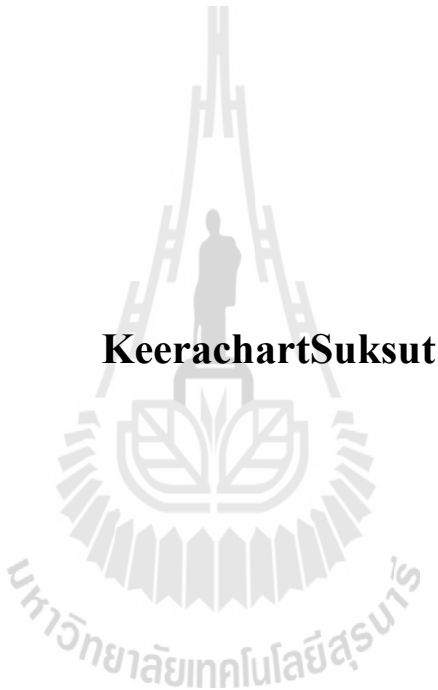
การตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัย

นายกีระชาติ สุขสุทธิ์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์
มหาวิทยาลัยเทคโนโลยีสุรนารี
ปีการศึกษา 2556

QUERY ANSWERING WITH FUZZY TOP-K RANKING TECHNIQUE IN DEDUCTIVE DATABASE

KeerachartSuksut



A Thesis Submitted in Partial Fulfillment of the Requirements for the

Degree of Master of Engineering in Computer Engineering

SuranareeUniversity of Technology

Academic Year 2013

การตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัย

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

คณะกรรมการสอบวิทยานิพนธ์

_____

(รศ.ดร.กิตติศักดิ์ เกิดประสพ)
ประธานกรรมการ

_____

(รศ.ดร.นิตยา เกิดประสพ)
กรรมการ (อาจารย์ที่ปรึกษาวิทยานิพนธ์)

_____

(ผศ. ดร.ชุติมา พรหมมาก)
กรรมการ

_____

(ผศ. ร.อ. ดร.ประโยชน์ คำสวัสดิ์)
กรรมการ

_____

(ศ. ดร.ชูกิจ ลิมปิจำนงค์)
รองอธิการบดีฝ่ายวิชาการและนวัตกรรม

_____

(รศ.ร.อ.ดร.กนต์ธร ชำนิประศาสน์)
คณบดีสำนักวิชาวิศวกรรมศาสตร์

กีระชาติ สุขสุทธิ์ : การตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัย (QUERY ANSWERING WITH FUZZY TOP-K RANKING TECHNIQUE IN DEDUCTIVE DATABASE) อาจารย์ที่ปรึกษา: รองศาสตราจารย์ ดร. นิตยา เกิดประสพ,121 หน้า.

งานวิจัยนี้ได้ศึกษาปัญหาการจัดอันดับคำตอบ เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัย ซึ่งการจัดอันดับ เค อันดับแรกเป็นการจัดอันดับเพื่อตอบคำถามที่มีความคลุมเครือในการสอบถามโดยเลือกเอาคำตอบที่ดีที่สุด เค อันดับ ในอดีตได้มีหลากหลายงานวิจัยที่ได้เสนอเทคนิคในการจัดอันดับและเลือกเอา เค อันดับแรกสำหรับคำถามที่มีความคลุมเครือ แต่เทคนิคส่วนใหญ่ทำการจัดอันดับและหา เค อันดับแรกในฐานข้อมูลนิรนัยด้วยการคำนวณหาค่าคะแนนจากข้อมูลทั้งหมดและเรียงลำดับด้วยคะแนน แต่การจัดอันดับและหา เค อันดับแรกในฐานข้อมูลนิรนัยโดยใช้วิธีการคำนวณค่าความเป็นสมาชิกด้วยฟังก์ชันสมาชิกและการนำฟังก์ชันสมาชิกเข้ามาช่วยจะมีความซับซ้อน จึงทำให้มีงานวิจัยด้านนี้ปรากฏค่อนข้างน้อย ผู้วิจัยได้เห็นความสำคัญในจุดนี้จึงได้เสนอเทคนิคการจัดเรียง เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัยโดยใช้ฟังก์ชันสมาชิกในการจัดอันดับ เพื่อเพิ่มประสิทธิภาพในการจัดอันดับ เค อันดับแรกในฐานข้อมูลนิรนัยให้สามารถจัดอันดับเฉพาะช่วงข้อมูลที่สนใจได้ โดยจะทำการเปรียบเทียบประสิทธิภาพของฟังก์ชันสมาชิกแต่ละฟังก์ชันที่ใช้ในการจัดอันดับ โดยใช้ภาษาดาต้าล็อกในส่วนที่เกี่ยวข้องกับการจัดอันดับ เค อันดับแรก และใช้ฟังก์ชันสมาชิก 4 ฟังก์ชันในการคำนวณค่าความเป็นสมาชิก

มหาวิทยาลัยเทคโนโลยีสุรนารี

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ลายมือชื่อนักศึกษา_____

ปีการศึกษา 2556 ลายมือชื่ออาจารย์ที่ปรึกษา_____

KEERACHART SUKSUT : QUERY ANSWERING WITH FUZZY TOP-K RANKING TECHNIQUE IN DEDUCTIVE DATABASE. THESIS ADVISOR : ASSOC. PROF. NITTAYA KERDPRASOP, Ph.D., 121 PP.

DEDUCTIVE DATABASE/TOP-K RANKING/MEMBERSHIP FUNCTION

In this research, we study the problem of ranking top-k answer for deductive database. Top-k ranking is the selection for the best k information to answer the question that has ambiguity in the query. In the past, there have been several researches proposed to find top-k answers with scoring function technique to reduce ambiguity in the question. But the research that uses membership function technique is quite a few. This research thus proposes top-k ranking technique with membership function for deductive database to improve performance in the ranking of top-k answers in deductive database by ranking only data range of interest. We show comparative results for each kind of membership function using datalog language for top-k ranking and using membership function to evaluate membership value.

มหาวิทยาลัยเทคโนโลยีสุรนารี

School of Computer Engineering Student's Signature_____

Academic Year 2013 Advisor's Signature_____

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงด้วยดี ผู้วิจัยขอกราบขอบพระคุณ บุคคล และกลุ่มบุคคลต่างๆที่ได้กรุณาให้คำปรึกษา แนะนำ ช่วยเหลืออย่างดียิ่ง ทั้งในด้านวิชาการ และด้านการดำเนินงานวิจัย ดังต่อไปนี้

รองศาสตราจารย์ ดร.นิตยา เกิดประสพ อาจารย์ที่ปรึกษาวิทยานิพนธ์ และรองศาสตราจารย์ ดร.กิตติศักดิ์ เกิดประสพ ที่ให้คำปรึกษาในการทำงานวิจัย การจัดการรูปแบบ และช่วยตรวจทานความถูกต้องของวิทยานิพนธ์

คุณกัลญา พับโพธิ์ เลขานุการสาขาวิชาวิศวกรรมคอมพิวเตอร์ ที่ให้ความช่วยเหลือในการประสานงานด้านเอกสารระหว่างศึกษา

คุณภาสพิชญ์ ชูใจ คุณไพชยนต์ คงไชยคุณนันทวุฒิ คะอังกุ คุณกิตติพงศ์ ชมบุญและนักศึกษาบัณฑิตสาขาวิชาวิศวกรรมคอมพิวเตอร์ ทุกท่านที่ให้คำปรึกษา ช่วยตรวจทานความถูกต้องและช่วยเหลือด้วยดีมาโดยตลอด

นอกจากนี้ขอขอบคุณครู อาจารย์ทั้งในอดีตและปัจจุบันที่ให้ความรู้แก่ผู้วิจัยจนประสบความสำเร็จในชีวิต

ท้ายที่สุดที่จะลืมไม่ได้ ขอกราบขอบพระคุณ บิดา มารดา ที่ให้กำเนิด อบรม เลี้ยงดูด้วยความรัก และส่งเสริมการศึกษาเป็นอย่างดีโดยตลอด ทำให้ผู้วิจัยมีความรู้ ความสามารถ มีจิตใจที่เข้มแข็ง รวมทั้งเป็นกำลังใจที่ยิ่งใหญ่แก่ผู้วิจัย จนทำให้ผู้วิจัยประสบความสำเร็จในชีวิตเรื่อยมา

กีระชาติสุขสุทธิ์

สารบัญ

หน้า

บทคัดย่อ(ภาษาไทย) ..... ก
บทคัดย่อ (ภาษาอังกฤษ) ..... ข
กิตติกรรมประกาศ ..... ค
สารบัญ ..... ง
สารบัญตาราง ..... ซ
สารบัญรูป ..... ญ

บทที่

1 บทนำ 1
1.1 ความสำคัญและที่มาของปัญหาการวิจัย 1
1.2 วัตถุประสงค์ของการวิจัย 2
1.3 ข้อตกลงเบื้องต้น 2
1.4 ขอบเขตของการวิจัย 3
1.5 ประโยชน์ที่ได้รับ 3
2 ปริทัศน์วรรณกรรมและงานวิจัยที่เกี่ยวข้อง 4
2.1 การทำเหมืองข้อมูล(Data Mining) 4
2.1.1 วิวัฒนาการของการประมวลผลและจัดการกับข้อมูล 4
2.1.2 วัตถุประสงค์ในการทำเหมืองข้อมูล 4
2.1.3 สถาปัตยกรรมของการทำเหมืองข้อมูล 5
2.1.4 ขั้นตอนการทำเหมืองข้อมูล 6
2.1.5 เทคนิคการจัดกลุ่ม (Clustering) 7
2.2 ฟัซซี่ลอจิก (Fuzzy Logic) 10
2.2.1 ข้อดีของฟัซซี่ลอจิก 11
2.2.2 ฟัซซี่เซต (Fuzzy Set) 11
2.2.3 ฟังก์ชันสมาชิก (Membership Function) 12

สารบัญ (ต่อ)

หน้า

2.3 ภาษาโปรล็อก (Prolog) และข้อมูลคลุมเครือ 15

2.4 ภาษาดาต้าล็อก (Datalog) 16

2.4.1 ชนิดข้อมูลในภาษาดาต้าล็อก 16

2.4.2 กลไกการทำงานและนิยามในดาต้าล็อก 18

2.5 ดาต้าล็อก$^{top-k}$ 21

2.6 โปรแกรมดีอีเอส (Datalog Education System) 22

2.7 งานวิจัยที่เกี่ยวข้อง 26

3 วิธีดำเนินการวิจัย 29

3.1 กรอบแนวคิดของการวิจัย 29

3.1.1 ขั้นตอนที่ 1 : วิธีการจัดกลุ่ม (Clustering) 29

3.1.2 ขั้นตอนที่ 2 : การจัดอันดับ top-k ของข้อมูล 31

3.2 การออกแบบอัลกอริทึม 32

3.2.1 ออกแบบอัลกอริทึมการจัดกลุ่มข้อมูลจำนวนประชากรโลกด้วย k-means 32

3.2.2 ออกแบบอัลกอริทึมการจัดอันดับ top-k 33

3.3 การใช้งานโปรแกรม 35

3.3.1 การเตรียมข้อมูล 35

3.3.2 การจัดกลุ่มข้อมูล 37

3.3.3 การจัดอันดับ top-k 40

3.4 เครื่องมือที่ใช้ในงานวิจัย 46

4 การทดสอบและอภิปรายผล 47

4.1 ข้อมูลที่ใช้ในการทดสอบ 47

4.2 การแบ่งกลุ่มด้วยอัลกอริทึม k-means 49

4.2.1 เตรียมข้อมูลสำหรับการแบ่งกลุ่มด้วยอัลกอริทึม k-means 49

4.2.2 ผลของการแบ่งกลุ่มด้วยอัลกอริทึมk-means 50

4.3 การตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรก 51

สารบัญ (ต่อ)

หน้า

4.3.1 สร้างฟังก์ชันสมาชิกแต่ละฟังก์ชัน 51

4.3.2 เตรียมข้อมูลสำหรับการตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรก แบบคลุมเครือในฐานข้อมูลนิรนัย 60

4.4 ผลการตอบคำถามการจัดเรียง เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัย 62

4.4.1 ผลการทดลองจากการตอบคำถามการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Trapezoidal Function 62

4.4.2 ผลการทดลองจากการตอบคำถามการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Triangular Function 65

4.4.3 ผลการทดลองจากการตอบคำถามการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Left Shoulder Function 67

4.4.4 ผลการทดลองจากการตอบคำถามการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Right Shoulder Function 70

4.5 เปรียบเทียบผลการทดลองด้วยฟังก์ชันสมาชิกแต่ละฟังก์ชัน 73

4.5.1 เปรียบเทียบการใช้งานระหว่างฟังก์ชัน Trapezoidal Function กับ Triangular Function 73

4.5.2 เปรียบเทียบการใช้งานระหว่างฟังก์ชัน Left Shoulder Function กับ Right Shoulder Function 75

4.6 อภิปรายผล 77

5 สรุปผลการวิจัยและข้อเสนอแนะ 79

5.1 ขั้นตอนการดำเนินงานวิจัย 79

5.2 สรุปผลการวิจัย 80

5.3 ปัญหาและข้อเสนอแนะ 81

รายการอ้างอิง 82

สารบัญ (ต่อ)

หน้า

ภาคผนวก

ภาคผนวก ก.รหัสต้นฉบับโปรแกรม 84

ภาคผนวก ข.บทความวิจัยที่ได้รับการตีพิมพ์เผยแพร่ 99

ประวัติผู้เขียน 121

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญตาราง

ตารางที่ **หน้า**

2.1 ข้อมูลหลังผ่านการ Cluster รอบที่ 1 9
2.2 ข้อมูลหลังผ่านการ Cluster รอบที่ 2 10
2.3 แสดงข้อมูลที่ผ่านกระบวนการแบ่งกลุ่มด้วย k-means 10
2.4 ชุดข้อมูลตัวอย่าง 15
2.5 สรุปเปรียบเทียบงานวิจัยที่เกี่ยวข้องกับการจัดอันดับ top-k 28
4.1 ตัวอย่างข้อมูลจำนวนประชากรโลก 48
4.2 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 90,000,000 คน ถึง 93,000,000 คน ด้วยฟังก์ชัน Trapezoidal Function 62
4.3 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 24,000,000 คน ถึง 27,000,000 คน ด้วยฟังก์ชัน Trapezoidal Function 63
4.4 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 2,050,000 คน ถึง 2,500,000 คน ด้วยฟังก์ชัน Trapezoidal Function 64
4.5 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 90,000,000 คน ด้วยฟังก์ชัน Triangular Function 65
4.6 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 25,000,000 คน ด้วยฟังก์ชัน Triangular Function 66
4.7 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 2,100,000 คน ด้วยฟังก์ชัน Triangular Function 67
4.8 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 90,000,000 คน ด้วยฟังก์ชัน Left Shoulder Function 68
4.9 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 25,000,000 คน ด้วยฟังก์ชัน Left Shoulder Function 69
4.10 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 2,100,000 คน ด้วยฟังก์ชัน Left Shoulder Function 70

สารบัญตาราง(ต่อ)

ตารางที่ **หน้า**

4.11 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 90,000,000 คน ด้วยฟังก์ชัน Right Shoulder Function 71
4.12 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 25,000,000 คน ด้วยฟังก์ชัน Right Shoulder Function 72
4.13 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 2,100,000 คน ด้วยฟังก์ชัน Right Shoulder Function 73

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญรูป

รูปที่ **หน้า**

2.1 สถาปัตยกรรมระบบการทำเหมืองข้อมูล 5
2.2 กระบวนการทำเหมืองข้อมูล 6
2.3 Partitional Clustering 7
2.4 Hierarchical Clustering 8
2.5 ตัวอย่างข้อมูลสำหรับการทำ k-means 9
2.6 ความแตกต่างระหว่างเซตชัดเจนและฟัซซี่เซต 11
2.7 ขอบเขตของฟัซซี่เซต 11
2.8 ความแตกต่างของเซตชัดเจนกับฟัซซี่เซต 12
2.9 ขอบเขตและวิธีการหาค่าความเป็นสมาชิกของ Trapezoidal Function 12
2.10 ขอบเขตและวิธีการหาค่าความเป็นสมาชิกของ Triangular Function 13
2.11 ขอบเขตและวิธีการหาค่าความเป็นสมาชิกของ Left Shoulder Function 14
2.12 ขอบเขตและวิธีการหาค่าความเป็นสมาชิกของ Right Shoulder Function 14
2.13 อินเตอร์เฟสของโปรแกรมดีอีเอส 22
2.14 ตัวอย่างการถามคำถามในโปรแกรมดีอีเอส 23
2.15 ตัวอย่างการตอบคำถามในรูปแบบ Disjunctive query ในโปรแกรมดีอีเอส 24
2.16 ตัวอย่างการตอบคำถามในรูปแบบการเรียกซ้ำในโปรแกรมดีอีเอส 25
2.17 ตัวอย่างการตอบคำถาม เค อันดับแรก ในโปรแกรมดีอีเอส 26
3.1 กรอบแนวคิดวิธีการจัดกลุ่ม (Clustering) 30
3.2 กรอบแนวคิดการจัดอันดับ top-k ของข้อมูลในกลุ่มย่อย 31
3.3 ผังงานแสดงขั้นตอนการจัดกลุ่มด้วยอัลกอริทึม k-means 32

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

3.4 ผังงานแสดงขั้นตอนการจัดอันดับ top-k 34
3.5 ส่วนต่าง ๆ ของข้อมูลสำหรับการจัดกลุ่ม 35
3.6 ส่วนต่าง ๆ ของข้อมูล 36
3.7 หน้าต่างโปรแกรมและส่วนประกอบต่าง ๆ ใน weka 3.6 37
3.8 ส่วนประกอบต่าง ๆ ของการจัดกลุ่มใน weka 3.6 38
3.9 โมเดลของการจัดกลุ่ม 39
3.10 รายละเอียดข้อมูลหลังผ่านการจัดกลุ่ม 40
3.11 รูปแบบการเรียกใช้โปรแกรมการจัดอันดับ top-k 40
3.12 เงื่อนไขในการจัดอันดับด้วยฟังก์ชันสมาชิก Triangular Function 41
3.13 ผลลัพธ์จากการจัดอันดับ top-k ด้วย Triangular Function 41
3.14 เงื่อนไขในการจัดอันดับด้วยฟังก์ชันสมาชิก Trapezoidal Function 42
3.15 ผลลัพธ์จากการจัดอันดับ top-k ด้วย Trapezoidal Function 43
3.16 เงื่อนไขในการจัดอันดับด้วยฟังก์ชันสมาชิก Left Shoulder Function 43
3.17 ผลลัพธ์จากการจัดอันดับ top-k ด้วย Left Shoulder Function 44
3.18 เงื่อนไขในการจัดอันดับด้วยฟังก์ชันสมาชิก Right Shoulder Function 45
3.19 ผลลัพธ์จากการจัดอันดับ top-k ด้วย Right Shoulder Function 45
4.1 ตัวอย่างข้อมูลจำนวนประชากรโลกในรูปแบบ .arff 49
4.2 ตัวอย่างข้อมูลที่ผ่านการแบ่งกลุ่มด้วยอัลกอริทึม k-means 50
4.3 Trapezoidal Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับสูง 51
4.4 กราฟ Trapezoidal Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับสูง 52
4.5 Trapezoidal Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง 52
4.6 กราฟ Trapezoidal Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง 52
4.7 Trapezoidal Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ 53
4.8 กราฟ Trapezoidal Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ 53

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.9	Triangular Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับสูง	53
4.10	กราฟ Triangular Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับสูง	54
4.11	Triangular Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง	54
4.12	กราฟ Triangular Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง	54
4.13	Triangular Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ	55
4.14	กราฟ Triangular Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ	55
4.15	Left Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับสูง	55
4.16	กราฟ Left Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับสูง	56
4.17	Left Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง	56
4.18	กราฟ Left Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง	56
4.19	Left Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ	57
4.20	กราฟ Left Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ	57
4.21	Right Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับสูง	57
4.22	กราฟ Right Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับสูง	58
4.23	Right Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง	58
4.24	กราฟ Right Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง	59
4.25	Right Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ	59
4.26	กราฟ Right Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ	59
4.27	ตัวอย่างข้อมูลจำนวนประชากรโลกในรูปแบบของภาษาดาต้าล็อก	61
4.28	กราฟเปรียบเทียบค่าความเป็นสมาชิกระหว่างการใช้ฟังก์ชัน Trapezoidal Function กับ Triangular Function ในคำถาม หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 90,000,000 คน	74

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

4.29 กราฟเปรียบเทียบค่าความเป็นสมาชิกระหว่างการใช้ฟังก์ชัน Trapezoidal Function กับ Triangular Function ในคำถาม หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 25,000,000 คน 74
4.30 กราฟเปรียบเทียบค่าความเป็นสมาชิกระหว่างการใช้ฟังก์ชัน Trapezoidal Function กับ Triangular Function ในคำถาม หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,100,000 คน 75
4.31 กราฟเปรียบเทียบค่าความเป็นสมาชิกระหว่างการใช้ฟังก์ชัน Left Shoulder Function กับ Right Shoulder Function ในคำถาม หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 90,000,000 คน 75
4.32 กราฟเปรียบเทียบค่าความเป็นสมาชิกระหว่างการใช้ฟังก์ชัน Left Shoulder Function กับ Right Shoulder Function ในคำถาม หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 25,000,000 คน 76
4.33 กราฟเปรียบเทียบค่าความเป็นสมาชิกระหว่างการใช้ฟังก์ชัน Left Shoulder Function กับ Right Shoulder Function ในคำถาม หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,100,000 คน 76

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มาของปัญหาการวิจัย

ปัจจุบันมีการเก็บรวบรวมข้อมูลด้วยรูปแบบดิจิทัลมากขึ้น จึงทำให้เทคนิคการทำเหมืองข้อมูลได้รับความนิยมในเชิงธุรกิจมากขึ้นด้วยเช่นกัน ด้วยการประยุกต์ใช้เทคนิคการทำเหมืองข้อมูลในการค้นหาความรู้ใหม่ๆจากฐานข้อมูลที่มีอยู่ ทั้งนี้การทำเหมืองข้อมูลเป็นการค้นหารูปแบบหรือแพทเทิร์นของข้อมูลโดยอาศัยหลักการสถิติ การรู้จำ และหลักการทางคณิตศาสตร์

ในแต่ละองค์กรรวมไปถึงธุรกิจหลากหลายแขนง ปฏิเสธไม่ได้ว่าข้อมูลที่อยู่ในรูปแบบดิจิทัลมีความสำคัญมาก เพราะการค้นหาข้อมูลที่ต้องการสามารถทำได้สะดวก และรวดเร็วกว่าข้อมูลที่อยู่ในรูปแบบกระดาษค่อนข้างมาก รวมไปถึงการจัดเก็บข้อมูลของแต่ละองค์กร แต่ละธุรกิจก็จะยิ่งมีข้อมูลเพิ่มมากขึ้น การจัดเก็บข้อมูลในรูปแบบดิจิทัล ส่งผลให้แทบจะไม่มีการสูญหายของข้อมูลเลย ซึ่งแตกต่างกับการเก็บข้อมูลในรูปแบบกระดาษ ที่มีความเสี่ยงต่อการสูญหายของข้อมูลอยู่มาก

เทคนิคการประมวลผลกับข้อมูลที่มีขนาดใหญ่ มีความสำคัญค่อนข้างมากต่อความสะดวกรวดเร็วในการค้นหาข้อมูลที่ต้องการหรือใกล้เคียงกับความต้องการ และได้ข้อมูลที่มีความถูกต้องแม่นยำมากที่สุด การทำเหมืองข้อมูลช่วยในการค้นหาความรู้ใหม่ๆจากฐานความรู้เดิมที่มีอยู่ ส่งผลให้ลดความผิดพลาดจากการทำงานของมนุษย์ที่อาจเกิดจากความเมื่อยล้า และความบกพร่องในการจัดการกับข้อมูลที่มีขนาดใหญ่ เพื่อหาความรู้ใหม่ๆจากข้อมูลนั้น

ในเชิงธุรกิจ การจัดอันดับของข้อมูลที่ค้นหาได้จากฐานข้อมูลขนาดใหญ่มีความสำคัญเป็นอย่างมาก เนื่องจากจะช่วยให้สามารถกำหนดขอบเขตของกลุ่มเป้าหมายได้ตรงกับสินค้าที่ต้องการจำหน่าย หรือออกแบบและสร้างผลิตภัณฑ์ใหม่ๆได้ตรงตามความต้องการของตลาดในช่วงเวลา ณ ขณะนั้น โดยการจัดอันดับข้อมูลสามารถจัดอันดับได้ด้วยการระบุเงื่อนไขเพียงเงื่อนไขเดียว หรือระบุหลายเงื่อนไขก็ได้

โดยทั่วไป การตัดสินใจซื้อสินค้าและบริการของผู้บริโภคจะมีความคลุมเครืออยู่ ตัวอย่างเช่นการเลือกหาสถานที่พักที่ใกล้กับสถานที่ประชุมโดยมีเงื่อนไข "สถานที่พักจะต้องมีราคาถูกและจะต้องอยู่ใกล้กับสถานที่ประชุม"[6]จากตัวอย่างดังกล่าวจะเห็นได้ว่าคำว่า "มีราคาถูก

และอยู่ใกล้กับสถานที่ประชุม" ไม่สามารถระบุเป็นข้อความที่ชัดเจนได้ กล่าวคือไม่สามารถบอกได้ว่า "ราคาถูก" คือจำนวนเงินเท่าไร และ "อยู่ใกล้" คือระยะทางเท่าไร

การจัดการกับการสอบถามข้อมูลที่มีความคลุมเครือเพื่อหาคำตอบที่คลุมเครือน้อยที่สุดและตรงกับความต้องการของผู้ใช้ ด้วยการใช้เทคนิคการคำนวณค่าฟังก์ชันสมาชิกเข้ามาช่วยเพื่อลดความคลุมเครือของข้อมูลลง ส่งผลให้การจัดอันดับในข้อมูลที่มีความคลุมเครือมีความแม่นยำมากยิ่งขึ้น

จากตัวอย่างการเลือกหาสถานที่พักที่ใกล้กับสถานที่ประชุม หากไม่มีการนำเทคนิคการคำนวณค่าฟังก์ชันสมาชิกเข้ามาช่วยจะให้คำตอบที่ตรงกับข้อคำถามมีเพียงอย่างเดียว ตัวอย่างเช่น "ต้องการหาที่พักในราคา 500 บาท และอยู่ห่างจากที่ประชุม 300 เมตร" คำตอบที่ได้จะมีเพียงคำตอบที่ตรงตามเงื่อนไขที่กำหนดเท่านั้น แต่หากใช้เทคนิคการคำนวณค่าฟังก์ชันสมาชิกเข้ามาช่วยจะสามารถหาคำตอบได้ครอบคลุมมากกว่า กล่าวคืออาจจะมีคำตอบ "ที่พักราคา 490 บาทและอยู่ห่างจากที่ประชุม 290 เมตรหรือ ที่พักราคา 520 บาทและอยู่ห่างจากที่ประชุม 320 เมตร" เป็นต้น และหากนำเทคนิคการจัดอันดับ top-k ในข้อมูลที่มีความคลุมเครือจะทำให้คำตอบที่ได้ คือคำตอบที่ดีที่สุด เคอันดับแรกดังนั้นการใช้เทคนิคการคำนวณค่าฟังก์ชันสมาชิกและเทคนิคการจัดอันดับ top-k ในข้อมูลที่มีความคลุมเครือจะช่วยให้ผู้บริโภคมีทางเลือกในการตัดสินใจได้มากยิ่งขึ้น ทั้งนี้หากนำเทคนิค data mining เข้ามาร่วมในการจัดกลุ่มของข้อมูลก่อนนำข้อมูลนั้นไปจัดอันดับ top-k จะทำให้สามารถจัดอันดับ top-k เฉพาะช่วงข้อมูลที่เราสนใจได้

จากความสำคัญของการประมวลผลกับข้อมูลที่มีความคลุมเครือดังที่กล่าวมาแล้วข้างต้นนั้น ผู้วิจัยจึงเสนอเทคนิคการจัดอันดับของข้อมูลที่มีความคลุมเครือด้วยการคำนวณค่าฟังก์ชันสมาชิก เพื่อหาวิธีการจัดเรียงอันดับที่ดีที่สุดของข้อมูลให้มีความถูกต้องและแม่นยำมากขึ้น

1.2 วัตถุประสงค์ของการวิจัย

จากแนวคิดในการทำงาน ผู้วิจัยได้ตั้งวัตถุประสงค์ในการวิจัยดังนี้

1)เพื่อศึกษาและพัฒนาวิธีการจัดอันดับข้อมูลที่มีความคลุมเครือในฐานข้อมูลนิรนัยด้วยการใช้วิธีคำนวณค่าฟังก์ชันสมาชิกในรูปแบบต่างๆ

2) เพื่อเปรียบเทียบการจัดอันดับด้วยฟังก์ชันสมาชิกในแต่ละรูปแบบในฐานข้อมูลนิรนัย

3) เพื่อหาความเหมาะสมของฟังก์ชันสมาชิกแต่ละฟังก์ชันในการนำไปประยุกต์ใช้งาน

1.3 ข้อตกลงเบื้องต้น

1) ข้อมูลที่นำมาใช้จะต้องเป็นข้อมูลที่เหมาะสำหรับการจัดอันดับจึงจะเป็นประโยชน์ต่อ

การนำไปประยุกต์ใช้งาน

2)งานวิจัยนี้เลือกใช้ภาษาดาต้าล็อกในการพัฒนาโปรแกรม

3)ฟังก์ชันสมาชิกที่ใช้ในการทดลองประกอบไปด้วย4 ฟังก์ชัน ได้แก่ Trapezoidal Function,Triangular Function, Left Shoulder Function และ Right Shoulder Function

4) การที่จะได้คำตอบของการจัดอันดับที่ถูกต้อง ผู้ใช้จะต้องกำหนดช่วงของข้อมูลที่ต้องการลงในกฎของฟังก์ชันสมาชิกด้วย

5) การจัดอันดับที่ถูกต้องจะขึ้นอยู่กับการระบุค่าให้กับฟังก์ชันสมาชิก และการเลือกใช้ฟังก์ชันสมาชิกให้เหมาะสมกับสิ่งที่ต้องการ

1.4 ขอบเขตการวิจัย

จากการศึกษาค้นคว้าข้อมูล ผู้วิจัยได้กำหนดขอบเขตของการวิจัยไว้ดังนี้

1) ในการจัดอันดับจะเลือกใช้ฟังก์ชันสมาชิก Trapezoidal Function,Triangular Function, Left Shoulder Function และ Right Shoulder Function

2) การเปรียบเทียบประสิทธิภาพการจัดอันดับของข้อมูล จะเปรียบเทียบระหว่างการเลือกใช้ฟังก์ชันสมาชิกในแต่ละรูปแบบ โดยใช้ชุดข้อมูลเดียวกัน

3)งานวิจัยนี้ใช้ข้อมูลประชากรจากฐานข้อมูลธนาคารโลก (http://databank.worldbank.org/)ในการจัดอันดับข้อมูล

4)งานวิจัยนี้ใช้โปรแกรมดีอีเอส(Datalog Education System) ในการศึกษาและพัฒนา หากนำซอสโค้ด (Source Code) ที่พัฒนาด้วยโปรแกรมดีอีเอสนี้ ไปประมวลผลกับโปรแกรมอื่นอาจจะไม่สามารถประมวลผลได้

1.5 ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ที่เกิดขึ้นจากการศึกษาและพัฒนางานวิจัยนี้ ได้แก่

1) สามารถนำฟังก์ชันสมาชิกที่สร้างขึ้นไปประยุกต์ใช้ในขั้นตอนการจัดอันดับได้

2) โปรแกรมการจัดอันดับ top-k สามารถจัดอันดับข้อมูลตามที่กำหนดได้อย่างถูกต้อง

3)ผู้ใช้สามารถเลือกใช้ฟังก์ชันสมาชิกทั้ง 4 ชนิด (Triangular Function, Trapezoidal Function, Left Shoulder Function และ Right Shoulder Function) ได้เหมาะสมกับขนาดของข้อมูลหรือประเภทของงาน

4)ผู้ใช้สามารถกำหนดขอบเขตของช่วงของข้อมูลที่ต้องการนำมาจัดอันดับได้

บทที่ 2
ปริทัศน์วรรณกรรมและงานวิจัยที่เกี่ยวข้อง

เนื้อหาในบทนี้ ประกอบด้วยการทบทวนวรรณกรรมและงานวิจัยที่เกี่ยวข้อง โดยมีรายละเอียดของการทำเหมืองข้อมูล(Data Mining) ฟัซซี่ลอจิก(Fuzzy Logic) ภาษาสำหรับการเขียนโปรแกรมเชิงตรรกะ(Datalog) โปรแกรมดีอีเอส(DES) และงานวิจัยที่เกี่ยวข้อง

2.1 การทำเหมืองข้อมูล (Data Mining)

การทำเหมืองข้อมูลคือการกระทำสิ่งใดสิ่งหนึ่งกับข้อมูลจำนวนมากเพื่อค้นหารูปแบบและความสัมพันธ์ในชุดข้อมูลนั้นๆ เพื่อนำรูปแบบและความสัมพันธ์ที่ได้ไปใช้ในการทำนายข้อมูลในอนาคตหรือเป็นการแสดงให้เห็นรูปแบบและความสัมพันธ์ที่ปรากฏอยู่ในข้อมูลปัจจุบันได้ชัดเจนยิ่งขึ้น จึงส่งผลให้ในปัจจุบันมีการนำการทำเหมืองข้อมูลไปใช้ประโยชน์ในเชิงธุรกิจได้อย่างกว้างขวาง

2.1.1 วิวัฒนาการของการประมวลผลและจัดการกับข้อมูล

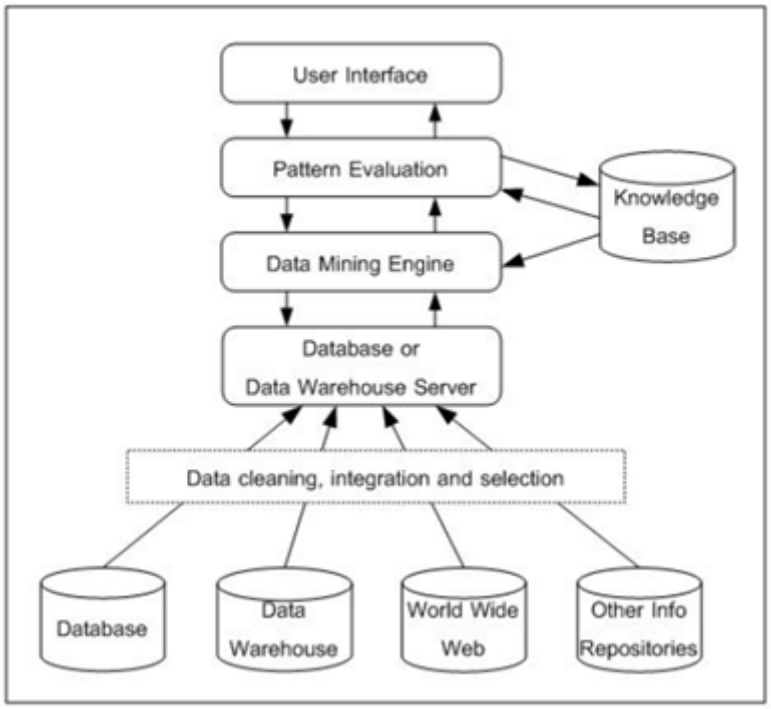
- ปี 1960 มีการนำข้อมูลมาจัดเก็บในอุปกรณ์ที่มีความน่าเชื่อถือและสามารถป้องกันการสูญหายของข้อมูลได้
- ปี 1980 มีการนำข้อมูลที่จัดเก็บมาสร้างความสัมพันธ์ต่อกันเพื่อนำไปช่วยเพิ่มประสิทธิภาพในการวิเคราะห์และตัดสินใจให้มีคุณภาพมากขึ้น
- ปี 1990 มีการรวบรวมข้อมูลมาจัดเก็บลงในฐานข้อมูลขนาดใหญ่โดยครอบคลุมส่วนประกอบในทุกๆด้านขององค์กรเพื่อช่วยสนับสนุนในการตัดสินใจ
- ปี 2000 มีการนำข้อมูลออกจากฐานข้อมูลเพื่อนำข้อมูลเหล่านั้นมาวิเคราะห์และประมวลผล โดยการสร้างแบบจำลองและความสัมพันธ์ทางสถิติ

2.1.2 วัตถุประสงค์ในการทำเหมืองข้อมูล

1. เพื่อค้นหาความรู้ใหม่ในฐานข้อมูล (Knowledge discovery in database)
2.เพื่อวิเคราะห์และแยกแยะความรู้ที่ซ่อนเร้นอยู่ (Knowledge extraction)
3. เพื่อจัดการกับข้อมูลในอดีต (Data archeology)

4. เพื่อสำรวจข้อมูล (Data exploration)
5. เพื่อค้นหารูปแบบของข้อมูลที่ซ่อนอยู่ (Data pattern processing)
6. เพื่อใช้ขุดเจาะข้อมูลในเชิงลึกเพื่อให้ได้มาซึ่งสารสนเทศที่มีประโยชน์

2.1.3 สถาปัตยกรรมของการทำเหมืองข้อมูล



รูปที่ 2.1สถาปัตยกรรมระบบการทำเหมืองข้อมูล[5]

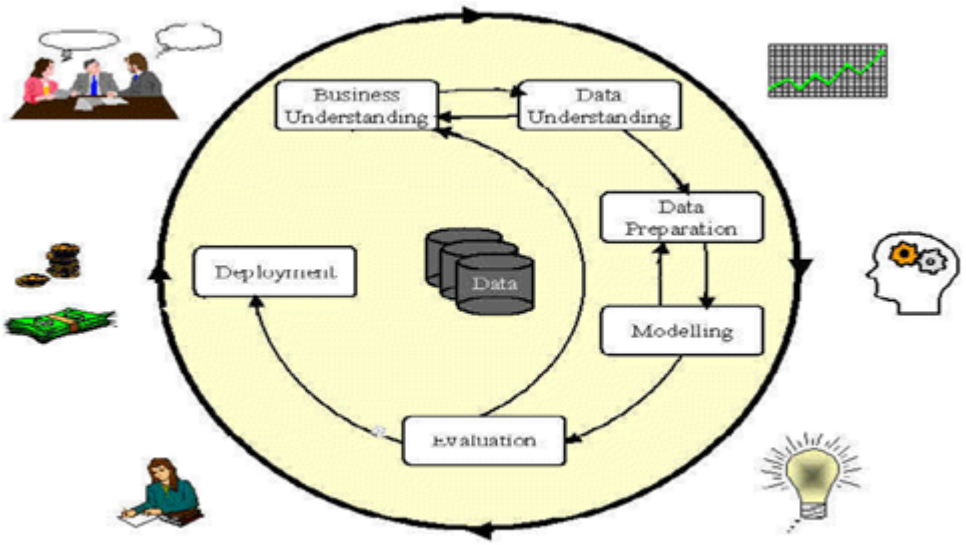
จากรูปที่ 2.1 สามารถอธิบายสถาปัตยกรรมระบบการทำเหมืองข้อมูลได้ดังนี้

- Database, Data Warehouse, World Wide Web และ Other Info Repositories คือแหล่งเก็บข้อมูลต่างๆ สำหรับนำมาทำเหมืองข้อมูลเพื่อให้ได้ Knowledge base ต่างๆ
- Database หรือ Data Warehouse Server ทำหน้าที่ค้นหาและนำเข้าข้อมูลตามคำสั่งจากผู้ใช้
- Knowledge base คือองค์ความรู้ใหม่ๆที่เป็นประโยชน์ต่อการค้นหา
- Data Mining Engine เป็นส่วนประกอบหลักของการทำเหมืองข้อมูล ซึ่งประกอบด้วยโมดูลที่ทำเหมืองข้อมูลประเภทต่างๆ
- Pattern Evaluation Module จะทำงานร่วมกันกับ Data Mining Engine โดยนำมาตร

วัดความน่าสนใจของผลลัพธ์ที่ได้ เพื่อให้การค้นหามุ่งเน้นเฉพาะรูปแบบที่น่าสนใจ

- Graphic User Interface เป็นส่วนติดต่อประสานงานระหว่างผู้ใช้กับระบบการทำเหมืองข้อมูล

2.1.4 ขั้นตอนการทำเหมืองข้อมูล



รูปที่ 2.2กระบวนการทำเหมืองข้อมูล [11]

จากรูปที่ 2.2 สามารถอธิบายกระบวนการทำเหมืองข้อมูลในแต่ละขั้นตอนได้ดังนี้

1. Business Understanding คือขั้นตอนที่สำคัญที่สุด เนื่องจากการทำเหมืองข้อมูลที่ดีนั้นจะต้องเข้าใจวัตถุประสงค์ของการทำเหมืองข้อมูลนั้นๆว่าต้องการหาอะไรจากข้อมูล

2.Data Understanding คือขั้นตอนของการรวบรวมข้อมูลที่เกี่ยวข้องที่ใช้ในการทำเหมืองข้อมูล

3.Data Preparation คือขั้นตอนของการเตรียมข้อมูลเพื่อนำไปใช้ในการทำเหมืองข้อมูล เนื่องจากโมเดลที่ได้จากการทำเหมืองข้อมูลนั้นจะให้ผลลัพธ์ที่ถูกต้องหรือไม่นั้นขึ้นอยู่กับคุณภาพของข้อมูลที่ใช้ การเตรียมข้อมูลสามารถแบ่งออกได้เป็น 3 ขั้นตอนดังนี้

- Data Selection คือการเลือกใช้เฉพาะข้อมูลที่เกี่ยวข้องกับสิ่งที่ต้องการวิเคราะห์
- Data Cleaning คือการกรองข้อมูลที่ไม่ถูกต้องหรือซ้ำซ้อนออก
- Data Transformation คือการเตรียมข้อมูลให้อยู่ในรูปแบบที่พร้อมนำไปใช้ในการวิเคราะห์

