โครงการวิจัยประจำปีงบประมาณเงินรายได้ 2548

เรื่อง
การศึกษาการทำเท็กโปรเซสซิ่งโคโฮเนนนิวรอลเน็ตเวิร์ค
โดยการใช้กฎระหว่างสองข้อมูล

หัวหน้าโครงการวิจัย

รศ. ดร.วรพจน์ กรีสุระเดช

Assoc. Professor Dr. Worapoj Kreesuradej

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2549

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ I

สารบัญ III

สารบัญตาราง IV

สารบัญรูป V

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญของปัญหา 1

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา 2

1.3 สมมติฐานของการวิจัย 3

1.4 ขอบเขตของการวิจัย 3

1.5 ขั้นตอนของการศึกษา 3

บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 4

2.1 โครงข่ายประสาทเทียม 4

2.2 การจัดกลุ่มข้อมูล 6

2.2.1 เทคนิคการจัดกลุ่มข้อมูล 7

2.2.2 การจัดกลุ่มข้อมูลโดยการใช้กฎระหว่างสองข้อมูล 8

2.3 เซ็ลฟออร์แกไนซิงแม็ป 10

2.4 การเปรียบเทียบความแตกต่างกันของเอกสาร 12

2.5 เท็กโปรเซสซิงโคโฮเนนนิวรอลเน็ตเวิร์ค 16

บทที่ 3 การจัดกลุ่มเอกสารโดยใช้โคโฮเนนนิวรอลเน็ตเวิร์คร่วมกับกฎระหว่างสองข้อมูล 18

3.1 การจัดกลุ่มเอกสารโดยใช้โคโฮเนนนิวรอลเน็ตเวิร์คร่วมกับกฎระหว่างสองข้อมูล 19

3.2 ขั้นตอนการทำงานของอัลกอริทึม 19

3.2.1 ส่วนของขั้นตอนการเรียนรู้แบบแข่งขัน 21

3.2.2 ส่วนของการหานิวรอลโหนดใกล้เคียง 22

สารบัญ (ต่อ)

หน้า

3.2.3 ส่วนของการปรับค่าเวทเวกเตอร์ 22
3.3 ตัวอย่างการทำงานของอัลกอริทึม 23

บทที่ 4 การทดลองและผลการทดลอง 29
4.1 การวัดประสิทธิภาพของอัลกอริทึม 29
4.1.1 F Measure 29
4.1.2 Entropy 30
4.2 ข้อมูลที่ใช้ในการทดลอง 31
4.2.1 ข้อมูลชุดตัวอักษร 31
4.2.2 ข้อมูลข่าว Reuters-21578 34
4.3 ผลการทดลอง 35
4.3.1 ข้อมูลชุดตัวอักษร 35
4.3.2 ข้อมูลข่าว Reuters-21578 36
4.4 สรุปผลการทดลอง 38

บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ 40
5.1 สรุปผลการวิจัย 40
5.2 ปัญหาที่พบในงานวิจัยนี้ 41
5.3 แนวทางการพัฒนาในอนาคต 41

เอกสารอ้างอิง 42
ภาคผนวก ก. รายชื่อผลงานวิจัยที่ได้รับการตีพิมพ์ 44

สารบัญตาราง

ตารางที่ หน้า

2.1 แสดงกฎเพิ่มเติมหลังจากการขยายโดยอาศัยคุณสมบัติการถ่ายทอด....9

2.2 แสดงข้อมูลตัวอย่างของเอกสารข่าว....13

3.1 แสดงรายละเอียดคุณสมบัติของเอกสารตัวอย่าง....23

4.1 แสดงตัวอย่างข้อมูลที่ใช้ในการเทรนนิง....32

4.2 แสดงข้อมูลของแต่ละกฎ....33

4.3 แสดงรายละเอียดของชุดข้อมูลที่นำมาใช้ในการทดสอบความถูกต้องของโมเดล....34

4.4 แสดงชุดข้อมูล 3 กลุ่มข่าว Reuters-21578 สำหรับใช้เทรนนิงและทดสอบของข้อมูล....34

4.5 แสดงชุดข้อมูล 5 กลุ่มข่าว Reuters-21578 สำหรับใช้เทรนนิงและทดสอบของข้อมูล....35

4.6 แสดงผลลัพธ์ที่ได้จากการเทรนนิงข้อมูลชุดตัวอักษร....35

4.7 แสดงผลลัพธ์ที่ได้จากการทดสอบโมเดลด้วยชุดข้อมูลตัวอักษร....36

สารบัญรูป

รูปที่ หน้า

2.1 แสดงโครงสร้างนิวรอลของเซลล์ประสาท 4
2.2 แสดงโครงสร้างพื้นฐานของโครงข่ายประสาทเทียม 5
2.3 Sigmoid Transfer Function 5
2.4 แสดงขั้นตอนการจัดกลุ่มข้อมูล 7
2.5 แสดงโครงสร้างต้นไม้ของการจัดกลุ่มข้อมูลแบบ Hierarchical Clustering 8
2.6 แสดงการทำงานการหาโหนดใกล้เคียง 11
2.7 แสดงการเรียนรู้ของอัลกอริทึมเซ็ลฟออร์แกไนซิงแม็ป 11
2.8 แสดงโครงสร้างของเท็กโปรเซสซิงโคโฮเนนนิวรอลเน็ตเวิร์ค 16
3.1 แสดงขั้นตอนการทำงานของการจัดกลุ่มเอกสารโดยใช้โคโฮเนนนิวรอลเน็ตเวิร์ค
ร่วมกับกฎระหว่างสองข้อมูล 20
3.2 แสดงโครงข่ายที่ใช้อธิบายตัวอย่างการทำงานของอัลกอริทึม 24
3.3 แสดงโครงข่ายที่ใช้อธิบายตัวอย่างการทำงานของอัลกอริทึมพิจารณานิวรอลโหนดที่ 1 25
3.4 แสดงโครงข่ายที่ใช้อธิบายตัวอย่างการทำงานของอัลกอริทึมพิจารณานิวรอลโหนดที่ 2 27
4.1 แสดงกลุ่มตัวอักษรที่ใช้สร้างข้อมูลในคุณสมบัติหัวเรื่อง 31
4.2 แสดงกลุ่มตัวอักษรที่ใช้สร้างข้อมูลในคุณสมบัติคำสำคัญ 32
4.3 แสดงค่าของ F Measure ในการทดลองกับข้อมูล 3 กลุ่ม 37
4.4 แสดงค่าของ Entropy ในการทดลองกับข้อมูล 3 กลุ่ม 37
4.5 แสดงค่าของ F Measure ในการทดลองกับข้อมูล 5 กลุ่ม 38
4.6 แสดงค่าของ Entropy ในการทดลองกับข้อมูล 5 กลุ่ม 38

บทคัดย่อ

ลักษณะการจัดกลุ่มแบบไม่มีการชี้นำนั้นจะทำงานได้ดีเมื่อข้อมูลแยกออกจากกันอย่างชัดเจน แต่ข้อมูลในปัจจุบันนั้นมีลักษณะแยกออกจากกันไม่ชัดเจนนัก จึงทำให้การจัดกลุ่มแบบไม่มีการชี้นำแบบเดิมนั้นมีประสิทธิภาพไม่เพียงพอเท่าที่ควร การจัดกลุ่มเอกสารโดยใช้โคโฮเนนนิวรอลเน็ตเวิร์คร่วมกับกฎระหว่างสองข้อมูลเป็นการพัฒนาปรับปรุงเท็กโปรเซสซิงโคโฮเนนนิวรอลเน็ตเวิร์ค ซึ่งเป็นเทคนิคการจัดกลุ่มแบบไม่มีการชี้นำให้มีประสิทธิภาพมากขึ้น โดยนำแนวคิดของโคโฮเนนเซ็ลฟออร์แกไนซิงแม๊ป มาปรับปรุงความสามารถเพื่อทำการจัดกลุ่มข้อมูลประเภทข้อความได้โดยตรงโดยทำการประยุกต์แนวคิดเรื่องการเปรียบเทียบความแตกต่างของข้อมูลเชิงสัญลักษณ์ รวมทั้งนำแนวคิดการจัดกลุ่มข้อมูลโดยใช้กฎระหว่างสองข้อมูลมาใช้ในการจัดกลุ่มข้อมูลประเภทข้อความด้วย แนวคิดการจัดกลุ่มเอกสารที่นำเสนอนี้สามารถจัดกลุ่มข้อมูลประเภทข้อความได้อย่างถูกต้อง และมีประสิทธิภาพ

ABSTRACT

Unsupervised clustering obtains a good performance when data are completely separated. However, most data are incompletely separated. As a result unsupervised clustering gives poor performance, when they are applied to such data. This paper proposes document clustering using Kohonen neural network with pairwise constraints to improve the text processing Kohonen neural network. This algorithm works directly on textual information without mapping document into some representation which has quantitative features. The input level of the proposed neural network can directly receive a qualitative value without mapping the qualitative value into numerical value. The proposed neural network is based on the architecture of text processing Kohonen neural network, the concepts of dissimilarity measure of symbolic objects and pairwise constrained concepts. As a result, the model can successfully assign cluster label to the objects.

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

การจัดกลุ่มข้อมูล (Clustering) มีวัตถุประสงค์หลัก คือ แบ่งกลุ่มข้อมูลที่มีลักษณะคล้ายกันให้อยู่ในกลุ่มเดียวกัน โดยที่ข้อมูลภายในกลุ่มเดียวกันนั้นจะมีความเหมือนกันมากกว่าข้อมูลที่อยู่ต่างกลุ่มกัน ข้อมูลที่มีการนำมาจัดกลุ่มในปัจจุบันนั้นมีทั้งที่เป็นการจัดกลุ่มข้อมูลเชิงตัวเลข และการจัดกลุ่มเอกสารหรือการจัดกลุ่มข้อมูลประเภทข้อความ ในการจัดกลุ่มข้อมูลมีสมมติฐานเริ่มต้นคือไม่ทราบเลยว่าข้อมูลหรือเอกสารที่นำมาจัดกลุ่มนั้นแบ่งได้เป็นกี่กลุ่ม และในแต่ละกลุ่มประกอบด้วยข้อมูลหรือเอกสารใดบ้าง ตัวอย่างเช่น สมมติว่ามีเอกสารข่าวจำนวนหนึ่ง นำคุณลักษณะ (Feature) หรือค่าคุณสมบัติที่สามารถแยกแยะเอกสารออกจากกันได้ เช่น หัวเรื่องข่าวและคำสำคัญของข่าว มาเป็นข้อมูลอินพุตให้กับอัลกอริทึมสำหรับการจัดกลุ่มเอกสาร ผลลัพธ์ที่ได้หลังจากสิ้นสุดการทำงานของอัลกอริทึม คือเอกสารจะถูกแบ่งออกเป็นกลุ่มๆ หลักการของการจัดกลุ่มข้อมูลทั่วๆ ไป คือการเลือกคุณสมบัติบางอย่างของข้อมูลซึ่งสามารถแยกแยะข้อมูลออกจากกันได้มาเป็นข้อมูลอินพุตให้กับอัลกอริทึมสำหรับการจัดกลุ่มข้อมูล โดยอัลกอริทึมนี้จะนำคุณสมบัติที่เป็นอินพุตมาใช้เพื่อทำการพิจารณาว่าแต่ละข้อมูลนั้นมีความเหมือนหรือต่างกันมากน้อยเพียงใด

การจัดกลุ่มข้อมูลแบบกึ่งชี้นำ (Semi-supervised Clustering) เป็นวิธีการจัดกลุ่มข้อมูลวิธีหนึ่งที่กำลังอยู่ในความสนใจในขณะนี้ วิธีการจัดกลุ่มข้อมูลนี้เป็นการนำข้อมูลบางตัวอย่างที่รู้กลุ่มแน่นอน หรือมีการนำกฎระหว่างสองข้อมูล (Pairwise Constraints) [1] และ [2] ของบางตัวอย่างมาช่วยในการจัดกลุ่มข้อมูลแบบไม่มีการชี้นำแบบเดิมที่มีอยู่ (Unsupervised Clustering) แนวคิดของกฎระหว่างสองข้อมูลประกอบด้วยกฎที่เป็น Must-link Constraints และ Cannot-link Constraints โดย Must-link Constraints เป็นกฎที่บอกว่าข้อมูลสองข้อมูลควรอยู่ในกลุ่มเดียวกัน ส่วน Cannot-link Constraints เป็นกฎที่บอกว่าข้อมูลสองข้อมูลควรอยู่ต่างกลุ่มกัน ตัวอย่างหนึ่งสำหรับการจัดกลุ่มแบบกึ่งชี้นำ คือการนำกฎระหว่างสองข้อมูลมาทำงานร่วมกับการจัดกลุ่มข้อมูลแบบไม่มีการชี้นำ K-Means [3] และ [4] ซึ่งเป็นการจัดกลุ่มข้อมูลที่เป็นข้อมูลเชิงตัวเลข การจัดกลุ่มข้อมูลแบบกึ่งชี้นำที่มีการนำกฎระหว่างสองข้อมูลมาทำงานร่วมกับการจัดกลุ่มข้อมูลแบบไม่มีการชี้นำจะเป็นการจัดกลุ่มกับข้อมูลเชิงตัวเลขเป็นส่วนใหญ่

ในขณะที่การจัดกลุ่มข้อมูลเชิงตัวเลขนั้นได้มีการนำเทคนิควิธีการต่างๆ มาใช้หลายวิธีเทคนิคหนึ่ง คือเซ็ลฟออร์แกไนซิงแม็ป [5] และ [6] ซึ่งเป็นนิวรอลเน็ตเวิร์คชนิดหนึ่งที่สามารถจัดกลุ่มข้อมูลเชิงตัวเลขได้เป็นอย่างดี แม้ว่าการจัดกลุ่มข้อมูลที่มีอยู่ในปัจจุบันจะสามารถจัดกลุ่ม

ข้อมูลประเภทข้อความได้ แต่จะต้องทำการแปลงข้อมูลให้อยู่ในรูปแบบเชิงตัวเลขก่อน ซึ่งอาจจะทำให้สูญเสียความหมายของข้อมูลได้ อีกทั้งยังทำให้เสียเวลาค่อนข้างสูงในการแปลงข้อมูลให้อยู่ในรูปแบบเชิงตัวเลข จากงานวิจัย [7] เป็นการจัดกลุ่มข้อมูลประเภทข้อความที่นำเทคนิคการเรียนรู้ของนิวรอลเน็ตเวิร์คมาใช้ เทคนิคดังกล่าวเป็นการจัดกลุ่มข้อมูลแบบไม่มีการชี้นำ ซึ่งการทำงานของการจัดกลุ่มข้อมูลแบบไม่มีการชี้นำจะทำงานได้ดีและมีประสิทธิภาพเมื่อลักษณะของข้อมูลมีการแยกออกจากกันอย่างชัดเจน อย่างไรก็ตามเนื่องจากลักษณะข้อมูลส่วนใหญ่ในปัจจุบันแยกออกจากกันไม่ชัดเจนนัก เพื่อให้การจัดกลุ่มข้อมูลมีประสิทธิภาพมากขึ้น จึงได้มีการนำความคิดการจัดกลุ่มข้อมูลแบบกึ่งชี้นำมาทำการพัฒนาอัลกอริทึมที่สามารถทำการจัดกลุ่มข้อมูลที่แยกออกจากกันไม่ชัดเจนได้มีประสิทธิภาพมากยิ่งขึ้น เทคนิคของการจัดกลุ่มแบบกึ่งชี้นำนั้นมีอยู่หลายเทคนิคด้วยกัน เทคนิคหนึ่งคืออาศัยพื้นความรู้ซึ่งมีอยู่หลายรูปแบบด้วยกัน ตัวอย่างรูปแบบของกฎเช่น Global Constraints, Cluster-level Constraints, Feature-level Constraints และ Instance-level Constraints [8]

การจัดกลุ่มเอกสารโดยใช้โคโฮเนนนิวรอลเน็ตเวิร์คร่วมกับกฎระหว่างสองข้อมูลเป็นการนำแนวคิดของเซ็ลฟออร์แกไนซิงแม็ปมาขยายความสามารถเพื่อทำการจัดกลุ่มข้อมูลประเภทข้อความได้โดยตรง โดยได้มีการประยุกต์แนวคิดเรื่องการเปรียบเทียบความแตกต่างของข้อมูลเชิงสัญลักษณ์ [9] และ [10] รวมทั้งนำแนวคิดการจัดกลุ่มข้อมูลโดยใช้กฎระหว่างสองข้อมูลซึ่งเป็นหนึ่งรูปแบบของกฎประเภท Instance-level Constraints มาใช้ในการจัดกลุ่มข้อมูลประเภทข้อความด้วย แนวคิดการจัดกลุ่มเอกสารที่นำเสนอนี้สามารถจัดกลุ่มข้อมูลประเภทข้อความได้อย่างถูกต้องและมีประสิทธิภาพเป็นอย่างดี

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

วัตถุประสงค์ของการศึกษานี้ เพื่อพัฒนาปรับปรุงการทำงานของเท็กโปรเซสซิงโคโฮเนนนิวรอลเน็ตเวิร์ค [7] ให้มีประสิทธิภาพมากขึ้น โดยศึกษาลักษณะการทำงานของการการจัดกลุ่มข้อมูล แนวคิดของเท็กโปรเซสซิงโคโฮเนนนิวรอลเน็ตเวิร์ค และลักษณะการทำงานของโคโฮเนนเซ็ลฟออร์แกไนซิงแม็ป รวมถึงการศึกษาลักษณะการทำงานของการจัดกลุ่มข้อมูลโดยใช้กฎระหว่างสอง และศึกษาแนวคิดการหาค่าความแตกต่างของข้อมูลเชิงสัญลักษณ์เพื่อที่จะนำมาประยุกต์ใช้ในการพัฒนาอัลกอริทึมสำหรับการจัดกลุ่มข้อมูลประเภทข้อความได้โดยตรง โดยไม่จำเป็นต้องทำการแปลงข้อมูลนี้ให้อยู่ในรูปแบบเชิงตัวเลขเช่นเดียวกับการจัดกลุ่มข้อมูลประเภทข้อความทั่วไปที่มีอยู่ นอกจากนั้นโครงการวิจัยนี้ยังมีวัตถุประสงค์ในการเผยแพร่ให้ความรู้ทางด้านเทคโนโลยีสารสนเทศชั้นสูงให้แก่สังคม

1.3 สมมติฐานของการวิจัย

1.3.1 เอกสารที่ใช้ในการศึกษาข่าว Reuters-21578 ซึ่งเป็นข่าวที่มีเนื้อข้อความเป็นภาษาอังกฤษ

