การจดจำอักขระภาษาไทยตัวพิมพ์โดยโครงข่ายประสาทเทียมแบบเซลฟ์ออร์แกนไนซิ่งแมปซ์

PRINTED THAI CHARACTER RECOGNITION BY SELF-ORGANIZING MAPS NETWORK

T028842

นางสาวอัญชลี วานิชทวีวัฒน์

MISS ANCHALEE WANICHTHAWEEWAT

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2540

ISBN 974-621-785-2

ลิขสิทธิ์ของบัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เลขหมู่.....
เลขทะเบียน.....28842
วัน, เดือน, ปี.....7 พ.ย. 2540

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRINTED THAI CHARACTER RECOGNITION BY SELF-ORGANIZING MAPS NETWORK

ANCHALEE WANICHTHAWEEWAT

THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE
MASTER OF SCIENCE (COMPUTER SCIENCE
AND INFORMATION TECHNOLOGY)
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUE OF TECHNOLOGY LADKRABANG
1997

ISBN 974-621-785-2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	การจดจำอักขระภาษาไทยตัวพิมพ์โดยโครงข่ายประสาทเทียมแบบเซลท์ออร์แกนไนซิ่งแมบซ์
นักศึกษา	นางสาวอัญชลี วานิชทวีวัฒน์
อาจารย์ผู้ควบคุมวิทยานิพนธ์	ดร.บุญธีร์ เครือตราชู
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการ-คอมพิวเตอร์และเทคโนโลยีสารสนเทศ
ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์
	คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ.	2540

บทคัดย่อ

งานวิจัยนี้ทำการจดจำตัวอักษรไทยโดยใช้โครงข่ายประสาทเทียมแบบ Self-Organizing Maps (SOM) และ Adaptive SOM (ASOM) ที่มีความยืดหยุ่นกว่าในการปรับโครงสร้างและขนาดโครงข่าย พร้อมทั้งแสดงการปรับเปลี่ยนอัลกอริทึมในการฝึกASOMเพื่อให้เหมาะสมในการจดจำตัวอักษรไทย นอกจากนี้ก็จะรายงานการเปรียบเทียบระหว่าง SOM และ ASOM ในแง่ของเวลาที่ใช้ในการฝึก จำนวนโหนดที่ใช้ และประสิทธิภาพในการจดจำ

จากการฝึกด้วยฟอนต์อังศนายูพีซี ขนาด 12 จุดต่อนิ้ว 75 แบบ จำนวน1500 ตัว ด้วยแบบจำลอง SOMจะต้องเสียเวลานานในการหาโครงข่ายที่เหมาะสม ส่วน ASOM จะใช้เวลาฝึกน้อยกว่าประมาณ 6 ชั่วโมง (บนเครื่อง hp 9000/827) 318 รอบ ใช้ 218 โหนด ประสิทธิภาพในการจดจำจากการทดสอบด้วยตัวอักษร 15000 ตัวที่เป็นตัวอักษรที่คมชัดและตัวอักษรที่มีสัญญาณรบกวนมีค่าเฉลี่ยเท่ากับ 99.77 , 96.87 เปอร์เซ็นต์ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Printed Thai Character Recognition by using Self-Organizing Maps Network
Student	Miss Anchalee Wanichthaweewat
Thesis Advisor	Dr.Boontee Kruatrachue
Level of Study	Master of Science program in Computer Science and Information Technology
Department	Mathematics and Computer Science Faculty of Science King Mongkut's Institute of Technology Ladkrabang
Year	1997

ABSTRACT

This research used Self-Organizing Maps neural network (SOM) and Adaptive SOM (ASOM) to recognize the printed Thai character. ASOM is more flexible than SOM in term of neural structure (node and connection). ASOM's training algorithm are modified to use in Thai character recognition. Furthermore, SOM and ASOM are presented in comparision with learning time, number of neural nodes and recognizing rate and efficiency.

The number of character use in training network are 1500 12-point Angsana UPC font which have 75 Thai character patterns. The SOM takes more training time to find the suitable network. However, the ASOM takes about 6 hours (on computer model, hp 9000/827), 381 cycles and 218 nodes. For testing, uses 15,000 characters which are unnoise characters and noise characters, the average percents of recognition are 99.77 and 96.87, respectively.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ ครู อาจารย์ ที่เคยอบรมสั่งสอนเป็นอย่างสูงที่ให้ความช่วยเหลือและสนับสนุนด้านการศึกษามาโดยตลอด

ขอกราบขอบพระคุณเป็นอย่างสูงในความกรุณาของ ดร. บุญธีร์ เครือตราชู อาจารย์ที่ปรึกษาวิทยานิพนธ์ ผู้ซึ่งทุ่มเทแรงกายแรงใจในการควบคุมงานวิจัย ให้คำแนะนำในการแก้ปัญหาและข้อบกพร่องต่าง ๆ เป็นอย่างดียิ่ง

ขอขอบพระคุณคณะกรรมการสอบวิทยานิพนธ์ทุกท่าน ที่ได้ช่วยกรุณาแก้ไขวิทยานิพนธ์ฉบับนี้ให้สมบูรณ์ยิ่งขึ้น

ขอขอบคุณ คุณทวี เปรมรัตนชัย ที่ช่วยเตรียมข้อมูล คุณอภิรักษ์ จิรายุสกุล คุณเกษมสันต์ คูภานุมาต ที่ช่วยดีบักโปรแกรม พี่กฤษฎา บุศรา ที่ช่วยด้านอุปกรณ์การพิมพ์ และเพื่อน ๆ ที่ให้ความช่วยเหลือด้านต่าง ๆ และเป็นกำลังใจที่ดีตลอดมา

ขอขอบคุณ พี่บุญช่วย ชาติทอง และเจ้าหน้าที่คณะเทคโนโลยีสารสนเทศทุกท่าน ที่ช่วยประสานงานและให้ความสะดวกในทุก ๆ ด้าน ตลอดระยะเวลาที่ผู้เขียนศึกษา ณ. สถาบันแห่งนี้

ขอขอบคุณมูลนิธิเพื่อการศึกษาคอมพิวเตอร์และการสื่อสารที่ให้ทุนสนับสนุนการศึกษา และขอขอบคุณทุนอุดหนุนการทำวิทยานิพนธ์จากบัณฑิตวิทยาลัย

อัญชลี วานิชทวีวัฒน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ.....IV

สารบัญตาราง.....VII

สารบัญภาพ.....XI

อภิธานศัพท์.....XIV

บทที่

1. บทนำ.....1

ความเป็นมาของงานวิจัย.....1

ผลงานวิจัยที่ผ่านมา.....2

แนวทางของวิทยานิพนธ์.....4

วัตถุประสงค์ของงานวิจัย.....4

โครงสร้างของวิทยานิพนธ์.....5

2. โครงข่ายประสาทเทียมพื้นฐาน.....6

ทฤษฎีเซลประสาทเบื้องต้น.....7

โครงสร้างเซลประสาทเทียมพื้นฐาน.....8

การฝึกโครงข่ายในแบบจำลองยุคแรก.....9

การแบ่งแยกเชิงเส้น.....10

ความสามารถในการแยกแยะเชิงเส้น.....12

การแบ่งแยกไม่เชิงเส้น.....15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

บทที่

หน้า

ฟังก์ชั่นโพลิโนเมียล....15

Madaline I....17

The Minimum Disturbance Principle....19

วิธีการฝึกโครงข่ายประสาทเทียม....19

3. โครงข่ายประสาทเทียมแบบ Adaptive Self-Organizing Maps (ASOM)....24

สถาปัตยกรรม SOM....26

สรุปอัลกอริทึมที่ 1 ที่ใช้สอน SOM....31

4. โครงข่ายประสาทเทียมแบบ Adaptive Self-Organizing Maps (ASOM)....32

สถาปัตยกรรม Adaptive Self-Organizing Maps (ASOM)....32

การฝึกโครงข่ายแบบ ASOM....36

อัลกอริทึมที่ 2 ใช้ฝึกสอน ASOM....42

5. การทดลอง SOM....43

จุดประสงค์การทดลอง....43

ข้อมูลเข้าที่ใช้ในการฝึกโครงข่าย....43

การทดลอง ผลการทดลอง และการแก้ปัญหา....45

สรุปปัญหาการฝึกโครงข่ายโดยใช้ SOM....67

6. การทดลอง ASOM....69

ชุดตัวอักษรที่ใช้ฝึก....69

การทดลอง....70

ผลการทดลอง....71

ปัญหา การแก้ปัญหา และการปรับเปลี่ยนขั้นตอนการฝึก ASOM....75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

บทที่

หน้า

อัลกอริทึมที่ 3 ใช้ฝึก ASOM.....83

อัลกอริทึมที่ 4 ใช้ฝึก ASOM.....84

7. สรุปผลการทดลอง.....96

แนวทางการพัฒนาต่อไปในอนาคต.....99

บรรณานุกรม.....100

ภาคผนวก.....101

บทความวิชาการ.....102

ประวัติผู้เขียน.....117

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่

1. แสดงฟังก์ชั่นตรรกที่เป็นไปได้ของ Adaline 1 เซลซึ่งมีข้อมูลแบบไบนารี่ โดยมีข้อมูลเข้า 2 ค่า และข้อมูลออก 1 ค่า 12
2. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 10 x 10 โหนด ในการฝึกรอบที่ 1 โดยค่า $\alpha_s = 0.6$ $\alpha_e = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว.....45
3. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 10 x 10 โหนด ในการฝึกรอบที่ 20 โดยค่า $\alpha_s = 0.6$ $\alpha_e = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว.....46
4. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 10 x 10 โหนด ในการฝึกรอบที่ 60 โดยค่า $\alpha_s = 0.6$ $\alpha_e = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว 46
5. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 15 x 15 โหนด ในการฝึกรอบที่ 60 โดยค่า $\alpha_s = 0.6$ $\alpha_e = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว 47
6. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 20 x 20 โหนด ในการฝึกรอบที่ 60 โดยค่า $\alpha_s = 0.6$ $\alpha_e = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว.....48
7. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 25 x 25 โหนด ในการฝึกรอบที่ 60 โดยค่า $\alpha_s = 0.6$ $\alpha_e = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว 49

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง (ต่อ)

หน้า

ตารางที่

8. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 25x25 โหนด
ในการฝึกรอบที่ 60 โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 5$
ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0 ตัวอักษร 75 แบบรวม 1500ตัว 50

9. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 10 x 10 โหนด
ในการฝึกรอบที่ 60 โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 5$
ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.4 - 0.6 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว 52

10. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 15 x 15 โหนด
ในการฝึกรอบที่ 60 โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 5$
ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.4 - 0.6 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว.....53

11. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 20x20 โหนด
ในการฝึกรอบที่ 60 โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 5$
ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.4 - 0.6 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว.....54

12. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 25 x 25 โหนด
ในการฝึกรอบที่ 60 โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 5$
ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.4 - 0.6 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว.....55

13. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 10 x 10 โหนด
ในการฝึกรอบที่ 80 โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 7$
ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว.....56

14. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 15 x 15 โหนด
ในการฝึกรอบที่ 80 โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 7$
ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว.....57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง (ต่อ)

หน้า

ตารางที่

15. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 20x20 โหนด
ในการฝึกรอบที่ 80 โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 7$
ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว.....58
16. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 20x20 โหนด
ในการฝึกรอบที่ 80 โดยค่า $\alpha_s = 0.9$ $\alpha_c = 0.001$ $N_c = 5$
ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว 59
17. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 20x20 โหนด
ในการฝึกรอบที่ 54 โดยค่า $\alpha_s = 0.3$ $\alpha_c = 0.001$ $N_c = 5$
ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว.....60
18. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 20x20 โหนด
ในการฝึกรอบที่ 80 โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ และ α จะลดลงทีละ 10
เปอร์เซ็นต์ตามลำดับความห่างจากโหนดชนะ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่ม
ระหว่าง 0.0 - 1.0 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว.....62
19. แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 20x20 โหนด
ในการฝึกรอบที่ 80 โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ และ α จะลดลงทีละ 20
เปอร์เซ็นต์ตามลำดับความห่างจากโหนดชนะ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่ม
ระหว่าง 0.0 - 1.0 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว 63
20. แสดงการแมปตัวอักษรของโครงข่ายแบบ SOM ขนาด 20x20 โหนด
ในการฝึกรอบที่ 60 โดยกำหนดให้ขอบเขตโหนดข้างเคียงมีการทดค่าที่เกิน
ไปด้านตรงข้าม ค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่ม
ระหว่าง 0.0 - 1.0 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว.....65

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง (ต่อ)

หน้า

ตารางที่

21. แสดงประสิทธิภาพการวิเคราะห์ตัวอักษร 15,000 ตัว.....86

22. แสดงตัวอักษรที่จดจำผิดโดยทดสอบจากตัวอักษร 75 แบบ แบบละ 200 ตัว ที่สะแกนจากกระดาษถ่ายเอกสารจากกระดาษที่พิมพ์ด้วยเครื่องพิมพ์เลเซอร์ ความสว่าง (brightness) 100 %.....92

23. แสดงตัวอักษรที่จดจำผิดโดยทดสอบจากตัวอักษร 75 แบบ แบบละ 200 ตัว ที่สะแกนจากกระดาษถ่ายเอกสารจากกระดาษที่พิมพ์ด้วยเครื่องพิมพ์เลเซอร์ ความสว่าง (brightness) 50 %.....92

24. แสดงตัวอักษรที่จดจำผิดโดยทดสอบจากตัวอักษร 75 แบบ แบบละ 200 ตัว ที่สะแกนจากกระดาษที่พิมพ์จากเครื่องพิมพ์เลเซอร์ ความสว่าง (brightness) 100 %.....93

25. แสดงตัวอักษรที่จดจำผิด โดยทดสอบจากตัวอักษร 75 แบบ แบบละ 200 ตัว ที่สะแกนจากกระดาษที่พิมพ์จากเครื่องพิมพ์เลเซอร์ ความสว่าง (brightness) 50 %.....93

26. แสดงการเปรียบเทียบจำนวนตัวอักษรแต่ละแบบที่โครงข่ายจดจำผิด เมื่อทดสอบตัวอักษรจากกระดาษซึ่งถ่ายเอกสารจากกระดาษที่พิมพ์จาก เครื่องพิมพ์เลเซอร์ ความสว่าง 100 และ 50 % กระดาษที่พิมพ์จาก เครื่องพิมพ์เลเซอร์ ความสว่าง 100 และ 50 %.....94

27. แสดงการเปรียบเทียบประสิทธิภาพในการวิเคราะห์ตัวอักษรโดยโครงข่ายแบบ Backpropagation, Counterpropagation (CPN) และ Adaptive Self-Organizing Maps (ASOM).....98

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

หน้า

1. แสดงการทำงานของเครื่องอ่านตัวอักษรด้วยแสง....2
2. แสดงการเชื่อมต่อกันของเซลประสาทมนุษย์....8
3. แสดงโครงสร้างของเซลประสาทเทียม 1 เซล....9
4. แสดง Adaline ที่มีข้อมูลเข้า (input) ที่เป็นไบนารี่จำนวน 2 ค่า....11
5. แสดงการแบ่งฟังก์ชั่นตรรกในตารางที่ 1 ที่บางฟังก์ชั่นแบ่งไม่ได้ด้วยเส้นตรงเส้นเดียว....13
6. กราฟแสดงความสัมพันธ์ระหว่างโอกาสในการที่ Adaline จะแยกแยะข้อมูลได้ กับอัตราส่วน $\frac{N_p}{N_w}$ 15
7. แสดง Adaline 1 เซล ที่มีการแปลงข้อมูลเข้าเชิงเส้นเป็นข้อมูลเข้าไม่เชิงเส้น....16
8. แสดง Madaline ที่ประกอบด้วย Adaline 2 หน่วย....17
9. แสดงเส้นแบ่งขอบเขตของ Madaline ในภาพที่ 8 18
10. แสดงตัวอย่าง Adaline ที่ใช้กับฟังก์ชั่นตรรก AND, OR MAJ....18
11. แสดงภาพการปรับน้ำหนักในลักษณะสามมิติ....21
12. แผนภาพ α-Perceptron ของโรเซนบลอต 22
13. แสดง adaptive threshold element ของ α-Perceptron 22
14. แสดงสมองส่วนเปลือกนอกในภาพด้านข้างที่หนาประมาณ 2-3 มิลลิเมตร โดยพื้นที่แต่ละส่วนของสมองจะมีหน้าที่ควบคุมการทำงานต่าง ๆ กัน....25
15. แสดงสถาปัตยกรรมของ SOM....26
16. แสดงสัญญาณการกระตุ้นแบบ Maxican Hat Function....28
17. แสดงการปรับน้ำหนักในภาพเวคเตอร์น้ำหนักและเวคเตอร์ข้อมูลเข้าขนาด 1 หน่วย....29
18. แสดงขอบเขตโหนดข้างเคียงในลักษณะต่าง ๆ....29
19. แสดงค่าน้ำหนักของโครงข่ายหลังการฝึกซึ่งเป็นน้ำหนักเฉลี่ย ของกลุ่มข้อมูลตัวอย่างแต่ละกลุ่มที่ใช้ฝึก....30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

หน้า

20. แสดงการเชื่อมต่อระหว่างโหนดในโครงข่ายที่มีโครงสร้างซิมเพลคเป็น 1, 2 และ 3 มิติ.....34
21. แสดงโครงสร้างพื้นฐานในโครงข่าย 2 มิติ.....34
22. แสดงเส้นแบ่งขอบเขตในแต่ละซิมเพลค.....35
23. แสดงภาพหลายเหลี่ยมเส้นประที่เป็นพื้นที่โวโรนอยของโหนด
ที่ภาพหลายเหลี่ยมนั้นล้อมรอบอยู่.....36
24. แสดงการเพิ่มโหนดในโครงข่ายสองมิติเมื่อฝึกไประยะเวลาหนึ่ง (λ).....37
25. แสดงการปรับโครงสร้างโครงข่ายหลังการเพิ่มโหนดใหม่.....38
26. แสดงการเปลี่ยนแปลงความถี่การตก และพื้นที่โวโรนอยด์เมื่อเพิ่มโหนดให้โครงข่าย.....39
27. แสดงการปรับโครงสร้างโครงข่ายประสาทหลังการลบโหนด.....41
28. แสดงตัวอย่างการป้อนข้อมูลภาพตัวอักษรเข้าฝึกในโครงข่ายโดย.....44
29. แสดงขอบเขตโหนดข้างเคียงที่ให้ถือขอบเขตโครงข่ายเป็นที่สิ้นสุดขอบเขตโหนดข้างเคียง
และขอบเขตโหนดข้างเคียงที่มีการทดค่าที่เกินไปยังด้านตรงข้าม (มีการ roll back).....64
30. แสดงทิศทางการปรับค่าน้ำหนักเมื่อตัวอักษรตกในโหนดที่ถูกต้องและไม่ถูกต้อง.....67
31. แสดงตัวอักษรไทยที่อยู่ในระดับกลาง.....69
32. แสดงตัวอักษรไทยที่อยู่ในระดับบน.....70
33. แสดงตัวอักษรไทยที่อยู่ในระดับล่าง.....70
34. แสดงการแมปตัวอักษรที่สามารถแยกแยะได้ดีขึ้นตามจำนวนรอบที่ใช้ฝึก.....72
35. แสดงการเพิ่มโหนดให้โครงข่าย.....75
36. แสดงขั้นตอนการลบโหนดที่ไม่มีอักษรใดตกออกแล้วมีผลทำให้
โหนดในโครงข่ายทั้งหมดถูกลบออกด้วย.....77
37. แสดงการพิจารณาโหนดที่ควรลบและไม่ควรลบออกจากโครงข่าย.....78
38. แสดงปัญหาการกำหนดสัญญาณให้โหนดใหม่ที่ได้จากการคาดคะเน.....80

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

หน้า

39. แสดงการเพิ่มโหนดใหม่ที่มีผลทำให้โหนดเก่าถูกผลักออกให้ห่างจาก.....81
40. แสดงการเพิ่มโหนดทีละ 2 โหนดเมื่อฝึกครบ 1 รอบ.....82
41. แสดงการแมปของโครงข่ายเมื่อสิ้นสุดการฝึกและลบโหนดที่ไม่มีอักษรใดตกออกแล้ว.....90
42. แสดงตัวอย่างภาพตัวอักษรที่ ASOM จดจำได้ เรียงตามลำดับดังนี้ ค,ฅ,ค,ค,ห,โ.....95
43. แสดงตัวอย่างภาพตัวอักษรที่ ASOM จดจำผิด เรียงตามลำดับดังนี้ ค,ค,ฅ,ใ,ห,โ.....95

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อภิธานศัพท์

Artificial Neural Network โครงข่ายประสาทเทียม ที่จำลองให้คอมพิวเตอร์ทำงานเลียนแบบสมองของสิ่งมีชีวิต

ASOM ย่อมาจาก Adaptive Self-Organizing Maps เป็นแบบจำลองโครงข่ายประสาทเทียมแบบหนึ่งที่เป็นแบบหลายชั้นและเรียนรู้แบบไม่มีผู้สอน ซึ่งพัฒนามาจาก SOM

axon ส่วนของเซลประสาทมนุษย์ที่ทำหน้าที่ส่งสัญญาณการกระตุ้นไปให้เซลอื่น

Backpropagation เป็นแบบจำลองโครงข่ายประสาทเทียมแบบหนึ่งที่เป็นแบบหลายชั้นและเรียนรู้แบบมีผู้สอน

Capacity ปริมาณข้อมูลที่ใช้ฝึกซึ่งโครงข่ายประสาทเทียมสามารถจดจำได้สูงสุด

competitive node โหนดในชั้นคอมเพททิทีฟซึ่งเป็นโหนดที่มีการประมวลผลสัญญาณการกระตุ้น

Counterpropagation เป็นแบบจำลองโครงข่ายประสาทเทียมแบบหนึ่งที่เป็นแบบหลายชั้นและเรียนรู้แบบมีผู้สอน

dendrite ส่วนของเซลประสาทมนุษย์ที่ทำหน้าที่รับสัญญาณการกระตุ้นจากเซลอื่น

Generalization คุณสมบัติของการที่โครงข่ายประสาทเทียมสามารถจดจำข้อมูลที่ไม่ได้ใช้ในการฝึกถูกต้องเป็นส่วนมาก ซึ่งข้อมูลเหล่านี้มีความหลากหลายสูง

Input ข้อมูลที่ป้อนให้โครงข่าย

input node โหนดที่รับสัญญาณข้อมูลเข้ามายังโครงข่ายประสาทเทียม

mapping การแสดงความสัมพันธ์ของข้อมูลที่มีหลาย ๆ มิติ ลงมาในมิติที่ต่ำกว่า

Minimum disturbance principle หลักการการปรับค่าน้ำหนักของโครงข่าย ขณะที่โครงข่ายเรียนรู้ที่จะจดจำ ค่าน้ำหนักเป็นค่าที่เก็บความรู้ของโครงข่าย ในการปรับน้ำหนักแต่ละครั้งจึงต้องพยายามรักษาความรู้ต่าง ๆที่โครงข่ายสามารถจดจำได้ก่อนหน้านี้แล้วไว้ โดยการพยายามให้มีการเปลี่ยนแปลงน้ำหนักที่เก็บความรู้เหล่านี้ได้แล้วให้น้อยที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อภิธานศัพท์ (ต่อ)

neighbor node โหนดที่ข้างเคียงโหนดที่ถูกเลือกเก็บความรู้

normalise การทำให้ข้อมูลอยู่ในมาตรฐานที่สามารถนำมาคำนวณโดยคอมพิวเตอร์ได้ข้อมูลเข้าโดยไม่เกิดการ overflow

Output ข้อมูลที่ออกจากโครงข่าย หลังจากรับข้อมูลเข้ามาประมวลผลแล้ว

Perceptron เป็นแบบจำลองโครงข่ายประสาทเทียมแบบหนึ่งที่กำเนิดในยุคแรก ๆ

SOM ย่อมาจาก Self-Organizing Maps เป็นแบบจำลองโครงข่ายประสาทเทียมแบบหนึ่งที่เป็นแบบหลายชั้นและเรียนรู้แบบไม่มีผู้สอน

Supervised learning จะสอนโครงข่ายด้วยการป้อนแบบข้อมูลเข้า และแบบข้อมูลออกที่ต้องการให้โครงข่ายเรียนรู้ตามกฎการเรียนรู้ในแต่ละแบบจำลองที่จะมีการวัดค่าผิดพลาด โดยคำนวณจากผลต่างระหว่างค่าข้อมูลออกจากโครงข่ายและค่าข้อมูลออกที่ต้องการ แล้วนำค่าผิดพลาดนี้ไปใช้ในการคำนวณย้อยกลับเพื่อปรับให้ได้ค่าข้อมูลออกตรงตามค่าข้อมูลออกที่ต้องการ

synepse ส่วนเชื่อมต่อระหว่าง axon และ dendrite ที่มีการรับและส่งกระแสประสาท

Threshold ค่าระดับสัญญาณการกระตุ้น ณ จุดวิกฤต โดยหากได้รับสัญญาณการกระตุ้นมากกว่าค่านี้ เซลก็จะถูกกระตุ้น แต่ถ้าได้รับสัญญาณต่ำกว่าค่านี้เซลก็จะไม่ถูกกระตุ้น ซึ่งหมายถึงทั้งในเซลประสาทมนุษย์และเซลประสาทเทียม

Threshold device สิ่งที่ใช้ในการแปลงสัญญาณการกระตุ้นแบบต่อเนื่องเป็นแบบไบนารี่ในโครงข่ายประสาทเทียม

Unsupervised learning จะสอนโครงข่ายด้วยการป้อนแบบข้อมูลเข้าอย่างต่อเนื่องเพียงอย่างเดียว ไม่มีการส่งค่าใด ๆ ให้คำนวณย้อนกลับ การเรียนรู้ในลักษณะนี้เป็นลักษณะของการแบ่งกลุ่มตามชุดตัวอย่างแบบข้อมูลเข้าทั้งหมดที่ใช้ในการเรียนรู้ ที่ไม่สามารถคาดการณ์ได้ว่า ข้อมูลฝึกจะมีลักษณะใดบ้าง

winner node โหนดที่ถูกเลือกให้เก็บความรู้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

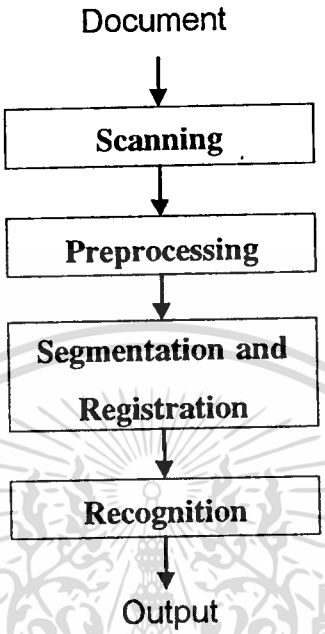
<u>ความเป็นมาของงานวิจัย</u>

ประเทศไทยในปัจจุบันมีการนำคอมพิวเตอร์ไปประยุกต์ใช้งานกันอย่างกว้างขวาง ซึ่งงานส่วนหนึ่งที่สำคัญไม่ว่าในสาขาใด ๆ ก็ตาม ก็คือ การจัดเก็บข้อมูลเพื่อนำไปประมวลผลให้ได้ข้อมูลที่มีประโยชน์ในการนำไปใช้ต่อไปในอนาคต ซึ่งข้อมูลต่าง ๆ เหล่านี้มักมีเป็นจำนวนมหาศาล แม้ว่า ปัจจุบันมีการพัฒนาเครื่องป้อนข้อมูลแบบใหม่ ๆ เช่น เครื่องป้อนข้อมูลเข้าจานแม่เหล็ก เครื่องป้อนข้อมูลเข้าเทปแม่เหล็ก เป็นต้น แต่เครื่องป้อนข้อมูลเหล่านี้ก็ต้องป้อนผ่านทางคีย์บอร์ด จึงทำให้ต้องใช้ทรัพยากรบุคคลจำนวนมากที่จะคีย์ข้อมูลผ่านแป้นพิมพ์ให้เครื่องคอมพิวเตอร์ เห็นว่าจะต้องสูญเสียทั้งเวลาและงบประมาณจำนวนมาก

เนื่องจากเทคโนโลยีได้ก้าวหน้าไปอย่างรวดเร็ว จึงทำให้เครื่องคอมพิวเตอร์มีประสิทธิภาพในการทำงานสูงขึ้น และมีอุปกรณ์สนับสนุนการทำงานอื่น ๆ มากมาย อุปกรณ์ชนิดหนึ่งก็คือ เครื่องอ่านตัวอักษรด้วยแสง (optical character reader) หลักการการทำงานของเครื่องจะอาศัยการฉายแสงลงบนแผ่นกระดาษแล้วนำปริมาณแสงที่สะท้อนกลับมาเปลี่ยนเป็นสัญญาณไฟฟ้า และสัญญาณไฟฟ้าจะถูกนำไปเปลี่ยนเป็นระหัสข้อมูลทางดิจิตอลแล้วนำไปเก็บไว้ในหน่วยความจำของเครื่องคอมพิวเตอร์ เพื่อส่งต่อให้คอมพิวเตอร์ทำการวิเคราะห์ต่อไปว่าตัวอักษรนั้นเป็นตัวอะไร ดังแสดงการทำงานของเครื่องอ่านตัวอักษรด้วยแสงเป็นแผนภาพในภาพที่ 1 การใช้เครื่องอ่านตัวอักษร (OCR) ดังกล่าวสามารถทดแทนการใช้แป้นพิมพ์ดีดได้เป็นอย่างดี เพราะช่วยประหยัดทั้งเวลาและต้นทุนในการจัดเก็บเอกสารได้อย่างมีประสิทธิภาพ เป็นอย่างมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 1



แสดงการทำงานของเครื่องอ่านตัวอักษรด้วยแสง

ในต่างประเทศมีการวิจัยระบบรู้จำตัวอักษรภาษาต่าง ๆ มากมาย แต่วิธีต่าง ๆ ที่มีการประยุกต์ใช้ในภาษาต่างประเทศก็จะเหมาะสมกับคุณลักษณะของภาษานั้น ๆ โดยเฉพาะ ยังไม่สามารถนำวิธีการเดียวกันนั้นมาใช้ในการจดจำอักษรภาษาไทยได้โดยตรงทันที จึงมีการศึกษาวิจัยให้คอมพิวเตอร์สามารถที่จะรับรู้และเข้าใจตัวอักษรภาษาไทยกันอย่างกว้างขวาง แต่ละวิธีในงานวิจัยที่ผ่าน ๆ มาให้ผลลัพธ์และประสิทธิภาพในการทำงานแตกต่างกัน ซึ่งงานวิจัยเหล่านี้ล้วนเป็นแนวทางในการพัฒนาระบบให้ดียิ่งขึ้นไปในอนาคต ในงานวิจัยด้านนี้ประสิทธิภาพในการวิเคราะห์ตัวอักษรให้ถูกต้องก็ถืองานที่สำคัญส่วนหนึ่ง งานวิจัยที่เสนอในวิทยานิพนธ์นี้จึงเน้นที่จะปรับปรุงและพัฒนาประสิทธิภาพให้คอมพิวเตอร์สามารถแยกแยะตัวอักษรให้ถูกต้องด้วยการนำโครงข่ายประสาทเทียม (neural network) มาประยุกต์ใช้ในการแก้ปัญหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลงานวิจัยที่ผ่านมา

1. ผลงานวิจัยของ รศ. ดร. ชม กิ้มปาน[10] เรื่อง การจดจำรูปแบบตัวอักษรพิมพ์ภาษาไทย โดยนำเสนอไว้ 2 วิธีคือ

การรู้จำแบบซ้อนทับ (Matching method) เป็นวิธีการที่ได้แนวคิดจากรูปร่างลักษณะของตัวอักษรภาษาไทยที่มีรูปร่างลักษณะการเขียนที่คล้าย ๆ กันอยู่หลายตัว ซึ่งสามารถนำมาจัดกลุ่มอย่างคร่าว ๆ ได้ ในส่วนของการรู้จำได้แบ่งการทำงานออกเป็น 2 ขั้นตอนใหญ่ ๆ คือ ขั้นตอนการแบ่งหยาบ ที่นำเทคนิคการทำเบลอมาใช้เพื่อที่จะกำจัดรายละเอียดปลีกย่อยที่เกินออกไปเสียก่อน แล้วนำทฤษฎีทางคณิตศาสตร์ชั้นสูงของการกระจายแบบการ์ฮูแนนโลบของไอเกนเวคเตอร์ที่มีค่าไอเกนสูงสุดมาคำนวณหารูปแบบมาตรฐานสำหรับตัวอักษรในแต่ละกลุ่มย่อย แล้วในขั้นตอนการแบ่งอย่างละเอียดจะเป็นการซ้อนทับตัวอักษรที่ต้องการการรู้จำแบบรูปแบบมาตรฐาน ส่วนผลของการรู้จำให้ผลลัพธ์เป็นที่น่าพอใจ ความถูกต้องสูงกว่า 90 เปอร์เซ็นต์

วิธีการวิเคราะห์โครงสร้าง (structural analysis method) วิธีการนี้ได้นำเอาคุณสมบัติทางโทโพโลยี่มาช่วยในการพิจารณา ภาพตัวอักษรที่ต้องการรู้จำจะถูกทำให้บาง แล้วจึงถูกแบ่งกลุ่มออกแบบหยาบด้วยฟังก์ชั่นการตัดสินใจเชิงเส้นโดยพิจารณาความกว้างและความสูงของตัวอักษร ในส่วนของการแบ่งละเอียดเป็นการคำนวณหาตัวเลขแสดงความต่อเนื่องของจุดโครงร่างตัวอักษร

2. ผลงานวิจัยของ นายประสาร ตังติสานนท์[11] เรื่อง การจดจำรูปแบบตัวอักษรลายมือภาษาไทยโดยวิธีแยกลักษณะเด่น จะพิจารณาโครงสร้างของรูปแบบตัวอักษรทั้งลายเส้นและพื้นเบื้องหลังของลายเส้นและลักษณะลายพื้นเบื้องหลังลายเส้นเป็นรหัสเบื้องต้น (initial feature extraction) ขั้นตอนที่สองเป็นการลดทอนรหัสเบื้องต้นที่ซ้ำซ้อน (unification) ขั้นตอนที่สามเป็นการรวมรหัสลายเส้นและลายพื้นเข้าด้วยกันเพื่อให้เป็นคุณสมบัติของรูปแบบตัวอักษร

3. ผลงานของ นายสุรพันธ์ เอื้อไพบูลย์[12] เรื่อง การจดจำตัวอักษรลายมือเขียนภาษาไทยโดยการพิจารณาหัวของตัวอักษร เป็นการเลือกพิจารณาส่วนหัวตัวอักษรภาษาไทยเพื่อวิเคราะห์ตัวอักษรด้วยเทคนิคพิเศษ 4 เทคนิค คือ 1. subhead region 2. feature code 3. head style และ 4. width per height ratio

4 . ผลงานวิจัยของ นายสุรสิทธิ์ ราตรี[13] เรื่อง การรู้จำตัวอักษรพิมพ์ภาษาไทยโดยวิธีการค้นหาลักษณะเด่นของตัวอักษร โดยการแยกพิจารณาโครงสร้างของตัวอักษรเป็นส่วน ๆ อย่างละเอียด แล้วนำคุณสมบัติต่าง ๆ มารวบรวมเป็นคุณลักษณะเด่นของอักษรแต่ละตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แนวทางของวิทยานิพนธ์

งานวิจัยในวิทยานิพนธ์นี้ได้เน้นวิจัยเฉพาะในส่วนของการวิเคราะห์ตัวอักษรภาษาไทย โดยการนำเอาโครงข่ายประสาทเทียมแบบจำลอง Self-Organizing Maps (SOM) และ Adaptive Self-Organizing Maps (ASOM) มาประยุกต์ใช้ในการจดจำตัวอักษรภาษาไทย จากการใช้แบบจำลอง SOM ตามทฤษฎีเดิมก็ประสบปัญหาบ้าง จึงได้เสนอแนวทางการแก้ไขปัญหาด้วยการปรับเปลี่ยนโครงสร้างของแบบจำลองที่แตกต่างไปจากแบบจำลอง SOM เดิมไปหลายส่วน เพื่อแก้ปัญหาดังกล่าวและเพื่อเพิ่มประสิทธิภาพในการจดจำให้ดีที่สุด

ปัจจุบันมีการพัฒนางานวิจัยด้านการจดจำตัวอักษรภาษาไทยไว้หลายวิธี แต่ละวิธีก็มีจุดเด่นและด้อยต่างกันเช่น วิธีหาคุณสมบัติทางโทโพโลยี (topology) ของตัวอักษร[10] หากตัวอักษรไม่ชัดเจนหรือมีลักษณะคล้ายกันจะทำให้การวิเคราะห์ผิดพลาดได้ง่ายและวิธี Backpropagation[15] ก็ต้องใช้เวลาฝึกนาน งานวิจัยนี้จึงใช้โครงข่ายประสาทเทียมแบบ Self-Organizing Maps (SOM) ซึ่งมีจุดเด่นที่ใช้เวลาฝึกน้อยกว่า แต่ประสิทธิภาพการจดจำใกล้เคียงกัน อย่างไรก็ตามเนื่องจาก SOM มีโครงสร้างและขนาดโครงข่ายคงที่ตลอดที่อาจไม่เหมาะสม จึงใช้แบบ Adaptive Self-Organizing Maps (ASOM) ที่เพิ่มและลดโหนดได้ในระหว่างการฝึก นอกจากนี้ยังได้ปรับอัลกอริทึม ASOMให้เหมาะสมกับการประยุกต์ใช้จดจำตัวอักษรไทยด้วย

วัตถุประสงค์ของงานวิจัย

1. เพื่อศึกษาการทำงานของโครงข่ายประสาทเทียมแบบ SOM, ASOM และนำมาประยุกต์ใช้ในการวิเคราะห์ตัวอักษรภาษาไทย
2. เพื่อศึกษาปัญหาในการวิเคราะห์ตัวอักษรภาษาไทยด้วย SOM, ASOM และแนวทางในการปรับปรุง เพื่อนำมาใช้งานได้อย่างมีประสิทธิภาพ
3. เพื่อเป็นแนวทางพัฒนาระบบ OCR ภาษาไทยต่อไปในอนาคต
4. เพื่อเปรียบเทียบประสิทธิภาพในการวิเคราะห์ตัวอักษรภาษาไทยด้วยวิธี SOM, ASOM และวิธีอื่น ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างของวิทยานิพนธ์

โครงสร้างของวิทยานิพนธ์ประกอบด้วย 6 ส่วน คือ

บทที่ 2 ทฤษฎีโครงข่ายประสาทเทียมเบื้องต้น กล่าวถึง แนวความคิดในการทำงานของสมองมนุษย์ที่เป็นต้นแบบในการนำมาพัฒนาโครงข่ายประสาทเทียม อธิบายถึงลักษณะการเรียนรู้และความสามารถในการจดจำของโครงข่ายประสาทเทียม ตัวอย่างแบบจำลองโครงข่ายประสาทเทียมอย่างง่าย และการนำไปประยุกต์ใช้งาน

บทที่ 3 ทฤษฎี SOM กล่าวถึง แนวความคิดในการทำงานของสมองมนุษย์ที่เป็นต้นแบบในการนำมาพัฒนาเป็นโครงข่ายประสาทเทียมแบบจำลอง SOM อธิบายทฤษฎีการทำงานของ SOM และอัลกอริทึมที่ใช้ในการฝึกสอนโครงข่าย SOM

บทที่ 4 กล่าวถึงทฤษฎี ASOM กล่าวถึง ทฤษฎีการทำงานของโครงข่ายประสาทเทียมแบบจำลอง ASOM และอัลกอริทึมที่ใช้ในการฝึกสอนโครงข่าย ASOM

บทที่ 5 การทดลอง SOM กล่าวถึง ผลการทดลองจากการใช้โครงข่าย SOM ในการวิเคราะห์ตัวอักษรภาษาไทย ปัญหา การปรับปรุงแก้ไขวิธีการสอนและขั้นตอนการสอนโครงข่าย SOM เพื่อแก้ปัญหาที่เกิดขึ้น ตลอดจนสรุปผลการแก้ปัญหา และสรุปอัลกอริทึมในการสอน SOM ที่ปรับปรุงแก้ไข

บทที่ 6 การทดลอง ASOM กล่าวถึง ผลการทดลองจากการใช้โครงข่าย ASOM ในการวิเคราะห์ตัวอักษรภาษาไทย ปัญหา การปรับปรุงแก้ไขวิธีการสอนและขั้นตอนการสอนโครงข่าย ASOM ตลอดจนสรุปผลการแก้ปัญหา และสรุปอัลกอริทึมในการสอน ASOM ที่ปรับปรุงแก้ไข

และบทที่ 7 สรุปผลการทดลอง จะสรุปการทดลองจากการใช้โครงข่ายประสาทเทียมแบบ SOM และแบบ ASOM ในการวิเคราะห์ตัวอักษรภาษาไทย เปรียบทียบประสิทธิภาพการทำงานของโครงข่ายประสาทเทียมแบบ Backpropagation , Counterpropagation และAdaptive Self-Organizing Maps จากการนำโครงข่ายทั้งสามแบบมาประยุกต์ใช้ในการวิเคราะห์ตัวอักษรภาษาไทยในแง่ของประสิทธิภาพในการวิเคราะห์ตัวอักษร เวลาที่ใช้ในการฝึกสอน จุดเด่นและจุดด้อยของโครงข่ายแต่ละแบบ และแนวทางในการพัฒนาต่อไปในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

โครงข่ายประสาทเทียมพื้นฐาน

บทนำ

คอมพิวเตอร์มีความสามารถในการนำมาใช้แก้ปัญหาหลายอย่าง ทั้งยังให้ผลลัพธ์ที่มีความน่าเชื่อถือความถูกต้องและมีความรวดเร็วกว่ามนุษย์ แต่ปัญหานั้น ๆ ต้องสามารถออกแบบขบวนการที่ชัดเจนในการดำเนินการกับข้อมูลเข้า (input) เพื่อจะสร้างข้อมูลออก (output) ให้ได้ตามต้องการ เช่น การแก้ปัญหาสมการอนุพันธ์ ซึ่งงานดังกล่าวมักเป็นงานที่มีการประมวลผลที่ซ้ำซ้อน (routine) มีขั้นตอนการประมวลผลที่ชัดเจน และเป็นงานที่ทำการคำนวณซ้ำ ๆ กันเป็นส่วนใหญ่ ในปัญหาที่ขบวนการมีความซับซ้อนและความคลุมเครือสูงจนไม่สามารถสรุปเป็นขั้นตอนที่ชัดเจนได้แล้ว มนุษย์หรือแม้แต่สัตว์บางชนิดก็ยังสามารถทำได้ดีกว่าคอมพิวเตอร์ เช่น การจดจำวัตถุต่าง ๆ ในที่ที่มีแสงสว่างมาก ในที่สลัว และการจดจำเสียงพูดต่างบุคคล ต่างระดับความสูงของเสียง ต่างความเร็วของการพูดและความดังของเสียง เนื่องจากสมองมนุษย์ประกอบด้วยเซลประสาทจำนวนหลายล้านเซลที่สามารถทำงานแบบขนานกันและมีขั้นตอนการประมวลผลที่มีความสลับซับซ้อนและการที่มนุษย์สามารถจดจำสิ่งต่าง ๆ ได้ ก็เกิดจากสมองมีการเรียนรู้ในสิ่งนั้น ๆ ก่อน ไม่เช่นนั้นแล้วสมองก็ไม่สามารถจดจำอะไรได้เลย แม้ปัจจุบันจะยังไม่สามารถสรุปขั้นตอนการเรียนรู้ของสมองได้อย่างชัดเจนนัก ก็อาจสรุปได้คร่าว ๆ ว่า เซลประสาทในสมองต่างประมวลผลเพื่อวิเคราะห์และจดจำลักษณะเฉพาะต่าง ๆ เช่น การจำว่าเป็นเสียงเด็ก เสียงผู้ใหญ่ เสียงผู้หญิง หรือเสียงผู้ชาย และเป็นเสียงของใคร และก็อาจลืมสิ่งต่าง ๆ ที่เคยจดจำได้ด้วย เนื่องจากสมองมีภาระกิจมากมายในการประมวลผลสิ่งต่าง ๆ ในแต่ละวัน แต่ก็สามารถที่จะเรียนรู้ใหม่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำงานสมองเพื่อให้คอมพิวเตอร์สามารถแก้ปัญหาที่ซับซ้อนและคลุมเครือได้เช่นเดียวกับสมอง และเรียกแบบจำลองนี้ว่า โครงข่ายประสาทเทียม (artificial neural network) เช่นเดียวกันกับสมองมนุษย์โครงข่ายประสาทเทียมจะถูกฝึกให้เรียนรู้จากตัวอย่างเพื่อจะได้มีความรู้ที่จะนำไปแก้ปัญหาต่อไป การฝึกโครงข่ายประสาทเทียมก็มีอยู่หลายวิธี ซึ่งหากแบ่งตามลักษณะการเรียนรู้จะแบ่งได้เป็น 2 ประเภท คือ

