การจัดกลุ่มเอกสารโดยใช้ Self-Organizing Map แบบความเร็วสูง

HIGH SPEED SELF-ORGANIZING MAP FOR DOCUMENT CLUSTERING

พรเทพ โรจนวสุ

PORNTHEP ROJANAVASU

วิทยานิพนธ์นี้เป็นส่วนหนึ่งตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2547

ISBN 974-9680-15-4

HIGH SPEED SELF-ORGANIZING MAP FOR DOCUMENT CLUSTERING

PORNTHEP ROJANAVASU

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF ENGINEERING IN COMPUTER ENGINEERING

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2004

ISBN 974-9680-15-4

หัวข้อวิทยานิพนธ์	การจัดกลุ่มเอกสารโดยใช้ Self-Organizing Map แบบความเร็วสูง
นักศึกษา	นาย พรเทพ โรจนวสุ
รหัสนักศึกษา	44061612
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
พ.ศ.	2547
อาจารย์ผู้ควบคุมวิทยานิพนธ์	รศ.ดร. เอื้อน ปิ่นเงิน

บทคัดย่อ

Self-Oranizing Map(SOM) เป็นนิวรอนเน็ตเวิร์กแบบไม่มีผู้สอน จุดเด่นคือเหมาะสำหรับการนำมาวิเคราะห์ข้อมูล โดยข้อมูลที่ได้จะอยู่ในรูปแบบของกริดที่สามารถแสดงเป็นแผนภาพช่วยในการวิเคราะห์ข้อมูลได้เป็นอย่างดี จุดอ่อนของแผนภาพ SOM แบบดั้งเดิมคือในกรณีที่แผนภาพมีขนาดใหญ่จะใช้เวลาในการเรียนรู้นาน เวลาที่ใช้ส่วนใหญ่คือการหาโหนดชนะ(winning node)ซึ่งเวลาที่ใช้เท่ากับ O(MN) (เมื่อ M,N เป็นจำนวนโหนดแนวกว้างและยาวของแผนภาพตามลำดับ) งานวิจัยนี้นำเสนอวิธีการลดเวลาในการคำนวณโดยเพิ่มแผนภาพใหม่ขึ้นอีกชั้นหนึ่ง ซึ่งแต่ละโหนดในแผนภาพใหม่คือค่าจุดศูนย์กลางมวลของกลุ่มโหนดในแผนภาพเดิม ในกระบวนการเรียนรู้ใหม่การคำนวณหาโหนดชนะจะคำนวณหาจากแผนภาพใหม่ก่อน จากนั้นจะคำนวณหาโหนดชนะที่แท้จริงภายในกลุ่มอีกครั้งหนึ่ง วิธีการใหม่นี้เรียกว่า High Speed Self-Organzing Map(HS-SOM) ในการทดลองได้ใช้ HS-SOM จัดกลุ่มเอกสารเปรียบเทียบกับ SOM แบบดั้งเดิม ผลปรากฏว่าสามารถลดเวลาในการจัดเอกสารลงในแผนภาพได้มากกว่า 20 เปอร์เซ็นต์

Thesis Title	High Speed Self-Organizing Map for Document Clustering
Student	Mr. Pornthep Rojanavasu
Student ID.	44061612
Degree	Master of Engineering
Programme	Computer engineering
Year	2004
Thesis Advisor	Assoc. Prof. Dr. Ouen Pinngern

ABSTRACT

Self-Oranizing Map(SOM) is an unsupervised neural network providing cluster analysis of high dimensional input data. Outputs from SOM are represented in map that helps us to explore data. The weak point of conventional SOM is when the map is large, it takes a longer time to train the system. The computing time taken is O(MN) for trainning to find the winning node (M,N are the number of nodes in width and height of the map). This research presents the new method to reduce the computing time by creating new map insert between input layer and output layer. Each node in the new map is the centroid of nodes' group that are in the original map. In new learning process, we find the winning node in new map, then find the winning node in original map only the nodes that are represented by the winning node from the new map. This new method is called "High Speed Self-Oranizing Map"(HS-SOM). The result from the experiment shows that HS-SOM can reduce computing time up to 20 percent over the conventional SOM.

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงด้วยดีเนื่องจากกำลังใจและพระคุณอันหาที่สุดมิได้ จากคุณพ่อ คุณแม่ คุณยาย และคุณตาผู้ล่วงลับ ข้าพเจ้าขอสำนึกในพระคุณนี้อย่างเป็นที่สุด

วิทยานิพนธ์นี้จะไม่สำเร็จลุล่วงหากปราศจากแรงผลักดัน และคำแนะนำที่มีประโยชน์ของ รศ.ดร. เอื้อน ปิ่นเงิน ผู้ควบคุมวิทยานิพนธ์ ข้าพเจ้าขอกราบขอบพระคุณเป็นอย่างสูง

ข้าพเจ้าขอกราบเท้า คุณครูและอาจารย์ทุกท่านตั้งแต่เล็กจนเติบใหญ่ ที่ได้มอบวิชาความรู้ให้แก่ข้าพเจ้า รวมทั้งคำสั่งสอนและอบรมให้ข้าพเจ้าเป็นคนดี ข้าพเจ้าขอกราบขอบพระคุณเป็นอย่างสูง

ข้าพเจ้าขอขอบคุณสำหรับกำลังใจ คำแนะนำ และประสบการณ์ที่ดีจากพี่ ๆ และเพื่อน ๆ นักศึกษาป.โททุกท่าน สามปีที่ลาดกระบังข้าพเจ้าจะไม่มีวันลืม และขอขอบคุณ นางสาว ปองเกษม พลสันติกุล ที่ช่วยแก้ไขภาษาในการส่งบทความตีพิมพ์ต่างประเทศ ข้าพเจ้าขอขอบคุณ

สุดท้ายนี้คุณค่าและประโยชน์อันพึงมีจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบให้กับผู้มีพระคุณทุกท่าน หากวิทยานิพนธ์ฉบับนี้มีข้อผิดพลาดประการใดข้าพเจ้าขอน้อมรับไว้เพียงผู้เดียว

พรเทพ โรจนวสุ

สารบัญ(ต่อ)

หน้า

สารบัญ(ต่อ)

หน้า

สารบัญตาราง

สารบัญรูป

สารบัญรูป(ต่อ)

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เซลฟ์ออร์แกนไนสซิ่งแม็พ (Self–Organizing Map) หรือ SOM เป็นนิวรอนเน็ตเวิร์กแบบไม่มีผู้สอน (unsupervised neural network) ถูกนำเสนอโมเดลขึ้นในปี ค.ศ. 1982 โดยศาสตราจารย์โคโฮเนน [1] มีคุณสมบัติที่สำคัญคือ สามารถแสดงผลข้อมูลที่มีมิติสูงให้อยู่ในรูปแบบของแผนภาพสองมิติ [2] หลังจากผ่านกระบวนการเรียนรู้แล้ว แผนภาพที่ได้จะอยู่ในลักษณะของแผนภาพจัดเรียงตัว(ordered map) กล่าวคือข้อมูลที่ใกล้เคียงกันจะถูกจัดลงในแผนภาพบริเวณใกล้เคียงกัน ช่วยในการวิเคราะห์คุณลักษณะของข้อมูลได้ เช่น การกระจาย ความหนาแน่น และความสัมพันธ์ของข้อมูล

ด้วยเหตุนี้จึงทำให้ SOM ถูกนำมาใช้อย่างกว้างขวางในหลาย ๆ ด้าน เช่น การประยุกต์ใช้ SOM ในงานควบคุมมอเตอร์ [3] การประยุกต์ใช้ SOM ในการจดจำเสียง [4] การประยุกต์ใช้ SOM ในการจัดกลุ่มข่าวบนอินเตอร์เน็ต [5,6] เป็นต้น นอกจากยังมีงานวิจัยที่ปรับปรุงโมเดลเดิมของ SOM เช่น การนำเอาเจเนติกอัลกอริทึม(genetic algorithm) เข้ามาช่วยในการปรับปรุงโมเดล [3] และ การขยายโหนดของแผนภาพแบบอัตโนมัติ [7,8] นอกจากนี้แล้วยังมีงานวิจัยอื่น ๆ ที่ใช้ประยุกต์ใช้งาน หรือปรับปรุงประสิทธิภาพของ SOM และได้รับการตีพิมพ์ผลงานอีกมากกว่า 3,000 บทความในช่วงปี ค.ศ. 1981-1997 [9]

อย่างไรก็ตามปัญหาที่สำคัญในการประยุกต์ใช้งาน SOM คือเวลาในการเรียนรู้ ซึ่งจะใช้เวลาในการเรียนรู้ค่อนข้างนาน ยิ่งถ้าข้อมูลมีมิติสูงเช่น เวคเตอร์ของเอกสารจะมีขนาดเวคเตอร์มากกว่าพันมิติ จะยิ่งใช้เวลาในการเรียนรู้นานยิ่งขึ้น จึงมีงานวิจัยที่พัฒนาลดเวลาในการเรียนรู้ เช่น การนำเอาอัลกอริทึม K-means ช่วยในการจัดกลุ่มข้อมูลก่อนนำไปเรียนรู้ในแผนภาพเพื่อให้แผนภาพจัดเรียงตัวได้เร็วขึ้นในจำนวนรอบที่น้อยกว่า [3] การลดการคำนวณในการหาโหนดชนะ (winning node) โดยแบ่งเป็นส่วนย่อย ๆ ที่ใช้ในการค้นหา ด้วยวิธีนี้สามารถลดเวลาในการคำนวณได้สูงสุดถึง 14 เปอร์เซ็นต์ [10]

ในงานวิจัยนี้ได้นำเสนอวิธีการลดเวลาในการคำนวณอีกรูปแบบหนึ่ง โดยพิจารณาที่เวลาในการหาโหนดชนะ แนวคิดใหม่ที่นำเสนอคือ การรวมโหนดต่าง ๆ เป็นกลุ่มย่อย ๆ และใช้โหนดตัวแทนของกลุ่มคำนวณหาโหนดตัวแทนที่ชนะก่อน จากนั้นจะคำนวณหาโหนดที่ชนะภายในกลุ่มของโหนดตัวแทนนั้นอีกครั้งหนึ่ง โดยโหนดตัวแทนนั้นเมื่อรวมกันหลายๆโหนดจะกลายเป็นแผนภาพใหม่ขั้นกลางระหว่างชั้นของอินพุตกับชั้นของเอาท์พุตหรือแผนภาพจริง ในกรณีที่แผนภาพมีความกว้างและความยาวมากเราสามารถแทรกแผนภาพใหม่ได้หลาย ๆ ชั้น โมเดลใหม่ที่

ผู้วิจัยนำเสนอให้ชื่อว่า SOM แบบความเร็วสูงหรือ High Speed Self-Organizing Map (HS-SOM) ด้วยวิธีการนี้เราสามารถที่จะลดเวลาการเรียนรู้ได้มากกว่า 30 เปอร์เซ็นต์โดยที่ประสิทธิภาพที่ได้ใกล้เคียงกับโมเดลเดิม นอกจากนั้นแผนภาพใหม่ที่สร้างขึ้นยังสามารถช่วยให้ผู้ใช้สำรวจ(browse) ข้อมูลแผนภาพจริงได้อย่างสะดวก

1.2 ความมุ่งหมายและวัตถุประสงค์ของการวิจัย

1. เพื่อศึกษาการโมเดล SOM อัลกอริทึมการเรียนรู้ของ SOM และขีดจำกัดของโมเดล SOM
2. เพื่อศึกษาแนวทางในการพัฒนา SOM และนำเสนอโมเดลใหม่ที่ผู้วิจัยได้พัฒนาขึ้น
3. เพื่อศึกษาการจัดกลุ่มเอกสารและการประยุกต์ใช้ SOM ในการจัดกลุ่มเอกสาร
4. เพื่อเปรียบเทียบประสิทธิภาพโมเดล SOM แบบเดิมและโมเดลใหม่ที่ผู้วิจัยได้พัฒนาขึ้นในการจัดกลุ่มเอกสาร

1.3 ขอบเขตของการวิจัย

1. ศึกษาเปรียบเทียบความเร็วในการเรียนรู้ และประสิทธิภาพในการจัดกลุ่มเอกสารของโมเดล SOM แบบดั้งเดิมกับโมเดลใหม่ที่ผู้วิจัยได้พัฒนาขึ้น
2. เอกสารที่ใช้ในการทดสอบเป็นบทความภาษาอังกฤษ โดยเลือกเฉพาะหัวเรื่องกับบทคัดย่อของเอกสารเท่านั้น

1.4 ขั้นตอนการศึกษา

1. ศึกษานิวรอนเน็ตเวิร์คโมเดล SOM
2. ศึกษางานวิจัยที่เกี่ยวข้องกับการประยุกต์ใช้ SOM ในงานจัดกลุ่มเอกสาร พร้อมทั้งวิเคราะห์ข้อดีข้อเสียของโมเดล SOM
3. พัฒนาโมเดลใหม่ที่มีชื่อว่า SOM แบบความเร็วสูงหรือ High Speed Self-Organizing Map (HS-SOM)
4. ทดสอบโมเดล SOM แบบเดิมและโมเดล HS-SOM ในการจัดกลุ่มเอกสารพร้อมทั้งปรับปรุงโมเดลใหม่
5. สรุปผลการทดลองพร้อมจัดทำบทความตีพิมพ์และวิทยานิพนธ์

1.5 รายละเอียดในแต่ละบท

ในวิทยานิพนธ์ฉบับนี้แบ่งเนื้อหาการนำเสนอออกเป็น 6 บทดังนี้

- บทที่ 1 กล่าวถึงความเป็นมาและความสำคัญของปัญหา แนวคิดที่นำเสนอเพื่อปรับปรุงโมเดลดั้งเดิม วัตถุประสงค์และขอบเขตของงานวิจัย

- บทที่ 2 กล่าวถึงนิวรอนเน็ตเวิร์คแบบไม่มีผู้สอน โดยจะเน้นที่โมเดลของโคโฮเนนที่มีชื่อว่า Self-Organizing Map(SOM) ซึ่งเป็นโมเดลที่งานวิจัยนี้นำมาพัฒนาต่อ
- บทที่ 3 นำเสนอโมเดลใหม่ที่มีชื่อว่า High Speed Self-Organizing Map (HS-SOM) ซึ่งเป็นโมเดลที่ได้รับการปรับปรุงจากโมเดล SOM แบบดั้งเดิมเพื่อที่จะลดระยะเวลาในการเรียนรู้ โดยที่ประสิทธิภาพของโมเดลยังคงเดิมอยู่
- บทที่ 4 กล่าวถึงอัลกอริทึมในการจัดกลุ่มเอกสารแบบต่าง ๆ โดยย่อและการจัดกลุ่มเอกสารโดยใช้แผนภาพ SOM
- บทที่ 5 กล่าวถึงการดำเนินการทดลองและผลการทดลอง
- บทที่ 6 กล่าวถึงการสรุป วิเคราะห์ผลการทดลอง รวมทั้งข้อเสนอแนะ และแนวทางการทำวิจัยต่อ

บทที่ 2

นิวรอนเน็ตเวิร์กแบบไม่มีผู้สอน

ในบทนี้จะกล่าวถึงโมเดลนิวรอนเน็ตเวิร์กของโคโฮเนน (Kohonen) ซึ่งเป็นนิวรอนเน็ตเวิร์กแบบไม่มีผู้สอนที่ได้รับความนิยมเป็นอย่างมาก โดยจะอธิบายถึงการทำงานของเซลล์ในเน็ตเวิร์กเมื่อได้รับอินพุตเข้ามา ขั้นตอนต่าง ๆ ในการเรียนรู้รวมทั้งแสดงตัวอย่างงานวิจัยที่นำเอาโมเดลนี้ไปประยุกต์ใช้ในด้านต่าง ๆ

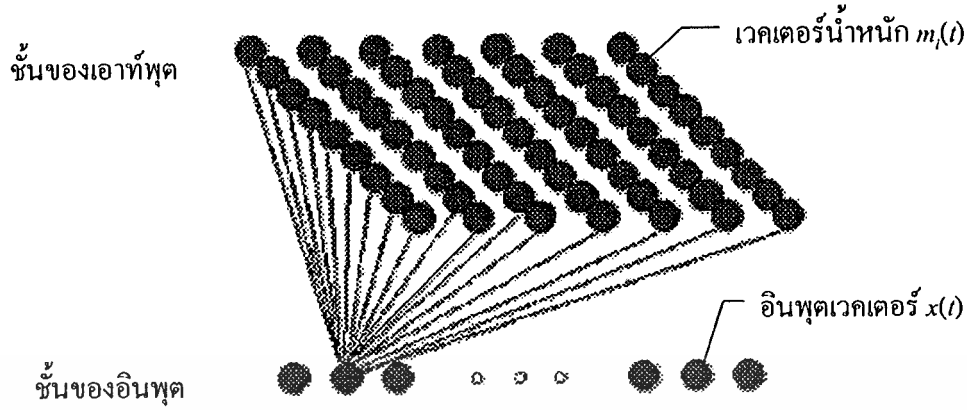
2.1 บทนำ

เซลฟ์ออร์แกนไนสซิ่งแม็พ (Self-Organzing Map) หรือ SOM เป็นนิวรอนเน็ตเวิร์กแบบไม่มีผู้สอนประเภทหนึ่ง [1] ซึ่งแตกต่างจากนิวรอนเน็ตเวิร์กแบบมัลติเลเยอร์เพอเซพตอล (Multi Layer Percepton MLP) หรือแบบแบ็คพรอพาเกชัน (Backpropagation) ซึ่งเป็นนิวรอนเน็ตเวิร์กแบบมีผู้สอน อัลกอริทึมของ SOM ถูกนำเสนอโดยศาสตราจารย์โคโฮเนน (Kohonen) ในปี ค.ศ. 1982 ซึ่งรู้จักกันในชื่อ แผนภาพคุณลักษณะของโคโฮเนน(Kohonen feature map)

SOM เป็นอัลกอริทึมที่ใช้ในการจัดกลุ่มข้อมูล โดยสามารถจัดกลุ่มข้อมูลที่มีมิติสูงให้อยู่ในรูปของแผนภาพ 2 มิติซึ่งประกอบไปด้วยโหนดของนิวรอน ข้อมูลจะถูกจัดลงในโหนดต่าง ๆ ของแผนภาพ หลังจากเสร็จสิ้นกระบวนการเรียนรู้แผนภาพจะถูกจัดเรียงตัว โดยข้อมูลที่มีความคล้ายคลึงกันจะอยู่ในกลุ่มโหนดใกล้เคียงกัน ดังนั้นแผนภาพที่ได้จะแสดงคุณลักษณะของข้อมูลได้เป็นอย่างดี เช่น การกระจายของข้อมูล ความสัมพันธ์ระหว่างข้อมูลในกลุ่มเป็นต้น

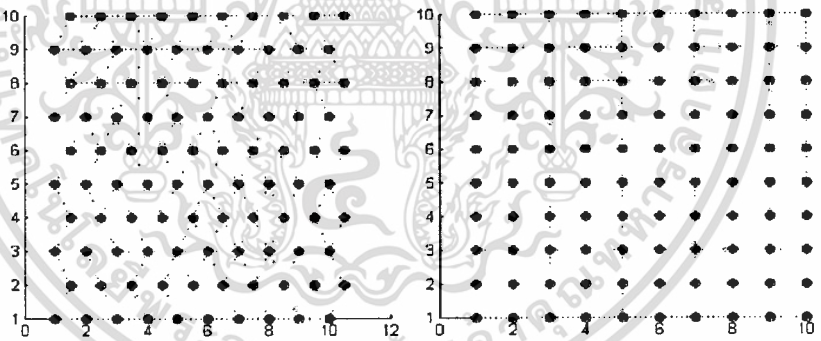
2.2 โมเดลและอัลกอริทึมของ Self-Organizing Map

โมเดลของ SOM ประกอบด้วยเซลล์ 2 ชั้น [2] ดังรูปที่ 2.1 ชั้นแรกคือชั้นของอินพุต(Input layer)ประกอบไปด้วยเซตของอินพุตเวคเตอร์ $x(t)$ ที่มีขนาด n มิติ(1 x n มิติ) ซึ่งเป็นอินพุตที่ใช้ในการเรียนรู้ของแผนภาพ โดยที่ t คืออินเด็กซ์ของอินพุตหรือแทน ณ เวลาใด ๆ ก็ได้ ชั้นที่สองคือชั้นของแผนภาพโคโฮเนน (Kohonen layer) หรือชั้นของเอาท์พุตประกอบไปด้วยโหนดของนิวรอนที่เรียงตัวอยู่ในรูปแบบของแผนภาพ 2 มิติ ในแต่ละโหนด i จะเป็นค่าเวคเตอร์น้ำหนักแทนด้วย $m_i(t)$ นั้นคือ $m_i(t) \in \Re^n$ โดยที่ $\Re^n$ คือโดเมนของขนาดของ n และขนาดของเวคเตอร์น้ำหนักจะต้องมีขนาดเท่ากับอินพุตเวคเตอร์ $x(t)$



รูปที่ 2.1 แสดงโมเดลพื้นฐานของ SOM แบบสี่เหลี่ยม

ในการออกแบบโครงสร้างโมเดล SOM เราสามารถกำหนดโครงสร้างของโหนดได้ดังรูปที่ 2.2 ก. เป็นการกำหนดโครงสร้างของ SOM แบบหกเหลี่ยม ในรูปที่ 2.2 ข. เป็นการกำหนดโครงสร้างของ SOM แบบสี่เหลี่ยม ซึ่งทั้งสองจะมีการกำหนดโหนดใกล้เคียงที่ต่างกัน โดยที่โครงสร้างแบบหกเหลี่ยมจะมีโหนดใกล้เคียงเป็นรูปหกเหลี่ยมมี แต่โครงสร้างแบบสี่เหลี่ยมจะมีโหนดใกล้เคียงเป็นรูปสี่เหลี่ยม



ก. โครงสร้าง SOM แบบหกเหลี่ยม ข. โครงสร้าง SOM แบบสี่เหลี่ยม

รูปที่ 2.2 แสดงโครงสร้างของ SOM

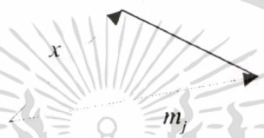
กระบวนการเรียนรู้ของ SOM เกิดขึ้นจากการปรับตัวของเวคเตอร์น้ำหนักที่มีต่ออินพุตเวคเตอร์ โดยเริ่มแรกจะทำกำหนดน้ำหนักเริ่มต้นขนาดเล็กให้กับโหนดทุกโหนด จากนั้นจะเริ่มต้นกระบวนการเรียนรู้ดังนี้

1. เลือกอินพุตเวคเตอร์แบบสุ่มเลือกจากอินพุตโดเมน
2. เปรียบเทียบอินพุตเวคเตอร์ $x(t)$ กับโหนด $m_i(t)$ ทุกโหนดเพื่อหาโหนดชนะจากโหนดทั้งหมด
3. ปรับเวคเตอร์น้ำหนักของโหนดชนะ เพื่อให้โหนดชนะเข้าใกล้อินพุตมากขึ้น

4. ปรับเวคเตอร์น้ำหนักของโหนดใกล้เคียง เพื่อให้อินพุตเวคเตอร์ถัดไปที่มีค่าใกล้เคียงมีโหนดชนะใหม่อยู่ใกล้กัน

กระบวนการเหล่านี้จะถูกทำซ้ำไปเรื่อย ๆ จนกว่าจะสอดคล้องตามเงื่อนไขหรือจนกว่าจะครบจำนวนรอบของการเรียนรู้ จากกระบวนการเรียนรู้ข้างต้นมีการคำนวณที่สำคัญอยู่ 2 ส่วนคือ

ส่วนแรกคือการคำนวณเพื่อหาโหนดชนะ(ขั้นตอนที่ 2)ในการคำนวณหาโหนดชนะอินพุตเวคเตอร์ $x(t)$ ถูกนำไปเปรียบเทียบกับโหนด $m_i(t)$ ทุกโหนดเพื่อหาโหนดชนะจากโหนดทั้งหมด ฟังก์ชั่นที่ใช้ในการเปรียบเทียบโดยทั่วไปแล้วจะใช้ฟังก์ชั่นวัดระยะทางแบบยูคลิด (Euclidean distance) ดังรูปที่ 2.3