1.3.2 เอกสารข่าวที่นำมาศึกษาต้องมีครบสมบูรณ์ทั้งหัวเรื่องข่าว (Topic) หัวข้อข่าว (Title) และ เนื้อข่าว (Body) กลุ่มของข้อมูลจะแยกกลุ่มตามหัวเรื่องของข่าว

1.3.3 ตัวอย่างเอกสารข่าวที่นำมาศึกษาเป็นเอกสารข่าวที่ทราบกลุ่มแน่นอนเพื่อนำไปสร้างเป็นกฎเพื่อช่วยในการจัดกลุ่มข้อมูล

1.4 ขอบเขตของการวิจัย

งานวิจัยนี้เสนออัลกอริทึมที่ใช้ในการจัดกลุ่มข้อมูลประเภทข้อความโดยตรง โดยที่อัลกอริทึมนี้จะรับข้อมูลอินพุตที่เป็นข้อความภาษาอังกฤษเท่านั้น การทำงานของอัลกอริทึมนี้สามารถจัดกลุ่มข้อมูลประเภทข้อความได้โดยตรงไม่ต้องทำการแปลงข้อมูลให้อยู่ในรูปของข้อมูลเชิงตัวเลขก่อน อัลกอริทึมนี้ได้มีการประยุกต์แนวคิดในการหาค่าความแตกต่างของข้อมูลเชิงสัญลักษณ์เข้ามาประยุกต์ใช้กับโคโฮเนนเซ็ลฟ์ออร์แกไนซิงแม็ป รวมทั้งได้นำแนวคิดการจัดกลุ่มข้อมูลแบบกึ่งการชี้นำโดยการใช้กฎระหว่างสองข้อมูลมาใช้ในการจัดกลุ่มข้อมูลประเภทข้อความโดยตรง และงานวิจัยนี้ครอบคลุมเฉพาะการจัดกลุ่มเอกสารข่าว Reuters-21578 เท่านั้น

1.5 ขั้นตอนของการศึกษา

1.5.1 ศึกษาทฤษฎีและแนวคิดจากบทความต่างๆ ที่เกี่ยวข้องกับงานวิจัย

1.5.2 ศึกษาปัญหาในการจัดกลุ่มข้อมูลประเภทข้อความ

1.5.3 เขียนโปรแกรมเพื่อทำการจัดกลุ่มข้อมูลประเภทข้อความ

1.5.4 ทดสอบโปรแกรมกับข้อมูลตัวอย่าง พร้อมทั้งแก้ไขข้อผิดพลาดของโปรแกรม

1.5.5 รวบรวมผลการทดลองที่ได้จากการทำงานของโปรแกรม

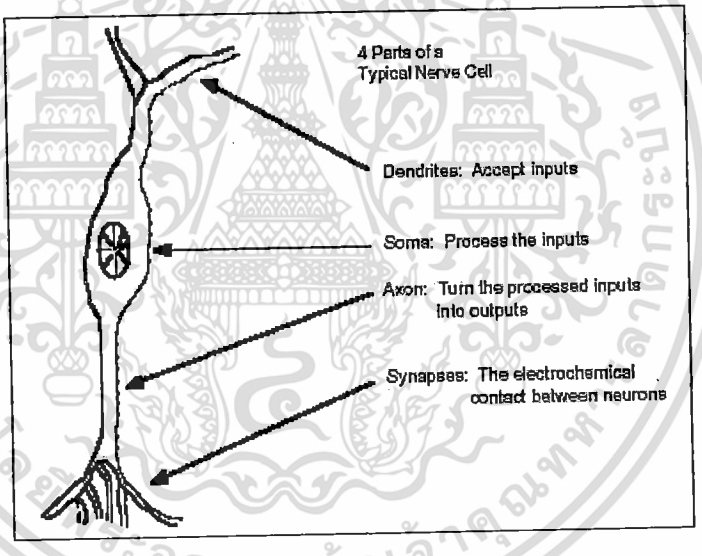
1.5.6 วิเคราะห์และสรุปผลการดำเนินการ

1.5.7 จัดทำเอกสาร

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

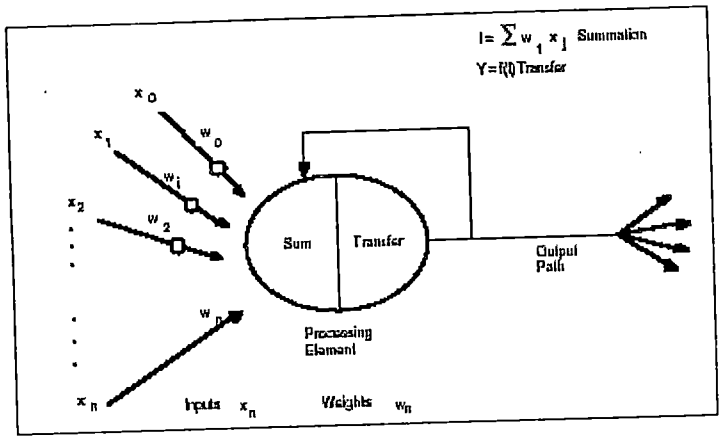
2.1 โครงข่ายประสาทเทียม (Artificial Neural Network)

โครงข่ายประสาทในสมองมนุษย์มีลักษณะที่ซับซ้อนมาก สามารถเรียนรู้และคิดหาเหตุผลได้ ภายในสมองมนุษย์มีเซลล์ประสาทที่เรียกว่า นิวรอล (Neural) ลักษณะโครงสร้างของแต่ละนิวรอลประกอบด้วยตัวเซลล์ (Soma) แกนของเซลล์ (Axon) กิ่งก้านของเซลล์ (Dendrites) และเส้นเชื่อมต่อเซลล์ (Synapses) [11] การทำงานของเซลล์ประสาทเกิดจากการที่แต่ละนิวรอลรับสัญญาณจากนิวรอลตัวอื่นๆ ผ่านกิ่งก้านของเซลล์ และส่งสัญญาณที่เกิดขึ้นโดยตัวเซลล์ของมันเองไปตามแกนของเซลล์ โดยสัญญาณนี้จะส่งผ่านไปยังจุดปลายเส้นเชื่อมต่อเซลล์ หรือจุดเชื่อมต่อระหว่างนิวรอล ลักษณะโครงสร้างนิวรอลของเซลล์ประสาทแสดงดังรูปที่ 2.1



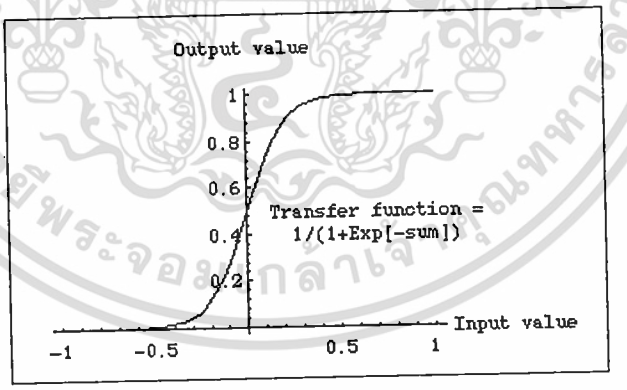
รูปที่ 2.1 แสดงโครงสร้างนิวรอลของเซลล์ประสาท

โครงข่ายประสาทเทียมเป็นการจำลองการทำงานของเซลล์สมองมนุษย์ ลักษณะโครงสร้างพื้นฐานของเครือข่ายประสาทเทียมประกอบด้วย ส่วนรับข้อมูลเข้า หรือข้อมูลอินพุต ส่วนส่งข้อมูลออก หรือข้อมูลเอาท์พุต และในแต่ละส่วนข้อมูลอินพุตจะประกอบด้วยค่าถ่วงน้ำหนัก ลักษณะโครงสร้างพื้นฐานของโครงข่ายประสาทเทียมแสดงดังรูปที่ 2.2



รูปที่ 2.2 แสดงโครงสร้างพื้นฐานของโครงข่ายประสาทเทียม

จากรูปที่ 2.2 โครงข่ายประสาทเทียมนี้จะรับข้อมูลอินพุต (x_1, …, x_n) เข้ามาสู่ขั้นต้อนการประมวลผลเบื้องต้น (Processing Element) ในแต่ละข้อมูลอินพุตจะมีค่าน้ำหนัก (w_1 ,… ,w_n) เพื่อนำมาใช้ในกระบวนการประมวลผลเบื้องต้น โดยในที่นี้การประมวลผลเบื้องต้น คือการหาผลรวมของข้อมูลอินพุต หลังจากที่หาผลรวมของข้อมูลอินพุตเรียบร้อยแล้วจะทำการแปลงค่าที่ได้โดยผ่านฟังก์ชันการแปลง (Transfer Function) เพื่อที่จะให้ได้มาซึ่งผลลัพธ์หรือเอาท์พุตนั่นเอง ตัวอย่างของฟังก์ชันการแปลงคือ Sigmoid Function การทำงานของฟังก์ชันนี้จะนำค่าที่ได้จากฟังก์ชันการหาผลรวมโดยในที่นี้เรียกว่าฟังก์ชัน Sum จากรูปที่ 2.3 ค่าที่ได้รับจากฟังก์ชันการแปลง Sigmoid Function นี้จะมีค่าอยู่ระหว่าง 0 ถึง 1



รูปที่ 2.3 Sigmoid Transfer Function

การเรียนรู้ของโครงข่ายประสาทเทียมจะมีประสิทธิภาพมากน้อยเพียงใดนั้น ทั้งนี้ขึ้นอยู่กับค่าถ่วงน้ำหนักของเครือข่าย โดยในการฝึกสอน (Training) โครงข่ายนั้นเป็นการหาค่าถ่วงน้ำหนักที่

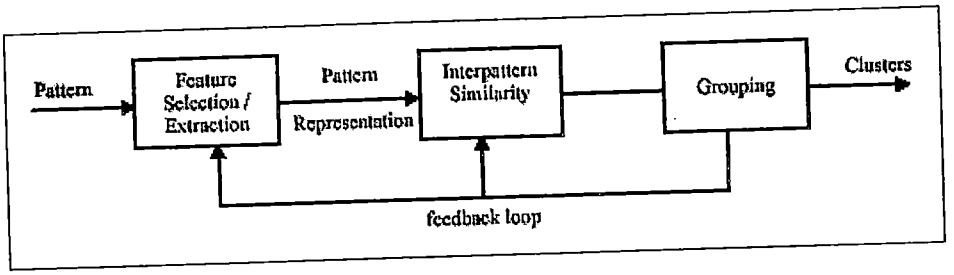
เหมาะสมให้กับโครงข่าย ในการฝึกสอนนี้เป็นการทำงานแบบวนซ้ำ ซึ่งในแต่ละรอบจะมีการปรับค่าถ่วงน้ำหนักเพื่อให้ได้ค่าถ่วงน้ำหนักที่เหมาะสม ซึ่งค่าถ่วงน้ำหนักที่เหมาะสมนี้ทำให้ผลของข้อมูลเอาท์พุตที่ได้จากโครงข่ายตรงกับข้อมูลเอาท์พุตที่ต้องการมากที่สุด โครงข่ายประสาทเทียมหากแบ่งตามลักษณะของการเรียนรู้จะสามารถแบ่งได้ออกเป็น 2 ประเภทคือ

1. การเรียนรู้แบบมีการชี้นำ (Supervised Learning) การสอนโครงข่ายวิธีนี้ทำโดยกำหนดเซตของอินพุตและเอาท์พุตที่ต้องการ เมื่อป้อนอินพุตเข้าไปให้กับโครงข่ายแล้วโครงข่ายจะทำการประมวลผลลัพธ์ออกมาพร้อมกับค่าถ่วงน้ำหนักชุดหนึ่ง ผลลัพธ์ที่ได้จากโครงข่ายจะนำมาคำนวณหาค่าความผิดพลาด การคำนวณหาค่าความผิดพลาดนี้ทำได้จากการหาผลต่างระหว่างค่าข้อมูลเอาท์พุตที่ได้จากโครงข่ายและค่าข้อมูลอินพุตที่ต้องการ ถ้าหากว่ายังมีความผิดพลาดสูงอยู่ก็จะมีการปรับค่าถ่วงน้ำหนัก และทำการสอนต่อไปจนกว่าค่าความผิดพลาดที่คำนวณได้มีค่าน้อยพอที่สามารถยอมรับได้จึงจะหยุดการสอน ตัวอย่างโครงข่ายที่มีการเรียนรู้แบบมีการชี้นำเช่น Backpropagation และ Perceptron เป็นต้น

2. การเรียนรู้แบบไม่มีการชี้นำ (Unsupervised Learning) การสอนโครงข่ายด้วยการป้อนอินพุตเข้าสู่โครงข่ายอย่างต่อเนื่องเพียงอย่างเดียวไม่มีการส่งค่าผลลัพธ์ให้กับอินพุตแต่ละตัว ภายในโครงข่ายจะมีเอาท์พุตโหนดอยู่ โดยแต่ละเอาท์พุตโหนดแทนกลุ่มของข้อมูลที่มีคุณสมบัติเหมือนกัน ค่าน้ำหนักของแต่ละอินพุตจะนำมาคำนวณเปรียบเทียบกับค่าน้ำหนักของโครงข่าย โดยที่ข้อมูลอินพุตที่มีความใกล้เคียงกับเอาท์พุตโหนดใดก็จะเป็นข้อมูลในกลุ่มนั้น การเรียนรู้แบบไม่มีการชี้นำนี้จะไม่สามารถระบุกลุ่มของข้อมูลได้ว่าแต่ละเอาท์พุตโหนดนั้นเป็นข้อมูลกลุ่มใด ซึ่งจะแตกต่างจากการเรียนรู้แบบมีการชี้นำซึ่งสามารถระบุกลุ่มของแต่ละเอาท์พุตโหนดได้อย่างแน่นอน ตัวอย่างโครงข่ายที่มีการเรียนรู้แบบไม่มีการชี้นำเช่น Adaptive Resonance Theory Neural Network (ART) และ Self-Organizing Map (SOM) เป็นต้น

2.2 การจัดกลุ่มข้อมูล

การจัดกลุ่มข้อมูล เป็นการแบ่งแยกข้อมูลออกเป็นกลุ่มย่อยๆ หรือคลัสเตอร์ตามลักษณะความเหมือนของข้อมูล โดยข้อมูลที่มีความเหมือนกันจะถูกจัดให้อยู่ในกลุ่มเดียวกัน ซึ่งข้อมูลที่อยู่ในกลุ่มเดียวกันนั้นจะมีความเหมือนกันมากกว่าข้อมูลที่อยู่ต่างกลุ่มกัน องค์ประกอบของงานทางด้านการจัดกลุ่มข้อมูลโดยทั่วไปแล้วประกอบด้วยขั้นตอนต่างๆ ดังรูปที่ 2.4



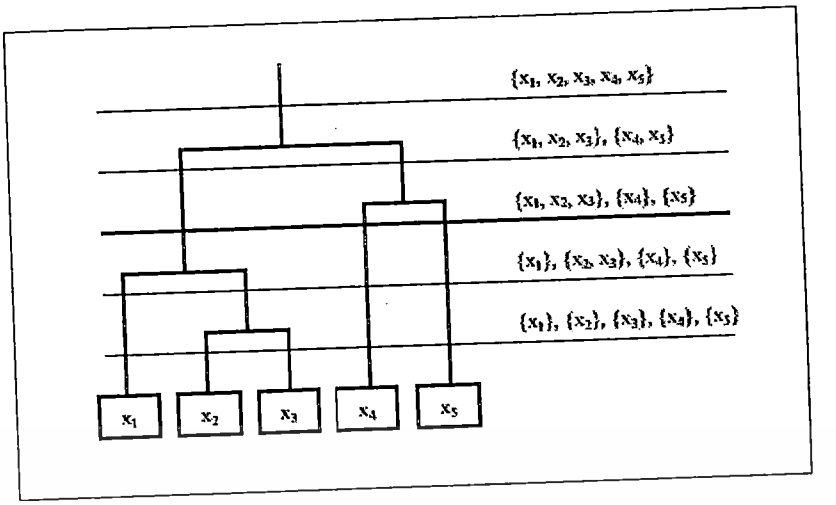
รูปที่ 2.4 แสดงขั้นตอนการจัดกลุ่มข้อมูล

จากรูปที่ 2.4 ขั้นตอนการจัดกลุ่มข้อมูลเริ่มต้นเมื่อมีรูปแบบของข้อมูล (Pattern) จำนวนหนึ่ง นำข้อมูลเหล่านั้นมาเลือกคุณลักษณะ (Feature) ที่สามารถแยกข้อมูลออกจากกันได้เพื่อเป็นตัวแทนของแต่ละข้อมูล (Feature Selection/Extraction) ซึ่งในการแทนรูปแบบของข้อมูลนั้นมีแตกต่างกันออกไปขึ้นอยู่กับการนำไปใช้ในอัลกอริทึม หลังจากที่ทราบคุณลักษณะของแต่ละข้อมูลแล้ว การดำเนินการขั้นตอนต่อไปของการจัดกลุ่มข้อมูล คือการคำนวณหาความสัมพันธ์ระหว่างข้อมูล (Interpatern Similarity) โดยทั่วไปแล้วการคำนวณหานี้หาจากฟังก์ชันการวัดระยะห่างระหว่างข้อมูลซึ่งมีอยู่หลากหลาย ตัวอย่างของฟังก์ชันการวัดระยะห่างระหว่างข้อมูล คือ Euclidean Distance เมื่อหาค่าความสัมพันธ์ระหว่างข้อมูลได้แล้วก็เข้าสู่ขั้นตอนการจัดกลุ่มข้อมูล (Grouping) โดยที่ข้อมูลที่มีค่าระยะห่างน้อยๆ จะอยู่ในกลุ่มเดียวกัน ส่วนข้อมูลที่มีค่าระยะห่างต่างกันมากก็จะอยู่ต่างกลุ่มกัน จากนั้นสุดท้ายข้อมูลจะถูกแยกออกเป็นกลุ่มๆ หรือคลัสเตอร์ นอกจากนี้ในขั้นตอนของการจัดกลุ่มข้อมูลสามารถย้อนกลับ (Feedback Loop) ไปยังขั้นตอนของการเลือกคุณลักษณะและการหาค่าความสัมพันธ์ระหว่างข้อมูลได้ในลักษณะวนรอบอีกด้วย ทั้งนี้ขึ้นอยู่กับอัลกอริทึมที่นำมาใช้ในการจัดกลุ่มข้อมูล

2.2.1 เทคนิคการจัดกลุ่มข้อมูล

เทคนิคหลักๆ ของการจัดกลุ่มข้อมูลแบ่งออกได้เป็นสองแบบ คือ Hierarchical Clustering และ Partional Clustering ซึ่งแต่ละเทคนิคมีรายละเอียดดังนี้ [12]

