สารบัญเนื้อหา

เรื่อง **หน้า**

บทที่ 1 บทนำ 1
1.1 ความเป็นมาหลักการและเหตุผล 1
1.2 วัตถุประสงค์หลักของแผนงานวิจัย 2
1.3 ประโยชน์ที่คาดว่าจะได้รับ 3
บทที่ 2 การทบทวนวรรณกรรมและทฤษฎีที่เกี่ยวข้อง 4
2.1 กระบวนการค้นคืนรูปภาพ 4
2.1.1 คุณลักษณะแบบโกบอล (Global features) 6
2.1.2 คุณลักษณะแบบโลคอล (Local features) 7
2.1.3 คุณลักษณะซิฟท์ (SIFT Descriptors) 8
2.2 การทำดัชนีข้อมูลรูปภาพ 9
2.3 การวัดความคล้ายคลึงของรูปภาพ 11
2.4 การค้นคืนสารสนเทศเชิงความหมาย 12
2.5 วิธีการทำดัชนีสำหรับระบบค้นคืนสารสนเทศเชิงความหมาย 13
2.6 Latent Semantic Indexing (LSI) 13
2.7 การทำดัชนีโดยใช้ออนโทโลจี 20
2.8 การออกแบบออนโทโลจี 21
2.9 เครื่องมือที่ใช้ในการออกแบบออนโทโลจี 21
2.10 วิธีการนำเสนอองค์ความรู้สำหรับระบบค้นคืนสารสนเทศเชิงความหมาย 22
2.10.1 RDF 22
2.10.2 RDFS (Schema) 24
2.10.3 OWL 26
2.11 วิธีการค้นหาข้อมูลจากฐานความรู้ 28
2.11.1 RDQL 29
2.11.2 SPARQL 30

สารบัญเนื้อหา (ต่อ)

เรื่อง **หน้า**

บทที่ 3 ระเบียบวิธีวิจัย 33
3.1 การออกแบบการวิจัยและกรอบแนวคิด 33
3.1.1 การวิเคราะห์และการดึงคุณลักษณะแบบโลคอลของรูปภาพ 34
3.1.1.1 การสกัดคุณลักษณะระดับต่อของรูปภาพ 35
3.1.1.2 การสร้างวิชวลเวิร์ด 35
3.1.1.3 การแปลงโครงสร้างของวิชวลเวิร์ด 36
3.1.2 การวิเคราะห์และสกัดสารสนเทศจากคำอธิบายรูปภาพ 38
3.1.3 การออกแบบองค์ความรู้ 39
3.2 ความสมบูรณ์ของออนโทโลจี 40
3.3 ขอบเขตของการวิจัย 41
บทที่ 4 ผลการวิจัยและการอภิปรายผล 42
4.1 สมมติฐานการวิจัย 42
4.2 การประเมินประสิทธิภาพของระบบ 43
4.3 การทดลองที่ 1: การทดลองเพื่อทดสอบการใช้คิวรี่ในรูปแบบข้อความและรูปภาพ 44
4.4 การทดลองที่ 2: การทดลองเพื่อประเมินประสิทธิภาพการค้นหาข้อมูลเชิงความหมาย . 46
4.5 การทดลองที่ 3: การทดลองเพื่อประเมินความไม่สมบูรณ์ของออนโทโลจี 47
4.6 การทดลองที่ 4: การทดลองเพื่อเปรียบเทียบประสิทธิภาพการค้นหากับระบบค้นหาข้อมูลที่มีอยู่ในปัจจุบัน 49
บทที่ 5 สรุปผลการวิจัย 52
5.1 สรุปแนวคิดในงานวิจัย 52
5.2 การเผยแพร่งานวิจัย 53
5.3 ปัญหาและอุปสรรค 53
5.4 แนวทางการพัฒนาในอนาคต 53
บทที่ 6 บรรณานุกรม 55
ภาคผนวก ก ประวัติผู้วิจัย 59
ภาคผนวก ข ผลงานการตีพิมพ์ 62

สารบัญรูป

รูปที่ **หน้า**

รูปที่ 1 การค้นหารูปภาพโดยใช้คอนเทนท์ (Content-based Image Retrieval) 7
รูปที่ 2 ตัวอย่างรูปภาพที่ถูกประมวลผลเพื่อหาจุดสำคัญต่างๆ บนภาพ..... 9
รูปที่ 3 เปรียบเทียบความเหมือนกันระหว่างเอกสารต่างๆ..... 11
รูปที่ 4 ตัวอย่างเมทริกซ์แสดงความสัมพันธ์ระหว่างคีย์เวิร์ดและเอกสารต่างๆ..... 14
รูปที่ 5 ผลลัพธ์การคำนวณหาน้ำหนักแบบโลคอลของคีย์เวิร์ดทั้งหมดในเมทริกซ์..... 16
รูปที่ 6 ผลลัพธ์การคำนวณหาน้ำหนักแบบโกบอลของคีย์เวิร์ดทั้งหมดในเมทริกซ์ 16
รูปที่ 7 เมทริกซ์ผลลัพธ์จากการทำนอร์มอไลเซชั่น 17
รูปที่ 8 เมทริกซ์ผลลัพธ์หลังจากคำนวณน้ำหนักของคีย์เวิร์ด 18
รูปที่ 9 เมทริกซ์ผลลัพธ์ *U S* และ *V* จากการคำนวณ SVD 19
รูปที่ 10 ตัวอย่าง Statement ในโมเดล RDF ซึ่งประกอบด้วย Resources และ Property 23
รูปที่ 11 ตัวอย่าง O-A-V statement (Triplets)..... 23
รูปที่ 12 ตัวอย่าง Statement ที่มี Object เป็นอีก Statement หนึ่ง..... 24
รูปที่ 13 ตัวอย่างไฟล์ประเภท RDFS..... 26
รูปที่ 14 ตัวอย่างไฟล์ประเภท OWL 28
รูปที่ 15 รูปแบบคำสั่งคิวรี่ของ RDQL 29
รูปที่ 16 รูปแบบคำสั่งคิวรี่ของ SPARQL..... 30
รูปที่ 17 ตัวอย่าง SPARQL เพื่อหาคนที่มีอายุมากกว่า 30 ปี ในออนโทโลจี 31
รูปที่ 18 ตัวอย่างผลลัพธ์การค้นหาข้อมูลโดย SPARQL 31
รูปที่ 19 กรอบแนวคิดงานวิจัย..... 33
รูปที่ 20 กระบวนการในการแปลงข้อมูลจากฐานข้อมูลให้อยู่ในรูปแบบออนโทโลจี..... 38
รูปที่ 21 โครงสร้างของออนโทโลจีในงานวิจัย 40
รูปที่ 22 เปรียบเทียบค่าพรีซิชั่นเฉลี่ยของคิวรี่รูปภาพ 45
รูปที่ 23 เปรียบเทียบค่าพรีซิชั่นเฉลี่ยของคิวรี่ข้อความ..... 46
รูปที่ 24 เปรียบเทียบประสิทธิภาพการค้นคืนสารสนเทศเชิงความหมายสำหรับ MMIO, LSI, และ Lucene 47

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

รูปที่ 25 ผลการค้นหารูปกีฬาชกมวยโดยใช้ LSI 48
รูปที่ 26 ผลลัพธ์การค้นหารูปภาพนักมวย Vijender Singh ที่กำลังชกมวยในกีฬาโอลิมปิค London 2012 50
รูปที่ 27 ผลลัพธ์การค้นหารูปภาพโดยใช้คิวรี่เป็นรูปภาพตัวอย่าง (Q) ของ Google และ MMIO ... 51

บทที่ 1
บทนำ

1.1 ความเป็นมาหลักการและเหตุผล

ปัจจุบันการค้นคืนข้อมูลประเภทรูปภาพนั้นสามารถทำได้โดยใช้คุณลักษณะระดับต่ำ (low-level Features) ของรูปภาพ ซึ่งคุณลักษณะระดับต่ำนี้สามารถแบ่งได้อีก 2 ประเภทย่อย 1) คุณลักษณะแบบโกบอล (Global features) ซึ่งจะเป็นคุณลักษณะแบบรวมๆ ไม่เฉพาะเจาะจง ได้แก่ สี (Color) รูปทรง (Shape) และพื้นผิว (Texture) และ 2) คุณลักษณะแบบโลคอล (Local features) ซึ่งจะเป็นคุณลักษณะเฉพาะของแต่ละส่วนในรูปภาพ ประโยชน์ของการใช้คุณลักษณะแบบโกบอลคือ มีความง่าย ไม่ซับซ้อนมากและมีความเร็วในการประมวลผลเพื่อดึงคุณลักษณะต่างๆ ของรูปภาพออกมา แต่ว่าคุณลักษณะแบบโกบอลนี้ก็มีข้อจำกัดอยู่คือ ความน่าเชื่อถือในการตรวจหาวัตถุในรูปภาพ (Object Recognition) เมื่อรูปภาพมีขนาด ความสว่าง หรือมุมของกล้องแตกต่างกันออกไปมีผลทำให้การตรวจหาวัตถุในรูปภาพมีประสิทธิภาพต่ำลง ดังนั้นคุณลักษณะแบบโลคอลจึงได้รับความสนใจจากนักวิจัย และถูกนำมาใช้ในการประมวลผลสำหรับระบบค้นคืนรูปภาพแทนคุณลักษณะแบบโกบอล

ทั้งนี้เนื่องจากคุณสมบัติแบบโกบอล มีปัญหาในการค้นคืนรูปภาพที่มีลักษณะต่างกันในเรื่องต่างๆ เช่น ขนาดของภาพที่ไม่เท่ากัน การหมุนภาพและมุมกล้องที่ไม่เหมือนกัน ตลอดจนความสว่างในรูปภาพที่ไม่เท่ากัน ปัจจัยที่แตกต่างกันเหล่านี้เป็นอุปสรรคในการเปรียบเทียบเพื่อหารูปภาพที่คล้ายคลึงกัน คุณลักษณะแบบโลคอลมีประโยชน์หลายประการมากกว่าคุณลักษณะแบบโกบอล โดยคุณลักษณะแบบโลคอลนี้จะเน้นไปที่การหาจุดสำคัญบนรูปภาพ (keypoints) สามารถหาได้โดยใช้เทคนิคการประมวลผลภาพทางคอมพิวเตอร์ต่างๆ อาทิเช่น Harris corner (Harris and Stephens, 1988) และ Difference of Gaussian (DoG) (1999) ในปี 2004 ได้มีนักวิจัยชื่อ Lowe ได้ค้นพบคุณลักษณะของรูปภาพแบบใหม่เรียกว่า Scale Invariant Feature Transform (SIFT) (Lowe, 2004) ซึ่งสามารถลดปัญหาของคุณลักษณะแบบโกบอลที่กล่าวมาแล้วเบื้องต้นได้อย่างมีประสิทธิภาพ แต่อย่างไรก็ตาม SIFT ยังรองรับได้แค่การเปลี่ยนมุมกล้องแบบ 2D หากเปลี่ยนมุมกล้องในลักษณะ 3D คุณลักษณะ SIFT ก็ยังไม่สามารถทำงานได้อย่างมีประสิทธิภาพมากนัก

ในเวลาต่อมาคุณลักษณะ SIFT ได้ถูกนำมาประมวลผลเพื่อสร้างเป็นชุดของข้อมูล เรียกว่า ถุงของคำ Bag of Visual Words (BVW) ซึ่งเป็นเทคนิคที่พัฒนามาจากเทคนิคการประมวลผลเอกสารที่เป็นข้อความ ซึ่งเทคนิคนี้จะรวมคำสำคัญต่างๆ ที่เป็นตัวแทนเอกสารต่างๆ ไว้เป็นชุดๆ หรือเป็นถุงๆ นั่นเอง และต่อมาเทคนิค BVW จึงเป็นที่นิยมกันอย่างแพร่เหลายในหมู่นักวิจัยที่เกี่ยวกับการประมวลผลรูปภาพ ตัวอย่างเช่น นำไปใช้ในการจำแนกรูปภาพ (classification) (Tirilly et al., 2008) นำไปใช้ในการสร้างคำอธิบายรูปภาพ (annotation) (Wu et al., 2009) หรือนำไปช่วยในการค้นหารูปภาพที่คล้ายคลึง (Zheng et al., 2008) อย่างไรก็ตามเทคนิคถุงของคำนี้มีลักษณะโครงสร้างเป็นเว็กเตอร์ (Vector) จึงไม่สามารถอธิบายรูปภาพในระดับที่มนุษย์เข้าใจได้ ซึ่งโดยปกติแล้วมนุษย์จะเข้าใจความหมายของสิ่งต่างๆ ได้นั้น จะใช้หลักการสร้างความสัมพันธ์ของสิ่งต่างๆ ที่เกิดขึ้นจากประสบการณ์ทีผ่านมา ความสัมพันธ์ต่างๆ เหล่านี้หากนำมาวาดเป็นไดอะแกรม จะพบว่าจะมีโครงสร้างที่เป็นลำดับชั้นต่างๆ เชื่อมโยงกันอย่างเป็นระบบ ดังนั้นแนวคิดในงานวิจัยนี้จึงพยายามแปลงโครงสร้างของ BVW จากรูปแบบของเว็กเตอร์ให้อยู่ในรูปแบบโครงสร้างลำดับชั้นแทน ซึ่งปัจจุบันนิยมเรียกว่า “ออนโทโลจี (Ontology)”

ในการแปลงโครงสร้างของ BVW นี้ ผู้วิจัยพบว่ามีความท้าทายหลายประการ ประการที่หนึ่ง ในการค้นหาจุดสำคัญบนรูปภาพนั้น จะพบจุดที่ไม่สำคัญเกิดขึ้นมากมาย ดังนั้นจึงต้องมีการกรองเอาจุดที่ไม่สำคัญเหล่านั้นทิ้งไปก่อน เพื่อเพิ่มประสิทธิภาพของ BVW ให้ดีขึ้นและสามารถใช้เป็นตัวแทนเนื้อหาหรือความหมายของรูปภาพได้ดีขึ้นนั่นเอง ประการที่สอง BVW ที่ได้นี้บางครั้งมีความกำกวม หมายถึงว่า BVW สามารถใช้เป็นตัวแทนรูปภาพได้มากกว่า 1 รูปภาพ หรือมีความหมายมากกว่า 1 ความหมายนั่นเอง ดังนั้นผู้วิจัยคิดว่าควรพัฒนาวิธีเพื่อหาความหมายที่แท้จริงของ BVW ให้ได้ก่อนจะนำไปใช้ จึงจะทำให้ระบบค้นคืนรูปภาพทำงานได้อย่างมีประสิทธิภาพมากขึ้น ดังนั้นงานวิจัยนี้จึงใช้ปัญหาดังกล่าวเป็นโจทย์ในการทำวิจัย เพื่อพัฒนาวิธีการใหม่ในการใช้คุณลักษณะระดับต่ำเป็นตัวแทนของรูปภาพให้มีประสิทธิภาพดียิ่งขึ้น

1.2 วัตถุประสงค์หลักของแผนงานวิจัย

1) ปรับปรุงโครงสร้างของวิชวลเวิร์ดเพื่อใช้แทนความหมายของรูปภาพ
2) เพื่อเพิ่มประสิทธิภาพการจำแนกรูปภาพและการค้นคืนรูปภาพ
3) ลดข้อจำกัดความไม่สมบูรณ์ของออนโทโลยีอันเนื่องมาจากการออกแบบออนโทโลยีที่ไม่ดีพอ

1.3 ประโยชน์ที่คาดว่าจะได้รับ

1) ได้กรอบแนวคิดที่จะพิจารณาความคล้ายคลึงของรูปภาพต่างๆ โดยพิจารณาจากความหมายของรูปภาพเป็นหลัก แทนการพิจารณาจากคุณลักษณะระดับต่ำ (Low level features) เหมือนวิธีการเดิมๆ ที่มีอยู่
2) ได้วิธีการที่จะค้นหาวิชวลเวิร์ดที่ไร้ประโยชน์ เพื่อเพิ่มประสิทธิภาพให้กับระบบการจำแนกรูปภาพ ทำให้มีความถูกต้องในการแบ่งกลุ่มมากยิ่งขึ้น และรวดเร็ว
3) ได้แนวคิดวิธีการในการตีความหมายที่อยู่ในรูปภาพและใช้เป็นต้นแบบในการพัฒนาระบบค้นคืนข้อมูลรูปภาพ (Image Retrieval System) โดยพิจารณาจากความหมายของคิวรี่และความหมายที่อยู่ในรูปภาพเป็นหลักหรือเรียกว่า “semantic search”

บทที่ 2
การทบทวนวรรณกรรมและทฤษฎที่เกี่ยวข้อง

2.1 กระบวนการค้นคืนรูปภาพ

ในระบบ CBIR รูปภาพจะถูกค้นหาและเปรียบเทียบความคล้ายคลึงโดยดูจากความคล้ายคลึงของคุณลักษณะรูปภาพเป็นหลัก ดังนั้นระบบจึงต้องทำการดัชนีโดยใช้คุณลักษณะเหล่านี้เช่นกัน ในระบบการค้นคืนรูปภาพมีส่วนประกอบหลักอยู่ 3 ส่วน คือ

1) ส่วนการประมวลผลเพื่อดึงเอาคุณลักษณะของรูปภาพออกมา (Visual features extraction)
2) ส่วนการทำดัชนีจากคุณลักษณะของรูปภาพดังกล่าว (Multidimensional indexing)
3) ส่วนการค้นคืนข้อมูลโดยใช้คอนเทนท์ (Content-based Retrieval)

ในส่วนที่หนึ่งนั้นคือเทคนิคการประมวลผลรูปภาพ (Image processing) เพื่อทำการดึงเอาคุณลักษณะที่ต้องการออกมาเพื่อทำดัชนี คุณลักษณะเหล่านี้จะแสดงอยู่ในรูปแบบของเว็กเตอร์หลายมิติ (Multi-dimensional vector) ตัวอย่างเช่น มิติที่หนึ่งคือสี มิติที่สองคือรูปทรง มิติที่สามคือพื้นผิว เป็นต้น คุณลักษณะภาพเหล่านี้จะถูกเก็บไว้ในลักษณะของเมต้าเดต้าของรูปภาพ และใช้เพื่อทำดัชนีในลักษณะของการทำดัชนีแบบหลายมิติ ดังนั้นการค้นหารูปภาพที่เหมือนกับคิวรี่จะทำการเปรียบเทียบความคล้ายคลึงของคิวรี่และคุณลักษณะของรูปภาพต่างๆ บนพื้นที่เว็กเตอร์ในทุกมิติ คุณลักษณะของรูปภาพใดที่มีค่าความเหมือนกับคุณลักษณะของรูปภาพคิวรี่เกินค่าๆ หนึ่ง (Threshold) จะถูกดึงออกมาแสดงต่อผู้ใช้หรืออธิบายได้อีกอย่างว่าการหาความคล้ายคลึงกันระหว่างรูปภาพคิวรี่และรูปภาพอื่นๆ ที่อยู่ในคอลเล็คชั่นนั้นดูจากระยะห่างของคุณลักษณะของรูปภาพคิวรี่และรูปภาพในคอลเล็คชั่น ซึ่งต้องมีระยะห่างไม่เกินค่าๆ หนึ่ง (ค่านี้ได้จากการทดลอง) วิธีการทำดัชนีแบบนี้เรียกว่า MAS (Multi-dimensional access structure) ในหัวข้อถัดไปจะทำการศึกษาถึงคุณลักษณะต่างๆ ของรูปภาพ และวิธีกาทำดัชนีด้วยคุณลักษณะดังกล่าว

สำหรับการค้นคืนข้อมูลรูปภาพนั้นในยุคแรกๆ จะใช้เมต้าเดต้า (metadata) ในการค้นหารูปภาพ เช่น ชื่อไฟล์รูปภาพหรือคำอธิบายภาพ (Text caption) นำไปเปรียบเทียบกับคำในคิวรี่ ถ้าคำในคิวรี่และชื่อไฟล์รูปภาพตรงกัน (String matching) รูปภาพนั้นจะถูกดึงมาเป็นผลลัพธ์ แต่ปัญหามีอยู่ว่าบางครั้งคนตั้งชื่อไฟล์ไม่ตรงกับเนื้อหาของรูปภาพ เช่น ชื่อไฟล์เป็นตัวเลข “1.JPG” ซึ่งทำให้

รูปภาพเหล่านี้ไม่สามารถถูกค้นพบจากระบบค้นหารูปภาพได้ และคำอธิบายรูปภาพซึ่งถูกเขียนขึ้นโดยมนุษย์นั้นบางครั้งไม่สามารถอธิบายได้ครอบคลุมเนื้อหาของรูปภาพได้หมดหรือไม่สื่อความหมายที่แท้จริงของรูปภาพ ดังนั้นการค้นหารูปภาพจากเมต้าเดต้าดังกล่าวจึงมีประสิทธิภาพต่ำ ดังนั้นนักวิจัยจึงให้ความสนใจการใช้คุณลักษณะของรูปภาพ (Image features) โดยจะดึงเอาคุณลักษณะต่างๆ ของรูปภาพออกมาเพื่อใช้ในการเปรียบเทียบกับคิวรี่โดยจะเริ่มจากการประมวลผลรูปภาพ (Image processing) เพื่อดึงคุณลักษณะของรูปภาพที่สำคัญออกมาคุณลักษณะของรูปภาพที่เป็นผลลัพธ์จากการประมวลผลรูปภาพนี้เป็นข้อมูลที่ไม่มีความหมายในตัวเอง และไม่สื่อถึงความหมายใดๆ ที่อยู่ในรูปภาพ ข้อมูลประเภทนี้จะเรียกว่า คุณลักษณะของรูปภาพในระดับต่ำ (Low-level features) ตัวอย่างข้อมูลประเภทนี้ได้แก่ สี (Color) รูปทรง (Shape) หรือลักษณะพื้นผิว (Texture) เป็นต้น การที่คุณลักษณะเหล่านี้ไม่สื่อถึงความหมายหรือเนื้อหาภายในรูปภาพ ทำให้ยากต่อผู้ใช้ในการที่จะใช้คุณลักษณะเหล่านี้ในการค้นหาข้อมูล ซึ่งปัญหานี้เป็นที่รู้จักกันดีในกลุ่มผู้พัฒนาระบบคืนคืนสารสนเทศมัลติมีเดียและเรียกว่า “ช่องว่างความหมาย (Semantic gap)” ดังนั้นระบบค้นคืนรูปภาพจะนำคุณลักษณะของรูปภาพในระดับต่ำนี้ไปใช้เพื่อทำการเปรียบเทียบกับคุณลักษณะของรูปภาพในคิวรี่ในขั้นตอนการค้นหา และการนำคุณลักษณะของรูปภาพในระดับต่ำนี้ไปใช้ทำได้สองแนวทาง คือ 1) นำคุณลักษณะของรูปภาพในระดับต่ำนี้ไปใช้โดยตรง *ไม่มีการแปลง*ให้อยู่ในรูปแบบของข้อมูลที่มีความหมายที่คนสามารถเข้าใจได้ จะเรียกระบบที่ใช้คุณลักษณะของรูปภาพในระดับต่ำนี้ในการค้นหาข้อมูลว่า การค้นหารูปภาพจากคอนเทนท์ (Content-based Image Retrieval-CBIR) และ 2) ระบบที่นำคุณลักษณะของรูปภาพในระดับต่ำนี้ไปแปลงให้เป็นข้อมูลที่คนสามารถเข้าใจได้ซึ่งจะเรียกว่าความหมายของรูปภาพในระดับสูง (High-level semantics) เช่น นำรูปทรงไปประมวลผลและแปลความหมายออกมาว่าเป็นรูปทรงของอะไร เช่น รูปทรงวงกลม (low-level feature) อาจจะถูกตีความหมายออกมาว่าเป็นพระอาทิตย์ (high-level semantics) ดังนั้นระบบจะทำการค้นหารูปภาพในคอเล็คชั่นที่มีพระอาทิตย์เป็นต้น