4. Modeling คือขั้นตอนการวิเคราะห์ข้อมูลเพื่อหารูปแบบข้อมูล ได้แก่การสร้างตัวทำนาย

(prediction model) ซึ่งในบางครั้งอาจมีการนำเทคนิคการทำเหมืองข้อมูลหลายเทคนิคมาใช้ในการวิเคราะห์เพื่อให้ได้ผลลัพธ์ที่ดีที่สุด จึงอาจจะมีการย้อนกลับไปทำงานที่ขั้นตอนData Preparation อีกครั้งเพื่อแปลงข้อมูลบางส่วนให้เหมาะสมกับแต่ละเทคนิค

5. Evaluation คือขั้นตอนการประเมินผลหรือวัดประสิทธิภาพของโมเดล

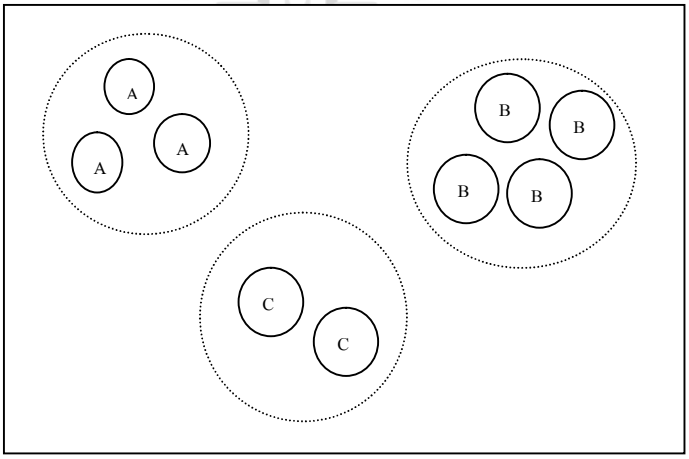
6. Deployment คือขั้นตอนของการนำองค์ความรู้ที่ได้ไปใช้งานจริง

2.1.5 เทคนิคการจัดกลุ่ม (Clustering)

เทคนิคการจัดกลุ่ม [1,12]คือการจัดข้อมูลที่มีความคล้ายคลึงกันให้อยู่ในกลุ่มเดียวกัน ซึ่งมีลักษณะคล้ายกับการแบ่งประเภท แต่จะแตกต่างกันโดยการแบ่งประเภทจะวิเคราะห์ข้อมูลตามต้นแบบ แต่สำหรับการแบ่งกลุ่มจะเป็นการวิเคราะห์โดยจะไม่พิจารณาจัดกลุ่มตามประเภทที่รู้จัก แต่จะใช้ขั้นตอนวิธีอัตโนมัติในการจัดกลุ่มเพื่อค้นหากลุ่มที่สามารถยอมรับได้เพื่อจัดเข้ากลุ่ม

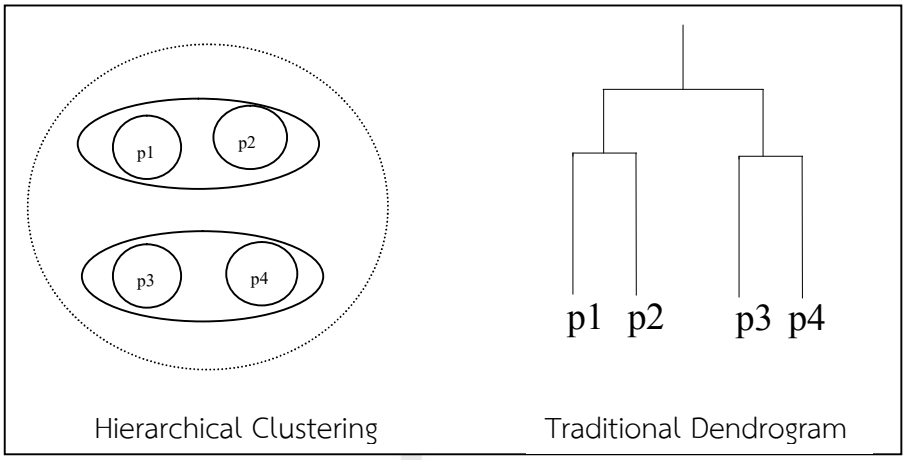
ประเภทของการจัดกลุ่ม

1) Partitional Clustering คือการแบ่งกลุ่มอย่างชัดเจนโดยไม่มีกลุ่มไหนซ้อนทับกันอยู่ ดังรูปที่ 2.3



รูปที่ 2.3 Partitional Clustering

2) Hierarchical clustering คือการแบ่งกลุ่มแบบที่มีความสัมพันธ์แบบลำดับชั้น ดังรูปที่ 2.4



รูปที่ 2.4 Hierarchical clustering

k-Means Clustering

การจัดกลุ่มด้วยเทคนิค k-Means [13] ใช้หลักการการตัดแบ่ง (Partition) วัตถุ n ตัวในฐานข้อมูล D ออกเป็นจำนวน k กลุ่ม (ในกรณีที่ทราบค่า k) อัลกอริทึม k-Means จะตัดแบ่งวัตถุออกเป็น k กลุ่ม โดยแทนแต่ละกลุ่มด้วยค่าเฉลี่ยของกลุ่ม ซึ่งจะใช้ค่าเฉลี่ยนี้เป็นจุดศูนย์กลางของกลุ่มในการวัดระยะห่างของตัวอย่างในกลุ่มเดียวกัน

ขั้นตอนการแบ่งกลุ่มด้วย k-Means

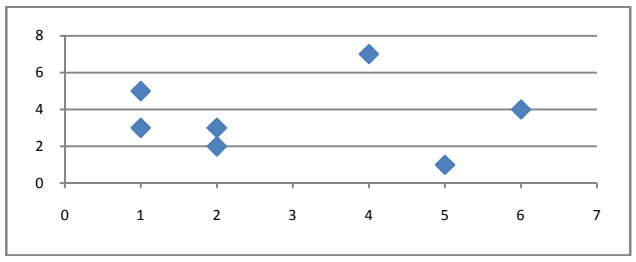
1) สุ่มค่าเริ่มต้นจำนวน k ค่า เรียกว่า cluster centers หรือจุด centroid

2) หาระยะห่างระหว่างข้อมูลกับจุดศูนย์กลางแต่ละกลุ่ม หากข้อมูลตัวไหนใกล้ค่าจุดศูนย์กลางตัวใดมากที่สุด จะถูกจัดอยู่ในกลุ่มนั้นๆ

3) หาค่าเฉลี่ย (Mean) ของแต่ละกลุ่ม เพื่อใช้เป็นค่าจุดศูนย์กลาง (centroid) ใหม่

4)ทำซ้ำตั้งแต่ขั้นตอนที่ 2 ถึง 4จนกระทั่งค่าเฉลี่ยในแต่ละกลุ่มไม่มีการเปลี่ยนแปลง

ตัวอย่างการทำ k-Meansแสดงดังรูปที่ 2.5



(a) ข้อมูลในลักษณะจุดบนแกนสองมิติ

ข้อมูล	จุด X	จุด Y
X1	1	5
X2	4	7
X3	2	3
X4	5	1
X5	2	2
X6	6	4

(b) ข้อมูลในลักษณะตาราง

รูปที่ 2.5ตัวอย่างข้อมูลสำหรับการทำ k-means

จากรูปที่ 2.5 สมมติให้ k = 2 และสุ่มเลือกc1 และ c2 เป็นจุด centroid เริ่มต้นที่ตำแหน่ง c1(2,2) และ c2(5,4)

หาระยะห่างระหว่างข้อมูลแต่ละจุดโดยอ้างอิงจากจุด centroid ด้วยนิยาม

$$\textbf{distance}\{(x_1,y_1),(x_2,y_2)\} = |x_2 - x_1| + |y_2 - y_1|$$

เมื่อคำนวณระยะห่างระหว่างข้อมูลแต่ละจุดโดยอ้างอิงจากจุด centroid ในรอบที่ 1 ข้อมูลจะถูกจัดกลุ่มดังตารางที่ 2.1

ตารางที่ 2.1 ข้อมูลหลังผ่านการจัดCluster รอบที่ 1

จุด	ตำแหน่ง	ระยะห่างจากC1	ระยะห่างจากC2	Cluster
X1	(1,5)	4	5	1
X2	(4,7)	7	4	2
X3	(2,3)	1	4	1
X4	(5,1)	4	3	2
X5	(2,2)	0	5	1
X6	(6,4)	6	1	2
X7	(1,3)	2	5	1

หาค่าเฉลี่ยของแต่ละกลุ่มเพื่อหาจุด centroid ใหม่

กลุ่มที่ 1 จะได้ c1((1+2+2+1)/4 , (5+3+2+3)/4) = c1(1.5 , 3.25)

กลุ่มที่ 2 จะได้ c2((4+5+6)/3 , (7+1+4)/3) = c2(5 , 4)

ผลที่ได้จากการทำงานรอบที่ 2 แสดงดังตารางที่ 2.2

ตารางที่ 2.2ข้อมูลหลังผ่านการจัดCluster รอบที่ 2

จุด	ตำแหน่ง	ระยะห่างจาก C1	ระยะห่างจาก C2	Cluster
X1	(1,5)	**2.25**	5	1
X2	(4,7)	6.25	**4**	2
X3	(2,3)	**0.75**	4	1
X4	(5,1)	5.75	**3**	2
X5	(2,2)	**1.75**	5	1
X6	(6,4)	5.25	**1**	2
X7	(1,3)	**0.75**	5	1

หาค่าเฉลี่ยของแต่ละกลุ่มเพื่อหาจุด centroid ใหม่

กลุ่มที่ 1 จะได้ c1((1+2+2+1)/4 , (5+3+2+3)/4) = c1(1.5 , 3.25)

กลุ่มที่ 2 จะได้ c2((4+5+6)/3 , (7+1+4)/3) = c2(5 , 4)

จะเห็นได้ว่าจุด centroid ไม่มีการเปลี่ยนแปลง ดังนั้นจะได้กลุ่มของข้อมูลดังตารางที่ 2.3

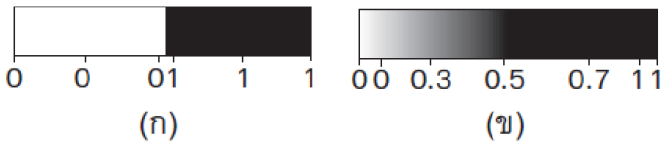
ตารางที่ 2.3แสดงข้อมูลที่ผ่านกระบวนการแบ่งกลุ่มด้วย k-Means

กลุ่มที่ 1	กลุ่มที่ 2
X1(1,5)	X2(4,7)
X3(2,3)	X4(5,1)
X5(2,2)	X6(6,4)
X7(1,3)	

2.2 ฟัซซี่ลอจิก (Fuzzy Logic)

ฟัซซี่ลอจิก (Fuzzy Logic) [3]หรือตรรกศาสตร์คลุมเครือ คือการใช้เหตุผลแบบประมาณ ซึ่งจะแตกต่างจากการใช้เหตุผลที่ชัดเจนในลักษณะใช่/ไม่ใช่ ของตรรกศาสตร์แบบบูลีน(Boolean logic) ฟัซซี่ลอจิกนั้นสามารถจำลองการตัดสินใจต่อปัญหาที่มีความซับซ้อนและไม่ชัดเจนได้

2.2.1 ข้อดีของฟัซซี่ลอจิก



รูปที่ 2.6ความแตกต่างระหว่าง (ก) เซ็ตชัดเจน (ข) ฟัซซี่เซต[5]

จากรูปที่ 2.6 จะเห็นข้อแตกต่างระหว่างเซ็ตชัดเจนกับฟัซซี่เซตที่ใช้หลักการของฟัซซี่ลอจิกดังนี้

- ฟัซซี่ลอจิกใช้ค่าจำนวนจริงในการระบุระดับความเป็นสมาชิก (degree of membership) เพื่อแทนค่าระดับความจริง ซึ่งจะอยู่ในช่วงระหว่าง 0 ถึง 1 ในขณะที่ตรรกศาสตร์แบบบูลีนนั้นจะประเมินค่าความจริงเป็นจำนวนเต็ม0 หรือ 1 ซึ่งจะแทนค่าจริงหรือเท็จเท่านั้น

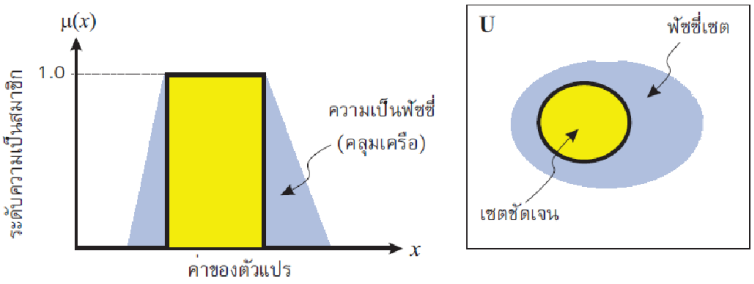
- ฟัซซี่ลอจิกมีเสถียรภาพค่อนข้างสูง สามารถรับคำสั่งที่มีความคลุมเครือได้ ในขณะที่คำสั่งที่อินพุตเข้าไปในตรรกศาสตร์แบบบูลีนนั้น จะต้องมีความชัดเจนและแน่นอน

- ฟัซซี่ลอจิกจะทำการประมวลผลด้วยการใช้กฎที่ผู้ใช้งานอินพุตเข้าไป กล่าวคือฟัซซี่ลอจิกจะประมวลผลด้วยกฎและนิยามที่ผู้ใช้ทำการอินพุตเข้ามา

- ฟัซซี่ลอจิกมีการรับอินพุตที่คลุมเครือเข้ามาและจะแสดงเอาต์พุตที่ค่อนข้างชัดเจนมากที่สุดออกไป

- ฟัซซี่ลอจิกจะกำหนดค่าระดับความจริงในการระบุค่าความเป็นสมาชิก (degree of membership) ด้วยฟังก์ชันสมาชิก(membership function)

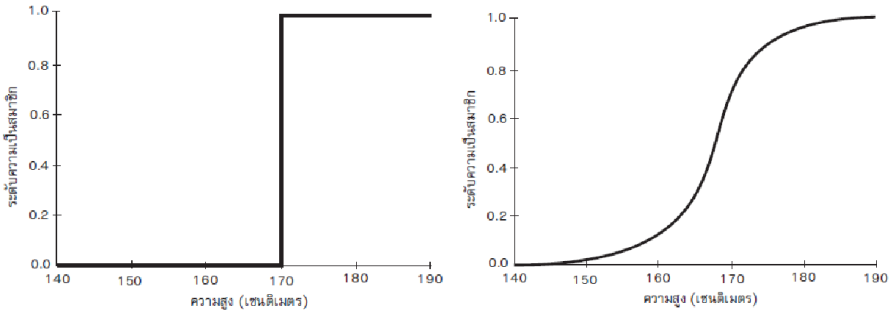
2.2.2 ฟัซซี่เซต (Fuzzy Set)



รูปที่ 2.7แสดงขอบเขตของฟัซซี่เซต

ฟัซซี่เซต (Fuzzy Set) [14]เป็นเซตที่มีการระบุขอบเขตของข้อมูลไม่เด่นชัดหรือคลุมเครือดังรูปที่ 2.7โดยฟัซซี่เซตจะแตกต่างจากเซตชัดเจน (Crisp Set) ดังรูปที่ 2.8 โดยที่เซตชัดเจนจะมีค่า

ความเป็นสมาชิกเพียงสองค่า คือ ใช่/ไม่ใช่ หรือ จริง/ไม่จริง ส่วนในฟัซซี่เซตจะใช้ฟังก์ชันสมาชิก (Membership Function) ในการกำหนดค่าระดับความจริงในการเป็นสมาชิกของเซตนั้น ๆ



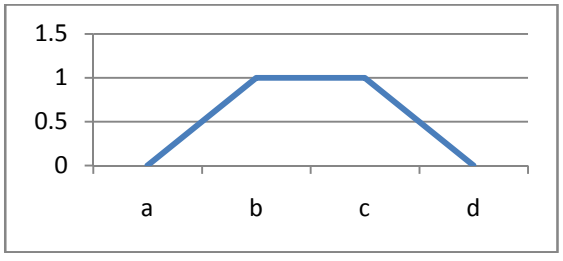
รูปที่ 2.8แสดงความแตกต่างของเซตชัดเจน (ซ้าย) กับฟัซซี่เซต(ขวา)

2.2.3 ฟังก์ชันสมาชิก (Membership Function)

ฟังก์ชันสมาชิก[4]คือฟังก์ชันที่ใช้คำนวณเพื่อช่วยในการระบุค่าความเป็นสมาชิกของฟัซซี่เซต ฟังก์ชันที่นิยมใช้กันได้แก่ Trapezoidal Function และ Triangular Function นอกจากนี้ยังมีฟังก์ชันพิเศษเพิ่มขึ้นมาอีกสองฟังก์ชัน ได้แก่ Left Shoulder Function และ Right Shoulder Function

1. Trapezoidal Function

ฟังก์ชันชนิดนี้จะมีรูปร่างเป็นสี่เหลี่ยมคางหมู โดยรูปร่างจะขึ้นอยู่กับตัวแปรทั้ง 4 ตัว ได้แก่ a, b, c และ d ดังรูปที่ 2.9



$$\mu(x) = \begin{cases} 0, & (x < a) \text{ or } (x > d) \\ \dfrac{x-a}{b-a}, & a \le x \le b \\ 1, & b \le x \le c \\ \dfrac{d-x}{d-c}, & c < x < d \end{cases}$$

รูปที่ 2.9แสดงขอบเขตและวิธีการหาค่าความเป็นสมาชิกของ Trapezoidal Function

จากรูปที่ 2.9 สามารถอธิบายตัวแปรต่าง ๆ ได้ดังนี้

a คือค่าที่น้อยที่สุดของการพิจารณาความเป็นสมาชิกในเซต

d คือค่าที่มากที่สุดของการพิจารณาความเป็นสมาชิกในเซต

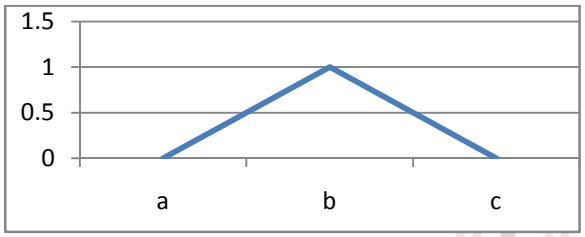
b คือค่าที่น้อยที่สุดที่ทำให้ค่าความเป็นสมาชิกเป็น 1

c คือค่าที่มากที่สุดที่ทำให้ค่าความเป็นสมาชิกเป็น 1

โดยที่ a < b < c < d

2. Triangular Function

ฟังก์ชันชนิดนี้จะมีรูปร่างเป็นสามเหลี่ยม ซึ่งขึ้นอยู่กับค่าตัวแปร 3 ตัว ได้แก่ a, b และ cดังรูปที่ 2.10



$$\mu(x) = \begin{cases} 0, & x \le a \text{ or } x \ge c \\ \dfrac{x-a}{b-a}, & a < x \le b \\ \dfrac{c-x}{c-b}, & b < x < c \\ 1, & x = b \end{cases}$$

รูปที่ 2.10แสดงขอบเขตและวิธีการหาค่าความเป็นสมาชิกของ Triangular Function

จากรูปที่ 2.10 สามารถอธิบายตัวแปรต่าง ๆ ได้ดังนี้

a คือค่าที่น้อยที่สุดของการพิจารณาความเป็นสมาชิกในเซต

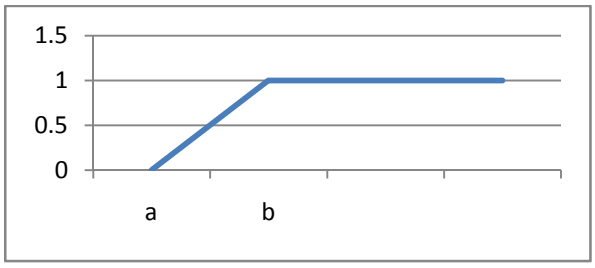
cคือค่าที่มากที่สุดของการพิจารณาความเป็นสมาชิกในเซต

bคือค่าที่สนใจซึ่งจะเป็นจุดสูงสุดของระดับความเป็นสมาชิก

โดยที่ a <b < c

3. Left Shoulder Function

ฟังก์ชันชนิดนี้จะมีรูปร่างเอียงลงทางซ้าย ซึ่งขึ้นอยู่กับค่าตัวแปร 2 ตัว ได้แก่ a และ b ดังรูปที่ 2.11



$$\mu(x) = \begin{cases} 0, & x < a \\ \dfrac{x-a}{b-a}, & a \le x \le b \\ 1, & x > b \end{cases}$$

มหาวิทยาลัยเทคโนโลยีสุรนารี

รูปที่ 2.11แสดงขอบเขตและวิธีการหาค่าความเป็นสมาชิกของ Left Shoulder Function

จากรูปที่ 2.11 สามารถอธิบายตัวแปรต่าง ๆ ได้ดังนี้

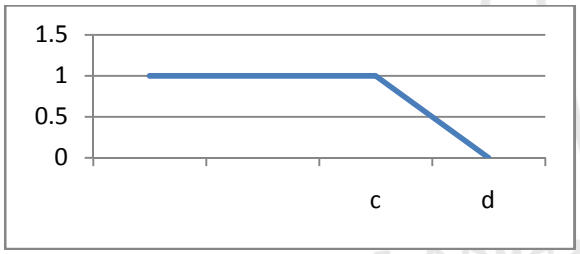
a คือค่าที่น้อยที่สุดของการพิจารณาความเป็นสมาชิกในเซต

b คือค่าที่เราสนใจ

โดยที่ a < b

4. Right Shoulder Function

ฟังก์ชันชนิดนี้จะมีรูปร่างเอียงลงทางขวา ซึ่งขึ้นอยู่กับตัวแปร 2 ตัว ได้แก่ c และ d ดังรูปที่ 2.12



$$\mu(x) = \begin{cases} 0, & x < d \\ \dfrac{d-x}{d-c}, & c \le x \le d \\ 1, & x > c \end{cases}$$

รูปที่ 2.12แสดงขอบเขตและวิธีการหาค่าความเป็นสมาชิกของ Right Shoulder Function

จากรูปที่ 2.12 สามารถอธิบายตัวแปรต่าง ๆ ได้ดังนี้

c คือค่าที่เราสนใจ

d คือค่าที่มากที่สุดของการพิจารณาความเป็นสมาชิกในเซต

โดยที่ c < d

2.3 ภาษาโปรล็อก (Prolog)และข้อมูลคลุมเครือ

ภาษาโปรล็อกเป็นภาษาสำหรับการเขียนโปรแกรมเชิงตรรกะ ได้รับการพัฒนาในปี ค.ศ. 1972 โดยมีพื้นฐานมาจากแคลคูลัสภาคแสดง (Predicate Calculus) นิยมนำไปใช้ในการแก้ปัญหาที่เป็นการคำนวณเชิงสัญลักษณ์โดยใช้พื้นฐานของตรรกศาสตร์อันดับหนึ่ง (First-Order Logic)

ภาษาโปรล็อกมีขั้นตอนและกระบวนการที่ใช้ข้อมูลในกลุ่มตัวอย่าง เพื่อหาโมเดลที่ใช้ในการคาดเดาคำตอบของคำถามที่ระบุเข้ามาโดยอ้างอิงจากชุดข้อมูลในกลุ่มตัวอย่างดังตารางที่ 2.4

ตารางที่ 2.4ชุดข้อมูลตัวอย่าง

วัน	เมฆมาก/เมฆน้อย	แดดออก/แดดไม่ออก	ฝนตก/ฝนไม่ตก
จันทร์	เมฆน้อย	แดดออก	ฝนไม่ตก
อังคาร	เมฆน้อย	แดดออก	ฝนไม่ตก
พุธ	เมฆมาก	แดดออก	ฝนไม่ตก
พฤหัส	เมฆน้อย	แดดไม่ออก	ฝนตก
ศุกร์	เมฆมาก	แดดออก	ฝนตก
เสาร์	เมฆมาก	แดดออก	ฝนไม่ตก
อาทิตย์	เมฆน้อย	แดดไม่ออก	ฝนตก

จากตารางที่ 2.4 จะได้โมเดลดังนี้

ถ้ามีเมฆน้อย และมีแดดออก ฝนจะไม่ตก

ถ้ามีเมฆมาก และมีแดดออก ฝนจะไม่ตก

ถ้ามีเมฆมาก และแดดไม่ออก ฝนจะตก

หากถามคำถามว่า "วันที่มีเมฆน้อย แดดออก ฝนจะตกหรือไม่" จะได้คำตอบว่า ฝนไม่ตก

หากถามคำถามว่า "วันที่มีเมฆมาก แดดออก ฝนจะตกหรือไม่" จะได้คำตอบว่า ฝนไม่ตก

จะมีกรณีเดียวที่ชุดโมเดลชุดนี้จะตอบคำตอบว่าฝนตกคือการถามว่า "วันที่มีเมฆมาก แดดไม่ออก ฝนจะตกหรือไม่"

จากข้อมูลตัวอย่าง จะเห็นได้ว่า ยังมีบางคำตอบที่มีโอกาสผิดอยู่ ซึ่งจะต้องมีการคำนวณความแม่นยำของแต่ละโมเดล เพื่อหาโมเดลที่สามารถทำนายผลได้แม่นยำมากยิ่งขึ้น

2.4 ภาษาดาต้าล็อก(Datalog)

ภาษาดาต้าล็อก (Datalog) [6]เป็นภาษาสำหรับการเขียนโปรแกรมเชิงตรรกะ สร้างขึ้นโดย Edgar Frank Codd โดยภาษาด้าต้าล็อกเป็นภาษาย่อยของภาษาโปรล็อก จึงมีโครงสร้างภาษาใกล้เคียงกัน

ภาษาดาต้าล็อกมีขั้นตอนและกระบวนการในการหาโมเดลจากข้อมูลที่มีอยู่ในฐานความรู้ และการหาความรู้ใหม่จากกฎที่ระบุเข้าไปในฐานความรู้ ในกรณีที่ข้อมูลไม่อยู่ในฐานความรู้ ภาษาดาต้าล็อกจะถือว่าข้อมูลเหล่านั้นไม่เป็นจริงเสมอ

2.4.1 ชนิดข้อมูลในภาษาดาต้าล็อก

1. อะตอม (Atom)

อะตอมคือค่าที่แทนด้วยข้อความที่มีขนาดเล็กที่สุด โดย 1 อะตอม จะประกอบไปด้วยเพรดิเคท (Predicate) และพจน์(Term) เช่น

path(A,B).

- เพรดิเคท(Predicate) คือค่าที่แทนด้วยข้อความ จะประกอบไปด้วยตัวอักษร ตัวเลข เส้นใต้ อักขระ โดยมีข้อบังคับ คือจะต้องขึ้นต้นด้วยตัวอักษรพิมพ์เล็กเท่านั้น ในตัวอย่างนี้ชื่อเพรดิเคทคือ path

- พจน์ (Term) คือส่วนประกอบภายในของเพรดิเคท โดยสามารถเป็นได้ทั้งค่าคงที่ (เขียนด้วยตัวพิมพ์เล็ก) หรือตัวแปร(เขียนด้วยตัวพิมพ์ใหญ่)ในตัวอย่างนี้ พจน์คือตัวแปร A และ B

2. ข้อเท็จจริง (Facts)

ข้อเท็จจริง คือความจริงที่เก็บอยู่ในฐานความรู้ และสามารถตอบคำถามของผู้ใช้ที่ป้อนคำถามเข้ามาเพื่อถามหาข้อเท็จจริงนั้นได้ทันที กล่าวคือค่าทุกค่าที่ปรากฏอยู่ในข้อเท็จจริงจะเป็นส่วนที่เล็กที่สุดของอะตอม หรือเรียกว่า กราวอะตอม (Ground Atom) เช่น

```
edge (a,b).
edge (a,c).
edge (b,d).
```

จากข้อเท็จจริงทั้งสามข้อข้างต้น จะสามารถทราบโดยทันทีว่ามีสิ่งใดที่เป็นจริงบ้าง หากผู้ใช้ถามคำถามว่า edge (a,b). เป็นจริงหรือไม่จะทราบได้ทันทีว่าเป็นจริง เนื่องจากมีข้อเท็จจริงที่ว่า edge (a,b) เป็นจริงปรากฏอยู่ในฐานความรู้ แต่หากผู้ใช้ถามคำถามว่า edge (a,d). เป็นจริงหรือไม่ คำตอบคือไม่เป็นจริง เนื่องจากข้อเท็จจริงที่ว่า edge (a,d) เป็นจริงไม่ได้ปรากฏอยู่ในฐานความรู้โดยในฐานความรู้นี้สามารถหาข้อเท็จจริงได้ 3 ข้อ ดังนี้

```
edge (a,b).
edge (a,c).
edge (b,d).
```

3. กฎ(Rule)

กฎคือรูปแบบของความจริงโดยอาศัยเงื่อนไขในการหาคำตอบ เช่น

```
path(X,Y) :- edge (X,Y).
```

ซึ่งโดยทั่วไป กฎที่เขียนขึ้นมักจะนำไปสู่การหาความรู้ใหม่ๆได้ เช่น

```
edge (a,b).
edge (a,c).
edge (b,d).
path(X,Y) :- edge (X,Y).
```

ทำให้ได้ข้อเท็จจริงดังต่อไปนี้

ข้อเท็จจริงที่ได้จากฐานความรู้ที่มีอยู่

```
edge (a,b).
edge (a,c).
edge (b,d).
```

และข้อเท็จจริงที่ได้จากกฎpath(X,Y) :- edge (X,Y).

```
path(a,b).
path(a,c).
path(b,d).
```

2.4.2 กลไกการทำงานและนิยามในดาต้าล็อก

1. การเรียกซ้ำ (Recursive)

เป็นเครื่องมือสำคัญของภาษาดาต้าล็อก เนื่องจากภาษาดาต้าล็อกไม่มีคำสั่งวนซ้ำ จึงจำเป็นต้องอาศัยการเขียนกฎที่มีการเรียกซ้ำเข้ามาช่วยในการหาความรู้ใหม่ๆ เช่น

```
edge (a,b).
edge (a,c).
edge (b,d).
path(X,Y) :- edge (X,Y).
path(X,Z) :- path(X,Y), edge (Y,Z).
```

ข้อเท็จจริงที่ได้จากฐานความรู้ที่มีอยู่

edge (a,b).

edge (a,c).

edge (b,d).

ข้อเท็จจริงที่ได้จากกฎpath(X,Y) :- edge (X,Y).

path(a,b).

path(a,c).

path(b,d).

และข้อเท็จจริงที่ได้จากการเรียกซ้ำจากกฎpath(X,Z) :- path(X,Y), edge(Y,Z).

path(a,d).

2. นิเสธ (Negation)

การใช้นิเสธ คือการที่เราต้องการให้ความสัมพันธ์ที่มีอยู่นั้นไม่เป็นจริง โดยจะใช้สัญลักษณ์ นิเสธ หรือ notดังตัวอย่างเช่น

```
edge (a,b).
edge (a,c).
edge (b,d).
path(X,Y) :- edge (X,Y), not edge(a,c).
```

ข้อเท็จจริงที่ได้จากกฎpath(X,Y) :- edge(X,Y), not edge(a,c). คือ

path(a,b),

path(b,d).

3. Extensional Database (EDB)

EDB คือเซตของข้อเท็จจริงที่ได้จากฐานความรู้ เช่น

```
edge(a,b).
edge(a,c).
edge(b,d).
```

จะได้เซตของ EDB คือ{edge(a,b), edge(a,c), edge(b,d)}

4. Intensional Database(IDB)

IDB คือเซตของกฎที่ได้จากจากฐานความรู้ เช่น

```
path(X,Y) :- edge(a,b).
path(X,Z) :- path(X,Y), edge(Y,Z).
```

จะได้เซตของ IDB คือ{(path(X,Y):-edge(a,b)), (path(X,Z) :- path(X,Y), edge(Y,Z))}

5. Herbrand Universe

เฮอร์แบรนด์ยูนิเวอร์ส คือเซตของค่าคงที่ทั้งหมดที่อยู่ในฐานความรู้ เช่น สมมุติว่ามีฐานความรู้ดังต่อไปนี้

```
edge (a,b).
edge (a,c).
edge (b,d).
```

จะได้ Herbrand Universe = {a, b, c, d}

6. Herbrand Base

เฮอร์แบรนด์เบส คือเซตของกราวอะตอมที่แทนค่าคงที่ที่ปรากฏใน Herbrand Universe โดยที่ไม่สนใจข้อเท็จจริงที่เกิดขึ้น เช่นสมมุติมีฐานความรู้ดังนี้

```
edge (a,b).
edge (a,c).
edge (b,d).
```

จะได้ Herbrand Universe = {a, b, c, d}

และ Herbrand Base = {

edge(a,a), edge(a,b), edge(a,c), edge(a,d),

edge(b,a), edge(b,b), edge(b,c), edge(b,d),

edge(c,a), edge(c,b), edge(c,c), edge(c,d),

edge(d,a), edge(d,b), edge(d,c), edge(d,d)

}

7. Minimal Model

คือเซตของคำตอบทั้งหมดที่เป็นไปได้โดยสนใจข้อเท็จจริงที่เกิดขึ้น ซึ่งเซ็ตคำตอบของ Minimal Model จะปรากฏอยู่ในเซตคำตอบของ Herbrand Base ด้วยเช่นกัน เช่น สมมุติว่ามีฐานความรู้ดังต่อไปนี้

```
edge (a,b).
edge (a,c).
edge (b,d).
path(X,Y) :- edge(X,Y).
path(X,Z) :- path(X,Y), edge(Y,Z).
```

จะได้ Herbrand Univers = {a, b, c, d}

Herbrand Base = {

edge(a,a), edge(a,b), edge(a,c), edge(a,d),

edge(b,a), edge(b,b), edge(b,c), edge(b,d),

edge(c,a), edge(c,b), edge(c,c), edge(c,d),

edge(d,a), edge(d,b), edge(d,c), edge(d,d),

path(a,a), path(a,b), path(a,c), path(a,d),

path(b,a), path(b,b), path(b,c), path(b,d),

path(c,a), path(c,b), path(c,c), path(c,d),

path(d,a), path(d,b), path(d,c), path(d,d)

}

และ Minimal Model = {

มหาวิทยาลัยเทคโนโลยีสุรนารี

edge(a,b), edge(a,c), edge(b,d),

path(a,b), path(a,c), path(b,d),

path(a,d)

}

2.5 ดาต้าล็อก$^{top-k}$

โดยทั่วไปแล้วการเลือกสิ่งที่ดีที่สุด หรือสิ่งที่มีความน่าเชื่อถือที่สุดจะใช้การจัดอันดับในการเลือกสิ่งต่างๆเหล่านั้น โดยสิ่งที่น่าเชื่อถือที่สุดมักจะอยู่อันดับต้นๆในการจัดอันดับ

ดาต้าล็อก$^{top-k}$ คือการค้นหาอันดับที่ดีที่สุด k อันดับโดยใช้ดาต้าล็อกในการค้นหา เพื่อหาชุดข้อมูลที่ดีที่สุด k อันดับ[7] นอกจากนั้นดาต้าล็อก$^{top-k}$ยังอนุญาตให้มีการระบุสิ่งที่คลุมเครือในกฎของดาต้าล็อกด้วย

ดาต้าล็อก$^{top-k}$จะอยู่ในรูปแบบดังต่อไปนี้

$$q(X,S) \leftarrow \exists Y body(X,Y), S = f(p_i(z_i)).$$

โดยที่

X คือตัวแปรหลักที่จะเป็นคำตอบของ query

S คือตัวแปรที่เก็บค่าคะแนนการจัดอันดับคำตอบ

Y คือตัวแปรรองที่ใช้ประกอบการค้นหาคำตอบ

body(X,Y) คือ disjunctive ของดาต้าล็อกอะตอม ใช้เป็นเงื่อนไขประกอบการค้นหาคำตอบ

z_iคือเซ็ตของค่าคงที่ หรือตัวแปรใน X หรือ Y

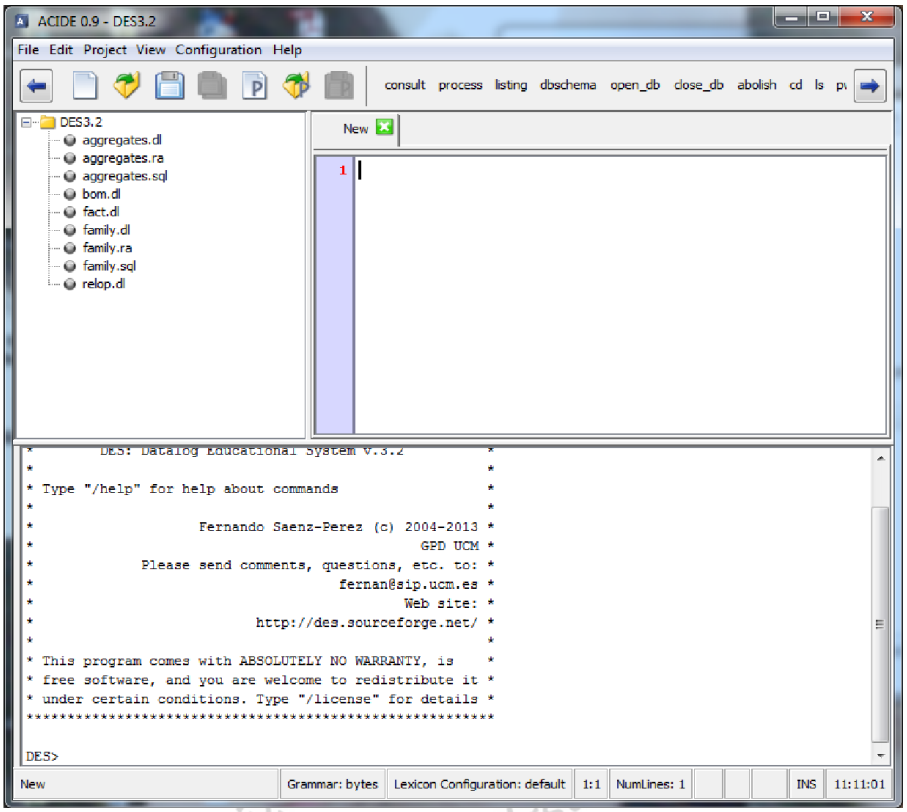
p_iคือฟังซชั่นเพรดิเคท

f คือ สกอร์ริงฟังก์ชัน หรือฟังก์ชันคำนวณคะแนนเพื่อจัดอันดับคำตอบ โดย f จะมีค่าอยู่ในช่วง [0,1]

ในดาต้าล็อก$^{top-k}$จะนำค่าคะแนนมาใช้ในการจัดอันดับ โดยงานวิจัยอื่น ๆ จะเลือกใช้วิธีการคำนวณค่าคะแนนจากการใช้ Scoring Function เพื่อคำนวณหาค่าคะแนนของข้อมูลทั้งหมดในฐานความรู้ที่มีอยู่ แต่ข้อเสียของการคำนวณด้วยการใช้ Scoring Function คือ จะไม่สามารถจัดอันดับเฉพาะกลุ่ม หรือช่วงข้อมูลที่สนใจได้

2.6 โปรแกรมดีอีเอส (Datalog Education System)

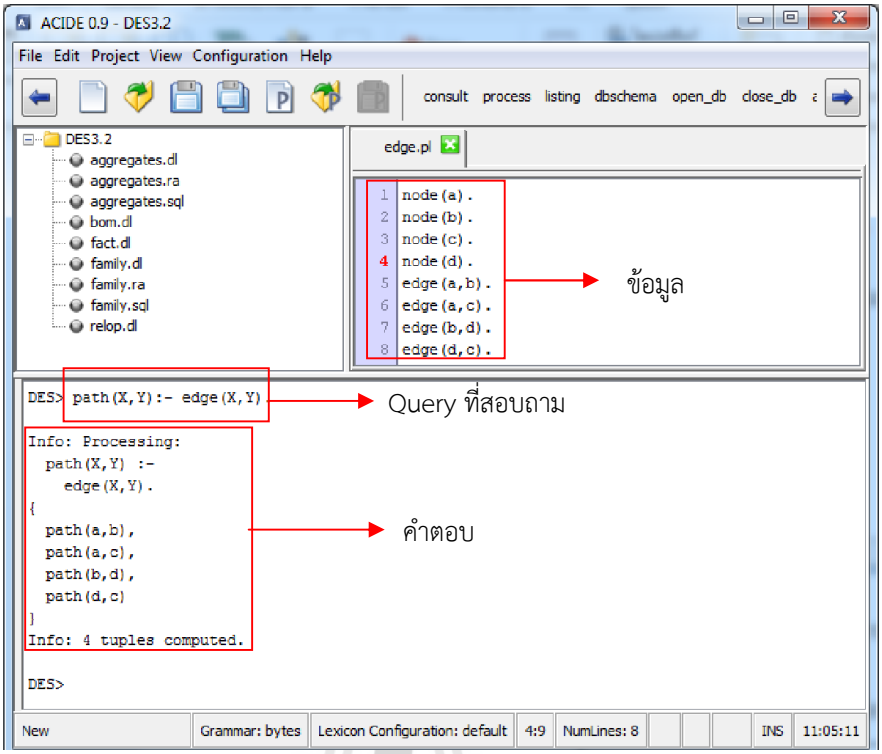
โปรแกรมดีอีเอส เป็นโปรแกรมที่สามารถใช้เขียนภาษาดาต้าล็อกได้ (สามารถดาวน์โหลดได้จาก http://des.sourceforge.net) ซึ่งโปรแกรมดีอีเอส มีทั้งแบบที่มีอินเตอร์เฟส และแบบ command prompt แสดงดังรูปที่ 2.13