1. เทคนิคการจัดกลุ่มข้อมูลแบบ Hierarchical Clustering เป็นการรวมกลุ่มข้อมูลตามระดับชั้น โดยเริ่มต้นจากข้อมูลทั้งหมดจะอยู่เพียงคลัสเตอร์เดียวในระดับบนสุด เมื่อผ่านขั้นตอนการจัดกลุ่มข้อมูลจะได้คลัสเตอร์ย่อยๆ จนกระทั่งได้คลัสเตอร์ที่มีข้อมูลเพียงชุดเดียวที่ระดับล่างสุด ในขั้นตอนสุดท้ายของการทำงานของอัลกอริทึมจะได้โครงสร้างต้นไม้ของกลุ่มข้อมูล (Dendrogram) ซึ่งแสดงถึงความสัมพันธ์ของกลุ่มข้อมูลว่ามีความสัมพันธ์กันอย่างไร โดยถ้าทำการตัดโครงสร้างต้นไม้ในระดับที่ต้องการข้อมูลก็จะแยกออกจากกันเป็นกลุ่มๆ ดังรูปที่ 2.5



รูปที่ 2.5 แสดงโครงสร้างต้นไม้ของการจัดกลุ่มข้อมูลแบบ Hierarchical Clustering

การสร้างลำดับชั้นของเทคนิคการจัดกลุ่มแบบ Hierarchical Clustering จำแนกออกเป็นสองวิธี วิธีแรกคือ Agglomerative จะเริ่มต้นจากข้อมูลอยู่ต่างคลัสเตอร์กัน ในแต่ละขั้นตอนการทำงานจะรวมข้อมูลที่มีความเหมือนกันหรือคล้ายคลึงให้อยู่ในคลัสเตอร์เดียวกัน การทำงานจะทำซ้ำไปเรื่อยๆ จนกระทั่งจำนวนของคลัสเตอร์มีค่าน้อยที่สุด วิธีที่สองคือ Divisive วิธีนี้เริ่มต้นข้อมูลทั้งหมดอยู่ในคลัสเตอร์เดียวกัน ในแต่ละขั้นตอนการทำงานจะทำการแยกออกเป็นคลัสเตอร์ย่อยๆ จนกระทั่งเหลือเพียงคลัสเตอร์เดียว สิ่งที่จำเป็นต้องพิจารณาในแต่ละขั้นตอนการทำงานของวิธี Divisive คือ คลัสเตอร์ใดควรที่จะต้องทำการแยก และการแยกข้อมูลออกจากคลัสเตอร์นั้นควรทำอย่างไร

2. เทคนิคการจัดกลุ่มข้อมูลแบบ Partitional Clustering จะมีลักษณะตรงกันข้ามกับการจัดกลุ่มแบบ Hierarchical Clustering ที่กล่าวมาข้างต้น กล่าวคือ การทำงานจะมีการระบุคลัสเตอร์ขึ้นมาจำนวนหนึ่งก่อน โดยข้อมูลทั้งหมดจะกระจายกันไปอยู่ในแต่ละคลัสเตอร์หลังจากนั้นอัลกอริทึมจะทำการปรับคลัสเตอร์ที่ซ้ำกันให้เหลือจำนวนของคลัสเตอร์ที่ซ้ำกันน้อยที่สุด การปรับคลัสเตอร์นี้ไม่ได้นำข้อมูลของคลัสเตอร์ที่ซ้ำกันมารวมกัน แต่ทำโดยการปรับเปลี่ยนโยกย้ายข้อมูลระหว่างคลัสเตอร์ที่มีความใกล้เคียงกัน ตัวอย่างของอัลกอริทึมแบบ Partitional Clustering ที่รู้จักกันดีคืออัลกอริทึมในกลุ่ม K-means

2.2.2 การจัดกลุ่มข้อมูลโดยการใช้กฎระหว่างสองข้อมูล

การจัดกลุ่มแบบกึ่งชี้นำเป็นการจัดกลุ่มข้อมูลที่มีการนำตัวอย่างที่ทราบกลุ่มแน่นอน หรือมีการนำกฎระหว่างสองข้อมูลบนบางตัวอย่างมาช่วยในการจัดกลุ่มแบบไม่ชี้นำที่มีอยู่ การจัดกลุ่มแบบกึ่งชี้นำนี้จะทำงานได้ดีเมื่อกลุ่มของข้อมูลแยกออกจากกันอย่างชัดเจน เทคนิคของการจัดกลุ่มแบบกึ่งชี้นำนั้นมีอยู่หลายเทคนิคด้วยกัน เทคนิคหนึ่งคืออาศัยพื้นความรู้ในรูปของกฎ หรือ

Constraints ซึ่งมีหลายประเภทตัวอย่างเช่น Global Constraints, Cluster-level Constraints, Feature-level Constraints และ Instance-level Constraints

กฎระหว่างสองข้อมูลเป็นหนึ่งในกฎประเภท Instance-level Constraints กฎระหว่างสองข้อมูลนี้แบ่งออกเป็น Must-link Constraints และ Cannot-link Constraints ซึ่งเป็นกฎที่บอกว่าข้อมูลสองข้อมูลควรที่จะอยู่ในกลุ่มเดียวกัน โดย Must-link Constraints เป็นกฎที่บอกว่าข้อมูลสองข้อมูลควรอยู่ในกลุ่มเดียวกัน และ Cannot-link Constraints เป็นกฎที่บอกว่าข้อมูลสองข้อมูลควรอยู่ต่างกลุ่มกัน กำหนดให้ M แทนเซตของสมาชิกใน Must-link Constraints ซึ่งเขียนแทนด้วย $(x_i, x_j) \in M$ กล่าวคือถ้า x_i อยู่ในกลุ่มที่ 1 แล้ว x_j ควรที่จะอยู่ในกลุ่มที่ 1 ด้วย และกำหนดให้ C แทนเซตของสมาชิกใน Cannot-link Constraints เขียนแทนด้วย $(x_i, x_j) \in C$ กล่าวคือ x_i และ x_j ไม่ควรอยู่ในกลุ่มเดียวกัน ทั้งนี้ทั้งเซต M และ C จะไม่ถือลำดับของข้อมูลที่เป็นสมาชิกในเซตทั้งสองเป็นสำคัญ กล่าวคือ $(x_i, x_j) \in M \Rightarrow (x_j, x_i) \in M$ ส่วนเซต C ก็เช่นเดียวกัน นอกจากนี้แล้วยังกำหนดให้ d_M และ d_C เป็นเมตริกซ์สองเมตริกซ์ซึ่งแทนค่า Cost เมื่อมีการฝืนกฎใน M และ C ตามลำดับ โดยกำหนดให้ $d_M(D_i, D_j) = \mathbb{1}[n_i \neq n_j]$ และ $d_C(D_i, D_j) = \mathbb{1}[n_i = n_j]$ เมื่อ $\mathbb{1}$ คืออินดิเคเตอร์ฟังก์ชันโดย $\mathbb{1}[true] = 1$ และ $\mathbb{1}[false] = 0$ และ n_i, n_j แทนกลุ่มของเอกสาร D_i และ D_j ตามลำดับ

จากกฎที่มีอยู่สามารถทำการขยายเพิ่มเติมออกไปอีกได้โดยอาศัยคุณสมบัติการถ่ายทอด (Transitive Closure) การขยายกฎที่มีอยู่นี้จะทำเฉพาะเซต M เท่านั้น โดยทำการขยายดังตารางที่ 2.1

ตารางที่ 2.1 แสดงกฎเพิ่มเติมหลังจากการขยายโดยอาศัยคุณสมบัติการถ่ายทอด

กฎที่มีอยู่	กฎที่เพิ่มเติม
$(x_i, x_j) \in M$ และ $(x_j, x_k) \in M$	$(x_i, x_k) \in M$
$(x_i, x_j) \in M$ และ $(x_j, x_k) \in C$	$(x_i, x_k) \in C$
$(x_i, x_j) \in C$ และ $(x_j, x_k) \in M$	$(x_i, x_k) \in C$

การให้ค่าน้ำหนักกับแต่ละสมาชิกในเซต M และ C นั้น แบ่งออกเป็น 2 ประเภท คือ Hard-Constraint และ Soft-Constraint โดยที่ค่าน้ำหนักที่ให้ใน Hard-Constraint นั้นจะกำหนดให้สมาชิกในเซต M มีค่าน้ำหนักเป็น 0 ส่วนในเซต C นั้นจะมีค่าน้ำหนักเป็นค่าที่มีค่ามาก ใน Soft-Constraint นั้นจะมีค่าน้ำหนักอยู่ระหว่าง -1 ถึง 1 โดยที่ค่าน้ำหนักที่มีค่าน้อยจะถูกนำมากำหนดให้กับสมาชิกในเซต M และค่าน้ำหนักที่มีค่ามากจะถูกนำมากำหนดให้กับสมาชิกในเซต C ซึ่งในกรณีนี้ค่าน้ำหนักที่เป็น 0 นั้นเราจะไม่สนใจ กล่าวคือไม่นำค่าน้ำหนักที่เป็น 0 มากำหนดให้กับสมาชิกในทั้งเซต M และ C

2.3 เซ็ลฟออร์แกไนซิงแม็ป

เซ็ลฟออร์แกไนซิงแม็ป หรือ SOM [2] และ [3] เป็นการเรียนรู้แบบหนึ่งที่รู้จักในการเรียนรู้นิวรอลเน็ตเวิร์คแบบไม่มีการชี้นำซึ่งนำเสนอโดยโคโฮเนน ลักษณะของ SOM ถูกกำหนดให้มีการแม็ปจากข้อมูลที่เป็นอินพุตไปยังชั้นของเอาท์พุต โดยใช้อัลกอริทึมสำหรับการเรียนรู้ โดยที่โมเดลของ SOM นิยามดังนี้

กำหนดให้ $x_i \in \Omega$, $(i = 1, 2, ..., d)$ เป็นเวกเตอร์คุณสมบัติของอินพุตขนาด n มิติ และ Ω เป็นอินพุตสเปซ หรือเป็นโดเมนของอินพุต ชั้นเอาท์พุตของ SOM จะประกอบไปด้วยโหนดในลักษณะของอาร์เรย์ 2 มิติ เมื่อ k คือจำนวนอินเด็กซ์ของโหนด m_j, $(j = 1, 2, ..., k)$ จะเป็นเวกเตอร์ขนาด n มิติ สำหรับทุกๆ โหนดในชั้นเอาท์พุต ลักษณะของชั้นเอาท์พุตนั้นจะมีลักษณะเป็นอาร์เรย์ขนาด $o \times p = k$ กล่าวคือมีจำนวนแถวเท่ากับ o และจำนวนคอลัมน์เท่ากับ p เมื่อในชั้นเอาท์พุตมีจำนวนโหนดทั้งหมด k โหนด

กำหนดให้ i คืออินเด็กซ์ของเวกเตอร์คุณสมบัติของอินพุต x_i สมมติให้ m_b คือโหนดในชั้นเอาท์พุตที่ใกล้เคียงที่สุดกับอินพุต x_i ของโหนดทั้งหมดในชั้นเอาท์พุต โดยการหาโหนดที่มีความใกล้เคียงกับอินพุต x_i นิยามได้ดังนี้

$$\|x_i - m_b\| = \min_j \|x_i - m_j\| \tag{2.1}$$

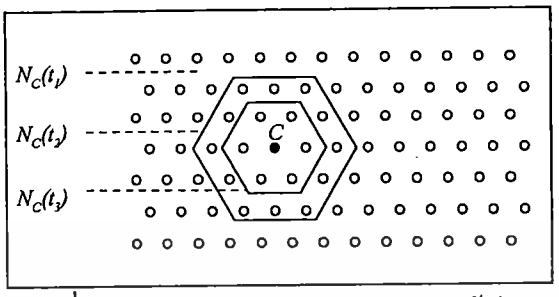
เมื่อ $\|x_i - m_b\|$ คือ Euclidean Distance ระหว่าง x_i และ m_j โดยที่โหนด m_b จะเป็นโหนดที่มีความใกล้เคียงกับเวกเตอร์คุณสมบัติของอินพุต x_i มากที่สุด

ขั้นตอนการทำงานของ SOM ประกอบด้วยขั้นตอนหลักๆ 3 ขั้นตอนที่สำคัญ คือ Competitive Process, Cooperative Process และ Adaptive Process ซึ่งแต่ละส่วนมีรายละเอียดดังนี้

1. Competitive Process เป็นขั้นตอนในการเปรียบเทียบข้อมูลอินพุตกับโหนดต่างๆ ในชั้นเอาท์พุต โดยโหนดในชั้นเอาท์พุตที่มีความใกล้เคียงกับโหนดที่เป็นอินพุตมากที่สุดจะถูกเลือกให้เป็นโหนดที่ชนะ หรือ Wining Node

2. Cooperative Process เมื่อได้รับโหนดที่เป็นโหนดที่ชนะแล้วจะเป็นขั้นตอนการเลือกโหนดใกล้เคียง (Neighborhood Node) ของโหนดนั้นๆ เพื่อทำการปรับค่าเวทเวกเตอร์ไปพร้อมๆ กัน ซึ่งจำนวนของโหนดใกล้เคียงจะเริ่มจากจำนวนที่ครอบคลุมแทบทุกโหนดในชั้นเอาท์พุต จากนั้นจำนวนของโหนดใกล้เคียงจะค่อยๆ ลดลงตามจำนวนรอบของการเทรนนิง และในรอบท้ายๆ จำนวนของโหนดใกล้เคียงจะเท่ากับ 0 ซึ่งหมายความว่าจะมีการปรับค่าเวทเวกเตอร์ในขั้นตอน Adaptive Process เฉพาะโหนดที่ชนะเท่านั้น โดยลักษณะการทำงานของการหาโหนดใกล้เคียง (Neighborhood Function) แสดงได้ดังรูปที่ 2.6

3. Adaptive Process ทำการปรับค่าเวทเวกเตอร์ให้กับโหนดที่ชนะ และโหนดใกล้เคียง เพื่อให้มีความใกล้เคียงกับข้อมูลอินพุตมากขึ้น โดยรายละเอียดการเรียนรู้ของอัลกอริทึม Self-Organizing Map แสดงดังได้รูปที่ 2.7



รูปที่ 2.6 แสดงการทำงานการหาโหนดใกล้เคียง

1. กำหนดค่าเริ่มต้นให้กับทุกๆ เวทเวกเตอร์ m_j โดยการสุ่มจากข้อมูลอินพุต
2. ทำซ้ำในขั้นตอนที่ 3, 4 และ 5 สำหรับเวลา $t = 0, 1, ..., T$
3. สำหรับแต่ละเวกเตอร์คุณสมบัติของอินพุต x_i ทำขั้นตอนที่ 4 และ 5
4. เปรียบเทียบข้อมูล x_i เพื่อหาโหนดที่ชนะ m_b โดยหาได้จาก

$$\|x_i - m_b\| = \min_j \|x_i - m_j\|$$

5. สำหรับแต่ละโหนดในชั้นเอาท์พุต ปรับค่าเวทเวกเตอร์สำหรับโหนด

$$m_j(t+1) = m_j(t) + \alpha(t)[N_{b,j}(t) \times (x_i - m_j(t))]$$

โดย ที่ $\alpha(t)$ เป็นค่าแฟกเตอร์ หรือค่า Learning-rate และ $N_{b,j}(t)$ คือการหาโหนดใกล้เคียง (Neighborhood Function) โดยที่จะมีค่าเท่ากับ 1 เมื่อโหนดนั้นๆ เป็นสมาชิกในโหนดที่ใกล้เคียง และจะมีค่าเท่ากับ 0 ในกรณีที่โหนดนั้นๆ ไม่เป็นสมาชิกในโหนดที่ใกล้เคียง โดยทั้ง $\alpha(t)$ และ $N_{b,j}(t)$ จะมีการเปลี่ยนแปลงไปในแต่ละรอบของการเรียนรู้

รูปที่ 2.7 แสดงการเรียนรู้ของอัลกอริทึมเซ็ลฟออร์แกไนซิงแม็ป

2.4 การเปรียบเทียบความแตกต่างกันของเอกสาร

ในงานทางด้านการจัดกลุ่มข้อมูลประเภทเอกสารนั้น จะมีการพิจารณาถึงคุณสมบัติต่างๆ ของเอกสาร เช่น หัวเรื่อง ชื่อผู้แต่ง หรือคำสำคัญต่างๆ ของเอกสาร ซึ่งในการแสดงคุณสมบัติต่างๆ ของเอกสารหนึ่งๆ สามารถเขียนให้อยู่ในรูปของ Cartesian Products [6] และ [7] ได้ดังนี้

$$Doc = D_1 * D_2 * D_3 * \ldots * D_d \tag{2.2}$$

เมื่อ d แทนจำนวนของคุณสมบัติของเอกสารที่นำมาพิจารณา เพื่อให้เห็นได้ชัดเจนยิ่งขึ้น จะขอยกตัวอย่างเอกสารที่มีคุณสมบัติ หัวเรื่อง (Title) และคำสำคัญ (Keyword) เราสามารถเขียนให้อยู่ในรูปของ Cartesian Products ได้ดังนี้

$$Doc = Title * Keyword \tag{2.3}$$

ในการเปรียบเทียบความแตกต่างของเอกสารสองเอกสาร A และ B ตามแนวคิดของ El-Sonbaty [7] ได้นิยามไว้ดังนี้

$$D(A,B) = \sum_{k=1}^{d} D(A_k, B_k) \tag{2.4}$$

เมื่อ d แทนจำนวนของคุณสมบัติของเอกสาร A และ B ซึ่งในการหาความแตกต่างของเอกสารสองเอกสารนั้นจะพิจารณาทุกๆ คุณสมบัติของเอกสาร โดยแต่ละคุณสมบัติลำดับที่ k ใดๆ ของเอกสารนั้นจะมีการพิจารณาออกเป็นสองส่วนคือ 1) $D_S(x_k, w_{ik})$ เป็นการเปรียบเทียบเชิง Span และ 2) $D_C(x_k, w_{ik})$ การเปรียบเทียบเชิง Content ซึ่งแต่ละส่วนนิยามดังนี้

$$D_S(A_k, B_k) = \frac{|\, Length\ of\ A_k - Length\ of\ B_k \,|}{Span\ length\ of\ A_k\ and\ B_k} \tag{2.5}$$

$$D_C(A_k, B_k) = \frac{|\, Length\ of\ A_k + Length\ of\ B_k - 2 * Length\ of\ intersection\ of\ A_k\ and\ B_k \,|}{Span\ length\ of\ A_k\ and\ B_k} \tag{2.6}$$