1. การเรียนรู้แบบมีผู้สอน (supervised learning) จะสอนโครงข่ายด้วยการป้อนแบบข้อมูลเข้า (input pattern) และ แบบข้อมูลออกที่ต้องการ (desired output pattern)ให้โครงข่ายเรียนรู้ตามกฎการเรียนรู้ในแต่ละแบบจำลองที่จะมีการวัดค่าผิดพลาด (error) โดยคำนวณจากผลต่างระหว่างค่าข้อมูลออกจากโครงข่าย (actual output) และค่าข้อมูลออกที่ต้องการ แล้วนำค่าผิดพลาดนี้ไปใช้ในการคำนวณย้อยกลับ (feedback) เพื่อปรับให้ได้ค่าข้อมูลออกตรงตามค่าข้อมูลออกที่ต้องการ ตัวอย่างแบบจำลองนี้เช่น Backpropagation

2. การเรียนรู้แบบไม่มีผู้สอน (unsupervised learning) จะสอนโครงข่ายด้วยการป้อนแบบข้อมูลเข้าอย่างต่อเนื่องเพียงอย่างเดียวไม่มีการส่งค่าใด ๆ ให้คำนวณย้อนกลับ (no feedback) การเรียนรู้ในลักษณะนี้เป็นลักษณะของการแบ่งกลุ่ม (categorization) ตามชุดตัวอย่างแบบข้อมูลเข้าทั้งหมดที่ใช้ในการเรียนรู้ ที่ไม่สามารถคาดการณ์ได้ว่า ข้อมูลฝึกจะมีลักษณะใดบ้าง ตัวอย่างแบบจำลองนี้เช่น Counterpropagation (CPN) และ Self-Organizing Maps (SOM)

แบบจำลองต่าง ๆ จะมีขั้นตอนการฝึกที่แตกต่างกัน ในบทนี้จะกล่าวถึงแบบจำลองโครงข่ายประสาทเทียมแบบง่าย ๆ ชนิดการเรียนรู้แบบมีผู้สอนซึ่งเป็นแบบจำลองที่ศึกษากันในยุคแรกของการศึกษาเรื่องโครงข่ายประสาทเทียม

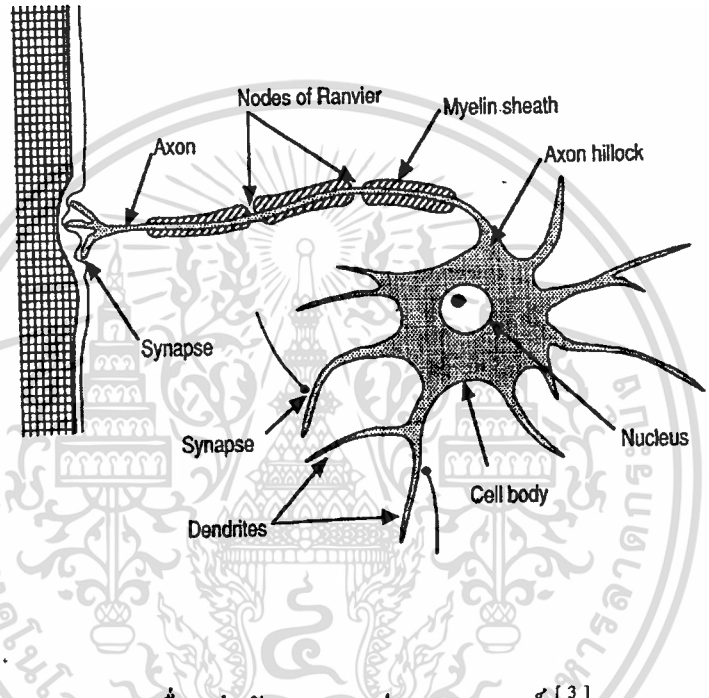
<u>ทฤษฎีเซลประสาทเบื้องต้น</u>

ระบบประสาทมนุษย์ประกอบด้วยเซลประสาทประมาณ 10^{11} เซล และมีการเชื่อมต่อกันระหว่างเซลประมาณ 10^{15} จุด เซลประสาทประกอบด้วยตัวเซล เดนไดรท์ (dendrite) และแอกซอน (axon) สัญญาณประสาทจะส่งจากเดนไดรท์ของเซลหนึ่งหรือหลาย ๆ เซล ไปให้แอกซอนอีกเซลหรือหลาย ๆ เซลด้วยการผ่านจุดไซแนบซ์ (synepse) ส่วนตัวเซลของเซลประสาทที่เป็นตัวรับสัญญาณจะทำหน้าที่รวบรวมสัญญาณจากข้อมูลเข้า ข้อมูลเข้าบางกลุ่มที่มีค่าผลรวมสัญญาณมากกว่าเทรสโฮลด์ (threshold) จะกระตุ้นเซลประสาทเซลนี้ได้ ส่วนกลุ่มที่มีผลรวมของสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รับสัญญาณจะทำหน้าที่รวบรวมสัญญาณจากข้อมูลเข้า ข้อมูลเข้าบางกลุ่มที่มีค่าผลรวมสัญญาณมากกว่าเทรสโฮลด์ (threshold) จะกระตุ้นเซลประสาทเซลนี้ได้ ส่วนกลุ่มที่มีผลรวมของสัญญาณประสาทน้อยกว่าเทรสโฮลด์ก็จะไม่สามารถกระตุ้นเซลประสาทได้ ดังแสดงภาพเซลประสาทของสิ่งมีชีวิตในภาพที่ 2

ภาพที่ 2



แสดงการเชื่อมต่อกันของเซลประสาทมนุษย์ [3]

โครงข่ายประสาทเทียมประกอบด้วยเซลประสาทหลาย ๆ เซล ที่ทำงานร่วมกันเพื่อแก้ปัญหาต่าง ๆ โครงสร้างพื้นฐานที่สามารถนำมาใช้อธิบายการทำงานของโครงข่ายประสาทเทียมในเบื้องต้นในที่นี้คือ adaptive linear combiner

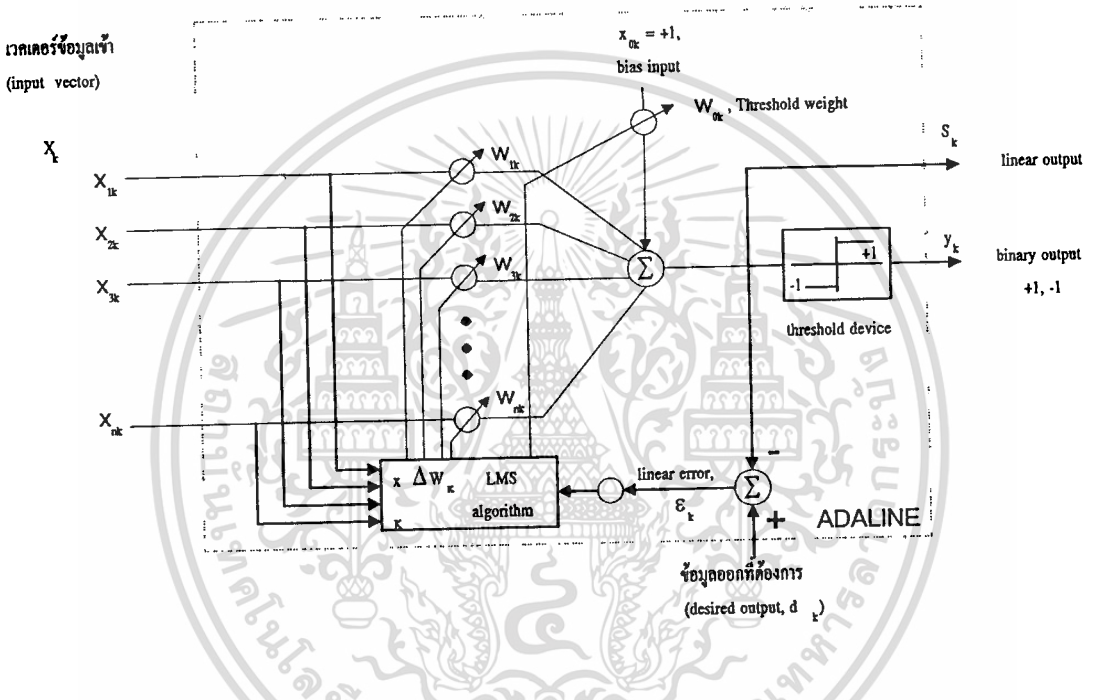
โครงสร้างเซลประสาทเทียมพื้นฐาน

โครงสร้างเซลประสาทเทียมพื้นฐานประกอบด้วย 3 ส่วนคือ ส่วนรับข้อมูลเข้า ส่วนประมวลผล และส่วนส่งข้อมูลออก ส่วนประมวลผลจะทำหน้าที่รวบรวมสัญญาณการกระตุ้นที่รับจากข้อมูลเข้า ถ้าสัญญาณการกระตุ้นมากกว่าค่าที่กำหนด (ซึ่งเรียกค่าที่กำหนดนี้ว่า เทรสโฮลด์,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

threshold) เซลประสาทเทียมก็จะกระตุ้น แต่ถ้ามีค่าต่ำกว่าเซลประสาทก็จะไม่กระตุ้น ซึ่งแทนความหมายการกระตุ้นด้วย +1 และไม่กระตุ้นด้วย -1 (หรือ 0) ดังแสดงตัวอย่างเซลประสาทเทียมในภาพที่ 3 โดยเวคเตอร์ข้อมูลเข้าแทนส่วนรับข้อมูลเข้า ส่วน ADALINE (มีเส้นประล้อมรอบ) แทนส่วนประมวลผล ข้อมูลออกไบนารี่แทนส่วนข้อมูลออก

ภาพที่ 3



แสดงโครงสร้างของเซลประสาทเทียม 1 เซล [2]

<u>การฝึกโครงข่ายในแบบจำลองยุคแรก</u>

โครงสร้างพื้นฐานที่ใช้จำลองโครงข่ายประสาทเทียมในหลาย ๆ แบบจำลอง (model) ก็คือ "Adaptive linear element" หรือเรียกว่า Adaline แสดงดังในภาพที่ 2.3 ประกอบด้วย adaptive linear combiner และ เทรสโฮลด์ดีไวซ์ (threshold device) ที่ใช้ในการแปลงข้อมูลออกให้อยู่ในภาพไบนารี่ +1 หรือ -1 โดยใช้ฟังก์ชันซิกมอยด์ ($Y_k = \mathrm{sgn}(s_k)$) ขณะเดียวกันค่าน้ำหนักไบเอส (bias weight, w_{0k}) และ ค่าข้อมูลเข้าคงที่ (x_0 = +1) จะถูกป้อนเข้าไปด้วยเพื่อช่วยเพิ่มความชัดเจนในการแยกแยะขอบเขตข้อมูลให้ดียิ่งขึ้น ซึ่งน้ำหนักไบเอสจะถูกปรับด้วยในระหว่างการฝึก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

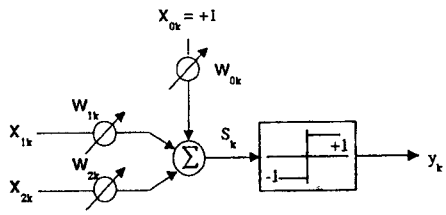
การฝึกโครงข่ายประสาทที่ประกอบด้วยเซลเพียงเซลเดียว จะใช้อัลกอริทึมอะเดฟทีฟ (adaptive algorithm) เช่น อัลกอริทึม LMS หรือ อัลกอริทึมเพอร์เซพตอน (Percepton) โดยจะเลือกชุดตัวอย่างเพื่อใช้ในการฝึกขึ้นมาจำนวนหนึ่งแล้วป้อนเข้าสู่โครงข่ายทีละตัวอย่าง ข้อมูลออกเชิงเส้นที่ได้จะนำไปเข้าฟังก์ชันซิกด์มอยด์ ซึ่งจะแปลงค่าข้อมูลออกเป็นไบนารี่ แล้วนำไปเปรียบเทียบกับค่าข้อมูลออกที่ต้องการ หากค่าที่ได้ไม่ตรงตามที่ต้องการก็จะนำค่าผิดพลาดซึ่งคำนวณจากความต่างระหว่างค่าข้อมูลออกที่ต้องการ และค่าข้อมูลออกเชิงเส้นไปปรับค่าน้ำหนักใหม่จนกว่าจะได้ค่าข้อมูลออกถูกต้องตามต้องการ เมื่อฝึกด้วยข้อมูลเข้าที่หลากหลายจนแยกแยะข้อมูลตัวอย่างฝึกได้ทั้งหมด และทดสอบการจดจำของโครงข่ายด้วยชุดตัวอย่างที่ไม่ได้ใช้ในการฝึกแล้ว Adaline ให้ผลการทดสอบที่ถูกต้องที่สูง แสดงว่า มีคุณสมบัติความเป็นทั่วไป (generalization) ซึ่งความเป็นทั่วไป และการเรียนรู้ ถือเป็นหัวใจสำคัญของ Adaline และโครงข่ายประสาทเทียม

<u>การแบ่งแบบเชิงเส้น (Linear Classifier)</u>

การใช้ Adaline 1 เซล ที่มีข้อมูลเข้า n ค่า และข้อมูลออก 1 ค่า โดยค่าข้อมูลเข้าและค่าข้อมูลออกที่ต้องการเป็นค่าไบนารี่ -1 และ +1 เท่านั้น ดังนั้น ข้อมูลเข้าที่เป็นไปได้มีเท่ากับ 2^n แบบ กรณีนี้จะมีฟังก์ชั่นตรรกที่เป็นไปได้ 2^{2^n} ฟังก์ชั่น การใช้ Adaline เพียง 1 หน่วยจะแก้ฟังก์ชั่นได้เพียงส่วนหนึ่งของฟังก์ชั่นทั้งหมดนี้เท่านั้น และเรียกฟังก์ชั่นที่แก้ได้ด้วย Adaline 1 หน่วยว่า ฟังก์ชั่นตรรกแบ่งแยกแบบเชิงเส้น (linearly separable logic function)

ภาพที่ 4 แสดง Adaline ที่มีข้อมูลเข้า (input) ที่เป็นไบนารี่จำนวน 2 ค่า($n = 2$) ดังนั้นจึงมีภาพแบบข้อมูลเข้า 4 (เท่ากับ 2^n) แบบ คือ (-1, -1) , (-1, +1) , (+1, -1) , (+1,+1) ฟังก์ชั่นตรรก (logic function) ที่เป็นไปได้มี 16 (เท่ากับ 2^{2^n}) ฟังก์ชั่น ดังแสดงในตารางที่ 2.1

ภาพที่ 4



แสดง Adaline ที่มีข้อมูลเข้าที่เป็นไบนารี่จำนวน 2 ค่า [2]

ภาพที่ 5 ได้นำข้อมูลเข้า x_1 และ x_2 มาพลอต (plot) เป็นกราฟในระนาบสองมิติ โดยให้แกนนอนแทน x_1 และแกนตั้งแทน x_2 เพื่อแสดงการแยกแยะข้อมูลเข้าจากตัวอย่างตารางที่ 1 จะเห็นว่า ในฟังก์ชั่นที่เป็นไปได้ทั้งหมด 16 ฟังก์ชั่น ฟังก์ชั่นที่แบ่งด้วยเส้นตรงได้มี 14 ฟังก์ชั่น อีก 2 ฟังก์ชั่นไม่สามารถแบ่งได้ด้วยเส้นตรงเพียงเส้นเดียวคือ ฟังก์ชั่นที่ 9, 10

ค่าข้อมูลออกเชิงเส้น (S) คือ ผลรวมของผลลัพธ์ของการคูณกันระหว่างเวคเตอร์ข้อมูลเข้าและเวคเตอร์น้ำหนัก ในกรณี Adaline ที่มีข้อมูลเข้า 2 ค่า ข้อมูลออกเชิงเส้น (S) คำนวนได้ดังสมการที่ 2.1

$$s = x_1 w_1 + x_2 w_2 + w_0 \quad \text{.....(2.1)}$$

แกนนอนและแกนตั้งในระนาบสองมิติก็คือ ค่าข้อมูลเข้า x_1 และ x_2 เมื่อให้สมการที่ 2.1 เท่ากับ 0 จะได้ว่า

$$x_2 = -\frac{w_1}{w_2} x_1 - \frac{w_0}{w_2} \quad \text{.....(2.2)}$$

$$slope = -\frac{w_1}{w_2} \quad \text{.....(2.3)}$$

x_1 , x_2 : สมาชิกในเวคเตอร์ข้อมูลเข้าตัวที่ 1 และ 2 ตามลำดับ

w_0 : ค่าน้ำหนักไบเอส

w_1 , w_2 : สมาชิกในเวค์เตอร์น้ำหนักตัวที่ 1 และ 2 ตามลำดับ

s : ข้อมูลออกเชิงเส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

slope : ความชัน

ตารางที่ 1

ข้อมูลเข้า		ฟังก์ชันตรรก															
X_1	X_2	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
-1	-1	-1	-1	-1	-1	+1	-1	+1	+1	-1	+1	-1	-1	+1	+1	+1	+1
-1	+1	-1	-1	-1	+1	-1	-1	-1	+1	+1	-1	+1	+1	-1	+1	+1	+1
+1	-1	-1	-1	+1	-1	-1	+1	-1	-1	-1	+1	+1	+1	+1	-1	+1	+1
+1	+1	-1	+1	-1	-1	-1	+1	+1	-1	+1	-1	-1	+1	+1	+1	-1	+1

แสดงฟังก์ชั่นตรรกที่เป็นไปได้ของ Adaline 1 เซลซึ่งมีข้อมูลแบบไบนารี่
โดยมีข้อมูลเข้า 2 ค่า และข้อมูลออก 1 ค่า

เมื่อสมการที่ 2.1 มีค่าเท่ากับ 0 หรือคือ ค่าข้อมูลออกเชิงเส้น (linear output ,s) เท่ากับ 0 จะเป็นเงื่อนไข ณ ตำแหน่งตรงเส้นตรงพอดี จะเรียกเงื่อนไขกรณีนี้ว่า เงื่อนไขเทรสโฮลด์วิกฤต (critcal thresholding condition)

เมื่อสมการที่ 2.1 มีค่าเท่ากับ 0 หรือคือ ค่าข้อมูลออกเชิงเส้น (linear output, s) เท่ากับ 0 จะเป็นเงื่อนไข ณ ตำแหน่งตรงเส้นตรงพอดี จะเรียกเงื่อนไขกรณีนี้ว่า เงื่อนไขเทรสโฮลด์วิกฤต (critcal thresholding condition)

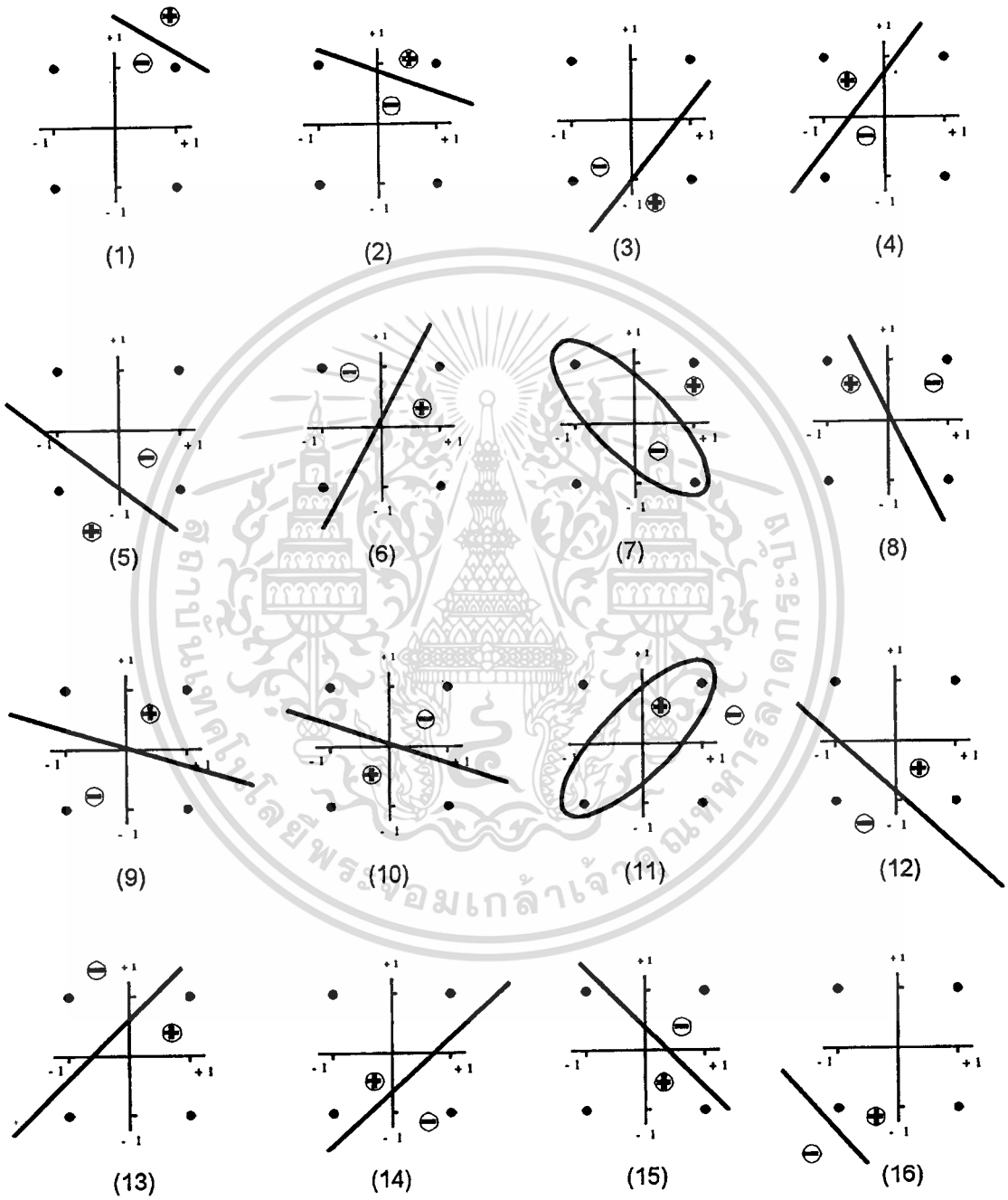
<u>ความสามารถในการแยกแยะเชิงเส้น (Capacity of Linear Classifier)</u>

ในการฝึกโครงข่ายประสาทมีเป้าหมายเพื่อให้โครงข่ายประสาทสามารถแยกแยะข้อมูลแบบต่าง ๆ ในชุดการฝึกได้ถูกต้อง กรณี Adaline 1 หน่วยที่มีข้อมูลเข้า 2 ค่า และข้อมูลออก 1 ค่าที่กล่าวในข้างต้น จะมีจำนวนฟังก์ชั่นที่สามารถใช้เส้นตรงแบ่งกลุ่มได้ 14 ในทั้งหมด 16 ฟังก์ชั่น ค่าความสามารถในการแยกแยะ (Capacity) จะเท่ากับ 14 และเนื่องจากเป็นการแยกแยะโดยใช้เส้นตรง จึงเรียกว่า ความสามารถในการแยกแยะเชิงเส้น (Capacity of Linear Classifier)

นิลสัน [2] (Nilsson) ได้ประมาณความสามารถในการแยกแยะของ Adaline ไว้ว่า จำนวนข้อมูลที่ Adaline แยกแยะได้ถูกต้องจะมีเท่ากับสองเท่าของจำนวนน้ำหนัก ดังแสดงในสมการที่ 2.4 (โดยค่าประมาณนี้เป็นค่าโดยเฉลี่ย) ซึ่งเรียกค่านี้ว่า " statistic pattern capacity, C_s "

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 5



แกนนอน แทนค่า x_1 และ แกนตั้ง แทนค่า x_2

⊕ แทนค่า ข้อมูลออก +1 และ ⊖ แทนค่า ข้อมูลออก - 1

แสดงการแบ่งฟังก์ชั่นตรรกในตารางที่ 1 ที่บางฟังก์ชั่นแบ่งไม่ได้ด้วยเส้นตรงเส้นเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$C_s = 2N_w \quad \text{.....(2.4)}$$

C_s : ค่าความสามารถในการแยกแยะเชิงเส้น

N_w : จำนวนเวคเตอร์น้ำหนัก

นอกจากนี้นิลสันยังคาดการณ์โอกาส (probability) ที่ข้อมูลในชุดฝึกจะแยกแยะได้ถูกต้องทั้งหมดด้วย Adaline ไว้ดังสมการที่ 2.5

$$P_{separable} = \begin{cases} 2^{-(N_p-1)} \sum_{i=0}^{N_w-1} \binom{N_p-1}{i} & \text{for } N_p > N_w \\ 1 & \text{for } N_p \le N_w \end{cases} \quad \text{.....(2.5)}$$

N_w : จำนวนเวคเตอร์น้ำหนัก

N_p : จำนวนแบบข้อมูลเข้าที่ใช้ในการฝึก

$P_{separable}$: โอกาสในการแยกแยะข้อมูลที่ใช้ฝึกได้ถูกต้อง

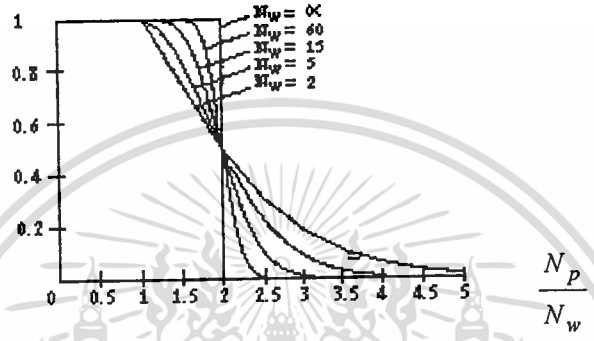
จากสมการที่ 2.5 เมื่อนำมาพลอต (plot) กราฟความสัมพันธ์ระหว่างโอกาสในการแบ่งแยกเชิงเส้น (probability of linear separability) และอัตราส่วนของจำนวนแบบข้อมูลต่อจำนวนน้ำหนัก (N_p/N_w) จะได้กราฟดังแสดงในภาพที่ 6 พบว่า ยิ่งจำนวนน้ำหนักมากขึ้นเท่าใด โอกาสที่ข้อมูลจะการแยกแยะได้ก็มากขึ้นหรือก็คือจำนวนข้อมูลที่จะแยกแยะได้จะมีเพิ่มมากขึ้น และหากน้ำหนักมีจำนวนมากอย่างไม่จำกัดจำนวนข้อมูลที่โครงข่ายสามารถแยกแยะได้จะเท่ากับสองเท่าของจำนวนน้ำหนัก ($C_s = 2N_w$) หรือก็คือ เท่ากับค่า statistical pattern capacity แต่ในความเป็นจริงไม่สามารถให้น้ำหนักมีจำนวนมากอย่างไม่จำกัดได้ ดังนั้น จำนวนข้อมูลที่โครงข่ายสามารถแยกแยะได้จึงน้อยกว่าค่า statistical pattern capacity แน่นอน เมื่อพิจารณาจากภาพกราฟ กรณีที่ N_p/N_w มีค่าเท่ากับหรือน้อยกว่า 1 โอกาสในการแยกแยะข้อมูลจะเท่ากับ 1 หมายความว่า ถ้าจำนวนแบบข้อมูลมีเท่ากับหรือน้อยกว่าจำนวนน้ำหนักจะสามารถแยกแยะข้อมูลได้ทั้งหมดแน่นอน โดยจำนวนข้อมูลที่แยกแยะได้จะเท่ากับจำนวนน้ำหนักในโครงข่าย ซึ่งจะกำหนดให้จำนวนข้อมูลที่แยกแยะได้ในกรณีนี้เป็นค่า deterministic pattern capacity , $C_d = N_w$ กรณีที่น้ำหนักมีจำนวนจำกัด ($N_w \neq \alpha$) และ N_p/N_w มีค่ามากกว่า 1 โอกาสที่ข้อมูลจะแยกแยะได้ก็จะมีน้อยลงไปตามจำนวนน้ำหนัก ซึ่งหมายถึง ข้อมูลแยกแยะไม่ได้ทั้งหมด ถ้าจำนวนข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มากกว่าจำนวนน้ำหนัก จึงสรุปได้ว่า การที่ Adaline จะสามารถแบ่งข้อมูลออกเป็น 2 ขอบเขตได้ ต้องมีจำนวนน้ำหนักมากกว่าหรือเท่ากับจำนวนข้อมูลเท่านั้น และจำนวนข้อมูลที่แบ่งแยกได้ก็จะมีจำนวนไม่เกินจำนวนน้ำหนัก

ภาพที่ 6

Probability ของการแยกแยะแบบเชิงเส้น



กราฟแสดงความสัมพันธ์ระหว่างโอกาสในการที่ Adaline แยกแยะข้อมูลได้กับอัตราส่วน $\frac{N_p}{N_w}$ [2]

<u>การแบ่งแบบไม่เชิงเส้น (Nonlinear Classifier)</u>

การแบ่งแบบเชิงเส้น มีข้อจำกัดอยู่ที่จะแก้ปัญหาได้เฉพาะข้อมูลที่สามารถแบ่งเป็นสองกลุ่มได้ด้วยเส้นตรงเท่านั้น กรณีปัญหาที่มีความซับซ้อนไม่อาจแบ่งเป็นสองกลุ่มด้วยเส้นตรงเส้นเดียวได้ จะต้องแก้ปัญหาด้วยวิธีอื่น การแบ่งแบบไม่เชิงเส้นเป็นวิธีหนึ่งที่สามารถใช้ในการแก้ปัญหานี้ได้ การแบ่งแบบไม่เชิงเส้นอาจใช้โครงข่ายที่มีเซลประสาทเทียม (ที่เรียกว่า โหนด) เพียงเซลเดียวหรือหลายเซลประกอบกันเป็นชั้น (layer) ก็ได้ ดังตัวอย่างที่จะกล่าวต่อไปนี้

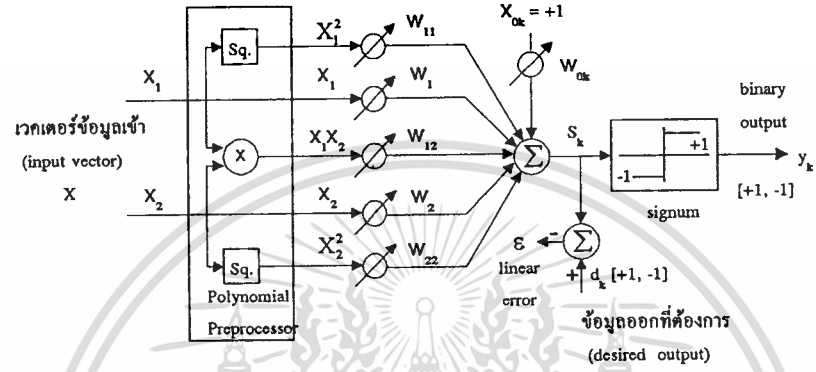
<u>ฟังก์ชั่นโพลิโนเมียล (Polynomial Discriminant Functions)</u>

ฟังก์ชั่นโพลิโนเมียลเป็นการแบ่งแบบไม่เชิงเส้น (nonlinear) ที่ใช้ Adaline เพียง 1 หน่วย (เซลเดียว) ในการแบ่งขอบเขต ภาพที่ 7 แสดงการแปลงข้อมูลเชิงเส้น (linear input) ให้อยู่ในภาพฟังก์ชั่นควอดราติก (quadratic function) ณ เส้นแบ่งขอบเขตข้อมูลถือเป็นเงื่อนไข ณ จุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิกฤต (critical thresholding condition) ดังแสดงตัวอย่างในสมการที่ 2.6 (มีจำนวนข้อมูลเข้า 2 ค่า)

ภาพที่ 7



แสดง Adaline 1 เซล ที่มีการแปลงข้อมูลเข้าเชิงเส้นเป็นข้อมูลเข้าไม่เชิงเส้น [2]

$$s = w_0 + x_1 w_1 + x_1^2 w_{11} + x_1 x_2 w_{12} + x_2^2 w_{22} + x_2 w_2 = 0 \quad \text{.....(2.6)}$$

ตัวอย่างปัญหาที่แก้โดยการแบ่งแบบเชิงเส้นไม่ได้แต่แก้โดยใช้ฟังก์ชั่นโพลิโนเมียลได้ เช่น ปัญหาของตรรก exclusive NOR ที่แสดงในตารางที่ 1 ฟังก์ชั่นที่ 7 เมื่อใช้ฟังก์ชั่นโพลิโนเมียลดีกรีสองดังในสมการที่ 2.6 แล้วทำการฝึกโครงข่ายจนได้ค่าน้ำหนักที่เหมาะสมแล้ว ก็จะสามาถแบ่งขอบเขตข้อมูลเป็นสองขอบเขตได้ ดังแสดงเส้นแบ่งขอบเขตได้ดังในภาพที่ 5(7)

ค่าความสามารถในการแยกแยะแบบไม่เชิงเส้น หมายถึง ค่าเฉลี่ยของจำนวนแบบข้อมูลที่ Madaline I สามารถเรียนรู้ที่จะแยกแยะข้อมูลได้ถูกต้อง จากการทดลองของโคฟอร์ด [2] (Koford) พบว่า แม้การใช้ค่าตรรกในชั้นที่สองของโครงข่ายประสาทไม่ว่าเป็น AND หรือ OR ก็จะให้ค่าความสามารถในการแยกแยะต่อ Adaline 1 เซลเท่ากัน ($C_s = 2N_w$) กรณีการแยกแยะแบบไม่เชิงเส้นโดยใช้แบบจำลอง Madaline I ซึ่งประกอบด้วย Adaline มากกว่า 1 เซล ค่าความสามารถในการแยกแยะแบบไม่เชิงเส้นจะเท่ากับจำนวนแบบข้อมูลที่ Adaline 1 เซลสามารถแยกแยะได้ถูกต้องคูณกับจำนวน Adaline ที่ประกอบเป็นโครงสร้าง Madaline I นั้น อย่างไรก็ตาม แม้จำนวนแบบข้อมูลที่สามารถแยกแยะได้ต่อหน่วย Adaline จะมีค่าเท่ากันในโครงข่ายที่แยกแยะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบเชิงเส้น (ใช้ Adaline 1 หน่วย) และโครงข่ายที่แยกแยะแบบไม่เชิงเส้น (Madaline I) แต่ Adaline ก็มีข้อจำกัดที่สามารถแยกแยะข้อมูลได้โดยใช้เส้นตรงเพียงเส้นเดียวเท่านั้น ขณะที่ Madaline I ไม่มีข้อจำกัดนี้

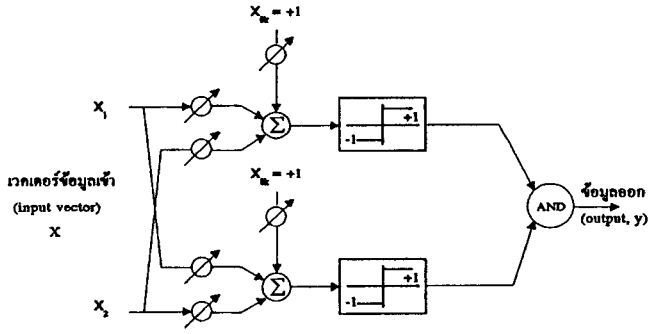
การแบ่งแบบไม่เชิงเส้น (nonlinear) อาจใช้กับข้อมูลเข้าที่มีมากกว่าสองจำนวน และใช้ดีกรีของโพลิโนเมียล (polynomial degree) มากกว่าสองก็ได้

นอกจากนี้ปัญหาบางอย่างที่ซับซ้อน การใช้โครงข่ายที่มีมากกว่า 1 เซล (multi adaptive element) มาประกอบกันเป็นชั้นโครงข่าย ซึ่งอาจมีเพียงชั้นเดียว (single layer) หรือมากกว่าหนึ่งชั้น (multilayer) อาจทำให้สามารถแก้ไขปัญหาได้อย่างดีและมีประสิทธิภาพสูง

Madaline I

Madaline I เป็นโครงข่ายประสาทเทียมชนิดที่เป็นชั้นซึ่งพัฒนาโดย Widrow และ Hoff Madaline I ประกอบด้วย adaptive element หลายหน่วย ในที่นี้จะแสดงการประยุกต์ใช้ Madaline I ในการแก้ปัญหา exclusive NOR ด้วยฟังก์ชั่นตรรก AND ซึ่งเป็นการแก้ปัญหาอีกวิธีหนึ่งนอกจากการใช้ฟังก์ชั่นโพลิโนเมียล โดยตัวอย่างนี้จะกำหนดให้ Madaline I ประกอบด้วย Adaline 2 หน่วย และข้อมูลเข้า 2 จำนวน มีการเชื่อมต่อระหว่างข้อมูลเข้าและ Adaline ทั้งสองอย่างทั่วถึง และเชื่อมต่อ Adaline ทั้งสองกับตรรก AND เพื่อประมวลผลให้ได้ค่าข้อมูลออก (Y_k) ดังแสดงในภาพที่ 8 หลังการฝึกจนได้ค่าน้ำหนักที่เหมาะสมแล้ว ก็จะสามารถแบ่งขอบเขตของข้อมูลได้อย่างถูกต้อง ซึ่งเป็นการแบ่งโดยใช้เส้นตรงสองเส้นดังแสดงในภาพที่ 9 ในการใช้ Madaline I กับปัญหาอื่น ๆ นอกจากตรรก AND ที่ใช้ในชั้นที่สองแล้ว ก็ยังสามารถใช้ตรรกอื่นในการแก้ปัญหาได้ด้วย เช่น ตรรก OR, MAJ (majority) ดังกล่าวในภาพที่ 10

ภาพที่ 8

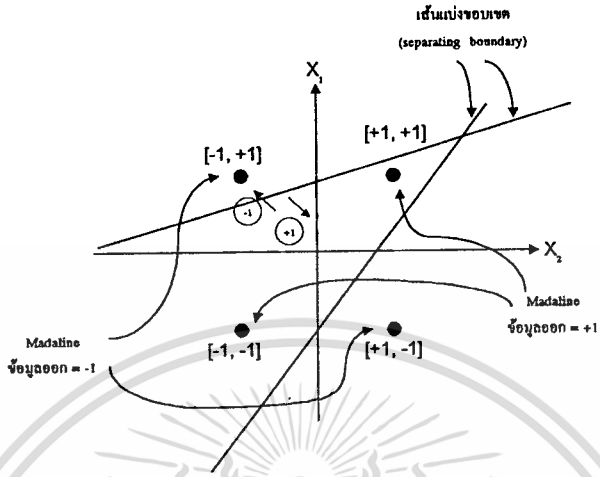


แสดง Madaline ที่ประกอบด้วย Adaline 2 หน่วย [2]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

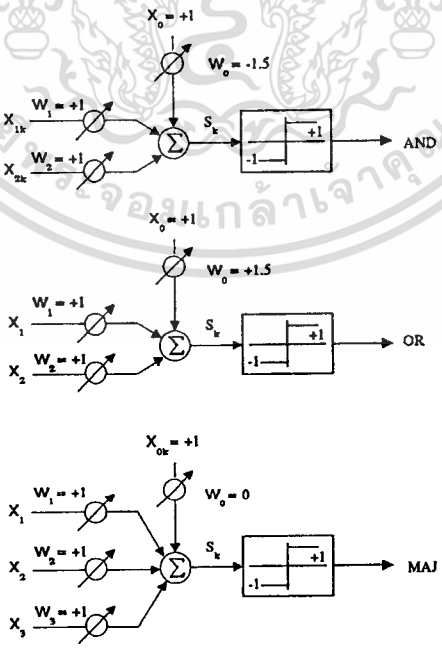
29942

ภาพที่ 9



แสดงเส้นแบ่งขอบเขตของ Madaline ในภาพที่ 8 [2]

ภาพที่ 10



แสดงตัวอย่าง Adaline ที่ใช้กับฟังก์ชั่นตรรก AND, OR MAJ [2]

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ห้ามนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The Minimal Disturbance Principle

สิ่งสำคัญของโครงข่ายประสาทเทียมคือ การฝึกให้โครงข่ายสามารถเรียนรู้ปัญหา ซึ่งไม่ว่าจะฝึกด้วยวิธีการใดก็ตามจะต้องยึดหลักที่ว่า การลดความผิดพลาดของข้อมูลออก (output error) ในขณะใด ๆ ด้วยแบบข้อมูลเข้าหนึ่ง ๆ นั้น จะต้องไปรบกวนการเรียนรู้ของแบบข้อมูลที่เรียนรู้ไปแล้วน้อยที่สุด ทั้งนี้ก็เพื่อพยายามที่จะให้โครงข่ายไม่ลืมแบบข้อมูลที่ได้เรียนรู้ไปแล้ว ซึ่งหลักการนี้เรียกว่า Minimal Disturbance Principle

วิธีการฝึกโครงข่ายประสาทเทียม

การฝึกโครงข่ายมีกฎที่ใช้ต่าง ๆ กันหลายกฎ ซึ่งมีหลักการใหญ่ ๆ ในการปรับค่าน้ำหนักอยู่ 2 ประการ คือ

1. Error Correction Rules - Single Threshold element

หลักการ Error Correction Rules จะพยายามปรับค่าน้ำหนักเพื่อลดค่าน้ำหนักที่ผิดพลาดของโหนดที่ข้อมูลเข้าขณะนั้น ๆ เลือกเป็นตัวแทน เพื่อให้ได้ค่าข้อมูลออกถูกต้องตามที่ต้องการ

2. Steepest Descent Rules

หลักการ Steepest Descent Rules จะพยายามปรับค่าน้ำหนัก เพื่อลดค่าน้ำหนักที่ผิดพลาดโดยเฉลี่ยของทั้งโครงข่าย เพื่อให้ได้ค่าข้อมูลออกถูกต้องตามที่ต้องการ

ในบทนี้จะยกตัวอย่างเฉพาะการปรับค่าน้ำหนักตามหลักการ Error Correction Rules

Error Correction Rules - Single Threshold element

1. กฎเชิงเส้น (Linear rule)

กฎเชิงเส้น อาจเรียกว่า กฎการปรับค่าผิดพลาดเชิงเส้น (linear error correction rule) ก็ได้ เนื่องจากระหว่างการฝึกจะทำการปรับค่าน้ำหนักด้วยค่าผิดพลาดเชิงเส้น (linear error) ตัวอย่างกฎเชิงเส้น ที่จะกล่าวถึงในที่นี้ก็คือ α-LMS

α-LMS หรือเรียกว่า กฎของ Widrow-Hoff จะใช้ Adaline 1 หน่วย ในระหว่างการฝึกปรับค่าน้ำหนัก ดังสมการที่ 2.7 ซึ่งการปรับน้ำหนักจะปรับตามหลักการ Minimum disturbance

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังแสดงในลักษณะเวคเตอร์สามมิติได้ดังภาพที่ 11 ค่าน้ำหนักที่เปลี่ยนไปจะแปรผันตรงกับค่าผิดพลาดเชิงเส้นซึ่งค่าผิดพลาดเชิงเส้น คือผลต่างระหว่างค่าข้อมูลออกที่ต้องการ (d_k) และข้อมูลออกเชิงเส้น ($_k^T X_k$) ดังคำนวณในสมการที่ 2.8

$$W_{k+1} = W_k + \alpha \frac{\varepsilon_k X_k}{|X_k|^2} \quad \text{.....(2.7)}$$

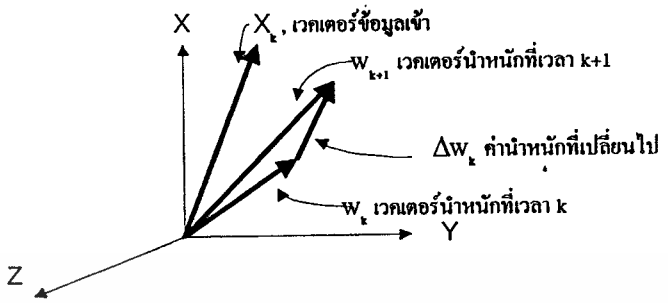
$$\varepsilon_k = d_k - W_k^T X_k \quad \text{.....(2.8)}$$

α	อัตราการเรียนรู้ของโครงข่าย
W_k , W_{k+1}	เวคเตอร์ของค่าน้ำหนักที่เวลา k และ k+1 ตามลำดับ
X_k	เวคเตอร์ข้อมูลเข้าที่เวลา k
d_k	ค่าข้อมูลออกที่ต้องการที่เวลา k
$W_k^T X_k$	ค่าข้อมูลออกจากโครงข่ายโดยตรงที่เวลา k
ε_k	ค่าความแตกต่างระหว่าง d_k และ $_k^T X_k$