รูปที่ 2.13อินเตอร์เฟสของโปรแกรมดีอีเอส

การใช้งานโปรแกรมดีอีเอสในงานวิจัยนี้ จะใช้ใน 4 รูปแบบ อธิบายได้ดังนี้

1. แบบสอบถาม (query) เป็นการระบุแบบสอบถาม คำตอบที่ได้จะตรงกับแบบสอบถามที่ระบุลงไป แสดงตัวอย่างการระบุคำถาม และคำตอบที่ได้ดังรูปที่ 2.14



รูปที่ 2.14ตัวอย่างการถามคำถามในโปรแกรมดีอีเอส

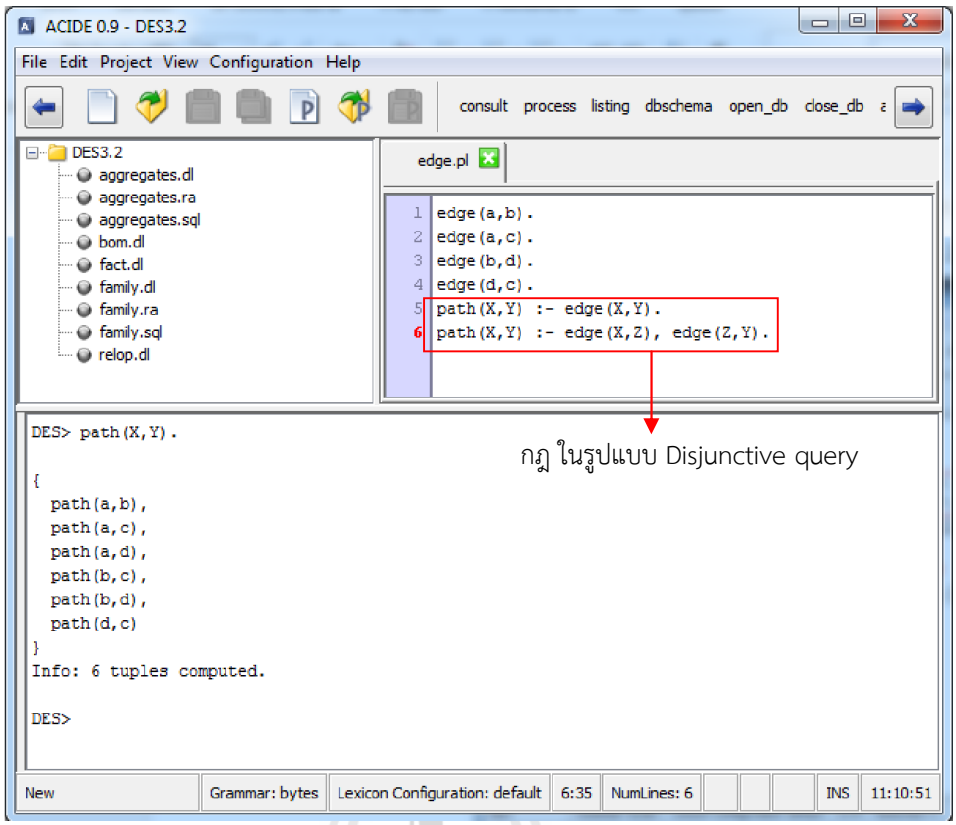
แบบสอบถามที่ระบุเข้าไปเพื่อค้นหาคำตอบ คือ path(X,Y) โดยมีเงื่อนไขดังต่อไปนี้

```
path(X,Y) :- edge(X,Y).
```

ความหมายของคำถามนี้คือ ให้แสดง path(X,Y) โดยมีเงื่อนไขว่า จะต้องมี edge ที่เชื่อมจาก X ไป Y จึงจะเรียกว่า path(X,Y)คำตอบที่เป็นไปตามเงื่อนไขที่ระบุเข้าไปคือ

```
{
  path(a,b),
  path(a,c),
  path(b,d),
  path(d,c)
}
```

2. Disjunctive query จะมีโครงสร้างในรูปแบบทางเลือก เป็นรูปแบบการถามคำถามที่มีเส้นทางค้นหาคำตอบมากกว่าหนึ่งเส้นทางแสดงตัวอย่างได้ดังรูปที่ 2.15



รูปที่ 2.15ตัวอย่างการตอบคำถามในรูปแบบ Disjunctive query ในโปรแกรมดีอีเอส

จากรูปที่ 2.15เป็นการถามคำถาม path(X,Y) นั่นคือให้แสดงเส้นทาง หรือ path ที่เชื่อมโยงจาก X ไปยัง Yกฎที่ระบุในฐานความรู้ จะทำให้ path(X,Y)สามารถหาคำตอบได้ 2 ทางเลือก โดยทางเลือกแรก ได้จากกฎpath(X,Y) :- edge(X,Y). จะได้คำตอบ ดังนี้

{

path(a,b),

path(a,c),

path(a,d),

path(b,c),

path(b,d),

path(d,c)

}

ส่วนทางเลือกที่สอง ได้จากกฎpath(X,Y) :- edge(X,Z), edge(Z,Y). จะได้คำตอบนอกเหนือจากทางเลือกแรกเพิ่มเติมขึ้นมา คือ

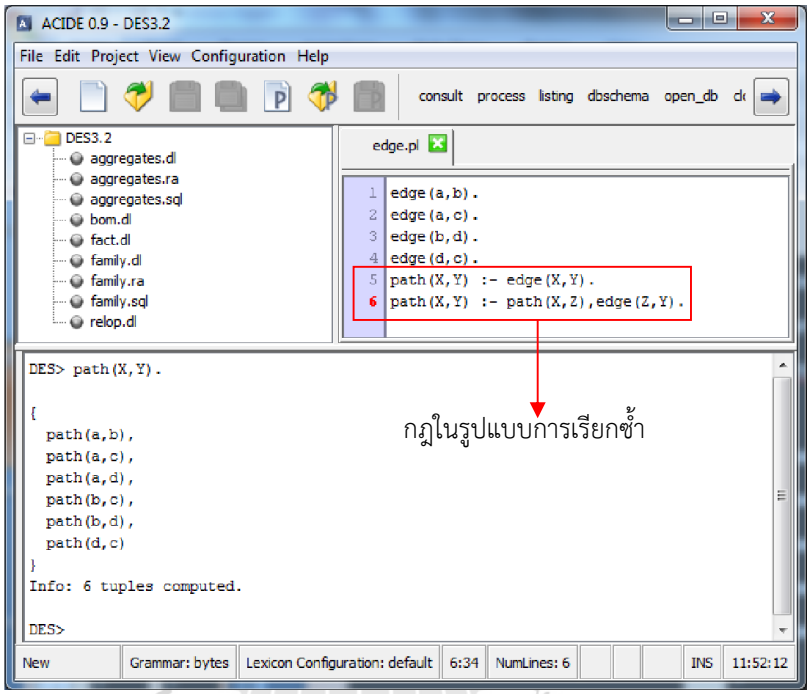
{

path(a,d),

path(b,c)

}

3. การเรียกซ้ำ (Recursive) คือการเรียกกฎซ้ำๆ เพื่อหาคำตอบในรูปแบบใหม่ ดังรูปที่ 2.16



รูปที่ 2.16ตัวอย่างการตอบคำถามในรูปแบบของการเรียกซ้ำในโปรแกรมดีอีเอส

จากรูปที่ 2.16 จะเห็นได้ว่า path(X,Y) มีการเรียกซ้ำเพรดิเคท path จากกฎpath(X,Y) :- path(X,Z), edge(Z,Y). เพื่อหาคำตอบที่เพิ่มเติมขึ้นมา

4. การจัดอันดับ top-K คือการค้นหาข้อมูลและเลือกข้อมูลมา K อันดับแรก โดยมีรูปแบบดังต่อไปนี้

top(K, order_by(p(X,Y), [Y], [d])).

โดยที่

K คือจำนวนที่ต้องการคำตอบ

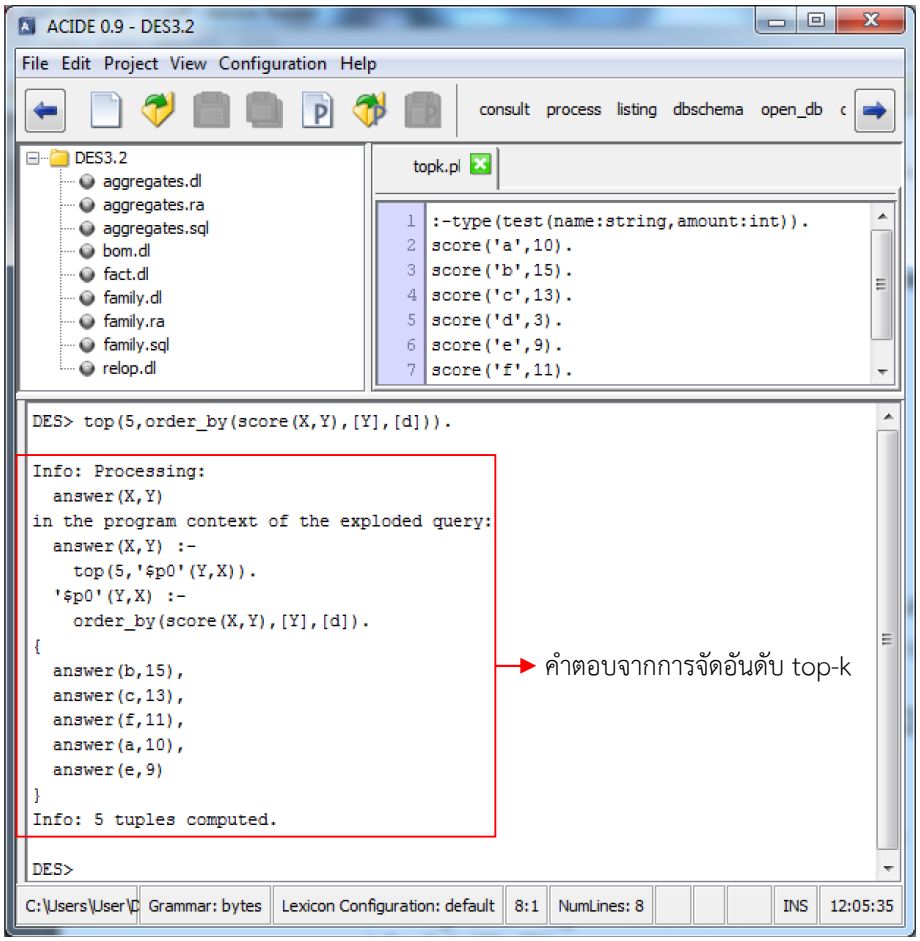
p คือเพรดิเคทที่ต้องการจัดอันดับ

(X, Y)คือตัวแปร หรือค่าคงที่ที่อยู่ภายในเพรดิเคท

[Y] คือตัวแปรที่ใช้ในการจัดอันดับ

[d] ใช้ระบุว่าจะให้จัดอันดับอยู่ในรูปแบบใด สามารถใช้ [d] หรือ [a] โดยที่ [d] คือให้เรียงลำดับจากมากไปน้อย และ[a] คือให้เรียงลำดับจากน้อยไปมาก

ตัวอย่างการตอบคำถาม เค อันดับแรกในโปรแกรม ดีอีเอส โดยกำหนดให้ k=5 แสดงได้ดังรูปที่ 2.17



รูปที่ 2.17ตัวอย่างการตอบคำถามเคอันดับแรก ในโปรแกรมดีอีเอส

2.7 งานวิจัยที่เกี่ยวข้อง

งานวิจัยที่เกี่ยวข้องกับการจัดอันดับ top-k นั้นมีมากมายซึ่งจะเห็นได้จากงานวิจัยที่มีการนำเสนอแนวคิดการจัดอันดับ top-k และการปรับปรุงอัลกอริทึมการจัดอันดับ top-k ดังต่อไปนี้

ในปี ค.ศ. 2006Umberto Straccia[7,8] ได้เสนอวิธีการจัดอันดับ top-k โดยใช้อัลกอริทึม 3 อัลกอริทึมคือ Disjunctive Threshold การใช้ top-k retrieve engines และ การใช้ Naive solution จัด

อันดับข้อมูลด้วยการใช้ scoring function ผลการทดลอง ปรากฏว่าหากใช้วิธี Naive solution นั้น จำเป็นจะต้องคำนวณในแต่ละ tuples ซึ่งอาจจะมีถึงหลักล้าน tuples ส่งผลให้ไม่สะดวกในการใช้การจัดอันดับ top-k ในชุดข้อมูลที่มีขนาดใหญ่ ส่วนวิธีการ top-k retrieve engines จะเหมาะสำหรับชุดข้อมูลที่มีขนาดเล็ก และวิธีการจัดอันดับ top-k ด้วยอัลกอริทึม Disjunctive Threshold ได้ผลลัพธ์เป็นที่น่าพอใจที่สุด

ในปีเดียวกัน Umberto Straccia[9] ได้เสนอวิธีการจัดอันดับ top-k ด้วยอัลกอริทึมแบบใหม่ด้วยการแยกส่วนการทำงานของอัลกอริทึมออกเป็น 2 ส่วน โดยในส่วนแรกจะทำหน้าที่เลือก predicate ที่ใช้อธิบายการทำงานของกฎ(rule) แล้วส่งต่อให้ส่วนที่ 2 ซึ่งทำหน้าที่คำนวณค่า score และจัดอันดับแต่ละ tuples

ในปี ค.ศ. 2012 Umberto Straccia และ Nicolas Madrid [10]ได้นำเสนอวิธีการคำนวณและจัดอันดับ top-k ด้วยอัลกอริทึมแบบใหม่ โดยการใช้ membership functionผลการทดลองปรากฏว่าการใช้ membership function เข้ามาช่วยจัดอันดับ สามารถทำให้การจัดอันดับมีประสิทธิภาพมากยิ่งขึ้น

Agnes Achs [2] ได้นำเสนอรูปแบบของการจัดการข้อมูลที่มีความไม่แน่นอนโดยการกำหนด fuzzy knowledge-base ให้มีค่าสูงกว่า background knowledge และนำเสนอโครงสร้างของการคำนวณที่เป็นไปได้เพื่อนำไปปรับปรุง หรือพัฒนาอัลกอริทึมอื่นๆ

จากการศึกษางานวิจัยที่เกี่ยวข้อง พบว่าการจัดอันดับ top-k ในแต่ละงานวิจัยนั้นมีการใช้ทั้ง scoring function และ membership function ในการจัดอันดับ top-k ของข้อมูล โดย scoring function เหมาะสำหรับการจัดอันดับข้อมูลที่มีความคลุมเครือโดยพิจารณาข้อมูลทั้งหมดที่มีอยู่ ส่วน membership function เหมาะสำหรับการจัดอันดับข้อมูลที่มีความคลุมเครือ โดยพิจารณาเฉพาะช่วงข้อมูลช่วงหนึ่งๆ ในงานวิจัยนี้ได้พัฒนาโปรแกรมการจัดอันดับ top-k ของข้อมูลที่สามารถใช้กับการพิจารณาข้อมูลในช่วงหนึ่งๆด้วยการใช้เทคนิคการคำนวณค่าฟังก์ชันสมาชิกเพื่อการจัดอันดับ top-k โดยแสดงให้เห็นข้อแตกต่างของการใช้ฟังก์ชันสมาชิกในแต่ละฟังก์ชันทั้ง 4 ฟังก์ชันและได้ทำการเปรียบเทียบกับงานวิจัยอื่น ๆ สรุปได้ดังตารางที่ 2.5

ตารางที่ 2.5สรุปเปรียบเทียบงานวิจัยที่เกี่ยวข้องกับการจัดอันดับ top-k

กระบวนการทำงาน	งานวิจัยที่เกี่ยวข้อง				
	ก	ข	ค	ง	จ
วิธีการจัดอันดับ top-k					
อัลกอริทึม Threshold	✓	✓	✓		✓
Naive solution	✓				
Top-k retrieve engines	✓				✓
Fuzzy set			✓	✓	✓
การพัฒนา					
Datalog	✓	✓	✓	✓	✓
Datalog$^{top\text{-}k}$	✓	✓	✓		✓
Scoring function	✓	✓		✓	✓
Membership function			✓		✓
ขอบเขตของการวิจัย					
วิจัยเพื่อทดสอบประสิทธิภาพ	✓	✓	✓	✓	✓
วิจัยเพื่อเสนอแนวคิดใหม่	✓	✓	✓	✓	✓
มีการประยุกต์ใช้กับข้อมูลจริง	✓		✓		✓

หมายเหตุ งานวิจัยที่เกี่ยวข้อง ประกอบไปด้วย

ก แทนงานวิจัยของ Umberto Straccia. (2006ab) [7,8]

ข แทนงานวิจัยของ Umberto Straccia. (2007c) [9]

ค แทนงานวิจัยของ Umberto Straccia และ Nicolas Madrid (2012) [10]

ง แทนงานวิจัยของ Agnes Achs (2007) [2]

จ แทนงานวิจัยของวิทยานิพนธ์ฉบับนี้

บทที่ 3
วิธีดำเนินการวิจัย

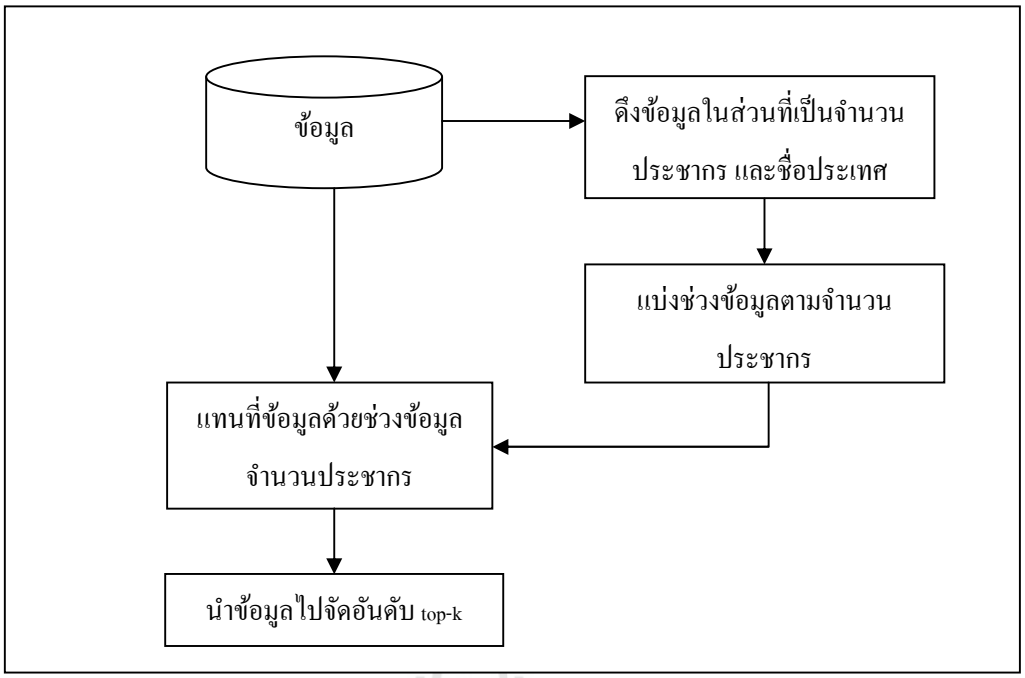
งานวิจัยนี้มีวัตถุประสงค์เพื่อศึกษาการจัดอันดับ top-k ของคำตอบในกรณีที่คำถามมีความคลุมเครือโดยประยุกต์ใช้จริงกับการจัดอันดับคำตอบที่เกี่ยวข้องกับจำนวนประชากรโลกซึ่งเป็นข้อมูลที่สืบค้นจากธนาคารโลก และนำมาบันทึกในรูปแบบของฐานข้อมูลนิรนัย ในบทนี้จะกล่าวถึง วิธีการวิจัย เครื่องมือที่ใช้ในการวิจัย และกระบวนการต่างๆ ของการวิจัย โดยมีรายละเอียดดังนี้

3.1 กรอบแนวคิดของการวิจัย

แนวคิดหลักของงานวิจัยนี้คือ การจัดอันดับ top-k ด้วยฟังก์ชันสมาชิกสำหรับนำไปใช้ในฐานข้อมูลนิรนัยโดยสามารถแบ่งกรอบแนวคิดของงานวิจัยนี้ออกเป็น 2 ขั้นตอน ได้แก่ ขั้นตอนที่ 1 การจัดกลุ่ม(Clustering) และขั้นตอนที่ 2 การจัดอันดับ top-k

3.1.1 ขั้นตอนที่ 1: วิธีการจัดกลุ่ม (Clustering)

ขั้นตอนแรกของการดำเนินการจะเป็นการจัดกลุ่มข้อมูล โดยจะนำข้อมูลจำนวนประชากรโลกที่รวมทุกประเทศและไม่ได้แบ่งระดับจำนวนประชากร มาแบ่งกลุ่มของข้อมูลออกเป็น 3 กลุ่ม โดยกลุ่มแรกคือประเทศที่มีจำนวนประชากรอยู่ในระดับน้อย กลุ่มที่สองคือประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง และกลุ่มสุดท้ายคือประเทศที่มีจำนวนประชากรอยู่ในระดับมากโดยจะมีกรอบแนวคิด ดังรูปที่ 3.1



รูปที่ 3.1กรอบแนวคิดวิธีการจัดกลุ่ม (Clustering)

จากรูปที่ 3.1 กรอบแนวคิดการจัดกลุ่ม จะประกอบไปด้วย 4 ขั้นตอนย่อย ดังนี้

1) ดึงข้อมูลในส่วนที่เป็นจำนวนประชากรและชื่อประเทศออกมา เนื่องจากในการวิจัยนี้มุ่งไปที่การจัดอันดับ top-k ของจำนวนประชากรโลก จึงใช้ข้อมูลเพียงจำนวนประชากรโลกและชื่อประเทศ

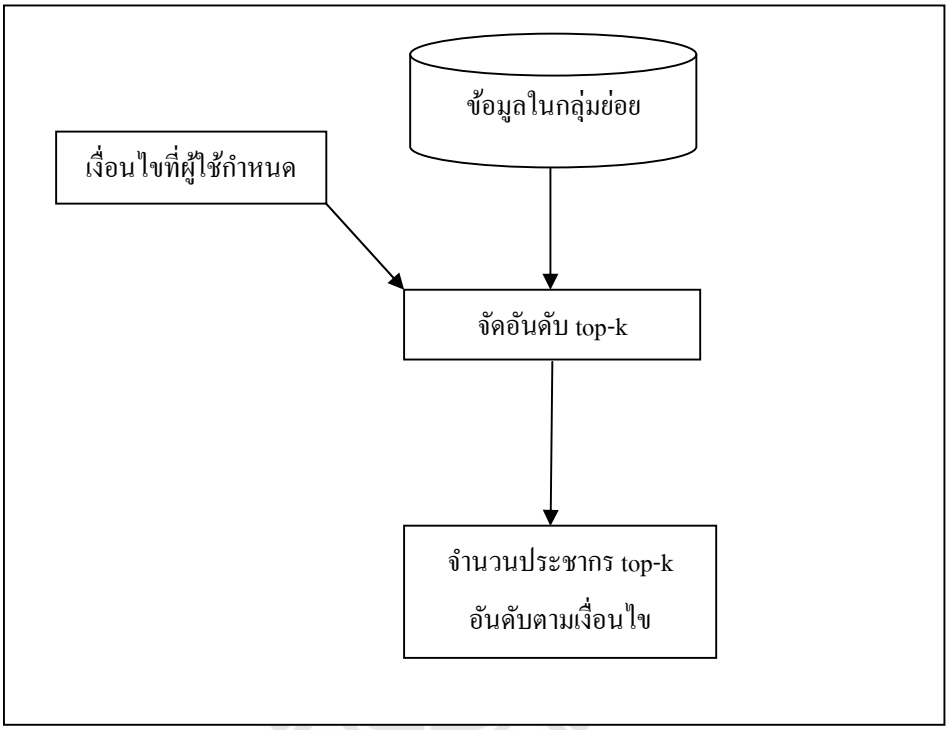
2) แบ่งช่วงข้อมูลตามจำนวนประชากร โดยจะแบ่งออกเป็น 3 กลุ่ม คือกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับน้อย กลุ่มถัดมาคือกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง และกลุ่มสุดท้ายคือกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับมาก

3) แทนที่ข้อมูลด้วยช่วงข้อมูลจำนวนประชากร คือการนำข้อมูลที่ผ่านการจัดกลุ่มในขั้นตอนก่อนหน้ามาแทนที่ชุดข้อมูลเดิม

4) จัดอันดับ top-k คือการจัดอันดับตามเงื่อนไขที่ผู้ใช้กำหนด จะได้ข้อมูลชุดใหม่ที่มีการจัดอันดับ top-k ตามเงื่อนไข

3.1.2 ขั้นตอนที่ 2: การจัดอันดับ top-k ของข้อมูล

การดำเนินการในขั้นตอนที่ 2 จะเป็นการจัดอันดับ top-k ของข้อมูลจำนวนประชากรโลกที่ผ่านการจัดกลุ่ม โดยจะนำชุดข้อมูลในแต่ละกลุ่มมาจัดอันดับ top-k ตามเงื่อนไขที่ผู้ใช้กำหนดเข้าไป โดยมีกรอบแนวคิดดังรูปที่ 3.2



รูปที่ 3.2กรอบแนวคิดการจัดอันดับ top-k ของข้อมูลในกลุ่มย่อย

จากกรอบแนวคิดการจัดอันดับ top-k ของข้อมูล จะประกอบไปด้วย 3 ขั้นตอนย่อย ดังนี้

1) ข้อมูลในกลุ่มย่อย คือข้อมูลที่ประกอบไปด้วย 3 กลุ่มจากการแบ่งกลุ่มของข้อมูล ได้แก่ กลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับน้อย กลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง และกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับมาก

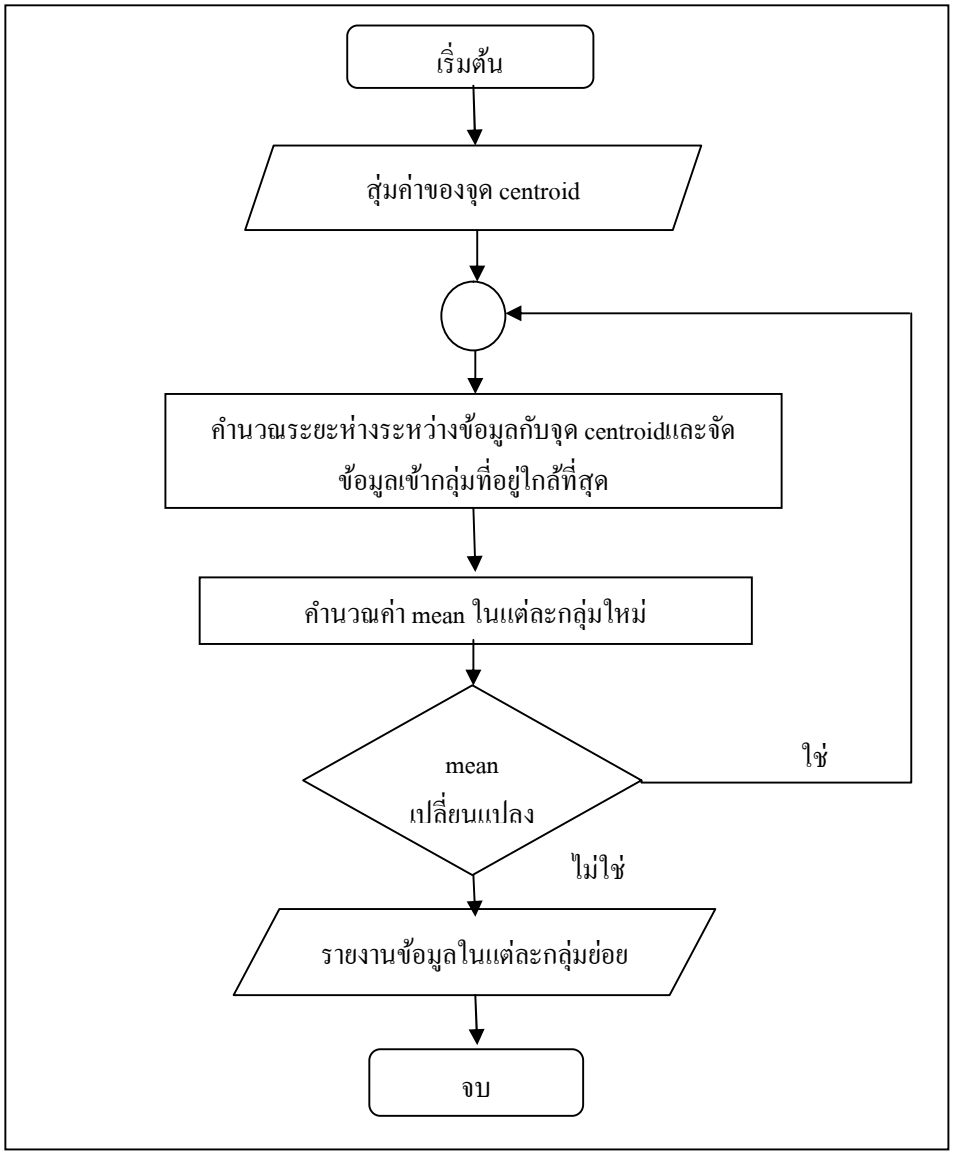
2) จัดอันดับ top-k ตามเงื่อนไขที่ผู้ใช้ระบุเข้ามา โดยการพิจารณาเงื่อนไขว่าผู้ใช้ต้องการจัดอันดับ top-k ของข้อมูลในกลุ่มใด

3) เมื่อผ่านการจัดอันดับ top-k ในขั้นตอนก่อนหน้าแล้ว จะได้อันดับประเทศที่ตรงตามเงื่อนไขที่ผู้ใช้กำหนด เพื่อนำข้อมูลชุดนี้ไปใช้งานต่อไป

3.2 การออกแบบอัลกอริทึม

3.2.1 ออกแบบอัลกอริทึมการจัดกลุ่มข้อมูลจำนวนประชากรโลกด้วย k-means

อัลกอริทึม k-meansจะใช้หลักการตัดแบ่ง (Partition) แบ่งวัตถุจำนวน n ตัวในฐานข้อมูลออกเป็นจำนวน kกลุ่ม ด้วยการแทนแต่ละกลุ่มด้วยค่าเฉลี่ยของกลุ่ม ซึ่งค่าเฉลี่ยนี้จะเป็นจุดศูนย์กลางของกลุ่มในการวัดระยะห่างของข้อมูลทั้งหมดในกลุ่มเดียวกัน ขั้นตอนการแบ่งกลุ่มด้วยอัลกอริทึม k-means แสดงได้ดังรูปที่ 3.3



รูปที่ 3.3ผังงานแสดงขั้นตอนการจัดกลุ่มด้วยอัลกอริทึม k-means

จากรูปที่ 3.3คือผังงานในการจัดกลุ่มด้วยอัลกอริทึม k-means โดยมีขั้นตอนการทำงานดังนี้

1) เมื่อมีการเริ่มทำงาน จะทำการสุ่มค่าcentroid เริ่มต้นจำนวน k ค่า

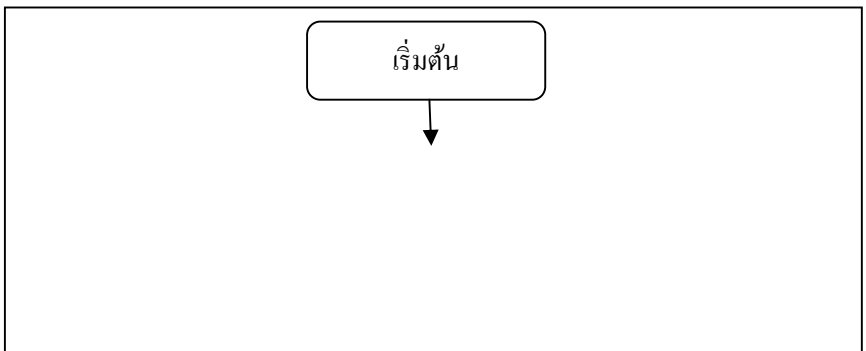
2) หลังจากสุ่มค่า centroid เริ่มต้นได้แล้วจะทำการหาระยะห่างระหว่างข้อมูลกับจุด centroid หากข้อมูลใดมีค่าใกล้ค่าจุด centroid กลุ่มใดมากที่สุด ข้อมูลชุดนั้นจะถูกจัดอยู่ในกลุ่มนั้น

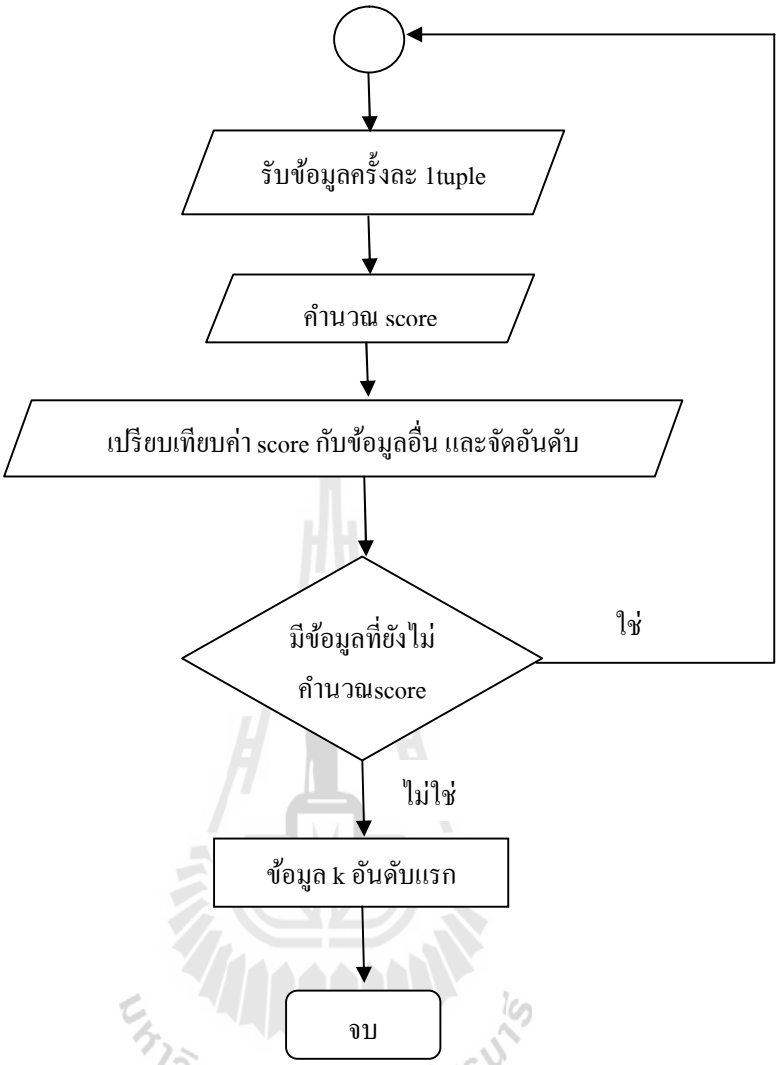
3) หลังจากหาระยะห่างของข้อมูลครบทุกตัวในรอบแรกแล้ว จะทำการคำนวณค่า mean ในแต่ละกลุ่มสำหรับกำหนดค่าcentroid ใหม่ และพิจารณาว่าค่าmean เปลี่ยนแปลงหรือไม่ หากค่า meanมีการเปลี่ยนแปลง จะกลับไปเริ่มทำข้อ 2 และข้อ 3 อีกครั้งจนกระทั่งค่า meanของแต่ละกลุ่มไม่มีการเปลี่ยนแปลง

4) เมื่อค่า meanทั้งหมดไม่มีการเปลี่ยนแปลง จะได้ข้อมูลชุดใหม่ที่ได้รับการแบ่งกลุ่มแล้ว

3.2.2 ออกแบบอัลกอริทึมการจัดอันดับ top-k

อัลกอริทึมการจัดอันดับ top-k จะใช้หลักการของการทำซ้ำสำหรับคำนวณค่า score ของแต่ละ tuples เพื่อนำค่า score ที่ได้มาจัดอันดับ จำนวน k อันดับแรก โดยรับข้อมูลเข้ามาครั้งละ 1 tuples และหลังจากคำนวณค่า score ใน tuple นั้นเสร็จสิ้นแล้ว จะทำการตรวจสอบว่ามีข้อมูลที่ยังไม่ถูกนำมาคำนวณค่า score หรือไม่ โดยขั้นตอนการจัดอันดับ top-k แสดงได้ดังรูปที่ 3.4





รูปที่ 3.4ผังงานแสดงขั้นตอนการจัดอันดับ top-k

จากรูปที่ 3.4แสดงผังงานขั้นตอนการจัดอันดับ top-k โดยมีขั้นตอนดังนี้

1) เมื่อมีการเริ่มทำงาน จะรับข้อมูลเข้ามาครั้งละ 1 tuple

2) นำข้อมูลที่รับเข้ามา มาคำนวณค่า score

3) หลังจากคำนวณค่า score ใน tuple นั้นๆ จะนำค่า score ที่ได้ไปเปรียบเทียบกับค่า score ของข้อมูลก่อนหน้า เพื่อจัดอันดับ

4) ตรวจสอบว่ามีข้อมูลที่ยังไม่ถูกนำมาคำนวณ score หรือไม่ หากมีข้อมูลที่ยังไม่ได้นำมาคำนวณค่า score จะวนกลับเข้าไปทำซ้ำข้อที่ 1, 2, 3 และข้อที่ 4 จนกระทั่งไม่มีข้อมูลที่ยังไม่ได้นำมาคำนวณค่า score

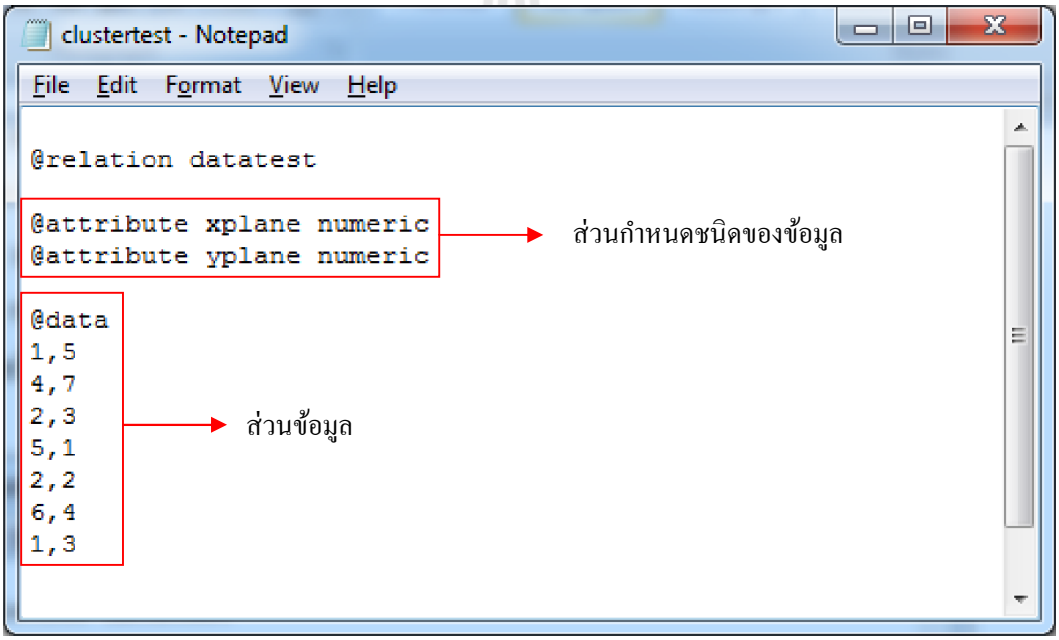
5) เมื่อคำนวณค่า score ครบทุก tupleจะสามารถคัดเลือกให้ได้ข้อมูล k อันดับแรกตามเงื่อนไขที่ผู้ใช้กำหนด

3.3 การใช้งานโปรแกรม

เนื้อหาในส่วนนี้ จะอธิบายการใช้งานของโปรแกรมการจัดอันดับ top-k โดยจะแบ่งการทำงานของโปรแกรมออกเป็นขั้นตอนดังนี้

3.3.1 การเตรียมข้อมูล

การใช้งานโปรแกรมการจัดกลุ่มข้อมูล ข้อมูลที่นำมาใช้ต้องมีการกำหนดชนิดข้อมูลในแต่ละคอลัมน์ ในงานวิจัยนี้มีการกำหนดให้ข้อมูลอยู่ในรูปแบบ .arffดังรูปที่ 3.5