เมื่อ $Length\ of\ A_k$ คือจำนวนคุณสมบัติของ A_k

$Span\ length\ of\ A_k\ and\ B_k$ คือจำนวนคุณสมบัติที่เกิดจากการยูเนียนคุณสมบัติของ A_k และ B_k

$Length\ of\ intersection\ of\ A_k\ and\ B_k$ คือจำนวนคุณสมบัติที่ซ้ำของ A_k และ B_k

ค่าความแตกต่างระหว่างเอกสาร A และ B ของคุณสมบัติที่ k ใดๆ หาได้จาก

$$D(A_k, B_k) = D_S(A_k, B_k) + D_C(A_k, B_k) \quad (2.7)$$

จากการเปรียบเทียบความแตกต่างของสองเอกสาร A และ B ค่าของ $D(A, B)$ จะมีค่าอยู่ระหว่าง 0 ถึง d (เมื่อ d คือจำนวนของคุณสมบัติ) โดยที่ $D(A, B) = 0$ แสดงว่าเอกสาร A และ B ไม่มีความแตกต่างกัน หรืออาจจะกล่าวได้ว่าเอกสาร A และ B มีความเหมือนกันมาก ส่วนถ้า $D(A, B)$ มีค่ามากๆ แสดงว่าเอกสาร A และ B มีความแตกต่างกันมาก หรือเอกสาร A และ B มีความเหมือนกันน้อยนั่นเอง

ตัวอย่างการคำนวณหาค่าความแตกต่างระหว่างเอกสาร

กำหนดให้มีตัวอย่างข้อมูลเอกสารข่าวซึ่งประกอบด้วยคุณสมบัติหัวเรื่อง และคำสำคัญ รายละเอียดของตัวอย่างเอกสารข่าวแสดงดังตารางที่ 2.2

ตารางที่ 2.2 แสดงข้อมูลตัวอย่างของเอกสารข่าว

เอกสาร	คุณสมบัติ	
	หัวเรื่อง	คำสำคัญ
Doc1	acqusition, complete, process, thermo	connection, electron, process, system, thermo
Doc2	american, group, pct, unicorp	american, control, intention, investment , purposes, seek, share, unicorp
Doc3	acquisition, complete, finance, telecast	bank, business, credit, dlr, finance, sanwa

ในการหาค่าความแตกต่างของเอกสาร Doc1 และ Doc2 หาได้จากผลรวมของค่าความแตกต่างของทั้งคุณสมบัติหัวเรื่อง (Title) และคุณสมบัติคำสำคัญ (Keyword) จากสมการที่ 2.4 ค่าความแตกต่างของเอกสาร Doc1 และ Doc2 หาได้จาก

$$D(Doc1, Doc2) = D(Doc1_{title}, Doc2_{title}) + D(Doc1_{keyword}, Doc2_{keyword})$$

จากที่กล่าวมาแล้วว่าค่าความแตกต่างของแต่ละคุณสมบัติของเอกสารนั้นหาได้จากผลรวมของการเปรียบเทียบความแตกต่างเชิง span และเชิง content จากสมการที่ 2.7 ค่าความแตกต่างของคุณสมบัติหัวเรื่องของเอกสาร Doc1 และ Doc2 หาได้จาก

$$D(Doc1_{title}, Doc2_{title}) = D_S(Doc1_{title}, Doc2_{title}) + D_C(Doc1_{title}, Doc2_{title})$$

จากสมการที่ 2.5 และ 2.6 หาค่าความแตกต่างเชิง span และเชิง content ได้ดังนี้

$$D_S(Doc1_{title}, Doc2_{title}) = \frac{|4-4|}{8} = 0$$

$$D_C(Doc1_{title}, Doc2_{title}) = \frac{|4+4-(2*0)|}{8} = 1$$

ค่าความแตกต่างของคุณสมบัติหัวเรื่องของเอกสาร Doc1 และ Doc2 มีค่าเท่ากับ

$$D(Doc1_{title}, Doc2_{title}) = 0+1 = 1$$

ต่อไปพิจารณาที่คุณสมบัติคำสำคัญ จากสมการที่ 2.7 ค่าความแตกต่างของคุณสมบัติหัวเรื่องของเอกสาร Doc1 และ Doc2 หาได้จาก

$$D(Doc1_{keyword}, Doc2_{keyword}) = D_S(Doc1_{keyword}, Doc2_{keyword}) + D_C(Doc1_{keyword}, Doc2_{keyword})$$

จากสมการที่ 2.5 และ 2.6 หาค่าความแตกต่างเชิง span และเชิง content ได้ดังนี้

$$D_S(Doc1_{keywrod}, Doc2_{keyword}) = \frac{|5-8|}{13} = \frac{3}{13}$$

$$D_C(Doc1_{keyword}, Doc2_{keyword}) = \frac{|5+8-(2*0)|}{13} = 1$$

ค่าความแตกต่างของคุณสมบัติคำสำคัญของเอกสาร Doc1 และ Doc2 มีค่าเท่ากับ

$$D(Doc1_{keyword}, Doc2_{keyword}) = \frac{3}{13} + 1 = 1.2308$$

ดังนั้นค่าความแตกต่างของเอกสารข่าว Doc1 และ Doc2 มีค่าเท่ากับ

$$D(Doc1, Doc2) = 1 + 1.2308 = 2.2308$$

ในขั้นตอนต่อมาพิจารณาหาค่าความแตกต่างของเอกสาร Doc1 และ Doc3 ซึ่งค่าความแตกต่างนี้หาได้จาก

$$D(Doc1, Doc3) = D(Doc1_{title}, Doc3_{title}) + D(Doc1_{keyword}, Doc3_{keyword})$$

พิจารณาที่คุณสมบัติหัวเรื่อง จากสมการที่ 2.7 ค่าความแตกต่างของคุณสมบัติหัวเรื่องของเอกสาร Doc1 และ Doc3 หาได้จาก

$$D(Doc1_{title}, Doc3_{title}) = D_S(Doc1_{title}, Doc3_{title}) + D_C(Doc1_{title}, Doc3_{title})$$

จากสมการที่ 2.5 และ 2.6 หาค่าความแตกต่างเชิง span และเชิง content ได้ดังนี้

$$D_S(Doc1_{title}, Doc3_{title}) = \frac{|4-4|}{6} = 0$$

$$D_C(Doc1_{title}, Doc3_{title}) = \frac{|4+4-(2*2)|}{6} = \frac{4}{6}$$

ค่าความแตกต่างของคุณสมบัติหัวเรื่องของเอกสาร Doc1 และ Doc3 มีค่าเท่ากับ

$$D(Doc1_{title}, Doc3_{title}) = 0 + \frac{4}{6} = 0.6667$$

พิจารณาที่คุณสมบัติคำสำคัญ จากสมการที่ 2.7 ค่าความแตกต่างของคุณสมบัติคำสำคัญของเอกสาร Doc1 และ Doc3 หาได้จาก

$$D(Doc1_{keyword}, Doc3_{keyword}) = D_S(Doc1_{keyword}, Doc3_{keyword}) + D_C(Doc1_{keyword}, Doc3_{keyword})$$

จากสมการที่ 2.5 และ 2.6 หาค่าความแตกต่างเชิง span และเชิง content ได้ดังนี้

$$D_S(Doc1_{keyword}, Doc3_{keyword}) = \frac{|5-6|}{11} = \frac{1}{11}$$

$$D_C(Doc1_{keyword}, Doc3_{keyword}) = \frac{|5+6-(2*0)|}{11} = 1$$

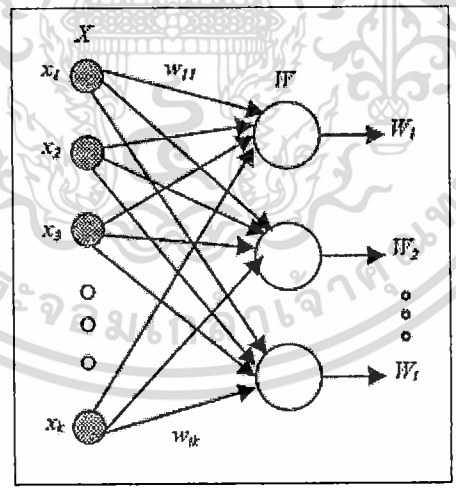
ค่าความแตกต่างของคุณสมบัติหัวเรื่องของเอกสาร Doc1 และ Doc3 มีค่าเท่ากับ

$$D(Doc1_{keyword}, Doc3_{keyword}) = \frac{1}{11} + 1 = 1.0909$$

จากตัวอย่างการจะเห็นว่าค่าความแตกต่างระหว่างเอกสาร Doc1 และ Doc2 มีค่ามากกว่าค่าความแตกต่างระหว่างเอกสาร Doc1 และ Doc3 โดยหลักการและแนวคิดของการเปรียบเทียบความแตกต่างระหว่างสองเอกสารที่ได้กล่าวมาข้างต้นนี้จะนำมาประยุกต์ใช้กับอัลกอริทึมสำหรับการจัดกลุ่มข้อมูลประเภทข้อความต่อไป

2.5 เท็กโปรเซสซิงโคโฮเนนนิวรอลเน็ตเวิร์ค

ลักษณะโครงสร้างของเท็กโปรเซสซิงโคโฮเนนนิวรอลเน็ตเวิร์ค [7] เป็นนิวรอลเน็ตเวิร์คที่ประกอบด้วยชั้นของอินพุต X และเอาท์พุต W โครงสร้างของนิวรอลเน็ตเวิร์คนี้แสดงดังรูปที่ 2.8 โดย x_k คือข้อมูลอินพุต และ W_i คือเอาท์พุต ซึ่งแต่ละโหนดของอินพุทจะถูกกำหนดตามจำนวนคุณสมบัติของข้อมูล



รูปที่ 2.8 แสดงโครงสร้างของเท็กโปรเซสซิงโคโฮเนนนิวรอลเน็ตเวิร์ค

ความแตกต่างเท็กโปรเซสซิงโคโฮเนนนิวรอลเน็ตเวิร์คกับนิวรอลเน็ตเวิร์คทั่วๆ ไป คือ ลักษณะของข้อมูลของโหนดอินพุตของจะมีลักษณะเป็นข้อมูลเชิงคุณภาพ (Qualitative Value) และ

ค่าเวทเวกเตอร์ w_{ik} สำหรับแต่ละข้อมูลอินพุต k กับข้อมูลเอาท์พุต i ในส่วนของชั้นอินพุต X นิยามและเอาท์พุต W นิยามดังนี้

$$X = (x_1, x_2, x_3, \ldots, x_k) \tag{2.8}$$

$$W = \{W_1, W_2, W_3, \ldots, W_i\} \tag{2.9}$$

โดยที่ k เป็นจำนวนของคุณสมบัติของอินพุต X และ i เป็นจำนวนของโหนดในชั้นเอาท์พุต W ในส่วนของชั้นเอาท์พุต W แต่ละโหนดเอาท์พุต W_i นิยามดังนี้

$$W_i = \{w_{i1}, w_{i2}, w_{i3}, \ldots, w_{ik}\} \tag{2.10}$$

เมื่อ w_{ik} เป็นค่าเวทเวกเตอร์ของนิวรอลโหนด W_i คุณสมบัติที่ k

ข้อมูลอินพุต X ในอินพุตยูนิตนั้นจะเชื่อมต่อกันอย่างสมบูรณ์กับนิวรอลโหนดทุกโหนดในเอาท์พุตยูนิต W จากรูปที่ 2.8 เส้นที่เชื่อมระหว่างอินพุตยูนิต X กับเอาพุตยูนิต W คือค่าเวทเวกเตอร์ ซึ่งนิยามดังนี้

$$w_{ik} = \{(A_{1ik}, e_{1ik}), (A_{2ik}, e_{21ik}), \ldots, (A_{pik}, e_{pik})\} \tag{2.11}$$

เมื่อ A_{pik} คือค่าข้อมูลเชิงคุณภาพลำดับที่ p ของเวทเวกเตอร์ w_{ik} และ e_{pik} คือค่าแสดงความเป็นสมาชิก หรือดีกรีของ A_{pik} โดยค่าของ e_{pik} จะมีค่าอยู่ระหว่าง 0 ถึง 1 โดยที่ e_{pik} จะมีค่าเท่ากับ 0 ถ้าค่าข้อมูลเชิงคุณภาพ A_{pik} ไม่ได้เป็นส่วนหนึ่งของข้อมูลเข้าในอินพุตยูนิตลำดับที่ i และ e_{pik} จะมีค่าเท่ากับ 1ถ้าค่าข้อมูลเชิงคุณภาพ A_{pik} เป็นข้อมูลเข้าอย่างสมบูรณ์ในอินพุตยูนิตลำดับที่ i

บทที่ 3
การจัดกลุ่มเอกสารโดยใช้โคโฮเนนนิวรอลเน็ตเวิร์ค ร่วมกับกฎระหว่างสองข้อมูล

การจัดกลุ่มเอกสารโดยใช้โคโฮเนนนิวรอลเน็ตเวิร์คร่วมกับกฎระหว่างสองข้อมูล เป็นนิวรอลเน็ตเวิร์คที่นำหลักการของโคโฮเนนเซ็ลฟออร์แกไนซิงแม็ป และแนวคิดเกี่ยวกับการเปรียบเทียบแตกต่างระหว่างของข้อมูลเชิงสัญลักษณ์ มาประยุกต์เพื่อให้นิวรอลเน็ตเวิร์คนี้สามารถรับข้อมูลที่เป็นข้อความได้โดยตรงไม่ต้องทำการแปลงให้เป็นข้อมูลเชิงปริมาณแต่อย่างใดซึ่งทำให้ความหมายของข้อมูลยังคงอยู่ นอกจากนี้แล้วในขั้นตอนการทำงานนิวรอลเน็ตเวิร์คนี้ยังได้นำแนวคิดของกฎระหว่างสองเข้ามาทำงานร่วมด้วย

3.1 การจัดกลุ่มเอกสารโดยใช้โคโฮเนนนิวรอลเน็ตเวิร์คร่วมกับกฎระหว่างสองข้อมูล

ลักษณะโครงสร้างนิวรอลเน็ตเวิร์คของการจัดกลุ่มเอกสารโดยใช้โคโฮเนนนิวรอลเน็ตเวิร์คร่วมกับกฎระหว่างสองข้อมูลนั้นเหมือนกันกับโครงสร้างของของเท็คโปรเซสซิงนิวรอลเน็ตเวิร์ค [7] แสดงดังรูปที่ 2.3 ซึ่งประกอบไปด้วยสองส่วนหลัก คือ ส่วนของอินพุตยูนิต X และส่วนของเอาท์พุตยูนิต W ซึ่งแต่ละส่วนของอินพุตยูนิตนิยามดังนี้

$$X = (x_1, x_2, x_3, \ldots, x_k) \tag{3.1}$$

โดยที่ k เป็นจำนวนของคุณสมบัติของอินพุตยูนิต X และส่วนของเอาท์พุตยูนิตนิยามได้ดังนี้

$$W_i = \{w_{i1}, w_{i2}, w_{i3}, \ldots, w_{ik}\} \tag{3.2}$$

เมื่อ w_{ik} เป็นค่าเวทเวกเตอร์ของนิวรอลโหนด W_i คุณสมบัติที่ k

ข้อมูลเข้า X ในอินพุตยูนิตนั้นจะเชื่อมต่ออย่างสมบูรณ์กับนิวรอลโหนดทุกโหนดในเอาท์พุตยูนิต W จากรูปที่ 3.1 เส้นที่เชื่อมระหว่างอินพุตยูนิต X กับเอาพุตยูนิต W คือค่าเวทเวกเตอร์ซึ่งนิยามดังนี้

$$w_{ik} = \{(A_{1ik}, e_{1ik}), (A_{2ik}, e_{21ik}), \ldots, (A_{pik}, e_{pik})\} \tag{3.3}$$

เมื่อ A_{pik} คือค่าข้อมูลเชิงคุณภาพ (Quantitative Value) ลำดับที่ p ของเวทเวกเตอร์ w_{ik} และ e_{pik} คือค่าแสดงความเป็นสมาชิกของ A_{pik} โดยค่าของ e_{pik} จะมีค่าอยู่ระหว่าง 0 ถึง 1 โดยที่ e_{pik} จะมีค่าเท่ากับ 0 ถ้าค่าข้อมูลเชิงคุณภาพ A_{pik} ไม่ได้เป็นส่วนหนึ่งของข้อมูลเข้าในอินพุตยูนิตลำดับที่ i และ e_{pik} จะมีค่าเท่ากับ 1 ถ้าค่าข้อมูลเชิงคุณภาพ A_{pik} เป็นข้อมูลเข้าอย่างสมบูรณ์ในอินพุตยูนิตลำดับที่ i

กฎที่นำมาใช้ในอัลกอริทึมนี้คือกฎระหว่างสองข้อมูล ซึ่งเป็นหนึ่งในกฎประเภท Instance-level Constraints โดยที่ค่าของกฎระหว่างสองข้อมูลประกอบด้วย ส่วนของ Must-link Constraints และ Cannot-link Constraints ซึ่งกำหนดให้ M และ C แทนเซตของ Must-link Constraints และ Cannot-link Constraints ตามลำดับ นอกจากนี้แล้วยังมีการกำหนดค่าน้ำหนักให้กับแต่สมาชิกทั้งในเซตของ Must-link Constraints (${}_mW$) และ Cannot-link Constraints (${}_cW$) โดยที่แต่ละส่วนของกฎนิยามดังนี้

$$M = \{(D_i, D_j)\} \quad , {}_mW = \{{}_m w_{ij}\} \tag{3.4}$$

$$C = \{(D_i, D_j)\} \quad , {}_cW = \{{}_c w_{ij}\} \tag{3.5}$$

ค่าน้ำหนักจะมีค่าเมื่อมีการละเมิดกฎเกิดขึ้นใน Must-link Constraints หรือ Cannot-link Constraints หรืออาจจะมีการละเมิดกฎทั้งสองกฎก็ได้ โดยนิยามดังนี้

$$V_{(i,j)}(n_i, n_j) = \begin{cases} {}_m w_{ij}\, \iota[n_i \neq n_j] & if\ (D_i, D_j) \in M \\ {}_c w_{ij}\, \iota[n_i = n_j] & if\ (D_i, D_j) \in C \\ 0 & otherwise \end{cases} \tag{3.6}$$

เมื่อ ι คืออินดิเคเตอร์ฟังก์ชันโดย ι[true] = 1 และ ι[false] = 0

n_i, n_j แทนกลุ่มของเอกสาร i และ j ใดๆ

3.2 ขั้นตอนการทำงานของอัลกอริทึม

กระบวนการเรียนรู้ของการจัดกลุ่มเอกสารโดยใช้โคโฮเนนนิวรอลเน็ตเวิร์คร่วมกับกฎระหว่างสองข้อมูล เป็นการขยายความสามารถของ Kohonen Self-Organizing Map ในส่วนของการเรียนรู้แบบแข่งขัน (Competitive Learning) โดยการเพิ่มแนวคิดของการเปรียบเทียบคุณสมบัติของข้อมูลที่เป็นข้อมูลเชิงคุณภาพ ซึ่งทำให้อัลกอริทึมนี้สามารถดำเนินการกับข้อมูลเชิงคุณภาพได้

โดยตรง อีกทั้งยังได้รวมแนวคิดของกฎระหว่างสองข้อมูลเข้าในอัลกอริทึมนี้ด้วย ซึ่งการเลือกกฎที่มีความเหมาะสมนี้จะทำให้ผลของการทำงานของอัลกอริทึมมีความถูกต้องมากขึ้นในกรณีของที่ข้อมูลแยกออกจากกันไม่ชัดเจน ขั้นตอนการทำงานของอัลกอริทึมแสดงได้ดังรูปที่ 3.1

Step 0: Initialize weight w_{ik} in each neural output W_i. Each weight can be initialized from the training data arbitrarily.

