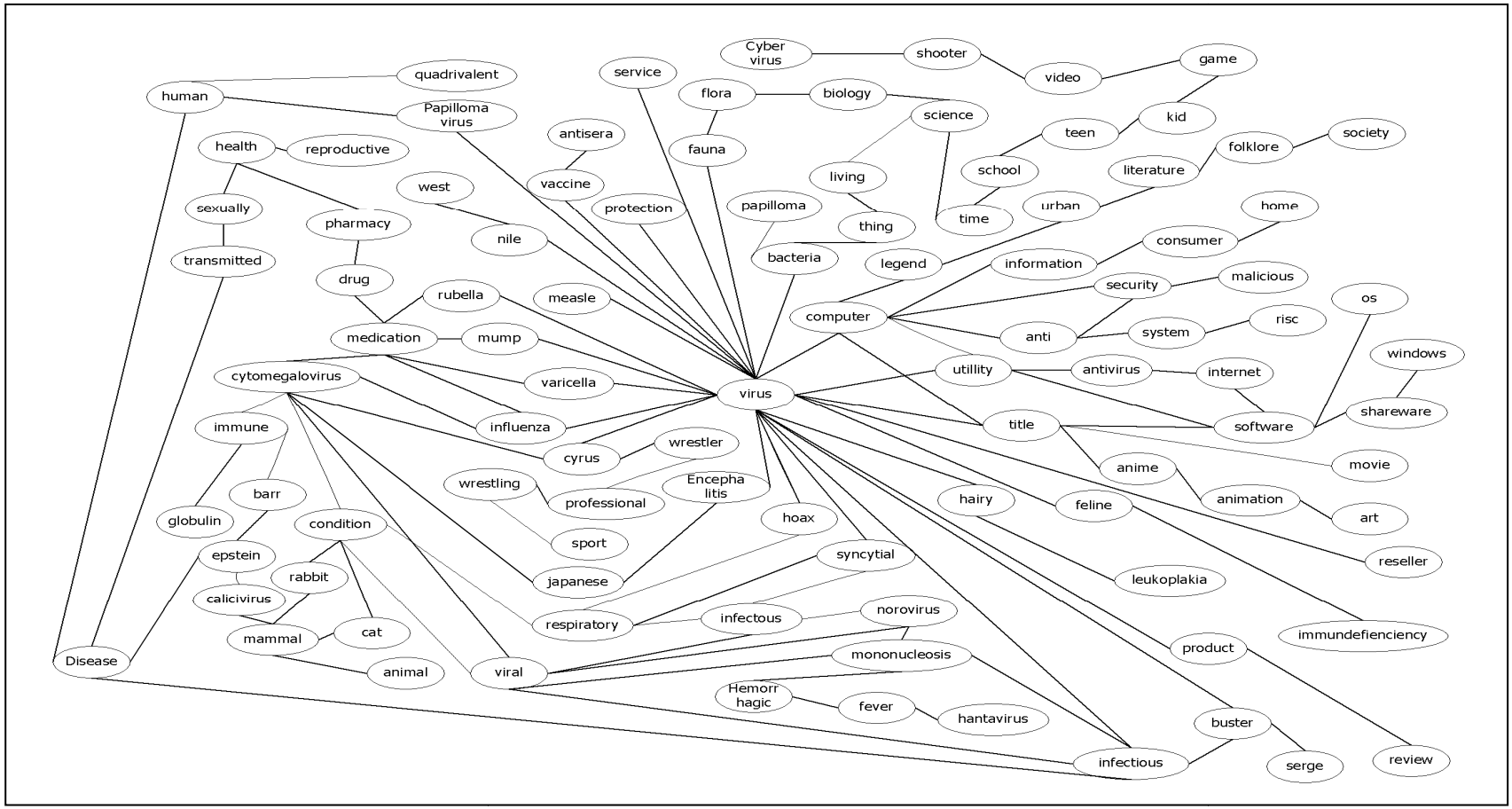
ภาพที่ ก.3

แสดงความสัมพันธ์ของคำค้น Mouse



ภาพที่ ก.4

แสดงความสัมพันธ์ของคำค้น Virus



ภาคผนวก ข.