อย่างไรก็ตามพื้นฐานที่สำคัญของระบบค้นคืนรูปภาพนั้นจะเริ่มจากคุณลักษณะของรูปภาพในระดับต่ำเสมอ ดังนั้นในหัวข้อนี้จะทำการแนะนำถึงลักษณะสำคัญของคุณลักษณะต่างๆ ของรูปภาพให้ทราบก่อน เพื่อที่จะเป็นความรู้พื้นฐานในบทถัดไปโดยทั่วไปคุณลักษณะของรูปภาพในระดับต่ำนี้จะแบ่งออกเป็น 2 ประเภทคือ 1) คุณลักษณะแบบโกบอล (Global features) ซึ่งจะเป็นคุณลักษณะแบบรวมๆ ไม่เฉพาะเจาะจง และ 2) คุณลักษณะแบบโลคอล (Local features) ซึ่งจะเป็นคุณลักษณะเฉพาะของแต่ละส่วนในรูปภาพ

2.1.1 **คุณลักษณะแบบโกบอล (Global features)**

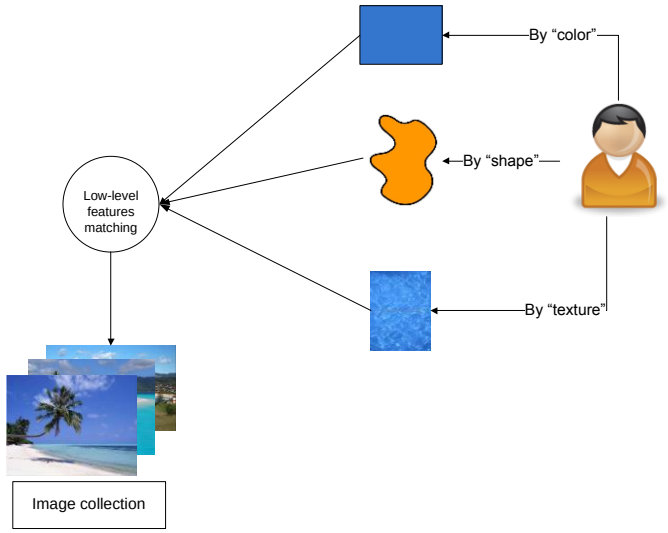
คุณลักษณะอันแรกแบบโกบอลคือ “**สี (Color)**” เป็นคุณลักษณะที่ถูกใช้มากที่สุดอันหนึ่งในระบบค้นคืนรูปภาพ ซึ่งสามารถนำไปใช้ในรูปแบบต่างๆ เช่น Color histogram Color moments Color coherence vector (Jing et al., 2003; Tong and Chang, 2001; Wang et al., 1999; Zheng et al., 2004) ในวิธีการเหล่านี้ ฮิสโตแกรมสี (Color histogram) เป็นวิธีที่ได้รับความนิยมมากที่สุด วิธีการนี้จะบอกค่าสถิติการกระจายของสีต่างๆได้แก่ สีแดง เขียว และน้ำเงิน หรือที่เรียกว่าสี RGB อธิบายได้อีกแบบหนึ่งคือวิธีการนี้จะดูจำนวนพิกเซลของแต่ละสีนั่นเอง แต่อย่างไรก็ตามการใช้สีแทนความหมายของรูปภาพนั้นยังมีประสิทธิภาพต่ำ เนื่องจากสีไม่สามารถสื่อความหมายของรูปภาพได้โดยตรงและถูกต้องเสมอไป ตัวอย่างเช่น สีส้ม อาจจะมีความหมายได้หลายอย่าง อาจจะหมายถึงพระอาทิตย์หรือไข่แดง ก็ได้

คุณลักษณะแบบโกบอลอันที่สองคือ “**พื้นผิว (Texture)**” ซึ่งให้ข้อมูลที่สำคัญเพื่อใช้ในการแยกประเภทของรูปภาพ เนื่องจากเป็นคุณลักษณะที่สามารถอธิบายสิ่งที่อยู่ในรูปภาพได้ เช่น ผลไม้ ก้อนเมฆ ต้นไม้ ผ้า หรือผิวหนังของสัตว์ เช่น ช้าง เสือ หรือคน อย่างไรก็ตาม พื้นผิวไม่สามารถตีความออกมาเป็นคำพูดของมนุษย์ได้ เนื่องจากจริงๆ แล้วเป็นข้อมูลทางสถิติ และเป็นข้อมูลที่ซ้ำๆ กันในรูปแบบเดียว (Pattern) พื้นผิวที่นำมาใช้ในระบบค้นคืนรูปภาพได้แก่ Spectral features ซึ่งได้มาจากการใช้เทคนิค Gabor filtering (Manjunath et al., 2001) Wavelet transform (Wang et al., 2001) หรือTamura texture features (Tamura et al., 1978) ซึ่ง 2 วิธีแรกได้รับความนิยมสูงสุดสำหรับนำมาประยุกต์ใช้กับระบบค้นคืนรูปภาพ

“**รูปทรง (Shape)**” เป็นคุณลักษณะที่สำคัญของรูปภาพถึงแม้ว่าจะไม่ค่อยนิยมนำมาใช้ในระบบค้นคืนรูปภาพมากนักเหมือนกัน สีและพื้นผิว อย่างไรก็ตามรูปทรงมีประโยชน์สำหรับระบบค้นคืนสารสนเทศบางระบบ เช่นระบบที่ค้นคืนข้อมูลรูปภาพที่มีลักษณะเฉพาะ การใช้รูปทรงในการค้นคืนข้อมูลนั้นมีความยุ่งยากอยู่พอสมควรเมื่อเปรียบเทียบกับวิธีการที่ใช้สีและพื้นผิว เนื่องจากต้องมีการแบ่งรูปภาพออกเป็นส่วนๆ (Segments) ก่อนเพื่อที่จะทำการดึงรูปทรงของวัตถุที่ปรากฏในรูปภาพออกมาได้ แต่ปัญหามีอยู่ว่าการแบ่งส่วนรูปภาพให้ได้รูปทรงที่ต้องการนั้นทำได้ยากมาก

ประโยชน์ของการใช้คุณลักษณะแบบโกบอลคือ มีความง่าย ไม่ซับซ้อนมากและมีความเร็วในการประมวลผลเพื่อดึงคุณลักษณะต่างๆ ของรูปภาพออกมา แต่ว่าคุณลักษณะแบบโกบอลนี้ก็มีข้อจำกัดอยู่คือ ความน่าเชื่อถือในการตรวจหาวัตถุในรูปภาพ (Object Recognition) เมื่อรูปภาพมีขนาด ความสว่าง หรือมุมของกล้องแตกต่างกันออกไปมีผลทำให้การตรวจหาวัตถุในรูปภาพมี

ประสิทธิภาพต่ำลง ดังนั้นคุณลักษณะแบบโลคอลจึงได้รับความสนใจจากนักวิจัย และถูกนำมาใช้ในการประมวลผลสำหรับระบบค้นคืนรูปภาพแทนคุณลักษณะแบบโกบอล



รูปที่ 1 การค้นหารูปภาพโดยใช้คอนเทนท์ (Content-based Image Retrieval)

การค้นหาข้อมูลรูปภาพโดยใช้คุณลักษณะของรูปภาพทั้งโกบอลและโลคอลนี้เรียกว่า "การค้นหารูปภาพจากคอนเทนท์" หรือเรียกสั้นๆ ว่า CBIR ดังรูปที่ 1

2.1.2 คุณลักษณะแบบโลคอล (Local features)

การที่จะค้นคืนข้อมูลรูปภาพโดยเปรียบเทียบรูปภาพในคิวรี่และรูปภาพในคอลเล็คชั่นโดยใช้คุณลักษณะแบบโกบอลนั้นค่อนข้างที่จะทำจะมีประสิทธิภาพต่ำ เนื่องจากเหตุผลหลายประการ เช่น รูปภาพเดียวกันแต่มีมุมกล้อง ความสว่างต่างกัน ระบบคอมพิวเตอร์ก็ไม่สามารถที่จะทราบได้ว่ารูปภาพสองรูปภาพนั้นเป็นรูปภาพเดียวกันหรือวัตถุที่อยู่ในรูปภาพเป็นวัตถุเดียวกัน ดังนั้นนักวิทยาศาสตร์จึงคิดหาวิธีที่จะนำคุณลักษณะอื่นๆ มาใช้ เพื่อให้ระบบคอมพิวเตอร์สามารถที่จะหารูปภาพลักษณะเดียวกันที่มีมุมกล้องหรือความสว่างต่างกันได้ ในปี 2004 นักวิจัยท่านหนึ่งชื่อ Lowe ได้เสนอคุณลักษณะของรูปภาพอันหนึ่งซึ่งไม่เปลี่ยนแปลงไปตามมุมกล้อง ความสว่าง หรือการหมุนของรูปภาพ คุณลักษณะนี้เรียกว่าคุณลักษณะซิฟ "(Scale Invariant Feature Transform-SIFT)" (Lowe, 2004)

SIFT จะหาจุดสำคัญต่างๆ ในรูปภาพ (keypoints) ซึ่งจะแสดงในรูปแบบของเว็กเตอร์ 128 ค่า (128 dimensional) และถูกใช้เป็นประโยชน์ในการแยกประเภทของรูปภาพ (Image classification) จากผลการเปรียบเทียบของนักวิจัยชื่อ Mikolajczyk (2003) ทำการทดสอบการแยกประเภทของรูปภาพโดยใช้คุณลักษณะของรูปภาพหลายๆ คุณลักษณะ และผลการทดสอบสรุปว่า SIFT ให้ประสิทธิภาพสูงสุดในการแยกประเภทของรูปภาพ

2.1.3 **คุณลักษณะซิฟท์ (SIFT Descriptors)**

คุณลักษณะ สี รูปทรงและพื้นผิวเป็นคุณลักษณะที่เรียกว่า *"โกบอล"* ซึ่งเป็นคุณลักษณะที่คำนวณจากข้อมูลทั้งหมดที่กระจายอยู่บนภาพ ข้อเสียของวิธีการนี้คือมีความน่าเชื่อถือต่ำเมื่อรูปภาพมีการเปลี่ยนขนาด มุมกล้อง หรือการจัดวาง ข้อมูลเหล่านี้จะมีการเปลี่ยนแปลงตามไปด้วยแม้ว่าจะเป็นรูปภาพเดียวกันก็ตาม ดังนั้นเมื่อนักวิทยาศาสตร์ท่านหนึ่งชื่อ Lowe ไดทำการเสนอคุณลักษณะหนึ่งซึ่งคุณลักษณะไม่เปลี่ยนไปตามสภาพแวดล้อมดังกล่าว คุณลักษณะนี้เรียกว่า ซิฟท์ (SIFT) ในการดึงคุณลักษณะซิฟท์จากรูปภาพออกมานั้น มีขั้นตอนการคำนวณอยู่ 4 ขั้นตอน ดังนี้

1) ทำการค้นหาจุดสำคัญ (keypoints) ของภาพโดยใช้อัลกอริทึม Different-of-Gaussian (DoG) ซึ่งจุดสำคัญของภาพนั้นจะไม่เปลี่ยนแปลงเมื่อมีการเปลี่ยนขนาดหรือการจัดวางของรูปภาพ ขั้นตอนนี้เรียกว่า "Scale-space extrema detection"
2) หลังจากนั้นจุดสำคัญที่หาได้จาก DoG จะถูกประมวลผลต่อโดยใช้ฟังก์ชั่น 3D quadratic จุดสำคัญใดที่มีคอนทราสต่ำ (contrast) และเป็นจุดที่อยู่บนเส้นขอบจะถูกกำจัดทิ้งเนื่องจากถือว่าเป็นจุดที่ไม่สำคัญ (noise) ขั้นตอนนี้เรียกว่า "Keypoint localization"
3) หลังจากนั้นจุดสำคัญที่เหลือมาจากขั้นตอนที่สองจะถูกกำหนดการจัดวาง (orientation) แบบต่างๆ ให้กับจุดสำคัญเหล่านั้นเพื่อให้มีความเสถียรต่อการเปลี่ยนแปลงของการจัดวางของรูปภาพ ขั้นตอนนี้เรียกว่า "Orientation assignment"
4) ขั้นตอนสุดท้ายจุดสำคัญจะถูกนำมาสร้างคำอธิบายภาพ (local image descriptor) ซึ่งคำนวณจากจุดสำคัญอื่นๆ ที่อยู่ใกล้เคียง (neighborhood) เพื่อแปลงเป็นเว็กเตอร์ที่มีมิติ 128 มิติซึ่งเรียกว่า "SIFT descriptors" ขั้นตอนนี้เรียกว่า "Keypoint descriptor"

เมื่อได้ข้อมูลซิฟท์มาแล้วข้อมูลนี้จะถูกนำไปใช้ต่อ เช่น นำไปใช้ในการจำแนกกลุ่มของภาพ (image classification) ในการเปรียบเทียบประสิทธิภาพของการจำแนกกลุ่มของภาพโดยใช้คุณลักษณะของรูปภาพต่างๆ ซิฟท์เป็นคุณลักษณะที่ให้ประสิทธิภาพในการจำแนกรูปภาพสูงสุดซึ่งทำการทดลองโดย Mikolajczyk (2003)

รูปที่ 2 ตัวอย่างรูปภาพที่ถูกประมวลผลเพื่อหาจุดสำคัญต่างๆ บนภาพ

ในระบบ CBIR นั้น เมื่อผู้ใช้ทำการใส่รูปภาพคิวรี่หรือที่เรียกว่า Query-by-Example (QBE) จากนั้นระบบจะทำการประมวลผลเพื่อทำการดึงเอาค่าซิฟท์ของรูปภาพออกมาเมื่อได้ข้อมูลซิฟท์ซึ่งเป็นข้อมูลของจุดสำคัญที่ไม่เปลี่ยนแปลงตามขนาดและการจัดวางของรูปภาพ ในการเปรียบเทียบความเหมือนระหว่างรูปภาพ 2 รูปนั้นจะไม่ทำการเปรียบเทียบโดยใช้ข้อมูลของจุดสำคัญโดยตรง แต่จุดสำคัญเหล่านี้จะถูกนำไปประมวลผลต่อ วิธีที่นิยมคือการนำจุดสำคัญเหล่านี้ไปสร้างวิชวลเวิร์ด (visual word) การสร้างวิชวลเวิร์ดทำได้โดยการจุดกลุ่ม (cluster) จุดสำคัญที่คล้ายคลึงกันออกเป็นกลุ่มๆ แต่ละกลุ่มคือหนึ่งวิชวลเวิร์ด ตัวอย่างอัลกอริทึมที่ใช้ในการจัดกลุ่มเช่น *k*-mean หรือ *x*-mean เป็นต้นหลังจากนั้นจะใช้หลักการของถุงคำ (bag of wods) มาเปรียบเทียบความเหมือนของคิวรี่และรูปภาพในคอลเล็คชั่น หลักการเปรียบเทียบจะเหมือนกันกับวิธีการ เปรียบเทียบค่าฮิสโตแกรมสี คือการนับความถี่วิชวลเวิร์ดที่ปรากฏในภาพ ถ้าฮิสโตแกรมของวิชวลเวิร์ดออกมามีค่าใกล้เคียงกันก็จะถือว่ารูปภาพนั้นๆ มีความเหมือนกัน ข้อเสียของวิธีการหาวิชวลเวิร์ดนี้คือใช้เวลาในการคำนวณเพื่อหาค่าซิฟท์นาน แต่ให้ประสิทธิภาพความถูกต้องในการจำแนกกลุ่มรูปภาพสูง

2.2 การทำดัชนีข้อมูลรูปภาพ

การทำดัชนีถือว่าเป็นส่วนสำคัญของระบบค้นคืนสารสนเทศ และมีผลโดยตรงต่อความถูกต้องและความเร็วในการค้นหาข้อมูล การทำดัชนีคือ วิธีการที่กำหนดคำสำคัญต่างๆ เพื่อใช้เป็นตัวแทนของเอกสาร ซึ่งจะทำให้เอกสารต่างๆ สามารถที่จะถูกค้นหาได้โดยการเปรียบเทียบคิวรี่กับคำสำคัญ

เหล่านั้น หลักการนี้มีลักษณะเช่นเดียวกับการทำดัชนีของระบบ CBIR นั่นคือการกำหนดคุณลักษณะระดับต่ำเพื่อใช้เป็นตัวแทนของรูปภาพ หลังจากนั้นคุณลักษณะเหล่านี้จะถูกเปลี่ยนโครงสร้างเพื่อให้ระบบสามารถที่จะทำการค้นหาและเปรียบเทียบข้อมูลได้ หลักการสำคัญในการทำดัชนีในระบบ CBIR คือ 1) ทำการลดขนาดของเว็กเตอร์คุณลักษณะลง (feature vector) 2) เลือกโครงสร้างข้อมูลที่เหมาะสมเพื่อใช้ในการทำดัชนี และ 3) หาวิธีวิธีการเปรียบเทียบความคล้ายคลึงที่มีประสิทธิภาพ การพัฒนาวิธีการทำดัชนีสำหรับระบบค้นคืนรูปภาพได้รับความสนใจจากนักวิจัยเป็นอย่างมาก เนื่องจากการทำดัชนีข้อมูลมัลติมีเดียมีความแตกต่างจากการทำดัชนีเอกสารทั่วไปอยู่พอสมควร ซึ่งลักษณะของการทำดัชนีไฟล์เอกสารนั้นจะเป็นโครงสร้างที่ชัดเจน มีการเรียงลำดับคีย์เวิร์ดและไฟล์ตามหมายเลขไฟล์ เช่น ตัวอย่างของการทำดัชนีแบบอินเวิร์ท (Inverted indexing) ซึ่งโครงสร้างของไฟล์ดัชนีนี้ช่วยให้การค้นหาเอกสารทำได้รวดเร็วยิ่งขึ้น จุดประสงค์หลักของการทำดัชนีคือการหาคำที่สำคัญและใช้คำนั้นเป็นคำที่เชื่อมต่อไปยังเอกสารที่คำนั้นปรากฏอยู่ ดังนั้นการทำดัชนีของข้อมูลรูปภาพก็ใช้หลักการเดียวกันคือ หาส่วนที่สำคัญของรูปภาพหรือวีดีโอมาทำดัชนีและเชื่อมต่อไปยังรูปภาพดังกล่าว ซึ่งข้อมูลที่จะนำมาทำดัชนีคือคุณลักษณะของรูปภาพนั่นเอง โดยเมื่อระบบทำการประมวลรูปภาพเพื่อดึงเอาคุณลักษณะที่สำคัญออกมา คุณลักษณะเหล่านี้จะถูกจัดโครงสร้างเพื่อความรวดเร็วต่อการค้นหาและเชื่อมต่อไปยังรูปภาพ แต่เนื่องจากคุณลักษณะที่ดึงออกมานั้นมีจำนวนมากจึงเกิดปัญหาขึ้นตามมา ดังนั้นหลักการสำคัญของการทำดัชนีโดยใช้คุณลักษณะของรูปภาพคือ

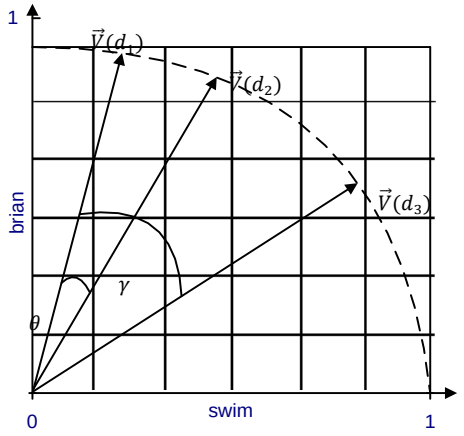
1) ต้องการลดขนาดของมิติของพื้นที่เว็กเตอร์ (Dimensionality of the vector space) ให้มีขนาดเล็กลง
2) ต้องหาโครงสร้างข้อมูลสำหรับดัชนีที่เหมาะสมกับคุณลักษณะของรูปภาพเพื่อการค้นหาและเปรียบเทียบที่รวดเร็ว
3) ต้องมีวิธีการวัดความคล้ายคลึงที่เหมาะสม

โดยปกติแล้ว ถ้าเราแสดงคุณลักษณะของรูปภาพที่จะนำมาใช้ทำดัชนีในรูปแบบของพื้นที่เว็กเตอร์จะทำให้ได้ตารางที่มีขนาดใหญ่มากหรือเรียกว่ามีจำนวนมิติของพื้นที่เว็กเตอร์ที่สูง (high dimensionality of feature vectors) ดังนั้นการที่จะทำดัชนีจึงต้องมีการลดจำนวนมิติของพื้นที่เว็ก-เตอร์ให้น้อยลง วิธีการที่นิยมคือวิธีการ PCA (Principal Component Analysis) หรือ SVD (Singular Value Decomposition) ซึ่งเป็นวิธีการที่สามารถนำไปประยุกต์ได้ทั้งข้อมูลที่เป็นตัวอักษรและรูปภาพเพื่อทำการลดมิติของพื้นที่เว็กเตอร์ หลังจากที่ทำการลดมิติของพื้นที่เว็กเตอร์แล้ว เราจะทำการพิจารณาหาโครงสร้างของการทำดัชนีที่เหมาะสมสำหรับคุณลักษณะของรูปภาพ โครงสร้างที่นิยมกันได้แก่ ไบนารีทรี (Binary tree) อาร์ทรี (R-tree) ในรูปแบบต่างๆ เช่น R*-tree SR-tree SS-

tree and Kd-tree เป็นต้น ซึ่งโครงสร้างเหล่านี้มีประสิทธิภาพในการทำดัชนีที่ดี แต่มีมิติของเว็กเตอร์ไม่เกิน 20 หากเว็กเตอร์มีมิติที่มากกว่านั้นประสิทธิภาพของการค้นหาจะลดลง

2.3 การวัดความคล้ายคลึงของรูปภาพ

เมื่อทำการเลือกโครงสร้างของการทำดัชนีได้แล้ว วิธีการวัดความคล้ายคลึงระหว่างคิวรี่และข้อมูลในดัชนีก็สำคัญเช่นเดียวกัน วิธีที่นิยมในการวัดความคล้ายคลึงระหว่างคิวรี่และข้อมูลในดัชนีคือการวัดความคล้ายคลึงโดยใช้ระยะทาง โดยอยู่บนพื้นฐานของพื้นที่เว็กเตอร์ซึ่งได้ถูกทำดัชนีไว้ก่อนหน้านี่แล้วนอกจากนี้ยังใช้วิธีการวัดความคล้ายคลึงกันโดยระยะทางระหว่างฮิสโตแกรมสำหรับระบบที่ทำดัชนีรูปภาพด้วยสี แม้ว่านักวิจัยจะพยายามพัฒนาวิธีการวัดความคล้ายคลึงกันของคุณลักษณะของภาพและคิวรี่ แต่ดูเหมือนว่าการวัดความคล้ายคลึงระหว่างคิวรี่และคุณลักษณะของรูปภาพยังต้องพัฒนาอีกมาก โดยการทำวิจัยในหัวข้อนี้ต้องใช้องค์ความรู้ในหลายๆ ด้านประกอบกันเช่น วิทยาการคอมพิวเตอร์ สถิติ และคณิตศาสตร์ประยุกต์ สำหรับการวัดความคล้ายคลึงกันของรูปภาพในงานวิจัยนี้จะใช้วิธีการที่ชื่อว่า “ความคล้ายคลึงโคซายน์ (Cosine similarity)”



รูปที่ 3 เปรียบเทียบความเหมือนกันระหว่างเอกสารต่างๆ

ความคล้ายคลึงโคซายน์ เป็นวิธีการหาคล้ายคลึงกันจากค่าความต่างของมุมของวัตถุ 2 อันที่เกิดขึ้นบนพื้นที่เว็กเตอร์ ดังแสดงในรูปที่ 3 ซึ่งความคล้ายคลึงกันแบบโคซายน์นี้จะมีค่าอยู่ระหว่าง 0 - 1 เท่านั้น วิธีการนี้เป็นที่นิยมและมีประสิทธิภาพสูงในการวัดความคล้ายคลึงระหว่างวัตถุ 2 อันและถูกนำมาประยุกต์ใช้กับศาสตร์การค้นคืนข้อมูลอย่างแพร่หลาย ซึ่งตัวหารมีชื่อเรียกฉพาะว่า Euclidean lengths/distance ซึ่งวิธีการนี้จะมีประสิทธิภาพในกรณีที่เอกสาร 2 เอกสารมีความยาวไม่เท่ากันหรือทำให้มีความยุติธรรมต่อเอกสารที่สั้นกว่านั้นเอง ความคล้ายคลึงโคซายน์คำนวณได้จากสมการที่ 1

$$sim(x, y) = \frac{\sum_{i=1}^{n} x_i \times y_i}{\left(\sum_{i=1}^{n} (x_i - y_i)^2 \right)^{1/2}} \quad (1)$$