Step 1: While stopping condition is false, do step 2-6

Step 2: For each input vector

$X = (x_1, x_2, \ldots, x_d)^T$, do step 3-6

Step 3: For each input unit 'i' compute

$$\|X - W_i\| = \sum_{k=1}^{d} D_i(x_i, w_{ik}) + \sum_{(D_i, D_j) \in M}^{m} w_{ij} l[n_i \neq n_j] + \sum_{(D_i, D_j) \in C}^{c} w_{ij} l[n_i = n_j]$$

Step 4: Find index 'I' such that $\|X\text{-}W_i\|$ is minimum and assign $n_i = I$.

Step 5: For all weight that connect to the winning node 'I' and its neighborhood ($\wedge_I$).

$$w_{ik}^{(new)} = \begin{cases} w_{ik}^{(old)} \cup x_k & if \quad i \in \wedge_I \\ w_{ik}^{(old)} & otherwise \end{cases}$$

And

$$e_{pik}^{(new)} = \begin{cases} f(e_{pik}^{(old)} - \eta) & if\ A_{pik} \notin w_{ik} \cap x_k \\ f(e_{pik}^{(old)} + \eta) & if\ A_{pik} \in w_{ik} \cap x_k \\ \eta_0 & if\ A_{pik} \in x_j - (w_{ik} \cap x_k) \end{cases}$$

Where $f(.)$ is defined as

$$f(x) = \begin{cases} x & if\ 0 \le x \le 1 \\ 0 & if\ x < 0 \\ 1 & if\ x > 1 \end{cases}$$

Step 6: Continue with step 1-5 until the stopping condition is true

รูปที่ 3.1 แสดงขั้นตอนการทำงานของการจัดกลุ่มเอกสารโดยใช้โคโฮเนนนิวรอลเน็ตเวิร์คร่วมกับกฎระหว่างสองข้อมูล

การทำงานของอัลกอริทึมการจัดกลุ่มเอกสารโดยใช้โคโฮเนนนิวรอลเน็ตเวิร์คร่วมกับกฎระหว่างสองข้อมูลประกอบด้วยส่วนหลักที่สำคัญ 3 ส่วนคือ ส่วนของขั้นตอนการเรียนรู้แบบ

แข่งขัน (Competitive Process) ส่วนของการหานิวรอลโหนดใกล้เคียงของนิวรอลโหนดที่ถูกเลือก (Cooperative Process) และส่วนสุดท้ายจะเป็นส่วนของการปรับค่าเวทเวกเตอร์ของนิวรอลโหนดที่ถูกเลือกและนิวรอลโหนดใกล้เคียงให้มีค่าใกล้เคียงกับอินพุตยูนิต X มากขึ้น (Adaptive Process) ซึ่งในแต่ละส่วนหลักที่สำคัญมีรายละเอียดดังนี้

3.2.1 ส่วนของขั้นตอนการเรียนรู้แบบแข่งขัน (Competitive Process)

เป็นส่วนที่ประยุกต์แนวคิดเรื่องการหาค่าความแตกต่างของเอกสาร และแนวคิดกฎระหว่างสองข้อมูลเข้ากับการทำงานของการเรียนรู้แบบแข่งขัน (Competitive Learning) โดยมีขั้นตอนการทำงานหลักคือ การหานิวรอลโหนด W_i ที่มีความเหมือนกันกับข้อมูลเข้าอินพุตยูนิต X มากที่สุด ในการหาค่าความเหมือนกันของนิวรอลโหนด W_i และข้อมูลอินพุตยูนิต X หาได้จากค่าความแตกต่างรวมของแต่ละคุณสมบัติของทั้งสองรวมกับผลรวมของกฎ หรือ Constraints ทั้งในส่วนของ Must-link Constraints และ Cannot-link Constraints โดยเมื่อค่าความแตกต่างของแต่ละคุณสมบัติหาได้จากผลรวมระหว่างค่าความแตกต่างในเชิง Span และค่าความแตกต่างในเชิง Content นิวรอลโหนดที่ถูกเลือกนั้นจะเป็นนิวรอลโหนดที่มีค่าความแตกต่างกับข้อมูลอินพุตยูนิต X น้อยที่สุด ซึ่งการหาค่าความเหมือนกันของนิวรอลโหนด W_i และข้อมูลอินพุตยูนิต X หาได้จาก

$$\|X - W_i\| = \sum_{k=1}^{d} D_i(x_i, w_{ik}) + \sum_{(D_i,D_j)\in M}^{m} w_{ij}\iota[n_i \neq n_j] + \sum_{(D_i,D_j)\in C}^{c} w_{ij}\iota[n_i = n_j] \tag{3.7}$$

ในการหาค่าความแตกต่างของคุณสมบัติที่ k ของนิวรอลเวกเตอร์ w_{ik} ($D(x_k, w_{ik})$) หาได้จากผลรวมของค่าความแตกต่างเชิง Span ($D_S(x_k, w_{ik})$) และค่าความแตกต่างเชิง Content ($D_C(x_k, w_{ik})$) โดยนิยามดังนี้

$$D(x_k, w_{ik}) = D_S(x_k, w_{ik}) + D_C(x_k, w_{ik}) \tag{3.8}$$

$$D_S(x_k, w_{ik}) = \frac{|\,Length\ of\ x_k - Length\ of\ w_{ik}\,|}{Span\ length\ of\ x_k\ and\ w_{ik}} \tag{3.9}$$

$$D_C(x_k, w_{ik}) = \frac{|\,Length\ of\ x_k + Length\ of\ w_{ik} - 2 * Length\ of\ intersection\ of\ x_k\ and\ w_{ik}\,|}{Span\ length\ of\ x_k\ and\ w_{ik}} \tag{3.10}$$

เมื่อ $Length\ of\ x_k$ คือจำนวนคุณสมบัติของ w_{ik}

$Span\ length\ of\ x_k\ and\ w_{ik}$ คือจำนวนคุณสมบัติที่เกิดจากการยูเนียนคุณสมบัติของ x_k และ w_{ik}

$Length\ of\ intersection\ of\ x_k\ and\ w_{ik}$ คือจำนวนคุณสมบัติที่ซ้ำกันของ x_k และ w_{ik}

ค่าของกฎประกอบด้วยสองส่วนคือส่วนของ Must-link Constraints และ Cannot-link Constraint ซึ่งแทนด้วย $\sum_{(D_i,D_j)\in M} {}_m w_{ij} \iota[n_i \neq n_j]$ และ $\sum_{(D_i,D_j)\in C} {}_c w_{ij} \iota[n_i = n_j]$ ตามลำดับ ซึ่งในการกำหนดค่าน้ำหนักให้กับแต่สมาชิกทั้งในเซตของ Must-link Constraints (${}_m w_{ij}$) และ Cannot-link Constraints (${}_c w_{ij}$) ในทางปฏิบัตินั้นทำได้ไม่ง่ายนัก ดังนั้นในการทำงานของอัลกอริทึมนี้จะกำหนดให้ค่าน้ำหนักของแต่ละสมาชิกทั้งในเซต Must-link Constrains และ Cannot-link Constraints มีค่าเท่ากันคือ 1 สำหรับทุกๆ สมาชิกของทั้งสองเซต ซึ่งจะมีค่าน้ำหนักเมื่อมีการละเมิดกฎเกิดขึ้นใน Must-link Constraints หรือ Cannot-link Constraints หรืออาจจะมีการละเมิดกฎทั้งสองก็ได้

จากการหาค่าความเหมือนกันของนิวรอลโหนด W_i และข้อมูลอินพุตยูนิต X นิวรอลโหนดที่จะถูกเลือกนั้นจะเป็นนิวรอลโหนดที่มีค่าความแตกต่างกับข้อมูลอินพุตยูนิต X น้อยที่สุด

3.2.2 ส่วนของการหานิวรอลโหนดใกล้เคียง (Cooperative Process)

เมื่อทราบนิวรอลโหนดที่มีค่าความแตกต่างกับข้อมูลอินพุตยูนิต X หรือนิวรอลโหนดที่ถูกเลือก ในส่วนนี้จะเป็นการหานิวรอลโหนดที่อยู่ใกล้เคียงกับนิวรอลโหนดที่ถูกเลือก ซึ่งจำนวนของนิวรอลโหนดที่อยู่ใกล้เคียงจะค่อยลดลงตามจำนวนรอบของการเทรนนิงขึ้นอยู่กับ Neighborhood Function ($\wedge$) จะเป็นตัวกำหนด ซึ่งนิวรอลโหนดที่ถูกเลือกและนิวรอลโหนดที่อยู่ใกล้เคียงจะนำมาพิจารณาทำการปรับค่าเวทเวกเตอร์ต่อไป

3.2.3 ส่วนของการปรับค่าเวทเวกเตอร์ (Adaptive Process)

ในส่วนนี้จะเป็นส่วนของการปรับค่าเวทเวกเตอร์ของนิวรอลโหนดและนิวรอลโหนดใกล้เคียงให้มีความใกล้เคียงกับอินพุตยูนิต X มากขึ้น โดยที่การปรับสามารถแบ่งออกเป็นส่วนย่อย 2 ส่วนคือ ส่วนการปรับค่าเวทเวกเตอร์ w_{ik} ในส่วนของ A_{pik} และ e_{pik} รายละเอียดมีดังนี้

1. ส่วนการปรับค่าเวทเวกเตอร์ w_{ik} ในส่วนของ A_{pik} เพื่อเพิ่มสมาชิกใหม่เข้าไปใน w_{ik} โดย

$$w_{ik}^{(new)} = \begin{cases} w_{ik}^{(old)} \cup x_k & if \quad i \in \wedge_I \\ w_{ik}^{(old)} & otherwise \end{cases} \tag{3.11}$$

กล่าวคือจะทำการเพิ่มสมาชิกใหม่เข้าไปใน w_{ik} เมื่อเป็นค่าเวทเวกเตอร์ในนิวรอลโหนดที่เป็น Neighborhood Nodes

2. ส่วนการปรับค่าเวทเวกเตอร์ w_{ik} ในส่วนของ e_{pik} โดยแบ่งออกเป็น 3 ส่วนย่อยๆ คือ

$$e_{pik}^{(new)} = \begin{cases} f(e_{pik}^{(old)} - \eta) & if\ A_{pik} \notin w_{ik} \cap x_k \\ f(e_{pik}^{(old)} + \eta) & if\ A_{pik} \in w_{ik} \cap x_k \\ \eta_0 & if\ A_{pik} \in x_j - (w_{ik} \cap x_k) \end{cases} \quad (3.12)$$

โดยส่วนแรกเป็นส่วนของ e_{pik} ที่มี A_{pik} ที่มีค่าซ้ำกับสมาชิกใน x_k ค่าของ e_{iok} ในส่วนนี้จะถูกปรับให้มีค่าเพิ่มขึ้นเท่ากับค่า Learning Rate (η) ในส่วนที่สองเป็นส่วนของ e_{pik} ที่มี A_{pik} มีค่าไม่ซ้ำกับสมาชิกใน x_k โดยค่าของ e_{pik} จะถูกปรับให้มีค่าลดลงเท่ากับค่า Learning Rate (η) และในส่วนสุดท้ายเป็นส่วนที่ A_{pik} เป็นค่าที่เพิ่งรับเข้ามาใหม่ ค่าของ e_{pik} จะถูกกำหนดขึ้นใหม่ η_0

3.3 ตัวอย่างการทำงานของอัลกอริทึม

เพื่อให้เข้าใจการทำงานของอัลกอริทึมขอยกตัวอย่างการทำงานของการหาโหนดที่ชนะ โดยกำหนดให้จำนวนเอกสารทั้งหมดเท่ากับ 6 เอกสาร ซึ่งแต่ละคุณสมบัติของแต่ละเอกสารแสดงดังตารางที่ 3.1

ตารางที่ 3.1 แสดงรายละเอียดคุณสมบัติของเอกสารตัวอย่าง

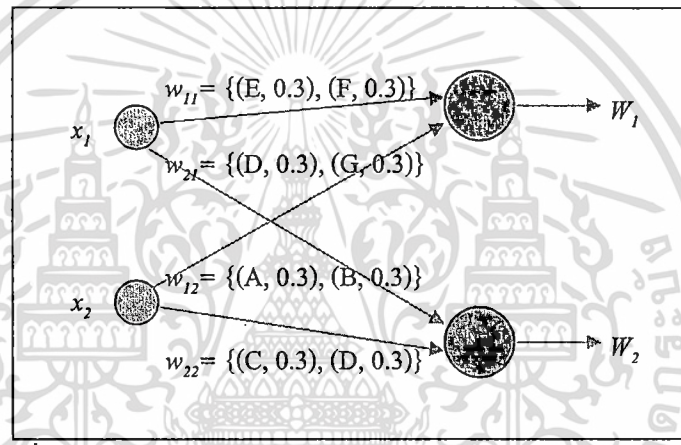
เอกสาร	คุณสมบัติ	
	หัวเรื่อง	คำสำคัญ
D_1	A,E	C,D
D_2	E,F	D,G
D_3	A,B	C,D
D_4	F	G
D_5	E	D
D_6	E	D

กำหนดให้มีกฎที่เป็น Must-link Constraints และ Cannot-link Constraints จำนวนละ 2 กฎ นอกจากนี้กำหนดให้ค่าน้ำหนักของแต่ละสมาชิกทั้งในเซต Must-link Constraints ($_mw_{ij}$) และ Cannot-link Constraints ($_cw_{ij}$) มีค่าเท่ากันคือ 1 สำหรับทุกๆ สมาชิกของทั้งสองเซต ซึ่งกฎที่เป็น Must-link Constraints และ Cannot-link Constraints แทนด้วยเซต M และ C ตามลำดับ โดยแต่ละเซตประกอบด้วยสมาชิกของกฎดังนี้

$$M = \{(D_1, D_3), (D_5, D_6)\} \quad , {}_mW = \{({}_m w_{13}, 1), ({}_m w_{56}, 1)\} \tag{3.13}$$

$$C = \{((D_1, D_4), (D_3, D_4)\} \quad , {}_cW = \{({}_c w_{14}, 1), ({}_c w_{34}, 1)\} \tag{3.14}$$

กำหนดให้โครงข่ายของตัวอย่างการทำงานนี้ประกอบด้วยนิวรอลโหนดในชั้นอินพุต จำนวน 2 โหนด และในชั้นเอาท์พุตโหนดจำนวน 2 โหนด โดยที่นิวรอลโหนดในชั้นอินพุตแทนคุณสมบัติหัวเรื่อง และคำสำคัญของข้อมูลเอกสาร ส่วนนิวรอลโหนดในชั้นเอาท์พุตแทนกลุ่มของข้อมูลซึ่งในที่นี้ตัวอย่างข้อมูลเอกสารมีจำนวน 2 กลุ่ม นอกจากนี้กำหนดค่าเริ่มต้นให้กับแต่ละโหนดในชั้นเอาท์พุตจากการสุ่มตัวอย่างของข้อมูลอินพุต ลักษณะโครงสร้างของโครงข่ายที่ใช้อธิบายการทำงานของตัวอย่างข้อมูลเอกสารนี้แสดงดังรูปที่ 3.2



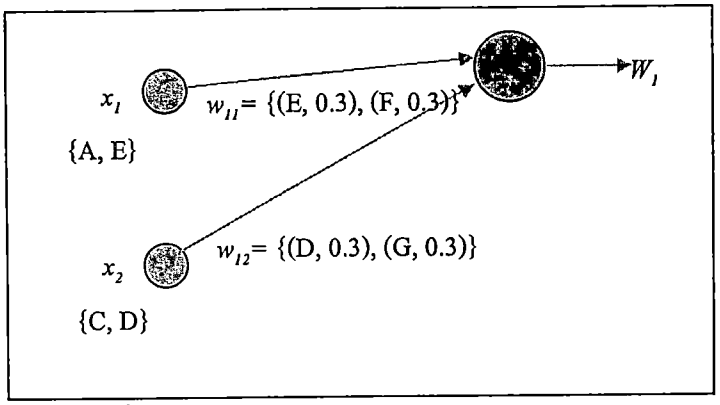
รูปที่ 3.2 แสดงโครงข่ายที่ใช้อธิบายตัวอย่างการทำงานของอัลกอริทึม

การทำงานของอัลกอริทึมจะเริ่มต้นกำหนดค่ากลุ่มเริ่มต้นให้กับทุกๆ เอกสารมีค่าเป็น 1 กล่าวคือ เริ่มต้นกำหนดให้ทุกๆ เอกสารมีโหนดที่ชนะเป็นโหนดลำดับที่ 1 ($n_i = 1$) กำหนดให้ W_i เป็นเอาท์พุตยูนิตที่ประกอบด้วยโหนดจำนวนสองโหนดกล่าวคือ $W_i = \{w_{i1}, w_{i2}\}$

ต่อไปขอยกตัวอย่างการหาโหนดที่ชนะของเอกสาร Doc1 โดยจะแทนด้วยเอกสาร D_1 ด้วย $X = \{x_1, x_2\}$ เมื่อ $x_1 = \{A, E\}$ และ $x_2 = \{C, D\}$ จากสมการที่ 3.7 การหาค่าความเหมือนกันของนิวรอลโหนด W_i และข้อมูลอินพุตยูนิต X โดยคำนวณหาค่าความเหมือนกันของทั้ง 2 นิวรอลโหนดในเอาท์พุตยูนิต ซึ่งการคำนวณค่าต่างๆ ของแต่ละนิวรอลโหนดแสดงได้ดังนี้