รูปที่ 2.3 แสดงระยะทางแบบยูคลิดระหว่างเวคเตอร์ x และ m_j

การหาโหนดที่ชนะ c สามารถหาได้จากโหนดที่มีระยะห่างระหว่างอินพุตเวคเตอร์กับเวคเตอร์น้ำหนักของโหนดนั้นน้อยที่สุดดังสมการที่ 1.1

$$c : m_c(t) = \min_i \| x(t) - m_i(t) \| \tag{1.1}$$

ส่วนที่สองคือการปรับเวคเตอร์น้ำหนัก หลังจากที่ได้โหนดชนะแล้วจะต้องทำการปรับน้ำหนักเพื่อให้เข้าใกล้อินพุตมากขึ้น นอกจากการเรียนรู้ที่เกิดขึ้นที่โหนดชนะแล้ว โหนดใกล้เคียงจะเกิดการเรียนรู้ด้วย ค่าเวคเตอร์น้ำหนักของโหนดใกล้เคียงจะปรับค่าให้เข้าใกล้กับอินพุตเวคเตอร์เดียวกัน เพื่อเพิ่มโอกาสให้อินพุตใหม่ที่ใกล้เคียงกับอินพุตเดิมสามารถที่จะมีโหนดชนะใหม่ใกล้กับโหนดชนะเดิมได้ สมการในการปรับค่าน้ำหนักสามารถแสดงได้ดังสมการที่ 1.2

$$m_i(t+1) = m_i(t) + \alpha(t) \times h_{ci}(t) \times [x(t) - m_i(t)] \tag{1.2}$$

เมื่อ

- t คือรอบปัจจุบันของการเรียนรู้
- $x(t)$ คืออินพุตเวคเตอร์ปัจจุบัน
- $m_i(t)$ คือเวคเตอร์น้ำหนัก
- $\alpha(t)$ คืออัตราการเรียนรู้

โดยที่อัตราการเรียนรู้ $\alpha(t)$ จะขึ้นอยู่กับจำนวนรอบซึ่งแสดงเป็นสมการเชิงเส้นได้ดังสมการที่ 1.3

$$\alpha(t) = \alpha(0) \times \frac{T - t}{T} \tag{1.3}$$

เมื่อ

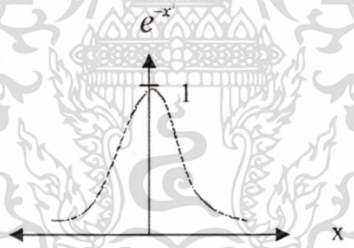
- T คือจำนวนรอบทั้งหมด
- t คือจำนวนรอบปัจจุบัน

$h_{ci}(t)$ คือฟังก์ชั่นที่ใช้ในการกำหนดน้ำหนักในการปรับค่าโหนดใกล้เคียงโดยทั่วไปแล้ว $h_{ci}(t)$ จะใช้ฟังก์ชั่นเกาส์เซียน (Gaussian) ซึ่งสามารถเขียนได้ดังสมการที่ 1.4

$$h_{ci}(t) = \exp(-\frac{\| r_c - r_i \|^2}{2\sigma^2(t)}) \tag{1.4}$$

เมื่อ

- $\|r_c - r_i\|$ คือระยะห่างของตำแหน่งของโหนด i กับโหนดชนะ c
- $\sigma(t)$ คือรัศมีของบริเวณโหนดใกล้เคียง

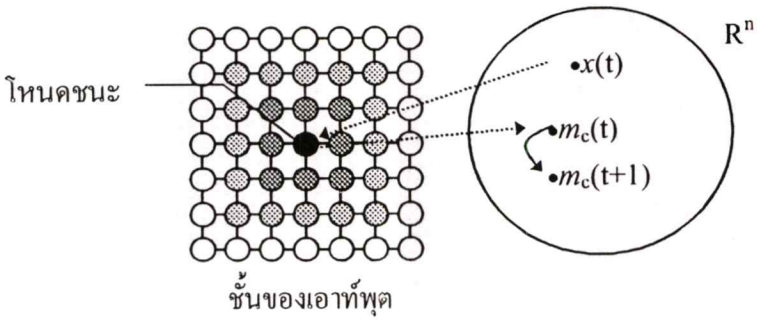


รูปที่ 2.4 แสดงกราฟของฟังก์ชั่น Gaussian ($y=e^{-x^2}$)

ในรูปที่ 2.4 ลักษณะของฟังก์ชั่นเกาส์เซียนคือ เมื่อค่า x คือค่าระยะห่างมีค่ามาก ค่าที่ส่งกลับมาจากฟังก์ชั่นจะลดลงไปเรื่อย ๆ จนเข้าใกล้ศูนย์ ซึ่งสอดคล้องกับการปรับน้ำหนักของโหนดชนะและโหนดใกล้เคียง โหนดชนะจะมีค่า x เป็นศูนย์ซึ่งจะให้ค่าเกาส์เซียนฟังก์ชั่นออกมาเป็นหนึ่งซึ่งมากที่สุด โหนดที่ใกล้กับโหนดชนะจะมีการปรับค่าเวคเตอร์น้ำหนักมากกว่าโหนดที่อยู่ไกล โดยจะมีการกำหนดรัศมีของโหนดใกล้เคียง

โดยปกติรัศมีของโหนดใกล้เคียงจะค่อย ๆ ลดลงตามจำนวนรอบในการเรียนรู้ t ดังสมการที่ 1.5

$$\sigma(t+1) = 1 + (\sigma(t) - 1) \times \frac{T - t}{T} \tag{1.5}$$

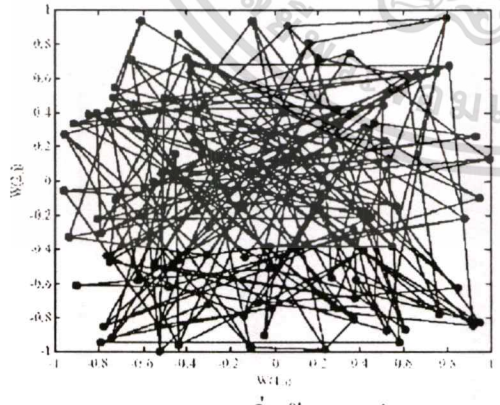


รูปที่ 2.5 แสดงโครงสร้างของ SOM ขนาด 7x7

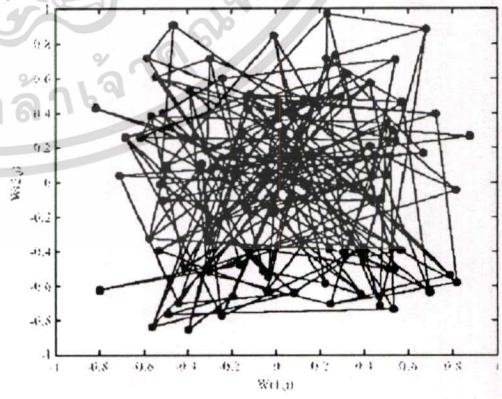
ในรูปที่ 2.5 แสดง SOM ขนาด 7x7 แบบสี่เหลี่ยมโหนดสีเข้มที่สุดคือโหนดชนะสำหรับอินพุตเวคเตอร์ $x(t)$ จากนั้นค่าเวคเตอร์น้ำหนักของโหนด $m_c(t)$ จะถูกปรับค่าให้เข้าใกล้กับอินพุตเวคเตอร์มากขึ้น หลังจากนั้นจะทำการปรับโหนดใกล้เคียงของโหนดชนะ โดยความเข้มสีของโหนดจะแสดงถึงปริมาณการปรับค่าของเวคเตอร์น้ำหนัก โหนดที่มีสีเข้มมากจะมีการปรับค่าเวคเตอร์น้ำหนักมากกว่าสีเข้มน้อย

ตัวอย่างที่ 2.1 การใช้งานแผนภาพ SOM ในการเรียนรู้อินพุตเวคเตอร์ 2 มิติ

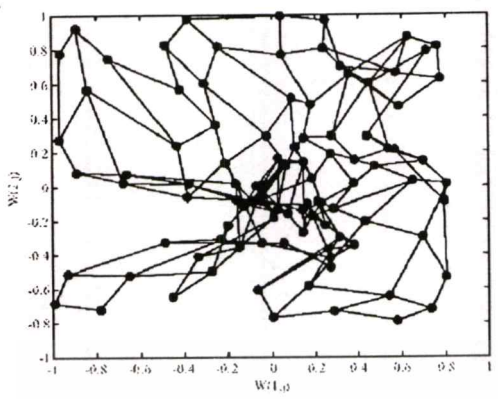
ในตัวอย่างนี้แสดงการเรียนรู้ของแผนภาพ SOM ขนาด 10x10 โดยจะทำการจัดกลุ่มอินพุตเวคเตอร์ที่มีขนาด 2 มิติ จำนวน 1000 เวคเตอร์ โดยอินพุตเวคเตอร์ได้จากการสุ่มค่าที่อยู่ในช่วงของ [-1,1] และเวคเตอร์น้ำหนักของโหนดได้จากการสุ่มอยู่ในช่วงของ [-1,1] ด้วย อัตราการเรียนรู้เริ่มต้น α=0.1 ผลของการเรียนรู้แสดงในรูปที่ 2.6



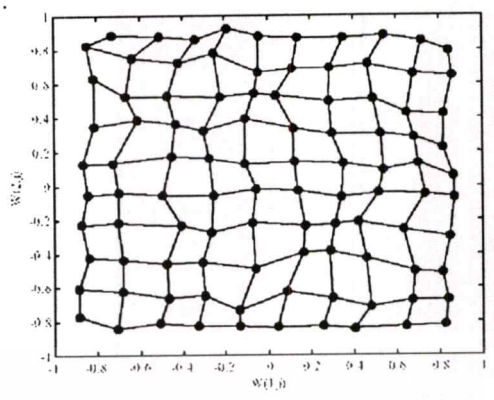
ก. แผนภาพเริ่มต้นแบบสุ่ม



ข. แผนภาพ SOM หลังจาก 100 รอบ



ค. แผนภาพ SOM หลังจาก 1000 รอบ



ง. แผนภาพ SOM หลังจาก 10000 รอบ

รูปที่ 2.6 แสดงแผนภาพ SOM ณ จำนวนรอบที่แตกต่างกัน

ในรูป 2.6 แสดงตัวอย่างแผนภาพ SOM ณ จำนวนรอบที่แตกต่างกัน จุดในรูปจะแสดงเวคเตอร์น้ำหนักของโหนด w_{1j}, w_{2j} ผลลัพธ์ที่ได้ในรูป 2.6 ง. เมื่อเสร็จสิ้นกระบวนการแผนภาพจะถูกจัดเรียงอย่างถูกต้อง โดยอินพุตเวคเตอร์ 1 ตัวก็จะตอบสนองกับโหนดเพียง 1 โหนด แต่โหนด 1 โหนดอาจจะตอบสนองกับอินพุตเวคเตอร์มากกว่า 1 ตัวก็ได้ หรือ บางโหนดอาจจะไม่ตอบสนองกับอินพุตเวคเตอร์ใด ๆ เลย

ตัวอย่างที่ 2.2 การใช้งานแผนภาพ SOM ในการเรียนรู้อินพุตเวคเตอร์ RGB

เรากำหนดแผนภาพ SOM ให้มีขนาด 9x9 เซตข้อมูลที่ใช้การเรียนรู้เป็นเวคเตอร์ RGB จำนวน 500 เวคเตอร์

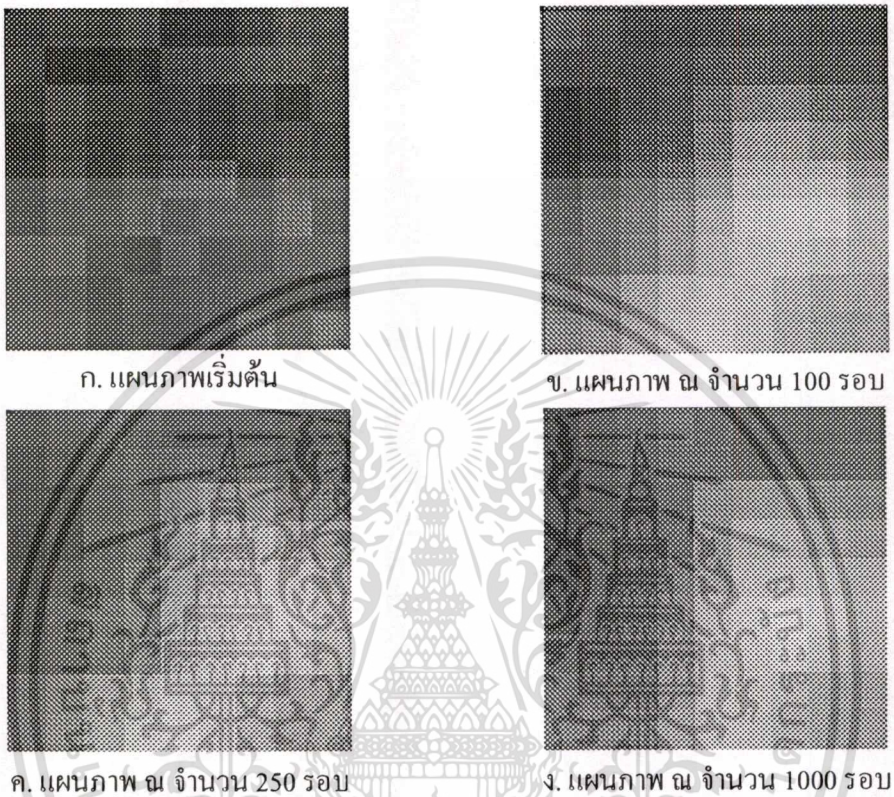
ตารางที่ 2.1 แสดงอินพุตเวกเตอร์ในรูปแบบของ RGB

R	G	B
250	235	215
165	042	042
210	105	30
255	140	0
233	150	122
...	...	...

ตารางที่ 2.1 แสดงตัวอย่างของข้อมูลสีซึ่งสามารถเขียนเป็นอินพุตเวคเตอร์ที่มีขนาด 3 มิติ ได้อยู่ในรูปของ (148R,52G,200B)

กำหนดจำนวนรอบในการเรียนรู้ T=1000 กำหนดรูปแบบของโหนดใกล้เคียงเป็นแบบสี่เหลี่ยม รัศมีของโหนดใกล้เคียง $\sigma(0)=5$ และอัตราเรียนรู้เริ่มต้น $\alpha(0)=0.2$ ในรูปที่ 2.7 ก. แสดง

แผนภาพในตอนเริ่มต้นซึ่งจะทำการสุ่มค่าเวคเตอร์น้ำหนักในที่นี้สุ่มเลือกตั้งแต่ 0-50 ทั้ง RGB รูปที่ 2.7 ข. รูปที่ 2.7 ค. รูปที่ 2.7 ง. แสดงแผนภาพ ณ จำนวนรอบที่ 100 250 และ 1000 รอบ ตามลำดับ



ก. แผนภาพเริ่มต้น ข. แผนภาพ ณ จำนวน 100 รอบ

ค. แผนภาพ ณ จำนวน 250 รอบ ง. แผนภาพ ณ จำนวน 1000 รอบ

รูปที่ 2.7 แสดงตัวอย่างแผนภาพ SOM ขนาด 9x9 ณ จำนวนรอบที่แตกต่างกัน

2.3 คุณสมบัติของ Self-Organizing Map

2.3.1 แผนภาพเรียงตัว

คุณสมบัติที่สำคัญที่ทำให้ SOM เป็นที่นิยมใช้งานกันอย่างแพร่หลายคือ คุณสมบัติในการวิเคราะห์ข้อมูลที่มีมิติสูง โดยผลลัพธ์ที่ได้จะถูกแสดงอยู่ในรูปแบบของแผนภาพ 2 มิติ เมื่อข้อมูลถูกกำหนดลงไปในโหนดต่าง ๆ ของแผนภาพ ข้อมูลที่คล้ายกันจะถูกกำหนดให้โหนดที่อยู่ใกล้เคียงกัน ด้วยเหตุนี้แผนภาพที่ได้ออกมาจะมีลักษณะของการจัดเรียงตัวกันของข้อมูล ซึ่งทำให้ผู้ใช้สามารถที่จะเข้าใจลักษณะโครงสร้างของข้อมูลได้ นอกจากนั้นการแสดงข้อมูลด้วยแผนภาพยังช่วยให้ผู้ใช้สามารถวิเคราะห์ข้อมูล มองเห็นความสัมพันธ์ของข้อมูล ซึ่งในบางครั้งไม่สามารถมองเห็นได้ด้วยการแสดงข้อมูลทั่วไปเช่น ตาราง หรือกราฟ

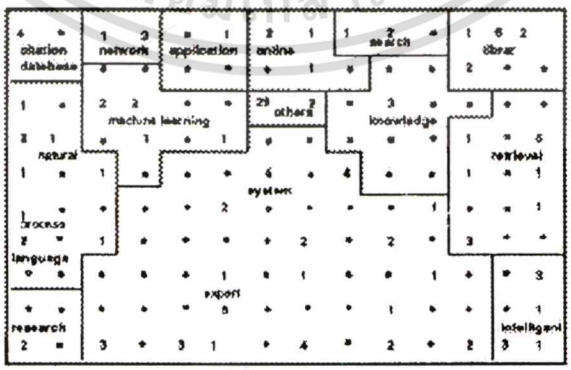
2.3.2 การจัดแบ่งกลุ่มข้อมูล

ในมุมมองของการแบ่งกลุ่ม ข้อมูลที่จัดเรียงสามารถมองเป็นการแบ่งกลุ่มข้อมูลได้ กลุ่มต่าง ๆ จะอยู่ตามโหนดต่าง ๆ ของแผนภาพ ลักษณะของการแบ่งกลุ่มข้อมูลโดยใช้แผนภาพ SOM นั้นจะคล้ายกับการแบ่งกลุ่มข้อมูลแบบ K-mean คือจะมีการกำหนดกลุ่มไว้ล่วงหน้าแล้วจัดข้อมูลให้ลงไปในกลุ่มต่าง ๆ จุดที่แตกต่างกันคือ การแบ่งกลุ่มข้อมูลแบบ K-mean ผลที่ได้จะได้กลุ่มของข้อมูลออกมาเป็นกลุ่มแบ่งแยกจากกันโดยเด็ดขาด แต่ลักษณะของการแบ่งกลุ่มโดยใช้แผนภาพ SOM นั้น จะพิจารณาถึงความสัมพันธ์ระหว่างกลุ่มด้วย โดยกลุ่มที่อยู่ใกล้กันจะมีข้อมูลที่คล้ายคลึงกัน

2.4 งานวิจัยที่ประยุกต์ใช้ SOM

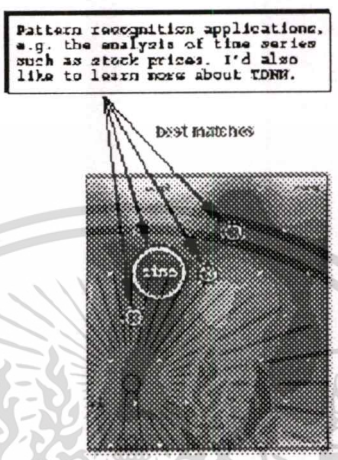
ในงานวิจัยของ Xia Lin, Dagobert Soergal, Gary Marchioninl [2] ได้แสดงให้เห็นถึงการนำเอา SOM มาประยุกต์ใช้กับระบบค้นคืนสารสนเทศ การทำงานจะประกอบไปด้วยเซตของอินพุตเวคเตอร์เป็น n มิติซึ่งเป็นอินพุตของแผนภาพโดยแผนภาพนั้นจะเป็นแผนภาพ 2 มิติ โดยที่อินพุตแต่ละตัวนั้นจะเรียกว่าเป็นคุณลักษณะ (feature) และแต่ละโหนดของกริดจะรับอินพุตจากเวคเตอร์โดยจะเรียกว่าเป็นน้ำหนัก(weight)

ระบบการสืบค้นเอกสารซึ่งระบบข้อมูลชื่อ LISA ซึ่งมีเอกสารทั้งหมดที่ทำอินเด็กซ์ 140 เอกสารอยู่ในรูปของ inverted index โดยได้กำจัดคำหยุด (stop word) คำที่มีความถี่มาก คำที่มีความถี่น้อยกว่า 3 คำ และคำที่มีรากเดียวกันเช่น librarian กับ library จะนับเป็นคำเดียวกัน เมื่อผ่านกระบวนการเรียบร้อยจะได้เซตของอินเด็กซ์เทอม เวคเตอร์ที่ใช้อินพุตกับระบบก็คือค่าเวคเตอร์ของเอกสารซึ่งมีทั้งหมด 25 คุณลักษณะตามจำนวนคีย์เทอม และมีโหนดทั้งหมด 140 โหนด (14x10) หลังจากการเรียนรู้แล้วแผนภาพที่ได้จะมีลักษณะดังรูปที่ 2.8



รูปที่ 2.8 แสดงแผนภาพ SOM จาก 140 เอกสาร โดยใช้ฐานข้อมูล LISA

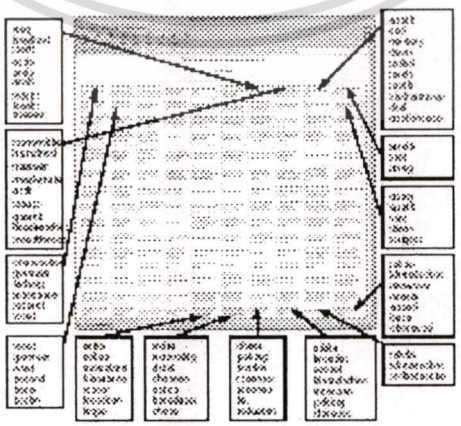
ในปี ค.ศ. 1997 Timo Honkela, Samuel Kaski, Krista Lugus และ Teuvo Kohonen ได้นำเสนอ WEBSOM [5,6] ซึ่งเป็นการประยุกต์ใช้ SOM ในการจัดกลุ่มข่าว โดยนำคำศัพท์มาเข้ารหัสเป็นแผนภาพความหมาย (Semantic Map) จากนั้นสร้างเป็นแผนภาพของเอกสาร ดังรูปที่ 2.9



รูปที่ 2.9 แสดงแผนภาพ SOM ในงานวิจัย WEBSOM

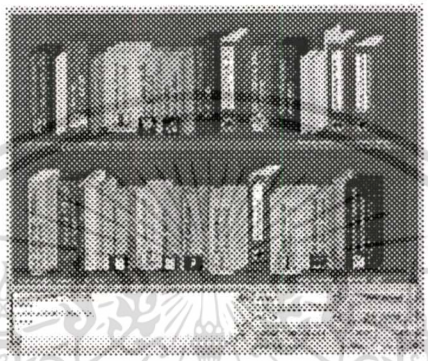
งานวิจัยนี้เป็นงานวิจัยแรก ๆ ที่นำเสนอลักษณะของ SOM ออกมาเป็นแผนภาพพร้อมสร้างอินเตอร์เฟสในการค้นหา ถือเป็นการนำเสนอการค้นคืนข้อมูลแบบใหม่

งานวิจัยของ Andreas Rauber, Dieter Merkl [11] ได้ประยุกต์ใช้งาน SOM เพื่อสร้างห้องสมุดอิเล็กทรอนิกส์(Digital Library) โดยนำเอาเอกสารมาสร้างเป็นอินเด็กซ์เทอมในรูปของ *tf* x *idf* [12,13] ซึ่งเป็นวิธีการที่ใช้ในวิทยานิพนธ์ฉบับนี้ ซึ่งจะกล่าวต่อไปในหัวข้อ 4.2 แผนภาพที่ใช้มีขนาด 10x15 ดังแสดงในรูปที่ 2.10 กระบวนการเรียนรู้จะเหมือนกับอัลกอริทึมที่ได้กล่าวในหัวข้อ 2.2



รูปที่ 2.10 แสดงการจัดกลุ่มเอกสารโดยใช้แผนภาพ SOM ขนาด 10x15

ในงานวิจัยนี้ผู้วิจัยได้สร้างอินเตอร์เฟสที่ชื่อว่า LibViewer ขึ้นมาเพื่อจำลองภาพของชั้นวางหนังสือเสมือนขึ้นมีลักษณะดังรูป 2.10 ในงานวิจัยนี้ได้แสดงให้เห็นถึงการนำแผนภาพ SOM ประยุกต์ใช้ในงานห้องสมุดเสมือนซึ่งเป็นอีกแนวคิดในการประยุกต์ใช้ SOM ในการจัดกลุ่มเอกสารเพื่อง่ายต่อการสืบค้น



รูปที่ 2.11 แสดงชั้นวางหนังสือเสมือนใน LibViewer

ในบทนี้ได้นำเสนอโครงสร้าง อัลกอริทึมการเรียนรู้ และคุณสมบัติของ SOM พร้อมทั้งแสดงตัวอย่างการใช้งาน เพื่อให้เห็นประโยชน์ของการใช้งาน SOM รวมถึงงานวิจัยที่มีผู้นิยมในการอ้างอิงถึง ในบทถัดไปจะนำเสนอปัญหาที่พบในการใช้งาน SOM พร้อมทั้งนำเสนอโมเดลใหม่