2.4 การค้นคืนสารสนเทศเชิงความหมาย

ปัญหาสำคัญของระบบการค้นหาข้อมูลเอกสารโดยใช้การเปรียบเทียบอักขระหรือการค้นหารูปภาพโดยใช้คุณลักษณะระดับต่ำคือการค้นหาและเปรียบเทียบระหว่างคิวรี่และข้อมูลในคอลเล็คชั่นไม่ได้พิจารณาจากคอนเซฟท์หรือความหมายที่จริงของคิวรี่ การค้นหาโดยดูจากความหมายของคิวรี่หมายความว่ากลไกการค้นหาไม่ได้เปรียบเทียบสายอักขระที่เหมือนกันเท่านั้น (String matching) แต่จะพิจารณาคำอื่นๆ ที่มีความเหมายเหมือนกันกับคำในคิวรี่ด้วยเช่นกัน ตัวอย่างเช่นถ้าผู้ใช้ต้องการค้นหาข้อมูลโดยใช้คำว่า Car ดังนั้นเอกสารใดๆ ที่มีคำว่า BMW Honda และ Toyota ควรจะถูกพิจารณาว่าเป็นเอกสารที่เกี่ยวข้องกับคิวรี่เช่นกัน เนื่องจากทั้ง BMW Honda และ Toyota ถือเป็นคำที่มีความเกี่ยวของกับคอนเซฟท์ Car ทั้งสิ้น ถึงแม้ว่าเอกสารเหล่านั้นจะไม่มีคำว่า Car ปรากฏอยู่เลยก็ตาม

สำหรับระบบค้นคืนรูปภาพแบบ CBIR ข้อจำกัดสำคัญคือคุณลักษณะเหล่านั้นไม่สามารถใช้เป็นข้อมูลแทนความหมายของเนื้อหาในรูปภาพได้อย่างมีประสิทธิภาพ เช่น สีเขียว อาจจะหมายถึงทุ่งหญ้า ต้นไม้ หรือสีของชุดทหารก็ได้ ปัญหาการที่คุณลักษณะระดับต่ำไม่สื่อถึงความหมายของเนื้อหาในรูปภาพเป็นปัญหาที่สำคัญที่นักวิจัยพยายามที่จะแก้ไขอยู่ ปัญหานี้เป็นที่รู้จักกันดีในชื่อว่าปัญหาช่องว่างเชิงความหมาย (Semantic gap) สำหรับการค้นหารูปภาพโดยพิจารณาจากความหมายของรูปภาพ หมายความว่าค้นหารูปภาพที่มีเนื้อหาในรูปภาพที่มีความสัมพันธ์หรือเกี่ยวข้องกันถึงแม้ว่ารูปภาพเหล่านั้นจะมีคุณลักษณะระดับต่ำที่แตกต่างกันก็ตาม เช่น มีสีหรือพื้นผิวที่ต่างกัน ตัวอย่างเช่น รูปภาพของรถและรูปภาพของปั้มน้ำมันถือว่ามีความสัมพันธ์กันเนื่องจากรถต้องเติมน้ำมัน หรือรูปภาพกีฬาทุ่มน้ำหนักและรูปภาพกีฬากระโดดสูง มีความสัมพันธ์กันเนื่องจากถือเป็นกีฬาประเภทลาน (Field event) เหมือนกันเป็นต้น ลักษณะของระบบที่มีกลไกการค้นคืนข้อมูลแบบนี้เรียกว่า “ระบบค้นหาข้อมูลเชิงความหมาย (Semantic-based Information Retrieval-SBIR)

2.5 วิธีการทำดัชนีสำหรับระบบค้นคืนสารสนเทศเชิงความหมาย

ในหัวข้อนี้จะแนะนำถึงวิธีการทำดัชนีเพื่อรองรับการค้นหาข้อมูลเชิงความหมาย ได้แก่วิธี Latent Semantic Indexing (LSI) และวิธีการทำดัชนีโดยใช้ออนโทโลจี (Ontology Indexing) อย่างไรก็ตามก่อนที่จะทำดัชนีนั้น เอกสารต้องมีการประมวลผลเช่นการเปลี่ยนรูปคำ การกำจัดคำที่ไม่สำคัญทิ้งไป เป็นต้น เพื่อทำการดึงเอาเมต้าเดต้าที่สำคัญออกมจากเอกสารเสียก่อน ในหัวข้อถัดไปจะอธิบายถึงขั้นตอนการทำดัชนีเพื่อรองรับการค้นหาเชิงความหมาย

2.6 Latent Semantic Indexing (LSI)

วิธีการ LSI เป็นวิธีการที่มีหลักการทำงานอยู่บนพื้นฐานของการคำนวณทางสถิติ โดยพิจารณาจากการปรากฏร่วมของคำต่างๆ (co-occurrence) เป็นเทคนิคการทำดัชนีแบบใหม่คิดโดย Deerwester (1990) โดยใช้หลักการความสัมพันธ์ของคีย์เวิร์ดและเอกสารมาใช้ในการทำดัชนี นอกจากจะพิจารณาความถี่ของคำที่ปรากฏในเอกสารหนึ่งแล้ว ยังจะพิจารณาถึงความสัมพันธ์ของคำนั้นๆ กับเอกสารอื่นๆ ใน คอลเล็คชั่นด้วย เรียกว่า “ความเกี่ยวพันธ์กัน (Interrelationship)” LSI จะพิจารณาเอกสารใดๆ ที่มีจำนวนคีย์เวิร์ดปรากฏร่วมกันสูง ถือว่าเอกสารเหล่านั้นมีความสัมพันธ์กันเชิงความหมาย (Semantically related) และเอกสารใดๆ ที่มีคีย์เวิร์ดปรากฏร่วมกันอยู่จำนวนน้อยจะถือว่ามีความสัมพันธ์กันเชิงความหมายต่ำ เนื่องจากว่าเอกสารใดๆ อาจจะมีความสัมพันธ์กันเชิงความหมายถึงแม้ว่าเอกสารนั้นๆ จะไม่มีคีย์เวิร์ดที่ซ้ำกันเลยก็ตาม LSI สามารถหาความสัมพันธ์ของเอกสารเหล่านั้นได้จากการคำนวณทางสถิติ ดังนั้น LSI จึงหาคำตอบของคิวรี่ที่มีประสิทธิภาพดีกว่าวิธีการหาเอกสารแบบเปรียบเทียบสายอักขระถึงแม้ว่าเอกสารนั้นๆ จะไม่ปรากฏคีย์เวิร์ดในคิวรี่เลยก็ตาม ตัวอย่างเช่น LSI สามารถที่จะทราบได้ว่าเอกสารเกี่ยวกับ BMW Toyota และ Honda เป็นเอกสารที่ควรจะเป็นคำตอบของคิวรี่ Car แม้ว่าเอกสารเหล่านั้นจะไม่มีคำว่า Car ปรากฏอยู่เลยก็ตาม หลังจากที่ทำการเตรียมเอกสารและตัดคำที่ไม่สำคัญทิ้งเรียบร้อยแล้ว ขั้นแรกของวิธีการ LSI คือการสร้างเมทริกซ์ความสัมพันธ์ระหว่างคีย์เวิร์ดและเอกสาร (Term-document matrix) เมทริกซ์นี้เป็นตารางที่เก็บความถี่ของคำต่างๆ ที่ปรากฏในเอกสารทั้งหมดในคอลเล็คชั่น

ถ้าให้เรามีเอกสารในคอลเล็คชั่นอยู่ทั้งหมด 9 เอกสาร ซึ่งปรากฏคำต่างๆ ดังต่อไปนี้สมมติให้คีย์เวิร์ดคือคำที่เป็นตัวหนาเอียง (ในความเป็นจริงแล้วคีย์เวิร์ดที่ได้โดยใช้หลักการกำจัดคำที่ไม่สำคัญทิ้ง (บทที่ 2) อาจจะได้คีย์เวิร์ดที่แตกต่างจากตัวอย่างนี้ ทั้งนี้ตัวอย่างนี้แสดงขึ้นเพื่อความเข้าใจในหลักการทำงานของ LSI เท่านั้น)

doc1: ***Human*** machine ***interface*** for ABC ***computer*** applications

doc 2: A ***survey*** of ***user*** opinion of ***computer system response time***

doc 3: The ***EPS user interface*** management ***system***

doc 4: ***System*** **and** ***human system*** engineering testing of ***EPS***

doc 5: Relation of ***user*** perceived ***response time*** to error measurement

doc 6: The generation of random, binary, ordered ***trees***

doc 7: The intersection ***graph*** of paths in ***trees***

doc 8: ***Graph minors*** IV: Widths of ***trees*** and well-quasi-ordering

doc 9: ***Graph minors***: A ***survey***

เมื่อทำการหาคีย์เวิร์ดเรียบร้อยแล้วให้ทำการสร้างเมทริกซ์ความสัมพันธ์ระหว่างคีย์เวิร์ดและเอกสาร (*A*) ซึ่งจะได้ผลลัพธ์ดังรูปที่ 4 คอลัมน์สุดท้ายคือจำนวนเอกสารทั้งหมดที่คีย์เวิร์ดปรากฏอยู่ (*df*) โดยไม่สนใจว่าคีย์เวิร์ดนั้นจะปรากฏในเอกสารเดียวกันกี่ครั้งจะนับเป็น 1 เสมอ

	Keywords	doc1	doc2	doc3	doc4	doc5	doc6	doc7	doc8	doc9	*df*
A=	human	1	0	0	1	0	0	0	0	0	2
	interface	1	0	1	0	0	0	0	0	0	2
	computer	1	1	0	0	0	0	0	0	0	2
	user	0	1	1	0	1	0	0	0	0	3
	system	0	1	1	2	0	0	0	0	0	3
	response	0	1	0	0	1	0	0	0	0	2
	time	0	1	0	0	1	0	0	0	0	2
	EPS	0	0	1	1	0	0	0	0	0	2
	survey	0	1	0	0	0	0	0	0	1	2
	tree	0	0	0	0	0	1	1	1	0	3
	graph	0	0	0	0	0	0	1	1	1	3
	minor	0	0	0	0	0	0	0	1	1	2

รูปที่ 4 ตัวอย่างเมทริกซ์แสดงความสัมพันธ์ระหว่างคีย์เวิร์ดและเอกสารต่างๆ

หลังจากนั้นเพื่อให้เกิดความยุติธรรมกับเอกสารที่มีความสั้นและยาวไม่เท่ากัน ความถี่ของคำเอกสารในเมทริกซ์จึงต้องมีการให้น้ำหนักและทำการนอร์มอไลซ์ (Normalize) โดยใช้สมการที่ 2

$$W_{ij} = L_{ij} G_i N_j \qquad (2)$$

ซึ่ง W_{ij} คือน้ำหนักของแต่ละคำ L_{ij} คือน้ำหนักของคีย์เวิร์ด i ในเอกสาร j เรียกว่า “น้ำหนักแบบโลคอล (Local weight)” G_i คือน้ำหนักของคีย์เวิร์ด i ในคอเล็คชั่นเรียกว่า “น้ำหนักแบบโกบอล (Global weight)” และ N_j คือค่านอร์มอลไลซ์สำหรับเอกสาร j ในการคำนวณน้ำหนักแบบต่างๆ นั้นสามารถทำได้หลายแบบ แต่ในบทนี้จะแนะนำสูตรการคำนวณหาน้ำหนักที่ถูกแนะนำและทดลองแล้วว่ามีประสิทธิภาพสูง (Chisholm and Kolda, 1999)

น้ำหนักแบบโลคอล คือการหาว่าคีย์เวิร์ดหนึ่งมีความถี่ในการปรากฏในเอกสารหนึ่งเท่าไหร่ ซึ่งสามารถคำนวณได้จากสมการที่ 3

$$L_{ij} = \begin{cases} \log(f_{ij} + 1) & \text{if } f_{ij} > 0L \\ 0 & \text{if } f_{ij} = 0 \end{cases} \qquad (3)$$

โดยที่ f_{ij} คือความถี่ของคีย์เวิร์ด i ปรากฏในเอกสาร j จากรูปที่ 7-1 นำมาคำนวณหาน้ำหนักแบบโลคอลและผลลัพธ์แสดงดังรูปที่ 5

การคิดน้ำหนักแบบโกบอลคือการหาความถี่ของคีย์เวิร์ดที่ปรากฏในทุกเอกสารในคอลเล็คชั่น สามารถคำนวณได้โดยใช้ Inverse Document Frequency (IDF) ผลลัพธ์การคำนวณหาน้ำหนักคำแบบโกบอลแสดงดังรูปที่ 6

Keywords	doc1	doc2	doc3	doc4	doc5	doc6	doc7	doc8	doc9
Human	0.30	0.00	0.00	0.30	0.00	0.00	0.00	0.00	0.00
interface	0.30	0.00	0.30	0.00	0.00	0.00	0.00	0.00	0.00
computer	0.30	0.30	0.00	0.00	0.00	0.00	0.00	0.00	0.00
User	0.00	0.30	0.30	0.00	0.30	0.00	0.00	0.00	0.00
System	0.00	0.30	0.30	0.48	0.00	0.00	0.00	0.00	0.00
response	0.00	0.30	0.00	0.00	0.30	0.00	0.00	0.00	0.00
Time	0.00	0.30	0.00	0.00	0.30	0.00	0.00	0.00	0.00
EPS	0.00	0.00	0.30	0.30	0.00	0.00	0.00	0.00	0.00
survey	0.00	0.30	0.00	0.00	0.00	0.00	0.00	0.00	0.30
tree	0.00	0.00	0.00	0.00	0.00	0.30	0.30	0.30	0.00
graph	0.00	0.00	0.00	0.00	0.00	0.00	0.30	0.30	0.30
minor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.30	0.30

รูปที่ 5 ผลลัพธ์การคำนวณหาน้ำหนักแบบโลคอลของคีย์เวิร์ดทั้งหมดในเมทริกซ์

Keywords	Global weight (G_i)
human	0.65
interface	0.65
computer	0.65
user	0.48
system	0.48
response	0.65
time	0.65
EPS	0.65
survey	0.65
tree	0.48
graph	0.48
minor	0.65

รูปที่ 6 ผลลัพธ์การคำนวณหาน้ำหนักแบบโกบอลของคีย์เวิร์ดทั้งหมดในเมทริกซ์

	doc1	doc2	doc3	doc4	doc5	doc6	doc7	doc8	doc9
N_j	2.94	2.26	2.90	2.78	3.20	3.82	4.92	3.54	3.20

รูปที่ 7 เมทริกซ์ผลลัพธ์จากการทำนอร์มอไลเซชั่น

ส่วนการทำนอร์มอไลซ์นั้นมีความจำเป็นที่ต้องทำเนื่องจาก เอกสารที่มีความยาวมากกว่าจะทำให้มีจำนวนคำสูงกว่าเอกสารที่สั้นๆ ทำให้มีโอกาสที่จะได้น้ำหนักของคำสูงกว่า และมีผลต่อการเปรียบเทียบความคล้ายคลึงระหว่างคิวรี่และเอกสารอื่นๆ สมการที่ 4 แสดงสูตรการคำนวณนอร์มอไลเซชั่นและรูปที่ 7 แสดงผลการทำนอร์มอไลซ์เซชั่น

$$N_j = \frac{1}{\sqrt{\sum_{i=0}^{m} (G_i L_{ij})^2}} \qquad \textbf{(4)}$$

จากสมการที่ 2 จะได้เมทริกซ์หลังจากคำนวณน้ำหนักของคีย์เวิร์ดดังแสดงดังรูปที่ 8 ในกรณีที่มีเอกสารจำนวนมากทำให้เมทริกซ์ความสัมพันธ์ระหว่างคีย์เวิร์ดและเอกสารมีขนาดใหญ่มาก ดังนั้นขั้นตอนต่อไปคือการลดมิติของเมทริกซ์ให้เล็กลง LSI ก็ใช้หลักการของ SVD ในการลดขนาดของเมทริกซ์ เมื่อทำการแตกเมทริกซ์ออกเป็น 3 เมทริกซ์ย่อยๆ แล้วเมทริกซ์ *U* จะเรียกว่า “เว็กเตอร์คอนเซฟท์ (concept vector)” ซึ่งเป็นตัวแทนหัวข้อหลักๆ หรือเป็นกลุ่มเอกสารหลักๆ ในคอลเล็คชั่นนั่นเอง และหัวข้อเหล่านี้ถูกเรียงลำดับตามความสำคัญของค่าเอกพจน์ตามแนวเส้นแทยงมุมของเมทริกซ์ *S* นอกจากนั้นแล้วเมทริกซ์ *S* ยังสามารถใช้บอกความสำคัญของหัวข้อต่างๆ อีกด้วย เมื่อทำการหาค่าของทั้ง 3 เมทริกซ์ได้แล้ว จะใช้หลักการของ Rank-*K* Approximation เพื่อลดขนาดของเมทริกซ์ลงจากเมทริกซ์ใน 8 ผลลัพธ์ของเมทริกซ์ *U S* และ *V* แสดงดังรูปที่ 9 ในตัวอย่างนี้สมมติให้ใช้ *Rank-2 Approximation* ดังนั้นข้อมูลที่เราจะใช้จริงๆ คือส่วนที่แรเงา

Keywords	doc1	doc2	doc3	doc4	doc5	doc6	doc7	doc8	doc9
human	0.58	0.00	0.00	0.55	0.00	0.00	0.00	0.00	0.00
interface	0.58	0.00	0.57	0.00	0.00	0.00	0.00	0.00	0.00
computer	0.58	0.44	0.00	0.00	0.00	0.00	0.00	0.00	0.00
user	0.00	0.32	0.42	0.00	0.46	0.00	0.00	0.00	0.00
system	0.00	0.32	0.42	0.63	0.00	0.00	0.00	0.00	0.00
response	0.00	0.44	0.00	0.00	0.63	0.00	0.00	0.00	0.00
time	0.00	0.44	0.00	0.00	0.63	0.00	0.00	0.00	0.00
EPS	0.00	0.00	0.57	0.55	0.00	0.00	0.00	0.00	0.00
survey	0.00	0.44	0.00	0.00	0.00	0.00	0.00	0.00	0.63
tree	0.00	0.00	0.00	0.00	0.00	0.55	0.71	0.51	0.00
graph	0.00	0.00	0.00	0.00	0.00	0.00	0.71	0.51	0.46
minor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.70	0.63

รูปที่ 8 เมทริกซ์ผลลัพธ์หลังจากคำนวณน้ำหนักของคีย์เวิร์ด

Keywords	*U*								
human	-0.1	0.29	0.36	-0.2	0.3	0.48	0.41	-0.2	-0.4
interface	-0.1	0.31	0.3	-0.1	0.32	-0.7	-0	-0.1	-0.2
computer	-0.2	0.26	-0	-0.3	0.57	0.11	-0.2	0.21	0.51
user	-0.2	0.33	-0.3	0.19	-0.1	-0.4	-0	-0	0
system	-0.2	0.4	0.23	0.09	-0.4	0.3	-0.2	0.17	0.39
response	-0.2	0.27	-0.5	0.15	0.05	0.1	0.2	-0.1	-0.1
time	-0.2	0.27	-0.5	0.15	0.05	0.1	0.2	-0.1	-0.1
EPS	-0.1	0.33	0.36	0.16	-0.4	-0.1	0.02	-0	-0.1
survey	-0.3	0.05	-0.2	-0.5	-0.2	0.12	-0.6	0.03	-0.5
tree	-0.5	-0.3	0.12	0.57	0.25	0.1	-0.2	0.44	-0.3
graph	-0.6	-0.3	0.06	0.04	0	-0	-0.1	-0.7	0.27
minor	-0.5	-0.2	-0	-0.5	-0.2	-0.2	0.54	0.4	0.13

S								
1.52	0	0	0	0	0	0	0	0
0	1.46	0	0	0	0	0	0	0
0	0	1.19	0	0	0	0	0	0
0	0	0	0.91	0	0	0	0	0
0	0	0	0	0.87	0	0	0	0
0	0	0	0	0	0.65	0	0	0
0	0	0	0	0	0	0.51	0	0
0	0	0	0	0	0	0	0.37	0
0	0	0	0	0	0	0	0	0.23

รูปที่ 9 เมทริกซ์ผลลัพธ์ *U S* และ *V* จากการคำนวณ SVD

V^T								
-0.16	-0.3	-0.21	-0.18	-0.19	-0.17	-0.47	-0.55	-0.47
0.34	0.41	0.46	0.4	0.33	-0.11	-0.29	-0.32	-0.17
0.32	-0.45	0.31	0.45	-0.61	0.06	0.11	0.07	-0.09
-0.3	-0.13	0.19	0.07	0.3	0.35	0.47	-0.01	-0.65
0.79	0.08	-0.31	-0.36	0.02	0.16	0.2	-0.03	-0.27
-0.07	0.25	-0.69	0.64	-0.07	0.09	0.1	-0.11	-0.05
0.16	-0.49	-0.19	0.21	0.5	-0.17	-0.34	0.5	-0.12
-0.07	0.27	0.03	-0.06	-0.22	0.65	-0.53	0.36	-0.17
-0.09	0.39	0.02	-0.06	-0.29	-0.59	0.06	0.45	-0.45

2.7 การทำดัชนีโดยใช้ออนโทโลจี

เนื่องจากมนุษย์มีระบบความคิดและการรับรู้ถึงความหมายของสิ่งต่างๆ ได้ง่ายขึ้น ถ้าความสัมพันธ์ของสิ่งต่างๆ แสดงในรูปแบบของโครงสร้างลำดับชั้น (Hierarchical model) ดังนั้นจึงเกิดแนวคามคิดการทำดัชนีในรูปแบบของโครงสร้างลำดับชั้น ซึ่งเป็นลักษณะคล้ายกลับโครงสร้างต้นไม้ (Tree)

ออนโทโลจี (Ontology) คือการจัดองค์ความรู้หรือฐานความรู้ (Knowledge base) ให้อยู่ในรูปแบบโครงสร้างลำดับชั้น โดยประกอบด้วยคอนเซฟท์ (Concept) ต่างๆ และความสัมพันธ์ของคอน-เซฟท์ (Relashionship) เหล่านั้นโครงสร้างนี้อำนวยความสะดวกในการค้นหาแบบ SBIR เพื่อลดช่องว่างของช่องว่างเชิงความหมาย (semantic gap) ให้แคบลง ประโยชน์ของการทำดัชนีด้วยออน-โทโลจีที่เหนือกว่าการทำดัชนีด้วยวิธีอื่นๆ สามารถสรุปได้ดังต่อไปนี้ (Chandrasekaran et al., 1999; Gruber, 1993; Guarino et al., 1995; McGuinness, 2003; Schreiber et al., 1994)

1) ออนโทโลจีเป็นแหล่งรวบรวมคีย์เวิร์ดต่างๆ ที่เกี่ยวข้องกับเรื่องหรือโดเมนที่สนใจ (Domain) นอกจากนี้ยังให้ข้อมูลความสัมพันธ์ของคีย์เวิร์ดต่างๆ อีกด้วย ซึ่งความสัมพันธ์ของคำเหล่านี้จะเป็นตัวกำหนดความหมายที่แท้จริงของคีย์เวิร์ดต่างๆ ซึ่งจะช่วยแก้ปัญหาความกำกวมของคีย์-เวิร์ด เช่น หนึ่งคำมีหลายความหมาย (Polysemy) หรือคำหลายๆ คำมีความหมายเดียวกัน (Synonym) (Chandrasekaran et al., 1999).
2) ออนโทโลจีใช้เป็นตัวกำหนดลำดับชั้นของคอนเซฟท์ต่างๆ (Taxonomy) ภายในโดเมนที่สนใจ (Gasevic et al., 2009) ซึ่งแต่ละคอนเซฟท์จะจัดกลุ่มข้อมูลที่สัมพันธ์กันไว้ในกลุ่มเดียวกัน ซึ่งโครงสร้างลำดับชั้นนี้จะเป็นตัวช่วยในการค้นคืนเอกสารเชิงความหมายนั่นเอง
3) ออนโทโลจีสามารถที่จะถูกใช้ร่วมกันระหว่างแอฟพลิเคชั่นหรือนำมาใช้ใหม่ได้ (Gasevic et al., 2009, p. 53) เนื่องจากออนโทโลจีเป็นแหล่งคีย์เวิร์ดที่ใช้อธิบายคอนเซฟท์ต่างๆ ของโดเมน รูปแบบการเก็บข้อมูลดัชนีสามารถที่จะแชร์กันระหว่างแอปพลิคชั่นต่างๆ ได้โดยวิธีการต่อไปนี้ (Neches et al., 1991)
    3.1) แชร์กันโดยทำการก๊อปปี้ฐานความรู้ไปใช้ในแอปพลิเคชั่นใหม่โดยอาจจะมีการปรับปรุงโครงสร้างหรือทำการเพิ่มเติมองค์ความรู้ให้มากขึ้นและปรับใช้กับแอปพลิเคชั่นใหม่ได้
    3.2) แชร์กันผ่านทางซอฟท์แวร์เอเจนท์ ซึ่งสามารถเข้าถึงฐานความรู้ของออนโทโลจีได้
    3.3) แชร์กันผ่านทางเว็บเซอร์วิสเทคโนโลยี (Web service)