1. พิจารณานิวรอลโหนดที่ 1 (W_1)

กำหนดให้ n_1=1 (เนื่องจากเรากำลังพิจารณาที่เอกสาร D_1 กับนิวรอลโหนดที่ 1) และเราทราบว่า n_3=1 และ n_4=1 โครงข่ายที่พิจารณาแสดงดังรูปที่ 3.3 เมื่อข้อมูลอินพุตที่พิจารณาคือ $X = \{x_1, x_2\}$ เมื่อ $x_1 = \{A, E\}$ และ $x_2 = \{C, D\}$



รูปที่ 3.3 แสดงโครงข่ายที่ใช้อธิบายตัวอย่างการทำงานของอัลกอริทึมพิจารณานิวรอลโหนดที่ 1

ค่าความเหมือนกันของนิวรอลโหนดที่ 1 (W_1) และข้อมูลอินพุตยูนิต X คำนวณได้จาก

$$\|X - W_1\| = \sum_{k=1}^{2} D_1(x_i, w_{1k}) + \sum_{(D_i,D_j)\in M}{}_m w_{ij}\, l[n_i \neq n_j] + \sum_{(D_i,D_j)\in C}{}_c w_{ij}\, l[n_i = n_j] \qquad (3.15)$$

จากสมการที่ 3.15 จะแยกการอธิบายออกเป็นสามส่วนคือ ส่วนแรกเป็นการคำนวณหาค่าความแตกต่างระหว่างเอกสาร D_1 และนิวรอลโหนดที่ 1 (W_1) ส่วนที่สองเป็นการคำนวณหาค่าการละเมิดกฎที่เป็น Must-link Constraints และส่วนสุดท้ายเป็นการคำนวณหาค่าการละเมิดกฎที่เป็น Cannot-link Constraints ซึ่งแต่ละส่วนมีรายละเอียดดังนี้

1. คำนวณหาค่าความแตกต่างระหว่างระหว่างเอกสาร D_1 กับนิวรอลโหนดที่ 1 (W_1)

$$\begin{aligned}\sum_{k=1}^{2} D(x_k, w_{1k}) &= D(x_1, w_{11}) + D(x_2, w_{12}) \\ &= D_c(x_1, w_{11}) + D_s(x_1, w_{11}) + D_c(x_2, w_{12}) + D_s(x_2, w_{12}) \\ &= \frac{|2-2|}{3} + \frac{|2+2-(2*1)|}{3} + \frac{|2-2|}{3} + \frac{|2+2-(2*1)|}{3} \\ &= 0 + \frac{2}{3} + 0 + \frac{2}{3} = 1.333\end{aligned}$$

2. คำนวณหาค่าการละเมิดกฎที่เป็น Must-link Constraints จากสมการที่ 3.13 แสดงรายละเอียดของกฎที่เป็น Must-link Constraints เนื่องจากกำลังพิจารณาที่เอกสาร D_1 และจากที่เราทราบว่า n_1=1 และ n_3=1 จะได้ค่าที่เกิดจากการละเมิดกฎของเอกสาร D_1 ดังนี้

$$M = \{\boxed{(D_1, D_3)}, (D_5, D_6)\} \qquad {}_mW = \{\boxed{({}_m w_{13}, 1)}, ({}_m w_{56}, 1)\}$$

$$\sum_{(D_x,D_y)\in M} {}_m w_{xy}\, \iota[n_x \neq n_y] = {}_m w_{13}\, \iota[n_1 \neq n_3] \quad \text{(False)}$$

$$= 1*0$$

$$= 0$$

3. คำนวณหาค่าการละเมิดกฎที่เป็น Cannot-link Constraints จากสมการที่ 3.14 แสดงรายละเอียดของกฎที่เป็น Cannot-link Constraints เนื่องจากกำลังพิจารณาที่เอกสาร D_1 และจากที่เราทราบว่า n_1=1 และ n_4=1 จะได้ค่าที่เกิดจากการละเมิดกฎของเอกสาร D_1 ดังนี้

$$C = \{\boxed{(D_1, D_4)}, (D_3, D_4)\} \qquad {}_cW = \{\boxed{({}_c w_{14}, 1)}, ({}_c w_{34}, 1)\}$$

$$\sum_{(D_x,D_y)\in C} {}_c w_{xy}\, \iota[n_x = n_y] = {}_c w_{14}\, \iota[n_1 = n_4] \quad \text{(True)}$$

$$= 1*1$$

$$= 1$$

จากสมการที่ 3.15 ค่าความเหมือนกันของนิวรอลโหนดที่ 1 (W_1) และเอกสาร D_1 มีค่าเท่ากับ

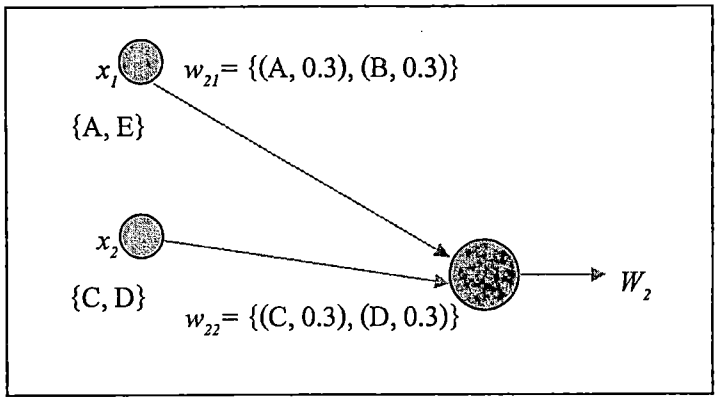
$$\|X - W_1\| = \sum_{k=1}^{2} D_1(x_i, w_{1k}) + \sum_{(D_i,D_j)\in M} {}_m w_{ij}\, \iota[n_i \neq n_j] + \sum_{(D_i,D_j)\in C} {}_c w_{ij}\, \iota[n_i = n_j]$$

$$= 1.333 + 0 + 1 = 2.333$$

ดังนั้นค่าความเหมือนระหว่างนิวรอลโหนดที่ 1 (W_1) และเอกสาร D_1 มีค่าเท่ากับ 2.333

2. พิจารณานิวรอลโหนดที่ 2 (W_2)

กำหนดให้ n_1=2 (เนื่องจากเรากำลังพิจารณาที่เอกสาร D_1 กับนิวรอลโหนดที่ 2) และเราทราบว่า n_3=1 และ n_4=1 โครงข่ายที่พิจารณาแสดงดังรูปที่ 3.4 เมื่อข้อมูลอินพุตที่พิจารณาคือ $X = \{x_1, x_2\}$ เมื่อ $x_1 = \{A, E\}$ และ $x_2 = \{C, D\}$



รูปที่ 3.4 แสดงโครงข่ายที่ใช้อธิบายตัวอย่างการทำงานของอัลกอริทึมพิจารณานิวรอลโหนดที่ 2

ค่าความเหมือนกันของนิวรอลโหนดที่ 2 (W_2) และข้อมูลอินพุตยูนิต X คำนวณได้จาก

$$\|X - W_2\| = \sum_{k=1}^{2} D_1(x_i, w_{2k}) + \sum_{(D_i,D_j)\in M}{}_m w_{ij}\, \mathbf{l}[n_i \neq n_j] + \sum_{(D_i,D_j)\in C}{}_c w_{ij}\, \mathbf{l}[n_i = n_j] \qquad (3.16)$$

จากสมการที่ 3.16 จะแยกการอธิบายออกเป็นสามส่วนคือ ส่วนแรกเป็นการคำนวณหาค่าความแตกต่างระหว่างเอกสาร D_1 และนิวรอลโหนดที่ 2 (W_2) ส่วนที่สองเป็นการคำนวณหาค่าการละเมิดกฎที่เป็น Must-link Constraints และส่วนสุดท้ายเป็นการคำนวณหาค่าการละเมิดกฎที่เป็น Cannot-link Constraints ซึ่งแต่ละส่วนมีรายละเอียดดังนี้

1. คำนวณหาค่าความแตกต่างระหว่างระหว่างเอกสาร D_1 กับนิวรอลโหนดที่ 2 (W_2)

$$\begin{aligned}\sum_{k=1}^{2} D(x_k, w_{2k}) &= D(x_1, w_{21}) + D(x_2, w_{22}) \\ &= D_c(x_1, w_{21}) + D_s(x_1, w_{21}) + D_c(x_2, w_{22}) + D_s(x_2, w_{22}) \\ &= \frac{|2-2|}{3} + \frac{|2+2-(2*1)|}{3} + \frac{|2-2|}{3} + \frac{|2+2-(2*2)|}{3} \\ &= 0 + \frac{2}{3} + 0 + 0 = 0.667\end{aligned}$$

2. คำนวณหาค่าการละเมิดกฎที่เป็น Must-link Constraints จากสมการที่ 3.13 แสดงรายละเอียดของกฎที่เป็น Must-link Constraints เนื่องจากกำลังพิจารณาที่เอกสาร D_1 และจากที่เราทราบว่า n_1=2 และ n_3=1 จะได้ค่าที่เกิดจากการละเมิดกฎของเอกสาร D_1 ดังนี้

$$M = \{\boxed{(D_1, D_3)}, (D_5, D_6)\} \qquad {}_mW = \{\boxed{({}_m w_{13}, 1)}, ({}_m w_{56}, 1)\}$$

$$\sum_{(D_x,D_y)\in M}{}_m w_{xy}\, l[n_x \neq n_y] = {}_m w_{13}\, l[n_1 \neq n_3] \quad \text{(True)}$$

$$= 1*1$$

$$= 1$$

3. คำนวณหาค่าการละเมิดกฎที่เป็น Cannot-link Constraints จากสมการที่ 3.14 แสดงรายละเอียดของกฎที่เป็น Cannot-link Constraints เนื่องจากกำลังพิจารณาที่เอกสาร D_1 และจากที่เราทราบว่า n_1=2 และ n_4=1 จะได้ค่าที่เกิดจากการละเมิดกฎของเอกสาร D_1 ดังนี้

$$C = \{\boxed{(D_1, D_4)}, (D_3, D_4)\} \qquad {}_cW = \{\boxed{({}_c w_{14}, 1)}, ({}_c w_{34}, 1)\}$$

$$\sum_{(D_x,D_y)\in C}{}_c w_{xy}\, l[n_x = n_y] = {}_c w_{14}\, l[n_1 = n_4] \quad \text{(False)}$$

$$= 1*0$$

$$= 0$$

จากสมการที่ 3.16 ค่าความเหมือนกันของนิวรอลโหนดที่ 2 (W_j) และเอกสาร D_1 มีค่าเท่ากับ

$$\|X - W_2\| = \sum_{k=1}^{2} D_1(x_i, w_{2k}) + \sum_{(D_i,D_j)\in M}{}_m w_{ij}\, l[n_i \neq n_j] + \sum_{(D_i,D_j)\in C}{}_c w_{ij}\, l[n_i = n_j]$$

$$= 0.667 + 1 + 0 = 1.667$$

ดังนั้นค่าความเหมือนระหว่างนิวรอลโหนดที่ 2 (W_2) และเอกสาร D_1 มีค่าเท่ากับ 1.667

จากการหาค่าความเหมือนกันระหว่างทั้งสองนิวรอลโหนด W_i ข้อมูลอินพุตยูนิต X จะได้ว่านิวรอลโหนดที่ 2 เป็นนิวรอลโหนดที่ชนะ ซึ่งเป็นนิวรอลโหนดที่มีค่าความแตกต่างระหว่างนิวรอลโหนดและข้อมูลอินพุตที่ต่ำที่สุด จากนั้นจึงทำการกำหนดค่า n_1 ให้มีค่าเท่ากับนิวรอลโหนดที่ชนะ กล่าวคือ n_1= 2 ส่วนข้อมูลอินพุตอื่นๆ ก็ทำการคำนวณในลักษณะเดียวกัน หลังจากที่ทราบนิวรอลโหนดที่ชนะของทุกๆ ข้อมูลอินพุตแล้วในขั้นตอนต่อไปจะเป็นการหานิวรอลโหนดที่อยู่ใกล้เคียงกับนิวรอลโหนดที่ถูกเลือก ซึ่งนิวรอลโหนดที่ถูกเลือกและนิวรอลโหนดที่อยู่ใกล้เคียงจะนำมาพิจารณาทำการปรับค่าเวทเวกเตอร์ต่อไป

บทที่ 4
การทดลองและผลการทดลอง

4.1 การวัดประสิทธิภาพของอัลกอริทึม

ในการวัดประสิทธิภาพการทำงานของอัลกอริทึมได้เลือกใช้ตัววัดที่ใช้ในการวิจัยด้านการค้นคืนสารสนเทศ คือ Harmonic Mean หรือ F Measure และ Entropy โดยตัววัด F Measure เป็นตัววัดที่ใช้ในการวัดความความถูกต้องของการจัดกลุ่มข้อมูล ส่วน Entropy เป็นตัววัดเพื่อวัดการซ้อนทับกันของกลุ่มข้อมูล รายละเอียดของตัววัดทั้งสองที่นำมาใช้มีดังนี้

4.1.1 F Measure

ตัววัด F Measure เป็นการคำนวณหาค่า Harmonic Mean ของค่า *Recall* และ *Precision* ค่าของ F Measure ที่ได้นั้นจะมีค่าอยู่ระหว่าง 0 ถึง 1 โดยที่ค่าที่ได้นั้นบ่งบอกว่า ค่าที่ได้มีค่าสูงแสดงว่าผลการจัดกลุ่มมีความถูกต้องสูง การคำนวณหาค่า F Measure *F(i)* ของ cluster ลำดับที่ *i* สามารถคำนวณได้จาก [13]

$$F(i) = \frac{2*Recall(i)*Precision(i)}{Recall(i) + Precision(i)} \tag{4.1}$$

โดยค่า *Recall* และ *Precision* ของคลัสเตอร์ลำดับที่ *i* หาได้จาก

$$Recall(i) = \frac{|Ra_i|}{|R_i|} \tag{4.2}$$

$$Precision(i) = \frac{|Ra_i|}{|A_i|} \tag{4.3}$$

เมื่อ Ra_i คือ จำนวนของเอกสารจากผลการทดลองที่จัดกลุ่มของคลัสเตอร์ลำดับที่ *i*

R_i คือ จำนวนของเอกสารที่เป็นสมาชิกของคลัสเตอร์ลำดับที่ *i*

A_i คือ จำนวนของเอกสารจากผลการทดลองทั้งหมด (ทั้งที่จัดกลุ่มได้ถูกต้องและไม่ถูกต้อง) ของคลัสเตอร์ลำดับที่ *i*

สำหรับค่า F Measure ของทุกๆ คลัสเตอร์หาได้จาก

$$F = \sum_i \frac{R_i}{N} F(i) \tag{4.4}$$

เมื่อ N เป็นจำนวนเอกสารทั้งหมด

4.1.2 Entropy

ตัววัด Entropy เป็นการวัดการซ้อนทับกันของกลุ่ม ซึ่งเป็นการตัววัดคุณภาพอีกอย่างหนึ่งของการทำงานของอัลกอริทึม โดยค่า Entropy จะดีที่สุดเมื่อมีค่าเป็น 0 นั่นคือไม่มีการซ้อนทับกันของข้อมูลเลย หรือผลลัพธ์ของแต่ละคลัสเตอร์ประกอบด้วยข้อมูลที่เป็นสมาชิกเพียงคลาสเดียวเท่านั้น การหาค่า Entropy เริ่มต้นจากการคำนวณหาค่าความน่าจะเป็นที่สมาชิกจากผลการทดลองของคลัสเตอร์ i เป็นสมาชิกของคลาส j ซึ่งแทนความน่าจะเป็นนี้ด้วย p_{ij} ทั้งนี้ค่า Entropy ของแต่ละคลัสเตอร์หาได้จาก [12]

$$E_i = -\sum_j p_{ij} \log(p_{ij}) \tag{4.5}$$

สำหรับค่า Entropy ของทุกๆ cluster หาได้จาก

$$E = \sum_{i=1}^{n} \frac{A_i * E_i}{N} \tag{4.6}$$

เมื่อ A_i คือ จำนวนของเอกสารจากผลการทดลองทั้งหมด (ทั้งที่จัดกลุ่มได้ถูกต้องและไม่ถูกต้อง) ของคลัสเตอร์ลำดับที่ i

n คือ จำนวนของคลัสเตอร์

N คือ จำนวนของเอกสารทั้งหมด

ค่า Entropy ที่ได้นั้นบ่งบอกว่า ค่า Entropy ที่ได้จะมีค่าน้อยเมื่อผลจากการจัดกลุ่มข้อมูลมีการซ้อนทับกันน้อย และในทางตรงกันข้ามค่า Entropy จะมีค่ามากเมื่อผลจากการจัดกลุ่มข้อมูลมีการซ้อนทับกันมาก

4.2 ข้อมูลที่ใช้ในการทดลอง

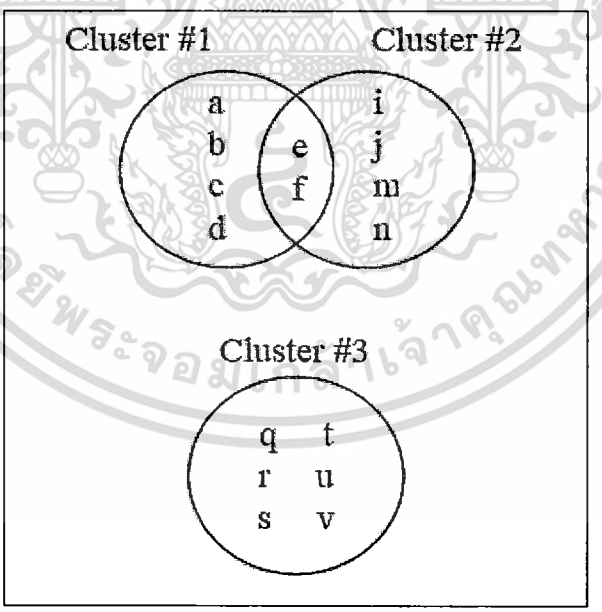
ข้อมูลที่นำมาใช้ในการทดสอบประสิทธิภาพการทำงานของอัลกอริทึมที่นำเสนอประกอบด้วยข้อมูลสองชุด ชุดแรกเป็นข้อมูลที่สร้างขึ้นจากการสุ่มตัวอักษรภาษาอังกฤษซึ่งจะเรียกชุดข้อมูลดังกล่าวนี้ว่า ข้อมูลชุดตัวอักษร ส่วนข้อมูลชุดที่สองเป็นข้อมูลข่าว Reuters-21578 [14] ซึ่งเป็นข้อมูลมาตรฐานชุดหนึ่งที่ใช้สำหรับทดสอบการทำงานของอัลกอริทึมด้านการจัดกลุ่มเอกสาร โดยข้อมูลแต่ละชุดมีรายละเอียดดังนี้