การใช้เครื่องมือในการสร้างกฎด้วยวิธีการโปรแกรมตรรกะเชิงอุปนัย ด้วยโปรแกรม Aleph

วิธีการโปรแกรมตรรกะเชิงอุปนัยเป็นวิธีการเรียนรู้ของเครื่องแบบหนึ่งโดยใช้กฎลำดับที่หนึ่งในการอธิบายตัวอย่างกฎที่ได้จากการเรียนรู้และใช้ความรู้ภูมิหลัง ชุดข้อมูลตัวอย่างบวก สำหรับการสร้างกฎที่ครอบคลุมตัวอย่างบวก และไม่ครอบคลุมตัวอย่างลบ

Aleph หรือ A Learning Engine for Proposing hypothesis เป็นเครื่องมือที่ใช้ในการสร้างกฎด้วยวิธีการโปรแกรมตรรกะเชิงอุปนัยที่ถูกเขียนขึ้นโดย Ashwin Srinivasan และ Rui Camacho ในปี 1993 โดยจุดมุ่งหมายเพื่อใช้ศึกษาวิธีการเรียนรู้ด้วยวิธีการโปรแกรมตรรกะเชิงอุปนัย ซึ่งสามารถดาวน์โหลดได้จากเว็บไซต์

http://www.comlab.ox.ac.uk/oucl/research/areas/machlearn/Aleph/aleph.pl

และผู้พัฒนาได้ทำคู่มือการใช้งานและหลักการทำงาน Aleph สามารถค้นอ่านเพิ่มเติมจาก

http://web2.comlab.ox.ac.uk/oucl/research/areas/machlearn/Aleph/aleph.html

เนื่องจาก Aleph นั้นถูกพัฒนาขึ้นด้วยภาษาโปรล็อกจำเป็นต้องใช้โปรล็อกคอมไพล์เลอร์ สำหรับในงานวิจัยนี้ใช้ โปรแกรม SWI Prolog สามารถดาวน์โหลดโปรแกรมและวิธีการติดตั้งรวมทั้งคู่มือการใช้งานได้จาก www.swi-prolog.org

ด้วยงานวิจัยนี้ใช้ Aleph เวอร์ชัน 5 เป็นเครื่องมือในการสร้างกฎด้วยวิธีการโปรแกรมตรรกะเชิงอุปนัย ตัวอย่างที่เป็นอินพุตของ Aleph และความรู้ภูมิหลังสำหรับเซตตัวอย่างเพื่อให้ Aleph ใช้ในการเรียนรู้เพื่อสร้างกฎ โดยข้อมูลอินพุตที่ Aleph รับเข้าไปนั้นจะอยู่ในรูปแบบของไฟล์ข้อมูล 3 ไฟล์ ประกอบด้วย

1. ไฟล์ข้อมูลความรู้ภูมิหลัง โดยที่นามสกุลของไฟล์ คือ .b
2. ไฟล์ข้อมูลตัวอย่างบวก โดยมีนามสกุลของไฟล์ คือ .f
3. ไฟล์ข้อมูลตัวอย่างลบ โดยมีนามสกุลของไฟล์ คือ .n

และชื่อไฟล์ของข้อมูลทั้ง 3 ไฟล์จะต้องเป็นชื่อเดียวกันเช่น train.f สำหรับไฟล์ข้อมูลตัวอย่างบวก, train.n สำหรับไฟล์ข้อมูลตัวอย่างลบ และ train.b สำหรับไฟล์ข้อมูลความรู้ภูมิหลัง เป็นต้น

วิธีการใช้งานโปรแกรม Aleph เพื่อสร้างกฎ และ ทดสอบกฎ เป็นดังนี้

1. คำสั่ง ***read_all***(ชื่อไฟล์)**.** เป็นการอ่านไฟล์ข้อมูลเหล่านี้ทั้งหมด

2. คำสั่ง ***induce.*** เป็นคำสั่งที่ให้ Aleph เริ่มสร้างกฎ

3. คำสั่ง ***write_rules***(ชื่อไฟล์). เป็นการบันทึกกฎที่ได้จากการเรียนรู้ซึ่งผลที่บันทึกลงไฟล์จะแสดงกฎที่ได้ พร้มทั้งจำนวนตัวอย่างที่ครอบคลุม และค่าความถูกต้องที่ได้ของกฎ

บรรณานุกรม

หนังสือ

บุญเสริม กิจศิริกุล. ปัญญาประดิษฐ์ (Artificial Intelligent). สำนักพิมพ์จุฬาลงกรณ์มหาวิทยาลัย, 2546.

Weiss, S.M. & Kulikowski, Computer Systems That Learn, Morgan Kaufmann, C.A., 1991.

Nils J, Nilsson. Introduction to Machine Learning. Robotic Laboratory Department of Computer Science Standford University, 1996.