2.8 การออกแบบออนโทโลจี

การสร้างออนโทโลจีไม่มีกฎเกณฑ์ที่ตายตัว ขึ้นอยู่กับคนออกแบบว่ามีความรู้และความเข้าใจต่อโดเมนของเรื่องที่สนใจมากน้อยเพียงใด และคนออกแบบต้องการนำเสนอองค์ความรู้ของตนอย่างไร หลักการเบื้องต้นของการออกแบบออนโทโลจีคือจะจัดกลุ่มข้อมูลที่มีคุณสมบัติ (property) และลักษณะ (charateristics) คล้ายๆ กันอยู่ในกลุ่มเดียวกัน ซึ่งเรียกว่า “คอนเซฟท์ (Concept)” นั่นเอง แต่ละคอนเซฟท์มีอาจจะความสัมพันธ์กันซึ่งไม่จำกัดจำนวนของคอนเซฟท์ที่จะมีความสัมพันธ์กัน อาจจะเป็น 1:1 หรือ 1:M ก็ได้โดยทั่วไปการออกแบบออนโทโลจีมีขั้นตอนดังต่อไปนี้ (Poslad, 2009, p. 266):

1) กำหนดคอนเซฟท์หลักๆ ว่ามีอะไรบ้างและแต่ละคอนเซทฟ์สามารถที่จะแตกออกเป็นคอนเซฟท์ย่อยๆ ได้ โดยทั่วไปคอนเซฟท์ที่อยู่ระดับสูงกว่าจะเป็นคอนเซฟท์ที่มีความหมายกว้างกว่า และครอบคลุมทุกคอนเซฟท์ที่อยู่ระดับต่ำลงไป
2) กำหนดความสัมพันธ์ระหว่างคอนเซฟท์ต่างๆ ความสัมพันธ์เหล่านี้จะมีลักษณะตั้งแต่ง่ายๆ เช่น is-a, kind-of จนถึงความสัมพันธ์ที่ซับซ้อนมากขึ้น
3) กำหนดคุณสมบัติต่างๆ ของคลาสและกำหนดรายละเอียดของคุณสมบัติ เช่น กำหนดค่าที่เป็นไปได้ทั้งหมด (Possible value) เช่น เป็นตัวอักษรเท่านั้น หรือเป็นตัวเลขที่เป็นค่าบวกเท่านั้น เป็นต้น
4) กำหนดสัจพจน์ (axiom) หรือความจริงของความสัมพันธ์ ตัวอย่างเช่น A is-owner B ซึ่งจะมีควาหมายตรงกับ B *is-belongs to* A เป็นต้น

โดยทั่วไปแล้วการออกแบบออนโทโลจีนี้จะต้องทำการตรวจสอบซ้ำไปซ้ำมา และทำการแก้ไขหลายๆ ครั้งจนกว่าจะได้ออนโทโลจีที่ให้ประสิทธิภาพในการค้นหาข้อมูลที่ยอมรับได้

2.9 เครื่องมือที่ใช้ในการออกแบบออนโทโลจี

สำหรับการสร้างออนโทโลจีนั้นมีเครื่องมือสำหรับช่วยสร้างออนโทโลจีในลักษณะของ GUI เช่น Protégé (http://protege.stanford.edu) SWOOP (http://code.google.com/p/swoop) และ KAON (http://kaon.semanticweb.org) ในปัจจุบันนี้การพัฒนาออนโทโลจีและการใส่เมต้าเดต้าเข้าไปใน ออนโทโลจียังต้องใช้มนุษย์เป็นคนช่วยทำซึ่งถือเป็นงานที่ต้องใช้เวลามาก นอกจากนี้เครื่องมือช่วยสร้างและออกแบบออนโทโลจีในปัจจุบันมีปัญหาในการจัดการกับเมต้าเดต้าขนาดใหญ่ เช่นใช้เวลาในการอ่านข้อมูลและทำการประมวลผลนานสำหรับออนโทโลจีที่มีข้อมูลขนาดใหญ่และมี

โครงสร้างที่มีความซับซ้อนสูง ในงานวิจัยนี้จะใช้ Protégé ในการพัฒนาระบบ ส่วนเครื่องมืออื่นๆ ผู้อ่านสามารถหาอ่านได้จาก URL ที่ให้ไว้

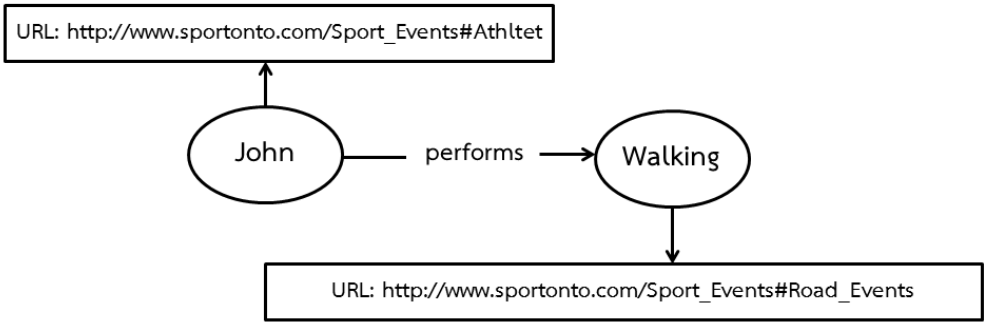
2.10 วิธีการนำเสนอองค์ความรู้สำหรับระบบค้นคืนสารสนเทศเชิงความหมาย

ในหัวข้อนี้จะอธิบายถึงวิธีการแทนองค์ความรู้ที่ถูกจัดเก็บในออนโทโลจี โดยองค์ความรู้จะถูกเก็บไว้ในรูปแบบของภาษามาร์คอัพต่างๆ ซึ่งที่นิยมกันในปัจจุบันได้แก่ RDF RDFS และ OWL ภาษาต่างๆ เหล่านี้มีข้อดีและข้อเสียต่างๆ กัน ซึ่งแล้วแต่ผู้พัฒนาระบบจะเลือกใช้

2.10.1 RDF

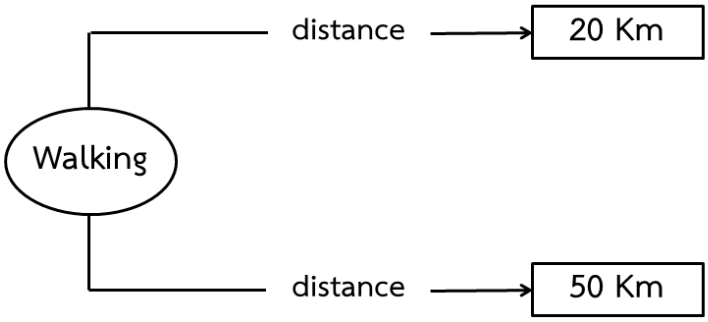
RDF (http://www.w3.org/TR/REC-rdf-syntax/) ย่อมาจาก Resource Description Framework ซึ่งพัฒนาโดย W3C (World Wide Web Consortium) เพื่อใช้อธิบายแหล่งข้อมูลบนเว็บไซต์ ข้อมูลในรูปแบบ RDF จะถูกจัดเก็บโดยใช้โครงสร้างของ semantic network ซึ่งเป็นรูปแบบหนึ่งของกราฟ ซึ่งประกอบด้วยโหนด (Node) และเส้นเชื่อมระหว่างโหนด (Edge) แสดงความสัมพันธ์ระหว่างโหนดต่างๆ โหนดใช้เป็นตัวแทนของคอนเซฟท์ หรือค่าของข้อมูลที่อยู่ในโหนด ส่วนเส้นเชื่อมระหว่างโหนดจะเป็นตัวแทนความสัมพันธ์ของโหนด หรือคุณสมบัติของโหนดต่างๆ โมเดล RDF จะประกอบด้วยส่วนสำคัญ 3 ส่วนคือ

1) Resource ซึ่งเป็นประเภทข้อมูลที่ถูกอธิบายโดยใช้นิพจน์ของ RDF ซึ่งเรียกว่า URI (Uniform Resource Identifier)
2) Propertiesเป็นส่วนที่ใช้อธิบายลักษณะและความสัมพันธ์ต่างๆ ของ Resource
3) Statements เป็นส่วนกำหนดค่าให้กับส่วน Property ของ Resource ที่ระบุ ในส่วนนี้จะถูกมองว่ามีลักษณะคล้ายๆ กับประโยคภาษาอังกฤษ ซึ่งประกอบด้วย ประธาน (Subject) กริยา (Verb) และกรรม (Object) ตัวอย่างเช่น John performs high jump ในประโยคนี้ John เป็นประธานperforms เป็นกริยา และ high jump เป็นกรรม ถ้าจะนำเสนอในรูปแบบของ RDF จะได้ว่า John และ high jump เป็น Resources และ performs เป็น Property แสดงดังรูปที่ 10



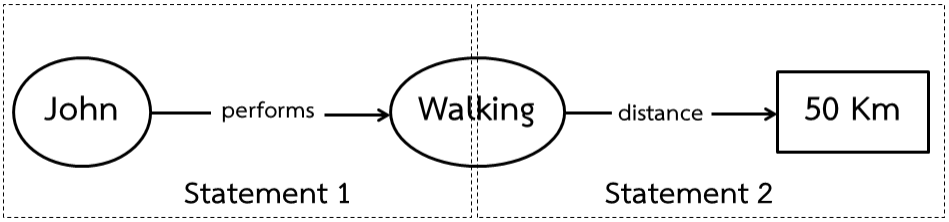
รูปที่ 10 ตัวอย่าง Statement ในโมเดล RDF ซึ่งประกอบด้วย Resources และ Property

นอกจากนี้ Statement อาจจะแสดงในรูปของ Object-Atribute-Value (O-A-V) หรือที่รู้จักกันว่า “Triplets” ตัวอย่างเช่นรูปที่ 11



รูปที่ 11 ตัวอย่าง O-A-V statement (Triplets)

นอกจากนี้แล้ว Object ของ Statement ไม่จำเป็นต้องเป็นคลาสเดี่ยวๆ เท่านั้น ยังสามารถเป็น Statement อื่นๆ ก็ได้ตัวอย่างเช่น John performs walking that has distance 50 km เป็นต้น นั่นคือนำ Statement จากรูปที่ 10 และรูปที่ 11 มารวมกันนั่นเอง ดังรูปที่ 12



รูปที่ 12 ตัวอย่าง Statement ที่มี Object เป็นอีก Statement หนึ่ง

2.10.2 RDFS (Schema)

หลักการทำงานของระบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System-RDBMS) คือการเก็บข้อมูลเป็นตาราง (Table) คอลัมน์ (Column) และคีย์หลัก (Primary key) เป็นต้น ข้อมูลเหล่านี้เป็นที่รู้จักในหลายๆ RDBMS ที่มีการทำงานแบบเดียวกัน นั่นหมายความว่าข้อมูลเหล่านี้สามารถถูกสร้างโดย RDBMS อันนึงและสามารถจะถูกใช้งานโดยอีก RDBMS หนึ่งได้ เนื่องจากระบบเหล่านั้นมีการจัดการข้อมูลโดยใช้โครงสร้างข้อมูลแบบเดียวกันและสามารถที่จะเข้ากฎต่างๆ ที่ถูกสร้างขึ้นในอีก RDBMS หนึ่งได้โดยอัตโนมัติ กฎเหล่านี้ใช้เป็นเมต้าเดต้าที่อธิบายข้อมูลที่อยู่ในตารางอีกทีหนึ่งซึ่งเป็นมาตรฐานกลางที่ทุกๆ RDBMS ใช้ร่วมกัน เช่นเดียวกัน RDFS ถูกสร้างขึ้นเพื่อวัตถุประสงค์เดียวกัน นั่นคือ RDFS จะให้ข้อมูลเพื่ออธิบายคลาสและคุณสมบัติต่างๆ เพื่อเป็นเมต้าเดต้าของข้อมูลในออนโทโลจี โดยมีการเพิ่มประเภทของคลาสและคุณสมบัติต่างๆ มากขึ้น ซึ่งสามารถแบ่งเป็นกลุ่มๆ ดังต่อไปนี้

- **คลาสหลัก (Core class):** ได้แก่ *rdfs:Resource rdfs:Literal rdf:XMLLiteral rdfs:Class rdf:Property* และ *rdfs:Datatype* โดยที่คลาส *rdfs:Resource* เป็นคลาสบนสุดหรือพูดได้ว่าเป็นคลาสที่ครอบคลุมคลาสอื่นๆ ทุกคลาสในออนโทโลจี เป็นคลาสที่กำหนดแหล่งข้อมูลของเว็บของ RDF นั้นๆส่วนคลาส *rdfs:Literal* และ *rdf:XMLLiteral* เป็นคลาสที่ใช้กำหนดชนิดข้อมูล เช่น ตัวอักขระ ตัวเลข และอธิบายชนิดข้อมูลในรูปแบบของ XML ตามลำดับ *rdfs:Classrdf:Property* และ *rdfs:Datatype* ใช้สำหรับกำหนดคลาสต่างๆ คุณสมบัติของคลาส และชนิดข้อมูลในรูปแบบของ RDF ในออนโทโลจี ตัวอย่างเช่น
- **คุณสมบัติหลัก (Core property):** ได้แก่ *rdf:type rdfs:subClassOf rdfs:subPropertyOf rdfs:domain rdfs:range rdfs:label* และ *rdfs:comment* โดยหน้าที่หลักของแต่ละคุณสมบัติมีดังต่อไปนี้ *rdf:type* ใช้เป็นตัวบอกว่า URI นั้นเป็นข้อมูล instance ของคลาส คุณสมบัติ *rdfs:subClassOf* และ *rdfs:subPropertyOf* เป็น

คุณสมบัติเพื่อกำหนดลำดับชั้นของคลาสและคุณสมบัติในออนโทโลจี *rdfs:domain* และ *rdfs:range* เป็นคุณสมบัติเพื่อบอกโดเมนและช่วงค่าข้อมูลที่เป็นไปได้ของคุณสมบัติต่างๆ สุดท้าย *rdfs:label* และ *rdfs:comment* เป็นคุณสมบัติที่ใช้อธิบาย URI โดยใช้ภาษาอังกฤษ โดยที่ *rdfs:label* จะใช้กับชื่อของคลาส และ *rdfs:comment* จะใช้อธิบายรายละเอียดของคลาสที่ยาวขึ้น

- **คลาสและคุณสมบัติแบบคอนเทนเนอร์ (Container classes and properties):** ได้แก่ *rdfs:Container rdf:Bag rdf:Seq rdf:Alt rdfs:ContainerMembershipProperty* และ *rdfs:member* โดยที่ *rdfs:Container* เป็นคุณสมบัติที่กำหนด URI ที่เป็นคอลเล็คชั่นซึ่งอาจจะเป็นแบบ *rdf:Bagrdf:Seq*หรือ *rdf:Alt* คุณสมบัติ *rdfs:member* ใช้เป็นคุณสมบัติที่กำหนดสมาชิกของคอนเท็นเนอร์ คุณสมบัติ *rdfs:ContainerMembershipProperty* ใช้เพื่อกำหนดความสัมพันธ์ระหว่าง URI และคอนเท็นเนอร์
- **คอลเล็คชั่น (Collections):** ได้แก่ *rdf:List rdf:first rdf:rest* และ *rdf:nil* คลาสประเภท *rdf:List* ใช้เพื่ออธิบายข้อมูลที่เป็นรายการ (List) คุณสมบัติ *rdf:first rdf:rest* ใช้เป็นคุณสมบัติเพื่อการจัดการข้อมูลที่เป็นรายการ และคุณสมบัติ *rdf:nil* เป็น instance ของข้อมูลรายการที่ว่างเปล่า
- **คำศัพท์ที่ใช้อธิบายองค์ความรู้ (reification vocabulary):** ได้แก่ *rdf:Statement rdf:predicate rdf:subject* และ *rdf:object* คลาสและคุณสมบัติเหล่านี้ใช้สำหรับอธิบายองค์ความรู้ในฐานความรู้ เช่น *rdf:Statement* อธิบายข้อมูลในลักษณะของ Triples และ *rdf:predicate rdf:subject* และ *rdf:object* ใช้กำหนดคุณสมบัติของคลาส แหล่งข้อมูลของคลาสที่ใช้เป็นประธานและกรรมตามลำดับ
- **คุณสมบัติอำนวยประโยชน์ (Utility properties):** ได้แก่ *rdfs:seeAlso rdfs:isDefinedBy* และ *rdf:value*เป็นคุณสมบัติที่ใช้อธิบายข้อมูลเพิ่มเติมของคลาส

ตัวอย่างด้านล่างแสดงให้เห็นถึงการใช้งานของ RDFS เพื่อกำหนดความหมายของกีฬาเดินทน (Walking) และคุณสมบัติ Walk_distance อย่าสับสนระหว่าง rdf:resource ซึ่งใช้อ้างถึงแหล่งข้อมูลที่ RDF จะอ้างถึงและ rdf:Resource ซึ่งหมายถึง URI ของไฟล์องค์ความรู้หรือที่อยู่ของ RDF นั่นเอง (จะแสดงอยู่ด้านบนของไฟล์)

```
<rdfs:Class rdf:about="&SportOntology;Walking"
            rdfs:label="Walking">
            <rdfs:subClassOf rdf:resource="&SportOntology;Road_Events"/>
<rdf:Property rdf:about="&SportOntology;Walk_Distance"
             a:maxCardinality="1"
             a:minCardinality="1"
            rdfs:label="Walk_Distance">
            <rdfs:domain rdf:resource="&SportOntology;Walking"/>
            <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
```

รูปที่ 13 ตัวอย่างไฟล์ประเภท RDFS

2.10.3 OWL

OWL (Ontology Web Language) ถูกพัฒนาขึ้นโดย W3C เช่นกันและถูกสร้างขึ้นบนพื้นฐานของ RDFS การแสดงองค์ความรู้โดยใช้ OWL มีอยู่ 3 ประเภทคือ OWL-Lite OWL-DL และ OWL-Full ซึ่งมีขอบเขตการใช้งานและความซับซ้อนที่ต่างกันเรียงจากน้อยไปหามาก OWL เหมาะสมกับระบบที่ต้องการวิเคราะห์เหตุผล (Reasoning) เพื่อหาผลลัพธ์การค้นหาข้อมูล เนื่องจากสามารถใช้ตัวดำเนินการบูลีน (Boolean) มาช่วยในการวิเคราะห์ข้อมูลได้ OWL จะใช้คลาสบางคลาสร่วมกับ RDFS และมีคลาสและคุณสมบัติบางอันที่พัฒนาขึ้นมาใหม่โดยเฉพาะ ประเภทของคลาสและคุณสมบัติใน OWL สามารถจัดกลุ่มได้ดังต่อไปนี้

- **คลาสสำหรับกำหนดคลาสและเงื่อนไขต่างๆ ในออนโทโลจี (Classes for defining classes and restrictions)** ได้แก่ *owl:Class* และ *owl:Restriction*
- **คลาสสำหรับกำหนดคุณสมบัติต่างๆ (Class for defining properties)** ได้แก่ *owl:ObjectProperty owl:DatatypeProperty owl:TransitiveProperty owl:SymmetricProperty owl:FunctionalProperty owl:InverseFunctionalProperty* และ *owl:AnnotationProperty* คลาสเหล่านี้ใช้กำหนดคุณสมบัติต่างๆ เช่น *owl:ObjectProperty* กำหนดคุณสมบัติเพื่อเชื่อมต่อคลาสหนึ่งไปยังอีกคลาสหนึ่ง *owl:DatatypeProperty* ใช้เพื่อเชื่อมต่อคลาสกับชนิดข้อมูล *owl:FunctionalProperty* และ *owl:InverseFunctionalProperty* ใช้เพื่อกำหนดจำนวนของความสัมพันธ์ระหว่างคลาส เช่น 1:1 หรือ 1:3 เป็นต้น

- **คลาสสำหรับแสดงความไม่เหมือนกันของข้อมูล (Classes for stating inequality among individuals)** ได้แก่ *owl:AllDifferent* เป็นการระบุว่าค่า instance ต่างๆ มีความแตกต่างกัน ซึ่งต้องกำหนดเนื่องจากบางครั้งมีข้อมูล 2 อันคนละชื่อแต่ทั้งสองข้อมูลอ้างถึงสิ่งเดียวกัน
- **คลาสสำหรับอธิบายถึงชนิดข้อมูลที่เป็นชุด (Classes for describing enumerations of datatypes)** ได้แก่ *owl:DataRange*
- **คลาสที่มีอยู่แล้ว (Predefined classes)** ได้แก่ *owl:Thing* ซึ่งเป็นคลาสที่อยู่สูงสุดครอบคลุมข้อมูลทุกอย่างในออนโทโลจี และ *owl:Nothing* เป็นคลาสที่มีความเฉพาะเจาะจงสูงซึ่งตรงข้ามกับ *owl:Thing*
- **คลาสที่ใช้อธิบายออนโทโลจี (Classes for describing ontologies)** ซึ่งจะเป็นคลาสที่เป็นรูท (root) ของออนโทโลจีครอบคลุมข้อมูลทั้งหมด

คุณสมบัติของ OWL นั้นไม่ต่างจากคุณสมบัติของ RDFS มากนัก แต่มีคุณสมบัติที่สำคัญเพิ่มเติมขึ้นมาจาก RDFS ซึ่งจะขอยกตัวอย่างเพียงบางคุณสมบัติเท่านั้น เช่น

- คุณสมบัติ *owl:inverseOf* เพื่อระบุว่าเป็นค่าผกผันกับคุณสมบัติใดเช่น *parentOf* ผกผันกับ *childOf*
- คุณสมบัติ *owl:equivalentClass* เพื่อระบุว่าคลาสดังกล่าวเหมือนกับคลาสอื่นในออนโทโลจี
- คุณสมบัติ *owl:TransitiveProperty* เพื่อถ่ายทอดความสัมพันธ์ระหว่างคุณสมบัติ เช่น P(x,y) และ P(y,z) ดังนั้น P(x,z)
- คุณสมบัติ *owl:SymmetricProperty* เพื่อกำหนดความเหมือนกันทุกประการระหว่างคุณสมบัติ เช่น P(x,y) เหมือนกันทุกประการกับ P(y,x)
- คุณสมบัติ owl:FunctionalProperty เพื่อกำหนดการอ้างถึงระหว่างคุณสมบัติ เช่น P(x,y) และ P(x,z) ดังนั้น y = z

เงื่อนไขบางอย่างที่ไม่สามารถกำหนดได้ในคุณสมบัติของ RDFS สามารถที่จะกำหนดในคุณสมบัติของ OWL ได้ ข้อแตกต่างสำคัญระหว่าง RDFS และ OWL คือสามารถกำหนดค่าที่เป็นไปได้ให้กับคุณสมบัติได้ เช่น กีฬาเดินทน มีระยะทางการแข่งขัน (Walking distance) อยู่เพียง 2 ระยะทาง คือ 20 และ 50 กิโลเมตร ซึ่งใน RDFS นั้นไม่สามารถกำหนดให้คุณสมบัติ Walking distance มีเพียง 2