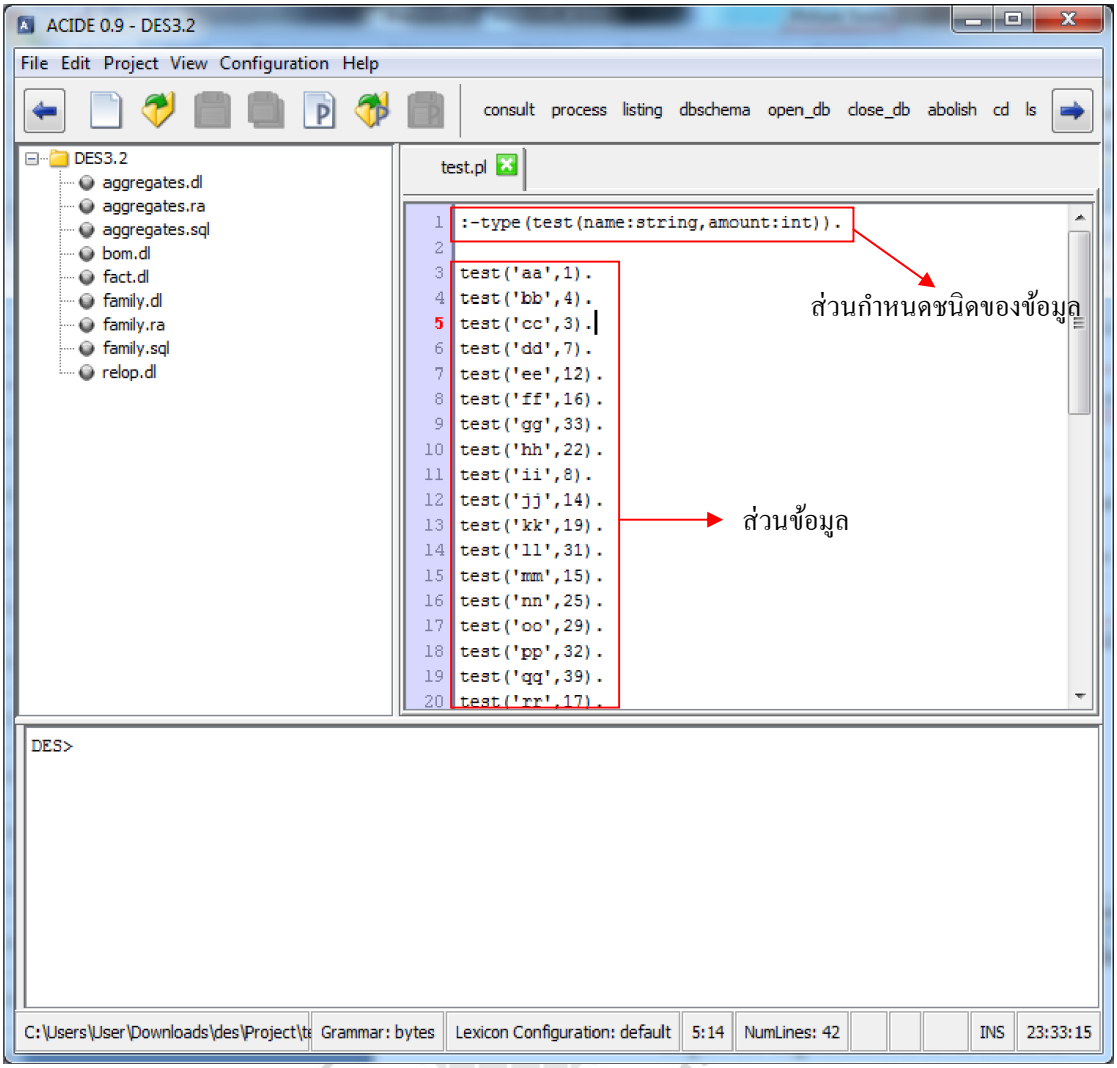
รูปที่ 3.5แสดงส่วนต่างๆของข้อมูลสำหรับการจัดกลุ่ม

จากรูปที่ 3.5 ข้อมูลจะประกอบไปด้วย 2 ส่วน คือ

1) ส่วนกำหนดชนิดข้อมูล เป็นส่วนคำอธิบายชนิดของข้อมูลในแต่ละคอลัมน์

2) ส่วนข้อมูล เป็นรายละเอียดข้อมูลในแต่ละเรคคอร์ด

การใช้งานโปรแกรมการจัดอันดับ top-k ข้อมูลที่นำมาใช้ต้องมีการกำหนดชนิดข้อมูลในแต่ละคอลัมน์ ดังนั้นการวิจัยนี้จึงกำหนดให้ข้อมูลต้องอยู่ในรูปแบบ .plดังรูปที่ 3.6



รูปที่ 3.6แสดงส่วนต่างๆของข้อมูล

จากรูปที่ 3.6 ข้อมูลจะประกอบไปด้วย 2 ส่วน คือ

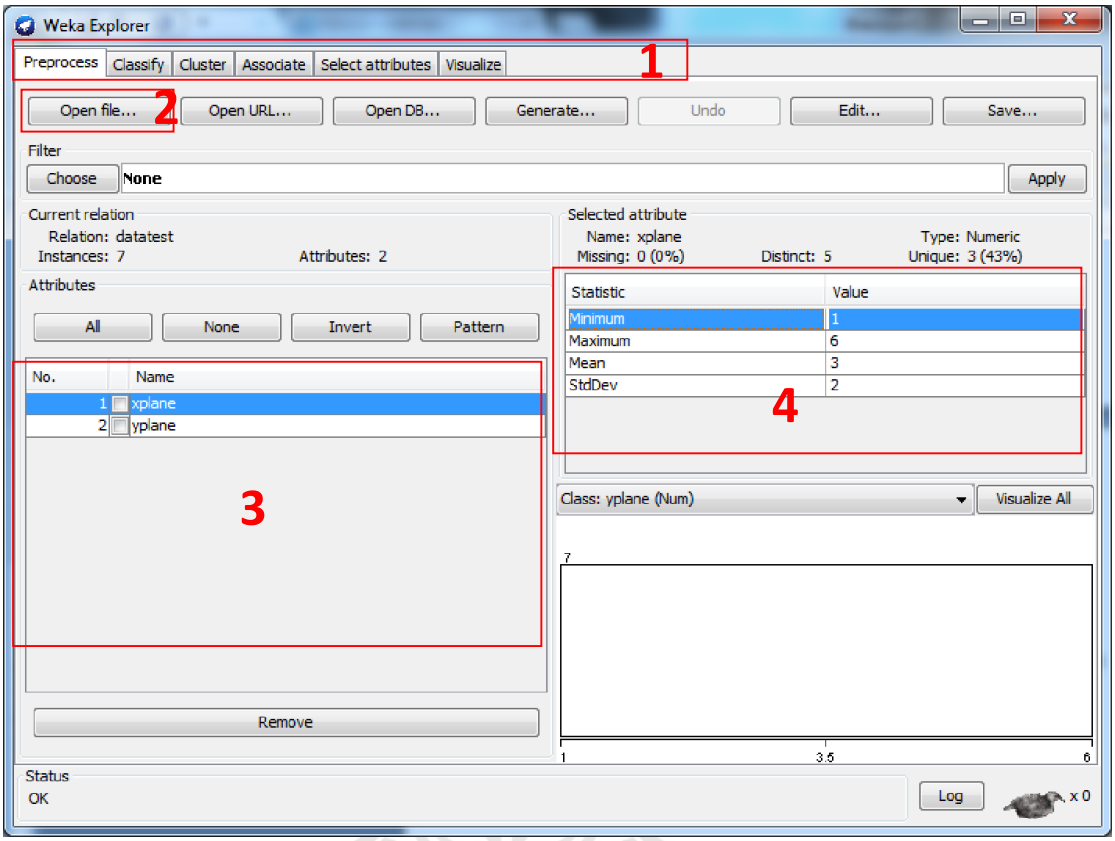
1) ส่วนกำหนดชนิดข้อมูล เป็นส่วนคำอธิบายชนิดของข้อมูลในแต่ละคอลัมน์

2) ส่วนข้อมูล เป็นรายละเอียดข้อมูลในแต่ละเรคคอร์ด

3.3.2 การจัดกลุ่มข้อมูล

การเรียกใช้โปรแกรมในส่วนของการจัดกลุ่ม จะใช้โปรแกรม weka 3.6 (http://www.cs.waikato.ac.nz/ml/weka/) ในการจัดกลุ่ม โดยมีขั้นตอนในการจัดกลุ่มดังต่อไปนี้

เมื่อเปิดโปรแกรม weka 3.6 จะปรากฏหน้าต่าง ดังรูปที่ 3.7



รูปที่ 3.7หน้าต่างโปรแกรมและส่วนประกอบต่างๆใน weka 3.6

จากรูปที่ 3.7 สามารถอธิบายส่วนประกอบต่างๆ ดังนี้

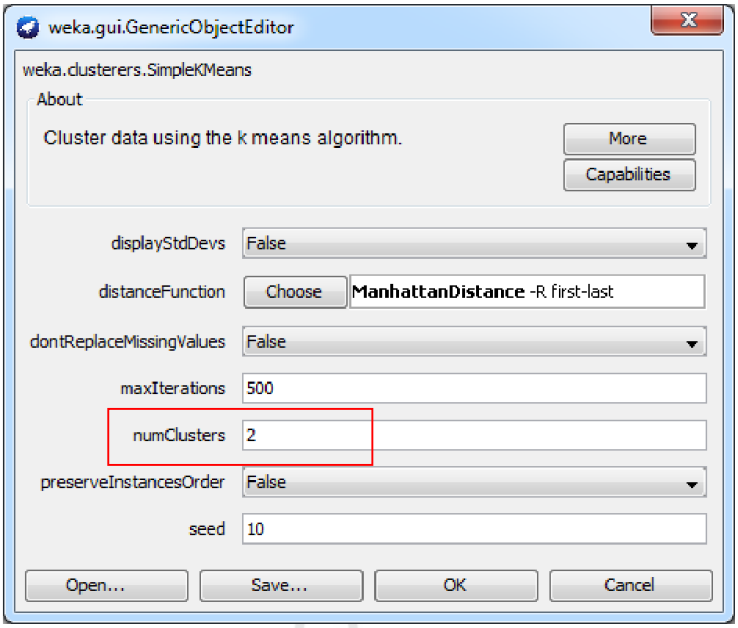
หมายเลข 1 แถบเมนูสำหรับเลือกการทำงาน

หมายเลข 2 ปุ่มคำสั่งโหลดไฟล์ที่จะนำมาจัดกลุ่ม

หมายเลข 3 แถบแสดงโครงสร้างของไฟล์ที่จะนำมาจัดกลุ่ม จากรูปที่ 3.7 ไฟล์ที่จะนำมาจัดกลุ่มก่อนที่จะคัดเลือกอันดับนี้ประกอบไปด้วย 2 attribute ได้แก่ xplaneและ yplane

หมายเลข 4 แถบแสดงข้อมูลสถิติ

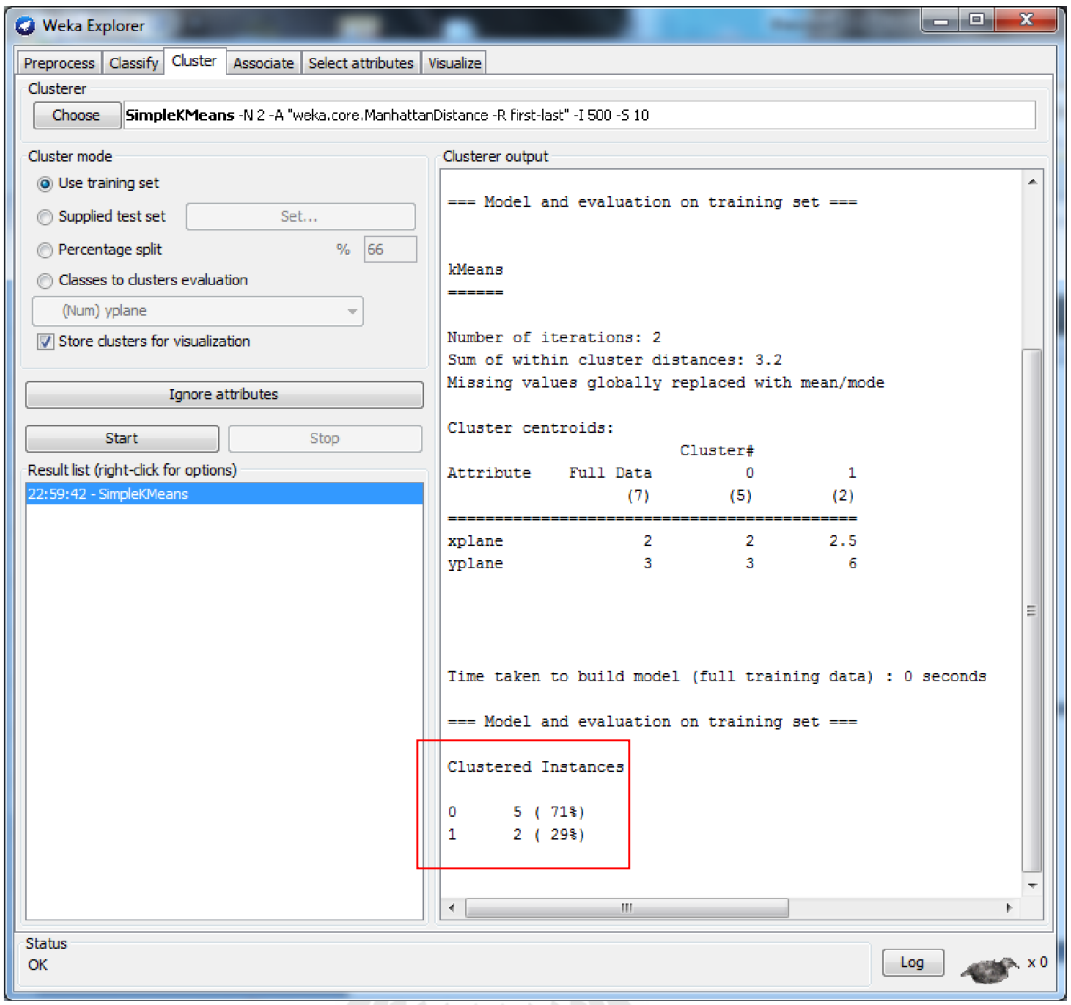
การจัดกลุ่มข้อมูล จะเลือกใช้เมนู Cluster ->Clusterer -> Choose ->SimpleKMeansเลือกจำนวนกลุ่มที่ต้องการแบ่งกลุ่มออกเป็น 2 กลุ่ม ดังรูปที่ 3.8



รูปที่ 3.8ส่วนประกอบต่างๆของการจัดกลุ่มใน weka 3.6

หลังจากกำหนดจำนวนกลุ่มที่ต้องการเรียบร้อยแล้ว จะได้โมเดลดังรูปที่ 3.9

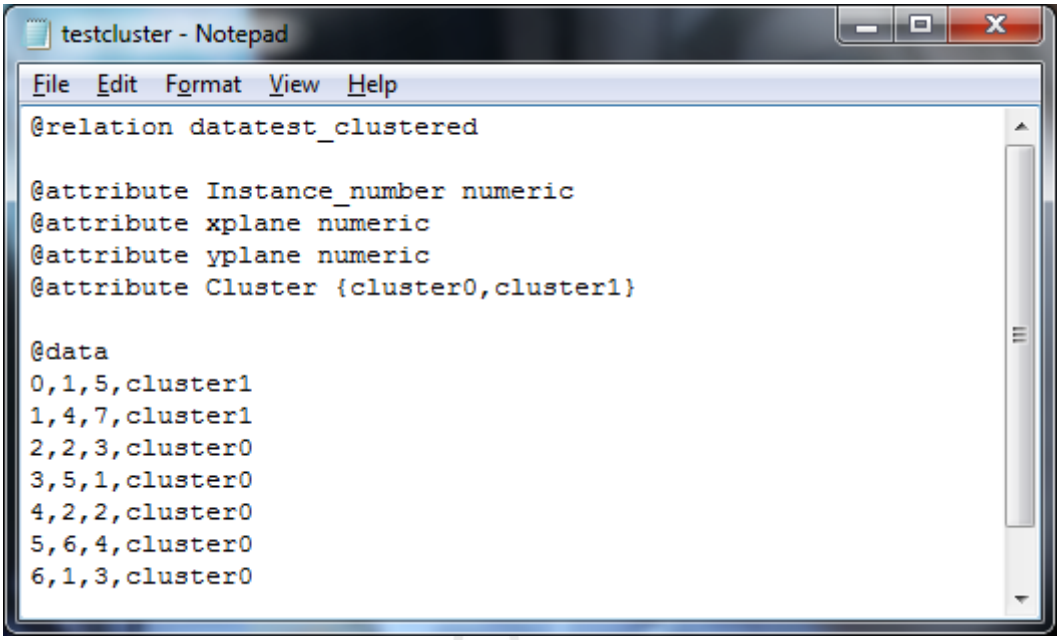
มหาวิทยาลัยเทคโนโลยีสุรนารี



รูปที่ 3.9 โมเดลของการจัดกลุ่ม

จากรูปที่ 3.9 ข้อมูลที่ถูกจัดให้อยู่ในกลุ่มที่ 1 มีทั้งหมด 5tuple คิดเป็น 71% ของข้อมูลทั้งหมด และส่วนที่เหลืออีก 2 tuple คิดเป็น 29% ถูกจัดให้อยู่ในกลุ่มที่ 2 เมื่อจุด centroid ของกลุ่มที่ 1 และกลุ่มที่ 2 คือ (2, 3) และ (2.5, 6) ตามลำดับ

ข้อมูลที่ถูกจัดอยู่ในกลุ่มที่ 1 ได้แก่ (2, 3), (5, 1), (2, 2), (6, 4) และ (1, 3) ส่วนข้อมูลที่ถูกจัดให้อยู่ในกลุ่มที่ 2 ได้แก่ (1, 5) และ (4, 7) ดังรูปที่ 3.10



รูปที่ 3.10แสดงรายละเอียดของข้อมูลหลังผ่านการจัดกลุ่ม

3.3.3 การจัดอันดับ top-k

การเรียกใช้โปรแกรมในส่วนของการจัดอันดับ top-k นั้นจะมีรูปแบบและตัวอย่างการเรียกใช้โปรแกรมการจัดอันดับ top-k ดังรูปที่ 3.11

```
top(K,order  by(predicate(Name,Amount,Score),[Score],[d])).
```

รูปที่ 3.11รูปแบบการเรียกใช้โปรแกรมการจัดอันดับ top-k

จากรูปที่ 3.11 คือรูปแบบการเรียกใช้โปรแกรมในส่วนของการจัดอันดับ top-k โดยมีตัวแปรดังนี้

K คือจำนวนของอันดับที่ต้องการ

predicateคือชื่อตารางที่เก็บข้อมูล

Name คือชื่อattribute

Amount คือค่าที่ใช้ในการคำนวณค่าความเป็นสมาชิก

Score คือค่าความเป็นสมาชิก

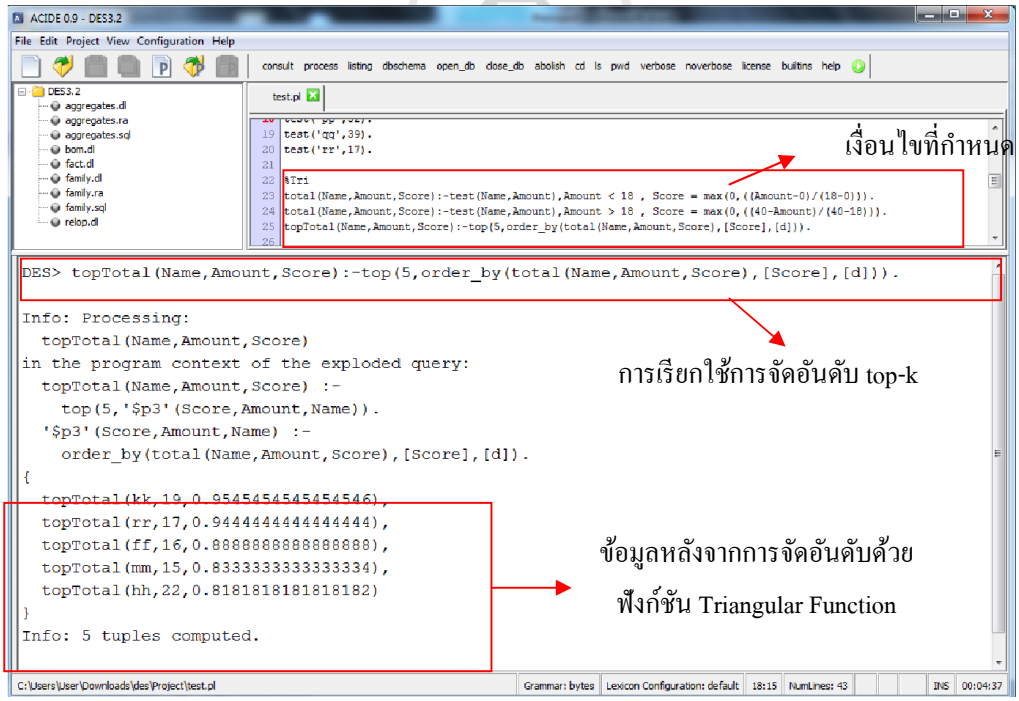
การจัดอันดับด้วยการใช้ฟังก์ชันสมาชิก Triangular Function โดยกำหนดเงื่อนไขว่าต้องการข้อมูลที่มีค่าใกล้เคียงกับ 18 มากที่สุด 5 อันดับแรก สามารถเขียนเงื่อนไขได้ดังรูปที่ 3.12

```
% Triangular Function
total(Name,Amount,Score):-test(Name,Amount),Amount < 18 ,
                                Score = max(0,((Amount-0)/(18-0))).
total(Name,Amount,Score):-test(Name,Amount),Amount > 18 ,
                                Score = max(0,((40-Amount)/(40-18))).
topTotal(Name,Amount,Score):-top(5,order_by(total(Name,Amount,Score),[Score],[d])).
```

รูปที่ 3.12แสดงเงื่อนไขในการจัดอันดับด้วยฟังก์ชันสมาชิก Triangular Function

จากรูปที่ 3.12 สามารถอธิบายเงื่อนไขการจัดอันดับได้ดังนี้

- ถ้า Amount มีค่าน้อยกว่า 18 การคำนวณค่า Score = (Amount-0)/(18-0)
- ถ้า Amount มีค่ามากกว่า 18 การคำนวณค่า Score = (40-Amount)/(40-18)
- เมื่อ Amount คือค่าที่จะนำมาคำนวณหาค่าความเป็นสมาชิกโดยที่ 40 และ 0 คือค่าสูงสุดและต่ำสุดในช่วงของข้อมูลที่สนใจตามลำดับ และ18 คือค่าที่สนใจจะได้ผลลัพธ์การจัดอันดับ top-k ด้วย Triangular function ดังรูปที่ 3.13



รูปที่ 3.13 แสดงผลลัพธ์จากการจัดอันดับ top-kด้วย Triangular Function

จากรูปที่ 3.13เมื่อจัดอันดับ top-k ด้วย Triangular Functionจะได้ข้อมูลที่มีค่าใกล้เคียงกับ 18 มากที่สุด 5 อันดับ ได้แก่ kk, rr, ff, mm และhhโดยมีค่าความเป็นสมาชิก 0.955, 0.944, 0.889, 0.833 และ 0.818 ตามลำดับ

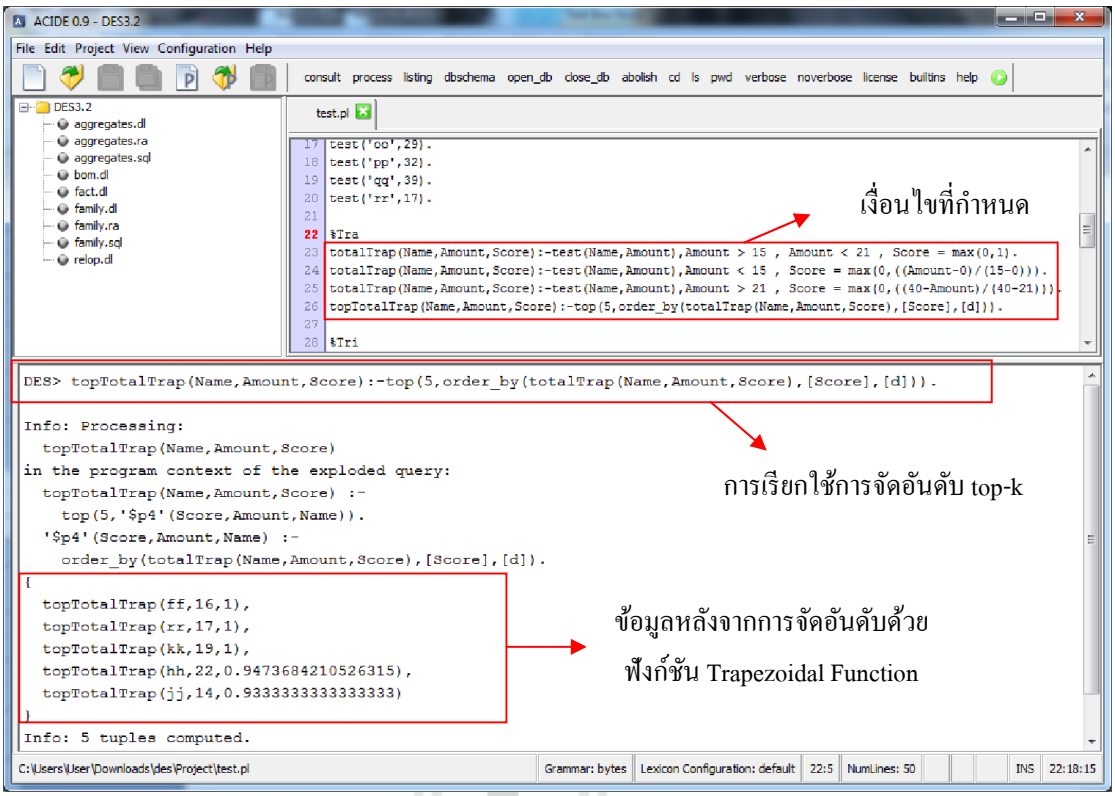
การจัดอันดับด้วยการใช้ฟังก์ชันสมาชิก Trapezoidal Function โดยกำหนดเงื่อนไขว่าต้องการข้อมูลที่มีค่าใกล้เคียงช่วง15 ถึง 21มากที่สุด 5 อันดับแรก สามารถเขียนเงื่อนไขได้ดังรูปที่ 3.14

```
% Trapezoidal Function
totalTrap(Name,Amount,Score):-test(Name,Amount),Amount > 15 , Amount < 21 , Score = max(0,1).
totalTrap(Name,Amount,Score):-test(Name,Amount),Amount < 15 ,
                                Score = max(0,((Amount-0)/(15-0))).
totalTrap(Name,Amount,Score):-test(Name,Amount),Amount > 21 ,
                                Score = max(0,((40-Amount)/(40-21))).
topTotalTrap(Name,Amount,Score):-  top(5,order_by(totalTrap(Name,Amount,Score),[Score],[d])).
```

รูปที่ 3.14แสดงเงื่อนไขในการจัดอันดับด้วยฟังก์ชันสมาชิก Trapezoidal Function

จากรูปที่ 3.14 สามารถอธิบายเงื่อนไขการจัดอันดับได้ดังนี้

- ถ้า Amount มีค่ามากกว่า 15 แต่ไม่เกิน 21 ค่า score จะมีค่าเท่ากับ 1
- ถ้า Amount มีค่าน้อยกว่า 15 การคำนวณค่า score = (Amount-0)/(15-0).
- ถ้า Amount มีค่ามากกว่า 21 การคำนวณค่า score = (40-Amount)/(40-21).
- เมื่อ Amount คือค่าที่จะนำมาคำนวณหาค่าความเป็นสมาชิกโดยที่ 40 และ 0 คือค่าสูงสุดและต่ำสุดในช่วงของข้อมูลที่สนใจตามลำดับ และค่าที่มากกว่า 15 แต่ไม่เกิน 21คือค่าที่สนใจจะได้ผลลัพธ์การจัดอันดับtop-k ด้วย Trapezoidal function ดังรูปที่ 3.15



รูปที่ 3.15 แสดงผลลัพธ์การจัดอันดับ top-k ด้วย Trapezoidal Function

จากรูปที่ 3.15 เมื่อจัดอันดับ top-k ด้วย Trapezoidal Function จะได้ข้อมูลที่มีค่าใกล้เคียงระหว่าง 15 ถึง 21 มากที่สุด 5 อันดับแรก ได้แก่ ff, rr, kk, hhและ jjโดยมีค่าความเป็นสมาชิก 1, 1, 1, 0.947 และ 0.933 ตามลำดับ

การจัดอันดับด้วยการใช้ฟังก์ชันสมาชิก Left Shoulder Function โดยกำหนดเงื่อนไขว่าต้องการข้อมูลที่มีค่าใกล้เคียงกับ 18 มากที่สุด 5 อันดับแรก สามารถเขียนเงื่อนไขได้ดังรูปที่ 3.16

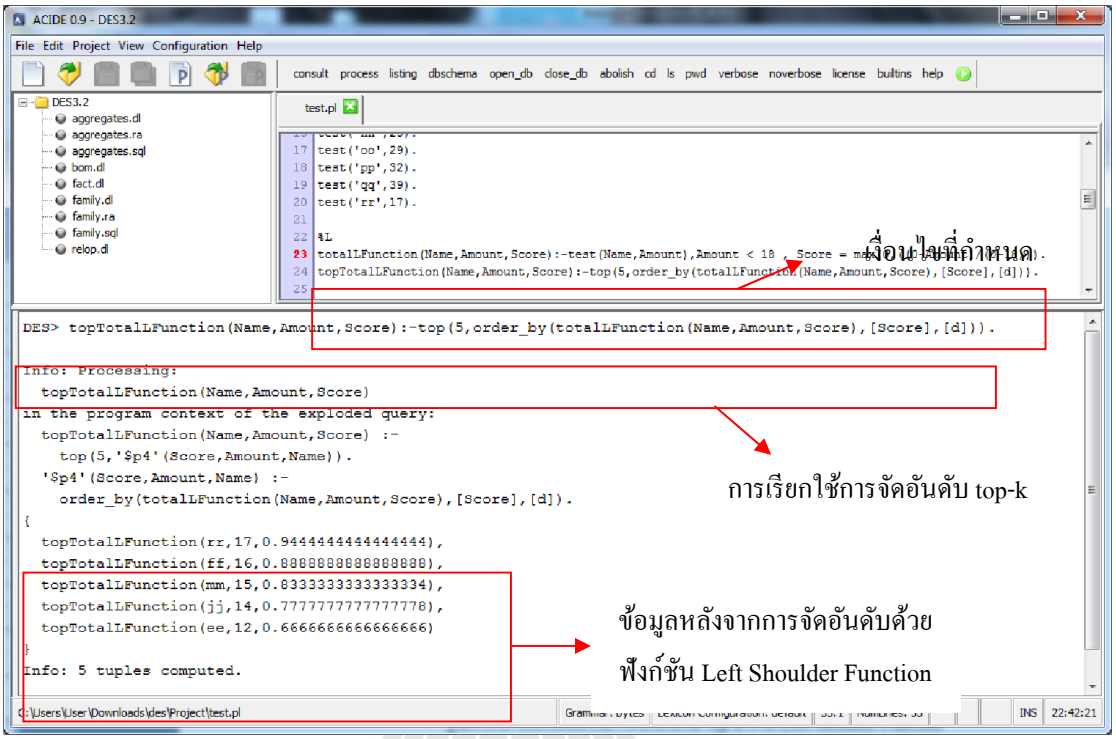
```
% Left Shoulder Function
totalLFunction(Name,Amount,Score):-test(Name,Amount),Amount > 18 ,
                                    Score = max(0,((0-Amount)/(0-18))).
topTotalLFunction(Name,Amount,Score):-
                                    top(5,order_by(totalLFunction(Name,Amount,Score),[Score],[d])).
```

รูปที่ 3.16แสดงเงื่อนไขในการจัดอันดับด้วยฟังก์ชันสมาชิก Left Shoulder Function

จากรูปที่ 3.16สามารถอธิบายเงื่อนไขการจัดอันดับได้ดังนี้

- ถ้า Amount มีค่าน้อยกว่า18 การคำนวณค่า Score = (0-Amount)/(0-18)

- เมื่อ Amount คือค่าที่เราจะนำมาคำนวณหาค่าความเป็นสมาชิกโดยที่0 คือค่าต่ำสุดในช่วงของข้อมูลที่สนใจและ18 คือค่าที่สนใจจะได้ผลลัพธ์การจัดอันดับ top-k ด้วย Left shoulder function ดังรูปที่ 3.17



รูปที่ 3.17 แสดงผลลัพธ์การจัดอันดับ top-k ด้วย Left Shoulder Function

จากรูปที่ 3.17 เมื่อจัดอันดับ top-k ด้วย Left Shoulder Function จะได้ข้อมูลที่มีค่าใกล้เคียงกับ 18 มากที่สุด 5 อันดับ ดังนี้ rr, ff, mm, jjและ eeโดยมีค่าความเป็นสมาชิก 0.944, 0.889, 0.833, 0.778 และ 0.667 ตามลำดับ

การจัดอันดับด้วยการใช้ฟังก์ชันสมาชิก Right Shoulder Function โดยกำหนดเงื่อนไขว่าต้องการข้อมูลที่มีค่าใกล้เคียงกับ 18 มากที่สุด 5 อันดับแรก สามารถเขียนเงื่อนไขได้ดังรูปที่ 3.18

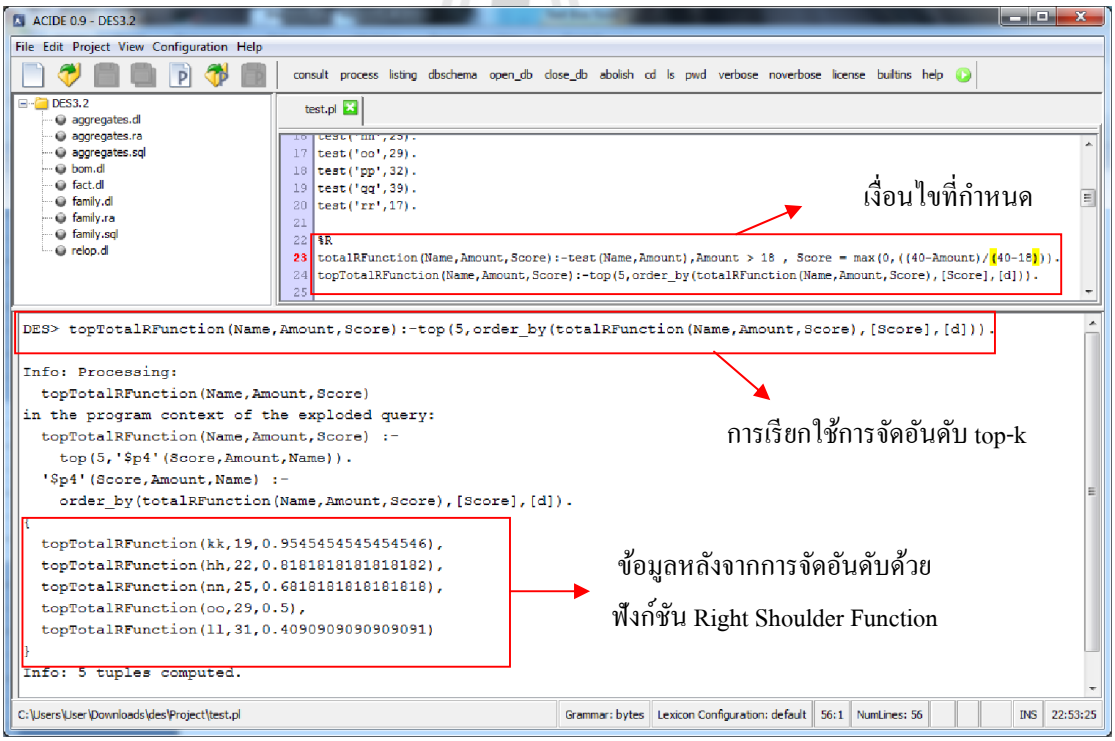
```
% Right Shoulder Function
totalRFunction(Name,Amount,Score):-test(Name,Amount),Amount > 18 ,
                                    Score = max(0,((40-Amount)/(40-18))).
topTotalLFunction(Name,Amount,Score):-
                        top(5,order_by(totalLFunction(Name,Amount,Score),[Score],[d])).
```

รูปที่ 3.18แสดงเงื่อนไขในการจัดอันดับด้วยฟังก์ชันสมาชิก Right Shoulder Function

จากรูปที่ 3.18สามารถอธิบายเงื่อนไขการจัดอันดับได้ดังนี้

- ถ้า Amount มีค่ามากกว่า 18 การคำนวณค่า Score = (40-Amount)/(40-18)

- เมื่อ Amount คือค่าที่เราจะนำมาคำนวณหาค่าความเป็นสมาชิกโดยที่40 คือค่าสูงสุดในช่วงของข้อมูลที่สนใจ และ18 คือค่าที่สนใจจะได้ผลลัพธ์การจัดอันดับ top-k ด้วย Right shoulder function ดังรูปที่ 3.19



รูปที่ 3.19 แสดงผลลัพธ์การจัดอันดับ top-k ด้วย Right Shoulder Function

จากรูปที่ 3.19เมื่อจัดอันดับ top-k ด้วย Right Shoulder Function จะได้ข้อมูลที่มีค่าใกล้เคียงกับ 18 มากที่สุด 5 อันดับ ดังนี้ kk, hh, nn, ooและ llโดยมีค่าความเป็นสมาชิก 0.955, 0.818, 0.682, 0.5 และ 0.409 ตามลำดับ

3.4 เครื่องมือที่ใช้ในการวิจัย

เครื่องมือที่ใช้ในงานวิจัยนี้ ประกอบด้วยฮาร์ดแวร์และซอฟต์แวร์ ดังนี้

1. เครื่องคอมพิวเตอร์สำหรับพัฒนาโปรแกรมการจัดอันดับ top-k ข้อมูลจำนวนประชากรโลก โดยมีรายละเอียด ดังนี้

- หน่วยประมวลผลกลาง : Intel® Core i5-2410M CPU @ 2.30 GHz 2.30 GHz
- หน่วยความจำสำรอง 640 GB
- หน่วยความจำหลัก 8.00 GB
- อุปกรณ์เสริมอื่นๆ เช่น เมาส์ แป้นพิมพ์ เป็นต้น

2. ระบบปฏิบัติการและโปรแกรมประยุกต์สำหรับการพัฒนาโปรแกรมการจัดอันดับ top-k ข้อมูลจำนวนประชากรโลก ประกอบไปด้วย

- ระบบปฏิบัติการ : Windows 7 Ultimate 64-bit Operating System
- เครื่องมือในการแบ่งกลุ่ม :Weka 3.6
- เครื่องมือในการพัฒนาโปรแกรม : Datalog Education System3.2

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 4
การทดสอบและอภิปรายผล

การทดสอบประสิทธิภาพของระบบนั้น จะเป็นการเปรียบเทียบการตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรกด้วยฟังก์ชันสมาชิกแต่ละฟังก์ชัน ซึ่งในการเปรียบเทียบการใช้ฟังก์ชันสมาชิกนั้นจะเป็นการเปรียบเทียบการใช้งานในระหว่างสองคู่ฟังก์ชัน คือTrapezoidal Function เปรียบเทียบกับ Triangular Function และระหว่างฟังก์ชัน Left Shoulder Function เปรียบเทียบกับ Right Shoulder Function

4.1 ข้อมูลที่ใช้ในการทดสอบ

ในการทดสอบการตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัยจะใช้ข้อมูลมาตรฐาน ซึ่งเป็นข้อมูลเกี่ยวกับจำนวนประชากรโลก (Population) ปี 2012 สามารถดาวโหลดได้ที่ http://databank.worldbank.org/data/ โดยข้อมูลทั้งหมดมี218 แถว 3 คอลัมน์ ประกอบด้วย ชื่อประเทศ(ย่อ)ชื่อประเทศ และจำนวนประชากร โดยมีรายละเอียดตัวอย่างข้อมูลดังตารางที่ 4.1

มหาวิทยาลัยเทคโนโลยีสุรนารี

ตารางที่ 4.1ตัวอย่างข้อมูลจำนวนประชากรโลก (Population)

ชื่อประเทศ (ย่อ)	ชื่อประเทศ	จำนวนประชากร
ABW	Aruba	102,384
ADO	Andorra	78,369
AFG	Afghanistan	29,824,536
AGO	Angola	20,820,525
ALB	Albania	3,162,083
ARE	United Arab Emirates	9,205,651
ARG	Argentina	41,086,927
ARM	Armenia	2,969,081
ASM	American Samoa	55,128
ATG	Antigua and Barbuda	89,069
AUS	Australia	22,683,600
AUT	Austria	8,426,446
AZE	Azerbaijan	9,297,507
BDI	Burundi	9,849,569
BEL	Belgium	11,142,157
BEN	Benin	10,050,702
BFA	Burkina Faso	16,460,141
BGD	Bangladesh	154,695,368
BGR	Bulgaria	7,304,632
BHR	Bahrain	1,317,827
BHS	Bahamas, The	371,960
BIH	Bosnia and Herzegovina	3,833,916
BLR	Belarus	9,464,000
BLZ	Belize	324,060
BMU	Bermuda	64,806
BOL	Bolivia	10,496,285
BRA	Brazil	198,656,019