ค่า α ซึ่งเป็นค่าอัตราการเรียนรู้ของโครงข่าย จะเป็นค่าที่ควบคุมความคงตัว (stability) ของค่าน้ำหนัก และความเร็วในการปรับค่าน้ำหนักให้เข้าสู่ค่าที่สมดุล (speed of convergence) โดย α มีค่าอยู่ระหว่าง 0 ถึง 1 หากกำหนดค่า α ให้เท่ากับ 1 จะเป็นการปรับให้ค่าผิดพลาดในขณะนั้นให้มีค่าเท่ากับ 0 ซึ่งอาจทำให้โครงข่ายลืมข้อมูลที่จดจำได้ในช่วงแรก การกำหนดค่า α สูงเกินไป ค่าน้ำหนักจะถูกปรับไปปรับมา (oscillation) ทำให้ถึงจุดที่มีค่าน้ำหนักที่สมดุลได้ยากหรืออาจไม่สามารถเข้าสู่สมดุลได้เลย และการกำหนดค่า α น้อยเกินไปก็จะทำให้โครงข่ายประสาทเทียมมีการเรียนรู้ช้ามาก จึงต้องมีการกำหนดค่า α ให้เหมาะสมกับปัญหาแต่ละอย่าง ส่วนค่าข้อมูลเข้าถ้าเป็น 0 หรือ 1 จะทำให้ไม่มีการปรับน้ำหนักเมื่อค่าข้อมูลเข้าเป็น 0 แต่ถ้าใช้ค่าข้อมูลเข้าเป็น -1 แทนค่า 0 จะมีการปรับน้ำหนักทุก ๆ ค่าซึ่งจะทำให้ค่าน้ำหนักถูกปรับเข้าสู่สมดุลเร็วขึ้น ด้วยเหตุนี้จึงนิยมใช้ค่าข้อมูลเข้าที่เป็น -1 หรือ +1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 11



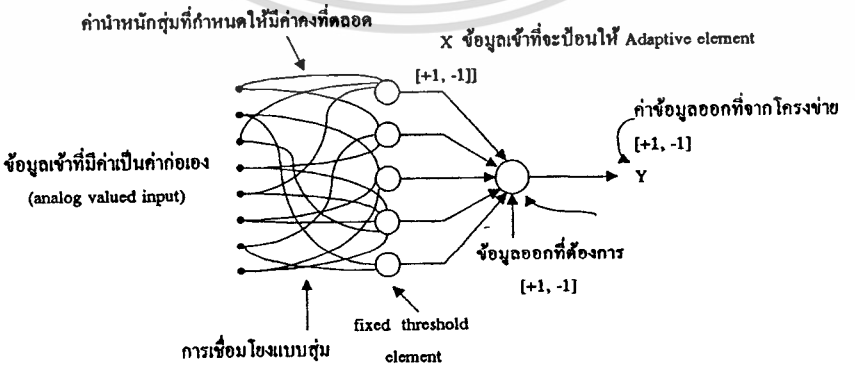
แสดงภาพการปรับน้ำหนักในลักษณะสามมิติ [2]

2. กฎไม่เชิงเส้น (Nonlinear Rules)

ปัญหาที่ไม่สามารถแยกแยะได้โดยใช้เส้นตรงเส้นเดียวจะเรียกว่า ปัญหาไม่เชิงเส้น ก็จะใช้กฎไม่เชิงเส้นฝึกโครงข่าย ซึ่งการปรับค่าน้ำหนักของกฎไม่เชิงเส้นก็ยังยึดหลักการ Minimum disturbance เช่นเดียวกับในกฎเชิงเส้น แต่ต่างกันตรงการคำนวณหาค่าผิดพลาด ดังจะอธิบายในหัวข้อถัดไป กฎไม่เชิงเส้นที่จะยกตัวอย่างในที่นี้ก็คือ กฎการเรียนรู้ของเพอร์เซปตรอน (Perceptron Learning Rule) และอัลกอริทึมของเมย์ (Mays algorithm)

2.1 กฎการเรียนรู้ของเพอร์เซปตรอน (Perceptron Learning Rule)

ภาพที่ 12

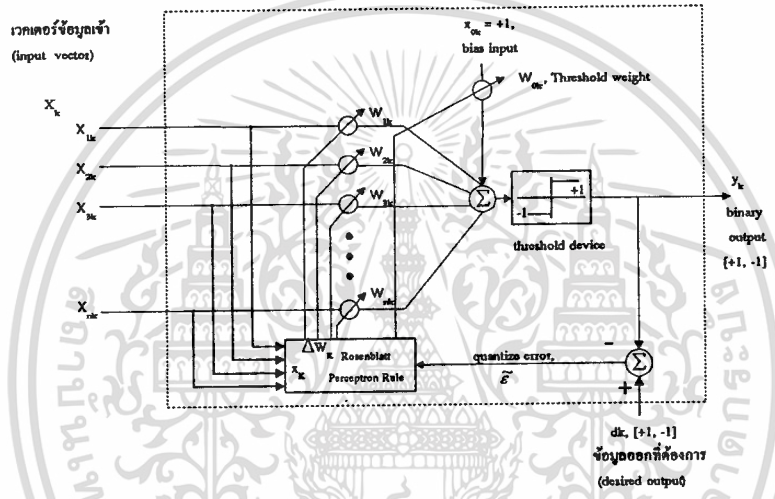


แผนภาพ α-Perceptron ของโรเซนบลอต [2]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กฎการเรียนรู้ของเพอร์เซบตรอน เสนอโดยโรเซนบลอต (Rosenblatt) ดังแสดงเป็นแผนภาพได้ดังภาพที่ 12 ซึ่งจะมีการเชื่อมต่อของข้อมูลเข้าในชั้นแรกกับ fixed logic device แบบสุ่ม ข้อมูลออกจากชั้นแรกจะส่งให้แก่ชั้นที่สองไปประมวลหาค่าข้อมูลออก (Y) สิ่งที่ต่างจาก α-LMS คือ ค่าข้อมูลเข้าจะเป็น +1 หรือ 0 และไม่มีค่าน้ำหนักไบแอส โครงสร้างนี้เทียบเท่ากับ Adaline 1 เซล

ภาพที่ 13



แสดง adaptive threshold element ของ α-Perceptron [2]

ภาพที่ 13 แสดง adaptive threshold element ของ α-Perceptron ที่ใช้กฎเพอร์เซพตรอนในการฝึก ซึ่งจะปรับค่าน้ำหนักด้วยค่าผิดพลาดควอนไทเซอร์ (quantizer error, $\tilde{\varepsilon}$) ค่าผิดพลาดควอนไทเซอร์ คือ ค่าความต่างระหว่างค่าข้อมูลออกที่ต้องการ (d_k) และข้อมูลออกที่ผ่านการควอนไทเซอร์แล้ว (Y_k , เป็นข้อมูลออกไบนารี่ที่แปลงจากข้อมูลออกเชิงเส้นโดยเทรสโฮลด์ดีไวซ์) ดังแสดงในสมการที่ 2.12

$$\tilde{\varepsilon} = d_k - Y_k \quad \text{.....(2.12)}$$

$\tilde{\varepsilon}$ ค่าผิดพลาดควอนไทเซอร์

d_k ค่าข้อมูลออกที่ต้องการที่เวลา k

Y_k ค่าข้อมูลออกที่ผ่านการควอนไทเซอร์แล้วที่เวลา k

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กฎการเรียนรู้ของ α-Perceptron จะปรับค่าน้ำหนักเฉพาะกรณีที่ข้อมูลออกที่ผ่านการควอนไทเซอร์แล้วให้ค่าไม่ถูกต้องตามที่ต้องการ โดยปรับน้ำหนักตามสมการที่ 2.13 กรณีที่ข้อมูลออกที่ผ่านการควอนไทเซอร์แล้วให้ค่าถูกต้องตามต้องการจะไม่ปรับค่าน้ำหนัก (คือ การที่ค่าผิดพลาดควอนไทเซอร์ มีค่าเท่ากับ 0)

$$W_{k+1} = W_k + \alpha \frac{\tilde{\varepsilon}}{2} X_k \quad \text{.....(2.13)}$$

โรเซนบลอต กำหนดให้ α ในกฎเพอร์เซบตรอนมีค่าเป็น 1 ค่า α จึงไม่มีผลต่อความคงตัว แต่จะมีผลต่อเวลาที่ใช้ในการปรับน้ำหนักให้เข้าสู่สภาวะสมดุล ข้อแตกต่างระหว่าง α-LMS และกฎเพอร์เซบตรอนก็คือ ค่าข้อมูลออกที่ต้องการ (d_k) ของเพอร์เซบตรอนจะเป็นเฉพาะค่าไบนารี่เท่านั้น แต่ α-LMS เป็นได้ทั้งค่าไบนารี่และค่าต่อเนื่อง และ α-LMS ใช้ฝึกโครงข่ายที่มีการแยกแยะแบบเชิงเส้นเท่านั้น แต่เพอร์เซบตรอนใช้ฝึกโครงข่ายที่มีการแยกแยะแบบไม่เชิงเส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

โครงข่ายประสาทเทียมแบบ
KOHONEN SELF-ORGANIZING MAPS (SOM)

บทนำ

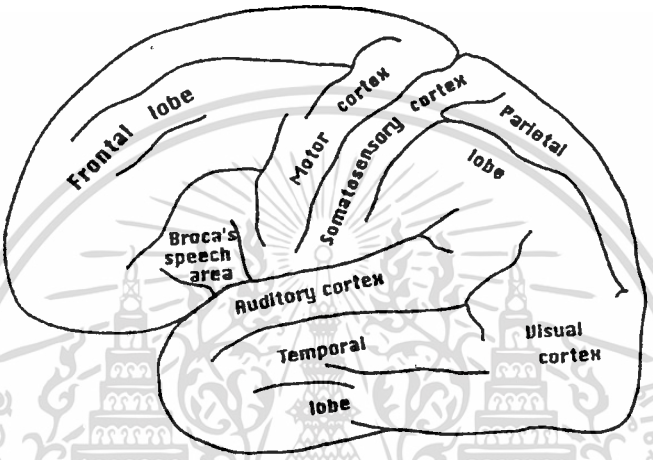
สมองส่วนเปลือกนอก (cerebral cortex) เป็นโครงสร้างที่ธรรมชาติสร้างได้ดีที่สุดอย่างหนึ่งในระบบกายวิภาคของมนุษย์ แม้ว่า ในรายละเอียดแล้วสมองจะมีโครงสร้างที่มีความซับซ้อนสูง แต่หากมองในภาพรวมแล้วส่วนสมองเปลือกนอกก็เป็นส่วนที่ส่งสัญญาณควบคุมการทำงานไปยังส่วนต่าง ๆ ของร่างกาย ถ้านำชั้นโครงข่ายประสาทมาแผ่ออกเป็นผืน 2 มิติ ภายในชั้นประสาทจะประกอบด้วยเส้นใยประสาทจำนวนมากมายที่เชื่อมต่อกัน เมื่อเซลประสาทเซลหนึ่งถูกกระตุ้นเซลประสาทที่อยู่รอบ ๆ เซลนั้นก็จะถูกกระตุ้นไปด้วย ซึ่งบริเวณรอบ ๆ ที่ถูกกระตุ้นจะกว้างเพียงใดก็ขึ้นกับขนาดความเข้มของสัญญาณการกระตุ้น สมองเป็นศูนย์รวมระบบการทำงานต่าง ๆ ของร่างกาย (ภาพที่ 14) เช่น การคิด การพูด การเห็น การได้ยิน โดยจะมีการแบ่งส่วนเปลือกสมองเป็นส่วน ๆ แต่ละส่วนก็จะควบคุมระบบการทำงานแต่ละอย่างไป และในแต่ละส่วนก็จะได้รับระดับของการกระตุ้นต่าง ๆ กันไปตามลำดับ ซึ่งเรียกว่า "order feature maps" เช่น ส่วนที่รับการได้ยิน เซลที่อยู่รอบ ๆ (neighboring neural) ก็จะได้รับสัญญาณเสียงที่มีความถี่เดียวกันแต่มีระดับเสียงต่างกัน เซลที่อยู่ใกล้เซลที่ถูกกระตุ้นโดยตรงก็จะได้รับสัญญาณเสียงที่มีระดับเสียงสูงกว่าเซลที่อยู่ห่างออกไป และระดับเสียงก็จะลดต่ำลงไปตามลำดับความห่างของเซลที่ห่างจากเซลที่ถูกกระตุ้น

ส่วนเปลือกสมองเมื่อแผ่ออกจะมีพื้นที่ขนาดใหญ่มาก (ประมาณ 1 ตารางเมตร ในวัยผู้ใหญ่) แต่บาง ๆ (หนา 2-4 มิลลิเมตร) ประกอบด้วยเซลประสาท 6 ชั้น ซึ่งในแต่ละส่วนจะมีชนิดและความหนาแน่นของเซลและหน้าที่ควบคุมการทำงานที่แตกต่างกัน โดยจะขดไปขดมาให้มีพื้นที่มากที่สุดในการบรรจุอยู่ในกระโหลกศีรษะ นอกจากนี้เซลที่อยู่ใกล้เคียงกันก็จะเก็บข้อมูลที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีความสัมพันธ์กัน สมองไม่ได้บันทึกข้อมูลทั้งหมดโดยตรงไว้ แต่เป็นข้อมูลที่ผ่านการบีบ (compress) แล้ว เพื่อลดพื้นที่สมองที่จะใช้เก็บโดยข้อมูลที่ผ่านการบีบแล้วนี้ยังคงความสัมพันธ์ของข้อมูลก่อนการบีบไว้เหมือนเดิม

ภาพที่ 14



แสดงสมองส่วนเปลือกนอกในภาพด้านข้างที่หนาประมาณ 2-3 มิลลิเมตร โดยพื้นที่แต่ละส่วนของสมองจะมีหน้าที่ควบคุมการทำงานต่าง ๆ กัน [4]

จากการศึกษาระบบการทำงานของสมอง โคโฮเนน (Kohonen) ได้สร้างโครงข่ายประสาทเทียมเลียนแบบการทำงานของสมอง ที่เรียกว่า " Kohonen Self-Organizing Map " ขึ้น โดยประกอบด้วยโหนด (หรือที่เรียกว่า เซลประสาทเทียม) วางเรียงต่อกันเป็นโครงสร้างสองมิติ ตำแหน่งของแต่ละโหนดในโครงข่ายจะถูกกำหนดไว้คงที่ก่อนเริ่มต้นฝึกโครงข่าย เฉพาะค่าพารามิเตอร์ต่าง ๆ ของโครงข่ายเท่านั้นที่มีการเปลี่ยนแปลง โหนดใดจะถูกกระตุ้นเมื่อมีการรับข้อมูลเข้า ขึ้นอยู่กับคุณลักษณะ(feature) ของข้อมูลเข้าคือ โหนดที่มีคุณลักษณะใกล้เคียงข้อมูลเข้ามากที่สุด (similarity) จะเป็นโหนดที่ถูกกระตุ้น ข้อมูลที่จะใช้ฝึกโครงข่ายส่วนใหญ่จะมีหลายมิติ (high dimensional) การบันทึกความรู้ให้โครงข่ายด้วยข้อมูลหลายมิติทำให้ต้องใช้พื้นที่หน่วยความจำในการบันทึกมาก แบบจำลองนี้จึงทำการแมป (mapping) ข้อมูลหลายมิติมาเป็นมิติที่ต่ำกว่าโดยยังคงรักษาความสัมพันธ์ของข้อมูลไว้ (preserve the topological relation) คือ ข้อมูลหลายมิติที่อยู่ใกล้เคียงกัน เมื่อแมปเป็นมิติที่ต่ำกว่าก็คงอยู่ใกล้เคียงกันเช่นเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Self-Organizing Maps เป็นแบบจำลองโครงข่ายประสาทชนิดการเรียนรู้แบบไม่มีผู้สอน ประกอบด้วยชั้นข้อมูลเข้า (input layer) ชั้นคอมเพททิทีฟ (competitive layer) และชั้นข้อมูลออก (output layer) แต่ในงานวิจัยนี้จะใช้โครงข่ายที่มี 2 ชั้นเท่านั้นคือ ชั้นรับข้อมูลเข้า และชั้นคอมเพททิทีฟ ดังจะกล่าวถึงรายละเอียดในหัวข้อต่อไป

สถาปัตยกรรม SOM

สถาปัตยกรรม SOM มีลักษณะดังแสดงในภาพที่ 15 โดย

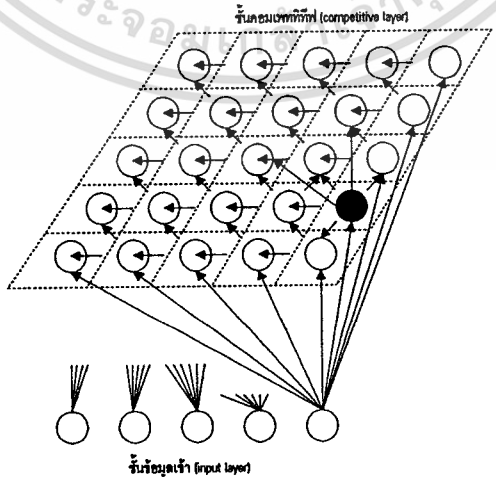
1. โครงข่ายมี 2 ชั้นคือ ชั้นข้อมูลเข้าและชั้นคอมเพททิทีฟ แต่ละชั้นประกอบด้วยโหนด (เซลประสาท) หลายโหนด โหนดในชั้นข้อมูลเข้าและชั้นคอมเพททิทีฟ เรียกว่า โหนดรับข้อมูลเข้า(input node) และโหนดคอมเพททิทีฟ (competitive node) ตามลำดับ

2. การเชื่อมต่อระหว่างชั้นที่ติดกันเป็นแบบทั่วถึง (fully connect) ด้วยค่าน้ำหนัก โดยทุก ๆ โหนดในชั้นข้อมูลเข้าจะส่งสัญญาณไปยังทุกโหนดในชั้นคอมเพททิทีฟ

3.ชั้นข้อมูลเข้าไม่มีการประมวลผล จะทำหน้าที่ในการรับสัญญาณเข้า และกระจายออกไปยังแต่ละโหนดในชั้นคอมเพททิทีฟเท่านั้นโดยมีการประมวลผลเฉพาะในชั้นคอมเพททิทีฟเท่านั้น

4. ชั้นคอมเพททิทีฟ โหนดจะมีการเรียงตัวกันเป็นระนาบ 2 มิติ เมื่อโหนดใดโหนดหนึ่งถูกกระตุ้น โหนดที่อยู่ข้างเคียง (neighbor node) ก็จะได้รับสัญญาณการกระตุ้นด้วย

ภาพที่ 15



แสดงสถาปัตยกรรมของ SOM [4]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ชั้นข้อมูลเข้า (Input Layer)

ส่วนมากชั้นข้อมูลเข้าจะไม่ค่อยกล่าวถึง เนื่องจากชั้นนี้จะทำการรับข้อมูลเข้าแล้วกระจายข้อมูลที่รับเข้ามาไปให้ยังชั้นคอมเพททิทีฟ เพื่อทำการประมวลผลต่อไปเท่านั้น การจำลองให้คอมพิวเตอร์ทำงานเลียนแบบสมอง ข้อมูลเข้าจะต้องสามารถนำไปคำนวณโดยใช้ซีพียูของคอมพิวเตอร์ได้ จึงต้องเตรียมข้อมูลให้เป็นมาตรฐานหรือเรียกว่า การทำนอร์มอลไลซ์ (normalise) ารนอร์มอลไลซ์ค่าข้อมูลเข้าก็เพื่อป้องกันการเกิดข้อผิดพลาดจากการเกิดโอเวอร์โฟลว์ (overflow) ในการคำนวณ ซึ่งในทางชีววิทยาแล้วจะไม่มีการโปรเซสข้อมูลเข้าก่อนที่จะส่งเข้าสู่ระบบประสาท เพราะ กลไกลภายในจะมีขบวนการป้องกันอย่างอัตโนมัติเมื่อมีสัญญาณข้อมูลเข้าขนาดใหญ่เข้ามากระตุ้นเซลประสาทเพื่อไม่ให้เซลเสียหาย การนอร์มอลไลซ์ทำได้หลายวิธี วิธีหนึ่งคือ การนำค่าข้อมูลเข้าจากแต่ละเซล หารด้วยผลรวมของข้อมูลเข้าของทุก ๆ เซล ดังสมการที่ 3.1

$$\Theta_i = I_i\left(\sum I_i\right)^{-1} \quad .....i = 0,1,...,n.....(3.1)$$

n จำนวนสมาชิกในเวคเตอร์ข้อมูลเข้า

Θ_i สมาชิกเวคเตอร์ข้อมูลเข้าตัวที่ 0,1...,n ที่ผ่านการนอร์มอลไลซ์แล้ว

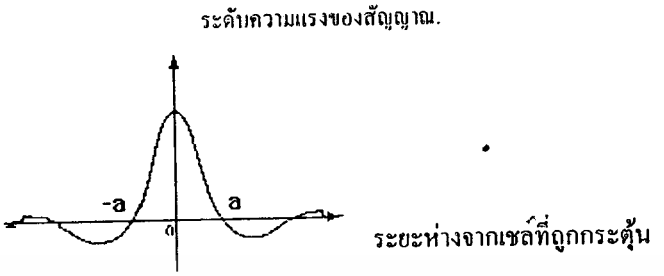
I_i สมาชิกเวคเตอร์ข้อมูลเข้าตัวที่ 0,1...,n

2 . ชั้นคอมเพททิทีฟ (Competitive Layer)

เซลประสาทในโครงข่ายประสาทสิ่งมีชีวิตจะเรียงกันแบบสองมิติ เมื่อเซลใดได้รับสัญญาณการกระตุ้น เซลที่อยู่รอบ ๆ ก็จะถูกกระตุ้นด้วย โดยความแรง (strenght) การกระตุ้นจะลดลงตามลำดับความห่างระหว่างเซลนั้น ๆ กับเซลที่รับสัญญาณการกระตุ้นโดยตรง ซึ่งแสดงได้ดังในภาพที่ 13 ที่เรียกว่า ."Maxican Hat Function" โดยเซลประสาทมนุษย์ที่อยู่ห่างจากเซลที่ถูกกระตุ้นโดยตรงในรัศมี 50 ถึง 100 ไมโครเมตรจะได้รับสัญญาณการกระตุ้นที่เป็นบวกด้วยความแรงที่ลดหลั่นกันไปตามระยะทาง ส่วนเซลที่อยู่ในรัศมี 200 ถึง 500 ไมโครเมตรจะได้รับสัญญาณการกระตุ้นที่เป็นลบ และในรัศมีไกลออกไปอีกก็จะกลับมีค่าสัญญาณการกระตุ้นที่เป็นบวกขึ้นมาอีกเล็กน้อย ซึ่งการนำมาประยุกต์ใช้ในโครงข่ายประสาทเทียมนี้ จะสมมุติให้โหนดข้างเคียงได้รับเฉพาะสัญญาณที่มีค่าเป็นบวกเท่านั้น ค่าสัญญาณส่วนที่เป็นลบและเป็นบวกเล็กน้อยในระยะที่ห่างจากโหนดที่ถูกกระตุ้นมาก ๆ จะไม่คิด (จากภาพที่ 13 โครงข่ายประสาทเทียมจะใช้สัญญาณการกระตุ้นจากช่วง a ถึง -a เท่านั้น)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 16



ที่ระยะห่างจากเซลที่ถูกกระตุ้นเท่ากับ 0 เป็นตำแหน่งเซลที่ถูกกระตุ้น

แสดงสัญญาณการกระตุ้นแบบ Maxican Hat Function [31]

ชั้นคอมเพททิทีฟของโครงข่ายประสาทเทียมได้เลียนแบบโครงสร้างเซลประสาทของสมองที่กล่าวในข้างต้น โดยมีโหนดคอมเพททิทีฟวางเรียงกันระนาบสองมิติ ซึ่งในงานวิจัยนี้กำหนดให้เป็นสี่เหลี่ยมจัตุรัส เมื่อโหนดคอมเพททิทีฟรับข้อมูลเข้าจากชั้นข้อมูลเข้าแล้ว จะนำข้อมูลไปประมวลผลหาโหนดที่จะเป็นตัวแทนข้อมูลเข้าด้วยสมการที่ 3.2 โดยเลือกโหนดที่ค่าข้อมูลเข้าและค่าน้ำหนักที่เชื่อมกับโหนดนั้นแตกต่างกันน้อยที่สุด และเรียกโหนดนี้ว่า โหนดชนะ (winner node) แล้วปรับค่าน้ำหนักให้โหนดชนะและโหนดข้างเคียง ที่อยู่ในขอบเขตพื้นที่ข้างเคียงที่กำหนดเท่านั้นด้วยสมการที่ 3.3 ดังแสดงการปรับค่าน้ำหนักในเวคเตอร์ขนาดหนึ่งหน่วยได้ในภาพที่ 17 ซึ่งจะกำหนดอัตราการเรียนรู้ (α) เริ่มต้นในช่วงเริ่มการฝึกไว้สูงแล้วจึงค่อย ๆ ลดค่าลง และขอบเขตของโหนดข้างเคียงก็กำหนดได้หลายลักษณะ เช่น เป็นกรอบสี่เหลี่ยมจัตุรัส (rectangular grid) ที่เป็นชั้น ๆ ซ้อนกันรอบ ๆ โหนดชนะ เป็นกรอบสี่เหลี่ยมขนมเปียกปูน (diamond grid) หรือเป็นกรอบหกเหลี่ยม (hexagonal grid) ดังแสดงในภาพที่ 18 ในทำนองเดียวกันเมื่อเริ่มต้นการฝึกจะกำหนดให้ขอบเขตของโหนดข้างเคียงสูง (มีจำนวนชั้นมาก) แล้วจึงค่อย ๆ ลดขอบเขตนี้ลงในระหว่างการฝึก

$$\|I - W_{WINNER}\| = \min_{c \in A} \|I - W_c\| \quad \text{.....(3.2)}$$

$$W_c^{k+1} = W_c^k + \alpha\left(I - W_c^k\right), \quad \text{โดย } c \in N_c, winner \text{.....(3.3)}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

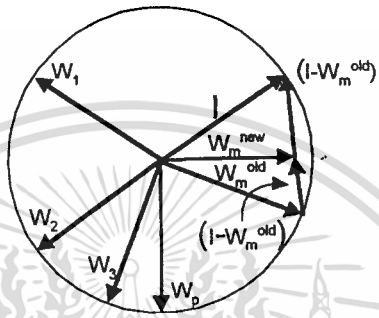
I : เวคเตอร์ข้อมูลเข้า

W_k : เวคเตอร์น้ำหนักของโหนดที่เป็นโหนดข้างเคียงโหนดชนะที่เวลา k

W_{k+1} : เวคเตอร์น้ำหนักของโหนดที่เป็นโหนดข้างเคียงโหนดชนะที่เวลา k+1

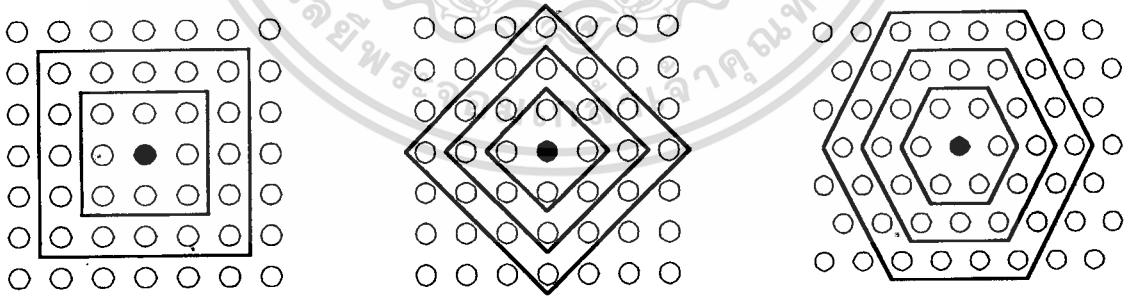
winner : โหนดชนะ

ภาพที่ 17



แสดงการปรับน้ำหนักในภาพเวคเตอร์น้ำหนักและเวคเตอร์ข้อมูลเข้าขนาด 1 หน่วย [3]

ภาพที่ 18



18.1) กรอบสี่เหลี่ยมจัตุรัส

18.2) กรอบสี่เหลี่ยมขนมเปียกปูน

18.3) กรอบหกเหลี่ยม

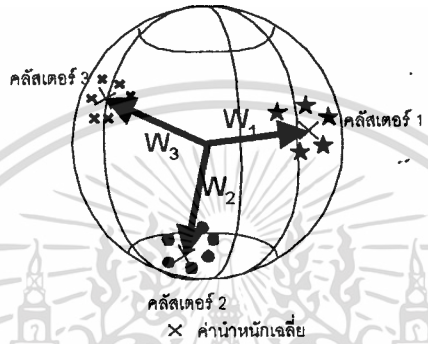
● แทน โหนดชนะ

แสดงขอบเขตโหนดข้างเคียงในลักษณะต่าง ๆ [3]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อแยกแยะตัวอักษรได้หมดค่าน้ำหนักแต่ละโหนดคือ ค่าน้ำหนักเฉลี่ยที่แทนตัวอักษรซึ่งอยู่ในคลัสเตอร์ (cluster) เดียวกันดังแสดงในภาพที่ 19 ซึ่งตัวอักษรแต่ละแบบจะอยู่คนละคลัสเตอร์กัน แต่ตัวอักษรหนึ่งแบบอาจมีหลายคลัสเตอร์ได้

ภาพที่ 19



แสดงค่าน้ำหนักของโครงข่ายหลังการฝึกซึ่งเป็นน้ำหนักเฉลี่ยของกลุ่มข้อมูลตัวอย่างแต่ละกลุ่มที่ใช้ฝึก [3]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปอัลกอริทึมที่ 1 ใช้สอน SOM

Step 0. initial weight, neighborhood region (N_c=5), start and stop learning rate (α_s=0.6,α_e=0.001)

Step 1. while not classify or stop condition False

Step 2.while $N_c \geq 0$

Step 3. while $\alpha \geq \alpha_e$

Step 4. for each training input I

Step 5. apply input to input node

Step 6. find winner node (eq. 3.2)

Step 7. update weight for winner and neighbor node (eq. 3.3)

Step 8 reduce $\alpha = \alpha / 2$

Step 9. reduce $N_c = N_c - 1$

Step 10. stop condition TRUE

N_c : จำนวนชั้นของโหนดข้างเคียงโหนด c

α_s, α_e : อัตราการเรียนรู้เริ่มต้นและสุดท้ายตามลำดับ

α : อัตราการเรียนรู้ขณะใด ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

โครงข่ายประสาทเทียมแบบ Adaptive Self-Organizing Maps

(Adaptive Self-Organizing Maps Neural Network)

สถาปัตยกรรม Adaptive Self-Organizing Maps (ASOM)

สถาปัตยกรรม ASOM เสนอโดย Bernd Fritzke [1] มีหลักการเดียวกันกับ SOM คือ จะทำการแมป (mapping) ข้อมูลเข้าหลายมิติไปยังโครงข่าย A ที่มีจำนวนมิติต่ำกว่า โดยข้อมูลที่อยู่ใกล้เคียงกันในหลายมิติ แมปเป็นมิติที่ต่ำกว่าแล้วข้อมูลที่อยู่ใกล้เคียงกันนี้ก็ยังอยู่ใกล้เคียงกันเช่นเดิม ซึ่งเป็นการลดมิติของข้อมูลโดยยังรักษาความสัมพันธ์ของข้อมูลไว้ นอกจากนี้จากการแมปยังทำให้สามารถรู้การกระจายข้อมูลในหลายมิติ คือ บริเวณโครงข่าย A ที่มีกลุ่มข้อมูลเข้าแมป (ข้อมูลเข้าไปตก) ไปจำนวนมาก (มี probability density สูง) การกระจายข้อมูลของกลุ่มข้อมูลนี้ในขณะอยู่ในหลายมิติก็จะมีความหนาแน่นสูง

SOM เป็นโครงข่ายที่มีโครงสร้างและจำนวนโหนดคงที่ตลอดการฝึก ขอบเขตของโหนดที่เป็นโหนดข้างเคียงโหนดใด ๆ จึงสามารถกำหนดได้อย่างแน่นอน เนื่องจากแต่ละโหนดในโครงข่ายจะมีตำแหน่งเฉพาะที่ไม่เปลี่ยนแปลงในระหว่างการฝึก ซึ่งบางตำแหน่งอาจไม่เหมาะสมที่จะเป็นตัวแทนข้อมูลใด ๆ เลย การใช้ข้อมูลหลากหลายแบบในการฝึกโครงข่าย SOM ให้เรียนรู้จะต้องใช้โครงข่ายที่มีขนาดใหญ่เพื่อให้โครงข่ายมีจำนวนโหนดที่เหมาะสมมากเพียงพอให้ข้อมูลทั้งหมดเลือกเป็นตัวแทนได้ ในขณะเดียวกันก็จะมีโหนดที่ไม่ได้เป็นตัวแทนข้อมูลใดเลยเป็นจำนวนมาก แต่ในโครงข่าย ASOM จะมีโครงสร้างที่ยืดหยุ่นกว่า SOM คือ สามารถปรับเปลี่ยนโครงสร้างและจำนวนโหนดในโครงข่ายได้ระหว่างการฝึกโดยการเพิ่มและลบโหนด (เพิ่มโหนดในตำแหน่งที่เหมาะสมที่จะช่วยให้มีการแยกแยะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

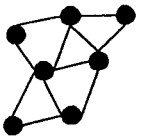
ข้อมูลได้ดีขึ้น และลบโหนดที่อยู่ในตำแหน่งที่ไม่เหมาะสมออกจากโครงข่ายได้) ทำให้จำนวนโหนดข้างเคียงโหนดใด ๆ มีการเพิ่มและลดตามไปด้วย ASOM จึงต้องมีหลักเกณฑ์การเชื่อมต่อระหว่างโหนดในระหว่างการปรับเปลี่ยนโครงข่าย เพื่อช่วยในการกำหนดโหนดข้างเคียง Bernd Fritzke ได้เสนอให้โครงข่าย ASOM มีการเชื่อมต่อกันในลักษณะโครงสร้างสามเหลี่ยม ที่จะเรียกว่า ซิมเพลค (simplex) ซึ่งถือเป็นโครงสร้างพื้นฐานของโครงข่าย และสามารถกำหนดให้ซิมเพลคมีมิติเท่าใด (k มิติ) ก็ได้ เช่น ซิมเพลค 1 มิติ จะประกอบด้วยโหนด 2 โหนดเชื่อมต่อกัน ซิมเพลค 2 มิติ จะประกอบด้วยโหนด 3 โหนดที่เชื่อมต่อกันเป็นสามเหลี่ยม และซิมเพลค 3 มิติ จะประกอบด้วยโหนด 4 โหนดเชื่อมต่อกันเป็นทรงเหลี่ยมสี่หน้า (tetrahedral) เป็นต้น ซึ่งแต่ละซิมเพลค ใน k มิติใด ๆจะประกอบด้วย k+1 โหนดที่ทุกโหนดมีการเชื่อมต่อกัน และมีเส้นเชื่อมระหว่างโหนดเท่ากับ k+1 เส้น เส้นเชื่อมนี้แสดงถึงความสัมพันธ์ในการเป็นโหนดข้างเคียงซึ่งกันและกัน (neighborhood relations) และความยาวของเส้นเชื่อมก็จะแสดงถึงความต่างของค่าน้ำหนักระหว่างโหนดทั้งสองที่เชื่อมต่อกันนั้น ซึ่งแสดงตัวอย่างโครงข่ายที่มีโครงสร้างซิมเพลคใน 1, 2 และ 3 มิติได้ดังภาพที่ 20 การกำหนดโครงข่ายให้มีโครงสร้างลักษณะสามเหลี่ยมทำให้ระหว่างการฝึกสามารถปรับเปลี่ยนโครงสร้างโครงข่ายได้ง่าย และอัตราการเพิ่มขึ้นของโหนดก็ไม่มากเกินไปคือ เป็นสัดส่วนโดยตรงกับมิติของซิมเพลค (kมิติ) ซึ่งเมื่อเทียบกับโครงข่ายที่มีการเชื่อมต่อกันในลักษณะโครงสร้างสี่เหลี่ยม เช่น ไฮเปอร์คิวบ์ (hypercube) จะมีจำนวนโหนดเพิ่มขึ้นแบบเอกโปเนนเชียล (exponential) เท่ากับ 2^k และการปรับโครงสร้างของโครงข่ายก็มีความซับซ้อนกว่า

ในงานวิจัยนี้กำหนดให้โครงสร้างซิมเพลคมี 2 มิติ ซึ่งโครงข่ายจะมีลักษณะดังตัวอย่างในภาพที่ 20.2 และการเชื่อมต่อของโหนดจะอยู่ในลักษณะสามเหลี่ยมดังภาพที่ 21 ซึ่งเป็นโครงสร้างพื้นฐานของโครงข่าย 2 มิติ ที่จะเป็นกลุ่มของโหนดซึ่งเชื่อมต่อกันทุกโหนดในกลุ่ม

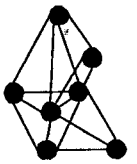
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 20

20.1) เชื่อมต่อกันเป็นเส้น 1 มิติ (k = 1)



20.2) เชื่อมต่อกันเป็นสามเหลี่ยม 2 มิติ (k = 2)



20.3) เชื่อมต่อกันเป็นทรงเหลี่ยมสี่หน้า 3 มิติ (k = 3)

แสดงการเชื่อมต่อระหว่างโหนดในโครงข่ายที่มีโครงสร้างซิมเพลคเป็น 1, 2 และ 3 มิติ [1]

ภาพที่ 21

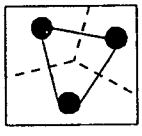
แสดงโครงสร้างพื้นฐานในโครงข่าย 2 มิติ [1]

ในโครงสร้างโครงข่ายแบบ 2 มิติ พื้นที่ของโครงข่ายจะแบ่งออกเป็นพื้นที่ย่อย ๆ ของโหนดต่าง ๆ ที่เรียกพื้นย่อยนี้ว่า พื้นที่โวโรนอย (voronoi region) ซึ่งจะใช้ในการคำนวณหาสัญญาณการตกเริ่มต้นให้โหนดใหม่และคำนวณหาโหนดที่ควรจะถูกลบออกจากโครงข่าย การหาพื้นที่โวโรนอย์ของโหนดต่าง ๆ ทำได้โดยการกำหนดเส้นแบ่งให้ตั้งฉากกับเส้นเชื่อมระหว่างสองโหนดคู่ใด ๆ (เรียกเส้นแบ่งนี้ว่า โวโรนอยเทสเซลเลชั่น, voronoi tessellation) ดังแสดงในภาพที่ 22 เส้นแบ่งรอบ ๆ โหนดใด ๆ เมื่อลากตัดกันจะเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลายเหลี่ยมที่เป็นพื้นที่โวโรนอยของโหนดที่พื้นที่นั้นล้อมรอบอยู่ ดังแสดงตัวอย่างในภาพที่ 23

ภาพที่ 22

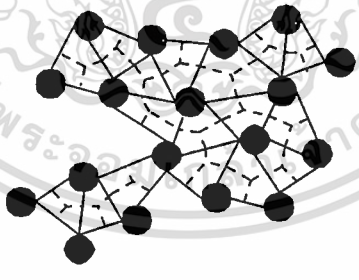


—— เส้นเชื่อมระหว่างโหนด

------ เส้นแบ่งขอบเขต

แสดงเส้นแบ่งขอบเขตในแต่ละซิมเพลค [1]

ภาพที่ 23



แสดงภาพหลายเหลี่ยมเส้นประที่เป็นพื้นที่โวโรนอยของโหนด
ที่ภาพหลายเหลี่ยมนั้นล้อมรอบอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การฝึกโครงข่ายแบบ ASOM

หลักการฝึกและการจดจำ ASOM ก็เป็นเช่นเดียวกับ SOM คือ มีการปรับน้ำหนักโหนดชนะและโหนดข้างเคียงตามสมการที่ 3.3 แต่ใน ASOMโหนดข้างเคียงเป็นโหนดที่เชื่อมกับโหนดชนะโดยตรง (direct neighbor, N_c เท่ากับ 1) เท่านั้น ความแรงของการปรับค่าน้ำหนัก (strength of adaptation) ใน SOM จะค่อย ๆ ลดลง (ค่า α ค่อย ๆ ลดลง) และขอบเขตของโหนดข้างเคียงที่กำหนดให้มีขนาดกว้างในตอนเริ่มต้นก็จะลดลงในระหว่างการฝึก ส่วนใน ASOM ความแรงในการปรับค่าน้ำหนักของโหนดชนะจะมากกว่าโหนดข้างเคียง ($\alpha_{winner} > \alpha_{neighbor}$) โดยในระหว่างการฝึกจะมีการนับสัญญาณที่แต่ละโหนดถูกเลือกเป็นโหนดชนะ โดยนับค่าสัญญาณเพิ่มขึ้นอีกหนึ่งเมื่อโหนดนั้นถูกเลือกเป็นโหนดชนะ ขณะเดียวกันก็ลดค่าสัญญาณนี้ของทุก ๆโหนดในโครงข่ายลงเท่ากับ β เท่าของค่าเดิม (ในที่นี้ $\beta = 0.05$) ดังในสมการที่ 4.1 และ 4.2 ตามลำดับ

$$\Delta\tau_{winner} = 1 \quad \text{.....(4.1)}$$

$$\Delta\tau_c = -\beta\tau_c, \ (c \in A) \quad \text{.....(4.2)}$$

$\Delta\tau_{winner}$ ค่าสัญญาณที่เปลี่ยนแปลงในโหนดชนะ

$\Delta\tau_c$ ค่าสัญญาณที่เปลี่ยนแปลงในโหนด c

τ_c ค่าสัญญาณการตกของตัวอักษรในโหนด c

β อัตราการลดลงของสัญญาณการตกของตัวอักษร (เป็นค่าคงที่)

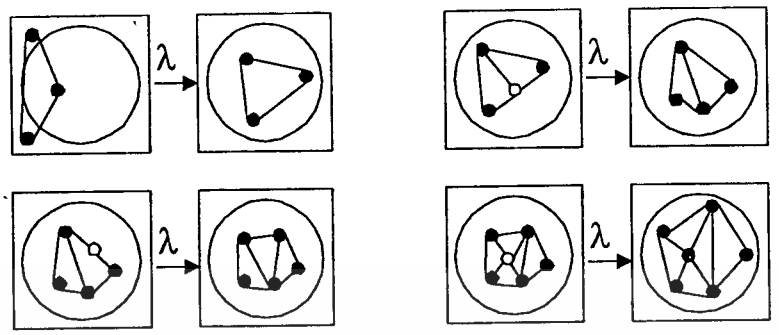
A เซตของโหนดทั้งหมดในโครงข่าย A

c โหนดใด ๆ ที่อยู่ในโครงข่าย A

ค่าน้ำหนักที่สุ่มในตอนแรกอาจมีบางค่าอยู่นอกขอบเขตการกระจายของข้อมูลที่ใช้ฝึก เมื่อผ่านการฝึกไประยะหนึ่งจะถูกปรับเข้าสู่ขอบเขตข้อมูลฝึก ดังตัวอย่างในภาพที่ 24 ที่สมมุติให้ขอบเขตการกระจายข้อมูลฝึกอยู่ในกรอบวงกลม แล้วเมื่อฝึกไประยะเวลาหนึ่ง (เช่น ระยะเวลาเท่ากับ λ) จะมีการเพิ่มโหนดใหม่ให้โครงข่าย โหนดจึงเพิ่มขึ้นเรื่อย ๆ และโหนดที่เพิ่มขึ้นก็ยังมีลักษณะเช่นเดียวกับ SOM คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 24



■ แทนโหนดเก่า ○ แทนโหนดใหม่ที่เพิ่มเข้าไป

แสดงการเพิ่มโหนดในโครงข่ายสองมิติเมื่อฝึกไประยะเวลาหนึ่ง (λ) [1]

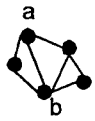
1.โหนดที่เพิ่มที่อยู่ใกล้เคียงกั น (เป็นโหนดข้างเคียงกัน) จะมีน้ำหนักใกล้เคียงกัน (ข้อมูลเข้าที่อยู่ใกล้เคียงกันในหลายมิติจะตกยังโหนดที่ใกล้เคียงกันใน ASOM)

2. ในบริเวณนี้ข้อมูลเข้ามีค่า probability density สูง จะมีจำนวนโหนด (ที่แตกออกมาในบริเวณนั้น) มาก