ค่าดังกล่าวได้ โดยภาพรวม OWL มีความเหมาะสมกับระบบที่ต้องมีการวิเคราะห์เหตุผลเพื่อหาข้อมูลที่เกี่ยวข้องกับคิวรี่ ข้อเสียคือมีความซับซ้อนสูงและส่งผลให้ใช้เวลาในการประมวลผลนาน
ตัวอย่างของ OWL แสดงดังหน้าถัดไป

```
<owl:Class rdf:ID="Walking">
<rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Walking</rdfs:label>
<rdfs:subClassOf rdf:resource="#Road_Events"/>
</owl:Class>
<owl:FunctionalProperty rdf:ID="Walk_Distance">
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
<rdf:typerdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
<rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Walk_Distance</rdfs:label>
<rdfs:domain>
<owl:Class>
<owl:unionOf rdf:parseType="Collection">
<rdf:Description rdf:about="http://www.w3.org/2002/07/owl#Thing"/>
<owl:Class rdf:about="#Walking"/>
</owl:unionOf>
</owl:Class>
</rdfs:domain>
</owl:FunctionalProperty>
```

รูปที่ 14 ตัวอย่างไฟล์ประเภท OWL

2.11 วิธีการค้นหาข้อมูลจากฐานความรู้

ออนโทโลจีคือโมเดลสำหรับเก็บเมต้าเดต้าหรือองค์ความรู้ของเรื่องที่เราสนใจ ซึ่งมีประสิทธิภาพในการดัชนีองค์ความรู้ในลักษณะของโครงสร้างลำดับชั้น สิ่งที่สำคัญอีกอันหนึ่งคือวิธีการเข้าถึงองค์ความรู้เหล่านี้อย่างมีประสิทธิภาพ การเข้าถึงข้อมูลในฐานความรู้ในออนโทโลจีนั้นมีลักษณะคล้ายๆ กับเทคนิคการเข้าถึงข้อมูลของ RDBMS ซึ่งมีการทำงานอยู่บนพื้นฐานของ SQL (Structure Query Language) และถูกดัดแปลงมาใช้เพื่อเป็นภาษาในการเข้าถึงข้อมูลในออนโทโลจีโมเดล เช่น RDF ในที่นี่จะยกตัวอย่างภาษาที่ใช้สำหรับคิวรี่ข้อมูลในออนโทโลจีบางภาษาที่เป็นที่นิยม ได้แก่ RDQL

และ SPARQL เนื้อหาในส่วนนี้จะอธิบายตัวอย่างการใช้งานเบื้องต้นเท่านั้น ซึ่งเป็นพื้นฐานสำหรับเพื่อปรับใช้ต่อไปในอนาคต

2.11.1 **RDQL**

RDQL (RDF Data Query Language) เป็นภาษาสำหรับคิวรี่ข้อมูลบน RDF ที่ถูกพัฒนาขึ้นให้มีลักษณะการทำงานเหมือนกับ SQL ซึ่งรองรับการทำงานในรูปแบบของ SELECT...FROM... WHERE... และ USINGนอกจากนี้ยังสามารถทำงานร่วมกับ Java API เช่น Jena และ PHP ได้อีกด้วย รูปแบบของคิวรี่แสดงดังรูปที่ 15

รูปแบบคำสั่ง	ตัวอย่าง
SELECT *var1, var2,..*	SELECT ?x,?y
FROM *rdfdocuments*	FROM <http://example.com/sample.rdf>
WHERE *Expressions*	WHERE (?x,<foo:name>,?y)
AND *Filters*	AND ?y=="Alex"
USING *Namespace declarations*	USING foo for <http://foo.org/properties#>

รูปที่ 15 รูปแบบคำสั่งคิวรี่ของ RDQL

รูปแบบคำสั่งและตัวอย่างในรูปที่ 7-19 จะเห็นว่า ตัวแปรที่ตามหลัง SELECT ต้องมีเครื่องหมายคำถาม (?) นำหน้าตัวแปรเสมอ ข้อมูลที่ตามหลังคำสั่ง SELECT คือข้อมูลที่ต้องการให้ RDQL ค้นหาและแสดงต่อผู้ใช้ ส่วนของ FROM แสดงถึงแหล่งองค์ความรู้หรือไฟล์ RDF ที่ต้องการใช้เพื่อค้นหาข้อมูล ซึ่งจะถูกระบุภายในเครื่องหมาย <> เสมอ ถ้าหากต้องการค้นหาข้อมูลมากกว่าหนึ่งไฟล์ สามารถทำได้โดยใช้เครื่องหมายคอมมา (,) เป็นตัวขั้น และสามารถที่จะระบุในรูปแบบของ URL หรือว่าเป็นชื่อไฟล์ธรรมดาก็ได้ เช่น

```
FROM <sample1.rdf>, <http://example.com/sample2.rdf>
```

เช่นเดียวกับ SQL ส่วนของ WHERE คือส่วนที่ใช้กำหนดเงื่อนไขในการค้นหา แต่ต่างจาก SQL ตรงที่เงื่อนไขใน RDQL นั้นอยู่ในรูปแบบของ RDF triples (Subject, Predicate, Object) ซึ่งข้อมูลจะตรง

กับทั้งสามเงื่อนไขจึงจะถือเป็นคำตอบที่ตรงกับคิวรี่ เงื่อนไขต้องอยู่ในเครื่องหมายวงเล็บ () เสมอ ส่วน Predicate ต้องอยู่ภายในเครื่องหมาย <> เสมอ เงื่อนไขสามารถมีมากกว่าหนึ่งเงื่อนไขโดยใช้คอมมาเป็นตัวขั้นเช่นกัน จากในตัวอย่างในรูปที่ 7-19 (?x,<foo:name>,?y) หมายถึงหาข้อมูลใดๆ (x) ที่มีคุณสมบัติ name ที่มีค่า (y) เท่ากับ “Alex” หรือพูดง่ายๆ คือ หาข้อมูลใดๆ ที่มีชื่อว่า Alex นั่นเอง ในตัวอย่างจะใช้ AND มาเป็นตัวช่วยเพื่อระบุค่าที่ต้องการค้นหา ส่วนสุดท้ายคือ USING เป็นการประกาศ namespace ที่จะใช้ทั้งหมดซึ่งจะอยู่ส่วนบนสุดของ RDF เช่น

```
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:ns1="http://www.w3.org/2001/vcard-rdf/3.0#"
xmlns:ns2="http://sampleVocabulary.org
```

2.11.2 SPARQL

SPARQL (SPARQL Protocol and RDF Query Language) อ่านว่า “สปาเคิ้ล” เป็นภาษาคิวรี่ที่พัฒนาต่อมาจาก RDQL โดย W3C ซึ่งมีคำสั่งเพิ่มเติมมากกว่า เช่น สามารถใช้คำสั่ง SELECT ซ้อนกันได้ มีคำสั่ง INSERT UPDATE DELETE และสามารถเรียงลำดับข้อมูลผลลัพธ์ได้ ซึ่งทำให้ SPARQL มีประสิทธิภาพมากกว่า RDQL และได้รับความนิยมมากกว่า รูปแบบคำสั่งคิวรี่ของ SPARQL สามารถแสดงได้ดังรูปที่ 16

รูปแบบคำสั่ง	ตัวอย่าง
PREFIX *Namespace declarations* SELECT [DISTINCT]*var1, var2,…* FROM *rdfdocuments or URI* WHERE {*Expressions* FILTER regex(var, "pattern" [, "flags"])}	PREFIX ns1: <http://www.w3.org/2001/vcard-rdf/3.0#> ns2: <http://sampleVocabulary.org/1.3/People#> SELECT ?givenName WHERE { ?x vcard:Family ?y . ?x vcard:Given ?givenName . FILTER (?y = “George”) }

รูปที่ 16 รูปแบบคำสั่งคิวรี่ของ SPARQL

จากรูปแบบคำสั่งในรูปที่ 16 จะเห็นว่า SPARQL มีความคล้ายคลึงกับ RDQL มากมีความแตกต่างเล็กน้อยคือ SPARQL ไม่ใช้คำสั่ง USING แต่เปลี่ยนไปใช้คำสั่ง PREFIX แทน และคำสั่ง

WHERE เงื่อนไขต้องอยู่ภายใน {} เท่านั้นและไม่ใช้เครื่องหมายวงเล็บ () และคอมมา (,) ขั้นระหว่างเงื่อนไขต่างๆ แต่จะใช้จุด (.) แทน รวมถึงส่วนของ Predicateไม่ใช่เครื่องหมาย <> อีกต่อไป ตัวอย่างเช่น ?x vcard:Family ?y. นอกจากนี้คำสั่ง AND ใน RDQL เปลี่ยนเป็นคำสั่ง FILTER () เพื่อป้องกันความสับสนกับคำสั่ง AND ที่เป็นคำสั่งทางตรรกะ (Logical operator) นอกจากนี้ SPARQL ยังรองรับคำสั่งอื่นๆ เช่น OPTION หรือORDERBY ซึ่งมีลักษณะใกล้เคียงกับคำสั่ง SQL มากขึ้นด้วย ดังนั้นหากนักศึกษามีความคุ้นเคยกับคำสั่งภาษา SQL จึงไม่เป็นการยากที่จะทำความเข้าใจกับคำสั่งของ SPARQL จากตัวอย่างเดิมของไฟล์ Employee.rdf เราสามารถเปลี่ยนคำสั่งของภาษา RDQL เป็นภาษา SPARQL ได้ดังรูปที่ 17 ตัวอย่างนี้ใช้เพื่อค้นหาคนที่มีอายุเกินกว่า 30 ปี

```
PREFIX ns1: <http://www.w3.org/2001/vcard-rdf/3.0#>
      ns2: <http://sampleVocabulary.org/1.3/People#>

SELECT ?givenName, ?age
FROM  <Employee.rdf>
WHERE {?xns1:N ?blank.
      ?blank ns1:Given ?givenName.
      ?x ns2:age ?age.
      FILTER(?age > 30)
     }
```

รูปที่ 17 ตัวอย่าง SPARQL เพื่อหาคนที่มีอายุมากกว่า 30 ปี ในออนโทโลจี

รูปที่ 18 แสดงผลลัพธ์การค้นหาข้อมูลของ SPARQL

```
g
===========
Monica
George
```

รูปที่ 18 ตัวอย่างผลลัพธ์การค้นหาข้อมูลโดย SPARQL

โดยปกติแล้ว RDQL หรือ SPARQL จะใช้ร่วมกับ PHP หรือ Jena ซึ่งจะสามารถแสดงผลลัพธ์ในรูปแบบของ HTML ได้หรือนำไปประมวลผลต่อแล้วแต่แอปพลิเคชั่น หนังสือเล่มนี้จะไม่ขอกล่าว

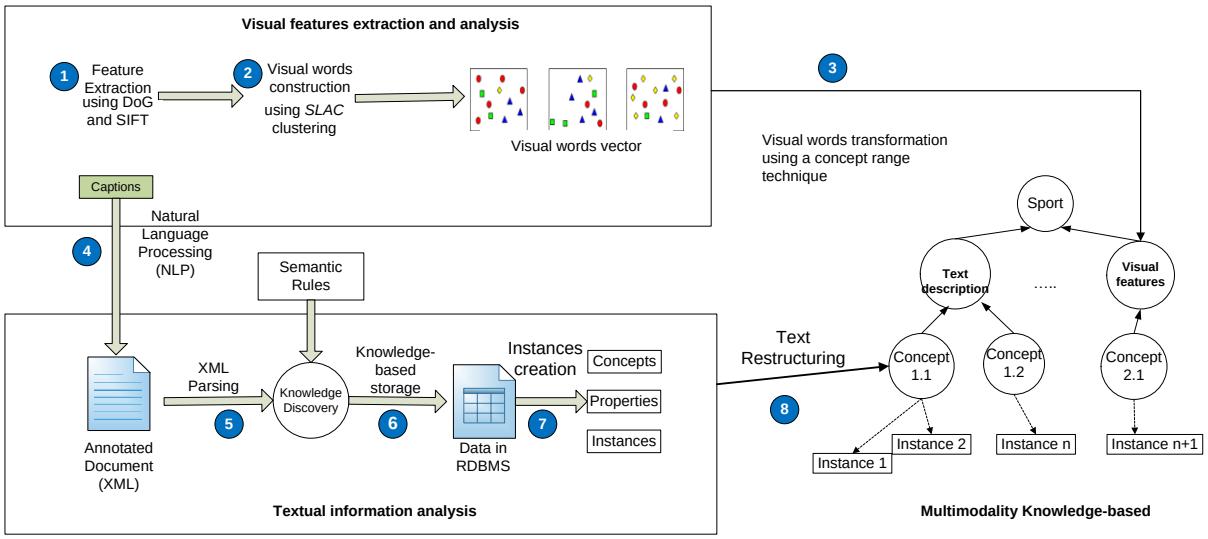
รายละเอียดของ Jena เพิ่มเติม และสามารถศึกษารายละเอียดได้ที่ (http://jena.sourceforge.net) ผู้ที่มีความรู้เกี่ยวกับ Java สามารถทำความเข้าใจและเรียนรู้การใช้งานได้ไม่ยากนัก

สำหรับในงานวิจัยนี้จะใช้ SPARQL เป็นหลักในการค้นหาข้อมูลรูปภาพที่เก็บไว้ในออนโทโลจีที่ได้ออกแบบไว้ สำหรับรายละเอียดของออนโทโลจีที่ได้ออกแบบไว้นั้น สามารถศึกษาได้ในบทถัดไป

บทที่ 3
ระเบียบวิธีวิจัย

3.1 การออกแบบการวิจัยและกรอบแนวคิด

ในการทำวิจัยนี้แนวคิดหลักคือพยายามที่จะใช้คุณลักษณะของรูปภาพแบบโลคอล เพื่อใช้เป็นตัวแทนของความหมายที่อยู่ในรูปภาพนั้นๆ และเก็บคุณลักษณะดังกล่าวในรูปแบบของโครงสร้างลำดับชั้น เช่น ออนโทโลจี นอกจากคุณลักษณะของรูปภาพแล้ว คำอธิบายรูปภาพก็มีส่วนสำคัญในการใช้ให้เป็นประโยชน์เพื่อยืนยันความหมายที่แท้จริงของรูปภาพ ดังนั้นในงานวิจัยนี้จึงจะใช้ข้อมูลจาก 2 แหล่ง ได้แก่ คุณลักษณะของรูปภาพ (Low-level features) และคำอธิบายรูปภาพ (Caption) รูปที่ 19 แสดงกรอบแนวคิดในภาพรวมของระบบที่พัฒนาขึ้น



รูปที่ 19 กรอบแนวคิดงานวิจัย

จากกรอบแนวคิดงานวิจัยในรูปที่ 19 จะประกอบด้วย 2 ส่วนหลักๆ ได้แก่ 1) ส่วนวิเคราะห์รูปภาพ และ 2) ส่วนวิเคราะห์ตัวอักษร ซึ่งทั้งส่วนนี้ทำหน้าที่คล้ายคลึงกันคือ การดึงเอาองค์ความรู้ (Knowledge extraction) ที่ซ่อนอยู่ในข้อมูลทั้ง 2 รูปแบบไปใส่ไว้ในออนโทโลจีหรือเรียกว่า ต้นแบบ

องค์ความรู้ (Knowledge-based model) นั่นเอง กรอบแนวคิดดังกล่าวประกอบด้วยทั้งหมด 8 ขั้นตอน ซึ่งมีการทำงานที่เชื่อมโยงกันเป็นระบบเดียว ซึ่งในแต่ละขั้นตอนจะอธิบายเบื้องต้นได้ดังต่อไปนี้

ขั้นตอนที่ 1 : การดึงคุณลักษณะรูปภาพ (Feature extraction) หมายถึงการประมวลผลรูปภาพและดึงเอาคุณลักษณะแบบโลคอล (SHIFT) ออกมา

ขั้นตอนที่ 2 : การสร้างวิชวลเวิร์ด (Visual word construction) เป็นขั้นตอนที่นำคุณลักษณะแบบโลคอลมาสร้างเป็นวิชวลเวิร์ด เพื่อใช้เป็นตัวแทนของรูปภาพนั้นๆ วิชวลเวิร์ดในขั้นตอนนี้จะมีลักษณะมีโครงสร้างเป็นเว็กเตอร์ (vector space model)

ขั้นตอนที่ 3 : การแปลงโครงสร้างวิชวลเวิร์ด (Visual word transformation) เป็นขั้นตอนที่นำวิชวลเวิร์ดที่ได้จากขั้นตอนที่ 2 มาแปลงโครงสร้างให้อยู่ในรูปของโครสร้างแบบลำดับชั้นในออนโทโลจี

ขั้นตอนที่ 4 : การประมวลผลภาษาธรรมชาติ (Natural language processing) หมายถึงขั้นตอนที่ใช้ประมวลผลคำอธิบายรูปภาพซึ่งเป็นตัวอักษร ผลลัพธ์ของขั้นตอนนี้จะอยู่ในรูปแบบ XML

ขั้นตอนที่ 5 : การถอดแท็กซ์ XML (XML parsing) เป็นการนำผลจากขั้นตอนที่ 4 มาถอดแท็กซ์ XML และนำไปใช้สร้างองค์ความรู้ในขั้นตอนถัดไป

ขั้นตอนที่ 6 : การเก็บองค์ความรู้จากข้อความ (Knowledge-based storage) เป็นขั้นตอนการนำข้อมูลจากขั้นตอนที่ 5 มาเก็บไว้ในฐานข้อมูลชั่วคราว เพื่อจะรอการถ่ายโอนเข้าสู่องค์ความรู้ในขั้นตอนถัดไป

ขั้นตอนที่ 7 : การสร้างอินสแตนซ์ (Instances creation) เป็นการสร้างองค์ความรู้ขึ้นมาเพื่อรองรับกับข้อมูลต่างๆ ที่สกัดมาดได้จากรูปภาพและคำอธิบายรูปภาพ

ขั้นตอนที่ 8 : การแปลงโครงสร้างข้อมูลตัวอักษร (Text restructuring) เป็นการแปลงโครงสร้างข้อมูลจากฐานข้อมูลในขั้นตอนที่ 6 นำไปสู่โครงสร้างลำดับชั้นที่สร้างขึ้นในขั้นตอนที่ 7

ในหัวข้อถัดไปจะอธิบายถึงรายละเอียดในการประมวลผลรูปภาพเพื่อดึงเอาคุณลักษณะแบบ. โลคอลออกมา การประมวลผลคำอธิบายรูปภาพ และการออกแบบออนโทโลจี ตามลับดับ

3.1.1 **การวิเคราะห์และการดึงคุณลักษณะแบบโลคอลของรูปภาพ**

ในขั้นตอนนี้ประกอบด้วย 2 กระบวนการย่อยเพื่อประมวลผลและวิเคราะห์คุณลักษณะแบบโลคอลของรูปภาพ ขั้นตอนแรกคือ การดึงเอาคุณลักษณะของรูปภาพออกมาและเก็บในรูปแบบที่จะนำไปใช้ต่อไปเพื่อแปลความหมายได้ง่าย ขั้นตอนนี้จะเรียกว่า “การวิเคราะห์รูปภาพ (image

analysis)”ขั้นตอนที่สองคือ แปลความหมายของคุณลักษณะเหล่านั้นให้เป็นคอนเซฟท์ระดับสูง (High level conceptualization) ที่เกี่ยวกับภาพกีฬาต่างๆ และทำสร้างคำอธิบายประกอบ (annotation) เพื่อใช้ในการสืบค้นต่อไป

ในงานวิจัยนี้จะใช้เทคนิค BVW สำหรับการรู้จำวัตถุต่างๆ เพื่อช่วยในการจำแนกรูปภาพ ประโยชน์หลักของวิธีการนี้คือแก้ข้อจำกัดที่พบในคุณลักษณะแบบโกบอล เช่น ขนาดรูปไม่เท่ากัน ความสว่างในรูป การอิ่มตัวของสี หรือมุมกล้องที่ต่างกัน ทำให้การรู้จำมีประสิทธิภาพมากยิ่งขึ้น (Csurka et al., 2004) อย่างไรก็ตามวิธีการ BVW แบบเดิมมีข้อจำกัดอยู่หลายประการ ตัวอย่างเช่น ไม่มีเก็บรักษาความหมายที่อยู่ในรูปภาพเอาไว้ ดังนั้นในงานวิจัยนี้จะนำเสนอวิธีการเพื่อลดข้อจำกัดดังกล่าวเพื่อเพิ่มประสิทธิภาพการค้นคืนรูปภาพให้สูงขึ้น ซึ่งขั้นตอนการสร้าง BVW นี้จะประกอบด้วย 3 ขั้นตอนหลัก ๆ ดังต่อไปนี้

3.1.1.1 การสกัดคุณลักษณะระดับต่อของรูปภาพ

ขั้นตอนนี้เริ่มจากการใช้อัลกอริทึม Difference of Gaussian (DOG) (Lowe, 2004) ประมวลผลรูปภาพเพื่อทำการดึงเอาจุดสำคัญ (Keypoint) ของรูปภาพออกมาก และจุดสำคัญบนรูปภาพเหล่านี้ ข้อมูลเหล่านี้มีลักษณะเป็นเว็กเตอร์ของตัวเลขที่เป็นตัวอธิบายรูปภาพ และหลังจากนั้นข้อมูลเหล่านี้จะถูกแปลงเป็นข้อมูลที่เรียกว่า SIFT descriptor ดังนั้นผลลัพธ์สุดท้ายของขั้นตอนนี้คือ รูปภาพแต่ละรูปภาพจะถูกแทนด้วยข้อมูลเว็กเตอร์ที่มีมิติ (ไดเมนชั่น) ที่เท่ากันทุกรูปภาพ

3.1.1.2 การสร้างวิชวลเวิร์ด

คุณลักษณะของซิฟท์ที่เหมือนกันจะถูกจัดกลุ่มไว้ในกลุ่มเดียวกันโดยใช้เทคนิคการจัดกลุ่ม (Clustering algorithm) เช่น *k*-mean แต่ข้อเสียของ *k*-mean คือไม่คำนึงถึงตำแหน่งของจุดสำคัญต่างๆ บนรูปภาพหรือที่เรียกว่า “Spatial location” ซึ่งทำให้สูญแต่การเชื่อมโยงกันระหว่างความหมายของรูปภาพและคุณลักษณะของรูปภาพดังกล่าว ดังนั้นในงานวิจัยนี้จึงนำเสนอการใช้เทคนิคที่ชื่อว่า Semantic Local Adaptive Clustering (SLAC) (AlSumait and Domeniconi, 2008) แทน *k*-mean โดยเทคนิคดังกล่าวจะเก็บคุณลักษณะที่สัมพันธ์กันมาอยู่ในกลุ่มเดียวกัน แต่ละกลุ่มนี้จะเรียกว่า วิชวลเวิร์ด และวิชวลเวิร์ดเหล่านี้ จะถูกนำไปสร้าง BVW ที่มีโครงสร้างในรูปแบบเว็กเตอร์ต่อไป การนำคุณลักษณะของรูปที่สัมพันธ์กันมาไว้ในกลุ่มเดียวกัน ทำให้คุณภาพของวิชวลเวิร์ดที่ได้ดีขึ้น และใช้เป็นตัวแทนของรูปภาพได้ดียิ่งขึ้นนั่นเอง

3.1.1.3 การแปลงโครงสร้างของวิชวลเวิร์ด

นักวิจัยหลายท่านได้พยายามลดความกำกวมของวิชวลเวิร์ด โดยการแปลงโครงสร้างของวิชวลเวิร์ดจากโครงสร้างแบบเว็กเตอร์เป็นโครงสร้างแบบลำดับชั้น (Jiang and Ngo, 2009; Sivic et al., 2008; Wang et al., 2009) โดยอัลกอริทึมในการจัดกลุ่ม ตัวอย่างเช่น Hierarchical Spatial Markov model (Wang et al., 2009) Agglomerative clustering (Walter et al., 2008) และ Hierarchical Latent Dirichlet Allocation algorithm หรือ hLDA (Blei et al., 2003). อย่างไรก็ตาม วิธีการเหล่านี้ยังมีข้อจำกัดบางประการซึ่งมีผลต่อการแปลความหมายของรูปภาพ สรุปได้ดังต่อไปนี้