บทที่ 3

Self-Organzing Map แบบความเร็วสูง

ในบทนี้จะกล่าวถึงจุดด้อยของโมเดล SOM รวมถึงงานวิจัยต่าง ๆ ที่พยายามปรับปรุงจุดด้อยของโมเดล และนำเสนอโมเดล SOM แบบความเร็วสูง หรือ "High Speed Self-Organizing Map " หรือ HS-SOM ซึ่งเป็นแนวคิดใหม่ของงานวิจัยนี้ในการปรับปรุงโมเดลแบบดั้งเดิม รวมทั้งข้อดีและข้อเสียของโมเดลที่นำเสนอใหม่นี้

3.1 บทนำ

ในบทที่ 2 เราได้กล่าวถึงอัลกอริทึมของ SOM แบบดั่งเดิมของโคโฮเนน ซึ่งเป็นนิวรอนเน็ตเวิร์กโมเดลที่เหมาะสำหรับการวิเคราะห์ข้อมูลที่มีมิติสูง โดยนิวรอนเน็ตเวิร์กจะสามารถเรียนรู้ได้โดยการปรับค่าเวคเตอร์น้ำหนักของโหนดให้สัมพันธ์กับอินพุตเวคเตอร์ที่เข้ามา การเรียนรู้จะเกิดขึ้นที่โหนดชนะและโหนดใกล้เคียง หลังการเรียนรู้เสร็จสิ้นเราจะได้แผนภาพที่แสดงความสัมพันธ์ของข้อมูล โดยข้อมูลที่ใกล้เคียงกันจะอยู่บริเวณเดียวกัน

ปัญหาสำคัญที่พบในการใช้งาน SOM คือ เมื่อมีข้อมูลจำนวนมากจำเป็นที่จะต้องเพิ่มจำนวนโหนดตาม ซึ่งส่งผลให้มีการคำนวณเพิ่มขึ้น ดังนั้นแผนภาพจะใช้เวลาในการเรียนรู้เพิ่มขึ้นตาม เวลาที่ใช้ในการเรียนรู้ของ SOM ส่วนใหญ่ใช้ในการหาโหนดชนะ ซึ่งจะต้องทำการเปรียบเทียบระหว่างอินพุตเวคเตอร์กับเวคเตอร์น้ำหนักของทุกโหนด ในกรณีที่เป็นเวคเตอร์ของเอกสารจะมีมิติสูงมาก ยิ่งเพิ่มเวลาในการหาโหนดชนะยิ่งขึ้นไป

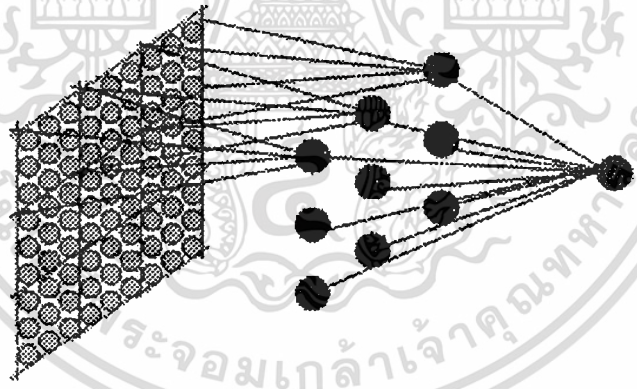
มีงานวิจัยหลายงานที่พยายามปรับปรุงโมเดล หรือพยายามคิดวิธีหาโหนดชนะแบบใหม่เพื่อลดเวลาในการเรียนรู้ลง ในงานวิจัยของ [3] ได้นำเอาอัลกอริทึม K-means ช่วยในการจัดกลุ่มข้อมูลก่อนนำไปเรียนรู้ในแผนภาพเพื่อให้แผนภาพจัดเรียงตัวได้เร็วขึ้นในจำนวนรอบที่น้อยกว่า ในงานวิจัยของ [9] ได้นำเสนอวิธีการลดการคำนวณในการหาโหนดชนะ(winning nodc) โดยแบ่งเป็นส่วนย่อย ๆ ที่ใช้ในการค้นหา ด้วยวิธีนี้สามารถลดเวลาในการคำนวณได้สูงสุด 14 เปอร์เซ็นต์

ในงานวิจัยฉบับนี้จะนำเสนอโมเดลใหม่และอัลกอริทึมในการเรียนรู้ใหม่ ซึ่งช่วยลดเวลาในการเรียนรู้ลงได้มากกว่า 20 เปอร์เซ็นต์ โดยที่ประสิทธิภาพของการเรียนรู้ยังคงเดิม โดยโมเดลใหม่นี้เราให้ชื่อว่า SOM แบบความเร็วสูง หรือ High Speed Self-Organizing Map (HS-SOM) ซึ่งมีรายละเอียดของโมเดลและอัลกอริทึมดังที่จะนำเสนอในหัวข้อถัดไป

3.2 แผนภาพ SOM แบบความเร็วสูง(High Speed Self-Organizing Map)

ในงานวิจัยนี้ผู้วิจัยจะพิจารณาปรับปรุงข้อจำกัดเรื่องระยะเวลาในการเรียนรู้ โดยนำเสนอวิธีการใหม่ช่วยลดเวลาในการคำนวณการหาโหนดชนะ ซึ่งอาศัยแนวคิดดังนี้คือ เมื่อพิจารณาความหมายของโหนดชนะ โหนดชนะเป็นโหนดหลักสำหรับข้อมูลที่มีเนื้อหาใกล้เคียงกัน เมื่อหาโหนดชนะได้แล้วจะทำการปรับน้ำหนักของโหนดชนะและโหนดใกล้เคียงให้สอดคล้องกับอินพุตนั้น นั่นคือข้อมูลที่มีเนื้อหาใกล้เคียงกันก็จะอยู่ตามโหนดรอบ ๆ โหนดชนะ เมื่อเป็นดังนี้เราสามารถที่จะรวมโหนดเป็นกลุ่มแล้วหาตัวแทนของกลุ่มเพื่อใช้เป็นตัวแทนในการพิจารณากลุ่มนั้นโดยที่เราไม่จำเป็นต้องพิจารณาโหนดในกลุ่มอื่น ๆ ที่ไม่มีเนื้อหาใกล้เคียงกัน ในที่นี้ตัวแทนของกลุ่มโหนด เราจะใช้จุดศูนย์ถ่วงน้ำหนัก(centroid) มาพิจารณาเป็นตัวแทนกลุ่ม ตัวแทนกลุ่มไหนมีค่าใกล้เคียงกับอินพุตมากที่สุด แสดงว่าโหนดบริเวณนั้นเนื้อหาสอดคล้องกับอินพุต หลังจากที่เราหาตัวแทนกลุ่มที่ใกล้เคียงที่สุดแล้ว จากนั้นเข้าไปหาโหนดชนะของกลุ่มจากกลุ่มนั้นอีกที

จากหลักการที่ได้กล่าวมา เราได้นำมาสร้างเป็นแผนภาพ SOM อีกชั้นขึ้นคั่นกลางระหว่างชั้นของอินพุตและชั้นของแผนภาพ SOM เดิม โดยที่โหนดแต่ละโหนดในแผนภาพ SOM ใหม่คือตัวแทนของกลุ่มของโหนดในแผนภาพ SOM เดิม แสดงให้เห็นดังรูปที่ 3.1



ชั้นที่ 1(แผนภาพเดิม) ชั้นที่ 2 (แผนภาพใหม่) อินพุต

รูปที่ 3.1 แสดงตัวอย่างโมเดลของ HS-SOM

จากรูปที่ 3.1 แสดงการรวมกลุ่มโหนดของแผนภาพ SOM ชั้นที่ 1 ซึ่งเป็นชั้นเอาท์พุต โดยแต่ละกลุ่มจะประกอบด้วยโหนดสมาชิก 9 โหนด ขนาด 3x3 โหนด และเวคเตอร์น้ำหนักของโหนดในแผนภาพ SOM ชั้นที่ 2 สามารถหาได้จากค่าจุดศูนย์กลางมวลของกลุ่มโหนดในแผนภาพชั้นที่ 1 ซึ่งมีสมการดังสมการที่ 3.1

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} \tag{3.1}$$

เมื่อ

x_i คือส่วนประกอบของเวคเตอร์ n คือจำนวนโหนดทั้งหมดในกลุ่ม

3.3 อัลกอริทึมในการสร้างและการเรียนรู้ของโมเดล HS-SOM

อัลกอริทึมสร้างแผนภาพ HS-SOM ใหม่สามารถแสดงได้ดังนี้

```
กำหนดจำนวนโหนด n และขนาดของแผนภาพ SOM ในชั้นที่ 1;
กำหนดรูปแบบรวมทั้งจำนวนสมาชิกในการรวมกลุ่มโหนด (ให้ M แทนจำนวนสมาชิกโหนดในกลุ่ม);
กำหนดค่าเริ่มต้นของเวคเตอร์น้ำหนักในแต่ละโหนด;
Max_Level= จำนวนชั้นทั้งหมดของ HS-SOM;
For i=2 to Max_Level
        M=จำนวนโหนดในชั้นปัจจุบัน;
        For j=1 to M
                หาเวคเตอร์น้ำหนักโดยการคำนวณหาค่าจุดศูนย์ถ่วงของกลุ่ม;
End-For
```

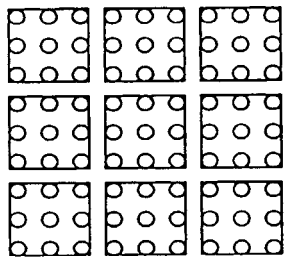
รูปที่ 3.2 แสดงอัลกอริทึมการสร้างแผนภาพ HS-SOM

รายละเอียดของอัลกอริทึมการสร้างแผนภาพ HS-SOM

```
1. กำหนดจำนวนโหนด n และขนาดของแผนภาพ SOM ในชั้นที่ 1;
2. กำหนดรูปแบบรวมทั้งจำนวนสมาชิกในการรวมกลุ่มโหนด (ให้ M แทนจำนวนสมาชิกโหนดในกลุ่ม);
3. กำหนดค่าเริ่มต้นของเวคเตอร์น้ำหนักในแต่ละโหนด
```

รูปที่ 3.3 แสดงอัลกอริทึมการสร้างแผนภาพ HS-SOM ในส่วนของการกำหนดค่าเริ่มต้น

1. เริ่มต้นด้วยการออกแบบจำนานโหนด รวมถึงรูปแบบของแผนภาพ SOM
2. กำหนดจำนวนชั้นและรูปแบบในการรวมกลุ่ม โดยสามารถกำหนดได้ทั้งจำนวนโหนดในกลุ่ม และรูปแบบของกลุ่ม ดังรูป 3.4



รูปที่ 3.4 แสดงตัวอย่างการออกแบบการรวมกลุ่มของโหนดในแผนภาพชั้นที่ 1

จากตัวอย่างรูปที่ 3.3 เป็นแผนภาพ HS-SOM 2 ชั้น โดยแผนภาพชั้นที่หนึ่งมีจำนวนโหนดทั้งหมดเป็น 9x9 โดยในรูปที่ 3.3 ก. ออกแบบให้มีชั้นของแผนภาพทั้งหมด 2 ชั้น โดยแผนภาพชั้นที่ 2 มีขนาด 4x4 (i=j=3) การกำหนดกลุ่มโหนดจะมีการซ้อนทับกัน ในรูปที่ 3.3 ข. ออกแบบให้แผนภาพชั้นที่ 2 มีขนาด 3x3 โดยไม่มีการซ้อนทับกันของกลุ่มโหนด

ในกรณีที่แผนภาพมีขนาดใหญ่มาก เราสามารถที่จะสร้างเป็นแผนภาพหลายระดับได้เพื่อลดเวลาการคำนวณลงอีกได้

3. กำหนดค่าเริ่มต้นให้กับเวคเตอร์น้ำหนักในแผนภาพชั้นที่ 1
4. หาค่าเวคเตอร์น้ำหนักในแผนภาพชั้นอื่น ๆ โดยหาจากค่าศูนย์กลางมวลของโหนดแผนภาพลำดับต่ำกว่าดังรูปที่ 3.5 ตามสมการที่ (1) โดยที่ Max_Level คือจำนวนชั้นของแผนภาพทั้งหมด M คือ จำนวนโหนดทั้งหมดในชั้น ซึ่งจะเปลี่ยนตามชั้นปัจจุบัน

```
For i=2 to Max_Level
        For j=1 to M
                หาเวคเตอร์น้ำหนักโดยการคำนวณหาค่าจุดศูนย์ถ่วงของกลุ่ม;
        M=จำนวนโหนดในชั้นปัจจุบัน;
End-For
```

รูปที่ 3.5 แสดงอัลกอริทึมการหาเวคเตอร์น้ำหนักของโหนดในชั้นอื่น ๆ

อัลกอริทึมในการเรียนรู้ของ HS-SOM สามารถแสดงได้ดังนี้

หลังจากทำการกำหนดโครงสร้างของโมเดลได้ จากนั้นจะเข้าสู่กระบวนการการเรียนรู้ของ HS-SOM ซึ่งจะแตกต่างจากกระบวนการเรียนรู้ของ SOM โดยอัลกอริทึมการเรียนรู้ของ HS-SOM สามารถแสดงได้ดังนี้

```
Max_Epoch = จำนวนรอบทั้งหมดในการเรียนรู้ ;
Num_Input = จำนวนอินพุตเวคเตอร์ทั้งหมด;
For x=0 to Max_Epoch do
        For y=1 to Num_Input do
                Level=Max_Level;
                While Level >= 1 do
                        If Level = Max_Level
                                คำนวณหาโหนดที่ใกล้เคียงกับอินพุตเวคเตอร์มากที่สุดโดยใช้การวัด
                                ระยะทางแบบยูคลิด;
                        Else
                                คำนวณหาโหนดที่ใกล้เคียงกับอินพุตเวคเตอร์มากที่สุดโดยใช้การวัด
                                ระยะทางแบบยูคลิด เฉพาะโหนดที่ถูกแทนจากโหนดชนะในชั้นก่อนหน้า
                                ;
                        Level=Level-1;
                End-while.
                ปรับโหนดใกล้เคียงในแผนภาพ SOM ในชั้นที่ 1;
                Level=2;
                While Level <=Max_level do
                        ปรับโหนดในชั้นโดยหาจุดศูนย์ถ่วงใหม่เฉพาะโหนดที่มีสมาชิกเปลี่ยนแปลง;
                        Level=Level+1;
                End-while.
        End-for.
End-for.
```

รูปที่ 3.6 แสดงอัลกอริทึมในการเรียนรู้ของ HS-SOM

รายละเอียดอัลกอริทึมในการเรียนรู้ของ HS-SOM

1. เริ่มต้นกำหนดจำนวนรอบในการเรียนรู้ Max_Epoch จากนั้นกำหนดจำนวนอินพุตเวคเตอร์ที่ใช้ในการเรียนรู้ของโมเดล Num_Input
ในกระบวนการเรียนรู้แต่ละรอบจะเรียนรู้ด้วยเซตของอินพุตเวคเตอร์ทั้งหมดดังรูปที่ 3.7

```
Max_Epoch = จำนวนรอบทั้งหมดในการเรียนรู้ ;
Num_Input = จำนวนอินพุตเวคเตอร์ทั้งหมด;
For x=0 to Max_Epoch do
        For y=1 to Num_Input do
```

รูปที่ 3.7 แสดงอัลกอริทึมในส่วนการกำหนดจำนวนรอบการเรียนรู้และจำนวนอินพุตเวคเตอร์

2. กระบวนการหาโหนดชนะจะเริ่มจากการหาโหนดชนะในแผนภาพชั้นบนสุดก่อน โดยทำการเปรียบเทียบกับโหนดทุกโหนดในแผนภาพชั้นบนสุด เมื่อได้โหนดชนะชั้นบนสุดมาแล้ว ในชั้นถัดไปจะหาโหนดชนะเฉพาะกลุ่มที่โหนดชนะชั้นก่อนหน้าเป็นตัวแทนอยู่จนกระทั่งถึงชั้นที่ 1 ดังอัลกอริทึมรูป 3.8

```
While Level >= 1 do
    If Level = Max_Level
        คำนวณหาโหนดที่ใกล้เคียงกับอินพุตเวคเตอร์มากที่สุดโดยใช้การวัด
        ระยะทางแบบยูคลิด;
    Else
        คำนวณหาโหนดที่ใกล้เคียงกับอินพุตเวคเตอร์มากที่สุดโดยใช้การวัด
        ระยะทางแบบยูคลิด เฉพาะโหนดที่ถูกแทนจากโหนดชนะในชั้นก่อนหน้า
        ;
    Level=Level-1;
End-while.
```

รูปที่ 3.8 แสดงอัลกอริทึมการหาโหนดชนะของโมเดล HS-SOM

3. เมื่อพบโหนดชนะในชั้นที่ 1 แล้วทำการปรับโหนดใกล้เคียงเหมือนโมเดล SOM หลังจากนั้นจะทำการปรับโหนดในชั้นสูงกว่า โดยทำการหาค่าศูนย์ถ่วงใหม่เฉพาะกลุ่มที่มีการเปลี่ยนแปลงเวคเตอร์น้ำหนัก ดังรูปที่ 3.9

```
ปรับโหนดใกล้เคียงในแผนภาพ SOM ในชั้นที่ 1;
Level=2;
While Level <=Max_level do
        ปรับโหนดในชั้นโดยหาจุดศูนย์ถ่วงใหม่เฉพาะโหนดที่มีสมาชิกเปลี่ยนแปลง;
        Level=Level+1;
End-while.
```

รูปที่ 3.9 แสดงอัลกอริทึมการปรับโหนดใกล้เคียงและการปรับโหนดในชั้นต่าง ๆ

หลังจากปรับค่าเวคเตอร์น้ำหนักในแต่ละชั้นครบก็จะเสร็จสิ้นกระบวนการเรียนรู้ 1 รอบ สำหรับอินพุตเวคเตอร์นั้น ในทางทฤษฎีเราสามารถแทรกแผนภาพใหม่ระหว่างชั้นของแผนภาพเอาท์พุตกับชั้นของอินพุต ในกรณีที่มีข้อมูลเป็นจำนวนมากและต้องใช้โหนดจำนวนมากในการจัดกลุ่มข้อมูล

ในบทนี้ได้นำเสนอโมเดล SOM แบบความเร็วสูง หรือ High Speed Self-Organizing Map(HS-SOM) ซึ่งเป็นโมเดลที่พัฒนาเพื่อเพิ่มความเร็วในการเรียนรู้ สำหรับการวัดประสิทธิภาพของโมเดล HS-SOM เปรียบเทียบกับโมเดล SOM แบบเดิมนั้น จะนำเสนอในบทที่ 5

ในบทถัดไปจะเป็นการนำเสนอความหมายในการจัดกลุ่มข้อมูลและอัลกอริทึมที่นิยมใช้ รวมถึงการนำ SOM ไปใช้ในการจัดกลุ่มข้อมูล และการวัดประสิทธิภาพของการจัดกลุ่มข้อมูล เนื่องจากในงานวิจัยนี้จะทำการทดสอบและวัดประสิทธิภาพในการนำเอาโมเดล SOM แบบดั่งเดิมจัดกลุ่มเอกสารเปรียบเทียบกับโมเดล HS-SOM ที่นำเสนอ

บทที่ 4

การจัดกลุ่มข้อมูลและเอกสาร

ในบทนี้จะกล่าวถึงอัลกอริทึมที่ใช้ในการกลุ่มจัดข้อมูล(Data clustering) รวมถึงใช้ในการจัดกลุ่มเอกสาร โดยจะเริ่มจากการดึงคุณลักษณะของเอกสารเพื่อใช้ในการจัดกลุ่ม หลังจากนั้นจะกล่าวถึงอัลกอริทึมแบบลำดับขั้น (Hierarchical Clustering) และแบบ K-mean รวมถึงการวัดประสิทธิภาพของการจัดกลุ่มเอกสาร

4.1 บทนำ

การจัดกลุ่มข้อมูลคือ การแบ่งข้อมูลออกเป็นกลุ่มของข้อมูลที่เหมือนกัน [14] ซึ่งแต่ละกลุ่มจะเรียกว่า คลัสเตอร์(Cluster) ข้อมูลที่อยู่ในคลัสเตอร์เดียวกันจะมีความคล้ายคลึงกัน ข้อมูลที่อยู่ต่างคลัสเตอร์จะมีความแตกต่างกัน การจัดกลุ่มข้อมูลเป็นประเภทหนึ่งของการจำแนกประเภท (Classification) ซึ่งเรียกว่าเป็นการจำแนกประเภทแบบไม่มีผู้สอน(Unsupervised Classificaition) ซึ่งไม่มีข้อมูลใด ๆ ของกลุ่มที่ถูกจัดไว้ก่อนแล้ว ต่างจากการจำแนกประเภทแบบมีผู้สอน (Supervised Classification) ซึ่งกลุ่มของข้อมูลจะถูกกำหนดไว้ก่อนล่วงหน้า จากนั้นข้อมูลจะถูกจัดลงในกลุ่มที่ถูกกำหนดไว้แล้ว เช่น ในงานของการรู้จำตัวอักษร หรือ การรู้จำใบหน้าคน

การจัดกลุ่มข้อมูลถูกนำมาประยุกต์ใช้ในงานการจัดกลุ่มเอกสารกันอย่างกว้างขวางในเรื่องของการทำเหมืองเอกสาร(Text mining) และ การค้นคืนสารสนเทศ(Information retrieval) [15] โดยเริ่มแรกนั้นการจัดกลุ่มเอกสารถูกพัฒนาขึ้นเพื่อเพิ่มค่าความแม่นยำ(Precision) หรือค่าความระลึก(Recall) และเพิ่มประสิทธิภาพในการหาเอกสารที่มีเนื้อหาใกล้เคียงกัน แต่ในปัจจุบันนี้การจัดกลุ่มเอกสารถูกนำมาขยายขอบเขตเพื่อใช้ในการทำเป็นเครื่องมือในสำรวจ(Browse)เอกสารหรือเป็นเครื่องมือในการการค้นหา (Search engine) เอกสารโดยรับคิวรีจากผู้ใช้เข้ามาจากนั้นแสดงรายการของเอกสารในกลุ่มที่ใกล้เคียงกับคิวรี

ขั้นตอนแรกของการจัดกลุ่มเอกสารคือการดึงเอาคุณลักษณะของเอกสารออกมา ในงานวิจัยนี้ใช้เวกเตอร์โมเดลแทนเอกสารซึ่งเป็นโมเดลที่ได้รับความนิยมในระบบค้นคืนสารสนเทศ

4.2 การดึงคุณลักษณะของข้อมูล

ขั้นตอนแรกของการจัดกลุ่มข้อมูลคือ การดึงเอาคุณลักษณะของข้อมูลมาทำการวิเคราะห์ ซึ่งจะต้องดึงคุณลักษณะให้สอดคล้องกับการจัดกลุ่มข้อมูลด้วย เช่น ในการจัดกลุ่มรูปภาพถ้าเราใช้สีเป็นคุณลักษณะในการจัดกลุ่ม เราก็จะได้กลุ่มของรูปภาพที่มีสีคล้ายกัน แต่ในกลุ่มเดียวกันนั้น

อาจจะได้ภาพของท้องฟ้า กับ เสื้อสีฟ้าอยู่ในกลุ่มเดียวกัน ความผิดพลาดที่เกิดขึ้นไม่ได้เกิดขึ้นจากอัลกอริทึม แต่เกิดขึ้นจากการเลือกใช้คุณลักษณะที่ไม่ถูกต้องหรือไม่สอดคล้อง

ในการดึงคุณลักษณะจากเอกสารเราจะใช้คำศัพท์ของเอกสารเป็นคุณลักษณะ โดยจะสร้างเป็นเวคเตอร์ของเอกสารซึ่งมีคำศัพท์เป็นส่วนประกอบของเวคเตอร์ ในการคำนวณน้ำหนักของแต่ละส่วนประกอบสำหรับแต่ละเอกสารจะใช้วิธี *tf-idf* (term frequency-inverse document frequency) [12,13] ซึ่งเป็นวิธีที่นิยมใช้ในระบบค้นคืนสารสารสนเทศ ซึ่งจะกล่าวถึงโดยละเอียดในบทที่ 5