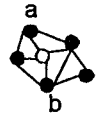
จากตัวอย่างในภาพที่ 25 โหนดที่มีจำนวนตัวอักษรตกมากที่สุด (โหนด cmax) เป็นตำแหน่งที่ควรเพิ่มโหนดใหม่ จะเพิ่มโหนดใหม่ระหว่างโหนดนี้ (โหนด cmax) กับโหนดที่มีค่าน้ำหนักห่างจากโหนดนี้มากที่สุด (โหนด far) โดยคำนวณค่าความห่างได้ตามสมการที่ 4.3 เมื่อเพิ่มโหนดใหม่จะต้องมีการกำหนดพารามิเตอร์ให้โหนดใหม่ด้วย โดยโหนดใหม่จะมีค่าน้ำหนักเป็นค่าเฉลี่ยระหว่างสองโหนดคือ โหนด cmax และโหนด far ดังคำนวนได้ในสมการที่ 4.4 และโหนดใหม่ที่เพิ่มจะต้องยังรักษาโครงสร้างซิมเพลกไว้ด้วย โดยโหนดข้างเคียงของโหนดใหม่คือ โหนด cmax และโหนด far รวมทั้งโหนดทุกโหนดที่อยู่ในซิมเพลกเดียวกับโหนด cmax และโหนด far

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 25



25.1) โครงข่ายประสาทก่อนการเพิ่มโหนด



25.2) โครงข่ายประสาทหลังการเพิ่มโหนด

● แทนโหนดเก่า ○ แทนโหนดใหม่ที่เพิ่มเข้าไป

a โหนดที่มีค่า normalized probability density มากที่สุด

b โหนดที่น้ำหนักต่างจากโหนด a มากที่สุด

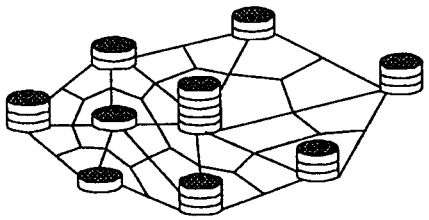
แสดงการปรับโครงสร้างโครงข่ายหลังการเพิมโหนดใหม่ [1]

นอกจากนี้จะต้องกำหนดค่าสัญญาณการตกเริ่มต้นให้โหนดใหม่ด้วย เมื่อมีการเพิ่มโหนดใหม่พื้นที่โวโรนอยด์ ของแต่ละโหนดที่อยู่รอบ ๆ โหนดใหม่ (ซึ่งเป็นโหนดข้างเคียงโหนดใหม่) ก็จะน้อยลง เนื่องจากพื้นที่บางส่วนถูกปรับเป็นพื้นที่โวโรนอยด์ของโหนดใหม่ สัญญาณเริ่มต้นของโหนดใหม่จะเท่ากับสัญญาณที่ตกในพื้นที่โวโรนอยด์ของโหนดใหม่ ซึ่งคำนวณได้โดยคำนวนหาพื้นที่โวโรนอยด์ของแต่ละโหนดข้างเคียงโหนดใหม่ที่ลดลงไปหลังการเพิ่มโหนดใหม่แล้วหาค่าสัญญาณการตกของแต่ละโหนดข้างเคียงโหนดใหม่ที่ลดลงซึ่งจะเป็นสัดส่วนโดยตรงกับพื้นที่โวโรนอยด์ที่เปลี่ยนแปลงไปเทียบกับสัญญาณการตกก่อนที่จะทำการเพิ่มโหนดใหม่ ดังคำนวณในสมการที่ 4.7 ค่าสัญญาณการตกเริ่มต้นของโหนดใหม่จะเท่ากับผลรวมสัญญาณของแต่ละโหนดข้างเคียงโหนดใหม่ที่ลดลงหลังการเพิ่มโหนด (โดยไม่คิดเครื่องหมายลบ) ดังคำนวณในสมการที่ 4.8 และแสดงตัวอย่างในภาพที่ 26

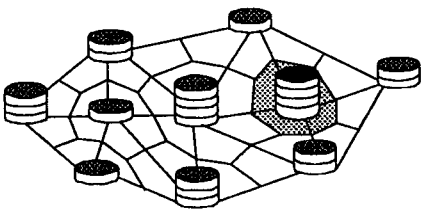
(การคำนวณหาพื้นที่โวโรน์อยด์ เนื่องจากพื้นที่โวโรนอยด์เป็นภาพหลายเหลี่ยมที่มีจำนวนเหลี่ยมและมุมในแต่ละเหลี่ยมไม่แน่นอนและมีการเปลี่ยนแปลงได้ขึ้นกับโหนดที่เปลี่ยนแปลงไป การคำนวณหาพื้นที่โดยตรงจึงมีความซับซ้อน เพื่อให้ง่ายต่อการคำนวณจึงจะคิดว่าโหนดใด ๆ มีพื้นที่โวโรนอยด์ลักษณะเป็นพื้นที่สี่เหลี่ยมจัตุรัสโดยค่าความยาวแต่ละด้านของสี่เหลี่ยมจัตุรัสคิดจากค่าเฉลี่ยของด้านในภาพหลายเหลี่ยมดังคำนวณได้จากสมการที่ 4.5 พื้นที่โวโรนอยด์ของโหนดใด ๆ ก็จะเท่ากับความยาวของด้านสี่เหลี่ยมจัตุรัสยกกำลังสองดังในสมการที่ 4.6)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 26



26.1) ก่อนเพิ่มโหนดใหม่



26.2) หลังเพิ่มโหนดใหม่

พื้นที่โวโรนอยด์ของโหนดที่เพิ่มใหม่ แทนความถี่การตกเท่ากับ 1

แทนเส้นเชื่อมระหว่างโหนด -------- แทนเส้นแบ่งโวโรนอยด์

แสดงการเปลี่ยนแปลงความถี่การตก และพื้นที่โวโรนอยด์เมื่อเพิ่มโหนดให้โครงข่าย [1]

$$\left\|W_{c\max} - W_{far}\right\| = \underset{i \in n_{c\max}}{MAX} \left\|W_{c\max} - W_i\right\| \quad \text{.....(4.3)}$$

$$W_{new} = \frac{\left(W_{c\max} + W_{far}\right)}{2} \quad \text{.....(4.4)}$$

$$\tilde{l}_c = \frac{1}{card\left(N_c\right)} \sum_{i \in N_c} \left\|W_c - W_i\right\| \quad \text{.....(4.5)}$$

$$\left|F_c\right| = \left(\tilde{l}_c\right)^n \qquad , n = 2 \quad \text{.....(4.6)}$$

$$\Delta\tau_c = \frac{\left|F_c^{new}\right| - \left|F_c^{pld}\right|}{\left|F_c^{old}\right|} \tau_c \qquad , c \in N_{new} \quad \text{.....(4.7)}$$

$$\tau_{new} = -\sum_{c \in N_{new}} \Delta\tau_c \quad \text{.....(4.8)}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

W	เวคเตอร์น้ำหนัก
new	โหนดใหม่
$cmax$	โหนดที่มีค่าสัญญาณการตกมากที่สุด
far	โหนดที่มีค่าผลต่างกำลังสองของน้ำหนักห่างจากโหนด cmax มากที่สุด
τ_{new}	ค่าสัญญาณการตกเริ่มต้นของโหนดใหม่
$\Delta\tau_c$	ผลต่างระหว่างค่าสัญญาณการตกของโหนด c ก่อนการเพิ่มโหนดใหม่ และค่าสัญญาณการตกของโหนด c หลังการเพิ่มโหนดใหม่
N_c	โหนดข้างเคียงใด ๆ ของโหนด c
card (N_c)	จำนวนโหนดข้างเคียงโหนด c
n	มิติของโครงข่าย (ในงานวิจัยนี้เท่ากับ 2 มิติ)
$\tilde{l}_c$	ความยาวด้านของสี่เหลี่ยมจัตุรัสที่ล้อมรอบโหนด c ซึ่งเป็นค่าเฉลี่ยผลต่างของน้ำหนักระหว่างโหนด c และโหนดข้างเคียงโหนด c
$\lvert F_c^{old}\rvert$	ขนาดพื้นที่โวโรนอยด์ของโหนด c ก่อนการเพิ่มโหนดใหม่
$\lvert F_c^{old}\rvert$	ขนาดพื้นที่โวโรนอยด์ของโหนด c หลังการเพิ่มโหนดใหม่

หลังจากเพิ่มโหนดใหม่ให้โครงข่ายแล้ว หากโหนดใดมีโอกาสที่จะถูกเลือกเป็นโหนดชนะน้อยมากก็จะทำการลบโหนดนั้นออก โดยจะคำนวณหาสัดส่วนระหว่างสัญญาณการตกในโหนดนั้น ๆ ต่อสัญญาณข้อมูลทั้งหมดที่ใช้ฝึก ดังสมการที่ 4.9 และเทียบสัดส่วนนี้ต่อหน่วยพื้นที่โวโรนอยด์ของโหนดนั้น ๆ ซึ่งจะเรียกว่า probability density ดังคำนวณในสมการที่ 4.10 ค่าที่ได้นี้จะนำไปนอร์มอนไลซ์ให้ค่า normalised probability density ดังคำนวณในสมการที่ 4.11 โหนดที่ควรจะถูกลบคือ โหนดที่มีค่า normalised probability density น้อยกว่าค่าเทรสโฮลด์ที่กำหนด (threshold, $\eta = 0.09$)

$$h_c = \frac{\tau_c}{\sum_{i \in A} \tau_i} \quad \text{.....(4.9)}$$

$$\tilde{p}_c = \frac{h_c}{|F_c|} \quad \text{.....(4.10)}$$

$$\hat{p} = \tilde{p} \sum_{c \in A} |F_c| \quad \text{.....(4.11)}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

h_c relative signal frequency ของโหนด c

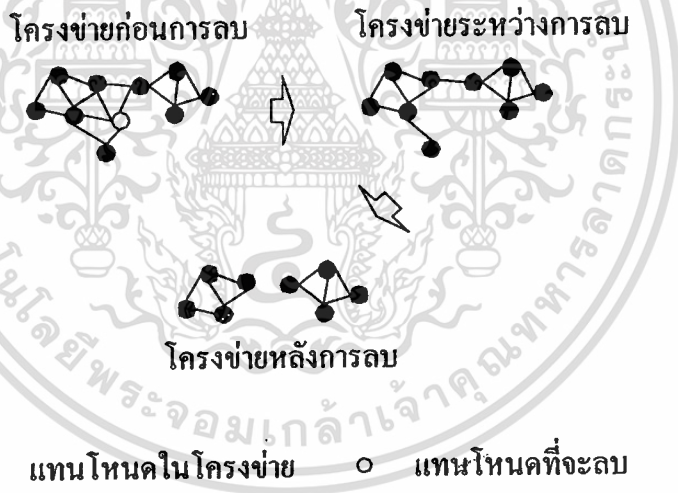
$\tilde{p}_c$ probability density ของโหนด c

$\hat{p}_c$ normalised probability density ของโหนด c

A เซตของโหนดทั้งหมดในโครงข่าย

เช่นเดียวกับการเพิ่มโหนดโครงข่ายหลังจากลบโหนดออกจะต้องยังคงโครงสร้างซิมเพลคไว้ โดยโหนดในโครงข่ายที่เหลือหลังการลบที่ไม่ได้อยู่ในซิมเพลคใดเลยต้องถูกลบออกด้วย เพื่อว่า หากมีการเพิ่มโหนดใหม่ให้โครงข่าย โหนดใหม่จะได้สามารถสร้างโครงสร้างเชื่อมต่อกับโหนดในโครงข่ายได้อย่างเหมาะสมดังตัวอย่างในภาพที่ 4.8

ภาพที่ 27



แสดงการปรับโครงสร้างโครงข่ายประสาทหลังการลบโหนด [1]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการฝึกที่กล่าวในบทนี้ก็สรุปเป็นขั้นตอนการฝึกได้ดังในอัลกอริทึมที่ 2

อัลกอริทึมที่ 2 ใช้ฝึก ASOM

Step 1. initial weight, learning rate of winner and neighbor node and number of node = 3 that all nodes connect to each other to be a triangular (simplex)

Step 2. select only one pattern of each character in training set to be inputs

Step 3. while classify False , do step 4-11

Step 4. for each training input I do step 5-8

Step 5. apply input to input node

Step 6. find winner node (eq. 3.2)

Step 7. update weight of winner and neighbor node(eq. 3.3)

Step 8 increase signal frequency of winner (eq. 4.1) and decrease all signal frequency (eq.4.2)

Step 9. insert new node

Step 9.1 find max signal node

Step 9.2 find node that weight is maximum distance from weight of max signal node(eq. 4.3)

Step 9.3 insert new node between max signal node and max distance node

Step 9.4 initial signal of new node (eq.4.8)

Step 9.5 initial weight of new (eq. 4.4)

Step 9.6 initial neighbornode for new node

Step 9.7 update neighbor of new_node's neighbor

Step 9.8 update signal of new_node's neighbor (eq.4.7)

Step 10. delete node

Step 10.1 calculate $\hat{p}_c$ (eq. 11)

Step 10.2 delete node which $\hat{p}_c < \eta$ ($\eta = 0.09$)

Step 11. if each node is allocated by same Thai character , classify TRUE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การทดลอง SOM

<u>จุดประสงค์การทดลอง</u>

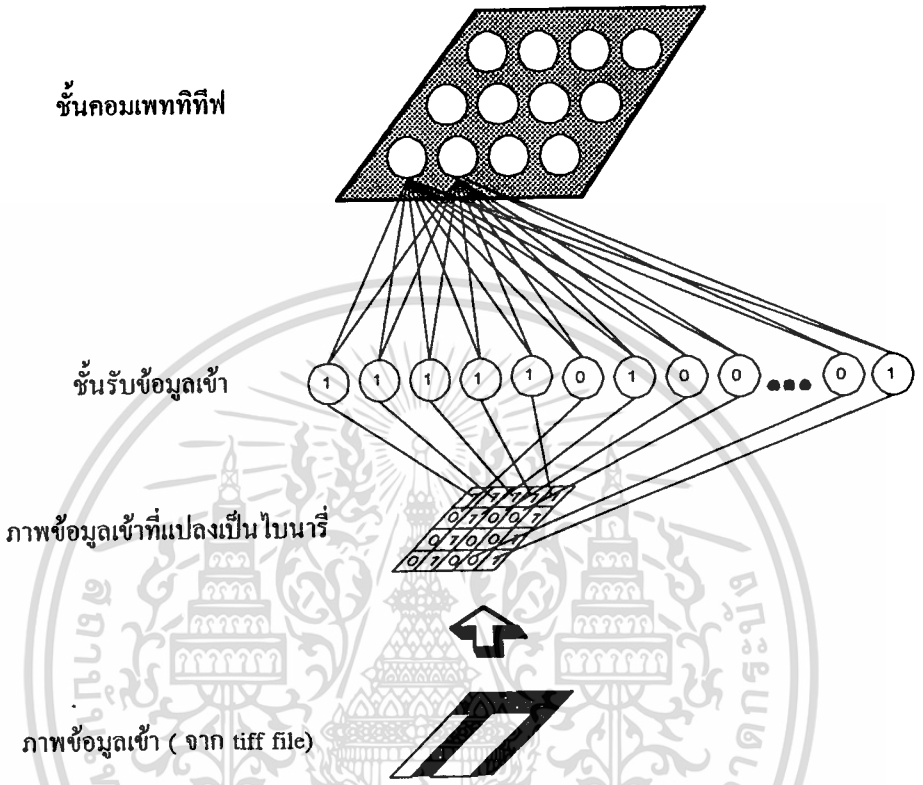
1. หาขนาดโครงข่าย SOM ที่สามารถแยกแยะตัวอักษรได้
2. หาค่าพารามิเตอร์ต่าง ๆ ที่เหมาะสมให้แก่โครงข่าย ได้แก่ ค่าอัตราการเรียนรู้ (α) ขอบเขตของโหนดข้างเคียง (N_c)
3. หาการช่วงค่าน้ำหนักเริ่มต้นที่เหมาะสมให้โครงข่าย

<u>ข้อมูลเข้าที่ใช้ในการฝึกโครงข่าย</u>

งานวิจัยนี้ใช้ตัวอักษรพิมพ์ฟอนท์ (font) AngsanaUPC ขนาด 12 พอยท์ (point) ความละเอียดการสะแกน (scan) ภาพตัวอักษรคือ 300 จุดต่อนิ้ว (dpi) และ scale 100 ในการสอนและทดสอบการจดจำโดยประกอบด้วยพยัญชนะไทย 44 ตัว เลขไทย เลขอังกฤษ า เ ไ ใ ๆ ๅ ฤ ฦ และไม้หันอากาศ รวม 75 ตัว โดยทำการแปลงภาพตัวอักษร (*.tiff) ให้เป็นไบนารี่ก่อน (0 แทนพื้นตัวอักษร และ 1 แทนเนื้อตัวอักษร) แล้วตัดตัวอักษรออกมาทีละตัวจากไฟล์ไบนารี่ แล้ววางตัวอักษรในเมตริก 2 มิติ ซึ่งในงานวิจัยนี้ให้เมตริกตัวอักษรมีขนาด 28 x 38 โดยวางตัวอักษรที่เป็นข้อมูลเข้าที่ตำแหน่งชิดบนด้านซ้ายของเมตริก ดังแสดงตัวอย่างในภาพที่ 28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 28



แสดงตัวอย่างการป้อนข้อมูลภาพตัวอักษรเข้าฝึกในโครงข่ายโดย
ภาพตัวอักษรถูกแปลงเป็นไบนารี่ที่เป็นเมตริกขนาด 5 x 4

ชุดตัวอย่างที่ใช้ฝึกมีตัวอักษร 75 แบบแบ่งเป็น 2 ชุด ชุดแรกมีตัวอักษรแบบละ 1 ตัว ชุดสองมีตัวอักษรแบบละ 20 ตัวรวม 1500 ตัว และฝึกตามอัลกอริทึมที่ 1

จากที่กล่าวข้างต้นก่อนจะใช้งานโครงข่ายจะต้องฝึกสอนโครงข่ายให้โครงข่ายมีความรู้เกี่ยวกับสิ่งที่ต้องการใช้งานก่อน ในบทนี้จะแสดงการฝึกสอนโครงข่ายให้สามารถจดจำตัวอักษรภาษาไทย ด้วยวิธี SOM โดยจะกล่าวถึง ผลการทดลอง ปัญหาและวิธีการแก้ไขปรับปรุงขั้นตอนการฝึกสอนให้โครงข่ายทำงานได้อย่างมีประสิทธิภาพสูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลอง ผลการทดลอง และการแก้ปัญหา

ในงานวิจัยนี้นอกจากการสร้างโครงข่ายประสาทเทียมที่สามารถแยกแยะตัวอักษรได้ถูกต้องแล้ว โครงข่ายยังต้องมีความเหมาะสมเพื่อสามารถทำงานได้อย่างมีประสิทธิภาพสูงสุดอีกด้วย ดังนั้นการนำโครงข่ายมาประยุกต์ใช้กับงานใด ๆ จึงจำเป็นต้องทดลองปรับเปลี่ยนพารามิเตอร์ (parameter) ต่าง ๆ ของโครงข่ายให้มีค่าที่เหมาะสมกับงานนั้น ๆ มากที่สุด ในโครงข่ายแบบ SOM พารามิเตอร์ที่สำคัญก็มีหลายพารามิเตอร์ เช่น ค่าอัตราการเรียนรู้ ขนาดโครงข่าย ขอบเขตของโหนดข้างเคียง และค่าน้ำหนัก เป็นต้น จึงได้ทดลองหาค่าที่เหมาะสมของพารามิเตอร์ต่าง ๆ เหล่านี้ ดังจะกล่าวต่อไป

1. การทดลองหาขนาดโครงข่ายที่เหมาะสม

กำหนดให้โครงข่ายเริ่มต้นมีโหนดเรียงต่อกันในลักษณะสองมิติเป็นผืนสี่เหลี่ยมจัตุรัสที่มีขนาด 10 x 10 โหนด ค่าน้ำหนักเริ่มต้นให้สุ่มค่าอยู่ระหว่าง 0 ถึง 1 ค่าอัตราการเรียนรู้เริ่มต้น (α_s) และค่าอัตราการเรียนรู้สุดท้าย (α_c) เท่ากับ 0.6 [5] และ 0.001 ตามลำดับ ส่วนขอบเขตของโหนดข้างเคียงกำหนดให้มี 5 ชั้น ($N_c = 5$) แล้วฝึกโครงข่ายตามอัลกอริทึมที่ 1 ดังกล่าวไว้ในบทที่ 3 โดยตัวอักษรที่ใช้ในการฝึกนี้มี 75 แบบ แบบละ 1 ตัว

ตารางที่ 2

ฐฐฑ๕๓ณฒฑพฏ	ฌญ			ใ	ล	ฦฦดดทฑพฐ๔	มฤ		0ค
	ตฝ	ฉ		๘	ศาผฏป9ฮอษ		ก		8๔
ฯ			๗				ส		
ช	ไ				ข	จธ6	o		
๒ฟ							43		
			๑	ย	ฒ	2ง		โ	5ๆถ
			บ	น	7				ร
									ๆ๖
									ฯแวะ

แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 10 x 10 โหนด ในการฝึกรอบที่ 1
โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0
ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3

ฎญฐ๒ฟ ๕ณฒฎ			ฮ	ผฝ	ฦ๑	ลส	ม	8๔	อดธฤคข6ย
ศฌพพ		ต		๓					9ฅจ0ถ
	ฑ								
	๗								
ซ			๔						ภกฉ
ษไ๘หฃชช					รๆ๖ฯ	๐า		ห	5
ป									
					3				
ใ,ท					4	ง	7		
ฒบ	น			โ					า261วะ

แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 10 x 10 โหนด ในการฝึกรอบที่ 20

โดยค่า $\alpha_s = 0.6$ $\alpha_e = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0

ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

ตารางที่ 4

๒	ฒ	ญ		ฐพ	ศ	ซ	ฑช	ฯไ	ใ
ฟ		พ				ผ	ษ		
ฎฎ		๕		ณณ	ฮ	ต	๑	ฑห	ฒ
		๗				๓	ล		บน
ฝ		๘		ป				ฦ	มข
		ย		โ		าร			
ส๔		อฤ	ฅดค			๐		35	4
		6	0ถ	ภกฉ	ๆ		7		ง
ฯ		ธ					2ว		ะ
๘		9จ		๖	ฯ		เ	1	า

แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 10 x 10 โหนด ในการฝึกรอบที่ 60

โดยค่า $\alpha_s = 0.6$ $\alpha_e = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0

ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2, 3 และ 4 แสดงการแมปตัวอักษรของโครงข่าย ในการฝึกรอบที่ 1, 20 และ 60 ตามลำดับ ในรอบที่ 1 เนื่องจากค่าน้ำหนักเป็นค่าสุ่มค่าจึงอาจไม่ค่อยเหมาะสมนัก ทำให้บางโหนดมีตัวอักษรหลายแบบเลือกโหนดนั้น ๆ เป็นตัวแทน (มีตัวอักษรหลายตัวตกซ้ำในโหนดเดียวกัน) เมื่อฝึกไปเรื่อย ๆ ค่าน้ำหนักก็จะถูกปรับให้มีค่าที่เหมาะสมขึ้นเรื่อย ๆ จึงทำให้การแยกแยะตัวอักษรดีขึ้น คือ โหนดที่มีตัวอักษรตกหลายตัวมีจำนวนลดลงและตัวอักษรมีการกระจายไปตกยังโหนดต่าง ๆ ดีขึ้นตามลำดับในรอบที่ 1, 20 และ 60 แต่ในรอบที่ 60 ซึ่งเป็นรอบสุดท้ายตามขั้นตอนการฝึกในอัลกอริทึมที่ 1 ก็ยังไม่สามารถแยกแยะตัวอักษรได้ทั้งหมด จึงได้ทดลองเพิ่มขนาดโครงข่ายเป็น 15 x 15 โหนด 20 x 20 โหนด และ 25 x 25 โหนด

ตารางที่ 5

ะ	า	*	ว	2	*	3	*	ๅ	ร	โ	*	A	*	ฑ
*	*	*	*	*	*	5	*	*	*	*	*	*	*	ซ
1	*	*	7	*	ง	*	*	น	บ	ฒ	ช	ฒ	*	ฟ
*	เ	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	o	*	4	*	ม	ท	*	*	ษ	*	ผ
๖	*	ๆ	*	*	*	*	*	*	*	*	*	ห	*	ต
*	*	ๅ	*	ก	*	0	*	*	ป	*	*	*	*	๓
ภ	*	*	*	ฉ	*	6	*	*	*	*	*	๑	*	*
*	*	ถ	*	ศ	*	*	*	๘	*	*	*	*	*	ฮ
9	*	*	*	B	*	๔	*	๗	*	๕	*	ฬ	*	*
*	*	ย	*	*	*	*	*	*	*	ศ	*	*	*	C
จ	*	*	*	*	ข	*	ฝ	*	*	*	*	ฐ	*	*
ธ	*	ฤ	อ	*	*	*	*	*	*	*	*	*	*	พ
*	*	*	*	*	*	*	ล	*	*	*	*	๒	*	ฒ
๘	8	*	ส	*	ฦ	*	*	*	ฎ	ฎ	*	*	*	ญ

A แทน ใ,ไ B แทน ด,ด C แทน ฌ,ณ * แทนหนดที่ไม่มีตัวอักษรใดตกเลย

แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 15 x 15 โหนด ในการฝึกรอบที่ 60
โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0
ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 6

๔	*	*	*	๗	*	ณ	ฌ	*	*	*	*	ฐ	*	*	*	ญ	*	*	*
*	*	๘	*	*	*	*	*	*	ฮ	*	*	*	*	ฒ	*	*	*	*	๒
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	ฑ	*	*	ผ	*	*	๕	*	ศ	*	พ	*	*	*	ฟ
ป	*	*	๓	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	ต	*	ษ	*	ฬ	*	*	*	*	*	*	ฎ
๙	*	*	*	*	*	*	*	*	*	*	*	*	*	ฃ	*	ซ	*	*	*
*	*	*	ส	*	ล	*	*	*	ห	*	*	*	ช	*	*	*	*	*	ฏ
8	*	*	*	*	*	*	ฦ	*	*	*	*	*	*	*	*	*	*	*	*
*	*	A	*	ย	*	*	*	*	*	ฆ	*	๑	*	*	ไ	*	*	ฝ	*
ธ	*	*	*	*	*	*	ข	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	6	*	ค	*	*	*	ท	*	*	*	*	ใ	*	*	*	*	*
จ	*	*	*	*	ด	*	*	*	บ	*	ม	*	*	*	*	*	*	ๅ	*
*	*	*	0	*	*	*	ศ	*	*	*	*	*	*	โ	*	ร	*	*	*
9	*	ถ	*	*	*	*	*	*	น	*	*	*	*	*	*	*	*	*	*
*	*	*	*	ก	*	ภ	*	*	*	*	๐	*	*	*	*	*	*	*	*
ฉ	*	ๆ	*	*	*	*	*	*	*	*	*	*	7	*	*	3	*	*	*
*	*	*	*	*	*	1	*	*	*	2	*	*	*	*	*	5	*	*	*
*	*	ฯ	*	*	*	*	*	เ	*	*	*	*	*	*	*	*	*	*	*
๖	*	*	*	ะ	*	า	*	*	*	ว	*	ง	*	4	*	*	*	*	*

A แทน อ,ฤ * แทนโหนดที่ไม่มีตัวอักษรใดตกเลย

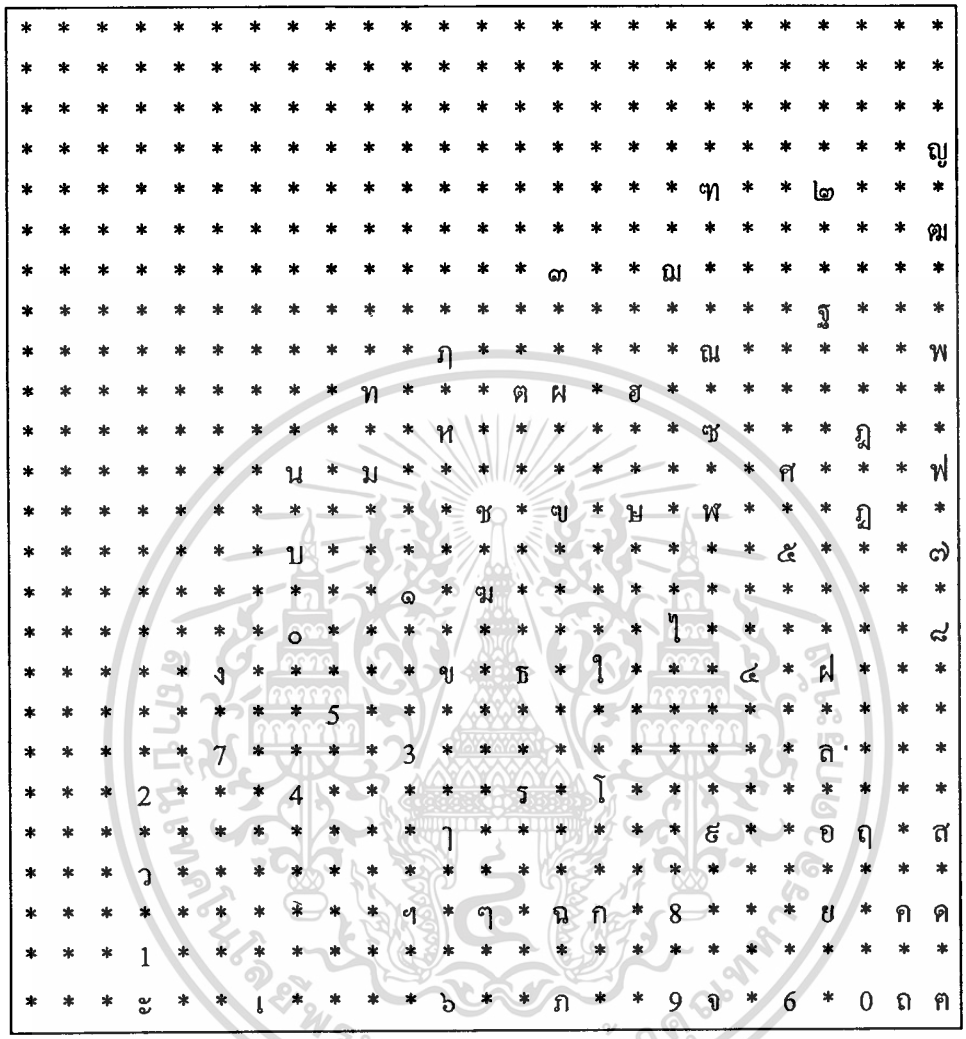
แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 20 x 20 โหนด ในการฝึกรอบที่ 60

โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0

ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 7



* แทนโหนดที่ไม่มีตัวอักษรใดตกเลย

แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 25 x 25 โหนด ในการฝึกรอบที่ 60

โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0

ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 8

๒	๒	๒	*	*	ฐ	ฐ	*	ท	พ	ฬ	*	A	ส	พ	ฬ	พ	พ	*	ฎ	ฎ	ฎ	ฎ	ฎ	ฎ
๒	๒	๒	ฐ	ฐ	*	*	พ	พ	*	ส	*	*	พ	*	*	*	ฎ	ฎ	ฎ	*	*	*	ฝ	*
*	*	*	*	*	ฐ	ฐ	*	พ	*	*	*	ฮ	*	*	*	ฟ	*	ฎ	ฎ	ฎ	*	ฝ	ฝ	ฝ
ญ	*	ฒ	ฒ	*	*	*	*	*	*	๘	ฮ	ฮ	*	ฟ	ฟ	*	*	*	*	ฎ	*	ฝ	ฝ	*
ญ	*	ฒ	ฒ	*	ฌ	ฌ	ฌ	*	*	*	*	*	*	*	*	*	๕	*	*	*	*	ฝ	ฝ	ฝ
ญ	*	ฒ	ฒ	*	*	*	ฒ	*	ธ	*	ส	ศ	ถ	*	๗	*	๕	๕	๔	*	*	*	*	*
ญ	ญ	*	*	ฒ	ฑ	ฑ	*	ฒ	*	*	*	*	*	*	*	*	๕	*	๔	*	อ	ย	อ	ย
*	*	ฌ	ฌ	*	ฑ	*	*	*	๓	๓	๓	*	๗	๗	๗	*	*	๔	๔	*	ย	*	*	*
ป	*	ฌ	ฌ	*	*	ฑ	ฑ	*	๓	๓	๓	*	๗	๗	*	๘	*	๔	๔	๔	*	ถ	*	B
ป	*	*	*	*	ซ	*	*	*	*	*	*	*	*	*	๙	*	๘	*	๔	๔	*	ฤ	*	C
ป	*	D	ท	*	*	*	ฃ	ฃ	*	8	8	*	*	๙	๙	*	๘	๘	*	*	*	*	E	ด
ป	*	ท	F	ม	ษ	*	ฒ	ฃ	*	*	8	*	*	๙	๙	*	๘	*	ธ	G	ฉ	*	H	J
*	*	ษ	*	ษ	*	*	ฃ	ฃ	ฃ	*	8	*	๙	๙	*	*	*	ย	จ	*	*	E	0	*
บ	บ	ท	*	*	ห	*	ซ	*	ฒ	*	6	*	*	*	4	L	จ	ฉ	M	*	N	P	Q	R
บ	บ	*	ษ	ม	*	*	ซ	*	S	*	6	6	6	*	*	*	*	2	จ	*	S	ล	T	อ
บ	น	ห	ห	U	น	*	ฃ	*	*	*	*	*	*	*	ร	ร	*	5	5	*	C	W	0	R
น	ห	ม	*	*	น	*	ฃ	*	6	6	ฯ	Y	*	ร	Z	*	*	5	*	*	ๆ	๗	0	อ
ห	a	น	ม	ม	*	*	*	*	*	*	*	ข	*	*	*	*	5	*	b	ๆ	ๆ	ฯ	0	c
*	*	*	*	*	*	ข	ช	ข	ข	ข	ช	ข	*	3	d	3	5	*	Y	b	*	*	*	*
ฦ	g	ศ	ฉ	๑	*	ช	ช	ช	ช	ช	*	*	*	*	2	*	*	*	h	ๆ	ฯ	*	3	ว
j	*	ฉ	k	*	ผ	*	ซ	*	*	ช	*	๖	๘	๖	๖	*	2	*	*	*	*	*	ว	7
ผ	*	*	*	*	ผ	*	ซ	ซ	*	*	*	๖	*	*	*	*	*	*	m	ะ	ะ	7	า	า
*	*	โ	ใ	*	*	*	*	*	9	9	*	๖	*	n	n	ๅ	ๅ	*	*	*	ะ	*	า	*
Z	โ	q	ไ	*	r	ไ	ไ	*	*	*	*	*	*	*	ร	*	*	5	*	*	*	*	ะ	ะ
ๅ	โ	โ	โ	r	ไ	ไ	ไ	*	9	*	๑	o	ฉ	o	ว	7	*	2	*	m	1	เ	m	า

(ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 8 (ต่อ)

A แทน ศ,ต	B แทน ภ,ภ,ก,ล,ถ	C แทน ฦ,ภ	D แทน ท,พ	E แทน ฤ,ถ
F แทน ช,บ	G แทน จ,ธ	H แทน ต,ค	J แทน ศ,ด,ค	L แทน 4,ง
N แทน ก,ด,ค	P แทน ภ,ๆ	Q แทน อ,ค	R แทน ศ,ก,ค	S แทน ก,ฤ,ถ
T แทน อ,ศ,ต,ด,ค		U แทน ท,ห	W แทน ก,ฤ	Y แทน ภ,ก
Z แทน โ,ๅ	a แทน ม,น	b แทน ๐,๑	c แทน ศ,ค,ถ	d แทน ว,7
g แทน อ,ฤ	h แทน ๐,ก	k แทน ด,ต	m แทน เ,1	n แทน 4,ง
q แทน โ,ไ	r แทน ใ,ไ	* แทนโหนดที่ไม่มีตัวอักษรใดตกเลย		

แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 25x25 โหนด ในการฝึกรอบที่ 60
โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0
ตัวอักษร 75 แบบรวม 1500ตัว

ตารางที่ 5,6 และ 7 แสดงการแมปตัวอักษรของโครงข่ายในการฝึกรอบที่ 60 จะเห็นว่าเมื่อเพิ่มขนาดโครงข่ายให้มากขึ้นจาก 10 x 10, 15 x 15, 20 x 20 และ 25 x 25 โหนดตามลำดับ จะทำให้ตัวอักษรสามารถแยกแยะได้ดีขึ้นไปตามลำดับเช่นกันดังแสดงในตารางที่ 4 ถึง 7 โดยเมื่อโครงข่ายมีขนาด 25 x25 โหนด ก็จะสามารถแยกแยะตัวอักษรได้ทั้งหมด แต่โครงข่ายขนาด 625 โหนดนี้ ใช้เพียง 75 โหนดเท่านั้นในการแยกแยะตัวอักษรในชุดฝึก ส่วนอีก 550 โหนดเป็นโหนดที่ไม่ได้ใช้ประโยชน์ใดเลย

เพื่อให้โครงข่ายสามารถแยกแยะตัวอักษรได้ดี ตัวอักษรที่ใช้ในการฝึกจึงต้องมีความหลากหลาย โดยการเลือกตัวอักษรแต่ละแบบมาหลาย ๆ ตัวทั้งที่มีความคมชัดและที่มีสัญญาณรบกวน ดังนั้นเมื่อสามารถแยกแยะตัวอักษร 75 แบบ แบบละ 1 ตัวได้ทั้งหมด จึงได้ทดลองฝึกโดยใช้ตัวอักษรแบบละ 20 ตัว รวมเป็น 1500 ตัวในการฝึกโครงข่าย

ตารางที่ 8 แสดงการแมปตัวอักษรจำนวน 1500 ตัว ในโครงข่ายขนาด 25 x 25 โหนด ซึ่งมีกลุ่มตัวอักษรที่แยกแยะได้ 31 กลุ่ม ดังนั้นจึงต้องใช้โครงข่ายที่มีขนาดใหญ่ขึ้นเมื่อมีแบบตัวอักษรหรือจำนวนตัวอักษรที่ใช้ฝึกมากขึ้น และเริ่มต้นการฝึกใหม่

2. การทดลองการกำหนดค่าน้ำหนักเริ่มต้น

จากที่กล่าวมาในข้อที่ 1. (ตารางที่ 2-8) จะกำหนดค่าน้ำหนักเริ่มต้นโดยการสุ่มให้มีค่าอยู่ระหว่าง 0 ถึง 1 ในงานวิจัยที่ผ่านมา [3] พบว่า มีการกำหนดค่าน้ำหนักให้อยู่ในช่วงแคบ ๆ และค่าจะเป็นค่ากลาง ๆ ซึ่งหากนำค่าน้ำหนักเริ่มต้นมาพลอตกราฟในแกนเวคเตอร์น้ำหนัก 2 มิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 10

ษ		6		0ถ		จ9		ภ		๖		1		า
ธ								ก		ๆ		เ		ะ
๑		ย		อฤ		ค	ค	ศ						
ล										4	ง			ว
		ฦ			ข		ร	ๅ					2	
ผ	ต													
	ฮ		ส					โ		3		5		7
					๓									
ฐ			ฌ					๘		๔				๐
		ณ				ฑ								ฯ
					พ				๗		ป		ฉ	
ญ	ฒ		พ		ศ		ซ							
					๕		ช		ห		ฒ			ธ
			ฎ				ฆ							
๒	ฟ		ฏ		ฝ		ษ		ท	ม	น	บ		ไ

แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 15 x 15 โหนด ในการฝึกรอบที่ 60 โดยค่า

$\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.4 - 0.6

ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 11

*	*	5	*	3	*	*	*	*	*	*	*	ฝ	*	*	ฎ	*	*	ฒ	*
*	*	*	*	*	*	*	ร	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	โ	*	*	*	*	*	*	ฎ	*	*	พ
*	เ	*	*	7	*	*	ๅ	*	*	*	ใ	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	ไ	*	*	ฐ	*	*	ญ
1	*	*	2	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	ง	*	4	*	ฒ	*	*	*	*	*	*	*	*	*	๒
*	*	*	ว	*	*	*	*	*	*	*	ช	*	ฃ	*	ฬ	*	*	*	*
ะ	า	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	น	*	บ	*	*	*	*	*	*	*	*	ผ	*	ฮ
*	*	*	ๆ	*	*	*	*	*	*	*	ท	*	ห	*	ต	*	*	*	*
๐	*	*	*	*	ก	*	*	*	*	*	*	*	*	*	*	*	*	*	ณ
*	*	*	*	*	ภ	*	*	ข	*	ม	*	ฤ	*	*	*	ษ	*	*	ฌ
ๆ	๖	*	ถ	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	0	*	ศ	*	*	*	*	*	*	*	*	*	*	๓	*	*	*
ฉ	*	*	*	*	*	ค	*	ฤ	อ	*	ส	*	*	*	*	*	*	*	ฑ
*	*	*	6	*	ด	*	*	*	*	*	*	*	ล	*	*	๑	*	*	*
9	*	*	*	*	*	*	ย	*	*	*	*	*	*	*	*	*	*	*	ซ
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
จ	ธ	8	*	๕	*	*	๔	*	๘	*	๗	*	*	ป	*	ฟ	*	ศ	๕

* แทนโหนดที่ไม่มีตัวอักษรใดตกเลย

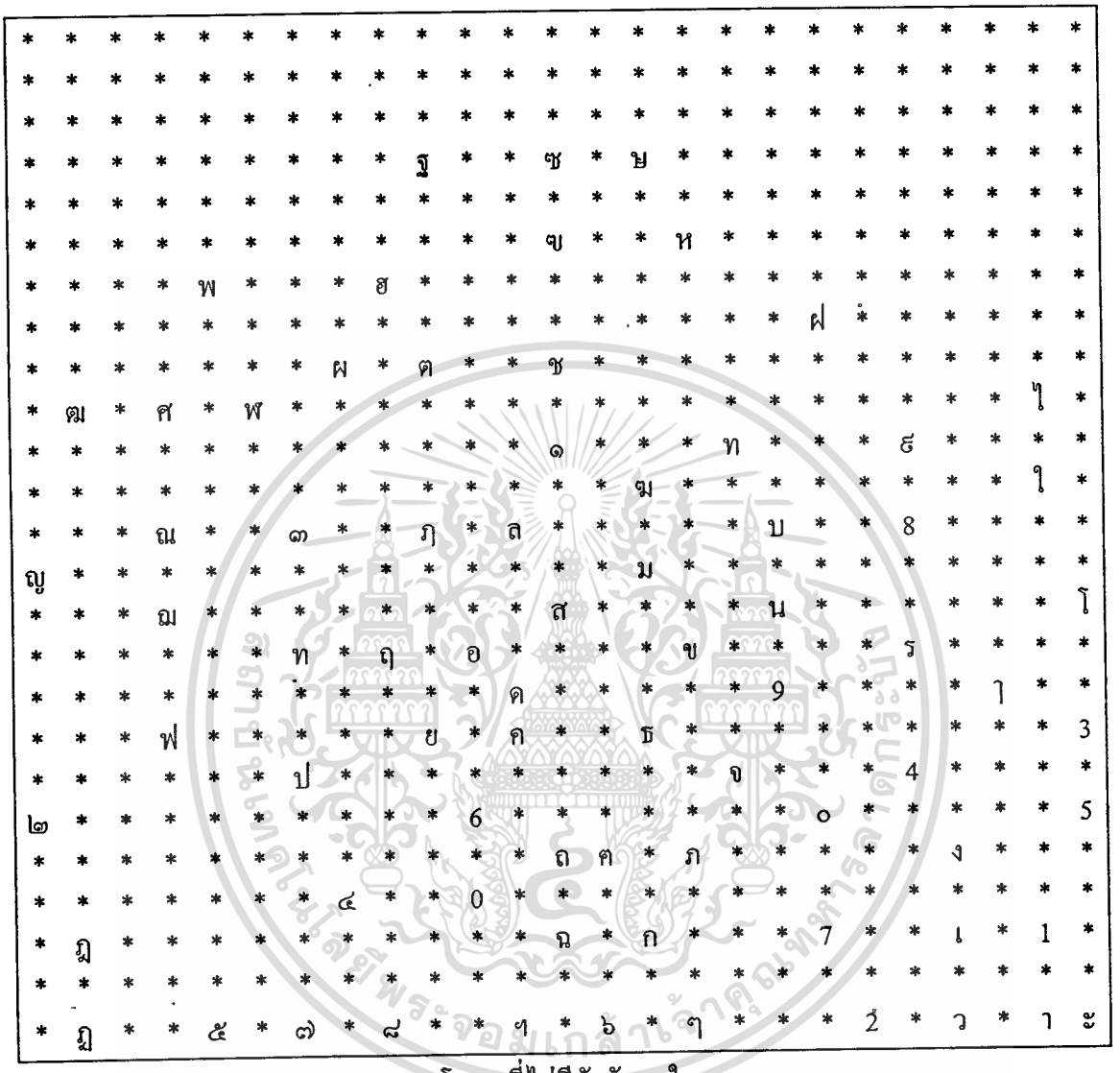
แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 20x20 โหนด ในการฝึกรอบที่ 60

โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.4 - 0.6

ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 12



* แทนโหนดที่ไม่มีตัวอักษรใดตกเลย

แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 25 x 25 โหนด ในการฝึกรอบที่ 60

โดยค่า $\alpha_s = 0.6$ $\alpha_e = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.4 - 0.6

ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

เมื่อเปรียบเทียบการแยกแยะตัวอักษรระหว่างตารางที่ 9-12 ที่ค่าน้ำหนักเริ่มต้นอยู่ในช่วง 0.4 ถึง 0.6 (ช่วงกลาง) กับตารางที่ 4-7 ที่ค่าน้ำหนักเริ่มต้นอยู่ระหว่าง 0 ถึง 1 จะพบว่า การกำหนดค่าน้ำหนักเริ่มต้นในช่วงกลาง ๆ (ตารางที่ 9-12) จะมีการแยกแยะตัวอักษรที่ดีกว่าการกำหนดน้ำหนักแบบสุ่ม โดยหากค่าน้ำหนักเริ่มต้นอยู่ในช่วงกลาง ๆ โครงข่ายขนาด 20 x 20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหนดก็จะแยกแยะข้อมูลได้ทั้งหมด (ตารางที่ 11) แต่ถ้าค่าน้ำหนักเริ่มต้นแบบสุ่มต้องใช้โครงข่ายที่มีขนาดใหญ่ขึ้นเป็น 25 x 25 โหนด (ตารางที่ 7)

เนื่องจากข้อมูลเข้าในงานวิจัยนี้ (ตัวอักษรภาษาไทย) มีมากกว่าสองมิติจึงไม่สามารถนำค่าน้ำหนักมาพลอตแสดงลักษณะการกระจายของค่าน้ำหนักระหว่างการฝึกในสองมิติได้ แสดงได้เพียงลักษณะการกระจายของตัวอักษรแบบต่าง ๆ ในโครงข่ายดังในตารางต่าง ๆ ที่กล่าวในบทนี้

3. การทดลองหาขนาดขอบเขตของโหนดข้างเคียง

จากการทดลองที่กล่าวมาข้างต้นจะกำหนดให้ขอบเขตโหนดข้างเคียงมี 5 ชั้น ($N_c = 5$) . ในการทดลองต่อไปนี้จะทดลองให้ $N_c = 7$ ดังแสดงการแมปในตารางที่ 13-15

ตารางที่ 13

ญฒ		ฟ		ฝ			ฤอ	ข	ธ๘
๒		ฎฏ			๔		ย		
พ		ฐ	ไ			6	ศดค	9	จธ
ฌณ	๗	ซ		ใ			ถ0		
ฬ		ฃ	ช		โ			๓ก	ฉ
ศ๕		ษ	ห			ๅร		ๆ	๖
ผฮ				บ		35		๐	ฯ
									1
ป		ฦส	ทม		4ง		เ		าะ

แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 10 x 10 โหนด ในการฝึกรอบที่ 80

โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 7$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0

ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ตารางที่ 14

ป		6		คด		กภ		ๆ	ฯ	o		1		ะ
						ฉ								า
๗	๘		0	ถ	ค	9			๖		เ			
												๋	ว	2
		๔		ธ		จ		โ			7			
ฦ														ง
		ไ	ใ		ส		อย		า					
ฝ					ล		ฤ		3		5		4	
		๚												
ฎฏ		ซ		ช		๑		ข					๙	8
											๓			
ฟ		๕	ศ		ฮ			ฑ					บ	น
๒											ฬ			
				ฐ		ณ					ษ		ม	
ญ	ฆ	พ				ฌ		ผ	ต		ห	ท		ฒ

แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 15 x 15 โหนด ในการฝึกรอบที่ 80

โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 7$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0

ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 15

ฆ่	*	ท	*	ฦ	*	ส	*	ใ	*	ไ	*	*	ฌ	ณ	*	๒	*	*	ญ
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	ฒ
*	ม	*	*	*	*	*	*	โ	*	*	*	*	ฮ	*	ฐ	*	พ	*	*
*	*	*	*	*	ร	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	า	*	*	๓	*	*	*	*	*	*	*	*	*
บ	*	น	*	*	*	*	*	*	*	*	*	*	ฑ	*	ฆ	*	ศ	*	๕
*	*	*	*	*	3	*	*	*	*	*	ฬ	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	5	*	ห	*	*	*	*	*	ฯ	*	ฏ	*	*
*	2	*	*	*	*	*	*	*	*	*	*	ผ	*	*	*	*	*	*	ฝ
*	*	*	*	7	*	ง	*	*	*	ย	*	*	*	*	ช	*	ฎ	*	*
*	*	ว	*	*	*	*	*	4	*	*	*	*	ต	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	ฟ	*
*	1	*	*	*	*	o	*	9	*	จ	*	ธ	*	*	ถ	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	๗	*	๘
ะ	*	*	*	*	*	*	*	*	*	*	*	ย	*	*	*	*	*	*	*
*	*	*	เ	*	*	*	*	ฉ	*	ข	*	*	*	*	๑	*	ป	*	๔
า	*	*	*	*	*	*	*	*	*	*	*	ฤ	อ	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	ค	*	*	*
ว	*	ฯ	*	ๆ	*	ก	ภ	*	ถ	ศ	*	0	*	6	*	ด	*	8	๙

*** แทนโหนดที่ไม่มีตัวอักษรใดตกเลย**

แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 20x20 โหนด ในการฝึกรอบที่ 80

โดยค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 7$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0

ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเปรียบเทียบการแยกแยะตัวอักษรกรณี N_c เท่ากับ 7 (ตารางที่ 13-15) กับ N_c เท่ากับ 5 (ตารางที่ 4-7) พบว่า N_c เท่ากับ 7 สามารถแยกแยะตัวอักษรได้ดีกว่า N_c เท่ากับ 5 โดยจะใช้ขนาดโครงข่ายเล็กกว่าคือ 20 x 20 โหนด ขณะที่ N_c เท่ากับ 5 ใช้โครงข่ายขนาด 25 x 25

4. การทดลองปรับเปลี่ยนค่าอัตราการเรียนรู้ (α)

ตารางที่ 16

ใ	*	*	ผ	*	พ	*	ฃ	*	ซ	*	ศ	*	*	ฐ	*	พ	ฒ	*	ญ
ไ	*	ต	*	*	*	*	*	*	*	*	๕	*	*	*	*	*	*	*	*
*	*	*	*	ษ	*	ช	*	*	*	*	*	*	*	*	*	*	*	*	*
๓	*	*	ห	*	*	*	*	ฑ	*	*	*	*	*	*	*	ฎ	*	*	๒
*	*	*	*	*	*	*	*	*	*	ฮ	*	*	ผ	*	*	ฎ	*	*	*
*	*	*	ท	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	ม	*	*	*	*	*	A	*	*	ล	*	*	*	*	*	*	*	*	ฌ
บ	*	*	*	ฒ	*	*	*	*	๑	*	*	ส	*	*	ฟ	*	*	*	*
น	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	ณ
ร	*	โ	*	ข	*	6	*	*	*	*	*	*	ย	*	*	*	*	*	*
ๅ	*	*	*	*	*	*	*	*	0	*	ค	*	B	*	*	*	*	*	*
*	*	7	*	*	*	*	*	ถ	*	*	*	*	*	*	*	๗	*	*	๘
*	*	*	*	*	*	ภ	ก	*	*	*	*	*	*	*	*	*	*	*	*
C	*	*	*	๐	*	*	*	*	*	*	9	*	*	*	*	*	*	*	*
*	*	*	*	*	*	ฯ	*	*	ฉ	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	ๆ	*	*	*	*	D	*	*	*	*	*	*	*	ป
*	1	*	*	ฌ	*	*	*	*	๖	*	*	*	*	*	*	*	๔	*	*
*	*	*	*	*	*	*	*	*	*	*	*	8	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
ว	*	2	*	ง	4	*	5	*	3	*	*	๘	*	*	*	*	*	ฦ	*

* แทนโหนดที่ไม่มีตัวอักษรใดตกเลย

A แทน ฤ,อ B แทน ค,ด C แทน า,ะ D แทน ธ,จ

แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 20x20 โหนด ในการฝึกรอบที่ 80

โดยค่า $\alpha_s = 0.9$ $\alpha_o = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0

ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองข้างต้นกำหนดค่าอัตราการเรียนรู้เริ่มต้นเท่ากับ 0.6 ในการทดลองนี้จะทดลองกำหนดให้มีค่าเท่ากับ 0.9

ตารางที่ 17

A	*	*	ว	*	ง	4	*	*	*	*	*	*	*	*	*	*	*	*
*	1	*	*	*	*	*	*	*	*	*	ๆ	*	*	*	*	*	*	*
เ	*	*	*	7	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	๖	*	ร	*	*	*	*	*	*	*	*
o	*	ๆ	*	ก	*	*	*	*	*	*	*	โ	*	*	*	*	*	*
ฯ	*	*	ภ	*	9	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	ฅ	ด	*	บ	*	น	*	*	*	*	*	*	*	*
ฉ	*	*	B	*	*	ค	*	*	*	บ	*	ฒ	*	*	*	*	*	*
จ	*	*	*	*	อ	ฤ	*	*	*	ม	*	*	*	*	*	*	*	*
ธ	*	*	6	ย	*	ส	*	ฦ	*	ท	*	*	ป	*	*	*	*	*
*	*	*	*	*	*	ล	*	*	*	*	*	*	*	*	*	*	*	*
8	๙	*	*	*	*	*	*	*	*	*	*	*	*	*	*	ฝ	*	*	ฟ
*	*	*	*	*	๔	*	*	๘	*	*	*	*	*	*	*	*	*	*
ใ	๑	*	๓	*	*	*	*	จ	*	*	*	ฎ	ฎ	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
ไ	*	*	ห	*	*	*	*	๕	*	*	*	*	*	*	*	*	*	*
ช	*	*	*	ษ	*	พ	*	ศ	พ	*	*	*	ญ	*	*	*	*	*
ฃ	*	ต	*	ผ	*	*	*	*	*	*	*	ฒ	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	๒	*	*	*
ฑ	ฆ	*	ฮ	*	*	ฌ	ณ	*	*	*	*	*	*	*	*	*	*	*

A แทน า, ะ B แทน 0, ถ * แทนโหนดที่ไม่มีตัวอักษรใดตกเลย

แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 20x20 โหนด ในการฝึกรอบที่ 54 โดยค่า $\alpha_s = 0.3$ $\alpha_c = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0

ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแมปตัวอักษรที่กำหนดให้โครงข่ายมีค่าอัตราการเรียนรู้เป็น 0.9 ดังแสดงในตารางที่ 16 เมื่อเปรียบเทียบกับตารางที่ 6 ที่มีอัตราการเรียนรู้เป็น 0.6 จะพบว่า ตารางที่ 16 กลุ่มตัวอักษรที่แยกไม่ได้มี 4 กลุ่มซึ่งมากกว่าในตารางที่ 6 ที่มีเพียง 1 กลุ่ม นั่นคือ ถ้าค่าอัตราการเรียนรู้สูง (เช่น 0.9) เกินไป การแยกแยะข้อมูลของโครงข่ายจะไม่ดีเท่ากับอัตราการเรียนรู้กลาง ๆ (เช่น 0.6)เมื่อทดลองกำหนดอัตราการเรียนรู้เป็น 0.3 ผลการทดลองได้ดังตารางที่ 17 จะเห็นว่ามีตัวอักษรอยู่ 2 กลุ่มที่แยกแยะไม่ได้ เมื่อเปรียบเทียบผลการทดลองระหว่างการกำหนดอัตราการเรียนรู้เป็น 0.3, 0.6 และ 0.9 ที่อัตราการเรียนรู้ 0.6 จะให้ผลการแยกแยะที่ดีที่สุด

5. การทดลองลดหลั่นค่าอัตราการเรียนรู้ตามระยะห่างจากโหนดชนะ

การทดลองข้างต้นให้ค่าอัตราการเรียนรู้ของโหนดชนะและโหนดข้างเคียงในการฝึกรอบใด ๆ มีค่าเท่ากัน (แต่ในการฝึกแต่ละรอบอัตราการเรียนรู้จะไม่เท่ากัน) ในการทดลองนี้จะกำหนดให้ค่าอัตราการเรียนรู้ของโหนดข้างเคียงมีค่าน้อยกว่าโดยมีค่าลดหลั่นตามระยะที่ห่างจากโหนดชนะโดยอัตราการเรียนรู้ลดลง 10 และ 20 เปอร์เซ็นต์ตามลำดับความห่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 18

*	*	*	๗	*	*	A	*	*	*	*	ฐ	*	*	ญ	*	*	*	*	*
*	*	๘	*	*	*	*	*	*	ฮ	*	*	*	*	ฒ	*	*	*	*	๒
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	ฑ	*	*	ผ	*	*	๕	*	ศ	*	พ	*	*	*	ฟ
ป	*	*	๓	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	ต	*	ษ	*	ฬ	*	*	*	*	*	*	ฎ
๙	*	*	*	*	*	*	*	*	*	*	*	*	*	ฃ	*	ซ	*	*	*
*	*	*	ส	*	ล	*	*	*	ห	*	*	ช	*	*	*	*	*	*	ฎ
๘	*	*	*	*	*	*	ฦ	*	*	*	*	*	*	*	*	*	*	*	*
*	*	B	*	ย	*	*	*	*	*	ฒ	*	๑	*	*	ไ	*	*	ฝ	*
ธ	*	*	*	*	*	*	ฃ	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	6	*	ค	*	*	*	ท	*	*	*	*	ใ	*	*	*	*	*
จ	*	*	*	*	ด	*	*	*	บ	*	ม	*	*	*	*	*	*	*	า
*	*	*	C	*	*	*	ค	*	*	*	*	*	*	โ	*	ร	*	*	*
9	*	*	*	*	*	*	*	*	น	*	*	*	*	*	*	*	*	*	*
*	*	*	*	ก	*	ภ	*	*	*	o	*	*	*	*	*	*	*	*	*
ฉ	*	ๆ	*	*	*	*	*	*	*	*	*	*	7	*	*	3	*	*	*
*	*	*	*	*	*	1	*	*	*	2	*	*	*	*	*	*	*	*	*
*	*	ฯ	*	*	*	*	*	เ	*	*	*	*	*	*	*	5	*	*	*
๖	*	*	*	ะ	*	า	*	*	*	ว	*	ง	*	4	*	*	*	*	*

* แทนโหนดที่ไม่มีตัวอักษรใดตกเลย A แทน ฌ,ฆ B แทน ฤ,อ C แทน ถ,0

แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 20x20 โหนด ในการฝึกรอบที่ 80 โดยค่า $\alpha_s = 0.6$ $\alpha_e = 0.001$ และ α จะลดลงทีละ 10 เปอร์เซ็นต์ตามลำดับความห่างจากโหนดชนะ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 19

A	า	*	B	*	ง	4	*	5	*	C	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	3	*	*	*	โ	*	ใ	*	*	*	*	*
1	*	เ	*	*	7	*	*	*	*	*	*	*	*	*	*	ไ	*	*	*
*	*	*	*	*	*	น	*	บ	*	ฆ	*	*	*	*	*	*	*	*	*
*	*	*	*	*	๐	*	*	*	*	*	*	*	*	*	*	*	*	๓	*
ๆ	๖	*	ฯ	*	*	*	ม	*	*	ท	*	ช	*	ฃ	*	*	*	*	*
*	*	*	ฉ	*	*	*	*	*	ฦ	*	*	*	*	*	*	*	ฑ	*	*
D	*	*	*	*	*	6	*	*	*	*	ห	*	ย	*	*	*	*	*	*
*	*	*	E	*	*	*	*	*	๑	*	*	*	*	*	ฬ	*	*	*	ฌ
F	*	*	*	*	*	ส	*	*	*	*	*	*	ซ	*	*	*	*	*	*
*	*	*	G	*	*	*	*	*	*	ต	ผ	*	*	*	ศ	*	*	*	ณ
จ	*	*	*	*	*	*	*	ล	*	*	*	*	ฮ	*	*	*	*	*	*
*	*	*	*	อ	*	ข	*	*	*	*	*	*	*	*	*	พ	*	*	๕
ธ	*	ย	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	ฝ่	*	*	*	ฐ	*	*	ฒ	*	*	*
*	*	๔	*	*	ป	*	*	*	*	*	*	*	*	*	*	*	*	ญ	*
8	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	ฎ	*	*	๒	*	*	*	*
*	*	*	*	*	*	*	*	*	ฟ	*	*	*	*	*	*	*	*	*	*
*	*	๔	*	๘	*	๗	*	*	*	*	*	ฏ	*	*	*	*	*	*	*

* แทนโหนดที่ไม่มีตัวอักษรใดตกเลย A แทน ะ,า B แทน 2,ว C แทน ๅ,ร D แทน ภ,ก E แทน ศ,ถ F แทน 9,0 G แทน ด,ค

แสดงการแมปตัวอักษรของโครงข่าย SOM ขนาด 20x20 โหนด ในการฝึกรอบที่ 80 โดยค่า

$\alpha_s = 0.6$ $\alpha_c = 0.001$ และ α จะลดลงทีละ 20 เปอร์เซ็นต์ตามลำดับ

ความห่างจากโหนดชนะ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0

ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

เมื่อเปรียบเทียบตารางที่ 18 และ 19 จะพบว่า ในตารางที่ 18 มีตัวอักษรที่แยกไม่ได้ 3 กลุ่ม และตารางที่ 19 มีตัวอักษรที่แยกแยะไม่ได้ 7 กลุ่ม จะพบว่า เมื่อเปอร์เซ็นต์การลดค่า α เพิ่มขึ้นจาก 10 เป็น 20 เปอร์เซ็นต์ จำนวนตัวอักษรที่แยกแยะได้ก็จะน้อยลงตามลำดับ

6. การกำหนดขอบเขตโหนดข้างเคียงแบบย้อนกลับ

ขอบเขตของโหนดข้างเคียงจะวัดจากโหนดชนะ ซึ่งวัดจากโหนดชนะไปทางซ้าย ทางขวา วัดขึ้นไปด้านบน และลงด้านล่าง ขอบเขตกว้างเพียงใดจะกำหนดที่ค่า N_c ในงานวิจัยนี้ให้ค่า N_c ส่วนวิธีการวัดขอบเขตโหนดข้างเคียงก็มี 2 แบบ คือ

6.1 หากระยะที่วัดรัศมีจากโหนดชนะไปเท่ากับ N_c แล้วเกินขอบเขตโครงข่ายให้ถือขอบเขตของโครงข่ายเป็นที่สิ้นสุดขอบเขตโหนดข้างเคียง ดังแสดงในภาพที่ 29.1

6.2 หากระยะที่วัดรัศมีจากโหนดชนะไปเท่ากับ N_c แล้วเกินขอบเขตโครงข่ายให้ทดค่าที่เกินไปยังด้านตรงข้าม (มีการ roll back) ดังแสดงในภาพที่ 29.2

ภาพที่ 29

29.1) 29.2)

● แทนโหนดชนะ ◍ แทนโหนดข้างเคียง

แสดงขอบเขตโหนดข้างเคียงที่ให้ถือขอบเขตโครงข่ายเป็นที่สิ้นสุดขอบเขตโหนดข้างเคียง และขอบเขตโหนดข้างเคียงที่มีการทดค่าที่เกินไปยังด้านตรงข้าม (มีการ roll back)

ในการทดลองทั้งหมดในข้างต้นจะกำหนดขอบเขตโหนดข้างเคียงแบบข้อ (6.1) ในการทดลองนี้จะทดลองกำหนดขอบเขตโหนดข้างเคียงแบบข้อ (6.2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ในการทดลองทั้งหมดในข้างต้นจะกำหนดขอบเขตโหนดข้างเคียงแบบข้อ (6.1) ในการทดลองนี้จะทดลองกำหนดขอบเขตโหนดข้างเคียงแบบข้อ (6.2)

ตารางที่ 20

ร	*	*	ไ	*	*	4	*	ง	*	*	2	*	*	*	*	1	*	*	ะ
*	*	*	*	*	*	*	*	*	*	*	*	*	*	ว	*	*	*	*	*
*	*	*	*	*	*	*	5	*	*	7	*	*	*	*	*	เ	*	*	า
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	โ	*	*	3	*	*	*	*	o	*	ๆ	*	*	*	*	ฯ
*	*	*	*	*	*	*	*	*	*	๖	*	*	*	*	*	ภ	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	ก	*	*	*	ฉ	*
ป	*	*	ใ	*	*	*	6	*	*	*	ถ	0	*	ศ	*	*	*	*	*
*	*	*	*	*	*	*	*	*	ส	*	*	*	*	*	*	*	*	*	*
*	*	*	*	ไ	*	*	*	*	*	*	*	*	*	A	*	*	จ	*	9
ฌ	*	*	*	*	*	*	๓	*	๑	*	ฦ	*	*	*	*	*	*	*	*
*	*	*	*	*	๗	*	*	*	*	*	*	*	*	อ	*	*	*	*	*
ณ	*	*	ซ	*	*	*	ต	*	*	*	*	ล	*	*	*	ม	*	*	ธ
*	*	*	*	*	*	*	*	*	*	B	*	*	*	ย	*	*	*	*	*
*	*	*	*	ฮ	*	*	ผ	*	*	*	*	*	*	*	*	*	*	*	ข
ฐ	*	*	*	*	*	*	*	*	*	ฃ	*	*	ท	*	*	น	*	*	*
*	*	๒	*	ศ	*	*	*	พ	*	*	*	*	*	ฒ	*	*	*	*	ษ
พ	*	*	*	*	*	*	*	*	*	*	ช	*	*	*	*	บ	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	๘	*	*	*	*	*
ญ	*	ฒ	*	*	ฟ	*	C	*	ฝ	*	๕	*	๗	*	*	๔	*	*	8

* แทนโหนดที่ไม่มีตัวอักษรใดตกเลย, A แทน ค,ด B แทน ห,ษ C แทน ฎ,ฏ

แสดงการแมปตัวอักษรของโครงข่ายแบบ SOM ขนาด 20x20 โหนด ในการฝึกรอบที่ 60 โดยกำหนดให้ขอบเขตโหนดข้างเคียงมีการทดค่าที่เกินไปด้านตรงข้าม ค่า $\alpha_s = 0.6$ $\alpha_c = 0.001$ $N_c = 5$ ค่าน้ำหนักเริ่มต้นสุ่มระหว่าง 0.0 - 1.0 ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขอบเขตโหนดข้างเคียง จะพบว่า ตารางที่ 6 มีกลุ่มตัวอักษรที่แยกแยะไม่ได้เพียง 1 กลุ่ม ในขณะที่ตารางที่ 20 มีกลุ่มตัวอักษรที่แยกแยะไม่ได้ 3 กลุ่ม แสดงให้เห็นว่า การที่ขอบเขตโหนดข้างเคียงมีการทดค่าไปด้านตรงข้ามจะแยกแยะตัวอักษรไม่ดีเท่ากับการไม่ทดค่าขอบเขตโหนดข้างเคียง

7. การเพิ่มจำนวนรอบในการฝึก

จากการทดลองข้างต้น ในบางกรณีที่เมื่อฝึกจนครบทุกรอบตามอัลกอริทึมที่ 1 แล้วก็ยังไม่สามารถแยกแยะตัวอักษรได้ จึงได้ทดลองฝึกโครงข่ายต่อโดยไม่มีการเปลี่ยนค่าพารามิเตอร์ คือ ใช้ค่าพารามิเตอร์รอบสุดท้ายของการฝึกนั้น แต่ไม่ว่าจะเพิ่มจำนวนรอบการฝึกมากขึ้นเท่าใดก็ไม่สามารถแยกแยะตัวอักษรได้ โดยกลุ่มตัวอักษรที่แยกแยะไม่ได้ (ตัวอักษรหลายแบบตกโหนดเดียวกัน) จะเปลี่ยนไป เช่น บางรอบจะแยกตัว ก กับ ถ ไม่ได้ ขณะที่แยกตัว ด กับ ต ได้ แต่บางรอบก็จะแยกตัว ก กับ ถ ได้ ขณะที่แยกตัว ด กับ ต ไม่ได้ และก็จะสลับไปสลับมาเช่นนี้ไปเรื่อย ๆ

8. การปรับน้ำหนักไปในทิศทางตรงข้ามในกรณีที่ตกผิดโหนด

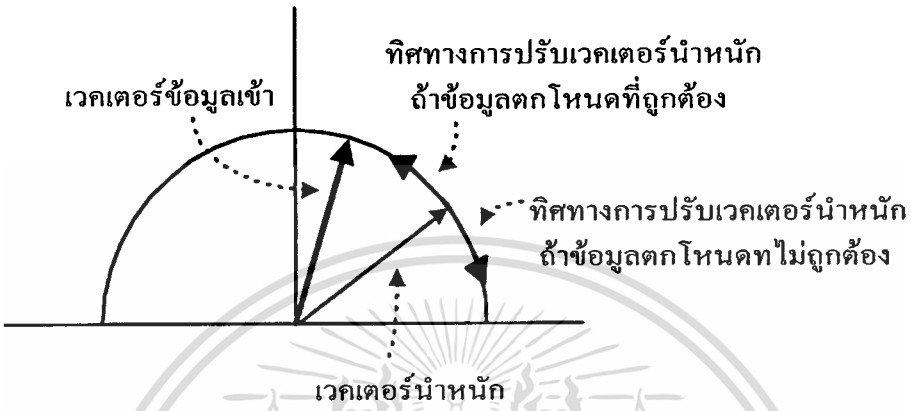
ในการฝึกบางรอบแม้โครงข่ายจะยังแยกแยะตัวอักษรไม่ได้ทั้งหมด ก็จะมีบางโหนดที่สามารถแยกแยะตัวอักษรได้แล้ว แต่เมื่อฝึกต่อไปจะพบว่า โหนดที่แยกแยะตัวอักษรได้แล้วนี้กลับไม่สามารถแยกตัวอักษรได้ คือ อาจไม่มีตัวอักษรใดมาตกอีกเลย หรืออาจมีตัวอักษรมาตกหลายตัว ที่เป็นเช่นนี้ก็เนื่องจากการปรับค่าน้ำหนักจะต้องปรับทั้งโหนดชนะและโหนดข้างเคียง ทำให้โหนดข้างเคียงบางโหนดมีการปรับน้ำหนักอยู่ตลอดเวลาถึงแม้จะเป็นโหนดที่เรียนรู้สมบูรณ์แล้วก็ตาม จึงทำให้โครงข่ายไม่เสถียร (nonstable) คือ ค่าน้ำหนักถูกปรับไปปรับมาตลอด (oscillate)

จากปัญหาที่กล่าวข้างต้นจึงทดลองปรับเปลี่ยนการฝึกใหม่ โดยในการฝึกที่ป้อนตัวอักษรเข้าฝึกทีละแบบ โหนดใดมีตัวอักษรแบบใดตกครั้งแรกให้ถือว่า โหนดนั้นเป็นโหนดแทนตัวอักษรที่ตกครั้งแรกนั้น ถ้ามีตัวอักษรแบบเดียวกันมาตกโหนดนี้อีกให้ถือว่า การตกของตัวอักษรถูกต้อง และปรับค่าน้ำหนักไปในทิศทางเดียวกับข้อมูลเข้า (สมการที่ 3.3) แต่ถ้าตัวอักษรที่ป้อนเข้าฝึกลำดับต่อมาไปตกยังโหนดที่มีตัวอักษรตกอยู่ก่อนแล้วเป็นแบบตัวอักษรต่างกัน จะถือว่าตัวอักษรตกไม่ถูกต้อง ให้ปรับค่าน้ำหนักไปในทิศทางตรงข้ามกับสมการที่ 3.3 ดังสมการที่ 5.1 [7] ซึ่งแสดงตัวอย่างการปรับน้ำหนักในลักษณะเวคเตอร์ 2 มิติของการปรับน้ำหนักทั้งสองแบบ ในภาพที่ 30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$W_{k+1} = W_i - \alpha\left(I - W_k\right) \quad \text{.....(5.1)}$$

ภาพที่ 30



แสดงทิศทางการปรับค่าน้ำหนักเมื่อตัวอักษรตกในโหนดที่ถูกต้องและไม่ถูกต้อง

แยกแยะตัวอักษรแย่ไปกว่าเดิม คือ ในช่วงแรกโหนดที่แยกแยะได้แล้วมีค่าน้ำหนักที่สมดุลแล้ว เมื่อฝึกต่อไปมีการตกผิดทำให้มีการปรับน้ำหนักไปในทิศตรงข้ามจึงทำให้น้ำหนักไม่สมดุลจากที่โหนดนี้แยกแยะตัวอักษรได้แล้วกลับแยกแยะไม่ได้

<u>สรุปปัญหาการฝึกโครงข่ายโดยใช้ SOM</u>

จากการทดลองในบทนี้

หากโครงข่ายมีขนาดเล็กเกินไปจะไม่สามารถแยกแยะตัวอักษรได้ เนื่องจากมีจำนวนโหนดที่เหมาะสมกับการแยกแยะตัวอักษรไม่เพียงพอ จึงต้องทดลองเพิ่มขนาดโครงข่ายให้มีขนาดใหญ่ขึ้นจึงจำทำให้สามารถแยกแยะตัวอักษรได้ โดยเมื่อมีจำนวนตัวอักษรที่ใช้ฝึกมากขึ้นก็ต้องเพิ่มให้โครงข่ายมีขนาดใหญ่ขึ้น ซึ่งทุกครั้งที่มีการปรับเปลี่ยนขนาดโครงข่ายจะต้องเริ่มต้นฝึกโครงข่ายใหม่ จึงทำให้ต้องเสียเวลาในการทดลองฝึกนานมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในบางการทดลองจะมีบางโหนดที่ไม่ว่าจะฝึกกี่รอบ ก็ไม่มีตัวอักษรใดมาตกยังโหนดนั้นเลย เนื่องจากโครงข่ายมีตำแหน่งของโหนดที่คงที่ตลอดอาจทำให้ค่าน้ำหนักของโหนดนั้น ๆ มีค่าที่กระจายอยู่ห่างจากขอบเขตการกระจายของค่าข้อมูลตัวอักษรมากจนค่าน้ำหนักไม่มีโอกาสที่จะถูกปรับให้มีค่าที่เหมาะสมได้

จากปัญหาดังกล่าวจึงนำโครงข่าย ASOM ที่มีความยืดหยุ่นในการปรับเปลี่ยนจำนวนโหนด และโครงสร้างของโครงข่ายมากกว่ามาประยุกต์ใช้ในการแก้ปัญหา ดังจะกล่าวถึงรายละเอียดในบทต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

การทดลอง ASOM

จากปัญหาการฝึกโครงข่ายที่กล่าวในบทที่ 6 จึงทดลองฝึกโครงข่ายโดยใช้แบบจำลอง ASOM ซึ่งมีขั้นตอนการฝึกตามอัลกอริทึมที่ 2 ดังแสดงในบทที่ 4 แต่การฝึก ASOM ตามอัลกอริทึมที่ 2 ก็ยังไม่สามารถแยกแยะตัวอักษรได้ทั้งหมด จึงมีการปรับเปลี่ยนอัลกอริทึมดังกล่าวเพื่อให้สามารถทำงานได้มีประสิทธิภาพดียิ่งขึ้น ซึ่งในบทนี้จะได้อธิบายถึงการทดลองเพื่อหาวิธีการฝึกที่จะทำให้โครงข่าย ASOM สามารถทำงานได้อย่างมีประสิทธิภาพ ปัญหาในการฝึก และการปรับปรุงแก้ไขขั้นตอนการฝึก

<u>ชุดตัวอักษรที่ใช้ฝึก</u>

เนื่องจากโครงสร้างภาษาไทยมี 3 ระดับ จึงแบ่งตัวอักษรเป็น 3 กลุ่ม และแบ่งฝึกเป็น 3 โครงข่าย เพื่อลดเวลาในการฝึกและการวิเคราะห์ตัวอักษร คือ

1. ตัวอักษรที่อยู่ในระดับกลาง ได้แก่ อักขระภาษาไทย สระที่อยู่ระดับเดียวกันกับอักขระ ตัวเลขไทย และตัวเลขอังกฤษ

ภาพที่ 31

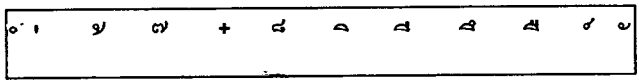
ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ฤ ฦ เ โ ไ ใ ฯ ะ า ๆ
๐ ๑ ๒ ๓ ๔ ๕ ๖ ๗ ๘ ๙
0 1 2 3 4 5 6 7 8 9

แสดงตัวอักษรไทยที่อยู่ในระดับกลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ตัวอักษรที่อยู่ระดับบน ได้แก่ สระที่อยู่ระดับบน และวรรณยุกต์

ภาพที่ 32



แสดงตัวอักษรไทยที่อยู่ในระดับบน

3. ตัวอักษรที่อยู่ระดับล่าง ได้แก่ สระที่อยู่ระดับล่าง

ภาพที่ 33

แสดงตัวอักษรไทยที่อยู่ในระดับล่าง

<u>การทดลอง</u>

การฝึกในช่วงแรกจะทดลองฝึกกับตัวอักษรระดับกลางก่อน โดยมีการทดลองดังนี้

1. ชุดตัวอย่างที่ใช้ฝึก ตัวอักษรที่ใช้ฝึกมี 75 แบบ จะแบ่งชุดการฝึกเป็น 3 ชุด คือ

ชุดแรก มีตัวอักษรแบบละ 1 ตัว รวม 75 ตัว

ชุดสอง มีตัวอักษรแบบละ 20 ตัวโดยแบบตัวอักษรเดียวกันจะอยู่ต่อเนื่องกันทั้ง 20 ตัว

ชุดสาม มีตัวอักษรแบบละ 20 ตัวโดยจะเรียงสลับแบบตัวอักษรกันตัวต่อตัวจนครบ 75 แบบแล้วเรียงสลับเช่นเดิมจนครบทุกตัวอักษรในชุดการฝึก (ชุด 2,3 มี 1500ตัว)

2. ทดลองหาวิธีการป้อนอักษรเข้าสู่โครงข่าย ทดลอง 3 วิธี คือ

วิธีแรก ฝึกด้วยตัวอักษรในชุดสองอย่างเดียว

วิธีที่สอง ฝึกด้วยตัวอักษรชุดแรกจนสามารถแยกแบบอักษรได้หมดก่อน แล้วจึงฝึกด้วยตัวอักษรชุดสองต่อจนกว่าจะแยกกันได้หมดทุกตัวจึงสิ้นสุดการฝึก

วิธีที่สาม ฝึกด้วยตัวอักษรชุดแรกจนสามารถแยกแบบอักษรได้หมดก่อน แล้วจึงฝึกด้วยตัวอักษรชุดสาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การทดลองหาค่าอัตราการเรียนรู้ที่เหมาะสม ทดลองกำหนดค่าอัตราการเรียนรู้ ของโหนดชนะ และโหนดข้างเคียง (α_{win} , α_{neigh}) ตามลำดับ เช่น (0.6, 0.2), (0.1, 0.04), (0.03, 0.001), (0.06, 0.002), (0.06, 0.00085), (0.06, 0.0008), (0.06, 0.0007)

4. การกำหนดโครงสร้างเริ่มต้นของโครงข่าย ให้โครงข่ายมีโครงสร้าง 2 มิติ ทดลองกำหนดจำนวนโหนดเริ่มต้นก่อนทำการฝึกเท่ากับ 3 และ 10 โหนด โดยให้มีการเชื่อมต่อกันแบบสุ่ม

5. **การทดสอบประสิทธิภาพการจดจำ** ทดสอบประสิทธิภาพการจดจำด้วยตัวอักษร 4 ชุด ซึ่งเป็นตัวอักษรที่มีความคมชัด (สะแกนจากกระดาษที่พิมพ์ด้วยเลเซอร์) และตัวอักษรที่มีสัญญาณรบกวน (สะแกนจากกระดาษที่ถ่ายเอกสาร) รวม 15,000 ตัว คือ

ตัวอักษรที่พิมพ์จากเครื่องเลเซอร์ที่ใช้ความสว่างในการสะแกน100 เปอร์เซ็นต์

ตัวอักษรที่พิมพ์จากเครื่องเลเซอร์ที่ใช้ความสว่างในการสะแกน 50 เปอร์เซ็นต์

ตัวอักษรที่ถ่ายเอกสารจากกระดาษที่พิมพ์ด้วยเลเซอร์ที่ใช้ความสว่างในการสะแกน 100 เปอร์เซ็นต์

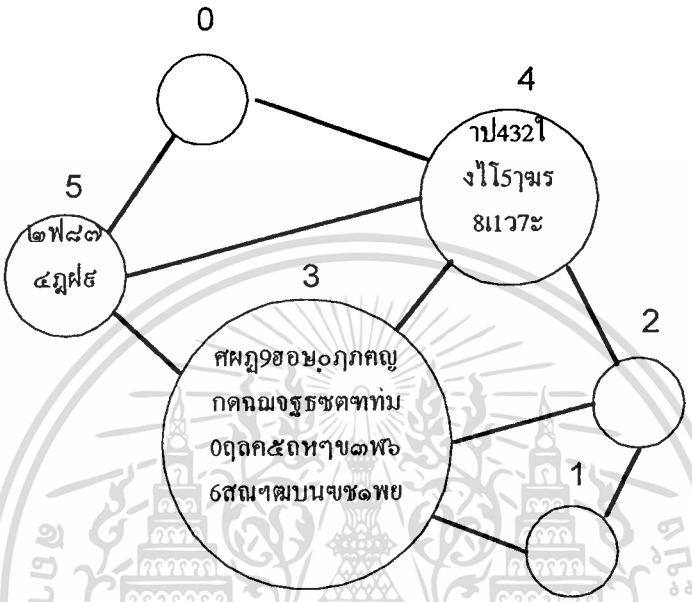
ตัวอักษรที่ถ่ายเอกสารจากกระดาษที่พิมพ์ด้วยเลเซอร์ที่ใช้ความสว่างในการสะแกน 50 เปอร์เซ็นต์

<u>ผลการทดลอง ASOM</u>

กำหนดให้โครงข่ายเริ่มต้นมีจำนวนโหนด 3 โหนด และเมื่อฝึกโครงข่ายครบทุกตัวอักษรในชุดฝึกจะถือว่าฝึกครบ 1 รอบ จากการทดลองฝึกตามอัลกอริทึมที่ 2 โดยใช้ตัวอักษรในชุดฝึกแรก พบว่า เมื่อเริ่มต้นฝึกตัวอักษรค่อนข้างจะตกยังโหนดเดียวกันจำนวนมาก แล้วจึงค่อย ๆ กระจายออกไปตกยังโหนดอื่นในรอบถัด ๆ ไป ดังแสดงในภาพที่ 34 1 เมื่อฝึกต่อไปตัวอักษรก็จะกระจายออกเป็นกลุ่ม ๆ ตัวอักษรที่คล้ายกันก็จะอยู่กลุ่มเดียวกันหรือตกอยู่โหนดเดียวกันในรอบถัด ๆ มา เช่น ตัว คดฅ ในรูปที่ 34.3 และตัวอักษรก็จะกระจายดีขึ้นเรื่อย ๆ จนสามารถแยกแยะได้เมื่อฝึกโครงข่ายต่อไป แต่การฝึกตามอัลกอริทึมที่ 2 ไม่สามารถแยกแยะตัวอักษรได้ จึงได้มีการปรับเปลี่ยนอัลกอริทึมใหม่ ดังจะกล่าวถึงปัญหาและวิธีปรับเปลี่ยนขั้นตอนการฝึกในหัวข้อต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

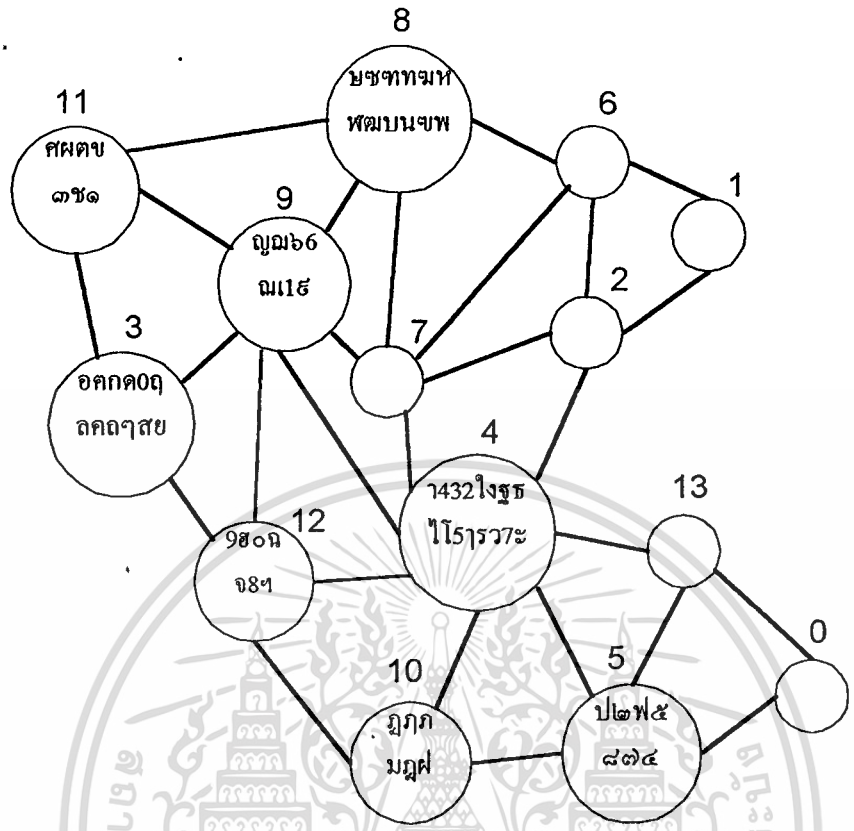
ภาพที่ 34



34.1) แสดงการแมปตัวอักษรของโครงข่ายในรอบที่ 2

ต่อ)

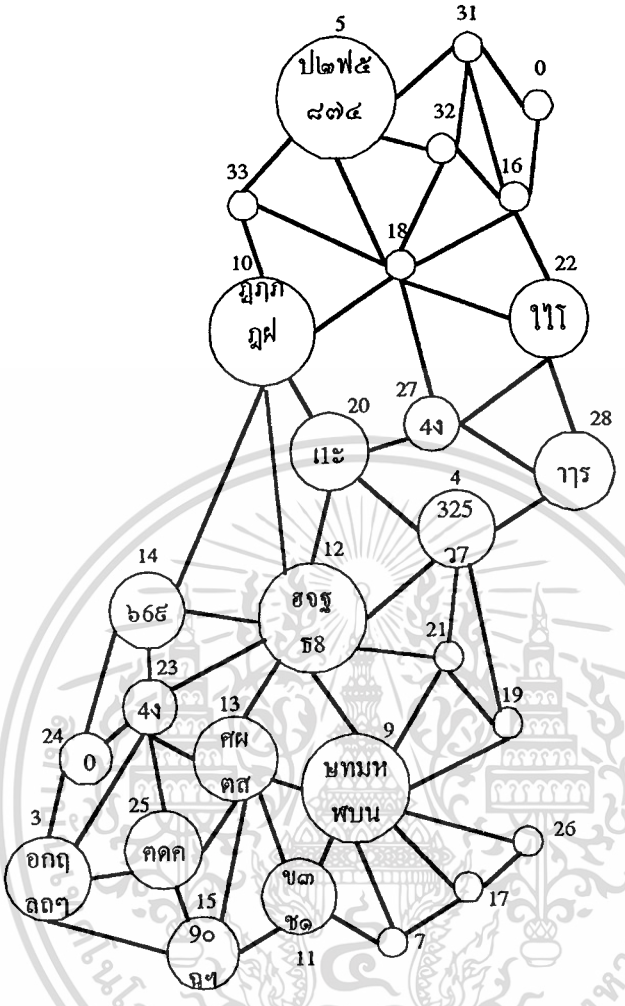
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



34.2) แสดงการแมปตัวอักษรของโครงข่ายในรอบที่ 10

(ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



34.3) แสดงการแมปตัวอักษรของโครงข่ายในรอบที่ 30

ตัวเลขนอกวงกลมแสดงลำดับของโหนดที่เพิ่มเข้าไป

แสดงการแมปตัวอักษรที่สามารถแยกแยะได้ดีขึ้นตามจำนวนรอบที่ใช้ฝึก

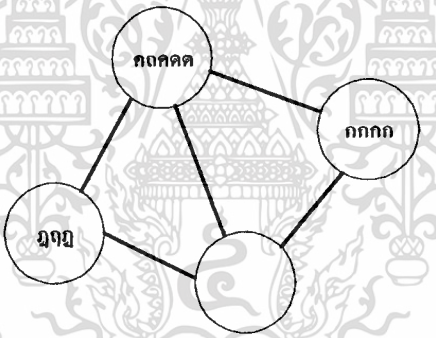
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัญหา การแก้ปัญหา และปรับเปลี่ยนขั้นตอนการฝึก ASOM