4.2 การแบ่งกลุ่มด้วยอัลกอริทึม k-means

4.2.1 เตรียมข้อมูลสำหรับการแบ่งกลุ่มด้วยอัลกอริทึม k-means

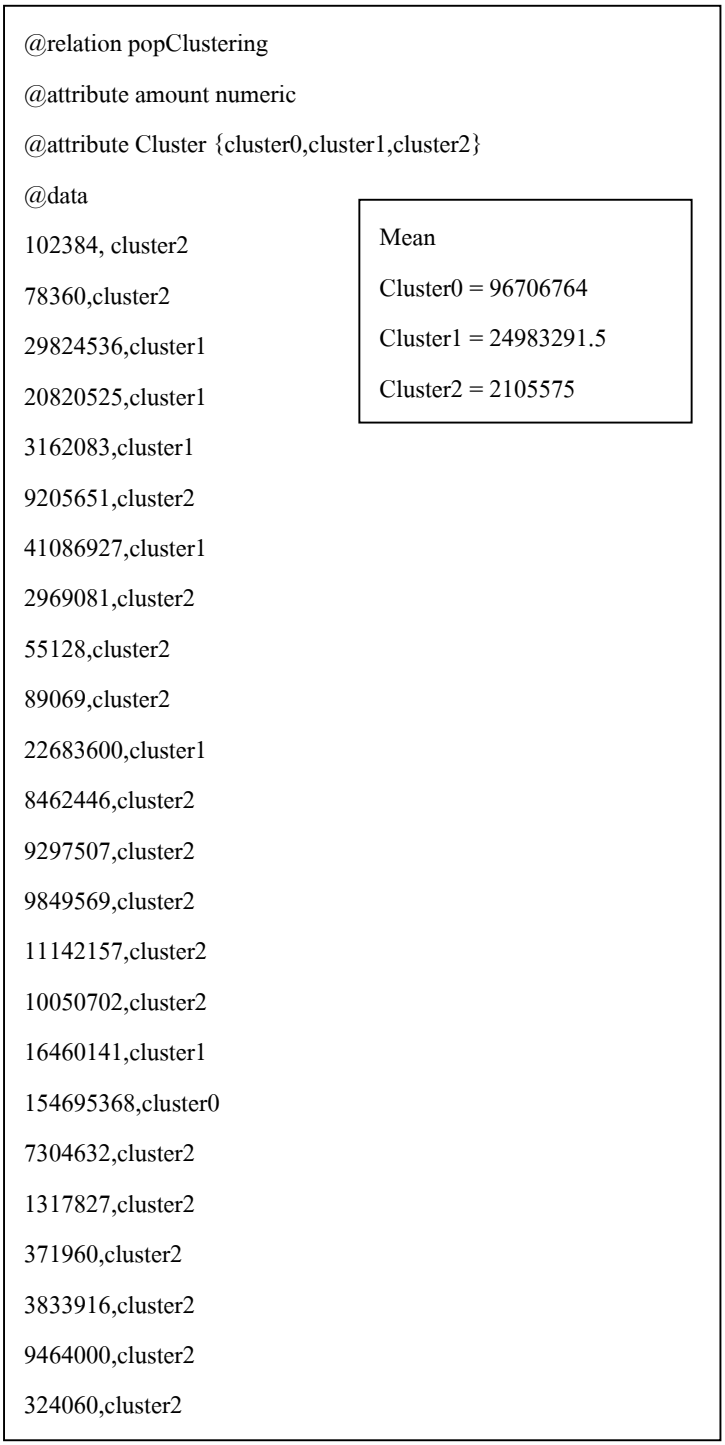
ในการแบ่งกลุ่มข้อมูลสำหรับการตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรก แบบคลุมเครือในฐานข้อมูลนิรนัย จะใช้โปรแกรม Weka ในการแบ่งกลุ่มด้วยอัลกอริทึม k-means และข้อมูลจะอยู่ในรูปแบบของไฟล์ .arff เพื่อให้มีความเหมาะสมกับการทำงานของโปรแกรมการจัดกลุ่ม ด้วยข้อจำกัดของการจัดกลุ่มด้วยเทคนิค k-means clustering การจัดกลุ่มด้วยเทคนิคนี้จำเป็นที่จะต้องนำเฉพาะคอลัมน์จำนวนประชากรโลกเพียงคอลัมน์เดียวไปใช้ในการจัดกลุ่มเพื่อความถูกต้องของการจัดกลุ่มโดยมีตัวอย่างข้อมูลจำนวนประชากรโลกในรูปแบบ .arff ดังรูปที่ 4.1

```
@relation popClustering
@attribute amount numeric
@data
102384
78360
29824536
20820525
3162083
9205651
41086927
2969081
55128
89069
22683600
8462446
9297507
9849569
11142157
10050702
16460141
154695368
7304632
1317827
```

รูปที่ 4.1ตัวอย่างข้อมูลจำนวนประชากรโลกที่อยู่ในรูปแบบของ .arff

4.2.2 ผลของการแบ่งกลุ่มด้วยอัลกอริทึม k-means

เมื่อนำข้อมูลที่เตรียมไว้สำหรับการแบ่งกลุ่มด้วยอัลกอริทึม k-means ไปแบ่งกลุ่มด้วยโปรแกรม Weka แล้ว จะได้ผลลัพธ์ดังรูปที่ 4.2



รูปที่ 4.2ตัวอย่างข้อมูลที่ผ่านการแบ่งกลุ่มด้วยอัลกอริทึม k-means

จากรูปที่ 4.2สามารถวิเคราะห์ผลการแบ่งกลุ่มได้ว่า Cluster0 แทนกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับสูง , Cluster1แทนกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง และ Cluster2แทนกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ

4.3 การตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรก

4.3.1 สร้างฟังก์ชันสมาชิกแต่ละฟังก์ชัน

เมื่อพิจารณาข้อมูลจำนวนประชากรโลกที่ผ่านการแบ่งกลุ่มทั้งหมด จะได้ว่า

- ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับสูง จะมีจำนวนประชากรต่ำสุดของกลุ่มอยู่ที่ 60,917,978 คน และสูงสุดอยู่ที่1,350,695,000 คน

- ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง จะมีจำนวนประชากรต่ำสุดของกลุ่มอยู่ที่ 13,724,317 คน และสูงสุดอยู่ที่ 52,797,319 คน

- ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ จะมีจำนวนประชากรต่ำสุดของกลุ่มอยู่ที่ 9,860 คน และสูงสุดอยู่ที่ 12,448,175 คน

การสร้างฟังก์ชันสมาชิกแต่ละฟังก์ชัน ผู้วิจัยได้กำหนดขอบเขตของจำนวนประชากรโลกด้วยการวิเคราะห์จากค่าสูงสุดและค่าต่ำสุดของแต่ละกลุ่ม โดยขอบเขตของจำนวนประชากรโลกในแต่ละกลุ่มจะต้องไม่อยู่นอกเหนือจากค่าต่ำสุดและค่าสูงสุดของแต่ละกลุ่ม

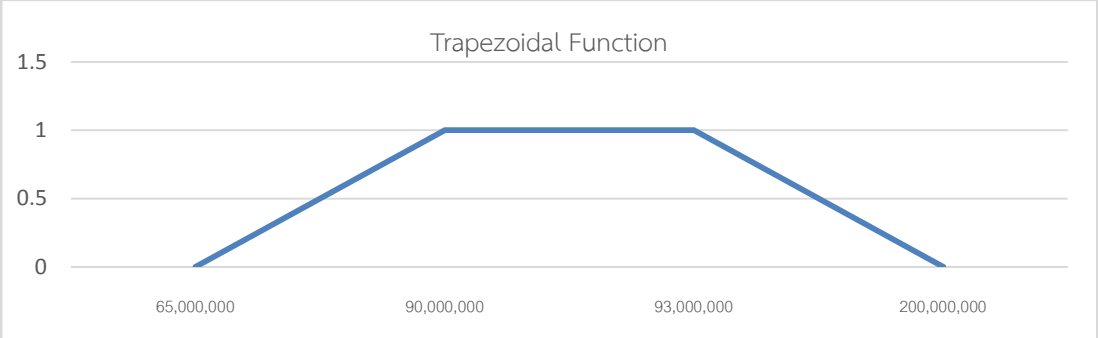
1. Trapezoidal Function

ฟังก์ชันสมาชิก Trapezoidal Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับสูง ผู้วิจัยได้เลือกใช้คำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 90,000,000 คน ถึง 93,000,000 คน โดยประเทศที่นำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 65,000,000 คน แต่ไม่ถึง 200,000,000 คน"ซึ่งทำให้ได้ฟังก์ชันสมาชิก Trapezoidal Function ดังรูปที่ 4.3

```
Amount > 90000000 , Amount < 93000000 , Score = max(0,1).
Amount < 90000000 , Score = max(0,((Amount-65000000)/(90000000-65000000))).
Amount > 93000000 , Score = max(0,((200000000-Amount)/(200000000-93000000))).
```

รูปที่ 4.3แสดง Trapezoidal Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับสูง

เมื่อนำมาแสดงในรูปแบบกราฟ จะสามารถแสดงได้ดังรูปที่ 4.4



รูปที่ 4.4กราฟแสดง Trapezoidal Function ของกลุ่มประเทศที่มีประชากรในระดับสูง

ฟังก์ชันสมาชิก Trapezoidal Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง ผู้วิจัยได้เลือกใช้คำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 24,000,000 คน ถึง 27,000,000 คน โดยประเทศที่นำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 15,000,000 คน แต่ไม่ถึง 35,000,000 คน" ซึ่งทำให้ได้ฟังก์ชันสมาชิก Trapezoidal Function ดังรูปที่ 4.5

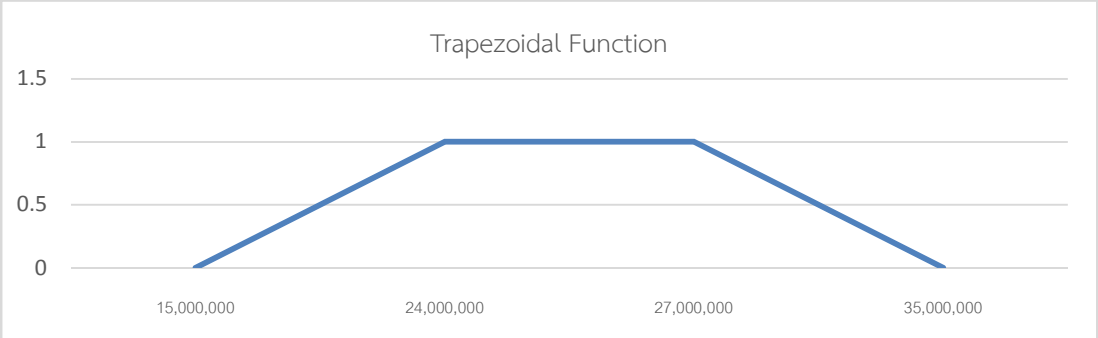
Amount > 24000000 , Amount < 27000000 , Score = max(0,1).

Amount < 24000000 , Score = max(0,((Amount-15000000)/(24000000-15000000))).

Amount > 27000000 , Score = max(0,((35000000-Amount)/(35000000-27000000))).

รูปที่ 4.5แสดง Trapezoidal Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง

เมื่อนำมาแสดงในรูปแบบกราฟ จะสามารถแสดงได้ดังรูปที่ 4.6



รูปที่ 4.6กราฟแสดง Trapezoidal Function ของกลุ่มประเทศที่มีประชากรในระดับปานกลาง

ฟังก์ชันสมาชิก Trapezoidal Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ ผู้วิจัยได้เลือกใช้คำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 2,050,000 คน ถึง 2,500,000 คน โดยประเทศที่นำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 1,000,000 คน แต่ไม่ถึง 5,000,000 คน" ซึ่งทำให้ได้ฟังก์ชันสมาชิก Trapezoidal Function ดังรูปที่ 4.7

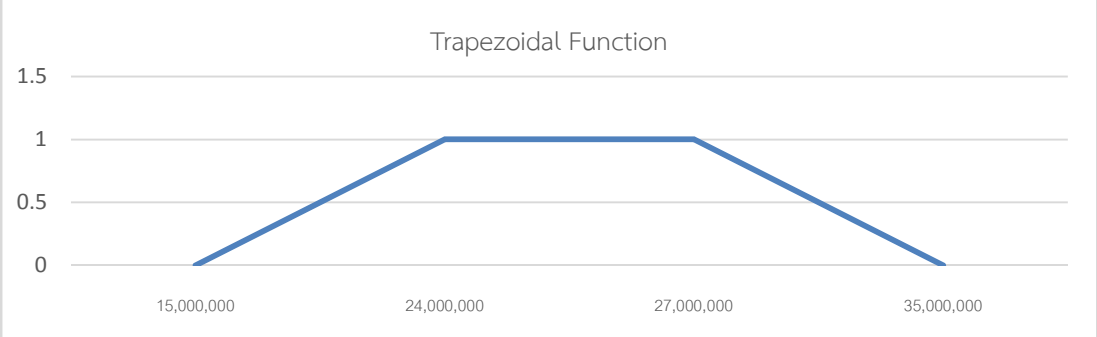
Amount >2050000 , Amount < 2500000 , Score = max(0,1).

Amount < 2050000 , Score = max(0,((Amount-1000000)/(2050000-1000000))).

Amount > 2500000 , Score = max(0,((5000000-Amount)/(5000000-2500000))).

รูปที่ 4.7กราฟแสดง Trapezoidal Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ

เมื่อนำมาแสดงในรูปแบบกราฟ จะสามารถแสดงได้ดังรูปที่ 4.8



รูปที่ 4.8กราฟแสดง Trapezoidal Function ของกลุ่มประเทศที่มีประชากรอยู่ในระดับต่ำ

2. Triangular Function

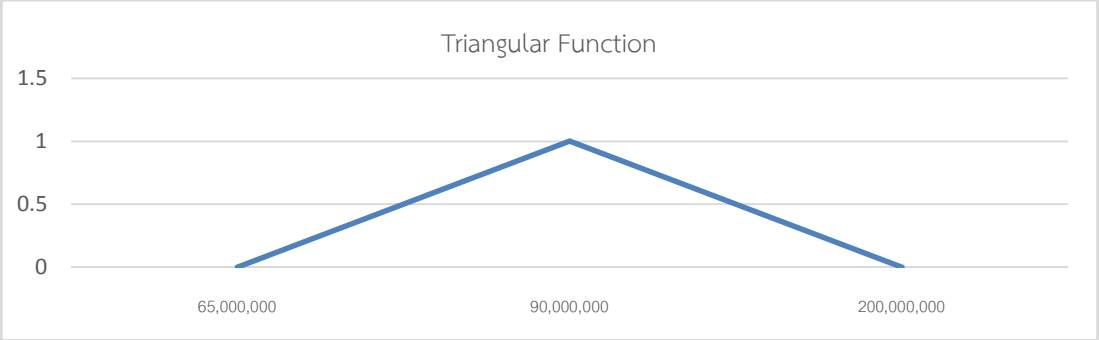
ฟังก์ชันสมาชิก Triangular Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับสูง ผู้วิจัยได้เลือกใช้คำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 90,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 65,000,000 คน แต่ไม่ถึง 200,000,000 คน" ซึ่งทำให้ได้ฟังก์ชันสมาชิก Triangular Function ดังรูปที่ 4.9

Amount < 90000000 , Score = max(0,((Amount-65000000)/(90000000-65000000))).

Amount > 90000000 , Score = max(0,((200000000-Amount)/(200000000-90000000))).

รูปที่ 4.9แสดง Triangular Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับสูง

เมื่อนำมาแสดงในรูปแบบกราฟ จะสามารถแสดงได้ดังรูปที่ 4.10



รูปที่ 4.10กราฟแสดง Triangular Function ของกลุ่มประเทศที่มีประชากรอยู่ในระดับสูง

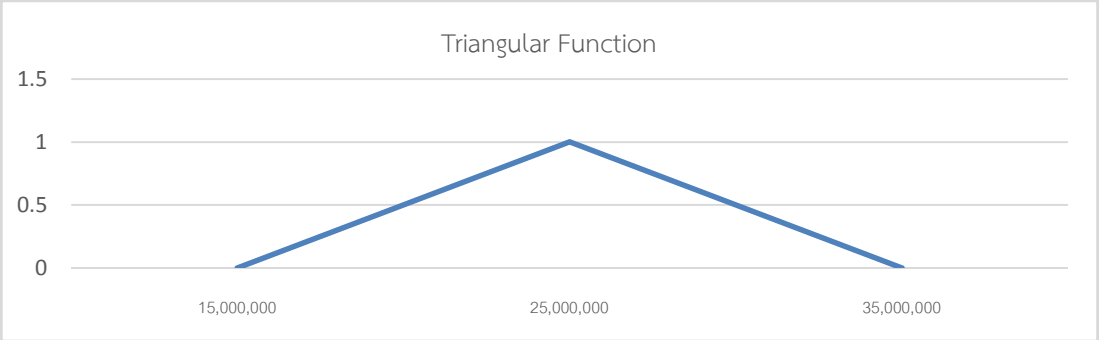
ฟังก์ชันสมาชิก Triangular Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง ผู้วิจัยได้เลือกใช้คำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 25,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 15,000,000 คน แต่ไม่ถึง 35,000,000 คน" ซึ่งทำให้ได้ฟังก์ชันสมาชิก Triangular Function ดังรูปที่ 4.11

Amount < 25000000 , Score = max(0,((Amount-15000000)/(25000000-15000000))).

Amount > 25000000 , Score = max(0,((35000000-Amount)/(35000000-25000000))).

รูปที่ 4.11แสดง Triangular Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง

เมื่อนำมาแสดงในรูปแบบกราฟ จะสามารถแสดงได้ดังรูปที่ 4.12



รูปที่ 4.12กราฟแสดง Triangular Function ของกลุ่มประเทศที่มีประชากรอยู่ในระดับปานกลาง

ฟังก์ชันสมาชิก Triangular Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ ผู้วิจัยได้เลือกใช้คำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,100,000 คน โดย

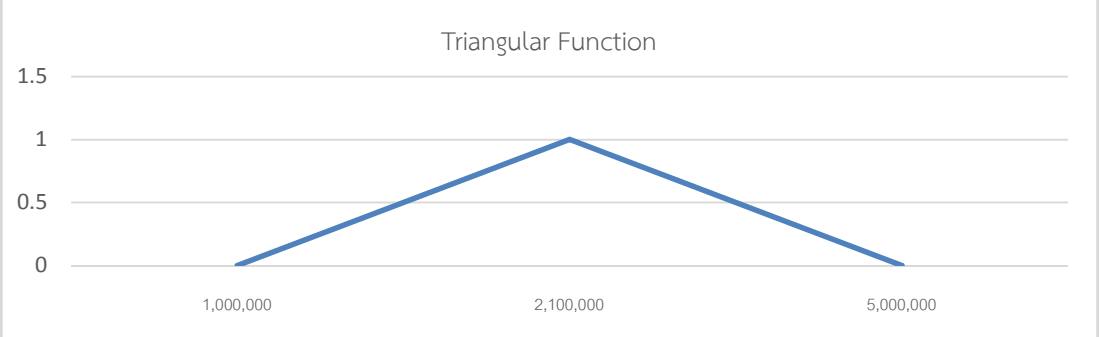
ประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 1,000,000 คน แต่ไม่ถึง 5,000,000 คน" ซึ่งทำให้ได้ฟังก์ชันสมาชิก Triangular Function ดังรูปที่ 4.13

Amount < 2100000 , Score = max(0,((Amount-1000000)/(2100000-1000000))).

Amount >2100000 , Score = max(0,((5000000-Amount)/(5000000-2100000))).

รูปที่ 4.13แสดง Triangular Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ

เมื่อนำมาแสดงในรูปแบบกราฟ จะสามารถแสดงได้ดังรูปที่ 4.14



รูปที่ 4.14กราฟแสดง Triangular Function ของกลุ่มประเทศที่มีประชากรอยู่ในระดับต่ำ

3. Left Shoulder Function

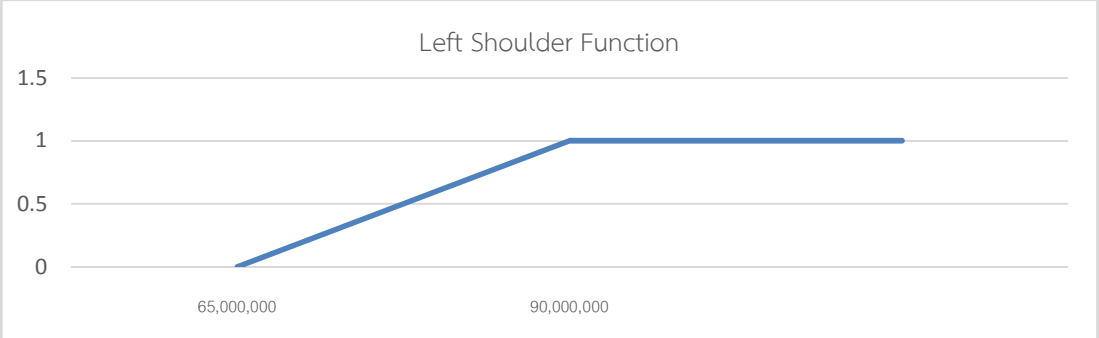
ฟังก์ชันสมาชิก Left Shoulder Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับสูง ผู้วิจัยได้เลือกใช้คำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 90,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 65,000,000 คน แต่ไม่เกิน90,000,000 คน" ซึ่งทำให้ได้ฟังก์ชันสมาชิก Left Shoulder Function ดังรูปที่ 4.15

Amount < 90000000, Amount > 65000000 ,

Score = max(0,((Amount-65000000)/(90000000-65000000))).

รูปที่ 4.15แสดง Left Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับสูง

เมื่อนำมาแสดงในรูปแบบกราฟ จะสามารถแสดงได้ดังรูปที่ 4.16



รูปที่ 4.16กราฟแสดง Left Shoulder Function ของกลุ่มประเทศที่มีประชากรอยู่ในระดับสูง

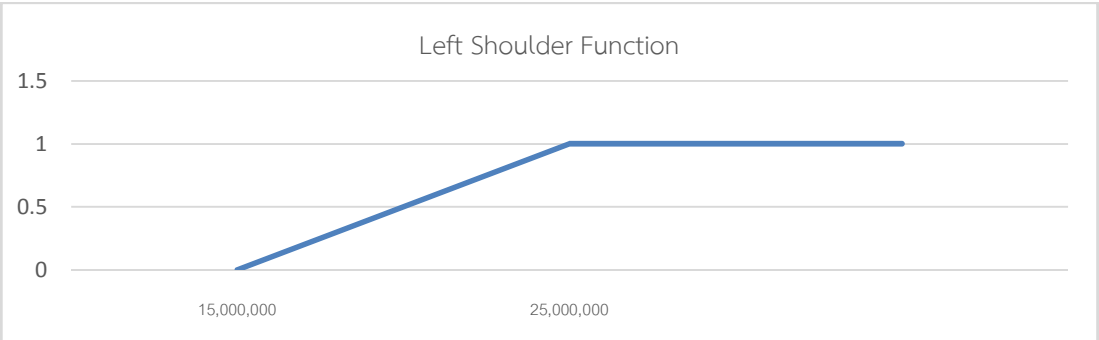
ฟังก์ชันสมาชิก Left Shoulder Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง ผู้วิจัยได้เลือกใช้คำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 25,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 15,000,000 คน แต่ไม่เกิน 25,000,000 คน" ซึ่งทำให้ได้ฟังก์ชันสมาชิก Left Shoulder Function ดังรูปที่ 4.17

Amount < 25000000, Amount > 15000000 ,

Score = max(0,((Amount-15000000)/(25000000-15000000))).

รูปที่ 4.17แสดง Left Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง

เมื่อนำมาแสดงในรูปแบบกราฟ จะสามารถแสดงได้ดังรูปที่ 4.18



รูปที่ 4.18กราฟแสดง Left Shoulder Function ของกลุ่มประเทศที่มีประชากรอยู่ในระดับปานกลาง

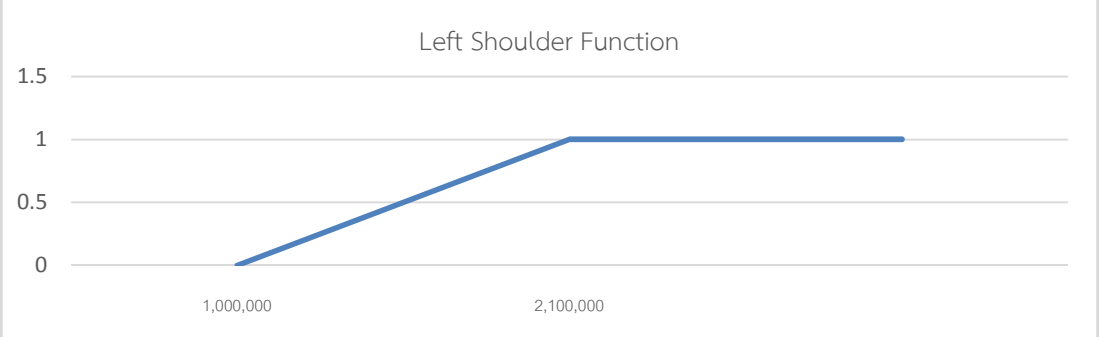
ฟังก์ชันสมาชิก Left Shoulder Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ ผู้วิจัยได้เลือกใช้คำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,100,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 1,000,000 คน แต่ไม่ถึง 2,100,000 คน" ซึ่งทำให้ได้ฟังก์ชันสมาชิก Left Shoulder Function ดังรูปที่ 4.19

Amount < 2100000, Amount > 1000000 ,

Score = max(0,((Amount-1000000)/(2100000-1000000))).

รูปที่ 4.19แสดง Left Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ

เมื่อนำมาแสดงในรูปแบบกราฟ จะสามารถแสดงได้ดังรูปที่ 4.20



รูปที่ 4.20กราฟแสดง Left Shoulder Function ของกลุ่มประเทศที่มีประชากรอยู่ในระดับต่ำ

4. Right Shoulder Function

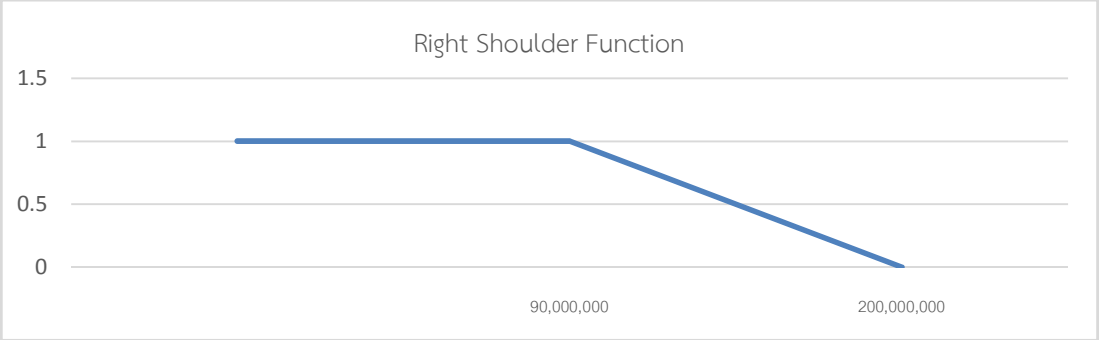
ฟังก์ชันสมาชิก Right Shoulder Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับสูง ผู้วิจัยได้เลือกใช้คำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 90,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 90,000,000คน แต่ไม่ถึง200,000,000 คน" ซึ่งทำให้ได้ฟังก์ชันสมาชิก Right Shoulder Function ดังรูปที่ 4.21

Amount > 90000000, Amount < 200000000 ,

Score = max(0,((200000000-Amount)/(200000000-90000000))).

รูปที่ 4.21แสดง Right Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับสูง

เมื่อนำมาแสดงในรูปแบบกราฟ จะสามารถแสดงได้ดังรูปที่ 4.22



รูปที่ 4.22กราฟแสดง Right Shoulder Function ของกลุ่มประเทศที่มีประชากรอยู่ในระดับสูง

ฟังก์ชันสมาชิก Right Shoulder Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง ผู้วิจัยได้เลือกใช้คำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 25,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 25,000,000 คน แต่ไม่เกิน 35,000,000คน" ซึ่งทำให้ได้ฟังก์ชันสมาชิก Right Shoulder Function ดังรูปที่ 4.23

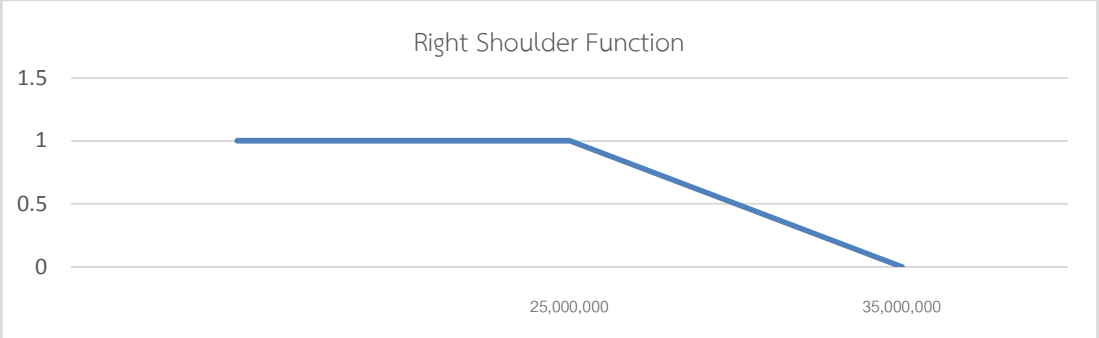
Amount > 25000000, Amount < 35000000 ,

Score = max(0,((35000000-Amount)/(35000000-25000000))).

รูปที่ 4.23แสดง Right Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง

มหาวิทยาลัยเทคโนโลยีสุรนารี

เมื่อนำมาแสดงในรูปแบบกราฟ จะสามารถแสดงได้ดังรูปที่ 4.24



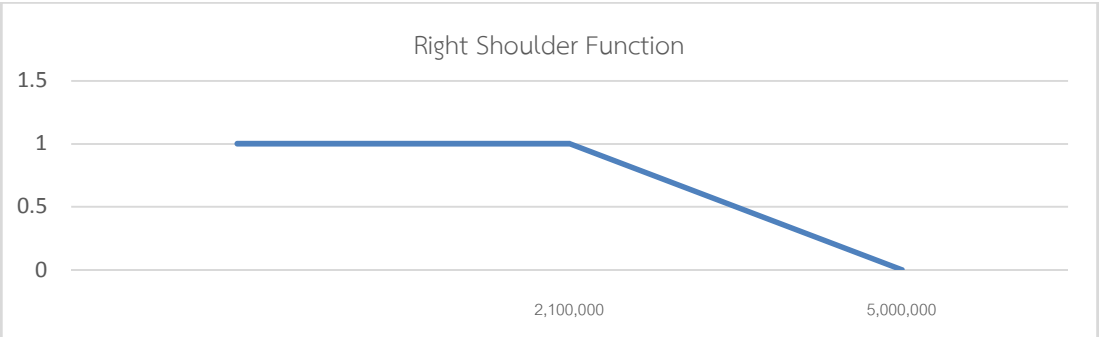
รูปที่ 4.24กราฟแสดง Right Shoulder Function ของกลุ่มประเทศที่มีประชากรอยู่ในระดับปานกลาง

ฟังก์ชันสมาชิก Right Shoulder Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ ผู้วิจัยได้เลือกใช้คำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,100,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 2,100,000 คน แต่ไม่ถึง 5,000,000 คน" ซึ่งทำให้ได้ฟังก์ชันสมาชิก Right Shoulder Function ดังรูปที่ 4.25

```
Amount > 2100000, Amount < 5000000 ,
Score = max(0,((5000000-Amount)/(5000000-2100000))).
```

รูปที่ 4.25แสดง Right Shoulder Function ของกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ

เมื่อนำมาแสดงในรูปแบบกราฟ จะสามารถแสดงได้ดังรูปที่ 4.26



รูปที่ 4.26กราฟแสดง Right Shoulder Function ของกลุ่มประเทศที่มีประชากรอยู่ในระดับต่ำ

4.3.2 เตรียมข้อมูลสำหรับการตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัย

ในการตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัย จะใช้โปรแกรม DESในการหา เค อันดับแรกแบบคลุมเครือด้วยฟังก์ชันสมาชิกทั้ง 4 ฟังก์ชัน และข้อมูลจะต้องอยู่ในรูปแบบของภาษาดาต้าล็อก(.pl) เพื่อให้มีความเหมาะสมกับการทำงานของโปรแกรมการตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัยโดยมีตัวอย่างข้อมูลจำนวนประชากรโลกที่ผ่านการจัดกลุ่มแล้วในรูปแบบภาษาดาต้าล็อก(.pl)ดังรูปที่ 4.27

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
%database code of national
:-type(popCode(name:string,code:string)).
popCode('Afghanistan','AFG').
popCode('Albania','ALB').
popCode('Algeria','DZA').
popCode('American Samoa','ASM').
popCode('Andorra','ADO').
popCode('Angola','AGO').
popCode('Antigua and Barbuda','ATG').
popCode('Argentina','ARG').
popCode('Armenia','ARM').
popCode('Aruba','ABW').
popCode('Australia','AUS').
popCode('Austria','AUT').
popCode('Azerbaijan','AZE').
popCode('Bahamas','BHS').
popCode('Bahrain','BHR').
popCode('Bangladesh','BGD').
popCode('Barbados','BRB').
popCode('Belarus','BLR').
popCode('Belgium','BEL').
popCode('Belize','BLZ').
popCode('Benin','BEN').
popCode('Bermuda','BMU').
popCode('Bhutan','BTN').
popCode('Bolivia','BOL').
popCode('Bosnia and Herzegovina','BIH').
popCode('Botswana','BWA').
popCode('Brazil','BRA').
popCode('Brunei Darussalam','BRN').
popCode('Bulgaria','BGR').
popCode('Burkina Faso','BFA').
popCode('Burundi','BDI').
popCode('Cambodia','KHM').
popCode('Cameroon','CMR').
popCode('Canada','CAN').
popCode('Cape Verde','CPV').
popCode('Caribbean small states','CSS').
popCode('Cayman Islands','CYM').
popCode('Central African Republic','CAF').
popCode('Chad','TCD').
popCode('Channel Islands','CHI').
popCode('Chile','CHL').
popCode('China','CHN').
```