4.3 การวัดความคล้ายของข้อมูล

เมื่อได้คุณลักษณะของข้อมูลแล้ว ขั้นตอนที่สำคัญอีกประการหนึ่งคือ การวัดความคล้ายของข้อมูล ในงานวิจัยนี้ใช้วิธีวัดระยะห่าง (Distance measurement)ซึ่งนิยมใช้เป็นอย่างมากเรียกว่าการวัดระยะห่างแบบยูคลิด(Euclidean distance) ซึ่งสามารถแสดงเป็นสมการได้ดังนี้

$$Sim(x_i, x_j) = \sqrt{\sum_{k=1}^{n} (x_{i,k} - x_{j,k})^2} \tag{4.1}$$

โดยที่ $Sim(x_i, x_j)$ หมายถึงความคล้ายกัน(Similarity)ระหว่าง x_i, x_j

x_i, x_j คือเวคเตอร์ข้อมูล

$x_{i,k}, x_{j,k}$ คือ ส่วนประกอบแต่ละส่วนของเวคเตอร์

วิธีวัดความคล้ายอื่น ๆ ที่นิยมใช้กันเช่นการวัดมุมโคไซน์(Cosine measurement) เป็นการวัดมุมระหว่างสองเวคเตอร์สามารถแสดงเป็นสมการได้ดังนี้

$$Sim(x_i, x_j) = \frac{\vec{x_i} \bullet \vec{x_j}}{|\vec{x_i}| \times |\vec{x_j}|} = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{i=1}^{t} w_{i,q}^2}} \tag{4.2}$$

โดยที่ x_i, x_j คือเวคเตอร์ข้อมูล

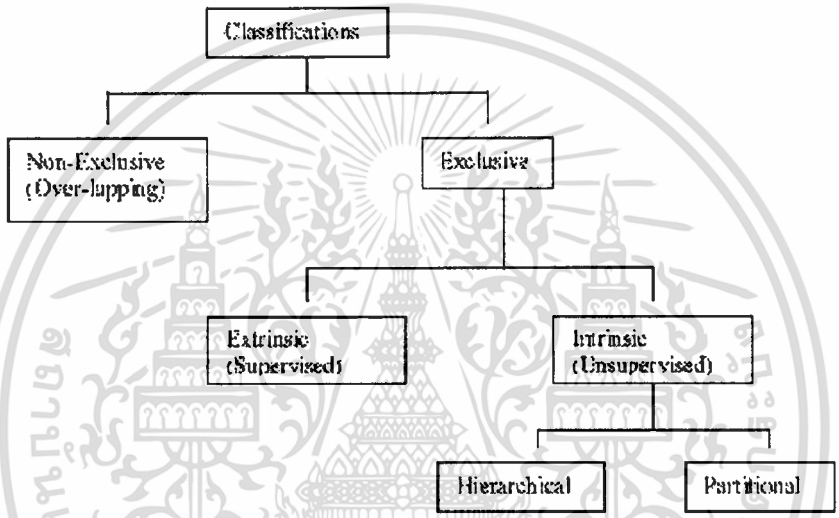
$x_{i,k}, x_{j,k}$ คือ ส่วนประกอบแต่ละส่วนของเวคเตอร์

นอกจากนี้ยังมีวิธีการวัดแบบอื่นเช่น Jaccard coefficient หรือ Correlation coefficient หรือ Probabilistic similarity coefficient ซึ่งสามารถหารายละเอียดเพิ่มเติมได้จาก [12] สำหรับงานวิจัยนี้เราเลือกใช้การวัดระยะทางแบบยูคลิดซึ่งนิยมใช้ในการจัดกลุ่มเอกสารแบบ SOM

4.4 การจัดกลุ่มข้อมูล

การจัดกลุ่มข้อมูลเป็นประเภทหนึ่งของการจำแนกประเภท ซึ่งแบ่งข้อมูลออกเป็นกลุ่ม ๆ เทคนิคที่ใช้ในการจัดกลุ่มข้อมูลมีหลายแบบ ในหัวข้อนี้จะกล่าวถึงเทคนิคที่นิยมใช้ในการจัดกลุ่มข้อมูล รวมถึงการจัดกลุ่มข้อมูลโดยใช้ SOM

ใน [14,15] ได้กล่าวถึงความสัมพันธ์ของการจำแนกประเภทกับการจัดกลุ่มไว้ โดยสามารถแสดงเป็นโครงสร้างได้ดังรูปที่ 4.1



รูปที่ 4.1 แสดงโครงสร้างประเภทของการจำแนกข้อมูล

1. Exclusive VS. nonexclusive

การจำแนกประเภทข้อมูลแบบ execlusive คือการแบ่งเซตของข้อมูลออกเป็นกลุ่ม โดยที่ข้อมูลแต่ละตัวจะสามารถอยู่ได้เพียงกลุ่มเดียวเท่านั้น ส่วนการจำแนกประเภทข้อมูลแบบ nonexeclusive ข้อมูลแต่ละตัวจะสามารถปรากฏได้ในกลุ่มหลายกลุ่ม เช่น การแบ่งกลุ่มคนโดยใช้เพศเป็นการแบ่งแบบ execlusive แต่การแบ่งกลุ่มคนตามความสามารถเฉพาะตัวจะเป็นการแบ่งแบบ nonexeclusive เนื่องจากแต่ละคนอาจจะมีความสามารถได้มากกว่า 1 อย่าง ในปัจจุบันมีสาขาด้าน fuzzy clustering ซึ่งก็จัดเป็น nonexeclusive ประเภทหนึ่ง

2. Intrinsic VS. extrinsic

การจำแนกประเภทข้อมูลแบบ Intrinsic จะใช้เพียงเมตริกซ์ของความคล้ายเป็นตัวจำแนกประเภทข้อมูล กล่าวคือ เราสามารถเรียกการจำแนกประเภทข้อมูลแบบ Intrinsic ได้ว่าเป็นการจำแนกประเภทข้อมูลแบบไม่มีผู้สอน เนื่องจากไม่มีการกำหนดการจัดกลุ่มใด ๆ เริ่มต้น ต่างจากการจำแนกประเภทข้อมูลแบบ extrinsic ซึ่งจะมีการกำหนดกลุ่มของข้อมูล

ก่อน เช่นในการสอนการรู้จำตัวอักษร เราจะต้องบอกระบบก่อนว่าจะทำการสอนตัวหนังสือใดจากนั้นจึงทำการสอนตัวหนังสือนั้น ๆ

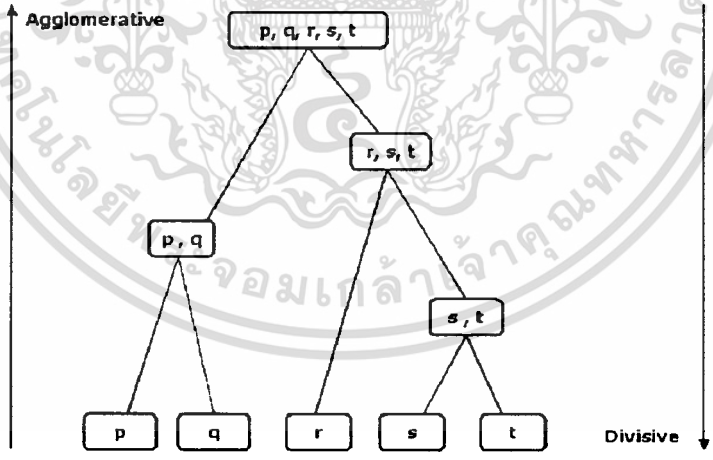
3. Hierarchical VS. partitional

ในการจำแนกประเภทข้อมูลแบบ intrinsic นั้นแบ่งออกเป็นสองประเภทหลักคือ แบบ hierarchical ข้อมูลที่ถูกแบ่งจะเป็นลำดับชั้นเชื่อมโยงกัน แต่แบบ partitional ข้อมูลจะถูกแบ่งเป็นกลุ่ม ๆ แยกจากกัน ซึ่งจะยกตัวอย่างอัลกอริทึมของทั้งสองแบบในหัวข้อถัดไป

เราจะใช้คำว่าการจัดกลุ่ม(clustering) สำหรับการจำแนกประเภทแบบ intrinsic hierarchical และแบบ intrinsic partitional เท่านั้น ส่วน extrinsic เราจะใช้คำว่าการจำแนกประเภท(classification) แทน

4.4.1 การจัดกลุ่มข้อมูลแบบ hierarchical

การจัดกลุ่มแบบ hierarchical จะได้เป็นโครงสร้างแบบลำดับชั้น โดยที่ชั้นบนสุดจะมองเห็นเป็นกลุ่มเดียวกันหมด และชั้นล่างสุดข้อมูลแต่ละอันถือว่าเป็นหนึ่งกลุ่ม ในชั้นตรงกลางเกิดจากการรวมกันของสองกลุ่มในชั้นต่ำกว่า หรือจะมองว่าเกิดจากการแตกออกเป็นสองกลุ่มของชั้นสูงกว่าก็ได้ แผนภาพที่นิยมใช้แสดงการจัดกลุ่มแบบ hierarchical เรียกว่าแผนภาพเดนโดแกรม (dendogram) ซึ่งมีลักษณะคล้ายกับโครงสร้างแบบต้นไม้ดังรูปที่ 4.2



รูปที่ 4.2 แสดงแผนภาพเดนโดแกรมของการจัดกลุ่มแบบ hierarchical

การจัดกลุ่มแบบ hierarchical สามารถแบ่งได้เป็นสองแบบคือ แบบ agglomerative ซึ่งเริ่มจากชั้นต่ำสุดของแผนภาพเดนโดแกรม โดยคิดว่าแต่ละข้อมูลคือแต่ละกลุ่ม จากนั้นรวมสองกลุ่มที่มีความคล้ายกันที่สุดเข้าด้วยกัน ต่างจากแบบ division ซึ่งจะเริ่มจากชั้นบนสุดของแผนภาพเดนโดแกรม โดยคิดว่าข้อมูลทั้งหมดคือกลุ่มเดียวกันหมด จากนั้นแตกกลุ่มออกไปจนกระทั่งเหลือหนึ่งกลุ่มหนึ่งข้อมูล

เทคนิคของ agglomerative จะได้รับความนิยมมากกว่า เนื่องจากสามารถทำได้สะดวกกว่าวิธีของ division เริ่มต้นจากการกำหนดฟังก์ชั่นที่ใช้ในการวัดความคล้ายซึ่งนิยมใช้ฟังก์ชั่นการวัดระยะห่างแบบยูคลิด เป็นตัววัดความคล้าย อัลกอริทึมของ agglomerative hierarchical สามารถแสดงได้ดังนี้

อัลกอริทึมของ agglomerative hierarchical

1. สร้างเมตริกซ์ความต่าง(dissimilarity matrix) NxN ของเอกสารทั้งหมด (เริ่มต้นแต่ละกลุ่มจะมีสมาชิกเป็นหนึ่งเอกสาร)
2. รวมสองกลุ่มที่เหมือนกันที่สุด(ในกรณีที่ใช้ฟังก์ชั่นยูคลิดจะคิดสองกลุ่มที่มีค่าระยะห่างน้อยที่สุด)
3. ทำการปรับปรุงเมตริกซ์ความไม่เหมือน โดยคิดรวมสองกลุ่มก่อนหน้าที่เหมือนกันมากที่สุดเป็นกลุ่มเดียวกัน
4. ทำซ้ำกระบวนการ 2 และ 3 จนกระทั่งเหลือกลุ่มเดียว

ตัวอย่างที่ 4.1 การจัดกลุ่มแบบ agglomerative hierarchical

ให้ A, B, C, D, E เป็นเอกสารซึ่งมีเมตริกซ์ความต่างดังนี้

ตารางที่ 4.1 แสดงเมตริกซ์ความต่าง

	A	B	C	D	E
A	-	9	3	6	11
B	9	-	7	5	10
C	3	7	-	9	2
D	6	5	9	-	8
E	11	10	2	8	-

กลุ่มที่มีความเหมือนกันมากที่สุดคือ C-E = 2 ดังนั้นทำการรวมกลุ่มเป็น CE จะได้ตารางความเหมือนใหม่ดังนี้

ตารางที่ 4.2 แสดงเมตริกซ์ความไม่เหมือนหลังจากรวมกลุ่ม C และ กลุ่ม E

	CE	A	B	D
CE	-	11	10	9
A	11	-	9	6
B	10	9	-	5
D	9	6	5	-

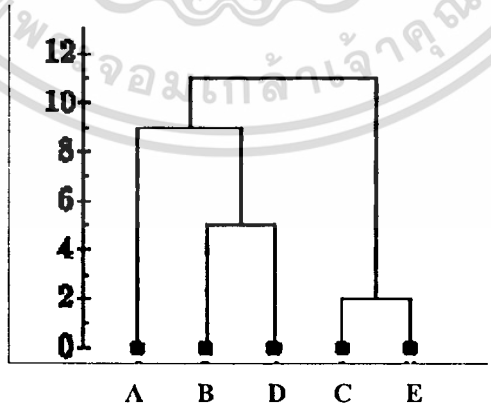
ในการคำนวณความเหมือนระหว่าง CE และ A จะสามารถหาได้จาก

ในกรณีถ้าเป็นแบบ single link เราจะเลือกค่าน้อยที่สุดระหว่าง Sim(A,C) และ Sim(A,E) ซึ่งมีค่าดังนี้ Sim(A,C)=3 และ Sim(A,E)=11 ดังนั้นถ้าเป็นแบบ single link ค่า Sim(CE,A) = 3

ในกรณีถ้าเป็นแบบ complete link เราจะเลือกค่าที่มากที่สุดระหว่าง Sim(A,C) และ Sim(A,E) ซึ่งมีค่าดังนี้ Sim(A,C)=3 และ Sim(A,E)=11 คือ Sim(CE,A)=11 ดังนั้นถ้าเป็นแบบ complete link ค่า Sim(CE,A) = 11

ในกรณีถ้าเป็นแบบ group average เราจะใช้ค่าเฉลี่ยระหว่าง Sim(A,C) และ Sim(A,E) ซึ่งมีค่าดังนี้ Sim(A,C)=3 และ Sim(A,E)=11 คือ Sim(CE,A)= 7 ดังนั้นถ้าเป็นแบบ group average ค่า Sim(CE,A) = 7

จากตัวอย่างตารางที่ 4.2 เราใช้การคำนวณแบบ complete link หลังจากนั้นทำการคำนวณความต่างของกลุ่มใหม่กับกลุ่มเก่าทุกกลุ่ม ทำต่อไปเรื่อย ๆ จนกระทั่งเหลือกลุ่มเดียวจะได้ดังรูปที่ 4.3



รูปที่ 4.3 แสดงแผนภาพเดนโดแกรมของตัวอย่างที่ 4.1แกนx แสดงลำดับก่อนหลังการรวมตัว

4.4.2 การจัดกลุ่มข้อมูลแบบ partition

ในทางตรงกันข้ามการจัดกลุ่มแบบ partition จะไม่มีการสร้างเป็นโครงสร้างต้นไม้ โดยกลุ่มแต่ละกลุ่มจะแยกจากกันและมีแค่ระดับเดียว อัลกอริทึมที่ใช้ในการจัดกลุ่มข้อมูลแบบ partition ที่ได้รับความนิยมมากที่สุดคือ K-means

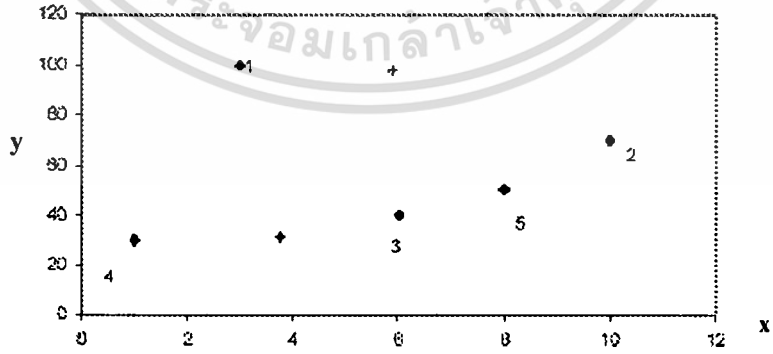
อัลกอริทึมของ K-means เริ่มต้นด้วยการกำหนดจำนวนกลุ่มที่ต้องการและค่าเริ่มต้นของแต่ละกลุ่ม K กลุ่ม จากนั้นกำหนดข้อมูลแต่ละอันลงในกลุ่ม โดยทำการวัดว่าข้อมูลนั้นใกล้กับกลุ่มไหนมากที่สุด ฟังก์ชั่นที่นิยมใช้คือ ฟังก์ชั่นการวัดระยะแบบยูคลิด อัลกอริทึมโดยรวมสามารถแสดงได้ดังนี้

อัลกอริทึมของการจัดกลุ่มแบบ K-means

1. หาสมาชิกกลุ่มของแต่ละกลุ่ม โดยคำนวณหาระยะห่างระหว่างข้อมูลทุกตัวกับกลุ่มทุกกลุ่ม ข้อมูลที่มีระยะห่างจากกลุ่มนั้นน้อยที่สุดก็จะถือว่าเป็นสมาชิกของกลุ่มนั้น
2. คำนวณหาตัวแทนของกลุ่มใหม่ โดยคำนวณจากค่าจุดศูนย์กลางมวลของสมาชิกในกลุ่มนั้น
3. ทำซ้ำกระบวนการ 1 และ 2 ใหม่ จนกระทั่งสมาชิกในกลุ่มไม่เปลี่ยนแปลง หรือค่าของตัวแทนกลุ่มไม่เปลี่ยนแปลง

ตัวอย่างที่ 4.2 การจัดกลุ่มแบบ K-means

สมมุติมีข้อมูลดังนี้ (3,100), (10,70), (6,40), (1,30), (8,50) กำหนดให้ K=2 นั้นคือเราจะทำการแบ่งกลุ่มข้อมูลทั้งหมดเป็น 2 กลุ่ม ค่าเริ่มต้นของทั้ง 2 กลุ่มคือ (4,30) และ (6,100) ดังรูปที่ 4.4



รูปที่ 4.4 แสดงกราฟ xy ของข้อมูล 1-5 และค่าเริ่มต้นของกลุ่มเป็นรูปกากบาท

คำนวณระยะห่างระหว่างข้อมูลกับค่าเริ่มต้นของกลุ่มซึ่งสามารถแสดงได้ดังตารางที่ 4.3

ตารางที่ 4.3 แสดงการจัดกลุ่มของอัลกอริทึมแบบ K-means

	กลุ่มที่ 1 (4,30)	กลุ่มที่ 2 (6,100)
1	70.0	3.0
2	40.4	30.3
3	10.2	60.0
4	3	70.2
5	20.4	50.0

จะได้ว่าในกลุ่มที่ 1 มีสมาชิกคือ ข้อมูลที่ 3 และ 4 ในกลุ่มที่ 2 มีสมาชิกคือ ข้อมูลที่ 1 2 และ 5 จากนั้นทำการคำนวณค่าตัวแทนของกลุ่มใหม่ซึ่งใช้ค่าจุดศูนย์กลางมวลของสมาชิกจะได้ตัวแทนกลุ่มที่ 1 ใหม่เป็น (3,5,30) ตัวแทนกลุ่มที่ 2 ใหม่เป็น (7,73.3)

หลังจากนั้นทำการหาสมาชิกกลุ่มใหม่ และหาตัวแทนกลุ่มใหม่ ทำต่อไปจนกระทั่งสมาชิกในกลุ่มไม่เปลี่ยนแปลงหรือตัวแทนของกลุ่มไม่เปลี่ยนแปลง

4.4.3 เปรียบเทียบการจัดกลุ่มข้อมูลแบบ Intrinsic และการจัดกลุ่มข้อมูลโดยใช้ Self-Organizing Map

ในการเปรียบเทียบการจัดกลุ่มข้อมูลของ Intrinsic กับ SOM นั้น ส่วนใหญ่ SOM จะถูกนำไปเปรียบเทียบกับอัลกอริทึม K-means ของการจัดกลุ่มข้อมูล Intrisic แบบ partition ในงานวิจัยของ [16] กล่าวว่าการจัดกลุ่มข้อมูลแบบ SOM ถือได้ว่าใกล้เคียงกับการจัดกลุ่มข้อมูลแบบ partition เนื่องจากข้อมูลที่ได้หลังจากการแบ่งกลุ่มแล้วจะได้กลุ่มที่แบ่งแยกออกมาชัดเจนอยู่ในโหนดของ SOM และมีการกำหนดจำนวนโหนดที่ใช้เหมือนกับอัลกอริทึมการจัดกลุ่มของ K-mean ที่มีการกำหนดกลุ่มไว้ล่วงหน้า

แต่การกำหนดโหนดของ SOM สามารถที่จะกำหนดขนาด ความกว้าง ความยาว และรูปร่างของนิวรอนได้ ซึ่งมีผลต่อความสัมพันธ์ระหว่างโหนดของแผนภาพ ต่างจาก K-means ที่เพียงแค่กำหนดจำนวนกลุ่มที่ต้องการแค่นั้น

ในการจัดกลุ่มแบบ K-means มีการกำหนดค่าเริ่มต้นของกลุ่มเหมือนกับ SOM ที่มีการกำหนดค่าเริ่มต้นของแต่ละโหนด ในกระบวนการเรียนรู้ของ SOM จะมีการหาโหนดชนะสำหรับข้อมูล ซึ่งเหมือนกับการหากลุ่มที่มีค่าใกล้เคียงกับข้อมูลในการจัดกลุ่มแบบ K-means ใน SOM เมื่อเจอหาโหนดชนะสำหรับข้อมูลนั้นได้แล้วจะมีการปรับค่าโหนดนั้นเลย แต่ในการจัดกลุ่มแบบ K-means จะต้องหากลุ่มให้ข้อมูลจนครบทุกตัวก่อนจึงจะมีการหาค่าตัวแทนกลุ่มใหม่

ข้อแตกต่างที่สำคัญอีกประการคือ ในการจัดกลุ่มแบบ SOM นั้นจะมีการคิดความสัมพันธ์ของโหนดใกล้เคียง โดย SOM พยายามให้โหนดที่ติดกันนั้นมีข้อมูลที่คล้ายกัน ซึ่งทำให้ผลลัพธ์ที่

ได้จากการจัดกลุ่มข้อมูลแบบ SOM เมื่อถูกแสดงอยู่ในรูปแบบของแผนภาพ 2 มิติ จะสามารถช่วยวิเคราะห์ความสัมพันธ์ ความหนาแน่นของข้อมูลที่มีมิติสูงได้เป็นอย่างดี

การจัดกลุ่มโดยใช้ SOM ค่อนข้างเหมาะสำหรับการจัดกลุ่มข้อมูลที่มีมิติสูงอย่างเช่น การจัดกลุ่มเอกสาร มากกว่าใช้วิธีการจัดกลุ่มแบบ K-mean เนื่องจากผลลัพธ์ที่ได้ของ SOM จะถูกแสดงในแผนภาพ ซึ่งเราสามารถวิเคราะห์ข้อมูลได้ดีกว่า

4.4.4 การวัดประสิทธิภาพการจัดกลุ่มข้อมูล

ในการวัดประสิทธิภาพของการจัดกลุ่มข้อมูลเราสามารถจำแนกประเภทการวัดได้เป็นสองประเภทหลักคือ

1. การวัดที่อ้างอิงจากภายใน(Internal quality)

   การวัดประเภทนี้ไม่จำเป็นต้องอาศัยข้อมูลใดๆ จากภายนอก โดยการวัดจะอาศัยการเปรียบเทียบฟังก์ชั่นที่ในการวัดความคล้าย เช่น การวัดค่า mean squared error ซึ่งเป็นที่นิยมเป็นอย่างมาก

2. การวัดที่อ้างอิงจากภายนอก(External quality)

   การวัดประเภทนี้จำเป็นต้องอาศัยข้อมูลจากภายนอกเข้ามาช่วยตัดสิน เช่น ข้อมูลกลุ่มของเอกสารที่ถูกกำหนดไว้ล่วงหน้าโดยผู้เชี่ยวชาญ เช่น การวัดค่าพลังงาน (entropy) และการวัดค่า F-measure

4.4.4.1 การวัดค่าแบบ Sum Squared Error

คือ ผลรวมของความผิดพลาดของข้อมูลแต่ละอันกับจุดศูนย์กลางกลุ่ม การวัดความผิดพลาดสามารถวัดได้จากการพิจารณาว่าข้อมูลนั้นอยู่ห่างจากจุดศูนย์กลางกลุ่มเท่าไหร่ ซึ่งสามารถแสดงสมการได้ดังนี้

$$SSE = \sum_{i=1}^{k} \sum_{x \in C_i} \| x - c_i \|^2 \qquad (4.3)$$

โดยที่ k คือจำนวนกลุ่มทั้งหมด

C_i คือ กลุ่มแต่ละกลุ่ม

x คือ สมาชิกของกลุ่มนั้น

$\|x - c_i\|$ คือ ระยะห่างระหว่างข้อมูลกับจุดศูนย์กลางของกลุ่ม

4.4.4.2 การวัดค่าแบบเอนโทรพี(entropy)