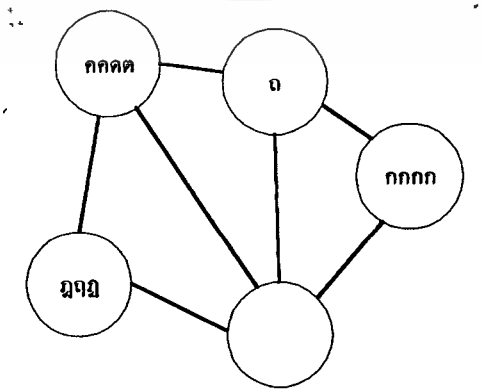
1. ปัญหาการเพิ่มโหนดใหม่ที่ไม่ทำให้การแยกยะดีขึ้น

ในการเพิ่มโหนด โหนดใหม่ที่เพิ่มเข้าไปในแต่ละรอบบางครั้งก็ทำให้การแยกแยะตัวอักษรดีขึ้น แต่บางครั้งก็ไม่ได้ทำให้การแยกแยะตัวอักษรดีขึ้นคือ ไม่มีตัวอักษรมาตกยังโหนดใหม่เลย ทั้งนี้อาจเพราะค่าน้ำหนักของโหนดนั้นไม่เหมาะสม ดังแสดงตัวอย่างในภาพที่ 35 เมื่อเพิ่มโหนดครั้งแรกตัว ถ จะแยกออกจากกลุ่มตัว ค,ค แต่เมื่อเพิ่มโหนดสองเข้าไป ตัว ก ที่แยกแยะได้แล้วก็แยกออกมาจากกลุ่มตัว ก อีก โดยตัวอักษรในกลุ่มที่ยังแยกแยะไม่ได้ก็ยังแยกแยะไม่ได้เหมือนเดิมซึ่ง

ภาพที่ 35



35.1) แสดงการแยกแยะตัวอักษรก่อนการเพิ่มโหนด

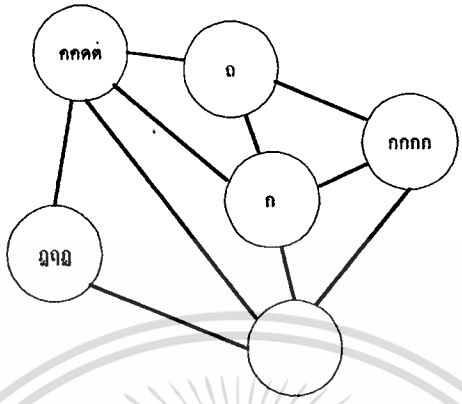


35.2) แสดงการเพิ่มโหนดที่ทำให้การแยกแยะตัวอักษรดีขึ้น

(ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ต่อ)



35.3) แสดงการเพิ่มโหนดที่ไม่ทำให้การแยกแยะตัวอักษรดีขึ้น

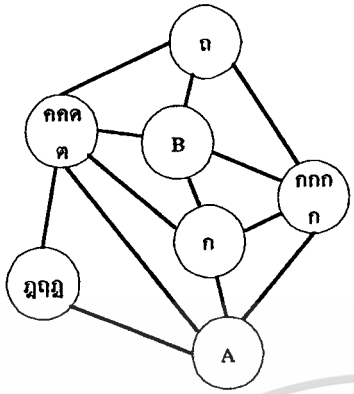
แสดงการเพิ่มโหนดให้โครงข่าย

2. ปัญหาโหนดในโครงข่ายถูกลบออกทั้งหมด

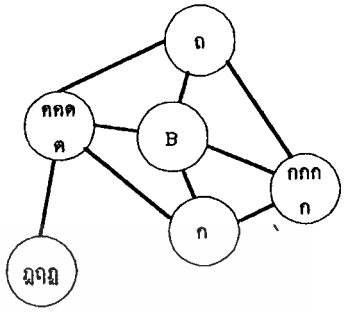
ในการลบโหนดจะลบโหนดที่มีค่า normalised probability density น้อยกว่าค่าเทรสโฮลด์ออก และเมื่อลบแล้วก็จะต้องปรับโครงสร้างโครงข่ายให้อยู่ในรูปซิมเพลกด้วย คือ ถ้าโหนดที่เหลือหลังการลบโหนดใดไม่อยู่ในซิมเพลกใดเลยก็จะต้องถูกลบออกด้วย ซึ่งบางครั้งโหนดที่ต้องถูกลบออกไปด้วยนี้ก็มีตัวอักษรตกจึงไม่ควรถูกลบออก และเนื่องจากเริ่มต้นฝึกด้วยโครงข่ายที่มีขนาดเล็ก (จำนวนโหนดน้อย) การลบโหนดโดยยึดหลักการนี้ทำให้บางครั้งโหนดในโครงข่ายทั้งหมดถูกลบออก ดังแสดงตัวอย่างลำดับการลบโหนดออกจากโครงข่ายดังในภาพที่ 36 ที่ในที่สุดโหนดในภาพที่ 36.3 ก็จะถูกลบออกทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

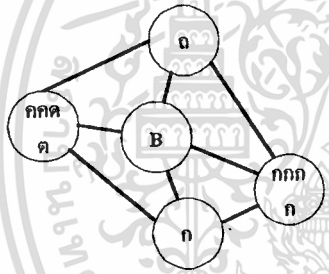
ภาพที่ 36



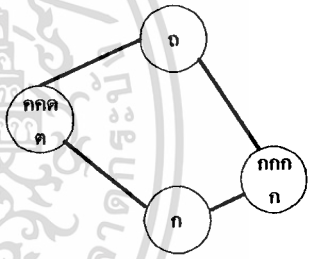
36.1) ก่อนลบโหนด



36.2) ระหว่างลบโหนด A



36.3) หลังลบโหนด A



36.4) ระหว่างลบโหนด B

แสดงขั้นตอนการลบโหนดที่ไม่มีอักษรใดตกออกแล้วมีผลทำให้โหนดในโครงข่ายทั้งหมดถูกลบออกด้วย

ปัญหานี้แก้โดยการให้มีการเพิ่มโหนดใหม่ในทุกรอบการฝึก แต่จะยืดระยะเวลาในการลบโหนดที่ไม่มีตัวอักษรใดตกออกไปจากการที่มีการลบเมื่อฝึกครบทุกรอบการฝึก ก็ให้ฝึกหลาย ๆ รอบแล้วค่อยมีการลบ 1 ครั้ง โดยทดลองให้มีการลบเมื่อฝึกครบ 3 รอบ ผลการทดลองพบว่าปัญหาที่โหนดในโครงข่ายทั้งหมดถูกลบแก้ได้ เมื่อจำนวนโหนดในโครงข่ายเพิ่มขึ้นถึงประมาณ 30 โหนด การเพิ่มและลดโหนดก็จะวนไปวนมาคือ เมื่อเพิ่มโหนดเข้าไปแล้วทำการฝึกต่อไปโหนดนั้นก็ไม่มีอักษรใดมาตก จึงทำให้โหนดถูกลบออก (เพราะค่า normalize probability density เป็น 0 ซึ่งน้อยกว่าค่าเทรสโฮลด์) เมื่อลบออกก็จะมีการเพิ่มโหนดใหม่อีกวนไปวนมาโดยที่

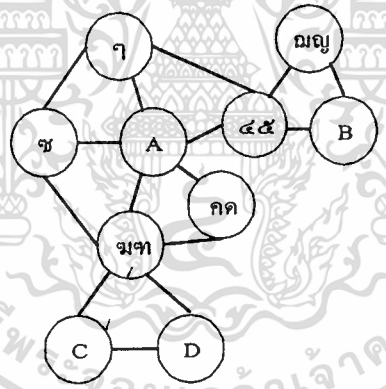
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวนโหนดจะไม่เพิ่มหรือลดลงไปมากกว่านี้อีก จึงปรับการทดลองให้มีการลบโหนดเมื่อฝึกครบทุก 10 รอบ จำนวนโหนดก็จะเพิ่มขึ้นจนถึงประมาณ 50 โหนดก็จะไม่มีการเพิ่มหรือลดไปมากกว่านี้เช่นเดิม

3. ปัญหาการลดค่าสัญญาณการตกระหว่างฝึก (จากปัญหาในข้อ 2)

หลังจากที่ลบโหนดที่มีค่า normalized probability density น้อยกว่าค่าเทรสโฮลด์ออก บางกรณีจะทำให้โหนดที่มีตัวอักษรตกถูกลบออกไปด้วยดังแสดงในภาพที่ 36 จึงปรับวิธีการลบโหนดใหม่ โดยจะตรวจสอบก่อนว่าเมื่อลบโหนดหนึ่งโหนดใดแล้ว โหนดที่เหลือจะมีโหนดที่มีตัวอักษรตกซึ่งไม่ได้อยู่ในโครงสร้างซิมเพลกถูกลบออกตามหรือไม่ ถ้าไม่มีจึงจะทำการลบ แต่ถ้ามีก็จะไม่ลบโหนดนั้น ๆ ดังแสดงตัวอย่างในภาพที่ 37 แม้จะทำให้โหนดที่มีตัวอักษรตกไม่ถูกลบออก แต่การแก้ปัญหาด้วยวิธีนี้ก็ยังเกิดการเพิ่มและลบโหนดที่วนไปวนมาเช่นเดิม

ภาพที่ 37



โหนด A,B ไม่ควรลบออก โหนด C,D ลบออกได้

แสดงการพิจารณาโหนดที่ควรลบและไม่ควรลบออกจากโครงข่าย

4. ปัญหาการลดค่าสัญญาณการตกระหว่างการฝึก

การลดสัญญาณการตกของโหนดในโครงข่ายทั้งหมดลงอีก β เท่าของสัญญาณเดิม (β = 0.05) ในทุกครั้งที่มีการป้อนตัวอักษรเข้าฝึก ทำให้โหนดที่มีตัวอักษรตกในช่วงแรกของการฝึกแต่ในช่วงหลังไม่มีตัวอักษรตกเลย มีค่าสัญญาณการตกลดลงเรื่อย ๆ ทำให้ค่า normalised probability density ($\hat{p}_c$) ซึ่งแปรผันตรงกับจำนวนแบบอักษรตกที่ในโหนดนั้นลดลง และมีค่าน้อยกว่าค่าเทรสโฮลด์ (0.09) โหนดจึงถูกลบออกแม้จะเป็นโหนดที่สามารถแยกแยะแบบตัวอักษรได้แล้วก็ตาม

เอกสารนี้เป็นลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย ห้ามมิให้ทำซ้ำ ดัดแปลง เผยแพร่ หรือนำไปใช้ในการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จึงปรับไม่ให้มีการลดสัญญาณของโหนดในโครงข่ายในระหว่างการฝึกเพื่อที่ลำดับการป้อนตัวอักษรเข้าฝึกจะได้ไม่มีผลต่อค่าสัญญาณการตก เมื่อปรับเช่นนี้แล้วโหนดที่มีแบบตัวอักษรตกหลายแบบจะมีค่าสัญญาณการตกสูงขึ้นเรื่อยๆ เพื่อป้องกันไม่ให้ค่าสัญญาณการตกเกิดโอเวอร์โฟว์ (overflow) เมื่อฝึกครบ 1 รอบจะกำหนดค่าสัญญาณการตกของทุกโหนดในโครงข่ายให้เป็นศูนย์

5. ปัญหาการเพิ่มโหนดในตำแหน่งที่ไม่เหมาะสม

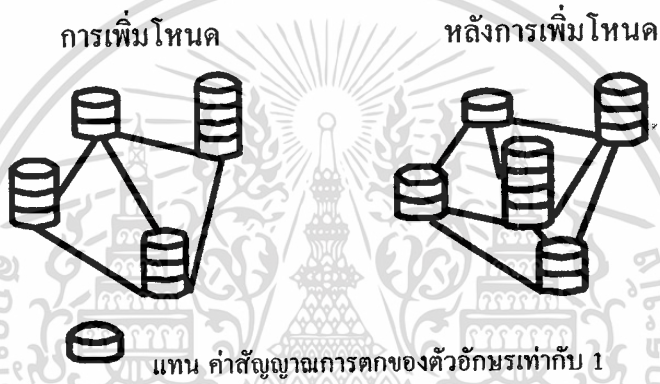
การเพิ่มโหนดในตำแหน่งที่มีค่า normalized probability density มากที่สุด ในบางกรณีก็ไม่เหมาะสม เช่น กรณีโหนดที่มีค่า normalized probability density มากที่สุดมีตัวอักษรเพียงแบบเดียวตก การเพิ่มโหนดที่ตำแหน่งนี้มักจะทำให้ตัวอักษรของโหนดนี้ซึ่งแยกแยะได้แล้วแยกออกมาตกยังโหนดใหม่แทนที่จะเป็นตัวอักษรจากโหนดที่ยังแยกแยะไม่ได้ ทำให้การเพิ่มโหนดใหม่ไม่ได้ช่วยให้โครงข่ายแยกแยะตัวอักษรได้ดีขึ้นเลย ดังแสดงในภาพที่ 35.3) จึงแก้ปัญหาด้วยการนับจำนวนแบบตัวอักษรที่ตกในแต่ละโหนด แล้วเพิ่มโหนดใหม่ในระหว่างโหนดที่มีจำนวนแบบของตัวอักษรมาตกมากที่สุด และโหนดข้างเคียงที่มีค่าน้ำหนักห่างจากโหนดนี้มากที่สุด จากการทดลองพบว่า มีการแยกแยะที่ดีขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ปัญหาการกำหนดสัญญาณการตกเริ่มต้นให้โหนดใหม่

โหนดใหม่ที่เพิ่มบางครั้งก็ไม่มีตัวอักษรใดตกเลย การกำหนดสัญญาณการตกเริ่มต้นให้โหนดใหม่ด้วยการคำนวณในสมการที่ 4.8 ซึ่งเป็นค่าเฉลี่ยของจำนวนโหนดที่อยู่รอบ ๆ โหนดใหม่ ซึ่งเป็นค่าที่ได้จากการคาดคะเนที่ไม่ใกล้เคียงความเป็นจริงเลย ดังแสดงในภาพที่ 38 ดังนั้นหากค่านี้สูงจะมีผลให้มีการเพิ่มโหนดใหม่ในตำแหน่งนี้อีก ซึ่งไม่เหมาะสมเพราะไม่ได้ช่วยให้การแยกแยะดีขึ้นเลย จึงกำหนดให้โหนดใหม่มีสัญญาณการตกเริ่มต้นเป็นศูนย์

ภาพที่ 38



แสดงปัญหาการกำหนดสัญญาณให้โหนดใหม่ที่ได้จากการคาดคะเน

7. ปัญหาการเลือกลบโหนดจากการพิจารณาค่า normalized probability density

กรณีที่มีตัวอักษรที่ใช้ฝึกจำนวนมาก ทำให้มีโหนดที่มีจำนวนตัวอักษรตกน้อยมีค่า normalized probability density น้อยกว่า 0.09 จึงถูกลบออก ซึ่งโหนดนี้ไม่ควรที่จะถูกลบเพราะมีตัวอักษรตก จึงแก้ปัญหาโดยการปรับให้ลบโหนดที่ไม่มีตัวอักษรใดตกเลยแทนการเลือกลบโหนดจากการพิจารณาค่า normalized probability density

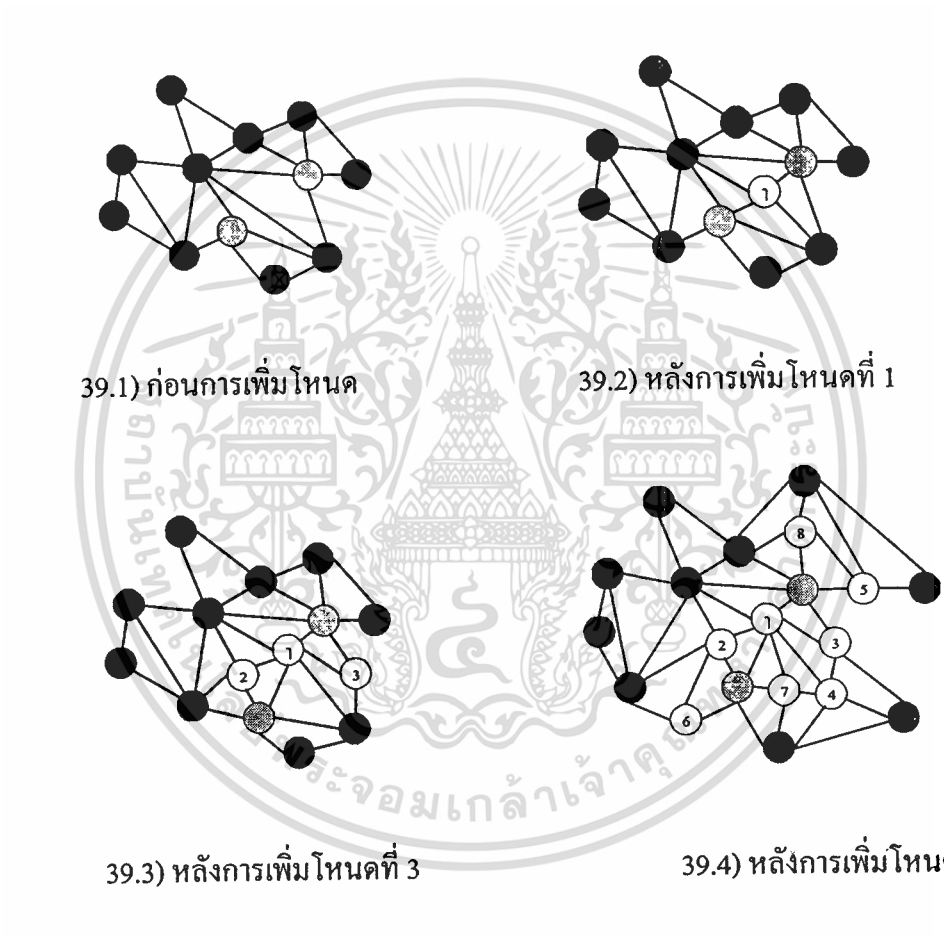
8. ปัญหาการลบและเพิ่มโหนดที่วนไปวนมา (จากปัญหาในข้อ 2 และ 3)

การที่โหนดถูกลบและเพิ่มวนไปวนมา แก้ปัญหาโดยให้มีการเพิ่มโหนดเพียงอย่างเดียวจนกว่าจะแยกแยะตัวอักษรได้หมดก่อน จึงจะทำการลบโหนดโดยให้พิจารณาลบเฉพาะโหนดที่ไม่มีตัวอักษรใดตกเท่านั้น และหลังการลบก็ไม่จำเป็นที่โครงข่ายจะต้องรักษาโครงสร้างซิมเพลคไว้ คือ หลังการลบหากโหนดใดไม่ได้อยู่ในโครงสร้างซิมเพลค และโหนดนั้นมีตัวอักษรตกอยู่ ก็ไม่ต้องลบออก วิธีนี้จะทำให้โหนดเก่าที่อาจเป็นโหนดที่แยกแยะตัวอักษรได้แล้ว และโหนดที่ไม่มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไปเรื่อย ๆ ดังแสดงในภาพที่ 39 ซึ่งทำให้ค่าน้ำหนักเริ่มต้นที่กำหนดให้แก่โหนดใหม่ตามสมการที่ 4.8 เบี่ยงเบนเข้าสู่ขอบเขตการกระจายของตัวอักษรมากขึ้นเรื่อย ๆ ตามลำดับการเพิ่มโหนดใหม่จากภาพที่ 39.1) ถึง 39.4) และจากที่มีการปรับค่าน้ำหนักในระหว่างการฝึกจึงทำให้ในที่สุดก็จะได้ค่าน้ำหนักที่เหมาะสม จนสามารถแยกแยะตัวอักษรได้

ภาพที่ 39



39.1) ก่อนการเพิ่มโหนด

39.2) หลังการเพิ่มโหนดที่ 1

39.3) หลังการเพิ่มโหนดที่ 3

39.4) หลังการเพิ่มโหนดที่ 8

○ แทนโหนดใหม่ , ตัวเลขภายในวงกลมแสดงลำดับการเพิ่มโหนด

◍ แทนโหนดที่ยังแยกแยะไม่ได้

● แทนโหนดที่แยกแยะได้แล้ว หรือโหนดที่ไม่เหมาะสมที่ไม่มีตัวอักษรตก

แสดงการเพิ่มโหนดใหม่ที่เพิ่มในบริเวณที่ยังแยกแยะตัวอักษรไม่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากขั้นตอนการฝึกที่กล่าวมา แล้วทดลองฝึกโครงข่ายจะแยกแยะตัวอักษรได้ดีขึ้น เพื่อให้โครงข่ายมีประสิทธิภาพการทำงานดีขึ้นจึงได้ทดลองปรับเปลี่ยนการฝึก ดังจะกล่าวถึงต่อไปนี้

9. ทดลองเพิ่มโหนดมากกว่า 1 โหนดเมื่อฝึกครบ 1 รอบ

เนื่องจากการเพิ่มโหนดทีละ 1 โหนดเมื่อฝึกครบ 1 รอบ ตัวอักษรจะค่อย ๆ ถูกแยกแยะไปตามลำดับ เพื่อปรับปรุงให้สามารถมีการแยกแยะตัวอักษรเพิ่มขึ้นมากกว่าเดิมจึงทดลองปรับเปลี่ยนให้มีการเพิ่มโหนดมากกว่า 1 โหนดเมื่อฝึกครบ 1 รอบ โดยทดลองเพิ่มมากกว่าหนึ่ง โหนด โดยเพิ่มในตำแหน่งโหนดที่มีสัญญาณการตกมากที่สุด และโหนดที่มีสัญญาณการตกมากเป็นอันดับสอง แต่หากโหนดทั้งสองอยู่ในซิมเพลคเดียวกัน อาจทำให้มีการเพิ่มโหนดที่อยู่ในตำแหน่งที่ใกล้กันเกินไป แสดงในภาพที่ 40 ค่าน้ำหนักของโหนดใหม่อันดับถัด ๆ มา ที่เพิ่มเข้าไปทีหลัง จะเป็นค่าเฉลี่ยจากโหนดแรกที่เพิ่มเข้าไปก่อน ซึ่งค่าน้ำหนักของโหนดแรก ๆ ที่เพิ่มเข้าไปก่อนยังไม่เคยถูกปรับเลย ทำให้ค่าน้ำหนักของโหนดหลัง ๆ มักไม่ค่อยเหมาะสม จึงไม่มีตัวอักษรใดมาตก การปรับน้ำหนักโดยการเพิ่มโหนดมากกว่า 1 โหนดเมื่อฝึกครบแต่ละรอบจึงไม่เหมาะสมนัก

ภาพที่ 40

40.1) ก่อนเพิ่มโหนด 40.2) หลังเพิ่มโหนดใหม่ 2 โหนด

● แทน โหนดเก่า ○ แทน โหนดใหม่

แสดงการเพิ่มโหนดทีละ 2 โหนดเมื่อฝึกครบ 1 รอบ

10. ทดลองปรับเปลี่ยนสมการปรับน้ำหนัก

จากการปรับขั้นตอนการฝึก แม้จะทำให้การแยกแยะดีขึ้น แต่ก็ยังมีปัญหาดังที่กล่าวในข้อที่ 1 ที่โหนดใหม่ที่เพิ่มเข้าไปบางครั้งก็ไม่สามารถทำให้ตัวอักษรแยกแยะออกจากโหนดที่มีตัวอักษรหลาย ๆ แบบตกได้ โดยเฉพาะกับตัวอักษรที่มีลักษณะคล้ายคลึงกันมาก จึงพยายามกำหนดค่าน้ำหนักเริ่มต้นของโหนดใหม่ให้อยู่ในขอบเขตตัวอักษรที่ยังแยกแยะไม่ได้มากที่สุด แต่จากการ

เอกสารนี้ ... เพื่อการศึกษาเท่านั้น ไม่ควรนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทดสอบ โดยทดลองกำหนดค่าน้ำหนักเริ่มต้นของโหนดใหม่ให้มีค่าเป็น 80 เปอร์เซ็นต์ของค่าน้ำหนักของโหนดที่ยังแยกแยะไม่ได้ ดังแสดงขั้นตอนการฝึกในอัลกอริทึมที่ 3 ก็ยังเกิดปัญหาที่โหนดใหม่ไปแยกตัวอักษรจากกลุ่มที่แยกแยะได้แล้วออกมาอีก จึงอาจสรุปได้ว่า การกำหนดค่าน้ำหนักเริ่มต้นให้โหนดใหม่ด้วยค่าเฉลี่ยระหว่างโหนดที่มีตัวอักษรหลายแบบตกมากที่สุดกับโหนดที่มีค่าน้ำหนักห่างจากโหนดนี้มากที่สุด เหมือนเดิมเป็นวิธีที่เหมาะสมแล้ว เนื่องจากระหว่างการฝึกตัวอักษรแต่ละแบบมีความหลากหลายมากจนไม่อาจคาดเดาได้ก่อน

อัลกอริทึมที่ 3

Step 1. initial weight, $\alpha_{winner} = 0.6$, $\alpha_{neighbor} = 0.2$ and number of node

Step 2. select 1 character / pattern in training set to be inputs (training set 1)

Step 3. while classify False , do step 4-14

Step 4. for each training input I do step 5-8

Step 5. apply input to input node

Step 6. find winner node (eq. 3.2)

Step 7. update winner and neighbor node

Step 8. if Thai pattern has never allocated winner,

$$signal = signal + 1$$

Step 9. find max-signal node (eq. 5)

Step10. find node that weight is max distance from weight of max-signal node (eq. 4.3)

Step11. insert new node between max-signal node and max-distance node

Step12. initial neighborhood node for new node, and weight of new node

if train by training set 1 , calculate weight according to (eq. 3.3)

if train by training set 2 or 3, weight is

$$W_{new\_node} = 0.8\ W_{\max\_signal\_node}$$

Step13. update signal (eq. 4.7) and neighborhood of new node's neighbor

Step14. if each node is allocated by same-character inputs, classify TRUE

Step 15. if train by training set 1 set classify FALSE, reduce $\alpha_{neighbor} = 0.002$

Step 16. repeat step 2 -14 train by all pattern (training set 2 or 3)

Step 17. delete all nodes that non-character allocate

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากปัญหา การแก้ปัญหา และการพยายามปรับเปลี่ยนอัลกอริทึม จึงได้นำมาปรับปรุงเป็นอัลกอริทึมที่ใช้ฝึกดังนี้

อัลกอริทึมที่ 4

Step 1. initial weight, α_{winner} , $\alpha_{neighbor}$ and number of node

Step 2. select 1 character / pattern in training set to be inputs (training set 1)

Step 3. while classify False , do step 4-14

Step 4. for each training input I do step 5-8

Step 5. apply input to input node

Step 6. find winner node (eq. 3.2)

Step 7. update winner and neighbor node

Step 8. if Thai pattern has never allocated winner,

$$signal = signal + 1$$

Step 9. find max-signal node

Step10. find node that weight is max distance from weight of max-signal node (eq. 4.3)

Step11. insert new node between max-signal node and max-distance node

Step12. initial neighborhood node for new node, and weight of new node is (eq. 3.3)

Step13. update signal (eq. 4.7) and neighborhood of new node's neighbor

Step14. if each node is allocated by same-character inputs, classify TRUE

Step 15. repeat step 2 -14 train by all pattern (training set 2 or 3)

Step 16. delete all nodes that non-character allocate

ต่อไปเป็นการทดลองหาค่าพารามิเตอร์ที่เหมาะสมให้โครงข่าย โดยการฝึกด้วยอัลกอริทึมที่ 4 ซึ่งมีการทดลองดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. ทดลองการกำหนดจำนวนโหนดเริ่มต้นที่เหมาะสม

ทดลองกำหนดจำนวนโหนดเริ่มต้นจากที่มี 3 โหนดเป็นมี 10 โหนดเพื่อตัวอักษรจะได้มีโหนดที่จะเลือกกระจายไปตกมากขึ้น แต่ก็พบว่า เมื่อเริ่มต้นฝึกจะมีเพียงไม่กี่โหนดเท่านั้นที่ตัวอักษรเลือกกระจายไปตก ต่อเมื่อมีการเพิ่มโหนดใหม่เข้าไปตัวอักษรจึงจะกระจายไปตกโหนดใหม่นั้น โดยโหนดที่ไม่มีตัวอักษรใดตกในช่วงเริ่มต้นฝึกส่วนใหญ่ก็จะไม่มีตัวอักษรใดตกเช่นเดิมจึงไม่ค่อยมีประโยชน์นักที่จะกำหนดจำนวนโหนดเริ่มต้นให้มีจำนวนมาก ๆ เพราะในที่สุดโหนดเหล่านี้ก็จะถูกลบออก และทำให้เสียเวลาในการที่พยายามปรับค่าน้ำหนักของโหนดเหล่านี้โดยไม่ได้ประโยชน์ใดเลย การค่อย ๆ เริ่มต้นด้วยจำนวนโหนดน้อย ๆ จึงค่อนข้างเหมาะสมกว่า จากผลการทดลองเมื่อนำไปเปรียบเทียบกับการฝึกแบบ SOM จะพบว่า เกิดการการฝึกแบบ ASOM ที่กำหนดให้มีจำนวนโหนดมาก ๆ ในตอนเริ่มต้นจะทำให้เกิดปัญหาคล้ายกับการฝึกแบบ SOM คือมีโหนดที่ไม่เหมาะสมที่ไม่มีตัวอักษรใดตกจำนวนมาก แต่ SOM ไม่มีการแทรกโหนดใหม่เข้าไปจึงทำให้แยกแยะตัวอักษรไม่ได้ ในขณะที่ ASOM มีการแทรกโหนดใหม่เข้าไปได้จึงทำให้มีโหนดที่มีค่าน้ำหนักเหมาะสมที่จะแยกแยะตัวอักษรได้

12. ทดลองหาวิธีการป้อนตัวอักษรเข้าฝึกที่เหมาะสม

ทดลองการฝึกทีเดียวด้วยตัวอักษรทั้งชุดฝึกและการแบ่งฝึกทีละหนึ่งตัวจนแยกได้แล้วจึงฝึกด้วยตัวอักษรในชุดฝึกทั้งหมด วิธีใดจะเหมาะสมกว่า โดยทดลองฝึกด้วยตัวอักษรชุดที่ 3 ซึ่งแต่ละแบบตัวอักษรมี 20 ตัว (รวมทั้งหมด 1500 ตัว) พบว่า จากผลการฝึกเมื่อฝึกไปประมาณ 300 รอบการแยกแยะก็ยังไม่ดีนัก การฝึกโดยการใช้ฝึกตัวอักษรแต่ละแบบให้แยกแยะได้ก่อน แล้วจึงฝึกด้วยชุด 1500 ตัว จะใช้เวลาน้อยกว่า

13. ทดลองว่าลำดับแบบตัวอักษรที่ป้อนเข้าโครงข่ายมีผลต่อประสิทธิภาพการฝึกอย่างไร

โดยเมื่อแยกแยะตัวอักษรในชุดฝึกที่มีตัวอักษรแบบละ 1 ตัวได้แล้ว ก็ทดลองฝึกโดยใช้ชุดฝึกที่ 2 ซึ่งตัวอักษรทั้ง 20 ตัวในแต่ละแบบเรียงเข้าฝึกต่อกัน และชุดที่ 3 ที่ตัวอักษรแต่ละแบบจะเรียงสลับกันทีละตัว ผลการทดสอบการวิเคราะห์ตัวอักษร จะพบว่า การป้อนตัวอักษรแบบต่อเนื่อง (ตัวอักษรแบบเดียวกันป้อนต่อกันจนหมด แล้วจึงป้อนตัวอักษรแบบใหม่) ในการทดลองที่ 8 ของตารางที่ 21 จะมีเปอร์เซ็นต์การวิเคราะห์ที่ดีกว่าป้อนแบบสลับ (การทดลองที่ 9) เล็กน้อย

14. การทดลองหาค่าน้ำหนักเริ่มต้นที่เหมาะสม

ได้ทดลองสุ่มค่าน้ำหนักทดลอง 2 กลุ่มคือ สุ่มให้ค่าอยู่ระหว่าง 0 ถึง 1 (การทดลองที่ 8) และ สุ่มให้ค่าอยู่ระหว่าง 0.4 ถึง 0.6 (การทดลองที่ 15) ซึ่งก็ให้ผลการแยกแยะที่ใกล้เคียงกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 21

การ ทดลองที่	α_{win}	α_{neigh}	จำนวนรอบที่ฝึก	จำนวนโหนดทั้งหมด	จำนวนโหนดที่ใช้	พิมพ์เลเซอร์ (100 %)		พิมพ์เลเซอร์ (50 %)		ถ่ายเอกสาร (100 %)		ถ่ายเอกสาร (50 %)	
						ผิด (ตัว)	% ถูก	ผิด (ตัว)	% ถูก	ผิด (ตัว)	% ถูก	ผิด (ตัว)	% ถูก
1.	0.1	0.04	337	339	193	71	99.53	596	96.03	80	99.47	784	94.77
2.	0.03	0.001	501	503	252	69	99.54	540	96.40	72	99.52	807	95.77
3.	0.06	0.001	389	392	222	48	99.68	388	97.41	55	99.63	550	96.33
4.	0.06	0.002	332	335	214	26	99.83	481	96.79	69	99.54	631	95.79
5.	0.06	0.0005	422	424	228	39	99.74	446	97.03	54	99.64	647	95.69
6.	0.06	0.0007	411	414	229	33	99.78	434	97.12	54	99.64	557	96.29
7.	0.06	0.00075	389	391	223	18	99.80	454	96.97	55	99.63	588	96.08
8.	**0.06**	**0.0008**	**381**	**383**	**218**	**21**	**99.86**	**415**	**97.23**	**48**	**99.68**	**525**	**96.50**
9.	0.06	0.0008	401	403	226	43	99.71	399	97.34	56	99.63	609	95.94
10.	0.06	0.0008	300	570	214	52	99.65	481	96.79	70	99.53	656	95.63

(ต่อ)

ตารางที่ 21 (ต่อ)

การ ทด ลองที่	α_{win}	α_{ncigh}	จำนวนรอบที่ฝึก	จำนวนโหนดทั้งหมด	จำนวนโหนดที่ใช้	พิมพ์เลเซอร์ (100 %)		พิมพ์เลเซอร์ (50 %)		ถ่ายเอกสาร (100 %)		ถ่ายเอกสาร (50 %)	
						ผิด (ตัว)	% ถูก	ผิด (ตัว)	% ถูก	ผิด (ตัว)	% ถูก	ผิด (ตัว)	% ถูก
11.	0.06	0.00085	395	397	229	30	99.80	454	96.97	60	99.60	631	95.79
12.	0.06	0.0009	417	420	240	33	99.78	522	96.52	54	99.64	666	95.56
13.	0.6	0.2, 0.002	153	156	154	171	98.86	945	93.70	222	98.52	1146	92.36
14.	0.6	0.2, 0.003	151	154	152	177	98.82	899	94.01	220	98.53	1106	92.62
15.	0.6	0.002	359	362	222	18	99.88	539	96.41	75	99.50	696	95.36

(ต่อ)

ตารางที่ 21 (ต่อ)

การทดลองที่ 1- 12	ฝึกโครงข่ายตามอัลกอริทึมที่ 4
การทดลองที่ 13	ฝึกโครงข่ายตามอัลกอริทึมที่ 3
การทดลองที่ 1- 13	สุ่มค่าน้ำหนักอยู่ระหว่าง 0 - 1
การทดลองที่ 14 - 15	สุ่มค่าน้ำหนักอยู่ระหว่าง 0.4 - 0.6
การทดลองที่ 10	เพิ่มโหนด 1 โหนด เมื่อฝึกตัวอักษรครบ 75 แบบ

แสดงประสิทธิภาพการวิเคราะห์ตัวอักษร 15,000 ตัว

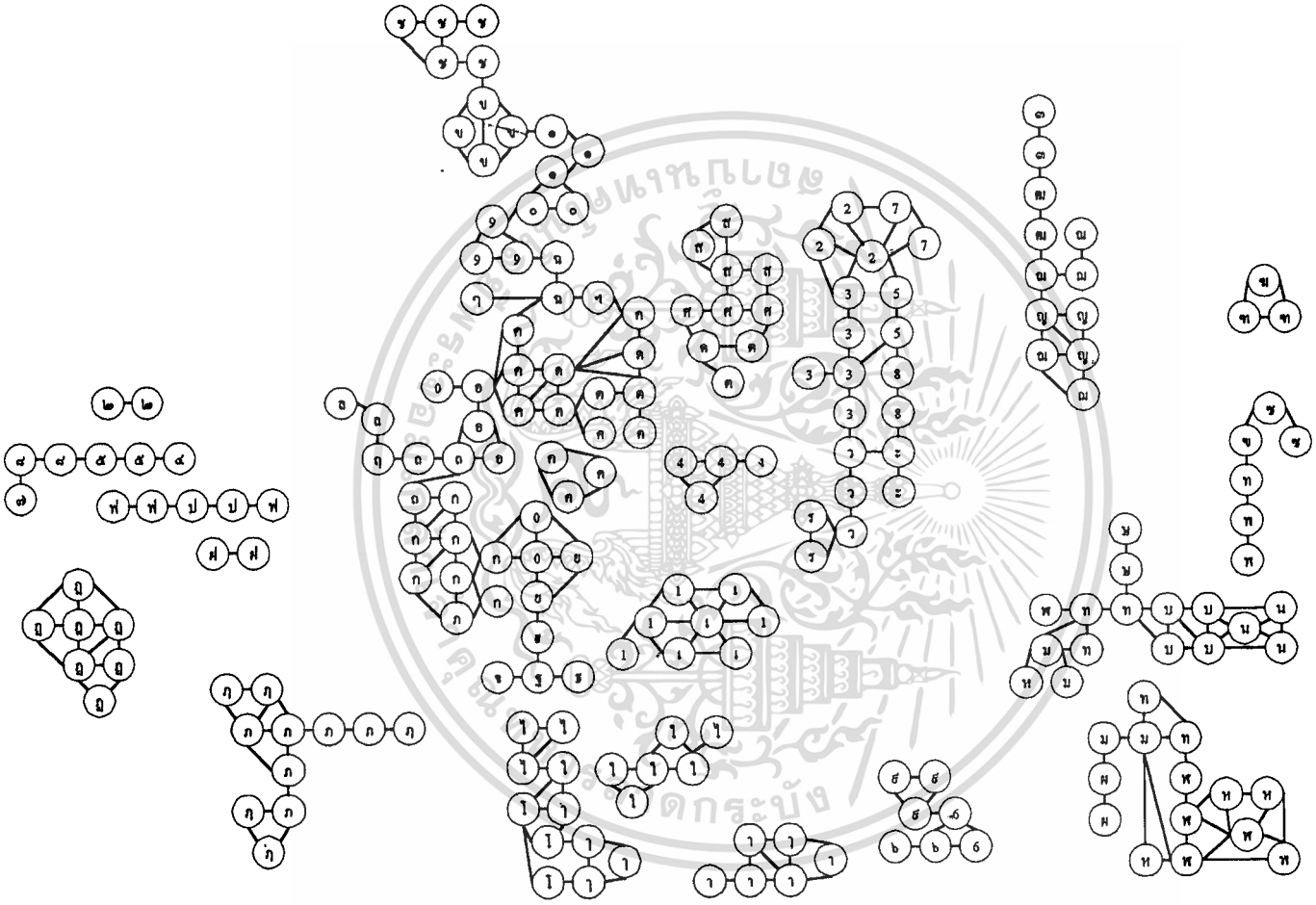
15. จากการฝึกตามอัลกอริทึมที่ 3 ด้วยค่า α_{winner} , $\alpha_{neighbor}$ เท่ากับ 0.6 , 0.2 ตามลำดับโดยใช้ตัวอักษรชุดแรกฝึก เมื่อฝึกไปประมาณ 1000 รอบก็ยังแยกแยะตัวอักษรไม่ได้ทั้งหมด แต่แนวโน้มการแยกแยะจะดีขึ้นตามจำนวนรอบการฝึก จึงคาดได้ว่า จะแยกตัวอักษรได้ถ้าฝึกต่อไป แต่เนื่องจากต้องใช้เวลาฝึกนานและมื่อแยกแยะตัวอักษรได้แล้วก็มีโหนดที่ไม่มีตัวอักษรใดตกจำนวนมากจึงยุติการฝึก การที่ต้องใช้โหนดมากขนาดนี้ก็เนื่องจากค่า α_{winner} , $\alpha_{neighbor}$ มากเกินไป ทำให้ค่าน้ำหนักของโหนดชนะและโหนดข้างเคียงที่ถูกปรับในแต่ละครั้งเข้าสู่สมดุล (converge) ขณะนั้น (local optimal) เท่านั้น ไม่ได้เข้าสู่ค่าสมดุลของโครงข่าย (global optimal) เมื่อป้อนตัวอักษรเข้าฝึกอีก ค่าน้ำหนักก็จะถูกปรับไปปรับมา (oscillate) ค่าที่ได้จึงไม่เสถียร เพื่อให้ค่าน้ำหนักปรับเข้าสู่สมดุลของโครงข่าย จึงทดลองใหม่โดยการฝึกด้วยตัวอักษร 75 แบบ แบบละ 1 ตัว ให้ค่า α_{winner} , $\alpha_{neighbor}$ เท่ากับ 0.6, 0.2 ตามลำดับ เมื่อฝึกจนแยกแยะตัวอักษรได้หมดแล้วให้ฝึกโครงข่ายต่อด้วยชุดตัวอักษร 1500 ตัว โดยลดค่า $\alpha_{neighbor}$ ลงซึ่งได้ทดลองลดเป็น 0.002 และ 0.003 ปรากฏว่า สามารถแยกแยะตัวอักษรได้ทั้งหมด ดังแสดงประสิทธิภาพการจดจำในตารางที่ 21 การทดลองที่ 13 และ 14 ตามลำดับ

16. เมื่อพิจารณาผลการทดลองที่ 1-8 และ 11-12 ในตารางที่ 21 ที่กำหนดค่า α_{winner} , $\alpha_{neighbor}$ ต่าง ๆ กัน พบว่า การใช้ค่า α_{winner} , $\alpha_{neighbor}$ น้อยให้ประสิทธิภาพการจดจำค่อนข้างดีกว่าใช้ค่าสูง ซึ่งค่า α_{winner} , $\alpha_{neighbor}$ ที่ทำให้ประสิทธิภาพการจดจำดีที่สุดคือ 0.6 และ 0.0008 ตามลำดับ (การทดลองที่ 8) โดยฝึก 381 รอบ (ประมาณ 6 ชั่วโมงบนเครื่อง hp 9000/827) และใช้ 218 โหนด ซึ่งแสดงการแมปในภาพที่ 41 จะเห็นว่า ตัวอักษรที่มีลักษณะคล้ายคลึงกันจะอยู่ในตำแหน่งที่ใกล้เคียงกัน ส่วนค่า α_{winner} , $\alpha_{neighbor}$ ที่สูงหรือต่ำกว่านี้ ประสิทธิภาพในการจดจะลดลงเล็กน้อย

เมื่อป้อนชุดฝึกที่มีตัวอักษรแบบละ 1 ตัว รวม 75 ตัวเข้าฝึกจนแยกแยะได้หมดแล้วฝึกต่อด้วยตัวอักษร 1500 ตัว ในชุดฝึกที่ 3 ที่ตัวอักษรมีการสลับกันทีละแบบ โดยจะเพิ่มโหนด 1 โหนดเมื่อป้อนตัวอักษรครบ 75 ตัว ปรากฏว่าสามารถแยกแยะได้หมด ซึ่งแสดงประสิทธิภาพการวิเคราะห์ดังในตารางที่ 21 การทดลองที่ 10 เมื่อเปรียบเทียบผลการทดลอง การเพิ่ม 1 โหนดเมื่อฝึกครบ 1500 ตัว (การทดลองที่ 8-9) จะให้เปอร์เซ็นต์การวิเคราะห์ที่สูงกว่า และใช้เวลาฝึกนานกว่าคือประมาณ 6 ชั่วโมง ส่วนการเพิ่มโหนดเมื่อฝึกครบทุก 75 ตัว (การทดลองที่ 10) จะให้เปอร์เซ็นต์การวิเคราะห์ต่ำกว่าเล็กน้อย แต่ใช้เวลาฝึกน้อยกว่าคือ ประมาณ 4 ชั่วโมง