4.2.1 ข้อมูลชุดตัวอักษร

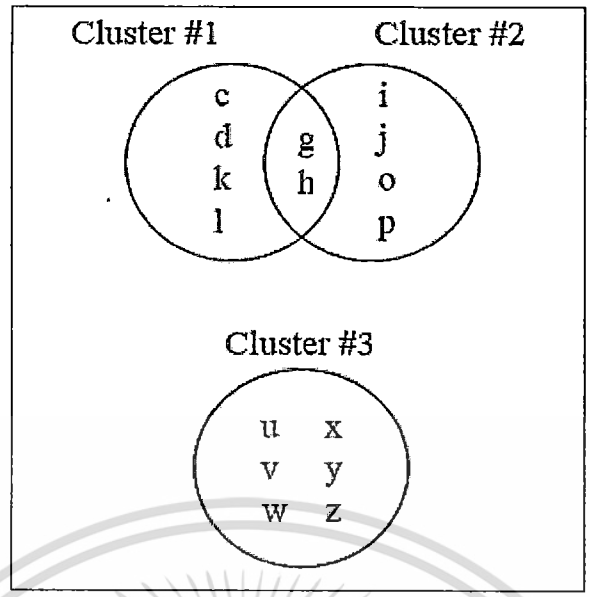
ข้อมูลชุดตัวอักษรนี้สร้างขึ้นด้วยอักษรภาษาอังกฤษ โดยข้อมูลแต่ละตัวอักษรคือตัวแทนของข้อมูลที่มีค่าคุณสมบัติที่เป็นข้อความ ข้อมูลชุดตัวอักษรที่สร้างขึ้นจะสร้างจากกลุ่มของตัวอักษร 3 กลุ่ม คุณสมบัติของแต่ละข้อมูลที่พิจารณานี้ คือ คุณสมบัติหัวเรื่อง (Title) และคุณสมบัติคำสำคัญ (Keyword) กล่าวคือ

$$Doc = Title * Keyword \tag{4.7}$$

กลุ่มตัวอักษรที่ใช้ในการสร้างคุณสมบัติหัวเรื่องและคำสำคัญสำหรับแต่ละข้อมูลชุดตัวอักษรแสดงได้ดังรูปที่ 4.1 และ 4.2



รูปที่ 4.1 แสดงกลุ่มตัวอักษรที่ใช้สร้างข้อมูลในคุณสมบัติหัวเรื่อง



รูปที่ 4.2 แสดงกลุ่มตัวอักษรที่ใช้สร้างข้อมูลในคุณสมบัติคำสำคัญ

จากกลุ่มตัวอักษรที่ใช้การสร้างข้อมูลในทั้งสองคุณสมบัติ คือ คุณสมบัติหัวเรื่อง และคุณสมบัติคำสำคัญ ที่นำมาใช้ในการเทรนนิงประกอบด้วยข้อมูลจำนวน 100 ข้อมูลโดยเป็นสมาชิกของคลัสเตอร์ 1 จำนวน 31 ข้อมูล คลัสเตอร์ 2 จำนวน 34 และคลัสเตอร์ 3 จำนวน 35 ข้อมูล ตัวอย่างของข้อมูลที่ใช้ในการเทรนนิงแสดงได้ดังตารางที่ 4.1

ตารางที่ 4.1 แสดงตัวอย่างข้อมูลที่ใช้ในการเทรนนิง

ลำดับข้อมูล	ค่าของคุณสมบัติ		คลัสเตอร์
	หัวเรื่อง	คำสำคัญ	
1	e, j, m, n	g, h, i, j	2
2	q, r, s, v	u, v, x, y, z	3
3	e, f, j, n	i, j, o, p	2
4	r, t, u, v	u, v, w, x, z	3
5	a, b, e, f	c, d, h, l	1
6	q, t, v	u, x, y	3
7	e, f, i, j	g, h, i, j, p	2
8	j, m	h, i, j, o, p	2
9	q, r, s, u	u, v, w, x, y	3
10	e, i, j, m, n	h, i, j, o, p	2

นอกจากนี้แล้วในการเทรนนิงยังมีการรวมกฎระหว่างสองข้อมูลจำนวน 15 กฎซึ่งเป็นกฎประเภท Must-link Constraints จำนวน 7 กฎ และเป็นกฎประเภท Cannot-link Constraints จำนวน 8 กฎ ข้อมูลของแต่ละกฎแสดงดังตารางที่ 4.2 โดยแต่ละกฎทั้ง Must-link Constraints และ Cannot-link Constraints เขียนแทนด้วยเซต M และ C ตามลำดับดังนี้

$$M = \{(1,11),(2,4),(3,4),(5,8),(6,7),(11,14),(14,15)\}$$
$$C = \{(1,3),(1,4),(1,7),(2,8),(5,15),(7,12),(12,13),(13,14)\}$$

ตารางที่ 4.2 แสดงข้อมูลของแต่ละกฎ

ลำดับข้อมูล	ค่าของคุณสมบัติ		คลัสเตอร์
	หัวเรื่อง	คำสำคัญ	
1	e, f, j, n	i, j, o, p	2
2	r, t, u, v	u, v, w, x, z	3
3	q, t, v	u, x, y	3
4	r, s, t	v, x, y	3
5	a, b, c, d	c, k, l	1
6	b, c, d, e, f	c, g, h, l	1
7	a, d, e, f	c, g, h, l	1
8	a, c, d, e, f	c, d, g, h, k	1
9	f, i, j, m	g, h, i, p	2
10	b, e, f	c, k, l	1
11	e, i, j, m	h, i, o, p	2
12	e, j, m	g, i, j, o, p	2
13	a, b, c, e, f	d, g, h, k, l	1
14	i, m, n	g, i, j, p	2
15	e, i, j, n	g, i, o	2

สำหรับข้อมูลที่นำมาใช้ทดสอบความถูกต้องของโมเดลที่ได้จากการเทรนนิงประกอบด้วยข้อมูลสร้างจากกลุ่มตัวอักษรเดียวกันกับที่ข้อมูลที่ใช้ในการเทรนนิงชุดละ 1000 ข้อมูลจำนวน 5 ชุด รายละเอียดแสดงได้ดังตารางที่ 4.3

ตารางที่ 4.3 แสดงรายละเอียดของชุดข้อมูลที่นำมาใช้ในการทดสอบความถูกต้องของโมเดล

ลำดับ ชุดข้อมูล	จำนวนสมาชิก		
	คลัสเตอร์ 1	คลัสเตอร์ 2	คลัสเตอร์ 3
1	322	320	358
2	322	342	336
3	326	349	325
4	349	328	333
5	335	331	334

4.2.2 ข้อมูลข่าว Reuters-21578

ข้อมูลข่าว Reuters-21578 เป็นชุดข้อมูลที่นิยมใช้ในงานวิจัยทางด้านการค้นคืนสารสนเทศ (Information Retrieval) และการจัดกลุ่มข้อมูลประเภทเอกสาร (Text Clustering) โดยข้อมูลดังกล่าวประกอบด้วยข่าวจำนวน 21578 ข่าวและมีหัวข้อข่าวทั้งหมด 135 หัวข้อ ข้อมูลข่าว Reuters-21578 จะเป็นข้อมูลที่อยู่ในรูปแบบของแฟ้มข้อมูล SGML

จากข้อมูลข่าว Reuters- 21578 ทั้งหมดในการทดลองได้เลือกกลุ่มข่าวทั้งหมดจำนวน 3 และ 5 ตามลำดับ โดยข้อมูลข่าวนี้เป็นข้อมูลชุดเดียวกันกับชุดที่ใช้ในงานวิจัย [7] ซึ่งได้มีการนำข้อมูลข่าวทั้งหมดมาผ่านการเตรียมข้อมูลก่อนที่จะนำไปใช้งานจริง ขั้นตอนการเตรียมข้อมูลขั้นตอนแรกแยกเอาเฉพาะข้อความที่อยู่ภายในแท็ก <Topic> <Title> และ <Body> ของทุกๆข่าว ขั้นตอนที่สองนำตัวเนื้อข่าวที่ได้จากแท็ก <Title> และ <Body> ทั้งหมดมาหาคำสำคัญด้วยโปรแกรม Copernic Summarizer ในการหาคำสำคัญของแต่ละข่าวได้กำหนดจำนวนของคำที่ซ้ำไว้ที่ 10 คำ และขั้นตอนสุดท้ายนำข้อความที่เป็นหัวเรื่องข่าวจากข้อความในส่วนของแท็ก <Title> และคำสำคัญที่ได้จากโปรแกรม Copernic Summarizer มาหารากศัพท์ (Stemming) รวมทั้งตัดคำที่เป็น Stop Word เช่น the, and หรือ something เป็นต้น รายละเอียดข้อมูลที่จะใช้เทรนนิงและทดสอบแสดงดังตารางที่ 4.4 และ 4.5

ตารางที่ 4.4 แสดงชุดข้อมูล 3 กลุ่มข่าว Reuters-21578 สำหรับใช้เทรนนิงและทดสอบของข้อมูล

กลุ่มข่าว	จำนวนข่าวที่ใช้เทรนนิง	จำนวนข่าวที่ใช้ทดสอบ
1. acq	331	1932
2. crude	221	496
3. grain	215	467
รวม	767	2895

ตารางที่ 4.5 แสดงชุดข้อมูล 5 กลุ่มข่าว Reuters-21578 สำหรับใช้เทรนนิงและทดสอบของข้อมูล

กลุ่มข่าว	จำนวนข่าวที่ใช้เทรนนิง	จำนวนข่าวที่ใช้ทดสอบ
1. earn	355	3181
2. acq	331	1932
3. money-fx	239	596
4. crude	221	496
5. grain	215	467
รวม	1361	6672

4.3 ผลการทดลอง

4.3.1 ข้อมูลชุดตัวอักษร

ข้อมูลชุดตัวอักษรที่สร้างขึ้นด้วยอักษรภาษาอังกฤษโดยรายละเอียดได้แสดงไว้ในหัวข้อที่ 4.2 ในการเทรนนิงนี้ประกอบด้วยจำนวนนิวรอลโหนดในชั้นเอาท์พุตจำนวน 4 โหนด โดยในแต่ละนิวรอลโหนดจะประกอบไปด้วยค่าเวท w_{ij} ที่เป็นตัวแทนของคุณสมบัติแต่ละคุณสมบัติของเอกสาร ในที่นี้คือคุณสมบัติที่นำมาพิจารณาคือคุณสมบัติหัวเรื่อง (Title) และคำสำคัญ (Keyword) หลังจากการเทรนนิงแล้วผลที่ได้แสดงดังตารางที่ 4.6

ตารางที่ 4.6 แสดงผลลัพธ์ที่ได้จากการเทรนนิงข้อมูลชุดตัวอักษร

นิวรอลโหนด ลำดับที่	ผลลัพธ์ที่ได้จากการเทรนนิง		ความเป็นตัวแทนของคลัสเตอร์
	หัวเรื่อง	คำสำคัญ	
1	a, b, c, d, e, f, i, j, m, n	g, j, k, l, p	Other
2	e, f, i, j, m, n	g, h, i, j, o, p	2
3	q, r, s, t, u, v	u, v, w, x, y, z	3
4	a, b, c, d, e, f	c, d, g, h, k, l	1

ผลที่ได้หลังจากการเทรนนิง คือได้ค่าเวท w_{ij} ของนิวรอลโหนดซึ่งเป็นตัวแทนของคุณสมบัติหัวเรื่องและคำสำคัญในข้อมูลชุดตัวอักษรที่นำมาใช้ในการเทรนนิง จากตารางนิวรอลโหนดลำดับที่ 2 จะเป็นตัวแทนของคลัสเตอร์ที่ 2 นิวรอลโหนดลำดับที่ 3 เป็นตัวแทนของคลัสเตอร์ที่ 3 และนิวรอลโหนดลำดับที่ 4 จะเป็นตัวแทนของคลัสเตอร์ 1 ส่วนนิวรอลโหนดลำดับที่ 1 นั้นเป็นตัวแทนของคลัสเตอร์อื่นๆ นอกเหนือจากทั้ง 3 คลัสเตอร์

การทดสอบความถูกต้องของนิวรอลเน็ตเวิร์คที่ได้จากการเทรนนิง ข้อมูลที่นำมาใช้ทดสอบความถูกต้องเป็นข้อมูลสร้างจากกลุ่มตัวอักษรเดียวกันกับที่ข้อมูลที่ใช้ในการเทรนนิงชุดละ 1000 ข้อมูลจำนวน 5 ชุด ดังรายละเอียดที่ได้แสดงไว้ดังตารางที่ 4.3 โดยผลลัพธ์ที่ได้จากการทดสอบความถูกต้องของนิวรอลเน็ตเวิร์คแสดงดังตารางที่ 4.7

ตารางที่ 4.7 แสดงผลลัพธ์ที่ได้จากการทดสอบโมเดลด้วยชุดข้อมูลตัวอักษร

ลำดับ	จำนวนสมาชิก			ผลการทดลอง				
ชุดข้อมูล	Cluster1	Cluster2	Cluster3	Cluster1	Cluster2	Cluster3	F Measure	Entropy
1	322	320	358	322	320	358	1.00	0
2	322	342	336	322	342	336	1.00	0
3	326	349	325	326	349	325	1.00	0
4	349	328	333	349	328	333	1.00	0
5	335	331	334	335	331	334	1.00	0
ค่าเฉลี่ย F Measure = 1.00 ค่าเฉลี่ย Entropy = 0								

4.3.2 ข้อมูลข่าว Reuters-21578

ในการเทรนนิงข้อมูลข่าว Reuters-21578 ได้ทำการทดลองเป็น 3 การทดลองด้วยกัน กล่าวคือทำการทดลองกับข้อมูล 3 กลุ่ม และ 5 กลุ่ม โดยรายละเอียดของข้อมูลที่ได้แสดงไว้ดังตารางที่ 4.4 และ 4.5 ตามลำดับ รายละเอียดของแต่ละการทดลองมีดังนี้

1. การทดลองกับข้อมูล 3 กลุ่ม

ในการเทรนนิงข้อมูลข่าว Reuters-21578 จำนวน 3 กลุ่มนี้ประกอบด้วยจำนวนนิวรอลโหนดในชั้นเอาท์พุตจำนวน 3 โหนด การหากฎทั้งที่เป็น Must-link Constraints (M) และ Cannot-link Constraints (C) ทำได้โดยการสุ่มข้อมูลที่ใช้ในการเทรนนิงมา 2 ข้อมูล (d_1, d_2) แล้วทำการตรวจสอบกลุ่มของข้อมูล (Label) โดยพิจารณาดังนี้

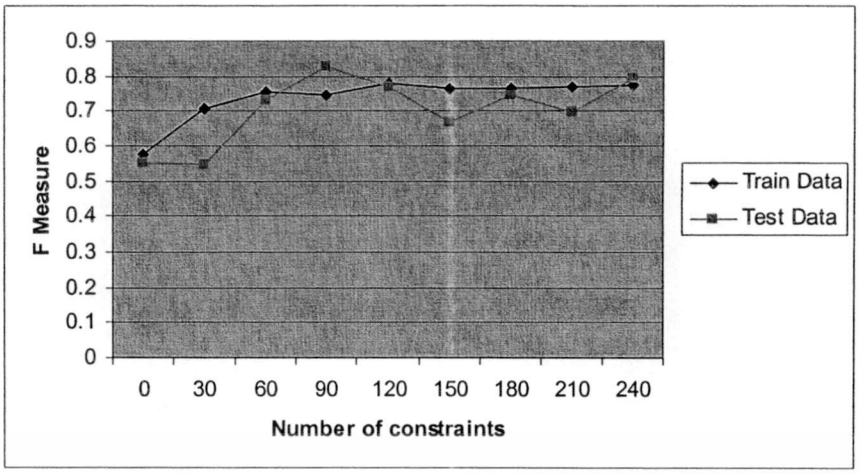
IF d_1.label ~= d_2.label THEN

$(d_1, d_2) \in C$

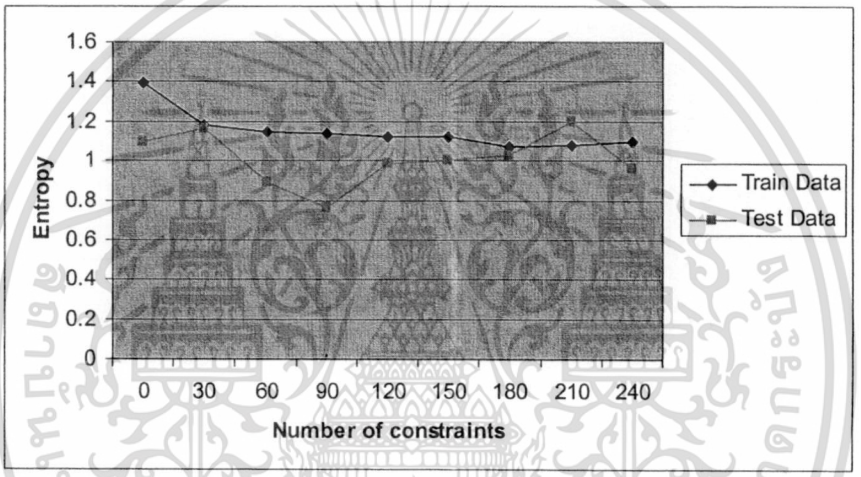
IF d_1.label = d_2.label THEN

$(d_1, d_2) \in M$

หลังจากที่การเทรนนิงสิ้นสุดลง จะนำผลที่ได้มาคำนวณหาค่าความถูกต้องในการจัดกลุ่มข้อมูลโดยคำนวณหาค่า F-Measure และ Entropy ของข้อมูลที่ใช้ในการเทรนนิง และรวมทั้งข้อมูลที่ใช้ในการทดสอบ ผลของค่า F-Measure และ Entropy ของการจัดกลุ่มข้อมูลข่าว 3 กลุ่มแสดงดังรูปที่ 4.3 และ 4.4 ตามลำดับ



รูปที่ 4.3 แสดงค่าของ F Measure ในการทดลองกับข้อมูล 3 กลุ่ม



รูปที่ 4.4 แสดงค่าของ Entropy ในการทดลองกับข้อมูล 3 กลุ่ม

2. การทดลองกับข้อมูล 5 กลุ่ม

ในการเทรนนิงข้อมูลข่าว Reuters-21578 จำนวน 5 กลุ่มนี้ประกอบด้วยจำนวนนิวรอลโหนดในชั้นเอาท์พุตจำนวน 5 โหนด การหากฎทั้งที่เป็น Must-link Constraints (M) และ Cannot-link Constraints (C) ทำในลักษณะเดียวกันกับการหากฎของข้อมูล 3 กลุ่ม โดยสุ่มข้อมูลที่ใช้ในการเทรนนิงมา 2 ข้อมูล (d_1, d_2) แล้วทำการตรวจสอบกลุ่มของข้อมูล (Label) โดยพิจารณาดังนี้