คือ การวัดโดยอาศัยค่าความน่าจะเป็นของข้อมูลที่อยู่ในกลุ่ม นั้นคือ ค่าเอนโทรพีจะมีค่าเป็นศูนย์เมื่อสมาชิกทุกตัวของกลุ่มจัดขึ้นใหม่นั้นเป็นกลุ่มเดียวกันกับกลุ่มที่จัดโดยผู้เชี่ยวชาญ ค่าเอนโทรพีจะมีค่าสูงสุดเมื่อสมาชิกทุกตัวในกลุ่มใหม่เป็นคนละกลุ่มกับกลุ่มที่จัดโดยผู้เชี่ยวชาญ

$$E_j = -\sum_i p_{ij} \log(p_{ij}) \tag{4.4}$$

โดยที่ค่าเอนโทรพีรวมของทุกกลุ่มสามารถหาได้จาก

$$E_{cs} = \sum_{j-1}^{m} \frac{n_j \times E_j}{n} \tag{4.5}$$

โดยที่ p คือความน่าจะเป็นที่สมาชิกของโหนด i ขึ้นอยู่กับกลุ่ม j

n_j คือ จำนวนข้อมูลทั้งหมดในโหนด j

n คือ จำนวนเอกสารทั้งหมด

ตัวอย่างเช่น เรามีข้อมูลที่ถูกจัดกลุ่มแล้วโดยผู้เชี่ยวชาญ 3 กลุ่ม ดังนี้ C_1={d_1, d_2, d_3}, C_2={ d_4, d_5, d_6}, C_2={ d_7, d_8, d_9, d_{10}} หลังจากที่เราจัดกลุ่มโดยใช้อัลกอริทึมใด ๆ แล้วได้ดังนี้ K_1= {d_1, d_2, d_3}, K_2= {d_4, d_6, d_7}, K_3= {d_5, d_8, d_9, d_{10}} เราสามารถวัดค่าเอนโทรพีของ K_1 K_2 K_3 ได้ดังนี้

E_1 = -(3/3 * log(3/3)) = 0

E_2 = - (2/3*log(2/3)) +(1/3*log1/3)) = -(-0.11739-0.15904) = 0.27643

E_3 = - (3/4*log(3/4))+(1/4*log(1/4)) = -(-0.09370-0.15051) = 0.24421

เราสามารถหาค่าเอนโทรพีรวมของทั้ง 3 กลุ่มได้ดังนี้

E_{cs} = 0 + (3*0.27643/10) + (4*0.24421/10) = 0.180613

ในบทนี้ได้กล่าวถึงอัลกอริทึม agglomerative hierarchical และ K-means ซึ่งเป็นอัลกอริทึมที่นิยมใช้ในการจัดกลุ่มข้อมูลและเอกสารและยกตัวอย่างให้เห็นอย่างชัดเจน พร้อมทั้งกล่าวถึง SOM ในแง่ของการจัดกลุ่มข้อมูลและเปรียบเทียบกับอัลกอริทึม K-means สุดท้ายได้กล่าวถึงค่าที่ใช้ในการวัดประสิทธิภาพของการจัดกลุ่มข้อมูลซึ่งมีทั้งแบบ internal และ external

ในบทถัดไปจะกล่าวถึงการทดลองเปรียบเทียบประสิทธิภาพของโมเดล SOM และ HS-SOM และการประยุกต์ใช้งานโมเดลทั้งสองในการจัดกลุ่มเอกสาร พร้อมทั้งวัดประสิทธิภาพของการจัดกลุ่มเอกสารที่ได้จากโมเดลทั้งสอง

บทที่ 5

การทดลองและผลการทดลอง

ในบทนี้จะกล่าวถึงการทดลองเพื่อสอบประสิทธิภาพของโมเดล SOM เปรียบเทียบกับโมเดล HS-SOM ที่ผู้วิจัยได้พัฒนาขึ้น โดยการทดลองประกอบไปด้วย 2 การทดลอง การทดลองที่ 1 เป็นการทดสอบกับชุดข้อมูล 2 มิติเพื่อสร้างเป็นแผนภาพโครงข่าย การทดลองที่ 2 เป็นการทดสอบการจัดกลุ่มเอกสาร เพื่อหาประสิทธิภาพในการจัดกลุ่มเอกสารของทั้งสองโมเดล

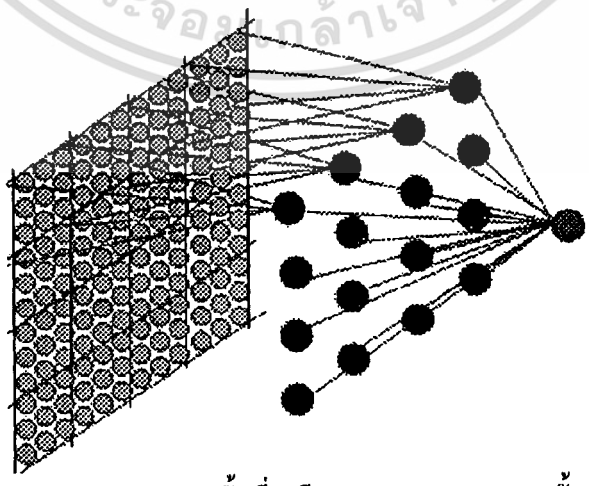
5.1 การทดลองที่ 1 การจัดกลุ่มข้อมูลสองมิติ

5.1.1 จุดประสงค์ของการทดลอง

เปรียบเทียบประสิทธิภาพของทั้ง 2 โมเดล โดยการสร้างแผนภาพ SOM และ HS-SOM จากชุดข้อมูลตัวเลข 2 มิติ

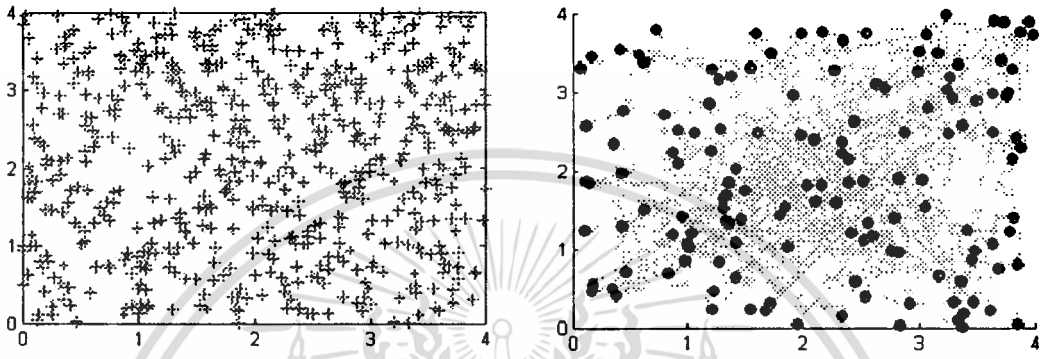
5.1.2 ขั้นตอนการทดลอง

เราได้กำหนดสภาวะในการทดลองดังนี้คือ โมเดล SOM มีขนาด 12x12 โดยมีโครงสร้างแบบสี่เหลี่ยม และโมเดล HS-SOM ในชั้นที่ 1 จะเป็นแผนภาพขนาด 12x12 เช่นเดียวกัน ทั้งนี้เพื่อจะได้เปรียบเทียบผลลัพธ์สุดท้ายของแผนภาพได้ ในชั้นที่ 2 ของโมเดล HS-SOM จะเป็นแผนภาพขนาด 4x4 ในแต่ละโหนดของแผนภาพชั้นที่ 2 จะประกอบไปด้วยสมาชิกของโหนดในแผนภาพชั้นที่ 1 จำนวน 9 โหนด ขนาด 3x3 สามารถแสดงได้ดังรูปที่ 5.1



รูปที่ 5.1 แสดงโมเดล HS-SOM แผนภาพชั้นที่ 1 มีขนาด 12x12 แผนภาพชั้นที่ 2 มีขนาด 4x4

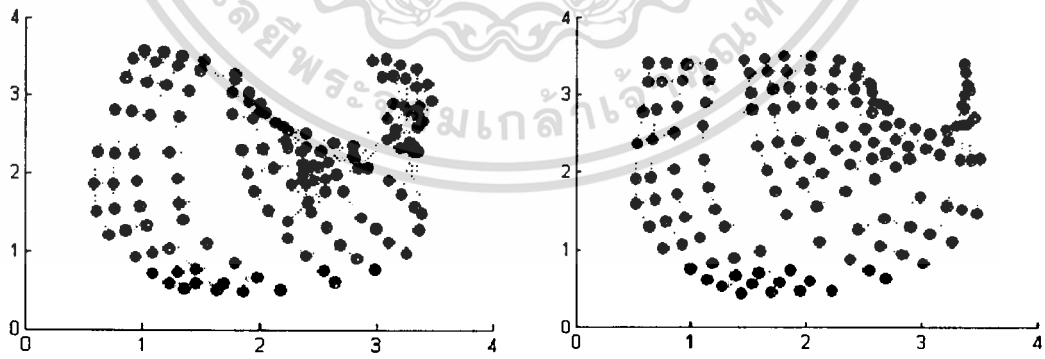
ในการทดลองเรากำหนดค่าเริ่มต้นเวกเตอร์น้ำหนักของแต่ละโหนดโดยการสุ่มค่าในช่วง 0-4 สำหรับข้อมูลที่ใช้ในการทดลองเป็นเวคเตอร์ 2 มิติ จำนวน 600 เวคเตอร์ โดยทำการสุ่มค่าตั้งแต่ 0-4 เช่นเดียวกัน ค่าเวคเตอร์น้ำหนักเริ่มต้น และ อินพุตเวคเตอร์ซึ่งเหมือนกันทั้งโมเดล SOM และ HS-SOM สามารถแสดงได้ดังรูปที่ 5.2



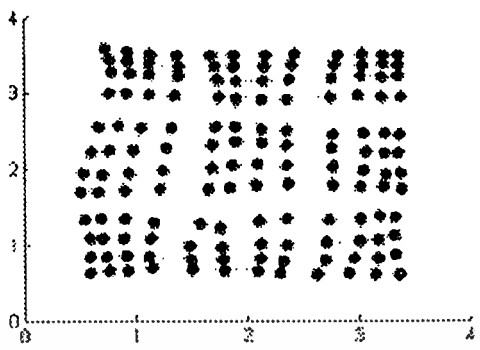
ก. กราฟอินพุตเวคเตอร์ ข. เวคเตอร์น้ำหนักเริ่มต้นของแผนภาพ SOM และ HS-SOM

รูปที่ 5.2 แสดงกราฟของอินพุตเวคเตอร์และค่าเวคเตอร์น้ำหนักเริ่มต้นของ SOM และ HS-SOM

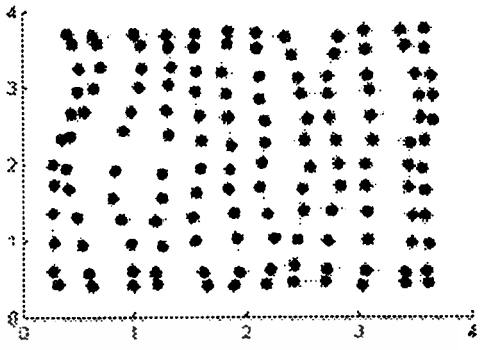
กำหนดอัตราการเรียนรู้ของสมการเป็น 0.09 และกำหนดรัศมีของโหนดใกล้เคียงเป็น 3 จำนวนรอบของการเรียนรู้ทั้งหมดเป็น 1000 รอบ แผนภาพ SOM และ HS-SOM หลังจากการเรียนรู้แสดงดังรูปที่ 5.3 และ 5.4 ตามลำดับ



ก. แผนภาพ SOM ณ จำนวนรอบที่ 10 ข. แผนภาพ SOM ณ จำนวนรอบที่ 50

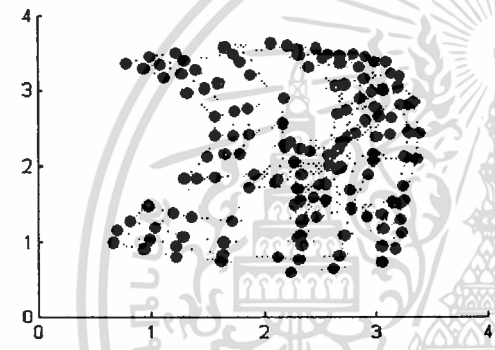


ค. แผนภาพ SOM ณ จำนวนรอบที่ 500

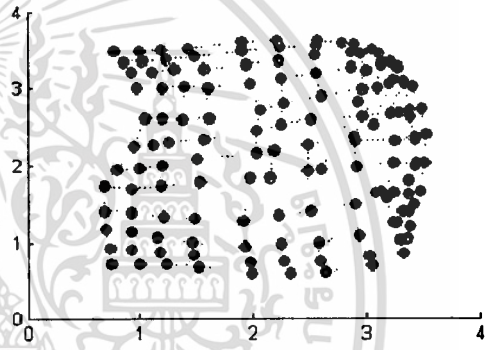


ง. แผนภาพ SOM ณ จำนวนรอบที่ 1000

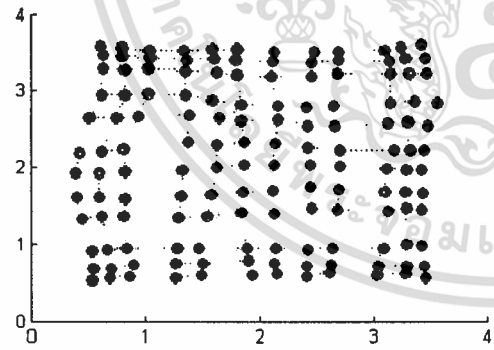
รูปที่ 5.3 แสดงแผนภาพ SOM ขนาด12x12 ที่ได้ ณ จำนวนรอบที่แตกต่างกัน



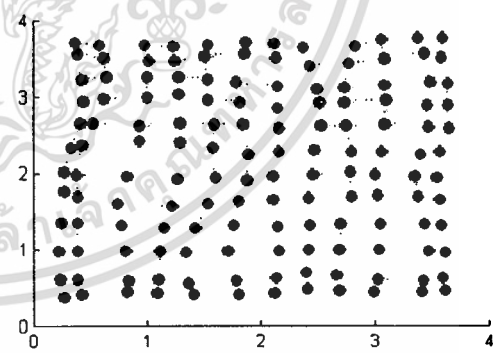
ก. แผนภาพ HS-SOM ณ จำนวนรอบที่ 10



ข. แผนภาพ HS-SOM ณ จำนวนรอบที่ 50



ค. แผนภาพ HS-SOM ณ จำนวนรอบที่ 500



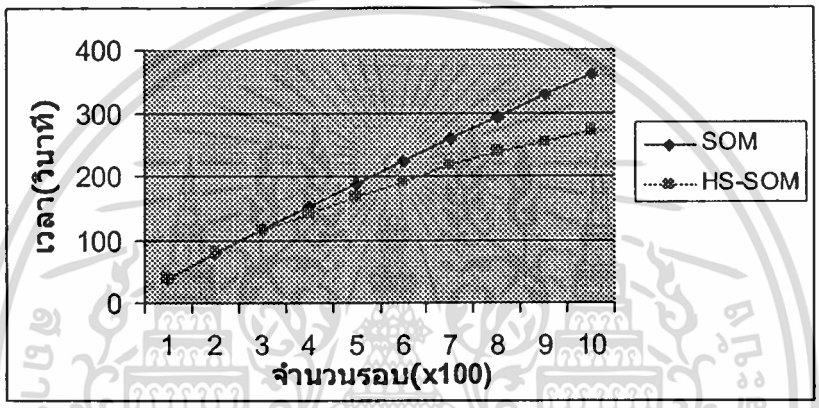
ง. แผนภาพ HS-SOM ณ จำนวนรอบที่ 1000

รูปที่ 5.4 แสดงแผนภาพ HS-SOM 2 ชั้น ชั้นแรก 12x12 ชั้นที่สอง 4x4 ที่ได้ ณ จำนวนรอบที่แตกต่างกัน

ในรูปที่ 5.2 ก. เป็นข้อมูลอินพุตเวคเตอร์ 2 มิติ 600 เวคเตอร์ เราต้องการที่จะจัดกลุ่มข้อมูลเหล่านี้โดยใช้โมเดล SOM และ HS-SOM โดยที่แผนภาพเริ่มต้นของโมเดลทั้งสองแสดงในรูป 5.2 ข. การปรับตัวของแผนภาพทั้งสองโมเดลเกิดขึ้นจากการปรับค่าของเวคเตอร์น้ำหนักของโหนด

ชนะและโหนดใกล้เคียงที่มีต่ออินพุตเวคเตอร์ที่เข้ามาในแต่ละรอบ จึงทำให้แผนภาพที่ไม่เป็นระเบียบในตอนแรกค่อยปรับตัวกระจากตามลักษณะของอินพุตที่เข้ามา จากรูปที่ 5.2 ก. ลักษณะของอินพุตกระจายอยู่ทั่วเป็นรูปสี่เหลี่ยม ดังนั้นลักษณะของแผนภาพที่ได้หลังจากการจะมีลักษณะเป็นรูปสี่เหลี่ยมกระจายตามลักษณะของอินพุตด้วย

จากรูปที่ 5.3 และ 5.4 ในรอบการเรียนรู้ที่เท่ากันการจัดเรียงตัวของแผนภาพ HS-SOM จะได้รูปทรงใกล้เคียงโมเดล SOM นั้นหมายถึงโมเดล HS-SOM ให้ประสิทธิภาพที่ใกล้เคียงกับโมเดล SOM ในจำนวนรอบที่เท่ากันแต่ใช้เวลาในการเรียนรู้น้อยกว่า



รูปที่ 5.5 แสดงกราฟเวลาในการเรียนรู้ของแผนภาพ SOM และ HS-SOM

จากกราฟแสดงเวลาในการเรียนรู้ของทั้งสองโมเดลในรูปที่ 5.5 จะเห็นได้ว่าในช่วงแรกโมเดล HS-SOM จะใช้เวลาใกล้เคียงกับโมเดล SOM ทั้งนี้เนื่องจากโมเดล HS-SOM ในช่วงแรกรัศมีของโหนดใกล้เคียงกว้างจึงทำให้ในการเรียนรู้แต่ละครั้งจะต้องมีการคำนวณหาค่าเวคเตอร์น้ำหนักในแผนภาพชั้นที่ 2 ใหม่ ซึ่งการหาจุดศูนย์ถ่วงของกลุ่มโหนดที่มีการเปลี่ยนแปลงค่าเวกเตอร์น้ำหนักในแผนภาพชั้นที่ 1 โดยการหาจุดศูนย์ถ่วงใหม่นี้อาจจะใช้เวลาใกล้เคียงการเปรียบเทียบหาระยะห่างของสองเวคเตอร์

พิจารณาในรอบที่ 300 รัศมีการเรียนรู้มีค่าเป็น 2 โมเดล HS-SOM เริ่มที่จะใช้เวลาในการเรียนรู้แต่ละรอบน้อยกว่าโมเดล SOM เนื่องจากเวลาที่ใช้ในการปรับโหนดในชั้นที่ 2 ของโมเดล HS-SOM ใช้น้อยกว่าเวลาที่หาโหนดชนะของโมเดล SOM

5.2 การทดลองที่ 2 การจัดกลุ่มข้อมูลเศรษฐกิจของแต่ละประเทศ

5.1.1 จุดประสงค์ของการทดลอง

เปรียบเทียบประสิทธิภาพของทั้ง 2 โมเดล โดยการสร้างแผนที่โลก 141 ประเทศแบ่งตามเขตเศรษฐกิจ จากชุดข้อมูลตัวบ่งชี้ 41 ตัว

5.2.2 ขั้นตอนการทดลอง

ในการทดลองนี้เป็นการจัดกลุ่มข้อมูลที่ได้มาจาก Worldbank (http://www.worldbank.org) เป็นข้อมูลที่เก็บในปี 2000 ลักษณะของข้อมูลที่ได้มาอยู่ในรูปของข้อมูล 141 ประเทศและแบ่งออกเป็น 4 กลุ่มเขตเศรษฐกิจ โดยแต่ละประเทศจะมีค่าบ่งชี้วัด โดยเราได้เลือกค่าบ่งชี้ทั้งหมด 41 ตัว ลักษณะตัวอย่างข้อมูลแสดงดังตารางที่ 5.1

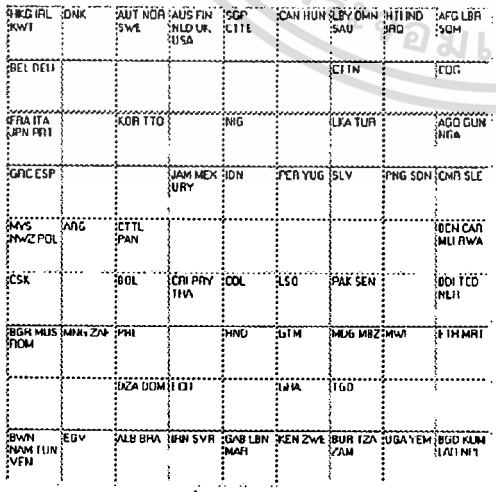
ตารางที่ 5.1 ตารางตัวอย่างข้อมูลของ Worldbank

Country	Indicator	Values
Thailand	Agriculture, value added (% of GDP)	10.26
Thailand	Aid per capita (current US$)	10.55
Thailand	Aircraft departures	101600
Thailand	CO2 emissions (metric tons per capita)	..
Thailand	Commercial energy use (kg of oil equivalent per capita)	1212.26
Thailand	Current revenue, excluding grants (% of GDP)	15.96
Thailand	Electric power consumption (kwh per capita)	1447.96
Thailand	Exports of goods and services (% of GDP)	66.86
Thailand	Fertility rate, total (births per woman)	1.84
Thailand	Fixed line and mobile phone subscribers (per 1,000 people)	142.67
Thailand	Foreign direct investment, net inflows (BoP, current US$)	3.37E+09
Thailand	GDP (current US$)	1.21E+11
Thailand	GDP growth (annual %)	4.65
Thailand	GNI per capita, Atlas method (current US$)	2020
Thailand	GNI, Atlas method (current US$)	1.23E+11
Thailand	Gross capital formation (% of GDP)	22.73
Thailand	High-technology exports (% of manufactured exports)	32.9

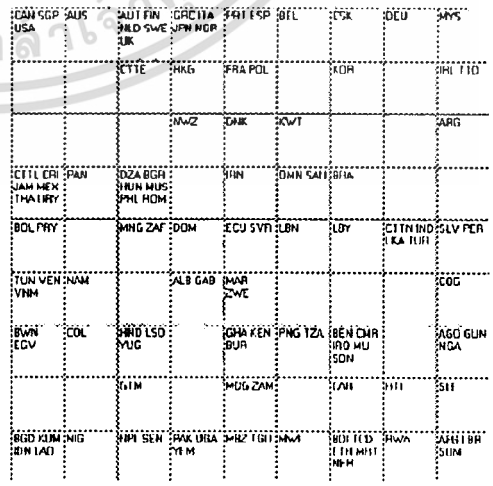
ตารางที่ 5.1 ตารางตัวอย่างข้อมูลของ Worldbank (ต่อ)

Country	Indicator	Values
Thailand	Illiteracy rate, adult female (% of females ages 15 and above)	6.14
Thailand	Illiteracy rate, adult total (% of people ages 15 and above)	4.52
Thailand	Immunization, measles (% of children under 12 months)	94
Thailand	Imports of goods and services (% of GDP)	58.22
Thailand	Improved sanitation facilities, urban (% of urban population with access)	96
Thailand	Improved water source (% of population with access)	84
Thailand	Industry, value added (% of GDP)	40.35
Thailand	Inflation, GDP deflator (annual %)	1.32
Thailand	Internet users	2300000
Thailand	Life expectancy at birth, total (years)	68.82
Thailand	Mortality rate, infant (per 1,000 live births)	25

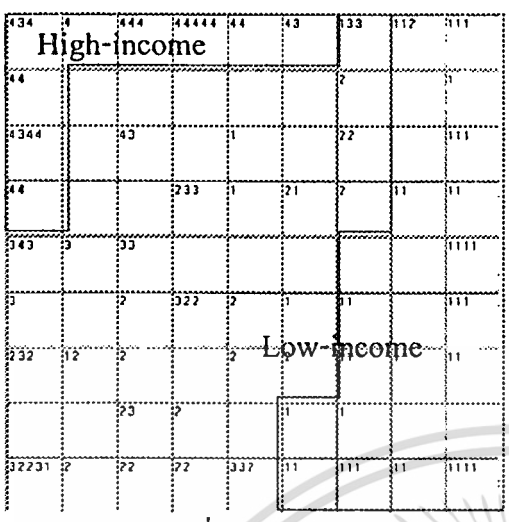
หลังจากนั้นเราจะทำการ Normalize ข้อมูลของทุกประเทศ ในการทดลองเราทำการทดลองเปรียบเทียบ SOM ขนาด 9x9 โหนด และ HS-SOM 2 ชั้น ชั้นแรกขนาด 9x9 ชั้นที่สองขนาด 3x3 โดยในแต่ละโหนดในชั้นที่สองมีสมาชิกโหนดในชั้นที่ 1 ขนาด 3x3 แบบไม่ซ้อนทับกัน(ดังรูปที่ 3.1) กำหนดรอบในการเรียนรู้ที่ 1000 รอบ ค่าเริ่มต้นของเวคเตอร์น้ำหนักในแต่ละโหนดสุ่มในช่วง 0.0001-0.00001 ผลลัพธ์ที่ออกมาแสดงดังรูปที่ 5.6