```
%database Population Total
:-type(popTotal(code:string,amount:int)).
popTotal('AFG',29824536).
popTotal('ALB',3162083).
popTotal('DZA',38481705).
popTotal('ASM',55128).
popTotal('ADO',78360).
popTotal('AGO',20820525).
popTotal('ATG',89069).
popTotal('ARG',41086927).
popTotal('ARM',2969081).
popTotal('ABW',102384).
popTotal('AUS',22683600).
popTotal('AUT',8462446).
popTotal('AZE',9297507).
popTotal('BHS',371960).
popTotal('BHR',1317827).
popTotal('BGD',154695368).
popTotal('BRB',283221).
popTotal('BLR',9464000).
popTotal('BEL',11142157).
popTotal('BLZ',324060).
popTotal('BEN',10050702).
popTotal('BMU',64806).
popTotal('BTN',741822).
popTotal('BOL',10496285).
popTotal('BIH',3833916).
popTotal('BWA',2003910).
popTotal('BRA',198656019).
popTotal('BRN',412238).
popTotal('BGR',7304632).
popTotal('BFA',16460141).
popTotal('BDI',9849569).
popTotal('KHM',14864646).
popTotal('CMR',21699631).
popTotal('CAN',34880491).
popTotal('CPV',494401).
popTotal('CSS',6968753).
popTotal('CYM',57570).
popTotal('CAF',4525209).
popTotal('TCD',12448175).
popTotal('CHI',161235).
popTotal('CHL',17464814).
popTotal('CHN',1350695000).
```

รูปที่ 4.27ตัวอย่างข้อมูลจำนวนประชากรโลกในรูปแบบของภาษาดาต้าล็อก (.pl)

4.4 ผลการตอบคำถามการจัดเรียง เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัย

4.4.1 ผลการทดลองจากการตอบคำถามการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Trapezoidal Function

จากการทดลองหาคำตอบการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Trapezoidal Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับสูง ด้วยคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 90,000,000 คน ถึง 93,000,000 คน โดยประเทศที่นำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 65,000,000 คน แต่ไม่ถึง 200,000,000 คน" ได้ผลการทดลองดังตารางที่ 4.2

ตารางที่ 4.2แสดง 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 90,000,000 คน ถึง 93,000,000 คน ด้วยฟังก์ชัน Trapezoidal Function

ชื่อประเทศ	จำนวนประชากร	ค่าความเป็นสมาชิก
Ethiopia	91,728,849	1
Philippines	96,706,764	0.9654
Vietnam	88,775,500	0.9510
Mexico	120,847,477	0.7397
Japan	127,561,489	0.6770
Germany	81,889,839	0.6756
Arab Republic of Egypt	80,721,874	0.6289
Russian Federation	143,533,000	0.5277
Islamic Republic of Iran	76,424,443	0.4570
Bangladesh	154,695,368	0.4234

จากตารางที่ 4.2สามารถสรุปได้ว่า ประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 90,000,000 คน ถึง 93,000,000 คน โดยประเทศที่นำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 65,000,000 คน แต่ไม่ถึง 200,000,000 คน 10 อันดับแรก ได้แก่ประเทศ Ethiopia, Philippines, Vietnam, Mexico, Japan, Germany, Arab Republic of Egypt, Russian Federation,Islamic Republic of Iran และประเทศ Bangladesh ซึ่งมีค่าความเป็นสมาชิก 1, 0.9654, 0.9510, 0.7397, 0.6770, 0.6756, 0.6289, 0.5277, 0.4570และ 0.4234ตามลำดับ โดยจะเห็นได้ว่าจะมีประเทศที่มีค่าความเป็น

สมาชิกเป็น 1 เนื่องจากประเทศเหล่านั้นมีจำนวนประชากรอยู่ในช่วง 90,000,000 คน ถึง 93,000,000 คน จึงส่งผลให้มีค่าความเป็นสมาชิกสูงที่สุด

จากการทดลองหาคำตอบการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Trapezoidal Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง ด้วยคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 24,000,000 คน ถึง 27,000,000 คน โดยประเทศที่นำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 15,000,000 คน แต่ไม่ถึง 35,000,000 คน" ได้ผลการทดลองดังตารางที่ 4.3

ตารางที่ 4.3แสดง 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 24,000,000 คน ถึง 27,000,000 คน ด้วยฟังก์ชัน Trapezoidal Function

ประเทศ	จำนวนประชากร	ค่าความเป็นสมาชิก
Ghana	25,366,462	1
Mozambique	25,203,395	1
Democratic People Republic of Korea	24,763,188	1
Republic of Yemen	23,852,409	0.9836
Nepal	27,474,377	0.9407
Australia	22,683,600	0.8537
Saudi Arabia	28,287,855	0.8390
Syrian Arab Republic	22,399,254	0.8221
Madagascar	22,293,914	0.8104
Cameroon	21,699,631	0.7444

จากตารางที่ 4.3สามารถสรุปได้ว่า ประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 24,000,000 คน ถึง 27,000,000 คน โดยประเทศที่นำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 15,000,000 คน แต่ไม่ถึง 35,000,000 คน 10 อันดับแรก ได้แก่ประเทศ Ghana, Mozambique, Democratic People Republic of Korea, Republic of Yemen, Nepal, Australia, Saudi Arabia, Syrian Arab Republic, Madagascar และประเทศ Cameroonซึ่งมีค่าความเป็นสมาชิก 1, 1, 1, 0.9836, 0.9407, 0.8537, 0.8390, 0.8221, 0.8104 และ0.7444 ตามลำดับ โดยจะเห็นได้ว่าจะมี

ประเทศที่มีค่าความเป็นสมาชิกเป็น 1 เนื่องจากประเทศเหล่านั้นมีจำนวนประชากรอยู่ในช่วง 24,000,000 คน ถึง 27,000,000 คน จึงส่งผลให้มีค่าความเป็นสมาชิกสูงที่สุด

จากการทดลองหาคำตอบการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Trapezoidal Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ ด้วยคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 2,050,000 คน ถึง 2,500,000 คน โดยประเทศที่นำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 1,000,000 คน แต่ไม่ถึง 5,000,000 คน" ได้ผลการทดลองดังตารางที่ 4.4

ตารางที่ 4.4แสดง 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 2,050,000 คน ถึง 2,500,000 คน ด้วยฟังก์ชัน Trapezoidal Function

ประเทศ	จำนวนประชากร	ค่าความเป็นสมาชิก
Lesotho	2,051,545	1
Qatar	2,050,514	1
Namibia	2,259,393	1
Macedonia FYR	2,105,575	1
Slovenia	2,058,152	1
Pacific island small states	2,252,782	1
Latvia	2,025,473	0.9766
Botswana	2,003,910	0.9561
Jamaica	2,712,100	0.9152
Mongolia	2,796,484	0.8814

จากตารางที่ 4.4สามารถสรุปได้ว่า ประเทศที่มีจำนวนประชากรใกล้เคียงกับช่วง 2,050,000 คน ถึง 2,500,000 คน โดยประเทศที่นำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 1,000,000 คน แต่ไม่ถึง 5,000,000 คน คน 10 อันดับแรก ได้แก่ประเทศLesotho, Qatar, Namibia, Macedonia FYR, Slovenia, Pacific island small states, Latvia, Botswana, Jamaica และประเทศ Mongoliaซึ่งมีค่าความเป็นสมาชิก 1, 1, 1, 1, 1, 1, 0.9766, 0.9561, 0.9152และ0.8814ตามลำดับ โดยจะเห็นได้ว่าจะมีประเทศที่มีค่าความเป็นสมาชิกเป็น 1 เนื่องจากประเทศเหล่านั้นมีจำนวนประชากรอยู่ในช่วง 2,050,000 คน ถึง 2,500,000 คน จึงส่งผลให้มีค่าความเป็นสมาชิกสูงที่สุด

4.4.2 ผลการทดลองจากการตอบคำถามการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Triangular Function

จากการทดลองหาคำตอบการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Triangular Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับสูง ด้วยคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 90,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 65,000,000 คน แต่ไม่ถึง 200,000,000 คน" ได้ผลการทดลองดังตารางที่ 4.5

ตารางที่ 4.5แสดง 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับ 90,000,000 คน ด้วยฟังก์ชัน Triangular Function

ชื่อประเทศ	จำนวนประชากร	ค่าความเป็นสมาชิก
Ethiopia	91,728,849	0.9843
Vietnam	88,775,500	0.9510
Philippines	96,706,764	0.9390
Mexico	120,847,477	0.7196
Germany	81,889,839	0.6756
Japan	127,561,489	0.6585
Arab Republic of Egypt	80,721,874	0.6289
Russian Federation	143,533,000	0.5133
Islamic Republic of Iran	76,424,443	0.4570
Bangladesh	154,695,368	0.4119

จากตารางที่ 4.5สามารถสรุปได้ว่า ประเทศที่มีจำนวนประชากรใกล้เคียงกับ 90,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 65,000,000 คน แต่ไม่ถึง 200,000,000 คน10 อันดับแรก ได้แก่ประเทศ Ethiopia, Vietnam, Philippines, Mexico, Germany, Japan, Arab Republic of Egypt, Russian Federation, Islamic Republic of Iran และประเทศ Bangladesh ซึ่งมีค่าความเป็นสมาชิก 0.9843, 0.9510, 0.9390, 0.7196, 0.6756, 0.6585, 0.6289, 0.5133, 0.4570 และ 0.4119ตามลำดับ

จากการทดลองหาคำตอบการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Triangular Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง ด้วยคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 25,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 15,000,000 คน แต่ไม่ถึง 35,000,000 คน" ได้ผลการทดลองดังตารางที่ 4.6

ตารางที่ 4.6แสดง 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับ 25,000,000 คน ด้วยฟังก์ชัน Triangular Function

ชื่อประเทศ	จำนวนประชากร	ค่าความเป็นสมาชิก
Mozambique	25,203,395	0.9797
Democratic People Republic of Korea	24,763,188	0.9763
Ghana	25,366,462	0.9633
Republic of Yemen	23,852,409	0.8852
Australia	22,683,600	0.7684
Nepal	27,474,377	0.7526
Syrian Arab Republic	22,399,254	0.7399
Madagascar	22,293,914	0.7294
Saudi Arabia	28,287,855	0.6712
Cameroon	21,699,631	0.6700

จากตารางที่ 4.6สามารถสรุปได้ว่า ประเทศที่มีจำนวนประชากรใกล้เคียงกับ 25,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 15,000,000 คน แต่ไม่ถึง 35,000,000 คน10 อันดับแรก ได้แก่ประเทศ Mozambique, Democratic People Republic of Korea, Ghana, Republic of Yemen, Australia, Nepal, Syrian Arab Republic, Madagascar, Saudi Arabia และประเทศ Cameroon ซึ่งมีค่าความเป็นสมาชิก 0.9797, 0.9763, 0.9633, 0.8852, 0.7684, 0.7526, 0.7399, 0.7294, 0.6712และ 0.6700 ตามลำดับ

จากการทดลองหาคำตอบการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Triangular Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ ด้วยคำถาม "หาประเทศที่มี

จำนวนประชากรใกล้เคียงกับ 2,100,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 1,000,000 คน แต่ไม่ถึง 5,000,000 คน" ได้ผลการทดลองดังตารางที่ 4.7

ตารางที่ 4.7แสดง 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,100,000 คน ด้วยฟังก์ชัน Triangular Function

ชื่อประเทศ	จำนวนประชากร	ค่าความเป็นสมาชิก
Macedonia FYR	2,105,575	0.9981
Slovenia	2,058,152	0.9620
Lesotho	2,051,545	0.9560
Qatar	2,050,514	0.9550
Pacific island small states	2,252,782	0.9473
Namibia	2,259,393	0.9450
Latvia	2,025,473	0.9323
Botswana	2,003,910	0.9127
Jamaica	2,712,100	0.7889
Mongolia	2,796,484	0.7598

จากตารางที่ 4.7สามารถสรุปได้ว่า ประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,100,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 1,000,000 คน แต่ไม่ถึง 5,000,000 คน10 อันดับแรก ได้แก่ประเทศ Macedonia FYR, Slovenia, Lesotho, Qatar, Pacific island small states, Namibia, Latvia, Botswana, Jamaica และประเทศ Mongolia ซึ่งมีค่าความเป็นสมาชิก 0.9981, 0.9620, 0.9560, 0.9550, 0.9473, 0.9450, 0.9323, 0.9127, 0.7889 และ 0. 7598 ตามลำดับ

4.4.3 ผลการทดลองจากการตอบคำถามการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Left Shoulder Function

จากการทดลองหาคำตอบการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Left Shoulder Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับสูง ด้วยคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 90,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 65,000,000 คน แต่ไม่เกิน 90,000,000 คน" ได้ผลการทดลองดังตารางที่ 4.8

ตารางที่ 4.8แสดง 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับ 90,000,000 คน ด้วยฟังก์ชัน Left Shoulder Function

ชื่อประเทศ	จำนวนประชากร	ค่าความเป็นสมาชิก
Vietnam	88,775,500	0.9510
Germany	81,889,839	0.6756
Arab Republic of Egypt	80,721,874	0.6289
Islamic Republic of Iran	76,424,443	0.4570
Turkey	73,997,128	0.3599
Thailand	66,785,001	0.0714
Democratic Republic of the Congo	65,705,093	0.0282
France	65,696,689	0.0279

จากตารางที่ 4.8สามารถสรุปได้ว่า ประเทศที่มีจำนวนประชากรใกล้เคียงกับ 90,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 65,000,000 คน แต่ไม่ถึง 200,000,000 คน 10 อันดับแรก ได้แก่ประเทศ Vietnam, Germany, Arab Republic of Egypt, Islamic Republic of Iran, Turkey, Thailand, Democratic , Republic of the Congo และประเทศ France ซึ่งมีค่าความเป็นสมาชิก 0.9510, 0.6756, 0.6289, 0.4570, 0.3599, 0.0714, 0.0282และ 0.0279 ตามลำดับ

จากการทดลองหาคำตอบการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Left Shoulder Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง ด้วยคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 25,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 15,000,000 คน แต่ไม่เกิน 25,000,000 คน" ได้ผลการทดลองดังตารางที่ 4.9

ตารางที่ 4.9แสดง 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับ 25,000,000 คน ด้วยฟังก์ชัน Left Shoulder Function

ชื่อประเทศ	จำนวนประชากร	ค่าความเป็นสมาชิก
Democratic People Republic of Korea	24,763,188	0.9763
Republic of Yemen	23,852,409	0.8852
Australia	22,683,600	0.7684
Syrian Arab Republic	22,399,254	0.7399
Madagascar	22,293,914	0.7294
Cameroon	21,699,631	0.6700
Romania	21,326,905	0.6327
Angola	20,820,525	0.5821
Sri Lanka	20,328,000	0.5328
Other small states	19,919,957	0.4920

จากตารางที่ 4.9สามารถสรุปได้ว่า ประเทศที่มีจำนวนประชากรใกล้เคียงกับ 25,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 15,000,000 คน แต่ไม่เกิน 25,000,000 คน10 อันดับแรก ได้แก่ประเทศ Democratic People Republic of Korea, Republic of Yemen, Australia, Syrian Arab Republic, Madagascar, Cameroon, Romania, Angola, Sri Lanka และประเทศ Other small states ซึ่งมีค่าความเป็นสมาชิก 0.9763, 0.8852, 0.7684, 0.7399, 0.7294, 0.6700, 0.6327, 0.5821, 0.5328 และ 0.4920 ตามลำดับ

จากการทดลองหาคำตอบการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Left Shoulder Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ ด้วยคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,100,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 1,000,000 คน แต่ไม่ถึง 2,100,000 คน" ได้ผลการทดลองดังตารางที่ 4.10

ตารางที่ 4.10แสดง 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,100,000 คน ด้วยฟังก์ชัน Left Shoulder Function

ชื่อประเทศ	จำนวนประชากร	ค่าความเป็นสมาชิก
Slovenia	2,058,152	0.9620
Lesotho	2,051,545	0.9560
Qatar	2,050,514	0.9550
Latvia	2,025,473	0.9323
Botswana	2,003,910	0.9127
Kosovo	1,806,366	0.7331
Gambia	1,791,225	0.7193
Guinea-Bissau	1,663,558	0.6032
Gabon	1,632,572	0.5751
Estonia	1,339,396	0.3085

จากตารางที่ 4.10สามารถสรุปได้ว่า ประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,100,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 1,000,000 คน แต่ไม่ถึง 2,100,000 คน10 อันดับแรก ได้แก่ประเทศ Slovenia, Lesotho, Qatar, Latvia, Botswana, Kosovo, Gambia, Guinea-Bissau, Gabonและประเทศ Estonia ซึ่งมีค่าความเป็นสมาชิก 0.9620, 0.9560, 0.9550, 0.9323, 0.9127, 0.7331, 0.7193, 0.6032, 0.5751และ 0.3085 ตามลำดับ

มหาวิทยาลัยเทคโนโลยีสุรนารี

4.4.4 ผลการทดลองจากการตอบคำถามการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Right Shoulder Function

จากการทดลองหาคำตอบการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Right Shoulder Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับสูง ด้วยคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 90,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 90,000,000 คน แต่ไม่ถึง 200,000,000 คน" ได้ผลการทดลองดังตารางที่ 4.11

ตารางที่ 4.11แสดง 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับ 90,000,000 คน ด้วยฟังก์ชัน Right Shoulder Function

ชื่อประเทศ	จำนวนประชากร	ค่าความเป็นสมาชิก
Ethiopia	91,728,849	0.9843
Philippines	96,706,764	0.9390
Mexico	120,847,477	0.7196
Japan	127,561,489	0.6585
Russian Federation	143,533,000	0.5133
Bangladesh	154,695,368	0.4119
Nigeria	168,833,776	0.2833
Pakistan	179,160,111	0.1895
Brazil	198,656,019	0.0122

จากตารางที่ 4.11สามารถสรุปได้ว่า ประเทศที่มีจำนวนประชากรใกล้เคียงกับ 90,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 90,000,000 คน แต่ไม่ถึง 200,000,000 คน10 อันดับแรก ได้แก่ประเทศ Ethiopia, Philippines, Mexico, Japan, Russian Federation, Bangladesh, Nigeria, Pakistan และประเทศ Brazil ซึ่งมีค่าความเป็นสมาชิก 0.9843, 0.9390, 0.7196, 0.6585, 0.5133, 0.4119, 0.2833, 0.1895และ 0.0122 ตามลำดับ

จากการทดลองหาคำตอบการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Right Shoulder Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง ด้วยคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 25,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 25,000,000 คน แต่ไม่เกิน 35,000,000 คน" ได้ผลการทดลองดังตารางที่ 4.12

ตารางที่ 4.12แสดง 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับ 25,000,000 คนด้วยฟังก์ชัน Right Shoulder Function

ชื่อประเทศ	จำนวนประชากร	ค่าความเป็นสมาชิก
Mozambique	25,203,395	0.9797
Ghana	25,366,462	0.9634
Nepal	27,474,377	0.7526
Saudi Arabia	28,287,855	0.6712
Small states	29,141,492	0.5859
Malaysia	29,239,927	0.5760
Uzbekistan	29,776,850	0.5223
Afghanistan	29,824,536	0.5176
Venezuela	29,954,782	0.5045
Peru	29,987,800	0.5012

จากตารางที่ 4.12สามารถสรุปได้ว่า ประเทศที่มีจำนวนประชากรใกล้เคียงกับ 25,000,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 25,000,000 คน แต่ไม่เกิน 35,000,000 คน10 อันดับแรก ได้แก่ประเทศ Mozambique, Ghana, Nepal, Saudi Arabia, Small states, Malaysia, Uzbekistan, Afghanistan, Venezuela และประเทศ Peru ซึ่งมีค่าความเป็นสมาชิก 0.9797, 0.9634, 0.7526, 0.6712, 0.5859, 0.5760, 0.5223, 0.5176, 0.5045และ 0.5012 ตามลำดับ

จากการทดลองหาคำตอบการจัดเรียง 10 อันดับแรกด้วยฟังก์ชันสมาชิก Right Shoulder Function ในกลุ่มของประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ ด้วยคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,100,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 2,100,000 คน แต่ไม่ถึง 5,000,000 คน" ได้ผลการทดลองดังตารางที่ 4.13

ตารางที่ 4.13แสดง 10 อันดับแรกของกลุ่มประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,100,000 คน ด้วยฟังก์ชัน Right Shoulder Function

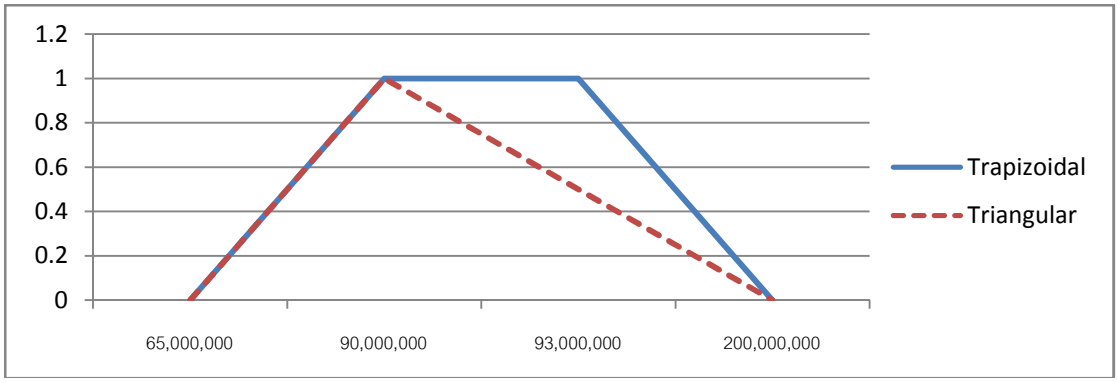
ชื่อประเทศ	จำนวนประชากร	ค่าความเป็นสมาชิก
Macedonia FYR	2,105,575	0.9981
Pacific island small states	2,252,782	0.9473
Namibia	2,259,393	0.9450
Jamaica	2,712,100	0.7889
Mongolia	2,796,484	0.7598
Armenia	2,969,081	0.7003
Lithuania	2,985,509	0.6947
Albania	3,162,083	0.6338
Kuwait	3,250,496	0.6033
Oman	3,314,001	0.5814

จากตารางที่ 4.13สามารถสรุปได้ว่า ประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,100,000 คน โดยประเทศที่จะนำมาพิจารณาจะต้องมีจำนวนประชากรมากกว่า 2,100,000 คน แต่ไม่ถึง 5,000,000 คน10 อันดับแรก ได้แก่ประเทศ Macedonia FYR, Pacific island small states, Namibia, Jamaica, Mongolia, Armenia, Lithuania, Albania, Kuwait และประเทศ Oman ซึ่งมีค่าความเป็นสมาชิก 0.9981, 0.9473, 0.9450, 0.7889, 0.7598, 0.7003, 0.6947, 0.6338, 0.6033และ 0.5814 ตามลำดับ

4.5 เปรียบเทียบผลการทดลองด้วยฟังก์ชันสมาชิกแต่ละฟังก์ชัน

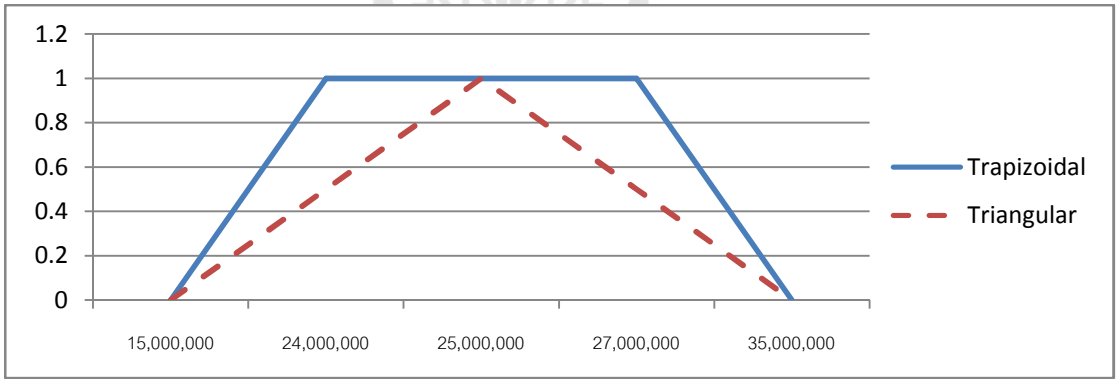
4.5.1 เปรียบเทียบการใช้งานระหว่างฟังก์ชัน Trapezoidal Function เปรียบเทียบกับ Triangular Function

ในกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับสูง เมื่อนำฟังก์ชัน Trapezoidal Function และฟังก์ชัน Triangular Function ไปใช้ในการจัดอันดับ ผลการใช้งานของฟังก์ชันทั้งสองแสดงได้ดังรูปที่ 4.28



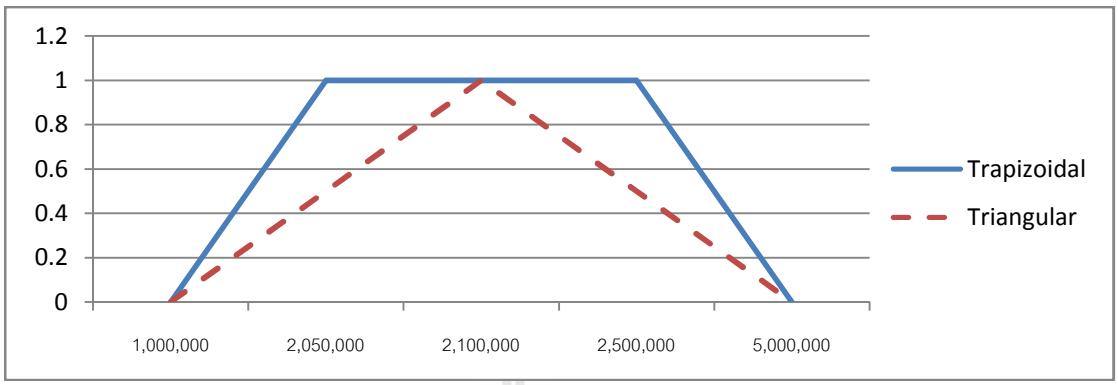
รูปที่ 4.28กราฟเปรียบเทียบค่าความเป็นสมาชิกระหว่างการใช้ฟังก์ชัน Trapezoidal Function กับ Triangular Function ในคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 90,000,000 คน"

ในกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง เมื่อนำฟังก์ชัน Trapezoidal Function และฟังก์ชัน Triangular Function ไปใช้ในการจัดอันดับ ผลการใช้งานของฟังก์ชันทั้งสองแสดงได้ดังรูปที่ 4.29



รูปที่ 4.29กราฟเปรียบเทียบค่าความเป็นสมาชิกระหว่างการใช้ฟังก์ชัน Trapezoidal Function กับ Triangular Function ในคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 25,000,000 คน"

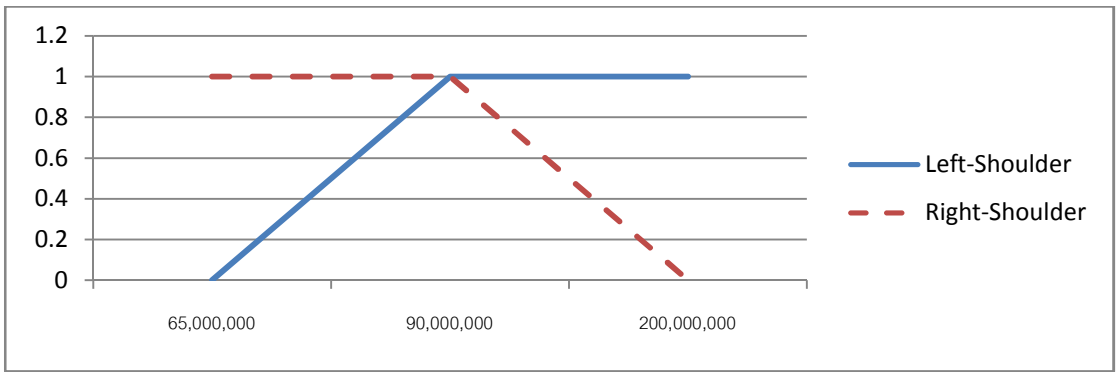
ในกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ เมื่อนำฟังก์ชัน Trapezoidal Function และฟังก์ชัน Triangular Function ไปใช้ในการจัดอันดับ ผลการใช้งานของฟังก์ชันทั้งสอง แสดงได้ดังรูปที่ 4.30



รูปที่ 4.30กราฟเปรียบเทียบค่าความเป็นสมาชิกระหว่างการใช้ฟังก์ชัน Trapezoidal Function กับ Triangular Function ในคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,100,000 คน"

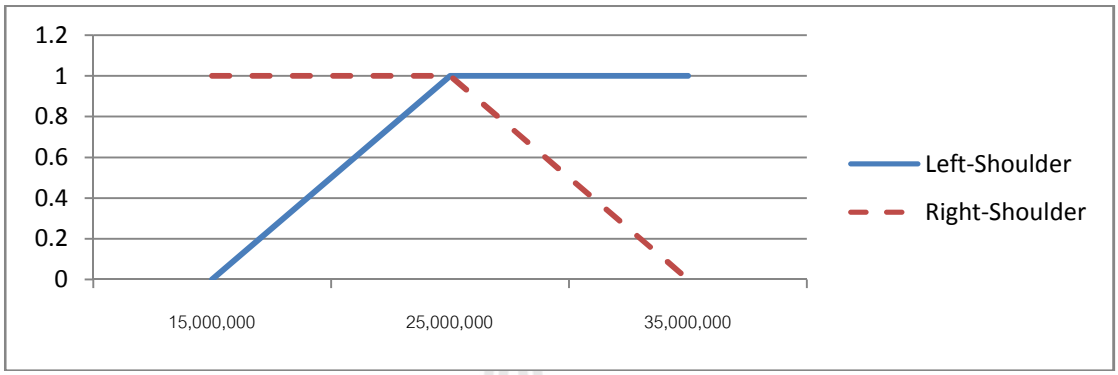
4.5.2 เปรียบเทียบการใช้งานระหว่างฟังก์ชัน Left Shoulder Function เปรียบเทียบกับ Right Shoulder Function

ในกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับสูง เมื่อนำฟังก์ชัน Left Shoulder Function และฟังก์ชัน Right Shoulder Function ไปใช้ในการจัดอันดับ ผลการใช้งานของฟังก์ชันทั้งสอง แสดงได้ดังรูปที่ 4.31



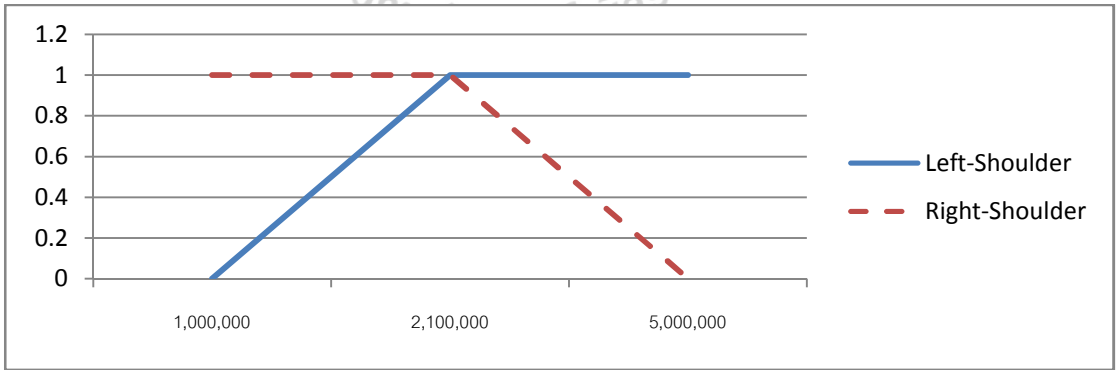
รูปที่ 4.31กราฟเปรียบเทียบค่าความเป็นสมาชิกระหว่างการใช้ฟังก์ชัน Left Shoulder Function กับ Right Shoulder Function ในคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 90,000,000 คน"

ในกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับปานกลาง เมื่อนำฟังก์ชัน Left Shoulder Function และฟังก์ชัน Right Shoulder Function ไปใช้ในการจัดอันดับ ผลการใช้งานของฟังก์ชันทั้งสอง แสดงได้ดังรูปที่ 4.32



รูปที่ 4.32กราฟเปรียบเทียบค่าความเป็นสมาชิกระหว่างการใช้ฟังก์ชัน Left Shoulder Function กับ Right Shoulder Function ในคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 25,000,000 คน"

ในกลุ่มประเทศที่มีจำนวนประชากรอยู่ในระดับต่ำ เมื่อนำฟังก์ชัน Left Shoulder Function และฟังก์ชัน Right Shoulder Function ไปใช้ในการจัดอันดับ ผลการใช้งานของฟังก์ชันทั้งสอง แสดงได้ดังรูปที่ 4.33



รูปที่ 4.33กราฟเปรียบเทียบค่าความเป็นสมาชิกระหว่างการใช้ฟังก์ชัน Left Shoulder Function กับ Right Shoulder Function ในคำถาม "หาประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,100,000 คน"

4.6 อภิปรายผล

จากผลการทดลองการตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัย ด้วยเทคนิคการใช้ฟังก์ชันสมาชิก 4 ฟังก์ชัน ได้แก่ Trapezoidal Function, Triangular Function, Left Shoulder Function และ Right Shoulder Function สามารถสรุปผลการเปรียบเทียบได้ดังนี้

1) การเปรียบเทียบระหว่างการใช้งานฟังก์ชันสมาชิก Trapezoidal Function เปรียบเทียบกับ Triangular Function จะเห็นได้ว่าทั้งสองฟังก์ชันสามารถคำนวณหา เค อันดับแรกภายใต้เงื่อนไขและขอบเขตที่กำหนด โดยทั้งสองฟังก์ชันสามารถนำไปใช้งานแทนกันได้ แต่จะมีความแตกต่างที่เห็นได้ชัดเจนคือค่าความเป็นสมาชิก และอันดับของแต่ละประเทศ เนื่องจากค่าความเป็นสมาชิกจะขึ้นอยู่กับเงื่อนไขในการคำนวณที่ผู้ใช้กำหนดเข้าไปจึงส่งผลให้ค่าความเป็นสมาชิกของทั้งสองฟังก์ชันแตกต่างกัน และเมื่อพิจารณาจากรูปที่ 4.28, 4.29และ 4.30จะเห็นได้ว่า ฟังก์ชันสมาชิก Trapezoidal Function จะเหมาะสำหรับการพิจารณาเงื่อนไขที่มากกว่า 1 เงื่อนไข (กล่าวคือเหมาะสำหรับนำไปใช้กับข้อมูลที่มีค่าที่สนใจมากกว่า 1 ค่า) ส่วนฟังก์ชัน Triangular Function เหมาะสำหรับการพิจารณาเงื่อนไขเพียงเงื่อนไขเดียว (กล่าวคือเหมาะสำหรับนำไปใช้กับข้อมูลที่มีค่าที่สนใจเพียงค่าเดียว)

2) การเปรียบเทียบระหว่างการใช้งานฟังก์ชันสมาชิก Left Shoulder Function เปรียบเทียบกับ Right Shoulder Function จากรูปที่ 4.31,4.32และ 4.33จะเห็นว่าทั้งสองฟังก์ชันมีความแตกต่างกันอย่างสิ้นเชิง โดยฟังก์ชันสมาชิก Left Shoulder Function เหมาะสำหรับการพิจารณาเงื่อนไขที่น้อยกว่าค่าที่เราสนใจ (กล่าวคือเหมาะสำหรับการหาค่าน้อยกว่าค่าที่เราสนใจ) ส่วนฟังก์ชันสมาชิก Right Shoulder Function เหมาะสำหรับการพิจารณาเงื่อนไขที่มากกว่าค่าที่เราสนใจ (กล่าวคือเหมาะสำหรับการหาค่าที่มากกว่าค่าที่เราสนใจ)

จากผลการทดลองการตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรกในฐานข้อมูลนิรนัย โดยการนำเทคนิคการใช้ฟังก์ชันสมาชิกเข้ามาช่วยในการจัดอันดับ พบว่าฟังก์ชันสมาชิก Trapezoidal Function และ Triangular Function จะให้ผลการจัดเรียง เค อันดับแรกมีอันดับที่แตกต่างกันบ้างเนื่องจากเงื่อนไขในการคำนวณค่าความเป็นสมาชิกที่แตกต่างกัน และการกำหนดค่าที่สนใจไม่เหมือนกัน จึงทำให้ผลที่ได้แตกต่างกัน ถึงแม้ เค อันดับแรกของทั้งสองฟังก์ชันจะแตกต่างกันบ้าง แต่ผลที่ได้มีความสอดคล้องกัน โดยทั้งสองฟังก์ชันสามารถคำนวณหา เค อันดับแรกภายใต้เงื่อนไขที่กำหนดได้ และฟังก์ชันสมาชิก Trapezoidal Function จะเหมาะสำหรับนำไปใช้กับข้อมูลที่มีค่าที่สนใจมากกว่า 1 ค่า ส่วนฟังก์ชัน Triangular Function เหมาะสำหรับนำไปใช้กับข้อมูลที่มีค่าที่สนใจเพียงค่าเดียว ส่วนฟังก์ชัน Left Shoulder Function เหมาะสำหรับ

การนำไปหา เค อันดับแรกของข้อมูลที่มีค่าน้อยกว่าค่าที่สนใจ และฟังก์ชัน Right Shoulder Function เหมาะสำหรับการนำไปหา เค อันดับแรกของข้อมูลที่มีค่ามากกว่าค่าที่สนใจ

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

ปัจจุบันการตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรกแบบคลุมเครือสำหรับฐานข้อมูลนิรนัยนั้นมีหลากหลายเทคนิค ซึ่งเทคนิคหนึ่งที่ถูกนำมาใช้คือการคำนวณหาค่าคะแนนด้วยเทคนิค Scoring Function แต่การคำนวณหาค่าคะแนนด้วยเทคนิค Scoring Function นั้นยังคงมีปัญหาในการพิจารณาข้อมูลเฉพาะบางช่วงที่เราสนใจ นั่นก็คือ Scoring Function จะนำข้อมูลทั้งหมดมาทำการคำนวณหาค่าค่าคะแนนเพื่อดูว่าข้อมูลในแถวใดบ้างมีค่าคะแนนสูงที่สุด (ข้อมูลมีความใกล้เคียงกับสิ่งที่เราสนใจมากที่สุด) เพื่อนำมาจัดเรียง เค อันดับแรก แต่เมื่อต้องการพิจารณาข้อมูลเฉพาะบางส่วนจึงต้องมีการนำเทคนิคอื่นมาใช้แทนการคำนวณค่า Scoring Function ซึ่งเทคนิคที่เหมาะสมจะนำมาใช้แทนการคำนวณค่า Scoring Function คือการใช้เทคนิคการหาค่าความเป็นสมาชิกด้วย Membership Function

ในงานวิจัยนี้จึงมุ่งเน้นพัฒนากระบวนการในการออกแบบฟังก์ชันสำหรับคำนวณค่าความเป็นสมาชิกและพัฒนาโปรแกรม เพื่อหา เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัยตามเงื่อนไขที่ผู้ใช้กำหนด ซึ่งจะทำให้ผู้ใช้ได้รับคำตอบเป็นข้อมูล เค อันดับแรกที่ใกล้เคียงกับสิ่งที่ผู้ใช้ให้ความสนใจ กระบวนการที่ใช้ในการตอบคำถาม เค อันดับแรกของงานวิจัยนี้ คือการคำนวณค่าความเป็นสมาชิกด้วยฟังก์ชันสมาชิก 4 ฟังก์ชัน ได้แก่ Trapezoidal Function, Triangular Function, Left Shoulder Function และ Right Shoulder Function

5.1 ขั้นตอนการดำเนินงานวิจัย

1. ศึกษาการทำงานของอัลกอริทึมเคมีนส์(k-means) และศึกษาการเขียนโปรแกรมด้วยภาษาดาต้าล็อกซึ่งในงานวิจัยนี้ได้พัฒนาโปรแกรมโดยใช้รูปแบบโปรแกรมดีอีเอส (Datalog Educational System) ซึ่งเป็นโปรแกรมที่เหมาะกับการหาคำตอบ เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัยโดยสามารถกำหนดเงื่อนไขในการคำนวณหาค่าความเป็นสมาชิกเข้าไปในรูปแบบของกฎสำหรับการหาความรู้ใหม่จากฐานข้อมูลที่มีอยู่แล้ว

2. ออกแบบฟังก์ชันสำหรับคำนวณค่าความเป็นสมาชิก โดยการนำข้อมูลที่ผ่านการแบ่งกลุ่มด้วยอัลกอริทึม k-meansมาวิเคราะห์ดูขอบเขตของค่าสูงสุดและต่ำสุดของแต่ละกลุ่มของข้อมูลโดยการสร้างฟังก์ชันสมาชิกแต่ละฟังก์ชัน ในงานวิจัยนี้ได้กำหนดขอบเขตของจำนวนประชากรโลกในกลุ่มย่อยด้วยข้อกำหนดว่า ขอบเขตของจำนวนประชากรโลกในแต่ละกลุ่มจะต้องไม่อยู่นอกเหนือจากค่าต่ำสุดและค่าสูงสุดของแต่ละกลุ่ม ทั้งนี้เพื่อลดความซ้ำซ้อนของคำตอบ

3. พัฒนาโปรแกรมจัดเรียง เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัยด้วยเทคนิคการใช้ฟังก์ชันสมาชิก โปรแกรมจัดเรียง เค อันดับแรกด้วยเทคนิคการใช้ฟังก์ชันสมาชิกเป็นโปรแกรมที่มีการทำงานคล้ายกับโปรแกรมจัดเรียง เค อันดับแรกด้วยเทคนิคคำนวณค่าคะแนนแต่จะมีความสามารถที่เพิ่มขึ้น คือสามารถนำไปใช้พิจารณาข้อมูลเฉพาะช่วงที่สนใจได้

การเปรียบเทียบประสิทธิภาพของการจัดเรียง เค อันดับแรกแบบคลุมเครือสำหรับฐานข้อมูลนิรนัย จะใช้ข้อมูลจำนวนประชากรโลก (Population)ซึ่งเป็นชุดข้อมูลมาตรฐานของธนาคารโลก โดยจะเปรียบเทียบการตอบคำถามหา เค อันดับแรกด้วยฟังก์ชันสมาชิกแต่ละฟังก์ชันและเปรียบเทียบว่าแต่ละฟังก์ชันสมาชิกเหมาะที่จะนำไปใช้งานในลักษณะใด

5.2 สรุปผลการวิจัย

จากผลการทดลองการตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรกในฐานข้อมูลนิรนัย โดยการนำเทคนิคการใช้ฟังก์ชันสมาชิกเข้ามาช่วยในการจัดอันดับ พบว่ากลไกการทำงานของระบบฐานข้อมูลนิรนัยสามารถจัดอันดับข้อมูลด้วยการใช้เทคนิคการคำนวณค่าความเป็นสมาชิกด้วยฟังก์ชันสมาชิกแต่ละประเภทมาใช้ในการจัดอันดับเฉพาะช่วงข้อมูลที่สนใจ โดยหากใช้เทคนิคอื่น ๆ จะไม่สามารถจัดอันดับเฉพาะช่วงข้อมูลที่เราสนใจได้ และใช้เทคนิค Clustering มาช่วยกำหนดค่าพารามิเตอร์ของแต่ละกลุ่ม พบว่าฟังก์ชันสมาชิก Trapezoidal Function และ Triangular Function จะให้ผลการจัดเรียง เค อันดับแรกมีอันดับที่แตกต่างกันบ้างเนื่องจากเงื่อนไขในการคำนวณค่าความเป็นสมาชิกที่แตกต่างกัน และการกำหนดช่วงของค่าที่สนใจไม่เหมือนกัน จึงทำให้ผลที่ได้แตกต่างกัน ถึงแม้ เค อันดับแรกของทั้งสองฟังก์ชันจะแตกต่างกันบ้าง แต่ผลที่ได้มีความสอดคล้องกัน โดยทั้งสองฟังก์ชันสามารถคำนวณหา เค อันดับแรกภายใต้เงื่อนไขที่กำหนดได้ และฟังก์ชันสมาชิก Trapezoidal Function จะเหมาะสำหรับนำไปใช้กับข้อมูลที่มีค่าที่สนใจมากกว่า 1 ค่า ส่วนฟังก์ชัน Triangular Function เหมาะสำหรับนำไปใช้กับข้อมูลที่มีค่าที่สนใจเพียงค่าเดียว ส่วนฟังก์ชัน Left Shoulder Function เหมาะสำหรับการนำไปหา เค อันดับแรกของข้อมูลที่มีค่าน้อยกว่าค่าที่สนใจ และฟังก์ชัน Right Shoulder Function เหมาะสำหรับการนำไปหา เค อันดับแรกของข้อมูลที่มีค่ามากกว่าค่าที่สนใจ

5.3 ปัญหาและข้อเสนอแนะ

ขั้นตอนการกำหนดขอบเขตและเงื่อนไขของแต่ละฟังก์ชัน ยังไม่สามารถกำหนดขอบเขตและเงื่อนไขในการสอบถามได้ ต้องกำหนดเข้าไปในฐานความรู้ก่อนจึงสามารถหา เค อันดับแรกตามขอบเขตและเงื่อนไขที่กำหนดได้และขั้นตอนของการจัดกลุ่มยังไม่สามารถกำหนดได้แน่นอนว่าควรจะแบ่งออกเป็นกี่ Cluster

ในอนาคตหากสามารถกำหนดขอบเขตและเงื่อนไขที่กำหนดในการสอบถามได้ โดยไม่ต้องกำหนดเข้าไปในฐานความรู้ จะสามารถช่วยให้ผู้ใช้งานมีความสะดวกมากยิ่งขึ้น และสามารถเปลี่ยนขอบเขตและเงื่อนไขได้ง่ายยิ่งขึ้น และสำหรับขั้นตอนของการจัดกลุ่ม ควรใช้วิธีการจัดกลุ่มข้อมูลที่สามารถพิจารณาค่า k ได้โดยอัตโนมัติ และการพิจารณาเลือกใช้ฟังก์ชันอาจจะเพิ่มการพิจารณามากกว่า 1 ฟังก์ชัน เช่นในการสอบถามข้อมูล ผู้ใช้อาจระบุสองฟังก์ชัน และเชื่อมโยงฟังก์ชันด้วยโอเปอเรเตอร์ AND, OR

มหาวิทยาลัยเทคโนโลยีสุรนารี

รายการอ้างอิง

นิตยา เกิดประสพ(2553). การค้นหาความรู้และการขุดค้นข้อมูล. **เอกสารประกอบการสอนวิชา 423304 Knowledge Discovery and Data Mining**. มหาวิทยาลัยเทคโนโลยีสุรนารี

A. Achs (2007). **From fuzzy- to bipolar- datalog**, Proceedings of the 5th EUSFLAT Conference, Ostrava, Czech Republic, pp. 221-227.

F. Bobillo and U. Straccia (2008). **fuzzyDL: an expressive fuzzy description logic reasoner**, Proceedings of the IEEE International Conference on Fuzzy System, Hong Kong, pp. 923-930.

J. Dombi (1990).**Membership function as an evaluation, Fuzzy Sets and Systems**, Vol. 35, No. 1, pp. 1-21

J. Han and M. Kamber (2000). **Data Mining: Concepts and Techniques.** Morgan Kaufmann Publishers.

N. Kerdprasop(2013). **Data Mining With Datalog**, Retrieved June 3, 2013, form http://sites.google.com/site/nittayak/home/clp

U.Straccia (2006a).**Towards top-kquery answering in deductive databases**, Proceedings of IEEE International Conference of Systems, Man and Cybernetics, Taipei, Taiwan, pp.4873–4879.

U.Straccia (2006b).**Towards top-k query answering in description logics: the case of DL-Lite**, Proceedings of the 10 th European Conference on Logics in Artificial Intelligence, Liverpool, UK, pp. 439–451.

U. Straccia (2006c).**Answering vague queries in fuzzy DL-Lite**, Proceedings of the 11th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, Paris, France, pp. 2238–2245.

U. Straccia and N. Madrid (2012).**A top-k query answering procedure for fuzzy logic programming**, Fuzzy Sets and Systems, vol.205, pp.1-29.

L. Squier (2001). **What is Data Mining? [Online]**.Available

URL: http://www.damancr.org/Library/2001 .11.14 -Laura% 20Squier.ppt.

Wikipedia, The Free Encyclopedia (2013a). **Cluster analysis [Online].**Available

URL: http://en.wikipedia.org/wiki/Cluster_analysis

Wikipedia, The Free Encyclopedia (2013b). **K-means clustering [Online].**Available

URL: http://en.wikipedia.org/wiki/K-means_clustering

Wikipedia, The Free Encyclopedia(2013c).**Fuzzy set[Online]**.Available

URL: http://en.wikipedia.org/wiki/Fuzzy_logic

ภาคผนวก ก

รหัสต้นฉบับของโปรแกรม

มหาวิทยาลัยเทคโนโลยีสุรนารี

โปรแกรมตอบคำถาม เค อันดับแรกในฐานข้อมูลนิรนัย

```
#---------Create table----------#
:-type(popCode(name:string,code:string)).
:-type(popTotal(code:string,amount:int)).