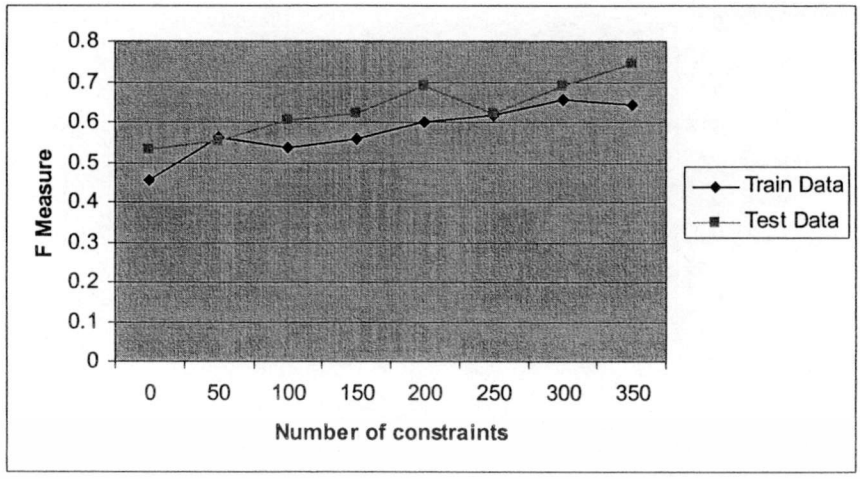
IF d_1.label ~= d_2.label THEN

$(d_1, d_2) \in C$

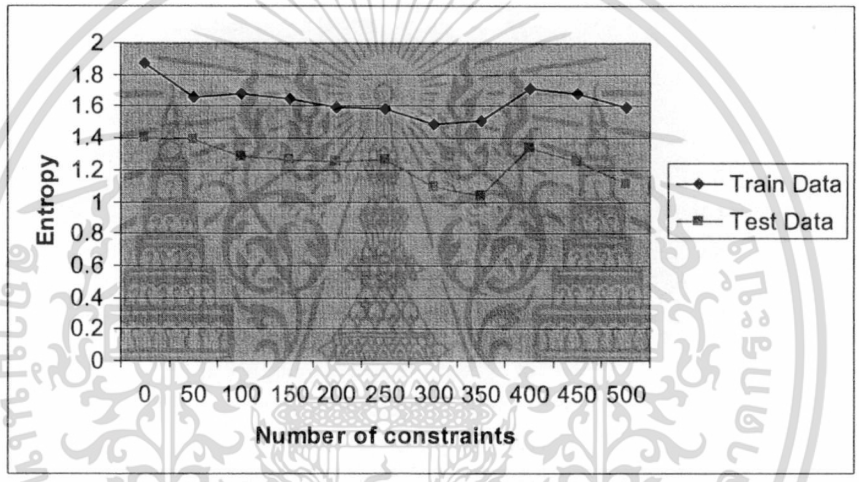
IF d_1.label = d_2.label THEN

$(d_1, d_2) \in M$

ผลการคำนวณหาค่า F-Measure และ Entropy แสดงดังรูปที่ 4.5 และ 4.6 ตามลำดับ



รูปที่ 4.5 แสดงค่าของ F Measure ในการทดลองกับข้อมูล 5 กลุ่ม



รูปที่ 4.6 แสดงค่าของ Entropy ในการทดลองกับข้อมูล 5 กลุ่ม

4.4 สรุปผลการทดลอง

จากผลการทดลองกับข้อมูลชุดตัวอักษรดังตารางที่ 4.7 แสดงให้เห็นว่าอัลกอริทึมที่นำเสนอสามารถทำการแยกกลุ่มข้อมูลได้เป็นอย่างดี โดยค่าเฉลี่ยความถูกต้องในการแยกกลุ่มโดยวัดจากตัววัด F Measure มีค่าเท่ากับ 1.00 และค่าเฉลี่ยการซ้อนทับกันโดยวัดจากตัววัด Entropy มีค่าเท่ากับ 0 แสดงว่าการแยกกลุ่มนี้แต่ละกลุ่มที่ได้ไม่มีข้อมูลของกลุ่มอื่นรวมอยู่ด้วย แสดงให้เห็นว่าประสิทธิของอัลกอริทึมนี้สามารถจัดกลุ่มได้ถูกต้อง 100 เปอร์เซ็นต์ สำหรับในส่วนของชุดข้อมูลข่าว Reuters-21578 จากรูปที่ 4.3 และ 4.4 เป็นผลการทดลองของข้อมูล 3 กลุ่ม เมื่อจำนวนของกฎเพิ่มมากขึ้นแนวโน้มความถูกต้องในการจัดกลุ่มจะมีค่าสูงขึ้น จากผลการทดลองที่มีจำนวนกฎเท่ากับ 90 กฎ ค่า F-measure ของข้อมูลทดสอบมีค่าเท่ากับ 0.8276 และค่า Entropy เท่ากับ 0.7616 ซึ่งประสิทธิภาพของการจัดกลุ่มกับข้อมูลอยู่ในเกณฑ์ที่ดีมีความถูกต้องประมาณ 82.76 เปอร์เซ็นต์

สำหรับผลการทดลองของข้อมูล 5 กลุ่มดังรูปที่ 4.5 และ 4.6 จากผลการทดลองที่มีจำนวนกฎเท่ากับ 350 กฎ ค่าของ F-measure ของข้อมูลทดสอบมีค่าเท่ากับ 0.7440 และค่า Entropy เท่ากับ 1.1165 ซึ่งประสิทธิภาพของการจัดกลุ่มกับข้อมูลยังถือว่าอยู่ในเกณฑ์ที่ดีมีความถูกต้องประมาณ 74.40 เปอร์เซ็นต์

เมื่อเปรียบเทียบผลที่ได้จากการจัดกลุ่มข้อมูล Reuters-21578 ระหว่างการจัดกลุ่มเอกสารโดยใช้โคโฮเนนนิวรอลเน็ตเวิร์คร่วมกับกฎระหว่างสองข้อมูล และเท็กโปรเซสซิงโคโฮเนนนิวรอลเน็ตเวิร์คจากงานวิจัย [7] ในการจัดกลุ่มข้อมูล 3 กลุ่ม และ 5 กลุ่ม วัดค่า F-Measure ได้ 0.80 และ 0.67 ตามลำดับ พบว่าค่าของ F-Measure ที่ได้จากอัลกอริทึมที่นำเสนอมีค่าใกล้เคียงและสูงกว่าทั้งข้อมูล 3 กลุ่ม และ 5 กลุ่ม แสดงให้เห็นว่าการนำกฎระหว่างสองข้อมูลมาช่วยในการจัดกลุ่มข้อมูลแบบไม่มีการชี้นำนั้นสามารถประสิทธิภาพในการจัดกลุ่มข้อมูลเพิ่มมากขึ้น

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

งานวิจัยการจัดกลุ่มเอกสารโดยใช้โคโฮเนนนิวรอลเน็ตเวิร์คร่วมกับกฎระหว่างสองข้อมูล เป็นการพัฒนาปรับปรุงเทคโปรเซสซิงโคโฮเนนนิวรอลเน็ตเวิร์ค ซึ่งเป็นเทคนิคการจัดกลุ่มแบบไม่มีการชี้นำให้มีประสิทธิภาพมากขึ้น โดยนำแนวคิดของโคโฮเนนเซ็ลฟออร์แกไนซิงแม็ปมาขยายความสามารถเพื่อทำการจัดกลุ่มข้อมูลประเภทข้อความได้โดยตรงโดยทำการประยุกต์แนวคิดเรื่องการเปรียบเทียบความแตกต่างของข้อมูลเชิงสัญลักษณ์ รวมทั้งนำแนวคิดการจัดกลุ่มข้อมูลโดยใช้กฎระหว่างสองข้อมูลมาใช้ในการจัดกลุ่มข้อมูลประเภทข้อความด้วย การทำงานของอัลกอริทึมการจัดกลุ่มเอกสารโดยใช้โคโฮเนนนิวรอลเน็ตเวิร์คร่วมกับกฎระหว่างสองข้อมูลประกอบด้วยขั้นตอนหลักที่สำคัญ 3 ขั้นตอนคือ

1. ขั้นตอนการเรียนรู้แบบแข่งขัน (Competitive Process) เป็นส่วนที่ประยุกต์แนวคิดเรื่องการหาค่าความแตกต่างของเอกสาร และแนวคิดกฎระหว่างสองข้อมูลเข้ากับการทำงานของ Competitive Learning โดยมีขั้นตอนการทำงานหลักคือ การหานิวรอลโหนด W_i ที่มีความเหมือนกันกับข้อมูลเข้าอินพุตยูนิต X มากที่สุด โดยการหาค่าความเหมือนกันของนิวรอลโหนด W_i และข้อมูลอินพุตยูนิต X หาได้จากค่าความแตกต่างรวมของแต่ละคุณสมบัติของทั้งสองรวมกับผลรวมของการละเมิดกฎ หรือ Constraint ในส่วนของ Must-link Constraint และ Cannot-link Constraint นิวรอลโหนดที่ถูกเลือกนั้นจะเป็นนิวรอลโหนดที่มีค่าความแตกต่างกับข้อมูลอินพุตยูนิต X น้อยที่สุด

2. ขั้นตอนการหานิวรอลโหนดใกล้เคียง (Cooperative Process) เมื่อทราบนิวรอลโหนดที่มีค่าความแตกต่างกับข้อมูลอินพุตยูนิต X หรือนิวรอลโหนดที่ถูกเลือก ในส่วนนี้จะเป็นการหานิวรอลโหนดที่อยู่ใกล้เคียงกับนิวรอลโหนดที่ถูกเลือก ซึ่งจำนวนของนิวรอลโหนดที่อยู่ใกล้เคียงจะค่อยลดลงตามจำนวนรอบของการเทรนนิงขึ้นอยู่กับ Neighborhood Function ($\wedge$) จะเป็นตัวกำหนด

3. ขั้นตอนการปรับค่าเวทเวกเตอร์ (Adaptive Process) เป็นส่วนของการปรับค่าเวทเวกเตอร์ของนิวรอลโหนด และนิวรอลโหนดใกล้เคียง เพื่อให้มีความใกล้เคียงกับอินพุตยูนิต X มากขึ้น

การวัดประสิทธิภาพของโมเดลของงานวิจัยนี้ได้เลือกตัววัดประสิทธิภาพ F-Measure และ Entropy โดยที่ตัววัด F-Measure เป็นการวัดความถูกต้องของการจัดกลุ่มข้อมูล หากค่าที่ได้มีค่าสูง

แสดงว่าประสิทธิภาพของการจัดกลุ่มข้อมูลมีความถูกต้องสูง ส่วนตัววัด Entropy เป็นตัววัดการซ้อนทับกันของข้อมูล โดยค่าที่ได้มีค่าน้อยแสดงว่าการจัดกลุ่มข้อมูลเกิดการซ้อนทับกันน้อย จากผลการทดลองกับข้อมูลชุดตัวอักษรแสดงให้เห็นว่าอัลกอริทึมที่นำเสนอในงานวิจัยนี้สามารถจัดกลุ่มข้อมูลได้อย่างถูกต้องและมีประสิทธิภาพ ในขณะที่เมื่อทำการทดลองกับข้อมูลข่าว Reuters-21578 ผลการทดลองลดต่ำลง แต่เมื่อเปรียบเทียบผลการจัดกลุ่มข้อมูลจากงานวิจัยใน [4] คือเท็กโปรเซสซิงโคโฮเนนนิวรอลเน็ตเวิร์คซึ่งเป็นการจัดกลุ่มข้อมูลแบบไม่มีการชี้นำ เมื่อนำกฎระหว่างสองข้อมูลมาช่วยในการจัดกลุ่มข้อมูลแบบไม่มีการชี้นำตามที่นำเสนอในงานวิจัยนี้สามารถจัดกลุ่มข้อมูลได้มีประสิทธิภาพเพิ่มขึ้น

5.2 ปัญหาที่พบในงานวิจัยนี้

ในงานวิจัยนี้จากที่ได้ทดลองจัดกลุ่มข้อมูลข่าว Reuters-21578 ปัญหาที่พบคือ การเลือกข้อมูลที่จะนำมาสร้างเป็นกฎ หรือ Constraints นั้นทำโดยพิจารณาเพียงกลุ่มของข้อมูลเท่านั้น เนื่องจากข้อมูลที่อยู่ต่างกลุ่มกันอาจจะมีคุณลักษณะหัวเรื่อง หรือคำสำคัญที่เหมือนกัน จึงทำให้ผลการจัดกลุ่มข้อมูลมีประสิทธิภาพไม่สูงเท่าที่ควร และถ้าหากเราทราบข้อมูลข่าวที่อยู่ต่างกลุ่มกันแต่มีส่วนของคุณลักษณะที่เป็นตัวแทนของข้อมูลที่ซ้ำกัน กล่าวคือมีหัวเรื่อง หรือคำสำคัญซ้ำกัน แล้วนำข้อมูลดังกล่าวนี้มาสร้างเป็น Constraints ก็จะทำให้การจัดข้อมูลได้มีความถูกต้องเพิ่มมากขึ้น

5.3 แนวทางการพัฒนาในอนาคต

จากที่กล่าวมาแล้วว่าปัญหาที่พบในงานวิจัยนี้คือ การเลือกข้อมูลที่จะนำมาสร้างเป็น Constraints แนวทางการพัฒนาในอนาคตคือ วิธีการพิจารณาข้อมูลที่จะนำมาสร้างเป็น Constraints เพิ่มเติมจากที่พิจารณาจากกลุ่มของข้อมูล เช่น อาจจะพิจารณาจากคุณลักษณะของข้อมูลเนื่องจากอัลกอริทึมนี้ดำเนินการกับข้อมูลโดยใช้คุณลักษณะเป็นสำคัญ นอกจากนี้แล้วงานวิจัยนี้ได้ทดลองเฉพาะข้อมูลที่เป็นข้อมูลข่าว Reuters-21578 เท่านั้น ขั้นต่อไปควรจะได้นำไปทดลองกับข้อมูลที่มีลักษณะของเนื้อหาแบบอื่น เช่น เนื้อหาเวปบนอินเตอร์เน็ต เป็นต้น รวมทั้งยังสามารถนำอัลกอริทึมจากงานวิจัยนี้ไปประยุกต์เพื่อพัฒนาเป็นแอพพลิเคชันสำหรับใช้เป็นเครื่องมือช่วยในการค้นหาข้อมูล ตัวอย่างเช่น เสิร์จเอนจิน เป็นต้น

เอกสารอ้างอิง

[1] Basu, S., Bilenko, M. and Mooney, J.R., "Comparing and Unifying Search-Based and Similarity-Based Approaches to Semi-Supervised Clustering", **Proceedings of the ICML-2003 Workshop on the Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining System**, 2003. pp.42-49.

[2] Basu, S., Banerjee, A. and Mooney, J.R., "Active Semi-Supervision for Pairwise Constrained Clustering", **Proceedings of the SIAM SDM-2004**, 2004.

[3] Wagstaff, K. and Cardie, C., "Clustering with Instance-Level Constraints", **Proceedings of the Seventeenth International Conference on Machine Learning**, 2000. pp.1103-1110.

[4] Wagstaff, K., Rogers, S. and Schroedl, S., "Constrained K-Means Clustering with Background Knowledge", **Proceedings of the Eighteenth International Conference on Machine Learning**, 2001. pp.577-584.

[5] Kohonen, T, "The Self-Organizing Map", **Proceedings of the IEEE**, vol.78, no.9, 1990. pp.1446-1480.

[6] Endo, M., Ueno, M., Tanabe, T. and Yamamoto M., "Clustering Method using Self-Organizing Map", **Proceedings of the 2000 IEEE Signal Processing Society Workshop**, vol.1, 2000. pp.261-270.

[7] ทรงพล ชุติพงศ์พัฒนกุล. "เท็กโปรเซสซิ่งโคโฮเนนนิวรอลเน็ตเวิร์คโดยใช้กระบวนการเรียนรู้แนวใหม่." วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต สาขาเทคโนโลยีสารสนเทศ บัณฑิตวิทยาลัย, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2546.

[8] Wagstaff, K.L., **"Intelligent Clustering with Instance-level Constraints"**, Ph.D. Thesis of Cornell University. 2002.

[9] Gowda, K.C. and Diday, E., "Symbolic Clustering using a New Similarity Measure", **IEEE Transactions on System, Man, and Cybernetics**, vol.22, no. 2, 1992.

[10] El-Sonbaty, Y. and Ismail, M.A., "Fuzzy Clustering for Symbolic Data", **IEEE Transactions on Fuzzy System**, vol.6, no.2, 1998. pp.195-204.

[11] Anderson, D. and McNeill, G., **Artificial Neural Networks Technology.** New York, Inc. 1992.

[12] Jain, A.K., Murty M.N. and Flynn, P.J., "Data Clustering: A Review", **ACM Computing Surveys,** vol.31, no. 3, 1999.

[13] Baeza-Yates, R., **Modern Information Retrieval.** New York : Addison-Wesley, Inc. 1999.

[14] Lewis D.D. **"Reuters-21578 text categorization test collection distribution 1.0."** [Online]. Available : http://daviddlewis.com, 2004.

ภาคผนวก ก.

ผลงานวิจัยที่ได้รับการตีพิมพ์

1. วรพจน์ กรีสุระเดช และ อภิญญา สุวรรณละมัย. "เท็กโปรเซสซิงโคโฮเนนนิวรอลเน็ตเวิร์คโดยการใช้กฎระหว่างสองข้อมูล." **การประชุมวิชาการวิทยาการคอมพิวเตอร์และวิศวกรรมคอมพิวเตอร์ ภาคตะวันออกเฉียงเหนือ ครั้งที่ 1 (NE-CSEC2005)**, เมษายน 2548. หน้า 277-282.
2. Worapoj Kreesuradej and Apinya Suwanlamai. "Document Clustering with Pairwise Constraints." **Proceedings of the 2005 IEEE International Conference on Intelligent Computing (ICIC2005),"** 23-26 August 2005.
3. Worapoj Kreesuradej and Apinya Suwanlamai. "Document Clustering with Pairwise Constraints. "**International Journal of Pattern Regcognition and Artificial Intelligence,"** vol. 20, no. 2, 2006, pp. 241-254.