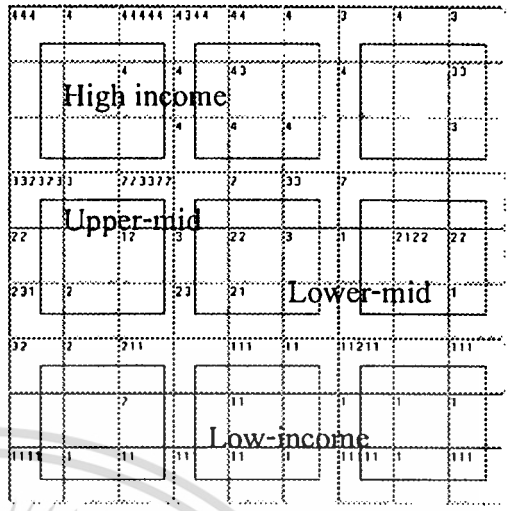
ก. แสดงแผนภาพที่ได้จากโมเดล SOM แสดงเป็นชื่อประเทศ



ข. แสดงแผนภาพที่ได้จากโมเดล HS-SOM แสดงเป็นชื่อประเทศ



ค. แสดงแผนภาพที่ได้จากโมเดล SOM แสดงเป็นกลุ่มของประเทศ



ง. แสดงแผนภาพที่ได้จากโมเดล HS-SOM แสดงเป็นกลุ่มของประเทศ

รูปที่ 5.6 แสดงลักษณะของแผนที่เขตเศรษฐกิจที่ได้หลังจากการเรียนรู้

พิจารณารูปที่ 5.6 ก และ ข แสดงแผนภาพเขตเศรษฐกิจโดยแสดงเป็นชื่อประเทศ จะเห็นได้ว่าประเทศที่พัฒนาแล้วเช่น ประเทศสหรัฐ แคนนาดา ออสเตรีย และกลุ่มสหภาพยุโรปจะอยู่บริเวณใกล้เคียงกัน เมื่อพิจารณารูป 5.6 ค และ ง แสดงแผนภาพโดยแบ่งเป็นกลุ่มประเทศ ประเทศที่พัฒนาแล้วคือ หมายเลข 4 ในรูปที่ 5.6 ค ซึ่งประเทศที่พัฒนาแล้วจะกระจายตัวมากกว่าและเป็นการยากในการกำหนดบริเวณของกลุ่ม ซึ่งต่างจากแผนภาพที่ได้จากโมเดล HS-SOM ในรูปที่ 5.6 ง จะเห็นว่ากลุ่มประเทศที่พัฒนาแล้วจะรวมกลุ่มกันอย่างเป็นระเบียบ สามารถที่จะกำหนดบริเวณหรือขอบเขตของกลุ่มโหนดได้ ทั้งนี้เนื่องจากโหนดในแผนภาพชั้นที่ 2 เป็นโหนดที่ทำหน้าที่เปรียบเสมือนอินเด็กซ์ที่ปรับค่าได้ ดึงข้อมูลที่มีค่าใกล้เคียงกันมาอยู่รวมกลุ่มกัน ดังนั้นจึงทำให้ข้อมูลไม่กระจายตัวจนเกินไป และสะดวกในการกำหนดขอบเขตและบริเวณของข้อมูลได้ง่าย

5.3 การทดลองที่ 3 การจัดกลุ่มเอกสารภาษาอังกฤษ

5.3.1 จุดประสงค์ของการทดลอง

เปรียบเทียบประสิทธิภาพของทั้ง 2 โมเดล ในการจัดกลุ่มเอกสารบทความภาษาอังกฤษ

5.3.2 การเตรียมการทดลอง

ในการทดลองนี้เป็นการจัดกลุ่มเอกสาร โดยเอกสารที่ใช้ในงานวิจัยนี้เป็นบทความภาษาอังกฤษ โดยใช้ชื่อบทความและบทคัดย่อเป็นตัวแทนของเอกสาร

เอกสารตัวอย่างที่ใช้ในการจัดกลุ่มเมื่อแบ่งตัวอย่างของเอกสารแสดงในรูปที่ 5.7

Title: A Review of Clustering Algorithm as Applied in IR.

Abstract: Cluster analysis is an important procedure not only in the social sciences but also in library and information science. For about half of a century, cluster analysis has been studied and employed in many fields. This paper is a review of cluster analysis. The definition of a cluster and the properties of clusters are introduced first.Then choice of variables and similarity measures is discussed. A general review of clustering methods is given in the third part while some related issues are addressed in the fourth part. The conclusion provides a few cautions on and a guide to cluster analysis.

รูปที่ 5.7 แสดงตัวอย่างเอกสารที่ใช้ในการจัดกลุ่ม

ก่อนการนำไปสร้างอินเด็กซ์ให้กับเอกสารเราจะนำเอกสารมาผ่านกระบวนการลดจำนวนคำ โดยเราจะตัดคำหยุด(stop words) [12] เช่น a and the และทำการหารากศัพท์ของคำเช่น classification classify และ classified จะแทนทุกคำด้วย classif อัลกอริทึมที่นำมาใช้ในการหารากศัพท์คือ อัลกอริทึม Porter stemming [12] หลังจากผ่านกระบวนการลดทอนคำศัพท์แล้วจะได้ดังรูปที่ 5.8

Title: review cluster algorithm appli ir.

Abstract: cluster analysi import procedur social scienc librari inform scienc half centuri cluster analysi studi emploi mani field paper review cluster analysi definit cluster properti cluster introduc first gener review cluster method given third part some relat issu address fourth part conclus provid few caution guid cluster analysi.

รูปที่ 5.8 แสดงตัวอย่างเอกสารหลังจากการลดทอนคำศัพท์แล้ว

เอกสารก่อนผ่านกระบวนการลดทอนคำศัพท์จะมีคำศัพท์ที่ไม่ซ้ำกันจำนวน 67 คำและมีจำนวนคำทั้งหมด 102 คำ หลังจากผ่านกระบวนการลดทอนจำนวนคำจะเหลือคำศัพท์ที่ไม่ซ้ำกันทั้งหมด 38 คำและมีจำนวนคำทั้งหมดได้ 52 คำ จะเห็นได้ว่าสามารถลดขนาดคำศัพท์ได้ประมาณ 50 เปอร์เซ็นต์

ในงานวิจัยนี้เราได้สร้างเวคเตอร์ต้นแบบของเอกสารทั้งหมดซึ่งมีส่วนประกอบของเวคเตอร์เป็นคำต่าง ๆ จากนั้นจะสร้างเป็นเวคเตอร์ของแต่ละเอกสาร โดยคิดน้ำหนักคำจากค่าผล

คูณของ *tf* (term frequency) ซึ่งเป็นความถี่ของคำที่ปรากฏในเอกสารและ ค่า *idf* (inverse document frequency) ซึ่งเป็นค่าที่ให้ความสำคัญกับคำที่มีพบในเอกสารจำนวนน้อยมากกว่าคำที่พบในเอกสารจำนวนมาก เนื่องจากจำที่พบในเอกสารจำนวนน้อยนั้นมีอำนาจในการจำแนกเอกสารได้ดีกว่า

การนำค่า *idf* มาคูณกับ *tf* เพื่อลดความสำคัญของคำที่ปรากฏในเอกสารจำนวนมากลง เพราะไม่สามารถเป็นตัวแทนของเอกสารได้ สมการที่ได้จะเป็นดังสมการที่ 5.1

$$w_{i,j} = f_{i,j} \times idf_i \tag{5.1}$$

โดยที่

$$f_{i,j} = \frac{freq_{i,j}}{\max(freq_{i,j})} \tag{5.2}$$

และ

$$idf_i = \log\left[\frac{N}{n_i}\right] \tag{5.3}$$

โดยที่ $f_{i,j}$ คือความถี่ของคำในรูปทั่วไป(normalized frequency)

$freq_{i,j}$ เป็นความถี่ของเทอม k_i ในเอกสาร d_j

N เป็นจำนวนเอกสารทั้งหมดในระบบ

n_i คือจำนวนเอกสารที่ปรากฏเทอม k_i

ดังนั้นเมื่อผ่านกระบวนการทั้งหมดแล้ว จะได้เอกสารที่ได้จะถูกแทนอยู่ในรูปของ $\vec{d}_j = \{w_{1,j}, w_{2,j}, w_{3,j}, ..., w_{t,j}\}$ โดยที่ $w_{i,j} \geq 0$ ในการทดลองเราแบ่งการทดลองเป็น 2 การทดลองย่อยดั้งนี้

5.3.3 การทดลองย่อยที่ 3.1

5.3.3.1 จุดประสงค์การทดลอง

ในการทดลองนี้ผู้วิจัยมีวัตถุประสงค์ต้องการที่จะทดสอบประสิทธิภาพในการจัดกลุ่มเอกสารที่มีจำนวนกลุ่มน้อยกว่าจำนวนโหนดในแผนภาพชั้นที่ 2 ของโมเดล HS-SOM โดยเปรียบเทียบกับโมเดล SOM

5.3.3.2 ขั้นตอนการทดลอง

ในการทดลองเปรียบเทียบการจัดเอกสารที่มีจำนวนกลุ่มของเอกสารน้อยกว่าโหนดในแผนภาพชั้นที่ 2 ของโมเดล HS-SOM เพื่อวิเคราะห์การกระจายและการรวมตัวของเอกสารเปรียบเทียบกับโมเดล SOM

เอกสารที่ผู้วิจัยนำมาใช้ในการจัดกลุ่มเอกสารมีจำนวน 150 บทความ โดยผู้วิจัยได้แบ่งกลุ่มเอกสารก่อนการจัดกลุ่มเป็นดังนี้คือ

ตารางที่ 5.2 แสดงจำนวนกลุ่มของเอกสารที่ใช้ในการทดลองย่อย 3.1

กลุ่มที่	เรื่อง	จำนวน
1	Self-organizing map	30
2	Hand writting	30
3	Knowledge base	30
4	Neural network	30
5	Information retrieval	30

จากนั้นจะนำเอกสารทุกเอกสารผ่านกระบวนการการตัดคำหยุด และหารากศัพท์ของคำเพื่อลดมิติของเวคเตอร์เอกสาร หลังจากนั้นจะได้เวกเตอร์ต้นแบบของเอกสาร โดยมีเวคเตอร์ต้นแบบที่ได้จะมีส่วนประกอบทั้งหมดเป็น 592 คำ จากนั้นจะนำเวคเตอร์ต้นแบบไปคำนวณเพื่อหาตัวแทนของเวกเตอร์แต่ละเอกสารโดยใช้วิธีการคำนวณน้ำหนักของแต่ละคำแบบ *tf-idf* ดังที่กล่าวมาในหัวข้อ 5.2.2 เมื่อกระบวนการเสร็จสิ้นจะได้เวกเตอร์ตัวแทนแต่ละเอกสารที่พร้อมจะใช้ในการทดสอบ

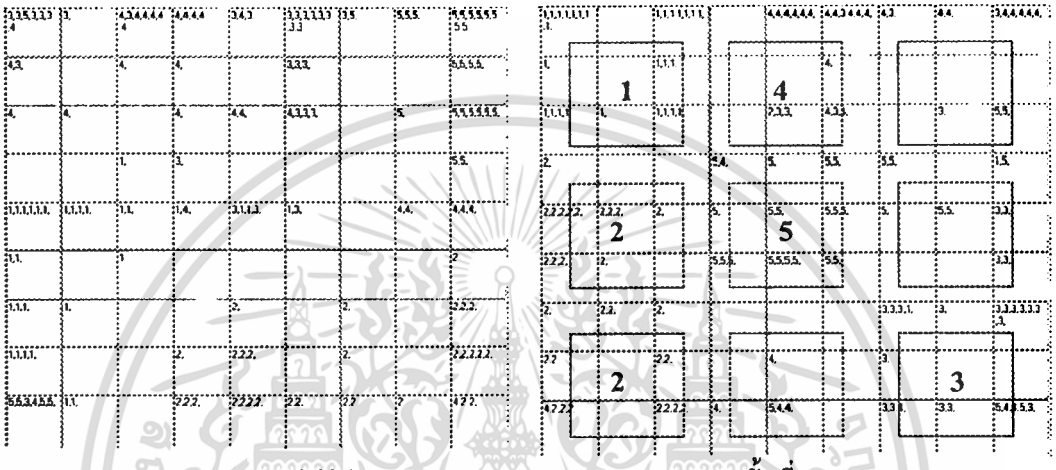
ในการทดลองผู้วิจัยใช้โมเดล SOM ขนาด 9x9 และใช้โมเดล HS-SOM แผนภาพชั้นที่ 1 มีขนาด 9x9 และแผนภาพชั้นที่ 2 มีขนาด 3x3 ในแต่ละโหนดของแผนภาพชั้นที่ 2 ประกอบด้วยจำนวนสมาชิกของโหนดในแผนภาพชั้นที่ 1 จำนวน 9 โหนด ซึ่งมีลักษณะโมเดลเหมือนรูป 3.1

ค่าเริ่มต้นการเรียนรู้ของโหนดในโมเดล SOM และ แผนภาพชั้นที่ 1 ของโมเดล HS-SOM สุ่มค่าช่วง 0-0.1 โดยใช้ทศนิยม 4 ตำแหน่ง อัตราการเรียนรู้เริ่มต้นที่ 0.09 และรัศมีการเรียนรู้เริ่มต้นที่ 3 หลังจากกระบวนการเรียนรู้เสร็จสิ้นเราจะนำตำแหน่งโหนดชนะของแต่ละเอกสารในการเรียนรู้รอบสุดท้ายของทั้งโมเดล SOM และ โมเดล HS-SOM มาสร้างเป็นแผนภาพดังรูป 5.8

จากผลการทดลองประการแรกพิจารณาการจัดเรียงตัวของเอกสารในโมเดล HS-SOM ในรูปที่ 5.8 ข พิจารณากลุ่มโหนดตามลักษณะการรวมกลุ่มในแผนภาพชั้นที่ 2 เอกสารกลุ่มที่ 1 จัดเรียงตัวอยู่ในกลุ่มโหนดมุมบนขวา เอกสารในกลุ่มที่ 2 ถูกแยกเป็น 2 กลุ่ม กลุ่มโหนดกลางซ้ายและกลุ่มโหนดล่างซ้าย ทั้งนี้เนื่องจากเอกสารในกลุ่มที่ 2 นั้นสามารถแยกเป็นกลุ่มย่อยข้างในได้อีก 2 กลุ่มใหญ่ เอกสารในกลุ่มที่ 3 ซึ่งมีเนื้อหาค่อนข้างต่างจากกลุ่มอื่นจะอยู่แยกไปในกลุ่มโหนด

บริเวณล่างขวา กลุ่มโหนดบริเวณขวาบน ขวากลางและ กลางขวา เป็นบริเวณที่เอกสารคลุมเครือ เช่น กลุ่มกลางขวาเป็นกลุ่มที่คลุมเครือระหว่างกลุ่ม 5 และกลุ่ม 3

พิจารณาการจัดเรียงตัวเอกสารในแผนภาพ SOM ในรูปที่ 5.9 ก การกระจายของกลุ่มเอกสารมีค่อนข้างมากกว่าเมื่อเปรียบเทียบกับการจัดเอกสารในโมเดล HS-SOM อีกทั้งยังกำหนดบริเวณของกลุ่มเอกสารได้ค่อนข้างยากกว่า



ก. แผนภาพ SOM ข. แผนภาพชั้นที่ 1 ของ HS-SOM

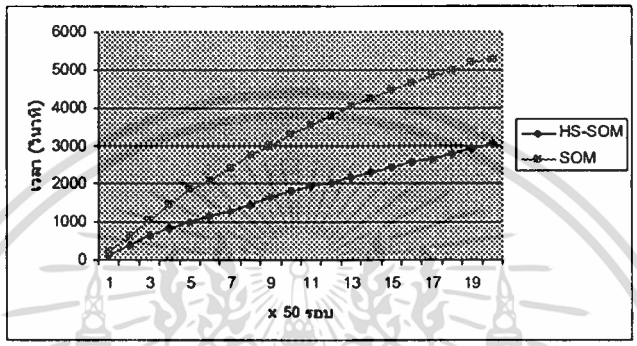
รูปที่ 5.9 แสดงแผนภาพ SOM และ HS-SOM ในการจัดเอกสารของการทดลองที่ 3.1

จะเห็นได้ว่าแผนภาพชั้นที่ 2 ที่แทรกเข้าไปใหม่นอกจากทำหน้าที่เป็นอินเด็กซ์ที่ปรับค่าได้ที่ใช้ช่วยในการจัดกลุ่มเอกสารไม่ให้เอกสารที่อยู่กลุ่มเดียวกันกระจายมากเกินไปแล้ว แผนภาพชั้นที่ 2 ยังสามารถช่วยให้ผู้ใช้มองเห็นภาพรวมของข้อมูลและลดเวลาในการสำรวจข้อมูลในแผนภาพชั้นที่ 1 โดยผู้ใช้จะเริ่มสำรวจเอกสารในแผนภาพชั้นที่ 2 ที่มีจำนวนโหนดน้อยกว่าก่อน จากนั้นจะทำการสำรวจโหนดในแผนภาพชั้นที่ 1 ในกลุ่มโหนดที่เป็นสมาชิกของแผนภาพชั้นที่ 2 ในแผนภาพ SOM รูป 5.9 ก. นั้นผู้ใช้จะต้องสำรวจเอกสารเองซึ่งไม่สามารถรู้ภาพรวมของตำแหน่งกลุ่มเอกสารได้ จะทำให้ผู้ใช้เสียเวลาในการสำรวจนานกว่า

ประการที่สองพิจารณาเวลาที่ใช้ในการจัดกลุ่มเอกสารของทั้งสองโมเดล จากรูปที่ 5.10 จะเห็นว่าช่วงแรกเวลาที่ใช้ในการเรียนรู้ของทั้งสองโมเดลใกล้เคียงกัน เนื่องจากโมเดล HS-SOM ในช่วงแรกรัศมีในการการปรับค่าโหนดใกล้เคียงในแผนภาพชั้นที่ 1 ยังกว้าง ดั้งนั้นจึงมีคำนวณค่าจุดศูนย์ถ่วงใหม่หลายกลุ่ม แต่ในรอบหลัง ๆ รัศมีของการเรียนรู้เริ่มลดลงจึงทำให้มีการคำนวณหาจุดศูนย์ถ่วงใหม่น้อยลงด้วยและใช้เวลาในการเรียนรู้น้อยลง

เวลาส่วนใหญ่ของการเรียนรู้ในโมเดล SOM คือ เวลาที่ใช้การหาโหนดชนะ แต่ในโมเดล HS-SOM เวลาส่วนใหญ่จะเป็นการปรับโหนดและการหาจุดศูนย์ถ่วงใหม่ของกลุ่มโหนด จะสังเกต

ได้จากในโมเดล SOM จะมีการเปลี่ยนแปลงเวลาเนื่องจากการลดค่ารัศมีในการเรียนรู้น้อยมาก นั้นหมายถึง เวลาที่ใช้ไปส่วนใหญ่ใช้ไปสำหรับการหาโหนดชนะเพราะเป็นค่าเวลาที่ค่อนข้างจะคงที่ แต่ในโมเดล HS-SOM เมื่อรัศมีในการเรียนรู้ลดลงเวลาที่ใช้ก็จะเริ่มลดลงไปเรื่อย ๆ ทำให้ที่จำนวนรอบเดียวกันระยะห่างของเวลาในการเรียนรู้ของทั้งสองโมเดลจะห่างขึ้นเรื่อย ๆ เมื่อจำนวนรอบเพิ่มขึ้นหรือรัศมีในการเรียนรู้ลดลง ซึ่งผลสุดท้ายทำให้ได้เวลาต่างกันถึง 24 %



รูปที่ 5.10 แสดงกราฟเวลาในการเรียนรู้ของการจัดเอกสารในการทดลองที่ 3.1

ประการที่สามพิจารณาประสิทธิภาพของการจัดกลุ่มเอกสาร จำนวนกลุ่มเริ่มต้นที่ผู้วิจัยจัดก่อนทำการเรียนรู้จะแบ่งออกเป็น 5 กลุ่มดังตารางที่ 5.2 จากตารางที่ 5.3 จะเห็นได้ว่าค่าเอนโทรพีของโมเดล HS-SOM จะมีค่าน้อยกว่าโมเดล SOM เนื่องจากโหนดในแผนภาพในชั้นที่ 2 ทำหน้าที่เป็นอินเด็กซ์แบบปรับค่าได้ เอกสารที่ใกล้เคียงกันจึงรวมกลุ่มกันได้ดีกว่า อย่างไรก็ตามถือว่าค่าเอนโทรพีของทั้งสองโมเดลใกล้เคียงกันมาก

ตารางที่ 5.3 แสดงค่าเอนโทรพีของการทดลองย่อยที่ 3.1

โมเดล	ผลรวมค่าเอนโทรพี
SOM	0.078
HS-SOM	0.063

จากจุดประสงค์ของการทดลองเพื่อหาความสัมพันธ์ในกรณีที่กลุ่มเอกสารมีน้อยกว่าจำนวนโหนดในแผนภาพชั้นที่ 2 นั้น พบว่าแต่ละโหนดในชั้นที่ 2 จะแทนแต่ละกลุ่มของเอกสาร ซึ่งทำให้เอกสารกลุ่มเดียวกันไม่กระจายมากเกินไป ซึ่งเป็นผลให้ค่าเอนโทรพีต่ำลงไปด้วย

5.3.4 การทดลองย่อยที่ 3.2

5.3.4.1 จุดประสงค์การทดลอง

ในการทดลองนี้ผู้วิจัยมีวัตถุประสงค์ต้องการที่จะทดสอบประสิทธิภาพในการจัดกลุ่มเอกสารที่มีจำนวนกลุ่มมากกว่าจำนวนโหนดในแผนภาพชั้นที่ 2 ของโมเดล HS-SOM โดยเปรียบเทียบกับโมเดล SOM

5.2.4.2 ขั้นตอนการทดลอง

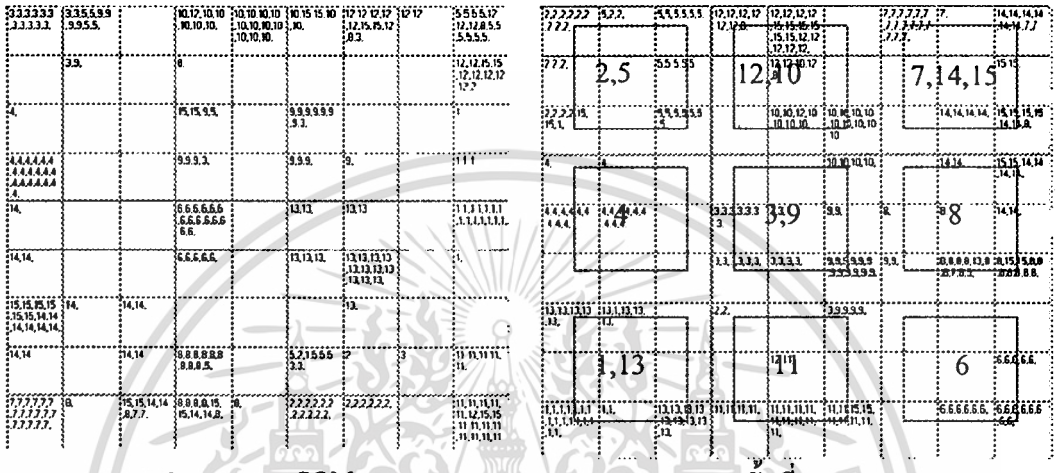
ในการทดลองเปรียบเทียบการจัดเอกสารที่มีจำนวนกลุ่มของเอกสารมากกว่าโหนดในแผนภาพชั้นที่ 2 ของโมเดล HS-SOM เพื่อวิเคราะห์การกระจายและการรวมตัวของเอกสารเปรียบเทียบกับโมเดล SOM

เอกสารที่ผู้วิจัยนำมาใช้ในการจัดกลุ่มเอกสารมีจำนวน 300 บทความ โดยผู้วิจัยได้แบ่งกลุ่มเอกสารเป็น 15 กลุ่มดังนี้คือ