#---------Insert data into table----------#
popCode('Afghanistan','AFG').
popCode('Albania','ALB').
popCode('Algeria','DZA').
popCode('American Samoa','ASM').
popCode('Andorra','ADO').
popCode('Angola','AGO').
popCode('Antigua and Barbuda','ATG').
popCode('Arab World','ARB').
popCode('Argentina','ARG').
popCode('Armenia','ARM').
popCode('Aruba','ABW').
popCode('Australia','AUS').
popCode('Austria','AUT').
popCode('Azerbaijan','AZE').
popCode('Bahamas','BHS').
popCode('Bahrain','BHR').
popCode('Bangladesh','BGD').
popCode('Barbados','BRB').
popCode('Belarus','BLR').
popCode('Belgium','BEL').
popCode('Belize','BLZ').
popCode('Benin','BEN').
popCode('Bermuda','BMU').
popCode('Bhutan','BTN').
popCode('Bolivia','BOL').
popCode('Bosnia and Herzegovina','BIH').
popCode('Botswana','BWA').
popCode('Brazil','BRA').
popCode('Brunei Darussalam','BRN').
popCode('Bulgaria','BGR').
popCode('Burkina Faso','BFA').
popCode('Burundi','BDI').
popCode('Cambodia','KHM').
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
popCode('Cameroon','CMR').
popCode('Canada','CAN').
popCode('Cape Verde','CPV').
popCode('Caribbean small states','CSS').
popCode('Cayman Islands','CYM').
popCode('Central African Republic','CAF').
popCode('Chad','TCD').
popCode('Channel Islands','CHI').
popCode('Chile','CHL').
popCode('China','CHN').
popCode('Colombia','COL').
popCode('Comoros','COM').
popCode('Democratic Republic of the Congo','ZAR').
popCode('Republic of the Congo','COG').
popCode('Costa Rica','CRI').
popCode('Ivoire Cote','CIV').
popCode('Croatia','HRV').
popCode('Cuba','CUB').
popCode('Curacao','CUW').
popCode('Cyprus','CYP').
popCode('Czech Republic','CZE').
popCode('Denmark','DNK').
popCode('Djibouti','DJI').
popCode('Dominica','DMA').
popCode('Dominican Republic','DOM').
popCode('East Asia & Pacific (all income levels)','EAS').
popCode('East Asia & Pacific (developing only)','EAP').
popCode('Ecuador','ECU').
popCode('Arab Republic of Egypt','EGY').
popCode('El Salvador','SLV').
popCode('Equatorial Guinea','GNQ').
popCode('Eritrea','ERI').
popCode('Estonia','EST').
popCode('Ethiopia','ETH').
popCode('Euro area','EMU').
popCode('Europe & Central Asia (all income levels)','ECS').
popCode('Europe & Central Asia (developing only)','ECA').
popCode('European Union','EUU').
popCode('Faeroe Islands','FRO').
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
popCode('Fiji','FJI').
popCode('Finland','FIN').
popCode('France','FRA').
popCode('French Polynesia','PYF').
popCode('Gabon','GAB').
popCode('Gambia','GMB').
popCode('Georgia','GEO').
popCode('Germany','DEU').
popCode('Ghana','GHA').
popCode('Greece','GRC').
popCode('Greenland','GRL').
popCode('Grenada','GRD').
popCode('Guam','GUM').
popCode('Guatemala','GTM').
popCode('Guinea','GIN').
popCode('Guinea-Bissau','GNB').
popCode('Guyana','GUY').
popCode('Haiti','HTI').
popCode('Heavily indebted poor countries (HIPC)','HPC').
popCode('High income','HIC').
popCode('High income: nonOECD','NOC').
popCode('High income: OECD','OEC').
popCode('Honduras','HND').
popCode('Hong Kong SAR China','HKG').
popCode('Hungary','HUN').
popCode('Iceland','ISL').
popCode('India','IND').
popCode('Indonesia','IDN').
popCode('Islamic Republic of Iran','IRN').
popCode('Iraq','IRQ').
popCode('Ireland','IRL').
popCode('Isle of Man','IMY').
popCode('Israel','ISR').
popCode('Italy','ITA').
popCode('Jamaica','JAM').
popCode('Japan','JPN').
popCode('Jordan','JOR').
popCode('Kazakhstan','KAZ').
popCode('Kenya','KEN').
popCode('Kiribati','KIR').
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
popCode('Democratic People Republic of Korea','PRK').
popCode('Republic of Korea','KOR').
popCode('Kosovo','KSV').
popCode('Kuwait','KWT').
popCode('Kyrgyz Republic','KGZ').
popCode('Lao PDR','LAO').
popCode('Latin America & Caribbean (all income levels)','LCN').
popCode('Latin America & Caribbean (developing only)','LAC').
popCode('Latvia','LVA').
popCode('Least developed countries: UN classification','LDC').
popCode('Lebanon','LBN').
popCode('Lesotho','LSO').
popCode('Liberia','LBR').
popCode('Libya','LBY').
popCode('Liechtenstein','LIE').
popCode('Lithuania','LTU').
popCode('Low & middle income','LMY').
popCode('Low income','LIC').
popCode('Lower middle income','LMC').
popCode('Luxembourg','LUX').
popCode('Macao SAR China','MAC').
popCode('Macedonia FYR','MKD').
popCode('Madagascar','MDG').
popCode('Malawi','MWI').
popCode('Malaysia','MYS').
popCode('Maldives','MDV').
popCode('Mali','MLI').
popCode('Malta','MLT').
popCode('Marshall Islands','MHL').
popCode('Mauritania','MRT').
popCode('Mauritius','MUS').
popCode('Mexico','MEX').
popCode('Federated States of Micronesia','FSM').
popCode('Middle East & North Africa (all income levels)','MEA').
popCode('Middle East & North Africa (developing only)','MNA').
popCode('Middle income','MIC').
popCode('Moldova','MDA').
popCode('Monaco','MCO').
popCode('Mongolia','MNG').
popCode('Montenegro','MNE').
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
popCode('Morocco','MAR').
popCode('Mozambique','MOZ').
popCode('Myanmar','MMR').
popCode('Namibia','NAM').
popCode('Nepal','NPL').
popCode('Netherlands','NLD').
popCode('New Caledonia','NCL').
popCode('New Zealand','NZL').
popCode('Nicaragua','NIC').
popCode('Niger','NER').
popCode('Nigeria','NGA').
popCode('North America','NAC').
popCode('Northern Mariana Islands','MNP').
popCode('Norway','NOR').
popCode('OECD members','OED').
popCode('Oman','OMN').
popCode('Other small states','OSS').
popCode('Pacific island small states','PSS').
popCode('Pakistan','PAK').
popCode('Palau','PLW').
popCode('Panama','PAN').
popCode('Papua New Guinea','PNG').
popCode('Paraguay','PRY').
popCode('Peru','PER').
popCode('Philippines','PHL').
popCode('Poland','POL').
popCode('Portugal','PRT').
popCode('Puerto Rico','PRI').
popCode('Qatar','QAT').
popCode('Romania','ROM').
popCode('Russian Federation','RUS').
popCode('Rwanda','RWA').
popCode('Samoa','WSM').
popCode('San Marino','SMR').
popCode('Sao Tome and Principe','STP').
popCode('Saudi Arabia','SAU').
popCode('Senegal','SEN').
popCode('Serbia','SRB').
popCode('Seychelles','SYC').
popCode('Sierra Leone','SLE').
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
popCode('Singapore','SGP').
popCode('Sint Maarten (Dutch part)','SXM').
popCode('Slovak Republic','SVK').
popCode('Slovenia','SVN').
popCode('Small states','SST').
popCode('Solomon Islands','SLB').
popCode('Somalia','SOM').
popCode('South Africa','ZAF').
popCode('South Asia','SAS').
popCode('South Sudan','SSD').
popCode('Spain','ESP').
popCode('Sri Lanka','LKA').
popCode('Saint Kitts and Nevis','KNA').
popCode('Saint Lucia','LCA').
popCode('Saint Martin (French part)','MAF').
popCode('Saint Vincent and the Grenadines','VCT').
popCode('Sub-Saharan Africa (all income levels)','SSF').
popCode('Sub-Saharan Africa (developing only)','SSA').
popCode('Sudan','SDN').
popCode('Suriname','SUR').
popCode('Swaziland','SWZ').
popCode('Sweden','SWE').
popCode('Switzerland','CHE').
popCode('Syrian Arab Republic','SYR').
popCode('Tajikistan','TJK').
popCode('Tanzania','TZA').
popCode('Thailand','THA').
popCode('Timor-Leste','TMP').
popCode('Togo','TGO').
popCode('Tonga','TON').
popCode('Trinidad and Tobago','TTO').
popCode('Tunisia','TUN').
popCode('Turkey','TUR').
popCode('Turkmenistan','TKM').
popCode('Turks and Caicos Islands','TCA').
popCode('Tuvalu','TUV').
popCode('Uganda','UGA').
popCode('Ukraine','UKR').
popCode('United Arab Emirates','ARE').
popCode('United Kingdom','GBR').
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
popCode('United States','USA').
popCode('Upper middle income','UMC').
popCode('Uruguay','URY').
popCode('Uzbekistan','UZB').
popCode('Vanuatu','VUT').
popCode('Venezuela','VEN').
popCode('Vietnam','VNM').
popCode('Virgin Islands','VIR').
popCode('West Bank and Gaza','WBG').
popCode('World','WLD').
popCode('Republic of Yemen','YEM').
popCode('Zambia','ZMB').
popCode('Zimbabwe','ZWE').

popTotal('AFG',29824536).
popTotal('ALB',3162083).
popTotal('DZA',38481705).
popTotal('ASM',55128).
popTotal('ADO',78360).
popTotal('AGO',20820525).
popTotal('ATG',89069).
popTotal('ARG',41086927).
popTotal('ARM',2969081).
popTotal('ABW',102384).
popTotal('AUS',22683600).
popTotal('AUT',8462446).
popTotal('AZE',9297507).
popTotal('BHS',371960).
popTotal('BHR',1317827).
popTotal('BGD',154695368).
popTotal('BRB',283221).
popTotal('BLR',9464000).
popTotal('BEL',11142157).
popTotal('BLZ',324060).
popTotal('BEN',10050702).
popTotal('BMU',64806).
popTotal('BTN',741822).
popTotal('BOL',10496285).
popTotal('BIH',3833916).
popTotal('BWA',2003910).
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
popTotal('BRA',198656019).
popTotal('BRN',412238).
popTotal('BGR',7304632).
popTotal('BFA',16460141).
popTotal('BDI',9849569).
popTotal('KHM',14864646).
popTotal('CMR',21699631).
popTotal('CAN',34880491).
popTotal('CPV',494401).
popTotal('CSS',6968753).
popTotal('CYM',57570).
popTotal('CAF',4525209).
popTotal('TCD',12448175).
popTotal('CHI',161235).
popTotal('CHL',17464814).
popTotal('CHN',1350695000).
popTotal('COL',47704427).
popTotal('COM',717503).
popTotal('ZAR',65705093).
popTotal('COG',4337051).
popTotal('CRI',4805295).
popTotal('CIV',19839750).
popTotal('HRV',4267000).
popTotal('CUB',11270957).
popTotal('CUW',151892).
popTotal('CYP',1128994).
popTotal('CZE',10514810).
popTotal('DNK',5590478).
popTotal('DJI',859652).
popTotal('DMA',71684).
popTotal('DOM',10276621).
popTotal('ECU',15492264).
popTotal('EGY',80721874).
popTotal('SLV',6297394).
popTotal('GNQ',736296).
popTotal('ERI',6130922).
popTotal('EST',1339396).
popTotal('ETH',91728849).
popTotal('FRO',49506).
popTotal('FJI',874742).
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
popTotal('FIN',5414293).
popTotal('FRA',65696689).
popTotal('PYF',273814).
popTotal('GAB',1632572).
popTotal('GMB',1791225).
popTotal('GEO',4511800).
popTotal('DEU',81889839).
popTotal('GHA',25366462).
popTotal('GRC',11280167).
popTotal('GRL',56840).
popTotal('GRD',105483).
popTotal('GUM',162810).
popTotal('GTM',15082831).
popTotal('GIN',11451273).
popTotal('GNB',1663558).
popTotal('GUY',795369).
popTotal('HTI',10173775).
popTotal('HND',7935846).
popTotal('HKG',7154600).
popTotal('HUN',9943755).
popTotal('ISL',320137).
popTotal('IND',1236686732).
popTotal('IDN',246864191).
popTotal('IRN',76424443).
popTotal('IRQ',32578209).
popTotal('IRL',4588798).
popTotal('IMY',85284).
popTotal('ISR',7907900).
popTotal('ITA',60917978).
popTotal('JAM',2712100).
popTotal('JPN',127561489).
popTotal('JOR',6318000).
popTotal('KAZ',16797459).
popTotal('KEN',43178141).
popTotal('KIR',100786).
popTotal('PRK',24763188).
popTotal('KOR',50004000).
popTotal('KSV',1806366).
popTotal('KWT',3250496).
popTotal('KGZ',5582100).
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
popTotal('LAO',6645827).
popTotal('LVA',2025473).
popTotal('LBN',4424888).
popTotal('LSO',2051545).
popTotal('LBR',4190435).
popTotal('LBY',6154623).
popTotal('LIE',36656).
popTotal('LTU',2985509).
popTotal('LUX',531441).
popTotal('MKD',2105575).
popTotal('MDG',22293914).
popTotal('MWI',15906483).
popTotal('MYS',29239927).
popTotal('MDV',338442).
popTotal('MLI',14853572).
popTotal('MLT',418366).
popTotal('MHL',52555).
popTotal('MRT',3796141).
popTotal('MUS',1291456).
popTotal('MEX',120847477).
popTotal('FSM',103395).
popTotal('MDA',3559541).
popTotal('MCO',37579).
popTotal('MNG',2796484).
popTotal('MNE',621081).
popTotal('MAR',32521143).
popTotal('MOZ',25203395).
popTotal('MMR',52797319).
popTotal('NAM',2259393).
popTotal('NPL',27474377).
popTotal('NLD',16767705).
popTotal('NCL',258121).
popTotal('NZL',4433100).
popTotal('NIC',5991733).
popTotal('NER',17157042).
popTotal('NGA',168833776).
popTotal('NAC',348859337).
popTotal('MNP',53305).
popTotal('NOR',5018869).
popTotal('OMN',3314001).
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
popTotal('OSS',19919957).
popTotal('PSS',2252782).
popTotal('PAK',179160111).
popTotal('PLW',20754).
popTotal('PAN',3802281).
popTotal('PNG',7167010).
popTotal('PRY',6687361).
popTotal('PER',29987800).
popTotal('PHL',96706764).
popTotal('POL',38542737).
popTotal('PRT',10526703).
popTotal('PRI',3667084).
popTotal('QAT',2050514).
popTotal('ROM',21326905).
popTotal('RUS',143533000).
popTotal('RWA',11457801).
popTotal('WSM',188889).
popTotal('SMR',31247).
popTotal('STP',188098).
popTotal('SAU',28287855).
popTotal('SEN',13726021).
popTotal('SRB',7223887).
popTotal('SYC',87785).
popTotal('SLE',5978727).
popTotal('SGP',5312400).
popTotal('SXM',39088).
popTotal('SVK',5410267).
popTotal('SVN',2058152).
popTotal('SST',29141492).
popTotal('SLB',549598).
popTotal('SOM',10195134).
popTotal('ZAF',51189307).
popTotal('SSD',10837527).
popTotal('ESP',46217961).
popTotal('LKA',20328000).
popTotal('KNA',53584).
popTotal('LCA',180870).
popTotal('MAF',30959).
popTotal('VCT',109373).
popTotal('SDN',37195349).
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
popTotal('SUR',534541).
popTotal('SWZ',1230985).
popTotal('SWE',9516617).
popTotal('CHE',7997152).
popTotal('SYR',22399254).
popTotal('TJK',8008990).
popTotal('TZA',47783107).
popTotal('THA',66785001).
popTotal('TMP',1210233).
popTotal('TGO',6642928).
popTotal('TON',104941).
popTotal('TTO',1337439).
popTotal('TUN',10777500).
popTotal('TUR',73997128).
popTotal('TKM',5172931).
popTotal('TCA',32427).
popTotal('TUV',9860).
popTotal('UGA',36345860).
popTotal('UKR',45593300).
popTotal('ARE',9205651).
popTotal('GBR',63227526).
popTotal('USA',313914040).
popTotal('URY',3395253).
popTotal('UZB',29776850).
popTotal('VUT',247262).
popTotal('VEN',29954782).
popTotal('VNM',88775500).
popTotal('VIR',105275).
popTotal('WBG',4046901).
popTotal('YEM',23852409).
popTotal('ZMB',14075099).
popTotal('ZWE',13724317).

#---------Mapping country code and country name---------#
totalName(Name,Amount):-popCode(Name,Code),popTotal(Code,Amount).

#----------Define membership function----------#
#----------Group low population-----------#
#---------Triangular function---------#
totalTri(Name,Amount,Score):-totalName(Name,Amount),Amount < 2100000 ,
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        Score = max(0,((Amount-1000000)/(2100000-1000000))).
totalTri(Name,Amount,Score):-totalName(Name,Amount),Amount >2100000 ,
        Score = max(0,((5000000-Amount)/(5000000-2100000))).
topTotalTri(Name,Amount,Score):-top(10,order_by(totalTri(Name,Amount,Score),[Score],[d])).

#---------Trapezoidal function-----------#
totalTra(Name,Amount,Score):-totalName(Name,Amount),Amount > 2050000 , Amount < 2500000 , Score = max(0,1).
totalTra(Name,Amount,Score):-totalName(Name,Amount),Amount < 2050000 ,
        Score = max(0,((Amount-1000000)/(2050000-1000000))).
totalTra(Name,Amount,Score):-totalName(Name,Amount),Amount > 2500000 ,
        Score = max(0,((5000000-Amount)/(5000000-2500000))).
topTotalTra(Name,Amount,Score):-top(10,order_by(totalTra(Name,Amount,Score),[Score],[d])).

#----------Left shoulder function----------#
totalLeft(Name,Amount,Score):-totalName(Name,Amount),Amount < 2100000, Amount > 1000000 ,
        Score = max(0,((Amount-1000000)/(2100000-1000000))).
topTotalLeft(Name,Amount,Score):-top(10,order_by(totalLeft(Name,Amount,Score),[Score],[d])).

#----------Right shoulder function----------#
totalRight(Name,Amount,Score):-totalName(Name,Amount),Amount > 2100000, Amount < 5000000 ,
        Score = max(0,((5000000-Amount)/(5000000-2100000))).
topTotalRight(Name,Amount,Score):-top(10,order_by(totalRight(Name,Amount,Score),[Score],[d])).


#----------Group normal population----------#
#----------Triangular function----------#
totalTriMed(Name,Amount,Score):-totalName(Name,Amount),Amount < 25000000 ,
        Score = max(0,((Amount-15000000)/(25000000-15000000))).
totalTriMed(Name,Amount,Score):-totalName(Name,Amount),Amount > 25000000 ,
        Score = max(0,((35000000-Amount)/(35000000-25000000))).
topTotalTriMed(Name,Amount,Score):-top(10,order_by(totalTriMed(Name,Amount,Score),[Score],[d])).

#----------Trapezoidal function----------#
totalTraMed(Name,Amount,Score):-totalName(Name,Amount),Amount > 24000000 , Amount < 27000000 , Score = max(0,1).
totalTraMed(Name,Amount,Score):-totalName(Name,Amount),Amount < 24000000 ,
        Score = max(0,((Amount-15000000)/(24000000-15000000))).
totalTraMed(Name,Amount,Score):-totalName(Name,Amount),Amount > 27000000 ,
        Score = max(0,((35000000-Amount)/(35000000-27000000))).
topTotalTraMed(Name,Amount,Score):-top(10,order_by(totalTraMed(Name,Amount,Score),[Score],[d])).
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
#------------Left shoulder function------------#
totalLeftMed(Name,Amount,Score):-totalName(Name,Amount),Amount < 25000000, Amount > 15000000 ,
        Score = max(0,((Amount-15000000)/(25000000-15000000))).
topTotalLeftMed(Name,Amount,Score):-top(10,order_by(totalLeftMed(Name,Amount,Score),[Score],[d])).

#------------Right shoulder function------------#
totalRightMed(Name,Amount,Score):-totalName(Name,Amount),Amount > 25000000, Amount < 35000000 ,
        Score = max(0,((35000000-Amount)/(35000000-25000000))).
topTotalRightMed(Name,Amount,Score):-top(10,order_by(totalRightMed(Name,Amount,Score),[Score],[d])).

#------------Group high population------------#
#------------Triangular function------------#
totalTriHigh(Name,Amount,Score):-totalName(Name,Amount),Amount < 90000000 ,
        Score = max(0,((Amount-65000000)/(90000000-65000000))).
totalTriHigh(Name,Amount,Score):-totalName(Name,Amount),Amount > 90000000 ,
        Score = max(0,((200000000-Amount)/(200000000-90000000))).
topTotalTriHigh(Name,Amount,Score):-top(10,order_by(totalTriHigh(Name,Amount,Score),[Score],[d])).

#------------Trapezoidal function------------#
totalTraHigh(Name,Amount,Score):-totalName(Name,Amount),Amount > 90000000 , Amount < 93000000 , Score = max(0,1).
totalTraHigh(Name,Amount,Score):-totalName(Name,Amount),Amount < 90000000 ,
        Score = max(0,((Amount-65000000)/(90000000-65000000))).
totalTraHigh(Name,Amount,Score):-totalName(Name,Amount),Amount > 93000000 ,
        Score = max(0,((200000000-Amount)/(200000000-93000000))).
topTotalTraHigh(Name,Amount,Score):-top(10,order_by(totalTraHigh(Name,Amount,Score),[Score],[d])).

#------------Left shoulder function------------#
totalLeftHigh(Name,Amount,Score):-totalName(Name,Amount),Amount < 90000000, Amount > 65000000,
        Score = max(0,((Amount-65000000)/(90000000-65000000))).
topTotalLeftHigh(Name,Amount,Score):-top(10,order_by(totalLeftHigh(Name,Amount,Score),[Score],[d])).

#------------Right shoulder function------------#
totalRightHigh(Name,Amount,Score):-totalName(Name,Amount),Amount > 90000000, Amount < 200000000 ,
        Score = max(0,((200000000-Amount)/(200000000-90000000))).
topTotalRightHigh(Name,Amount,Score):-top(10,order_by(totalRightHigh(Name,Amount,Score),[Score],[d])).
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

ภาคผนวก ข

บทความวิจัยที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างศึกษา

มหาวิทยาลัยเทคโนโลยีสุรนารี

รายชื่อบทความวิจัยที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างศึกษา

กีระชาติ สุขสุทธิ์, ภาสพิชญ์ ชูใจ, กิตติศักดิ์ เกิดประสพ, นิตยา เกิดประสพ. 2556. **การตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัย.**ในงานประชุมสัมมนาวิชาการและการเสนอผลงานวิจัยระดับชาติและนานาชาติ. มหาวิทยาลัยราชภัฏนครราชสีมา. 30 พฤศจิกายน - 1ธันวาคม 2556

กีระชาติ สุขสุทธิ์, กิตติศักดิ์ เกิดประสพ, นิตยา เกิดประสพ. 2557. **การตอบคำถามด้วยภาษาที่มีการควบคุม.** ในงานประชุมวิชาการระดับชาติด้านเทคโนโลยีสารสนเทศ ครั้งที่ 6. ไร่ทองสมบูรณ์คลับ จังหวัดนครราชสีมา. 27 – 28 กุมภาพันธ์ 2557

KeerachartSuksut, PasapitchChujai, KittisakKerdprasop, NittayaKerdprasop. 2014. **Top-k Ranking with Membership Function for Deductive Database.**In Proceedings of the 6th International Conference on Computer Engineering and Technology. Seoul, South Korea. 10 - 11April2014

มหาวิทยาลัยเทคโนโลยีสุรนารี

การตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัย

กีระชาติ สุขสุทธิ์ * , ภาสพิชญ์ ชูใจ **
กิตติศักดิ์ เกิดประสพ * , นิตยา เกิดประสพ *****

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อศึกษาการจัดอันดับ top-k ในคำถามที่คลุมเครือสำหรับจัดอันดับจำนวนประชนกรโลกสำหรับฐานข้อมูลนิรนัย โดยปัญหาส่วนมากเกิดจากการจัดอันดับที่ไม่ถูกต้องเนื่องจากคำถามที่ถามเข้ามามีความคลุมเครือ เช่น "หาประเทศที่มีจำนวนประชากรที่อยู่ระหว่าง 1,500,000 ถึง 3,000,000 คน โดยที่มีจำนวนประชากรใกล้เคียงกับ 2,400,000 มากที่สุด" ดังนั้นผู้วิจัยจึงทำการศึกษาการจัดอันดับโดยการใช้ฟังก์ชันสมาชิกเข้ามาช่วยในการจัดอันดับ โดยการวิเคราะห์ผลจะเป็นการเปรียบเทียบประสิทธิภาพของฟังก์ชันสมาชิกแต่ละฟังก์ชัน และฟังก์ชันแต่ละฟังก์ชันเหมาะสมกับการนำไปใช้งานในลักษณะใด

คำสำคัญ : การทำเหมืองข้อมูล การจัดอันดับ top-k ภาษาดาต้าล็อก

* นักศึกษาปริญญาโท สาขาวิชาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยเทคโนโลยีสุรนารี อีเมลล์ mikaiterng@gmail.com
** นักศึกษาปริญญาเอก สาขาวิชาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยเทคโนโลยีสุรนารี อีเมลล์ apornchujai@hotmail.com
*** ที่ปรึกษาวิทยานิพนธ์หลัก/ร่วม คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

Query Answering with Fuzzy Top-k Ranking Technique in Deductive Database

Keerachart Suksut *, Pasapitch Chujai **
Kittisak Kerdprasop ***, Nittaya Kerdprasop ***

Abstract

The objective of this research is to study top-k ranking in the queries that are ambiguous. In this paper we used world population for ranking in deductive data base .The cause of most problems are wrong ranking because some questions have ambiguity such as "Find the country which have the population between 1,500,000 and 3,000,000 people by the most densely population is approximately 2,400,000 people." This research proposes top-k ranking technique using membership function to evaluate and rank possible answers. We show comparative results for each kind of membership function.

Keyword : Data mining, Datalog$^{top\text{-}k}$, Datalog, Deductive database

* Master Student of Computer Engineering School Suranaree University of Technology E-mail mikaiterng@gmail.com
** Ph.D. Student of Computer Engineering School Suranaree University of Technology E-mail apornchujai@hotmail.com
*** Thesis Advisor/Thesis Co-advisor School of Computer Engineering Suranaree University of Technology

บทนำ

ในเชิงธุรกิจ การจัดอันดับของข้อมูลมีความสำคัญเป็นอย่างมาก เนื่องจากจะช่วยให้สามารถกำหนดขอบเขตของกลุ่มเป้าหมายได้ตรงกับสินค้าที่ต้องการจำหน่าย หรือออกแบบและสร้างผลิตภัณฑ์ใหม่ๆได้ตรงตามความต้องการของตลาดในช่วงเวลานั้นๆ โดยการจัดอันดับสามารถจัดอันดับได้ด้วยการกำหนดเงื่อนไขเพียงเงื่อนไขเดียว หรือกำหนดหลายเงื่อนไขก็ได้[2].

การจัดการกับข้อมูลที่มีความคลุมเครือของข้อมูลเพื่อหาคำตอบที่คลุมเครือน้อยที่สุด ด้วยการใช้เทคนิคการใช้ฟังก์ชันสมาชิกเข้ามาช่วยเพื่อลดความคลุมเครือของข้อมูลลง ส่งผลให้การจัดอันดับในข้อมูลที่มีความคลุมเครือของข้อมูลมีความแม่นยำมากยิ่งขึ้น การจัดอันดับข้อมูลได้ถูกต้อง จะสามารถเพิ่มผลประกอบการขององค์กรได้เป็นอย่างมาก[6].

จากที่กล่าวมาแล้วนั้น ผู้วิจัยจึงเสนอเทคนิคการจัดอันดับของข้อมูลที่มีความคลุมเครือของข้อมูล ด้วยฟังก์ชันสมาชิก เพื่อหาอันดับที่ดีที่สุดของข้อมูลให้มีความถูกต้องและแม่นยำมากขึ้น

งานวิจัยนี้จึงสนใจการจัดอันดับ top-k ด้วยฟังก์ชันสมาชิก 4 ฟังก์ชัน ดังนี้ Triangular Function, Trapezoidal Function, Left Shoulder Function และ Right Shoulder Function[5,6,7]. เพื่อแสดงให้เห็นถึงความสามารถในการจัดอันดับ top-k ของแต่ละฟังก์ชั่น และลักษณะงานที่เหมาะสมที่จะนำแต่ละฟังก์ชั่นไปใช้

การนำฟังก์ชันสมาชิกไปใช้งานในเชิงธุรกิจสามารถนำไปประยุกต์ใช้งานได้อย่างกว้างขวาง เช่นในธุรกิจรถมือสอง ลูกค้าต้องการซื้อรถยนต์โดยมีเงื่อนไข "รถยนต์คันนั้นราคาถูก และผ่านการใช้งานไม่มาก" จะเห็นได้ว่าคำว่า "ราคาถูก" และ "ใช้งานไม่มาก" มีความคลุมเครือ ซึ่งเงื่อนไขของราคาถูก และใช้งานไม่มาก จะขึ้นอยู่กับเงื่อนไขของลูกค้าแต่ละคน การนำฟังก์ชันสมาชิกเข้ามาใช้งานสามารถช่วยให้ลูกค้าตัดสินใจที่จะซื้อรถยนต์คันนั้นๆ ได้ง่ายยิ่งขึ้น

วัตถุประสงค์ของการวิจัย

การวิจัยนี้มีวัตถุประสงค์เพื่อศึกษาการจัดอันดับ top-k ของจำนวนประชากรโลก ซึ่งมีความคลุมเครืออยู่ในช่วงของข้อมูล โดยมุ่งเน้นการจัดอันดับด้วยฟังก์ชันสมาชิกแต่ละฟังก์ชันสำหรับการจัดอันดับ top-k โดยใช้ฟังก์ชันสมาชิกทั้งหมด 4 ฟังก์ชัน และนำค่าที่ได้มาสรุปผลของแต่ละฟังก์ชันของการจัดอันดับ top-k

วิธีการดำเนินการวิจัย

งานวิจัยนี้เป็นการศึกษาการจัดอันดับ top-k ของจำนวนประชากรโลก ซึ่งมีความคลุมเครือของข้อมูล โดยจะแบ่งส่วนของการดำเนินงานออกเป็น 4 ส่วน คือ ศึกษาภาษาดาต้าล็อก, ศึกษาการจัดอันดับ top-k , ศึกษาฟังก์ชันสมาชิก และวัดประสิทธิภาพของแต่ละฟังก์ชันสำหรับการจัดอันดับ top-k ด้วยฟังก์ชันสมาชิกทั้ง 4 ฟังก์ชัน

1) ศึกษาภาษาดาต้าล็อก

ภาษาดาต้าล็อก (Datalog) เป็นภาษาสำหรับการเขียนโปรแกรมเชิงตรรกะ รูปแบบของภาษาดาต้าล็อกประกอบไปด้วย[1,2].

```
path(X,Y) :- edge(X,Y).
```

Head คือส่วนที่อยู่ฝั่งซ้าย ก่อนถึงเครื่องหมาย :- (path(X,Y) ดังตัวอย่าง)

Body คือส่วนที่อยู่ฝั่งขวา นับตั้งแต่เครื่องหมาย :- (edge(X,Y) ดังตัวอย่าง)

Fact คือข้อมูลที่เป็นจริงที่เก็บอยู่ในฐานความรู้ เช่น edge(a,b).

Rule คือรูปแบบของความจริงโดยอาศัยเงื่อนไขในการหาคำตอบ เช่น path(X,Y) :- edge(X,Y).

2) ศึกษาการจัดอันดับ top-k

ดาต้าล็อก$^{top\text{-}k}$ คือการค้นหาอันดับที่ดีที่สุด k อันดับโดยจะอยู่ในรูปแบบดังต่อไปนี้[2].

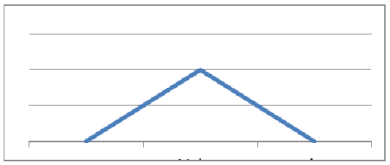
$$q(X,S) \leftarrow \exists y\, body(X,Y),\ S = f(p_i(z_i)).$$

1. X คือตัวแปรหลัก
2. S คือตัวแปรของคะแนน
3. Y คือตัวแปรรอง
4. body(X,Y) คือ disjunctive ของดาต้าล็อกอะตอม
5. z_i คือเซ็ตของค่าคงที่ หรือตัวแปรใน X หรือ Y
6. p_i คือฟัซซี่เพรดดิเคท
7. f คือ สกอร์ริ่งฟังก์ชั่น โดย f จะมีค่าอยู่ในช่วง [0,1]

3) ศึกษาฟังก์ชันสมาชิก

ฟังก์ชันสมาชิก (Membership Function) คือฟังก์ชันที่ช่วยในการกำหนดค่าความเป็นสมาชิก ซึ่งโดยทั่วไปนิยมใช้อยู่ 2 ฟังก์ชัน ได้แก่ Trapezoidal Function และ Triangular Function นอกจากนี้ยังมีฟังก์ชันพิเศษเพิ่มขึ้นมาอีกสองฟังก์ชัน ได้แก่ Left Shoulder Function และ Right Shoulder Function [5,7]. ซึ่งฟังก์ชันพิเศษที่เพิ่มขึ้นมานั้นมีจุดเด่นอยู่ที่การหาค่าน้อยกว่าและค่ามากกว่า ได้ง่ายกว่าการใช้ 2 ฟังก์ชันแรก

1. Triangular Function



$$\mu(x) = \begin{cases} 0, & x \le a \\ \dfrac{x-a}{m-a}, & a < x \le m \\ \dfrac{b-x}{b-m}, & m < x < b \\ x, & x \ge b \end{cases}$$

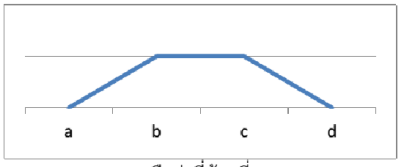
a คือค่าที่น้อยที่สุด

d คือค่าที่มากที่สุด

Value คือค่าที่สนใจ

ที่ a < Value < b

2. Trapezoidal Function



$$\mu(x) = \begin{cases} 0, & (x < a) \text{ or } (x > d) \\ \dfrac{x-a}{b-a}, & a \le x \le b \\ 1, & b \le x \le c \\ \dfrac{d-x}{d-c}, & c < x < d \end{cases}$$

a คือค่าที่น้อยที่สุด

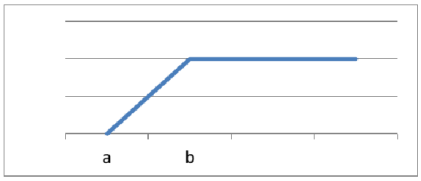
d คือค่าที่มากที่สุด

b คือค่าที่น้อยที่สุดที่ทำให้ค่าความเป็นสมาชิกเป็น 1

c คือค่าที่มากที่สุดที่ทำให้ค่าความเป็นสมาชิกเป็น 1

ที่ a < b < c < d

3. Left Shoulder Function



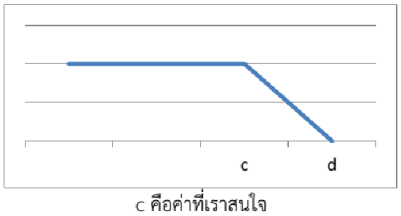
$$\mu(x) = \begin{cases} 0, & x < a \\ \dfrac{x-a}{b-a}, & a \le x \le b \\ 1, & x > b \end{cases}$$

a คือค่าที่น้อยที่สุด

b คือค่าที่เราสนใจ

ที่ a < b

4. Right Shoulder Function



$$\mu(x) = \begin{cases} 0, & x < d \\ \dfrac{d-x}{d-c}, & c \le x \le d \\ 1, & x > c \end{cases}$$

c คือค่าที่เราสนใจ

d คือค่าที่มากที่สุด

ที่ c < d

4) วัดประสิทธิภาพการจัดอันดับ top-k ด้วยฟังก์ชันสมาชิกแต่ละฟังก์ชัน

การจัดอันดับ top-k ของจำนวนประชากรโลก ผู้วิจัยได้เลือกใช้ฟังก์ชันสมาชิก Trapezoidal Function, Triangular Function, Left Shoulder Function และ Right Shoulder Function ในการจัดอันดับจำนวนประชากรโลกโดยมีความคลุมเครือของข้อมูล ซึ่งจะใช้ข้อมูลจำนวนประชากรโลกในปี 2012 จาก

http://databank.worldbank.org/ ข้อมูลนี้มีจำนวน 245 เรคคอร์ด จากตารางที่1 เป็นตัวอย่างข้อมูลจำนวนประชากรโลกจำนวน 15 เรคคอร์ด

ตารางที่ 1 ตัวอย่างข้อมูลจำนวนประชากรโลก

ประเทศ	ประชากร(คน)	ประเทศ	ประชากร(คน)	ประเทศ	ประชากร(คน)
Lithuania	2985509	Armenia	2969081	Mongolia	2796484
Jamaica	2712100	Namibia	2259393	Pacific island small states	2252782
Macedonia FYR	2105575	Slovenia	2058152	Lesotho	2051545
Qatar	2050514	Latvia	2025473	Botswana	2003910
Kosovo	1806366	Gambia	1791225	Guinea-Bissau	1663558

1) Trapezoidal Function

เมื่อนำข้อมูลจำนวนประชากรโลกผ่านกระบวนการจัดอันดับ top-k ด้วยการกำหนดเงื่อนไข "ต้องการหาประเทศที่มีจำนวนประชากรที่อยู่ระหว่าง 1,500,000 คน ถึง 3,000,000 คน โดยที่จะต้องมีจำนวนประชากรใกล้เคียงกับช่วง 2,200,000 คน ถึง 2,500,000 คน มากที่สุด 10 อันดับ" ด้วย Trapezoidal Function ซึ่งผลลัพธ์ที่ได้ออกมาก็คือชื่อประเทศ จำนวนประชากร และค่าความเป็นสมาชิกตามเงื่อนไขที่กำหนดเข้าไป ดังตารางที่ 2

ตารางที่ 2 ช่วงข้อมูลที่ได้จากการจัดอันดับด้วยฟังก์ชันสมาชิก Trapezoidal Function

ประเทศ	จำนวนประชากร(คน)	ค่าความเป็นสมาชิก
Namibia	2,259,393	1
Pacific island small states	2,252,782	1
Macedonia FYR	2,105,575	0.865
Slovenia	2,058,152	0.797
Lesotho	2,051,545	0.788
Qatar	2,050,514	0.786
Latvia	2,025,473	0.751
Botswana	2,003,910	0.720
Jamaica	2,712,100	0.576
Kosovo	1,806,366	0.438

จากตารางที่ 2 จะเห็นได้ว่าเมื่อนำข้อมูลจำนวนประชากรโลกไปผ่านกระบวนการจัดอันดับ top-k ด้วยฟังก์ชัน Trapezoidal Function ประเทศที่มีจำนวนประชากรที่อยู่ในช่วง 2,200,000 ถึง 2,500,000 จะมีค่าความเป็นสมาชิกเป็น 1 ได้แก่ประเทศ Namibia ซึ่งมีจำนวนประชากร 2,259,393 คน และประเทศ Pacific

island small states ซึ่งมีจำนวนประชากร 2,252,782 คน ส่วนประเทศอื่นๆจะมีค่าความเป็นสมาชิกตามเงื่อนไขที่กำหนดเข้าไป

2) Triangular Function

เมื่อนำข้อมูลจำนวนประชากรโลกผ่านกระบวนการจัดอันดับ top-k ด้วยการกำหนดเงื่อนไข "ต้องการหาประเทศที่มีจำนวนประชากรที่อยู่ระหว่าง 1,500,000 คน ถึง 3,000,000 คน โดยที่จะต้องมีจำนวนประชากรใกล้เคียงกับ 2,400,000 คน มากที่สุด 10 อันดับ" ด้วย Triangular Function ซึ่งผลลัพธ์ที่ได้ออกมาก็คือชื่อประเทศ จำนวนประชากร และค่าความเป็นสมาชิกตามเงื่อนไขที่กำหนดเข้าไป ดังตารางที่ 3

ตารางที่ 3 ช่วงข้อมูลที่ได้จากการจัดอันดับด้วยฟังก์ชันสมาชิก Triangular Function

ประเทศ	จำนวนประชากร(คน)	ค่าความเป็นสมาชิก
Namibia	2,259,393	0.844
Pacific island small states	2,252,782	0.836
Macedonia FYR	2,105,575	0.673
Slovenia	2,058,152	0.620
Lesotho	2,051,545	0.613
Qatar	2,050,514	0.612
Latvia	2,025,473	0.584
Botswana	2,003,910	0.560
Jamaica	2,712,100	0.480
Kosovo	1,806,366	0.340

จากตารางที่ 3 จะเห็นได้ว่าเมื่อนำข้อมูลจำนวนประชากรโลกไปผ่านกระบวนการจัดอันดับ top-k ด้วยฟังก์ชัน Triangular Function ประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,400,000 คน มากที่สุดได้แก่ประเทศ Namibia, Pacific island small states, Macedonia FYR, Slovenia, Lesotho, Qatar, Latvia, Botswana, Jamaica และประเทศ Kosovo ตามลำดับ โดยมีจำนวนประชากรและค่าความเป็นสมาชิกดังที่แสดงในตารางที่ 3

3) Left Shoulder Function

เมื่อนำข้อมูลจำนวนประชากรโลกผ่านกระบวนการจัดอันดับ top-k ด้วยการกำหนดเงื่อนไข "ต้องการหาประเทศที่มีจำนวนประชากรที่อยู่ระหว่าง 1,500,000 คน ถึง 3,000,000 คน โดยที่จะต้องมีจำนวนประชากรใกล้เคียงกับ 2,400,000 คน มากที่สุด 10 อันดับ" ด้วย Left Shoulder Function ซึ่งผลลัพธ์ที่ได้ออกมาก็คือชื่อประเทศ จำนวนประชากรและค่าความเป็นสมาชิกตามเงื่อนไขที่กำหนดเข้าไปดังตารางที่4

ตารางที่ 4 ช่วงข้อมูลที่ได้จากการจัดอันดับด้วยฟังก์ชันสมาชิก Left Shoulder Function

ประเทศ	จำนวนประชากร(คน)	ค่าความเป็นสมาชิก
Namibia	2,259, ,393	0.844
Pacific island small states	2,252,782	0.836
Macedonia FYR	2,105,575	0.673
Slovenia	2,058,152	0.620
Lesotho	2,051,545	0.613
Qatar	2,050,514	0.612
Latvia	2,025,473	0.584
Botswana	2,003,910	0.560
Kosovo	1,806,366	0.340
Gambia	1,791,225	0.324

จากตารางที่ 4 จะเห็นได้ว่าเมื่อนำข้อมูลจำนวนประชากรโลกไปผ่านกระบวนการจัดอันดับ top-k ด้วยฟังก์ชัน Left Shoulder Function จะได้ประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,400,000 คน มากที่สุด ได้แก่ประเทศ Namibia, Pacific island small states, Macedonia FYR, Slovenia, Lesotho, Qatar, Latvia, Botswana, Kosovo และประเทศ Gambia ตามลำดับ โดยมีจำนวนประชากร และค่าความเป็นสมาชิกดังที่แสดงในตารางที่ 4

4) Right Shoulder Function

เมื่อนำข้อมูลจำนวนประชากรโลกผ่านกระบวนการจัดอันดับ top-k ด้วยการกำหนดเงื่อนไข "ต้องการหาประเทศที่มีจำนวนประชากรที่อยู่ระหว่าง 1,500,000 คน ถึง 3,000,000 คน โดยที่จะต้องมีจำนวนประชากรใกล้เคียงกับ 2,400,000 คน มากที่สุด 10 อันดับ" ด้วย Right Shoulder Function ซึ่งผลลัพธ์ที่ได้ออกมาก็คือชื่อประเทศ จำนวนประชากรและค่าความเป็นสมาชิกตามเงื่อนไขที่กำหนดเข้าไป ดังตารางที่ 5

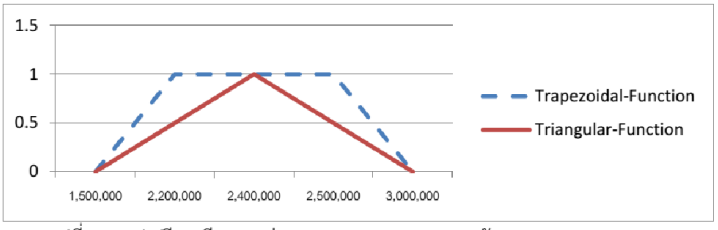
ตารางที่ 5 ช่วงข้อมูลที่ได้จากการจัดอันดับด้วยฟังก์ชันสมาชิก Right Shoulder Function

ประเทศ	จำนวนประชากร(คน)	ค่าความเป็นสมาชิก
Jamaica	2,712,100	0.480
Mongolia	2,796,484	0.339
Armenia	2,969,081	0.052
Lithuania	2,985,509	0.024

จากตารางที่ 5 จะเห็นได้ว่าเมื่อนำข้อมูลจำนวนประชากรโลกไปผ่านกระบวนการจัดอันดับ top-k ด้วยฟังก์ชัน Right Shoulder Function จะได้ประเทศที่มีจำนวนประชากรใกล้เคียงกับ 2,400,000 คน มากที่สุด ได้แก่ประเทศ Jamaica, Mongolia, Armenia และประเทศ Lithuania ตามลำดับ โดยมีจำนวนประชากร และค่าความเป็นสมาชิกดังที่แสดงในตารางที่ 4 โดยจะมีเพียง 4 อันดับที่ตรงตามเงื่อนไขตามที่กำหนดเข้าไป

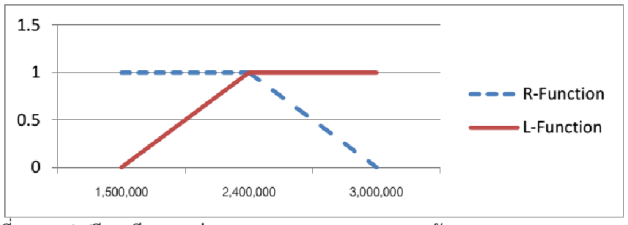
ผลการวิจัย

งานวิจัยนี้ได้ทำการทดลองเปรียบเทียบประสิทธิภาพการจัดอันดับ top-k ของข้อมูลจำนวนประชากรโลกด้วยการใช้ฟังก์ชันสมาชิก ทั้ง 4 แบบ (Trapezoidal Function, Triangular Function, Left Shoulder Function และ Right Shoulder Function) จากการทดลองจะเห็นได้ว่าการใช้ Trapezoidal Function และ Triangular Function จะได้ผลลัพธ์ที่ออกมาเหมือนกัน แต่แตกต่างกันตรงที่ค่าความเป็นสมาชิก ซึ่งจะขึ้นอยู่กับเงื่อนไขที่กำหนดเข้าไป ดังรูปที่ 1



รูปที่ 1 กราฟเปรียบเทียบระหว่าง Trapezoidal function กับ Triangular function

ส่วนการใช้ Left Shoulder Function จะเปรียบเสมือนกับการหาข้อมูลที่มีค่าน้อยกว่าหรือเท่ากับค่าที่ต้องการ ส่วน Right Shoulder Function จะเป็นการหาข้อมูลที่มีค่ามากกว่าหรือเท่ากับค่าที่ต้องการ ดังรูปที่ 2



รูปที่ 2 กราฟเปรียบเทียบระหว่าง Right shoulder function กับ Left shoulder function

สรุปและอภิปรายผล

การจัดอันดับ top-k ของข้อมูลจำนวนประชากรโลกนั้นสิ่งสำคัญที่สุดคือการกำหนดเงื่อนไขของช่วงประชากรที่เราต้องการค้นหา เนื่องจากการกำหนดเงื่อนไขในการค้นหาเปรียบเสมือนกับการสร้างสมการเพื่อคำนวณค่าความเป็นสมาชิก ซึ่งค่าความเป็นสมาชิกจะใช้ในการจัดอันดับ top-k อันดับ ดังนั้นหากกำหนดเงื่อนไขผิดพลาด อาจจะส่งผลให้ผลลัพธ์ที่ได้มีความคลาดเคลื่อนไปได้เช่นกัน

จากการทดลองดังกล่าวจะเห็นได้ว่า ฟังก์ชันสมาชิก Trapezoidal Function กับ Triangular Function จะให้ค่าความเป็นสมาชิกออกมาแตกต่างกัน แต่จะมีค่าไปในทิศทางเดียวกัน เหมาะสำหรับการนำไปจัดอันดับข้อมูลที่มีความคลุมเครือในส่วนของการกำหนดเงื่อนไขที่มีการระบุขอบเขตของสิ่งที่ต้องการจัดอันดับ โดยหากต้องการจัดอันดับข้อมูลที่มีค่าที่เราให้ความสนใจเพียงค่าเดียว การใช้ฟังก์ชันสมาชิก Triangular Function จะมีความถูกต้องและแม่นยำมากกว่าการใช้ฟังก์ชันสมาชิก Trapezoidal Function ในทางกลับกัน หากมีค่าที่เรา

สนใจมากกว่าหนึ่งค่า การใช้ฟังก์ชันสมาชิก Trapezoidal Function จะมีความถูกต้องและแม่นยำมากกว่าการใช้ฟังก์ชันสมาชิก Triangular Function ส่วนฟังก์ชันสมาชิก Left Shoulder Function และ Right Shoulder Function เหมาะสำหรับการนำไปจัดอันดับข้อมูลที่มีค่าน้อยกว่า หรือมากกว่าค่าที่ต้องการ ตามลำดับ

เอกสารอ้างอิง

[1] N. Kerdprasop (2013). "Data Mining With Datalog", Retrieved June 3, 2013, form https://sites.google.com/site/nittayak/home/clp

[2] U.Straccia, "Towards top-*k* query answering in deductive databases", IEEE, pp.4873–4879, 2006.