1) โครงสร้างลำดับชั้นที่ได้จากอัลกอริทึมเหล่านี้ ส่วนใหญ่มีโครงสร้างเป็นต้นไม้ไบนารี่ (Binary tree) เท่านั้น แต่ในความเป็นจริงแล้ว ข้อมูลที่อยู่ในรูปภาพอาจจะมีความสัมพันธ์กันมากกว่าโครงสร้างแบบไบนารี่ก็ได้ ซึ่งทำให้มีผลเสียต่อการแทนข้อมูลในรูปภาพไม่ถูกต้อง
2) ไม่มีการสืบทอดคุณสมบัติระหว่างโหนดแม่และโหนดลูกในลักษณะ Multiple-inhertitance คือการที่โหนดลูกสามารถสืบทอดคุณสมบัติมาจากโหนดแม่หลายๆ โหนด แต่ในความเป็นจริงแล้วการสืบทอดคุณสมบัติในลักษณะดังกล่าวสามารถเกิดขึ้นได้ ตัวอย่างเช่น ไตรกีฬาสามารถสืบทอดคุณสมบัติมาจากกีฬาประเภทลู่และกีฬาประเภทลาน เนื่องจากไตรกีฬาเป็นการผสมกีฬาทั้งสองประเภทไว้ด้วยกัน ดังนั้นการใช้โครงสร้างแบบไบนารี่จึงไม่สามารถใช้แทนข้อมูลในรูปภาพได้อย่างมีประสิทธิภาพเท่าที่ควร

ดังนั้นในงานวิจัยนี้จึงนำเสนอแนวคิดใหม่ในการใช้แปลงโครงสร้างของวิชวลเวิร์ดซึ่งจะอธิบายได้ดังต่อไปนี้

ในขั้นแรกของการสอนระบบ วัตถุที่สนใจในรูปภาพจะถูกแยกออกจากพื้นหลังโดยใช้คนทำ ทั้งนี้เพื่อให้ได้ข้อมูลของวัตถุที่เราสนใจจริงๆ และกำจัดข้อมูลรบกวน (noise) ออกเพื่อจุดประสงค์ในการสอนระบบให้รู้จำข้อมูลของวัตถุต่างๆ เช่น ลูกบอล ลูกเหล็ก ไม้คาน ไม้คำถ่อ หรือเบาะรอง เป็นต้น ผลที่ได้ของการแยกวัตถุที่สนใจออกมานี้ทำให้เราได้ข้อมูลชุดหนึ่งของแต่ละวัตถุ และข้อมูลแต่ละชุดนั้นมีความเกี่ยวข้องกันเชิงความหมาย (Semantically relevant) เนื่องจากเป็นข้อมูลรูปภาพของวัตถุเดียวกันนั่นเอง หลักจากนั้นคุณลักษณะของแต่ละวัตถุ (C) แต่ละชุดจะถูกนำไปสร้าง BVW (ϖ) ซึ่ง ($\varpi_i \in C_i$) เมื่อเราได้ BVW ของแต่ละวัตถุมาแล้วจะนำมาคำนวณค่าที่เรียกว่า “Concept

range” ซึ่งเป็นเทคนิคที่ใช้ในการนำข้อมูลเข้าออนโทโลจีหรือโครงสร้างลำดับชั้นสำหรับข้อมูลรูปภาพนั่นเอง

กำหนดให้รัศมี (r_i) ของคอนเซฟท์ i หมายถึงระยะทางระหว่างจุดศูนย์กลางของคอนเซฟท์ (c) และจุดศูนย์กลาง (v) ของวิชวลเวิร์ด (ϖ) ที่อยู่ไกลที่สุดของวัตถุนั้นๆ ซึ่งสามารถเขียนเป็นสมการได้ดังที่ 5

$$r_i = \max|v - c|, v \in \varpi \quad (5)$$

ข้อดีของวิธีการนี้คือ จะสามารถช่วยลดความกำกวมของวิชวลเวิร์ดได้และช่วยให้การจำแนกรูปภาพมีประสิทธิภาพมากขึ้น นอกจากนี้วิธีการนี้ยังแก้ปัญหาของต้นไม้ไบนารีได้ด้วยคือสามารถสืบทอดคุณลักษณะมาจากโหนดแม่ได้มากกว่าหนึ่งโหนด เนื่องจากค่าของวิชวลเวิร์ดสามารถอยู่ในช่วงของรัศมีของคอนเซฟท์ได้มากกว่าหนึ่งคอนเซฟท์นั่นเอง และในขณะเดียวกันวิชวลเวิร์ดบางอันก็จะถูกทิ้งไปหากค่าจุดศูนย์กลางของวิชวลเวิร์ดนั่นไม่อยู่ในรัศมีของคอนเซฟท์ใดๆ เลย ซึ่งแนวคิดนี้สอดรับกับความจริงที่ว่า คำนึงคำ (วิชวลเวริ์ด) สามารถมีได้หลายความหมาย (Polysemy) ด้วยเช่นกัน ตัวอย่างเช่น วิชวลเวิร์ดหนึ่งอันอาจจะอยู่ในคอนเซฟท์ของไม้ค้ำถ่อ (Pole) กับไม้คานกระโดดสูง (Horizontal bar) ได้ เนื่องจากวัตถุทั้ง 2 ชนิดมีรูปร่างที่เหมือนกัน

การรู้จำวัตถุต่างๆ ในรูปภาพนั้นจะใช้หลักการของฮิสโตแกมความถี่ของวิชวลเวิร์ดเป็นตัวแทนของวัตถุต่างๆ โดยถ้าความถี่ของวิชวลเวิร์ด ($f(v_i)$) มีค่ามากกว่าค่าที่กำหนด (threshold) ซึ่งได้จาค่าเฉลี่ยที่ทดลองซ้ำๆ หลายๆ ครั้ง ระบบจะตีความว่ามีวัตถุดังกล่าวอยู่ในรูปภาพนั้นๆ แต่เนื่องจากรูปภาพแต่ละรูปภาพมีขนาดไม่เท่ากัน ดังนั้นการพิจารณาเฉพาะความถี่อย่างเดียวอาจจะทำให้เกิดความผิดพลาดในการตีความ ดังนั้นผู้วิจัยจึงทำการนอร์มอไรซ์ โดยนำความถี่ของวิชวลเวิร์ดทั้งหมดของทุกรูปภาพในระบบมาหาร แสดงสูตรการคำนวรได้ดังสมการที่ 6

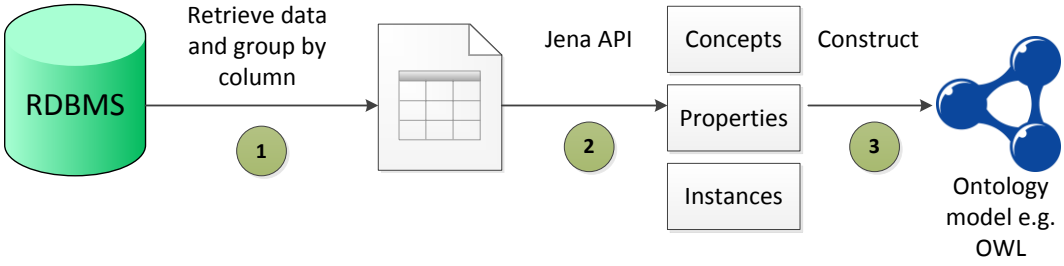
$$n_i = f(v_i) / \sum_{j=0}^{N} f(v_j) \quad (6)$$

3.1.2 การวิเคราะห์และสกัดสารสนเทศจากคำอธิบายรูปภาพ

ในหัวข้อนี้จะอธิบายถึงการประมวลผลข้อความหรือคำอธิบายรูปภาพซึ่งมีลักษณะข้อมูลเป็นตัวอักษร ซึ่งเป็นการทำงานหลักของระบบส่วนหนึ่งในรูปที่ 19 ขั้นตอนที่ 5-8 ซึ่งจะเรียกส่วนนี้ว่า “ส่วนวิเคราะห์ข้อความ (Textual Information Analysis-TIA)” แนวคิดในขั้นตอนนี้เกิดมาจากความจริงที่ว่า ข้อความอธิบายรูปภาพส่วนใหญ่เป็นคำพูดที่มนุษย์เขียนขึ้นเพื่อพยายามถึงความหมายที่อยู่ในรูปภาพ ดังนั้นข้อความเหล่านี้จึงมีความน่าเชื่อถือและควรจะถูกนำไปใช้ให้เป็นประโยชน์เพื่อตีความหมายของรูปภาพ ไม่ควรจะถูกตัดทิ้งไป ในส่วนของ TIA นี้ จะประกอบด้วยขั้นตอนการทำงานดังต่อไปนี้

1) ข้อความที่ติดมากับรูปภาพจะถูกถอดแท็ก HTML และถูกประมวลผลโดยใช้อัลกอริทึมการประมวลผลภาษาธรรมชาติ (Natural Language Processing-NLP) เพื่อสกัดเอาข้อมูลสำคัญที่อยู่ในรูปภาพ ได้แก่ วัน เวลา ชื่อคน ชื่อสถานที่ หรือชื่อกีฬา เป็นต้น ในงานวิจัยนี้จะจำกัดอยู่ที่คำอธิบายรูปภาพที่เป็นภาษาอังกฤษเท่านั้น เนื่องจากอัลกอริทึม NLP ไม่ได้อยู่ในขอบเขตของงานวิจัยนี้ ดังนั้นผู้วิจัยจึงนำเทคนิค NLP ซึ่งมีผู้ทำไว้แล้วเรียกว่า ESpotter (Zhu et al., 2005) มาใช้งาน ซึ่งความสามารถของ ESpotter สามารถรู้จำชื่อสำคัญต่างๆ (Name Entity Recognition) ที่ระบบต้องการได้เป็นอย่างดี ผลลัพธ์ที่ได้จากขั้นตอนนี้ คือเมต้าเดต้าอธิบายรูปภาพ (Annotation) ซึ่งอยู่ในรูปแบบของ XML ซึ่งจะถูกนไปประมวลผลต่อไป

2) เมต้าเดต้าที่ได้จากขั้นตอนก่อนหน้าจะถูกนำมาประมวลผลต่อเพื่อใช้แทนความหมายที่อยู่ในรูปภาพได้อย่างมีประสิทธิภาพ เมต้าเดต้าเหล่านี้จะถูกนำไปจัดเก็บไว้ในองค์ความรู้หรือออนโทโลจีร่วมกับคุณลักษณะของรูปภาพที่อธิบายไว้ในขั้นตอนที่ 3.1.1 เมต้าเดต้าที่ได้จากขั้นตอนก่อนหน้าจะถูกนำมาถอดแท็ก XML ออกและถูกนำไปใส่เก็บไว้ในฐานข้อมูลและหลังจากนั้นข้อมูลเหล่านี้จะถูกนำไปใส่ในออนโทโลจี ซึ่งเก็บในลักษณะ OWL โดยใช้ JDBC connector กระบวนการในการแปลงจากฐานข้อมูลเป็นออนโทโลจีแสดงได้ดังรูปที่ 20



รูปที่ 20 กระบวนการในการแปลงข้อมูลจากฐานข้อมูลให้อยู่ในรูปแบบออนโทโลจี

จากรูปที่ 20 สามารถอธิบายขั้นตอนต่างๆ ได้ดังต่อไปนี้

- ขั้นตอนที่ 1 ข้อมูลที่อยู่ในฐานข้อมูล (RDBMS) จะถูกค้นคืนโดยใช้ภาษา SQL (Structure Query Language)
- ขั้นตอนที่ 2 ออนโทโลจีจะถูกสร้างขึ้นโดยการสร้าง Class Property และ Instance ต่างๆ สำหรับข้อมูลที่ดึงมาจากฐานข้อมูล ในขั้นตอนนี้จะใช้ Jena API (http://jena.apache.org) พัฒนาโดยภาษาจาวาในการเชื่อมโยงฐานข้อมูลและดึงข้อมูลมาใส่ในออนโทโลจี
- ขั้นตอนที่ 3 ข้อมูลจากฐานข้อมูลจะถูกนำไปใส่ในออนโทโลจีตาม Class Property และ Instance ต่างๆ ที่สร้างขึ้นและเก็บในรูปแบบ OWL

สำหรับโครงสร้างขององค์ความรู้ที่สร้างขึ้นสำหรับเก็บข้อมูลรูปภาพเกี่ยวกับกีฬานั้น จะอธิบายในหัวข้อถัดไป

3.1.3 **การออกแบบองค์ความรู้**

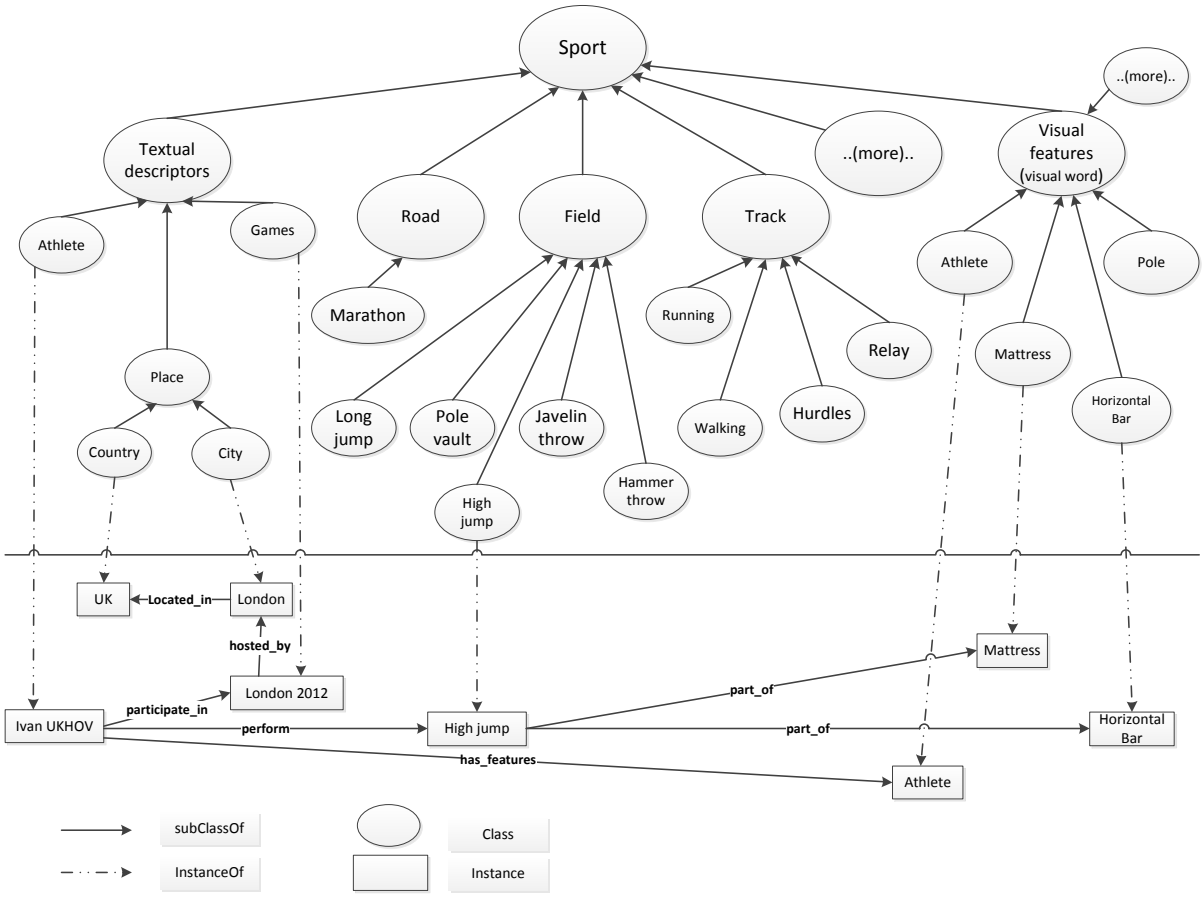
การออกแบบออนโทโลจีในงานวิจัยนี้มีความท้าทายอย่างยิ่ง เนื่องจากเป็นออนโทโลจีที่มีการเก็บข้อมูล 2 ประเภท ได้แก่ ข้อมูลที่เป็นคุณลักษณะของรูปภาพ (Visual features) และข้อมูลที่เป็นตัวอักษร (Textual features) ดังนั้นออนโทโลจีดังกล่าวจึงต้องมีการออกแบบเพื่อรองรับกับข้อมูลดังกล่าวได้อย่างมีประสิทธิภาพ ผู้วิจัยได้พัฒนาออนโทโลจีซึ่งประกอบด้วย 3 ส่วนย่อยๆ ได้แก่ 1) ออนโทโลจีสำหรับเก็บข้อมูลกีฬาต่างๆ 2) ออนโทโลจีสำหรับเก็บข้อมูลตัวอักษร และ 3) ออนโทโลจีสำหรับเก็บข้อมูลคุณลักษณะรูปภาพ ซึ่งสามารถอธิบายรายละเอียดได้ดังต่อไปนี้

(1) ออนโทโลจีข้อมูลกีฬา : ออนโทโลจีจะจัดเก็บข้อมูลกีฬาประเภทต่างๆ คำศัพท์ที่เกี่ยวข้องกับกีฬานั้นๆ รวมถึงกฎระเบียบ กติกาการเล่น อุปกรณ์ที่ใช้เล่น เป็นต้น ซึ่งข้อมูลเหล่านี้ถูกรวบรวมมาจากเว็บไซต์โอลิมปิค (www.olympic.org)

(2) ออนโทโลจีสำหรับตัวอักษร : เก็บข้อมูลคำอธิบายรูปภาพต่างๆ ในลักษณะข้อความ เช่น นักกีฬา ชื่อกีฬา สถานที่จัดกีฬา เมือง ประเทศ วัน และเวลา เป็นต้น

(3) ออนโทโลจีสำหรับคุณลักษณะรูปภาพ : เก็บคุณลักษณะของรูปภาพต่างๆ เช่น BVW รูปแบบไฟล์ ความละเอียดของรูปภาพ เป็นต้น

ข้อมูลเหล่านี้จะถูกนำมาใช้ร่วมกันในการเปรียบเทียบความคล้ายคลึงของรูปภาพต้นแบบที่ผู้ใช้ต้องการค้นหา โครงสร้างของออโทโลจีโดยย่อสแดงได้ดังรูปที่ 21



รูปที่ 21 โครงสร้างของออนโทโลจีในงานวิจัย

3.2 ความสมบูรณ์ของออนโทโลจี

เป็นที่ทราบกันดีในหมู่ของนักวิจัยเกี่ยวกับออนโทโลจีว่า เป็นเรื่องยากที่จะทำให้ออนโทโลจีมีความสมบูรณ์ครอบคลุมข้อมูลที่สนใจภายในโดเมนได้ทุกอย่าง ดังนั้นงานวิจัยนี้จึงพยายามลดข้อจำกัดดังกล่าวโดยการทำดัชนี (Index) ข้อมูลสำรอง เพื่อไม่ให้การทำงานของระบบขึ้นอยู่กับออน-โทโลจีเพียงอย่างเดียว ในงานวิจัยนี้จึงนำวิธีการทำดัชนีที่เรียกว่า Latent Semantic Indexing (LSI) มาใช้ แต่นำมาใช้กับข้อมูลที่เป็นตัวอักษรเพียงเท่านั้น เมื่อระบบไม่สามารถหาข้อมูลในออนโทโลจีพบ ระบบจะมาค้นหาข้อมูลใน LSI อีกครั้ง เพื่อลดข้อจำกัดของออนโทโลจีที่ไม่สามารถเก็บข้อมูลได้อย่างครบถ้วนนั่นเอง

คุณสมบัติของ LSI มีหลายประการ เช่น สามารถลดความกำกวมของคำและสามารถรองรับการค้นหาข้อมูลเชิงความหมายได้อย่างมีประสิทธิภาพเช่นกัน โดย LSI จะพิจารณาจากค่าสถิติที่

คำนวณได้จากความสัมพันธ์ของคำต่างๆ ในระบบ การทำงานของ LSI สามารถอ่านรายละเอียดได้ในหัวข้อ 2.6

3.3 ขอบเขตของการวิจัย

ในงานวิจัยนี้จะทดลองกับรูปภาพจำนวน 20,000 รูปภาพ ซึ่งเป็นรูปภาพเกี่ยวกับกีฬาประเภทต่างๆ ที่รวบรวมจากเว็บไซต์ต่างๆ และ Google เหตุผลที่ต้องรวบรูปภาพขึ้นมาเองเนื่องจากยังไม่มีองค์กรใดทำชุดรูปภาพมาตรฐานสำหรับทดสอบการค้นคืนรูปภาพกีฬาขึ้นมา และในการวิจัยนี้จะทดสอบกับรูปภาพกีฬาทั้งหมด 20 ประเภท ได้แก่

ประเภทกีฬา	ประเภทกีฬา
(1) แบดมินตัน	(11) ค้ำถ่อ
(2) ชกมวย	(12) วิ่งแข่ง
(3) ปั่นจักรยาน	(13) เรือใบ
(4) ขว้างจักร	(14) ยิงปืน
(5) พุ่งแหลม	(15) เทนนิส
(6) ฟุตบอล	(16) วอลเล่ย์บอล
(7) ขว้างค้อน	(17) ยกน้ำหนัก
(8) กระโดดสูง	(18) กระโดดน้ำ
(9) กระโดดไกล	(19) ยูโด
(10) วิ่งข้ามรั้ว	(20) วิ่งผลัด

ข้อมูลทั้งหมดจะถูกแบ่งออกเป็น 2 กลุ่ม ได้แก่ กลุ่มที่ 1 จำนวน 12,000 รูปภาพ (600 รูปจากแต่ละประเภทกีฬา) ใช้สำหรับสอนระบบ (Training set) และกลุ่มที่ 2 จำนวน 8,000 รูปภาพ (400 รูปจากแต่ละประเภทกีฬา) ใช้เพื่อทดสอบระบบ (Testing set)

จากระเบียบวิธีวิจัยที่อธิบายมาทั้งหมดในบทนี้ แนวคิดจะถูกนำไปทดสอบประสิทธิภาพการทำงานของระบบในด้านต่างๆ เช่น ความถูกต้องของการค้นคืนรูปภาพและความน่าเชื่อถือของระบบ ในบทถัดไปจะอธิบายถึงการทดลองและอภิปรายผลการทดลองที่ได้

บทที่ 4
ผลการวิจัยและการอภิปรายผล

บทนี้จะอธิบายถึงผลการวิจัยตามแนวคิดที่นำเสนอไว้ในบทที่ 3 และวิเคราะห์ผลการวิจัยเชิงลึก การทดสอบระบบถูกออกแบบเพื่อทดสอบระบบตามประเด็นต่างๆ ดังต่อไปนี้

(1) **ความถูกต้องของการค้นหาข้อมูล (Correctness)** คือการทดสอบว่าระบบสามารถค้นหาข้อมูลได้ถูกต้องเพียงใดตามขอบเขตขององค์ความรู้ (Domain of knowledge) ที่ได้ระบุไว้ ซึ่งจะวัดจากประสิทธิภาพการค้นหาข้อมูลเชิงความหมาย (Semantic search) การทดสอบนี้หมายถึงการที่ระบบสามารถหารูปภาพที่เกี่ยวข้องกับคิวรี่ (Query) ได้อย่างมีประสิทธิภาพถึงแม้ว่ารูปภาพนั้นอาจจะไม่มีคำอธิบายที่ตรงกับคิวรี่เลยก็ตาม

(2) **การค้นหาข้อมูลที่หลากหลาย (Multimodality)** เนื่องออนโทโลจีที่ออกแบบในงานวิจัยนี้เก็บข้อมูลที่ต่างชนิดกัน 2 รูปแบบ ดังนั้นระบบต้องสามารถรองรับการค้นหาข้อมูลทั้ง 2 รูปแบบได้อย่างมีประสิทธิภาพ

(3) **ความน่าเชื่อถือของระบบการค้นหา (Robustness)** หมายถึงการที่ระบบจัดการกับคิวรี่ที่ไม่สามารถหาคำตอบได้อย่างมีประสิทธิภาพ

4.1 สมมติฐานการวิจัย

จากประเด็นการทดสอบทั้ง 3 ประเด็นดังกล่าว ผู้วิจัยได้ตั้งสมมติฐานการวิจัยขึ้นมาให้สอดคล้องกับประเด็นการทดสอบระบบดังต่อไปนี้

สมมติฐานข้อแรกเน้นไปที่การวัดประสิทธิภาพการออกแบบออนโทโลจี ถ้าโครงสร้างของออนโทโลจีถูกออกแบบไว้อย่างเหมาะสม จะช่วยให้เพิ่มประสิทธิภาพการค้นหาข้อมูลทั้งแบบปกติและเชิงความหมายได้

สมมติฐานที่ 1: ออนโทโลจีโดยทั่วไปจะเก็บข้อมูลประเภทเดียวเท่านั้น ดังนั้นแนวคิดการเก็บข้อมูลทั้ง 2 ประเภทไว้ในออนโทโลจีเดียวกันจะช่วยให้ระบบสารมารถรองรับกับคิวรี่ทั้งแบบที่ใช้ตัวอักษรหรือข้อความในการค้นหาและคิวรี่ที่ใช้คุณลักษณะของรูปภาพในการค้นหาได้