ตารางที่ 5.4 แสดงจำนวนกลุ่มของเอกสารที่ใช้ในการทดลองย่อยที่ 3.2

กลุ่มที่	เรื่อง	จำนวน
1	Digital signal processing	15
2	Dataware house	15
3	XML	15
4	Self organizing map	15
5	Rough set	15
6	Intelligent agent	15
7	Offline handwriting recognition	15
8	Neural network	15
9	Knowledge base	15
10	Information extraction	15
11	Genetic Algorithm	15
12	Information retrieval	15
13	Image processing	15
14	Charactor recognition	15
15	Fuzzy set	15

หลังจากผ่านกระบวนการเตรียมเอกสารจะได้เวคเตอร์ต้นแบบที่มีส่วนประกอบทั้งหมดเป็น 875 คำ จากนั้นจะหาเวคเตอร์ของตัวแทนแต่ละเอกสารโดยใช้วิธีการคำนวณน้ำหนักของแต่ละคำแบบ *tf-idf* ในการออกแบบโมเดลและการกำหนดค่าเริ่มต้นของตัวแปรต่าง ๆ จะเหมือนกับการทดลองย่อยที่ 2.1 แผนภาพที่ได้หลังจากการเรียนรู้เสร็จสิ้นสามารถแสดงได้ดังนี้

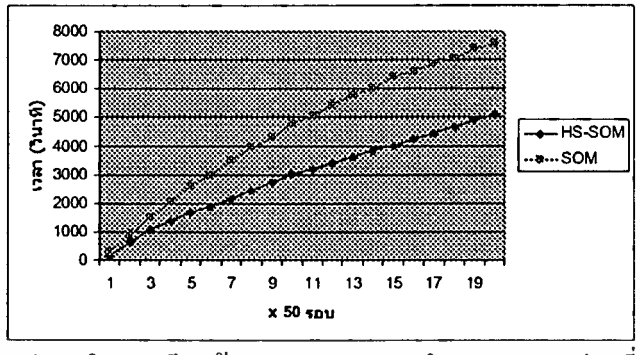


ก. แผนภาพ SOM ข. แผนภาพชั้นที่ 1 ของ HS-SOM

รูปที่ 5.11 แสดงแผนภาพ SOM และ HS-SOM ในการจัดเอกสารของการทดลองที่ 3.2

จากผลการทดลองประการแรกพิจารณาการจัดเรียงตัวของเอกสารในโมเดล HS-SOM แผนภาพชั้นที่ 2 ดังรูป 5.11 ข. จะเห็นได้ว่าเมื่อจำนวนกลุ่มของเอกสารที่ใช้ในการทดลองมีจำนวนมากกว่าจำนวนโหนดในแผนภาพชั้นที่สอง โหนดแต่ละโหนดในชั้นที่ 2 จะมีจำนวนกลุ่มของข้อมูลมากกว่าหนึ่งกลุ่ม เช่นโหนดบนขวาจะมีกลุ่มข้อมูลสามกลุ่มคือ กลุ่มที่ 7 14 และ 15 นั้นหมายถึงว่า เอกสารทั้งสามกลุ่มนี้มีความสอดคล้องกัน นั้นคือ Offline handwriting recognition, Charactor recognition และ Fuzzy set พิจารณาแผนภาพ SOM ในรูปที่ 5.11 ก. จะเห็นได้ว่าทั้งสามกลุ่มนั้นก็อยู่บริเวณเดียวกันและมีการจัดเรียงตัวใกล้เคียงกันมาก นั้นหมายความว่า แผนภาพชั้นที่ 2 ที่แทรกเข้าไปนั้นไม่ได้เข้าไปทำให้การจัดกลุ่มของเอกสารนั้นผิดเพี้ยนไป แต่ทำหน้าที่เป็นตัวอินเด็กซ์ของบริเวณกลุ่มโหนดนั้น ซึ่งทำให้การจัดกลุ่มข้อมูลสามารถทำได้เร็วขึ้น

ประการที่สองพิจารณาเวลาที่ใช้ในการเรียนรู้สามารถแสดงเป็นกราฟได้ดังรูปที่ 5.12 จะเห็นได้ว่าเวลาที่ใช้ในการเรียนรู้จะมากกว่าในการทดลองแรก เนื่องจากขนาดของเวกเตอร์เอกสารในการทดลองย่อยที่ 3.2 มีขนาดมากกว่า



รูปที่ 5.12 แสดงกราฟเวลาในการเรียนรู้ของการจัดเอกสารในการทดลองย่อยที่ 3.2

ประการที่สามพิจารณาประสิทธิภาพของการจัดกลุ่มเอกสาร ก่อนการทดลองเราคาดว่าค่าเอนโทรพีของ HS-SOM อาจจะมีค่ามากกว่า SOM เนื่องจากว่ากลุ่มเอกสารมีมากกว่าจำนวนโหนดในชั้นที่ 2 ของ HS-SOM แต่จากตารางที่ 5.2 จะเห็นได้ว่าค่าเอนโทรพีของโมเดล HS-SOM มีค่าน้อยกว่าโมเดล SOM เมื่อเทียบกันที่รอบเท่ากัน ทั้งนี้เนื่องจากก่อนการทดลองเราคิดว่าเมื่อจำนวนกลุ่มของเอกสารมีมากกว่าจำนวนโหนดในชั้นที่ 2 จะทำให้เอกสารไปรวมกันที่โหนดใดโหนดหนึ่งเยอะ แต่จากผลการทดลองที่ได้โหนดในชั้นที่ 2 ที่มีกลุ่มเอกสารอยู่มากที่สุดคือ 3 กลุ่ม ซึ่งในแผนภาพชั้นที่ 1 เอกสารทั้ง 3 กลุ่มจะอยู่แยกโหนดกันมีปะปนกันบ้างเล็กน้อยซึ่งพิจารณาจากแผนภาพ SOM ก็จะมีลักษณะการรวมกลุ่มเอกสารคล้ายกัน

จากตารางที่ 5.5 เมื่อพิจารณาค่าเอนโทรพีที่รอบเดียวกันของทั้งสองโมเดลพบว่า ค่าเอนโทรพีที่ได้จากการจัดกลุ่มเอกสารโดยใช้โมเดล HS-SOM นั้นมีค่าน้อยกว่าโมเดล SOM นั้นหมายความว่าที่จำนวนรอบเดียวกันโมเดล HS-SOM ให้ประสิทธิภาพที่ดีกว่าโมเดล SOM ด้วย

ตารางที่ 5.5 แสดงค่าเอนโทรพีของการทดลองย่อยที่ 3.2

ค่า Entropy รวม / โมเดล	รอบที่ 100	รอบที่ 500	รอบที่ 1000
SOM	0.1852	0.1641	0.1411
HS-SOM	0.1754	0.1505	0.1394

5.3.5 การทดลองย่อยที่ 3.3

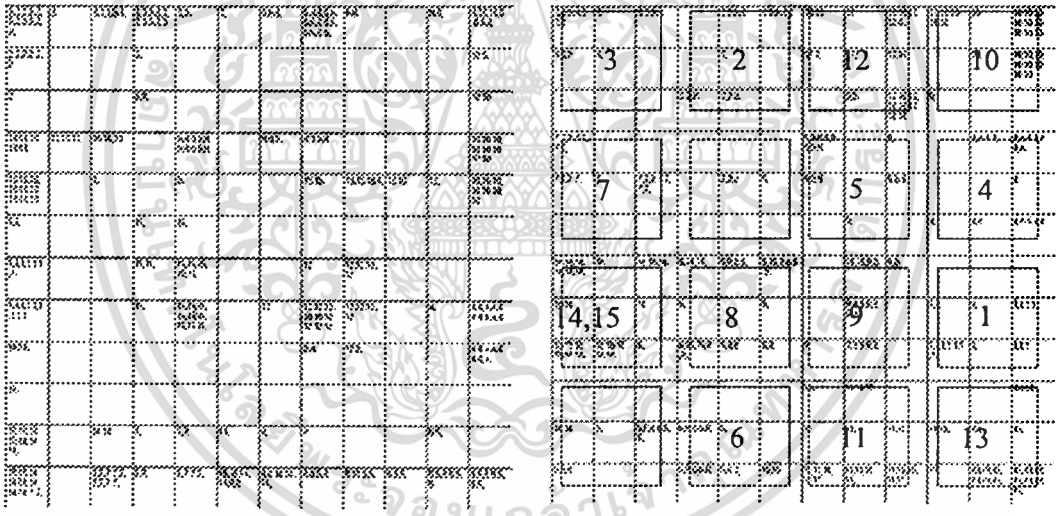
5.3.5.1 จุดประสงค์การทดลอง

ในการทดลองนี้ผู้วิจัยมีวัตถุประสงค์ต้องการที่จะทดสอบค่าเอนโทรพีที่ได้เมื่อทำการขยายแผนภาพ SOM และ HS-SOM

5.3.5.2 ขั้นตอนการทดลอง

ในการทดลองนี้ผู้วิจัยได้ใช้ข้อมูลที่ใช้ในการทดสอบชุดเดียวกับการทดลองย่อยที่ 3.2 คือ เอกสารจำนวน 300 เอกสาร โดยแบ่งกลุ่มไว้ล่วงหน้าเป็น 15 กลุ่ม ในการทดลอง

ในการทดลองผู้วิจัยใช้โมเดล SOM ขนาด 12x12 และใช้โมเดล HS-SOM แผนภาพชั้นที่ 1 มีขนาด 12x12 และแผนภาพชั้นที่ 2 มีขนาด 4x4 ในแต่ละโหนดของแผนภาพชั้นที่ 2 ประกอบด้วยจำนวนสมาชิกของโหนดในแผนภาพชั้นที่ 1 จำนวน 9 โหนด ขนาด 3x3 ซึ่งมีลักษณะโมเดลเหมือนรูป 5.1



ก. แผนภาพ SOM ข. แผนภาพชั้นที่ 1 ของ HS-SOM

รูปที่ 5.13 แสดงแผนภาพ SOM และ HS-SOM ในการจัดเอกสารของการทดลองที่ 3.3

ประการแรกพิจารณาการจัดเรียงตัวของแผนภาพ HS-SOM ในรูปที่ 5.13 ข. กลุ่มของเอกสารมีการกระจายไปตามโหนดชั้นที่ 2 ซึ่งเหมือนกับแผนภาพ SOM เนื่องจากเราเพิ่มจำนวนโหนดให้มากขึ้น ดังนั้นแนวโน้มที่กลุ่มของเอกสารจะแยกตัวกันไปสร้างอาณาเขตของตนเองก็เป็นไปได้มากขึ้น ประการที่สองพิจารณาเวลาในการเรียนรู้ ซึ่งจะใช้เวลาการเรียนรู้มากกว่าการทดลองที่ 3.2 เนื่องจากจำนวนโหนดเพิ่มขึ้น

ประการที่สามพิจารณาที่ค่าเอนโทรพีของทั้งสองโมเดล จะเห็นได้ว่าค่าเอนโทรพีที่ได้จะมีค่าน้อยกว่าการทดลองที่ 3.2 ทั้งนี้เนื่องจากการขยายโหนดในแผนภาพ SOM และ แผนภาพชั้นที่ 1

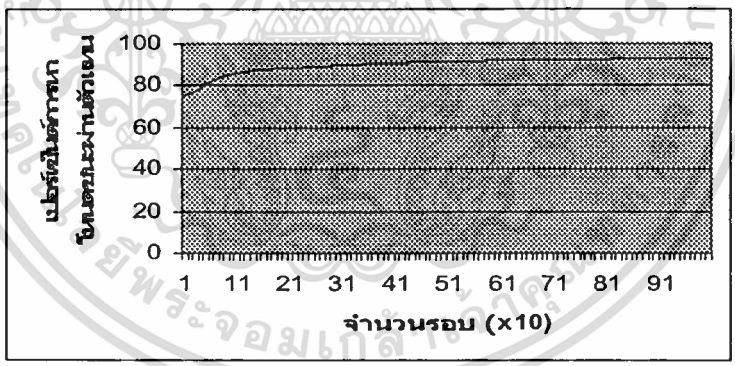
และชั้นที่ 2 ของ HS-SOMจึงทำให้กลุ่มของเอกสารค่อนข้างที่จะกระจาย แนวโน้มในการซ้อนทับกลุ่มกันมีน้อยเนื่องจากมีจำนวนโหนดมาก

ตารางที่ 5.6 แสดงค่าเอนโทรพีของการทดลองย่อยที่ 3.2

ค่า Entropy รวม / โมเดล	รอบที่ 100	รอบที่ 500	รอบที่ 1000
SOM	0.1558	0.1135	0.1047
HS-SOM	0.1487	0.1092	0.0952

5.4 วิเคราะห์เปอร์เซ็นต์การหาโหนดชนะและเวลาการหาโหนดชนะ

การหาโหนดชนะของโมเดล HS-SOM เป็นการหาโหนดชนะจากตัวแทนเฉลี่ยของกลุ่มก่อนจากนั้นถึงหาโหนดชนะจริงในแผนภาพชั้นที่ 1 ซึ่งอาจจะไม่ใช่โหนดชนะจริงเมื่อหาแบบเปรียบเทียบกับโหนดในแผนภาพชั้นที่ 1 ทุกตัว



รูปที่ 5.14 กราฟแสดงเปอร์เซ็นต์การหาโหนดชนะผ่านตัวแทนเฉลี่ยของโมเดล HS-SOM

จากรูปที่ 5.14 จะเห็นได้ว่าการหาโหนดชนะผ่านแผนภาพชั้นที่ 2 มีเปอร์เซ็นต์ใกล้เคียงกับการค้นหาโหนดชนะเมื่อเปรียบเทียบกับโหนดทุกตัว นั้นหมายความว่าเราสามารถใช้โหนดในแผนภาพชั้นที่ 2 ช่วยหาโหนดชนะจริงในแผนภาพชั้นที่ 1 ได้โดยที่ประสิทธิภาพของโมเดลยังใกล้เคียงกับของเดิม อีกทั้งยังสามารถลดเวลาในการหาโหนดชนะในแผนภาพจริงได้มากกว่า 20%

กรณีที่หาโหนดชนะผ่านแผนภาพชั้นที่ 2 ไม่ตรงกันกับการหาจากโหนดทุกตัวในแผนภาพที่ 1 ส่วนใหญ่จะเป็นโหนดที่อยู่ติดกันหรือภายในรัศมีโหนดใกล้เคียงแต่อยู่คนละโหนดในแผนภาพชั้นที่ 2 เนื่องจากมีการปรับค่าโหนดใกล้เคียงข้ามกลุ่มโหนดกัน

พิจารณาเวลาในการคำนวณของโมเดล SOM เปรียบเทียบกับโมเดล HS-SOM ที่ขนาดเวคเตอร์ที่แตกต่างกันดังตารางที่ 5.7

ตารางที่ 5.7 เปรียบเทียบเวลาการคำนวณของโมเดล SOM และ HS-SOM ที่มีมิติแตกต่างกัน

การทดลอง / จำนวนรอบ	การทดลองที่ 1		การทดลองที่ 3.1		การทดลองที่ 3.2	
	SOM	HS-SOM	SOM	HS-SOM	SOM	HS-SOM
100	100	102	446	286	761	519
250	190	167	1864	1042	2419	1832
500	280	232	3341	1836	4198	2912
1000	363	273	5275	3017	7623	5075

ในการทดลองที่ 1 จะเห็นได้ว่าในช่วง 100 รอบแรกโมเดล HS-SOM ใช้เวลาเฉลี่ยในการคำนวณมากกว่าโมเดล SOM ทั้งนี้เป็นเพราะในการทดลองที่ 1 เวลาเฉลี่ยที่ใช้ในการคำนวณส่วนใหญ่ของ HS-SOM จะเป็นการปรับหาค่าจุดศูนย์ถ่วงใหม่ในแผนภาพชั้นที่ 2 ซึ่งในช่วงแรกรัศมีของโหนดใกล้เคียงยังกว้างอยู่ดังนั้นจึงมีการปรับค่าจุดศูนย์ถ่วงมาก เมื่อเปรียบเทียบกับการหาโหนดชนะของโมเดล SOM แล้วจะใช้เวลามากกว่า แต่เมื่อรัศมีการปรับโหนดใกล้เคียงลดลงในรอบที่ 250 จะเห็นได้ว่าโมเดล SOM จะใช้เวลามากกว่า

นั้นคือในช่วงแรกเวลาที่เป็นการคำนวณส่วนใหญ่ของโมเดล HS-SOM คือการปรับหาค่าจุดศูนย์ถ่วงเพื่อปรับโหนดในแผนภาพชั้นที่ 2 ซึ่งใช้เวลามากกว่าการหาโหนดชนะในโมเดล SOM แต่เมื่อรัศมีของโหนดใกล้เคียงเริ่มลดลงเวลาปรับโหนดในแผนภาพชั้นที่ 2 ของโมเดล HS-SOMจะลดลงตาม แต่เวลาในการหาโหนดชนะของโมเดล SOM ยังมีค่าเฉลี่ยคงเดิม จึงทำให้ โมเดล HS-SOM จะใช้เวลาในการเรียนรู้น้อยกว่าโมเดล SOM

ในการทดลองที่ 3.2 และ 3.3 เมื่อเวคเตอร์มีมิติที่สูงขึ้นในรอบที่ 100 โมเดล HS-SOM ใช้เวลาน้อยกว่าโมเดล SOM นั้นหมายความว่าการหาโหนดชนะโดยใช้วิธีการวัดระยะห่างนั้นจะใช้เวลามากกว่าเวลาเฉลี่ยในการหาค่าจุดศูนย์ถ่วงเพื่อปรับค่าโหนดในแผนภาพชั้นที่ 2 ของโมเดล HS-SOM

บทที่ 6

สรุปการวิจัย และข้อเสนอแนะ

6.1 สรุปผลการวิจัย

Self-Organizing Map เป็นซึ่งเป็นนิวรอนเน็ตเวิร์กที่เหมาะสำหรับการวิเคราะห์ข้อมูลที่มีมิติสูงปัญหาสำคัญที่พบในการใช้งาน SOM คือ เมื่อมีข้อมูลจำนวนมากจำเป็นที่จะต้องเพิ่มจำนวนโหนดตาม ซึ่งส่งผลให้มีการคำนวณเพิ่มขึ้น ดังนั้นแผนภาพจะใช้เวลาในการเรียนรู้เพิ่มขึ้นตามเวลาที่ใช้ในการเรียนรู้ของ SOM ส่วนใหญ่ใช้ในการหาโหนดชนะ

ในงานวิจัยผู้วิจัยได้พัฒนาโมเดลใหม่ที่มีชื่อว่า High Speed Self-Organizing Map หรือ HS-SOM มีจุดประสงค์คือปรับปรุงโมเดล Self-Organizing Map และพัฒนาความเร็วในการเรียนรู้ โดยที่ประสิทธิภาพของโมเดลที่พัฒนาขึ้นมาจะต้องใกล้เคียงกับโมเดลเดิม

จากการทดลองที่ 1 ได้แสดงให้เห็นว่าโมเดล HS-SOM นั้นสามารถจัดเรียงแผนภาพข้อมูล 2 มิติ ได้รูปทรงมากกว่า SOM ในจำนวนรอบที่เท่ากัน โดยใช้เวลาในการเรียนรู้ที่น้อยกว่า

ในการทดลองในการจัดกลุ่มเอกสารพบว่าโหนดในแผนภาพที่แทรกเข้าไปทำหน้าที่เปรียบเสมือนอินเด็กซ์แบบปรับค่าได้ คอยรวมกลุ่มเอกสารที่มีความคล้ายกันไม่ให้กระจายมากเกินไป แต่ในขณะเดียวกันนั้นกลไกในการปรับโหนดใกล้เคียงจะเป็นตัวคอยช่วยไม่ให้เอกสารมารวมตัวกันที่กลุ่มใดกลุ่มหนึ่งมากเกินไป

เวลาเฉลี่ยในการเรียนรู้ของโมเดล HS-SOM ส่วนใหญ่ใช้ไปกับการปรับโหนดในแผนภาพชั้นที่แทรกแต่เวลาจะลดลงไปเรื่อย ๆ เมื่อรัศมีของการเรียนรู้ลดลง แต่ในโมเดล SOM เวลาที่ใช้ส่วนใหญ่จะเป็นการหาโหนดชนะซึ่งเป็นเวลาเฉลี่ยค่อนข้างคงที่

6.2 ข้อเสนอแนะ

ประการแรกเนื่องจากเอกสารที่ใช้ในการทดสอบเป็นบทความที่ผู้วิจัยได้ทำการจัดกลุ่มเอง ซึ่งถือได้ว่ายังไม่เป็นมาตรฐานพอ ในโอกาสต่อไปผู้เขียนจะทดสอบโมเดลกับฐานข้อมูล TREC ซึ่งเป็นฐานข้อมูลมาตรฐานที่ได้มีการจัดกลุ่มข้อมูลไว้แล้วโดยผู้เชี่ยวชาญ และเป็นฐานข้อมูลมาตรฐานที่นิยมใช้ในงานระบบค้นคืนเอกสาร

ประการที่สองข้อมูลหรือเอกสารในบางครั้งความจริงอาจจะสามารถอยู่ได้มากกว่าหนึ่งกลุ่มหรือเรียกเอกสารนั้นว่ามีความคลุ่มเครืออยู่ ซึ่งสอดคล้องกับอัลกอริทึมของ SOM ที่มีการปรับโหนดใกล้เคียง เพื่อที่ให้ข้อมูลที่ใกล้เคียงกันมาอยู่ในบริเวณเดียวกัน นั้นหมายถึงในหนึ่งโหนดนั้นอาจจะมีข้อมูลได้มากกว่าหนึ่งกลุ่มได้ไม่ผิดถ้าข้อมูลทั้งสองมีความใกล้เคียงกัน

ประการที่สามคือการจัดกลุ่มข้อมูลโดยใช้ SOM นั้นจุดประสงค์ไม่ได้มุ่งหวังที่ความถูกต้องของการจัดกลุ่มเพียงอย่างเดียว แต่คำนึงถึงความสัมพันธ์ของข้อมูลแต่ละกลุ่มด้วย ดังนั้นค่าเอนโทรพีซึ่งเป็นตัววัดประสิทธิภาพการจัดกลุ่มข้อมูลอาจจะไม่สามารถใช้วัดประสิทธิภาพของ SOM ได้เพียงอย่างเดียว ทั้งนี้ยังไม่มีค่าใดที่สามารถวัดประสิทธิภาพของแผนภาพ SOM ได้ ซึ่งในอนาคตผู้วิจัยจะทำการศึกษาค่าในการวัดประสิทธิภาพของแผนภาพ SOM เพื่อใช้เปรียบเทียบโมเดล HS-SOM กับ โมเดลที่ได้รับการปรับปรุงอื่น ๆ

ประการที่สี่โมเดล HS-SOM ที่ในเสนอในงานวิจัยนี้เป็นการนำเสนอแนวคิดการแทรกชั้นแผนภาพระหว่างชั้นของอินพุตและชั้นของเอาท์พุต โดยแผนภาพที่แทรกเกิดจากการรวมกลุ่มโหนดในแผนภาพชั้นที่ต่ำกว่า ซึ่งผลการทดลองขั้นต้นแสดงให้เห็นว่าโมเดล HS-SOM ให้ประสิทธิภาพที่ใกล้เคียงกับโมเดล SOM โดยที่ใช้เวลาน้อยกว่ามาก ในโอกาสต่อไปผู้เขียนจะวิจัยถึงผลกระทบของรูปแบบการรวมกลุ่มโหนดและจำนวนสมาชิกต่อประสิทธิภาพของโมเดล

เอกสารอ้างอิง

[1] Helge Ritter, Thomas Martinetz and Klaus Schulten. **Neural computation and self-organizing maps : an introduction** Massachusetts : Addison-Wesley. 1992.

[2] Xia Lin, Dagobert Soergal, Gary Marchioninl. "A Self-Organizing Semantic Map" ACM, 1991.

[3] Mu-Chun Su, Hsiao-Te Chang. "Fast Self-Organizing Feature Map Algorithm." IEEE Transaction on neural networks, vol. 11, no. 3, May 2000.

[4] J. Kangas. "Time dependent self-organizing maps for speech recognition." ICANN-91, Espoo, June 24-28, 1991.

[5] Kaski S., Honkela T., Lagus K., and Kohonen T. "WEBSOM-self-organizing maps of document collections" Neurocomputing, volume 21, 1998, pp. 101-117.

[6] Kohonen T. "Self-organization of very large document collections: State of the art", ICANN98, Springer, London, 1998, pages 65-74.

[7] Michael Dittenbach, Dieter Merkl, Andreas Rauber. "The Growing Hierarchical Self-Organizing Map" Proceedings of Int' l Joint Conference on Neural Networks(IJCNN 2000), Como, Italy, July 2000, pp. VI-15 – VI-19.