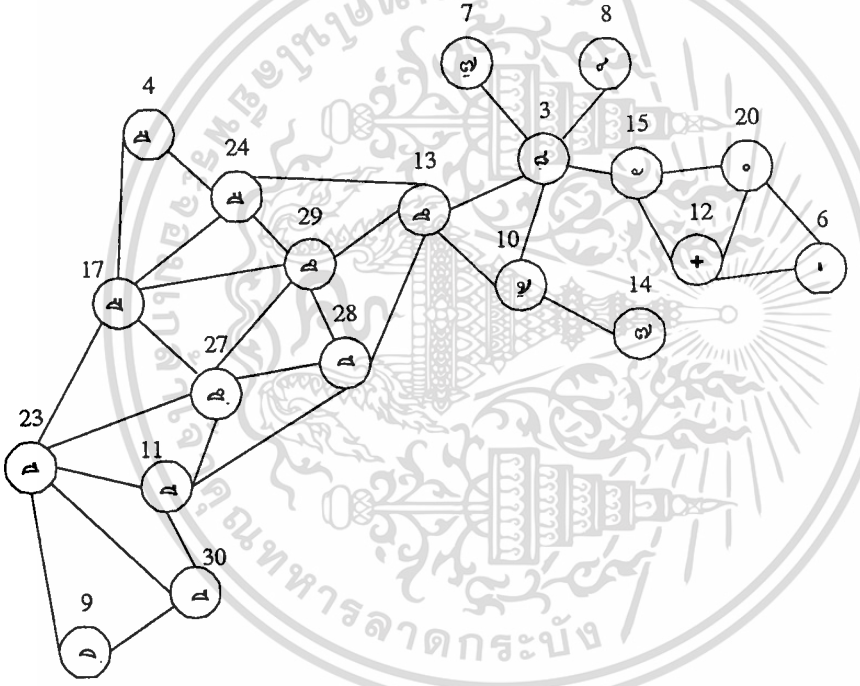
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 41



41.1) การแมปของตัวอักษรระดับกลาง (ต่อ)

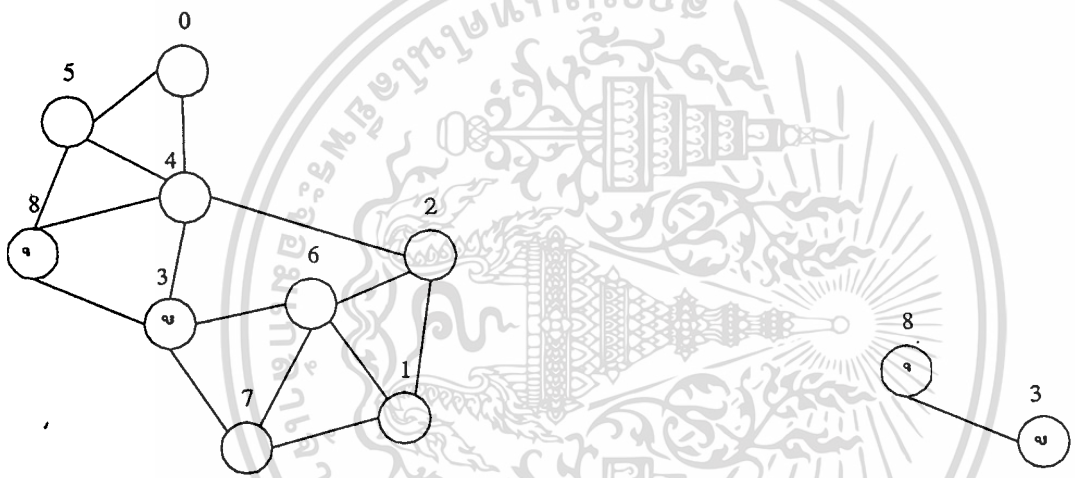
ภาพที่ 41 (ต่อ)



41.2) การแมปตัวอักษรระดับบน

(ต่อ)

ภาพที่ 41 (ต่อ)



การแมปก่อนลบโหนดที่ไม่ได้ใช้ออก

การแมปหลังลบโหนดที่ไม่ได้ใช้ออก

41.2) การแมปตัวอักษรระดับล่าง

แสดงการแมปโครงข่ายเมื่อสิ้นสุดการฝึกและลบโหนดที่ไม่มีตัวอักษรใดตกออกแล้วของตัวอักษรระดับกลาง

ตารางที่ 22

ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)
3	1	1	ด	ต	1	ฤ	ถ	4	า	ว	1
ค	ต	2	ต	ด	3	ล	ท	1	โ	ใ	3
ค	ค	2	ต	ร	1	ฦ	ภ	5	ใ	ไ	2
จ	เ	1	ต	พ	1	ส	ธ	1	ไ	ใ	4
ฌ	ณ	1	ถ	ท	1	ห	ท	2	๐	ย	1
ฎ	ฏ	2	ท	พ	1	ฯ	า	1	๖	6	2
ฏ	ฎ	1	ย	เ	1	ะ	ร	1	ธ	เ	1

แสดงตัวอักษรที่จดจำผิดโดยทดสอบจากตัวอักษร 75 แบบ แบบละ 200 ตัวที่สะแกนจากกระดาษถ่ายเอกสารจากกระดาษที่พิมพ์ด้วยเครื่องพิมพ์เลเซอร์ ความสว่าง (brightness) 100 %

ตารางที่ 23

ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)
5	ฐ	4	ณ	ฒ	3	ย	ฮ	1	า	ว	10
ก	ภ	2	ฎ	ฏ	6	ย	เ	1	เ	1	2
ก	ค	2	ฑ	ฒ	1	ร	ธ	30	โ	ใ	6
ข	ช	7	ณ	ฌ	28	ฤ	ฦ	1	ใ	ไ	32
ค	ต	26	ด	ต	42	ล	ส	1	ไ	ใ	1
ค	ศ	1	ต	พ	10	ว	เ	1	ๆ	ฦ	2
ค	ด	1	ต	ด	2	ศ	ส	1	๓	ฒ	2
ค	ฑ	12	ต	ศ	5	ศ	พ	1	๔	๕	6
ศ	ต	32	ท	พ	76	ศ	ฬ	3	๕	ร	1
ศ	ค	1	ธ	ช	1	ษ	ธ	2	๖	ใ	1
จ	ธ	1	ธ	ข	2	ษ	บ	2	๗	ง	1
จ	ฉ	8	น	บ	2	ส	ศ	2	๘	๒	1
ฉ	ข	1	น	ษ	1	ห	ฬ	13	๘	เ	1
ช	ซ	14	ฝ	ฟ	1	ห	พ	50	ธ	ง	1
ซ	พ	1	ภ	ฦ	1	ห	ท	1			
ซ	ฒ	9	ม	บ	4	ฬ	พ	4			
ซ	พ	9	ย	อ	4	ะ	ร	1			

แสดงตัวอักษรที่จดจำผิดโดยทดสอบจากตัวอักษร 75 แบบ แบบละ 200 ตัวที่สะแกนจากกระดาษถ่ายเอกสารจากกระดาษที่พิมพ์ด้วยเครื่องพิมพ์เลเซอร์ ความสว่าง (brightness) 50 %

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 24

ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)
ต	ค	4	จ	ฉ	2	ด	ต	3	ห	พ	1
โ	ไ	3	ไ	ไ	1	ไ	ไ	7			

แสดงตัวอักษรที่จดจำผิดโดยทดสอบจากตัวอักษร 75 แบบ แบบละ 200 ตัวที่สะแกนจากกระดาษที่พิมพ์จากเครื่องพิมพ์เลเซอร์ ความสว่าง (brightness) 100 %

ตารางที่ 25

ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)
ก	ภ	2	ต	ค	2	ท	พ	62	ห	ฬ	10
ก	ศ	1	ต	ด	1	น	ม	7	ห	พ	22
ก	ท	1	ช	ซ	12	น	ษ	2	โ	ไ	18
ข	ช	11	ฆ	ฒ	5	น	บ	2	ไ	ไ	6
ฃ	ฒ	1	จ	ฉ	13	พ	ห	3	ไ	ไ	1
ค	ต	27	จ	ธ	2	พ	ท	54	๔	๕	1
ค	ฅ	7	ฏ	ฏ	1	บ	ม	1			
ค	ด	9	ณ	ฌ	32	ย	อ	1			
ค	ต	42	ด	ต	25	ร	ธ	29			

แสดงตัวอักษรที่จดจำผิด โดยทดสอบจากตัวอักษร 75 แบบ แบบละ 200 ตัวที่สะแกนจากกระดาษที่พิมพ์จากเครื่องพิมพ์เลเซอร์ ความสว่าง (brightness) 50 %

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 26

ตัวอักษร	เลเซอร์ 100%	เลเซอร์ 50%	ถ่ายเอกสาร 100%	ถ่ายเอกสาร 50%	ตัวอักษร	เลเซอร์ 100%	เลเซอร์ 50%	ถ่ายเอกสาร 100%	ถ่ายเอกสาร 50%
ก	-	4	-	1	ฤ	-	-	4	1
ข	-	11	-	7	ล	-	-	1	1
ฃ	-	1	-	-	ฦ	-	-	5	1
ค	-	27	-	59	ว	-	-	-	1
ฅ	-	45	2	32	ศ	-	-	-	5
ฆ	-	5	-	-	ษ	-	-	-	5
ง	2	15	1	9	ส	-	-	1	2
จ	-	-	-	1	ห	1	32	2	6
ช	-	12	-	14	ฬ	-	-	-	4
ซ	-	-	-	19	อ	-	-	1	1
ฌ	-	-	1	3	ฮ	-	-	1	10
ญ	-	32	-	28	เ	-	-	-	2
ฎ	-	1	2	6	โ	3	18	3	6
ฏ	-	1	-	-	ใ	1	6	2	32
ฑ	-	-	-	1	ไ	7	1	4	1
ฒ	3	25	1	42	ๆ	-	-	-	2
ต	-	-	5	17	๓	-	-	1	-
ถ	-	-	1	-	๗	-	-	-	2
ท	-	62	1	76	๔	-	1	-	6
ธ	-	-	-	3	๕	-	-	-	1
น	-	11	-	3	๖	-	-	2	1
บ	-	1	-	-	๗	-	-	-	1
ฝ	-	-	-	1	๘	-	-	-	2
ภ	-	-	-	1	๙	-	-	1	1
ม	-	-	-	1	ฯ	-	-	1	-
พ	-	54	-	1	๐	-	-	1	-
ย	-	1	1	6	5	-	-	1	-
ร	-	26	-	30					

แสดงการเปรียบเทียบจำนวนตัวอักษรแต่ละแบบที่โครงข่ายจดจำผิด เมื่อทดสอบตัวอักษรจากกระดาษซึ่งถ่ายเอกสารจากกระดาษที่พิมพ์จากเครื่องพิมพ์เลเซอร์ ความสว่าง 100 และ 50 % กระดาษที่พิมพ์จากเครื่องพิมพ์เลเซอร์ ความสว่าง 100 และ 50 %

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 42

แสดงตัวอย่างภาพตัวอักษรที่ ASOM จดจำได้ เรียงตามลำดับดังนี้ ค,ต,ด,ค,ห,โ

ภาพที่ 43

แสดงตัวอย่างภาพตัวอักษรที่ ASOM จดจำผิด เรียงตามลำดับดังนี้ ค,ค,ต,ใ,ห,โ

จากตารางที่ 22-26 จะเห็นว่าตัวอักษรที่มีความคล้ายคลึงกันมากจะมีโอกาสแยกแยะผิดได้สูงกว่าตัวอักษรที่มีลักษณะแตกต่างจากอักษรตัวอื่น เพราะการมีสัญญาณรบกวนเพียงเล็กน้อยก็จะทำให้วิเคราะห์ผิดแล้ว

17. จากผลการทดลองที่กล่าวในข้างต้นเมื่อฝึกโครงข่ายจนแยกแยะตัวอักษรได้หมดแล้ว จะลบโหนดที่ไม่มีตัวอักษรใดตกออกก่อน แล้วจึงทำการวิเคราะห์ ต่อไปจะทดลองวิเคราะห์โดยเมื่อฝึกโครงข่ายจนแยกแยะได้หมดแล้ว จะยังไม่ลบโหนดที่ไม่มีตัวอักษรใดตกออก โดยถ้าในการวิเคราะห์หากมีตัวอักษรใดตกยังโหนดที่ไม่ได้เป็นตัวแทนตัวอักษรใดเลย และถ้ากลุ่มตัวอักษรที่ตกยังโหนดที่ไม่ได้เป็นตัวแทนตัวอักษรใดเลยนี้เป็นตัวอักษรแบบเดียวกัน ก็จะให้โหนดนี้เป็นตัวแทนแบบตัวอักษรที่ตกโหนดนี้เลย ผลจากการวิเคราะห์จะให้ประสิทธิภาพการจดจำที่ต่ำประมาณ 90 เปอร์เซ็นต์ จึงไม่เหมาะที่จะวิเคราะห์ตัวอักษรด้วยวิธีนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

สรุปผลการทดลอง

การฝึกโครงข่ายประสาทเทียมแบบ SOM โดยกำหนดค่าน้ำหนักเริ่มต้นแบบสุ่ม อัตราการเรียนรู้เริ่มต้นและสุดท้ายเท่ากับ 0.6 และ 0.001 ขอบเขตเริ่มต้นของโหนดข้างเคียงจะห่างจากโหนดชนะออกไป 5 ชั้น หากกำหนดขนาดโครงข่ายเล็กเกินไปจะแยกแยะตัวอักษรไม่ได้ เนื่องจากโครงข่ายมีขนาดและตำแหน่งคงที่ บางโหนดจึงอาจอยู่ในตำแหน่งที่ไม่เหมาะสมที่ตัวอักษรจะมาตก เมื่อเพิ่มขนาดโครงข่ายให้ใหญ่เพียงพอจะแยกแยะแบบตัวอักษรทั้งหมดได้ แต่การเพิ่มขนาดโครงข่ายทำให้จำนวนค่าน้ำหนักที่อยู่นอกขอบเขตของชุดข้อมูลฝึกเพิ่มตามด้วย จากการทดลองพบว่าการแยกแยะตัวอักษร 75 แบบแบบละ 1 ตัวด้วย SOM ต้องใช้ขนาดโครงข่าย 25x25 โหนดซึ่งโหนดที่มีแบบตัวอักษรตกมีเพียง 75 โหนดมีโหนดที่ไม่ได้ใช้ประโยชน์เลยถึง 88 เปอร์เซ็นต์

นอกจากนี้ก็ได้ทดลองปรับค่าพารามิเตอร์ต่าง ๆ ที่ใช้ในการฝึกโครงข่าย คือ การกำหนดค่าน้ำหนักเริ่มต้นและการปรับค่าน้ำหนักระหว่างการฝึก, การกำหนดค่าขอบเขตโหนดข้างเคียงเริ่มต้นและวิธีการวัดขอบเขต, การกำหนดค่าอัตราการเรียนรู้เริ่มต้นและการปรับค่าอัตราการเรียนรู้ระหว่างฝึก, การปรับเปลี่ยนลำดับตัวอักษรที่ป้อนเข้าฝึก แต่เมื่อใช้ชุดตัวอักษร 1500 ตัวฝึกในโครงข่ายขนาด 25x25 โหนดก็ไม่สามารถแยกแยะตัวอักษรได้ทั้งหมด จะต้องเพิ่มขนาดโครงข่าย และเริ่มทดลองฝึกใหม่ซึ่งทำให้เสียเวลามาก แบบจำลอง SOM จึงไม่เหมาะที่จะนำมาประยุกต์ใช้ในการจดจำอักษรภาษาไทย

จากเหตุผลข้างต้นจึงใช้ ASOM ที่มีความยืดหยุ่นกว่าในการเพิ่มและลดโหนดในระหว่างการฝึก โดยฝึกตามอัลกอริทึมที่ 2 แต่อัลกอริทึมนี้ก็ทำให้ติดปัญหาคือ ปัญหาการลบและเพิ่มโหนดที่ไม่เหมาะสม จึงมีการปรับอัลกอริทึมดังนี้คือ การนับสัญญาณการตกให้นับแบบตัวอักษรที่ตกในแต่ละโหนดในระหว่างการฝึก (แทนการนับจำนวนตัวอักษรที่ตกในแต่ละโหนด และจะไม่มีการลดสัญญาณการตกในโหนดอื่น ๆ ที่ไม่มีตัวอักษรใดตกเพื่อลำดับการป้อนตัวอักษรเข้าฝึกจะได้ไม่มีผลต่อสัญญาณการตก) และให้เพิ่มโหนดในตำแหน่งที่มีแบบตัวอักษรตกแตกต่างกันสูงสุด และค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณการตกจะถูกกำหนดใหม่ให้เท่ากับศูนย์ทุกครั้งที่ฝึกตัวอักษรในชุดฝึกครบ 1 ครั้ง (ซึ่งเรียกว่าครบ 1 รอบ) และในช่วงแรกของการฝึกจะเพิ่มโหนดอย่างเดียวจนแยกแยะตัวอักษรได้หมดจึงลบโหนดที่ไม่มีตัวอักษรใดตกออก ซึ่งสามารถสรุปสิ่งที่ปรับเปลี่ยนไปได้ดังแสดงในอัลกอริทึมที่ 3 นอกจากนี้ได้ทดลองกำหนดค่าอัตราการเรียนรู้ของโหนดชนะ และโหนดข้างเคียงให้มีค่าต่าง ๆ กัน พบว่า ค่าอัตราการเรียนรู้ที่สูงเกินไป เช่น อัตราการเรียนรู้ของโหนดชนะและโหนดข้างเคียงเท่ากับ 0.6, 0.2 ตามลำดับ ทำให้ใช้เวลาฝึกโครงข่ายนานเพราะค่าน้ำหนักถูกปรับไปปรับมาเข้าสู่ค่าเสถียรช้า จึงปรับอัตราการเรียนรู้ให้ลดลง จากข้อมูลเข้า (ตัวอักษรที่ใช้ฝึก 1500 ตัว) ที่ใช้ในงานวิจัยนี้ พบว่า ค่าอัตราการเรียนรู้ของโหนดชนะและโหนดข้างเคียงที่เหมาะสมที่สุดในการนำมาประยุกต์ใช้ในการจดจำอักษรภาษาไทยในชุดการฝึกนี้คือ 0.06 และ 0.0008 ตามลำดับ โดยใช้จำนวนรอบในการฝึก 381 รอบ ใช้ 218 โหนด เวลาที่ใช้ในการฝึกประมาณ 6 ชั่วโมง และประสิทธิภาพในการจดจำตัวอักษรจากข้อมูลที่พิมพ์ด้วยเลเซอร์ ที่สะแกนที่ความสว่าง 100 เปอร์เซ็นต์ และ 50 เปอร์เซ็นต์ และจากข้อมูลที่ได้จากการถ่ายเอกสารจากข้อมูลที่พิมพ์ด้วยเลเซอร์ความสว่าง 100 เปอร์เซ็นต์ และ 50 เปอร์เซ็นต์ เท่ากับ 99.86, 97.23, 99.68 และ 96.50 เปอร์เซ็นต์ตามลำดับ

จากการป้อนแบบตัวอักษรเดียวกันต่อเนื่องกันเข้าฝึก กับการสลับตัวอักษรตัวต่อตัว พบว่าประสิทธิภาพในการจดจำใกล้เคียงกัน ลำดับการป้อนแบบตัวอักษรเข้าฝึกไม่มีผลต่อประสิทธิภาพการจดจำ

การเพิ่มโหนดทุกครั้งที่ฝึกครบ 1500 ตัวและเพิ่มโหนดทุกครั้งที่ฝึกครบ 75 ตัว มีประสิทธิภาพการจดจำใกล้เคียงกัน การฝึกครบ 1500 ตัวแล้วเพิ่ม 1 โหนดมีจำนวนโหนดก่อนการลบโหนด 403 โหนด ส่วนการเพิ่มโหนดทุกครั้งที่ฝึกครบ 75 ตัวมีจำนวนโหนดก่อนการลบโหนด 570 โหนด ซึ่งการลบโหนดจะลบเมื่อแยกแยะตัวอักษรได้แล้ว หากมีโหนดที่ไม่ได้ใช้ก่อนที่จะแยกแยะตัวอักษรได้หมดทำให้ต้องเสียเวลาการฝึกโหนดดังกล่าวโดยไม่ได้ผลการฝึกที่ดีขึ้นเลย จึงควรฝึกโดยการเพิ่มโหนดใหม่ทุกครั้งที่ฝึกครบ 1 รอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 27

ชนิดกระดาษที่ทดสอบ (% ความสว่าง)	Backpropagation	Counterpropagation (CPN)	Adaptive Self-Organizing Maps (ASOM)
พิมพ์เลเซอร์ (100 %)	99.96	99.86	99.86
พิมพ์เลเซอร์ (50 %)	98.90	97.34	97.23
ถ่ายเอกสาร (100 %)	99.74	99.68	99.68
ถ่ายเอกสาร (50 %)	96.23	95.69	96.50
เวลาที่ใช้ฝึก (ชั่วโมง)	336	0.6	6

แสดงการเปรียบเทียบประสิทธิภาพในการวิเคราะห์ตัวอักษรโดยโครงข่ายแบบ Backpropagation Counterpropagation (CPN) และ Adaptive Self-Organizing Maps (ASOM)

ตารางที่ 27 แสดงการเปรียบเทียบเปอร์เซ็นต์ความถูกต้องในการวิเคราะห์ตัวอักษรระหว่างโครงข่ายแบบ Backpropagation, Counterpropagation (CPN) และ Adaptive Self-Organizing Maps (ASOM) โดยในแต่ละโครงข่ายใช้ตัวอักษรฝึกชุดเดียวกัน (1500 ตัว) และทดสอบการวิเคราะห์ตัวอักษรด้วยชุดทดสอบเดียวกันด้วย (15,000 ตัว) จากการเปรียบเทียบโครงข่ายทั้งสามแบบ พบว่า โครงข่ายแบบ Backpropagation จะมีเปอร์เซ็นต์ความถูกต้องสูงกว่าโครงข่ายแบบ CPN และ ASOM เล็กน้อย แต่ใช้เวลาฝึกโครงข่ายนานกว่า CPN และ ASOM มาก เมื่อเปรียบเทียบระหว่าง CPN และ SOM จะพบว่า มีเปอร์เซ็นต์ความถูกต้องใกล้เคียงกันมาก แต่ ASOM ใช้เวลาฝึกนานกว่า อย่างไรก็ตาม แม้การฝึกโครงข่ายแบบ CPN และ ASOM จะใช้วิธีการเพิ่มโหนดให้โครงข่ายเหมือนกัน การฝึกโดย ASOM จะอัตโนมัติกว่า CPN คือ เมื่อป้อนชุดตัวอักษรเข้าฝึกโครงข่ายจะจัดการเพิ่มหรือลดโหนดเองโดยไม่ต้องวิเคราะห์ด้วยสายตาก่อน (manual) ในขณะที่CPN จะมีการฝึกโครงข่ายกึ่งอัตโนมัติ คือ จะต้องมีการวิเคราะห์ลักษณะความเหมือนและความแตกต่างกันของตัวอักษรก่อน เพื่อเป็นเกณฑ์ในการกำหนดค่าเทรสโฮลด์ที่เป็นค่าที่ใช้พิจารณาในการเพิ่มโหนด ซึ่งก็ต้องเสียเวลาในส่วนนี้ด้วย และในขั้นตอนนี้ผู้ที่จะวิเคราะห์เพื่อกำหนดค่าเทรสโฮลด์จะต้องมีประสบการณ์พอสมควร ค่าเทรสโฮลด์ที่กำหนดขึ้นจะได้เป็นค่าที่เหมาะสมเพื่อให้ประสิทธิภาพในการวิเคราะห์ดีที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นในระหว่างโครงข่ายทั้งสามแบบ ASOM จึงค่อนข้างจะเหมาะสมในการใช้วิเคราะห์ตัวอักษร เพราะ เปอร์เซ็นต์การวิเคราะห์ตัวอักษรค่อนข้างใกล้เคียงกัน ขณะที่ Backpropagation ใช้เวลาฝึกนานกว่ามาก และ CPN ก็ต้องยุ่งยากในการวิเคราะห์ตัวอักษรก่อนทำการฝึก

แนวทางการพัฒนาต่อไปในอนาคต

1. ปรับปรุงให้โครงข่ายสามารถวิเคราะห์ตัวอักษรได้หลากหลายมากกว่านี้ คือวิเคราะห์ตัวอักษรได้หลายฟอนต์และหลายขนาด โดยอาจใช้เทคนิคอื่นเข้าช่วยด้วย

2. เพื่อเพิ่มความถูกต้องในการวิเคราะห์ให้สูงขึ้น นอกจากจะวิเคระห์ตัวอักษรจากความรู้ที่โครงข่ายเรียนรู้มาซึ่งเป็นการวิเคราะห์ตัวอักษรทีละตัว ก็ให้มีการวิเคราะห์ในลักษณะเป็นคำโดยหลังจากวิเคราะห์ทีละตัวแล้ว ก็ให้ทำการตัดคำแล้วนำคำไปเทียบกับคำในพจนานุกรม ในกรณีที่คำนั้นไม่มีในพจนานุกรมแต่วิเคราะห์ได้ถูกต้อง ก็ให้เพิ่มคำลงในพจนานุกรม ถ้าคำนั้นไม่มีในพจนากรมและผลการวิเคราะห์ผิด ก็ให้หาคำในพจนานุกรมที่มีความใกล้เคียงคำนั้น ๆ แทน ก็จะทำให้การวิเคราะห์ตัวอักษรถูกต้องมากขึ้น หรืออีกวิธีก็อาจใช้ฟัซซีมาช่วยในการวิเคราะห์

3. การวิเคราะห์ตัวอักษรในงานวิจัยนี้ เมื่อป้อนตัวอักษรที่ต้องการวิเคราะห์เข้าไปในโครงข่าย แล้วทำการค้นหาตัวอักษรที่โครงข่ายเรียนรู้และจดจำได้ในระหว่างการฝึก โดยการค้นหาจะทำการเปรียบเทียบตัวอักษรที่ต้องการวิเคราะห์กับตัวอักษรทุก ๆ ตัวที่โครงข่ายจดจำได้ ตัวอักษรใดในโครงข่ายที่เหมือนตัวอักษรที่ต้องการวิเคราะห์มากที่สุด ผลการวิเคราะห์ก็จะออกมาเป็นอักษรตัวนั้น ซึ่งการค้นหานี้จะค้นไปตามลำดับตัวอักษรที่โครงข่ายจดจำได้ (sequential search) ซึ่งอาจจะช้า จึงอาจหาวิธีการค้นหาที่มีประสิทธิภาพกว่า เช่น การใช้ทฤษฎีของเจนเนติคอัลกอริทึม (genetic algorithm)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. Bernd Fritzke, "Growing cell structures A self-Organizing network for unsupervised and unsupervised learning", Neural Network. Vol. 7 no. 9 (1994) pp. 1441-1460
2. Bernard Widrow and Michael A. Lehr, "30 Years of Adaptive Neural Networks: Perceptron, Madaline, and Backpropagation", Proceeding of IEEE Vol. 78 no. 9 (1990) pp. 1415 - 1441
3. Jack M. Zurada. Artificial neural systems. Singapore: West Publishing Company, 1992
4. James A. Freeman and David M. Skapura. Neural Networks Algorithms, Applications, and Programming Techinques. Massachusetts : Addison-Wesley, 1991.
5. Laurene Fausett. Fundamentals of Neural Networks Architectures, Algorithms, and Applications. New Jersey : Prentice Hall International Inc, 1994.
6. Maureen Caudill, " A Little Knowledge is a Dangerous Thing", AI Expert June (1993) pp. 16 - 22
7. Maureen Caudill and Charles Butler. Understanding neural networks:Computer Explorations. Massachusetts : The MIT press, 1993
8. Robert Hecht-Nielsen. Neurocomputing. Massachusetts : Addison-Wesley, 1990.
9. Valluru B. Rao and Hayagriva V. Rao. C++ Neural Networks and Fuzzy Logic. USA : MIS press, 1993.
10. ชม กิ้มปาน. การรู้จำรูปแบบอักษรพิมพ์ภาษาไทยของคอมพิวเตอร์. วิทยานิพนธ์ วิศวกรรมศาสตรดุษฎีบัณฑิต. กรุงเทพฯ : บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2529.
11. ประสาร ตังตสานนท์. การจดจำรูปแบบตัวอักษรคัดลายมือภาษาไทย โดยใช้วิธีแยกลักษณะเด่น. วิทยานิพนธ์ วิศวกรรมศาสตรมหาบัณฑิต. กรุงเทพฯ : บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2529.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12. สุรพันธ์ เอื้อไพบูลย์. การจดจำอักษรลายมือเขียนภาษาไทยโดยการพิจารณาหัวตัวอักษร. วิทยานิพนธ์ วิศวกรรมศาสตรมหาบัณฑิต. กรุงเทพฯ : บัณฑิตวิทยาลัย สถาบันเทคโนโลยี-พระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2531.

13. ชาย เกษมอมรกุล. การออกแบบพจนานุกรมสำหรับการเรียนรู้อักษรลายมือไทย-อังกฤษอัตโนมัติบนเครื่องไมโครคอมพิวเตอร์. วิศวกรรมศาสตรมหาบัณฑิต. กรุงเทพฯ : บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2532.

14. สุรสิทธิ์ ราตรี. การรู้จำอักษรตัวพิมพ์ภาษาไทยโดยวิธีค้นหาลักษณะโครงสร้างลายเส้น. วิทยานิพนธ์วิศวกรรมศาสตรมหาบัณฑิต. กรุงเทพฯ : บัณฑิตวิทยาลัย สถาบันเทคโนโลยี-พระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2532.

15. สมศักดิ์ วลัยรัชต์. การวิเคราะห์และระบุส่วนประกอบของหน้าเอกสารและการรู้จำอักษร. วิทยานิพนธ์ วิศวกรรมศาสตร์มหาบัณฑิต. กรุงเทพฯ : บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2537.

16. ทวี เปรมรัตนชัย. การรู้จำอักขระตัวพิมพ์ภาษาไทยโดยวิธีโครงข่ายนิวรอลแบบแบลกพรอพา - เกชั่น วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต. กรุงเทพฯ : บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2539.

17. อภิรักษ์ จิรายุสกุล. การวิเคราะห์ตัวพิมพ์อักษรภาษาไทยโดยใช้ CPN วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต. กรุงเทพฯ : บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2539.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การจดจำอักษรพิมพ์ภาษาไทยโดยโครงข่ายประสาทเทียมแบบจำลองเซลท์ออร์แกนไนซิ่งแมบซ์

Recognizing Printed Thai Character with Self-Organizing Maps Neural Network

ดร.บุญธีร์ เครือตราชู อัญชลี วานิชทวีวัฒน์
คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

บทคัดย่อ

งานวิจัยนี้ทำการจดจำตัวอักษรไทยโดยใช้โครงข่ายประสาทเทียมแบบ Self-Organizing Maps(SOM)และAdaptive SOM(ASOM)ที่มีความยืดหยุ่นกว่าในการปรับโครงสร้างและขนาดโครงข่าย พร้อมทั้งแสดงการปรับเปลี่ยนอัลกอริทึมในการฝึกASOMเพื่อให้เหมาะสมในการจดจำตัวอักษรไทย นอกจากนี้ก็จะรายงานการเปรียบเทียบระหว่างSOM และ ASOM ในแง่ของเวลาที่ใช้ในการฝึก จำนวนโหนดที่ใช้ และประสิทธิภาพในการจดจำ

จากการฝึกด้วยฟอนต์อังศนายูพีซี ขนาด 12 จุดต่อนิ้ว 75 แบบ จำนวน1500 ตัว ด้วยแบบจำลอง SOMจะต้องเสียเวลานานในการหาโครงข่ายที่เหมาะสม ส่วนASOM จะใช้เวลาฝึกน้อยกว่าประมาณ 6 ชั่วโมง (บนเครื่อง hp 9000/827) 318 รอบ ใช้ 218 โหนด ประสิทธิภาพในการจดจำจากการทดสอบด้วยตัวอักษร 15000 ตัวที่เป็นตัวอักษรที่คมชัดและตัวอักษรที่มีสัญญาณรบกวนมีค่าเฉลี่ยเท่ากับ 99.77 , 96.87 เปอร์เซ็นต์ตามลำดับ

Abstract

This research used Self-Organizing Maps neural network(SOM) and Adaptive SOM(ASOM) to recognize the printed Thai character. ASOM is more flexible than SOM in term of neural structure(node and connection). ASOM's training algorithm are modified to use in Thai character recognition. Furthermore, SOM and ASOM are presented in comparision with learning time, number of neural nodes and recognizing rate and efficiency.

The number of character use in training network are 1500 12-point Angsana UPC font which have 75 Thai character patterns. The SOM takes more training time to find the suitable network. However, the ASOM takes about 6 hours(on computer model, hp 9000/827), 381 cycles and 218 nodes. For testing, uses 15,000 characters which are unnoise characters and noise characters, the average percents of recognition are 99.77 and 96.87, respectively.

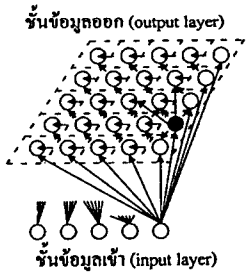
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. บทนำ

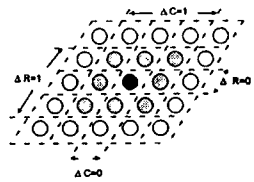
ปัจจุบันมีการพัฒนางานวิจัยด้านการจดจำตัวอักษรภาษาไทยไว้หลายวิธี แต่ละวิธีก็มีจุดเด่นและด้อยต่างกัน เช่นวิธีหาคุณสมบัติทางโทโพโลยี (topology) ของตัวอักษร[5]หากตัวอักษรไม่ชัดเจนหรือมีลักษณะคล้ายคลึงกันจะทำให้การวิเคราะห์ผิดพลาดได้ง่ายและวิธี backpropagation[6] ก็ต้องใช้เวลาฝึกนาน งานวิจัยนี้จึงใช้โครงข่ายประสาทเทียมแบบSelf-Organizing Maps(SOM)ซึ่งมีจุดเด่นที่ใช้เวลาฝึกน้อยกว่า แต่SOMมีโครงสร้างและขนาดโครงข่ายคงที่ตลอดที่อาจไม่เหมาะสมจึงใช้แบบAdaptive SOM(ASOM)ที่เพิ่มและลดโหนดได้ในระหว่างการฝึก นอกจากนี้ได้ปรับอัลกอริทึม ASOMให้เหมาะสมกับการจดจำตัวอักษรไทยด้วย

2. สถาปัตยกรรม SOM

สถาปัตยกรรมSOMเสนอโดยKohonen[3,4]ดังรูปที่1 มีสองชั้นโดยระหว่างชั้นจะเชื่อมกันแบบทั่วถึงด้วยค่าน้ำหนักในชั้นinputจะรับข้อมูลเข้า(input)อย่างเดียว ส่วนชั้น competitive ที่มีโหนดวางเรียงกันในระนาบสองมิติจะมีการเรียนรู้ตามอัลกอริทึมที่ 1 ซึ่งมีการเลือกโหนดชนะ(winner)ดังสมการที่1 แล้วปรับค่าน้ำหนักให้โหนดชนะและโหนดข้างเคียง(neighbor)ดังสมการที่2 ที่แสดงการปรับน้ำหนักได้ดังรูปที่ 3 โดยโหนดข้างเคียงจะมีเป็นกรอบสี่เหลี่ยมซ้อนเป็นชั้นๆ รอบโหนดชนะ (ชั้นแรกΔC,ΔRเท่ากับ1)ดังรูปที่2 เมื่อแยกแยะตัวอักษรได้หมดค่าน้ำหนักแต่ละโหนดคือ ค่าน้ำหนักเฉลี่ยที่แทนตัวอักษรซึ่งอยู่ในกลุ่ม(class) เดียวกันดังรูปที่4 ซึ่งตัวอักษรแต่ละแบบจะอยู่คนละกลุ่มกัน แต่ตัวอักษรหนึ่งแบบอาจมีหลายกลุ่มได้



รูปที่ 1 แสดงสถาปัตยกรรม SOM



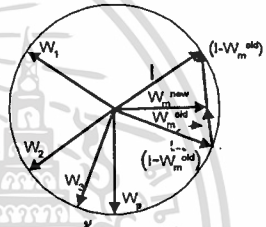
รูปที่ 2 แสดงขอบเขตโหนดข้างเคียง(neighbor)
● โหนดชนะ ◎ โหนดข้างเคียงโหนดชนะ
ΔC,ΔR ขอบเขตโหนดข้างเคียงในแนวแกน x และ y

ทดลองใช้โครงข่ายที่การเรียงตัวกันของโหนด competitiveเป็นแบบสี่เหลี่ยมจตุรัสขนาด 10x10, 20x20 และ25x25 โดยΔCและΔR เริ่มจาก 5 ชั้นจนถึง 0 ชั้น

$$\left\| I - W_{WINNER} \right\| = \underset{c \in A}{MIN} \left\| I - W_c \right\| \quad (1)$$

$$W_c^{new} = W_c^{old} + \alpha\left(I - W_c^{old} \right) \quad (2)$$

โดย $c \in N_c, winner$



รูปที่ 3 แสดงการปรับน้ำหนักในรูปเวคเตอร์น้ำหนักและเวคเตอร์ข้อมูลเข้าขนาด 1 หน่วย

3. อัลกอริทึมที่ 1 ใช้สอน SOM

step 0 initial weight, neighborhood region(N_c=5),
start and stop learning rate (α_s=0.6,α_e=0.001)
Step 1 while not classify or stop condition False
Step 2.while $N_c \geq 0$
Step 3. while $\alpha \geq \alpha_e$
Step 4. for each training input I
Step 5. apply input to input node
Step 6. find winner node (eq. 1)
Step 7. update weight for winner
and neighborhood node(eq. 2)
Step 8 reduce $\alpha = \alpha / 2$
Step 9. reduce $N_c = N_c - 1$
Step 10. stop condition TRUE

สัญลักษณ์ต่าง ๆ :

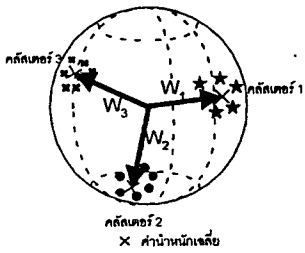
I เวคเตอร์ค่าข้อมูลเข้า (input)
W_c เวคเตอร์น้ำหนัก(weight)ที่เชื่อมกับโหนด c
$\| x^2 - y^2 \|$ ผลต่างยกกำลังสองของ x และ y

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

winner โหนดที่เลือกเป็นโหนดชนะ

N_c โหนดข้างเคียงโหนด c (neighborhood)

α_s, α_e อัตราการเรียนรู้เริ่มต้นและสุดท้ายตามลำดับ



รูปที่ 4 แสดงค่าน้ำหนักของโครงข่ายหลังการฝึก ซึ่งเป็นน้ำหนักเฉลี่ยของกลุ่ม(class)นั้น

4. การทดลอง SOM

งานวิจัยนี้ ใช้ตัวอักษรพิมพ์ ฟอนท์ (font) AngsanaUPC ขนาด 12 พอยท์(point) ความละเอียดการสะแกน(scan)ภาพอักษรคือ 300dpi และscale 100 ในการสอนและทดสอบการจดจำโดยประกอบด้วย พยัญชนะไทย 44 ตัว เลขไทย เลขอังกฤษ า ิ ี ไ ใ ๆ ่ ฤๅ และไม้หันอากาศ รวม 75 ตัว ให้เมตริกของข้อมูลเข้า (input) มีขนาด 28 x 38 โดยวางตัวอักษรที่เป็นข้อมูลเข้า (input) ที่ตำแหน่งชิดบนด้านซ้ายของเมตริก

ชุดตัวอย่างที่ใช้ฝึกมีตัวอักษร 75 แบบแบ่งเป็น 2 ชุด ชุดแรกมีตัวอักษรแบบละ 1 ตัว ชุดสองมีตัวอักษรแบบละ20ตัวรวม1500 ตัว กำหนดค่าน้ำหนัก (weight) เริ่มต้นแบบสุ่มโดยมีค่าอยู่ระหว่าง 0 ถึง 1 และค่าอัตราการเรียนรู้เริ่มต้นและสุดท้ายเท่ากับ 0.6 และ 0.001 ตามลำดับ โดยฝึกตามอัลกอริทึมที่ 1 ทั้งสองชุดการฝึก

5. ผลการทดลอง SOM

ตารางที่1-2แสดงการแยกแยะตัวอักษรที่ดีขึ้นตามลำดับในการฝึกรอบที่ 20 และ 60 โดยโครงข่ายมีขนาด10x10โหนด และใช้ชุดการฝึกแรก แต่บางโหนดยังมีตัวอักษรหลายแบบเลือกโหนดนั้นเป็นตัวแทน (มีตัวอักษรตก)จึงทดลองปรับขนาดโครงข่ายเป็น 20x20 และ25x25 โหนด โครงข่ายขนาด 25x25 โหนด จึงสามารถแยกแยะแบบตัวอักษรได้ทั้งหมดดังผลที่แสดงในตารางที่ 3

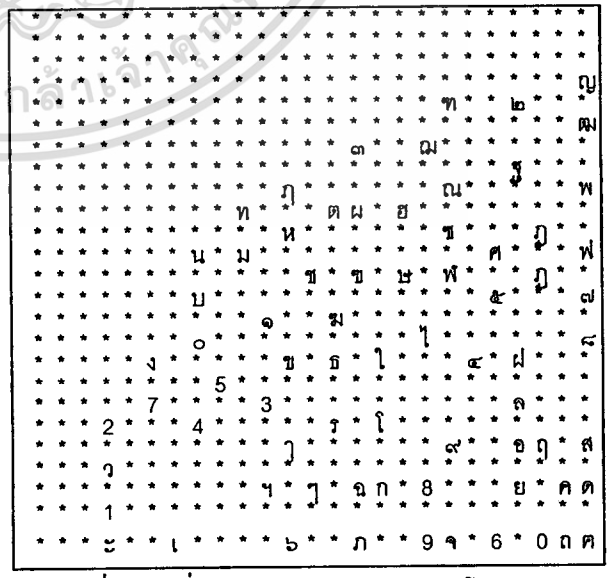
เมื่อฝึกด้วยตัวอักษร 75 แบบแบบละ 20 ตัว รวม1500 ตัวโดยโครงข่ายขนาด 25x25 เมื่อสิ้นสุดการฝึกก็ยังแยกแยะตัวอักษรไม่ได้ ดังแสดงในตารางที่ 4 จะต้องทดลองเพิ่มขนาดโครงข่ายให้ใหญ่ขึ้นแล้วเริ่มต้น

ฎญฐ ๒พ ๕ณ ฒฎ			ฮ	ผ ฝ	ฦ ๑	ลส	ม	8 ๙	อดธ ฤคข 6ย
ศฌ ฟพ		ต		๓					9ตจ 0ถ
	ฑ								
	๗								
ฃ			๔						ภก ฉ
ษ๘ หฃช					ฐๆ ๖ฯ	๐า		ห	5
ป									
					3				
ไ,ท					4	ง	7		
ฆบ	น			ไ					ๅ26 1าะ

ตารางที่ 1 รอบที่ 20 ขนาดโครงข่าย 10 x 10 โหนด ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว

๒	ฒ	ญ		ฐพ	ศ	ฺ	ฑช	ฃไ	ไ
ฟ		พ				ผ	ย		
ฎฏ		๕		ณฒ	ฮ	ต	๑	ท ห	ฆ
		๗				๓	ล		บน
ฝ		๘		ป				ฦ	มข
		ย		ไ		ๅร			
ส๔		อ ฤ	ฅด ก			๐		35	4
		6	0ถ	ภก ฉ	ๆ		7		ง
ฯ		ธ					2ว		ะ
ธ		9 จ		๖	ฯ		เ	1	า

ตารางที่ 2 รอบที่ 60 ขนาดโครงข่าย 10 x 10 โหนด ตัวอักษรไทยที่ใช้ฝึกมี 75 ตัว



ตารางที่ 3 รอบที่ 60 ขนาดโครงข่าย 25x25 โหนด,
* โหนดที่ไม่มีตัวอักษรใดตกเลย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

๒	๒	๒	*	*	ฐ	ฐ	*	ท	พ	ฬ	*	A	ส	พ	ฬ	พ	พ	*	ฎ	ฎ	ฎ	ฎ	ฎ	ฎ
๒	๒	๒	ฐ	ฐ	*	*	พ	ฬ	*	ส	*	*	พ	*	*	*	ฎ	ฎ	ฎ	*	*	*	ฝ	*
*	*	*	*	*	ฐ	ฐ	*	พ	*	*	*	ฮ	*	*	*	ฟ	*	ฎ	ฎ	ฎ	*	ฝ	ฝ	ฝ
ญ	*	ฒ	ฒ	*	*	*	*	*	*	๙	ฮ	ฮ	*	ฟ	ฟ	*	*	*	*	ฎ	*	ฝ	ฝ	*
ญ	*	ฒ	ฒ	*	ณ	ณ	ณ	*	*	*	*	*	*	*	*	*	๕	*	*	*	*	ฝ	ฝ	ฝ
ญ	*	ฒ	ฒ	*	*	*	ฆ	*	ธ	*	ส	ศ	ล	*	๗	*	๕	๕	๔	*	*	*	*	*
ญ	ญ	*	*	ฒ	ฑ	ฑ	*	ฆ	*	*	*	*	*	*	*	*	๕	*	๔	*	อ	ย	อ	ย
*	*	ณ	ณ	*	ฑ	*	*	*	๓	๓	๓	*	๗	๗	๗	*	*	๔	๔	*	ย	*	*	*
ป	*	ณ	ณ	*	*	ฑ	ฑ	*	๓	๓	๓	*	๗	๗	*	๘	*	๔	๔	๔	*	ล	*	B
ป	*	*	*	*	ซ	*	*	*	*	*	*	*	*	*	๙	*	๘	*	๔	๔	*	ฤ	*	C
ป	*	D	ท	*	*	*	ข	ข	*	8	8	*	*	๙	๙	*	๘	๘	*	*	*	*	E	ค
ป	*	ท	F	ม	ย	*	ฆ	ข	*	*	8	*	*	๙	๙	*	๘	*	ธ	G	ฉ	*	H	J
*	*	ย	*	ย	*	*	ข	ข	ข	*	8	*	๙	๙	*	*	*	ย	จ	*	*	E	0	
บ	บ	ท	*	*	ห	*	ซ	*	ฆ	*	6	*	*	*	4	L	จ	ฉ	M	*	N	P	Q	R
บ	บ	*	ย	ม	*	*	ซ	*	S	*	6	6	6	*	*	*	*	2	จ	*	S	ล	T	อ
บ	น	ห	ห	U	น	*	ข	*	*	*	*	*	*	*	ร	ร	*	5	5	*	C	W	0	R
น	ห	ม	*	*	น	*	ข	*	6	6	ฯ	Y	*	ร	Z	*	*	5	*	*	ๆ	ค	0	อ
ห	a	น	ม	ม	*	*	*	*	*	*	*	ข	*	*	*	*	5	*	b	ๆ	ๆ	ฯ	0	c
*	*	*	*	*	*	ข	ซ	ข	ข	ข	ซ	ข	*	3	d	3	5	*	Y	b	*	*	*	*
ฎ	g	ค	ฉ	๑	*	ซ	ซ	ซ	ซ	ซ	*	*	*	*	2	*	*	*	h	ๆ	ฯ	*	3	ว
j	*	ฉ	k	*	ผ	*	ซ	*	*	ซ	*	๖	๘	๖	๖	*	2	*	*	*	*	*	ว	7
ผ	*	*	*	*	ผ	*	ซ	ซ	*	*	*	๖	*	*	*	*	*	*	m	ะ	ะ	7	า	า
*	*	โ	ไ	*	*	*	*	*	9	9	*	๖	*	n	n	า	า	*	*	*	ะ	*	า	*
Z	โ	q	ไ	*	r	ไ	ไ	*	*	*	*	*	*	*	ร	*	*	5	*	*	*	*	ะ	ะ
ฤ	โ	โ	โ	r	ไ	ไ	ไ	*	9	*	๑	๐	ฉ	๐	ว	7	*	2	*	m	l	ι	m	า

* เป็นโหนดที่ไม่มีตัวอักษรใดตกเลย.