Christiane, F. (editor), WordNet: An Electronic Lexical Database. Rider University and Princeton University Cambridge, MA: The MIT Press, 1998

คู่มือการใช้งาน

Aaron Swartz. The Semantic Web in Breadth. http://logicerror.com/semanticWeb-long

Brian McBride. RDF Primer. W3C Recommendation, 2004. http://www.w3.org/TR/rdf-primer

Deborah L. McGuinness and Frank van Harmelen eds. OWL Web Ontology Language Overview. W3C Recommendation, 2004. http://www.w3.org/TR/owl-features/

Ashwin Srinivasan. The Aleph Manual Version 4 and above. 2007

งานวิจัย

สุกรี สินธุภิญโญ. “การโปรแกรมตรรกะเชิงอุปนัยและแบ็กพรอพาเกชันนิวรอลเน็ตเวิร์กในการรู้จำตัวพิมพ์อักษรไทย.” วิทยานิพนธ์วิศวกรรมศาตรมหาบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2541.

สุกรี สินธุภิญโญ. ”การประมาณกฎจากวิธีการโปรแกรมตรรกะเชิงอุปนัยด้วยวิธีการแบ็กพรอพาเกชันนิวรอลเน็ตเวิร์กในการรู้จำตัวพิมพ์อักษรไทย”. วิทยานิพนธ์วิศวกรรมศาตรดุษฎีบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์

มหาวิทยาลัย, 2544.

บังอร กลับบ้านเกาะ. “การค้นคืนสารสนเทศออนไลน์โดยใช้จีเนติกอัลกอริธึม”. สาขาวิชาเทคโนโลยีสารสนเทศ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2543.

ภูวดล เรืองกิจภิญโญ. “การจำแนกผลการค้นหาบนเว็บเชิงความหมายตามความพอใจของผู้ใช้”. สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์, 2549.

Arul Prakas Asirvatham, Kranthi Kumur. Ravi. “Web Page Classification based on Document Structure”, 2001

R. Guha, Rob McCool and Eric Miller. “Semantic Search”. Paper presented at the International World Wide Web Conferendes. USA, 2003.

Li Ding, Tim Finin, Anupam Joshi, Yun Peng, R. Scott Cost and Joel Sachs. “Swoogle: a search and metadata engine for the semantic web”, 2004

Zhenjiang Lin. “PageSim: A Novel Link-based Measure of Web Page Similarity”. Edinburgh Scotland. 2006

Raymond J. Mooney, Prem Melville, Lappoon Rupert Tang, Jude Shavlik, Ines de Castro Dutra, David Page and Vitor Santos Costa. “Relational Data Mining with Inductive Logic Programming for Link Discovery”, 2002

Monika Zakova, Petr Kremen, Javier A. Garcia-Sedano, Cyril Masia Tissot, Nada Lavrac, Filip Zelezny and Javier Molina. “Relational Data Mining Applied to Virtual Engineering of Product Designs”, 2006

Muggleton, S.H. ed., “Inductive logic programming”, New York NY: Academic Press, 1992

M. Young, “The Technical Writer’s Handbook,” Mill Valley, CA: University Science, 1989

Xiaoguang Qi, and Brain D. Davison, “Web page classification: Features and Algorithms”, 2007

Zhenjiang Lin, “PageSim: A Novel Link-based Measure of Web Page Similarity”, Edinburgh Scotland. 2006

Dou Shen and Rong Pan, "Query enrichment for web-query classification", New York 2006

Olivier Corby, Rose Dieng, and Cedric Hebert, "A conceptual graph model for W3C resource description framework", France

Hua-Jun Zeng, Qi-Cai He, Zheng Chen, Wei-Ying Ma, Jinwen Ma, "Learning to cluster web search results", UK, 2004

Monika Zakova, Petr Kremen, Javier A. Garcia-Sedano, Cyril Masia Tissot, Nada Lavrac, Filip Zelezny and Javier Molina, "Relational Data Mining Applied to Virtual Engineering of Product Designs", 2006

J. F. Sowa, "Conceptual Structures, Information Processing in Mind and Machine", Addison-Wesley, 1984

J. Sowa, "Knowledge Representation, Logical, Philosophical, and Computational Foundations", Brooks Cole Publishing Co., 2000

Adam Schenker, Mark Last Horst Bunke, and Abraham Kandel, "Classification of Web Document Using a Graph Model", 2003

Sofia Stamou, Alexandros Ntoulas, Vlassis Krikos, Pavlos Kokosis, and Dimitris Christodoulakis, "Classifying Web Data in Directory Structures", 2006

Google: web search engine. (http://www.google.com)

YaHoo: web search engine. (http://www.yahoo.com)

MSN: web search engine. (http://www.msn.com)

DMOZ: open directory project. (http://dmoz.org)

WordNet Web site: http://www.cogsci.princeton.edu/~wn/

RDF: Resource Description Framework (http://www.w3.org/RDF)

OWL: Web Ontology Language (http://www.w3.org/2004/OWL)

SWI-Prolog: (http://www.swi-prolog.org)