[8] Merkl, A. Rauber. "Document Classification with Unsupervised Neural Networks" Soft Computing in Information Retrieval, 2000, pp. 102 – 121.

[9] Samuel Kaski, Jari Kangast, Teuvo Kohonen. "Bibiography of Self-Organizing Map(SOM) Paper: 1981-1977" Neural Computing Survey I, 1998.

[10] Kaski, S. "Fast winner search for SOM-based monitoring and retrieval of high-dimensional data." Proceedings of ICANN99, Ninth International Conference on Artificial Neural Networks, London, vol. 2, pp 940-945.

[11] A.Rauber, D. Merkl. "The SOMLib Digital Library System" Proceedings of the 3rd Europ. Conf. on Research and Advanced Technology for Digital Libraries (ECDL'99), Paris, France, September 22 – 24 1999.

[12] R.Baeza-Yates and B. Ribeiro-Neto. **Modern Information Retrieval**. New York: ACM-Press. 1999.

[13] Qing Ma, Min Zhang, Ming Zhou. "Self-Organization of Chinese Semantic Maps Using TFIDF Term Weighting," The Second Workshop on Natural Language Processing and Neural Networks, Tokyo, Japan, November, 2001.

[14] Anil K. Jain and Richard C. Dubes. **Algorithms for Clustering Data**. New Jersey: Prentice Hall. 1988.

[15] Michael Steinbach, George Karypis and Vipin Kumar. "A comparison of Document Clustering Techniques." Technical Report, Department of Computer Science and Engineering University of Minnesota, 2000.

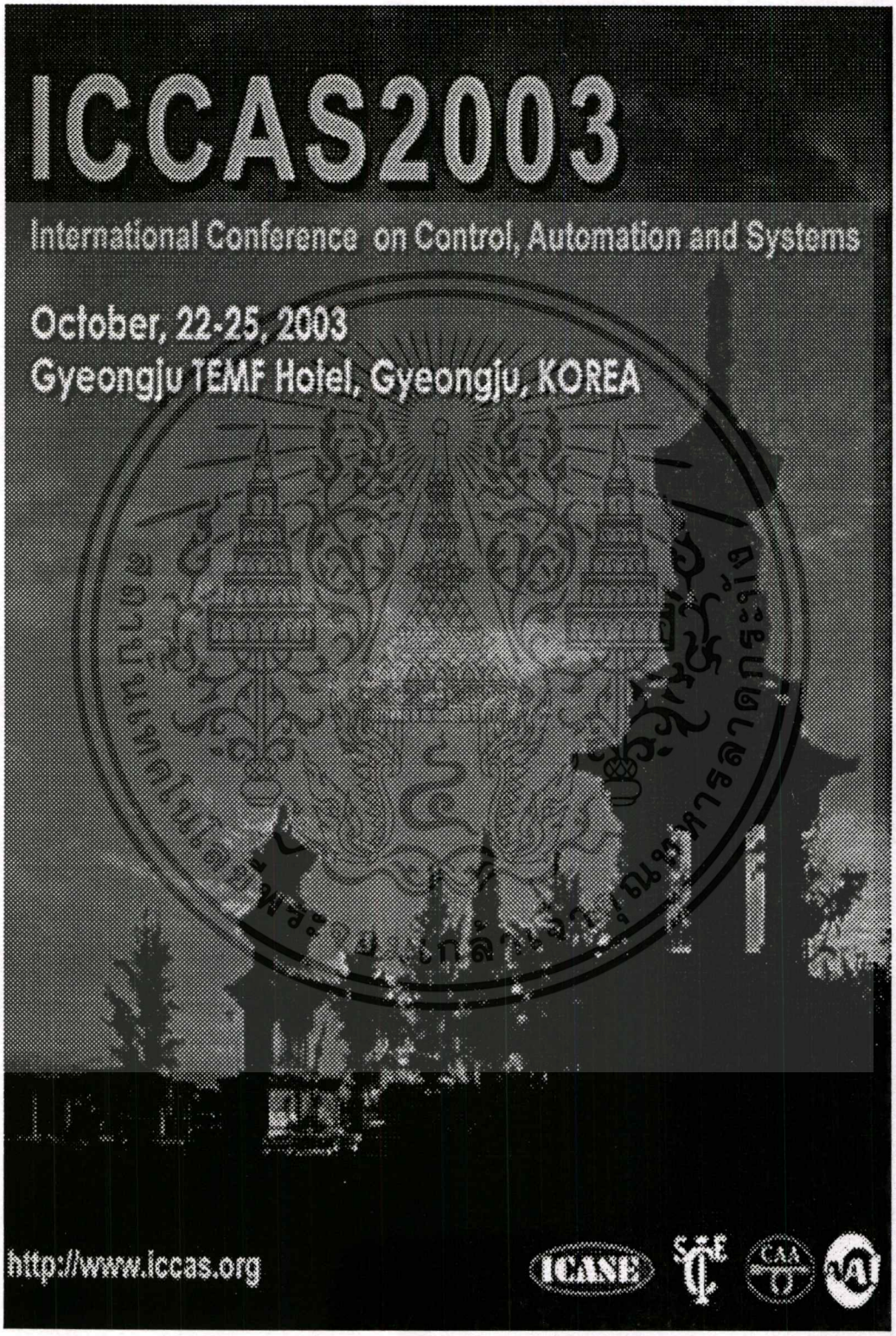
[16] A.Ultsch, C.Vetter. "Self-Organizing-Feature-Maps versus Statistical Clustering Methods" Research report, 1994.

ภาคผนวก

งานวิจัยที่ได้รับการตีพิมพ์

งานวิจัยที่ได้รับการตีพิมพ์

1. Rojanavasu P., Pinngern O. "Extended rough fuzzy sets for web search agent." Information Technologiy Interface (ITI 2003), Cavtat, Croatia, June 2003, pp. 403-407.
2. พรเทพ โรจนวสุ, เอื้อน ปิ่นเงิน. "การจัดกลุ่มเอกสารโดยใช้ Self-Organizing Map แบบความเร็วสูง." วิศวลาดกระบัง, ปีที่ 20, ฉบับที่ 3, กันยายน 2546, หน้า 30-35
3. Rojanavasu P., Pinngern O. "High Speed Self-Organizing Map for Document Clustering." International Conference on Control, Automation and Systems (ICCAS 2003), Gyeongju, Korea, October 2003, pp. 1056-1059.
4. Rojanavasu P., Ponnsuntikul P., Pinngern O. "Modified Self-Organizing Map for Document Clustering." Information and Computer Engineering Postgraduate Workshop 2004 (ICEP 2004), Phuket, Thailand, January 2004, pp. 111-115



ICCAS2003 October 22-25, Gyeongju TEMF Hotel, Gyeongju, Korea

High-Speed Self-Organizing Map for Document Clustering

*Ponlahp Rojanavasu and **Onsa Pinngern

Department of Computer Engineering, Faculty of Engineering
King Mongkut's Institute of Technology Ladkrabang
Bangkok, Thailand
(Tel : +66-2-739-2662; E-mail: s4061612@kmitl.ac.th*, kponsa@kmitl.ac.th **)

Abstract: Self-Oranizing Map(SOM) is an unsupervised neural network providing cluster analysis of high dimensional input data. The output from the SOM is represented in map that help us to explore data. The weak point of conventional SOM is when the map is large, it take a long time to train the data. The computing time is known to be O(MN) for training to find the winning node (M,N are the number of nodes in width and height of the map). This paper presents a new method to reduce the computing time by creating new map. Each node in a new map is the centroid of nodes' group that are in the original map. After create a new map, we find the winning node of this map, then find the winning node in original map only in nodes that are represented by the winning node from the new map. This new method is called "High Speed Self-Organizing Map"(HS-SOM). Our experiment use HS-SOM to cluster documents and compare with SOM. The results from the experiment shows that HS-SOM can reduce computing time by 30%-50% over conventional SOM.

Keywords: Self-organizing map, SOM, HS-SOM, Document clustering, Unsupervised neural network

1. INTRODUCTION

Self-Oranizing Map(SOM), unsupervised neural network is an providing cluster analysis of high dimensional input data[2,4,9]. The advantage of this approach is that its result is represented by using two dimensions mapping. This mapping show data's relationship including distribution and density of data. Unfortunately, high dense data affect the map to increase its nodes and take longer time for computation. Most time consuming for computation are finding the winning nodes. If there are MxN dimension in the map, finding the winning node will take O(MN) steps , Each step is taken for computing distance of vectors. For document clustering, vectors have a very high dimension so it take a long time to compute. In this paper, we propose new method to reduce computing time for finding the winning node. We separate nodes in map into small group and use centroid vector to find the winning node of each group. Then we find winning node from them again.

2. SELF-ORGANIZING MAP(SOM)

The self-organizing map (SOM) is one of the most prominent unsupervised artificial neural network models[1,3]. The model consists of neural elements called nodes. Each node i is assigned an n-dimensional weight vector m_i. That is $m_i \in \Re^n$, where $\Re^n$ is an n-dimensional space. It is necessary to note that the weight vectors have the same dimensionality as the input patterns.

The learning process of SOM may be described in terms of adaptive node for input vectors[5-7]. In each t learning-iteration, input vector $x(t)$ is randomed to compare with every node in the map for finding the output vector's winning node. Generally, function for comparision is Euclidean distance. The winning node c can thus be defined the smallest distance between input vector and nodes by

$$c : \| x(t) - m_c(t) \| = \min_i \{ \| x(t) - m_i(t) \| \} \quad (1)$$

The weight of winning node, c , is tuned by consider the differentiation of input vector and weight vector. Each learning-iteration is gradually reduced. Not only winning node is learning but also its neighborhood node, and thus the weight vectors become more similar to the input pattern. The respective node is more likely to win at future presentations of this input pattern shown in :

$$m_i(t+1) = m_i(t) + \alpha(t) \times h_{ci}(t) \times [x(t) - m_i(t)] \quad (2)$$

where t is the number of learning-iteration, $x(t)$ is current input vector, $m_i(t)$ is weight vector, $\alpha(t)$ is learning rate depended on number of iteration shown in:

$$\alpha(t) = \alpha_0(t) \times \frac{T - t}{T} \quad (3)$$

when T is the total number of iteration, t is the current iteration

$h_{ci}(t)$ is the neighborhood function. For convergence it is necessary that $h_{ci}(t) \to 0$ when $t \to \infty$. This means that with learning increasing, the neighborhood within which the nodes are activated will shrink and the same time the modifying rate of reference vectors will decrease. Usually we use Gaussian function.

$$h_{ci}(t) = \exp\left(-\frac{\| r_c - r_i \|^2}{2\sigma^2(t)}\right) \quad (4)$$

when $\| r_c - r_i \|$ is the distance between node i from the winning node c. $\sigma(t)$ is the radius of neighborhood node at t

$$\sigma(t+1) = 1 + (\sigma(t) - 1) \times \frac{T - t}{T} \quad (5)$$

Fig 1. shows 5x5 SOM.Firstly, the input vector $x(t)$ is randomly selected from input domain, then find winning node as the darkest node. Secondly, the weight vector $m_i(t)$ is tuned to $m_i(t+1)$ which will get close to the input vector. Finally, the neighborhood nodes of node c (lighter shading) are turned according to the previous equations.

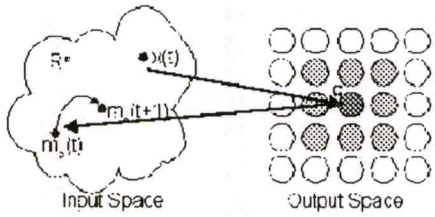


Fig 1. Structure of SOM and its learning

3. HIGH SPEED SELF-ORGANIZING MAP(HS-SOM)

Main problem for conventional SOM is that if the map has large size and too many dimensional data, it takes a long time to train the data. e.g. if there are MxN map, the time taken for finding the winning node is O(MN). This paper proposes the method to reduce this computation time by create new SOM map between input layer and SOM layer as shown in Fig. 2

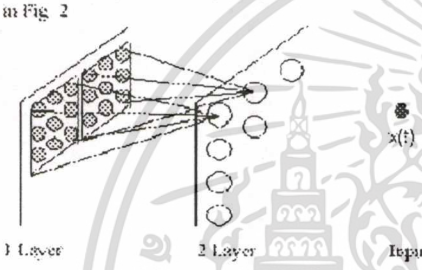


Fig 2. New layer is represented for reducing computation time

Weight vectors in the second layer are computed from nodes' centroid in first layer as equation (6)

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} \qquad (6)$$

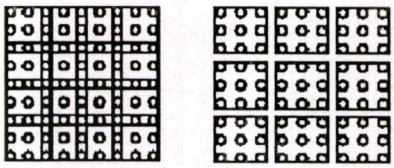
when x_i is vector component, n is total number of nodes in group

The new algorithm is used to find the winning node and adjust the neighborhood nodes according to the following steps:

New Algorithm

1. Find the winning node in the second layer map(Fig. 1)
2. Find the winning node in the first layer only in nodes that are represented by the winning node from the second layer
3. Adjust neighborhood nodes in the first layer
4. Adjust nodes in the second layer by finding new centroids for the changed nodes in the first layer

The computation for new winning node depend on the design of grouping in first layer which we can design both the number of member and its shape. We also can design the overlapping of groups(Fig 3). We will discuss the effect of the design in next section



a.) The overlapping design b.) The non-overlapping design

Fig 3. The design of node grouping

Computing time for non-overlap square group of HS-SOM(Fig. 3 (b)) is :

$$\text{time} = (axb)+(i \times j) \qquad (7)$$

where

i = number of groups' members in width of first layer,

j = number of groups' members in height of first layer,

a = number of nodes in width of the second layer,

b = number of nodes in height of the second layer,

and (ixj) is computation time for winning node in first layer (axb) is computation time for winning node in second layer. For example if M=9,N=9, Fig 3 a) is 4x4 overlapping design that has i=j=3, a=b=4 Fig 3 b) is 3x3 non-overlapping design that has a=b=3 and i=j=3. The computing time to find the winning node in conventional SOM is 9x9(MxN)=81, while computing time in HS-SOM Fig 3 a) is (4x4)+(3x3) = 25 and Fig 3 b) is (3x3)+(3x3) = 18. If i=j=1 then a=M, b=N and the computation time is in order of O(MN) that is the conventional computation time. If the map has large size we can increase some more layer to reduce computation time.

4. THE IMPLEMENTATION OF HS-SOM IN DOCUMENT CLUSTERING

4.1 Characteristic extraction of document

There are 1141 documents for clustering. We use document title and abstract to represent each document[1,2]. Before creating index term of document, we eliminate the stop words such as "a", "and", "the" and we use Porter stemming [8] to find the stemming word such as "classification", "classify", "classified" which we will replace with "classif"

After reducing the number of word in documents, the template vector is created for all documents. Its component compose of 1250 words. Each document uses this template vector to create its own vector by compute the term weight using equation (8)

$$w_{i,j} = f_{i,j} \times idf_i \qquad (8)$$

$$\text{when } f_{i,j} = \frac{freq_{i,j}}{\max_l (freq_{l,j})} \qquad (9)$$

$$\text{and } idf_i = \log\left[\frac{N}{n_i}\right] \qquad (10)$$

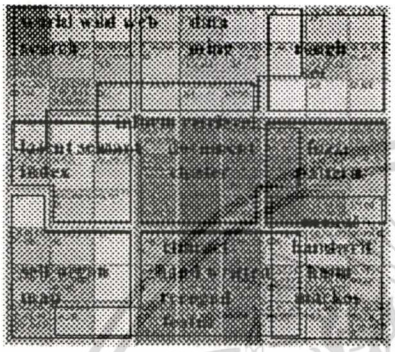
where f_{ij} is normalized frequency words. $freq_{ij}$ is frequency of term k_i in document d_j, N is the total number of documents and n_i is the number of document appeared in term k_i

Thus , the documents are represented in term of $\vec{d}_j = \{w_{1,j}, w_{2,j}, w_{3,j}, ..., w_{t,j}\}$ where $w_{i,j} \geq 0$.

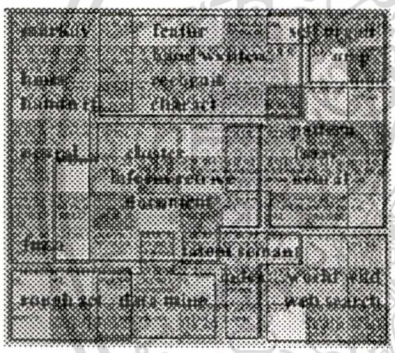
4.2 Document clustering by HS-SOM

From the experiment of HS-SOM in document clustering, we design first layer size in 9x9, t=3. For the second layer, we design two size in 3x3 and 4x4 as in Fig 4 a) and 4 b)

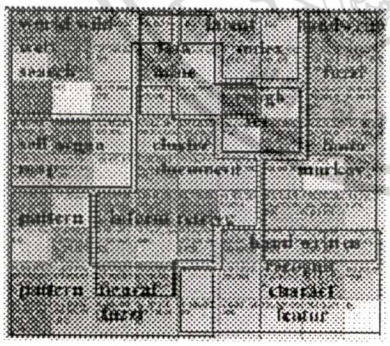
In learning process, we use number of iteration T=5000, the initial learning rate $\alpha(0)=0.1$ and the initial radius of neighborhood nodes $\sigma(0)=5$. For weight vector we random the initial value between 0-0.1



a.) The 3x3 second layer document clustering.



b.) The 4x4 second layer document clustering



c.) The conventional SOM document clustering by KOHONEN

Fig 4. The order maps of HS-SOM and conventional SOM

It is necessary to consider the data characteristic when we design of HS-SOM . If the domain of documents for clustering are not related to each other, the non-overlapping map is used. For example, Fig 4 a) has show the domain of "inform retrive" is clearly not related to the domain of "hand written". For the documents that their domains are unclear, we propose the overlapping map in second layer.

The advantages of HS-SOM for document clustering, first, from Fig 5 a) and 5 b) show that it is easier to browse document in HS-SOM than in the conventional SOM because we can browse the group of document in second layer to guide the group of documents in first layer.

a.) 4×4 Second layer b.) 3×3 Second layer

Fig 5. The sample of second layer in different size

From Fig 5 a), some nodes have two domains so there are overlapping data, for example in lower left node has two domain of "rough set" and "fuzzi set". But for Fig 5 b) the upper right has domain of "rough set" separated from "fuzzi set".

Second, as a result, we can reduce time for finding winning node, the speed of traning in HS-SOM is better than conventional SOM. The winning node from HS-SOM is the winning node of sub-domain which may not be a winning node of the whole domain of document. Considerately, the winning node is the significant node for the associated documents that will be the neighborhood nodes. As a consequence, we can group nodes and use the centroid to find the represent node. We will choose the shortest centroid to find winning node in that group.

Fig 6 is representing the time taken to train the learning of different SOM. The iteration T is 5000. The graph show, that HS-SOM 4x4 and 3x3 take less time, depend on the pattern of grouping, than conventinal SOM about 50%.

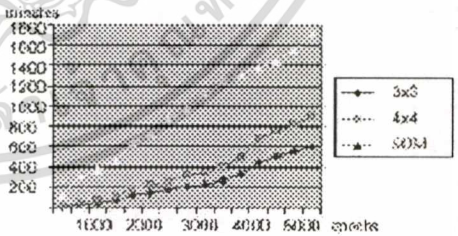


Fig 6. Show time taking to train HS-SOM 3x3, 4x4, and conventional SOM

The measurment of efficient for SOM use the entropy to indicate the document clustering. If there is one domain in the cluster, its entropy is zero. If there are many domains in the cluster, its entropy is going to be high as equation (11)

$$E_j = -\sum_i p_{ij} \log(p_{ij}) \quad (11)$$

The summation of the entropy is in equation (12)

$$E_w = \sum_{j=1}^{m} \frac{n_j \times E_j}{n} \quad (12)$$

where p_{ij} is the probability of the members in node j belong to group i. n_j is the number of document in node j, m is the number of clusters, that is the number of nodes and n is the number of document.

Table 1. The entropy of the different clustering.

Type	Summation of Entropy
Conventional SOM	0.295
HS-SOM 3x3	0.282
HS-SOM 4x4	0.315

From Table 1, the entropy of three SOM is similar. The entropy of HS-SOM 3x3 is less than HS-SOM 4x4 that mean the domain of sample documents for our experiment is almost not relate. The implicit documents are located at the overlapping area between groups.

We created the interface to browse the relevant documents. We assigned index terms for each node and compared with keyword. In Fig 7 we searched by using keyword "self organizing map". Before searching, the keyword was taken to find its stem words and displayed the most relevant node. We can move the arrows to browse the neighborhood node for related document.

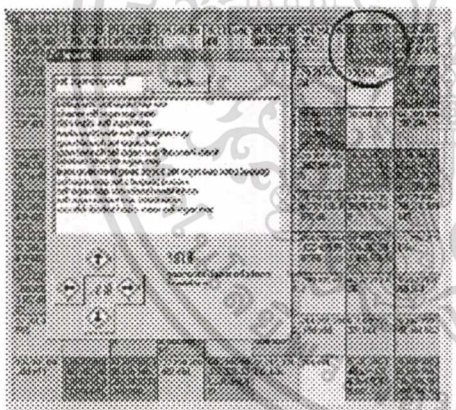


Fig 7. The interface to browse the relevant documents in HS-SOM4x4

5. CONCLUSION AND FUTURE

Our research focus on technique for reducing computing time in SOM by finding the winning node in the original map only in nodes that were represented by the winning node from the new map. We can increase the layer if there are a large number of data. Our experiments show that HS-SOM outperforms conventional SOM by 30%-50%. One of special characteristic of HS-SOM is the ability to initialize the direction the user's browsing by using the upper layer map. For this approch we can reduce time for browsing.

We can apply the HS-SOM to various applications such as image clustering, signal classification and pattern recognition.

REFERENCES

[1] A. Rauber, D. Merkl, and M. Dittenbach. "The Growing Hierarchical Self-Organizing Map: Exploratory Analysis of High-Dimensional Data", *IEEE Transactions on Neural Networks*, Vol. 13, No 6, pp 1331-1341, November 2002

[2] A.Rauber, D. Merkl. "The SOMLib Digital Library System," *Proceedings of the 3rd Europ. Conf. on Research and Advanced Technology for Digital Libraries (ECDL'99)*, Paris, France, September 22. - 24. 1999, Springer, 1999.

[3] Helge Ritter, Thomas Martinetz, Klaus Schulten, *Neural computation and self-organizing maps : an introduction*, Impring Massachusetts : Addison-Wesley, 1992.

[4] Kohonen, T. (1998). "Self-organization of very large document collections: State of the art." *ICANN98*, pages 65-74. Springer, London.

[5] Lagus, K., Honkela, T., Kaski, S., and Kohonen, T. "WEBSOM for textual data mining," *Artificial Intelligence Review*, volume 13, pages 345-364, Kluwer Academic Publishers, 1999

[6] Merkl and A. Rauber, "Document Classification with Unsupervised Neural Networks", *Soft Computing in Information Retrieval*,pp. 102 - 121, 2000.

[7] Qing Ma, Min Zhang, Ming Zhou. "Self-Organization of Chinese Semantic Maps Using TFIDF Term Weighting," *The Second Workshop on Natural Language Processing and Neural Networks*, Tokyo, Japan, November, 2001.

[8] R.Baeza-Yates B. Ribeiro-Neto, *Modern Information Retrieval*. ACM- Press, Addison-Wesley, 1999.

[9] Xia Lin, Dagobert Soergal, Gary Marchionini. "A Self-Organizing Semantic Map," *SIGIR*,1991.

ประวัติผู้เขียน

นายพรเทพ โรจนวสุ เกิดเมื่อวันที่ 19 พฤศจิกายน 2521 ที่จังหวัดราชบุรี สำเร็จการศึกษาชั้นอนุบาลและประถมศึกษาจากโรงเรียนอัสสัมชัญลำปาง ในปีการศึกษา 2533 ชั้นมัธยมศึกษาตอนต้นจากโรงเรียนบุญวาทย์วิทยาลัยจังหวัดลำปาง ในปีการศึกษา 2536 ชั้นมัธยมศึกษาตอนปลายจากศูนย์การศึกษานอกโรงเรียนจังหวัดลำปาง ในปีการศึกษา 2538 และปริญญาตรีวิศวกรรมศาสตร์บัณฑิต (วิศวกรรมคอมพิวเตอร์) จากมหาวิทยาลัยเชียงใหม่ ในปีการศึกษา 2542

ในปี พ.ศ. 2542-2544 เป็นลูกจ้างชั่วคราวตำแหน่งอาจารย์ สังกัดภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่

ปัจจุบันรับทุนศึกษาต่อระดับปริญญาโทสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง จากมหาวิทยาลัยนเรศวร วิทยาเขตสารสนเทศจังหวัดพะเยา