สมมติฐานที่ 2: โครงสร้างของออนโทโลจีที่ได้ออกแบบไว้ สามารถทำให้ระบบค้นหาข้อมูลเชิงความหมายได้อย่างมีประสิทธิภาพ เช่น ค้นหาข้อมูลที่ไม่มีคำอธิบายตรงกับคิวรี่ แต่คำอธิบายเหล่านั้นเกี่ยวข้องกับคิวรี่เชิงความหมาย

สมมติฐานที่ 3: การออกแบบออนโทโลจีเป็นเรื่องที่มีความซับซ้อนสูง ส่งผลให้การเก็บข้อมูลในออนโทโลจีให้ครอบคลุมทุกอย่างในโดเมนที่สนใจเป็นเรื่องที่เป็นไปได้ยาก และต้องใช้เวลาในการปรับแต่งออนโทโลจีนาน ดังนั้นหากระบบไม่ขึ้นอยู่กับข้อมูลในออนโทโลยีเพียงอย่างเดียว แต่ควรมีระบบสำรองในการทำดัชนีเพื่อค้นหาข้อมูลที่ไม่สามารถใช้โครงสร้างของออนโทโลจีหาได้

4.2 การประเมินประสิทธิภาพของระบบ

วิธีการประเมินประสิทธิภาพการค้นคืนข้อมูลแบบพื้นฐานและเป็นที่นิยมใช้กันอย่างแพร่หลาย โดยใช้ค่าสองค่าคือ “พรีซิชั่น (Precision)” และ “รีคอล (Recall)” โดยที่พรีซิชั่นจะบอกถึงประสิทธิภาพของระบบในการค้นคืนเอกสารโดยดูจากอัตราส่วนของจำนวนเอกสารที่ถูกต้องกี่เอกสารจากเอกสารที่ถูกเลือกมาทั้งหมด (สมการที่ 7) ส่วนรีคอลหมายถึง ประสิทธิภาพของการค้นคืนเอกสารโดยดูจากอัตราส่วนจำนวนเอกสารที่ถูกต้องที่เลือกมาต่อจำนวนเอกสารที่ถูกต้องทั้งหมดที่อยู่ในคอลเล็คชั่น (สมการที่ 8)

$$\text{Precision} = \frac{no.of\ \ relevant\ document\ retrieved}{no.of\ \ retreived\ documents} \qquad (7)$$

$$\text{Recall} = \frac{no.of\ \ relevant\ document\ retrieved}{no.of\ \ \text{all}\ relevant\ documents\ in\ collect\ ions} \qquad (8)$$

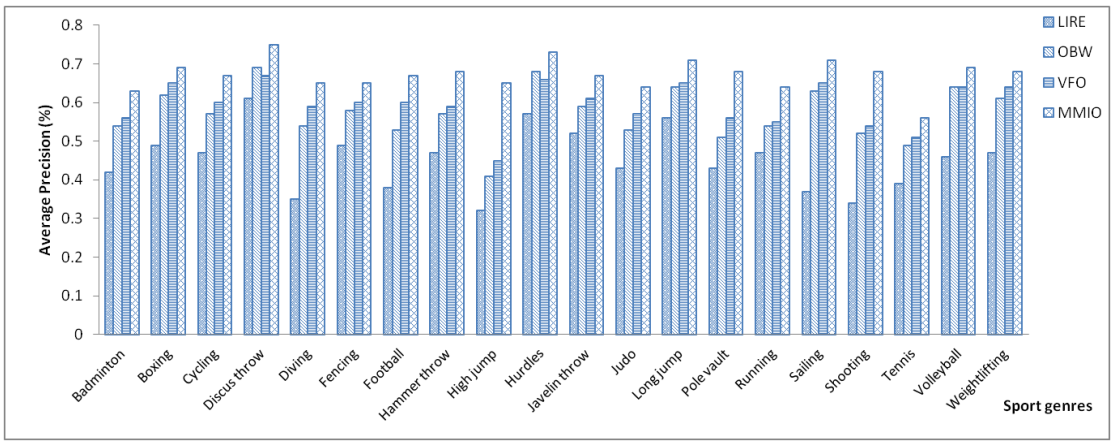
ดังนั้นในงานวิจัยนี้จะวัดประสิทธิภาพของระบบโดยดูจากค่าพรีซิชั่นและรีคอลเป็นหลัก โดยการทดลองจะแบ่งออกเป็น 4 การทดลองย่อย ดังต่อไปนี้ 1) การทดลองเพื่อประเมินประสิทธิภาพ

การค้นหาข้อมูลเชิงความหมาย 2) การทดลองเพื่อทดสอบความไม่สมบูรณ์ของออนโทโลจี 3) การทดลองเพื่อเปรียบเทียบประสิทธิภาพการค้นหากับระบบค้นหาข้อมูลที่มีอยู่ในปัจจุบัน

4.3 การทดลองที่ 1: การทดลองเพื่อทดสอบการใช้คิวรี่ในรูปแบบข้อความและรูปภาพ

ในการทดลองนี้จะทำการส่งคิวรี่ไปยังระบบ และประเมินผลลัพธ์ที่ได้กลับมาจากระบบว่ามีความถูกต้องมากน้อยเพียงใด โดยคิวรี่ที่ส่งไปจะมี 2 รูปแบบคือ 1) แบบข้อความ (Textual query) และ 2) แบบรูปภาพ (Visual query) จะทำการส่งคิวรี่ไปยังระบบทั้งหมด 10 ครั้งในแต่ละรูปแบบ แต่ละครั้งจะใช้คิวรี่ที่แตกต่างกันออกไป และนำค่าพรีซิชั่นและรีคอลมาหาค่าเฉลี่ย โดยระบบที่นำเสนอในงานวิจัยนี่จะถูกเรียกว่า “MMIO-Multi-Modal Incompleteness Ontology” และระบบที่นำมาเปรียบเทียบประสิทธิภาพการค้นคืนข้อมูลได้แก่ LSI (Deerwester et al., 1990) Original BVW (OBW) (Csurka et al., 2004) LIRE (www.semanticmetatdata.net/LIRE) Lucene (http://lucene.apache.org) และ Visual Features Ontology (VFO) ซึ่งเป็นออนโทโลจีที่เก็บเฉพาะคุณลักษณะของรูปภาพอย่างเดียว ไม่เก็บข้อความที่เป็นตัวอักษรใดๆ ทั้งสิ้นแต่มีโครงสร้างในลักษณะของต้นไม้ไบนารี ตรงกันข้ามกับ OBW นั้นหมายถึงเทคนิค bag-of-word แบบเดิมที่อยู่ในรูปแบบเว็กเตอร์และไม่มีการแปลงโครงสร้างเป็นโครงสร้างลำดับชั้น สำหรับ LIRE เป็นกรอบแนวคิดที่ทำการค้นคืนข้อมูลรูปภาพบนพื้นฐานของสีและพื้นผิวที่อยู่ในรูปภาพ หรือเป็นการใช้คุณลักษณะแบบโกบอลในการเปรียบเทียบความคล้ายคลึงนั่นเอง ส่วน Lucene นั้นเป็นกรอบแนวคิดที่ทำการค้นหาข้อมูลโดยใช้ข้อความหรือคิวรี่ที่เป็นตัวอักษร

โดยในการทดลองในแบบแรกคือการทดลองโดยใชคิวรี่เป็นรูปภาพ โดยทำการเปรียบเทียบกับเทคนิคอื่นๆ ได้แก่ OBW, LIRE, VFO และ MMIO ผลการทดสองแสดงได้ดังรูปที่ 22

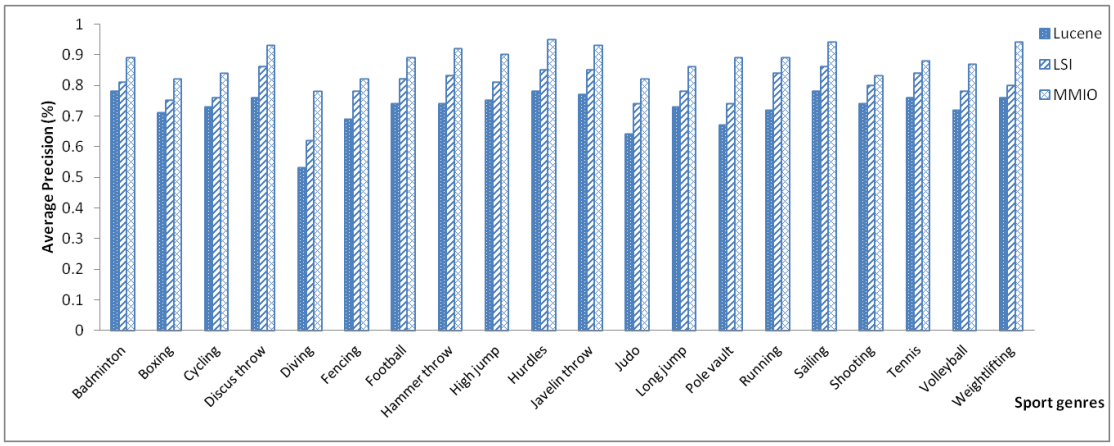


รูปที่ 22 เปรียบเทียบค่าพรีซิชั่นเฉลี่ยของคิวรี่รูปภาพ

จากรูปที่ 22 จะเห็นว่า LIRE ให้ประสิทธิภาพในการค้นหาต่ำที่สุดเนื่องจากใช้คุณลักษณะแบบโกบอล ตามที่ได้อธิบายไว้แล้วในบทก่อนหน้านี่ว่า คุณลักษณะโกบอลมีข้อจำกัดหลายประการ เช่น ขนาดรูปภาพ ความสว่าง การหมุนภาพ หรือมุมกล้องที่แตกต่างกัน ก็ทำให้ระบบไม่สามารถค้นคืนภาพได้อย่างถูกต้อง ในขณะที่อีก 3 วิธีใช้คุณลักษณะแบบโลคอลเหมือนกันทั้งหมด แต่มีโครงสร้างของข้อมูลที่ไม่เหมือนกันตามที่นำเสนอข้างต้น OBW มีประสิทธิภาพที่ต่ำกว่า VFO และ MMIO เนื่องจากมีโครงสร้างการเก็บข้อมูลในลักษณะเว็กเตอร์ ซึ่งไม่การเก็บข้อมูลเชิงความหมายของข้อมูล ส่งผลให้การค้นคืนข้อมูลมีประสิทธิภาพต่ำ โดยรูปภาพที่เกี่ยวข้องเชิงความหมาย (สีหรือพื้นผิวของภาพไม่เหมือนคิวรี่) จะไม่ถูกพิจารณามาเป็นคำตอบนั่นเอง และในบางครั้งรูปที่ไม่เกี่ยวข้องก็ถูกนำมาเป็นคำตอบเนื่องจากมีคุณลักษณะแบบโลคอลที่ตรงกับคิวรี่นั่นเอง ทำให้ค่าพรีซิชั่นต่ำลง สำหรับ VFO มีประสิทธิภาพที่ต่ำกว่า MMIO ถึงแม้ว่าจะมีโครงสร้างลำดับชั้นเหมือนกัน เหตุผลสำคัญเนื่องจาก VFO มีโครงสร้างที่เป็นไบนารี ทำให้ไม่สามารถใช้สื่อข้อมูลในรูปภาพได้อย่างมีประสิทธิภาพ โครงสร้างของ MMIO ที่ออกแบบไว้ช่วยให้ระบบสามารถวิเคราะห์ความเกี่ยวข้องกันของรูปภาพต่างๆ ได้อย่างมีประสิทธิภาพ และระบบสามารถสกัดรูปภาพที่ไม่เกี่ยวข้องทิ้งไปได้มากกว่า จึงทำให้ MMIO ได้ค่าพรีซิชั่นสูงกว่าวิธีการอื่นๆ ที่นำมาเปรียบเทียบในการทดลองนี้

สำหรับคิวรี่ที่เป็นข้อความหรือตัวอักษรนั้น ในการทดลองนี้จะมีการเปรียบเทียบ 3 วิธี ได้แก่ Lucene, LSI และ MMIO รูปที่ 23 เปรียบเทียบผลการค้นหาข้อมูลของทั้ง 3 วิธี จะเห็นว่า MMIO เพิ่มประสิทธิภาพการค้นหาข้อมูลอย่างเห็นได้ชัด เนื่องจากโครงสร้างออนโทโลจีของ MMIO ช่วยในการค้นหาข้อมูลที่สัมพันธ์กันได้อย่างมีประสิทธิภาพมากกว่า Lucene ซึ่งทำการค้นหาโดยใช้เพียงเทคนิคของการเปรียบเทียบตัวอักษรเท่านั้น (String matching) สำหรับ LSI ได้ผลลัพธ์ที่ดีกว่า Lucene แต่ก็ยังไม่เท่ากับ MMIO ทั้งนี้เนื่องจาก LSI ทำงานโดยใช้การคำนวณทางสถิติเท่านั้น ดังนั้น

ข้อมูลหรือรูปภาพบางรูปอาจจะถูกคำนวณว่าเกี่ยวข้องกันในทางสถิติ แต่ในความเป็นจริงแล้วอาจจะไม่เกี่ยวข้องกันเลยก็ได้ ดังนั้นทำให้ค่าพรีซิชั่นของ LSI จึงต่ำกว่า MMIO นั่นเอง



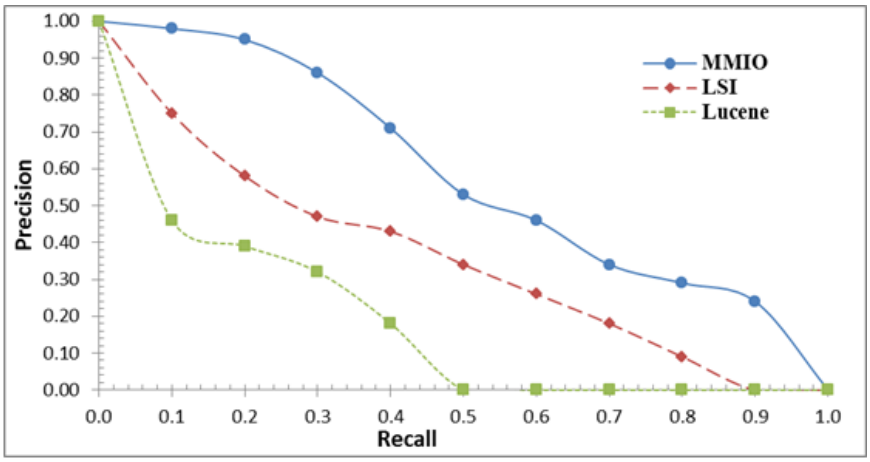
รูปที่ 23 เปรียบเทียบค่าพรีซิชั่นเฉลี่ยของคิวรี่ข้อความ

4.4 การทดลองที่ 2: การทดลองเพื่อประเมินประสิทธิภาพการค้นหาข้อมูลเชิงความหมาย

ในการทดลองมีจุดประสงค์เพื่อทดสอบการวิเคราะห์คิวรี่จากผู้ใช้ที่ซับซ้อนได้อย่างมีประสิทธิภาพ ตัวอย่างเช่น คิวรี่ “Thailand athletes who participate at the Olympic games in Great Britain” ถูกใช้ทดสอบเพื่อวัดประสิทธิภาพของระบบและนำมาเปรียบเทียบกับ LSI และ Lucene รูปที่ 24 แสดงให้เห็นว่า MMIO มีประสิทธิภาพการค้นคืนสารสนเทศที่สูงกว่า Lucene และ LSI เนื่องจาก Lucene ไม่สามารถแยกแยะความแตกต่างระหว่างคิวรี่ “*Thailand athletes participate at the Olympic games in Great Britain*” และคิวรี่ “*Great Britain athletes participate at the Olympic games in Thailand*” เนื่องจาก Lucene พิจารณาจากเอกสารที่เกี่ยวข้องโดยดูจากคำสำคัญ (Keyword) ต่างๆ ที่ปรากฏในเอกสารและเปรียบคำสำคัญในคิวรี่และคำในเอกสารว่าตรงกันหรือไม่ ทั้งนี้จะเห็นว่าทั้งสองคิวรี่นั้นมีคำสำคัญที่เหมือนกันคือ Thailand, Athletes, Olympic games, และ Great Britain ซึ่งทำให้เอกสารทั้งหมดที่มีคำเหล่านี้อยู่ถูกดึงมาเป็นผลลัพธ์ทั้งที่เกี่ยวข้องและไม่เกี่ยวข้อง ดังนั้นจึงทำให้ Lucene มีประสิทธิภาพการค้นคืนสารเทศ (พรีซิชั่น) ที่ต่ำกว่า LSI และ MMIO

ในขณะที่ LSI ใช้การคำนวณทางสถิติเพื่อหาเอกสารที่เกี่ยวข้อง ดังนั้นจึงทำให้เอกสารบางเอกสารที่เกี่ยวข้อง แต่ไม่ปรากฏคำในคิวรี่ยังสามารถถูกดึงมาเป็นผลลัพธ์ของการค้นหาได้ ส่งผลให้ LSI มีค่าพรีซิชั่นที่สูงกว่า Lucene แต่ยังต่ำกว่า MMIO ทั้งนี้เนื่องจาก MMIO มีการค้นหาข้อมูลโดยพิจารณาถึงความสัมพันธ์ของข้อมูลด้วยโดยใช้ภาษา SPARQL โดยที่คิวรี่ที่ผู้ใช้ใส่เข้ามาในระบบจะถูก

แปลงเป็นภาษา SPARQL ในตัวอย่างนี้ได้แก่ “Great Britain athletes”-<*participate*>-“Olympic games”-<*located*>-“Thailand” ดังนั้นเอกสารใดๆ ที่ปรากฏความสัมพันธ์ดังกล่าวจะถูกดึงมาเป็นคำตอบทั้งหมด และระบบจะไม่สนใจเอกสารใดๆ ที่มีความสำพันธ์ไม่ตรงกับคิวรี่ ถึงแม้จะมีคำสำคัญตรงกับคิวรี่ทุกคำ ตัวอย่างเช่น ระบบจะไม่สนใจเอกสารที่มีความสัมพันธ์ “Thailand athletes”-<*participate*>-“Olympic games”-<*located*>-“Great Britain” ซึ่งจะเห็นว่าทั้งสองตัวอย่างนี่มีคำสำคัญที่เหมือนกันทุกคำ แต่มีความหมายต่างกันโดยสิ้นเชิง ด้วยกลไกลดังกล่าวทำให้ MMIO เข้าใจความหมายที่แท้จริงของคิวรี่และค้นหาเอกสารที่เกี่ยวข้องได้มีประสิทธิภาพมากกว่าอีก 2 เทคนิคที่นำมาเปรียบเทียบนั่นเอง ดังนั้นจึงสรุปได้ว่า สมมติฐานที่ 2 ของการวิจัยเป็นจริง



รูปที่ 24 เปรียบเทียบประสิทธิภาพการค้นคืนสารสนเทศเชิงความหมายสำหรับ MMIO, LSI, และ Lucene

4.5 การทดลองที่ 3: การทดลองเพื่อประเมินความไม่สมบูรณ์ของออนโทโลจี

ในการทดลองที่ 3 นี่มีจุดประสงค์เพื่อตรวจสอบว่า การที่นำเอาออนโทโลจีมาผสมกับ LSI จะช่วยเพิ่มประสิทธิภาพการทำงานของระบบได้มากขึ้น เมื่อระบบไม่สามารถค้นหาข้อมูลที่อยู่ในออน-โทโลจีหรือเมื่อระบบไม่เข้าใจความหมายของคิวรี่นั่นเอง เนื่องจากความจริงที่ว่าระบบที่สร้างขึ้นนี้คงไม่สามารถเข้าใจความหมายของคิวรี่จากผู้ใช้ได้ทุกคิวรี่ เช่น ในกรณีที่คิวรี่มีความยาวมากๆ และซับซ้อน ดังนั้นระบบการค้นหาข้อมูลที่ดีควรจะจัดการกับความไม่แน่นอน (Uncertainty) นี่ที่อาจจะเกิดขึ้นได้ เพื่อทำการทดลองนี้ข้อมูลบางข้อมูลในออนโทโลจีจะถูกลบทิ้งเพื่อให้เกิดความไม่สมบูรณ์ของข้อมูลและไม่สามารถหาข้อมูลนั้นพบ เพื่อตรวจสอบปฏิกิริยาของระบบว่าจะทำอย่างไร ในการ

ทดลองนี้ข้อมูลเกี่ยวกับกีฬาชกมวยได้ถูกลบทิ้งจากออนโทโลยี และผู้วิจัยได้ทำการทดลองใส่คิวรี่เพื่อหารูปภาพเกี่ยวกับกีฬาชกมวยเข้าไปในระบบ เช่น “*Boxing in London 2012*” ค่าพรีซีชั่นที่ได้จากการค้นหาของคิวรี่นี่แสดงดังรูปที่ 25 เมื่อออนโทโลจีไม่มีข้อมูลที่ผู้ใช้ต้องการ ระบบจะเรียกใช้ LSI ซึ่งได้ทำดัชนีเอาไว้ในอีกรูปแบบหนึ่งและสามารถใช้ดัชนีนั้นเพื่อค้นคืนเอกสารที่ต้องการได้ จากรูปที่ 25 จะเห็นว่ารูปบางรูปไม่ปรากฏคำว่า London อยู่เลยก็สามารถถูกพิจารณามาเป็นคำตอบได้ ทั้งนี่เนื่องจาก LSI พบว่าคำดังกล่าวมีการปรากฏร่วมกับคำว่า London ในความถี่ที่สูงนั่นเอง ดังนั้นจึง

10 August 2012, Australian light heavyweight Damien Hooper celebrates at the end of a round in his fight with Marcus Browne of the USA.