A:ศต B:ฦภกลถ C:ฦภ D:ทพ E:ฤถ F:ชบ G:จธ H:ตค J:คดค L:4ง N:กดค P:ฦๆ Q:อค R:ฅกค S:กฤถ T:อตตคค U:ทห W:กฤ Y:ภก Z:โๅ a:มน b:อ๑ c:คกถ d:ว7 g:อฤ h:อก k:คด m:ι l n: 4ง q: โไ r: ไไ

ตารางที่ 4 รอบที่ 60 ขนาดโครงข่าย 25x25โหนด ฝึกโดยตัวอักษร 75 แบบรวม 1500ตัว

ฝึกโครงข่ายใหม่จนกว่าจะแยกแยะตัวอักษรได้หมด ทำให้ต้องเสียเวลามาก

6. ปัญหาการใช้ SOM แยกตัวอักษร

เมื่อฝึกด้วยตัวอักษร 75 แบบ แบบละ 1 ตัว โดยโครงข่ายมีขนาด 10x10 จะไม่สามารถแยกแยะตัวอักษรได้เนื่องจากโครงข่ายมีขนาดเล็กเกินไป การปรับน้ำหนักจะต้องปรับทั้งโหนดชนะและโหนดข้างเคียง ทำให้โหนดข้างเคียงบางโหนดมีการปรับน้ำหนักอยู่ตลอดถึงแม้จะเป็นโหนดที่เรียนรู้สมบูรณ์แล้วก็ตาม จึงทำให้โครงข่ายไม่เสถียร(stable) คือ ค่าน้ำหนัก ถูกปรับไปปรับมา (oscillate) ตลอด ดังนั้นโหนดนี้จึงไม่มีตัว อักษรใดมาตกเลย นอกจากนี้บางโหนดจะมีตัวอักษรหลายแบบมาตกขณะที่บางโหนดไม่มีตัวอักษรใดมาตกเลยเนื่องจากโหนดในโครงข่ายมีตำแหน่งคงที่ตลอดทำให้บางตำแหน่งอาจไม่เหมาะสมที่จะกระจายตัวอักษรไปตก จึงทดลองเพิ่มโหนดขึ้นไปเรื่อย ๆ จนเมื่อขนาดโครงข่ายเป็น 25x25 จึงสามารถแยกแยะตัวอักษรทั้งหมดได้ถูกต้อง โดยโหนดที่มีตัวอักษรตกมีเพียง 75 โหนด ส่วนโหนดที่ไม่มีตัวอักษรใดตกเลยมี 550 โหนด

ทดลองฝึกด้วยชุดฝึกที่สอง ที่มีตัวอักษร 1500 ตัว โดยโครงข่ายมีขนาด 25x25 โหนด บางโหนดจะมีตัวอักษรมาตกหลายแบบและบางโหนดไม่มีอักษรใดตกเลย ดังในตารางที่ 4 การเพิ่มจำนวนตัวอักษรที่ใช้ฝึกมากขึ้นจึงจำเป็นต้องเพิ่มขนาดโครงข่ายให้ใหญ่ขึ้นด้วย

7. สถาปัตยกรรม Adaptive SOM (ASOM)

ASOMเสนอโดยBernd Fritzke[1]จะมีโหนดเชื่อมต่อกันดังตัวอย่างในรูปที่5(b)การเชื่อมต่อของโหนดจะเป็นสามเหลี่ยมดังรูปที่5(a) ซึ่งเรียกว่า simplex เมื่อมีการลบหรือเติมโหนดใหม่เข้าไปจะต้องรักษาลักษณะของ simplex ไว้ (SOM จะมีจำนวนโหนดข้างเคียงและรูปร่างคงที่)

รูปที่ 5 (a)แสดงโครงสร้างพื้นฐานของโครงข่าย (b)แสดงตัวอย่างโครงข่าย

การฝึกASOMมีการปรับน้ำหนักตามสมการที่2 เช่นเดียวกับSOM แต่สิ่งที่ต่างคือ โหนดข้างเคียงเป็นโหนดที่เชื่อมกับโหนดชนะโดยตรง(ΔC และ ΔR เท่ากับ 1) และค่าอัตราการเรียนรู้(α)จะคงที่ตลอดการฝึก และเพิ่มสัญญาณการตกให้แก่โหนดที่ข้อมูลตกและลดสัญญาณการตกของทุกโหนดในโครงข่ายลงดังสมการที่ 3,4 ตามลำดับ หลังจากฝึกไปช่วงหนึ่งจะพิจารณาว่าโหนดใดมีข้อมูลเข้าตกมากที่สุด จะเพิ่มโหนดใหม่เข้าไประหว่างโหนดนั้น(โหนดcmax)กับโหนดข้างเคียงที่มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

น้ำหนักห่างจากโหนดนั้นมากที่สุด (โหนดfar) ซึ่งคำนวณได้จากสมการที่ 5,6 ตามลำดับ กำหนดค่าน้ำหนักเริ่มต้นให้โหนดใหม่ดังสมการที่7 และจำนวนตัวอักษรที่ตกเริ่มต้นให้แก่โหนดใหม่ได้จากสมการที่ 8,9,10 โดยกำหนดโหนดข้างเคียงของโหนดใหม่ก็คือโหนดcmaxและโหนดfar รวมทั้งโหนดทุกโหนดที่อยู่ในsimplexเดียวกับโหนดcmaxและโหนดfar ดังตัวอย่างในรูปที่ 6

การลบจะเลือกลบโหนดที่มีค่า normalized probability density ที่คำนวณจากสมการที่ 11, 12 น้อยกว่าค่าเทรสโฮลด์ที่กำหนด(threshold, $\eta = 0.009$) หลังการลบโหนดที่ไม่ได้อยู่ใน simplex ใดเลยต้องถูกลบออกด้วยดังตัวอย่างในรูปที่7 และสามารถสรุปขั้นตอนการฝึกได้ดังในอัลกอริทึมที่ 2

$$\Delta\tau_{winner} = 1 \quad (3)$$

$$\Delta\tau_c = -\beta\tau_c \text{ ,}(c \in A, \beta=0.05) \quad (4)$$

τ_{winner} จำนวนตัวอักษรที่ตกในโหนดชนะ(winner)

τ_c จำนวนตัวอักษรที่ตกในโหนด c

A เซตของโหนดทั้งหมดในโครงข่าย

$$\tau_{c\max} = \underset{i\in A}{MAX}(\tau_i) \quad (5)$$

$$\left\|W^{c\max} - W^{far}\right\| = \underset{i\in n_{c\max}}{MAX}\left\|W^{c\max} - W^{i}\right\| \quad (6)$$

$$W^{new} = \frac{\left(W^{c\max} + W^{far}\right)}{2} \quad (7)$$

$$\tau_{new} = -\sum_{c\in N_{new}} \Delta\tau_c \quad (8)$$

$$\Delta\tau_c = \frac{\left|F_c^{new}\right| - \left|F_c^{old}\right|}{\left|F_c^{old}\right|}\tau_c, \quad c \in N_{new} \quad (9)$$

$$\left|F_c\right| = \left(\tilde{l}_c\right)^n$$

$$\tilde{l}_c = \left(1/card(N_c)\right)\sum_{i\in N_c}\left\|W_c - W_i\right\| \quad (10)$$

W เวคเตอร์น้ำหนัก

new โหนดใหม่

cmax โหนดที่มีค่าจำนวนนับมากสุด

far โหนดที่มีค่าผลต่างกำลังสองน้ำหนักแตกต่างจากโหนด cmax มากสุด

τ_{new} ค่าเริ่มต้นของจำนวนตัวอักษรที่ตกโหนดใหม่

$\Delta\tau_c$ ผลต่างระหว่างค่าvoronoi region ของโหนด c ก่อนการเพิ่มโหนดใหม่และค่า voronoi region ของโหนด c หลังการเพิ่มโหนดใหม่

N_c โหนดข้างเคียงโหนด c

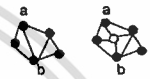
card(N_c) จำนวนโหนดข้างเคียงโหนด c

n มิติของ Voronoi Region ในที่นี้เท่ากับ 2 มิติ

$\tilde{l}_c$ ค่าเฉลี่ยผลต่างของน้ำหนักระหว่างโหนด c และโหนดข้างเคียงโหนด c

$\left|F_c^{old}\right|$ ค่า Voronoi Region ของโหนด c ก่อนการเพิ่มโหนดใหม่

$\left|F_c^{new}\right|$ ค่า Voronoi Region ของโหนด c หลังการเพิ่มโหนดใหม่



(ก) (ข)

รูปที่ 6 แสดงการปรับโครงสร้างโครงข่ายประสาทหลังการเพิ่มโหนดใหม่

●แทนโหนดเก่า ○แทนโหนดใหม่ที่เพิ่มเข้าไป

a โหนดที่มีค่าจำนวนนับมากที่สุด

b โหนดที่น้ำหนักต่างจากโหนด a มากที่สุด

(ก,ข)โครงข่ายประสาทก่อนและหลังการเพิ่มโหนด

$$\tilde{p}_c = h_c / \left|F_c\right|$$

$$h_c = \tau_c / \sum_{i\in A}\tau_i \quad (11)$$

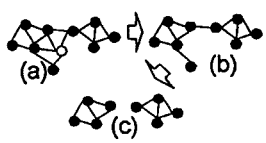
$$\hat{p}_c = \tilde{p}_c \sum_{c\in A}\left|F_c\right| \quad (12)$$

h_c relative signal frequency ของโหนด c

$\tilde{p}_c$ probability density density ของโหนด c

$\hat{p}_c$ normalized probability density ของโหนด c

A เซตของโหนดทั้งหมดในโครงข่าย



รูปที่ 7 แสดงการปรับโครงสร้างโครงข่ายประสาทหลังการลบโหนด

●แทนโหนดในโครงข่าย ○แทนโหนดที่จะลบ

(a)โครงข่ายก่อนการลบ(b)โครงข่ายระหว่างการลบ

(c) โครงข่ายหลังการลบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.อัลกอริทึมที่ 2 ใช้ฝึก ASOM

step 0 initial weight, learnig rate of winner and neighbor node and number of node = 3 that all nodes connect to each other to be a triangular (simplex)

Step 1 select only one pattern of each character in training set to be inputs

Step 2. while classify False , do step 4-11

Step 4. for each training input I do step 5-8

Step 5. apply input to input node

Step 6. find winner node (eq. 1)

Step 7. update weight of winner and neighborhood node(eq. 2)

Step 8 increase signal counters of winner(eq.3) and decrease all signal counters (eq.4)

Step 9 insert new node

9.1 find max signal node

9.2 find node that weight is max distance from weight of max signal node(eq. 3)

9.3 insert new node between max signal node and max distance node

9.4 initial signal of new node (eq.7)

9.5 initial weight of new (eq. 6)

9.6 initial neighborhood node for new node

9.7 update neighbor of new_node's neighbor

Step 10 delete node

10.1 calculate $\hat{p}_c$ (eq. 11)

10.2 delete node which $\hat{p}_c < \eta$, (η=0.09)

Step11 if each node is allocated by same Thai character , classify TRUE

9. การทดลอง adaptive SOM

ทดลองฝึกโครงข่ายตามอัลกอริทึมที่ 2 ชุดตัวอย่างที่ใช้ฝึกมีตัวอักษร 75 แบบแบ่งชุดการฝึกเป็น 3 ชุด ชุดแรกมีตัวอักษรแบบละ 1 ตัว ชุดสองมีตัวอักษรแบบละ 20 ตัวโดยแบบตัวอักษรเดียวกันจะอยู่ต่อเนื่องกันทั้ง 20 ตัว ชุดสามมีตัวอักษรแบบละ 20 ตัวโดยจะเรียงสลับแบบตัวอักษรกันตัวต่อตัวจนครบ 75 แบบแล้วเรียงสลับเช่นเดิมจนครบทุกตัวอักษรในชุดการฝึก (ชุด 2, 3 มี 1500 ตัว)

ทดลองฝึก 3 วิธี คือ วิธีแรกการฝึกด้วยตัวอักษรในชุดสองอย่างเดียว วิธีที่สองทดลองโดยการฝึกด้วยตัวอักษรชุดแรกจนสามารถแยกแบบอักษรได้หมดก่อน แล้วจึงฝึกด้วยตัวอักษรชุดสองต่อจนกว่าจะแยกกันได้หมดทุกตัวจึงสิ้นสุดการฝึก วิธีที่สามฝึกด้วยตัวอักษรชุดสาม และได้ทดลองกำหนดค่าอัตราการเรียนรู้ของโหนดชนะ และโหนดข้างเคียง (α_{win}, α_{neigh}) ตามลำดับ 6 ชุด ดังนี้ (0.1, 0.04),, (0.03, 0.001), (0.06, 0.002), (0.06, 0.00085), (0.06, 0.0008), (0.06, 0.0007)

ทดสอบประสิทธิภาพการจดจำด้วยตัวอักษร 4 ชุด รวม 15,000 ตัว คือ ตัวอักษรที่พิมพ์จากเครื่องเลเซอร์ที่ใช้ความสว่างในการสะแกน (scan) 100 และ 50 เปอร์เซ็นต์ 2 ชุด และตัวอักษรที่ถ่ายเอกสารจากกระดาษที่พิมพ์ด้วยเลเซอร์นี้เพื่อเพิ่มสัญญาณรบกวนให้กับตัวอักษรที่ใช้ทดสอบ อีก 2 ชุด รวมเป็น 4 ชุด

10. ผลการทดลอง adaptive SOM

จากการฝึกด้วยอัลกอริทึมที่ 2 โดยจำนวนโหนดเริ่มต้น 3 โหนด เมื่อฝึกตัวอักษรในชุดฝึกครบทุกตัวแล้วจึงเพิ่มและลบโหนด1ครั้ง ในระหว่างการฝึกโครงข่ายจะมีจำนวนโหนดประมาณ 30โหนด การเพิ่มและลดโหนดจะวนไปวนมาคือ โหนดที่ถูกเพิ่มเข้าไปมักจะถูกลบออกเพราะไม่มีตัวอักษรมาตก เมื่อลบออกแล้วจะเพิ่มใหม่อีกก็จะถูกเพิ่มในตำแหน่งเดิมนี้อีกซึ่งก็จะถูกลบอีกเช่นเดิม

11.ปัญหาและวิธีแก้ในการฝึกadaptive SOM

1. จากปัญหาการเพิ่มและลดโหนดที่วนไปวนมาตลอด จึงปรับเวลาในการเพิ่มและลดโหนดโดยฝึกครบทุก 3 รอบแล้วจึงมีการเพิ่มและลดโหนด (1 รอบคือการฝึกชุดฝึกครบทุกตัวอักษร 1 ครั้ง) เพื่อค่าน้ำหนักจะได้เสถียรมากพอที่จะมีตัวอักษรมาตกจากการทดลองนี้จำนวนโหนดระหว่างการฝึกจะเพิ่มถึงประมาณ 50 โหนดก็จะติดปัญหาการเพิ่มและลดโหนดที่วนไปวนมาเหมือนเดิม แสดงให้เห็นว่ามีการเพิ่มและลดโหนดในตำแหน่งที่ไม่เหมาะสม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การเพิ่มโหนดในตำแหน่งที่มีตัวอักษรตกมากที่สุดในบางกรณีก็ไม่เหมาะสม เช่น กรณีโหนดที่มีตัวอักษรตกมากที่สุดมีตัวอักษรเพียงแบบเดียวตก การเพิ่มโหนดที่ตำแหน่งนี้มักจะทำให้ตัวอักษรของโหนดนี้ซึ่งแยกแยะได้แล้วแยกออกมาตกยังโหนดใหม่แทนที่จะเป็นตัวอักษรจากโหนดที่ยังแยกแยะไม่ได้ ทำให้การเพิ่มโหนดใหม่ไม่ได้ช่วยให้โครงข่ายแยกแยะตัวอักษรได้ดีขึ้นเลย จึงแก้ปัญหาด้วยการเพิ่มโหนดใหม่ในตำแหน่งโหนดที่มีจำนวนแบบของตัวอักษรมาตกมากที่สุด ซึ่งทำให้มีการเพิ่มโหนดในตำแหน่งที่เหมาะสมขึ้น

3. การลดสัญญาณการตกของโหนดในโครงข่ายทั้งหมดลงอีก β เท่าของสัญญาณเดิมในทุกครั้งที่มีการป้อนตัวอักษรเข้าฝึก ทำให้โหนดที่มีตัวอักษรตกในช่วงแรกของการฝึกแต่ในช่วงหลังไม่มีตัวอักษรตกเลย มีค่าสัญญาณการตกลดลงเรื่อย ๆ ทำให้ค่า normalised probability density ($\hat{P}_c$)ซึ่งแปรผันตรงกับจำนวนแบบอักษรตกที่ในโหนดนั้น ลดลงน้อยกว่า 0.09 โหนดจึงถูกลบออกแม้จะเป็นโหนดที่มีสามารถแยกแยะแบบตัวอักษรได้แล้วก็ตาม จึงปรับไม่ให้มีการลดสัญญาณของโหนดในโครงข่ายในระหว่างการฝึกเพื่อที่ลำดับการป้อนตัวอักษรเข้าฝึกจะได้ไม่มีผลต่อสัญญาณการตก เมื่อปรับเช่นนี้แล้วโหนดที่มีแบบตัวอักษรตกหลายแบบจะมีค่าสัญญาณการตกสูงขึ้นเรื่อย ๆ เพื่อป้องกันไม่ให้ค่าสัญญาณการตกเกิดโอเวอร์โฟว์(overflow) เมื่อฝึกครบ 1 รอบจะกำหนดค่าสัญญาณการตกของแบบตัวอักษรในแต่ละโหนดใหม่ให้เป็นศูนย์

4. โหนดใหม่ที่เพิ่มบางครั้งก็ไม่มีตัวอักษรใดตกเลย การกำหนดสัญญาณการตกเริ่มต้นให้โหนดใหม่ด้วยการคำนวนในสมการที่ 8 ซึ่งเป็นค่าที่ได้จากการคาดคะเนไม่ใช่ค่าที่แท้จริง หากค่านี้สูงจะมีผลให้มีการเพิ่มโหนดใหม่ในตำแหน่งนี้อีกซึ่งเป็นตำแหน่งที่ไม่เหมาะสม

5. กรณีที่มีตัวอักษรที่ใช้ฝึกจำนวนมากทำให้มีโหนดที่มีจำนวนตัวอักษรตกน้อยมีค่า normalised probability density น้อยกว่า 0.09 ได้ โหนดนี้ไม่ควรจะถูกลบเพราะมีตัวอักษรตก จึงได้ปรับให้ลบโหนดที่ไม่มีตัวอักษรใดตกเลย

6. การลบโหนดออกจากโครงข่ายจะต้องปรับโครงสร้างของโครงข่ายใหม่โดยโหนดที่เหลือหลังการลบจะต้องอยู่ในโครงสร้าง simplex หากโหนดใดไม่อยู่ในโครงสร้าง simplex จะต้องถูกลบออกด้วยแม้ว่าโหนดนี้จะแยกแยะตัวอักษรได้แล้วก็ตาม ในรอบถัดไปก็มีการเพิ่มโหนดใหม่ในตำแหน่งเดิมนี้อีก และหากโหนดนี้เชื่อมกับโหนดที่ไม่มีตัวอักษรใดตกอีกอาจต้องถูกลบออกอีก จึงมีการลบและเพิ่มโหนดวนไปวนมาตลอดและไม่สามารถที่จะแยกแยะตัวอักษรได้ จึงไม่ให้มีการลบโหนดในระหว่างการฝึกในช่วงแรก จะลบโหนดที่ไม่มีตัวอักษรใดตกก็ต่อเมื่อสามารถแยกแยะตัวอักษรทุกแบบได้ถูกต้องทั้งหมด และเพื่อไม่ให้โหนดทีแยกแยะได้แล้วถูกลบออกไปด้วยในการลบจึงไม่จำเป็นที่ทุกโหนดจะต้องอยู่ในโครงสร้าง simplex

การแก้ปัญหาตามขั้นตอนที่กล่าวข้างต้นสามารถสรุปขั้นตอนได้ดังอัลกอริทึมที่ 3

12. อัลกอริทึมที่ 3ที่ใช้ฝึก ASOM

step0 initial weight, α_{win} , α_{neigh} and number of node = 3
that connect to each other to be a tritangular (simplex)
Step 1 select 1 character / pattern in training set to be inputs
Step 2. while classify False , do step 4-14
Step 4. for each training input I do step 5-8
Step 5. apply input to input node
Step 6. find winner node (eq. 1)
Step 7. update winner and neighbor node(eq. 2)
Step8 if Thai pattern has never allocated winner,
signal = signal+1
Step 9 find max-signal node (eq. 5)
Step10 find node that weight is max distance
from weight of max-signal node (eq. 6)
Step11 insert new node between max-signal node and
max-distance node
Step12 initial neighborhood node for new node, and
weight of new node (eq.7)
Step13 update neighborhood for neighborhood node
of new node
Step14 if each node is allocated by same-character
inputs, classify TRUE

เอกสารนี้มีลิขสิทธิ์ ... เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ... ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 8 แสดงการแมปตัวอักษรจากโครงสร้างหลายมิติเป็นโครงสร้างสองมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

13. ผลการทดลอง ASOM ตามอัลกอริทึมที่ 3

การทดลองที่	α_{win}	α_{neigh}	จำนวนรอบที่ใช้ฝึก	จำนวนโหนดทั้งหมด	จำนวนโหนดที่ใช้	พิมพ์เลเซอร์ (100 %)		พิมพ์เลเซอร์ (50 %)		ถ่ายเอกสาร (100 %)		ถ่ายเอกสาร (50 %)	
						ผิด (ตัว)	% ถูก	ผิด (ตัว)	% ถูก	ผิด (ตัว)	% ถูก	ผิด(ตัว)	% ถูก
1	0.1	0.04	337	339	193	71	99.53	596	96.03	80	99.47	784	94.77
2	0.03	0.001	501	503	252	69	99.54	540	96.40	72	99.52	807	95.77
3	0.06	0.002	332	335	214	26	99.83	481	96.79	69	99.54	631	95.79
4	0.06	0.00085	395	397	229	30	99.80	454	96.97	60	99.60	631	95.79
5	0.06	0.00075	389	391	223	18	99.80	454	96.97	55	99.63	588	96.08
6	0.06	0.0008	381	383	218	21	99.86	415	97.23	48	99.68	525	96.50
7*	0.06	0.0008	401	403	226	43	99.71	399	97.34	56	99.63	609	95.94
8*	0.06	0.0008	300	570	214	52	99.65	481	96.79	70	99.53	656	95.63

ตารางที่ 5 ผลการทดลองของ adaptive SOM

ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด(ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด(ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด(ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด(ตัว)
3	1	1	ค	ต	1	ฤ	ถ	4	า	ว	1
ค	ต	2	ต	ค	3	ล	ท	1	เ	เ	3
ต	ค	2	ต	ร	1	ฦ	ภ	5	เ	เ	2
จ	เ	1	ต	พ	1	ส	ธ	1	เ	เ	4
ฌ	ณ	1	ถ	ท	1	ห	ท	2	o	ย	1
ฏ	ฏ	2	ท	พ	1	ฯ	า	1	๖	6	2
ฏ	ฏ	1	ย	เ	1	ะ	ร	1	ธ	เ	1

ตารางที่ 6 แสดงตัวอักษรที่จดจำผิด โดยทดสอบจากตัวอักษร 75 แบบ แบบละ 200 ตัวที่สะแกนจากกระดาษถ่ายเอกสารจากกระดาษที่พิมพ์ด้วยเครื่องพิมพ์เลเซอร์ ความสว่าง (brightness) 100 %

ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด(ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด(ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด(ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด(ตัว)
5	ฐ	4	ฌ	ฒ	3	ย	ฮ	1	า	ว	10
ก	ภ	2	ฏ	ฏ	6	ย	เ	1	เ	เ	2
ก	ค	2	ฑ	ฒ	1	ร	ธ	30	เ	เ	6
ข	ช	7	ณ	ฌ	28	ฤ	ภ	1	เ	เ	32
ค	ต	26	ค	ต	42	ล	ส	1	เ	เ	1
ค	ศ	1	ต	พ	10	ว	เ	1	ๅ	ภ	2
ค	ด	1	ต	ค	2	ศ	ส	1	๓	ฒ	2
ค	ต	12	ต	ศ	5	ศ	พ	1	๔	๕	6
ค	ต	32	ท	พ	76	ศ	ฬ	3	๕	ร	1
ค	ค	1	ธ	ช	1	ษ	ธ	2	๖	เ	1
จ	ธ	1	ธ	ข	2	ษ	บ	2	๗	ง	1
จ	ฉ	8	น	บ	2	ส	ศ	2	๘	๒	1
ฉ	ข	1	น	ษ	1	ห	ฬ	13	๘	เ	1
ช	ซ	14	ฝ	ฟ	1	ห	พ	50	๘	ง	1
ซ	พ	1	ภ	ภ	1	ห	ท	1			
ซ	ฒ	9	ม	บ	4	ฬ	พ	4			
ซ	พ	9	ย	อ	4	ะ	ร	1			

ตารางที่ 7 แสดงตัวอักษรที่จดจำผิด โดยทดสอบจากตัวอักษร 75 แบบ แบบละ 200 ตัวที่สะแกนจากกระดาษถ่ายเอกสารจากกระดาษที่พิมพ์ด้วยเครื่องพิมพ์เลเซอร์ ความสว่าง (brightness) 50 %

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด(ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด(ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด(ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด(ตัว)
ค	ค	4	จ	ฉ	2	ค	ต	3	ห	พ	1
ไ	ไ	3	ไ	ไ	1	ไ	ไ	7			

ตารางที่ 8 แสดงตัวอักษรที่จดจำผิด โดยทดสอบจากตัวอักษร 75 แบบ แบบละ 200 ตัวที่สะแกนจากกระดาษที่พิมพ์จากเครื่องพิมพ์เลเซอร์ ความสว่าง (brightness) 100 %

ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)	ตัวอักษรที่ทดสอบ	จดจำผิดเป็น	จำนวนที่ผิด (ตัว)
ก	ภ	2	ค	ค	2	ท	พ	62	ห	พ	10
ก	ศ	1	ค	ด	1	น	ม	7	ห	พ	22
ก	ท	1	ช	ซ	12	น	ษ	2	ไ	ไ	18
ข	ช	11	ฆ	ฒ	5	น	บ	2	ไ	ไ	6
ฆ	ฒ	1	จ	ฉ	13	พ	ห	3	ไ	ไ	1
ค	ต	27	จ	ธ	2	พ	ท	54	๔	๕	1
ค	ต	7	ฎ	ฏ	1	บ	ม	1			
ค	ต	9	ณ	ฌ	32	ย	อ	1			
ต	ต	42	ด	ต	25	ร	ธ	29			

ตารางที่ 9 แสดงตัวอักษรที่จดจำผิด โดยทดสอบจากตัวอักษร 75 แบบ แบบละ 200 ตัวที่สะแกนจากกระดาษที่พิมพ์จากเครื่องพิมพ์เลเซอร์ ความสว่าง (brightness) 50 %

ตัวอักษร	เลเซอร์100%	เลเซอร์50%	ถ่ายเอกสาร100%	ถ่ายเอกสาร50%	ตัวอักษร	เลเซอร์100%	เลเซอร์50%	ถ่ายเอกสาร100%	ถ่ายเอกสาร50%
ก	-	4	-	1	ฤ	-	-	4	1
ข	-	11	-	7	ล	-	-	1	1
ฆ	-	1	-	-	ฦ	-	-	5	1
ค	-	27	-	59	ว	-	-	-	1
ต	-	45	2	32	ศ	-	-	-	5
ฒ	-	5	-	-	ษ	-	-	-	5
จ	2	15	1	9	ส	-	-	1	2
ฉ	-	-	-	1	ห	1	32	2	6
ช	-	12	-	14	ฬ	-	-	-	4
ซ	-	-	-	19	ะ	-	-	1	1
ฌ	-	-	1	3	า	-	-	1	10
ญ	-	32	-	28	เ	-	-	-	2
ฎ	-	1	2	6	ไ	3	18	3	6
ฏ	-	1	-	-	ใ	1	6	2	32
ฑ	-	-	-	1	ไ	7	1	4	1
ฒ	3	25	1	42	ๅ	-	-	-	2
ต	-	-	5	17	๓	-	-	1	-
ถ	-	-	1	-	๓	-	-	-	2
ท	-	62	1	76	๔	-	1	-	6
ธ	-	-	-	3	๕	-	-	-	1
น	-	11	-	3	๖	-	-	2	1
บ	-	1	-	-	๗	-	-	-	1
ฝ	-	-	-	1	๘	-	-	-	2
ภ	-	-	-	1	๙	-	-	1	1
ม	-	-	-	1	ๆ	-	-	1	-
พ	-	54	-	1	๐	-	-	1	-
ย	-	1	1	6	5	-	-	1	-
ร	-	26	-	30					

ตารางที่ 10 แสดงการเปรียบเทียบจำนวนตัวอักษรแต่ละแบบที่จดจำผิด เมื่อทดสอบจาก
กระดาษที่พิมพ์จากเครื่องพิมพ์เลเซอร์ ความสว่าง 100 และ 50 %
กระดาษซึ่งถ่ายเอกสารจากกระดาษที่พิมพ์จากเครื่องพิมพ์เลเซอร์ ความสว่าง 100 และ 50 %

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 8 ตัวอย่างภาพตัวอักษรที่ ASOM จดจำได้ เรียงตามลำดับดังนี้ ค,ศ,ด,ด,ห,โ

รูปที่ 9 ตัวอย่างภาพตัวอักษรที่ ASOM จดจำผิด เรียงตามลำดับดังนี้ ค,ค,ศ,ใ,ห,โ

จากการฝึกตามอัลกอริทึมที่3ด้วยค่า α_{win} , α_{neigh} เท่ากับ0.6,0.2ตามลำดับโดยใช้ตัวอักษรชุดแรกฝึก เมื่อฝึกไปประมาณ 1000 รอบก็ยังแยกแยะตัวอักษรทั้งหมดไม่ได้แต่แนวโน้มการแยกก็จะดีขึ้นตามจำนวนรอบของการฝึกจึงคาดได้ว่า จะแยกแยะตัวอักษรได้หากฝึกต่อไป แต่เนื่องจากต้องใช้เวลาฝึกนานและเมื่อแยกแยะตัวอักษรได้แล้วก็มีโหนดที่ไม่มีตัวอักษรใดเลยตกเหลือจำนวนมากจึงได้ยุติการฝึกการที่ต้องใช้โหนดจำนวนมากเช่นนี้ก็เนื่องจากค่า α_{win} , α_{neigh} มากเกินไป ทำให้ค่าน้ำหนักของโหนดชนะและโหนดข้างเคียงที่ถูกปรับในแต่ละครั้งเข้าสู่สมดุล(converge)ขณะนั้น(local optimal)เท่านั้น ไม่ได้เข้าสู่ค่าสมดุลของโครงข่าย (global optimal) เมื่อป้อนตัวอักษรเข้าฝึกอีก ค่าน้ำหนักก็จะถูกปรับไปปรับมา(oscillate) ค่าที่ได้จึงไม่เสถียร (stable) เพื่อที่ให้ค่าน้ำหนักปรับเข้าสู่สมดุลของโครงข่าย (global optimal) จะต้องปรับค่าน้ำหนักทีละน้อยคโดยการใช้ค่า α_{win}, α_{neigh} น้อย ๆ ดังแสดงในตารางที่ 5

เมื่อพิจารณาผลการทดลองที่ 1-6 ในตารางที่ 5 จะเห็นว่า การใช้ค่า α_{win} , α_{neigh} น้อย ๆ มีจำนวนรอบและจำนวนโหนดที่ใช้น้อยกว่าใช้ค่า α_{win} และ α_{neigh} สูง ๆ และ ประสิทธิภาพในการจดจำก็ค่อนข้างดี ซึ่งค่า α_{win} และ α_{neigh} ที่ทำให้ประสิทธิภาพการจดจำดีที่สุดคือ 0.6 และ 0.0008 ตามลำดับ (การทดลองที่ 6) โดยฝึก 383 รอบ (ประมาณ 6 ชั่วโมงบนเครื่อง hp 9000 / 827)และใช้ 218 โหนด ซึ่งแสดงภาพโครงข่ายที่ได้จากการแมปในรูปที่ 10 จะเห็นว่าตัวอักษรที่มีลักษณะคล้ายกันจะอยู่ในตำแหน่งที่ใกล้เคียงกัน ส่วนค่าα_{win} และ α_{neigh} ที่สูงหรือต่ำกว่านี้ (การทดลองที่ 1-5) ประสิทธิภาพในการจดจำก็จะลดลงเล็กน้อย

การทดลองที่ 6-7 ฝึกด้วยการป้อนแบบตัวอักษรเดียวกันเข้าฝึกอย่างต่อเนื่องกัน และการป้อนแบบสลับทั้งสองวิธีให้ประสิทธิภาพการจดจำใกล้เคียงกัน

จากการทดลองที่ 7-8 ทดลองด้วยการเพิ่มโหนดเมื่อฝึกครบ 1500 และ 75 ตัวตามลำดับ ประสิทธิภาพการจดจำก็ใกล้เคียงกัน การเพิ่มโหนดเมื่อฝึกครบทุก 75 ตัวใช้จำนวนรอบในการฝึกมากกว่า และมีโหนดที่ไม่มีตัวอักษรใดตกก่อนที่จะถูกลบหลังจากที่สามารถแยกแยะอักษรได้ทุกตัวแล้วจำนวนมากกว่า ในขณะที่มีจำนวนโหนดที่ใช้ในการจดจำไม่ต่างกันนักโดยการทดลองที่ 7 มีเปอร์เซ็นต์การจดจำสูงกว่าเล็กน้อย และมีจำนวนรอบที่ใช้ฝึกมากกว่า ในขณะที่การทดลองที่ 8 มีจำนวนโหนดที่ไม่ได้ใช้มากกว่า จากตารางที่ 6-10 ตัวอักษรที่มีลักษณะคล้ายคลึงกันมาก เช่น ด กับ ค จะมีโอกาสจำผิดได้มากกว่าตัวอักษรที่มีลักษณะแตกต่างจากตัวอักษรอื่น ๆ อย่างชัดเจน เช่น ฮ กับ ป โดยเฉพาะเมื่อตัวอักษรนั้นมีสัญญาณรบกวนมาก (เช่น ตัวอักษรที่สะแกนจากกระดาษที่ถ่ายเอกสาร)

14. สรุปผลการทดลอง

การฝึกโครงข่ายประสาทเทียมแบบ SOM โดยกำหนดค่าน้ำหนักเริ่มต้นแบบสุ่ม อัตราการเรียนรู้เริ่มต้นและสุดท้ายเท่ากับ 0.6 และ 0.001 ขอบเขตเริ่มต้นของโหนดข้างเคียงมีรัศมีที่วัดจากโหนดชนะออกไป5 โหนด หากกำหนดขนาดโครงข่ายเล็กเกินไปจะแยกแยะตัวอักษรไม่ได้ เนื่องจากโครงข่ายมีขนาดคงที่ตำแหน่งของบางโหนดจึงไม่เหมาะสมที่ตัวอักษรจะมาตก เมื่อเพิ่มขนาดโครงข่ายให้ใหญ่เพียงพอจะแยกแยะแบบตัวอักษรทั้งหมดได้ แต่การเพิ่มขนาดโครงข่ายทำให้จำนวน

เอกสารนี้มีลิขสิทธิ์ สงวนไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าน้ำหนักที่อยู่นอกขอบเขตของชุดข้อมูลฝึกเพิ่มตามด้วย จากการทดลองพบว่า การแยกแยะตัวอักษร 75 แบบแบบละ 1 ตัวด้วย SOM ต้องใช้ขนาดโครงข่าย 25x25 โหนดซึ่งโหนดที่มีแบบตัวอักษรตกมีเพียง 75 โหนดมีโหนดที่ไม่ได้ใช้ประโยชน์เลยถึง 88 เปอร์เซ็นต์ และเมื่อใช้โครงข่ายขนาด25x25และชุดข้อมูลฝึก1500ตัวก็จะแยกแยะตัวอักษรไม่ได้ ต้องเพิ่มขนาดโครงข่ายและเริ่มทดลองฝึกใหม่ซึ่งทำให้เสียเวลามาก แบบจำลอง SOM จึงไม่เหมาะที่จะนำมาประยุกต์ใช้ในการจดจำอักษรภาษาไทย

จากเหตุผลข้างต้นจึงใช้ adaptive SOM ที่มีความยืดหยุ่นกว่าในการเพิ่มและลดโหนดในระหว่างการฝึก โดยฝึกตามอัลกอริทึมที่ 2 แต่อัลกอริทึมนี้ก็ทำให้ติดปัญหาคือ มีการลบและเพิ่มโหนดในตำแหน่งที่แยกแยะตัวอักษรได้แล้ว จึงมีการปรับอัลกอริทึมดังนี้ คือ ให้เพิ่มโหนดในตำแหน่งที่มีสัญญาณการตกของแบบตัวอักษรสูงสุด และจะไม่มีการลดสัญญาณการตกในโหนดอื่น ๆ ที่ไม่มีตัวอักษรใดตกเพื่อลำดับการป้อนตัวอักษรเข้าฝึกจะได้ไม่มีผลต่อสัญญาณการตก และค่าสัญญาณการตกจะถูกกำหนดใหม่(reset) ให้เท่ากับศูนย์ทุกครั้งที่ฝึกตัวอักษรในชุดฝึกครบ 1 ครั้ง(ซึ่งเรียกว่าครบ 1 รอบ) และในช่วงแรกของการฝึกจะเพิ่มโหนดอย่างเดียวจนแยกแยะตัวอักษรได้หมดจึงลบโหนดที่ไม่มีตัวอักษรใดตกออก ซึ่งสามารถสรุปสิ่งที่ปรับเปลี่ยนไปได้ดังแสดงในอัลกอริทึมที่ 3 นอกจากนี้ได้ทดลองกำหนดค่าอัตราการเรียนรู้ของโหนดชนะ และโหนดข้างเคียงให้มีค่าต่าง ๆ กัน พบว่า ค่าอัตราการเรียนรู้ที่สูงเกินไป เช่น อัตราการเรียนรู้ของโหนดชนะและโหนดรอบข้างเคียงเท่ากับ 0.6, 0.2 ตามลำดับ ทำให้ใช้เวลาฝึกโครงข่ายนานเพราะค่าน้ำหนักถูกปรับไปปรับมา เข้าสู่ค่าเสถียรช้า จึงปรับอัตราการเรียนรู้ให้ลดลง จากตารางที่ 5 ค่าอัตราการเรียนรู้ของโหนดชนะและโหนดข้างเคียงที่เหมาะสมที่สุดในการนำมาประยุกต์ใช้ในการจดจำอักษรภาษาไทยในชุดการฝึกนี้คือ0.06 และ 0.0008 ตามลำดับ โดยใช้จำนวนรอบในการฝึก 381 รอบ ใช้ 218 โหนด และประสิทธิภาพในการจดจำตัวอักษรจากข้อมูลที่พิมพ์ด้วยเลเซอร์ ที่สะแกนที่ความสว่าง 100 % และ 50 % และจากข้อมูลที่ได้จากการถ่ายเอกสารจากข้อมูลที่พิมพ์ด้วยเลเซอร์ความสว่าง 100 % และ 50 % เท่ากับ 99.86, 97.23, 99.68 และ 96.50 เปอร์เซ็นต์ตามลำดับ

จากการป้อนแบบตัวอักษรเดียวกันต่อเนื่องกันเข้าฝึก กับการสลับตัวอักษรตัวต่อตัว พบว่าประสิทธิภาพในการจดจำใกล้เคียงกัน ลำดับการป้อนแบบตัวอักษรเข้าฝึกไม่มีผลต่อประสิทธิภาพการจดจำ

การเพิ่มโหนดทุกครั้งที่ฝึกครบ 1500 ตัวและเพิ่มโหนดทุกครั้งที่ฝึกครบ 75 ตัว มีประสิทธิภาพการจดจำใกล้เคียงกัน การฝึกครบ 1500 ตัวแล้วเพิ่ม 1 โหนดมีจำนวนโหนดก่อนการลบโหนด 403 โหนด ส่วนการเพิ่มโหนดทุกครั้งที่ฝึกครบ 75 ตัวมีจำนวนโหนดก่อนการลบโหนด 570 โหนด ซึ่งการลบโหนดจะลบเมื่อแยกแยะตัวอักษรได้แล้ว หากมีโหนดที่ไม่ได้ใช้ก่อนที่จะแยกแยะตัวอักษรได้หมดทำให้ต้องเสียเวลาการฝึกโหนดดังกล่าวโดยไม่ได้ผลการฝึกที่ดีขึ้นเลย จึงควรฝึกโดยการเพิ่มโหนดใหม่ทุกครั้งที่ฝึกครบ1 รอบ

16. บรรณานุกรม

1. Bernd Fritzke,(1994) Growing cell structures A self-Organizing network for unsupervised and unsupervised learning, Neural Network, pp 1441-1460 , Vol. 7, No-9
2. Jack M. Zurada, Artificial neural systems,West Publishing Company, 1992
3. Jame A.freeman and David, Neural Networks Algorithms Applications and Programming Techniques, Addison-Wesley , 1991
4. Laurene Fausett, Fundamentals of Neural Networks Architectures, Algorithms and Application, Prentice-Hall, 1994

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. รศ.ดร.ชม กิ้มปาน, " การรู้จำรูปแบบอักษรพิมพ์ภาษาไทยของคอมพิวเตอร์ ", ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง กรุงเทพฯ

6. ทวี เปรมรัตนชัย, "การทดสอบความรู้จำพยัญชนะภาษาไทยโดยวิธีโครงข่ายนิวรอนแบบ back-propagation ", วารสารคอมพิวเตอร์, สมาคมคอมพิวเตอร์แห่งประเทศไทย, หน้า 48 - 59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อผู้เขียน	นางสาวอัญชลี วานิชทวีวัฒน์
วันเดือนปีเกิด	23 มีนาคม 2512
สถานที่เกิด	จังหวัดหนองคาย
วุติการศึกษาระดับปริญญาตรี	วท.บ.(ชีวเคมีและชีวเคมีเทคโนโลยี)
สถานที่สำเร็จการศึกษา	คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่ ปีการศึกษา 2533
ประสบการณ์	2539 ผู้ช่วยที่ปรึกษาโครงการระบบงานคอมพิวเตอร์เพื่อการจัดเก็บภาษีของกรมสรรพากร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้