[3] U.Straccia, "Towards top-*k* query answering in description logics : the case of DL-Lite", Proceedings of the 10 th European Conference on Logics in Artificial Intelligence (JELIA-06), Liverpool, UK, pp. 439–451, 2006.

[4] U. Straccia, "Answering vague queries in fuzzy DL-Lite", Proceedings of the 11th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU-06), Paris, pp. 2238–2245, 2006.

[5] F. Bobillo and U. Straccia, "fuzzyDL: An Expressive Fuzzy Description Logic Reasoner", Proceedings of the IEEE International Conference on Fuzzy System, Hong Kong, pp. 923-930, 2008.

[6] J. Dombi, "Membership function as an evaluation", Fuzzy sets and systems, 1990

[7] U. Straccia and N. Madrid, "A top-k query answering procedure for fuzzy logic programming", Fuzzy Sets and Systems, vol.205, pp.1-29, 2012.

การประชุมวิชาการระดับประเทศด้านเทคโนโลยีสารสนเทศ (National Conference on Information Technology: NCIT) ครั้งที่ 6

การตอบคำถามด้วยภาษาที่มีการควบคุม

กีระชาติ สุขสุทธิ์, กิตติศักดิ์ เกิดประสพ, นิตยา เกิดประสพ

สาขาวิชาวิศวกรรมคอมพิวเตอร์ สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี
Emails: mikaiterng@gmail.com

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อนำเสนอแนวคิดและวิธีการตอบคำถาม และป้อนข้อมูลให้มีความใกล้เคียงกับภาษาที่ผู้ใช้ทั่วไปสามารถเข้าใจง่าย (ให้มีความใกล้เคียงกับภาษามนุษย์มากที่สุด) โดยนำเสนอขั้นตอนและวิธีการแปลงจากข้อมูลที่เป็นภาษาดาต้าล็อกให้อยู่ในรูปแบบของ Controlled Natural Language ดังนั้นผู้วิจัยจึงศึกษาการการเขียน Controlled Natural Language ด้วยเครื่องมือ ACE (Attempto Controlled English) โดยการวิเคราะห์ผลจะเป็นการถามคำถามด้วยภาษาอังกฤษเพื่อหาคำตอบจากการระบุองค์ความรู้ที่อยู่ในรูปแบบของ Controlled Natural Language

คำสำคัญ-- ภาษาควบคุม, เอซีอี, ดาต้าล็อก

1. บทนำ

ปัจจุบันข้อมูลถูกจัดเก็บให้อยู่ในรูปแบบดิจิตอลโดยอาจจะเก็บอยู่ในฐานความรู้ แต่โดยทั่วไปแล้วผู้ใช้ทั่วไปทำความเข้าใจภาษาที่ใช้ในการเรียกดูข้อมูลเหล่านั้นได้ยาก การนำข้อมูลที่เก็บอยู่ในฐานความรู้จึงจำเป็นต้องอาศัยผู้เชี่ยวชาญทางด้านฐานข้อมูลทำการแปลงข้อมูลให้อยู่ในรูปแบบที่ผู้ใช้ทั่วไปต้องการ

จากปัญหาที่ผู้ใช้ทั่วไปทำความเข้าใจภาษาที่ใช้ในการเรียกดูข้อมูลเหล่านั้นได้ยาก หากมีระบบหรือการเก็บข้อมูลหรือการถามคำถามโดยที่ภาษาที่ใช้เก็บข้อมูลลงในฐานความรู้ หรือถามคำถาม มีความใกล้เคียงกับภาษาที่ผู้ใช้ทั่วไปใช้งาน (ภาษาอังกฤษ) ซึ่งจะส่งผลให้ผู้ใช้ทั่วไปสามารถทำงานได้สะดวกมากยิ่งขึ้น

งานวิจัยนี้จึงนำเสนอวิธีการแปลงข้อมูลจากภาษาดาต้าล็อก (Datalog) ให้อยู่ในรูปแบบ Controlled Natural Language เพื่อนำข้อมูลไปเก็บในฐานความรู้และถามคำถามเพื่อหาคำตอบ โดยมุ่งเน้นที่ภาษาที่ใช้คือภาษาอังกฤษในการนำข้อมูลไปเก็บในฐานความรู้และถามคำถาม

2. ภาษาดาต้าล็อก

ภาษาดาต้าล็อก (Datalog) เป็นภาษาสำหรับการเขียนโปรแกรมเชิงตรรกะ รูปแบบของภาษาดาต้าล็อกประกอบไปด้วย[1,2].

```
head(X,Y) :- body(X,Y).
```

Head คือส่วนที่อยู่ฝั่งซ้าย ก่อนถึงเครื่องหมาย :-
Body คือส่วนที่อยู่ฝั่งขวา นับตั้งแต่เครื่องหมาย :-
Fact คือข้อมูลที่เป็นจริงที่เก็บอยู่ในฐานความรู้ เช่น

```
country(Thailand, Asia).
```

Rule คือรูปแบบของความจริงโดยอาศัยเงื่อนไขในการหาคำตอบ เช่น

```
continent(Name, Con) :- country(Name, Con).
```

3. Controlled Natural Language

Natural Language เป็นภาษาที่สามารถสั่งงานคอมพิวเตอร์โดยใช้รูปแบบของภาษามนุษย์ได้เลย คำสั่งอยู่ในรูปแบบที่ไม่แน่นอนตายตัว แต่คอมพิวเตอร์จะทำการแปลให้ออกมาในรูปที่คอมพิวเตอร์เข้าใจได้ ภาษานี้ถูกสร้างขึ้นมาจากเทคโนโลยีทางด้านระบบผู้เชี่ยวชาญ (Expert system) ภาษาธรรมชาติจะใช้ระบบฐานความรู้ (Knowledge Base System) ช่วยในการแปลความหมายของคำสั่งต่างๆ

Controlled Natural Language เป็นภาษากึ่งธรรมชาติ กล่าวคือ เป็นภาษาที่สามารถสั่งงานคอมพิวเตอร์ด้วยการใช้

รูปแบบของภาษามนุษย์ แต่ยังต้องมีการควบคุมบางคำ เพื่อให้คอมพิวเตอร์สามารถทำงานได้และมีข้อผิดพลาดน้อยลง ตัวอย่างเช่น

ในฐานความรู้ มีข้อมูลเป็นภาษาดาต้าล็อก ดังนี้

```
:-type(counTry (name:string, topography:string)).
counTry(Thailand, Tropical-wet).
```

จากฐานความรู้ จะเห็นได้ว่ามนุษย์ทั่วไปอาจจะทำความเข้าใจข้อมูลข้างต้นได้ยาก แต่เมื่อนำข้อมูลมาแปลงให้อยู่ในรูปแบบของ Controlled Natural Language จะได้

```
p:Thailand is a country and has a n:Tropical-wet.
n: Tropical-wet is a n:Tropography.
```

จากการแปลงให้อยู่ในรูปแบบของ Controlled Natural Language จะได้ว่า Thailand เป็นประเทศ และมีสภาพภูมิอากาศอยู่ในเขตร้อนชื้น ซึ่งจะเห็นว่าคนทั่วไปสามารถเข้าใจได้ง่ายขึ้น งานวิจัยนี้จะใช้เครื่องมือในการเขียน Controlled Natural Language ด้วย ACE (Attempto Controlled English)

4. ACE (Attempto Controlled English)

ACE (Attempto Controlled English) เป็นเครื่องมือหนึ่งที่สามารถทำงานในรูปแบบของ Controlled Natural Language ได้รับการพัฒนาเพื่อใช้เป็นภาษาที่ใช้แทนความรู้ กล่าวคือไม่จำเป็นต้องนำความรู้จากการเก็บข้อมูลด้วยภาษาคอมพิวเตอร์ และทำการค้นหาด้วยภาษาคอมพิวเตอร์เพียงอย่างเดียว แต่สามารถเก็บข้อมูล และสอบถามข้อมูลด้วยภาษาที่ใกล้เคียงกับภาษามนุษย์ โดยสามารถใช้ ACE (Attempto Controlled English) ผ่านหน้าเว็บไซต์ http://attempto.ifi.uzh.ch/site/resources/

5. แปลงข้อมูลเป็น Controlled Natural Language

การแปลงข้อมูลเพื่อทำให้อยู่ในรูปแบบของ Controlled Natural Language ผู้วิจัยได้เลือกใช้ข้อมูลจำนวนประชากรโลก อัตราการเกิด และ อัตราการค้า ในการนำมาใช้เป็นฐานความรู้ ซึ่งจะใช้ข้อมูลจาก http://databank.worldbank.org/ ข้อมูลนี้มีจำนวน 245 เรคคอร์ด จากตารางที่ 1 เป็นตัวอย่างข้อมูลจำนวน 14 เรคคอร์ด

ตารางที่ 1 ตัวอย่างข้อมูลจำนวนประชากรโลก

ประเทศ	จำนวนประชากร	อัตราการเกิด	อัตราการค้า
Japan	127,817,277	8.30	0.781
China	1,344,130,000	13.50	0.875
India	1,221,156,319	21.00	1.084
Singapore	5,183,700	9.50	0.981
Thailand	66,576,332	10.75	1.015
Switzerland	7,912,398	10.20	1.061
United States	311,587,816	12.70	0.968
United Kingdom	62,752,472	12.90	0.914
Angola	20,180,490	45.57	2.084
Australia	22,323,900	13.29	1.472
Argentina	40,728,738	17.06	1.333
Spain	46,174,601	10.2	0.983
Saudi Arabia	27,761,728	20.32	1.427
Brazil	196,935,134	15.33	1.318

จากข้อมูลในตารางที่ 1 เมื่อนำข้อมูลไปจัดกลุ่มด้วยภาษาดาต้าล็อคจากเงื่อนไขที่เขียนขึ้น ดังนี้

```
lowOfPop(Name,Pop) :-
counTry(Name,Pop,BirthOfRate,TermOfTrade),
Pop < 60,000,000.
medOfPop(Name,Pop) :-
counTry(Name,Pop,BirthOfRate,TermOfTrade),
Pop > 60,000,000, Pop < 1,000,000,000.
highOfPop(Name,Pop) :-
counTry(Name,Pop,BirthOfRate,TermOfTrade),
Pop > 1,000,000,000.
lowOfBirth(Name,BirthOfRate) :-
counTry(Name,Pop,BirthOfRate,TermOfTrade),
BirthOfRate < 10.
medOfBirth(Name,BirthOfRate) :-
counTry(Name,Pop,BirthOfRate,TermOfTrade),
BirthOfRate > 10, BirthOfRate < 20.
```

```
highOfBirth(Name,BirthOfRate) :-
counTry(Name,Pop,BirthOfRate,TermOfTrade),
BirthOfRate > 20.
lowOfTerm(Name,TermOfTrade) :-
counTry(Name,Pop,BirthOfRate,TermOfTrade),
TermOfTrade < 1.0.
medOfTerm(Name,TermOfTrade) :-
counTry(Name,Pop,BirthOfRate,TermOfTrade),
TermOfTrade>1.0,TermOfTrade<1.4.
highOfTerm(Name,TermOfTrade) :-
counTry(Name,Pop,BirthOfRate,TermOfTrade),
TermOfTrade > 1.4.
```

จะแบ่งกลุ่มข้อมูลในตารางที่ 1 ได้ดังตารางที่ 2

ตารางที่ 2 ข้อมูลที่ผ่านการแบ่งกลุ่มด้วยภาษาดาต้าล็อก

ประเทศ	จำนวนประชากร	อัตราการเกิด	อัตราการค้า
Japan	normal	low	low
China	high	normal	low
India	high	high	normal
Singapore	low	low	low
Thailand	normal	normal	normal
Switzerland	low	normal	normal
United States	normal	normal	low
United Kingdom	normal	normal	low
Angola	low	high	high
Australia	low	normal	high
Argentina	normal	high	normal
Spain	normal	normal	low
Saudi Arabia	low	high	high
Brazil	normal	normal	normal

จากตารางที่ 2 หากเก็บข้อมูลในฐานความรู้ด้วยภาษาดาต้าล็อก จะได้ฐานความรู้ดังนี้

```
:-type(counTry(name:string, pop:string,
birthrate:string, termoftrade:string)).
counTry(Japan, normal, low, low).
counTry(China, high, normal, low).
counTry(India, high, high, normal).
counTry(Singapore, low, low, low).
counTry(Thailand, normal, normal, normal).
counTry(Switzerland, low, normal, normal).
counTry(United States, normal, normal, low).
counTry(United Kingdom, normal, normal, low).
counTry(Angola, low, high, high)
counTry(Australia, low, normal, high)
counTry(Argentina, normal, high, normal).
counTry(Spain, normal, normal, low).
counTry(Saudi Arabia, low, high, high).
counTry(Brazil, normal, normal, normal).
```

เมื่อนำข้อมูลจากตารางที่ 2 มาแปลงให้อยู่ในรูปแบบของ Controlled Natural Language จะได้ฐานความรู้ที่อยู่ในรูปแบบของ Controlled Natural Language ดังต่อไปนี้

```
p:Japan has a normal population and has a low
n:term-of-trade and has a low birthrate.
p:China has a high population and has a low n:term-
of-trade and has a normal birthrate.
p:India has a high population and has a normal
n:term-of-trade and has a high birthrate.
p:Singapore has a low population and has a low
n:term-of-trade and has a low birthrate.
p:Thailand has a normal population and has a
normal n:term-of-trade and has a normal birthrate.
p:Switzerland has a low population and has a
normal n:term-of-trade and has a normal birthrate.
p:United-States has a normal population and has a
low n:term-of-trade and has a normal birthrate.
p:United-Kingdom has a normal population and has
a low n:term-of-trade and has a normal birthrate.
p:Angola has a low population and has a high
n:term-of-trade and has a high birthrate.
```

The following minimal subsets of the axioms answer the query:

- Subset 1
  - 10: p:Australia has a low population and has a high n:term-of-trade and has a normal birthrate.
  - Substitution: what = Australia
- Subset 2
  - 6: p:Switzerland has a low population and has a normal n:term-of-trade and has a normal birthrate.
  - Substitution: what = Switzerland

ภาพที่ 1 ผลลัพธ์จากการถามคำถาม " What has a low population and has a normal birthrate?"

The following minimal subsets of the axioms answer the query:

- Subset 1
  - 9: p:Angola has a low population and has a high n:term-of-trade and has a high birthrate.
  - Substitution: what = Angola

ภาพที่ 2 ผลลัพธ์จากการถามคำถาม " What has a high birthrate and has a high n:term-of-trade?"

p:Australia has a low population and has a high n:term-of-trade and has a normal birthrate.
p:Argentina has a normal population and has a high n:term-of-trade and has a normal birthrate.
p:Spain has a normal population and has a low n:term-of-trade and has a normal birthrate.
p:Saudi-Arabia has a low population and has a high n:term-of-trade and has a high birthrate.
p:Brazil has a normal population and has a normal n:term-of-trade and has a normal birthrate.

6. ผลการวิจัย

เมื่อทดสอบทำการถามคำถามเพื่อหาคำตอบ จากคำถาม

1) What has a low population and has a normal birthrate? จะได้ผลลัพธ์ดังภาพที่ 1

จากภาพที่ 1 จะเห็นได้ว่า คำตอบที่ได้ออกมาจะมีอยู่ 2 ประเทศที่ตรงตามเงื่อนไข "low population and normal birthrate" ได้แก่ประเทศ Australia ซึ่งได้จากฐานความรู้ที่บอกว่า "p:Australia has a low population and has a high n:term-of-trade and has a normal birthrate." และประเทศ Switzerland จากฐานความรู้ที่บอกว่า " p:Switzerland has a low population and has a normal n:term-of-trade and has a normal birthrate."

2) What has a high birthrate and has a high n:term-of-trade? จะได้ผลลัพธ์ดังภาพที่ 2

จากภาพที่ 2 จะเห็นได้ว่า คำตอบที่ได้ออกมาจะมีอยู่ 1 ประเทศที่ตรงตามเงื่อนไข "high birthrate and high term-of-trade" ซึ่งได้แก่ประเทศ Angola จากฐานความรู้ที่บอกว่า " p:Angola has a low population and has a high n:term-of-trade and has a high birthrate."

3) What has a high population and has a high n:term-of-trade?

จะไม่มีประเทศที่ตรงตามเงื่อนไขเนื่องจากไม่มีฐานความรู้ที่บอกว่าประเทศไหนมีจำนวนประชากรอยู่ในระดับสูงและมีอัตราการค้า อยู่ในระดับสูง

7. สรุปและอภิปรายผล

การแปลงข้อมูลจากภาษาดาต้าล็อกให้อยู่ในรูปแบบของ Controlled Natural Language สิ่งสำคัญคือต้องทราบไวยากรณ์ของ Controlled Natural Language หากไม่กำหนดไวยากรณ์ของคำศัพท์ที่นอกเหนือจากที่มีอยู่จะส่งผลให้ไม่สามารถสอบถามข้อมูลนั้นๆได้ การกำหนดไวยากรณ์ให้คำศัพท์จึงมีความสำคัญ

จากการทดลองดังกล่าวจะเห็นได้ว่าเมื่อแปลงข้อมูลจากภาษาดาต้าล็อกให้อยู่ในรูปแบบของ Controlled Natural Language และนำไปเก็บในฐานความรู้แล้วจะสามารถถาม

คำถามด้วยภาษาที่ผู้ใช้ทั่วไปสามารถทำความเข้าใจได้ง่าย และได้ผลลัพธ์ถูกต้องตามฐานความรู้ที่มีอยู่

ในครั้งต่อไป งานวิจัยนี้อาจมุ่งเน้นไปที่การแปลงฐานความรู้จาก Controlled Natural Language ไปสู่การทำงานในรูปแบบของ OWL Language

8. เอกสารอ้างอิง

[1] N.E. Fuchs and K. Kaljurand and T. Kuhn, "Attempto Controlled English for Knowledge Representation", 4th International Summer School 2008, Venice, Italy, pp 104-124, 2008.

[2] N.E. Fuchs and K. Kaljurand and G. Schneider, "Attempto Controlled English Meets the Challenges of Knowledge Representation, Reasoning, Interoperability and User Interfaces", FLAIRS Conference, 2006.

[3] N.E. Fuchs and U. Schwertel and S. Torge, "Controlled Natural Language Can Replace First-Order Logic", IEEE, pp.295–298, 1999.

[4] N.E. Fuchs and U. Schwertel, " Reasoning in Attempto Controlled English", International Workshop, Mumbai, India, pp 174-188, 2003.

[5] G. Hart and M. Johnson and C. Dolbear, "Rabbit: Developing a Control Natural Language for Authoring Ontologies", 5th European Semantic Web Conference, ESWC 2008, Tenerife, Canary Islands, Spain, 2008.

[6] N. Kerdprasop (2013). "Data Mining With Datalog", Retrieved June 3, 2013, form https://sites.google.com/site/nittayak/home/clp

[7] D.D. Lewis and K.S. Jones, "Natural language processing for information retrieval", Magazine Communications of the ACM Volume 39 Issue 1, pp. 92–101, 1996.

Top-k Ranking with Membership Function for Deductive Database

Keerachart Suksut, Pasapitch Chujai, Nittaya Kerdprasop and Kittisak Kerdprasop

***Abstract*—The objective of this research is to study top-k ranking in the queries that are ambiguous. In this paper we demonstrate our query answering strategy for ranking world population for deductive data base .The cause of most problems are wrong ranking because some questions have ambiguity such as "Find the country which have the population between 1,500,000 and 3,000,000 people by the most densely population is approximately 2,400,000 people." This research proposes top-k ranking technique using membership function to evaluate and rank possible answers. We show comparative results for each kind of membership function.**

***Index Terms*—ER Data mining, Datalogtop-k, Datalog, Deductive database**

I. Introduction

Ranking of data is highly important because its help to define direct scope of target with goods which we want to attribute or design and create new products which market demand during that time[1].

Ranking can make by determination only one condition or more. Ranking the fuzzy data for finding least fuzzy answer by using membership function technique to reduce fuzzy of data. Result ranking the fuzzy data have more precision. Correct ranking can increase business profit highly[2].

In paper, main idea is apply the membership function to reduce fuzzy data in part of membership value computation. We show some difference of apply the membership function (Triangular Function, Trapezoidal Function, Left Shoulder Function and Right Shoulder Function).

Aforementioned, researcher exhibit ranging technique of the fuzzy data by member function for find the best ranking of data to increase more accuracy and precise

Manuscript received December 13, 2013; revised January 3, 2014. This work was supported in part by grant from Suranaree University of Technology through the funding of Data Engineering Research Unit.

K. Suksut is a master student of Computer Engineering School Suranaree University of Technology, Nakhon Ratchasima 30000, Thailand (e-mail: mikaiterng@gmail.com).

P. Chujai is a doctoral student of Computer Engineering School Suranaree University of Technology, Nakhon Ratchasima 30000, Thailand.

N. Kerdprasop is an associate professor of Computer Engineering School Suranaree University of Technology, Nakhon Ratchasima 30000, Thailand.

K. Kerdprasop is an associate professor of Computer Engineering School Suranaree University of Technology, Nakhon Ratchasima 30000, Thailand.

II. Basic Datalog

Datalog language used for logical programming in the form [1,3]

```
head(X,Y) :- body(X,Y).
```

where head(X,Y) is the head of the rule, and body(X,Y). is the body of the rule.

Fact is the data is actually stored in the knowledge base and in the form

```
data(X,Y).
```

Rule is form of fact-based conditions of find the answer. Rule in the form

```
findData(X,Y) :- data(X,Y), Y < 10,000.
```

III. Datalog$^{TOP\text{-}K}$

Datalog$^{top\text{-}k}$[4,5] is finding ranking top k in the form

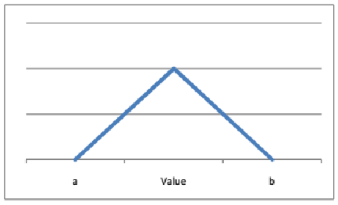
$$q(X,S) \leftarrow \exists ybody\,(X,Y), S = f(Pi(Zi)) \quad (1)$$

1. X are the distinguished cariables;
2. S is the score variable;
3. Y are non-distinguished;
4. body(X,Y) is disjunctive query of atom;
5. zi is set of constants or variables;
6. pi is all fuzzy predicate;
7. f is scoring function taking values in [0,1];

Fuzzy Set is scope of the fuzzy information and different from crisp, Crisp Set have 2 values of membership degree (true/false or 0/1) but fuzzy set have multiples value and taking values in [0,1] , fuzzy set used membership function for indentify value of membership.

membership function[2,6,7]

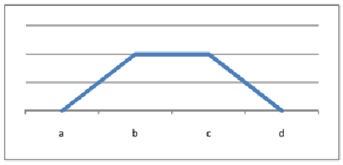
A. Triangular Function



a is a lower limit
b is an upper limit
Value is interested value
Where a < Value < b

$$\mu(x)=\begin{cases}0, x \le a \\ \dfrac{x-a}{m-a}, a < x \le m \\ \dfrac{b-x}{b-m} m < x < b \\ x, x \ge b\end{cases} \quad (2)$$

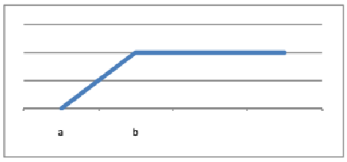
B. Trapezoidal Function



a is a lower limit
d is an upper limit
b is a lower support limit
c is an upper support limit
Where a < b < c < d

$$\mu(x)=\begin{cases}0, (x \le a) or (x > d) \\ \dfrac{x-a}{b-a}, a \le x \le b \\ 1, b \le x \le c \\ \dfrac{d-x}{d-c}, c < x < b\end{cases} \quad (3)$$

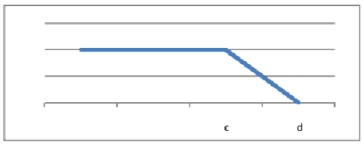
C. Left Shoulder Function



a is a lower limit
b is interested value
Where a < b

$$\mu(x)=\begin{cases}0, x < a \\ \dfrac{x-a}{b-a}, a \le x \le b \\ 1, x > b\end{cases} \quad (4)$$

D. Right Shoulder Function



c is interested value
d is an upper limit
Where c < d

$$\mu(x)=\begin{cases}0, x < d \\ \dfrac{d-x}{d-c}, c \le x \le d \\ 1, x > c\end{cases} \quad (5)$$

IV. TOP-K COMPUTING

Datalog$^{top\text{-}k}$ of population, Researcher selected membership function such as Trapezoidal Function, Triangular Function, Left Shoulder Function and Right Shoulder Function for find ranking top-k answering. and select data of population in 2012 from http://databank.worldbank.org/ , this data have 245 record, in table I is example data of population 15 record:

Table I. example data of population

Country	Population
Lithuania	2,985,509
Jamaica	2,712,100
Macedonia FYR	2,105,575
Qatar	2,050,514
Kosovo	1,806,366
Armenia	2,969,081
Namibia	2,259,393
Slovenia	2,058,152
Latvia	2,025,473
Gambia	1,791,225
Mongolia	2,796,484
Pacific island small states	2,252,782
Lesotho	2,051,545
Botswana	2,003,910
Guinea-Bissau	1,663,558

A. Triangular Function

Population top-10 ranking "finding the country which have the population between 1,500,000 and 3,000,000 people by the population approximate 2,200,000 to 2,500,000 people most" with trapezoidal function in table II.

Table II. Population top-10 ranking with triangular function.

Country	Population	Membership Value
Namibia	2,259,393	1
Pacific island small states	2,252,782	1
Macedonia FYR	2,105,575	0.865
Slovenia	2,058,152	0.797
Lesotho	2,051,545	0.788
Qatar	2,050,514	0.786
Latvia	2,025,473	0.751
Botswana	2,003,910	0.720
Jamaica	2,712,100	0.576
Kosovo	1,806,366	0.438

Form table II. After top-k ranking with trapezoidal function. Country which have the population between 1,500,000 and 3,000,000 people by the population approximate 2,200,000 to 2,500,000 people most and have value of membership value is 1 is Namibia which have the population 2,259,393 people and Pacific island small states which have the population 2,252,782 people. Other country have the value of membership according to the condition.

B. Trapezoidal Function

Population top-10 ranking "finding the country which have the population between 1,500,000 and 3,000,000 people by the population approximate to 2,400,000 people most" with triangular function in table III.

Table III. Population top-10 ranking with trapezoidal function.

Country	Population	Membership Value
Namibia	2,259,393	0.844
Pacific island small states	2,252,782	0.836
Macedonia FYR	2,105,575	0.673
Slovenia	2,058,152	0.620
Lesotho	2,051,545	0.613
Qatar	2,050,514	0.612
Latvia	2,025,473	0.584
Botswana	2,003,910	0.560
Jamaica	2,712,100	0.480
Kosovo	1,806,366	0.340

Form table III. After top-k ranking with trapezoidal function. Country which have the population between 1,500,000 and 3,000,000 people by the population approximate to 2,400,000 people most is Namibia, Pacific island small states, Macedonia FYR, Slovenia, Lesotho, Qatar, Latvia, Botswana, Jamaica and Kosovo respectively and have the value of membership according to the condition.

C. Left Shoulder Function

Population top-10 ranking "finding the country which have the population between 1,500,000 and 3,000,000 people by the population approximate to 2,400,000 people most" with left shoulder function in table IV.

Table IV. Population top-10 ranking with left shoulder function.

Country	Population	Membership Value
Namibia	2,259,393	0.844
Pacific island small states	2,252,782	0.836
Macedonia FYR	2,105,575	0.673
Slovenia	2,058,152	0.620
Lesotho	2,051,545	0.613
Qatar	2,050,514	0.612
Latvia	2,025,473	0.584
Botswana	2,003,910	0.560
Kosovo	1,806,366	0.340
Gambia	1,791,225	0.324

Form table IV. After top-k ranking with left shoulder function. Country which have the population approximate 2,400,000 people most is Namibia, Pacific island small states, Macedonia FYR, Slovenia, Lesotho, Qatar, Latvia, Botswana, Kosovo and Gambia respectively and have the value of membership according to the condition.

D. Right Shoulder Function

Population top-10 ranking "finding the country which have the population between 1,500,000 and 3,000,000 people by the population approximate to 2,400,000 people most" with left shoulder function in table V.

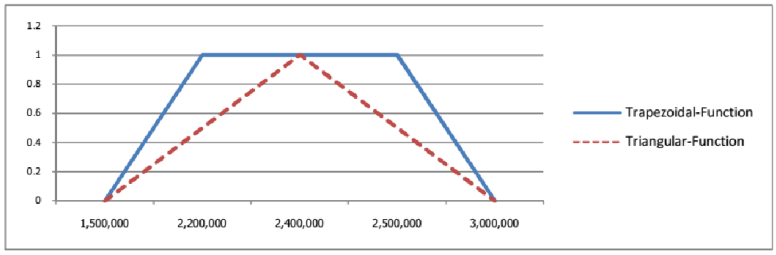


Figure I. Difference between trapezoidal function and triangular function.

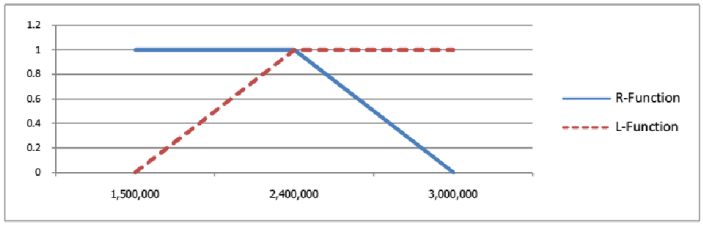


Figure II. Difference between trapezoidal function and triangular function.

Table V. Population top-10 ranking with right shoulder function.

Country	Population	Membership Value
Jamaica	2,712,100	0.480
Mongolia	2,796,484	0.339
Armenia	2,969,081	0.052
Lithuania	2,985,509	0.024

Form table V. After top-k ranking with right shoulder function. Country which have the population approximate 2,400,000 people most is Jamaica, Mongolia, Armenia and Lithuania respectively and have the value of membership according to the condition.

V. Conclusion

Top-k ranking of population. Condition step is very important because condition for find item is a create equation for evaluate membership function. Which membership function is a ranking top-k of population if wrong evaluate of membership value then wrong top-k ranking.

According to experiments can be seen that membership function Trapezoidal Function and Triangular Function will generate in different membership value while both of function have same value direction. The ambiguity information is suitably to sort in condition range. If interested value have one value. Triangular function have accuracy more than Trapezoidal function. If interested value have more one value. Trapezoidal function have accuracy more than Triangular function (In Figure I).

Additionally, membership function Left Shoulder Function and Right Shoulder Function are properly to classify information that less or greater value, respectively (In Figure II).

References

[1] U.Straccia, "Towards top-*k* query answering in deductive databases", IEEE, pp.4873–4879, 2006.

[2] J. Dombi, "Membership function as an evaluation", Fuzzy sets and systems, 1990.

[3] N. Kerdprasop (2013). "Data Mining With Datalog" , form https://sites.google.com/site/nittayak/home/clpJ. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[4] U.Straccia, "Towards top-*k* query answering in description logics : the case of DL-Lite", Proceedings of the 10 th European Conference on Logics in Artificial Intelligence (JELIA-06), Liverpool, UK, pp. 439–451, 2006.

[5] U. Straccia, "Answering vague queries in fuzzy DL-Lite", Proceedings of the 11th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU-06), Paris, pp. 2238–2245, 2006.

[6] F. Bobillo and U. Straccia, "fuzzyDL: An Expressive Fuzzy Description Logic Reasoner", Proceedings of the IEEE International Conference on Fuzzy System, Hong Kong, pp. 923-930, 2008.

[7] U. Straccia and N. Madrid, "A top-k query answering procedure for fuzzy logic programming", Fuzzy Sets and Systems, vol.205, pp.1-29, 2012.

Keerachart Suksut is currently a master student with the School of Computer Engineering, Suranaree University of Technology, Thailand. He received his bachelor degree in Computer Engineering from Suranaree University of Technology in 2011. His current research includes top-k ranking with datalog.

Pasapitch Chujai is currently a doctoral student with the School of Computer Engineering, Suranaree University of Technology, Thailand. She received her bachelor degree in Computer Science from Ramkhamhaeng University (RU), Thailand, in 2000, and master degree in Computer and Information Technology from King Mongkut's University of Technology Thonburi (KMUTT), Thailand, in 2004. Her current research includes Ontology, Recommendation System.

Nittaya Kerdprasop is an associate professor at the School of Computer Engineering, Suranaree University of Technology, Thailand. She received her bachelor degree in Radiation Techniques from Mahidol University, Thailand, in 1985, master degree in Computer Science from the Prince of Songkla University, Thailand, in 1991 and doctoral degree in Computer Science from Nova Southeastern University, U.S.A, in 1999. She is a member of ACM and IEEE Computer Society. Her research of interest includes Knowledge Discovery in Databases, Artificial Intelligence, Logic Programming, and Intelligent Databases.

Kittisak Kerdprasop is an associate professor and chair of the School of Computer Engineering, Suranaree University of Technology, Thailand. He received his bachelor degree in Mathematics from Srinakarinwirot University, Thailand, in 1986, master degree in Computer Science from the Prince of Songkla University, Thailand, in 1991 and doctoral degree in Computer Science from Nova Southeastern University, U.S.A., in 1999. His current research includes Data mining, Artificial Intelligence, Functional and Logic Programming Languages, Computational Statistics.

ประวัติผู้เขียน

นายกีระชาติ สุขสุทธิ์ เกิดเมื่อวันที่ 12สิงหาคม พ.ศ. 2533 ที่ จังหวัดนครราชสีมา เริ่มเข้าศึกษาระดับชั้นอนุบาล 1 ถึงชั้นประถมศึกษาปีที่ 6 ที่โรงเรียนประชารัฐสามัคคี อำเภอสูงเนิน จังหวัดนครราชสีมา จากนั้นได้เข้าศึกษาต่อในระดับมัธยมศึกษาตอนต้นและตอนปลาย ที่โรงเรียนสูงเนิน อำเภอสูงเนิน จังหวัดนครราชสีมา ปีการศึกษา 2551 ได้เข้าศึกษาต่อระดับปริญญาตรีในสาขาวิชาวิศวกรรมคอมพิวเตอร์ สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรานารี และสำเร็จการศึกษาเมื่อปี พ.ศ. 2554 ภายหลังสำเร็จการศึกษาในระดับปริญญาตรี ได้เข้าทำงานเป็นลูกจ้างชั่วคราวในบริษัทการบินไทย เมื่อปี พ.ศ. 2554 หลังจากนั้นได้เข้าศึกษาในระดับปริญญาโท สาขาวิชาวิศวกรรมคอมพิวเตอร์ สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรานารี ในปี 2556

ในระหว่างการศึกษาได้รับความอนุเคราะห์อย่างดียิ่งจากอาจารย์ประจำวิชา Database System ได้รับความไว้วางใจให้เป็นผู้ช่วยสอนปฏิบัติการ ได้รับการตีพิมพ์เผยแพร่บทความวิชาการ ซึ่งรายละเอียดสามารถดูได้ที่ภาคผนวก ข

รายละเอียดการนำเสนอผลงาน และรางวัลที่เคยได้รับ

- ได้รับรางวัลนำเสนอผลงานดีเด่นในงาน “ประชุมสัมมนาวิชาการและการเสนอผลงานวิจัยระดับชาติและนานาชาติ” จัดที่มหาวิทยาลัยราชภัฏนครราชสีมา จังหวัดนครราชสีมา ในหัวข้อ “การตอบคำถามด้วยเทคนิคการจัดเรียง เค อันดับแรกแบบคลุมเครือในฐานข้อมูลนิรนัย”
- เข้าร่วมนำเสนอผลงานในงาน “ประชุมวิชาการระดับชาติด้านเทคโนโลยีสารสนเทศครั้งที่ 6”จัดที่ไร่ทองสมบูรณ์คลับ จังหวัดนครราชสีมาในหัวข้อ “การตอบคำถามด้วยภาษาที่มีการควบคุม”
- เข้าร่วมนำเสนอผลงานในงาน “2014 The 6th International Conference on Computer Engineering and Technology” จัดที่กรุงโซล ประเทศเกาหลีใต้ ในหัวข้อ “Top-k Ranking with Membership Function for Deductive Database”