Team USA's Marcus Browne takes a standing count as Australia's world number two Damien Hooper pulls out a fantastic last round to beat him 13-11 in their light heavyweight bout in the London 2012 Olympics Games

Nicola Adams (in blue) jabs Chungneijang Mery Kom Hmangte of India during their 51kg fly women’s boxing match in the Olympic Park

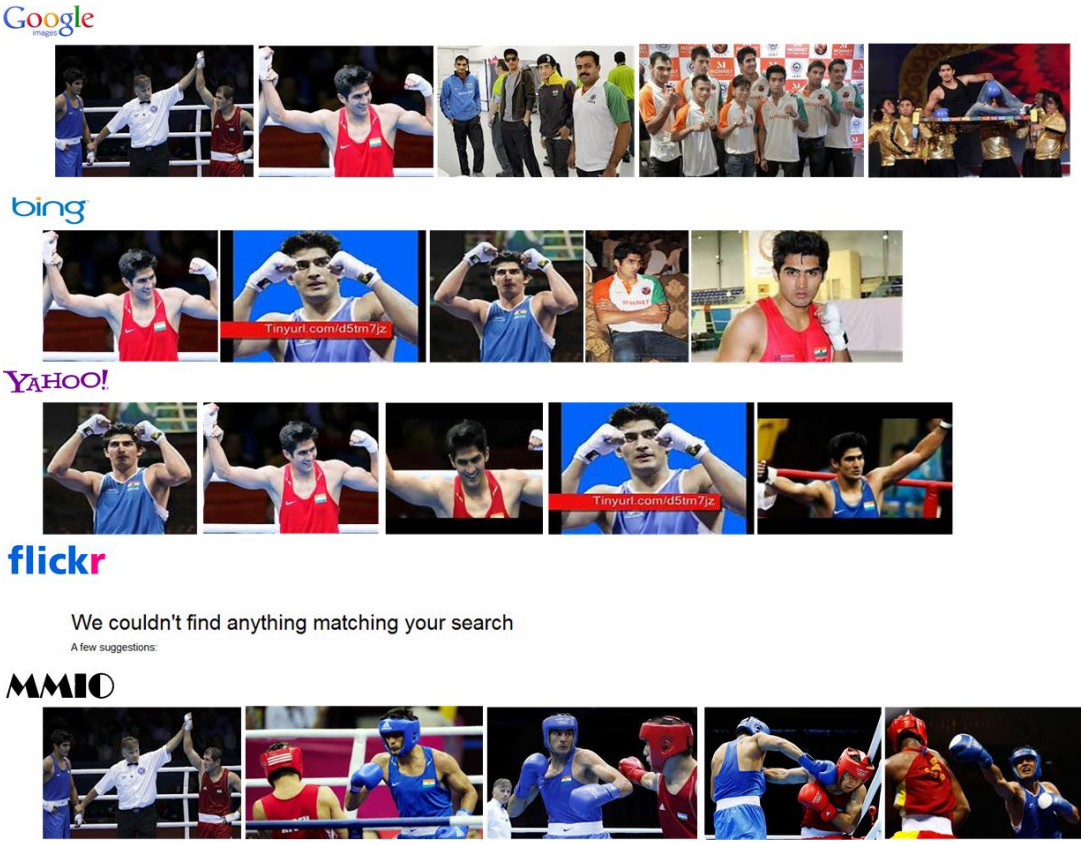
สรุปได้ว่าระบบที่นำเสนอในงานวิจัยนี้สามารถจัดการกับความไม่สมบูรณ์ของออนโทโลยีได้อย่างมีประสิทธิภาพ ดังนั้นสมมติฐานที่ 3 จึงถูกต้อง

รูปที่ 25 ผลการค้นหารูปกีฬาชกมวยโดยใช้ LSI

4.6 การทดลองที่ 4: การทดลองเพื่อเปรียบเทียบประสิทธิภาพการค้นหากับระบบค้นหาข้อมูลที่มีอยู่ในปัจจุบัน

การทดลองนี้เป็นการทดลองเพิ่มเติมเพื่อเปรียบเทียบประสิทธิภาพการค้นหาข้อมูลกับระบบค้นหาข้อมูลที่มีใช้อยู่ในปัจจุบันบนเครือข่ายอินเทอร์เน็ต ได้แก่ Google Bing Yahoo และ Flickr ในการทดลองนี้ได้มีการส่งคิวรี่เข้าไปในระบบทั้งหมด 20 คิวรี่ แต่ในรายงานวิจัยเล่มนี่จะเสนอผลการค้นหาข้อมูลเป็นตัวอย่างเพียงคิวรี่เดียวเท่านั้น และจากผลการค้นหาของระบบค้นหาข้อมูลแต่ละอัน ผู้วิจัยจะสนใจเฉพาะรูปภาพ 5 อันดับแรกเท่านั้นเพื่อใช้ในการเปรียบเทียบกัน การทดลองแรกคิวรี่ที่ใช้คนหารูปภาพคือ *“Vijender Singh performs in a boxing competition in London 2012”* รูปที่ 26 แสดงผลลัพธ์การค้นหารูปภาพของระบบต่างๆ ที่นำมาทดสอบ ซึ่งจะเห็นว่าทุกระบบสามารถค้นหารูปภาพของ Viender ซึ่งเป็นนักมวยชาวอินเดียได้อย่างถูกต้องทุกระบบ ยกเว้น Flickr ซึ่งระบบดังกล่าวไม่เข้าใจคิวรี่ที่เป็นภาษาธรรมชาติ (NLP) เมื่อพิจารณารูปภาพที่ได้จากการค้นหาของระบบค้นหาข้อมูลอย่างละเอียดพบว่า การจัดลำดับของรูปภาพในแต่ละระบบไม่เหมือนกัน

นอกจากนี้มีรูปภาพบางรูปยังไม่สอดคล้องกับความหมายของคิวรี่อีกด้วย ความหมายที่แท้จริงของคิวรี่คือ “หาภาพการต่อยมวยของนักมวยชื่อ Vijender Singh ในกีฬาโอลิมปิคที่ลอนดอน ปี 2012” จากผลลัพธ์ที่ได้จะเห็นว่า Google คนหารูปภาพที่ถูกต้องใน 2 อันดับแรกเท่านั้น ส่วนภาพที่เหลือไม่ใช่รูปภาพการชกมวยของนักมวยคนดังกล่าว ส่วนระบบค้นหาข้อมูลอื่นๆ ก็มีลักษณะเดียวกันคือได้รูปภาพที่ไม่ใช่การชกมวยมาปะปนกับรูปที่ผู้ใช้ต้องการ ในขณะที่ MMIO สามารถหารูปภาพการชกมวยได้อย่างถูกต้องทั้ง 5 รูปภาพแรก เนื่องเมต้าเดต้าในออนโทโลจีบอกได้ว่ารูปภาพนั้นมีเหตุการณ์อะไรเกิดขึ้น ซึ่งเมต้าเดต้าเหล่านี้ได้มาจากคำอธิบายรูปภาพซึ่งเขียนโดยมนุษย์ ดังนั้นหากคำอธิบายรูปผิด ระบบจะเข้าใจความหมายของรูปผิดไปด้วยเช่นกัน ซึ่งยังถือเป็นข้อจำกัดของระบบในงานวิจัยนี้อยู่



รูปที่ 26 ผลลัพธ์การค้นหารูปภาพนักมวย Vijender Singh ที่กำลังชกมวยในกีฬาโอลิมปิค London 2012

ในการทดลองถัดไป ผู้ทดลองได้ทดลองใช้คิวรี่ที่เป็นรูปภาพหรือเรียกอีกอย่างหนึ่งว่า “Query-by-Example (QBE)” เพื่อทดสอบการค้นหาข้อมูลระหว่าง MMIO และ Google ผลการทดลองแสดงดังรูปที่ 27 โดยที่รูปภาพต้นแบบ (Q) คือรูปภาพของนักมวย Vijender Singh กำลังต่อยมวย ผลลัพธ์การค้นหาของ Google ได้ผลลัพธ์ 10 อันดับแรกที่มีคุณลักษณะของรูป เช่น สีและรูปทรง ตรงกับ Q แต่จะสังเกตว่ามีรูปภาพบางรูปที่มีสีที่ตรงแต่เนื้อหาของภาพไม่ตรงกับ Q เช่น รูปที่ 7 และรูปที่ 9 ซึ่งเป็นกีฬาคนละประเภทกับ Q ทั้งนี้เนื่องจาก Google ใช้วิธีการเปรียบเทียบความเหมือนของรูปภาพจากคุณลักษณะของรูป ไม่ได้พิจารณาความหมายในรูปภาพเหล่านั้น ตรงกันข้ามกับ MMIO ใช้ความสัมพันธ์ของข้อมูลในออนโทโลจีและยังใช้คุณลักษณะของรูปภาพประกอบกัน ในตัวอย่างนี้จะเห็นว่า 5 รูปภาพแรกของ MMIO ล้วนแต่เป็นรูปภาพของ Vijender Singh ซึ่งกำลังชก

มวยเหมือน Q ทั้งสิ้น บางครั้งรูปภาพอาจจะไม่ชัดเจนในการตีความ แต่ MMIO พิจารณาข้อความอธิบายรูปภาพที่เก็บไว้ในออนโทโลยีประกอบเพื่อการตีความหมายของรูปภาพที่ถูกต้องมากขึ้นนั่นเอง

รูปที่ 27 ผลลัพธ์การค้นหารูปภาพโดยใช้คิวรี่เป็นรูปภาพตัวอย่าง (Q) ของ Google และ MMIO

บทที่ 5
สรุปผลการวิจัย

5.1 สรุปแนวคิดในงานวิจัย

งานวิจัยนี้ได้นำเสนอแนวคิดในการพัฒนาระบบค้นคืนรูปภาพกีฬา โดยหัวใจสำคัญของงานวิจัยนี้สามารถสรุปได้ดังนี้

1) แนวคิดในการแปลงโครงสร้างของคุณลักษณะระดับต่ำของรูปภาพจากรูปแบบเว็ก-เตอร์ (Vector space model) ให้เป็นโครงสร้างแบบลำดับชั้น (Hierarchical model) ซึ่งทำให้เกิดประโยชน์ดังต่อไปนี้
   - ใช้เป็นตัวแทนข้อมูลของรูปภาพได้มีประสิทธิภาพมากขึ้น
   - เพิ่มประสิทธิภาพในการจำแนกรูปภาพ
   - โครงสร้างแบบลำดับชั้นช่วยลดความกำกวมของวิชวลเวิร์ดได้อย่างมีประสิทธิภาพ
2) แนวคิดในการรวมข้อมูลต่างชนิดกันไว้ในออนโทโลจีเดียวกัน เพื่อเพิ่มประสิทธิภาพในการค้นคืนรูปภาพ ซึ่งในการวิจัยนี้ทำการรวมข้อมูลประเภทตัวอักษรและคุณลักษณะระดับต่ำของรูปภาพไว้ด้วยกัน
3) แนวคิดที่ลดข้อจำกัดความไม่สมบูรณ์ของออนโทโลจี โดยใช้ LSI มาช่วยในการค้นหาข้อมูล เพื่อทำให้ระบบไม่ขึ้นอยู่กับโครงสร้างของออนโทโลจีเพียงอย่างเดียว ในกรณีที่นักออกแบบไม่ชำนาญหรือออกแบบออนโทโลจีไม่เหมาะสม ทำให้ไม่สามารถหาข้อมูลที่ต้องการได้ แนวคิดที่นำเสนอจะลดจำกัดจุดนี้ลงไป

จากผลการทดลองโดยเปรียบเทียบกับเทคนิคเดิมๆ ที่มีอยู่และเปรียบเทียบกับระบบค้นหาข้อมูลรูปภาพที่มีอยู่ในปัจจุบันในท้องตลอดพบว่า สามารถเพิ่มความถูกต้องของรูปภาพที่ทำการค้นหาได้จริง สามารถทำการค้นหารูปภาพเชิงความหมายได้ถึงแม้ว่าคำอธิบายรูปภาพ (Annotation) จะไม่ใช่คำเดียวกับคิวรี่ก็ตาม สรุปผลการวิจัยเป็นไปตามวัตถุประสงค์ที่ตั้งไว้ทุกประการ

5.2 การเผยแพร่งานวิจัย

แนวคิดในงานวิจัยนี้ได้รับการยอมรับจากการประชุมวิชาการนานาชาติและวารสารระดับนานาชาติที่เป็นที่ยอมรับในวงการวิชาการ โดยได้รับการตอบรับการตีพิมพ์จากวารสารระดับประเทศและระดับนานาชาติ ดังรายการต่อไปนี้

1) **Kraisak Kesorn**, (2013), “Athletics images interpretation using structural ontology model”, ***วิศวกรรมสาร มข.***, ปีที่ 40, ฉบับที่ 1, หน้า. 11472-11481 (TCI กลุ่มที่ 1, IF 2013 = 0.098)
2) Stefan Poslad and **Kraisak Kesorn**, (2014), “A Multi-Modal Incompleteness Ontology Model (MMIO) to enhance information fusion for image retrieval”, Information Fusion, Vol. 40, pp. 225-241. (IF 2013 = 3.472)

5.3 ปัญหาและอุปสรรค

ในการทำวิจัยนี้พบปัญหาในการทำวิจัยน้อยมาก ส่วนใหญ่เป็นเรื่องของอุปกรณ์ด้านฮาร์ดแวร์ เช่น เครื่องคอมพิวเตอร์ส่วนบุคคลที่ใช้ในการทดลองมีความเร็วไม่พอ เนื่องจากเป็นงานวิจัยเกี่ยวกับการประมวลผลรูปภาพ ดังนั้นจึงมีความต้องการใช้ฮาร์ดแวร์ที่มีประสิทธิภาพการทำงานที่สูงกว่าปกติ แต่เนื่องจากทุนวิจัยไม่สนับสนุนให้ซื้อฮาร์ดแวร์ ผู้วิจัยจึงจำเป็นต้องเครื่องคอมพิวเตอร์ส่วนบุคคลที่มีอยู่ จึงทำให้การทดลองใช้เวลานานกว่าปกติ ส่วนปัญหาด้านอื่นๆ ได้แก่ เวลาในการทำวิจัยซึ่งผู้วิจัยมีภาระต้องสอนนิสิตและงานบริการวิชาการอื่นๆ ทำให้ความก้าวหน้าของงานวิจัยล่าช้ากว่ากำหนด

5.4 แนวทางการพัฒนาในอนาคต

การพัฒนาต่อยอดงานวิจัยจะมุ่งเน้นไปที่ 2 ส่วน ได้แก่ การลดเวลาในการประมวลผลคิวรี่และการค้นคืนรูปภาพเฉพาะบุคคล (Personalized Image Retrieval) ทั้งนี้เนื่องจากข้อจำกัดสำคัญของระบบที่นำเสนอคือใช้เวลานานในการประมวลผลคิวรี่เพื่อให้เข้าใจความหมายของคิวรี่อย่างแท้จริง และการค้นหารูปภาพในคอลเล็คชั่นยังใช้เวลานาน ทั้งนี่เนื่องจากระบบต้องประมวลผลข้อมูล 2 ประเภท ทั้งข้อความและรูปภาพ จึงต้องใช้เวลานานกว่าระบบอื่นๆ ทั่วไป นอกจากนี้การค้นคืนรูปภาพที่ตรงกับความสนใจเฉพาะบุคคลยังเป็นแนวคิดที่น่าสนใจในการพัฒนาต่อยอดงานวิจัยต่อไปในอนาคต ซึ่งมีความท้าทายหลายประการ เช่น จะค้นหาความสนใจของผู้ใช้อย่างไร จะระบุตัวตนของผู้ใช้ว่าเป็นใครทำอย่างไร ตลอดจนการเก็บความสนใจของผู้ใช้เพื่อนำไปวิเคราะห์หาความสนใจ

ของผู้ใช้เมื่อมีการกลับมาใช้ระบบใหม่อีกครั้งยังเป็นหัวข้อที่น่าสนใจในงานวิจัยเกี่ยวกับการพัฒนาระบบการค้นคืนรูปภาพอยู่มาก

บทที่ 6

บรรณานุกรม

AlSumait, L., Domeniconi, C., 2008. Text Clustering with Local Semantic Kernels, in: Berry, M.W., Castellanos, M. (Eds.), Survey of Text Mining II: Clustering, Classification, and Retrieval. Springer-Verlag London Limited, London, United Kingdom, pp. 87–105.

Blei, D.M., Ng, A.Y., Jordan, M.I., 2003. Latent Dirichlet Allocation. J. Mach. Learn. Res. 3, 993–1022.

Chandrasekaran, B., Josephson, J.R., Benjamins, V.R., 1999. What Are Ontologies, and Why Do We Need Them? IEEE Intell. Syst. 14, 20–26.

Chisholm, E., Kolda, T.G., 1999. New Term Weighting Formulas For The Vector Space Method In Information Retrieval. Computer Science and Mathematics Division, Oak Ridge National Laboratory, USA.

Csurka, G., Dance, C.R., Fan, L., Willamowski, J., Bray, C., 2004. Visual Categorization with Bags of Keypoints, in: International Workshop on Statistical Learning in Computer Vision. pp. 1–22.

Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R., 1990. Indexing by Latent Semantic Analysis. J. Am. Soc. Inf. Sci. 41, 391–407.

Gasevic, D., Djuric, D., Devedzic, V., 2009. Model Driven Engineering and Ontology Development, 2nd ed. Springer, London, United Kingdom.

Gruber, T.R., 1993. A Translation Approach to Portable Ontology Specifications. Knowl. Acquis. 5, 199–220.

Guarino, E.N., Poli, R., Guarino, N., 1995. Formal Ontology, Conceptual Analysis and Knowledge Representation. Int. J. Hum.-Comput. Stud. 43, 625–640.

Harris, C., Stephens, M., 1988. A Combined Corner and Edge Detector, in: Proceedings of the 4th Alvey Vision Conference. pp. 147–151.

Jiang, Y.-G., Ngo, C.-W., 2009. Visual Word Proximity and Linguistics for Semantic Video Indexing and Near-Duplicate Retrieval. Comput. Vis. Image Underst. 113, 405–414.

Jing, F., Li, M., Zhang, L., Zhang, H.-J., Zhang, B., 2003. Learning in Region-Based Image Retrieval. pp. 206–215.

Kesorn, K., Poslad, S., 2012. An Enhanced Bag of Visual Word Vector Space Model to Represent Visual Content in Athletics Images. IEEE Trans. Multimed. 14, 1520–1532.

Lowe, D.G., 1999. Object Recognition from Local Scale-Invariant Features, in: Proceedings of the International Conference on Computer Vision. pp. 1150–1157.

Lowe, D.G., 2004. Distinctive Image Features from Scale-Invariant Keypoints. Int. J. Comput. Vis. 60, 91–110.

Manjunath, B.S., Ohm, J.-R., Vasudevan, V.V., Yamada, A., 2001. Color and Texture Descriptors. IEEE Trans. Circuits Syst. Video Technol. 11, 703–715.

McGuinness, D., 2003. Ontologies Come of Age, in: Fensel, D., Hendler, J., Lieberman, H., Wahlster, W. (Eds.), The Semantic Web: Why, What, and How. MIT Press, Boston, MA.

Mikolajczyk, K., Schmid, C., 2003. A Performance Evaluation of Local Descriptors. Presented at the Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 257–263.

Neches, R., Fikes, R., Finin, T., Gruber, T., Patil, R., Senator, T., Swartout, W.R., 1991. Enabling Technology for Knowledge Sharing. AI Mag. 12, 36–56.

Poslad, S., 2009. Ubiquitous Computing Smart Devices, Environments and Interactions. John Willey & Sons, London, United Kingdom.

Schreiber, G., Wielinga, B., de Hoog, R., Akkermans, H., Van de Velde, W., 1994. CommonKADS: A Comprehensive Methodology for KBS Development. IEEE Expert 9, 28–37.

Sivic, J., Russell, B.C., Zisserman, A., Freeman, W.T., Efros, A.A., 2008. Unsupervised Discovery of Visual Object Class Hierarchies, in: IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–8.

Tamura, H., Mori, S., Yamawaki, T., 1978. Textural Features Corresponding to Visual Perception. IEEE Trans. Syst. Man Cybern. 8, 460–473.

Tirilly, P., Claveau, V., Gros, P., 2008. Language Modeling for Bag-of-Visual Words Image Categorization, in: Proceedings of the 2008 International Conference on Content-Based Image and Video Retrieval. pp. 249–258.

Tong, S., Chang, E., 2001. Support Vector Machine Active Learning for Image Retrieval, in: Proceedings of the 9th ACM International Conference on Multimedia, MULTIMEDIA '01. ACM, pp. 107–118.

Walter, B., Bala, K., Kulkarni, M., Pingali, K., 2008. Fast Agglomerative Clustering for Rendering, in: IEEE Symposium on Interactive Ray Tracing. Presented at the IEEE Symposium on Interactive Ray Tracing, pp. 81–86.

Wang, J.-Z., Li, J., Chan, D., Wiederhold, G., 1999. Semantics-Sensitive Retrieval for Digital Picture Libraries. Digit. Libr. Mag. 5.

Wang, J.-Z., Li, J., Wiederhold, G., 2001. SIMPLIcity: Semantics-Sensitive Integrated Matching for Picture Libraries. IEEE Trans. Pattern Anal. Mach. Intell. 23, 947–963.

Wang, L., Lu, Z., Ip, H.H., 2009. Image Categorization Based on a Hierarchical Spatial Markov Model, in: Proceedings of the 13th International Conference on Computer Analysis of Images and Patterns. pp. 766–773.

Wu, L., Hoi, S.C.H., Yu, N., 2009. Semantics-Preserving Bag-of-Words Models for Efficient Image Annotation, in: Proceedings of the 1st ACM Workshop on Large-Scale Multimedia Retrieval and Mining. pp. 19–26.

Zheng, X., Cai, D., He, X., Ma, W.-Y., Lin, X., 2004. Locality preserving clustering for image database, in: Proceedings of the 12th Annual ACM International Conference on Multimedia. ACM, New York, NY, USA, pp. 885–891.

Zheng, Y.-T., Neo, S.-Y., Chua, T.-S., Tian, Q., 2008. Toward a Higher-Level Visual Representation for Object-based Image Retrieval. Vis. Comput. 25, 13–23.

Zhu, J., Uren, V., Motta, E., 2005. ESpotter: Adaptive Named Entity Recognition for Web Browsing, in: Intelligent IT Tools for Knowledge Management Systems, KMTOOLS 2005. pp. 518–529.

ภาคผนวก ก
ประวัติผู้วิจัย

1. **ชื่อ - นามสกุล (ภาษาไทย)** นายไกรศักดิ์ เกษร
2. **ชื่อ - นามสกุล (ภาษาอังกฤษ)** Mr.Kraisak Kesorn
   เลขหมายบัตรประจำตัวประชาชน 3-6501-00083-57-8
3. **ตำแหน่งปัจจุบัน** ผู้ช่วยศาสตราจารย์
4. **หน่วยงาน** ภาควิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ
   คณะวิทยาศาสตร์ มหาวิทยาลัยนเรศวร อ.เมือง จ.พิษณุโลก 65000

   หมายเลขโทรศัพท์ 055-963-263, 081-555-7499

   โทรสาร 055-963113

   อีเมล์ kraisakk@nu.ac.th

5. **ประวัติการศึกษา**

ระยะเวลา	วุฒิการศึกษา	สาขาวิชา	มหาวิทยาลัย
2549-2553	ป.เอก	วิศวกรรมอิเล็กทรอนิคและวิทยาการคอมพิวเตอร์	Queen Mary University of London, United Kingdom
2544-2546	ป.โท	เทคโนโลยีสารสนเทศ	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
2537-2540	ป.ตรี	วิทยการคอมพิวเตอร์	มหาวิทยาลัยเชียงใหม่

6. ผลงานวิจัยที่ได้รับการตีพิมพ์ระดับนานาชาติ

1) Stefan Poslad and **Kraisak Kesorn**, “A Multi-Modal Incompleteness Ontology Model (MMIO) to enhance information fusion for image retrieval”, Information Fusion, Vol. 40, pp. 225-241, 2014.
2) Phatsavee Ongruk, Padet Siriyasatien, and **Kraisak Kesorn**, “New Key Factors Discovery to Enhance Dengue Fever Forecasting Model”, Advanced Materials Research, Vol. 931-932, pp. 1461-1475, 2014.
3) Duangduen Asavasuthirakul, Antony Harfield, and **Kraisak Kesorn**, (๒๐๑๔), “A Framework of Personalized Travelling Information Services for Thailand”, Advanced Materials Research, Vol. 931-932, pp. 1382-1386, 2014.
4) Intaraprapan, C., **Kesorn, K.**, “Searching and Ranking Techniques for Cross-Language (Thai-English) Information Retrieval”, *In The 10th International Joint Conference on Computer Science and Software Engineering (JCSSE'13)*, Khon Khan, Thailand, 29-31 May, 2013.
5) **Kesorn K.** (2013), Athletics Images Interpretation using Structural Ontology Model, KKU Engineering Journal , (In press). (Impact Factor=0.104)
6) **Kesorn K.**,S., Poslad (2012),, An Enhanced Bag of Visual Word Vector Space Model to Represent Visual Content in Athletics Images, *IEEE Transactions on Multimedia*, 14(1), 1520-1531. (Impact Factor 2010=1.964)
7) **Kesorn K.**,Chimlek, S., Poslad, S., & Piamsa-nga, P. (2011),, Visual Content Representation using Semantically Similar Visual Words. *Expert Systems with Applications*, 38(9), 11472-11481. (Impact Factor 2011=2.234)
8) Liang, Z.,**Kesorn K.**, Poslad, S. The USHER System to Generate Personalised Spatial Maps for Travellers, In: Semantics in Adaptive and Personalised Services: Methods, Tools and Applications., Wallace, M., Anagnostopoulos, I., Mylonas, P., and Bielikova, M. (Eds)

9) **Kesorn K.**, Poslad, S., Semantic Restructuring of Natural Language Image Captions to Enhance Image Retrieval, *Journal of Multimedia (JMM)*, Vol.4, Issue 5, October 2009, pp.284-297.

10) Chimlek S.,**Kesorn K.**, Piamsa-nga P., Poslad S., "Semantically Similar Visual Word Identification for Efficient Visual Content Representation", *In IEEE International Conference on Multimedia & Expo (ICME 2010)*, Singapore, 19-23, July, 2010.

11) **Kesorn K.**, Poslad, S. "Enhanced Sports Image Annotation and Retrieval Based upon Semantic Analysis of Multimodal Cues", In IEEE/ACM SIG Multimedia Proceedings of the 3rd Pacific-Rim Symposium on Image and Video Technology (PSIVT2009), Tokyo, Japan, 14-16 January, 2009.

ภาคผนวก ข

ผลงานการตีพิมพ์